CHAPTER 6: PREPROCESING

Grado en Ingeniería Informática

Curso 2014 / 15

Topics

1. Data encoding
2. Obtaining a complete dataset
3. Normalization
4. Dimensionality reduction

DATA ENCODING

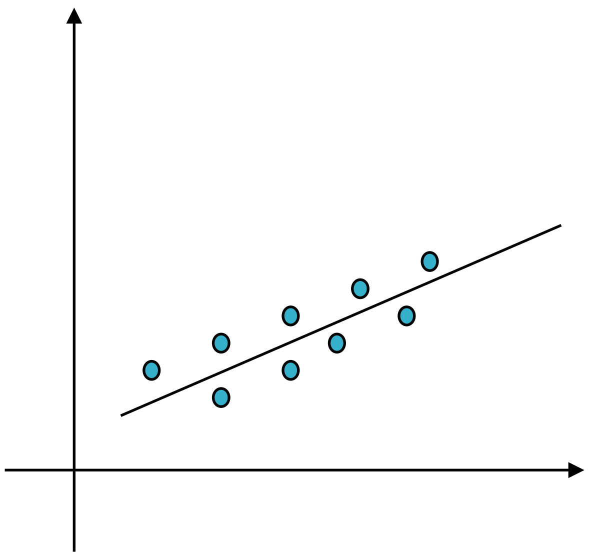
Encoding

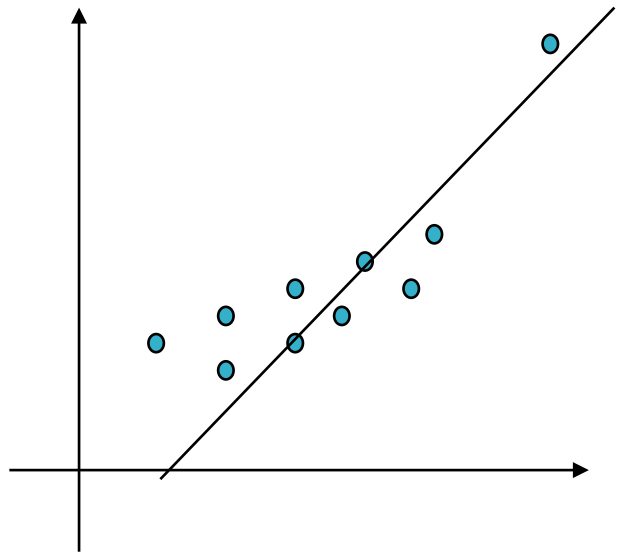
- Numerical
  - Normalize to the range [0,1] or [-1,1]
- Ordinal
  - Ordered set of discrete values. Examples: {much, pretty, little, nothing}, {always, often, normal, sometimes, never}. They are typically represented as numerical data
- Nominal
  - Unordered set of discrete values. Examples: {yes, no}, {white, red, green}. They are represented as N bits, one of which is 1 and the rest 0's.

OBTAINING A COMPLETE DATASET

Isolated data

- The isolated or erroneous data ALWAYS are a problem
- The existence of isolated data can totally invalidate a solution





Isolated data: solution

1. Suppose that the data come from a multivariate normal distribution function
2. Estimating the mean and the covariance matrix
3. Determine data whose corresponding probability density falls below a certain threshold
4. Deleting the data from the database
5. Thus, the isolated or erroneous data is detected and removed
6. There is a danger of eliminating correct data

Incomplete data

- A common problem is the existence of patterns with unknown variable or characteristics
- Solutions:
  - Remove incomplete patterns
  - Random packing
  - Replaced by the average of data known
  - Regression modeling and subsequent use of it, to fill in the missing values

NORMALIZATION

Simple normalization

- Applies a linear transformation so that all entries have similar values
  - Each variable are treated independently for each xi, its mean and variance
  - The set of variables rescaled are defined as:

$$\overline{x_i} = \frac{1}{N}\sum_{n=1}^{N} x_i^n \qquad \sigma_i^2 = \frac{1}{N-1}\sum_{n=1}^{N}\left(x_i^n - \overline{x_i}\right)^2$$

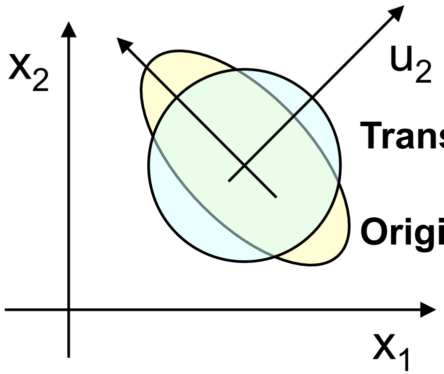
$$\tilde{x}_i^n = \frac{x_i^n - \overline{x_i}}{\sigma_i} \qquad \tilde{x}_i^n = \left\langle \begin{array}{l} \overline{\tilde{x}_i^n} = 0 \\ \sigma^2_{\tilde{x}_i^n} = 1 \end{array} \right.$$

Eigenvalues and eigenvectors

- **Defintion**: An eigenvector or characteristic vector of a square matrix is a non-zero vector that, when multiplied with, yields a scalar multiple of itself; the scalar multiplied is often denoted by λ. That is: $Av = \lambda v$. The number is called the eigenvalue or characteristic value of corresponding to ν.
- **En Matlab:**
  - E = eig(X) -> returns a column vector containing the eigenvalues of square matrix X.
  - [V, D] = eig(X) -> D is a diagonal matrix containing the eigenvalues. V is a matrix whose columns are the corresponding right eigenvectors of matrix X such that X*V = V * D

Normalization - Whitening

- Features are not statistically independent, and then the correlation should be taken into account



$$\Sigma = \frac{1}{N-1}\sum_{n=1}^{N}\left(x^n - \overline{x}\right)\left(x^n - \overline{x}\right)^T$$

$$\Sigma u_j = \lambda_j u_j$$

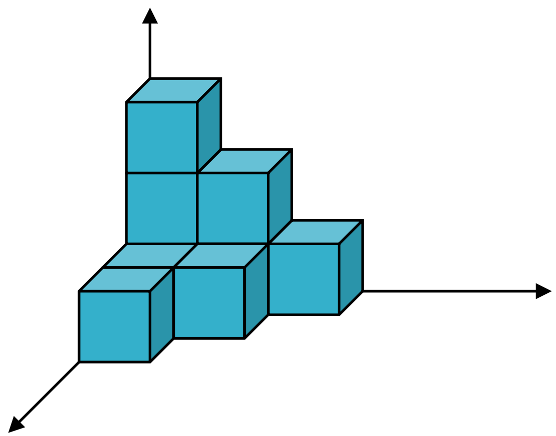
Getting the eigenvectors and linear transformation:

$$\widetilde{x}^n = \Lambda^{-1/2} U^T \left(x^n - \overline{x}\right) \qquad \widetilde{x}^n = \left\langle \begin{array}{l} \overline{\widetilde{x}^n} = 0 \\ \Sigma = I \end{array} \right.$$

DIMENSIONALITY REDUCTION

Dimensionality reduction

- Advantage
  - The smaller is the dimension of the input space, the less the number of parameters to determine
  - Faster learning
  - Both parameters have aquadratic dimension dependence
- Major drawback
  - By reducing the dimension ALWAYS information is lost



Feature selection

- It is the selection of those characteristics that influence the problem, and discard those that do not
- The most common methods are:
  - Comprehensive methods
  - Stepwise selection

Selection: comprehensive

- Consists of looking in depth the m characteristics (m <d) between the d original features

- The process requires selecting all possible forms d elements from m in m, thus equal to:

$$\begin{pmatrix} d \\ m \end{pmatrix}$$

Stepwise selection

- Incremental
  - Select the feature that further increases the recognition rate
  - Add the next best combined with the earlier
  - etc. Etc.

- Decremental
  - Eliminating feature least reduces the recognition rate
  - Remove the next worst combined with the earlier
  - Etc. Etc.

Feature combination

- It is the transformation from the original features in other more efficient, yielding a new representation.
- This process usually involves a reduction in the dimension of the input space
- The most common methods are:
  - Linear Transformations: PCA, Fisher
  - Nonlinear transformations: ICA

Feature combination

- Linear
  - Consist of seek a matrix $W_{m \times d}$, such that:

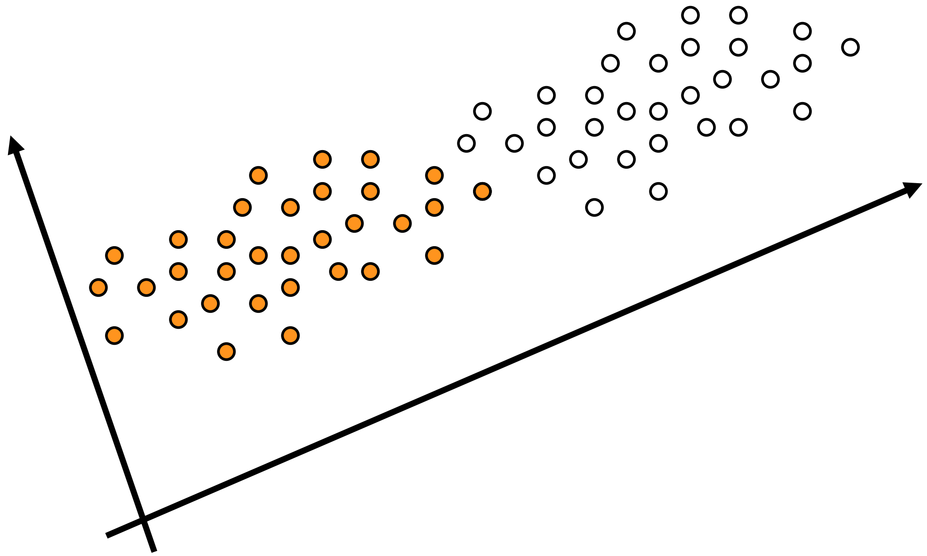
$$x' = W * x$$

  - There are two basic methods:
    - Transformations uncorrelated (PCA)
    - Discriminant transformation (Fisher)

- Nonlinear
  - Very complex systems

PCA – Principal Components Analysis

- The aim is to perform a linear transformation that meets that:
  - The transformed features are uncorrelated
  - You must conserve the variance of the data
- The classic method is the Principal Component Analysis (PCA)



PCA Procedure

1. Subtract each feature their average value
2. Calculating the covariance matrix (C)
3. Calculate the eigenvalues and eigenvectors of C
4. Sort highest to lowest eigenvectors according to their corresponding eigenvalue
5. Create the matrix M with so many eigenvectors as features are desired in the transformed space
6. Get the vectors transformed as:

$$y = M * x$$

PCA Properties

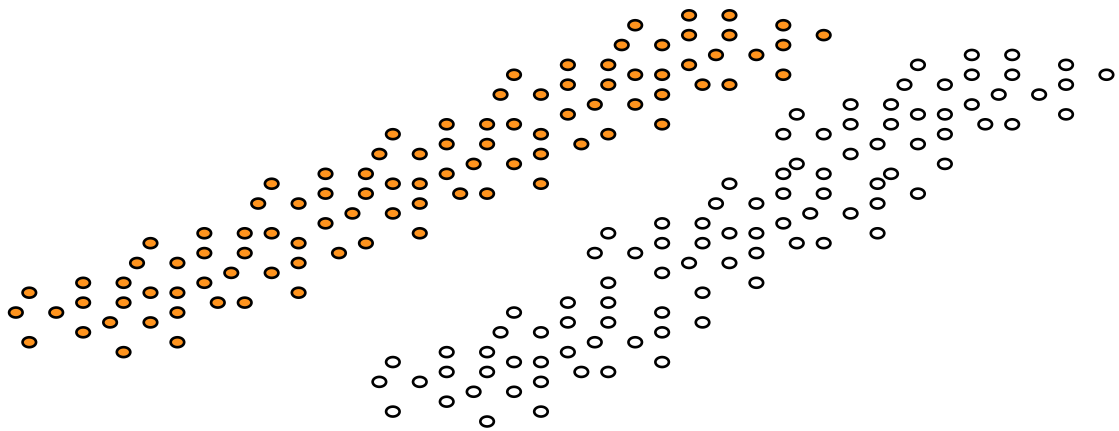
- The effectiveness of each characteristic is given by the corresponding eigenvalue. Therefore, if we order the eigenvalues, we can determine the best features

- Retained variance in the new space is the sum of the eigenvalues of the characteristics retained

- The covariance transform matrix is diagonal, the transformed features are uncorrelated

Drawback of PCA classification

- The characteristics with higher eigenvalues not always retain the separation between classes.



- This transformation is suitable for data compression but it is not always valid for classification.

Fisher discriminant transformation

1. The classical discriminant transformation is due to Fisher
2. S_w is the average of the covariance matrices
3. m the average of the averages of each class
4. $S_b = (m_i - m) * (m_i - m)^T$
5. W are determined as the maximize ratio between S_b and S_w
6. W are determined following a procedure similar to PCA, as the eigenvectors of:

$$C = S_w^{-1} * S_b$$

Fisher procedure

1. Calculating the covariance matrix of each class (C_i)
2. S_w calculated as the average of C_i
3. Calculate the mean of each class (m_y)
4. Calculate the mean of the mean (m)
5. Calculate $S_b = (m_i - m) * (m_i - m)^T$
6. Get $C = S_w^{-1} * S_b$, and its eigenvectors
7. Sort highest to lowest eigenvectors according to their corresponding eigenvalue
8. Create the matrix M with so many eigenvectors as features are desired in the transformed space
9. Get the vectors transformed as:

$$y = M * x$$

Fisher discriminant drawback

- The number of extracted features is at most equal to the number of classes least 1