Walmart Trip Type Multi-Classification

Michelle Hu
Carnegie Mellon University
mhsu1@andrew.cmu.edu

Ya Ting Chang
Carnegie Mellon University
yatingc2@andrew.cmu.edu

1 INTRODUCTION

The purpose of the project is to predict trip types for each visit for Walmart. To improve customer's shopping experience, what Walmart has already done is segmenting their store visits into different trip types. Whether the customers are purchasing stuff for parties or making their way through a weekly grocery list, Walmart wants to target them with different marketing strategies. Therefore, accurately predicting trip types with machine learning algorithm is extremely important for Walmart. Currently, Walmart's customer trip type is classified into 38 categories. We want to employ multiclass classification to solve the problem of classifying each visit into one single type.

2 DATA PREPARATION

The dataset is from Kaggle and contains 1,300,700 rows and 7 columns. Each row is an item purchased corresponding to a single trip by a single customer. The features are TripType, Upc, ScanCount, DepartmentDescription, FinelineNumber, VisitNumber, and Weekday.

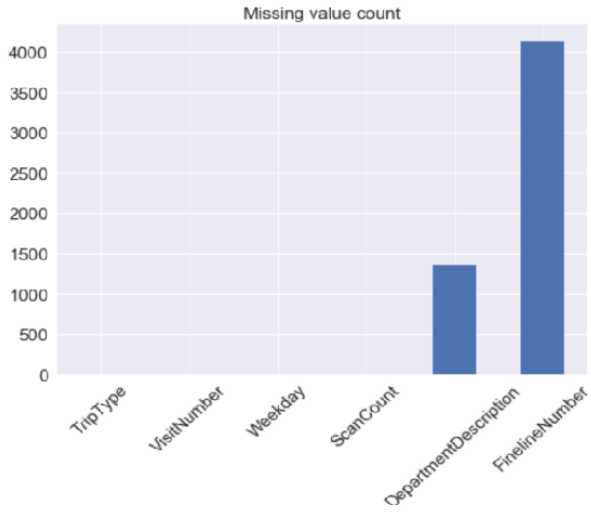
Feature	Description
DepartmentDescription	Description of the product's department
FinelineNumber	A refined category for the product purchased
ScanCount	The number of the given item that was purchased. A negative value indicates a product returned
TripType	The 38 original trip types.
Upc	The Universal Product Code of the product purchased
VisitNumber	A unique id corresponding to a single trip by a single customer
Weekday	The weekday of the customer visit

We selected 5 columns including ScanCount, DepartmentDescription, FinelineNumber, VisitNumber, Weekday as our primary features. We didn't include Upc as the Upc we were provided with was not complete and accessing to Walmart API to extract product information based on Upc is not available. Our data preparation includes:

- Missing values imputation
- Categorize some features
- One Hot Encoding for each feature

2.1 Missing Values Imputation

All missing values are in DepartmentDescription and FinelineNumber. We found that lots of missing value in FinelineNumber are from Pharmacy department. Instead of dropping these instances, we created a new FinelineNumber – 9999 for them.



2.2 Feature Categorization

Since the range of ScanCount is large, we decided to category this feature into three categories based on the number of the ScanCount

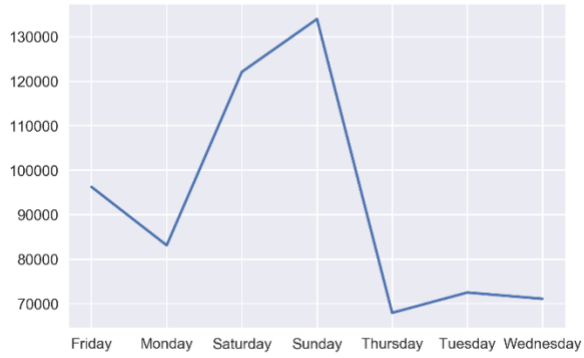
ScanCount	Category
< 0	return
< 5	few
< 15	medium
others	many

2.3 One Hot Encoding for each feature

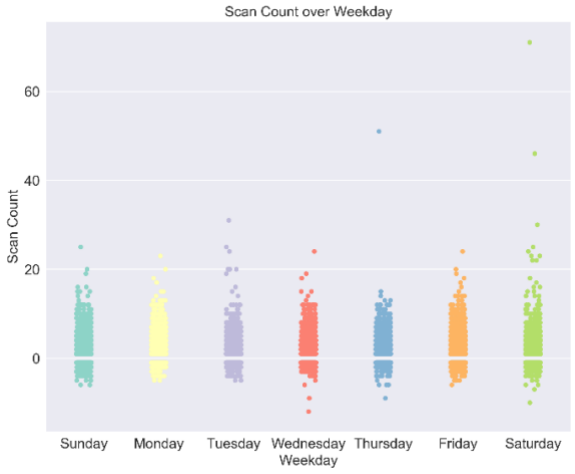
We spread the FinelineNumber, Weekday, and DepartmentDescription, ScanCount into columns, turning our data frame into a large sparse matrix. The shape of our matrix now becomes: 95,516 rows and 5,275 columns.

3 DATA VISUALIZATION

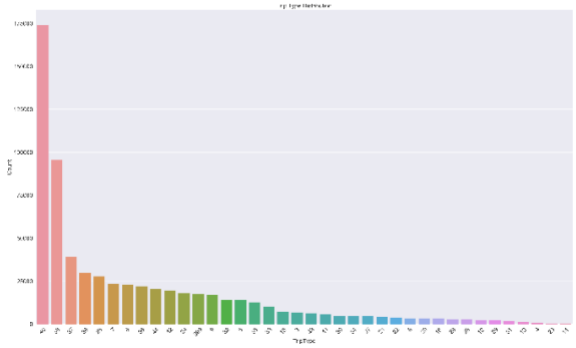
We first want to know visit frequency over weekdays. Most customers come to Walmart during the weekend.



However, the number of product purchased is not different across the weekdays. Besides, from the boxplot we could see that the range on Saturday is larger than other days. (scan count smaller than 0 means product returned).



We also want to know our outcome (trip type)'s distribution. And found that the data set is highly imbalanced with some trip types accounting for most of our outcomes.



4 METHODOLOGY

Our dataset is comprised of 38,206 instances with 5,257 dimensions, which is a 40% subset of the whole dataset due to computation limitation. Each row represents a unique customer visit characterized by weekday, FinelineNumber, ScanCount, and DepartmentDescription. The project goal is to produce classifiers that can precisely distinguish the 38 customer trip types with higher accuracy. In the machine learning setting, the business problem can be transformed into a multiclass classification task. To tackle the problem, we used 3 supervised learning methods to approach the problem, which are Logistic Regression, Support Vector Machine, and Random Forest. All these algorithms can deal with multiclass classification task. The underlying reasons for choosing these methods is to explore the possibility of the dataset. For instance, Logistic Regression serves as a good starting point for training a model with its simple implementation and interpretability of the coefficients,

which can be interpreted as exponentiated odds ratios with respect to the outcome. However, Logistic Regression assumes linear decision boundary. Unless we transform the features, the non-linearity cannot be captured.

As for Support Vector Machine (SVM), the method uses only a subset of the samples to generate decision function, which is memory efficient. Embedded with various kernel functions, SVM can preserve the similarity between data points without additional cost for learning in high dimensional space. Hence, SVM also supports data that is not linearly separable. The drawbacks of SVM lie in its long training time and the difficulty in selecting the proper kernel.

Regarding the tree-based technique, we first started at Decision Tree to explore the dataset in terms of the decision path and the nodes used to split. Then, we expanded our model to Random Forest to leverage the power of ensemble method, combining a group of weak learners into a strong learner. By randomly selecting a subset of features to build each single tree, Random Forest is a collection of de-correlated trees, which can improve variance reduction without sacrificing its low-bias nature. In terms of the model baseline, we implemented Naïve Bayes, which holds a strong assumption on the independent relationship among covariates.

Before training the model, we first addressed the issue of imbalanced data among different customer trip types. We used the stratified sampling to split the training and test sets to ensure the occurrence of every trip type during the training and testing phase. After the partition of dataset into train set (30,564 rows) and test set (7,642 rows), we proceeded to the model training process.

First, we adopted Logistic Regression, which does not require linear relationship between the outcome and the predictors, yet the decision boundary is assumed linear. While training the model, we used 5-fold cross validation and tuned the hyperparameter "C", which controls the regularization strength of our model. The regularization method used in our model is Lasso regression, which will penalize and zero out unnecessary covariates and thus enhance the model performance.

Second, we trained SVM also with 5-fold cross validation, tuning on the hyperparameter "C", which represents the penalty of the error term while training the model. Specifically, we implemented Linear Support Vector Classification, which takes advantage of large-scale dataset with support for multiclass classification task using "one-vs-the-rest" method.

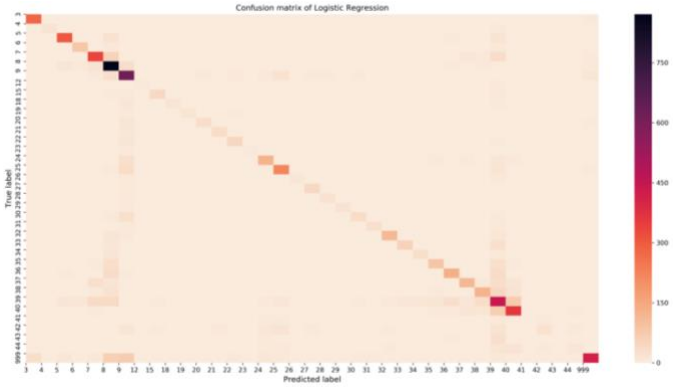
Finally, we used the tree-based method including Decision Tree and Random Forest. For Decision Tree, we mainly wanted to utilize its advantages of interpretability and visualization. Therefore, we didn't enforce any constraint during the training phase and let the tree fully grow. After the exploration on the decision path, we found out that DepartmentDescription and certain FinelineNumber play a very important role while growing the tree. However, since we didn't have access to the detailed information regarding the FinelineNumber, the interpretation on these essential FinelineNumber is limited. Regarding the Random Forest, we used the 5-fold cross validation while tuning the number of trees and the maximum depth of the model.

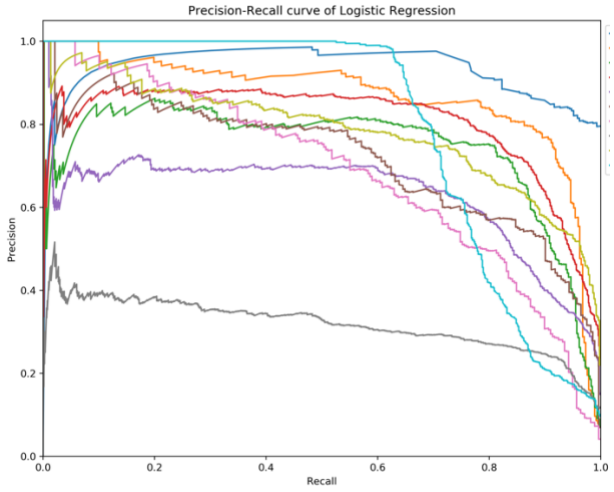
5 EVALUATION AND RESULTS

To assess our models, we adopted different metrics such as Precision-Recall curve and Confusion Matrix to strengthen the evaluation process.

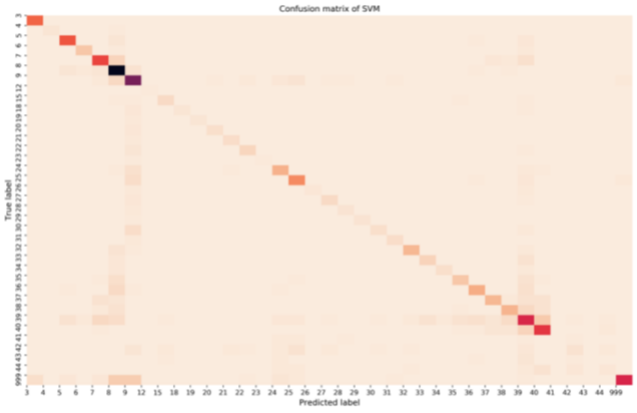
In our evaluation, we intentionally excluded the ROC (Receiver Operating Characteristic) curve due to the data imbalance among our outcome variables (TripType). Suppose we transform our multiclass classification problem into 38 binary classification tasks, the positive class in the classifying setting is relatively small compared to the negative class. Therefore, using ROC curve to evaluate the result will be misleading since the False Positive Rate will decline very slowly and yield seemingly perfect ROC curve with AUC (Aura Under Curve) approximating to 1. Hence, ROC curve is not included in our analysis of the result.

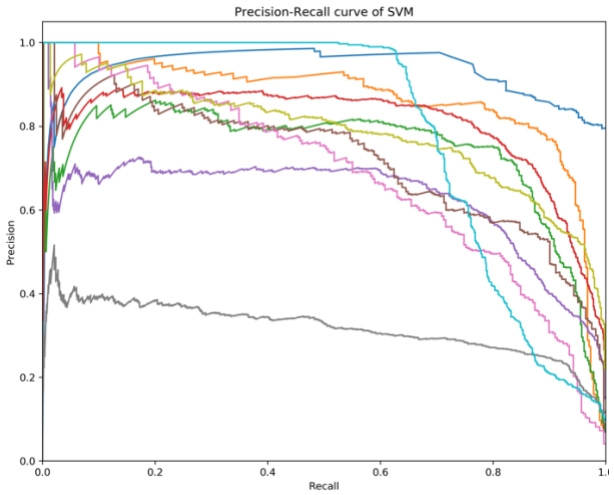
For Logistic Regression, we first looked at the confusion matrix. From the graph below, we can see that the classifier performed relatively well on certain trip type such as 8 and 9, indicating by the darker color in the confusion matrix. In the Precision-Recall (PR) curve, we only extracted the top 10 trip type in terms of frequency to do the visualization. From the PR curve, we can clearly spot the issue of the imbalanced data, which is revealed by the quickly declining PR curve due to large Recall. In addition, the F1-score is 0.6589 and the accuracy is 0.6710.



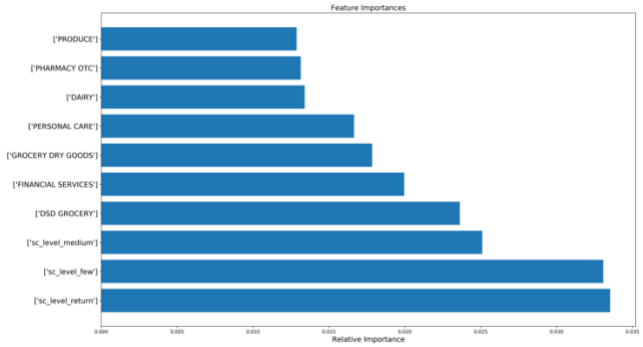


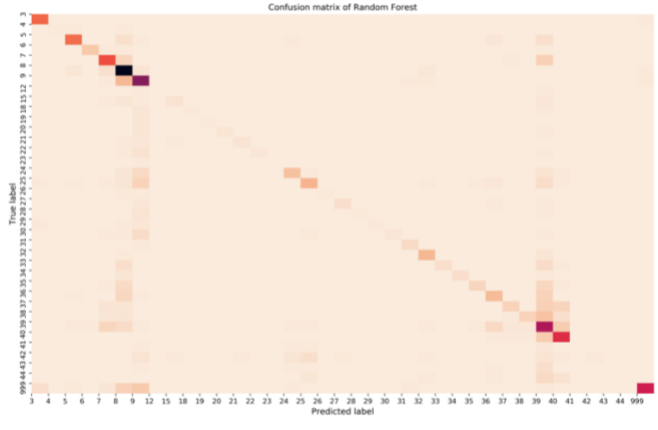
For Support Vector Machine, the confusion matrix yielded similar result with Logistic Regression, classifying certain trip type well. The F1-score is 0.6566 and the accuracy is 0.6705. As for PR curve, similar interpretation can be applied to SVM. The recall increased very quickly, causing the PR curve to decline.

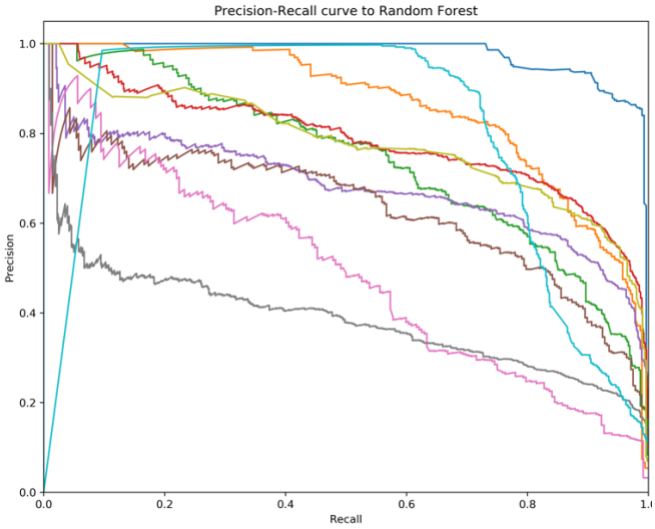




As for Random Forest, we can see from the feature variance plot, the top-10 feature includes different ScanCount levels and certain DepartmentDescription. The interpretation of the PR curve is similar to the other model with quickly downward sloping PR curve due to imbalance data. The F1-score and the accuracy is 0.582 and 0.611 respectively.







From the comparison table, we can see that Logistic Regression and SVM outperformed Random Forest and our baseline Naïve Bayes. Both the accuracy and F1-score of Logistic Regression and SVM are higher than the other two models.

	Accuracy	F1-score
Naïve Bayes	0.6154	0.6010
Logistic Regression	0.6710	0.6589
SVM	0.6705	0.6566
Random Forest	0.6110	0.5820

Regarding the limitations of the project, we cannot properly interpret the FinelineNumber and TripType owing to the lack of detailed information of the dataset. Second, the dataset is imbalanced data in terms of multiclass, which is not easy to handle compared to binary classification task. For future improvement, we will investigate on the reason for underperformance of Random Forest. Also, we will try other machine learning techniques such as Adaboost and Neural Network to see whether we can further boost the model performance.

REFERENCES

[1] Assumptions of Logistic Regression. (n.d.). Retrieved from http://www.statisticssolutions.com/assumptions-of-logistic-regression/

[2] Logistic Regression vs Decision Trees vs SVM: Part II. (n.d.). Retrieved from https://www.edvancer.in/logistic-regression-vs-decision-trees-vs-svm-part2/

[3] Logistic Regression vs Decision Trees vs SVM: Part II. (n.d.). Retrieved from https://www.edvancer.in/logistic-regression-vs-decision-trees-vs-svm-part2/

[4] Oza, N. C., & Tumer, K. (2008). Classifier ensembles: Select real-world applications. Information Fusion, 9(1), 4-20. doi:10.1016/j.inffus.2007.07.002

[5] (n.d.). Retrieved April 10, 2018, from https://www.kaggle.com/c/walmart-recruiting-trip- type-classification/discussion/18158

A PROJECT PIPELINE