2021-12-01

Predicting Zero Bin In The Semiconductor Manufacturing Industry: Machine Learning Algorithms

Yazmin Montoya
The University of Texas at El Paso

PREDICTING ZERO BIN IN THE SEMICONDUCTOR MANUFACTURING INDUSTRY: MACHINE LEARNING ALGORITHMS

YAZMIN MONTOYA

Master's Program in Systems Engineering

APPROVED:

Sreenath Chalil Madathil, Ph.D., Chair

Megan Vaughan Kendall, Ph.D.

Jose Espiritu Nolasco, Ph.D.

Stephen L. Crites, Jr., Ph.D.
Dean of the Graduate School

Dedication

This thesis is dedicated to me: great job on your commitment, ambition, passion for education, and willingness to challenge yourself, even during the Covid-19 Pandemic.

PREDICTING ZERO BIN IN THE SEMICONDUCTOR MANUFACTURING INDUSTRY: MACHINE LEARNING ALGORITHMS

by

YAZMIN MONTOYA, B.S. ENGINEERING LEADERSHIP, MBA

THESIS

Presented to the Faculty of the Graduate School of

The University of Texas at El Paso

in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE

Department of Industrial, Manufacturing and Systems Engineering

THE UNIVERSITY OF TEXAS AT EL PASO

December 2021

Abstract

The semiconductor industry has faced supply chain manufacturing shortages that ultimately led to a worldwide chip shortage during the COVID-19 pandemic. These chip manufacturers use sophisticated and advanced manufacturing machinery in their fabs to manufacture chips. As experienced during the pandemic, manufacturing unavailability is often due to the lack of critical manufacturing-related spare parts. This thesis evaluates the effectiveness of machine learning algorithms to identify significant factors contributing to manufacturing part outages (i.e., zero-bin) to keep manufacturing equipment running at total capacity within the organization. We propose clustering methods to segment the data and use logistic regression, logistic lasso regression, random forest, and kNN approaches to identify important factors for those parts that could go to zero-bin. Extant research applies classic inventory management strategies based on expenditure, criticality, or usage to manage their parts' inventory throughout the year. Instead, the proposed methods explore whether predefined, static inventory parameters can predict whether a spare part reaches zero bin. To demonstrate the viability of this approach, we present a case study using one year's worth of data from a leading chip manufacturing company. Based on the modeling approaches, a lasso-based logistic regression proved the best predictive model amongst the five clusters with lead-time, current quantity available, days on inventory (usage remained relevant), and the part's reorder point being the most significant parameters.

Table of Contents

Dedication ii

Abstract iv

List of Tables vi

List of Figures vii

Chapter 1: Introduction 1

Chapter 2: Methods 6

Data Cleanup and Validation 6

Clustering 7

K-Means 8

Logistic Regression 9

Logistic Lasso Regression 9

Chapter 3: Experimentation Setup 11

Identifying the Problem – Output 11

Data Cleanup – Output 12

Data Validation 14

K-Means 16

Chapter 4: Results 18

Chapter 5: Conclusions and Future Work 21

References 23

Appendix A 26

Vita 33

List of Tables

Table 1.1: Current Inventory Management Research 5

Table 2.2: K-Fold Analysis Example... 10

Table 3.1: Parameters.. 13

Table 3.2: Cluster Descriptions... 17

Table 4.1: Coefficients Table... 20

List of Figures

Figure 1.1: Spare Parts Supply Chain Overview. 1

Figure 3.1: Part Availability Fishbone Diagram. 11

Figure 3.2: Histogram with Outliers. 14

Figure 3.3: Box Plots with Outliers. 14

Figure 3.4: Histogram with Log Scale. 15

Figure 3.5: Box Plots without Outliers. 15

Figure 3.6: Silhouette Method Results. 15

Figure 3.7: Elbow Method Plot. 16

Figure 3.8: K-Means Clustering Results. 17

Figure 4.1: Confusion Matrices. 19

Figure 4.2: Confusion Matrices Comparison. 19

Chapter 1: Introduction

This thesis explores the importance of spare part inventory management in the semiconductor manufacturing sector. The machines, fabs, and manufacturing tools that manufacture chips use spare parts whose size ranges very small to very large. The cost of these spare parts varies from cents to thousands of dollars. The COVID-19 pandemic has adversely impacted every supply chain in every sector **(**Helper and Soltas**)**. These spare parts have a unique supply chain depending on the manufacturer, country of origin, repairability, and several other parameters that the supply chain department manages. Figure 1.1 represents a high-level overview of the touchpoints for each part of the supply chain. The flow chart was created by interviewing all direct and indirect stakeholders.

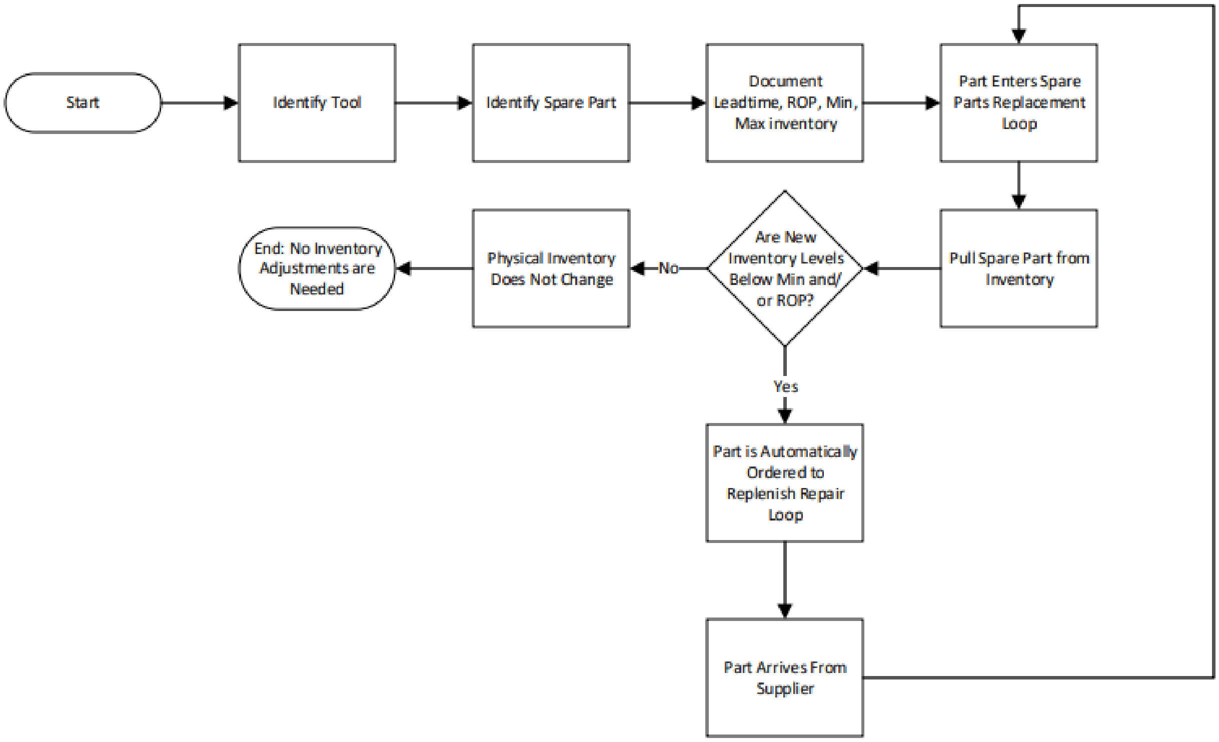

Start
Identify Tool
Identify Spare Part
Document Leadtime, ROP, Min, Max inventory
Part Enters Spare Parts Replacement Loop
Pull Spare Part from Inventory
Are New Inventory Levels Below Min and/ or ROP?
No
Physical Inventory Does Not Change
End: No Inventory Adjustments are Needed
Yes
Part is Automatically Ordered to Replenish Repair Loop
Part Arrives From Supplier

Figure 1.1: Spare Parts Supply Chain

Each spare part used in the manufacturing plant depends on the type of machinery. Hence, identifying a spare part begins with the need and usage of a manufacturing tool. Next, the buyer manually inputs the part's inventory information (maximum and minimum quantity and reorder point (ROP), based on historical and forecasted manufacturing plans). The warehouse stocks most of these spare parts as inventory items for future use. Based on the usage of these spare parts, further orders can be placed for replacement, known as the 'replacement loop.' This is called a loop because, as Figure 1.1 illustrates, the part enters a repair cycle or 'loop.' The plant uses these spare parts from the inventory as needed. The system will analyze whether the new inventory level reached the ROP. Nothing is done if the inventory level is above ROP, and the replenishment loop continues until another part is pulled from the inventory. More spare parts are ordered to fulfill the inventory levels somewhere between the maximum and the ROP if the current inventory is below the ROP. The new spare parts that arrived at the systems stay in the inventory stock until the manufacturing tools need these parts for replacement or repair. Major suppliers have dedicated teams for managing their inventory levels. In contrast, a vendor-managed inventory (VMI) system manages minor, site-specific, or niche-specific suppliers. In this case study, I review the data for a major supplier at one manufacturing site for a major semiconductor manufacturer.

The inventory management system documents every step in the process that creates a log of the movement of each part and records the reason for that part's movement. This extensive part history with thousands of parts and millions of data records is known as big data. Big data is characterized as large or complex datasets usually larger than an exabyte used for descriptive, predictive, or prescriptive analytics (Romeral et al., 177). I work with a portion of this big data in this research.

The world is in the middle of developing smart factories through an industrial revolution known as Industry 4.0 focused on three paradigms: "the smart product, smart machine, and augmented operator" (Weyer et al., 580). The *Smart Product* plays an active role within the system documenting data such as usage, run time, environment, and users that can improve its role in the overall system. The *Smart Machine* can self-organize by understanding its role in the system and improving its operations (Loskyll et al., 742). Finally, the *Augmented Operator* makes decisions based on data vs. needing years of experience to understand how their machine, cell, department work (Weyer et al., 580). The goal in the Industry 4.0 movement is to make strategic decisions instantly to optimize any system. This research explores how a spare part, as a smart product, can play a role in predicting its usage.

Current inventory management research includes 1) inventory management to provide a product to a customer and 2) inventory management used in manufacturing. This research focuses on the latter. Both research strategies heavily rely on historical consumption to build predictive models (K B et al., 867). The goal of these predictive models is to obtain the necessary inventory levels to prevent the outage of critical manufacturing tools' spare parts (prevent zero-bin). Preventing these outages is vital to maintaining productivity and improving profit. Markov decision-making is also relevant in the literature due to the cause-and-effect nature of the supply chain; however, the time-dependent Markovian processes are rare (Nasr and Elshar, 199).

Data-based decision-making is not new to the spare parts inventory management principles. Still, the type of characteristics used for decision-making is unique within the inventory management strategies. Current research identified inventory levels, lead-time, forecast based on usage, issues in the last 6-12 months, a risk measure, and minimum inventory quantities as significant factors using classical logistic regressions techniques (De Santis et al., 5). However, I will only use predefined inventory parameters captured by the company's data in this research due to their inventory management policies. This inventory management policy uses the part's lead time provided by the supplier and the days on inventory based on the period designated by the manufacturing company. Table 1.1 outlines the analyzed parameters. This research proved to be as accurate as current research that considers many more parameters that often need extensive research to create. The significance of this research revolves around the simplicity of the parameters and the model's ability to predict the zero-bin parts accurately.

Table 1.1: Current Inventory Management Research

Paper	**Methods**
Supply chain design and optimization: Challenges and opportunities (Garcia and You, 159)	- Multi-Scale Life-Cycle Optimization Frameworks - Multi-Objective Optimization
Decision Support Model for Inventory Management Using AHP Approach: A Case Study on a Malaysian Semiconductor Firm (Wong, 56)	- Analytic Hierarchy Process (AHP) - AHP in planning
The fourth industrial revolution (Industry 4.0): technologies disruption on operations and supply chain management (Koh, Lenny, et al., 822)	- Outcomes and Impacts of Industry 4.0 - Influence Policy Makers and Managers - Interdisciplinary Need
Inventory Management in the Era of Big Data (Bertsimas et al., 2009)	- Perfect Forecast Policy - Conditional Stochastic Optimization Problem
Predicting material backorders in inventory management using machine learning (de Santis et al., 2)	- Supervised Learning - Imbalanced Learning – SMOTE
Inventory management in supply chains: a reinforcement learning approach (Giannoccaro and Pontrandolfo, 154)	- Markov Decision Processes - Reinforcement Learning
Continuous inventory control with stochastic and non-stationary Markovian demand (Nasr and Elshar, 212)	- Markov Decision Processes - Monte-Carlo Simulation
A simulation-based multi-objective optimization framework: A case study on inventory management (Tsai and Chen,154)	- Ranking and Selection Procedures - Multi-Objective Optimization Simulation
Towards Industry 4.0 - Standardization as the crucial challenge for highly modular, multi-vendor production systems (Weyer et al.,582)	- Control Architectures - Manual Workstation - Smart Infrastructure - Plug and Produce - Production Line and Process
Context-Based Orchestration for Control of Resource-Efficient Manufacturing Processes (Loskyll et al.,740)	- Manufacturing Semantics Ontology - ADACOR-Ontology - AVILUS Ontology -
Inventory Management Using Machine Learning (K B et al.,867)	- XGBoost Regression Model - Decision Trees - Demand Forecasting

Chapter 2: Methods

I collected inventory data from a major semiconductor manufacturing company. The data is from the year 2020, during the COVID-19 pandemic (when anything that could go wrong in a supply chain did). The volatile year caused a greater risk of zero-bins and realized more zero-bins. The pandemic highlighted previously insignificant issues in every supply chain. These outages and the risk of outages in 2020 emphasized the need to prevent critical manufacturing spare parts from hitting zero-bin. The data better lends itself to predictive analysis because of these issues.

With hundreds of suppliers at the chosen site, a major supplier was selected for the case study. The supplier was chosen based on the large spends and volume of individual parts managed with that supplier. I selected a year's worth of data to capture a clear picture of a part's consumption history. The site was chosen because it is one of the company's largest manufacturing sites. Supply chains are incredibly complicated interconnecting systems made of subsystems. Narrowing the data to one supplier and one manufacturing site narrowed the case study's scope. The following methods were used to simplify the available data further while respecting the complexity behind each parameter.

Data Cleanup and Validation

The raw data was reviewed and rearranged per the chosen programming language. All static columns which were the same for each line of data were removed. After reviewing with the end-users, irrelevant data such as parts with no current consumption were removed from the dataset. Dummy variables were made to have a binomial distinction between the part's various categories**.** Preliminary data analyses were conducted to review the data's distributions. This consisted of histograms and boxplots with and without outliers.

CLUSTERING

Clustering is defined as creating homogeneous data groups in a dataset (Likas et al., 1). Categorizing data promotes learning and decision-making in machine learning. Machine learning is a 'component' in artificial intelligence where problems are solved by typing or finding patterns within the data. There are two forms of machine learning: supervised and unsupervised. In supervised learning, the data is given a set of rules or controls that will predetermine how the data will be classified. In unsupervised learning, no predefined labels are assigned to the data, and rather patterns and inherent groups are found within the data, and the data is assigned to the group that most accurately represents the relationships between the data (Alloghani et al., 4). Clustering is an unsupervised learning methodology. Clustering is broken down further into partitioning, hierarchical, density, or grid-based clustering.

Hierarchical methods help create a decision tree on where and how each cluster is related to the whole data (Aggarwal and Reddy Ch. 19). The scope of this thesis does not cover creating these decision trees; therefore, this method was not chosen. Density-based methods focus on data regions with more values centered around specific points and ignore the areas with less data (Aggarwal and Reddy Ch. 18). Since the data proved to have several outliers, this clustering method would not be the right fit for the dataset. Grid-based clustering is similar to density-based methods. The regions with more data points are segregated in data space by cells and then categorized based on their densities (Aggarwal and Reddy Ch. 6). Again, due to the varying outliers and long ranges, this method was not selected. Clustering using partitioning algorithms work in a loop. The data points are plotted repeatedly based on their distance to a local point. They ultimately get placed into clusters at the point where the sum of squared distances is minimized based on the identified local point (Aggarwal and Reddy, Ch. 17). This form of clustering made the most sense with the dataset, where each cluster would be created based on the different local points for each cluster. Therefore, I chose a partitioning method for this case study.

K-means and K-Medoids are two partitioning type clustering methods. K-Medoids uses a data point within the analyzed dataset to cluster the data points around. This aspect of the algorithm was concerning seeing the varying outliers within the dataset. K-means uses the Euclidean distance to find the distance between two points. The centralized point does not necessarily have to be a data point within the dataset but rather within space local to the specific cluster. K-means was chosen to analyze the dataset (Alva, Ch. 8).

K-Means

With a defined K, the K-Means algorithm works by randomly choosing "K" number of centroids. Then, in a loop, each data point's distance to each centroid is calculated to properly place the data point in the cluster with the nearest centroid. Next in the loop, the average distance of every data point in the cluster is calculated to verify that the chosen centroid for each cluster is valid. If the outcomes are different, new centroids are chosen again, averages recalculated and reverified until the centroids are accurate for each cluster, or the max number of iterations set up is met, and the loop is completed (Ahmad, 50).

Selecting K

The K in K-means represents the number of clusters. This information must be known prior to run the K-Means algorithm. Calculating the correct number of clusters can be a trial and error: running the algorithm with a different number of clusters, analyzing the descriptive statistics for each cluster, and validating the number of clusters that make sense based on the dataset. Two straightforward ways of verifying the number of clusters needed for analysis are the Elbow and Silhouette methods (Burkov, 114).

The Silhouette method is based on creating clusters that are equally separated from each other so that each cluster contains similar elements. This means "the silhouette score is based on the principle of maximum internal cohesion and maximum cluster separation (Bonaccorso Ch. 6).

The Elbow method is calculated using distortion. The larger the K value, the smaller the average distortion. This inverse relationship is because every data point will be closer to the centroid data point. However, the improvements in average distortion will decrease as K increases, and the K value at which distortion is at its highest decline is represented by the inflection point (Bonnin Ch. 3). I took both methods into consideration when selecting the appropriate K value.

LOGISTIC REGRESSION

A logistic regression starts with a linear regression but uses log-odds that are passed through a sigmoid function to output a probability between 0 and 1 and model the decision boundary for classification (Rai, The math bhind Logistic Regression). A first trial of the Logistic Regression proved that the data was not balanced. There were too few instances of zero-bin. I used the SMOTE method to balance the data, creating artificial data points based on the already existing data. Logistic regression was used on the SMOTE data. The data was modeled through a kNN regression and lasso-based regression to compare and contrast confusion matrices results.

LOGISTIC LASSO REGRESSION

A Lasso-based regression is very similar to logistic regression. Still, it has a hyperparameter (λ) and shrinkage that manipulate the coefficients to rid the model of the parameters/coefficients that are not significantly impacting the model's results. Another difference between the computation of the logistic regression and the lasso-based regression is in this method I used a K-fold analysis. This approach means the data is divided into K number of groups; For this research, an industry standard of K=5 was used. This method works iteratively to test and train the model using all of the available data. For example, when K=5, the data is divided into 5 test and train instances. In the first instance, all but one of the groups is used to train the model, and the final group is used to test. This is done until all groups have been used to either test or train the model (Brownlee, A Gentle Introduction to k-Fold Cross-Validation). This approach ensures all of the available data is used in both the testing and training aspect of the model. Table 2.2 demonstrates the K-fold approach.

Table 2.2: K-Fold Analysis Example

K	Test	Train
K = 1	x_1, x_2, x_3, x_4	x_5
K = 2	x_1, x_2, x_3, x_5	x_4
K = 3	x_1, x_2, x_4, x_5	x_3
K = 4	x_1, x_3, x_4, x_5	x_2
K = 5	x_2, x_3, x_4, x_5	x_1

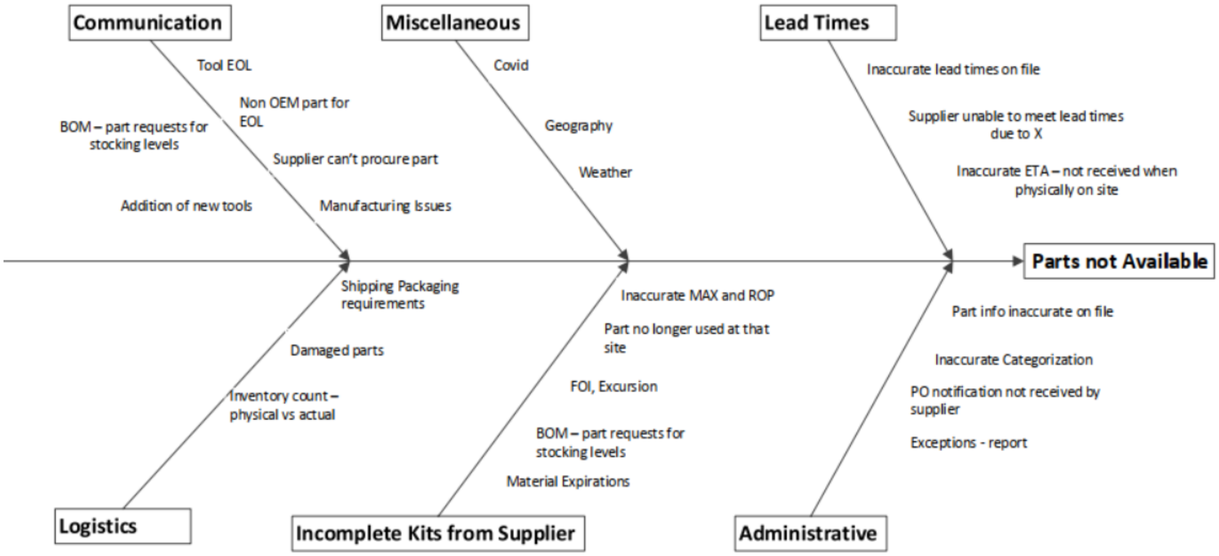
The K-Nearest Neighbor Regression (kNN) also uses k-fold validation. As with K-means, K is identified. Based on the prediction point, the K training observations closest to the prediction point and estimates the result using the average of the training data (Singh, K-Nearest Neighbors Algorithm: KNN Regression Python). The significant parameters were identified and then used to rerun the modeling formulas to analyze the final confusion matrices and develop a formula.

Chapter 3: Experimentation Setup

Following the methods outlined in Chapter 2, the experimentation setup will focus on the results of the methods. The decisions made throughout the experiment are validated through the preliminary analysis results. As more assumptions are made, and the behavior of the data is better understood, the following decision is validated. This section will shed light on the logic and reason for rejecting and selecting the experiment, choosing the suitable parameters, selecting the best clustering method, and how I navigated selecting the predictive models.

IDENTIFYING THE PROBLEM – OUTPUT

I met with industry experts in warehousing, procurement, commodity managers, supply chain engineers, department managers, tool engineers, and tool technicians. They all work on ensuring operations are running daily and are directly affected by zero-bins. With everyone working towards meeting their own goals, it is difficult for everyone to have a universal understanding of the entire process to procure and use a spare part. After identifying the various factors contributing to zero-bins, the research question had to be simplified further. By selecting one major supplier at one primary manufacturing site, the scope of the project was narrowed.



Communication
Tool EOL
Non OEM part for EOL
BOM – part requests for stocking levels
Supplier can't procure part
Addition of new tools
Manufacturing Issues
Miscellaneous
Covid
Geography
Weather
Lead Times
Inaccurate lead times on file
Supplier unable to meet lead times due to X
Inaccurate ETA – not received when physically on site
Parts not Available
Shipping Packaging requirements
Damaged parts
Inventory count – physical vs actual
Logistics
Inaccurate MAX and ROP
Part no longer used at that site
FOI, Excursion
BOM – part requests for stocking levels
Material Expirations
Incomplete Kits from Supplier
Part info inaccurate on file
Inaccurate Categorization
PO notification not received by supplier
Exceptions - report
Administrative

Figure 3.1 Part Availability Fishbone Diagram

Expert input was also taken into consideration when deciding the supplier and manufacturing location. I obtained raw manufacturing data that reflected a year's worth of spare part usage with the major supplier over the last calendar year. Figure 3.1 illustrates the cause-and-effect relationships that ultimately lead to zero-bin. I make the following significant assumptions: 1) All documented part information is accurate 2) Suppliers can meet demand 3) Parts without a reorder point are not stocked in the warehouse, and therefore will not be reviewed 4) Data without all measured parameters will not be considered. Within the supply chain, major domains such as logistics, part repositories, inaccurate data entry, tools reaching the end of life (EOL), and human error (such as shipping the wrong part) can all contribute to zero-bin inventory. I recognize there are miscellaneous events such as the Covid-19 pandemic or weather-related delays that can significantly impact the supply chain at any point.

DATA CLEANUP – OUTPUT

The data file was cleaned and reviewed in Python, using an integrated development environment (IDE) JupyterLab. The language and IDE were chosen for being a simple, popular language with several resources available for reference. The following are the packages I used:Numpy, Pandas, Statsmodels, Matplotlib, Sklearn, Scipy. A year's worth of data for one supplier accumulated to 1.38 million lines of data. The data is arranged on a week-by-week basis where each unique part number has a record of the quantity available for that week. The data were merged with a separate file that included the part's category and lead-time. Table 3.1 lists all the parameters identified in the combined data set.

Table 3.1: Parameters

Parameters	**Python Code Name**	**Description**
Site	Site	The location part is used in manufacturing
Supplier	Supplier	Company the part is purchased from
PN	PN	Unique Part Number
Cost	UnitCost_Org	Cost of purchasing part
Reorder point (ROP)	ROP_Org	Quantity at which to order more inventory
Quantity Available	QtyAvailable_Org	Current stock available
Days on Inventory (DOI)	DOI_Org	Quantity available to support manufacturing for X days
Workweek (W.W.)	WW_Org	Calendar workweek (Ex. WW1 = first workweek of the year)
Category	A_Org, B_Org, C_Org, D_Org, J_Org, Q_Org, U_Org,	Rating is given to part based on purchasing history
Leadtime	LeadTime_Org	Amount of time to get part from the supplier to the warehouse
Zero Bin (Predicted Value)	ZB_Org	Variable created to identify whether a part has zero bin for that W.W.
Lag	ZB_Lag	Inventory available in the previous week

The site and supplier columns were dropped as these are the same for every single line item in the file. A column called 'Z.B.' was created by using dummy variables to indicate whether the quantity available for that week was zero-bin (1) or not (0). Data entries where ROP = 0 indicated that the part is not stocked. These data points were dropped from the file. Data entries where DOI = 9999 are entries showing the part had not been consumed in a considerable amount of time were also dropped (they are not relevant to current inventory consumption patterns). At this point, 78,217 lines of data with all the identified parameters remained in the file. Dummy variables were also made for the eight unique part categories. A lag column was created by subtracting one from the work week and margining the data onto itself to see the quantity available the previous week and using the 'lag' (whether quantity was zero or not the previous week) in the regression. All unnecessary and repeated columns were removed. The suffixes '_Org' and '_ Lag1' were used to differentiate between the original data and the lag data. All rows with any NaN values were also dropped. A CSV file was then printed and saved to facilitate the coding process. The final data file consisted of 61,544 rows of data.

DATA VALIDATION

A preliminary data review was done after cleaning up the data file. Figure 3.2 represents the Histograms for ROP, Leadtime, and Unit Cost. _Org, UnitCost_Org, and LeadTime_Org. These parameters were selected to cluster with because once assigned to the unit, they do not change; QtyAvailable, DOI, W.W., all change. The categories were not considered for the data validation analysis due to the variability between the percentages. Figure 3.3 represents the boxplots, and Figure 3.4 depicts the boxplots for the same three parameters without any outliers. As presented, the plots do not communicate any valuable information. Trends and densities are challenging to understand in this format. Based on the histograms, the data does not seem to have any distribution, and all look like one uninformative cluster. However, as dispersed as the information

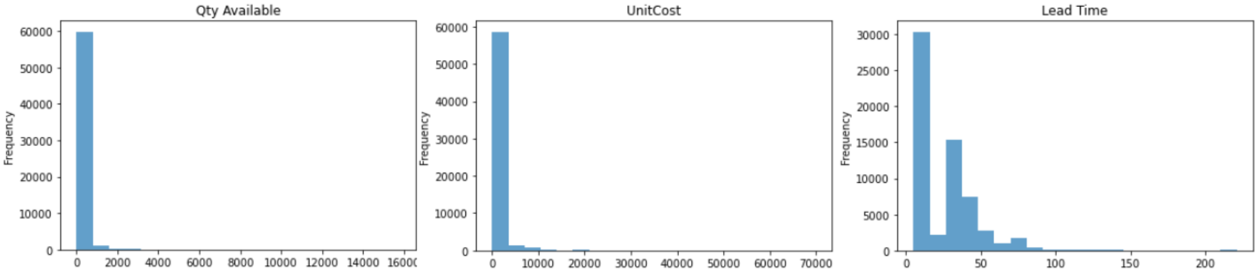


Figure 3.2: Histograms with Outliers

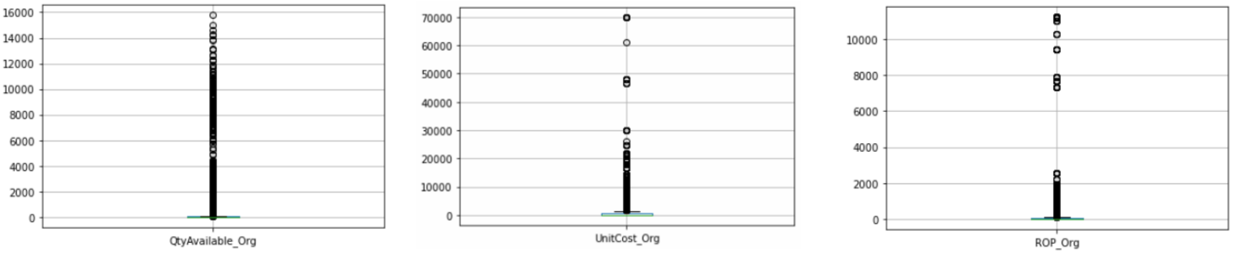


Figure3.3: Box Plots with Outliers

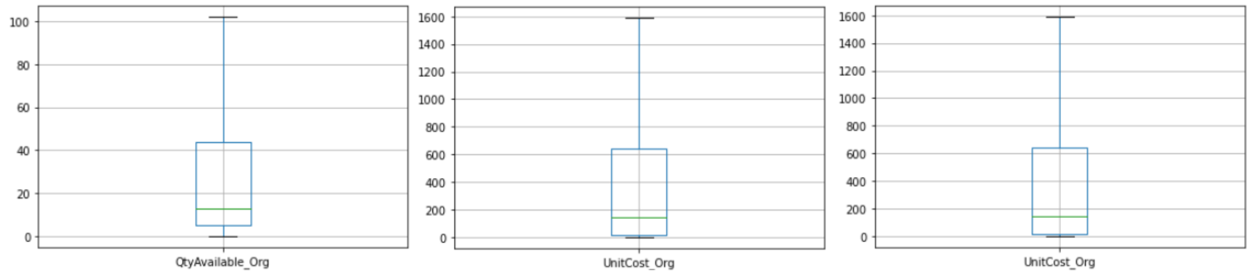


Figure 3.4: Boxplot without Outliers

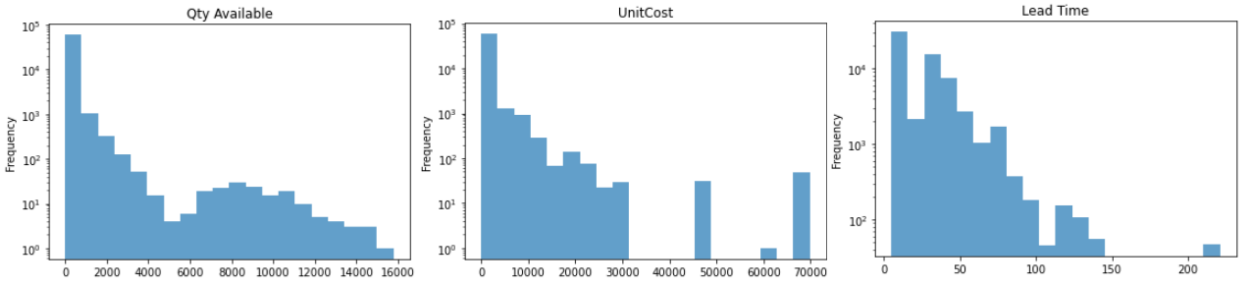
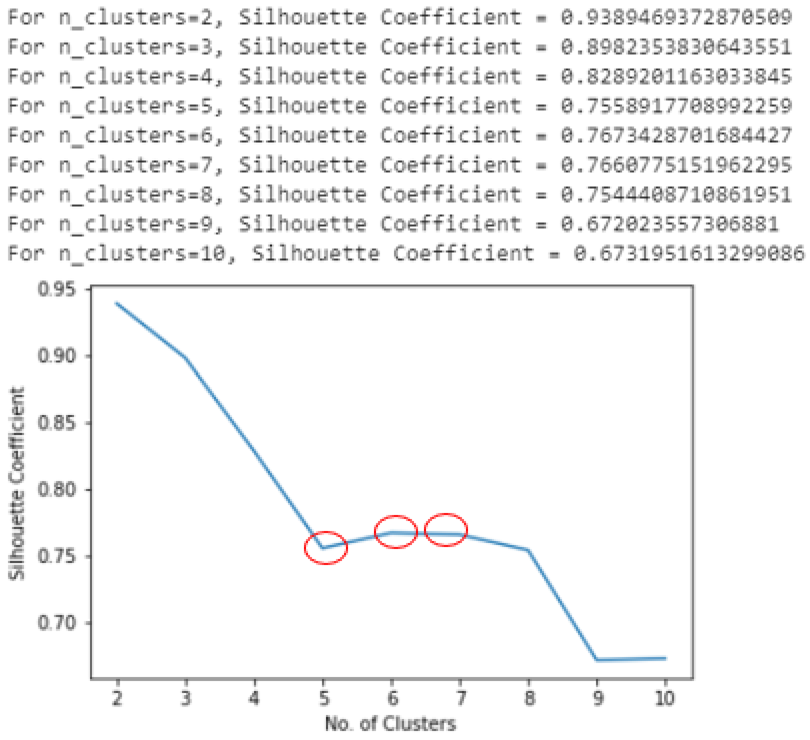


Figure 3.5: Histogram against Log Scale

is, all the data points are real-world and valid data. They should all be considered for a fair analysis.

Figure 3.5 illustrates the histogram of the three parameters when placed against a log scale. The



```
For n_clusters=2, Silhouette Coefficient = 0.9389469372870509
For n_clusters=3, Silhouette Coefficient = 0.8982353830643551
For n_clusters=4, Silhouette Coefficient = 0.8289201163033845
For n_clusters=5, Silhouette Coefficient = 0.7558917708992259
For n_clusters=6, Silhouette Coefficient = 0.7673428701684427
For n_clusters=7, Silhouette Coefficient = 0.7660775151962295
For n_clusters=8, Silhouette Coefficient = 0.7544408710861951
For n_clusters=9, Silhouette Coefficient = 0.672023557306881
For n_clusters=10, Silhouette Coefficient = 0.6731951613299086
```

Figure 3.6: Silhouette Method Results

data seems to have a more defined distribution. This led to the idea of transforming the data to facilitate the analysis. Considering that most of the data had vast ranges and that there was no way of knowing which way to classify the data based on the current parameters, clustering was proposed.

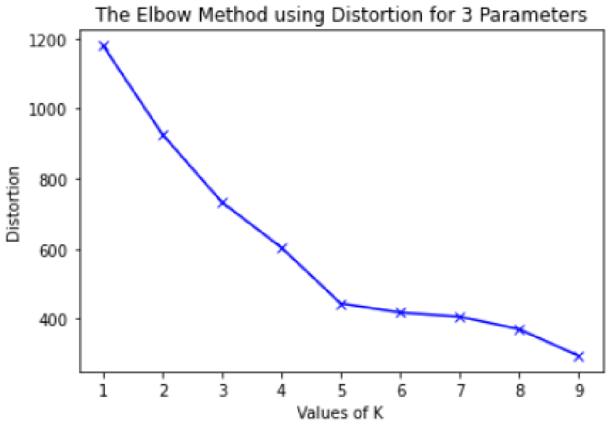


Figure 3.7: Elbow Method Plot

Figure 3.6 depicts the results of the Silhouette Method. Data peaks at n=6 by just under .012 at n=5 0and .001 at n=7. Based on the silhouette method, the appropriate number of clusters can be defined as 6. However, the elbow method Figure 3.7 was proposed because the silhouette scores are very close in value. The resulting graph has an inflection point (elbow) at k=5. Because the silhouette method's results were very close for n=5, n=6, and n=7, the results of both the silhouette and elbow method concluded with choosing 5 clusters

K-Means

The clusters were named based on the descriptive statistics and distinctive attributes per cluster. The descriptive statistics for each cluster can be found in Appendix A. Descriptive statistics (appendix A) can be used to fit new data into each cluster. Each cluster's data was made into its own data frame to be used for modeling. Figure 3.7 depicts the data distribution between each cluster. Leadtime and UnitCost heavily influence the behavior of the clusters. Table 3.2 illustrates the defining characteristics of each cluster. It is important to note that K-means presents different results every time it is run. Although I do only have 4 clusters, the number associated with each cluster changes every time. Therefore the best way to categorize each cluster is through descriptive statistics. Table 3.2 depicts the classification of each cluster. Appendix A provides the descriptive statistics for each cluster. Although the number assigned to the cluster can change, the number of instances per cluster tends to remain about the same. Table 3.2 provides the descriptions used to label the clusters. I chose average cost and average lead time because these two parameters highly influence the cluster's behaviors, as seen in the boxplots. The average cost is rounded to the nearest ten dollars, and lead time to the nearest day.

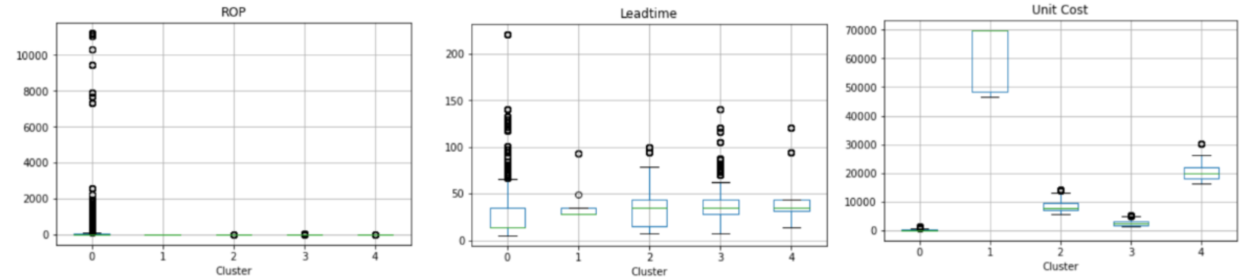


Figure 3.8: K-Means Clustering Results

Table 3.2: Cluster Descriptions

Cluster ID	Number of Instances	Avg Cost	Avg Lead Time
A	80	$61,250	35 Days
B	307	$20,860	48 Days
C	1,596	$8,700	37 Days
D	6,454	$2,530	38 Days
E	53,107	$240	25 Days

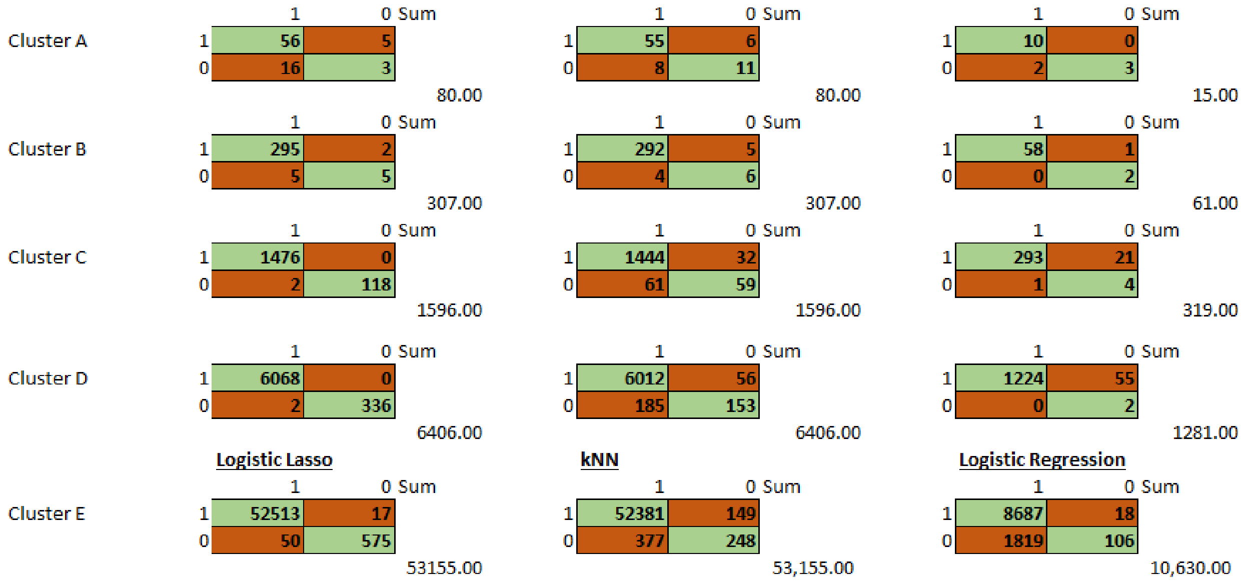
Chapter 4: Results

Each cluster was modeled three times using the following three modeling techniques: Logistic Regression, Lasso-based Logistic Regression, and kNN. This methodology resulted in 15 unique confusion matrices.

The accuracy, recall, precision, and F-measure of each confusion matrix are analyzed to select the best-performing model. kNN was not the best model for any of the clusters. The Lasso-based Logistic regression outperformed in three of the five clusters. The classic Logistic regression outperformed in the two clusters with the least number of data points. In trying to prevent zero-bins, the recall and precision of the model are very significant, making the F-measure a great way to compare amongst the three models. The significant parameters are Unit Cost, ROP, QtyAvailable, LeadTime, and QtyAvailable_Lag.

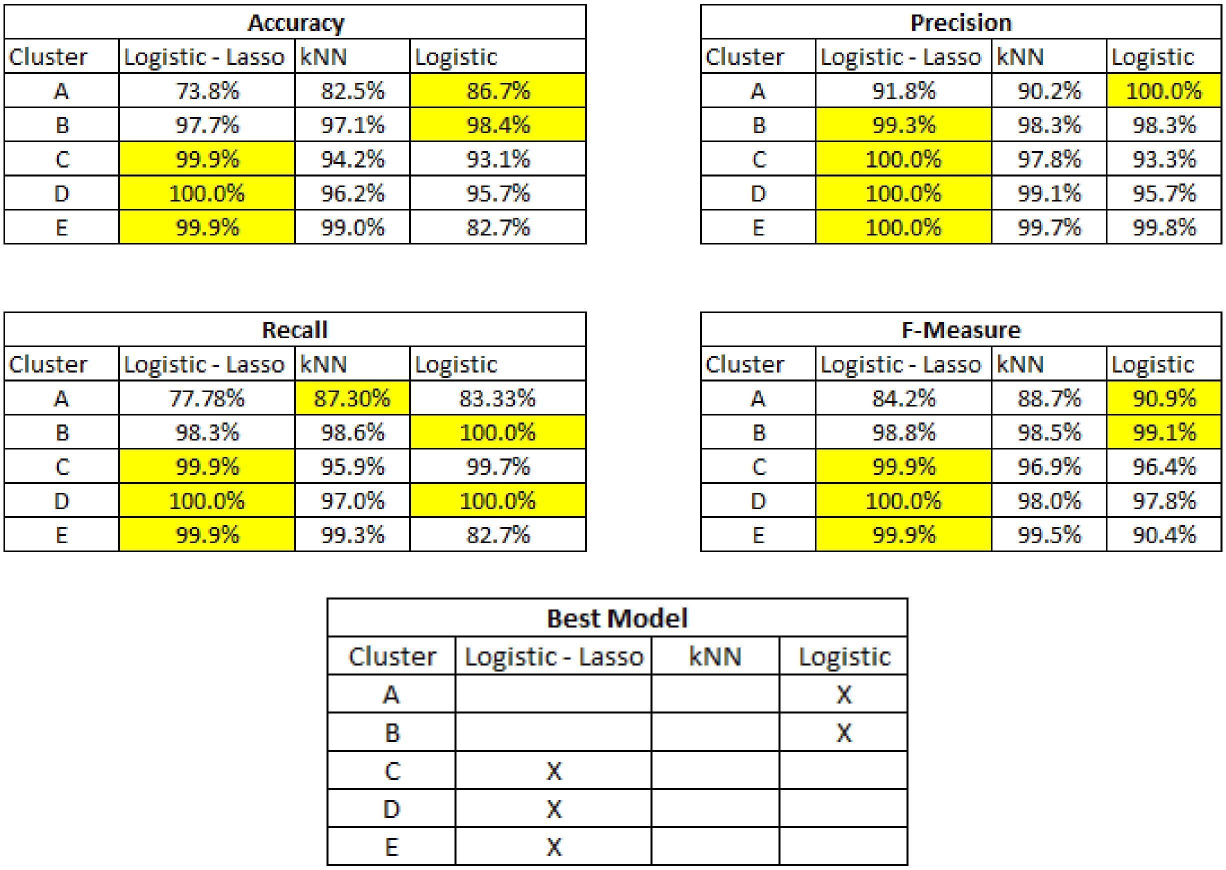
Confusion Matrices

I used all three predictive models (Logistic – Lasso, kNN, and Logistic Regression) on each cluster. This resulted in a total of 15 confusion matrices, as shown in Figure 4.1. The sums on the right of each confusion matrix are the number of data points used in the confusion matrix. The matrices for the Logistic-Lasso and kNN used all of the data points in the cluster due to the k-fold analysis method. I applied the SMOTE method to the Logistic Regression and therefore used fewer data points on this model. Table 3.2 in the Design Experiment presents the number of data points in each cluster. To compare the results of each model regardless of the number of instances used, I calculated the Accuracy, Recall, Precision, and F-Measures. The results of these calculations are represented in Figure 4.2. Accuracy represents the number of accurately made predictions over the total number of predictions. Recall calculates the number of true positives



		1	0	Sum		1	0	Sum		1	0	Sum
Cluster A	1	56	5		1	55	6		1	10	0	
	0	16	3		0	8	11		0	2	3	
				80.00				80.00				15.00
		1	0	Sum		1	0	Sum		1	0	Sum
Cluster B	1	295	2		1	292	5		1	58	1	
	0	5	5		0	4	6		0	0	2	
				307.00				307.00				61.00
		1	0	Sum		1	0	Sum		1	0	Sum
Cluster C	1	1476	0		1	1444	32		1	293	21	
	0	2	118		0	61	59		0	1	4	
				1596.00				1596.00				319.00
		1	0	Sum		1	0	Sum		1	0	Sum
Cluster D	1	6068	0		1	6012	56		1	1224	55	
	0	2	336		0	185	153		0	0	2	
				6406.00				6406.00				1281.00
		Logistic Lasso				kNN				Logistic Regression		
		1	0	Sum		1	0	Sum		1	0	Sum
Cluster E	1	52513	17		1	52381	149		1	8687	18	
	0	50	575		0	377	248		0	1819	106	
				53155.00				53,155.00				10,630.00

Figure 4.1 Confusion Matrices



Accuracy			
Cluster	Logistic - Lasso	kNN	Logistic
A	73.8%	82.5%	86.7%
B	97.7%	97.1%	98.4%
C	99.9%	94.2%	93.1%
D	100.0%	96.2%	95.7%
E	99.9%	99.0%	82.7%

Precision			
Cluster	Logistic - Lasso	kNN	Logistic
A	91.8%	90.2%	100.0%
B	99.3%	98.3%	98.3%
C	100.0%	97.8%	93.3%
D	100.0%	99.1%	95.7%
E	100.0%	99.7%	99.8%

Recall			
Cluster	Logistic - Lasso	kNN	Logistic
A	77.78%	87.30%	83.33%
B	98.3%	98.6%	100.0%
C	99.9%	95.9%	99.7%
D	100.0%	97.0%	100.0%
E	99.9%	99.3%	82.7%

F-Measure			
Cluster	Logistic - Lasso	kNN	Logistic
A	84.2%	88.7%	90.9%
B	98.8%	98.5%	99.1%
C	99.9%	96.9%	96.4%
D	100.0%	98.0%	97.8%
E	99.9%	99.5%	90.4%

Best Model			
Cluster	Logistic - Lasso	kNN	Logistic
A			X
B			X
C	X		
D	X		
E	X		

Figure 4.2: Confusion Matrices Comparison

divided by the number of correctly predicted positives and correctly predicted negatives. Precision communicates how precise the model was in detecting the number of true positives divided by the number of all predicted positives. The F-Measure is the harmonic mean between Recall and Precision. The goal of this model is to accurately predict the number of zero-bins, which means the F-measure is a great way to compare results amongst the models. Figure 4.1 illustrates the advantage that the Logistic-Lasso regression has over kNN. The classic Logistic regression with Logistic-Lasso outperforms in clusters C, D, and E. Classic-Logistic regression beat kNN and Logistic-Lasso in clusters A and B. The most significant parameters in the Logistic-Lasso are outlined in table 4.1. Below is the probability equation based on the significant parameters for the Logistic regression, keeping in mind that a shrinkage parameter of λ=.05 was used in modeling.

$$\begin{aligned} P(\text{ZeroBin} = 1|x_i) & \qquad (1) \\ = \Phi(\beta_0 + \beta_1 * UnitCost_{Org} + \beta_2 * ROP_{Org} + \beta_3 * QtyAvailable_{Org} \\ + \beta_4 * LeadTime_{Org} + \beta_5 * A_{Org} + \beta_6 * B_{Org} + \beta_7 * D_{org} + \beta_8 \\ * QtyAvailable_{Lag1} + \beta_9 * Doi_{Lag1} \end{aligned}$$

Table 4.1: Coefficients Table

Parameter	A	B	C	D	E
UnitCost_Org	0.000125943	3.19E-05	4.61E-05	-4.95E-05	-2.45E-06
ROP_Org	0	0.00058304	0	0.072292	0.035833
QtyAvailable_Org	0	-0.00320413	-3.97488	-6.08909	-6.52321
LeadTime_Org	0.0239108	0.016295	0.006828	0.00536062	-0.00142
A_Org=0	0	0.00020599	0	0	0
B_Org=0	0	-0.00036556	0	0	0
D_Org=0	0	0.00015725	0	0	0
QtyAvailable_Lag1	0	-0.00126622	0		0.003796
DOI_Lag1	-0.0294556	-0.0876153	-0.00471	-0.0014888	-0.00065

Chapter 5: Conclusions and Future Work

This research is inspired by the COVID-19 supply chain issues in the semiconductor industry. I conducted a case study with a major semiconductor manufacturing company and analyzed a year's worth of inventory data with one of its major suppliers. The data lent itself to a 5 cluster K-Means analysis. The clustered data was modeled using a classic Logistic regression, Lasso-based logistic regression, and a kNN regression. Lasso-based Logistic regression proved to be the overall best predictive model with the most significant parameters listed in table 431. Future work involves running the models on data that has not been tested or trained and evaluating how the models function. Once the models are further validated, the next step is optimizing the models and implementing them.

Research Bias

The Logistic-Lasso model proved to be a better fit for 3/5 clusters. This is because the Logistic-Lasso model implemented K-fold analysis. The model is more accurate in most of the clusters because all of the data was used in the testing and training of the model. The same cannot be said of the kNN model. More work needs to be done to obtain the best K value for the model. In this case, the industry standard of K=5 was used. K-fold analysis was not used on the classic logistic regression; instead, SMOTE, an oversampling technique, was used, and the Logistic-Lasso still managed to outperform the oversampled data.

Future Work

ROP is a very significant parameter. Therefore further research is needed to validate the current ROP calculation method. Cluster categories (A_Org, B_Org, D_Org) are only significant in one cluster and can be removed from the overall analysis. The QtyAvailable for the previous week was only significant for the smallest and largest cluster. Further analysis with more data is needed to understand whether this is significant.

In trying to keep the scope of the research project to only predefined characteristics of a unit, more parameters should be included in future model analysis. The locality is where the unit is located. If the unit is in the same state as the manufacturing site, then the risk of the amount of time to respond to a possible zero-bin is much less than were that unit in a different country. The next step with the current model results is to run new data in the model by fitting this data into the descriptive statics of each cluster and assigning a cluster. Implementing the model and understanding how untrained and untested data responds to the models.

References

Aggarwal, Charu C., and Chandan K. Reddy. *Data Clustering: Algorithms and Applications*. Chapman and Hall/CRC, 2018.

Alva, Jalil Villalobos. *Beginning Mathematica and Wolfram for Data Science: Applications in Data Analysis, Machine Learning, and Neural Networks*. Apress, 2021.

Ahmad, Imran. "The Logic of k-Means Clustering." *40 Algorithms Every Programmer Should Know: Hone Your Problem-Solving Skills by Learning Different Algorithms and Their Implementation in Python*, Packt, Birmingham, AL, 2020, pp. 50–200.

Alloghani M., Al-Jumeily D., Mustafina J., Hussain A., Aljaaf A.J. (2020) A Systematic Review on Supervised and Unsupervised Machine Learning Algorithms for Data Science. In: Berry M., Mohamed A., Yap B. (eds) Supervised and Unsupervised Learning for Data Science. Unsupervised and Semi-Supervised Learning. Springer, Cham. https://doi.org/10.1007/978-3-030-22475-2_1

Bertsimas, Dimitris, et al. "Inventory Management in the Era of Big Data." *Production and Operations Management*, vol. 25, no. 12, 2016, pp. 2006–2009., https://doi.org/10.1111/poms.2_12637.

Bonaccorso, Giuseppe. "Machine Learning Algorithms - Second Edition." *O'Reilly Online Learning*, Packt Publishing, 2018, https://learning.oreilly.com/library/view/machine-learning-algorithms/9781789347999/00886023-626e-404d-9dab-3b9a45ec1124.xhtml

Bonnin, Rodolfo. “Clustering.” *Machine Learning for Developers*, Packt Publishing, 2017.

Brownlee, Jason. "A Gentle Introduction to k-Fold Cross-Validation." *Machine Learning Mastery*, 2 Aug. 2020, https://machinelearningmastery.com/k-fold-cross-validation/.

Burkov, Andriy. "Unsupervised Learning ." *The Hundred-Page Machine Learning Book*, Andriy Burkov, Quebec City, Canada, 2019, pp. 107–121.

Dangeti, Pratap. "Statistics for Machine Learning." *O'Reilly Online Learning*, Packt Publishing, https://learning.oreilly.com/library/view/statistics-for-machine/9781788295758/c71ea970-0f3c-4973-8d3a-b09a7a6553c1.xhtml.

De Santis, Rodrigo Barbosa, et al. "Predicting Material Backorders in Inventory Management Using Machine Learning." *2017 IEEE Latin American Conference on Computational Intelligence (LA-CCI)*, 2017, pp. 1–6., https://doi.org/10.1109/la-cci.2017.8285684.

Garcia, Daniel J., and Fengqi You. "Supply Chain Design and Optimization: Challenges and Opportunities." *Computers & Chemical Engineering*, vol. 81, Mar. 2015, pp. 153–170., https://doi.org/10.1016/j.compchemeng.2015.03.015.

Giannoccaro, Ilaria, and Pierpaolo Pontrandolfo. "Inventory Management in Supply Chains: A Reinforcement Learning Approach." *International Journal of Production Economics*, vol. 78, no. 2, 2002, pp. 153–161., https://doi.org/10.1016/s0925-5273(00)00156-0.

Helper, Susan, and Evan Soltas. "Why the Pandemic Has Disrupted Supply Chains." *The White House*, The United States Government, 30 Nov. 2021, https://www.whitehouse.gov/cea/written-materials/2021/06/17/why-the-pandemic-has-disrupted-supply-chains/. James, Gareth, et al. *An Introduction to Statistical Learning: With Applications in R*. 2nd ed., Springer, 2021.

Jayaswal, Vaibhav. "Performance Metrics: Confusion Matrix, Precision, Recall, and F1 Score." *Medium*, Towards Data Science, 15 Sept. 2020, https://towardsdatascience.com/performance-metrics-confusion-matrix-precision-recall-and-f1-score-a8fe076a2262.

K B, Praveen, et al. "Inventory Management Using Machine Learning." *International Journal of Engineering Research And*, vol. V9, no. 06, 2020, pp. 866–869., https://doi.org/10.17577/ijertv9is060661.

Koh, L., Orzes, G. and Jia, F.(J). (2019), "The fourth industrial revolution (Industry 4.0): technologies disruption on operations and supply chain management", International Journal of Operations & Production Management, Vol. 39 No. 6/7/8, pp. 817-828. https://doi.org/10.1108/IJOPM-08-2019-788

Likas, Aristidis, et al. "The Global K-Means Clustering Algorithm." Pattern Recognition, vol. 36, no. 2, 2003, pp. 451–461., https://doi.org/10.1016/s0031-3203(02)00060-2.

Loskyll, Matthias, et al. "Context-Based Orchestration for Control of Resource-Efficient Manufacturing Processes." *Future Internet*, vol. 4, no. 3, 2012, pp. 737–761., https://doi.org/10.3390/fi4030737.

Nasr, Walid W., and Ibrahim J. Elshar. "Continuous Inventory Control with Stochastic and Non-Stationary Markovian Demand." *European Journal of Operational Research*, vol. 270, no. 1, 2018, pp. 198–217., https://doi.org/10.1016/j.ejor.2018.03.023.

Performance by Michael Keith, *Machine Learning with Regression in Python: With Ordinary Least Squares, Ridge, Decision Trees and Neural Networks*, Apress, a Springer Nature Company, Sept. 2020, https://learning.oreilly.com/videos/machine-learning-with/9781484265833/9781484265833-Keith_Overview/. Accessed 2021.

Rai, Khushwant. "The Math behind Logistic Regression." *Medium*, Analytics Vidhya, 14 June 2020, https://medium.com/analytics-vidhya/the-math-behind-logistic-regression-c2f04ca27bca.

Romeral José Luis, A., O. R. R., & Prieto, D. M. (2020). Chapter 10 / Big Data Analytics and Its Applications in Supply Chain Management. In *New trends in the use of Artificial Intelligence for the industry 4.0* (pp. 175–193). essay, IntechOpen.

Singh, Aishwarya. “K-Nearest Neighbors Algorithm: KNN Regression Python.” *Analytics Vidhya*, 25 May 2020, https://www.analyticsvidhya.com/blog/2018/08/k-nearest-neighbor-introduction-regression-python/.

Tibshirani, Robert. "Regression Shrinkage and Selection via the Lasso: A Retrospective." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 73, no. 3, 2011, pp. 273–282., https://doi.org/10.1111/j.1467-9868.2011.00771.x.

Tsai, Shing Chih, and Sin Ting Chen. "A Simulation-Based Multi-Objective Optimization Framework: A Case Study on Inventory Management." *Omega*, vol. 70, 2017, pp. 148–159., https://doi.org/10.1016/j.omega.2016.09.007.

Weyer, Stephan, et al. "Towards Industry 4.0 - Standardization as the Crucial Challenge for Highly Modular, Multi-Vendor Production System." *15th IFAC Symposium OnInformation Control Problems in manufacturing*, vol. 48, no. 3, 2015, pp. 1–6., https://doi.org/10.1016/j.ifacol.2015.06.143. Accessed 2021.

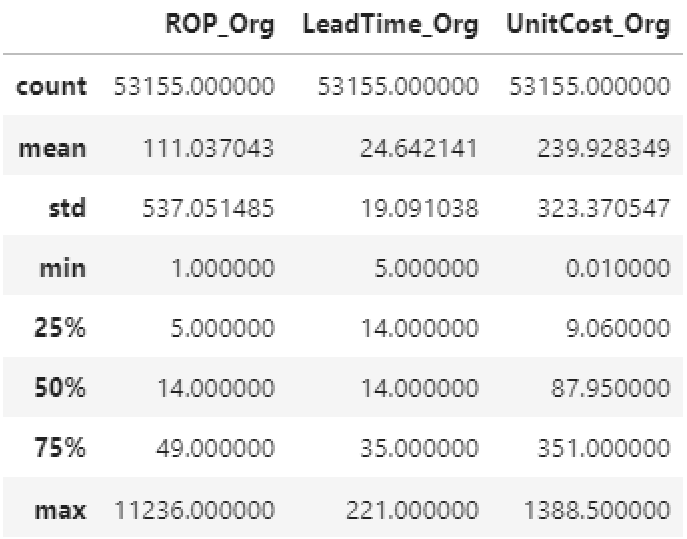
Wong, Wai Peng. "Decision Support Model for Inventory Management Using AHP Approach: A Case Study on a Malaysian Semiconductor Firm." *California Journal of Operations Management*, vol. 8, no. 2, Nov. 2010, pp. 55–71., https://doi.org/www.researchgate.net/publication/266606613.

Appendix A

Descriptive Statistics

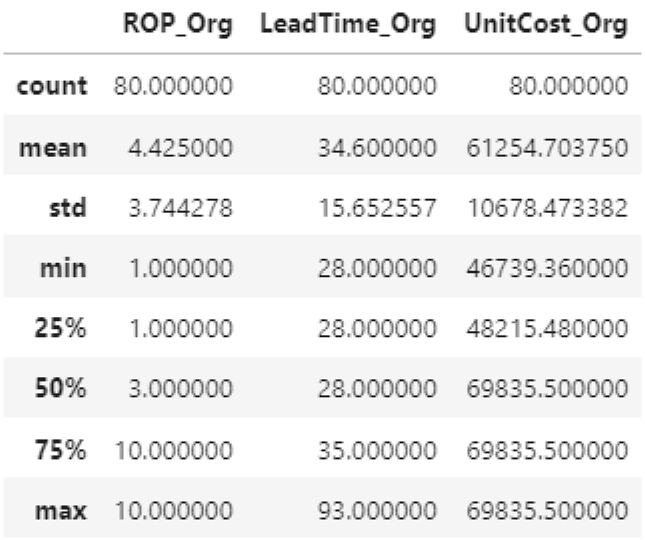
The following are the descriptive statistics for each cluster based on the three clustered parameters.

Cluster Zero Descriptive Statistics



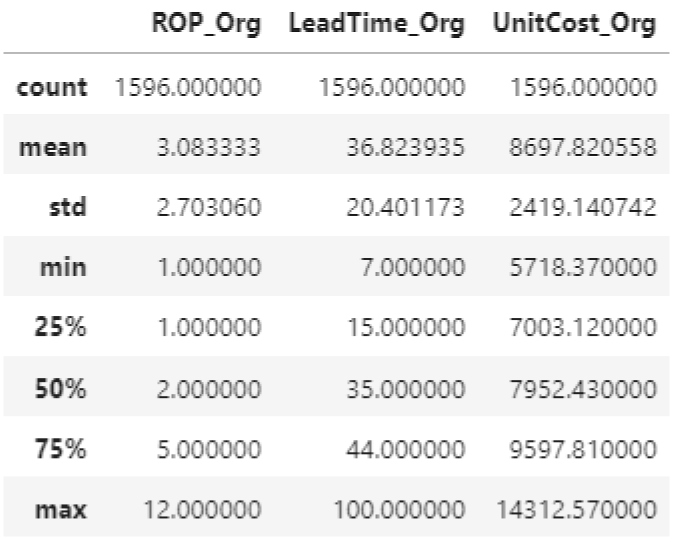
	ROP_Org	LeadTime_Org	UnitCost_Org
count	53155.000000	53155.000000	53155.000000
mean	111.037043	24.642141	239.928349
std	537.051485	19.091038	323.370547
min	1.000000	5.000000	0.010000
25%	5.000000	14.000000	9.060000
50%	14.000000	14.000000	87.950000
75%	49.000000	35.000000	351.000000
max	11236.000000	221.000000	1388.500000

Cluster One Descriptive Statistics



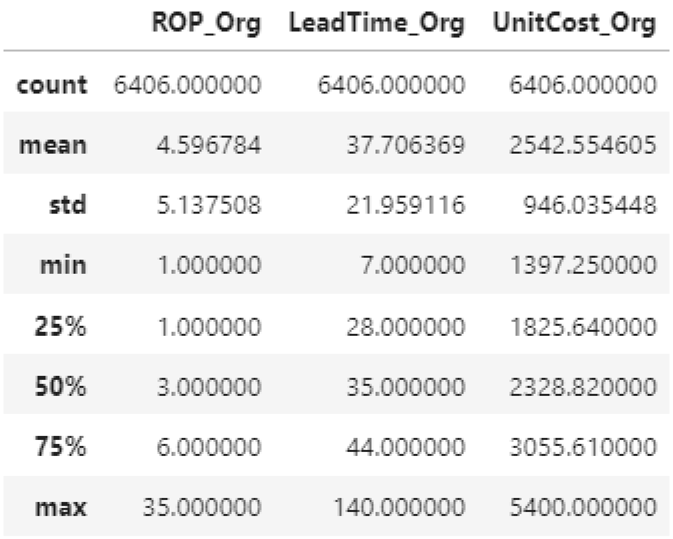
	ROP_Org	LeadTime_Org	UnitCost_Org
count	80.000000	80.000000	80.000000
mean	4.425000	34.600000	61254.703750
std	3.744278	15.652557	10678.473382
min	1.000000	28.000000	46739.360000
25%	1.000000	28.000000	48215.480000
50%	3.000000	28.000000	69835.500000
75%	10.000000	35.000000	69835.500000
max	10.000000	93.000000	69835.500000

Cluster Two Descriptive Statistics



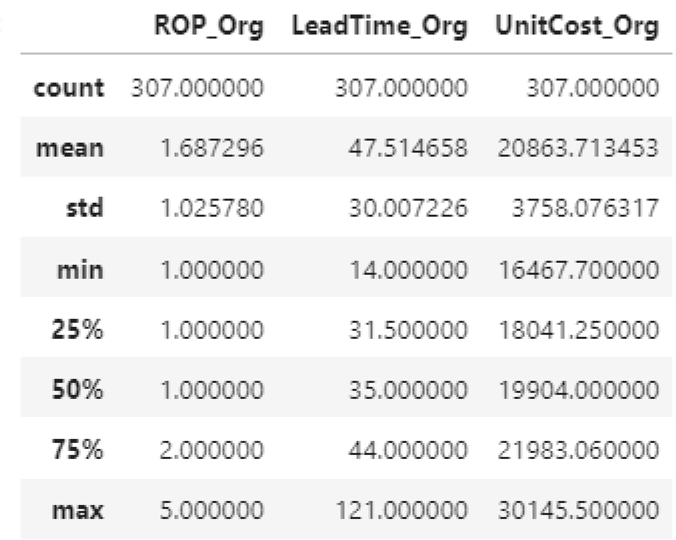
	ROP_Org	LeadTime_Org	UnitCost_Org
count	1596.000000	1596.000000	1596.000000
mean	3.083333	36.823935	8697.820558
std	2.703060	20.401173	2419.140742
min	1.000000	7.000000	5718.370000
25%	1.000000	15.000000	7003.120000
50%	2.000000	35.000000	7952.430000
75%	5.000000	44.000000	9597.810000
max	12.000000	100.000000	14312.570000

Cluster Three Descriptive Statistics



	ROP_Org	LeadTime_Org	UnitCost_Org
count	6406.000000	6406.000000	6406.000000
mean	4.596784	37.706369	2542.554605
std	5.137508	21.959116	946.035448
min	1.000000	7.000000	1397.250000
25%	1.000000	28.000000	1825.640000
50%	3.000000	35.000000	2328.820000
75%	6.000000	44.000000	3055.610000
max	35.000000	140.000000	5400.000000

Cluster Four Descriptive Statistics



	ROP_Org	LeadTime_Org	UnitCost_Org
count	307.000000	307.000000	307.000000
mean	1.687296	47.514658	20863.713453
std	1.025780	30.007226	3758.076317
min	1.000000	14.000000	16467.700000
25%	1.000000	31.500000	18041.250000
50%	1.000000	35.000000	19904.000000
75%	2.000000	44.000000	21983.060000
max	5.000000	121.000000	30145.500000

Vita

Yazmin Montoya earned her B.S. in Engineering Leadership 2017, MBA 2018, and Master of Systems Engineering 2021 from the University of Texas at El Paso. During her undergrad, she worked on biomedical engineering research in low-cost prosthetics. She also conducted research in engineering education and published two papers titled "Student-led curriculum development and instruction of introduction to engineering leadership course" and "Developing Leaders by Putting Students in the Curriculum Development Driver Seat." She interned in oil and gas, aerospace, and mining. After earning her MBA, she worked in several aspects of a supply chain in a copper mining company while pursuing her Master of Systems Engineering degree with a focus on modeling and simulation. She moved into program management after transitioning into the tech sector. She co-founded a nonprofit organization, 'Nontraditional College Success,' where she helps students land their dream jobs.