DISSERTATION

CLASSIFIED REGRESSION FOR SYSTEMS ENGINEERING

Submitted by

Robert Dockstader

Department of Systems Engineering

In partial fulfillment of the requirements

for the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Fall 2025

Doctoral Committee:

Advisor: Jim Adams
Gregory Marzolf
Steve Simske
Azer Yalin

ABSTRACT

CLASSIFIED REGRESSION FOR SYSTEMS ENGINEERING

Systems engineering is critical for the successful realization of today's complex technology-driven systems. Responsibilities range from operations enhancement and efficiency optimization to providing the necessary technical oversight to achieve system compliance, including on-time delivery, within allocated cost, and performance at or exceeding work scope throughout the system's life cycle. Commensurate with these responsibilities is the capability to design, assess, and mitigate the system triad components. The system evaluations are performed according to established standards and practices that use highly detailed models to calculate, predict, and forecast behavior. These approaches, which rely on an average understanding of the system design, often overgeneralize and lack precision.

To improve the system engineering resources for system prediction and forecasting, the method of classified regression is presented. Classified regression is a novel machine learning method that leverages the support vector machine architecture and combines regression and clustering. This stochastic methodology organizes and extracts patterns from small, unstructured datasets while demonstrating high-quality, accurate performance. Built on simplicity, the algorithm is a robust, interpretable alternative to conventional prediction algorithms. Its implementation approach avoids the black-box tendency of complex, non-linear, multivariate models, which often overfit and lack transparency. Classified regression has direct applications in healthcare, finance, marketing, and system development and monitoring.

ACKNOWLEDGEMENTS

I want to thank Dr. Jim Adams for his ongoing encouragement, support, and guidance. Our time spent together was a privilege.

I want to thank Dr. Steve Simske for his instruction, patience, and guidance.

I want to thank Dr. Gregory Marzolf and Dr. Azer Yalin for their support and insight during my committee's participation, as well as for their assistance throughout my educational journey.

I want to extend a special thank you to Ingrid Bridge, who consistently provided direction and assistance to meet the requirements and deadlines of the Systems Engineering program.

DEDICATION

I dedicate this work to my friends, family, teachers, and students who have supported and encouraged me throughout this process, tolerated me, and assisted me in achieving this and other goals during our time together.

I also dedicate this work to my mother and grandparents, who never wavered in their belief in my ability to achieve anything, provided unwavering support in overcoming life's hurdles, and had a profound influence as role models.

It is impossible to achieve anything in life without friends and mentors, including Joan Brickman, Levent Aydin, Joseph Kopesky, Rex Bordwell, Walter Kaden, Paul Jenkins, and Hillary Mongeau.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

AoA	Analysis of Alternatives
AI	Artificial Intelligence
AISD	Aerospace Space-based Instrument Database
AMCM	Advanced Mission Cost Model
ATP	Available-to-promise
BOL	Beginning of Life
BoM	Beginning of Mission design
BOLP	Beginning of Life Power
CAA	Computer-Aided Analysis
CAD	Computer Aided Design
CADRe	Cost Analysis Data Requirement
CAM	Computer Aided Manufacturing
CASTS	Crew and Space Transportation Systems Cost Model
CATE	Cost and Technical Evaluation
CBA	Cost-Benefit Analysis
CCER	Complexity-based Cost Estimating Relationship
CDF	Cumulative Density Function
CE	Cost Estimate
CEA	Cost-Effective Analysis
CEH	Cost Estimation Handbook
CER	Cost Estimating Relationship
CL	Confidence Level
CLT	Central Limit Theorem
CoBRA	Complexity-Based Risk Analysis
COSPAR	Committee on Space Research
CPt	Critical Point

CPI	Consumer Product Index
CPI-AUC	CPI with adjustment for All Urban Consumers
CR	Classified Regression
CSoS	Complex System of Systems
DAS	Defense Acquisition System
DoD	Department of Defense
dof	Degrees of freedom
DoL	Department of Labor
EAC	Estimate at Completion
EC	Exhaustive Classifier
EOM	End of Mission design
F	Feature
FC	Feature Combinations
F/FC	Feature and Feature Combinations
FY	Fiscal Year
GAO	General Accounting Office
GPRA	Government Performance and Results Act
ICA	Independent Cost Assessment
ICE	Independent Cost Estimate
IGCE	Independent Government Cost Estimate
ISS	International Space Station
JCIDS	Joint Capabilities and Development System
JCL	Joint Cost and Schedule Confidence Level
JPL	Jet Propulsion Laboratory
KDP	Key Decision Point
LCCE	Life Cycle Cost Estimate
LEO	Low Earth Orbit
MBSE	Model-Based System Engineering

MICM	Multivariable Instrument Cost Model
MISERLY	Method of Improved Cost Estimation from Historical Data
ML	Machine Learning
MOCET	Mission Operations Cost Estimation Tool
MOE	Measures of Effectiveness
MRO	Mars Reconnaissance Orbiter
NAFCOM	NASA Air Force Cost Model
NASCOM	NASA Cost Model
NASA	National Aeronautics and Space Administration
NICM	NASA Instrument Cost Model
NIM	NASA Interplanetary Mission
NN	Nearest Neighbor
NRE	Non-recurring Engineering
NSII	NASA New Start Inflation Index
OIG	Office of Inspector General
OLR	Ordinary Linear Regression
OMB	Office of Management and Budget
OSAM-1	On-orbit Servicing, Assembly, and Manufacturing
PA	Predictive Analytics
PBCM	Performance-Based Cost Modeling
PCC	Pearson Correlation Coefficient
PCEC	Project Cost Estimation Tool
PDF	Probability Density Function
PDR	Preliminary Design Review
PEBD	Planetary Exploration Budget Database
PCEPI	Personal Consumption Expenditures Price Index
PMCM	Planetary Mission Cost Model
POE	Program Office Estimate

PPBE	Planning, Programming, Budgeting, and Execution
PRICE	Parametric Review of Information for Costing and Evaluation
R&D	Research and Development
RCF	Reference Class Forecasting
RE	Recurring Engineering
ROM	Rough Order of Magnitude
RRW	Relative Risk Weighting
S&P 500	Standard and Poor 500
SCdry	Space Craft Dry (mass)
SEP	System Engineering Plan
SEER	Systems Evaluation and Estimation of Resources
SOSCM	Space-Based Optical Sensor Cost Model
SSCM	Small Satellite Cost Model
SLC	System Life Cycle
SOI	System of Interest
SSDB	Small Satellite Database
SVbus	Space Vehicle bus (mass)
SVM	Support Vector Machine
TCASE	Technology Cost and Schedule Estimating
TDCT	Total Development Cost in year of occurrence dollars (Then $)
TMC	Technical, Management, Cost
TMC	Total Mission Cost
TMCT	Total Mission Cost in year of occurrence dollars (Then $)
TPS	The Planetary Society
TRL	Technology Readiness Level
USCM	Unmanned Space Vehicle Cost Model
WBS	Work Breakdown Structure

Chapter 1: Introduction

Problem Statement

Modern systems are complex. Capturing and accurately applying legacy knowledge, technology, and products is critical for success. Artificial intelligence (AI) and machine learning (ML) are frequently deployed to facilitate information distillation and distribution. However, knowledge often comes at a cost. Modern implementations of AI and ML usually evolve into complex, multivariate, and non-linear applications that perpetuate the Black Box problem. Through intelligent automation, next-generation AI is overcoming traditional boundaries. Advancements in predictive analytics (PA) offer improvements to infrastructure management, system efficiency achievement, and optimization(Jonnalagadda, 2024). Based on general systems theory, like-systems share common patterns of behavior and structural similarities. (Tan, 2003). Understanding correlated behavior enables improvements in decision-making, control, analysis, diagnosis, planning, and other cognitive system engineering activities.

Research Objective

The objective of this research is to develop a high-accuracy system-level predictor of response attributes for complex systems of systems (CSoS). The predictor utility includes predicting known training data relationships and forecasting unmodeled data. Prediction occurs in single-predictor-to-response-attribute model relationships. Models are configured to estimate response performance with uncertainty of a single component or a complement of system triad elements.

The development and demonstration utilize a small, high-quality dataset developed for this thesis, the NASA Interplanetary Mission (NIM) dataset (Dockstader, 2025). Based on the results of this research, the classified regression predictor provides a simple, model-based system engineering

(MBSE) resource for rapid assessment, evaluation, and prototyping of candidate architectures using legacy data. The process to achieve this objective includes:

- Establishment of “the need” for this resource
- Literature review of current system modeling practice
- Definition of “the system” including the requirements, attributes, and resources that differentiate a system from another
- Establish a quality dataset of system attributes for model development and performance demonstration
- Develop a ML process that models and predicts system resource attributes
- Validate the modeling technique through an initial predictor performance assessment
- Verify initial predictor performance in accuracy, uncertainty, and sensitivity
- Demonstrate compatibility and validate the model’s algorithm with alternate data
- Apply the algorithm to a complement of system triad components and elements, demonstrating the feasibility as a system model tool

Research Overview

The 2014 DoD System Engineering Process guides the development of classified regression (CR), which guides this work (see *Figure 1*). The decomposition of the predictor process is accomplished in the literature review of Chapter 2. Operational needs are defined through an assessment of the funding system, and performance requirements are developed from the capabilities of current estimation methodologies. The review describes the criteria for system modeling in terms of data constraints and estimation performance, including prediction uncertainty. The design of the predictor is framed within modeling methodologies, system databases, and artificial intelligence and machine learning.

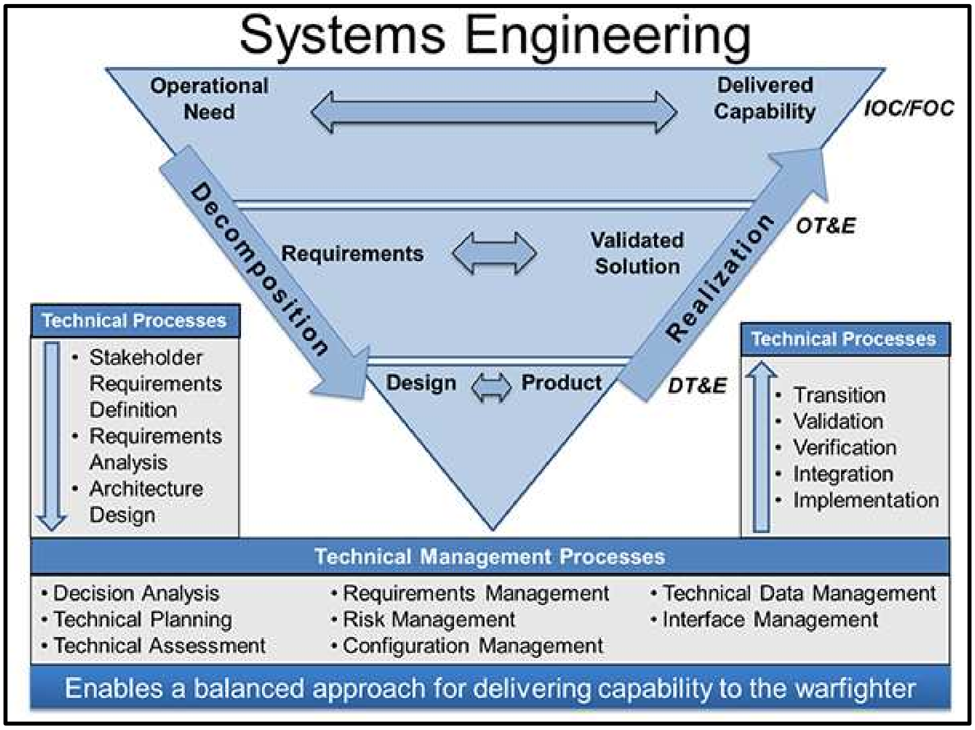


Figure 1: DoD System Engineering Process (Department of Defense, 2022)

The CR predictor process is realized in Chapters 3 and 4. Product development is described in the method of classified regression in Chapter 3. The assessment of CR as a predictor of complex systems is described in the data analysis and results section of Chapter 4. Validation includes demonstrating that CR is an attribute predictor of complex system attributes at the system triad element and component levels. Modeling capability encompasses the successful demonstration of CR for predicting components of the scope, cost, and schedule triad. Additional modeling demonstrates CR as a component of a system triad modeler based on independent predictors, summed predictors, and bootstrap aggregation. Independent CR validation is demonstrated using alternative datasets that explore CR modeling performance for civil construction project proposal margin management and computer-aided design to cost modeling. The delivered capability of CR is presented in a top-level triad component predictor for the development of the CSoS dataset.

Chapter 5 presents the research's conclusions, including initial algorithm lessons learned, predictor variable assessment, and the use of classification regression as an ML predictor of quantitative performance. Future work discusses the application of CR as a modeling component of a complete system prediction capability.

Problem Identification Summary

There is a need to improve cost credibility and system assessment throughout the program's life cycle for enterprises involved in procurement. Improvements are needed to:

1. Cost prediction – improvement in the process of evaluating and quantifying estimations and identifying associated shortcomings
2. Prediction process – establish a framework to minimize or eliminate data bias and inadequacies
3. Human decision points – create a method that minimizes human-in-the-loop decision points and relies on choice structure rather than judgment-based estimates
4. Data resource and normalization – develop a comprehensive dataset structure that normalizes the practice and decision rationale for dataset construction
5. Efficiency – improve the accuracy and speed of system estimation and trade comparison.
6. Self-preservation – eliminate bias and politics of the estimation process
7. Contingency management – improve the performance of the estimation process by revisiting contingency allocation and inclusion.

Research Questions

The ability to estimate, validate, and monitor CSoS performance is based on archiving, retrieval, and interpretation of legacy data. Estimation methodologies and their associated accuracies are limited by data visibility, record accuracy, and milestone structure standards. The research questions for this work focus on enhancing the quantification of system performance and improving the credibility of the estimation process by mitigating conventional shortcomings. The validation of the proposed methodology of system prediction raises the following research questions:

Question #1: Can a simple, classified regression predictor demonstrate performance exceeding conventional prediction standards?

Conventional prediction estimators rely on analogous, parametric, or summation methodologies or combinations thereof. While significant improvements have been realized in the current era of computational advances, CSoS applications continue to experience poor performance and program termination. Improvements in performance estimation —from planning through system realization and deployment —are critical for mission success in the current political environment of funding constraints.

The challenge of question #1 is to demonstrate the improved predictive performance and confidence of CSoS systems, as well as the limitations of small data in legacy systems.

Question #2: Can ML analytics successfully apply to CSoS small quantity datasets?

ML techniques are primarily applied to large volumes of normally distributed data. There is a growing application of ML techniques for smaller data sets containing fewer than 100 data elements. While limited in example, the CSoS is defined using dozens of parameters, hundreds of features, and thousands of components and design details.

The challenge of question #2 is to apply large-scale data analytics to a small-data application, avoid oversampling, and improve the current standard of "goodness of fit."

Question #3: Can normal distribution statistics be applied to small datasets?

ML often relies on large amounts of normally distributed data to verify and evaluate the statistical properties of normal distributions in the data. This establishes a quality measure of the data. When the population distribution is unknown, the Student's t-distribution is often used instead of the normal distribution. Both distributions exhibit a classical bell-shaped curve, where the Student's t-distribution deploys "thicker tails". In CSoS evaluations, the applied confidence level ranges from 70% to 85% in most assessments. This use of distribution for uncertainty assessment and application of the level of confidence remains within the bell curve of the distribution, avoiding the tail effect. An established practice for conventional estimators involves pruning ill-behaved data samples by labeling them as outliers. While a convention in large-scale data analytics, a CSoS is derived from an elaborate process, standardized procedures, and intensive verification that preclude the occurrence of outliers. Since CSoS are reproducible, they are not representative of atypical events or outcomes.

The challenge of question #3 is to demonstrate that prediction performance is benign with respect to distribution form, provided the prediction is well-behaved and distributed, and that all training elements are accurately predicted within the model without the isolation of outliers.

Question #4: Can heteroscedastic residual behavior be improved?

Current system modeling and prediction predominantly rely upon regression algorithms. This process requires that the prediction residual exhibit homoscedasticity. The use of small-sample datasets yields small residual responses, which, by nature, closely mimic the predictor variable. If the predictor variable is clustered or biased, the residuals will be clustered or biased. According to this thesis, clumping or lumping of predictors leads to heteroscedasticity in the limited-response residual. With small residual counts, these lumping biases the data, biases the understanding of the residuals, and produces plots that exhibit funnel or bulge shapes, which are identified as heteroscedastic.

The challenge posed by question #4 is to augment the small data residuals to meet the homoscedasticity requirement.

Question #5: Can ML predictors apply to non-cost data?

Algorithms are indifferent to the nature of the data being evaluated. The primary application of data is cost modeling and cost prediction. These applications are specifically developed using cost estimation relationships (CER) that adjust historical costs. The challenge of question #5 is to create a predictor that performs well for all elements of the system triangle, minimizes model degrees of freedom (*dof*) and employs a minimal number of predictors to avoid overfitting of the system.

Chapter 2: Literature Review

Introduction

The literature review is biased to the current state of development in system modeling and performance measurement standards of cost evaluation and estimation. The bulk of research focuses on cost-based application and assessment, with limited attention to topics specific to system scheduling or technical scope prediction. System scheduling is typically derived from the spending rate assessment, which, in turn, translates into task duration. System scope is based on cost estimates or budget allocations and verified through rigorous bottom-up summations.

System of Interest

The System of Interest (SoI) comprises all components within the system's lifecycle. The components include operational and enabling contributions to the system. The boundaries define the system's constraints, the interfaces provide input and output access to the system, and the partitions segregate the system into segments, subsystems, and components.

Funding System

The United States' fiscal path is unsustainable (Bureau of the Fiscal Service, 2025). The US national debt currently exceeds $35 trillion, with the individual's debt obligation exceeding $268,800 per taxpayer (US Debt Clock.org, 2025). While primary debt drivers are healthcare and interest on the debt, increasing debt generates budgetary pressure at all levels and across all government-dependent organizations reliant on government discretionary spending. An organization or program's current and prior performance are significant indicators of fiscal credibility and the ability to maintain proposed funding commitments.

Discretionary Spending

The highly debated topic within the U.S. Legislature was recently summarized in a 2023 NASA letter to the House Appropriations Committee. Current Administrator and former U.S. Senator Bill Nelson described the devastating impact proposed budget cuts would have on a revised 2024 NASA budget. Lawmakers were considering two cost-cutting approaches:

1. Roll back discretionary spending to 2022 levels. This act would reduce the 2024 NASA budget by $1.4 billion, less than the prior 2023 funding levels, and
2. Exempt defense spending from cuts. This act would require deeper reductions in non-defense agencies. This would result in a 22% reduction in the NASA budget from the previous 2023 funding level to approximately $19.8 billion.

Nelson concluded that the impact of either option would be “devastating and potentially unrecoverable” to the “objectives that the President and Congress have set for NASA.” He summarized that cuts would delay or cancel missions across all five Mission Directorates currently in development, and that resulting layoffs would impact 4,000 center and contractor personnel (Foust, 2023). This NASA example illustrates the effect of significant budget reductions throughout the technology and sectors that involve large, CSoS procurements.

Department of Defense

The Department of Defense (DOD) uses a decision-support triad comprising the Planning, Programming, Budgeting, and Execution (PPBE), the Joint Capabilities Integration and Development System (JCIDS), and the Defense Acquisition System (DAS). JCIDS' primary function is to identify and define capability requirements. DAS manages development and procures systems and equipment. The PPBE contribution is a calendar-driven process that operates within multiple overlapping cycles, as outlined in *Table 1*. Cost assessment is the responsibility of PPBE. Appropriation categories define both the purpose and the funding plan duration that must be considered for each proposed or extended initiative.

Table 1: PPBE Appropriation Categories

Category	Purpose	Funding Period
Operations and Maintenance (O&M)	Funds day-to-day expenses	1 year
Military Personnel (MILPERS)	Funds employment expenses for uniformed military personnel	1 year
Procurement	Purchase major capital equipment	3 years (5-year exception)
Research, Development, Test & Evaluation (RDT&E)	Basic scientific research and technology advancement	2 years
Military Construction	Infrastructure	5 years

In recent years, the DoD has reconsidered the PPBE process, enhancing agility and improving responsiveness to rapid technological advancements and evolving global threats. MBSE offers enhancements to current practices and methodologies for achieving these objectives. Rapid prototyping, assessment, and development are cornerstones for future success. The method of Classified Regression (CR) is an alternative predictor algorithm that offers improved performance, thereby satisfying the evolving needs of the PPBE process (Gortney, 2012).

The PPBE process consists of four elements:

- Planning provides a foundation for all subsequent resource decisions. Developed with stakeholders, planning translates broad goals and qualitative objectives into actionable plans. The plan identifies principal players, a defined timeline, an estimate of requirements and constraints, and a preliminary implementation concept and architecture. A product of planning includes a broad description of the effort required to achieve specific goals for enterprise reconciliation.

- Programming advances the preliminary concepts selected during the reconciliation process. Programming establishes a baseline for performance assessments and system trades. Mission utility and alternative concept architectures are also considered, with full life-cycle resource implications taken into account. An output of the programming phase is a System Request and an associated 5-year resource plan.

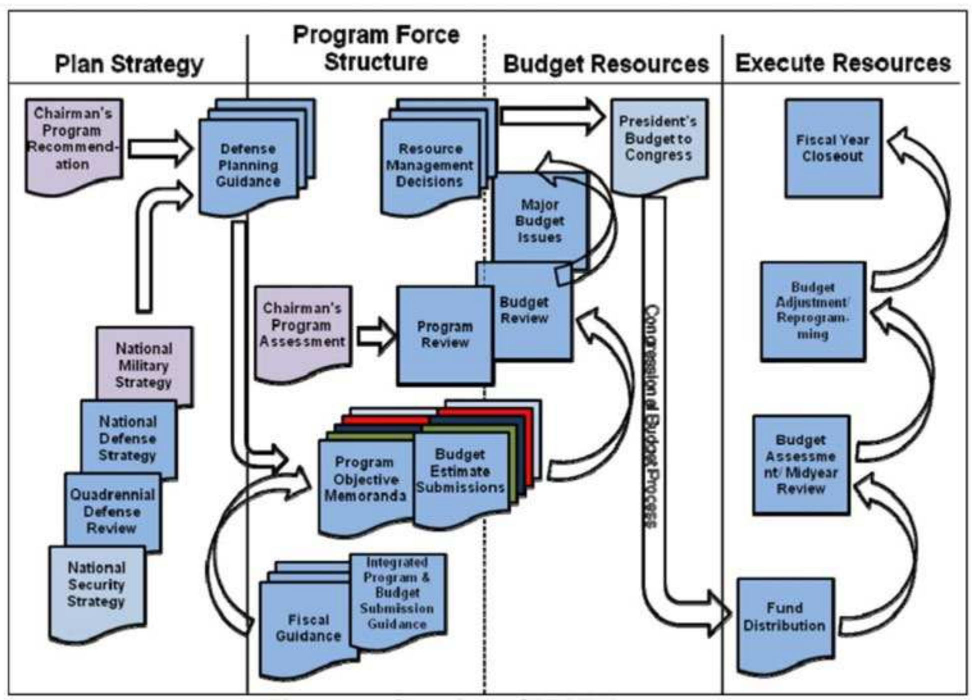


Figure 2: PPBE Process Flow (Gortney, 2012)

- Budget acceptance establishes a procurement commitment. The budgeting phase builds on the baseline concept, focusing on formulation, justification, and resource management control. Before submission to Congress, budget items develop detailed life-cycle plans, with scrutiny placed on the first two years of implementation. Budget approval initiates the execution phase.

- Execution phase implements funded programs, adjusting resources as required during realization. This phase ensures that plans and budgets are efficiently transformed into successful real-world applications.

Failed and Cancelled Programs

Wasteful spending occurs when high-cost, capital-intensive programs fail and are subsequently cancelled. These cancellations pose a threat to continued and future spending allocations. In March 2024, NASA announced the cancellation of the On-Orbit Servicing, Assembly, and Manufacturing 1 (OSAM-1) program. This decision was based on the findings of the October 2023 NASA Office of Inspector General (OIG) report. NASA OIG cited "continued technical, cost, and schedule challenges" along with "lack of a committed partner" as the root causes for the cancellation. NASA concluded that the prime contractor underestimated the project's scope and complexity, lacked an understanding of the technical requirements, and lacked the necessary expertise. The findings also noted that the fixed-price appropriation mechanism inhibited NASA's ability to incentivize the contractor and improve the program performance (Dvorsky, 2024). The cancellation terminated a $2 billion technology investment, resulting in little or no return.

The large number of program failures and cancellations is not unique to NASA. From 1997 through 2016, the DoD spent $62 billion on programs that were cancelled *Figure 3* (Clowney, 2016).

Program	Service	Cost ($billion)	Source
Future Combat Systems	Army	20.00	GAO (2014)
Joint Tactical Radio System	Army	11.00	Rodriquez (2014)
Comanche Helicopter	Army	5.90	GAO (2014)
nPOESS Satellite	Air Force	5.80	Reed (2011)
Airborne Laser	Air Force	5.00	Rodriquez (2014)
VH-71 Presidential Helicopter	Marines	3.30	GAO (2014)
Expeditionary Fighting Vehicle (EFV)	Marines	3.30	Reed (2011)
Transformational SATCOM (TSAT)	Air Force	2.90	Reed (2011)
Crusader	Army	2.20	GAO (2014)
Kinetic Energy Interceptor	Missile Defense Agency	1.30	Rodriquez (2014)
Advanced SEAL Delivery System	Navy	0.60	Reed (2011)
Armed Reconnaissance Helicopter	Army	0.50	Reed (2011)
Aerial Common Sensor	Army	0.19	GAO (2014)
CG(X) Next Generation Cruiser	Navy	0.20	Reed (2011)
CSAR-X	Air Force	0.20	Reed (2011)
TOTAL		**62.39**	

Note. CSAR = Combat Search and Rescue; NPOESS = National Polar-orbiting Operational Environmental Satellite System; SATCOM = Satellite Communications; SEAL = Sea, Air, and Land.

Figure 3: DoD Program Terminations and associated Costs, 1997-2016 (Clowney, 2016)

When CSoS programs are cancelled, significant capital investments are lost. The taxpayer-stakeholder realizes little benefit. The U.S. legislature is responsible for spending and budget decisions, with oversight from the executive. At the request of congressional committees and subcommittees, the General Accounting Office (GAO) is tasked with providing timely, fact-based, non-partisan reviews of:

- Organizational policies and procedures centered on spending and cost accountability,
- Program Cost Performance Assessment, and

- Transparency and cost savings services for U.S. taxpayers.

As a result, the GAO acts as the legislature's audit and accountability arm.

The GAO

Reliable cost estimation and expenditure management have long been challenges for government agencies (GAO-20-195G, 2020). In 1993, the Government Performance and Results Act (GPRA) introduced program performance reform to enhance program delivery and effectiveness by increasing accountability. The initiative was updated under the GPRA Modernization Act of 2010. The initiative's objective is to assess the achievement of program goals within the budgeted schedule and cost. The assessments occur throughout a program's life cycle. The GAO attributes program success to the credibility of budgets and plans (OFFICE OF MANAGEMENT AND BUDGET, 2024). The GAO currently employs annual performance reviews linked to Key Decision Point (KDP) milestones. This process serves as a control to prevent uncontrolled spending and maintain fiscal accountability *Figure 4*.

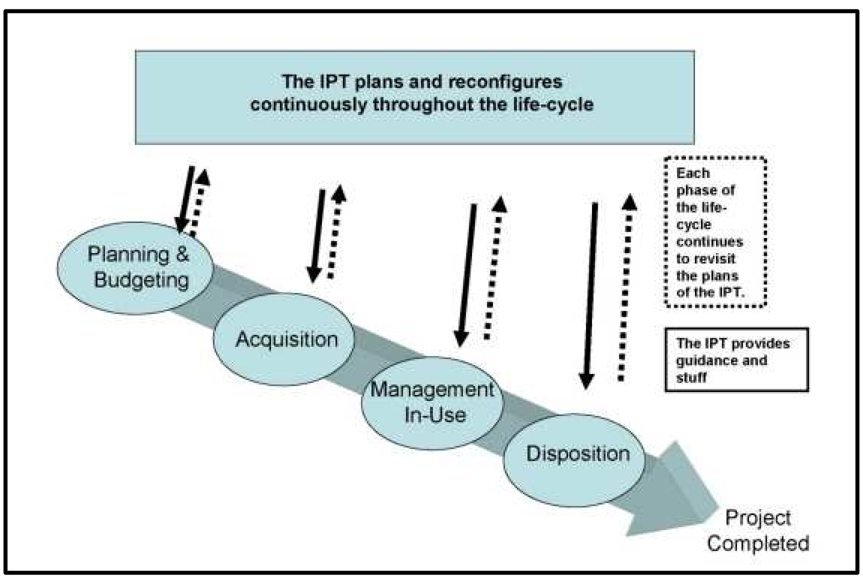


Figure 4: Capital Acquisition Planning Lifecycle

(OFFICE OF MANAGEMENT AND BUDGET, 2024)

Traditional system assessments evaluate the summation of cost elements. The estimation, prediction, and tracking of cost is a complex and arduous process. In early program phases, time constraints and resource availability hinder the development of reliable cost estimates. As a program develops, performance is realized, and estimates improve. Through greater data visibility, improved reporting accuracy, and refined milestone scheduling, methods enhance performance quantification and improve estimation credibility by mitigating shortcomings.

Credible cost estimates help ensure a program is less likely to fall short of the cost, schedule, and scope performance goals. Poor cost estimates arise from assumptions that aren’t well-defined, assumptions that aren’t supported by facts, gaps in comparison with similar programs, incomplete or ill-formed data, archaic methods or estimating techniques, and decisions that aren't grounded in evidence.

The GAO has recognized that the development of high-quality cost estimates directly depends on referencing historical databases (GAO-20-195G, 2020). GAO estimators rely on historical data to assign cost estimating relationships (CER). The GAO and procurement agencies allocate significant resources to collect and normalize references to historical data. The traditional funding challenges of cost estimation, resource availability, and time constraints on participation in cost evaluation exercises often limit the extent to which trade studies, sensitivity analyses, and uncertainty analyses can be conducted.

The infusion of new technologies poses significant risks to program scope and schedule goals, which ultimately drive cost. Proponents of solutions often present overly optimistic estimates based on unrealistic assumptions. In 2012, a NASA OIG report stated that a culture of optimism is beneficial when procuring state-of-the-art technology. However, it often leads management to overestimate the ability to deliver integrated products within baseline cost and schedule (Martin, 2012). In a competitive marketplace, suppliers usually overestimate their capabilities relative to competitors'.

System Modeling

The success of AI-driven system modeling is founded on modularity, scalability, security, interpretability, and user interaction. A quality model provides valuable insight into the behavior of complex, real-world systems. The model's purpose is to provide decision support analytics that help the user rationalize the decision-making process and justify choices. The model is not intended to replace the decision maker, but rather to offer a simplified representation of the system from a particular perspective. A key application of MBSE is prediction analytics. MBSE predictor applications provide predictions of known and forecasted data performance. Predictor designs range from component-centric models, X, to element-centric modules, x_i, or a combination.

The focus of the predictor methodology publication is centered on the cost model. Expanding this concept is the idea of system optimization, which requires a balance among cost, schedule, and scope within the system triad *Figure 5*.

Figure 5: System Triad (Ismail, 2014)

System Cost Prediction

Cost prediction is achieved through various estimation measures, including lifecycle cost estimates (LCCE), independent cost assessments (ICA), budget estimates, rough order of magnitude (ROM), estimate-at-completion (EAC), and independent government cost estimates (IGCE). Each methodology supports different phases of the system life cycle (SLC). While organizations define the terms differently, the concept of the SLC is consistent across most industries.

The DoD defines the SLC process as consisting of four phases: research and development, investment, operating and support, and disposal *Figure 6* (Office of the Secretary of Defense, 2014). Alternatively, the Office of Management and Budget (OMB) defines the program acquisition phases as: life cycle and concept analysis, technology definition, requirements planning (R&D), acquisition (Investment), operations and maintenance (Operations and Support), and disposal (Disposal) (OFFICE OF MANAGEMENT AND BUDGET, 2024). The sum of these four phases represents the system's total cost.

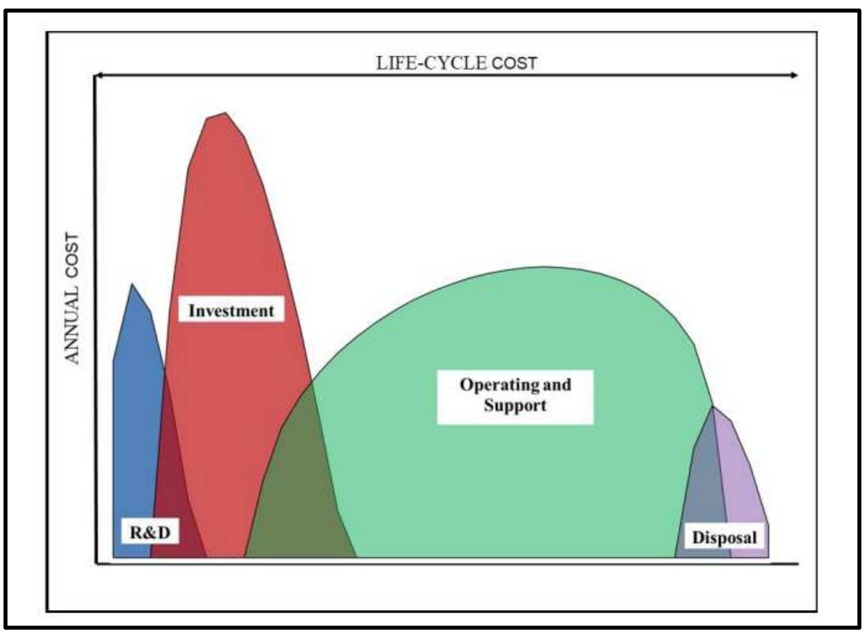


Figure 6: DoD Illustrated System Life Cycle Cost

(Office of the Secretary of Defense, 2014)

The primary function of predictive cost analytics is:

- Analysis of alternatives: an analytical study of the operational effectiveness, cost, and risks of potential options to a baseline for operational capability comparison.
- Cost Effectiveness Analysis: a systematic quantitative method for comparing costs of alternative methods of achieving a given objective.
- Benefit-cost analysis: a quantitative method of assessing the desirability of projects or policies in the long view of future effects and potential side effects.

Cost estimation is a critical element in any planning or acquisition process, helping decision-makers evaluate resource requirements at key milestones. It provides the basis for defending investment and drives affordability analyses. Credible results convey realistic perspectives of likely cost and schedule outcomes, which are used for resource planning, including facilities, equipment, labor, and other resources. It also provides an assessment of a program's feasibility, including how it should be implemented and the resources required to achieve objectives.

Acquisition Programs

The acquisition program focuses on the cost of developing and producing an end item. The objective of the acquisition process is to define a baseline program capability that meets the users' requirements, including affordability. During the acquisition process, decisions are made to determine how to consume finite resources most efficiently. Cost estimate validation requires testing to ensure that the estimates are reasonable and include all necessary components to execute the program successfully. The validation process relies on the assessment and application of historical data from similar programs.

In early program development, less is understood about the requirements, and the opportunity for change is greater. The GAO refers to this as the cone of uncertainty *Figure 7*. As more knowledge is gained and a program progresses, risk is reduced, and the potential for unexpected cost and schedule growth is reduced. Cost estimates become more certain as realized costs replace estimates. At its widest point, uncertainty about a cost estimate is highest. As the cone narrows, the uncertainty is reduced, and the risks are reduced (GAO-20-195G, 2020).

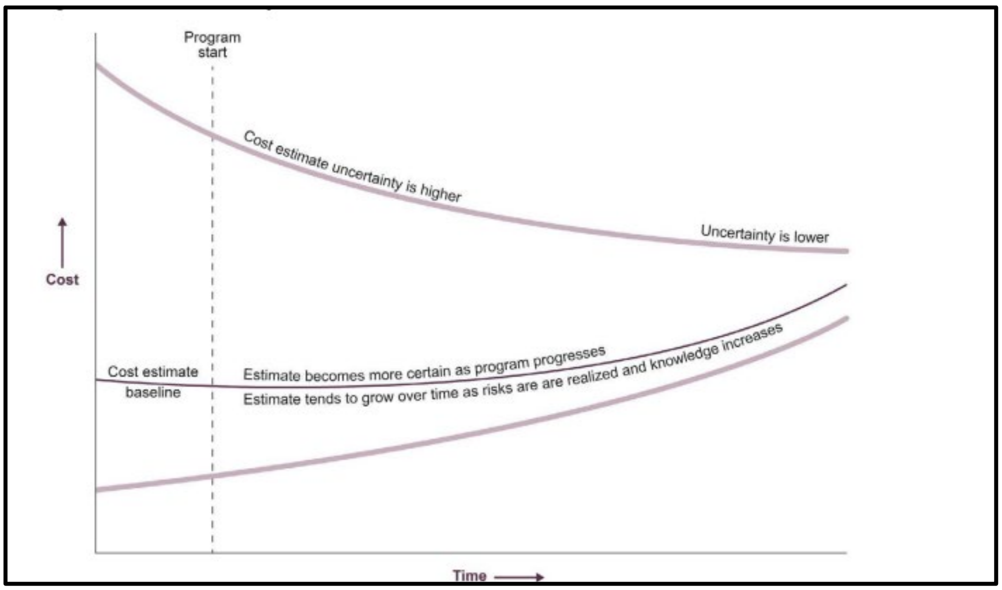


Figure 7: Cone of Uncertainty (GAO-20-195G, 2020)

Importance of Budgets

The program cost estimate creates a baseline budget spending plan that outlines the allocation and rate of program funding over time. Cost estimates are often used to demonstrate how budget cuts and spending delays hinder a program's progress or effectiveness. When contractors are overly aggressive or optimistic about program challenges, underestimation presents significant risk. Ultimately, either the contractor or the customer must bear the overrun costs.

Cost Affordability

Affordability is the extent to which an acquisition program's funding requirements fit within the agency's overall projected budget as part of enterprise planning. A program's affordability depends on the quality of its cost estimate. An affordability analysis demonstrates whether a program's acquisition strategy has an adequate budget. The DoD policy for affordability analysis is to address the nominal life cycle over 30 to 40 years (DOD INSTRUCTION 5000.02, 2020).

System Modeling – Current Practices and Understanding

The objective of cost estimation is to predict the final cost of a system. Substantial differences develop from the initial concept to the final implementation. Most conventional modeling tools evaluate system-level predictions, including spacecraft buses, payload instruments, systems engineering, and I&T costs, as well as other components of the system. Full system-level estimators are uncommon (Bitten, 2023).

The industry has developed a series of cost estimation models. Original models relied on product-oriented, weight-based cost estimations. They provided estimates structured around discrete hardware elements identified within a work breakdown structure (WBS) (Bui, 1996). These early models were deployed and utilized in the late 1980s and early 1990s:

- Unmanned Space Vehicle Cost Model (USCM6)
- Unmanned Spacecraft Cost Model, 5th Edition (USCM5)
- NASA Cost Model (NASCOM)

Additional tools supporting space-based mission cost estimation have been developed over the past decade *Table 2*.

Table 2: Industry Cost Models

Model		Source/Ownership	Level	Users	Notes
USCM	Unmanned Space Vehicle Cost Model	Space and Missile Systems Organization (SAMSO), Cost Analysis Division		Public	Released 1969; Updated 2015 (Foreman et al., 2016)
MOCET	Mission Operations Cost Estimation Tool	Aerospace Corporation and NASA Science Office for Mission Assessments		Industry via NASA ONCE Portal	(Foreman et al., 2016)
NASA CEH	NASA Cost Estimating Handbook	NASA	System	NASA, Industry	(NASA, 2008)
PCEC	Project Cost Estimating Capability	NASA Headquarters and Marshall Space Flight Center	Subsystem	Public	Released 2014; Updated 2016 (Foreman et al., 2016) replacement for NAFCOM
SSCM	Small Satellite Cost Model	Aerospace Corporation	Subsystem - post 1990 missions	Aerospace Corporation	Released 1995; Updated 2015 (Foreman et al., 2016)
AMCM	Advanced Mission Cost Model	Exploration Programs Office of Johnson Space Center			(Hayes, 2015)
TCASE	Technology Cost and Schedule Estimating	NASA Headquarters			(Hayes, 2015)
NAFCOM	NASA / Air Force Cost Model	NASA Cost Analysis Division and Marshall Space Flight Center/Air Force SAIC	Subsystem	NASA / Air Force / Industry	Released 1990; Retired 2012 (Kwan, 2005)
USMC 8	Unmanned Spacecraft Cost Model	Tecolote Research, Inc.	Subsystem		
PMCM	Planetary Mission Cost Model	JPL	Subsystem	JPL	(Kwan, 2005)
NICM	NASA Instrument Cost Model	NASA/JPL	Instrument	Air Force / NASA ONCE Portal	Released 2007; Updated 2014 (Foreman et al., 2016)
MICM	Multivariable Instrument Cost Model	GSFC	Instrument	NASA	(Bitten, 2023)
SOSCM	Space-Based Optical Sensor Cost Model	Aerospace Corporation	Instrument	Aerospace Corporation	(Bitten, 2023)
PRICE-H	Parametric Review of Information for	Price Systems, Inc.	Component	NASA / Air Force / Industry	(Kwan, 2005)

	Costing and Evaluation				
SEER	Systems Evaluation and Estimation of Resources	Golorath, Inc.	Component	NASA / Air Force / Industry	(Webb, 2023)
CASTS	Crew and Space Transportation Systems Cost Model	NASA MSFC ECO	Component	Unrestricted / Restricted versions	(Webb, 2023)

The GAO has identified four characteristics of a credible cost estimation:

- Comprehensive – clearly defines the program, including all elements of the schedule and technical baseline,
- Well-documented – can be easily repeated with traceability to the sources of basis,
- Accurate – adjusts for normalization. In terms of cost, this represents inflation. They are based on the realized performance of comparable programs.
- Credible – defines the limitations of the analysis, including uncertainties and biases in the data, as well as major assumptions. They also assess the sensitivity of the prediction to changes, including risk assessment.

The NASA CEH (NASA, 2008) outlines three cost estimating methodologies:

- Analogy
- Parametric
- Engineering build-up (summation or bottom-up)

Analogy (top-down estimating)

Using analogy, a new system draws its foundation from one or more previously implemented programs. When new technology and processes are integrated into the process, an underlying actual cost from a similar program can be identified and adjusted to account for the differences between the existing and proposed system, subsystem, or component. This methodology is based on the system engineer's (SE) ability and expertise in evaluating compatibility and assigning quantitative complexity measures.

Parametric (comparative system estimating)

The parametric method defines statistical relationships between system elements, encompassing cost, program, physical, and performance characteristics. The approach is considered top-down and traditionally based on measurable physical characteristics such as mass, power, or lines of code. The physical characteristics are referred to as cost drivers or system attributes. The methodology assumes that the same factors that affected cost in prior programs will continue to affect cost in comparable systems at an established prediction rate. The methodology is founded on historical data and the ability to identify relationships between system attributes.

Engineering Build-Up (summation or bottom-up estimating)

Engineering build-up is an industry standard. Sometimes referred to as a "grassroots" or "bottom-up" estimate, it involves establishing a WBS to make initial predictions and then aggregating detailed, lower-level estimates to obtain the top-level prediction. Accurately defining the system at a detailed level, consistent with a WBS, including all elements required for its realization, forms the foundation of this methodology.

The pros, cons, and areas of application for each estimation method are summarized *Table 3*.

Foreman (Foreman et al., 2016) evaluated cost modeling methodologies for distributed space systems. They concluded that current modeling technologies and practices:

- Often focus on a single system of a spacecraft mission.
- Require extensive design knowledge to produce high-fidelity estimates.
- Base estimates on assumptions that limit accuracy.

- Assign risk and complexity factors that are arbitrary and based on an individual's assessments.
- Tend to underestimate the cost growth during early formulation (Net et al., 2014)(Nag et al., 2014) (National Research Council, 2010)
- Require a deeper understanding of the nuances associated with the mission life cycle.
- Focus on goals that minimize cost, provide rapid delivery, and/or achieve maximum science data.
- Are sensitive to application. CERs developed from monolithic spacecraft are the foundation for cost-optimization advancements; however, recent trends are moving toward distributed spacecraft or constellations with no demonstrated cross-correlation.
- Are critical for mission proposal success.
- Allow management to assess feasibility, cost risk, and alternative designs while allowing stakeholders to understand mission requirements and project development better.

Effective and adaptive cost modeling is essential to successful mission design, implementation, and enabling mission success (Foreman et al., 2016).

Parametric Modeling

Bui conducted a comprehensive analysis of parametric cost modeling across 23 satellite programs. Data was extracted from the USCM7 model, functional costs provided by TRW and Rockwell International, and additional data supplied by NASA (Bui, 1996). The missions represented a pool of programs manufactured by 10 different prime space contractors: RCA, Fairchild, Ball, General Electric, TRW, Rockwell, Hughes, Ford Aerospace, Ball Aerospace, and Boeing.

Table 3: Estimation Methodology Comparison

Methodology	Pros	Cons	Applicability
Analogy	• Requires few data points • Based on actual data • Rapid response	• Subjective/biased • Accuracy depends on basis similarity • Difficult to evaluate alternatives • Cost driver insensitive	• When little foundational data is available • Rough-Order-of-Magnitude estimate • Verification of other estimates
Parametric	• Rapid response • Statistical basis • Objectivity • Cost driver identification • Incorporates as realized elements	• Lacks detail • Time-intensive Model investment • Black-box tendency requires model behavior understanding	• Early phase budgetary estimates • Baseline Estimate • Alternative and Trade study assessment • Design-to-Cost or other parameter
Engineering Build-up	• Sensitive to rates (labor, etc.) • Incorporates subcontract quotations • Industry accepted	• Requires design details • Time-intensive • Complex	• Production-based estimate • Negotiated procurement

The parametric model relies on system performance, physical characteristics, and weight attributes. They roll up subsystem estimates modeled from attributes to obtain mission estimates. Performance and programmatic cost drivers are determined independently for each of the business sectors represented in the data: DoD, NASA, and commercial satellites. For each industry, a unique set of CERs is defined by:

- Organization unit (engineering, manufacturing, etc.), and
- Recurring versus non-recurring elements.

The Bui model includes combinations of programmatic *Table 4*, performance-based *Table 5*, and weight-based *Table 6* cost drivers, placing less weight on single weight-based CERs.

Table 4: Parametric-Based Programmatic Cost Drivers

Prototype	Programs where prototypes were built in the development phase	⇧
Design Newness	Significant new design	⇧
Follow-on Acquisition	Same contractor	⇩
Missions	Communications	⇧
	Scientific – shorter mission	⇩
	Operational	⇧
	Experimental	⇩
User	NASA	⇩
	DoD	⇧
Launcher	Expendable	⇩
	Shuttle	⇧
Survivability	Antijamming	⇧
	Nuclear radiation hardening	⇧
Stabilization	3-axis stabilized	⇧
	1-axis stabilized	⇩
Operational Life	Longer operational life	⇧

Table 5: Parametric-Based Performance Cost Drivers

Power	Higher power satellites require higher Beginning-of-Life (BoL) power	⇧
	Longer-life satellites require higher End-of-Life (EoL) power	⇧
Solar Array Area	Higher power requires more capable solar arrays and costs more	⇧
Aluminum Content	Composite construction is lighter and more expensive than aluminum	⇩
Steay State Low Temperature	Operating at lower temperatures requires more capable thermal control	⇧
Quantity of Communication and Data-Handling Channels	More channels require increased capabilities and more communications equipment	⇧
Telemetry, Tracking, and Control	Higher power telemetry systems are more capable and cost more	⇧
	Higher precision and control systems are more capable and cost more	⇧

Parametric Modeling is defined for cost. Elements represent recurring or non-recurring cost elements (Work Breakdown Structure, or WBS, elements). X_i representing the cost driver and A and B_i parameters, where:

$$Cost = f\left(\frac{X_i}{A}, B_i\right), i = 1, \ldots, n \tag{1}$$

The CERs or X_i are defined using linear relationships in the log domain:

$$Cost_i = A\, X_i^B \tag{2}$$

Table 6: Weight-Based Programmatic Drivers

Total Dry Weight	⇧
Structure Weight	⇧
Thermal Control System Weight	⇧
Attitude Determination and Control Subsystem Weight	⇧
Sensor Component Weight	⇧
Reaction Control System Weight	⇧
Propulsion System Weight	⇧
TT&C System Weight	⇧
Communications Payload Weight	⇧

A is determined from ordinary least squares regression. Classical normal regression assumes normally distributed residuals in the log-log domain with an expected value and mode of zero. This regression is right-skewed, leading to values where the dependent variable equals the distribution's mode. They adjust the relevant CERs by using a half regression mean square error (σ^2) to the log-log intercept. CERs are determined independently for the recurring engineering (RE) and non-recurring engineering (NRE) based on subsystem design. The subsystem definition consists of seven elements, and the overall model is highly detailed and complex.

Existing parametric models for space mission cost estimation include a wide range of applications, systems, and objectives. Cost estimating methodologies evolve in response to technological trends. The current focus consists of the development of complexity-based cost estimating relationships

(CCERs). CCERs permit stakeholders to consider schedule and cost as part of the design process, rather than as a byproduct of designing a mission first and assessing the cost and schedule demands second (Foreman et al., 2016).

Performance-Based Cost Modeling (PBCM) quantifies the relationship between cost and measures of effectiveness (MoE) or performance (Shao et al., 2014). PBCM is designed to identify design drivers early and reduce mission costs from the outset. It is traditionally weight-based and designed to determine cost as a function of performance. Weight and mass are preferred as they are:

- easily quantifiable,
- relate directly to equipment lists and WBS elements,
- can be assigned ownership at the mission, system, subsystem, or component level, and
- historically correlate well with hardware cost (Shao et al., 2014).

A fundamental concern with current cost estimation and modeling practices is that they are not integrated with engineering models, physical constraints, or the propagation of uncertainty through systems of equations. Customer requirements, design, and cost are loosely coupled dimensions of a system design, referred to as a Frankenstein design, and attributed to the fragile design process. The coupling of uncertainty with cost-to-mass modeling is shown *Figure 8* (Edwards et al., 2022).

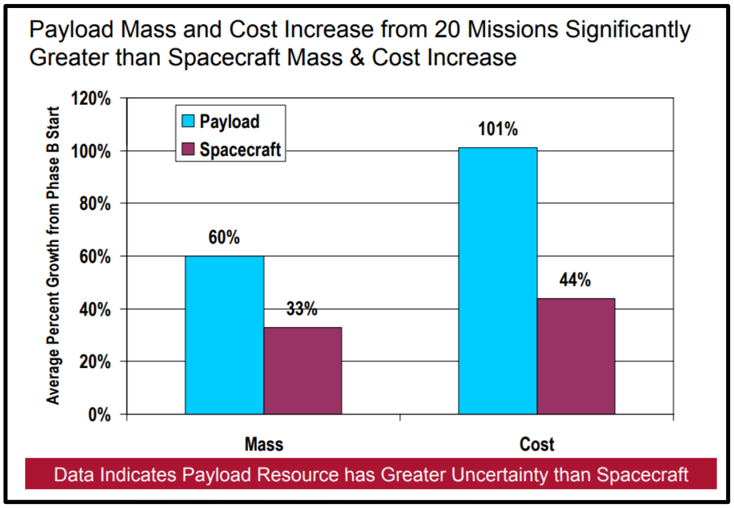


Figure 8: Aerospace Assessment of Cost/Mass Growth for 20 EO Missions

Spacecraft Cost Modeling

Most cost and schedule estimators are either mass (top-level) or WBS-centric (bottom-up summation) algorithms. Mass-based systems rely on launch mass, either wet (fueled) or dry (unfueled), as the basis for estimation. WBS-based systems are bottom-up summation tools that require a significant understanding of the mission, its segments, and associated architectures.

All space-based missions are complex systems of systems. However, they are constructed from a finite set of standardized building blocks, arranged in predictable construction patterns, undergo a variable yet constrained verification process, and operate within a single physical environment. Tools that provide ROM estimates rely on simple equations, such as:

$$Cost = 0.45 \cdot WeightPerUnit^{0.83} \text{ (Hayes, 2015)} \tag{3}$$

or

$$\begin{cases} C_{spacecraft} = 781 + 26.1\, {m_{dry}}^{1.261} \\ C_{payload} = 0.4\, C_{spacecraft} \end{cases} \tag{4}$$

From one mission to the next, satellite construction has similar design, definition, and test verification practices. ROM estimators tend to oversimplify and do not address the unique mission subtleties that exist between industries, lifecycle processes, or specific design requirements. As an example, (3) and (4) fail to incorporate launch or operational life considerations. Consequently, total cost estimators should not yield identical predictions for a six-month disposable mission as they do for 15-year deep space explorer missions. ROM estimators fail to differentiate these distinctly different applications when relying solely on a single variable, such as a mass.

Jet Propulsion Laboratory Cost Modeling

A significant stakeholder in NASA's interplanetary missions is the Jet Propulsion Laboratory (JPL). JPL maintains a formal cost estimation process, established in 2005 (Kwan, 2005). The process employs a standardized WBS, extensive review and evaluation by the JPL project team, NASA/industrial partner inputs, and independent external estimation for contributions outside of JPL. A mission estimate receives input from grassroots, program management, and independent cost estimates (ICE).

The grassroots estimates are submitted to a JPL program from internal and industry contributors, populating the WBS. Estimates include schedule and staffing requirements, along with a cost basis derived from metrics and prior program history. Independent of the primary cost estimate, external source contributors provide estimates to form each ICE. Cost contributions are reconciled to ensure concurrence, accuracy, and consistency of content. A final estimate is subject to multiple technical, management, and cost (TMC) reviews, each challenging the basis of estimation and the technical drivers that form the final mission budgets.

A JPL internal study of three interplanetary missions identified eight cost risk drivers and associated cost impact. Critical drivers are identified as mission complexity, technology development status, new software development, low scope margins, new architectures, participant experience, inadequate cost and schedule margins, and management inexperience. These risk drivers are attributed to cost overruns (Kwan, 2005).

Small Satellite Cost Model of the Aerospace Corporation

The Aerospace Corporation SSCM uses "in the absence of complete system design information, three approximate cost-estimating relationships (CERs)." The model is a hybrid that uses mass threshold values that partition the model's solution into separate regression components. Equations (5) and (6) demonstrate the approach and application of thresholds. The equations limiting spacecraft dry masses to less than 400 kg are from the SSCM. For SSCM modules, the system cost Y_i is proportional to the spacecraft's dry mass X_1 or dry propulsion system mass X_2. Equation (7) is from the USAF USCM. Inclusion of (7) provides for spacecraft dry mass X_1 in excess of 400 kg. The resulting hybrid model is:

$$Y_1 = X_1^{0.661} \; for \; X_1 \leq 400 \; kg \tag{5}$$

$$Y_2 = 1.0096^{X_2} \; for \; X_2 \leq 35 \; kg \tag{6}$$

$$Y_3 = \frac{43}{1000} X_1 \; for \; X_1 \geq 400 \; kg \tag{7}$$

This model is discontinuous at the points of intersection, requiring corrections to provide a continuous function. This model's utility is limited to ROM estimation and mission class comparisons (Crisp et al., 2019).

Modeling Methodologies

System Development

Systems are collections of interrelated elements. The elements consist of system components (objects) and abstract descriptors (attributes) that characterize the relationships between components. The fusion of objects and attributes enables the system to achieve greater unity than the individual parts. The system definitions are expressed at the top-level X or decomposed to include embedded subsystems x_i. Subsystems serve a unified purpose by contributing a specific function to the larger system.

Predictors

Predictor advancements have paralleled the achievements in computing and AI. Cost estimation methodologies include analogical, parametric, and bottom-up approaches, as well as combinations of these methods. As a system's development progresses, the process of estimation transitions from one method to another based on inherent strengths and limitations, improving estimation accuracy *Table 7*. Quality models, for estimation of other purposes, exhibit robustness and achieve a balance between simplicity and accuracy (Sayama, 2015). An overemphasis on performance accuracy tailoring often drives a model to become overly fit, often at the expense of generalization and obscuring simplicity.

Table 7: Estimation Methods

	Pros	Cons	Applicability	Accuracy (Bell & Hsu, 1995)
Analogy (Rule of Thumb)	• Requires limited data • Based on actual data • Rapid response	• Subjective bias • Similarity-based Accuracy • Difficult to verify • Cost driver insensitive	• Limited data available • Rough Order of Magnitude • Independent Assessment	50%
Parametric (MBSE)	• Rapid • Statistical basis • Objective • Cost drivers • Historical elements	• Lacks detail • Time-intensive Model investment • Black-box tendency requires understanding model behavior	• Early phase • Budgetary estimates • Alternative and Trade Assessment • Design-to-Cost or other parameter	30%
Engineering Build-Up (Summation)	• Rate Sensitive • Subcontract Inclusion • Industry accepted	• Requires design details • Labor-intensive • Complex	• Production-based • Negotiated procurement • WBS	15%

Accurately predicting elements of the program triangle for CSoS is a challenging task. Space missions are difficult to predict, given the limited data available and the unique operating environments in which they operate. The current practice for LCCEs (GAO-20-195G, 2020) rely upon four methods:

- Program office estimate (POE) — used to prepare customer objectives for translation into program and budget documentation, including resource requirement requests (cost, schedule, scope).
- Independent cost estimate — used primarily to validate POE. As an independent source, ICE provides the POE with unbiased support and identifies risks related to resource shortages or excessive requests.
- Budget Estimate — provides a short-term, limited life cycle that is translated to budget year dollars through the application of an inflation index and effort time phasing.

- Rough Order of Magnitude — based on a “conceptual estimate,” a ROM is developed when a quick estimate is required for needs analysis based on limited details or design. A ROM relies on historical information to support “what if” analysis, supporting limited or complete lifecycle analysis.

For mission optimization, concept definition requires the ability to predict the baseline's resource needs accurately and to compare them with those of alternative mission designs. These activities include:

- Analysis of Alternatives (AoA) —identifies requirements, needs, and operational shortfalls or excesses in capability.
- Cost-effectiveness analysis (CEA) — assesses different ways of reaching the same goals, comparing a measurement in terms of value.
- Cost-Benefit Analysis (CBA) — looks at long-term value and possible side effects of programs or policies to decide if they are valuable or desirable at the enterprise level.

Cost estimation (CE) is a critical activity within the acquisition process. CE enables decision-makers to assess time-phased resource needs at KDPs. The estimate substantiates budget requests and serves as the primary factor in affordability analyses. The CE’s purpose is to communicate the likely cost, schedule, and scope outcomes. Incorporating uncertainty enables the creation of a work plan defining development, production, maintenance, and disposal activities.

Predictor Performance

Beginning with the 1966 Surveyor-1 mission, NASA has launched sixty-six interplanetary missions to date. The introduction of computation in the mid-1980s significantly altered space mission design and the processes by which missions were realized. Advancements in AI and ML continue to evolve the space mission development process.

The development and performance demonstration of classified regression (CR) relies on a subset of NASA interplanetary missions documented in the NIM dataset. The NIM dataset contains space mission attributes for thirty-five interplanetary missions launched between 1989 and 2022. A summary of the realized NIM program cost and schedule duration performance is presented *Table 8*. The average overrun is reported as the ratio of reported values to the initial budget. One outlier, the 1989-launched Galileo mission, the first chronological mission of the NIM, straddled the advancements in local computing.

Table 8: NIM Historical Performance

	Cost	Schedule
Average Overrun (+ bias)	131.0 % Proposal	121.7% Proposal
Standard Deviation of Overrun	56.6%	39.6%
Dataset		
Abs Mean Error, $\|\mu\|$	34%	28%
St.dev Abs Mean Error, $\|\sigma\|$	55%	44%
Historical Performance, $\|\mu\|+\|\sigma\|$	89%	72%
Outlier Removed (Galileo mission)		
Abs Mean Error, $\|\mu\|$	25%	22%
St.dev Abs Mean Error, $\|\sigma\|$	30%	30%
Historical Performance, $\|\mu\|+\|\sigma\|$	55%	52%

NASA's standard for cost management is to estimate costs with a 70% confidence level. Other industries and the measure used for CR performance rely on a 1σ standard, which corresponds to an 84% confidence level for normally distributed data. Therefore, model and historical performance are defined at the prediction absolute mean error with an 84% confidence:

$$Performance = |\mu| + 1|\sigma| \quad (8)$$

Performance is a measure of accuracy, |μ| error, and uncertainty, k|σ| of the mean error. The factor k is a statistical multiplier based on a population or sample. The value of k is typically derived from the t-distribution, for an unknown population, or a normal z-distribution, for a known population, at a specified confidence level. Since the NIM dataset elements represent the total population of the computational era of NASA interplanetary missions, either statistical method is acceptable. Improved model performance is achieved when the covariance approaches unity, establishing a balance between accuracy (|μ|) and precision (|σ|).

The NIM dataset's realized performance measure is 55% growth in cost and 52% growth in schedule from proposal through realization, excluding the Galileo outlier. This results in $155 being realized for every $100 NIM budgeted, and 18 ¼ months being realized for every 12 months NIM scheduled. This level of performance is consistent with analogous estimation accuracy.

Cost Accountability

The 2008 NASA Authorization Act mandated "lifecycle cost and technical readiness" review for all NASA programs. The resulting Cost and Technical Evaluation (CATE) process provides an independent, standardized assessment tool for NASA programs. CATE predicts growth based on historical data and the design state, thereby assessing readiness and assigning technical risk. The process monetizes the risk to project growth and quantifies the threat to cost and schedule (National Academies of Sciences, 2015).

As CSoS missions increase in cost and complexity, the final price is not well understood until a minimum state of development maturity is reached, equivalent to Preliminary Design Review (PDR). A review of 20 space missions launched between 2008 and 2010 identified average mass growth of 41% and power growth of 37% from Phase B to launch. This growth is attributed to a

56% increase in cost and a 38% increase in schedule. The primary objective of CATE is to identify potential growth in advance using figure-of-merit indicators (National Academies of Sciences, 2015).

The CATE process decomposes the system into its major components and examines each one with respect to design heritage. The process draws an analogy to historical systems that have demonstrated cost, performance, and schedule. The technical evaluation consists of two components:

- proposed concept compared with legacy systems with known performance, and
- launch vehicle “mass to orbit” capability.

If the spacecraft's mass growth exceeds launch capability, the system requires either a reduction in mass or a more capable, and thus more costly, launch vehicle. Heritage status considers deviations from the state of the art, system complexity, and operational complexity. The key technical output of the process is a risk assessment. CATE utilizes multiple archive databases, including proprietary systems such as the Aerospace SSCM and industry standards such as NAFCOM, to minimize inherent industry- and source-bias. System-level estimates are generated by summing the costs of individual WBS elements, ensuring that the forecast reflects the system complexity. Complexity-based Risk Assessments (CoBRA) adjust the cost and schedule estimates. CATE quantifies the growth threat to the schedule and technology scope. Traditional S-curves, at the 70th percentile, are used to quantify cost growth probability (National Academies of Sciences, 2015).

Database of Systems

Data is the foundation of every estimate or system prediction algorithm for cost, schedule, technical design, or risk assessment. The quality of the data is reflected in the credibility and capability of the predictor. Estimators and validation tools rely on legacy mission data, applying various adjustment techniques that compensate for differences.

A challenge in data collection is obtaining or identifying the most accurate and appropriate data for reference, comparison, or modeling. CSoS are, by nature, complex, multifunctional, and distributed architectures with varying processes and methods for tracking and accounting for data. Enterprises that oversee or conduct assessment activities often view this data as proprietary, competitively sensitive, or even classified. CSoS details are rarely published to the public sector.

To evaluate the CSoS effectively, data is required that defines the system's technical scope, schedule, implementation, and cost attributes that describe how the CSoS was achieved. Data is obtained from a variety of sources, including historical data records, interviews, surveys, focus group publications, and market assessment studies.

Understanding the system's objective is critical to the system estimation process. This considers the program acquisition plan, the contracting process, marketplace conditions, and the contractor's past performance. Statistical analysis of mission-critical data identifies attributes that drive the system definition and resource requirements. The three main classes of data are: cost, schedule, and technical scope.

Databases of Space Missions

Few publications summarize space missions. Bitten et al (Bitten, 2023) presented to NASA a summary of industry cost databases *Table 9*. Other sources include the series of semi-annual publications, Jane's Space Systems and Industry (Bond, 2011). A comprehensive listing of space mission characteristics and attributes is not published.

Table 9: Industry Cost Databases

Model		Source/Ownership	Missions	Instruments	Notes
CADRe	Cost Analysis Data Requirement	NASA HQ	~100	From ~100 missions	NASA Cost Community (Bitten, 2023)
NAFCOM	NASA/Air Force Cost Model	SAIC	>100	>350	NASA/Air Force Community (Bitten, 2023)
NICM	NASA Instrument Cost Model	JPL	~	160	NASA Cost Community (Bitten, 2023)
SSDB	Small Satellite Database	Aerospace	~140	~	Aerospace Only
ASID	Aerospace Space-based Instrument Database	Aerospace	~	~600	Aerospace Only
PEDB	Planetary Exploration Budget Dataset	The Planetary Society	~50 (missions or programs)	~	Cost

Planetary Exploration Budget Database

The Planetary Society (TPS) is a nonprofit organization that brings together scientists and volunteer members to advocate for future space exploration, educate the public about space exploration, and invest in innovative technology. The Planetary Society publishes and maintains the Planetary Exploration Budget Database (PEBD). The PEBD provides a summary of the spending history by year for every NASA planetary science mission and related activity. The data includes costs compiled from the annual NASA budgets submitted to Congress from 1959 through 1997 and from 2002 to the present. From 1998 through 2001, NASA neither reported nor requested program-level funding.

NASA Interplanetary Mission Database

The NIM database includes published mission design data for 33 NASA missions and 35 spacecraft (Dockstader, 2025). Data sources include published NASA Budget Requests from 1984 through 2024, the NSSDCA database, and other references. NASA Budget Requests provide the input for proposed mission costs and schedules. To be included in the dataset, the mission had to have achieved Phase E operational status before 2023. Risk attributes include mission success or failure, a comparison of budget versus realized total mission cost, design schedule duration from funding to launch, and operational life.

The NIM dataset includes:

- Mission Types: Orbiter, Fly By, Lander, Sample Return, Defense Test, Impact, Test Flight
- Primary Mission Destination: Lunar (moon), Venus, Mars, Jupiter, Saturn, Asteroid, Comet, Solar Wind, Pluto

Cost attributes for the years 1959–1997 and 2002–present (as of 2024) are obtained from annual NASA budget requests submitted to Congress. These requests outline expenditure obligations for the two years preceding the fiscal year requested and project spending over four years into the future. TPS provides download access to annual NASA budget requests. This excludes a six-year period during which budget requests lacked program-level, detailed data. The cost was reconstructed from prior NASA planning and subsequent performance reporting budgets during this period. Costs that could not be reconstructed were obtained from alternative NASA documentation sources, including press kits, program summaries, NASA and JPL document archives, historical records, and presentations.

Schedule attributes collected included proposed or funding plans, realized KDP activities, launch records, public information releases, and news articles regarding operational milestones and events. Sufficient data was obtained to obtain attributes for all programs based on recorded dates:

- Development duration is the time difference between the launch date (launch manifests and news releases) and the contract initiation/award date (NASA budgets and KDP documents).
- Launch duration is the launch date.

- Cruise duration is the difference between the operational planning or news release date and the launch date.
- Primary operation duration is the time between mission event dates (failure, planned mission operations, notification/approval of extended operations, or end of life) and an on-station or operational status date.
- Extended operation duration is the time difference between 3/1/2024 (current date at the time of recording) or End of Mission (press releases) and the notification/approval of the extended operations date (NASA budgets, press releases).

The scope technical design attributes include mass, power, payload, and mission design details. The data is obtained from a variety of technical sources, including NASA and JPL archives, press releases, NASA open-source databases and records, Jane's biannual publications, and alternative sources. When data sources conflicted, sources were prioritized based on proximity, and relevant data was selected. Priority was defined, from most reliable to least reliable, as NASA, NASA affiliates, NASA contractors, alternative technical sources, and news sources.

The program attributes include non-numerical mission information, including mission purpose, mission destination, program source, and prime contractor. Additional attributes include program designation, COSPAR number, mission success and status, failure information, and launch vehicle.

Data Pruning

Significant consideration was given to the missions that compose the NIM. NASA experienced a void in the development of interplanetary missions from the 1970s through the late 1980s. This lack of mission was attributed to NASA funding constraints and the logistics of the International Space Station (ISS) and Space Shuttle programs. The dataset includes missions that were launched after 1989. The mid-1980s represented an engineering revolution. The introduction of desktop computers and associated computer-aided engineering resources, processes, and practices significantly altered the CSoS engineering process. Process improvements were implemented throughout the system development process, from the mission level to component design, manufacture, and testing. The transition shifted from design-by-test and human-based engineering to design-by-analysis and simulation. Computer-based applications significantly accelerated and improved the efficiency of traditional practices. This facilitated alternative assessment, deeper understanding, and greater flexibility in how systems are implemented, monitored, and achieved. The resources supported computer-aided design (CAD), computer-aided manufacturing (CAM), computer-aided analysis (CAA), and the ability to perform complex analytical tasks repeatedly at the convenience of a workstation.

Data Normalization

The NIM dataset consists of 35 interplanetary missions launched between 1989 and 2024. The primary data consist of cost, schedule, scope, and program attributes. While measured units of pounds, watts, and years are standardized, the cost and time attributes require analysis.

NASA standard practice is to inflate year-of-occurrence dollars to a standard reference through the application of the NASA New Start Inflation Index (NSII). NASA and GAO universally accept this practice as a credible cost data normalizer.

A series of cost inflation indices is compared to the NSII for forward and reverse pricing consideration *Figure 9* and *Figure 10*. *Figure 9* illustrates the index performance, considering forward pricing from a 1961 baseline, for each model. Costs were modeled to 2022 equivalent dollars, then further projected to 2023 and 2024, applying a 1.0 index for each subsequent year.

Figure 10 applies a reverse-pricing model based on a $100 value in 2022 and projects back to 1961 to obtain “equivalent year dollars” for each index.

The indices yield distinct curves for comparison among inflation models. T-tests evaluate the correlation between pairs of indices. Alternative inflation models demonstrated statistical significance compared to the NASA NSII index. Comparison of the alternative indices did not yield similar significance levels, failing to reject the null hypothesis. The NSII model correlates well with labor rate standards reported by the U.S. Department of Labor (DoL). However, this data is limited and was obtained from the Department of Labor reports for 1961, 1965, 1990, and 2022. Despite a small sample size ($N = 4$), a relationship is observed between technology labor rates and the NSII.

The alternatives to the NSII include:

- CPI – Consumer Product Index, historical basis of “standard product” costs from 1913 through 2023 (US Bureau of Labor Statistics, 2023)
- PCEPI – Personal Consumption Expenditures Price Index (FRED Economic Data, 2023a)
- CPI-AUC – CPI with adjustment for all urban consumers, less food and energy expenditures (FRED Economic Data, 2023b)
- S&P 500 – Standard and Poor 500 NYSE stocks (not included but considered)
- S&P 500–AS – S&P 500 NYSE stocks adjusted for inflation (Market, 2023)

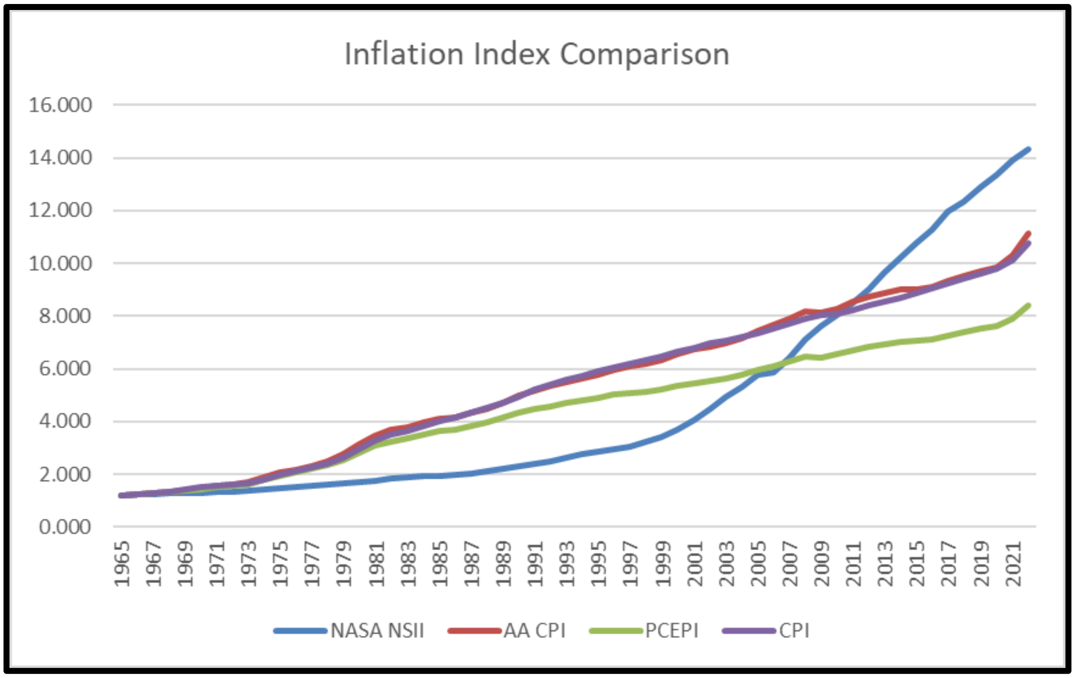


Figure 9: NASA NSII Comparison to other Inflation Indexes (forward priced)

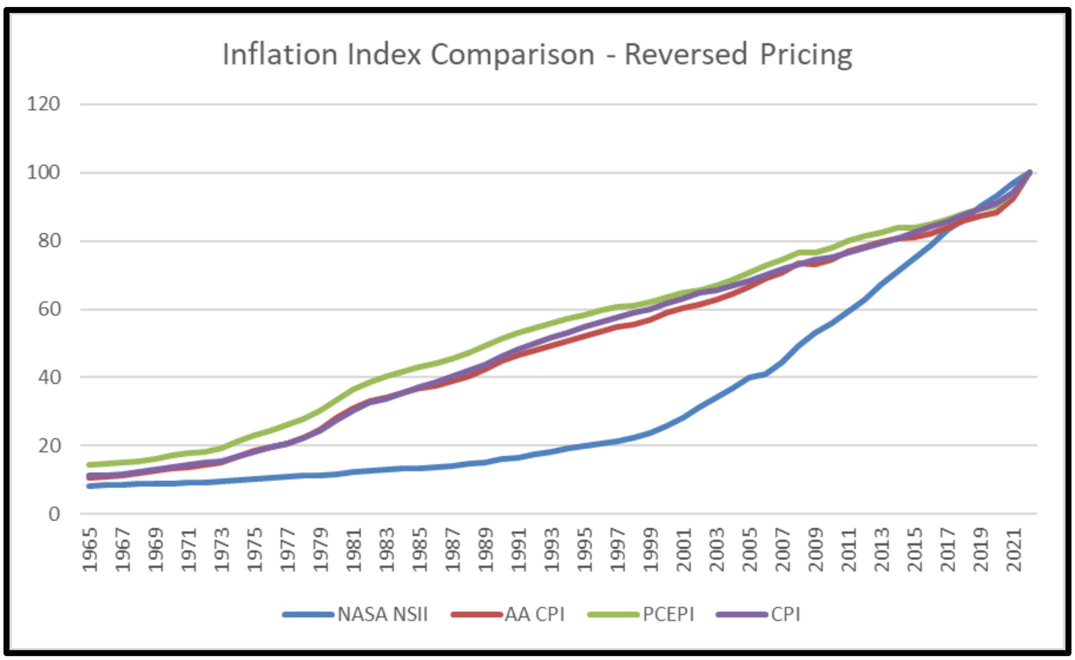


Figure 10: NASA NSII Comparison to other Inflation Indexes (reverse priced)

For the development of CR, three cost references are considered:

- The then-year dollars (Then-$) as reported in the NASA budget requests documented in the NIM
- NSII inflated (NSII-$) to fiscal year 2024 values (Appendix A)
- PCEPI inflated (PCEPI-$) to fiscal year 2024 values (Appendix A)

The funding profiles for each NIM mission are obtained from the NIM dataset. Values are inflated for both the NSII and PCEPI indices, producing cost attributes in NSII-$ and PCEPI-$ to 2024 equivalent dollars to provide normalized cost data.

Artificial Intelligence and Machine Learning

ML is the fusion of AI, computer science, and mathematics that focuses on leveraging data and algorithms based on human learning to improve throughput and accuracy. The practice utilizes algorithms to parse data, learn, and make predictions about other entities based on the input data. When applied to data science, ML leverages statistical and optimization methods, enabling dataset analysis and pattern identification. ML is often used to implement data mining algorithms that are specifically trained to recognize historical trends. This ability to link existing data to prediction models directly drives decision-making across both applications and businesses.

The origins of machine learning date back to 1950 and the "Turing Test." This test was designed to determine whether a computer demonstrates real intelligence by deceiving a human into believing it is also human. In 1967, the "nearest neighbor" (NN) algorithm was developed, enabling computers to utilize basic pattern recognition and optimize routes. The NN-algorithm allowed a salesman, starting at a random city, to determine the most efficient path to visit all cities during a tour. During the 1990s, developments in machine learning shifted from knowledge-driven approaches with defined variables and discrete solutions to optimization, to data-driven methods that process large quantities of data to draw conclusions and "learn" from the results.

Types of Machine Learning

The absence or presence of human intervention, interactions with raw data, and processing decisions classify the different types of ML. There are five standard classes of machine learning:

- Supervised learning — the algorithm is trained from defined data that clearly identifies the resultant. These algorithms are evaluated based on the accuracy of the training data. Supervised learning is primarily applied to classification and regression problems. Classification problems require the algorithm to identify discrete data and predict a class for data elements. The algorithm is evaluated based on its ability to accurately classify the training data. Regression problems involve evaluating continuous data and determining the correlation between input and output. Supervised learning is suited to data problems where the data is grounded in truth.
- Unsupervised learning — the algorithm self-trains on data without a specific definition through deep learning. In unsupervised learning, a network attempts to identify structure within data through feature extraction and structure analysis. Learning can be achieved through various methods, including clustering, anomaly detection, association, and autoencoders. Because unsupervised learning is not based on a "ground truth," it is difficult to define the accuracy of an algorithm. While unsupervised learning is being tuned or trained, it can produce wrong answers. Through repetition, training, and vast quantities of data, the algorithm is tuned to yield highly accurate answers.

- Semi-supervised learning — the training data set contains both classified and unclassified data. Semi-supervised learning is a compromise that benefits from the efficiency of labeling a small segment of the data.
- Reinforcement learning — a reward-based learning common in gaming. Reinforced learning is an iterative process with integral feedback that attempts to maximize the summed feedback rather than relying on individual attempts. This is common in robotic activities and strategy, such as chess.
- Deep learning — learns automatically without human intervention, including rules and knowledge, simulating human brain function and processing. Deep learning requires significantly more data than alternative learning methods. Deep learning models are a nascent subset of machine learning paradigms. They exploit interconnected intermediate layers that enable rapid, efficient learning of complex prediction models.

Supervised Learning

A supervised machine learning algorithm is constructed of three core components (Salian, 2025), *Figure 11*:

- A decision process that takes input data, implements a series of calculations, and guesses the pattern of interest
- An error function that evaluates a measure of the accuracy of the guess
- An optimization process that evaluates errors and updates the decision process to improve iterative accuracy

As the machine learning algorithm updates, the accuracy improves through optimization.

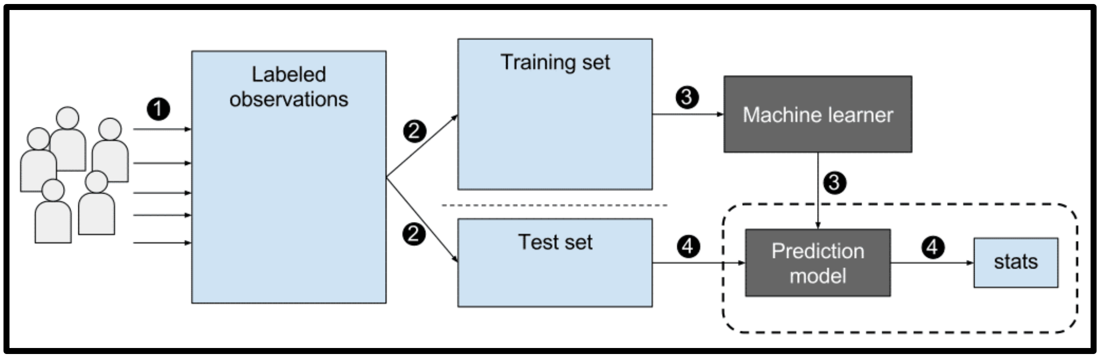


Figure 11: Supervised Learning Process (Salian, 2025)

Machine Learning Algorithms

ML's primary goal is to utilize learning algorithms to analyze and classify data. Machine learning leverages the computational efficiency of computers to evaluate large quantities of data with or without human intervention. There are several standard machine learning algorithms:

- Linear regression analyzes the relationship between independent input variables (features) and at least one target variable (class). This algorithm is practical for continuous outcomes with constraints. Linear relationships tend to occur when the observed data follow a predictable pattern, such as a straight line. This is the most common form of machine learning algorithms.
- Neural networks are AI algorithms that emulate the way the human brain processes and intelligently classifies data. Data is fed through a layered nodal network that processes and assigns weights before passing it to the next nodal layer. There are limitations to neural networks. Artificial neural networks exhibit discrete layering, connections, and data propagation flow, unlike the human brain, where any neuron can connect to any other neuron within a certain physical proximity. The neural network is engineered.

- Decision trees are node-based structures that are used to test data against some conditional input data. To filter the correct output, the input data is compared against the leaf nodes of the tree. Categorization is typically based on a categorization schema under supervised learning.
- Random forest models classify data using a variety of decision tree models simultaneously. The user-specified decision trees are collectively referred to as an ensemble. Each decision tree results in a prediction, and the random forest machine makes a final decision by combining the predictions of each decision tree in the ensemble.

Regression Analysis

Regression is the process of determining trends in data, represented by a line or curve. Linear regression generates a line of best fit. We choose the best fit by summing the squares of the vertical distances between the data points and the proposed regression line. The solution aims to determine the minimum value or optimal fit for the data. Other forms of regression include polynomial and exponential functions *Table 10*.

Linear Regression

Linear regression is a popular form of predictive analysis. It establishes the relationship between a set of independent predictor variables and a dependent outcome variable. The simplest form of regression is a relationship between a single dependent and a single independent variable, where:

$$y = c + b \cdot x \tag{9}$$

Table 10: Regression Analysis Types

Form of regression	Dependent	Independent	Assumptions
Linear	1 dependent (interval or ratio)	1 independent (interval, ratio, dichotomous)	Linear relationship Multivariate normality Little multicollinearity No autocorrelation Homoscedasticity 20 cases per independent variable
Multi-linear	1 dependent (interval or ratio)	2+ independent (interval, ratio, dichotomous)	
Logistic	1 dependent (dichotomous)	2+ independent (interval, ratio, dichotomous)	When the dependent variable is binary
Ordinal	1 dependent (ordinal)	1+ independent (nominal, dichotomous)	The dependent variable is the order response category
Multinomial	1 dependent (nominal)	1+ independent (interval, ratio, dichotomous)	
Discriminant	nominal	1+ independent (interval, ratio)	Predicts group members by fitting a linear regression line through a scatter plot. If more than two independent variables, it fits a plane through the scatter plot, separating observations into one of two groups – predicting group membership.

In this relationship, y represents the dependent variable value at x, c is a constant that defines the y-axis intercept, b is the regression coefficient (the slope of the regression curve), and x represents the independent variable value. Regression is used to:

- quantify the strength of a predictor (independent),
- forecast an effect (trend and sensitivity), and
- trend forecasting (prediction).

The equation for the regression line is found using the least squares method:

$$m = \frac{(n\sum xy - \sum x \sum y)}{(n\sum x^2 - (\sum x)^2)} \tag{10}$$

and

$$b = \frac{(\sum y - m \sum x)}{n} \quad (11)$$

Collinearity is a concern in regression analysis when selecting variables. When independent predictor variables in the same regression model are correlated, they cannot independently predict the dependent variable and are not statistically significant. The selection of the proper regression solution is based on model fitting; however, overfitting data is a considerable concern. Adding independent variables to a linear regression model will improve the variance of the model (expressed as R^2) but reduce the model's generalizability.

Linear regression identifies a single peer-to-peer relationship between the independent variable and the dependent variable *Figure 12*. Linear regression does not require a normality assumption for the inputs; the estimator is generated via linear least squares. A byproduct of ordinary linear regression (OLR) is the presence of homoscedasticity in the residuals or errors. The response variable is of the form:

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i \quad (12)$$

Where $\varepsilon_i$is the residual and is approximated as N(0,σ) for some value of σ and i is the sample size of the dataset representing pairs of data (x_i, y_i). x_i is the predictor or independent variable. The predictor variable is used to calculate the response variable y_i (outcome). For a normal distribution, distributions exhibit the same variance that is centered about the regression line (Wicklin, 2015). Independence is strictly defined in probability. x_i and y_i are not strictly defined as independent and dependent, but instead are co-dependent.

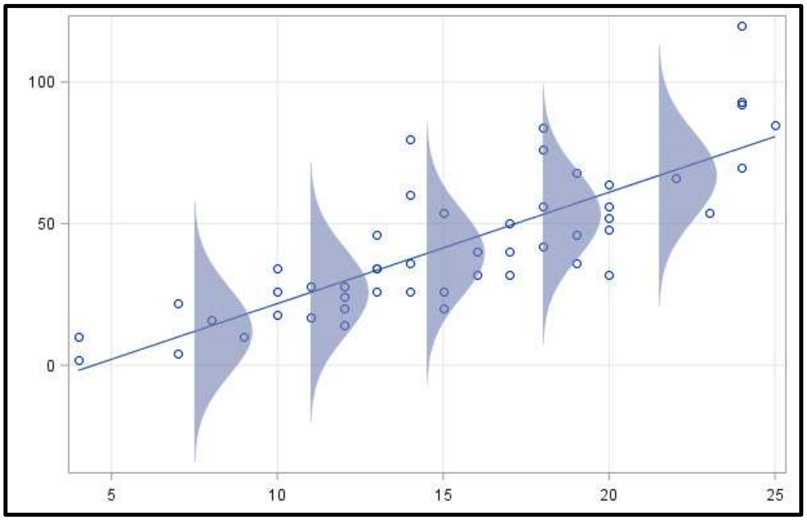


Figure 12: Linear Regression for a Normal Distribution (Wicklin, 2015)

Linear regression for a normal distribution develops relationships between the predictor variables (attributes) and the response variable (resource needs) using classification features. This relationship can be defined as:

- Response = Prediction + Residual (error)
- Response = Modeled + Unexplained
- Response = Deterministic + Random
- Response = Explained + Unexplained

Regression solutions that display responses "far from the regression line" represent examples of underfitting. Higher-order regression, which relies on multiple terms or variables and is characterized by "wiggly" functions, is representative of overfitting. The data is not representative of a function relation, but instead multiple values of y_i for a single value in x, explained by our normal distribution of data in y. Regression assumes that the data can be represented as a function or a step-wise function at x_i where the residual is normally distributed about each x_i and for any value of x, y_i has the same variance (σ^2) *Figure 13* (Dalpiaz, 2016).

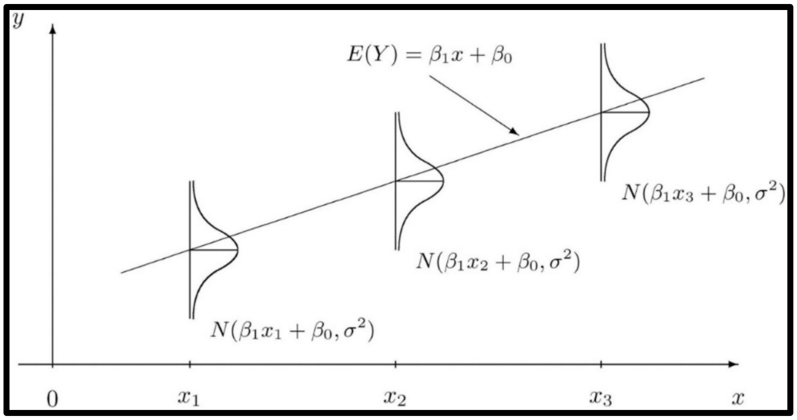


Figure 13: Simple Linear Regression Model (Dalpiaz, 2016)

The assumptions of simple linear regression for a normal distribution are:

- Simple: using a single predictor variable, opposed to multiple predictor variables of higher order or multivariable
- Linear: the relationship between the predictor variable and response variable is linear in form: $y_i = f(x) = \beta_0 + \beta_1 x$
- Independent: the error residual ε_i are independent of the independent variable x_i
- Normal: The error residual ε_i are normally distributed, and the error about the predictor follows a normal distribution
- Equal Variance: at each value of x the variance of y is the same, σ^2 is consistent throughout the predictor

The process of regression involves fitting constants $\beta_0 \; and \; \beta_1$ such that they minimize $\sum \sigma_i^2$ through an established method, commonly the least-squares approach. A standard measure of "fitness" is the coefficient of determination R^2. The R^2 defines the portion of the observed variation in y explained by the regression. The value indicates the percentage of observed variability explained by the linear relationship and how "tightly" the data fit a line. The coefficient of determination and Pearson’s coefficient are standard measures of “fitness” for regression models.

Classifiers

Classification is a machine learning technique in which an algorithm identifies patterns and relationships between input features and output labels. The classifier systematically assigns elements, whether abstract or concrete, to a set of classes based on the model's training. Classifiers are typically trained using supervised learning systems, where the output label is known during training. Once trained, the model can forecast the class assignment of new data. There are various types of classifiers in ML, broadly categorized as binary, multiclass, and multilabel classifiers.

Support Vector Machine

The support vector machine (SVM) was developed in the 1990s (Vapnik, 1996) for application as a classifier. The SVM defines an optimal hyperplane that separates data into different classes. The points closest to the hyperplane are known as support vectors, which determine the decision boundaries. The support vectors define the margin around the decision boundary *Figure 14*. The SVM operates on linearly separable data. When not, kernel functions are used to transform the data to higher dimensionality, enabling separation.

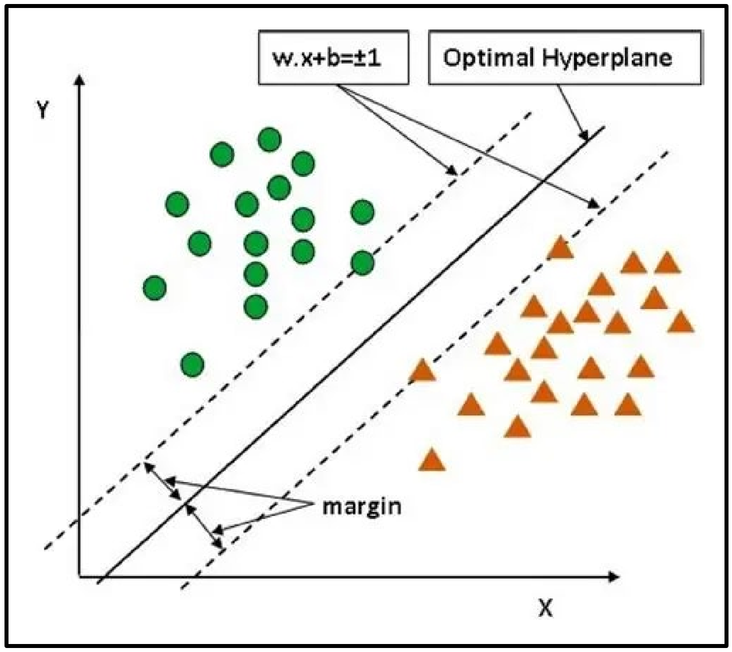


Figure 14: Support Vector Machine (Pathak et al., 2022)

Exhaustive Classifier

The Exhaustive Classifier (EC) is a relatively simple numerical classification algorithm that incorporates human knowledge and learning from user-defined data. The user-defined options include feature (F) selection and feature combination (FC). The EC uses F/FC vectors and mean values to define critical-value thresholds that predict class assignments. The threshold between classes is identified as the critical point (CPt). The CPts are defined for each F/FC used by the classifier. In *Figure 15* a two-class classifier (N = 2) is shown, where both classes are represented as normal distributions.

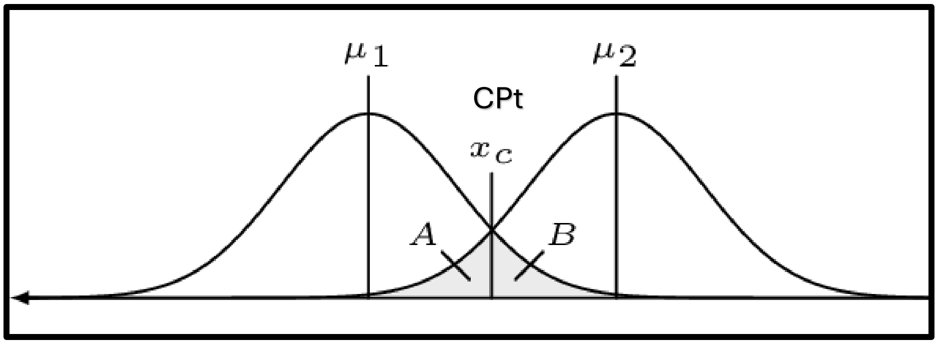


Figure 15: Probability Distribution for an N=2 Exhaustive Classifier *(Pirani & Cale, 2022)*

The CPt is the vertical line segment between class 1 (μ_1) and class 2 (μ_2), or the mean value of each class. Except for perfect classification, as shown with the SVM *Figure 14*, overlap is represented by the shaded regions A and B. The value for CPt is located an equal number of standard deviations, defined as $\mathrm{n}\sigma_{CPt}$, from μ_1 and μ_2. The F/FC populations do not need to have equal variance and σ_1 and σ_2 are the standard deviations of the F/FC data for Class 1 and Class 2, respectively. The equal number of standard deviations is computed as:

$$\mathrm{n}\sigma_{CPt} = \frac{\mu_2 - \mu_1}{\sigma_1 + \sigma_2}\ , where\ \mu_2 > \mu_1 \tag{13}$$

The CPt can be calculated as:

$$CP_t = \mu_1 + \mathrm{n}\sigma_{CPt} \cdot \sigma_1 = \mu_2 + \mathrm{n}\sigma_{CPt} \cdot \sigma_2 \tag{14}$$

By defining CP_t, the class prediction is simplified. If the sample value is less than CP_t, it belongs to cluster 1 and is identified as class 1. If the sample value is greater than CP_t, it belongs to cluster 2 and is identified as class 2.

Feature Identification

A desirable attribute of the EC is the ability to select, evaluate, and define relevant F/FC. Commonly, Fs represent the attributes of the system, and FCs represent combinations of the attributes. The F/FC data, both linear and non-linear, are selected for evaluation using the EC and assessed for their relevance and accuracy in predicting classification. Feature combinations are product or ratio combinations of raw features:

- Feature: A
- FC linear product: $A \cdot B$
- FC linear ratio: $A/_B$, $B \neq 0$
- FC Non-linear product: $A^2 \cdot B$

Dimensionality

Allowing N_c to represent the number of data classes within our classifier and N_f to represent the number of relevant F/FC, the mean and standard deviation for each feature are determined in an $N_c \; x \; N_f$ matrix of μ and σ . Each F/FC is evaluated with respect to pairwise nσ_{CPt} values. To define critical F/FC for classification, the Feature $p - value$ is summed. F/FC demonstrates statistical significance, $p < 0.05$, are selected for further evaluation, F/FC where $p > 0.05$ are discarded. The ranking of F/FC is achieved by ordering the most significant, lowest $p - value$, to least significant, highest $p - value$, of the statistically significant. Based on the ranking, the N most relevant F/FCs are evaluated. Higher-performing F/FC are selected as classifier(s) and used for modeling (Pirani & Cale, 2022).

T-tests (14) evaluate each F/FC based on the deviation of the mean from the critical point(s). The t-test determines the numerical relevance or priority for each F/FC. P-values from the t-tests identify the F/FC that are most useful for classification.

$$t = \frac{\mu_1 - \mu_2}{\sqrt{\frac{(n_1-1)\sigma_1^2 + (n_2-1)\sigma_2^2}{n_1+n_2-2}}\sqrt{\frac{1}{n_1}+\frac{1}{n_2}}} \tag{15}$$

Weighting

When using multiple F/FC for classification, the relevant Features are weighted, assigning higher weights to those with greater relevance, which, in turn, has a greater influence on the classifier's prediction. Weighting is assigned using a normalized probability vector. Weighted solutions are summed (combined) to either define a solution or evaluate whether the solution is correct or incorrect.

F/FC weighting is based on a simple binary boundary for each of the N relevant F/FC. Weighting can be defined as a function of accuracy or error.

For accuracy:

$$Weighting_{accuracy(i)} = \frac{Accuracy(i)}{\sum_k Accuracy(k)} \tag{16}$$

For error:

$$Weighting_{accuracy(i)} = \frac{1/_{error(i)}}{\sum_k 1/_{error(k)}} \tag{17}$$

Overfitting

Small sample datasets provide a challenge for classification algorithms and a desire to overfit (Chollet, 2017). Data augmentation is the preferred method to mitigate this concern.

Class Assignment

Classification or the class assignment begins with the z-score (distance) calculation. The z-score function is:

$$z(y: \mu, \sigma, n) = \frac{|y-\mu|}{\sigma/n} \tag{18}$$

Where y is the F/FC value input, and the parameters $\mu, \sigma, and\ n$ are the mean, standard deviation, and n-number of samples used to calculate the mean and standard deviation.

Using the z-score, a conditioning function is defined as:

$$f(z) = \frac{1}{z} \tag{19}$$

or

$$f(z) = \frac{1}{z^2} \tag{20}$$

Both (19) and (20) define increasing function values (greater weighting) for decreasing z-score. After normalizing the values of F/FC, weights determined from the training data set are applied to F/FC. The classifier sums these values across each F/FCs, assigning the sample to the class of the highest value. The values of the $N - F/FC$ classifications are then summed or tabulated with the classifier assigning the sample a final class.

Small Sample Classification

Machine learning struggles to classify small datasets due to high dimensionality. Kim and Cho (Kim & Cho, 2015) implemented a meta-level classifier construct for gene expression. Their investigation evaluated GA-based ensemble optimization using different combinatorial methods and various evolutionary search algorithms. They utilized the outputs from MxN classifiers as input for meta-classification using a simple clustering algorithm. The meta-classifier determined the sample's final class by combining predictions from multiple base classifiers. Classifiers were selected from a pool of models based on their training accuracy. This approach demonstrated limited success due to overfitting and unstable performance across multiple datasets. They concluded that the MxN space was too large and reverted to a simple classifier to combine the results from the base classifiers. The meta-learner constructed a set of several classification algorithms and different feature subset selection algorithms. Their meta-classifier:

- Evoked a two-step process: training and meta-classifier learning.
- Relied on N feature selection ranking all genes based on different criteria. Top-ranked genes were used to generate training datasets for each selection method.
- Used M learning methods were repeated for N different training sets, creating MxN base classifiers learned.
- Formed a meta-classifier from the classifiers. Meta-classification included ensemble searching and learning a meta-level classifier. Used optimization to find the best subset of the base classifiers for an ensemble.
- Used all base classifiers' decisions as inputs and produced an outcome. For their cancer problem, only two classes were available. Therefore, they could cluster the training samples into two groups based on the predictions from multiple classifiers. After clustering, each cluster is mapped into one of the two classes. To classify a new sample, base classifiers were used to generate vector predictions. The sample was assigned to a cluster, which classified the sample as one of the two classes.

Fitness Functions

Kim and Cho (Kim & Cho, 2015) considered the importance of measuring the value of ensembles in evolutionary algorithms. They defined performance metrics as:

- Accuracy — by using the accuracy of the ensemble on the training data, the number of correctly classified samples is divided by the total number of training samples.
- Confidence — the sum of the ensembles' confidences for the class label of the training samples.

- Accuracy x Confidence — a combination measure.
- Minimum description length (MDL) — a parameter that provides preference to higher accuracy and the least number of members.

Chapter 3: Method of Classified Regression

Classified Regression Framework

Classified Regression (CR) is an ensemble of ML methods based on support vector machine (SVM) margin boundaries for unsupervised classification, class-wise regression for the predictor, and an exhaustive classifier for supervised forecasting. Model construction accommodates a $(predictor \rightarrow response)$ variable relationship for attribute pairs. Models serve as a predictor (Model #1) or are combined to form a summation triad component-level predictor (Model #2).

Data

Planetary Exploration Budget Dataset

The initial CR development investigated system cost modeling. Cost data was derived from the PEDB (Callahan, 2025). This public dataset compiles the annual budget requests and spending histories for NASA planetary science missions from the late 1950s to the present as of 2023. Cost entries are reported as annual, reconciled expenditures by cost element in the year of incurrence dollars (Then-$). The cost elements are consistent across missions but reflect changes in nomenclature and procedures within NASA over the reporting history. The PEBD provided the primary data for evaluating a system cost model, segmented into components for development, launch, and operations. The sum of the cost components provides the total mission cost. The operational costs did not differentiate between primary and extended mission operations.

The PEBD dataset is determined to be deficient in several critical areas:

- Expendable launch vehicle costs are included in the mission totals as discrete cost entries, included within the implementation costs, or extracted from referenced sources. When launch costs are extracted, the total launch cost is applied to the final two years of implementation, or 50% of each reported year, after extraction.

- Fiscal years 2003-2007 do not include Deep Space infrastructure and operations costs.
- The planetary program costs for fiscal years 1968 and 1969 include development costs for the Pioneer program.
- NASA's accounting, reporting, and budget request systems have evolved, with gaps or inconsistencies in the reporting details.
- During the 1990s and early 2000s, NASA consolidated all its science activities (astrophysics, space physics, planetary science, and Earth science), resulting in a loss of the fidelity of mission data.
- Extended mission operation costs were not clearly presented until 2013, when a revision of NASA's budgeting format was implemented.
- In 2004, NASA changed to a complete cost-accounting method for all projects and included the assignment of overhead expenses directly to individual programs, improving the accuracy of cost data. Cost entries before 2004 used a rule of thumb that added 10% to costs, described as "not exact but gets you in the ballpark."
- Cost data for the years 1997 – 2001 are "Jason Callahan, Budgeting for Exploration: History and Political Economy in Space Science 1959-2010," at the AAS 45th Meeting of the Division for Planetary Sciences, Denver, Colorado. October 7, 2013" — this reference is not available to the public aside from the event summary document that lists the abstract only.

The PEBD is a cost-centric summary of the NASA interplanetary mission catalog. It lacks data critical to evaluating the triad's Schedule and Scope components. Relying on the organization of the PEBD and the associated annual NASA Budget documents, significant effort was dedicated to establishing a CR development dataset.

Development Dataset

The development Models, #1 and #2, utilize the attributes of the NIM dataset *Figure 16*. The NIM includes system triad attribute data for program, schedule, cost, and technical design (scope) attributes. Three cost silos are presented based on inflation format: 1) Then$ representing realized expenses in the FY of recording, 2) NASA’s NSII adjusted expenses, and 3) PCEPI adjusted expenses, which project Then $ to 2022 equivalent dollars for cost normalization. Indices of 1.00 were used to adjust the 2022 costs to 2023 and 2024 levels during CR development.

Algorithm Design

Initial Algorithm Exploration

The initial modeling examined the development of an exhaustive classifier as a predictor of performance. The model included 70 NASA interplanetary missions from the PEBD. Data attributes included mission, NASA Program, BoM (beginning year for mission design development), EoM (ending year of mission design development), the number of years for development (Feature 1), launch year, launch vehicle, destination, purpose, objective, spacecraft name, spacecraft mass in kilograms (Feature 2), satellite beginning-of-life (BoL) power in watts (Feature 3), mission management, the number of instruments (Feature 4), and cost elements of development (Feature 5), launch vehicle (Feature 6), operations (Feature 7), and total mission cost (Feature 8). Due to data consistency, the dataset was limited to post-1987 satellite launches. Four additional missions were pruned as they represented in-process programs with post-2023 launch manifests. Three additional missions listed as line items in NASA's budget requests were omitted due to their immaturity and early development status. The resultant dataset included 30 interplanetary missions that were classified into three cost clusters:

- Class A: $122.4 M - $520.3 M cost range (Low Cost)
- Class B: $534.6 M - $996 M cost range (Mid-level Cost)
- Class C: $1065.4 M – $5741.3 M cost range (High Cost)

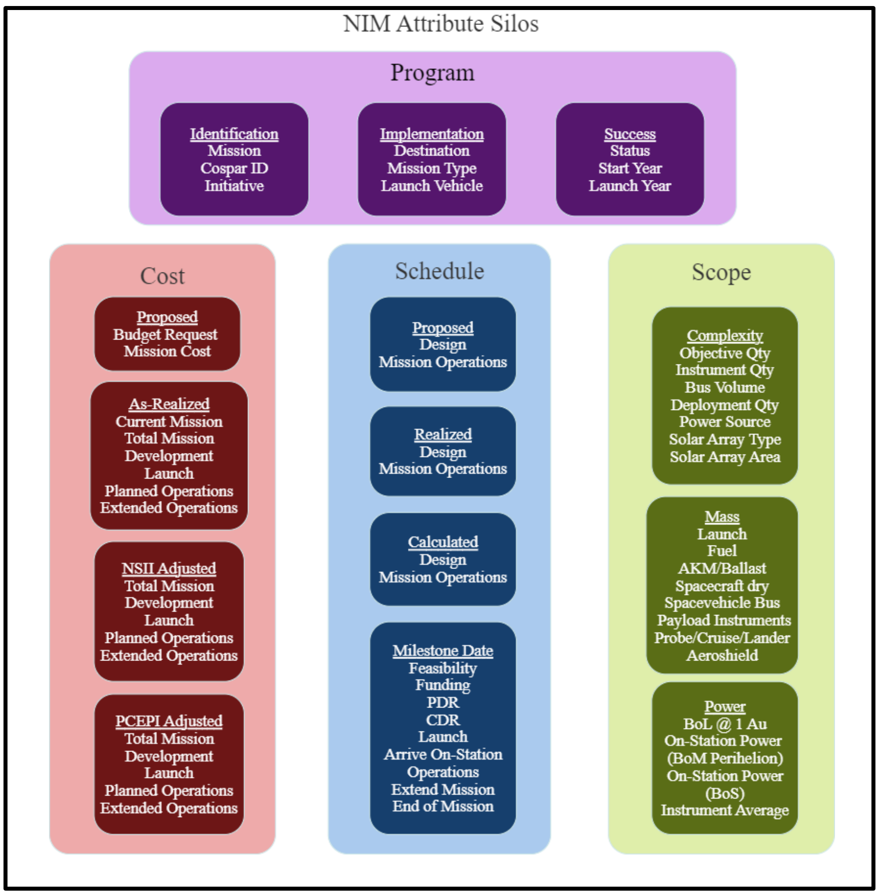


Figure 16: NIM Attributes

The three cost classes, each containing 10 missions, evenly distribute the data. An exhaustive classifier was constructed to evaluate the performance of feature (F) and feature combination (FC) predictions. FC included the product and the ratio of feature pairs *Table 11*. Feature 7 was removed from the FC options because data element 2 of the dataset contained a $0 value, resulting in a divide-by-0 error within the model. Rather than pruning element 2, Feature 7 was omitted from the FC classifiers. This is justified because Feature 7 is a redundant variable, given that Feature 8 is the sum of Features 5, 6, and 7.

Table 11: Derived Feature Combinations for the Classifier Model

Feature	2	3	4	5	6	7	8
1	✓	✓	✓	✓	✓	X	✓
2	-	✓	✓	✓	✓	X	✓
3	-	-	✓	✓	✓	X	✓
4	-	-	-	✓	✓	X	✓
5	-	-	-	-	✓	X	✓
6	-	-	-	-	-	X	✓
7	X	X	X	X	X	X	X
8	-	-	-	-	-	X	-

For a classifier model, eight features F and twenty-one feature combinations FC were evaluated for classifier prediction. The evaluation of F and FC considered the class cluster mean, standard deviation, and coefficient of variance for each. Calculation of $t-score$ and $p-value$ was performed for each class comparison (A to B, B to C, A to C). Results with a $p-value < 0.05$ were considered significant and highlighted for further evaluation.

The $CPT_{i\text{-}j}$ and nSTD- $CPT_{i\text{-}j}$ for each F/FC considered were determined. The values of ΣnSTD-$CPT_{i\text{-}j}$ and $\Sigma P_{t\text{-}score}$ were determined to select the "best F/FC" for classification modeling. The "best F/FC" selection was determined from the normalized sum squared of the nSTD- CPT_k values. Model F/FC calculations were determined independently for each class assigned to data sample clusters. Weighting was calculated using the $z-value$ for each sample, as determined by F/FC. $Z-value$ was calculated by class group relative to each class group mean and standard deviation using:

$$Z_{value} = \frac{(x-\mu)}{\sigma} \tag{21}$$

Inclusion of $\frac{1}{\sqrt{n}}$ was unnecessary as each class contained an identical set of clustered values ($n = 10$). $f(z) = \frac{1}{z^2}$ was determined for each cluster, by Feature, against each cluster class A, B, and C. The $f(z)$ values were normalized, and then each normalized $f(z)$ value was weighed. Each cluster was then evaluated by summing the weights within each cluster for the selected Feature. The maximum value was then applied to each cluster, and each cluster element was assigned a class label of A, B, or C. The class-label determination used binary features and a total weight for model evaluation.

Exploration Modeling Conclusions

A classifier-based prediction model demonstrated improved performance compared to traditional prediction approaches. The approach also identified limitations within a classifier-only predictor model approach:

- The classifier model approach attempts to define CER-type relationships between the system predictor input and a response system output using system data. The data is derived from the system data attributes and transformed into F/FC.
- The F/FC to output response relationship, which is evaluated using weighting, is combined as a components that sum to the prediction.
- The prediction, based upon clustering, attempts to identify class assignment based on a one-dimensional variable. Although a relationship plot is n-dimensional, classification seeks to identify an n-dimensional centroid of the variable. In a multi-dimensional model, the exploration model is three-dimensional, yielding a radius vector that determines the class assignment based on the minimum distance to the centroid.

- Application of clustering for classification and prediction is non-traditional.
- Traditional approaches use multivariate regression analysis almost exclusively.
- Regression analysis is also an n-dimensional predictor.
- The exploration model raised a question about how to infuse the results of the classifier predictor within a regression-based predictor, the basis for classified regression.

Predictor Algorithm Data Approach

The approach for developing the predictor algorithm follows the Systems Engineering Process (SEP). A system is composed of the elements of the system triad *Figure 5*. Successful modeling requires establishing a set of standard attributes that are input to the MBSE tool. The tool creates F/FC, used to model and predict the enterprise resources needed to achieve the system's lifecycle goals. The block diagram of the MBSE data flow is presented *Figure 17*.

Model input and output data include attributes, features, and resources:

- Attributes — are the data elements that define the CSoS or NIM dataset. The dataset includes the published program and engineering definition of each NIM element, including cost, schedule, and scope measures that describe the NIM.
- Features — are MBSE data elements derived from the mission attributes, to develop attribute-to-resource-need relationships through parametric construction. The parametric construction is transformed into parametric regression analysis and parametric classification. Intermediate results of parametric classification are incorporated within the parametric regression analysis for correction, and the intermediate results of both the parametric regression and classification are input to the Classification/Mission Adjustment

algorithm to provide a pseudo-Cost Estimation Relationship (CER) for "application unique" data modification.

- Resources — are Enterprise data value points used to evaluate candidate mission performance: total cost, total schedule, and total technical needs.

Resources are traditionally measured with respect to the requirements and needs of the enterprise to achieve mission success, where:

$$Mission\ success = total\ cost + total\ schedule \tag{22}$$

Optimizing a system requires measuring and comparing how well the initial predictor performs with respect to alternative architectures. Three value-based combinations of mission effectiveness define this:

- Practicality — a compromise or the dependency between Cost and Technical measures (probability of success or Ps),
- Feasibility — a compromise or the dependency between Cost and Schedule (heritage versus new), and
- Complexity — a compromise or the dependency between schedule and technical (defined by Technology Readiness Levels (TRL).

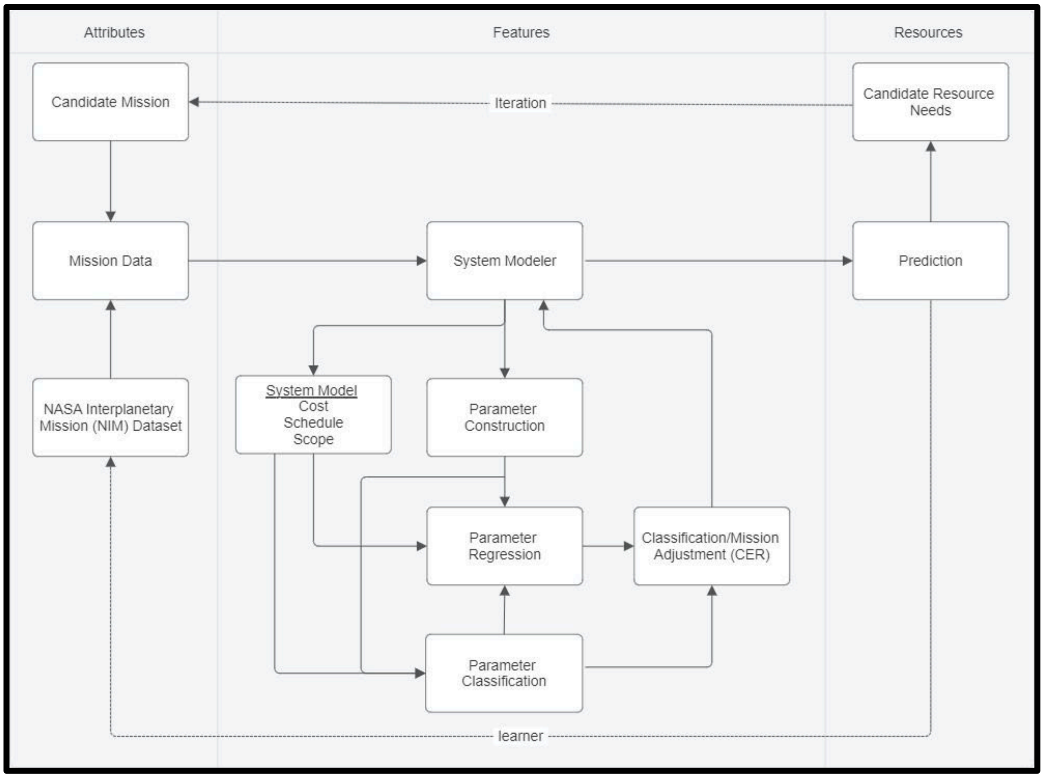


Figure 17: MBSE Data Input/Output Plan Diagram

Predictor Development Approach

The standard used for space system assessment relies on the relationship between the mass of the space vehicle and the mission cost. Spacecraft mass has two common interpretations: the total launched mass, which includes fuel and encapsulation, and the spacecraft dry mass, which represents the core spacecraft and payload systems. Mission cost typically refers to the cost of developing all space system components, including the spacecraft bus and spacecraft payload. The launch vehicle and operational systems are maintained as separate budget line items with alternative estimates. During data collection, it is imperative to identify the mass and cost reference used in the data source.

Based on those standards, most estimations focus on the relationship between SCdry mass or its associated components and total development cost or its associated components. Traditionally, SCdry is the predictor variable, and the price is the dependent variable, as most space system modeling is focused on providing cost estimates. This standard was evaluated during the initial development of the algorithm. An assessment was conducted to determine each NIM attribute's predictive value for all remaining attribute response variables. As a result of that analysis, along with developing a predictor algorithm that supports all components of the system triad, the algorithm development utilizes SCdry as the output response variable, representing the total development cost in the year of occurrence dollars (TDCT).

Predictor Development Variable Assessment

The variable assessment considered the attributes of the NIM data set. Data was extracted directly from the NIM values, such as launch mass and SCdry, or derived from the NIM data, such as development schedule duration. The development schedule duration was calculated as the difference, in months, between the mission's Phase A selection or award date and the launch date. All NIM missions are identified with some form of Announcement of Opportunity procurement mechanism, regardless of the sourcing method. The NASA budget request does not include funding for mission development, concept development, or initial feasibility studies. The costs associated with those activities are not included in the NIM and are not defined in the model. For schedule durations, the initiation of a mission is identified as a program-documented start, contract award, or identification as a NASA budget line item.

All of the NIM data is "like data," sharing standard units and interpretation. Plots of the raw data were generated for visualization; an example is shown in *Figure 18*. The Total Mission Cost (TMC) includes development, launch, and primary operations costs. In *Figure 18*, costs are adjusted using the NSII inflation values to be priced in FY2023 dollars. The launch mass represents the spacecraft's mass property in the launch configuration, with the vehicle fully fueled and encapsulated. The NIM missions are identified by mission type: Flyby, Impact, Lander, Orbiter, Rendezvous, Rover, or Sample Return. This distinction was included in the variable assessment to identify trends in non-numeric variables. Element labeling identifies the NIM elements depicted in *Figure 18* and *Figure 19*.

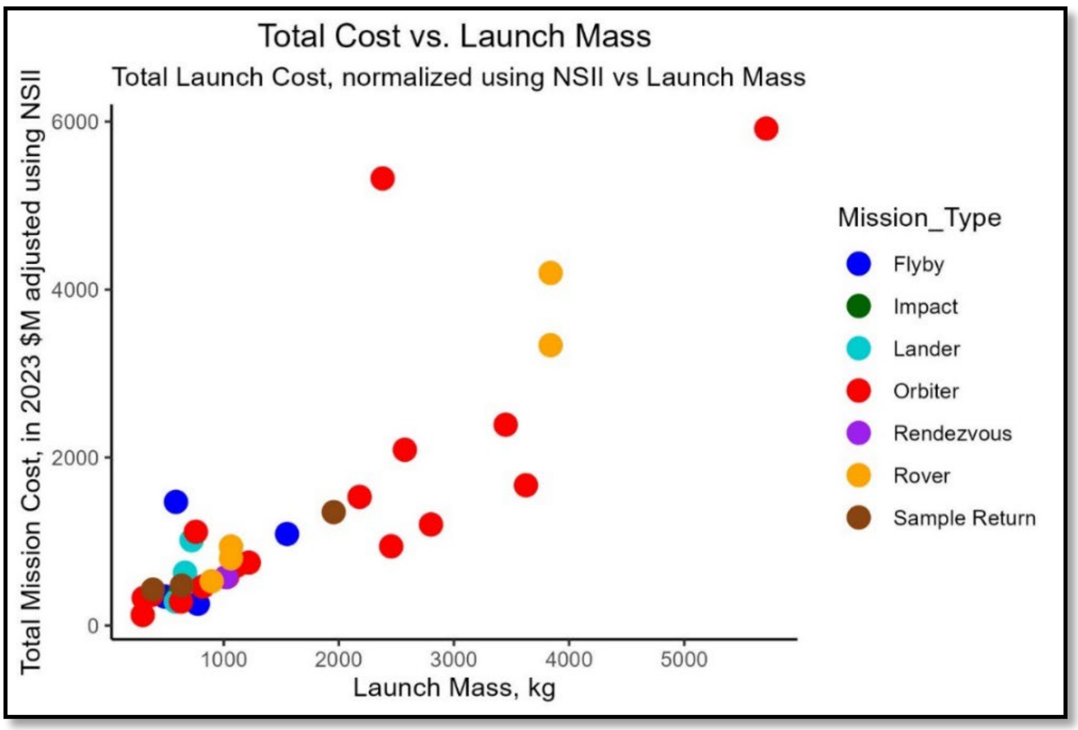


Figure 18: Initial Plotting of NIM Data Attribute Relationships

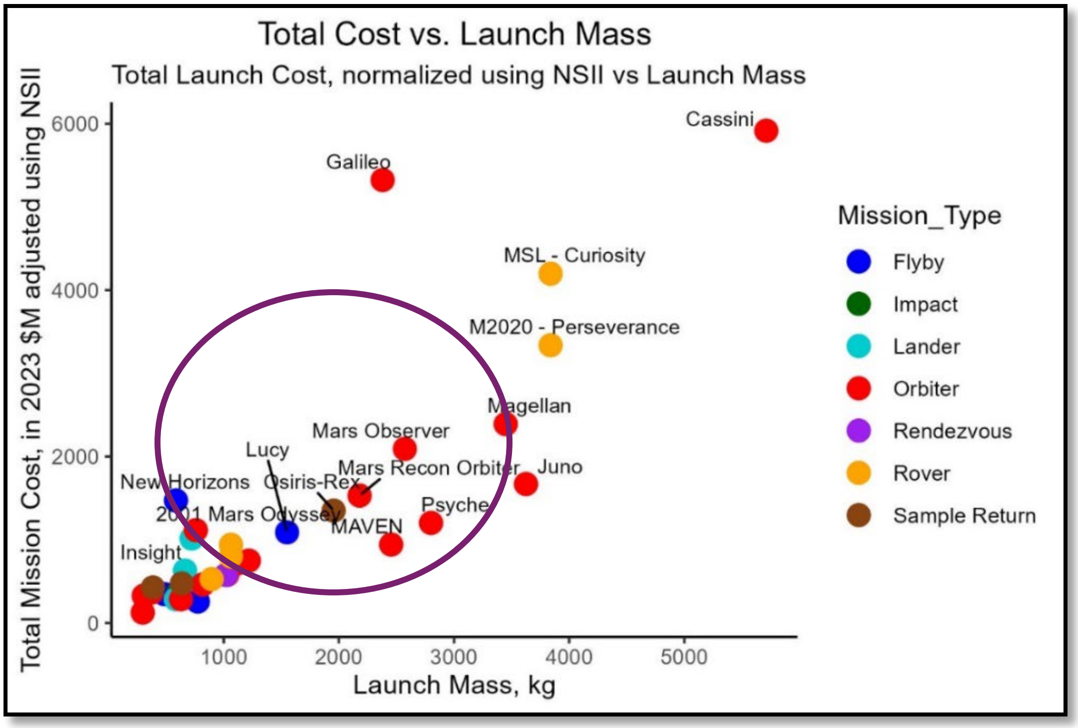


Figure 19: NIM Missions, Upper Scale, Attribute Models

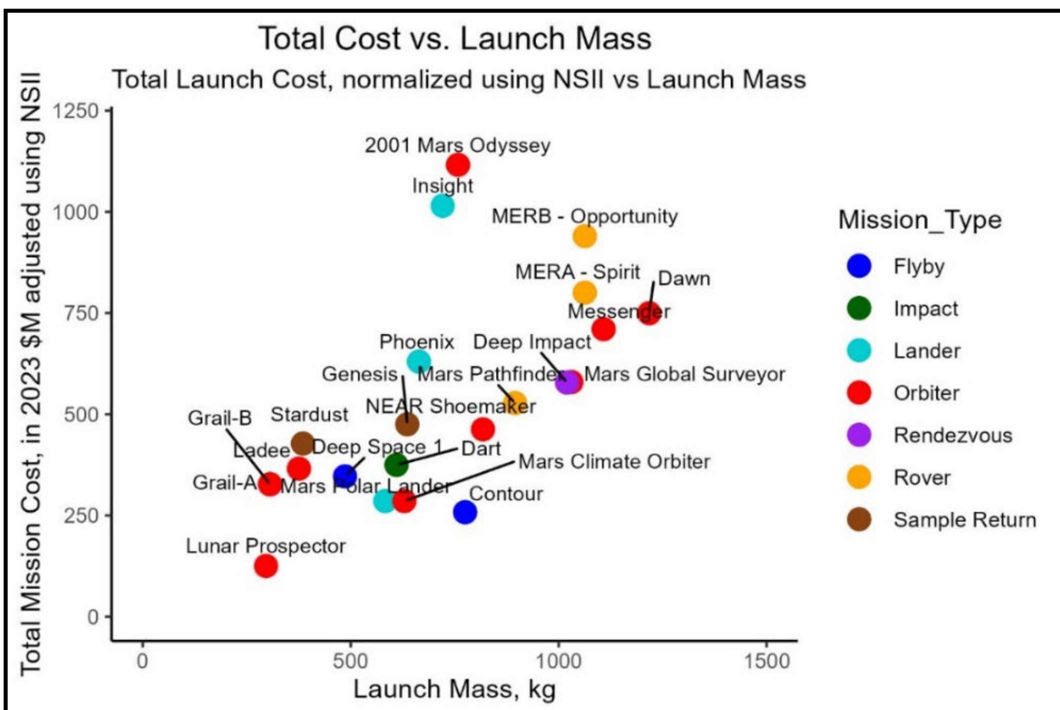


Figure 20: NIM Missions, Lower Scale, Attribute Models

For each attribute pair, the linear regression solution and $R^2$is determined *Figure 21*. This data is collected and organized in a heat matrix of variable relationships. The initial variable assessment focused on mass and cost attributes in the NIM dataset. Mass data includes four mass elements:

- Total Launch Mass,
- Spacecraft Dry Mass,
- Space Vehicle Mass (Spacecraft bus mass), and
- Spacecraft Payload mass.

Cost data includes five cost elements and three valuation methods. The valuation methods included:

- Then $, a summation of actual annual costs,
- NASA's NSII annually adjusted costs, and
- PECPI annually adjusted costs.

The elements of cost included:

- Total Cost is the sum of Development, Launch, and Operations elements
- Total formulation and development costs are summed as Development cost
- Launch vehicle and launch operations summed into Launch costs
- Operation cost is segregated into the primary mission operations and extended mission operations elements based on the mission extension date.

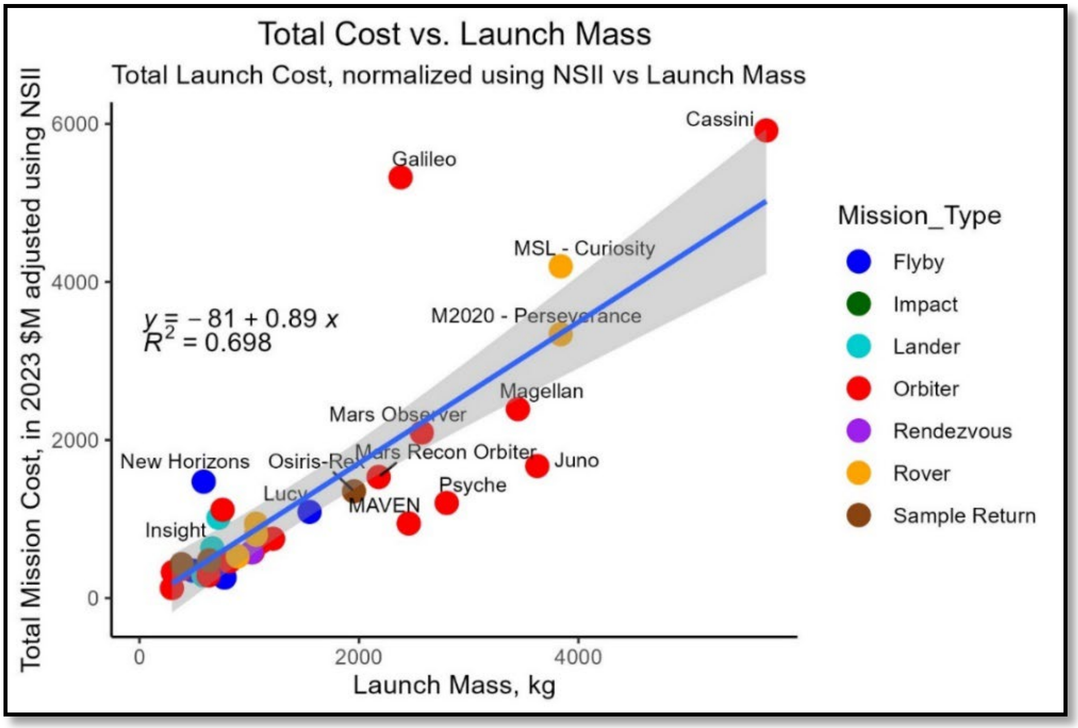


Figure 21: Traditional Linear Regression Model

The portion of the attribute correlation heat map is included *Table 12*. The table is colorized for trend evaluation. Analysis of the table data results provides insight into the mass and cost attribute correlation as clusters of information:

- $\mu = 0.44$ (for all values reported),
- $\mu = 0.92$ (for all green values),
- $\mu = 0.83$ (for all yellow values),
- $\mu = 0.65$ (for all orange values),
- $\mu = 0.85$ (for all green and yellow values), and
- $\mu = 0.72$ (for all green, yellow and orange values).

Based on the heat map of linear regression analysis, trends are observed:

- SCdry and SVbus mass provide the highest-valued correlation with cost using linear regression.
- From the cost perspective, total mission cost and development cost provide the most significant measure of correlation with mass.
- Launch mass and SVpayload mass do not correlate well with cost.
- Launch costs, primary operations costs, and extended operations costs do not correlate to mass.

Table 12: Variable Assessment Heat Map

	Cost	Launch Mass		S/C_{DRY} Mass		S/V_{BUS} Mass		$S/V_{PAYLOAD}$ Mass	
Then $	Total Mission	1	0.77	1	0.869	1	0.876	1	0.636
		mass = 400 + 1.4 * cost		mass = 170 + 0.84 * cost		mass = 120 + 0.76 * cost		mass = 400 + 1.4 * cost	
	Development	1	0.639	1	0.904	1	0.944	1	0.822
		mass = 580 + 2.0 * cost		mass = 180 + 1.5 * cost		mass = 130 + 1.4 * cost		mass = 580 + 2.0 * cost	
	Launch	1	0.647	1	0.428	1	0.437	1	0.15
		mass = 170 + 11 * cost		mass = 230 + 4.8 * cost		mass = 170 + 11 * cost		mass = 170 + 11 * cost	
	Primary Operations	1	0.509	1	0.388	1	0.382	1	0.119
		mass = 840 + 6.4 * cost		mass = 500 + 2.8 * cost		mass = 840 + 6.4 * cost		mass = 840 + 6.4 * cost	
	Extended Operations	1	0.621	1	0.703	1	0.698	1	0.273
		mass = 770 + 6.4 * cost		mass = 390 + 3.5 * cost		mass = 770 + 6.4 * cost		mass = 770 + 6.4 * cost	
NSII Inflation Adjusted	Total Mission	1	0.698	1	0.656	1	0.647	1	0.273
		mass = 520 + 0.78 * cost		mass = 320 + 0.39 * cost		mass = 520 + 0.78 * cost		mass = 520 + 0.78 * cost	
	Development	1	0.671	1	0.719	1	0.724	1	0.334
		mass = 540 + 1.2 * cost		mass = 290 + 0.68 * cost		mass = 540 + 1.2 * cost		mass = 540 + 1.2 * cost	
	Launch	1	0.427	1	0.246	1	0.245	1	0.489
		mass = 710 + 3.8 * cost		mass =490 + 1.4 * cost		mass = 710 + 3.8 * cost		mass = 710 + 3.8 * cost	
	Primary Operations	1	0.461	1	0.36	1	0.344	1	0.029
		mass = 950 + 3.5 * cost		mass = 550 + 1.6 * cost		mass = 950 + 3.5 * cost		mass = 950 + 3.5 * cost	
	Extended Operations	1	0.616	1	0.686	1	0.633	1	0.071
		mass = 760 + 5.2 * cost		mass = 390 + 2.8 * cost		mass = 760 + 5.2 * cost		mass = 760 + 5.2 * cost	
PECPI Inflation Adjusted	Total Mission	1	0.745	1	0.735	1	0.727	1	0.222
		mass = 440 + 0.95 cost		mass = 260 + 0.50 cost		mass = 440 + 0.95 cost		mass = 440 + 0.95 cost	
	Development	1	0.699	1	0.814	1	0.825	1	0.424
		mass = 490 +1.5 * cost		mass = 230 +0.89 * cost		mass = 490 +1.5 * cost		mass = 490 +1.5 * cost	
	Launch	1	0.488	1	0.292	1	0.292	1	0.049
		Mass =560 + 5.2 * cost		Mass =430 + 2.0 * cost		Mass =560 + 5.2 * cost		Mass =560 + 5.2 * cost	
	Primary Operations	1	0.457	1	0.349	1	0.334	1	0.071
		Mass = 920 + 4.0 * cost		Mass = 540 + 1.8 * cost		Mass = 920 + 4.0 * cost		Mass = 920 + 4.0 * cost	
	Extended Operations	1	0.610	1	0.686	1	0.667	1	0.233
		Mass = 770 + 5.4 * cost		Mass = 390 + 2.9 * cost		Mass = 770 + 5.4 * cost		Mass = 770 + 5.4 * cost	
		$R^2 > 0.9$		$0.9 > R^2 > 0.75$		0.75 > R2 > 0.65		$R^2 < 0.65$	

The heat map indicates a focus on SCdry and SVbus mass attributes, while Total Mission and Total Development cost attributes provide the focus for initial predictor development. The attribute correlation heat map is included *Table 13*. When comparing equivalent NSII-adjusted and PECPI-adjusted costs, the PECPI-adjusted costs provide overall improvement to the "goodness of fit" of the linear regression modeling. A measure of the improvement is $\mu_{improved} = 0.129$ and $\sigma_{improved} = 0.009$. The $\mu_{NSII} = 0.687$ and $\mu_{PECPI} = 0.775$ *Table 13*. This demonstrates a 28.3% improvement in linear regression "goodness of fit" using a R^2 Comparison method indicating that the PECPI-adjusted cost provides an improvement over the NSII-adjusted cost for a linear predictor model.

Table 13: Refined Mass and Cost Attribute Heat Map

	Cost	S/C_{DRY} Mass		S/V_{BUS} Mass	
Then $	Total Mission	1	0.869	1	0.876
		mass = 170 + 0.84 * cost		mass = 400 + 1.4 * cost	
	Development	1	0.904	1	0.944
		mass = 180 + 1.5 * cost		mass = 580 + 2.0 * cost	
NSII Inflation Adjusted	Total Mission	1	0.656	1	0.647
		mass = 320 + 0.39 * cost		mass = 520 + 0.78 * cost	
	Development	1	0.719	1	0.724
		mass = 290 + 0.68 * cost		mass = 540 + 1.2 * cost	
PECPI Inflation Adjusted	Total Mission	1	0.735	1	0.727
		mass = 260 + 0.50 cost		mass = 440 + 0.95 cost	
	Development	1	0.814	1	0.825
		mass = 230 +0.89 * cost		mass = 490 +1.5 * cost	

When comparing equivalent PECPI-adjusted costs with the year incurred (Then $) costs, the Then $ costs provide a significant improvement in "goodness of fit" of the linear regression modeling, with an improvement in the mean $\mu_{improved} = 0.161$ and $\sigma_{improved} = 0.042$. The $\mu_{Then-\$} = 0.898$, providing a 54.7% improvement in linear regression "goodness of fit" using the R^2 factor compared to the PECPI-adjusted values.

For this analysis, "goodness of fit" is defined as the decrease in "badness of fit" or $(1 - R^2)$. The measure is defined as:

$$I = \frac{(1 - R^2)_{NSII\ Total\ Mission-SC\ dry\ mass} - (1 - R^2)_{Then\ \$\ Total\ Mission-SC\ dry\ mass}}{(1 - R^2)_{NSII\ Total\ Mission-SC\ dry\ mass}} \tag{23}$$

The "badness of fit" measure provides a metric for the improvement scale when comparing one cost or mass classification to others.

Predictor Algorithm Development

A series of preliminary algorithms is developed to evaluate their performance as predictors of the SCdry response variable as a function of the TDCT. The algorithm configurations included:

- Initial exploration model,
- Simple linear regression model,
- Simple classifier model,
- Classified regression,
- Classified regression with residual class correction,
- Polynomial regression,
- Classified polynomial regression, and
- Classified polynomial regression with residual class correction.

Polynomial regression complicated the model by increasing the model's *dof* at a comparable level of performance to classified regression. Simple regression demonstrated the lowest value for R^2. CR demonstrates significant improvement over simple linear regression, and incorporating a residual classification compensator yields considerable gains over alternative models and current performance standards for parametric modeling.

Classified Regression Design

The CR process begins with attribute selection and identification of the decision boundary, similar to the SVM. This boundary provides a two-dimensional centroid axis for the data pairs and serves as the initial model predictor (prediction #1) for regression classification. Symmetric margin boundaries are established around the decision boundary, as shown in *Figure 22*, based on the percent error.

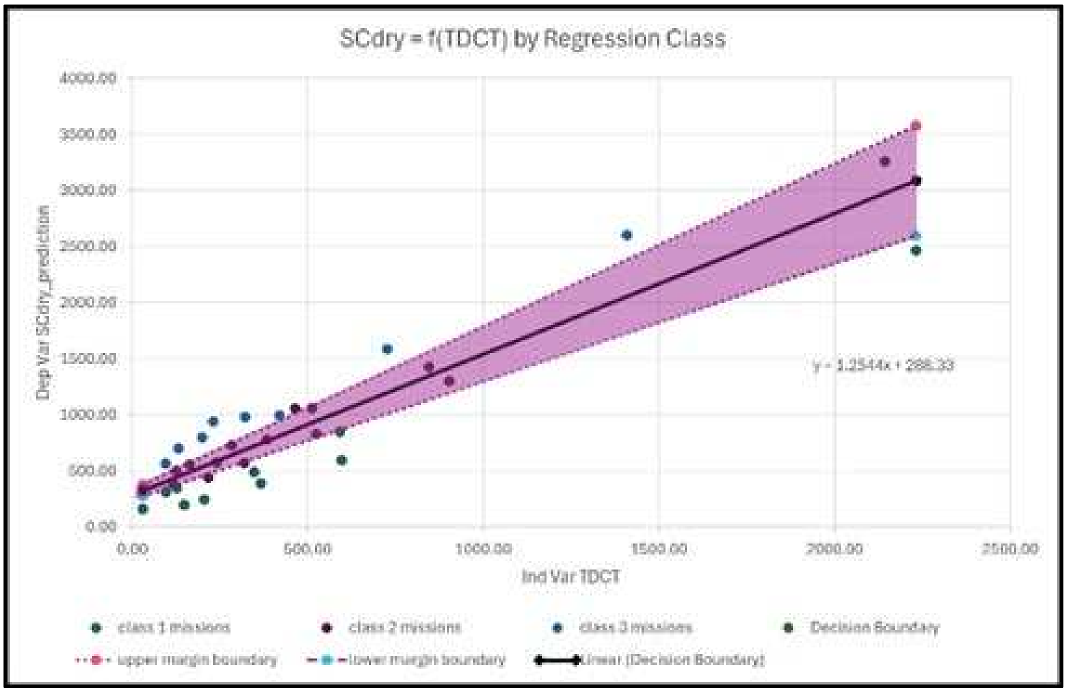


Figure 22: SVM Decision Boundary with Margin Boundaries

The margin boundaries define three regions containing the upper, mid, and lower striation data clusters. Each cluster forms a class for a regression solution. The margin boundaries expand or contract, reassigning classifications to boundary elements, until the model is locally optimized (1) for regression performance. Local optimization solidifies the regression class for each data element. The regression class solutions provide prediction #2, *Figure 23*.

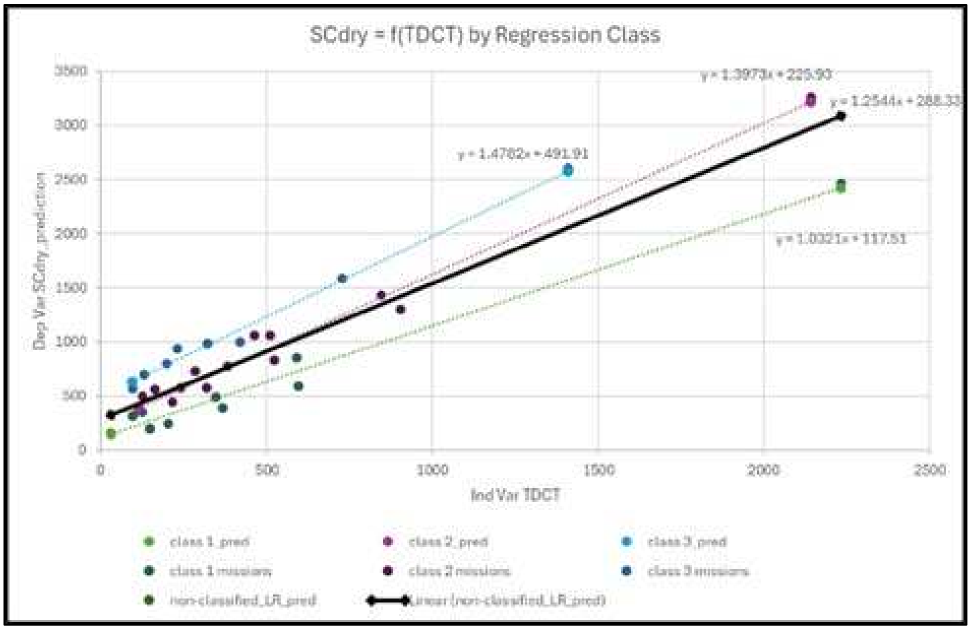


Figure 23: Classified Regression Predictors (Prediction #2)

Data residuals are determined from prediction #2. The SVM classification is repeated for the residual adjusted as a percentage of the predictor *Figure 24*. The residual margin boundaries identify the homoscedastic region in which the residuals are independent of the predictors. The upper and lower margin boundaries identify heteroscedastic increasing (> + threshold) and heteroscedastic decreasing (< - threshold).

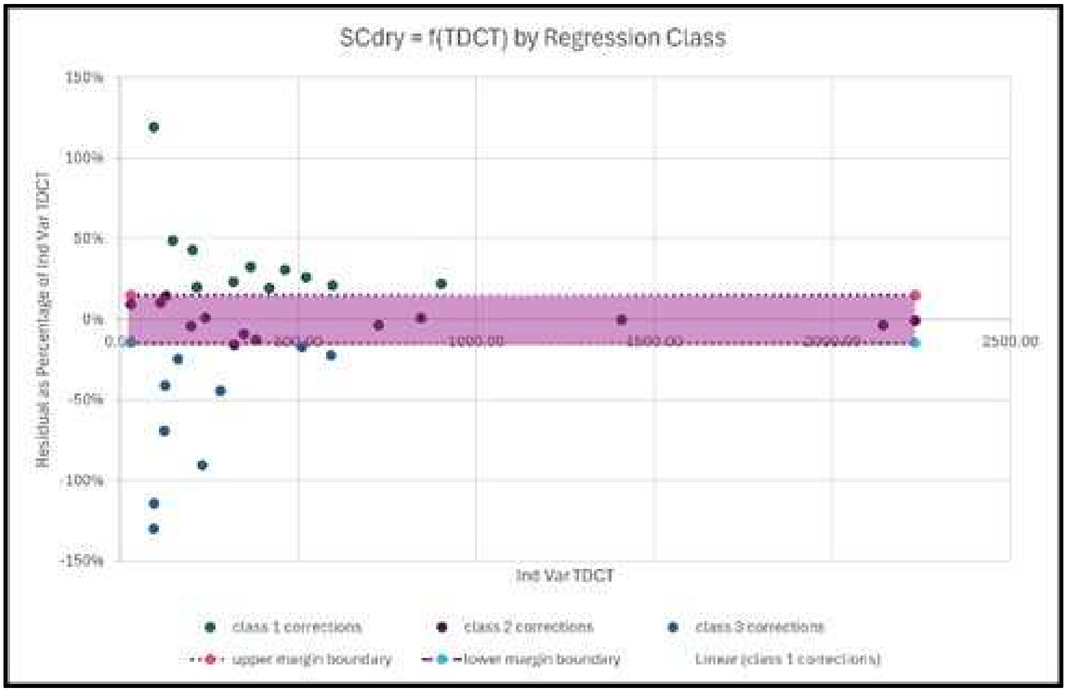


Figure 24: Percent Residual Error with Residual Margin Boundaries

The elements are classified based on their residual, using the margin boundaries. Residual corrections are determined for each residual class as correctors *Figure 25*. The margin boundaries expand or contract, reassigning classifications to boundary elements, until the model is locally optimized (1) for residual performance. Local optimization solidifies the residual class for each data element. The residual class solutions provide correction #1.

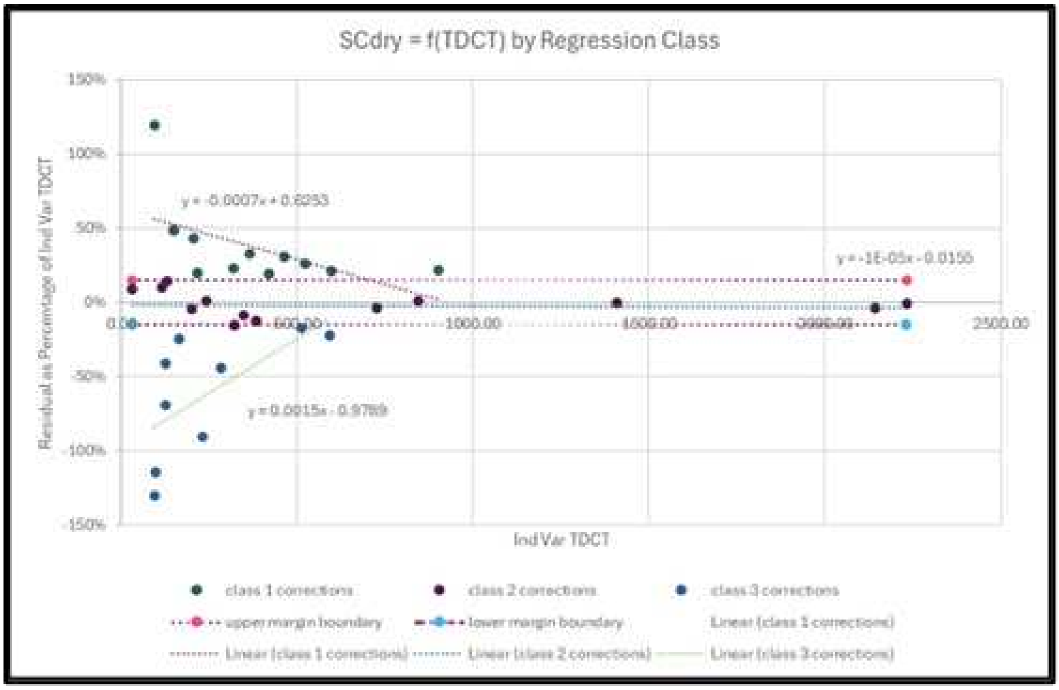


Figure 25: Residual Class Correctors

For each data element, prediction #2 is corrected by subtracting the element's residual correction from the regression class predictors, producing prediction #3. The regression and residual classification process is repeated until global optimization (1) is achieved. Global optimization occurs when the (1) achieves a model-wise minimum or best predictor performance. The prediction #3 final equations complete the CR class-wise *Figure 25* or elementwise *Figure 26* predictor process. *Figure 26* represents a series of predictor curves for the Mars Reconnaissance Orbiter (MRO) mission. The lower curve is the MRO predictor. The MRO predictor is less than the class |μ| predictor; therefore, the class |μ| is used as a conservative value, representing a 50% confidence level. Based on the user confidence level input, the estimate is shifted for uncertainty to the upper curve. Figure 6 includes a 75% confidence level.

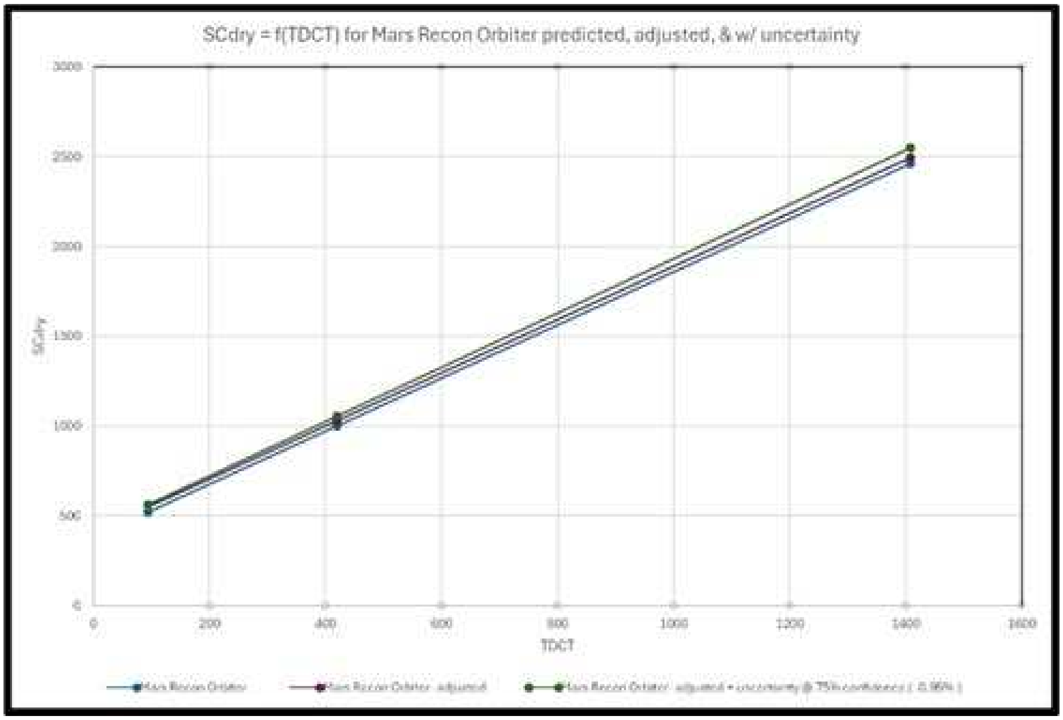


Figure 26: MRO Predictor with Uncertainty

Classified Regression Segmentation

The construction of a CR model constructs three levels of prediction utilized by the Funding System based on the user's needs and use case for the predictor. Blue boxes represent algorithm modules or enhancements to predictor accuracy. The complete classified regression model structure is presented in the subsequent discussion of *Figure 37*. Classified regression modeling is constructed from three segments: a linear regression predictor from the SVM and margin boundaries, a set of classified regression predictors, and a set of classified residual correctors.

The first level provides model-wise linear regression prediction, primarily used for the initial classification and system-level comparison. This module is comparable to traditional parametric predictors. Typical of most linear regression predictors, the initial predictor module's accuracy is nominal to poor. This constitutes prediction #1 *Figure 27*.

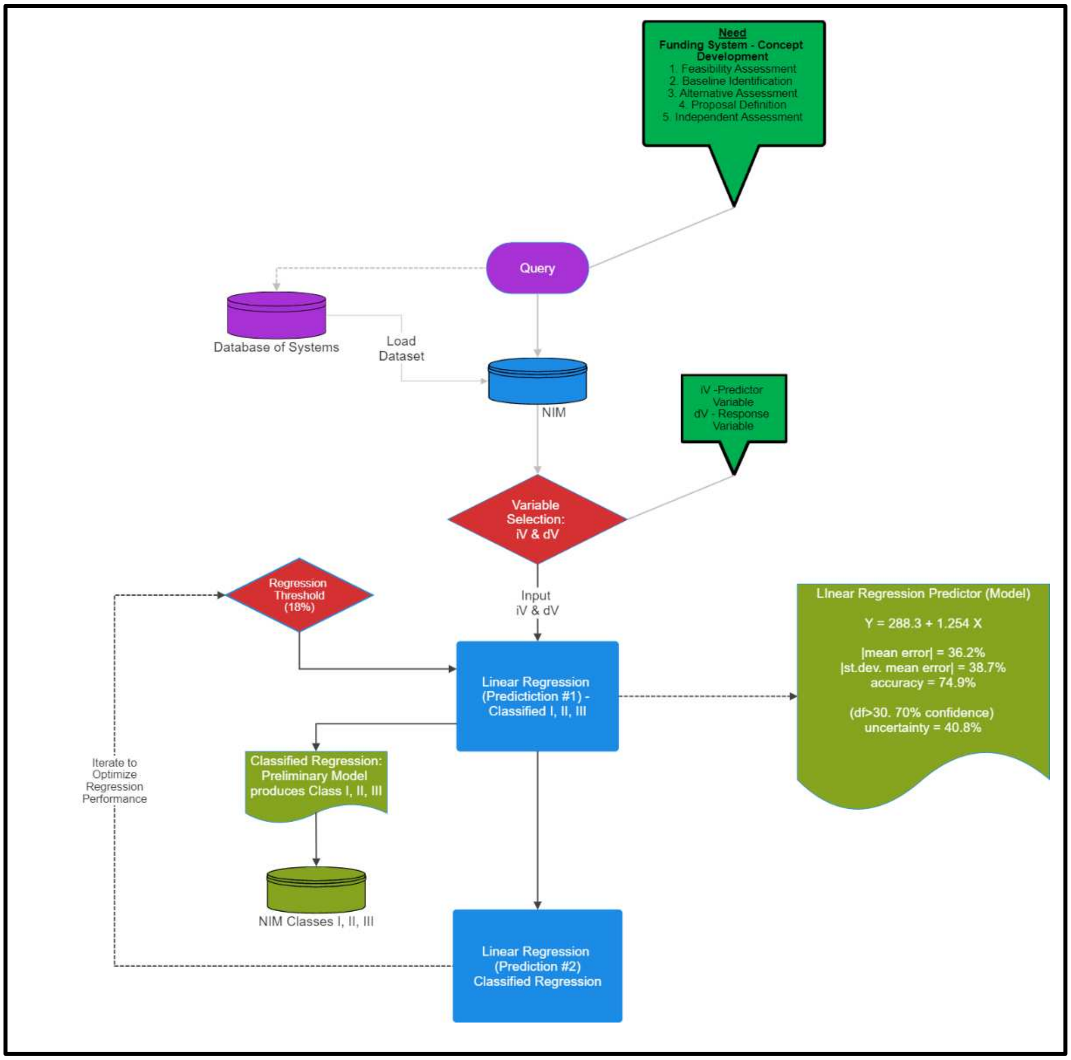


Figure 27: Linear Regression CR Modeler Segment

The second (mid) algorithm uses the classification from prediction #1 and determines class-wise regression solutions. This level of prediction provides a comparison of system acquisition mechanisms. The elements of a class may exhibit characteristics that indicate that one acquisition mechanism is superior to an alternative. The predictor can also be applied to element-specific analyses to establish a baseline or an alternative assessment. This model segment provides a predictor for comparative performance assessment. This is prediction #2 *Figure 28*.

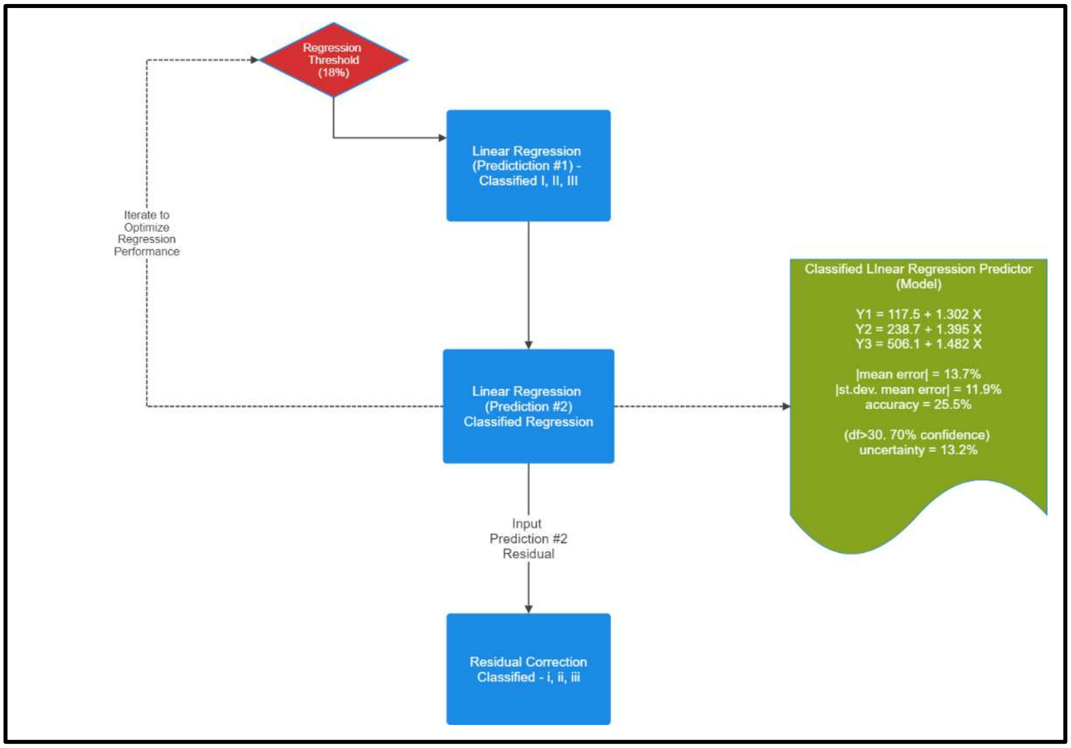


Figure 28: Classified Regression CR Modeler Segment

The third (bottom) algorithm takes the classified regression predictions and, operating in the residual space, provides a residual correction to the model. The output of the third module is a highly functioning, class and element-wise predictor. This is the interpretable (White Box) predictor. This is prediction #3 *Figure 29*.

A fourth algorithm evaluates the full data set attributes, generating features and feature combinations to provide a supervised cluster classifier. The cluster classifier allows the predictor to be applied to unknown, external data, providing forecast capability. The predictor model operates independently of the use case.

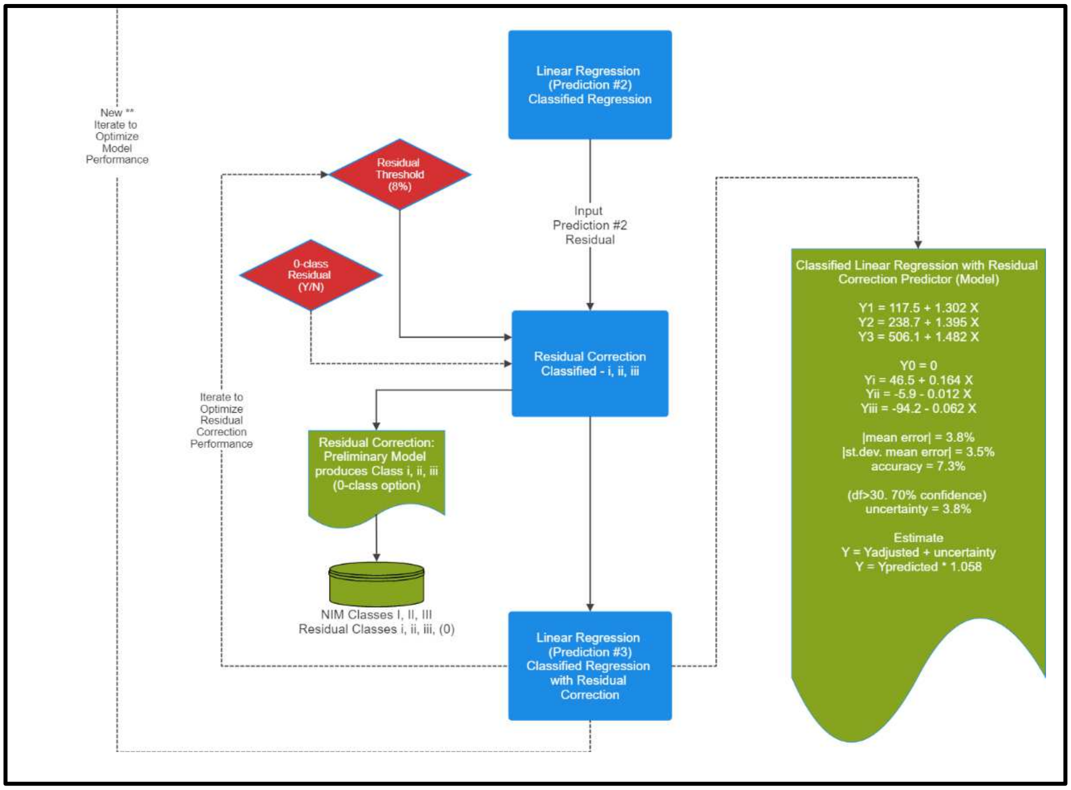


Figure 29: Classified Regression with Residual Correction CR Modeler Segment

CR supports a range of Funding System prediction and forecasting use cases, including:

- Feasibility assessment and strategic planning
- Baseline development and resource estimation
- Alternative assessment
- Proposal generation and validation
- Independent assessment
- Monitoring, growth identification, and risk mitigation

For each use case, CR is well-suited to provide quality and robust prediction and forecasting. Use cases support needs internal to the Enterprise, internal to the program of interest, or as an independent assessment or monitor. The application of performance measures, accuracy, and uncertainty is explicitly defined in relation to the use case and applicable standards. The Funding

System assessment process is presented *Figure 30*. The current framework processes the need into a query that establishes a baseline, implements an estimation process or set of processes, and produces the requested documentation. CR offers an alternative to the existing process flow.

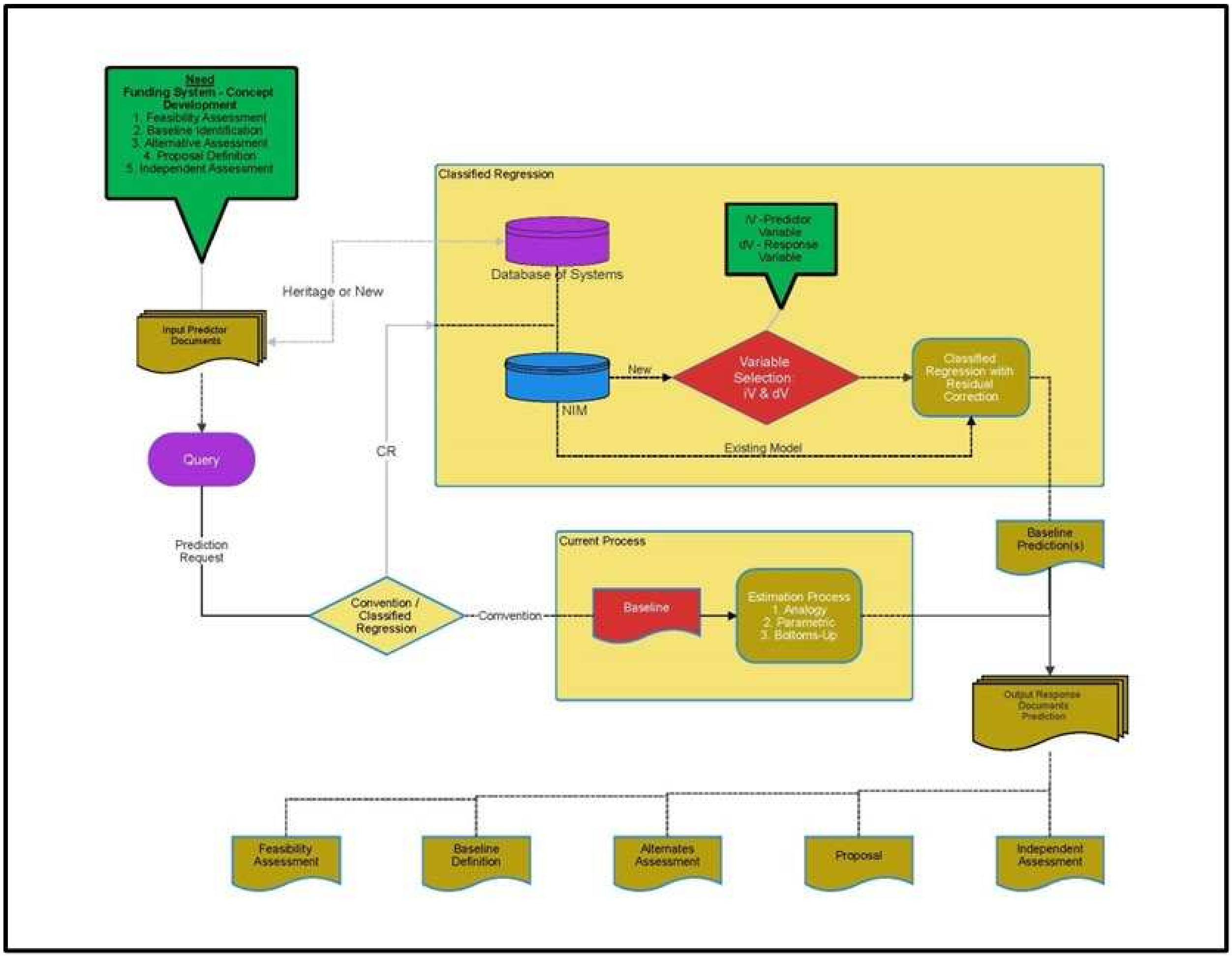


Figure 30: Funding System Assessment Process

The CR process is initiated with a funding system query. The query is passed along to CR, which evaluates the relevant database of system solutions. The CR process organizes the query attributes (inputs), prioritizes the query resources (outputs), and defines an optimal prediction path. For a simple prediction, this involves modeling a predictor-response relationship and interpreting a single model. For a more complex system assessment, this involves constructing a predictive neural network using available attributes. A benefit of CR is that the baseline is an output of the CR process, rather than an input requirement of the current process. This allows greater flexibility in defining an optimal solution and avoids inherent bias during the assessment process.

An example of a predictor network is presented *Figure 31* for the attribute SCdry as the response variable.

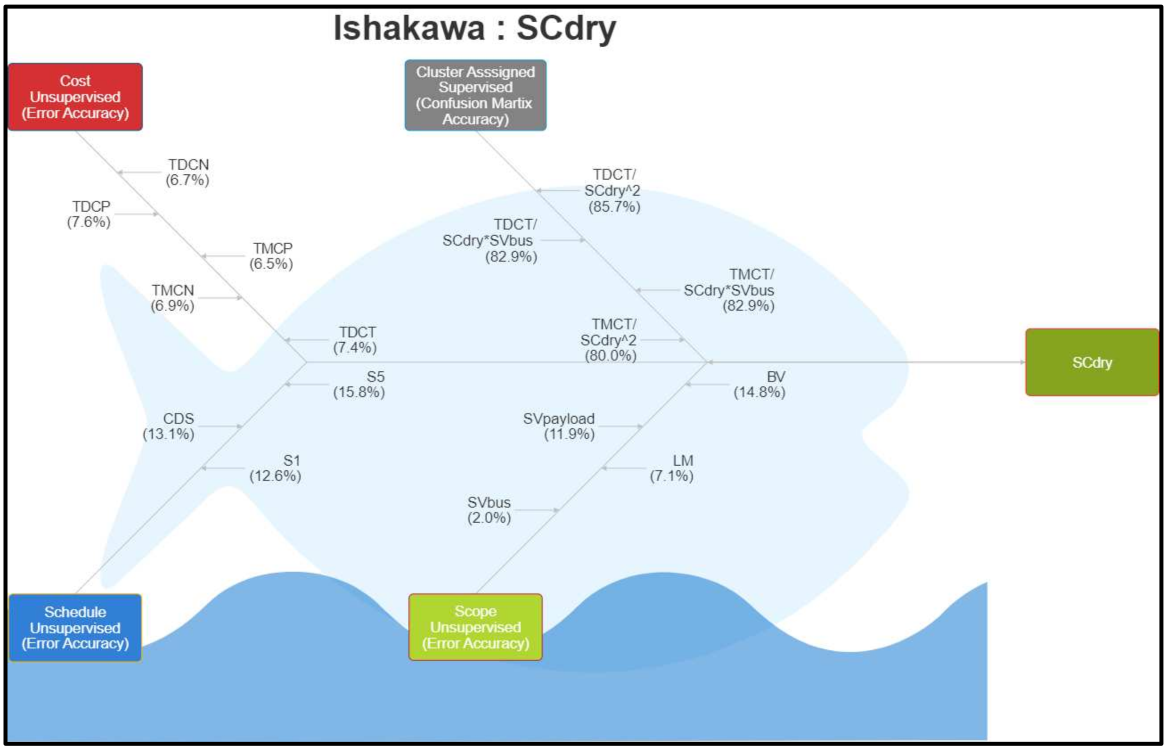


Figure 31: Predictor Network for SCdry Response

The SCdry is a simplified predictor network consisting of prediction paths emanating from the Scope, Schedule, or Cost components of the system triad. A fourth path is included for supervised classification, providing access to the predictor network for predictions not based on an existing NIM element. CR models are constructed for each element of the NIM to predict the SCdry response. The four best performance predictors are included for each triad component. Accuracy is provided for the supervised classifier models. From the predictor fishbone, multiple paths are available for SCdry response prediction, allowing a system model to be further tailored through bootstrap aggregating, boosting, and confidence propagation.

Classified Regression Modeler

Based on the success of developing a classification regression algorithm as a predictor, an automated CR modeler was constructed in Microsoft Excel. Excel was selected over Python or R-studio due to the application's portability, accessibility, and ease with which the modeler can be adapted and evaluated for alternative applications.

User interface defines:

- The selection of the independent predictor variable from the dataset of NIM attributes
- The selection of the dependent response variable from the dataset of NIM attributes
- The regression threshold value for margin definition is symmetric about the regression decision boundary. The regression threshold is iteratively selected based on localized optimization for model accuracy.
- The residual threshold value for margin definition is symmetric about the residual decision boundary. The residual threshold is iteratively selected through localized optimization to enhance the model's accuracy.
- Option of a zero-class margin definition, symmetric about the residual decision boundary, where the residual correction is omitted. These values represent initial regression predictions that are "good enough," and omitting the correction improves overall model performance. The residual for each of the zero class elements is included in the determination of the residual class correctors.

- The iteration of the regression threshold, residual threshold, and zero-class margin definition as a user-input loop to optimize the model performance. Optimization is defined as the minimum model error.
- Selection of accuracy performance thresholds at a level 1 and level 2. The performance thresholds enable the visualization of model performance across all elements as the model margins are evaluated for optimization.
- As a model feature, a confidence level (CL) option is provided for user identification of class-wise uncertainty calculations. Options include CL = {70%, 75%, 80%, 85%, 90%, 92%, 96%, 98%, 99%}. The confidence level identifies the uncertainty based on the class-wise standard deviation (s) and class element count (n), as well as the model-wise standard deviation (σ) and element count (N).

Within the Plotting tab, the sub-model of the CR modeler, an additional user interface defines:

- Selection of the NIM mission for prediction, $M = \{all\ NIM\ missions\}$
- As a model feature, a confidence level (CL2) option is provided for user identification of mission-wise uncertainty inclusion within the predictor. Options include CL2={75%, 80%, 85%, 90%, 95%}

The CR modeler includes five tabs. Each tab provides an independent function within the modeler:

- CR Model – provides the user interface inputs, performs the classified regression solutioning, and data of model performance for margin iteration and model optimization, and tabular data output of the CR model.
- Plotting – provides visualization of the class-wise CR model solutions as plots for each class based on the class domain limits, visualization of the regression and residual margin

boundaries with respect to data elements, and class assignment. An additional feature is added to allow for plot and visualization of a Funding System Use Case, where a NIM element is selected as the predictor model, and a plot of the element-wise predictor, element-adjusted based on μ_{class} and uncertainty as a function of σ_{class} and confidence level.

- Database – contains the attribute data of the NIM database. The Database tab includes the initial "sortby" function for CR to structure the data for classification, as imported to the CR Model tab. The attribute option pull-down menu is also configured in the tab. Data attributes contained within gray fields were derived from the NIM data and not expressly stated within the NIM.
- Classifier – contains the algorithm for the exhaustive and rank sort classifiers. The rank sort evolved from the exhaustive classifier. The F/FC combination is determined for each element. The values are ranked from maximum to minimum, sorted, and assigned class assignments based on their rank position within the list and the CR model class distribution. Comparison is evaluated using the rank-sorted classifier and the CR model's class assignment. A confusion matrix is assessed for each F/FC, considering a forward matrix and an inverted matrix. The forward/inverted matrix accounts for inverse F/FC calculations. For the standard NIM dataset, the classifier evaluates 3970 F/FC, 3970 $(F/FC)^{-1}$, and based on the best performing F/FC classifier, an additional 45 F product and 45 F quotient terms are applied to the best-performing classifier. This allows for assessment of a third order FC that contains three features in combination. A bypass feature is incorporated to evaluate the individual F/FC for the inclusion of a third feature.

- Neural Network – contains the preliminary outline for the System Predictor model and data structure of the CR predictors. This is included in the Future Work section. The predictor matrix contains predictors involving SCdry as the predictor or response, and reports the performance of each model at the model level for comparison.

Uncertainty

Uncertainty is the unexpected variation in the value of a measurement. For any random variable X, the relative likelihood that X takes a specific value is defined by the probability density function (PDF) or p(x). Integrating over the PDF, the probability that the value is less than or equal to X is given by the cumulative distribution function (CDF), denoted F(x). For a selected confidence level (CL), we determine the value of X such that

$$F(x) \leq CL \ \leq 1.00 \tag{24}$$

The challenge in determining uncertainty lies in identifying the probability density function. Predictive analytics assumes a Gaussian distribution for the predictions, which is incorrect in many applications. When a predictor is normally distributed and linear in X, the prediction is also normally distributed. For summation algorithms where each term is linear in X, the sum of the predictions is also normally distributed. However, when the predictor is not linear in X, such as a logarithmic or exponential function, it may be logarithmically or exponentially distributed, but not normally distributed. For these predictors, the CDF must be independently determined for the predictor.

Classified regression differs from traditional predictors. The prediction is evaluated in the predictor, which, for a simple linear model, is normally distributed. While the predictor may or may not be normally distributed and may or may not have sufficient data to substantiate normality, the residuals for each model class are normally distributed. Therefore, it is valid to evaluate the model's predictor output using statistical measures based on the Gaussian distribution.

The uncertainty formula is defined as:

$$U = \sqrt{\frac{\sum(x_i - \mu)^2}{n(n-1)}}$$

where x_i is the value of the i^{th} dataset element, μ is the mean of the dataset, and n is the number of elements in the dataset. Incorporating uncertainty within the CR predictor relies on confidence interval formulas. For large data sets, the Central Limit Theorem (CLT) is the convention. For large samples $\bar{X}$, the CLT states that x_i is normally distributed with a mean μ and a standard deviation $\sigma/\sqrt{n}$. For populations with a small sample, $n \leq 30$, the CLT may not be applicable. The basis for this standard is the inability to determine whether the small sample is normally distributed, as there is insufficient data to confirm it. If the data is normally distributed, then the CLT should apply regardless of the sample size.

When the population standard deviation is unknown and the normal approximation is no longer valid, the sample standard deviation, s, is substituted for the population standard deviation σ. This distribution is evaluated using Student's t-distribution with n-1 degrees of freedom. The t-distribution is similar to the normal distribution. It is centered about zero, exhibits the qualitative bell-shaped curve, and exhibits heavier tails than the standard normal distribution. A comparison is shown in *Figure 32*.

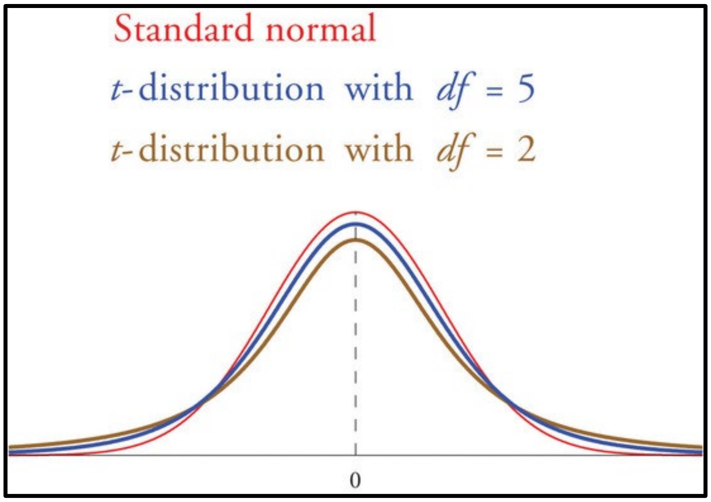


Figure 32: Comparison of normal and t-distributions

As the sample size increases, the Student's t-distribution converges to the standard normal distribution. It is an accepted practice in statistics that once the sample size is $n \geq 30$, the normal distribution applies. Based on this conclusion, a small sample is considered when $n < 30$, although this is not an established standard in the literature.

The Normal Distribution

The normal distribution is the most commonly applied distribution for independent, randomly occurring variables. Complexity arises when the data distribution is not normal or when the sample size is insufficient to test for normality. Common reasons why data is not normally distributed are presented *Table 14*.

Table 14: Normal Distribution Exclusion Criteria

Issue	How CR addresses the normal distribution exclusion criteria
Existence of outliers	CR classifies all data as outliers. Classes are defined for “like” elements, similar to clustering, and “outliers” are assigned to a different class.
Data (response variables) affected by more than one process	CR is based on a 1-to-1 regression relationship that models a single process. The corrector term is based on the original predictor variable, eliminating the need to introduce modeling complexity; it does not add degrees of freedom.
Insufficient data quantity	This concern is not related to the quantity of data, but rather the ability to demonstrate normal distribution. CR is computed from the prediction residual, and the residual of a linear regression solution is normally distributed by definition.
Poor measurement resolution	The CR measurement is a posteriori data, legacy data. For the NIM dataset, the measurement resolution is precise according to the standards for recording and reporting established within NASA.
Alternate distributions offer a better description of the data	CR is based on linear regression (the simplest form of modeling). Predictor performance is measured using the prediction residual, which is normally distributed by definition.
Data approach zero (boundary or constraint)	CR establishes a domain constraint for each class predictor. Caution should be exercised when the predictor approaches or exceeds the model's domain, as demonstrated by the validation example.
Data is overly discrete	CR is a component-based, continuous predictor methodology. Any prediction can be further partitioned, provided the input dataset supports partitioning. This was demonstrated within the “total cost” predictor. A module was developed to predict total cost based on a specific predictor. A series of modules was developed to predict the components of cost (development, launch, and operations). These modules were combined to create an alternative predictor of total cost. The total cost model was derived from the average of the two modules, providing an optimized predictor performance.

NASA Cost Risk and Uncertainty

E is the margin of error of the estimate about the population mean error. The coefficient $z\alpha/_2$ constructs the confidence interval for large samples and $t\alpha/_2$ constructs the confidence interval for small samples.

If σ is known:

$$E = \pm z\alpha_{/2}(\frac{\sigma}{\sqrt{n}})$$

If σ is unknown:

$$E = \pm t\alpha_{/2}(\frac{s}{\sqrt{n}})$$

For $z\alpha_{/2}$ the cumulative normal probability applies, as shown in Table 2, for a desired confidence interval. NASA has extensively studied cost-risk analysis (NASA, 2008). The interpretation of cost modeling and technical risk is summarized in Figure *33*, where a distribution about the predictor corresponds to a distribution in the response.

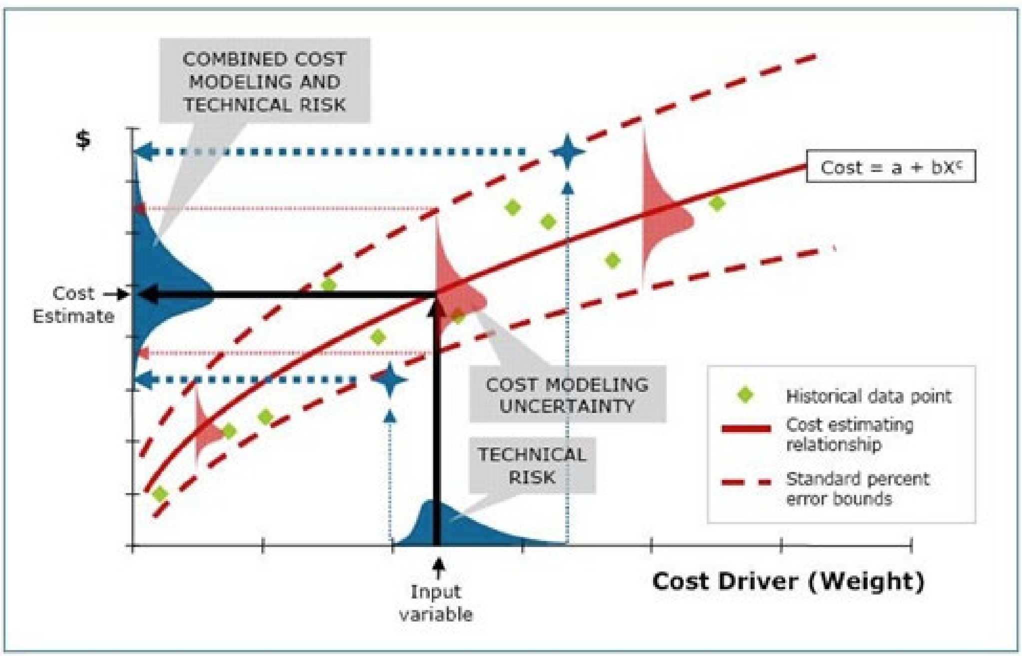


Figure 33: Cost Modeling and Technical Risk Input (NASA, 2008)

The analysis considers a variety of data profiles *Figure 34*, applies statistical summation and defines a probability distribution for the projected cost element.

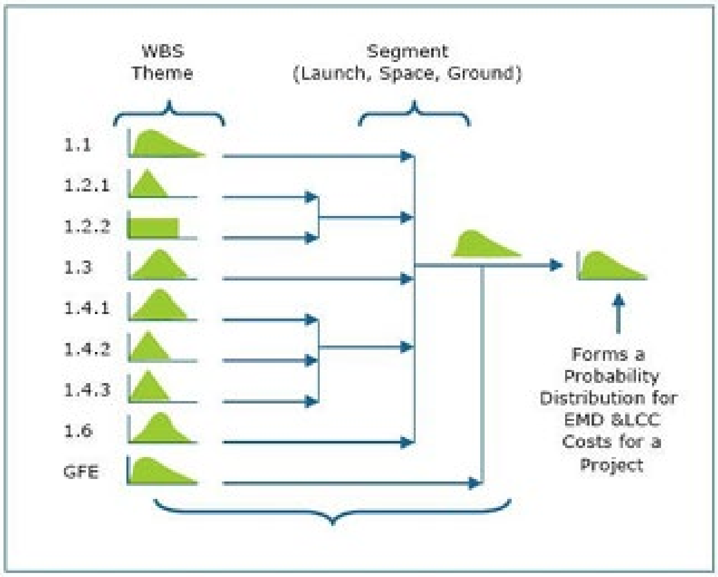


Figure 34: Statistical Summation Process Summary (NASA, 2008)

For each element, relative risk weighting (RRW) is applied *Figure 35*.

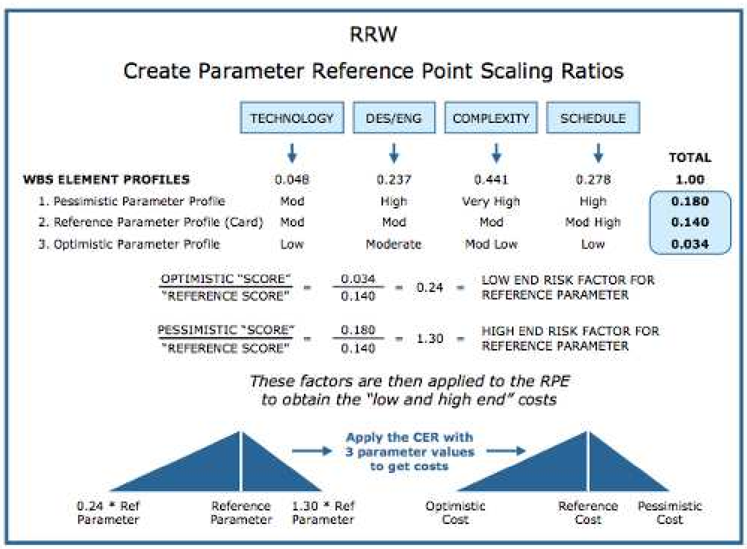


Figure 35: Discrete Risk Analysis Using Relative Risk Weighting (NASA, 2008)

NASA cost and schedule risk assessment is mitigated through the application of the Joint Cost and Schedule Confidence Level (JCL) (NASA HQ, 2022). Planning requires JCL of 50-70% for most initiatives, with 70% applied for most NIM class missions. This practice accepts a 30% chance of cost overrun. The implementation of this standard practice evaluates the cost-risk distribution and the resultant cumulative density S-curve *Figure 36*, at a 70% confidence level. This value corresponds to the point along the probability density curve at the upper inflection point, where the curve transitions from linear to asymptotic behavior at 1.00.

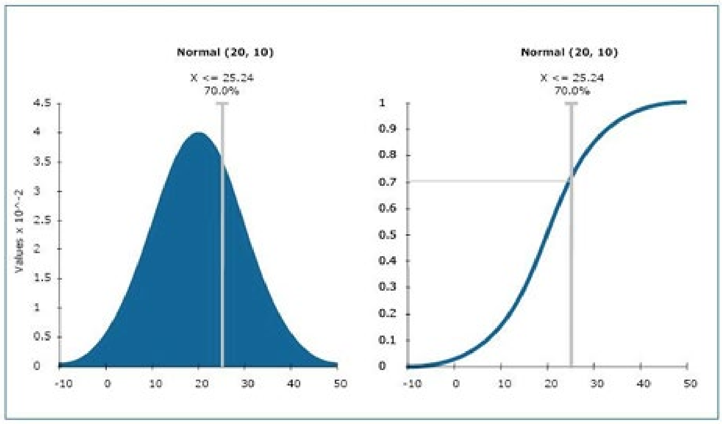


Figure 36: Probability Density Function (PDF)

A 70% confidence level for a normal distribution corresponds to:

$$z\alpha/_2 = 1.033$$

$$and\ E = \pm\left(\frac{1.033}{\sqrt{n}}\right)\sigma\ or$$

$$E = \pm\, 0.243\ \sigma\ where\ n = 30$$

For a sample where $n < 30$, $\alpha = 0.30, {}^{\alpha}/_{2} = 0.15, P = 0.85$ from *Table 16*, or

$$t\alpha/_2 = 1.474\ \ and\ E = \pm\, 1.042\ \sigma\ \ for\ s = 2$$

$$t\alpha/_2 = 1.308\ \ and\ E = \pm\, 0.755\ \sigma\ \ for\ s = 3$$

$$t\alpha/_2 = 1.198\ \ and\ E = \pm\, 0.536\ \sigma\ \ for\ s = 5$$

$$t\alpha/_2 = 1.126\ and\ E = \pm\, 0.356\ \sigma\ for\ s = 10$$

$$t\alpha/_2 = 1.104\ and\ E = \pm\, 0.285\ \sigma\ for\ s = 15$$

$$t\alpha/_2 = 1.082\ and\ E = \pm\, 0.198\ \sigma\ for\ s = 30$$

Linear interpolation for ${}^{\alpha}/_{2} = 0.15$ as determined from the listed values of 0.100 and 0.200 in *Table 15*. For values where $E = \pm 1.00\,\sigma$, (as used to define model performance), this correlates to

$${}^{\alpha}/_{2} = .040\ or\ 92\%\ confidence\ for\ s = 5$$

$${}^{\alpha}/_{2} = .005\ or\ 99\%\ confidence\ for\ s = 10$$

$${}^{\alpha}/_{2} = .001\ or\ 99.8\%\ confidence\ for\ s = 15$$

$${}^{\alpha}/_{2} = .0005\ or\ 99.9\%\ confidence\ for\ s = 30$$

Sample size does influence the uncertainty error of the predictor. The relative magnitude of this is significant compared to the small sample threshold. However, the absolute magnitude of the uncertainty error of the prediction is significantly more influenced by the method of modeling and the coefficient of variance (CV) of the predictor:

$$CV = {}^{\sigma}/_{\mu}\ x\ 100 \tag{25}$$

Another significant factor in modeling performance is the use of the mean and standard deviation. With most predictive algorithms, the object is to “best fit” the data. Linear regression employs the method of least squares, which, by definition, minimizes the residual error by distributing the residuals as evenly as possible about the regression curve. Sign cancellation significantly influences the reported mean. The absolute mean error and its standard deviation, |µ| and |σ|, are better indicators of the model's prediction performance in the residuals.

Bessel Sample Correction

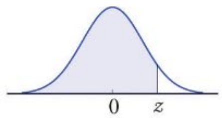
The Bessel correction is commonly applied to sample statistics. This occurs because sample statistics tend to underestimate the population parameters. In such cases, the sample standard deviation is

$$s = \sqrt{\frac{\sum(x_i - \bar{x})^2}{(n-1)}} \tag{26}$$

Where n is the sample size. The correction accounts for the sample variation using the sample mean $\bar{x}$ opposed to the population mean μ.

For small sample sizes, the Bessel correction is considered quite severe until the sample size approaches the normal transition n $\simeq$ 30. Warne (Warne, 2021) recommends using Bessel correction only if you have a sufficiently large sample and are primarily interested in the population statistics. If the focus is on the sample statistics rather than the population statistics, his suggestion is to omit the correction. For small sample predictors, where the data population is small and limited to the sample, this is appropriate.

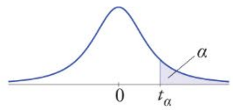
Table 15: Cumulative Normal Probability *(Shafer & Zhang, 2019)*



Cumulative Probability $P(Z \le z)$

z	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09
0.0	0.5000	0.5040	0.5080	0.5120	0.5160	0.5199	0.5239	0.5279	0.5319	0.5359
0.1	0.5398	0.5438	0.5478	0.5517	0.5557	0.5596	0.5636	0.5675	0.5714	0.5753
0.2	0.5793	0.5832	0.5871	0.5910	0.5948	0.5987	0.6026	0.6064	0.6103	0.6141
0.3	0.6179	0.6217	0.6255	0.6293	0.6331	0.6368	0.6406	0.6443	0.6480	0.6517
0.4	0.6554	0.6591	0.6628	0.6664	0.6700	0.6736	0.6772	0.6808	0.6844	0.6879
0.5	0.6915	0.6950	0.6985	0.7019	0.7054	0.7088	0.7123	0.7157	0.7190	0.7224
0.6	0.7257	0.7291	0.7324	0.7357	0.7389	0.7422	0.7454	0.7486	0.7517	0.7549
0.7	0.7580	0.7611	0.7642	0.7673	0.7704	0.7734	0.7764	0.7794	0.7823	0.7852
0.8	0.7881	0.7910	0.7939	0.7967	0.7995	0.8023	0.8051	0.8078	0.8106	0.8133
0.9	0.8159	0.8186	0.8212	0.8238	0.8264	0.8289	0.8315	0.8304	0.8365	0.8389
1.0	0.8413	0.8438	0.8461	0.8485	0.8508	0.8531	0.8554	0.8577	0.8599	0.8621
1.1	0.8643	0.8665	0.8686	0.8708	0.8729	0.8749	0.8770	0.8790	0.8810	0.8830
1.2	0.8849	0.8869	0.8888	0.8907	0.8925	0.8944	0.8962	0.8980	0.8997	0.9015
1.3	0.9032	0.9049	0.9066	0.9082	0.9099	0.9115	0.9131	0.9147	0.9162	0.9177
1.4	0.9192	0.9207	0.9222	0.9236	0.9251	0.9265	0.9279	0.9292	0.9306	0.9319
1.5	0.9332	0.9345	0.9357	0.9370	0.9382	0.9394	0.9406	0.9418	0.9429	0.9441
1.6	0.9452	0.9463	0.9474	0.9484	0.9495	0.9505	0.9515	0.9525	0.9535	0.9545
1.7	0.9554	0.9564	0.9573	0.9582	0.9591	0.9599	0.9608	0.9616	0.9625	0.9633
1.8	0.9641	0.9649	0.9656	0.9664	0.9671	0.9678	0.9686	0.9693	0.9699	0.9706
1.9	0.9713	0.9719	0.9726	0.9732	0.9738	0.9744	0.9750	0.9756	0.9761	0.9767
2.0	0.9772	0.9778	0.9783	0.9788	0.9793	0.9798	0.9803	0.9808	0.9812	0.9817
2.1	0.9821	0.9826	0.9830	0.9834	0.9838	0.9842	0.9846	0.9850	0.9854	0.9857
2.2	0.9861	0.9864	0.9868	0.9871	0.9875	0.9878	0.9881	0.9884	0.9887	0.9890
2.3	0.9893	0.9896	0.9898	0.9901	0.9904	0.9906	0.9909	0.9911	0.9913	0.9916
2.4	0.9918	0.9920	0.9922	0.9925	0.9927	0.9929	0.9931	0.9932	0.9934	0.9936
2.5	0.9938	0.9940	0.9941	0.9943	0.9945	0.9946	0.9948	0.9949	0.9951	0.9952
2.6	0.9953	0.9955	0.9956	0.9957	0.9959	0.9960	0.9961	0.9962	0.9963	0.9964
2.7	0.9965	0.9966	0.9967	0.9968	0.9969	0.9970	0.9971	0.9972	0.9973	0.9974
2.8	0.9974	0.9975	0.9976	0.9977	0.9977	0.9978	0.9979	0.9979	0.9980	0.9981
2.9	0.9981	0.9982	0.9982	0.9983	0.9984	0.9984	0.9985	0.9985	0.9986	0.9986
3.0	0.9987	0.9987	0.9987	0.9988	0.9988	0.9989	0.9989	0.9989	0.9990	0.9990
3.1	0.9990	0.9991	0.9991	0.9991	0.9992	0.9992	0.9992	0.9992	0.9993	0.9993
3.2	0.9993	0.9993	0.9994	0.9994	0.9994	0.9994	0.9994	0.9995	0.9995	0.9995
3.3	0.9995	0.9995	0.9995	0.9996	0.9996	0.9996	0.9996	0.9996	0.9996	0.9997
3.4	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9998
3.5	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998
3.6	0.9998	0.9998	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999
3.7	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999
3.8	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999
3.9	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

Table 16: Critical Values of t *(Shafer & Zhang, 2019)*



Critical Values of *t*

df	$t_{0.200}$	$t_{0.100}$	$t_{0.050}$	$t_{0.025}$	$t_{0.010}$	$t_{0.005}$	$t_{0.0025}$	$t_{0.001}$	$t_{0.0005}$
1	1.376	3.078	6.314	12.706	31.821	63.657	127.321	318.309	636.619
2	1.061	1.886	2.920	4.303	6.965	9.925	14.089	22.327	31.599
3	0.978	1.638	2.353	3.182	4.541	5.841	7.453	10.215	12.924
4	0.941	1.533	2.132	2.776	3.747	4.604	5.598	7.173	8.610
5	0.920	1.476	2.015	2.571	3.365	4.032	4.773	5.893	6.869
6	0.906	1.440	1.943	2.447	3.143	3.707	4.317	5.208	5.959
7	0.896	1.415	1.895	2.365	2.998	3.499	4.029	4.785	5.408
8	0.889	1.397	1.860	2.306	2.896	3.355	3.833	4.501	5.041
9	0.883	1.383	1.833	2.262	2.821	3.250	3.690	4.297	4.781
10	0.879	1.372	1.812	2.228	2.764	3.169	3.581	4.144	4.587
11	0.876	1.363	1.796	2.201	2.718	3.106	3.497	4.025	4.437
12	0.873	1.356	1.782	2.179	2.681	3.055	3.428	3.930	4.318
13	0.870	1.350	1.771	2.160	2.650	3.012	3.372	3.852	4.221
14	0.868	1.345	1.761	2.145	2.624	2.977	3.326	3.787	4.140
15	0.866	1.341	1.753	2.131	2.602	2.947	3.286	3.733	4.073
16	0.865	1.337	1.746	2.120	2.583	2.921	3.252	3.686	4.015
17	0.863	1.333	1.740	2.110	2.576	2.898	3.222	3.646	3.965
18	0.862	1.330	1.734	2.101	2.552	2.878	3.197	3.610	3.922
19	0.861	1.328	1.729	2.093	2.539	2.861	3.174	3.579	3.883
20	0.860	1.325	1.725	2.086	2.528	2.845	3.153	3.552	3.850
21	0.859	1.323	1.721	2.080	2.518	2.831	3.135	3.527	3.819
22	0.858	1.321	1.717	2.074	2.508	2.819	3.119	3.505	3.792
23	0.858	1.319	1.714	2.069	2.500	2.807	3.104	3.485	3.768
24	0.857	1.318	1.711	2.064	2.492	2.797	3.091	3.467	3.745
25	0.856	1.316	1.708	2.060	2.485	2.787	3.078	3.450	3.725
26	0.856	1.315	1.706	2.056	2.479	2.779	3.067	3.435	3.707
27	0.855	1.314	1.703	2.052	2.473	2.771	3.057	3.421	3.690
28	0.855	1.313	1.701	2.048	2.467	2.763	3.047	3.408	3.674
29	0.854	1.311	1.699	2.045	2.462	2.756	3.038	3.396	3.659
30	0.854	1.310	1.697	2.042	2.457	2.750	3.030	3.385	3.646
31	0.853	1.309	1.696	2.040	2.453	2.744	3.022	3.375	3.633
32	0.853	1.309	1.694	2.037	2.449	2.738	3.015	3.365	3.622
33	0.853	1.308	1.692	2.035	2.445	2.733	3.008	3.356	3.611
34	0.852	1.307	1.691	2.032	2.441	2.728	3.002	3.348	3.601
35	0.852	1.306	1.690	2.030	2.438	2.724	2.996	3.340	3.591
36	0.852	1.306	1.688	2.028	2.434	2.719	2.990	3.333	3.582
37	0.851	1.305	1.687	2.026	2.431	2.715	2.985	3.326	3.574
38	0.851	1.304	1.686	2.024	2.429	2.712	2.980	3.319	3.566
39	0.851	1.304	1.685	2.023	2.426	2.708	2.976	3.313	3.558
40	0.851	1.303	1.684	2.021	2.423	2.704	2.971	3.307	3.551
41	0.851	1.303	1.683	2.020	2.421	2.701	2.967	3.301	3.544
42	0.851	1.302	1.682	2.018	2.418	2.698	2.963	3.296	3.538
43	0.851	1.302	1.681	2.017	2.416	2.695	2.959	3.291	3.532
44	8.850	1.301	1.680	2.015	2.414	2.692	2.956	3.286	3.526
45	0.850	1.301	1.679	2.014	2.412	2.690	2.952	3.281	3.520
46	0.850	1.300	1.679	2.013	2.410	2.687	2.949	3.277	3.515
47	0.849	1.300	1.678	2.012	2.408	2.685	2.946	3.273	3.510
48	0.849	1.299	1.677	2.011	2.407	2.682	2.943	3.269	3.505
49	0.849	1.299	1.677	2.010	2.405	2.680	2.940	3.265	3.500
50	0.849	1.299	1.676	2.009	2.403	2.678	2.937	3.261	3.496

Classified Regression Uncertainty

The CR modeler is assembled from the *Figure 27 – 29* model segments in *Figure 37*, allowing for assessment of linear regression, classified regression, and classified regression with residual correction segments. The absolute mean error provides the average error of the model, also known as the model error. Model uncertainty is represented as the standard deviation of the absolute mean error. The sum of the model performance is (8). This is an extreme interpretation of uncertainty. For normally distributed data, the ±1 σ value corresponds to a confidence level of 84%, which exceeds the standard practice in NASA cost estimation. As the model predictor evolves from one segment to the next, the model's performance significantly improves *Table 17*.

Table 17: CR Model Performance, Accuracy of SCdry = F(TDCT)

Model	**Abs Error $\|\mu\|$**	**Uncertainty $+1\|\sigma\|$**	**Performance $\|\mu\| + 1\|\sigma\|$**	**Performance @ 70% confidence**
Linear Regression	36.2%	38.7%	74.9%	40.8%
Classified Regression	13.7%	11.9%	25.5%	13.2%
Classified Regression with Residual Correction	3.8%	3.5%	7.3%	5.3%

For thoroughness, an option is included in the CR predictor that calculates the model and class-wise performance based on the class sample size and the selection of confidence levels: 70% *Figure 38*, 80% *Figure 39*, 90% *Figure 40*, and 96% *Figure 41*. All values reported are determined from the $SCdry = F(TDCT)$ model with a Regression Threshold of 0.14, a Residual Threshold of 0.07, and the zero-class limit set to null. The performance is:

$$Performance = |\mu| + z\alpha/_2 \cdot \frac{s}{\sqrt{n}} \tag{27}$$

The absolute mean error is the class $|\mu|$, $z\alpha/_2$ is the z value of the confidence interval based on the class size n, and s is the class |σ|. Since the class contains all elements of the class, the sample and population are synonymous. For model performance, |μ| and |σ| are used with n = 35, and all NIM elements constitute the population.

The high-performance accuracy of the $SCdry = F(TDCT)$ demonstrates the correlation between the predictor and the NIM data. The R-squared of the raw data is 0.858. The R^2 of the CR with residual correction is 0.864. Similarly, the Pearson correlation coefficient (PCC) for the raw data is 0.926, and for the CR with residual correction, it is 0.929. This indicates that the CR model accurately captures the data dispersion of the correlation between the NIM attributes.

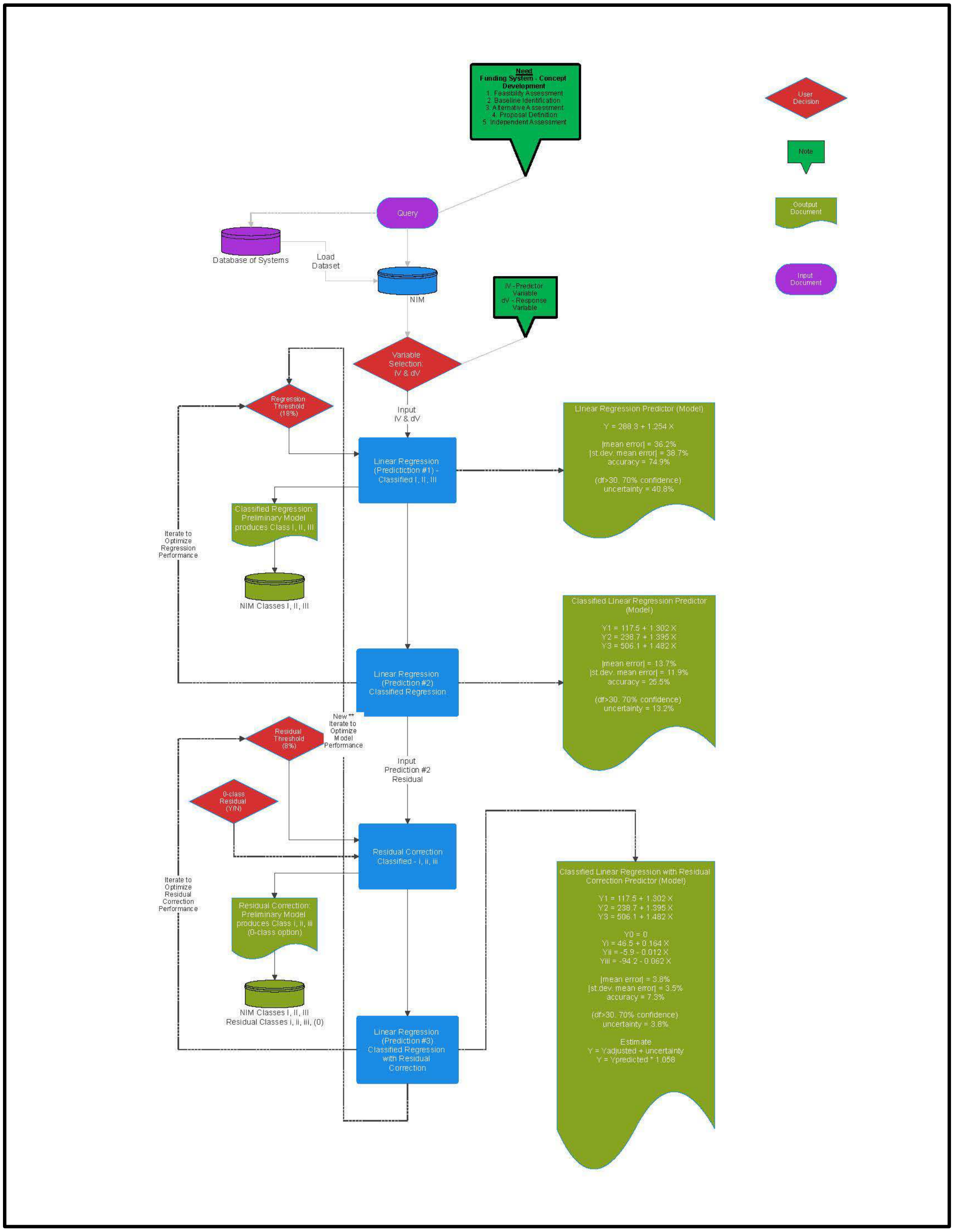


Figure 37: Classified Regression Model Structure

The output summaries indicate the interaction between confidence level and model performance, including uncertainty for 70%, 80%, 90%, and 96% confidence in the prediction (Figures 7-10).

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
SCdry = f(TDCT)				

Class Regression Threshold	16%
Class Residual Threshold	7%

	Regression n	Residual n	Min %e	Max %e
Class 1	12	12	-7.0%	16.4%
Class 2	13	14	-8.3%	12.7%
Class 3	10	9	-9.8%	9.1%

Model	LR	CR	CR w/corr
mean	17.5%	-1.6%	0.5%
|mean|	36.2%	13.5%	3.8%
st.dev	50.4%	18.1%	5.3%
st.dev | |	38.7%	11.9%	3.6%
co.var	34.7%	-8.8%	9.81
co.var | |	93.5%	113.8%	1.39
|mean| + st.dev | |	74.9%	25.4%	7.4%

Modeling Error & Uncertainty				
Model Error				
Accuracy	Raw Data	LR	CR	CR w/corr
mean + stdev		67.8%	16.5%	5.8%
|mean| + |stdev|		74.9%	25.4%	7.4%
Pearson	0.926	1.000	0.934	0.929
R-squared	0.858	1.000	0.873	0.864
Model Performance with Uncertainty(n) @ 70%				
@ 70% Confidence (Z)		0.5344		
mean + stdev		44.4%	8.1%	3.4%
|mean| + |stdev|		56.9%	19.9%	5.7%
delta to |stdev|		-32.7%	-47.1%	-51.5%
Class Performance Uncertainty(n)				
Confidence 70.0%percent				
	n	LR	CR	CR w/corr
Model	35	43.0%	15.6%	4.4%
Class 1	12	86.2%	27.2%	4.9%
Class 2	13	11.3%	11.6%	5.0%
Class 3	10	30.5%	10.5%	4.9%
Mission	Mars Recon Orbiter for Plots			

Figure 38: SCdry Model Performance at 70% Confidence

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
SCdry = f(TDCT)				

Class Regression Threshold	16%
Class Residual Threshold	7%

	Regression n	Residual n	Min %e	Max %e
Class 1	12	12	-7.0%	16.4%
Class 2	13	14	-8.3%	12.7%
Class 3	10	9	-9.8%	9.1%

Model	LR	CR	CR w/corr
mean	17.5%	-1.6%	0.5%
|mean|	36.2%	13.5%	3.8%
st.dev	50.4%	18.1%	5.3%
st.dev | |	38.7%	11.9%	3.6%
co.var	34.7%	-8.8%	9.81
co.var | |	93.5%	113.8%	1.39
|mean| + st.dev | |	74.9%	25.4%	7.4%

Modeling Error & Uncertainty				
Model Error				
Accuracy	Raw Data	LR	CR	CR w/corr
mean + stdev		67.8%	16.5%	5.8%
|mean| + |stdev|		74.9%	25.4%	7.4%
Pearson	0.926	1.000	0.934	0.929
R-squared	0.858	1.000	0.873	0.864
Model Performance with Uncertainty(n) @ 70%				
@ 70% Confidence (Z)		0.5344		
mean + stdev		44.4%	8.1%	3.4%
|mean| + |stdev|		56.9%	19.9%	5.7%
delta to |stdev|		-32.7%	-47.1%	-51.5%
Class Performance Uncertainty(n)				
Confidence 80.0%percent				
	n	LR	CR	CR w/corr
Model	35	44.6%	16.1%	4.6%
Class 1	12	89.5%	28.2%	5.2%
Class 2	13	11.6%	12.2%	5.2%
Class 3	10	31.1%	11.1%	5.1%
Mission	Mars Recon Orbiter for Plots			

Figure 39: SCdry Model Performance at 80% Confidence

Model Summary (35 missions)

Classified Linear Reg with Classified Residual Reg

SCdry = f(TDCT)

Class Regression Threshold	14%
Class Residual Threshold	7%

	Regress	Residual	Min	Max
Class 1	16	10	-29.7%	9.0%
Class 2	9	15	-4.3%	9.6%
Class 3	10	10	-10.2%	4.6%

Model	LR	CR	CR w/corr
mean	17.5%	-3.4%	0.5%
|mean|	36.2%	15.3%	4.3%
st.dev	50.4%	22.0%	6.9%
st.dev ||	38.7%	15.9%	5.4%
co.var	34.7%	-15.5%	13.40
co.var||	93.5%	96.4%	1.60
|mean| + st.dev ||	74.9%	31.2%	9.7%

Modeling Error & Uncertainty

Model Error				
Accuracy	**Raw Data**	**LR**	**CR**	**CR w/corr**
mean + stdev		67.8%	18.6%	7.4%
|mean| + |stdev|		74.9%	31.2%	9.7%
Pearson	0.926	1.000	0.935	0.933
R-squared	0.858	1.000	0.875	0.870
Model Uncertainty				
@ 70% Confidence (Z)		0.5344		
mean + stdev		44.4%	8.3%	4.2%
|mean| + |stdev|		56.9%	23.8%	7.2%
shift to |mean|		20.7%	8.5%	2.9%

Class Uncertainty				
Confidence 90.0%percent				
	n	**LR**	**CR**	**CR w/corr**
Model	35	47.0%	19.7%	5.8%
Class 1	16	77.5%	32.5%	8.3%
Class 2	9	9.7%	10.1%	5.4%
Class 3	10	32.1%	12.0%	5.1%
Mission	Dart			

Figure 40: SCdry Model Performance at 90% Confidence

Model Summary (35 missions)

Classified Linear Reg with Classified Residual Reg

SCdry = f(TDCT)

Class Regression Threshold	16%
Class Residual Threshold	7%

	Regression n	Residual n	Min %e	Max %e
Class 1	12	12	-7.0%	16.4%
Class 2	13	14	-8.3%	12.7%
Class 3	10	9	-9.8%	9.1%

Model	LR	CR	CR w/corr
mean	17.5%	-1.6%	0.5%
|mean|	36.2%	13.5%	3.8%
st.dev	50.4%	18.1%	5.3%
st.dev ||	38.7%	11.9%	3.6%
co.var	34.7%	-8.8%	9.81
co.var||	93.5%	113.8%	1.39
|mean| + st.dev ||	74.9%	25.4%	7.4%

Modeling Error & Uncertainty

Model Error				
Accuracy	**Raw Data**	**LR**	**CR**	**CR w/corr**
mean + stdev		67.8%	16.5%	5.8%
|mean| + |stdev|		74.9%	25.4%	7.4%
Pearson	0.926	1.000	0.934	0.929
R-squared	0.858	1.000	0.873	0.864
Model Performance with Uncertainty(n) @ 70%				
@ 70% Confidence (Z)		0.5344		
mean + stdev		44.4%	8.1%	3.4%
|mean| + |stdev|		56.9%	19.9%	5.7%
delta to |stdev|		-32.7%	-47.1%	-51.5%

Class Performance Uncertainty(n)				
Confidence 96.0%percent				
	n	**LR**	**CR**	**CR w/corr**
Model	35	49.6%	17.6%	5.1%
Class 1	12	100.1%	31.2%	6.2%
Class 2	13	12.7%	14.1%	6.0%
Class 3	10	33.2%	13.0%	5.9%
Mission	Mars Recon Orbiter for Plots			

Figure 41: SCdry Model Performance at 96% Confidence

Chapter 4: Data Analysis and Results

CR Modeling

Performance Accuracy

Two evaluations measure the performance accuracy of the CR predictor:

- Inter-model performance is measured as the accuracy of the predictor versus known data for all dataset elements. This measure is reported as the mean error $|\mu|$ and the uncertainty $|\sigma|$, where the sum of the model's mean error and uncertainty is the performance accuracy. The measure is reported as model, class-wise, and element-wise performance. The standard uncertainty is reported at the 85% confidence level for normally distributed data, 1σ. Alternatively, this feature allows the user to select an uncertainty at a different confidence level. This standard for reporting performance accuracy applies to all model results.
- Comparative model performance is an analysis of the CR total development cost in NSII inflated dollars (TDCT) as a function of the SCdry mass attribute. The NSII inflation index is used for consistency across alternative models and to align with NASA's practice and industry standards. Alternative models use spacecraft mass to predict development costs.

The comparative model performance is evaluated against five industry-published predictors for interplanetary satellite missions. The TDCT prediction is for the first unit development cost, which is consistent for NIM programs. Each predictor was used to predict the set of NIM elements and compared to the Development Cost CR predictor. The results are presented *Table 18*.

Table 18: Performance Comparison of CR and Five Industry Cost Models

	CR model	2nd Order Pol Reg (24)	Foreman (25) (26)	USCM (27) (28)	NASA SSE CEM (29)	Wertz (30) (31)
Abs error, $\lvert\mu\rvert$	8.3 %	42.7 %	113.1 %	68.7 %	46.7 %	25.5 %
St.dev of abs error $\lvert\sigma\rvert$	7.6 %	37.8 %	84.2 %	20.4 %	32.7 %	16.4 %
Performance $\lvert\mu\rvert + \lvert\sigma\rvert$	15.9 %	80.5 %	197.4 %	89.1 %	79.3 %	41.9 %

The comparative models and associated equations include:

1. Polynomial Regression – using a 2nd-order polynomial regression predictor:

   a. $Y = -6.537 + 0.413\, X + .000089\, X^2$ (28)

2. The Foreman equation (Foreman et al., 2016), previously referenced as (4):

   a. $Y = C_{spacecraft} + C_{payload}$ (29)

   b. $\begin{cases} C_{spacecraft} = 781 + 26.1\, {m_{dry}}^{1.261} \\ C_{payload} = 0.4\, C_{spacecraft} \end{cases}$ (30)

3. USCM Cost Model (Crisp et al., 2019):

   a. $Y_1\ or\ Y_2\ as\ appropriate$

   b. $Y_1 = X_1^{0.661}\ \ for\ X_1 \leq 400\ kg$ (31)

   c. $Y_3 = \frac{43}{1000} X_1\ \ for\ X_1 \geq 400\ kg$ (32)

4. NASA Space System Engineering Cost Estimating Module (NASA, 2008):

   a. $Y = 11.279\, X^{0.5}$ (33)

5. Wertz (Wertz et al., 2018) Table 11-9 equation:

a. $Y = (1.332) * 283.5\, X^{0.716}\, /100$ (34)

b. $NSII\ conversion\ from\ 2010\$\ to\ 2023\$ = 1.332$ (35)

Wertz (Wertz et al., 2018) provides a comprehensive cost-estimating algorithm for a space mission cost estimator that accounts for a standard error of estimate of 41%. This equation incorporates a term for planetary versus non-planetary missions. The Wertz algorithm equation is:

$$Y = 2.829\ x\ (Dry\ Mass^{0.457})x\ (Power^{0.157})x\ \left(2.718^{(0.171\ x\ Data\ \%)}\right)\ x\ \left(2.718^{(0.00209\ x\ Life)}\right)\ x\ \left(2.718^{(1.52\ x\ New)}\right)\ x\ (2.718^{(0.258\ x\ Planetary)}\ x\ \frac{1}{2.718^{(0.0145\ x\ (Year-1960))}}\ x\ \left(2.718^{(0.467\ x\ InstComp\ \%)}\right)x\ \frac{1}{2.718^{0.237\ x\ Team)}} \quad (36)$$

The variables in (32) are:

- Dry Mass – dry mass of spacecraft bus and instruments in kg
- Power – Low earth orbit (LEO) equivalent beginning of life power (BOLP) in watts
- Data% - Data Rate Percentile (fraction) relative to the state-of-the-art at authority to proceed, 0.5 for median data rates, < 0.5 for lower data rates, and > 0.5 for higher data rates based on user judgement
- Life – Advertised design life (in months) – excludes extended operations
- New – Percent new as a fraction of the overall design: 0.2-0.3 = Simple Modifications, 0.3-0.7 = Extensive Modifications, 0.7-1.0 = New, > 1.0 for New Technology based on user judgement
- Planetary – 0 for Earth orbital, 1 for planetary
- Year – Available-to-promise (ATP) date in 4-digit calendar year minus 1960

- InstrComp% - Instrument complexity percentile (fraction) relative to nominal instrument complexity, < 0.5 for lower complexity, > 0.5 for higher complexity based on user judgement
- Team – Team Experience: 1 = unfamiliar, 2 = Mixed, 3 = Normal, 4 = Extensive based on user judgement

The estimate provides a Cost estimate in 2010 millions of dollars, including Phases B through C/D, excluding launch costs, and is consistent with the CR predictor. Further adjustments are included based on Wertz recommendations:

- Add 2% for Phase A (rule of thumb)
- Add 9% for Ground Station (rule of thumb for retrofitting existing facilities)
- Add 5% per year for Mission Operations and Data Analysis (rule of thumb)
- Includes contractor fees (10%) and NASA full cost support (17%)
- Estimate for 1st flight unit
- Includes all flight and ground software
- Excludes launch services (add $150 to $250M for Delta IV or Atlas V variants)

The Wertz algorithm relies on nine independent variables consuming eighteen *dof* and demonstrates the complexity of higher-order algorithms. In addition, the Wertz algorithm relies on four user-defined variables.

A survey of the NIM missions using the Wertz algorithm demonstrates the difficulty in system prediction. The attributes of the NIM utilized the Wertz algorithm to predict the development cost of an interplanetary mission. The user-defined attributes were set at nominal values to evaluate the algorithm's prediction performance. A nominal model exhibited an absolute mean error $|\mu| =$

77.8% with a standard deviation $|\sigma| = 144.7\%$ compared to the NIM TDCT. Similar performance was exhibited when comparing the Wertz algorithm predictions to the CR model. The user-defined attributes were evaluated for minimum and maximum ranges of values, along with nominal values, and an optimized model was constructed with {Data% = 1.0}, {New = 1.2}, {%InstComp = 1.0}, and {Team = 1.0}. The overall predictor improved to an absolute mean error $|\mu| = 73.4\%$ with a standard deviation $|\sigma| = 14.8\%$ or $performance = 88.2\%$. The CR TDCT predictor's performance demonstrated significant accuracy improvement compared to the Wertz algorithm, with an absolute mean error of $|\mu| = 9.3\%$ and a standard deviation of $|\sigma| = 8.6\%$, or a $performance = 17.9\%$.

CR Sensitivity to Margin Boundary

The SCdry predictor model optimization is presented as a demonstration of model sensitivity to margin boundary selection. The initial user input requires the identification of the independent variable $TDCT$ and the dependent variable $SCdry$ for predictor model development. The measure of performance is the absolute mean error |μ| and standard deviation of the absolute mean error |σ| in the predictor at a confidence level of 84% normal distribution residual. The model performance, as the regression boundary and residual boundary are adjusted, is shown *Table 19*.

Table 19: Model Performance Optimization

Iteration	Regression Margin	Residual Margin	Reg/Res Class 1 Elements	Reg/Res Class 2 Elements	Reg/Res Class 3 Elements	|μ|	|σ|	% error
1	20%	13%	10/13	13/11	12/11	12.2	12.9	25.1
2	45%	13%	6/10	22/18	7/7	14.3	24.0	38.3
3	40%	13%	7/11	21/13	7/11	16.3	16.7	33.0
4	70%	13%	4/15	25/14	6/6	13.0	10.8	23.8
5	65%	13%	5/12	24/14	6/9	11.3	11.7	23.0
6	65%	25%	5/8	24/23	6/4	15.0	14.5	29.5
7	30%	16%	7/8	16/21	12/6	9.3	7.6	16.9
8	30%	15%	7/9	16/20	12/6	8.9	6.5	15.4
9	30%	13%	7/9	16/16	12/10	9.0	6.6	15.6

Table 18 demonstrates how the data classification is reconfigured as the model is optimized for accuracy performance. The sensitivity of the model's performance is significant compared to the selection of regression and residual margin boundaries.

Scope Model

Model #1 demonstrates the effectiveness of CR as a predictor of outcomes between the attributes of the ($Scope \iff Cost$) for the NIM small dataset. The model #1 attributes are total development cost in Then-Year dollars (TDCT) as the predictor and spacecraft dry mass (SCdry) as the response. This relationship is the inverse of the traditional cost model construction, demonstrating the modeling flexibility of CR across a range of attributes as both a predictor and a response, without bias. The Model #1 development is presented in Figures 2-6. The predictor's performance is summarized *Table 20*. The element-wise predictions with correction are presented *Table 21*.

Table 20: Model #1 Predictor Performance

Model Summary (35 missions)

Classified Linear Reg with Classified Residual Reg

SCdry = f(TDCT)

Class Regression Threshold	16%
Class Residual Threshold	7%

	Regression n	Residual n	Min %e	Max %e
Class 1	12	12	-7.0%	16.4%
Class 2	13	14	-8.3%	12.7%
Class 3	10	9	-9.8%	9.1%

Model	LR	CR	CR w/corr
mean	17.5%	-1.6%	0.5%
|mean|	36.2%	13.5%	3.8%
st.dev	50.4%	18.1%	5.3%
st.dev ||	38.7%	11.9%	3.6%
co.var	34.7%	-8.8%	9.81
co.var||	93.5%	113.8%	1.39
|mean| + st.dev ||	74.9%	25.4%	7.4%

Table 21: Model #1 Predictions

				CR Modeler [SCdry = f(TDCT)]			
Data ID	Mission	Ind Var TDCT	Dep Var SCdry	reg class	res class	Predict	Pred %e
10	Grail-A	147.95	198	1	1	197.1	-0.5%
11	Grail-B	147.95	198	1	1	197.1	-0.5%
14	Ladee	204.3	240.7	1	1	247.0	2.6%
16	Lunar Prospector	30.6	158.7	1	2	161.1	1.5%
1	2001 Mars Odyssey	366.2	376.3	1	1	390.4	3.8%
12	Insight	586.4	608	1	1	594.4	-2.2%
35	Stardust	126.4	300	1	3	349.1	16.4%
31	New Horizons	347.7	508	1	2	492.8	-3.0%
17	M2020 - Perseverance	2232.2	2439	1	2	2464.2	1.0%
3	Contour	96.8	328	1	3	317.0	-3.4%
19	Mars Climate Orbiter	94.5	338	1	3	314.5	-7.0%
32	Osiris-Rex	591.6	860	1	3	853.6	-0.7%
7	Deep Space 1	114.2	373.7	2	2	398.7	6.7%
28	Messenger	216.2	485.2	2	1	444.9	-8.3%
33	Phoenix	318.2	597	2	1	572.5	-4.1%
15	Lucy	523.1	821	2	1	828.9	1.0%
8	Galileo	902.3	1289	2	1	1303.3	1.1%
4	Dart	240.6	560	2	2	577.1	3.1%
9	Genesis	163	494	2	3	556.7	12.7%
34	Psyche	844.5	1400	2	2	1429.3	2.1%
25	MAVEN	382.4	809	2	2	777.2	-3.9%
30	NEAR Shoemaker	124.9	487	2	3	501.5	3.0%
21	Mars Observer	511.2	1028	2	3	1061.5	3.3%
29	MSL - Curiosity	2142.4	3300	2	2	3261.0	-1.2%
5	Dawn	282.9	747.1	2	3	730.5	-2.2%
18	Magellan	463.2	1035	3	1	1057.4	2.2%
2	Cassini	1408.5	2581	3	2	2605.3	0.9%
24	Mars Recon Orbiter	419.2	1031	3	1	998.8	-3.1%
23	Mars Polar Lander	94.5	519	3	1	566.3	9.1%
13	Juno	726.2	1593	3	2	1587.2	-0.4%
26	MERA - Spirit	321.5	1018	3	2	983.3	-3.4%
27	MERB - Opportunity	321.5	1018	3	2	983.3	-3.4%
20	Mars Global Surveyor	130.7	666.3	3	2	698.5	4.8%
22	Mars Pathfinder	199.2	795	3	2	800.8	0.7%
6	Deep Impact	230.9	1042	3	3	939.8	-9.8%

Interpretable (class-wise) CR Model

Class	n	SCdry = f(TDCT)	TDCT(min)	TDCT(max)	mean %e	stdev %e
1	12	SCdry(c1) = 117.51 + 1.03 * TDCT	30.6	2232.2	3.54%	4.43%
2	13	SCdry(c2) = 225.93 + 1.40 * TDCT	114.2	2142.40	4.04%	3.37%
3	10	SCdry(c3) = 491.91 + 1.48 * TDCT	94.5	1408.50	3.79%	3.30%
				Model	3.80%	3.65%

Class	n	Corrector = f(TDCT)	Correction = - Corrector
1	12	Corrector(c1) = 51.50 + 0.15 * TDCT	Corrector(c1) = -51.50 + -0.15 * TDCT
2	14	Corrector(c1) = -11.61 + -0.01 * TDCT	Corrector(c1) = 11.61 + 0.01 * TDCT
3	9	Corrector(c1) = -94.46 + -0.05 * TDCT	Corrector(c1) = 94.46 + 0.05 * TDCT

As a class-wise predictor, the CR model performance is:

- $|\mu|\, regression\ =\ 13.5\,\%$
- $|\sigma|\, regression\ =\ 11.9\%$
- $Performance\ =\ +\,25.4\%\, max$

The class-wise prediction performance of CR is consistent with that of complex parametric methodologies for cost estimation. Inclusion of the residual corrector significantly improves the model's performance:

- $|\mu|\, residual\ =\ 3.8\,\%$
- $|\sigma|\, residual\ =\ 3.6\,\%$
- $Performance\ =\ |\mu|\ +\ 1|\sigma|\, error\ =\ +\,7.4\%\, max$

Correction in the predictor variable compensates for the algorithm's simplicity without introducing additional complexity, often resulting in fewer parameters or terms that decrease the model's degrees of freedom (*dof*). Complex models introduce multivariate, multi-term solutions that tailor the predictor response to the training data to optimize performance, consuming degrees of freedom. These approaches capture the multi-variable dependency and trend behavior through complexity. CR incorporates the influence of excluded variables and data nonlinearity as secondary relationships of the predictor through correction.

Scope Model Forecasting

The domain of the training dataset imposes a limit on the predictor; only known elements with known regression classifications can be predicted. To expand the predictor's utility, CR incorporates supervised classification for application as a forecaster. Forecast models predict an unknown response from a known predictor by linking the predictor to the known predictor model.

For CR, the link is achieved through an exhaustive classifier (EC). The EC is a simple algorithm that utilizes Feature F and Feature Combination FC vectors, along with their corresponding mean values, to establish critical thresholds for predicting class assignments.

The NIM dataset includes 45 numerical attributes a_i. A set of 3862 1st-order features F_i and 2nd-order feature combinations FC_i is constructed from NIM attributes a_i. The F_i and FC_i considered:

$$F_i = a_i \tag{37}$$

$$FC_i = a_i \cdot a_j \ (i \neq j) \tag{38}$$

$$FC_i = a_i / a_j \ \ (i \neq j) \tag{39}$$

Each F_i and FC_i is evaluated using a confusion matrix for classification accuracy. Better performers are selected for expansion to 3rd-order combinations to further improve accuracy by adding an extra multiplicative or divisor term. A supervised class assigner for Model #1 forecasting is:

$$Class\ Assigner = {}^{TDCT}\!\big/_{SC_{dry}{}^{2}} \tag{40}$$

$$Confusion\ Matrix = \begin{Bmatrix} 10 & 1 & 1 \\ 2 & 10 & 1 \\ 0 & 2 & 8 \end{Bmatrix} \tag{41}$$

$$Accuracy = \frac{28}{35} = 80\% \tag{42}$$

For Model #1, the supervised classifier (5) and the CR predictor share common a_i, although the class assigner may contain any combination of NIM attributes, based on supervised classification accuracy. The inclusion of supervised classification (5) establishes CR as a forecast method.

Funding System Use Cases

CR is well-suited to support the varying needs of the funding system:

- Strategic Planning – margin management, objective optimization
- Feasibility assessment – resource quantification
- Baseline definition and resource estimation – identification of best solution capability and needs
- Alternative assessment – comparison of alternatives' capability and needs
- Proposal generation and validation – requirement development, performance assessment, and proposed compliance
- Independent assessment – viability of proposed solutions
- Monitoring, onset growth identification, and risk mitigation

CR offers the enterprise a rapid, high-quality, and robust prediction and forecasting capability throughout the system life cycle *Figure 42*. CR utility supports the needs of the Enterprise, the program of interest, or serves as an independent assessment and monitoring resource. For demonstration purposes, the CR SCdry model is used to illustrate the application of CR modeling to meet each funding system's needs.

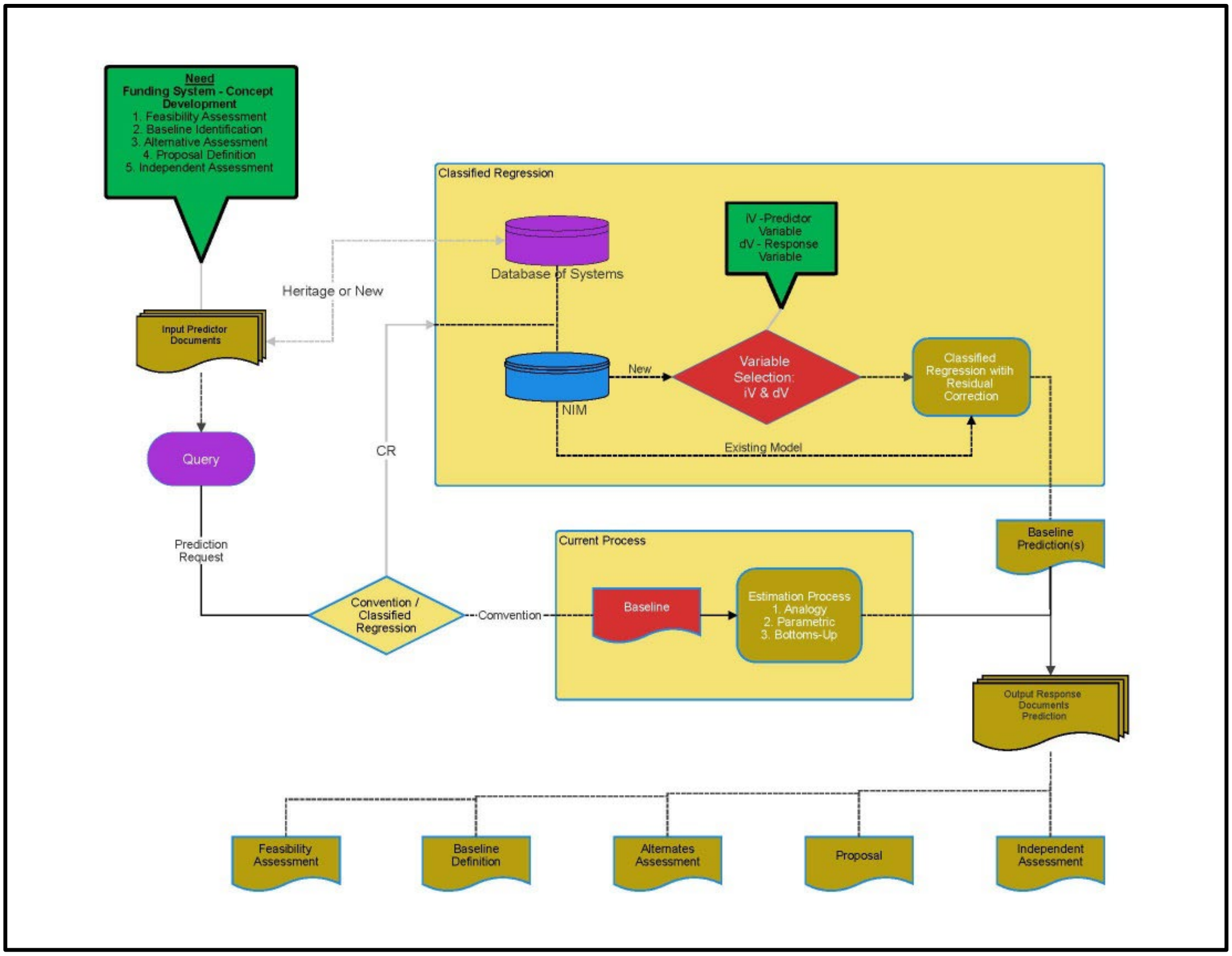


Figure 42: CR Contributions to the Funding System

Strategic Planning

Planning occurs at levels within the funding system. Long-term planning establishes NASA's Strategic Plan, outlining the Vision and Mission, Core Values, and Strategic Goals and Objectives. The Strategic Plan is a cyclical, ongoing process that is reconstituted every four years. Objectives are assigned to directorates, which determine the architectures to achieve goals. Architecture can include multi-mission or single-mission acquisitions, collaboration among objectives, or the repurposing of existing systems.

CR provides a utility to assess objective optimization by highly accurately predicting system resources. The regression-based classification of the NIM dataset identifies critical ranges and limitations for each mission class. Based on an evaluation of the NIM SCdry capability, the Class 3 elements represent the mission cluster with the best mass-to-value ratio. A mass-to-cost ratio quantifies the value. A summary of these is presented *Figure 43* and *Table 22*. The mission class 3 cluster and "like mission" attributes represent the best value for the dollar in terms of mass. This evaluation serves two purposes: it identifies program acquisition and realization structures that are best performers, based on the model attributes, and establishes a standard for acquisition expectations. The attributes of the cluster class of "similar to" missions achieve objectives at the most efficient SC dry resource level.

It would be prudent for NASA to evaluate the attributes and identify the drivers that differentiate Class 3 missions from other class clusters, and to implement Class 3 acquisition and realization practices across new activities to achieve the maximum return on investment. This statement assumes that SCdry is a parameter of interest for optimization.

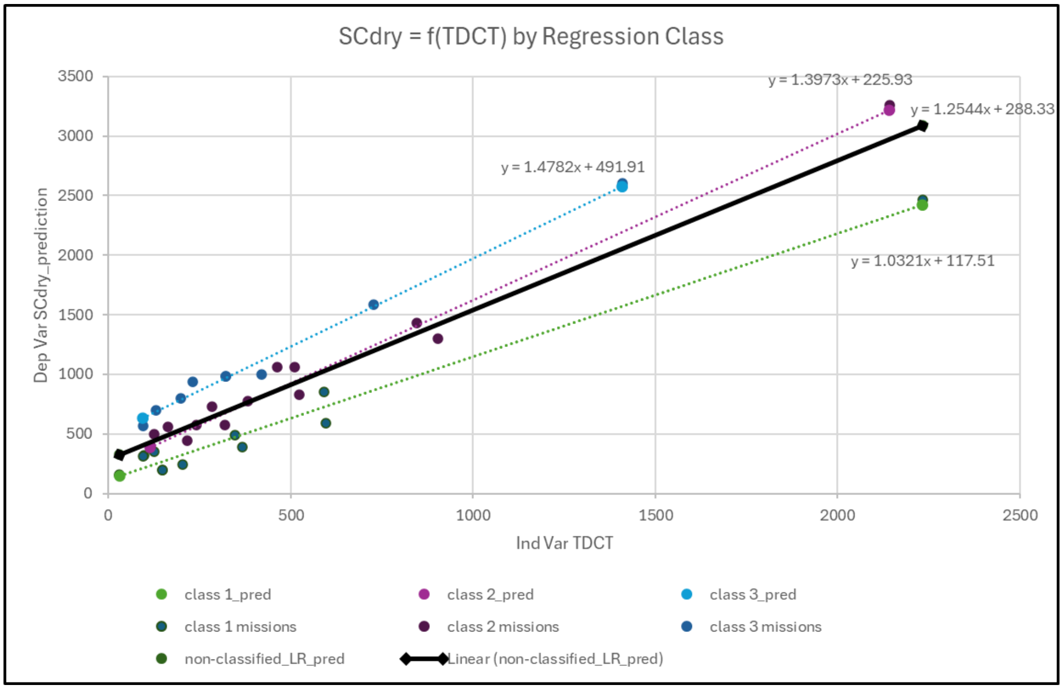


Figure 43: NIM CR SCdry Model by Class

NASA's four most recent launches —Psyche (Class 2), Lucy (Class 2), Mars 2020 Perseverance (Class 1), and Insight (Class 1) —are members of the lower-mass-to-cost Class 2 and Class 3 clusters. Using the boundaries defined in *Figure 43* NASA can identify a plan that categorizes a set of space mission classes, achieving NASA's long-term objectives within cost and schedule plan constraints.

Table 22: NIM Class 3 Missions - Optimized Mass To Cost

Mission	LY	SY	Type	Destination
Magellan	1989	1984	Orbiter	Venus
Cassini	1992	1985	Orbiter	Saturn
Mars Recon Orbiter	1999	1996	Orbiter	Mars
Mars Polar Lander (F)	1996	1999	Lander	Mars
Juno	2006	2001	Orbiter	Jupiter
MerA – Sprirt	2003	2000	Rover	Mars
MerB – Opportunity	2003	2000	Rover	Mars
Mars Global Surveyor	1996	1994	Orbiter	Mars
Mars Pathfinder	1996	1993	Rover	Mars
Deep Impact	2005	1999	Rendezvous	Comet

Feasibility Assessment

The output of Planning is a complement of space mission(s) that achieve NASA's long-term objectives. *Figure 43* defines the capability limits, based on historical NIM performance, that support mission feasibility assessment. Feasibility examines the ability to achieve the mission objective within specified constraints. For this example, the constraint is the budgeted cost limit. The assessment determines which NIM class is compliant with the cost limit and associated feasibility margin, measured in mass. For the cost cap of $500M in 2024 dollars:

- A class 1 "type" acquisition can achieve 633 kg at 84% confidence
- A class 2 "type" acquisition can achieve 915 kg at 84% confidence
- A class 3 "type" acquisition can achieve 1231 kg at 84% confidence

Assuming that the class "types" have differentiable attributes, class banding in SCdry mass identifies the feasibility margin for risk assessment and potential cost-risk mitigation alternatives.

Baseline Definition

The second element of the Funding System process is Program Structure, which advances planning concepts during the reconciliation process. The objective of the process is to develop and document a baseline concept. The baseline evaluates mission-oriented objectives, identifies system-level architecture, refines a candidate solution through comprehensive performance assessment and system trade studies, and constructs an architecture that supports a standard of mission success. The standard identifies the quantifiable measures for cost, schedule, and scope within acceptable risk.

Baseline architectures evolve from analogous methods. These, most often, rely on established standards of performance success, relying on adaptations of legacy systems. The NIM systems include launch access, instrument payload complements, spacecraft bus, communications, and data dissemination. In the CR model of SCdry, this includes evaluating legacy spacecraft and assessing CERs for each evaluated level of a spacecraft system. The CR SCdry model supports these efforts in a variety of methods:

- directly supports the application of "like" spacecraft to baseline assessment, Figure *44*
- allows for "clustering" of legacy missions to evaluate class-wise implementation and the system-level impact
- provides an understanding of comparable accommodations. Legacy systems that are represented within the same class cluster will exhibit similar behavior and paths from concept to realization. Legacy systems represented within different class clusters will exhibit dissimilar behavior and follow independent paths from concept to realization. The CR SCdry supports class-wise risk assessment when applied to a single opportunity.

Figure 44 considers the design reuse of an MRO mission from the perspective of SC dry mass, defined as the total development cost in year-of-occurrence dollars (2024), labeled TDCT. From a baseline perspective, the proposed baseline is not the MRO spacecraft bus. It is a "like" spacecraft, realized through a "like" process to the legacy MRO program bus. This considers not only the design details but also the method of program procurement, the standards and procedures used to achieve the SCdry product, and the complexity of the MRO mission's development. Identification of a legacy baseline considers far more than the geometry of a spacecraft bus.

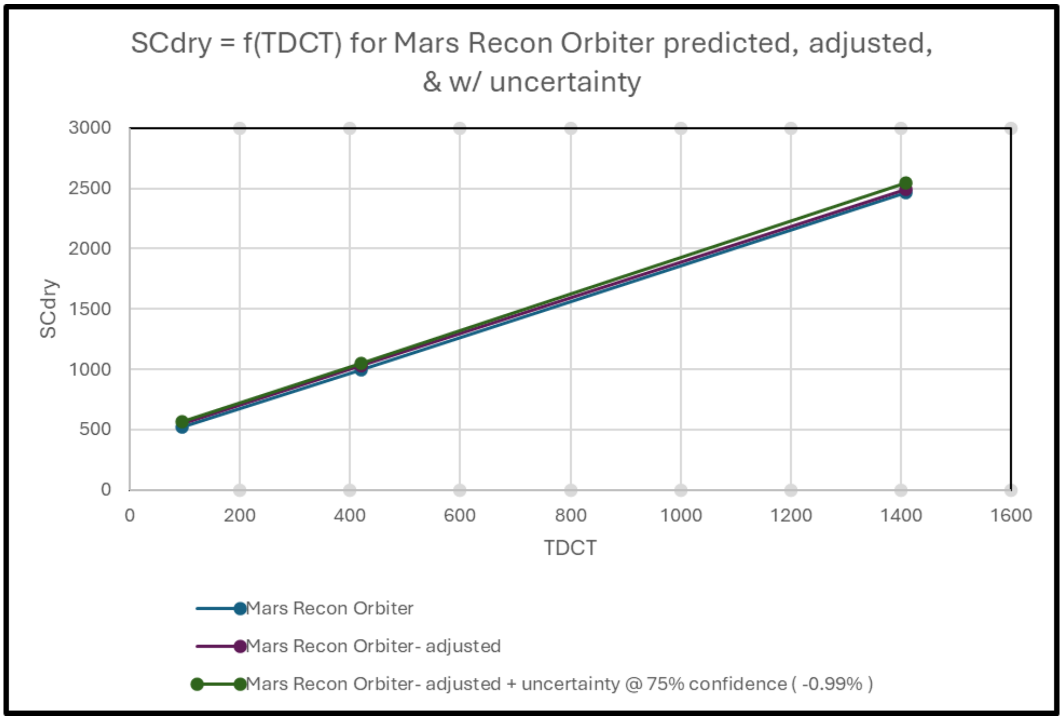


Figure 44: MRO Baseline Predictor Model

For this baseline assessment example, the MRO mission attributes identify TDCT = 419.2 M$ and SCdry = 1031 kg. The CR model predicts the SCdry_predict = 998.3 kg, underpredicting the realized performance by 3.1%. *Figure 44* lower curve (blue) predicts "like" MRO SCdry performance based on its class 3 assignment over the range of class 3 missions from 94.5 M$ to 1408.5 M$. Because the CR SCdry model underpredicts MRO performance, the prediction curve is shifted to the class 3 mean (purple), providing SCdry predictions at a 50% confidence level. For *Figure 44*, uncertainty (green) is applied to adjust the SCdry prediction at the 75% confidence level. From this chart, the legacy MRO mission can be scaled within the range of class capability, from \$ 94.5 M to \$1,408.5 M, at a specified confidence level of 75%. If the baseline includes a constraint in TDCT of 800.0 M$, an MRO "like" SCdry can be scaled to 1238.4 kg, based on the MRO predictor $SCdry = 440.41 + 1.33\,TDCT$. The estimate provides a baseline for reference, enabling the assessment of concept-level SCdry margin based on TDCT confidence, as well as the relationship between TDCT and SCdry for MRO “like” applications as TDCT evolves during development.

Alternative Assessment

Alternative assessment supports the baseline and development process by considering alternative components, methods, and acquisition strategies. As with the MRO “like” mission previously discussed, the CR SCdry model enables the assessment of all NIM elements within a standard, consistent framework. This is particularly insightful when selecting a baseline and comparing the baseline to alternatives. Each NIM element is supported by standard class-wise predictor models or the ability to uniquely model each NIM element, allowing for alternative assessment at both the class and element levels. The ability to assess classes is specifically critical if characteristics of each class can be isolated (procurement mechanism, redundancy philosophy, overall TRR level) and risk assessment can be evaluated across classes and among elements. It is critical during the alternative assessment phase that alternatives are evaluated within a common framework to isolate and eliminate inherent bias, error, and deceitful intentions. In the highly competitive interplanetary mission market, success and failure are closely intertwined, often driven by differing interpretations. Standardization of product generation and evaluation is a crucial element in achieving optimal objectives.

Proposal Generation

The MRO Baseline Predictor model provides a standard for proposal generation, whether in support of Request for Quote (RFQ) documentation or in developing a Proposal Response. From the previous example, the model predictor in TDCT of 800.0 M$ provides an MRO “like” SCdry of 1238.4 kg at a 75% confidence level. From the RFQ perspective, credible proposals should support the 1238.4 kg estimate. From a planning perspective, the 1238.4 kg represents a realized deliverable prediction that provides the basis for resource management and margin assessment for a conventional procurement. This also provides a quantifiable cap for an unconventional procurement.

Likewise, the CR SCdry predictor provides a validation measure for a proposal summation estimate and the ability to assess the impact of growth, reallocation, and overall resource limits. For this example, the model defines the effect of changes to TDCT upon SCdry and the ability to mitigate risk and ensure success.

Independent Assessment

For the MRO proposal baseline, assume that a Proposal Response uses a Dart "like" SCdry estimate. The CR SCdry model allows for assessment of that alternative in a compatible format with the MRO baseline. Assessment can be performed with respect to the DART predictor, a class-adjusted Dart predictor, or application of a level of confidence for the DART or Dart-class predictor. The consistency inherent to the CR SCdry model allows for direct comparison of MRO "like" versus Dart "like proposals, including assessment of different confidence levels of estimate and ancillary properties to SCdry, including both included and inherent margin.

Suppose a DART "like" reference is not established. In that case, the CR modeling and supervised classifiers permit evaluation of an Undefined "like" through forecast classification and associated prediction at either a class-wise or element-wise reference. A DART "like" SCdry predicts an SCdry of 1365.5 kg at TDCT of 800 M$ *Figure 45*.

It is noted that the class 3 MRO "like" mission predicts a lower value of the class 2 DART "like SCdry primarily due to the corrector applied to the MRO "like" SCdry.

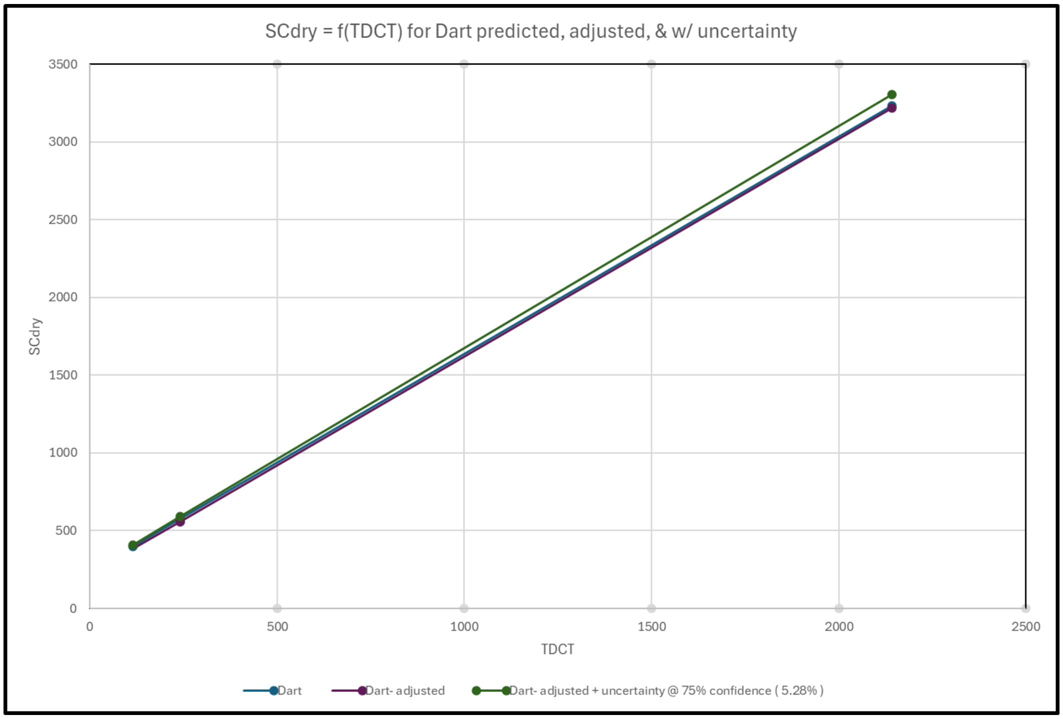


Figure 45: DART Assessment Predictor Model

Monitoring

Monitoring and risk mitigation are achieved through various mechanisms. CR provides several support options. The variable relationships established in the CR system model provide insight into dependencies that bypass the delays in cause-and-effect propagation. For the previous example, SCdry is the resource of concern, and TDCT is the attribute available for monitoring. As TDCT management identifies growth or reduction in TDCT, the cause-and-effect relationship is defined in either the MRO class-domain as $\Delta SCdry \approx 1.48\,\Delta TDCT$ or in the MRO element-domain as $\Delta SCdry \approx 1.33\,\Delta TDCT$. This monitor enables instant propagation to the SCdry resource, allowing mitigation either in the TDCT attribute or by identifying alternate attributes that can be managed to offset the ΔTDCT in terms of growth or to develop additional margin through ΔTDCT reduction.

An alternate method of monitoring is demonstrated in the Model #4 validation model presented in a subsequent section of this document. Model #4 establishes the predictive relationship between the proposed cost and the realized cost for the development of "uplift". Uplift is a method for establishing and managing margins. In Model #4, the uplift is defined by the predictor, which validates both the need for and the measure of cost margin applied in the proposal phase of a project. The approximately 20% growth should be used for all estimates for accurate budgeting or assigned as a margin at that phase of the project, to support accurate budgeting.

Cost Model

A set of cost models is constructed using SCdry as the predictor, defining the response as the total mission cost in 2024, in NSII-adjusted, and PECPI-adjusted dollars *Table 23*. All cost models are included in Appendix C.

Table 23: Total Cost Models with SCdry Predictor

SCdry	f()	Regression Margin	Residual Margin	Zero-class	|μ|	|σ|	Performance |μ| + |σ|	Compliance (15%)
TMCT	f(SCdry)	.41	.11	.05	9.0%	7.3%	16.3%	71.4%
TMCN	f(SCdry)	.32	.18	.03	9.4%	8.0%	17.5%	77.1%
TMCP	f(SCdry)	.37	.07	.05	7.0%	6.3%	13.2%	82.9%

A set of cost models is constructed using SCdry as the predictor, defining the response as the development cost in 2024, NSII-adjusted, and PECPI-adjusted dollars *Table 24*.

Table 24: Development Cost Models with SCdry Predictor

SCdry	f()	Regression Margin	Residual Margin	Zero-class	|μ|	|σ|	Performance |μ| + |σ|	Compliance (15%)
TDCT	f(SCdry)	.16	.14	.06	8.7%	5.9%	14.6%	88.6%
TDCN	f(SCdry)	.47	.24	.00	8.3%	7.6%	15.9%	85.7%
TDCP	f(SCdry)	.19	.12	.03	10.2%	8.1%	18.3%	80.0%

A set of cost models is constructed using SCdry or Launch Mass as the predictor, defining the response as the launch cost in 2024, NSII-adjusted, and PECPI-adjusted dollars *Table 25*.

Table 25: Launch Cost Models with SCdry or Launch Mass Predictors

SCdry	f()	Regression Margin	Residual Margin	Zero-class	|μ|	|σ|	Performance |μ| + |σ|	Compliance (15%)
TLCN	f(SCdry)	.26	.09	.06	7.0%	7.7%	14.8%	91.4%
TLCN	f(SCdry)	.26	.09	.03	7.6%	5.8%	13.4%	91.4%
TLCT	f(SCdry)	.34	.14	.06	8.6%	5.2%	13.8%	82.8%
TLCN	f(LM)	.39	.18	.04	10.3%	8.7%	19.0%	74.1%
TLCN	f(LM)	.39	.22	.04	11.5%	5.9%	17.3%	77.1%
TLCT	f(LM)	.20	.10	.04	8.9%	8.9%	17.8%	80.0%

A modified set of cost models is constructed using SCdry as the predictor, defining the response as the annual operations cost in 2024, NSII-adjusted, and PECPI-adjusted dollars Table 24. This model adjusted the total operations cost to the annual operations cost rate.

Table 26: Operations Cost Rate Models with SCdry Predictor

SCdry	f()	Regression Margin	Residual Margin	Zero-class	|μ|	|σ|	Performance |μ| + |σ|	Compliance (15%)
ROCT	f(SCdry)	.32	.17	.00	8.6%	11.2%	19.7%	82.8%
ROCN	f(SCdry)	.32	.26	.00	8.4%	8.3%	16.7%	82.8%
ROCP	f(SCdry)	.29	.19	.01	11.9%	9.2%	21.0%	74.3%

Schedule Model

A set of schedule models is constructed, defining the response as the development schedule duration using SCdry or development cost in 2024, in NSII-adjusted, and PECPI-adjusted dollars *Table 27*. All schedule models are included in Appendix D.

Table 27: Development Schedule Models

	f()	Regression Margin	Residual Margin	Zero-class	|μ|	|σ|	Performance |μ| + |σ|	Compliance (15%)
S1	f(SCdry)	.22	.08	.01	3.4%	2.5%	5.9%	100%
S1	f(TDCT)	.20	.05	.00	2.8%	2.1%	5.0%	100%
S1	f(TDCN)	.10	.05	.00	2.8%	2.2%	5.0%	100%
S1	f(TDCP)	.15	.05	0	3.0%	2.3%	5.3%	100%

System Model

A series of models is constructed demonstrating the use of CR as a top-level system predictor. The criteria for a successful demonstration include the ability to predict a critical element for each component in the system triad and for the top-level component as a stand-alone or as a sum of element models. From a methodology comparison, the top-level component predictor is consistent with analogous and parametric methods. In contrast, the element summation is consistent with a bottom-up process.

The system-level triad predictor consists of three components:

- Scope predictor for the spacecraft dry mass, $SCdry$
- Schedule predictor for total development cost, $S1$
- Cost predictor for total mission cost, $TDCT_i$.

A series of modeling approaches for total mission cost is evaluated:

- the Total cost model $TMCT = f(Scdry)$
- the Sum cost model $TMCT_{sum} = \sum TDCT = f(Scdry) + TLCT = f(SCdry) + ROCT = F(Scdry) \cdot ops\ duration$
- the Average cost model $TMCT_{average} = {TMCT + TMCT_{sum}}/{2}$
- the Weighted average cost model $TMCT_{weighted=} {TMCT \cdot \frac{1}{k_1} + TMCT_{sum} \cdot \frac{1}{k_2}} \Big/ {(\frac{1}{k_1} + \frac{1}{k_2})}$ where $\frac{1}{k_1}$ is the $TMCT$ model accuracy and $\frac{1}{k_2}$ is the $TMCT_{sum}$ model accuracy.

The *mean* is the absolute mean error of the predictor, the *stdev* is the standard deviation of the absolute mean error, and the *perfor* is the model performance accuracy, where $performance = |\mu| + |\sigma|$ and for normally distributed data, represents an 85% confidence interval in the prediction *Table 28*. The model comparison is included in Appendix E.

The resultant simple system triad predictor model relies on a single NIM attribute, SCdry. Using SCdry as the predictor and a set of CR models, the system triad components of total mission cost, total development duration, and SCdry are predicted.

The system model is initialized with a prediction of the SCdry response. This process is traditionally initiated with a cost not to exceed limit:

- $SCdry$ is expressly stated,
- The mission level ($TMCT$) is stated, or
- The space vehicle appropriation ($TDMT$) is stated.

The system model continues with the prediction of development duration (S1):

- $S1 = f(SCdry)$, or
- $S1 = f(TDCT)$.

The system model continues with the prediction of cost components:

- The total mission cost is predicted as $TMCT = f(SCdry)$,
- The total development cost is predicted as $TDCT = f(SCdry)$,
- The total launch cost is predicted as $TLCT = f(Scdry)$ or $TLCT = f(LM)$ where $LM = Scdry + ballast + fuel$, and
- The operations cost predicted as an annual rate ROCT = f(SCdry) · mission duration.

The system model is completed with the prediction of the total mission cost:

- $TMCT = f(SCdry)$ from the cost component predictor,
- $TMCT_{sum}$ from the sum of the $TMCT_{sum} = TDCT + TLCT + ROCT \cdot mission\ duration$

- $TMCT_{average} = {TMCT + TMCT_{sum}}/_{2}$, or
- $TMCT_{weighted} = {TMCT \cdot \frac{1}{k_1} + TMCT_{sum} \cdot \frac{1}{k_2}}/_{(\frac{1}{k_1} + \frac{1}{k_2})}$

The system model and performance of the predictor models are presented in *Table 28*.

Table 28: Triad Model Results

	Predictor	|μ|	|σ|	Performance |μ|+|σ|	Compliance (15%)	Appendix
Scope						
$SCdry$	$stated$	-	-	-	100%	-
$SCdry$	$f(TDCT)$	3.8%	3.6%	7.4%	97.1%	B
$SCdry$	$f(TMCT)$	4.1%	4.6%	8.8%	94.3%	B
Schedule						
$S1_{dev}$	$f(SCdry)$	3.4%	2.5%	5.9%	100%	D
$S1_{dev}$	$f(TDCT)$	2.8%	2.1%	5.0%	100%	D
Cost						
$TDCT$	$f(SCdry)$	8.7%	5.9%	14.6%	88.6%	C
$TLCT$	$f(SCdry)$	8.6%	5.2%	13.8%	82.8%	C
$TLCT$	$f(LM)$	8.9%	8.9%	17.8%	80.0%	C
$ROCT$	$f(SCdry)$	11.9%	9.2%	21.0%	74.3%	C
Total Mission Cost						
$TMCT$	$f(SCdry)$	9.0%	7.3%	16.3%	71.4%	C
$TMCT_{sum}$	$TDCT + TLCT + TOCT$	9.6%	11.8%	21.4%	85.7%	E
$TMCT_{average}$	$\frac{TMCT + TMCT_{sum}}{2}$	6.7%	5.7%	12.5%	88.5%	E
$TMCT_{weighted}$	$f(TMCT, TMCT_{sum})$	6.7%	5.4%	12.1%	88.5%	E

The CR model's performance is stated at 84% confidence level. This assumes a stated initiation attribute (SCdry) at 100% confidence. If SCdry were derived from either the TDCT or TMCT predictors, the confidence level for SCdry as a predictor would be 84%. As SCdry propagated through the model as a predictor, in models at 84% confidence, the resulting predictions would exhibit 70.5% do the propagation of confidence (84% confidence at 84% confidence is 70.5% confidence). This allows for a second layer in a neural network path for the system predictor to maintain the NASA standard 70% confidence level. Effectively, using a predicted attribute as a predictor of a second attribute maintains the NASA standard 70% confidence level.

Compliance represents the percentage of the NIM elements that comply with the element-wise performance goal of prediction performance goal not to exceed 15% $(performance < 15\%)$. The operations cost model F_{Ops} is based on an annual rate model that includes all NIM missions, including four failed missions that did not commence science operations (Mars Climate Orbiter, Mars Observer, Mars Polar Lander, and Mars Global Surveyor). This is noted, as it may skew the predictive model; however, it is representative of the full NIM population's behavior.

Validation Models

To further validate CR, CR is applied to two published, non-NIM data sets, Models #3 and #4. The model configuration, predictors, and performance are included in Appendix F.

Model #3 compares CR performance to an early assessment of neural networks (NNs) and regression analysis (Shtub & Ronen Versano, 1999). The study considered the aggregated cost AC of thirty-six steel pipe bending projects. Hart (Hart et al., 2012) compared the Method of Improved Cost Estimation from Historical Data (MISERLY) with Shtub's earlier results. Their cumulative predictions were duplicated, and the algorithm performance was determined *Table 29*. The NIM model was adjusted to incorporate Shtub's dataset for a pair of pipe diameter predictors and AC response. Thirty-five elements from Shtub's data were used to train the predictor model;

a thirty-sixth element was omitted. The predictors include the inner diameter (d_1) and outer diameter (d_2). The CR model outperforms the Shtub and Hart techniques.

Table 29: Model #3 Performance Comparison

Cost Model	Abs mean error |μ|	Std Dev of Abs mean error |σ|	Model Performance |μ| + |σ| at 1σ
Regression Baseline	24.9%	15.3%	40.2%
Neural Network	16.4%	14.1%	30.5%
MISERLY	13.5%	13.7%	27.2%
Classified Regression (d_1)	3.3%	2.0%	5.4%
Classified Regression (d_2)	3.4%	2.4%	5.8%

Model #4 develops from Servranckx (Servranckx et al., 2021). Servranckx evaluated reference class forecasting (RCF), a supervised learner. Servranckx's corrector incorporates "uplift" based on historical project performance. The CR model prunes the data to the 36 Combination project types, limiting the data to "like" projects. This dataset allows for a training set of 35 project elements. The results of the methods are summarized *Table 30*. CR demonstrates improved prediction performance compared to Servanckx's method.

Table 30: Model #4 Performance Comparison

Predictor Model Performance Comparison	Abs mean error |μ|	Std Dev of Abs mean error |σ|	Model Performance |μ| + |σ| at 1σ
Cost Model			
Baseline (Servranckx et al., 2021)	24.8%	27.7%	52.5%
Reference Class Forecaster (Servranckx et al., 2021)	19.8%	22.7%	42.5%
Classified Regression	3.4%	3.3%	6.6%
Duration Model			
Baseline (Servranckx et al., 2021)	26.6%	46.0%	72.6%

Reference Class Forecaster (Servranckx et al., 2021)	22.9%	42.3%	65.1%
Classified Regression	7.9%	8.0%	15.9%

Sensitivity

Sensitivity provides validation by quantifying the model's ability to generalize. Predicted R-squared is an $(N\ x\ N-1)$ method of evaluating the model's sensitivity to individual training elements. The baseline model includes $(N = 35)$ data elements, where the Predicted R-squared models are developed using $(N - 1 = 34)$ data elements. The omitted element is applied as a test element to the manipulated $(N-1)$ model. The model is reconstructed with $(N-1)$ elements and, based on the baseline test element's original class, the Nth element is predicted. The manipulated model prediction is compared to the baseline prediction, and the results are compared *Table 31*. The sensitivity of the CR to the small data sample was minimal when evaluated using R-predicted performance within the SCdry predictor model, with a slight degradation in prediction performance from 10.3% to 12.8%. The sensitivity analysis predictions are presented *Table 32*.

Table 31: Sensitivity Performance

Statistic	R-Predicted Predictions	Baseline Model Prediction
Abs Mean error, $\lvert\mu\rvert$	7.1%	5.4%
The standard deviation of error $\lvert\sigma\rvert$ about $\lvert\mu\rvert$,	5.7%	4.9%
Co-variation of absolute mean, ${}^{\lvert\mu\rvert}/_{\lvert\sigma\rvert}$	1.30	1.31

Table 32: R Predicted Sensitivity Analysis

		NIM Database		Validation Model - testing data set				Prediction Model - testing data set			
ID	Mission	DMC	SCdry	reg class	res class	Predict	Pred %e	reg class	res class	Predict	Pred %e
2	MERA - Spirit	321.5	1018	1	2	978.2	3.9%	1	2	984.8	3.3%
5	MERB - Opportunity	321.5	1018	1	2	978.2	3.9%	1	2	984.8	3.3%
4	Mars Pathfinder	199.2	795	1	2	799.5	-0.6%	1	2	800.2	-0.7%
3	Mars Global Surveyor	130.7	666.3	1	2	696.2	-4.5%	1	2	696.8	-4.6%
1	Deep Impact	230.9	1042	1	3	920.5	11.7%	1	3	979.8	6.0%
7	Cassini	1408.5	2581	1	2	2677.1	-3.7%	1	2	2625.2	-1.7%
19	Mars Recon Orbiter	419.2	1031	1	2	1145.0	-11.1%	1	2	1132.2	-9.8%
6	Juno	726.2	1593	1	2	1592.7	0.0%	1	2	1595.5	-0.2%
18	Mars Polar Lander	94.5	519	1	1	637.9	-22.9%	1	1	571.2	-10.1%
24	Osiris-Rex	591.6	860	2	1	892.4	-3.8%	2	1	904.0	-5.1%
14	Lucy	523.1	821	2	1	840.8	-2.4%	2	1	818.5	0.3%
12	Galileo	902.3	1289	2	1	1373.3	-6.5%	2	1	1291.7	-0.2%
25	Phoenix	318.2	597	2	2	581.1	2.7%	2	2	672.0	-12.6%
22	Messenger	216.2	485.2	2	2	546.2	-12.6%	2	2	527.3	-8.7%
26	Psyche	844.5	1400	2	2	1419.7	-1.4%	2	2	1418.4	-1.3%
9	Dart	240.6	560	2	2	562.9	-0.5%	2	2	561.9	-0.3%
11	Deep Space 1	114.2	373.7	2	2	384.7	-3.0%	2	2	382.7	-2.4%
21	MAVEN	382.4	809	2	2	755.7	6.6%	2	2	763.0	5.7%
17	Mars Observer	511.2	1028	2	2	936.2	8.9%	2	2	945.7	8.0%
13	Genesis	163	494	2	2	445.9	9.7%	2	2	451.9	8.5%
15	Magellan	463.2	1035	2	3	1113.7	-7.6%	2	3	1044.2	-0.9%
20	MSL - Curiosity	2142.4	3300	2	2	2817.0	14.6%	2	2	3259.0	1.2%
10	Dawn	282.9	747.1	2	3	758.2	-1.5%	2	3	761.4	-1.9%
23	NEAR Shoemaker	124.9	487	2	3	494.2	-1.5%	2	3	513.6	-5.5%
16	Mars Climate Orbiter	94.5	338	3	3	286.7	15.2%	3	3	312.3	7.6%
27	M2020 - Perseverance	2232.2	2439	3	2	5686.5	-10.1%	3	2	2452.0	-0.5%
8	Contour	96.8	328	3	1	292.8	10.7%	3	1	315.1	3.9%
33	Grail-A	147.95	198	3	1	202.9	-2.5%	3	1	177.3	10.5%
34	Grail-B	147.95	198	3	1	202.9	-2.5%	3	1	177.3	10.5%
32	Ladee	204.3	240.7	3	1	252.4	-4.9%	3	1	227.0	5.7%
31	2001 Mars Odyssey	366.2	376.3	3	1	388.2	-3.2%	3	1	369.9	1.7%
30	Insight	596.4	608	3	1	551.1	9.4%	3	1	573.1	5.7%
35	Lunar Prospector	30.6	158.7	3	2	140.6	11.3%	3	2	133.8	16.7%
28	New Horizons	347.7	508	3	2	472.2	7.1%	3	2	467.7	7.9%
29	Stardust	126.4	300	3	3	363.7	-21.2%	3	3	350.7	-16.9%

Predictor Comparison

In Wertz, an example mission is used to demonstrate the predictor function of (32). This predictor, based on NASA historical cost data, provides budgetary estimates and top-level trade studies at NASA HQ. The algorithm predicts full program cost for development plus the first flight unit, consistent with the CR TDCT model. Predictions are defined as the median cost or 50% confidence level for the single prediction. A database of predictions and performance would be required to determine the standard deviation and additional uncertainty, thereby improving the confidence level of the estimate. Uncertainty of a single-variable single-term algorithm requires four predictions to establish the standard deviation. Equation (32) includes nine variables.

The example mission assumes a spacecraft dry mass of 664 kg (SCdry = 664) and (32) predicts a development cost of $268 M (in 2010 dollars). Since the CR TDCT model is independent of NSII type inflation indices, the "actual value" is assumed to be $268. Other variables input into (32) for prediction include:

- Ddry Mass – 664 kg
- Power – 653 watts
- Data Rate Percentile – 0.50
- Design Life – 36 months
- Percent New – 0.68
- Planetary – 0 (no, which causes a bit of a problem – lunar missions are the limitation)
- ATP Date Year – 1987
- Instrument Complexity Percentile (fraction) – 0.50
- Team Experience – 2.0

- Year Dollar of Output – 2010
- Regression Model Output - $268 M

Using the TDCT, an interplanetary space mission defined with a SCdry = 664 (kg) and target DMC = 268 ($M) has a DMC/SCdry ratio of 0.404. The DMC module classifier assigns this mission regression class 2. If the mission were misclassified as Class 1 or Class 3, the results would deviate from the baseline Class 2. These results are shown in *Table 33*.

Table 33: Wertz Predictor Algorithm Comparison

	SCdry	DMC	Reg Class	Res Class	Prediction	% error
Wertz baseline	664	268	-		268	0 %
DMC Class 1	664	268	1	1	515.6	-92.4 %
DMC Class 2	664	268	2	1	246.5	8.0 %
DMC Class 3	664	268	3	3	238.9	10.8 %

The results of this validation analysis demonstrate the CR's accuracy in incorporating unstructured data with the CR predictor model and the credibility of CR as a predictor, showing consistent and equally accurate performance when compared to proprietary NASA planning algorithms.

Small Data

Small data is discouraged in AI and ML. The reluctance stems from statistical uncertainty. The formula for uncertainty is:

$$U = \sqrt{\frac{\Sigma(x_i - \mu)^2}{n(n-1)}} \tag{43}$$

where x_i is the value of the i^{th} dataset element, μ is the mean of the dataset, and n is the number of elements in the dataset. Incorporating uncertainty in a predictor involves using confidence intervals (CIs). CI formulas are based on the Central Limit Theorem (CLT). For large samples $\bar{X}$, the CLT states that x_i are normally distributed with a mean μ and a standard deviation $\sigma/\sqrt{n}$. For populations with a small sample ($n \leq 30$) CLT assumptions may not apply. This conclusion is based on an inability to determine if the small sample is normally distributed or representative of the population. If the data are normally distributed, the CLT should apply regardless of the sample size.

When the population σ is unknown and the normal approximation is not valid, the sample standard deviation s is substituted, and the distribution is evaluated using Student's t-distribution, $(n-1)$ *dof*. The t-distribution is centered about zero and exhibits the qualitative bell-shaped curve with heavier tails than the normal distribution. As the sample size increases, the Student's t-distribution converges to the normal distribution. This occurs as the sample size approaches ($n \geq 30$), the normal distribution applies. The small sample is defined when ($n < 30$), although this is not an established standard.

The NIM dataset element quantity is near the transition from small to large, while class clusters are composed of small data. The NIM also represents a population. For the Model #1 class assignments, and numerous other models not discussed, the application of Student's or normal distribution did not prove significant. The primary concern with small data is the model's ability to capture natural variability without oversimplification or overfitting. Data variability and model variability are compared using two measures, the R^2 and PCC *Table 34*.

Table 34: Small Data Correlation

Coefficient	Raw Data	Linear Regression	Classified Regression	CR with Correction
R^2	0.8576	1.000	0.8730	0.8638
PCC	0.9261	1.000	0.9343	0.9294

Both the R^2 and PCC exhibit quality correlation for CR and CR with correction, capturing data variability and statistical tendencies within the small data.

Chapter 5: Conclusions and Recommendations

Conclusions

Initial Algorithm

Conclusion #1

The exhaustive classifier offers a high-quality classification tool for the NIM data. The utility of the EC is limited to classification, specifically supervised classification. Attempts to resolve the classification expressions as a predictor algorithm were not sufficiently accurate to warrant future development.

Predictor Variable Assessment

Conclusion #2

Using the results of the predictor variable assessment, a “best fit” linear regression model of the correlation between NIM mass and cost attributes is developed, utilizing the Then-$ cost methodology. The coefficient of determination for SCdry and SVbus mass predictors exceed $R^2 > 0.90$, highly correlated

Conclusion #3

A comparison of NSII and PECPI inflation-adjusted costs shows a higher correlation in mass-to-cost data for PECPI. This implies that improved model performance may be achievable using an alternate inflation index to the current standard.

A comparison of the PECPI inflation-adjusted costs with the unadjusted year-of-incurring or Then-$ cost shows that the Then-$ costs offer the best overall predictor correlation.

Inflation indices measure the rate of cost inflation. This measure does not capture the process improvement or cost-saving benefits of technological advances. Applying a cost inflator means that the same activity today will require the same effort at an inflated cost rate. Therefore, it is theorized and demonstrated in principle that as technology advances and process improvements are realized, effort decreases at an inverse rate to the inflation increase, resulting in a net zero impact.

Conclusion #4

Through CR, it is demonstrated that complex systems of systems, their system triad components, and component elements can be predicted with high accuracy (<15%) based on historical “like” system data. It is also demonstrated that forecast data, i.e., non-training data, can be successfully incorporated into the CR model via supervised classification.

Conclusion #5

Complex systems of systems based on standardized processes for acquisition and realization exhibit cluster classification behavior that can be modeled. These systems are predictable and represent complicated rather than complex system behavior. They are founded upon standardized, engineered processes that follow proven design, analysis, and verification flows. The ability to predict future system performance from prior performance is highly correlated.

Classified Regression

Conclusion #6

The research developed a predictor methodology based on classification regression with correction for the small-sample residual problem in complex system data. The predictor demonstrated utility for simple predictor response modeling of system triad elements and for complex modeling of system triad components and the system triad as a whole. The significant contributions of this research include predictor modeling that:

- Manages small sample datasets with very high prediction accuracy: measured using prediction of testing/training data; comparison to comparable, industry standard, simple models; and as compared to more detailed and complex models and projected missions
- Infuses Gaussian (normal) distribution through classification to create a multi-regression solution (class-dependent solutions)
- Corrects residual heteroscedastic behavior through residual classification by creating a behavior-dependent corrector based on the classified regression solutions, improving prediction
- Incorporates a unique testing sample dataset through $N-1$ training with a single test sample for small-sample datasets. Testing is repeated N times, creating a $N-1\; x\; N$ (training x testing) validation space for the module/model
- Improves classical "cost modeling" and expands the application to predict elements of each component of the system triad successfully.

Conclusion #7

Classified regression is an improvement to conventional regression analysis methodologies. The method is not highly coupled to the coefficient of determination, R^2, as other methods are. Regression practice is to improve model performance by introducing higher-order predictors, additional predictors, or higher-order predictor functions. All of these techniques compromise the model's integrity through *dof* reduction, overfitting, and tailoring the method to specific applications. Classified regression demonstrates compatibility with both high- and low-capacity

R-squared correlations. It is shown that CR mimics input data correlation, through R^2 and Pearson Coefficient and matches the input correlation in the output prediction. This is unique to CR.

CR as a System Analysis Tool

The method of CR provides a Systems Engineering resource as a $predictor \rightarrow response$ modeling tool that supports the system triad component and complete system prediction needs:

- Trend Analysis – classification identifies clusters of program implementations that optimize, normalize, and penalize system performance based on predictor to response behavior.
- Margin Analysis – accurate prediction and forecasting support adequate margin allocation. CR behavior analysis can predict the cause-and-effect relationships for increases or decreases in quantitative attributes.
- Driver identification – CR, as a component of system modeling, provides a quantitative measure of the inter-attribute relationships of the system. Evaluation of these relationships identifies critical relationships and, within the structure of a system optimizer, the essential drivers for response attributes and system optimization. For CR modeling based on linear regression, the predictor slope provides a measure for the "criticality" of a predictor-response relationship, where a "higher slope" model indicates elevated dependency between the predictor and response and "critical driver" status
- Requirement-to-Resource – the fundamental purpose of CR is to provide a higher quality model for attribute-to-resource prediction. Attributes are derived directly or indirectly from the mission objectives and resultant requirements. The CR model can be reverse-engineered to identify behavior related to requirements and resources directly through the model equations or indirectly through cluster classes.

- Baseline Analysis – CR predicts resource performance based on the input predictor attribute. A set of CR models provides a system assessment of the baseline or other.
- Alternate Assessment – CR predicts resource performance based on the input predictor attribute. A set of CR models provides a system assessment of an alternative or of other options. Using a standard CR model for baseline and alternative assessment offers a “like” measure and evaluation.
- Proposal Generation – CR predicts resource performance based on the input predictor attribute. A set of CR models provides a system-level assessment of the proposal, verifying the proposed system triad.
- Independent Assessment – CR is based on the historical performance of “like” systems. The clustered classes provide a “trend” of performance that is assessable to procurement systems, providers, acquisition structures, and philosophy. Trend analysis identifies optimal performance, while the CR model enables assessment of the range of classes across standard inputs. Supervised classification allows for consideration of any system, regardless of its heritage. Legacy systems can be directly correlated with past performance and associated classification. CR enables an independent assessment at either a class-wise or, when a legacy system is identified, an element-wise metric.
- Performance Monitoring – As a system modeler, the CR models define a network of identifiers for system prediction. In an operational system, there is a system inertia for the propagation of growth. Growth is often rooted in a “change” that occurs in one component of the system triad and eventually propagates throughout the system, manifesting as a delta resource. By establishing a system modeler, the change in any modeled system attribute directly correlates with the response attributes. As changes in source attributes are

identified, their effects across the system can be predicted, managed, and mitigated independently of the normal propagation process.

- Growth Onset Identification – Similar to performance monitoring, critical system response attributes can be reverse-engineered to determine critical “root causes” for growth (growth considered undesirable). By identifying essential causes, system engineering can more actively monitor system development, performance, and potential growth.
- Mitigation Assessment – Similar to growth onset identification, an extension of this process is to evaluate critical “root mitigators” or attributes that may negate or inversely affect a critical growth attribute. In this capacity, the system model can monitor, manage, and mitigate growth or other undesirable effects, independent of the information flow and distribution processes, thereby improving the effectiveness of system management.

Future Work

CR is a building block for a complete system predictor MBSE tool. The results of this thesis demonstrate the high performance of CR as a predictor of complex systems of systems, even with small populations and high-quality data. The limits of CR include:

- Striated linear correlation in the data, where the data demonstrates “banding” of clusters in a linear pattern, opposed to
- Predictor within the domain boundary of the class model
- Class-wise element quantity greater than four elements for the determination of the standard deviation
- A lower model element limit was not determined. Most modeling considered datasets of 35 elements for training and class-wise assignment of at least four elements

The further development of CR will involve the creation of a predictor matrix and a predictor graph, comprising a neural network of predictors, where model optimization is determined using Hamiltonian circuit analysis, bagging, and boosting *Figure 46*. Cluster classifiers for supervised class assignment were demonstrated within the context of this thesis.

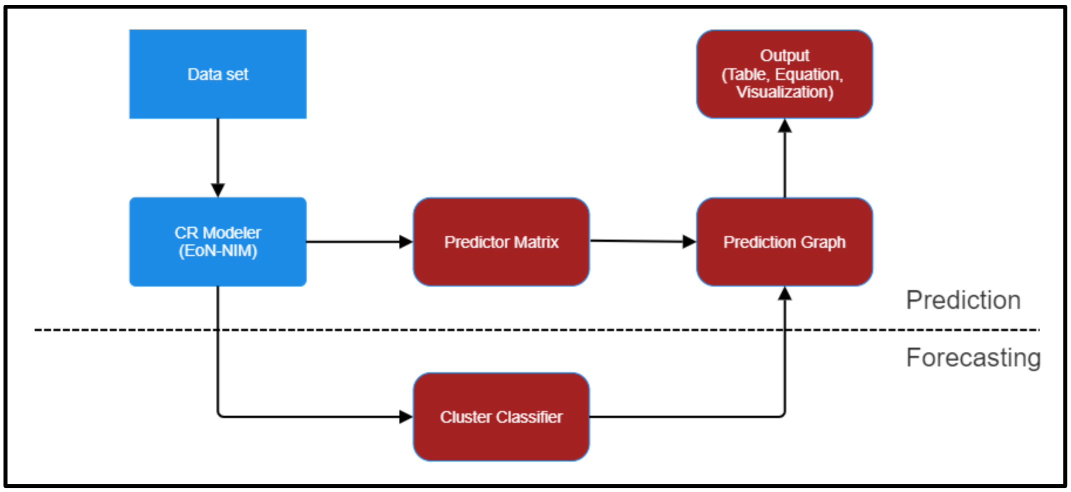


Figure 46: System Predictor

The prototype of the neural path predictor was demonstrated using total mission cost models. Through the inclusion of the $TMCT_{sum}$, a preliminary concept of bootstrap aggregating was successfully demonstrated. The $TMCT_{average}$model demonstrated the concept of model fusion, traditionally applied to weak learners, to improve the overall model performance. The final example demonstrated enhanced performance through the use of the $TMCT_{weighted}$ model where a simple $^1/_{k_{accuracy}}$ weighting was deployed. The neural path predictor would implement similar techniques to develop a comprehensive system predictor that is adjustable based on the initial system predictor(s) and optimized for the system response *Figure 47*.

$$\begin{bmatrix} \square & \cdots & \square \\ \vdots & \ddots & \vdots \\ \square & \cdots & \square \end{bmatrix} \rightarrow \{A, B, C, D\} \rightarrow A \begin{cases} B \rightarrow C \rightarrow D \\ (B \cap C) \rightarrow D \\ D \end{cases}$$

Figure 47: Predictor Neural Path Construction

Works Cited

Bell, K. D., & Hsu, L. A. (1995). Balancing performance and cost for cost-effective satellite systems design using an integrated cost engineering model. *1995 IEEE Aerospace Applications Conference. Proceedings*, 153–167. https://doi.org/10.1109/AERO.1995.468872

Bitten, R. E. ; E. M. M. & R. C. K. (2023). Cost Estimating of Space Science Missions. In *NASA Headquarters* . AEROSPACE REPORT NO. ATR-2013-00108. https://science.nasa.gov/wp-content/uploads/2023/04/secure_MAHR_BITTEN_Aerospace_Costing_Space_Science_Missions_ATR_2013_00108.pdf

Bond, Peter. (2011). *Jane's Space Systems and Industry 2011-2012*. Janes Information Group.

Bui, J. ; O. N. I. ; R. J. K. ;Corso, M. L. ; T. J. A. (1996). *Functional Cost-Estimating Relationships for Spacecraft*. https://apps.dtic.mil/sti/pdfs/ADA308665.pdf

Bureau of the Fiscal Service. (2025, February 25). *Executive Summary to the Fiscal Year 2024 Financial Report of U.S. Government*. U. S. Department of the Treasury. https://www.fiscal.treasury.gov/reports-statements/financial-report/unsustainable-fiscal-path.html

Callahan, J. (2025). *The Planetary Exploration Budget Dataset*. The Planetary Society. https://www.planetary.org/space-policy/planetary-exploration-budget-dataset

Chollet, F. ; A. J. J. (2017). Image Classification on Small Datasets with Keras. In *Posit AI Blog*. r-bloggers. https://blogs.rstudio.com/tensorflow/posts/2017-12-14-image-classification-on-small-datasets/

Clowney, Lt. Col. P. J. D. & S. S. (2016). Department of Defense Acquisition Program Terminations: Analysis of 11 Program Management Factors. *Defense ARJ*, *23*(3), 298–328. https://www.dau.edu/sites/default/files/Migrate/ARJFiles/ARJ78/ARJ78-Clowney_Article03.pdf?Web=1

Crisp, N. H., Smith, K. L., & Hollingsworth, P. M. (2019). An integrated design methodology for the deployment of constellations of small satellites. *The Aeronautical Journal*, *123*(1266), 1193–1215. https://doi.org/10.1017/aer.2019.57

Dalpiaz, D. (2016). *Applied Statistics with R*. https://book.stat420.org/index.html#about-this-book

Department of Defense. (2022). *System Engineering Guidebook*. Office of the Under Secretary of Defense for Research and Engineering .

Dockstader, R. (2025, June 5). *NASA interplanetary mission (NIM) dataset*. Zenodo. https://doi.org/10.5281/zenodo.15603780

DOD INSTRUCTION 5000.02. (2020). *OPERATION OF THE ADAPTIVE ACQUISITION FRAMEWORK* . https://www.esd.whs.mil/Portals/54/Documents/DD/issuances/dodi/500002p.pdf?ver=2020-01-23-144114-093

Dvorsky, G. (2024, March 24). *NASA Ends $2 Billion Satellite Refueling Project After Contractor Accused of 'Poor Performance.'* Gizmodo. https://gizmodo.com/nasa-cancels-maxar-osam-1-mission-poor-performance-1851305230

Edwards, C. D., Fleischer, S. R., Austin, A., Barba, N. J., Bjornstad, P., Woolley, R. C., Hihn, J. M., Kolanjian, A., Saing, M., & Lock, R. (2022). Assessing Cost Drivers for Mars Small Spacecraft Missions. *2022 IEEE Aerospace Conference (AERO)*, 1–12. https://doi.org/10.1109/AERO53065.2022.9843293

Foreman, V. L., Jacqueline Le Moigne, & Olivier De Weck. (2016, September 13). A Survey of Cost Estimating Methodologies for Distributed Spacecraft Missions. *AIAA SPACE 2016*. https://doi.org/10.2514/6.2016-5245

Foust, J. (2023, March 23). *NASA warns of "devastating" impacts of potential budget cuts*. Space News. https://spacenews.com/nasa-warns-of-devastating-impacts-of-potential-budget-cuts/

FRED Economic Data. (2023a). *Consumer Price Index for All Urban Consumers: All Items Less Food and Energy in U.S. City Average (CPILFESL)*. ST. Louis FED. https://fred.stlouisfed.org/series/CPIAUCSL

FRED Economic Data. (2023b). *Personal Consumption Expenditures: Chain-type Price Index (PCEPI)*. St. Louis FED. https://fred.stlouisfed.org/series/PCEPI

GAO-20-195G. (2020). *Cost Estimating and Assessment Guide: Best Practices for Developing and Managing Program Costs*. https://www.gao.gov/assets/710/706933.pdf

GORTNEY, W. E. (2012). *JOINT CAPABILITIES INTEGRATION AND DEVELOPMENT SYSTEM, CJCSI 3170.01H*.

Hart, C. G., Z. He, R Sbragio, & N. Vlahopoulos. (2012). An advanced cost estimation methodology for engineering systems. *Systems Engineering*, *15*(1), 28–40. https://doi.org/10.1002/sys.20192

Hayes, S. & M. J. (2015). Mining for Cost Estimating Relations from Limited Complex Data. In *2015 ICEAA Workshop*. Victory Solutions MIPSS Team.

https://www.iceaaonline.com/wp-content/uploads/2015/06/GP08-Presentation-Jacobs-Mining-For-Relations.pdf

Ismail, H. (2014, May 4). *Project scheduling*. Https://Planningengineer.Net/Project-Scheduling/. https://planningengineer.net/project-scheduling/

Jonnalagadda, A. M. C. (2024). Optimizing Infrastructure Resources with Artificial Intelligence: A Technical Analysis. *International Journal For Multidisciplinary Research*. https://www.ijfmr.com/papers/2024/6/32063.pdf

Kim, K.-J., & Cho, S.-B. (2015). Meta-classifiers for high-dimensional, small sample classification for gene expression analysis. *Pattern Analysis and Applications*, *18*(3), 553–569. https://doi.org/10.1007/s10044-014-0369-7

Kwan, E. H.-A. H. R. L. (2005). Cost Modeling for low-cost planetary missions. *Sixth IAA International Conference on Low-Cost Planetary Missions (ICLCPM)*. https://dataverse.jpl.nasa.gov/file.xhtml?fileId=12145&version=2.0

Market, financial, and economic data. (2023). *Inflation-Adjusted S&P 500 by Year*. Multpl.Com. https://www.multpl.com/

Martin, P. K. (2012). *NASA'S CHALLENGES TO MEETING COST, SCHEDULE, AND PERFORMANCE GOALS*. https://oig.nasa.gov/wp-content/uploads/2024/02/IG-12-021.pdf

Nag, S., LeMoigne, J., & de Weck, O. (2014). Cost and risk analysis of small satellite constellations for earth observation. *2014 IEEE Aerospace Conference*, 1–16. https://doi.org/10.1109/AERO.2014.6836396

NASA. (2008). *NASA Cost Estimating Handbook (CEH)* (Fourth Edition). National Aeronautics and Space Administration (NASA). https://www3.nasa.gov/sites/default/files/files/01_CEH_Main_Body_02_27_15.pdf

NASA HQ. (2022). *NASA Space Flight Program and Project Management Handbook*. National Aeronautics and Space Administration. https://ntrs.nasa.gov/api/citations/20220009501/downloads/PM%20Handbook_June%202024.pdf

National Academies of Sciences, E. and M. (U. S.). C. on S. of S. (2015). The space science decadal surveys : lessons learned and best practices (Internet Archive). The National Academies Press.

National Research Council. (2010). *Controlling Cost Growth of NASA Earth and Space Science Missions*. National Academies Press. https://doi.org/10.17226/12946

Net, M. S., Selva, D., & Golkar, A. (2014). Exploring classification algorithms for early mission formulation cost estimation. *2014 IEEE Aerospace Conference*, 1–14. https://doi.org/10.1109/AERO.2014.6836326

OFFICE OF MANAGEMENT AND BUDGET. (2024). *CAPITAL PROGRAMMING GUIDE: Vol. A–11*. https://bidenwhitehouse.archives.gov/wp-content/uploads/2021/01/capital_programming_guide.pdf

Office of the Secretary of Defense. (2014). *Operating and Support Cost-Estimating Guide*. http://everyspec.com/DoD/DoD-PUBLICATIONS/OS_CE_GUIDE_V9_MAR2014_51821/

Pathak, D. K., Kalita, S. K., & Bhattacharya, D. K. (2022). Hyperspectral image classification using support vector machine: a spectral spatial feature based approach. *Evolutionary Intelligence*, *15*(3), 1809–1823. https://doi.org/10.1007/s12065-021-00591-0

Pirani, R., & Cale, J. (2022). Comparison of Pattern Recognition Approaches for Identification of Failure-prone Battery Cells. *2022 IEEE International Systems Conference (SysCon)*, 1–8. https://doi.org/10.1109/SysCon53536.2022.9773796

Salian, I. (2025). *SuperVize Me: What's the Difference Between Supervised, Unsupervised, Semi-Supervised, and Reinforcement Learning?* Edge Ai + Vision Alliance. https://www.edge-ai-vision.com/2018/08/supervize-me-whats-the-difference-between-supervised-unsupervised-semi-supervised-and-reinforcement-learning/

Sayama, Hiroki. (2015). *Introduction to the modeling and analysis of complex systems*. Published by Open SUNY Textbooks, Milne Library, State University of New York at Geneseo.

Servranckx, T., Mario Vanhoucke, & Tarik Aouam. (2021). Practical application of reference class forecasting for cost and time estimations: Identifying the properties of similarity. *European Journal of Operational Research*, *295*(3), 1161–1179. https://doi.org/10.1016/j.ejor.2021.03.063

Shafer, D. S. ., & Zhang, Zhiyi. (2019). *Introductory Statistics : A first course*. FlatWorld.

Shao, A., Koltz, E. A., & Wertz, J. R. (2014, August 4). Quantifying the Cost Reduction Potential for Earth Observation Satellites. *AIAA SPACE 2014 Conference and Exposition*. https://doi.org/10.2514/6.2014-4260

Shtub, A., & Ronen Versano. (1999). Estimating the cost of steel pipe bending: a comparison between neural networks and regression analysis. *International Journal of Production Economics*, *62*(3), 201–207. https://doi.org/10.1016/S0925-5273(98)00212-6

Tan, J. K. (2003). Health Care, Information Systems in. In *Encyclopedia of Information Systems* (pp. 519–536). Elsevier. https://doi.org/10.1016/B0-12-227240-4/00085-X

US Bureau of Labor Statistics. (2023). *Occupational Employment and Wage Statistics*. Www.Bls.Gov/OES. https://www.bls.gov/oes/current/naics4_336400.htm

US Debt Clock.org. (2025). *US Debt Clock*. Https://Www.Usdebtclock.Org/. https://www.usdebtclock.org/

Vapnik, V. N. , G. S. E. , & S. A. (1996). Support Vector Method for Function Approximation, Regression Estimation and Signal Processing. *Neural Information Processing Systems.* https://proceedings.neurips.cc/paper_files/paper/1996/file/4f284803bd0966cc24fa8683a34afc6e-Paper.pdf

Warne, R. T. . (2021). *Statistics for the social sciences : a general linear model approach*. Cambridge University Press.

Webb, R. J. M. H. S. & S. N. (2023). PCEC v2.3 CASTS Users Guide - Unrestricted.pdf. In *NASA Marshall Space Flight Center*. Engineering Cost Team (CS50). https://www.nasa.gov/wp-content/uploads/2025/08/pcec-v2-4-overview.pdf?emrc=8cac21

Wertz, J. Richard., Everett, D. F. ., & Puschell, J. John. (2018). *Space mission engineering : the new SMAD*. Microcosm Press.

Wicklin, R. (2015, September 10). *Plot the conditional distribution of the response in a linear regression model*. SAS Blog. https://blogs.sas.com/content/iml/2015/09/10/plot-distrib-reg-model.html

Appendix A

The forward and look-back inflation multipliers for each of the inflation indices are included in the following table for comparison. These indices were applied to the NASA budget accounting for each of the NIM missions to equate to equivalent 2022 pricing. An index of 1.00 was applied for years 2023 and 2024 for forward price mission costs to those year dollars during the model development.

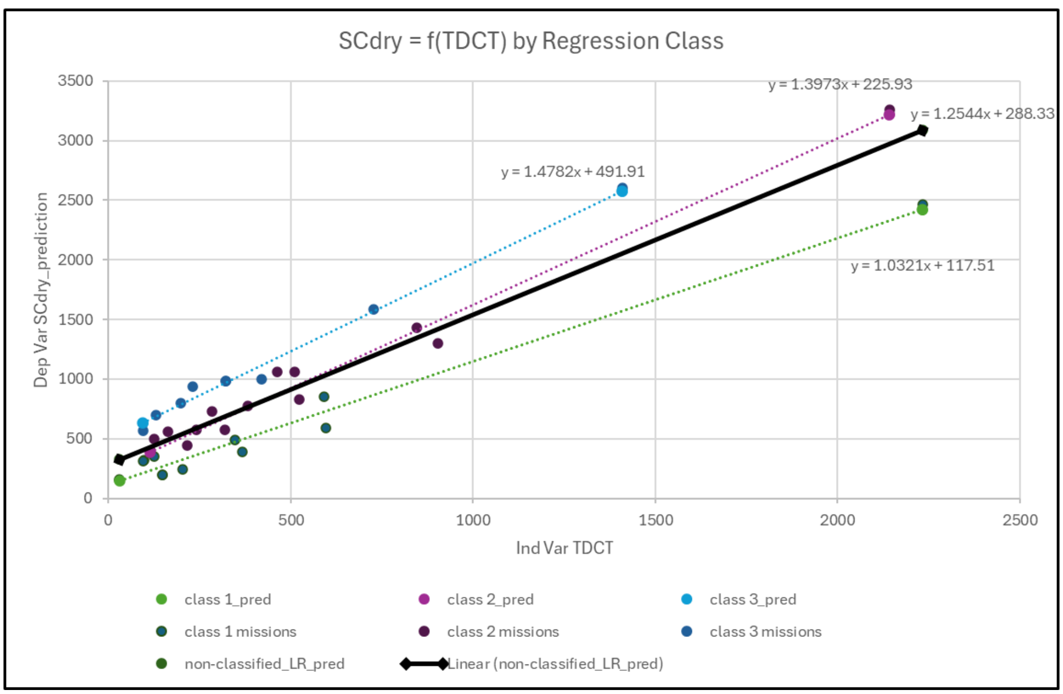
Year	NNSI Year	NSSI 2022	CPI	CPI Model 2022	PCEPI	PCEPI 2022	CPILFESL	CPILFESL 2022	S&P 500 /I	S&P 500 /I 2022
1959	1959	14.965	29.20	10.024	10.00	7.020	10.00	8.991	585.18	8.482
1960	1960	14.348	29.60	9.889	10.00	7.020	10.00	8.991	604.28	8.214
1961	1961	13.903	29.90	9.789	10.00	7.020	10.00	8.991	611.45	8.118
1962	1962	13.368	30.30	9.660	10.00	7.020	10.00	8.991	702.46	7.066
1963	1963	12.916	30.60	9.565	10.00	7.020	10.00	8.991	652.97	7.602
1964	1964	12.360	31.00	9.442	10.00	7.020	10.00	8.991	754.87	6.575
1965	1965	11.953	31.50	9.292	10.00	7.020	10.00	8.991	842.18	5.894
1966	1966	11.277	32.50	9.006	10.25	6.847	10.25	8.772	895.37	5.544
1967	1967	10.750	33.40	8.763	10.51	6.679	10.60	8.479	783.17	6.338
1968	1968	10.199	34.80	8.411	10.92	6.427	11.10	8.102	850.37	5.837
1969	1969	9.649	36.70	7.975	11.41	6.150	11.74	7.661	874.19	5.678
1970	1970	9.026	38.80	7.544	11.95	5.876	12.47	7.210	728.95	6.809
1971	1971	8.491	40.50	7.227	12.45	5.636	13.05	6.887	716.70	6.926
1972	1972	8.034	41.80	7.002	12.88	5.450	13.45	6.682	766.86	6.473
1973	1973	7.600	44.40	6.592	13.57	5.172	13.92	6.458	848.00	5.853
1974	1974	7.090	49.30	5.937	14.99	4.684	15.08	5.963	629.27	7.888
1975	1975	6.399	53.80	5.441	16.24	4.324	16.47	5.459	424.93	11.681
1976	1976	5.870	56.90	5.144	17.13	4.099	17.55	5.123	531.53	9.338
1977	1976 TQ	5.750	60.60	4.830	18.24	3.849	18.65	4.822	541.37	9.169
1978	1977	5.299	65.20	4.489	19.51	3.598	20.00	4.495	440.58	11.266
1979	1978	4.916	72.60	4.032	21.24	3.305	21.95	4.096	445.42	11.144
1980	1979	4.489	82.40	3.552	23.53	2.983	24.68	3.643	434.92	11.413
1981	1980	4.055	90.90	3.220	25.64	2.738	27.27	3.297	466.43	10.642
1982	1981	3.683	96.50	3.033	27.06	2.594	29.28	3.070	379.53	13.078
1983	1982	3.417	99.60	2.939	28.21	2.488	30.43	2.955	450.18	11.026
1984	1983	3.211	103.90	2.817	29.28	2.398	31.99	2.811	498.23	9.963
1985	1984	3.047	107.60	2.720	30.30	2.317	33.39	2.693	496.27	10.002
1986	1985	2.947	109.60	2.671	30.96	2.267	34.74	2.588	579.60	8.564
1987	1986	2.861	113.60	2.577	31.91	2.200	36.10	2.490	725.73	6.839
1988	1987	2.748	118.30	2.474	33.16	2.117	37.71	2.384	660.59	7.514
1989	1988	2.610	124.00	2.360	34.61	2.028	39.40	2.282	719.06	6.903
1990	1989	2.490	130.70	2.239	36.13	1.943	41.38	2.173	814.19	6.096
1991	1990	2.383	136.20	2.149	37.34	1.880	43.42	2.071	737.82	6.727
1992	1991	2.300	140.30	2.086	38.33	1.831	45.01	1.998	919.26	5.400
1993	1992	2.187	144.50	2.026	39.29	1.787	46.50	1.934	931.22	5.330
1994	1993	2.098	148.20	1.975	40.11	1.750	47.81	1.881	987.10	5.028
1995	1994	2.035	152.40	1.921	40.95	1.714	49.25	1.825	944.46	5.256
1996	1995	1.984	156.90	1.866	41.83	1.678	50.58	1.778	1214.15	4.088
1997	1996	1.935	160.50	1.824	42.55	1.650	51.78	1.736	1469.39	3.378
1998	1997	1.909	163.00	1.796	42.89	1.637	52.97	1.697	1818.87	2.729
1999	1998	1.862	166.60	1.757	43.52	1.613	54.07	1.663	2319.00	2.140
2000	1999	1.818	172.20	1.700	44.62	1.573	55.38	1.623	2576.78	1.926
2001	2000	1.747	177.10	1.653	45.51	1.542	56.86	1.581	2327.31	2.133
2002	2001	1.686	179.90	1.627	46.11	1.522	58.18	1.545	1964.36	2.527
2003	2002	1.640	184.00	1.591	47.08	1.491	59.03	1.523	1504.29	3.300
2004	2003	1.605	188.90	1.549	48.25	1.455	60.07	1.497	1865.78	2.660
2005	2004	1.552	195.30	1.499	49.64	1.414	61.36	1.465	1890.19	2.626
2006	2005	1.505	201.60	1.452	51.04	1.375	62.91	1.429	1967.48	2.523
2007	2006	1.459	207.30	1.412	52.34	1.341	64.38	1.397	2146.69	2.312
2008	2007	1.405	215.30	1.359	53.89	1.303	65.85	1.365	1992.95	2.491
2009	2008	1.357	214.50	1.365	53.74	1.306	66.98	1.342	1250.79	3.968
2010	2009	1.332	218.10	1.342	54.71	1.283	67.62	1.330	1582.07	3.137
2011	2010	1.314	224.90	1.301	56.09	1.252	68.74	1.308	1777.01	2.793
2012	2011	1.293	229.60	1.275	57.14	1.229	70.19	1.281	1750.68	2.835
2013	2012	1.279	233.00	1.256	57.91	1.212	71.43	1.259	1961.45	2.531
2014	2013	1.260	236.70	1.237	58.79	1.194	72.68	1.237	2377.00	2.088
2015	2014	1.236	237.00	1.235	58.92	1.191	74.01	1.215	2647.83	1.875
2016	2015	1.211	240.00	1.220	59.51	1.180	75.64	1.189	2470.84	2.009
2017	2016	1.197	245.10	1.194	60.59	1.158	77.03	1.167	2858.52	1.736
2018	2017	1.172	251.10	1.166	61.89	1.134	78.68	1.143	3434.07	1.445
2019	2018	1.143	255.70	1.145	62.81	1.118	80.41	1.118	3160.51	1.571
2020	2019	1.121	258.80	1.131	63.50	1.105	81.78	1.099	3877.22	1.280
2021	2020	1.097	271.00	1.080	66.06	1.063	84.70	1.061	4425.02	1.122
2022	2021	1.057	292.70	1.000	70.20	1.000	89.91	1.000	4963.62	1.000

Appendix B

SCdry = f(TDCT)

Zero class limit = 0.00

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
SCdry = f(TDCT)				
Class Regression Threshold			16%	
Class Residual Threshold			7%	
	Regression n	Residual n	Min %e	Max %e
Class 1	12	12	-7.0%	16.4%
Class 2	13	14	-8.3%	12.7%
Class 3	10	9	-9.8%	9.1%
Model	LR	CR	CR w/corr	
mean	17.5%	-1.6%	0.5%	
|mean|	36.2%	13.5%	3.8%	
st.dev	50.4%	18.1%	5.3%	
st.dev | |	38.7%	11.9%	3.6%	
co.var	34.7%	-8.8%	9.81	
co.var| |	93.5%	113.8%	1.39	
|mean| + st.dev | |	74.9%	25.4%	7.4%	



				CR Modeler [SCdry = f(TDCT)]			
Data ID	Mission	Ind Var TDCT	Dep Var SCdry	reg class	res class	Predict	Pred %e
10	Grail-A	147.95	198	1	1	197.1	-0.5%
11	Grail-B	147.95	198	1	1	197.1	-0.5%
14	Ladee	204.3	240.7	1	1	247.0	2.6%
16	Lunar Prospector	30.6	158.7	1	2	161.1	1.5%
1	2001 Mars Odyssey	366.2	376.3	1	1	390.4	3.8%
12	Insight	596.4	608	1	1	594.4	-2.2%
35	Stardust	126.4	300	1	3	349.1	16.4%
31	New Horizons	347.7	508	1	2	492.8	-3.0%
17	M2020 - Perseverance	2232.2	2439	1	2	2464.2	1.0%
3	Contour	96.8	328	1	3	317.0	-3.4%
19	Mars Climate Orbiter	94.5	338	1	3	314.5	-7.0%
32	Osiris-Rex	591.6	860	1	3	853.6	-0.7%
7	Deep Space 1	114.2	373.7	2	2	398.7	6.7%
28	Messenger	216.2	485.2	2	1	444.9	-8.3%
33	Phoenix	318.2	597	2	1	572.5	-4.1%
15	Lucy	523.1	821	2	1	828.9	1.0%
8	Galileo	902.3	1289	2	1	1303.3	1.1%
4	Dart	240.6	560	2	2	577.1	3.1%
9	Genesis	163	494	2	3	556.7	12.7%
34	Psyche	844.5	1400	2	2	1429.3	2.1%
25	MAVEN	382.4	809	2	2	777.2	-3.9%
30	NEAR Shoemaker	124.9	487	2	3	501.5	3.0%
21	Mars Observer	511.2	1028	2	3	1061.5	3.3%
29	MSL - Curiosity	2142.4	3300	2	2	3261.0	-1.2%
5	Dawn	282.9	747.1	2	3	730.5	-2.2%
18	Magellan	463.2	1035	3	1	1057.4	2.2%
2	Cassini	1408.5	2581	3	2	2605.3	0.9%
24	Mars Recon Orbiter	419.2	1031	3	1	998.8	-3.1%
23	Mars Polar Lander	94.5	519	3	1	566.3	9.1%
13	Juno	726.2	1593	3	2	1587.2	-0.4%
26	MERA - Spirit	321.5	1018	3	2	983.3	-3.4%
27	MERB - Opportunity	321.5	1018	3	2	983.3	-3.4%
20	Mars Global Surveyor	130.7	666.3	3	2	698.5	4.8%
22	Mars Pathfinder	199.2	795	3	2	800.8	0.7%
6	Deep Impact	230.9	1042	3	3	939.8	-9.8%

Interpretable (class-wise) CR Model

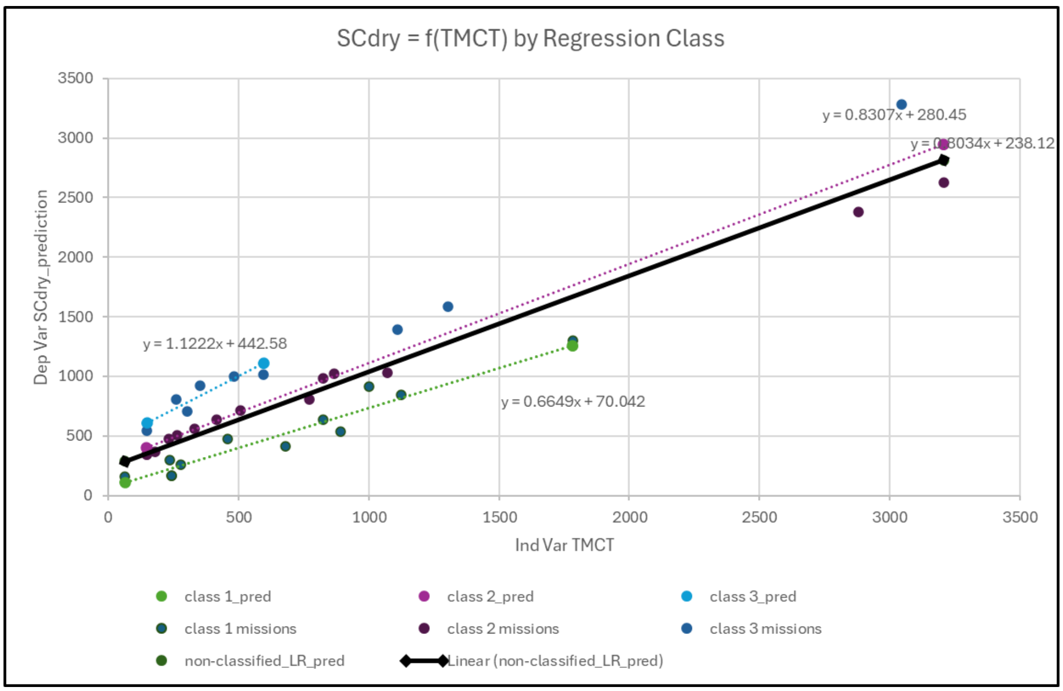
Class	n	SCdry = f(TDCT)	TDCT(min)	TDCT(max)	mean %e	stdev %e
1	12	SCdry(c1) = 117.51 + 1.03 * TDCT	30.6	2232.2	3.54%	4.43%
2	13	SCdry(c2) = 225.93 + 1.40 * TDCT	114.2	2142.40	4.04%	3.37%
3	10	SCdry(c3) = 491.91 + 1.48 * TDCT	94.5	1408.50	3.79%	3.30%
				Model	3.80%	3.65%

Class	n	Corrector = f(TDCT)	Correction = - Corrector
1	12	Corrector(c1) = 51.50 + 0.15 * TDCT	Corrector(c1) = -51.50 + -0.15 * TDCT
2	14	Corrector(c1) = -11.61 + -0.01 * TDCT	Corrector(c1) = 11.61 + 0.01 * TDCT
3	9	Corrector(c1) = -94.46 + -0.05 * TDCT	Corrector(c1) = 94.46 + 0.05 * TDCT

SCdry = f(TMCT)

Zero class limit = 0.00

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
SCdry = f(TMCT)				
Class Regression Threshold			20%	
Class Residual Threshold			7%	
	Regression n	Residual n	Min %e	Max %e
Class 1	12	14	-17.7%	11.1%
Class 2	17	11	-4.6%	6.6%
Class 3	6	9	-11.3%	6.3%
Model	LR	CR	CR w/corr	
mean	14.5%	-2.3%	-0.1%	
|mean|	34.5%	13.1%	4.1%	
st.dev	46.6%	15.9%	6.2%	
st.dev | |	34.1%	9.2%	4.6%	
co.var	31.0%	-14.6%	-124.73	
co.var| |	100.9%	142.6%	1.52	
|mean| + st.dev | |	68.6%	22.2%	8.8%	



				CR Modeler [SCdry = f(TMCT)]			
Data ID	Mission	Ind Var TMCT	Dep Var SCdry	reg class	res class	Predict	Pred %e
10	Grail-A	243.65	198	1	1	163.0	-17.7%
11	Grail-B	243.65	198	1	1	163.0	-17.7%
1	2001 Mars Odyssey	680.2	376.3	1	1	416.1	10.6%
14	Ladee	279.2	240.7	1	2	261.3	8.5%
31	New Horizons	889.7	508	1	1	537.6	5.8%
16	Lunar Prospector	63.9	158.7	1	3	157.8	-0.6%
12	Insight	822.8	608	1	2	638.2	5.0%
35	Stardust	236.4	300	1	3	297.0	-1.0%
32	Osiris-Rex	1125.3	860	1	2	847.9	-1.4%
8	Galileo	1783.1	1289	1	2	1304.0	1.2%
15	Lucy	999.2	821	1	3	912.5	11.1%
28	Messenger	457.8	485.2	1	3	475.6	-2.0%
2	Cassini	3206.8	2581	2	1	2623.2	1.6%
3	Contour	145.8	328	2	1	340.9	3.9%
24	Mars Recon Orbiter	1072.2	1031	2	1	1031.6	0.1%
25	MAVEN	771.2	809	2	1	807.2	-0.2%
19	Mars Climate Orbiter	147.95	338	2	1	342.5	1.3%
17	M2020 - Perseverance	2880.6	2439	2	1	2380.0	-2.4%
7	Deep Space 1	180.6	373.7	2	1	366.8	-1.8%
33	Phoenix	417	597	2	2	636.4	6.6%
9	Genesis	264	494	2	2	504.9	2.2%
4	Dart	332	560	2	0	556.2	-0.7%
18	Magellan	864.8	1035	2	2	1021.1	-1.3%
21	Mars Observer	825.5	1028	2	2	987.3	-4.0%
30	NEAR Shoemaker	230.8	487	2	2	476.4	-2.2%
5	Dawn	505.6	747.1	2	2	712.5	-4.6%
29	MSL - Curiosity	3046	3300	2	3	3279.4	-0.6%
13	Juno	1304.5	1593	2	3	1585.5	-0.5%
34	Psyche	1108.1	1400	2	3	1394.5	-0.4%
20	Mars Global Surveyor	303	666.3	3	1	708.6	6.3%
27	MERB - Opportunity	595.6	1018	3	1	1012.0	-0.6%
23	Mars Polar Lander	147.95	519	3	1	547.7	5.5%
26	MERA - Spirit	483.3	1018	3	2	996.4	-2.1%
22	Mars Pathfinder	258.8	795	3	3	806.0	1.4%
6	Deep Impact	352.5	1042	3	3	924.4	-11.3%

Interpretable (class-wise) CR Model

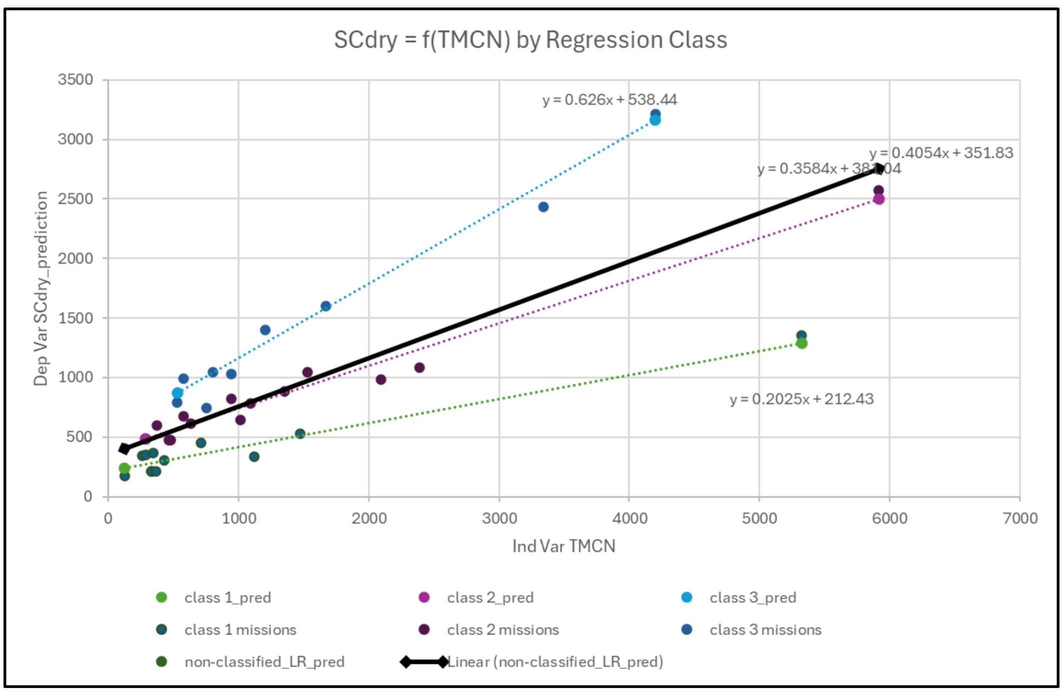
Class	n	SCdry = f(TMCT)	TMCT(min)	TMCT(max)	mean %e	stdev %e
1	12	SCdry(c1) = 70.04 + 0.66 * TMCT	63.9	1783.1	6.87%	6.28%
2	17	SCdry(c2) = 280.45 + 0.83 * TMCT	145.8	3206.80	2.03%	1.81%
3	6	SCdry(c3) = 442.58 + 1.12 * TMCT	147.95	595.60	4.54%	4.03%
				Model	4.12%	4.64%

Class	n	Corrector = f(TMCT)	Correction = - Corrector
1	14	Corrector(c1) = 48.29 + 0.09 * TMCT	Corrector(c1) = -48.29 + -0.09 * TMCT
2	10	Corrector(c1) = 2.35 + -0.03 * TMCT	Corrector(c1) = -2.35 + 0.03 * TMCT
3	9	Corrector(c1) = -36.22 + -0.14 * TMCT	Corrector(c1) = 36.22 + 0.14 * TMCT

SCdry = f(TMCN)

Zero class limit = 0.00

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
SCdry = f(TMCN)				
Class Regression Threshold			28%	
Class Residual Threshold			7%	
	Regression n	Residual n	Min %e	Max %e
Class 1	12	13	-10.5%	12.2%
Class 2	15	11	-6.4%	6.7%
Class 3	8	11	-5.1%	2.9%
Model	LR	CR	CR w/corr	
mean	23.6%	-2.6%	0.4%	
|mean|	43.0%	13.2%	3.9%	
st.dev	56.9%	17.8%	5.0%	
st.dev | |	43.7%	12.0%	3.1%	
co.var	41.5%	-14.4%	12.35	
co.var| |	98.5%	109.8%	1.28	
|mean| + st.dev | |	86.7%	25.2%	6.9%	



				CR Modeler [SCdry = f(TMCN)]			
Data ID	Mission	Ind Var TMCN	Dep Var SCdry	reg class	res class	Predict	Pred %e
16	Lunar Prospector	125.2	158.7	1	1	178.1	12.2%
10	Grail-A	327.15	198	1	1	210.4	6.3%
11	Grail-B	327.15	198	1	1	210.4	6.3%
1	2001 Mars Odyssey	1116.2	376.3	1	1	336.7	-10.5%
14	Ladee	365.8	240.7	1	1	216.6	-10.0%
8	Galileo	5323.7	1289	1	2	1353.9	5.0%
31	New Horizons	1472.6	508	1	2	527.2	3.8%
35	Stardust	427.6	300	1	2	302.9	1.0%
3	Contour	258.1	328	1	3	343.5	4.7%
19	Mars Climate Orbiter	285.6	338	1	3	349.9	3.5%
28	Messenger	710.3	485.2	1	3	449.0	-7.5%
7	Deep Space 1	346.6	373.7	1	3	364.1	-2.6%
18	Magellan	2390.8	1035	2	1	1082.0	4.5%
12	Insight	1015.1	608	2	1	647.4	6.5%
21	Mars Observer	2092.7	1028	2	1	987.9	-3.9%
30	NEAR Shoemaker	462.8	487	2	1	472.9	-2.9%
9	Genesis	475.2	494	2	1	476.8	-3.5%
2	Cassini	5916.8	2581	2	2	2572.5	-0.3%
32	Osiris-Rex	1350.7	860	2	2	880.3	2.4%
33	Phoenix	630.1	597	2	2	613.2	2.7%
15	Lucy	1089.6	821	2	2	783.5	-4.6%
24	Mars Recon Orbiter	1531.4	1031	2	3	1048.0	1.6%
25	MAVEN	942.5	809	2	3	818.7	1.2%
23	Mars Polar Lander	285.6	519	2	2	485.6	-6.4%
4	Dart	374.8	560	2	3	597.7	6.7%
20	Mars Global Surveyor	579.6	666.3	2	3	677.5	1.7%
5	Dawn	749.7	747.1	2	3	743.7	-0.5%
27	MERB - Opportunity	940.1	1018	3	1	1032.7	1.4%
22	Mars Pathfinder	527.6	795	3	1	791.9	-0.4%
17	M2020 - Perseverance	3338.1	2439	3	1	2432.0	-0.3%
26	MERA - Spirit	800.5	1018	3	2	1048.0	2.9%
13	Juno	1669.8	1593	3	2	1602.8	0.6%
29	MSL - Curiosity	4197.7	3300	3	2	3216.1	-2.5%
34	Psyche	1202.7	1400	3	3	1399.2	-0.1%
6	Deep Impact	578.3	1042	3	3	989.1	-5.1%

Interpretable (class-wise) CR Model

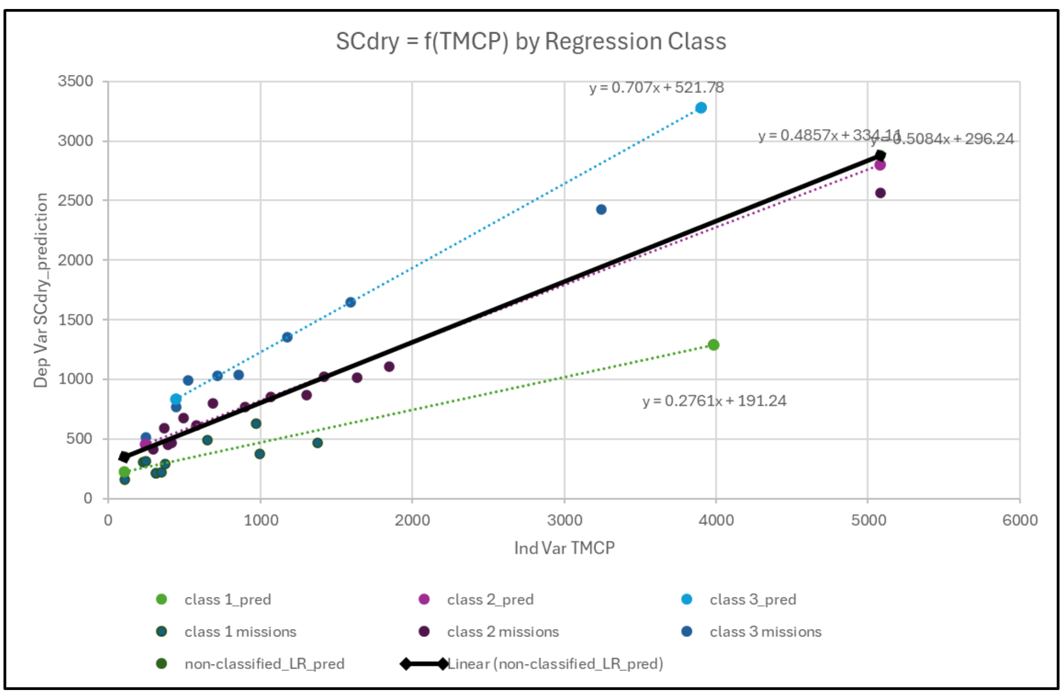
Class	n	SCdry = f(TMCN)	TMCN(min)	TMCN(max)	mean %e	stdev %e
1	12	SCdry(c1) = 212.43 + 0.20 * TMCN	125.2	5323.7	6.11%	3.42%
2	15	SCdry(c2) = 381.04 + 0.36 * TMCN	285.6	5916.80	3.30%	2.13%
3	8	SCdry(c3) = 538.44 + 0.63 * TMCN	527.6	4197.70	1.67%	1.75%
				Model	3.89%	3.05%

Class	n	Corrector = f(TMCN)	Correction = - Corrector
1	13	Corrector(c1) = 54.38 + 0.04 * TMCN	Corrector(c1) = -54.38 + -0.04 * TMCN
2	11	Corrector(c1) = 1.32 + -0.01 * TMCN	Corrector(c1) = -1.32 + 0.01 * TMCN
3	11	Corrector(c1) = -70.81 + -0.03 * TMCN	Corrector(c1) = 70.81 + 0.03 * TMCN

SCdry = f(TMCP)

Zero class limit = 0.00

Model Summary (35 missions)					
Classified Linear Reg with Classified Residual Reg					
SCdry = f(TMCP)					
	Class Regression Threshold			21%	
	Class Residual Threshold			5%	
		Regression n	Residual n	Min %e	Max %e
	Class 1	12	15	-8.4%	6.9%
	Class 2	16	10	-6.4%	10.5%
	Class 3	7	10	-4.9%	3.5%
	Model	LR	CR	CR w/corr	
	mean	19.4%	-3.1%	0.0%	
	|mean|	38.6%	13.8%	3.7%	
	st.dev	51.7%	17.8%	4.7%	
	st.dev | |	39.1%	11.4%	2.8%	
	co.var	37.6%	-17.5%	-2503.25	
	co.var| |	98.7%	121.2%	1.26	
|mean| + st.dev | |		77.7%	25.2%	6.5%	



Data ID	Mission	Ind Var TMCP	Dep Var SCdry	CR Modeler [SCdry = f(TMCP)] reg class	res class	Predict	Pred %e
10	Grail-A	312.25	198	1	1	211.8	6.9%
11	Grail-B	312.25	198	1	1	211.8	6.9%
16	Lunar Prospector	106.3	158.7	1	1	162.2	2.2%
1	2001 Mars Odyssey	994.4	376.3	1	1	376.0	-0.1%
14	Ladee	348.7	240.7	1	1	220.5	-8.4%
31	New Horizons	1375.3	508	1	1	467.8	-7.9%
8	Galileo	3983.5	1289	1	2	1295.6	0.5%
35	Stardust	371.8	300	1	2	290.9	-3.0%
12	Insight	973	608	1	3	629.8	3.6%
28	Messenger	649.6	485.2	1	3	490.8	1.2%
3	Contour	226.7	328	1	3	309.1	-5.8%
19	Mars Climate Orbiter	243.85	338	1	3	316.5	-6.4%
7	Deep Space 1	296.6	373.7	2	1	413.0	10.5%
18	Magellan	1845.6	1035	2	1	1110.7	7.3%
2	Cassini	5079.8	2581	2	1	2567.4	-0.5%
32	Osiris-Rex	1302.5	860	2	1	866.1	0.7%
21	Mars Observer	1636	1028	2	1	1016.3	-1.1%
9	Genesis	414.4	494	2	1	466.1	-5.7%
15	Lucy	1069.5	821	2	2	852.0	3.8%
30	NEAR Shoemaker	391.2	487	2	1	455.6	-6.4%
24	Mars Recon Orbiter	1421.6	1031	2	2	1023.8	-0.7%
33	Phoenix	576.4	597	2	2	611.5	2.4%
25	MAVEN	901.2	809	2	2	769.9	-4.8%
5	Dawn	690.7	747.1	2	3	796.1	6.6%
4	Dart	365.2	560	2	3	588.0	5.0%
20	Mars Global Surveyor	496.6	666.3	2	3	672.0	0.9%
23	Mars Polar Lander	243.85	519	2	3	510.5	-1.6%
17	M2020 - Perseverance	3244.5	2439	2	3	2428.6	-0.4%
27	MERB - Opportunity	854.6	1018	3	1	1041.1	2.3%
13	Juno	1594	1593	3	2	1648.2	3.5%
29	MSL - Curiosity	3902.8	3300	3	2	3285.1	-0.5%
22	Mars Pathfinder	443.6	795	3	1	765.1	-3.8%
26	MERA - Spirit	720.7	1018	3	2	1029.0	1.1%
34	Psyche	1176.1	1400	3	2	1351.9	-3.4%
6	Deep Impact	521.6	1042	3	3	991.1	-4.9%

Interpretable (class-wise) CR Model

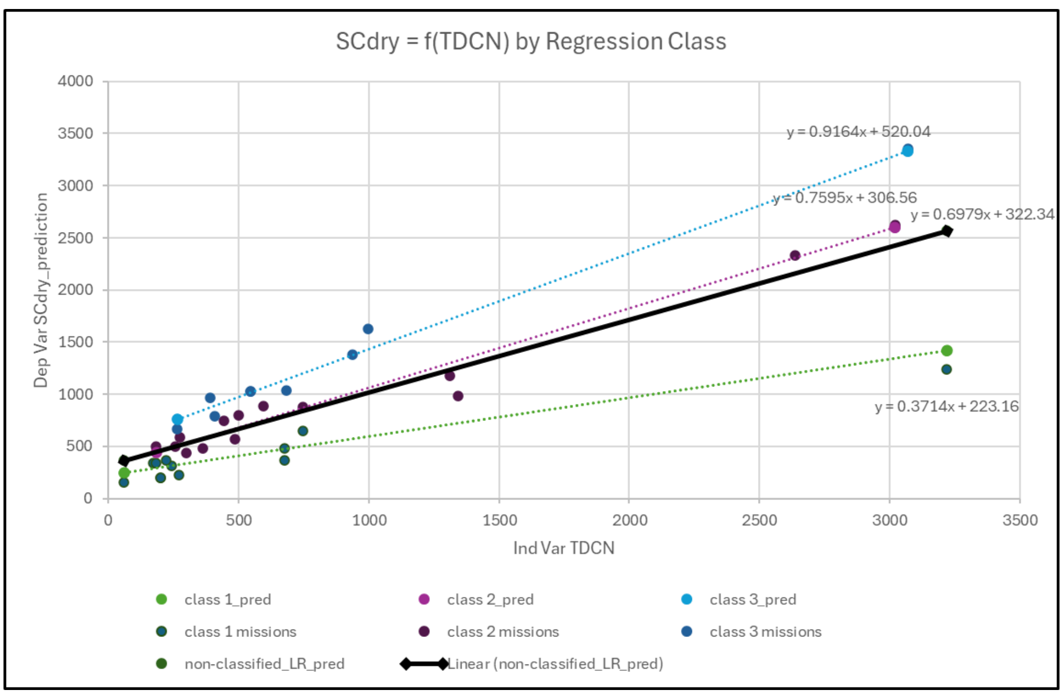
Class	n	SCdry = f(TMCP)	TMCP(min)	TMCP(max)	mean %e	stdev %e
1	12	SCdry(c1) = 191.24 + 0.28 * TMCP	106.3	3983.5	4.40%	3.00%
2	16	SCdry(c2) = 334.11 + 0.49 * TMCP	243.85	5079.80	3.66%	3.07%
3	7	SCdry(c3) = 521.78 + 0.71 * TMCP	443.6	3902.80	2.77%	1.57%
				Model	3.74%	2.80%

Class	n	Corrector = f(TMCP)	Correction = - Corrector
1	15	Corrector(c1) = 54.68 + 0.04 * TMCP	Corrector(c1) = -54.68 + -0.04 * TMCP
2	10	Corrector(c1) = 3.71 + 0.00 * TMCP	Corrector(c1) = -3.71 + 0.00 * TMCP
3	10	Corrector(c1) = -20.47 + -0.15 * TMCP	Corrector(c1) = 20.47 + 0.15 * TMCP

SCdry = f(TDCN)

Zero class limit = 0.00

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
SCdry = f(TDCN)				
Class Regression Threshold			21%	
Class Residual Threshold			7%	
	Regression n	Residual n	Min %e	Max %e
Class 1	13	13	-5.9%	7.4%
Class 2	13	10	-11.5%	14.4%
Class 3	9	12	-7.3%	2.3%
Model	LR	CR	CR w/corr	
mean	20.2%	-4.4%	-0.1%	
|mean|	40.2%	15.3%	3.5%	
st.dev	53.0%	20.5%	4.8%	
st.dev | |	39.6%	14.1%	3.2%	
co.var	38.1%	-21.3%	-31.96	
co.var| |	101.4%	107.8%	1.35	
|mean| + st.dev | |	79.8%	29.4%	6.7%	



				CR Modeler [SCdry = f(TDCN)]			
Data ID	Mission	Ind Var TDCN	Dep Var SCdry	reg class	res class	Predict	Pred %e
10	Grail-A	200.4	198	1	1	202.9	2.5%
11	Grail-B	200.4	198	1	1	202.9	2.5%
16	Lunar Prospector	60.2	158.7	1	1	154.7	-2.5%
14	Ladee	269.8	240.7	1	1	226.7	-5.8%
1	2001 Mars Odyssey	677	376.3	1	1	366.7	-2.6%
8	Galileo	3218.2	1289	1	1	1240.0	-3.8%
35	Stardust	244	300	1	2	313.6	4.5%
31	New Horizons	676.2	508	1	2	477.8	-5.9%
12	Insight	745.8	608	1	3	652.7	7.4%
3	Contour	172	328	1	3	336.3	2.5%
19	Mars Climate Orbiter	183.25	338	1	3	342.5	1.3%
7	Deep Space 1	221.2	373.7	1	3	363.4	-2.8%
18	Magellan	1343.8	1035	1	3	982.5	-5.1%
21	Mars Observer	1309.3	1028	2	1	1175.5	14.4%
28	Messenger	363.3	485.2	2	1	483.2	-0.4%
33	Phoenix	484.8	597	2	1	572.2	-4.2%
9	Genesis	300.5	494	2	1	437.3	-11.5%
30	NEAR Shoemaker	257.9	487	2	2	502.3	3.1%
32	Osiris-Rex	747.5	860	2	2	878.4	2.1%
2	Cassini	3019.5	2581	2	2	2623.5	1.6%
4	Dart	274.8	560	2	3	583.0	4.1%
15	Lucy	594.1	821	2	3	883.0	7.6%
17	M2020 - Perseverance	2635.5	2439	2	2	2328.6	-4.5%
23	Mars Polar Lander	183.25	519	2	3	497.0	-4.2%
5	Dawn	444.7	747.1	2	3	742.6	-0.6%
25	MAVEN	499.5	809	2	3	794.1	-1.8%
24	Mars Recon Orbiter	681.6	1031	3	1	1036.5	0.5%
22	Mars Pathfinder	407.8	795	3	1	793.2	-0.2%
20	Mars Global Surveyor	263.9	666.3	3	1	665.4	-0.1%
29	MSL - Curiosity	3068.5	3300	3	2	3355.8	1.7%
34	Psyche	935.9	1400	3	2	1383.4	-1.2%
26	MERA - Spirit	547.2	1018	3	2	1023.8	0.6%
27	MERB - Opportunity	547.2	1018	3	2	1023.8	0.6%
13	Juno	995.9	1593	3	3	1630.2	2.3%
6	Deep Impact	390.5	1042	3	3	966.4	-7.3%

Interpretable (class-wise) CR Model

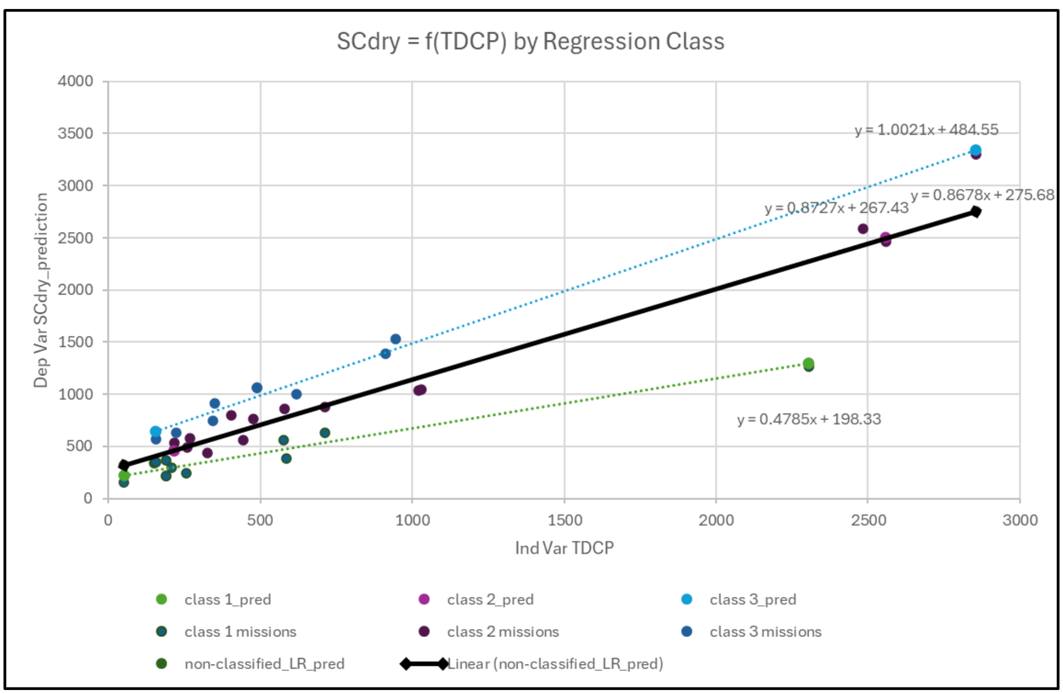
Class	n	SCdry = f(TDCN)	TDCN(min)	TDCN(max)	mean %e	stdev %e
1	13	SCdry(c1) = 223.16 + 0.37 * TDCN	60.2	3218.2	3.78%	1.80%
2	13	SCdry(c2) = 306.56 + 0.76 * TDCN	183.25	3019.50	4.63%	4.18%
3	9	SCdry(c3) = 520.04 + 0.92 * TDCN	263.9	3068.50	1.61%	2.23%
				Model	3.54%	3.15%

Class	n	Corrector = f(TDCN)	Correction = - Corrector
1	13	Corrector(c1) = 89.17 + 0.03 * TDCN	Corrector(c1) = -89.17 + -0.03 * TDCN
2	10	Corrector(c1) = 2.32 + -0.01 * TDCN	Corrector(c1) = -2.32 + 0.01 * TDCN
3	12	Corrector(c1) = -18.26 + -0.18 * TDCN	Corrector(c1) = 18.26 + 0.18 * TDCN

SCdry = f(TDCP)

Zero class limit = 0.00

Model Summary (35 missions)					
Classified Linear Reg with Classified Residual Reg					
SCdry = f(TDCP)					
	Class Regression Threshold			15%	
	Class Residual Threshold			4%	
		Regression n	Residual n	Min %e	Max %e
	Class 1	13	13	-10.6%	10.6%
	Class 2	11	7	-5.6%	9.2%
	Class 3	11	15	-12.2%	9.0%
	Model	LR	CR	CR w/corr	
	mean	16.6%	-3.1%	0.7%	
	|mean|	35.7%	13.8%	4.2%	
	st.dev	48.1%	18.8%	5.4%	
	st.dev | |	35.9%	12.9%	3.4%	
	co.var	34.4%	-16.6%	8.27	
	co.var| |	99.3%	107.6%	1.28	
|mean| + st.dev | |		71.6%	26.7%	7.6%	



				CR Modeler [SCdry = f(TDCP)]			
Data ID	Mission	Ind Var TDCP	Dep Var SCdry	reg class	res class	Predict	Pred %e
10	Grail-A	190.85	198	1	1	212.7	7.4%
11	Grail-B	190.85	198	1	1	212.7	7.4%
1	2001 Mars Odyssey	586.3	376.3	1	1	380.9	1.2%
14	Ladee	257.6	240.7	1	1	241.1	0.2%
16	Lunar Prospector	51	158.7	1	1	153.2	-3.5%
8	Galileo	2303.7	1289	1	2	1265.2	-1.8%
31	New Horizons	576.3	508	1	3	561.7	10.6%
35	Stardust	208.3	300	1	2	297.4	-0.9%
12	Insight	712.9	608	1	3	631.5	3.9%
3	Contour	150.8	328	1	3	344.0	4.9%
19	Mars Climate Orbiter	156.2	338	1	3	346.8	2.6%
7	Deep Space 1	188.7	373.7	1	3	363.4	-2.8%
28	Messenger	326.4	485.2	1	3	433.8	-10.6%
21	Mars Observer	1021.2	1028	2	1	1037.6	0.9%
18	Magellan	1030.2	1035	2	1	1045.0	1.0%
33	Phoenix	442.8	597	2	1	563.5	-5.6%
32	Osiris-Rex	712.2	860	2	2	880.0	2.3%
17	M2020 - Perseverance	2558.7	2439	2	2	2460.6	0.9%
9	Genesis	260	494	2	2	492.9	-0.2%
30	NEAR Shoemaker	216.2	487	2	3	531.8	9.2%
15	Lucy	579.2	821	2	3	860.6	4.8%
2	Cassini	2484.1	2581	2	3	2585.9	0.2%
4	Dart	267.9	560	2	3	578.6	3.3%
25	MAVEN	475.6	809	2	3	766.7	-5.2%
5	Dawn	404.6	747.1	3	1	801.7	7.3%
29	MSL - Curiosity	2853.8	3300	3	2	3299.5	0.0%
23	Mars Polar Lander	156.2	519	3	1	565.9	9.0%
24	Mars Recon Orbiter	618.1	1031	3	1	1004.3	-2.6%
34	Psyche	912.5	1400	3	2	1386.6	-1.0%
22	Mars Pathfinder	342.5	795	3	1	742.7	-6.6%
20	Mars Global Surveyor	222.5	666.3	3	1	628.9	-5.6%
26	MERA - Spirit	489.9	1018	3	3	1060.2	4.1%
27	MERB - Opportunity	489.9	1018	3	3	1060.2	4.1%
13	Juno	943.9	1593	3	3	1530.1	-3.9%
6	Deep Impact	349.4	1042	3	3	914.7	-12.2%

Interpretable (class-wise) CR Model

Class	n	SCdry = f(TDCP)	TDCP(min)	TDCP(max)	mean %e	stdev %e
1	13	SCdry(c1) = 198.33 + 0.48 * TDCP	51	2303.7	4.44%	3.52%
2	11	SCdry(c2) = 267.43 + 0.87 * TDCP	216.2	2558.70	3.06%	2.87%
3	11	SCdry(c3) = 484.55 + 1.00 * TDCP	156.2	2853.80	5.14%	3.55%
				Model	4.23%	3.35%

Class	n	Corrector = f(TDCP)	Correction = - Corrector
1	13	Corrector(c1) = 66.83 + 0.05 * TDCP	Corrector(c1) = -66.83 + -0.05 * TDCP
2	7	Corrector(c1) = -2.85 + 0.02 * TDCP	Corrector(c1) = 2.85 + -0.02 * TDCP
3	15	Corrector(c1) = -68.52 + -0.03 * TDCP	Corrector(c1) = 68.52 + 0.03 * TDCP

Appendix C

Tmct = f(scdry)

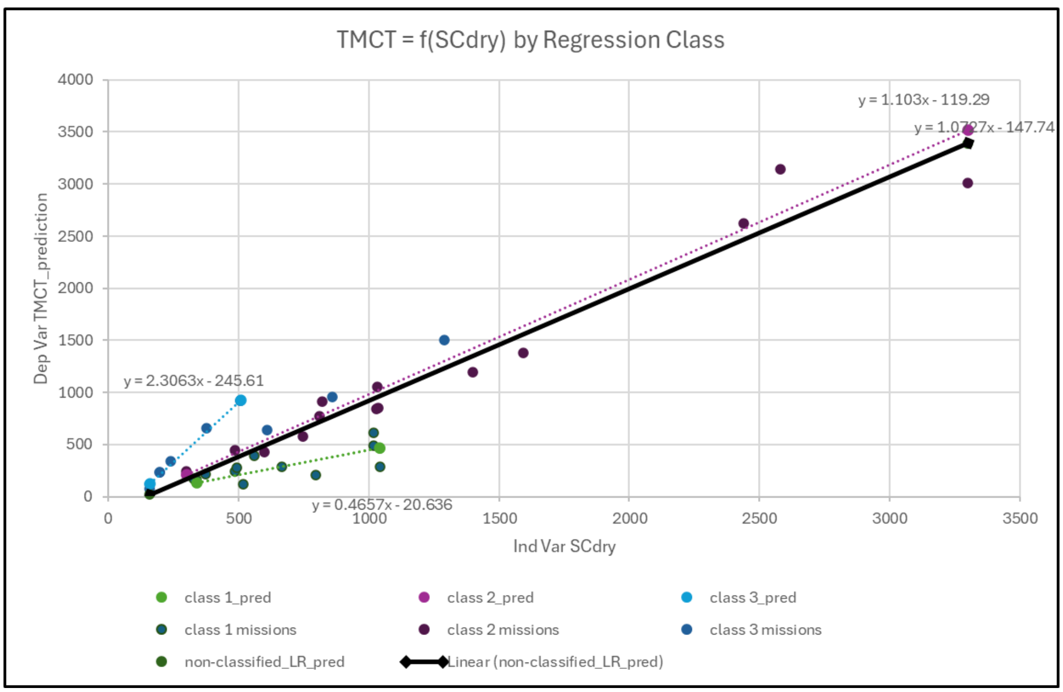
Zero class limit = 0.05

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
TMCT = f(SCdry)				

Class Regression Threshold	41%
Class Residual Threshold	11%

	Regression n	Residual n	Min %e	Max %e
Class 1	9	14	-18.2%	13.7%
Class 2	20	9	-22.0%	20.2%
Class 3	6	9	-3.9%	21.7%

Model	LR	CR	CR w/corr
mean	15.7%	-8.9%	0.6%
|mean|	50.7%	24.1%	9.0%
st.dev	66.3%	30.1%	11.7%
st.dev | |	44.8%	19.7%	7.3%
co.var	23.7%	-29.6%	20.36
co.var| |	113.1%	122.6%	1.30
|mean| + st.dev | |	95.5%	43.8%	16.3%



				CR Modeler [TMCT = f(SCdry)]			
Data ID	Mission	Ind Var SCdry	Dep Var TMCT	reg class	res class	Predict	Pred %e
23	Mars Polar Lander	519	147.95	1	1	123.7	-16.4%
6	Deep Impact	1042	352.5	1	1	290.2	-17.7%
22	Mars Pathfinder	795	258.8	1	1	211.6	-18.2%
26	MERA - Spirit	1018	483.3	1	2	492.1	1.8%
20	Mars Global Surveyor	666.3	303	1	0	289.6	-4.4%
30	NEAR Shoemaker	487	230.8	1	2	239.1	3.6%
27	MERB - Opportunity	1018	595.6	1	3	610.7	2.5%
19	Mars Climate Orbiter	338	147.95	1	2	168.2	13.7%
9	Genesis	494	264	1	3	280.9	6.4%
7	Deep Space 1	373.7	180.6	2	1	217.0	20.2%
3	Contour	328	145.8	2	1	173.3	18.9%
4	Dart	560	332	2	1	395.0	19.0%
5	Dawn	747.1	505.6	2	1	573.9	13.5%
34	Psyche	1400	1108.1	2	1	1197.9	8.1%
13	Juno	1593	1304.5	2	1	1382.3	6.0%
33	Phoenix	597	417	2	1	430.4	3.2%
21	Mars Observer	1028	825.5	2	1	842.3	2.0%
29	MSL - Curiosity	3300	3046	2	1	3013.8	-1.1%
18	Magellan	1035	864.8	2	1	849.0	-1.8%
25	MAVEN	809	771.2	2	0	773.1	0.2%
24	Mars Recon Orbiter	1031	1072.2	2	2	1056.7	-1.4%
17	M2020 - Perseverance	2439	2880.6	2	2	2624.9	-8.9%
2	Cassini	2581	3206.8	2	3	3140.8	-2.1%
28	Messenger	485.2	457.8	2	2	448.9	-1.9%
35	Stardust	300	236.4	2	2	242.6	2.6%
15	Lucy	821	999.2	2	3	911.3	-8.8%
8	Galileo	1289	1783.1	2	3	1504.2	-15.6%
32	Osiris-Rex	860	1125.3	2	3	960.7	-14.6%
12	Insight	608	822.8	2	3	641.5	-22.0%
31	New Horizons	508	889.7	3	0	926.0	4.1%
14	Ladee	240.7	279.2	3	2	339.9	21.7%
1	2001 Mars Odyssey	376.3	680.2	3	2	654.1	-3.8%
16	Lunar Prospector	158.7	63.9	3	1	76.2	19.2%
10	Grail-A	198	243.65	3	3	234.1	-3.9%
11	Grail-B	198	243.65	3	3	234.1	-3.9%

Interpretable (class-wise) CR Model

Class	n	TMCT = f(SCdry)	SCdry(min)	SCdry(max)	mean %e	stdev %e
1	9	TMCT(c1) = -20.64 + 0.47 * SCdry	338	1042	9.41%	6.94%
2	20	TMCT(c2) = -119.29 + 1.10 * SCdry	300	3300.00	8.60%	7.48%
3	6	TMCT(c3) = -245.61 + 2.31 * SCdry	158.7	508.00	9.44%	8.57%
				Model	8.95%	7.32%

Class	n	Corrector = f(SCdry)	Correction = - Corrector
1	14	Corrector(c1) = 20.88 + 0.15 * SCdry	Corrector(c1) = -20.88 + -0.15 * SCdry
2	6	Corrector(c1) = -27.80 + -0.01 * SCdry	Corrector(c1) = 27.80 + 0.01 * SCdry
3	9	Corrector(c1) = 9.35 + -0.16 * SCdry	Corrector(c1) = -9.35 + 0.16 * SCdry

TMCN = f(SCdry)

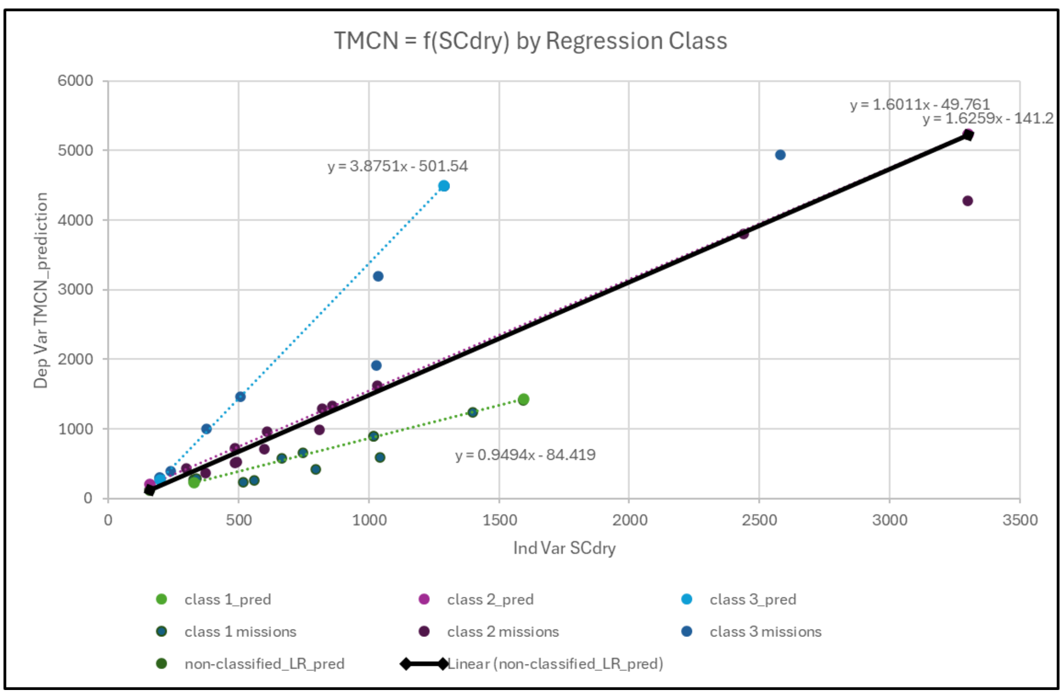
Zero class limit = 0.03

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
TMCN = f(SCdry)				

Class Regression Threshold	32%
Class Residual Threshold	18%

	Regression n	Residual n	Min %e	Max %e
Class 1	15	9	-30.1%	12.0%
Class 2	14	15	-16.6%	18.3%
Class 3	6	7	-15.6%	33.5%

Model	LR	CR	CR w/corr
mean	22.4%	-3.9%	-0.5%
|mean|	48.0%	19.5%	9.4%
st.dev	57.7%	24.9%	12.5%
st.dev | |	38.4%	15.6%	8.0%
co.var	38.9%	-15.7%	-22.72
co.var| |	124.9%	125.2%	1.32
|mean| + st.dev | |	86.4%	35.1%	17.5%



				CR Modeler [TMCN = f(SCdry)]			
Data ID	Mission	Ind Var SCdry	Dep Var TMCN	reg class	res class	Predict	Pred %e
6	Deep Impact	1042	578.3	1	1	584.9	1.1%
23	Mars Polar Lander	519	285.6	1	1	234.5	-17.9%
22	Mars Pathfinder	795	527.6	1	1	419.4	-20.5%
4	Dart	560	374.8	1	1	262.0	-30.1%
26	MERA - Spirit	1018	800.5	1	2	896.5	12.0%
34	Psyche	1400	1202.7	1	2	1241.0	3.2%
20	Mars Global Surveyor	666.3	579.6	1	2	579.4	0.0%
27	MERB - Opportunity	1018	940.1	1	2	896.5	-4.6%
3	Contour	328	258.1	1	2	274.3	6.3%
13	Juno	1593	1669.8	1	2	1415.0	-15.3%
5	Dawn	747.1	749.7	1	2	652.2	-13.0%
19	Mars Climate Orbiter	338	285.6	1	2	283.4	-0.8%
30	NEAR Shoemaker	487	462.8	1	3	515.3	11.4%
9	Genesis	494	475.2	1	3	524.4	10.3%
7	Deep Space 1	373.7	346.6	1	3	369.1	6.5%
33	Phoenix	597	630.1	2	1	710.5	12.8%
25	MAVEN	809	942.5	2	1	990.7	5.1%
29	MSL - Curiosity	3300	4197.7	2	1	4283.2	2.0%
17	M2020 - Perseverance	2439	3338.1	2	2	3801.9	13.9%
15	Lucy	821	1089.6	2	2	1288.6	18.3%
24	Mars Recon Orbiter	1031	1531.4	2	2	1614.8	5.4%
16	Lunar Prospector	158.7	125.2	2	1	131.2	4.8%
32	Osiris-Rex	860	1350.7	2	0	1327.2	-1.7%
28	Messenger	485.2	710.3	2	0	727.1	2.4%
12	Insight	608	1015.1	2	2	957.7	-5.7%
35	Stardust	300	427.6	2	0	430.6	0.7%
21	Mars Observer	1028	2092.7	2	3	1918.3	-8.3%
2	Cassini	2581	5916.8	2	3	4935.0	-16.6%
14	Ladee	240.7	365.8	2	2	387.1	5.8%
18	Magellan	1035	2390.8	3	1	3191.2	33.5%
10	Grail-A	198	327.15	3	3	304.4	-6.9%
11	Grail-B	198	327.15	3	3	304.4	-6.9%
31	New Horizons	508	1472.6	3	0	1467.0	-0.4%
1	2001 Mars Odyssey	376.3	1116.2	3	2	1001.7	-10.3%
8	Galileo	1289	5323.7	3	2	4494.9	-15.6%

Interpretable (class-wise) CR Model

Class	n	TMCN = f(SCdry)	SCdry(min)	SCdry(max)	mean %e	stdev %e
1	15	TMCN(c1) = -84.42 + 0.95 * SCdry	328	1593	10.20%	8.38%
2	14	TMCN(c2) = -49.76 + 1.60 * SCdry	158.7	3300.00	7.39%	5.73%
3	6	TMCN(c3) = -501.54 + 3.88 * SCdry	198	1289.00	12.26%	11.51%
				Model	9.43%	8.02%

Class	n	Corrector = f(SCdry)	Correction = - Corrector
1	9	Corrector(c1) = 28.84 + 0.28 * SCdry	Corrector(c1) = -28.84 + -0.28 * SCdry
2	11	Corrector(c1) = -63.00 + 0.05 * SCdry	Corrector(c1) = 63.00 + -0.05 * SCdry
3	7	Corrector(c1) = 28.89 + -0.34 * SCdry	Corrector(c1) = -28.89 + 0.34 * SCdry

TMCP = f(SCdry)

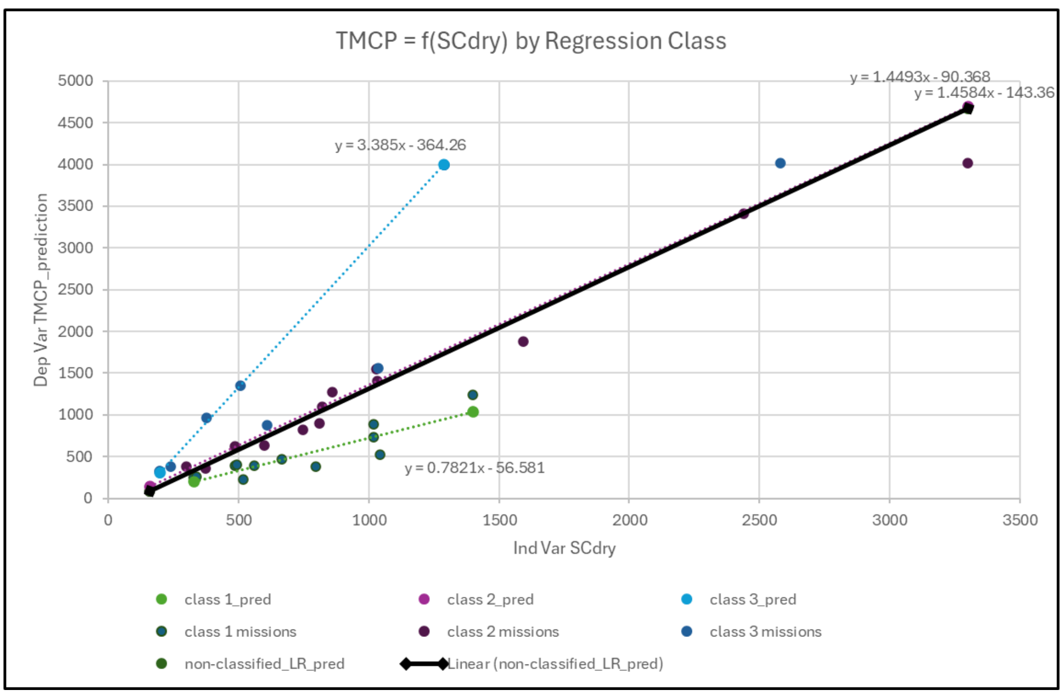
Zero class limit = 0.05

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
TMCP = f(SCdry)				

Class Regression Threshold	37%
Class Residual Threshold	7%

	Regression n	Residual n	Min %e	Max %e
Class 1	12	11	-13.3%	8.3%
Class 2	17	8	-21.0%	19.3%
Class 3	6	13	-3.3%	8.8%

Model	LR	CR	CR w/corr
mean	19.0%	-5.2%	0.3%
|mean|	46.8%	17.8%	7.0%
st.dev	57.8%	22.6%	9.4%
st.dev | |	38.2%	14.6%	6.3%
co.var	32.8%	-23.0%	29.59
co.var| |	122.4%	121.5%	1.36
|mean| + st.dev | |	85.0%	32.4%	13.2%



				CR Modeler [TMCP = f(SCdry)]			
Data ID	Mission	Ind Var SCdry	Dep Var TMCP	reg class	res class	Predict	Pred %e
6	Deep Impact	1042	521.6	1	1	529.0	1.4%
23	Mars Polar Lander	519	243.85	1	1	223.2	-8.5%
22	Mars Pathfinder	795	443.6	1	1	384.6	-13.3%
26	MERA - Spirit	1018	720.7	1	2	736.9	2.3%
4	Dart	560	365.2	1	2	388.8	6.5%
20	Mars Global Surveyor	666.3	496.6	1	2	469.6	-5.4%
34	Psyche	1400	1176.1	1	3	1235.9	5.1%
27	MERB - Opportunity	1018	854.6	1	3	883.0	3.3%
3	Contour	328	226.7	1	3	245.5	8.3%
30	NEAR Shoemaker	487	391.2	1	3	392.4	0.3%
19	Mars Climate Orbiter	338	243.85	1	3	254.8	4.5%
9	Genesis	494	414.4	1	3	398.9	-3.7%
5	Dawn	747.1	690.7	2	1	821.2	18.9%
13	Juno	1593	1594	2	1	1880.2	18.0%
7	Deep Space 1	373.7	296.6	2	1	353.7	19.3%
33	Phoenix	597	576.4	2	1	633.3	9.9%
29	MSL - Curiosity	3300	3902.8	2	1	4017.3	2.9%
25	MAVEN	809	901.2	2	1	898.7	-0.3%
17	M2020 - Perseverance	2439	3244.5	2	2	3410.5	5.1%
15	Lucy	821	1069.5	2	2	1101.1	3.0%
24	Mars Recon Orbiter	1031	1421.6	2	0	1403.8	-1.2%
28	Messenger	485.2	649.6	2	2	621.8	-4.3%
32	Osiris-Rex	860	1302.5	2	3	1277.0	-2.0%
21	Mars Observer	1028	1636	2	3	1544.3	-5.6%
16	Lunar Prospector	158.7	106.3	2	1	84.5	-20.5%
35	Stardust	300	371.8	2	3	386.0	3.8%
12	Insight	608	973	2	3	876.0	-10.0%
18	Magellan	1035	1845.6	2	3	1555.4	-15.7%
2	Cassini	2581	5079.8	2	3	4015.1	-21.0%
14	Ladee	240.7	348.7	3	1	379.2	8.8%
10	Grail-A	198	312.25	3	2	321.3	2.9%
11	Grail-B	198	312.25	3	2	321.3	2.9%
8	Galileo	1289	3983.5	3	0	3999.1	0.4%
31	New Horizons	508	1375.3	3	0	1355.3	-1.5%
1	2001 Mars Odyssey	376.3	994.4	3	3	961.9	-3.3%

Interpretable (class-wise) CR Model

Class	n	TMCP = f(SCdry)	SCdry(min)	SCdry(max)	mean %e	stdev %e
1	12	TMCP(c1) = -56.58 + 0.78 * SCdry	328	1400	5.22%	3.58%
2	17	TMCP(c2) = -90.37 + 1.45 * SCdry	158.7	3300.00	9.49%	7.65%
3	6	TMCP(c3) = -364.26 + 3.39 * SCdry	198	1289.00	3.28%	2.90%
				Model	6.96%	6.29%

Class	n	Corrector = f(SCdry)	Correction = - Corrector
1	11	Corrector(c1) = 23.78 + 0.20 * SCdry	Corrector(c1) = -23.78 + -0.20 * SCdry
2	5	Corrector(c1) = -19.69 + 0.02 * SCdry	Corrector(c1) = 19.69 + -0.02 * SCdry
3	13	Corrector(c1) = 0.92 + -0.14 * SCdry	Corrector(c1) = -0.92 + 0.14 * SCdry

TDCT = f(SCdry)

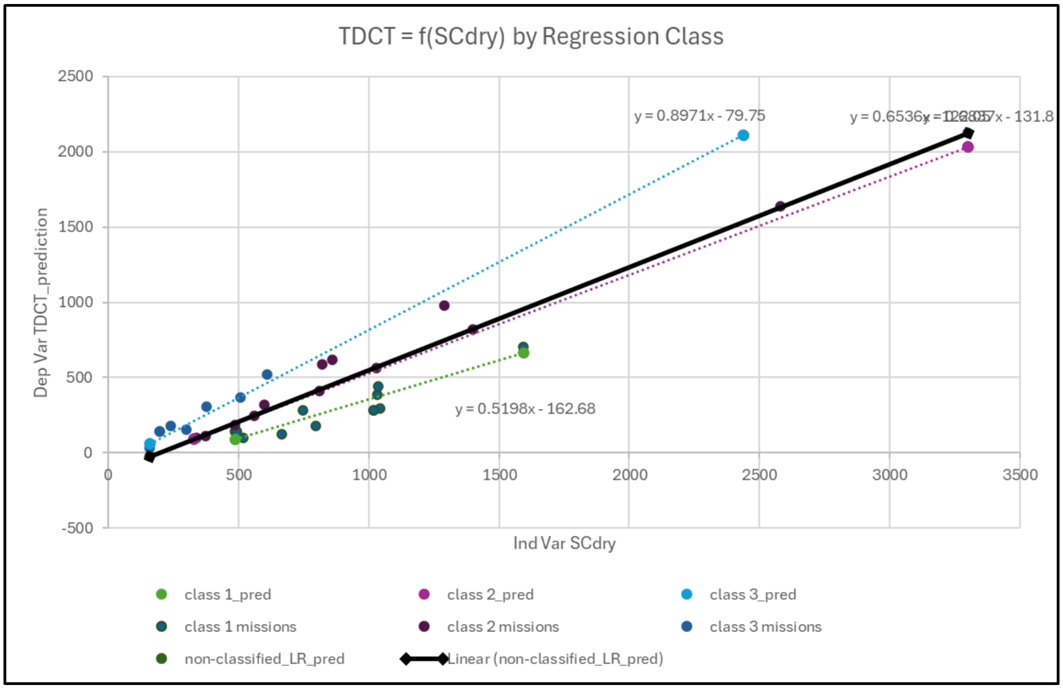
Zero Class Limit = 0.06

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
TDCT = f(SCdry)				

Class Regression Threshold	16%
Class Residual Threshold	14%

	Regression n	Residual n	Min %e	Max %e
Class 1	12	10	-11.6%	27.8%
Class 2	11	10	-12.8%	16.4%
Class 3	12	10	-16.6%	20.8%

Model	LR	CR	CR w/corr
mean	5.5%	-3.0%	0.3%
|mean|	49.5%	21.8%	8.7%
st.dev	69.4%	29.9%	10.6%
st.dev | |	48.2%	20.5%	5.9%
co.var	7.9%	-10.2%	38.41
co.var| |	102.7%	106.4%	1.22
|mean| + st.dev | |	97.8%	42.2%	14.6%



					CR Modeler [TDCT = f(SCdry)]			
Data ID	Mission	Ind Var SCdry	Dep Var TDCT	reg class	res class	Predict	Pred %e	
6	Deep Impact	1042	230.9	1	1	295.1	27.8%	
20	Mars Global Surveyor	666.3	130.7	1	1	123.7	-5.4%	
23	Mars Polar Lander	519	94.5	1	2	101.8	7.8%	
22	Mars Pathfinder	795	199.2	1	1	182.4	-8.4%	
26	MERA - Spirit	1018	321.5	1	1	284.2	-11.6%	
27	MERB - Opportunity	1018	321.5	1	1	284.2	-11.6%	
30	NEAR Shoemaker	487	124.9	1	3	141.3	13.2%	
24	Mars Recon Orbiter	1031	419.2	1	2	387.9	-7.5%	
5	Dawn	747.1	282.9	1	3	283.8	0.3%	
13	Juno	1593	726.2	1	2	701.9	-3.3%	
9	Genesis	494	163	1	3	145.2	-10.9%	
18	Magellan	1035	463.2	1	3	441.5	-4.7%	
2	Cassini	2581	1408.5	2	2	1639.7	16.4%	
21	Mars Observer	1028	511.2	2	2	564.3	10.4%	
25	MAVEN	809	382.4	2	2	412.7	7.9%	
7	Deep Space 1	373.7	114.2	2	2	111.3	-2.6%	
19	Mars Climate Orbiter	338	94.5	2	0	98.9	4.6%	
4	Dart	560	240.6	2	0	243.9	1.4%	
29	MSL - Curiosity	3300	2142.4	2	0	2034.7	-5.0%	
34	Psyche	1400	844.5	2	2	821.9	-2.7%	
3	Contour	328	96.8	2	0	92.3	-4.6%	
28	Messenger	485.2	216.2	2	2	188.5	-12.8%	
33	Phoenix	597	318.2	2	3	322.1	1.2%	
8	Galileo	1289	902.3	3	1	977.0	8.3%	
15	Lucy	821	523.1	3	1	586.9	12.2%	
32	Osiris-Rex	860	591.6	3	1	619.4	4.7%	
17	M2020 - Perseverance	2439	2232.2	3	0	2108.2	-5.6%	
31	New Horizons	508	347.7	3	2	370.3	6.5%	
35	Stardust	300	126.4	3	1	152.7	20.8%	
12	Insight	608	596.4	3	3	519.9	-12.8%	
1	2001 Mars Odyssey	376.3	366.2	3	3	305.6	-16.6%	
14	Ladee	240.7	204.3	3	3	180.1	-11.8%	
10	Grail-A	198	147.95	3	3	140.6	-4.9%	
11	Grail-B	198	147.95	3	3	140.6	-4.9%	
16	Lunar Prospector	158.7	30.6	3	1	34.9	14.0%	

Interpretable (class-wise) CR Model

Class	n	TDCT = f(SCdry)	SCdry(min)	SCdry(max)	mean %e	stdev %e
1	12	TDCT(c1) = -162.68 + 0.52 * SCdry	487	1593	9.37%	6.94%
2	11	TDCT(c2) = -122.05 + 0.65 * SCdry	328	3300.00	6.33%	4.99%
3	12	TDCT(c3) = -79.75 + 0.90 * SCdry	158.7	2439.00	10.26%	5.26%
				Model	8.72%	5.89%

Class	n	Corrector = f(SCdry)	Correction = - Corrector
1	10	Corrector(c1) = 17.65 + 0.06 * SCdry	Corrector(c1) = -17.65 + -0.06 * SCdry
2	5	Corrector(c1) = 25.44 + -0.04 * SCdry	Corrector(c1) = -25.44 + 0.04 * SCdry
3	10	Corrector(c1) = -37.24 + -0.03 * SCdry	Corrector(c1) = 37.24 + 0.03 * SCdry

TDCN = f(SCdry)

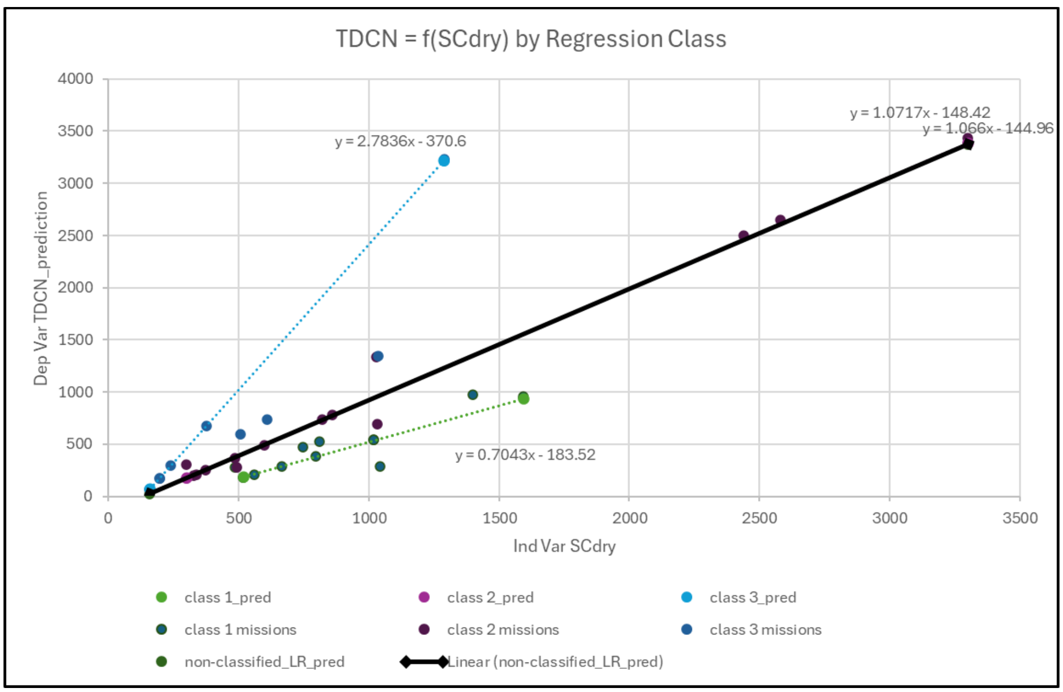
Zero Class Limit = 0.00

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
TDCN = f(SCdry)				

Class Regression Threshold	47%
Class Residual Threshold	24%

	Regression n	Residual n	Min %e	Max %e
Class 1	8	7	-27.3%	9.1%
Class 2	21	23	-12.2%	24.4%
Class 3	6	5	-12.8%	9.1%

Model	LR	CR	CR w/corr
mean	14.4%	-5.0%	1.1%
|mean|	46.1%	18.7%	8.3%
st.dev	55.9%	23.9%	11.3%
st.dev | |	34.0%	15.4%	7.6%
co.var	25.7%	-20.8%	10.31
co.var| |	135.6%	121.7%	1.36
|mean| + st.dev | |	80.1%	34.1%	15.9%



				CR Modeler [TDCN = f(SCdry)]			
Data ID	Mission	Ind Var SCdry	Dep Var TDCN	reg class	res class	Predict	Pred %e
6	Deep Impact	1042	390.5	1	1	283.9	-27.3%
23	Mars Polar Lander	519	183.25	1	2	181.6	-0.9%
20	Mars Global Surveyor	666.3	263.9	1	2	287.9	9.1%
22	Mars Pathfinder	795	407.8	1	2	380.7	-6.6%
26	MERA - Spirit	1018	547.2	1	2	541.5	-1.0%
27	MERB - Opportunity	1018	547.2	1	2	541.5	-1.0%
4	Dart	560	274.8	1	2	211.2	-23.1%
13	Juno	1593	995.9	1	2	956.2	-4.0%
5	Dawn	747.1	444.7	2	1	475.7	7.0%
30	NEAR Shoemaker	487	257.9	2	1	276.2	7.1%
34	Psyche	1400	935.9	2	1	976.6	4.3%
25	MAVEN	809	499.5	2	1	523.2	4.7%
24	Mars Recon Orbiter	1031	681.6	2	1	693.5	1.7%
9	Genesis	494	300.5	2	1	281.5	-6.3%
15	Lucy	821	594.1	2	2	736.2	23.9%
3	Contour	328	172	2	2	199.5	16.0%
19	Mars Climate Orbiter	338	183.25	2	2	210.4	14.8%
7	Deep Space 1	373.7	221.2	2	2	249.3	12.7%
29	MSL - Curiosity	3300	3068.5	2	2	3435.0	11.9%
32	Osiris-Rex	860	747.5	2	2	778.7	4.2%
28	Messenger	485.2	363.3	2	2	370.7	2.0%
33	Phoenix	597	484.8	2	2	492.4	1.6%
17	M2020 - Perseverance	2439	2635.5	2	2	2497.7	-5.2%
2	Cassini	2581	3019.5	2	2	2652.3	-12.2%
21	Mars Observer	1028	1309.3	2	3	1334.1	1.9%
35	Stardust	300	244	2	3	303.5	24.4%
18	Magellan	1035	1343.8	2	3	1344.0	0.0%
12	Insight	608	745.8	2	3	739.5	-0.8%
31	New Horizons	508	676.2	2	3	598.0	-11.6%
14	Ladee	240.7	269.8	3	2	294.3	9.1%
16	Lunar Prospector	158.7	60.2	3	2	64.7	7.5%
8	Galileo	1289	3218.2	3	2	3230.1	0.4%
1	2001 Mars Odyssey	376.3	677	3	2	674.1	-0.4%
10	Grail-A	198	200.4	3	2	174.8	-12.8%
11	Grail-B	198	200.4	3	2	174.8	-12.8%

Interpretable (class-wise) CR Model

Class	n	TDCN = f(SCdry)	SCdry(min)	SCdry(max)	mean %e	stdev %e
1	8	TDCN(c1) = -183.52 + 0.70 * SCdry	519	1593	9.14%	10.40%
2	21	TDCN(c2) = -148.42 + 1.07 * SCdry	300	3300.00	8.31%	7.13%
3	6	TDCN(c3) = -370.60 + 2.78 * SCdry	158.7	1289.00	7.16%	5.63%
				Model	8.30%	7.57%

Class	n	Corrector = f(SCdry)	Correction = - Corrector
1	7	Corrector(c1) = -51.02 + 0.30 * SCdry	Corrector(c1) = 51.02 + -0.30 * SCdry
2	23	Corrector(c1) = 9.14 + -0.02 * SCdry	Corrector(c1) = -9.14 + 0.02 * SCdry
3	5	Corrector(c1) = -27.23 + -0.34 * SCdry	Corrector(c1) = 27.23 + 0.34 * SCdry

TDCP = f(SCdry)

Zero Class Limit = 0.03

Model Summary (35 missions)

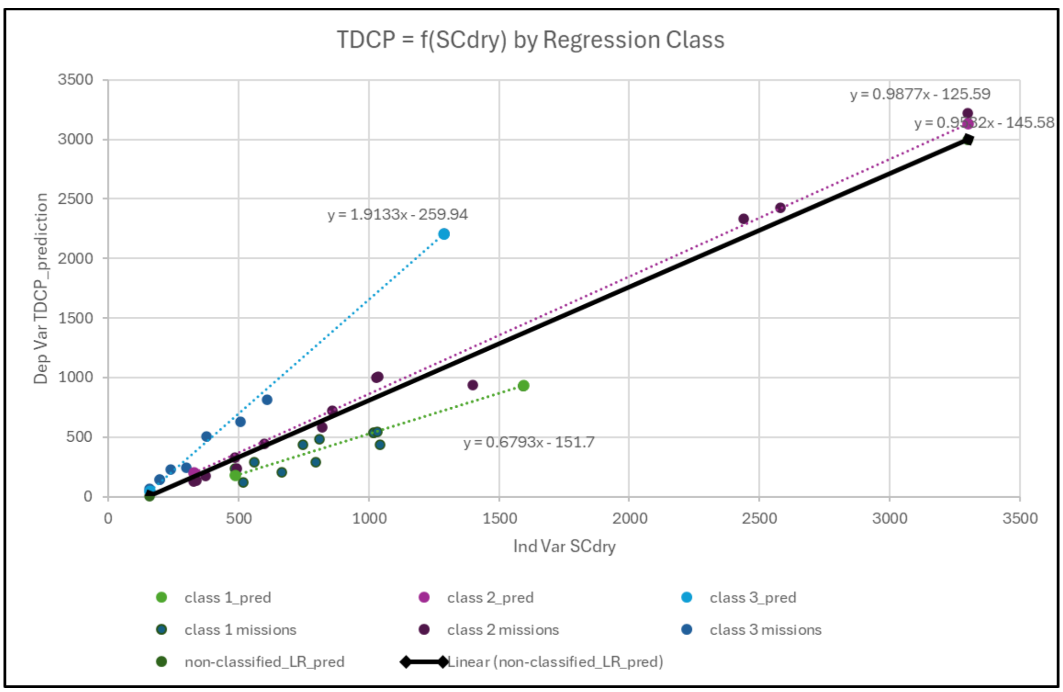
Classified Linear Reg with Classified Residual Reg

TDCP = f(SCdry)

Class Regression Threshold	19%
Class Residual Threshold	12%

	Regression n	Residual n	Min %e	Max %e
Class 1	14	11	-24.1%	25.5%
Class 2	12	8	-14.4%	12.9%
Class 3	9	13	-23.5%	31.7%

Model	LR	CR	CR w/corr
mean	9.7%	-3.0%	-1.0%
|mean|	44.7%	19.6%	10.2%
st.dev	57.2%	23.9%	13.1%
st.dev | |	36.2%	13.7%	8.1%
co.var	17.0%	-12.3%	-13.75
co.var| |	123.5%	143.2%	1.29
|mean| + st.dev | |	80.9%	33.3%	18.3%



				CR Modeler [TDCP = f(SCdry)]			
Data ID	Mission	Ind Var SCdry	Dep Var TDCP	reg class	res class	Predict	Pred %e
6	Deep Impact	1042	349.4	1	1	438.4	25.5%
23	Mars Polar Lander	519	156.2	1	1	118.6	-24.1%
20	Mars Global Surveyor	666.3	222.5	1	1	208.6	-6.2%
22	Mars Pathfinder	795	342.5	1	1	287.3	-16.1%
26	MERA - Spirit	1018	489.9	1	2	534.2	9.0%
27	MERB - Opportunity	1018	489.9	1	2	534.2	9.0%
30	NEAR Shoemaker	487	216.2	1	3	233.8	8.2%
13	Juno	1593	943.9	1	0	930.4	-1.4%
4	Dart	560	267.9	1	3	290.4	8.4%
5	Dawn	747.1	404.6	1	3	435.3	7.6%
24	Mars Recon Orbiter	1031	618.1	1	2	543.5	-12.1%
25	MAVEN	809	475.6	1	3	483.3	1.6%
34	Psyche	1400	912.5	1	3	941.0	3.1%
9	Genesis	494	260	1	3	239.3	-8.0%
19	Mars Climate Orbiter	338	156.2	2	1	138.3	-11.5%
7	Deep Space 1	373.7	188.7	2	1	171.1	-9.3%
3	Contour	328	150.8	2	1	129.1	-14.4%
15	Lucy	821	579.2	2	1	582.6	0.6%
29	MSL - Curiosity	3300	2853.8	2	2	3221.5	12.9%
28	Messenger	485.2	326.4	2	2	326.2	-0.1%
33	Phoenix	597	442.8	2	2	441.2	-0.4%
32	Osiris-Rex	860	712.2	2	0	723.8	1.6%
2	Cassini	2581	2484.1	2	0	2423.7	-2.4%
17	M2020 - Perseverance	2439	2558.7	2	2	2335.9	-8.7%
21	Mars Observer	1028	1021.2	2	3	996.1	-2.5%
18	Magellan	1035	1030.2	2	3	1003.6	-2.6%
35	Stardust	300	208.3	3	1	246.6	18.4%
12	Insight	608	712.9	3	1	815.0	14.3%
31	New Horizons	508	576.3	3	1	630.5	9.4%
8	Galileo	1289	2303.7	3	2	2211.7	-4.0%
1	2001 Mars Odyssey	376.3	586.3	3	3	504.2	-14.0%
14	Ladee	240.7	257.6	3	3	231.9	-10.0%
10	Grail-A	198	190.85	3	3	146.1	-23.5%
11	Grail-B	198	190.85	3	3	146.1	-23.5%
16	Lunar Prospector	158.7	51	3	3	67.2	31.7%

Interpretable (class-wise) CR Model

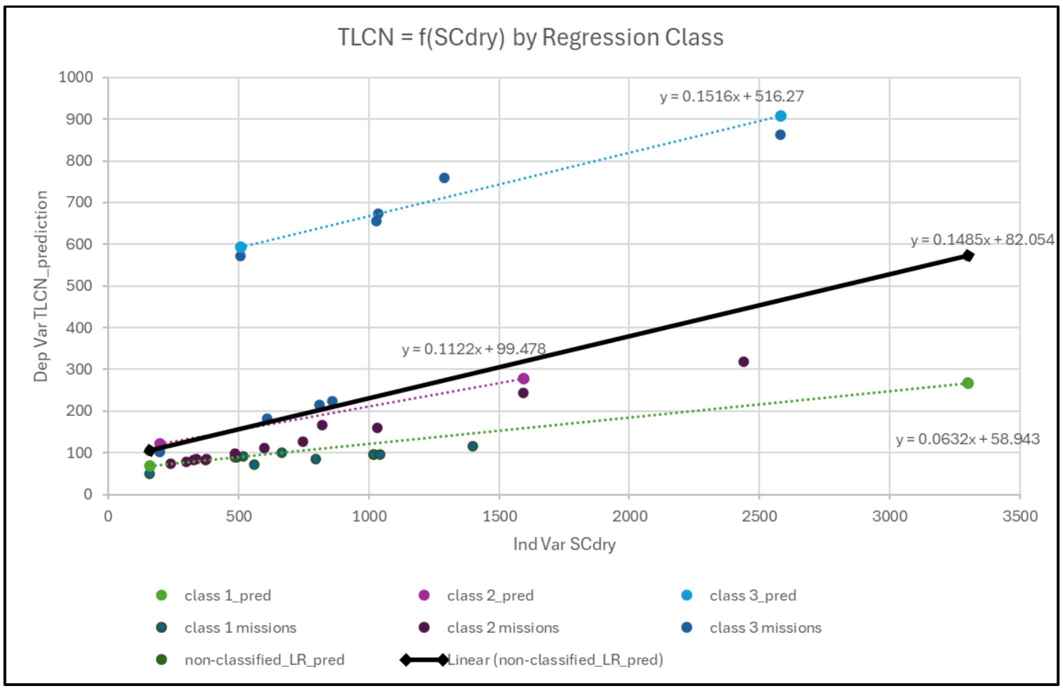
Class	n	TDCP = f(SCdry)	SCdry(min)	SCdry(max)	mean %e	stdev %e
1	14	TDCP(c1) = -151.70 + 0.68 * SCdry	487	1593	10.02%	7.35%
2	12	TDCP(c2) = -125.59 + 0.99 * SCdry	328	3300.00	5.58%	5.37%
3	9	TDCP(c3) = -259.94 + 1.91 * SCdry	158.7	1289.00	16.52%	8.59%
				Model	10.17%	8.09%

Class	n	Corrector = f(SCdry)	Correction = - Corrector
1	11	Corrector(c1) = 47.08 + 0.07 * SCdry	Corrector(c1) = -47.08 + -0.07 * SCdry
2	5	Corrector(c1) = 47.27 + -0.04 * SCdry	Corrector(c1) = -47.27 + 0.04 * SCdry
3	13	Corrector(c1) = -8.33 + -0.10 * SCdry	Corrector(c1) = 8.33 + 0.10 * SCdry

TLCN = f(SCdry)

Zero Class Limit = 0.06

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
TLCN = f(SCdry)				
	Class Regression Threshold		26%	
	Class Residual Threshold		9%	
	Regression n	Residual n	Min %e	Max %e
Class 1	23	13	-14.8%	20.8%
Class 2	7	3	-8.5%	1.0%
Class 3	5	10	-19.6%	37.1%
Model	LR	CR	CR w/corr	
mean	47.9%	-3.5%	-0.5%	
|mean|	67.1%	16.2%	7.0%	
st.dev	65.5%	20.0%	10.5%	
st.dev | |	45.0%	12.0%	7.7%	
co.var	73.1%	-17.6%	-19.44	
co.var| |	149.0%	134.5%	1.49	
|mean| + st.dev | |	112.1%	28.2%	14.8%	



				CR Modeler [TLCN = f(SCdry)]			
Data ID	Mission	Ind Var SCdry	Dep Var TLCN	reg class	res class	Predict	Pred %e
26	MERA - Spirit	1018	86.8	1	1	95.7	10.3%
27	MERB - Opportunity	1018	86.8	1	1	95.7	10.3%
34	Psyche	1400	115.9	1	1	115.7	-0.2%
6	Deep Impact	1042	96.1	1	1	97.0	0.9%
29	MSL - Curiosity	3300	255.7	1	0	267.5	4.6%
4	Dart	560	76.3	1	1	71.7	-6.0%
22	Mars Pathfinder	795	95.7	1	1	84.0	-12.2%
16	Lunar Prospector	158.7	51.1	1	1	50.7	-0.8%
23	Mars Polar Lander	519	87.3	1	0	91.7	5.1%
20	Mars Global Surveyor	666.3	105.2	1	0	101.1	-3.9%
9	Genesis	494	90.4	1	0	90.2	-0.3%
30	NEAR Shoemaker	487	90.9	1	0	89.7	-1.3%
17	M2020 - Perseverance	2439	263.9	1	3	318.8	20.8%
24	Mars Recon Orbiter	1031	142.1	1	3	159.8	12.5%
35	Stardust	300	78	1	0	77.9	-0.1%
7	Deep Space 1	373.7	87.3	1	0	82.6	-5.4%
14	Ladee	240.7	75.5	1	0	74.2	-1.8%
3	Contour	328	86.1	1	2	83.9	-2.6%
19	Mars Climate Orbiter	338	87.3	1	2	84.2	-3.5%
5	Dawn	747.1	137.4	1	3	127.7	-7.0%
1	2001 Mars Odyssey	376.3	99.7	1	3	85.9	-13.9%
28	Messenger	485.2	115.2	1	3	98.2	-14.8%
33	Phoenix	597	127.7	1	3	110.8	-13.2%
13	Juno	1593	254.1	2	1	244.5	-3.8%
15	Lucy	821	167.4	2	1	166.2	-0.7%
10	Grail-A	198	101.95	2	1	103.0	1.0%
11	Grail-B	198	101.95	2	1	103.0	1.0%
32	Osiris-Rex	860	224.5	2	3	223.2	-0.6%
25	MAVEN	809	218.1	2	3	214.9	-1.4%
12	Insight	608	199.3	2	3	182.4	-8.5%
2	Cassini	2581	809.3	3	1	863.2	6.7%
31	New Horizons	508	416.8	3	1	571.3	37.1%
18	Magellan	1035	649.8	3	0	673.2	3.6%
21	Mars Observer	1028	736.2	3	2	655.4	-11.0%
8	Galileo	1289	946	3	3	760.3	-19.6%

Interpretable (class-wise) CR Model

Class	n	TLCN = f(SCdry)	SCdry(min)	SCdry(max)	mean %e	stdev %e
1	23	TLCN(c1) = 58.94 + 0.06 * SCdry	158.7	3300	6.58%	5.80%
2	7	TLCN(c2) = 99.48 + 0.11 * SCdry	198	1593.00	2.44%	2.88%
3	5	TLCN(c3) = 516.27 + 0.15 * SCdry	508	2581.00	15.59%	13.43%
				Model	7.04%	7.73%

Class	n	Corrector = f(SCdry)	Correction = - Corrector
1	13	Corrector(c1) = 16.56 + 0.01 * SCdry	Corrector(c1) = -16.56 + -0.01 * SCdry
2	-6	Corrector(c1) = -14.06 + 0.03 * SCdry	Corrector(c1) = 14.06 + -0.03 * SCdry
3	10	Corrector(c1) = 15.57 + -0.05 * SCdry	Corrector(c1) = -15.57 + 0.05 * SCdry

TLCP = f(SCdry)

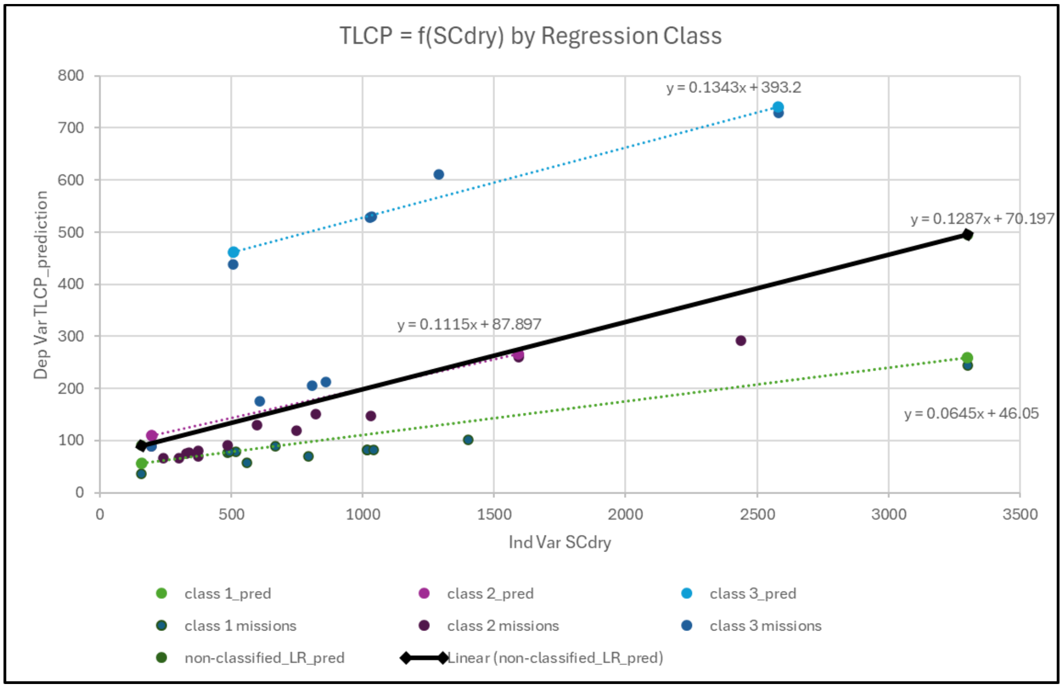
Zero Class Limit = 0.03

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
TLCP = f(SCdry)				

Class Regression Threshold	26%
Class Residual Threshold	9%

	Regression n	Residual n	Min %e	Max %e
Class 1	22	12	-22.9%	14.4%
Class 2	8	7	-9.0%	10.5%
Class 3	5	12	-14.7%	22.5%

Model	LR	CR	CR w/corr
mean	41.6%	-3.2%	-1.9%
|mean|	61.8%	16.0%	7.6%
st.dev	62.6%	19.5%	9.5%
st.dev | |	42.0%	11.3%	5.8%
co.var	66.5%	-16.3%	-5.06
co.var| |	147.1%	141.3%	1.24
|mean| + st.dev | |	103.9%	27.2%	13.4%



				CR Modeler [TLCP = f(SCdry)]			
Data ID	Mission	Ind Var SCdry	Dep Var TLCP	reg class	res class	Predict	Pred %e
26	MERA - Spirit	1018	77.5	1	1	81.6	5.2%
27	MERB - Opportunity	1018	77.5	1	1	81.6	5.2%
6	Deep Impact	1042	86.8	1	1	82.8	-4.6%
34	Psyche	1400	112.7	1	1	101.6	-9.8%
22	Mars Pathfinder	795	80.5	1	1	69.8	-13.2%
16	Lunar Prospector	158.7	43.4	1	1	36.4	-16.1%
29	MSL - Curiosity	3300	244.1	1	2	244.2	0.0%
4	Dart	560	74.6	1	1	57.5	-22.9%
23	Mars Polar Lander	519	74.6	1	2	79.1	6.0%
20	Mars Global Surveyor	666.3	88.9	1	0	89.0	0.1%
30	NEAR Shoemaker	487	76.2	1	0	77.5	1.7%
9	Genesis	494	78.1	1	0	77.9	-0.2%
35	Stardust	300	66.9	1	0	65.4	-2.2%
7	Deep Space 1	373.7	74.7	1	2	70.5	-5.7%
24	Mars Recon Orbiter	1031	129.1	1	3	147.7	14.4%
19	Mars Climate Orbiter	338	74.6	1	3	76.5	2.6%
17	M2020 - Perseverance	2439	257.8	1	3	292.3	13.4%
3	Contour	328	75.9	1	3	75.5	-0.5%
14	Ladee	240.7	71.6	1	3	66.5	-7.1%
1	2001 Mars Odyssey	376.3	86.3	1	3	80.5	-6.8%
5	Dawn	747.1	125.7	1	3	118.6	-5.7%
28	Messenger	485.2	104.2	1	3	91.6	-12.0%
33	Phoenix	597	117.1	2	1	129.4	10.5%
13	Juno	1593	243.8	2	2	259.6	6.5%
15	Lucy	821	163.6	2	1	151.7	-7.3%
10	Grail-A	198	97.8	2	1	89.6	-8.4%
11	Grail-B	198	97.8	2	1	89.6	-8.4%
25	MAVEN	809	207.3	2	3	204.8	-1.2%
32	Osiris-Rex	860	217.6	2	3	212.4	-2.4%
12	Insight	608	192	2	3	174.7	-9.0%
2	Cassini	2581	679.7	3	2	728.7	7.2%
18	Magellan	1035	501.1	3	2	529.1	5.6%
31	New Horizons	508	357.1	3	1	437.4	22.5%
21	Mars Observer	1028	576.2	3	2	528.2	-8.3%
8	Galileo	1289	716.8	3	3	611.3	-14.7%

Interpretable (class-wise) CR Model

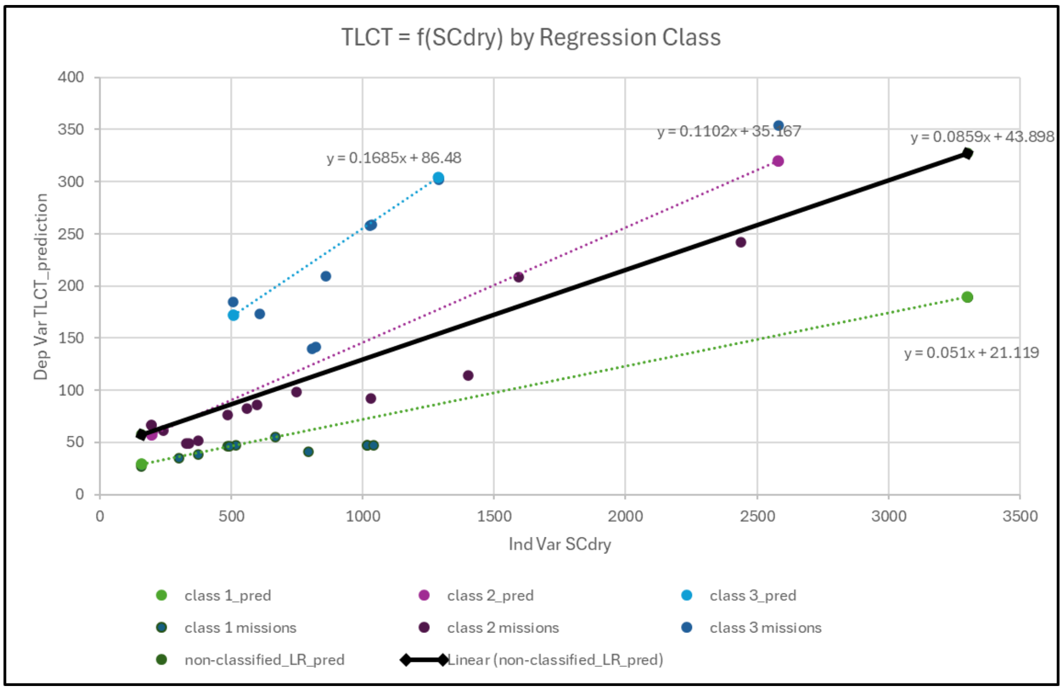
Class	n	TLCP = f(SCdry)	SCdry(min)	SCdry(max)	mean %e	stdev %e
1	22	TLCP(c1) = 46.05 + 0.06 * SCdry	158.7	3300	7.07%	6.08%
2	8	TLCP(c2) = 87.90 + 0.11 * SCdry	198	1593.00	6.69%	3.26%
3	5	TLCP(c3) = 393.20 + 0.13 * SCdry	508	2581.00	11.67%	6.96%
				Model	7.64%	5.79%

Class	n	Corrector = f(SCdry)	Correction = - Corrector
1	12	Corrector(c1) = 17.98 + 0.01 * SCdry	Corrector(c1) = -17.98 + -0.01 * SCdry
2	3	Corrector(c1) = -2.24 + 0.01 * SCdry	Corrector(c1) = 2.24 + -0.01 * SCdry
3	12	Corrector(c1) = 4.24 + -0.04 * SCdry	Corrector(c1) = -4.24 + 0.04 * SCdry

TLCT = f(SCdry)

Zero Class Limit = 0.06

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
TLCT = f(SCdry)				
Class Regression Threshold			34%	
Class Residual Threshold			14%	
	Regression n	Residual n	Min %e	Max %e
Class 1	17	11	-17.5%	8.8%
Class 2	12	7	-16.9%	19.8%
Class 3	6	11	-15.2%	14.1%
Model	LR	CR	CR w/corr	
mean	34.4%	-3.1%	-1.8%	
|mean|	58.5%	19.2%	8.6%	
st.dev	64.1%	22.3%	10.0%	
st.dev ||	42.4%	11.4%	5.2%	
co.var	53.7%	-14.0%	-5.55	
co.var||	138.0%	168.4%	1.16	
|mean| + st.dev ||	100.9%	30.6%	13.8%	



				CR Modeler [TLCT = f(SCdry)]			
Data ID	Mission	Ind Var SCdry	Dep Var TLCT	reg class	res class	Predict	Pred %e
26	MERA - Spirit	1018	50.6	1	1	46.9	-7.3%
27	MERB - Opportunity	1018	50.6	1	1	46.9	-7.3%
22	Mars Pathfinder	795	47.1	1	1	41.2	-12.5%
6	Deep Impact	1042	57.6	1	1	47.5	-17.5%
16	Lunar Prospector	158.7	26	1	2	27.3	4.9%
23	Mars Polar Lander	519	45.3	1	0	47.6	5.0%
30	NEAR Shoemaker	487	44	1	0	45.9	4.4%
20	Mars Global Surveyor	666.3	52.6	1	0	55.1	4.7%
9	Genesis	494	48.7	1	0	46.3	-4.9%
35	Stardust	300	40.8	1	2	34.5	-15.5%
29	MSL - Curiosity	3300	194.7	1	0	189.3	-2.8%
7	Deep Space 1	373.7	45.4	1	2	38.2	-15.8%
19	Mars Climate Orbiter	338	45.3	1	3	49.3	8.8%
24	Mars Recon Orbiter	1031	90	1	3	91.8	2.1%
3	Contour	328	49	1	3	48.7	-0.7%
34	Psyche	1400	112.7	1	3	114.5	1.6%
1	2001 Mars Odyssey	376.3	53.9	1	3	51.6	-4.2%
4	Dart	560	68.8	2	1	82.5	19.8%
28	Messenger	485.2	69	2	1	76.1	10.3%
5	Dawn	747.1	91.3	2	1	98.3	7.7%
33	Phoenix	597	86.2	2	1	85.6	-0.7%
17	M2020 - Perseverance	2439	230.4	2	1	241.8	5.0%
14	Ladee	240.7	58.7	2	0	61.7	5.1%
13	Juno	1593	190.4	2	2	208.8	9.7%
10	Grail-A	198	76.4	2	3	66.5	-13.0%
11	Grail-B	198	76.4	2	3	66.5	-13.0%
15	Lucy	821	149.4	2	3	141.7	-5.2%
25	MAVEN	809	168.7	2	3	140.2	-16.9%
2	Cassini	2581	398.4	2	3	354.1	-11.1%
32	Osiris-Rex	860	183.5	3	1	209.4	14.1%
12	Insight	608	163.4	3	1	173.3	6.1%
18	Magellan	1035	235	3	2	259.0	10.2%
8	Galileo	1289	323.9	3	2	301.8	-6.8%
21	Mars Observer	1028	293.1	3	2	257.8	-12.0%
31	New Horizons	508	218	3	3	184.8	-15.2%

Interpretable (class-wise) CR Model

Class	n	TLCT = f(SCdry)	SCdry(min)	SCdry(max)	mean %e	stdev %e
1	17	TLCT(c1) = 21.12 + 0.05 * SCdry	158.7	3300	7.05%	5.22%
2	12	TLCT(c2) = 35.17 + 0.11 * SCdry	198	2581.00	9.79%	5.46%
3	6	TLCT(c3) = 86.48 + 0.17 * SCdry	508	1289.00	10.74%	3.76%
				Model	8.62%	5.20%

Class	n	Corrector = f(SCdry)	Correction = - Corrector
1	11	Corrector(c1) = 0.22 + 0.03 * SCdry	Corrector(c1) = -0.22 + -0.03 * SCdry
2	1	Corrector(c1) = 1.93 + 0.00 * SCdry	Corrector(c1) = -1.93 + 0.00 * SCdry
3	11	Corrector(c1) = -7.40 + -0.01 * SCdry	Corrector(c1) = 7.40 + 0.01 * SCdry

TLCN = f(LM)

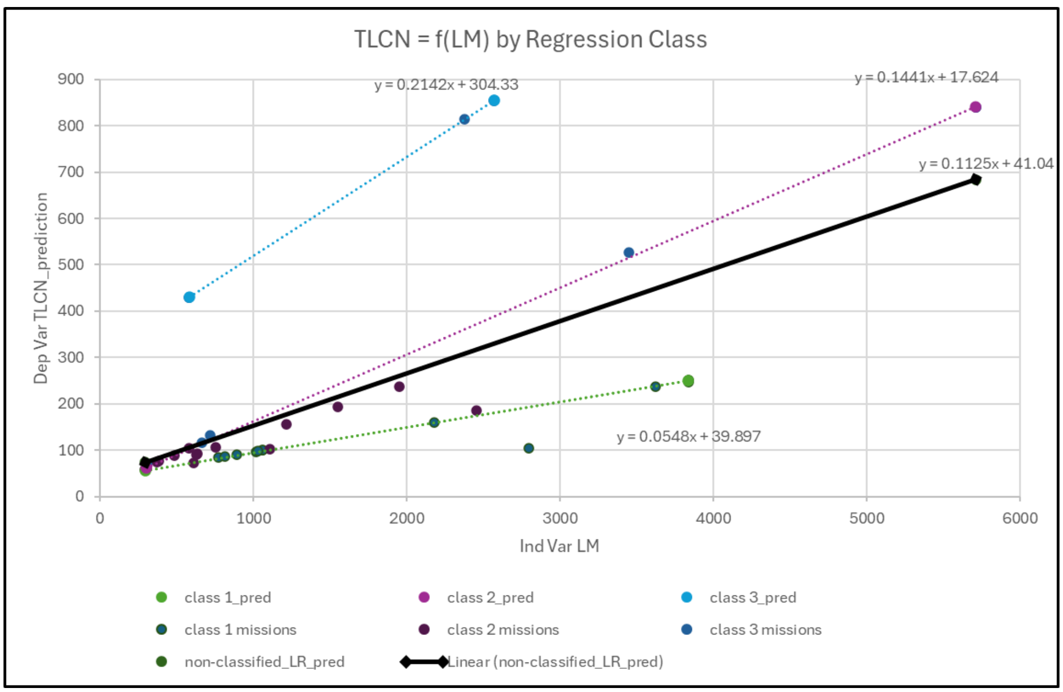
Zero Class Limit = 0.04

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
TLCN = f(LM)				

Class Regression Threshold	39%
Class Residual Threshold	18%

	Regression n	Residual n	Min %e	Max %e
Class 1	16	7	-14.9%	16.1%
Class 2	16	17	-34.0%	19.5%
Class 3	3	5	-14.0%	16.1%

Model	LR	CR	CR w/corr
mean	28.9%	-2.6%	-1.8%
|mean|	48.7%	17.1%	10.3%
st.dev	54.6%	23.1%	13.5%
st.dev | |	37.4%	15.5%	8.7%
co.var	52.9%	-11.4%	-7.44
co.var| |	130.2%	110.3%	1.30
|mean| + st.dev | |	86.1%	32.5%	19.0%



				CR Modeler [TLCN = f(LM)]			
Data ID	Mission	Ind Var LM	Dep Var TLCN	reg class	res class	Predict	Pred %e
34	Psyche	2800	115.9	1	1	103.4	-10.8%
24	Mars Recon Orbiter	2180	142.1	1	2	159.5	12.2%
26	MERA - Spirit	1063	86.8	1	2	100.1	15.3%
27	MERB - Opportunity	1063	86.8	1	2	100.1	15.3%
29	MSL - Curiosity	3839	255.7	1	0	250.2	-2.2%
17	M2020 - Perseverance	3839	263.9	1	2	247.7	-6.1%
13	Juno	3625	254.1	1	2	236.3	-7.0%
6	Deep Impact	1020	96.1	1	0	95.8	-0.3%
20	Mars Global Surveyor	1030.5	105.2	1	2	98.3	-6.5%
3	Contour	775	86.1	1	2	84.8	-1.6%
22	Mars Pathfinder	895	95.7	1	2	91.1	-4.8%
30	NEAR Shoemaker	818	90.9	1	2	87.0	-4.2%
16	Lunar Prospector	296.4	51.1	1	2	59.3	16.1%
25	MAVEN	2454	218.1	1	3	185.7	-14.9%
28	Messenger	1108	115.2	1	2	102.5	-11.1%
4	Dart	610	76.3	1	0	73.3	-3.9%
5	Dawn	1217.7	137.4	2	1	156.4	13.8%
15	Lucy	1550	167.4	2	1	193.1	15.3%
19	Mars Climate Orbiter	629	87.3	2	1	91.3	4.6%
1	2001 Mars Odyssey	758	99.7	2	1	105.6	5.9%
9	Genesis	636	90.4	2	1	92.1	1.9%
23	Mars Polar Lander	583	87.3	2	2	104.3	19.5%
32	Osiris-Rex	1955	224.5	2	1	237.8	5.9%
14	Ladee	375.5	75.5	2	2	74.8	-1.0%
7	Deep Space 1	486.3	87.3	2	0	87.7	0.4%
35	Stardust	385	78	2	2	76.1	-2.4%
33	Phoenix	664	127.7	2	2	115.9	-9.3%
2	Cassini	5712	809.3	2	0	840.5	3.9%
10	Grail-A	306	101.95	2	3	71.4	-29.9%
11	Grail-B	306	101.95	2	3	71.4	-29.9%
18	Magellan	3449	649.8	2	3	526.6	-19.0%
12	Insight	721	199.3	2	3	131.5	-34.0%
21	Mars Observer	2573	736.2	3	2	854.9	16.1%
8	Galileo	2380	946	3	2	813.9	-14.0%
31	New Horizons	585	416.8	3	0	429.6	3.1%

Interpretable (class-wise) CR Model

Class	n	TLCN = f(LM)	LM(min)	LM(max)	mean %e	stdev %e
1	16	TLCN(c1) = 39.90 + 0.05 * LM	296.4	3839	8.27%	5.38%
2	16	TLCN(c2) = 17.62 + 0.14 * LM	306	5712.00	12.30%	11.25%
3	3	TLCN(c3) = 304.33 + 0.21 * LM	585	2573.00	11.05%	7.00%
				Model	10.35%	8.68%

Class	n	Corrector = f(LM)	Correction = - Corrector
1	7	Corrector(c1) = -4.26 + 0.03 * LM	Corrector(c1) = 4.26 + -0.03 * LM
2	11	Corrector(c1) = -3.65 + 0.00 * LM	Corrector(c1) = 3.65 + 0.00 * LM
3	5	Corrector(c1) = -9.49 + 0.00 * LM	Corrector(c1) = 9.49 + 0.00 * LM

TLCP = f(LM)

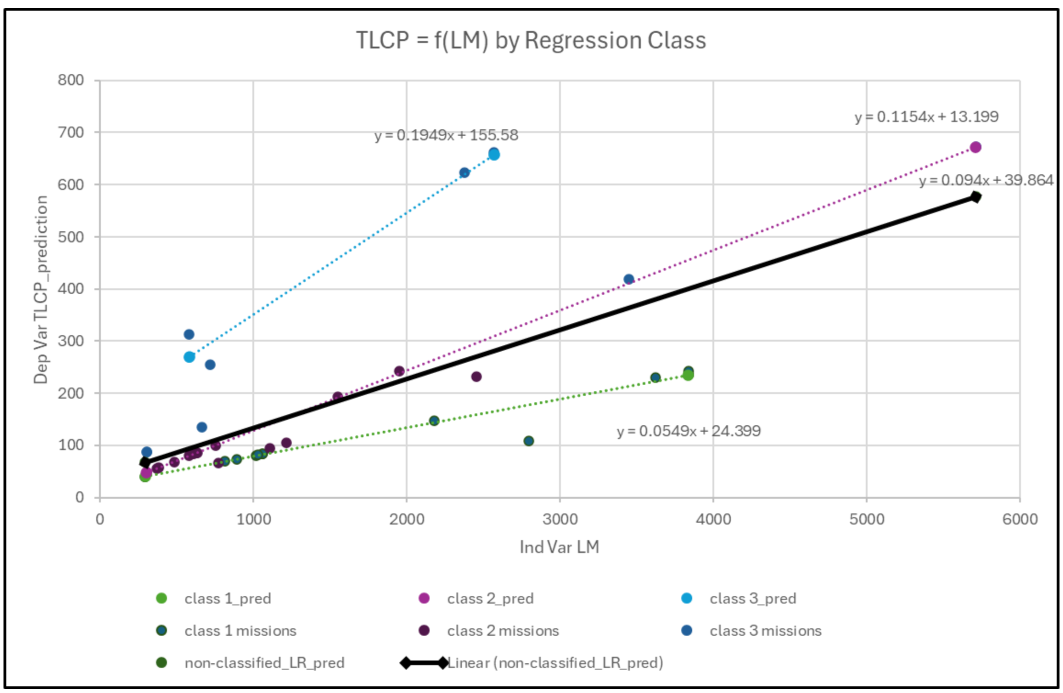
Zero Class Limit = 0.04

Model Summary (35 missions)					
Classified Linear Reg with Classified Residual Reg					
TLCP = f(LM)					

Class Regression Threshold	39%
Class Residual Threshold	22%

	Regression n	Residual n	Min %e	Max %e
Class 1	13	5	-11.8%	14.4%
Class 2	18	24	-22.5%	18.6%
Class 3	4	4	-13.0%	32.3%

Model	LR	CR	CR w/corr
mean	24.5%	-1.3%	-0.4%
|mean|	44.3%	18.2%	11.5%
st.dev	48.8%	24.0%	13.0%
st.dev | |	31.2%	15.3%	5.9%
co.var	50.3%	-5.5%	-33.64
co.var| |	142.0%	118.9%	1.14
|mean| + st.dev | |	75.5%	33.6%	17.3%



				CR Modeler [TLCP = f(LM)]			
Data ID	Mission	Ind Var LM	Dep Var TLCP	reg class	res class	Predict	Pred %e
34	Psyche	2800	112.7	1	1	109.2	-3.1%
24	Mars Recon Orbiter	2180	129.1	1	2	147.7	14.4%
26	MERA - Spirit	1063	77.5	1	2	83.5	7.7%
27	MERB - Opportunity	1063	77.5	1	2	83.5	7.7%
29	MSL - Curiosity	3839	244.1	1	0	235.3	-3.6%
6	Deep Impact	1020	86.8	1	2	81.0	-6.7%
13	Juno	3625	243.8	1	2	230.7	-5.4%
16	Lunar Prospector	296.4	43.4	1	2	39.4	-9.1%
17	M2020 - Perseverance	3839	257.8	1	2	243.0	-5.7%
22	Mars Pathfinder	895	80.5	1	2	73.8	-8.3%
20	Mars Global Surveyor	1030.5	88.9	1	2	81.6	-8.2%
30	NEAR Shoemaker	818	76.2	1	2	69.4	-8.9%
3	Contour	775	75.9	1	2	66.9	-11.8%
28	Messenger	1108	104.2	2	1	94.0	-9.8%
19	Mars Climate Orbiter	629	74.6	2	2	85.4	14.5%
25	MAVEN	2454	207.3	2	1	231.9	11.9%
4	Dart	610	74.6	2	2	83.1	11.4%
1	2001 Mars Odyssey	758	86.3	2	2	100.6	16.6%
9	Genesis	636	78.1	2	2	86.2	10.4%
23	Mars Polar Lander	583	74.6	2	2	80.0	7.2%
5	Dawn	1217.7	125.7	2	1	105.2	-16.3%
7	Deep Space 1	486.3	74.7	2	2	68.6	-8.2%
35	Stardust	385	66.9	2	2	56.6	-15.4%
15	Lucy	1550	163.6	2	2	194.0	18.6%
14	Ladee	375.5	71.6	2	2	55.5	-22.5%
32	Osiris-Rex	1955	217.6	2	2	241.8	11.1%
33	Phoenix	664	117.1	2	3	134.7	15.0%
2	Cassini	5712	679.7	2	0	672.4	-1.1%
18	Magellan	3449	501.1	2	2	418.0	-16.6%
10	Grail-A	306	97.8	2	3	87.4	-10.6%
11	Grail-B	306	97.8	2	3	87.4	-10.6%
12	Insight	721	192	3	1	254.0	32.3%
21	Mars Observer	2573	576.2	3	2	661.5	14.8%
8	Galileo	2380	716.8	3	2	623.4	-13.0%
31	New Horizons	585	357.1	3	3	313.1	-12.3%

Interpretable (class-wise) CR Model

Class	n	TLCP = f(LM)	LM(min)	LM(max)	mean %e	stdev %e
1	13	TLCP(c1) = 24.40 + 0.05 * LM	296.4	3839	7.74%	3.09%
2	18	TLCP(c2) = 13.20 + 0.12 * LM	306	5712.00	12.65%	4.87%
3	4	TLCP(c3) = 155.58 + 0.19 * LM	585	2573.00	18.12%	9.53%
				Model	11.45%	5.87%

Class	n	Corrector = f(LM)	Correction = - Corrector
1	5	Corrector(c1) = 32.69 + 0.01 * LM	Corrector(c1) = -32.69 + -0.01 * LM
2	22	Corrector(c1) = 2.01 + 0.00 * LM	Corrector(c1) = -2.01 + 0.00 * LM
3	4	Corrector(c1) = -33.83 + -0.02 * LM	Corrector(c1) = 33.83 + 0.02 * LM

TLCT = f(LM)

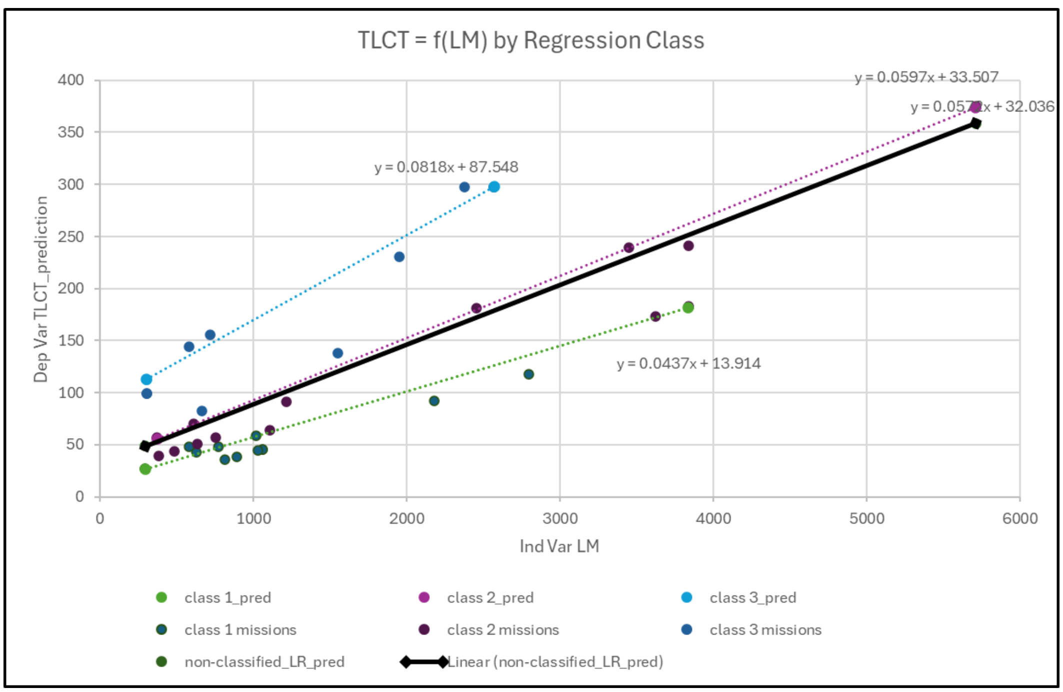
Zero Class Limit = 0.04

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
TLCT = f(LM)				

Class Regression Threshold	20%
Class Residual Threshold	10%

	Regression n	Residual n	Min %e	Max %e
Class 1	19	12	-19.6%	6.9%
Class 2	9	7	-7.3%	7.3%
Class 3	7	10	-33.7%	30.1%

Model	LR	CR	CR w/corr
mean	20.9%	-2.0%	-1.4%
|mean|	41.8%	14.9%	8.9%
st.dev	45.2%	19.1%	12.6%
st.dev | |	26.3%	11.8%	8.9%
co.var	46.2%	-10.3%	-9.21
co.var| |	158.9%	126.0%	1.42
|mean| + st.dev | |	68.1%	26.7%	17.8%



				CR Modeler [TLCT = f(LM)]			
Data ID	Mission	Ind Var LM	Dep Var TLCT	reg class	res class	Predict	Pred %e
16	Lunar Prospector	296.4	26	1	0	26.9	3.4%
26	MERA - Spirit	1063	50.6	1	1	45.6	-10.0%
27	MERB - Opportunity	1063	50.6	1	1	45.6	-10.0%
30	NEAR Shoemaker	818	44	1	1	35.4	-19.6%
22	Mars Pathfinder	895	47.1	1	1	38.6	-18.1%
24	Mars Recon Orbiter	2180	90	1	1	91.9	2.1%
20	Mars Global Surveyor	1030.5	52.6	1	1	44.2	-16.0%
34	Psyche	2800	112.7	1	1	117.6	4.4%
6	Deep Impact	1020	57.6	1	0	58.5	1.6%
3	Contour	775	49	1	0	47.8	-2.4%
19	Mars Climate Orbiter	629	45.3	1	2	42.6	-5.9%
23	Mars Polar Lander	583	45.3	1	3	48.4	6.9%
9	Genesis	636	48.7	1	3	50.9	4.6%
1	2001 Mars Odyssey	758	53.9	1	3	56.7	5.2%
28	Messenger	1108	69	1	2	63.6	-7.9%
35	Stardust	385	40.8	1	3	39.1	-4.2%
7	Deep Space 1	486.3	45.4	1	3	43.9	-3.4%
29	MSL - Curiosity	3839	194.7	1	2	183.0	-6.0%
13	Juno	3625	190.4	1	2	173.7	-8.8%
5	Dawn	1217.7	91.3	2	1	90.9	-0.4%
17	M2020 - Perseverance	3839	230.4	2	1	241.4	4.8%
25	MAVEN	2454	168.7	2	2	181.1	7.3%
18	Magellan	3449	235	2	0	239.2	1.8%
4	Dart	610	68.8	2	0	69.9	1.6%
14	Ladee	375.5	58.7	2	2	57.1	-2.7%
2	Cassini	5712	398.4	2	2	375.4	-5.8%
33	Phoenix	664	86.2	2	3	82.4	-4.4%
15	Lucy	1550	149.4	2	3	138.4	-7.3%
32	Osiris-Rex	1955	183.5	3	1	230.6	25.7%
10	Grail-A	306	76.4	3	1	99.4	30.1%
11	Grail-B	306	76.4	3	1	99.4	30.1%
21	Mars Observer	2573	293.1	3	0	298.0	1.7%
8	Galileo	2380	323.9	3	3	297.6	-8.1%
12	Insight	721	163.4	3	3	156.0	-4.5%
31	New Horizons	585	218	3	3	144.4	-33.7%

Interpretable (class-wise) CR Model

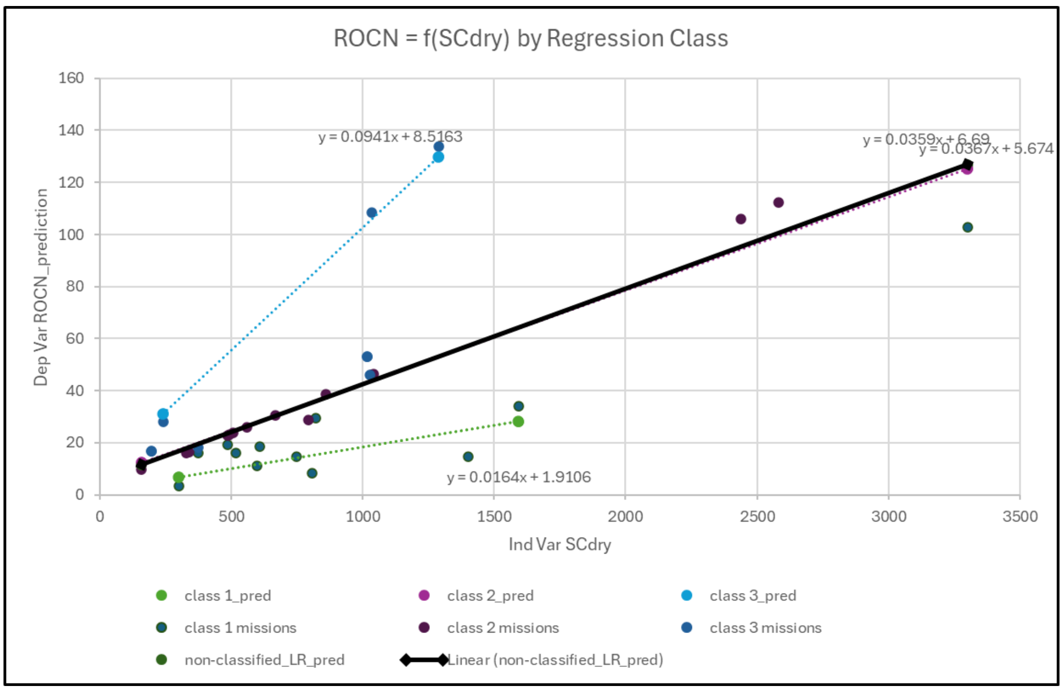
Class	n	TLCT = f(LM)	LM(min)	LM(max)	mean %e	stdev %e
1	19	TLCT(c1) = 13.91 + 0.04 * LM	296.4	3839	7.39%	5.30%
2	9	TLCT(c2) = 33.51 + 0.06 * LM	375.5	5712.00	4.01%	2.54%
3	7	TLCT(c3) = 87.55 + 0.08 * LM	306	2573.00	19.14%	13.78%
				Model	8.87%	8.89%

Class	n	Corrector = f(LM)	Correction = - Corrector
1	12	Corrector(c1) = 12.46 + 0.00 * LM	Corrector(c1) = -12.46 + 0.00 * LM
2	1	Corrector(c1) = -1.19 + 0.00 * LM	Corrector(c1) = 1.19 + 0.00 * LM
3	10	Corrector(c1) = -6.96 + 0.00 * LM	Corrector(c1) = 6.96 + 0.00 * LM

ROCN = f(SCdry)

Zero Class Limit = 0.00

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
ROCN = f(SCdry)				
Class Regression Threshold			32%	
Class Residual Threshold			17%	
	Regression n	Residual n	Min %e	Max %e
Class 1	8	9	-26.6%	57.5%
Class 2	24	20	-22.5%	15.4%
Class 3	3	6	-5.1%	8.3%
Model	LR	CR	CR w/corr	
mean	37.0%	-5.9%	-1.3%	
|mean|	57.5%	21.2%	8.6%	
st.dev	113.3%	36.6%	14.1%	
st.dev | |	104.2%	30.3%	11.2%	
co.var	32.6%	-16.1%	-11.25	
co.var| |	55.2%	69.9%	1.65	
|mean| + st.dev | |	161.6%	51.4%	19.7%	



				CR Modeler [ROCN = f(SCdry)]			
Data ID	Mission	Ind Var SCdry	Dep Var ROCN	reg class	res class	Predict	Pred %e
25	MAVEN	809	5.3892617	1	1	8.5	57.5%
35	Stardust	300	4.4897494	1	1	3.3	-26.4%
34	Psyche	1400	19.75556	1	1	14.5	-26.6%
33	Phoenix	597	11.158031	1	2	11.1	-0.7%
5	Dawn	747.1	16.085331	1	2	14.6	-9.5%
13	Juno	1593	33.817372	1	2	34.2	1.2%
12	Insight	608	16.256959	1	3	18.4	13.1%
23	Mars Polar Lander	519	16.446856	1	3	16.2	-1.6%
15	Lucy	821	27.322952	2	1	29.4	7.5%
7	Deep Space 1	373.7	14.977204	2	1	16.1	7.5%
29	MSL - Curiosity	3300	101.71944	2	1	102.9	1.1%
28	Messenger	485.2	19.219552	2	1	19.4	1.0%
22	Mars Pathfinder	795	29.590535	2	1	28.6	-3.3%
16	Lunar Prospector	158.7	9.9361111	2	1	9.7	-2.1%
20	Mars Global Surveyor	666.3	26.382322	2	2	30.4	15.4%
31	New Horizons	508	21.734541	2	2	23.7	8.9%
30	NEAR Shoemaker	487	22.837541	2	2	22.8	-0.2%
3	Contour	328	0	2	2	16.0	0.0%
32	Osiris-Rex	860	39.082815	2	2	38.7	-1.0%
24	Mars Recon Orbiter	1031	45.72816	2	2	46.0	0.6%
19	Mars Climate Orbiter	338	19.207168	2	2	16.4	-14.5%
2	Cassini	2581	107.24208	2	2	112.2	4.6%
4	Dart	560	28.269608	2	2	25.9	-8.4%
6	Deep Impact	1042	49.580925	2	2	46.5	-6.3%
9	Genesis	494	27.229646	2	2	23.1	-15.2%
17	M2020 - Perseverance	2439	111.00056	2	2	106.1	-4.4%
1	2001 Mars Odyssey	376.3	23.289069	2	2	18.1	-22.5%
21	Mars Observer	1028	52.206061	2	2	45.9	-12.1%
26	MERA - Spirit	1018	52.667785	2	3	53.2	1.0%
27	MERB - Opportunity	1018	54.686411	2	3	53.2	-2.7%
10	Grail-A	198	17.274809	2	3	16.8	-2.6%
11	Grail-B	198	17.274809	2	3	16.8	-2.6%
14	Ladee	240.7	29.2	3	2	28.1	-3.7%
8	Galileo	1289	123.68898	3	2	133.9	8.3%
18	Magellan	1035	114.12399	3	2	108.3	-5.1%

Interpretable (class-wise) CR Model

Class	n	ROCN = f(SCdry)	SCdry(min)	SCdry(max)	mean %e	stdev %e
1	8	ROCN(c1) = 1.91 + 0.02 * SCdry	300	1593	17.08%	19.44%
2	24	ROCN(c2) = 6.69 + 0.04 * SCdry	158.7	3300.00	6.07%	6.01%
3	3	ROCN(c3) = 8.52 + 0.09 * SCdry	240.7	1289.00	5.71%	2.32%
				Model	8.56%	11.17%

Class	n	Corrector = f(SCdry)	Correction = - Corrector
1	9	Corrector(c1) = 1.66 + 0.01 * SCdry	Corrector(c1) = -1.66 + -0.01 * SCdry
2	20	Corrector(c1) = 4.70 + -0.01 * SCdry	Corrector(c1) = -4.70 + 0.01 * SCdry
3	6	Corrector(c1) = -1.35 + -0.01 * SCdry	Corrector(c1) = 1.35 + 0.01 * SCdry

ROCP = f(SCdry)

Zero Class Limit = 0.00

Model Summary (35 missions)

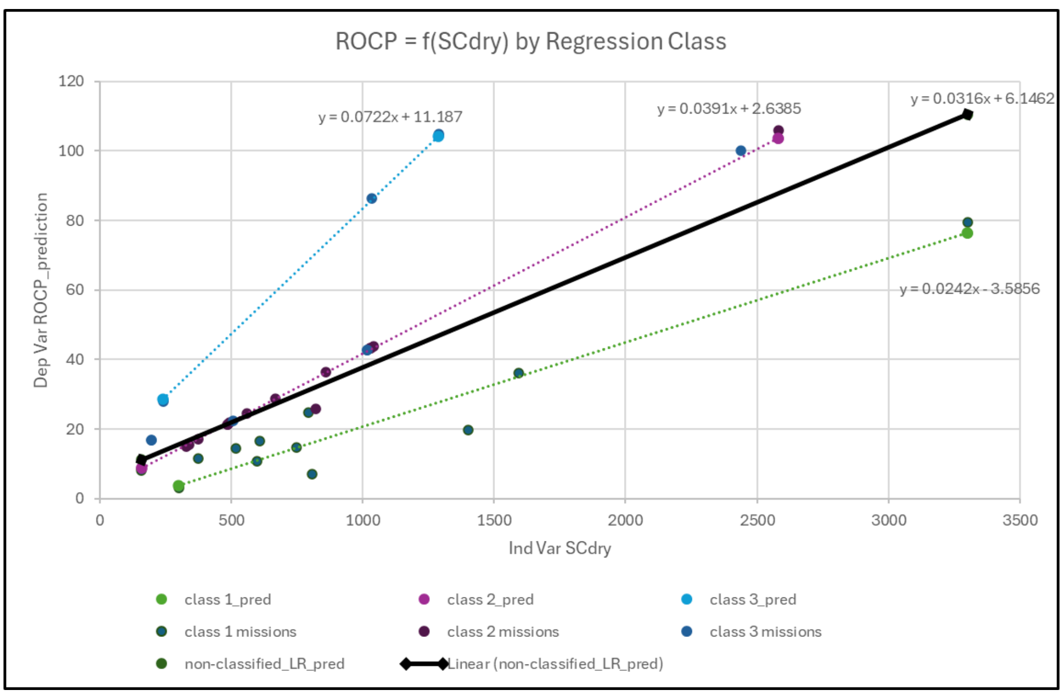
Classified Linear Reg with Classified Residual Reg

ROCP = f(SCdry)

Class Regression Threshold	32%
Class Residual Threshold	26%

	Regression n	Residual n	Min %e	Max %e
Class 1	10	4	-20.4%	36.8%
Class 2	22	26	-20.4%	25.5%
Class 3	3	5	-3.4%	3.3%

Model	LR	CR	CR w/corr
mean	34.1%	-2.4%	-0.9%
|mean|	56.2%	21.2%	8.4%
st.dev	107.5%	42.2%	11.9%
st.dev | |	97.5%	36.4%	8.3%
co.var	31.7%	-5.8%	-13.72
co.var| |	57.6%	58.3%	1.41
|mean| + st.dev | |	153.7%	57.6%	16.7%



				CR Modeler [ROCP = f(SCdry)]			
Data ID	Mission	Ind Var SCdry	Dep Var ROCP	reg class	res class	Predict	Pred %e
25	MAVEN	809	5.1442953	1	1	7.0	36.8%
35	Stardust	300	4.0075171	1	2	3.2	-20.4%
34	Psyche	1400	19.75556	1	1	19.8	0.1%
33	Phoenix	597	10.401554	1	2	10.8	3.4%
5	Dawn	747.1	15.348423	1	2	14.6	-5.0%
13	Juno	1593	32.54922	1	2	36.1	11.0%
12	Insight	608	15.983405	1	3	16.4	2.8%
23	Mars Polar Lander	519	14.261228	1	3	14.5	1.9%
7	Deep Space 1	373.7	12.980243	1	3	11.4	-11.9%
29	MSL - Curiosity	3300	82.039185	1	2	79.6	-3.0%
16	Lunar Prospector	158.7	8.5166667	2	2	8.2	-4.0%
22	Mars Pathfinder	795	25.234568	2	1	24.8	-1.8%
20	Mars Global Surveyor	666.3	22.83159	2	2	28.7	25.5%
28	Messenger	485.2	18.071919	2	2	21.4	18.2%
15	Lucy	821	27.206366	2	1	25.7	-5.4%
30	NEAR Shoemaker	487	19.792536	2	2	21.4	8.3%
19	Mars Climate Orbiter	338	16.65472	2	2	15.4	-7.5%
3	Contour	328	0	2	2	15.0	0.0%
2	Cassini	2581	95.308666	2	2	105.9	11.1%
24	Mars Recon Orbiter	1031	42.42703	2	2	43.4	2.2%
21	Mars Observer	1028	42.693939	2	2	43.3	1.3%
9	Genesis	494	24.645575	2	2	21.7	-11.9%
4	Dart	560	27.076797	2	2	24.4	-10.0%
32	Osiris-Rex	860	38.231337	2	2	36.5	-4.6%
6	Deep Impact	1042	45.150289	2	2	43.8	-2.9%
1	2001 Mars Odyssey	376.3	21.308907	2	2	17.0	-20.4%
26	MERA - Spirit	1018	47.768456	2	2	42.9	-10.3%
31	New Horizons	508	27.858339	2	2	22.3	-20.0%
17	M2020 - Perseverance	2439	106.66111	2	2	100.2	-6.1%
27	MERB - Opportunity	1018	49.599303	2	2	42.9	-13.6%
10	Grail-A	198	16.438931	2	3	16.9	2.5%
11	Grail-B	198	16.438931	2	3	16.9	2.5%
14	Ladee	240.7	27.74	3	2	28.0	0.9%
8	Galileo	1289	101.6733	3	2	105.0	3.3%
18	Magellan	1035	89.331536	3	2	86.3	-3.4%

Interpretable (class-wise) CR Model

Class	n	ROCP = f(SCdry)	SCdry(min)	SCdry(max)	mean %e	stdev %e
1	10	ROCP(c1) = -3.59 + 0.02 * SCdry	300	3300	9.63%	11.36%
2	22	ROCP(c2) = 2.64 + 0.04 * SCdry	158.7	2581.00	8.64%	7.14%
3	3	ROCP(c3) = 11.19 + 0.07 * SCdry	240.7	1289.00	2.52%	1.37%
				Model	8.40%	8.33%

Class	n	Corrector = f(SCdry)	Correction = - Corrector
1	4	Corrector(c1) = 6.82 + 0.00 * SCdry	Corrector(c1) = -6.82 + 0.00 * SCdry
2	26	Corrector(c1) = 0.87 + 0.00 * SCdry	Corrector(c1) = -0.87 + 0.00 * SCdry
3	5	Corrector(c1) = -7.04 + 0.00 * SCdry	Corrector(c1) = 7.04 + 0.00 * SCdry

ROCT = f(SCdry)

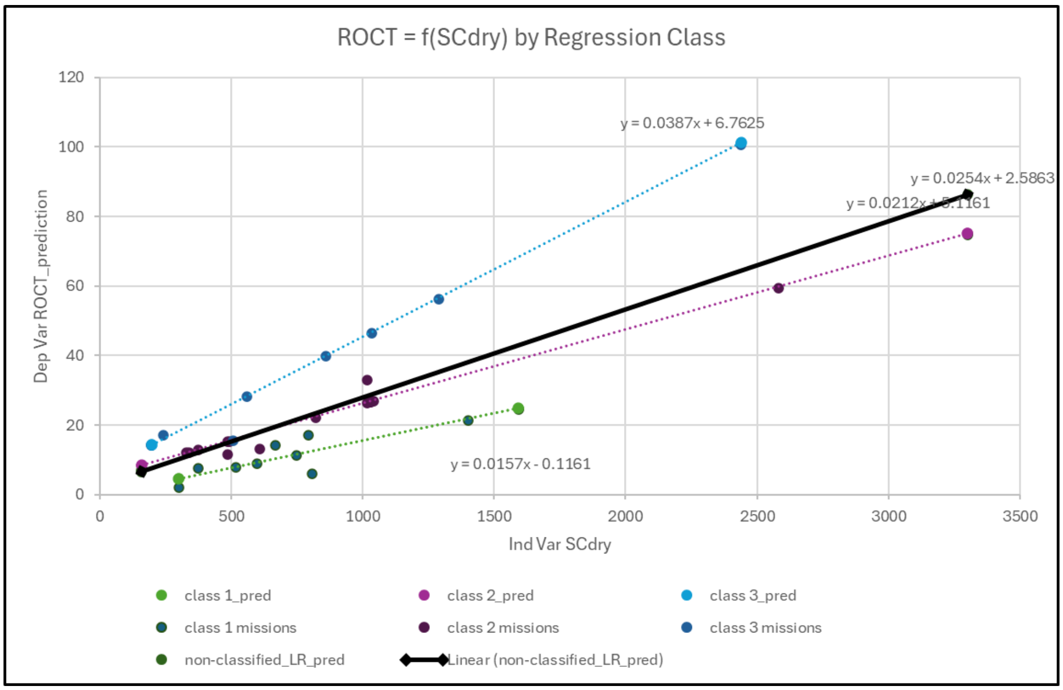
Zero Class Limit = 0.01

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
ROCT = f(SCdry)				

Class Regression Threshold	29%
Class Residual Threshold	19%

	Regression n	Residual n	Min %e	Max %e
Class 1	10	5	-21.6%	39.6%
Class 2	17	24	-19.9%	37.7%
Class 3	8	5	-26.0%	13.4%

Model	LR	CR	CR w/corr
mean	28.1%	-6.7%	-1.1%
|mean|	53.2%	21.7%	11.9%
st.dev	94.8%	39.2%	15.1%
st.dev | |	83.0%	33.2%	9.2%
co.var	29.7%	-17.1%	-13.99
co.var| |	64.1%	65.3%	1.27
|mean| + st.dev | |	136.2%	54.8%	21.0%



				CR Modeler [ROCT = f(SCdry)]			
Data ID	Mission	Ind Var SCdry	Dep Var ROCT	reg class	res class	Predict	Pred %e
25	MAVEN	809	4.3114094	1	1	6.0	39.6%
35	Stardust	300	2.6689066	1	1	2.1	-21.6%
33	Phoenix	597	7.9430052	1	2	8.9	12.6%
34	Psyche	1400	19.75556	1	2	21.4	8.5%
23	Mars Polar Lander	519	8.9064371	1	2	7.7	-13.2%
5	Dawn	747.1	12.435331	1	2	11.3	-9.3%
13	Juno	1593	27.281514	1	2	24.4	-10.4%
7	Deep Space 1	373.7	8.0987842	1	3	7.6	-6.7%
22	Mars Pathfinder	795	15.320988	1	3	17.1	11.5%
20	Mars Global Surveyor	666.3	14.298379	1	3	14.2	-0.9%
29	MSL - Curiosity	3300	67.469697	2	2	74.6	10.6%
16	Lunar Prospector	158.7	5.1708333	2	1	7.1	37.7%
12	Insight	608	14.342077	2	1	13.1	-8.7%
21	Mars Observer	1028	23.448485	2	2	26.6	13.5%
30	NEAR Shoemaker	487	12.400384	2	1	11.5	-7.4%
28	Messenger	485.2	13.824359	2	2	15.1	9.5%
2	Cassini	2581	63.259202	2	2	59.4	-6.1%
19	Mars Climate Orbiter	338	10.401224	2	2	12.0	15.6%
3	Contour	328	0	2	0	12.1	0.0%
6	Deep Impact	1042	31.647399	2	2	26.9	-15.0%
24	Mars Recon Orbiter	1031	31.792138	2	2	26.7	-16.1%
9	Genesis	494	16.893363	2	2	15.3	-9.3%
26	MERA - Spirit	1018	32.947987	2	2	26.4	-19.9%
15	Lucy	821	27.206366	2	2	22.2	-18.3%
1	2001 Mars Odyssey	376.3	14.334008	2	2	12.8	-10.4%
27	MERB - Opportunity	1018	34.210801	2	3	33.0	-3.4%
31	New Horizons	508	19.396526	2	2	15.6	-19.5%
32	Osiris-Rex	860	35.038297	3	2	39.7	13.4%
17	M2020 - Perseverance	2439	102.60556	3	2	100.7	-1.8%
18	Magellan	1035	46.092318	3	2	46.5	0.9%
4	Dart	560	26.957516	3	2	28.2	4.5%
8	Galileo	1289	57.654248	3	2	56.3	-2.3%
10	Grail-A	198	13.443702	3	2	14.2	5.5%
11	Grail-B	198	13.443702	3	2	14.2	5.5%
14	Ladee	240.7	22.942857	3	3	17.0	-26.0%

Interpretable (class-wise) CR Model

Class	n	ROCT = f(SCdry)	SCdry(min)	SCdry(max)	mean %e	stdev %e
1	10	ROCT(c1) = -0.12 + 0.02 * SCdry	300	1593	13.43%	10.58%
2	17	ROCT(c2) = 5.12 + 0.02 * SCdry	158.7	3300.00	13.00%	8.45%
3	8	ROCT(c3) = 6.76 + 0.04 * SCdry	198	2439.00	7.48%	8.43%
				Model	11.86%	9.15%

Class	n	Corrector = f(SCdry)	Correction = - Corrector
1	5	Corrector(c1) = 0.11 + 0.01 * SCdry	Corrector(c1) = -0.11 + -0.01 * SCdry
2	23	Corrector(c1) = 0.23 + 0.00 * SCdry	Corrector(c1) = -0.23 + 0.00 * SCdry
3	5	Corrector(c1) = 0.78 + -0.01 * SCdry	Corrector(c1) = -0.78 + 0.01 * SCdry

Appendix D

S1 = f(scdry)

Zero class limit = 0.01

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
S1 = f(SCdry)				

Class Regression Threshold	22%
Class Residual Threshold	8%

	Regression n	Residual n	Min %e	Max %e
Class 1	9	10	-5.3%	5.5%
Class 2	23	15	-6.9%	7.3%
Class 3	3	9	-3.5%	11.7%

Model	LR	CR	CR w/corr
mean	9.0%	-1.6%	0.3%
|mean|	22.2%	10.8%	3.4%
st.dev	31.2%	13.6%	4.3%
st.dev | |	23.4%	8.3%	2.5%
co.var	28.8%	-11.4%	15.11
co.var| |	95.1%	129.1%	1.25
|mean| + st.dev | |	45.6%	19.1%	5.9%

				CR Modeler [S1 = f(SCdry)]			
Data ID	Mission	Ind Var SCdry	Dep Var S1	reg class	res class	Predict	Pred %e
20	Mars Global Surveyor	666.3	25.6	1	1	27.0	5.5%
30	NEAR Shoemaker	487	25.9	1	1	24.6	-4.9%
35	Stardust	300	28.633333	1	2	29.5	2.9%
26	MERA - Spirit	1018	37.833333	1	2	39.8	5.3%
27	MERB - Opportunity	1018	38.766667	1	2	39.8	2.7%
22	Mars Pathfinder	795	38.666667	1	2	36.6	-5.3%
7	Deep Space 1	373.7	37.3	1	3	38.3	2.8%
16	Lunar Prospector	158.7	34.766667	1	3	35.4	1.8%
29	MSL - Curiosity	3300	72.866667	1	2	72.8	-0.1%
19	Mars Climate Orbiter	338	37.866667	2	1	39.6	4.7%
24	Mars Recon Orbiter	1031	46.966667	2	1	49.1	4.6%
21	Mars Observer	1028	49.5	2	1	49.1	-0.8%
1	2001 Mars Odyssey	376.3	41.6	2	1	40.2	-3.5%
17	M2020 - Perseverance	2439	69.6	2	1	68.4	-1.7%
3	Contour	328	41.6	2	1	39.5	-5.1%
9	Genesis	494	44.766667	2	1	41.8	-6.7%
23	Mars Polar Lander	519	46.266667	2	2	49.7	7.3%
2	Cassini	2581	77.6	2	2	80.4	3.6%
33	Phoenix	597	48.7	2	2	50.8	4.4%
32	Osiris-Rex	860	53.033333	2	2	54.7	3.2%
6	Deep Impact	1042	57.233333	2	0	57.0	-0.3%
10	Grail-A	198	44.466667	2	2	44.9	0.9%
11	Grail-B	198	44.466667	2	2	44.9	0.9%
31	New Horizons	508	50.4	2	2	49.5	-1.8%
34	Psyche	1400	66.366667	2	2	62.8	-5.4%
15	Lucy	821	58.2	2	2	54.2	-6.9%
13	Juno	1593	73.266667	2	3	72.6	-0.9%
18	Magellan	1035	65	2	3	64.7	-0.4%
28	Messenger	485.2	55.866667	2	3	56.9	1.9%
4	Dart	560	57.633333	2	3	58.0	0.6%
5	Dawn	747.1	61.733333	2	3	60.6	-1.8%
25	MAVEN	809	63	2	3	61.5	-2.4%
12	Insight	608	70.1	3	1	78.3	11.7%
14	Ladee	240.7	66.166667	3	3	63.8	-3.5%
8	Galileo	1289	146.66667	3	2	141.7	-3.4%

Interpretable (class-wise) CR Model

Class	n	S1 = f(SCdry)	SCdry(min)	SCdry(max)	mean %e	stdev %e
1	9	S1(c1) = 25.31 + 0.01 * SCdry	158.7	3300	3.47%	1.88%
2	23	S1(c2) = 42.12 + 0.01 * SCdry	198	2581.00	3.03%	2.23%
3	3	S1(c3) = 36.40 + 0.08 * SCdry	240.7	1289.00	6.21%	4.77%
				Model	3.42%	2.49%

Class	n	Corrector = f(SCdry)	Correction = - Corrector
1	10	Corrector(c1) = 7.12 + 0.00 * SCdry	Corrector(c1) = -7.12 + 0.00 * SCdry
2	14	Corrector(c1) = 0.19 + 0.00 * SCdry	Corrector(c1) = -0.19 + 0.00 * SCdry
3	9	Corrector(c1) = -7.90 + 0.00 * SCdry	Corrector(c1) = 7.90 + 0.00 * SCdry

S1 = f(TDCT)

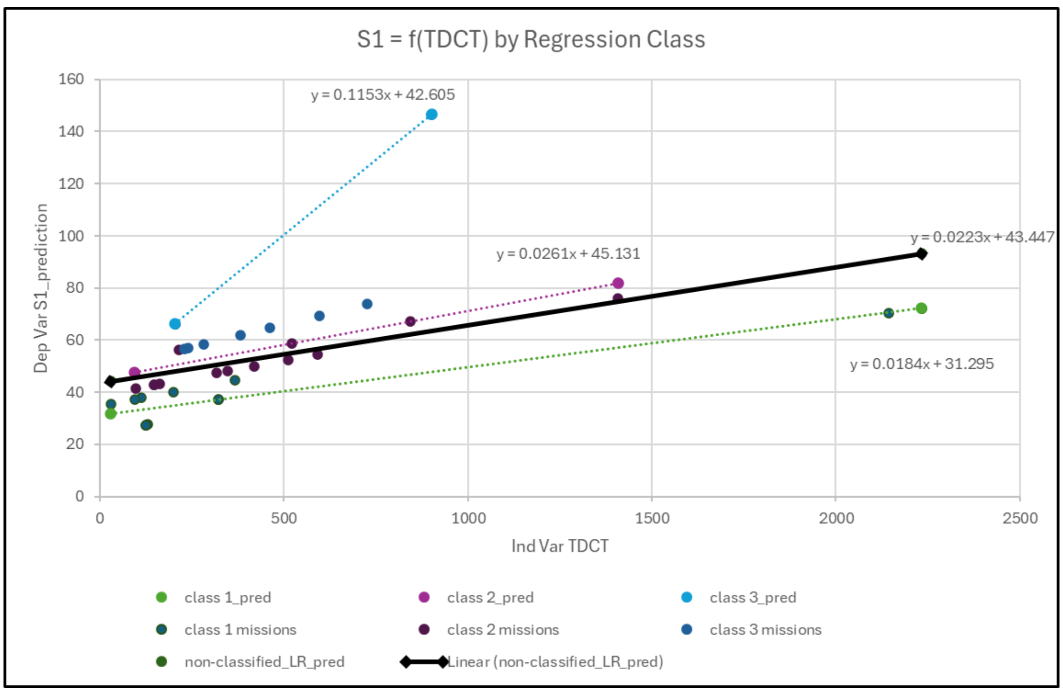
Zero class limit = 0.00

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
S1 = f(TDCT)				

Class Regression Threshold	20%
Class Residual Threshold	5%

	Regression n	Residual n	Min %e	Max %e
Class 1	12	13	-4.1%	7.6%
Class 2	21	9	-5.3%	6.5%
Class 3	2	13	-0.1%	0.0%

Model	LR	CR	CR w/corr
mean	8.2%	-1.5%	0.2%
|mean|	21.7%	10.3%	2.8%
st.dev	28.6%	12.7%	3.6%
st.dev | |	20.0%	7.3%	2.1%
co.var	28.6%	-11.9%	14.78
co.var| |	108.6%	142.5%	1.26
|mean| + st.dev | |	41.7%	17.6%	5.0%



				CR Modeler [S1 = f(TDCT)]			
Data ID	Mission	Ind Var TDCT	Dep Var S1	reg class	res class	Predict	Pred %e
20	Mars Global Surveyor	130.7	25.6	1	1	27.6	7.6%
30	NEAR Shoemaker	124.9	25.9	1	1	27.4	6.0%
35	Stardust	126.4	28.633333	1	1	27.5	-4.0%
17	M2020 - Perseverance	2232.2	69.6	1	2	72.0	3.4%
26	MERA - Spirit	321.5	37.833333	1	2	37.2	-1.8%
27	MERB - Opportunity	321.5	38.766667	1	2	37.2	-4.1%
16	Lunar Prospector	30.6	34.766667	1	3	35.5	2.2%
29	MSL - Curiosity	2142.4	72.866667	1	2	70.3	-3.5%
1	2001 Mars Odyssey	366.2	41.6	1	3	44.7	7.5%
22	Mars Pathfinder	199.2	38.666667	1	3	40.2	3.8%
7	Deep Space 1	114.2	37.3	1	3	37.8	1.4%
19	Mars Climate Orbiter	94.5	37.866667	1	3	37.3	-1.5%
24	Mars Recon Orbiter	419.2	46.966667	2	1	50.0	6.5%
21	Mars Observer	511.2	49.5	2	1	52.5	6.0%
3	Contour	96.8	41.6	2	1	41.5	-0.2%
32	Osiris-Rex	591.6	53.033333	2	1	54.6	2.9%
9	Genesis	163	44.766667	2	1	43.3	-3.4%
10	Grail-A	147.95	44.466667	2	1	42.9	-3.6%
11	Grail-B	147.95	44.466667	2	1	42.9	-3.6%
33	Phoenix	318.2	48.7	2	1	47.4	-2.8%
31	New Horizons	347.7	50.4	2	1	48.1	-4.5%
23	Mars Polar Lander	94.5	46.266667	2	2	47.6	2.9%
2	Cassini	1408.5	77.6	2	1	76.1	-1.9%
15	Lucy	523.1	58.2	2	2	58.7	0.9%
34	Psyche	844.5	66.366667	2	2	67.1	1.1%
28	Messenger	216.2	55.866667	2	3	56.1	0.5%
6	Deep Impact	230.9	57.233333	2	3	56.6	-1.0%
4	Dart	240.6	57.633333	2	3	57.0	-1.1%
18	Magellan	463.2	65	2	3	64.8	-0.3%
25	MAVEN	382.4	63	2	3	62.0	-1.6%
13	Juno	726.2	73.266667	2	3	74.0	1.0%
12	Insight	596.4	70.1	2	3	69.5	-0.9%
5	Dawn	282.9	61.733333	2	3	58.5	-5.3%
14	Ladee	204.3	66.166667	3	2	66.2	0.0%
8	Galileo	902.3	146.66667	3	2	146.6	-0.1%

Interpretable (class-wise) CR Model

Class	n	S1 = f(TDCT)	TDCT(min)	TDCT(max)	mean %e	stdev %e
1	12	S1(c1) = 31.29 + 0.02 * TDCT	30.6	2232.2	3.91%	2.15%
2	21	S1(c2) = 45.13 + 0.03 * TDCT	94.5	1408.50	2.47%	1.90%
3	2	S1(c3) = 42.60 + 0.12 * TDCT	204.3	902.30	0.05%	0.04%
				Model	2.83%	2.14%

Class	n	Corrector = f(TDCT)	Correction = - Corrector
1	13	Corrector(c1) = 6.18 + 0.00 * TDCT	Corrector(c1) = -6.18 + 0.00 * TDCT
2	9	Corrector(c1) = -0.02 + 0.00 * TDCT	Corrector(c1) = 0.02 + 0.00 * TDCT
3	13	Corrector(c1) = -3.41 + -0.01 * TDCT	Corrector(c1) = 3.41 + 0.01 * TDCT

S1 = f(TDCN)

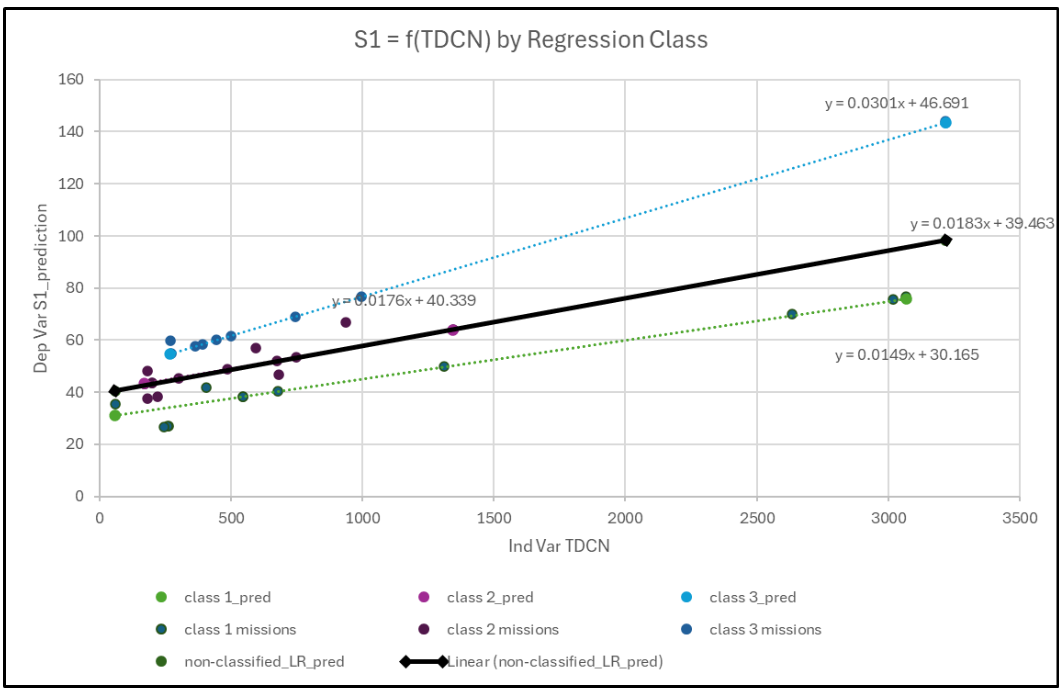
Zero class limit = 0.00

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
S1 = f(TDCN)				

Class Regression Threshold	10%
Class Residual Threshold	5%

	Regression n	Residual n	Min %e	Max %e
Class 1	15	5	-6.5%	8.0%
Class 2	9	23	-1.7%	4.4%
Class 3	11	7	-9.6%	4.7%

Model	LR	CR	CR w/corr
mean	6.1%	-0.9%	0.3%
|mean|	20.9%	6.7%	2.8%
st.dev	26.8%	10.5%	3.6%
st.dev | |	17.4%	8.1%	2.2%
co.var	22.6%	-8.4%	11.98
co.var| |	119.9%	83.0%	1.28
|mean| + st.dev | |	38.3%	14.8%	5.0%



				CR Modeler [S1 = f(TDCN)]			
Data ID	Mission	Ind Var TDCN	Dep Var S1	reg class	res class	Predict	Pred %e
20	Mars Global Surveyor	263.9	25.6	1	1	27.0	5.6%
30	NEAR Shoemaker	257.9	25.9	1	1	27.0	4.1%
35	Stardust	244	28.633333	1	1	26.8	-6.5%
29	MSL - Curiosity	3068.5	72.866667	1	2	76.6	5.1%
26	MERA - Spirit	547.2	37.833333	1	2	38.3	1.2%
21	Mars Observer	1309.3	49.5	1	2	49.8	0.7%
27	MERB - Opportunity	547.2	38.766667	1	2	38.3	-1.3%
17	M2020 - Perseverance	2635.5	69.6	1	2	70.0	0.5%
1	2001 Mars Odyssey	677	41.6	1	2	40.2	-3.3%
2	Cassini	3019.5	77.6	1	2	75.8	-2.3%
22	Mars Pathfinder	407.8	38.666667	1	3	41.7	8.0%
16	Lunar Prospector	60.2	34.766667	1	3	35.4	1.8%
7	Deep Space 1	221.2	37.3	1	3	38.3	2.8%
19	Mars Climate Orbiter	183.25	37.866667	1	3	37.6	-0.6%
24	Mars Recon Orbiter	681.6	46.966667	1	3	46.8	-0.5%
31	New Horizons	676.2	50.4	2	2	52.2	3.5%
3	Contour	172	41.6	2	2	43.2	3.9%
9	Genesis	300.5	44.766667	2	2	45.5	1.6%
32	Osiris-Rex	747.5	53.033333	2	2	53.4	0.8%
33	Phoenix	484.8	48.7	2	2	48.8	0.1%
18	Magellan	1343.8	65	2	2	64.1	-1.4%
10	Grail-A	200.4	44.466667	2	2	43.7	-1.7%
11	Grail-B	200.4	44.466667	2	2	43.7	-1.7%
23	Mars Polar Lander	183.25	46.266667	2	3	48.3	4.4%
15	Lucy	594.1	58.2	3	1	57.1	-2.0%
34	Psyche	935.9	66.366667	3	1	66.9	0.8%
28	Messenger	363.3	55.866667	3	2	57.5	2.9%
6	Deep Impact	390.5	57.233333	3	2	58.3	1.9%
13	Juno	995.9	73.266667	3	2	76.7	4.7%
4	Dart	274.8	57.633333	3	2	54.8	-4.9%
25	MAVEN	499.5	63	3	2	61.6	-2.2%
5	Dawn	444.7	61.733333	3	2	60.0	-2.8%
12	Insight	745.8	70.1	3	2	69.1	-1.4%
14	Ladee	269.8	66.166667	3	3	59.8	-9.6%
8	Galileo	3218.2	146.66667	3	2	144.1	-1.8%

Interpretable (class-wise) CR Model

Class	n	S1 = f(TDCN)	TDCN(min)	TDCN(max)	mean %e	stdev %e
1	15	S1(c1) = 30.16 + 0.01 * TDCN	60.2	3068.5	2.94%	2.41%
2	9	S1(c2) = 40.34 + 0.02 * TDCN	172	1343.80	2.13%	1.45%
3	11	S1(c3) = 46.69 + 0.03 * TDCN	269.8	3218.20	3.17%	2.47%
				Model	2.81%	2.20%

Class	n	Corrector = f(TDCN)	Correction = - Corrector
1	5	Corrector(c1) = 6.72 + 0.00 * TDCN	Corrector(c1) = -6.72 + 0.00 * TDCN
2	23	Corrector(c1) = 0.20 + 0.00 * TDCN	Corrector(c1) = -0.20 + 0.00 * TDCN
3	7	Corrector(c1) = -4.13 + 0.00 * TDCN	Corrector(c1) = 4.13 + 0.00 * TDCN

S1 = f(TDCP)

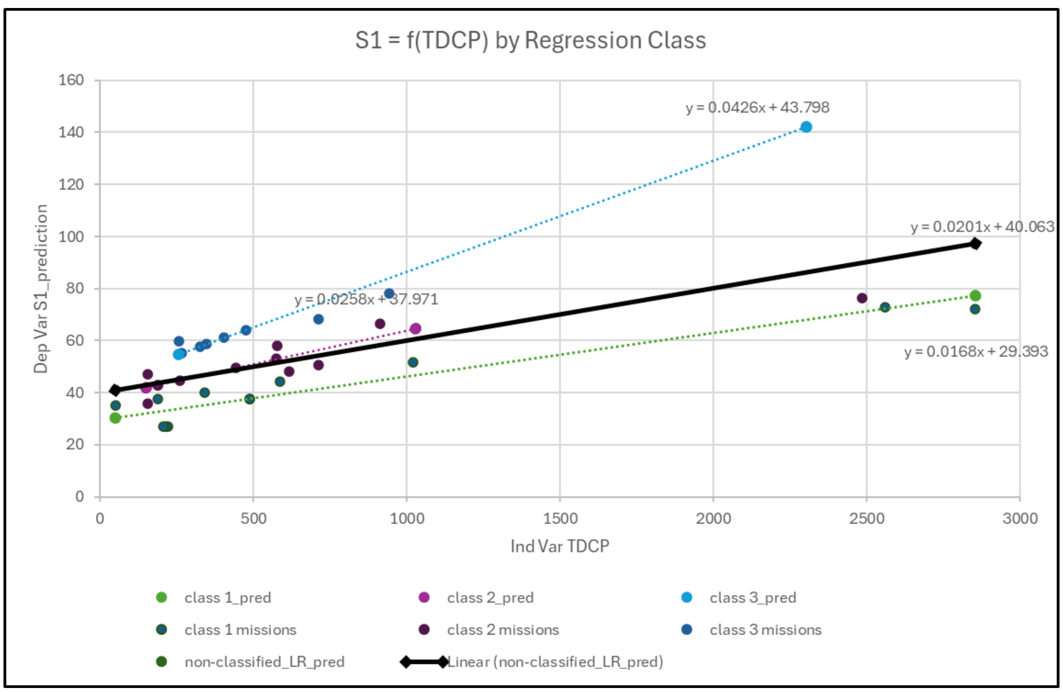
Zero class limit = 0.00

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
S1 = f(TDCP)				

Class Regression Threshold	15%
Class Residual Threshold	5%

	Regression n	Residual n	Min %e	Max %e
Class 1	13	9	-6.1%	6.4%
Class 2	13	16	-5.0%	5.1%
Class 3	9	10	-9.7%	6.8%

Model	LR	CR	CR w/corr
mean	6.5%	-0.9%	0.2%
|mean|	21.0%	7.6%	3.0%
st.dev	27.0%	10.4%	3.8%
st.dev | |	17.9%	7.0%	2.3%
co.var	24.2%	-8.6%	15.85
co.var| |	117.4%	108.5%	1.26
|mean| + st.dev | |	38.8%	14.6%	5.3%



				CR Modeler [S1 = f(TDCP)]			
Data ID	Mission	Ind Var TDCP	Dep Var S1	reg class	res class	Predict	Pred %e
20	Mars Global Surveyor	222.5	25.6	1	1	27.1	5.9%
30	NEAR Shoemaker	216.2	25.9	1	1	27.0	4.3%
35	Stardust	208.3	28.633333	1	1	26.9	-6.1%
29	MSL - Curiosity	2853.8	72.866667	1	1	72.1	-1.1%
26	MERA - Spirit	489.9	37.833333	1	2	37.7	-0.3%
17	M2020 - Perseverance	2558.7	69.6	1	2	72.9	4.7%
27	MERB - Opportunity	489.9	38.766667	1	2	37.7	-2.7%
1	2001 Mars Odyssey	586.3	41.6	1	3	44.3	6.4%
21	Mars Observer	1021.2	49.5	1	3	51.6	4.3%
22	Mars Pathfinder	342.5	38.666667	1	3	40.1	3.8%
16	Lunar Prospector	51	34.766667	1	3	35.2	1.2%
7	Deep Space 1	188.7	37.3	1	3	37.5	0.6%
2	Cassini	2484.1	77.6	1	3	76.4	-1.5%
19	Mars Climate Orbiter	156.2	37.866667	2	1	36.0	-5.0%
24	Mars Recon Orbiter	618.1	46.966667	2	1	48.1	2.3%
3	Contour	150.8	41.6	2	2	41.9	0.7%
32	Osiris-Rex	712.2	53.033333	2	1	50.5	-4.7%
31	New Horizons	576.3	50.4	2	2	53.0	5.1%
9	Genesis	260	44.766667	2	2	44.8	0.0%
33	Phoenix	442.8	48.7	2	2	49.5	1.7%
10	Grail-A	190.85	44.466667	2	2	42.9	-3.4%
11	Grail-B	190.85	44.466667	2	2	42.9	-3.4%
18	Magellan	1030.2	65	2	2	64.8	-0.3%
23	Mars Polar Lander	156.2	46.266667	2	3	47.0	1.5%
15	Lucy	579.2	58.2	2	3	58.0	-0.4%
34	Psyche	912.5	66.366667	2	3	66.6	0.4%
28	Messenger	326.4	55.866667	3	2	57.8	3.4%
6	Deep Impact	349.4	57.233333	3	2	58.8	2.7%
13	Juno	943.9	73.266667	3	1	78.2	6.8%
4	Dart	267.9	57.633333	3	2	55.3	-4.1%
25	MAVEN	475.6	63	3	2	64.2	1.9%
5	Dawn	404.6	61.733333	3	2	61.1	-1.0%
12	Insight	712.9	70.1	3	1	68.3	-2.5%
14	Ladee	257.6	66.166667	3	3	59.7	-9.7%
8	Galileo	2303.7	146.66667	3	2	142.4	-2.9%

Interpretable (class-wise) CR Model

Class	n	S1 = f(TDCP)	TDCP(min)	TDCP(max)	mean %e	stdev %e
1	13	S1(c1) = 29.39 + 0.02 * TDCP	51	2853.8	3.29%	2.20%
2	13	S1(c2) = 37.97 + 0.03 * TDCP	150.8	1030.20	2.23%	1.90%
3	9	S1(c3) = 43.80 + 0.04 * TDCP	257.6	2303.70	3.88%	2.73%
				Model	3.05%	2.28%

Class	n	Corrector = f(TDCP)	Correction = - Corrector
1	9	Corrector(c1) = 6.07 + 0.00 * TDCP	Corrector(c1) = -6.07 + 0.00 * TDCP
2	16	Corrector(c1) = -0.01 + 0.00 * TDCP	Corrector(c1) = 0.01 + 0.00 * TDCP
3	10	Corrector(c1) = -4.91 + 0.00 * TDCP	Corrector(c1) = 4.91 + 0.00 * TDCP

Appendix E

			Predicted								Predicted	Average		1/k weighting	
Ref	Mission element	TMCT	TMCT	TMCT %e	Sum(costs)	Sum %e	TDCT	TLCT	TOCT	S3	TOCT*	TMCT*	Ave %e	Weighted(model accuracy)	
1	2001 Mars Odyssey	680.2	654.1	-3.8%	665.8	-2%	305.6	51.6	308.6	24.04	12.8	660.0	-3.0%	659.2	-3.1%
2	Cassini	3206.8	3140.8	-2.1%	3178.3	-1%	1639.7	354.1	1184.5	19.93	59.4	3159.5	-1.5%	3157.0	-1.6%
3	Contour	145.8	173.3	18.9%	141.0	-3%	92.3	48.7	0.0	0.00	12.1	157.1	7.8%	159.3	9.3%
4	Dart	332	395.0	19.0%	350.0	5%	243.9	82.5	23.6	0.84	28.2	372.5	12.2%	375.6	13.1%
5	Dawn	505.6	573.9	13.5%	507.4	0%	283.8	98.3	125.2	11.10	11.3	540.6	6.9%	545.1	7.8%
6	Deep Impact	352.5	290.2	-17.7%	573.6	63%	295.1	47.5	230.9	8.58	26.9	431.9	22.5%	412.7	17.1%
7	Deep Space 1	180.6	217.0	20.2%	173.3	-4%	111.3	38.2	23.8	3.15	7.6	195.2	8.1%	198.1	9.7%
8	Galileo	1783.1	1504.2	-15.6%	2063.5	16%	977.0	301.8	784.7	13.93	56.3	1783.8	0.0%	1745.9	-2.1%
9	Genesis	264	280.9	6.4%	238.9	-10%	145.2	46.3	47.4	3.10	15.3	259.9	-1.6%	262.8	-0.5%
10	Grail-A	243.65	234.1	-3.9%	225.1	-8%	140.6	66.5	18.0	1.27	14.2	229.6	-5.8%	230.2	-5.5%
11	Grail-B	243.65	234.1	-3.9%	225.1	-8%	140.6	66.5	18.0	1.27	14.2	229.6	-5.8%	230.2	-5.5%
12	Insight	822.8	641.5	-22.0%	753.6	-8%	519.9	173.3	60.4	4.62	13.1	697.6	-15.2%	690.0	-16.1%
13	Juno	1304.5	1382.3	6.0%	1251.7	-4%	701.9	208.8	341.0	13.95	24.4	1317.0	1.0%	1325.9	1.6%
14	Ladee	279.2	339.9	21.7%	252.2	-10%	180.1	61.7	10.4	0.61	17.0	296.1	6.0%	302.0	8.2%
15	Lucy	999.2	911.3	-8.8%	806.5	-19%	586.9	141.7	77.9	3.50	22.2	858.9	-14.0%	866.0	-13.3%
16	Lunar Prospector	63.9	76.2	19.2%	73.3	15%	34.9	27.3	11.1	1.56	7.1	74.7	16.9%	74.9	17.2%
17	M2020 - Perseveranc	2880.6	2624.9	-8.9%	2828.0	-2%	2108.2	241.8	478.0	4.75	100.7	2726.4	-5.4%	2712.6	-5.8%
18	Magellan	864.8	849.0	-1.8%	953.7	10%	441.5	259.0	253.1	5.44	46.5	901.3	4.2%	894.2	3.4%
19	Mars Climate Orbiter	147.95	168.2	13.7%	157.6	6%	98.9	49.3	9.4	0.78	12.0	162.9	10.1%	163.6	10.6%
20	Mars Global Surveyo	303	289.6	-4.4%	320.4	6%	123.7	55.1	141.6	9.99	14.2	305.0	0.7%	302.9	0.0%
21	Mars Observer	825.5	842.3	2.0%	846.2	3%	564.3	257.8	24.1	0.90	26.6	844.3	2.3%	844.0	2.2%
22	Mars Pathfinder	258.8	211.6	-18.2%	237.5	-8%	182.4	41.2	13.9	0.81	17.1	224.6	-13.2%	222.8	-13.9%
23	Mars Polar Lander	147.95	123.7	-16.4%	156.5	6%	101.8	47.6	7.1	0.92	7.7	140.1	-5.3%	137.9	-6.8%
24	Mars Recon Orbiter	1072.2	1056.7	-1.4%	976.1	-9%	387.9	91.8	496.3	18.61	26.7	1016.4	-5.2%	1021.9	-4.7%
25	MAVEN	771.2	773.1	0.2%	615.1	-20%	412.7	140.2	62.2	10.33	6.0	694.1	-10.0%	704.8	-8.6%
26	MERA - Spirit	483.3	492.1	1.8%	510.2	6%	284.2	46.9	179.1	6.79	26.4	501.1	3.7%	499.9	3.4%
27	MERB - Opportunity	595.6	610.7	2.5%	824.4	38%	284.2	46.9	493.3	14.93	33.0	717.5	20.5%	703.0	18.0%
28	Messenger	457.8	448.9	-1.9%	427.2	-7%	188.5	76.1	162.7	10.75	15.1	438.1	-4.3%	439.5	-4.0%
29	MSL - Curiosity	3046	3013.8	-1.1%	3143.0	3%	2034.7	189.3	919.0	12.32	74.6	3078.4	1.1%	3069.6	0.8%
30	NEAR Shoemaker	230.8	239.1	3.6%	245.1	6%	141.3	45.9	57.8	5.04	11.5	242.1	4.9%	241.7	4.7%
31	New Horizons	889.7	926.0	4.1%	838.9	-6%	370.3	184.8	283.8	18.17	15.6	882.5	-0.8%	888.4	-0.1%
32	Osiris-Rex	1125.3	960.7	-14.6%	1127.9	0%	619.4	209.4	299.1	7.53	39.7	1044.3	-7.2%	1033.0	-8.2%
33	Phoenix	417	430.4	3.2%	419.2	1%	322.1	85.6	11.5	1.29	8.9	424.8	1.9%	425.5	2.1%
34	Psyche	1108.1	1197.9	8.1%	945.6	-15%	821.9	114.5	9.2	0.43	21.4	1071.7	-3.3%	1088.9	-1.7%
35	Stardust	236.4	242.6	2.6%	212.5	-10%	152.7	34.5	25.4	12.13	2.1	227.6	-3.7%	229.6	-2.9%
				TMCT		sum(costs)							Average	Weighted Average	
			mean	9.0%		9.6%							6.7%		6.7%
			st.dev	7.3%		11.8%							5.7%		5.4%
			perfor	16.3%		21.4%							12.5%		12.1%

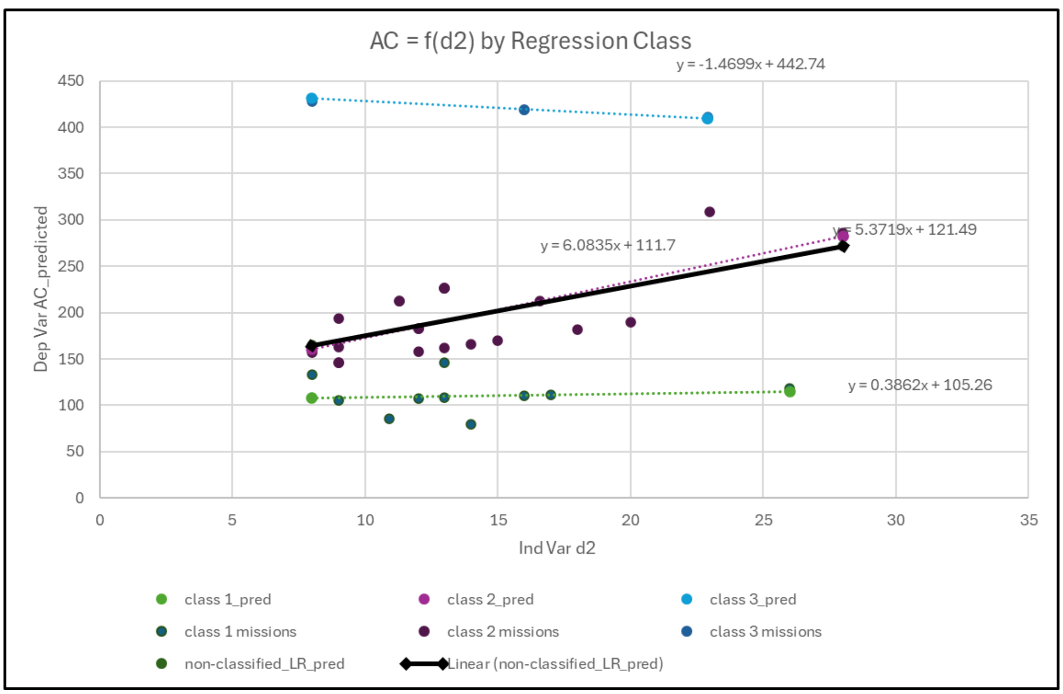
Appendix F

Hart model – model of actual cost response to inner diameter d2 of pipe predictor. Demonstrates CAM-to-cost modeling utility.

Ac = f(d2)

Zero class limit = 0.00

Model Summary (35 projects)				
Classified Linear Reg with Classified Residual Reg				
AC = f(d2)				
Class Regression Threshold			29%	
Class Residual Threshold			9%	
	Regress	Residual	Min	Max
Class 1	10	10	-9.4%	4.7%
Class 2	21	16	-7.2%	8.0%
Class 3	4	9	-1.5%	2.6%
Model	Step 1	Step 2	Step 3	
mean	18.9%	-2.1%	0.1%	
|mean|	37.1%	11.5%	3.4%	
st.dev	48.2%	15.0%	4.2%	
st.dev | |	35.7%	9.7%	2.4%	
co.var	39.1%	-13.8%	40.15	
co.var| |	103.8%	118.2%	1.24	
|mean| + st.dev | |	72.8%	21.2%	5.8%	



				AC = f(d2)			
Data ID	Mission	Ind Var d2	Dep Var AC	reg class	res class	Predict	Pred %e
24	Proj24	14	88	1	1	79.7	-9.4%
18	Proj18	26	118.5	1	2	118.1	-0.4%
35	Proj35	10.9	84	1	1	85.2	1.5%
6	Proj06	17	109	1	2	111.4	2.2%
25	Proj25	16	113.9	1	2	110.6	-2.9%
2	Proj02	13	106.2	1	2	108.4	2.1%
8	Proj08	9	101.6	1	2	105.4	3.8%
20	Proj20	12	112.7	1	2	107.7	-4.5%
28	Proj28	13	145.1	1	3	146.0	0.6%
15	Proj15	8	127.2	1	3	133.2	4.7%
7	Proj07	15	157.2	2	1	169.8	8.0%
9	Proj09	18	170.4	2	1	181.5	6.5%
4	Proj04	14	156.6	2	1	165.9	5.9%
16	Proj16	12	153.8	2	1	158.1	2.8%
11	Proj11	9	148.6	2	1	146.4	-1.5%
26	Proj26	9	151	2	1	146.4	-3.1%
29	Proj29	20	204	2	1	189.4	-7.2%
22	Proj22	13	170.8	2	1	162.0	-5.2%
34	Proj34	12	170.6	2	2	182.5	7.0%
13	Proj13	9	166	2	2	163.2	-1.7%
30	Proj30	12	182.4	2	2	182.5	0.0%
27	Proj27	16.6	207.8	2	2	212.1	2.1%
19	Proj19	8	162.5	2	2	156.7	-3.6%
12	Proj12	28	289.4	2	2	285.5	-1.3%
32	Proj32	11.3	199.3	2	3	212.5	6.6%
10	Proj10	11.3	210	2	3	212.5	1.2%
31	Proj31	13	221.3	2	3	226.5	2.4%
23	Proj23	9	200.1	2	3	193.5	-3.3%
33	Proj33	13	237.8	2	3	226.5	-4.7%
14	Proj14	23	306	2	3	309.0	1.0%
1	Proj01	13	239.5	2	3	226.5	-5.4%
21	Proj21	22.9	412	3	2	410.8	-0.3%
3	Proj03	16	408	3	2	418.4	2.6%
17	Proj17	16	425	3	2	418.4	-1.5%
5	Proj05	8	433.5	3	2	427.3	-1.4%

Hart model – model of actual cost response to the inner diameter d1 of the pipe predictor.

Demonstrates CAM to cost modeling utility.

AC = f(d1)

Zero class limit = 0.00

Model Summary (35 projects)

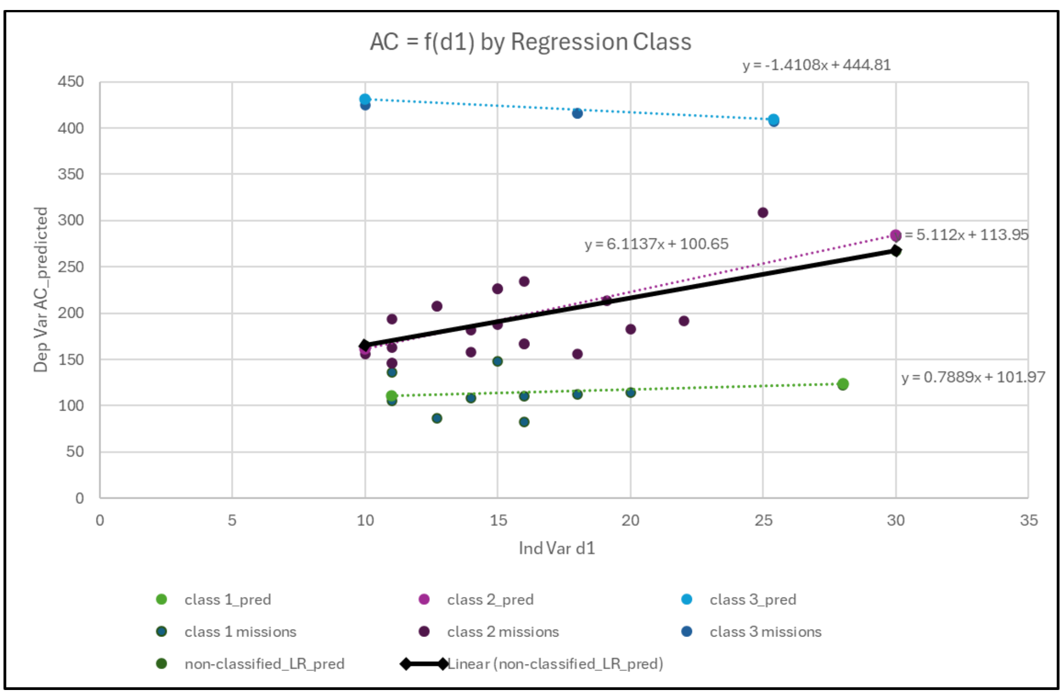
Classified Linear Reg with Classified Residual Reg

AC = f(d1)

Class Regression Threshold	27%
Class Residual Threshold	10%

	Regress	Residual	Min	Max
Class 1	11	9	-6.0%	6.9%
Class 2	20	16	-6.1%	7.6%
Class 3	4	10	-2.2%	1.8%

Model	Step 1	Step 2	Step 3
mean	19.0%	-2.3%	0.4%
|mean|	37.5%	12.3%	3.3%
st.dev	48.2%	15.5%	3.9%
st.dev | |	35.3%	9.5%	1.9%
co.var	39.3%	-14.5%	9.89
co.var| |	106.3%	130.0%	1.16
|mean| + st.dev | |	72.9%	21.8%	5.3%



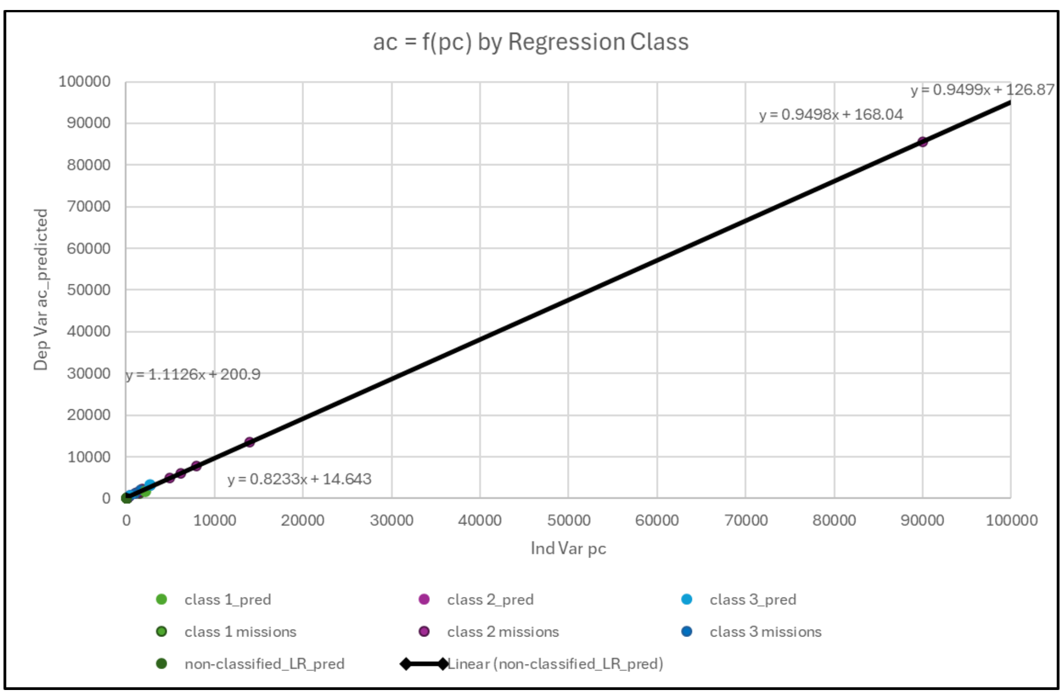
				AC = f(d1)			
Data ID	Mission	Ind Var d1	Dep Var AC	reg class	res class	Predict	Pred %e
24	Proj24	16	88	1	1	82.7	-6.0%
18	Proj18	28	118.5	1	2	122.1	3.1%
35	Proj35	12.7	84	1	1	86.6	3.0%
6	Proj06	20	109	1	2	114.2	4.8%
2	Proj02	16	106.2	1	2	110.2	3.8%
25	Proj25	18	113.9	1	2	112.2	-1.5%
8	Proj08	11	101.6	1	2	105.3	3.6%
20	Proj20	14	112.7	1	2	108.2	-4.0%
15	Proj15	11	127.2	1	3	136.0	6.9%
28	Proj28	15	145.1	1	3	147.7	1.8%
7	Proj07	18	157.2	1	3	156.4	-0.5%
9	Proj09	20	170.4	2	1	183.3	7.6%
4	Proj04	16	156.6	2	1	166.6	6.4%
16	Proj16	14	153.8	2	1	158.3	2.9%
22	Proj22	16	170.8	2	1	166.6	-2.5%
11	Proj11	11	148.6	2	1	145.8	-1.9%
26	Proj26	11	151	2	1	145.8	-3.5%
29	Proj29	22	204	2	1	191.6	-6.1%
34	Proj34	14	170.6	2	2	181.5	6.4%
19	Proj19	11	162.5	2	2	162.5	0.0%
30	Proj30	15	182.4	2	2	187.8	3.0%
27	Proj27	19.1	207.8	2	2	213.7	2.8%
13	Proj13	10	166	2	2	156.2	-5.9%
12	Proj12	30	289.4	2	2	282.5	-2.4%
32	Proj32	12.7	199.3	2	3	207.3	4.0%
31	Proj31	15	221.3	2	3	226.2	2.2%
10	Proj10	12.7	210	2	3	207.3	-1.3%
23	Proj23	11	200.1	2	3	193.2	-3.4%
1	Proj01	16	239.5	2	3	234.5	-2.1%
33	Proj33	15	237.8	2	3	226.2	-4.9%
14	Proj14	25	306	2	3	308.7	0.9%
21	Proj21	25.4	412	3	2	406.5	-1.3%
3	Proj03	18	408	3	2	415.5	1.8%
17	Proj17	18	425	3	2	415.5	-2.2%
5	Proj05	10	433.5	3	2	425.1	-1.9%

Servramckx – model of actual cost response to proposed cost predictor. Demonstrates margin management.

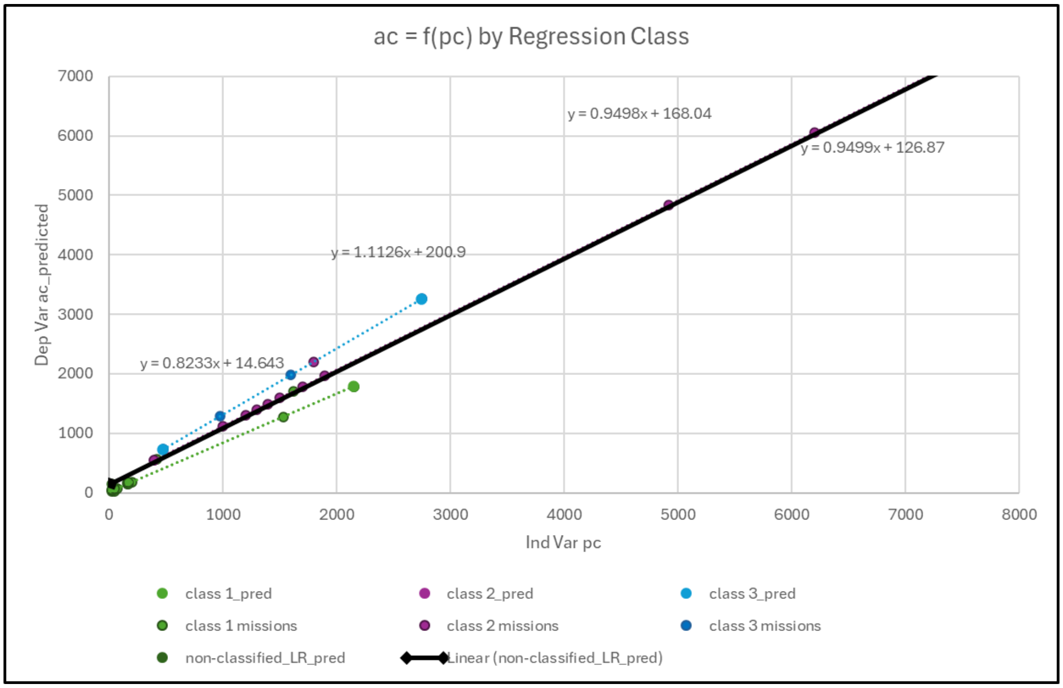
ac = f(pc)

Zero class limit = 0.00

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
ac = f(pc)				
	Class Regression Threshold		10%	
	Class Residual Threshold		13%	
	Regress	Residual	Min	Max
Class 1	14	2	-11.0%	8.5%
Class 2	16	29	-4.6%	8.7%
Class 3	5	4	-11.1%	7.7%
Model	Step 1	Step 2	Step 3	
mean	68.1%	-3.5%	0.7%	
|mean|	75.1%	9.4%	3.4%	
st.dev	136.7%	19.8%	4.7%	
st.dev | |	132.9%	17.7%	3.3%	
co.var	49.8%	-17.8%	6.25	
co.var| |	56.5%	53.0%	1.39	
|mean| + st.dev | |	208.0%	27.1%	6.6%	



Enlarging the lower scale of the Regression Class Plot shows the boundary constraints for each predictor.



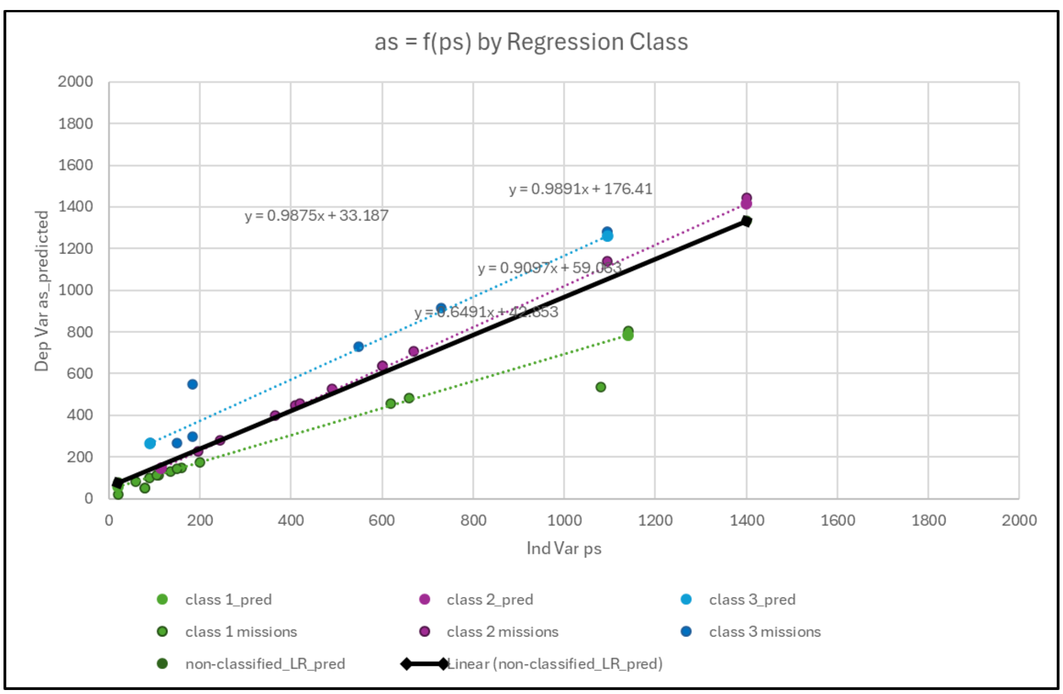
				ac = f(pc)			
Data ID	Mission	Ind Var pc	Dep Var ac	reg class	res class	Predict	Pred %e
2	Proj02	26.3	23	1	1	23.0	0.0%
4	Proj04	53.226	30.303	1	1	30.3	0.0%
9	Proj09	37.267	44.747	1	2	44.3	-1.0%
12	Proj12	30.41	49.221	1	3	48.0	-2.4%
1	Proj01	63.4	64.63	1	2	65.8	1.8%
8	Proj08	78	72	1	2	77.8	8.1%
3	Proj03	48.074	63.423	1	3	64.8	2.1%
10	Proj10	162.981	136.18	1	2	147.8	8.5%
14	Proj14	205.7	170	1	2	183.0	7.6%
5	Proj05	167.773	170.508	1	2	151.8	-11.0%
11	Proj11	165.674	174.549	1	3	176.0	0.8%
26	Proj26	168.15	180	1	3	178.4	-0.9%
29	Proj29	1533.725	1252.832	1	2	1276.3	1.9%
24	Proj24	2150	1800	1	2	1783.7	-0.9%
13	Proj13	1618.5	1605.532	2	2	1704.2	6.1%
6	Proj06	14000	13200	2	2	13463.6	2.0%
28	Proj28	425	525	2	2	570.7	8.7%
22	Proj22	8000	7700	2	2	7765.1	0.8%
17	Proj17	400000	380000	2	2	380070.3	0.0%
33	Proj33	1300	1367.6	2	2	1401.7	2.5%
23	Proj23	90000	86000	2	2	85645.2	-0.4%
25	Proj25	4919.05	4845.591	2	2	4838.9	-0.1%
7	Proj07	6200	6100	2	2	6055.5	-0.7%
16	Proj16	1900	1958.9	2	2	1971.6	0.6%
31	Proj31	395	515.6	2	2	542.2	5.2%
15	Proj15	1200	1304.4	2	2	1306.7	0.2%
34	Proj34	1500	1630.5	2	2	1591.7	-2.4%
18	Proj18	1400	1555.54	2	2	1496.7	-3.8%
20	Proj20	1700	1868.3	2	2	1781.6	-4.6%
19	Proj19	1000	1160	2	2	1116.8	-3.7%
21	Proj21	1800	2044.8	3	2	2202.6	7.7%
35	Proj35	1600	1926.4	3	2	1980.0	2.8%
30	Proj30	475	699	3	2	728.4	4.2%
27	Proj27	2750	3345	3	2	3259.5	-2.6%
32	Proj32	975	1445	3	2	1284.7	-11.1%

Servramckx – model of actual cost response to proposed cost predictor. Demonstrates margin management.

as = f(ps)

Zero class limit = 0.00

Model Summary (35 missions)				
Classified Linear Reg with Classified Residual Reg				
as = f(ps)				
Class Regression Threshold			11%	
Class Residual Threshold			14%	
	Regress	Residual	Min	Max
Class 1	15	6	-37.2%	12.7%
Class 2	12	28	-12.1%	14.4%
Class 3	8	1	-2.7%	14.4%
Model	Step 1	Step 2	Step 3	
mean	18.8%	-6.9%	-1.6%	
|mean|	33.6%	15.9%	7.9%	
st.dev	57.1%	33.6%	11.2%	
st.dev | |	49.5%	30.4%	8.0%	
co.var	32.9%	-20.6%	-6.84	
co.var| |	67.9%	52.3%	1.42	
|mean| + st.dev | |	83.2%	46.2%	15.9%	



				as = f(ps)			
Data ID	Mission	Ind Var ps	Dep Var as	reg class	res class	Predict	Pred %e
20	Proj20	20	20	1	1	21.2	5.9%
24	Proj24	1080	540	1	1	535.4	-0.8%
27	Proj27	80	73	1	1	50.3	-31.1%
21	Proj21	80	80	1	1	50.3	-37.2%
14	Proj14	110	102	1	2	115.0	12.7%
15	Proj15	60	76	1	2	81.4	7.1%
28	Proj28	135	129	1	2	131.8	2.2%
2	Proj02	90	105	1	2	101.5	-3.3%
23	Proj23	200	180	1	2	175.4	-2.5%
29	Proj29	160	157	1	2	148.6	-5.4%
30	Proj30	105	120	1	2	111.6	-7.0%
35	Proj35	150	155	1	2	141.9	-8.5%
25	Proj25	620	510	1	2	457.6	-10.3%
22	Proj22	660	540	1	2	484.4	-10.3%
17	Proj17	1140	900	1	2	806.9	-10.3%
18	Proj18	410	390	2	2	445.6	14.3%
19	Proj19	490	460	2	2	526.4	14.4%
26	Proj26	420	415	2	2	455.7	9.8%
32	Proj32	115	164	2	2	147.6	-10.0%
1	Proj01	600	620	2	2	637.6	2.8%
33	Proj33	600	625	2	2	637.6	2.0%
4	Proj04	1095	1095	2	2	1137.6	3.9%
34	Proj34	245	300	2	2	278.9	-7.0%
13	Proj13	670	720	2	2	708.3	-1.6%
16	Proj16	1400	1447	2	2	1445.7	-0.1%
10	Proj10	365	425	2	2	400.2	-5.8%
3	Proj03	195	260	2	2	228.4	-12.1%
12	Proj12	730	825	3	2	913.3	10.7%
31	Proj31	150	235	3	1	268.8	14.4%
6	Proj06	1095	1278	3	2	1282.6	0.4%
11	Proj11	1095	1278	3	2	1282.6	0.4%
8	Proj08	183	274	3	1	296.0	8.0%
5	Proj05	548	730	3	2	729.1	-0.1%
9	Proj09	91	274	3	2	266.7	-2.7%
7	Proj07	183	548	3	3	548.0	0.0%