Machine Learning Engineer Nanodegree

Capstone Proposal

Dmytro Illarionov

January 2nd, 2019

Proposal

Domain Background

Building a model which will calculate and predict the prices of real estate items based on different information about them is a classical regression problem in machine learning. The real estate world deals with a lot of money and the prediction of house prices is clearly a much needed and important part of it. Both the seller and the buyer are interested in the accurate prediction of the prices.
In this project i am going to explore the data from kaggle.com [1] that contains information about different apartments in Melbourne and find a model which will calculate and predict the price based on information such as number of rooms, distance from central business district etc.
For me as a beginning data scientist and machine learning engineer this project is an opportunity to apply knowledge in data analysis and preprocessing, feature engineering, using of machine learning algorithms obtained the course.

Problem Statement

The challenge of the project is to solve a regression problem namely to build a model which will predict the price of different real estate items based on a lot of different explanatory features provided in the dataset.

Datasets and Inputs

The data bases on publicly available results posted on domain.com.au. The data for the project provides 14 explanatory variables describing different aspects of 13580 apartments.

Input variables:

1. Rooms: Number of Rooms
2. Method: S - property sold; SP - property sold prior; PI - property passed in; PN - sold prior not disclosed; SN - sold not disclosed; NB - no bid; VB - vendor bid; W - withdrawn prior to auction; SA - sold after auction; SS - sold after auction price not disclosed. N/A - price or highest bid not available.
3. Type: br - bedroom(s); h - house,cottage,villa, semi,terrace; u - unit, duplex; t - townhouse; dev site - development site; o res - other residential.

4. SellerG: Real Estate Agent
5. Date: Date sold
6. Distance: Distance from CBD
7. Regionname: General Region (West, North West, North, North east ...etc)
8. Propertycount: Number of properties that exist in the suburb
9. Bedroom : Scraped # of Bedrooms (from different source)
10. Bathroom: Number of Bathrooms
11. Car: Number of carspots
12. Landsize: Land Size
13. BuildingArea: Building Size
14. CouncilArea: Governing council for the area

Output variable:

1. Price: Price in dollars

The data set represents mixed data with numerical and categorical values. There are also several missing values in some attributes. These missing values can be treated using deletion or imputation techniques [3].

Solution Statement

After analyzing, cleaning and preparing the data to fit it into the algorithms, the predicting problem will be solved in two ways. The first solution consist in exploring regression models using python's scikit learn library (e.g. DecisionTreeRegressor, KNeighborsRegressor, RandomForestRegressor etc.), choosing one of them, which performed better and improve the results through tuning model's hyperparameters . The prediction model based on neural network will represent the second solution. For creating a neural net the Keras library will be used.

Benchmark Model

LinearRegression model will be chosen as the benchmark model. The best regression model from scikit learn library and the neural net will be compared to the initial prediction result of LinerRegression model.

Evaluation Metrics

To evaluate the performance of regression models we will be using the root-mean-square-error (RMSE) [4] and r2_score also known as coefficient of determination [5].
Root-mean-square-error is a frequently used measure of the differences between values predicted by a model or an estimator and the values observed.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\widehat{y}_i - y_i)^2}$$

n - number of observations,
$\widehat{y}$ - predicted values,
y_i - true values.
Important to mention that RMSE error is in the same units as the data outcome. It means on this case it will be in units of output variable. Low RMSE values are desired. r2_score is the proportion of the

variance in the dependent variable that is predictable from the independent variable(s). Best possible score is 1.0

Project Design

All calculations, visualizations and data analysis will be performed in jupyter notebook. The main steps of the project will be the following:

1. Exploring the data
    a. Loading libraries
    b. Loading data
    c. Statistical information about the data
2. Data preprocessing
    a. Handle missing data
    b. Calculating unique values
    c. Searching for duplicates
    d. Handling missing values
    e. Investigating of cross correlation
    f. Analyzing dependencies between features and target variable
    g. Handling outliers
    h. Data scaling/normalization (if needed)
    i. Label encoding, one hot encoding (if needed)
    j. Logarithmic transformation (if needed)
    k. Investigating feature importance
    l. Extracting features and target variables
    m. Split the data in training and testing set
    n. Evaluating benchmark model (LinearRegression)
3. Evaluate regression model using scikit learn
    a. Build models
    b. Train models on training set
    c. Evaluate models on testing set
    d. Choosing the best model
    e. Tuning the hyperparameters
    f. Evaluate the best model after tuning the hyperparameters on testing set
4. Building neural net using Keras
    a. Splitting training set using K-Fold
    b. Designing a neural net (choosing the number of layers, neurons, activation function etc.)
    c. Training the neural net
    d. Evaluating the neural net performance on testing set
5. Conclusions and performance comparison of regression model and neural net with benchmark model

References

[1] https://www.kaggle.com/dansbecker/melbourne-housing-snapshot/data
[2] https://www.kaggle.com/anthonypino/melbourne-housing-marketn
[3] https://www.theanalysisfactor.com/seven-ways-to-make-up-data-common-methods-to-imputing-missing-data/
[4] https://en.wikipedia.org/wiki/Root-mean-square_deviation
[5] https://en.wikipedia.org/wiki/Coefficient_of_determination