Instacart Market Basket Analysis

Which product will an instacart consumer purchase again

Nutan Mandale

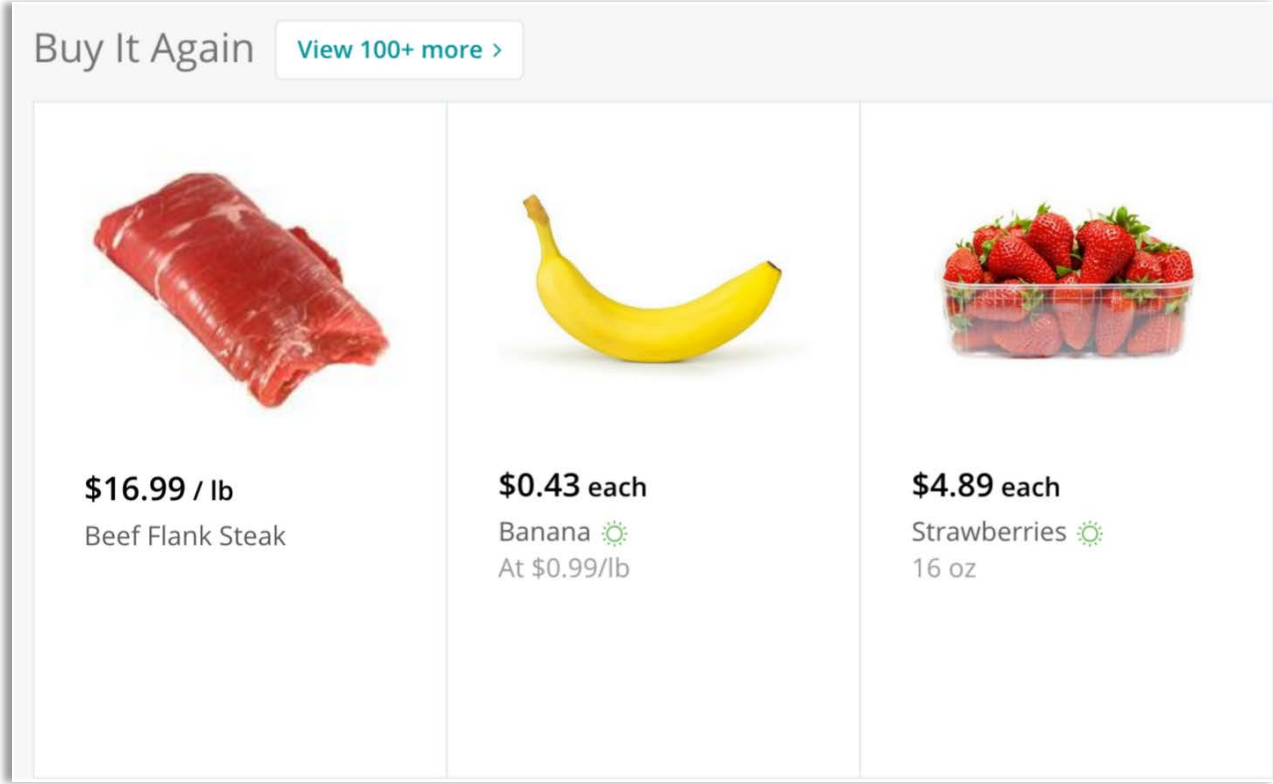
About Instacart

- Instacart is a grocery ordering and delivery app.
- Instacart aims to make it easy to fill consumer's refrigerator and pantry with their personal favorites and staples when they need them.
- Instacart team has been doing transaction data analysis and build the models that predicts which products a user will buy again, try for the first time, or add to their cart next during a session.

.

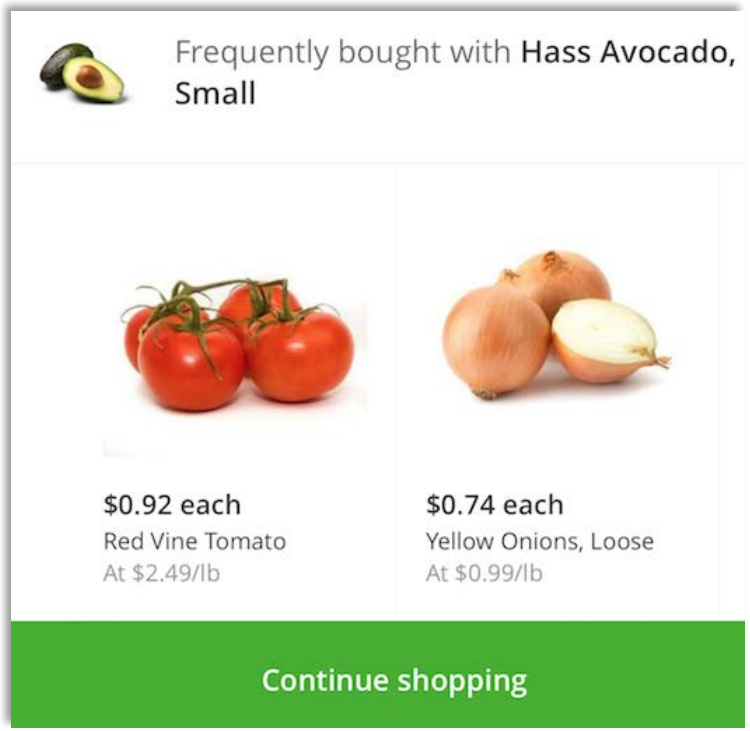
About Instacart

- For each user, Instacart provide between 4 and 100 of their orders, with the sequence of products purchased in each order.



About Instacart

- They also recommend different items to the users while they shop.



About Competition

- Instacart is challenging the Kaggle community to use their anonymized data on customer orders over time to predict which previously purchased products will be in a user's next order.

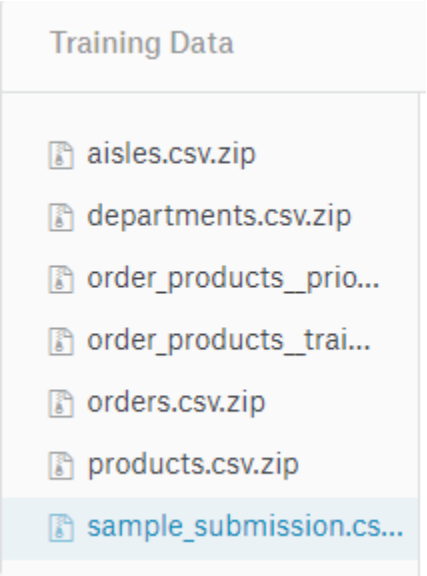
- Evaluation is based on the mean F1 score.

Data Description

- The dataset is a relational set of files describing customers' orders over time.
- The dataset is anonymized and contains a sample of over 3 million grocery orders from more than 200,000 Instacart users.
- Each entity (customer, product, order, aisle, etc.) has an associated unique id.

Data Description

- The aisles.csv file consists of the information of aisles.
- The departments.csv consists of the information of departments.
- The order_products_prior.csv specify which products were purchased in each order
- The orders.csv tells to which an order belongs to.
- The products.csv describes the products for sale.



Exploratory Data Analysis

After merging all datasets we get a dataset with total

- 32434489 samples
- 15 columns

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 32434489 entries, 0 to 32434488
Data columns (total 15 columns):
order_id                  int64
user_id                   int64
eval_set                  object
order_number              int64
order_dow                 int64
order_hour_of_day         int64
days_since_prior_order    float64
product_id                int64
add_to_cart_order         int64
reordered                 int64
product_name              object
aisle_id                  int64
department_id             int64
aisle                     object
department                object
dtypes: float64(1), int64(10), object(4)
```

Exploratory Data Analysis

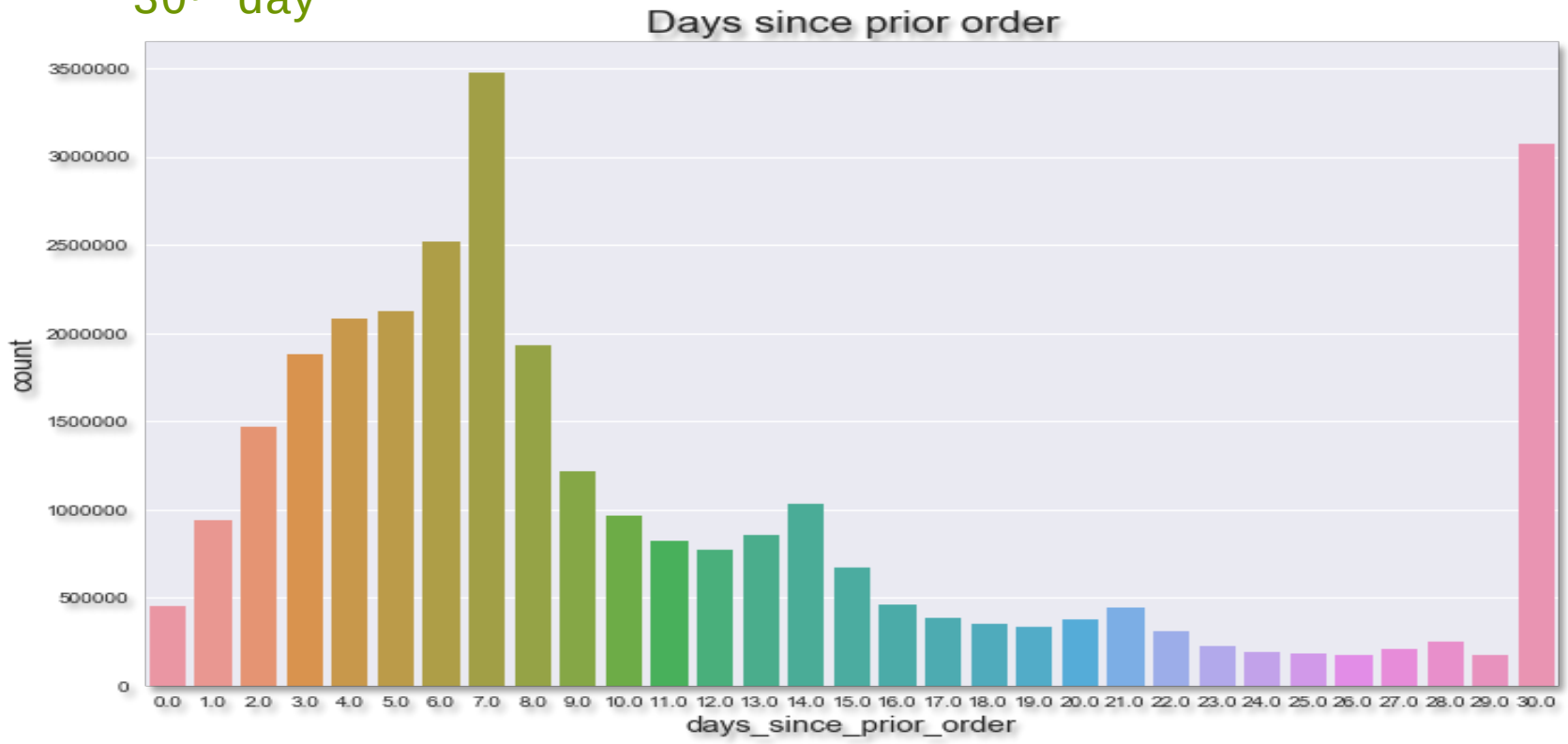
- The sample size too big.

- Worked with different sample sizes.
  - 50k
  - 100k
  - 500k
  - 3 million

Exploratory Data Analysis

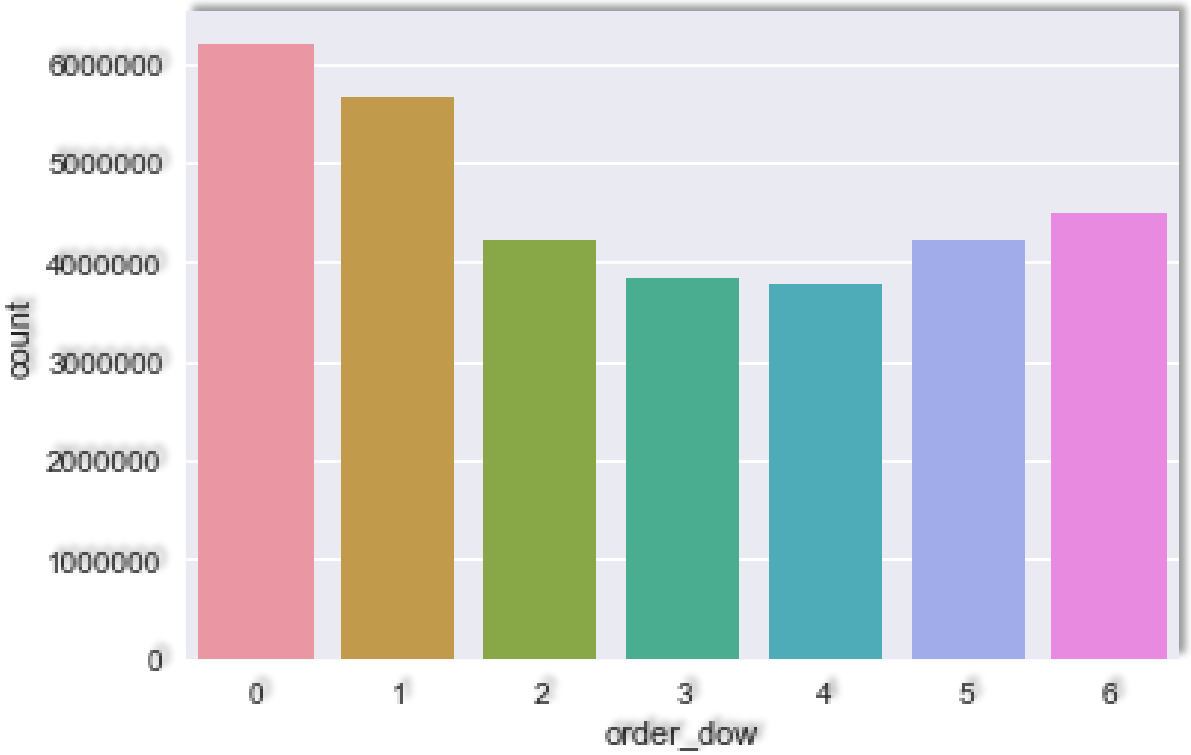
Some interesting shopping patterns…

Shopping frequency of consumers is maximum on 7th and 30th day



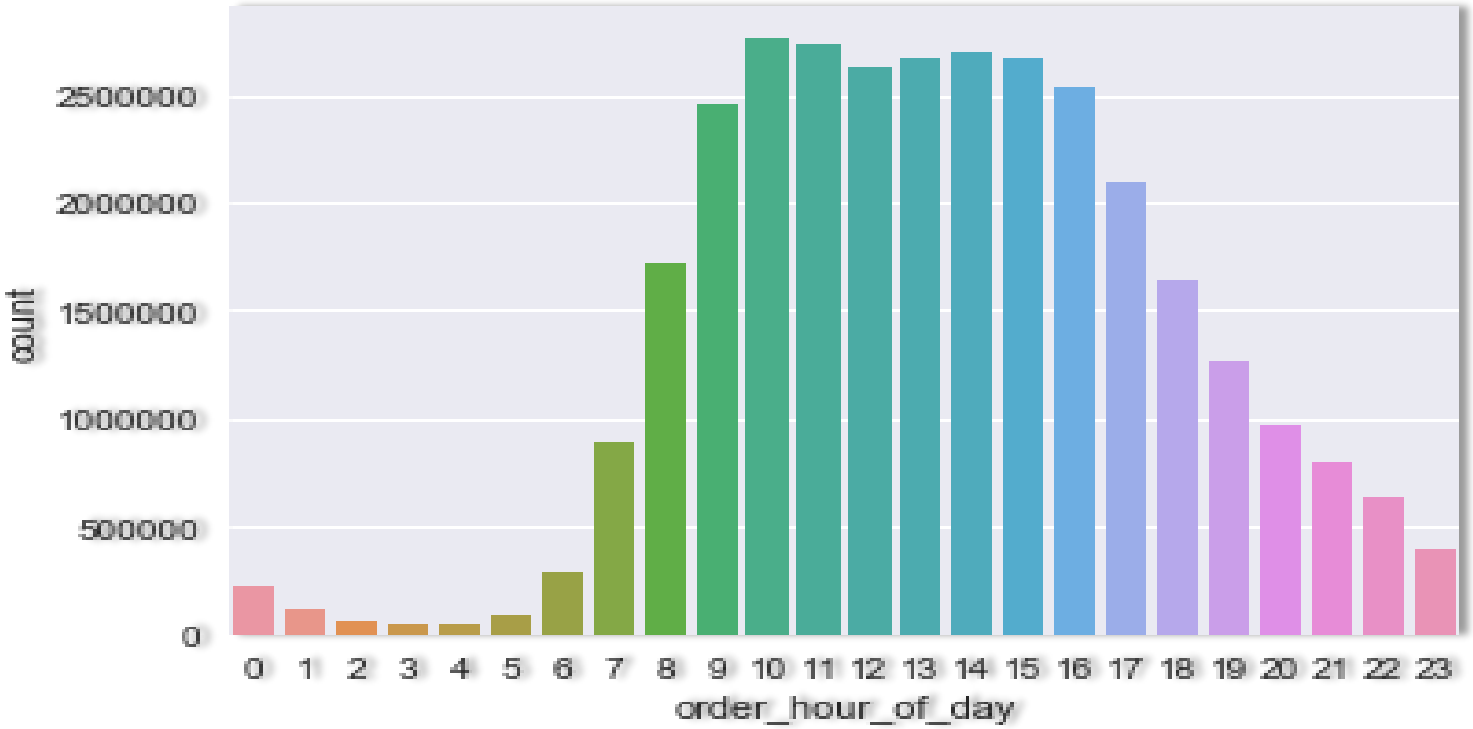
Exploratory Data Analysis

Sunday is most popular day for shopping



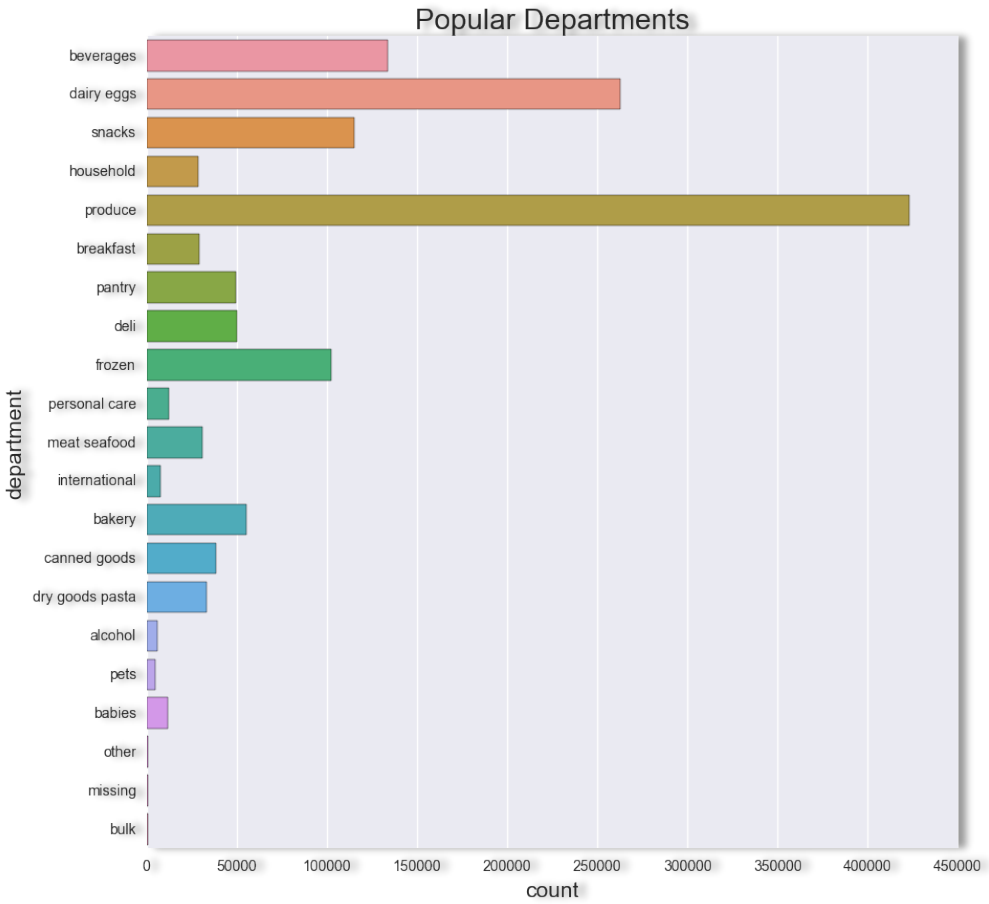
Exploratory Data Analysis

Consumer traffic is highest during 9:00am – 4:00pm of the day



Exploratory Data Analysis

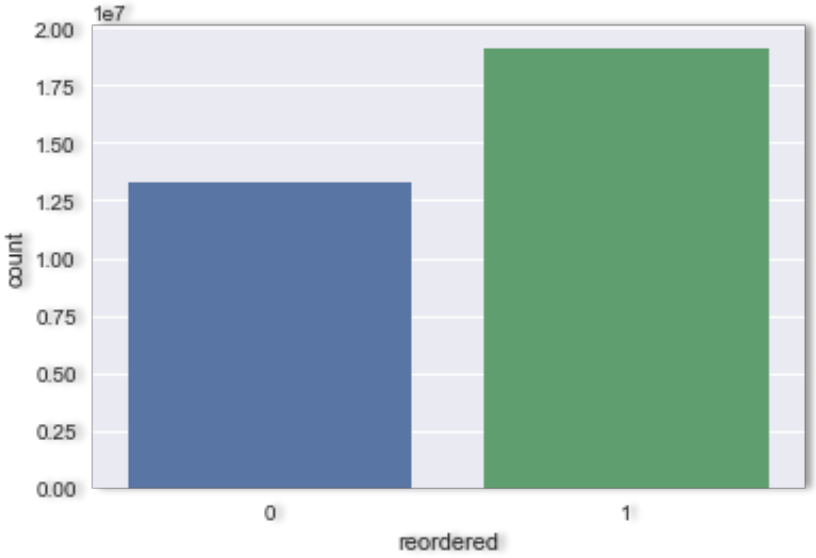
Produce department is the most popular department of all and consumers reorder most from produce department.



Exploratory Data Analysis

Dependent variable is reordered variable.

- This is a binary classification exercise.



Exploratory Data Analysis

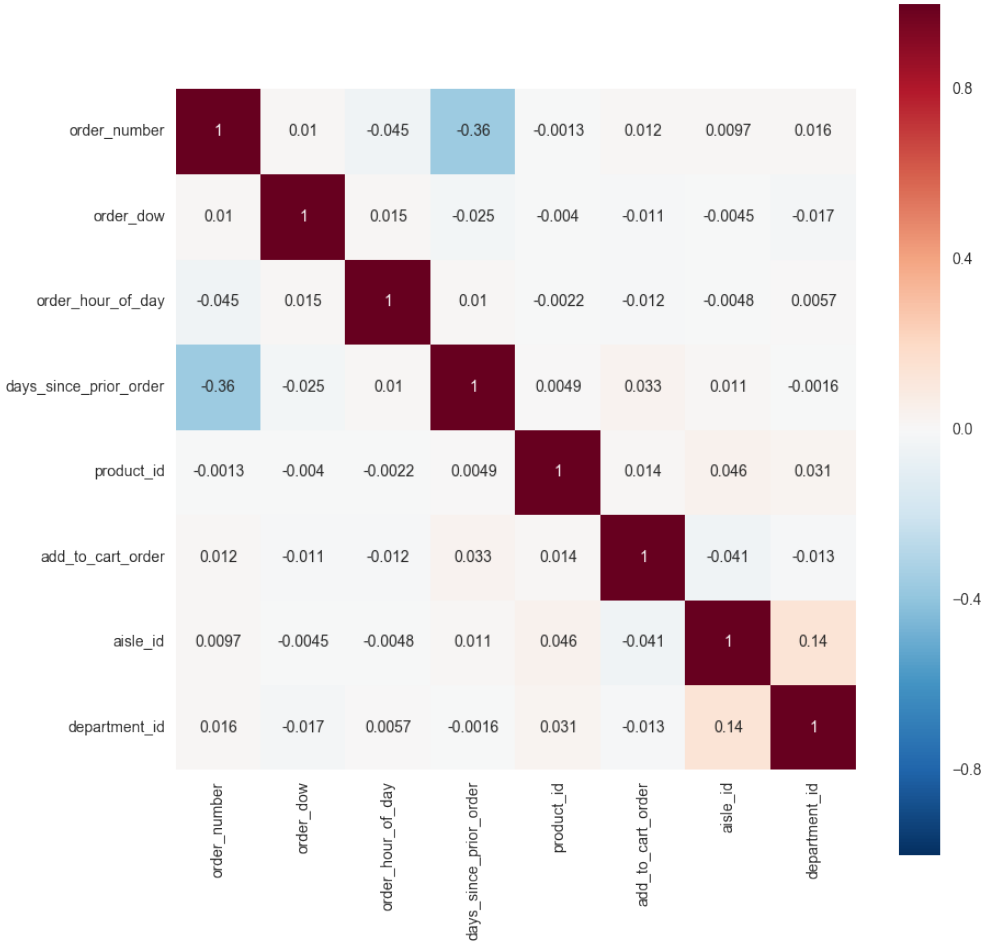
- There are total 8 independent variables.
- Total observations = 2807851 after dropping empty rows from the sample of 3 millions.

```
X.dtypes
```

```
order_number                int64
order_dow                   int64
order_hour_of_day           int64
days_since_prior_order    float64
product_id                  int64
add_to_cart_order           int64
aisle_id                    int64
department_id               int64
dtype: object
```

Correlation Analysis

- There is very little correlation between independent features.
- There is little correlation between order_number and reorder of the item



Feature Importance

- Order_number is the most influential feature.

Models Used

Total number of models used:

- Ridge Regression
- Logistic Regression
- KNN Classifier
- Decision Tree Classifier
- Gradient Boosted Classifier
- Random Forest Classifier
- Bootstrap Aggregation (Bagging)

Model Evaluation

All the models are evaluated on following criterion

- Confusion Matrix
- Accuracy
- ROC curve
- AUC(Area under curve)

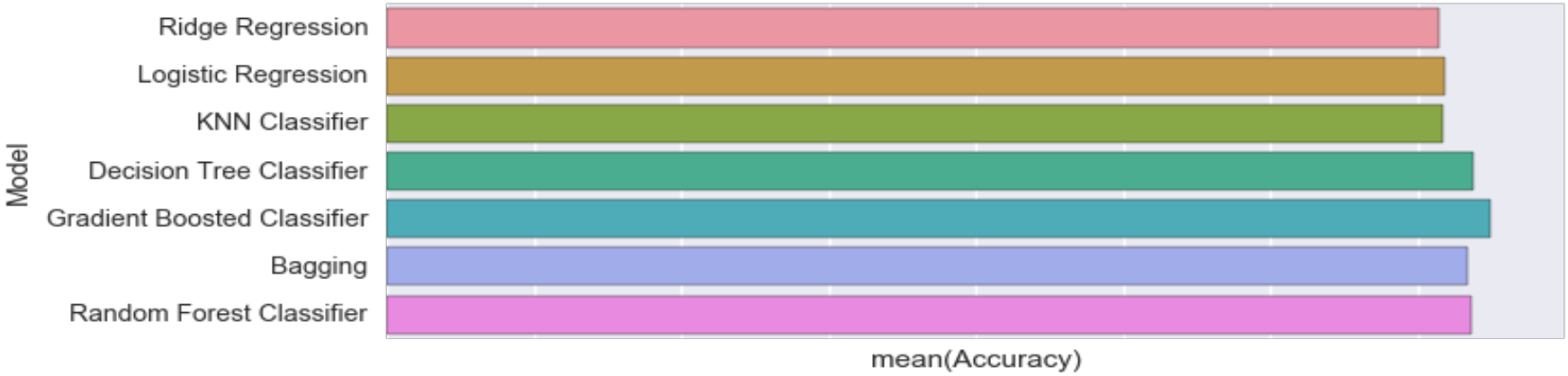
Revised Model

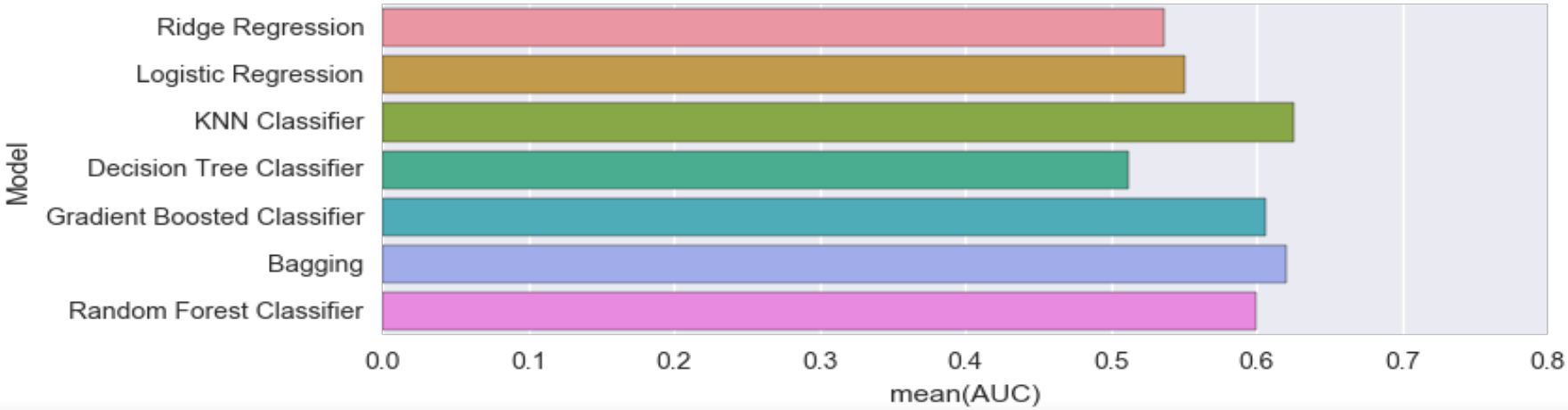
Measures taken to improve the performance of the model

- Bagging(Bootstrap aggregation)
- Gradient Boosting

Initial Model Result

Bar plot of different models with their accuracy and AUC





Initial Model Result

Gradient Boosted Classifier shows maximum accuracy

	AUC	Accuracy	Model
0	0.537	0.715	Ridge Regression
1	0.551	0.719	Logistic Regression
2	0.625	0.717	KNN Classifier
3	0.512	0.738	Decision Tree Classifier
4	0.606	0.750	Gradient Boosted Classifier
5	0.621	0.734	Bagging
6	0.600	0.737	Random Forest Classifier

Hyper parameter Tuning

Results tuning the hyper parameters

	Default	Tuned
KNN	0.668	0.717
Decision Tree	0.707	0.738

Revised Model

- Standardize features by removing the mean and scaling to unit variance.
- Used 10-fold cross validation with mean score of AUC,Accuracy and f1 score

```
10-fold cross validation
ROC AUC: 0.70 (+/- 0.00) | Accuracy: 0.72 (+/- 0.00) | f1 Score: 0.83 (+/- 0.0
0) [Logistic Regression]
ROC AUC: 0.74 (+/- 0.00) | Accuracy: 0.74 (+/- 0.00) | f1 Score: 0.83 (+/- 0.0
0) [Decision Tree]
ROC AUC: 0.71 (+/- 0.00) | Accuracy: 0.73 (+/- 0.00) | f1 Score: 0.83 (+/- 0.0
0) [KNN]
ROC AUC: 0.74 (+/- 0.00) | Accuracy: 0.74 (+/- 0.00) | f1 Score: 0.83 (+/- 0.0
0) [Random Forest]
ROC AUC: 0.74 (+/- 0.00) | Accuracy: 0.75 (+/- 0.00) | f1 Score: 0.84 (+/- 0.0
0) [Gradient Boosting]
```

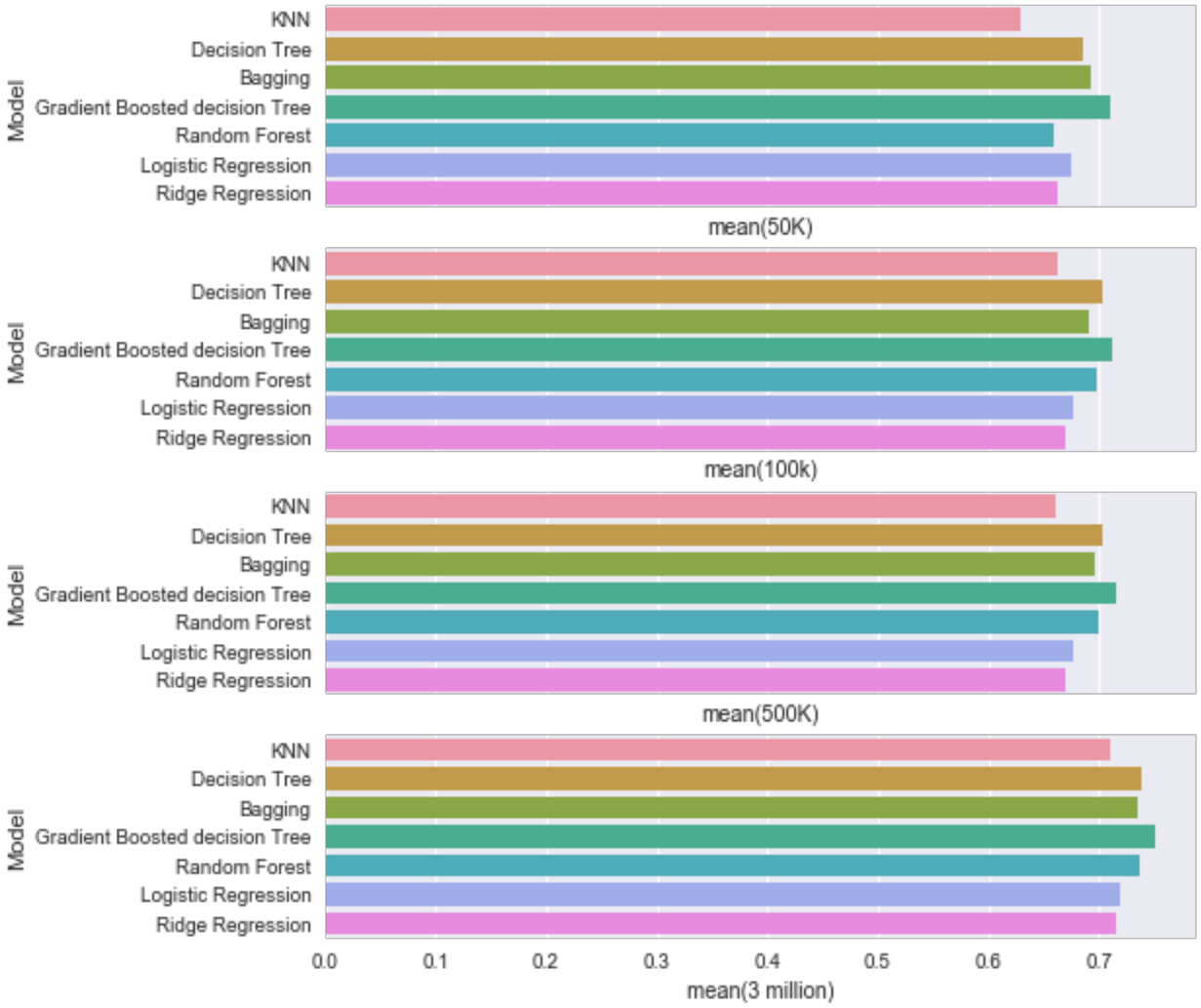
Different Sample Size Comparison

As the sample size increased there is increase in the accuracy score of the model.

	100k	3 million	500K	50K	Model
0	0.663	0.710	0.660	0.628	KNN
1	0.703	0.738	0.703	0.686	Decision Tree
2	0.691	0.734	0.695	0.693	Bagging
3	0.712	0.750	0.715	0.710	Gradient Boosted decision Tree
4	0.697	0.737	0.700	0.659	Random Forest
5	0.676	0.719	0.676	0.674	Logistic Regression
6	0.669	0.715	0.669	0.662	Ridge Regression

Different Sample Size Comparison

As the sample size increased there is increase in the accuracy score of the model.



Conclusion

- Standardizing features increases the accuracy and area under curve.
- As the sample size increased there is increase in the accuracy score of the model.

Future Improvements

- Only 10% of the data has been used for the models to train as well as test due to computational limits.
- Using multiple weighted models could also help in improving the performance.
- SVM model could have been used.