

Analysis of Data from the U.S. Shipbuilding Industry and Application to Improve Performance Metrics

by

Heather L. Willis

B.S. Mathematics, United States Naval Academy, 2014

Submitted to the Department of Mechanical Engineering and the
Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degrees of

NAVAL ENGINEER

and

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING AND COMPUTER
SCIENCE

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2024

© 2024 Heather L. Willis. All rights reserved.

The author hereby grants to MIT a nonexclusive, worldwide, irrevocable, royalty-free license to exercise any and all rights under copyright, including to reproduce, preserve, distribute and publicly display copies of the thesis, or release the thesis under an open-access license.

Authored by: Heather L. Willis
Department of Mechanical Engineering
Department of Electrical Engineering and Computer Science
May 10, 2024

Certified by: Andrew Gillespy
Professor of the Practice, Department of Mechanical Engineering, Thesis Supervisor

Certified by: Luca Daniel
Professor of Electrical Engineering and Computer Science, Thesis Supervisor

Accepted by: Nicolas Hadjiconstantinou
Graduate Officer
Department of Mechanical Engineering

Accepted by: Leslie A. Kolodziejcki
Professor of Electrical Engineering and Computer Science
Chair, Department Committee on Graduate Students

Analysis of Data from the U.S. Shipbuilding Industry and Application to Improve Performance Metrics

by

Heather L. Willis

Submitted to the Department of Mechanical Engineering and the
Department of Electrical Engineering and Computer Science
on May 10, 2024 in partial fulfillment of the requirements for the degrees of

NAVAL ENGINEER

and

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING AND COMPUTER
SCIENCE

ABSTRACT

The U.S. Navy is seeking to increase the number of ships in the fleet due to growing threats, however shipyards are facing numerous issues leading to a delay in the delivery of naval warships along with cost overruns. At the same time, there is significant data available from the construction process, creating an opportunity for data analysis with the intention of identifying and hopefully resolving some of these issues. Addressing these concerns, this thesis scrutinizes Earned Value Management (EVM) data from actual shipbuilding projects, capitalizing on the datasets available to help identify the root causes of such delays. The study begins with data cleaning, an essential step that ensures the real-world data's integrity and relevance. Preliminary data analysis was then conducted to explore cost variance, schedule adherence, and the learning curve effect observed across different hulls, setting the stage for deeper investigative modeling. Following model exploration and selection, the core of the thesis is a predictive model that uses polynomial and linear regression to predict the progression of costs over time and comparison to the prediction metrics currently in use. A regression model was chosen over more complex models like a long short-term memory (LSTM) neural network due to its simplicity, interpretability, and ease of retraining with new data, ensuring that stakeholders can readily understand and apply the model's insights while maintaining its relevance over time. The target prediction metric for this model is the Actual Cost of Work Performed (ACWP), however similar models could also be leveraged to predict schedule. In creating this model, several features were analyzed including both the Budgeted Cost of Work Scheduled (BCWS) and the Budget at Completion (BAC), both known metrics at the start of construction. After testing various combinations of these features and comparing the mean squared error (MSE), the chosen model uses time and BCWS divided by BAC as input features, serving as a budgeted completion percentage. The model is tailored further to reflect industry-specific cost behaviors, enforcing non-negative, cumulative cost predictions. This model was trained, tested and validated using EVM data from one key event (KE), a specific subset of the overall ship construction process with the intent that it could be applied to all key events and aggregated to provide cost predictions for an

entire hull. This thesis will ideally serve as a framework for shipyards to improve project cost predictions and identify indicators of large cost overruns early enough to correct them within the ship construction timeline.

Thesis supervisor: Andrew Gillespy

Title: Professor of the Practice, Department of Mechanical Engineering

Thesis supervisor: Luca Daniel

Title: Professor of Electrical Engineering and Computer Science

Acknowledgments

I would like to thank the shipbuilders and SUPSHIP employees for providing me with the data and resources necessary to complete this project. Without their continued support this thesis would not have existed.

To my advisors, CAPT Gillespy and Professor Daniel, thank you for your technical guidance and feedback. I am grateful for the time you invested in me and this project.

To my good friend Saad, who provided me with countless hours of guidance, programming expertise, and friendship. Your willingness to assist me in all aspects of this thesis truly made it all possible.

To the 2N program and the U.S. Navy, thank you for the opportunity of a lifetime. The growth I have experienced in the past three years, professionally and personally, was beyond my expectations. It is an honor to be a graduate of this esteemed program and to be affiliated with the world-class institution that is MIT.

To my 2N classmates, friends and family, thank you for always having my back, inspiring me to grow into an improved version of myself, and always being there as an outlet for fun during these sometimes stressful years. Words can't describe the gratitude I have for all of the support, love and guidance I have received from you all throughout this program.

Contents

Title page	1
Abstract	2
List of Figures	7
List of Tables	9
1 Introduction	10
1.1 Industry Overview	10
1.2 Earned Value Management Overview	13
1.3 Project Motivation	15
1.4 Problem Statement and Approach	16
1.5 Thesis Overview	16
2 Data Analysis	17
2.1 Data Introduction	17
2.1.1 Data Source & Collection	17
2.1.2 Data Attributes	17
2.2 Data Cleaning	18
2.3 Exploratory Data Analysis	20
2.3.1 Sources of Error	20
2.3.2 Cost Variance Analysis	20
2.3.3 Schedule Analysis	26
2.3.4 Learning Curve	27
3 Machine Learning and Model Selection	29
3.1 Introduction to Machine Learning	29
3.1.1 Types of Machine Learning Models and Applicability	29
3.2 Model Selection Criteria, Considerations and Limitations	30
3.2.1 Key Criteria for Model Selection	30
3.2.2 Challenges in Applying Machine Learning to EVM Data	31
3.2.3 Models Not Suitable for EVM Data	31
3.3 Models Considered	32
3.3.1 Prophet	32
3.3.2 Autoformer/Transformer	32

3.3.3	Long Short-Term Memory (LSTM) Network	33
3.3.4	Autoregressive Integrated Moving Average (ARIMA)	34
3.3.5	Regression	35
4	Model Development	36
4.1	Model Selection	36
4.1.1	Regression Justification	36
4.1.2	Regression Models Considered	37
4.2	Model Architecture	38
4.2.1	Model Objective	38
4.2.2	Data Granularity	38
4.2.3	Regression Model Type, Equation and Interpretation	38
4.2.4	Model Assumptions	40
4.3	Data Preprocessing	42
4.3.1	One-hot Encoding & Baseline Hull	44
4.4	Feature Engineering	45
4.4.1	Feature Preprocessing Techniques	45
4.4.2	Polynomial Features	45
4.4.3	Feature Engineering Results	48
4.5	Model Training	49
4.5.1	Milestone Prediction	49
4.6	Model Evaluation	50
4.7	Sensitivity Analysis	54
4.7.1	Ordinal Date	55
4.7.2	BCWS/BAC	56
4.7.3	Hull Identifier Variables	56
4.8	Results Analysis	57
5	Conclusion	60
5.1	Summary of Findings	60
5.2	Lessons Learned	61
5.3	Limitations	62
5.4	Future Work	62
5.5	Final Thoughts	63
A	Code Listing	65
B	Additional Graphs	68
B.1	Graphs of ACWP Prediction Comparisons	68
	References	72

List of Figures

1.1	Visual Layout of Fleet Makeup as Defined by Force Design 2045 [3]	11
1.2	The Construction of a Liberty Ship at the Bethlehem-Fairfield Shipyards, Baltimore, Maryland, in March/April 1943 [7]	12
1.3	Final Superlift Marks Structural Completion of CVN 79 [8]	12
1.4	Sample Graph of an EVM Project [9]	14
1.5	Block Diagram of an EVM Project Structure [9]	14
1.6	Changes in Sustainment Metrics per Ship across Selected Navy Ship Classes, Fiscal Years 2011 through 2021 [10]	15
2.1	Ship Construction Organization Chart	19
2.2	Example of a Re-baseline Event	20
2.3	50 LT/KE Combinations with the Largest Negative Cost Variances for Hull XJ	21
2.4	50 LT/KE Combinations with the Largest Negative Cost Variances for Hull XK	21
2.5	50 LT/KEs with the Largest Negative Cost Variances Across All Hulls	22
2.6	50 KEs with the Largest Negative Cost Variances Across All Hulls	22
2.7	50 LTs with the Largest Negative Cost Variances Across All Hulls	23
2.8	Cost Variance Across Hulls for LT:915, KE:25569	23
2.9	Cost Variance Across Hulls for KE:13514	24
2.10	Cost Variance Across Hulls for LT:915	24
2.11	Cost Variance Across Complete Hulls for LT:915, KE:25569	25
2.12	Cost Variance Across Complete Hulls for KE:30114	25
2.13	Days to Complete Work for LT: X11A, KE: 9A198 Across Hulls	26
2.14	Days to Complete Work for LT: X15, KE: 9A235 Across Hulls	26
2.15	ACWP Learning Curve for LT:915, KE:25569	28
2.16	ACWP Learning Curve for KE:30114	28
4.1	ACWP vs. Date for Each Hull	39
4.2	ACWP vs. Ordinal Date for Each Hull	43
4.3	ACWP vs. BCWS/BAC for Each Hull	44
4.4	Actual & Predicted ACWP for Test Hull XR with Second Order Polynomial	46
4.5	Actual & Predicted ACWP for Test Hull XR with Third Order Polynomial	46
4.6	Actual & Predicted ACWP for Test Hull XR with Second Order Polynomial & Postprocessing	47

4.7	Actual & Predicted ACWP for Test Hull XR with Third Order Polynomial & Postprocessing	47
4.8	Actual Cost (ACWP) vs. Budgeted Cost (BCWS) for Each Hull	48
4.9	Actual & Predicted ACWP for Test Hull XR with BCWS	49
4.10	Actual & Predicted ACWP for Test Hull XS at 0% Milestone	52
4.11	Actual & Predicted ACWP for Test Hull XS at 15% Milestone	52
4.12	Actual & Predicted ACWP for Test Hull XS at 25% Milestone	53
4.13	Actual & Predicted ACWP for Test Hull XS at 50% Milestone	53
4.14	Actual & Predicted ACWP for Test Hull XS at 75% Milestone	54
4.15	Sensitivity Analysis: Predicted ACWP vs. Ordinal Date	55
4.16	Sensitivity Analysis: Predicted ACWP vs. BCWS/BAC	56
4.17	ACWP Prediction Comparison for Test Hull XP	57
4.18	Average ACWP Prediction Difference on a Logarithmic Scale	58
B.1	ACWP Prediction Comparison for Test Hull X2	68
B.2	ACWP Prediction Comparison for Test Hull XL	69
B.3	ACWP Prediction Comparison for Test Hull XM	69
B.4	ACWP Prediction Comparison for Test Hull XN	70
B.5	ACWP Prediction Comparison for Test Hull XR	70
B.6	ACWP Prediction Comparison for Test Hull XS	71
B.7	ACWP Prediction Comparison for Test Hull XT	71

List of Tables

4.1	MSE [$\times 10^6$] and R-squared for Each Milestone and Test Hull	51
-----	--	----

Chapter 1

Introduction

The United States shipbuilding industry is struggling more than ever, unable to compete with the likes of China, South Korea, and Japan. China alone now accounts for nearly 40% of the global commercial shipbuilding output, compared to 0.002% in the United States [1]. With limited commercial shipbuilding in the country, the ability to produce large ships and complex submarines for the Navy and the military as a whole has suffered. The cancellation of large programs such as the Seawolf-class submarine and more recently the Zumwalt-class destroyer has contributed to an unsteady demand, making it difficult for shipbuilders to invest in infrastructure upgrades and maintain a proficient work force. The COVID-19 pandemic exacerbated these issues by hindering an already struggling material vendor base and forcing a mass exodus of retirement-eligible workers, leaving a deficit of experienced shipbuilders.

Amid these challenges, the Chief of Naval Operations (CNO) Navigation Plan 2022 lays out the desired fleet layout by 2045, as shown in Figure 1.1, with the desire to grow the fleet by over 50 manned ships [2]. As the budget for new construction warships increases, the shipbuilders are unable to meet this demand. For example, the Navy is funding the construction of three Arleigh Burke Flight III destroyers and two Virginia-class attack submarines per year, but the shipyards are delivering only 1.5 and 1.2 of each platform, respectively [3], [4]. At the same time the shipbuilders have been collecting large amounts of earned value management (EVM) data as a performance and progress metric. The analysis of this data may provide an opportunity to identify areas of improvement in the construction process.

1.1 Industry Overview

Ship construction techniques have matured from traditional "stick building" (shown in Figure 1.2) to the currently favored modular construction (shown in Figure 1.3). Stick building construction begins with the laying of the keel after which the ship is built from the bottom up. Modular construction is the process in which the ship is built and outfitted in large sections, or modules, that are ultimately assembled together. This construction technique results in increased efficiency due to the widely accepted "1-3-8 rule" among shipbuilders. This rule states that 1 hour of work in a workshop is equivalent to 3 hours of work in an assembly area or 8 hours of work after fabrication (or launch) based on compounding

Navy's Force Design 2045

CNO Gilday's latest Navigation Plan outlines the service's plan for its future fleet design.



■ Aircraft Carriers ■ Ballistic Missile Submarines ■ Guided Missile Submarine ■ Attack Submarines
■ Large Surface Combatants ■ Small Surface Combatants ■ Mine Countermeasures ■ Large Amphibious Ships
■ Light Amphibious Warships ■ Auxiliary/Combat Logistics Ships ■ Unmanned

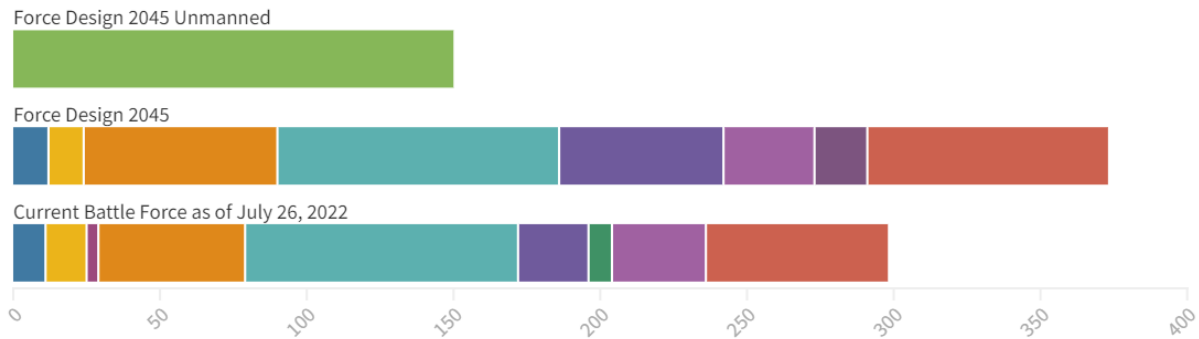


Figure 1.1: Visual Layout of Fleet Makeup as Defined by Force Design 2045 [3]

limitations of access, services, and ergonomics as the construction progresses [5].

As described previously, the U.S. commercial shipbuilding industry has declined to its lowest point for various reasons, to include a lack of government subsidies and the rise in the East Asian shipbuilding industry (China, South Korea, and Japan specifically). Without the infrastructure, workforce, and knowledge of a commercial shipbuilding force, military shipbuilding has similarly struggled. Adm. Mike Gilday, the senior U.S. Navy officer at the time, stated in 2022 that the industrial base capacity is the “biggest barrier” to increasing the number of ships in the Navy [3]. As the U.S. struggles, China has made significant growth in both commercial and military shipbuilding in recent years, with a navy now surpassing the U.S. Navy in number of ships (about 350 compared to less than 300) [1].

There are several active shipbuilders in the U.S. that build naval ships to include Huntington Ingalls Industries (HII), General Dynamics, Fincantieri Marine Group, and Austal USA [6]. The common challenge that these shipbuilders face is difficulty with delivering ships on schedule and within budget. As mentioned previously the delivery of Arleigh Burke-class destroyers and Virginia-class attack submarines are significantly below the contracted rate. This will not improve as the contracts for the Columbia-class missile submarines, more surface combatants, and unmanned vessels are added to the already struggling workforce. These struggles stem from issues with the vendor and supply base, retirement of many experienced workers, and difficulty in hiring new workers to fill these gaps.

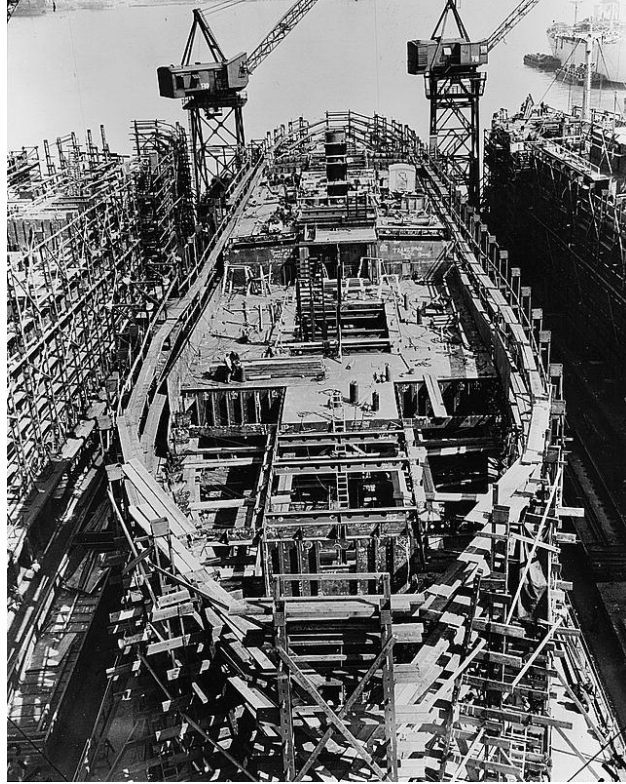


Figure 1.2: The Construction of a Liberty Ship at the Bethlehem-Fairfield Shipyards, Baltimore, Maryland, in March/April 1943 [7]

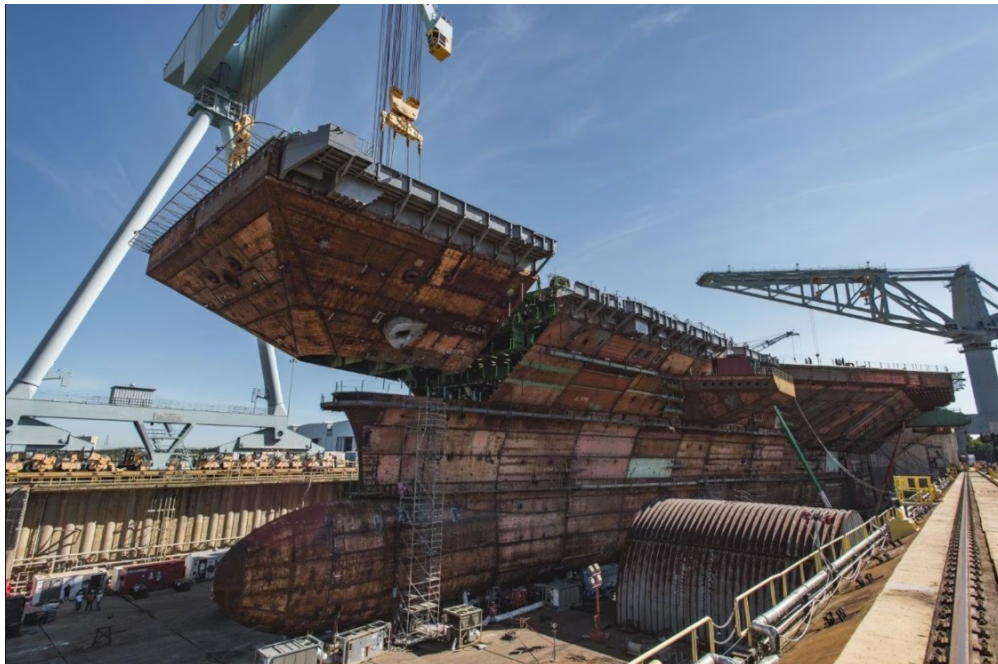


Figure 1.3: Final Superlift Marks Structural Completion of CVN 79 [8]

1.2 Earned Value Management Overview

Earned Value Management (EVM) is a powerful project management tool that integrates cost, schedule, and performance metrics to provide a comprehensive overview of a project's health and progress. It is widely used in industries such as construction, defense, and information technology to measure project performance, identify potential issues, and facilitate informed decision-making. There are minor differences in the terms used in EVM, but this project uses the definitions as laid out by the Defense Acquisition University (DAU) [9]. Figures 1.4 and 1.5 show examples of an EVM project on a graph of cost vs. time and the organizational breakdown, respectively [9].

Some key terms used in EVM include:

- Budgeted Cost of Work Scheduled (BCWS): BCWS represents the time-phased budget plan for work scheduled. It is the equivalent of the planned value.
- Budgeted Cost of Work Performed (BCWP): BCWP is the value of completed work in terms of the work's assigned budget. It is the equivalent of work done or earned value.
- Actual Cost of Work Performed (ACWP): ACWP is the cost actually incurred in accomplishing work performed. It is the equivalent of actual cost.
- Performance Measurement Baseline (PMB): PMB is the contract time-phased budget plan of a project as shown in Figure 1.5.
- Total Allocated Budget (TAB): TAB is the sum of all budgets for work on a contract as shown in Figure 1.5.
- Budget at Completion (BAC): BAC is the sum of all budgets for the contract through any given Work Breakdown Structure (WBS) level.
- Estimate at Completion (EAC): EAC is the estimate of total cost for the contract through any given level generated by the contractor.
- Management Reserve (MR): MR is the amount of the total budget withheld for management control purposes for future considerations to handle execution risks. It is not part of the PMB as shown in Figure 1.5.
- Cost Performance Index (CPI): CPI is a measure of cost efficiency, calculated by dividing BCWP by ACWP. A CPI greater than 1 indicates cost efficiency, while a value less than 1 suggests cost overruns.
- Schedule Performance Index (SPI): SPI measures schedule efficiency by dividing BCWP by BCWS. An SPI greater than 1 indicates that the project is ahead of schedule, while a value less than 1 suggests delays.
- Cost Variance (CV): CV is defined as BCWP minus ACWP and represents the difference in budgeted and actual cost of the work performed. It is illustrated on the graph in Figure 1.4.

- Schedule Variance (SV): SV is defined as BCWP minus BCWS and represents the difference in budgeted and actual schedule. It is illustrated on the graph in Figure 1.4.

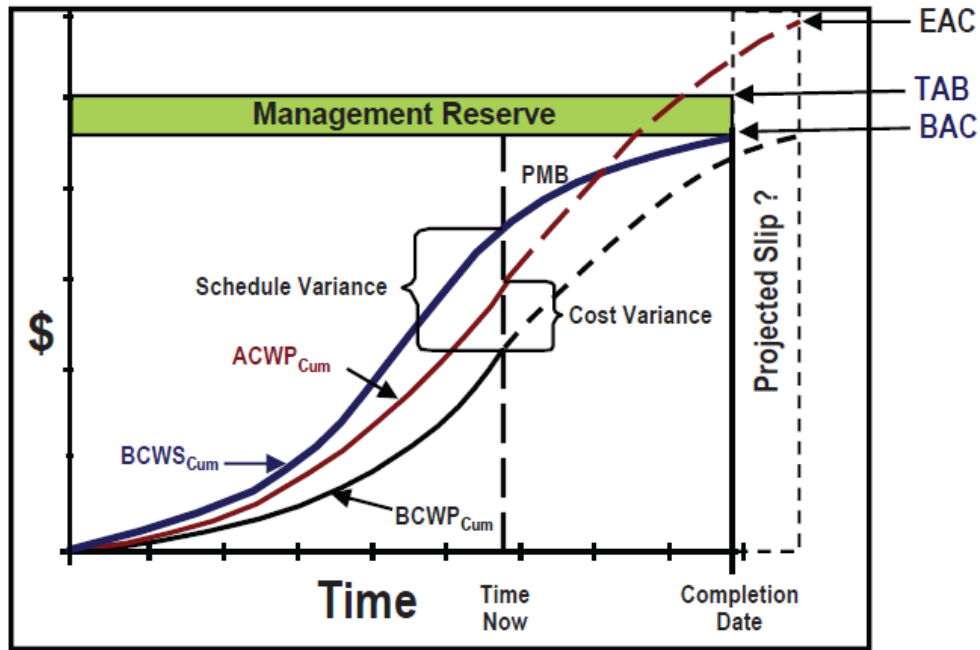


Figure 1.4: Sample Graph of an EVM Project [9]

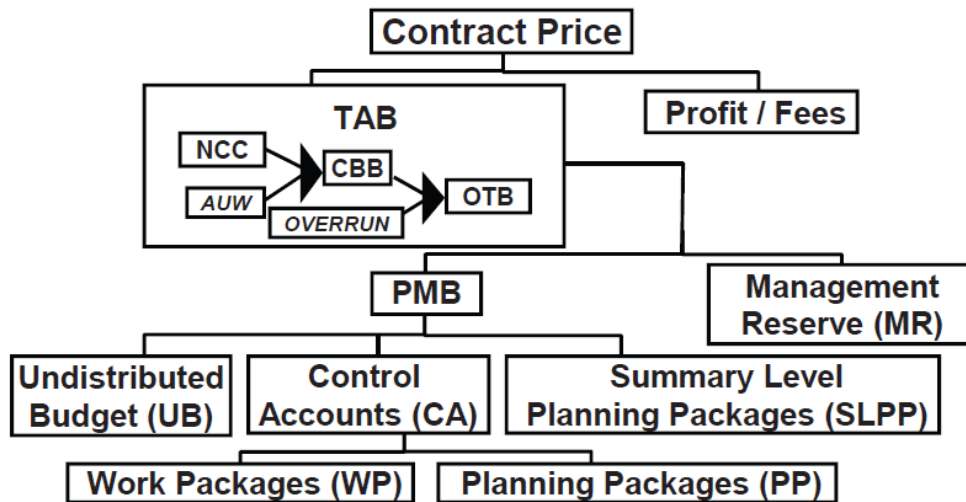


Figure 1.5: Block Diagram of an EVM Project Structure [9]

In the context of this thesis, the unit of cost is measured in the form of time (i.e. man-hours) as opposed to actual money. This eliminates inconsistencies from the changing value of money over time or changes in the contract structure. It also removes the rates (\$/hr) which can fluctuate with the cost of health care, inflation, overhead, etc. EVM enables

project managers to assess project performance against the baseline plan. By comparing BCWP, BCWS, and ACWP, project managers can determine if the project is on budget and on schedule. CPI and SPI provide insights into cost and schedule performance trends, allowing proactive measures to be taken to address potential issues before they escalate. EVM also facilitates accurate forecasting, helping project teams make data-driven decisions to optimize project outcomes. Overall, EVM serves as a valuable tool for project control, performance measurement, and continuous improvement.

1.3 Project Motivation

With the obstacles facing the U.S. shipbuilding industry as a whole, the military industry is unable to produce ships and submarines at the contracted rate. Beyond the challenges with new ship construction, the U.S. Navy is similarly struggling with the maintenance of existing ships. A Government Accountability Office (GAO) report published in January of 2023 reviewed 10 classes of surface ships from 2011-2021 and reported three key findings (summarized in Figure 1.6) [10]:

- An increase in the number of maintenance cannibalizations, which is when a maintenance part is taken from another ship instead of replaced through the supply system.
- An increase in casualty reports, which are messages sent when the material condition of the ship inhibits the ability to conduct a primary mission.
- A decrease in steaming hours, or the hours when a ship is in an operating or training status.

Ship class	Total inventory	Maintenance cannibalizations*	Category 3 and 4 casualty reports	Days of maintenance delay
<i>Ticonderoga</i> -class cruiser (CG-47)	22	+3 ▲	-1 ▼	+7 ▲
<i>Nimitz</i> -class aircraft carrier (CVN-68)	10	+4 ▲	+2 ▲	+7 ▲
<i>Arleigh Burke</i> -class destroyer (DDG-51)	68	+7 ▲	+19 ▲	+20 ▲
<i>Freedom</i> -class littoral combat ship (LCS-1)	10	+15 ▲	+26 ▲	0 ●
<i>Independence</i> -class littoral combat ship (LCS-2)	12	+3 ▲	+26 ▲	+19 ▲
<i>America</i> -class amphibious assault ship (LHA-6) ⁹	2	-1 ▼	+13 ▲	0 ●
<i>Wasp</i> -class amphibious assault ship (LHD-1)	8	+9 ▲	+43 ▲	+10 ▲
<i>San Antonio</i> -class amphibious transport dock (LPD-17)	11	+3 ▲	+10 ▲	+33 ▲
<i>Whidbey Island</i> -class dock landing ship (LSD-41)	8	+6 ▲	+24 ▲	+19 ▲
<i>Harpers Ferry</i> -class dock landing ship (LSD-49)	4	+7 ▲	-11 ▼	-16 ▼
Fleetwide		+6 ▲	+15 ▲	+14 ▲

● No change (neutral) ▲ Increase (negative) ▼ Decrease (positive)

Source: GAO analysis of U.S. Navy data. | GAO-23-106440

Figure 1.6: Changes in Sustainment Metrics per Ship across Selected Navy Ship Classes, Fiscal Years 2011 through 2021 [10]

Alone these challenges with ship construction and maintenance are concerning, but with the growing threat of our adversaries’ navies they have become dire. The fleet size currently sits at 291 ships as of November 2023, but the goal of 355 ships became official policy in

the FY2018 National Defense Authorization Act [11]. In order to achieve this goal not only do the shipbuilders need to start delivering the contracted number of ships on time, but the maintenance community must be able to keep ships operational through their expected service lives.

Major ship maintenance periods, or overhauls, also use EVM to track and manage projects, meaning there is substantial data available. However, in both ship maintenance and new construction projects, EVM analysis is currently limited to the use of spreadsheets with significant manual analysis to identify issues. Access to more sophisticated software tools could significantly improve the data analysis process and have a lasting impact on the industry. While this project deals with only one shipbuilder and one type of ship, any findings that may improve efficiency in the construction process can potentially help in chipping away at this daunting task.

1.4 Problem Statement and Approach

This thesis aims to draw meaningful insights into the ship construction process based on EVM data for 20 hulls with the goal of improving management’s ability to use EVM information to make decisions. This project is split into two major pieces: an exploratory data analysis and the implementation of an ML model as a predictive tool. Chapter 2 details the data cleaning process and explains the results of the exploratory data analysis. The intention was to ask and subsequently answer questions the shipbuilder would find valuable and actionable. Chapters 3 & 4 cover the second half of the project to include the ML model selection process, prediction goals, and results.

1.5 Thesis Overview

The remainder of this thesis consists of four chapters:

Chapter 2 provides a discussion on the data used in this thesis. It begins with sharing the steps associated with data cleaning followed by the results from an exploratory data analysis using Python.

Chapter 3 provides the necessary background on various machine learning models. It begins with a presentation of relevant criteria for this type of prediction problem, then transitions to a discussion of the models considered and ultimately the model selected.

Chapter 4 shares the model development process used in this thesis. It begins by describing the model architecture before transitioning into the feature engineering process. It then describes the final model training, evaluation and results.

Chapter 5 provides a conclusion to this thesis. It begins with a summary, shares important lessons learned throughout the work, and concludes with a set of ideas for future work on related topics.

Chapter 2

Data Analysis

2.1 Data Introduction

2.1.1 Data Source & Collection

Located at each of the Navy's major shipbuilders is a detachment of government employees (civilians and military) called the Supervisor of Shipbuilding (SUPSHIP). The role of the SUPSHIP is "to independently administer and manage the execution of Department of Defense (DoD) contracts awarded to assigned commercial entities at the contractors' facilities in the shipbuilding and ship repair industry [12]." The data used in this thesis is from a specific U.S. naval ship construction program and was provided by the SUPSHIP with permission from the shipbuilding company. It contains all the EVM data from the construction of a ship class over multiple hulls.

The data that was collected and analyzed in this project focuses on the EVM metrics of BCWS, BCWP & ACWP. The BCWS is a fixed baseline budget schedule that the shipbuilder sets. The BCWP, or progress, is collected by the shipbuilder at a low level, and can be summed to provide information on the desired level of work. The ACWP, or actuals, is collected based off of what jobs were actually charged. The actuals come in through a system in which an employee clocks in and charges a shop order for the work they are currently performing. The shipbuilders collect the data weekly with progress and actuals. All of this sums up to the official reports that are distributed to the SUPSHIP. This data is received generally every two weeks from the shipbuilder, however the data used in this thesis is end of month data.

2.1.2 Data Attributes

The data consists of over four million rows, with each row representing a discrete entry of EVM data. The dates of the data span from 2014 until 2023 when it was provided for analysis in this thesis, and includes information on twenty different hulls. A list and explanation of relevant attributes (columns) in the data is provided below:

- Hull: A unique two digit code identifying the specific hull

- Corp: The shipbuilding corporation. Many ships now have different modules constructed by different builders.
- LT: Lead trade (see Figure 2.1)
- MM: Major milestone (see Figure 2.1)
- KE: Key event (see Figure 2.1)
- BAC: Budget at completion (see Section 1.2)
- BCWS: Budgeted cost of work scheduled (see Section 1.2)
- BCWP: Budgeted cost of work performed (see Section 1.2)
- ACWP: Actual cost of work performed (see Section 1.2)
- Code: The code is the concatenation of Hull, Corp, MM, KE & LT (in that order). This identifies each row of data to the lowest level.
- FileName: This contains the date of the entry and whether the data is patched, among other things.
- Personnel columns of names at various levels to include Vice President (VP), Director, and Cost Account Manager (CAM).

There are other columns in the original data that were not applicable in this thesis. Figure 2.1 shows a visual representation of the breakdown of the ship construction process. The modular construction process is broken down into major milestones and key events, both of which are the same across each hull of a given ship class. An example of a major milestone is the construction of a ship module, and that is broken down even further into key events such as outfitting, inspections, and fabrication. A lead trade is a specific workshop such as pipefitters, welders, electricians, etc. As shown in Figure 2.1, each key event consists of jobs requiring multiple lead trades. On top of that, there is always more than one hull under construction at any given time.

As illustrated in Figure 2.1, the lowest level of data available is for a lead trade within a specific key event (and associated major milestone) on a given hull (referred to as LT/KE combination moving forward). That said, the data can be aggregated in numerous ways to make the results meaningful and/or actionable.

2.2 Data Cleaning

Python is the programming language used throughout this project along with pandas, an open source data analysis tool built to use with Python, and Matplotlib for graphing [13], [14]. The primary data structure in pandas is a ‘DataFrame’, a table with labeled columns and rows that makes it easy to visualize and manipulate the data [15]. After familiarization, some pre-processing of the data was conducted to include the steps described below:

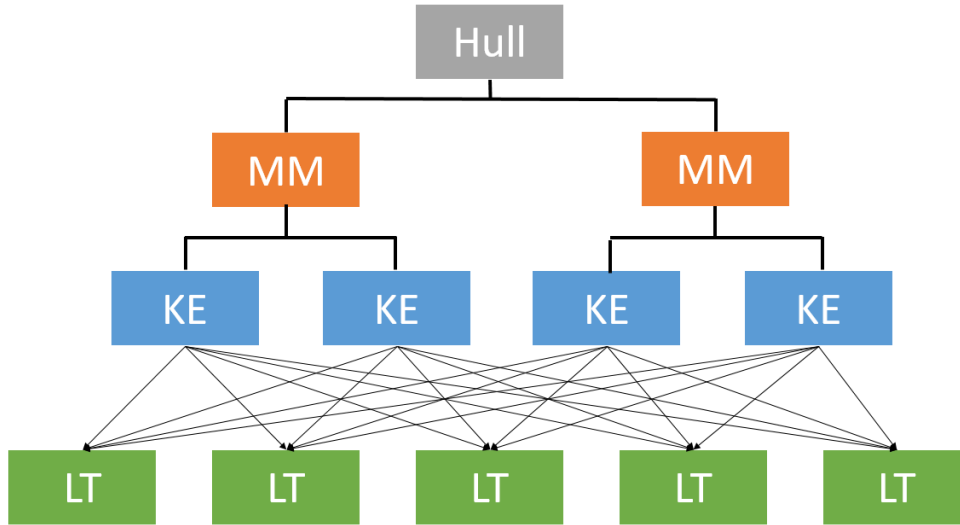


Figure 2.1: Ship Construction Organization Chart

- Extract the date from the 'FileName' and convert to date/time format
- Remove rows with missing data in necessary EVM columns (BAC, BCWS, BCWP & ACWP)
- Verify the data types of each column
- Clean columns with any extra spaces
- Handle "patched" data (explained below)

Patched data are manual adjustments that the shipbuilder makes to the data when there is an error or the data needs corrections. After exploring the patched data present in the dataset it was determined that there were two different types of patched entries made:

- A new row (with non-zero BAC, BCWS, BCWP, and/or ACWP) with no other corresponding unpatched data entry for the same code & date.
- An additive row (with non-zero BAC, BCWS, BCWP, and/or ACWP) that is summed with the unpatched data entry for the same code & date.

To account for the patched data, the entire dataset DataFrame was grouped by Code & Date, with the numerical EVM categories summed together. This preserved any patched rows that had no other entries for the same code & date while summing all rows that did. The resulting DataFrame was used for all further analyses.

2.3 Exploratory Data Analysis

2.3.1 Sources of Error

This dataset, as with all real-world data, contains errors because the collection process relies on human inputs. This was determined early on in the analysis process and while data cleaning helped to some extent, there was no way to fully remove all irregularities or sources of error. For this reason, the correct level to analyze the data had to be determined both by minimizing the impact of these errors and by the desired task.

Another source of inconsistency in the data comes from the practice of changing the schedule known as "re-baselines." Re-baselines can occur for a variety of reasons and are internal to the shipbuilder's Baseline Change Request (BCR) system in which they re-baseline hours, requiring reviews and signatures from the stakeholders. Following the approval of a BCR the hours become re-baselined. An example of a re-baseline from the data is shown in Figure 2.2. While this graph is for one specific key event on one hull, the inconsistencies and differences from the graph in Figure 1.4 are apparent.

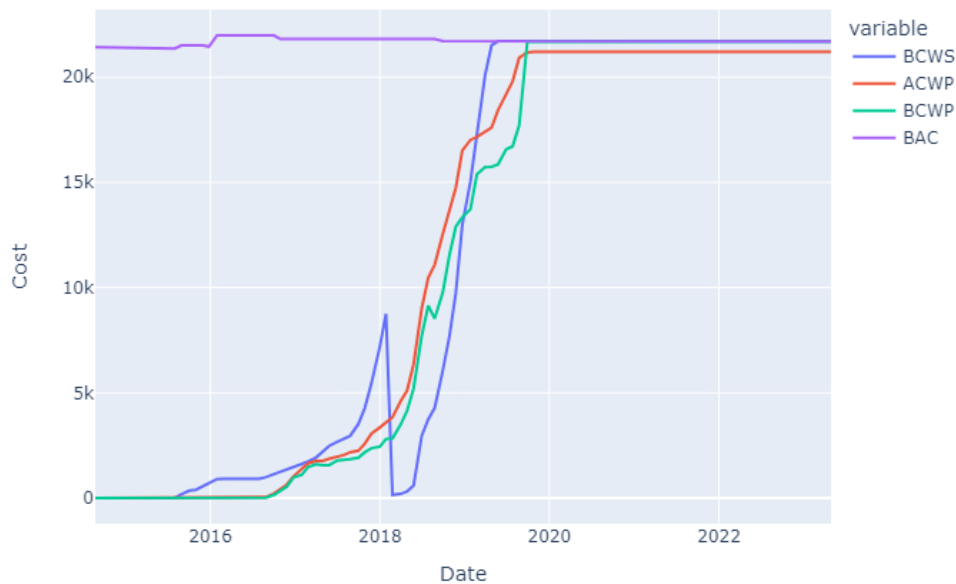


Figure 2.2: Example of a Re-baseline Event

2.3.2 Cost Variance Analysis

As defined in Section 1.2, cost variance is BCWP minus ACWP. This represents the difference between the predicted and actual cost of the work that has been performed. A negative cost variance means that the actual cost of the work was higher than predicted, and as such has negative implications on the budget. For this reason this analysis focuses on negative cost

variance. Additionally, the decision was made to only consider direct labor and therefore service and support trades were excluded from these analyses. This decision was made due to the fact that the service and support trades dwarfed all other trades with respect to cost and time, which is both expected and not especially insightful.

The first question to be answered is which LT/KE combinations are the most over-budget (largest negative CV) at the end of the work (or the most recent date). This was done with the use of a Pareto chart depicting the cost variance on the left y-axis and the cumulative percentage of negative cost variance on the right y-axis. The plots for the 50 largest negative cost variances for hulls XJ and XK are shown in Figures 2.3 & 2.4, respectively.

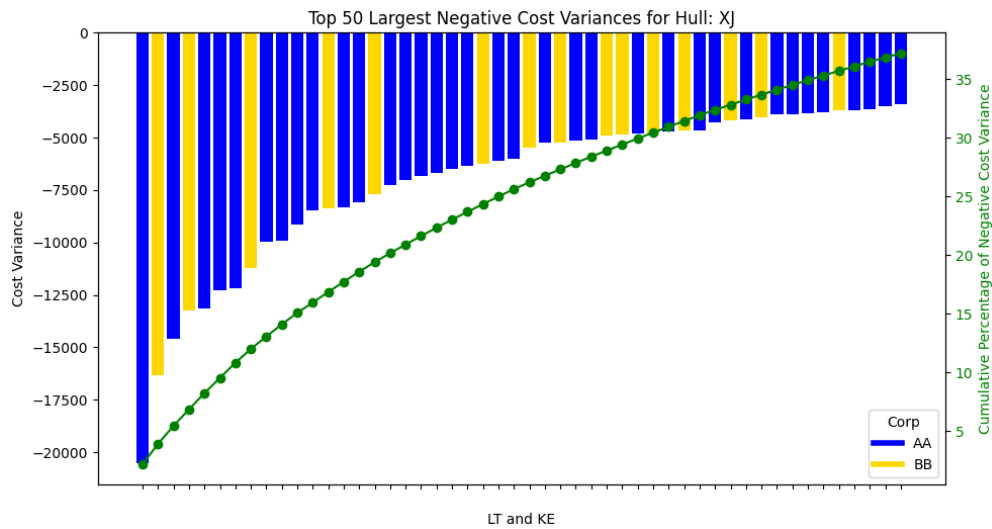


Figure 2.3: 50 LT/KE Combinations with the Largest Negative Cost Variances for Hull XJ

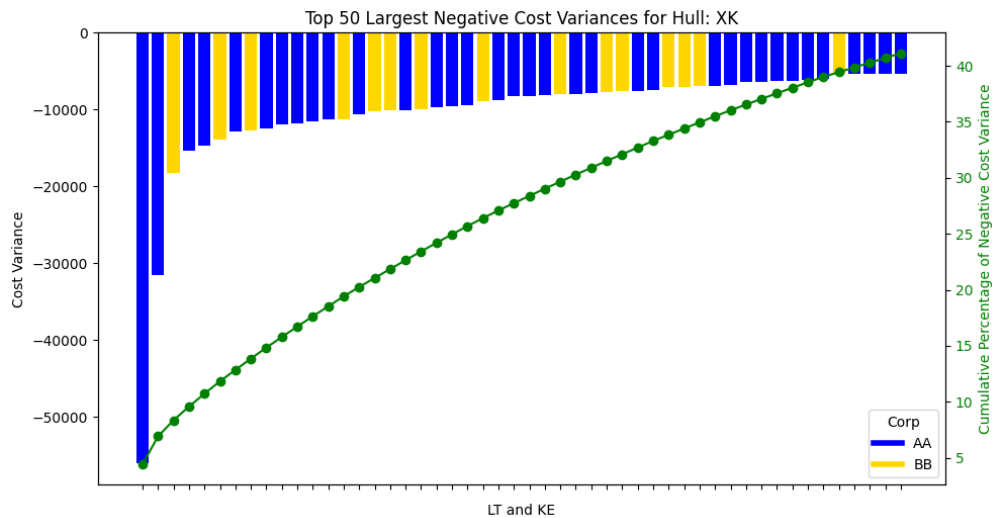


Figure 2.4: 50 LT/KE Combinations with the Largest Negative Cost Variances for Hull XK

These graphs show which LT/KE combinations are the most over-budget for a given hull, but they do not necessarily capture any common issues across hulls. Even in two consecutive hulls as shown in Figures 2.3 & 2.4, the top contributing LT/KE combinations were not the same. The logical next question to answer is which LT/KE combinations have the largest negative CV across all hulls (of those with provided data). This was also done at the KE level and the LT level separately and the results are shown in Figures 2.5 - 2.7. These graphs will help the shipbuilders identify areas in the construction process in which the budget is either predicting far too low or the work is taking far more resources to complete.

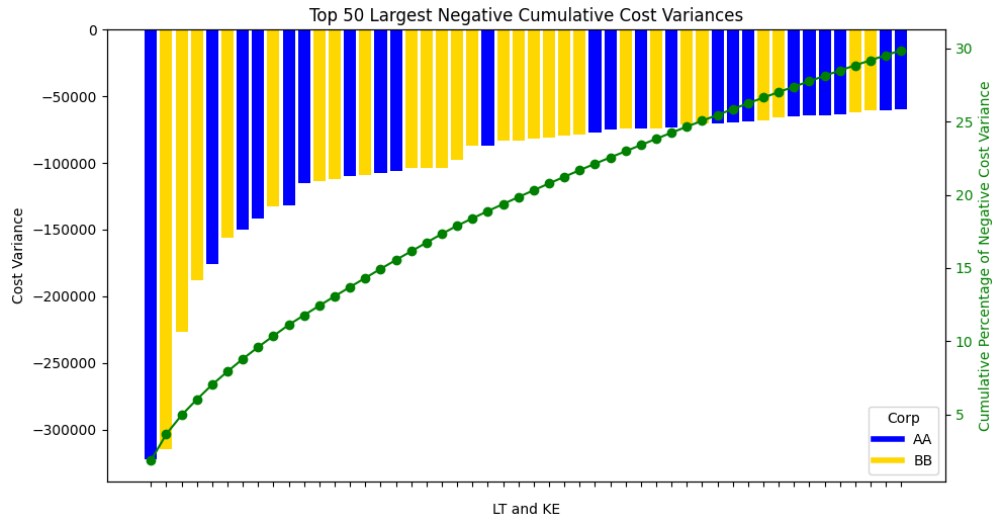


Figure 2.5: 50 LT/KEs with the Largest Negative Cost Variances Across All Hulls

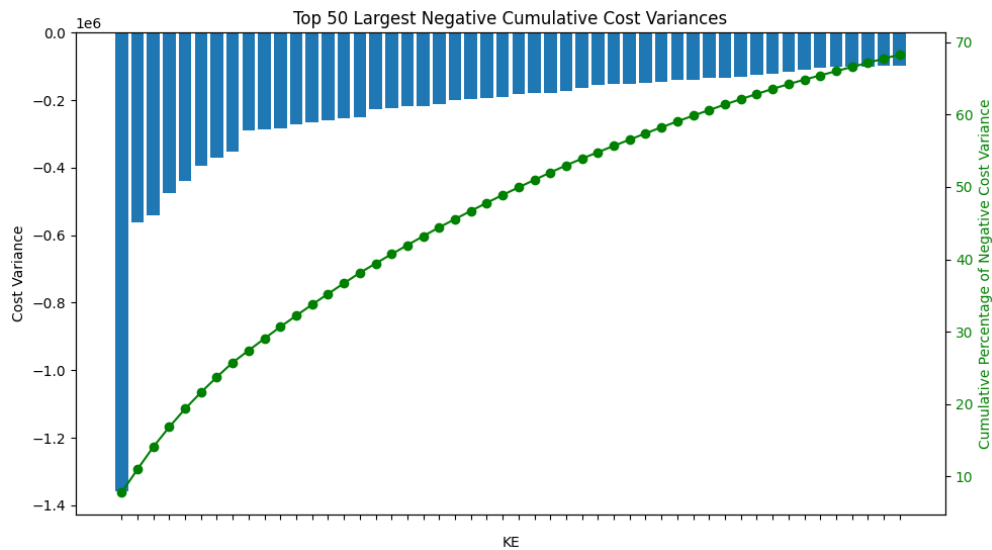


Figure 2.6: 50 KEs with the Largest Negative Cost Variances Across All Hulls

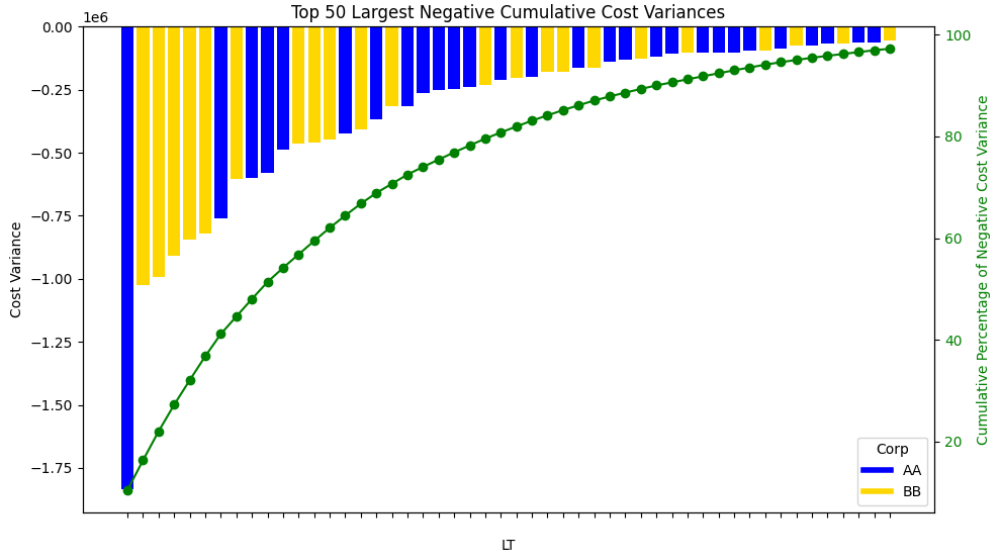


Figure 2.7: 50 LTs with the Largest Negative Cost Variances Across All Hulls

The final question to answer with cost variance is how it changes across hulls. Ideally the negative cost variance would decrease as the predictions improved or the work was performed more efficiently. Choosing the three largest contributors to negative cost variance from Figures 2.5 - 2.7, the resulting changes across hulls are shown in Figures 2.8 - 2.10. These graphs highlight one of the difficulties with this data - when to define the work as complete. Clearly some of the newer hulls are only partially complete on these LT/KE combinations, and therefore are skewing the graphs.

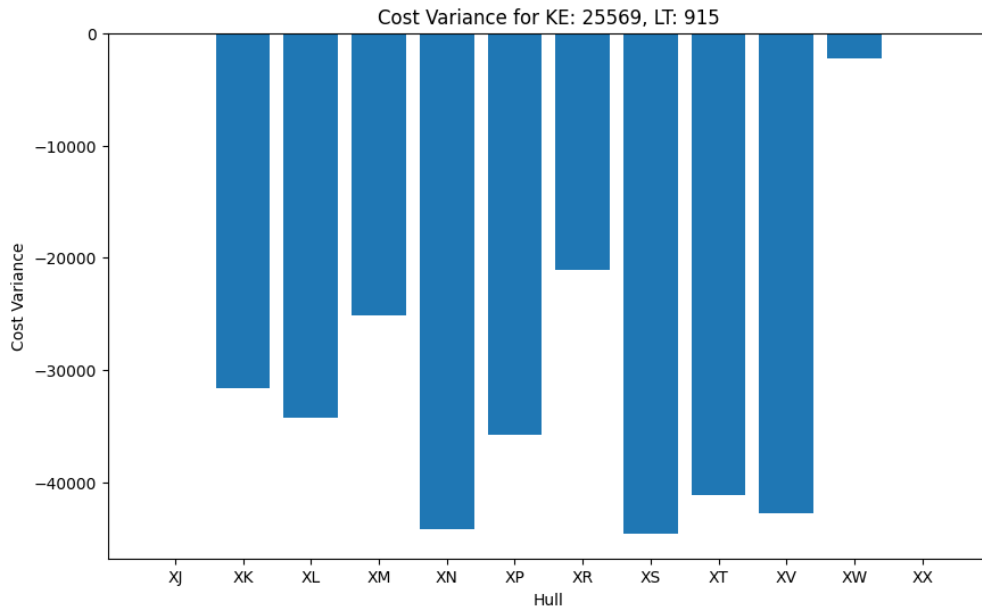


Figure 2.8: Cost Variance Across Hulls for LT:915, KE:25569

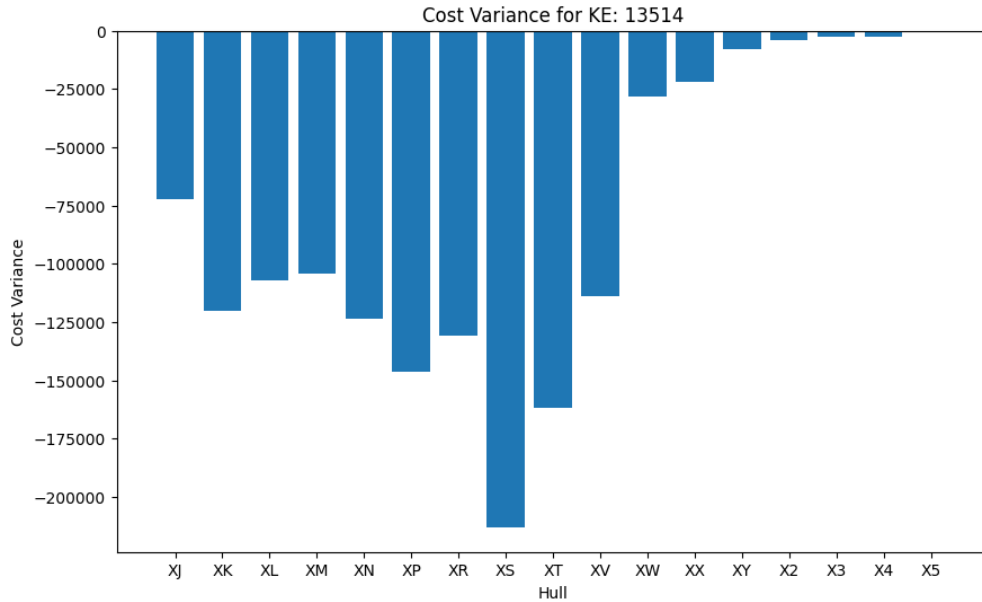


Figure 2.9: Cost Variance Across Hulls for KE:13514

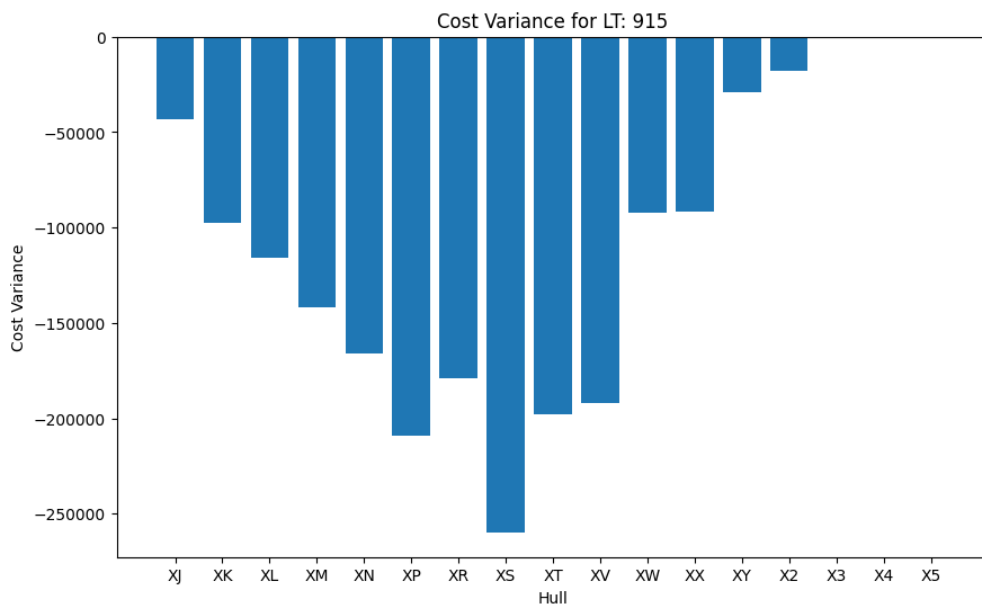


Figure 2.10: Cost Variance Across Hulls for LT:915

In terms of EVM, work is complete when BCWP equals BAC. After applying this across the aggregated groups of LTs, KEs, and the combinations of both, it became clear that it did not make sense to do this at the LT level as only 2 LTs met this criteria across 3 hulls, and none met it over more than 3 hulls. At the KE level there were 45 KEs that were complete on 5 or more hulls, an improvement over LT but the most useful results were still shown at the lowest LT/KE level. After applying the definition of complete as BCWP equals BAC (within some margin shown here using 0.1%), the resulting graphs are shown in Figures 2.11

& 2.12. Even after applying the EVM definition of completed work there are still outliers in the graphs, both satisfying this definition. This highlights another difficulty in this data as the budgeted EVM values can be inconsistent. Regardless, these graphs are still useful and show meaningful trends across hulls.

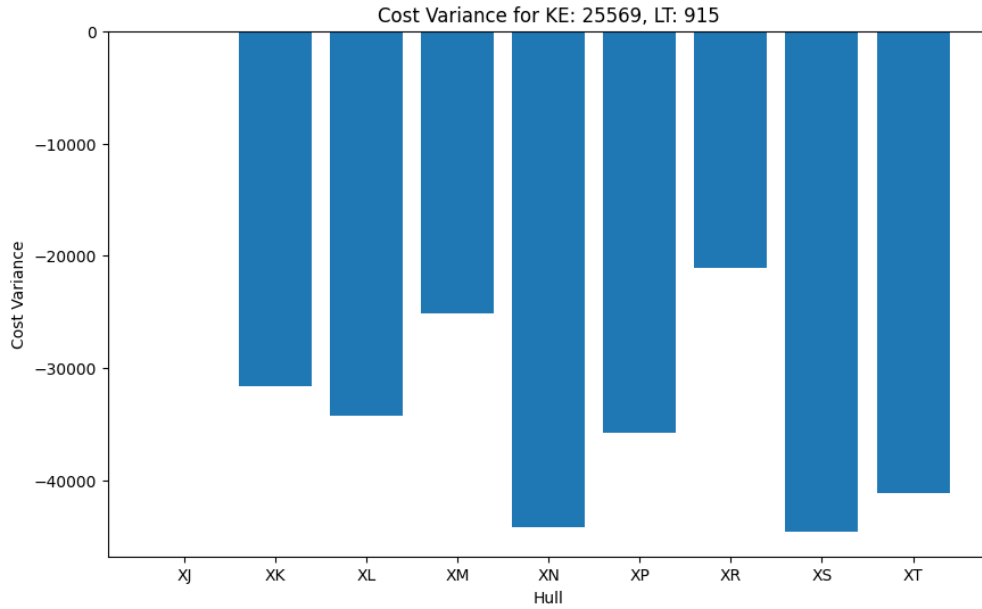


Figure 2.11: Cost Variance Across Complete Hulls for LT:915, KE:25569

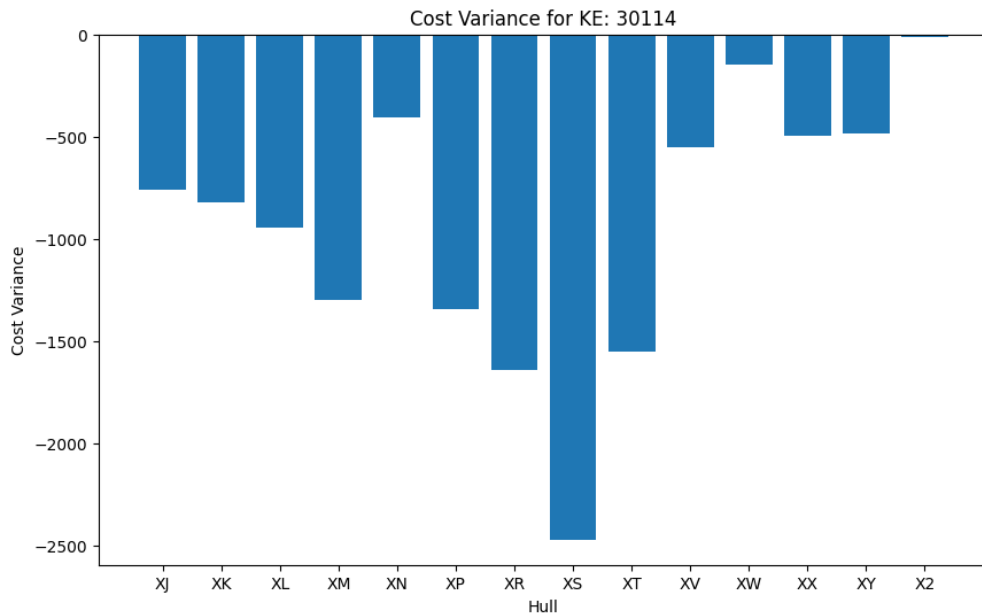


Figure 2.12: Cost Variance Across Complete Hulls for KE:30114

2.3.3 Schedule Analysis

While cost variance is a useful metric to track when assessing trends or problem areas relating to cost, schedule variance poses some issues. Specifically, schedule variance when work is complete is zero because BCWP equals BCWS, whereas cost variance at the end captures the overall difference across the entire job. One thought was to use the maximum (negative) schedule variance, but it was ultimately determined that calculating the time it took to complete the work was a better metric. This was then compared across hulls to determine if the time was improving as hoped, as shown in Figures 2.13 & 2.14. These graphs make it easy to spot outliers and see an overall trend, but with over 7,000 LT/KE combinations this is not as broadly useful. For this reason, the next section discusses finding a metric that can be calculated and compared across any level of the data.

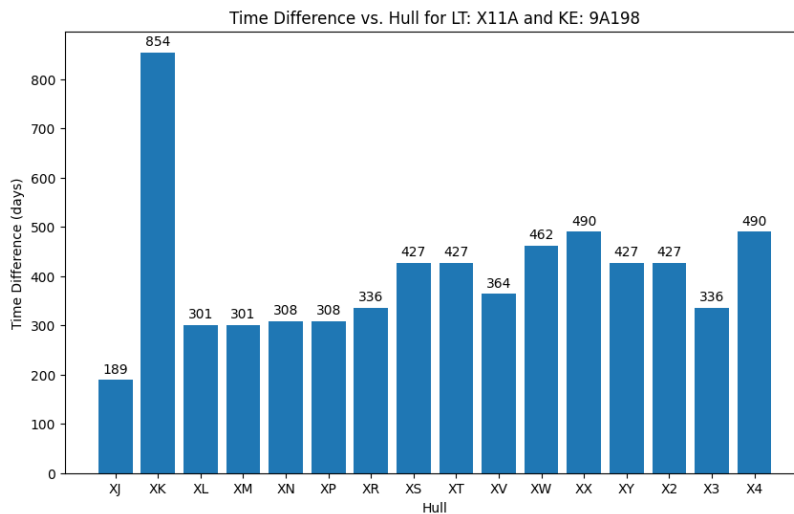


Figure 2.13: Days to Complete Work for LT: X11A, KE: 9A198 Across Hulls

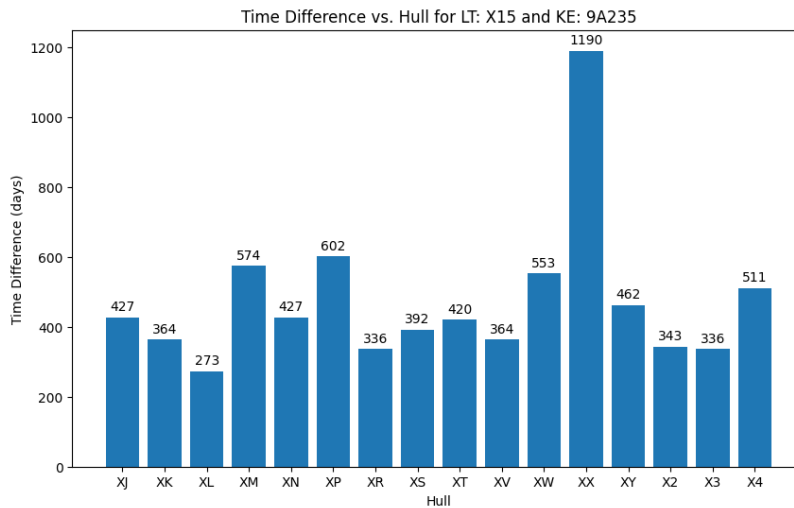


Figure 2.14: Days to Complete Work for LT: X15, KE: 9A235 Across Hulls

2.3.4 Learning Curve

The learning curve is a concept widely used in various fields, depicting the relationship between experience and performance improvement. In the context of project management and cost forecasting, it's employed to understand how costs evolve as a project advances. The model assumes that as more units (hulls in this case) are produced, the cumulative average cost per unit decreases due to increased efficiency and learning. The formula used in the learning curve model is $y = a * x^{-b}$, where 'y' is the cumulative average cost, 'a' is the initial cost per unit, 'x' is the cumulative number of units produced, and 'b' is the learning coefficient [16]. If b is positive, it indicates a positive learning curve or a situation where the cumulative average decreases as the cumulative number of hulls completed increases. If b is negative, it indicates a negative learning curve or a situation where the cumulative average increases as the cumulative number of hulls completed increases. Clearly a positive learning curve is desirable and expected following the assumption that learning & experience increase efficiency.

To implement this concept using ACWP for cost, a function was written to calculate the learning curve metric using the SciPy library for curve fitting to estimate the learning coefficient 'b' [17]. As discussed in Section 2.3.2, because this can only be measured on completed work it should only be analyzed on the LT/KE and KE levels where there is enough data across hulls. The same definition and margin of complete was applied. This function sets the value of 'a' to the ACWP at completion of the first hull. The graphs for the same LT/KE and KE from Section 2.3.2 are shown in Figures 2.15 & 2.16. In both these cases there is a slightly negative learning coefficient and therefore a negative learning curve. A related function was created to iterate through a DataFrame containing various KE and LT combinations, applying the learning curve function and collecting the results in a structured format. As an example, when run on LT/KE combinations with at least 5 complete hulls approximately 55% had negative learning coefficients. Similarly when run on KE combinations also with at least 5 complete hulls nearly 69% had negative learning coefficients. This demonstrates a method for a comprehensive analysis of cost trends in shipbuilding, shedding light on the learning dynamics within different project parameters.

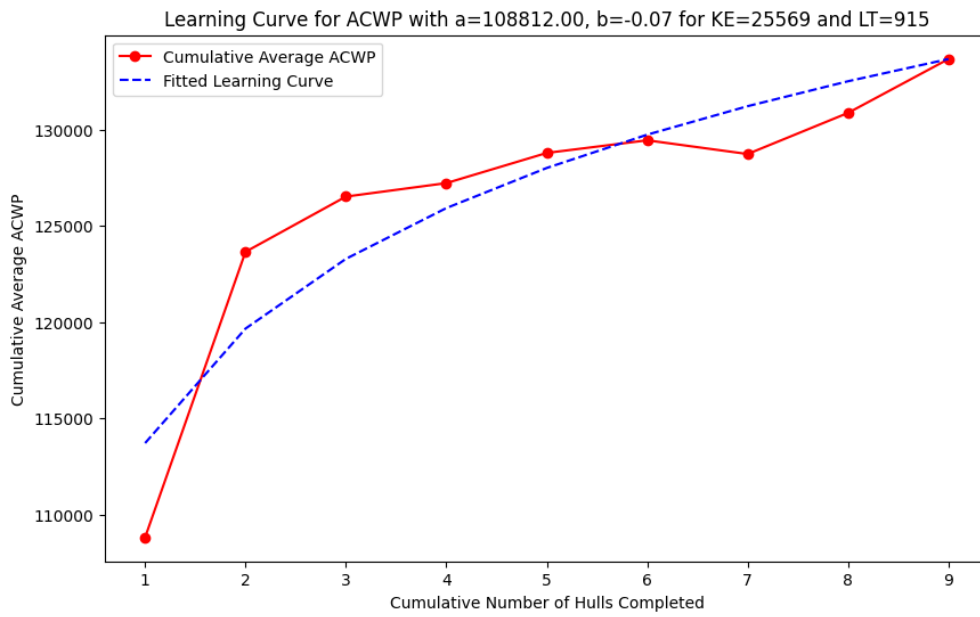


Figure 2.15: ACWP Learning Curve for $LT:915$, $KE:25569$

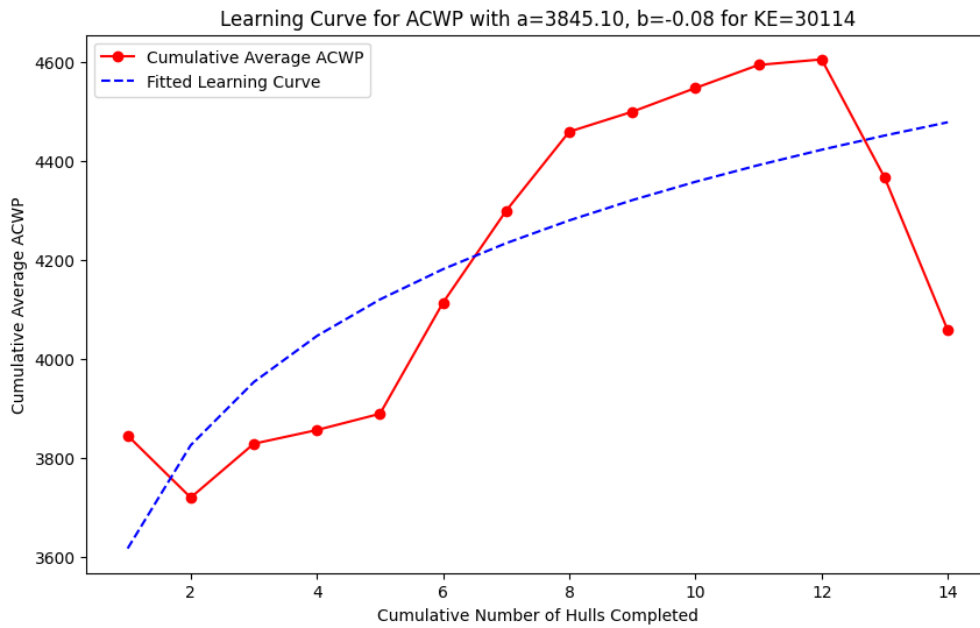


Figure 2.16: ACWP Learning Curve for $KE:30114$

Chapter 3

Machine Learning and Model Selection

3.1 Introduction to Machine Learning

In recent years the integration of machine learning techniques has revolutionized the landscape of data analysis, offering new opportunities for extracting insights and patterns from complex datasets. Machine learning, a subset of artificial intelligence, empowers systems to learn from data patterns, adapt, and make informed predictions or decisions without explicit programming. This chapter delves into the introduction of machine learning and its applicability to the analysis of EVM data in the construction of US naval ships.

EVM serves as a foundational framework for assessing project performance, offering a structured approach to monitor and control costs and schedules. However, as projects grow in complexity, so does the volume and intricacy of the data generated. Traditional analytical methods may face challenges in effectively harnessing this amount of information. Machine learning, with its ability to discern hidden patterns, predict trends, and categorize data, presents a compelling solution for deriving deeper insights from EVM datasets. The dynamic nature of naval ship construction projects, with evolving variables and dependencies, aligns well with the adaptability and predictive capabilities inherent in machine learning models. By leveraging these models, the aim is to enhance the precision of performance assessments and contribute to more informed decision-making processes within the naval construction industry. This chapter will explore various types of machine learning models and their potential to address the specific demands of EVM data.

3.1.1 Types of Machine Learning Models and Applicability

While there are more types of machine learning, four types will be briefly introduced here: supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning [18]. Supervised learning involves training a model on labeled data, making it suitable for predicting predefined outcomes. Unsupervised learning, on the other hand, operates on unlabeled data to reveal hidden patterns and relationships. Semi-supervised learning combines aspects of both, utilizing limited labeled data alongside a larger pool of unlabeled data. Reinforcement learning is a trial and error based method in which rewards and penalties are used to train a model.

In the context of predicting cost and time in the construction of US naval ships, only supervised learning is considered in this thesis. This approach was chosen due to the availability of labeled datasets, where historical EVM metrics are paired with corresponding project outcomes. Supervised learning models, like regression for cost prediction and classification for time estimation, align well with the goal of making accurate predictions based on well-defined labeled data. By concentrating on supervised learning, the aim was to harness the precision and predictive power it offers, tailoring the machine learning approach to the specific needs and characteristics of the EVM data in naval ship construction projects.

3.2 Model Selection Criteria, Considerations and Limitations

3.2.1 Key Criteria for Model Selection

In selecting machine learning models for EVM data analysis, several key criteria were considered to ensure the appropriateness and effectiveness of the chosen models.

Interpretability

In ship construction projects, interpretability is essential for stakeholders to comprehend the factors influencing cost and time predictions. Models like regression, which provides clear relationships between input features and outcomes, enhance interpretability and foster confidence in decision-making. These relationships allow the stakeholders to take corrective actions early on in projects when the predictions indicate a potential problem.

Performance Metrics

Selecting appropriate performance metrics is crucial in assessing the accuracy and efficacy of predictive models in machine learning. Metrics like Mean Squared Error (MSE) and R^2 score are valuable in evaluating regression models, providing insights into the model's ability to capture the variance in the data and the magnitude of prediction errors. However, the choice of metrics should align with the specific objectives of the predictions.

Computational Efficiency

Considering the dynamic nature of naval construction projects, models must be computationally efficient to allow for timely decision-making. The efficiency of the chosen models impacts their feasibility for real-time or near-real-time applications. Efficient models are imperative for timely insights into project performance. Regression models for cost prediction are an excellent example of a computationally efficient model.

3.2.2 Challenges in Applying Machine Learning to EVM Data

Limited Dataset Size

One notable challenge lies in the potential limitations of the dataset size. Machine learning models, particularly complex ones, may require extensive amounts of labeled data for robust training. In scenarios where the dataset is limited, the risk of overfitting or underfitting must be addressed to ensure the generalizability of the models.

Real-World Data Complexity

Naval ship construction projects are inherently complex, involving numerous variables and dependencies. This complexity, coupled with real-world data issues such as noise and human error in data collection and entry, may pose challenges for machine learning models. Noise can introduce variability in the data that does not reflect actual patterns, while human error can lead to inconsistencies or inaccuracies. Identifying models that can effectively process and learn from such complex and potentially noisy data is crucial for accurate predictions.

Handling Outliers and Data Anomalies

EVM data, like any real-world dataset, may contain outliers or anomalies—data points that deviate significantly from the norm due to various factors, including measurement errors, data entry errors, or unexpected events in the project. Some machine learning models are particularly sensitive to outliers, which may yield skewed predictions or biased results. Developing strategies to detect, understand, and address these outliers and anomalies is vital for maintaining the integrity of the analysis and ensuring that the models produce reliable and robust predictions.

3.2.3 Models Not Suitable for EVM Data

Overly Complex Models

Machine learning models that are excessively complex may not be suitable for EVM data analysis, especially if the dataset is not sufficiently large. Models that demand a high number of parameters relative to the dataset size might overfit the data, providing overly optimistic predictions that do not generalize well to new instances.

Sensitivity to Outliers

Models sensitive to outliers (e.g., certain clustering algorithms or distance-based methods) may introduce biases in the analysis of EVM data. Given the presence of potential outliers in cost and time metrics, selecting models robust to extreme values becomes imperative to ensure the reliability of predictions.

3.3 Models Considered

3.3.1 Prophet

Description

Prophet is a forecasting tool developed by Facebook's Core Data Science team, designed for making precise and flexible forecasts, especially for time series data that exhibits strong seasonal effects and historical trends [19]. It is particularly well-suited for data with daily observations that display patterns on different time scales, such as yearly, weekly, and daily seasonality, as well as holiday effects. Prophet was considered as a potential model as it works well with missing data, handles outliers robustly, and is available for open-source use.

Implementation & Decision Rationale

Prophet was not implemented in this thesis primarily because it requires data in a very specific format, which did not align well with the nature of the provided data. Prophet typically expects a time series dataset with two columns: one for the timestamp and one for the metric to be forecasted. This format constraint makes it challenging to use Prophet with EVM data, which is approximately monthly and not strictly regular, thus complicating the direct application of a daily cycle-based model like Prophet. Moreover, the EVM data encompasses multiple input features that influence the forecasts, but Prophet is primarily designed to handle univariate time series forecasting, limiting its ability to directly model the complex, multivariate relationships present in EVM data. Additionally, Prophet's "black box" nature, while offering ease of use and automatic detection of seasonal patterns, also means it lacks transparency in how predictions are derived. This makes it difficult to interpret the model's behavior and understand the influence of different factors on the forecasts, which is a critical aspect for gaining insights and making informed decisions in the context of ship construction. Despite Prophet's strengths in handling seasonality and trend changes in time series data, its structural limitations and lack of transparency made it an unsuitable choice for the specific needs of this thesis.

3.3.2 Autoformer/Transformer

Description

The Autoformer is a time series forecasting model that builds upon the Transformer architecture by adapting it to handle the specific challenges of time series data. Unlike traditional Transformers that treat all parts of the input sequence equally, the Autoformer incorporates a decomposition mechanism that separates the trend and seasonal components from the time series data, allowing for more efficient and accurate modeling of long-range dependencies [20].

This model is designed to automatically identify and model complex patterns in time series data, leveraging attention mechanisms to focus on the most relevant parts of the input sequence for making predictions. The Autoformer excels in handling long sequences and can capture both long-term trends and short-term fluctuations in the data. On platforms like Hugging Face, the Autoformer is made available with pre-trained options, facilitating

easy implementation and experimentation [20]. It is particularly useful for datasets where understanding and modeling intricate temporal dynamics are crucial, making it a powerful tool for advanced time series forecasting tasks.

Implementation & Decision Rationale

The Autoformer was not chosen for implementation in this thesis due to several key factors. First, its requirement for regular, structured time series data did not work well with the EVM data's approximately monthly and somewhat irregular intervals. Similar to Prophet, Autoformer's format and frequency preferences made it less suitable for the nuanced and sporadic nature of EVM data. Furthermore, the Autoformer, being a complex deep learning model, often demands large datasets to train effectively and can be sensitive to outliers, which could skew predictions if not adequately addressed. This model's sophisticated architecture, while powerful for capturing temporal dependencies, also leads to challenges in interpretability. Understanding the "why" behind its predictions is not straightforward, making it difficult to derive clear, actionable insights from its forecasts, which is crucial for this project. These considerations, combined with the potential need for extensive data and computational resources, led to the decision against deploying the Autoformer for this project.

3.3.3 Long Short-Term Memory (LSTM) Network

Description

LSTMs (Long Short-Term Memory networks) are a type of recurrent neural network (RNN) designed to handle the challenges of learning from sequences with long-term dependencies. They are characterized by their unique structure of memory cells and gates (input, forget, and output gates) that regulate the flow of information, enabling them to capture and remember patterns over extended periods [21]. This makes LSTMs highly effective for time series forecasting tasks. Their ability to process and learn from data with varying time intervals and incorporate multiple input features aligns well with the complexities and irregularities of EVM data, providing a useful modeling approach for project cost analysis and forecasting.

Implementation & Decision Rationale

An LSTM model was trained on the EVM dataset using TensorFlow, an open-source machine learning framework known for its robust support for deep learning models [22]. The LSTM model was designed to utilize time series data, one-hot encoded hull identifiers, as well as other EVM metrics like BCWS & BAC to forecast the Actual Cost of Work Performed (ACWP). This setup aimed to leverage the LSTM's strength in handling sequential data to model the intricate temporal relationships and patterns inherent in the ship construction projects' cost and progress data.

Despite the initial implementation, several factors led to the decision that the LSTM model was not the ideal choice for this project. One significant limitation was the model's lack of interpretability; as a complex deep learning model, the LSTM's internal workings and decision-making processes were not easily understandable, making it challenging to

derive actionable insights or explain the forecasts in a transparent manner. Additionally, the available dataset posed constraints: only 4 out of 20 hulls had sufficient data to be considered "complete" (here defined as $BCWP/BAC \geq 99\%$), limiting the amount of training data and potentially affecting the model's ability to learn and generalize across different construction projects. Computational efficiency also emerged as a concern, with the LSTM requiring considerable resources for training and inference, which could be impractical for real-time or large-scale applications. The sensitivity to outliers, a common issue in real-world datasets, might have necessitated further data preprocessing and model tuning efforts to ensure robust predictions. These challenges, combined with the need for a more transparent, data-efficient, and computationally viable model, led to the decision to explore alternative approaches better suited to the constraints and objectives of ship construction prediction.

3.3.4 Autoregressive Integrated Moving Average (ARIMA)

Description

The ARIMA (AutoRegressive Integrated Moving Average) model is a statistical technique for time series forecasting, combining autoregressive (AR) components, differencing (I) for stationarity, and moving average (MA) elements [23]. It leverages past data points to predict future values, adjusting for trends and seasonality through differencing, thus offering a balanced approach for analyzing time-dependent data. An ARIMA model's strength lies in its simplicity, interpretability, and effectiveness in capturing linear relationships and trends in historical data, making it a go-to method for many forecasting tasks where understanding the time series behavior is crucial.

Implementation & Decision Rationale

An ARIMA model was implemented using the statsmodels library, which is well-suited for statistical analysis in Python [24]. However, it became apparent that ARIMA was not the optimal model for this dataset due to several inherent challenges:

- **Non-Stationary Data:** ARIMA models require the data to be stationary, meaning the statistical properties of the series must not depend on the time at which the series is observed. The EVM data exhibited non-stationary characteristics, with varying mean and variance over time, complicating the model's ability to generate accurate forecasts.
- **Frequency Definition Issues:** The approximately monthly data in the EVM dataset, without exact regularity, posed difficulties in defining a specific frequency for the ARIMA model. This irregularity made it challenging to set the appropriate model parameters that align with the data's inherent temporal structure.
- **Long-Term Forecasting Limitations:** ARIMA models are generally better suited for short-term rather than long-term forecasting. Predicting far into the future with ARIMA can be problematic, especially when the future trend deviates significantly from historical patterns, which seemed to be a concern with the extended timeline of ship construction projects.

Given these issues, particularly the non-stationary nature of the data and the challenges with irregular intervals and complex seasonal trends, ARIMA did not prove to be the most effective tool for forecasting ACWP in this context. The need for a model capable of handling the EVM data's unique characteristics led to the exploration of alternatives more suited to these complex, real-world project scenarios.

3.3.5 Regression

Description

Regression models are statistical methods used to estimate the relationships between a dependent variable and one or more independent variables. The goal is to find a linear or non-linear function that best fits the observed data, allowing for the prediction of the dependent variable based on known values of the independent variables. Regression is widely used in various fields for its simplicity, interpretability, and effectiveness in predicting outcomes, making it a fundamental tool for both descriptive and predictive analysis.

Implementation & Decision Rationale

In exploring various machine learning approaches for the EVM data, different regression models were tested using the scikit-learn library and ultimately found to be the most suitable choice [25]. These models stood out for their computational efficiency, allowing for quick processing and analysis, which is crucial to train a large number of models quickly. Their high interpretability is another significant advantage, as it provides clear insights into how different variables affect the ACWP, allowing stakeholders to take action early in the construction process. Additionally, regression models can effectively manage the intricacies of real-world data which is essential for the provided EVM datasets. For these reasons, the decision was made to move forward with a regressive prediction model in this thesis.

Chapter 4

Model Development

4.1 Model Selection

A comprehensive review of models considered for this prediction task is discussed in Section 3.3 to include LSTM, Autoformer, and ARIMA models.

4.1.1 Regression Justification

The choice to use a regression prediction model was primarily due to the following reasons:

Interpretability

- **Clear Understanding of Relationships:** Regression models offer straightforward interpretability, allowing stakeholders to understand how changes in predictor variables (e.g., hull classifiers, BCWS) impact the predicted ACWP.
- **Model Coefficients:** The coefficients of the regression model provide quantitative insights into the magnitude and direction of the relationship between each predictor variable and ACWP. This information can guide project managers in identifying critical factors affecting cost outcomes and taking early action to address issues that would cause an overrun in cost or schedule.
- **Visualizations:** Regression models facilitate the creation of intuitive visualizations such as scatter plots and coefficient plots, which further enhance interpretability. These visual aids help stakeholders grasp complex relationships and trends in the data, fostering better understanding and communication of project cost dynamics.

Computational Efficiency

- **Scalability:** Regression models are computationally efficient, especially for datasets with moderate to large numbers of observations and features. Their scalability makes them well-suited for ship construction projects, which often involve extensive datasets spanning multiple hulls and construction phases.

- **Real-Time Updates:** The computational efficiency of regression models enables real-time or near-real-time updates, allowing for continuous monitoring and retraining as new data becomes available throughout the ship construction process. This capability is essential for maintaining the model's accuracy and relevance in dynamic project environments.

Lower Learning Curve

- **Ease of Implementation:** Regression models have a lower learning curve compared to more complex machine learning algorithms like neural networks or ensemble methods. Their simplicity and intuitive nature make them accessible to project stakeholders with varying levels of technical expertise, including project managers, engineers, and decision-makers.
- **Reduced Training Time:** The straightforward nature of regression models streamlines the model development process, requiring less time and resources for training and implementation.
- **Minimal Hyperparameter Tuning:** Regression models typically have fewer hyperparameters to tune compared to more advanced machine learning techniques. This simplifies the model development process and reduces the need for extensive hyperparameter optimization, further lowering the barrier to implementation.

This does not mean that a regression model is the optimal model to use in this task or that there are not better suited models, but the combination of interpretability, computational efficiency, and lower learning curve are the reasons why regression was chosen for this thesis.

4.1.2 Regression Models Considered

In the process of model selection various regression models were considered and compared against each other using the Mean Squared Error (MSE) metric to assess their suitability and performance. The range of models evaluated include classic linear regression, regularized regression techniques (Ridge and Lasso regression), tree-based methods (Decision Tree and Random Forest regression), and support vector regression (SVR). Additionally, polynomial regression models of different degrees (2nd and 3rd degree) were explored to capture potential nonlinear relationships between predictor variables and ACWP. Each regression model offered distinct advantages and trade-offs in terms of interpretability, flexibility, and predictive accuracy.

The choice of regression model to move forward with has to be balanced with the feature engineering, but eventually a decision has to be made to continue moving forward. Following some initial comparative analysis between models and discussions with my thesis advisor, a combination of polynomial and linear regression was chosen for its ability to balance interpretability with the flexibility to capture nonlinear patterns in the data. This approach leverages the strengths of both regression techniques to provide a robust model for predicting ACWP while ensuring transparency and ease of implementation in practical settings.

4.2 Model Architecture

4.2.1 Model Objective

While there are countless prediction goals that could be useful in a ship construction project, the goal of this specific model is to predict ACWP at the end of construction. The end of construction here is defined as the earliest date when $\frac{BCWP}{BAC} \geq 99\%$. The target variable of ACWP at this point represents the total actual cost of the work done and often exceeds the budget of the project. Even within this specific context there is some uncertainty in determining the date of the end of construction as the schedule often also faces overruns. To isolate the goal of cost prediction in this model the known dates were used during model testing, but this highlights the relationship between cost and schedule in these projects.

The goal of predicting ACWP at the end of construction is important at various milestones throughout the length of the project. Initially the model was designed to predict ACWP at any point within the project without any ACWP data of the specific hull being tested. The model was then adjusted to allow for retraining at a certain milestone (defined as a percentage of $\frac{BCWS}{BAC}$). For example, at a milestone of 15% the model is retrained with the ACWP data up to $\frac{BCWS}{BAC} = 15\%$. This leverages the known data from a project to ideally better predict future ACWP.

4.2.2 Data Granularity

As discussed in Section 2.1.2, the lowest level of the EVM data is a lead trade (LT) within a key event (KE) within a major milestone (MM) for a given hull. From there the data can be aggregated either by LT for different KEs (and/or MMs) or by KE (or MM) for all LTs. After data exploration and based on the goal of predicting final ACWP for a given hull, the decision was made to begin with analysis of a single KE (for a given hull). At a lower level than KE there was too much noise and inconsistency within the data. Even for a given KE the data could be inconsistent (as evidenced in Figure 2.2). Working with my advisor, a specific KE was selected to initially train and test the model based on minimizing these inconsistencies and choosing an event early in the process to maximize the amount of hulls with complete data. Creating and training a model for each KE preserves the unique features & patterns of each, while still allowing an overall ship cost prediction by simple aggregation of each KE model prediction. The graph of ACWP vs. Date for each hull for the chosen KE is shown in Figures 4.1. This graph again highlights some of the inconsistencies with the data and the importance of preprocessing for use in the model.

4.2.3 Regression Model Type, Equation and Interpretation

The regression model used in this project consists of a combination of linear and polynomial regression. To preserve the hull data in the model, the hulls were translated to one-hot encoded categorical features. One-hot encoding is discussed further in Section 4.3. The other features incorporated into the model are the date (treated as number of days from earliest date referred to from this point forward as ordinal date) and BCWS/BAC. A third order polynomial regression is applied to the date, while the other features (hull variables

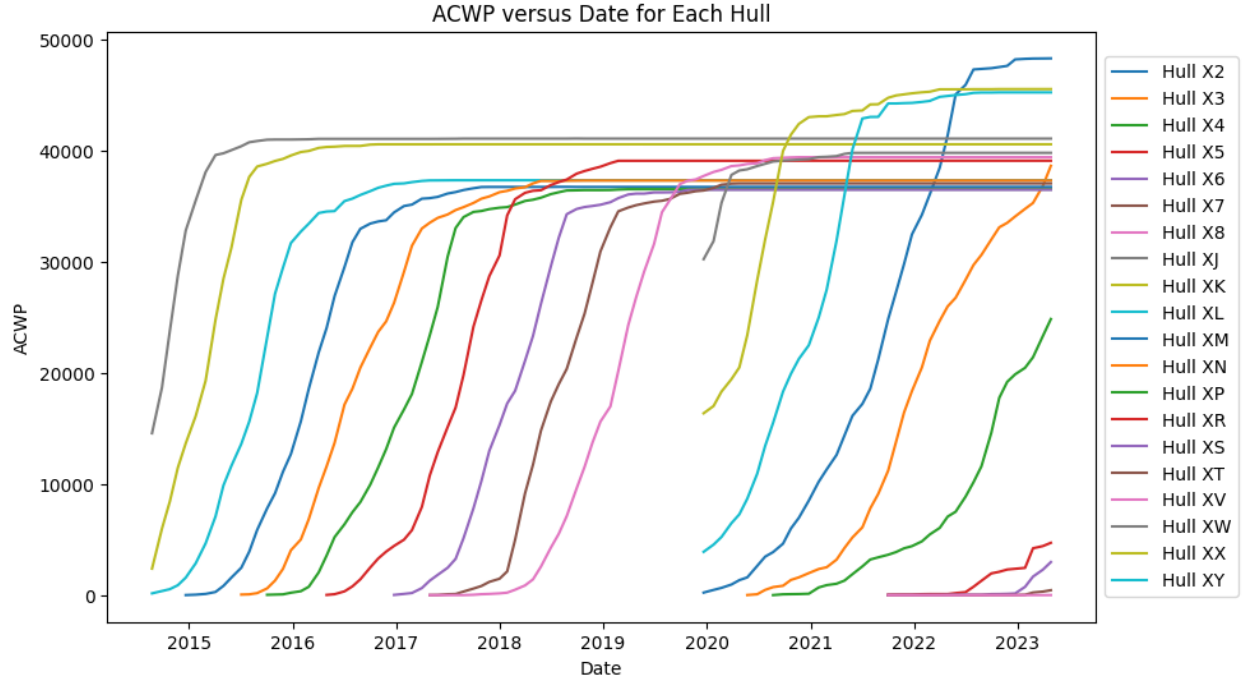


Figure 4.1: ACWP vs. Date for Each Hull

and BCWS/BAC) use linear regression. The features are discussed more in depth in Section 4.4.

Model Equation

The regression equation for predicting ACWP in ship construction projects using this model is shown in Equation 4.1.

$$ACWP = \beta_1 * \frac{BCWS}{BAC} + \sum(\beta_{hull} * Hull\ Dummy\ Variables) + \beta_{x1} * x + \beta_{x2} * x^2 + \beta_{x3} * x^3 + c \quad (4.1)$$

Where:

- $\beta_1, \beta_{x1}, \beta_{x2}, \beta_{x3}$ are the coefficients for $BCWS/BAC$ and the polynomial terms of the ordinal date, respectively.
- $Hull\ Dummy\ Variables$ represent the one-hot encoded hull identifiers, with each coefficient β_{hull} indicating the differential impact of a particular hull compared to the baseline hull.
- x is the ordinal date, transformed into polynomial terms to capture non-linear time effects on ACWP.
- c is the intercept of the regression, representing the baseline ACWP when all predictors are set to zero.

Model Interpretation

A basic interpretation of the components in the model is as follows:

- *BCWS/BAC*: This coefficient β_1 indicates how changes in the planned budget utilization ratio impact the actual costs. A positive coefficient suggests that higher efficiency or faster consumption of the budget relative to the plan increases ACWP.
- *Hull Dummy Variables*: These coefficients β_{hull} measure the influence of constructing specific hulls on the ACWP. Positive values indicate a higher cost relative to the baseline hull, while negative values suggest lower costs. The baseline hull is discussed in Section 4.3.1.
- Ordinal Date Polynomial Terms:
 - $\beta_{x1} * x$: The linear term reflects the direct effect of time on ACWP.
 - $\beta_{x2} * x^2$: The quadratic term captures the acceleration or curvature in the cost-time relationship.
 - $\beta_{x3} * x^3$: The cubic term allows for the modeling of more complex fluctuations in cost over time.
- Intercept (c): The constant term represents the estimated ACWP at the start of the project timeline if it were possible for all other variables to be zero (often theoretical unless all variables are mean-centered).

4.2.4 Model Assumptions

In regression analysis, the reliability and validity of the model are underpinned by several key statistical assumptions. These assumptions ensure that the model provides meaningful and accurate estimates of the relationships between the predictor variables and the response variable. In the context of polynomial regression, the analysis is based on the premise that certain conditions hold. These include the linearity in the parameters, independence of errors, homoscedasticity of errors, normality of the error distribution, and absence of multicollinearity among predictor variables. In the following sections, each of these assumptions are evaluated in the context of this polynomial regression model, assessing their validity and discussing the implications of any deviations from these ideal conditions.

Linearity

In traditional linear regression, the assumption of a linear relationship between predictor variables and the response variable is fundamental. However, in the context of this model the relationship between time (Ordinal Date) and ACWP is inherently non-linear, as evident in Figure 4.1. To address this, the model uses polynomial regression which allows for the modeling of non-linear relationships by incorporating polynomial terms of the predictor variable (Ordinal Date) in the regression equation. This approach provides the flexibility needed to capture the curvature and non-linearity observed in the data, thereby improving the accuracy of predictions.

Despite deviating from the assumption of linearity, the use of polynomial regression enhances the model's flexibility and interpretability. By accommodating non-linear relationships, the model can better capture the complexities inherent in the cost dynamics of ship construction projects. This approach ensures that the model can effectively capture the nuances of the relationship between time and ACWP, leading to improved performance in predicting cost outcomes. Therefore, while the assumption of linearity may not hold, the adoption of polynomial regression provides a suitable framework for modeling the non-linear relationship between time and ACWP in this context.

Independence of Errors

The next assumption assessed in this polynomial regression model is the assumption of error independence. The resulting Durbin-Watson statistic value of 0.01 indicates a positive autocorrelation among residuals. This finding is consistent with the cumulative and increasing characteristics of ACWP, where present values are inherently influenced by past ones. Such autocorrelation underscores the sequential nature of ACWP measurements, thereby deviating from the assumption of error independence in regression models.

While the detected autocorrelation poses methodological challenges, potentially impacting regression coefficient estimates and their standard errors, it aligns with the expected behavior of ACWP data. This property suggests that linear regression may not be suitable for this data context without adjustments or additional techniques accommodating temporal dependency. Despite the deviation from the assumption of independent errors, this does not denote a flaw in the model but rather highlights the necessity for modeling enhancements to better capture the data's inherent sequential nature. Further exploration of such enhancements is delineated as future work in [Section 5.4](#).

Homoscedasticity

The assumption of homoscedasticity, ensuring constant variance of error terms across independent variable levels, is needed in regression models for optimal coefficient estimates and accurate calculation of standard errors. In this polynomial regression model, homoscedasticity is evaluated through a residuals vs. fitted values plot. While slight widening at higher fitted values was observed, potentially indicating heteroscedasticity, its impact on predictive performance may be minor given the model's primary focus on prediction. While ideal adherence to homoscedasticity is recognized, the model's effectiveness in predicting ACWP as indicated by a high R-squared value mitigates practical concerns related to potential heteroscedasticity.

Normality of Residuals

The assumption of normality of residuals in regression analysis is based on the expectation that the error terms should follow a normal distribution. This assumption enables the reliable estimation of confidence intervals and hypothesis tests for the regression coefficients. In the polynomial regression model, the normality of residuals was assessed using a Quantile-Quantile (Q-Q) plot. Ideally, if the residuals were normally distributed, the points on this plot would fall along the reference line. However, the model's Q-Q plot displays deviations

from normality, particularly at the tails of the distribution, suggesting that the residuals may not be perfectly normal, possibly due to outlier values or the non-linear nature of cost over time.

While the assumption of normality is crucial for inferential statistics, it is less critical in the context of prediction-focused models, where the primary objective is the accurate forecasting of values rather than understanding the precise behavior of individual predictors. The slight departure from normality in the residuals of this model does not necessarily impede its predictive capabilities, as evidenced by the high R-squared value. Consequently, for the practical purposes of this study, which centers on forecasting rather than hypothesis testing, the normality of residuals, while noted as a deviation from assumptions, does not diminish the model's utility in predicting future ACWP for ship construction projects.

Multicollinearity

Multicollinearity in regression analysis refers to the phenomenon where predictor variables are highly interdependent, leading to redundancy and instability in the coefficient estimates. Because this regression model incorporates polynomial terms to capture the non-linear progression of ACWP, it inherently introduces multicollinearity. This is because polynomial terms are derived from the same underlying variable (Ordinal Date), causing the predictors (e.g., x , x^2 , x^3) to be correlated. The Variance Inflation Factor (VIF) for these terms confirms their interdependence; high VIF values are indicative of multicollinearity and are expected due to the nature of polynomial regression.

Despite the anticipated multicollinearity introduced by polynomial terms, the model's use of one-hot encoding for hull identifiers with a baseline hull helps prevent the issue for these categorical variables. By designating one hull as the baseline and comparing all other hulls against it, this ensures that only the incremental effects of hull differences on ACWP are modeled, thereby reducing multicollinearity among hull-type predictors. While multicollinearity remains a consideration due to polynomial terms, its effect is somewhat contained within the model structure, and it does not preclude the model from making accurate predictions, as reflected in the model's high R-squared value.

4.3 Data Preprocessing

The initial steps of data cleaning were covered in Section 2.2, but further processing of the data was required prior to training the regression model. As mentioned in Section 4.2.2, this model was initially trained and tested on a single key event, so the data was filtered by this key event. The data was then grouped by hull and date and aggregated by summing BCWP, BCWS, BAC, and ACWP, effectively summing across the different lead trades. The results were then plotted as shown previously in Figure 4.1.

Focusing on the graph of ACWP vs. Date (Figure 4.1), it is clear that only some of the hulls would be useful in training the model. The following steps were implemented to determine the appropriate "complete" hulls:

- Filter for hulls with a final BCWP/BAC of at least 0.99

- Eliminate hulls where the initial value of ACWP was not zero (or near-zero)
- Trim the data:
 - Remove all data after the earliest date when BCWP/BAC is at least 0.99
 - Remove all data when ACWP and BCWP are both zero

It was important to trim the data as described so that the potential long "tails" at either end of the data did not skew the model. The last step at this point in preprocessing was to convert the datetime format "Date" column into the number of days from the first date in the trimmed data (for each hull). This was necessary to create a usable time format for linear regression. The results of this data preprocessing is shown in Figure 4.2.

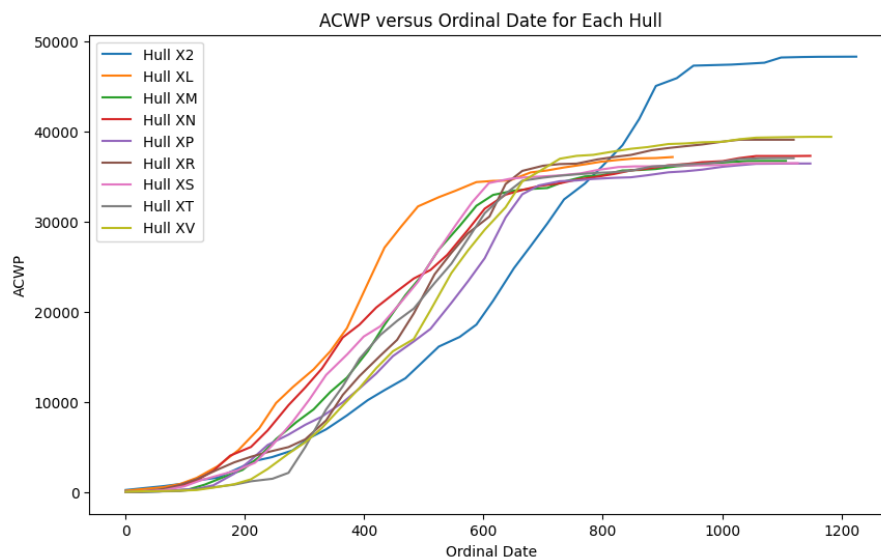


Figure 4.2: ACWP vs. Ordinal Date for Each Hull

The final data preprocessing steps were done within the function `prep_data_with_milestone`, the code for which is shown in Appendix A. This function takes three inputs: a pandas DataFrame (`df`), the string identifier of the test hull (`test_hull`), and a percentage of project "completion" (`milestone`). The outputs of the function are the train and test DataFrames `df_train` and `df_test`, respectively. These DataFrames are created through the following steps inside the function:

- The "Ordinal Date" column of the input `df` is scaled to between 0 and 1 to match the scale of BCWS/BAC. This is done using `MinMaxScaler` from `scikit-learn` [25].
- `df_test` is filtered to only include the test hull.
- `df_train` is filtered to remove the test hull.
- The BCWS/BAC column is defined in both DataFrames.
- `df_test` is split as defined by the milestone based on BCWS/BAC.

- `df_test` is filtered to only contain the data after the defined milestone.
- The data for the test hull prior to the milestone is added to `df_train`.
- The "Hull" column is one-hot encoded and the baseline hull column is removed.
- All other columns not used as features in the regression model are removed.

The returned DataFrames `df_train` and `df_test` from this function are properly formatted for the polynomial regression model.

4.3.1 One-hot Encoding & Baseline Hull

One-hot encoding is a common method for converting categorical variables into a format that can be provided to machine learning algorithms to improve prediction accuracy. This technique transforms each categorical value into a new binary column, where each category is represented by a column of zeros and a single one. This binary representation allows models to better handle categorical data by treating each category as a distinct entity without assuming any ordinal relationship between them. In the context of this model, one-hot encoding is applied to the hull identifiers, turning each distinct hull into its own predictor within the regression framework. This method ensures that the model can assess and quantify the impact of each hull on the project costs independently.

The selection of a baseline hull when applying one-hot encoding is crucial as it serves as the reference category against which the effects of other hulls are compared. By omitting one hull from the binary encoding process, multicollinearity is avoided ensuring that the model coefficients remain interpretable and the computational process stable. The choice of a baseline hull should be guided by the desire for a representative or typical category within the dataset. Often, the most common or the median category is selected as the baseline to provide a meaningful comparison point. Taking this into account and analyzing Figures 4.2 and 4.3 led to the choice of hull XV as the baseline hull.

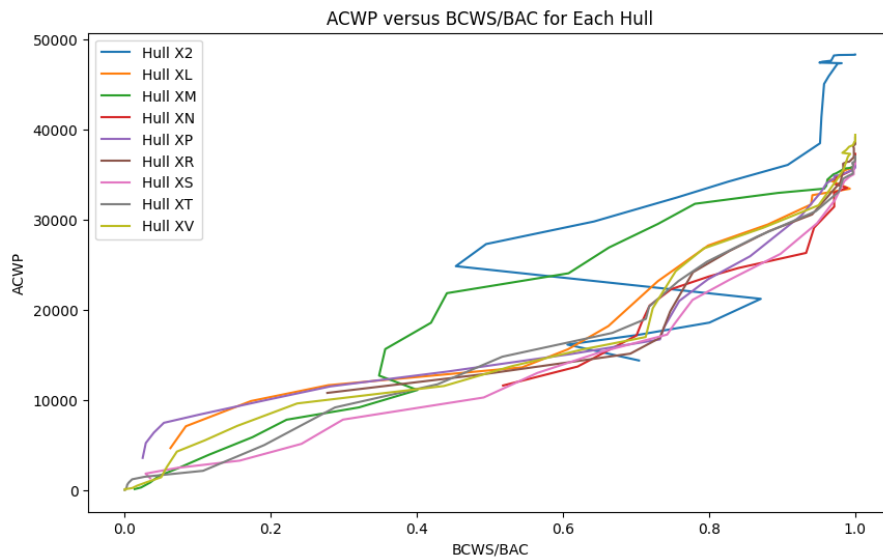


Figure 4.3: ACWP vs. BCWS/BAC for Each Hull

4.4 Feature Engineering

Feature engineering is a crucial component of predictive model development, involving techniques to transform raw data into informative features that boost a model's predictive efficacy. By selecting, creating, or altering features, this process aims to extract pertinent information from data, enhancing the model's capacity to capture underlying patterns and relationships. In this thesis, feature engineering was done in order to refine the regression model's predictive accuracy. Feature engineering began by creating a baseline model that used only time ("Ordinal Date") as a feature and added or modified features from there. These steps are discussed in the following sections.

4.4.1 Feature Preprocessing Techniques

Most of the feature preprocessing is covered in Section 4.3, including the one-hot encoding of the hull variables and the format of the date into number of days for a given hull. By only choosing hulls with "complete" datasets for this KE, there was no need to handle any missing values. The polynomial features are created in the function described in Section 4.5 using PolynomialFeatures from scikit-learn [25]. Scaling of the input features was also done to preserve the interpretability of the model's equation.

4.4.2 Polynomial Features

After deciding on the use of polynomial regression for the "Ordinal Date" feature in the model, the appropriate order of polynomial had to be selected. Following some experimentation with the data, the choice came down to a second or third order polynomial as anything higher unnecessarily complicated the model without any additional value. Second and third order polynomial regression models were then applied to the data and the results were compared. This was done by systematically removing one hull as the test hull, training the model on all the remaining hulls, and comparing the test hull's actual ACWP values to the model's predictions. This was done for every hull (except the baseline hull), and the MSE was averaged across each test hull. The resulting average MSE values for the data were 1.95×10^7 and 1.72×10^7 for second and third order, respectively. Because these errors are relatively close and have the same order of magnitude, the graphs of the results were also very important to consider. The graphs of predicted vs. actual ACWP values for test hull XR are shown below in Figures 4.4 & 4.5.

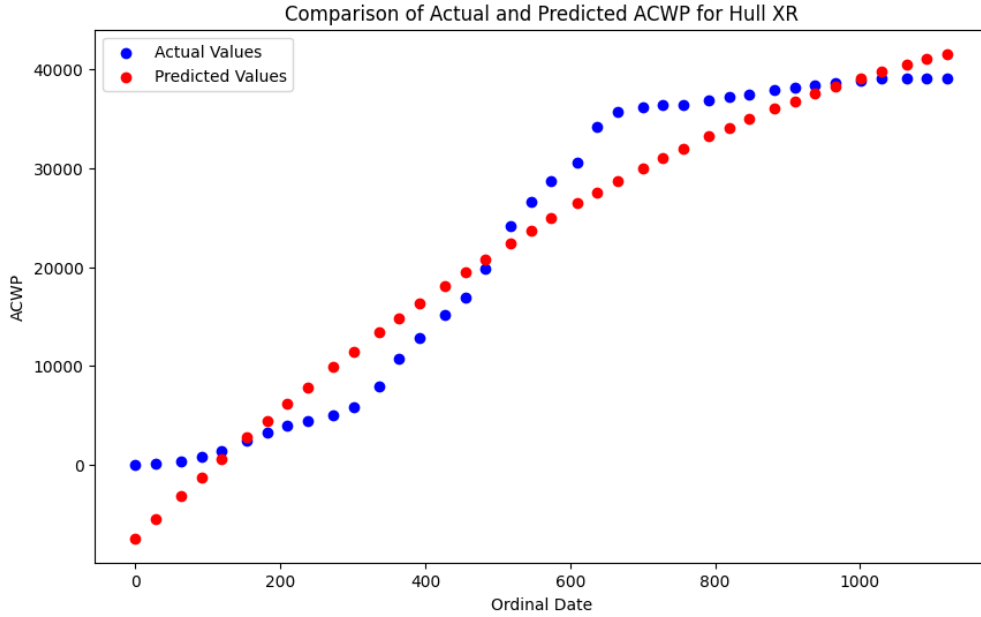


Figure 4.4: Actual & Predicted ACWP for Test Hull XR with Second Order Polynomial

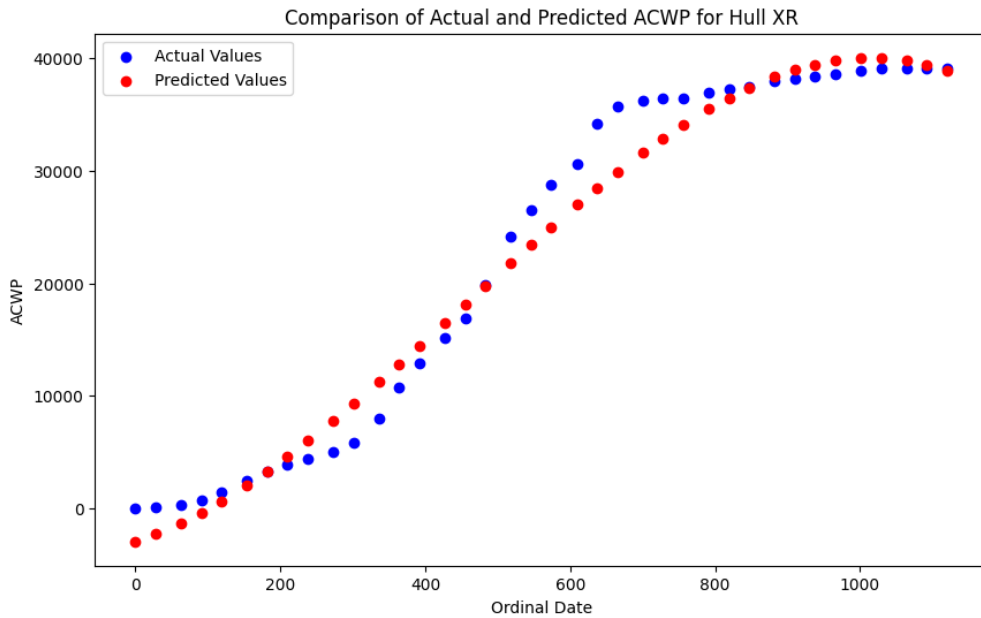


Figure 4.5: Actual & Predicted ACWP for Test Hull XR with Third Order Polynomial

Postprocessing of Predictions

Figures 4.4 & 4.5 also highlight the need for some postprocessing of the predictions based on rules for ACWP. ACWP is a cost metric that is both nonnegative and increasing, so these rules were enforced on the predictions. Specifically, the postprocessing code shifts all negative predictions to zero and applies an `enforce_increasing` function on the predictions

that chooses the maximum value between the current and previous prediction. The results for hull XR are shown in Figures 4.6 & 4.7. With the added postprocessing of the predictions, the average MSE values for the data dropped to 1.68×10^7 and 1.43×10^7 for second and third order, respectively. With the postprocessing, the shape of the third order polynomial is more consistent as it flattens out at the end as compared to the still increasing second order polynomial. This in addition to the lower MSE led to the decision to move forward with the third order polynomial for the "Ordinal Date" feature.

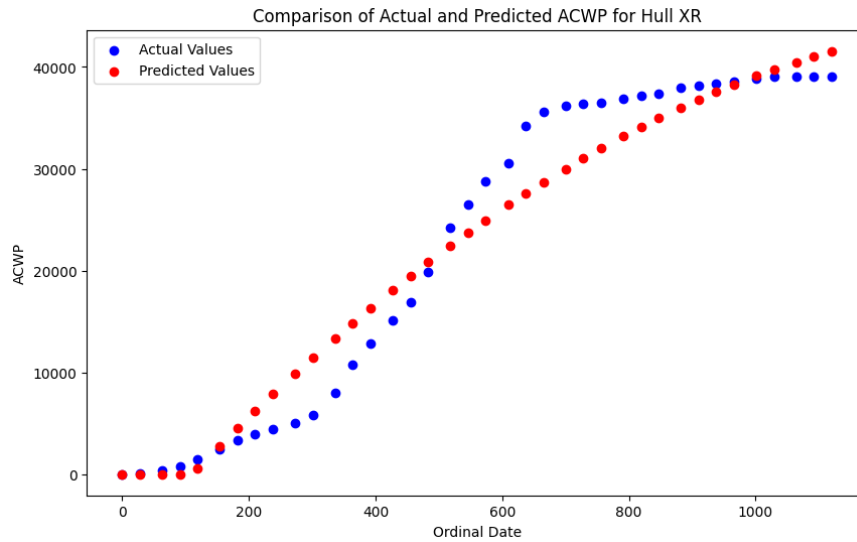


Figure 4.6: Actual & Predicted ACWP for Test Hull XR with Second Order Polynomial & Postprocessing

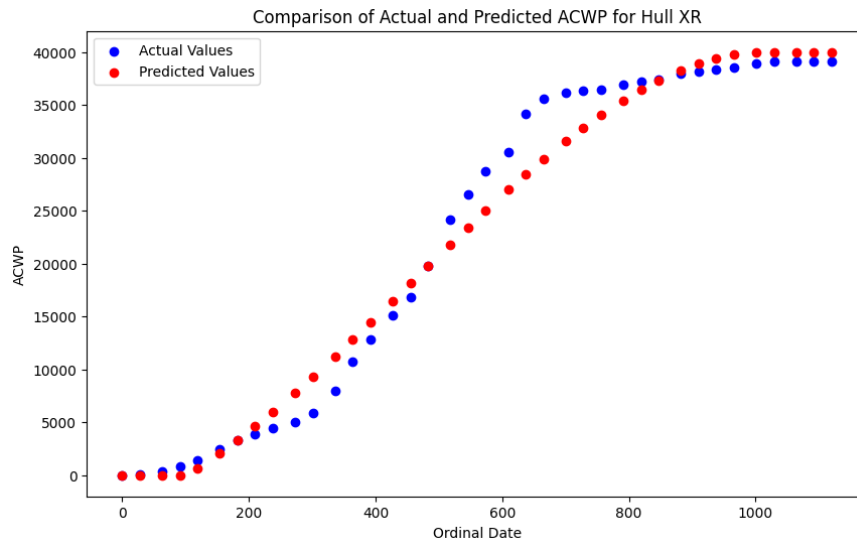


Figure 4.7: Actual & Predicted ACWP for Test Hull XR with Third Order Polynomial & Postprocessing

4.4.3 Feature Engineering Results

Following the decision to apply a third order polynomial to the "Ordinal Date" feature, the next steps were determining what other features should be included in the model. This was done by adding one feature at a time and comparing the results and test metrics.

BCWS

The first feature that was added to the model was BCWS. BCWS was chosen as it represents the scheduled budget of the project and is determined prior to the start of construction. It also provides a good indicator of where a project is projected to be both with cost and schedule. The decision to add it as a linear term to the regression model was based on the relationship shown in Figure 4.8. While not perfectly linear, there is enough of a linear trend that it was worth exploring the effect on the model. With BCWS added, the average MSE dropped to 1.08×10^7 and the resulting graph for hull XR is shown in Figure 4.9.

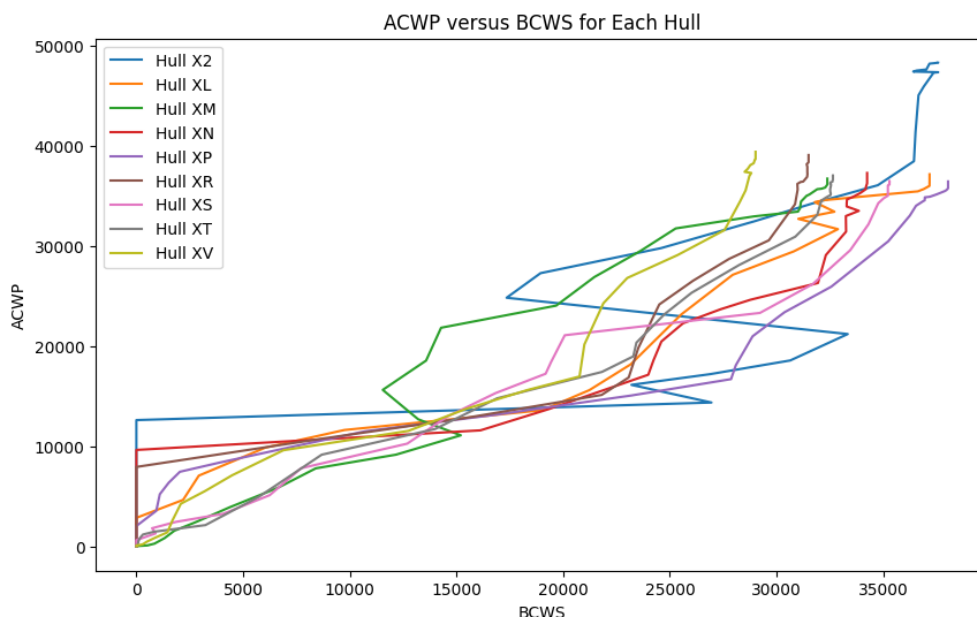


Figure 4.8: Actual Cost (ACWP) vs. Budgeted Cost (BCWS) for Each Hull

BAC

The next potential feature considered was BAC, however BAC itself should be a constant value that does not change with time or ACWP as it simply represents the budget at completion. In order to still incorporate BAC into the model, the idea of adding BCWS/BAC as a sort of budget utilization ratio was tested. As alluded to in previous sections describing the final model, this feature was implemented instead of BCWS and the results are shown in the final results of the model.

No other features were assessed in this thesis due to time limitations, but further feature engineering is discussed as potential future work in Section 5.4.

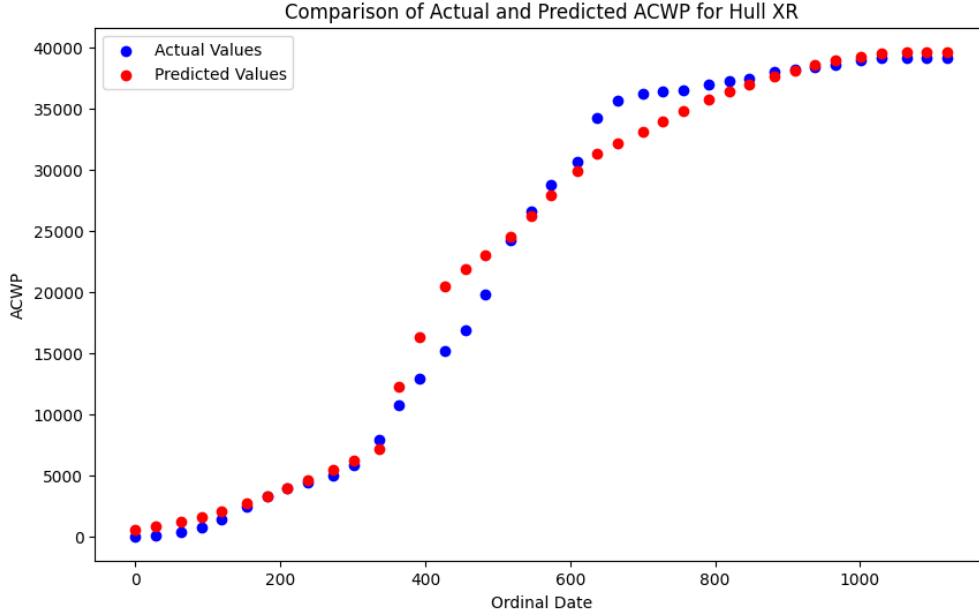


Figure 4.9: Actual & Predicted ACWP for Test Hull XR with BCWS

4.5 Model Training

As alluded to in Section 4.4.2, one hull at a time was removed from the data as a test hull and the model was trained on the data from the remaining hulls. This technique is an adaptation of the Leave-One-Out Cross-Validation (LOOCV) method except instead of removing a single data point, here an entire hull’s dataset is removed [26]. This validation method is typically used to ensure that the model does not overfit to the data and can generalize well to unseen data. This technique is well-suited to this scenario because it simulates using historical data to predict ACWP for a new hull.

Parameter tuning, particularly the selection of polynomial degree, was performed manually with an emphasis on optimizing performance metrics MSE and R-squared (discussed in Section 4.4.2). This approach not only aids in preventing overfitting but also ensures that the model retains its predictive power when applied to unseen data. The choice of validation technique and careful parameter adjustment focus on the ability to provide reliable ACWP predictions, the ultimate goal of this model.

4.5.1 Milestone Prediction

Up to this point most of the discussion has been around predicting ACWP for a test hull in which no data for the hull is known or trained on. This represents the performance of the model before a project begins or shortly after, with little to no useful data recorded. While this is still useful to the shipbuilder, perhaps even more valuable is a model that takes into account the construction data up to the current time for a given hull and leverages it to aid the prediction model. In the discussion about data preprocessing in Section 4.3, the idea of inputting a construction milestone was introduced. This milestone was defined as

a percentage of BCWS/BAC, which is also a feature of the model. By inputting a certain milestone, the data is split such that the model is trained on the data up to that milestone of the test hull along with all the data from the other hulls. The ACWP predictions are then compared to the actual ACWP values of the test hull after the chosen milestone.

The model training was done within the function `poly_regression_final`, the code for which is shown in Appendix A. This function takes the following steps up to and including the training of the linear regression model:

- Prepares features for training:
 - Creates a DataFrame of all features on which a linear model is trained (BCWS/BAC, all one-hot encoded hulls).
 - Creates a DataFrame of only "Ordinal Date" for polynomial transformation.
 - Creates a DataFrame of the target variable, ACWP.
- Prepares features for testing (same as above).
- Applies polynomial transformation to "Ordinal Date" using PolynomialFeatures from scikit-learn [25].
- Combines the transformed polynomial features with the other features.
- Fits a linear regression model to the training data using LinearRegression from scikit-learn [25].

4.6 Model Evaluation

The final model was evaluated and trained in the function `poly_regression_final`, the code for which is shown in Appendix A. The steps up to the model training in this function are described in Section 4.5. The remaining steps in the function do the following:

- Print the regression equation of the trained model.
- Make predictions on the test data using the trained model.
- Apply postprocessing to the predictions as described in Section 4.3.
- Evaluate and print performance metrics MSE and R-squared.
- Optionally plot the predicted vs. actual ACWP values.

MSE and R-squared were chosen as the performance metrics for evaluating this model for several reasons. MSE is useful as it measures the average magnitude of the model's prediction errors, providing a clear and straightforward indication of model accuracy in terms of how close the predicted values are to the actual values. This makes MSE practical for assessing the performance of regression models where minimizing error is a priority. R-squared offers a measure of the proportion of variance in the dependent variable that is predictable from

the independent variables, providing insight into the goodness of fit of the model. It is a relative measure, making it particularly useful for comparing the predictive power of models. Together, these metrics provide a comprehensive view of both the absolute accuracy (via MSE) and the relative efficacy (via R-squared) of the model.

Based on discussions with the shipbuilders, it was determined that results at the following milestones would be insightful: 15%, 25%, 50%, & 75%. The performance metrics at these milestones along with pre-construction (0%) are summarized in Table 4.1 for each test hull.

Table 4.1: MSE [$\times 10^6$] and R-squared for Each Milestone and Test Hull

Milestone	0%		15%		25%		50%		75%	
	MSE	R ²	MSE	R ²	MSE	R ²	MSE	R ²	MSE	R ²
X2	36.5	0.89	53.6	0.63	53.6	0.63	53.6	0.63	64.4	0.32
XL	10.0	0.95	7.19	0.91	6.48	0.90	6.34	0.88	7.72	-0.01
XM	6.54	0.97	8.58	0.92	9.07	0.88	8.88	0.20	9.41	-3.37
XN	4.19	0.98	11.4	0.81	11.4	0.81	11.4	0.81	6.48	0.60
XP	5.80	0.97	13.6	0.83	14.1	0.80	12.9	0.73	5.61	0.70
XR	3.65	0.98	5.75	0.93	5.75	0.93	4.67	0.91	2.32	0.87
XS	4.07	0.98	5.64	0.95	4.95	0.94	4.92	0.91	5.45	0.80
XT	2.74	0.99	2.57	0.97	2.52	0.97	2.73	0.94	3.29	0.78
Average	9.19	-	13.5	-	13.5	-	13.2	-	13.1	-

Analyzing the MSE and R-squared values for different milestones across various test hulls in the polynomial regression model reveals significant insights into the model’s performance and adaptability. Initially, the model displays consistent and stable performance for early project stages across several hulls, suggesting it effectively captures the general cost trends. This is also driven by a lack of fidelity in BCWS/BAC (i.e. large, unevenly spaced jumps). However, as projects progress beyond 50%, significant fluctuations and deteriorations in model performance are observed, particularly for hulls like XM and XL where R-squared values drop sharply, even turning negative at the 75% milestone. This indicates potential issues to include overfitting, where the model learns noise rather than useful predictive signals from the training data.

Despite the challenges observed with the model’s performance at higher milestones, where increased variability and potential overfitting impact its accuracy, the model still holds substantial value for predicting ACWP. Particularly at earlier stages of the projects, the model demonstrates predictive capabilities, providing reliable estimates that can help in early-stage financial planning and resource allocation. Its ability to capture and reflect the general cost trends across different hulls up to the mid-project milestones allows project managers to make informed decisions based on predicted expenditures. This predictive utility, especially in the initial and middle phases of construction, facilitates proactive project management and cost control, helping to mitigate risks associated with budget overruns and scheduling delays.

The equations and plots are shown below for hull XS at each of these milestones. Hull XS was chosen as it had higher fidelity data, specifically with respect to BCWS/BAC.

Milestone: 0%

Model Equation:

$$ACWP = 17251 * \frac{BCWS}{BAC} + 2951 * Hull_{X2} + 2333 * Hull_{XL} + 482 * Hull_{XM} + 73 * Hull_{XN} - 955 * Hull_{XP} + 612 * Hull_{XR} + 365 * Hull_{XS} - 712 * Hull_{XT} + 5000 * x + 75227 * x^2 - 57920 * x^3 - 740$$

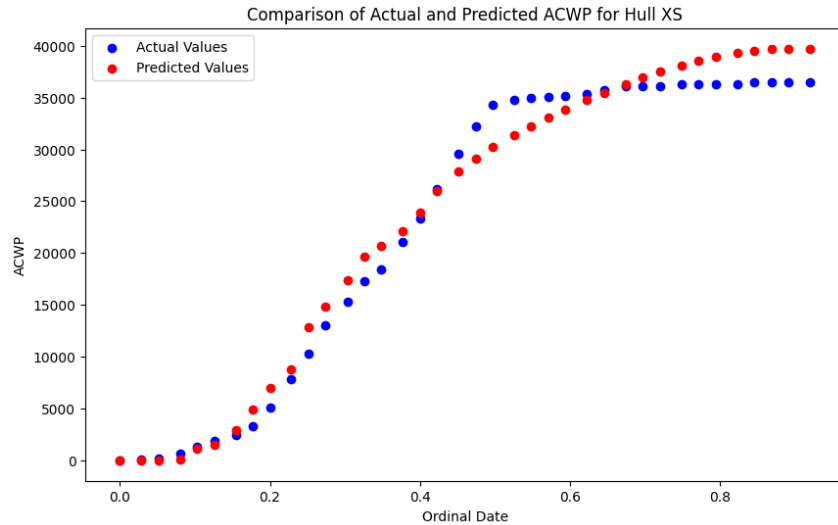


Figure 4.10: Actual & Predicted ACWP for Test Hull XS at 0% Milestone

Milestone: 15%

Model Equation:

$$ACWP = 17230 * \frac{BCWS}{BAC} + 2948 * Hull_{X2} + 2336 * Hull_{XL} + 483 * Hull_{XM} + 75 * Hull_{XN} - 955 * Hull_{XP} + 613 * Hull_{XR} - 711 * Hull_{XT} + 5272 * x + 74722 * x^2 - 57625 * x^3 - 767$$

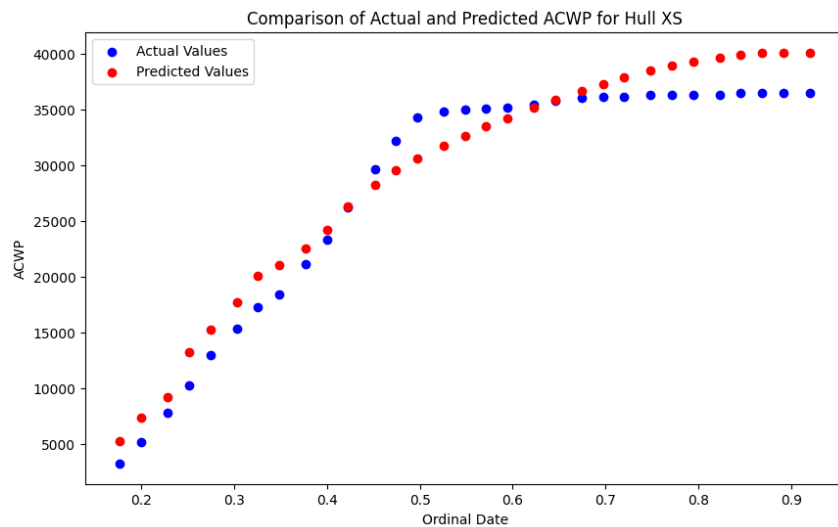


Figure 4.11: Actual & Predicted ACWP for Test Hull XS at 15% Milestone

Milestone: 25%

Model Equation:

$$ACWP = 17246 * \frac{BCWS}{BAC} + 2951 * Hull_{X2} + 2333 * Hull_{XL} + 482 * Hull_{XM} + 73 * Hull_{XN} - 955 * Hull_{XP} + 612 * Hull_{XR} - 112 * Hull_{XS} - 712 * Hull_{XT} + 4374 * x + 76649 * x^2 - 58814 * x^3 - 676$$

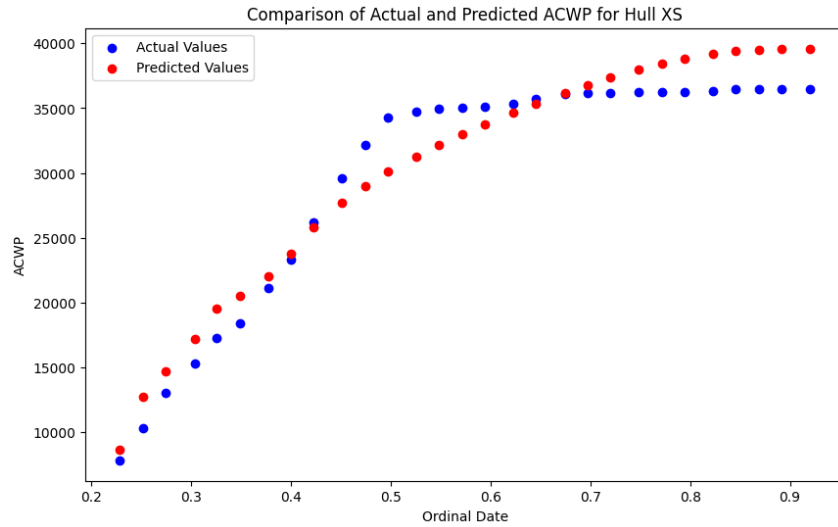


Figure 4.12: Actual & Predicted ACWP for Test Hull XS at 25% Milestone

Milestone: 50%

Model Equation:

$$ACWP = 17161 * \frac{BCWS}{BAC} + 2942 * Hull_{X2} + 2343 * Hull_{XL} + 481 * Hull_{XM} + 77 * Hull_{XN} - 958 * Hull_{XP} + 611 * Hull_{XR} - 426 * Hull_{XS} - 712 * Hull_{XT} + 3963 * x + 77992 * x^2 - 59745 * x^3 - 631$$

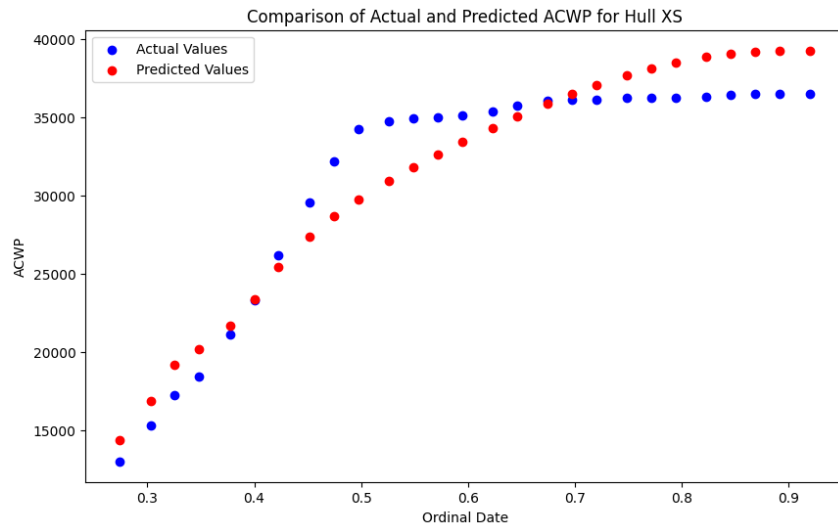


Figure 4.13: Actual & Predicted ACWP for Test Hull XS at 50% Milestone

Milestone: 75%

Model Equation:

$$ACWP = 16983 * \frac{BCWS}{BAC} + 2922 * Hull_{X2} + 2362 * Hull_{XL} + 480 * Hull_{XM} + 87 * Hull_{XN} - 962 * Hull_{XP} + 611 * Hull_{XR} - 765 * Hull_{XS} - 709 * Hull_{XT} + 3556 * x + 79743 * x^2 - 61012 * x^3 - 582$$

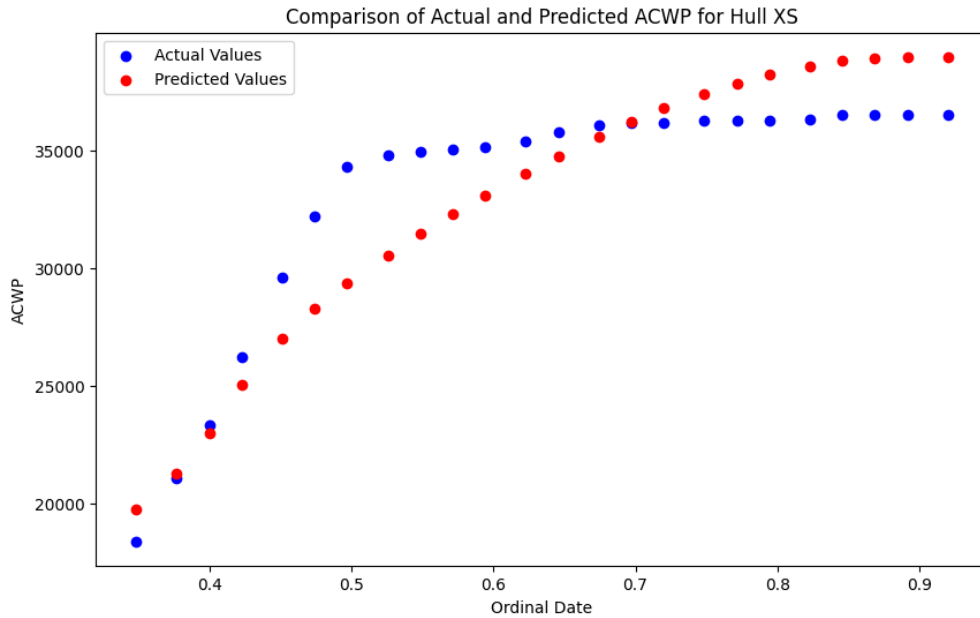


Figure 4.14: Actual & Predicted ACWP for Test Hull XS at 75% Milestone

In analyzing the polynomial regression model for hull XS, notable changes in the coefficients were observed across various project milestones. Initially, at the 15% milestone, coefficients for BCWS/BAC and the hull XS indicator ($Hull_{XS}$) are positioned to reflect early project estimations, with $Hull_{XS}$ showing a positive influence on ACWP. As the project progresses through 25%, 50%, and up to the 75% milestone, there is a gradual decrease in the coefficient for BCWS/BAC, suggesting a diminishing rate of cost increase as the project nears completion, which aligns with a typical project expenditure curve where major costs are incurred during the middle phases. Additionally, the coefficient for $Hull_{XS}$ shows a decline, turning increasingly negative, indicating that as more specific data from hull XS becomes available, its deviation from baseline predictions becomes more pronounced, possibly due to unique project complexities or management efficiencies becoming more influential.

4.7 Sensitivity Analysis

Sensitivity analysis is an important component of predictive modeling that assesses how the variation in the output of a model can be attributed to different sources of variability in its inputs. This process is fundamental for validating the robustness and reliability of models, especially when they are used for decision-making in complex and dynamic environments. By systematically varying key input parameters and observing the resulting changes in the

output, sensitivity analysis provides insight into which factors most significantly influence the model’s predictions.

One-at-a-Time (OAT) sensitivity analysis is an approach where each input variable of a model is varied independently while holding all other inputs constant to observe the resulting change in the model’s output [27]. This technique is used for its simplicity and interpretability, providing clear insights into the direct relationship between individual variables and the outcome. Employing OAT sensitivity analysis to this regression model allows for evaluation of the individual contribution of input variables to the cost predictions.

4.7.1 Ordinal Date

For the purpose of sensitivity analysis the model was trained on all hulls without removing a test hull. The first variable analyzed with the OAT method was Ordinal Date. Because Ordinal Date is scaled as described in Section 4.3, this analysis was conducted over ordinal dates varying from zero to one. BCWS/BAC was held constant at its mean value, and all hull identifiers were set to false thus aligning with the baseline hull. The resulting ACWP predictions were then plotted against the varying Ordinal Date as shown in Figure 4.15.

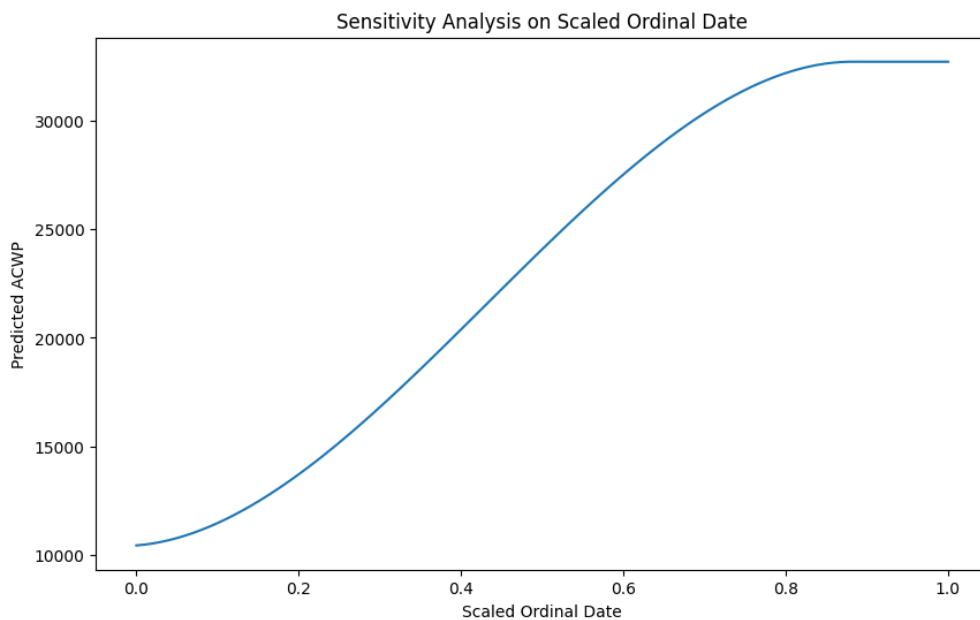


Figure 4.15: Sensitivity Analysis: Predicted ACWP vs. Ordinal Date

Figure 4.15 reveals a smooth and progressive relationship between the Scaled Ordinal Date and Predicted ACWP, validating the model’s continuous and stable predictive capacity over time. The observed trend indicates a gradual increase in ACWP with time, suggesting that project costs naturally accumulate as work progresses. Notably, the curve begins with a modest incline, gains momentum, and eventually levels off, which mirrors the typical cost behavior in ship construction—initially conservative, escalating with active development, and stabilizing as completion approaches.

4.7.2 BCWS/BAC

The next step was assessing ACWP when varying BCWS/BAC from zero to one and holding the remaining variables constant. Ordinal Date was held constant at 0.5, the midpoint of the scaled timeline, while the hull identifiers were again all set as false to align with the baseline hull. The resulting ACWP predictions were plotted against the varying BCWS/BAC as shown in Figure 4.16.

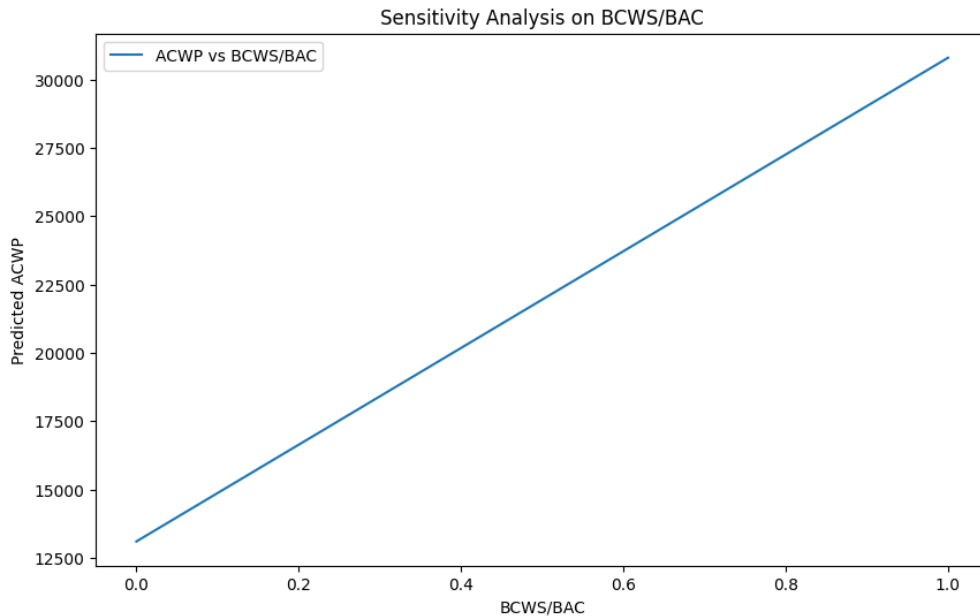


Figure 4.16: Sensitivity Analysis: Predicted ACWP vs. BCWS/BAC

Figure 4.16 presents a clear linear correlation between BCWS/BAC and Predicted ACWP while holding the scaled date constant. This depicts a model where costs predictably rise in direct proportion to project completion, without nonlinear escalation or unexpected cost spikes as completion nears. Such uniform sensitivity across the BCWS/BAC spectrum implies that cost forecasts can be reliably based on project progress, a valuable insight for managing budgets effectively. The linear trend suggests that early project cost estimates remain consistent throughout the project lifecycle, providing a straightforward interpretation that underscores the model's predictive strength. However, it's essential to validate whether this simplified relationship fully captures the nuances of actual cost behavior as projects advance, to ensure the model's applicability in real-world scenarios.

4.7.3 Hull Identifier Variables

Conducting sensitivity analysis on the hull identifiers was deemed unnecessary for several reasons. Primarily, the hulls represent individual construction projects that are expected to share similar structural and cost characteristics, rendering the analysis of hull-specific effects less informative for the overall objective. Additionally, sensitivity analysis on categorical variables like these can be computationally intensive and complex, especially when the num-

ber of categories is large. This complexity does not align with our goal of a streamlined and computationally efficient analytical process. The potential computational overhead and limited actionable insights that such an analysis might offer do not justify its inclusion at this point.

4.8 Results Analysis

Perhaps the best way to evaluate the potential contribution of this model is to compare it to prediction metrics currently in use. Within EVM there is a metric known as Estimate at Completion (EAC) which is used to forecast the total cost of a project upon its completion based on the project’s current performance. There are several methods for calculating EAC, including using actual costs to date, projecting future costs based on current performance trends, or incorporating management’s revised estimates. For the purpose of comparison to the model, a composite EAC formula as shown in Equation 4.2 was used. The individual terms of the equation are as defined in Section 1.2.

$$EAC_{Composite} = ACWP_{CUM} + [(BAC - BCWP_{CUM}) / (CPI_{CUM} * SPI_{CUM})] \quad (4.2)$$

Because EAC only predicts the cost at the end of a project, the comparison to the model’s predictions was only done at this single point. This highlights another benefit of the prediction model in that it predicts the ACWP at any point within the project, not just at the end. The prediction model was run for a variety of milestones and on each test hull, recording the results. These were then compared to calculations of EAC under these conditions. In order to effectively visualize these results, bar graphs were created for each test hull comparing the results at each evaluated milestone. The graph for hull XP as the test hull is shown in Figure 4.17, while the plots for all other test hulls are shown in Appendix B.

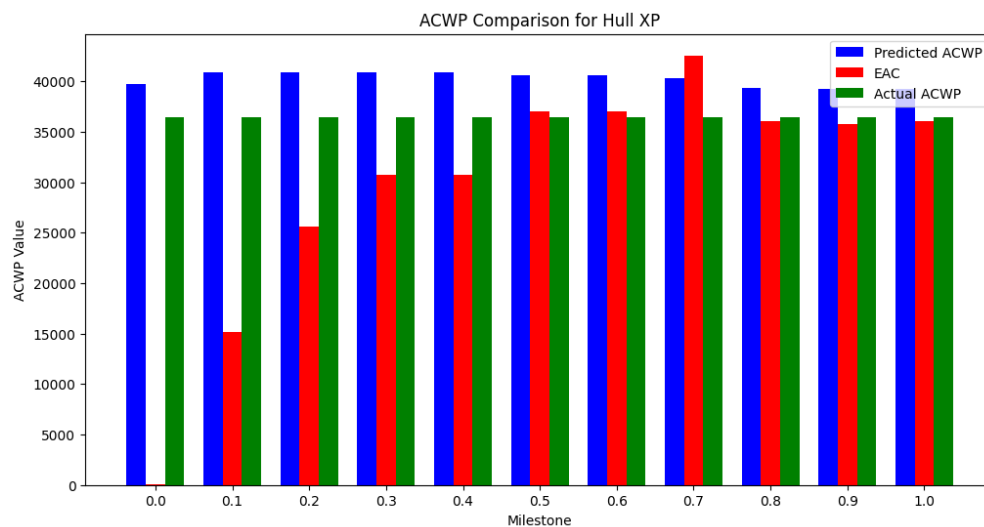


Figure 4.17: ACWP Prediction Comparison for Test Hull XP

There is significant variation in EAC among the test hulls, largely driven by the reliance of the EAC calculation on complete, accurate data. Comparatively, the regression model is better equipped to handle outliers or missing data and still make reasonable predictions. In Figure 4.17 the EAC predictions are more reasonable than those for the other test hulls, however they all follow a similar trend. From this graph it is clear that by the 80% milestone and later EAC is a very good prediction metric, even outperforming the regression model. To confirm this result, the difference between the two predictions and actual ACWP was averaged over all the test hulls and plotted on a logarithmic scale in Figure 4.18.

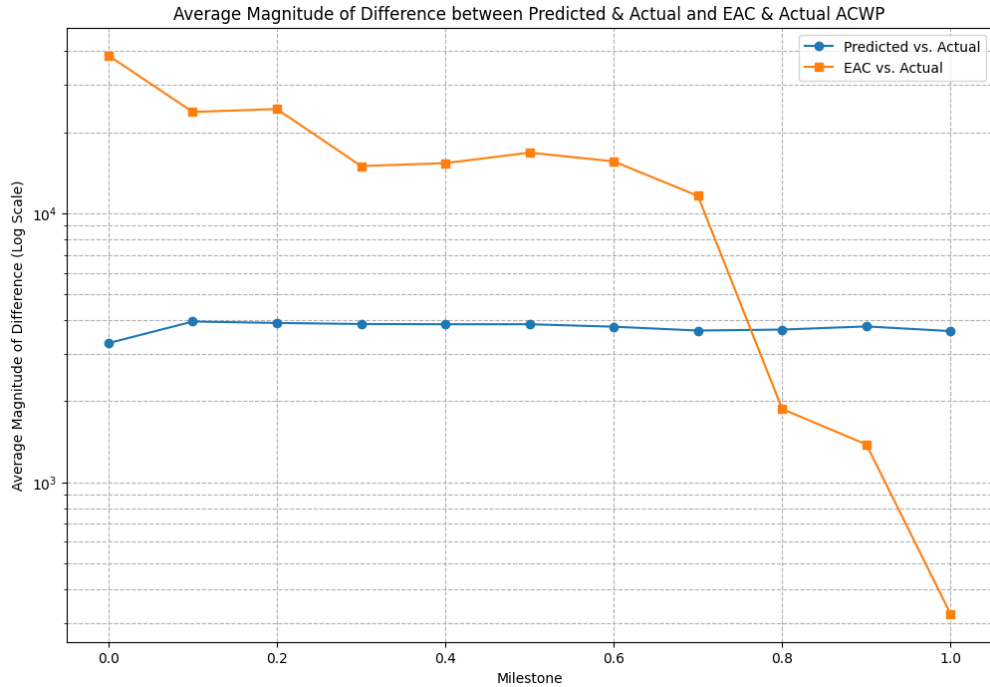


Figure 4.18: Average ACWP Prediction Difference on a Logarithmic Scale

Figure 4.18 confirms the fact that from the onset of the project until a milestone of ≈ 75 -80%, the polynomial regression model consistently outperforms EAC, as evidenced by the lower magnitude of difference between the model's predictions and actual ACWP values. This graph also highlights the fact that the regression model maintains a relatively stable magnitude of difference throughout the project, despite training on more project specific data at later milestones. In contrast, the EAC's higher and more fluctuating line underscores larger errors, especially pronounced at earlier project stages. Notably, the EAC method exhibits a sharp decline in the magnitude of difference after the initial milestone, which is consistent with the EAC formula which without enough data simplifies to just the cumulative ACWP.

The model's strength is particularly significant at early milestones, where its predictive capability allows for more accurate forecasting and thus better project planning and budgeting. The constancy in the performance of the polynomial regression model across different milestones, rather than an expected improvement in accuracy, may indicate that the model's initial training captured the primary cost-driving patterns of the construction projects quite

effectively. Additionally, this could imply that the complexities and variations introduced in the later stages of ship construction are not substantially influencing the cost predictions or are not being captured effectively due to potential model limitations. This consistency, while demonstrating robustness in early stages, highlights a potential area for refinement in the model to better incorporate and react to new data, allowing for a more nuanced and precise prediction as a project progresses and more information becomes available.

The constant model predictions could also be due to the postprocessing step where predictions are enforced to be nonstrictly increasing based on the nature of ACWP. However, this constraint might inadvertently dampen the model's responsiveness to genuine variations in the data at later milestones. Instead of allowing the model's predictions to naturally reflect the project's progression, the increasing-only enforcement could be masking subtleties and nuances in the cost development, leading to a plateau in predictive performance improvement. Such postprocessing might help avoid unrealistic dips in predicted costs, yet it also prevents the model from fine-tuning its accuracy with the influx of new data points, which ideally should provide a more detailed and accurate representation of the project's cost trajectory as it advances. This reveals a delicate balance between incorporating domain knowledge into model predictions and preserving the model's ability to adapt and learn from new data. While there was not time within this thesis to address these model limitations, it is discussed further in Section [5.4](#).

Chapter 5

Conclusion

The U.S. Navy faces growing demands to expand and modernize its fleet amidst increasing global maritime threats. However, the shipbuilding industry is facing significant challenges that hinder its capacity to deliver warships timely and within budget. In response to these critical issues, this thesis focused on leveraging EVM data to enhance predictive accuracy and operational efficiency in naval ship construction projects. Despite the widespread collection of EVM data, its potential to significantly improve project management practices remains underutilized. Through a detailed analysis of this data, the thesis aimed to develop an interpretable predictive model that forecasts costs more accurately and earlier in the shipbuilding process. This research serves as a starting point for more sophisticated tools in naval project management to enhance the efficiency and reliability of shipbuilding operations. This is done with the goal of ensuring that new ships are delivered faster and within budget, which is vital for maintaining naval supremacy.

5.1 Summary of Findings

This thesis began with the analysis of EVM data from 20 hulls of a specific shipbuilding project. The data was cleaned and processed into useful visual graphics highlighting which project subsets (MM, KE, LT/KE) contributed the most to cost and schedule overruns. Analysis was also done to measure the learning curve across hulls, illustrating if the shipbuilders were improving over time. For subsets with at least 5 complete hulls, approximately 55% of LT/KE combinations and nearly 69% of KE combinations had negative learning coefficients. This means the cumulative cost average is trending up across hulls in these cases and the shipbuilders are becoming less efficient despite any learning and experience.

This thesis then moved on to examine the effectiveness of predictive modeling of these shipbuilding projects. Through the analysis of various ML models with a focus on the EVM data, a predictive model was developed using polynomial and linear regression techniques. The model primarily aimed to forecast ACWP over time, providing stakeholders with a tool to anticipate potential cost overruns. The findings demonstrate that the model can successfully predict the progression of costs, which align well with the actual expenditures recorded during ship construction. Key features such as BCWS and BAC were identified as significant predictors of cost outcomes along with time. The model's ability to integrate

these features into its forecasts allows for a dynamic adjustment of predictions in response to project updates, enhancing the reliability of cost management practices. Furthermore, sensitivity analyses conducted as part of the research highlighted the model's responsiveness to changes in key input variables, affirming its utility in managing the inherent uncertainties of large-scale construction projects.

When compared to EAC, the EVM metric currently used to predict the cost at the end of construction, the predictive model outperformed EAC up to nearly 80% of completion (as measured by BCWS/BAC). This means that the model provides a more accurate prediction than currently exists for most of the project, highlighting its potential for immediate positive contributions to shipbuilding projects.

5.2 Lessons Learned

Throughout the course of this research, several valuable lessons were learned, not only in the application of EVM data but also in the broader context of predictive modeling within naval engineering. The study highlighted the importance of data preparation, the selection of appropriate modeling techniques, and the interpretation of data to support effective decision-making in shipbuilding projects.

Throughout this research, one of the foremost lessons learned was that **data cleaning is an ongoing process** rather than a one-time event at the beginning of a study. The initial phases of data handling revealed numerous inconsistencies within the EVM data sets, which required continuous attention and refinement throughout the study. This iterative cleaning process was crucial to maintain the integrity of the data as new insights and discrepancies were uncovered. It emphasized the need for vigilance and adaptability in working with real-world data, particularly in dynamic project environments like shipbuilding where data inputs are frequently updated and revised.

Another significant insight gained was the importance of **validation at every level of coding**. Ensuring that each segment of the code was correctly implemented and produced expected outcomes was vital for the reliability of the entire model. This involved regular debugging sessions and cross-validation with subsets of data to verify the consistency and accuracy of the results. Such meticulous validation practices helped in building a robust model that stakeholders could trust, and also facilitated easier troubleshooting and refinements in the modeling process.

The research also highlighted the **sheer amount of data required for some machine learning models** to function effectively. Complex models, such as deep learning networks, often demand large volumes of data to train on, which can be a challenge to procure in specialized fields like naval engineering. This requirement can limit the choice of feasible models and necessitates a careful balance between model complexity and the availability of adequate training data. It led to an appreciation for simpler, yet powerful, models that could deliver comparable predictive accuracy without the extensive data demands of more complex systems.

Lastly, the **time-consuming process of selecting the right machine learning model** was a critical learning point. Each model required thorough evaluation to assess its suitability for the data and the specific predictive tasks at hand. This involved not only

understanding the theoretical underpinnings of each model but also conducting empirical tests to compare their performance on the actual EVM data. The process was iterative and sometimes required going back to the drawing board to redefine the modeling approach based on preliminary results. This aspect of the research underscored the importance of patience and persistence in machine learning endeavors, where there often is no "optimal" model.

5.3 Limitations

While the research conducted provides valuable insights into the predictive modeling of shipbuilding costs using EVM data, it is important to acknowledge the limitations encountered during the study. These limitations affect the scope and applicability of the findings and are crucial for contextualizing the conclusions drawn from the research. Understanding these constraints helps to frame future research directions and refine the methodology employed. Below is a detailed discussion of the key limitations that were identified during the course of this thesis project.

- **Model Performance at Later Milestones:** The predictive accuracy of the model did not significantly improve when trained with more data at later milestones. This suggests a potential plateau in the learning curve of the model, where additional data inputs do not necessarily contribute to better predictions.
- **Limited Data Availability:** The study was constrained by the initial dataset provided by the shipbuilders. The limited amount of data, particularly complete hulls that span all stages of construction, restricted the model's training and may have impacted its generalizability to other shipbuilding projects or KEs.
- **Model Development for a Single KE:** Currently, the predictive model is developed and validated only for a specific key event within the shipbuilding process. This limits the scope of application and necessitates additional development for other KEs to create a more comprehensive predictive tool across the entire ship construction project.
- **Challenges with Model Assumptions:** The regression model relies on certain statistical assumptions (such as linearity, normality, and homoscedasticity) that were not fully met. This misalignment might have influenced the accuracy and reliability of the model's predictions. Issues like residual non-linearity, heteroscedasticity, or non-normal distribution of errors could potentially skew the results and impact the interpretability of the model.

5.4 Future Work

The findings of this thesis help pave the way for several promising directions for future research, which could enhance the robustness, applicability, and accuracy of the predictive model in shipbuilding projects. Addressing the limitations and building upon the groundwork laid by this research will allow for more sophisticated analyses and more comprehensive predictive capabilities. Below are key areas identified for future work:

- Find a way to apply this to all KEs to make total cost predictions. This could be accomplished by creating a model for each KE and aggregating the predictions. Another option is to extend the developed model to include all KEs within the shipbuilding process. Both options would provide a more holistic view of the project costs and improve resource allocation and planning at a macro level.
- Create a parallel predictive model focused on the schedule to complement the cost prediction model. This would help in providing a comprehensive project management tool that addresses both cost and time, critical factors in project success.
- Explore additional features that could enhance the model’s predictive power. Investigate the integration of other EVM metrics, categorical variables, etc. that could influence project costs.
- Conduct a more thorough investigation into alternative machine learning models that might offer better accuracy or different insights. With more data, time and resources, models such as ensemble methods, advanced regression techniques, or neural networks could be re-evaluated for their suitability.
- Experiment with different performance metrics, to include using MSE calculations at different project milestones or considering other statistical measures that might better capture the accuracy and reliability of predictions throughout the project lifecycle.
- Assess the impact of using a weighted average approach for hulls instead of one-hot encoding to potentially improve the model’s performance. The hull weights could be assigned based on recency or numerous other metrics.
- Address the assumptions that were not fully met in the current model to include non-linearity and heteroscedasticity. Implementing techniques like transformation of variables or adopting models that inherently manage these issues could refine predictions.
- Explore why additional data from later milestones does not significantly enhance model performance. This could involve a deeper dive into the data quality or incorporating dynamic project features such as updated expenditure rates or risk assessments. Continuous model refinement and testing, to include regularization techniques to help prevent overfitting, could also improve the model’s accuracy and reliability throughout the project lifecycle.

5.5 Final Thoughts

The journey of this research has been both challenging and rewarding, illuminating the complexities of predictive modeling within the shipbuilding industry. This thesis represents an important starting point in the exploration and development of more advanced predictive models within the shipbuilding industry. While the current model provides valuable insights, it also underscores the necessity for continued refinement and testing in predictive analytics.

The work completed serves as a foundational layer upon which future research can build, incorporating more complex algorithms and broader datasets.

As we look to the future, the integration of advanced analytics and machine learning into traditional industries like shipbuilding is poised to transform how projects are managed and executed. It is hoped that the foundations laid by this thesis will inspire further research and innovation, driving the shipbuilding industry towards a more data-driven and precision-oriented approach. In closing, the experience garnered through this research underscores the importance of persistence, creativity, and rigorous analysis in tackling complex engineering challenges. It is a vivid reminder that at the intersection of data and domain expertise lies the potential to effect substantial improvements and achieve operational excellence.

Appendix A

Code Listing

```
1 # Define a function that preps the train & test data
2 def prep_data_with_milestone(df, test_hull, milestone=0):
3
4     # Scale Ordinal Date to be consistent with BCWS/BAC
5     scaler = MinMaxScaler()
6     df_scaled = df.copy()
7     df_scaled['Ordinal Date'] = scaler.fit_transform(df[['Ordinal
8         Date']])
9
10    # Filter df for the test hull and preserve it in df_test
11    df_test = df_scaled[df_scaled['Hull'] == test_hull].copy()
12    df_test.reset_index(drop=True, inplace=True)
13
14    # Define BCWS/BAC column and fill any NaN values with 0
15    df_test['BCWS/BAC'] = df_test['BCWS'] / df_test['BAC']
16    df_test['BCWS/BAC'].fillna(0, inplace=True)
17
18    # Find the index where BCWS/BAC first reaches the milestone based
19    # on BCWS/BAC
20    milestone_index = df_test[df_test['BCWS/BAC'] >= milestone]
21    .index.min()
22
23    # Split the test hull data at the milestone
24    df_test_train = df_test.iloc[:milestone_index]
25    df_test_test = df_test.iloc[milestone_index:]
26
27    # Remove the test hull data from df to create a training dataset
28    df_train = df_scaled[df_scaled['Hull'] != test_hull].copy()
29
30    # Combine the first part of the test hull data with the data from
31    # other hulls for training
32    df_train = pd.concat([df_train, df_test_train])
33
34    # One-hot encode the hull identifiers & remove the baseline hull
```

```

31 df_train = pd.get_dummies(df_train, columns=['Hull'])
32 df_train.drop(baseline_hull_column, axis=1, inplace=True)
33
34 # Define BCWS/BAC column and fill any NaN values with 0
35 df_train['BCWS/BAC'] = df_train['BCWS'] / df_train['BAC']
36 df_train['BCWS/BAC'].fillna(0, inplace=True)
37
38 # Drop all other columns
39 columns_to_drop = ['Date', 'BCWP', 'BAC', 'BCWS', '
    Completion_Ratio']
40 df_train.drop(columns_to_drop, axis=1, inplace=True)
41
42 # Add missing one-hot columns for the test set that are present
    in the training set
43 df_test = pd.get_dummies(df_test_test, columns=['Hull'])
44 missing_cols = set(df_train.columns) - set(df_test.columns)
45 for col in missing_cols:
46     df_test[col] = 0 # Add missing columns with default value of
        0
47
48 # Ensure the order of columns in the test set matches that of the
    training set
49 df_test = df_test[df_train.columns]
50
51 return df_train, df_test, test_hull

```

```

1 # Define a function to train and test the model
2 def poly_regression_final(df_train, df_test, test_hull, degree=3,
    show_plot=False):
3     # Prepare features for training
4     features_train = df_train.drop(['ACWP', 'Ordinal Date'], axis=1)
5     ordinal_date_train = df_train[['Ordinal Date']]
6     y_train = df_train['ACWP'].values
7
8     # Prepare features for testing
9     features_test = df_test.drop(['ACWP', 'Ordinal Date'], axis=1)
10    ordinal_date_test = df_test[['Ordinal Date']]
11    y_test_true = df_test['ACWP'].values
12
13    # Apply polynomial features to 'Ordinal Date' only
14    poly = PolynomialFeatures(degree=degree)
15    ordinal_date_train_poly = poly.fit_transform(ordinal_date_train)
16    ordinal_date_test_poly = poly.transform(ordinal_date_test)
17
18    # Concatenate polynomial features of ordinal date with other
    features
19    X_train = np.concatenate([features_train.values,
        ordinal_date_train_poly[:, 1:]], axis=1) # Skip the first

```

```

    column to avoid multicollinearity
20 X_test = np.concatenate([features_test.values,
    ordinal_date_test_poly[:, 1:]], axis=1)
21
22 # Fit the linear regression model
23 model = LinearRegression()
24 model.fit(X_train, y_train)
25
26 # Print regression equation
27 feature_names = features_train.columns.tolist() + [f'x^{i}' for i
    in range(1, degree + 1)]
28 equation_terms = [f"{coef:.3f}*{name}" for coef, name in zip(
    model.coef_, feature_names)]
29 equation = " + ".join(equation_terms) + f" + {model.intercept_:
    .3f}"
30 print(f"Regression Equation for Test Hull {test_hull}: ACWP =",
    equation)
31
32 # Make predictions on test data
33 y_test_pred = model.predict(X_test)
34
35 # Apply postprocessing to predictions
36 y_test_pred = enforce_increasing(y_test_pred)
37 y_test_pred[y_test_pred < 0] = 0
38
39 # Evaluate and print test metrics
40 mse = mean_squared_error(y_test_true, y_test_pred)
41 r2 = r2_score(y_test_true, y_test_pred)
42 print(f"Mean Squared Error for Hull {test_hull} predictions: {mse
    :.2e}")
43 print(f"R-squared for Hull {test_hull} predictions: {r2:.2f}")
44
45 # Plot results (if desired)
46 if show_plot:
47     plt.figure(figsize=(10, 6))
48     plt.scatter(df_test['Ordinal Date'], y_test_true, label='
    Actual Values', color='blue')
49     plt.scatter(df_test['Ordinal Date'], y_test_pred, label='
    Predicted Values', color='red')
50     plt.title(f'Comparison of Actual and Predicted ACWP for Hull
    {test_hull}')
51     plt.xlabel('Ordinal Date')
52     plt.ylabel('ACWP')
53     plt.legend()
54     plt.show()
55
56 return mse, y_test_true[-1], y_test_pred[-1]

```

Appendix B

Additional Graphs

B.1 Graphs of ACWP Prediction Comparisons

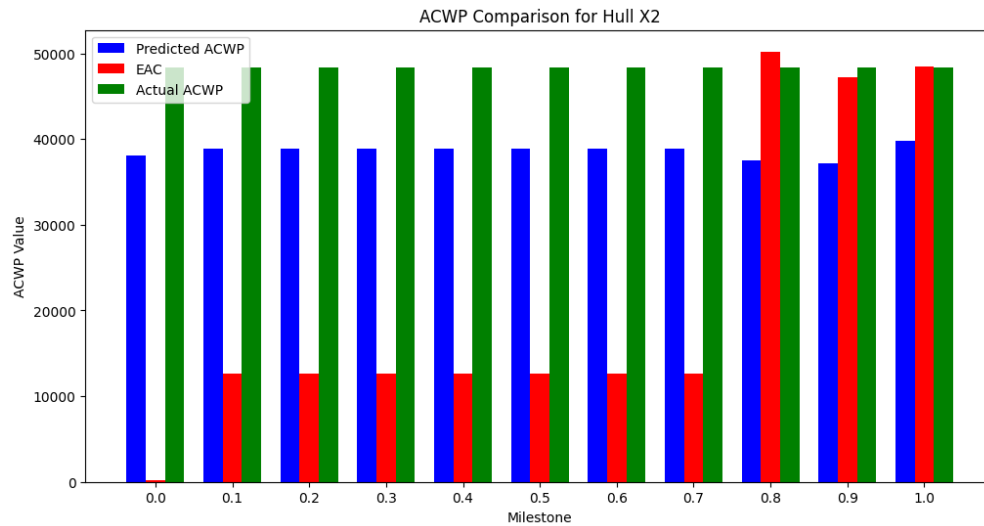


Figure B.1: ACWP Prediction Comparison for Test Hull X2

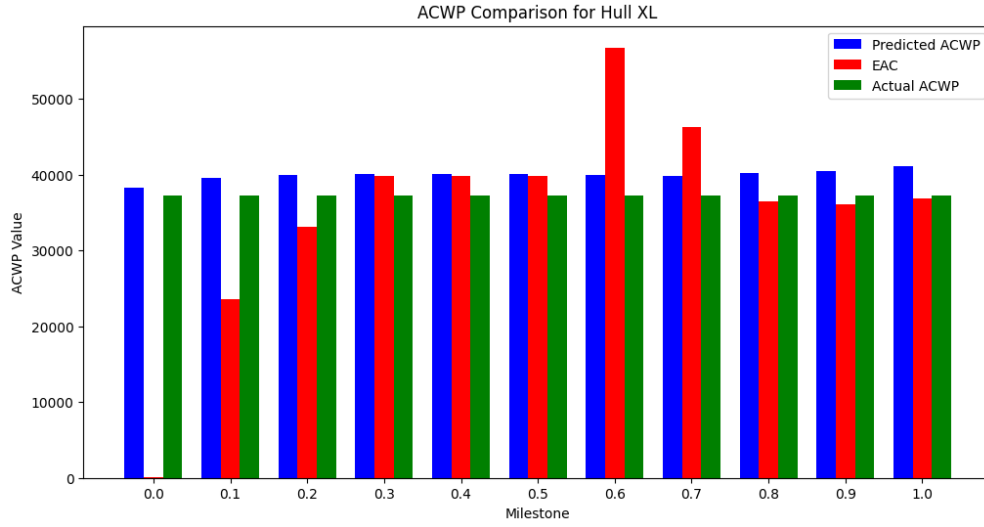


Figure B.2: ACWP Prediction Comparison for Test Hull XL

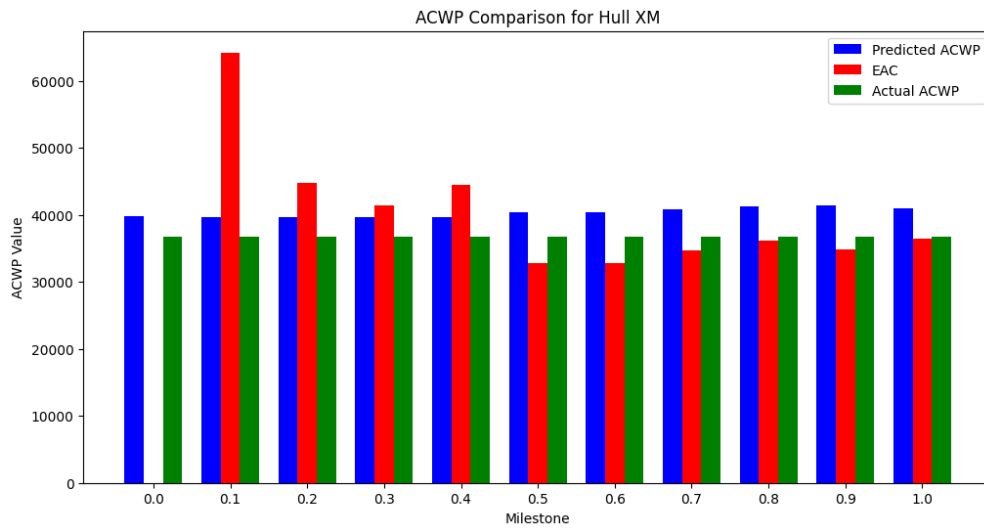


Figure B.3: ACWP Prediction Comparison for Test Hull XM

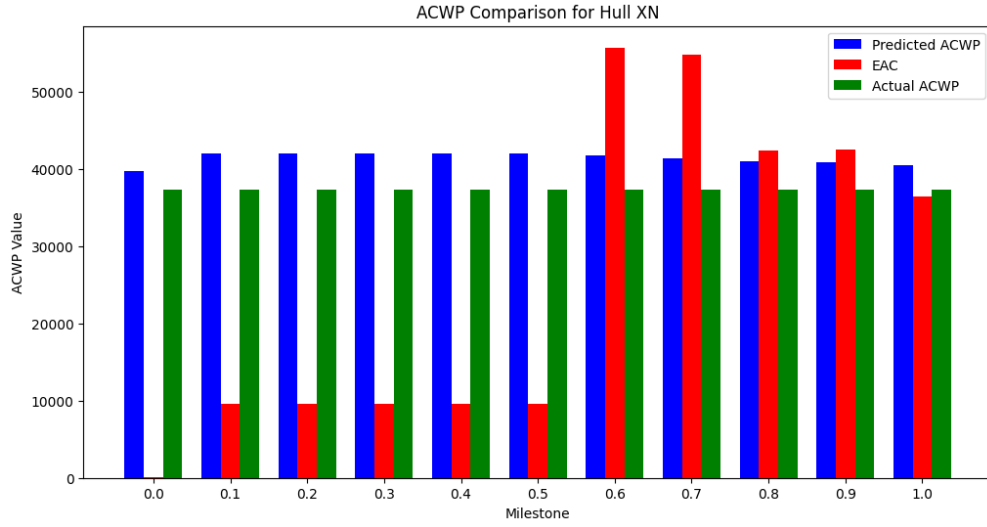


Figure B.4: ACWP Prediction Comparison for Test Hull XN

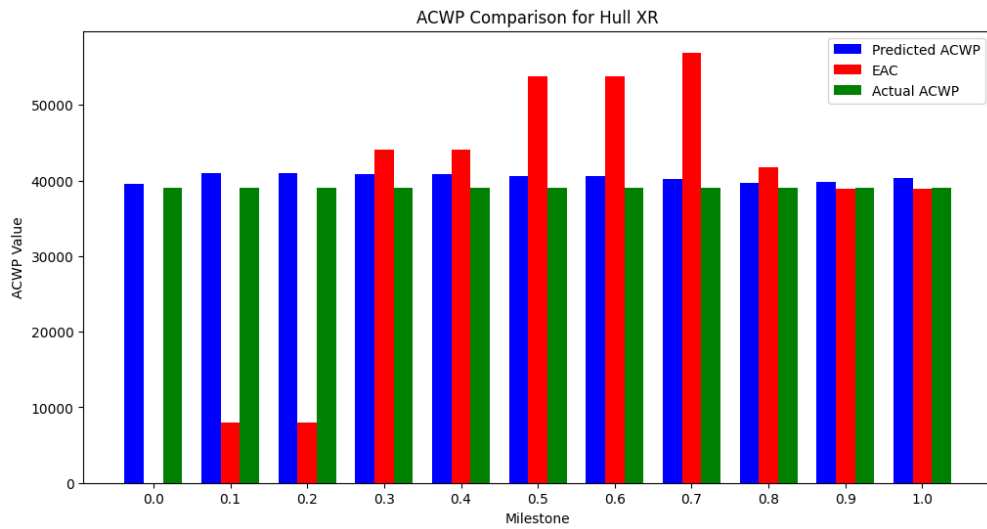


Figure B.5: ACWP Prediction Comparison for Test Hull XR

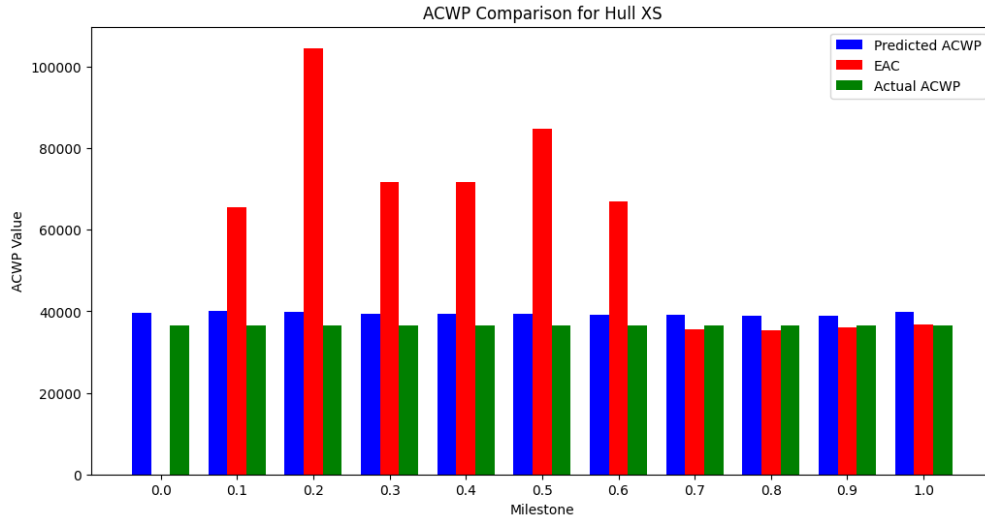


Figure B.6: ACWP Prediction Comparison for Test Hull XS

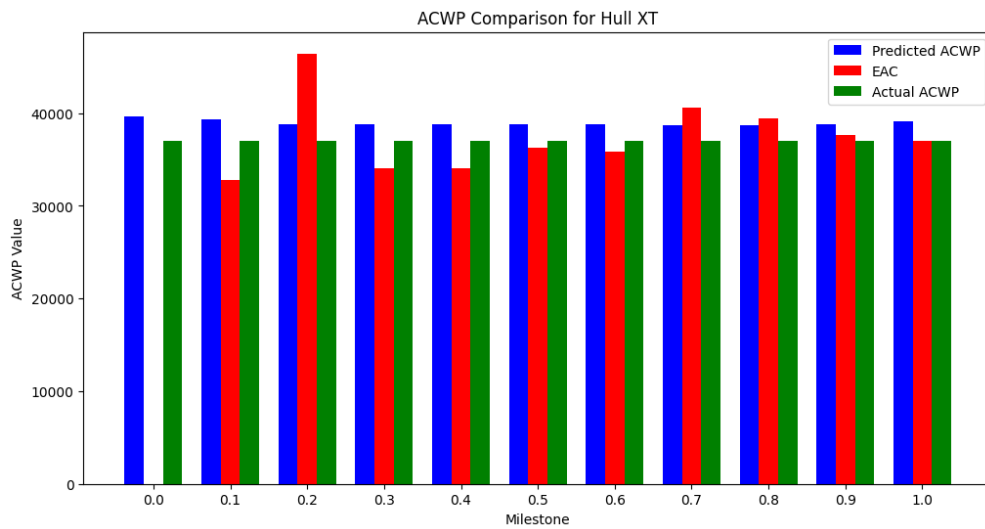


Figure B.7: ACWP Prediction Comparison for Test Hull XT

References

- [1] L. Thompson, *The U.S. Commercial Ship Industry Has Collapsed. Fallout For National Security Could Follow.* en, Section: Aerospace & Defense. [Online]. Available: <https://www.forbes.com/sites/lorenthompson/2024/02/08/the-us-commercial-ship-industry-has-collapsed-fallout-for-national-security-could-follow/> (visited on 04/20/2024).
- [2] *Chief of Naval Operations Navigation Plan 2022*, Jul. 2022. [Online]. Available: https://media.defense.gov/2022/Jul/26/2003042389/-1/-1/1/NAVIGATION%20PLAN%202022_SIGNED.PDF (visited on 06/14/2023).
- [3] M. Shelbourne, “CNO Gilday: Industrial Capacity Largest Barrier to Growing the Fleet,” en-US, *USNI News*, Aug. 2022, Section: News & Analysis. [Online]. Available: <https://news.usni.org/2022/08/25/cno-gilday-industrial-capacity-largest-barrier-to-growing-the-fleet> (visited on 06/14/2023).
- [4] M. Shelbourne and S. LaGrone, “CNO Gilday to Shipbuilders: ‘Pick Up the Pace’,” en-US, *USNI News*, Jan. 2023, Section: Budget Industry. [Online]. Available: <https://news.usni.org/2023/01/10/cno-gilday-to-shipbuilders-pick-up-the-pace> (visited on 06/14/2023).
- [5] U. S. G. A. Office, “High Levels of Knowledge at Key Points Differentiate Commercial Shipbuilding from Navy Shipbuilding,” U.S. GAO, Tech. Rep. GAO-09-322, May 2009. [Online]. Available: <https://www.gao.gov/assets/gao-09-322.pdf> (visited on 01/26/2024).
- [6] S. Lyson, “Military Shipbuilding Companies for the Federal Government,” en-US, *GovCon Wire*, Jul. 2023, Section: Articles. [Online]. Available: <https://www.govconwire.com/articles/military-shipbuilding-companies-for-the-federal-government/> (visited on 01/10/2024).
- [7] *Liberty ship*, en, Page Version ID: 1187974361, Dec. 2023. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Liberty_ship&oldid=1187974361 (visited on 01/25/2024).
- [8] P. A. C. P. Affairs, *Final Superlift Marks Structural Completion of CVN 79*, en-US, Jul. 2019. [Online]. Available: <https://www.navsea.navy.mil/Media/News/Article/1909959/final-superlift-marks-structural-completion-of-cvn-79/https%3A%2F%2Fwww.navsea.navy.mil%2FMedia%2FNews%2FArticle->

[View%2FArticle%2F1909959%2Ffinal-superlift-marks-structural-completion-of-cvn-79%2F](#) (visited on 01/25/2024).

- [9] *Earned Value Management (EVM) | www.dau.edu*. [Online]. Available: <https://www.dau.edu/acquipedia-article/earned-value-management-evm> (visited on 01/10/2024).
- [10] U. S. G. A. Office, *Weapon System Sustainment: Navy Ship Usage Has Decreased as Challenges and Costs Have Increased | U.S. GAO*, en, Jan. 2023. [Online]. Available: <https://www.gao.gov/products/gao-23-106440> (visited on 01/11/2024).
- [11] “Navy Force Structure and Shipbuilding Plans: Background and Issues for Congress,” Congressional Research Service, Tech. Rep. RL32665, Dec. 2023. [Online]. Available: <https://sgp.fas.org/crs/weapons/RL32665.pdf> (visited on 01/11/2024).
- [12] *Supervisor of Shipbuilding, Conversion and Repair*. [Online]. Available: <https://www.navsea.navy.mil/Home/SUPSHIP/> (visited on 01/26/2024).
- [13] W. McKinney, “Data Structures for Statistical Computing in Python,” in *Proceedings of the 9th Python in Science Conference*, 2010, pp. 56–61. [Online]. Available: <https://pandas.pydata.org/about/citing.html> (visited on 01/29/2024).
- [14] J. D. Hunter, “Matplotlib: A 2D Graphics Environment,” *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007, ISSN: 1521-9615. DOI: [10.1109/MCSE.2007.55](https://doi.org/10.1109/MCSE.2007.55). [Online]. Available: <http://ieeexplore.ieee.org/document/4160265/> (visited on 01/29/2024).
- [15] *Pandas.DataFrame — pandas 2.2.0 documentation*. [Online]. Available: <https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.html> (visited on 01/26/2024).
- [16] *Learning Curve*, en. DOI: [10.1093/oi/authority.20111110205317270](https://doi.org/10.1093/oi/authority.20111110205317270). [Online]. Available: <https://www.oxfordreference.com/display/10.1093/oi/authority.20111110205317270> (visited on 01/29/2024).
- [17] P. Virtanen, R. Gommers, T. E. Oliphant, *et al.*, “SciPy 1.0: Fundamental algorithms for scientific computing in Python,” en, *Nature Methods*, vol. 17, no. 3, pp. 261–272, Mar. 2020, ISSN: 1548-7091, 1548-7105. DOI: [10.1038/s41592-019-0686-2](https://doi.org/10.1038/s41592-019-0686-2). [Online]. Available: <https://www.nature.com/articles/s41592-019-0686-2> (visited on 01/29/2024).
- [18] *Types of Machine Learning*, en-US, Section: AI-ML-DS, Aug. 2023. [Online]. Available: <https://www.geeksforgeeks.org/types-of-machine-learning/> (visited on 04/22/2024).
- [19] *Prophet*. [Online]. Available: <http://facebook.github.io/prophet/> (visited on 04/10/2024).
- [20] E. Simhayev, K. Rasul, and N. Rogge, *Yes, Transformers are Effective for Time Series Forecasting (+ Autoformer)*, Jun. 2023. [Online]. Available: <https://huggingface.co/blog/autoformer> (visited on 01/10/2024).

- [21] *Long short-term memory*, en, Page Version ID: 1219734577, Apr. 2024. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Long_short-term_memory&oldid=1219734577 (visited on 04/22/2024).
- [22] *TensorFlow*, en. [Online]. Available: <https://www.tensorflow.org/> (visited on 04/11/2024).
- [23] *Autoregressive integrated moving average*, en, Page Version ID: 1193440274, Jan. 2024. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Autoregressive_integrated_moving_average&oldid=1193440274 (visited on 04/22/2024).
- [24] J. Perktold, S. Seabold, K. Sheppard, *et al.*, *Statsmodels/statsmodels: Release 0.14.1*, Dec. 2023. DOI: [10.5281/ZENODO.593847](https://doi.org/10.5281/ZENODO.593847). [Online]. Available: <https://zenodo.org/doi/10.5281/zenodo.593847> (visited on 04/11/2024).
- [25] F. Pedregosa, G. Varoquaux, A. Gramfort, *et al.*, “Scikit-learn: Machine Learning in Python,” *Journal of Machine Learning Research*, vol. 12, no. 85, pp. 2825–2830, 2011, ISSN: 1533-7928. [Online]. Available: <http://jmlr.org/papers/v12/pedregosa11a.html> (visited on 01/29/2024).
- [26] Zach, *A Quick Intro to Leave-One-Out Cross-Validation (LOOCV)*, en-US, Nov. 2020. [Online]. Available: <https://www.statology.org/leave-one-out-cross-validation/> (visited on 04/22/2024).
- [27] *Sensitivity analysis*, en, Page Version ID: 1220081365, Apr. 2024. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Sensitivity_analysis&oldid=1220081365 (visited on 04/23/2024).