Market Basket Prediction (Walmart Kaggle Competition)

Daniel Diaz, Di Zhu, Vijay Sarathy

BIA 656

12/11/2015

**Business Understanding**

Walmart uses both art and science to continually make progress on their core mission of better understanding and serving their customers. One way Walmart is able to improve customers' shopping experiences is by segmenting their store visits into different trip types. In this competition Walmart seeks to understand which products are purchased together. Usually when a customer goes to shop at Walmart, he or she goes for a particular purpose, i.e. weekly grocery shopping, birthday gifts, emergency night shopping for the baby. Knowing when an individual customer is most likely to shop for a particular set of products demand and which are often purchased together is immensely valuable to predict and service increased customer demand. If Walmart is not able to predict its’ customers shopping patterns and behaviors they will miscalculate customer demand and fail to supply the desired products at the right time, losing a substantial amount of potential revenue.

Understanding which items are most often purchased together walmart could increase product stock and optimize product placement within the store. As the largest retailer in the world by revenue, with one of the largest databases of customer preference on the planet, making sense of all the data is very difficult. Customer transactions detailing millions of products all from different departments being purchased by millions of customers are stored, but underlying this is can be extracted an understanding of consumer preference, and demand for particular sets of products perhaps varying by location.

Therefore, categorizing a collection of purchases from diverse departments into one of thirty-eight categories makes it easier for Walmart to analyze customer behavior. Future customers can then be assigned to a set of categories. By categorizing a collection of purchases as one, Walmart can more efficiently segment its customers by targeting them with a specific set of products based on their previous purchases and people that share similar purchase patterns. Store locations can also optimize their product offering based on the profile of the area’s customers and the customer categories they fall into most frequently.

**Data Understanding**

The data was obtained from: <https://www.kaggle.com/c/walmart-recruiting-trip-type-classification/data>

It has the following characteristics (Table 1):

* Number of instances: 647055
* Number of attributes: 7
* Target: TripType

Data:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TripType | VisitNumber | Weekday | Upc | ScanCount | DepartmentDescription | Fineline  Number |
| 39 | 26 | Friday | 89470001034 | 2 | DAIRY | 1450 |
| 39 | 26 | Friday | 84943400209 | 1 | COOK AND DINE | 2072 |
| 39 | 26 | Friday | 3663200828 | 1 | FROZEN FOODS | 1850 |
| 39 | 26 | Friday | 81829001273 | 3 | DAIRY | 1450 |
| 39 | 26 | Friday | 89470001006 | 1 | DAIRY | 1450 |
| 39 | 26 | Friday | 79108330121 | 1 | PHARMACY OTC | 535 |

Feature Description:

|  |  |  |
| --- | --- | --- |
| Variable | Type | Description |
| TripType | Char | a categorical id  representing the type of shopping trip the customer made |
| VisitNumber | ID | an id corresponding to a  single trip by a single customer |
| Weekday | Date | the weekday of the trip |
| UPC | Char | the UPC number of the  product purchased |
| ScanCount | Int | the number of the given  item that was purchased. A negative value indicates a product return |
| DepartmentDescription | Char | a high-level description of  the item's department |
| FinelineNumber | Char | a more refined category for  each of the products, created by Walmart |

Table 1: Characteristics of the data

Target Variable TripType has the following attributes (Figure 1):

* Consists of 38 categories
* 80% of the observations fall in 15 categories, 90% fall in 22, 95% fall in 28

The objective is to produce one target output, the trip type,that categorizes a collection of items, purchased during one trip, from diverse departments such as dairy, financials, pharmacy, personal care, into just one number that represents one category. This target output is constructed by Walmart with more features than shown above and shared.

**Data Preparation**

The original data set was organized by item transaction so there may be multiple rows for one trip . To mitigate this problem, pandas pivot tables were used to group the items into buckets that represent each trip (Table 2). Therefore, a TripType can be used by the model to predict a corresponding value based on the characteristics of each basket.

**Model**

*Classification Algorithms and Ensemble Methods*

The labelled data set was split into two components: training (70% of data) and testing (30% of data) sets.

The benchmark is logistic regression, with 20% cross accuracy.

Models [1]

* K-Nearest Neighbors

The principle behind nearest neighbor methods is to find a predefined number of training samples closest in distance to the new point, and predict the label from these. The number of samples can be a user-defined constant (k-nearest neighbor learning), or vary based on the local density of points (radius-based neighbor learning). The distance can, in general, be any metric measure: standard Euclidean distance is the most common choice. Neighbors-based methods are known as *non-generalizing* machine learning methods, since they simply “remember” all of its training data.

*Rationale for using*:

* The cost of the learning process is zero.
* No assumptions about the characteristics of the concepts to learn have to be done.
* CART

Classification and Regression Trees ([CART](http://en.wikipedia.org/wiki/Predictive_analytics#Classification_and_regression_trees)) is very similar to C4.5, but it differs in that it supports numerical target variables (regression) and does not compute rule sets. CART constructs binary trees using the feature and threshold that yield the largest information gain at each node.

*Rationale:*

* Simple to understand and interpret. People are able to understand decision tree models after a brief explanation.
* Requires little data preparation. Other techniques often require data normalization, [dummy variables](https://en.wikipedia.org/wiki/Dummy_variable_(statistics)) need to be created and blank values to be removed.
* Able to handle both numerical and [categorical](https://en.wikipedia.org/wiki/Categorical_variable) data. Other techniques are usually specialized in analysing datasets that have only one type of variable.

Ensemble methods [2]:

* Random Forest

A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and use averaging to improve the predictive accuracy and control over-fitting. The sub-sample size is always the same as the original input sample size but the samples are drawn with replacement if there exists bootstrap.

*Rationale:*

* It gives estimates of what variables are important in the classification.
* It has methods for balancing error in class population unbalanced data sets.
* Bagging(added to the first two models)

Involves having each model in the ensemble vote with equal weight. In order to promote model variance, bagging trains each model in the ensemble using a randomly drawn subset of the training set.

*Rationale:*

* As we have fewer than 100,000 trips, Bagging alleviates this with replications of the data we have.
* Bagging removes the variance while leaving bias unchanged.
* Gradient Boosting:

Gradient Boosting builds an additive model in a forward stage-wise fashion; it allows for the optimization of arbitrary differentiable loss functions.

*Rationale for using:*

* It’s slow to train, however, we don’t have time limitation.
* It’s proper for features measured on different scale.

*Multi class Classification Method* [3]

* One vs Rest

The strategy consists in fitting one classifier per class. For each classifier, the class is fitted against all the other classes. In addition to its computational efficiency (only n\_classes classifiers are needed), one advantage of this approach is its interpretability. Since each class is represented by one and one classifier only, it is possible to gain knowledge about the class by inspecting its corresponding classifier.

* One vs One

At prediction time, the class which received the most votes is selected. In the event of a tie (among two classes with an equal number of votes), it selects the class with the highest aggregate classification confidence by summing over the pair-wise classification confidence levels computed by the underlying binary classifiers.

*Voting Method*

Voting classifier implementation combines conceptually different machine learning classifiers and uses a majority vote or the average predicted probabilities (soft vote) to predict the class labels (Figure 2). Such a classifier can be useful for a set of equally well performing models in order to balance out their individual weaknesses.

**Results**

The results (accuracy scores near 50%) indicate that overall there is a 50% chance that a collection of purchases made during one trip are correctly classified into one of Walmart’s 38 preconstructed market basket categories. Meaning that the most relevant coupons are sent to the customers roughly 50% of the time(Figure 4). The feature importances order is: Department Description > Scan Count > Weekday > Count (Figure 5). Neither one vs the rest nor one vs one are not better than the default multi class methods of each algorithms. From a data and business standpoint, the difference in categories of trip type must be investigated further to answer two main questions:

(1) Are trip type categories similar to each other (i.e. are there clusters?);

(2) What is the risk of a misclassification? If the wrong coupon is sent to the wrong customer, how different is one category from another?

The ramifications of the wrong coupon to the wrong customer needs to be analyzed; it could either dissuade the customer from revisiting the store.One way that Walmart may protect itself is by disguising its hypertargeted marketing strategy by sending a collection of relevant and irrelevant coupons: baby diapers, mixed in with irrelevant ones: fishing rods.

Alternatively, if the customer is part of a diverse social network he or she can learn about promotions of items that may be of interest from other customers.

As our model methodology is very robust (the voting method of random forest and bagging of KNN, CART), the explanation of the 50% accuracy can be attributed to the challenge of using 5 features to classify 38 possible categories. An additional feature may be able to be created such as looking at the 2nd most popular department; however, this requires extensive effort to code and the net improvement may only be relatively small. Further it is also challenging to cut down the number of categories to predict as the frequency distribution, although not uniform, is not skewed to an extent that certain trips can be neglected.

For future classification, collecting data by customer ID can help improve the accuracy of trip type especially if a customer buys similar products at a steady frequency. However, for this to be meaningful the customer information must be tracked by an online account. Perhaps a promotion encouraging the use of such a service can mitigate this concern in future data collection.

*Deployment*

The decision support system interface is an iPython script with the option to input the attributes of one trip. The interface is easy-to-use.

With these considerations mentioned in “Results” in mind and the fact that some trip types are more accurately classified than others (Table 3), coupons should be initially sent to the following trip types: 3,5,6,8,32,40. Their f1 scores are significantly greater than 50, therefore reducing the risk of misclassification as well as providing the following benefits:

1. Send the right coupons to the right customers
2. Organize the store sections to accommodate frequent item combinations
3. Improved in-store service based on looking at a customer’s purchase history

**Business Recommendations**

The following are recommendations for future data collection.

* Reduce the amount of the classes

Data Science is a team work. The amount of trip type categories might be determined by the marketing team according to the context of Walmart. However, it is pretty hard to get a high accuracy among 38 categories. We suggest the Data Science Team and the Marketing team should interact together and try to reduce the number of labels.

* Track items and trips by customer ID to profile different customers
* Include time in data to track any changes in purchasing habits

Not only will these facilitate classification, but customer service can significantly improve with the increase in information thereby resulting in more tailored recommendations.

**Future Work**

The addition of more features is being consider to try to improve accuracy (please see Appendix B). This can be accomplished by expanding the Department Description to a tabulation of item scan counts under each different department for each visit. This results in over a ten size increase of the data set- 70 columns total. This large increase requires greater resources to be used: time spent running the algorithms, and more CPU, memory devoted. With even a larger initial dataset, this effect will be further magnified. We are currently seeing an accuracy of 0.6036 using the Voting method that we previously used for 6 features. Also, TripType 999 has a high f1 score: 0.78.

Kaggle Submission:

With 6 features, our current rank is 622/ 742 based on multiclass loss(15.92885). The deadline is December 27th. submission.PNG

With 70 features, our rank is 611/751 with a multiclass loss of 13.39.



**References**

[1]. <http://scikit-learn.org/stable/>, Dec 11th, 2015

[2]. Germán Creamer, Learning with decision trees: ADTrees, bagging, and random forests, Stevens Institute of Technology, Oct, 2015

[3]. Ryan Rifkin, Multiclass Classification, MIT, 25 Feb 2008

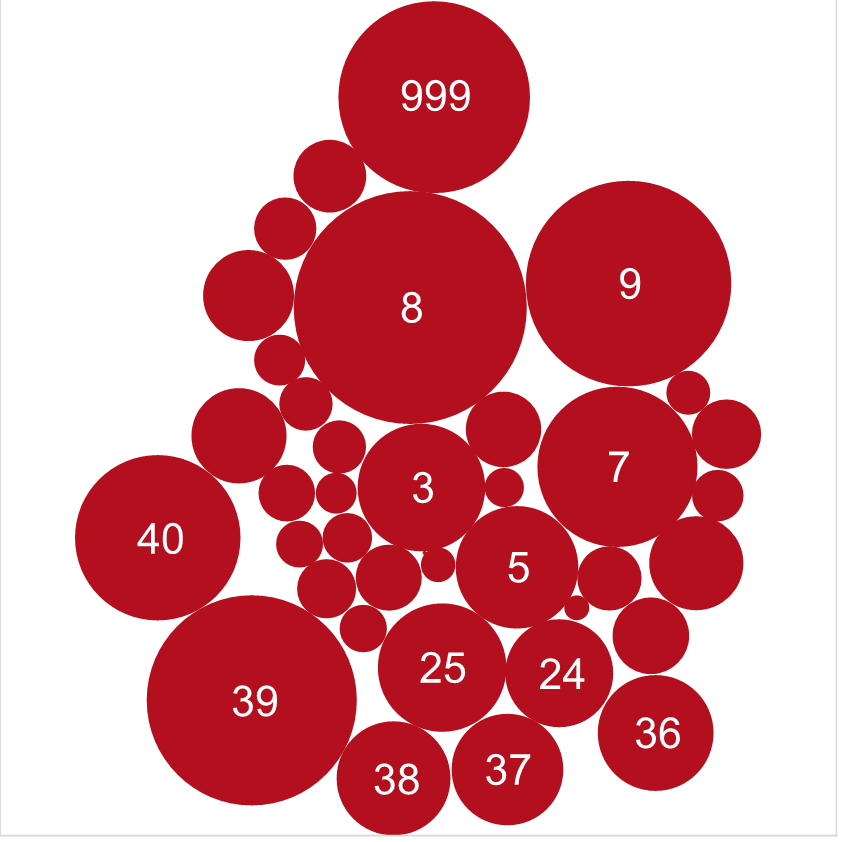


Figure 1: Frequency of the TripType distribution. Trips 8,9,39, 999, 7 and 40 are the most prevalent.

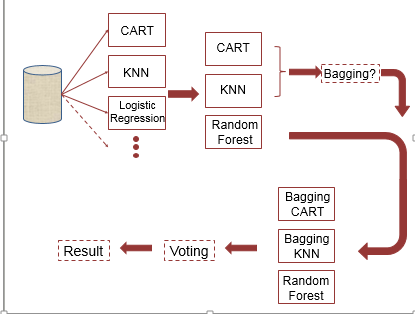


Figure 2: The progression of modeling used. We first tried CART, KNN, Logistic Regression and then proceeded to comparing CART,KNN with Random Forest. We then bagged the first two algorithms and incorporated them into a voting method to obtain a weighted average.

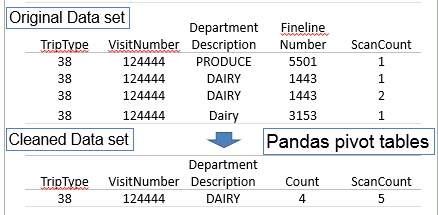


Figure 3: The transformation of the original data set. The number of observations become independent after the transformation and are reduced by approximately sixfold.

|  |  |  |
| --- | --- | --- |
| Index | Method | Accuracy Score |
| 1 | KNN | 0.4829 |
| 2 | CART | 0.4911 |
| 3 | Random Forest | 0.4992 |
| 4 | Bagging | 0.5012 |
| 5 | Bagging of KNN | 0.5135 |
| 6 | Bagging of CART | 0.5019 |
| 7 | Voting of 4,5,6 | 0.5342 |
| 8 | Gradient Boosting | 0.5514 |

Table 2: The accuracy scores of the different methods used for 6 features. Gradient boosting is outperforming the Voting method of Random Forest, Bagging of CART, and Bagging of KNN. However, Gradient Boosting takes longer to run.

precision recall f1-score support

**3 0.77 0.63 0.69 1060**

4 0.00 0.00 0.00 93

**5 0.67 0.66 0.67 1026**

**6 0.79 0.51 0.62 410**

7 0.60 0.47 0.53 1799

**8 0.59 0.85 0.70 3595**

9 0.52 0.67 0.58 2831

12 0.00 0.00 0.00 65

15 0.45 0.32 0.37 318

18 0.36 0.24 0.29 155

19 0.28 0.09 0.14 96

20 0.54 0.56 0.55 207

21 0.48 0.55 0.51 221

22 0.47 0.25 0.33 282

23 0.40 0.33 0.36 42

24 0.48 0.47 0.48 772

25 0.61 0.62 0.62 1097

26 0.35 0.28 0.31 144

27 0.55 0.58 0.56 234

28 0.45 0.29 0.35 147

29 0.14 0.01 0.03 138

30 0.41 0.20 0.27 312

31 0.63 0.47 0.54 171

**32 0.62 0.66 0.64 586**

33 0.49 0.52 0.51 372

34 0.60 0.56 0.58 230

35 0.47 0.48 0.47 605

36 0.55 0.58 0.57 927

37 0.52 0.52 0.52 793

38 0.46 0.27 0.34 801

39 0.43 0.65 0.52 2895

**40 0.73 0.84 0.78 1829**

41 0.00 0.00 0.00 180

42 0.34 0.09 0.15 565

43 0.20 0.00 0.01 274

44 0.37 0.05 0.09 332

999 0.48 0.23 0.31 2449

avg / total 0.53 0.55 0.53 28053

Table 3: The Classification report for our 6 feature model: trips 3,5,6,8, 32, and 40 can all be reasonably predicted.

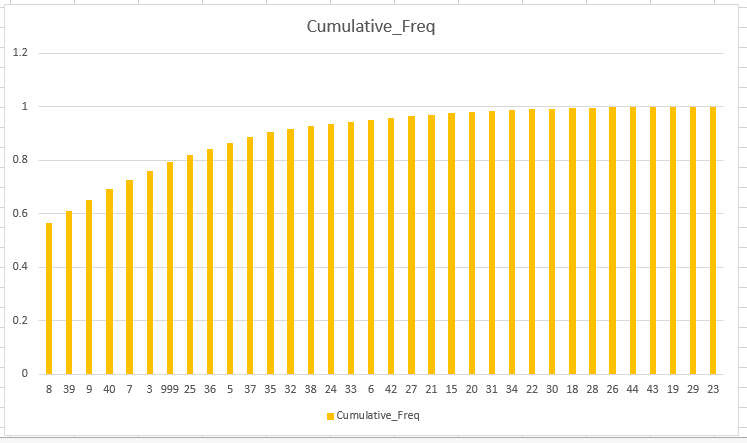


Figure 4: The cumulative frequency of the TestTypes predicted by our 6 feature model for the test data set. 7 variables account for 80% of the trip categories while 12 account for 90%.

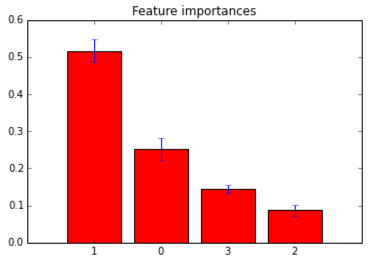


Figure 5: Feature Importances. Department Description > Scan Count > Weekday > Count.

**Appendix A: Contribution of each team member (as per writeup requirement)**

Index Component Group Member

1 Topic and poster idea Danny

2 Pre-processing program Vijay

3 Classification model Di, Vijay

4 Initial Poster Powerpoint Di, Vijay

5 Poster Presentation Danny

6 Final Powerpoint Di

7 Demo Vijay, Di

8 Paper Di, Vijay, Danny

**Appendix B: Results with using 70 features**

precision recall f1-score support

**3 0.77 0.93 0.85 1095**

4 0.00 0.00 0.00 114

**5 0.66 0.87 0.75 1416**

**6 0.64 0.68 0.66 384**

7 0.62 0.66 0.64 1720

**8 0.56 0.88 0.68 3670**

9 0.57 0.72 0.64 2864

12 0.38 0.07 0.11 75

14 0.00 0.00 0.00 1

15 0.46 0.33 0.38 292

18 0.00 0.00 0.00 178

19 0.39 0.13 0.19 118

20 0.57 0.48 0.52 191

21 0.54 0.61 0.57 206

22 0.48 0.25 0.33 282

23 0.20 0.07 0.10 43

24 0.58 0.45 0.51 741

25 0.70 0.59 0.64 1076

26 0.54 0.08 0.15 154

27 0.61 0.53 0.57 225

28 0.47 0.25 0.33 144

29 0.14 0.01 0.01 132

30 0.53 0.23 0.33 307

31 0.00 0.00 0.00 165

**32 0.61 0.76 0.68 551**

33 0.50 0.57 0.53 404

34 0.48 0.47 0.48 228

35 0.44 0.60 0.51 618

36 0.52 0.61 0.56 927

37 0.55 0.58 0.57 848

38 0.45 0.42 0.43 838

39 0.49 0.43 0.46 2937

**40 0.83 0.60 0.70 1821**

41 0.00 0.00 0.00 191

42 0.44 0.01 0.01 554

43 0.50 0.00 0.01 271

44 0.00 0.00 0.00 368

**999 0.82 0.74 0.78 2446**

avg / total 0.58 0.60 0.57 28595

Table A:: The Classification report for our 70 feature model: trips 3, 5,6,8,32,40,999 can all be reasonably predicted.

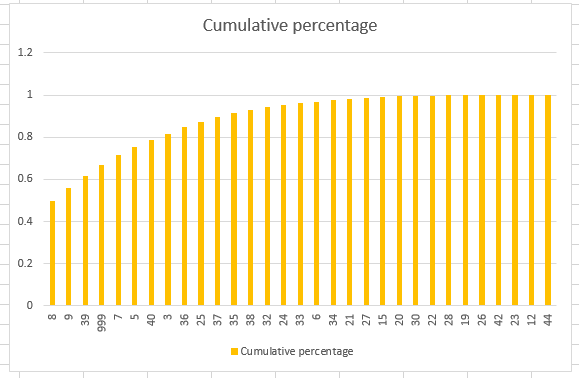


Figure A: The cumulative frequency of the TestTypes predicted by our 70 feature model for the test data set. 11 variables account for 80% of the trip categories while 14 account for 90%.