Dataset 1: Breast Cancer Wisconsin (Diagnostic Dataset)

Link: https://www.kaggle.com/uciml/breast-cancer-wisconsin-data

About Dataset:

https://www.kaggle.com/uciml/breast-cancer-wisconsin-data

Addition to pre-processing done for previous project, we did normalization using range as parameter.

Algorithm Implementation:

1. **KNN**

Results:

Linear Kernel with No-cross validation set

Test Dataset Accuracy

```
Prediction   0   1
         0  73   5
         1   4  18
Accuracy : 0.91
```

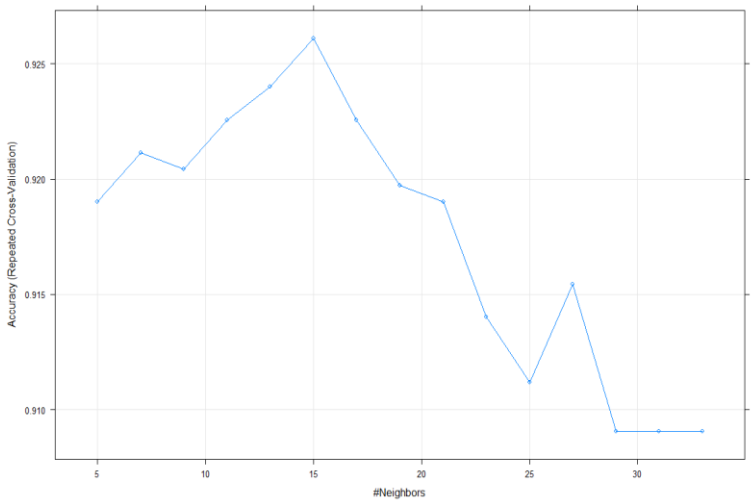
Train Dataset Accuracy

```
Prediction   0   1
         0 279  30
         1   1 159
Accuracy : 0.9339
```

Test dataset

```
Prediction   0   1
         0  73   4
         1   4  19
Accuracy : 0.92
```

Learning Curve:



Value of k is chosen such that it is small and gives maximum accuracy.
The final value used for the model was k = 15.

Though the accuracies are comparable, accuracy of cross validation being slightly higher. The false negative rate of the model built using cross-validation is less than the model which is built with no cross-validation.

The false Negative rates (Type II errors) more severely impact the performance in case of model performance in case of cancer detection problems. False Negative rate with cross validation approximately 0.21% and without cross validation it is 0.22%.

Dataset 2: Customer Churn Dataset

Link: https://www.ibm.com/communities/analytics/watson-analytics-blog/predictive-insights-in-the-telco-customer-churn-data-set/

Data Preprocessing

Addition to pre-processing done for previous project, we did normalization using range as parameter.

KNN Implementation:

Results:

KNN without cross validation set

Test dataset Accuracy

```
Prediction    0    1
         0 1319  259
         1  210  321
Accuracy : 0.7776
```

KNN with cross validation set

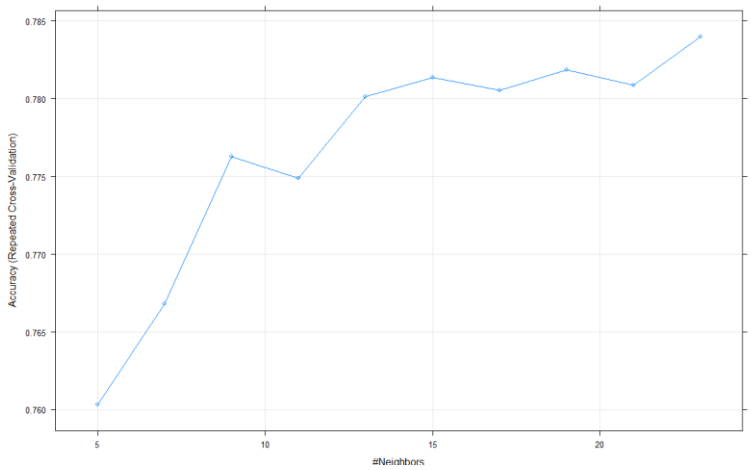
Train Dataset Accuracy

```
Prediction    0    1
         0 3201  510
         1  433  779
Accuracy : 0.8085
```

Test dataset

```
Prediction    0    1
         0 1344  274
         1  185  306
Accuracy : 0.7824
```

Learning Curve:



Value of k is chosen such that it is small and gives maximum accuracy.
The final value used for the model was k = 23.

We can see from the confusion matrix that when we used the cross validation, the accuracy of prediction has been increased on test dataset.

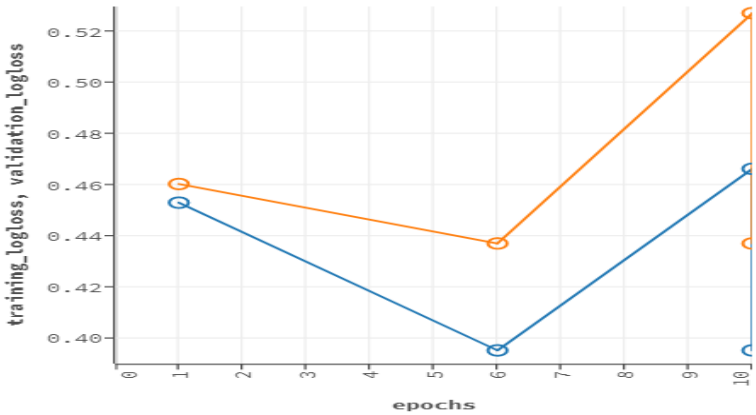
Also as this dataset is an example of class imbalance then the more accurate parameter to judge model performance is F1 score, calculating F1 Score given by 2*(p*r)/ (p + r) where p is precision and r is recall. The F1 value for model with no cross validation is 0.57 and for model with cross validation is 0.63 where 1 being the highest value.

As a rule of thumb in case of no-cross validation, the value of k is chosen as the square root of the training examples. But using caret package we got the value of k which is very lower, which implies less computation hence less time to get the results.

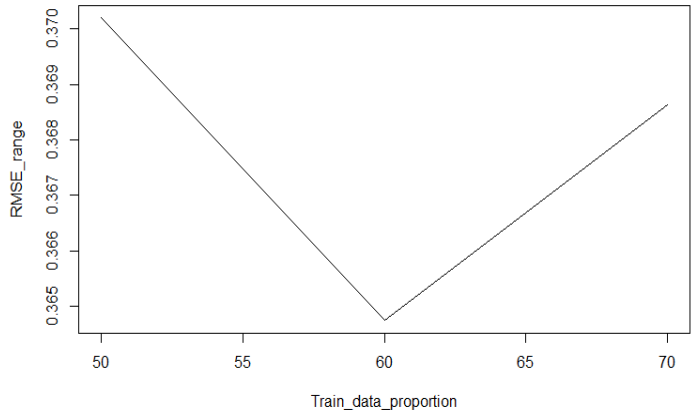
With Cross validation the accuracy was found to be increased.

Neural Network Implementation:

Dataset 2: Customer Churn Dataset



When we experimented with different epoch values for the Churn dataset, we find that the logloss is the least for epoch = 6 which is valid for train as well as test sets.

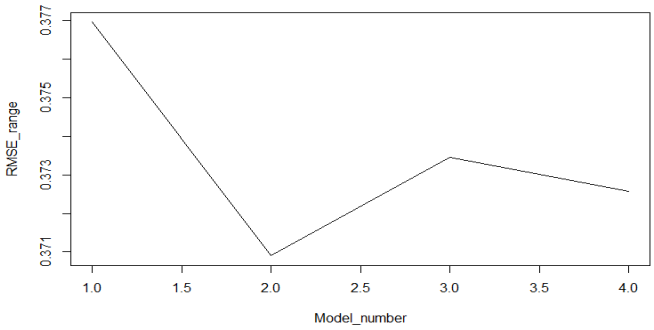


As we can see from the figure given on the left, the RMSE of the resulting model changes albeit very slightly when we change the proportion of total data that we used to train the model.

It has been argued for long in ANN history that 2 hidden layered ANN is sufficient to represent any complex functional form, however for our dataset we varied the number of hidden layers and the number of neurons in each layer and observed the effect on the error of the resulting models which resulted in the plot given below –

The legend for the below graph is as follows-

Model number	Hidden layer configuration
1	50,50
2	200,200
3	100,50,50
4	50,50,50,50



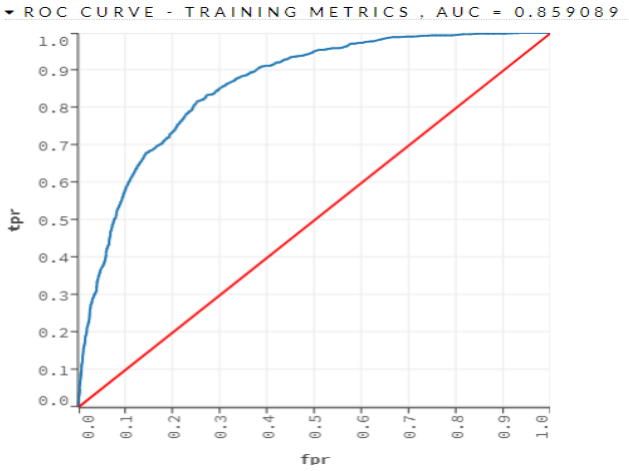
The variation in number of hidden layers and number of nodes is indicative of the underlying trend of change in error. We can see that the least amount of error is for the combination of (200,200).

This is in keeping with the heuristic knowledge that ANN with 2/3 hidden layers and optimum number of nodes can typically represent most complex functional forms and give better accuracy than ANN with more than 3 hidden layers.

Activation Function	AUC
tanh	0.829175
Rectifier	0.892588
Maxout	0.917157

When we change the activation functions for the above data, we find that the AUC is the most for 'Maxout' activation function which might not necessarily be the case for all data sets.

The ROC/AUC curve for the final ANN model is –



Choosing Best/Final model for customer churn dataset:

- Varied the proportion of train dataset.
- Varied number of hidden layers and number of neurons used.
- Varied activation function in h2o Flow
- Varied Number of epochs in h2o Flow

Built final model choosing the best characteristics obtained from above 4 experiments.

We chose to plot AUC curve as it is used in classification analysis in order to determine which of the used models predicts the classes best. The final model selected gave the AUC as 85.90%.

Performance of final model on test data -

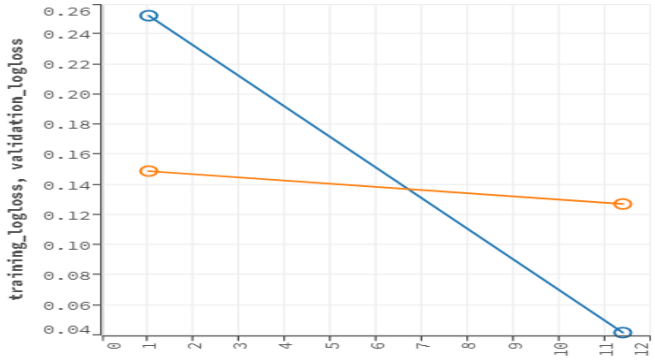
```
Confusion Matrix (vertical: actual; across: predicted)
        No Yes    Error       Rate
No     555 226 0.289373   =226/781
Yes     65 209 0.237226    =65/274
Totals 620 435 0.275829  =291/1055
```

Accuracy = 80.09%

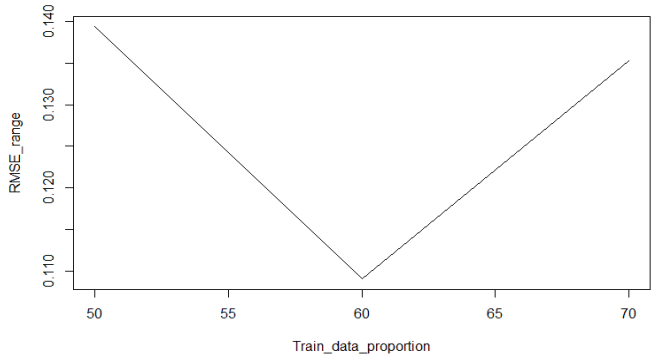
From the confusion matrix we have the accuracy as 80.09% on test dataset.

Neural Network Implementation

Dataset 1: Breast Cancer Dataset

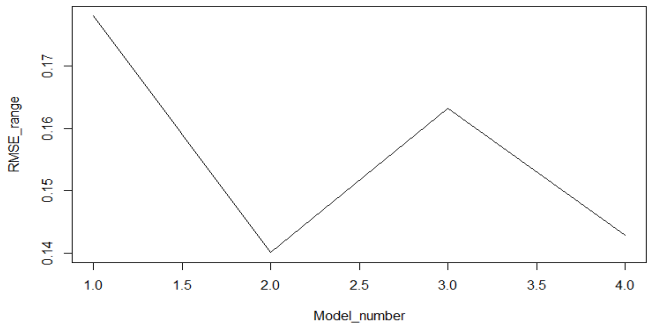


When we experimented with different epoch values for the Churn dataset, we find that the logloss is the least for epoch = 11 which is valid for train as well as test sets. When we ran the models, we found that we found that 10 epochs yielded very similar results, so we chose to keep epochs as 10.



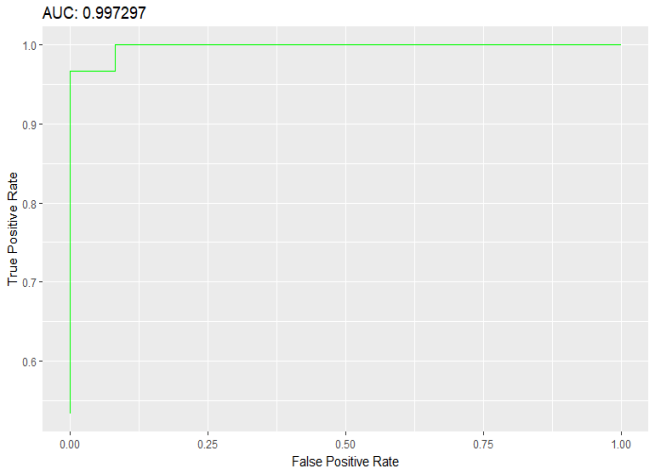
As we can see from the figure given on the left, the RMSE of the resulting model changes when we change the proportion of total data that we used to train the model. From the graph, it is clear that RMSE is least for train proportion as 60%.

Model number	Hidden layer configuration
1	100,100
2	200,200
3	200,100,50
4	200,200,200



We can see that the least amount of error is for the combination of (200,200).

Thus we decided to go ahead with number of neurons as 200 and 2 hidden layers.



Choosing Best/Final model for customer churn dataset:

- Varied the proportion of train dataset.
- Varied number of hidden layers and number of neurons used.
- Varied Number of epochs in h2o Flow

Built final model choosing the best characteristics obtained from above 3 experiments.

We chose to plot AUC curve as it is used in classification analysis in order to determine which of the used models predicts the classes best. The final model selected gave the AUC as 99.72%.

Performance of final model on test data -

```
Confusion Matrix (vertical: actual; across: predicted)
         B  M    Error   Rate
B       52  1 0.018868  =1/53
M        3 35 0.078947  =3/38
Totals 55 36 0.043956  =4/91
```

Accuracy = 95.60%

We obtained the best accuracy from the final model as 95.6%

Comparison of 5 different models is summarized as follows:

	Test Accuracies		**Test Accuracies**
Model	**Breast cancer**		**Churn Data**
SVM(Linear Kernel)	0.9123	SVM(Polynomial Kernel)	0.7242
Decision Trees (Information Gain)	0.9064	Decision Trees(GINI)	0.7896
Boosting(GBM)	0.9532	Boosting(GBM)	0.7905
KNN	0.9200	KNN	0.7824
Artificial Neural Networks	0.9560	Artificial Neural Networks	0.8009

From above summary we can see that:

For dataset 1: We got the highest accuracy with Artificial Neural Networks with the parameters that we chose to build model on.

For dataset 2: Again, we got the highest accuracy with Artificial Neural Networks with the parameters that we chose to build model on. But the boosting and ANN yield almost same results.

- With the hyper parameters that we experimented yield the results that we documented. It is possible that with more experimentation of hyper parameters maybe we can get more accurate results
- Last but not the least more data is always going to help!