## Housing Prices Predictions by 19229028 Khashbat Enkhbat

Introduction

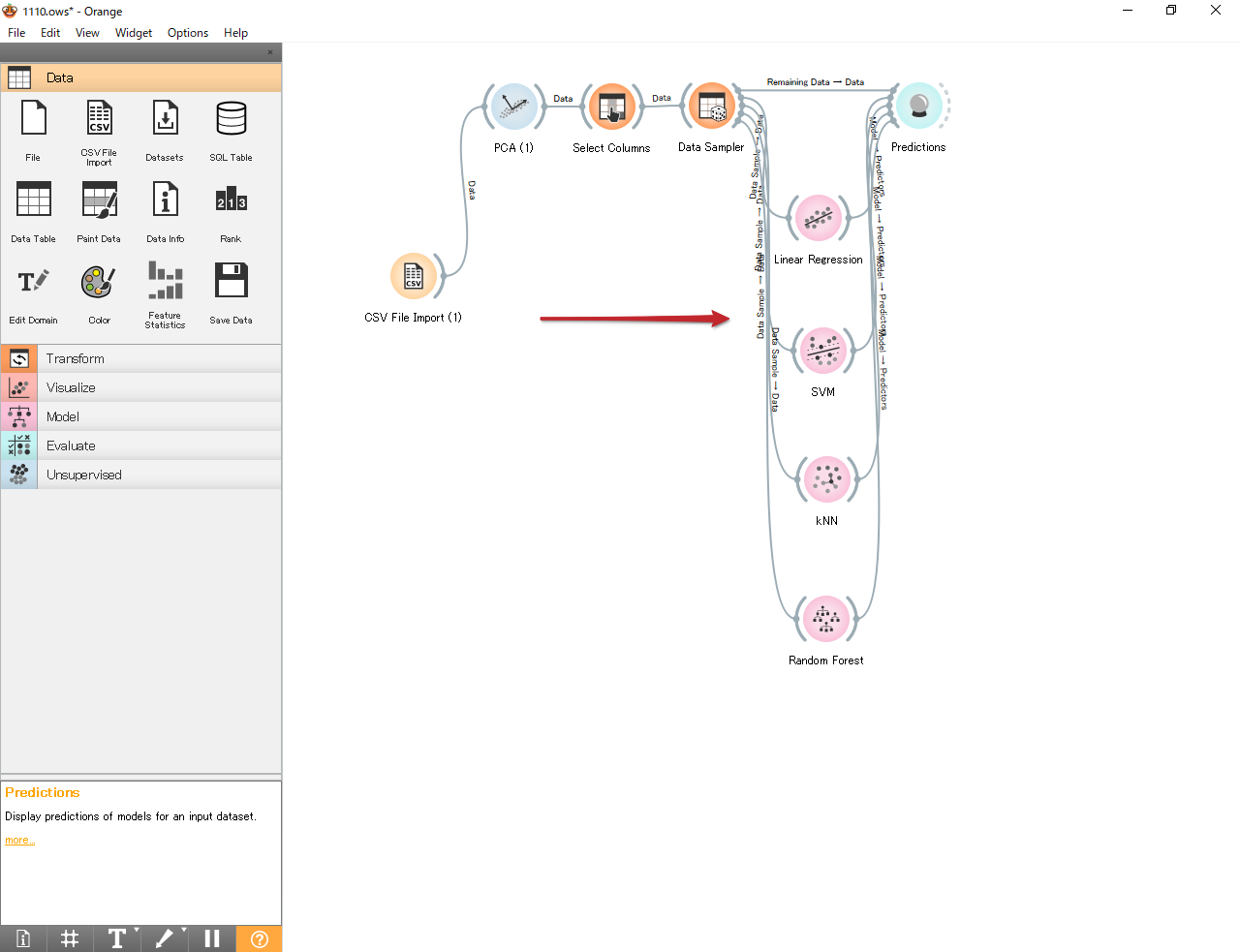
California Housing Prices dataset is made from 10 features. These features include longitude, latitude, total rooms, total bedrooms, population, households, median income, median age, median house value, and ocean proximity. All these features are closely related to the indicators that can approximate the median house value. For example, longitude and latitude can be key factors because higher value of median house value can be found farther west and farther north. To predict the value of median house value, the features must be logical and rational. In this case, all of the features have high correlation to each other and are logical to predict a feature among them. In addition, the data must be clean for complete analysis. Missing data or values are common occurrences in data analysis. If the data has missing values, it could significantly affect the conclusion of our analysis. Therefore, checking if the data is missing from the dataset, is a must. In the California Housing Prices dataset, however, there were not any missing values, so there is no need to use the “Impute” method in our orange workflow, in which case we would either use the mean of the data replace the null values.

Machine Learning Models

In my study-case, I used four machine learning models. These include Lasso Regression, Support Vector Machine (SVM), kNN, and Random Forest. These supervised learning methods are suited in each of their areas, but in California Housing Prices dataset, I need to find out which one can predict better. With the training dataset of 70% of all the records (nearly 14000) and the rest (nearly 7600) as the test data, we could find out which one can better predict median household value.

The Workflow of The Orange

The Workflow of the Orange can be found below, which demonstrate steps used for analysis. First, we use CSV File Import to import the file into Orange. We could also use API to use time series analysis if it was needed. However, in our case, there is no need. Second, we use PCA to reduce the dimensionality of the features from 10 to 5. Third, we use the select column to select which features to use to predict a feature and which one to choose as a target feature. In our case, we want to predict median house value, and select it as a target value. Fourth, we use Data Sampler to divide how much portion to take it as training and test data. We use default values of 70% and 30%. From this point, we select which models to use on our data. We can see that data goes to models and the remaining 30% of the test data directly goes into predictions.

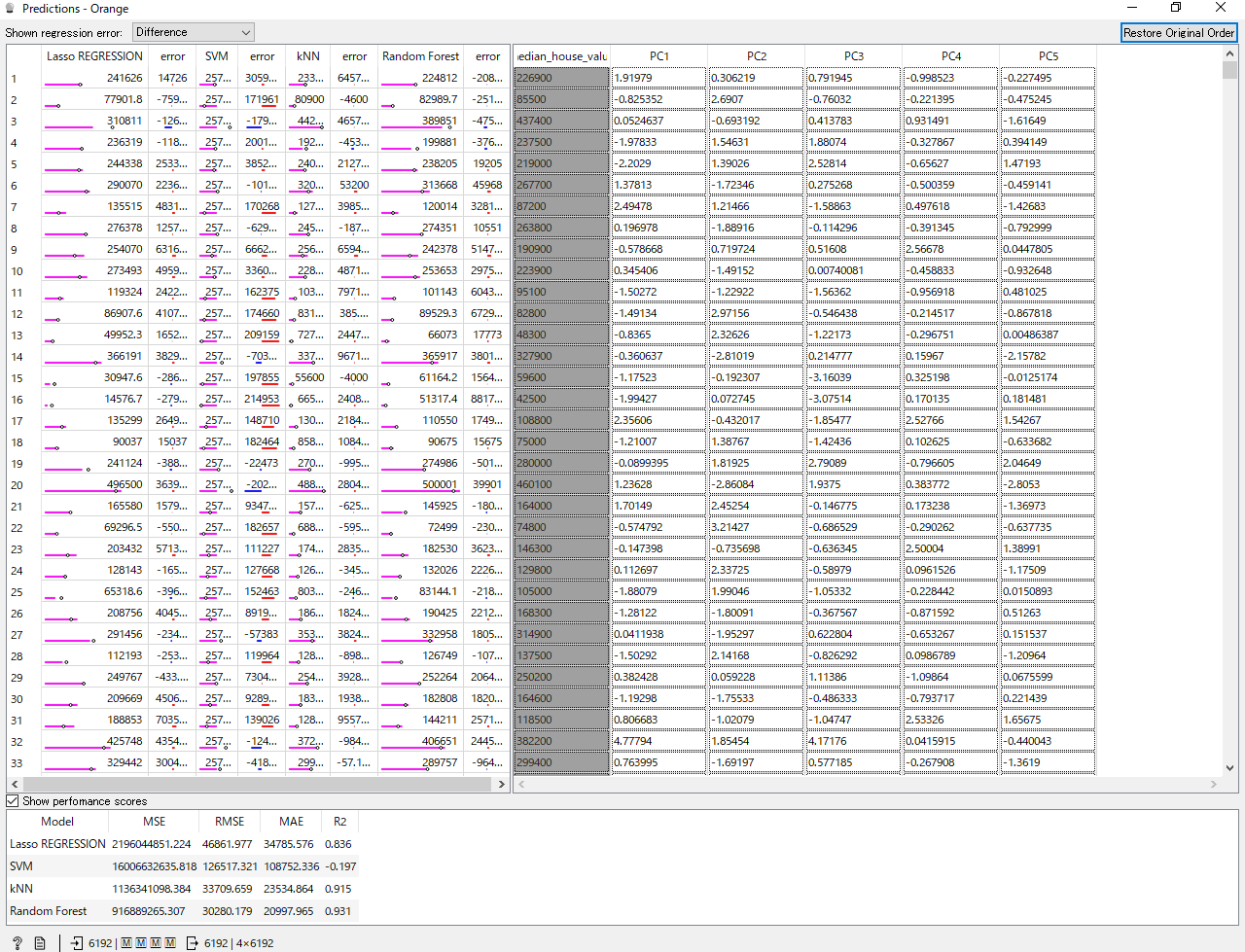


Results and Analysis

From the predictions, we can see our results. With PCA 5 components, the models generated their versions of predictions. In R2 section of the result, we can find out the accuracy of each model. The closer it gets to 1, the better it is. The results:

Lasso Regression – 0.836, SVM - (-0.197), Knn – 0.915 and Random Forest – 0.931.

In this dataset, the most suited model seems to be Random Forest model with approximately 90% chance of getting the prediction right.



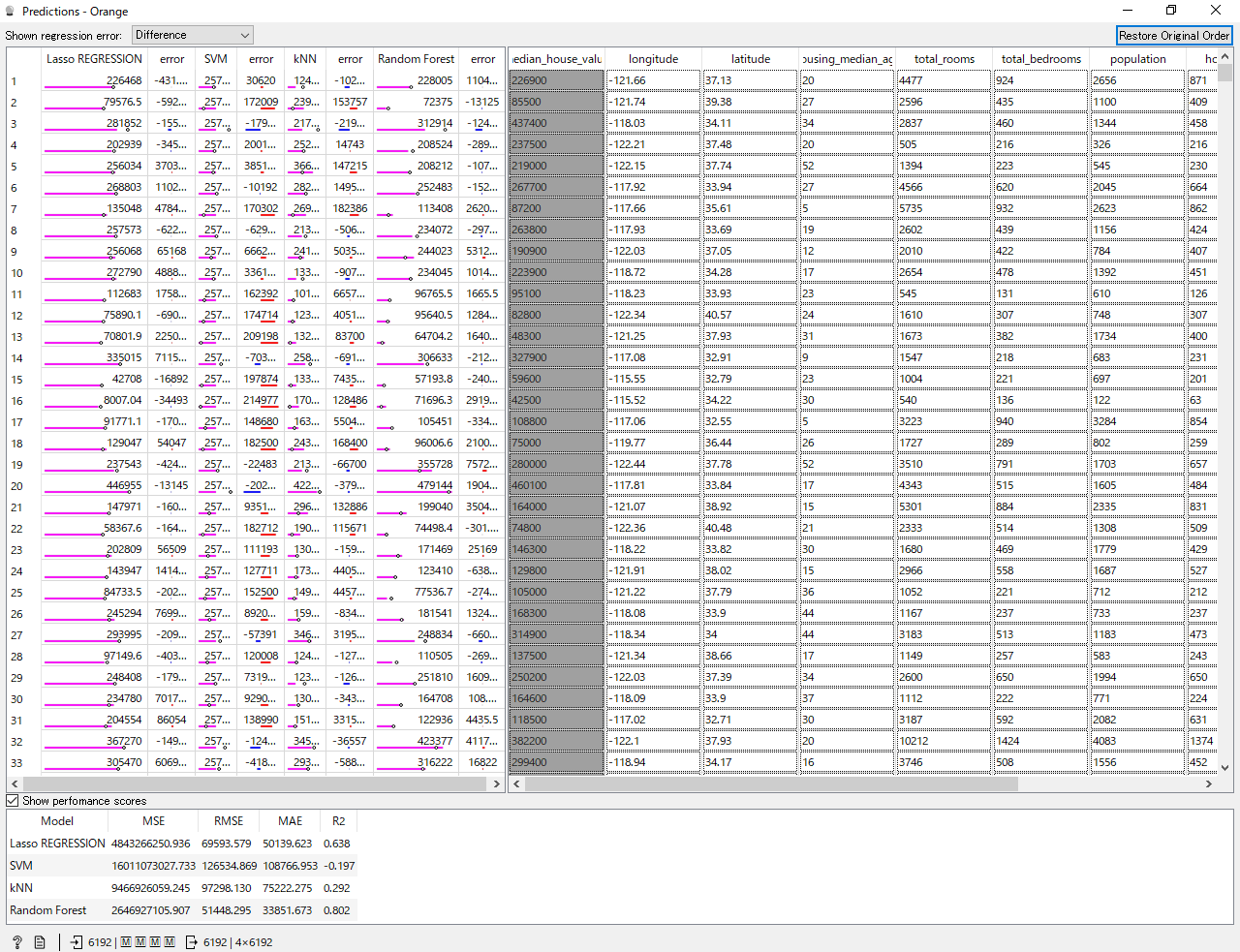
However, if we do not use PCA and try to see the results, we can see different results. In the case below, I did not use PCA to predict. PCA is mainly used to decrease the dimensionality of the data, but with PCA it seems to increase the accuracy of the models. In theory it should make no difference. Without PCA predictions are as follows:

Lasso Regression – 0.638, SVM - (-0.197), Knn – 0.292 and Random Forest – 0.802.

Almost every model has decreased in accuracy.

Why is it so?

It seems that dimensionality is a problem for neural networks, so reducing the dimension of the data using PCA can hasten process and enhance the quality of the output. The PCA doesn't change anything in theory, but in practice it increases training speed, makes the neural network needed to represent the data simpler, and produces systems that more accurately characterize the "intermediate structure" of the data rather than having to take into account different scales.



Conclusion

Orange is a powerful tool for machine learning applications. It houses several powerful machine learning algorithms and useful tools for data analyst. It is simple yet complex and powerful machine.

In our case of California Housing Prices, it seems that the most suitable model was Random Forest from the selected methods. As we attempted to make predictions, we discovered that PCA can not only reduce the dimensionality of the data but also make prediction accurate by making data simpler for models.