Very brief introduction to dimensionality reduction

CS 550: Machine Learning

Problems of Dimensionality

- It is often reasonable to believe that the performance will improve with the use of additional features
  - No feature is useless unless the means for the two classes are the same

- However, it has frequently been observed in practice that beyond a certain point, the use of additional features leads to worse performance
  - **Curse of dimensionality:** As the dimensionality increases, much more samples are necessary to have a good generalization (to avoid overfitting)
  - Ignoring irrelevant features would improve accuracy

Dimensionality Reduction

- We may want to reduce the dimensionality and find the "intrinsic" dimensionality of data
  - To avoid overfitting and disregard irrelevant features
  - To visualize high dimensional data
- The dimensionality reduction is typically achieved by
  - Selecting a subset of the existing features or
  - Combining the existing features

Feature Selection

- Select a subset of features that yields the highest score
- Need to examine all possible subsets of the given size
  - Impractical (an exhaustive search)
  - Sequential procedures are often used
    - They add or remove features sequentially
    - Common procedures are forward selection and backward elimination
- Common scoring methods:
  - Training or cross-validation accuracy (not test set accuracy)
  - Mutual information between the features and the output
    - Mutual information between two random variables quantifies their mutual dependence

$$\hat{I}(X,Y) = \sum_x \sum_y \hat{P}(X = x, Y = y) \log \frac{\hat{P}(X = x, Y = y)}{\hat{P}(X = x)\hat{P}(Y = y)}$$

Feature Selection

Forward selection

- Start with an empty set of features
- Incrementally expand the subset by adding a feature
  - Features are added so that the subsequent subsets lead to the highest score
- Terminate the algorithm if the specified number of features are reached
  - Or alternatively, if no additional feature yields a better score

Feature Selection

Backward elimination

- Start with a complete set of features
- Incrementally remove the features one at a time
  - Features are removed so that the subsequent subsets lead to the highest score
- Terminate the algorithm if the specified number of features are reached
  - Or alternatively, if the score significantly decreases with a removal of a feature

Feature Selection

- Forward selection and backward elimination are greedy algorithms
  - They do not guarantee to find the global optimal solution

- These algorithms select the features assuming that they are independent
  - However, there might be features that do not yield a good score when they are used alone but yield better scores when they are used in conjunction with other features
    - Such complimentary features cannot be captured by these algorithms

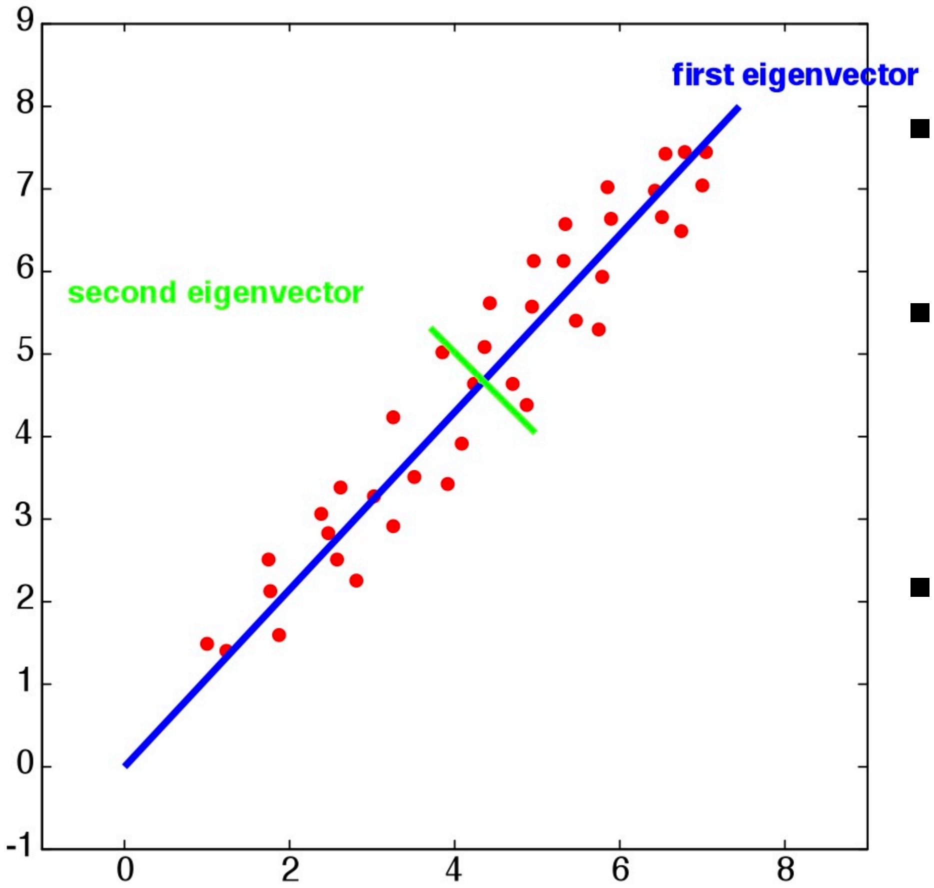
Feature Reduction

- We create new features as functions of the existing ones (instead of choosing a subset of the existing features)
  - New features may not have a clear physical meaning
  - We may use linear or non-linear combinations
- Linear combinations are particularly attractive
  - They are simple to compute and analytically tractable
  - They project the high-dimensional data onto a lower dimensional space
- This could be achieved in
  - Unsupervised manner (e.g., principal component analysis chooses a projection that is efficient for representation)
  - Supervised manner (e.g., linear discriminant analysis chooses a projection that is efficient for discrimination)

Principal Component Analysis

- The aim is to find a new feature space with minimum loss of information
- It is assumed that the "most important" aspects of the data lies on the projection with the greatest variance
  - It is often the case, but of course it depends on the application
- Principal component analysis (PCA) transforms the data to a new coordinate system such that
  - The greatest variance lies on the first coordinate (the first principal component), the second greatest variance lies on the second coordinate (the second principal component), and so on
  - The eigenvectors of the covariance matrix of the data correspond to these principal components

Principal Component Analysis



- Find the covariance matrix of the data set
- Find the eigenvectors and eigenvalues of the covariance matrix
- First *n* eigenvectors (with the largest eigenvalue magnitudes) will correspond to the first *n* principal components