분석프로그래밍 I.11
문자열

국민대학교 경영학부 빅데이터경영통계학 전공

2018. 5. 27(월)

1 인코딩 관련 함수들

- 인코딩 바꾸기

  - 레이블만 바꾸기 `Encoding(x) <- "UTF-8"`

  - 레이블과 내용을 함께 바꾸기 `iconv(x, to="UTF-8")`

- 인코딩 추측하기

  - `stringi::stri_enc_detect(x)`[1]

- 인코딩 종류

  - `iconvlist()`

- 인코딩을 다루는 예

  - Listing 1 : R의 문자열은 문자를 특정한 인코딩 방법을 사용하여 저장한다. 인코딩이란 문자를 컴퓨터가 저장하는 방법을 의미한다. 구체적으로 (컴퓨터가 다룰 수 있는) 숫자(`0/1` 또는 16진수)를 문자 하나에 연결하는 방법을 의미한다. 대표적으로 ASCII 코드에서는 'A'는 16진수로 41, 'B'는 16진수로 42이다. 이렇게 컴퓨터가 저장하고 있는 숫자는 특정한 인코딩 방법에 따라 문자로 해석되는데, 이때 인코딩 방법을 알아야 주어진 숫자를 정확한 문자로 해석할 수 있다. 현재 변수 `x`에 저장되어 있는 문자의 인코딩 방법은 `Encoding(x)`로 확인할 수 있고, `Encoding(x) <- "UTF-8"`을 통해 `UTF-8`으로 변경할 수도 있다.

  - 내용은 동일하게 보존하면서 인코딩을 바꾸는 경우 : Listing 2

[1] 화일의 경우에는 `readr::guess_encoding()`을 활용한다. 문자열 벡터의 경우에는 `readr::guess_encoding(charToRaw(x))`를 쓸 수 있다.

```r
x <- "안녕?"

Encoding(x)
charToRaw(x)

Encoding(x) <- "UTF-8"
print(x)
charToRaw(x)

Encoding(x) <- "unknown"
print(x)
charToRaw(x)

Encoding(x) <- "latin1"
print(x)
charToRaw(x)

iconvlist()

Encoding(x) <- "CP949"
print(x)
charToRaw(x)

y <- iconv(x, to='UTF-8')
print(y)
Encoding(y)

x <- 'A'
charToRaw(x)
y <- '\x41'
print(y)
Encoding(y) <- 'UTF-8'
print(y)

x <- '안녕'
charToRaw(x)
```

```
y <- '\xbe\xc8\xb3\xe7\x3f'
print(y)
Encoding(y)
```

Listing 1: 문자열 인코딩만 바꾸는 함수, Encoding

```
x <- "안녕?"

Encoding(x)
charToRaw(x)

y <- iconv(x, to='UTF-8')
print(y)
Encoding(y)
charToRaw(y)

x <- 'ABCD'
Encoding(x)
charToRaw(x)

y <- iconv(x, to='UTF-8')
print(y)
Encoding(y)
charToRaw(x)
```

Listing 2: 문자열의 내용은 보존하면서 인코딩만 변환, iconv

2 문자열 관련 함수들

- 표 1 참조.

3 정규표현식Regular Expression

3.1 참고

- `stringr::str_view()`
- `https://regex101.com/`
- `https://regexr.com/`
- `https://regexper.com/`

3.2 대표적인 정규표현식

- 표 2 참조.

3.3 POSIX 브라켓

- 표 3 참조.

표 1: 문자열 관련 함수들

기능	R의 베이스 함수	package:stringr
알파벳 대/소문자	`letters[], LETTERS[]`	
알파벳 대/소문자 변환	`tolower(), toupper()`	`str_to_title()`
문자 수	`nchar()`	`str_length()`
문자열 내 공란 제거	`trimws()`	`str_trim()`
문자열 연결	`paste(), sprintf()`	`str_c()`
문자열 분해	`strsplit()`	`str_split()`
문자열의 일부	`substring()`	`str_sub()`
특정 패턴의 존재 여부	`grepl()`	`str_detect(), str_count()`
특정 패턴의 위치	`gregexpr(), regexec()`	`str_locate()`
특정 패턴의 교체	`gsub()`	`str_replace()`
특정 패턴의 추출/배제	`regmatches(t, gregexpr(p, t))`	`str_extract(t, p)`, `str_match`

3.4 정규표현식을 활용할 수 있는 함수들

- `dir`(`list.files`), `apropos`, `find` 등

3.5 Approximate matching(edit-based distance)

- `stringdist::stringdist(a, b, method = )`
- `method =`
  - `'osa'`
  - `'lv'` : delete, insert, substitution
  - `'dl'` : delete, insert, substitution, transposition
  - `'hamming'`
  - `'lcs'` : delete, insert
  - `'gram'`
  - `'cosine'`
  - `'jaccard'`
  - `'jw'`
  - `'soundex'`

3.6 정규표현식을 활용한 몇 가지 예제

```
library(stringr)
library(readr)

# 구텐베르그 프로젝트에서 로미오와 줄리엣 전체를 하나의 문자열로 다운 받는다.
txt <- read_file('http://www.gutenberg.org/cache/epub/1112/pg1112.txt')

# 각 장(Scene)의 장소만 추출한다.
match <- str_match_all(txt, "Scene[^.]*[.]\\r\\n([^\\r]*)\\r")
scenes <- match[[1]][,2]
```

```
10 head(scenes)
11
12 # 각 장(Scene)의 장소만 추출한다. 다른 방법.
13 extract <- str_extract_all(txt, "Scene[^.]*[.]\\r\\n([^\\r]*)\\r")
14 ex <- gsub("Scene [^.]*.\\r\\n", "", extract[[1]])
15 scenes <- gsub("\\r", "", ex)
16 head(scenes)
17
18 #  로미오와 줄리엣을 한 줄을 하나의 문자열로 다운 받는다.
19 txt <- readLines('http://www.gutenberg.org/cache/epub/1112/pg1112.txt',
       encoding='UTF-8')
20
21 l <- grep("^Scene", txt)
22 txt[l+1]
```

Listing 3: 정규표현식을 활용한 예제

표 2: 정규표현식

정규표현식	의미
n2ow	n2ow
[abc]	a, b, 또는 c
[a-z]	문자 a에서 z까지
[0-9]	0에서 9까지
[^abc]	a, b, c를 **제외한** 모든 문자
[\\^], [\\-]	^, -를 [ ] 안에서 나타내기
\d	숫자
\D	숫자를 **제외한** 모든 문자
\w	알파벳, 한글, 숫자(alphanumeric)
\W	알파벳, 한글, 숫자를 **제외한** 모든 문자(non-alphnumeric)
\s	공란(' ', '\t', '\r', '\n')
\S	공란을 **제외한** 모든 문자
\b	단어 경계(word boundary): w와 W의 연결
\B	단어 경계가 아닌 경우
.	**모든** 문자
\.	.(마침표)
\[, \]	[,]
\\	\
a{m}	a를 m번 반복
[cd]{m,n}	c 또는 d를 m에서 n번의 반복
*	0번 이상의 반복
+	1번 이상의 반복
?	0번 또는 1번
{m,n}?, *?, +?, ??	더 적은 횟수를 선호
\{, \}, *, \+, \?	{, }, *, +, ?
^ $	시작과 끝
( )	그룹
(ab(cd))	그룹 내의 그룹
(now|never)	now 또는 never
\^ , \$, \(, \)	^, $, (,)
\1, \2	첫 번째, 두 번째 그룹

표 3: POSIX 브라켓

POSIX 브라켓	의미	다른표현
[:alnum:]	알파벳과 숫자	[a-zA-Z0-9]
[:alpha:]	알파벳	[a-zA-Z]
[:ascii:]	아스키 문자	[\x00-\x7F]
[:blank:]	Space와 Tab	[\t]
[:cntrl:]	제어문자	[\x00-\x1F\x7F]
[:digit:]	숫자	[0-9]
[:graph:]	볼 수 있는 문자(공란, 제어문자 제외 모든 문자)	[\x21-\x7E]
[:lower:]	소문자	[a-z]
[:print:]	볼 수 있는 문자와 공란(제어문자 제외)	[\x20-\x7E]
[:punct:]	문장부호와 기호	[!"\#$%&'()*+,\-./:; <=>?\[\\\]^_`{|}]
[:space:]	모든 공란 문자(줄바꿈 포함)	[\t\r\n\v\f]
[:upper:]	대문자	[A-Z]
[:word:]	단어 문자(문자, 숫자, 밑줄)	[A-Za-z0-9_]
[:xdigit:]	16진수 숫자	[A-Fa-f0-9]