EXAM 12.02.19 A

Question 1

1). In a Reinforcement Learning problem we have a dataset $D = \{x_1, a_1, \dots x_n, a_n, r_n\}$ and we want to find an optimal policy function maximizing the reward. In a RL problem we don't have outputs, while the input is a collection of triples state, action and reward.

2). One possible example of RL algorithm is Q-learning:

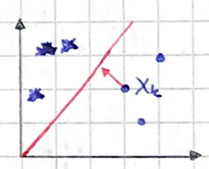
foreach x we initialize $\hat{Q}_{(0)}(x,a) \leftarrow 0$
we observe the current state
foreach time $t = 1 \dots T$
  we select an action a
  we execute the action
  we observe the new state x'
  we collect the immediate reward
  $\hat{Q}_{(t)}(x,a) \leftarrow \bar{r} + \gamma \max_{a' \in A} \hat{Q}_{(t-1)}(x', a')$

$\pi^*(x) = \arg\max_{a \in A} \hat{Q}_{(T)}(x,a)$

Question 2

- Dropout: we remove randomly some units of our network with probability α (Dropout probability), in this mode with a reduced number of parameters we can avoid overfitting.
- Early stopping: we stop the iterations early to avoid overfitting. We can stop for example when the performance on the validation set of our model starts to degrade.

Question 3



1) Let x_k be the closest point to the separation surface $h = \bar{w}_0 + \bar{w}^T x = 0$. We define the margin as $\frac{|y(x_k)|}{||w||}$. SVM aims at maximum margin with the better accuracy. We can define the margin as $\min_{n=1 \dots N} t_n [w^T x_n + w_0] \cdot \frac{1}{||w||}$. So we want to find with SVM $w^*, w_0^* = \arg\max_{w_0, w} \frac{1}{||w||} \min_{n=1 \dots N} t_n [w^T x_n + w_0]$. Scaling we are not affecting our solution. We obtain $t_k y(x_k) = 1$ and $t_n y(x_n) \geq 1 \ \forall n = 1 \dots N$.

In this mode $w^*, w_0^* = \arg\min_{w_0, w} \frac{1}{2} ||w||^2$. Now we move to a Lagrangian domain.

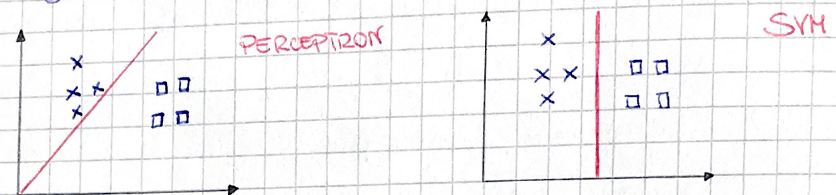
Using the KKT-hypothesis if $t_n y(x_n) \geq 1 \Rightarrow a^*_n = 0$.

In this mode we are simplifying our problem.

$w^* = \sum_{n=1}^{N} a_n^* t_n x_n$. We define the support vector as $SV = \{x_n \in D : t_n y(x_n) = 1\}$

$$w^* = \sum_{x_j \in SV} a_j^* t_j x^T x_j + w_0^* = 0$$

2).



3). We prefer maximum margin because we have no uncertainty in classification decision, because we have the maximum margin and a sort of "safety mask". For example for the solution of perceptron is very possible that one new sample will be missclassified because we have 0 margin and a very high probability of error.

Question 4

1) Given a classification problem with many classes is possible to compute, using a confusion matrix, how many times an instance of the class c_i is classified in the class c_j.

2).

	LOW	MED	HIGH
LOW	20	40	40
MED	10	80	10
HIGH	10	40	50

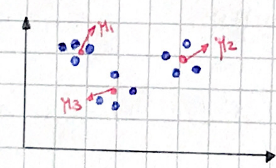
3). The accuracy is the sum of the elements in the main diagonal divided by the sum of all the elements.

$$\text{accuracy} = \frac{150}{300} = 50\%$$

Question 5

1). $P(x) = \sum_{k=1}^{N} \pi_k \mathcal{N}(x, \mu_k, \Sigma_k)$ with π_k prior probability of the class k, μ_k mean and Σ_k covariance matrix.

2).



$K=3$

3). Two unknown prior probabilities. 3 unknown mean vectors of size 2. 3 unknown covariance matrices (symmetric) with 3 unknown parameters each one.

$$\text{Size} = 2 + (3 \cdot 2) + (3 \cdot 3) = 17$$