Machine Learning Worksheet 07

Linear Classification

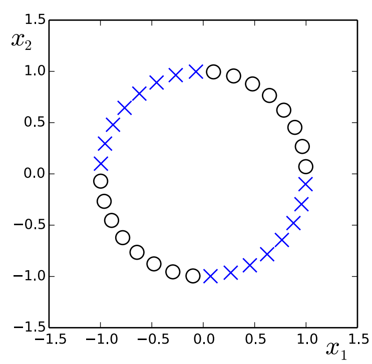
This is a humongous homework sheet. For this one, we waive the 66% rule. This time, it suffices if you do only half of the exercises.

We suggest that you do the last three exercises, and complete that with a random four of the computational ones—that would be your easiest way out.

1 Linear separability

Problem 1: Show that for a linearly separable data set, the maximum likelihood solution for the logistic regression model is obtained by finding a vector $\boldsymbol{w}$ whose decision boundary $\boldsymbol{w}^T\phi(\boldsymbol{x}) = 0$ separates the classes and then taking the magnitude of $\boldsymbol{w}$ to infinity.

Problem 2: Which basis function $\phi(x_1, x_2)$ makes the data in the example below linearly separable (crosses in one class, circles in the other)?



2 Basis functions

Problem 3: The decision boundary for a linear classifier on two-dimensional data crosses axis x_1 at 2 and x_2 at 5. Write down the general form of this linear classifier model with a bias term (how many parameters do you need, given the dimensions?) and calculate possible coefficients (parameters).

3 Bounds

Problem 4: Suppose we test a classification method on a set of n new test cases. Let $X_i = 1$ if the classification is wrong and $X_i = 0$ if it is correct. Then $\hat{X} = n^{-1} \sum X_i$ is the observed error rate. If we regard each X_i as a Bernoulli with unknown mean p, then p should be the true, but unknown, error rate of our method. How likely is $\hat{X}$ to not be within ε of p. How many test cases are necessary to ensure that the observed error rate is with probability at most 5% further than 0.01 away from the true one?

4 The perceptron

An important example of a so called *linear discriminant* model is the *perceptron* of Rosenblatt. The following questions will look more closely at this algorithm. We will assume the following:

- The parameters of the perceptron learning algorithm are called *weights* and are denoted by $\boldsymbol{w}$.
- The *training set* consists of training inputs $\boldsymbol{x_i}$ with labels $t_i \in \{+1, -1\}$.
- The *learning rate* is 1.
- Let k denote the number of weight updates the algorithm has performed at some point in time and $\boldsymbol{w^k}$ the weight vector after k updates (initially, $k = 0$ and $\boldsymbol{w^0} = \boldsymbol{0}$).
- All training inputs have bounded euclidean norms, i.e. $||\boldsymbol{x_i}|| < R$, for all i and some $R \in \mathbb{R}^+$.
- There is some $\gamma > 0$ such that $t_i \boldsymbol{\tilde{w}^T x_i} > \gamma$ for all i and some suitable $\boldsymbol{\tilde{w}}$ (γ is called a *finite margin*).

Problem 5: Write down the perceptron learning algorithm.

Problem 6: Given the following training set $\mathcal{D}$ of labeled 2D training inputs, find a *separating hyperplane* using the perceptron learning rule. Illustrate the consecutive updates of the weight $\boldsymbol{w}$ with a series of plots (do not plot the bias weight)!

$$\begin{aligned}\mathcal{D} = &\{((-0.7, 0.8), +1), ((-0.9, 0.6), +1), ((-0.3, -0.2), +1), ((-0.6, 0.7), +1)\} \\ &\cup \{((0.6, -0.8), -1), ((0.2, -0.5), -1), ((0.3, 0.2), -1)\}\end{aligned}$$

You will now show that the perceptron algorithm converges in a finite number of updates (if the training data is linearly separable).

Problem 7: Let $\boldsymbol{w^k}$ be the k^{th} update of the weight during the perceptron algorithm. Show that $(\boldsymbol{\tilde{w}^T w^k}) \geq k\gamma$. (Hint: How are $(\boldsymbol{\tilde{w}^T w^k})$ and $(\boldsymbol{\tilde{w}^T w^{k-1}})$ related?)

Problem 8: Show that $||\boldsymbol{w^k}||^2 < kR^2$. Note that the algorithm updates the weights only in response to a mistake (i.e., $t_i \boldsymbol{x_i^T w^{k-1}} \leq 0$ for some i). (Hint: Triangle inequality for the Euclidean norm.)

Problem 9: Consider the cosine of the angle between $\boldsymbol{\tilde{w}}$ and $\boldsymbol{w^k}$ and derive

$$k \leq \frac{R^2 ||\boldsymbol{\tilde{w}}||^2}{\gamma^2}.$$

Now consider a new data set, $\mathcal{D}'$ (again 2D inputs and two different classes):

$$\begin{aligned}\mathcal{D}' = &\{((0,0),+1),((-0.1,0.1),+1),((-0.3,-0.2),+1),((0.2,0.1),+1)\}\\ &\cup\{((0.2,-0.1),+1),((-1.1,-1.0),-1),((-1.3,-1.2),-1),((-1,-1),-1)\}\\ &\cup\{((1,1),-1),((0.9,1.2),-1),((1.1,1.0),-1)\}\end{aligned}$$

Problem 10: Can you separate this data with the perceptron algorithm? Why/why not?

Problem 11: Transform every input $\boldsymbol{x_i} \in \mathcal{D}'$ to $\boldsymbol{x'_i}$ with $\boldsymbol{x'_{i1}} = \exp(\frac{-||\boldsymbol{x_i}||^2}{2})$ and $\boldsymbol{x'_{i2}} = \exp(\frac{-||\boldsymbol{x_i}-(1,1)||^2}{2})$. If the labels stay the same, are the $\boldsymbol{x'_i}$s now linearly separable? Why/ why not?

Problem 12: Solve the first of the four programming exercises (i.e., fill in the blanks) in the file `LinearClassification.ipynb` in piazza's "General Resources". Put the code here.

Problem 13: Solve the second of the four programming exercises (i.e., fill in the blanks) in the file `LinearClassification.ipynb` in piazza's "General Resources". Put the code here.

Problem 14: Solve the third of the four programming exercises (i.e., fill in the blanks) in the file `LinearClassification.ipynb` in piazza's "General Resources". Put the code here.