Working with Tabular Data

PU Tools, Ressourcen, Infrastruktur

Nils Reiter,
nils.reiter@uni-koeln.de

November 26, 2020
(Winter term 2020/21)

Motivation

- Machine learning (ML): Pattern detection in data
- Classification: Items $\rightarrow$ classes
- Features used to describe items

Motivation

- Machine learning (ML): Pattern detection in data
- Classification: Items $\rightarrow$ classes
- Features used to describe items
- ML 'learns' relations between feature combinations and target classes

Features

- Used to describe classification items
- Feature extraction: Code to determine feature values for an item
- Features encode expected influence of item properties and target class
  - If we think a property could be relevant → make it a feature

Example

- Task: Assign part of speech information to words in context
  - "The dog barks." → (Det, Noun, Verb, Punct)
- Target class: Parts of speech (noun, verb, adjective, ...)

Features

- Used to describe classification items
- Feature extraction: Code to determine feature values for an item
- Features encode expected influence of item properties and target class
  - If we think a property could be relevant → make it a feature

Example

- Task: Assign part of speech information to words in context
  - "The dog barks." → (Det, Noun, Verb, Punct)
- Target class: Parts of speech (noun, verb, adjective, ...)
- Features
  - Case (upper vs. lower)
  - Length
  - Suffix (last two characters)

Features

Data Types

Feature	Type
Case	
Length	
Suffix	

Features

Data Types

Feature	Type
Case	Three categories: upper/lower/other
Length	Integer
Suffix	String

Features

Feature Values

Word	Case	Length	Suffix	Class
The	upper	3	he	Det
dog	lower	3	og	Noun
barks	lower	5	ks	Verb
.	other	1	.	Punct

Table: Extracted features for example sentence, plus target class annotation

- This will be the input to the machine learning algorithm

Tables

- Tables are the backbone of quantitative analysis
- Convention: Items in rows, properties/features in columns

Tables

- Tables are the backbone of quantitative analysis
- Convention: Items in rows, properties/features in columns
- Main data types: Numbers, categories
  - If all entries are numeric, it's a (mathematical) matrix
- Various file formats
  - CSV/TSV: Comma/tab-separated values
  - XLS/XLSX: Excel format
  - ARFF: Weka file format (= CSV + type declarations)

Comma-Separated Values (CSV)

```
1 The,upper,3,he,Det
2 dog,lower,3,og,Noun
3 barks,lower,5,ks,Verb
4 .,other,1,.,Punct
```

Comma-Separated Values (CSV)

```
1 The,upper,3,he,Det
2 dog,lower,3,og,Noun
3 barks,lower,5,ks,Verb
4 .,other,1,.,Punct
```

- Items separated by newline, feature values by comma
- Problems?

Comma-Separated Values (CSV)

```
1 The,upper,3,he,Det
2 dog,lower,3,og,Noun
3 barks,lower,5,ks,Verb
4 .,other,1,.,Punct
```

- Items separated by newline, feature values by comma
- Problems? What if the sentence contains a comma?

Comma-Separated Values (CSV)

```
1 The,upper,3,he,Det
2 dog,lower,3,og,Noun
3 barks,lower,5,ks,Verb
4 .,other,1,.,Punct
```

- Items separated by newline, feature values by comma
- Problems? What if the sentence contains a comma?
  - Escaping: Use special characters without their special meaning: `\\,`

Comma-Separated Values (CSV)

```
1 The,upper,3,he,Det
2 dog,lower,3,og,Noun
3 barks,lower,5,ks,Verb
4 .,other,1,.,Punct
```

- Items separated by newline, feature values by comma
- Problems? What if the sentence contains a comma?
  - Escaping: Use special characters without their special meaning: `\\,`
  - Quoting: Enclose them in quote characters `","`

Comma-Separated Values (CSV)

```
1 The,upper,3,he,Det
2 dog,lower,3,og,Noun
3 barks,lower,5,ks,Verb
4 .,other,1,.,Punct
```

- Items separated by newline, feature values by comma
- Problems? What if the sentence contains a comma?
  - Escaping: Use special characters without their special meaning: `\\,`
  - Quoting: Enclose them in quote characters `","`
- Different strategies, all are used

Tab-Separated Values (TSV)

Listing 1: A TSV representation, with tabs represented as →

```
1 The→upper→3→he→Det
2 dog→lower→3→og→Noun
3 barks→lower→5→ks→Verb
4 .→other→1→.→Punct
```

- Similar to CSV, but with a tab instead of a comma
- Tab character: A single character with variable width
  - Often used for indentation
- Can be escaped with `\t` (e.g., in regular expressions)

Tab-Separated Values (TSV)

Listing 2: A TSV representation, with tabs represented as →

```
1 The→upper→3→he→Det
2 dog→lower→3→og→Noun
3 barks→lower→5→ks→Verb
4 .→other→1→.→Punct
```

- Similar to CSV, but with a tab instead of a comma
- Tab character: A single character with variable width
  - Often used for indentation
- Can be escaped with `\t` (e.g., in regular expressions)
- CSV/TSV have undefined 'edge cases'
  - Escaping, quoting, comments
  - Inspect before processing

CSV/TSV Tools

- Most spreadsheets programs can import and export CSV/TSV (MS Excel, Apple Numbers, Google Spreadsheets, OpenOffice Calc)

CSV/TSV Tools

- Most spreadsheets programs can import and export CSV/TSV (MS Excel, Apple Numbers, Google Spreadsheets, OpenOffice Calc)

Reading/writing CSV

- Java: Apache Commons CSV `https://commons.apache.org/proper/commons-csv/`
- Python: Module in standard library `https://docs.python.org/3/library/csv.html`
- Command line
  - csvkit `https://csvkit.readthedocs.io/en/latest/`
  - awk/gawk `https://www.gnu.org/software/gawk/manual/gawk.html`

XLS/XLSX

- File format used by MS Excel
- Binary, closed
- Don't use Excel as a database: `https://www.youtube.com/watch?v=zUp8pkoeMss`
- Useful for lightweight calculation/visualisation
- Difficult to integrate with other tools

ARFF

- Used by machine learning toolkit Weka
- Data as CSV
- Header to define attributes/features
- Name/type for each attribute
  - Nominal values: Possible values

Exercise 4