CW01b

Ralph\_Huang

2020/8/21

## The rest of packages for CW01

Well, in order to finish the homework in time, I have to find some easy packages.

### jsonlite

This is a important and extremely easy package to turn json file into data frame.

library(jsonlite)  
json <-   
'[  
 {"Name" : "Mario", "Age" : 32, "Occupation" : "Plumber"},  
 {"Name" : "Peach", "Age" : 21, "Occupation" : "Princess"}  
]'  
mydf <- fromJSON(json)  
mydf

## Name Age Occupation  
## 1 Mario 32 Plumber  
## 2 Peach 21 Princess

### utf-8

UTF-8 is the most widely used encoding nowadays. It would be confulsed when encoding goes wrong.

#install.packages("utf8")  
library(utf8)  
x <- c("fa\u00E7ile", "fa\xE7ile", "fa\xC3\xA7ile")  
Encoding(x) <- c("UTF-8", "latin1", "bytes")  
as\_utf8(x)

## [1] "fa<U+00E7>ile" "fa<U+00E7>ile" "fa<U+00E7>ile"

utf8\_normalize("𝖸𝗈 𝐔𝐧𝐢𝐜𝐨𝐝𝐞 𝗅 𝗁𝖾𝗋𝖽 𝕌 𝗅𝗂𝗄𝖾 𝑡𝑦𝑝𝑒𝑓𝑎𝑐𝑒𝑠 𝗌𝗈 𝗐𝖾 𝗉𝗎𝗍 𝗌𝗈𝗆𝖾 𝚌𝚘𝚍𝚎𝚙𝚘𝚒𝚗𝚝𝚜 𝗂𝗇 𝗒𝗈𝗎𝗋 𝔖𝔲𝔭𝔭𝔩𝔢𝔪𝔢𝔫𝔱𝔞𝔯𝔶 𝔚𝔲𝔩𝔱𝔦𝔩𝔦𝔫𝔤𝔳𝔞𝔩 𝔓𝔩𝔞𝔫𝔢 𝗌𝗈 𝗒𝗈𝗎 𝖼𝖺𝗇 𝓮𝓷𝓬𝓸𝓭𝓮 𝕗𝕠𝕟𝕥𝕤 𝗂𝗇 𝗒𝗈𝗎𝗋 𝒇𝒐𝒏𝒕𝒔.",  
 map\_compat = TRUE)

## [1] "<U+0001D5B8><U+0001D5C8> <U+0001D414><U+0001D427><U+0001D422><U+0001D41C><U+0001D428><U+0001D41D><U+0001D41E> <U+0001D5C5> <U+0001D5C1><U+0001D5BE><U+0001D5CB><U+0001D5BD> <U+0001D54C> <U+0001D5C5><U+0001D5C2><U+0001D5C4><U+0001D5BE> <U+0001D461><U+0001D466><U+0001D45D><U+0001D452><U+0001D453><U+0001D44E><U+0001D450><U+0001D452><U+0001D460> <U+0001D5CC><U+0001D5C8> <U+0001D5D0><U+0001D5BE> <U+0001D5C9><U+0001D5CE><U+0001D5CD> <U+0001D5CC><U+0001D5C8><U+0001D5C6><U+0001D5BE> <U+0001D68C><U+0001D698><U+0001D68D><U+0001D68E><U+0001D699><U+0001D698><U+0001D692><U+0001D697><U+0001D69D><U+0001D69C> <U+0001D5C2><U+0001D5C7> <U+0001D5D2><U+0001D5C8><U+0001D5CE><U+0001D5CB> <U+0001D516><U+0001D532><U+0001D52D><U+0001D52D><U+0001D529><U+0001D522><U+0001D52A><U+0001D522><U+0001D52B><U+0001D531><U+0001D51E><U+0001D52F><U+0001D536> <U+0001D51A><U+0001D532><U+0001D529><U+0001D531><U+0001D526><U+0001D529><U+0001D526><U+0001D52B><U+0001D524><U+0001D533><U+0001D51E><U+0001D529> <U+0001D513><U+0001D529><U+0001D51E><U+0001D52B><U+0001D522> <U+0001D5CC><U+0001D5C8> <U+0001D5D2><U+0001D5C8><U+0001D5CE> <U+0001D5BC><U+0001D5BA><U+0001D5C7> <U+0001D4EE><U+0001D4F7><U+0001D4EC><U+0001D4F8><U+0001D4ED><U+0001D4EE> <U+0001D557><U+0001D560><U+0001D55F><U+0001D565><U+0001D564> <U+0001D5C2><U+0001D5C7> <U+0001D5D2><U+0001D5C8><U+0001D5CE><U+0001D5CB> <U+0001D487><U+0001D490><U+0001D48F><U+0001D495><U+0001D494>."

print(intToUtf8(0x1F600 + 0:79))

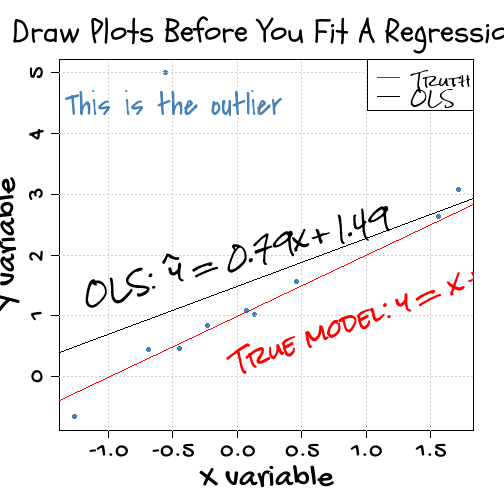
## [1] "<U+0001F600><U+0001F601><U+0001F602><U+0001F603><U+0001F604><U+0001F605><U+0001F606><U+0001F607><U+0001F608><U+0001F609><U+0001F60A><U+0001F60B><U+0001F60C><U+0001F60D><U+0001F60E><U+0001F60F><U+0001F610><U+0001F611><U+0001F612><U+0001F613><U+0001F614><U+0001F615><U+0001F616><U+0001F617><U+0001F618><U+0001F619><U+0001F61A><U+0001F61B><U+0001F61C><U+0001F61D><U+0001F61E><U+0001F61F><U+0001F620><U+0001F621><U+0001F622><U+0001F623><U+0001F624><U+0001F625><U+0001F626><U+0001F627><U+0001F628><U+0001F629><U+0001F62A><U+0001F62B><U+0001F62C><U+0001F62D><U+0001F62E><U+0001F62F><U+0001F630><U+0001F631><U+0001F632><U+0001F633><U+0001F634><U+0001F635><U+0001F636><U+0001F637><U+0001F638><U+0001F639><U+0001F63A><U+0001F63B><U+0001F63C><U+0001F63D><U+0001F63E><U+0001F63F><U+0001F640><U+0001F641><U+0001F642><U+0001F643><U+0001F644><U+0001F645><U+0001F646><U+0001F647><U+0001F648><U+0001F649><U+0001F64A><U+0001F64B><U+0001F64C><U+0001F64D><U+0001F64E><U+0001F64F>"

### showtext

The default font is too normal to make the whole picture shiny. Merely changing the font can beautify the plot a lot.

## Loading required package: sysfonts

## Loading required package: showtextdb



### redoc

Note that the echo = FALSE parameter was added to the code chunk to prevent printing of the R code that generated the plot.