

Chapter IV: Preprocessing

Knowledge Discovery in Databases

Luciano Melodia M.A.

Evolutionary Data Management, Friedrich-Alexander University Erlangen-Nürnberg

Summer semester 2021



Chapter IV: Preprocessing

This is our agenda for this lecture:

Data preprocessing: an overview.

Data quality.

Major tasks in data preprocessing.

Data cleaning.

Data integration.

Data reduction.

Data transformation and data discretization.

Summary.

Data quality: why preprocess the data?

This is our agenda for this lecture:

Measures for data quality: A multidimensional view:

Accuracy: correct or wrong, accurate or not.

Completeness: not recorded, unavailable.

Consistency: some modified but some not, dangling refs, etc.

Timeliness: timely updated?

Believability: how trustworthy is it, that the data is correct?

Interpretability: how easily can the data be understood?

And even many more!

Major tasks in data preprocessing

Data cleaning:

- Fill in missing values.
- Smooth noisy data.
- Identify or remove outliers.
- Resolve inconsistencies.

Data integration:

- Integration of multiple databases.
- Data cubes or files.

Data reduction:

- Dimensionality reduction.
- Numerosity reduction.
- Data compression.

Data transformation and data discretization:

- Normalization.
- Concept-hierarchy generation.

Chapter IV: Preprocessing

Data preprocessing: an overview.

Data quality.

Major tasks in data preprocessing.

Data cleaning.

Data integration.

Data reduction.

Data transformation and data discretization.

Summary.

Data cleaning

Data in the real world is **dirty**. Lots of potentially incorrect data:

E.g. instrument faulty, human or computer error, transmission error.

Incomplete: lacking attributes, lacking certain attributes of interest or containing aggregate data.

E.g. occupation = "" (missing data).

Noisy: containing noise, errors or outliers.

Stochastic deviation, imprecision.

E.g. measurements.

Inconsistencies: containing discrepancies in codes or names.

E.g. age = "42", birthday = "03/07/2010".

Was rating "1,2,3" and now it is "A,B,C".

Discrepancy between duplicate records (e.g. address old and new).

Intentional (only default value, e.g. disguised missing data):

Jan. 1 as everyone's birthday?

Incomplete (missing) data

Data is not always available.

E.g. many tuples have no recorded value for several attributes.

Examples are customer income in sales data.

Missing data may be due to:

Equipment malfunction.

Inconsistency 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 registered history or changes of the data.

Missing data may need to be inferred.

How to handle missing data?

Ignore the tuple:

Usually done when class label is missing (when doing classification).

Not effective when the percentage of missing values per attribute varies considerably.

Fill in the missing value manually.

Tedious or infeasible.

Fill in automatically with:

A global constant, e.g. "unknown", maybe a new class.

The attribute mean.

The attribute mean for all samples belonging to the same class.

The most probable value: Inference-based such as Bayesian formula or decision tree.

Noisy data?

Noise:

- Random error or variance in a measured variable.

- Stored value a little bit off the real value, up or down.

- Leads to (slightly) incorrect attribute values.

May be due to:

- Faulty or imprecise data-collection instruments.

- Data-entry problems.

- Data-transmission problems.

- Technology limitation.

- Inconsistency in naming conventions.

How to handle noisy data?

Beginning:

First sort data and partition into (equal-frequency) bins.

Then smooth by bin mean, by bin median or by bin boundaries.

Regression:

Smooth by fitting the data to regression functions.

Clustering:

Detect and remove outliers.

Combined computer and human inspection:

Detect suspicious values and check by human.

E.g. deal with possible outliers.

Data cleaning as a process

Data-discrepancy detection:

Use **metadata** (e.g. domain, range, dependency, distribution).

Check field overloading.

Check uniqueness rule, consecutive rule and null rule.

Use commercial tools:

Data scrubbing: use simple domain knowledge (e.g. postal code, spell-check) to detect errors and make corrections.

Data auditing: by analyzing data to discover rules and relationships to detect violators (e.g. correlation and clustering to find outliers).

Data migration and integration:


Data-migration tools: allow transformations to be specified.

ETL (Extraction/Transformation/Loading) tools: allow users to specify transformations through a graphical user interface.

Integration of the two processes.

Iterative and interactive (e.g. the Potter's Wheel tool).

Thank you for your attention.
Any questions about the forth chapter?

Ask them now, or again, drop me a line:
 `luciano.melodia@fau.de`.