Homework Assignment # 2

Due: Wednesday, October 14, 2015, 11:59 p.m.

Total marks: 110

Question 1. [15 MARKS]

Suppose that the number of accidents occurring daily in a certain plant has a Poisson distribution with an unknown mean λ. Based on previous experience in similar industrial plants, suppose that our initial feelings about the possible value of λ can be expressed by an exponential distribution with parameter $\theta = \frac{1}{2}$. That is, the prior density is

$$f(\lambda) = \theta \mathrm{e}^{-\theta\lambda}$$

where $\lambda \in (0, \infty)$. Assume there are 79 accidents over the next 9 days.

(a) [5 MARKS] Determine the maximum likelihood estimate of λ.

(b) [5 MARKS] Determine the maximum a posteriori estimate of λ.

(c) [5 MARKS] Look at the plots of some exponential distributions to better understand the prior chosen on λ. Imagine that now new safety measures have been put in place and you believe that the number of accidents per day should sharply decrease. For example, maybe you now believe that 4 accidents per day would be a pretty high estimate. How might you change θ to better reflect this new belief about the number of accidents?

Question 2. [25 MARKS]

Let $X_1, \ldots, X_n$ be i.i.d. Gaussian random variables, each having an unknown mean θ and known variance σ_0^2.

(a) [5 MARKS] Assume θ is itself selected from a normal distribution $\mathcal{N}(\mu, \sigma^2)$ having a known mean μ and a known variance σ^2. What is the maximum a posteriori (MAP) estimate of θ?

(b) [10 MARKS] Assume θ is itself selected from a Laplace distribution $\mathcal{L}(\mu, b)$ having a known mean (location) μ and a known scale (diversity) b. Recall that the pdf for a Laplace distribution is

$$p(x) = \frac{1}{2b} \exp\left(\frac{-|x-\mu|}{b}\right)$$

For simplicity, assume $\mu = 0$. What is the maximum a posteriori estimate of θ? If you cannot find a closed form solution, explain how you would use an iterative approach to obtain the solution.

(c) [10 MARKS] Now assume that we have **multivariate** i.i.d. Gaussian random variables, $\mathbf{X}_1, \ldots, \mathbf{X}_n$ with each $\mathbf{X}_i \sim \mathcal{N}(\boldsymbol{\theta}, \boldsymbol{\Sigma}_0)$ for some unknown mean $\boldsymbol{\theta} \in \mathbb{R}^d$ and known $\boldsymbol{\Sigma}_0 = \mathbf{I} \in \mathbb{R}^{d\times d}$, where $\mathbf{I}$ is the identity matrix. Assume $\boldsymbol{\theta} \in \mathbb{R}^d$ is selected from a zero-mean multivariate Gaussian $\mathcal{N}(\boldsymbol{\mu} = \mathbf{0}, \boldsymbol{\Sigma} = \sigma^2\mathbf{I})$ and a known variance parameter σ^2 on the diagonal. What is the MAP estimate of $\boldsymbol{\theta}$?

Question 3. [10 MARKS]

Program the two iterative algorithms, gradient descent and Newton's method, to find the minimum of a function of two-dimensional variable $\mathbf{x} = (x_1, x_2)$

$$f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$$

Set the step length to $\eta = 1$ and try two different starting points: $x^{(0)} = (1.2, 1.2)$ and a more difficult $x^{(0)} = (-1.2, 1)$. Reduce the step size to some $\eta < 1$ and repeat the minimization. Show all your work and discuss the optimization processes you tested.

Question 4. [60 MARKS]

In this question, you will implement linear regression and Poisson regression. An initial script in python has been given to you, called `script_classify.py`, and associated python files. You will be running on a blog dataset, with 230 features and 60,000 samples. Note that the first 50 features are constants for each sample, giving information about the features, and so are removed when the data is loaded. Baseline algorithms, including mean and random predictions, are used to serve as sanity checks. We should be able to outperform random predictions, and the mean value of the target in the training set.

We will be examining some of the practical aspects of implementing regression. As a suggestion, you should start or complete Question 3 before doing this question.

(a) [5 MARKS] The main linear regression class is `FSLinearRegression`. The FS stands for FeatureSelect. The provided implementation has subselected features and then simply explicitly solved for $\mathbf{w} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$. Increase the number of selected features (including all the way to including all the features). What do you find? How can this be remedied?

(b) [10 MARKS] Now implement Ridge Regression, where a ridge regularizer $\lambda \|\mathbf{w}\|_2^2$ is added to the optimization. Run this algorithm on all the features. How does the result differ from (a)? Discuss results for different regularization parameter λ values. Modify the code to report error averaged over multiple splits of the data (at least 10 splits).

(c) [10 MARKS] Imagine that the dataset size continues to grow, which causes the matrix $\boldsymbol{X} \in \mathbb{R}^{n \times d}$ for n samples and d features to become quite large. One option is to go back to subselecting features. In `FSLinearRegression`, add an approach to select 10 features and explain your choice. How does accuracy change compared to the original `FSLinearRegression` and `RidgeRegression`? What happens to the accuracy of the solution if you add ridge regularization with this subset of features?

(d) [15 MARKS] Now imagine that your dataset has gotten even larger, to the point where dropping features is not enough. Instead of removing features, implement a stochastic optimization approach to obtaining the linear regression solution (see Section 4.5.3). Explain your implementation choices.

(e) [20 MARKS] Next you notice that the target is always positive, with many zeros and small numbers and a few large values. These target values look a little like they could come from a Poisson distribution. You recall that generalized linear models allow non-linear transfers on the linear solution, and that conveniently the Poisson distribution happens to have a nice link function. Implement Poisson regression on this data. Hint: using an exponential transfer can cause numerical instabilities. For training, you might consider scaling down the target so that there are not such large values. Moreover, the approach can be somewhat sensitive to features being collinear. You could consider subselecting some number of features once again. Explain the choices you use. You should be able to obtain better performance than linear regression.

Homework policies:

Your assignment must be typed; for example, in Latex, Microsoft Word, Lyx, etc. Images may be scanned and inserted into the document if it is too complicated to draw them properly. Submit a single pdf document or, if you are attaching your code, submit your code together with the typed (single) document as one .zip file.

All code (if applicable) should be turned in when you submit your assignment. Use Matlab, Python, R, Java or C.

Policy for late submission assignments: Unless there are legitimate circumstances, late assignments will be accepted up to 5 days after the due date and graded using the following rule:

on time: your score 1

1 day late: your score 0.9

2 days late: your score 0.7

3 days late: your score 0.5

4 days late: your score 0.3

5 days late: your score 0.1

For example, this means that if you submit 3 days late and get 80 points for your answers, your total number of points will be $80 \times 0.5 = 40$ points.

All assignments are individual, except when collaboration is explicitly allowed. All the sources used for problem solution must be acknowledged, e.g. web sites, books, research papers, personal communication with people, etc. Academic honesty is taken seriously; for detailed information see Indiana University Code of Student Rights, Responsibilities, and Conduct.

Good luck!