Elias Elfarri - TDT4173 - HW1 - September 2021

1 Decision Trees

1.1 How does the algorithm work on a technical level and what kind of machine learning problems is it suited for?

The algorithm tries to build a tree recursively from the root up and branches out by splitting up the input data into left and right branch-nodes based on different conditions or "if-statements". The algorithm tries different splitting conditions and measures the success based on some sort of metric, where the ID3 uses entropy to measure information gain. Specifically subtracting the parent node from the created splitting nodes. The goal is to get leaf/end nodes in the tree that are "pure", or in other words, the leaf node should only represent one type of categorical data. This all happens in the fitting stage.

Decision Trees are suited for decision making predictions and an attractive algorithm to use if one wants to combine numerical and categorical data. It is fairly intuitive to set up and visualize. Also compared to deep learnings "blackbox-ness", decision trees is a very transparent algorithm and could be suited for scenarios where explainability is important. The trade off is how ever that the accuracy might not be as high as some other more blackboxed methods.

1.2 What is its inductive bias, i.e., what assumptions does it make about the data in order to generalize?

At the beginning all the data is at the root node. It is assumed that the data is categorical and if it is numerical it needs to be preprocessed to fit this assumption. Nodes are considered leaf nodes when the entropy is 0. It also assumes that all categorical features are somehow related to eachother in providing the answer we are trying to predict.

1.3 What happens in the second dataset that makes it harder than the first and how does this problem relate to the algorithm's inductive bias?

We have "Zodiac Signs" as part of predicting business venture success and the Decision Tree assumes that this is an important part of predicting business success or failure. This is how ever not true and that category is not relevant information for the prediction of business success, resulting in volatile as well as poor prediction accuracy.

1.4 What modifications did you do to get around this problem?

Drop features that are not relevant for what we are trying to predict. Obviously the Zodiac sign of the founder does not have anything to do with the probability of success of their business venture. This increases accuracy from around $\sim 70\%$ to $\sim 85\%$ validation and test accuracy.

2 K-Mean Clustering

2.1 How does the algorithm work on a technical level and what kind of machine learning problems is it suited for?

K-means clustering is an unsupervised learning algorithm that tries to identify categories/clusters in a dataset. Based on how many K clusters you want to identify, the algorithm creates K centroids and tries to minimize the euclidean distance between itself and an identified neighbouring cluster. It is a very intuitive algorithm in that sense. It is a very good algorithm to use to confirm or try to identify groups that exist in a dataset,and is in a sense very self sustained as it is does not need ground truth labels (hence unsupervised).

2.2 What is its inductive bias, i.e., what assumptions does it make about the data in order to generalize?

K-means clustering assumes that the dataset has identifiable clusters in the dataset in order to identify them.

2.3 What happens in the second dataset that makes it harder than the first and how does this problem relate to the algorithm's inductive bias?

In the second dataset the data is not as distanced from each other as in the first dataset such that randomly initializing centroids will result in randomly selected clusters each time, where most of those identified clusters are wrong.

2.4 What modifications did you do to get around this problem?

Kmeans++ initialization helped remove some of the randomness of the initialization such that only the first centroid is randomly initialized and then the K - 1 remaining centroids are selected based on being the point farthest away from its nearest centroid as the next initialized centroid. That way each centroid gets initialized in an identifiable cluster.

3 Logistic Regression

3.1 How does the algorithm work on a technical level and what kind of machine learning problems is it suited for?

Logistic regression uses a sigmoid or some other type of activation function to make predictions and then it minimizes the cost function cross entropy function with respect to the bias and weights, where weights and biases are updated after Stochastic Gradient Descent steps. This is the boilerplate forward/backstepping method used in deep learning.

3.2 What is its inductive bias, i.e., what assumptions does it make about the data in order to generalize?

The inductive bias of this method is that it assumes that the classes of data can be seperated by a linear boundary.

3.3 What happens in the second dataset that makes it harder than the first and how does this problem relate to the algorithm's inductive bias?

The dataset is nonlinear which makes it hard for the logistic regression algorithm to give any good results.

3.4 What modifications did you do to get around this problem?

Feature transformation of the data to a polynomial maps it into a coordinate system where the data is constrained by a linear boundary, this dramatically increases the accuracy from around 40% to 90% accuracy.