Machine Learning Engineer Nanodegree

Capstone Proposal

German Rezzonico

January 14, 2016

House Prices: Advanced Regression Techniques Using Ensemble Method

1. Domain Background

Machine learning is the subfield of computer science that "gives computers the ability to learn without being explicitly programmed" (Arthur Samuel, 1959)[1]. There are different learning styles in machine learning: supervised learning (task of inferring a function from labeled training data)[2] and unsupervised machine (task of inferring a function to describe hidden structure from unlabeled data). Ensemble methods use multiple learning algorithms to obtain better predictive performance than could be obtained from any of these constituent learning algorithms alone.[3]

A real estate is "property consisting of land and the buildings on it".[4] A real estate appraisal, property valuation or land valuation is the process of developing an opinion of value for real property (usually market value)… and every property is unique..."[5]. "The real estate industry is a big business generating billions of dollars in revenue annually. In 2016 there were approximately 210,000 companies operating in the residential brokerage and management field, which generated $200 billion in revenue; there were 35,000 companies operating in the

[1] Phil Simon (March 18, 2013). Too Big to Ignore: The Business Case for Big Data. Wiley. p. 89. ISBN 978-1-118-63817-0.

[2] Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar (2012) *Foundations of Machine Learning*, The MIT Press ISBN 9780262018258.

[3] Opitz, D.; Maclin, R. (1999). "Popular ensemble methods: An empirical study". *Journal of Artificial Intelligence Research*. **11**: 169–198. doi:10.1613/jair.614.

[4] https://en.wikipedia.org/wiki/Real_estate

[5] https://en.wikipedia.org/wiki/Real_estate_appraisal

commercial brokerage and management field, generating $35 billion in revenue”.[6] Is it possible to train a model, based on real state data, to be used to make predictions about a home’s monetary value? Such model could be invaluable for someone like a real estate agent or for one of the many companies operating in the real state industry.

The Boston Housing Data Set (Harrison and Rubinfeld 1978)[7] have been publicly available for some years and many research works have been made. On the other hand, the Ames Housing dataset[8] compiled by Dean De Cock for use in data science education, is a modernized and expanded version of the often cited Boston Housing dataset. And a competition on Kaggle[9], the House Prices: Advanced Regression Techniques[10], using that dataset presents an incredible opportunity in using machine learning ensemble methods to predict the house prices using the Ames dataset.

1.1. Personal Motivation

Participating in a Kaggle competition for the first time, the House Prices: Advanced Regression Techniques, will allow me to use concepts learned in the Udacity Machine Learning Nanodegree, and go in depth through the implementation of ensemble methods. I will learn more about machine learning and be more prepared for future works, tasks and Kaggle competitions.

2. Problem Statement

In this problem we will predict, through supervised machine learning methods, housing prices using the Ames Housing dataset. Some or all of the 81 features presented in the following section may be used to train the model.[11]

To make these predictions, we will use some of the following regression techniques, such as: LinearRegression[12], DecisionTreeRegressor[13], SVR[14], ElasticNet[15], Lasso[16], Ridge[17], LassoLars [18], BayesianRidge[19], GradientBoostingRegressor[20], ExtraTreesRegressor[21], BaggingRegressor[22]

[6] https://www.franchisehelp.com/industry-reports/real-estate-franchise-industry-report/

[7] Harrison, D. and Rubinfeld, D. L. (1978). “Hedonic Housing Prices and the Demand for Clean Air,” Journal of Environmental Economics and Management, 5, 81-102.

[8] https://ww2.amstat.org/publications/jse/v19n3/decock.pdf

[9] https://www.kaggle.com

[10] https://www.kaggle.com/c/house-prices-advanced-regression-techniques

[11] 3. Datasets and Inputs

[12] http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html

[13] http://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeRegressor.html

[14] http://scikit-learn.org/stable/modules/generated/sklearn.svm.SVR.html

[15] http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.ElasticNet.html

[16] http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Lasso.html

[17] http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Ridge.html

[18] http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LassoLars.html

[19] http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.BayesianRidge.html

[20] http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingRegressor.html

[21] http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.ExtraTreesRegressor.html

[22] http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.BaggingRegressor.html

, AdaBoostRegressor[23], XGBRegressor[24]. After selecting the best models, according to their performances, an ensemble generation will be implemented. Then, to determine if any improvement was made, the performance metrics of the ensemble will be calculated and compared with the ones of the benchmark model.

3. Datasets and Inputs

The Ames Housing dataset[25] compiled by Dean De Cock for use in data science education, in particular the version available in Kaggle[26], will be used in this project. This dataset is divided into two parts: the train dataset has 1460 data points with 81 features each and the test dataset has 1459 data points with 80 features each. In the test dataset the feature ‘SalePrice’ have been removed, because this is the predicted feature. The dataset describes the sale of individual residential property in Ames, Iowa from 2006 to 2010. There are a large number of explanatory features (23 nominal, 23 ordinal, 14 discrete, and 20 continuous) involved in assessing home values. Below are the 81 features:

MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	LotConfig
Nominal	Nominal	Continuous	Continuous	Nominal	Nominal	Ordinal	Nominal	Ordinal	Nominal

LandSlope	Neighborhood	Condition1	Condition2	BldgType	HouseStyle	OverallQual	OverallCond	YearBuilt	YearRemodAdd
Ordinal	Nominal	Nominal	Nominal	Nominal	Nominal	Ordinal	Ordinal	Discrete	Discrete

RoofStyle	RoofMatl	Exterior1st	Exterior2nd	MasVnrType	MasVnrArea	ExterQual	ExterCond	Foundation	BsmtQual
Nominal	Nominal	Nominal	Nominal	Nominal	Continuous	Ordinal	Ordinal	Nominal	Ordinal

BsmtCond	BsmtExposure	BsmtFinType1	BsmtFinSF1	BsmtFinType2	BsmtFinSF2	BsmtUnfSF	TotalBsmtSF	Heating	HeatingQC
Ordinal	Ordinal	Ordinal	Continuous	Ordinal	Continuous	Continuous	Continuous	Nominal	Ordinal

CentralAir	Electrical	1stFlrSF	2ndFlrSF	LowQualFinSF	GrLivArea	BsmtFullBath	BsmtHalfBath	FullBath	HalfBath

[23] http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostRegressor.html
[24] http://xgboost.readthedocs.io/en/latest/
[25] http://www.amstat.org/publications/jse/v19n3/decock.pdf
[26] https://www.kaggle.com/c/house-prices-advanced-regression-techniques/data

Nominal	Ordinal	Continuous	Continuous	Continuous	Continuous	Discrete	Discrete	Discrete	Discrete

BedroomAbvGr	KitchenAbvGr	KitchenQual	TotRmsAbvGrd	Functional	Fireplaces	FireplaceQu	GarageType	GarageYrBlt	GarageFinish
Discrete	Discrete	Ordinal	Discrete	Ordinal	Discrete	Ordinal	Nominal	Discrete	Ordinal

GarageCars	GarageArea	GarageQual	GarageCond	PavedDrive	WoodDeckSF	OpenPorchSF	EnclosedPorch	3SsnPorch	ScreenPorch
Discrete	Continuous	Ordinal	Ordinal	Ordinal	Continuous	Continuous	Continuous	Continuous	Continuous

PoolArea	PoolQC	Fence	MiscFeature	MiscVal	MoSold	YrSold	SaleType	SaleCondition	SalePrice
Continuous	Ordinal	Ordinal	Nominal	Continuous	Discrete	Discrete	Nominal	Nominal	Continuous

The train dataset will be divided into a training and testing set, using ‘train_test_split’[27] from sklearn.cross_validation[28], to shuffle and split the features and gross data into the training and the testing sets.

3.1 Working with the data

The 20 continuous variables relate to various area dimensions for each observation. The 14 discrete variables typically quantify the number of items occurring within the house. There are a large number of categorical variables (23 nominal, 23 ordinal). They range from 2 to 28 classes. The nominal variables identify various types of dwellings, garages, materials, and environmental conditions. The ordinal variables rate various items within the property.

Entries with missing data will be remove or replace with some arbitrary value. Some variables may be dropped from the dataset since they will simply serve to complicate the results, for example ‘SaleCondition’ (the different types of sales), ‘Street’, ‘Alley’, etc. On the numeric variables some normalization may be used. If some data is not normally distributed, especially if the mean and median vary significantly, a non-linear scaling may be applied (for example the feature ‘SalePrice’ will be logarithmically scaled). To the categorical variables some feature encoding will be applied, One Hot Encoder[29] may be applied, to create a group of n dummy features[30] (Yes/No (1/0) variables). Also, larger five and ten point quality scales and some discrete variables may be collapsed into fewer categories. Analyzing the correlation between

[27] http://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html

[28] http://scikit-learn.org/stable/modules/cross_validation.html

[29] http://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.OneHotEncoder.html

[30] http://pandas.pydata.org/pandas-docs/version/0.18.1/generated/pandas.get_dummies.html

the selected features and ‘SalePrice’, we will perform a Feature Selection of the most significant features.

4. Solution Statement

After all the data processing[31] and in order to make the predictions we will use some regression techniques, such as: LinearRegression, DecisionTreeRegressor, SVR, ElasticNet, Lasso, Ridge, LassoLars, BayesianRidge, GradientBoostingRegressor, ExtraTreesRegressor, BaggingRegressor, AdaBoostRegressor, XGBRegressor. To tune these models the grid search technique[32] [33] [34] will be used. After selecting the best models, according to their performances, an ensemble generation[35] will be implemented. Then, to determine if any improvements were made, the performance metrics of the ensemble will be calculated and compared with the ones of the benchmark model.

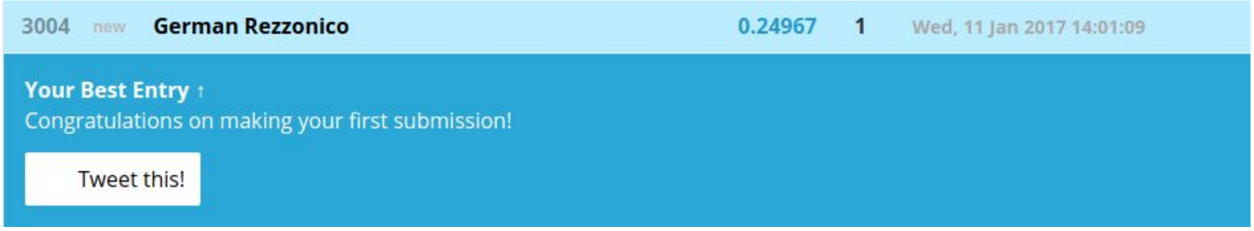
5. Benchmark Model

The House Prices: Advanced Regression Techniques competition uses RMSLE[36] as evaluation metric.
The public leaderboard for this competition presents the following stats for RMSLE values of the competing teams:

mean	0.328228
min	0.038390
25%	0.124012
50%	0.140710
75%	0.177783

As a first time in a competition of this nature, and due to the limited resources of my computer (Ubuntu 14.04.5 LTS[37] via crouton[38] on a Chromebook Acer C720-2802[39]), finishing in the top 20% of the 3306 participating teams will be quite an accomplishment.
That would mean that the developed model should obtain, at least, an RMSLE value of approximately 0.12050 to be in the position 650. A first submission shows this objective is obtainable:

[31] 3.1 Working with the data
[32] https://en.wikipedia.org/wiki/Hyperparameter_optimization
[33] http://scikit-learn.org/stable/modules/grid_search.html
[34] http://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html
[35] http://link.springer.com/chapter/10.1007%2F3-540-33019-4_19
[36] 6. Evaluation Metrics
[37] http://releases.ubuntu.com/14.04/
[38] https://github.com/dnschneid/crouton
[39] http://laptops.specout.com/l/2780/C720-2802



3004 new **German Rezzonico** 0.24967 1 Wed, 11 Jan 2017 14:01:09

Your Best Entry ↑
Congratulations on making your first submission!

6. Evaluation Metrics

To quantify the model's performance, some of the following evaluation metrics may be used, for example:

- Coefficient of determination[40] [41]: $R^2 = \left(\frac{1}{n} \times \sum_{i=1}^{n} \frac{(x_i - x) - (y_i - y)}{(\sigma_x - \sigma_y)}\right)^2$

Where n is the number of observations used to fit the model, Σ is the summation symbol, x_i is the x value for observation i, x is the mean x value, y_i is the y value for observation i, y is the mean y value, σ_x is the standard deviation of x, and σ_y is the standard deviation of y.

The values for R^2 range from 0 to 1, which captures the percentage of squared correlation between the predicted and actual values of the target variable. A model with an R^2 of 0 is no better than a model that always predicts the mean of the target variable, whereas a model with an R^2 of 1 perfectly predicts the target variable. R^2 values indicates what percentage of the target variable, using this model, can be explained by the features. A model can be given a negative R^2 as well, which indicates that the model is arbitrarily worse than one that always predicts the mean of the target variable.

- Mean Squared Error (MSE)[42] [43]: $MSE = \frac{1}{n}\sum_{i=1}^{n}(\widehat{y}_i - y_i)^2$

Where $\widehat{y}_i = predictions$ and $y_i = true\ values$.

MSE measures the average of the squares of the errors or deviations and is a measure of the quality of an estimator. It is always non-negative, and values closer to zero are better.

- Root Mean Square Deviation (RMSD) or Root Mean Square Error (RMSE)[44]:

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

RMSE is used as a measure of the differences between values (sample and population values) predicted by a model or an estimator and the values actually observed. The RMSE represents the sample standard deviation of the differences between predicted values and observed values. RMSE is a good measure of accuracy, but only to compare forecasting errors of different models for a particular variable.

[40] http://stattrek.com/statistics/dictionary.aspx?definition=coefficient_of_determination
[41] http://scikit-learn.org/stable/modules/generated/sklearn.metrics.r2_score.html
[42] https://en.wikipedia.org/wiki/Mean_squared_error
[43] http://scikit-learn.org/stable/modules/generated/sklearn.metrics.mean_squared_error.html
[44] https://en.wikipedia.org/wiki/Root-mean-square_deviation

Always between 0 and 1, since R is between -1 and 1. It tells us how much smaller the RSME will be than the SD.
For example, if all the points lie exactly on a line with positive slope, then R will be 1, and the RMSE will be 0.

- Root Mean Squared Logarithmic Error (RMSLE)[45]:

$$RMSLE = \varepsilon = \sqrt{\frac{1}{n} \times \sum_{i=1}^{n} \left(log\left(\widehat{y}_i + 1\right) - log\left(y_i + 1\right)\right)^2}$$

Where: n is the total number of observations in the (public/private) data set, $\widehat{y}_i$ is the prediction, and y_i is the actual response for i and log(x) is the natural logarithm of x.
RMSLE penalizes an under-predicted estimate greater than an over-predicted estimate. In the case of the Ames dataset, logs means that errors in predicting expensive houses and cheap houses will affect the result equally. Values closer to zero are better.

7. Project Design

This project may use the following libraries: NumPy[46], SciPy[47], matplotlib[48], Seaborn[49], pandas[50], sklearn[51], xgboost[52] and pickle[53]. The dataset will be divided into two parts, using train_test_split from sklearn.cross_validation[54], to shuffle and split the features and ‘SalePrice’ data into a training and a testing set. First EDA (Exploratory Data Analysis) will be perform to get a better understanding of the data. Entries with missing data will be remove or replace with some arbitrary value. Some variables may be dropped from the dataset since they will simply serve to complicate the results, for example ‘SaleCondition’ (the different types of sales), ‘Street’, ‘Alley’, etc. On the numeric variables some normalization may be used. If some data is not normally distributed, especially if the mean and median vary significantly, a non-linear scaling may be applied (for example the feature ‘SalePrice’ will be logarithmically scaled). To the categorical variables some feature encoding will be applied, One Hot Encoder may be applied, to create a group of n dummy features (Yes/No (1/0) variables). Also, larger five and ten point quality scales and some discrete variables may be collapsed into fewer categories. After analyzing the correlation between the features and ‘SalePrice’’, we will perform a Feature Selection of the most significant features. Then Model Selection will be performed, the regression techniques used may be: LinearRegression, DecisionTreeRegressor, SVR, ElasticNet, Lasso, etc[55]. To tune the models, the grid search technique will be used. The results from these regressions will be analyzed and the best models will be chosen. Once the best models are selected, according

[45] https://www.kaggle.com/wiki/RootMeanSquaredLogarithmicError
[46] http://www.numpy.org/
[47] https://www.scipy.org/
[48] http://matplotlib.org/
[49] http://seaborn.pydata.org/
[50] http://pandas.pydata.org/
[51] http://scikit-learn.org/stable/
[52] https://github.com/dmlc/xgboost
[53] https://docs.python.org/2/library/pickle.html
[54] 3. Datasets and Inputs
[55] 2. Problem Statement

to their performances, an ensemble generation will be implemented. The performance metrics calculated and compared with the ones of the benchmark model[56], to determine if the target was reached.

[56] 5. Benchmark Model