11/17/2017

Forecasting King County House prices Using SAS E-Miner
Final Project Report

Instructor: Prof. Zhe (James) Zhang

Group 12:
Riti Kumari
Aishwarya Nandapurkar
Paryag Mehta
Madhukar Jadhav
Radha Chavali

Table of Contents

Introduction:

The three basic items that human being concern themselves are: Food, Clothing and Shelter. Considering the everyday struggles of finding a right shelter for a family at the right price is been a struggle. So, to reduce the painstaking process of determining the right house at the right place and at the right time, this project enlightens the consumer to plan ahead of time for the purchase of the house. In this project, we aim to perform data analysis on King County, USA data set. We plan to perform predictive analysis using SAS Enterprise Miner. King County is a county located in Washington. It is the 13th most populous county in the United States. The county seat is Seattle which is the state's largest city. As of the 2010 US Census report, there were 851,261 housing units at an average density of 402.4 per square mile (155.4/Km^2). The goal of the project is to enable us to study and evaluate the variation of house prices in King County based on different house attributes.

Problem Statement:

Our main aim is to build a best model to predict the prices of a house in king county, for which the data set has been collected. The following information can be derived once we analyze the data

- Pricing of a specific house
- Factors affecting the prices of a house in King County

This analysis can be helpful for an individual consumer as well as real estate agents to incur profits.

- Individual consumer can determine the price of a house depending on the floor size, number of rooms and other factors.
- Similarly, an Individual consumer can predict the appropriate price for selling his house.
- Real estate agents can determine the right price and secure good profits in their business
- Additionally, King County government can keep track of the demographic population using the data and housing prices.

Data Set Description:

Our project uses second-hand data retrieved from Kaggle. This dataset encapsulates a varied set of variables which play a major role in defining the actual price of the house. This dataset contains house sale prices for homes sold between May 2014 and May 2015 in King County, USA. In all, it contains 19 house features plus the "price" and the "id columns", along with 21613 observations.

High Level Data summary:

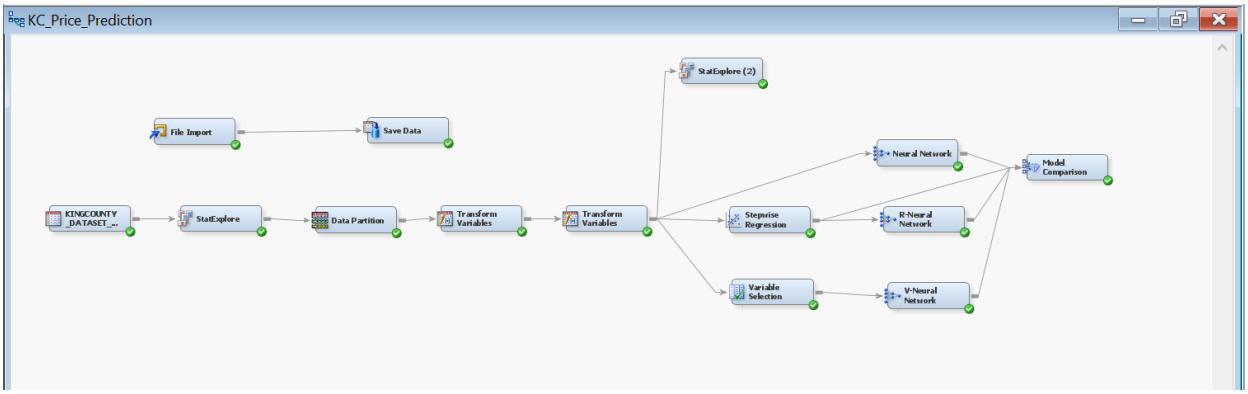
Total observations in the dataset	21613
Total number of binary variables	2
Total number of nominal variables	5
Total number of interval variables	14
Outcome / target variable	"price"
Level of the target variable (nominal, binary or interval)	Interval

Below is the description about each variable 'column' used in the dataset.

id - a unique notation for a house
Year – Year the house was sold
price - price is prediction target (price of the house)
bedrooms - number of Bedrooms/House
bathrooms - number of bathrooms/bedrooms
sqft_living - square footage of the home
sqft_lot - square footage of the lot
floors - total floors (levels) in house
waterfront - house which has a view to a waterfront view
condition - how good the condition is (Overall rating [1 - 5] - 5 being the best condition)
grade - overall grade given to the housing unit, based on King County grading system (Rating [1 - 13] - 13 being the best)
sqft_above - square footage of house apart from basement
sqft_basement - square footage of the basement
yr_built – Year in which the house was built
yr_renovated - Year when house was renovated
zipcode – zip code of the area
lat - Latitude coordinate
long - Longitude coordinate
sqft_living15 - Living room area in 2015(implies-- some renovations) This might or might not have affected the lot size area
sqft_lot15 - lot Size area in 2015(implies-- some renovations)

Data Mining Technique:
In our dataset, initially data cleaning will be performed to replace/remove missing values. We tried to classify the target variable using Regression analysis, to predict the output variable. We would also do analysis using Neural network and compare models to pick the best fit.

Project Diagram:

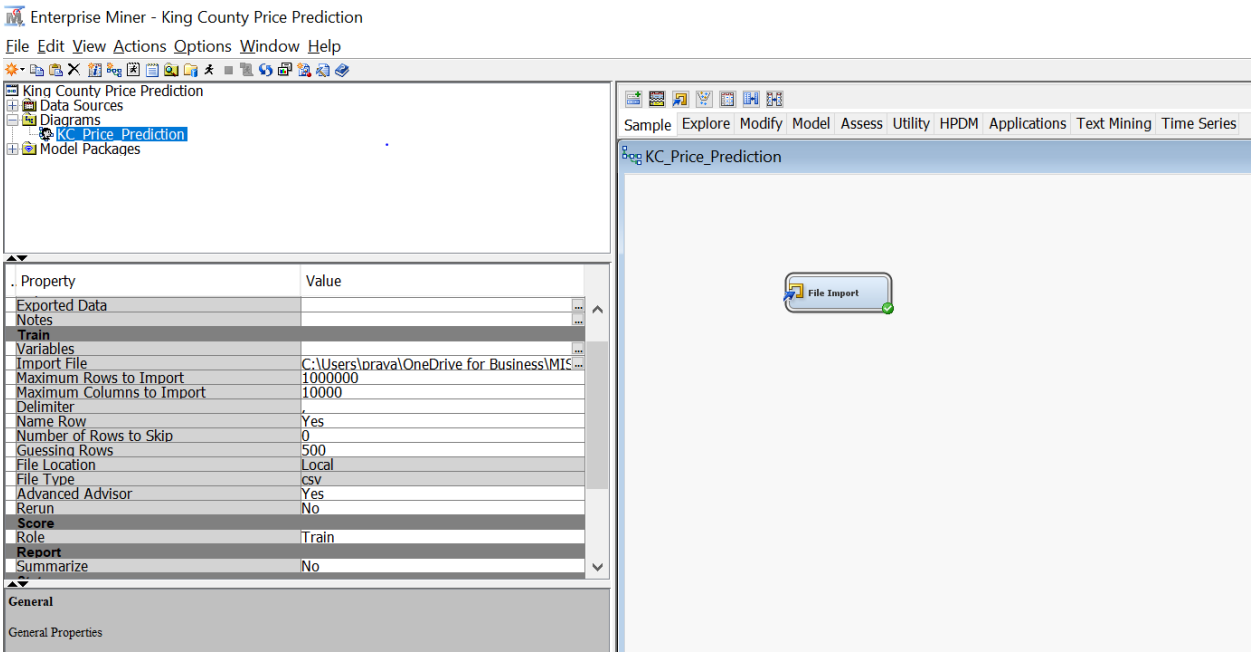


Data Pre-Processing:

Feature Selection:

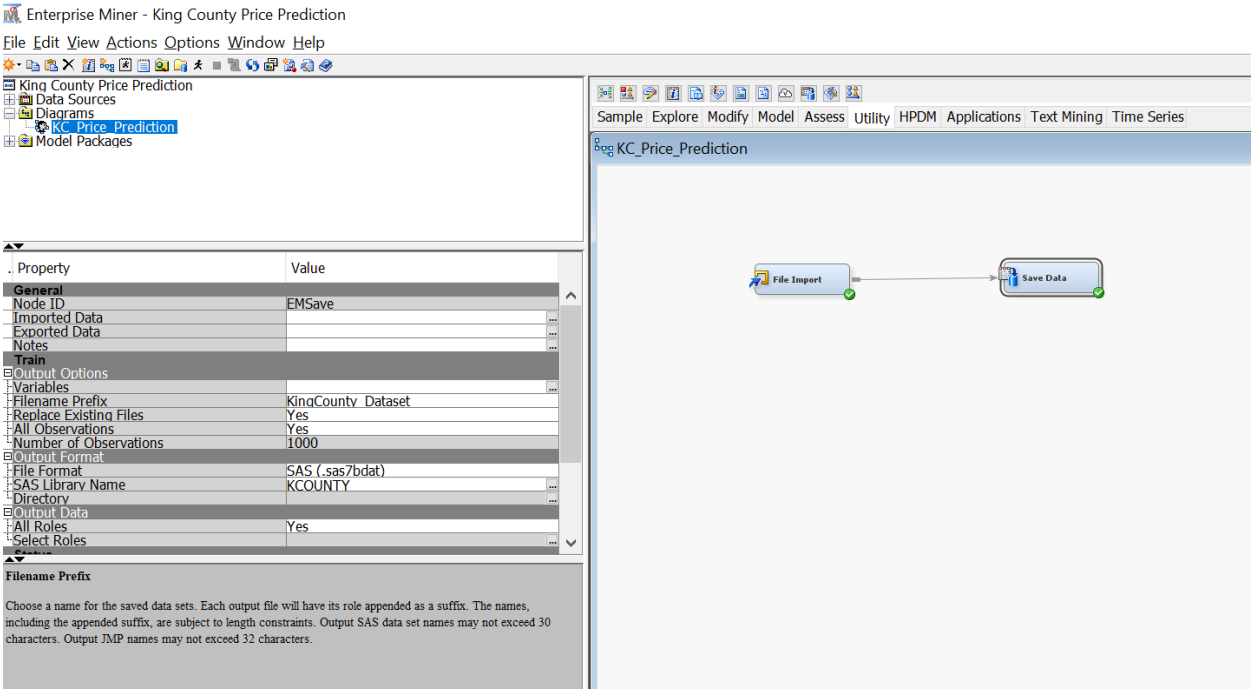
- We changed the format of the variable 'DATE' to 'YYYY'.
- Additionally, the number of bathroom variable represents numbers in decimals, which is unrealistic. Hence the values of bathroom are rounded off to a closest integer using round function. For the above modifications, we will use transform variable node.
- The variable "yr_renovated" has a minimum value of 0, which is incorrect, we replaced the zero with the year in which the house was built.
  The zero indicates that the house is not renovated from the time it was built.

1. File Import:
Our dataset consists of one csv file. The csv file was imported using the File import node. The local path was provided to import file field to successfully import the dataset. Also, changed the property 'Advanced Advisor' to 'Yes'.



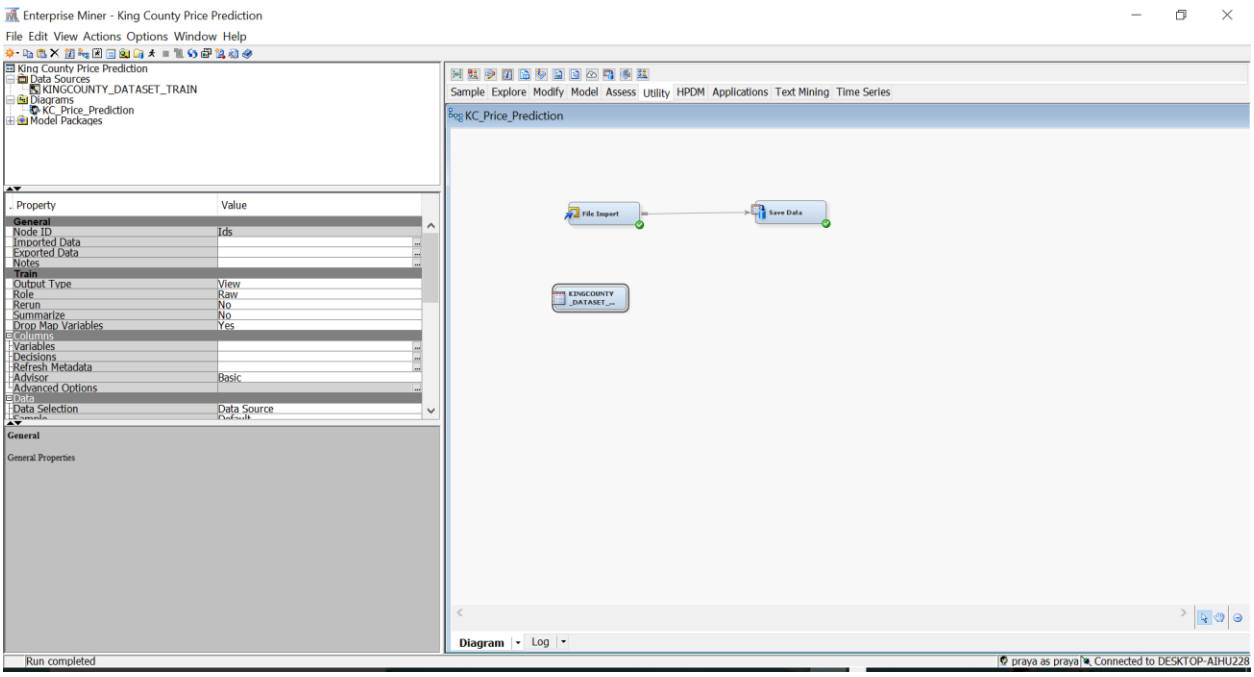
2. Save Data:

The node was used to convert the dataset to sas7bdat file. The file is saved as KingCounty_Dataset by passing the name to the field "Filename Prefix" and the library in which the file should be stored was provided to the field "SAS Library name".

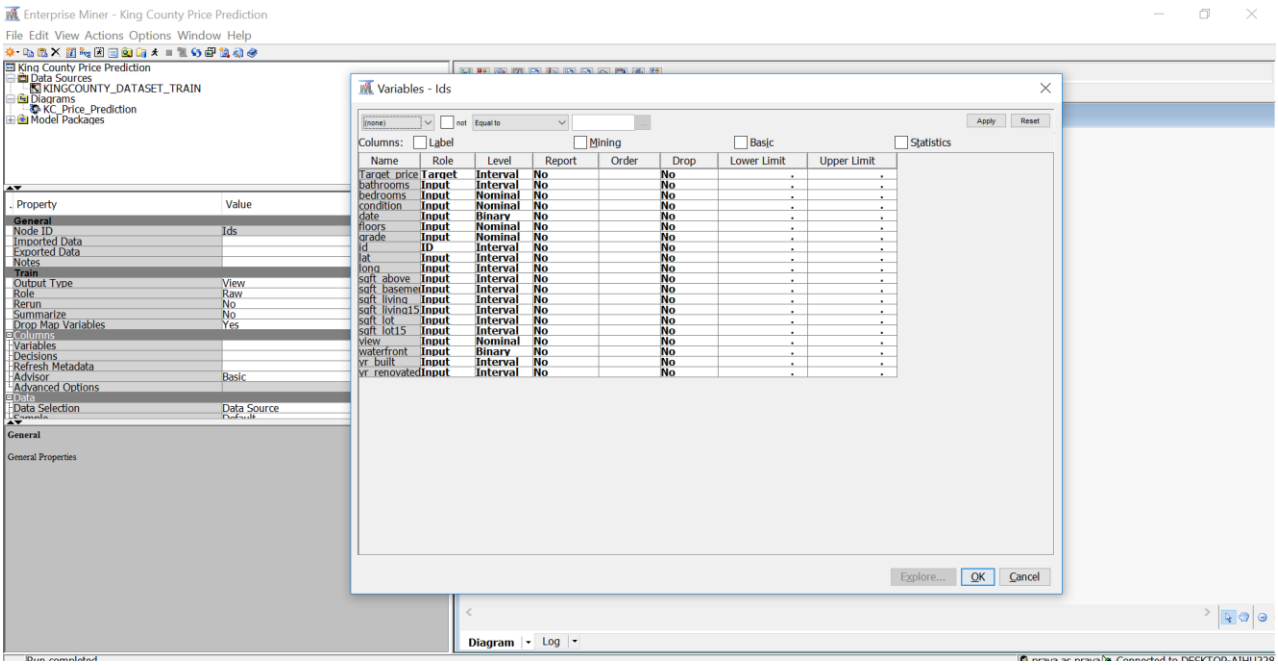


3. Data Source:

The above saved sas7bdat file was then created as a new data source.

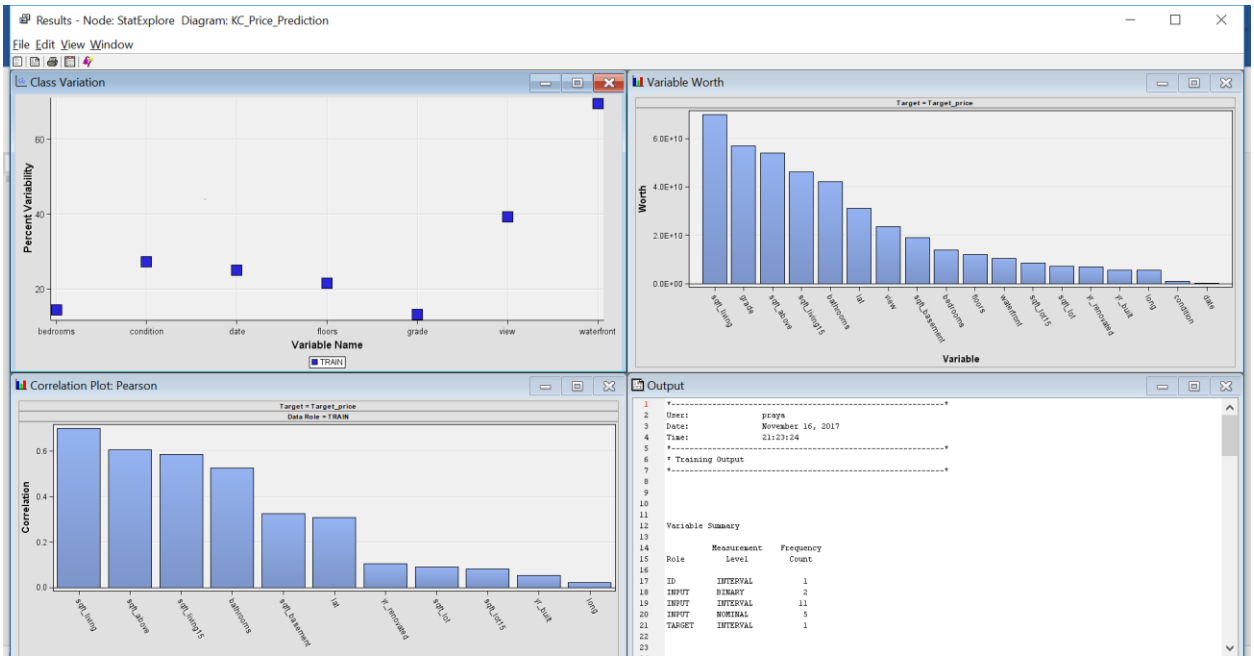


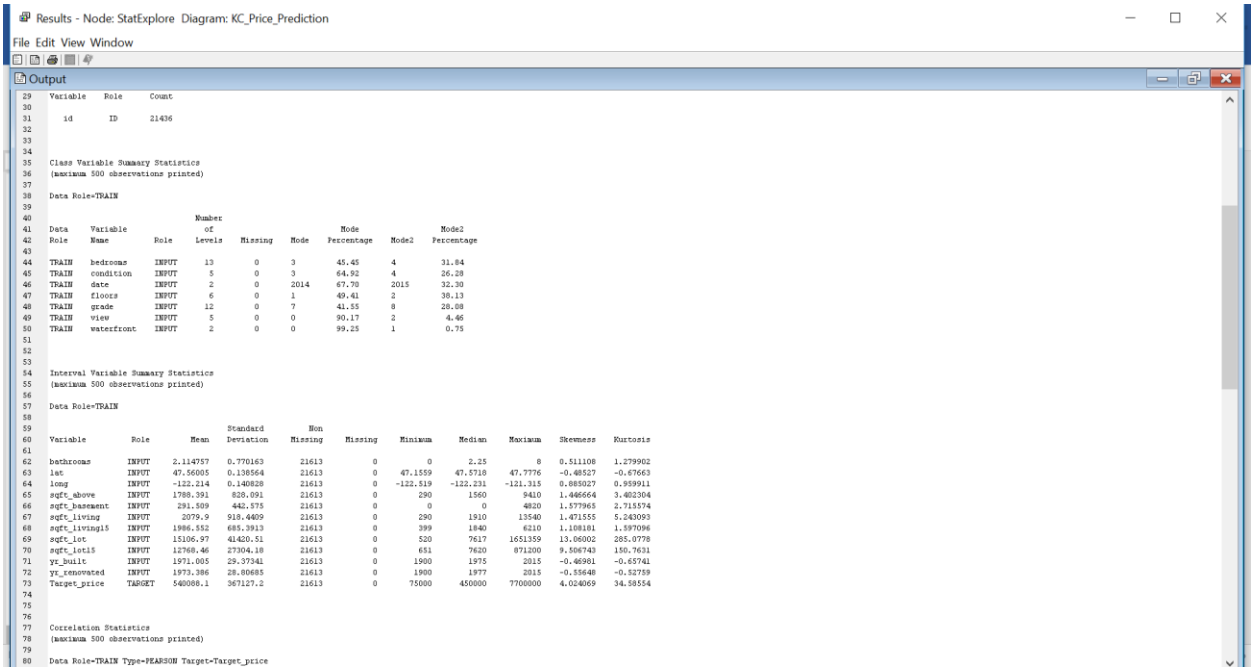
Below is the list showing all the 20 variables: the "Zipcode" variable is rejected as it does not hold any significance in predicting the target "Price" .



4. Stat Explore:

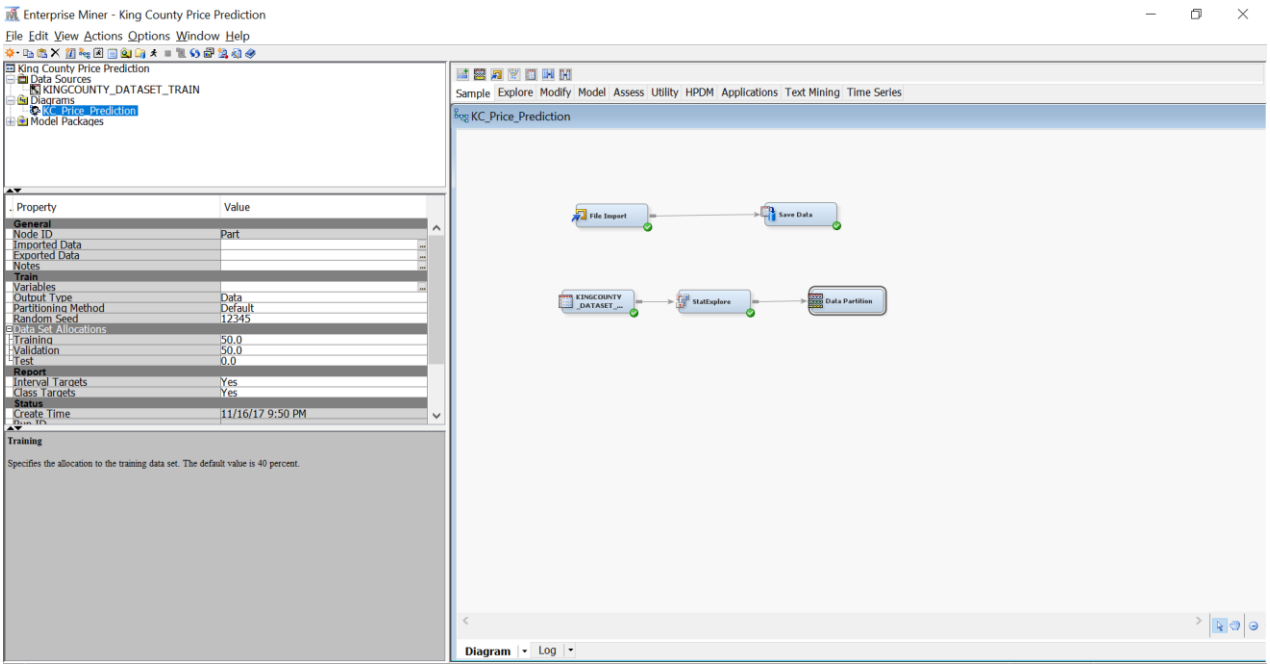
This node is used to generate a summary about the input variables. From the results, we found that there are **no missing values** in the dataset. As a result, we don't need to use impute node in our model to take care of missing values. Below is the summary:

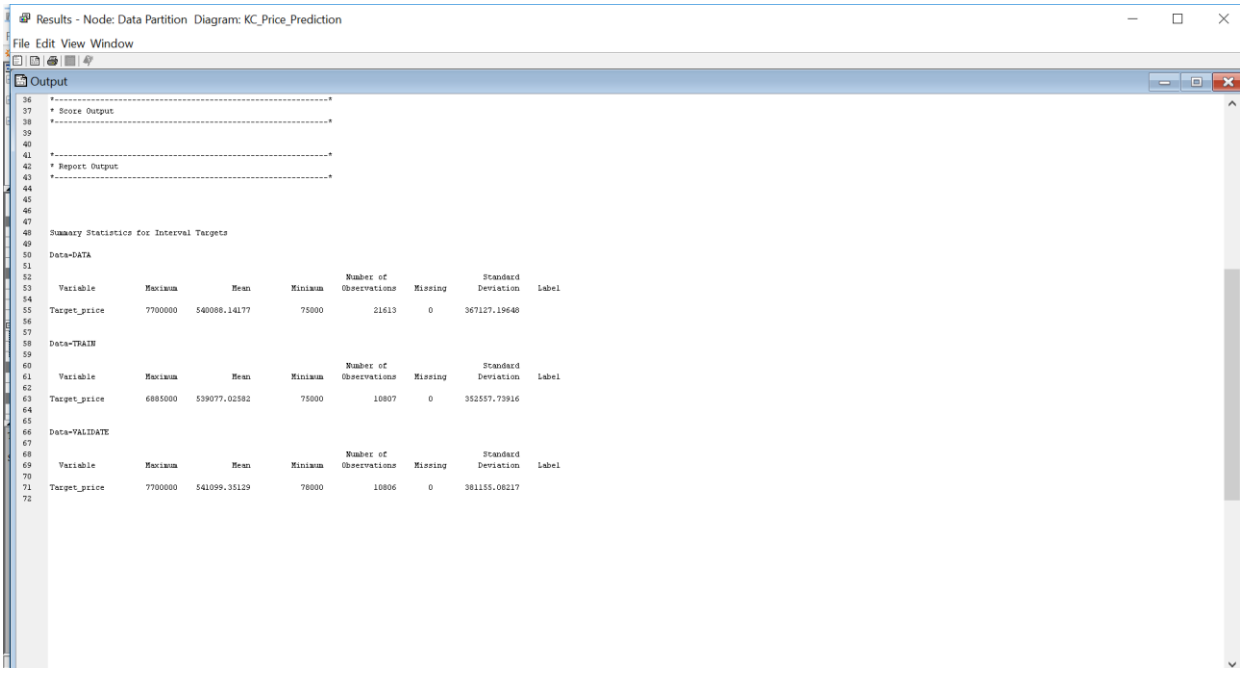




5. Data Partition:

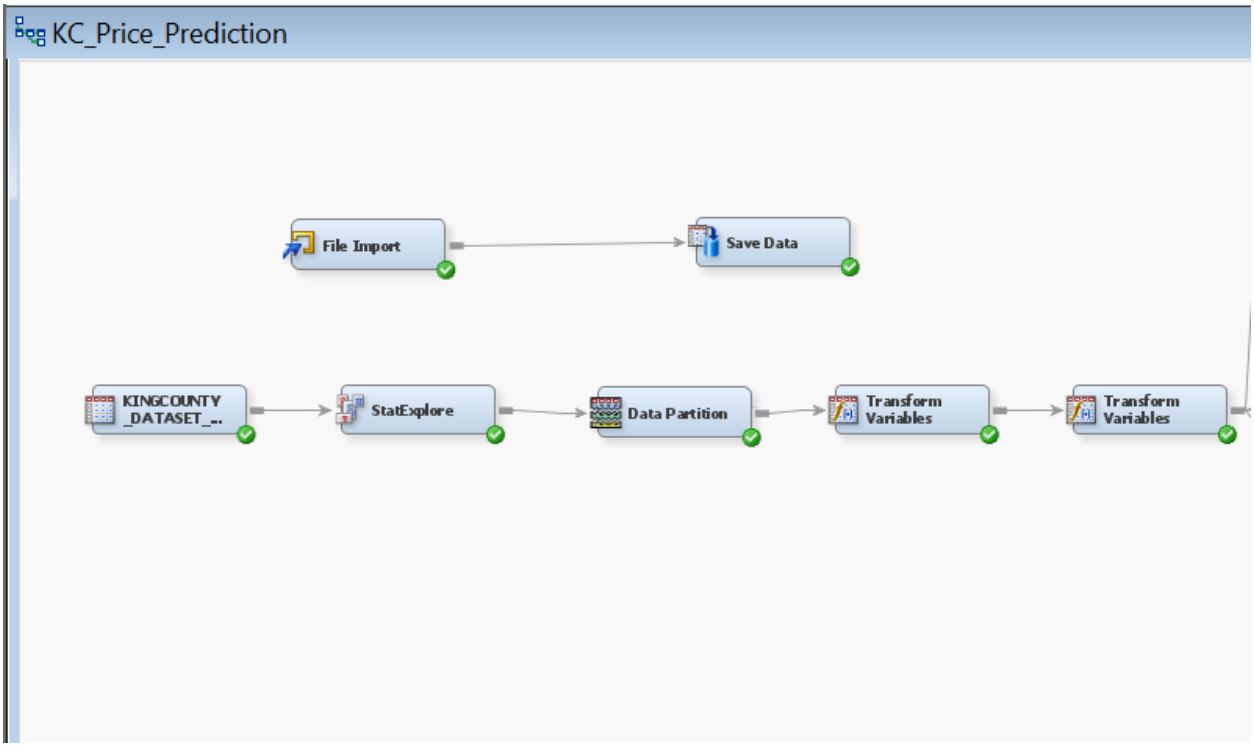
This node is used to partition the dataset into two parts namely,
Training set: 50%
Validation set: 50%
Since we are not going to use Test set for Model assessment, we did not create Test Data set.

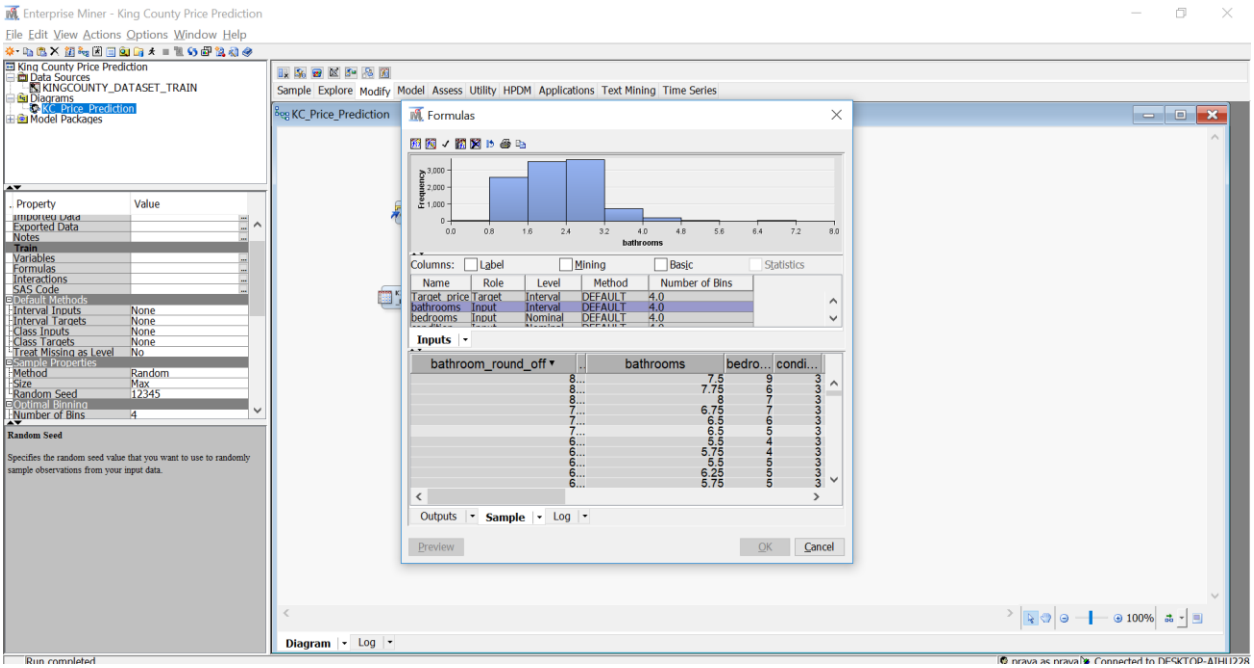




6. Transform Variables:

This node is used to create a new column "bathroom_round_off" in which the values for the variable 'bathrooms' are rounded off. As we have decimal values in Bathroom such as 1.25,1.5 and 1.75, these values are replaced with the closest integer value. We used the formula - 'ROUND(bathroom)'- to round off the decimal values of variable 'bathrooms'



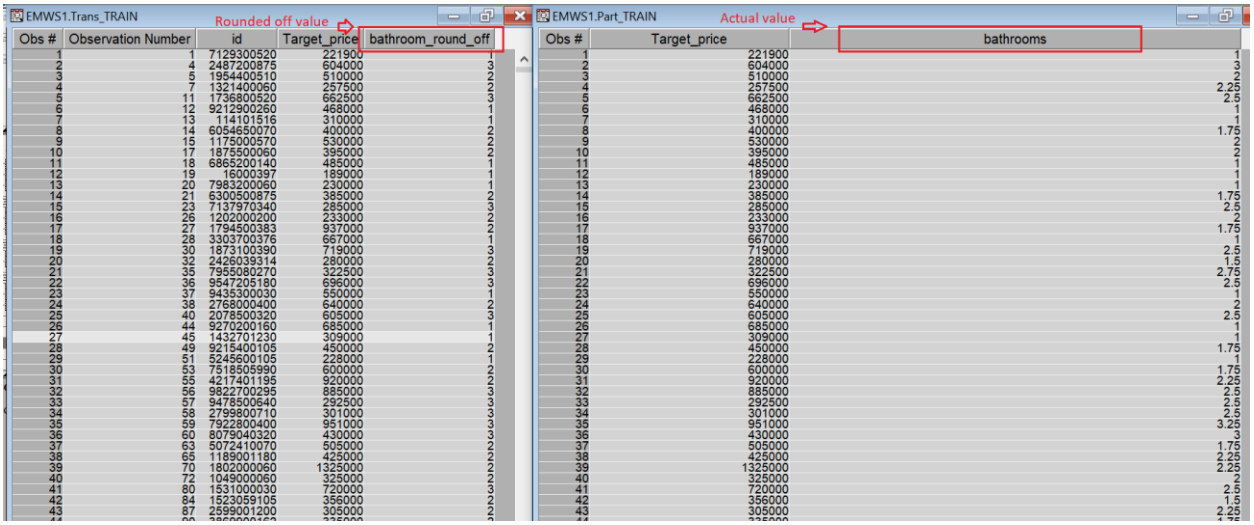


The second transform node is used to standardize all the interval variables.
Standardization is performed to make sure that all the variable fall under same scale.

Name	Method	Number of Bins	Role	Level
Target_price	Standardize	4	Target	Interval
bathroom_ro	Standardize	4	Input	Interval
bedrooms	Default	4	Input	Nominal
condition	Default	4	Input	Nominal
date	Default	4	Input	Binary
floors	Default	4	Input	Nominal
grade	Default	4	Input	Nominal
lat	Standardize	4	Input	Interval
long	Standardize	4	Input	Interval
sqft_above	Standardize	4	Input	Interval
sqft_baseme	Standardize	4	Input	Interval
sqft_living	Standardize	4	Input	Interval
sqft_living15	Standardize	4	Input	Interval
sqft_lot	Standardize	4	Input	Interval
sqft_lot15	Standardize	4	Input	Interval
view	Default	4	Input	Nominal
waterfront	Default	4	Input	Binary
yr_built	Standardize	4	Input	Interval
yr_renovated	Standardize	4	Input	Interval

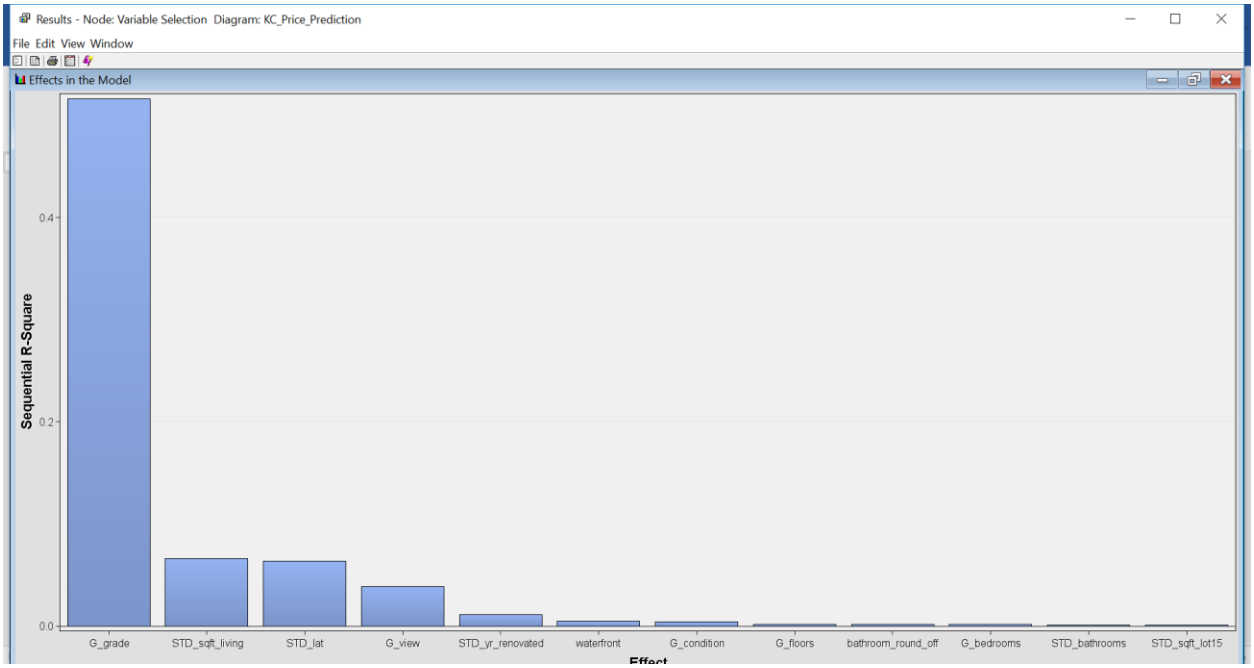
7. Stat Explore Transformed:

This node is generated to generate the summary of the transformed variable.

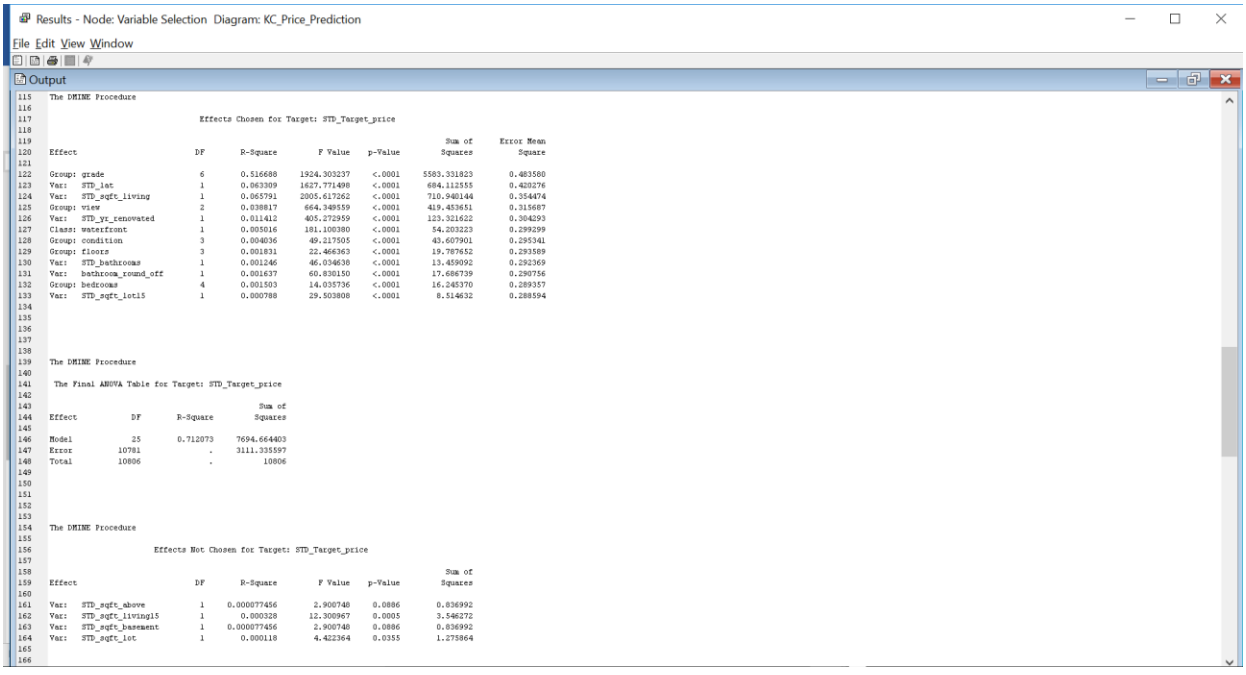


8. Variable Selection:

This node is used to filter the variables based on their R-square values.



The below screenshot describes which variables are chosen (selected for analysis) and rejected as per the R-square analysis.



As our target variable(price) is an interval type variable we are going to perform regression analysis as well as neutral network analysis. In case of regression analysis, we checked with forward, backward as well as stepwise regression and found that stepwise was the most effective in our case. We also performed a neural network analysis after this. But neural network has a problem that it will consider all the input variables and won't reject any of the insignificant variables. To counter this issue, we applied a variable selection node to our transformed data. This node selects all the necessary variables and determines the insignificant ones and reject them. Once done, we will supply the output of this node to the input of our neural network node just to analyze if the results are any better. We are considering one more model, we will be performing a regression analysis on the transformed data and this will be provided as an input to the Neural network node. We are performing such analysis with this path since we are aware that regression analysis will perform its operations on the variables and determine the most significant variables affecting the target variable. Hence, such a result when provided to Neural network should provide a good predictive model.
To summarize, we have 4 predictive models:

a. Neural network

b. Regression analysis

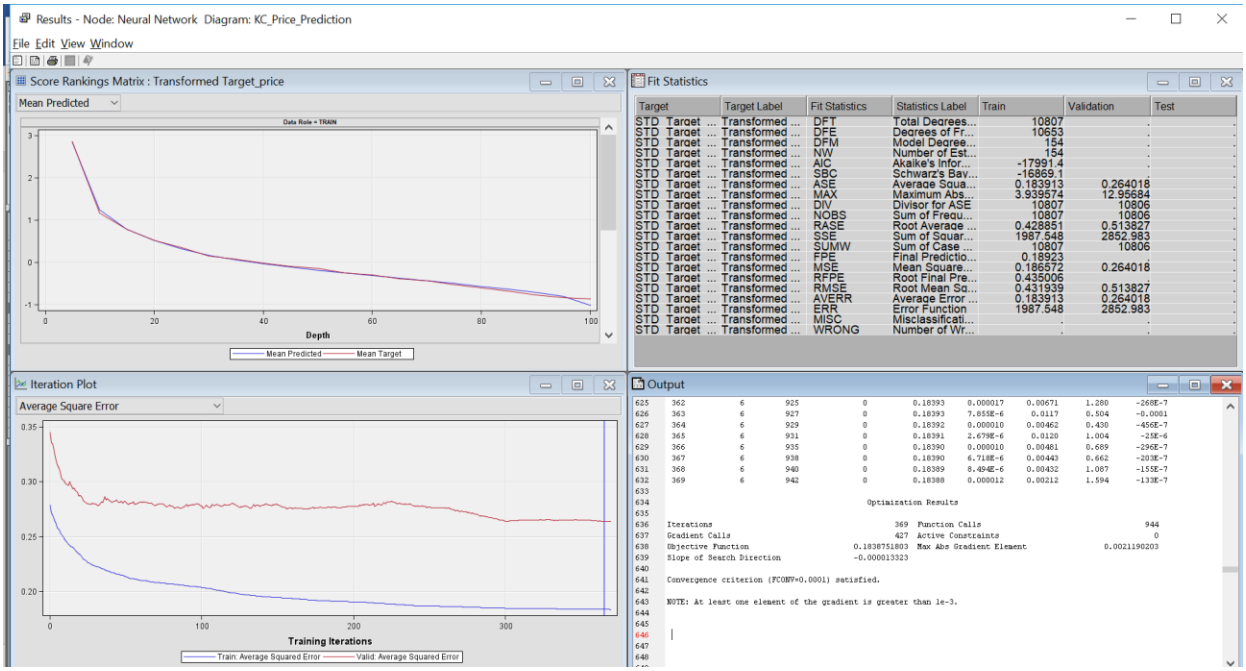
c. Neural network with regression node output as the input

d. Neural network with variable selection node as the input

Now we perform a model comparison to check which of these models perform better. This can be determined using the value of average squared error. The model with the lowest average squared error will be the best fit and will be chosen to predict the target price. Hence, we selected average squared error to be our criterion to determine the best model in the model comparison node.

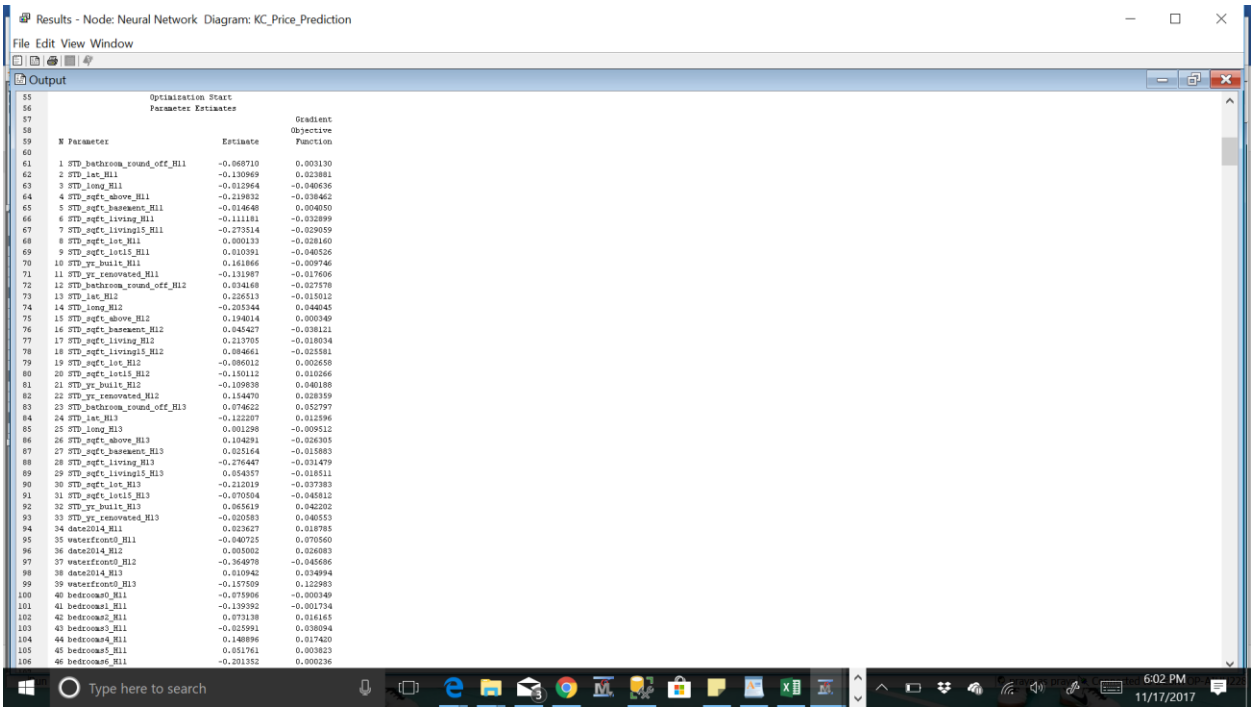
9. Neural Network Node:

Applying the neural network node to the output of transformed variables node. Model selection criterion is changed to "Average Error".



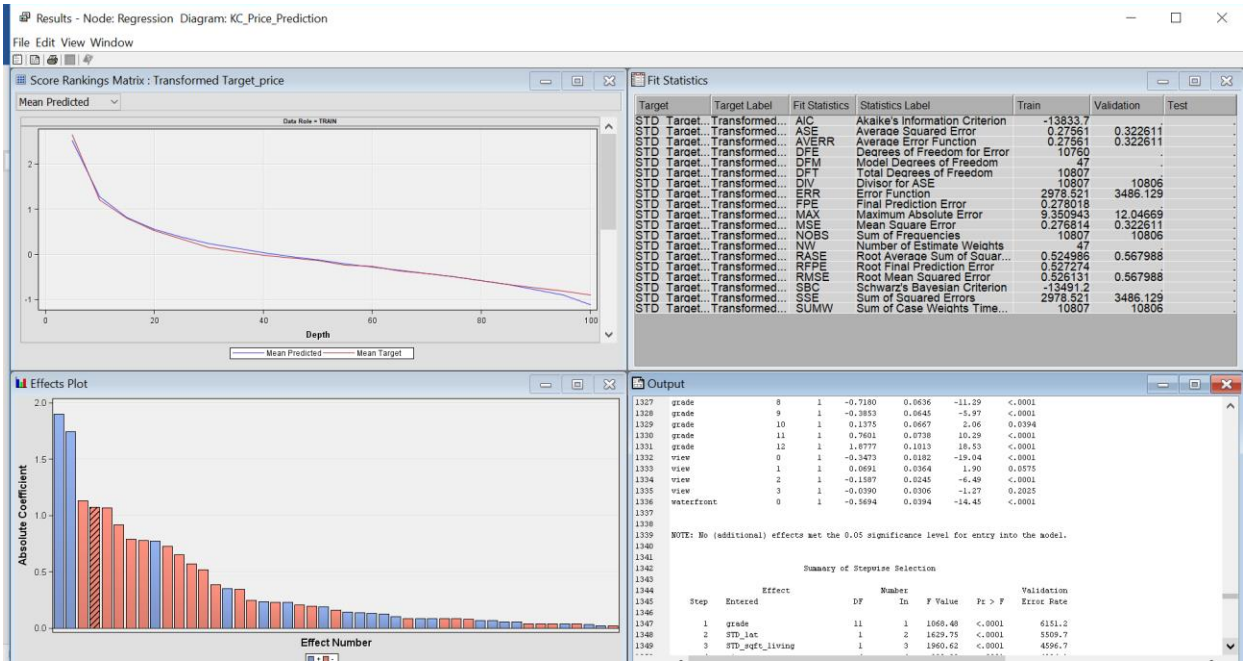
The output of Neural Network Node suggests that Average Square error for the validation data set (i.e. 0.2640) is more than the Train data set (i.e. 0.1839). Also, the number of weight estimates(NW) is 154, which is quite high; which may also impact the model performance.

Since Neural Network Node does not exclude any input variable, all the variables are significant. The same can be seen from the below screenshot of the output. Some of the variable show a positive estimate while some reflect negative estimates. The positive estimates reflect direct proportionality with the target price. Hence as the variable estimate increased the price increases and vice versa.

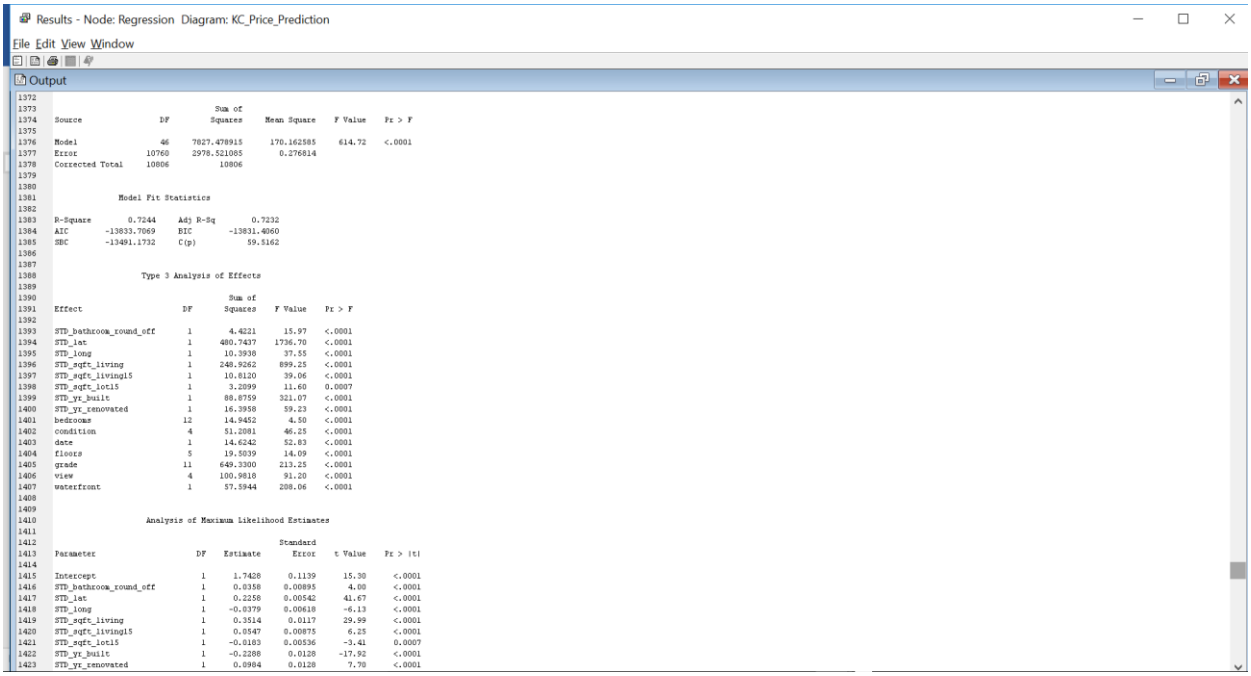

Results - Node: Neural Network Diagram: KC_Price_Prediction
File Edit View Window
Output
Optimization Start
Parameter Estimates
N Parameter Estimate Gradient Objective Function
1 STD_bathroom_round_off_H11 -0.068710 0.003130
2 STD_lat_H11 -0.130969 0.023881
3 STD_long_H11 -0.012964 -0.040636
4 STD_sqft_above_H11 -0.219832 -0.038462
5 STD_sqft_basement_H11 -0.014648 0.004050
6 STD_sqft_living_H11 -0.111181 -0.032899
7 STD_sqft_living15_H11 -0.273514 -0.029059
8 STD_sqft_lot_H11 0.000133 -0.028160
9 STD_sqft_lot15_H11 0.010391 -0.040526
10 STD_yr_built_H11 0.161866 -0.009746
11 STD_yr_renovated_H11 -0.131987 -0.017606
12 STD_bathroom_round_off_H12 0.034168 -0.027578
13 STD_lat_H12 0.226513 -0.015012
14 STD_long_H12 -0.205344 0.044045
15 STD_sqft_above_H12 0.194014 0.000349
16 STD_sqft_basement_H12 0.045427 -0.038121
17 STD_sqft_living_H12 0.213705 -0.018034
18 STD_sqft_living15_H12 0.084661 -0.025581
19 STD_sqft_lot_H12 -0.086012 0.002658
20 STD_sqft_lot15_H12 -0.150112 0.010266
21 STD_yr_built_H12 -0.109838 0.040188
22 STD_yr_renovated_H12 0.154470 0.028359
23 STD_bathroom_round_off_H13 0.074622 0.052797
24 STD_lat_H13 -0.122207 0.012596
25 STD_long_H13 0.001298 -0.009512
26 STD_sqft_above_H13 0.104291 -0.026305
27 STD_sqft_basement_H13 0.025164 -0.015883
28 STD_sqft_living_H13 -0.276447 -0.031479
29 STD_sqft_living15_H13 0.054357 -0.018511
30 STD_sqft_lot_H13 -0.212019 -0.037383
31 STD_sqft_lot15_H13 -0.070504 -0.045812
32 STD_yr_built_H13 0.065619 0.042202
33 STD_yr_renovated_H13 -0.020583 0.040553
34 date2014_H11 0.023627 0.018785
35 waterfront0_H11 -0.040725 0.070560
36 date2014_H12 0.005002 0.026083
37 waterfront0_H12 -0.364978 -0.045686
38 date2014_H13 0.010942 0.034994
39 waterfront0_H13 -0.157509 0.122983
40 bedrooms0_H11 -0.075906 -0.000349
41 bedrooms1_H11 -0.139392 -0.001734
42 bedrooms2_H11 0.073138 0.016165
43 bedrooms3_H11 -0.025991 0.038094
44 bedrooms4_H11 0.148896 0.017420
45 bedrooms5_H11 0.051761 0.003823
46 bedrooms6_H11 -0.201352 0.000236
Type here to search
6:02 PM
11/17/2017

10. Regression:

Applying the regression node to the output of Transform variables. In property panel, made changes for the regression type to be 'Linear regression' and selection model to be 'Stepwise'. Changed the selection criterion to 'Validation error'



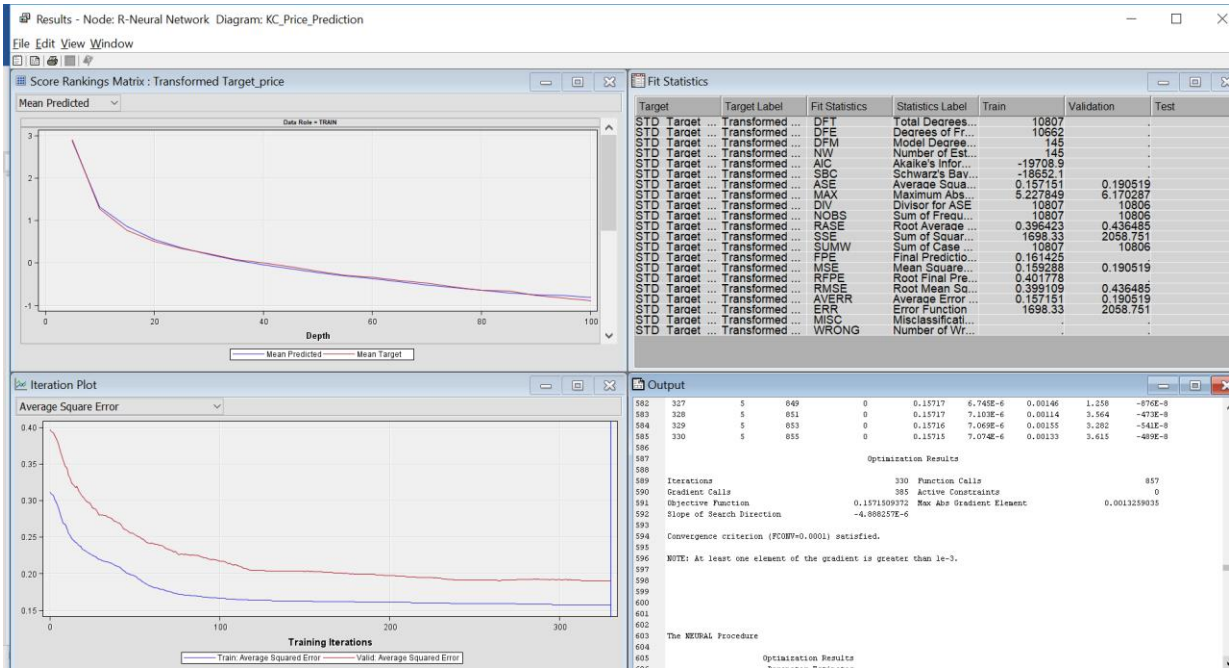
The output of Regression Node suggests that Average Square error for the validation data set (i.e. 0.3226) is more than the Train data set (i.e. 0.2756). Also, the number of weight estimates(NW) is 47, which is very less compared to Neural Network.



The above diagram shows the list of variables that have significant impact on the Target variable – Target Price. The Impact of the 2 variables **sqft_lot** and **sqft_above** turned out to be insignificant.

11. **Neural network with Regression analysis as the input:**

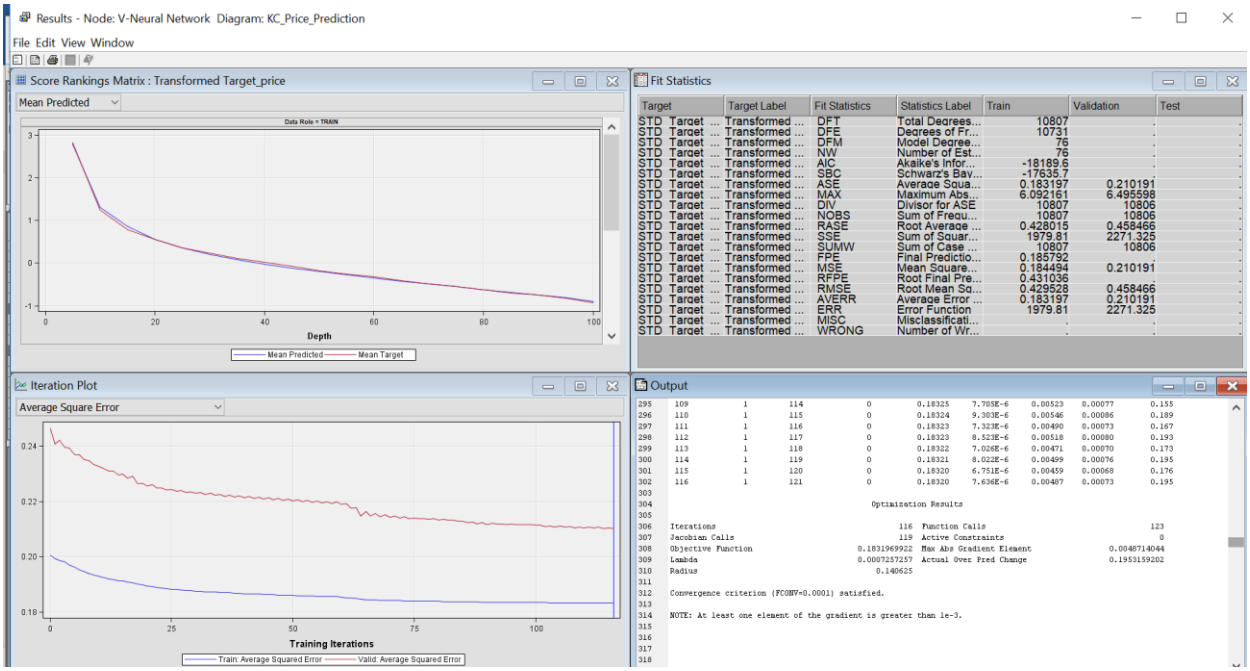
To overcome the Neural Network problem of selection of useful inputs, we are applying the output of regression node as an input to the neural network node.
Changed the selection criterion to 'Validation error'.



The output of 'Neural Network node with Regression Node output as an input' suggests that Average Square error for the validation data set (i.e. 0.1905) is more than the Train data set (i.e. 0.1571). Also, the number of weight estimates(NW) is 145, which is less compared to Neural Network. Also, the convergence criterion satisfied in 330 iterations compared to 365 iterations in Neural Network Model.

Since Neural Network does not omit any input variable, all significant variables of Regression Node are significant in this model.

12. Neural network with variable selection as input: Pulling in variable selection node and applying the output of transformed variables to it. Then applying the output of this variable selection to the Neural network node. Changed the selection criterion to 'Validation error'.



The output of 'Neural Network node with Variable Selection node output as an input' suggests that Average Square error for the validation data set (i.e. 0.2102) is more than the Train data set (i.e. 0.1831). Since the variable selection node had rejected many input variables, the number of weight estimates(NW) is 76, which is less compared to Neural Network or Neural Network with regression input. Also, the convergence criterion satisfied in 116 iterations compared to 365 iterations in Neural Network Model.

Model Comparison:

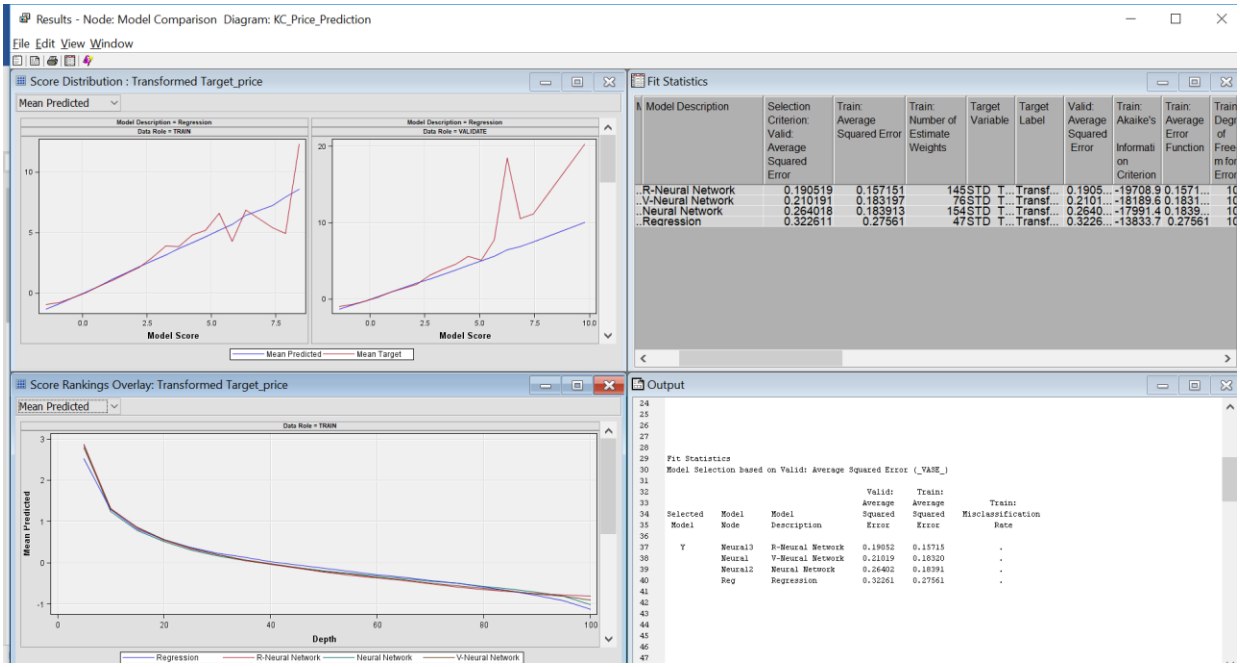
This node is used to compare all the models. The main factor distinguishing these models is their Average Squared Error. The following are the Average squared error for all models with Validation data:

Neural Network: **0.26402**

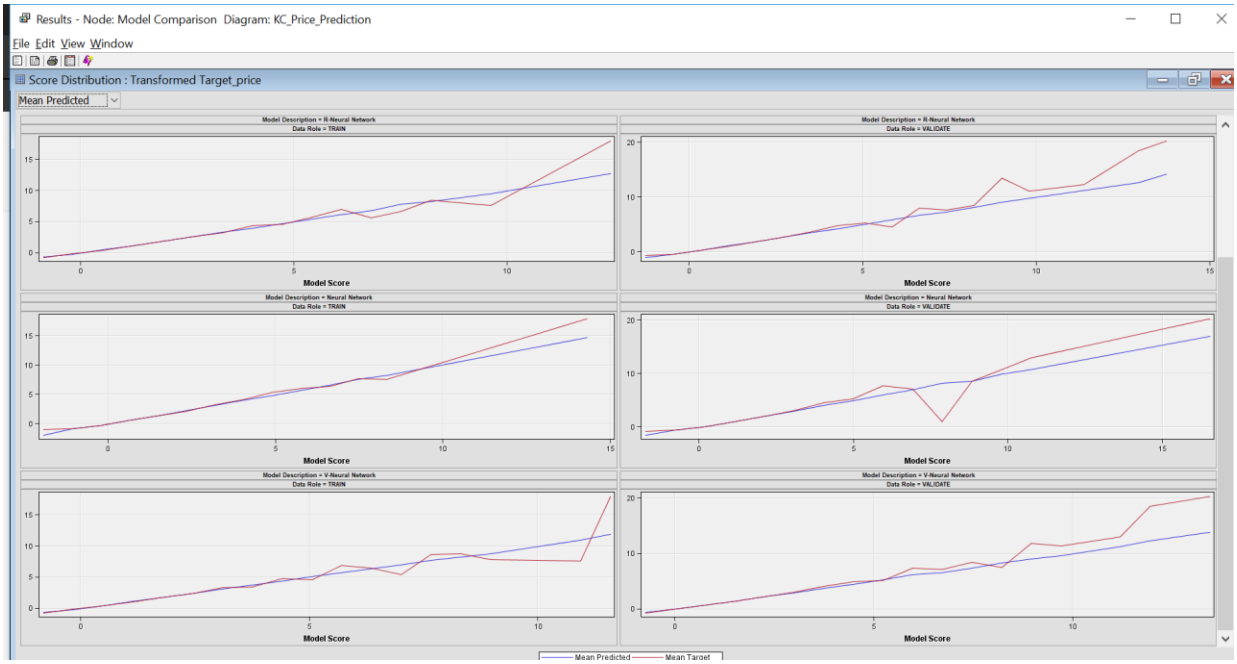
Regression: **0.3226**

Neural Network with Regression input: **0.1905**

Neural Network with Variable selection node input: **0.2102**



The output of Model Comparison node suggests that the model 'Neural Network with Regression node output as an input' is the best model among all 4 models. It has the least Average Square Error of 0.1905.



Even though the Mean Predicted plot for both models V-Neural Network and R-Neural Network showing the similar behavior, yet R-Neural Network is a better model based on least Average Square Error.

Conclusion:

- The model works best when output of regression node is used as an input to Neural Network node for predicting the output.
- Even though we had less number of input variables and less number of weight estimates in Neural Network Model with Variable selection node output, Neural Network with Regression input performed better.

References

1. Data Mining for Business Intelligence: by Galit Shmueli, Nitin Patel and Peter Bruce.

2. Statistics For Management And Economics, 10th EDITION, by G. Keller, 2015.

3. http://support.sas.com/documentation/cdl/en/emgsj/66375/HTML/default/viewer.htm#p0tad07m88xmotn1c78zunl7sm5i.htm

4. http://support.sas.com/documentation/cdl/en/emgsj/66375/HTML/default/viewer.htm#n1docbb4hkr3nwn1ukqzqmme48yn.htm

5. http://support.sas.com/documentation/cdl/en/emgsj/66375/HTML/default/viewer.htm#n0vexerj3lzzr8n11jgynpitzf5j.htm

6. http://blog.minitab.com/blog/adventures-in-statistics-2/how-to-interpret-regression-analysis-results-p-values-and-coefficients

7. https://stats.idre.ucla.edu/other/mult-pkg/faq/general/faq-how-do-i-interpret-odds-ratios-in-logistic-regression/

8. https://statistics.laerd.com/stata-tutorials/linear-regression-using-stata.php