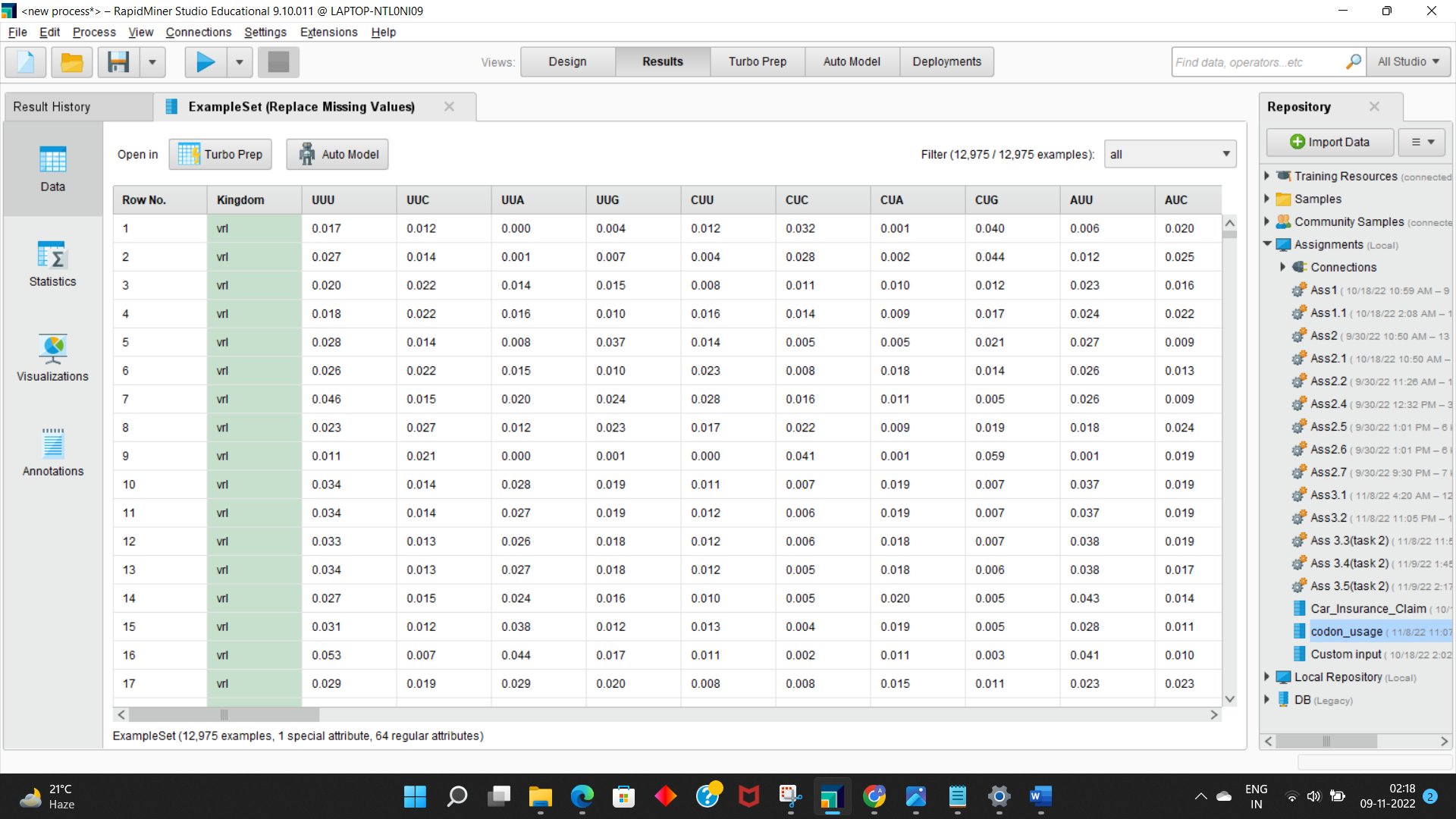
A BRIEF REPORT

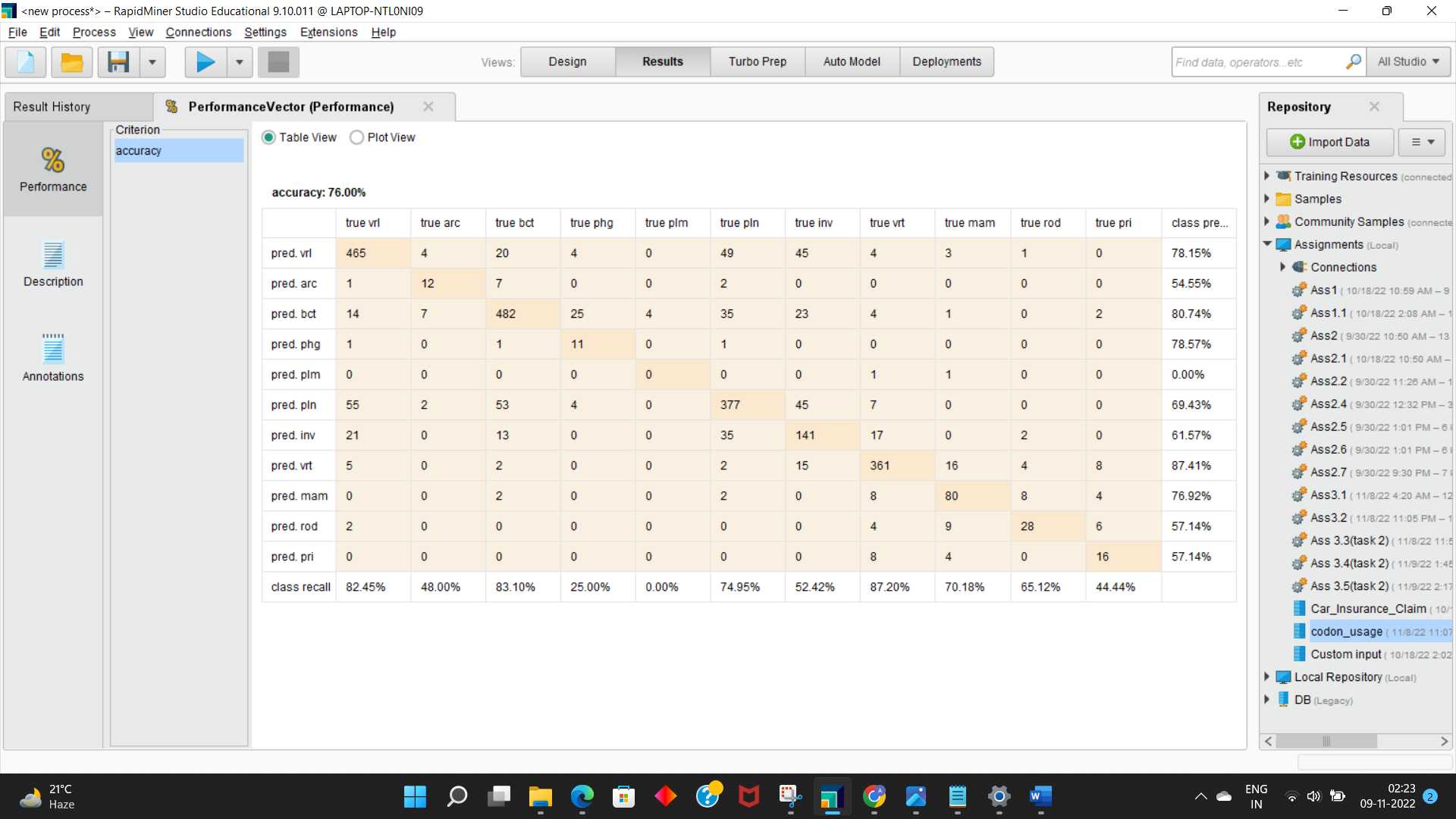
**TASK 1:**

Mention dataset used is:codon-usage <https://www.kaggle.com/datasets/salikhussaini49/codon-usage>

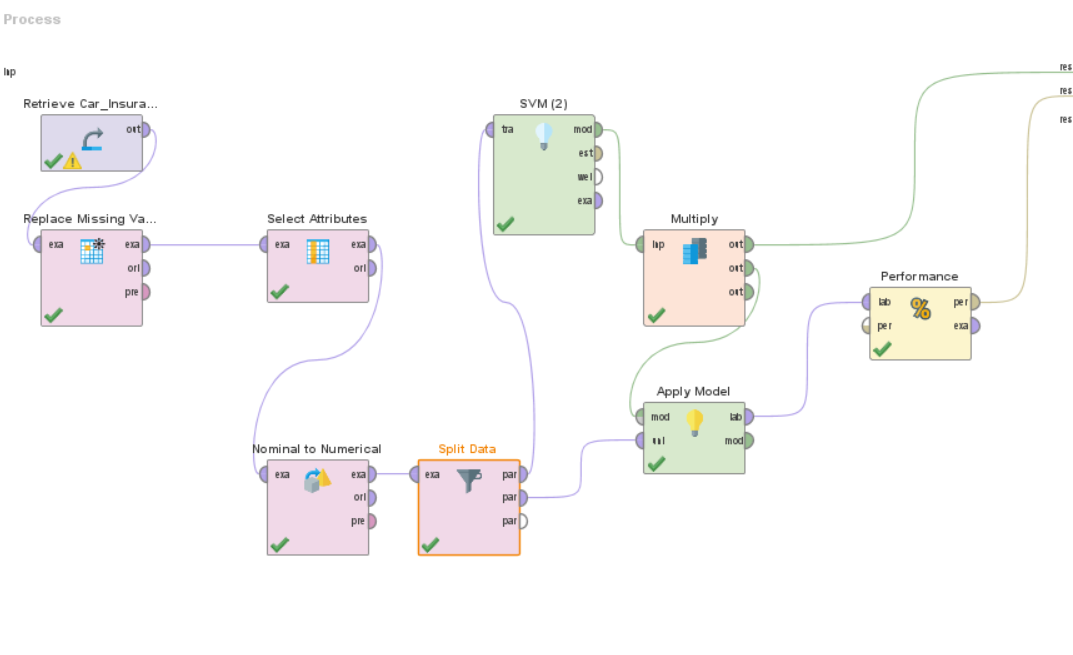
The modifications made such as excluding column containing categorical inputs.



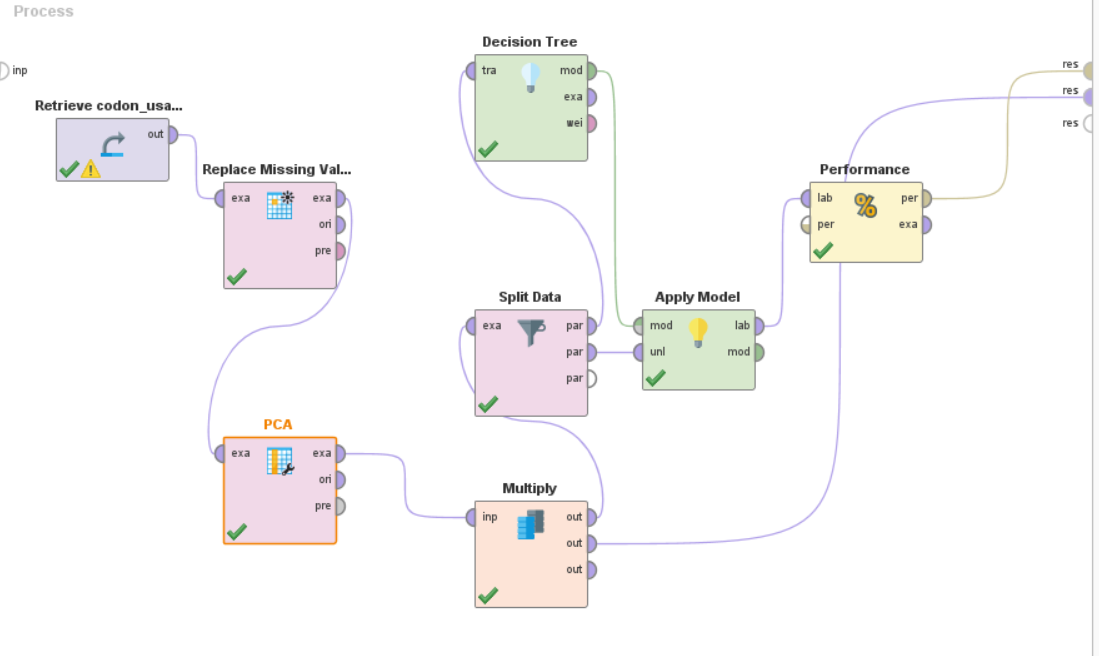
Decision tree has been used to train the model and 90%-10% split gives the best accuracy of 76%

The criterion used is information gain.

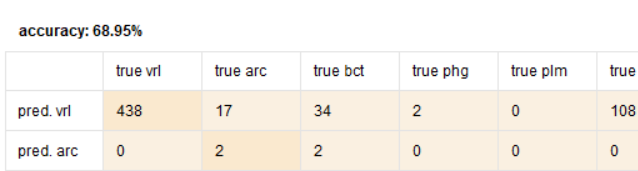
The process structure:



**Applying PCA:**

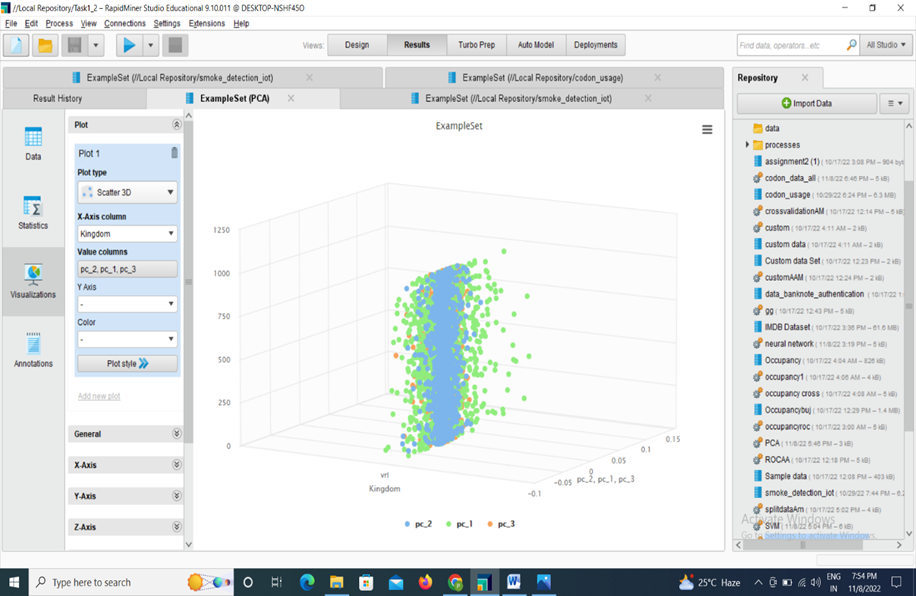
****

When Keep variance (0.95) default was used accuracy obtained was 77% approx. For fixed number ‘k = 5’, accuracy obtained is 68.95%

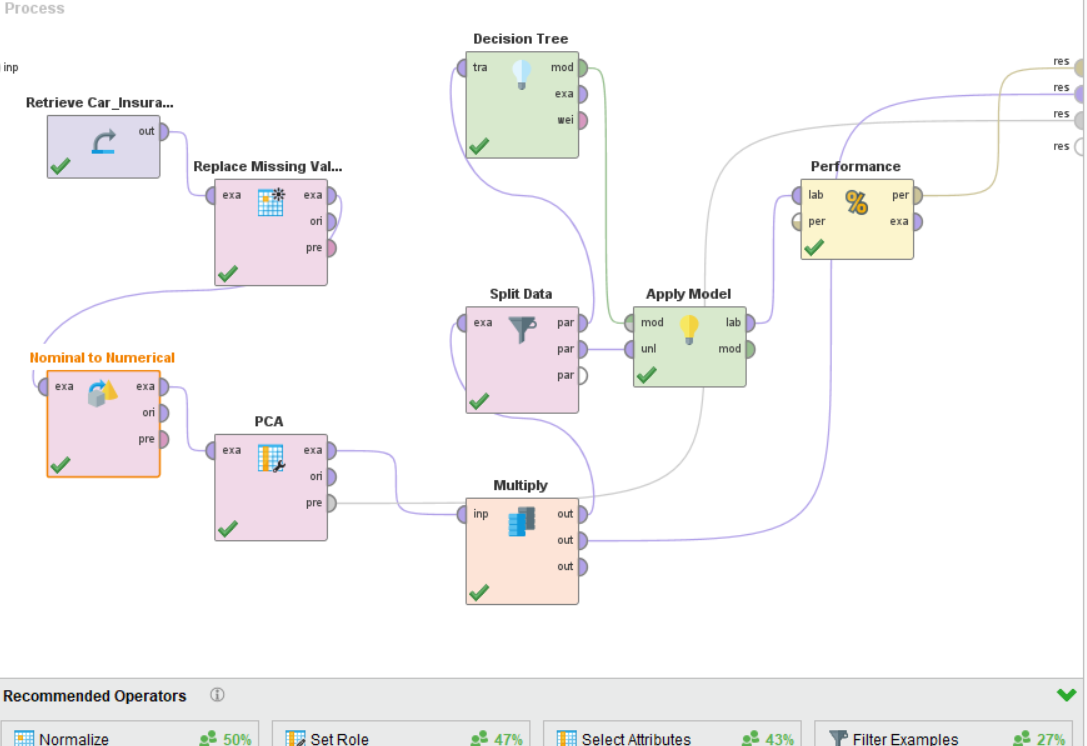


As K is increasing, accuracy is also increasing. For K = 30 best model is obtained.

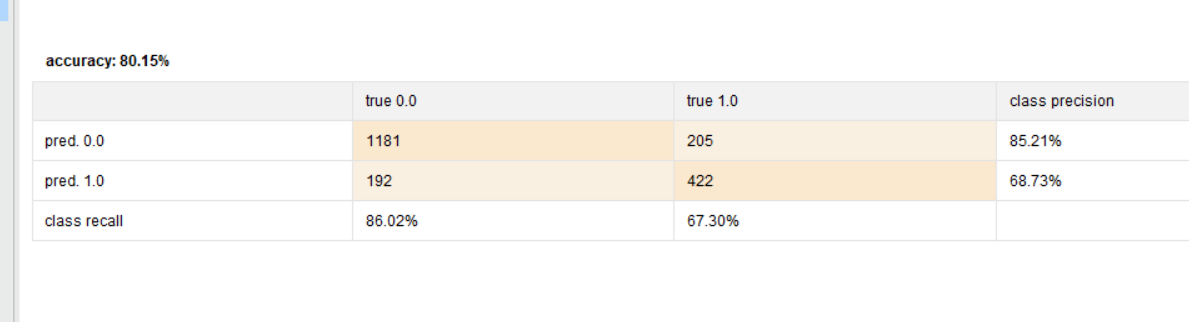
3D Scatter plot:



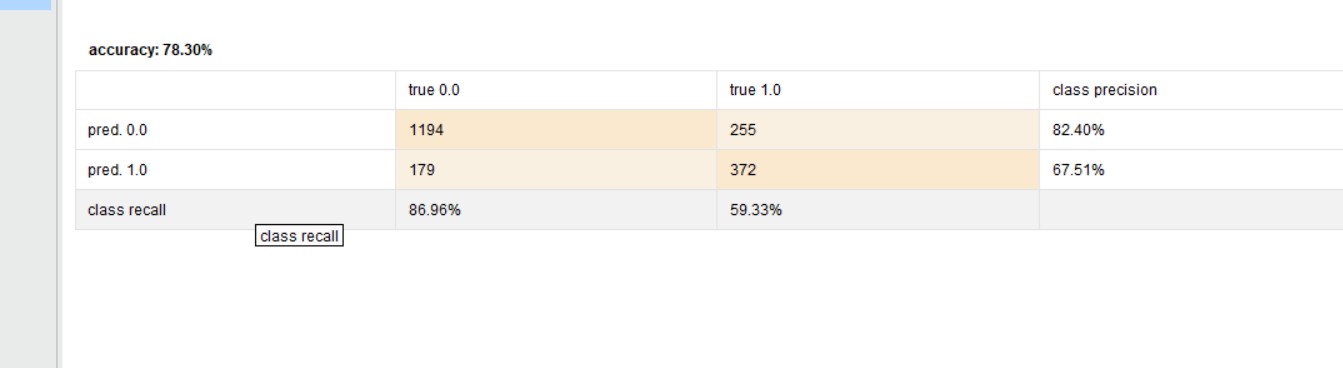
**Second Dataset is Car insurance claim**

****

For K = 30, 80.16% accuracy is obtained



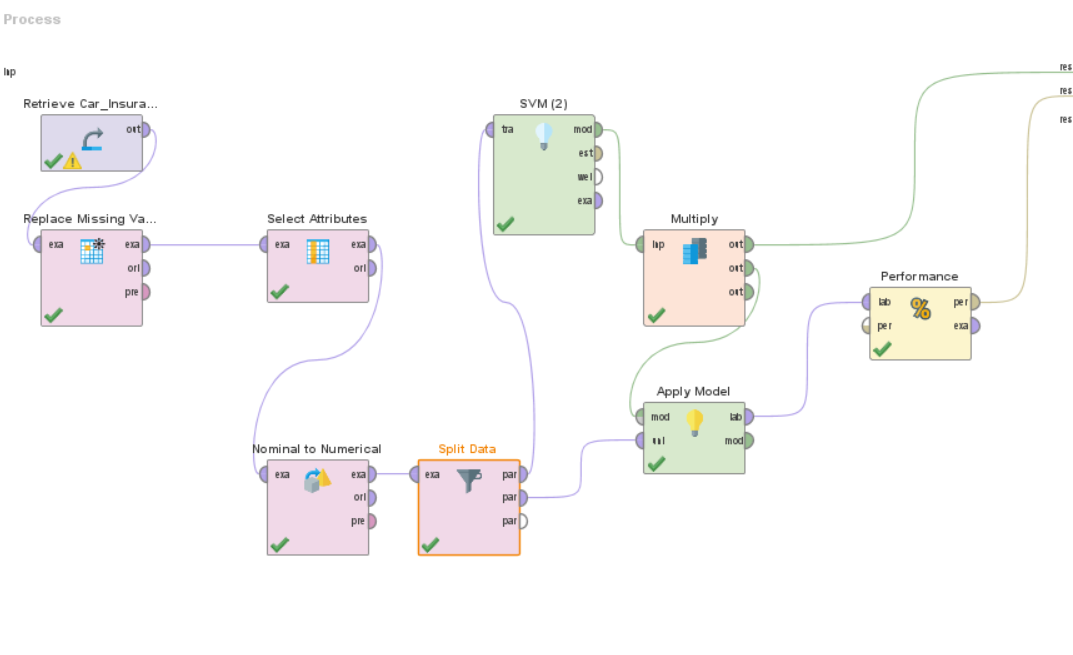
When we kept variance, 78% accuracy was shown.



**TASK 2:**

The dataset we used is Car insurance claim.( used in Assignment 1)

Label is outcome which is binary classification. There are non numerical attributes. We have used operator ‘nominal to numerical’ to convert attributes into numerical values.



1. **Radial-**

The maximum accuracy obtained is 80.20% and the split was 90% - 10%

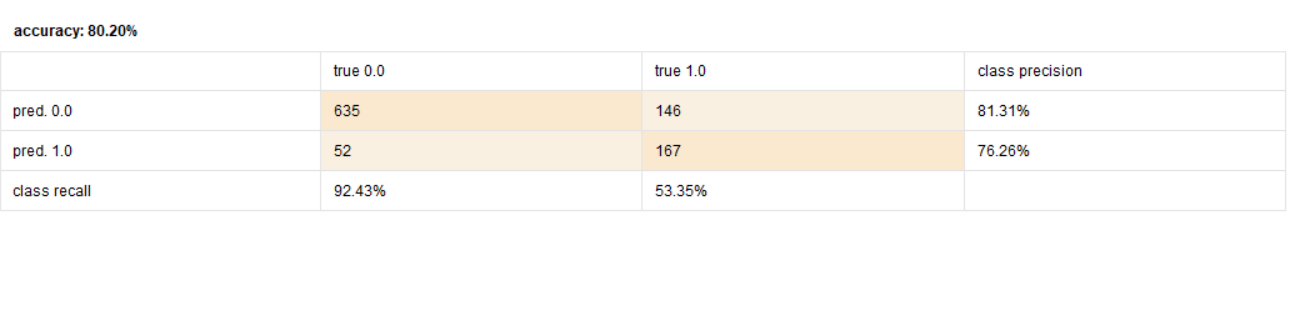
The following are the obtained results:

70 - 30: 79.87%

80 - 20: 80.10

90 - 10: 80.20

60 - 40: 79.17



1. **Dot-**

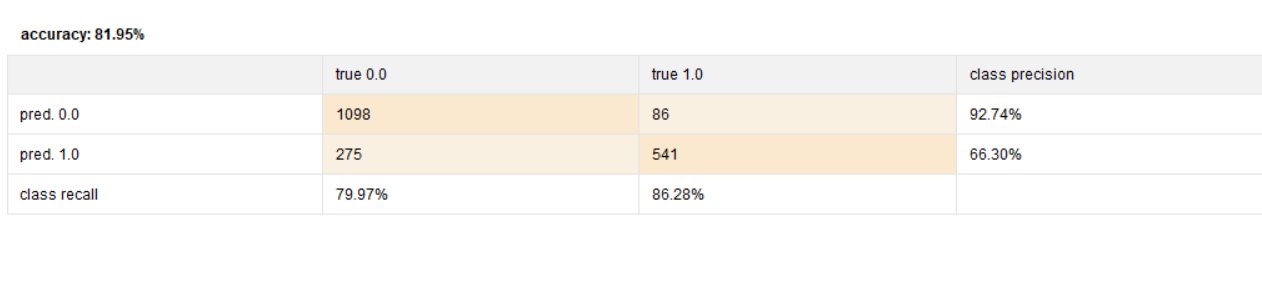
The maximum accuracy obtained is 81.95% and the split was 80%-20%

Following are the results of different splits-

90 - 10:81%

70 - 30:81.40%

60 - 40:81.50%



1. **Polynomial -**

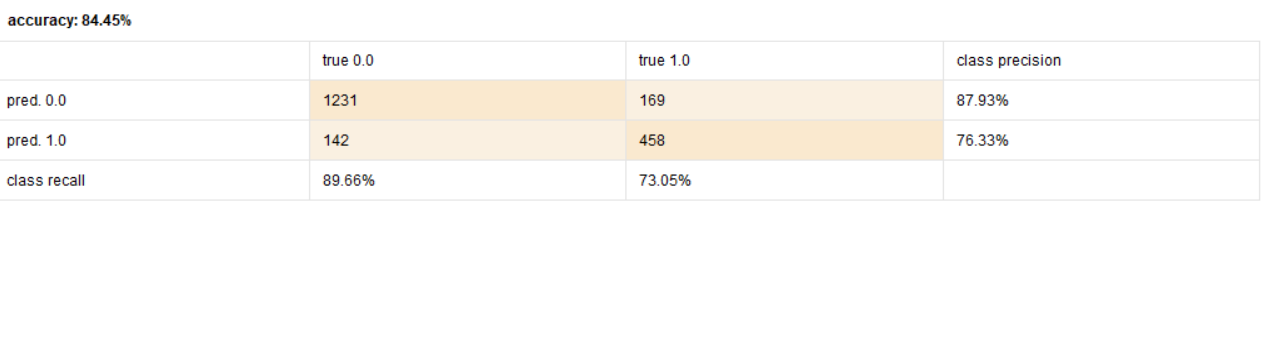
The maximum accuracy obtained is 84.45% and the split was 80%-20%

Following are the results of different splits-

90 - 10:83%

70 - 30:83.97%

60 - 40:84.20%



**Class accuracy and False positive Rates:**

Again from the above figures of performance vector(confusion matrix), we can conclude the class-wise accuracy for

Radial - 92.43%

Dot - 79.97%

Polynomial - 89.66%

The best model according to accuracy is where Polynomial Kernel function is used but with class-wise accuracy, it is Radial kernel function used model.

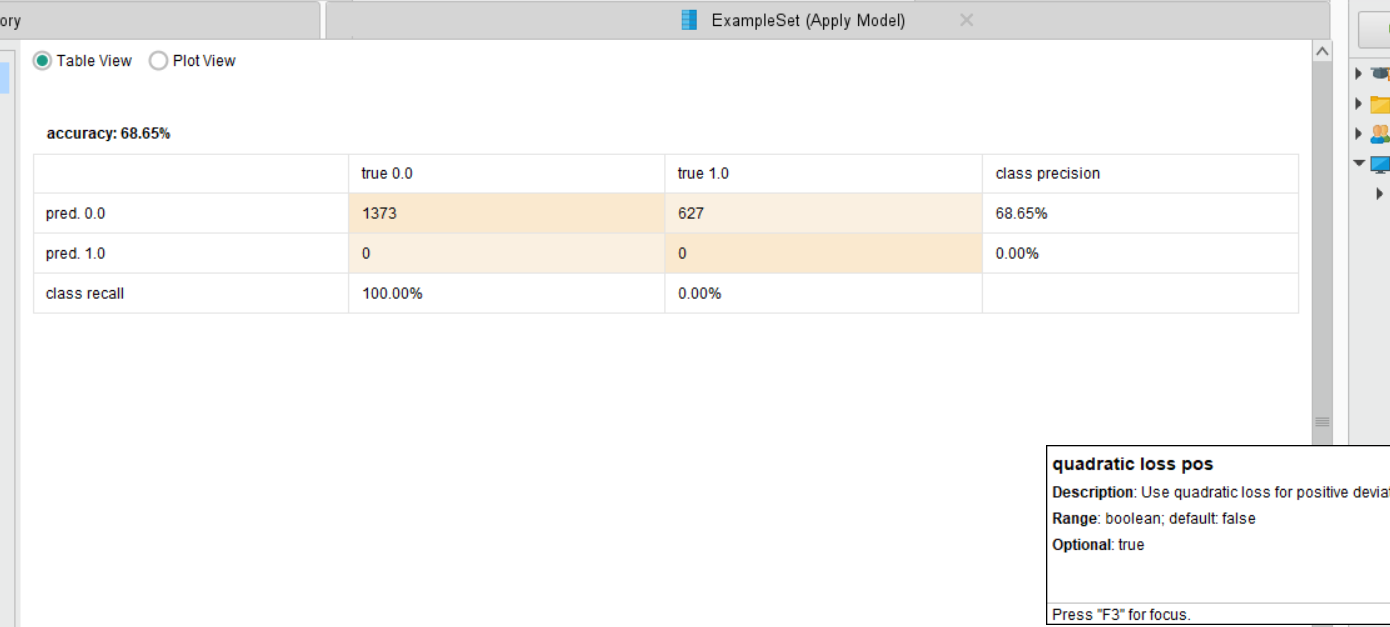
We will go with a polynomial kernel function used model having high ***false positive Rate = 0.1034.***

**Best SVM Model:**

The best SVM model is Polynomial Kernel function used model.

**Epsilon parameter:**

It’s an insensitivity constant parameter.By increasing the epsilon accuracy and class wise accuracy both are falling, it tell’s about the loss function.



The above accuracy was obtained when epsilon was 2.

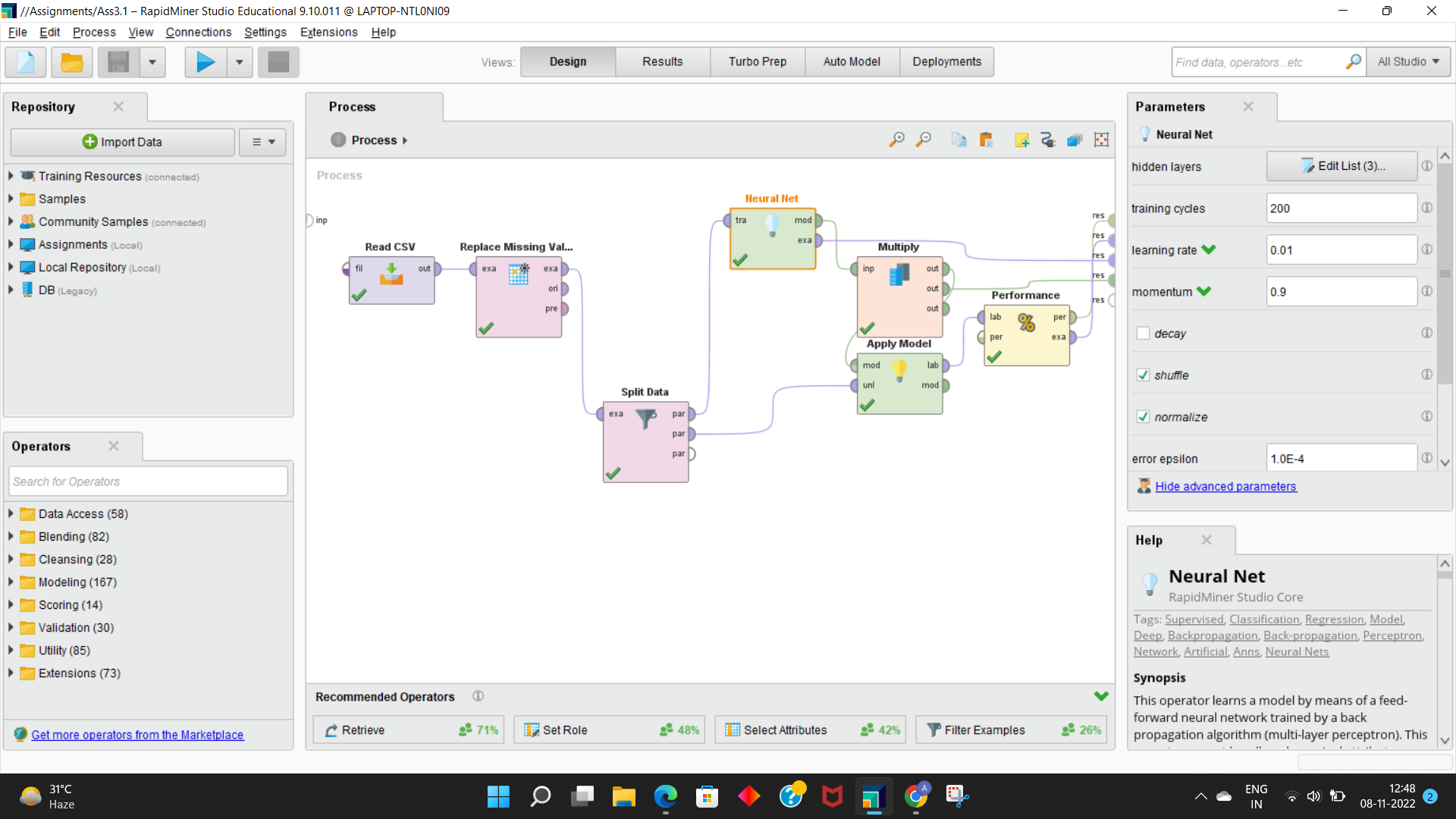
**TASK 3:**

We have took the dataset given in the assignment sheet, which is

* codon-usage <https://www.kaggle.com/datasets/salikhussaini49/codon-usage>

Some modifications have been made like the first 4 attributes which contain categorical inputs have been ignored(excluded). Attribute ‘Kingdom’ has been assigned as a label. ‘Replace missing value’ operator has been used to replace the missing value of attribute with average value.

80% - 20% is used as mentioned in the assignment. The design panel is shown below.



The operators used are ***Read CSV, Replace Missing value,Split Data, Neural Net, Multiply, Apply model*** and ***Performance.***

1. **Neural Networks:**

The following networks have been made with default settings of

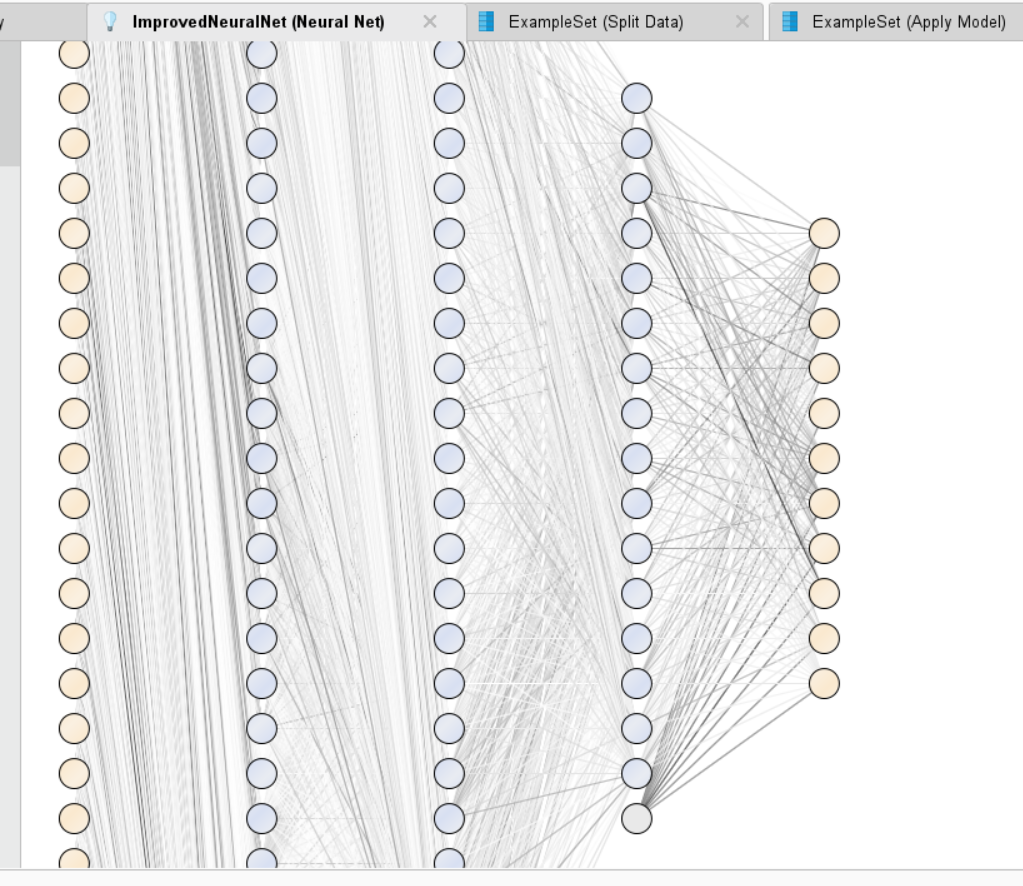
*Training steps*: 200

*Learning rate*: 0.01

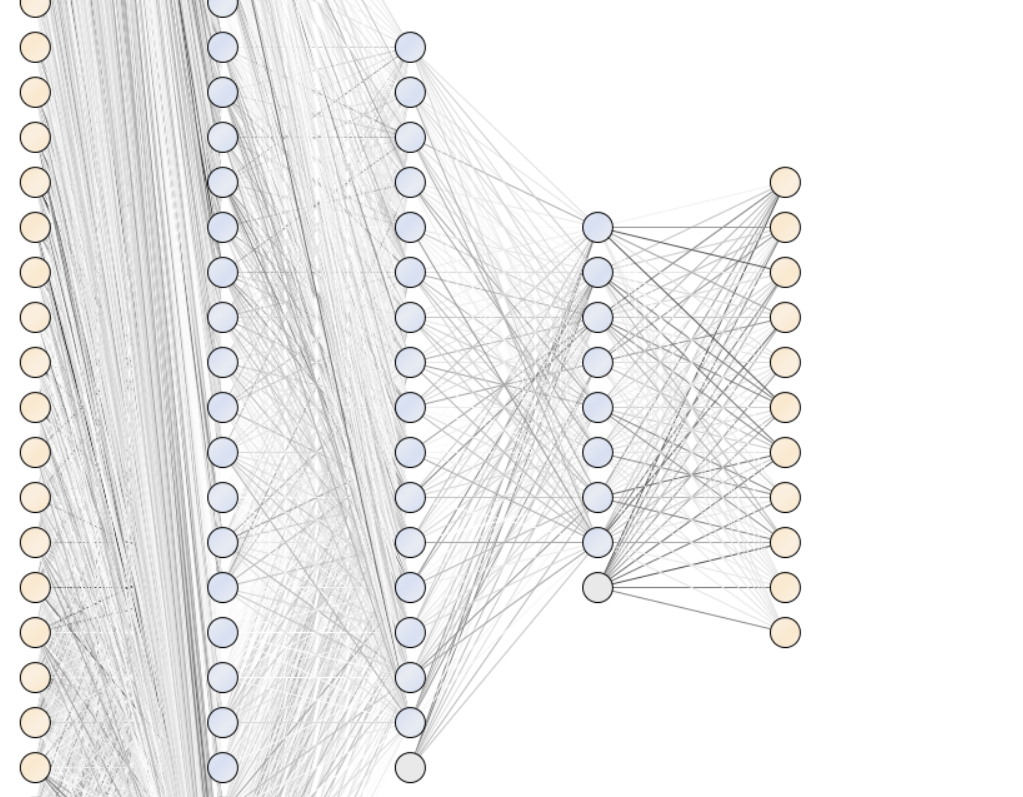
*Momentum*: 0.9

*Local Random seed*: 1992

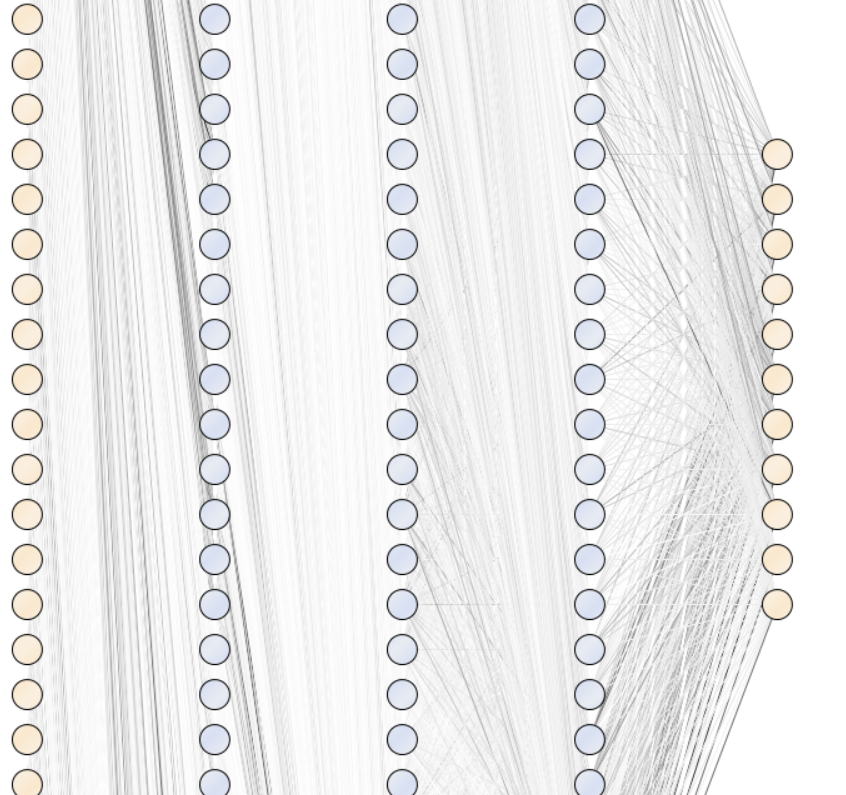
Input - hidden layer 1 (64) - hidden layer 2 (32) - hidden layer 3 (16) - Output



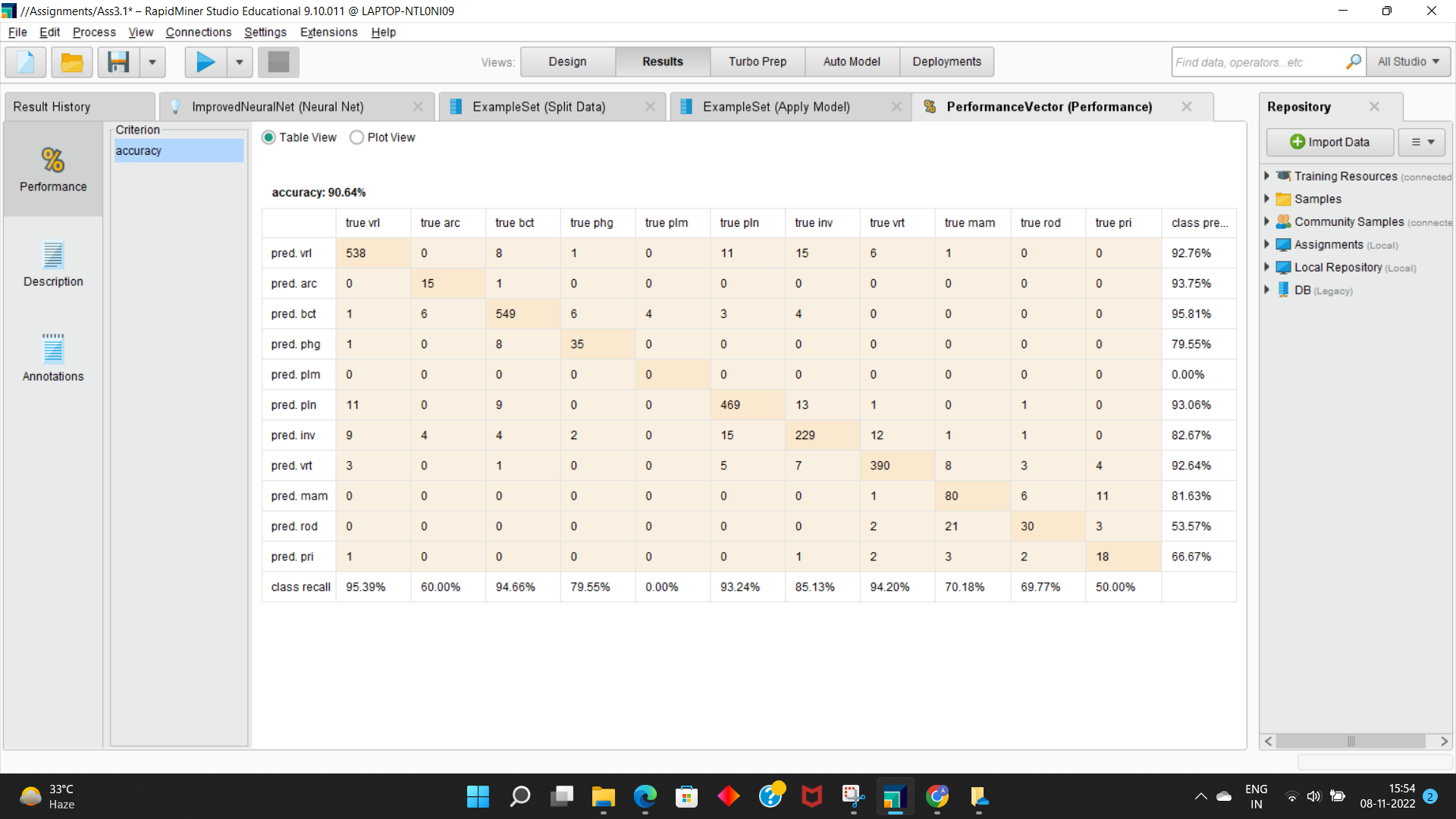
* 1. Input - hidden layer 1 (32) - hidden layer 2 (16) - hidden layer 3 (8) - Output



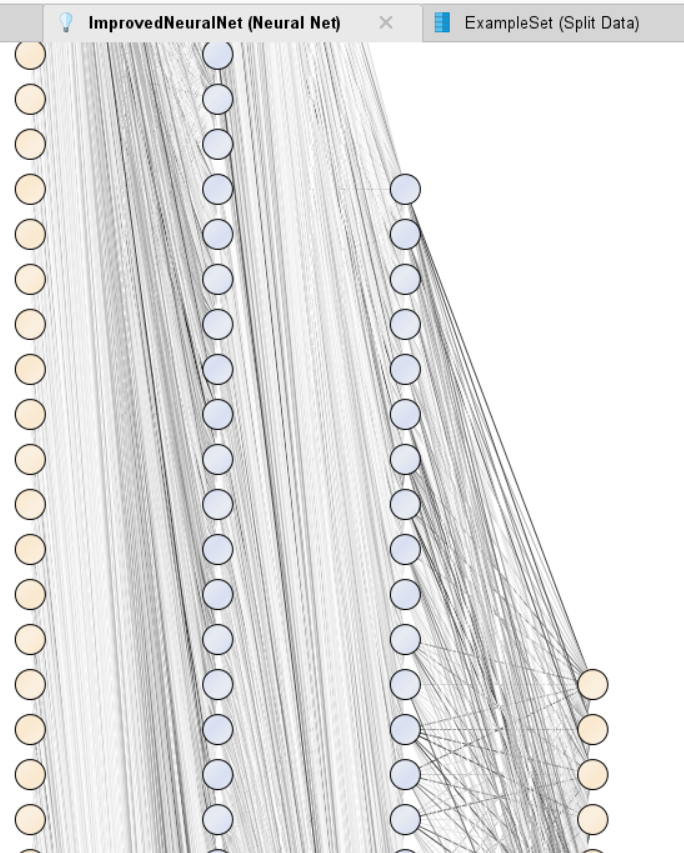
* 1. Input - hidden layer 1 (128) - hidden layer 2 (64) - hidden layer 3 (32) - Output



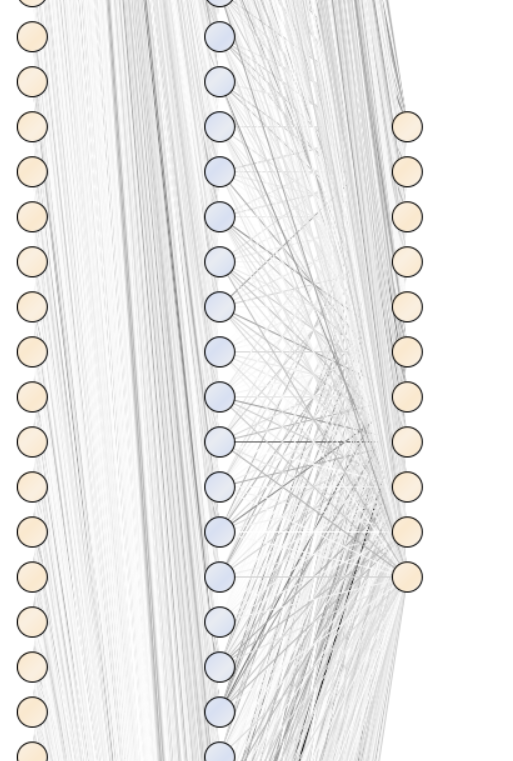
* 1. Input - hidden layer 1 (64) - hidden layer 2 (32) - Output



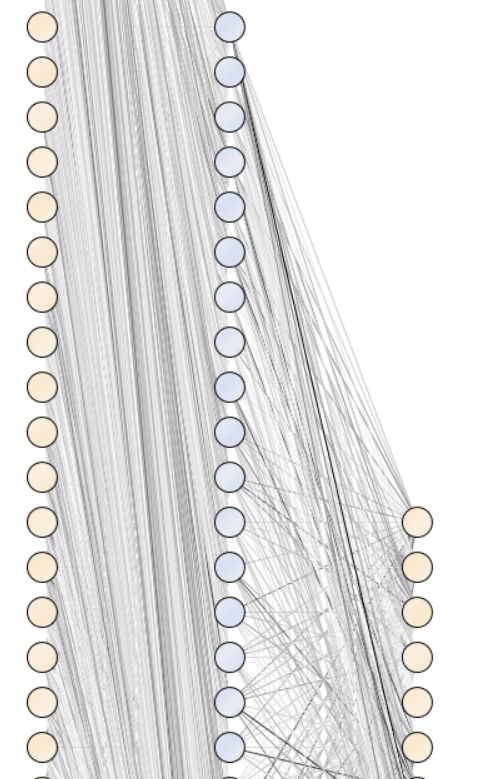
* 1. Input - hidden layer 1 (64) - hidden layer 3 (16) - Output



* 1. Input - hidden layer 1 (64) - Output



* 1. Input - hidden layer 1 (32) - Output



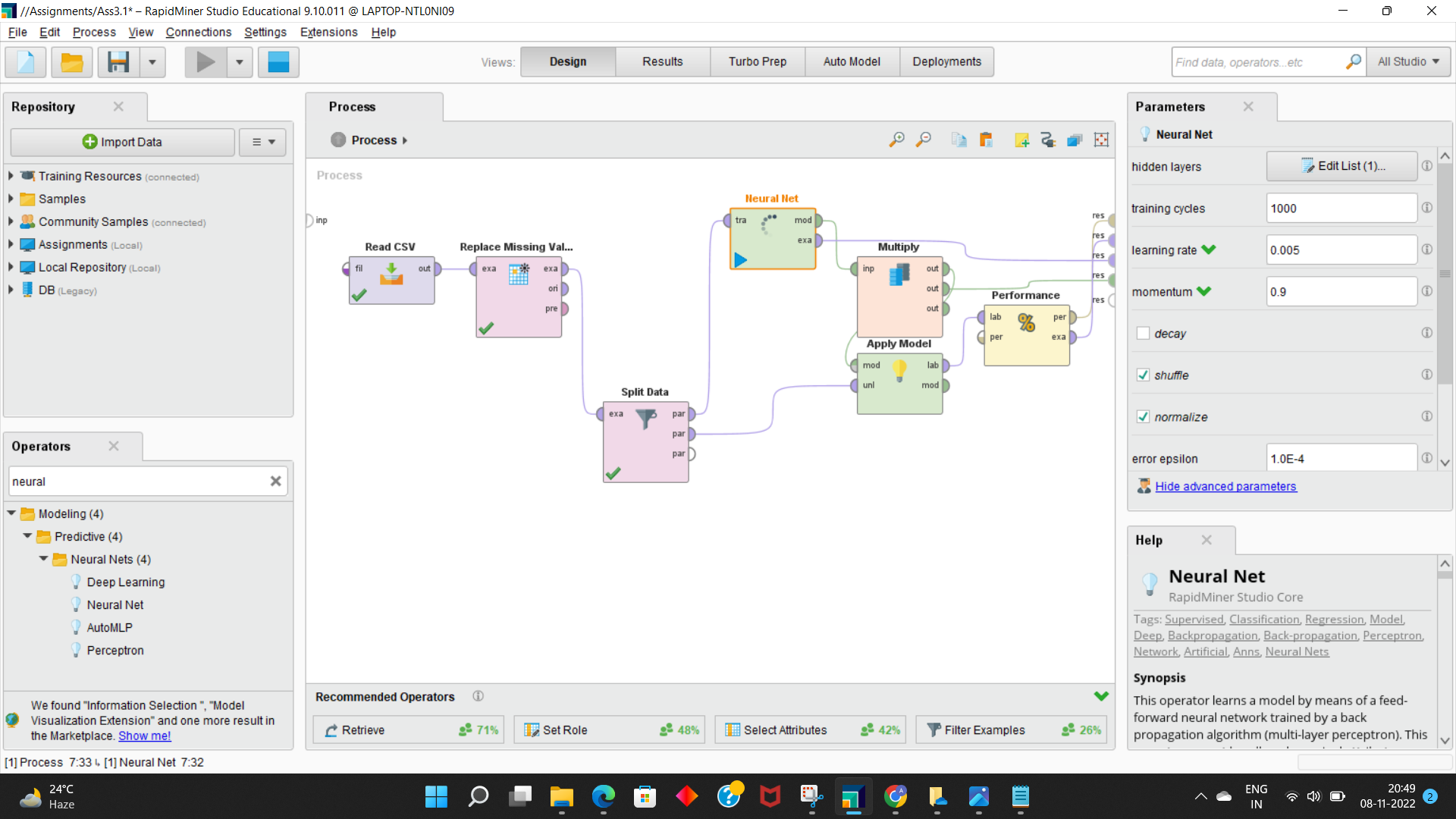
The best is with the 1 hidden layer of 64 nodes having accuracy of 92.30%

**C)Variation of training Steps:**

|  |  |
| --- | --- |
| **Training steps** | **Accuracy** |
| 20 | **84.51%** |
| 50 | **89.52%** |
| 100 | **91.37%** |
| 400 | **92.18%** |
| 1000 | **92.49%** |

We can observe that at 1000 training steps, there is slight increase in accuracy. The table also shows the change in training examples affects the performance of neural networks.

On variation of learning rate,



For 0.005 - The accuracy of the neural network was 92.16% and also the process was time consuming.(Execution Time 12 minutes)

For 0.001 - The accuracy was 91.91%, execution time was 11 minutes.

For 0.05 - The accuracy was 92.37%,execution time 11 minutes approx

For 0.2 - The accuracy was 22.5%, execution time 12 minutes.

**F)**

Thus, the final conclusion is as learning rate is increasing i.e. from 0.005 to 0.2., accuracy is decreasing.

Also, the default setting of learning rate as 0.01 is giving maximum accuracy of 92.49% with 1000 training examples.

**G)**

For default momentum of 0.9, we have accuracy of 92.49%

When we decrease the momentum , ***accuracy is decreasing*** , for momentum of 0, we have accuracy of 92.26%

**I)**

If we reduce the number of examples and increase the momentum we will get better accuracy.

**Research Paper Conclusion:**

It is found that Artificial Neural Network models give better results than Radial Basis Function Neural Network and Generalize Regression Neural Network models for all relevant input variables due to its generalization capability. The least influencing variables are Lat, Long, CI and ER. Therefore Rapid Miner can be used to find out relevant input variables for SR prediction which is useful for variable selection in prediction. To check the authentication of Rapid Miner for relevant input variable selection five ANN models are developed with nftool and its prediction accuracy is compared with five Radial Basis Function Neural Networks and five Generalized Regression Neural Networks. It is found that ANN models developed with nftool give better results than RBFNN and GRNN for solar radiation prediction.

https://www.sciencedirect.com/science/article/pii/S1364032115008035