# 10301/601 Homework 6 (Written)

Joong Ho Choi

TOTAL POINTS

**65.5 / 79**

QUESTION 1

1 1 **0 / 0**

✓ **- 1 pts** Incorrect

QUESTION 2

## CNNs and RNNs 18 pts

2.1 2.1.a **0 / 1**

✓ **- 1 pts** Incorrect

2.2 2.1.b **1 / 1**

✓ **- 0 pts** Correct

2.3 2.2.a **1 / 1**

✓ **- 0 pts** Correct

2.4 2.2.b **1 / 1**

✓ **- 0 pts** Correct

2.5 2.2.c **1 / 1**

✓ **- 0 pts** Correct

2.6 2.2.d **1 / 1**

✓ **- 0 pts** Correct

2.7 2.2.e **0 / 1**

✓ **- 1 pts** Blank/Incorrect

2.8 2.2.f **1 / 1**

✓ **- 0 pts** Correct (anything about finding features across different parts of the image, translational invariance, or translational equivariance)

2.9 2.3.a **2.75 / 3**

✓ **- 0.25 pts** Nodes are not rectangles

2.10 2.3.b.i **2 / 2**

✓ **- 0 pts** Correct: $$\hat{y_t}- y_t$$

2.11 2.3.b.ii **1.5 / 2**

✓ **- 0.5 pts** Correct terms with shape mismatch or other minor error

2.12 2.3.b.iii **0 / 2**

✓ **- 2 pts** Incorrect

2.13 2.3.b.iv **0 / 1**

✓ **- 1 pts** Incorrect

QUESTION 3

## Learning Theory 18 pts

3.1 3.1.a **2 / 2**

✓ **- 0 pts** Correct

3.2 3.1.b **3 / 3**

✓ **- 0 pts** Correct

3.3 3.2.a **2 / 2**

✓ **- 0 pts** Correct

3.4 3.2.b **2 / 2**

✓ **- 0 pts** Correct

3.5 3.3 **3 / 3**

✓ **- 0 pts** Correct, Paul is right, with justification.

3.6 3.4.a **3 / 3**

✓ **- 0 pts** Correct

3.7 3.4.b **3 / 3**

✓ **- 0 pts** Correct

QUESTION 4

## MLE/MAP 24 pts

### 4.1 4.1 2 / 2
✓ - 0 pts Correct

### 4.2 4.2 2 / 2
✓ - 0 pts Correct

### 4.3 4.3 2 / 2
✓ - 0 pts Correct: (a)

### 4.4 4.4 2 / 2
✓ - 0 pts Correct

### 4.5 4.5 3 / 3
✓ - 0 pts Correct

### 4.6 4.6 1.5 / 3
✓ - 0.5 pts missed (b)
✓ - 0.5 pts missed (d)
✓ - 0.5 pts missed (e)

### 4.7 4.7 3 / 3
✓ - 0 pts Correct: (d)

### 4.8 4.8 2 / 2
✓ - 0 pts Correct

### 4.9 4.9 0 / 2
✓ - 2 pts Incorrect/Blank

### 4.10 4.10 3 / 3
✓ - 0 pts Correct

QUESTION 5

## Naive Bayes 19 pts

### 5.1 5.1 3 / 3
✓ - 0 pts Correct: 0.0680

### 5.2 5.2.a 0 / 2
✓ - 2 pts Blank/Incorrect

### 5.3 5.2.b 2 / 2
✓ - 0 pts Correct: 0.0417

### 5.4 5.2.c 1 / 1
✓ - 0 pts Correct

### 5.5 5.3 2.25 / 3
✓ - 0.75 pts missed (b)

### 5.6 5.4.a 3 / 3
✓ - 0 pts Correct

### 5.7 5.4.b 2 / 2
✓ - 0 pts Correct

### 5.8 5.5 1.5 / 3
✓ - 0.5 pts missed (b)
✓ - 0.5 pts added (c)
✓ - 0.5 pts missed (d)

QUESTION 6

## 6 Collaboration 0 / 0
✓ - 0 pts Correct

# HOMEWORK 6: LEARNING THEORY AND GENERATIVE MODELS

10-301/10-601 Introduction to Machine Learning (Spring 2022)
http://www.cs.cmu.edu/~mgormley/courses/10601/
OUT: Friday, March 18th
DUE: Friday, March 25th
TAs: Zachary, Sami, Yuxin, Junhui, Rita

Homework 6 covers topics on learning theory, MLE/MAP, Naive Bayes, and CNNs and RNNs. The homework includes multiple choice, True/False, and short answer questions.

## START HERE: Instructions

- **Collaboration Policy**: Please read the collaboration policy here: http://www.cs.cmu.edu/~mgormley/courses/10601/syllabus.html
- **Late Submission Policy:** See the late submission policy here: http://www.cs.cmu.edu/~mgormley/courses/10601/syllabus.html
- **Submitting your work:** You will use Gradescope to submit answers to all questions and code. Please follow instructions at the end of this PDF to correctly submit all your code to Gradescope.
  - **Written:** For written problems such as short answer, multiple choice, derivations, proofs, or plots, please use the provided template. Submissions can be handwritten onto the template, but should be labeled and clearly legible. If your writing is not legible, you will not be awarded marks. Alternatively, submissions can be written in LaTeX. Each derivation/proof should be completed in the boxes provided. You are responsible for ensuring that your submission contains exactly the same number of pages and the same alignment as our PDF template. If you do not follow the template, your assignment may not be graded correctly by our AI assisted grader.
  - **Programming:** You will submit your code for programming questions on the homework to Gradescope (https://gradescope.com). After uploading your code, our grading scripts will autograde your assignment by running your program on a virtual machine (VM). When you are developing, check that the version number of the programming language environment (e.g. Python 3.9.6) and versions of permitted libraries (e.g. `numpy` 1.21.2 and `scipy` 1.7.1) match those used on Gradescope. You have 10 free Gradescope programming submissions. After 10 submissions, you will begin to lose points from your total programming score. We recommend debugging your implementation on your local machine (or the Linux servers) and making sure your code is running correctly first before submitting your code to Gradescope.
- **Materials:** The data that you will need in order to complete this assignment is posted along with the writeup and template on the course website.

## Instructions for Specific Problem Types

For "Select One" questions, please fill in the appropriate bubble completely:

**Select One:** Who taught this course?

- ● Matt Gormley
- ○ Marie Curie
- ○ Noam Chomsky

If you need to change your answer, you may cross out the previous answer and bubble in the new answer:

**Select One:** Who taught this course?

- ● Matt Gormley
- ○ Marie Curie
- ⊗ Noam Chomsky

For "Select all that apply" questions, please fill in all appropriate squares completely:

**Select all that apply:** Which are scientists?

- ■ Stephen Hawking
- ■ Albert Einstein
- ■ Isaac Newton
- □ I don't know

Again, if you need to change your answer, you may cross out the previous answer(s) and bubble in the new answer(s):

**Select all that apply:** Which are scientists?

- ■ Stephen Hawking
- ■ Albert Einstein
- ■ Isaac Newton
- ⊠ I don't know

For questions where you must fill in a blank, please make sure your final answer is fully included in the given space. You may cross out answers or parts of answers, but the final answer must still be within the given space.

**Fill in the blank:** What is the course number?

10-601

10-~~6~~301

# Written Questions (80 points)

## 1 Latex Bonus Point (1 points)

1. (1 point) **Select one:** Did you use Latex for the entire written portion of this homework?

   ● Yes

   ○ No

## 2 CNNs and RNNs (18 points)

1. In this problem, consider only the convolutional layer of a standard implementation of a CNN as described in Lecture 13. We are given image $X$ and filter $F$ below.

$X =$

| | | | | | |
|---|---|---|---|---|---|
| 1 | 0 | -2 | 3 | 4 | 1 |
| 2 | 9 | 5 | 6 | 0 | -1 |
| 0 | -3 | 1 | 3 | 4 | 4 |
| 6 | 5 | 2 | 0 | 6 | 8 |
| -5 | 4 | -3 | 1 | 3 | -2 |
| 4 | 1 | 2 | 8 | 9 | 7 |

$F =$

| | | |
|---|---|---|
| -1 | -1 | -1 |
| -1 | 8 | -1 |
| -1 | -1 | -1 |

$Y =$

| | | | |
|---|---|---|---|
| a | b | c | d |
| e | f | g | h |
| i | j | k | l |
| m | n | o | p |

   (a) (1 point) Let $X$ be convolved with $F$ using no padding and a stride of 1 to produce an output $Y$. What is value of $j$ in the output $Y$?

   **Answer**

   38

   (b) (1 point) Suppose you had an input feature map of size (height $\times$ width) 6x4 and filter size 2x2, using no padding and a stride of 2, what would be the resulting output size? Write your answer in the format height $\times$ width.

   **Answer**

   3 x 2

2. Parameter sharing is a very important concept for CNN because it drastically reduces the complexity of the learning problem. For the following questions, assume that there is no bias term in our convolutional layer.

   (a) (1 point) Which of the following are parameters of a convolutional layer?
   **Select all that apply:**

   - ☐ stride size
   - ☐ padding size
   - ☐ image size
   - ☐ filter size
   - ■ weights in the filter
   - ☐ None of above.

   (b) (1 point) Which of the following are hyperparameters of a convolutional layer?
   **Select all that apply:**

   - ■ stride size
   - ■ padding size
   - ☐ image size
   - ■ filter size
   - ☐ weights in the filter
   - ☐ None of above.

   (c) (1 point) Suppose for the convolutional layer, we are given grayscale images of size $22 \times 22$. Using one single $4 \times 4$ filter with a stride of 2 and no padding, what is the number of parameters you are learning in this layer?

   Answer: 16

   (d) (1 point) Now suppose we do no parameter sharing. That is, each output pixel of this layer is computed by a separate $4 \times 4$ filter. Again we use a stride of 2 and no padding. What is the number of parameters you are learning in this layer?

   Answer: 1600

   (e) (1 point) Now suppose you are given a $40 \times 40$ colored image, which consists of 3 channels (so your input is a $40 \times 40 \times 3$ tensor), each representing the intensity of one primary color. Suppose you again do no parameter sharing, using a unique $4 \times 4$ filter per output pixel, with a stride of 2 and no padding. What is the number of parameters you are learning in this layer?

**Answer**

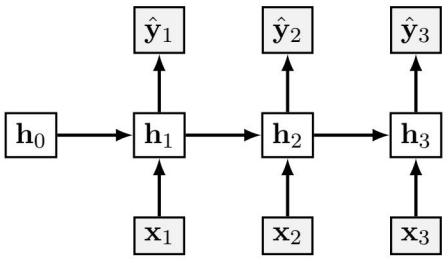
4800

(f) (1 point) In a sentence, describe a reason why parameter sharing is a good idea for a convolutional layer applied to image data, besides the reduction in number of learned parameters.

**Answer**

By doing a parameter sharing, the way of extracting information from image data remains consistent throughout. This helps pick up on similar features from the data

3. Consider the following simple RNN architecture:




Where the layers and their corresponding weights are given below:

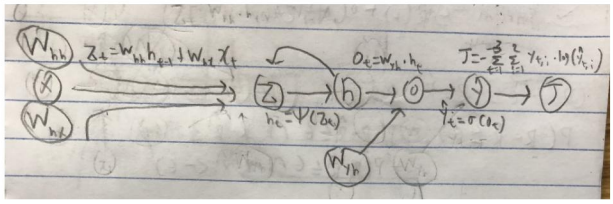
$$\mathbf{x}_t \in \mathbb{R}^3 \qquad \mathbf{W}_{hx} \in \mathbb{R}^{4\times 3}$$
$$\mathbf{h}_t \in \mathbb{R}^4 \qquad \mathbf{W}_{yh} \in \mathbb{R}^{2\times 4}$$
$$\mathbf{y}_t, \hat{\mathbf{y}}_t \in \mathbb{R}^2 \qquad \mathbf{W}_{hh} \in \mathbb{R}^{4\times 4}$$

$$J = -\sum_{t=1}^{3}\sum_{i=1}^{2} y_{t,i}\log(\hat{y}_{t,i})$$
$$\hat{\mathbf{y}}_t = \sigma(\mathbf{o}_t)$$
$$\mathbf{o}_t = \mathbf{W}_{yh}\mathbf{h}_t$$
$$\mathbf{h}_t = \psi(\mathbf{z}_t)$$
$$\mathbf{z}_t = \mathbf{W}_{hh}\mathbf{h}_{t-1} + \mathbf{W}_{hx}\mathbf{x}_t$$

Where $\sigma$ is the softmax activation and $\psi$ is the identity activation (i.e. no activation). Note here that we assume that we have no intercept term.

(a) (3 points) Draw the computational graph for the given model. *(Hint: think about how you can express the recurrence in the model through clever placement of the directed edges in the graph, rather than separating the graph out across timesteps.)*

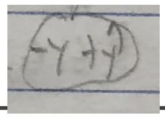
Answer



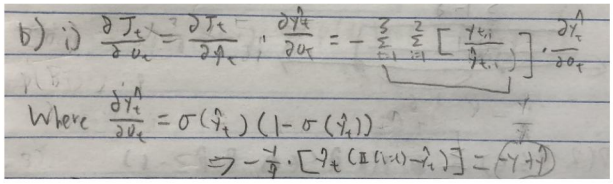
(b) Now you will derive the steps of the backpropagation algorithm that lead to the computation of $\frac{dJ}{dW_{hh}}$. For all parts of this question, please write your answer in terms of $W_{hh}$, $W_{yh}$, $y$, $\hat{y}$, $h$, and any additional terms specified in the question (note: this does not mean that every term listed shows up in every answer, but rather that you should simplify terms into these as much as possible when you can).

i. (2 points) Let $J_t = -\sum_{i=1}^{2} y_{t,i} \log(\hat{y}_{t,i})$. What is $\frac{\partial J_t}{\partial o_t}$? Write your solution in the first box, and show your work in the second.

$\frac{\partial J_t}{\partial \mathbf{o}_t}$

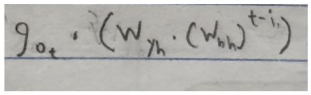


Work



ii. (2 points) Suppose you have a variable $\mathbf{g}_{\mathbf{o}_t}$ that stores the value of $\frac{\partial J_t}{\partial \mathbf{o}_t}$. What is $\frac{\partial J_t}{\partial \mathbf{h}_i}$ for an arbitrary $i \in [1, 3]$? Write your solution in terms of the $\mathbf{g}_{\mathbf{o}_t}$ and the aforementioned variables in the first box, and show your work in the second.

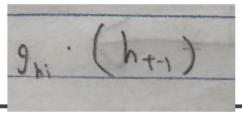
$\frac{\partial J_t}{\partial \mathbf{h}_i}$



Work

iii. (2 points) Suppose you have a variable $\mathbf{g}_{\mathbf{h}_i}$ that stores the value of $\frac{\partial J_t}{\partial \mathbf{h}_i}$. What is $\frac{\partial J_t}{\partial \mathbf{W}_{hh}}$? Write your solution in terms of the $\mathbf{g}_{\mathbf{h}_i}$ and the aforementioned variables in the first box, and show your work in the second.

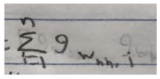
$\frac{\partial J_t}{\partial W_{hh}}$



Work

iv. (1 point) Suppose you have a variable $\mathbf{g}_{W_{hh},t}$ that stores the value of $\frac{\partial J_t}{\partial W_{hh}}$. What is $\frac{\partial J}{\partial \mathbf{W}_{hh}}$? Write your solution in terms of the $\mathbf{g}_{W_{hh},t}$ and the aforementioned variables in the first box, and show your work in the second.

$\frac{\partial J}{\partial W_{hh}}$



Work

# 3 Learning Theory (18 points)

1. Alex is given a classification task to solve. He has no idea where to start, so he decided to try out a decision tree learner with 2 binary features $X_1$ and $X_2$. On the other hand, Sally thinks Alex shouldn't have used a decision tree with 2 binary features. Instead, she thinks it would be best to use logistic regression with 16 real-valued features in addition to a bias term. Sally overherd Alex talking about this cool concept called PAC learning and she too would like to use it to analyze her method. She first trains her logistic regression model on $N$ examples to obtain a training error $\hat{R}$.

   (a) (2 points) What is the the upper bound on the true error $R$ in terms of $\hat{R}$, $\delta$, and $N$. You may use big-$\mathcal{O}$ notation.

   **Answer**

   R $\leq \hat{R}$ +$\mathcal{O}(\sqrt{((1/N) * [17 + ln(1/\delta)) )}$

   (b) (3 points) Sally wants to argue her method has lower bound on the true error. Assuming Sally has obtained enough data points to satisfy PAC criterion with $\epsilon = 0.1$ and $\delta = 0.01$. Which of the following is true?
   **Select one:**

   - ◯ Sally is wrong. Alex's method will always classify unseen data more accurately since it is simpler as it only needs 2 binary features.
   - ◯ She must first regularize her model by removing 14 features to make any comparison at all.
   - ◯ It is sufficient to show that the VC Dimension of her classifier is higher than Alex's, therefore having lower bound for the true error.
   - ● It is necessary to show that the training error she achieves is lower than the training error Alex achieves.

2. In lecture, we saw that we can use our sample complexity bounds to derive bounds on the true error for a particular algorithm. Use the sample complexity bound for the infinite, agnostic case:

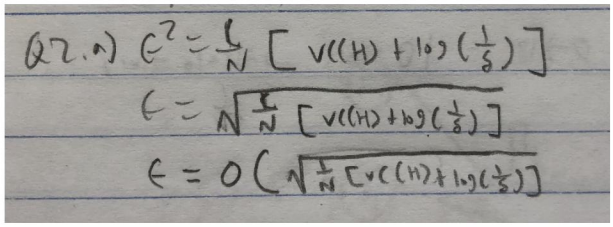
$$N = O\left(\frac{1}{\epsilon^2}\left[\text{VC}(\mathcal{H}) + \log(\frac{1}{\delta})\right]\right)$$

   to prove that with probability at least $(1 - \delta)$:

$$R(h) \leq \hat{R}(h) + O\left(\sqrt{\frac{1}{N}\left[\text{VC}(\mathcal{H}) + \log(\frac{1}{\delta})\right]}\right)$$

   (a) (2 points) Rewrite the big-O bound in terms of $N$ and $\delta$ using the definition of big-O notation (i.e. if $N = O(M)$ (for some value $M$), then there exists a $c \in \mathbb{R}$ such that $N \leq cM$).

Answer

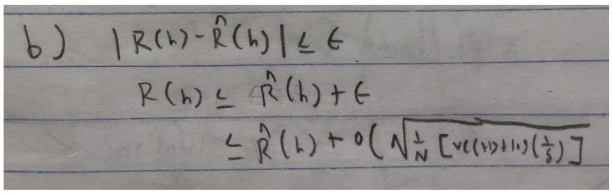



(b) (2 points) Now, using the definition of $\epsilon$ (i.e. $|R(h) - \hat{R}(h)| \leq \epsilon$), show that with probability at least $(1 - \delta)$:

$$R(h) \leq \hat{R}(h) + O\left(\sqrt{\frac{1}{N}\left[\text{VC}(\mathcal{H}) + \log(\frac{1}{\delta})\right]}\right)$$

.

Answer




3. (3 points) Consider the hypothesis space of functions that map $M$ binary attributes to a binary label. A function $f$ in this space can be characterized as $f : \{0,1\}^M \Rightarrow \{0,1\}$. Your friend Paul says that no matter the value of $M$, a function from this space can always shatter $2^M$ points. Is Paul wrong? If so, provide a counterexample. If Paul is right, briefly explain why in 1-2 *concise* sentences.

Answer

$\{0,1\}^M$ has $2^M$ unique inputs and outputs are binary(either 0 or 1). Decision tree of depth M has $2^M$ nodes

4. Consider instance space $\mathcal{X}$ which is the set of real numbers.

   (a) (3 points) What is the VC dimension of hypothesis class $H$, where each hypothesis $h$ in $H$ is of the form "if $a < x < b$ or $c < x < d$ then $y = 1$; otherwise $y = 0$"? (i.e., $H$ is an infinite hypothesis class where $a, b, c$, and $d$ are arbitrary real numbers).

   **Select one:**

   ○ 2

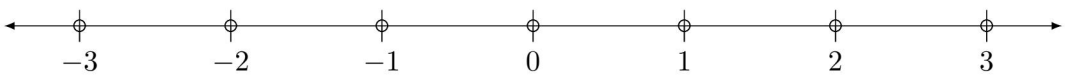
   ○ 3

   ● 4

   ○ 5

   ○ 6

   (b) (3 points) Given the set of points in $\mathcal{X}$ below, construct a labeling of some subset of the points to show that any dimension larger than your choice of VC dimension in part (a) by *exactly* 1 is incorrect (e.g. if the VC dimension of $H$ is 3, only fill in the answers for 4 of the points). Fill in the boxes such that for each point in your example, the corresponding label is either 1 or 0 (for points you are not using in your example, leave the boxes blank).




-3: [ ]

-2: [1]

-1: [0]

0: [1]

1: [0]

2: [1]

3: [ ]

# 4 MLE/MAP (24 points)

1. (2 points) **True or False:** Suppose you place a Beta prior over the Bernoulli distribution, and attempt to learn the parameter of the Bernoulli distribution from data. Further suppose an adversary chooses "bad", but finite hyperparameters for your Beta prior in order to confuse your learning algorithm. As the number of training examples grows to infinity, the MAP estimate of $\theta$ can still converge to the MLE estimate of $\theta$.

   **Select One:**

   - ● True
   - ○ False

2. (2 points) Let $\Gamma$ be a random variable with the following probability density function (pdf):

$$f(\gamma) = \begin{cases} 2\gamma & \text{if } 0 \leq \gamma \leq 1 \\ 0 & \text{otherwise} \end{cases}$$

   Suppose another random variable Y, which is conditioning on $\Gamma$, follows an exponential distribution with $\lambda = 3\gamma$. Recall that the exponential distribution with parameter $\lambda$ has the following pdf:

$$f_{exp}(y) = \begin{cases} \lambda e^{-\lambda y} & \text{if } y \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

   What is the MAP estimate of $\gamma$ given $Y = \frac{2}{3}$ is observed?

   **Select one:**

   - ○ 0
   - ○ 1/3
   - ● 1
   - ○ 2

3. (2 points) In HW3, you have derived the closed form solution for linear regression. Now, we are coming back to linear regression, viewing it as a statistical model, and deriving the MLE and MAP estimate of the parameters in the following questions.

   As a reminder, in MLE, we have

$$\hat{\theta}_{MLE} = \underset{\theta}{\operatorname{argmax}}\, p(\mathcal{D}|\theta)$$

   For MAP, we have

$$\hat{\theta}_{MAP} = \underset{\theta}{\operatorname{argmax}}\, p(\theta|\mathcal{D})$$

   Assume we have data $D = \{\mathbf{x}^{(i)}, y^{(i)}\}_{i=1}^{N}$, where $\mathbf{x}^{(i)} = (x_1^{(i)}, \cdots, x_M^{(i)})$ . So our data has $N$ instances and each instance has $M$ attributes/features. Each $y^{(i)}$ is generated given $\mathbf{x}^{(i)}$ with additive noise $\epsilon^{(i)} \sim$

$N(0, \sigma^2)$, that is $y^{(i)} = \mathbf{w}^T\mathbf{x}^{(i)} + \epsilon^{(i)}$ where $\mathbf{w}$ is the parameter vector of linear regression. Given this assumption, what is the distribution of y?

**Select one:**

- ● $y^{(i)} \sim N(\mathbf{w}^T\mathbf{x}^{(i)}, \sigma^2)$
- ○ $y^{(i)} \sim N(0, \sigma^2)$
- ○ $y^{(i)} \sim \text{Uniform}(\mathbf{w}^T\mathbf{x}^{(i)} - \sigma, \mathbf{w}^T\mathbf{x}^{(i)} + \sigma)$
- ○ None of the above

4. (2 points) The next step is to learn the MLE of the parameters of the linear regression model. Which expression below is the correct conditional log likelihood $\ell(\mathbf{w})$ with the given data?

**Select one:**

- ● $\sum_{i=1}^{N}[-\log(\sqrt{2\pi\sigma^2}) - \frac{1}{2\sigma^2}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)})^2]$
- ○ $\sum_{i=1}^{N}[\log(\sqrt{2\pi\sigma^2}) + \frac{1}{2\sigma^2}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)})^2]$
- ○ $\sum_{i=1}^{N}[-\log(\sqrt{2\pi\sigma^2}) - \frac{1}{2\sigma^2}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)})]$
- ○ $-\log(\sqrt{2\pi\sigma^2}) + \sum_{i=1}^{N}[-\frac{1}{2\sigma^2}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)})^2]$

5. (3 points) Then, the MLE of the parameters is just $\text{argmax}_{\mathbf{w}}\, \ell(\mathbf{w})$ . Among the following expressions, select ALL that can yield the correct MLE.

**Select all that apply:**

- □ $\text{argmax}_{\mathbf{w}} \sum_{i=1}^{N}[-\log(\sqrt{2\pi\sigma^2}) - \frac{1}{2\sigma^2}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)})]$
- ■ $\text{argmax}_{\mathbf{w}} \sum_{i=1}^{N}[-\log(\sqrt{2\pi\sigma^2}) - \frac{1}{2\sigma^2}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)})^2]$
- ■ $\text{argmax}_{\mathbf{w}} \sum_{i=1}^{N}[-\frac{1}{2\sigma^2}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)})^2]$
- □ $\text{argmax}_{\mathbf{w}} \sum_{i=1}^{N}[-\frac{1}{2}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)})]$
- ■ $\text{argmax}_{\mathbf{w}} \sum_{i=1}^{N}[-\frac{1}{2}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)})^2]$

6. (3 points) Now we are moving on to learn the MAP estimate of the parameters of the linear regression model. Which expression below is the correct optimization problem the MAP estimate is trying to solving? (recall that $D$ refers to the data, and $\mathbf{w}$ to the regression parameters (weights)).

**Select all that apply:**

- ■ $\mathbf{w}_{MAP} = \arg\max_{\mathbf{w}} p(D, \mathbf{w})$
- □ $\mathbf{w}_{MAP} = \arg\max_{\mathbf{w}} \frac{p(D|\mathbf{w})p(\mathbf{w})}{p(D)}$
- □ $\mathbf{w}_{MAP} = \arg\max_{\mathbf{w}} \frac{p(D,\mathbf{w})}{p(\mathbf{w})}$
- □ $\mathbf{w}_{MAP} = \arg\max_{\mathbf{w}} p(D|\mathbf{w})p(\mathbf{w})$
- □ $\mathbf{w}_{MAP} = \arg\max_{\mathbf{w}} p(\mathbf{w}|D)$

7. (3 points) Suppose we are using a Gaussian prior distribution with mean 0 and variance $\frac{1}{\lambda}$ for each element $w_m$ of the parameter vector $\mathbf{w}(1 \leq m \leq M)$, i.e. $w_m \sim \mathcal{N}(0, \frac{1}{\lambda})$. Assume that $w_1, \cdots, w_M$ are mutually independent of each other. Which expression below is the correct log joint-probability of the data and parameters $\log p(D, \mathbf{w})$)? Please show your work below.

**Select one:**

- ○ $\sum_{i=1}^{N} \left( -\log(\sqrt{2\pi\sigma^2}) - \frac{1}{2\sigma^2}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)})^2 \right) - \sum_{m=1}^{M} \log(\sqrt{2\pi\lambda}) - \lambda(w_m)^2$
- ○ $\sum_{i=1}^{N} \left( -\log(\sqrt{2\pi\sigma^2}) - \frac{1}{2\sigma^2}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)}) \right) + \sum_{m=1}^{M} -\log(\sqrt{2\pi\lambda}) - \lambda(w_m)^2$
- ○ $\sum_{i=1}^{N} \left( -\log(\sqrt{2\pi\sigma^2}) - \frac{1}{2\sigma^2}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)}) \right) - \sum_{m=1}^{M} \log(\sqrt{\frac{2\pi}{\lambda}}) - \frac{\lambda}{2}(w_m)^2$
- ● $\sum_{i=1}^{N} \left( -\log(\sqrt{2\pi\sigma^2}) - \frac{1}{2\sigma^2}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)})^2 \right) + \sum_{m=1}^{M} -\log(\sqrt{\frac{2\pi}{\lambda}}) - \frac{\lambda}{2}(w_m)^2$

Answer

8. (2 points) Consider the same linear regression model with a Gaussian prior on the parameters as in the previous question. Maximizing the log posterior probability $\ell_{MAP}(\mathbf{w})$ gives you the MAP estimate of the parameters. Which of the following is an equivalent definition of $\max_{\mathbf{w}} \ell_{MAP}(\mathbf{w})$?

**Select one:**

- ○ $\max_{\mathbf{w}} \sum_{i=1}^{N} \frac{1}{2\sigma^2}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)})^2 + \frac{\lambda}{2}\|\mathbf{w}\|_2$
- ● $\min_{\mathbf{w}} \sum_{i=1}^{N} \frac{1}{2\sigma^2}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)})^2 + \frac{\lambda}{2}\|\mathbf{w}\|_2^2$
- ○ $\max_{\mathbf{w}} \sum_{i=1}^{N} \frac{1}{2\sigma^2}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)})^2 + \lambda\|\mathbf{w}\|_2$
- ○ $\min_{\mathbf{w}} -\sum_{i=1}^{N} \frac{1}{2\sigma^2}(y^{(i)} - \mathbf{w}^T\mathbf{x}^{(i)})^2 - \frac{\lambda}{2}\|\mathbf{w}\|_2^2$

9. (2 points) A MAP estimator with a Gaussian prior $\mathcal{N}(0, \sigma^2)$ you trained gives significantly higher test error than train error. What could be a possible approach to fixing this?

**Select one:**

- ● Increase variance $\sigma^2$
- ○ Decrease variance $\sigma^2$
- ○ Try MLE estimator instead
- ○ None of the above

10. (3 points) MAP estimation with what prior is equivalent to L1 regularization? Please show your work below.

Note:
The pdf of a Uniform distribution over $[a, b]$ is $f(x) = \frac{1}{b-a}$ if $x \in [a, b]$ and 0 otherwise.
The pdf of an exponential distribution with rate parameter $a$ is $f(x) = a \exp(-ax)$ for $x > 0$.
The pdf of a Laplace distribution with location parameter $a$ and scale parameter $b$ is $f(x) = \frac{1}{2b} \exp\left(\frac{-|x-a|}{b}\right)$ for all $x \in \mathbb{R}$.

**Select one:**

- ○ Uniform distribution over $[-\mathbf{w}^T\mathbf{x}^{(i)}, \mathbf{w}^T\mathbf{x}^{(i)}]$
- ○ Exponential distribution with rate parameter $a = \frac{1}{2}$
- ○ Exponential distribution with rate parameter $a = \mathbf{w}^T\mathbf{x}^{(i)}$
- ● Laplace prior with location parameter $a = 0$
- ○ Laplace prior with location parameter $a = \mathbf{w}^T\mathbf{x}^{(i)}$
- ○ Uniform distribution over [-1, 1]

Your answer

# 5 Naïve Bayes (19 points)

1. (3 points) Suppose that 0.3% of all people have cancer. If someone is tested for cancer, the outcome of the test can either be positive (cancer) or negative (no cancer). The test is not perfect - among people who have cancer, the test comes back positive 97% of the time. Among people who don't have cancer, the test comes back positive 4% of the time. For this question, you should assume that the test results are independent of each other, given the true state (cancer or no cancer). What is the probability of a test subject having cancer, given that the subject's test result is positive?

   If your answer is in decimals, answer with precision 4, e.g. (6.051, 0.1230, 1.234e+7)

   **Fill in the blank:**

   0.0680

2. The following dataset describes several features of a person and then whether or not they are a fan of the K-pop boy band, BTS.

| Hair Color | Height | BTS Fan? |
|---|---|---|
| Blonde | Low | N |
| Brown | Low | N |
| Brown | Low | Y |
| Brown | Medium | Y |
| Blonde | Medium | N |
| Black | Medium | Y |
| Black | High | Y |
| Brown | High | Y |

   Zack, a new friend that you've met, has brown hair and medium level height. We would like to determine whether they are a BTS fan, using the naïve bayes assumption to estimate the following probabilities.

   (a) (2 points) What is the probability that someone has brown hair, medium height, and is a BTS fan?
   **Fill in the blank:**

   3/20

   (b) (2 points) What is the probability that someone has brown hair, medium height, and is *not* a BTS fan?
   **Fill in the blank:**

   1/24

   (c) (1 point) Is Zack a BTS fan?
   **Select one:**

   - ● Yes
   - ○ No

3. (3 points) In a Naive Bayes problem, suppose we are trying to compute $P(Y \mid X_1, X_2, X_3, X_4)$. Furthermore, suppose $X_2$ and $X_3$ are identical (i.e., $X_3$ is just a copy of $X_2$). Finally, assume $X_2$ is not

independent of $Y$. Which of the following are true in this case?

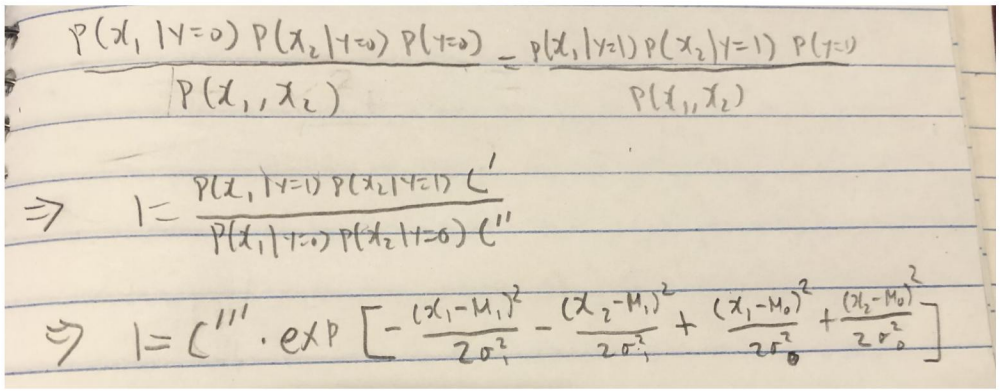
**Select all that apply:**

- ■ Naive Bayes will learn identical parameter values for $P(X_2|Y)$ and $P(X_3|Y)$.
- □ Naive Bayes will output probabilities $P(Y|X_1, X_2, X_3, X_4)$ that are closer to 0 and 1 than they would be if we removed the feature corresponding to $X_3$.
- □ There is not enough information to determine the change in the output $P(Y|X_1, X_2, X_3, X_4)$.
- □ None of the above

4. We've seen in class that a Gaussian Naïve Bayes classifier can learn more than just a *linear* decision boundary.

   (a) (3 points) Show that the decision boundary for a 2-D Gaussian Naïve Bayes classifier, $p(y = 1 \mid x_1, x_2) = p(y = 0 \mid x_1, x_2)$, is quadratic. That is, show that $p(y = 1 \mid x_1, x_2) = p(y = 0 \mid x_1, x_2)$ can be written as a polynomial function of $x_1$ and $x_2$ where the degree of each variable is at most 2. (In your proof, you may fold constants into terms $C', C'', C'''$ so long as you are clearly showing each step.)

Answer




   (b) (2 points) Select all possible decision boundary that can be produced by a Gaussian Naïve Bayes classifier. The shaded region is assigned class 1 and the unshaded regions is assigned class 0.

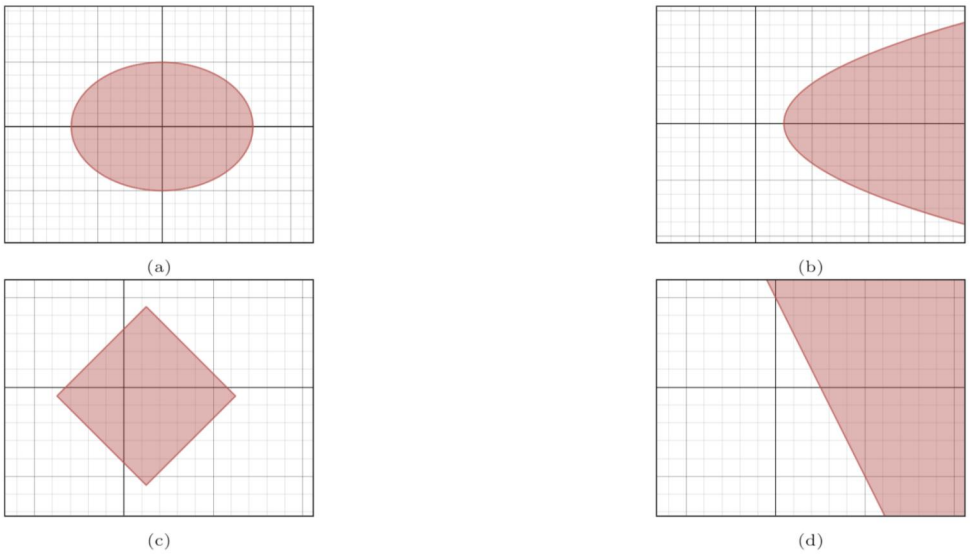



Figure 1: Decision Boundaries

**Select all that apply:**

- ■ (a)
- ■ (b)
- □ (c)
- ■ (d)
- □ None of the above

5. (3 points) Gaussian Naive Bayes in general can learn non-linear decision boundaries. Consider the simple case where we have just one real-valued feature $X_1 \in \mathbb{R}$ from which we wish to infer the value of label $Y \in \{0, 1\}$.The corresponding generative story would be:

   $Y \sim$ Bernoulli$(\phi)$
   $X_1 \sim$ Gaussian$(\mu_y, \sigma_y^2)$
   where the parameters are the Bernoulli parameter $\phi$ and the class-conditional Gaussian parameters $\mu_0, \sigma_0^2$ and $\mu_1, \sigma_1^2$ corresponding to $Y = 0$ and $Y = 1$ , respectively.

   A linear decision boundary in one dimension, of course, can be described by a rule of the form "if $X_1 > c$ then $Y = k$, else $Y = 1 - k$", where $c$ is a real-valued threshold and $k \in \{0, 1\}$. Is it possible in this simple one-dimensional case to construct a Gaussian Naive Bayes classifier with a decision boundary that cannot be expressed by a rule in the above form?

   **Select all that apply:**

   - ☐ Yes, this can occur if the Gaussians are of equal means and equal variances.
   - ☐ Yes, this can occur if the Gaussians are of equal means and unequal variances.
   - ■ Yes, this can occur if the Gaussians are of unequal means and equal variances.
   - ☐ Yes, this can occur if the Gaussians are of unequal means and unequal variances.
   - ☐ None of the above.

## 6 Collaboration Questions

After you have completed all other components of this assignment, report your answers to these questions regarding the collaboration policy. Details of the policy can be found here.

1. Did you receive any help whatsoever from anyone in solving this assignment? If so, include full details.
2. Did you give any help whatsoever to anyone in solving this assignment? If so, include full details.
3. Did you find or come across code that implements any part of this assignment? If so, include full details.

Your Answer