Sharif University of Technology
Computer Engineering Department
Ali Sharifi-Zarchi

Homework Assignment 2
Machine Learning for Bioinformatics,
Spring 2023

1402-01-19

Hossein Kargar, Ali Salmani, Nima Roshanzadeh

1 SVM for Classification

1.1

In the linearly separable case if one of the training samples is removed, will the decision boundary shift toward the point removed or shift away from the point removed, or remain the same? Justify your answer. Now if we consider that the decision boundary is of Logistic Regression, will the decision boundary change or remain the same? Explain your answer. (No need to mention the direction of change)

1.2

Recall from the lecture notes that if we allow some misclassification in the training data, the primal optimization of SVM(soft margin) is given by

$$\begin{aligned} &\min_{\omega,\xi_i} 1/2\|\mathbf{w}\|_2^2 + C\Sigma_{i=1}^{n}\xi_i \qquad (1)\\ \text{s.t.}\quad & y_i(\mathbf{w}^\top(x_i)) \geq 1-\xi_i,\ \forall i \in \{1,\ldots,n\}\\ & \xi_i \geq 0,\ \forall i \in \{1,\ldots,n\} \end{aligned}$$

where $\xi_1, ..., \xi_n$ are called slack variables.
Suppose the optimal $\xi_1, ..., \xi_n$ have been computed. Use the ξ_i to obtain an upper bound on the number of misclassified instances.

1.3

In the primal optimization of SVM, what's the role of the coefficient C? Briefly explain your answer by considering two extreme cases, i.e., C$\rightarrow$0 and C$\rightarrow\infty$

1.4

Compare Hard SVM and Logistic Regression when the two classes are linearly separable. Give any significant differences. (*Hint* - think in terms of decision boundary)

1.5

Compare Soft SVM and Logistic Regression when the two classes are not linearly separable. Give any significant differences.

2 Composing Kernel Functions

A key benefit of SVM training is the ability to use kernel functions $K(x, x')$ as opposed to explicit basis functions $\phi(x)$. Kernels make it possible to implicitly express large or even infinite dimensional basis features. We do this by computing $\phi(x)^\top\phi(x')$ directly, without ever computing $\phi(x)$.
When training SVMs, we begin by computing the kernel matrix K, over our training data $\{x_1, \ldots, x_n\}$. The kernel matrix, defined as $K_{i,i'} = K(x_i, x_{i'})$, expresses the kernel function applied between all pairs of training points.
In class, we saw Mercer's theorem, which tells us that any function K that yields a positive semi-definite kernel matrix forms a valid kernel, i.e. corresponds to a matrix of dot-products under *some* basis ϕ. Therefore instead of using an explicit basis, we can build kernel functions directly that fulfill this property.
A particularly nice benefit of this theorem is that it allows us to build more expressive kernels by composition. In this problem, you are tasked with using Mercer's theorem and the definition of a kernel matrix to prove that the following compositions are valid kernels, assuming $K^{(1)}$ and $K^{(2)}$ are valid kernels. Recall that a positive semi-definite matrix K requires $\mathbf{z}^\top\mathbf{K}\mathbf{z} \geq 0,\ \forall\ \mathbf{z} \in \mathbb{R}^n$.

1. $K(x, x') = c\, K^{(1)}(x, x') \quad \text{for } c > 0$
2. $K(x, x') = K^{(1)}(x, x') + K^{(2)}(x, x')$
3. $K(x, x') = K^{(1)}(x, x')\, K^{(2)}(x, x')$
   [Hint: Use the property that for any $\phi(x)$, $K(x, x') = \phi(x)^\top \phi(x')$ forms a positive semi-definite kernel matrix.]
4. (Bonus)
   (a) The $\exp$ function can be written as,
   $$\exp(x) = \lim_{i \to \infty} \left(1 + x + \cdots + \frac{x^i}{i!}\right).$$
   Use this to show that $\exp(xx')$ (here $x, x' \in \mathbb{R}$)) can be written as $\phi(x)^\top \phi(x')$ for some basis function $\phi(x)$. Derive this basis function, and explain why this would be hard to use as a basis in standard logistic regression.
   (b) Using the previous identities, show that $K(x, x') = \exp(K^{(1)}(x, x'))$ is a valid kernel.
5. (Bonus) Finally use this analysis and previous identities to prove the validity of the Gaussian kernel:
   $$K(x, x') = \exp\left(\frac{-||x - x'||_2^2}{2\sigma^2}\right)$$

3 K-fold Cross-Validation

3.1

Explain how k-fold cross-validation is implemented.

3.2

What are the advantages and disadvantages of k-fold cross-validation relative to:
a) The validation set approach?
b) LOOCV?

3.3

Read about Monte Carlo Cross Validation (MCCV) and explain this method briefly.

3.4

Compare MCVV with k-Fold Cross Validation and state its pros and cons.

4 Hyperparameter Optimization

4.1

What hyperparameters have we seen so far in this course? What bad things can happen if we set them inefficiently?

4.2

What's the difference between hyperparameter optimization and regular training? To put it another way, what's the difference between the model parameters and the hyperparameters?

4.3

How can we incorporate our own knowledge and insights about the problem into an optimization method?

5 Decision Tree

The following table contains training data that help predict whether a patient is likely to have a heart attack.

Patient ID	Chest Pain	Male	Smokes	Exercises	Heart Attack
1	Yes	Yes	No	Yes	Yes
2	Yes	Yes	Yes	No	Yes
3	No	No	Yes	No	Yes
4	No	Yes	No	Yes	No
5	Yes	No	Yes	Yes	Yes
6	No	Yes	Yes	Yes	No

5.1

Use entropy to construct a minimal decision tree that predicts whether or not a patient is likely to have a heart attack. Show each step of the computation.

5.2

Based on the tree that you have built in the previous section, now classify someone who has chest pain.

6 AdaBoost Algorithm

6.1

As we know in every step of AdaBoost algorithm, a classifier with the least error (considering the distribution of that step) would be chosen. Prove that this algorithm never chooses two identical functions in two successive steps.($h_t \neq h_{t+1}$)

6.2

Assume the distribution vector $(D_{t+1}(1), D_{t+1}(2), ..., D_{t+1}(m))$ in AdaBoost algorithm in which m represents the number of samples. Prove that this vector and the vector with components ($y_i h_t(x_i)$) are uncorrelated. (meaning that their dot product is 0).