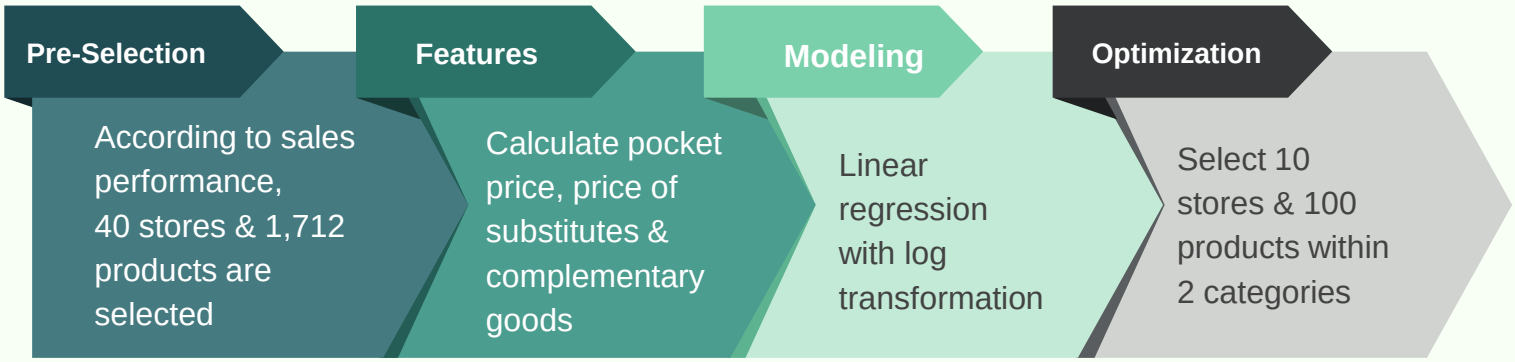
01 Marketing Analytics – Pricing

Overview:
This project is to develop a pricing scheme for the same superstore in last project. We set a pricing strategy that will maximize store revenues while still maintaining profits. There are a set of constraints that we must abide by. These constraints include 100 products that are to be priced, belonging to 2 product categories, to be offered at 10 stores. We find the combination of products, categories, and stores that optimize total revenue based on the new prices we set.

- Keywords:
  - Demand Curve
  - Linear Regression
  - Optimization
- Tools: Python
- Data: 18 × 29.6M



Relationship between Demand & Price

$$\log D_{ps} = \alpha + \beta_p * \log L_{ps} + \varepsilon_p * \log P_{ps} + \sum_i \sigma_{psi} * \log S_{psi} + \lambda_p * \log K_{ps} + \sum_i \gamma_{psi} * season_{psi}$$

Relationship between Revenue & Price

$$\log(revenue) = \mu + \sigma * \log L_{ps} + \log L_{ps} + \log(1 - promotion_{rate}) = (\sigma + 1) * \log L_{ps} + \mu'$$

Where
D is the historical demand for product p in store s
L is the historical list price for product p in store s
P is the historical pocket price for product p in store s
S is the historical price for product p's substitutes in store s
K is the historical pocket price for product p's complementary goods in store s
Season is the seasonal factor, which is 12 months
μ, σ, μ', σ' are all constants

More details:
https://github.com/mays1770/Pricing

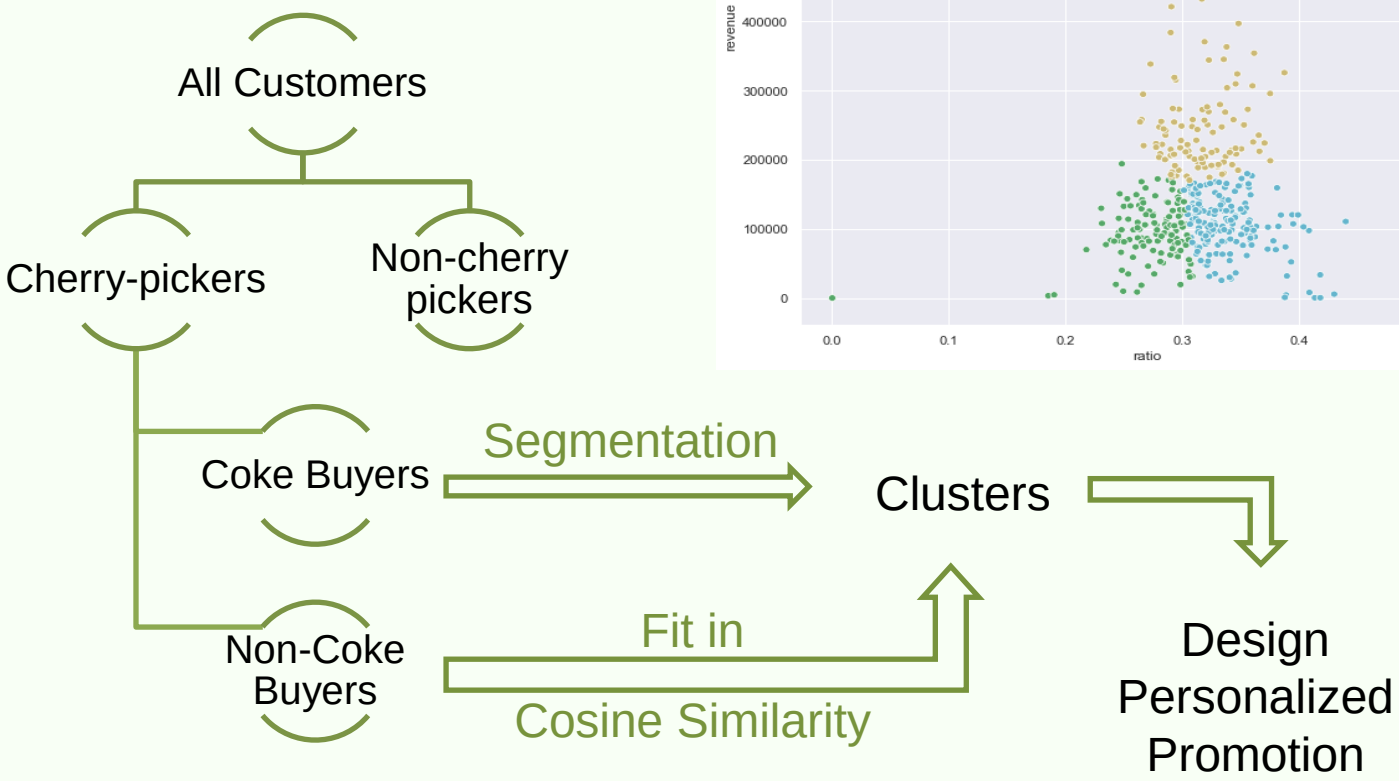
Created By: May Shao

02 Marketing Analytics – Recommender System

Overview:
This project is to design personalized promotions to offer customers for a leading supermarket chain of over 400 stores. The full personalized promotion includes details about: all customers that will be targeted by the campaign, the soft drink product being offered, and the discount price assigned to each customer. We also generated an estimate of the total discount redemption cost along with the incremental volume as a result of our campaign.

- Keywords:
K-Means, Cosine similarity
- Tools: Python
- Data: 18 × 29.6M

K-Means Clusters Example



More details:
https://github.com/mays1770/RecommenderSystem

Created By: May Shao

03 Machine Learning - Spam Email Detection

Overview:
The dataset can be found at http://archive.ics.uci.edu/ml/datasets/Spambase
I used common machine learning algorithms such as lightGBM, SVM and tried stacking to classify spam vs. non-spam emails. The evaluation is based on overall accuracy and cost analysis. I also considered precision, recall, f1-score, ROC curves and other metrics in deciding the final model. The overall accuracy reaches 0.96.

- Keywords:
  - Stacking
  - lightGBM
  - SVM
- Tools: Python
- Data: 58 × 4,601

Data Processing

- Polynomial Transformation
- Chi-square Test
- Standarization

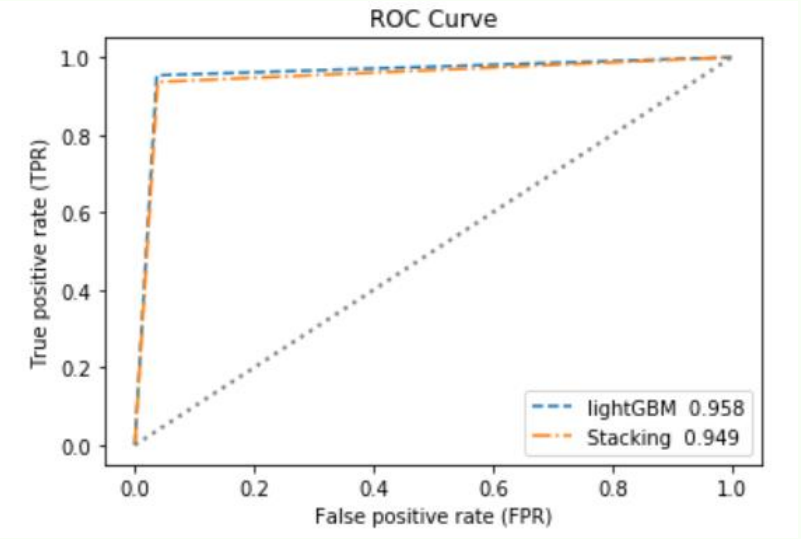
Modeling

- Decision Tree
- Naïve Bayes
- lightGBM
- Support Vector Machine
- Random Forest
- Logistic Regression
- Stacking

Evaluation

- Avg_cost_score
  - Take cost into consideration
- Accuracy
- F-1 Measure
- ROC Curve

Model	Accuracy	Avg_cost
lightGBM	0.9592	0.2745
Stacking	0.9574	0.2407



More details:
https://github.com/mays1770/Spam_Email_Detection

Created By: May Shao

04 Machine Learning - Titanic with PySpark

Overview:
The dataset can be found at:
https://www.kaggle.com/c/titanic/data
The goal is to predict for each passenger whether he/she survived the Titanic tragedy by using the pipeline and feature functionality of pyspark.ml.
In the pipeline, I assembled the following elements: StringIndexer, OneHotEncoder, VectorAssembler, LinearRegression. The AUC achieves 0.86.

- Keywords:
  - Logistic Regression
  - Pipeline
- Tools: PySpark
- Data: 12 × 891

More details:
https://github.com/mays1770/Titanic_PySpark

05 Machine Learning - Online-ad click-through-rate prediction

Overview:
This project involves predicting clicks for on-line advertisements. The training data consists of data for 9 days from October 21, 2014 to October 29, 2014. Our goal is to predict the probability of a click. The performance criterion used to evaluate the performance of prediction is log-loss.
I used hashing and principal component analysis in data processing and adopted ensemble methods based on linear regression. The log-loss reaches 0.40.

- Keywords:
  - Hashing
  - Pandas Profiling
  - GridSearch CV
- Tools: Python
- Data: 24 × 32M

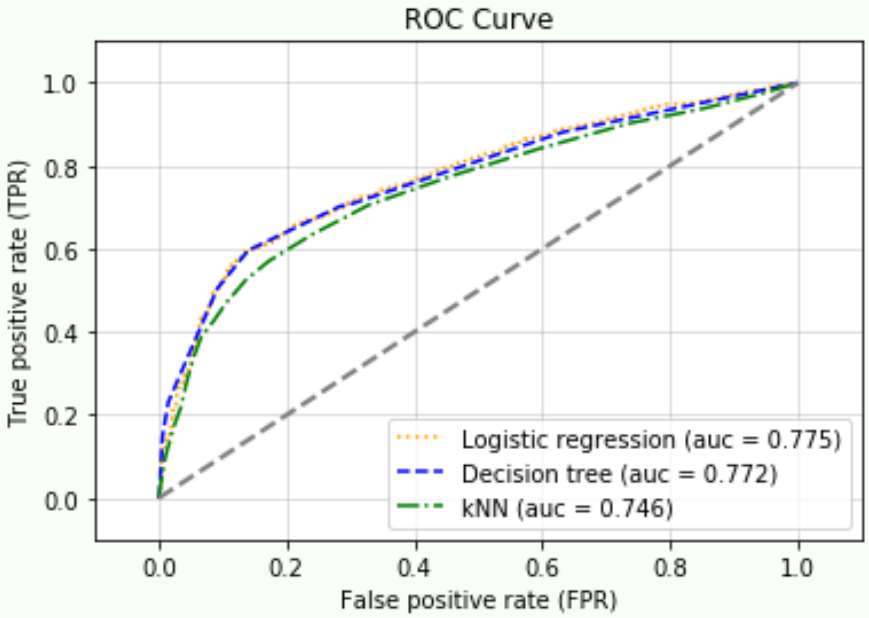
More details:
https://github.com/mays1770/OnlineAd_Click-Through-Rate_Prediction

Created By: May Shao

06 Machine Learning - Bank Deposit Campaign

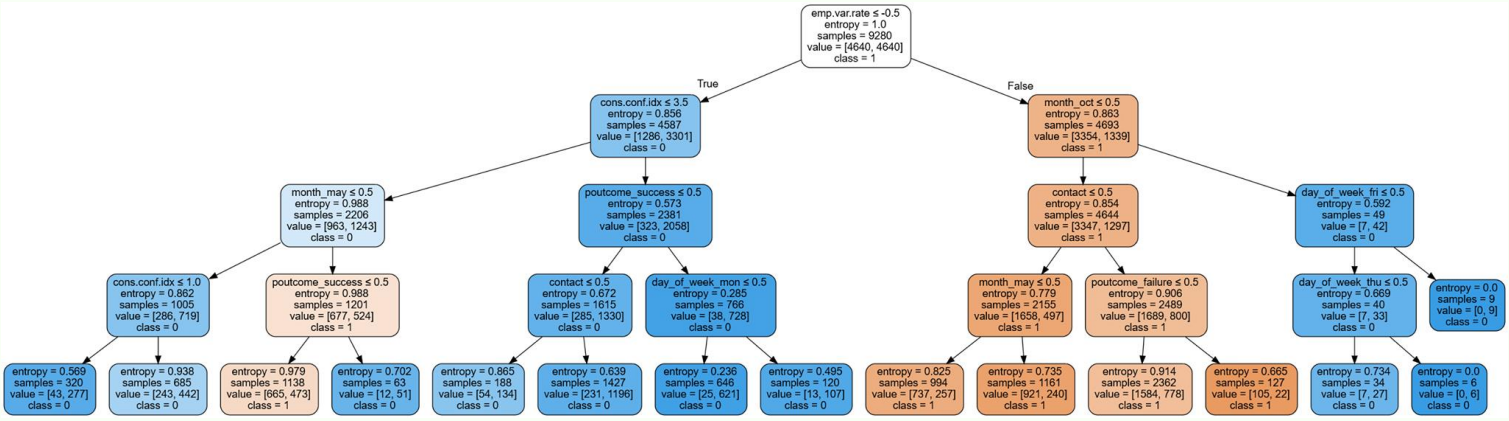
Overview:
A Portuguese bank has made marketing campaigns in order to sell deposits. Our goal is to utilize the data in order to predict whether or not a specific client will subscribe to a term deposit. We compared the performance of kNN, Decision Tree and Logistic Regression in this classification problem. We choose decision tree as our final model based on its overall accuracy (0.741) and business insights it can generate.

- Keywords:
  - Decision Tree
  - GridSearch CV
- Tools: Python
- Data: 17 × 11,163



Model Selection

Decision Tree in this case has similar performance with Logistic Regression. Moreover, the tree map can provide insights for management.



More details:
https://github.com/mays1770/Bank_Deposit_Campaign

Created By: May Shao