FEBRUARY 13, 2018

DIAMOND DATA MINING

MICROSOFT SQL SERVER ANALYSIS SERVICES 2015

HUONG PHAM & SURUCHI CHAUDHARAY

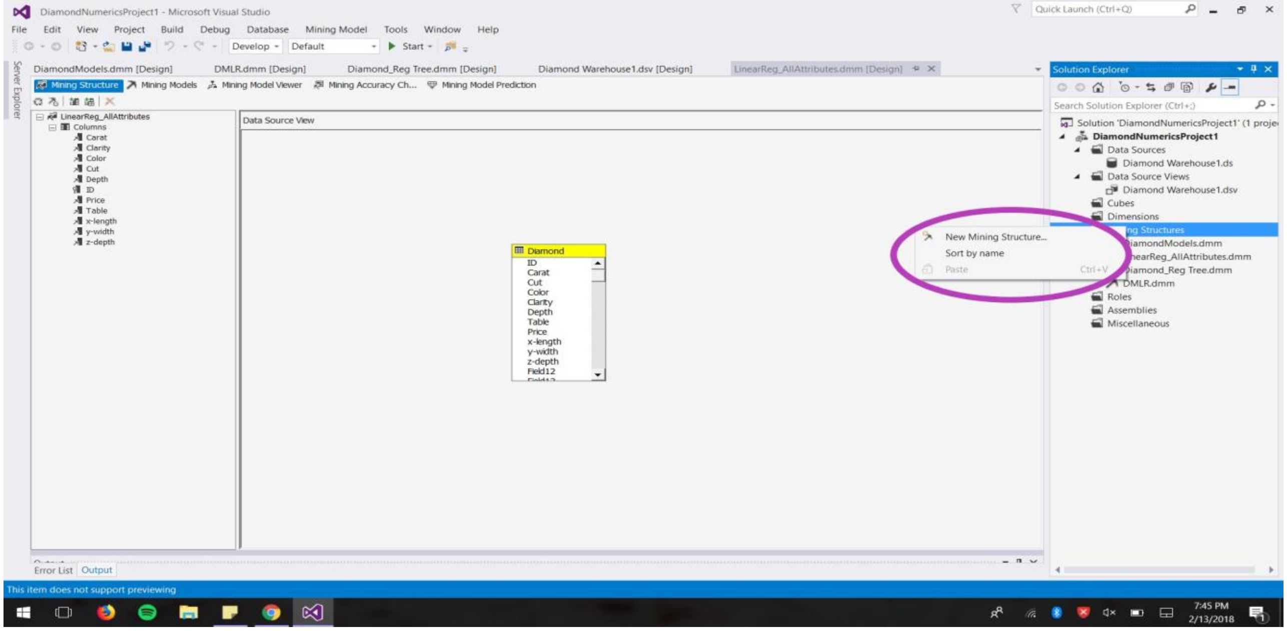
BUSINESS CASE

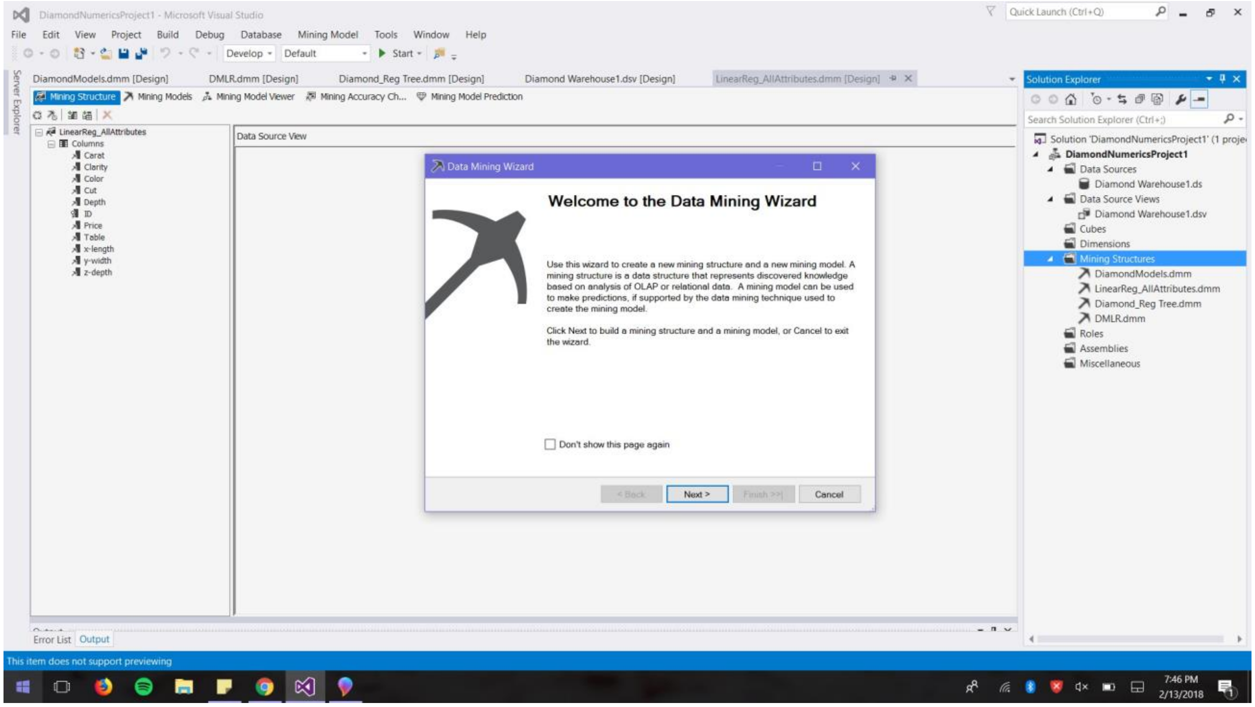
The goal of this data mining project is to generate an algorithm model to predict the price of diamond. This model will predict the price of model based on the most significant attributes, being carat, clarity, color, cut, x-length. With this prediction, a seller can estimate the market price of diamond. On the other hand, this predictive analysis is also helpful for buyers to evaluate the price and quality of the diamond that they are purchasing. The size of our data set is almost 54,000 records. The tool used to do this project is Microsoft SQL Server 2015

DATA IMPORT PROCESS

We imported our dataset into SQL Server using the SQL Server Import Wizard. We created the data source and the data source view. The following screenshots will show how we create a mining model in SASS

Step 1: Right click on the Mining Structure, and choose New Mining Structure





Data Mining Wizard

Welcome to the Data Mining Wizard

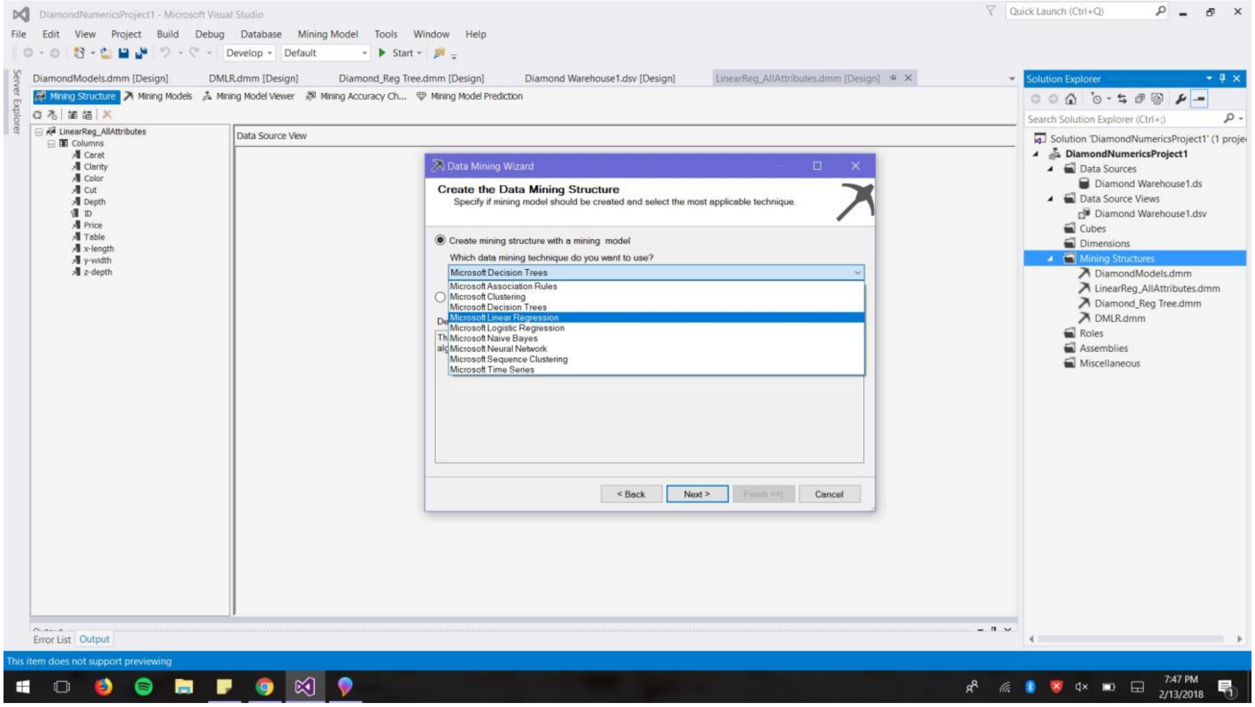
Use this wizard to create a new mining structure and a new mining model. A mining structure is a data structure that represents discovered knowledge based on analysis of OLAP or relational data. A mining model can be used to make predictions, if supported by the data mining technique used to create the mining model.

Click Next to build a mining structure and a mining model, or Cancel to exit the wizard.

□ Don't show this page again

< Back | Next > | Finish >>| | Cancel

Step 2: We can choose different mining models from the drop bar.


DiamondNumericsProject1 - Microsoft Visual Studio
File Edit View Project Build Debug Database Mining Model Tools Window Help
LinearReg_AllAttributes.dmm [Design]
Mining Structure
Data Source View
Solution Explorer
Mining Structures
This item does not support previewing
7:47 PM
2/13/2018

Data Mining Wizard

Create the Data Mining Structure

Specify if mining model should be created and select the most applicable technique.

● Create mining structure with a mining model

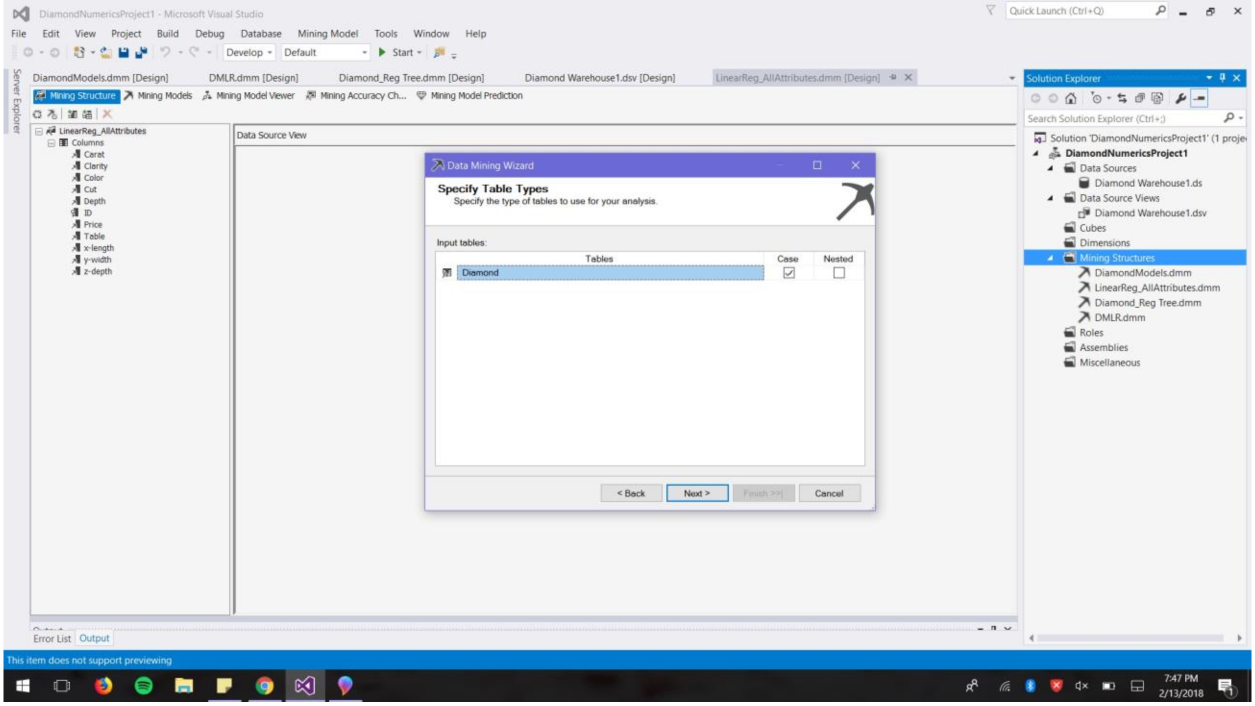
Which data mining technique do you want to use?

Microsoft Decision Trees

- Microsoft Association Rules
- Microsoft Clustering
- Microsoft Decision Trees
- Microsoft Linear Regression
- Microsoft Logistic Regression
- Microsoft Naive Bayes
- Microsoft Neural Network
- Microsoft Sequence Clustering
- Microsoft Time Series

< Back | Next > | Finish >>| | Cancel

Step 3: Since we only have one table, the Diamond table will be our Case table


DiamondNumericsProject1 - Microsoft Visual Studio
File Edit View Project Build Debug Database Mining Model Tools Window Help
Mining Structure
Mining Models
Mining Model Viewer
Mining Accuracy Ch...
Mining Model Prediction
Data Source View
Solution Explorer
< Back
Next >
Finish >>|
Cancel
This item does not support previewing

Data Mining Wizard

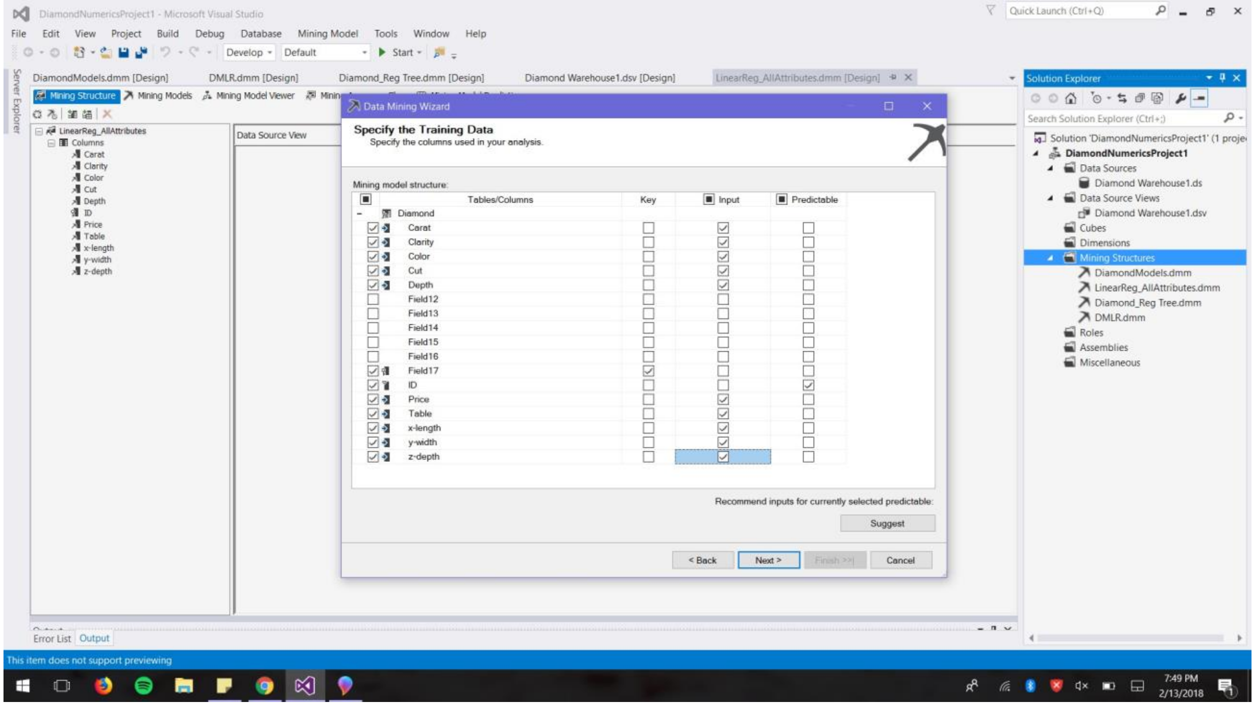
Specify Table Types

Specify the type of tables to use for your analysis.

Input tables:

Tables	Case	Nested
Diamond	☑	☐

Step 4: We choose which attributes to be the inputs, ID key and predictable attribute. If we click Suggest at the bottom right, MSS will provide a list of attributes that are more significant relevant to the predictable value.


DiamondNumericsProject1 - Microsoft Visual Studio
File Edit View Project Build Debug Database Mining Model Tools Window Help
Mining Structure
Mining Models
Mining Model Viewer
Data Source View
Solution Explorer
< Back
Next >
Finish >>|
Cancel
This item does not support previewing

Data Mining Wizard

Specify the Training Data

Specify the columns used in your analysis.

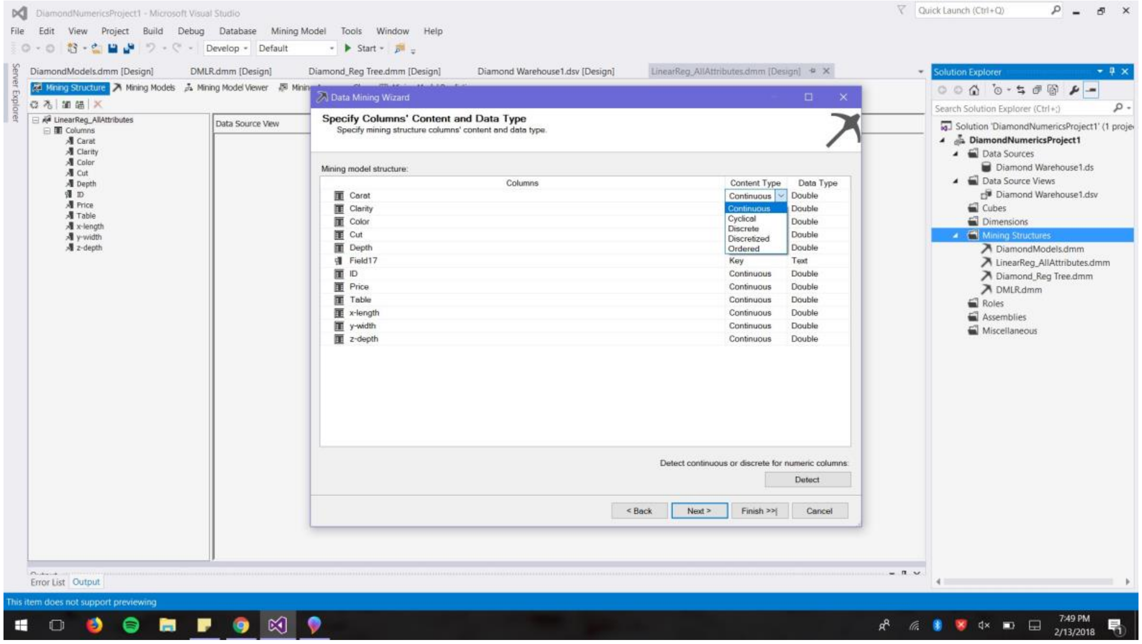
Mining model structure:

■	Tables/Columns	Key	■ Input	■ Predictable
–	Diamond			
☑	Carat	☐	☑	☐
☑	Clarity	☐	☑	☐
☑	Color	☐	☑	☐
☑	Cut	☐	☑	☐
☑	Depth	☐	☑	☐
☐	Field12	☐	☐	☐
☐	Field13	☐	☐	☐
☐	Field14	☐	☐	☐
☐	Field15	☐	☐	☐
☐	Field16	☐	☐	☐
☑	Field17	☑	☐	☐
☑	ID	☐	☐	☑
☑	Price	☐	☑	☐
☑	Table	☐	☑	☐
☑	x-length	☐	☑	☐
☑	y-width	☐	☑	☐
☑	z-depth	☐	☑	☐

Recommend inputs for currently selected predictable:

Suggest

Step 5: If we click the drop bar, we can change the Content Type of the data


Data Mining Wizard

Specify Columns' Content and Data Type

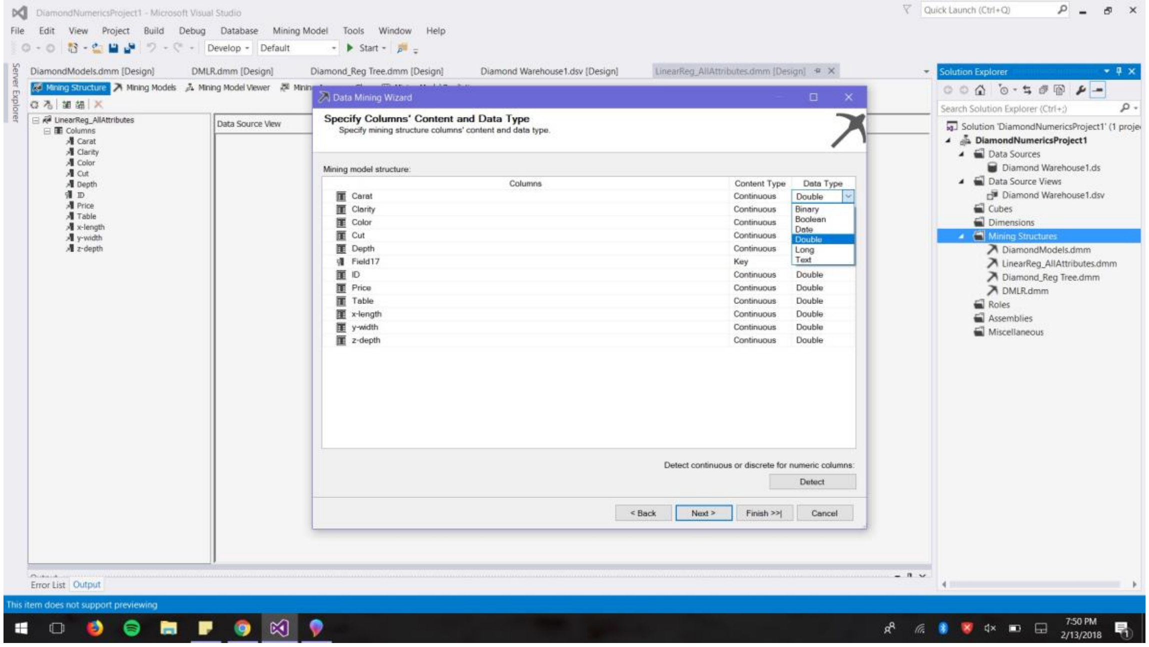
Specify mining structure columns' content and data type.

Mining model structure:

Columns	Content Type	Data Type
Carat	Continuous	Double
Clarity		Double
Color		Double
Cut		Double
Depth		Double
Field17	Key	Text
ID	Continuous	Double
Price	Continuous	Double
Table	Continuous	Double
x-length	Continuous	Double
y-width	Continuous	Double
z-depth	Continuous	Double

Detect continuous or discrete for numeric columns:

Step 6: We can also change the data type with the drop bar. Once we hit Finish, we have a mining model.


Data Mining Wizard

Specify Columns' Content and Data Type

Specify mining structure columns' content and data type.

Mining model structure:

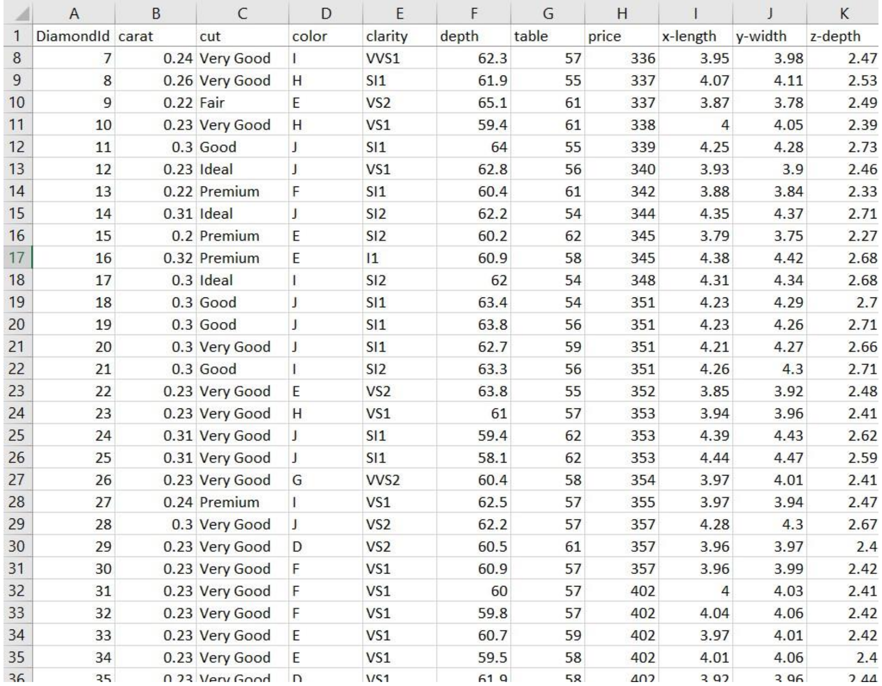
Columns	Content Type	Data Type
Carat	Continuous	Double
Clarity	Continuous	
Color	Continuous	
Cut	Continuous	
Depth	Continuous	
Field17	Key	
ID	Continuous	Double
Price	Continuous	Double
Table	Continuous	Double
x-length	Continuous	Double
y-width	Continuous	Double
z-depth	Continuous	Double

Detect continuous or discrete for numeric columns:

We decided to go with Linear Regression since it was the algorithm that met the requirements of our dataset (only numerical data) and our prediction value (numeric). We wanted to show the correlation between the significant attributes in our dataset and its effect on the price of the diamond. We also wanted to understand which attributes are the most important and if there is one single attribute that dominates the decision making when it comes to diamond pricing.

DATA SOURCE & DATA PREPARATION

We found the dataset of diamond on Kaggle. The size of the data is appropriate for the scope of the project which was 54000 rows. We agreed that the data is relatively clean for us to start our first data mining project. We examined three datasets and chose this diamond dataset because of its simplicity. This allowed us to spend more time to learn how to use Microsoft SQL Server, and learn how to interpret the results in this tool. Below is a screenshot of the original data.



	A	B	C	D	E	F	G	H	I	J	K
1	DiamondId	carat	cut	color	clarity	depth	table	price	x-length	y-width	z-depth
8	7	0.24	Very Good	I	VVS1	62.3	57	336	3.95	3.98	2.47
9	8	0.26	Very Good	H	SI1	61.9	55	337	4.07	4.11	2.53
10	9	0.22	Fair	E	VS2	65.1	61	337	3.87	3.78	2.49
11	10	0.23	Very Good	H	VS1	59.4	61	338	4	4.05	2.39
12	11	0.3	Good	J	SI1	64	55	339	4.25	4.28	2.73
13	12	0.23	Ideal	J	VS1	62.8	56	340	3.93	3.9	2.46
14	13	0.22	Premium	F	SI1	60.4	61	342	3.88	3.84	2.33
15	14	0.31	Ideal	J	SI2	62.2	54	344	4.35	4.37	2.71
16	15	0.2	Premium	E	SI2	60.2	62	345	3.79	3.75	2.27
17	16	0.32	Premium	E	I1	60.9	58	345	4.38	4.42	2.68
18	17	0.3	Ideal	I	SI2	62	54	348	4.31	4.34	2.68
19	18	0.3	Good	J	SI1	63.4	54	351	4.23	4.29	2.7
20	19	0.3	Good	J	SI1	63.8	56	351	4.23	4.26	2.71
21	20	0.3	Very Good	J	SI1	62.7	59	351	4.21	4.27	2.66
22	21	0.3	Good	I	SI2	63.3	56	351	4.26	4.3	2.71
23	22	0.23	Very Good	E	VS2	63.8	55	352	3.85	3.92	2.48
24	23	0.23	Very Good	H	VS1	61	57	353	3.94	3.96	2.41
25	24	0.31	Very Good	J	SI1	59.4	62	353	4.39	4.43	2.62
26	25	0.31	Very Good	J	SI1	58.1	62	353	4.44	4.47	2.59
27	26	0.23	Very Good	G	VVS2	60.4	58	354	3.97	4.01	2.41
28	27	0.24	Premium	I	VS1	62.5	57	355	3.97	3.94	2.47
29	28	0.3	Very Good	J	VS2	62.2	57	357	4.28	4.3	2.67
30	29	0.23	Very Good	D	VS2	60.5	61	357	3.96	3.97	2.4
31	30	0.23	Very Good	F	VS1	60.9	57	357	3.96	3.99	2.42
32	31	0.23	Very Good	F	VS1	60	57	402	4	4.03	2.41
33	32	0.23	Very Good	F	VS1	59.8	57	402	4.04	4.06	2.42
34	33	0.23	Very Good	E	VS1	60.7	59	402	3.97	4.01	2.42
35	34	0.23	Very Good	E	VS1	59.5	58	402	4.01	4.06	2.4
36	35	0.23	Very Good	D	VS1	61.9	58	402	3.92	3.96	2.44

We had to prepare the data by changing the following attributes to numeric values.

Cut

Fair (worst)	1
Good	2

Very Good	3
Premium	4
Ideal (best)	5

Color

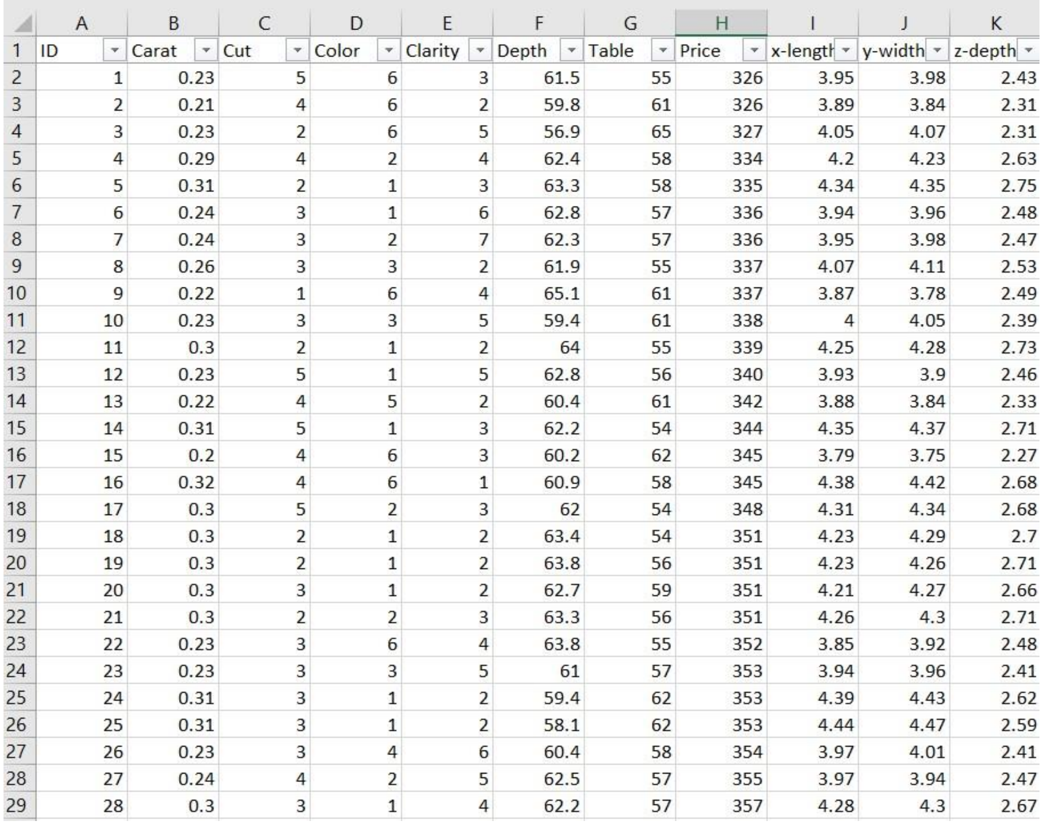
J(worst)	1
I	2
H	3
G	4
F	5
E	6
D(best)	7

Clarity

1I (worst)	1
SI2	2
SI1	3
VS2	4
VS1	5

VVS2	6
VVS1	7
IF (best)	8

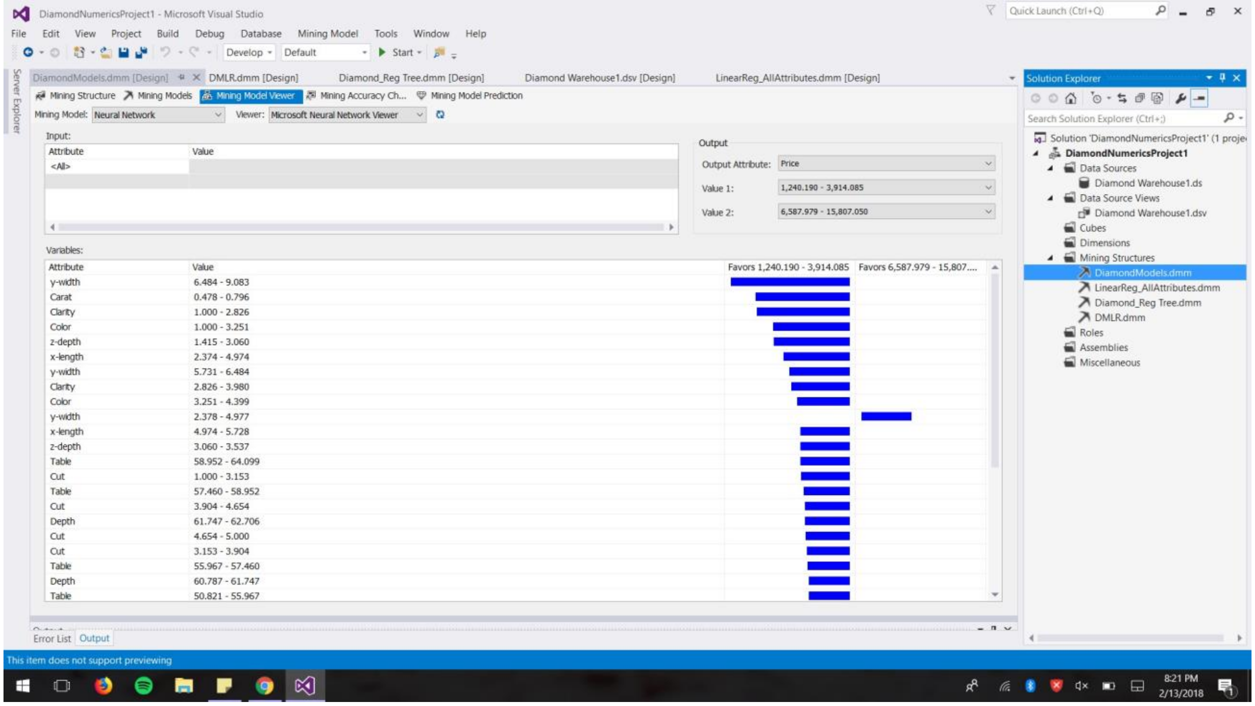
This is how the data looks after we cleaned it.



	A	B	C	D	E	F	G	H	I	J	K
1	ID	Carat	Cut	Color	Clarity	Depth	Table	Price	x-lengt	y-width	z-depth
2	1	0.23	5	6	3	61.5	55	326	3.95	3.98	2.43
3	2	0.21	4	6	2	59.8	61	326	3.89	3.84	2.31
4	3	0.23	2	6	5	56.9	65	327	4.05	4.07	2.31
5	4	0.29	4	2	4	62.4	58	334	4.2	4.23	2.63
6	5	0.31	2	1	3	63.3	58	335	4.34	4.35	2.75
7	6	0.24	3	1	6	62.8	57	336	3.94	3.96	2.48
8	7	0.24	3	2	7	62.3	57	336	3.95	3.98	2.47
9	8	0.26	3	3	2	61.9	55	337	4.07	4.11	2.53
10	9	0.22	1	6	4	65.1	61	337	3.87	3.78	2.49
11	10	0.23	3	3	5	59.4	61	338	4	4.05	2.39
12	11	0.3	2	1	2	64	55	339	4.25	4.28	2.73
13	12	0.23	5	1	5	62.8	56	340	3.93	3.9	2.46
14	13	0.22	4	5	2	60.4	61	342	3.88	3.84	2.33
15	14	0.31	5	1	3	62.2	54	344	4.35	4.37	2.71
16	15	0.2	4	6	3	60.2	62	345	3.79	3.75	2.27
17	16	0.32	4	6	1	60.9	58	345	4.38	4.42	2.68
18	17	0.3	5	2	3	62	54	348	4.31	4.34	2.68
19	18	0.3	2	1	2	63.4	54	351	4.23	4.29	2.7
20	19	0.3	2	1	2	63.8	56	351	4.23	4.26	2.71
21	20	0.3	3	1	2	62.7	59	351	4.21	4.27	2.66
22	21	0.3	2	2	3	63.3	56	351	4.26	4.3	2.71
23	22	0.23	3	6	4	63.8	55	352	3.85	3.92	2.48
24	23	0.23	3	3	5	61	57	353	3.94	3.96	2.41
25	24	0.31	3	1	2	59.4	62	353	4.39	4.43	2.62
26	25	0.31	3	1	2	58.1	62	353	4.44	4.47	2.59
27	26	0.23	3	4	6	60.4	58	354	3.97	4.01	2.41
28	27	0.24	4	2	5	62.5	57	355	3.97	3.94	2.47
29	28	0.3	3	1	4	62.2	57	357	4.28	4.3	2.67

DATA MINING ALGORITHMS

We tried regression tree and neural network. With the result of the regression tree, the errors of the predicted prices were very big compared to the actual prices. The neural network model in Microsoft SQL Server did not provide any helpful insight with the prediction of diamond price.



Mining Model: Neural Network
Viewer: Microsoft Neural Network Viewer

Input:

Attribute	Value
<All>	

Output

Output Attribute: Price
Value 1: 1,240.190 - 3,914.085
Value 2: 6,587.979 - 15,807.050

Variables:

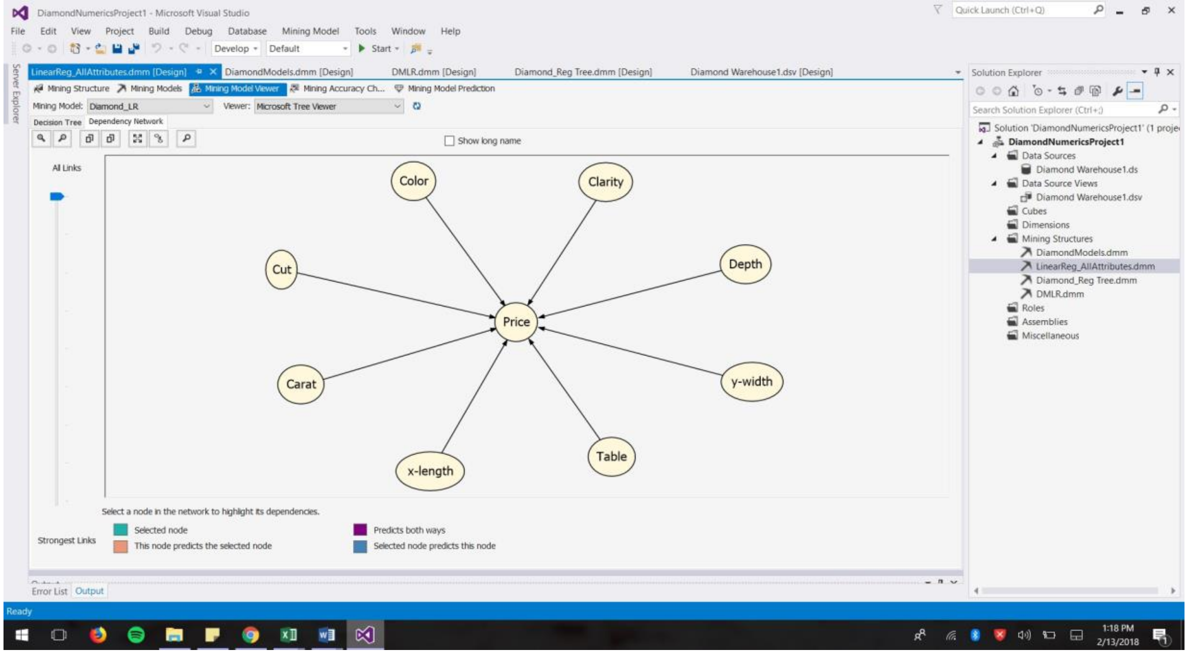
Attribute	Value	Favors 1,240.190 - 3,914.085	Favors 6,587.979 - 15,807....
y-width	6.484 - 9.083	▬	
Carat	0.478 - 0.796	▬	
Clarity	1.000 - 2.826	▬	
Color	1.000 - 3.251	▬	
z-depth	1.415 - 3.060	▬	
x-length	2.374 - 4.974	▬	
y-width	5.731 - 6.484	▬	
Clarity	2.826 - 3.980	▬	
Color	3.251 - 4.399	▬	
y-width	2.378 - 4.977		▬
x-length	4.974 - 5.728	▬	
z-depth	3.060 - 3.537	▬	
Table	58.952 - 64.099	▬	
Cut	1.000 - 3.153	▬	
Table	57.460 - 58.952	▬	
Cut	3.904 - 4.654	▬	
Depth	61.747 - 62.706	▬	
Cut	4.654 - 5.000	▬	
Cut	3.153 - 3.904	▬	
Table	55.967 - 57.460	▬	
Depth	60.787 - 61.747	▬	
Table	50.821 - 55.967	▬	

This item does not support previewing

This screenshot is the neural network (NN). Based on what we learned from the tutorials on Youtube, the NN model is helpful to determine which attributes are significant to predictable value. However, for this diamond dataset, there were too many ranges of values for each individual attribute. The price ranges are also broad. Therefore, we did not see any significant correlation between the inputs and the different ranges of price.

The third algorithm we ran was linear regression because regression is a data mining function that predicts a number. Regression task begins with a data set in which the target values are known. A regression algorithm estimates the value of the target attribute as a function of the input attributes for each case. These relationships between target and input attributes are summarized in a model, which can then be applied to a different data set in which the predicted values not known. Since we wanted to predict the diamond price and our attributes are numerical, linear regression was the best model for our data.

We used the Dependency Network function to determine the correlation between the continuous attributes and the predictive attributes.



Mining Model: Diamond_LR

Viewer: Microsoft Tree Viewer

□ Show long name

Select a node in the network to highlight its dependencies.

Strongest Links

🟩 Selected node

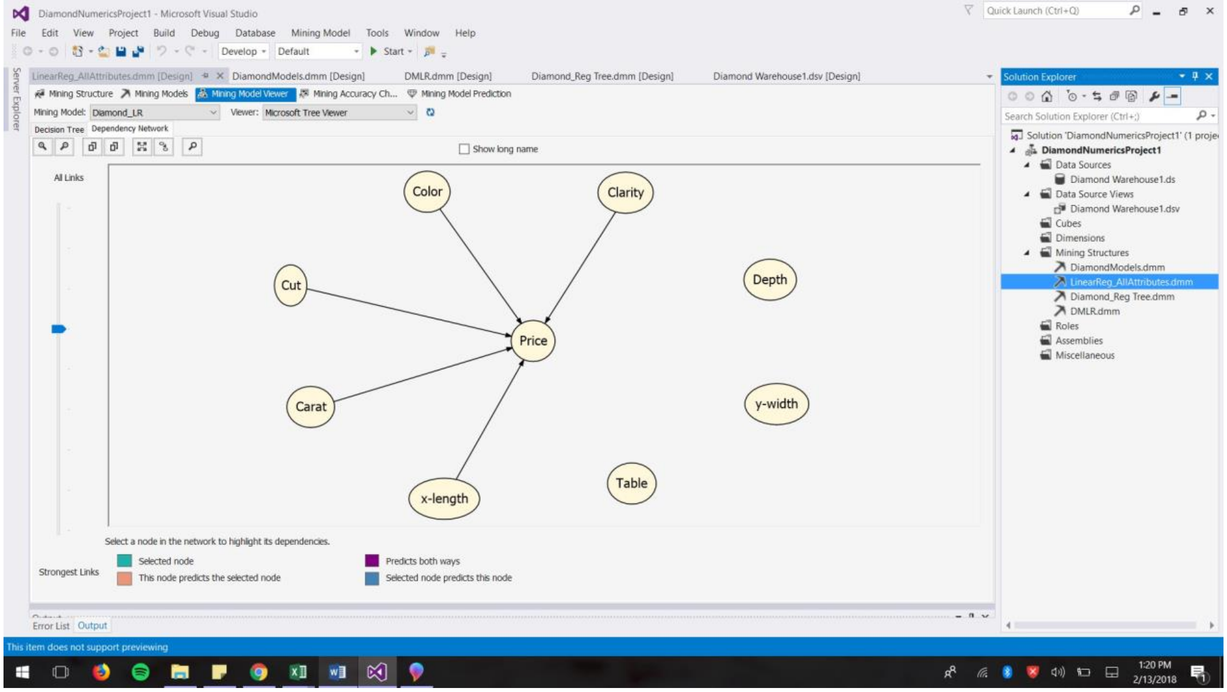
🟧 This node predicts the selected node

🟪 Predicts both ways

🟦 Selected node predicts this node

Exhibit A

As we can see in Exhibit A, Microsoft SQL Server automatically removed attribute Z-depth from the equation, which means Z is not significantly relevant to the price. When we dragged down the scale on the left side, it will show us which attribute has the stronger link or is more significantly relevant to the predictive attribute, price. (Exhibit B).



Mining Model: Diamond_LR

Viewer: Microsoft Tree Viewer

Select a node in the network to highlight its dependencies.

🟩 Selected node

🟥 This node predicts the selected node

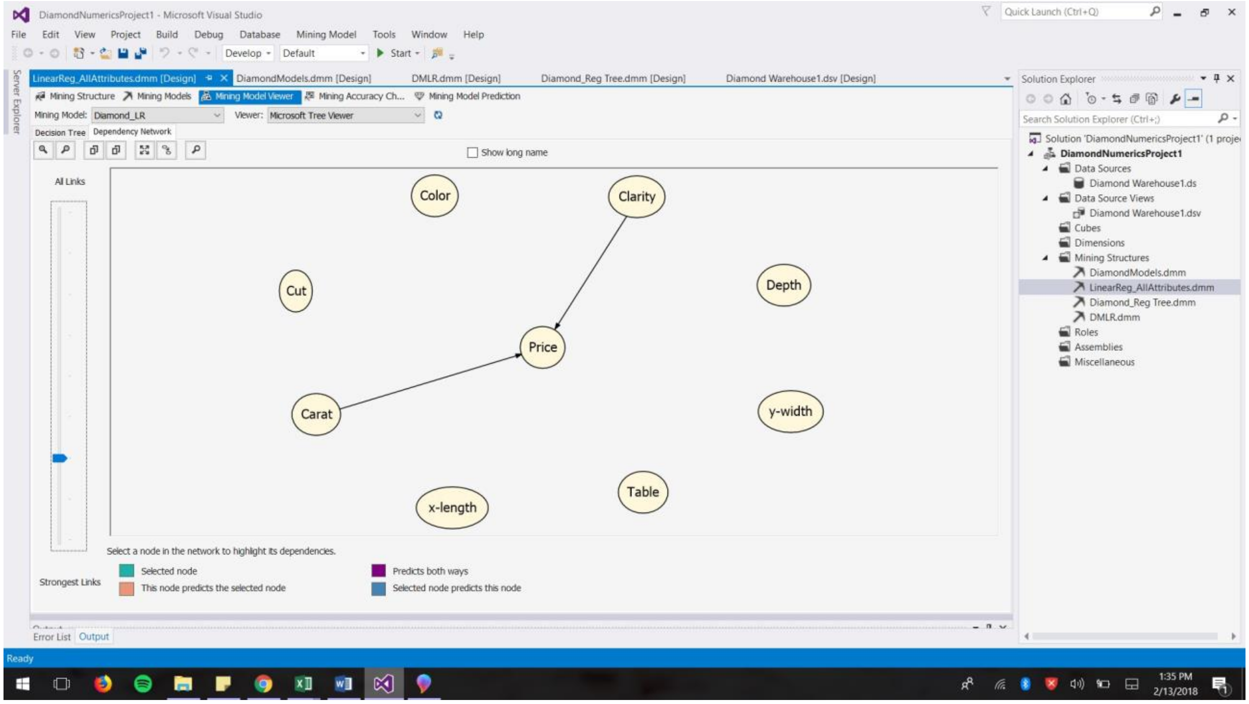
🟪 Predicts both ways

🟦 Selected node predicts this node

This item does not support previewing

Exhibit B

For the first linear regression model, we include all the attributes as our inputs for the model. Microsoft SQL Server produced a formula, not including the Z-depth. With the information from Dependency Network, we ran three more regression models. Each time we eliminated one attribute that is less significant than others are. The fourth model only included Carat and Clarity, since these two had the strongest links to price. See Exhibit C and Exhibit D for more references.



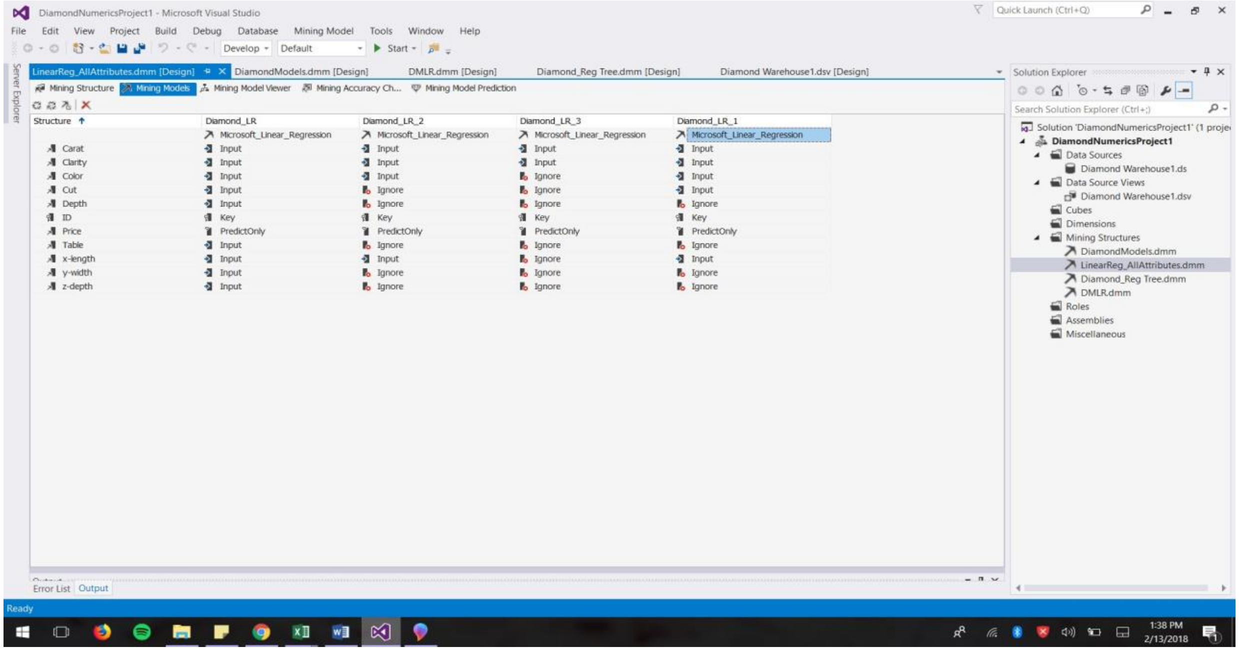
Mining Model: Diamond_LR

Viewer: Microsoft Tree Viewer

Select a node in the network to highlight its dependences.

- 🟩 Selected node
- 🟧 This node predicts the selected node
- 🟪 Predicts both ways
- 🟦 Selected node predicts this node

Exhibit C

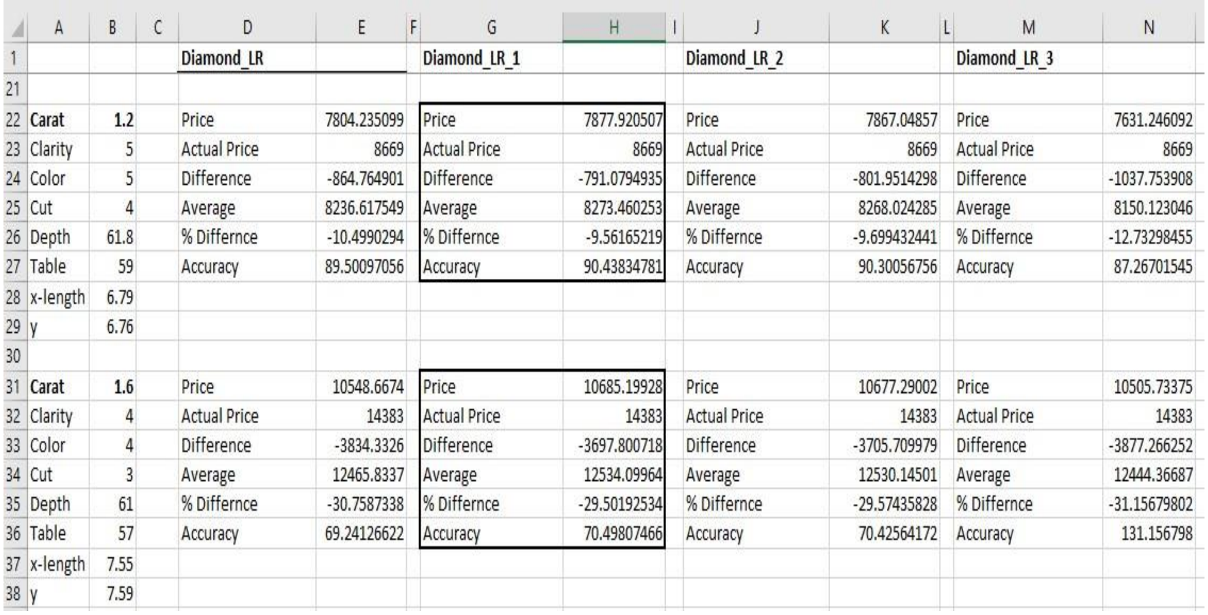


Structure ↑	Diamond_LR	Diamond_LR_2	Diamond_LR_3	Diamond_LR_1
	Microsoft_Linear_Regression	Microsoft_Linear_Regression	Microsoft_Linear_Regression	Microsoft_Linear_Regression
Carat	Input	Input	Input	Input
Clarity	Input	Input	Input	Input
Color	Input	Input	Ignore	Input
Cut	Input	Ignore	Ignore	Input
Depth	Input	Ignore	Ignore	Ignore
ID	Key	Key	Key	Key
Price	PredictOnly	PredictOnly	PredictOnly	PredictOnly
Table	Input	Ignore	Ignore	Ignore
x-length	Input	Input	Ignore	Input
y-width	Input	Ignore	Ignore	Ignore
z-depth	Input	Ignore	Ignore	Ignore

Exhibit D

Model Names	Model Formulas
Diamond_LR	Attributes Carat, Clarity, Color, Cut, Depth, Table, x-length Price = 3972.08 + 10673.837*(Carat) + 396.976*(Clarity) + 301.695*(Color) + 134.608*(Cut) – 84.747*(Depth) – 27.009*(Table) - 1104.033(x) + 195.129(y)
Diamond_ LR _1	Attributes Carat, Clarity, Color, Cut,x-length Price = 10440.272*(Carat) + 404.419*(Clarity) + 304.223*(Color) + 179.191*(Cut) – 812.536*(X) – 3393.263
Diamond_LR_2	Attributes: Carat, Clarity, Color, x-length Price = 10309.547*(Carat) + 422.715*(Clarity) + 303.575*(Color) – 772.747*(x) – 2888.908
Diamond_LR_3	Attributes: Carat, Clarity Price = 8189.855*(Carat) + 401.454*(Clarity) – 4203.851

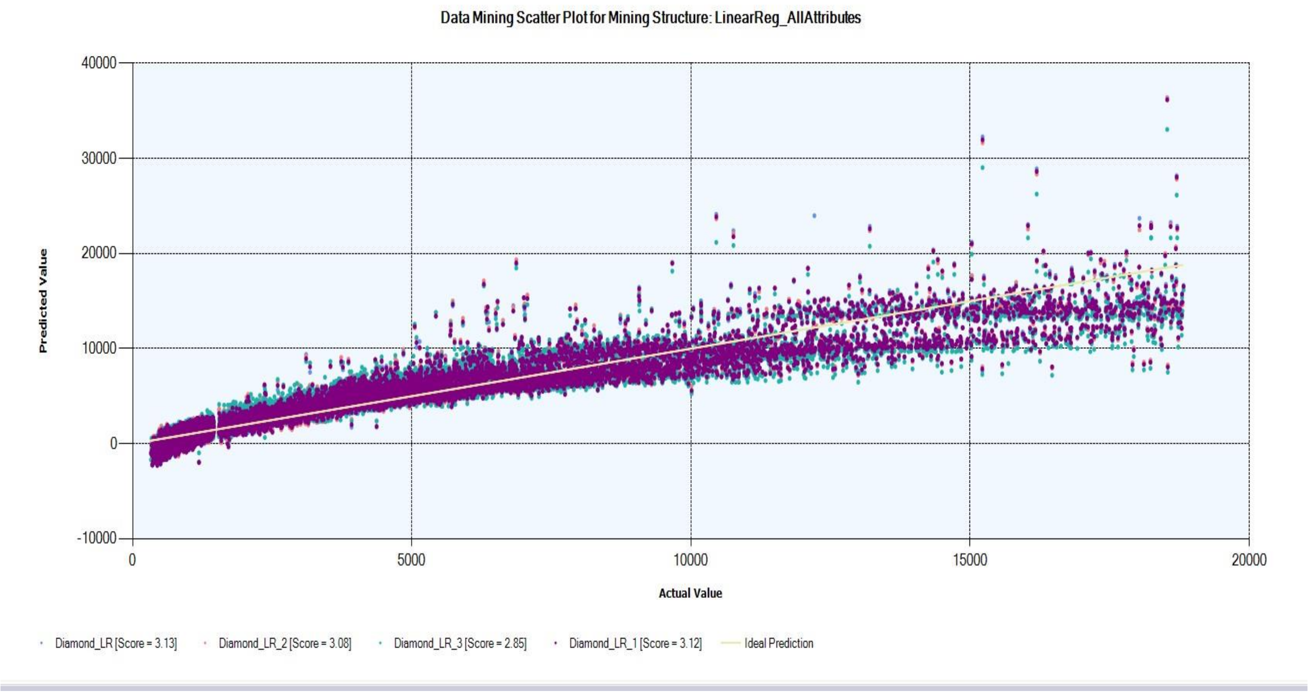
We used three different methods to verify the results. First, we used the Mining Model Prediction function in Visual Studio to predict prices for 10 different diamonds. Then we compared these predicted prices with the actual prices in order to see the percentage difference between these two values in Excel.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1				**Diamond_LR**			**Diamond_LR_1**			**Diamond_LR_2**			**Diamond_LR_3**	
21														
22	**Carat**	**1.2**		Price	7804.235099		Price	7877.920507		Price	7867.04857		Price	7631.246092
23	Clarity	5		Actual Price	8669		Actual Price	8669		Actual Price	8669		Actual Price	8669
24	Color	5		Difference	-864.764901		Difference	-791.0794935		Difference	-801.9514298		Difference	-1037.753908
25	Cut	4		Average	8236.617549		Average	8273.460253		Average	8268.024285		Average	8150.123046
26	Depth	61.8		% Differnce	-10.4990294		% Differnce	-9.56165219		% Differnce	-9.699432441		% Differnce	-12.73298455
27	Table	59		Accuracy	89.50097056		Accuracy	90.43834781		Accuracy	90.30056756		Accuracy	87.26701545
28	x-length	6.79												
29	y	6.76												
30														
31	**Carat**	**1.6**		Price	10548.6674		Price	10685.19928		Price	10677.29002		Price	10505.73375
32	Clarity	4		Actual Price	14383		Actual Price	14383		Actual Price	14383		Actual Price	14383
33	Color	4		Difference	-3834.3326		Difference	-3697.800718		Difference	-3705.709979		Difference	-3877.266252
34	Cut	3		Average	12465.8337		Average	12534.09964		Average	12530.14501		Average	12444.36687
35	Depth	61		% Differnce	-30.7587338		% Differnce	-29.50192534		% Differnce	-29.57435828		% Differnce	-31.15679802
36	Table	57		Accuracy	69.24126622		Accuracy	70.49807466		Accuracy	70.42564172		Accuracy	131.156798
37	x-length	7.55												
38	y	7.59												

By doing this, we were able to have a quick examination on which model has the highest frequency of low percentage difference. The result shows that Diamon_LR_3 has the most of low percentage difference. Model Diamond_LR_1 comes in the second place.

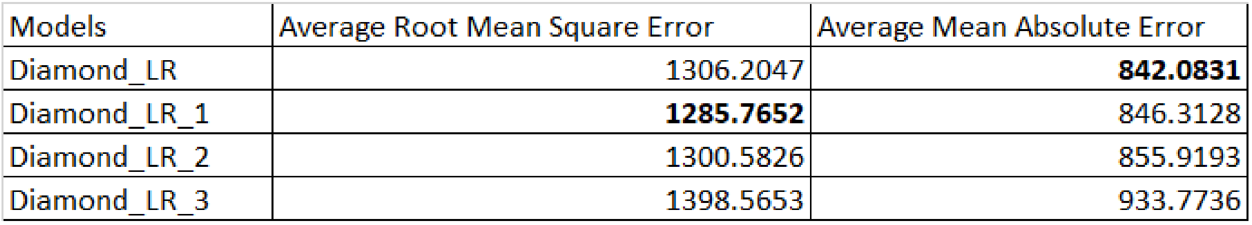
Secondly, we used the Lift Chart function to see the visual fit of these models. This function produced a scatter plot graph with the y-axis representing Predicted Value and x-axis representing Actual Value. It also gave us a score for each model. The higher the score is the better fit the model will be. Based on this Lift Chart, Diamond_LR and Diamond_LR_2 have the highest scores, and therefore, are better fit to the testing dataset.



Thirdly, we used Cross Validation to verify the accuracy rate of these models. The parameters are presented below:

- Training data set = 70%
- Testing data set = 30%
- Max cases = 40,0000
- Fold Count = 10
- Target Attribute = Price

The results include values of Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Log Score for 10 different partitions. We decided to look at the average values of RMSE and MAE to compare the accuracy between 4 linear regression models. The results are shown in the table below.



Models	Average Root Mean Square Error	Average Mean Absolute Error
Diamond_LR	1306.2047	**842.0831**
Diamond_LR_1	**1285.7652**	846.3128
Diamond_LR_2	1300.5826	855.9193
Diamond_LR_3	1398.5653	933.7736

RMSE is the square root of the mean error for all partition cases, divided by the number of cases in the partition, excluding rows that have missing values for the target attribute. MAE is the average error when predicted values are compared to actual values, calculated as the mean of the absolute sum of errors. A smaller RMSE (or MAE) means predictions were more accurate. Diamond_LR has the smallest value of Average MAE, while Diamond_LR_1 has the lowest value of Average RMSE. The MAE is very similar to the RMSE, but is less sensitive to large errors. Therefore, we think RMSE is a better indication for the accuracy of a model. For this Cross Validation method, Diamond_LR_1 has the highest average RMSE.

Overall, the first method confirms that Diamond_LR_1 model predicts a price that has low percentage difference compared to the actual price. The Lift Chart method confirms that Diamond_LR_1 is a good fit to the testing data. This model also scores the highest in RMSE, meaning it is more accurate than other models. We come to the conclusion that linear regression model Diamond_LR_1 is best one to predict the diamond price.

LIMITATIONS OF MICROSOFT SQL SERVER ANALYSIS SERVICES

1. The R Square value is only available in the paid version of the tool, which made it difficult for us to compare our different regression model.
2. SVM not available in the unpaid version of MS SQL server analysis, and so we were not able to use that our project.
3. Interpreting the results was not easy (no p-value). Unlike in RapidMiner, where p-value is shown in the analysis, the SQL server does not do so. However, at the same time SQL Server has more ways to showing and interpreting the result. You can drill down further to understand the results and the factors that predict the outcome.
4. Cannot use the classification matrix to evaluate the model since we are not answering a yes/no question.

Overall, MS SQL Server is not the best tool in its current form with limited feature. It is also not very user-friendly in terms of using it or the visualization of the results. The Azure version is said to be much advanced in features and easier to use.

THE 3W QUESTIONS

WHAT WENT WELL?

1. Getting our dataset imported into SQL server and getting it to work was easier than in Orange.
2. Getting our regression equation to predict the price that was pretty close to the actual prices.
3. We have a fair understanding of SQL Server Analysis tool now, which was the goal when we started this project, to tie it in with our KMBI class where we learnt to build the multidimensional cube.

WHAT WENT WRONG?

1. We were limited in our analysis because of the dataset and the tool we chose.
2. It was difficult to understand the tool and the interpret the results. SSAS is not the most user-friendly and visually appealing analysis tool, hence we had to go through a lot of blogs and Youtube tutorials to understand how to use it and make sense of the results.
3. We would have liked to explore more classification algorithms like Decision trees, Naive Bayes, etc. It was due to the dataset we chose which all numeric was.

WHAT WOULD BE DO DIFFERENTLY?

1. Choose a dataset with nominal values as well where more classification models could be used.
2. If we use the same dataset, we would choose a different tool that is easier and more informative with linear regression model, in terms of statistical implications (i.e R-squared, p-value). We would look into more open-source tools such as Weka, SPSS, KNMIE to see if these tools work better for linear regression model
3. If we have more time, we will explore more with the SQL query in the Visual Studio. We saw a couple a blog posts about how to calculate R-square value by writing a SQL query. However, the dataset demonstrated in these examples are different, and we do not have the knowledge of SQL query with this tool. We failed to explore the tool in this aspect.