ECE 47300: Midterm 2 (Spring 2023), Version: B

Instructions

Please fill out this cover page but do NOT start until instructed to do so.

What is your full name and Purdue ID (10-digit number)?

Name (please PRINT, no cursive): Paloma Arellano

Purdue ID (10 digits): 0031926985

Honor Pledge Signature: PA

- **Total number of points**: 100 (Roughly 50 questions or subquestions)
- **Question format notes**
  - Circular checkboxes are for multiple choice (i.e., a single correct answer)
  - Square checkboxes are for select all questions (i.e., possibly multiple correct answers).
  - Only put your answer in the boxes like [ ] .
- **No partial credit**: Every (sub-)question is all or nothing credit. Thus, you must get the answer exactly right to get credit for the question (including SELECT ALL questions). No partial credit will be given.
- **Number Format**: When giving numbers as short answers, please give in standard decimal notation with preceding "0." for decimals if needed but no trailing 0s (e.g., "0.15", "2.9", "0.001", "100" but NOT "0.15000" NOR ".15" NOR ".001" NOR "6.0").
- **Honor Pledge**: I assert that I have not received any information about this exam and will not share any exam content with anyone else, even after the exam. I understand that any violation of this will result in a failing grade for the whole class (not just this exam).

1. (2 points) Unlike other layers, if the layer parameters are frozen, a BatchNorm layer behaves differently during training and testing/evaluation.

   - (●) True
   - ( ) False

2. (2 points) Suppose the input tensor is an ImageNet image (224×224 size with 3 channels) and we apply a "3 × 3" convolution with 9 filters, no padding, and a stride of 1, how many channels will the output of the convolution have?

   Answer: 3

3. (2 points) Which statements are true for BatchNorm? (Select all that apply)

   - [ ] For spatial BatchNorm, the mean is obtained by averaging over the channel dimension.
   - [■] If the input shape is (B, C, H, W), the computed standard deviation of spatial BatchNorm has a shape of (C,).
   - [■] BatchNorm normalizes the minibatch so that the features have a mean of zero and variance of one and then applies a shift and scale.

4. (2 points) Suppose the input shape to a spatial BatchNorm layer is (16, 32, 8, 8), corresponding to batch, channels, height and width dimensions. What is the number of **trainable** parameters updated by gradient descent of this BatchNorm layer?

   - (●) 16
   - ( ) 32
   - ( ) 64
   - ( ) 128

5. (2 points) Which statements about probability density functions are always true?

   - [■] $p(y|x) = p(x,y)/p(x)$
   - [ ] $p(y)p(y|x) = p(x)p(x|y)$
   - [■] $p(x)p(y|x) = p(y)p(x|y)$
   - [ ] $p(x|y) = p(y|x)$

6. (2 points) Suppose we have a probability mass function (PMF) over binary random variables defined as:

$$P(x,y) = \begin{bmatrix} P(0,1) = 0.5 & P(1,1) = ? \\ P(0,0) = 0.2 & P(1,0) = 0.2 \end{bmatrix} \quad (1)$$

What is the value of $P(y=0|x=1)+P(y=1|x=1)$?
Note: $P(1,1)$ can be inferred from other values.

- (●) 1
- ( ) 0.7
- ( ) 0.5
- ( ) 0.2
- ( ) 0

7. (2 points) Let X and Y be discrete random variables. If we know that $P(X=2|Y=1) = 0.8$, $P(Y=1) = 0.3$, and $P(X=2) = 0.4$, what is $P(Y=1|X=2)$?

- ( ) 1
- ( ) 0.6
- (●) 0.24
- ( ) 0.096
- ( ) 0
- ( ) Insufficient information.

8. (2 points) Suppose the input is an image of small resolution (8×8 spatial size with only 1 channel) and we perform a 2×2 convolution with no padding and a stride of 2. What will be the width dimension of the output?

Answer: 4

9. (2 points) Suppose $g(x) = \mathbb{E}_{z\sim q(z|x)}[f(x,z)\log f(x,z)] - \mathrm{KL}(p(z|x), q(z|x))$, where f is a concave function and p and q are conditional distributions of z given x. What can we claim without any more information?

- ( ) $g(x)$ is an upperbound of $\mathbb{E}_{z\sim q(z|x)}[f(x,z)\log f(x,z)]$
- ( ) $g(x) = \mathbb{E}_{z\sim q(z|x)}[f(x,z)\log f(x,z)]$
- (●) $g(x)$ is a lowerbound of $\mathbb{E}_{z\sim q(z|x)}[f(x,z)\log f(x,z)]$
- ( ) None of the above can be claimed without more information

10. (2 points) Let $\{x_i\}_{i=1}^n$ be samples from the true distribution p.
If $\theta^*_{\mathrm{MLE}} = \arg\max_\theta \sum_{i=1}^n \log \hat{q}(x_i;\theta)$ and $\theta^*_{\mathrm{KL}} = \arg\min_\theta \widehat{KL}(p(\cdot), \hat{q}(\cdot;\theta))$ (where the KL term is approximated using samples as in lecture), which statement is always true?

- (●) $\theta^*_{\mathrm{MLE}} = \theta^*_{\mathrm{KL}}$
- ( ) $\theta^*_{\mathrm{MLE}} > \theta^*_{\mathrm{KL}}$
- ( ) $\theta^*_{\mathrm{MLE}} < \theta^*_{\mathrm{KL}}$
- ( ) None of the statements is always true.

11. (2 points) Let X and Y be independent continuous random variables.
If we know that $p(X=2|Y=1) = 3$, $p(Y=2) = 2$, and $p(Y=1|X=3) = 5$, what is $p(X=2, Y=1)$?
(Insufficient information [-1] is a valid answer.)

Answer: 15

12. (2 points) The inception score (IS) and Frechet inception distance (FID) compare sets of images based on the output probabilities and intermediate representations of pretrained deep neural networks, respectively.

- [x] True
- [ ] False

13. (2 points) For the DCGAN model discussed in class, computing gradients of the discriminator with respect to a real batch of samples and a fake batch of samples separately is equivalent to computing the gradients of a concatenated batch of both real and fake samples.

- [ ] True
- [x] False

14. (2 points) Assuming we could solve the inner maximization of GAN objective perfectly (i.e., find the theoretic solution), optimizing the adversarial objective is the same as minimizing the JSD between P_{real} and P_{fake}.

- [x] True
- [ ] False

15. (2 points) Which of the following distributions are valid generative models for count-valued data? (Select all)

- [ ] Gaussian
- [ ] Bernoulli
- [x] Multinomial
- [ ] Mixture of Gaussians
- [x] Mixture of Multinomials

16. The LDA joint distribution can be formulated as

$$p(\theta, \beta, Z, W) = p_\eta(\beta) \prod_{i=1}^{M} p_\alpha(\theta_i) \prod_{\ell=1}^{L_i} P(z_{i,\ell}|\theta_i) P(w_{i,\ell}|\beta_{z_{i,\ell}}),$$

where β are the topic-word distributions, θ_i are the sample-specific topic distributions, W are the words in the training data, and Z are the topic assignments for each corresponding words in W.

(a) (2 points) Which of the following is the posterior distribution of LDA?

- [ ] $p(\theta|\beta, Z, W)$
- [ ] $p(\theta, \beta, Z, W)$
- [x] $p(\theta, \beta, Z|W)$
- [ ] $p(\theta, \beta, Z)$

(b) (2 points) What is the prior distribution of θ_i?

- [x] $p(\theta_i)$
- [ ] $p(z_{i,1}, z_{i,2}, \cdots, z_{i,L_i}|\theta_i)$
- [ ] $p(\theta_i|z_{i,1}, z_{i,2}, \cdots, z_{i,L_i})$
- [ ] $p(\theta_i, z_{i,1}, z_{i,2}, \cdots, z_{i,L_i})$

17. (2 points) We have learned that the key for the conditional topic assignment of a single word of LDA is two counting matrices: C^{WT} and C^{DT}, where $P(z_{i,\ell} = j | Z_{-(i,\ell)}, W) \propto \left(\frac{C^{DT}_{i,j}+\alpha}{\sum_{j'} C^{DT}_{i,j'}+\alpha}\right)\left(\frac{C^{DT}_{w_{i,\ell},j}+\eta}{\sum_{w} C^{DT}_{w,j}+\eta}\right)$. Suppose the information of the observations can be summarized as shown in the table.

Vocabulary size	1892
# of topics	10
Max length of words in documents	321
# of documents	4520

What is the shape of each counting matrix?
C^{WT}: [10] × [1892]
C^{DT}: [4520] × [10]

18. (2 points) Suppose we have near ideal word embeddings, where w_i denotes the embedding of the i-th word. Select all boxes where the equations should approximately hold.

- ☐ $w_{king} - w_{queen} \approx w_{woman} - w_{man}$
- ● $w_{fast} - w_{faster} \approx w_{long} - w_{longer}$
- ☐ $w_{father} - w_{son} \approx w_{daughter} - w_{mother}$

19. (2 points) CBOW is one of the techniques for Word2Vec. Its objective function can be formulated as:

$$\log p(y|C) = \log \sigma(A(\sum_{i \in \mathcal{C}} w_i) + b),$$

where A, b are linear layer parameters, W is a word embedding matrix, and C represents the context words, and $\mathcal{C}$ is a list of indices corresponding to each context word. Which of the following contexts C would *always* produce the same probabilities $p(y|C)$ as the context C ="happy am i" (if any)?

- ☐ C_1 = "i am very happy"
- ■ C_2 ="i am happy"
- ☐ C_3 ="we are happy"
- ■ C_4 ="happy i am"
- ☐ None of the above.

20. (2 points) Suppose we want to do the second Word2Vec task of predicting the context from the target word. Given the input sentence "the red fox jumped over the lazy dog", fill in the data table entries assuming the current center word of the window is "over", where the context is 2 words on either side. (Hint: This is before simplifying to a binary problem.)

Training Dataset

Input Word	Target Word
⋮	⋮
over	jumped
over	fox
over	the
over	lazy
⋮	⋮

21. (2 points) Assuming that A, B, C, D are 3 x 3 matrices, which of the following statements are ALWAYS true? Select all that apply.

- ☐ $A(B+C) = AB + CA$ AB+CA
- ■ $C(A+B) = CA + CB$ CA+CB
- ☐ $(AB)^T = A^T B^T$
- ■ $ABC + ABD = (AB)(D+C)$ ABD+ABC
- ■ A is not singular if and only if rank$(A) = 3$

22. (2 points) The KNN algorithm uses less computation during training rather than test/inference time.

- ● True
- ○ False

23. The following questions are related to the Convolutional Neural Network, where you need to calculate either the input or output shape in the format "CxHxW" or the filter size as "f_Hxf_W".

(a) (2 points) Given an input image with dimensions 3x224x224, what would be the size of the output feature map after applying a convolutional layer with 5 filters, a filter size of 7x7, a stride of 1, and a padding of 3? (Format: CxHxW)
Ans: 3x224x224 $\frac{224+2(3)-7}{1}+1 =$

(b) (3 points) If the input shape is 3x64x64 and the output shape is 64x32x32, what was the filter size of the convolutional layer assuming a stride of 2 and no padding? (Format: f_Hxf_W)
Ans: 2x2

(c) (2 points) If the stride of a convolutional layer is 2 and the filter size is 3x3, what would be the corresponding stride and filter size in the transpose convolutional layer that aims to undo the effect of the original convolutional layer?
Ans: Filter size (f_Hxf_W) = 2x2, Stride = 1

(d) (3 points) Given an input shape of 64x32x32, what would be the output shape of a transpose convolutional layer with 16 filters, a filter size of 5x5, a stride of 1, and a padding of 2? (Format: CxHxW)
Ans: 64x32x32

$32 = \frac{64-k}{2}+1$

$62 = 64-k$

$k = 2$

9x9 →

$= \frac{32+4-5}{1}+1$

$= 32$

24. (3 points) The code below is an incomplete template for training a classifier.

```
1 def train(classifier, epoch):
2  classifier.train()
3  for batch_idx, (images, targets) in enumerate(train_loader):
4    output = classifier(images)
5    loss = F.nll_loss(output, targets) # Negative log likelihood loss function
6
7
8
9    print(f'Current loss = {loss.item()}')
```

Write the line number where each code line should be inserted (i.e., lines 6, 7, and 8).

7 `optimizer.zero_grad()`
8 `optimizer.step()`
6 `loss.backward()`

25. (3 points) The following code snippet is intended to update a model using 2 different loss functions (MSE and Cross Entropy) by weighting them and then summing them together. The code snippet contains a mistake that doesn't raise errors but is incorrect. Which line has a mistake and what is the corrected line?

```
1    output = model(images)
2    loss = F.nll_loss(output, targets)
3    loss2 = F.mse_loss(output, targets)
4    loss_total = loss + weight_loss2 * loss2
5    loss.backward()
```

Line number: 4 Corrected line: losstotal = loss * weight_loss 2 + loss 2

26. The code below extracts the latent representations layer by layer.

```
1    def compute_and_extract_representations(self, x, ind_list):
2        # MNIST data is used, and the shape of x is (B,1,28,28).
3        z_list = []
4        for i, residual in enumerate(self.residual_layers):
5            x = residual(x)
6            x = self.maxpool(x)
7            if i in ind_list:
8                z_list += [x]
9        x = x.view(x.shape[0], -1)  # Unravel tensor dimensions
10       out = self.fc(x)
11       return out, z_list
```

3 residual layers

(a) (3 points) Assuming `self.residual_layers` are simple convolutional residual layers and `self.maxpool` is a max pool layer with a stride of 2, what should `ind_list` be to extract only the latent feature with a shape of (B,1,7,7)?

- ○ `[0]`
- ● `[-1]`
- ○ `[2]`
- ○ `[1]`

(b) (2 points) Suppose `[0,1,2]` is used for `ind_list`. What is the correct shape information of the elements in `z_list`?

- ○ `[(B,1,14,14), (B,1,7,7), (B,1,3,3)]`
- ● `[(B,1,28,28), (B,1,14,14), (B,1,7,7)]`
- ○ `[(B,1,28,28), (B,1,28,28), (B,1,28,28)]` ✗

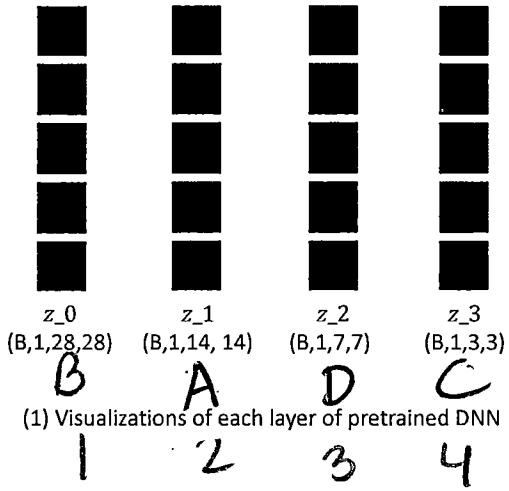
(c) (2 points) Assuming our model is trained with MNIST dataset for classification (10 classes), and `self.fc` is the last layer. What is the correct input and output size of the last layer, which is defined as `self.fc = nn.Linear(in_ch, out_ch)` in the `__init__` function?

- ○ `in_ch=1, out_ch=1` ✗
- ○ `in_ch=1, out_ch=10` ✗
- ● `in_ch=B*9, out_ch=1`
- ○ `in_ch=B*9, out_ch=10` ✗
- ○ `in_ch=9, out_ch=10` ✗
- ○ `in_ch=49, out_ch=10` ✗

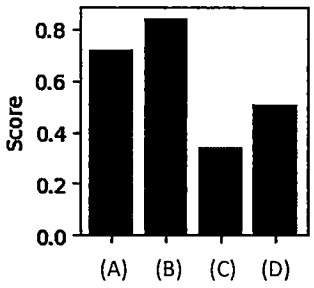
(d) (1 point) Suppose we want to append the original `x` (i.e., the original image) to the list of `z_list`. What is the correct answer for where to put the line "`z_list += [x]`"?

- ○ After line 2 and before line 3
- ● After line 3 and before line 4
- ○ After line 4 and before line 5
- ○ After line 9 and before line 10

27. We have visualized the latent features of a pretrained Deep Neural Network (DNN), where z_0 corresponds to the original image. The right figure shows the clustering score for each latent representation as in the homework. Solve the questions referring to the following figure.



(1) Visualizations of each layer of pretrained DNN



(2) Clustering score

(a) (2 points) What should be the latent representations corresponding to (A)-(D) in the right subfigure (2)? Answers below should be arranged in an order of (A), (B), (C) and (D).

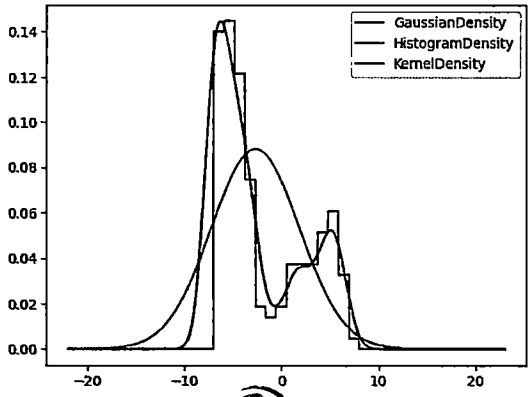
- ○ z_0, z_1, z_2, z_3
- ○ z_3, z_2, z_1, z_0
- ● z_1, z_0, z_3, z_2 ✓
- ○ z_2, z_3, z_0, z_1 ✗

(b) (2 points) Select correct statements on the latent space of DNN.

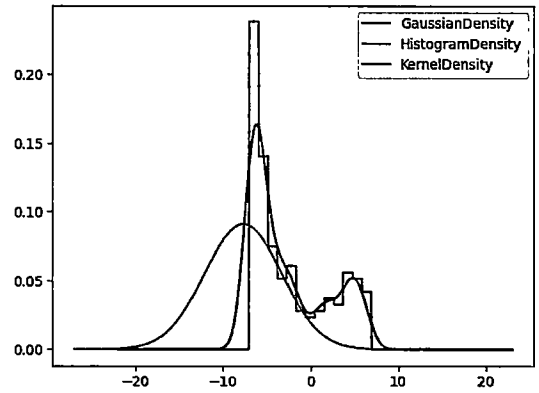
- ■ Deeper representations usually encode more high-level information.
- ■ A pretrained DNN can be leveraged as a feature extractor.
- □ The Euclidean distance between two points is the same in all latent spaces.

28. (3 points) Each figure (A)-(D) correspond to the density plots for different datasets. However, one or more of the plots may contain a coding bug, i.e., the different density estimates are not consistent with each other—or put another way, the density estimates could not have come from the same training data. Which plots could be correct?

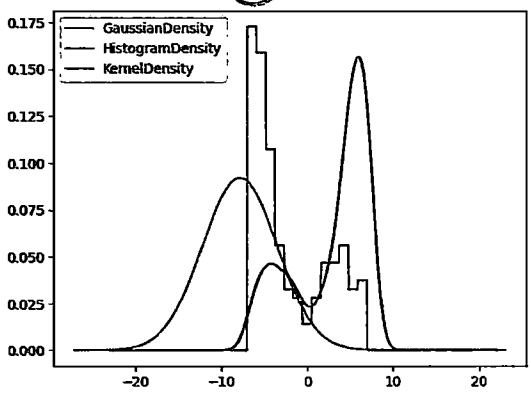
AP (Format example: If A, C, and D are correct, write "ACD")



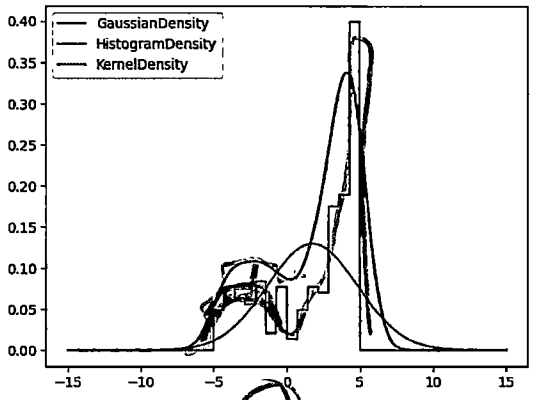
A



B



C



D

29. Given a list of points X_train, estimate the density and return the probability corresponding to the given points.

(a) (4 points) Fit a histogram density model for X_train with min = 0, max = 6, bin width = 2, where X_train is:

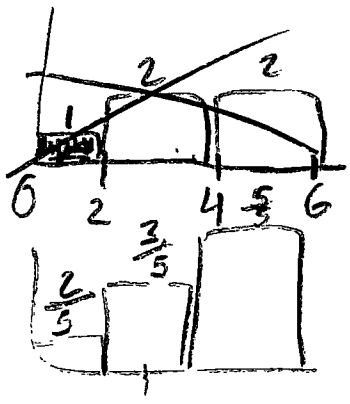
```
X_train = [0.2, 1.8, 3.4, 4.2, 5.8]
```

[0,2) [2,4) [4,6)

What are the histogram PDF values for the following test points?

- x = -2 : 0
- x = 6.2: 1
- x = 2.2: 0.6
- x = 5.4: 1

2, 1, 2



(b) (4 points) Fit a Kernel density model on X_train where the kernel is a *Uniform* distribution centered around the training point, i.e., Uniform($[x_i - 0.5, x_i + 0.5]$), where x_i is a training point. X_train is:

```
X_train = [0.1, 0.9, 1.7, 2.1, 2.9]
```

What are the kernel density PDF values for the following test points?

- x = -1 : 0
- x = 3.1: 1
- x = 1.1: 0.6
- x = 2.7: 0.8

30. (2 points) The following code is for training a VAE model. What should be the argument to bce_loss_fn?

TORCH.NN.FUNCTIONAL.BINARY_CROSS_ENTROPY

torch.nn.functional.binary_cross_entropy(*input*, *target*, *weight=None*, *size_average=None*, *reduce=None*, *reduction='mean'*) [SOURCE]

Function that measures the Binary Cross Entropy between the target and input probabilities.

See BCELoss for details.

Parameters:

- **input** (*Tensor*) – **Tensor of arbitrary shape as probabilities.**
- **target** (*Tensor*) – **Tensor of the same shape as input with values between 0 and 1.**

Torch documentation of BCELoss

```
def vae_loss(output, mu, log_var, images):
  """
  :param output: this the output of your neural network
  :param mu: this is the mu from the latent space
  :param log_var: this is the log_var from the latent space
  :param images: this is the original sets of images
  """

  bce_loss_fn = nn.BCELoss(reduction='sum')
  BCE = bce_loss_fn(XXXXXX, XXXXXXX)
  KLD = -0.5 * torch.sum(1 + log_var - mu.pow(2) - log_var.exp())

  return BCE, KLD
```

Ans: bce_loss_fn(images , mu)

31. (2 points) The following code does the reparameterization. What should be A, B?

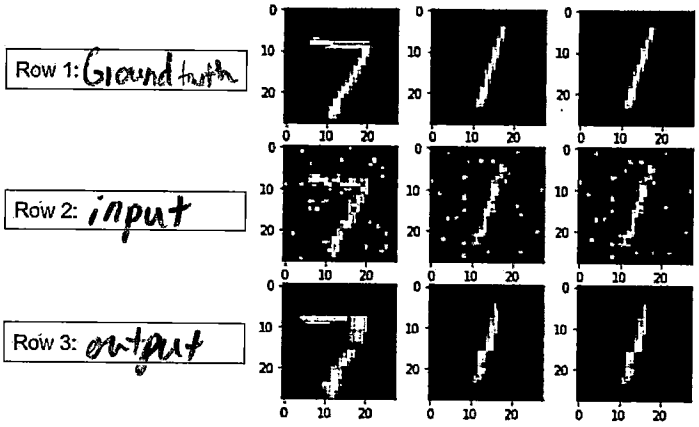
```
def reparameterize(self, mu, log_var):
    """
    :param mu: mean from the latent space
    :param log_var: the log variance from the latent space

    You should return a sample with gaussian distribution N(mu, var) using
    the reparameterization trick.
    """

    std = torch.exp(0.5*log_var)
    eps = torch.randn_like(std)
    sample = __A__ * eps + __B__

    return sample
```

0.5 · Var

5

Ans:
A = [mu], B = [std]

32. (3 points) There are three rows in the following figure corresponds to "Input" and "Output" of a trained denoising autoencoder along with the "Ground Truth". Please label each row with these 3 labels.



33. Suppose we want to train a 1D linear regression model using a mean squared error (MSE) objective. For simplicity, we will assume that the model does not include a shift/bias term and is simply $f_\theta(x) = \theta x$, where both θ and x are scalars. The training data is two points (note that X only has 1 column since this is a 1D problem):

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2.0 \\ 4.0 \end{bmatrix} \text{ and } \quad y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 5.0 \\ 10.0 \end{bmatrix} \tag{2}$$

(a) (2 points) Assuming that $\theta = 1$, what is the gradient of the parameters for sample 1?

$$\nabla_\theta \ell(y_1, f_\theta(x_1)) = \boxed{10} \tag{3}$$

(b) (2 points) Assuming that $\theta = 1$, what is the gradient of the parameters for sample 2?

$$\nabla_\theta \ell(y_2, f_\theta(x_2)) = \boxed{40} \tag{4}$$

(c) (2 points) Assuming that $\theta = 1$ at the current iteration and the step size is $\eta = 0.02$, what is the updated θ after one step of gradient descent?

$$\theta^{\text{new}} = \boxed{10.02}$$

(d) (2 points) Assuming that $\theta = 1$ at the current iteration, the step size is $\eta = 0.01$, and the batchsize is 1 (where the first sample is in the first batch), what is the updated θ after one step of stochastic gradient descent?

$$\theta^{\text{new}} = \boxed{0.8}$$