MACHINE LEARNING & IMAGE PROCESSING

WEEK 06 - PART II

DIMENTIONALITY REDUCTION ALGORITHMS

Thakshila Dasun
BSc. Hons in Mechanical Engineering
(Mechatronics Specialization)
CIMA, UK
Academy of Innovative Education

Principal Component Analysis (1)

- Using PCA for dimensionality reduction involves zeroing out one or more of the smallest principal components, resulting in a lower-dimensional projection of the data that preserves the maximal data variance.

```
pca = PCA(n_components=1)
pca.fit(X)
X_pca = pca.transform(X)
print("original shape:   ", X.shape)
print("transformed shape:", X_pca.shape)
```

```
original shape:    (200, 2)
transformed shape: (200, 1)
```

- The transformed data has been reduced to a single dimension

Principal Component Analysis (2)

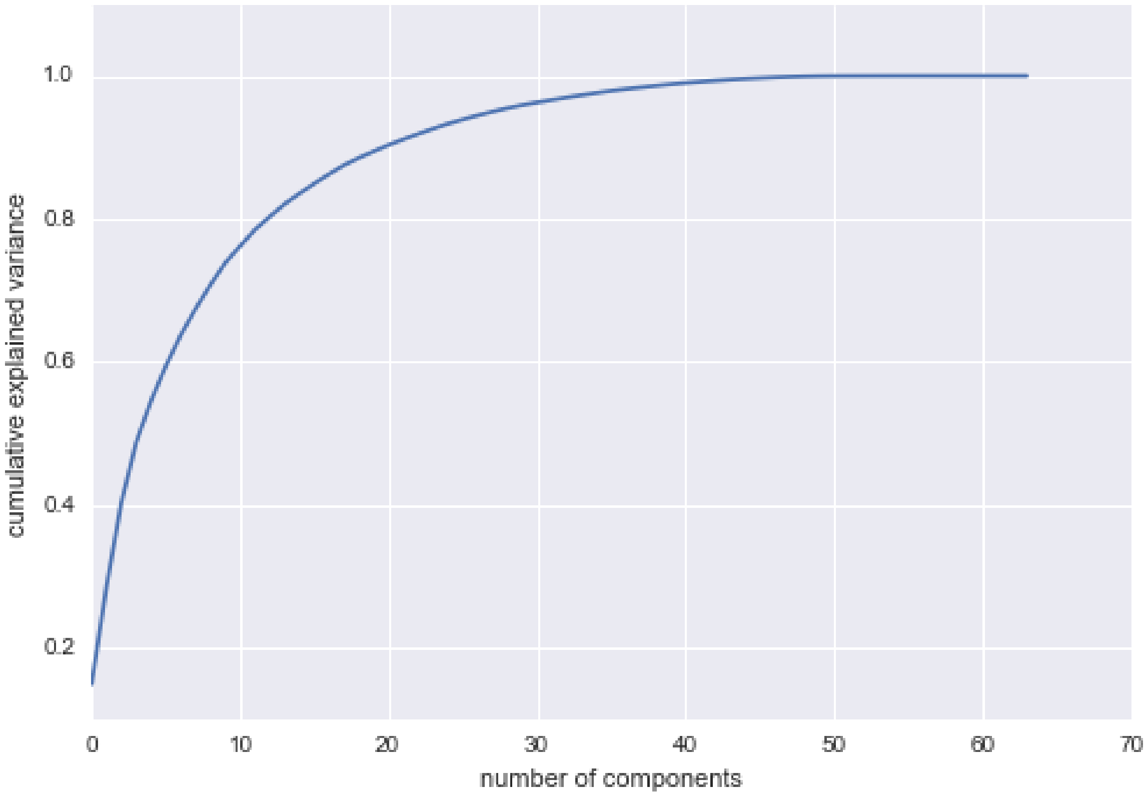
- The information along the least important principal axis or axes is removed, leaving only the component(s) of the data with the highest variance.
- The fraction of variance that is cut out (proportional to the spread of points about the line formed in this figure) is roughly a measure of how much "information" is discarded in this reduction of dimensionality.

Principal Component Analysis (2)

- This reduced-dimension dataset is in some senses "good enough" to encode the most important relationships between the points: despite reducing the dimension of the data by 50%, the overall relationship between the data points are mostly preserved.
- The usefulness of the dimensionality reduction may not be entirely apparent in only two dimensions, but becomes much more clear when looking at high-dimensional data. To see this, let's take a quick look at the application of PCA to the digits dataset.

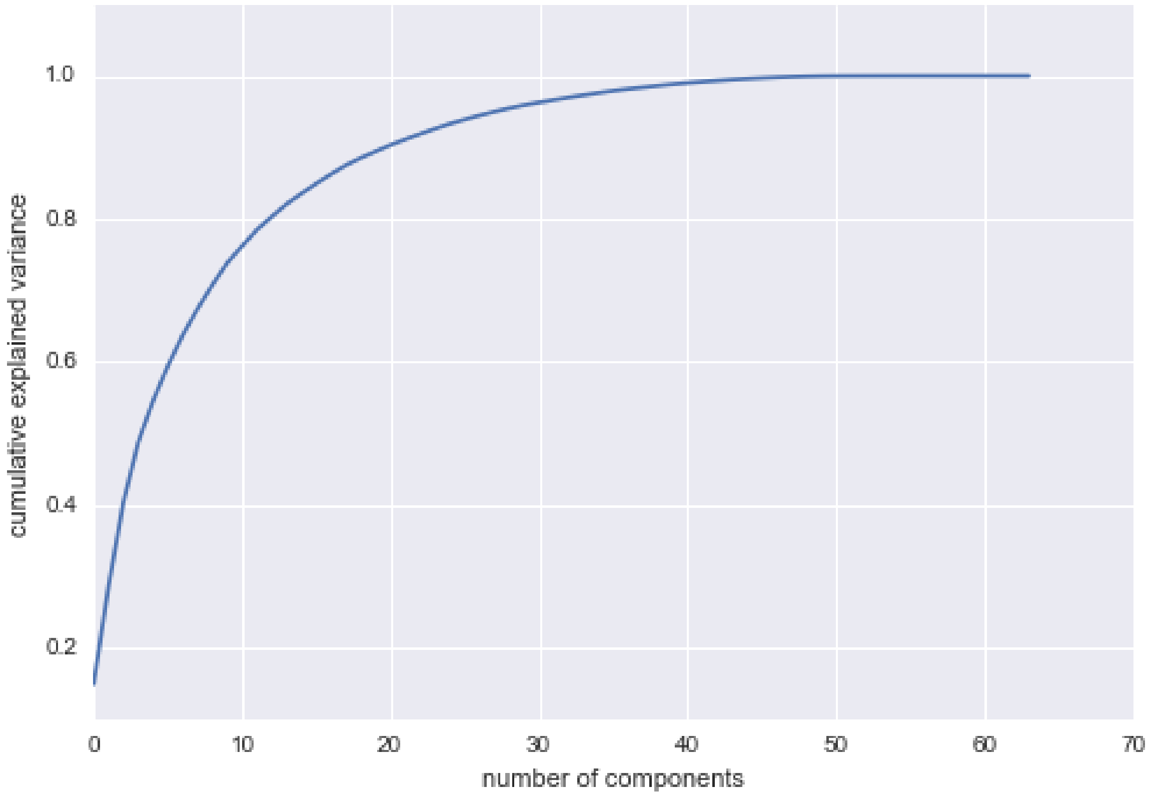
Choosing the number of components (1)

- A vital part of using PCA in practice is the ability to estimate how many components are needed to describe the data.
- This can be determined by looking at the cumulative *explained variance ratio* as a function of the number of components:



Choosing the number of components (2)

- This curve quantifies how much of the total, 64-dimensional variance is contained within the first N components.
- For example, we see that with the digits the first 10 components contain approximately 75% of the variance, while you need around 50 components to describe close to 100% of the variance.



Choosing the number of components (3)

- Here we see that our two-dimensional projection loses a lot of information (as measured by the explained variance) and that we'd need about 20 components to retain 90% of the variance.
- Looking at this plot for a high-dimensional dataset can help you understand the level of redundancy present in multiple observations.