

Table of Contents

1	Data Pre-processing	4
2	Modelling.....	17
2.1	HP Regression	17
2.2	Dmine Regression	21
2.3	Simple Linear Regression	25
2.4	HP Neural Network.....	33
3	Model Selection.....	44
4	Discussion & Conclusion	45

Figure 1: Original Dataset.....	4
Figure 2: Pre-processed Dataset.....	5
Figure 3: File Import Node	6
Figure 4: Variables Type & Role.....	6
Figure 5: Data Exploration Setting	7
Figure 6: Bedrooms Variable.....	8
Figure 7: Bathrooms Variable.....	8
Figure 8: Parking Variable.....	9
Figure 9: Date Sold Variable	9
Figure 10: Property Type Variable	10
Figure 11: Suburb Variable.....	10
Figure 12: Price Target Variable.....	11
Figure 13: Sample Statistics.....	11
Figure 14: Result of File Import	12
Figure 15: Metadata Node	13
Figure 16: Metadata Result.....	13
Figure 17: Imputation	14
Figure 18: Data Partition Node	15
Figure 19: Data Partition Result	16
Figure 20: High Performance Regression Model	17
Figure 21: ANOVA of HP Regression	17
Figure 22: HP Regression Parameter Estimates	18
Figure 23: HP Regression Score Rankings Matrix	19
Figure 24: HP Regression T-Value.....	19
Figure 25: HP Regression VIF.....	20
Figure 26: Dmine Regression	21
Figure 27: Details of Dmine Regression.....	22
Figure 28: Dmine Regression ANOVA.....	23
Figure 29: Dmine Regression Score Rankings Matrix	23
Figure 30: Dmine Regression Sequential R-square	24
Figure 31: Linear Regression.....	25
Figure 32: Linear Regression ANOVA	26
Figure 33: Analysis of Maximum Likelihood Estimates	27
Figure 34: Linear Regression Fit Statistics.....	28

Figure 35: Assessment Score Rankings	29
Figure 36: Model Comparison 1	30
Figure 37: Model Comparison Fit Statistics	30
Figure 38: Model Comparison Fit Statistics Table	31
Figure 39: Stepwise, Forward, Backward Linear Regression.....	32
Figure 40: Model Comparison 2	32
Figure 41: HP Neural Network	33
Figure 42: HP Neural Network Score Ranking Matrix	34
Figure 43: HP Neural Network Structure	35
Figure 44: HP Neural Network Heat Map	36
Figure 45: HP Neural Network Iteration Plot	37
Figure 46: HP Neural Network Fit Statistics	37
Figure 47: Model Comparison 3	38
Figure 48: HP Neural Network Neuron	38
Figure 49: Model Comparison 4	39
Figure 50: HP Neural Network Architecture	39
Figure 51: Model Comparison 5	40
Figure 52: HP Neural 6N Architecture	40
Figure 53: HP Neural 6N Heat Map	41
Figure 54: HP Neural 6N Iteration Plot	42
Figure 55: HP Neural 6N Score Ranking Matrix.....	42
Figure 56: HP Neural 6N Fit Statistics	43
Figure 57: Overall Process Flow Diagram.....	43
Figure 58: Dataset Observations.....	45
Figure 59: House Price Trend	46
Figure 60: Location of Expensive House.....	47

1 Data Pre-processing

datesold	price	suburb	postcode	lat	lon	parking	bathrooms	bedrooms	propertyType	suburbid
9/6/2000	223000	Nicholls	2913	NULL	NULL	2	2	4	house	ACT708
1/1/2001	350000	Ngunnawa	2913	NULL	NULL	1	NULL	3	house	ACT706
11/12/2003	550000	Weston	2611	NULL	NULL	2	NULL	4	house	ACT441
21/9/2005	276000	Isabella Pl	2905	NULL	NULL	1	1	3	house	ACT612
1/11/2005	400000	Conder	2906	NULL	NULL	2	NULL	5	house	ACT613
22/11/2005	350000	Theodore	2905	NULL	NULL	2	2	3	house	ACT610
8/12/2005	447500	Conder	2906	NULL	NULL	2	NULL	5	house	ACT613
14/12/2005	355000	Conder	2906	NULL	NULL	2	NULL	4	house	ACT613
22/12/2005	350000	Conder	2906	NULL	NULL	1	NULL	4	house	ACT613
9/1/2006	435000	Banks	2906	NULL	NULL	2	NULL	4	house	ACT614
17/1/2006	270000	Gordon	2906	NULL	NULL	1	1	3	house	ACT616
7/2/2006	272500	Chisholm	2905	NULL	NULL	1	1	3	house	ACT609
17/2/2006	280500	Gordon	2906	NULL	NULL	1	1	3	house	ACT616
20/2/2006	385000	Banks	2906	NULL	NULL	2	NULL	4	house	ACT614
24/2/2006	310000	Waramang	2611	NULL	NULL	2	NULL	3	house	ACT442
1/3/2006	255000	Conder	2906	NULL	NULL	0	2	3	house	ACT613
31/3/2006	250000	Theodore	2905	NULL	NULL	1	1	3	house	ACT610
2/5/2006	250000	Conder	2906	NULL	NULL	1	2	3	house	ACT613
2/5/2006	420000	Conder	2906	NULL	NULL	2	2	4	house	ACT613
23/5/2006	250000	Isabella Pl	2905	NULL	NULL	1	1	3	house	ACT612
13/6/2006	270000	Gordon	2906	NULL	NULL	1	1	3	house	ACT616
29/6/2006	308000	Banks	2906	NULL	NULL	2	NULL	3	house	ACT614
18/7/2006	350000	Conder	2906	NULL	NULL	1	2	3	house	ACT613
20/7/2006	425000	Wanniasse	2903	NULL	NULL	2	NULL	5	house	ACT602
4/8/2006	319000	Gilmore	2905	NULL	NULL	1	1	3	house	ACT608
17/8/2006	292500	Banks	2906	NULL	NULL	2	2	3	house	ACT614
30/8/2006	450000	Conder	2906	NULL	NULL	2	1	5	house	ACT613
8/9/2006	275000	Gordon	2906	NULL	NULL	1	1	3	house	ACT616
11/9/2006	250000	Isabella Pl	2905	NULL	NULL	1	1	3	house	ACT612
29/9/2006	270000	Gordon	2906	NULL	NULL	1	1	3	house	ACT616
29/9/2006	545000	Banks	2906	NULL	NULL	2	2	4	house	ACT614
10/10/2006	250000	Gowrie	2904	NULL	NULL	1	1	2	house	ACT604

Figure 1: Original Dataset

As shown in Figure 1 is the original dataset of house price in Canberra, Australia from year 2000 to 2019. Before importing the dataset to SAS Enterprise Miner (SAS EM), the dataset is pre-processed by manually deleting the noise, for example the rows contain 'NULL' in column price are deleted because price is the target for the prediction, without the price the observations are useless. Furthermore, the 'latitude' and 'longitude' columns are deleted because both columns are not useful for prediction and both columns contain too much 'NULL'. In addition, the rows with 'bathrooms' column with 'NULL' are deleted as well, because the columns with 'NULL' SAS EM will assume the variable type as character, therefore rows with 'NULL' are deleted to ensure the variable is numerical type for regression task.

datesold	price	suburb	postcode	parking	bathroom	bedrooms	propertyType	suburbid
9/6/2000	223000	Nicholls	2913	2	2	4	house	ACT708
21/9/2005	276000	Isabella Plains	2905	1	1	3	house	ACT612
22/11/2005	350000	Theodore	2905	2	2	3	house	ACT610
17/1/2006	270000	Gordon	2906	1	1	3	house	ACT616
7/2/2006	272500	Chisholm	2905	1	1	3	house	ACT609
17/2/2006	280500	Gordon	2906	1	1	3	house	ACT616
1/3/2006	255000	Conder	2906	0	2	3	house	ACT613
31/3/2006	250000	Theodore	2905	1	1	3	house	ACT610
2/5/2006	250000	Conder	2906	1	2	3	house	ACT613
2/5/2006	420000	Conder	2906	2	2	4	house	ACT613
23/5/2006	250000	Isabella Plains	2905	1	1	3	house	ACT612
13/6/2006	270000	Gordon	2906	1	1	3	house	ACT616
18/7/2006	350000	Conder	2906	1	2	3	house	ACT613
4/8/2006	319000	Gilmore	2905	1	1	3	house	ACT608
17/8/2006	292500	Banks	2906	2	2	3	house	ACT614
30/8/2006	450000	Conder	2906	2	1	5	house	ACT613
8/9/2006	275000	Gordon	2906	1	1	3	house	ACT616
11/9/2006	250000	Isabella Plains	2905	1	1	3	house	ACT612
29/9/2006	270000	Gordon	2906	1	1	3	house	ACT616
29/9/2006	545000	Banks	2906	2	2	4	house	ACT614
10/10/2006	250000	Gowrie	2904	1	1	2	house	ACT604
23/10/2006	250000	Isabella Plains	2905	1	1	3	house	ACT612
27/10/2006	250000	Gordon	2906	1	1	3	house	ACT616
4/11/2006	250000	Gordon	2906	1	1	3	house	ACT616
16/11/2006	345000	Chisholm	2905	2	2	3	house	ACT609
24/11/2006	320000	Conder	2906	1	1	3	house	ACT613
1/12/2006	350000	Gordon	2906	1	2	3	house	ACT616
19/12/2006	274000	Isabella Plains	2905	1	1	3	house	ACT612
22/12/2006	367000	Bonython	2905	0	1	4	house	ACT619
23/12/2006	250000	Banks	2906	2	1	3	house	ACT614
21/1/2007	450000	Conder	2906	2	2	3	house	ACT613
16/2/2007	250000	Isabella Plains	2905	1	1	3	house	ACT612

Figure 2: Pre-processed Dataset

The dataset is pre-processed as shown in Figure 2, rows with 'NULL' are deleted before importing the file to SAS EM. Although, columns such as 'suburb', 'postcode' and 'suburbid' are not useful for price prediction, it can be rejected in SAS EM during analysis. The pre-processed dataset is saved as excel file for more accurate analysis in SAS EM.

Property	Value
General	
Node ID	FIMPORT2
Imported Data	...
Exported Data	...
Notes	...
Train	
Variables	...
Import File	C:\Users\tehya\OneDrive\De...
Maximum Rows to Import	1000000
Maximum Columns to Import	10000
Delimiter	,
Name Row	Yes
Number of Rows to Skip	0
Guessing Rows	500
File Location	Local
File Type	xlsx
Advanced Advisor	Yes
Rerun	No
Score	
Role	Train
Report	
Summarize	No
Status	
Create Time	9/16/20 6:18 AM
Run ID	
Last Error	
Last Status	
Last Run Time	
Run Duration	
Grid Host	
User-Added Node	No




Figure 3: File Import Node

As shown in Figure 3, the file is imported using 'File Import' node, the file is imported using 'Import File' property with comma as delimiter and excel file type.

Variables - FIMPORT
✕

(none)

▼

☐ not

Equal to

...

Apply

Reset

Columns:

☐ Label
 ☐ Mining
 ☐ Basic
 ☐ Statistics

Name	Role	Level	Report	Order	Drop	Lower Limit	Upper Limit
bathrooms	Input	Interval	No		No	.	.
bedrooms	Input	Interval	No		No	.	.
datesold	Time ID	Interval	No		No	.	.
parking	Input	Interval	No		No	.	.
postcode	Rejected	Interval	No		No	.	.
price	Target	Interval	No		No	.	.
propertyType	Input	Nominal	No		No	.	.
suburb	Input	Nominal	No		No	.	.
suburbid	Rejected	Nominal	No		No	.	.

Explore...

OK

Cancel

Figure 4: Variables Type & Role

Figure 4 shows the role and type for each variable, as mentioned the 'postcode' and 'suburbid' are not useful for prediction therefore it is set to 'Rejected'. Most of the variables are set to interval for regression task.

6

Property	Value
Rows	41777
Columns	9
Library	EMWS6
Member	FIMPORT_DATA
Type	DATA
Sample Method	Top
Fetch Size	Max
Fetched Rows	41777
Random Seed	12345

Apply Plot...

Figure 5: Data Exploration Setting

Before exploring the variables, the setting as shown in Figure 5 must be done to ensure all observations are explored by changing the 'Fetch Size' to max.

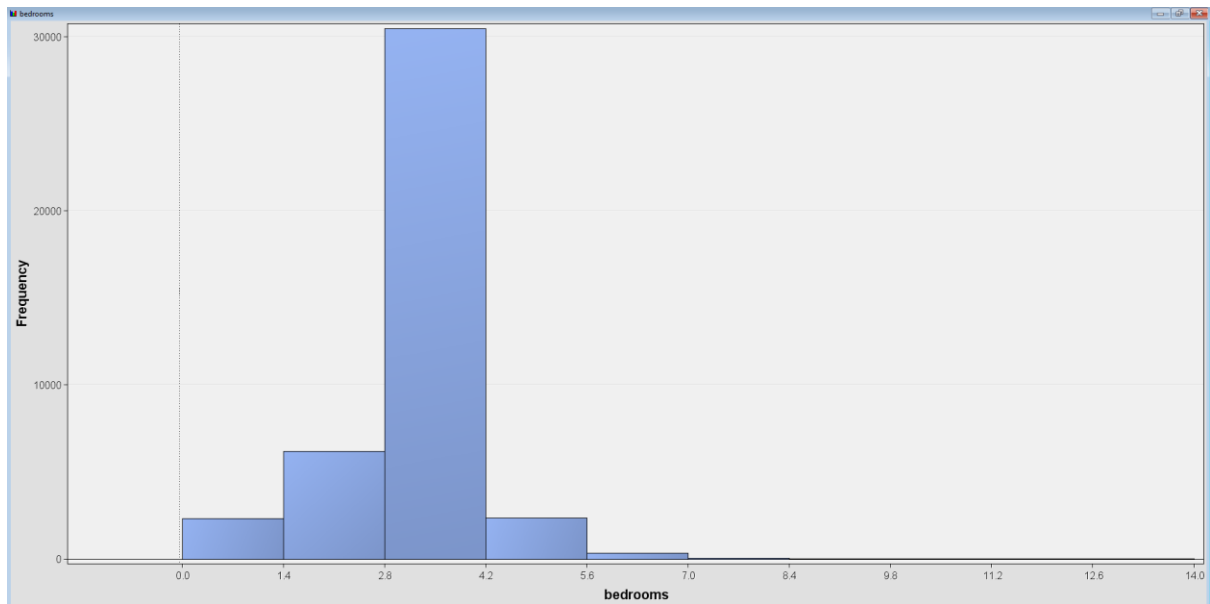


Figure 6: Bedrooms Variable

As shown in Figure 6 is the data exploration of 'bedrooms' variable, most of the houses have between 1 to 7 bedrooms.

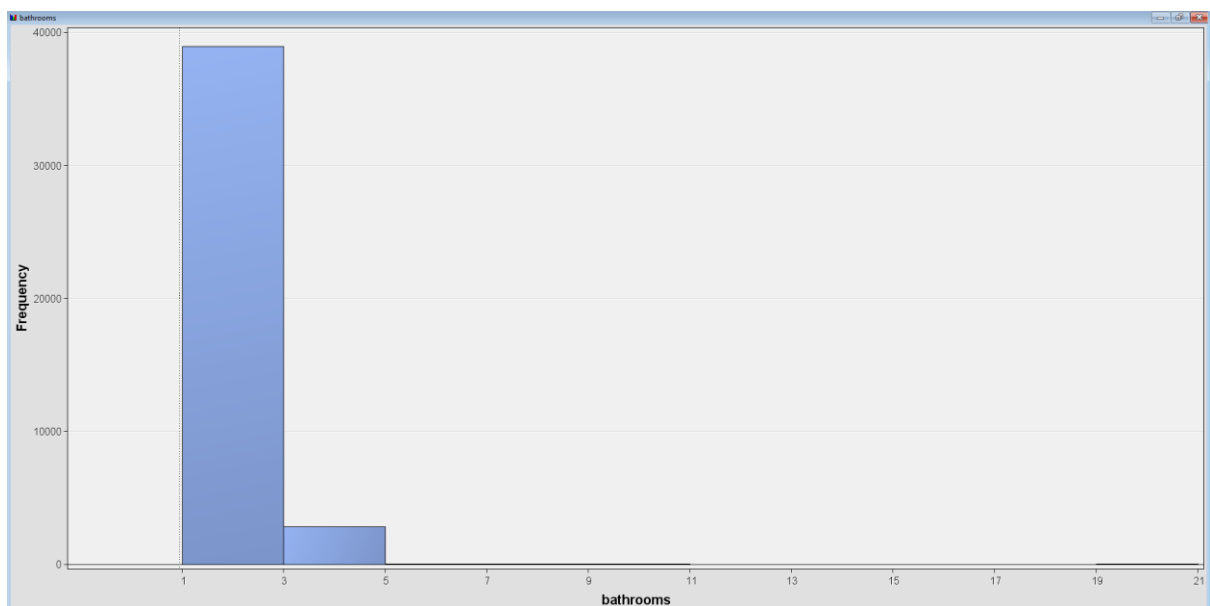


Figure 7: Bathrooms Variable

As shown in Figure 7 is the data exploration of 'bathrooms' variable, most houses in Canberra have between 1 to 5 bathrooms.

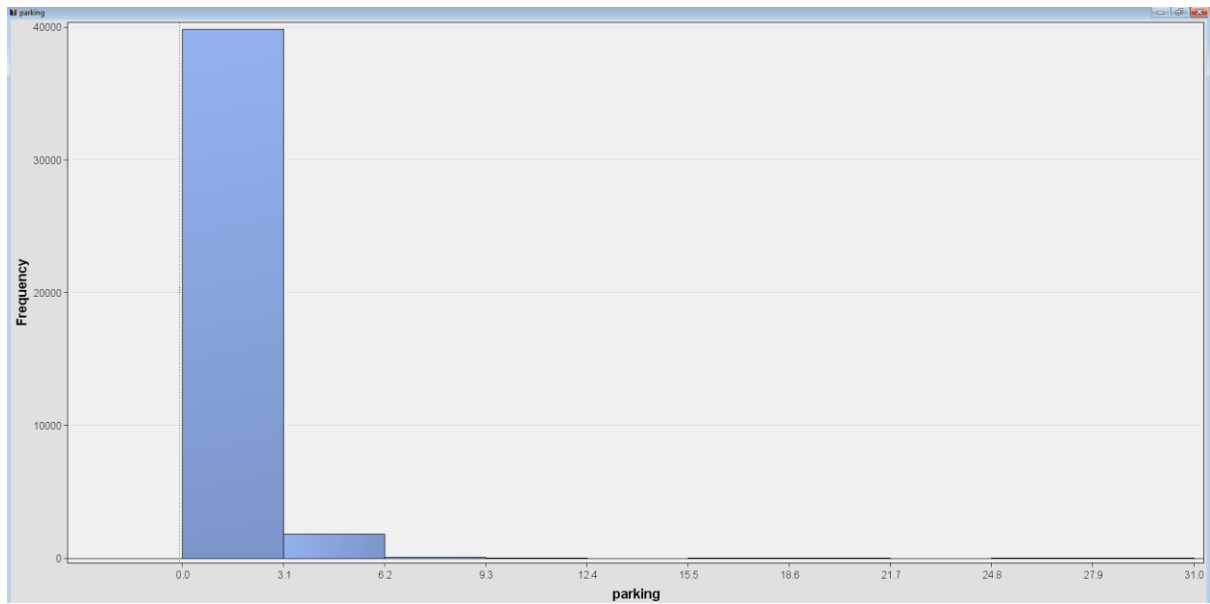


Figure 8: Parking Variable

As shown in Figure 8 is the data exploration of 'parking' variable, most houses in Canberra have between 2 to 6 parking lots.

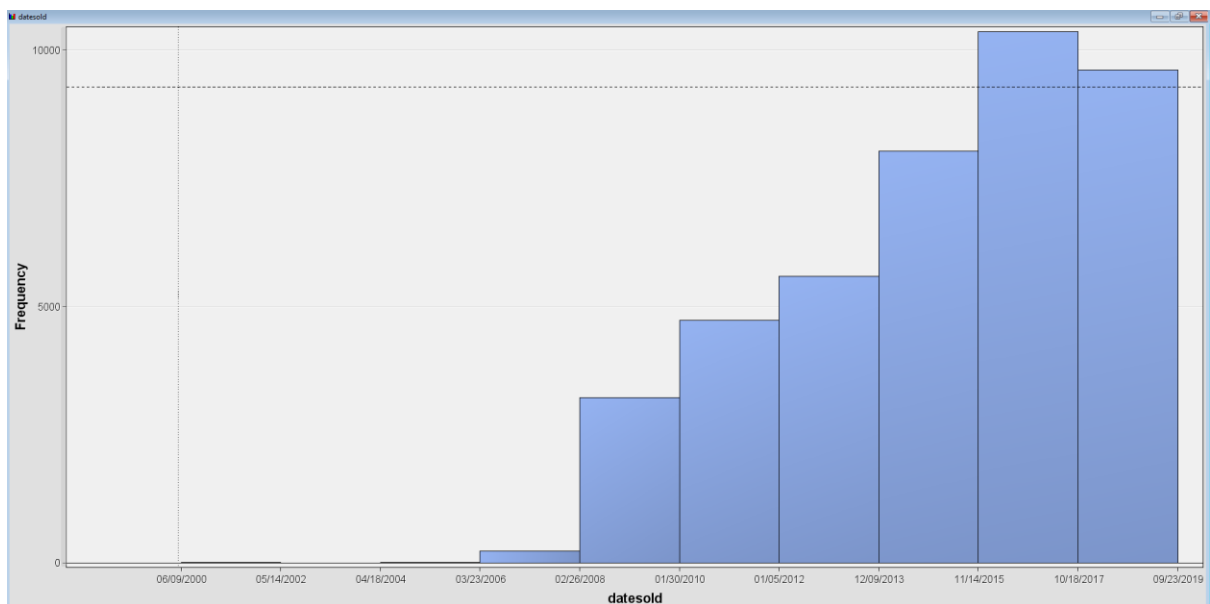


Figure 9: Date Sold Variable

As shown in Figure 9 is the data exploration of 'datesold' variable, the house sales in Canberra increase gradually from year 2006 until 2019.

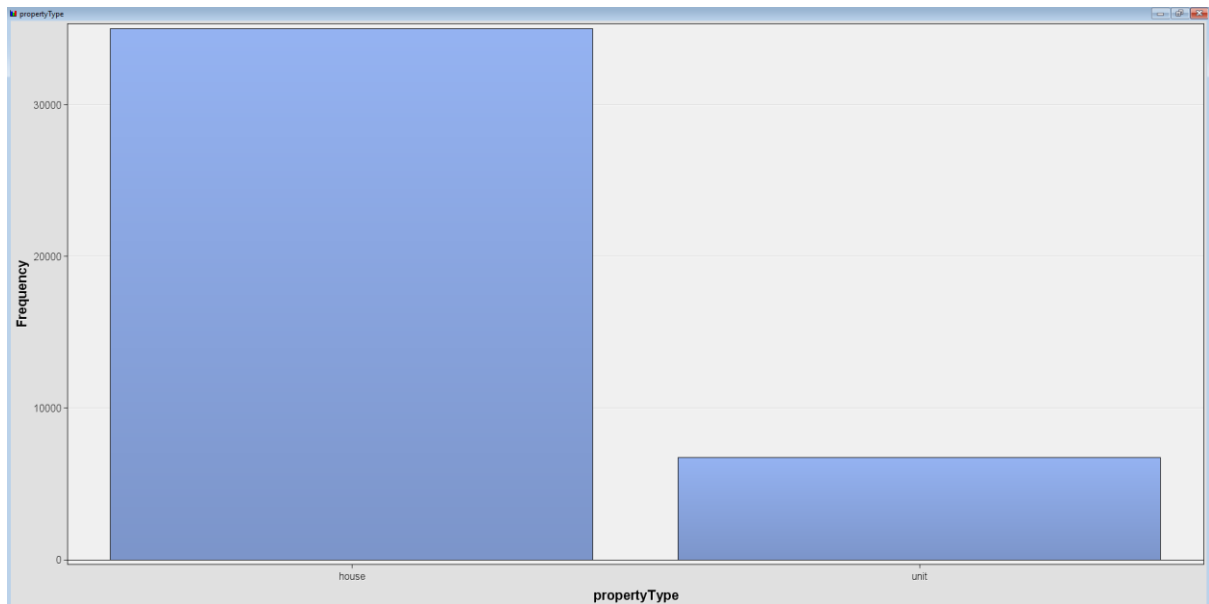


Figure 10: Property Type Variable

As shown in Figure 10 is the data exploration of 'propertyType' variable, majority of the property type in Canberra is house.

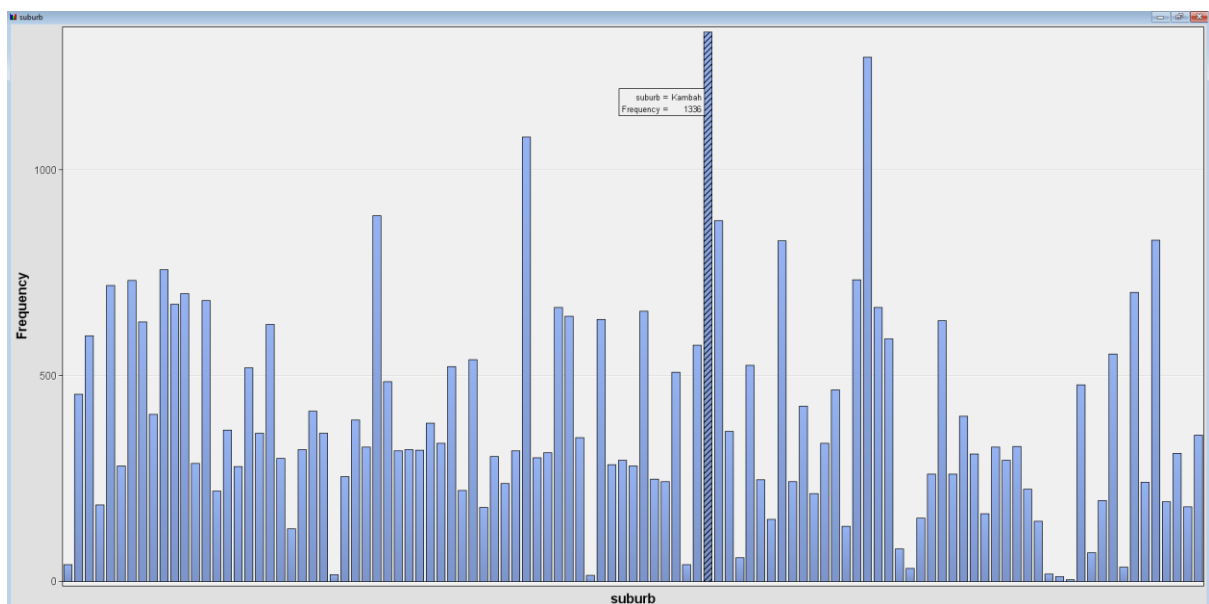


Figure 11: Suburb Variable

As shown in Figure 11 is the data exploration of 'suburb' variable, majority of the houses located in Kambah, Ngunnawal and Gordon suburb, which are the top 3.

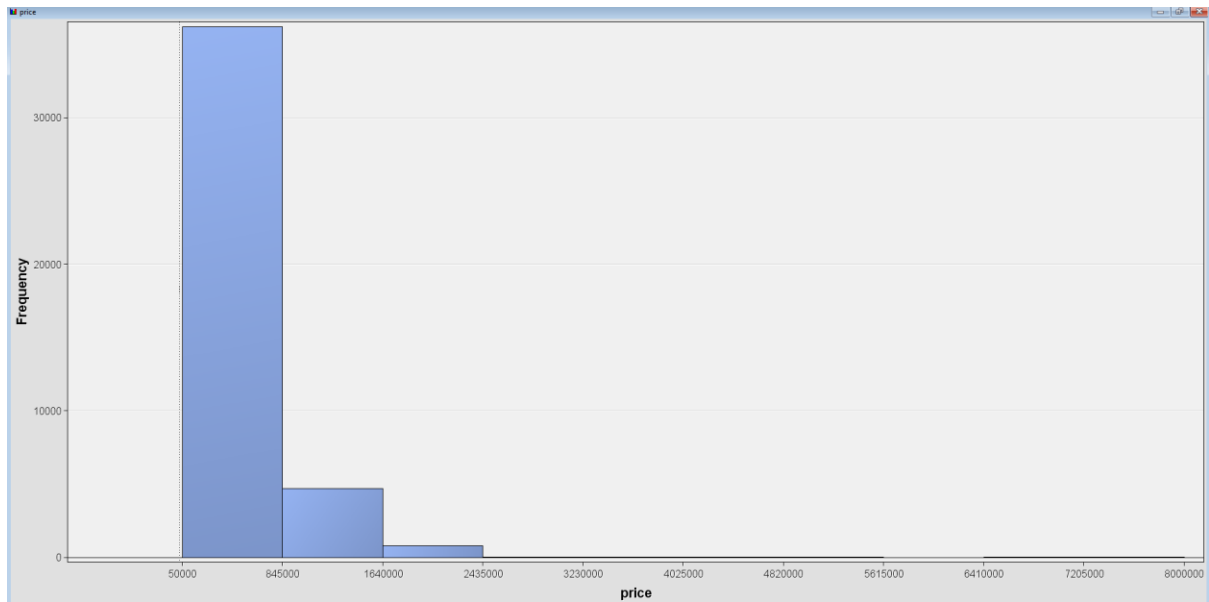


Figure 12: Price Target Variable

As shown in Figure 12 is the data exploration of 'price' variable, majority of the house price in Canberra is between AUD 50, 000 to AUD 84, 500.

Sample Statistics										
Obs #	Variabl...	Label	Type	Percen...	Minimum	Maximum	Mean	Numbe...	Mode ...	Mode
1	propertyT...	propertyT...	CLASS	02	83.86193	HOUSE
2	suburb	suburb	CLASS	0107	3.197932	KAMBAH
3	suburbid	suburbid	CLASS	0107	3.197932	ACT601
4	bathrooms	bathrooms	VAR	0	1	21	1.641334	.	.	.
5	bedrooms	bedrooms	VAR	0	0	14	3.189769	.	.	.
6	datesold	datesold	VAR	0	14770	21815	20130.97	.	.	.
7	parking	parking	VAR	0	0	31	1.758073	.	.	.
8	postcode	postcode	VAR	0	2092	2914	2729.097	.	.	.
9	price	price	VAR	0	50000	8000000	604188.8	.	.	.

Figure 13: Sample Statistics

As shown in Figure 13 is the overall statistics of the dataset, majority of property type are house and most of the houses are located in Kambah suburb. Plus, the highest house price in Canberra can go up to 8 million AUD.

Alphabetic List of Variables and Attributes						
#	Variable	Type	Len	Format	Informat	Label
6	bathrooms	Num	8	BEST.		bathrooms
7	bedrooms	Num	8	BEST.		bedrooms
1	datesold	Num	8	MMDDYY10.		datesold
5	parking	Num	8	BEST.		parking
4	postcode	Num	8	BEST.		postcode
2	price	Num	8	BEST.		price
8	propertyType	Char	5	\$5.	\$5.	propertyType
3	suburb	Char	15	\$15.	\$15.	suburb
9	suburbid	Char	6	\$6.	\$6.	suburbid

* Score Output						

* Report Output						

Exported Attributes for TRAIN Port						
Role	Measurement Level	Frequency Count				
INPUT	INTERVAL	3				
INPUT	NOMINAL	2				
REJECTED	INTERVAL	1				
REJECTED	NOMINAL	1				
TARGET	INTERVAL	1				
TIMEID	INTERVAL	1				

Figure 14: Result of File Import

As shown in Figure 14 is the result of the File Import node, the 'NULL' in variable 'price' and 'bathrooms' are deleted to ensure SAS EM take the variables as numeric type and all numeric variables are set to interval while character variables are set to nominal.

Property	Value
General	
Node ID	Meta2
Imported Data	...
Exported Data	...
Notes	...
Train	
Import Selection	...
Summarize	No
Advanced Advisor	Yes
Rejected Variables	
Hide Rejected Variables	No
Combine Rule	None
Variables	
Train	...
Transaction	...
Validate	...
Test	...
Score	...
Status	
Create Time	9/17/20 4:28 AM
Run ID	
Last Error	
Last Status	
Last Run Time	
Run Duration	
Grid Host	
User-Added Node	No

Metadata

Figure 15: Metadata Node

The next node that link to File Import node is Metadata node as shown in Figure 15, it provides the convenience to modify the metadata in the process flow, for example the role can change from input, rejected or target.

Statistics Table											
Variable Name	Role	Measurement Level	Type	Number of Levels	Percent Missing	Minimum	Maximum	Mean	Standard Deviation	Skewness	Kurtosis
bathrooms	INPUT	NOMINAL	N	9	0						
bedrooms	INPUT	NOMINAL	N	13	0						
datesold	TIMED	INTERVAL	N		0	14770	21815	20130.97	1113.64	-0.50799	-0.78167
parking	INPUT	NOMINAL	N	18	0						
postcode	REJECTED	INTERVAL	N		0	2082	2914	2728.097	146.587	0.398855	-1.80968
price	TARGET	INTERVAL	N		0	50000	800000	604188.8	314261	3.809105	33.32735
propertyType	INPUT	BINARY	C	2	0						
suburb	REJECTED	NOMINAL	C	107	0						
suburbid	REJECTED	NOMINAL	C	107	0						

Figure 16: Metadata Result

As shown in Figure 16 is the result of metadata node, it shows the variable role and type. Moreover, indicates the target variable is right skewed and no missing value in variable because the dataset is pre-processed before using SAS EM.

Property	Value
General	
Node ID	Impt2
Imported Data	...
Exported Data	...
Notes	...
Train	
Variables	...
Nonmissing Variables	No
Missing Cutoff	50.0
<input type="checkbox"/> Class Variables	
Default Input Method	Count
Default Target Method	None
Normalize Values	Yes
<input type="checkbox"/> Interval Variables	
Default Input Method	Mean
Default Target Method	None
<input type="checkbox"/> Default Constant Value	
Default Character Value	
Default Number Value	.
<input type="checkbox"/> Method Options	
Random Seed	12345
Tuning Parameters	...
Tree Imputation	...
Score	
Hide Original Variables	Yes
<input type="checkbox"/> Indicator Variables	
Type	Unique
Source	Imputed Variables
Role	Input
Report	
Validation and Test Data	No
Distribution of Missing	No
Status	
Create Time	9/17/20 4:40 AM
Run ID	
Last Error	
Last Status	
Last Run Time	
Run Duration	
Grid Host	
User-Added Node	No


 Impute

Figure 17: Imputation

As shown in Figure 17 is imputation node, it replaces missing values with the mean of the variable, but since no missing values the node is included in the process flow for complete presentation.

Property	Value
General	
Node ID	Part2
Imported Data	...
Exported Data	...
Notes	...
Train	
Variables	...
Output Type	Data
Partitioning Method	Default
Random Seed	12345
<input checked="" type="checkbox"/> Data Set Allocations	
Training	70.0
Validation	30.0
Test	0.0
Report	
Interval Targets	Yes
Class Targets	Yes
Status	
Create Time	9/17/20 5:58 AM
Run ID	
Last Error	
Last Status	
Last Run Time	
Run Duration	
Grid Host	
User-Added Node	No



Figure 18: Data Partition Node

As shown in Figure 18 is setting of data partition node, the dataset is split into 70% training set and 30% validation set.

Partition Summary

Type	Data Set	Number of Observations
DATA	EMWS6.Impt_TRAIN	41777
TRAIN	EMWS6.Part_TRAIN	29244
VALIDATE	EMWS6.Part_VALIDATE	12533

```

*-----*
* Score Output
*-----*

```

```

*-----*
* Report Output
*-----*

```

Summary Statistics for Interval Targets

Data=DATA

Variable	Maximum	Mean	Minimum	Number of Observations	Missing	Standard Deviation	Label
price	8000000	604188.77885	50000	41777	0	314260.99089	price

Data=TRAIN

Variable	Maximum	Mean	Minimum	Number of Observations	Missing	Standard Deviation	Label
price	8000000	606060.72658	50000	29244	0	317644.29278	price

Data=VALIDATE

Variable	Maximum	Mean	Minimum	Number of Observations	Missing	Standard Deviation	Label
price	5425000	599820.85103	50000	12533	0	306189.18073	price

Figure 19: Data Partition Result

As shown in Figure 19 is the result of data partition node, total of 41777 observations are split into 29244 observations for training set and 12533 observations for validation set. In addition, both training and validation sets have mean price of AUD 60,000.

2 Modelling

2.1 HP Regression

Property	Value
General	
Node ID	HPReq2
Imported Data	
Exported Data	
Notes	
Train	
Variables	
Equation	
Main Effects	Yes
Two-Factor Interactions	No
Polynomial Terms	No
Polynomial Degree	2
Suppress Intercept	No
Use Missing as Level	No
Modeling	
Regression Type	Linear Regression
Link Function	Logit
Optimization Options	
Convergence Options	
Model Selection	
Selection Method	None
Selection Criterion	DEFAULT
Stop Criterion	DEFAULT
Selection Options	
Score	
Excluded Variables	Reject
Status	
Create Time	9/17/20 6:23 AM
Run ID	
Last Error	
Last Status	
Last Run Time	
Run Duration	
Grid Host	
User-Added Node	No

Figure 20: High Performance Regression Model

A high-performance regression model is applied as shown in Figure 20 with regression type is set to linear regression and polynomial degree set to 2.

Analysis of Variance					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	37	7.64858E14	2.067184E13	276.22	<.0001
Error	29206	2.185699E15	74837335831		
Corrected Total	29243	2.950557E15			
Root MSE	273564				
R-Square	0.25922				
Adj R-Sq	0.25829				
AIC	761512				
AICC	761512				
SBC	732581				
ASE (Train)	74740091310				
ASE (Validate)	67907825399				

Figure 21: ANOVA of HP Regression

As shown in Figure 21 is the analysis of variance of HP regression model, the Akaike Information Criterion (AIC) will be used to compare the performance with other models, low AIC indicates better model. However, the R-squared is low because the model only 26% variability of the response data around its mean.

Parameter	DF	Estimate	Standard Error	t Value	Pr > t	Variance Inflation
Intercept	1	255687	473901	0.54	0.5895	0
propertyType HOUSE	1	-20278	6410.317255	-3.16	0.0016	2.20302
propertyType UNIT	0	0
bathrooms 1	1	116078	273597	0.42	0.6714	7183.20958
bathrooms 2	1	146334	273583	0.53	0.5927	7311.98003
bathrooms 3	1	268061	273655	0.98	0.3273	1671.88714
bathrooms 4	1	549543	274422	2.00	0.0452	166.09481
bathrooms 5	1	584844	279197	2.09	0.0362	27.05765
bathrooms 6	1	146114	300271	0.49	0.6265	6.02287
bathrooms 8	1	1557736	393961	3.95	<.0001	2.07383
bathrooms 9	1	27659	353186	0.08	0.9376	3.33342
bathrooms 21	0	0
bedrooms 0	1	50628	279016	0.18	0.8560	25.98422
bedrooms 1	1	-79005	273758	-0.29	0.7729	1566.21159
bedrooms 2	1	1379.936500	273645	0.01	0.9960	3656.47314
bedrooms 3	1	98013	273586	0.36	0.7202	7144.20926
bedrooms 4	1	254334	273583	0.93	0.3526	6197.74629
bedrooms 5	1	429609	273657	1.57	0.1165	1582.79671
bedrooms 6	1	500562	274180	1.83	0.0679	233.17405
bedrooms 7	1	405971	277370	1.46	0.1433	39.01390
bedrooms 8	1	836928	317216	2.64	0.0083	4.03337
bedrooms 9	1	164329	316480	0.52	0.6036	6.69069
bedrooms 11	1	1025227	390931	2.62	0.0087	2.04205
bedrooms 12	0	0
bedrooms 14	0	0
parking 0	1	59296	273679	0.22	0.8285	1228.65395
parking 1	1	46628	273590	0.17	0.8647	6384.37359
parking 2	1	90257	273588	0.33	0.7415	7267.29349
parking 3	1	154520	273677	0.56	0.5723	1373.82687
parking 4	1	125136	273712	0.46	0.6475	983.62836
parking 5	1	148715	274665	0.54	0.5882	126.46869
parking 6	1	146680	274666	0.53	0.5933	126.47022
parking 7	1	133226	280695	0.47	0.6351	19.99046
parking 8	1	150884	280139	0.54	0.5902	23.05291
parking 9	1	53084	335068	0.16	0.8741	3.00018
parking 10	1	75871	292482	0.26	0.7953	7.99973
parking 11	1	42695	292485	0.15	0.8839	7.99987
parking 12	1	-97804	386958	-0.25	0.8005	2.00075
parking 16	1	161923	386909	0.42	0.6756	2.00025
parking 18	1	81266	387383	0.21	0.8338	4.01018
parking 21	1	126600	386934	0.33	0.7435	2.00051
parking 27	1	-24667	386927	-0.06	0.9492	2.00044
parking 31	0	0

Figure 22: HP Regression Parameter Estimates

From the parameter estimates list as shown in Figure 22, it indicates which variable has high significance towards the target in regression. When the property type is house, it will affect significantly on house price. In addition, with 4, 5 and 8 number of bathrooms will significantly affect house price. Plus, 8 and 11 number of bedrooms will affect house price significantly. However, number of parking lot will not affect the house price in general.

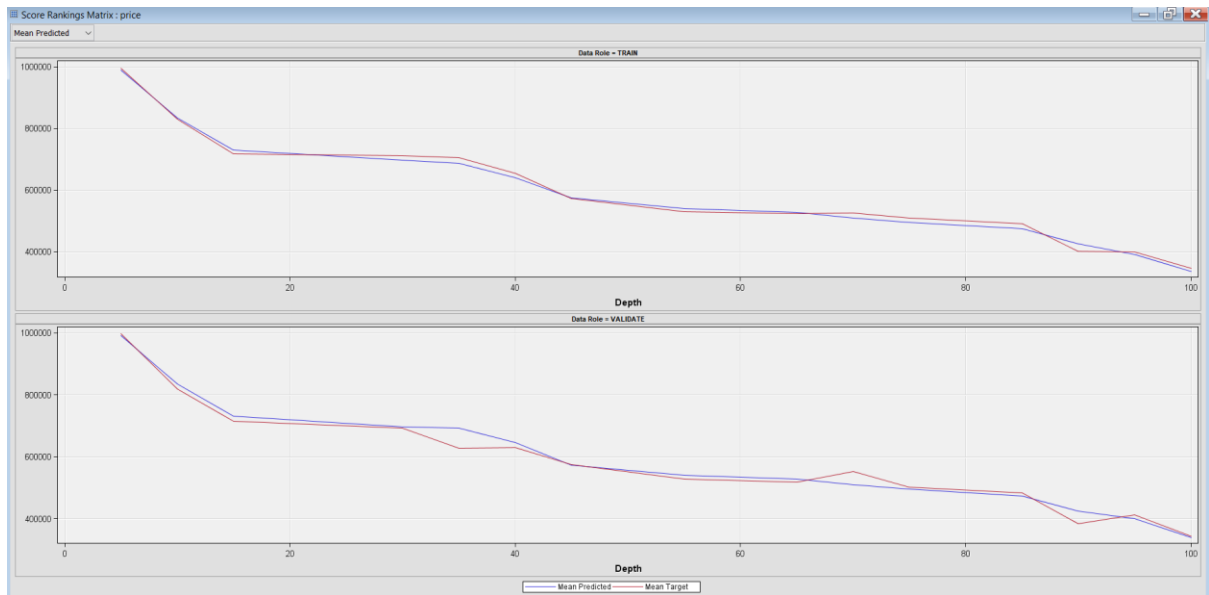


Figure 23: HP Regression Score Rankings Matrix

The score rankings matrix of predicted and target mean are good for both training data and validation data as shown in Figure 23, model learning is high during the depth of 20.

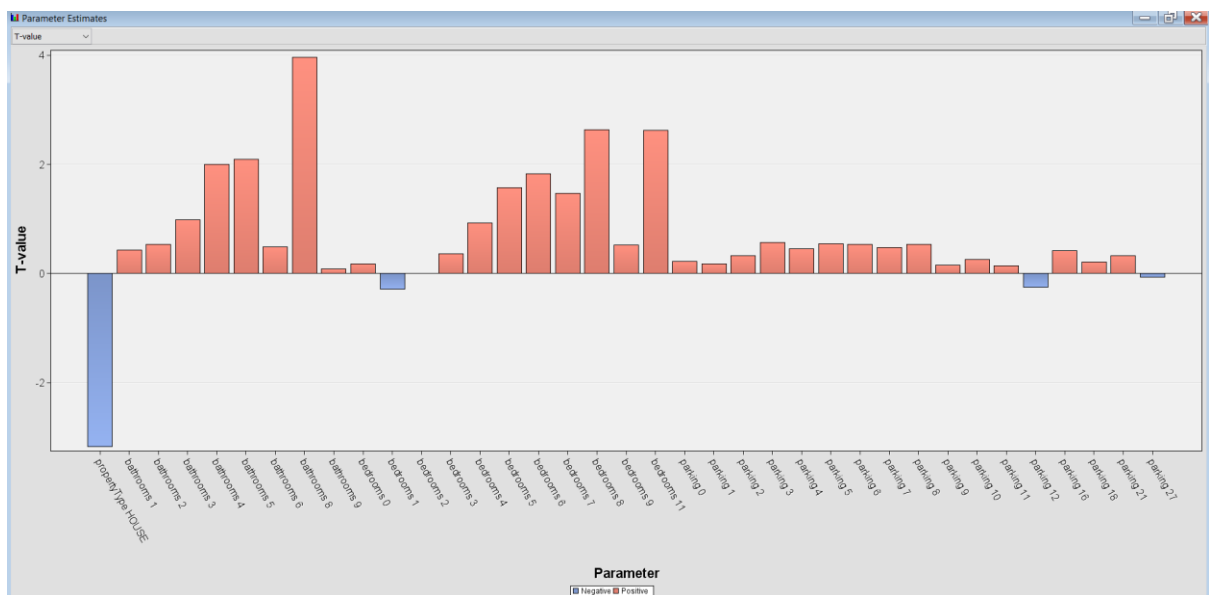


Figure 24: HP Regression T-Value

The T-Value graph as shown in Figure 24 indicates the significant variables, when the variable has T-value of more than +2 or less than -2 it will significantly affect the house price.

Effect	Parameter	Variance Inflation
propertyType	propertyType HOUSE	2.20302
bathrooms	bathrooms 1	7183.20958
bathrooms	bathrooms 2	7311.98003
bathrooms	bathrooms 3	1671.88714
bathrooms	bathrooms 4	166.09481
bathrooms	bathrooms 5	27.05765
bathrooms	bathrooms 6	6.02287
bathrooms	bathrooms 8	2.07383
bathrooms	bathrooms 9	3.33342
bedrooms	bedrooms 0	25.98422
bedrooms	bedrooms 1	1566.21159
bedrooms	bedrooms 2	3656.47314
bedrooms	bedrooms 3	7144.20926
bedrooms	bedrooms 4	6197.74629
bedrooms	bedrooms 5	1582.79671
bedrooms	bedrooms 6	233.17405
bedrooms	bedrooms 7	39.01390
bedrooms	bedrooms 8	4.03337
bedrooms	bedrooms 9	6.69069
bedrooms	bedrooms 11	2.04205
parking	parking 0	1228.65395
parking	parking 1	6384.37359
parking	parking 2	7267.29349
parking	parking 3	1373.82687
parking	parking 4	983.62836
parking	parking 5	126.46869
parking	parking 6	126.47022
parking	parking 7	19.99046
parking	parking 8	23.05291
parking	parking 9	3.00018
parking	parking 10	7.99973
parking	parking 11	7.99987
parking	parking 12	2.00075
parking	parking 16	2.00025
parking	parking 18	4.01018
parking	parking 21	2.00051
parking	parking 27	2.00044

Figure 25: HP Regression VIF

As shown in Figure 25 indicates the multicollinearity of independent variables. House with 1 and 2 bathrooms have high collinearity, house with 3 and 4 bedrooms have high collinearity and house with 1 or 2 parking lots have high collinearity.

2.2 Dmine Regression

Property	Value
General	
Node ID	DmineReq2
Imported Data	...
Exported Data	...
Notes	...
Train	
Variables	...
Maximum Variable Number	3000
R-Square Options	
Minimum R-Square	0.005
Stop R-Square	5.0E-4
Created Variables	
Use AOV16 Variables	Yes
Use Group Variables	Yes
Use Interactions	No
Print Option	Default
Use SPD Engine Library	Yes
Status	
Create Time	9/17/20 9:13 AM
Run ID	
Last Error	
Last Status	
Last Run Time	
Run Duration	
Grid Host	
User-Added Node	No




Figure 26: Dmine Regression

Next, Dmine regression model is added for comparing with HP regression model, default settings are used.

The DMINE Procedure

R-Squares for Target Variable: price		
Effect	DF	R-Square
Class: bathrooms*bedrooms	53	0.259495
Group: bathrooms*bedrooms	29	0.259493
Class: bedrooms*parking	80	0.245151
Group: bedrooms*parking	37	0.245147
Class: bedrooms*propertyType	17	0.231782
Group: bedrooms*propertyType	12	0.231781
Class: bedrooms	11	0.229132
Group: bedrooms	10	0.229131
Class: bathrooms*parking	61	0.162924
Group: bathrooms*parking	32	0.162921
Class: bathrooms*propertyType	13	0.138403
Group: bathrooms*propertyType	10	0.138403
Class: parking*propertyType	22	0.113907
Group: parking*propertyType	14	0.113906
Class: bathrooms	8	0.100152
Class: parking	17	0.089324
Group: parking	7	0.089322
Class: propertyType	1	0.052095

The DMINE Procedure

Effects Chosen for Target: price						
Effect	DF	R-Square	F Value	p-Value	Sum of Squares	Error Mean Square
Group: bedrooms	10	0.229131	868.915012	<.0001	6.7606489E14	77805640898
Class: bathrooms	8	0.023391	114.318301	<.0001	6.9016592E13	75465379249
Group: parking	7	0.006408	36.090295	<.0001	1.8906072E13	74836389793

Figure 27: Details of Dmine Regression

As shown in Figure 27, Dmine regression is not as detailed as HP regression, it only indicates the variable ‘bedrooms’, ‘bathrooms’ and ‘parking’ are significant to the target price. Also, it shows the combination of different variables to find the highest r-squared, the combination of bathrooms/bedrooms, bedrooms/parking, bedrooms/property type give the highest r-squared.

The DMINE Procedure						
The Final ANOVA Table for Target: price						
Effect	DF	R-Square	Sum of Squares			
Model	25	0.258930	7.6398756E14			
Error	29218	.	2.1865696E15			
Total	29243	.	2.9505572E15			
The DMINE Procedure						
Effects Not Chosen for Target: price						
Effect	DF	R-Square	F Value	p-Value	Sum of Squares	
Class: propertyType	1	0.000255	10.057540	0.0015	752436707191	

Figure 28: Dmine Regression ANOVA

As shown in Figure 28 is the ANOVA of the model and property type is not chosen for target price prediction.

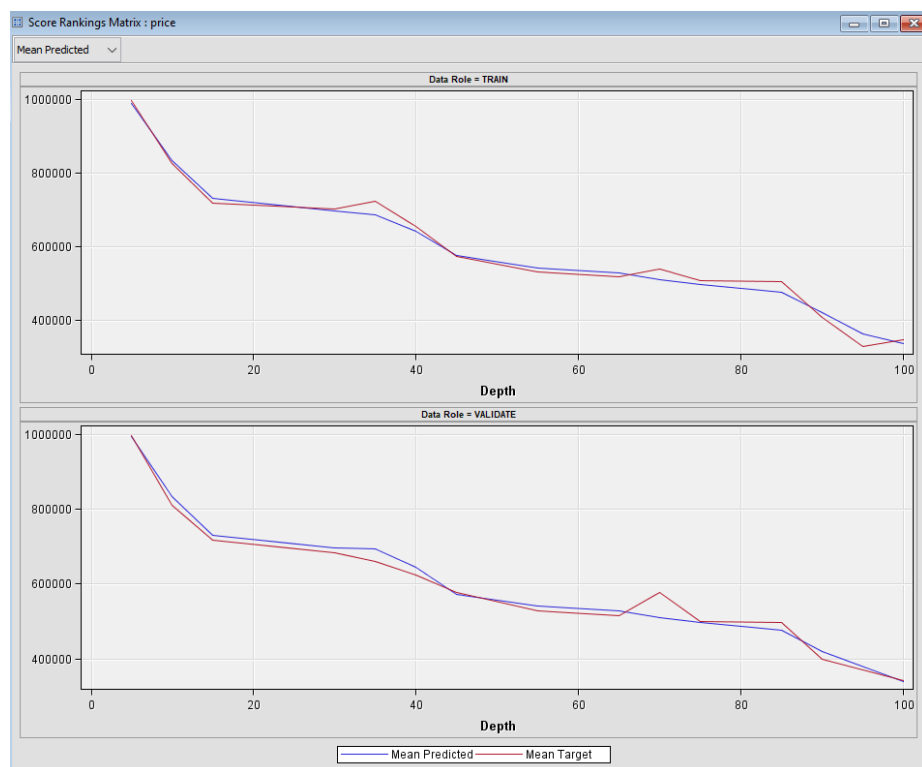


Figure 29: Dmine Regression Score Rankings Matrix

In Figure 29, Dmine regression model learning is good as the rate is high during the depth of 20.

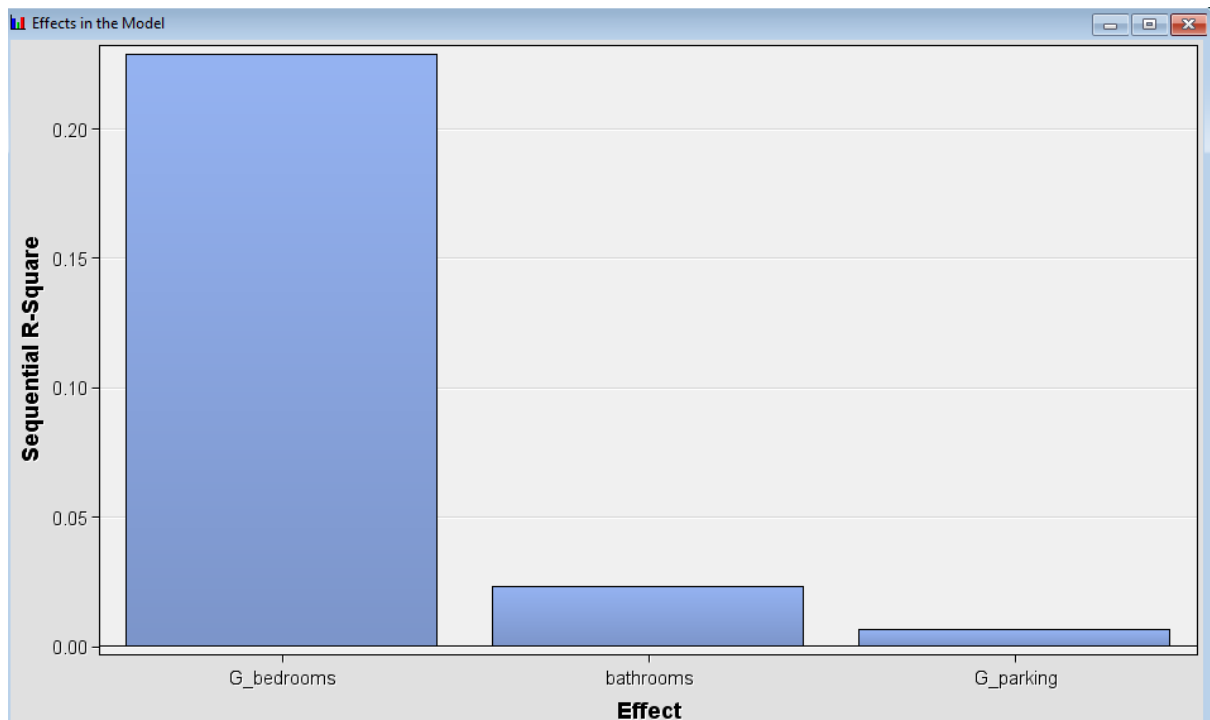


Figure 30: Dmine Regression Sequential R-square

Variables 'bedrooms', 'bathrooms' and 'parking' explain the variability of the response data around its mean, 'bedrooms' variable exhibits the greatest effect for the model.

2.3 Simple Linear Regression

Property	Value
General	
Node ID	Reg2
Imported Data	...
Exported Data	...
Notes	...
Train	
Variables	...
Equation	
Main Effects	Yes
Two-Factor Interactions	No
Polynomial Terms	No
Polynomial Degree	2
User Terms	No
Term Editor	...
Class Targets	
Regression Type	Linear Regression
Link Function	Logit
Model Options	
Suppress Intercept	No
Input Coding	Deviation
Model Selection	
Selection Model	None
Selection Criterion	Default
Use Selection Defaults	Yes
Selection Options	...
Optimization Options	
Technique	Default
Default Optimization	Yes
Max Iterations	0
Max Function Calls	0
Maximum Time	1 Hour
Convergence Criteria	
Uses Defaults	Yes
Options	...
Output Options	
Confidence Limits	No
Save Covariance	No
Covariance	No
Correlation	No
Statistics	No
Suppress Output	No
Details	No


 Regression

Figure 31: Linear Regression

As shown in Figure 31 is a simple linear regression node and its settings, regression type is set to linear regression and the polynomial degree is set to 2.

Analysis of Variance					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	37	7.6485796E14	2.0671837E13	276.22	<.0001
Error	29206	2.1856992E15	74837335831		
Corrected Total	29243	2.9505572E15			

Model Fit Statistics			
R-Square	0.2592	Adj R-Sq	0.2583
AIC	732266.2889	BIC	732268.3878
SBC	732581.0593	C(p)	38.0000

Type 3 Analysis of Effects				
Effect	DF	Sum of Squares	F Value	Pr > F
bathrooms	8	6.24195E13	104.26	<.0001
bedrooms	11	2.6307E14	319.57	<.0001
parking	17	1.87832E13	14.76	<.0001
propertyType	1	7.48855E11	10.01	0.0016

Figure 32: Linear Regression ANOVA

As shown in Figure 32, the statistics table indicates the AIC of linear regression is lower than HP regression, which indicates a simple linear regression model has better performance than HP regression model. Moreover, the analysis indicates variables such as ‘bathrooms’, ‘bedrooms’, ‘parking’ and ‘propertyType’ are significant in the prediction.

Analysis of Maximum Likelihood Estimates

Parameter	DF	Estimate	Standard Error	t Value	Pr > t
Intercept	1	1012160	71737.9	14.11	<.0001
bathrooms 1	1	-261296	52896.9	-4.94	<.0001
bathrooms 2	1	-231040	52858.2	-4.37	<.0001
bathrooms 3	1	-109314	53089.9	-2.06	0.0395
bathrooms 4	1	172169	55882.9	3.08	0.0021
bathrooms 5	1	207470	71982.6	2.88	0.0040
bathrooms 6	1	-231261	120457	-1.92	0.0549
bathrooms 8	1	1180361	255130	4.63	<.0001
bathrooms 9	1	-349715	203960	-1.71	0.0864
bedrooms 0	1	-256704	63190.7	-4.06	<.0001
bedrooms 1	1	-386336	39596.5	-9.76	<.0001
bedrooms 2	1	-305951	39009.3	-7.84	<.0001
bedrooms 3	1	-209319	38743.1	-5.40	<.0001
bedrooms 4	1	-52997.4	38762.0	-1.37	0.1716
bedrooms 5	1	122278	39084.5	3.13	0.0018
bedrooms 6	1	193231	41754.7	4.63	<.0001
bedrooms 7	1	98639.9	56541.1	1.74	0.0811
bedrooms 8	1	529597	150702	3.51	0.0004
bedrooms 9	1	-143002	149507	-0.96	0.3388
bedrooms 11	1	717895	257100	2.79	0.0052
parking 0	1	-22610.5	40627.1	-0.56	0.5778
parking 1	1	-35277.9	40091.9	-0.88	0.3789
parking 2	1	8351.1	40012.8	0.21	0.8347
parking 3	1	72614.1	40540.7	1.79	0.0733
parking 4	1	43230.0	40714.6	1.06	0.2883
parking 5	1	66809.3	46113.1	1.45	0.1474
parking 6	1	64773.9	46112.7	1.40	0.1601
parking 7	1	51320.1	71446.8	0.72	0.4726
parking 8	1	68978.2	69428.9	0.99	0.3205
parking 9	1	-28822.3	186713	-0.15	0.8773
parking 10	1	-6035.5	105406	-0.06	0.9543
parking 11	1	-39211.2	105383	-0.37	0.7098
parking 12	1	-179710	261076	-0.69	0.4912
parking 16	1	80017.1	261009	0.31	0.7592
parking 18	1	-640.4	261623	-0.00	0.9980
parking 21	1	44693.4	261046	0.17	0.8641
parking 27	1	-106573	261060	-0.41	0.6831
propertyType house	1	-10138.9	3205.2	-3.16	0.0016

Figure 33: Analysis of Maximum Likelihood Estimates

In Figure 33, the significant variables are indicated, the variables with p-value less than 0.05 are ‘bathrooms 1, 2, 3, 4, 5 and 8’, ‘bedrooms 0, 1, 2, 3, 5, 6, 8, 11’ and ‘propertyType is house’.

Fit Statistics

Target=price Target Label=price

Fit Statistics	Statistics Label	Train	Validation
AIC	Akaike's Information Criterion	732266.29	.
ASE	Average Squared Error	74740091310.45	67907285210.47
AVERR	Average Error Function	74740091310.45	67907285210.47
DFE	Degrees of Freedom for Error	29206.00	.
DFM	Model Degrees of Freedom	38.00	.
DFT	Total Degrees of Freedom	29244.00	.
DIV	Divisor for ASE	29244.00	12533.00
ERR	Error Function	2.18569923E15	8.510820055E14
FPE	Final Prediction Error	74934580351.73	.
MAX	Maximum Absolute Error	7273666.02	4698666.02
MSE	Mean Square Error	74837335831.09	67907285210.47
NOBS	Sum of Frequencies	29244.00	12533.00
NW	Number of Estimate Weights	38.00	.
RASE	Root Average Sum of Squares	273386.34	260590.26
RFPE	Root Final Prediction Error	273741.81	.
RMSE	Root Mean Squared Error	273564.13	260590.26
SBC	Schwarz's Bayesian Criterion	732581.06	.
SSE	Sum of Squared Errors	2.18569923E15	8.510820055E14
SUMW	Sum of Case Weights Times Freq	29244.00	12533.00

Figure 34: Linear Regression Fit Statistics

As shown in Figure 34 is the fit statistics of simple liner regression model, the model performs better with lower error and AIC.

Assessment Score Rankings			
Data Role=TRAIN Target Variable=price Target Label=price			
Depth	Number of Observations	Mean Target	Mean Predicted
5	1810	994656.83	989082.60
10	1428	830160.95	834574.33
15	5244	717688.36	730232.95
30	1377	711345.67	696552.13
35	908	705249.15	686856.27
40	970	654128.72	641195.28
45	4083	573263.46	574784.29
55	2863	529466.62	540740.21
65	1592	524474.89	528012.01
70	475	527100.61	509595.68
75	2648	509476.70	496186.46
85	1776	490731.54	474664.70
90	1555	401108.03	425655.13
95	1133	400493.49	392373.36
100	1382	347278.77	337564.34
Data Role=VALIDATE Target Variable=price Target Label=price			
Depth	Number of Observations	Mean Target	Mean Predicted
5	744	997559.90	991783.84
10	651	818993.37	834883.42
15	2216	714886.40	730555.09
30	623	693642.11	696591.08
35	161	628500.00	693296.50
40	720	630945.60	646537.64
45	1665	575127.31	572559.20
55	1213	529256.12	540479.31
65	702	519199.76	527744.84
70	192	553239.29	510220.69
75	1152	501223.82	496216.90
85	761	483868.74	473749.58
90	653	384878.60	425262.87
95	463	413354.52	400263.88
100	617	343259.65	341042.93

Figure 35: Assessment Score Rankings

The model learns well at the beginning and then gradually decreases as the depth increases as shown in Figure 35.

Property	Value
General	
Node ID	MdlComp6
Imported Data	...
Exported Data	...
Notes	...
Train	
Variables	...
<input type="checkbox"/> Assessment Reports	
Number of Bins	20
ROC Chart	Yes
Recompute	No
<input type="checkbox"/> Model Selection	
Selection Data	Default
Selection Statistic	Default
HP Selection Statistic	Default
SAS Viya Selection Statistic	...
Selection Table	Train
Selection Depth	10
Score	
Selection Editor	...
Report	
<input type="checkbox"/> Selected Model	
Target	price
Model Node	Reg
Model Description	Regression
Selection Criteria	Valid: Average Squared Error
Status	
Create Time	9/19/20 9:04 AM
Run ID	
Last Error	
Last Status	
Last Run Time	
Run Duration	
Grid Host	
User-Added Node	No

Model Comparison 1

Figure 36: Model Comparison 1

Model comparison node as shown in Figure 36 is used with default setting to observe the performance of HP regression, Dmine regression and linear regression.

Fit Statistics

Model Selection based on Valid: Average Squared Error (_VASE_)

Selected Model	Model Node	Model Description	Valid: Average Squared Error	Train: Average Squared Error
Y	Reg	Regression	67907285210	74740091310
	HPReg	HP Regression	67907825399	74740091310
	DmineReg	Dmine Regression	68033660040	74769854909

Figure 37: Model Comparison Fit Statistics

The fit statistics as shown in Figure 37 compares performance between linear regression, HP regression and Dmine regression, as a result linear regression performs the best with the lowest error overall.

Fit Statistics Table			
Target: price			
Data Role=Train			
Statistics	Reg	HPReg	DmineReg
Train: Akaike's Information Criterion	732266.29	.	.
Train: Average Squared Error	74740091310.45	74740091310.45	74769854909.45
Train: Average Error Function	74740091310.45	.	.
Selection Criterion: Valid: Average Squared Error	67907285210.47	67907825399.25	68033660040.14
Train: Degrees of Freedom for Error	29206.00	.	.
Train: Model Degrees of Freedom	38.00	.	.
Train: Total Degrees of Freedom	29244.00	.	.
Train: Divisor for ASE	29244.00	29244.00	29244.00
Train: Error Function	2.18569923E15	.	.
Train: Final Prediction Error	74934580351.73	.	.
Train: Maximum Absolute Error	7273666.02	7273666.02	7273115.79
Train: Mean Square Error	74837335831.09	.	.
Train: Sum of Frequencies	29244.00	29244.00	29244.00
Train: Number of Estimate Weights	38.00	.	.
Train: Root Average Squared Error	273386.34	273386.34	273440.77
Train: Root Final Prediction Error	273741.81	.	.
Train: Root Mean Squared Error	273564.13	.	.
Train: Schwarz's Bayesian Criterion	732581.06	.	.
Train: Sum of Squared Errors	2.18569923E15	2.18569923E15	2.186569637E15
Train: Sum of Case Weights Times Freq	29244.00	.	.
Data Role=Valid			
Statistics	Reg	HPReg	DmineReg
Valid: Average Squared Error	67907285210.47	67907825399.25	68033660040.14
Valid: Average Error Function	67907285210.47	.	.
Valid: Divisor for ASE	12533.00	12533.00	12533.00
Valid: Error Function	8.510820055E14	.	.
Valid: Maximum Absolute Error	4698666.02	4698666.02	4698115.79
Valid: Mean Square Error	67907285210.47	.	.
Valid: Sum of Frequencies	12533.00	12533.00	12533.00
Valid: Root Average Squared Error	260590.26	260591.30	260832.63
Valid: Root Mean Square Error	260590.26	.	.
Valid: Sum of Squared Errors	8.510820055E14	8.510887757E14	8.526658613E14
Valid: Sum of Case Weights Times Freq	12533.00	.	.

Figure 38: Model Comparison Fit Statistics Table

Although, the AIC of HP regression is not displayed in Figure 38, linear regression has lower AIC than HP regression. Next, since the linear regression is the best model, the linear regression model will be set to stepwise, forward and backward for comparison.

Property	Value
General	
Node ID	Reg5
Imported Data	...
Exported Data	...
Notes	...
Train	
Variables	...
Equation	
Main Effects	Yes
Two-Factor Interactions	No
Polynomial Terms	No
Polynomial Degree	2
User Terms	No
Term Editor	...
Class Targets	
Regression Type	Linear Regression
Link Function	Logit
Model Options	
Suppress Intercept	No
Input Coding	Deviation
Model Selection	
Selection Model	Stepwise
Selection Criterion	Backward
Use Selection Defaults	Forward
Selection Options	Stepwise
Optimization Options	
Technique	None
Default Optimization	Yes
Max Iterations	0
Max Function Calls	0
Maximum Time	1 Hour
Convergence Criteria	
Uses Defaults	Yes
Options	...
Output Options	
Confidence Limits	No
Save Covariance	No
Covariance	No
Correlation	No
Statistics	No
Suppress Output	No
Details	No

Stepwise Regression

Figure 39: Stepwise, Forward, Backward Linear Regression

As shown in Figure 39, stepwise, forward, and backward linear regression models are created to compare with simple linear regression model.

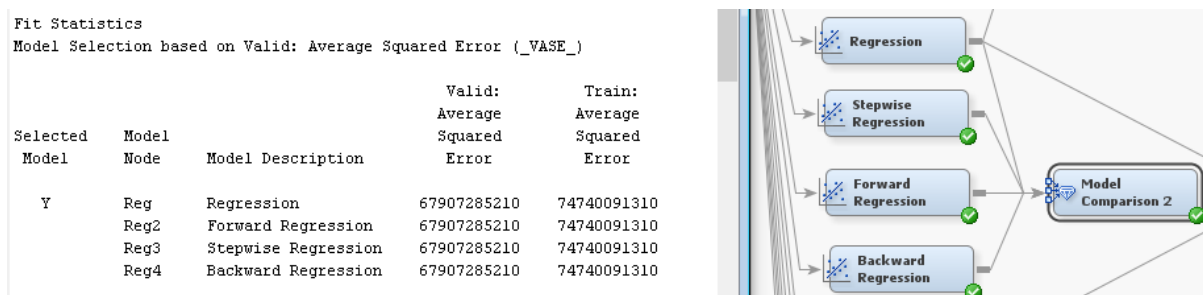


Figure 40: Model Comparison 2

As shown in Figure 40, the stepwise, forward and backward models do not provide any improvement for linear regression, thus results are identical in model comparison 2.

2.4 HP Neural Network

Property	Value
General	
Node ID	HPNNA8
Imported Data	...
Exported Data	...
Notes	...
Train	
Variables	...
Use Inverse Priors	No
Create Validation	No
Network Options	
Input Standardization	Range
Architecture	One Layer
Number of Hidden Neurons	3
Number of Hidden Layers	3
Hidden Layer Options	...
Direct Connections	No
Target Standardization	Range
Target Activation Function	Identity
Target Error Function	Normal
Number of Tries	2
Maximum Iterations	300
Use Missing as Level	No
Report	
Maximum Number of Links	1000
Status	
Create Time	9/19/20 9:59 AM
Run ID	
Last Error	
Last Status	
Last Run Time	
Run Duration	
Grid Host	
User-Added Node	No




Figure 41: HP Neural Network

A simple linear regression has the best performance among all the linear regression models, then a HP neural network model is added to compare with simple linear regression. As shown in Figure 41 is the setting of the HP neural network, the hidden neuron is set to 3, one layer of neurons and other settings are remained as default.

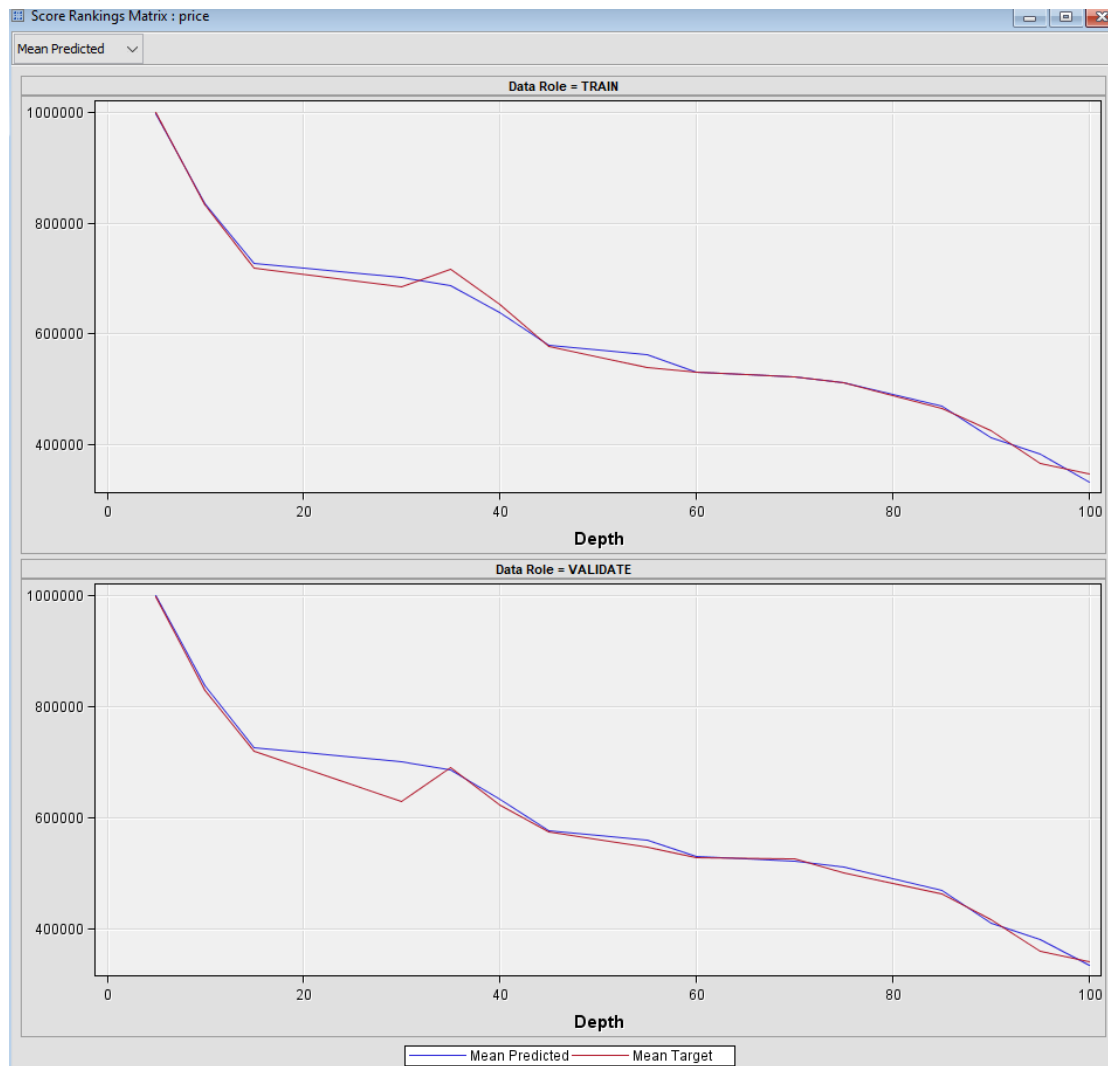


Figure 42: HP Neural Network Score Ranking Matrix

As shown in Figure 42, the HP neural network model learns well within the depth of 20, it exhibits similar properties with linear regression model.

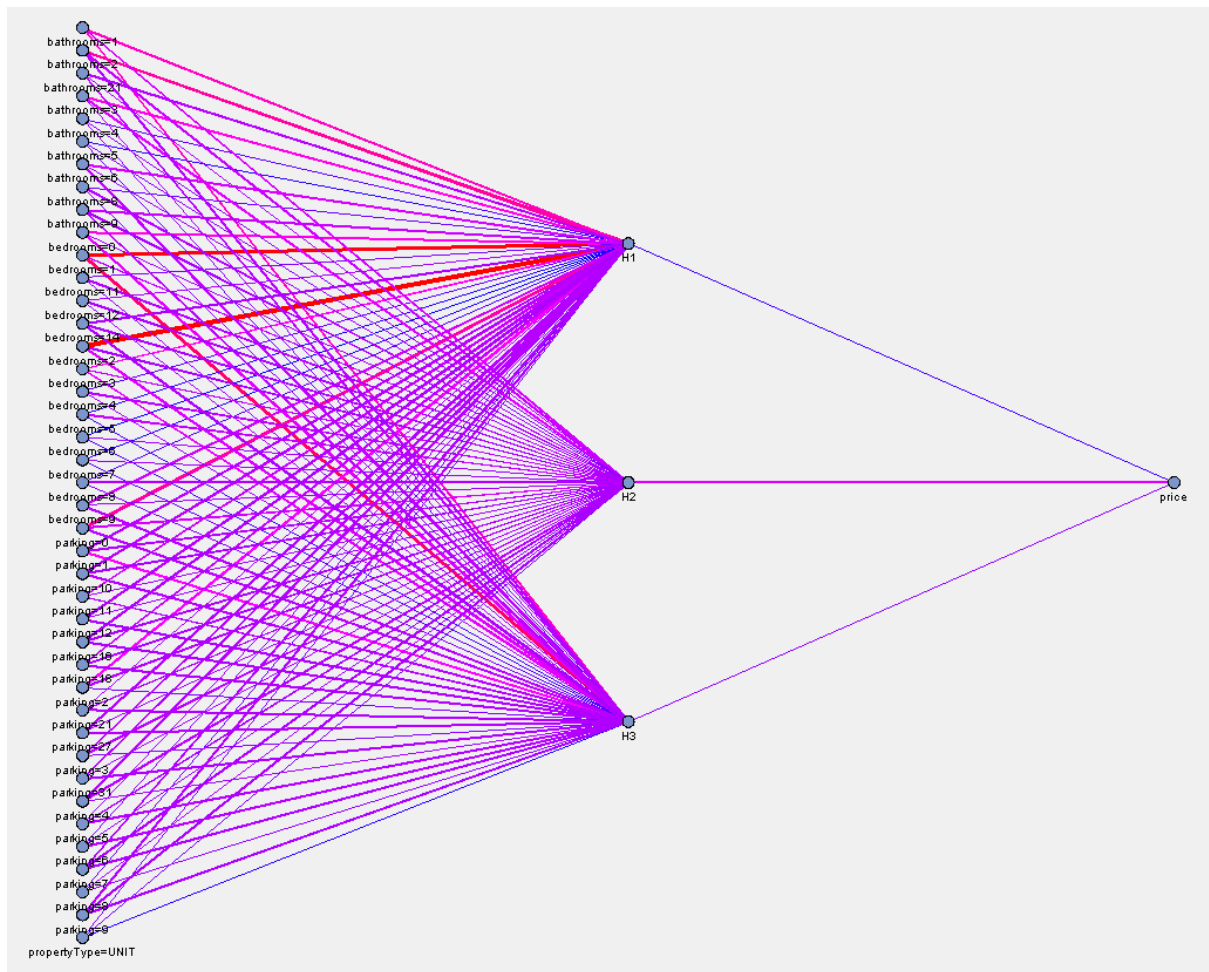


Figure 43: HP Neural Network Structure

As shown in Figure 43, the neural network consists of one hidden layer with three hidden nodes, the blue to red colour lines indicate the weight of the variable towards the hidden node, the strong weight will show red colour line. The structure also indicates hidden node 2 (H2) has the strongest weight towards the target price.

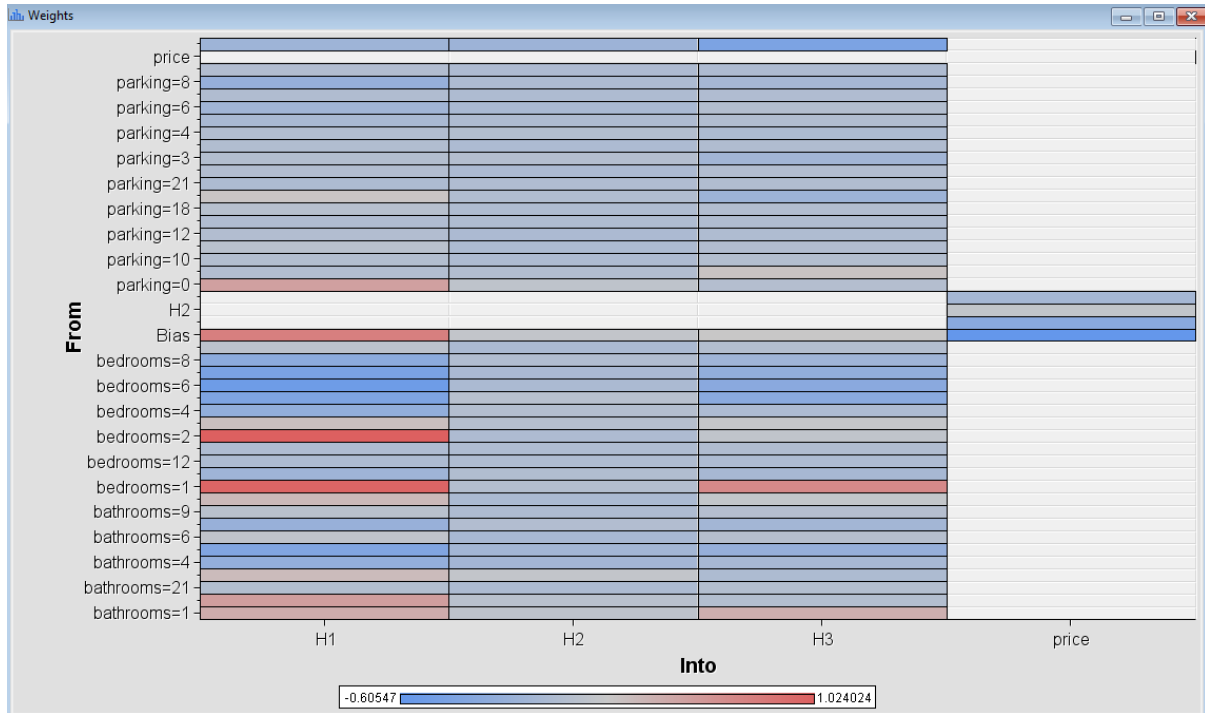


Figure 44: HP Neural Network Heat Map

The weight of each variable towards a hidden node can be clearly observed in the heat map as shown in Figure 44. The strong variables weight towards hidden node one (H1) are ‘0 parking’, ‘2 bedrooms’, ‘1 bedrooms’, and ‘1 & 2 bathrooms’, H2 has no significant weight and H3 are ‘1 bedrooms’ and ‘1 bathrooms’. Plus, it also shows H1 has high bias.

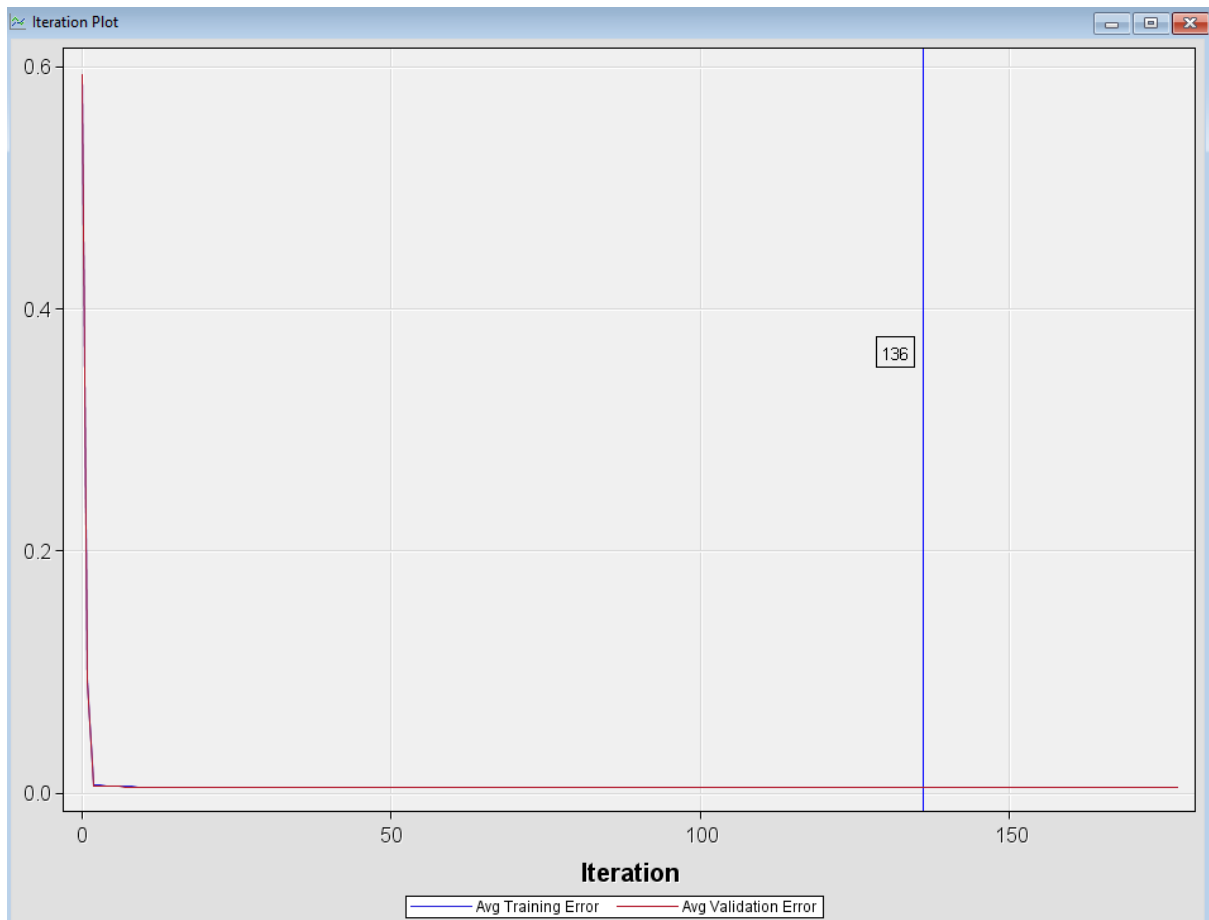


Figure 45: HP Neural Network Iteration Plot

The iteration plot as shown in Figure 45 shows optimal validation average squared error occurring on iteration 136.

Fit Statistics

Target=price Target Label=price

Fit Statistics	Statistics Label	Train	Validation
ASE	Average Squared Error	74147581342.17	67530084380.87
DIV	Divisor for ASE	29244.00	12533.00
MAX	Maximum Absolute Error	7279494.85	4704494.85
NOBS	Sum of Frequencies	29244.00	12533.00
RASE	Root Average Squared Error	272300.53	259865.51
SSE	Sum of Squared Errors	2.168371869E15	8.463545475E14

Figure 46: HP Neural Network Fit Statistics

The HP neural network obtains better average squared error compare to linear regression as shown in Figure 46.

Fit Statistics
Model Selection based on Valid: Average Squared Error (_VASE_)

Selected Model	Model Node	Model Description	Valid: Average Squared Error	Train: Average Squared Error
Y	HPNNA Reg	HP Neural 3N Regression	67530084381 67907285210	74147581342 74740091310

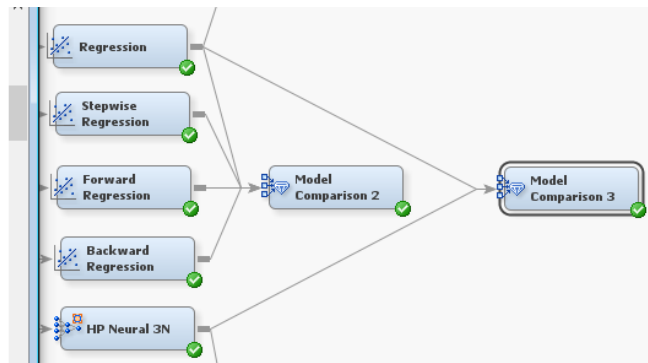


Figure 47: Model Comparison 3

As shown in Figure 47, simple linear regression obtains the best result among the linear regression models, therefore it is compared with HP neural network model. As a result, HP neural network performance with minimum settings surpasses the linear regression model with lower average squared error in training and validation.

Property	Value
General	
Node ID	HPNNA8
Imported Data	...
Exported Data	...
Notes	...
Train	
Variables	...
Use Inverse Priors	No
Create Validation	No
Network Options	
Input Standardization	Range
Architecture	One Layer
Number of Hidden Neurons	3
Number of Hidden Layers	3
Hidden Layer Options	...
Direct Connections	No
Target Standardization	Range
Target Activation Function	Identity
Target Error Function	Normal
Number of Tries	2
Maximum Iterations	300
Use Missing as Level	No
Report	
Maximum Number of Links	1000
Status	
Create Time	9/20/20 10:04 AM
Run ID	
Last Error	
Last Status	
Last Run Time	
Run Duration	
Grid Host	
User-Added Node	No

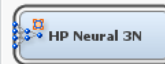


Figure 48: HP Neural Network Neuron

As shown in Figure 48, the number of hidden neurons is increased to find the best performance HP Neural Network by using model comparison node.

Fit Statistics
Model Selection based on Valid: Average Squared Error (_VASE_)

Selected Model	Model Node	Model Description	Valid: Average Squared Error	Train: Average Squared Error
Y	HPNNA5	HP Neural 6N	67265318025	74046983019
	HPNNA6	HP Neural 7N	67370069210	73651900950
	HPNNA4	HP Neural 5N	67435602902	74284905949
	HPNNA3	HP Neural 4N	67480349117	73840962382
	HPNNA	HP Neural 3N	67530084381	74147581342

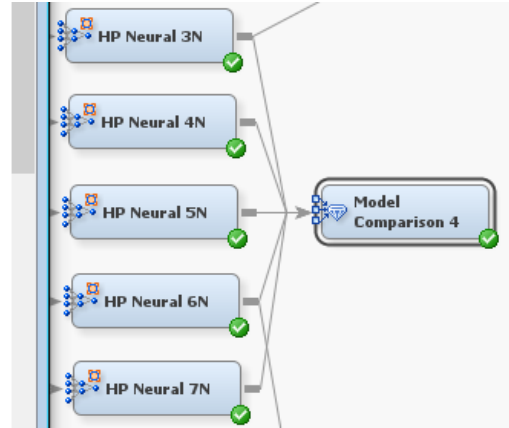


Figure 49: Model Comparison 4

The number of hidden neurons is increased from three up to seven neurons, noted the number of hidden neurons are denoted as 6N when the number of hidden neurons is six as shown in Figure 49. Increasing the number of hidden neurons improves the overall performance of HP neural network, however at 7N the performance starts to drop and 6N remains as the best performing HP neural network.

Property	Value
General	
Node ID	HPNNA8
Imported Data	...
Exported Data	...
Notes	...
Train	
Variables	...
Use Inverse Priors	No
Create Validation	No
Network Options	
Input Standardization	Range
Architecture	One Layer with Direct
Number of Hidden Neurons	User-Defined
Number of Hidden Layers	Logistic
Hidden Layer Options	One Layer
Direct Connections	One Layer with Direct
Target Standardization	Two Layers
Target Activation Function	Two Layers with Direct
Target Error Function	Normal
Number of Tries	2
Maximum Iterations	300
Use Missing as Level	No
Report	
Maximum Number of Links	1000
Status	
Create Time	9/20/20 10:16 AM
Run ID	
Last Error	
Last Status	
Last Run Time	
Run Duration	
Grid Host	
User-Added Node	No

HP Neural 6N
1LD

HP Neural 6N
2L

Figure 50: HP Neural Network Architecture

Therefore, HP neural network with six hidden neurons are chosen to observe how architecture can affect the performance as shown in Figure 50, noted that 1LD stands for one layer with direct and 2L is two layers.

Fit Statistics
Model Selection based on Valid: Average Squared Error (_VASE_)

Selected Model	Model Node	Model Description	Valid: Average Squared Error	Train: Average Squared Error
Y	HPNNA5	HP Neural 6N	67265318025	74046983019
	HPNNA7	HP Neural 6N 2L	67614232523	74112073377
	HPNNA2	HP Neural 6N 1LD	67921475329	74758711941

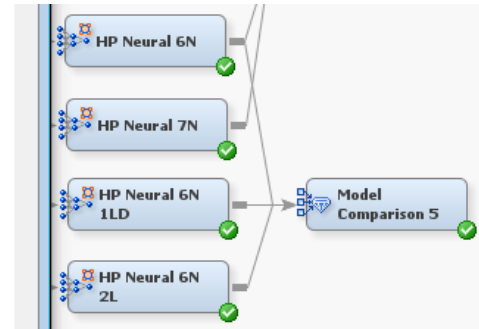


Figure 51: Model Comparison 5

As shown in Figure 51, changing the architecture to two layers or one layer with direct does not improve overall performance of HP neural network. Hence, HP neural network with one layer and six hidden neurons still performs the best among all HP neural networks.

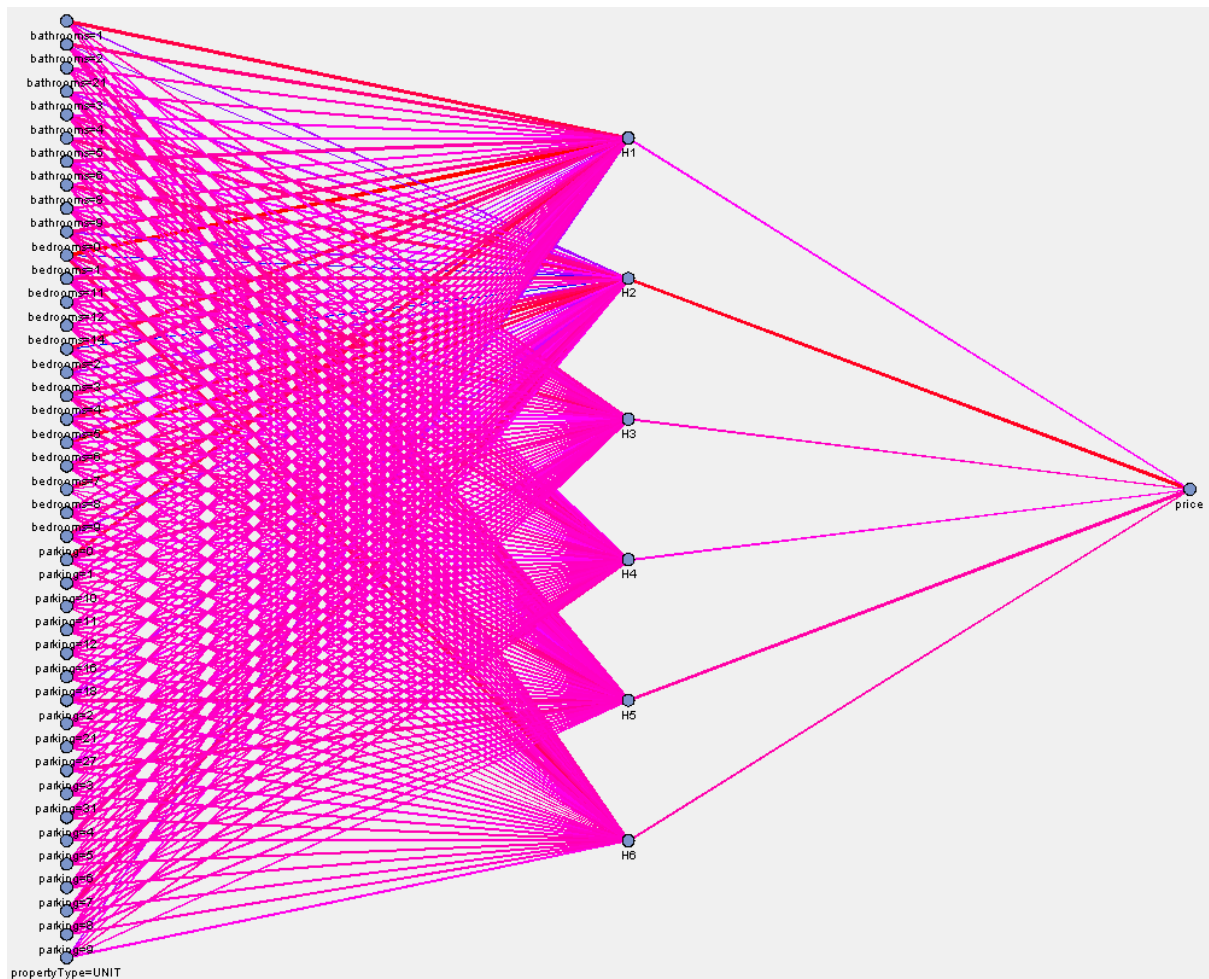


Figure 52: HP Neural 6N Architecture

Since HP neural 6N has the best performance, the analysis will be done based on this model to predict the house price. As shown in Figure 52, hidden neuron 2 (H2) has the strongest weight towards the price while H5 is the second strongest.

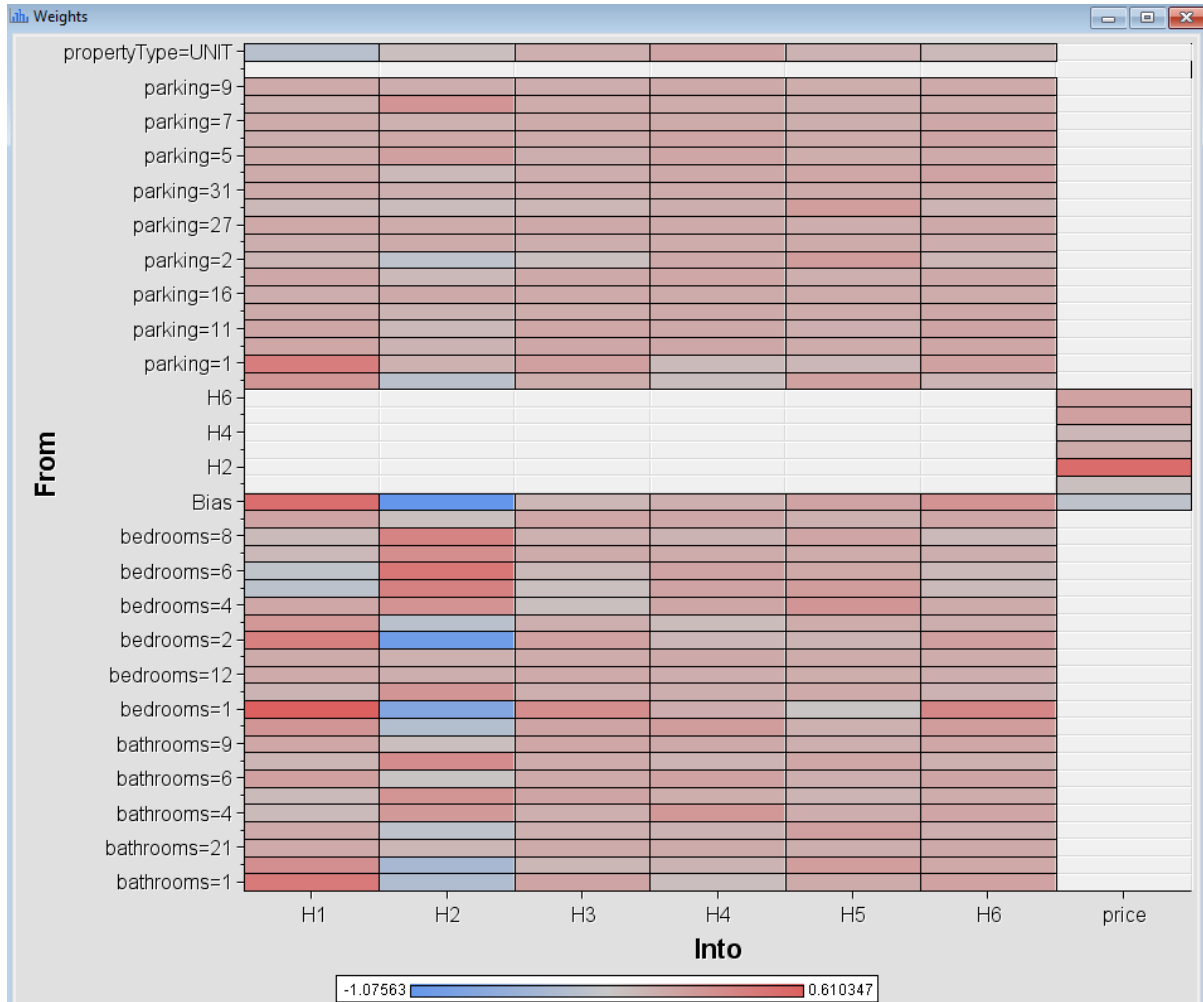


Figure 53: HP Neural 6N Heat Map

Since H2 has the strongest weight on price, the heat map of H2 is observed. As shown in Figure 53, the variables that caused strong weight on H2 are 8 parking lots, 4-8 bedrooms and 4, 5 and 8 bathrooms. Although, H5 is the second strongest weight, the weight of variables toward H5 is balanced across the variables as shown in Figure 53.



Figure 54: HP Neural 6N Iteration Plot

The iteration plot as shown in Figure 54 shows optimal validation average squared error occurring on iteration 130, which is faster than HP Neural 3N as shown in Figure 45.

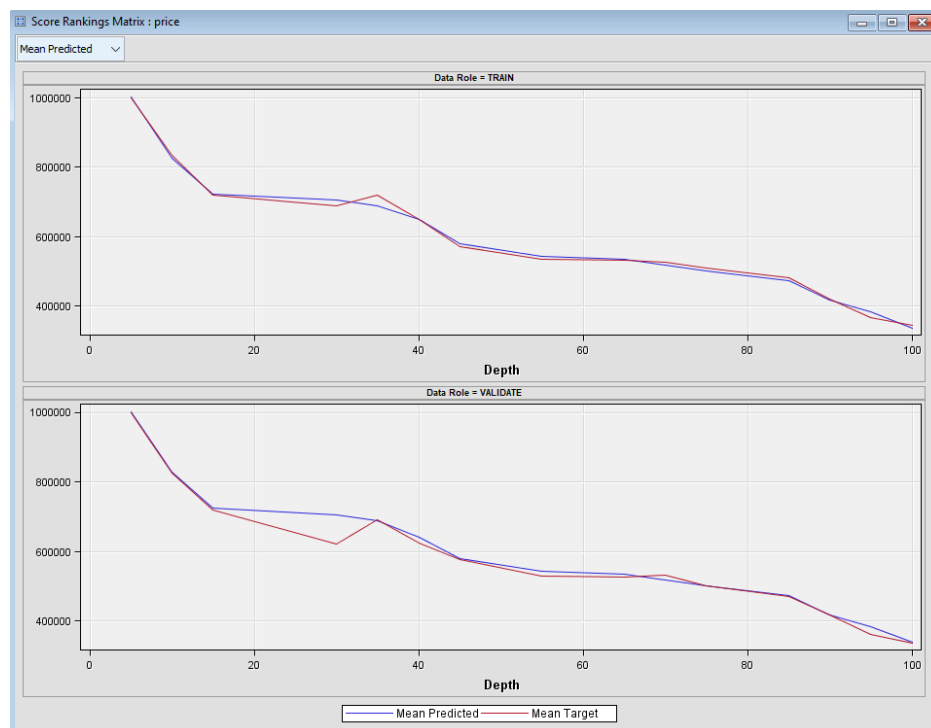


Figure 55: HP Neural 6N Score Ranking Matrix

As shown in Figure 55, HP neural 6N exhibits same characteristic as a linear regression model in score ranking matrix graph.

Fit Statistics				
Target=price Target Label=price				
Fit Statistics	Statistics Label	Train	Validation	
ASE	Average Squared Error	74046983019.29	67265318024.53	
DIV	Divisor for ASE	29244.00	12533.00	
MAX	Maximum Absolute Error	7283261.00	4708261.00	
NOBS	Sum of Frequencies	29244.00	12533.00	
RASE	Root Average Squared Error	272115.75	259355.58	
SSE	Sum of Squared Errors	2.165429971E15	8.430362308E14	

Figure 56: HP Neural 6N Fit Statistics

HP Neural 6N has the best statistical result among all the models as shown in Figure 56, the data analysis and final decision made are based on this model.

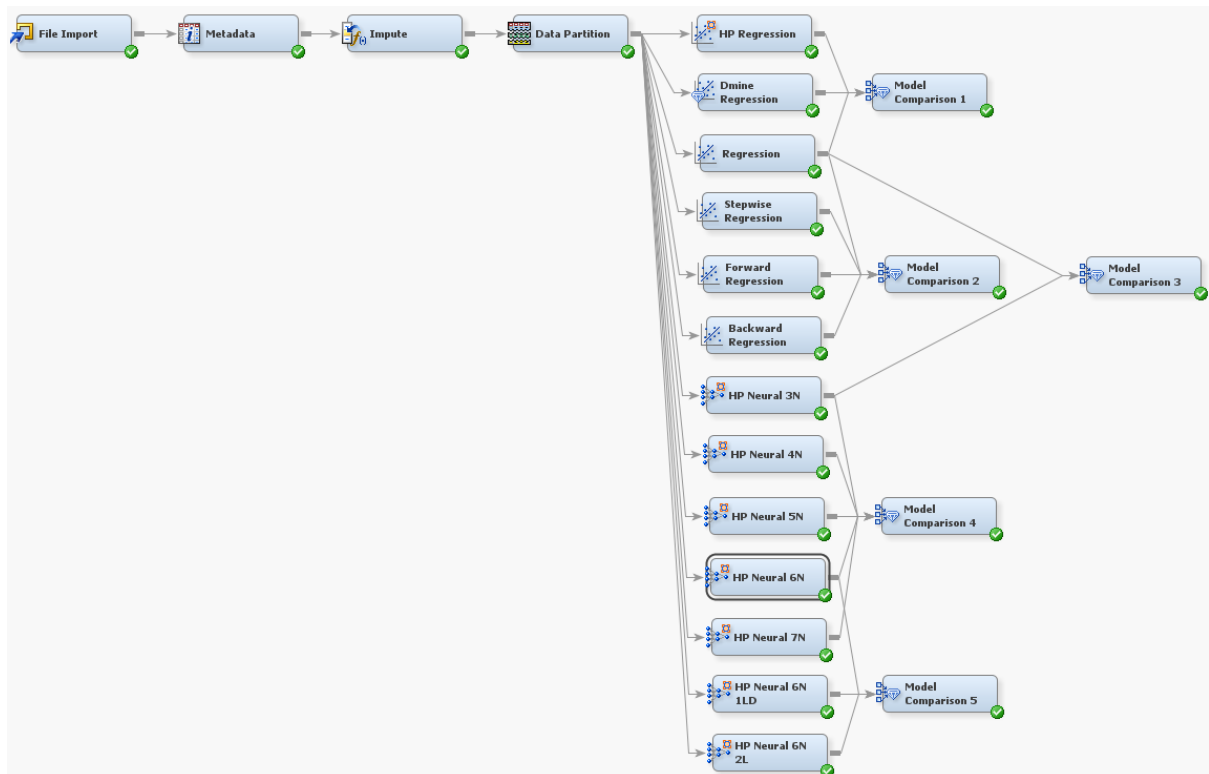


Figure 57: Overall Process Flow Diagram

As shown in Figure 57 is the overall process flow, first the file is imported and pre-processed, then the dataset is split into 70% training set and 30% validation set. After that, different types of linear regression models are compared and simple linear regression has the best result. HP neural network performs better when comparing with simple linear regression, among the HP neural networks, the HP Neural 6N has the best performance.

3 Model Selection

Table 1: Linear Regression Comparison 1

Selection	Model	ASE (Valid Train)
✗	HP Regression	67907825399 74740091310
✗	Dmine Regression	68033660040 74769854909
✓	Simple Linear Regression	67907285210 74740091310

Table 2: Linear Regression Comparison 2

Selection	Model	ASE (Valid Train)
✓	Simple Linear Regression	67907285210 74740091310
✗	Stepwise Regression	67907285210 74740091310
✗	Forward Regression	67907285210 74740091310
✗	Backward Regression	67907285210 74740091310

Table 3: Linear Regression vs. HP Neural Network

Selection	Model	ASE (Valid Train)
✗	Simple Linear Regression	67907285210 74740091310
✓	HP Neural 3N	67530084381 74147581342

Table 4: HP Neural Network Comparison 1

Selection	Model	ASE (Valid Train)
✗	HP Neural 3N	67530084381 74147581342
✗	HP Neural 4N	67480349117 73840962382
✗	HP Neural 5N	67435602902 74284905949
✓	HP Neural 6N	67265318025 74046983019
✗	HP Neural 7N	67370069210 73651900950

Table 5: HP Neural Network Comparison 2

Selection	Model	ASE (Valid Train)
✓	HP Neural 6N	67265318025 74046983019
✗	HP Neural 6N 1LD	67921475329 74758711941
✗	HP Neural 6N 2L	67614232523 74112073377

4 As shown in Model Selection

Table 1, Table 2, Table 3, Table 4 and Table 5 are the comparison of all the regression models for house price prediction in Canberra, at the end HP Neural 6N is selected.

5 Discussion & Conclusion

The dataset of house price in Canberra is pre-processed manually by deleting unnecessary variables such as ‘latitude’ and ‘longitude’ and observations that consist ‘NULL’ character. The decision of deleting the unnecessary data manually is made because the original dataset consists of 43179 observations and the pre-processed dataset consists of 41777 observations as shown in Figure 58, the deleted observations is only 3.25% of the original dataset, therefore the effect is minimal in building model for house price prediction.

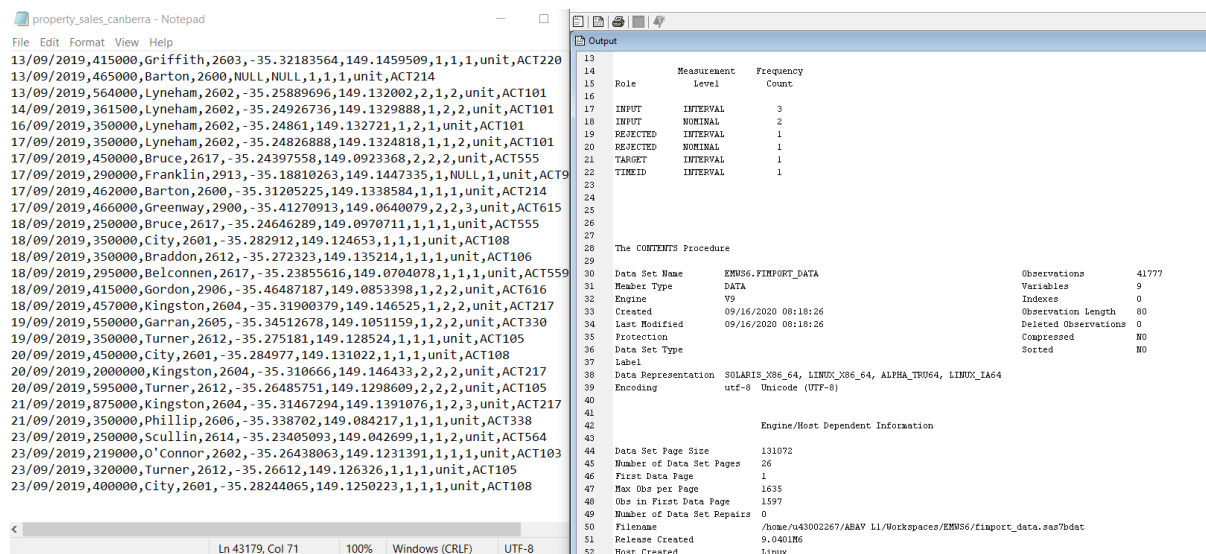


Figure 58: Dataset Observations

Furthermore, the ‘NULL’ characters in the dataset cause SAS EM to take variables that consist ‘NULL’ as a character type variable, even the target variable ‘price’ is categorised as character type variable. To ensure SAS EM to take variables like ‘bedrooms’, ‘bathrooms’ and ‘parking’ as numeric variable to build regression model and set it as interval variables, the manual pre-processed is needed. After that, the pre-processed dataset is split to 70% training set and 30% validation set.

First, HP regression, Dmine regression and simple linear regression models are built for model comparison. Among the regression models, surprisingly simple linear regression has the best overall performance with the lowest AIC and average square error, therefore simple linear regression is chosen for the second model comparison. Then, stepwise, forward and backward simple linear regression models are built to observe any improvement, however the results remain the same and no improvement is provided to the model.

Next, a HP neural network with default settings as 1 layer with 3 hidden neurons (HP neural 3N) is built to compare with simple linear regression. In the model comparison, HP neural 3N outperforms simple linear regression with lower average squared error. In addition, HP neural 3N is further enhanced by adding hidden neurons to the model. The overall performance is increased gradually until HP neural 7N (7 hidden neurons), the performance starts to drop at 7N and HP neural 6N is the best model with the lowest average square error for price prediction. Then, the architecture of HP neural 6N is changed to one layer with direct and two layers to observe any improvement, however the performance drops significantly. Therefore, HP neural 6N is the best model, data analysis and insights will be made according to the output of this model.

According to the interpretation on the output of HP neural 6N, hidden neuron H2 has the strongest weight towards the house price in Canberra, the variables that are contributing to the weight are 8 parking lots, 4-8 bedrooms and 4, 5, 8 bathrooms. In conclusion, the investor who want to purchase house in Canberra can consider the house with 8 parking lots, 4 to 8 number of bedrooms and 4, 5 and 8 number of bathrooms, the houses with these properties will have better price increment and prospect in the future, also the price is increasing gradually from year 2013 as shown in Figure 59.

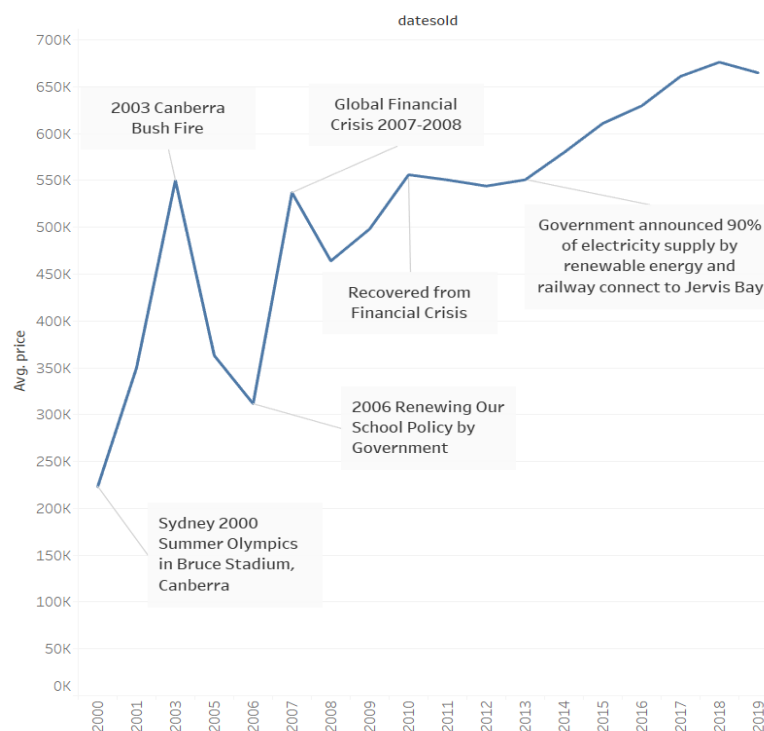


Figure 59: House Price Trend

Expensive House Location

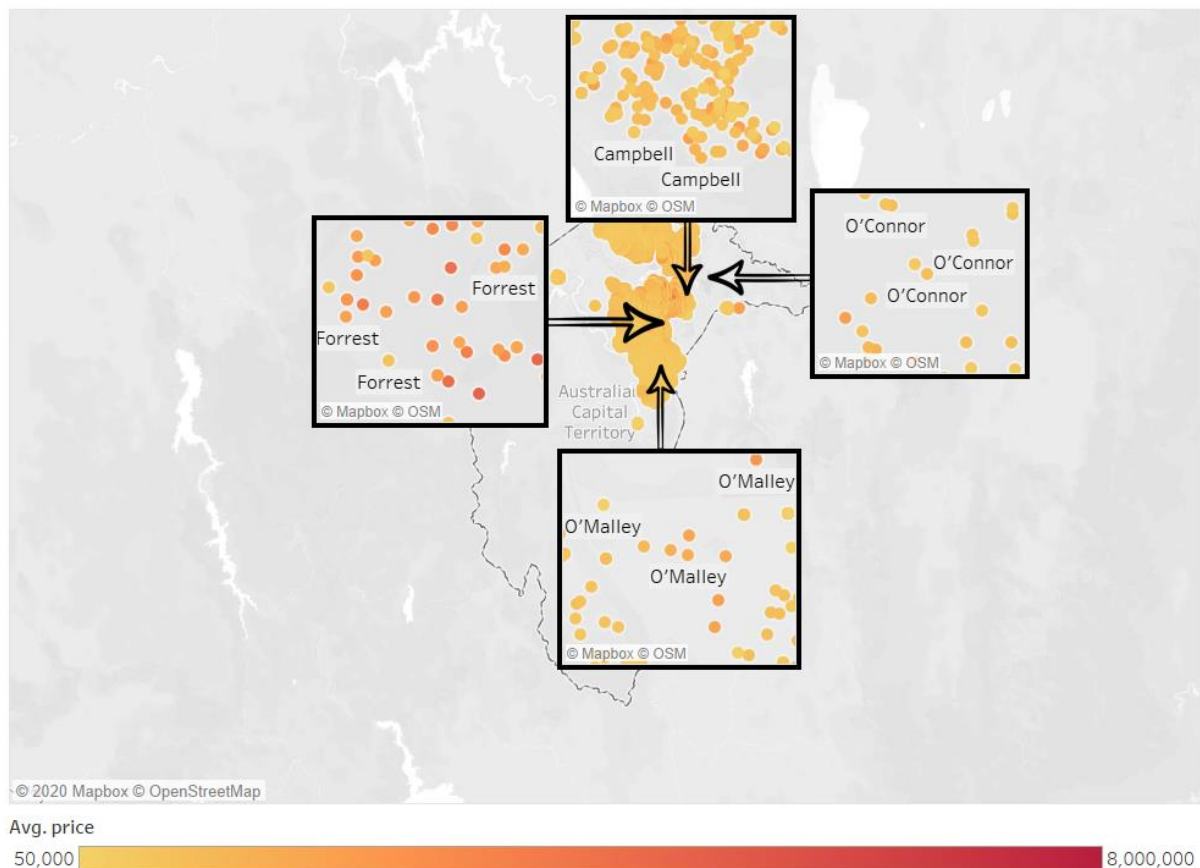


Figure 60: Location of Expensive House

Furthermore, most of the house with high price located at the suburbs such as Campbell, Forrest, O'Connor and O'Malley in Canberra as shown in Figure 60, investors can focus on these locations for their properties investment. In conclusion, the suggestions for investors are considering invest houses with 8 parking lots, 4-8 bedrooms and 4, 5, 8 bathrooms located at Campbell, Forrest, O'Connor and O'Malley suburbs.

In a nutshell, all the suggestions are provided to investors to meet the aim and objectives of this assignment, however the house price prediction is based on past dataset of house price in Canberra, there is no guarantee in return of investment. Although, the trend shows the house price is increasing gradually from year 2013 to year 2019, but extenuating circumstances such as Covid-19 cannot be predicted and there is no past dataset can predict this kind of pandemic will happen. Finally, the final suggestion to investors is **“BUY AT YOUR OWN RISK”**