Machine Learning Engineer Nanodegree
Capstone Proposal

Hugo Saito

February 21st, 2018

Capstone Project Justification

My proposal for Capstone project of Udacity Machine Learning Nanodegree Program is a Kaggle competition with *Getting Started* level. All information about this competition can be found HERE. That is excellent competition for Machine Learning students who have a basic knowledge in this field and need an entry level problem to practice and improve your skills about regression techniques and feature engineering. Therefore, that is perfect as my Capstone Project.

House Prices: Advanced Regression Techniques

Domain Background

Nowadays, Machine Learning(ML) techniques are applied in many different fields. Fast-forwards in the last decade placed Machine Learning as the code of many high-tech products: ranking web search (Google), recommending videos (YouTube), driving car (Tesla), predicting stock prices, etc. Almost all field could (or are) use ML techniques to improve some desired goal, among these a potential field is the housing market.

The real estate agents have the challenging work of evaluate the price of house, but it is so difficult because each house is different and have many features to be evaluated. In addition, all humans suffer influence by sentimental state, family problems, stress levels and diverse others type of problems that probably will influence the result of the agents' evaluations. Thus, housing market is a good field to apply Machine Learning techniques for predicting house price.

Problem Statement

The housing market is highly competitive. A property can be evaluated by different real estate agents and certainly all prices provided will be different from each other, because the

complexity of many characteristics makes the agents being prove to errors. So, use machine learning techniques looking for predict the correct sale price can be an effective way to find the best price for each house. The *correct price* means that based on the quantity and quality of many attributes of the properties, a learner algorithm will seek the fairest price for each house.

The goal is to create a learner algorithm using supervised learning methods, where the learner will use a label data set to extract the pattern and correlation between all house's features looking for find an efficient way to determine the price of a new house (new data point) based on their characteristics, where the price is the target variable. It's easy to observe the performance of the learner if I use just the features of some points that doesn't were used in the learning phase as a new point and compare the true result with the learner prediction.

Datasets and Inputs

The Ames Housing Data Set was synthesized by the professor Dean De Cock for an end of semester project for an undergraduate regression course and contains information from the individual houses sold in Ames, Iowa, United States from 2006 to 2010. It's a modernized and expanded version of the Boston House Data Set (70's). The features are information that a home buyer or a real estate agent need to know to evaluate a property, thus is perfect for the project proposal. This dataset should be used as a teacher for learner algorithm, because the model will use their labels points to extract the pattern, seeking to become an effective learner.

The data set was obtained from Kaggle Competition, where are divided into training and testing subsets. The full and original dataset can be found HERE.[3]

The data set consists of 2930 samples and 82 features of distinct types: 23 nominals, 23 ordinals, 14 discrete and 20 continuous. One of the continuous variable is the target variable (*SalePrice*) for this project. As the number of variables are large, the brief version of data description can be found HERE[2] and if you want the full description, click HERE[4]

Solution Statement

Firstly, the Ames Housing Data Set will be explored (simple visualizations, correlations) to gain a basic insight. So, I will prepare the data to be used by machine learning algorithms executing Data Cleaning, Feature Selection, Feature Engineering, Feature Scaling and Dimensionality Reduction when these steps are necessary.

After that, I will apply many supervised models seeking identify a short-list with the most promising and will fine-tune these models for improve their results into testing set, using an appropriate evaluation metric as comparator. Finally, I will try to apply Ensemble Learning method to verify if I the aggregate model result is better than the individual predictors.

Benchmark Model/Score

As this project is based on ***House Prices: Advanced Regression Techniques*** Kaggle competition, there are many submissions with different evaluation metric (Root Mean Squared Error) score. The Public Leaderboard exhibit a list of all submission scores and the first place (Dark Yoshi) in the competition achieved a score metric equal ZERO. For my Benchmark, I will try to reach the score approximately equal 0.14 or less because that's the median score of this competition.

Evaluation Metrics

The Kaggle's competition specified in the description (section about Evaluation) that the submissions will be evaluated on Root Mean Squared Error (RMSE) between the logarithm of the predicted value and the logarithm of the observed sales price. [4] So, I will use this metric to evaluate the performance of the models utilized.

OBS: The logarithm means that positive or negative error will affect the result equally.

The RMSE obtains a measure of the spread of the predicted values around that average. So, it gives an idea of how much error the learner makes in its prediction.

The math behind RMSE:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (y_{true} - y_{pred})^2}{n}}$$

where, y_{true} is the true value of target variable and y_{pred} is the prediction from the model and n is the number of sample that were tested.

Project Design

The workflow to develop the problem solution will be divided into six steps. Each one is explained below.

Explore the Data

Firstly, I will take a look at the Data Structure using *head()*, *info()* and *describe()* methods. To get a feel of the distribution about features, I will plot a histogram for each numerical attribute using *hist()* method.

After that, I will seek understand the correlation between the features plotting the correlation matrix using *heatmap()*. To get a feel about dispersion I will plot the Box Plot between the

target variable (*SalePrice*) and the other features using *boxplot()* method. Scatter matrix can be useful here to better understand the correlation.

Prepare the Data

In this step, I will try find noises in the data, the documentation suggests some outliers, but probably there are others. The data exhibits missing values and is necessary to treat them removing some feature with considerable number of missing values or filling missing values with median or mode in numerical or categorical features respectively.

After that, I will perform the feature scaling in the numerical features using Box-Cox test looking for reduce the data skewness. Lastly, I will try to apply dimensionality reduction using principal components analysis (PCA), seeking to reduce the computational cost from this large number of features, but I need to test to know if this step will improve my result.

Model selection

In this section, I will try many machine learning models used to regression problems:

Linear Regression (with Ridge and Lasso regularization); Support Vector Regression; Linear Support Vector Regression; Stochastic Gradient Descent; K Neighbors Regressor; Kernel Ridge Regression; Decision Tree Regressor;

The method Cross Validation Score with score parameter defined as Root Mean Squared Error will be used to evaluate and compare the result of each individual learner. Just the bests model will be fine-tuned in the next step.

Fine-Tune Promising Models

As the promising models were defined in the last step, now I will fine-tune the most promising models. The method to perform the fine-tune the hyper parameters will be Randomized Search CV configured according to the project requisites. Thereby, each individual model will find the best parameters configuration.

Ensemble Method

In this last step of coding, I will try to apply Ensemble Methods appropriated for regression problems. Tree most common type of ensemble methods are classified as:

- Averaging/Bagging: take several estimators and average their predictions. The combined result is often better than any individual result (variance is reduced). Examples: Random Forest Regression and Bagging Regressor.
- Boosting: "*involves incrementally building an ensemble by training each new model instance to emphasize the training instances that previous models mis-classified*." [6] Examples: AdaBoost Regressor, Gradient Boosting Regressor.

For determine the best ensemble method I will test the methods cited above.

Present Solution

Finally, I will discuss the result that were found. The final score and the all conclusion analysis will be report. Visualization that summarize the result will be provided to facilitate the understanding of the result.

References

[1] Kaggle Competition - House Prices: Advanced Regression Techniques

[2] Ames, Iowa: Alternative to the Boston Housing Data as an End of Semester Regression Project

[3] Ames Housing Dataset Description

[4] Kaggle Metric Evaluation

[5] Root Mean Square Error - Wikipedia

[6] Ensemble Learning - Wikipedia

[7] Ensemble Methods – Scikit Learn