Data Preparation
(Data pre-processing)
Data Preparation

- Introduction to Data Preparation
- Types of Data
- Discretization of Continuous Variables
- Outliers
- Data Transformation
- Missing Data
- Handling Redundancy
- Sampling and Unbalanced Datasets
INTRODUCTION TO DATA PREPARATION
Why Prepare Data?

• Some data preparation is needed for all mining tools

• The purpose of preparation is to transform data sets so that their information content is best exposed to the mining tool

• Error prediction rate should be lower (or the same) after the preparation as before it
Why Prepare Data?

• Preparing data also prepares the miner so that when using prepared data the miner produces better models, faster

• GIGO - good data is a prerequisite for producing effective models of any type
Why Prepare Data?

• Data need to be formatted for a given software tool
• Data need to be made adequate for a given method
• Data in the real world is dirty
  • incomplete: lacking attribute values, lacking certain attributes of interest, or containing only aggregate data
    • e.g., occupation=""
  • noisy: containing errors or outliers
    • e.g., Salary="-10", Age="222"
  • inconsistent: containing discrepancies in codes or names
    • e.g., Age="42" Birthday="03/07/1997"
    • e.g., Was rating “1,2,3”, now rating “A, B, C”
    • e.g., discrepancy between duplicate records
    • e.g., Endereço: travessa da Igreja de Nevogilde Freguesia: Paranhos
Major Tasks in Data Preparation

• Data discretization
  • Part of data reduction but with particular importance, especially for numerical data

• Data cleaning
  • Fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies

• Data integration
  • Integration of multiple databases, data cubes, or files

• Data transformation
  • Normalization and aggregation

• Data reduction
  • Obtains reduced representation in volume but produces the same or similar analytical results
Data Preparation as a step in the Knowledge Discovery Process

- Data preparation
- Cleaning and Integration
- Selection and Transformation
- Data Mining
- Evaluation and Presentation
- Knowledge
CRISP-DM is a comprehensive data mining methodology and process model that provides anyone—from novices to data mining experts—with a complete blueprint for conducting a data mining project.

A methodology enumerates the steps to reproduce success
CRISP-DM Phases and Tasks

- **Business Understanding**:
  - Determine Business Objectives
  - Assess Situation
  - Determine Data Mining Goals
  - Produce Project Plan

- **Data Understanding**:
  - Collect Initial Data
  - Describe Data
  - Explore Data
  - Verify Data Quality

- **Data Preparation**:
  - Select Data
  - Clean Data
  - Construct Data
  - Integrate Data
  - Format Data

- **Modelling**:
  - Select Modeling Technique
  - Generate Test Design
  - Build Model
  - Assess Model

- **Evaluation**:
  - Evaluate Results
  - Review Process
  - Determine Next Steps

- **Deployment**:
  - Plan Deployment
  - Plan Monitoring & Maintenance
  - Produce Final Report
  - Review Project
CRISP-DM Phases and Tasks

Business Understanding -> Data Understanding -> Data Preparation -> Modelling -> Evaluation -> Deployment

- Business Understanding
  - Determine Business Objectives
  - Assess Situation
  - Determine Data Mining Goals
  - Produce Project Plan

- Data Understanding
  - Collect Initial Data
  - Describe Data
  - Explore Data
  - Verify Data Quality

- Data Preparation
  - Select Data
  - Clean Data
  - Construct Data
  - Integrate Data
  - Format Data

- Modelling
  - Assess Model
  - Plan Deployment
  - Plan Monitoring & Maintenance
  - Produce Final Report
  - Review Project

- Evaluation
  - Review Results
  - Determine Next Steps

- Deployment
CRISP-DM: Data Understanding

- **Collect data**
  - List the datasets acquired (locations, methods used to acquire, problems encountered and solutions achieved).

- **Describe data**
  - Check data volume and examine its gross properties.
  - Accessibility and availability of attributes. Attribute types, range, correlations, the identities.
  - Understand the meaning of each attribute and attribute value in business terms.
  - For each attribute, compute basic statistics (e.g., distribution, average, max, min, standard deviation, variance, mode, skewness).
CRISP-DM: Data Understanding

- Explore data
  - Analyze properties of interesting attributes in detail.
    - Distribution, relations between pairs or small numbers of attributes, properties of significant sub-populations, simple statistical analyses.
- Verify data quality
  - Identify special values and catalogue their meaning.
  - Does it cover all the cases required? Does it contain errors and how common are they?
  - Identify missing attributes and blank fields. Meaning of missing data.
  - Do the meanings of attributes and contained values fit together?
  - Check spelling of values (e.g., same value but sometime beginning with a lower case letter, sometimes with an upper case letter).
  - Check for plausibility of values, e.g. all fields have the same or nearly the same values.
CRISP-DM: Data Preparation

- **Select data**
  - Reconsider data selection criteria.
  - Decide which dataset will be used.
  - Collect appropriate additional data (internal or external).
  - Consider use of sampling techniques.
  - Explain why certain data was included or excluded.

- **Clean data**
  - Correct, remove or ignore noise.
  - Decide how to deal with special values and their meaning (*99 for marital status*).
  - Aggregation level, missing values, etc.
  - Outliers?
CRISP-DM: Data Preparation

- **Construct data**
  - Derived attributes.
  - Background knowledge.
  - How can missing attributes be constructed or imputed?

- **Integrate data**
  - Integrate sources and store result (new tables and records).

- **Format Data**
  - Rearranging attributes  (*Some tools have requirements on the order of the attributes, e.g. first field being a unique identifier for each record or last field being the outcome field the model is to predict.*)
  - Reordering records  (*Perhaps the modelling tool requires that the records be sorted according to the value of the outcome attribute.*)
  - Reformatted within-value  (*These are purely syntactic changes made to satisfy the requirements of the specific modelling tool, remove illegal characters, uppercase lowercase.*)
TYPES OF DATA
Types of Measurements

- Nominal scale
- Categorical scale
- Ordinal scale
- Interval scale
- Ratio scale

More information content

Qualitative

Quantitative

Discrete or Continuous
Types of Measurements: Examples

• Nominal:
  • ID numbers, Names of people

• Categorical:
  • eye color, zip codes

• Ordinal:
  • rankings (e.g., taste of potato chips on a scale from 1-10), grades, height in {tall, medium, short}

• Interval:
  • calendar dates, temperatures in Celsius or Fahrenheit, GRE (Graduate Record Examination) and IQ scores

• Ratio:
  • temperature in Kelvin, length, time, counts
### Types of Measurements: Examples

<table>
<thead>
<tr>
<th>Day</th>
<th>Outlook</th>
<th>Temperature</th>
<th>Humidity</th>
<th>Wind</th>
<th>PlayTennis?</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Sunny</td>
<td>85</td>
<td>85</td>
<td>Light</td>
<td>No</td>
</tr>
<tr>
<td>2</td>
<td>Sunny</td>
<td>80</td>
<td>90</td>
<td>Strong</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>Overcast</td>
<td>83</td>
<td>86</td>
<td>Light</td>
<td>Yes</td>
</tr>
<tr>
<td>4</td>
<td>Rain</td>
<td>70</td>
<td>96</td>
<td>Light</td>
<td>Yes</td>
</tr>
<tr>
<td>5</td>
<td>Rain</td>
<td>68</td>
<td>80</td>
<td>Light</td>
<td>Yes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Day</th>
<th>Outlook</th>
<th>Temperature</th>
<th>Humidity</th>
<th>Wind</th>
<th>PlayTennis?</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>Rain</td>
<td>Sunny</td>
<td>Hot</td>
<td>High</td>
<td>Light</td>
</tr>
<tr>
<td>7</td>
<td>Overcast</td>
<td>Sunny</td>
<td>Hot</td>
<td>High</td>
<td>Strong</td>
</tr>
<tr>
<td>8</td>
<td>Sunny</td>
<td>Overcast</td>
<td>Hot</td>
<td>High</td>
<td>Light</td>
</tr>
<tr>
<td>9</td>
<td>Sunny</td>
<td>Rain</td>
<td>Mild</td>
<td>High</td>
<td>Light</td>
</tr>
<tr>
<td>10</td>
<td>Rain</td>
<td>Rain</td>
<td>Cool</td>
<td>Normal</td>
<td>Light</td>
</tr>
<tr>
<td>11</td>
<td>Sunny</td>
<td>Rain</td>
<td>Cool</td>
<td>Normal</td>
<td>Strong</td>
</tr>
<tr>
<td>12</td>
<td>Overcast</td>
<td>Overcast</td>
<td>Cool</td>
<td>Normal</td>
<td>Strong</td>
</tr>
<tr>
<td>13</td>
<td>Overcast</td>
<td>Sunny</td>
<td>Mild</td>
<td>High</td>
<td>Light</td>
</tr>
<tr>
<td>14</td>
<td>Rain</td>
<td>Sunny</td>
<td>Cool</td>
<td>Normal</td>
<td>Light</td>
</tr>
</tbody>
</table>
Data Conversion

• Some tools can deal with nominal values but other need fields to be numeric

• Convert ordinal fields to numeric to be able to use “>” and “<” comparisons on such fields.
  • A $\rightarrow$ 4.0
  • A- $\rightarrow$ 3.7
  • B+ $\rightarrow$ 3.3
  • B $\rightarrow$ 3.0

• Multi-valued, unordered attributes with small no. of values
  • e.g. Color=Red, Orange, Yellow, ..., Violet
  • for each value $\nu$ create a binary “flag” variable $C_\nu$, which is 1 if Color=$\nu$, 0 otherwise
Conversion: Nominal, Many Values

- **Examples:**
  - US State Code (50 values)
  - Profession Code (7,000 values, but only few frequent)

- Ignore ID-like fields whose values are unique for each record

- For other fields, group values “naturally”:
  - e.g. 50 US States → 3 or 5 regions
  - Profession - select most frequent ones, group the rest

- Create binary flag-fields for selected values
DISCRETIZATION OF CONTINUOUS VARIABLES
Discretization

• Divide the range of a continuous attribute into intervals

  • Some methods require discrete values, e.g. most versions of Naïve Bayes, CHAID
  • Reduce data size by discretization
  • Prepare for further analysis

• Discretization is very useful for generating a summary of data

• Also called “binning”
Equal-width Binning

- It divides the range into $N$ intervals of equal size (range): uniform grid
- If $A$ and $B$ are the lowest and highest values of the attribute, the width of intervals will be: $W = (B - A)/N$.

Temperature values:
64 65 68 69 70 71 72 72 75 75 80 81 83 85

Count
2
2
4
2
0
2
2

[64,67) [67,70) [70,73) [73,76) [76,79) [79,82) [82,85]

Equal Width, bins Low $\leq$ value $< \text{High}$
Equal-width Binning

Disadvantage
(a) Unsupervised
(b) Where does N come from?
(c) Sensitive to outliers

Advantage
(a) simple and easy to implement
(b) produce a reasonable abstraction of data

Salary in a corporation

Count

[0 - 200,000) ... .... [1,800,000 - 2,000,000]
Equal-depth (or height) Binning

- It divides the range into $N$ intervals, each containing approximately the same number of samples.
  - Generally preferred because avoids clumping.
  - In practice, “almost-equal” height binning is used to give more intuitive breakpoints.

- Additional considerations:
  - don’t split frequent values across bins.
  - create separate bins for special values (e.g. 0).
  - readable breakpoints (e.g. round breakpoints).
Equal-depth (or height) Binning

Temperature values:
64 65 68 69 70 71 72 72 75 75 80 81 83 85

Count

[64 .. .. .. .. 69] 4
[70 .. 72] 4
[73 .. .. .. .. .. .. .. .. 81] 4
[83 .. 85] 2

Equal Height = 4, except for the last bin
Discretization considerations

- **Class-independent methods**
  - Equal Width is simpler, good for many classes
    - can fail miserably for unequal distributions
  - Equal Height gives better results
- **Class-dependent methods** can be better for classification
  - Decision tree methods build discretization on the fly
  - Naïve Bayes requires initial discretization
- Many other methods exist ...
Method 1R

- Developed by Holte (1993).
- It is a supervised discretization method using binning.
- After sorting the data, the range of continuous values is divided into a number of disjoint intervals and the boundaries of those intervals are adjusted based on the class labels associated with the values of the feature.
- Each interval should contain a given minimum of instances (6 by default) with the exception of the last one.
- The adjustment of the boundary continues until the next values belongs to a class different to the majority class in the adjacent interval.
1R Example

Interval contains at least 6 elements

Adjustment of the boundary continues until the next values belongs to a class different to the majority class in the adjacent interval.

<table>
<thead>
<tr>
<th>Var</th>
<th>65</th>
<th>78</th>
<th>79</th>
<th>81</th>
<th>82</th>
<th>83</th>
<th>84</th>
<th>84</th>
<th>84</th>
<th>84</th>
<th>84</th>
<th>85</th>
<th>85</th>
<th>85</th>
<th>85</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>majority</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>new class</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

Comment: The original method description does not mention the criterion of making sure that the same value for Var is kept is the same interval (although that seems reasonable).
The results above are given by the method available in the R package Dprep.

See the following papers for more detail:

Very Simple Classification Rules Perform Well on Most Commonly Used Datasets by Robert C. Holte
The Development of Holte’s 1R Classifier by Craig Nevill-Manning, Geoffrey Holmes and Ian H. Witten
Exercise

• Discretize the following values using EW and ED binning

• 13, 15, 16, 16, 19, 20, 21, 22, 22, 25, 30, 33, 35, 35, 36, 40, 45
Entropy Based Discretization

**Class dependent (classification)**

1. Sort examples in increasing order
2. Each value forms an interval ('m' intervals)
3. Calculate the entropy measure of this discretization
4. Find the binary split boundary that minimizes the entropy function over all possible boundaries. The split is selected as a binary discretization.

\[
E(S,T) = \frac{|S_1|}{|S|} \text{Ent}(S_1) + \frac{|S_2|}{|S|} \text{Ent}(S_2)
\]

5. Apply the process recursively until some stopping criterion is met, e.g.,

\[
\text{Ent}(S) - E(T,S) > \delta
\]
Entropy

\[ Ent = - \sum_{c=1}^{N} p_c \cdot \log_2 p_c \]

<table>
<thead>
<tr>
<th>p</th>
<th>1-p</th>
<th>Ent</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2</td>
<td>0.8</td>
<td>0.72</td>
</tr>
<tr>
<td>0.4</td>
<td>0.6</td>
<td>0.97</td>
</tr>
<tr>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
</tr>
<tr>
<td>0.6</td>
<td>0.4</td>
<td>0.97</td>
</tr>
<tr>
<td>0.8</td>
<td>0.2</td>
<td>0.72</td>
</tr>
</tbody>
</table>

\[ \log_2(2) \]

\[ \log_2(3) \]
Entropy/Impurity

- $S$ - training set, $C_1, ..., C_N$ classes
- Entropy $E(S)$ - measure of the impurity in a group of examples

- $p_c$ - proportion of $C_c$ in $S$

\[
\text{Impurity}(S) = -\sum_{c=1}^{N} p_c \cdot \log_2 p_c
\]
Impurity

Very impure group

Less impure

Minimum impurity

high entropy

null entropy
An example of entropy disc.

Test split temp < 71.5

<table>
<thead>
<tr>
<th>Temp.</th>
<th>Play?</th>
</tr>
</thead>
<tbody>
<tr>
<td>64</td>
<td>Yes</td>
</tr>
<tr>
<td>65</td>
<td>No</td>
</tr>
<tr>
<td>68</td>
<td>Yes</td>
</tr>
<tr>
<td>69</td>
<td>Yes</td>
</tr>
<tr>
<td>70</td>
<td>Yes</td>
</tr>
<tr>
<td>71</td>
<td>No</td>
</tr>
<tr>
<td>72</td>
<td>No</td>
</tr>
<tr>
<td>72</td>
<td>Yes</td>
</tr>
<tr>
<td>75</td>
<td>Yes</td>
</tr>
<tr>
<td>75</td>
<td>Yes</td>
</tr>
<tr>
<td>80</td>
<td>No</td>
</tr>
<tr>
<td>81</td>
<td>Yes</td>
</tr>
<tr>
<td>83</td>
<td>Yes</td>
</tr>
<tr>
<td>85</td>
<td>No</td>
</tr>
</tbody>
</table>

(4 yes, 2 no)

\[
\text{Ent(split 71.5)} = \frac{6}{14} \cdot \left( \frac{4}{6} \log\frac{4}{6} + \frac{2}{6} \log\frac{2}{6} \right) + \frac{8}{14} \cdot \left( \frac{5}{8} \log\frac{5}{8} + \frac{3}{8} \log\frac{3}{8} \right) = 0.939
\]

<table>
<thead>
<tr>
<th></th>
<th>yes</th>
<th>no</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt; 71.5</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>&gt; 71.5</td>
<td>5</td>
<td>3</td>
</tr>
</tbody>
</table>

(5 yes, 3 no)

<table>
<thead>
<tr>
<th></th>
<th>yes</th>
<th>no</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt; 77</td>
<td>7</td>
<td>3</td>
</tr>
<tr>
<td>&gt; 77</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

\[
\text{Ent(split 77)} = \frac{10}{14} \cdot \left( \frac{7}{10} \log\frac{7}{10} + \frac{3}{10} \log\frac{3}{10} \right) + \frac{4}{14} \cdot \left( \frac{2}{4} \log\frac{2}{4} + \frac{2}{4} \log\frac{2}{4} \right) = 0.915
\]
An example (cont.)

The method tests all split possibilities and chooses the split with smallest entropy.

In the first iteration a split at 84 is chosen.

The two resulting branches are processed recursively.

The fact that recursion only occurs in the first interval in this example is an artifact. In general both intervals have to be split.

<table>
<thead>
<tr>
<th>Temp.</th>
<th>Play?</th>
</tr>
</thead>
<tbody>
<tr>
<td>64</td>
<td>Yes</td>
</tr>
<tr>
<td>65</td>
<td>No</td>
</tr>
<tr>
<td>68</td>
<td>Yes</td>
</tr>
<tr>
<td>69</td>
<td>Yes</td>
</tr>
<tr>
<td>70</td>
<td>Yes</td>
</tr>
<tr>
<td>71</td>
<td>No</td>
</tr>
<tr>
<td>72</td>
<td>No</td>
</tr>
<tr>
<td>72</td>
<td>Yes</td>
</tr>
<tr>
<td>75</td>
<td>Yes</td>
</tr>
<tr>
<td>75</td>
<td>Yes</td>
</tr>
<tr>
<td>80</td>
<td>No</td>
</tr>
<tr>
<td>81</td>
<td>Yes</td>
</tr>
<tr>
<td>83</td>
<td>Yes</td>
</tr>
<tr>
<td>85</td>
<td>No</td>
</tr>
</tbody>
</table>
The stopping criterion

Previous slide did not take into account the stopping criterion.

\[ \text{Ent}(S) - E(T, S) > \delta \]

\[ \delta > \frac{\log(N-1)}{N} + \frac{\Delta(T, S)}{N} \]

\[ \Delta(T, S) = \log_2(3^c - 2) - [c\text{Ent}(S) - c_1\text{Ent}(S_1) - c_2\text{Ent}(S_2)] \]

- \( c \) is the number of classes in \( S \)
- \( c_1 \) is the number of classes in \( S_1 \)
- \( c_2 \) is the number of classes in \( S_2 \).

This is called the Minimum Description Length Principle (MDLP)
Exercise

- Compute the gain of splitting this data in half

<table>
<thead>
<tr>
<th>Humidity</th>
<th>play</th>
</tr>
</thead>
<tbody>
<tr>
<td>65</td>
<td>Yes</td>
</tr>
<tr>
<td>70</td>
<td>No</td>
</tr>
<tr>
<td>70</td>
<td>Yes</td>
</tr>
<tr>
<td>70</td>
<td>Yes</td>
</tr>
<tr>
<td>70</td>
<td>Yes</td>
</tr>
<tr>
<td>75</td>
<td>Yes</td>
</tr>
<tr>
<td>80</td>
<td>Yes</td>
</tr>
<tr>
<td>80</td>
<td>Yes</td>
</tr>
<tr>
<td>85</td>
<td>No</td>
</tr>
<tr>
<td>86</td>
<td>Yes</td>
</tr>
<tr>
<td>90</td>
<td>No</td>
</tr>
<tr>
<td>90</td>
<td>Yes</td>
</tr>
<tr>
<td>91</td>
<td>No</td>
</tr>
<tr>
<td>95</td>
<td>No</td>
</tr>
<tr>
<td>96</td>
<td>Yes</td>
</tr>
</tbody>
</table>
OUTLIERS
Outliers

- Outliers are values thought to be out of range.
  - “An outlier is an observation that deviates so much from other observations as to arouse suspicion that it was generated by a different mechanism”
  - Can be detected by standardizing observations and label the standardized values outside a predetermined bound as outliers.
  - Outlier detection can be used for fraud detection or data cleaning.

- Approaches:
  - do nothing
  - enforce upper and lower bounds
  - let binning handle the problem
Outlier detection

• Univariate

  • Compute mean and std. deviation. For $k=2$ or $3$, $x$ is an outlier if outside limits (normal distribution assumed)

  $$(\bar{x} - ks, \bar{x} + ks)$$
Outlier detection

- **Univariate**
  
  - **Boxplot:** An observation is an extreme outlier if
    
    $$(Q1-3\times IQR, Q3+3\times IQR), \text{ where } IQR=Q3-Q1$$
    
    ($IQR = \text{Inter Quartile Range}$)

    and declared a mild outlier if it lies outside of the interval

    $$(Q1-1.5\times IQR, Q3+1.5\times IQR).$$

   ![Boxplot diagram](http://www.physics.csbsju.edu/stats/box2.html)
Outlier detection

- Multivariate
  - Clustering
    - Very small clusters are outliers

Outlier detection

- Multivariate
  - Distance based
    - An instance with very few neighbors within D is regarded as an outlier

Knn algorithm
A bi-dimensional outlier that is not an outlier in either of its projections.
Recommended reading

Only with hard work and a favorable context you will have the chance to become an outlier!!!
DATA TRANSFORMATION
Normalization

- For distance-based methods, normalization helps to prevent that attributes with large ranges out-weight attributes with small ranges

  - min-max normalization
  - z-score normalization
  - normalization by decimal scaling
Normalization

• min-max normalization

\[ v' = \frac{v - \min_v}{\max_v - \min_v} (\text{new}\_\text{max}_v - \text{new}\_\text{min}_v) + \text{new}\_\text{min}_v \]

• z-score normalization

\[ v' = \frac{v - \bar{v}}{\sigma_v} \]

It does not eliminate outliers.

• normalization by decimal scaling

\[ v' = \frac{v}{10^j} \]

Where \( j \) is the smallest integer such that \( \text{Max}(|v'|) < 1 \)

range: -986 to 917 => j=3
-986 -> -0.986
917 -> 0.917
<table>
<thead>
<tr>
<th>Age</th>
<th>min-max (0-1)</th>
<th>z-score</th>
<th>dec. scaling</th>
</tr>
</thead>
<tbody>
<tr>
<td>44</td>
<td>0.421</td>
<td>0.450</td>
<td>0.44</td>
</tr>
<tr>
<td>35</td>
<td>0.184</td>
<td>-0.450</td>
<td>0.35</td>
</tr>
<tr>
<td>34</td>
<td>0.158</td>
<td>-0.550</td>
<td>0.34</td>
</tr>
<tr>
<td>34</td>
<td>0.158</td>
<td>-0.550</td>
<td>0.34</td>
</tr>
<tr>
<td>39</td>
<td>0.289</td>
<td>-0.050</td>
<td>0.39</td>
</tr>
<tr>
<td>41</td>
<td>0.342</td>
<td>0.150</td>
<td>0.41</td>
</tr>
<tr>
<td>42</td>
<td>0.368</td>
<td>0.250</td>
<td>0.42</td>
</tr>
<tr>
<td>31</td>
<td>0.079</td>
<td>-0.849</td>
<td>0.31</td>
</tr>
<tr>
<td>28</td>
<td>0.000</td>
<td>-1.149</td>
<td>0.28</td>
</tr>
<tr>
<td>30</td>
<td>0.053</td>
<td>-0.949</td>
<td>0.3</td>
</tr>
<tr>
<td>38</td>
<td>0.263</td>
<td>-0.150</td>
<td>0.38</td>
</tr>
<tr>
<td>36</td>
<td>0.211</td>
<td>-0.350</td>
<td>0.36</td>
</tr>
<tr>
<td>42</td>
<td>0.368</td>
<td>0.250</td>
<td>0.42</td>
</tr>
<tr>
<td>35</td>
<td>0.184</td>
<td>-0.450</td>
<td>0.35</td>
</tr>
<tr>
<td>33</td>
<td>0.132</td>
<td>-0.649</td>
<td>0.33</td>
</tr>
<tr>
<td>45</td>
<td>0.447</td>
<td>0.550</td>
<td>0.45</td>
</tr>
<tr>
<td>34</td>
<td>0.158</td>
<td>-0.550</td>
<td>0.34</td>
</tr>
<tr>
<td>65</td>
<td>0.974</td>
<td>2.548</td>
<td>0.65</td>
</tr>
<tr>
<td>66</td>
<td>1.000</td>
<td>2.648</td>
<td>0.66</td>
</tr>
<tr>
<td>38</td>
<td>0.263</td>
<td>-0.150</td>
<td>0.38</td>
</tr>
</tbody>
</table>

28 minimum
66 maximum
39.50 average
10.01 standard deviation
MISSING DATA
Missing Data

• Data is not always available
  • E.g., many tuples have no recorded value for several attributes, such as customer income in sales data

• Missing data may be due to
  • equipment malfunction
  • inconsistent with other recorded data and thus deleted
  • data not entered due to misunderstanding
  • certain data may not be considered important at the time of entry
  • not register history or changes of the data

• Missing data may need to be inferred.

• Missing values may carry some information content: e.g. a credit application may carry information by noting which field the applicant did not complete
Missing Values

• There are always MVs in a real dataset

• **MV**s may have an impact on modelling, in fact, they can destroy it!

• Some tools ignore missing values, others use some metric to fill in replacements
  • The modeller should avoid default automated replacement techniques
    • Difficult to know limitations, problems and introduced bias

• Replacing missing values without elsewhere capturing that information removes information from the dataset
How to Handle Missing Data?

- **Ignore records** (use only cases with all values)
  - Usually done when class label is missing as most prediction methods do not handle missing data well
  - Not effective when the percentage of missing values per attribute varies considerably as it can lead to insufficient and/or biased sample sizes

- **Ignore attributes with missing values**
  - Use only features (attributes) with all values (may leave out important features)

- **Fill in the missing value manually**
  - tedious + infeasible?
How to Handle Missing Data?

• Use a global constant to fill in the missing value
  • e.g., “unknown”. (May create a new class!)

• Use the attribute mean to fill in the missing value
  • It will do the least harm to the mean of existing data
  • If the mean is to be unbiased
  • What if the standard deviation is to be unbiased?

• Use the attribute mean for all samples belonging to the same class to fill in the missing value
How to Handle Missing Data?

• Use the most probable value to fill in the missing value
  • Inference-based such as Bayesian formula or decision tree

• Identify relationships among variables
  • Linear regression, Multiple linear regression, Nonlinear regression

• Nearest-Neighbour estimator
  • Finding the k neighbours nearest to the point and fill in the most frequent value or the average value
  • Finding neighbours in a large dataset may be slow
Nearest-Neighbour
How to Handle Missing Data?

• Note that, it is as important to avoid adding bias and distortion to the data as it is to make the information available.
  
  • bias is added when a wrong value is filled-in

• No matter what techniques you use to conquer the problem, it comes at a price. The more guessing you have to do, the further away from the real data the database becomes. Thus, in turn, it can affect the accuracy and validation of the mining results.
HANDLING REDUNDANCY
Handling Redundancy in Data Integration

• Redundant data occur often when integrating databases
  • The same attribute may have different names in different databases
  • False predictors are fields correlated to target behavior, which describe events that happen at the same time or after the target behavior
    • Example: Service cancellation date is a leaker when predicting attriters

• One attribute may be a “derived” attribute in another table, e.g., annual revenue

• For numerical attributes, redundancy may be detected by correlation analysis

\[ r_{xy} = \frac{\frac{1}{N-1} \sum_{n=1}^{N} (x_n - \bar{x})(y_n - \bar{y})}{\sqrt{\frac{1}{N-1} \sum_{n=1}^{N} (x_n - \bar{x})^2 \cdot \frac{1}{N-1} \sum_{n=1}^{N} (y_n - \bar{y})^2}} \]  
\(-1 \leq r_{xy} \leq 1\)
Scatter Matrix
SAMPLING AND UNBALANCED DATASETS
Sampling

• The cost of sampling is proportional to the sample size and not to the original dataset size, therefore, a mining algorithm’s complexity is potentially sub-linear to the size of the data

• Choose a representative subset of the data

  • Simple random sampling (SRS) (with or without reposition)
  • Stratified sampling:
    • Approximate the percentage of each class (or subpopulation of interest) in the overall database
    • Used in conjunction with skewed data
Unbalanced Target Distribution

- Sometimes, classes have very unequal frequency
  - Attrition prediction: 97% stay, 3% attrite (in a month)
  - medical diagnosis: 90% healthy, 10% disease
  - eCommer: 99% don't buy, 1% buy
  - Security: >99.99% of Americans are not terrorists
- Similar situation with multiple classes
- Majority class classifier can be 97% correct, but useless
Handling Unbalanced Data

- With two classes: let positive targets be a minority
- Separate raw held-aside set (e.g. 30% of data) and raw train
  - put aside raw held-aside and don’t use it till the final model
- Select remaining positive targets (e.g. 70% of all targets) from raw train
- Join with equal number of negative targets from raw train, and randomly sort it
- Separate randomized balanced set into balanced train and balanced test
Building Balanced Train Sets

- **Targets**: Raw set for estimating accuracy of final model
- **Non-Targets**: Balanced set
- **Raw Held**: 70/30
- **SRS**: Same % of Y and N

Balanced Train
Balanced Test
Summary

• Every real world data set needs some kind of data pre-processing
  • Deal with missing values
  • Correct erroneous values
  • Select relevant attributes
  • Adapt data set format to the software tool to be used

• In general, data pre-processing consumes more than 60% of a data mining project effort
References

• ‘Data preparation for data mining’, Dorian Pyle, 1999

• ‘Data Mining: Concepts and Techniques’, Jiawei Han and Micheline Kamber, 2000


• DM: Introduction: Machine Learning and Data Mining, Gregory Piatetsky-Shapiro and Gary Parker
  (http://www.kdnuggets.com/data_mining_course/dm1-introduction-ml-data-mining.ppt)

• ESMA 6835 Minería de Datos (http://math.uprm.edu/~edgar/dm8.ppt)