MATH 390.4 / 650.2 Spring 2018 Homework #2t

Professor Adam Kapelner

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?

$\mathcal{H} = \mathbb{1}_{\vec{w} \odot \vec{x} + b > 0}, \vec{w} \in \mathbb{R}^p, b \in \mathbb{R}$

(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 Perceptron tries to minimize the error while the SVM tries to maximize the margin

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

using Hesse normal form $\mathcal{H} = \mathbb{1}_{\vec{w}\vec{x} - b = 0} : \vec{w} \in \mathbb{R}, b \in \mathbb{R}$. Since data is linearly seperable, we can draw two "support vector lines" to create the boundary lines for the "support vector machine" line/plane. The equation of the SVM line/plane is

$$\vec{w} \cdot \vec{x} - b = 0 \tag{1}$$

The equation of the uuper line is

$$\vec{w} \cdot \vec{x} - (b + \delta) = 0 \tag{2}$$

and the equation of the lower line is

$$\vec{w} \cdot \vec{x} - (b - \delta) = 0 \tag{3}$$

where δ is the distance between the each boundary line and the middle line.

Now we constrain all $y = 1$ to be greater than or equal to the upper line and all $y = -1$ to be less than or equal to the lower line.

$$y_i * (\vec{w} \cdot \vec{x} - b) \geq y_i * \delta \tag{4}$$

$\forall$i s.t $y_i = 1$, $y_i * (\vec{w} \cdot \vec{x} - b) \geq \delta$
and $\forall$i s.t $y_i = -1$, $-1 * (\vec{w} \cdot \vec{x} - b) \leq -1 * \delta$. Multiply both sides by -1, flip the inequality sign and we get $\vec{w} \cdot \vec{x} - b \geq \delta$

(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.

Problem 2

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

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

Given the usual suspets, we want to predict $\hat{y}^* = g(\vec{x}^*)$. KNN finds a predefined amount of $x_i \in \mathbb{D}$ and returns $\hat{y} = MODEL[y(1), ...y(k)]$ where each y_i represents the "closest" or most "similiar" x_is. Since k tells $\mathbb{A}$ what to return, it can be considered a tuning knob or a hyper parameter.

(b) [difficult] Assuming $\mathcal{A}$ = KNN, describe the input $\mathcal{H}$ as best as you can.
The set of all classes we can classify the data with.

(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).

Using $k = 1$ for predictions on $\mathbb{D}$, there should always be zero error. This is because $\mathbb{A}$ is taking in a variable from $\mathbb{D}$ and using that same set $\mathbb{D}$ to pull the nearest neighbor from and the nearest neighbor to anything is always itself. Having zero error is a bad thing because it means that your model will fail when exposed to future data since we "overfitted" the model, made it to specific.

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 $\mathcal{Y}$?

$\mathcal{X} = [x_1, x_2, ..., x_n]$ and $\mathcal{Y} = [y_1, y_2, ...y_n]$ where $\forall y_i : y \in \mathbb{R}$. Then $\mathbb{D} = [< 1, 0 >, < x_1, y_1 >, ... < x_n, y_n >]$

(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.
We know that the line fit using OLS is

$$g(x) = y = b_0 + b_1 x \tag{5}$$

and

$$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i \tag{6}$$

and

$$\bar{y} = \frac{1}{n} \sum_{i=1}^{n} y_i \tag{7}$$

$g(x)$ comes from minimimizing the SSE equation

$$\sum_{i=1}^{n}(y_i - (b_0 + b_1 x_i))^2 \tag{8}$$

If we foil this out

$$\sum_{i=1}^{n} y_i^2 + b_0^2 + b_1^2 \sum_{i=1}^{n} x_i^2 - 2b_0 \sum_{i=1}^{n} y_i - 2b_1 \sum_{i=1}^{n} y_i x_i + 2b_0 b_1 \sum_{i=1}^{n} x_i \tag{9}$$

Now we can replace all the indivdivial sums of x_i and y_i with equations $n * \bar{x}$ and $n * \bar{y}$ respectively, also distribute sum of constants.

$$\sum_{i=1}^{n} y_i^2 + nb_0^2 + b_1^2 \sum_{i=1}^{n} x_i^2 - 2nb_0\bar{y} - 2b_1 \sum_{i=1}^{n} y_i x_i + 2nb_0 b_1 \bar{x} \tag{10}$$

To get the OLS line, we need to minimize this equation. Take the partial derivate of the equation above with respect to b_0 and set it equal to zero

$$-2n\bar{y} + 2nb_0 + 2nb_1\bar{x} = 0 \tag{11}$$

Solving for b_0 gives us

$$b_0 = \bar{y} - b_1\bar{x} \tag{12}$$

Since any line in 2D space is a linear combination of it's constants,the line will pass through all points that are multiples of these linear combiniations. Since one of our constants, b_0 is a function of $\bar{y}$ and $\bar{x}$, our OLS line $g(x) = b_0 + b_1 x$ has to pass through the point $< \bar{x}, \bar{y} >$

(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 $\bar{y}$.

During the OLS process, we are generating the line that has the sum of its' indivudal residuals(distance from the ith point on the line to the ith data point) equal to 0, $\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = 0$. If we are prediciting for some $x_i \in \mathbb{D}$, then we should use the sample average $\bar{y_i}$. Since on average $\sum_{i=1}^{n} y_i = \sum_{i=1}^{n} \hat{y}_i$, we can say that $\bar{y} = \bar{y_i}$, our average prediction for x_i is $\bar{y}$

(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.

OLS line is $y = b_0 + b_1 x$. where

$$b_0 = \bar{y} - b_1\bar{x} \tag{1}$$

and

$$\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i \tag{2}$$

and

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{3}$$

We want to prove that

$$\sum_{i=1}^{n} e_i = \sum_{i=1}^{n}(y_i - b_0 - b_1 x_i) = 0 \tag{4}$$

Plug b_0 into equation 4

$$\sum_{i=1}^{n}(y_i - \bar{y} + b_1\bar{x} - b_i x_i)$$

Distribute sum and Pull out the constants

$$\sum_{i=1}^{n} y_i - \bar{y}\sum_{i=1}^{n} 1 + b_1\bar{x}\sum_{i=1}^{n} 1 - b_1\sum_{i=1}^{n} x_i$$

Rewrite

$$\sum_{i=1}^{n} y_i - n\bar{y} + nb_1\bar{x} - b_1\sum_{i=1}^{n} x_i$$

Plug equation 2 and 3 in

$$\sum_{i=1}^{n} y_i - (\frac{1}{n}\sum_{i=1}^{n} y_i)n + (\frac{1}{n}\sum_{i=1}^{n} x_i)nb_1 - b_1\sum_{i=1}^{n} x_i$$

cancel the n's

$$\sum_{i=1}^{n} e_i = \sum_{i=1}^{n} y_i - \sum_{i=1}^{n} y_i + b_1\sum_{i=1}^{n} x_i - b_1\sum_{i=1}^{n} x_i = 0$$

Hence

$$\sum_{i=1}^{n} e_i = 0 \tag{5}$$

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

R^2 is relative because it is unit less, so there is no way to compare models using this metric. Additionally, the R^2 value gets inflated the more features we add to our model.
R^2
RMSE measures how close your predicted values are the real data we are trying to model so RMSE can be viewed as a measure of how accurate your model is. RMSE is sesitive to large errors-penalizes based on the size of the error. RMSE is easier to communicate to someone who klnows nothing about statistics because its' reported value is in the same units as the depenedent variables being modelled.

(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!
$\mathbb{D} = <1, 5>, <2, 5>, <3, 5>$ and $\vec{w} = [5, -1]$

(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.

(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?