1. The file `dataset.csv` was created artificially by simulation. It has two features X_1 and X_2 and the response (target) Y with two categories 0 and 1. Read the dataset into a dataframe `df` then split the rows in `dataset.csv` into a train (40%) and a test set (use `stratify = y, random_state = 0`). The data is already scaled. Call `X` the dataframe with the features only.

   a) (20 pts.) Make a scatterplot of X_1 (x-axis) vs X_2, (y-axis) for the train set only. Use red color for rows with `y=1`, otherwise black color. This plot shows the data points we want to classify.

   b) (20 pts.) Function `LogisticRegression( )` can be used with argument `C` for regularization. `C` is equal to the inverse of the shrinkage parameter α. Therefore $\alpha = 0$ (no regularization) is accomplished by using a large value of C, for instance `C=1e20`. Fit a logistic regression model (using predictors x_1 and x_2) with no regularization using `LogisticRegression(solver='lbfgs',C=1e20)` to the train set and report the test accuracy rate.

   c) (20 pts.) Use `X = PolynomialFeatures().fit_transform(X)` to expand `X` with columns x_1^2, x_2^2, and x_1x_2 as additional predictors (in addition to a column of ones). Report the top five rows of `X`.

   d) (20 pts.) Use `random_state = 0` to split the rows of this expanded dataset, into a train (40%) and a test set. Build a logistic regression model with no regularization using `C=1e20.` Find the test accuracy rate and the confusion matrix (cross tabulation table) for the test set. This matrix compares the predictions with the true Y values.

   e) (20 pts.) Use holdout cross validation to find the value of `C` that yields the largest test accuracy rate. Report this value and the new test accuracy rate.

Submit your report as a pdf file onto Blackboard showing your name and USC ID. Report must be made of letter size pages in portrait format (not landscape). Truncated Python commands are not acceptable.