COGS 118A Spring 2017 - Assignment #3
Due: May, 1, 2017, 11:59pm
Grade: ____ out of 100 points
Instructions:
Please answer the questions below and insert your answers to create a pdf file (you can have the hand-written version and scan it to create a pdf file); submit your file to TED (ted.ucsd.edu) by 11:59pm, 05/01/2016. You may search information online but you have to use your own words to answer the questions.
Late policy: 5% of the total points will be deducted for the first late day, then every 10% of the total points will be deducted for every extra day past due.

1 (20 points) Entropy and Mutual Information

Given two random variables X and Y, and their joint probability distributions $P(X,Y)$ as shown below(please use ln=log_e for your computation):

	$X=1$	$X=2$	$X=3$	$X=4$
$Y=1$	0.15	0.03	0.05	0.07
$Y=2$	0.02	0.05	0.03	0.05
$Y=3$	0.03	0.20	0.02	0.10
$Y=4$	0.05	0.02	0.10	0.03

(1) Compute the entropy of X, subject to $P(X)$,that is, compute $H(X)$.

(2) Compute the entropy of Y, subject to $P(Y)$,that is, compute $H(Y)$.

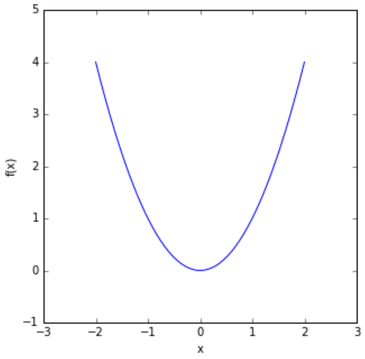
(3) Compute the entropy of X, subject to $P(X|Y)$, that is, compute $H(X|Y)$

(4) Compute the entropy of Y, subject to $P(Y|X)$, that is, compute $H(Y|X)$(Hint:Do you really need to compute from $P(Y|X)$?)

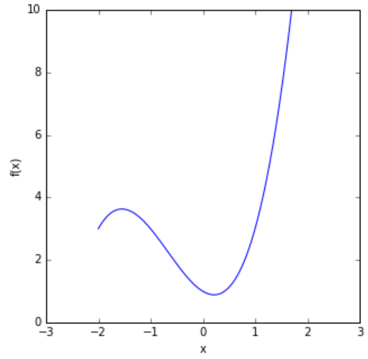
(5) Compute the mutual information of X and Y, subject to $P(X,Y)$, that is, compute $I(X;Y)$

2 (12 points) Convex

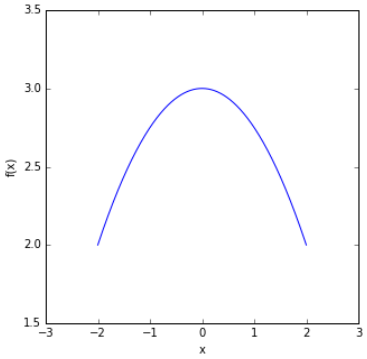
Please identify the convexity for the following six functions (a-f). Simply write down whether the function is convex or non-convex.



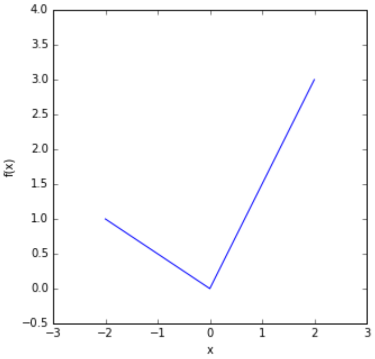
(a)



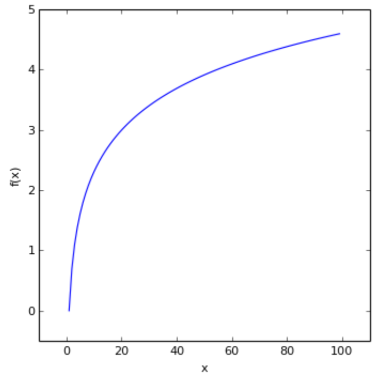
(b)



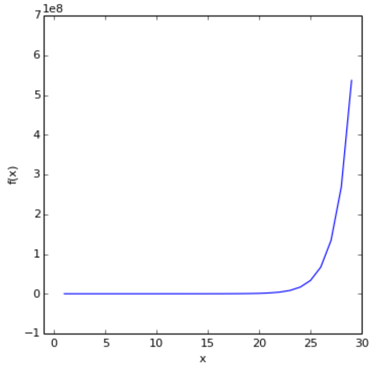
(c)



(d)



(e)



(f)

3 (10 points) Normal Distribution

Normal distribution experiment. Please do the following procedures, paste your source code and show the results in your report:

(1) Use **randn** function in MATLAB or Python Numpy library to generate 1000 1-D samples from normal distribution and name it as variable X.

(2) Use **hist** in MATLAB or **pyplot.hist** function in Python to compute the histogram of the samples and paste it to the report.

(3) Generate another 1000 1-D samples again and name it as variable Y.

(4) Plot the scattered points (X, Y) in a 2-D figure using **scatter** function and paste the figure in the report.

(5) Now compute two new variables $X' = 3X + Y$ and $Y' = X - 2Y$ and plot the scattered points (X', Y') in a 2-D figure and paste it in the report.

(6) Compute the histogram again for X' and Y' and paste the figures in the report.

(7) Explain what you see for this experiment from histograms and scattered points.

4 (8 points) Decision Boundary

Given a classifier that performs classification in $\mathbb{R}^2$(data points with 2 features (x_1, x_2)) with the following classification rule.

$$h(x_1, x_2) = \begin{cases} 1, & \textit{if} \quad 4x_1 + 10x_2 + 3 \geq 0 \\ 0, & \textit{otherwise.} \end{cases}$$

Please draw the decision boundary of this classifier and shade the area that the classifier predicts 1. Make sure you have marked the x_1 and x_2 axises and the intercept points on those axises.

5 (25 points) Least Square Estimation Via Gradient Descent

In homework2, you have implemented Least Square Estimation using the close form solution. In this homework, you will be implementing Least Square Estimation using gradient descent. In the lecture, you have derived that the gradient of L2-norm as the loss function is:

$$\frac{dg(W)}{dW} = 2X^T XW - 2X^T Y,$$

where

$$g(W) = (X \cdot W - Y)^T (X \cdot W - Y).$$

Then, by applying the update rule

$$W_{t+1} = Wt - \lambda * gradient,$$

where λ is the learning rate, you can get a similar result as you had in homework2. The algorithm works as the following:

1. Load data.mat(from homework2) and process X following homework2
2. Initialize the parameter $W = \mathbf{0}$ (zero vector)
3. Compute the gradient $\frac{dg(W)}{dW}$
4. Apply the update rule(set $\lambda \leq 0.0001$)
5. Repeat step 3 and 4 until convergence (e.g. $(||W_{t+1} - W_t||_{L1}) < 0.001$)

Plot your computed curve over the data. Print and compare our your W computed via gradient decent and close form solution, does your W computed via gradient descent make sense? Paste your figures and code in the report.

Hint:

1. What is the dimension of $\frac{dg(W)}{dW}$?
2. What is the dimension of W

6 (25 points) L1 Distance

Similar to the above, but now use L1 norm as your loss function:

$$\text{error} = \sum_{i=1}^{N} |y_i - f(x_i)|$$

where (x_i, y_i) are the given data points, and f is the function by design. Plot your computed curve over the data. Print and compare our your W to the Ws computed via gradient decent using L2-norm and close form solution, does your W computed make sense? Paste your figures and code in the report.

Hint:

$$\frac{d|x|}{dx} = \begin{cases} 1, & x > 0 \\ 0, & x = 0 \\ -1, & x < 0. \end{cases}, \frac{d|y - f(x)|}{dx} = \begin{cases} \frac{-df(x)}{dx}, & y - f(x) > 0 \\ 0, & y - f(x) = 0 \\ \frac{df(x)}{dx}, & y - f(x) < 0. \end{cases}$$