COMP47750\_Exam\_2022\_23\_Spr

Question 1  
a. In k-Nearest Neighbour retrieval how can distances be calculated  
for Ordinal features?

We can assign each possible value for the ordinal feature a numerical value corresponding to their rank and then calculate the absolute difference. For example: {low,mid,high} get assigned values {1,2,3} so the distance d(low,high) = 3-1 = 2 and d(low,mid) = 2-1 = 1

b. With scikit-learn, when StandardScalar normalisation (also known  
as N(0,1) normalisation) is applied to data, what is the distribution  
of the data after normalisation?

The distribution will have a mean value of 0 and a standard deviation of 1 and it follows roughly a standard distribution.

c. If you have a collection of 3 green and 4 red balls and you add 2  
green balls to the collection what happens to the entropy of the  
collection?

It will increase, because a 5/9 to 4/9 split is closer to a 50/50 (maximum entropy) than a 3/7 to 4/7 split.

d. If we have two decision trees, one simple and one more complex  
that both perfectly explain the training data, which should we  
prefer? Briefly explain why.

We should always prefer the simpler tree. This is because we make less assumptions about the data and have a lower chance of overfitting. The more complex the tree gets the likelier it is to overfit.

e. When scoring the performance of classifiers what is the  
motivation for using balanced measures such as Balanced  
Accuracy Rate or Balanced Error Rate?

BAR and BER consider both the TP and TN rate (or FP and FN rate respectively). This means that even in highly skewed datasets (for example a 90/10 split between to classes) we don’t reach get a high accuracy by just predicting the majority class. Instead we also have to predict the minority class somewhat well to get a good overall score.  
  
f. Name two options for dealing with numeric (real valued) features  
in a Naive Bayes classifier.

Option 1: we discretise the feature into a set number of values. For example, temperature will be transformed into (cold, warm, hot)

Option 2: we assume the data follows a distribution (e.g. Normal distribution) and use the mean and standard deviation for that feature to calculate the probabilities   
  
g. Why is it not possible to use k-Means clustering with categorical  
data.

Categorical Data has no mean value which makes it impossible to calculate the new centroid of the cluster.  
  
h. Briefly describe one method to achieve diversity in an ensemble.

We can use Random Subspacing to achieve diversity. So instead of training each ensemble member on all features we only select a random subset of features for each ensemble member so that each member gets trained on different features.

Question 2  
a. Explain why training a multi-layer feedforward neural network is  
considerably more difficult than training a single layer network.

In a single layer network, we only have one layer of weights meaning we can easily calculate the errors and adjust the weights because we can directly see how an input contributes to the output. In multilayer network we have the hidden layers which abstract the contribution of the inputs and make it harder to calculate the errors.  
  
b. Even in a simple single layer Feedforward Neural Network the  
units (neurons) will have a fixed bias input. What is the reason for  
this bias input?

The function of the bias input is to make it harder for the neuron to activate, i.e. to have a certain threshold. Having a fixed bias input makes it possible to learn and adjust the threshold.

c. Explain the operation of the following components in the training  
of a neural network using gradient descent;  
i. Cost function  
ii. Weight update  
iii. Stopping condition

The cost function calculates the error rates of the output, i.e. how “far” we are from a perfect output. We can then use the Gradient Descent to minimize this cost function (i.e. get closer to a perfect output) and calculate how we need to adjust the weights. If we have calculated the new weights the weight update happens which update the old weights with the new ones to further minimize the error. This cycle continues until we reach a stopping condition, for example the error is below a certain threshold.   
  
d. What is the difference between stochastic gradient descent and  
batch gradient descent?

In Batch gradient descent we calculate the loss with all of our training values which can be quite ineffective for large datasets. In stochastic gradient descent we instead calculate the loss only with a singular training example. This makes the movement more random but still on average we still move downhill (for batch gradient descent we move steadily downhill, not just on average).  
  
Question 3  
a. When grid search is being used as part of a model selection  
process, explain the difference between random and exhaustive  
grid search. Mention one advantage of each strategy.

Exhaustive grid search performs a search over ALL possible value combinations. This has the advantage of always finding the optimal combination. Random search only looks at a subset of randomly chosen combinations. This makes it considerably faster than the exhaustive search, especially if the possible feature space is very large.  
  
b. Explain the difference between hyperparameters and ordinary  
model parameters using neural networks as an example.

Hyperparameters are parameters that have to be set manually at the initialisation of the model (e.g. number of neurons per layer, number of layers, activation function etc.). ordinary model parameters are learned by the model during training (e.g. the weights, bias etc.)

c. An important principle in evaluating machine learning models is  
that the test data should not be accessed in the model training  
process because it can result in unrealistic estimates of  
generalisation accuracy. This applies to the data preprocessing  
pipeline as well as the model fitting itself. In practice some  
scenarios are more serious than others; comment on the  
seriousness of each of the following scenarios:  
i. The training data is used during the feature selection  
process.  
ii. The training data is used to fit a One-Hot Encoder.  
iii. Missing values are replaced using the mean values for  
features across the training and test sets, i.e. the test data is  
used when the means are being calculated.

1. It is normal to use the training data during feature selection. However this should be the only data used for feature selection. The test data should never be used for this.
2. Since a one-hot-encoder only generates new binary features based on possible feature values it doesn’t matter if the training data is used for this.
3. The test data should not be used when imputing missing values since this can lead to overfitting. This is by far the most serious scenario out of all three.

Question 4  
a. Sample data from the synthetic half-moons dataset from  
scikit-learn is shown in the plot below. This is a tabular dataset  
with each point represented by two features, the x and y  
coordinates. The clusters found by k-Means clustering are shown  
(black and grey), k-Means has not been effective for finding  
clusters in this dataset, why is that?

k-Means assumes that all clusters are spherical. However for this dataset this is not the case, meaning it is impossible for k-means to find the correct clusters.  
  
b. In contrast, spectral clustering is able to uncover the correct  
clusters in this data.  
i. Explain in outline how spectral clustering can be applied  
to tabular data.  
ii. Explain why spectral clustering would work well on this  
data.

1. Spectral clustering can be applied to tabular data by calculating the Laplace matrix for the data and then calculating the eigenvalues and eigenvectors for it. The eigenvector with the second smallest eigenvalue then can be used to find the first two clusters (by looking at its values and finding where they switch from negative to positive). When more clusters should be considered we only have to consider the following eigenvectors (e.g for 3 clusters also the 3rd smallest eigenvector).
2. This data is good for spectral clustering, because in spectral clustering we try to minimize the “cut size”, meaning we try to separate clusters which have low interconnection. The data in this case has exactly that. The data points inside the clusters are very close together (thus very highly connected) while the points between the clusters are further apart and thus only loosely connected.

c. The half-moons data has a simple feature vector format where  
the features are the x and y coordinates. Explain how this could  
be converted into the affinity matrix format required for spectral  
clustering.

To convert this data we can create an affinity matrix where each data point is connected to its k Nearest Neighbours. Meaning we calculate the distances between the data points and mark them as connected if they are kNNs to each other.