Data Analysis & Visualisation

Raoul Grouls, 6 September 2024

Hello world

- Raoul Grouls, data alchemist, https://the-pttrn.nl/
- Research @ HAN, ‘internal representations of the world in AI models’
- Teaching @ HAN, HU

raoul.grouls@han.nl
https://github.com/raoulg

Leerdoelen les 1

De student begrijpt:

- Gestalt principles: proximity, similarity, enclosure, closure, continuity, connection
- preattentive processing (first 200ms)
- five guidelines: show the data / reduce clutter / integrate text / avoid spaghetti / start with grey
- Wat de motivatie is achter een virtual environment

De student kan:

- de gestalt principes & five guidelines toepassen op visualisaties
- een virtual environment activeren met uv
- nieuwe features extraheren met behulp van regular expressions
- Een script op de VM vanaf de terminal opstarten
- click gebruiken voor command line arguments bij een script
- begrijpt de opzet van een project (src folder, data/raw en data/processed, pyproject.toml, notebooks) en kan dit zelf opzetten
- kan een eigen git-repo maken
- Regular expressions toepassen:
  - start ^
  - end $
  - or: [Bb]
  - ranges [a-zA-Z]
  - any char .
  - zero or more a*
  - one or more a+
  - not in range [^a-z]
  - shortcuts (\w, \s, \d)
  - lookbehind (?<=...)
  - lookahead (?=…)

Code
Wiskunde
Communicatie

Business probleem
Wiskundig probleem
Implementatie probleem
Communicatie probleem
Data science project

Homo Fabulans

The first technology

Stateful communication

The mathematical theory of communication, Shannon [3]

The transducer (encoder/decoder) can be described by two functions:

$$y_n = f(x_n, \alpha_n)$$

$$\alpha_{n+1} = g(x_n, \alpha_n)$$

Where x_n is the nth input, α_n is the state of the transducer, and y_m is the output signal.

A single bit of information lands differently, depending on state α

Homomorphism

- A homomorphism is a mapping that preserves algebraic structure between domain and range
- $\phi(x \star y) = \phi(x) \star \phi(y), \forall x, y$

Homomorphism

- For example, lets take
  - $\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$
  - $\mathbb{Z}_3 = \{0, 1, 2\}$, the set of integers modulo 3
  - $\phi : \mathbb{Z} \to \mathbb{Z}_3,\ \phi(x) = x \bmod 3$
  - $\phi(5) + \phi(7) = \phi(5 + 7) = 0$
- ϕ is a homomorphism between $\mathbb{Z}$ and $\mathbb{Z}_3$ for addition because the structure that matters is preserved.

Reading is a homomorphism between ideas

Laws of physics are homomorphic to the structure of the real world

Some maps are less useful…

…than others

Model of the Earth, Leonardo Da Vinci

FIG. 2-7. Geometric model of the Earth.
Codex Leicester, folio 35v (detail).

Models don’t need to be “correct” to be useful or insightful.
Just homomorphic enough

rollover for more detail

What Makes a Good Visualization?

explicit (implicit)

David McCandless
InformationisBeautiful.net

taken from new book
Knowledge is Beautiful

find out more
bit.ly/KIB_Books

Gestalt principles

Preattentive processing

Five guidelines

0. Avoid spurious correlations
1. Show the data
2. Reduce the clutter
3. Integrate graphics and text
4. Avoid the spaghetti chart
5. Start with grey

The world is awful
The world is much better
The world can be much better
All three statements are true at the same time

The world is awful

Our World in Data

Globally 4.3% of all children die before they are 15 years old.
This means every day 16,000 children die.

Data: 2020 data from UN IGME.

OurWorldinData.org – Research and data to make progress against the world's largest problems.

Licensed under CC-BY by the author Max Roser

The world is much better

Our World in Data

Data: 2020 data from UN IGME and historical data from Volk and Atkinson (2013).

OurWorldinData.org – Research and data to make progress against the world's largest problems.

Licensed under CC-BY by the author Max Roser

The world can be much better

Our World in Data

Data: 2020 data from UN IGME.

Licensed under CC-BY by the author Max Roser

OurWorldinData.org – Research and data to make progress against the world's largest problems.

The distance between Saturn and the Sun

correlates with

Google searches for 'how to make baby'

The average distance between Saturn and the Sun as measured on the first day of each month · Source: Caclculated using Astropy

Relative volume of Google searches for 'how to make baby' (Worldwide), with quotes) · Source: Google Trends

2004-2023, $r=0.967$, $r^2=0.935$, $p<0.01$ · tylervigen.com/spurious/correlation/1522

Bericht lengtes van de hobbies zijn gemiddelde langer
Avg message length: 3.00
Log Message Length
8
7
6
5
4
3
2
1
other
fishing
football
formula 1
mixed
Category

Authors' Distinct Messaging Styles Unveiled

Each author's messaging habits are distinct, showcased through the peaks and patterns in message length

Source: Family WhatsApp Group Analysis (2016-2024)

War dominating in lengthy messages
a12
klimaat
oorlog
politiek
Frequency
0.0
0.2
0.4
0.6
0.8
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Message Length (log10 scale)

Opleving eten en drinken emoticons na opheffen lockdown Corona
Food & Drink emoticons
Trendlijn
Restaurants, concertzalen en theaters mogen weer open
Kans op eten of drinken emoticon in bericht
0.0175
0.0150
0.0125
0.0100
0.0075
0.0050
0.0025
0.0000
2021-04
2021-07
2021-10
2022-01
2022-04
2022-07
2022-10
2023-01
2023-04
2023-07
2023-10
2024-01

Import Ouddorp spreekt echt een andere taal
Niet uit Ouddorp
Oorspronkelijke Ouddorper
15
10
5
0
−5
−10
−15
−10
−5
0
5
10
15

Regular Expressions

WHENEVER I LEARN A NEW SKILL I CONCOCT ELABORATE FANTASY SCENARIOS WHERE IT LETS ME SAVE THE DAY.
OH NO! THE KILLER MUST HAVE FOLLOWED HER ON VACATION!
BUT TO FIND THEM WE'D HAVE TO SEARCH THROUGH 200 MB OF EMAILS LOOKING FOR SOMETHING FORMATTED LIKE AN ADDRESS!
IT'S HOPELESS!
EVERYBODY STAND BACK.
I KNOW REGULAR EXPRESSIONS.
PERL!
TAP TAP

Introduction regular expressions

- Regular expressions (regex) are a powerful tool for matching patterns in text.
- Used for searching, editing, or manipulating text and data.
- Use https://regex101.com/ to develop and test your regexes
- LLMs are pretty good at creating and explaining regexes.
- https://www.alf.nu/RegexGolf if you like some puzzles

Basic symbols

- **^ (Start):** Matches the start of a line.
- **$ (End):** Matches the end of a line.
- **. (Any Char):** Matches any character except a newline.

- Example: To match any line that starts with "A", we use `^A`

The OR operator and Ranges

- **[Bb] (Or):** Matches either "B" or "b".
- **[a-zA-Z] (Ranges):** Matches any letter, regardless of case.
- **[0-9]:** matches any number from 0 to 9

Example:

- To find any line that starts with a lowercase letter or number, use `^[a-z0-9]`

Quantifiers

- **a* (Zero or More):** Matches zero or more occurrences of "a".
- **a+ (One or More):** Matches one or more occurrences of “a".
- **a{3} (Exactly Three):** Matches exactly three occurrences of "a".
- **a{2,5} (Two to Five):** Matches between two and five occurrences of “a".

- **Example**: Combined with ranges: `[a-z]+`
- To match a string that contains four to six a’s in a row, use `a{4,6}`

Negation and Shortcuts

- **[^a-z] (Not in Range):** Matches any character not in the range "a" to "z".
- Shortcuts:
  - **\w (Word Char):** Matches any word character (letter, number, underscore).
  - **\s (Whitespace):** Matches any whitespace character (space, tab, newline).
  - **\d (Digit):** Matches any digit.

- **Question:** How to find lines <u>not</u> starting with any lowercase letter?

Lookaround

- **Lookahead (?=...):** Matches a group after the main expression without including it in the result.
- **Lookbehind (?<=...):** Matches a group before the main expression without including it in the result.
- **Example:** To find words directly after a colon without including the colon, use `(?<=:)\s*\w+`
- Question: what does the \s* do?

Combining everything

Message format:

- [11-01-2022 08:05:25] ~Raoul: Ah
- Can you explain what the other formats should be, as defined by the regexes?

```
iosRegexes = BaseRegexes(
    timestamp=r"(?<=\[)\d{2}-\d{2}-\d{4} \d{2}:\d{2}:\d{2}(?=\])",
    author=r"(?<=\]\s)(.*?)(?=:)",
    clear=r"\[\d{2}-\d{2}-\d{4} \d{2}:\d{2}:\d{2}\]\s[~a-zA-Z\s]+:",
    fmt="%d-%m-%Y %H:%M:%S",
)


androidRegexes = BaseRegexes(
    timestamp=r"^\d{2}-\d{2}-\d{4} \d{2}:\d{2}",
    author=r"(?<=\s-\s)(.*?)(?=:)",
    clear=r"^\d{2}-\d{2}-\d{4} \d{2}:\d{2}[-~a-zA-Z\s]+:",
    fmt="%d-%m-%Y %H:%M",
)

oldRegexes = BaseRegexes(
    timestamp=r"^\d{1,2}/\d{1,2}/\d{2}, \d{2}:\d{2}",
    author=r"(?<=\s-\s)(.*?)(?=:)",
    clear=r"^\d{1,2}/\d{1,2}/\d{2}, \d{2}:\d{2}[-~a-zA-Z0-9\s]+:",
    fmt="%m/%d/%y, %H:%M",
)
```

Improving with chatGPT

[…] this is the dataformat, […] this the regex, can you improve it?

\[([0-2]?\d|3[01])-(0?\d|1[012])-(\d{4}) ([01]?\d|2[0-3]):([0-5]?\d):([0-5]?\d)\]

This regex breaks down as follows:

- `([0-2]?\d|3[01])`: Matches days from `1-31`, allowing for a leading zero.
- `(0?\d|1[012])`: Matches months from `1-12`, also allowing for a leading zero.
- `(\d{4})`: Matches any four digits for the year.
- `([01]?\d|2[0-3])`: Matches hours from `0-23`, with optional leading zero.
- `([0-5]?\d)`: Matches minutes and seconds from `0-59`, with optional leading zero.