Assignment 2 CS405/505 Data Mining Fall 2019 Professor: Russell Butler, Johnson 114A, Office hours: MWF 9-11am
Due: Wednesday October 30 2019 11:59pm, max group size=2 students
This assignment will use the Linnerrud dataset: https://scikit-learn.org/stable/datasets/index.html
import using: from sklearn.datasets import load_linnerud
Objective 1: familiarize with vector operations for manipulating multi-dimensional arrays in numpy
Objective 2: learn basic data visualization (scatter plots and line charts)
Objective 3: understand how to code basic machine-learning algorithms from scratch
N.B: there will be no usage of sklearn or any other machine learning libraries in this assignment, the ONLY imports permitted are numpy and matplotlib (you can use sklearn to load the Linnerrud dataset only).
you must submit your scripts, any assignment submitted without python scripts attached, or with python scripts that use the sklearn libraries for question 2,3 will be penalized

Question 1: (question 1 does not use the Linnerrud dataset)
set up a working version of numpy/matplotlib in your IDE of choice (spyder, pycharm, etc.)
A) using numpy, initialize an array of random numbers each number ranging between 0 and 1
-array should have shape=[1000,50] (1000 rows, 50 columns)
B) create the correlation matrix of pearson correlations between all pairs of rows from (1A)
- correlation matrix should have shape=[1000,1000])
C) using matplotlib, plot a 100-bin histogram, using values from lower triangle of 1000x1000 correlation coefficient (r-values) matrix obtained in 1B (omit the diagonal and all cells above the diagonal)
*hint - the histogram will be shaped like a gaussian
using the histogram, estimate the probability of obtaining an r-value > 0.75 or <-0.75 from correlating two random vectors of size 50. repeat A-C with only 10 columns in (A), how does the smaller sample affect the histogram in (C)?

QUESTION 1 OUTPUT: a figure with two histograms, hist1 based on correlations of vectors of size 50, hist2 based on correlations of vectors of size 10. display the probability from (C) as the title of the histograms

Question 2:
A) get the Linnerrud data using: data = load_linnerrud()
-weight, waist, and heartrate are attributes, chinups, situps, and jumps are outcomes
B) using numpy's matrix functions (np.dot, np.transpose, etc.), compute the linear-least-squares solution, finding the intercept and slope of best fit line for each [attribute, outcome] pair (attribute on x-axis, outcome on y-axis)
*hint - be sure to augment the attribute vectors with a column of 1's (so LLS can find the intercept)

QUESTION 2 OUTPUT: a figure with a 3x3 grid of nine (9) subplots, each showing a scatter plot and best fit line:
i) x=weight, y=chinups. ii) x=weight, y=situps. iii) x=weight, y=jumps.
iv) x=waist, y=chinups. v) x=waist, y=situps. vi) x=waist, y=jumps.
vii) x=heartrate,y=chinups. viii)x=heartrate, y=situps. ix) x=heartrate,y=jumps
display the slope and intercept of each scatter plot' as the title of each scatter plot, as well as the attribute/outcome name on the x/y axis respectively

Question 3:
Implement the following two algorithms, from scratch, in python (using only the numpy import)
A) Gaussian Naive Bayes (probabilistic modeling)
B) Perceptron learning rule (Linear modeling) if perceptron does not converge run for 1000 iterations
do NOT copy-paste the sklearn code, or any other code from the internet (i will check this)
test your algorithms on the Linnerrud dataset using all 3 attributes, and only the chinups outcome,
first define new vector assigning binary classe to the outcome of chinups as follows:
if(chinups>median(chinups)) then chinups=0 else chinups=1
use these classes (0/1) to train the perceptron and build the probability table
QUESTION 3 OUTPUT: two .txt files:
gnb_results.txt => 20 probability values output by Gaussian Naive Bayes,
each value is P(chinups=1 | instance_i), where instance_i are the attributes of ith instance
perceptron_results.txt => 20 prediction values output by perceptron
each value is a weighted sum (dot product of perceptron's weights with attribute values)

Evaluation: summarize your results (plots, algorithm, outputs) in a .pdf, and attach your scripts as well in a single .zip to moodle (1per group)
marking: Question 1: 25%, Question 2: 35%, Question 3: 40%, assignment overall is worth 15% of final grade