DS-GA-1003: Machine Learning (Spring 2023)

Final Exam (4:00pm–5:50pm, May 15)

- You should finish the exam within **1 hours and 50 minutes**.
- Closed book.

Question	Points	Score
Probabilistic models	5	
Bayesian methods	6	
Multiclass classification	7	
Decision trees and ensemble methods	7	
Gradient boosting	5	
Neural networks	8	
K-means and GMM	4	
EM	6	
Total:	48	

1. **Probabilistic models**

(a) (2 points) List 2 assumptions that are made in logistic regression

Solution:

- The dependent variable is binary or ordinal.
- The relationship between the log-odds of the dependent variable and the independent variables is linear.
- There is little or no multicolline inearity among the independent variables.
- The observations are independent of each other.

(b) (1 point) Write the partial derivative of the likelihood function with respect to θ_i for logistic regression.

Solution:

$$\frac{\partial L(\theta)}{\partial \theta_i} = \sum_{j=1}^{m} (h_\theta(x^{(j)} - y^{(j)}))x_i^{(j)}$$

(c) (1 point) What is the sigmoid function and how is it related to logistic regression?

Solution: The sigmoid function is defined as $\sigma(z) = \frac{1}{1+e^{-z}}$. In logistic regression, the sigmoid function is used to model the probability of a binary outcome by transforming the linear combination of the independent variables into a probability value between 0 and 1.

(d) (1 point) In Poisson regression, what is the transfer function typically used for?

Solution: In Poisson regression, the transfer function typically used is the logarithm function, which links the mean of the Poisson distribution to the linear combination of predictor.

2. **Bayesian methods**

(a) Multiple choices. State your explanation in one sentence.

i. (1 point) Which of the following is NOT a merit of Bayesian models?

(A) More efficient way of estimating model parameters than frequentist approaches

(B) Allow us to predict uncertainty of parameters

(C) Help us regularize the model complexity when data is scarce

(D) Allow us to predict uncertainty of future predictions

Solution: A. Bayesian models are typically more expensive to compute than frequentist models since it needs to integrate over the weight space.

Explanation:

ii. (1 point) Which of the following is the reason of using conjugate priors?

(A) Conjugate priors describe the probability of natural events more precisely.

(B) Conjugate priors make it possible to derive the posterior distribution in the same family.

(C) Conjugate priors make our parameter estimation unbiased.

(D) Conjugate priors make the posterior a proper probability distribution.

Solution: B. Conjugate priors make it possible to derive the posterior distribution in the same family so it will be easy to integrate over.

Explanation:

(b) Long answers. You are at a casino. On each round of the game, a machine generates a real number $x \in \mathcal{R}$. If the number is positive, you wins x dollars. If the number is negative, you must pay the casino x dollars. So far, you have played 3 times and

observed the following dataset:

$$\mathcal{D} = \{-5, 3, -10\}$$

Angela believes the machine is generating its numbers from a normal distribution with mean μ and variance 10:

$$x \sim \mathcal{N}(\mu, 10)$$

For this question, you may find the probability density function of the normal distribution useful:

$$\mathcal{N}(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right)$$

i. (2 points) Write the log-likelihood function $\ell(\mu) = \log p(\mathcal{D}|\mu)$.

Solution:

$$\begin{aligned}
\log p(\mathcal{D}|\mu) &= \log\{\mathcal{N}(-5|\mu, 10) \times \mathcal{N}(3|\mu, 10) \times \mathcal{N}(-10|\mu, 10)\} \\
&= \log \mathcal{N}(-5|\mu, 10) + \log \mathcal{N}(3|\mu, 10) + \log \mathcal{N}(-10|\mu, 10) \\
&= \left(-\frac{1}{2}\log(20\pi) - \frac{1}{20}(-5-\mu)^2\right) + \left(-\frac{1}{2}\log(20\pi) - \frac{1}{20}(3-\mu)^2\right) + \\
&\quad \left(-\frac{1}{2}\log(20\pi) - \frac{1}{20}(-10-\mu)^2\right) \\
&= -\frac{3}{2}\log(20\pi) - \frac{1}{20}\left((-5-\mu)^2 + (3-\mu)^2 + (-10-\mu)^2\right)
\end{aligned}$$

ii. (2 points) You believe that the casino will make you lose money in the long run. This belief is a prior distribution on μ: $p(\mu) = \mathcal{N}(\mu|-1, 5)$. Find the maximum a posteriori (MAP) estimate of the mean μ under this prior distribution.

Solution: The MAP solution μ_{MAP} satisfies:

$$\mu_{\text{MAP}} = \arg\max_{\mu} \log p(\mu) + \log p(\mathcal{D}|\mu)$$

We have:

$$\begin{aligned}
&\log p(\mu) + \log p(\mathcal{D}|\mu) \\
=&-\frac{1}{10}(\mu+1)^2 - \frac{1}{20}\left((-5-\mu)^2 + (3-\mu)^2 + (-10-\mu)^2\right) + \text{const.}
\end{aligned}$$

Taking the derivative with respect to μ:

$$\begin{aligned}
&\frac{\partial}{\partial \mu}\left\{\log p(\mu) + \log p(\mathcal{D}|\mu)\right\} \\
=&-\frac{1}{5}(\mu+1) + \frac{1}{10}\left((-5-\mu) + (3-\mu) + (-10-\mu)\right) \\
=&\frac{1}{5}(-\mu-1) + \frac{1}{10}\left(-12-3\mu\right)) \\
=&\frac{2}{10}(-\mu-1) + \frac{1}{10}\left(-12-3\mu\right)) \\
=&\frac{1}{10}(-2\mu-2) + \frac{1}{10}\left(-12-3\mu\right))
\end{aligned}$$

We set the derivative equal to zero and solve for μ:

$$\begin{aligned}
\frac{1}{10}(-2\mu-2) + \frac{1}{10}\left(-12-3\mu\right)) = 0 &\implies -2\mu - 2 - 12 - 3\mu = 0 \\
&\implies -5\mu - 14 = 0 \\
&\implies \mu = -\frac{14}{5}
\end{aligned}$$

3. **Multiclass classification.**

(a) (1 point) How many binary classifiers do you need, in order to obtain a 100-way classifier, if you choose to use **One-vs-All** (OvA) classifier?

Solution: 100

(b) (1 point) How many binary classifiers do you need, in order to obtain a 100-way classifier, if you choose to use **All-vs-All** (AvA) classifier?

Solution: $99 \times 100/2 = 4950$

(c) (1 point) Suppose each class has an equal number of examples, what are some advantages of AvA over OvA?

Solution: Balanced problem, more flexibility on the decision boundary.

(d) (1 point) Which of the following is NOT a motivation for using score functions for structured prediction?

(A) A more general formulation for modeling both x and y.
(B) Less computation than performing dot product if we choose clever features.
(C) Help us identify hard examples.
(D) Help us regularize the model from overfitting.
(E) More flexible way of modeling structures such as linear chains or trees.

Solution: D. Score functions do not help with overfitting.

(e) Recall that the generalized hinge loss is

$$l_{\text{hinge}}(y, x, w) = \max_{y'}(\Delta(y, y') - w^\top(\Psi(x, y) - \Psi(x, y')))$$

i. (1 point) What is the meaning of $\Delta(y, y')$?

Solution: The target margin between the predicted answer and the correct answer.

ii. (1 point) What is the value of $\Delta(y, y')$ in a binary class SVM?

Solution: 1

iii. (1 point) What is the meaning of $w^\top(\Psi(x, y) - \Psi(x, y'))$?

Solution: The actual margin between the predicted answer and the correct answer.

4. **Decision trees and ensemble methods**

(a) For each of the following statements, indicate whether it is true or false, and explain your answer in one sentence.

i. (1 point) Decision trees cannot be used for regression problems.

Solution: False. You can output a real value for each leaf node.

ii. (1 point) Decision trees can have a high variance problem since it can develop deep levels to overfit the dataset.

Solution: True. Decision tree can go very deep and can be sensitive to minor changes in the dataset.

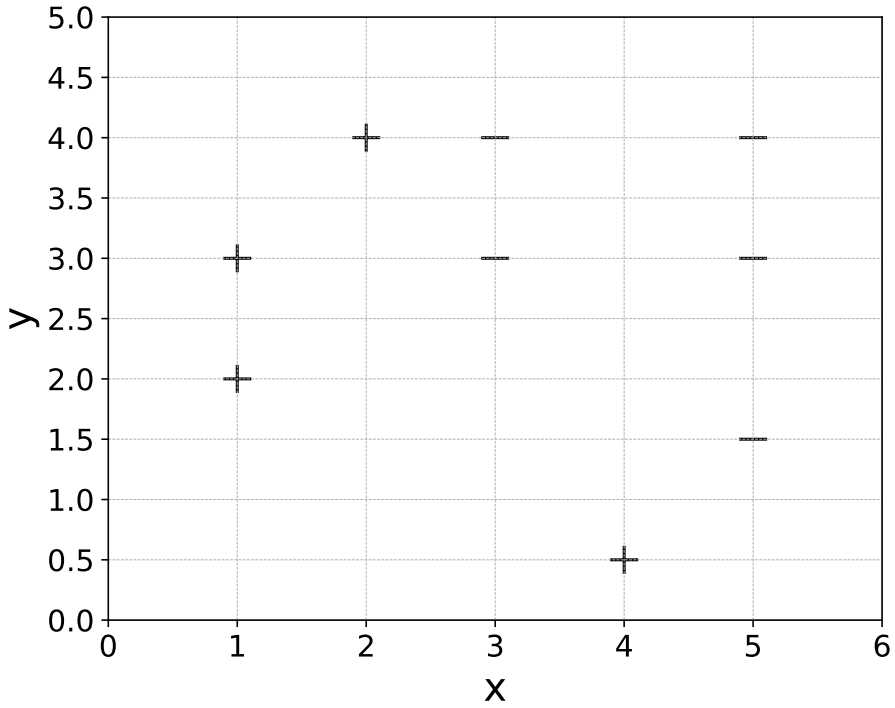
iii. (1 point) Bagging addresses the high variance problem by averaging the predictions.

Solution: True. Averaging predictions lowers the variance.

iv. (1 point) Bootstrapping helps us sample different model weights that are independent from each other.

Solution: False. They are conditionally independent but dependent on the original dataset.

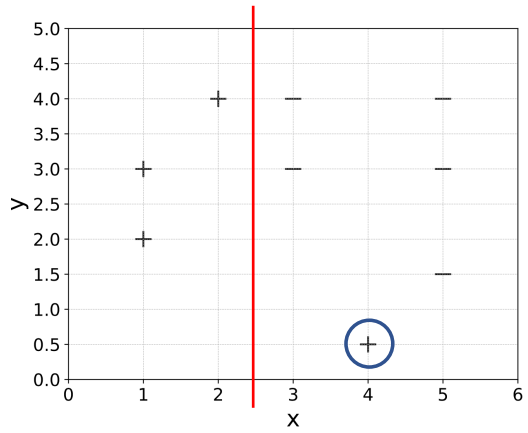
(b) The figure below shows a dataset. Each example in the dataset has two input features x and y and may be classified as a positive example (labelled $+$) or a negative example (labelled $-$). We wish to apply an ensemble of axis-aligned decision stumps to solve the classification problem.



i. (2 points) Draw the decision boundary corresponding to the first decision stump. Lightly shade the side of the boundary corresponding to a positive ($+$) classification.

ii. (1 point) In the previous figure, suppose you decide to apply the AdaBoost algorithm and would like to now add another decision stump. Circle the point(s) which has/have the highest weight after the first stump.

Solution:



5. **Gradient boosting.**

(a) (1 point) Gradient boosting typically uses which type of base learner?

A. Deep neural networks

B. Support vector machines

C. Shallow decision trees

D. k-Nearest Neighbors

(b) (1 point) What is a key characteristic of functional gradient descent?

A. It optimizes a function in a vector space.

B. It optimizes a function in a functional space.

C. It does not converge.

D. It requires a fixed learning rate.

(c) (1 point) What is the main difference between random forests and gradient boosting?

A. Random forests use trees, while gradient boosting uses linear models.

B. Random forests build trees independently, while gradient boosting builds trees sequentially.

C. Random forests are prone to overfitting, while gradient boosting is not.

D. Gradient boosting is a supervised learning algorithm, while random forests are unsupervised.

(d) (2 points) In the m'th round of FSAM with exponential loss, what is the objective function?

Solution:

$$(v_m, h_m) = \arg\min_{v \in \mathbb{R}, h \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \exp\left(-y_i \left(f_{m-1}(x_i) + v h(x_i)\right)\right).$$

6. **Neural networks.**

(a) (1 point) Which of the following is NOT a cause for the difficulty of neural network optimization?

(A) Backpropagation is computationally expensive.

(B) When using the chain rule, the gradient may explode or vanish since they are multiplied together.

(C) The optimization may get stuck in local minima.

(D) The learning rate (i.e. step size) schedule is hard to set.

Solution: A. BP is only a little more expensive than forward pass.

(b) (1 point) Why would you prefer the ReLU activation function over the sigmoid?

Solution: Less gradient vanishing.

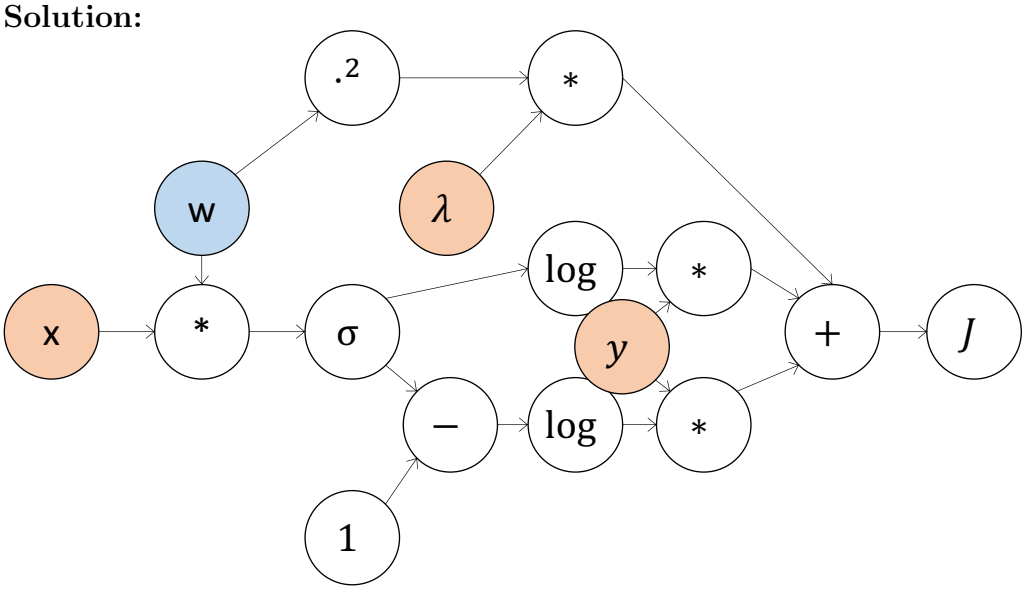
(c) (1 point) Why would you prefer neural networks over linear models?

Solution: Learning non-linear functions using hidden nodes. Learning useful features/representations.

(d) You are working on a dataset with input $\mathbf{x} \in \mathbb{R}^D$, paired with a binary label $y \in [0, 1]$. You would like to build a neural network that performs logistic regression with L2 regularization on the weights $\mathbf{w}$ with coefficient λ. Use the sigmoid activation function. Assume that the data is centered and there is no bias term b. Note that the binary cross entropy loss is $L(\hat{y}) = -y \log \hat{y} - (1 - y) \log(1 - \hat{y})$, where $\hat{y}$ is the predicted probability of the positive class. See next page.

i. (2 points) Draw the computational graph below that takes inputs from $\{\mathbf{x}, y, \lambda, \mathbf{w}\}$ and outputs the total loss J (both the loss and the regularization):

Solution:



ii. (2 points) Use the backpropagation algorithm to derive the gradient of J with respect to the weights $\mathbf{w}$ ($\frac{\partial J}{\partial \mathbf{w}}$). Note that the derivative of the sigmoid function is $\sigma'(z) = \sigma(z)(1 - \sigma(z))$. Let $\hat{y}$ be your model's predicted probability of the positive class, simplify your answer in terms of $\mathbf{x}, y, \hat{y}, \lambda, \mathbf{w}$.

Solution: Let's define $x^\top x = z$, $\sigma(z) = \hat{y}$ and the logistic loss as L, and the regularizer as R.

$$\begin{aligned}
\frac{\partial J}{\partial \mathbf{w}} &= \frac{\partial L}{\partial \mathbf{w}} + \frac{\partial R}{\partial \mathbf{w}} \\
&= \frac{\partial L}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial z} \frac{\partial z}{\partial \mathbf{w}} + \lambda \mathbf{w} \\
&= \left[-y \frac{1}{\hat{y}} \hat{y}(1 - \hat{y}) + (1 - y) \frac{1}{1 - \hat{y}} \hat{y}(1 - \hat{y}) \right] \frac{\partial z}{\partial \mathbf{w}} + \lambda \mathbf{w} \\
&= \left[-y(1 - \hat{y}) + (1 - y)\hat{y} \right] \mathbf{x} + \lambda \mathbf{w} \\
&= \left[-y + y\hat{y} + \hat{y} - y\hat{y} \right] \mathbf{x} + \lambda \mathbf{w} \\
&= \mathbf{x}(\hat{y} - y) + \lambda \mathbf{w}.
\end{aligned}$$

iii. (1 point) Suppose that the sigmoid function is replaced by a hard threshold function $h(x) = \mathbb{1}[x > 0.5]$. Explain in one to two sentences why backpropagation cannot be used.

Solution: The hard threshold function is not differentiable, and it will break the chain rule when performing backpropagation.

7. **K-means and GMM.**

(a) Short answers.

i. (1 point) Under what condition(s), will GMM be equivalent to k-means?

Solution: σ's are equal and approaching to zero, equal mixing coefficients π.

ii. (1 point) Write down a objective function for the k-means algorithm, where the function decreases its value in every iteration of k-means.

Solution: $J(c, \mu) = \sum_{i=1}^{n} \|x_i - \mu_{c_i}\|^2$, or $J(\mu) = \sum_{i=1}^{n} \min_j \|x_i - \mu_j\|^2$.

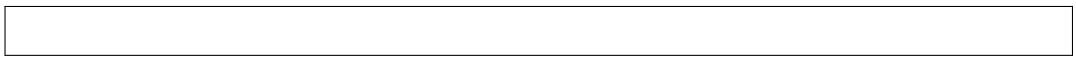
(b) True or false. State your explanation in one sentence.

i. (1 point) The optimization of k-means may get stuck in local minima.

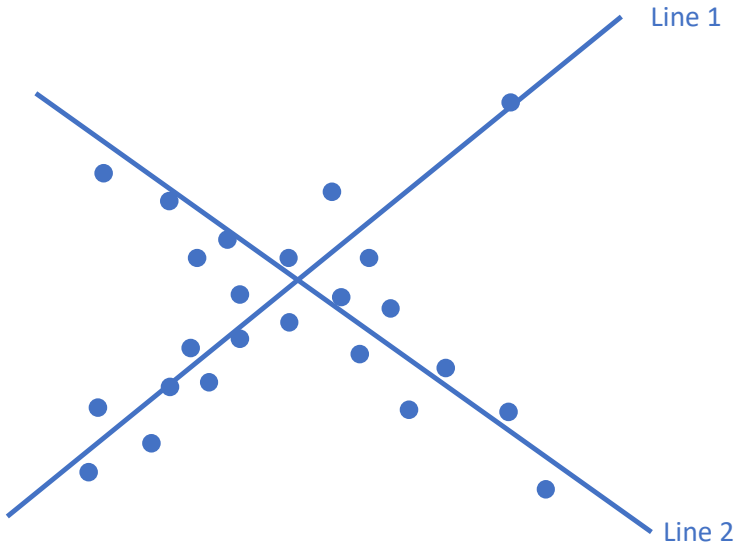
Solution: True. The objective function is non-convex.

ii. (1 point) k-means and GMM can directly give you a prediction on the number of clusters.

Solution: False. You get a better fit with more clusters, so it is hard to determine the number of clusters.



8. **EM.** Suppose we believe the targets form a linear function of the input on some region of input space and a different linear function on another region. We can encode these beliefs using a binary latent variable, and the resulting model allows us to model a mixture of multiple linear relations such as the following figure:



In this model, a latent variable $z \in \{0, 1\}$ is first selected, then the target t is generated as a linear function of $\mathbf{x}$, where the weights depend on which z was chosen. More formally, we use the following probabilistic model:

$$p(z = 1|\boldsymbol{\theta}) = \pi$$

$$p(y|\mathbf{x}, z; \boldsymbol{\theta}) = \begin{cases} \mathcal{N}(y|\mathbf{w}_0^\top \mathbf{x}, \sigma_0^2), & z = 0 \\ \mathcal{N}(y|\mathbf{w}_1^\top \mathbf{x}, \sigma_1^2), & z = 1 \end{cases}$$

The parameters of this model are $\boldsymbol{\theta} = \{\pi, \mathbf{w}_0, \mathbf{w}_1, \sigma_0, \sigma_1\}$. Suppose we observe a dataset $\mathcal{D} = \{(\mathbf{x}^{(n)}, y^{(n)})\}_{n=1}^N$. See next page.

(a) (2 points) Assuming that you know z for each example, and z is part of the dataset, write the log-likelihood for this model, i.e. $\log(p(y, z|\mathbf{x}; \boldsymbol{\theta}))$. Do not replace or substitute $\mathcal{N}$ for the pdf of the normal.

Solution: The log-likelihood here assumes we observe the latent variable $z^{(n)}$ for each datapoint, i.e. we observe the dataset

$$\mathcal{D}_{\text{complete}} = \{(\mathbf{x}^{(n)}, z^{(n)}, y^{(n)})\}_{n=1}^{N}.$$

Thus, the complete data log-likelihood is:

$$\begin{aligned}
&\sum_{n=1}^{N} \log p(y^{(n)}, z^{(n)}|\mathbf{x}^{(n)}) \\
=&\sum_{n=1}^{N} \log p(y^{(n)}|\mathbf{x}^{(n)}, z^{(n)}) p(z^{(n)}) \\
=&\sum_{n=1}^{N} \log \left\{ \pi^{z^{(n)}} (1-\pi)^{1-z^{(n)}} \mathcal{N}(y^{(n)}|\mathbf{w}_1^\top \mathbf{x}^{(n)}, \sigma_1^2)^{z^{(n)}} \mathcal{N}(y^{(n)}|\mathbf{w}_0^\top \mathbf{x}^{(n)}, \sigma_0^2)^{1-z^{(n)}} \right\} \\
=&\sum_{n=1}^{N} z^{(n)} \log \pi + (1-z^{(n)}) \log(1-\pi) + \\
&\qquad z^{(n)} \log \mathcal{N}(y^{(n)}|\mathbf{w}_1^\top \mathbf{x}^{(n)}, \sigma_1^2) + (1-z^{(n)}) \log \mathcal{N}(y^{(n)}|\mathbf{w}_0^\top \mathbf{x}^{(n)}, \sigma_0^2).
\end{aligned}$$

(b) (2 points) Give an expression for the posterior probability $p(z = 1|\mathbf{x}, y; \boldsymbol{\theta})$. Do not replace or substitute $\mathcal{N}$ for the pdf of the normal. (HINT: You will need to use Bayes rule.)

Solution: Using Bayes Rule, we have:

$$p(z = 1|\mathbf{x}, t; \boldsymbol{\theta}) = \frac{p(y, z = 1|\mathbf{x}; \boldsymbol{\theta})}{p(y|\mathbf{x}; \boldsymbol{\theta})}$$
$$= \frac{p(z = 1|\boldsymbol{\theta})p(y|z = 1, \mathbf{x}; \boldsymbol{\theta})}{p(z = 0|\boldsymbol{\theta})p(y|z = 0; x\boldsymbol{\theta}) + p(z = 1|\boldsymbol{\theta})p(y|z = 1, \mathbf{x}; \boldsymbol{\theta})}$$

Substituting in the model definition:

$$p(z = 1|\mathbf{x}, y; \boldsymbol{\theta}) = \frac{\pi\mathcal{N}(y|\mathbf{w}_1^\top \mathbf{x}, \sigma_1^2)}{(1 - \pi)\mathcal{N}(y|\mathbf{w}_0^\top \mathbf{x}, \sigma_0^2) + \pi\mathcal{N}(y|\mathbf{w}_1^\top \mathbf{x}, \sigma_1^2)}$$

(c) (2 points) Now let $p_n = p(z^{(n)} = 1|\mathbf{x}^{(n)}, y^{(n)}; \boldsymbol{\theta}^{\text{old}})$. Use p_n to derive a lower bound (ELBO) for the marginal log likelihood $\log p(y|\mathbf{x}; \boldsymbol{\theta})$. Do not replace or substitute $\mathcal{N}$ for the pdf of the normal. (HINT: $z^{(n)}$ should not appear in the resulting expression.)

Solution:

$$
\begin{aligned}
&\log p(y|\mathbf{x}) \\
&= \sum_{n=1}^{N} \log p(y^{(n)}|\mathbf{x}^{(n)}) \\
&= \sum_{n=1}^{N} \log \sum_{z^{(n)}} p(y^{(n)}, z^{(n)}|\mathbf{x}^{(n)}) \\
&= \sum_{n=1}^{N} \log \left[p_n \frac{p(y^{(n)}, z^{(n)}|\mathbf{x}^{(n)})}{p_n} + (1-p_n) \frac{p(y^{(n)}, z^{(n)}|\mathbf{x}^{(n)})}{1-p_n} \right] \\
&\geq \sum_{n=1}^{N} p_n \log \frac{p(y^{(n)}, z^{(n)}|\mathbf{x}^{(n)})}{p_n} + (1-p_n) \log \frac{p(y^{(n)}, z^{(n)}|\mathbf{x}^{(n)})}{1-p_n} \\
&= \sum_{n=1}^{N} p_n \log p(y^{(n)}, z^{(n)}|\mathbf{x}^{(n)}) - p_n \log p_n + \\
&\quad (1-p_n) \log p(y^{(n)}, z^{(n)}|\mathbf{x}^{(n)}) - (1-p_n) \log(1-p_n) \\
&= \sum_{n=1}^{N} p_n \log \pi + p_n \log \mathcal{N}(y^{(n)}|\mathbf{w}_1^\top \mathbf{x}^{(n)}, \sigma_1^2) - p_n \log p_n + \\
&\quad (1-p_n) \log(1-\pi) + (1-p_n) \log \mathcal{N}(y^{(n)}|\mathbf{w}_0^\top \mathbf{x}^{(n)}, \sigma_0^2) - (1-p_n) \log(1-p_n).
\end{aligned}
$$

Congratulations! You have reached the end of the exam.