Data Mining & Predictive Analytics

Midterm Exam
Fall 2020

I hereby certify that I have completed the attached examination materials, using only my own efforts. I have not asked for or received help from any person in completing this exam.

Piyush Garewal	October 07, 2020
(Name)	(date)

Q1. Using the concept of overfitting, explain why when a model is perfectly fit to training data, this is typically not good?

Answer: Overfitting a model is a condition where a statistical model begins to describe the random error in the data rather than the relationships between variables. This problem occurs when the model is too complex. Sometimes overfitting can lead to misleading R-Squared values, that it is not an accurate model.

It's not good because when looking at models you want to see the relationship between the data, and if there are zero errors in the data then the information you get is skewed and may not be a true reflection. And can lead to poor predictive performance because it can exaggerate minor fluctuations in the data.

Reference: https://statisticsbyjim.com/regression/overfitting-regression-models/

Q2. Consider the distance between records of the following dataset. Which are the closest records?

Ob. No.	**Age**	**Income ($)**	**City**
1	25	49000	Albany
2	56	156000	Albany
3	65	99000	NYC
4	32	192000	Poughkeepsie

To calculate the closest records, we will compute the Euclidean distance between each record.

Answer: We start by normalizing the data:

Ob. No.	**Age**	**Income ($)**	**City**
1	(25 -25)/ (65-25) = 0	(49000 – 49000) / (192000 - 49000) = 0	Albany
2	(56 -25)/ (65-25) = 0.775	(156000 – 49000) / (192000 - 49000) = 748.251	Albany
3	(65 – 25) / (65-25) = 1	(99000 – 49000) / (192000 - 49000) = 349.65	NYC
4	(32 – 25)/ (65 - 25) = 0.175	(192000 – 49000) / (192000 - 49000) = 100	Poughkeepsie

Then we compute the distances:

Distance between Ob. i and j	***Age (i) – Age (j)***	***Income ($)i – Income ($)j***	***City***	***Distance***
1 and 2	(0 – 0.775) = - 0.775	(0 - 748.251) = - 748.251	0	D = sqrt (0.6 + 559878.06 + 0) = 748.250
2 and 3	0.775 – 1 = -0.225	(748.251- 349.65) = 398.601	1	D = sqrt (0.05 + 158,882.757+ 1) = 398.602
3 and 4	(1 – 0.175) = 0.825	(349.65 – 100) = 249.65	1	D = sqrt (0.05 + 158,882.757+ 1) = 249.66
1 and 4	(0 –0.175) = -0.175	(0 – 100) = -100	1	D = sqrt (0.05 + 158,882.757+ 1) = 100.005

From the above Object Records 1 and 4 are the closest, as the distance between the two is the smallest of them all (100.005)

Q3. The dataset cars.xlsx contains data on used cars on sale. Explore the numeric data using data visualization capabilities available in SPSS Modeler: which of the pairs of variables seem to be correlated? Verify it numerically.

a) First, we have uploaded the dataset into SPSS modeler.
b) To have the visual of the given data, we have connected a table node, below is the attachment from table node.

	Id	Model	Price	KM	HP	Weight	ABS	Airbag	Boardc...	Pow...	Power...
1	1.0...	TOYOTA Corolla 2.0 D4D HATCHB TERRA 2/3-Doors	13500.000	46986.000	90.0...	1165....	1.0...	1.000	1.000	1.000	1.000
2	2.0...	TOYOTA Corolla 2.0 D4D HATCHB TERRA 2/3-Doors	13750.000	72937.000	90.0...	1165....	1.0...	1.000	1.000	0.000	1.000
3	3.0...	TOYOTA Corolla 2.0 D4D HATCHB TERRA 2/3-Doors	13950.000	41711.000	90.0...	1165....	1.0...	1.000	1.000	0.000	1.000
4	4.0...	TOYOTA Corolla 2.0 D4D HATCHB TERRA 2/3-Doors	14950.000	48000.000	90.0...	1165....	1.0...	1.000	1.000	0.000	1.000
5	5.0...	TOYOTA Corolla 2.0 D4D HATCHB SOL 2/3-Doors	13750.000	38500.000	90.0...	1170....	1.0...	1.000	1.000	1.000	1.000
6	6.0...	TOYOTA Corolla 2.0 D4D HATCHB SOL 2/3-Doors	12950.000	61000.000	90.0...	1170....	1.0...	1.000	1.000	1.000	1.000
7	7.0...	TOYOTA Corolla 2.0 D4D 90 3DR TERRA 2/3-Doors	16900.000	94612.000	90.0...	1245....	1.0...	1.000	1.000	1.000	1.000
8	8.0...	TOYOTA Corolla 2.0 D4D 90 3DR TERRA 2/3-Doors	18600.000	75889.000	90.0...	1245....	1.0...	1.000	1.000	1.000	1.000
9	9.0...	TOYOTA Corolla 1800 T SPORT VVT I 2/3-Doors	21500.000	19700.000	192....	1185....	1.0...	1.000	0.000	1.000	1.000
10	10....	TOYOTA Corolla 1.9 D HATCHB TERRA 2/3-Doors	12950.000	71138.000	69.0...	1105....	1.0...	1.000	1.000	0.000	1.000
11	11....	TOYOTA Corolla 1.8 VVTL-i T-Sport 3-Drs 2/3-Doors	20950.000	31461.000	192....	1185....	1.0...	1.000	0.000	1.000	1.000
12	12....	TOYOTA Corolla 1.8 16V VVTLI 3DR T SPORT BNS 2...	19950.000	43610.000	192....	1185....	1.0...	1.000	1.000	1.000	1.000
13	13....	TOYOTA Corolla 1.8 16V VVTLI 3DR T SPORT 2/3-Do...	19600.000	32189.000	192....	1185....	1.0...	1.000	1.000	1.000	1.000
14	14....	TOYOTA Corolla 1.8 16V VVTLI 3DR T SPORT 2/3-Do...	21500.000	23000.000	192....	1185....	1.0...	1.000	1.000	1.000	1.000
15	15....	TOYOTA Corolla 1.8 16V VVTLI 3DR T SPORT 2/3-Do...	22500.000	34131.000	192....	1185....	1.0...	1.000	1.000	1.000	1.000
16	16....	TOYOTA Corolla 1.8 16V VVTLI 3DR T SPORT 2/3-Do...	22000.000	18739.000	192....	1185....	1.0...	1.000	1.000	1.000	1.000
17	17....	TOYOTA Corolla 1.8 16V VVTLI 3DR T SPORT 2/3-D...	22750.000	34000.000	192....	1185....	1.0...	1.000	1.000	1.000	1.000
18	18....	TOYOTA Corolla 1.6 VVTI Linea Terra Comfort 2/3-D...	17950.000	21716.000	110....	1105....	1.0...	1.000	0.000	1.000	1.000
19	19....	TOYOTA Corolla 1.6 16v L.SOL 2/3-Doors	16750.000	25563.000	110....	1065....	1.0...	1.000	1.000	1.000	1.000
20	20....	TOYOTA Corolla 1.6 16V VVT I 3DR TERRA 2/3-Doors	16950.000	64359.000	110....	1105....	1.0...	1.000	1.000	1.000	1.000
21	21....	TOYOTA Corolla 1.6 16V VVT I 3DR TERRA 2/3-Doors	15950.000	67660.000	110....	1105....	1.0...	1.000	1.000	1.000	1.000
22	22....	TOYOTA Corolla 1.6 16V VVT I 3DR SOL AUT4 2/3-D...	16950.000	43905.000	110....	1170....	1.0...	1.000	1.000	1.000	1.000
23	23....	TOYOTA Corolla 1.6 16V VVT I 3DR SOL 2/3-Doors	15950.000	56349.000	110....	1120....	1.0...	1.000	1.000	1.000	1.000
24	24....	TOYOTA Corolla 1.6 16V VVT I 3DR SOL 2/3-Doors	16950.000	32220.000	110....	1120....	1.0...	1.000	1.000	1.000	1.000
25	25....	TOYOTA Corolla 1.6 16V VVT I 3DR SOL 2/3-Doors	16250.000	25813.000	110....	1120....	1.0...	1.000	1.000	1.000	1.000
26	26....	TOYOTA Corolla 1.6 16V VVT I 3DR SOL 2/3-Doors	15950.000	28450.000	110....	1120....	1.0...	1.000	1.000	1.000	1.000
27	27....	TOYOTA Corolla 1.6 16V VVT I 3DR SOL 2/3-Doors	17495.000	34545.000	110....	1120....	1.0...	1.000	1.000	1.000	1.000
28	28....	TOYOTA Corolla 1.6 16V VVT I 3DR SOL 2/3-Doors	15750.000	41415.000	110....	1120....	1.0...	1.000	1.000	1.000	1.000

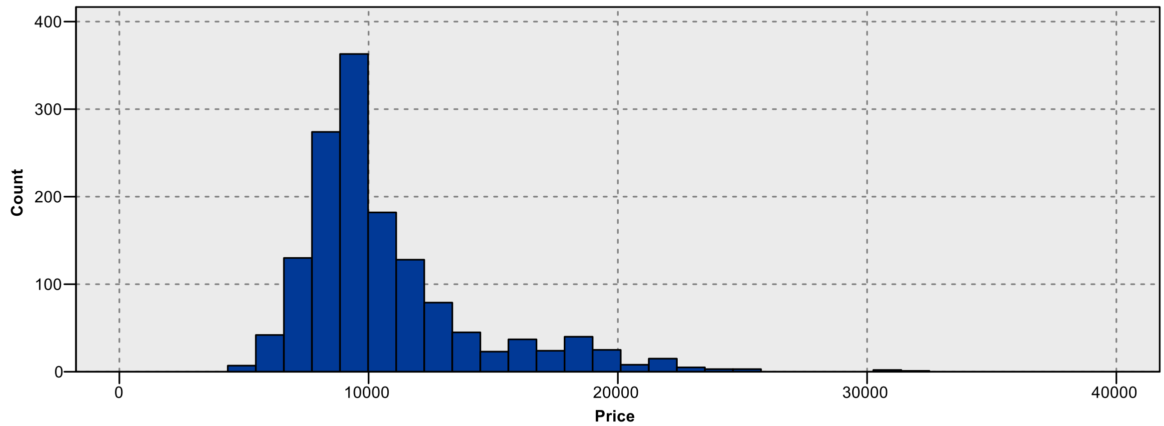
OK

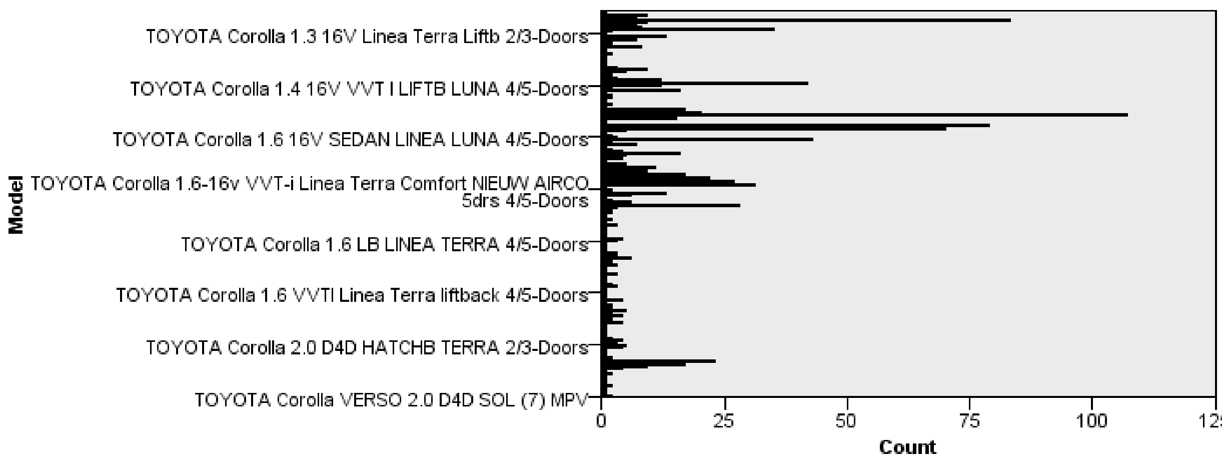
c) Then do check the quality of the data, we have connected the data audit node to see if there is any missing data from the given data set, there is no missing data, below is the attachment from data audit node.

Complete fields (%): 100% Complete records (%): 100%

Field	Measurement	Outliers	Extremes	Action	Impute Missing	Method	% Complete	Valid Records	Null Value
Id	Continuous	0	0	None	Never	Fixed	100	1436	
Model	Categorical	--	--	--	Never	Fixed	100	1436	
Price	Continuous	23	3	None	Never	Fixed	100	1436	
KM	Continuous	18	0	None	Never	Fixed	100	1436	
HP	Continuous	0	11	None	Never	Fixed	100	1436	
Weight	Continuous	25	5	None	Never	Fixed	100	1436	
ABS	Continuous	0	0	None	Never	Fixed	100	1436	
Airbag	Continuous	0	42	None	Never	Fixed	100	1436	
Boardcompu...	Continuous	0	0	None	Never	Fixed	100	1436	
Powered_Wi...	Continuous	0	0	None	Never	Fixed	100	1436	
Power_Steer...	Continuous	0	32	None	Never	Fixed	100	1436	

d) To have a visual of the data numerical as well as the categorical fields, below is the graph visuals:





e) Then we have connected a statistics node to the excel file node to see the correlation from the given data set. We have selected all the node except the first field the ID

Statistics Annotations

Collapse All Expand All

Price

Pearson Correlations

KM	-0.570	Strong
HP	0.315	Strong
Weight	0.581	Strong
ABS	0.306	Strong
Airbag	0.094	Strong
Boardcomputer	0.601	Strong
Powered_Windows	0.357	Strong
Power_Steering	0.064	Strong

KM

Pearson Correlations

Price	-0.570	Strong
HP	-0.334	Strong
Weight	-0.029	Strong
ABS	-0.177	Strong
Airbag	-0.018	Medium
Boardcomputer	-0.354	Strong
Powered_Windows	-0.156	Strong
Power_Steering	0.007	Weak

HP

Pearson Correlations

Price	0.315	Strong
KM	-0.334	Strong
Weight	0.090	Strong
ABS	0.058	Strong
Airbag	0.025	Medium
Boardcomputer	0.130	Strong
Powered_Windows	0.266	Strong
Power_Steering	0.049	Strong

Weight

Pearson Correlations

Price	0.581	Strong
KM	-0.029	Strong
HP	0.090	Strong
ABS	0.103	Strong
Airbag	0.030	Strong
Boardcomputer	0.274	Strong
Powered_Windows	0.213	Strong
Power_Steering	0.048	Strong

ABS

Pearson Correlations

Price	0.306	Strong
KM	-0.177	Strong
HP	0.058	Strong
Weight	0.103	Strong
Airbag	0.278	Strong
Boardcomputer	0.310	Strong
Powered_Windows	0.099	Strong
Power_Steering	0.255	Strong

Airbag

Pearson Correlations

Price	0.094	Strong
KM	-0.018	Medium
HP	0.025	Medium
Weight	0.030	Strong
ABS	0.278	Strong
Boardcomputer	0.112	Strong
Powered_Windows	0.122	Strong
Power_Steering	0.562	Strong

Boardcomputer

Pearson Correlations

Price	0.601	Strong
KM	-0.354	Strong
HP	0.130	Strong
Weight	0.274	Strong
ABS	0.310	Strong
Airbag	0.112	Strong
Powered_Windows	0.213	Strong
Power_Steering	0.087	Strong

File Edit Generate

Statistics Annotations

Collapse All Expand All

- Price
- KM
- HP
- Weight
- ABS
- Airbag
- Boardcomputer
- Powered_Windows
  - Pearson Correlations

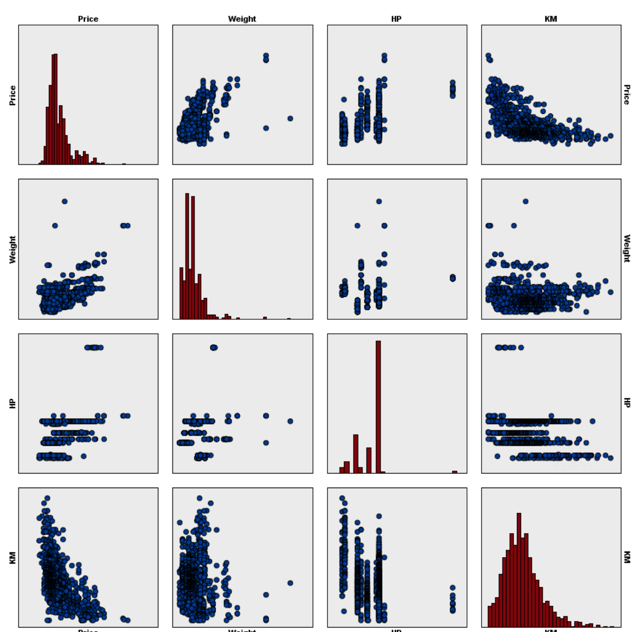
Price	0.357	Strong
KM	-0.156	Strong
HP	0.266	Strong
Weight	0.213	Strong
ABS	0.099	Strong
Airbag	0.122	Strong
Boardcomputer	0.213	Strong
Power_Steering	0.123	Strong

- Power_Steering
  - Pearson Correlations

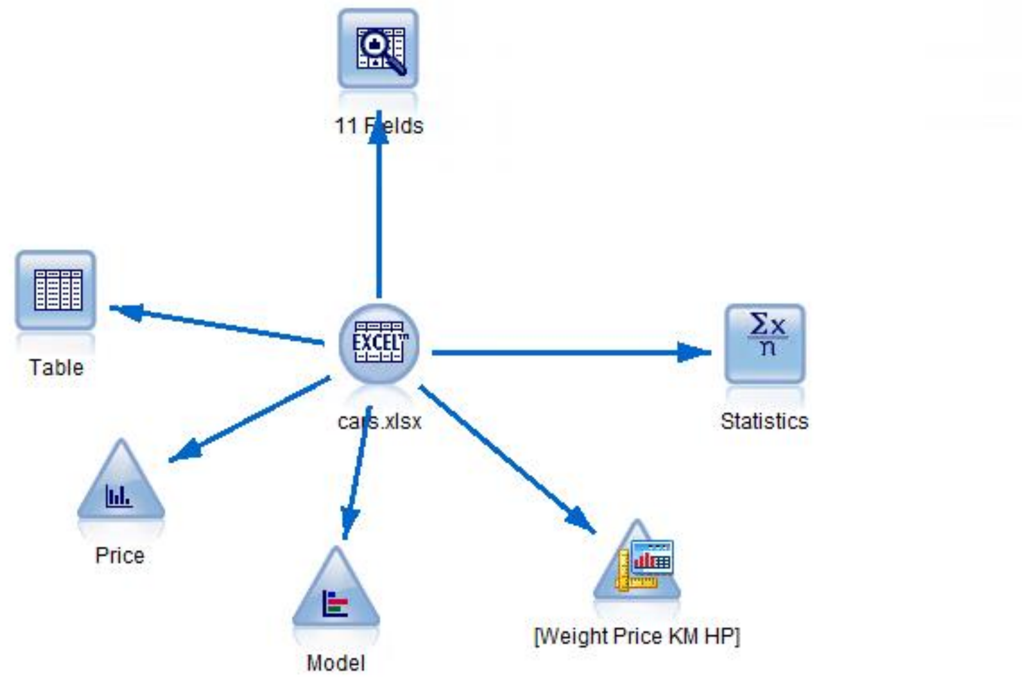
Price	0.064	Strong
KM	0.007	Weak
HP	0.049	Strong
Weight	0.048	Strong
ABS	0.255	Strong
Airbag	0.562	Strong
Boardcomputer	0.087	Strong
Powered_Windows	0.123	Strong

As we can see from the above that there are some strong correlations.

f) Finally, we have a scatter matrix plot for the Price, Weight, HP and KM



As we can see from the graph above, there are some strong correlations. Below is the complete stream visual.



For more details please refer to attached stream file from submitted zip folder.

Q4. The file BostonHousing.xlsx contains information over 500 census tracts in Boston, where for each tract 14 variables are recorded. Attribute MEDV represents the median value of a tract given the information of the other 13 attributes. The last attribute (CAT.MEDV) is a discrete recoding of MEDV such that it carries value 1 if MEDV >30 and 0 otherwise

Build kNN classification models of the median value of a tract (CAT.MEDV) using the attached dataset, with a 70%- 30% partition, and varying k between 1 and 5. Report the predictive performance of the models. What k would you choose? For one of the k values the training error rate is zero. Why would you say that the error is zero? (Use SPSS Modeler to complete this question)

a) First, we have uploaded the dataset into SPSS modeler.

b) To have the visual of the given data, we have connected a table node, below is the attachment from table node.

Table | Annotations

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV	CAT. MEDV
1	0.006	18.000	2.310	0.000	0.538	6.575	65.200	4.090	1.000	296.000	15.300	396.900	4.980	24.000	0.000
2	0.027	0.000	7.070	0.000	0.469	6.421	78.900	4.967	2.000	242.000	17.800	396.900	9.140	21.600	0.000
3	0.027	0.000	7.070	0.000	0.469	7.185	61.100	4.967	2.000	242.000	17.800	392.830	4.030	34.700	1.000
4	0.032	0.000	2.180	0.000	0.458	6.998	45.800	6.062	3.000	222.000	18.700	394.630	2.940	33.400	1.000
5	0.069	0.000	2.180	0.000	0.458	7.147	54.200	6.062	3.000	222.000	18.700	396.900	5.330	36.200	1.000
6	0.030	0.000	2.180	0.000	0.458	6.430	58.700	6.062	3.000	222.000	18.700	394.120	5.210	28.700	0.000
7	0.088	12.500	7.870	0.000	0.524	6.012	66.600	5.561	5.000	311.000	15.200	395.600	12.430	22.900	0.000
8	0.145	12.500	7.870	0.000	0.524	6.172	96.100	5.950	5.000	311.000	15.200	396.900	19.150	27.100	0.000
9	0.211	12.500	7.870	0.000	0.524	5.631	100.000	6.082	5.000	311.000	15.200	386.630	29.930	16.500	0.000
10	0.170	12.500	7.870	0.000	0.524	6.004	85.900	6.592	5.000	311.000	15.200	386.710	17.100	18.900	0.000
11	0.225	12.500	7.870	0.000	0.524	6.377	94.300	6.347	5.000	311.000	15.200	392.520	20.450	15.000	0.000
12	0.117	12.500	7.870	0.000	0.524	6.009	82.900	6.227	5.000	311.000	15.200	396.900	13.270	18.900	0.000
13	0.094	12.500	7.870	0.000	0.524	5.889	39.000	5.451	5.000	311.000	15.200	390.500	15.710	21.700	0.000
14	0.630	0.000	8.140	0.000	0.538	5.949	61.800	4.707	4.000	307.000	21.000	396.900	8.260	20.400	0.000
15	0.638	0.000	8.140	0.000	0.538	6.096	84.500	4.462	4.000	307.000	21.000	380.020	10.260	18.200	0.000
16	0.627	0.000	8.140	0.000	0.538	5.834	56.500	4.499	4.000	307.000	21.000	395.620	8.470	19.900	0.000
17	1.054	0.000	8.140	0.000	0.538	5.935	29.300	4.499	4.000	307.000	21.000	386.850	6.580	23.100	0.000
18	0.784	0.000	8.140	0.000	0.538	5.990	81.700	4.258	4.000	307.000	21.000	386.750	14.670	17.500	0.000
19	0.803	0.000	8.140	0.000	0.538	5.456	36.600	3.796	4.000	307.000	21.000	288.990	11.690	20.200	0.000
20	0.726	0.000	8.140	0.000	0.538	5.727	69.500	3.796	4.000	307.000	21.000	390.950	11.280	18.200	0.000
21	1.252	0.000	8.140	0.000	0.538	5.570	98.100	3.798	4.000	307.000	21.000	376.570	21.020	13.600	0.000
22	0.852	0.000	8.140	0.000	0.538	5.965	89.200	4.012	4.000	307.000	21.000	392.530	13.830	19.600	0.000
23	1.232	0.000	8.140	0.000	0.538	6.142	91.700	3.977	4.000	307.000	21.000	396.900	18.720	15.200	0.000

OK

c) Then we have check quality of the data, by connecting the audit node to Excel node, there are no missing data, below is the attached snapshot.

Data Audit of [15 fields]

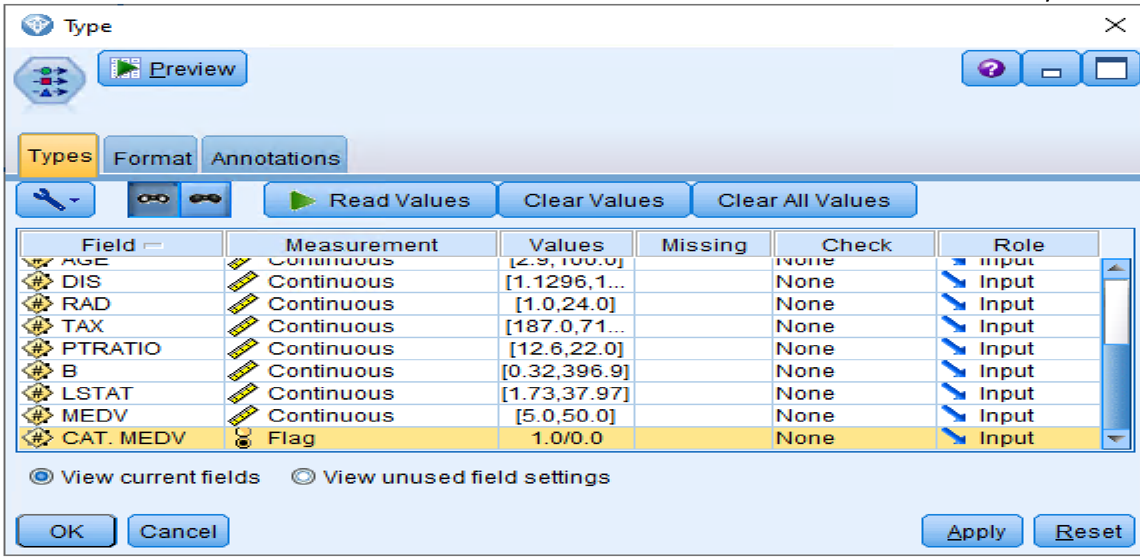
File | Edit | Generate

Audit | Quality | Annotations

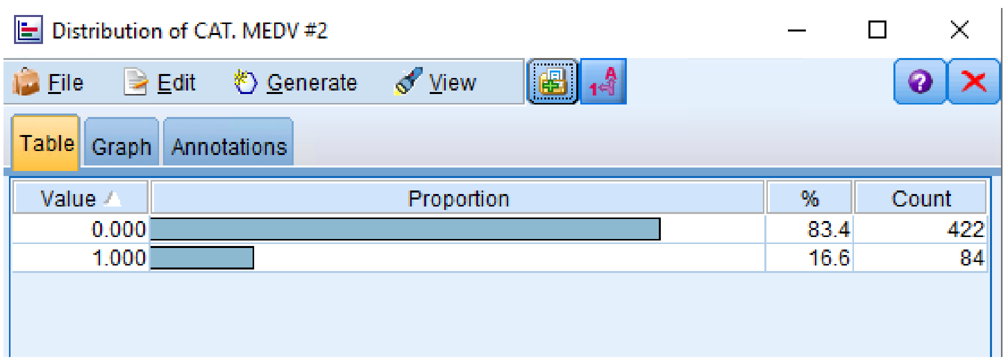
Complete fields (%): 100% Complete records (%): 100%

Field	Measurement	Outliers	Extremes	Action	Impute Missing	Method	% Complete	Valid Records	Null Value	Empty Stri
CRIM	Continuous	4	4	None	Never	Fixed	100	506	0	
ZN	Continuous	14	0	None	Never	Fixed	100	506	0	
INDUS	Continuous	0	0	None	Never	Fixed	100	506	0	
CHAS	Continuous	35	0	None	Never	Fixed	100	506	0	
NOX	Continuous	0	0	None	Never	Fixed	100	506	0	
RM	Continuous	8	0	None	Never	Fixed	100	506	0	
AGE	Continuous	0	0	None	Never	Fixed	100	506	0	
DIS	Continuous	5	0	None	Never	Fixed	100	506	0	
RAD	Continuous	0	0	None	Never	Fixed	100	506	0	
TAX	Continuous	0	0	None	Never	Fixed	100	506	0	
PTRATIO	Continuous	0	0	None	Never	Fixed	100	506	0	
B	Continuous	25	0	None	Never	Fixed	100	506	0	
LSTAT	Continuous	5	0	None	Never	Fixed	100	506	0	
MEDV	Continuous	0	0	None	Never	Fixed	100	506	0	
CAT. MEDV	Continuous	0	0	None	Never	Fixed	100	506	0	

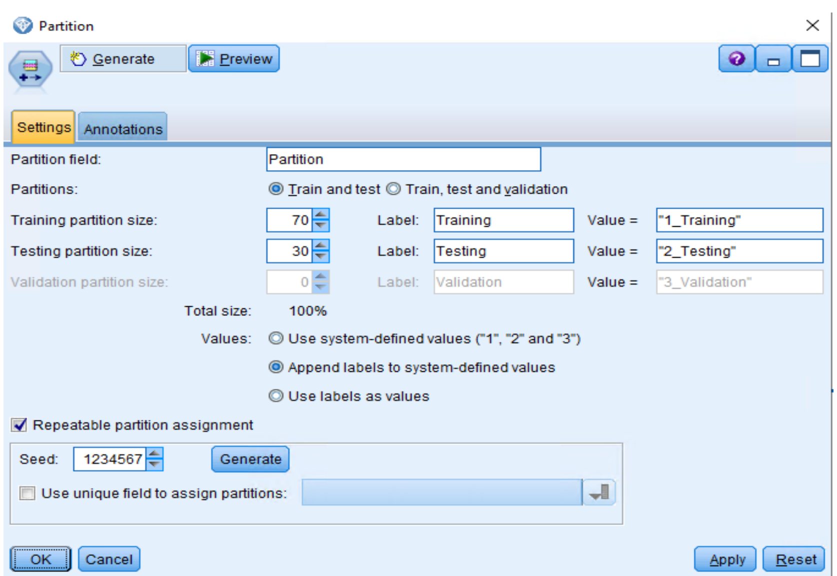
d) As we have built kNN classification models of the median value of a tract (CAT.MEDV), we first check the type of the CAT.MEDV, here we have to change the measurement to flag, which can be seen from the snapshot below:



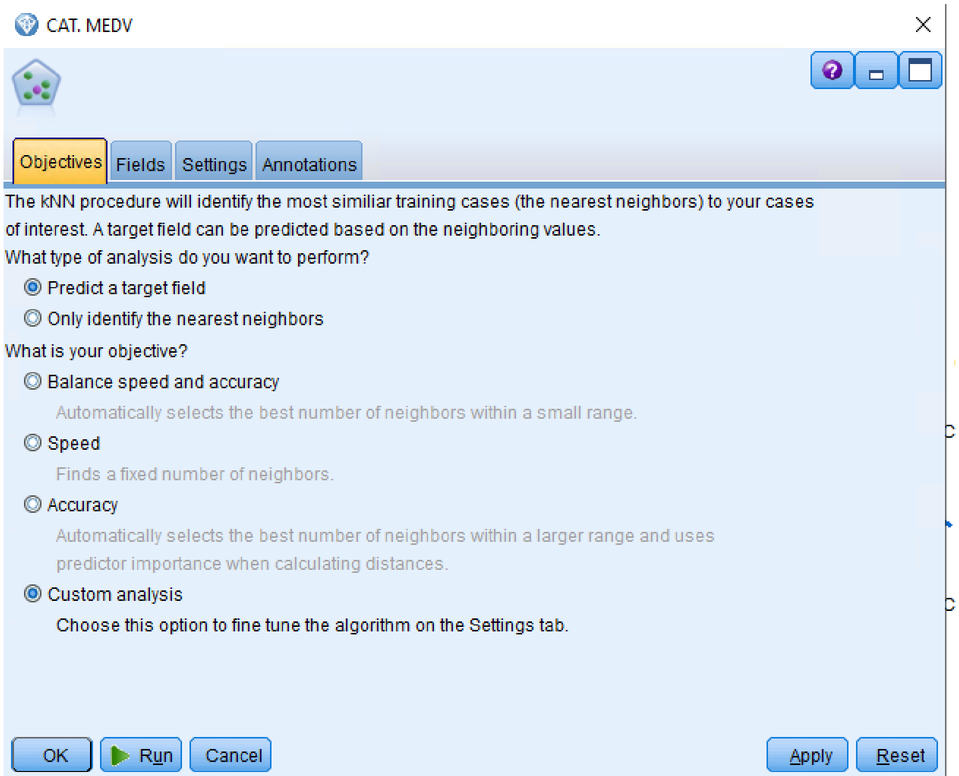
e) We have connected distribution node to have a visual of the CAT.MEDV attribute, below is the snapshot.

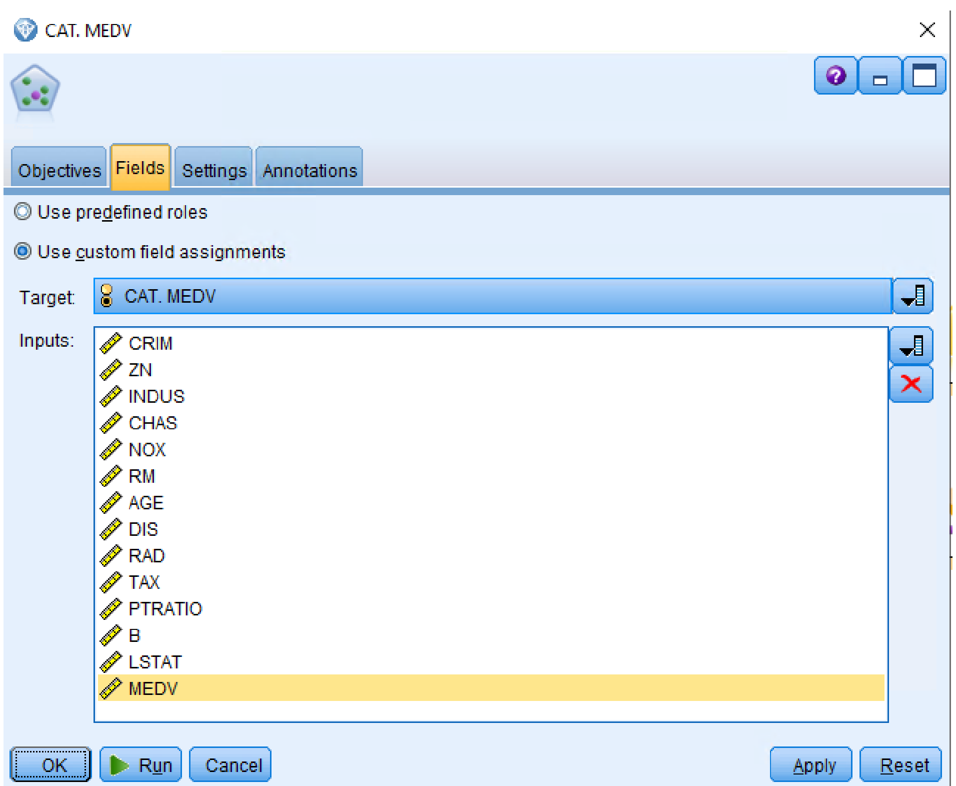


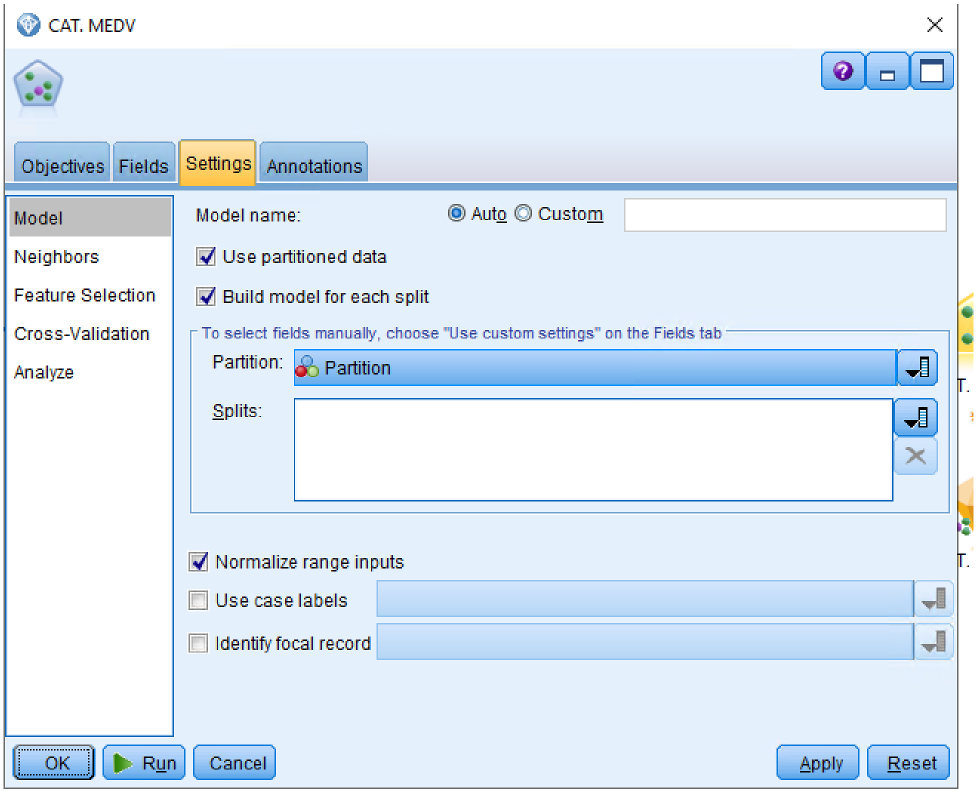
f) The next step is the partition the dataset in required ratio (70: 30).

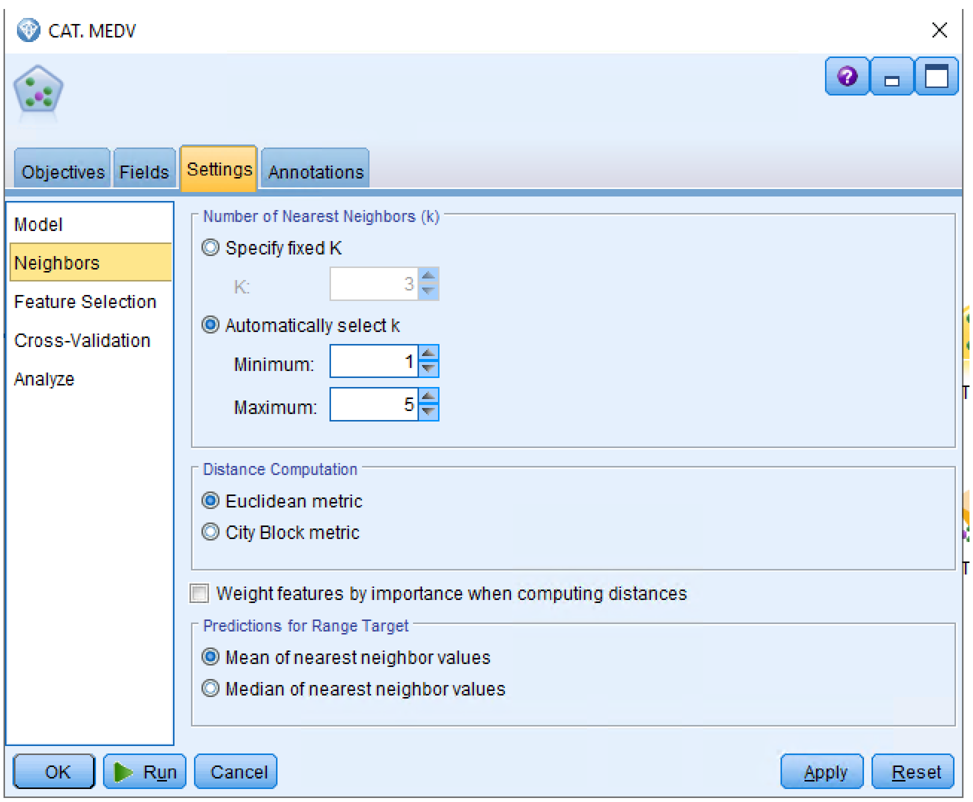


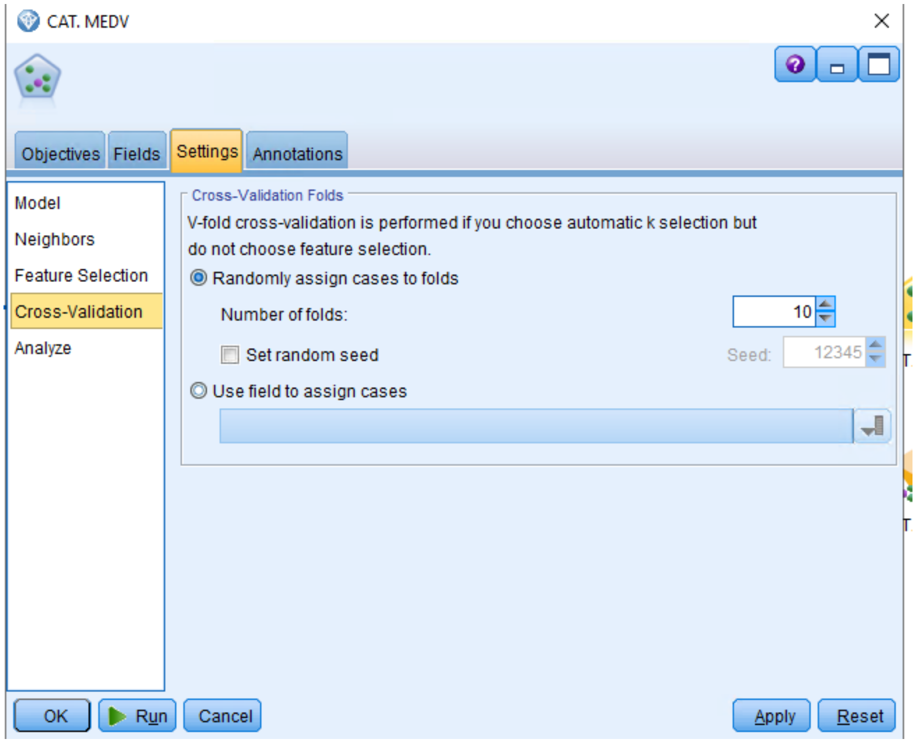
g) Using the KNN node from modeling we have implement the KNN with the following features are asked:

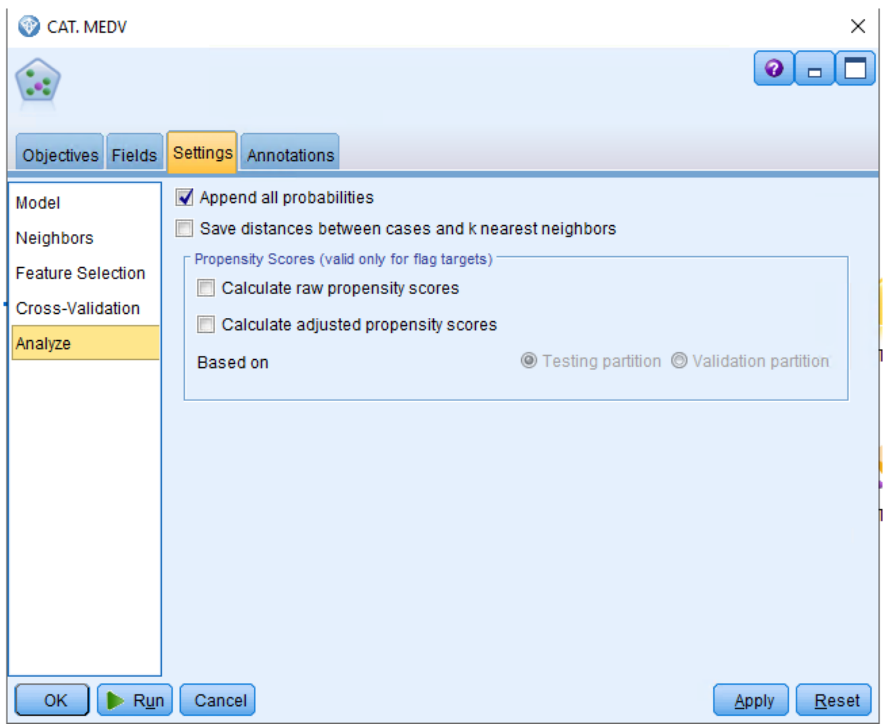




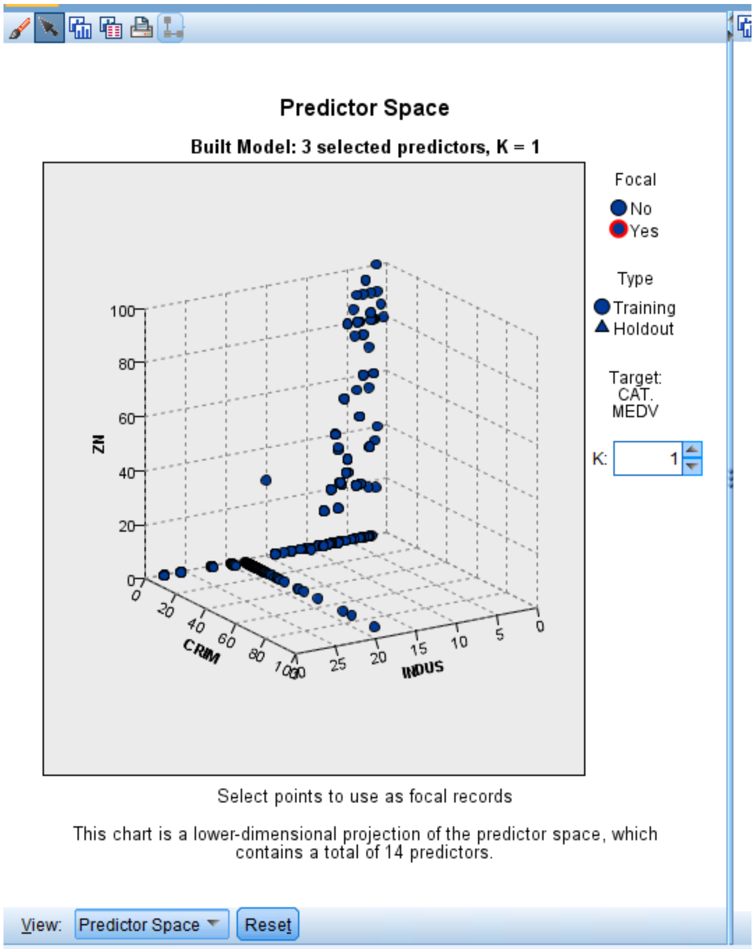

CAT. MEDV
Objectives
Fields
Settings
Annotations
Model
Neighbors
Feature Selection
Cross-Validation
Analyze
Model name:
Auto
Custom
Use partitioned data
Build model for each split
To select fields manually, choose "Use custom settings" on the Fields tab
Partition:
Partition
Splits:
Normalize range inputs
Use case labels
Identify focal record
OK
Run
Cancel
Apply
Reset


CAT. MEDV
Objectives
Fields
Settings
Annotations
Model
Neighbors
Feature Selection
Cross-Validation
Analyze
Number of Nearest Neighbors (k)
Specify fixed K
K:
3
Automatically select k
Minimum:
1
Maximum:
5
Distance Computation
Euclidean metric
City Block metric
Weight features by importance when computing distances
Predictions for Range Target
Mean of nearest neighbor values
Median of nearest neighbor values
OK
Run
Cancel
Apply
Reset

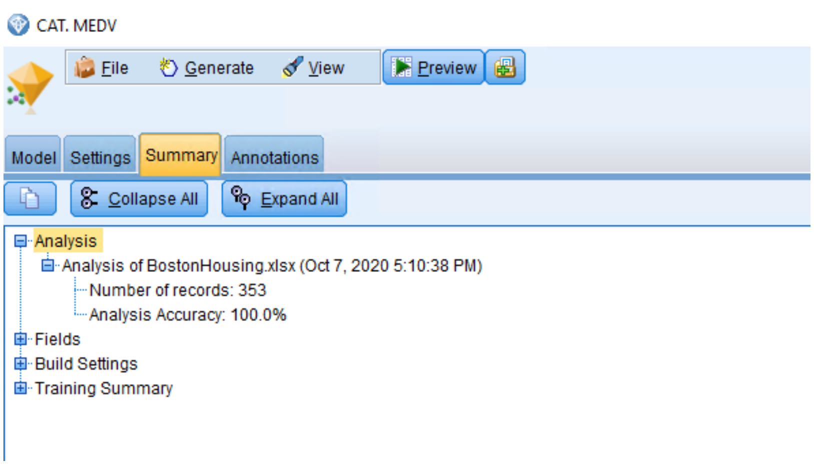




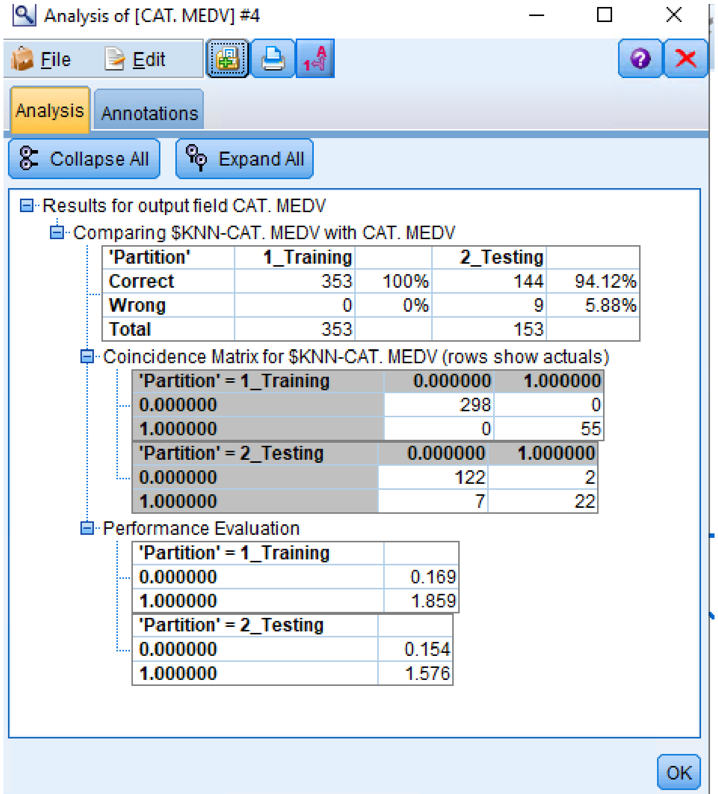
After creating the implementing the KNN with above specification, we have the diamond where the computed K value is 1:



When we had the view for summary tab from our diamond we can see the analysis accuracy is 100%



To have a better understanding, we have added a Analysis node from our KNN model, break down analysis field of KNN_CAT.MEDV, we get:



Results for output field CAT. MEDV

Comparing $KNN-CAT. MEDV with CAT. MEDV

'Partition'	1_Training		2_Testing	
Correct	353	100%	144	94.12%
Wrong	0	0%	9	5.88%
Total	353		153	

Coincidence Matrix for $KNN-CAT. MEDV (rows show actuals)

'Partition' = 1_Training	0.000000	1.000000
0.000000	298	0
1.000000	0	55

'Partition' = 2_Testing	0.000000	1.000000
0.000000	122	2
1.000000	7	22

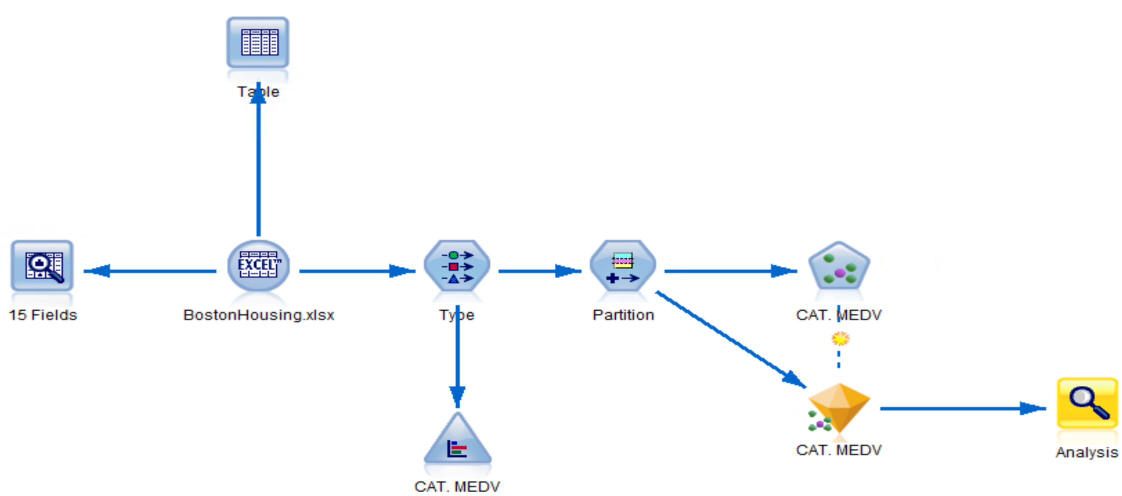
Performance Evaluation

'Partition' = 1_Training	
0.000000	0.169
1.000000	1.859

'Partition' = 2_Testing	
0.000000	0.154
1.000000	1.576

From the above image,
Predictive accuracy of Training set = (298+55)/ (298+0+55+0) = 100%
Predictive accuracy of Testing set = (122+22)/ (122+2+7+22) = 94.12%
As we can see from the above that the given data set is skewed and we are not able to predict the true prediction.

Below is the complete stream file:



Q5. This is a modified version of the Car Evaluation Database was derived from a simple hierarchical decision model originally developed for the demonstration of DEX, M. Bohanec, V. Rajkovic: Expert system for decision making. Sistemica 1(1), pp. 145-157, 1990.).

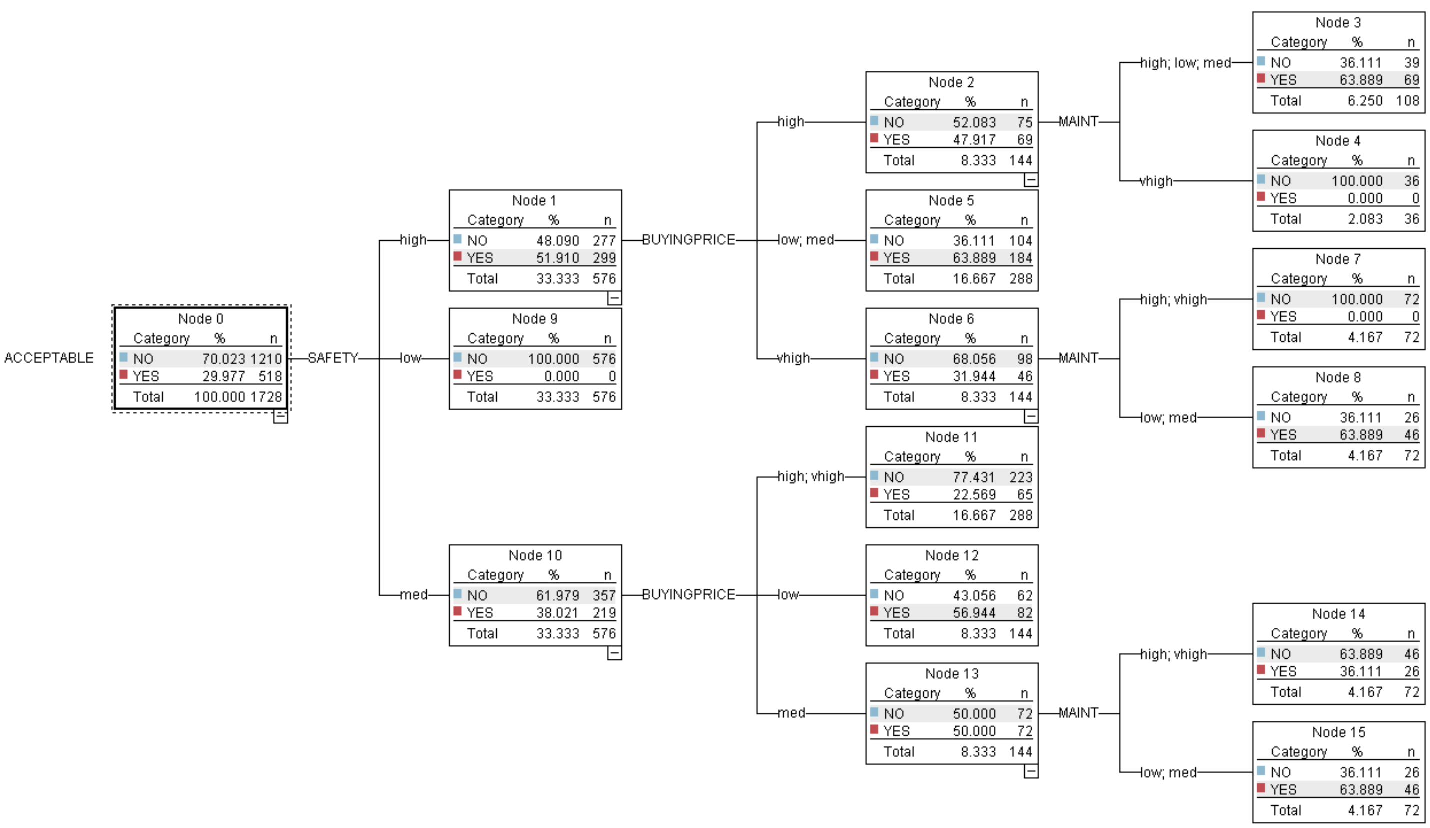
The Car Evaluation Database contains examples with the structural information removed, i.e., directly relates CAR to three input attributes: buyingprice, maintenance, safety.

<u>Attribute Information</u>**:**

Class attribute Acceptable: YES, NO

Attributes:
buyingprice: vhigh, high, med, low.
maint: vhigh, high, med, low.
safety: low, med, high.

A C5.0 algorithm is applied on the whole data set, yielding the decision tree below.



Write the rules for car acceptability that can be derived from the decision tree, together with the support and the confidence of each rule (hint: check my notes and the textbook to calculate confidence and

Answer: The rules that can be derived from the decision tree together is given as below:

Antecedent	Consequence	Support	Confidence
If (safety = high) and (buying price = high) and (Maintenance = high, low, med)	Then Acceptable = Yes	69/1728	69/108 = 63.889%
If (safety = high) and (buying price = high) and (Maintenance = very high)	Then Acceptable = No	36/1728	36/36 = 100%
If (safety = high) and (buying price = very high) and (Maintenance = high, very high)	Then Acceptable = No	72/1728	72/72 = 100%
If (safety = high) and (buying price = very high) and (Maintenance = low, med)	Then Acceptable = Yes	46/1728	46/72 = 63.889%
If (safety = med) and (buying price = med) and (Maintenance = high, very high)	Then Acceptable = No	46/1728	46/72 = 63.889%
If (safety = med) and (buying price = med) and (Maintenance = low, med)	Then Acceptable = Yes	46/1728	46/72 = 63.889%
If (safety = high) and (buying price = low, med)	Then Acceptable = No	184/1728	184/288 = 63.889%
If (safety = med) and (buying price = high, very high)	Then Acceptable = No	233/1728	223/288 = 77.43%
If (safety = med) and (buying price = low)	Then Acceptable = Yes	82/1728	82/144 = 56.944%
If (Safety = low)	Then Acceptable = No	576/1728	576/576 = 100%