Lab Course Machine Learning and Data Analysis
Problem Set 3

Benjamin Pietrowicz - Matriculation Number: 332542

June 20, 2014

Part 1: Implementation

In this part I used the handbook pdf provided on ISIS to implement the algorithms.

Assignment 1: Crossvalidation

This was a straight forward method to implement, as it was nicely documented in the handbook. The only differences were the application of the kernel ridge regression in the cross validation and the computation of the estimation of needed time. But this was done very easily.

Assignment 2: Kernel Ridge Regression

Kernel Ridge Regression was a bit harder to implement. What was not clear from the beginning is, that after fitting the model the prediction yields another kernel than the fitting. The Kernel in the prediction is computed by the training and test data, and not only on the training data like in the fitting. But the α computed in the fitting uses the predictions kernel, meanining it uses only the training data, to yield a weighting of the training data. Additionally the leave one out cross validation was very hard to grasp as an idea. The fact that we had to use logarithmically spaced candidates around the mean of the eigenvalues of the kernel matrix K for c in computing the minimal error was not very easy to understand. What would have been easier to understand, is that saying once regularization is set to zero as a flag, a new c has to be determined. The problem was that setting the regularization to zero was somehow final to me and the formula for α did not make sense for me because the c was ultimately set to zero. Fortunately the formulas given in the handbook for the LOOCV were straight forward to implement.

Part 2: Application

Assignment 3

For the empirical part we can simply draw samples from the given distribution

$$P(x) = P(x|y = -1) \cdot P(y = -1) + P(x|y = +1) \cdot P(y = +1) \ .$$

Once we have the samples with their labels, we can sort the samples according to their values. We then take a look at the labels and receive and ordering of the labels. Everytime there is a -1 we move in the graph from the origin into the positive y direction, and everytime there is a +1 we move into the positive x direction. Of cource we have to scale correctly, such that the ROC is between 0 and 1 in both x and y direction.

For the analytical part we make use of the true known distributions. We can use the cumulative probability function Φ to calculate the FPR and TPR for n thresholds. We here go from the lower $\mu - 3$ to the higher $\mu + 3$, because more than 99% of the data lies within $\mu \pm 3\sigma$ and $\sigma = 1$.

We then can calculate the FPR and TPR, because we can interprete the linear classifier as cumulative probability:

$$TN(z) = \Phi(z - \mu_N) \qquad FP(z) = 1 - TN(z) = \Phi(\mu_n - z)$$
$$FN(z) = \Phi(z - \mu_P) \qquad TP(z) = 1 - FN(z) = \Phi(\mu_p - z)$$
$$TPR(z) = \frac{TP(z)}{TP(z) + FN(z)} \qquad FPR(z) = \frac{FP(z)}{FP(z) + TN(z)}$$

For a small sample size we can get very nice results for the empirical curve, but also suboptimal curves as depicted in Figure 1. When we take more samples

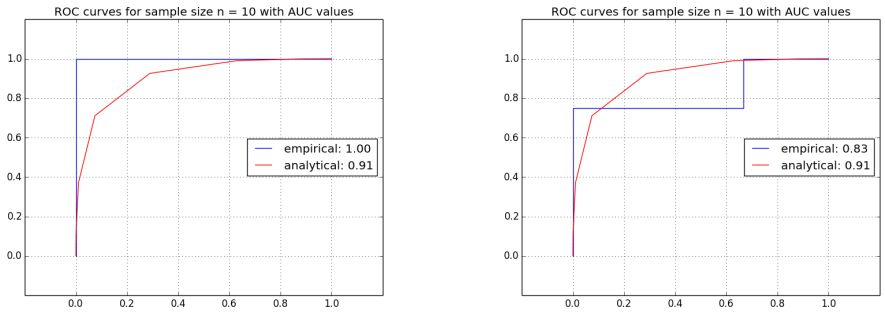

Figure 1: ROC curves for sample size n = 10

with n = 100, we can see that the empirical curve moves closer to the analytical curve, what can be seen in Figure 2 for both better and worse results. When

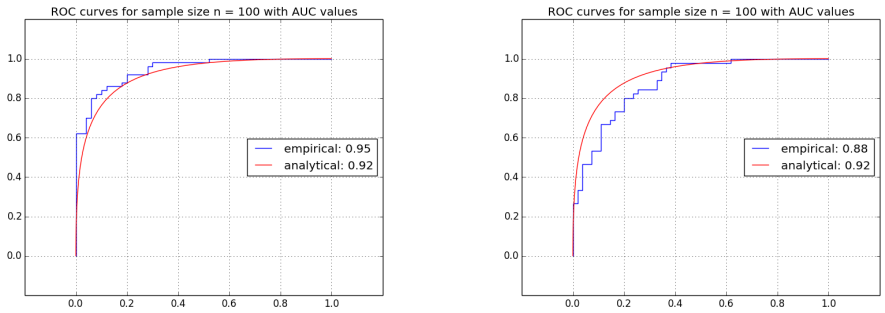

Figure 2: ROC curves for sample size n = 100

we then take even more samples (Figure 3), i.e. n = 1000 or even n = 10000, the empirical curves almost overlap with the analytical curve and the AUC is the same.

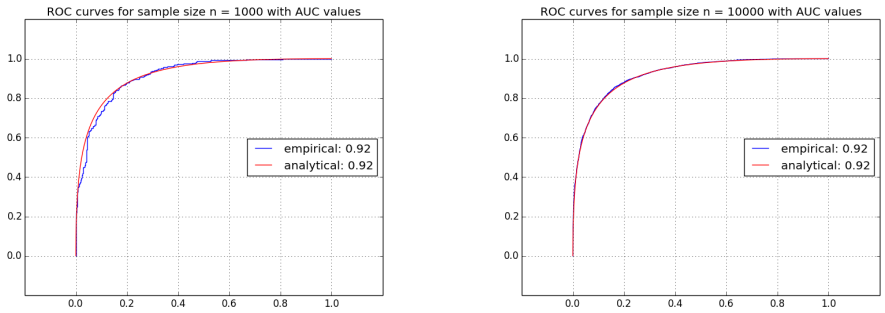

Figure 3: ROC curves for sample size n = 1000 and n = 10000

Assignment 4

Since executing the big for loop over all the data sets takes too much time and sometimes ended in throwing an exception due a a singular matrix error, I ran the code over all datasets individually and stored them via ipython into the file.

It was not clear to me, what had to be done to plot the ROC-curves by properly using the cross-validation. When calculating TPR and FPR we need the true and the predicted labels for the data sets. ALthough the prediction takes a lot time and we receive the predicted labels, we cannot compare them to the real labels. That is why I could not plot ROC curves for the datasets in terms of the test sets.