Exercise Sheet #12

Problem 1 (*Support Vector Machine*) 15 Pts

The Iris Flower Dataset (link) is a classical example of real-world data that can be used as a toy example for classification algorithms. The data set consists of 50 samples from each of three species of Iris (Iris setosa, Iris virginica and Iris versicolor). Four features were measured from each sample: the length and the width of the sepals and petals, in centimeters. Two of the three classes are linearly separable.

(a) Write a program that trains a SVM in arbitrary dimensions. You can take the code given in thelecture on support vector machines as a starting point. (5 pts)

(b) Download the dataset from the link. Write a function that loads the file and writes the features into three $4 \times N$ arrays, where three is the number of classes (Iris setosa, Iris virginica and Iris versicolor), four the number of features and N the number of samples in the dataset. (3 pts)

(c) By plotting the third and fourth feature against each other, you can observe a high correlation. Discarding either one of these features will not lead to a significant loss of information. Therefore, ignore the fourth feature for classification.

Run `svmGradientAscent()` on each three pairwise combinations of classes. How can you tell that linear separation does not work for one of these pairs? Compare the lagrangian multipliers you get in the separable/non-separable case. What differences do you observe and why? (4 pts)

(d) Make a 3d-plot showing the data in feature space (omitting the fourth feature) and the separating planes determined by `svmGradientAscent()`. You should extend `xyValuesGnu()` to three dimensions and use `splot` instead of `plot`. For the plane, you can use a command of the form `splot a*x + b*y + c`. (3 pts)

Problem 2 (*Detailed balance*) 5 Pts

The Metropolis algorithm is widely used for sampling from complex probability distributions P, e.g. the Ising model. The commonly used algorithm starts with a random state x and iteratively samples other neighbouring states according to the following steps:

- Pick a random state x.

- Choose a new target state x' according to a proposed probability $Q(x'|x)$, which can be for example a uniform distribution over neighbours.
- Move to the new state x' with probability

  $$\alpha(x', x) = \min(1, \frac{P(x')Q(x|x')}{P(x)Q(x'|x)}), \tag{1}$$

  otherwise stay in state x.
- Repeat the last two steps many times until sample size is sufficient.

Note that the probability P does not have to be normalized, allowing for example to set it to $e^{-\beta H(x)}$ for thermodynamic systems.

The success of this model hangs on the fact that detailed balance is conserved, but this is not directly evident from the steps above. Prove that detailed balance is maintained for large sample sizes.

(a) Write an expression for $T(x \to x')$, the probability to transition from **current** state x to state x'.

(b) Show that detailed balance is conserved, i.e. the probability to transition from **any** state x to x' is equal to the reverse:

$$P(x)T(x \to x') = P(x')T(x' \to x) \tag{2}$$

Hint: Notice that $\min(x, y) = \min(y, x)$.