Preventing Churn

Jaime, Jiexi, Sherry, Ian, and Trevor

Agenda

- Customer’s Business Objective
- The Data
- Expected Profit Per Customer Model
- Model Evaluation
- Model Tuning
- Conclusion

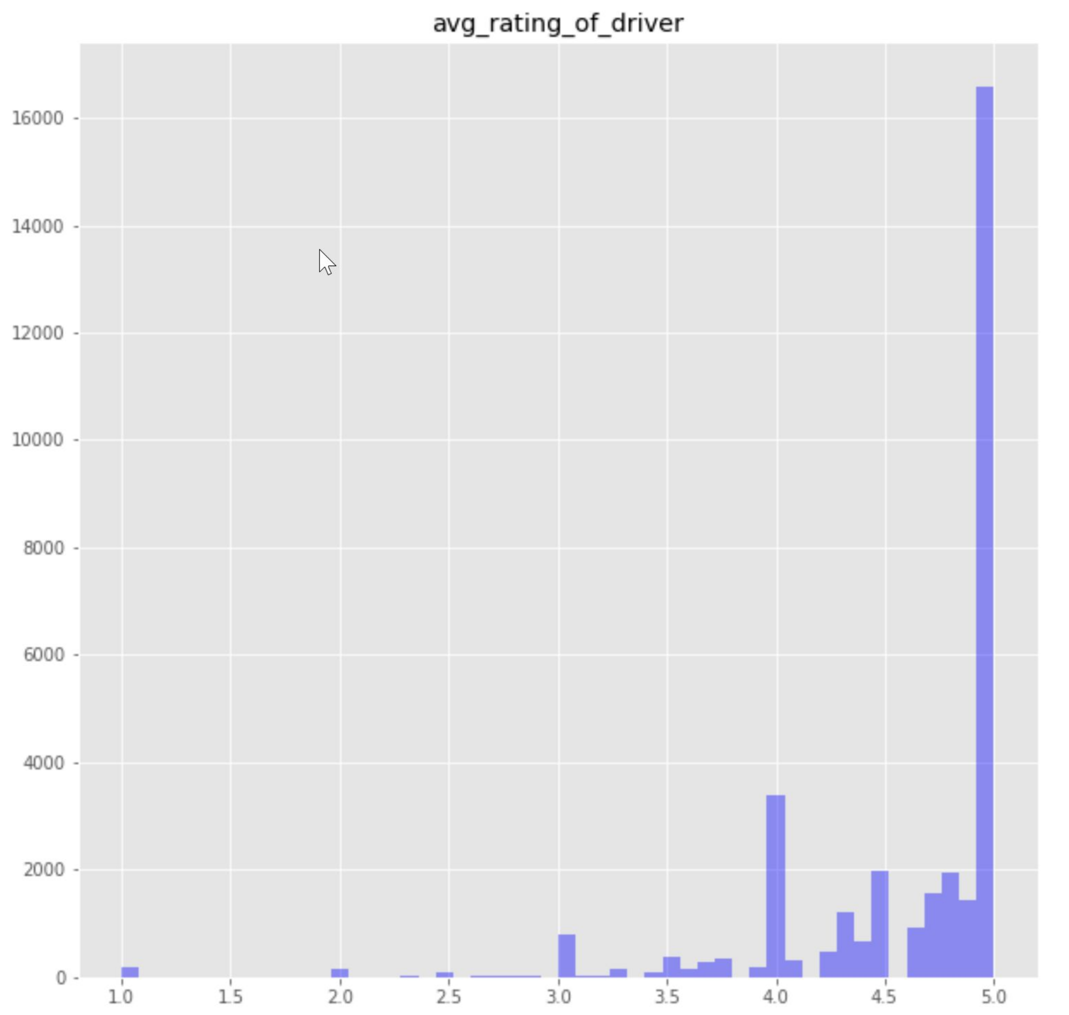
Customer’s Business Objective

- Company X wants to be able to predict customer ‘churn’, so that they can potentially intervene.
- Churn: defined as customers who’s last ride was more than thirty days in the past and thus are considered lost.
- Whether to intervene or not is a business decision based upon:
  - The value of a customer, the cost of intervention, & the likelihood of the customer churning
  - To aid in decision making, we provided the customer a decision support tool.

The Data

Ridership data for ‘Company X’ covering January to June 2014.

- 50,000 rows separated into training (40k) and test (10k) sets.
- 11 columns, seven numerical, four objects
- The four objects are two dates and two values taken from small sets
- Not a lot of nulls except in the ‘average_rating_of_driver’ column.
- Each row represents a summary of a single customer, not a single ride.



avg_rating_of_driver
16000
14000
12000
10000
8000
6000
4000
2000
0
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0

Feature Engineering

Created:

- ‘days_since_last_ride’ feature
- ‘days_since_signup’ feature
- ‘churned’ feature

Converted:

- ‘avg_rating_by_driver’ NaNs to mean
- ‘avg_rating_of_driver’ NaNs to mean
- One-hot encoded phone and city

Considered transformations, but tree-based models do not benefit so we didn’t

Metric: **Expected Profit per Customer**

By predicting which customers will churn ahead of time, we can attempt to intervene and prevent churn to optimize company profits using the following assumptions...

Cost Benefit Matrix Applied to Predictions

True Positives **+$40**	**False Positive** **-$20**
False Negative **$0**	**True Negative** **$0**

- **Assumptions**
- $120 - Value of Customer Retained from Jan cohort
- -$20 - Gift we apply to customers we project to churn
- 50% of Customers we give free rides are expected to stay on
- No action taken to customers not predicted to churn

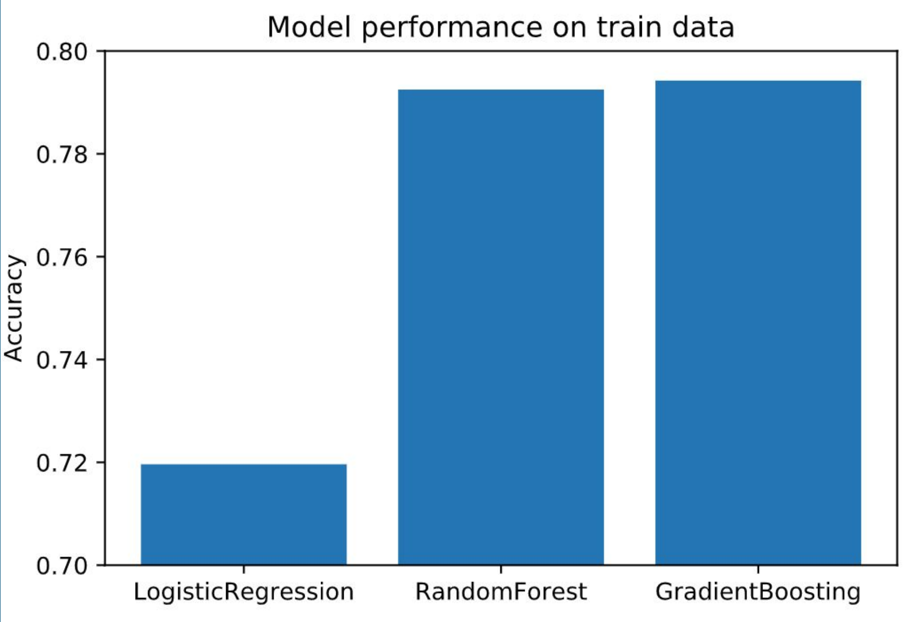
Baseline Model

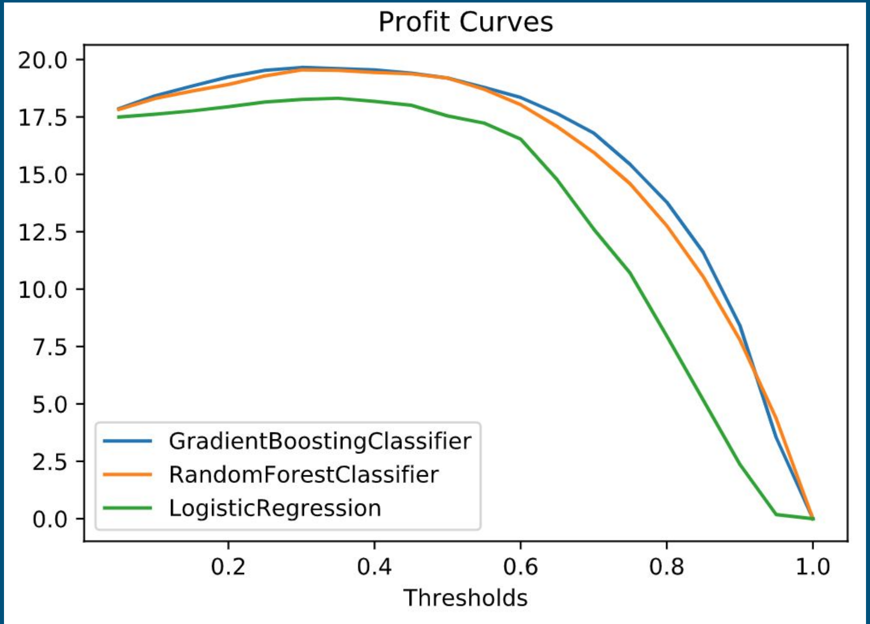
Tested out a number of different iterations with the following models:

1. LogisticRegression (defaults)
2. RandomForestClassifier(min_samples_leaf=4, n_estimators=1000)
3. GradientBoostingClassifier(learning_rate=0.1, n_estimators=500)

Numerical features with mean inserted for NaN; categorical features dummied

Performance on training set (80:20 split)





Results on unseen data

Model used: GradientBoostingClassifier

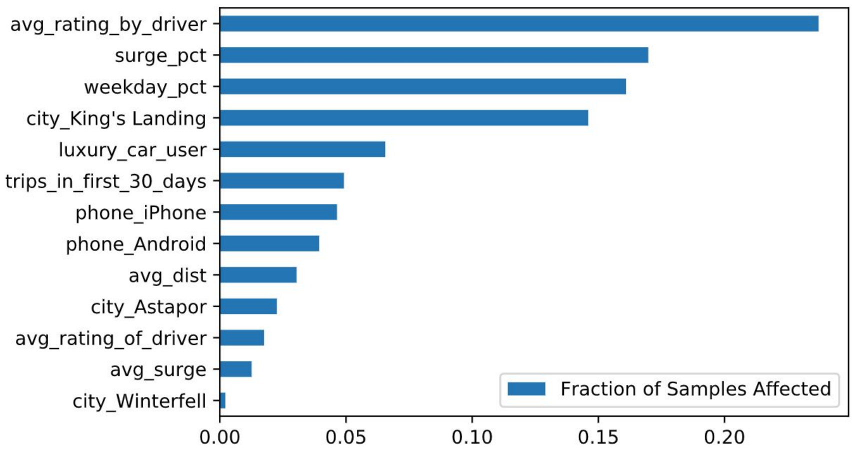
Accuracy: 0.784

Feature importance (top 3):

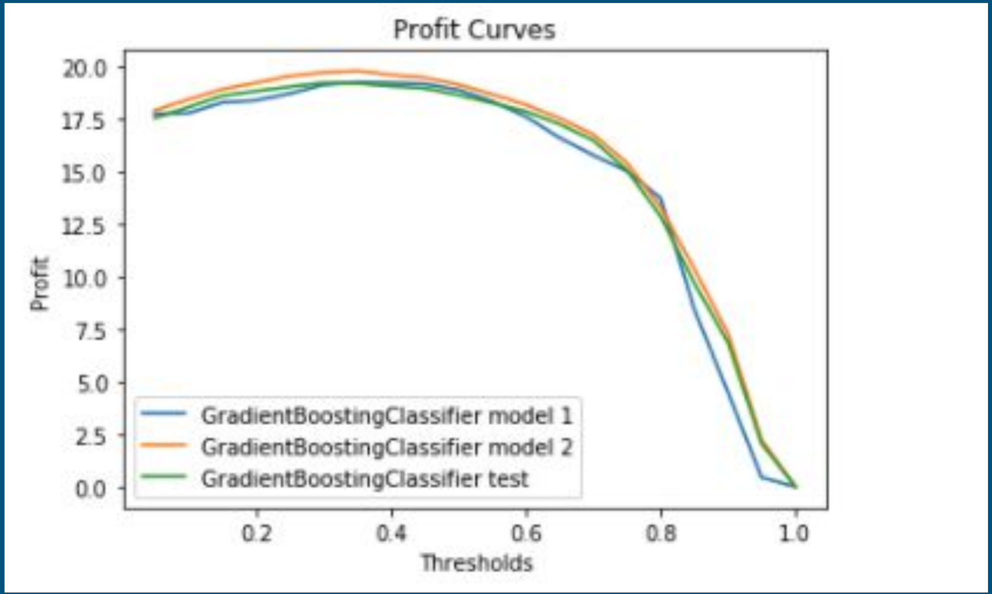
Avg_rating_by_driver

Surge_pct

weekday_pct



Focus on GradientBoostingClassifier



- **Best Model: Gradient Boosting Classifier**
    - **Learning_rate = 0.05**
    - **N_estimators = 500**
    - **Min_samples_leaf = 10**
    - **Min_samples_split = 10**
- **Max Profit per Customer: $19.18**
- **Best Threshold: 0.3**
- **Relevant Features: avg_dist, weekday_pct, surge_pct, avg_rating_by_driver, King's Landing, luxury_car_user**

Summary

Business Impact: $19.18 Profit Per customer*

Final Model: Gradient Boosting Classifier @ threshold of 0.3

Relevant Features to Churn: Avg Rating by Driver, Surge Pct, and Weekday %

Proposed Actions: Gift $20 in Free Rides to identified churn risks.

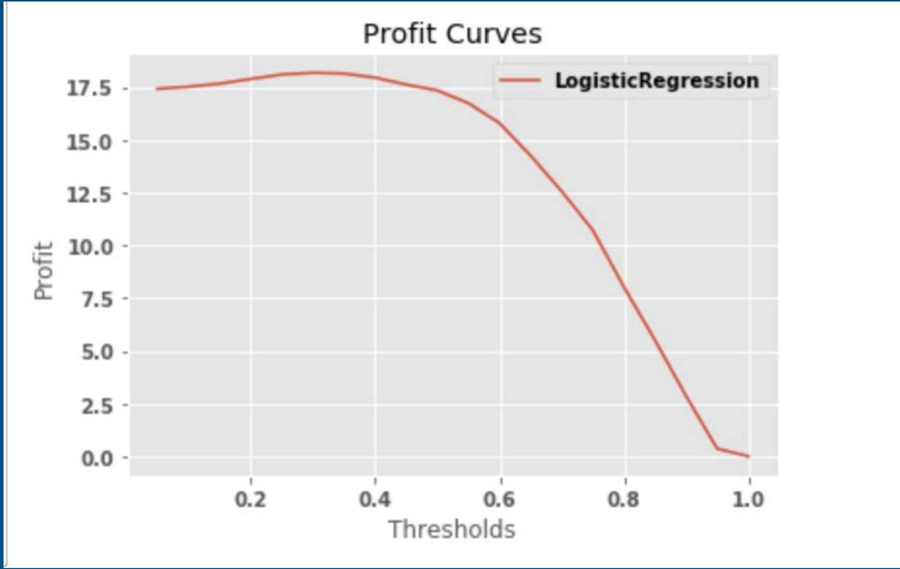
- Can further optimize by excluding Users w/ low ratings by drivers.

**Note: Using assumptions to create cost-benefit matrix.*

Questions?

Appendix

Simple Model Example - (@group - We can delete! Included as pseudo placeholder)



- **Logistic Regression Example**
- 12 Features (incl 1 created feature)
  - Added a constant and removed iPhone and Winterfell to avoid Dummy Var trap

- Log Loss = 9.85
- Accuracy = 0.71

<- results are against the unseen Test data.

Max Profit Per Customer is $18.18 at threshold of 0.3 … Can we beat this?