Assignment 6

Statistical Machine Learning

Sebastian Bordt / David Künstle / Martina Contisciani / Nicolò Ruggeri /
Rabanus Derr / Prof. Ulrike von Luxburg

Summer term 2020 — due on **June 11th at 14:00**

Please take a minute to answer our quick survey: https://forms.gle/kVKmR9nuHwQHfUhf6

Exercise 1 (Treasure hunt! 0.5 points per question + Treasure) The following statements are either true (`T`) or false (`F`). Answering all questions results in a word of length 10 over the alphabet {`T`, `F`} (e.g. `theword=TFTFTTFTFF`). Visit the website

`www.tml.cs.uni-tuebingen.de/teaching/2020_statistical_learning/`*theword/*

to obtain the GPS coordinates of where to find a treasure in Tübingen! Take a stroll when the weather is nice, walk to the treasure and pick a reward from the treasure. There is a reward for the first 70 visitors to the treasure ...

(1) Given an ERM approach, there always exists a corresponding probabilistic model.

(2) Linear regression can not overfit.

(3) The number of weights in linear regression increases with the number of data points.

(4) The lasso regression can perform variable selection.

(5) L2 regularization can ensure that the objective has a unique solution.

(6) Logistic loss punishes incorrectly classified examples at a less-than-linear rate.

(7) When the prior distributions in the Bayesian framework follows a normal distribution, Bayesian maximum a posteriori gives the same result as maximum likelihood.

(8) The support vector of a linear SVM is the direction of the decision boundary in weight space.

(9) In the dual hard margin SVM optimization problem, data points appear only via inner products.

(10) If the parameter of the Gaussian kernel σ is very small, the kernel support vector machine is going to underfit.

Exercise 2 (Building new kernels, 1+1+1 points) Assume that $k_1, k_2 : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ are kernel functions. Are the following functions kernel functions, too? Prove your claim.

(a) $k = \alpha_1 k_1 + k_2$ for $\alpha_1 \geq 0$

(b) $k = k_1 - k_2$

(c) $k(x, y) = f(x)k_1(x, y)f(y)$ for any function $f : \mathcal{X} \to \mathbb{R}$

Exercise 3 (Kernel SVM, 1+1+2+0+0 points) Download the dataset `train.csv` that you can find on our website.

(a) Demonstrate that a kernel SVM with a Gaussian kernel is powerful enough to interpolate the training data (achieve zero training error). What is the range of hyperparameters γ and C for which interpolation happens?

Hint: Look at a grid of hyperparameters.

(b) Assuming that the kernel matrix is of full rank, can kernel SVM achieve zero training error on any dataset? Explain.

(c) Now use cross-validation to obtain reasonable values for all hyperparameters. What is your training error with respect to the 0–1-loss? Download `test.csv` from our website. What is your test error with respect to the 0–1-loss?

(d) Use the provided code to plot the decision boundary of your classifiers together with the datapoints.

(e) Play around with different kernels and hyperparameters. Use the provided code to have a look at the decision boundaries.

Hint: For this exercise, use `sklearn.svm` and GridSearchCV.

Exercise 4 (Cross-validation, 1+2+2+1+1+1) In the last inverted lecture there was a lengthy discussion about the choice of the parameter k in cross validation. Let's look at it more closely now. We prepared two datasets that you can download from our website as `cvtrain.csv` and `cvtest.csv`. The training dataset contains 400 observations of 12 numerical features and a binary outcome. We want to use logistic regression with L2-regularization to predict the binary outcome. Thus, we need to choose a single hyperparameter.

(a) We start with a simplified scenario of validation (without "cross"): For $k \in \{2, 3, 5, 10\}$, use $\frac{100(k-1)}{k}$% of `cvtrain.csv` for training and the rest for validation. Train the model for different choices of the hyperparameter and evaluate the performance on the validation set. Which k performs best? Then evaluate the performance of your model on the test set.

(b) Thinking about it, there are many different ways of how you might split `cvtrain.csv` into training and validation set. This is curious: What might be the effect of choosing a different split on model performance? Consider both mean and variance of the estimator that selects the model based on a random train/val split (we are interested in the validation and the test error). Perform a Monte-Carlo simulation to estimate these quantities for $k \in \{2, 3, 5, 10\}$.

(c) Now we turn to k-fold cross-validation. Think about how cross-validation relates to simply using a single validation set. For cross-validation, the result depends on the parameter k and the folds that you choose. For $k \in \{2, 3, 5, 10\}$, perform a Monte-Carlo simulation as in b). Before seeing the results: What do you expect?

(d) How would you describe the influence of the parameter k in this example?

(e) Is it fair to directly compare the procedures from b) and c)? Argue in terms of computational costs: How many models does each method need to train?

(f) For $k = n$, k-fold cross-validation is called leave-one-out (LOO) estimation. For LOO, what is the variance due to the splitting of data? In this example, does LOO perform better or worse than the best k found in c)?

Hint: GridSearchCV has an argument `cv`. Use ShuffleSplit and KFold. If confused, have a look at `https://scikit-learn.org/stable/modules/cross_validation.html`

Exercise 5 (Pentecoast Prediction Competition, Optional, 3x10, 7x5, otherwise 3 Bonus Points) This exercise is a machine learning competition! It is a bonus exercise, you do not have to participate. However, participation is highly encouraged. The idea is that you experience the process of machine learning first-hand by yourself. The best submissions can receive up to 10 bonus points. All students who take part in this competition receive 3 bonus points. The machine learning problem and competition details are as follows:

A group of biologists has laboriously collected a gene expression dataset. Unfortunately, half of the expressions of gene `GXYLT1` are lost. The biologists are heartbroken. You offer some relief by predicting the lost gene expressions. The biologists tell you that

$$\min\left\{1, \frac{|y-\hat{y}|}{50+y}\right\} \tag{1}$$

is a meaningful measure of loss for this application (where y is the true gene expression level, and you predict $\hat{y}$).

- Download the gene expression dataset `geneexp.csv` from our website. This dataset contains 900 measurements of 1000 different genes. A gene expression is a non-negative number. The lost expressions of gene `GXYLT1` are indicated by the value `-1`.
- Predict the missing gene expressions, replace the value `-1` with your predictions and submit the modified `geneexp.csv`, together with a short description of your solution and all code to reproduce your results to `sml2020competition@gmail.com`. The three best submissions (according to the test error with loss function (1) on our held out data with the true gene expressions) receive 10 bonus points, the next seven best submissions 5 bonus points and all other meaningful submissions 3 bonus points. The deadline of the competition is the deadline of this assignment. Have fun!
- You can use all machine learning methods and programming tools that you know about.
- Here are some things that you might want to do and think about:
    - Perform some exploratory data analysis. What does the distribution of features and outcome look like? Perhaps you should pre-process this dataset before you apply machine learning.
    - You might want to separate the data into a training and test set, or use cross-validation.
    - You should not spend too much time theorizing what might be the best machine learning method for this problem - get your hands dirty and try something! This will give you a first impression of what can be achieved. After you have this baseline: How can you improve?
    - How do you want to deal with the unusual loss function? At first, you might simply use another loss, like mean-squared error, and then see how it relates to (1). In a more advanced step, think how the loss can be incorporated more directly.

Hint: You can assume that the gene expressions are missing at random. You can also use our forum on Ilias in order to discuss this problem - but not to exchange solutions.