Project 1:
Dimensionality Reduction and Association Analysis

Aswin Shakil Balasubramanian
UB Name: aswinsha

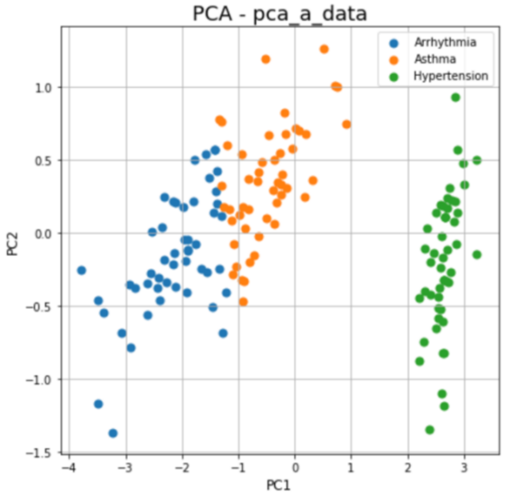
Sanidhya Chopde
UB Name: schopde

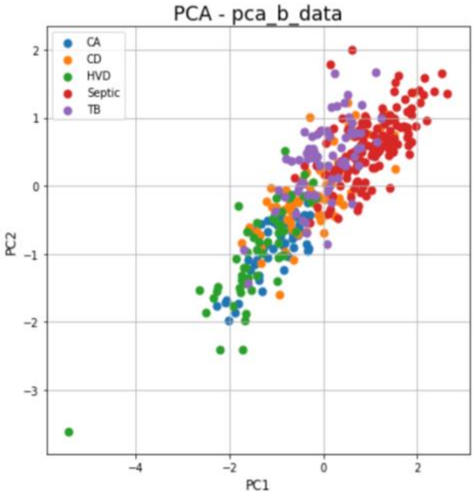
Shashank Raghunathan
UB Name: raghuna2

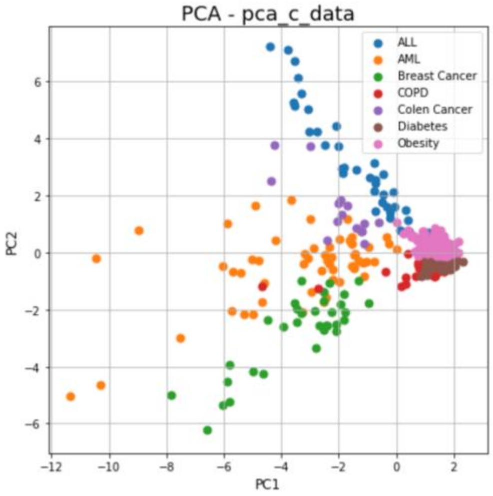
Part 1: Dimensionality Reduction

i. Following are the plots obtained by running PCA, SVD and t-SNE algorithms on the 3 medical datasets.

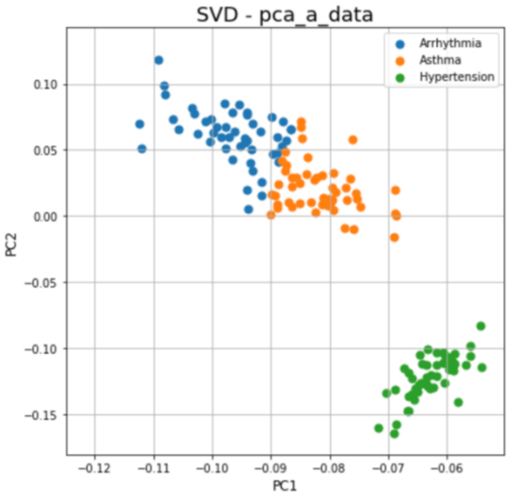
- **Principal Component Analysis (PCA)**

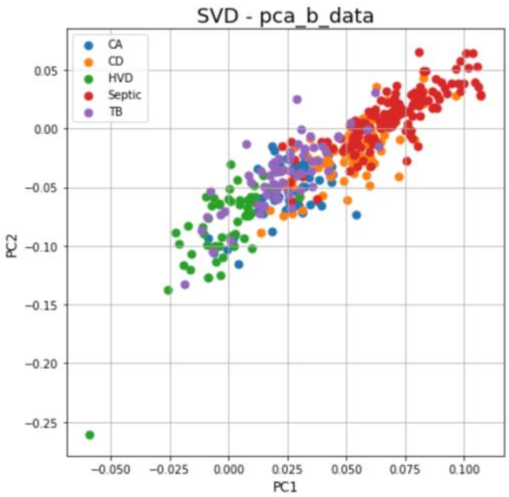


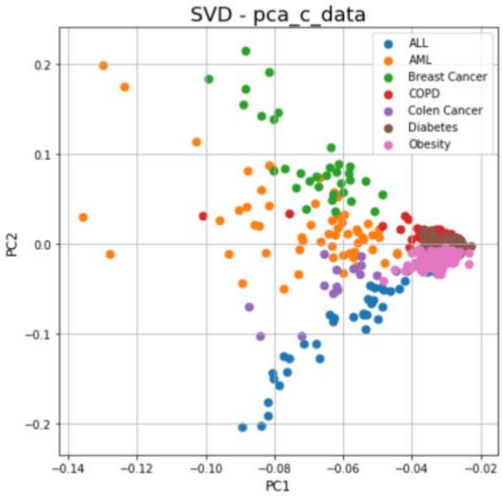




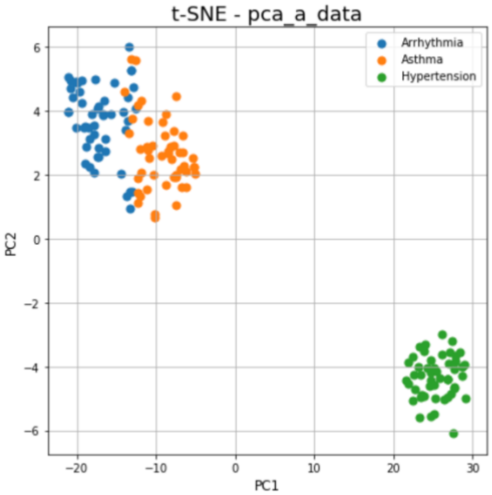
- **Singular Value Decomposition (SVD)**

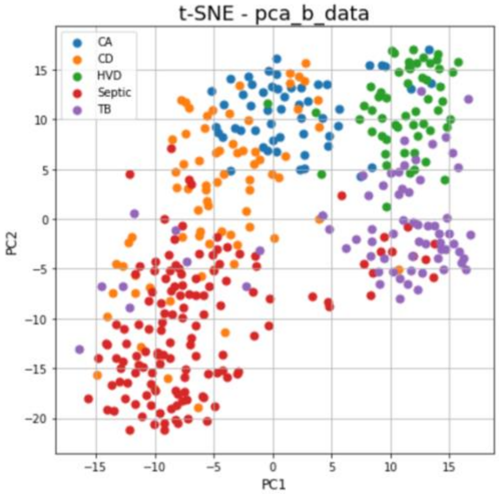


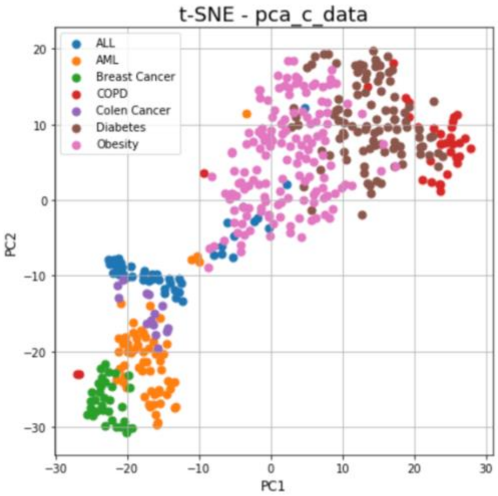




- **t-distributed stochastic neighbor embedding (t-SNE)**


t-SNE - pca_a_data


t-SNE - pca_b_data


t-SNE - pca_c_data

ii. Algorithm Flow for PCA:

We need to reduce dimensions in a n-dimensional space to a two-dimensional space. We then plot the two dimensions in which they are orthogonal and represent directions with maximum variability. To achieve this:

- First, we read the dataset into a pandas dataframe.
- We then take the mean of all columns of the dataframe excluding the last column.
- We then subtract the mean of each column with the values in that column to get the variance.
- We then convert the data into a matrix and calculate the covariance matrix by taking the product of the mean adjusted matrix with its transpose and dividing it with the total number of rows.
- Then we calculate the eigenvalues and eigenvectors using the numpy function "np.linalg.eig" which takes the covariance matrix as it's argument.
- We now choose two eigenvectors based on the top two eigenvalues.
- Doing so gives us the required two principal components.
- Finally, we plot this data using a scatter plot.

Results:

- **PCA & SVD comparison**
  After observing the plots of principal components obtained from PCA and SVD, we came to a conclusion that the results obtained for both were very similar. This is because both the algorithms are using the same technique for dimensionality reduction. The only thing that PCA does is, it skips the less significant components.

- **PCA & t-SNE comparison**
  After observing the plots of principal components obtained from PCA and t-SNE, we came to a conclusion that t-SNE is doing a better job when the dimensions are more. Also, it was observed that t-SNE was taking more time to execute in comparison to PCA.

References:

1. Lecture notes and videos by Professor Jing Gao.
2. https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html
3. https://numpy.org/doc/stable/reference/generated/numpy.linalg.svd.html
4. https://numpy.org/doc/stable/reference/generated/numpy.linalg.eig.html