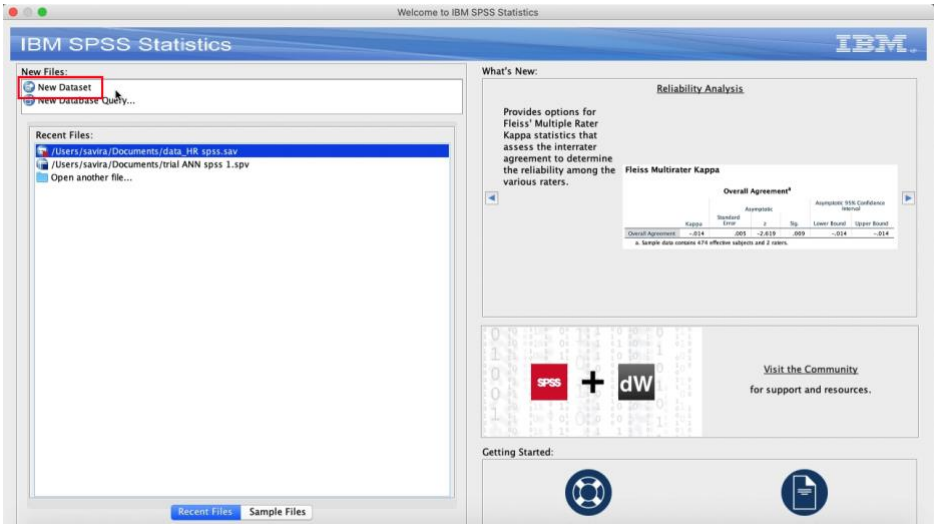
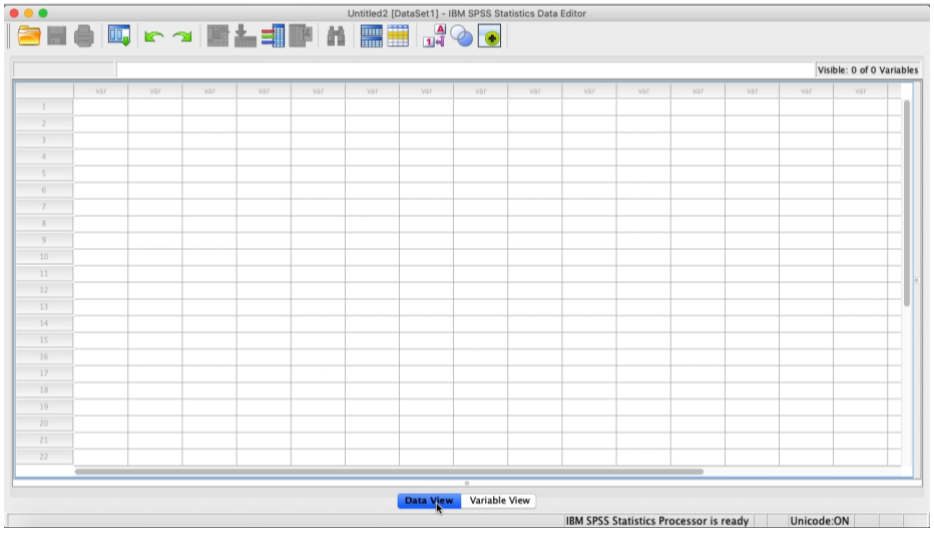
Data Processing using K-Nearest Neighbor in IBM SPSS

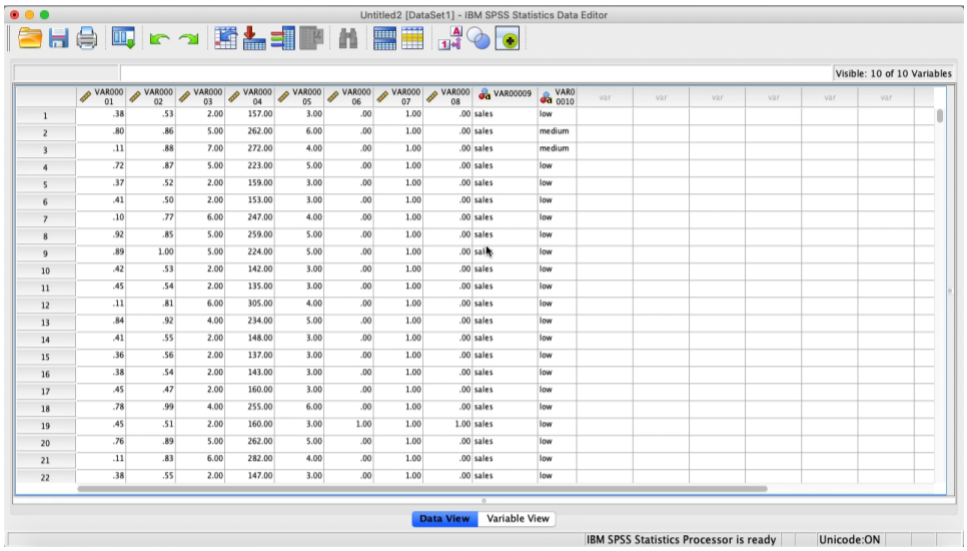
First, a new dataset file was created.



A new dataset window in SPSS is shown as below.

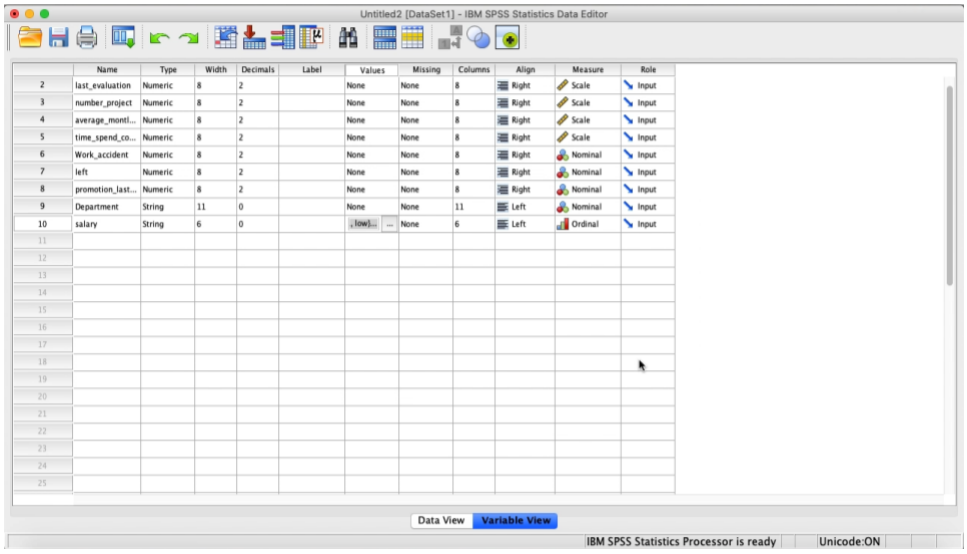


Paste the data into the new dataset. The window now will look like this.

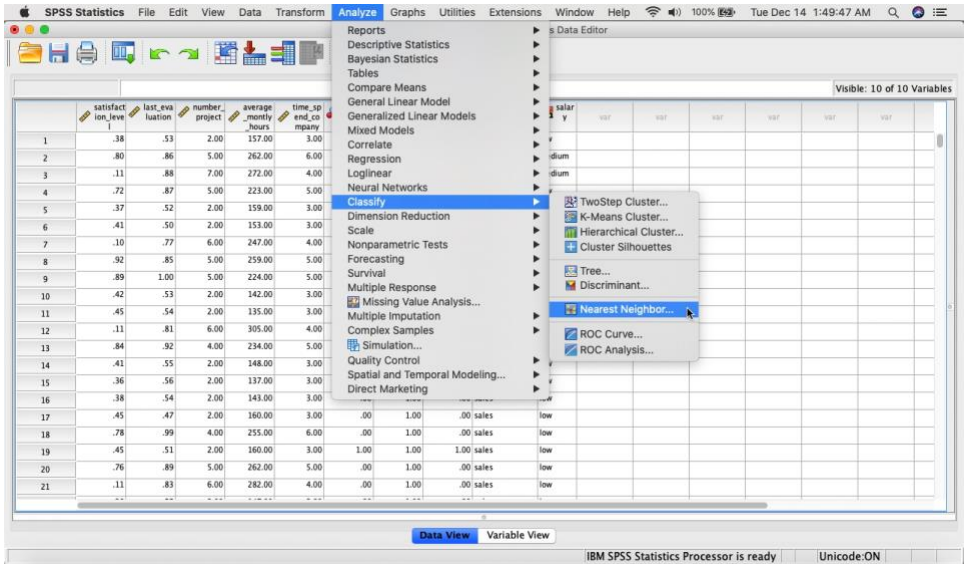


Atribut dari tiap variabel diperiksa dengan cara memilih kolom Variable View di bawah tengah. Tiap variabel diberi nama sesuai dengan nama yang ada pada dataset untuk mempermudah identifikasi variabel. Kemudian tipe skala dari tiap variabel disesuaikan, apakah termasuk ke dalam skala, ordinal, atau nominal.

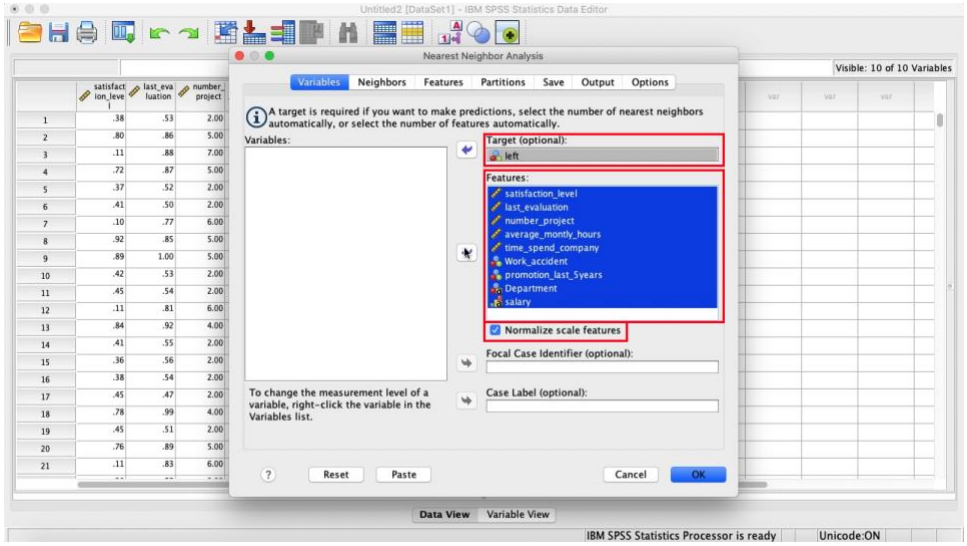
To change the attribute of the variable (from nominal to scale), change the window into Variable View. Under “Measure” tab, we can change the scale of each variable. Each variable was also given new names to make it easier for us to identify the variables.



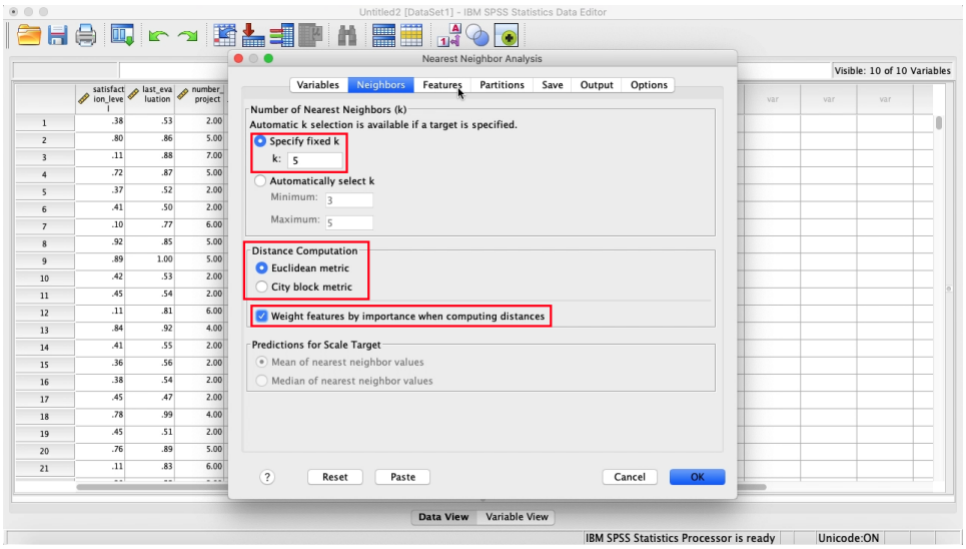
Now it’s time to execute a KNN classification. From the tab “Analyze”, choose Classify and then Nearest Neighbor.



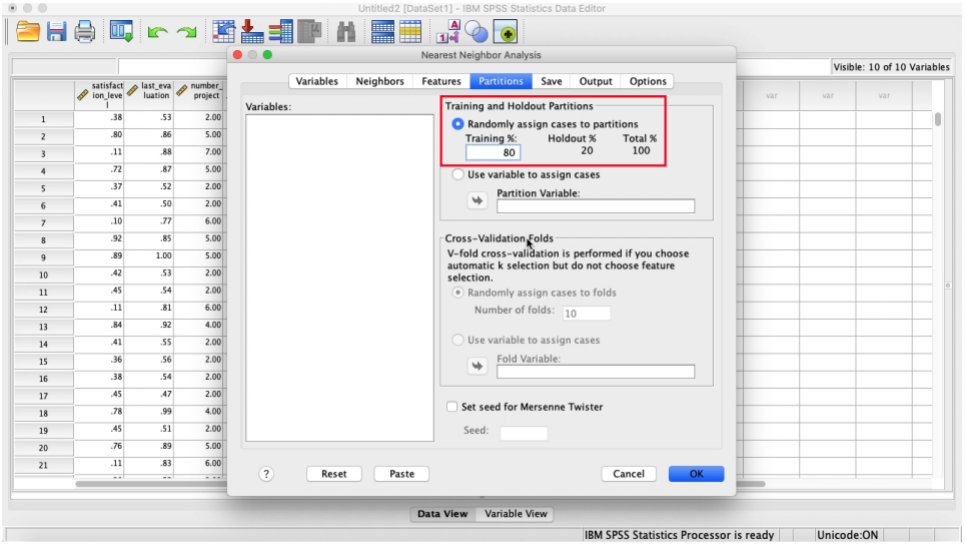
In a new window, we can decide which variables are going to be dependent (response) or independent (predictor) variables. In this study, the dependent variable is *left* which shows whether an employee leave the company or not, and the rest will be the predictors.



The second tab shows how many *k* we are going to use and which method we prefer in measuring the distance. This time the number of *k* is set to 5 and the distance will be calculated using Eucledian method. We can also choose to give different weights to the predictors based on its importance in determining the value of *y*.



In “Partitions” tab, we can determine how many data we are going to divide into training and testing data. Then we can finish by clicking OK.



Result

There are two important outputs obtained from using KNN in SPSS, Classification Table and Error Summary.

Classification Table

Partition	Observed	Predicted		
		0.00	1.00	Percent Correct
Training	0.00	8771	401	95.6%
	1.00	323	2547	88.7%
	Overall Percent	75.5%	24.5%	94.0%
Holdout	0.00	2155	101	95.5%
	1.00	59	642	91.6%
	Missing	0	0	
	Overall Percent	74.9%	25.1%	94.6%

Classification table, or more well-known as Confusion Matrix, compares the predicted data and the actual data. From the table it can be seen that the model was able to predict employee turnover in testing data with 94.6% accuracy.

Error Summary

Partition	Percent of Records Incorrectly Classified
Training	6.0%
Holdout	5.4%

Error Summary shows the percentage of incorrect predictions made by the model in both training data and testing data.