New York University Tandon School of Engineering
Computer Science and Engineering

CS-UY 4563: Midterm Exam 1.
Monday, Mar. 9th, 2020, 9:00 - 10:15pm
50 Total Points

Directions

- Show all of your work to receive full (and partial) credit.
- If more space is required, you may use extra sheets of paper clearly marked with your name, netid, and the problem you are working on.

1. Always, Sometimes, Never. (12pts – 3pts each)

Indicate whether each of the following statements is ALWAYS true, SOMETIMES true, or NEVER true. **No justification is necessary to receive full credit for a correct answer**. To earn partial credit if you are wrong, you may provide a short justification or example to explain your choice.

(a) The empirical risk of a model is lower than the population risk.

ALWAYS **SOMETIMES** NEVER

Empirical risk is a random quantity which depends on the data. Even when it's equal to the population risk in expectation, it could be larger or smaller depending on chance.

(b) You train a multiple linear regression model with varying levels of ℓ_2 regularization. Let $\vec{\beta}^{(1)} = \arg\min_{\vec{\beta}} \|X\vec{\beta} - \vec{y}\|_2^2 + \lambda_1\|\vec{\beta}\|_2^2$ and let $\vec{\beta}^{(2)} = \arg\min_{\vec{\beta}} \|X\vec{\beta} - \vec{y}\|_2^2 + \lambda_2\|\vec{\beta}\|_2^2$.

If $\lambda_1 > \lambda_2$, is $\|X\vec{\beta}^{(1)} - \vec{y}\|_2^2 < \|X\vec{\beta}^{(2)} - \vec{y}\|_2^2$?

ALWAYS SOMETIMES **NEVER**

By the optimality of $\vec{\beta}^{(1)}$ we have that $\|X\vec{\beta}^{(1)} - \vec{y}\|_2^2 + \lambda_1\|\vec{\beta}^{(1)}\|_2^2 \leq \|X\vec{\beta}^{(2)} - \vec{y}\|_2^2 + \lambda_1\|\vec{\beta}^{(2)}\|_2^2$. At the same time, By the optimality of $\vec{\beta}^{(b)}$, we have that $\|X\vec{\beta}^{(2)} - \vec{y}\|_2^2 + \lambda_2\|\vec{\beta}^{(2)}\|_2^2 \leq \|X\vec{\beta}^{(1)} - \vec{y}\|_2^2 + \lambda_2\|\vec{\beta}^{(1)}\|_2^2$. Negating this inequality gives $-\|X\vec{\beta}^{(1)} - \vec{y}\|_2^2 - \lambda_2\|\vec{\beta}^{(1)}\|_2^2 \leq -\|X\vec{\beta}^{(2)} - \vec{y}\|_2^2 - \lambda_2\|\vec{\beta}^{(2)}\|_2^2$. Adding the two inequalities gives $(\lambda_1 - \lambda_2)\|\vec{\beta}^{(1)}\|_2^2 \leq (\lambda_1 - \lambda_2)\|\vec{\beta}^{(2)}\|_2^2$, which means that $\|\vec{\beta}^{(1)}\|_2^2 \leq \|\vec{\beta}^{(2)}\|_2^2$ since $(\lambda_1 - \lambda_2)$ is positive.

But as we already noted, we also have that $\|X\vec{\beta}^{(2)} - \vec{y}\|_2^2 + \lambda_2\|\vec{\beta}^{(2)}\|_2^2 \leq \|X\vec{\beta}^{(1)} - \vec{y}\|_2^2 + \lambda_2\|\vec{\beta}^{(1)}\|_2^2$. If $\|\vec{\beta}^{(1)}\|_2^2 \leq \|\vec{\beta}^{(2)}\|_2^2$, this can only be true if $\|X\vec{\beta}^{(2)} - \vec{y}\|_2^2 \leq \|X\vec{\beta}^{(1)} - \vec{y}\|_2^2$.

(c) The linear classifier found by logistic regression minimizes error rate (0-1 loss) on the training data.

ALWAYS **SOMETIMES** NEVER

In general it does not, unless we get lucky. It minimizes the logistic loss. An answer of NEVER would also be accepted.

(d) Consider a multiple linear regression problem where each data example has the form $(\vec{x}, y) = ([x_1, x_2], y)$. Transform the predictor variables by adding quadratic terms, so each new data example has the form $(\vec{x}_{trans}, y) = ([x_1, x_2, x_1^2, x_2^2, x_1x_2], y)$. Let L^* be the minimum training loss for the original problem and let L^*_{trans} be the minimum training loss for the transformed problem. Is $L^*_{trans} \leq L^*$?

ALWAYS SOMETIMES NEVER

For any $\vec{\beta} = [\beta_1, \beta_2]$, $L_{trans}([\beta_1, \beta_2, 0, 0, 0]) = L([\beta_1, \beta_2])$. So it is always possible to find parameters that ensure $L_{trans} \leq L$. It must therefore be that $\min L_{trans} \leq \min L$.

2. Model Diagnosis Short Answer (8pts)

You are trying to solve a prediction problem using a multiple linear regression model with ℓ_2 loss. You first split the data set into a train set (80%) and a test set (20%). You then train the model on the train set to obtain a parameter vector $\vec{\beta}$. Using $\vec{\beta}$, you evaluate the average squared loss of the regression model on the train and test set, separately.

For each of the following scenarios, circle all answers that apply. **No justification is necessary to receive full credit for a correct answer**. To earn partial credit if you are wrong, you may provide a short justification.

(a) (4pts) The average squared loss on the train set is 1.5 and the average squared loss on the test set is 12.6. **Which of the following techniques is likely to improve your average test loss?**

REGULARIZATION **FEATURE SELECTION** FEATURE TRANSFORM DATA SCALING

We appear to be overfitting since our train loss is much less than our test loss. Regularization or feature selection would be the most appropriate cures. Feature transformation typically leads to a richer model, so only makes overfitting worse. And data scaling doesn't do anything for linear regression!

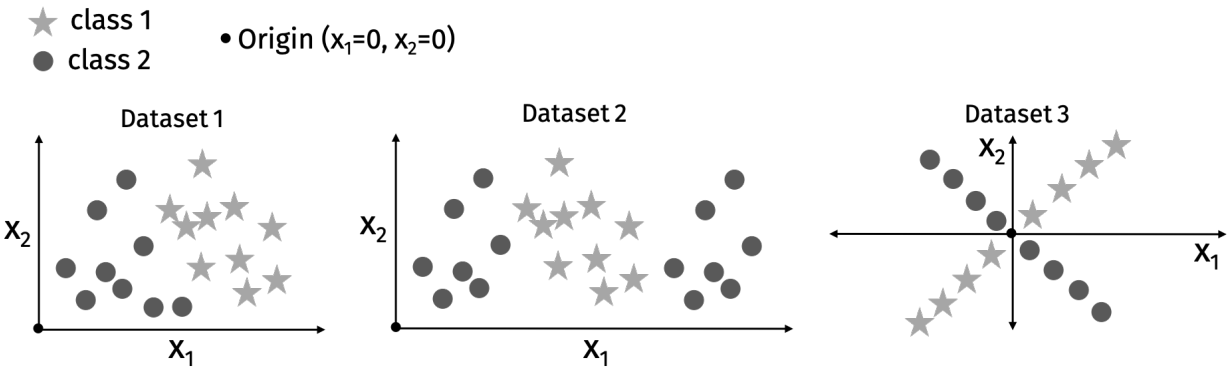
(b) (4pts) The average squared loss on the train set is 10.2 and the average squared loss on the test set is 9.9. **Which of the following techniques is likely to improve your average test loss?**

REGULARIZATION FEATURE SELECTION **FEATURE TRANSFORM** DATA SCALING

We appear to be performing poorly on both train and test loss, so we might need a richer model. The best cure would therefore be some sort of feature transformation. Of course, it's not guaranteed to work, but would be worth a try.

3. Model Diagnosis 2 (10pts)

Consider the following scatter plots of data for three binary classification problems. x_1 and x_2 are the independent variables and class labels are indicated by points with a different shape and shade.



(a) (4pts) Indicate which of the three clustering problems could be solved to high accuracy (small error rate) using a logistic regression model with no regularization and no feature transformations.

Logistic regression will perform well on any dataset which can be classified using a *linear classifier*. This would include **only Dataset 1**. Logistic regression will perform poorly on Datasets 2 and 3.

(b) (6pts) For any of the problems that you believe are not *directly solvable* with logistic regression, suggest a possible feature transformation which *would make it possible* to obtain a high accuracy solution with logistic regression. For each problem, your solution should be a set of new features $\phi_1(x_1, x_2), \phi_2(x_1, x_2), \ldots, \phi_q(x_1, x_2)$ that depend on the original features x_1 and x_2. You may use as large a q as you need.

There are many possible answers here:

Dataset 2 $\phi_1 = x_1, \phi_2 = x_2, \phi_3 = x_1^2, \phi_4 = x_2^2, \phi_5 = 1$ would work since it would allow use to classify using a shifted oval: e.g. classify star if $(x_1 - 2)^2 + .8(x_2 - 1)^2 < .5$.

Dataset 3 $\phi_1 = x_1/x_2$ works. For Class 1 this is always ≈ 1 and for Class 2 it is ≈ -1, so these values can be seperated via a linear classifier.

4. Loss Minimization. (10pts)

For data with one predictor and one target: $(x_1, y_1), \ldots, (x_n, y_n)$, consider a linear regression model:

$$f_{\beta_0,\beta_1}(x) = \beta_0 + \beta_1 x$$

with *exponential loss*:

$$L(\beta_0, \beta_1) = \sum_{i=1}^{n} e^{(y_i - f_{\beta_0,\beta_1}(x_i))^2}$$

(a) (5pts) Write down an expression for the gradient of the loss L.

We can compute both partial derivatives to get the gradient. For both we need to use chain rule:

$$\frac{\partial L}{\partial \beta_0} = \sum_{i=1}^{n} -2 \cdot (y_i - \beta_0 - \beta_1 x_i) \cdot e^{(y_i - \beta_0 - \beta_1 x_i)^2}$$

$$\frac{\partial L}{\partial \beta_1} = \sum_{i=1}^{n} -2x_i \cdot (y_i - \beta_0 - \beta_1 x_i) \cdot e^{(y_i - \beta_0 - \beta_1 x_i)^2}$$

And then we have $\nabla L(\beta_0, \beta_1) = \begin{bmatrix} \partial L/\partial \beta_0 \\ \partial L/\partial \beta_1 \end{bmatrix}$.

If you wanted to do everything in matrix form, let $\mathbf{X}$ be a data matrix with first column containing $x_1, \ldots, x_n$ and second column containing all 1s. Then let $\vec{w} = (\vec{y} - \mathbf{X}\vec{\beta}) \cdot exp(\vec{y} - \mathbf{X}\vec{\beta})$ where all operations are applied entrywise. Then the gradient is $\mathbf{X}^T \vec{w}$.

(b) (2pts) Name two algorithms/methods which could be used to minimize L.

- Brute force search.
- Gradient descent.

(c) (3pts) In general, is this exponential loss more or less robust to outliers when compared to ℓ_2 loss? How about when compared to ℓ_∞ loss?

If the linear model under exponential loss is effected *more* by an outlier, we say the loss is *less* robust.

It is therefore less robust than ℓ_2 loss because the loss function punishes outliers significantly more: you pay a cost of $e^{\text{squared error}}$ instead of just the square error. It will thus try to more closely fit outliers. On the other hand, the exponential loss is more robust than ℓ_∞ loss, which punishes outliers pretty much as much as possible: we only care about the loss of the worst data example.

5. Bayesian Crab Classification (10pts)

A biologist is collecting specimens from two species of crabs, species S_0 and S_1. These species live in the same habitat and look similar to the human eye. To accelerate crab sorting by species, the biologist wants to develop a simple classification rule based on body measurements. She observes that the ratio of *forehead breadth* to overall *body length* differs between crabs in species S_0 and S_1. The biologist proposes to measure this ratio (denoted by R) and use it as a single predictor variable for classification.

The biologist assumes that the crab data comes from a "mixture of Gaussians" probabilistic model. In particular, she assumes that for each species, R follows a normal (Gaussian) probability distribution, with different parameters for each species. The biologist makes the following concrete observations:

- 35% of all crabs collected belong to S_0 and the remaining 65% belong to S_1.
- For crabs in S_0, the average value of R is .5. For crabs in S_1, the average value of R is .4.
- For both species, the standard deviation of R is .1.

(a) (6pts) Suppose we collect a new crab with forehead breadth to body length ratio R_{new}. The biologist would like to assign this crab to S_0 or S_1 using a maximum a posterior (MAP) classification rule. Denote this rule by $f : \mathbb{R} \to \{S_0, S_1\}$. The rule takes as input the ratio R_{new} and outputs S_0 or S_1.

Write down all mathematical expressions that would need to be evaluated to compute f for a given input R_{new}. Your expressions do not need to be simplified, but they should not involve unknown variables besides R_{new}. **Hint:** Use Bayes rule.

To implement a MAP estimator we need to compute:

$$\Pr(S_0 \mid R_{new}) \qquad \text{and} \qquad \Pr(S_1 \mid R_{new})$$

Using Bayes rule we have:

$$\Pr(S_0 \mid R_{new}) = \frac{\Pr(R_{new} \mid S_0)\Pr(S_0)}{\Pr(R_{new})} \qquad \text{and} \qquad \Pr(S_1 \mid R_{new}) = \frac{\Pr(R_{new} \mid S_1)\Pr(S_1)}{\Pr(R_{new})}$$

Note that, we have:

$$\Pr(S_0) = .35 \qquad \text{and} \qquad \Pr(S_1) = .65.$$

And using the equation for the Gaussian distribution, we also have:

$$\Pr(R_{new} \mid S_0) = \frac{1}{\sqrt{2\pi}\cdot .1} e^{-\frac{(R_{new}-.5)^2}{2\cdot .1^2}} \qquad \text{and} \qquad \Pr(R_{new} \mid S_1) = \frac{1}{\sqrt{2\pi}\cdot .1} e^{-\frac{(R_{new}-.4)^2}{2\cdot .1^2}}.$$

(b) (4pts) Show that, for this problem, the classification rule f has the following form:

$$f(R_{new}) = \begin{cases} S_0 \text{ if } R_{new} \geq \lambda \\ S_1 \text{ if } R_{new} < \lambda, \end{cases}$$

for some fixed threshold parameter λ (you do not need to explicitly compute λ).

Note: A less formal argument than what I give below would suffice.

The MAP classification rule f is as follows:

$$f(R_{new}) = \begin{cases} S_0 \text{ if } \Pr(S_0 \mid R_{new}) \geq \Pr(S_1 \mid R_{new}), \\ S_1 \text{ if } \Pr(S_0 \mid R_{new}) < \Pr(S_1 \mid R_{new}). \end{cases}$$

Substituting in our equations from part (a), we see that we will classify a crab with ratio R_{new} into class S_0 under this rule if:

$$f(R_{new}) = S_0 \text{ if } \quad \frac{.35\frac{1}{\sqrt{2\pi}\cdot.1}e^{-\frac{(R_{new}-.5)^2}{.02}}}{\Pr(R_{new})} \geq \frac{.65\frac{1}{\sqrt{2\pi}\cdot.1}e^{-\frac{(R_{new}-.4)^2}{.02}}}{\Pr(R_{new})}$$

which is equivalent to checking if

$$f(R_{new}) = S_0 \text{ if } \quad .35e^{-\frac{(R_{new}-.5)^2}{.02}} \geq .65e^{-\frac{(R_{new}-.4)^2}{.02}}.$$

Taking logs and rearranging, this in turn is equivalent to checking:

$$f(R_{new}) = S_0 \text{ if } \quad (R_{new}-.4)^2 - (R_{new}-.5)^2 \geq .02\log(.65) - .02\log(.35)$$
$$f(R_{new}) = S_0 \text{ if } \quad .2 \cdot R_{new} \geq .02\log(.65) - .02\log(.35) - .16 + .25$$

So clearly we classify in S_0 if $R_{new} \geq \lambda$ for some λ.

(c) (3pts – extra credit) Given the biologist's data above, will the threshold λ for the MAP classification rule be EQUAL TO, LARGER, or SMALLER than .45? Justify your answer in a sentence or two. This problem can be solved without a calculator.

λ will be **LARGER** than .45. It would equal .45 exactly if $\Pr(S_0) = \Pr(S_1) = .5$. However, since the prior probability of $\Pr(S_1)$ is larger, we will classify an example as S_1 even if its ratio slightly exceeds .45. To see this from the calculations above, note that $(-.16 + .25)/.2 = .45$ and $\log(.65) - \log(.35)$ is positive. So $\lambda > .45$. The actually threshold would be around .512.