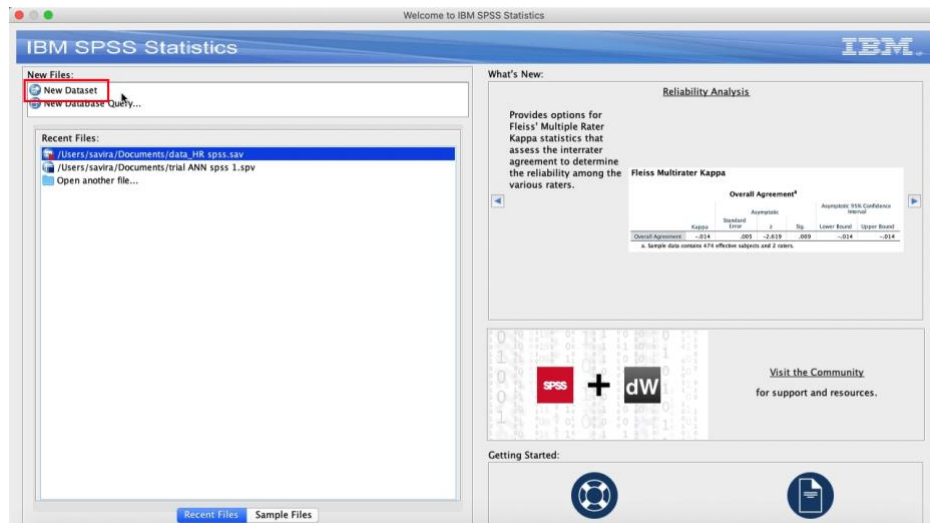
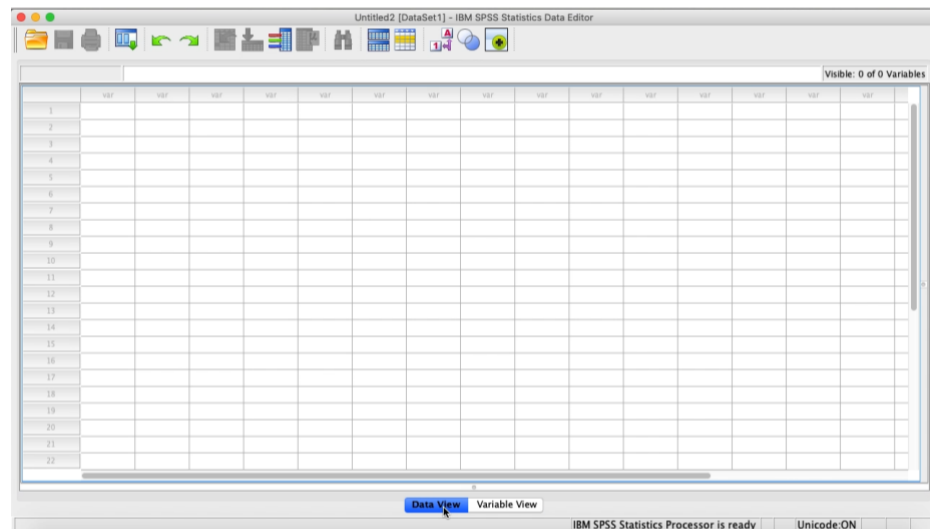


## Data Processing using K-Nearest Neighbor in IBM SPSS

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.

Visible: 10 of 10 Variables

	VAR00001	VAR00002	VAR00003	VAR00004	VAR00005	VAR00006	VAR00007	VAR00008	VAR00009	VAR00010
1	.38	.53	2.00	157.00	3.00	.00	1.00	.00	sales	low
2	.80	.86	5.00	262.00	6.00	.00	1.00	.00	sales	medium
3	.11	.88	7.00	272.00	4.00	.00	1.00	.00	sales	medium
4	.72	.87	5.00	223.00	5.00	.00	1.00	.00	sales	low
5	.37	.52	2.00	159.00	3.00	.00	1.00	.00	sales	low
6	.41	.50	2.00	153.00	3.00	.00	1.00	.00	sales	low
7	.10	.77	6.00	247.00	4.00	.00	1.00	.00	sales	low
8	.92	.85	5.00	259.00	5.00	.00	1.00	.00	sales	low
9	.89	1.00	5.00	224.00	5.00	.00	1.00	.00	sales	low
10	.42	.53	2.00	142.00	3.00	.00	1.00	.00	sales	low
11	.45	.54	2.00	135.00	3.00	.00	1.00	.00	sales	low
12	.11	.81	6.00	305.00	4.00	.00	1.00	.00	sales	low
13	.84	.92	4.00	234.00	5.00	.00	1.00	.00	sales	low
14	.41	.55	2.00	148.00	3.00	.00	1.00	.00	sales	low
15	.36	.56	2.00	137.00	3.00	.00	1.00	.00	sales	low
16	.38	.54	2.00	143.00	3.00	.00	1.00	.00	sales	low
17	.45	.47	2.00	160.00	3.00	.00	1.00	.00	sales	low
18	.78	.99	4.00	255.00	6.00	.00	1.00	.00	sales	low
19	.45	.51	2.00	160.00	3.00	1.00	1.00	1.00	sales	low
20	.76	.89	5.00	262.00	5.00	.00	1.00	.00	sales	low
21	.11	.83	6.00	282.00	4.00	.00	1.00	.00	sales	low
22	.38	.55	2.00	147.00	3.00	.00	1.00	.00	sales	low

Data View Variable View

IBM SPSS Statistics Processor is ready Unicode:ON

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.

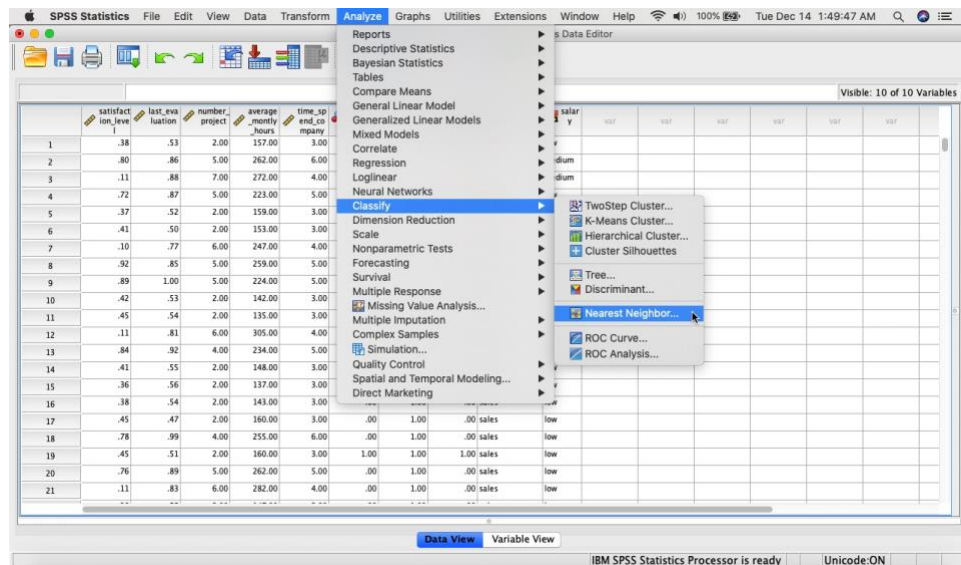
To change the attribute of the variable (from nominal to scale), change the window into Variable View. Under “Measure” tab, we can change the scale of each variable. Each variable was also given new names to make it easier for us to identify the variables.

	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure	Role
2	last_evaluation	Numeric	8	2		None	None	8	Right	Scale	Input
3	number_project	Numeric	8	2		None	None	8	Right	Scale	Input
4	average_monthly...	Numeric	8	2		None	None	8	Right	Scale	Input
5	time_spent_coding	Numeric	8	2		None	None	8	Right	Scale	Input
6	work_accident	Numeric	8	2		None	None	8	Right	Nominal	Input
7	left	Numeric	8	2		None	None	8	Right	Nominal	Input
8	promotion_last_5...	Numeric	8	2		None	None	8	Right	Nominal	Input
9	Department	String	11	0		None	None	11	Left	Nominal	Input
10	salary	String	6	0		low	None	6	Left	Ordinal	Input
11											
12											
13											
14											
15											
16											
17											
18											
19											
20											
21											
22											
23											
24											
25											

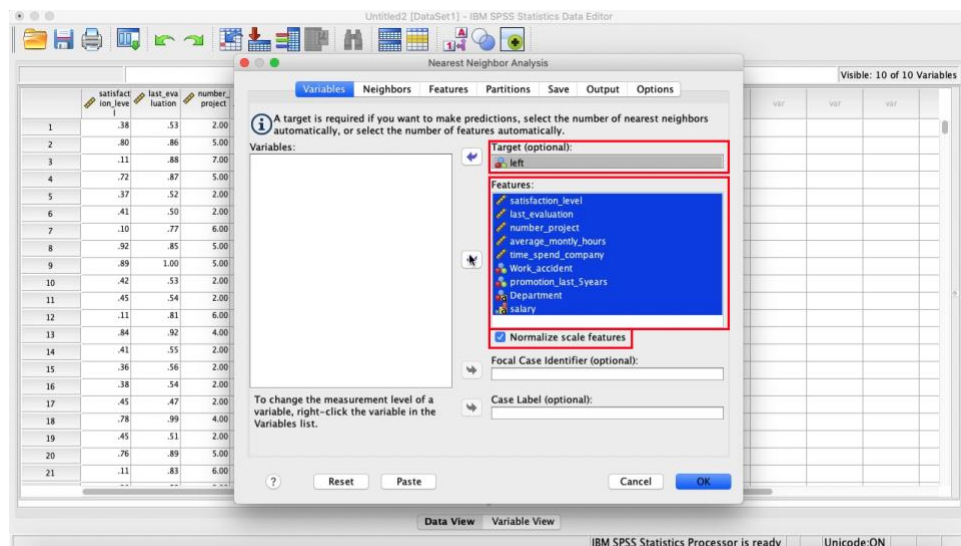
Data View Variable View

IBM SPSS Statistics Processor is ready Unicode:ON

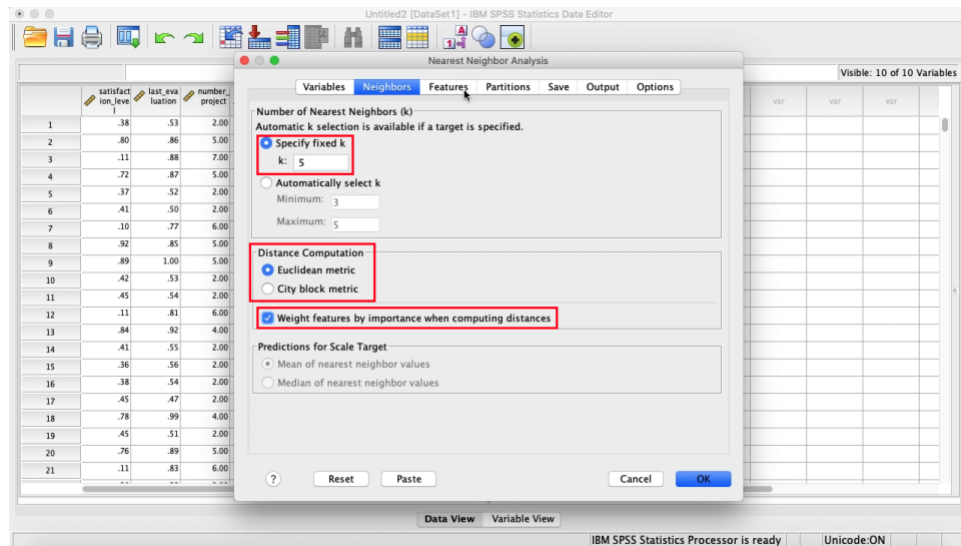
Now it's time to execute a KNN classification. From the tab “Analyze”, choose Classify and then Nearest Neighbor.



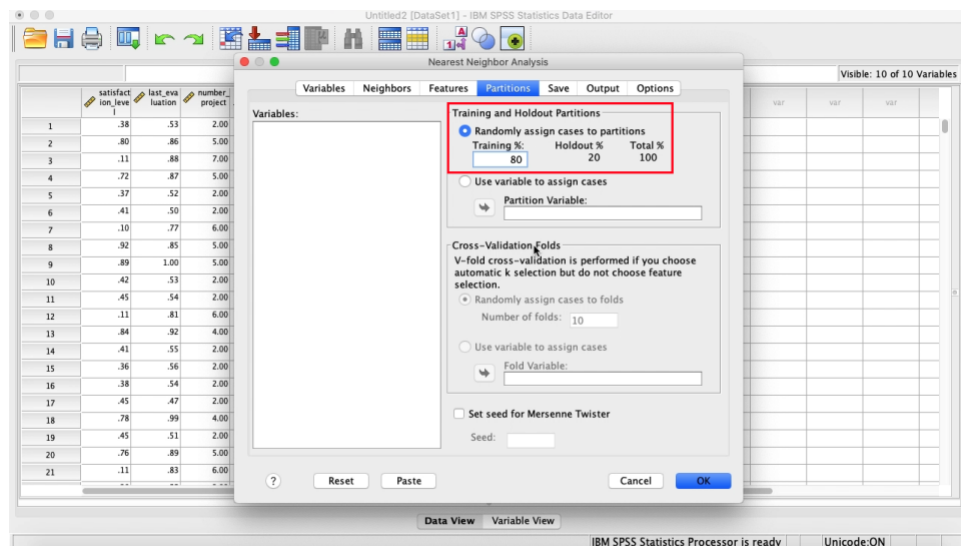
In a new window, we can decide which variables are going to be dependent (response) or independent (predictor) variables. In this study, the dependent variable is *left* which shows whether an employee leave the company or not, and the rest will be the predictors.



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 Euclidian 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.