Name: Nicolas Oulianitski Essipova
Course: ID2223 HT19
Review Question 1

1. (a) True. Normal equation has a closed-form solution, wherein a mathematical equation gives us the results directly.

   (b) True. The equation has a computational complexity of $O(n^{2.4})$ to $O(n^3)$. Meaning that if you double the number of features, then you'd multiply the computation time by 5.3 to 8 times.

   (c) True. If we wish to have an iterative approach, then we'd turn to Gradient Descent.

2. Calculating squared error of the prediction:
   $(-0.2)^2 + (0.4)^2 + (-0.8)^2 + (1.3)^2 + (-0.7)^2 =$
   $0.04 + 0.16 + 0.68 + 1.69 + 0.49 = 3.06$

   And if we are also interested in the mean squared error of the prediction, then we simply divide by m (m = 5, number of predictions) and get 0.612.

3. (a) Generally **true**.
   (b) Generally **false**.
   (c) Generally **false** - but depends on the task.
   (d) Generally **true** - but depends on the task. It can still be easy to overfit the data.

   Ultimate depends on what we mean by "more" observations. A little bit more? A lot more? What about how many observations relative to features? What is the predictive task? What is the distribution skewness of the data? Et cetera.

4. In a simple linear regression with one independent variable; we have two coefficients – one of which is the weight (a) and the other is the bias (b).

   $y = aX + b$

5. Cross validation (also known as k-fold cross validation, rotation estimation, or out-of-sample testing), is a technique used to assess the predictive performance of a model (in regards to how it will generalize to independent data set) and to alleviate the effects that give rise to overfitting in a model – particularly in cases where data may be limited.

   In k-fold cross validation, you make a k-amount of partitions of the data, run the model on each partition (also known as folds), and then take the average overall error estimate of all the models.

6. Softmax function with two classes (k = 2) is indeed equivalent to the sigmoid function, as shown below. First, let us put forth the two equations for logistic- and softmax functions.

   Logistic Equation:

$$f(x) = \sigma(x) = \frac{1}{1 + e^{-x}}$$

   Softmax Equation:

$$f_i(\vec{x}) = \frac{e^{x_i}}{\sum_{j=1}^{J} e^{x_j}}$$

   Now we present the predicted probabilities using the sigmoid function.

$$\Pr(Y_i = 0) = \frac{e^{-\beta \cdot \mathbf{X}_i}}{1 + e^{-\beta_0 \cdot \mathbf{X}_i}}$$

$$\Pr(Y_i = 1) = 1 - \Pr(Y_i = 0) = \frac{1}{1 + e^{-\beta \cdot \mathbf{X}_i}}$$

   And finally the predicted probabilities using the softmax function.

$$\Pr(Y_i = 0) = \frac{e^{\beta_0 \cdot \mathbf{X}_i}}{\sum_{0 \le c \le K} e^{\beta_c \cdot \mathbf{X}_i}} = \frac{e^{\beta_0 \cdot \mathbf{X}_i}}{e^{\beta_0 \cdot \mathbf{X}_i} + e^{\beta_1 \cdot \mathbf{X}_i}} = \frac{e^{(\beta_0 - \beta_1) \cdot \mathbf{X}_i}}{e^{(\beta_0 - \beta_1) \cdot \mathbf{X}_i} + 1} = \frac{e^{-\beta \cdot \mathbf{X}_i}}{1 + e^{-\beta \cdot \mathbf{X}_i}}$$

$$\Pr(Y_i = 1) = \frac{e^{\beta_1 \cdot \mathbf{X}_i}}{\sum_{0 \le c \le K} e^{\beta_c \cdot \mathbf{X}_i}} = \frac{e^{\beta_1 \cdot \mathbf{X}_i}}{e^{\beta_0 \cdot \mathbf{X}_i} + e^{\beta_1 \cdot \mathbf{X}_i}} = \frac{1}{e^{(\beta_0 - \beta_1) \cdot \mathbf{X}_i} + 1} = \frac{1}{1 + e^{-\beta \cdot \mathbf{X}_i}}$$

   And as we can see; the probabilities end up being equivalent.

7. The reason why -log(t) is used is because it grows very large when the value of t approaches zero, therefore the cost will be very large when the model estimates incorrect probabilities.

   More precisely; the cost will be large if the model estimates a probability close to 0 for a positive instance, and it will also be very large if the model estimates a probability close to 1 for a negative instance – and vice versa.

8. They are related through maximum likelihood estimation. More specifically the maximum likelihood principle, which seeks to minimize the negative log-likelihood – which cross entropy used in logistic regression is derived from.

9. The ROC curve (receiver operating characteristic curve) is a plot that illustrates the diagnostic predictive performance of a model classifier system for binary classification. More specifically, the true positive rate (TPR) is plotted against the false positive rate (FPR). A model with perfect discrimination would have its curve pass through the upper left corner – whereas a purely random classifying model would be a straight line from bottom left to upper right (assuming 50% probability).

   More specifically, the equation of true positive rate (also known as sensitivity) is given by:
   True Positive / (True Positive + False Negative) = TPR

   and the equation of false positive would require us first to know the equation for true negative rate, also known as the specificity:
   True Negative / (True Negative + False Positive) = TNR

   And we then get the equation for the false positive rate by:
   1 – TNR = 1 – (True Negative / (True Negative + False Positive)) = FPR

   And then we plot TPR against FPR, and we get our ROC curve.