MATH 390.4 / 650.2 Spring 2018 Homework #2t

Adina Bechhofer

Wednesday 7th March, 2018

Problem 1

These are questions about the SVM.

(a) [easy] State the hypothesis set $\mathcal{H}$ inputted into the support vector machine algorithm. Is it different than the $\mathcal{H}$ used for $\mathcal{A}$ = perceptron learning algorithm?

$\mathcal{H} = \{\mathbb{1}_{w \bullet x + b > 0} | x \in \mathcal{X}\}$ No, this is the same hypothesis space as for the perceptron. However, here b is written explicitly, while in the perceptron it was embedded in the w vector as w_0.

(b) [E.C.] Why is the SVM better than the perceptron? A non-technical discussion that makes sense is fine. Write it on a separate page

The perceptron algorithm crashes when the data isn't linearly separable. The SVM, on the other hand, can handle non linear separable cases. Even in the linearly separable case, the SVM gives the separation that maximizes the margin. The perceptron will give the first line it finds.

(c) [difficult] Let $\mathcal{Y} = \{-1, 1\}$. Rederive the cost function whose minimization yields the SVM line in the linearly separable case.

Lower line: $w \cdot X - b = -1$
Upper line: $w \cdot X - b = 1$
We want all 1's to be above the upper line, and all -1's to be below the bottom line. Thus we have:
if $y_i = 1$, $w \cdot X - b \geq 1$
if $y_i = -1$, $w \cdot X - b \leq -1$
Minimize $||w||$ with the constraint $\forall i$, $y_i(w \cdot X - b) \geq 1$

(d) [easy] Given your answer to (c) rederive the cost function using the "soft margin" i.e. the hinge loss plus the term with the hyperparameter λ. This is marked easy since there is just one change from the expression given in class.

Cost $= max\{0, 1 - y_i(w \cdot X - b)\} + \lambda ||w||^2$

Problem 2

These are questions are about the k nearest neighbors (KNN) algorithm.

(a) [easy] Describe how the algorithm works. Is k a "hyperparameter"?

For each new value x^*, he algorithm looks at the k data points in the training set, $\mathbb{D}$, that are closest to x^*, (different distance metrics can be used). Then, the algorithm returns the mode of the y's of the neighbors. K is a hyperparameter. It is not determined by the algorithm or the data. It is a choice that the model builder must make. For each k, the knn algorithm will generate a different prediction.

(b) [difficult] Assuming $\mathcal{A}$ = KNN, describe the input $\mathcal{H}$ as best as you can.

$\mathcal{H}$ is a combination of an indicator function, an argmin function, and a mode. The output $\hat{\boldsymbol{y}} \in \{0, 1\}$ will be 1 if the mode of the responses to the k x's with minimum distance from it is 1, and 0 otherwise.

(c) [difficult] When predicting on $\mathbb{D}$ with $k = 1$, why should there be zero error? Is this a good estimate of future error when new data comes in? (Error in the future is called *generalization error* and we will be discussing this later in the semester).

When testing nearest neighbor on $\mathcal{D}$, the error will be zero because every point is mapped to its own response. This is because minimum distance is zero distance. However, this is not the expected error for never before seen x's, since they don't appear in $\mathcal{D}$ and cannot be mapped to their own response.

Problem 3

These are questions about the linear model with $p = 1$.

(a) [easy] What does $\mathbb{D}$ look like in the linear model with $p = 1$? What is $\mathcal{X}$? What is $\mathcal{Y}$?

In a linear model with $p = 1$, $\mathbb{D}$ is a 1×n vector of one feature.
$\mathcal{X} = \mathbb{R}$
$\mathcal{Y} = \mathbb{R}$

(b) [easy] Consider the line fit using the ordinary least squares (OLS) algorithm. Prove that the point $< \bar{x}, \bar{y} >$ is on this line. Use the formulas we derived in class.

$y = b_0 + b_1 x$

Where:

$$b_1 = \frac{\Sigma x_i y_i - n\bar{y}\bar{x}}{\Sigma x_i^2 - n\bar{x}^2} \qquad b_0 = \bar{y} - \frac{\Sigma x_i y_i - n\bar{y}\bar{x}}{\Sigma x_i^2 - n\bar{x}^2}\bar{x}$$

Show that $\bar{y} = b_0 + b_1\bar{x}$

$$y = \frac{\Sigma x_i y_i - n\bar{y}\bar{x}}{\Sigma x_i^2 - n\bar{x}^2}\bar{x} + \bar{y} - \frac{\Sigma x_i y_i - n\bar{y}\bar{x}}{\Sigma x_i^2 - n\bar{x}^2}\bar{x}$$

$$y = \bar{y}$$

(c) [harder] Consider the line fit using OLS. Prove that the average prediction $\hat{y}_i := g(x_i)$ for $x_i \in \mathbb{D}$ is $\bar{y}$.

Show that $\frac{1}{n}\sum_{j=1}^{n} \hat{y}_j = \bar{y}$

$$\begin{aligned}\frac{1}{n}\sum_{i=1}^{n} \hat{\boldsymbol{y}}_i &= \frac{1}{n}\sum_{i=1}^{n} b_0 + \frac{1}{n}\sum_{i=1}^{n} b_1 x_i \\ &= \frac{1}{n} n b_0 + b_1 \frac{1}{n}\sum_{i=1}^{n} x_i \\ &= b_0 + b_1 \bar{x} \\ &= \bar{y}\end{aligned}$$

By the proof for question b.

(d) [harder] Consider the line fit using OLS. Prove that the average residual e_i computed from all predictions for $x_i \in \mathbb{D}$ and its true response value y_i is 0.

Show that $\frac{1}{n}\sum_{i=1}^{n} e_i = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i) = 0$

$$\frac{1}{n}\sum_{i=1}^{n} e_i = \frac{1}{n}\sum_{i=1}^{n} \hat{y}_i - \frac{1}{n}\sum_{i=1}^{n} y_i$$

By problem c, $\frac{1}{n}\sum_{i=1}^{n} \hat{y}_i = \bar{y}$. By definition, $\frac{1}{n}\sum_{i=1}^{n} y_i = \bar{y}$ Thus,

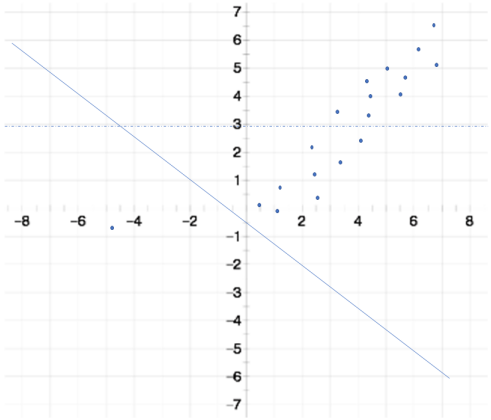
$$\frac{1}{n}\sum_{i=1}^{n} e_i = \bar{y} - \bar{y} = 0$$

(e) [harder] Why is the RMSE usually a better indicator of predictive performance than R^2? Discuss in English.

The "fake" Central Limit Theorem states that 95% of predictions will fall within 2 RMSE above or below the true value. This gives us a concrete sense of how far off are the predictions made by the model.

(f) [harder] R^2 is commonly interpreted as "proportion of the variance explained by the model" and proportions are constrained to the interval $[0, 1]$. While it is true that $R^2 \leq 1$ for all models, it is not true that $R^2 \geq 0$ for all models. Construct an explicit example $\mathbb{D}$ and create a linear model $g(x) = w_0 + w_1 x$ whose $R^2 < 0$. Hint: do not use the OLS line. Hint: draw a picture!

This happens when $g(x)$ doesn't fit the data at all, and does a worse job than the null model ($g_0 = \bar{y}$).



(g) [E.C.] Prove that the OLS line always has $R^2 \in [0, 1]$ on a separate page.

(h) [difficult] You are given $\mathbb{D}$ with n training points $< x_i, y_i >$ but now you are also given a set of weights $[w_1 \ w_2 \ \ldots \ w_n]$ which indicate how costly the error is for each of the i points. Rederive the least squares estimates b_0 and b_1 under this situation. Note that these estimates are called the *weighted least squares regression* estimates. This variant $\mathcal{A}$ on OLS has a number of practical uses, especially in Economics. No need to simplify your answers like I did in class (i.e. you can leave in ugly sums.

SSWE $= \sum_{i=1}^{n} w_i(y_i - (b_0 + b_1 x_i)^2$
SSWE $= \sum_{i=1}^{n} w_i(y_i^2 + b_0^2 + b_1^2 x_i^2 - 2y_i b_0 - 2y_i b_1 x_i + 2b_0 b_1 x_i^2)$

$$= \sum_{i=1}^{n} y_i^2 w_i + b_0^2 \sum_{i=1}^{n} w_i + b_1^2 \sum_{i=1}^{n} w_i y_i^2 - 2b_0 \sum_{i=1}^{n} w_i y_i - 2b_1 \sum_{i=1}^{n} w_i x_i y_i + 2b_0 b_1 \sum_{i=1}^{n} w_i x_i$$

$$\frac{\partial}{\partial b_0}[SSWE] = 2b_0 \sum w_i - 2 \sum w_i y_i + 2b_1 \sum w_i x_i = 0$$

$$b_0 = \frac{\sum w_i y_i}{\sum wi} - b_1 \frac{\sum w_i x_i}{\sum w_i}$$

$$\frac{\partial}{\partial b_1}[SSWE] = 2b_1 \sum w_i y_i^2 - 2 \sum w_i x_i y_i + 2b_0 \sum w_i x_i = 0$$

replace b_0 with $\frac{\sum w_i y_i}{\sum wi} - b_1 \frac{\sum w_i x_i}{\sum w_i}$

$$2b_1 \sum w_i y_i^2 - 2 \sum w_i x_i y_i + 2(\frac{\sum w_i y_i}{\sum wi} - b_1 \frac{\sum w_i x_i}{\sum w_i}) \sum w_i x_i = 0$$

$$b_1 \sum w_i y_i^2 - b_1 \frac{(\sum w_i x_i)^2}{\sum w_i} = \sum w_i x_i y_i - \frac{(\sum w_i y_i)(\sum w_i x_i)}{\sum wi}$$

$$b_1 = \frac{\sum w_i x_i y_i - \frac{(\sum w_i y_i)(\sum w_i x_i)}{\sum wi}}{\sum w_i y_i^2 - \frac{(\sum w_i x_i)^2}{\sum w_i}}$$

(i) [E.C.] Interpret the ugly sums in the b_0 and b_1 you derived above and compare them to the b_0 and b_1 estimates in OLS. Does it make sense each term should be altered in this matter given your goal in the weighted least squares?