Homework 1 **CSCE 633**
Due: 11.59pm on February 15, 2019

Instructions for homework submission
Please submit on eCampus a **single pdf** file containing your solutions.
a) Please typewrite in Latex the answers to the math problems. If this is not possible, please handwrite your solution *very clearly*, scan it and merge it to the final pdf file. Make sure that your solution is visible after scanning. Non-visible solutions will not be graded: we wouldn't like our TA to have to guess what you are writing :)
b) Please write a brief report for the experimental problems. At the end of the pdf file, please include your code. The code has to be directly converted instead of scanned (i.e. the text in the code must be selectable).
c) Please start early :)

Question 1
1-dimensional linear regression: Assume a 1-dimensional linear regression model $y = w_0 + w_1 x$. The residual sum of squares (RSS) of the training data $\mathcal{D}^{train} = \{(x_1, y_1), \ldots, (x_N, y_N)\}$ can be written as:

$$RSS(w_0, w_1) = \sum_{n=1}^{N} (y_n - w_0 - w_1 x_n)^2$$

We estimate the weights w_0, w_1 by minimizing the above error.

(a) (1 point) Show that minimizing RSS results in the following closed-form expression:

$$w_1^* = \frac{\sum_{n=1}^{N} x_n y_n - N\left(\frac{1}{N}\sum_{n=1}^{N} x_n\right)\left(\frac{1}{N}\sum_{n=1}^{N} y_n\right)}{\sum_{n=1}^{N} x_n^2 - N\left(\frac{1}{N}\sum_{n=1}^{N} x_n\right)^2}$$

$$w_0^* = \left(\frac{1}{N}\sum_{n=1}^{N} y_n\right) - w_1\left(\frac{1}{N}\sum_{n=1}^{N} x_n\right)$$

Tip: Set the partial derivatives $\frac{\vartheta RSS(w_0,w_1)}{\vartheta w_0}$ and $\frac{\vartheta RSS(w_0,w_1)}{\vartheta w_1}$ equal to 0. Then solve a 2×2 system of linear equations with respect to w_0 and w_1.

(b) (1 point) Show that the above expressions for w_0^* and w_1^* are equivalent to the following:

$$w_1^* = \frac{\sum_{n=1}^{N} (x_n - \bar{x})(y_n - \bar{y})}{\sum_{i=1}^{N} (x_n \bar{x})^2}$$

$$w_0^* = \bar{y} - w_1 \bar{x}$$

where $\bar{x} = \frac{1}{N}\sum_{n=1}^{N} x_n$ and $\bar{y} = \frac{1}{N}\sum_{n=1}^{N} y_n$ are the sample means of input features and outcome values, respectively.

(c) (0.5 point) How would you interpret the above expression in terms of the descriptive statistics (e.g. sample mean, variance, co-variance) of populations $\{x_n\}_{n=1}^N$ and $\{y_n\}_{n=1}^N$?

Question 2

Principled method for learning the step size in gradient descent: In class we discussed that when we use gradient descent to minimize target function $J(\mathbf{w})$ with respect to $\mathbf{w}$, the step size $\alpha(k)$ at iteration k is a crucial hyperparameter. We further said that we can experimentally determine $\alpha(k)$ through cross-validation. There is actually a principled way for computing the optimal $\alpha(k)$ in each iteration and we are going to derive the expression for that.

(a) (0.5 point) According to Taylor series expansion, a differentiable function $f(\mathbf{x})$ can be written around $\mathbf{x_0}$ as follows:

$$f(\mathbf{x}) = f(\mathbf{x_0}) + (\nabla f|_{\mathbf{x}=\mathbf{x_0}})^T \cdot (\mathbf{x} - \mathbf{x_0}) + \frac{1}{2}(\mathbf{x} - \mathbf{x_0})^T \cdot \mathbf{H}_f|_{\mathbf{x}=\mathbf{x_0}} \cdot (\mathbf{x} - \mathbf{x_0}) + \dots$$

where ∇f are the gradient vector and $\mathbf{H}_f$ Hessian matrix of f evaluated at $\mathbf{x_0}$.
Let $\mathbf{w}(k)$ be the value of $\mathbf{w}$ at the k^{th} iteration of gradient descent. Show that the second order Taylor expansion of the target function $J(\mathbf{w})$ around $\mathbf{w}(k)$ is the following:

$$J(\mathbf{w}) \approx J(\mathbf{w}(k)) + (\nabla J|_{\mathbf{w}=\mathbf{w}(k)})^T \cdot (\mathbf{w} - \mathbf{w}(k)) + \frac{1}{2}(\mathbf{w} - \mathbf{w}(k))^T \cdot \mathbf{H}_J|_{\mathbf{w}=\mathbf{w}(k)} \cdot (\mathbf{w} - \mathbf{w}(k))$$

where ∇J are the gradient vector and $\mathbf{H}_J$ Hessian matrix of J evaluated at $\mathbf{w}(k)$.

(b) (1 point) Show that the above expression of $J(\mathbf{w})$ evaluated at $\mathbf{w}(k+1)$ (i.e. at the $(k+1)^{th}$ gradient descent iteration) can be written as:

$$\begin{aligned} J(\mathbf{w}(k+1)) \approx & J(\mathbf{w}(k)) - \left\|\nabla J|_{\mathbf{w}=\mathbf{w}(k)}\right\|_2^2 \cdot \alpha(k) + \\ & + \frac{1}{2}\left(\nabla J|_{\mathbf{w}=\mathbf{w}(k)}\right)^T \mathbf{H}_J|_{\mathbf{w}=\mathbf{w}(k)} \left(\nabla J|_{\mathbf{w}=\mathbf{w}(k)}\right) \cdot \alpha^2(k) \end{aligned}$$

Tip: Take into account the gradient descent update rule $\mathbf{w}(k+1) = \mathbf{w}(k) - \alpha(k) \cdot \nabla J|_{\mathbf{w}=\mathbf{w}(k)}$

(c) (1 point) Show that minimizing the above expression with respect to the step size $\alpha(k)$ results in:

$$\alpha(k) = \frac{\left\|\nabla J|_{\mathbf{w}=\mathbf{w}(k)}\right\|_2^2}{\left(\nabla J|_{\mathbf{w}=\mathbf{w}(k)}\right)^T \mathbf{H}_J|_{\mathbf{w}=\mathbf{w}(k)} \left(\nabla J|_{\mathbf{w}=\mathbf{w}(k)}\right)}$$

The above expression gives a closed-form solution of the step size at iteration k (i.e. $a(k)$) that minimizes the target function at the next iteration.

(d) (Bonus) What is the cost of computing $a(k)$ at each iteration k using the above expression?

Question 3

Predicting forest fires: Forest fires are a major environmental issue endangering human lives. This renders their fast detection a key element for controlling them and potentially preventing them. Since it is hard for humans to monitor all forests, we can use automatic tools based on local sensors to do that. Through these sensors we can get information regarding the meteorological conditions, such as temperature, wind, relative humidity (RH), and amount of rain. We can also compute several fire hazard indexes, such as the forest fire weather index (FWI), fine fuel moisture code (FFMC), duff moisture code (DMC), drought code (DC), and initial spread index (ISI). Using these measures, we can predict whether fire is going to occur in the forest, as well as to estimate the amount of burned area. Such data are part of the "Forest Fires Data Set" of the UCI Machine Learning Repository and their description can be found here: `http://archive.ics.uci.edu/ml/datasets/Forest+Fires`.

Inside "Homework 1" folder on Piazza you can find two files including the train and test data (named "train.csv" and "test.csv") for our experiments. The rows of these files refer to the data samples, while the columns denote the features (columns 1-12) and the outcome variable (column 13), as describe bellow:

1. **X**: x-axis spatial coordinate of the forest: 1 to 9
2. **Y**: y-axis spatial coordinate of the forest: 2 to 9
3. **month**: month of the year: 1 to 12 to denote "jan" to "dec"
4. **day**: day of the week: 1 to 7 to denote "mon" to "sun"
5. **FFMC**: FFMC index from the FWI system
6. **DMC**: DMC index from the FWI system
7. **DC**: DC index from the FWI system
8. **ISI**: ISI index from the FWI system
9. **temp**: temperature in Celsius degrees
10. **RH**: relative humidity
11. **wind**: wind speed in km/h
12. **rain**: outside rain in mm/m2
13. **area**: the burned area of the forest (this is the **outcome** variable)

(a) (1 point) Data exploration: Inspect the input features (e.g. you can plot histograms, scatter plots, etc.). Which of the features are continuous and which categorical?

(b) Classification: From data exploration, we can notice that the the outcome value (i.e. the burned area) is zero for many samples, meaning that the corresponding forests are not affected by fire. Therefore we can dichotomize the outcome variable, based on whether its corresponding value is zero or greater than zero. This creates the following two classes:
Class 0: Forests not affected by the fire, i.e. area = 0
Class 1: Forests affected by the fire, i.e. area > 0
After dichotomizing the outcome variable, we can run a classification task to *predict whether or not fire will occur in a certain forest* based on the input features.

(b.i) (1 point) Implement a K-Nearest Neighbor classifier (K-NN) using the euclidean distance as a distance measure to perform the above binary classification task. *Reminder:* Don't forget to normalize the features.

(b.ii) (0.5 point) Explore different values of K through cross-validation on the training set. Plot the classification accuracy, i.e. (#samples correctly classified) / (total #samples), against the different values of K.

(b.iii) (0.5 point) Report the classification accuracy on the test set using the best K from cross-validation.

(b.iv) (Bonus) Instead of using the euclidean distance for all features, experiment with different types of distances or distance combinations, i.e. Hamming distance for categorical features. Report your findings.

(c) Linear Regression: Among the forests that were affected by the fire, we can use linear regression to predict the actual amount of area that was burned. For this task, **we will only use the samples of the train and test set with burned area (column 13) greater than zero, i.e. area > 0**.

(c.i) (1 point) Plot the histogram of the outcome variable. What do you observe? Plot the histogram of the logarithm of the outcome value, i.e. log(area). What do you observe now?

(c.ii) (0.5 point) Implement a linear regression model to fit the outcome data using the ordinary least squares (OLS) solution.

(c.iii) (0.5 point) Test your model on the test data and compute the residual sum of squares error (RSS) and the correlation between the actual and predicted outcome variable.

(c.iv) (Bonus) Experiment with different non-linear functions of the input features. Report your findings on the train and test sets.