Machine Learning and Case-Based Reasoning
TDT4173 - Assignment 1

Øyvind Aukrust Rones, NTNU

January 30, 2018

1 Theory

1. Concept learning seeks to classify concepts by looking at a set of common features. For example, one can try to classify a given assignment as doable by looking at the amount of exercises, time until deadline and difficulty of the content.

2. Function approximation is the action of finding a function that approximately represents a series of data points from a given problem. Finding a function that describes the problem may make it possible to predict future outcomes.

3. Inductive bias makes assumption about a process so that unobserved samples can be classified. Without inductive bias the learner can not generalize beyond the observed samples. The candidate elimination algorithm assumes that the target concept can be represented by a conjunction of attribute values. Decision trees assume that the classes of the data is grouped together.

4. Overfitting is a result of poor generalization, where a model is too exact for a given dataset. Thus, new data points added to the set may not be properly represented. In other words, the model captures a trend in the data that can not be justified. Underfitting, on the other hand, occurs when the model fails to capture the trend of the data.

   A validation set is a subset of the available data used to detect overfitting, and is used independently of the *training set* to verify the current model.

   Cross validation uses several models concurrently, based on different partitions of training and validation set. The result from the models are inferred by averaging the result of every model, thus, although one partition of the data set may yield an overfitted model, the contributions of the models that are not, help mitigate this problem.

5.

Initially:

$$S_0 = < \emptyset, \emptyset, \emptyset, \emptyset >$$
$$G_0 = <?, ?, ?, ? >$$

Iterating through each sample:

$$S_1 = < Female, Back, Medium, Medium >$$
$$S_2 = < Female, ?, Medium, ? >$$
$$G_1 = <?, ?, Low, ? >$$
$$S_3 = <?, ?, ?, ? >$$

S_3 yields a hypothesis that will always be as general as any G. The Candidate Elimination algorithm does not work for the given samples.

2 Programming

2.1 Linear regression

1. Given n inputs x_i and n corresponding outputs y_i, the following is found:

$$\mathbf{X} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ & \vdots \\ 1 & x_n \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

The outputs are then approximated by $\mathbf{Xw}$, where $\mathbf{w}$ is a set of weights that will be estimated.

Estimating the weights that minimize the mean square error is easily done by applying the Moore-Penrose pseudo inverse:

$$\mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \tag{1}$$

2. Applying the method described above to the two dimensional datasets yields the weights $\mathbf{w} = \begin{bmatrix} 0.2408 & 0.4816 & 0.0586 \end{bmatrix}^T$. The mean square error is found to be 0.0095 and 0.0104 for the test and training set, respectively. Thus, the MSE is actually smaller for the test set, indicating that the model generalizes very well.

3. The weights are found as $\mathbf{w} = \begin{bmatrix} 0.1956 & 0.6129 \end{bmatrix}^T$. Figure 1 shows the training data and resulting function approximation together. There are few outliers in the data and the function approximates the data well.

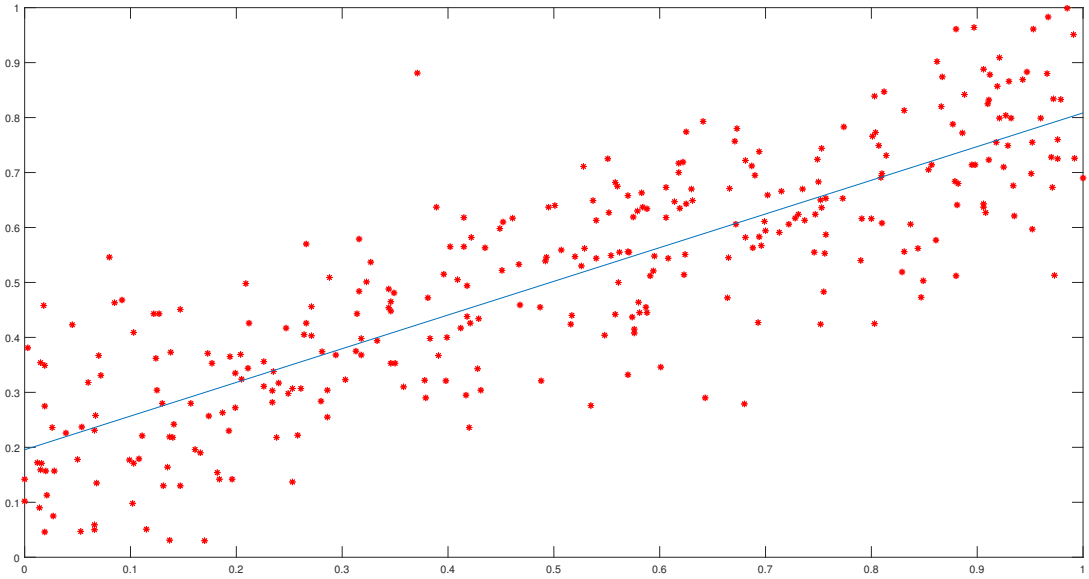

Figure 1: Estimated function together with the training data

2.2 Logistic regression

1. Figure 3 clearly shows the training data to be linearly separable as the decision boundary splits the positive and negative samples into two different sets. Figure 2 shows the cross entropy of the training and test set. The model does a decent job of generalizing although a single point in the test set is always misclassified regardless of learning rate and number of training iterations. This probably means that the training and test set together isn't linearly separable. These results were achieved with a learning rate $\eta = 0.1$ and initial weights equal to zero.

2. Figure 4 shows the training data of the second set. It is evident that the positive and negative samples each follow a circular pattern and is not linearly separable. This is also indicated in figure 5, where the estimated decision boundary is plotted with the training data and is in no way successful of partitioning the samples. Clearly all the training data is classified as positive in this case.

   The trend of the data can be taken advantage of by transforming the coordinates to a new space that may be easier to handle. Figure 6 shows the training data after employing a kernel function along with the accompanying decision boundary. The kernel function transfers the set to a one dimensional, linearly separable space. This is done by defining the new coordinate as the distance from the center of the data, $(0.5, 0.5)$.

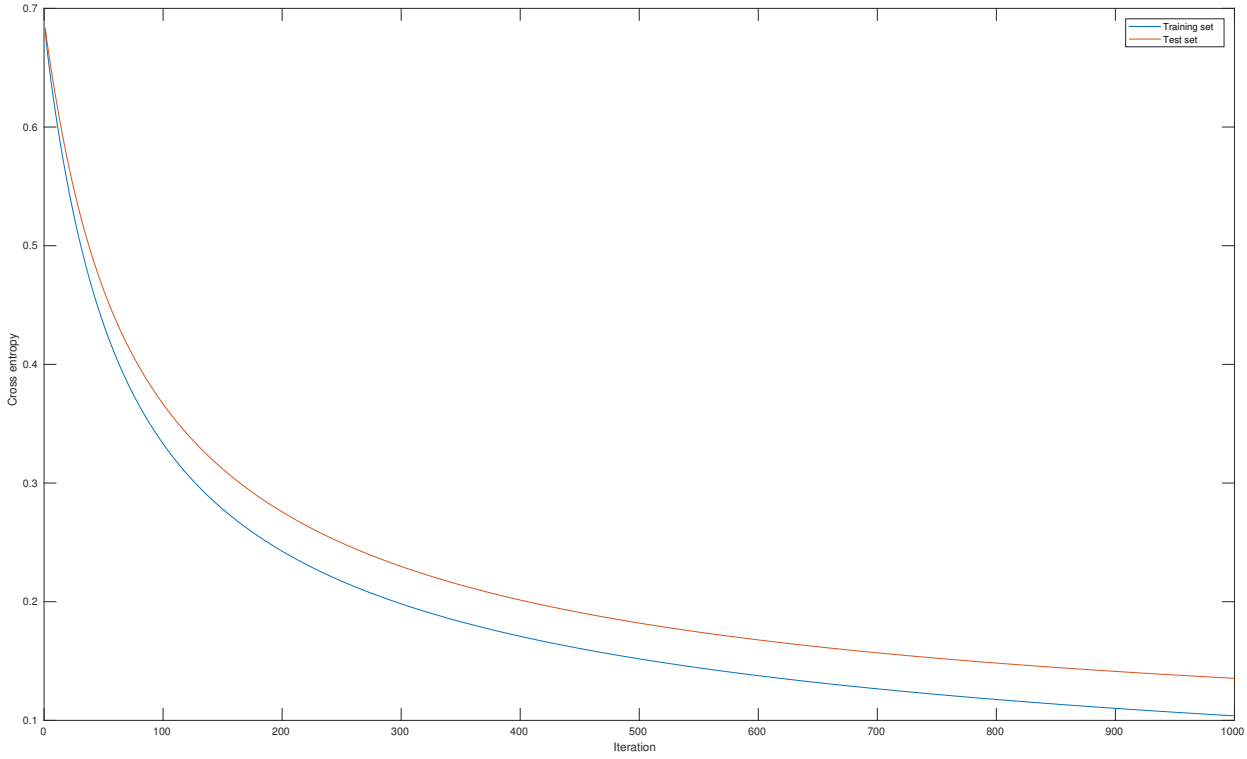

Figure 2: Cross entropy of training and test set

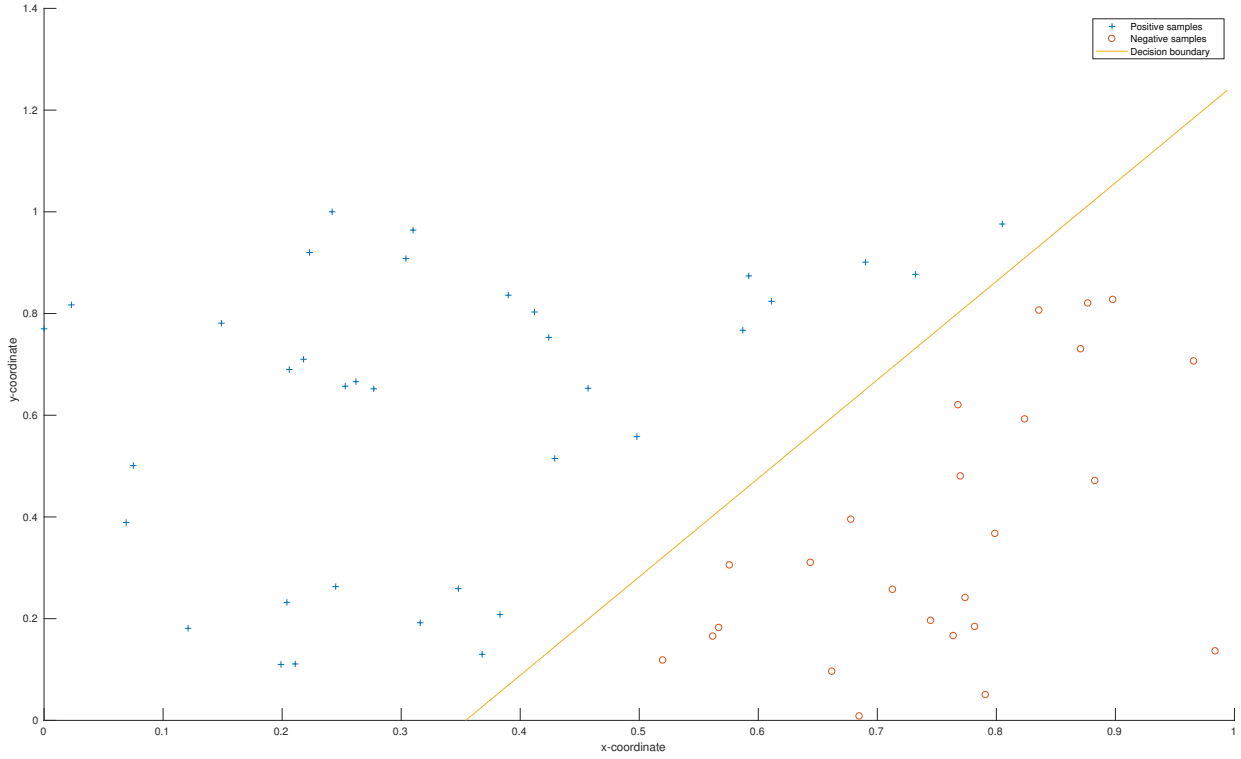

Figure 3: First training data set with decision boundary

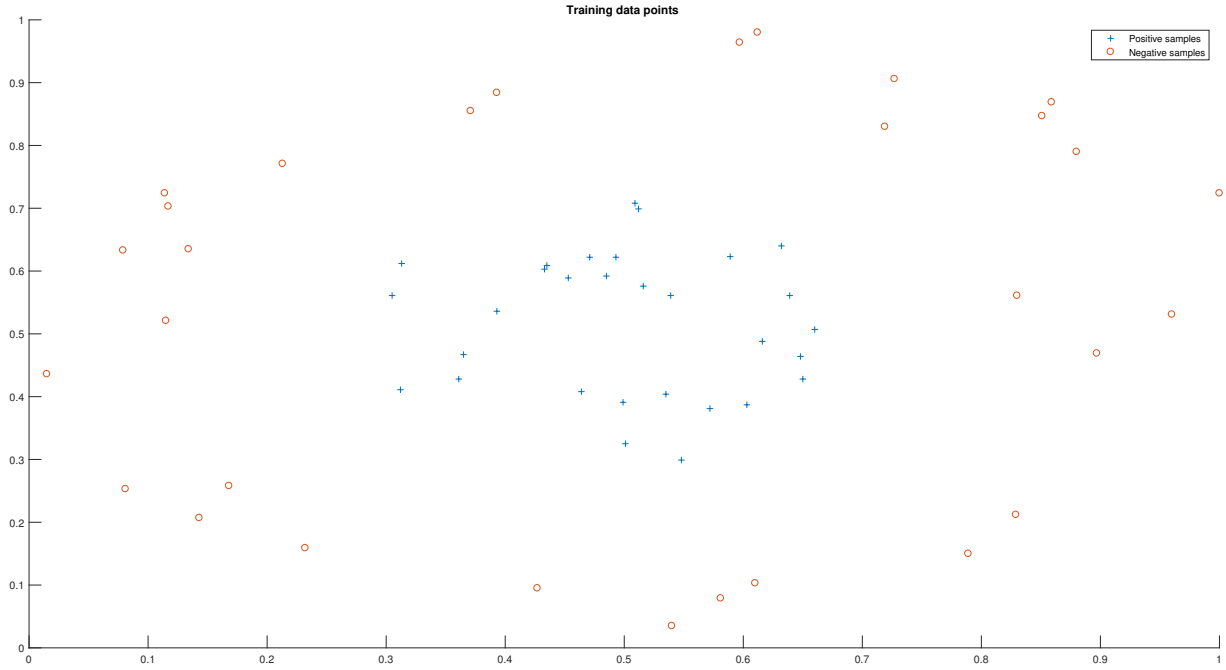

Figure 4: Second training data set

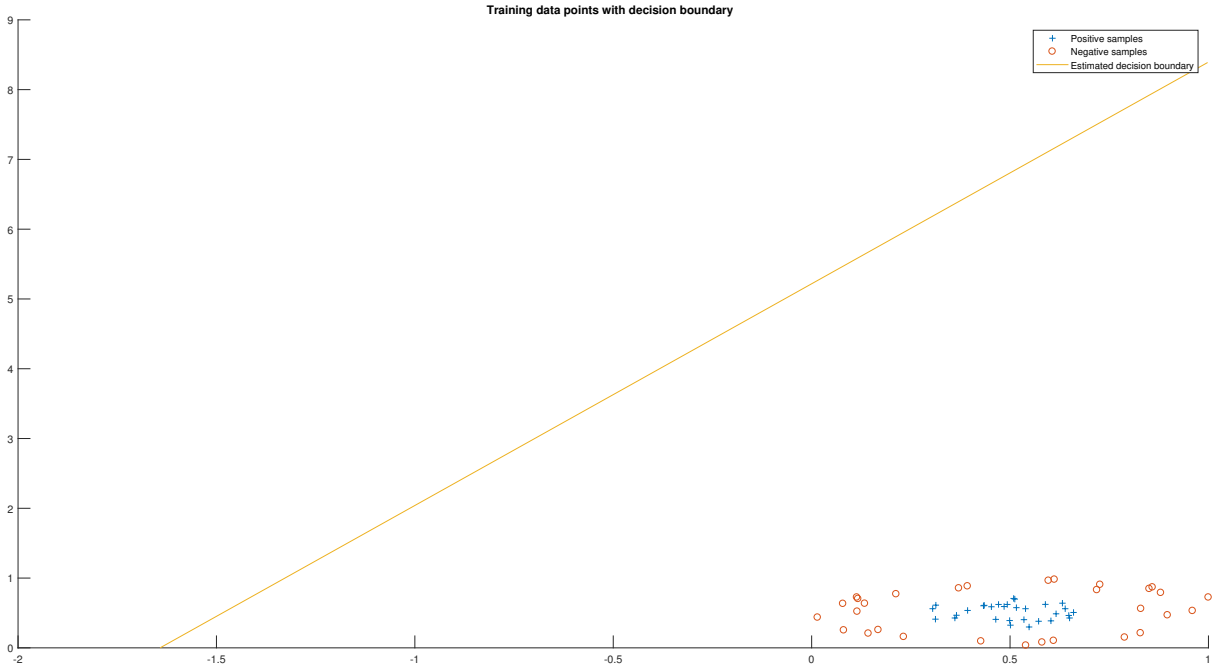

Figure 5: Second training data set with estimated decision boundary

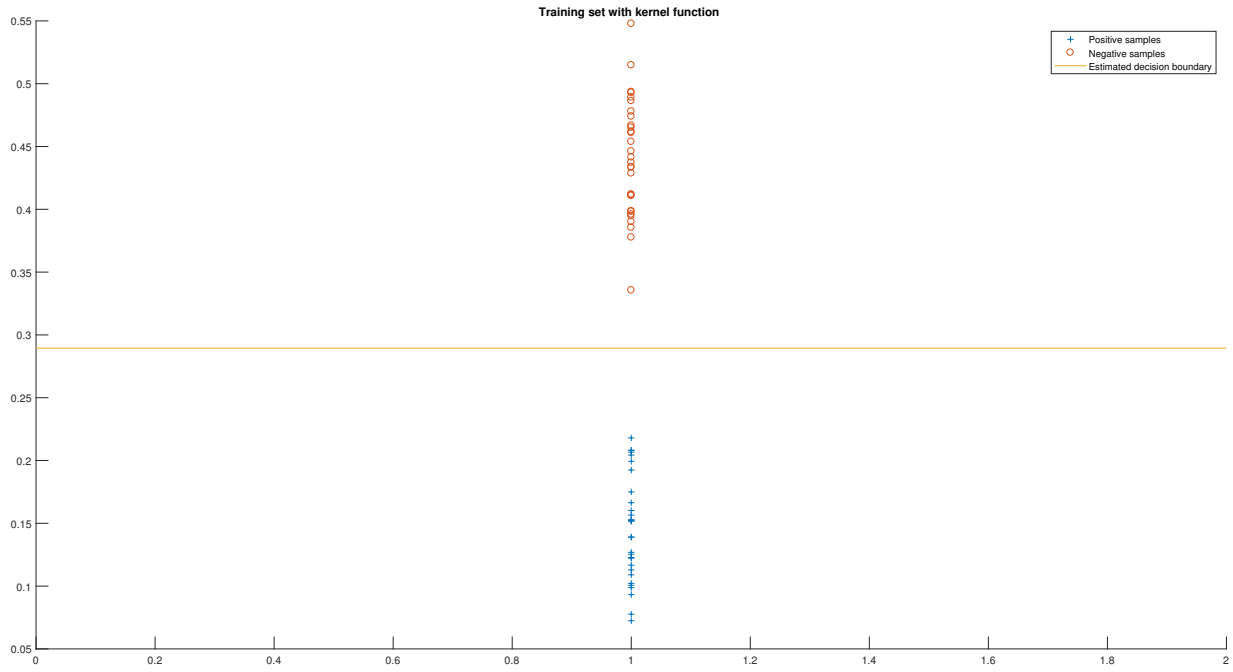

Figure 6: Second training data set with estimated decision boundary after employing a kernel function