Machine Learning

Lab 1 - Prepping Data
Spring 2025

Introduction

This lab will introduce loading, converting, and visualizing data.

You may not use any functions from machine learning library in your code, however you may use statistical functions. For example, if available you **MAY NOT** use functions like

- zscore

however you **MAY** use basic statistical functions like:

- std
- mean

Your task will be to write a *single script* such so that we can run it in the command line and it output (and/or displays) the requested information/figures.

Grading

- +1pt Generates something for Part 1
- +2pt Generates mostly correct results for Part 1
- +2pts Generates correct results for Part 1
- +1pts Does some conversion and visualization for Part 2
- +2pts Does mostly correct conversion and visualization for Part 2
- +2pts Does correct conversion and visualization for Part 2

DataSets

Yale Faces Datasaet This dataset consists of 154 images (each of which is 243x320 pixels) taken from 14 people at 11 different viewing conditions (for our purposes, the first person was removed from the official dataset so person ID=2 is the first person).

The filename of each images encode class information:

$$\text{subject} < ID > . < condition >$$

Data obtained from: http://cvc.cs.yale.edu/cvc/projects/yalefaces/yalefaces.html

Medical Cost Personal Dataset This dataset consists of data for 1338 people in a CSV file. This data for each person includes:

1. age
2. sex
3. bmi
4. children
5. smoker
6. region
7. charges

For more information, see https://www.kaggle.com/mirichoi0218/insurance

1 Processing Image Data

Download and extract the dataset *yalefaces.zip* from Blackboard. This dataset has 154 images ($N = 154$) each of which is a 243x320 image ($D = 77760$). In order to process this data your script will need to:

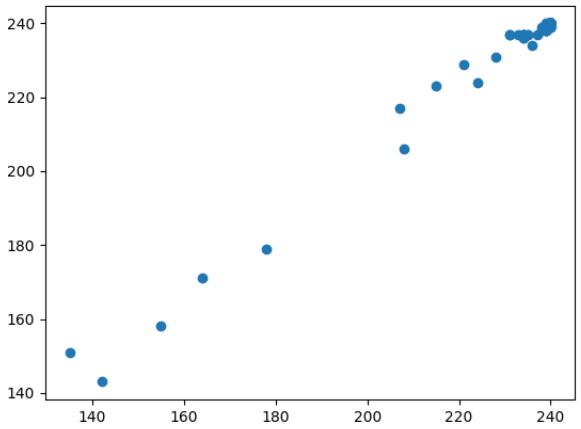
1. Read in the list of files
2. Create a 154x1600 data matrix such that for each image file
   (a) Read in the image as a 2D array (243x320 pixels)
   (b) Subsample/resize the image to become a 40x40 pixel image (for processing speed). I suggest you use your image processing library to do this for you (like pillow's *resize* method).
   (c) *Flatten* the image to a 1D array (1x1600)
   (d) Concatenate this as a row of your data matrix.

In the interest of time, I have provided a starter script, Lab1Starter.py, with code to do this.

Once you have your data matrix, your script should display two figures, as *scatter plots*:

1. Plot the first feature vs the second feature.
2. Z-score all the features, and plot the first zscored feature vs the second.

Your first plot should be something like the following:



2 Processing Mixed Data

Download the CSV files *insurance.cvs*. Write a script that converts all features to binary features. Namely:

- If any feature is continuous, use its mean to convert them to binary (age, bmi, children, charges).
- If any feature is categorical (sex, smoker, region), convert them to a set of binary features (one-hot-encode).

Have your script visualize the first vs second feature (no z-scorring necessary, since all features are now binary).