**CSCE 623 Spring 2019 - Machine Learning In Class Work, Day 3 (2 Apr 2019)**

From Chapter 1: Linear algebra review – **Do this before the lecture**

1. What is the difference between a scalar, a row vector, a column vector and a matrix?

Scalar is a constant. A row vector is a single of row of k columns. A column vector is k rows of a single column. A matrix is an m x n collection of values. Can be interpreted as m column vectors of size n or n row vectors of size m.

1. Suppose you wanted to examine the *i*th **row** of a *m* × *n* matrix called *X*. What kind of linear algebra structure would the entity be? How would the indices be numbered? Write the resulting structure below.

The ith row is a row vector

1. Which of the following are valid multiplications operations? If the operation is valid, give the dimensions of the resulting product. Assume that *W* is a *m* × *m* two-dimensional matrix, X is a *m* × *n* two-dimensional matrix and y is a m × 1 one-dimensional vector

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Valid? | Resulting dimensions if valid | Operation | Valid? | Resulting dimensions if valid |
| *W* × *X* | *y* | *M x n* | *X* × *y* | n |  |
| *X* × *W* | *n* |  | *X* × *yT* | n |  |
| *XT* × *W* | *y* | *N x m* | *XT* × *y* | y | N x 1 |
| *XT* × *WT* | *y* | *N x m* | *XT* × *yT* | n |  |
| *WT* × *X* | *y* | *M x n* | *y* × *y* | n |  |
| *y* × *X* | *n* |  | *yT* × *y* | y | scalar |
| *yT* × *X* | *y* | *1 x n* | *y* × *yT* | y | M x m |

1. Given the following matrix, write the multiplication formula for each cells in the resulting matrix



Simple Linear Regression as Matrix Algebra, part 1: **Do this when instructed**

Below is a simple dataset with 5 students scores on an aptitude test (X1), and their resulting scores on a course final exam (*y*). Your goal is to build a model which predicts *y* based on (X1). You will pick values for 2 coefficients (*β*0 and *β* 1) to try to minimize the prediction errors and draw the best line through the data. Note that *β0* is the y-intercept and *β*1 is the effect of the aptitude test (X1) on the slope of the line.

|  |  |  |
| --- | --- | --- |
| Student (i) | X1 | y |
| 1 | 95 | 85 |
| 2 | 85 | 95 |
| 3 | 80 | 70 |
| 4 | 70 | 65 |
| 5 | 60 | 70 |

Step 1: In the following equation, you will notice that if you expand the equation to list every row i = 1…5, then the result of the equation *Yhat* could be represented as a matrix multiplication on the *β* and a *design* matrix of 1’s and the X1 values.



Using the design matrix X shown below, write the matrix equation for computing   
Y (which is a vector of length 5), from *β* (which is a vector of length 2), and **X** (which is a 5 x 2 matrix)

|  |  |
| --- | --- |
| 1 | 95 |
| 1 | 85 |
| 1 | 80 |
| 1 | 70 |
| 1 | 60 |

CODING – PYTHON (to be coded in the instructor-provided ‘shell’ STUDENT CODE locations):

Step 2: in python, implement a matrix for X (X), a matrix for *β* (beta), and the code required to perform the matrix multiplication which produces the vector *Yhat* (yhat) Note that you will need to pick the initial values for the two values for beta. These two beta values are the y-intercept and slope of the line (remember your grade school math?)

Step 3: in python, compare the vector *Yhat* to the true final exam scores (y) in the dataset. Your code should produce an error vector (ydiff) – a vector of the errors in each of the 5 predictions.

Step 4: write code to compute the value of three *loss functions*

RSS (rss): the sum of the residual squared error terms

MSE (mse): the mean of the sum of the squared error terms

RMSE (rmse): the square root of the mean of the sum of squared error terms

Step 5: in python repeatedly choose different values for your beta vector and recompute the MSE until you make the MSE very small. If you want, use a grid search in a double-nested for loop. What do you notice about the MSE – is it possible to make it zero?

The actual values should be around *β*0 and = 26.768, and *β* 1= 0.644. If you are having trouble getting a good MSE, try these values.

Step 6: After you got the MSE as low as you have time to make it, try writing code to use your model to predict the value of someone who got an 80 on their aptitude test. Did you get about 78.288?