CS416 – HW1

Introduction

Due Friday 9/10 at 11:59pm. Please complete all exercises and problems below.

- All the files can be found in

  `http://www.cs.wm.edu/~liqun/teaching/cs416_20f/hw1/`

- You can also copy to your directory on a department machine by:

  `cp ~liqun/public_html/teaching/cs416_20f/hw1/* .`

Your submission consists of three steps:

1. Create hw1.pdf with your solutions to the following problems. The solutions can be typed or written and scanned but the resulting pdf must be high quality and easily readable. Put your name and your login id in the file.
2. You'll need to create or edit these files in the directory hw1. Complete the requested code in these files.
   - exercise_1.ipynb
   - gd.py
   - gradient-descent.ipynb
   - multivariate-linear-regression.ipynb
3. Submit:

   ```
   ~wye/bin/submit CS416 HW1 hw1.pdf exercise_1.ipynb gd.py
   gradient-decent.ipynb multivariate-linear-regression.ipynb
   ```

Problem 1: Partial Derivatives

Consider the following functions of the variables u, v, and w. Assume the variables x, y, $x^{(i)}$ and $y^{(i)}$ are **constants**: they represent numbers that will not change during the execution of a machine learning algorithm (e.g., the training data). (2 points for each problem)

$$f(u,v) = 8u^2v^4 + 4v^3 + 6u$$

$$g(u,v,w) = x\log(u) + yuvw^3 + 13x^3$$

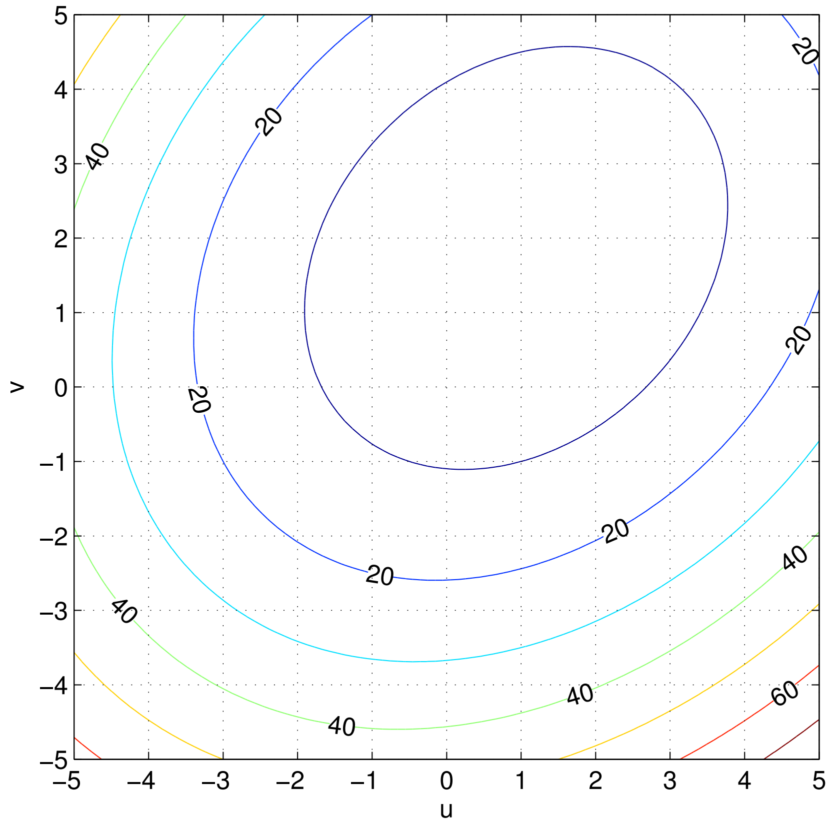
$$h(u,v) = \sum_{i=1}^{m} \frac{1}{2}(x^{(i)}u + y^{(i)}v)^2$$

Write the following partial derivatives:

1. $\frac{\partial}{\partial u} f(u,v)$
2. $\frac{\partial}{\partial v} f(u,v)$
3. $\frac{\partial}{\partial u} g(u,v,w)$
4. $\frac{\partial}{\partial v} g(u,v,w)$
5. $\frac{\partial}{\partial w} g(u,v,w)$
6. $\frac{\partial}{\partial u} h(u,v)$
7. $\frac{\partial}{\partial v} h(u,v)$

Problem 2: Partial Derivative Intuition

Consider the following contour plot of a function $f(u,v)$:



For each of the following partial derivatives, state whether it is positive, negative, or equal to zero. Briefly explain. These questions can be answered from the contour plot without knowing the formula for the function. (2 points for each problem)

(**Note**: for two numbers a and b we will use the notation $\frac{\partial}{\partial u} f(a, b)$ to mean "the partial derivative of $f(u, v)$ with respect to u at the point where $u = a$ and $v = b$". This notation is succinct but obfuscates the original variable names. A more explicit way to write the same thing is $\frac{\partial}{\partial u} f(a, b)|_{u=a,v=b}$)

1. $\frac{\partial}{\partial u} f(-2, -2)$
2. $\frac{\partial}{\partial v} f(-2, -2)$
3. $\frac{\partial}{\partial u} f(3, -3)$
4. $\frac{\partial}{\partial v} f(3, -3)$
5. To the nearest integer, estimate the values of u and v that minimize $f(u, v)$.

Problem 3: Matrix Manipulation I

Matrix multiplication practice (2 points each for the first 5 problems and 5 points for the last problem).

1. Write the result of the following matrix-matrix multiplication. Your answer should be written in terms of u, v, a and b.

$$\begin{bmatrix} 3 & -1 \\ 2 & 5 \\ -2 & 2 \end{bmatrix} \cdot \begin{bmatrix} u & a \\ v & b \end{bmatrix}$$

2. Suppose $A \in \mathbb{R}^{2\times 2}$, $B \in \mathbb{R}^{2\times 4}$. Does the product AB exist? If so, what size is it?
3. Suppose $A \in \mathbb{R}^{3\times 5}$, $B \in \mathbb{R}^{4\times 1}$. Does the product AB exist? If so, what size is it?
4. Suppose $A \in \mathbb{R}^{3\times 2}$, $y \in \mathbb{R}^3$. Is $y^T A$ a row vector or a column vector?
5. Suppose $A \in \mathbb{R}^{3\times 2}$, $x \in \mathbb{R}^2$. Is Ax a row vector or a column vector?
6. Suppose $(Bx + y)^T A^T = 0$, where A and B are both invertible $n \times n$ matrices, x and y are vectors in $\mathbb{R}^n$, and 0 is a vector of all zeros. Use the properties of multiplication, transpose, and inverse to show that $x = -B^{-1}y$. Show your work.

Problem 4: Matrix Manipulation II

(10 points) Create a jupyter notebook called **exercise_1.ipynb** and write code to do the following.

1. Enter the following matrices and vectors

$$A = \begin{bmatrix} -2 & -3 \\ 1 & 0 \end{bmatrix}, B = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}, x = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

2. Compute $C = A^{-1}$
3. Check that $AC = I$ and $CA = I$
4. Compute Ax
5. Compute $A^T A$
6. Compute $Ax - Bx$
7. Compute $||x||$ (use the dot product)

8. Compute $||Ax - Bx||$
9. Print the first column of A (do not use a loop – use array "sliding" instead)
10. Assign the vector x to the first column of B (do not use a loop – use "array slicing" instead)
11. Compute the element-wise product between the first column of A and the second column of A

Problem 5: Linear Regression

The problems consider linear regression based on the following hypothesis and function. (10 points each)

$$h_\theta(x) = \theta_0 + \theta_1 x, J(\theta_0, \theta_1) = \frac{1}{2}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2$$

Consider the following small data set:

x	y
2	5
-1	-1
1	3

1. Solve for the values of θ_0 and θ_1 that minimize the cost function by substituting the value from the training set into the cost function, setting the derivatives (with respect to both θ_0 and θ_1) equal to zero, and solving the system of two equations for θ_0 and θ_1. Show your work.
2. Now do the same thing, but do not substitute the values of the training set into the cost function. Instead. leave the $x^{(i)}$ and $y^{(i)}$ variables, take the derivatives with respect to both θ_0 and θ_1, set them equal to zero, and solve for θ_0 and θ_1. This will give you a general expression for θ_0 and θ_1 in terms of the training data.

   Check your answer by plugging in the training data from the previous problem into your expression for θ_0 and θ_1. You should get the same values for θ_0 and θ_1 that you got in that problem.
3. In this problem you will implement gradient descent for linear regression. Open the notebook **gradient-descent.ipynb** in Jupyter and follow the instructions to complete the problem.

Problem 6: Polynomial Regression

(30 points) In this problem you will implement methods for multivariate linear regression and use them to solve a polynomial regression problem. The purpose of this problem is:

1. To practice writing "vectorized" versions of algorithms in Python
2. To understand how feature expansion can be used to fit non-linear hypotheses using linear methods
3. To understand feature normalization and its impact on numerical optimization for machine learning. Open the notebook **multivariate-linear-regression.ipynb** and follow the instructions to complete the problem.