Multi-Modal Transit Time Prediction for E-Commerce Fulfillment Optimization and Carbon Emissions Reduction

by

Kathryn Angevine

B.S., Industrial Engineering
University of Massachusetts Amherst, 2018

Submitted to the MIT Sloan School of Management
Operations Research Center
in partial fulfillment of the requirements for the degrees of

Master of Business Administration

and

Master of Science in Operations Research

in conjunction with the Leaders for Global Operations program
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
June 2023

Author . . . .
MIT Sloan School of Management
Operations Research Center
May 12, 2023

Certified by . . . .
Stephen Graves, Thesis Supervisor
Abraham J. Siegel Professor of Management

Certified by . . . .
Y. Karen Zheng, Thesis Supervisor
George M. Bunker Professor and Associate Professor, Operations Management

Accepted by . . . .
Georgia Perakis
William F. Pounds Professor of Management Science
Co-Director, Operations Research Center

Accepted by . . . .
Maura Herson
Assistant Dean, MBA Program
MIT Sloan School of Management

Multi-Modal Transit Time Prediction for E-Commerce Fulfillment Optimization and Carbon Emissions Reduction

by

Kathryn Angevine

Submitted to the MIT Sloan School of Management
Operations Research Center
on May 12, 2023, in partial fulfillment of the
requirements for the degrees of
Master of Business Administration
and
Master of Science in Operations Research

Abstract

Consumers are purchasing an increasing amount of goods through digital channels as compared to brick and mortar and expect fast, reliable delivery. At the same time, society is facing the urgent challenge of reducing carbon emissions to limit global warming to levels considered safe by climate scientists. A global sportswear retailer is investing in improving the digital consumer experience while meeting its aggressive 2030 carbon reduction goals. This work studies how machine learning can be used to both improve the retailer's digital fulfillment operations and reduce their carbon emissions footprint. It focuses on enhancing the decision-making used to select a distribution center to fulfill a consumer's order from, and aims to do so by increasing the accuracy of a key input into that process. Specifically, the work targets accuracy improvement of transit time estimates, which quantify the number of days between a parcel's carrier induction and delivery.

Machine learning techniques are leveraged to develop a model for predicting transit times. Model development begins with data preparation, which is inclusive of sourcing, cleaning, sampling and feature engineering. It then continues with a series of experiments to provide insights into favorable model design elements. A final model is created under consideration of experimentation results. This model is associated with an accuracy of 67%, which is a improvement beyond the current state accuracy of 45%. A counterfactual analysis is conducted to assess the impact of improved transit time estimates on key fulfillment metrics. On a one month sample, the model enables improved fulfillment decisions; namely ones that are associated with a 4.5% decrease in lead time, a 3% reduction in CO2 emissions, and a 1.5% reduction in cost.

Thesis Supervisor: Stephen Graves
Title: Abraham J. Siegel Professor of Management

Thesis Supervisor: Y. Karen Zheng
Title: George M. Bunker Professor and Associate Professor, Operations Management

Acknowledgments

I would like to thank my host company and product team for generously welcoming me and investing in the success of this project. The mentorship I received and friendships I made enabled this experience to be an incredibly valuable and enjoyable one.

To my advisors, Karen Zheng and Stephen Graves, thank you for your technical guidance and feedback. I am grateful for the close relationship we established and the time you invested in me and this project. Your expertise allowed me to grow as a researcher.

To the LGO program, the Operations Research Center and the Sloan School of Management, thank you for the opportunity of a lifetime. The growth I have experienced in the past two years, professionally and personally, was beyond my expectations. I will look back on this experience and cite it as being instrumental to my future successes. It is an honor to be a part of these communities and to be affiliated with the world-class institution that is MIT.

To my LGO classmates, friends and family, thank you for always having my back, inspiring me to grow into improved versions of myself, and always being there as an outlet for fun. 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.

Note on Company Proprietary Information

To protect information that is proprietary to the host company, the data presented throughout this thesis has been modified and does not represent actual values. Data labels have been altered, converted or removed to protect competitive information, while still conveying the findings of this project.

Contents

List of Figures

THIS PAGE INTENTIONALLY LEFT BLANK

List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 1

Introduction

Consumers are purchasing an increasing amount of goods through digital channels as compared to brick and mortar. Their expectations of digital shopping experiences are shaped by Amazon and the like. As a result, a global sportswear retailer is focused on improving the digital consumer experience through investment in analytics technologies. To fulfill a digital order in a fast and cost-effective manner, knowledge of the time it takes to transport the goods from the distribution center to the consumer is required. These values, known as final mile transit time estimates, are currently provided by third-party transportation partners but suffer from inaccuracy and missing entries. This data integrity issue compromises the firm's ability to make optimal fulfillment decisions and thus risks a poor digital consumer experience. While consumer experience is of utmost important to the firm, so is its contribution to climate change. The firm has recognized the importance of limiting global warming to 1.5 degrees Celsius and has set corresponding 2030 carbon footprint reduction goals. [6] [12]

The goal of this work is to improve consumer experience and reduce carbon emissions through improved digital fulfillment decision-making. This will be done through the development of a machine learning model that generates enhanced final mile transit time predictions. This model will be assessed in terms of its accuracy, and in terms of its impact on key fulfillment performance indicators. The global footwear and apparel retailer that motivated this work will be referred to as Victory for the remainder of this thesis.

1.1 Industry Overview

1.1.1 Distribution Channels

The global sportswear industry consists of large multinational retailers such as Nike, Adidas, Under Armour and Puma. These firms sell apparel, footwear and accessories for use in athletic activity, as well as for normal everyday wear. Distribution is bifurcated across two channels: brick and mortar and e-commerce. Across the industry, there has been an increase in e-commerce demand due to convenience and public health factors. This trend is expected to continue [15] . As a result, retailers are competing on and investing in their e-commerce offerings.

1.1.2 Climate Commitments

The carbon footprint of the apparel industry is attracting attention as society increases its focus on climate change mitigation efforts. Fast-fashion retailers, or those that sell high-velocity, low cost items, sit at one end of the carbon footprint spectrum, while corporate sustainability pioneers like Victory, sit at the other. These corporate sustainability pioneers have made aggressive carbon reduction commitments to support the limitation of global warming to levels considered safe for human society. Commitments around waste reduction, water conservation, and hazardous chemical elimination are often considered by these pioneers as well. As consumers become more aware of the importance of climate change mitigation, sustainability is becoming an important aspect of a brand and an aspect which retailers compete upon [16].

1.2 Supply Chain Overview

Victory's supply chain not only is responsible for a large share of the firm's carbon emissions, but also is critically important in providing an exceptional experience for digital consumers. The supply chain begins with the manufacturing of goods, either by contract or in-house manufacturers. Once the goods are manufactured, they are shipped, often from overseas, to Victory-owned distribution centers. Once a consumer places their order on either the website or app, a critical set of decisions are made: which distribution center to fulfill the order from and whether to use ground, two day air or next day air shipping. Once the order has a distribution center and shipping method assigned to it, warehouse employees in the selected distribution center pick and pack the items. Sometime after the parcel is prepared, it is manifested to the third party transportation partner. At this point, the package becomes the carrier's responsibility. They are responsible for picking up the parcel from the distribution center, inducting or scanning it into their transportation network and delivering it to the consumer. Victory's effectiveness in operating this supply chain directly translates to how quickly the consumer receives their item and thus their satisfaction. The sequence of events described in this section is illustrated in Figure 1-1. Please note, some details of the supply chain that are not pertinent to this thesis are omitted from this description.

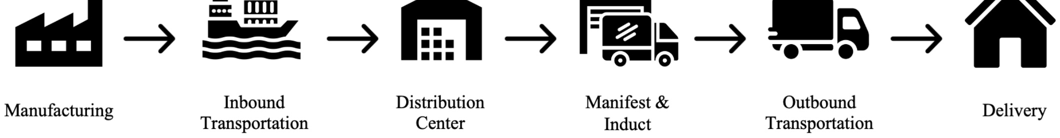

Figure 1-1: Simplified Supply Chain

1.3 Project Motivation

Victory evaluates the performance of their digital business using various speed, cost and emissions metrics. In general, Victory strives to meet or beat their delivery date targets while minimizing costs and emissions. Their ability to achieve success is directly tied to the set of fulfillment decisions mentioned above: which distribution center to fulfill the order from and which shipping speed to use. The closest distribution center to a consumer's delivery address may not have inventory. If it does, shipping from there may not be the cheapest option. Selecting the fastest shipping option will produce strong delivery speed metrics, but may be expensive and carbon intensive. In the face of such trade-offs and constraints, an optimization algorithm is used to make these fulfillment decisions. These decisions are considered optimal given a set of weights that score the importance of each performance metric.

An important data input to this optimization model is final mile transit time. Final mile transit time is the number of delivery eligible days between when the parcel was inducted into a third party carrier's fulfillment network, and when it was delivered to the customer. Days where carriers are not making deliveries are not included in transit time estimates, nor are they included in the calculation of actual transit times. Knowing the expected transit times for each distribution center with inventory for a given shipping speed allows the algorithm to effectively weigh cost, time and carbon trade-offs. The algorithm will then enable Victory to select the lowest cost and carbon option that still gets the parcel to the consumer on time.

Carrier-provided transit time estimates are currently used as an input to the optimization. Unfortunately, this data suffers from quality issues. The data, which provides the expected transit time in number of days, is incorrect about 56% of the time. Further, the data set does not provide estimates for all available distribution center and shipping speed combinations. These data problems prevent the fulfillment optimization algorithm from making the best choices, which leads to higher costs, higher emissions, or missing the delivery date targets. For example, when transit time is overestimated for a given distribution center, Victory may be forced to select a different, more costly distribution center, or may be forced to use carbon intensive air-based shipping methods to ensure the delivery date promise is kept. When transit time is underestimated, the risk of a late delivery increases. In each case, Victory and its consumers are worse off. This opportunity area is what motivates this thesis. If the quality of final mile transit time data could be improved, so could fulfillment decisions and the associated performance.

1.4 Problem Statement and Approach

This thesis aims to generate final mile transit time estimates for all available distribution center, shipping speed, destination zip code, and carrier combinations that are more accurate than the ones currently used. Specifically, the discrete number of delivery eligible days between when a carrier inducts the parcel into their network and when they deliver it to the consumer will be estimated. Predictive modeling techniques from machine learning will be leveraged to generate such estimates. Experimentation is conducted with generating estimates at two distinct time points.

1.4.1 Post-Purchase Prediction

For use in fulfillment decision-making, estimates must be generated immediately after a consumer places their order. This is when a distribution center and shipping speed are selected by the fulfillment optimization model. Models for generating estimates at this point in time may not use any information from after the order placement timestamp. For example, such models cannot include the time of the day distribution center employees finished preparing the parcel for pickup because that event has not occurred yet.

This type of model exists very early in an order's life cycle and thus is limited in the information it can use to make a prediction. Making a transit time prediction later in the order life cycle would allow for additional data to be used. Although this would likely enable a stronger prediction, the corresponding estimates cannot directly be used for fulfillment decision-making.

1.4.2 Post-Induction Prediction

Models that make the transit time prediction right after the parcel has been inducted or scanned into a carrier's network are also explored in this thesis. These models are included for two reasons. First, they support development of high-performance post-purchase models. Later stage prediction models have a broader view of fulfillment and can capture relationships that enable enhanced awareness of the factors that influence transit times. For example, a later stage model would have access to data regarding a later stage fulfillment event, and therefore would be able to demonstrate that this feature is important in predicting transit time. Since late stage models aid in establishing a strong understanding of the factors that influence transit time, they are used in this thesis during initial exploration efforts.

Post-induction models are also capable of directly adding value to the Victory organization. Having accurate transit time estimates later in the order fulfillment journey allows Victory to manage delivery date promises with their consumers. For example, if a transit time was expected to be shorter than the original estimate, an updated delivery date could be provided to the consumer.

1.5 Thesis Overview

The remainder of this thesis consists of five chapters:

Chapter 2 provides the necessary background on predictive transit time models. It begins with a review of related academic work, then transitions to a presentation of relevant data for this type of prediction problem in the context of Victory. The chapter concludes by detailing the performance associated with the current transit time estimates used in fulfillment decision-making.

Chapter 3 shares the model development process used in this thesis. It begins with sharing the steps associated with analytical data set creation, namely data sourcing, cleaning, sampling and feature engineering. It provides insights from an exploratory data analysis, before transitioning to describing the model experimentation process. It shares the performance metrics used to evaluate the predictive models, then details the extensive number of experiments conducted to determine the best model design elements.

Chapter 4 makes an assertion of which model is best and details the performance of this model. A comparison to the carrier provided estimates is made and the importance scores of the model features are shared. Finally, Chapter 4 investigates the impact of improving transit time estimates on key fulfillment performance metrics.

Chapter 5 shares a set of recommendations on how to unlock the business value associated with this work in production. It suggests next steps related to the creation and deployment of a live model, and discusses the importance of prioritizing this work over additional model enhancement efforts.

Chapter 6 provides a conclusion to this thesis. It begins with a summary of the thesis, shares important lessons learned throughout the work, and concludes with a set of ideas for future work on related topics. The suggestions for future work span three distinct areas related to geographic expansion, data preparation and model experimentation.

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 2

Background on Predicting Transit Time

2.1 Related Work

In this section, work related to this thesis is presented. Specifically, travel time prediction problems are shared. This thesis considers the final mile transit time prediction problem a subset of the more general travel time prediction problem. A wide range of techniques to predict travel time have been used in the literature. For a focused comparison to the methods used in this thesis, studies with machine learning based methodologies will be review in this section.

The problem of predicting travel time has been studied in various forms in the literature [1]. Studies can be classified into various groups based on several key characteristics. First, studies can be grouped based on the duration of time they are aiming to predict. Some are focused on predicting short range travel times, on the order of hours, while others are focused on longer range predictions in days. Longer range predictions are considered to be more challenging [9].

The travel time prediction literature is often segmented based on whether the problem is route-based or origin destination-based. Route-based prediction problems incorporate information on the path taken to get from the origin to the destination. For example, in ground transit prediction, information on the roads taken will be used. Some route-based prediction problems make predictions on independent segments of the path and then aggregate to the path level. Others make predictions directly at the path level. Travel time prediction methods that do not utilize route information are known as origin destination-based methods.

Finally, travel time prediction can also be categorized based on how many modes of transportation are spanned. Multi-modal problems predict a travel duration across more than one mode of transportation. A single mode problem predicts travel duration across one mode of transportation. Sometimes, prediction problems face uncertainty in the type of transportation modes leveraged or the number of modes utilized.

In the literature, a range of studies exist across the various attributes described above. In [14], a long-range, route-based problem is studied in the context of a German logistics company. A regression-based technique is used to make path-level travel time predictions, with consideration of route uncertainty.

Another path-based travel time problem is studied in [11], where vehicle trajectory data is used to generate short-range predictions for freight vehicles. A gradient boosting regression tree methodology is employed, and model experiments are conducted over three distinct routes. Interestingly, this work investigates both pre-start and post-start predictions.

In [18], a route-based problem is again studied, this time using a segment-based approach for a short-range prediction. The travel time prediction is broken up in to separate prediction problems for a series of Maryland highway segments, each less than two miles long. This work leverages historical travel time data and a gradient boosting approach that outperforms classical statistical methods.

Another short-range prediction problem is studied in [8]. This work also leverages boosting techniques, but uses an origin destination form instead of a route-based one. Travel time predictions of less than 24 hours are made for a postal service application. Instead of using route information, features such as scheduled trip duration and day of the week are leveraged.

An origin destination based travel time problem is also studied in [17]. In this work, a diverse feature set, including spatial attributes such as distance, temporal attributes such as time of day, weather attributes, and historical traffic attributes, is used to generate predictions for urban travel time. The authors find success using a deep-learning based model. Their model is deemed superior than others on the basis of mean average percent error, mean average error and computational cost.

Finally, a multi-modal, origin destination problem in the context of container transport is studied in [13]. Travel times of up to thirty days are estimated using support vector machines and tree-based methods. This work resembles that in this thesis most closely. However, key differences exist related to model input data and prediction timing. [13] uses goods-level, real-time tracking data. The authors use this data to infer routes using unsupervised learning techniques and base their predictions on these inferences. Finally, [13] appears to make post-start predictions only, as a pre-start prediction would not have any feature data available to it under the authors' current formulation.

Multi-modal, origin destination, long-range prediction studies are rare in the literature. This thesis contributes to this currently sparse research area. Furthermore, this work introduces a novel approach by making a pre-start prediction and omitting the use of real-time tracking data.

2.2 Data Sources

This section presents the data sources identified as being the most relevant to and capable of solving Victory's transit time prediction problem.

2.2.1 Order-Level Fulfillment Data

The primary data set used in this work is parcel-level fulfillment data. This data set provides attributes related to a parcel's journey through the supply chain after the associated order has been placed. It contains details about which distribution center the parcel originated from, what shipping speed was selected, what carrier serviced the delivery, and the order destination address. It also includes timestamps for important milestones in the fulfillment process such as when the order was placed and when the parcel was manifested, inducted and delivered.

2.2.2 Carrier-Provided Transit Times

The second data set used in this work is the set of carrier-provided transit time estimates. This data provides an estimated number of days between induction and delivery for a set of distribution center, destination zip code, shipping speed combinations, and are differentiated by carrier. Induction is when the parcel enters the carrier's network. Delivery is when the parcel is dropped off at the final destination. These carrier-provided transit time files are standard files that carriers provide Victory as a method of communicating their service levels. As mentioned previously, these files are not exhaustive and are missing certain distribution center, destination zip code, shipping speed, carrier combinations. From now on, the combination of distribution center, destination zip code, shipping speed and carrier attributes will be referred to as a configuration.

2.3 Current State Performance

The final piece of background that must be shared prior to discussing model development work is the performance of the carrier provided transit time estimates used currently. First, the metrics used to evaluate the performance of transit time estimates in this section and throughout the remainder of the thesis are described. Then, the current state performance is presented through the set of performance metrics and through a series visualizations.

2.3.1 Performance Evaluation Metrics

Four metrics are used to evaluate the performance of transit time estimates. These metrics are tailored to the business setting this problem is being solved in. The first metric is accuracy, which quantifies the percentage of estimates that match the actual value. Any error in transit time estimation is expected to compromise the performance of fulfillment-decisions. Therefore, this accuracy metric is of the utmost importance when evaluating the performance of transit time estimates.

The second metric used to assess estimate performance is mean absolute error. This metric is calculated by averaging the absolute value differences between the estimated and actual values. This metric quantifies the magnitude of error and is relevant because a larger error is expected to have larger negative effects to Victory's customer and business compared to a smaller one.

The third and fourth performance metrics are focused on negative errors, or errors in which the transit time is underestimated. Negative errors increase the risk of a late delivery and therefore seek to be minimized. A late delivery is when a parcel is delivered to the customer after the delivery date Victory promised the customer at order placement. Late deliveries are thought to have a severe negative impact on Victory customer satisfaction. Victory, therefore, aims to minimize their occurences. The third performance metric is percent negative error, which calculates the percentage of estimates that underestimate transit time. The fourth and final metric is mean negative error which calculates the average negative error across all records and enables the magnitude of negative error to be quantified. If an error is positive, that value is set to zero in this calculation.

2.3.2 Carrier-Provided Estimate Performance

Fulfillment decision-making currently is based on carrier-provided transit time estimates. This section shares a performance analysis of these estimates, including presentation of the evaluation metrics shared above. The value of these metrics change slightly based on the time period considered. For this section, we consider a two month window of data that is used several times throughout the remainder of this thesis, primarily in post-induction experiments. Calculating current state performance metrics based on this data enables direct comparison of model estimates to carrier-provided ones.

Carrier-provided estimates used currently are associated with an accuracy of 44.63%, a mean absolute error of 0.69 days, a percent negative error of 17.65%, and a mean negative error of 0.23 days. The distribution of prediction errors for the current state is shown in Figure 2-1, where the prediction error in days is calculated as the difference between the estimated and actual values. In this plot, errors for incorrect estimates are shown. Correct estimates, or those with zero error, are omitted from the visualization. This was done so the reader could more easily see the distribution of errors.

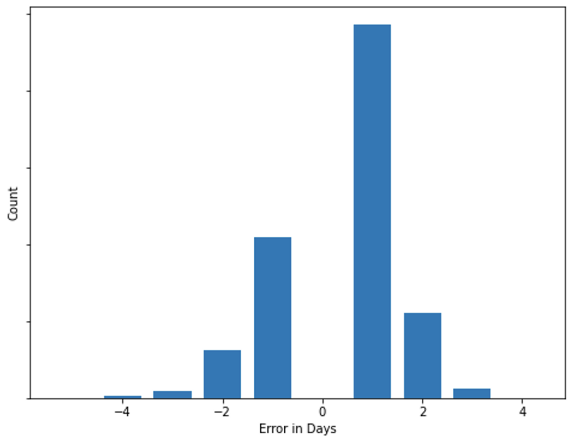

Figure 2-1: Current State Prediction Error Distribution For Incorrect Estimates

When analyzing the confusion matrix of prediction errors (Figure 2-2), one can see that the most frequent type of error, occurring for 14% of all records, is when the carrier predicts a two-day transit time while the actual transit time is one day.

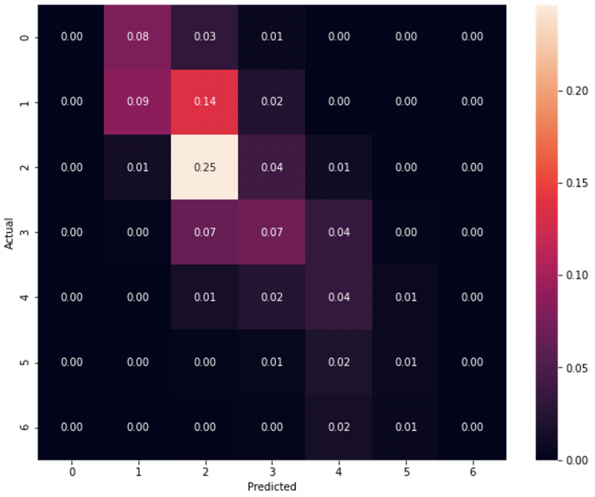

Figure 2-2: Current State Prediction Error Confusion Matrix

By viewing the box plots for prediction errors by shipping speed as shown in Figure 2-3, one can see that there is no significant variation across categories. When doing the same by carrier (Figure 2-4), one can see significant variation across carriers. Some carriers have a very narrow error distribution, while others have a much wider one. This observation demonstrates that there is variability across carriers in their ability to accurately predict transit time.

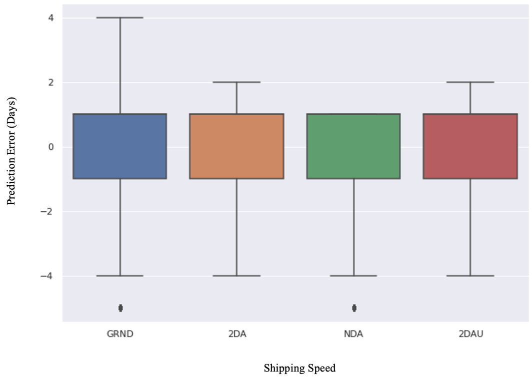

Figure 2-3: Current State Prediction Error by Shipping Speed

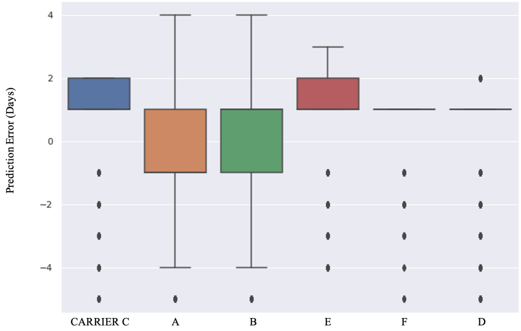

Figure 2-4: Current State Prediction Error by Carrier

Chapter 3

New Models for Transit Time Estimation

This chapter presents the development process used to create a predictive model for transit time estimation. The process begins with the creation of the analytical data set through data sourcing, cleaning, sampling and feature engineering. The process continues with experimental iterations on model parameters and hyperparameters. The aim of this experimentation is to determine the design elements most favorable for accurately predicting transit time. The model development process concludes with selecting the best model attributes, discovered through experimentation, and thus arriving at the best model. Please note, the term "best" in this thesis refers to the best model within the scope of this work. It is akin to a local best, rather than a global best.

3.1 The Analytical Data Set

The analytical data set is used to train, validate and test the model. Its creation begins with merging the order-level fulfillment and carrier-provided transit time data sets on the configuration attribute variables (distribution center, destination zip code, shipping speed and carrier). From there, cleaning, sampling and feature engineering occur.

3.1.1 Data Cleaning

The merged data set is cleaned by deleting rows with missing data and removing duplicate rows. Records not associated with Victory's main distribution centers are removed for data density purposes. Finally, rows with incorrect state code data or state codes outside of the main 50 states (e.g. AA for Armed Forced Americas) are removed. The total number of rows removed is insignificant compared to the initial data set size.

3.1.2 Sampling

Sampling is a powerful strategy as it allows for the creation of a manageable sized data set without losing valuable information. In other words, a data set with representative variability can be created without requiring every data observation. This supports the development of high-performance and computationally efficient model. Computation efficiency is important from a cost and emissions perspective, and from an implementation perspective as it enables the model to train and generate estimates quickly.

Without sampling, the time window of the analytical data set is forced to be much narrower, inherently limiting the information the model sees. With a limited view of transit time dynamics, the model would be less equipped to make correct predictions.

Proportional stratified sampling is implemented with each configuration (distribution center, destination zip code, carrier, shipping speed) considered as stratum. Within each stratum records are randomly selected. The number of records selected is proportional to the number of records in the stratum compared to the overall data set. For example, if the data set had 100 records, and there were 10 records in a given stratum, when implementing 10% proportional stratified sampling 1 record from that stratum would be selected. If there were 20 records in another stratum, 2 records would be selected for that configuration. This approach enables the creation of a smaller data set that is still representative across configurations.

The sampling percentage selected is a function of the chosen time period for the data set. A longer time period is associated with more records in the original data set and thus a smaller sampling percentage. This relationship exists to keep the final analytical data set within a reasonable size. This size is selected based on a target model training time that enables rapid experimental iteration. A time period of two months is selected using an experimental approach which will be described in a subsequent section of this thesis. The sampling percentage associated with this time period is 50%.

3.1.3 Feature Engineering

Features are a term used in machine learning to describe the independent variables used in model training and estimate generation. In this work, a set of features is curated by using as-is variables in the data set and by creating new variables based on existing variables in the data set. The as-is variables used in the feature set are displayed in Table 3.1. The initial features from newly created variables are shown in Table 3.2. In later sections of this thesis, additional features will be added to the feature set and will be explained then.

The dependent variable in this prediction problem is actual transit days. This is calculated by finding the number of delivery eligible days between when a parcel was inducted into a carrier's network, and when it was delivered to the consumer. Bank holidays are not considered delivery eligible. Weekend eligibility is determined by carrier. If a carrier delivers to more than 50% of zip codes on a given weekend day, then that day is considered delivery eligible. Records with negative actual transit days, due to data errors, are removed from the data set.

Table 3.1: Features from Existing Variables

Feature Name	**Description**	**Type**
PLNTCD	Distribution center parcel was shipped out of	Categorical
WMS_STDSHPGSVCLVL	Shipping speed associated with the parcel, consisting of the following categories: ground (GRND), two day air (2DA) and next day air (NDA)	Categorical
SHIPTOZIP_FIVE_DIGIT	Full 5 digit zip code associated with consumers' delivery address	Categorical
SHIPTOST	State associated with consumers' delivery address	Categorical
CARRIERGROUPNAME	Third party carrier responsible for delivery of parcel	Categorical
TNT	Carrier provided transit time estimate in days	Categorical/Numeric

Table 3.2: Initial Features from Newly Created Variables

Feature Name	**Description**	**Type**
ACTUAL_TNT	Number of delivery eligible days between when carrier inducted parcel into their network and when they delivered it to the consumer	Numeric
SHIP_TO_ZIP_TWO_DIGIT	Two digit zip associated with destination address	Categorical
SHIP_TO_ZIP_THREE_DIGIT	Two digit zip associated with destination address	Categorical
MANIFEST_HOUR	The hour of the day manifestation occurred	Categorical
FIRST_SCAN_HOUR	The hour of the day the carrier inducted, or first-scanned, the parcel into their fulfillment network	Categorical
MANIFEST_DAY	The day of the week manifestation occurred	Categorical
FIRST_SCAN_DAY	The day of the week the carrier inducted, or first-scanned, the parcel into their fulfillment network	Categorical
MANIFEST_MONTH	The month of the year manifestation occurred	Categorical
FIRST_SCAN_MONTH	The month of the year the carrier inducted, or first-scanned, the parcel into their fulfillment network	Categorical
DISTANCE	Road miles between distribution center and destination zip codes; removed rows where distance calculation could not be performed	Numeric

3.1.4 Exploratory Data Analysis

An Exploratory Data Analysis is conducted to develop a baseline understanding of the data set. Specifically, it is conducted to generate insights related to the composition of the data set across configuration attributes and related to transit time patterns. A new data set, consisting of three million records from a six month time period, is used for the Exploratory Data Analysis. It is created using the same method described above, except random sampling is leveraged instead of proportional stratified sampling. The data set spans the months November to April and is inclusive of the annual retail peak period. This sample is selected because it provides the data necessary to pick up on seasonal trends, while also allowing for adequate data density and fast computations.

Analyzing the distribution of records across configuration attributes, it is observed that a dominate share of parcel volume is associated with the ground shipping speed as shown in Figure 3-1. A large portion of volume is also associated with two primary carriers as shown in Figure 3-2. Finally, it can be seen in Figure 3-3 that records are not uniformly distributed across distribution centers, with one distribution center being much larger than the rest. Analysis of volume across destination zip codes is not conducted because it will not provide useful insights given the large number of destination zip codes available. Transit time behavior is also analyzed as part of the exploratory data analysis.

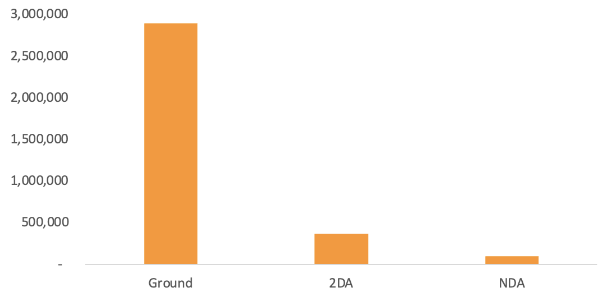

Figure 3-1: Record Count Per Shipping Speed

The dependent variable, actual transit time, is analyzed alone and in relation to several independent variables. In the data sample used for exploratory data analysis, 99.77% of actual transit time values are seven days or less. The remainder are distributed unevenly between 8 and 97. Most values in this range are extreme and likely are erroneous data. The distribution of actual transit times that are 7 days or less have a slight positive skewness and are centered on 2 days as shown in Figure 3-4. The mean and standard deviation associated with this distribution are 2.06 days and 1.36 days, respectively.

Actual transit times vary considerably across configuration attributes. Figure 3-5, 3-6 and 3-7 show actual transit time box plots by distribution center, shipping speed,

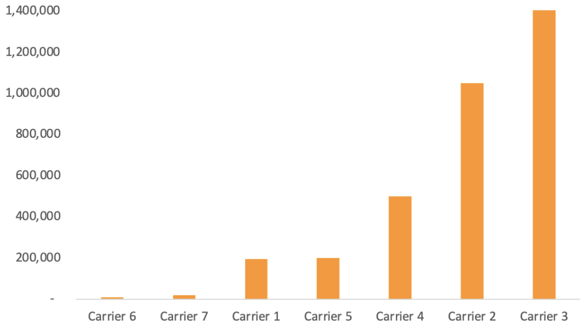

Figure 3-2: Record Count Per Carrier

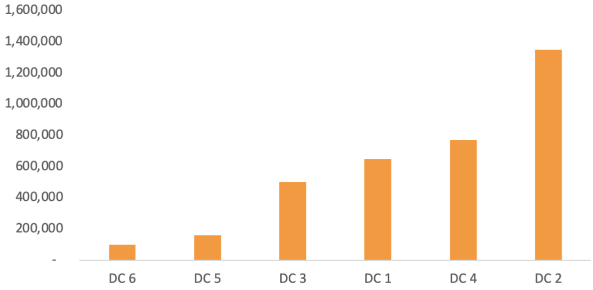

Figure 3-3: Record Count Per Distribution Center

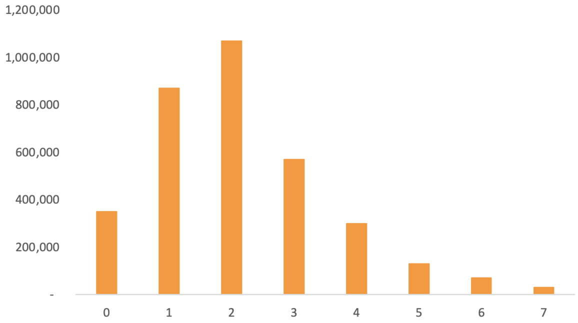

Figure 3-4: Actual Transit Time Distribution in Days

and carrier, respectively. Although some distribution centers have similar transit time distributions, considerable variation exists across all distribution centers. Across shipping speeds, transit time distributions vary considerably, which is expected based on the nature of this variable. Finally, it can be seen that transit time distributions vary considerably across carriers too. These findings indicate that the configuration attributes should be useful in making transit time predictions. Transit time distributions are also analyzed across time of day, day of week and month of year features. No considerable variation is seen. Rather, the distributions look similar across the various time points. This is an important finding as it indicates that observed seasonality is not a factor in this prediction problem. Transit times behave similarly in and out of retail peaks and thus incorporating peak status in the model is likely not necessary.

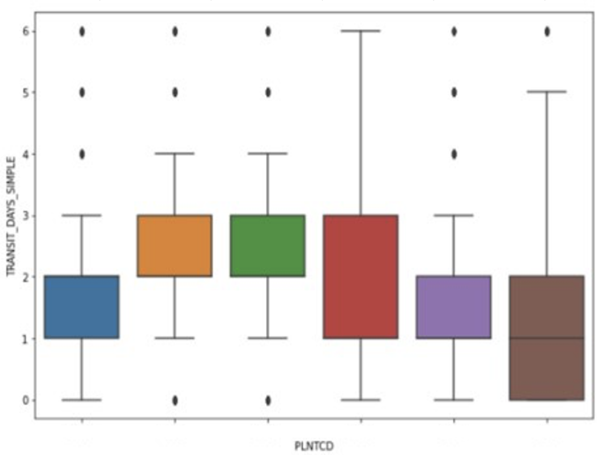

Figure 3-5: Actual Transit Time Distribution (in Days) By Distribution Center

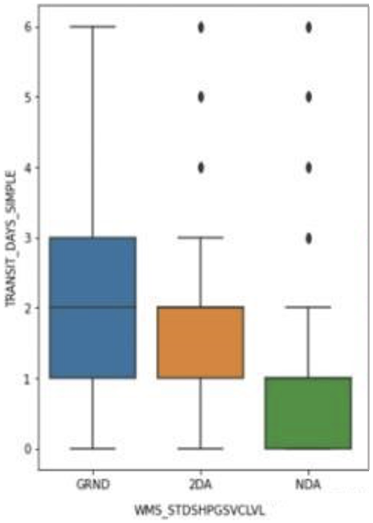

Figure 3-6: Actual Transit Time Distribution (in Days) By Shipping Speed

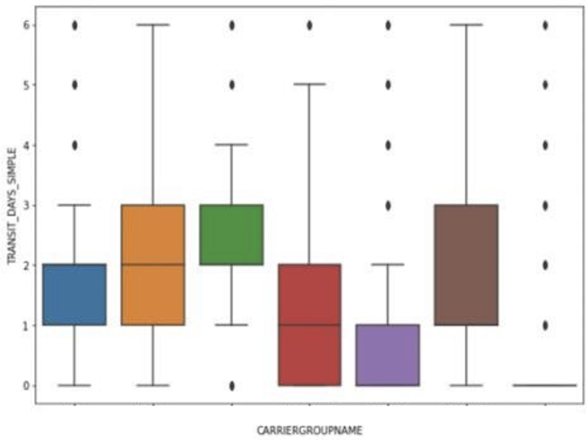

Figure 3-7: Actual Transit Time Distribution (in Days) By Carrier

To conclude the exploratory data analysis, the correlation coefficients between actual transit time and the numeric independent variables are calculated. The correlation between actual transit time and distance is 0.45, which is fairly strong considering Victory's delivery promises are not highly tailored to the customer based on the customer's proximity to the selected distribution center. This is a surprising finding, as a longer distance is expected to be associated with a longer transit duration. The correlation between actual transit time and carrier provided transit is 0.60, which is indicative of the fact that carriers do sometimes predict transit time correctly.

3.2 Experiment Structure

With the initial analytical data set created, a series of experiments can now be conducted to determine the best model design elements. Again, the term best is used here to describe a local best, rather than a global one, since all possible scenarios cannot be tested. The term design element is used here to describe aspects of the model such as its structure, parameters, and hyperparameters. The majority of model experiments are run sequentially. In a scenario with unlimited computing resources, the experiments could be run simultaneously in a massive grid search algorithm. Grid search is a technique were all possible combinations of settings or design elements are tested to determine the best combination. This approach is exhaustive but not feasible in this setting due to computing limitations.

The analytical data set is split into training and testing data using a 70/30 ratio prior to running model experiments. Shuffling, or randomizing the data, is not conducted prior to the split to maintain the chronology of the observations. This means that the testing set corresponds to the last 30% of the data in time. It is important to maintain the chronology of the observations when splitting into training and testing data sets because this is representative of model's eventual production environment. Once the model is deployed in production, it will only have historical data to train on.

It will not have access to data from the future when making predictions. In model experimentation, ensuring the training set only contains data from prior to the testing set simulates this constraint.

Post-induction models, or those that use data from when the order was placed up until when it was inducted into a carrier's network, are experimented with first. These are called post-induction models because they make a prediction right after induction occurs. As described previously, this type of model cannot be used for fulfillment decision-making. However, they can enable understanding of transit time dynamics and provide insights that enhance the post-purchase models that can be used for fulfillment decision-making. After a series of post-induction model experiments are conducted, focus shifts to post-purchase models. Findings from post-induction model experiments are considered and additional experiments are run using post-purchase models. The experimentation process concludes with the creation of a model that takes into consideration the experimental findings. This model is considered the best and final for this scope of work.

3.3 Post-Induction Model Experiments

Post-induction models use the full feature set described previously as independent variables and actual transit time as the dependent variable. The following subsections detail various experiments conducted on post-induction models and their findings. Throughout experimentation, models are compared on the basis of accuracy. The other three performance metrics introduced earlier are not included in the comparison in order to simplify the experimentation process. The other performance metrics are used when selecting the best model at the conclusion of the experimentation process, and when comparing the model performance to the current state performance.

3.3.1 Model Type

Various regression and classification models are tested to determine which has the best performance. Regression models predict the continuous number of transit days, which is then rounded up to the nearest whole number prior to assessing model performance. This is because transit times are always communicated as integers in the Victory organization. Classification models predict categories associated with the discrete number of transit days. The linear methods tested include linear regression, ridge regression [7] and elastic net [19]. The non-linear methods tested include logistic regression, decision trees, gradient boosted trees, and random forest [3]. These models are selected because they are well known methods in machine learning and together form a diverse set: varying on dimensions such as complexity, interpretability, and performance. Where possible, models are tested on both a regression and a classification problem formulation.

Random forest models perform the best out of the tested set, in terms of accuracy.

Decision trees are a moderately close second. Linear methods in general perform worse than tree-based methods. Performance of regression and classification models within the same model type are similar. Decision tree classification is used as the model type to run most subsequent post-induction model experiments with due to its strong predictive performance, short training time, and interpretability. However, when time permits, some experiments are conducted across the entire set of model types or using a random forest approach. Experiments that use a different model type than a decision tree classifier are noted as such.

3.3.2 Data Set Size

The size of the data set is varied to determine the impact on predictive performance. Seven data sets ranging from five thousand records to two million records are created, keeping the time period constant. The data sets are split into test and train, and seven decision tree classification models are built.

Unsurprisingly, the larger the data set is, the better performance the associate model had. However, the relationship is not linear. The degree of performance improvement from 5,000 records to 100,000 records is much greater than that from 100,000 records to 2,000,000 records.

3.3.3 Upper Winsorization

In the exploratory data analysis, extreme values of actual transit time were discovered. A hypothesis is formed that these outliers could negatively impact model performance. To address these outliers and test this hypothesis, an experiment on upper winsorization of the dependent variable is conducted. Upper winsorization works as follows: an upper winsor parameter of 0.005 will set any actual transit time greater than the 99.5^{th} percentile to the 99.5^{th} percentile value. For the current analytical data set, this serves to narrow the dependent variable distribution to values between zero and seven inclusive. In other words, upper winsorization is used to replace outliers with upper bound values in the training and testing sets.

Models are created using the two different data sets: the winsorized data set and the non-winsorized data set. Although the two models have the same model type, they are not identical models because they are trained on different data sets. The model with the winsorized data set outperforms the one with the non-winsorized data set. Therefore, the above hypothesis is proven true. Removing outliers from the analytical data set supports model performance.

3.3.4 Independent Models per Carrier

Experimentation is conducted to determine the impact of creating separate models per carrier instead of utilizing one model for all carriers. Two models are created for the two largest carriers, and another is created for all other carriers. The analytical data set used in previous model iterations is split in the same manner and used for model training and testing.

Slight performance improvement is seen for one of the large carrier models, while the remaining models result in performance degradation. This degradation could be a result of reducing the number of training observations per model when the training data was split by carrier.

3.3.5 Sampling

An experiment is conducted to confirm the hypothesis that proportional stratified sampling improves model performance. In this experiment, the full set of model types is run with random sampling and compared to the performance of the models with proportional stratified sampling. Across nearly all model types, models with proportional stratified sampling perform better than those with random sampling. This confirms the proportional stratified sampling hypothesis.

3.3.6 Data Set Time Period

The time period associated with the analytical data set is experimented with. Time periods of one month, two months, and fourteen months are tested. Time periods begin with the most recent full month. For example, for the time period of two months, the most recent complete two months are included. A test using two of the most recent complete months plus one month from twelve months ago is also included. This is in an effort to represent both recent behavior as well as any seasonal behavior that would be seen during the same month of the previous year. Sampling percentages that preserve the size of the analytical data set across each test case are used and are shown in table 3.3.

Again, models are created for each test case and the accuracy performance metrics are calculated and compared. The one month time period is associated with the best performance, followed by two month, two plus one month, and fourteen months. Please note, this experiment is run using a random forest classifier.

Table 3.3: Time Period Cases and Associated Sampling Percentages

Time Period	**Sampling Percentage**
1 Month	100%
2 Months	50%
14 Month	10%
2 + 1 Month	25%

3.3.7 Feature Importances in the Best Post-Induction Model

Prior to advancing to post-purchase model experimentation, the feature importance scores of the best-performing post-induction model are assessed. Specifically, the relative magnitude of the feature importance scores are compared. This provides insights into which variables are most useful in predicting transit times and informs the design of post-purchase models. The best-performing post-induction model is a random forest classifier created using a one month, two million record analytical data set with dependent variable outliers replaced via winsorization. The performance metrics of this model compared to the current state are presented in Table 3.4. Note that the current state performance metrics are slightly different than the ones present earlier in this thesis due to the difference in time period considered. The top twenty features, in terms of importance scores, are shown in Table 3.5. Importance scores are based on the mean decrease in impurity for a given feature across all trees in the random forest model.

The carrier provided transit estimate is considered the most important feature for predicting transit times. This makes sense given the moderate correlation between this independent variable and the dependent variable. Configuration attributes, namely carrier name, distribution center and shipping speed also have high importance scores. Recalling our findings from the exploratory data analysis, specifically that transit time distributions vary significantly across configuration attributes, this also makes sense.

Interestingly, we observe that first-scan day features have high importance scores. This communicates that the day a parcel is scanned into a carrier's network impacts its transit time. Knowing this information can aid in making a stronger prediction. Three first-scan hour features, namely those associated with the first three hours of the day, also appear in the top twenty feature list. The importance of first-scan related variables will be kept in mind in future model experimentation.

Table 3.4: Post-Induction Random Forest Classifier Results

Metric	**Random Forest Classifier**	**Current State**
Accuracy	77%	40%
Mean Absolute Error	0.28 days	0.73 days
Percent Negative Error	14%	15%
Mean Negative Error	0.17 days	0.20 days

Table 3.5: Post-Induction Random Forest Classifier Feature Importance Scores

Rank	**Feature**	**Score**
1	Carrier Provided Transit Time	0.086107
2	Carrier A	0.018591
3	Carrier B	0.01694
4	Destination State California	0.015586
5	Carrier B	0.015552
6	First Scan Day Friday	0.013816
7	Distribution Center 4	0.013161
8	Distribution Center 2	0.013048
9	First Scan Day Saturday	0.012684
10	First Scan Day Monday	0.010966
11	Shipping Speed Ground	0.009812
12	Carrier D	0.009787
13	Distribution Center 1	0.008586
14	Manifest Day Friday	0.008551
15	Shipping Speed Two Day Air	0.008081
16	First Scan Hour 1	0.007875
17	First Scan Hour 2	0.007528
18	First Scan Day Thursday	0.00752
19	First Scan Hour 0	0.007161
20	First Scan Day Tuesday	0.007022

3.4 Post-Purchase Model Experiments

The post-induction models discussed in the previous section use features up to and including the first-scan event. Post-purchase models, the focus of this section, use features up to and including the purchase or order creation event. As discussed, post-purchase models are preferred for the fulfillment decision-making use case. In this section, experiments conducted on post-purchase models are shared. Post-induction model experiments are not rerun, but rather built upon based on the assumption that findings are transferable between these two model designations.

3.4.1 Initial Post-Purchase Models

For post-purchase models, a new analytical data set is created. The initial version is the same as the post-induction data set except that all features after the purchase event are removed and new purchase features are added. Features related to manifestation and induction are removed. Hour, day and month features are added for the order purchase or click event. This is done to add temporal features back into the data set. In the post-induction data set all temporal features occur after the purchase event. When the transformation from post-induction data to post-purchase data occurs, these temporal features are removed.

The full set of model types are run using the initial post-purchase data set. Model performances are worse than the post-induction model results. This is expected, given the prediction is being made earlier in the order cycle and utilizes less information about the parcel's fulfillment journey. It also makes sense when recalling that features with high importance scores, namely those related to the first-scan event, are not included in the post-purchase model.

3.4.2 Model Type: CatBoost

Due to the strong performance of tree-based methods in the experimentation so far, additional tree-based methods are experimented with. In particular, a CatBoost regressor and classifier are tested. CatBoost is a popular gradient boosting algorithm known for its efficiency, strong performance on complex prediction tasks, and support of various feature types [5]. Root mean squared error and multiclass are used as loss functions for the regressor and classifier, respectively.

These new models outperform all previous models, with the exception of the CatBoost regressor having slightly lower accuracy than the random forest classifier. However, the CatBoost regressor is associated with approximately 88% less training time than the random forest classifier (3 hours versus 25 hours). The CatBoost classifier performs the best out of all tested models, in terms of accuracy.

3.4.3 Lagged Features

In hopes of increasing performance closer to post-induction model levels, new features are experimented with. Recalling the importance of the first-scan features in the post-induction models, two new classes of features using historical order information are created: one related to the first-scan event and the other related to actual transit time values. These are referred to as lagged features.

The aim of the lagged first-scan features is to provide the post-purchase model with information on first-scan dynamics. Instead of using the known first-scan day, which is not allowed in the post-purchase model designation, features are created to represent the distribution of first-scan days across historical similar orders. Specifically, seven numeric features are created for the percentage of historical similar orders with a first-scan day of Monday, Tuesday, Wednesday, etc. Similar orders were first defined as ones with the same configuration (distribution center, destination, carrier, shipping speed). After additional thought, the approach was revised and similar orders are defined as ones with the same distribution center, carrier, shipping speed and click day. Click day was added because of the strong relationship that exists between click day and first-scan day. Destination was removed in an effort to broaden the definition of similar orders and expand the historical order set available for a given record.

Only similar orders that had been inducted by the time the given order was placed are included in the historical order set. This is to avoid calculating lagged features using data that would not actually be available when the transit time prediction is made in production. In production, only first-scan information for orders that have been inducted by the time the given order was placed would be available. It is important to capture this constraint accurately in feature creation and model development.

The second class of lagged features created for post-purchase predictions are related to actual transit time values. These features are created by gathering historical similar orders and calculating various metrics across those orders' actual transit time values. In this case, similar orders are defined as ones with the same configuration attributes. Only historical orders that had been delivered by the time the given order was placed are included in the calculation of this class of lagged features, for the same reason described above. Descriptions of these lagged actual transit time features are provided in Table 3.6.

The initial post-purchase data set is updated to include all new lagged features. The full set of model types are run against this updated data set. Model performance improves with the inclusion of the lagged features across all model types except for the decision tree classifier and regressor. Improvements ranged from 1.5% to 14.5% in terms of percent change in accuracy, while degradations were on the order of -15%. Percent change in accuracy is calculated using the equation below.

Table 3.6: Lagged Actual Transit Time Features

Feature Name	**Description**
5_OBS_MAX	Maximum actual transit time across last 5 similar orders
5_OBS_MIN	Minimum actual transit time across last 5 similar orders
5_OBS_MED	Median actual transit time across last 5 similar orders
5_OBS_AVG	Average actual transit time across last 5 similar orders
TO_DATE_AVG	Average actual transit time across all historical similar orders
PREV_VALUE	Actual transit time associated with the most recent similar order

$$\text{Percent Change in Accuracy} = \frac{\text{Current Accuracy Score - Previous Accuracy Score}}{\text{Previous Accuracy Score}}$$

The feature importance scores for the best-performing model, the CatBoost classifier, are provided in Figure 3-8, where days of the week represent the lagged first-scan features. It can be seen that all lagged features have non-zero importance scores. This indicates they have predictive power in the context of this transit time problem. The feature describing the average actual transit time across all historical similar orders replaces the carrier provided transit time feature as the most important. This indicates that historical transit time values may be a good predictor of future values. All lagged first-scan features have moderately high feature importance scores, notably beating the importance scores of distance, destination and shipping speed. This indicates these features may be picking up on the signals previously provided by the first-scan day features.

To further validate the hypothesis that lagged features positively impact performance, several model trials are conducted using a limited feature set. In these trials, the lagged features are removed, in addition to some other features. Unexpectedly, one of these simpler models outperforms the model with the full, lagged feature inclusive, feature set. This finding motivates experimentation on alternative lagged feature designs and feature selection methodologies.

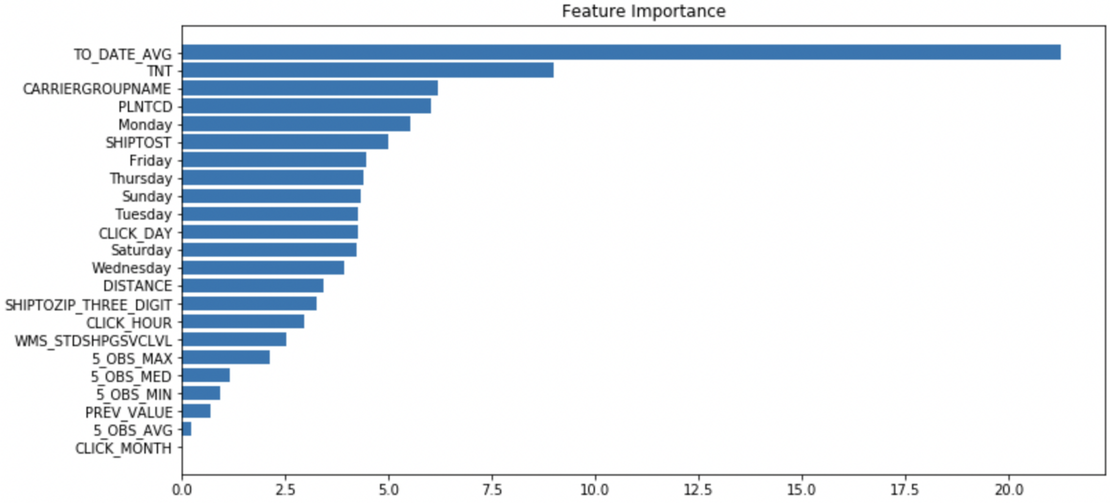

Figure 3-8: Feature Importance Scores of CatBoost Classifier with Lagged Features

3.4.4 Updated First-Scan Lagged Features

In post-induction model experiments, first-scan attributes proved to be useful in predicting transit times. However, the initial iteration of lagged first-scan features did not improve predictive performance. Another iteration of first-scan lagged features is created in hopes of providing the valuable first-scan signal to the post-purchase models. This second iteration encompasses the creation of two new features that represent the following: 1) for a given order, the most frequent first-scan day seen in the set of all historical similar orders and 2) for a given order, the first-scan day associated with the most recent historical similar order.

Again, historical orders are defined as those that have been delivered by the time the given order was placed. Similar orders are defined as those with a common set of attributes. Experiments are conducted with different sets of attributes to define similarity. The first trial defines similar orders as those with the same distribution center, carrier, shipping speed and click day. The second trial defines similar orders as those with the same distribution center, carrier, shipping speed, click day and click hour. When creating similarity definitions, the need to be specific enough to find orders with the same transit behavior is balanced with the need to have ample records in the historical similar order set.

The percentage of time each of the two new lagged first-scan features correctly represented the actual first-scan day is calculated to gauge how strong these new features may be. The percentages range from 35% to 55%, with the mode feature and second similarity definition combination achieving the best matching capability and highest percentage.

The performance impact of the new lagged first-scan features is evaluated in the feature selection experimentation described in the next section.

3.4.5 Feature Selection

A simpler feature set having stronger performance than a more complex one motivates the development of a systematic feature selection process. This process begins with creating a model based on only configuration attribute features. It continues with adding new features to the model one at a time. If the feature improves model performance, the feature remains in the set. If not, it is removed.

In this process, experimentation is also conducted on representing features as categorical or numeric. If a feature can be represented as both categorical and numeric, two separate instances of the feature for each representation are included in the feature selection process. This provides insight into whether the feature as a categorical or numeric variable is more useful in predicting transit times.

Using this process, the best feature set for the CatBoost model is found to consist of the following features: distribution center, three digit destination zip code, destination state, shipping speed, carrier, carrier provided transit time, distance, day of week of purchase, hour of day of purchase, average actual transit time over last 5 historical similar orders, minimum actual transit time over last 5 historical similar orders, and median actual transit time over last 5 historical similar orders. Lagged first-scan features are not included in the best feature set, indicating that the second iteration of features still does not provide the predictive power hoped for. Categorical representations of carrier provided transit time and lagged actual transit time features outperform numeric representations, while the reverse is true for the click hour feature.

3.4.6 CatBoost Hyperparameter Tuning

Improving transit time estimates through hyperparameter tuning is explored in the context of the CatBoost model. Hyperparameters are parameters that influence the model learning process and are set prior to the model being fit on training data. The goal of the tuning process is to discover the hyperparameter values that unlock the best predictive performance. The following hyperparameters are experimented with in this process: iterations, depth and L2 leaf regularization. Iterations is the number of boosting rounds. Depth describes the maximum allowable level, counted from root to leaf, of the trees in the model. L2 leaf regularization is used to penalize the complexity of the trees and helps prevent overfitting. These hyperparameters are selected based on recommendations provided in Catboost documentation. Hyperparameters are tuned in an iterative manner rather than simultaneously. Experimentation starts with the baseline model, or that with default hyperparameters. Iterations is tuned first. The optimal iterations value is then used in the next experiment on depth. Finally, the optimal iterations and depth values are used in the L2 leaf regularization experiment.

Iterations is tuned by setting the number of iterations to a large value (10,000), setting the learning rate, or gradient step size, to a small value (0.03) and using the CatBoost overfit detector. When supplied an evaluation set, this detector will stop model training when evaluation set performance improvement stagnates. In this case, the overfit detector was set to stop training after 200 consecutive iterations of accuracy deterioration. The optimal iterations value is considered the one with the best accuracy on the evaluation set. For this process, new train, validation and test splits are created using 70%, 15% and 15% of the data set, respectively. The validation set is used as the evaluation set for the overfit detector.

The best depth and L2 leaf regularization values are determined using two grid search processes. For depth, values of 6,7,8,9 and 10 are tested. For L2 leaf regularization, values of 1,5,25,100, and 1000 are tested. CatBoost grid search does not support use of the overfit detector. Therefore, the best hyperparameter values are chosen based on training set performance.

This hyperparameter tuning process is conducted on three different models and associated feature sets: the full feature set including lagged features, a feature set without lagged features, and the best feature set determined through the feature selection process. In two out of the three models, iterations tuning results in a model with worse performance than the baseline. This could be explained by the overfit detector being too conservative and stopping too soon. This would result in the number of iterations being suboptimal, thus compromising performance. The iterations tuning process results in improved model performance for the feature set created by feature selection. The difference in iterations tuning results across models confirms the expectation that hyperparameter tuning results are not transferable across models. Hyperparameter tuning should be conducted again once the final model design is chosen.

Depth and L2 leaf regularization tuning do not meaningfully improve performance in any of the models. This means that either the default values are similar or exactly the same as the best value found, or that these hyperparameters do not meaningfully impact model performance in general for this prediction problem.

3.4.7 Model Type: XGBoost

Another advanced tree-based model type, namely XGBoost, is tested given the success with CatBoost. XGBoost is another popular gradient boosted trees method that is known to be highly performant and efficient [4]. The performance of an XGBoost classifier is evaluated by running the feature selection process described above and comparing the results to those of the CatBoost feature selection process. A softmax objective function is used for the XGBoost model given the multi-class nature of transit time prediction problem.

The best XGBoost model's performance is slightly worse than the best Catboost model's. However, the XGBoost model takes 5 minutes to train versus the CatBoost's 45 minutes when using the one month data set with proportional stratified sampling.

The feature set associated with the best XGBoost model is different than the feature set associated with the best CatBoost model. A visual comparison of the feature sets is provided in Table 3.7. The XGBoost feature set is smaller and includes fewer lagged actual transit time features. Additionally, there is consistency between the XGBoost and CatBoost results in terms of the feature describing the hour of day the order was placed enabling better performance as a numeric variable and the carrier provided transit time enabling better performance as a categorical variable. Finally, in both XGBoost and CatBoost models, the lagged first-scan day features do not increase model performance.

Table 3.7: Best CatBoost Model and Best XGBoost Model Feature Comparison

Feature Name	**Used in CatBoost Model?**	**Used in XGBoost Model?**
Distribution Center	Yes	Yes
Three Digit Destination Zip Code	Yes	Yes
Destination State	Yes	Yes
Shipping Speed	Yes	Yes
Carrier	Yes	Yes
Carrier Provided Transit Time	Yes	Yes
Distance	Yes	Yes
Day of Week of Purchase	Yes	Yes
Hour of Day of Purchase	Yes	Yes
Average Actual Transit Time Over Last 5 Historical Similar Orders	Yes	No
Minimum Actual Transit Time Over Last 5 Historical Similar Orders	Yes	No
Median Actual Transit Time Over Last 5 Historical Similar Orders	Yes	No
Actual Transit Time Associated With Previous Historical Similar Order	No	Yes

3.4.8 Encoding of Categorical Variables

The final post-induction model experiment conducted is related to the encoding of categorical variables. In models up to this point, one hot encoding is used to encode categorical variables prior to model training. One hot encoding consists of transforming a categorical variable to a set of binary variables, one for each level of the categorical variable, and setting the binary variable to one if a given observation is associated with that binary variable's level.

Three alternative encoding techniques are tested using XGBoost models: leave one out encoding, target encoding, and ordinal encoding. Leave one out encoding consists of replacing each level of a categorical variable with a numerical value based on the mean of the target variable for that level, computed over all observations except the current observation. Target encoding uses the same technique as leave one out encoding, but includes the current observation in the mean calculation. Ordinal encoding transforms categorical variables to numeric ones by replacing each level of a categorical variable with an integer value. Ordinal encoding, fit on the full data set, is found to be associated with meaningfully better performance than the other methods across all performance metrics.

The final aspect of the categorical encoding experiment is to determine the best feature set using ordinarily encoded variables. Since the features technically change with the updated encoding methodology, reassessing the best feature set is considered necessary. The same feature selection process described above is followed.

The best feature set changes with the change of encoding methodology. The best set with ordinally encoded categorical variables is smaller than the one with one hot encoded variables. The best set with ordinal encoding consists of the following features: distribution center, three digit destination zip code, destination state, shipping speed, carrier, carrier provided transit time, day of week of purchase and hour of day of purchase (as a numeric variable). Notably, this set does not include distance or the lagged feature that provided the actual transit time of the most recent historical similar order, both of which were included in the one hot encoding best feature set.

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 4

Results

4.1 The Best Model

Insights from post-induction and post-purchase model experiments are leveraged to create what is considered the best model in this scope of work. In the following subsections, the design elements, performance and feature importance scores of this model are discussed.

4.1.1 Design Elements

It is concluded that the best model is an XGBoost classifier. This model uses a one month, roughly two million record analytical data set curated through proportional stratified sampling on configuration attributes. Winsorization is applied to the analytical data set as a method to replace outliers and restrict actual transit time values to be between zero and seven. The model leverages the following features to train and make predictions: distribution center, three digit destination zip code, destination state, shipping speed, carrier, carrier provided transit time, day of week the order was placed, and hour of day the order was placed (as a numeric variable). Categorical variables are encoded using ordinal encoding.

Although the CatBoost model has slightly better performance, the XGBoost model is selected instead for two reasons. First, the XGBoost model trains in only 5 minutes, which is 88% less time than the CatBoost model. Second, the XGBoost model requires a smaller feature set to unlock roughly the same performance as the CatBoost model. A smaller feature set supports interpretability and also simplifies the analytical data set creation process.

4.1.2 Model Performance Versus Current State

The performance of the best model in relation to the carrier provided estimates for the same period are presented in Table 4.1. The XGBoost model is associated with significant improvements in accuracy and mean absolute error. The XGBoost model has marginally higher percent negative error and mean negative error.

This means that the XGBoost model is associated with more correct estimates. When the XGBoost estimates are wrong, they are wrong by a smaller amount than the carrier estimates on average. In terms of minimizing negative impact on fulfillment decision-making, a smaller error is preferred to a larger one. Carrier estimates have slightly less negative error occurrences and slightly smaller negative error on average compared to the XGBoost model estimates.

Table 4.1: Best Model Estimates vs Carrier Provided Estimates: Performance Metrics

Performance Metric	**Best Model**	**Carrier Provided**
Accuracy	67%	45%
Mean Absolute Error	0.38 days	0.68 days
Percent Negative Error	18%	17%
Mean Negative Error	0.23 days	0.22 days

When viewing the error distribution of the XGBoost estimates, as shown in Figure 4-1, it can be seen that the vast majority of errors are only one day. Additionally, it can be seen that underestimations are more frequent than overestimations. Comparing the XGBoost error distribution to that of the carrier provided estimates shown in Figure 4-2, it can be seen that the carrier estimates are more likely to underestimate transit time too. In the case of overestimation, the carriers might be quoting longer delivery times as a way to decrease late deliveries and improve customer satisfaction.

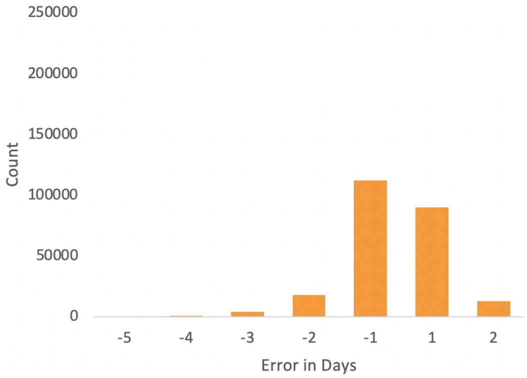

Figure 4-1: Best Model Error Distribution

To investigate the XGBoost model's errors further, a confusion matrix is generated and provided in Figure 4-3. Through this visualization it can be seen that errors are fairly well distributed across error types. For example, the same number of records

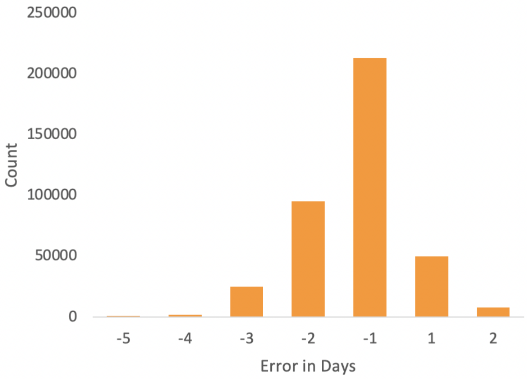

Figure 4-2: Carrier Provided Estimate Error Distribution

are associated with the following error types: predicting 1 day when the actual transit time is 2 days, predicting 2 days when the actual is 3 days, predicting 2 days when the actual is 1 day, and predicting 3 days when the actual is 2 days. To further compare XGBoost estimates to the carrier's, a confusion matrix for the carrier provided estimates is generated and is displayed in Figure 4-4.

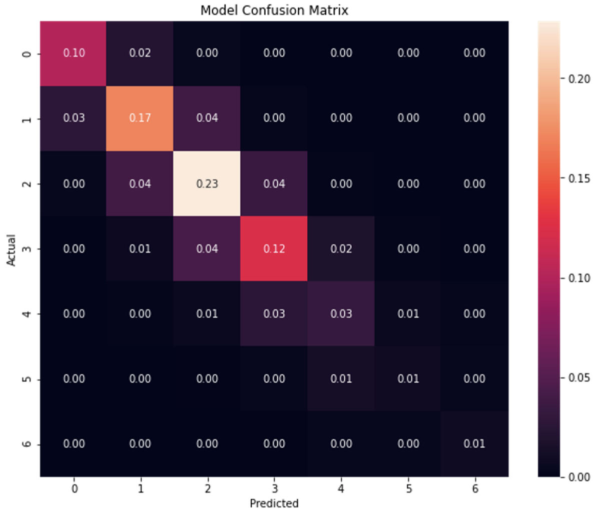

Figure 4-3: Best Model Confusion Matrix

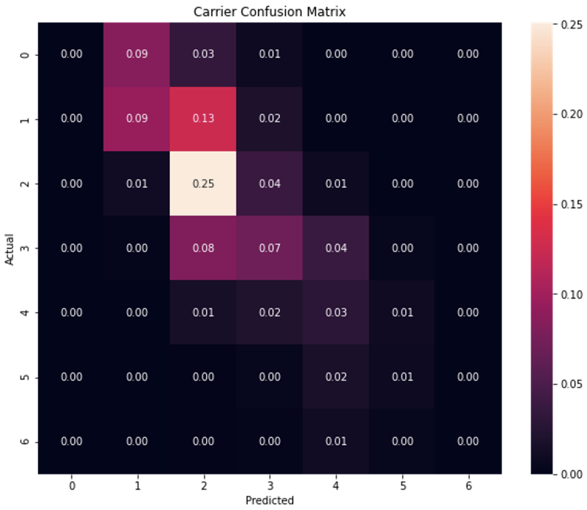

Figure 4-4: Carrier Provided Estimate Confusion Matrix

Viewing the carrier provided estimate confusion matrix and observing the smaller proportion of records on the diagonal, it is apparent that the carrier estimates have lower accuracy. It can also be seen that the carrier's errors are more unevenly distributed. The errors of predicting 2 days when the actual is 1 day, and predicting 1 day when the actual is zero days are the tree most common types of errors. The top two most-occurring types align with the hypothesis that carriers are being conservative in their estimates.

The final comparison of XGBoost and carrier provided estimates is conducted by creating cumulative density function plots for the absolute error of the predictions. These plots are shown in Figure 4-5 and Figure 4-6. The XGBoost plot shows that over 95% of records are accurately predicted or have an error of one day. Further, it shows that very few records have an error greater than two days. The carrier estimate plot shows that the percentage of records with error of one day or less is lower at 89%. This supports the point that the carrier's errors are on average of a larger magnitude than the XGBoost model's errors.

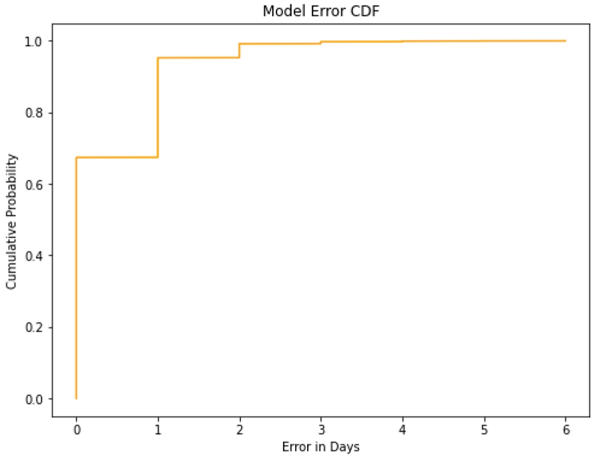

Figure 4-5: Best Model Error CDF

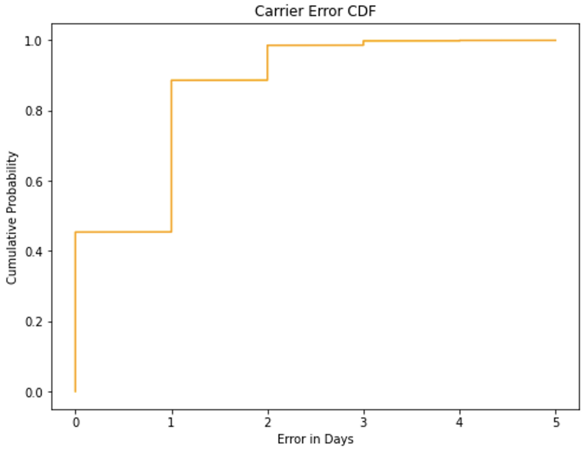

Figure 4-6: Carrier Provided Estimate Error CDF

4.1.3 Outperformance Analysis

An investigation is conducted to characterize where the XGBoost model outperforms carrier estimates. Charts displaying the percent change in accuracy across distribution centers, carriers, and shipping speeds are shown in Figures 4-7, 4-8 and 4-9, respectively. For convenience, the equation used to calculate percent change in accuracy is shown again below.

$$\text{Percent Change in Accuracy} = \frac{\text{Current Accuracy Score - Previous Accuracy Score}}{\text{Previous Accuracy Score}}$$

The XGBoost model estimates are more accurate than the carrier estimates on average across all distribution centers, all carriers, and all shipping speeds. In other words,

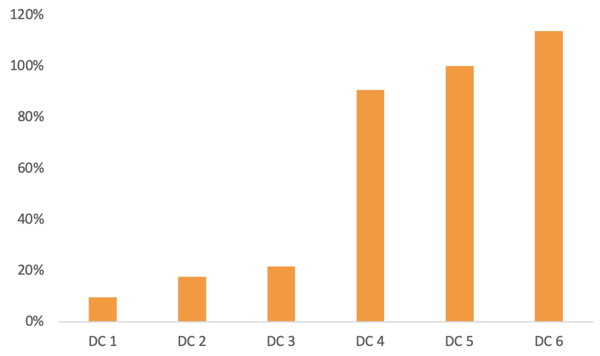

Figure 4-7: Percent Improvement in Accuracy by Distribution Center

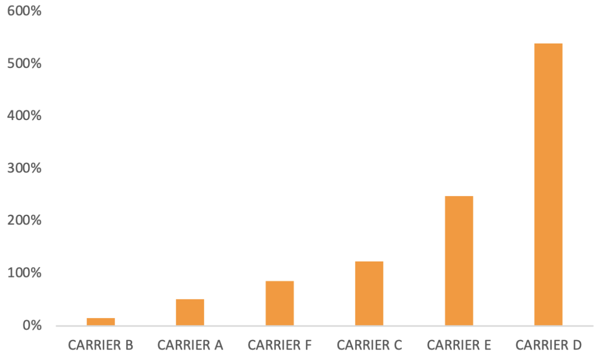

Figure 4-8: Percent Improvement in Accuracy by Carrier

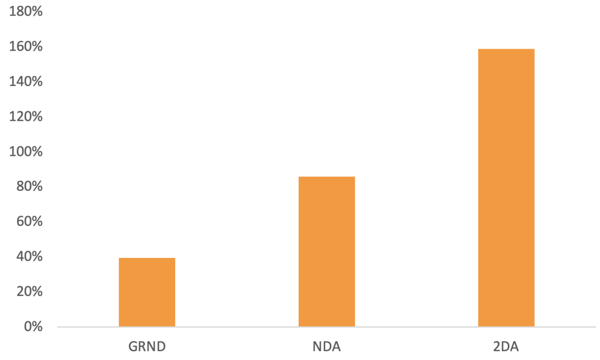

Figure 4-9: Percent Improvement in Accuracy by Shipping Speed

there is not one subset of data, based on the configuration attributes, where the carrier estimates outperform the XGBoost model estimates.

The XGBoost model significantly outperforms carrier estimates for three distribution centers in specific, with percent improvements in accuracy of over 90%. The same is true for three out of the six carriers analyzed, this time with percent improvements of over 100%. In terms of shipping speeds, the model especially outperforms the carrier provided estimates for two day air and next day air.

4.1.4 Feature Importances

To enhance model interpretability, feature importance scores are calculated using weight and gain metrics. The weight metric quantifies importance by counting the number of times the feature is used in a split. This metric favors high cardinality features. The gain metric quantifies importance by calculating the average performance gain across all splits where the feature is used. The top eight important features' scores are plotted in Figures 4-10 and 4-11 using the weight and gain metrics respectively.

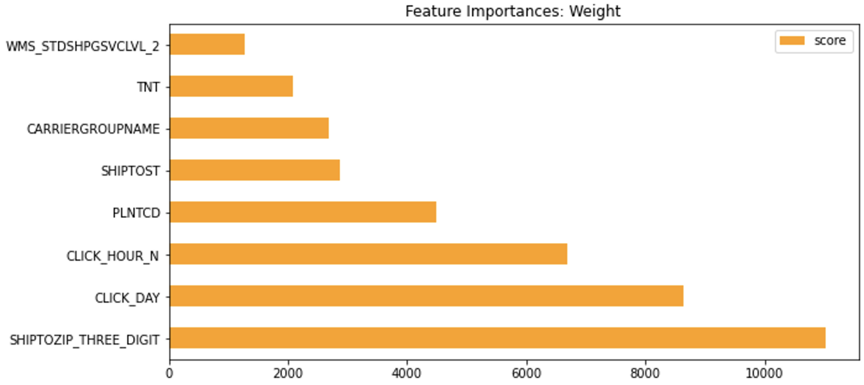

Figure 4-10: Best Model Feature Importance Scores: Weight Metric

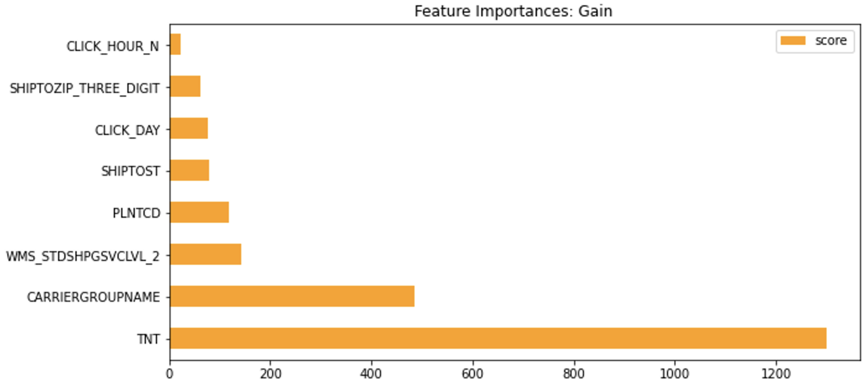

Figure 4-11: Best Model Feature Importance Scores: Gain Metric

The two top 8 sets consist of the same features but in somewhat reverse order. The weight metric set scores the destination zip code feature very high, which makes sense given its high cardinality. Destination zip code (SHIPTOZIP_THREE_DIGIT), day of week the order was placed (CLICK_DAY) and hour of day the order was placed (CLICK_HOUR) are the top three most important features in terms of weight scores. These features make up the bottom of the top eight set by the gain metric. Carrier provided transit time, carrier name and shipping speed are the top three most important features in terms of gain scores. Interestingly, again, these features make up the bottom of the top eight set by the weight metric.

4.2 Impact Assessment

An analysis is conducted to determine the impact of improved transit time estimates on fulfillment decision-making. Specifically, a counterfactual analysis is executed to compare the performance of the fulfillment optimization algorithm using carrier provided estimates to the performance using XGBoost model estimates. The performance using the XGBoost model estimates is determined using Victory's fulfillment simulation environment. This environment is a copy of the production fulfillment optimization system and enables simulation of the fulfillment outcomes that would have occurred if XGBoost model estimates were used in production historically. The performance of carrier provided estimates is known, and does not require simulation, given these were the estimates that were actually used in production.

The test of XGBoost estimates is conducted on a one month sample of historical data. The set of performance metrics assessed are based on what is used by Victory leadership to monitor fulfillment decision-making performance and consist of the following: average fulfillment cost in dollars, average CO2 emissions in tons, average lead time in days and percentage of orders shipped in multiple boxes. Note that in this context, lead time refers to the number of days between when a customer places their order and when they receive it.

Performance metrics for the set of fulfillment decisions made using the carrier provided transit times were calculated, as were those for the set of decisions based on the XGBoost model estimates. The XGBoost model was associated with a 4.5% decrease in lead time, a 3% reduction in CO2 emissions, a 1.5% reduction in cost and a 1% reduction in multiple box shipments.

Chapter 5

Recommendations

This chapter presents a set of recommendations for Victory's supply chain data science leadership. The primary recommendation is to build a production version of the XGBoost model and replace carrier provided estimates with XGBoost estimates in the fulfillment decision-making process. The XGBoost model generates more accurate estimates and enables improvement across all key fulfillment performance indicators.

The XGBoost model trains quickly compared to other models analyzed which is advantageous when production implementation is considered. Its training speed is also beneficial in the context of computational resources, as a quicker training time is associated with lower cost and a smaller carbon footprint [10].

The XGBoost model also has a relatively small feature set. This reduces the complexity and time associated with creating and running production model pipelines. It also promotes interpretability which will aid in establishing the stakeholder support necessary for production release.

Conversations with software engineering regarding the development of a production model have been initiated as part of this work. Effort should be allocated toward continuing these conversations. Specifics around data preparation pipelines and automated model training and estimate generation processes should be discussed. Monitoring and alerting to ensure software systems are running as expected and model performance is in line with historical behavior should also be a topic of conversation. Finally, a plan regarding integration with the production fulfillment optimization software, to enable automated transit time estimate consumption, should be discussed.

Victory's supply chain data science leaders should also consider first releasing the XGBoost model on a subset of live orders to confirm expected performance. After this pilot, the model should be released across all U.S. digital orders. This strategy is advantageous because it increases the likelihood of receiving sufficient stakeholder support; gaining organizational buy-in for a pilot is much easier than for a full scale release. The strategy also increases the likelihood that stakeholders will support a full scale release, given trust will be established in the pilot phase. Finally, releasing via a pilot strategy enables valuable insights to be gained and improvements to be made prior to releasing across all orders.

An alternative next step is to continue experimentation in hopes of extracting additional model performance. This course of action is not recommended. The model already significantly outperforms the current state. Steps towards realizing that value should be taken, hence, the recommendation to build a production model. Continuing to invest in performance gains will only delay value creation. Implementing the model in production will also enable valuable lessons to be learned earlier on. The feedback loop necessary for model enhancements will be initiated, unlocking a better model sooner.

Chapter 6

Conclusion and Future Work

6.1 Thesis Summary

In this thesis, a model to predict transit times is developed. Considerable experimentation is conducted to determine favorable model design elements, across model structure, parameters and hyperparameters. The final model's estimates are significantly more accurate than the estimates used currently in Victory's fulfillment decision-making processes.

In this thesis, an impact assessment is conducted to determine the effect of improved transit estimates on key fulfillment performance indicators. Using the predictive model estimates instead of the current ones in the fulfillment optimization algorithm was associated with a 4.5% decrease in lead time, a 3% reduction in CO2 emissions, a 1.5% reduction in cost and a 1% reduction in multiple box shipments on a historical one month sample.

These results support the hypothesis that improving transit time estimates can improve fulfillment decision-making. Furthermore, the results demonstrate that improving the digital consumer experience while reducing carbon emissions is possible, especially through the application of modern machine learning techniques. This thesis provides a recommendation to implement the predictive model in production to realize the expected, sizeable benefits.

6.2 Lessons Learned

This thesis conveys many lessons learned that are value to the Victory organization and others managing supply chains for digital channels. This thesis demonstrates how many different methods can be used to predict transit time under an origin-destination formulation. Classification and regression methods can both be used, and within those, various model types can be applied. This thesis shows how model type and feature list selection are among the most influential aspects of model development. It also shows how careful selection of data set time periods, processing of outliers, and implementation of sampling methodologies can meaningfully impact model performance.

While model development can become complex quickly, this thesis shows how interpretability and simplicity can be maintained by using a relatively small feature set. This thesis also demonstrates that a relatively small feature set can enable the generation of high quality transit time estimates. This is an important lesson, given data sourcing, data cleaning and feature engineering are usually time intensive and significant barriers to model development. By using a small feature set, these barriers can be reduced and the likelihood of creating business value through machine learning can be increased.

This thesis also conveys an important lesson around how the best machine learning model should be selected. It shows how considering aspects like interpretability and model train time, alongside accuracy, can result in a final model that better meets the needs of an organization. Further, the thesis addresses how using a holistic selection process like this can increase the likelihood that stakeholders understand and support the machine learning work, and thus increase the likelihood that it is implemented in production.

Finally, this thesis demonstrates the reality of computational resource limitations, and shows how machine learning practitioners can design their experimentation given such constraints. This work highlights the reality that decisions must be made related to which experiments to run sequentially versus which to run simultaneously. In an ideal state, all experiments could be run simultaneously, but given computational resource limitations, this usually is not possible. Considering which experiments are likely to be most influential, and which ones are likely to have dependencies on others, can assist machine learning practitioners in deciding which ones to run simultaneously. Careful consideration like this supports the development of highly performant machine learning models in the face of computational resource limitations.

6.3 Future Work

In the future, additional work can be done to further improve transit time estimates. There are three main areas of future work: geographic expansion, data preparation, and model experimentation.

Geographic Expansion
In addition to the United States market, Victory operates in European and Asian markets. The work presented in this thesis is limited to the United States. Extension of the model into European and Asian businesses should be investigated. The current model may perform well in these new regions, or the model may need to be tweaked to reflect unique attributes associated with those geographic areas. Exploratory data analysis, feature engineering and model experimentation should be revisited in this

case. Victory should seek to realize the benefit of improving transit estimates globally rather than just in the United States market.

Data Preparation

In the future, enhancements to the data preparation portion of model development should occur. Investment in an automated data cleaning process should be made. Open source cleaning packages should be researched and leveraged if possible. Further, data imputation processes should be developed, evaluated and implemented. Currently, data observations with missing values are deleted. This increases the risk of bias in the analytical data set and can be avoided by implementing a robust data imputation process. The final data related enhancement opportunity is to continue iterating on lagged first-scan features. Different metrics, definitions of similar orders and time periods should be investigated with the goal of creating features that emulate the importance of first-scan features in post-induction models.

Model Experimentation

The final area of future work is related to model experimentation. Additional investigation of separate models per carrier should occur. In the future, separate models with the same number of training observations as the consolidated model should be tested. This methodology will enable better isolation of the impact associated with the model structure change.

Additional computational resources and an enhanced model development environment should be pursued in the future. These would enable more robust experimentation through the ability to run models faster and execute simultaneous experimentation. A subset of design elements, ideally those considered to be most influential in model performance, should be tested simultaneously through a grid search technique. This approach is more exhaustive than running experiments sequentially, provides insight into the best combination of design elements and could likely unlock additional performance improvements.

The final area of future work related to model experimentation is to investigate more advanced model types. Specifically, deep learning and neural network methods should be implemented and their performance should be compared to existing model types. When making this comparison, the complex tuning processes, high data volume requirements, and large computational burdens associated with these techniques should be considered [2].

THIS PAGE INTENTIONALLY LEFT BLANK

Bibliography

[1] Asad Abdi and Chintan Amrit. "A review of travel and arrival-time prediction methods on road networks: classification, challenges and opportunities". In: *PeerJ Computer Science* 7 (Sept. 8, 2021), e689. ISSN: 2376-5992. DOI: 10.7717/peerj-cs.689. URL: https://peerj.com/articles/cs-689 (visited on 01/24/2023).

[2] Laith Alzubaidi et al. "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions". In: *Journal of Big Data* 8.1 (Mar. 31, 2021), p. 53. ISSN: 2196-1115. DOI: 10.1186/s40537-021-00444-8. URL: https://doi.org/10.1186/s40537-021-00444-8 (visited on 01/24/2023).

[3] Leo Breiman. "Random Forests". In: *Machine Learning* 45.1 (Oct. 1, 2001), pp. 5–32. ISSN: 1573-0565. DOI: 10.1023/A:1010933404324. URL: https://doi.org/10.1023/A:1010933404324 (visited on 01/24/2023).

[4] Tianqi Chen and Carlos Guestrin. "XGBoost: A Scalable Tree Boosting System". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* Aug. 13, 2016, pp. 785–794. DOI: 10.1145/2939672.2939785. arXiv: 1603.02754[cs]. URL: http://arxiv.org/abs/1603.02754 (visited on 01/24/2023).

[5] Anna Veronika Dorogush, Vasily Ershov, and Andrey Gulin. "CatBoost: gradient boosting with categorical features support". In: ().

[6] *FY21 NIKE, Inc. Impact Report — NIKE, Inc.* URL: https://about.nike.com/en/newsroom/reports/fy21-nike-inc-impact-report-2 (visited on 01/24/2023).

[7] Arthur E. Hoerl and Robert W. Kennard. "Ridge Regression: Biased Estimation for Nonorthogonal Problems". In: *Technometrics* 12.1 (Feb. 1, 1970), pp. 55–67. ISSN: 0040-1706. DOI: 10.1080/00401706.1970.10488634.

[8] Jihed Khiari and Cristina Olaverri-Monreal. "Boosting Algorithms for Delivery Time Prediction in Transportation Logistics". In: *2020 International Conference on Data Mining Workshops (ICDMW).* Nov. 2020, pp. 251–258. DOI: 10.1109/ICDMW51313.2020.00043. arXiv: 2009.11598[cs,stat]. URL: http://arxiv.org/abs/2009.11598 (visited on 01/24/2023).

[9] Lajos Kisgyörgy and Laurence R. Rilett. "Travel Time Prediction by Advanced Neural Network". In: *Periodica Polytechnica Civil Engineering* 46.1 (2002). Number: 1, pp. 15–32. ISSN: 1587-3773. URL: https://pp.bme.hu/ci/article/view/617 (visited on 01/24/2023).

[10] Alexandre Lacoste et al. *Quantifying the Carbon Emissions of Machine Learning.* Nov. 4, 2019. DOI: 10.48550/arXiv.1910.09700. arXiv: 1910.09700[cs]. URL: http://arxiv.org/abs/1910.09700 (visited on 01/24/2023).

[11] Xia Li and Ruibin Bai. "Freight Vehicle Travel Time Prediction Using Gradient Boosting Regression Tree". In: *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA).* 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA). Anaheim, CA, USA: IEEE, Dec. 2016, pp. 1010–1015. ISBN: 978-1-5090-6167-9. DOI: 10.1109/ICMLA.2016.0182. URL: http://ieeexplore.ieee.org/document/7838286/ (visited on 01/24/2023).

[12] Hans-O Portner et al. "Summary for Policymakers: Climate Change 2022: Impacts, Adaptation, and Vulnerability. Contribution of Working Group II to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change". In: (Aug. 3, 2022). DOI: 10.1017/9781009325844.001.

[13] Nikolaos Servos et al. "Travel Time Prediction in a Multimodal Freight Transport Relation Using Machine Learning Algorithms". In: *Logistics* 4.1 (Mar. 2020). Number: 1 Publisher: Multidisciplinary Digital Publishing Institute, p. 1. ISSN: 2305-6290. DOI: 10.3390/logistics4010001. URL: https://www.mdpi.com/2305-6290/4/1/1 (visited on 01/24/2023).

[14] Axel Simroth and Henryk Zähle. "Travel Time Prediction Using Floating Car Data Applied to Logistics Planning". In: *IEEE Transactions on Intelligent Transportation Systems* 12.1 (Mar. 2011). Conference Name: IEEE Transactions on Intelligent Transportation Systems, pp. 243–253. ISSN: 1558-0016. DOI: 10.1109/TITS.2010.2090521.

[15] *Sporting goods 2022: The new normal is here.* McKinsey. URL: https://www.mckinsey.com/industries/retail/our-insights/sporting-goods-2022-the-new-normal-is-here (visited on 01/24/2023).

[16] *Sustainable fashion - A survey on global perspectives.* KPMG, 2019.

[17] Alfateh M. Tag Elsir et al. "JSTC: Travel Time Prediction with a Joint Spatial-Temporal Correlation Mechanism". In: *Journal of Advanced Transportation* 2022 (May 23, 2022). Publisher: Hindawi, e1213221. ISSN: 0197-6729. DOI: 10.1155/2022/1213221. URL: https://www.hindawi.com/journals/jat/2022/1213221/ (visited on 01/24/2023).

[18] Yanru Zhang and Ali Haghani. "A gradient boosting method to improve travel time prediction". In: *Transportation Research Part C: Emerging Technologies.* Big Data in Transportation and Traffic Engineering 58 (Sept. 1, 2015), pp. 308–324. ISSN: 0968-090X. DOI: 10.1016/j.trc.2015.02.019. URL: https://www.sciencedirect.com/science/article/pii/S0968090X15000741 (visited on 01/24/2023).

[19] Hui Zou and Trevor Hastie. “Regularization and Variable Selection via the Elastic Net”. In: *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* 67.2 (2005). Publisher: [Royal Statistical Society, Wiley], pp. 301–320. ISSN: 1369-7412. URL: `https://www.jstor.org/stable/3647580` (visited on 01/24/2023).