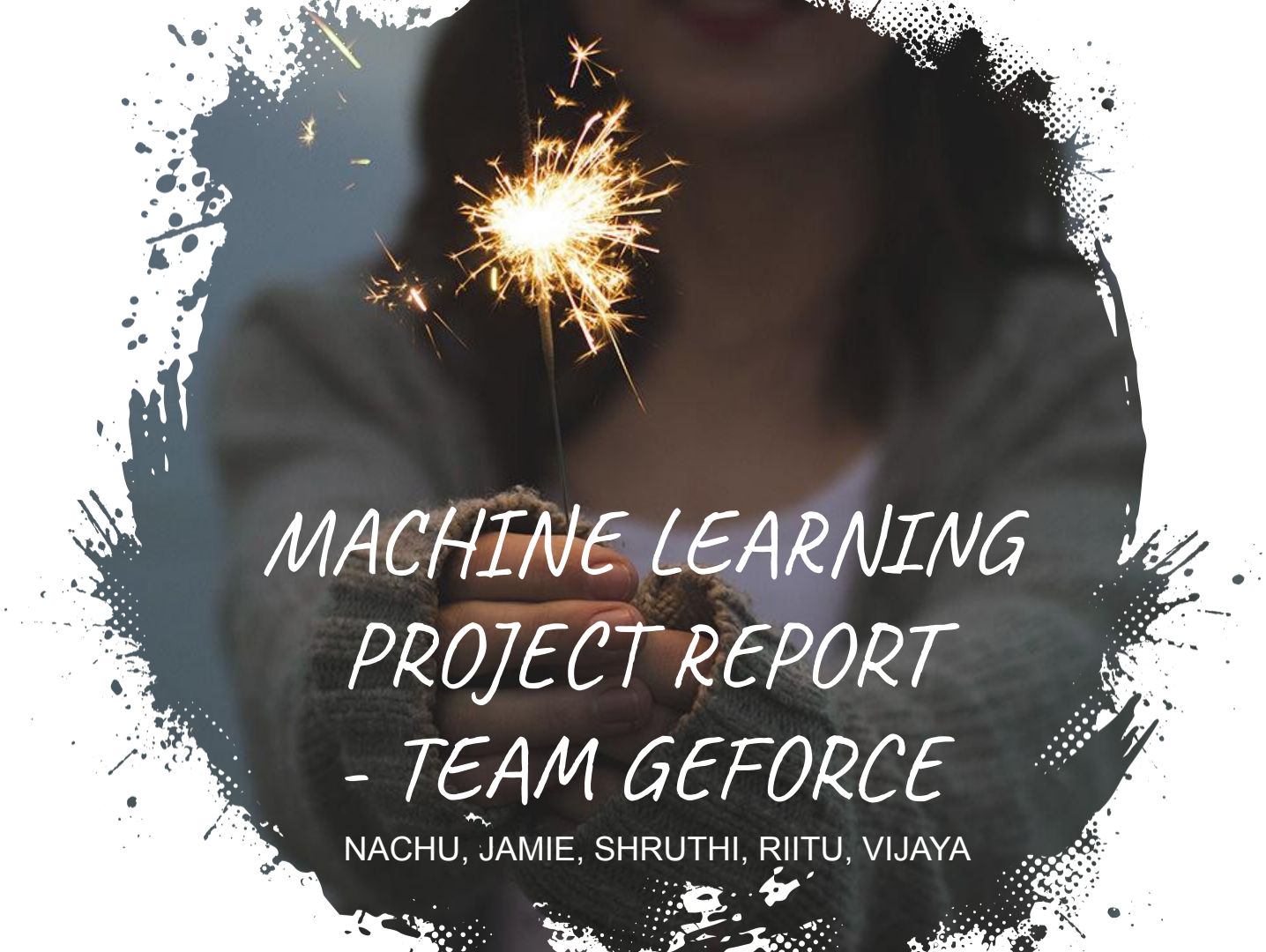


MACHINE LEARNING PROJECT REPORT - TEAM GEFORCE

NACHU, JAMIE, SHRUTHI, RIITU, VIJAYA

Problem Statement

- The Dataset given has historic data of houses sold between May 2014 to May 2015 at King County, Washington State, USA

- The training dataset consists of 21 attributes and 10000 rows. The testing dataset for which the price has to be predicted consists 20 rows.

- The aim is predict the sales of houses in King County with an accuracy of at least 85-90% (relative error less than 15%) and understand which factors are responsible for higher property value - $650K and above

Targeted Solution and Our Role

Team Geforce is property agent team working for URA. Our goal is to predict the price of house as close to actual transaction prices as possible.

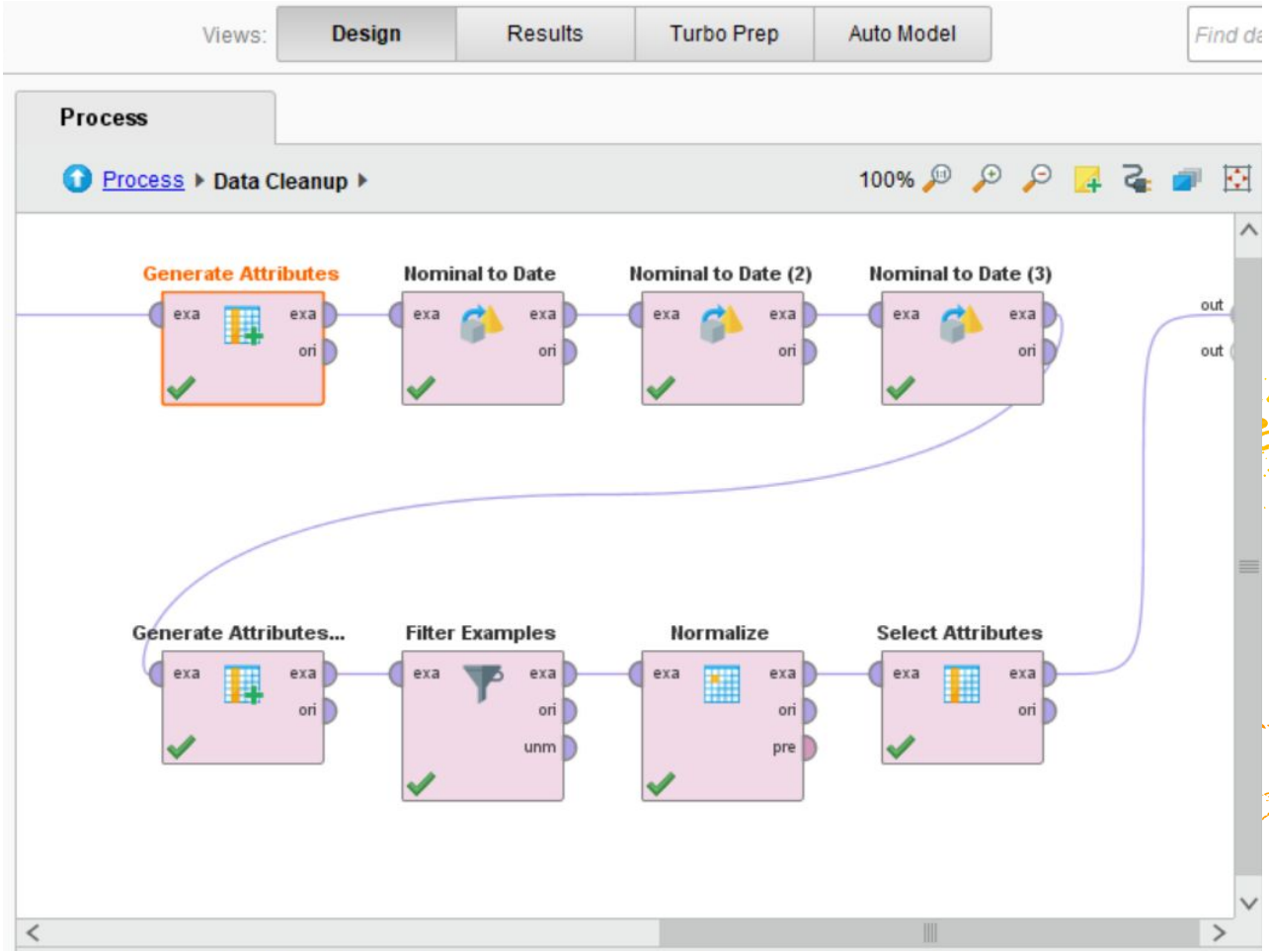
Some of the data provided such as zip code, year built, date of renovation, latitude and longitude are not understood by our algorithm. We did some feature engineering and created new attributes such as Distance to City#, Area Score, House Age*, Latest Date of Renovation so that these new attributes can help our algorithm's learning better

* - House Age is calculated based on last renovated date.
# - Distance to city is calculated assuming that capital city's latitude and longitude as 47, -122.

Modus Operandi:

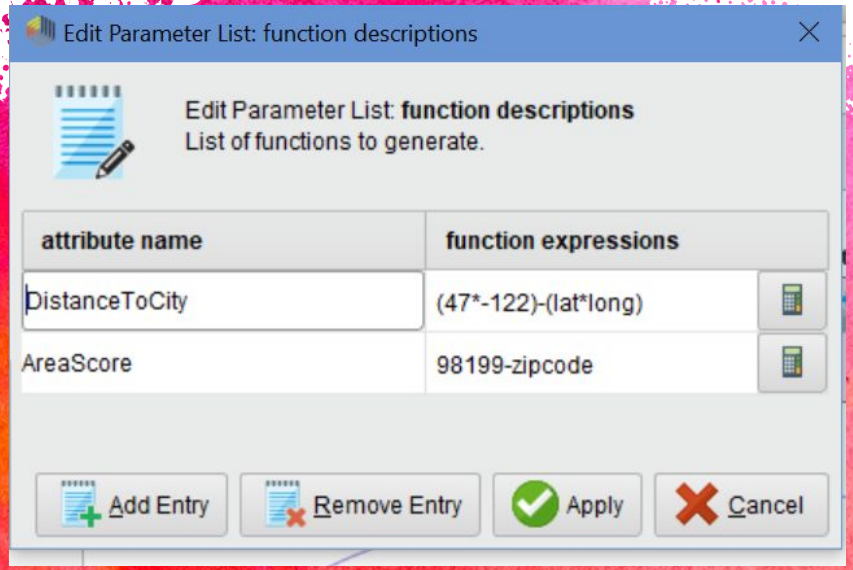
Data Cleanup and Feature Construction

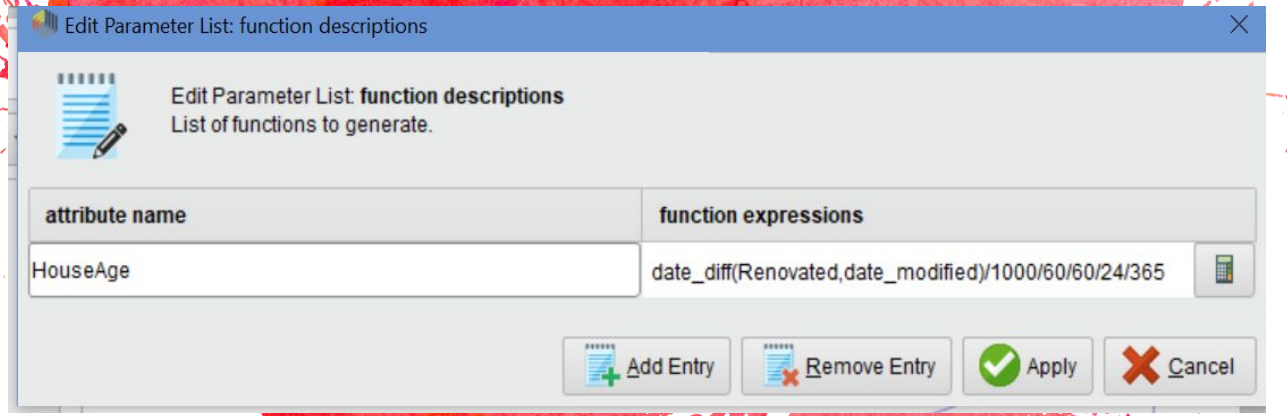
- Converted all the years and date columns to the date format by; setting it to Polynominal while importing the data and then changing it to the date format using the "Nominal to Date" operator.
- Normalized the training set for the 6 "sqft" attributes
- Engineered 4 new attributes using the generate attributes operator:
  Distance to city
  Area Score
  Latest Date of Renovation
  House Age



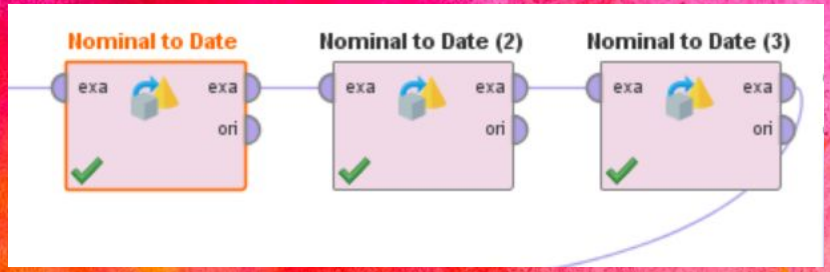
Data Cleanup and Feature Construction

Feature Construction





Nominal to date



Parameters

Nominal to Date (2) (Nominal to Date)

attribute name	Renovated
date type	date
date format	yyyy
time zone	SYSTEM
locale	English (United State...

Hide advanced parameters

Parameters

Nominal to Date

attribute name	date_modified
date type	date
date format	yyyyMMdd
time zone	SYSTEM
locale	English (United State...

Hide advanced parameters

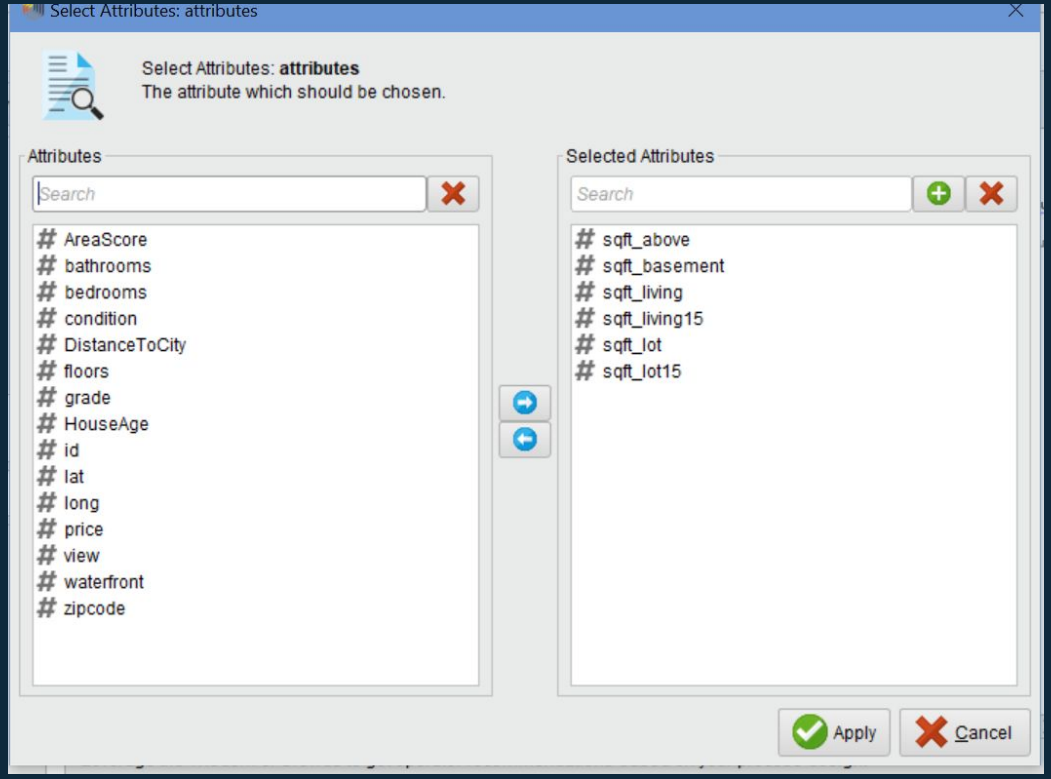
Parameters

Nominal to Date (3) (Nominal to Date)

attribute name	yr_built
date type	date
date format	yyyy
time zone	SYSTEM
locale	English (United S...

Hide advanced parameters

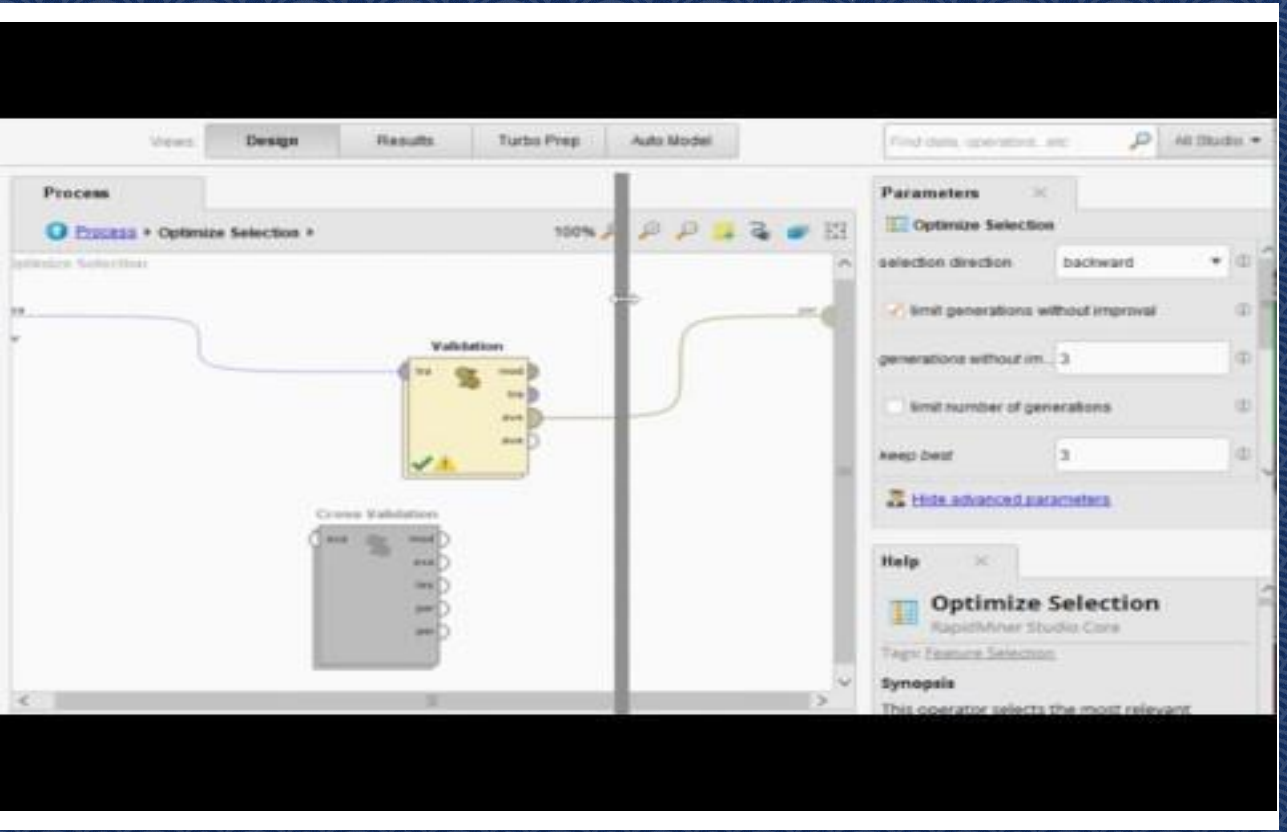
Normalizing only 6 attributes to maintain proportionality



DATA CLEANUP RESULTS

DistanceTo...	AreaScore	date_modifi...	Renovated	yr_built	HouseAge
74.577	21	Oct 13, 2014	Jan 1, 1955	Jan 1, 1955	59.822
103.185	74	Dec 9, 2014	Jan 1, 1991	Jan 1, 1951	23.953
101.147	171	Feb 25, 2015	Jan 1, 1933	Jan 1, 1933	82.205
82.213	63	Dec 9, 2014	Jan 1, 1965	Jan 1, 1965	49.970
77.392	125	Feb 18, 2015	Jan 1, 1987	Jan 1, 1987	28.151
80.282	146	May 12, 2014	Jan 1, 2001	Jan 1, 2001	13.367
53.254	196	Jun 27, 2014	Jan 1, 1995	Jan 1, 1995	19.499
64.893	1	Jan 15, 2015	Jan 1, 1963	Jan 1, 1963	52.074
78.512	53	Apr 15, 2015	Jan 1, 1960	Jan 1, 1960	55.323
46.413	161	Mar 12, 2015	Jan 1, 2003	Jan 1, 2003	12.200
80.188	192	Apr 3, 2015	Jan 1, 1965	Jan 1, 1965	50.285
98.105	84	May 27, 2014	Jan 1, 1942	Jan 1, 1942	72.449
103.144	171	May 28, 2014	Jan 1, 1927	Jan 1, 1927	87.463

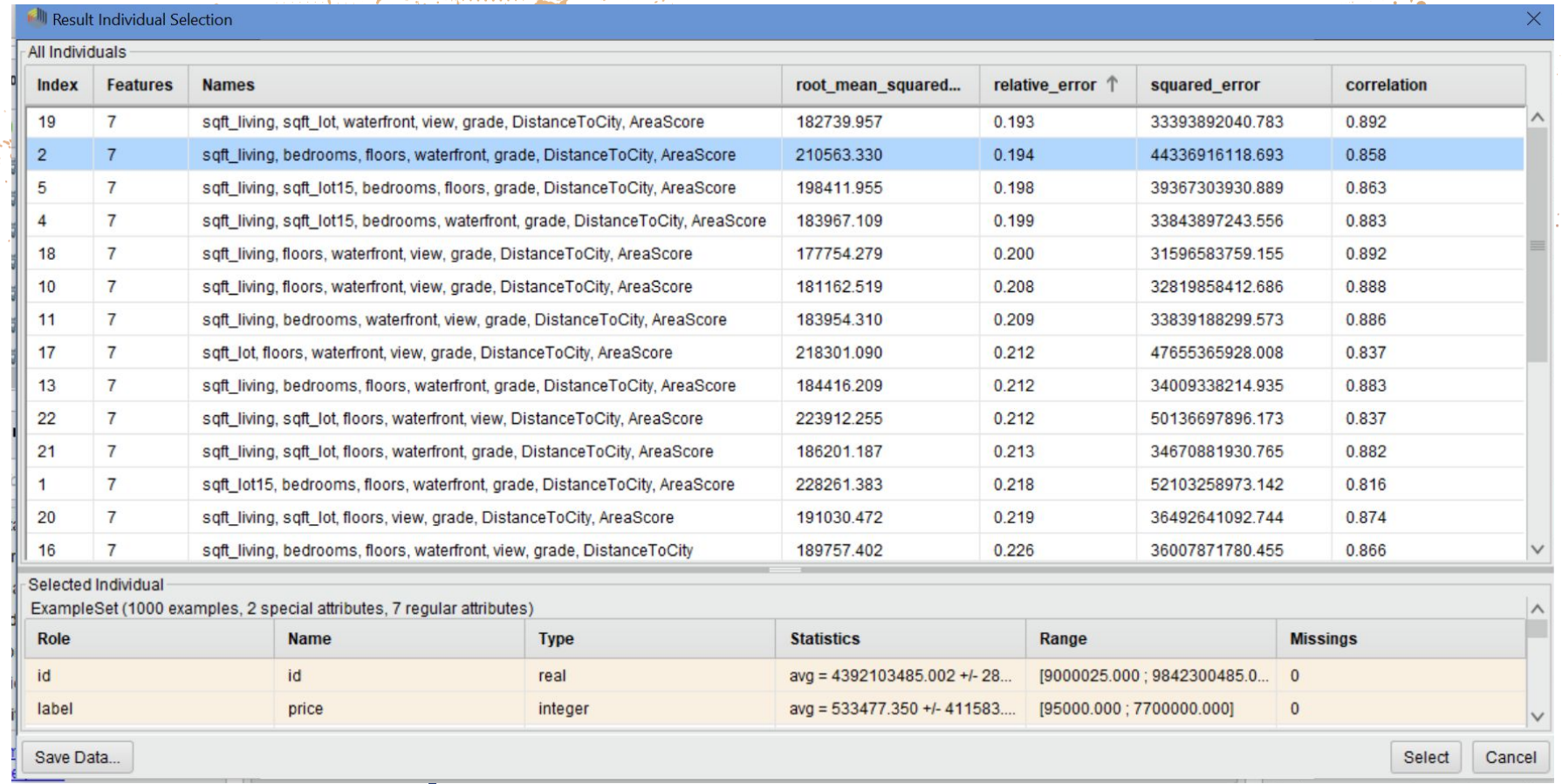
OPTIMIZE SELECTION

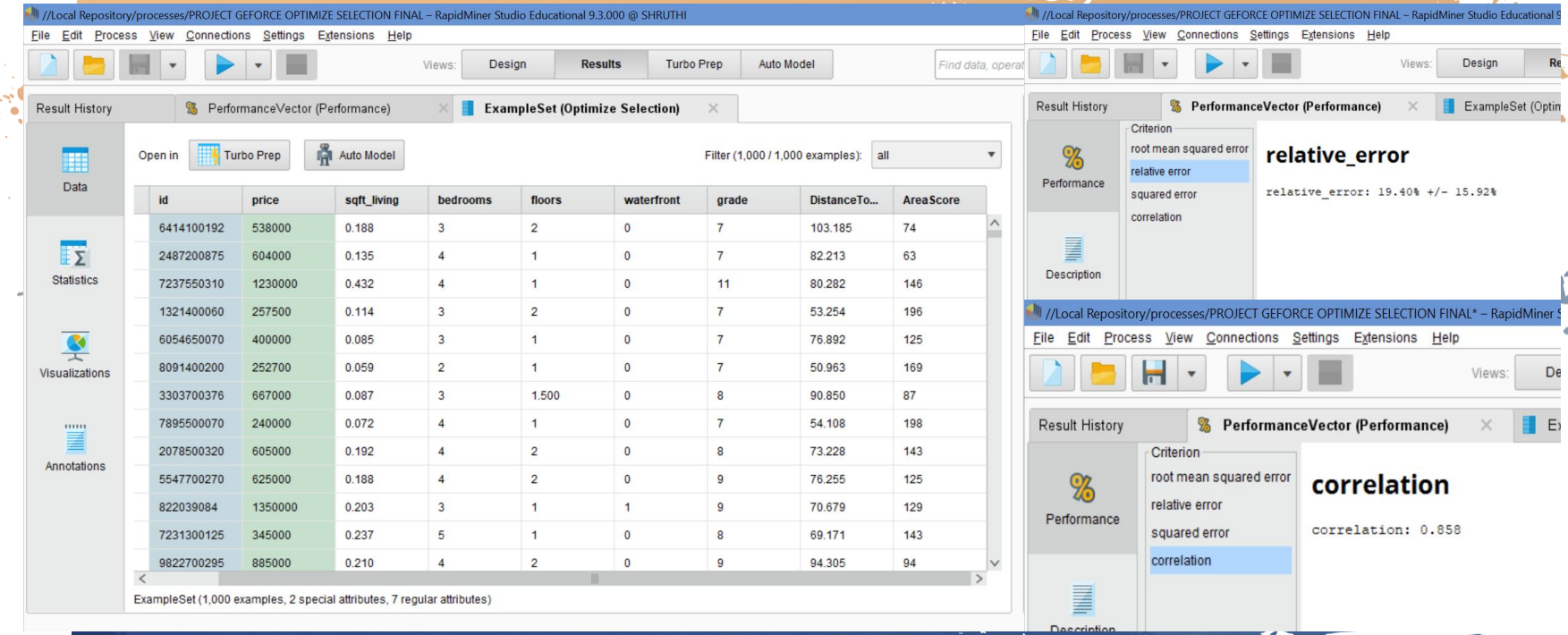


Done to choose the best combination of columns for our model .

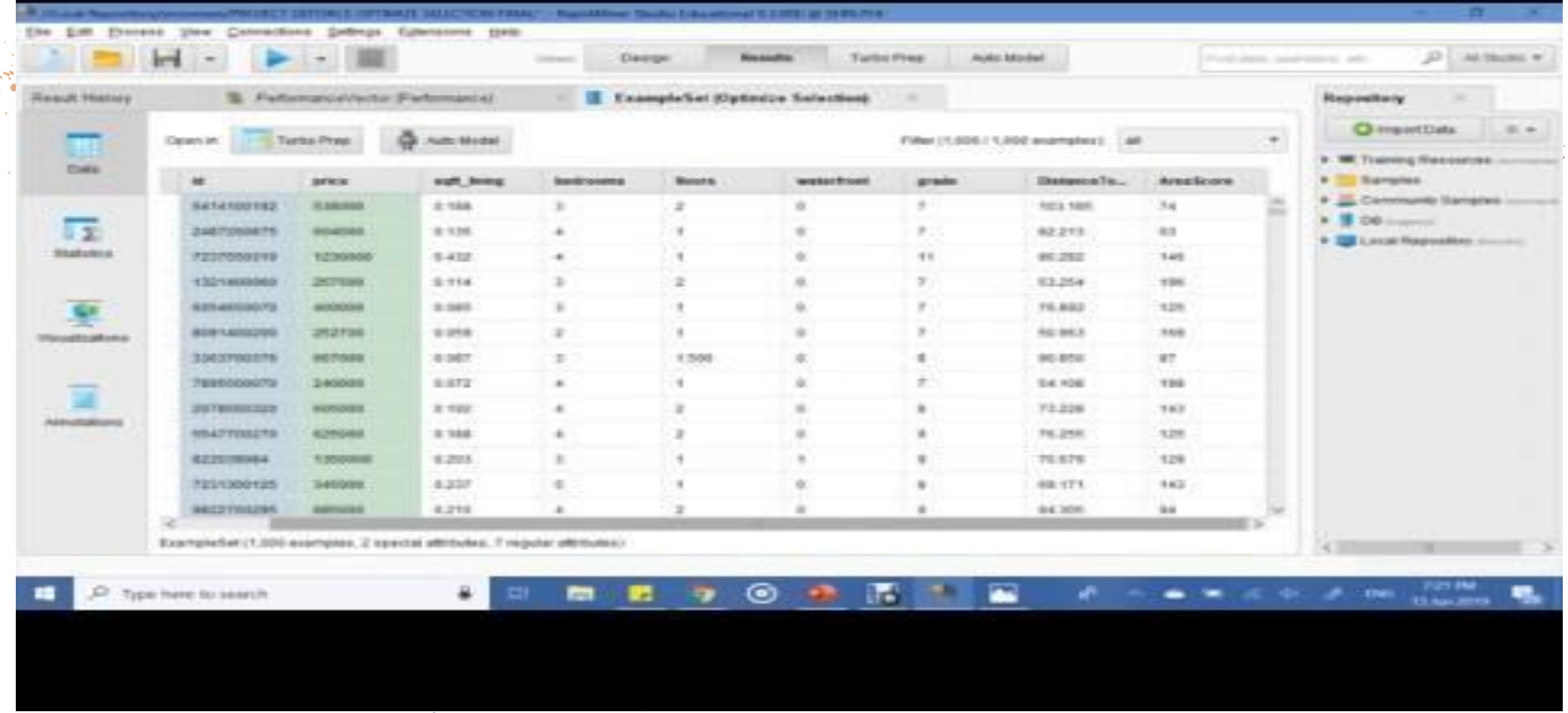
RESULT:
Relative error of 19.4% with 7 attributes

Used Optimize Selection operator for choosing the best attribute combination to get optimal result

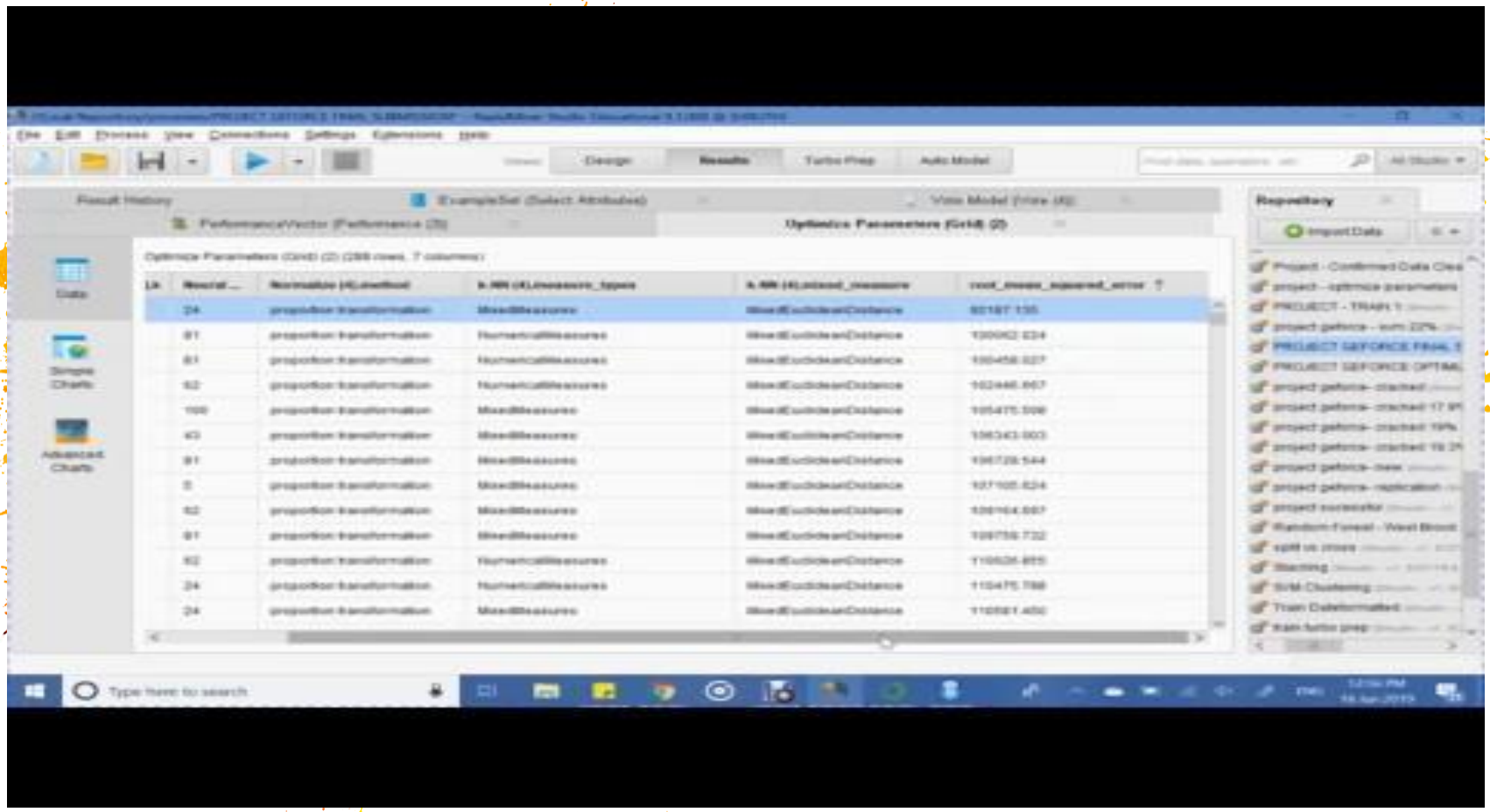



//Local Repository/processes/PROJECT GEFORCE OPTIMIZE SELECTION FINAL – RapidMiner Studio Educational 9.3.000 @ SHRUTHI
File Edit Process View Connections Settings Extensions Help
Views: Design Results Turbo Prep Auto Model
Result History
PerformanceVector (Performance)
ExampleSet (Optimize Selection)
Data
Statistics
Visualizations
Annotations
Open in Turbo Prep Auto Model
Filter (1,000 / 1,000 examples): all
id price sqft_living bedrooms floors waterfront grade DistanceTo... AreaScore
6414100192 538000 0.188 3 2 0 7 103.185 74
2487200875 604000 0.135 4 1 0 7 82.213 63
7237550310 1230000 0.432 4 1 0 11 80.282 146
1321400060 257500 0.114 3 2 0 7 53.254 196
6054650070 400000 0.085 3 1 0 7 76.892 125
8091400200 252700 0.059 2 1 0 7 50.963 169
3303700376 667000 0.087 3 1.500 0 8 90.850 87
7895500070 240000 0.072 4 1 0 7 54.108 198
2078500320 605000 0.192 4 2 0 8 73.228 143
5547700270 625000 0.188 4 2 0 9 76.255 125
822039084 1350000 0.203 3 1 1 9 70.679 129
7231300125 345000 0.237 5 1 0 8 69.171 143
9822700295 885000 0.210 4 2 0 9 94.305 94
ExampleSet (1,000 examples, 2 special attributes, 7 regular attributes)
//Local Repository/processes/PROJECT GEFORCE OPTIMIZE SELECTION FINAL – RapidMiner Studio Educational 9
File Edit Process View Connections Settings Extensions Help
Views: Design
Result History
PerformanceVector (Performance)
ExampleSet (Optim
Performance
Description
Criterion
root mean squared error
relative error
squared error
correlation
relative_error
relative_error: 19.40% +/- 15.92%
//Local Repository/processes/PROJECT GEFORCE OPTIMIZE SELECTION FINAL* – RapidMiner S
File Edit Process View Connections Settings Extensions Help
Views: De
Result History
PerformanceVector (Performance)
Performance
Description
Criterion
root mean squared error
relative error
squared error
correlation
correlation
correlation: 0.858

Optimize Selection - Process Runthrough



After using the optimize selection operator we have chosen the feature combination that gave the lowest relative error and fed it into an optimize parameters operator for hyperparameter tuning. This then gave us the best parameter settings for a model with a relative error of 13.89% tested on the testing dataset.



Result History | ExampleSet (Select Attributes) | Vote Model (Vote (4)) | PerformanceVector (Performance (3)) | **Optimize Parameters (Grid) (2)**

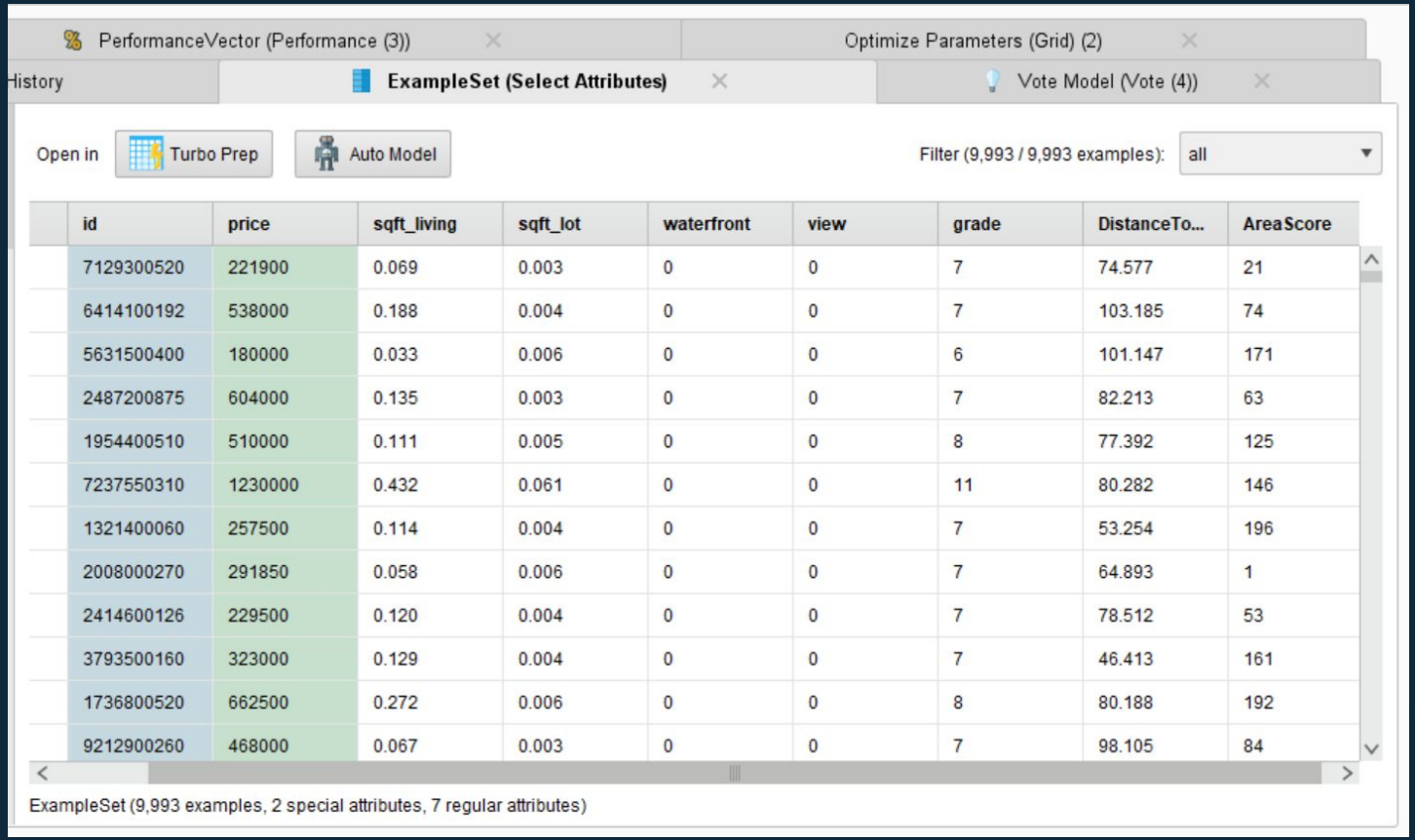
Data | Simple Charts | Advanced Charts

Optimize Parameters (Grid) (2) (288 rows, 7 columns)

iteration	k-NN (4).k	Neural ...	Normalize (4).method	k-NN (4).measure_types	k-NN (4).mixed_measure	root_mean_squared_... ↑
41	42	24	proportion transformation	MixedMeasures	MixedEuclideanDistance	92187.135
194	42	81	proportion transformation	NumericalMeasures	MixedEuclideanDistance	100062.024
193	3	81	proportion transformation	NumericalMeasures	MixedEuclideanDistance	100458.027
192	80	62	proportion transformation	NumericalMeasures	MixedEuclideanDistance	102446.667
52	3	100	proportion transformation	MixedMeasures	MixedEuclideanDistance	105475.508
45	80	43	proportion transformation	MixedMeasures	MixedEuclideanDistance	106343.003
51	80	81	proportion transformation	MixedMeasures	MixedEuclideanDistance	106728.544
39	80	5	proportion transformation	MixedMeasures	MixedEuclideanDistance	107105.624

The 41st iteration with a k of 42 for the k-NN algorithm , 24 training cycles for the neural network, using the proportion transformation with the mixed euclidean distance in the mixed measures to give us our chosen root mean squared error of 92187.135

OUR FINAL TRAINED MODEL WITHOUT PREDICTIONS



id	price	sqft_living	sqft_lot	waterfront	view	grade	DistanceTo...	AreaScore
7129300520	221900	0.069	0.003	0	0	7	74.577	21
6414100192	538000	0.188	0.004	0	0	7	103.185	74
5631500400	180000	0.033	0.006	0	0	6	101.147	171
2487200875	604000	0.135	0.003	0	0	7	82.213	63
1954400510	510000	0.111	0.005	0	0	8	77.392	125
7237550310	1230000	0.432	0.061	0	0	11	80.282	146
1321400060	257500	0.114	0.004	0	0	7	53.254	196
2008000270	291850	0.058	0.006	0	0	7	64.893	1
2414600126	229500	0.120	0.004	0	0	7	78.512	53
3793500160	323000	0.129	0.004	0	0	7	46.413	161
1736800520	662500	0.272	0.006	0	0	8	80.188	192
9212900260	468000	0.067	0.003	0	0	7	98.105	84

ExampleSet (9,993 examples, 2 special attributes, 7 regular attributes)

Our final relative error: 13.89% ± 10.90%

Solution Designed

- ✓ Team Geforce of URA created a Machine Learning model using Split Validation and Voting on 4 algorithms such as k-NN, Linear Regression, Decision Tree and Neural Network to arrive at this solution
- ✓ With the sample data of 1000 rows we achieved 14.53% relative error and with the full training dataset (10000 rows) we achieved 13.89%
- ✓ Also, we achieved a Root Mean Squared Error of 92187.135
- ✓ With this combination of algorithms, parameter settings, generate attributes and attribute selections we are confident that Cross Validation will be able to achieve much better results than this in a high configuration server/laptop
- ✓ This model will be able to predict results as close as 85% to 90% accuracy for the expected price range of the houses in King County, Washington state, USA

Solution Designed

- Based on the analysis done by Team Geforce the factors responsible for higher property value - $650K and above are:

1. Sqft Living
2. Sqft Lot
3. Waterfront
4. View
5. Grade
6. Distance to City
7. Area Score