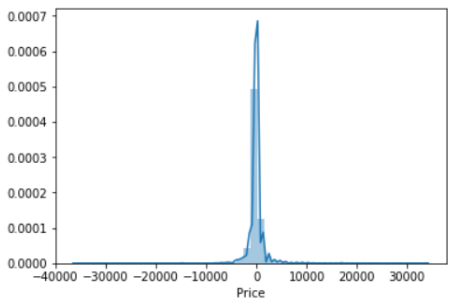
Project Name:

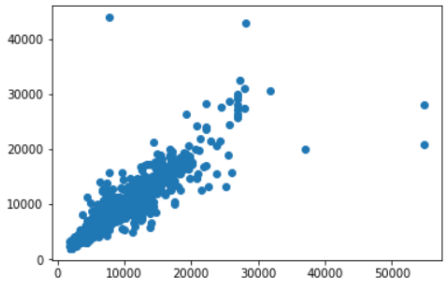
Analysis Feature-Engineering Feature-Selection Model-Building Hyper-Parameter Visualization Flight-Ticket-Price-Dataset

Table of Contents

Demo




```
In [63]: plt.scatter(y_test,y_pred)
Out[63]: <matplotlib.collections.PathCollection at 0x247a89c4ef0>
```



Overview

This is diving into Feature Engineering and Feature Selection Concept.
Feature Engineering is the act of extracting important features from raw data and transforming them into formats that are suitable for machine learning. Feature Engineering is a data preparation process.
Feature Selection: All features aren't equal. It is all about selecting a small subset of features from a large pool of features. We select those attributes which best explain the relationship of an independent variable with the target variable. There are certain features which are more important than other features to the accuracy of the model. It is different from dimensionality reduction because the dimensionality reduction method does so by combining existing attributes, whereas the feature selection method includes or excludes those features.
The methods of Feature Selection are Chi-squared test, correlation coefficient scores, LASSO, Ridge regression etc.
Flight ticket prices can be something hard to guess, today we might see a price, check out the price of the same flight tomorrow, it will be a different story. We might have often heard travellers saying that flight ticket prices are so unpredictable. Huh! Here we take on the challenge! As data scientists, we are going to prove that given the right data anything can be predicted. Here you will be provided with prices of flight tickets for various airlines between the months of March and June of 2019 and between various cities.
This repository contains the code for Feature Selection and Feature Importance using python's various libraries.
It used numpy, pandas, matplotlib, seaborn and sklearn libraries.
These libraries help to perform individually one particular functionality.
Numpy is used for working with arrays. It stands for Numerical Python.
Pandas objects rely heavily on Numpy objects.
Matplotlib is a plotting library.
Seaborn is data visualization library based on matplotlib.
Sklearn has 100 to 200 models.
The purpose of creating this repository is to gain insights into how to extract and transform data.
These python libraries raised knowledge in discovering these libraries with practical use of it.
It leads to growth in my ML repository.
This above screenshot will help you to understand flow of output.

Motivation

The reason behind building this is, of course all businesses want that their final result should be accurate therefore need to learn Feature Engineering which has a major influence on the performance of machine learning models and even the quality of insights derived during EDA. The models take features as input. A feature is generally a numeric representation of an aspect of real-world phenomena or data. My job as a Data Scientist is to find a clear path to the end goal of insights. Feature engineering helps extract information from raw data, i.e., it has created

a lot of features. This means we need to find the main features of the whole lot. This is also known as the Curse of Dimensionality. I sometimes try to relate things of both the world that is IT and Life Philosophy, In a normal life also it is not possible for one person to study and grab all degrees in the world or all the knowledge in the world so people extract degrees as per their interests so similar analogy is here I have to extract relevant features from raw data as per requirement. Through feature engineering, you can isolate key information, highlight patterns, and bring in domain expertise. If I wish to convey my point to buyer than this can be helpful as I can highlight relevant pattern and isolate key information, which is manipulation of data and hence of decision which is a tricky strategy which is what is call for apart from just a normal coder. I believe that, it's not all always about coding but also about convincing or confusing or ponder them to imagination or influence which is also a must which can be achieved through Feature Engineering. Feature engineering, also known as feature creation, is the process of constructing new features from existing data to train a machine learning model. This step can be more important than the actual model used because a machine learning algorithm only learns from the data, we give it, and creating features that are relevant to a task is absolutely crucial. Statistically, 8 out of 10 people does not like to perform Feature Engineering as it is time-consuming but what fascinates me is, it is art of direction. You know that, master plan can make it to millions too. It enables the machine learning algorithm to train faster. It reduces the complexity of a model and makes it easier to interpret. It improves the accuracy of a model if the right subset is chosen. Feature Selection is the process where you automatically or manually select those features which contribute most to your prediction variable or output in which you are interested in. Having irrelevant features in your data can decrease the accuracy of the models. Hence, I continue to gain knowledge while practicing the same and spread literary wings in tech-heaven.

Technical Aspect

Size of training set: 10683 records
Size of test set: 2671 records
Numpy contains a multi-dimensional array and matrix data structures. It works with the numerical data. Numpy is faster because is densely packed in memory due to its homogeneous type. It also frees the memory faster.
Pandas module mainly works with the tabular data. It contains Data Frame and Series. Pandas is 18 to 20 times slower than Numpy. Pandas is seriously a game changer when it comes to cleaning, transforming, manipulating and analyzing data.
Matplotlib is used for EDA. Visualization of graphs helps to understand data in better way than numbers in table format. Matplotlib is mainly deployed for basic plotting. It consists of bars, pies, lines, scatter plots and so on. Inline command display visualization inline within frontends like in Jupyter Notebook, directly below the code cell that produced it.
Seaborn provides a high-level interface for drawing attractive and informative statistical graphics. It provides a variety of visualization patterns and visualize random distributions.

Sklearn is known as scikit learn. It provides many ML libraries and algorithms for it. It provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python.

Encode categorical features using an ordinal encoding scheme. Encode categorical features as a one-hot numeric array. LabelEncoder can be used to normalize labels. It can also be used to transform non-numerical labels to numerical labels. Label Encoding refers to converting the labels into numeric form so as to convert it into the machine-readable form. Machine learning algorithms can then decide in a better way on how those labels must be operated. It is an important pre-processing step for the structured dataset in supervised learning.

Reason for doing Hyper-Parameter Tuning is, Setting the correct combination of hyperparameters is the only way to extract the maximum performance out of models.

The most important arguments in RandomizedSearchCV are n_iter, which controls the number of different combinations to try, and cv which is the number of folds to use for cross validation. RandomizedSearchCV implements a “fit” and a “score” method. It also implements “predict”, “predict_proba”, “decision_function”, “transform” and “inverse_transform” if they are implemented in the estimator used. If at least one parameter is given as a distribution, sampling with replacement is used.

In the realm of machine learning, the random forest regression algorithm can be more suitable for regression problems than other common and popular algorithms. Below are a few cases where you’d likely prefer a random forest algorithm over other regression algorithms:

1. There are non-linear or complex relationships between features and labels.
2. You need a model that’s robust, meaning its dependence on the noise in the training set is limited. The random forest algorithm is more robust than a single decision tree, as it uses a set of uncorrelated decision trees.

3. If your other linear model implementations are suffering from overfitting, you may want to use a random forest.

Need to train_test_split - Using the same dataset for both training and testing leaves room for miscalculations, thus increases the chances of inaccurate predictions.
The train_test_split function allows you to break a dataset with ease while pursuing an ideal model. Also, keep in mind that your model should not be overfitting or underfitting.

Installation

Using intel core i5 9th generation with NVIDIA GFORECE GTX1650.
Windows 10 Environment Used.
Already Installed Anaconda Navigator for Python 3.x
The Code is written in Python 3.8.
If you don't have Python installed then please install Anaconda Navigator from its official site.
If you are using a lower version of Python you can upgrade using the pip package, ensuring you have the latest version of pip, *python* -m pip install --*upgrade pip and press Enter*.

Run/How to Use/Steps

Keep your internet connection on while running or accessing files and throughout too.
Follow this when you want to perform from scratch.

Open Anaconda Prompt, Perform the following steps:

cd <PATH>
pip install numpy
pip install pandas
pip install matplotlib
pip install seaborn
pip install sklearn

You can also create requirement.txt file as, pip freeze > requirements.txt

run files.

Follow this when you want to just perform on local machine.
Download ZIP File.
Right-Click on ZIP file in download section and select Extract file option, which will unzip file.
Move unzip folder to desired folder/location be it D drive or desktop etc.
Open Anaconda Prompt, write cd <PATH> and press Enter.
eg: cd C:\Users\Monica\Desktop\Projects\Python Projects 1\15)Feature_Engineering+Selection+Importance\ Feature_Engineering_3Different_Datasets\Analysis_FE_FS_ModelBuilding_HyperParameter_Visualization_FlightTicketPricePredictionDataset

In Anconda Prompt, pip install -r requirements.txt to install all packages.
Open in Jupyter Notebook, <filename>.ipynb
That is,

Open in Jupyter Notebook,
1)Feature_Engineering_and_Feature_Selection_for_HousePricePredictionDataset.ipynb

This takes Test_set.xlsx and Train_set.xlsx file as input dataset.

Please be careful with spellings or numbers while typing filename and easier is just copy filename and then run it to avoid any silly errors.
Note: cd <PATH>
[Go to Folder where file is. Select the path from top and right-click and select copy option and paste it next to cd one space <path> and press enter, then you can access all files of that folder] [cd means change directory]

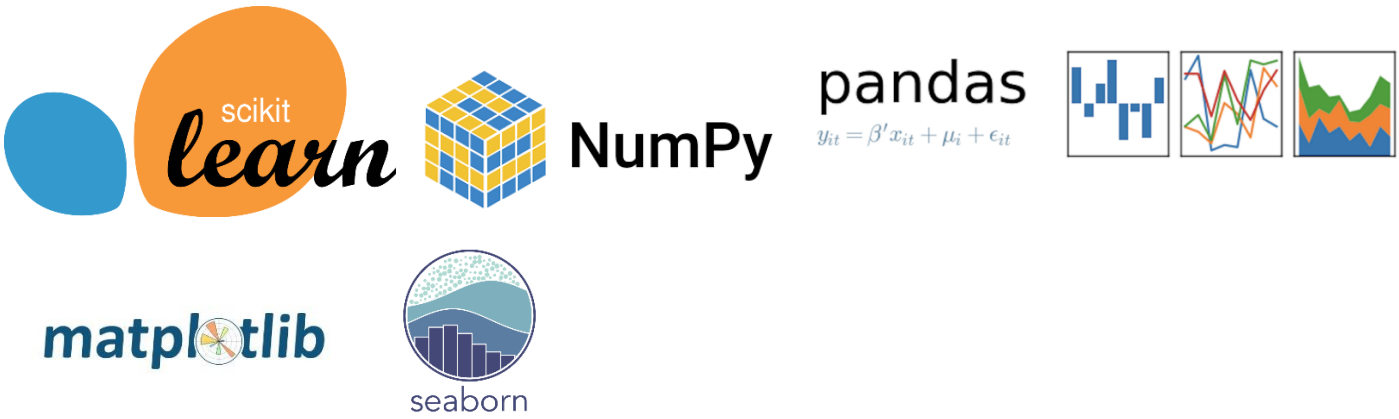
Directory Tree/Structure of Project

Folder: 15)Feature_Engineering+Selection+Importance >
Feature_Engineering_3Different_Datasets >
Analysis_FE_FS_ModelBuilding_HyperParameter_Visualization_FlightTicketPricePredictionDataset
1)Feature_Engineering_and_Feature_Selection_for_HousePricePredictionDataset.ipynb

To Do/Future Scope

Can do more data cleaning.

Technologies Used/System Requirements/Tech Stack



Credits

Krish Naik Channel

https://towardsdatascience.com/random-forest-in-python-24d0893d51c0

https://heartbeat.fritz.ai/random-forest-regression-in-python-using-scikit-learn-9e9b147e2153