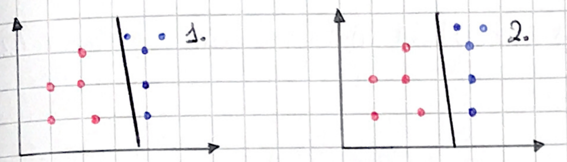
EXAM 12.02.18

Question 1

In supervised learning the dataset is $D = \{(x_i, y_i)_{i=1}^{N}\}$. To each sample of the input set we have an associated label in the dataset. In unsupervised learning the dataset is $D = \{x_n\}$, so in this case we don't have labels but only input values.

Question 2



The plot number 1 represents the solution obtained with perceptron, the plot number 2 with SVM. The difference between SVM and perceptron is that SVM always aims at maximum margin with the better accuracy, while perceptron is based on a sequential algorithm that depends on the learning rate η. Is possible to have different solutions with different ηs but is also possible that we don't have a solution.

Question 3

1). If $C = c_1$ AND $B = b_1$ then NO (1)
If $C = c_1$ AND $B = b_2$ then YES (2)
If $C = c_2$ AND $A = a_1$ then YES (3)
If $C = c_2$ AND $A = a_2$ AND $B = b_1$ then YES (4)
If $C = c_2$ AND $A = a_2$ AND $B = b_2$ then NO (5)
If $C = c_2$ AND $A = a_3$ then NO (6)
If $C = c_3$ then NO (7)

2). T is consistent with S_1 because of (1).
" " " " " " " (4).
" " " " " " " (7).
" " " " " " " (2).

Question 4

1). Boosting is an ensemble method. Instead of training one single complex classifier we train different classifiers (in this case in sequence) and we combine their results. The main idea of boosting is that we "specialize" a layer on the error of the previous layer. We can do this using weights, errors are weighted with very high weights while correct predictions with very low weights.

2). $E = \exp[-t_n f_M(x_n)]$ with $f_M(x) = \frac{1}{2}\sum_{m=1}^{M} \alpha_M y_M(x)$

$$E = \sum_{n=1}^{N} \exp\left[-t_n f_{M-1}(x_n) - \frac{1}{2}\alpha_M y_M(x_n) t_n\right] = \sum_{n=1}^{N} w_n^{(M)} \left[\exp -\frac{1}{2}\alpha_M y_M(x_n) t_n\right]$$

$$y_M(x) = \operatorname{sign}\left(\sum_{m=1}^{M} \alpha_M y_m(x)\right)$$

Question 5

1). This is a regression task. We can use CNNS to solve this problem.

$$f(x, \theta) = f^{(3)}(f^{(2)}(f^{(1)}(x, \theta^{(1)}), \theta^{(2)}), \theta^{(3)})$$

2). For the hidden units we can use Relu, for the output units we can use the identity activation function $y = w^T h + b$.

3). The error function that we can use is Mean Squared Error (MSE).

$$E(\theta) = \frac{1}{N}\sum_{n=1}^{N} (t_n - f(x, \theta))^2$$

Question 6

1). The gram matrix is $K = XX^T$. If we have a model $y(x, \alpha) = \sum_{n=1}^{N} \alpha_n x_n^T x$ with a kernel function $k(x, x') = x^T x' \Rightarrow K = \begin{pmatrix} x_1^T x_1 & x_1^T x_N \\ x_N^T x_1 & x_N^T x_N \end{pmatrix}$

If we have a model $y(x, \alpha) = \sum_{n=1}^{N} \alpha_n k(x_n, x)$ with a generical kernel function $k(x, x')$

$$K = \begin{pmatrix} k(x_1, x_1) & k(x_1, x_n) \\ k(x_n, x_1) & k(x_n, x_n) \end{pmatrix}$$

2). The error function becomes $E(w) = \sum_{n=1}^{N} E_2(y_n, t_n) + \lambda \|w\|^2$ with $E_2(y_n, t_n) = (y_n - t_n)^2$. If we use $\alpha = (XX^T + \lambda I_n)^{-1} t$ and $X = (x_1 \dots x_n)^T$ we can obtain a solution: $y(x, w^*) = \sum_{n=1}^{N} \alpha_n x_n^T x$. Now we apply the kernel trick:

$$y(x, w^*) = \sum_{n=1}^{N} \alpha_n k(x_n, x)$$