1ai) Supervised – we already have classes

1aii) Unsupervised – create classes

1aiii) Reinforcement – based on history ???????????(Supervised. Reinforcement learning is based on reward). doesn’t this seem like a regression problem?

1bi)



H(spam) = 1gen

IG(cash) = 0.1245

**IG(win) = 0.61** is the highest so should be root node of decision tree

IG(home)=0.035

IG(debt)=0.029

1bii) Why prune ? When overfit (or if we have time constraint and train for too long). Validation set – compare performance with & without pruning and keep the best

1ci)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| i | X(i) | Y(i) | D(x(i), x(q)) | wq |
| 1 | 1.5 | 3.16 | 2.7 | 0.3704 |
| 2 | 2.3 | 1.45 | 1.9 | 0.5263 |
| 3 | 3.0 | 1.07 | 1.2 | 0.8333 |
| 4 | 3.8 | 2.01 | 0.4 | 2.5 |
| 5 | 4.9 | 4.51 | 0.7 | 1.429 |

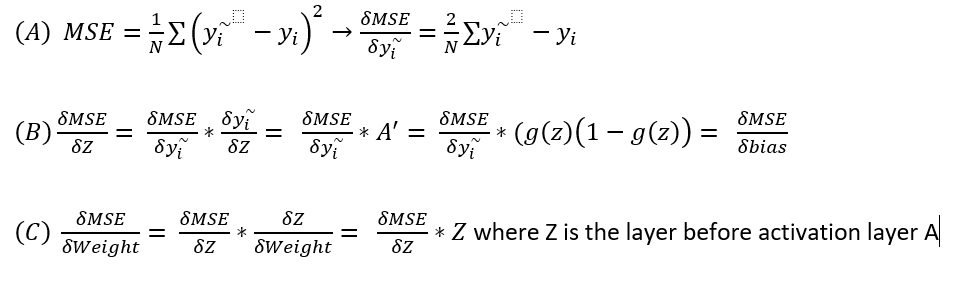
1cii) k=3 --> 3 closest neighbours are nodes 3,4,5

y(q) = 1/3(y(3)+y(4)+y(5)) = 2.53

1ciii) Same nodes,just add weight

y(q) = [y(3)w(3)+y(4)w(4)+y(5)w(5)] / [w(3)+w(4)+w(5)] = 2.596

2a) (not sure about this) seems right but mind double usage of Z

#

2b) Overfitting – model works very well only on the training dataset but cannot be generalised to the rest of the data

* 1. Dropout
  2. Early stopping
  3. Pruning
  4. More dat
  5. Decreasing size of Neural Netowrk
  6. L1 and L2 Regularisation

2ci)

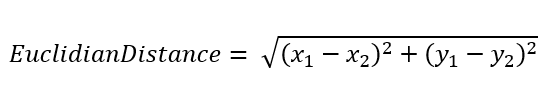
|  |  |  |
| --- | --- | --- |
|  | Predicted Real | Predicted fraud |
| Actual real | 8 | 1 |
| Actual fraud | 3 | 2 |

2cii) Accuracy = (8+2)/14 = 10/14

1. Fraud: precision = 2/3 , recall = 2/5, F1 = 1/2
2. Real: precision = 8/11, recall = 8/9, F1 = 4/5

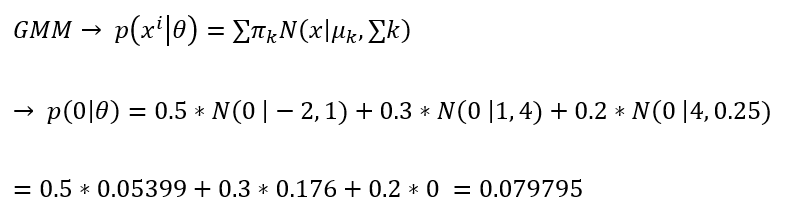
2ciii) Fraud has low recall high precision, real has high recall low precision --> too many emails are predicted as fraud

3a)



* D(x(i), µ(1)) = sqrt(2) --> cluster 1
* D(x(i), µ(2)) =~~6~~ sqrt(13)
* sD(x(i), µ(3)) =~~2~~ sqrt(5)

3b)



3c) Predicting the temperature for tomorrow, based on the weather today. Regression task (continuous values)

1. Activation function : Litnear
2. Loss function : MSE

Generating text by predicting the next word in the sequence based on the previous words – need 1 label from N classes

1. Activation function : softmax
2. Loss function: cross entropy

Detecting whether the camera image from a self-driving car contains a stop sign. Can only be yes or no, binary outcome

1. Activation function: sigmoid or tanh
2. Loss function: binary cross entropy

3d) Hyperparameters : specific to each machine learning model, what makes it “work”. E.G ‘k’ in KNN, max depth in a tree

10k datapoints - enough data, can do 80/10/10 split

1. Shuffle the data
2. K fold validation
3. Iteratively leave 1 fold for testing, for each HP set:
   1. Training + validation of k-1 folds
   2. ~~Hyperparameter tuning~~
4. Repeat for all validations | rotate the validation fold until entire dataset used
5. Aggregate & average accuracy
6. Get the best values on the left out fold each and get the best params for each split