Exercise no 8

Corentin Lacroix 1812554

MTH 6312: Méthodes Statistiques D'Apprentissage

November 4, 2015

Exercise 1

Solution 1 Here the optimization problem is to find β that solves :

$$\begin{cases} \underset{\beta,\beta_0}{min}|\beta| \\ subject\ to\ y_i(x_i\beta + \beta_0) \geq 1, i = 1, \cdots, N \end{cases}$$

But this case is very simple and : we clearly know which vectors will be the support vectors and what will be the optimal separating plane. It will be the line defined by $x = 7$. The margin will be 3 and thus, β will be $\frac{1}{3}$ and β_0, $-\frac{7}{3}$.

The optimization problem admits 1 solution, we could easily check that this one is a solution.

The image in annex sums up the case.

Exercise 2

Solution 2 These 2 cost functions don't seem equivalent. The Gaussian process cost function is the joint likelihood and the term in $\frac{1}{2}\boldsymbol{f}^\top \boldsymbol{K}^{-1}\boldsymbol{f}$ is the contribution of the prior to this joint likelihood.

Gaussian processes :

+ it is a fully statistical method, hence has an exact statistical interpretation

+ We will get exact confidence intervals for prediction.

+ we will be able to learn hyperparameters of the model (parameters of the Kernel K from the data, regularization parameter) notably by cross validation (because it is a statistical method)

SVM :

- geometrical basis/interpretation

- No possible simple exact hyperparameters optimizations because cross validation is not relevant with SVM.

- No exact statistical interpretation

Exercise 3

Solution 3 Let $\boldsymbol{K}$ be the matrix defined by $\boldsymbol{K}_{ij} = K(x_i, x_j) = \phi(x_i)^\top \phi(x_j)$.

$\forall (i,j) \in 1, \cdots, N,\ \boldsymbol{K}_{ij} = \phi(x_i)^\top \phi(x_j) = \phi(x_j)^\top \phi(x_i)$ if we transpose this scalar quantity.

Ant trivially, $\forall (i,j) \in 1, \cdots, N,\ \boldsymbol{K}_{ij} = \boldsymbol{K}_{ij}$, that is to say

$$\boxed{\boldsymbol{K} \textit{ is symetric}}$$

$\forall i \in\ 1, \cdots, N,\ \phi(x_i) = (\phi_1(x_i), \phi_2(x_i), \cdots, \phi_q(x_i))^\top$

Exercise 4

Solution 4 I had difficulties finding answers and solutions to this problem. I tried to find an optimization problem whose solution is the k-NN matrix and then tried to transform this constrained optimization problem to a simple optimization problem (without constraints).

Using the same notations of the question, we can easily remark that finding M the 2-NN neighbors matrix is equivalent to finding a solution to the following optimization problem :

$$\begin{cases} \underset{M \in \{0,1\}^{n \times n}}{min} \sum_{i=1}^{n} \sum_{j=1}^{n} \boldsymbol{M}_{ij} \boldsymbol{W}_{ij} \\ subject\ to\ \sum_{j=1}^{n} \boldsymbol{M}_{ij} = 2\ and\ \boldsymbol{M}_{ii} = 0\ \forall i \in 1, \cdots, N \end{cases}$$

In fact, here $\sum_{j=1}^{n} \boldsymbol{M}_{ij} \boldsymbol{W}_{ij}$ is the diagonal element (i, i) of $\boldsymbol{MW}$ (as $\boldsymbol{W}$ is symmetric), that is to say the total sum of the weights between the i^{th} data point to its 2-NN neighbors. This is a positive sum (as $\boldsymbol{M}$ can only take values in $\{0, 1\}$). Thus, minimizing all these quantities is equivalent to minimize the sum of all these quantitites. This is how I got to this optimization problem for finding the exact 2-NN neighbors matrix. The constraints come from the fact that :

(i) By line in $\boldsymbol{M}$, we must have exactly 2 1 and 2 0

(ii) Diagonal elements must be set to 0 as a data point can't be among his own nearest neigbhors.

We can rewrite this optimization problem with the matrix Trace function as :

$$\begin{cases} \underset{M \in \{0,1\}^{n \times n}}{min} Tr(\boldsymbol{MW}) \\ subject\ to\ \sum_{j=1}^{n} \boldsymbol{M}_{ij} = 2\ and\ \boldsymbol{M}_{ii} = 0\ \forall i \in 1, \cdots, N \end{cases}$$

We could transform this optimization problem into an optimization problem without constraints. We should add a quantity $Q(\boldsymbol{M})$ to the quantity we must optimize that satisfy the following constraints :
(ii) $Q(\boldsymbol{M})$ is zero if all constraints on $\boldsymbol{M}$ are respected (iii) $Q(\boldsymbol{M})$ is at least the maximum value taken by $Tr(\boldsymbol{MW}) = \sum_{i=1}^{n}\sum_{j=1}^{n}\boldsymbol{M}_{ij}\boldsymbol{W}_{ij}$ (which is $\sum_{i=1}^{n}\sum_{j=1}^{n}\boldsymbol{W}_{ij}$, 2 * the total sum of weights) without constraints on $\boldsymbol{M}$ to avoid tradeoff effects.

This gives us the following cost function to minimize :

$$\begin{aligned} L(\boldsymbol{M}) &= Tr(\boldsymbol{MW}) + \sum_{i=1}^{n}\sum_{j=1}^{n}\boldsymbol{W}_{ij} \times (\sum_{i=1}^{n}(\sum_{j=1}^{n}\boldsymbol{M}_{ij} - 2)^2 + \sum_{j=1}^{n}\boldsymbol{M}_{ii} \\ &= Tr(\boldsymbol{MW}) + \sum_{i=1}^{n}\sum_{j=1}^{n}\boldsymbol{W}_{ij} \times (\sum_{i=1}^{n}(\sum_{j=1}^{n}\boldsymbol{M}_{ij} - 2)^2 + Tr(\boldsymbol{M})) \end{aligned} \tag{1}$$

- The term $Tr(\boldsymbol{M})$ will force the Trace of $\hat{M}$ the solution of the problem to be 0 and M to have only zeros coefficients on its diagonal.
- the term $\sum_{i=1}^{n}(\sum_{j=1}^{n}\boldsymbol{M}_{ij} - 2)^2$ will force the lines sums of $\hat{M}$ to be exactly zero. We could eventually replace in this function $\sum_{i=1}^{n}\sum_{j=1}^{n}\boldsymbol{W}_{ij}$ by a constant since it is a problem constant and $\sum_{i=1}^{n}(\sum_{j=1}^{n}\boldsymbol{M}_{ij} - 2)^2$ by a trace af a more complex matrix.

The optimization problem for getting a symmetric 2-NN is the same than previously but with one more constraints on the $\hat{M}$ coefficients :

$$\begin{cases} \underset{M \in \{0,1\}^{n \times n}}{min} Tr(\boldsymbol{MW}) \\ subject\ to\ \sum_{j=1}^{n} \boldsymbol{M}_{ij} = 2,\ \boldsymbol{M}_{ii} = 0\ \forall i \in 1, \cdots, N \\ and\ \boldsymbol{M}_{ij} = \boldsymbol{M}_{ji}\ \forall (i,j) \in 1, \cdots, N \end{cases}$$

Then, we get a modified cost function including that forces M to be symmetric :

$$L(\boldsymbol{M}) = Tr(\boldsymbol{MW}) + \sum_{i=1}^{n} \sum_{j=1}^{n} \boldsymbol{W}_{ij} \times (\sum_{i=1}^{n} (\sum_{j=1}^{n} \boldsymbol{M}_{ij} - 2)^2 + Tr(\boldsymbol{M}) + \sum_{i=1}^{n} \sum_{j=1}^{n} (M_{ij} - M_{ji})^2) \tag{2}$$

These cost functions can lead to our desired solutions for any n and k (if $k \leq n$)...