Predict the price of Renfe AVE ticket

EE 660 Course Project

Project Type: Design a system based on real-world data

Number of student authors: 1

Yun-Te Yeh, yunteyeh@usc.edu

Date:12/8/2019

1. Abstract

Our purpose is to predict the price of Renfe AVE ticket, However, there are some problems such as time series, missing data and categorical before training data, so we use linear regression, Random Forest regression and Gradient Boosting Regression to find the best result.

2. Introduction

2.1. Problem Type, Statement and Goals

Our problem is to predict the price of Renfe AVE ticket, which is regression. In fact, the ticket of train is basically influenced by distance and time, but perhaps there are other reasons which affect the price of ticket. Moreover, there are no particle rules which compute the price. Thus, we want to understand how company predict the price of tickets. Furthermore, there are some problems in the dataset. First, the features of inser_date, start_date and end_date are time series. Second, the features of origin, destination, train type, train_class and fare are categorical. Third, there are missing data in the price, train_class and fare. Final, the dataset which is too big is 7644664 rows x 52 columns, so we have to spend a lot of time running the code. Therefore, our goal is to solve these problems and obtain the high accuracy of prediction.

2.2. Our Prior and Related Work (Mandatory)

Prior and Related Work - None.

2.3. Overview of Our Approach

Because of regression problem, we use linear regression, random forest regression and gradient boosting regression. At the beginning, we have to analysis each feature and select the features which are important. After preprocessing, we obtain the R square, mean square error, train score and learning curve to compare the models. Finally, we can compute the best model.

3. Implementation

3.1. Data Set

There are 9 features in the dataset in table.1. The first feature is insert_date which is the date time when the price was inserted in the dataset. The second feature and third feature are origin and destination, which are the cities from beginning to destination. The fourth and fifth feature are date time from beginning to destination. The sixth feature is train_type which is different types of trains' name. The seventh feature is price which is the price of ticket. The eighth feature is train_class which is ticket class such as business, tourist and so on. The ninth feature is fare which is ticket fare.

Insert_ date	**origin**	**destina tion**	**start_d ate**	**end_d ate**	**train_t ype**	**price**	**train_cl ass**	**fare**
Time series	Catego rical	Categor ical	Time series	Time series	Catego rical	numer ical	Catego rical	Catego rical

Table.1 9 features

3.2. Preprocessing, Feature Extraction, Dimensionality Adjustment

There are three problems in the dataset such as time series, missing data and categorical. Thus, first, there are missing data in the features of price, train_class and fare. We drop the data which has empty value because there are at least 70000000x52 data in the dataset, which doesn't affect the result a lot. Moreover, for the price part, we fill the mean of price. Second, we divide the feature of insert_date to the features of month, day, hour, minute and second. We don't need the feature of year because all features of year are the same in table.2. [1]

Insert_date maximum date time	2019-04-11 21:49:46
Insert_date minimum date time	2019-05-09 21:19:16

Table.2 maximum and minimum date time

Third, the features of start_date and end_date are time series, so we change the datetime to hour, which we calculate between start_date and end_date

to represent the features of start_date and end_date because all of people spent less than one day taking the train. Moreover, if we divided the features of datetime, it would increase the features which waste a lot of time to train the model. Final, the features of origin, destination, train_type, train_class and fare are categorical, so our solution is one-hot encoding from pandas. For one-hot encoding method, if there are 5 units in the categorical feature, we should add 5 new features to represent the original feature, so it will increase lots of features. [2]

3.3. Dataset Methodology

My dataset is renfe.csv. In the processing, we use train_test_split from scikit-learn to decrease the sample because the dataset is too large, so we use 10% of dataset to be trained. Moreover, we split two datasets and Training set is 90% of dataset and testing data is 10% of dataset.

When we use any models, we use training set to fit model and use testing set to obtain the Mean square error, R square and so on. At the end, we compute the learning curve to select the best model. Before using learning curve, we use ShuffleSplit from scikit-learn to split 10 datasets in order to find the best model in figure.1.

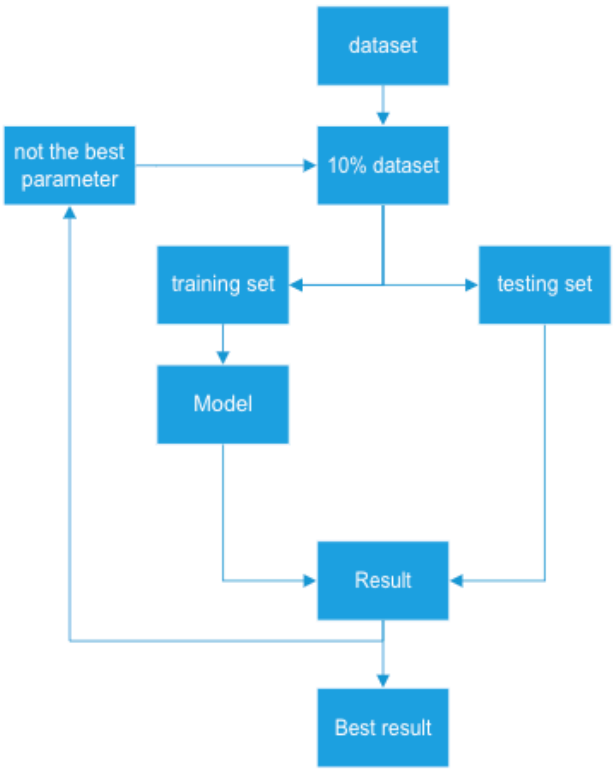

figure.1 Flow chart

3.4. Training Process

- Linear Regression

Linear regression is basic solution to solve regression problem. Also, it is a linear which explain the relationship between one dependent variable and one or more independent variables.

First step, we solve the problems of time series, missing data and categorical.

Second step, we split training data to testing set and training set. Training set is 90% of dataset and testing data is 10% of dataset.

Third step, we observe the R square to find the best parameter. However, the polynomial regression has to spend at least 6 hours because of large data, so we choose linear regression.

Final step, running the model to obtain the result in table.3 and learning curve in figure.2.

MSE	129.33
R square	0.79046
Train score	0.78597

Table.3 The report of linear regression

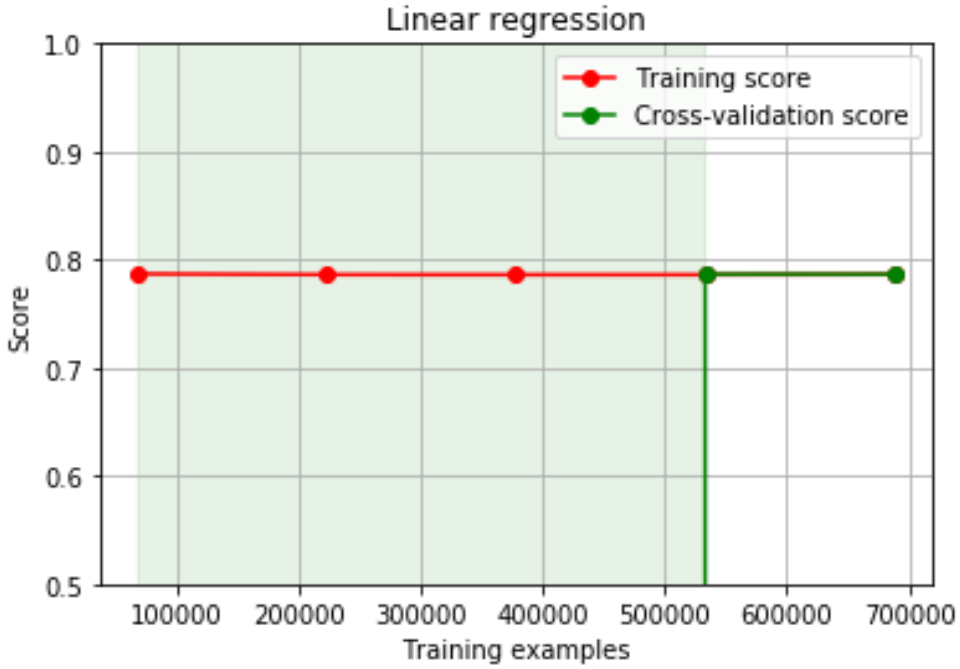

Figure.2 learning curve of learning regression

- Random Forest Regression

Random Forest is an ensemble learning method for classification and regression, which is based on decision trees. The method of decision trees is to divide the feature to two different results for each epoch. Moreover, this result is divided to two different new results again and again until it cannot be divided. Thus, Random forest consists of many decision trees, so if there are lots of trees, it means we can obtain the higher accuracy.

First step, we solve the problems of time series, missing data and categorical.

Second step, we split training data to testing set and training set. Training set is 90% of dataset and testing data is 10% of dataset.

Third step, we observe the R square to find the best parameter. The parameter of random forest regression is n_estimators which is 10, 50 and 100. At the end, the best parameter is n_estimators = 100.

Final step, running the model to obtain the result in table.4 and learning curve in figure.3.

	10	50	100
MSE	97.49	91.49	90.71
R square	0.8420	0.8517	0.8530
Train score	0.9644	0.9704	0.9712

Table.4 The report of Random Forest Regression

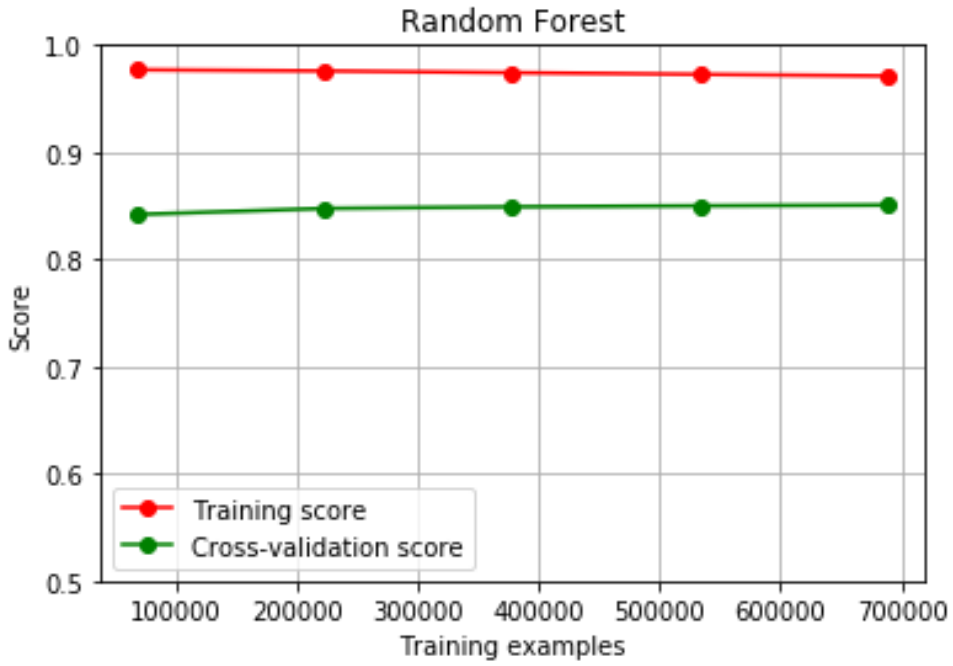

Figure.3 The learning curve of Random Forest Regression

- Gradient Boosting Regression

Gradient Boosting regression is based on Boosting and decision trees.

First step, we solve the problems of time series, missing data and categorical.

Second step, we split training data to testing set and training set. Training set is 90% of dataset and testing data is 10% of dataset.

Third step, we observe the R square to find the best parameter. The parameter of random forest regression is n_estimators which is 10, 50 and 100. At the end, the best parameter is n_estimators = 100.

Final step, running the model to obtain the result in table.5 and learning curve in figure.4.

	10	50	100
MSE	248.60	113.43	106.17
R square	0.5972	0.8162	0.8279
Train score	0.5964	0.8135	0.8246

Table.5 The report of Gradient Boosting Regression

Figure.4 the learning curve of Gradient Boosting Regression

3.5. Model Selection and Comparison of Results

For the learning curve part, the learning curve of learning regression is underfitting and the learning curve of Gradient Boosting regression is overfitting. Moreover, compare all the best parameter model in table.6. Thus, the best model is Random Forest regression.

	Linear regression	Random Forest regression	Gradient Boosting regression
MSE	129.33	90.71	106.17
R square	0.79046	0.8530	0.8279
Train score	0.78597	0.9712	0.8246

Table.6 comparison of all models

4. Final Results and Interpretation

The best model is Random Forest Regression and the parameter is n_estimators = 100. However, this is not the best one. First, in the preprocessing, we can change the

feature of origin and destination to value of distances. Also, in order to decrease more features, we can compare different kinds of method of reducing dimension to obtain the best result. Second, the parameters of Random Forest are at least 6 different kinds of parameter, but we only choose one parameter. Thus, we still need to find the best parameter.

5. Summary and conclusions

In the processing, the preprocessing is most important part in this project especially the feature selection. For example, if we find the feature which doesn't affect the label features, we can drop it because the more features we drop, the more time we save. As a result, preprocessing plays an important role on this project whatever the model is.

6. References

[1] "A Very Extensive Renfe analysis," July 2019. Available: https://www.kaggle.com/ratan123/a-very-extensive-renfe-analysis#2.Importing-data-.

[2] "Spanish Train Ticket Price Prediction-Renfe," July 2019. Available: https://www.kaggle.com/scsaurabh/spanish-train-ticket-price-prediction-renfe#2.-Bivariate-Analysis.