Model

- Model the demand function for avocados (detailed later).

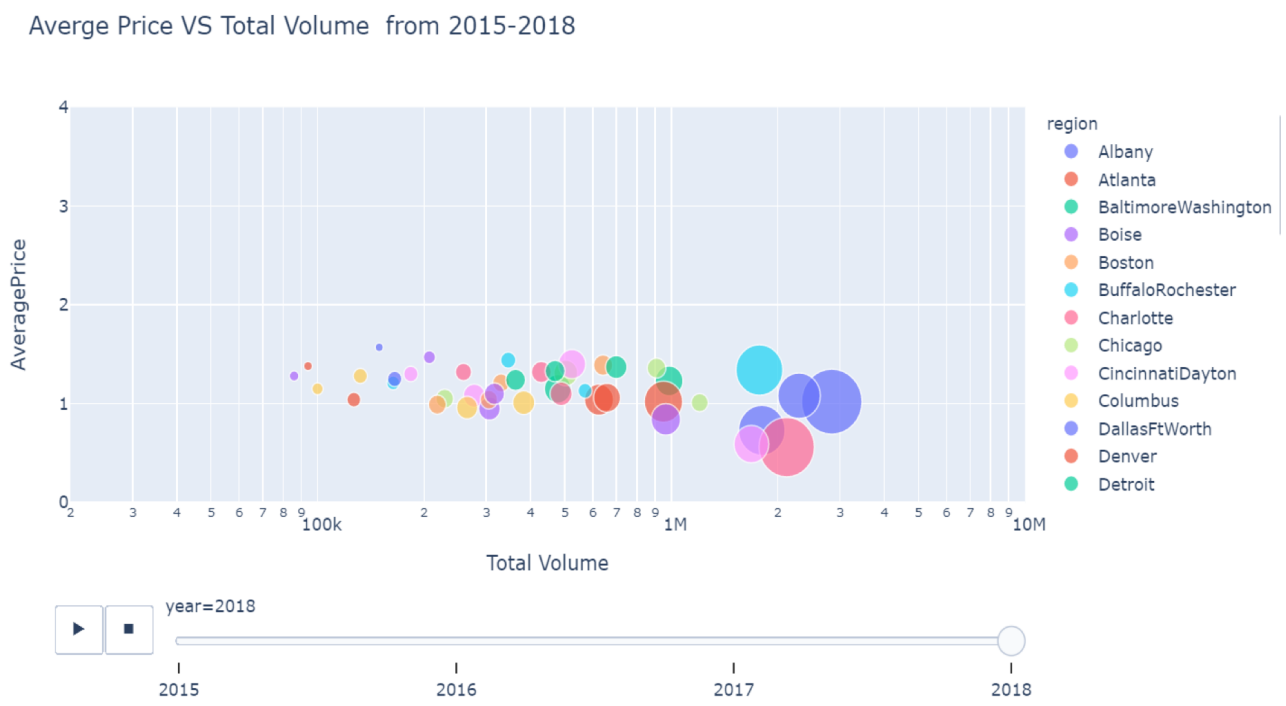
- Train on 2015-2017 data, test on true 2018 data.

Predictions

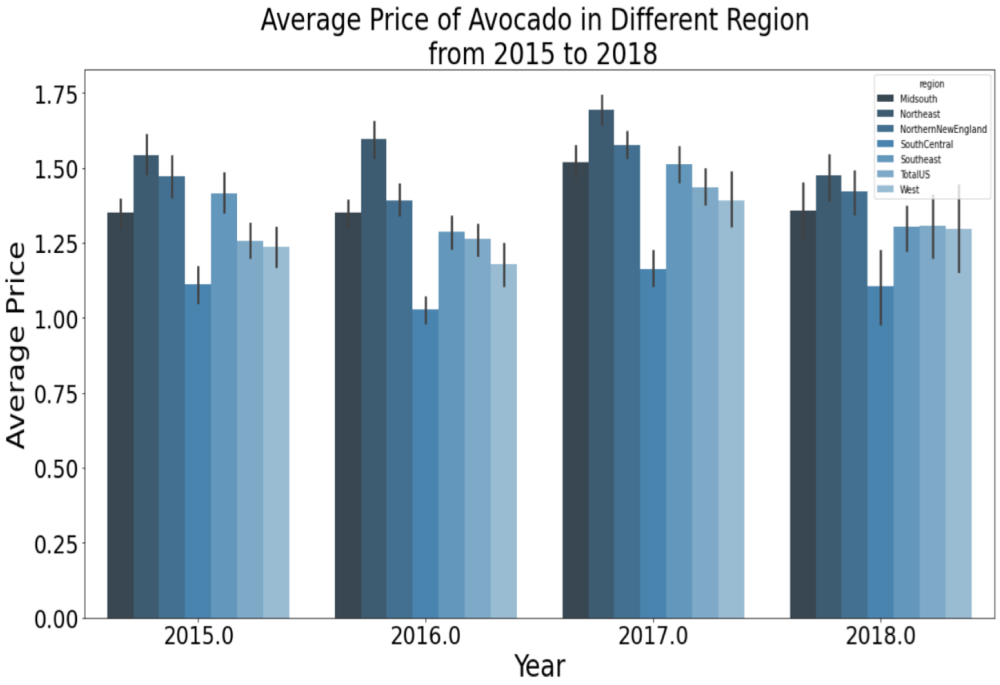
- Focus on the Volume parameter.

- Input adjusted avocado volume # from previous analysis to predict how prices need to be adjusted to meet new equilibrium in 2018

Price and Volume over Four Years



Price in Different Region

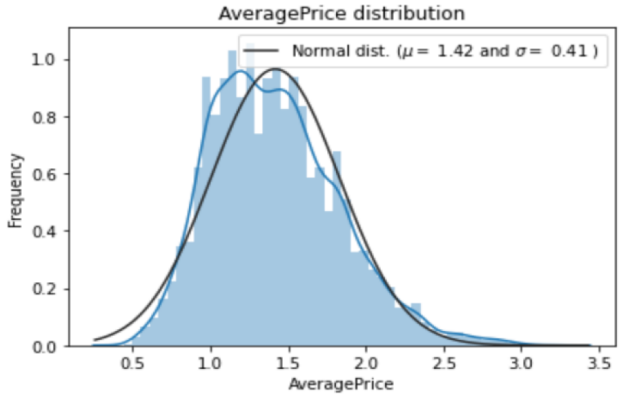


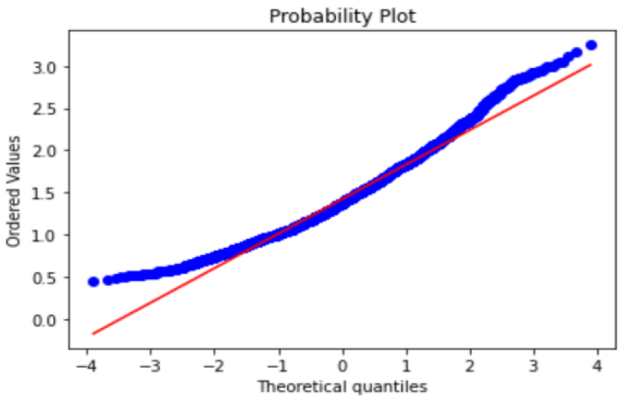
- Price in Northeast area is the highest among 4 years
- Price in SouthCentral area is lowest among 4 years
- In 2018, the price are becoming more close in those areas than before

Data Preprocessing

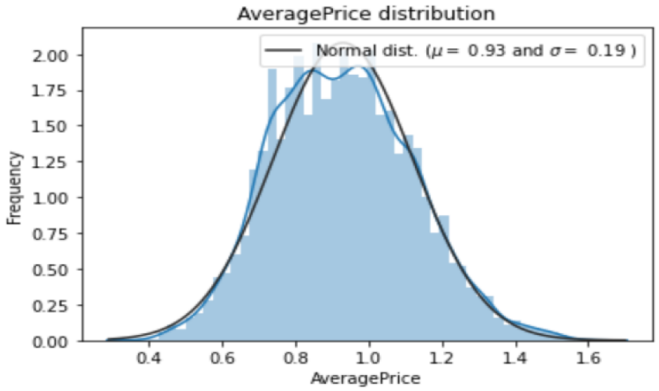
- There are only 3 missing value, so we just simply removed them from our data
- For categorical data, region, we transformed them into dummy variables
- Check the skew of all numerical features

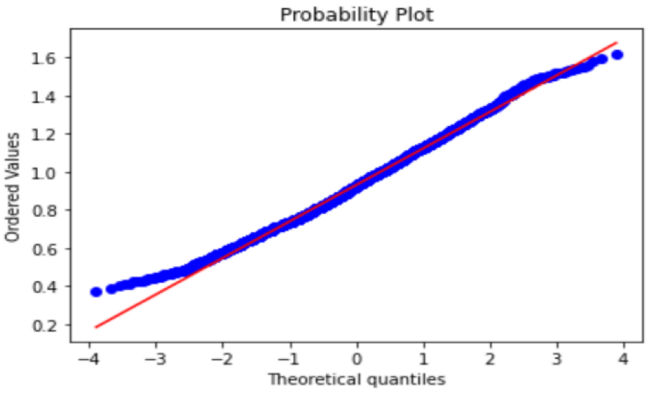
Box-Cox Transformation



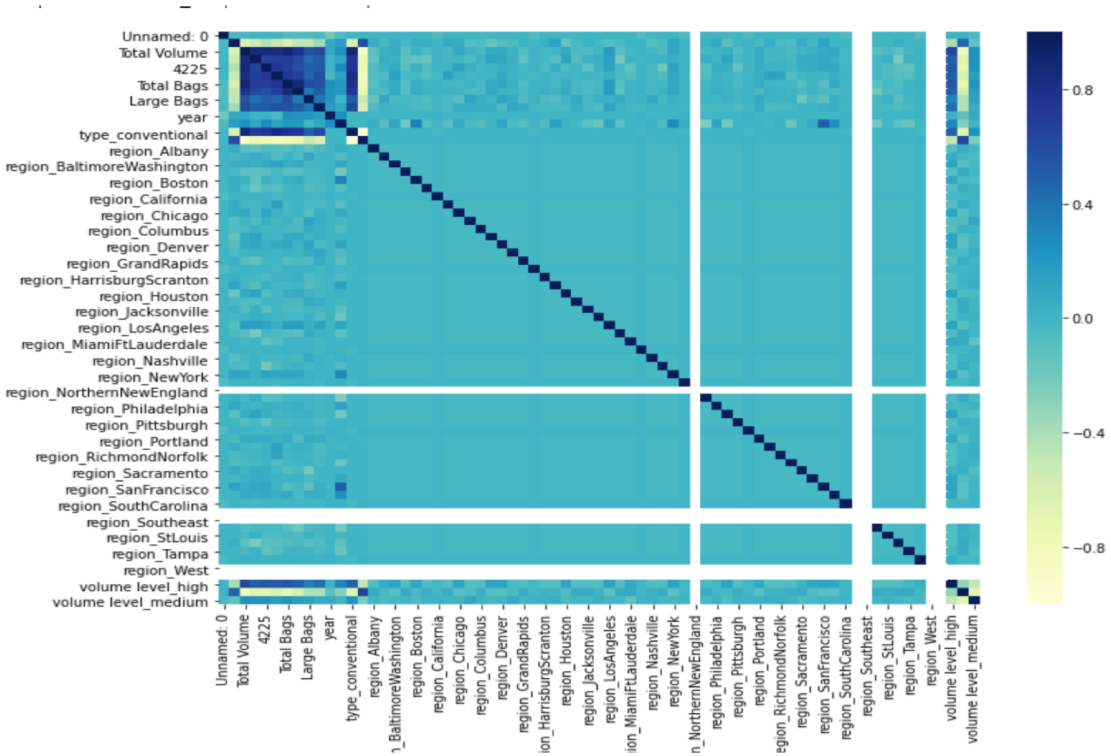


- 72 Skew Numerical features to Box Cox transform
- Fit the normal distribution





Correlation Heat Map

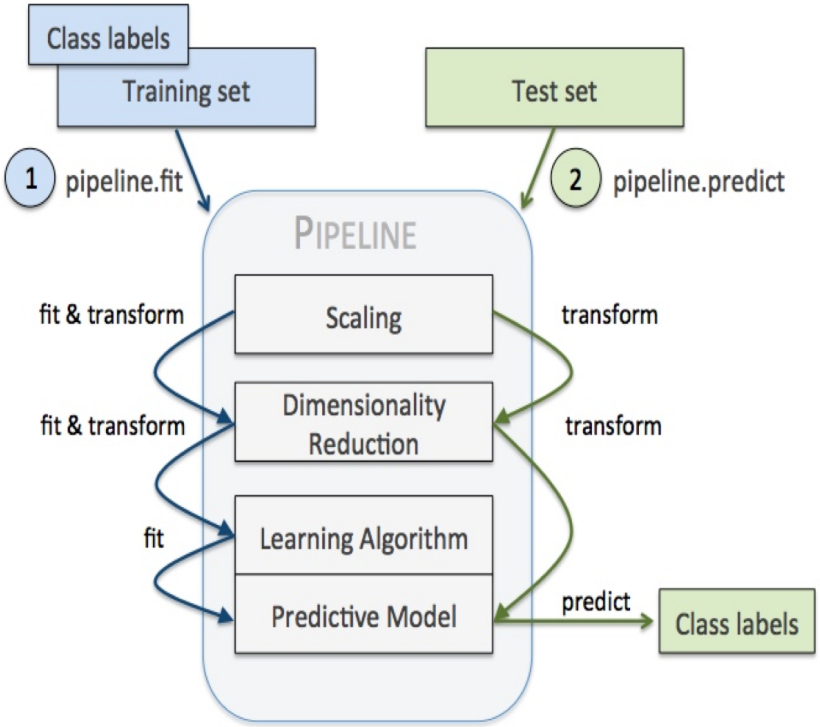


- Most of them don’t have high correlation
- Some variables are highly correlated
- It may indicate the multicollinearity problem for ols model

Model : Overview

- We pick the ordinary least square to approach the regression model
- To improve the performance of our model, we try three boost algorithm to optimize the model
- Compare each model to choose the best one
- Use that model to predict 2018 data and compare the results

Model: Construction

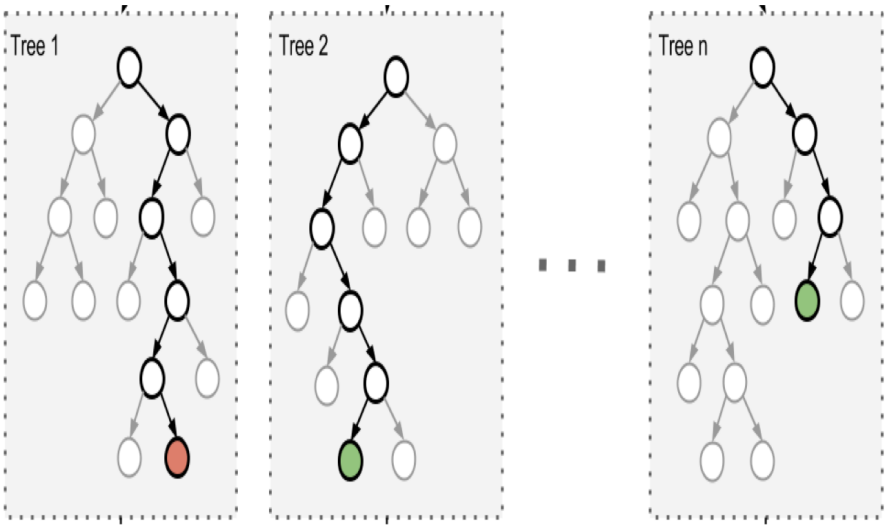


- Standardized data
- Use principal components analysis (PCA) to do dimensionality reduction
- Construct learning algorithm
- Predict model

Model: OLS

- Durbin-Watson Test: near 2, no autocorrelation
- Cook’s distance: 4 outliers
- Variance test: constant
- R-square: 0. 03 (no meaning, since we use the transformed model)
- MAE: 0.98

Model: Boost Overview



- We try three boost methods: Gradient Boost, XGBoost, LightGBM
- Use Grid Search Cross Validation to optimize the parameter of each boost model
- Pick the best one based on time consuming and MAE

Model: Comparison

Features	GBoost	XGBoost	LightGBM
Time	31.2 min	13.36 min	17.27 min
Huber Loss	0.67	0.82	0.87
Best Parameter	min_sample_split:0.5, max_features: sqrt	colsample_bytree: 1.0, XGB__learning_rate: 0.07	learning_rate=0.1, n_estimators=1000