Missing Values and Outliers; Removing Outliers using k-means Algorithm

WS 2015/2016

Introduction

The aim of this tutorial is to emphasize the importance of data preprocessing. You will learn basic methods how to cope with missing values and outliers.

1 Missing values

The occurrence of missing values has various causes, be it a broken sensor or people who forgot to fill in some questionnaire. In some cases some attributes make no sense for some types of objects (for example: pregnancy for males).

It is very important to identify type of the missing value, because otherwise the analysis could lead to meaningless conclusions. If there are missing values, we often suppose that they are *Missing Completely At Random*. Such an error can be imagined as a random ink blot on the paper with the printed data. Contrarily, *Missing At Random* model works with the assumption, that probability of a missing value depends on some other variable. Thermometer may measure incorrect values with probability $p > 0$ just in case of rain, otherwise it may work well. *Nonignorable Missing Values* are the worst. An example could be a broken thermometer which may not measure temperatures under 0°C.

Imputation

Missing values cannot be always excluded. The simplest way is to replace a missing value by the mean value in case of numeric attributes, or by modus in case of categorical attributes. Another option is to replace the missing value by the respective value of its nearest neighbour (NN).

2 Normalisation

Normalisation is a procedure used to eliminate unequal contribution of attributes. Typical approaches are *min-max normalisation* and *z-score standardisation*.

- **min-max normalisation:** For a numeric attribute x the min-max normalisation is defined as:

$$x \mapsto \frac{x - min_x}{max_x - min_x},$$

where min_x and max_x are minimum and maximum values of the attribute

- **Z-score standardisation:** For a numeric attribute x the z-score standardisation is defined as:

$$x \mapsto \frac{x - \hat{\mu}_x}{\hat{\sigma}_x},$$

where $\hat{\mu}_x$ is mean and $\hat{\sigma}_x$ is standard deviation of the attribute

3 Outliers

Outliers usually result from erroneous measurements. If it is not possible to correct such data, it is necessary to remove it from the dataset. It can happen that some outliers are not really outliers, their values are significantly different just by chance. Despite this fact, it can be suitable to remove them and analyse them separately.

Outliers depend on the distribution function. When detecting outliers for categorical data, we assume that outliers occur with extremely low frequency. The situation is more difficult when dealing with numerical attributes. The problem is that in case of large datasets there will be always some points marked as outliers.

Univariate outliers

Let $\overline{x}$ be average value and s be standard deviation of a normal distribution. We say that an observation is an outlier if lies outside of the interval

$$(\overline{x} - ks, \overline{x} + ks),$$

where k is usually 2 or 3.

Multivariate outliers

When dealing with multi-dimensional data, we usually do not use any special assumptions about the distribution function. It is possible to exploit dimensionality reduction methods and try to identify the outliers by visual inspection of a low-dimensional plot of these data. Alternatively, we can use clustering techniques, where we mark the data as outliers when they cannot be meaningfully assigned to any cluster. An example of such an approach is the algorithm COR [**?**].

Algorithm COR

IN

X	data
m	number of clusters
th	threshold
R	number of iterations

OUT

X	data without outliers

$i \leftarrow 0$
repeat
  $i \leftarrow i + 1$
  $len \leftarrow numOfSamples(X)$
  $P_{new}, C_{new} \leftarrow k\_means(X, m)$
  {BEGIN: outlier removal}
  $X_{new} \leftarrow X$
  for cluster $c \subseteq X_{new}$ **do**
    if $numOfSamples(c) > 1$ **then**
      $s_{max} \leftarrow maxDistantSampleFromCentroid(c)$
      $s_{min} \leftarrow minDistantSampleFromCentroid(c)$
      $distortion \leftarrow distanceFromCentroid(s_{min}, c)/$
        $/distanceFromCentroid(s_{max}, c)$
      if $distortion < th$ **then**
        $P_{new}, C_{new}, X_{new} \leftarrow removeSample(s_{max}, X_{new})$
      end if
    else
      $P_{new}, C_{new}, X_{new} \leftarrow removeCluster(c, X_{new})$
      $m \leftarrow m - 1$
    end if
  end for
  {END: outlier removal}
  $len_{new} \leftarrow numOfSamples(X_{new})$
  $X \leftarrow X_{new}$
until $i > R \wedge len_{new} == len$
$return(X)$

Work assignment

1. Replace missing values using the nearest neighbours method.
2. Normalize data from the previous point using z-score standardisation. Discuss influence of standardisation on PCA reduction according to the two-dimensional plots showing data projected to the first two principal components.
3. Find and remove outliers in `outlier_data.mat` using algorithm COR.

Expected results

1. Replacing missing values using nearest neighbours method should yield results which look better. A clear difference between both algorithms is visible after rotating the 3D plot properly.[1]

2. The effect of z-score on the data gives two different plots, one of which is correct. You should be able to pick the correct one at first sight.

3. The data contain five artificially added outliers. It is possible to detect three of them by univariate methods. The remaining two objects are visibly outliers, but it is not possible to remove them by univariate approaches.

Theory questions

1. What breaks down if z-score standardization is omitted before PCA?

2. Is PCA affected by the choice of units (for example inches vs. cm)?

3. Both PCA and linear regression find a mapping from a high-dimensional space to a low-dimensional space and both use the least-square-error criterion to fit the data. Is there a difference between the two techniques?

4. What is the asymptotic complexity of the basic NN algorithm implemented in Matlab? Can you do better? Which data structure would you use?

[1] Fill the missing values only using the data without any missing value. For finding the NN, use Euclidean distance.