COMS 4721/4771 HW3 (Spring 2024)

Due: Thu April 04, 2024 at 11:59pm

This homework is to be done **alone**. No late homeworks are allowed. To receive credit, a typesetted copy of the homework pdf must be uploaded to Gradescope by the due date. You must show your work to receive full credit. Discussing possible approaches for solutions for homework questions is encouraged on the course discussion board and with your peers, but you must write your own individual solutions and **not** share your written work/code. You **must cite** all resources (including online material, books, articles, ai generative bots, help taken from/given to specific individuals, etc.) you used to complete your work.

It is your responsibility to protect your work, and it is a violation of academic integrity policy to post any part of these questions or your answers to public websites such as: github, bitbucket, chegg, coursehero, etc. **Violators will be reported to the dean for disciplinary action**.

1 Strange consequences of high dimensionality

As discussed in class, we often represent our data in high dimensions. Thus to understand our data better and design effective prediction algorithms, it is good to understand how things behave in high dimensions. Obviously, since we cannot visualize or imagine high dimensional spaces, we often tend to rely on how data behave in one-, two- or three-dimensions and extrapolate how they may behave in hundreds of dimensions. It turns out that our low dimensional intuition can be very misleading about data and distributions in high dimensional spaces. In this problem we will explore this in more detail.

Consider the Gaussian distribution with mean μ and identity covariance I_d in $\mathbb{R}^d$. Recall that the density assigned to any point $x \in R^d$, then becomes

$$p(x) = (2\pi)^{-d/2} \exp\big\{ - \|x - \mu\|^2/2 \big\}.$$

(i) Show that when $x = \mu$, x gets assigned the highest density.

(This, of course, makes sense: the Gaussian density peaks at its mean and thus $x = \mu$ has the highest density.)

(ii) If mean has the highest density, it stands to reason that if we draw a large i.i.d. sample from the distribution, then a large fraction of the points should lie close to the mean. Let's try to verify this experimentally. For simplicity, let mean $\mu = 0$ (covariance is still I_d). Draw 10,000 points i.i.d. from a Gaussian $N(0, I_d)$.

To see how far away a sampled datapoint is from the mean, we can look at the distance $\|x - \mu\|^2 = \|x\|^2$ (that is, the squared length of the sampled datapoint, when mean is zero). Plot the histogram of squared length of the samples, for dimensions $d = 1, 2, 3, 5, 10, 50$ and 100. You should plot the all these histograms on the same figure for a better comparison.

What interesting observations do you see from this plot? Do you notice anything strange when the samples that were drawn from the high dimensional Gaussian distribution? Do most of the samples lie close to the mean?

(iii) Let's mathematically derive where we *expect* these samples to lie. That is, calculate

$$\mathbb{E}_{x \sim N(0, I_d)} \Big[\|x\|^2 \Big].$$

Is the empirical plot in part (ii) in agreement with the mathematical expression you derived here?

(iv) This "strangeness" is not specific to Gaussian distribution, you can observe something similar even for the simplest of distributions in high dimensions. Consider the uniform distribution over the cube $[-1, 1]^d$. Just like in part (ii), draw 10,000 i.i.d. samples from this d-dimensional cube with uniform density, and plot the histogram of how far away from the origin the sample points lie. (do this for $d = 1, 2, 3, 5, 10, 50$ and 100, again on the same plot).

Recall that the cube has side length of 2, while most of the high-dimensional samples have length of far more than 2! This means even though you are drawing uniformly from the cube, most of your samples lie in the corners (and not the interior) of the cube!

(v) Again, calculate the expected (squared) length of the samples. That is, calculate

$$\mathbb{E}_{x \sim \text{unif}([-1,1]^d)} \Big[\|x\|^2 \Big].$$

Does the plot in part (iv) in agreement with the expression you derive here?

2 An alternate learning paradigm

In class you have seen that when building classifiers, one wants to minimize the expected classification error over a distribution $\mathcal{D}$. That is, we want to find the classifier f that minimizes:

$$\mathbb{E}_{(x,y)\sim\mathcal{D}} \Big[\mathbf{1}[f(x) \neq y] \Big]. \tag{1}$$

Since this quantity is not estimable in practice (since we don't know $\mathcal{D}$ and only have access to finite samples drawn from it), it is usually approximated via its empirical equivalent:

$$\frac{1}{|S|} \sum_{(x,y)\in S} \mathbf{1}[f(x) \neq y]. \tag{2}$$

This latter quantity is the training error if S is the training set, and is the testing error if S is the testing set.

However for certain applications, obtaining a positively and negatively labelled samples S is not possible. Consider, for example, the problem of modelling user preferences based on news-feed that gets shown. Very simply, if a user interacts with a particular news item (such as they clicked and read it) shows that they are interested in the contents of the article, thus providing a positive label. But if a user does not interact with a particular news item, it is not clear whether the user dislikes the contents of the article, or simply didn't get around to viewing it. In such a scenario obtaining a good quality negatively labelled data sample is not possible. We thus need a slightly different learning paradigm where the training samples obtained are only either labelled as positive examples, or they are simply unlabeled examples. We can model this as follows:

- $\mathcal{D}$ is an unknown distribution over $\mathbb{R}^D \times \{0,1\} \times \{0,1\}$. $(x, y, s) \sim \mathcal{D}$ is a sample, where x is the input feature vector, y is the true label, and s (the "selection" variable) is whether x was interacted with (ie, selected) or not. Note that only x and s are observed.
- $\Pr[s = 1 \mid x, y = 0] = 0$, that is, a negatively labelled x is never selected.
- Given y, s and x are conditionally independent. That is, which x gets selected (given that, say, x positively labelled) is chosen independently.

The goal of this problem is to find an empirical estimator of (1) similar to (2) but using the unlabeled and positive data only.

(i) Prove that $\Pr[y = 1 \mid x] = \frac{\Pr[s=1|x]}{\Pr[s=1|y=1]}$.

(ii) Using (i) prove that $\Pr[y = 1 \mid x, s = 0] = \frac{1-\Pr[s=1|y=1]}{\Pr[s=1|y=1]} \frac{\Pr[s=1|x]}{1-\Pr[s=1|x]}$.

For the rest of the problem, assume that both quantities on the RHS can be estimated from (x, s) data only. This is trivially true for $\Pr[s = 1 \mid x]$ (since it does not depend on y). And while estimating $\Pr[s = 1 \mid y = 1]$ with only(x, s) data is nontrivial, it can be done under suitable conditions.

(iii) Letting p denote the PDF of $\mathcal{D}$ show that:

$$\begin{aligned}\mathbb{E}_{(x,y)\sim\mathcal{D}}\big[\mathbf{1}[f(x)\neq y]\big] = \int_x & p(x,s=1)\mathbf{1}[f(x)\neq 1] \\ & + p(x,s=0)(\Pr[y=1\mid s=0,x]\mathbf{1}[f(x)\neq 1] \\ & \qquad + \Pr[y=0\mid s=0,x]\mathbf{1}[f(x)\neq 0])dx\end{aligned}$$

(iv) Using parts (ii) and (iii) suggest an empirical estimator of (1) similar to (2) but that uses only (x, s) data.

Hint: Try viewing unlabeled points as part positive and part negative. That is, replace unlabeled points by two "partial" points. One that is positive with weight $w(x)$ and one negative with weight $1 - w(x)$.

3 Bayesian interpretation of ridge regression

Consider the following data generating process for linear regression problem in $\mathbb{R}^d$. Nature first selects d weight coefficients $w_1, \ldots, w_d$ as $w_i \sim N(0, \tau^2)$ i.i.d. Given n examples $x_1, \ldots, x_n \in \mathbb{R}^d$, nature generates the output variable y_i as

$$y_i = \sum_{j=1}^{d} w_j x_{i,j} + \epsilon_i,$$

where $\epsilon_i \sim N(0, \sigma^2)$ i.i.d.

Show that finding the coefficients $w_1, \ldots, w_d$ that maximizes $P[w_1, \ldots, w_d | (x_1, y_1) \ldots, (x_n, y_n)]$ is equivalent to minimizing the ridge optimization criterion.

4 1-Norm Support Vector Machine

Recall the standard support vector machine formulation:

$$\begin{aligned}&\text{minimize} && \|\mathbf{w}\|_2^2 \\ &\text{subject to} && y_i(\mathbf{w}\cdot\mathbf{x}_i + w_0) \geq 1, \qquad i = 1, \ldots, m,\end{aligned}$$

where m is the number of points and n is the number of dimensions, is a *quadratic program* because the objective function is quadratic and the constraints are affine. A *linear program* on the other hand uses only affine objective function and constraints, and there are efficient polynomial-time algorithms for solving them. By replacing the 2-norm in the objective function with the 1-norm ($\|\mathbf{x}\|_1 = \sum_{i=1}^{n} |x_i|$), we get

$$\begin{aligned}&\text{minimize} && \|\mathbf{w}\|_1 \\ &\text{subject to} && y_i(\mathbf{w}\cdot\mathbf{x}_i + w_0) \geq 1, \qquad i = 1, \ldots, m.\end{aligned} \tag{3}$$

Note that this is still not a linear program because the objective value contains absolute values, but we will convert this problem into an equivalent linear problem.

(i) Suppose we know that the output y depends only on a few input variables (i.e. the optimal $\mathbf{w}$ is sparse). Would the 1-norm or 2-norm SVM make more sense? Justify your answer.

(ii) The Chebyshev (ℓ_∞) distance between two points $\mathbf{x}$ and $\mathbf{y}$ is defined as $\max_i |x_i - y_i|$. Show that the 1-norm SVM maximizes the Chebyshev distance between the two separating hyperplanes $\mathbf{w}\cdot\mathbf{x} + w_0 = \pm 1$, which is the smallest value of $||\mathbf{x} - \mathbf{y}||_\infty$ such that $\mathbf{x}$ is a point on the first hyperplane and $\mathbf{y}$ is a point on the second hyperplane.

Hint: Show that the l_∞ distance from the origin to the plane $\mathbf{w}\cdot\mathbf{x} = 2$ is minimized along the vector $(\text{sign}(w_1), \ldots \text{sign}(w_n))$. Then, the distance from the origin to $\mathbf{w}\cdot\mathbf{x} = 2$ is the value of λ satisfying $\mathbf{w}\cdot\lambda(\text{sign}(w_1), \ldots, \text{sign}(w_n)) = 2$.

(iii) Consider the unconstrained minimization problem

$$\text{minimize} \quad \max(ax + b, cx + d), \tag{4}$$

over $x \in \mathbb{R}$. Argue that (2) is equivalent to the following linear program with an auxiliary variable:

$$\begin{aligned} \text{minimize} \quad & t \\ \text{subject to} \quad & t \geq ax + b, \\ & t \geq cx + d. \end{aligned}$$

In other words, show that both problems will always obtain the same optimal objective value.

Note that by increasing the dimension of the problem's *decision space*, we have transformed the original problem into a linear program.

(iv) Observe that the 1-norm function $||\mathbf{w}||_1$ in (1) can be written as $\sum_{i=1}^{n} \max(w_i, -w_i)$. Using this fact and the equivalency from part (i), rewrite (1) as a linear program with $2n + 1$ variables and $m + 2n$ constraints.

(v) When the input data are not perfectly separable, we can apply a *soft-margin* approach (this is an alternative to the usual *slack-variables* approach discussed in class):

$$\text{minimize} \quad \|\mathbf{w}\|_1 + \sum_{i=1}^{m} [1 - y_i(\mathbf{w} \cdot \mathbf{x}_i + w_0)]_+ \,, \tag{5}$$

where $[\cdot]_+$ is the hinge loss function given by $\max(0, \cdot)$. Note that we've replaced the constraints with a penalty in the objective function. Rewrite the soft-margin 1-norm SVM problem as a linear program. How many variables and constraints are present in this formulation?

(vi) *Extra credit:* Duality is a rich topic in optimization theory because it provides an alternate lens for viewing optimization problems. It is typically not possible to use duality to find closed-form solutions to optimization problems as we did in class. Nevertheless, duality gives rise to other ways of solving and analyzing problems. For example, weak daulity says that any feasible dual solution gives a lower bound on the p^*, the optimal value of the primal. Also, adding a constraint to the primal is equivalent to adding a variable to the dual, which is really useful in delayed constraint generation in which complex linear programs are solved by iteratively adding constraints to a smaller problem.

Show that the dual of the linear program constructed in part (v) can be expressed as

$$\begin{aligned} \text{maximize} \quad & \|\boldsymbol{\pi}\|_1 \\ \text{subject to} \quad & \left| \sum_{i=1}^{m} y_i x_{ij} \pi_i \right| \leq 1 \qquad j = 1, \ldots, n, \\ & \sum_{i=1}^{m} y_i \pi_i = 0, \\ & 0 \leq \pi_i \leq 1 \qquad\qquad i = 1, \ldots, m, \end{aligned}$$

where $\boldsymbol{\pi} \in \mathbb{R}^m$ [1]. Since strong duality always holds for linear programs, this problem is equivalent to the soft-margin formulation in (3).

Hint: Think about what constraints need to be satisfied for $\min_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda})$ to be bounded. You do not have to take any partial derivatives.

[1] This formulation is taken from `https://arxiv.org/pdf/1901.01585.pdf`, which describes ways of combining the primal and dual formulations to efficiently solve the 1-norm SVM. It is a good exploration of many important optimization topics beyond the scope of this class.