IN4320 : Machine Learning Assignment 1

February 17, 2019

We consider a novel classifier that works in a similar vain as the nearest mean classifier (NMC). It uses trained medians and the 1-norm, rather than the 2-norm, to decide which median is closest to a new sample x and so to decide to which class this sample x is assigned. We will refer to this classifier as the nearest median classifier or NmC for short.

Training the classifier comes down to minimizing the objective function

$$L(m_-, m_+, a) := \left(\sum_{c\in\{-,+\}} \sum_{i=1}^{N_c} \|x_i^c - m_c\|_1 \right) + \lambda \|m_+ - m_- + a\|_1 \,. \tag{1}$$

Its variables are $m_+ \in \mathbb{R}^d$, $m_- \in \mathbb{R}^d$, and scalar $a \in \mathbb{R}$. The scalar λ is nonnegative. The vectors m_+ and m_- are referred to as the median for the positive and negative class, respectively. The function $||\cdot||_1$ denotes the 1-norm (also referred to as the ℓ_1-norm). The ith data point in $\mathbb{R}^d$ from the positive class is denoted by x_i^+, while $x_i^- \in \mathbb{R}^d$ are the data from the negative class. The positive class has N_+ observations and the negative has N_-. The total number is $N = N_+ + N_-$.

Note : IN4320 assignments should be made individually. You must provide your own answers.

Convexity!

1 A true machine learner isn't afraid of a bit of mathematics of course. A known challenge is to proof an objective function to be convex in its variables. One way to do this for a general function $f : \mathbb{R}^d \to \mathbb{R}$ is by showing that for any pair of inputs a and b and any $c \in [0, 1]$, it hold that $f(ca + (1 - c)b) \leq cf(a) + (1 - c)f(b)$.

Proof that L is convex on its domain $\mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}$. You may start by showing that the norm $||\cdot||$ is convex. (Hint : you could use the properties of a norm.) You can then build it up from there using the *Operations that preserve convexity* on the *Convex function* Wikipedia. You should give your proof using these "basics" only and *cannot* use more advanced prior results. **/ 2 points**

Optimality & Geometry?

2 Consider the 1-dimensional case (so $d = 1$) and say we have one observation for the positive class, which is 0 and two for the negative, which are 1 and 3. The regularization

parameter, λ, is fixed to 1. Also, m_+ is already fixed to 0 and the only free variables left are m_- and a. Determine all the corner points of the contour $\{(m_-, a)|L(0, m_-, a) = \pi\}$. (It should suffice that you provide the detailed derivation for one of the corner points, while you merely mention the solution coordinates for the remaining ones). **/ 1 point**

3 For the general setting, explain what the term $\|m_+ - m_- + a\|_1$ tries to enforce. Does it lead to sparse solutions? Assuming that m_+ and m_- are fixed, what optimal value does a take on? (Hint : plotting $\|m_+ - m_- + a\|_1$ as a function of a may get help in understanding especially the last question.) **/ 2 points**

Try, Try, Try...

Through the course page you can find the data for a two-class classification task in $d = 21$ dimensions. It is named `digits.txt`, but has nothing to do with digits whatsoever. The feature vectors are stored in rows and the first 10000 belong to the positive class, while the remaining 10000 rows belong to the negative class.

4 Implement an optimizer for the NmC from Equation (1) and convince yourself that it indeed optimizes what it should optimize. You can use gradient descent or any other approach of your liking. You are even allowed to use standard optimization toolboxes and the like. You can either implement a general version of the NmC or one that is completely dedicated to the data set that is given. (Note, however, that the latter is not necessarily easier to implement and it is probably harder to debug.)

Describe *in no more than 200 words* the main ingredients of and/or considerations behind your optimization routine. In particular, sketch the search strategy that you employ and explain how you decide when you have reached the sought-after minimum. **/ 1 point**

5 Make so-called regularization curves in which you vary the amount of regularization. You can compare this to plotting learning or feature curves, only now we have λ on the horizontal axis. Report both the apparent error and the estimated true error. Show classification results for training set sizes of 10 and 1000 samples *per class* and, at least, estimate the necessary error rates for $\lambda \in \{0, 1, 10, 100, 1000, 10000\}$. Make separate plots for the 10 and 100 sample cases. Make sure you repeat the experiment often enough so to get somewhat stable curves. In addition, make the same plots (again in separate figures) for the optimal value of the objective that you find at training time. **/ 2 points**

6 Based on the foregoing experiment, make four plots : for the two settings of 10 and 1000 training samples per class and for $\lambda \in \{0, 10000\}$. Plot the average medians (against the 21 feature dimensions) and indicate the standard deviations. **/ 2 points**

My assessment : you should be able to keep your report within three pages or 1000 words...