Homework 4

Deadline: Friday, Mar. 8, at 11:59pm.

Submission: You need to submit your solutions as a PDF file, `hw4_writeup.pdf`, through MarkUs[1].
Neatness Point: One of the 10 points will be given for neatness. You will receive this point as long as we don't have a hard time reading your solutions or understanding the structure of your code.

Late Submission: 10% of the marks will be deducted for each day late, up to a maximum of 3 days. After that, no submissions will be accepted.

Collaboration. Weekly homeworks are individual work. See the Course Information handout[2] for detailed policies.

1. **[4pts] AlexNet.** For this question, you will first read the following paper:

   > A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems (NIPS), 2012.
   > `http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks`

   This is a highly influential paper (over 35,000 citations on Google Scholar!) because it was one of the first papers to demonstrate impressive performance for a neural network on a modern computer vision benchmark. It generated lots of excitement both in academia and in the tech industry. The architecture presented in this paper widely used today, and is known as "AlexNet", after the first author. Reading this paper will also help you review a lot of the important concepts from this class.

   (a) **[3pts]** They use a conv net architecture which has five convolution layers and three fully connected layers (one of which is the output layer). Your job is to count the number of units, the number of weights, and the number of connections in each layer. I.e., you should complete the following table:

	# Units	# Weights	# Connections
Convolution Layer 1			
Convolution Layer 2			
Convolution Layer 3			
Convolution Layer 4			
Convolution Layer 5			
Fully Connected Layer 1			
Fully Connected Layer 2			
Output Layer			

   You can ignore the pooling layers when doing these calculations, i.e. you don't need to consider the units in the pooling layers or the connections between convolution and pooling layers. You can also ignore the biases. Note that the paper gives you the answers

[1] `https://markus.teach.cs.toronto.edu/csc411-2019-01`
[2] `http://www.cs.toronto.edu/~mren/teach/csc411_19s/syllabus.pdf`

for the numbers of units in the caption to Figure 2. Therefore, we won't mark the column for units, though you would benefit from trying to work it out yourself.

When counting the number of connections, we'll adopt the convention that when the input to a convolution layer is zero-padded, the connections to the dummy zero values count towards the total. (This is the most convenient way to do it, since it means the number of incoming connections is the same for each unit in a given layer.)

(b) **[1pt]** Now suppose you're working at a software company and want to use an architecture similar to AlexNet in a product. Your project manager gives you some additional instructions; for each of the following scenarios, based on your answers to Part 1, suggest a change to the architecture which will help achieve the desired objective. E.g., modify the sizes of one or more layers. (These scenarios are independent.)

i. You want to reduce the memory usage at test time so that the network can be run on a cell phone; this requires reducing the number of parameters for the network.

ii. Your network will need to make very rapid predictions at test time. You want to reduce the number of connections, since there is approximately one add-multiply operation per connection.

2. **[5pts] Gaussian Naïve Bayes.** In this question, you will derive the maximum likelihood estimates for Gaussian Naïve Bayes, which is just like the naïve Bayes model from lecture, except that the features are continuous, and the conditional distribution of each feature given the class is (univariate) Gaussian rather than Bernoulli. Start with the following generative model for a discrete class label $y \in (1, 2, ..., K)$ and a real valued vector of D features $\mathbf{x} = (x_1, x_2, ..., x_D)$:

$$p(y = k) = \alpha_k \tag{1}$$

$$p(\mathbf{x}|y = k, \boldsymbol{\mu}, \boldsymbol{\sigma}) = \left(\prod_{d=1}^{D} 2\pi\sigma_d^2\right)^{-1/2} \exp\left\{-\sum_{d=1}^{D} \frac{1}{2\sigma_d^2}(x_d - \mu_{kd})^2\right\} \tag{2}$$

where α_k is the prior on class k, σ_d^2 are the variances for each feature, which are shared between all classes, and μ_{kd} is the mean of the feature d conditioned on class k. We write $\boldsymbol{\alpha}$ to represent the vector with elements α_k and similarly $\boldsymbol{\sigma}$ is the vector of variances. The matrix of class means is written $\boldsymbol{\mu}$ where the kth row of $\boldsymbol{\mu}$ is the mean for class k.

(a) **[1pt]** Use Bayes' rule to derive an expression for $p(y = k|\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\sigma})$. *Hint: Use the law of total probability to derive an expression for* $p(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\sigma})$.

(b) **[1pt]** Write down an expression for the negative likelihood function (NLL)

$$\ell(\boldsymbol{\theta}; \mathcal{D}) = -\log p(y^{(1)}, \mathbf{x}^{(1)}, y^{(2)}, \mathbf{x}^{(2)}, \cdots, y^{(N)}, \mathbf{x}^{(N)}|\boldsymbol{\theta}) \tag{3}$$

of a particular dataset $\mathcal{D} = \{(y^{(1)}, \mathbf{x}^{(1)}), (y^{(2)}, \mathbf{x}^{(2)}), \cdots, (y^{(N)}, \mathbf{x}^{(N)})\}$ with parameters $\boldsymbol{\theta} = \{\boldsymbol{\alpha}, \boldsymbol{\mu}, \boldsymbol{\sigma}\}$. (Assume the data are i.i.d.) You may find it helpful to use the indicator notation $\mathbb{I}[y^{(n)} = k]$.

(c) **[2pts]** Take partial derivatives of the likelihood with respect to each of the parameters μ_{kd} and with respect to the shared variances σ_d^2. Based on this, find the maximum likelihood estimates for $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$. You may assume that each class appears at least once in the dataset. You may use the notation $N_k = \sum_{n=1}^{N} \mathbb{I}[y^{(n)} = k]$ in your answers.

(d) **[1pt]** Show that the MLE for α_k is given by the following equation:

$$\alpha_k = \frac{1}{N} \sum_{n=1}^{N} \mathbb{I}[y^{(n)} = k] = \frac{N_k}{N} \tag{4}$$

You may assume that each class appears at least once. You will find it helpful to read about Lagrange multipliers[3].

[3] https://en.wikipedia.org/wiki/Lagrange_multiplier