University of Essex

Regular expressions in Python

Akitaka Matsuo
Essex IADS

Regular expressions

- What is regular expressions:
  - A description of patterns in a text
  - Usage:
    - Detect match in text
    - Find and replace
  - Not so simple to read until you get used to it:
    - e.g. `/^(a|e).{1,3}\d+\b/`
      - What this regex match?
- All major languages have own version of regex
  - Perl
  - R
  - Python

Quantifiers: * + ?

- Candidate strings:
  - `essex, esse, esex, eex, essexex, elex`
- `*`: 0 or more occurrences
  - `essex*`
  - `ess(ex)*`
  - `e.*x`
- `+`: 1 or more occurrences
  - `essex+`
  - `es+ex`
- `?`: 0 or 1 occurrences
  - `essex?`
  - `ess(ex)?`

Character class

Expression	Meaning
\d	Matches any digit (Arabic numeral)
\D	Matches any character that is not a digit (Arabic numeral)
\w	Matches any alphanumeric character from the basic Latin alphabet, including the underscore
\W	Matches any character that is not a word character from the basic Latin alphabet
\s	Matches a single white space character, including space, tab, line feed etc
\S	Matches a single character other than white space

Assertion

Expression	Meaning
^	Matches the beginning of input
$	Matches the end of input
\b	Matches a word boundary
\B	Matches a non-word boundary
.	Wildcard (matches any character)
|	Disjunction (e.g. (bread|rice))
[0-9]	A range of characters (also ,[a-z], [A-Z])
[abk]	a or b or k
[^abk]	Negation of [abk] (not a and b and k)

Quantifiers: {,}

- `{,}` derermines the range of matches:
  - \d{1,4}
  - \s{1,3}
- `{}` exact number of characters
  - \d{4}

Miscellaneous

- Greedy v Non-greedy match
  - ".+a": greedy
  - ".+?a": non-greedy
  - Example:
    - Matching: Exaggeration
      - "Exa" v "Exaggera”

Examples

Expression	Candidates
\d+-\d+-\d+	Phone numbers?
[a-z]\w+@\w+\.(\w{2,3}){1,2}	?
\d{4}-\d{1,2}-\d{1,2}	?
@\w+\b	?
#\w+\b	?
[A-Z]{3}	?

In python

- Use re package
  - `re.search(pattern, string, flags)`
  - `re.sub(pattern, repl, string, flags)`

- Use pd.Series.str.methods()
  - `df["text"].str.contains(pattern)`