

# Unsupervised Learning and Dimensionality Reduction

Tichakunda Mangono, [tmangono3](#)

March 24<sup>th</sup>, 2019

## Introduction

This paper will apply several unsupervised learning and dimensionality reduction algorithms to two separate datasets – one for **handwritten digit recognition** and another for **breast cancer diagnosis** in order to demonstrate the utility and drawbacks of feature transformation and feature selection approaches using unsupervised learning and dimensionality reduction. For unsupervised learning, k-means clustering (K-means) and the Expectation-Maximization (EM) algorithms will be explored while Principal Components Analysis (PCA), Independent Component Analysis (ICA), Randomized Projections (RP), Forward Stepwise and Backward Elimination will be investigated. The main focus is to comparatively analyze. The main benefits of clustering and dimensionality reduction is reduction of number of features which increases data interpretability, reduces training time for supervised learning if the new dimensions are used as features and possibly increases performance. Some disadvantages include loss of information, additional steps before training, and sometimes loss of the straight-forward descriptions of the original features. For each algorithm the report explores all these benefits and drawbacks in different settings.

This analysis relies heavily on the Python libraries machine learning algorithms while using Pandas, Numpy, Matplotlib and Seaborn for data manipulation and visualization.

## Datasets

The two datasets selected for this analysis are the following classification problems: i) **handwritten digit recognition** (referred to as “*digits*” for the rest of this report) – a labelled dataset of 1797 (8x8 pixel) images of handwritten digits with 10 classes (from 0 to 9), and ii) **breast cancer diagnosis** (referred to as “*diagnosis*” for the rest of this report) – a binary labelled dataset of 569 instances with 30 features computed from a digitized image of a fine needle aspirate of a breast mass, describing the characteristics of cell nuclei present in the image.

While the digits and diagnosis datasets are individually interesting for representing the major topics of optical character recognition and predictive disease diagnosis, they also represent two contrasting approaches of classifying data from images. The digits dataset represents raw, primary pixel-level information, while the diagnosis dataset is a set of carefully computed secondary features of the area of interest (cell nucleus) in an image. As a result, the feature-space of the two datasets represent two extremes - the digits data have a homogenous while diagnosis data has heterogeneous feature-space. As such, we should observe interesting similarities and contrasts in the performance of different unsupervised learning algorithms on these datasets. Furthermore, both datasets will result in interesting performance and analytic differences for dimensionality reduction as well.

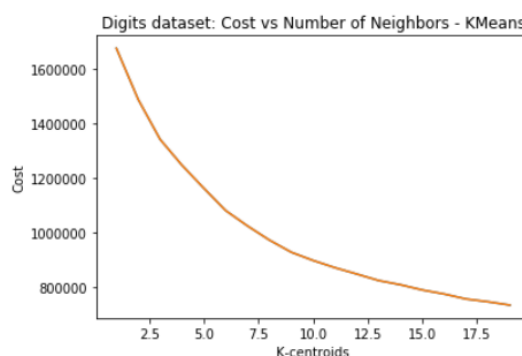
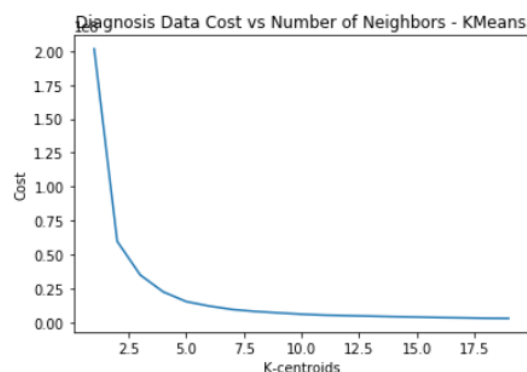
For analysis, the datasets are split into train, validation and test sets comprising 60%, 20%, and 20% of the full dataset respectively. The test set is extracted first, and then the algorithms learn on the train set, first performing clustering methods then dimensionality reduction and analyzing the results. Both datasets are available as native datasets within the Scikit Learn library.

## Clustering Approaches

Applying the two unsupervised learning algorithms of K-means and EM, using different distance measures and analyzing the results.

### K-Means Clustering

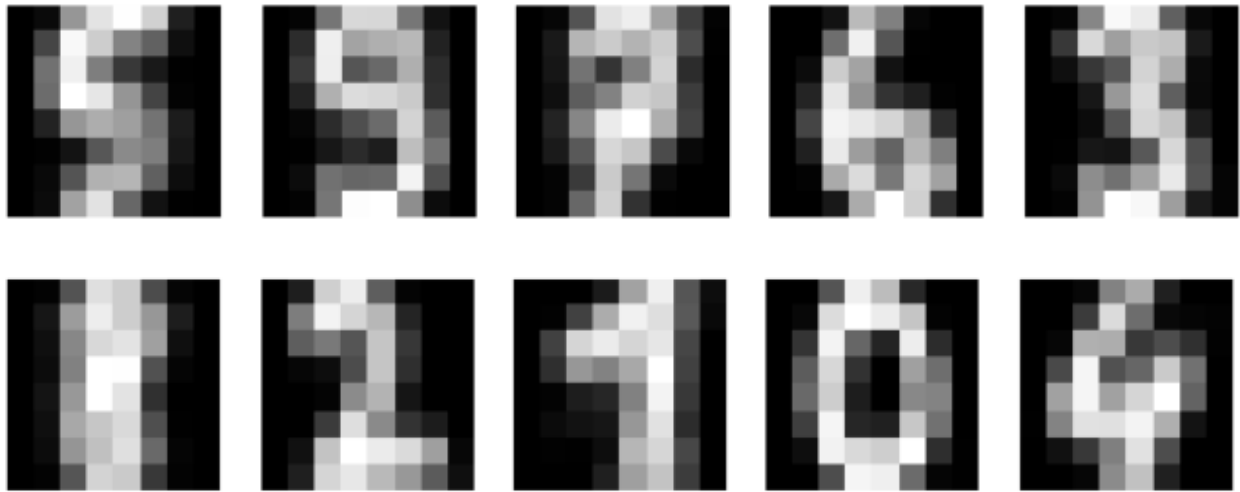
This method works by picking K centroids, applying them to the data and classifying every point to the centroid to which it is nearest, using Euclidean distance and then calculating new centroids based on the new clusters, and so on until “convergence”. Using Euclidean distance makes sense for both data sets because they are both image-based data sets and images represent the physical world in pixels, and additionally the features extracted from images are based on Euclidean distance. The presence and absence of pixels and their relative positions is what distinguishes images and this is best measured by Euclidean distance.



To choose number of centroids, the charts above show that the elbows for the diagnosis and digits datasets fall at 2, and 10 respectively. These are the points at which adding an extra centroid leads to only a marginal decrease in the cost/loss function, and thus it is not as valuable to optimizing the clustering. So use k=2 and k=10 as number centroids.

```
confusion_matrix((y2_ts-1)*-1, model.predict(x2_ts))  
array([[87,  1],  
       [ 9, 17]], dtype=int64)  
  
accuracy_score((y2_ts-1)*-1, model.predict(x2_ts))  
0.9122807017543859
```

Using Kmeans with  $k=2$  alone achieves over 91% accuracy on the diagnosis dataset with the confusion matrix shown above, which means that kmeans was able to learn the data description very well across the 30+ features.



For the digits dataset when we cluster and predict on the held out test set, and average the pixels in each of the predicted clusters in the test set, plotting the average image in each predicted cluster gives the above which clearly resembles digits. Even though we the clustering for 8's and to a lesser extent, for 7's seems kind of tough for the clustering to represent clearly, we can make out all the other digits with clarity. This means that the clustering learned the description of the pixel data very well! If this was not the case, we would have seen more jumbled images with random pixels not even resembling digits of any kind.

### **Expectation Maximization**

This method maximizes the likelihood of the data having been sampled from  $K$ - Gaussian processes of known variances. The expectation step computes the probabilities of the data points belonging to each of the  $k$  gaussians then the maximization steps looks at all probabilities and computes the mean of the Gaussian if all the points allocated to it actually belonged to it.

Since we already know the optimal number of clusters from K-means, we assume the same here and we also adopt the same assumptions on distance metrics for the same rationale as above. The EM algorithm, actually outperforms K-Means for the cancer (95% vs. 91%) with a less mistakes on the confusion table. This is likely because it benefits for the data being at least linearly separable (only two classes) and additionally benefits for EM being able to make better decisions than K-means for ambiguous data points since it is probability based as opposed to k-means which is a heuristic method.

```
Accuracy: 0.9473684210526315
```

```
Confusion Matrix:
```

```
array([[82, 6],  
       [ 0, 26]], dtype=int64)
```

However, on the digits dataset, the EM algorithm has trouble making out the 1's, 5's, 7's and 8's which suggests a lot more confusion than K-means when classifying these 10 classes. This must be because there are more classes which can easily have similar probabilities and easily mistaken for each other. However, K-means can do a little better at this as it tends to use an average statistic (centroids) which makes rigid decisions as opposed to distribution statistic which can be confused when clusters are more similar, thus can be indecisive in the clustering.

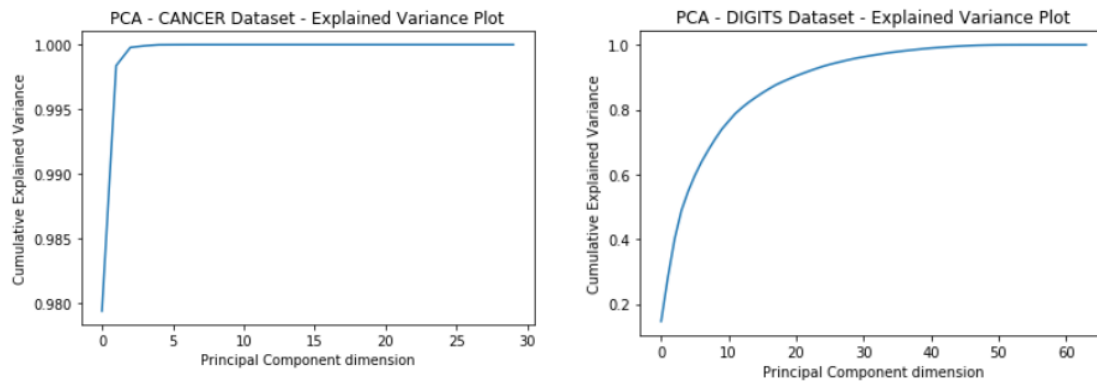


## Dimensionality Reduction Approaches

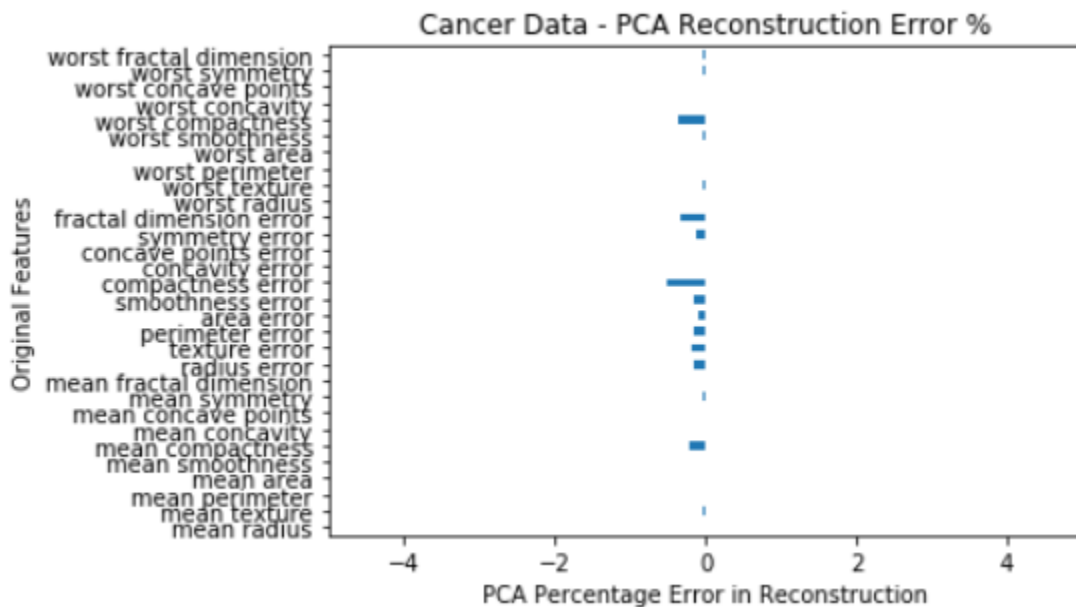
### PCA

Principal Component Analysis is a dimensionality reduction technique which finds principal components (linear combinations of the existing features) in such a way to preserve the variance (and likely the information) of the data while making sure that consecutive components are not correlated (i.e. they

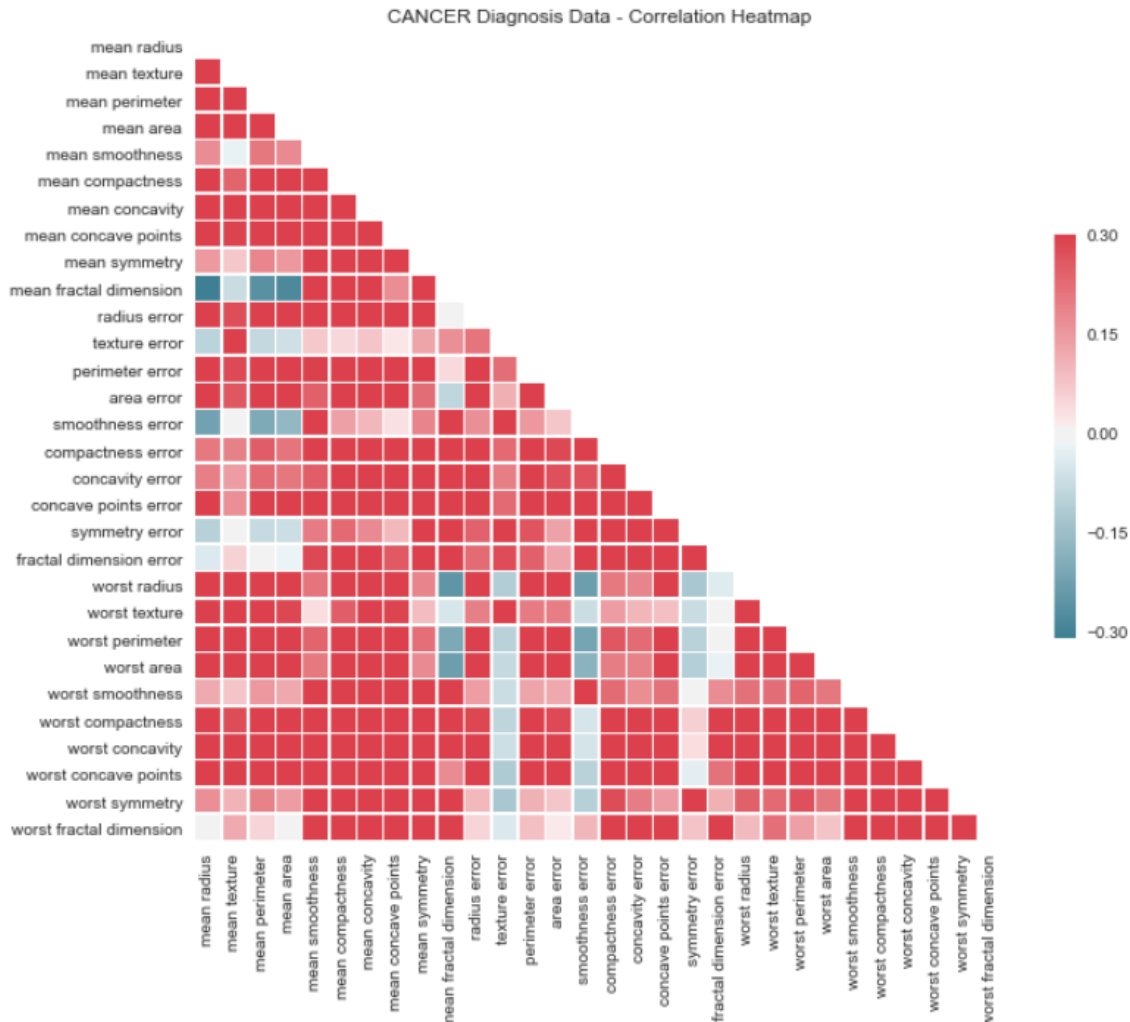
are orthogonal to each other. This preserves the information in the data while reducing redundancy.













For the diagnosis data, the first 2 components contain almost all of the information/variance while in the digits data, about 10 components contain almost 80% of the variance (and 29 components contain 95% of the variance). This is because there is more diversity of expression in the 10-class, 64-feature digits data than the 2-class, 34-feature diagnosis data.








When calculated for across all features and normalized, the error on the diagnosis data is very small, less than 1% error in each feature. This suggests the reduction from 34 features to only 2 principal components is very effective for this data set. A correlation heatmap confirms this is due to high multi-collinearity within the dataset where most variables are correlated to every other variable.



When PCA with 10 components is applied to digits data, there is a slight loss in image quality and distinctness i.e. the image becomes more blurry but there is also potentially a saving in training time (if we use principal components as features in supervised learning) because the features have now been reduced from 64 to 10, so only 15% number of features to retain about 80% of the variance in the data which is a good advantage

Index#	5	111	567	33	45
Original					
Reconstructed PCA					

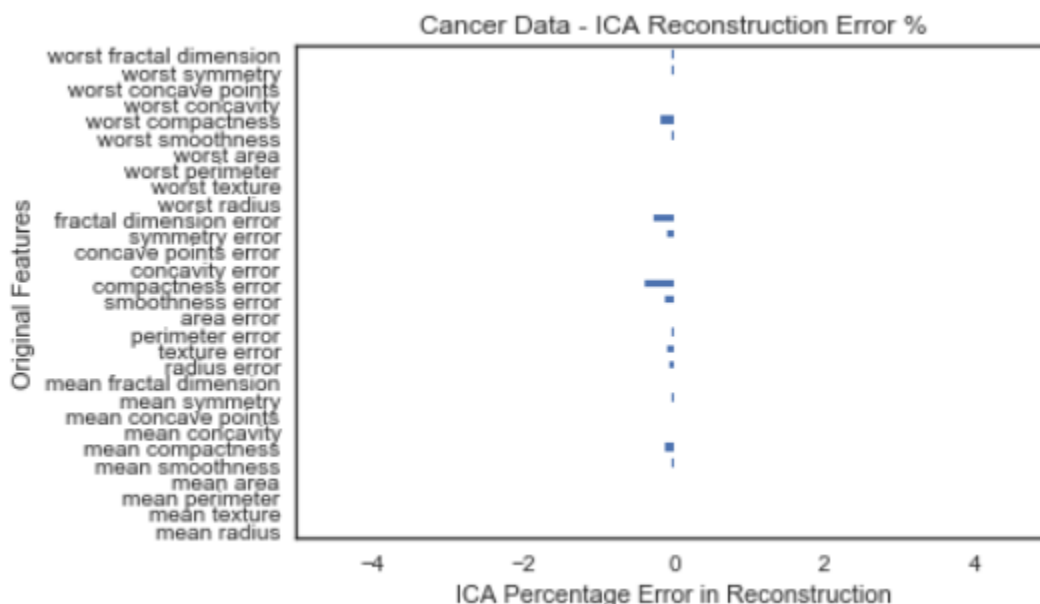
Reconstructed ICA					
-------------------	---	---	---	---	---

## ICA

Independent Component Analysis (ICA) is similar to PCA but instead of maximizing variance, it maximizes the independence of principal components such that they are linearly independent.

For the digits data, as shown in the table above, the reconstructed ICA image samples are similar to PCA – they both lose a little clarity when compared to original.

For the diagnosis data, the average for each feature is also very small (see chart below) just like the error for PCA on the same data - indicating performance for ICA and PCA are at par for this data.



## Forward Stepwise Selection

Feature selection methods use tests of information and relevance to select the best features. Below is a chart of features organized by increasing p-values for the diagnosis dataset. Taking the first 5 features, this would suggest that area error, mean area, worst perimeter, worst area and mean perimeter are the best variables to include as selected features in a supervised learning model.

