Homework 1: The sales of Toyota Corolla cars

Dataset: ToyotaCorolla.xlsx

The file ToyotaCorolla.xls contains data on used cars (Toyota Corollas) on sale during late summer of 2004 in The Netherlands. It has 1436 records containing details on 38 attributes, including Price, Age, Kilometers, HP, and other specifications. The goal will be to predict the price of a used Toyota Corolla based on its specifications.

1. Identify at least 2 categorical variables. (10 points)

2. Convert the categorical variables in this dataset into dummy binaries, and explain in words, for one record, the values in the derived binary dummies. (15 points)

3. Produce scatterplot matrix plots. Please include your screenshots of the scatterplot matrix plots (5 points). Comment on the relationships among variables you found interesting (5 points).

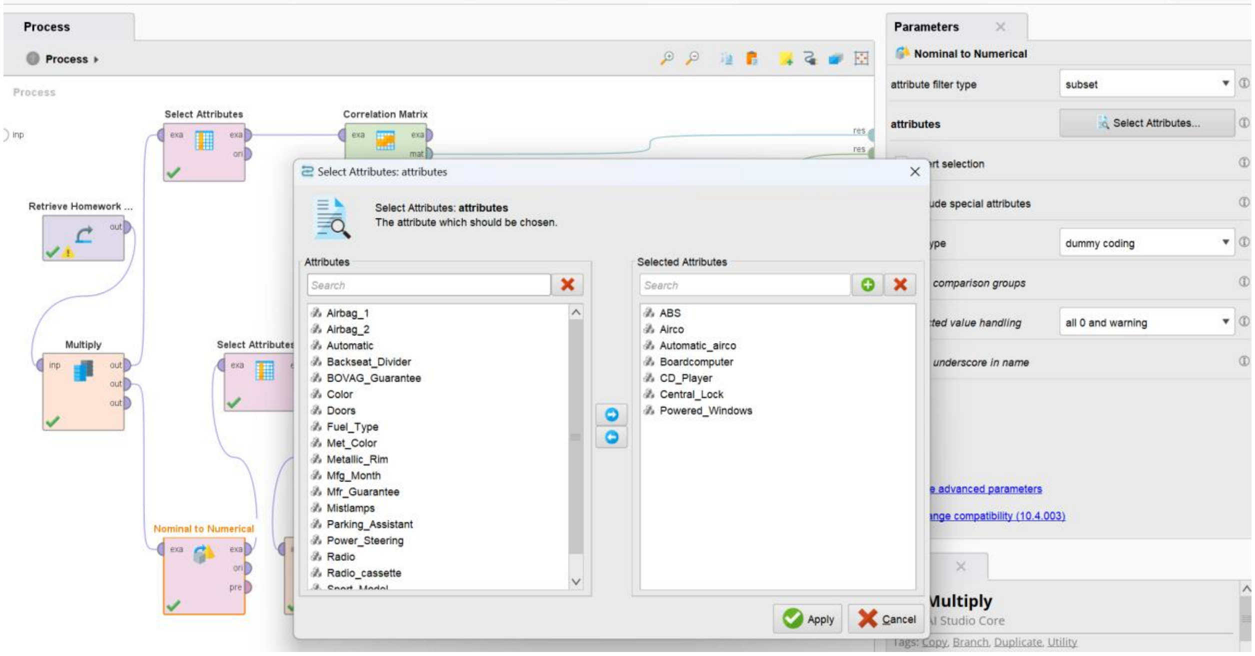
4. The goal will be to predict the price of a used Toyota Corolla based on its specifications. Please prepare a report including screenshots and description of the steps you took to train your model.
   Please **indicate if/what data transformation you did, which attributes you decided to remove (and why)**, **how you decided to validate your model and what was the result. Include a very brief interpretation of the results and potential ideas on how the model could be improved**. Please include your screenshots of relevant processes. (15 points)

Answers:

1. - **Fuel_Type** is a categorical variable with three possible outcomes **Petrol, Diesel, CNG**.
- **Color** is another categorical variable with ten possible outcomes such as **Blue, Red, Grey, Silver, Black, etc**.

---

2. To convert the categorical variables in the dataset into dummy binaries, I had to use the **Nominal to Numerical** operator, then choose the attributes that I wanted to convert into **dummy binaries**. Although attributes like Color, Fuel_Type and Mfg_Month exist, I only chose the categorical variables that have an acceptable **correlation** with the **label** attribute. This led to me choosing only the variables in the image below.



Select Attributes: attributes

Select Attributes: **attributes**
The attribute which should be chosen.

Attributes

Search

- Airbag_1
- Airbag_2
- Automatic
- Backseat_Divider
- BOVAG_Guarantee
- Color
- Doors
- Fuel_Type
- Met_Color
- Metallic_Rim
- Mfg_Month
- Mfr_Guarantee
- Mistlamps
- Parking_Assistant
- Power_Steering
- Radio
- Radio_cassette
- Sport_Model

Selected Attributes

Search

- ABS
- Airco
- Automatic_airco
- Boardcomputer
- CD_Player
- Central_Lock
- Powered_Windows

Apply   Cancel

Parameters

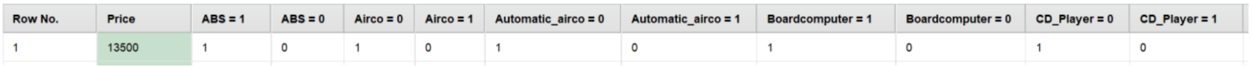
Nominal to Numerical

attribute filter type	subset
attributes	Select Attributes...
...rt selection	
...ude special attributes	
...ype	dummy coding
comparison groups	
...ted value handling	all 0 and warning
underscore in name	

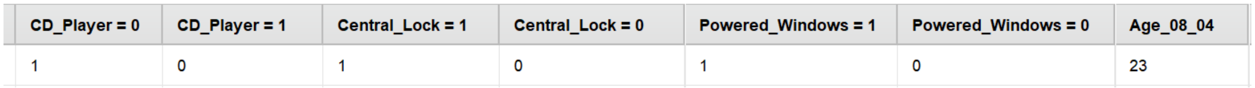
...e advanced parameters
...ange compatibility (10.4.003)

Multiply
...I Studio Core
Tags: Copy, Branch, Duplicate, Utility

This method transforms each categorical variable into multiple binary variables, with each representing a possible category. Each new binary feature then reflects the **presence** (1) or **absence** (0) of the specific category as shown in the image below.



Row No.	Price	ABS = 1	ABS = 0	Airco = 0	Airco = 1	Automatic_airco = 0	Automatic_airco = 1	Boardcomputer = 1	Boardcomputer = 0	CD_Player = 0	CD_Player = 1
1	13500	1	0	1	0	1	0	1	0	1	0



CD_Player = 0	CD_Player = 1	Central_Lock = 1	Central_Lock = 0	Powered_Windows = 1	Powered_Windows = 0	Age_08_04
1	0	1	0	1	0	23

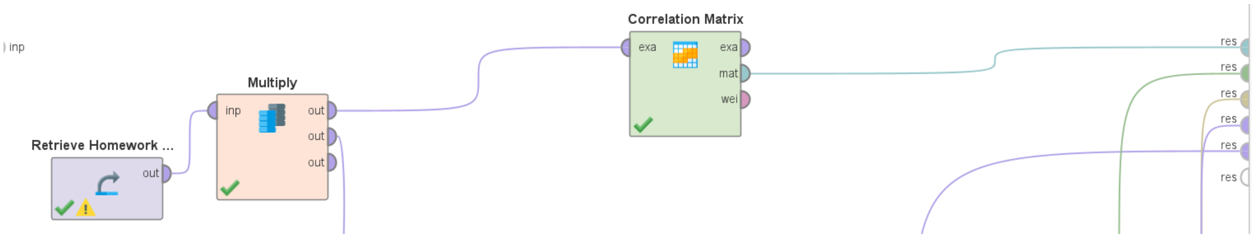
We can see for the first record the following conversions have occurred to make the variable into binary dummies so that the Linear Regression operator can be applied on it.

- ABS: "ABS = 1" means the car has an Anti-lock Braking System, and "ABS = 0" means it doesn't. In the first row, "ABS = 1" shows the car **has ABS**.
- Airco: "Airco = 1" indicates the car has air conditioning, and "Airco = 0" means it doesn't. Here, "Airco = 0" shows **no air conditioning**.
- Automatic_airco: " Automatic_airco = 1" means the car has automatic air conditioning, and " Automatic_airco = 0" means it doesn't. Here, " Automatic_airco = 1", so the car has **no automatic air conditioning**.
- Boardcomputer: "Boardcomputer = 1" means the car **has a board computer**, which is true in this case.
- CD_Player: "CD_Player = 1" means the car has a CD player, and "CD_Player = 0" means it doesn't. This car **lacks a CD player**, as shown by "CD_Player = 0".
- Central_Lock: “Central_Lock = 1” means the car has central locking, and “Central_Lock = 0” means it lacks it. With “Central_Lock = 1”, this car **has central locking**.

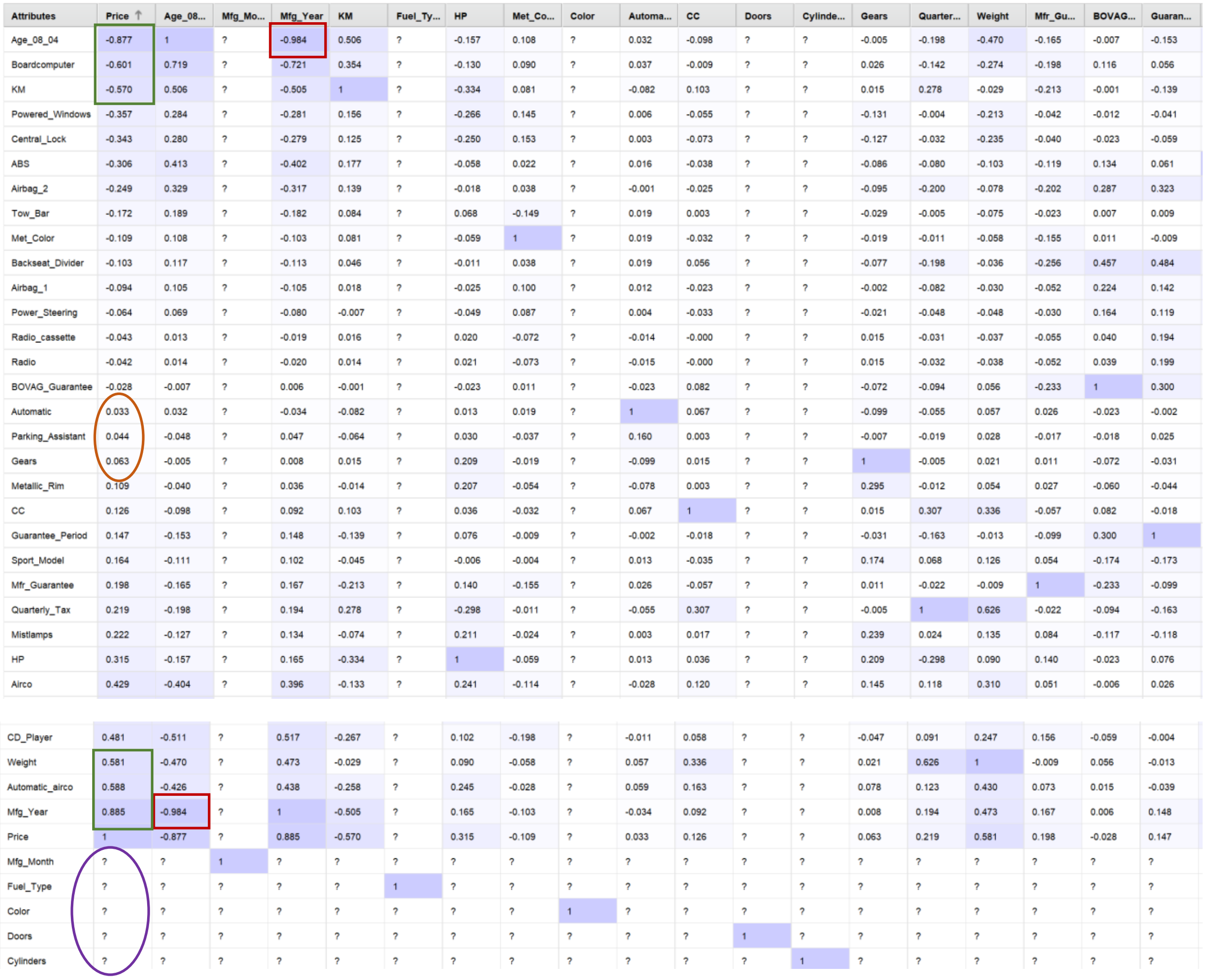
- Powered_Windows: “Powered_Windows = 1” indicates that the car has powered windows and “Powered_Windows = 0" indicates that it does not. Here, “Powered_Windows = 1” means the car **has powered windows**.

---

3. To generate the scatterplot matrix plots we need to use the **Correlation Matrix** operator.



The results for it are as below, having 38 attributes make the matrix large.



Attributes	Price ↑	Age_08...	Mfg_Mo...	Mfg_Year	KM	Fuel_Ty...	HP	Met_Co...	Color	Automa...	CC	Doors	Cylinde...	Gears	Quarter...	Weight	Mfr_Gu...	BOVAG...	Guaran...
Age_08_04	-0.877	1	?	-0.984	0.506	?	-0.157	0.108	?	0.032	-0.098	?	?	-0.005	-0.198	-0.470	-0.165	-0.007	-0.153
Boardcomputer	-0.601	0.719	?	-0.721	0.354	?	-0.130	0.090	?	0.037	-0.009	?	?	0.026	-0.142	-0.274	-0.198	0.116	0.056
KM	-0.570	0.506	?	-0.505	1	?	-0.334	0.081	?	-0.082	0.103	?	?	0.015	0.278	-0.029	-0.213	-0.001	-0.139
Powered_Windows	-0.357	0.284	?	-0.281	0.156	?	-0.266	0.145	?	0.006	-0.055	?	?	-0.131	-0.004	-0.213	-0.042	-0.012	-0.041
Central_Lock	-0.343	0.280	?	-0.279	0.125	?	-0.250	0.153	?	0.003	-0.073	?	?	-0.127	-0.032	-0.235	-0.040	-0.023	-0.059
ABS	-0.306	0.413	?	-0.402	0.177	?	-0.058	0.022	?	0.016	-0.038	?	?	-0.086	-0.080	-0.103	-0.119	0.134	0.061
Airbag_2	-0.249	0.329	?	-0.317	0.139	?	-0.018	0.038	?	-0.001	-0.025	?	?	-0.095	-0.200	-0.078	-0.202	0.287	0.323
Tow_Bar	-0.172	0.189	?	-0.182	0.084	?	0.068	-0.149	?	0.019	0.003	?	?	-0.029	-0.005	-0.075	-0.023	0.007	0.009
Met_Color	-0.109	0.108	?	-0.103	0.081	?	-0.059	1	?	0.019	-0.032	?	?	-0.019	-0.011	-0.058	-0.155	0.011	-0.009
Backseat_Divider	-0.103	0.117	?	-0.113	0.046	?	-0.011	0.038	?	0.019	0.056	?	?	-0.077	-0.198	-0.036	-0.256	0.457	0.484
Airbag_1	-0.094	0.105	?	-0.105	0.018	?	-0.025	0.100	?	0.012	-0.023	?	?	-0.002	-0.082	-0.030	-0.052	0.224	0.142
Power_Steering	-0.064	0.069	?	-0.080	-0.007	?	-0.049	0.087	?	0.004	-0.033	?	?	-0.021	-0.048	-0.048	-0.030	0.164	0.119
Radio_cassette	-0.043	0.013	?	-0.019	0.016	?	0.020	-0.072	?	-0.014	-0.000	?	?	0.015	-0.031	-0.037	-0.055	0.040	0.194
Radio	-0.042	0.014	?	-0.020	0.014	?	0.021	-0.073	?	-0.015	-0.000	?	?	0.015	-0.032	-0.038	-0.052	0.039	0.199
BOVAG_Guarantee	-0.028	-0.007	?	0.006	-0.001	?	-0.023	0.011	?	-0.023	0.082	?	?	-0.072	-0.094	0.056	-0.233	1	0.300
Automatic	0.033	0.032	?	-0.034	-0.082	?	0.013	0.019	?	1	0.067	?	?	-0.099	-0.055	0.057	0.026	-0.023	-0.002
Parking_Assistant	0.044	-0.048	?	0.047	-0.064	?	0.030	-0.037	?	0.160	0.003	?	?	-0.007	-0.019	0.028	-0.017	-0.018	0.025
Gears	0.063	-0.005	?	0.008	0.015	?	0.209	-0.019	?	-0.099	0.015	?	?	1	-0.005	0.021	0.011	-0.072	-0.031
Metallic_Rim	0.109	-0.040	?	0.036	-0.014	?	0.207	-0.054	?	-0.078	0.003	?	?	0.295	-0.012	0.054	0.027	-0.060	-0.044
CC	0.126	-0.098	?	0.092	0.103	?	0.036	-0.032	?	0.067	1	?	?	0.015	0.307	0.336	-0.057	0.082	-0.018
Guarantee_Period	0.147	-0.153	?	0.148	-0.139	?	0.076	-0.009	?	-0.002	-0.018	?	?	-0.031	-0.163	-0.013	-0.099	0.300	1
Sport_Model	0.164	-0.111	?	0.102	-0.045	?	-0.006	-0.004	?	0.013	-0.035	?	?	0.174	0.068	0.126	0.054	-0.174	-0.173
Mfr_Guarantee	0.198	-0.165	?	0.167	-0.213	?	0.140	-0.155	?	0.026	-0.057	?	?	0.011	-0.022	-0.009	1	-0.233	-0.099
Quarterly_Tax	0.219	-0.198	?	0.194	0.278	?	-0.298	-0.011	?	-0.055	0.307	?	?	-0.005	1	0.626	-0.022	-0.094	-0.163
Mistlamps	0.222	-0.127	?	0.134	-0.074	?	0.211	-0.024	?	0.003	0.017	?	?	0.239	0.024	0.135	0.084	-0.117	-0.118
HP	0.315	-0.157	?	0.165	-0.334	?	1	-0.059	?	0.013	0.036	?	?	0.209	-0.298	0.090	0.140	-0.023	0.076
Airco	0.429	-0.404	?	0.396	-0.133	?	0.241	-0.114	?	-0.028	0.120	?	?	0.145	0.118	0.310	0.051	-0.006	0.026
CD_Player	0.481	-0.511	?	0.517	-0.267	?	0.102	-0.198	?	-0.011	0.058	?	?	-0.047	0.091	0.247	0.156	-0.059	-0.004
Weight	0.581	-0.470	?	0.473	-0.029	?	0.090	-0.058	?	0.057	0.336	?	?	0.021	0.626	1	-0.009	0.056	-0.013
Automatic_airco	0.588	-0.426	?	0.438	-0.258	?	0.245	-0.028	?	0.059	0.163	?	?	0.078	0.123	0.430	0.073	0.015	-0.039
Mfg_Year	0.885	-0.984	?	1	-0.505	?	0.165	-0.103	?	-0.034	0.092	?	?	0.008	0.194	0.473	0.167	0.006	0.148
Price	1	-0.877	?	0.885	-0.570	?	0.315	-0.109	?	0.033	0.126	?	?	0.063	0.219	0.581	0.198	-0.028	0.147
Mfg_Month	?	?	1	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Fuel_Type	?	?	?	?	?	1	?	?	?	?	?	?	?	?	?	?	?	?	?
Color	?	?	?	?	?	?	?	?	1	?	?	?	?	?	?	?	?	?	?
Doors	?	?	?	?	?	?	?	?	?	?	?	1	?	?	?	?	?	?	?
Cylinders	?	?	?	?	?	?	?	?	?	?	?	?	1	?	?	?	?	?	?

The label (special attribute/dependent variable) is **Price**.

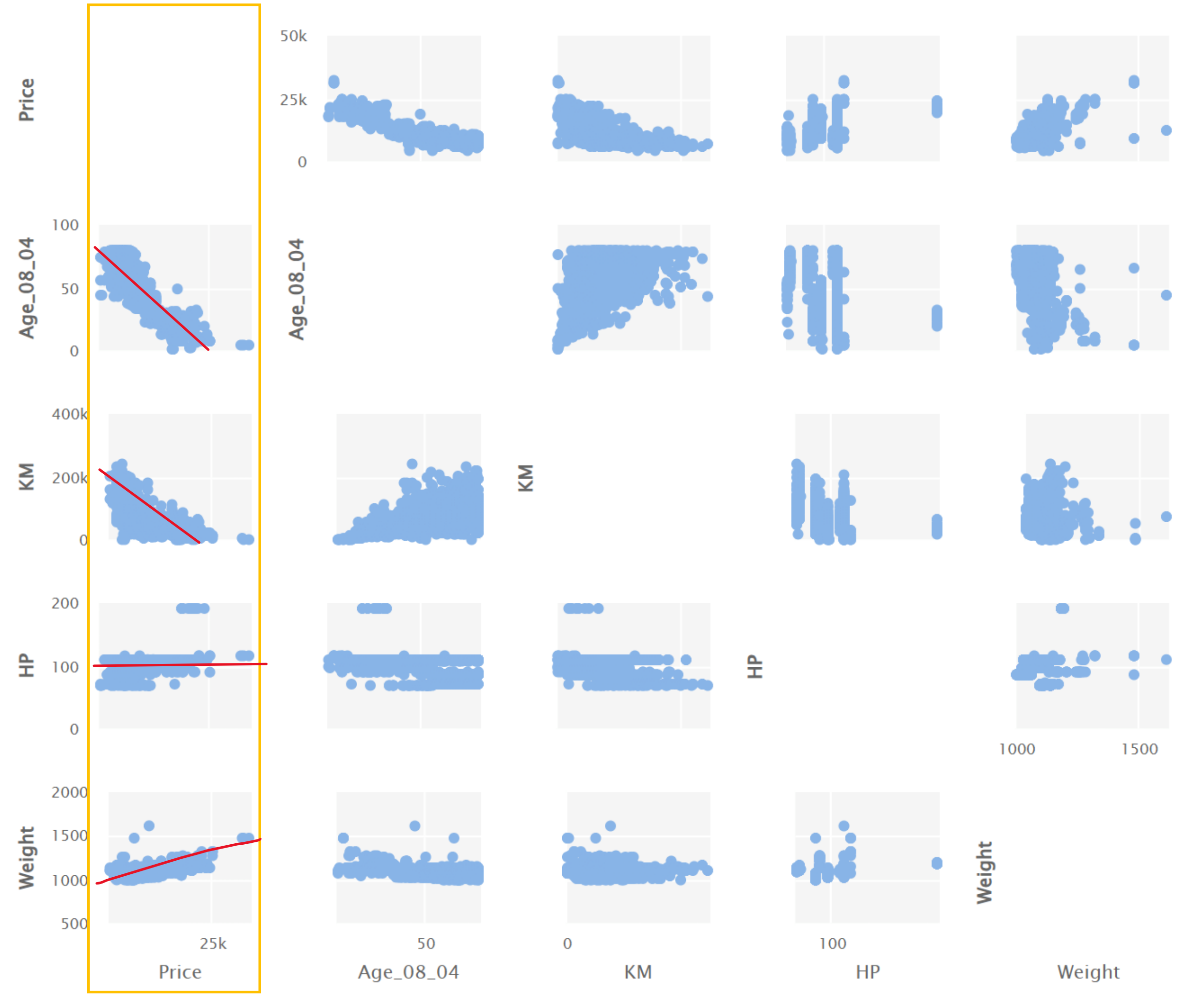
From the correlation matrix, outlined in green, we can see that attributes like Age_08_04 (-0.877), Boardcomputer (-0.601), and KM (-0.570), have good **negative correlation** with the special attribute, and these might be **good predictors** for Price.

At the same time attributes like Weight (-0.581), Automatic_airco (-0.588) and Mfg_Year (-0.885) show **good positive correlation** with the special attribute, which tells us that these attributes might be **good predictors** for Price too.

We also can see few attributes circled in orange like Automatic (0.033), Parking_Assistant (0.044) and Gears (0.063) have **poor correlation**, these attributes might be **bad predictors** for Price.

Along with the above observations, we noticed that Age_08_04 and Mfg_Year are **highly negatively correlated** to each other, to avoid **multicollinearity** we can omit one of these attributes.

The attributes circled in purple were not able to provide a correlation value as they were **polynomial categorical variables**, I did convert them into binary dummies and their **correlation was poor** as well.

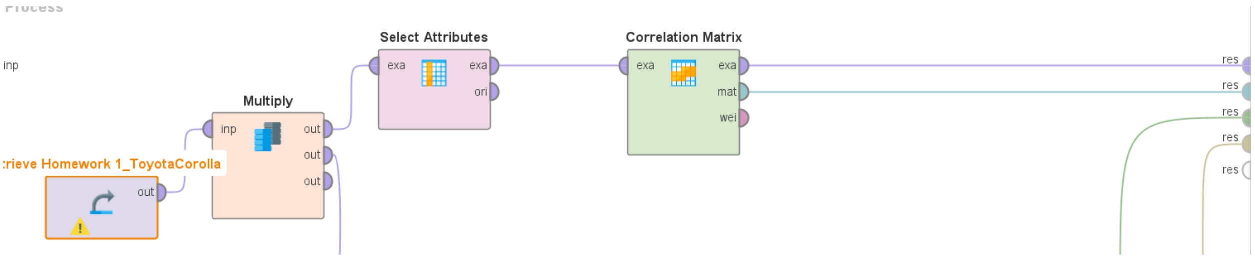


The above shows the scatterplot matrix of the attributes after we filtered out the poorly correlated attributes using the Select Attributes operator.

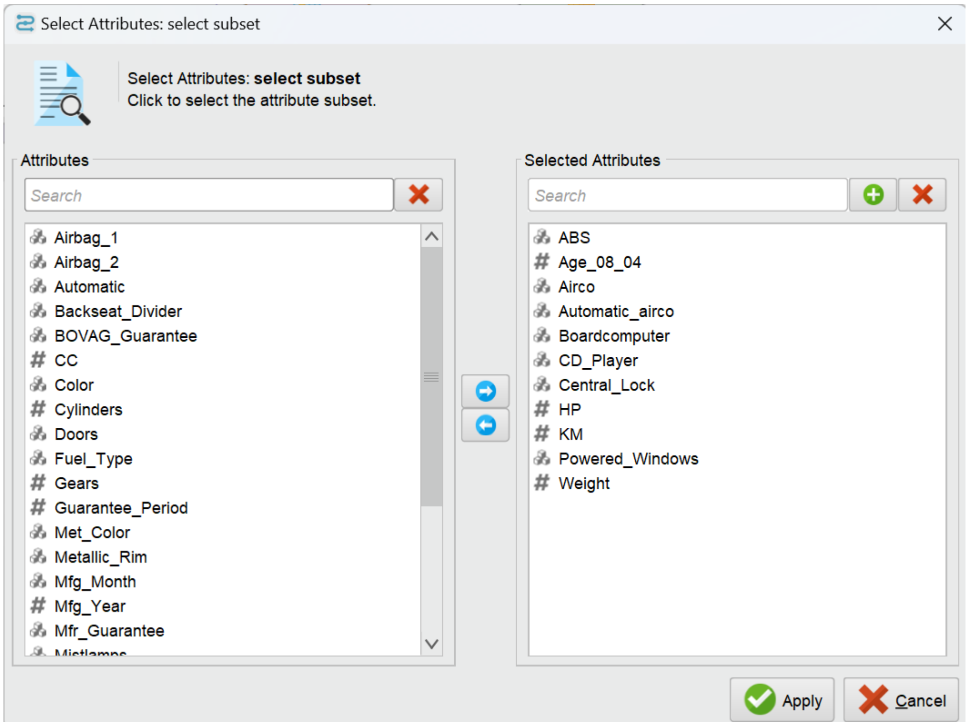
We can also observe that the special attribute has a negative correlation with Age_08_04 and KM, meanwhile has a positive correlation with weight and very slight to no correlation with HP.

4. Firstly, I started off by importing the data using the "Import Data" wizard, where I chose to omit columns such as Id and Model as they wouldn't help with the task of predicting the Price. I followed this by choosing the label as Price and then changing attributes that were categorized as integer when they were categorical data.

I then proceeded to use the **Multiply** operator as I needed to look at the correlation matrix to choose the attributes which correlate the most.



After examining the correlation matrix and choosing the attributes with above 0.3 and below -0.3, so all attributes with correlation between **-0.3 and +0.3 were rejected**. I also omitted the attributes **which had high correlation between them** to avoid multicollinearity. This left me with only 11 attributes of the 38 as shown below.



Select Attributes: select subset

Select Attributes: **select subset**
Click to select the attribute subset.

Attributes

Search

- Airbag_1
- Airbag_2
- Automatic
- Backseat_Divider
- BOVAG_Guarantee
- # CC
- Color
- # Cylinders
- Doors
- Fuel_Type
- # Gears
- # Guarantee_Period
- Met_Color
- Metallic_Rim
- Mfg_Month
- # Mfg_Year
- Mfr_Guarantee
- Mistlamps

Selected Attributes

Search

- ABS
- # Age_08_04
- Airco
- Automatic_airco
- Boardcomputer
- CD_Player
- Central_Lock
- # HP
- # KM
- Powered_Windows
- # Weight

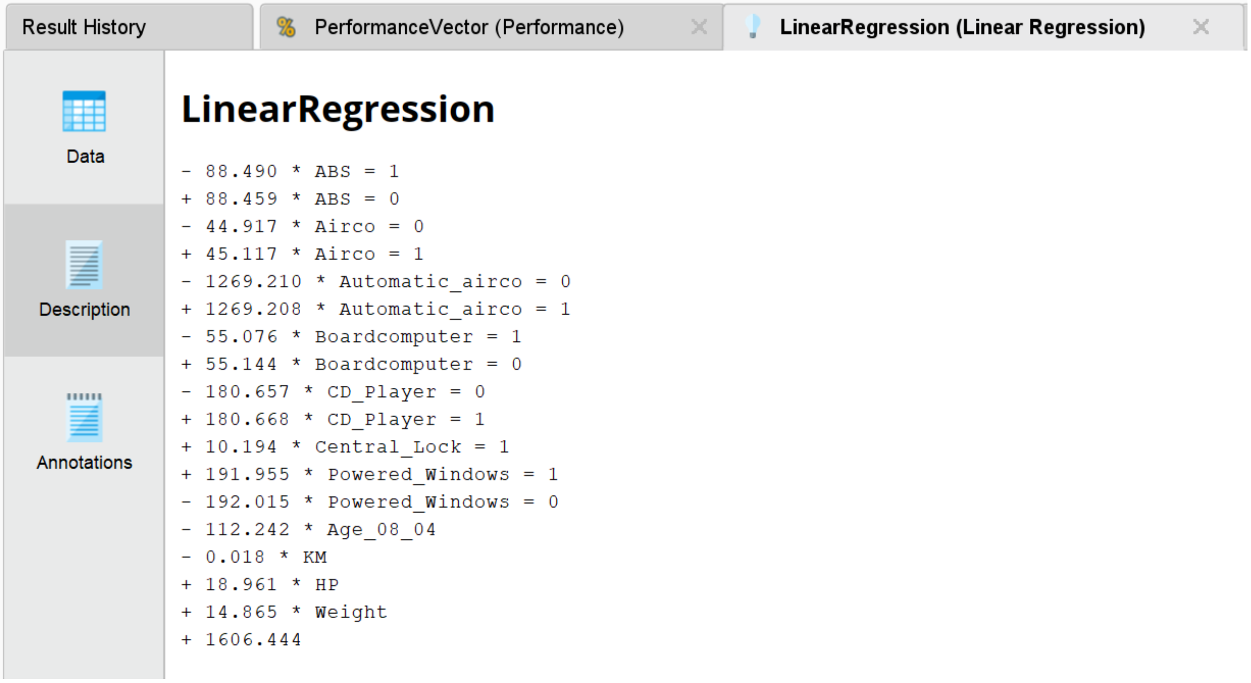
We can see that few of the attributes are categorical variables, and to use linear regression, these variables must be converted into numerical values, for this I used the **Nominal to Numerical** operator, where I converted all the categorical variables into dummy binaries. Since I had already pinpointed attributes with the most correlation earlier, I used **Select Attributes** again to filter out all the poorly correlated attributes.

To predict the Price using linear regression I had to use operators such as **Split Data**, **Linear Regression** and **Apply Model**. To measure it's performance I used the **Performance (Regression)** operator.

Split Data is used to split the data into **training** and **validation** datasets, this is done at a **70:30** split ratio.

To measure the performance of the model we use **root mean squared error** and **squared correlation**.

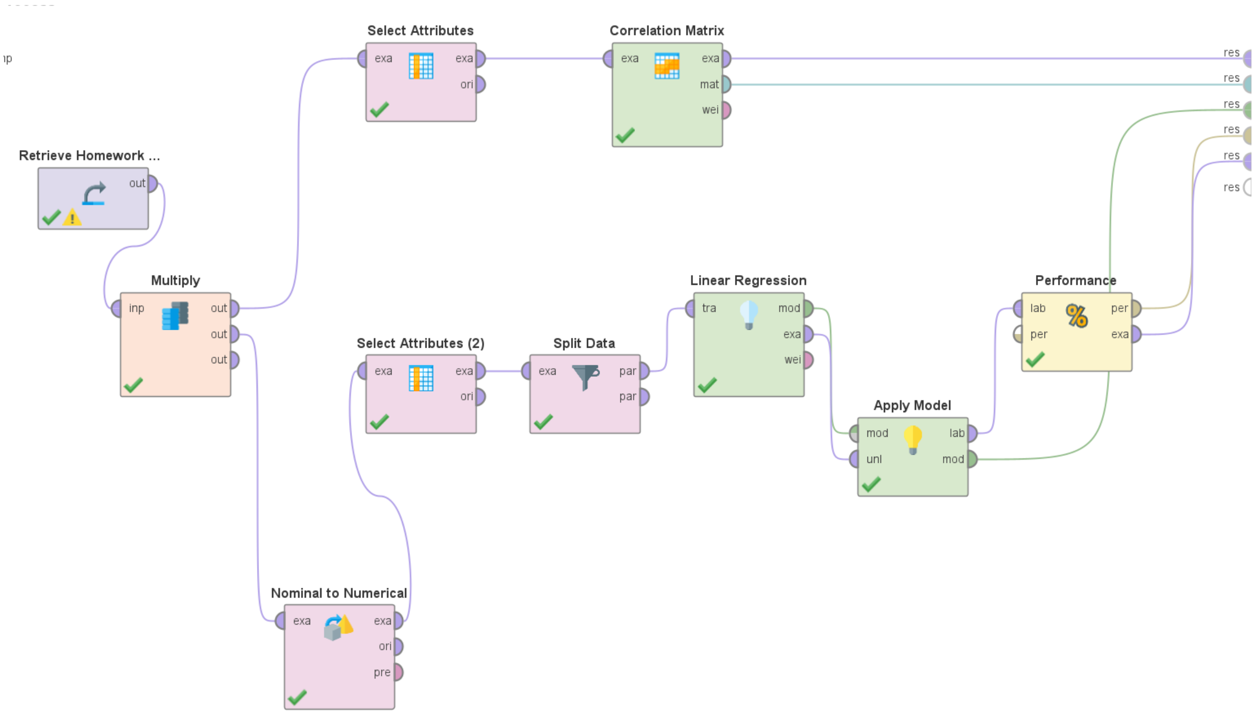
Our linear regression model is as follows which can be mathematically represented as
Y(Price) = 1606.444 + 14.865×Weight + 18.961×HP − 0.018×KM − …. + 88.459×ABS_0 − 88.490×ABS_1

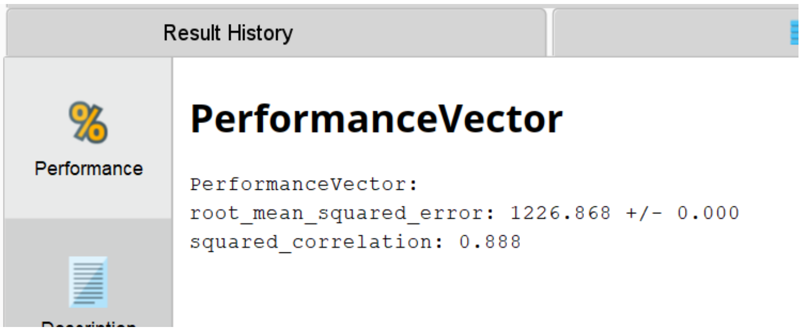

Result History
PerformanceVector (Performance)
LinearRegression (Linear Regression)
Data
Description
Annotations

LinearRegression

```
- 88.490 * ABS = 1
+ 88.459 * ABS = 0
- 44.917 * Airco = 0
+ 45.117 * Airco = 1
- 1269.210 * Automatic_airco = 0
+ 1269.208 * Automatic_airco = 1
- 55.076 * Boardcomputer = 1
+ 55.144 * Boardcomputer = 0
- 180.657 * CD_Player = 0
+ 180.668 * CD_Player = 1
+ 10.194 * Central_Lock = 1
+ 191.955 * Powered_Windows = 1
- 192.015 * Powered_Windows = 0
- 112.242 * Age_08_04
- 0.018 * KM
+ 18.961 * HP
+ 14.865 * Weight
+ 1606.444
```

The first model is generated using the operators below which are based off the **training dataset** (the first split).



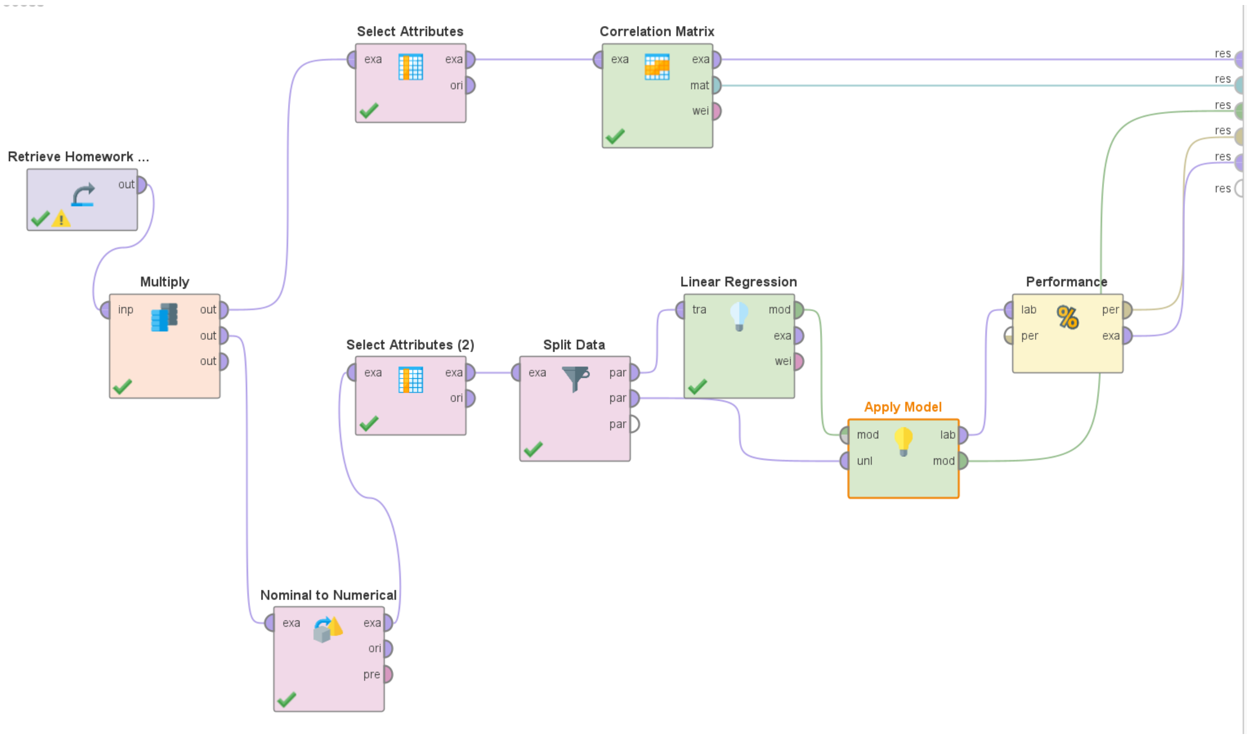


PerformanceVector

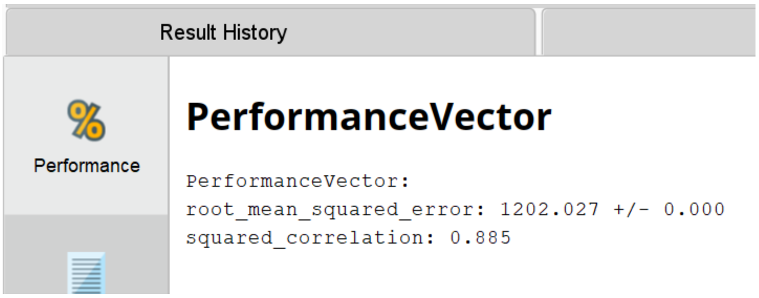
```
PerformanceVector:
root_mean_squared_error: 1226.868 +/- 0.000
squared_correlation: 0.888
```

The RMSE is **1226.868** when we apply the model on the training dataset.
R squared is **0.888**, which means that 88.8% of the **variance** in the label (Price) are explained by the attributes using the model on the training dataset.

The following operators are used to generate a model based on the **validation dataset** (the second split).



For the validation dataset, there are a few observations to be made



PerformanceVector

```
PerformanceVector:
root_mean_squared_error: 1202.027 +/- 0.000
squared_correlation: 0.885
```

The RMSE is **1202.027** when the model is tested on the validation dataset.
R square is **0.885**, which indicates that **88.5%** of the **variance** in the label (Price) are explained by the attributes using the model on the validation dataset.

To validate the model, we can compare the results for the training dataset to the validation dataset.

RMSE(T) = 1226.868 **>** RMSE(V) = 1202.027

Generally, we would prefer the RMSE to be small, even though values being around 1200 seems to be a poor RMSE, considering the label is in 1000s, this is a **good RMSE** for our data.

If the RMSE for Training (T) and Validation (V) are close, that means that the model developed is good.
If the RMSE(V) < RMSE(T), that means the model does a really good job predicting new data.
If the RMSE(V) is significantly larger than RMSE(T), that means that the model does a poor job (low predictive value) in predicting new data and it is likely that we over fit the data.

Considering the above reasonings and our RMSE(V) being **less than** RMSE(T) and the values being close, we can conclude that we have developed a **good model** with a **high predictive value**.

To improve our model further, we can start by **normalizing/scaling** our features accordingly, this would also give us low RMSE values. We can also experiment with **Random Forest** and **Neural Networks** instead of Linear Regression to see if that helps use develop a better model. And using **k-fold cross validation** rather than splitting the data into just one training and validation set can give us a better estimate of how the model performs on unseen data.
