Conclusion

May 24, 2019

1 House Prices: Advanced Regression Techniques

1.1 Abstract

Predict sales prices with advanced regression techniques and practice feature engineering.

1.2 Introduction

This is our team's final project for Applied Machine Learning course. This team is consisted of Wenyu Fan, Tamer Ibranhim and Yaohua Chang.

1.3 Background

This dataset was constructed by Dean De Cock for use in Data Science education. It is viewed as a modern alternative to the Boston Housing dataset.

This dataset iseasy to understand so that we can put most time on studying and applying machine learning techniques rather than trying to understand it.

In addition, The community concerning this dataset is large so that we can study lots of brilliant ideas from there.

1.4 Data

Dataset from Kaggele:
https://www.kaggle.com/c/house-prices-advanced-regression-techniques

With 79 explanatory variables describing (almost) every aspect of residential homes in Ames, Iowa, this project challenges us to predict the final price of each home.

1.5 Method

1.5.1 Data Preprocessing

- Dealing with missing values
  - Filling LotFrontage NaN values using a linear regression model.
- Dealing with zeros
  Adding dummy variables for features with a lot of zeros to improve model fits.
- Encoding categorical variables

- Creating new features
  - Creation of features for different basement finish types.
- transform target variable into normal distribution
- Remove outliers

1.5.2 Exploratory Data Analysis

- Distribution of SalePrice for all the features
- Correlation between the features and SalePrice (or each other)

1.5.3 Feature Selection

As we can see there are many features in this dataset.Random forest can be used to do feature selection to remove least important features.

1.5.4 Fit and Optimise Models

To avoid repeating code I create a function to perform the fitting process for each model type I try. I use k-fold cross-validation to reduce the chances of overfitting the training data - specifically 5-fold cross-validation repeated 5 times.

The parameters of each model are optimised using a grid search, and the function returns the best model found and some stats on the model performance.

The models we tried are: * Linear Regression: * sklearn.linear_model.Ridge * sklearn.linear_model.Lasso * sklearn.linear_model.ElasticNet

- Support Vector Machines:
  - sklearn.svm.SVR
- Tree Based:
  - sklearn.ensemble.RandomForestRegressor
- Neural Networks:
  - keras

1.6 Evaluation

1.6.1 Compare Model with actual sale price

Model	R square (R2)	Mean absolute error (MAS)
Lasso	0.883223	19150.15
Ridge	0.95590900	11291.2240
Elastic Net	0.91388614	15122.4778
SVM	0.860785143	18799.0263
Neural Networks	0.868273	19768.23857
Random Forest	0.98622979	5580.97214

1.6.2 Compare Model with log of the sale price

Model	R square (R2)	Mean absolute error (MAS)
Lasso	0.902168	0.124385
Ridge	0.96326875	0.05546042
Elastic Net	0.9473058	0.07037367
SVM	0.932746	0.091442938
Random Forest	0.9877892	0.03351962

1.7 Conclusion

- We applied sveral algorithms in this project including Lasso, Ridge, Elastic Net, SVM, Neural Networks, Random Forest.
- After removing outliers which identified by Ridge, the MAS has slightly been decreased.
- Using log of the sale price as target value to train models, the R2 has slightly been increased.
- Random Forest perfroms best among all models, then linear regression models, then SVM.
- Neural Network gives a very bad results, we should do feature engineering use some advanced tools to imporove the NN performance.