Wallet Hub Data Scientist Test Modeling Project Report

Aiden V. Johnson

December 17, 2018

Overview

Goal: Create a model to predict the target variable y while striving for efficiency in the modeling process.

Data Sources:

A csv file containing 100,000 rows and 304 labeled features:x001 to x304

Modeling Response: y values

Classification or Regression Model: Regression

Model Evaluation metrics: RMSE

Modeling Assumptions:

* The variables are unique and don't include any duplicates with unique names.
* Variable collinearity and potential synonymousness will be managed with modeling techniques implemented.
* There are no outliers in the dataset, that require mitigation prior to modeling via IQR or z-score evaluation.

Modeling Methodology Description:

After reading in the dataset the null values were identified, reviewed for counts and distribution

across the dataset. The data size allowed for column-wise dropping of null values. Imputation of null values is considered another reasonable approach, however, the number of features remaining after column-wise dropping of null values did not necessitate imputation.

The large number of features indicate an opportunity to apply dimension reduction techniques in the modeling process. The data was first scale standardized and then a Principle Component Analysis was applied to capture the majority of the variance in the data. Reviewing the plot of the cumulative explained variance against the number of components indicates 96 components are needed to capture 95% of the total variance in the data set.

The data was split into a 70/30 training and testing set, in order to allow for holdout model performance evaluation.
Based on the limited exploratory analysis completed and potential for variable correlation the two modeling algorithms chosen for model development include; Lasso Regression and Random Forest Regression. Both of these techniques can effectively optimize in high feature dimensionality. Using grid search optimization with a testing holdout sample the best parameters for each model were identified. Each grid search optimization was tested first on the 96 components making up the features for training data and then as the combination of the 96 PC's and the original training set concatenated together, for a total of 263 features. The combination of the PC features + the original features provided the best model performance metrics for both the Lasso and the Random Forest regressor.

The final model chosen was the Random Forest Regression regression. The final step in the modeling process was to read in the entire dataset and re-train the random forest model on the entire data set rather than the initial training data subset.

Modeling Methodology Overview:

Data Processing Steps:
-Add PCA plot
Model Development Steps

Model Evaluation Steps

Final Model desc:

Performance Metrics and Error Distribution Review:
-Add Error distribution plot

Modeling Discussion:

Potential Improvements for future work as needed.

Model Implementation Instructions:

Requirements:

- Python version 3.6.1

Packages:

- pandas
- numpy
- sklearn
- joblib
- math
- pickle
- warnings
- os

Steps:

- Unzip the `RF_ML.zip`
- Save the `run_model.py` and `rf_model.joblib` to your working directory
- Open the command line and change the directory to where you saved `run_model.py` and `rf_model.joblib`
- Type `python run_model.py` and press Enter
- Once you see a message that says `Enter file name:` type the name of your input CSV dataset for example `file.csv`
- Model should run in one minutes or less on a laptop with 16GB of RAM
- When the model is complete you will see both the RMSE and MAE printed in the command line.
- The program will have written the model predictions out to a file in the same directory calling it `predictions.csv`