CSL 341: Assignment 1

Due Date: 11:50 pm, Sunday Sep 7, 2014. Total Points: 75

Notes:

- This assignment has a mix of theoretical as well as implementation questions.
- Only the implementation questions will be graded.
- You are strongly encouraged to try out theoretical questions though they are not graded.
- You should submit all your code as well as any graphs that you might plot. Do not submit answers to theoretical questions.
- For each problem that you implement, you should include a file <question-no.writeup.txt> which should have a brief explanation of what you did.
- You should use MATLAB for all your programming solutions.
- Your code should have appropriate documentation for readability.
- You will be graded based on what you have submitted as well as your ability to explain your code.
- Refer to the course website for assignment submission instructions.
- This assignment is supposed to be done individually. You should carry out all the implementation by yourself.
- We plan to run Moss on the submissions. Any cheating will result in a penalty of **-7** points on your total course score (in addition to a 0 on the assignment). Stricter penalties (**including a fail grade**) may follow.
- Many of the problems below have been adapted from the Machine Learning course offered by Andrew Ng at Stanford.

1. **(20 points) Linear Regression**
   In this problem, we will implement least squares linear regression for the housing problem as discussed in class. Recall that the error metric for least squares is given by:

   $$J(\theta) = \sum_{i=1}^{m} \frac{1}{2}\left(y^{(i)} - h_\theta(x^{(i)})\right)^2$$

   where $h_\theta(x) = \theta^T x$ and all the symbols are as discussed in the class. The files "q1x.dat" and "q1y.dat" contain the areas of the houses in our dataset ($x^{(i)}$'s, $x^{(i)} \in \mathcal{R}$) and their prices ($y^{(i)}$'s, $y^{(i)} \in \mathcal{R}$), respectively, with one training example per row. We will implement least squares linear regression to learn the relationship between $x^{(i)}$'s and $y^{(i)}$'s.

   (a) (**8 points**) Implement batch gradient descent method for optimizing $J(\theta)$. Choose an appropriate learning rate and the stopping criteria (as a function of the change in the value of $J(\theta)$). You can initialize the parameters as $\theta = \vec{0}$ (the vector of all zeros). Do not forget to include the intercept term. Report your learning rate, stopping criteria and the final set of parameters obtained by your algorithm.

(b) (**3 points**) Plot the data on a two-dimensional graph and plot the hypothesis function learned by your algorithm in the previous part.

(c) (**3 points**) Draw a 3-dimensional mesh showing the error function ($J(\theta)$) on z-axis and the parameters in the $x - y$ plane. Display the error value using the current set of parameters at each iteration of the gradient descent. Include a time gap of 0.2 seconds in your display for each iteration so that the change in the function value can be observed by the human eye.

(d) (**3 points**) Repeat the part above for drawing the contours of the error function at each iteration of the gradient descent. Once again, chose a time gap of 0.2 seconds so that the change be perceived by the human eye.

(e) (**3 points**) Repeat the part above (i.e. draw the contours at each learning iteration) for the step size values of $\eta = \{0.1, 0.5, 0.9, 1.3, 2.1, 2.5\}$. What do you observe? Comment.

2. **(15 points) Logistic Regression**
Consider the log-likelihood function for logistic regression:

$$L(\theta) = \sum_{i=1}^{m} y^{(i)} \log h_\theta(x^{(i)}) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)}))$$

For the following, you will need to calculate the value of the Hessian H of the above function.

(a) **(10 points)** The files "q2x.dat"and "q2y.dat"contain the inputs ($x^{(i)} \in R^2$) and outputs ($y^{(i)} \in \{0, 1\}$) respectively for a binary classification problem, with one training example per row. Implement[1] Newton's method for optimizing $L(\theta)$, and apply it to fit a logistic regression model to the data. Initialize Newton's method with $\theta = \vec{0}$ (the vector of all zeros). What are the coefficients θ resulting from your fit? (Remember to include the intercept term.)

(b) **(5 points)** Plot the training data (your axes should be x_1 and x_2 , corresponding to the two coordinates of the inputs, and you should use a different symbol for each point plotted to indicate whether that example had label 1 or 0). Also plot on the same figure the decision boundary fit by logistic regression. (i.e., this should be a straight line showing the boundary separating the region where $h(x) > 0.5$ from where $h(x) \leq 0.5$.)

3. **(15 points) Locally Weighted Linear Regression**
In this problem, we will generalize the ideas of linear regression to implement locally weighted linear regression where we want to "weigh" different training examples differently. Specifically, suppose we want to minimize

$$J(\theta) = \frac{1}{2} \sum_{i=1}^{m} w^{(i)} (y^{(i)} - \theta^T x^{(i)})^2.$$

In the matrix notation, the error function above can alternately be written as:

$$J(\theta) = \frac{1}{2}(X\theta - Y)^T W (X\theta - Y)$$

for an appropriate diagonal matrix W, and where X and Y are as defined in class. In class, we worked out what happens for the case where all the weights (the $w^{(i)}$'s) are the same. We derived an analytical solution for the parameters θ using what is called the normal equation:

$$X^T X\theta = X^T Y,$$

The value of θ that minimizes $J(\theta)$ is given by $(X^T X)^{-1} X^T Y$. By finding the derivative $\nabla_\theta J(\theta)$ and setting that to zero, we can generalize the normal equation to the weighted setting above. Find the new value of θ that minimizes $J(\theta)$ in closed form as a function of X, W and Y.

The files "q3x.dat"and "q3y.dat"contain the inputs ($x^{(i)}$'s) and outputs ($y^{(i)}$'s), respectively, for a regression problem, with one training example per row.

(a) **(4 points)** Implement (unweighted) linear regression ($y = \theta^T x$) on this dataset (using the normal equations), and plot on the same figure the data and the straight line resulting from your fit. (Remember to include the intercept term.)

[1] Write your own version, and do not call a built-in library function.

(b) **(8 points)** Implement locally weighted linear regression on this dataset (using the weighted normal equations you derived above), and plot on the same figure the data and the curve resulting from your fit. When evaluating $h_\theta(.)$ at a query point x, use weights

$$w^{(i)} = \exp\big(-\frac{(x - x^{(i)})^2}{2\tau^2}\big)$$

with a bandwidth parameter $\tau = 0.8$. (Again, remember to include the intercept term.)

(c) **(3 points)** Repeat (b) four times, with $\tau = 0.1, 0.3, 2$ and 10. Which value of τ do you think works best? Comment on what happens to the fit when τ is too small or too large.

4. **(25 points) Gaussian Discrmimant Analysis**
In this problem, we will implement GDA for separating out salmons from Alaska and Canada. Each salmon is represented by two attributes x_1 and x_2 depicting growth ring diameters in 1) fresh water, 2) marine water, respectively. File "q4x.dat" stores the two attribute values with one entry on each row. File "q4y.dat" contains the target values ($y^{(i)}$'s $\in$ {Alaska, Canada}) on respective rows.

(a) **(6 points)** Implement Gaussian Discriminant Analysis using the closed form equations described in class. Assume that both the classes have the same co-variance matrix i.e. $\Sigma_0 = \Sigma_1 = \Sigma$. Report the values of the means, μ_0 and μ_1, and the co-variance matrix Σ.

(b) **(2 points)** Plot the training data corresponding to the two coordinates of the input features, and you should use a different symbol for each point plotted to indicate whether that example had label Canada or Alaska.

(c) **(3 points)** Describe the equation of the boundary separating the two regions in terms of the parameters μ_0, μ_1 and Σ. Recall that GDA results in a linear separator when the two classes have identical co-variance matrix. Along with the data points plotted in the part above, plot (on the same figure) decision boundary fit by GDA.

(d) **(6 points)** In general, GDA allows each of the target classes to have its own covariance matrix. This results (in general) results in a quadratic boundary separating the two class regions. In this case, the maximum-likelihood estimate of the co-variance matrix Σ_0 can be derived using the equation:

$$\Sigma_0 = \frac{1}{m}\sum_{i=1}^{m} \mathbb{1}\{y^{(i)} = 0\}(x^{(i)} - \mu_{y^{(i)}})(x^{(i)} - \mu_{y^{(i)}})^T \tag{1}$$

where $\mathbb{1}$ is the indicator function. And similarly, for Σ_1. The expressions for the means remain the same as before. Implement GDA for the above problem in this more general setting. Report the values of the parameter estimates i.e. μ_0, μ_1, Σ_0, Σ_1.

(e) **(5 points)** Describe the equation for the quadratic boundary separating the two regions in terms of the parameters μ_0, μ_1 and Σ_0, Σ_1. On the graph plotted earlier displaying the data points and the linear separating boundary, also plot the quadratic boundary obtained in the previous step.

(f) **(3 points)** Carefully analyze the linear as well as the quadratic boundaries obtained. Comment on your observations.

Following problems are for your practice and will not be graded.

1. **Logistic Regression**
Recall the log-likelihood function for logistic regression:

$$L(\theta) = \sum_{i=1}^{m} y^{(i)} \log h(x^{(i)}) + (1 - y^{(i)}) \log(1 - h(x^{(i)}))$$

Show that for any vector z, it holds true that $z^T H z \leq 0$.

[Hint: You might want to start by showing the fact that $\sum_i \sum_j z_i x_i x_j z_j = (x^T z)^2 \geq 0$.]

Remark: This is one of the standard ways of showing that the matrix H is negative semi-definite, written "$H \leq 0$". This implies that l is concave, and has no local maxima other than the global one. If you have some other way of showing $H \leq 0$, you are also welcome to use your method instead of the one above.

2. **Locally Weighted Linear Regression**
Recall that, for locally weighted linear regression, we specified $J(\theta)$ as:

$$J(\theta) = (X\theta - Y)^T W (X\theta - Y)$$

for an appropriate diagonal matrix W, and where X and Y are as defined in class. Suppose we have a training set $\{(x^{(i)}, y^{(i)}); i = 1 \ldots, m\}$ of m independent examples, but in which the $y^{(i)}$'s were observed with differing variances. Specifically, suppose that

$$p(y^{(i)} \mid x^{(i)}; \theta) = \frac{1}{\sqrt{2\pi}\sigma^{(i)}} \exp\big(-\frac{(y^{(i)} - \theta^T x^{(i)})^2}{2(\sigma^{(i)})^2}\big)$$

i.e., $y^{(i)}$ has mean $\theta^T x^{(i)}$ and variance $(\sigma^{(i)})^2$ (where the $\sigma^{(i)}$'s are fixed, known, constants). Show that finding the maximum likelihood estimate of θ reduces to solving a weighted linear regression problem. State clearly what the $w^{(i)}$'s are in terms of the $\sigma^{(i)}$'s.

3. **Poisson Regression**

   (a) Consider the Poisson distribution parameterized by λ:

   $$p(y; \lambda) = \frac{e^{-\lambda}\lambda^y}{y!}$$

   Show that the Poisson distribution is in the exponential family, and clearly state what are $b(y), \eta, T(y)$, and $a(\eta)$.

   (b) Consider performing regression using a GLM model with a Poisson response variable. What is the canonical response function for the family? (You may use the fact that a Poisson random variable with parameter λ has mean λ.)

   (c) For a training set $\{(x^{(i)}, y^{(i)}); i = 1, \ldots, m\}$, let the log-likelihood of an example be $\log p(y^{(i)} \mid x^{(i)}; \theta)$. By taking the derivative of the log-likelihood with respect to θ_j , derive the stochastic gradient ascent rule for learning using a GLM model with Poisson responses y and the canonical response function.

4. **Gaussian Discriminant Analysis**
Suppose we are given a dataset $\{(x^{(i)}, y^{(i)}); i = 1, \ldots, m\}$ consisting of m independent examples, where $x^{(i)} \in R^n$ are n-dimensional vectors, and $y^{(i)} \in \{0, 1\}$. We will model the joint distribution of (x, y) according to:

$$p(y) = \phi^y (1 - \phi)^{1-y}$$

$$p(x \mid y = 0) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\big(-\frac{1}{2}(x - \mu_0)^T \Sigma^{-1} (x - \mu_0)\big)$$
$$p(x \mid y = 1) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\big(-\frac{1}{2}(x - \mu_1)^T \Sigma^{-1} (x - \mu_1)\big)$$

Here, the parameters of our model are ϕ, Σ, μ_0 and μ_1 . (Note that while there're two different mean vectors μ_0 and μ_1 , there's only one covariance matrix Σ.)

   (a) Suppose we have already fit ϕ, Σ, μ_0 and μ_1, and now want to make a prediction at some new query point x. Show that the posterior distribution of the label at x takes the form of a logistic function, and can be written

   $$p(y = 1|x; \phi, \Sigma, \mu_0, \mu_1) = \frac{1}{1 + \exp(-\theta^T x)},$$

   where θ is some appropriate function of ϕ, Σ, μ_0, μ_1. (Note: To get your answer into the form above, for this part of the problem only, you may have to redefine the $x^{(i)}$'s to be $n + 1$-dimensional vectors by adding the extra coordinate $x_0^{(i)} = 1$, like we did in class.)

   (b) For this part of the problem only, you may assume n (the dimension of x) is 1, so that $\Sigma = [\sigma^2]$ is just a real number, and likewise the determinant of Σ is given by $|\Sigma| = \sigma^2$. Given the dataset, we claim that the maximum likelihood estimates of the parameters are given by

   $$\phi = \frac{1}{m}\sum_{i=1}^{m} 1\{y^{(i)} = 1\}$$

$$\mu_0 = \frac{\sum_{i=1}^m 1\{y^{(i)} = 0\} x^{(i)}}{\sum_{i=1}^m 1\{y^{(i)} = 0\}}$$

$$\mu_0 = \frac{\sum_{i=1}^m 1\{y^{(i)} = 1\} x^{(i)}}{\sum_{i=1}^m 1\{y^{(i)} = 1\}}$$

$$\Sigma = \frac{1}{m} \sum_{i=1}^m (x^{(i)} - \mu_{y^{(i)}})(x^{(i)} - \mu_{y^{(i)}})^T$$

The log-likelihood of the data is

$$\begin{aligned} L(\phi, \mu_0, \mu_1, \Sigma) &= \log \prod_{i=1}^m p(x^{(i)}, y^{(i)}; \phi, \mu_0, \mu_1, \Sigma) \\ &= \log \prod_{i=1}^m p(x^{(i)} | y^{(i)}; \mu_0, \mu_1, \Sigma) p(y^{(i)}; \phi) \end{aligned}$$

By maximizing l with respect to the four parameters, prove that the maximum likelihood estimates of ϕ, μ_0, μ_1 and Σ are indeed as given in the formulas above. (You may assume that there is at least one positive and one negative example, so that the denominators in the definitions of μ_0 and μ_1 above are non-zero.)