Week 4 similarity-based Learning

Theory

1. What is meant by an N-dimensional feature space?

- We can treat the value of each feature as coordinate on one axis of an N-dimensional space.
- Although we can't graph dimensions higher than three, conceptually, an instance of N features can be represented as coordinates in an N-dimensional feature space.

2. What is a distance metric and what does it measure? Give an example.

- we can define our similarity measure between feature instances as a distance metric. As distance metric is some measure of the spatial distance between two points in space.
- When starting out with similarity-based modelling, it is common to choose a simple distance metric such as Euclidean distance or Manhattan distance, both of which are relatively easy to compute. Of these, Euclidean distance is the most popular and is based on Pythagorean ratios.

3. How does the Nearest Neighbor algorithm work? What is there no training step?

- We can make a prediction using Nearest Neighbor by considering an unlabelled feature we have not previously seen. To do this we calculate a distance metric between our unlabelled feature values and each of the features in our training data set.
- No training step which transforms the input data in any way is not a formal training step.

4. How does the K Nearest Neighbor (kNN) variant of this algorithm work and how is it an improvement?

- This is known as K Nearest Neighbors in which the algorithm considers not just one, but k neighbors as the discriminator for class and value predictions. When making a prediction for an unlabelled feature, we find the k nearest labelled features using the similarity metric and use their labels to determine the predicted class using a majority vote.
- One approach to improving the performance of Nearest Neighbors is to reduce the number of features which need to be considered in the prediction. A K-dimensional tree (called a K-D Tree) can be used to used to improve the performance from linear in N to logarithmic in N where N is the length of the dataset. The idea is that each sub-tree root node partitions the dimensions of the feature space in turn (say using a

median value of the axis to split the sets into two parts). The search space for predictions is dramatically smaller leading to faster runtime performance.

5. What is the difference in the way we make predictions between categorical and continuous target features?

- Categorical feature values should be contiguous in range, this is there should be no gaps in the class values.
- Continuous target features should be normalise in advance of performing predictions to smooth any differences between the value ranges on each axis.

6. Why is it recommended to normalise continuous feature values before using kNN

- This is because distance metrics can be sensitive to very high or very low values on one axis with respect to another access. Normalisation eliminates these magnitude differences by forcing all of the feature ranges into a common, constrained range.

7. The kNN algorithm performs poorly over large datasets. Explain one way we can improve its O(N) performance

- A K-dimensional tree (called a K-D Tree) can be used to used to improve the performance from linear in N to logarithmic in N where N is the length of the dataset. The idea is that each sub-tree root node partitions the dimensions of the feature space in turn (say using a median value of the axis to split the sets into two parts).

8. Explain why the choice of distance metric can be important when using similarity-based learning

- This is because distance metrics can be sensitive to very high or very low values on one axis with respect to another access. Normalisation eliminates these magnitude differences by forcing all of the feature ranges into a common, constrained range. This will improve the overall performance of the predictions on average.