Final project

Ruijie Ma

2018/5/6

datacleaning

In this project, I have used skills that I learnt from this class including tidyverse,ggplot,shinny app.I follows the guideline that using real time data to do analyzation. My topic is about effect that governmental regulation towards the price of Bitcoin which is a digital currency. I choose to use the api from Alphavantage which is a set of real time data including open price, daily high and low and volume and these factors can be found in the shinny sidebar.about the Bitcoin's price in both USD and CNY. I delete the part of USD since double currency would make the scale of the graph chaotic. I choose to analyze with CNY because it has relatively lower value of exchange rate which make the change more obvious on the graph. I consider that the governmental regulation does influence the price of Bitcoin and my study confirms this point. We will see the change of South Korean's regulation towards Bitcoin in this year in the graph that creates a lowest open price of Bitcoins in recent two years.This R project shows only the data cleanning part and others are showed in the shinny app.

```
knitr::opts_chunk$set(echo = TRUE)
library(shiny)
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching packages ---------------------------- tidyverse 1.2.1 --
```

```
## <U+221A> tibble  1.4.2     <U+221A> purrr   0.2.4
## <U+221A> tidyr   0.8.0     <U+221A> dplyr   0.7.4
## <U+221A> readr   1.1.1     <U+221A> stringr 1.2.0
## <U+221A> tibble  1.4.2     <U+221A> forcats 0.2.0
```

```
## -- Conflicts ------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(shinythemes)
library(shinydashboard)
```

```
##
## Attaching package: 'shinydashboard'
```

```
## The following object is masked from 'package:graphics':
##
##     box
```

```
library(curl)
```

```
## Warning: package 'curl' was built under R version 3.4.4
```

```
##
## Attaching package: 'curl'

## The following object is masked from 'package:readr':
##
##     parse_date
library(jsonlite)

##
## Attaching package: 'jsonlite'

## The following object is masked from 'package:purrr':
##
##     flatten

## The following object is masked from 'package:shiny':
##
##     validate
url<-'https://www.alphavantage.co/query?function=DIGITAL_CURRENCY_DAILY&symbol=BTC&market=CNY&apikey=7U

data1 <- read_json(url,format="jsonl")

thedata<-lapply(data1$`Time Series (Digital Currency Daily)`,unlist)

thedata<-as.data.frame(thedata)

thedata<-t(thedata)

thedata<-thedata[,-c(2,4,6,8,10)]

col_names<-colnames(thedata)

col_names<-unlist(lapply(col_names,function(x){
  temp<-unlist(strsplit(x,"[.]"))
  temp[2]
}))

col_names<-gsub(" ","",col_names)

col_names<-gsub("[(]","_",col_names)

col_names<-gsub("[)]","",col_names)

time<-rownames(thedata)

time<-gsub("X","",time)

time<-as.Date(gsub("[.]","-",time),format="%Y-%m-%d")
mydata<-data.frame(thedata)

names(mydata)<-col_names

mydata$time<-time

mydata[,1:5]<-apply(mydata[,1:5],1,as.numeric)
```

```
ggplot()
```


ui part

```
knitr::opts_chunk$set(echo = TRUE)

ui <- fluidPage(theme = shinytheme("united"),
                titlePanel("Bitcoin price"),

                navbarPage(
                  # theme = "cerulean",  # <--- To use a theme, uncomment this
                  "Myapp",
                  tabPanel("Part1",
                           sidebarPanel(
                             selectInput("var",
                                         label = "Choose the varaible to display",
                                         choices = "",
                                         selected = ""),
                             sliderInput("range", "Time interval:",min =as.Date("2014-04-01"),
                                         max = as.Date("2018-05-02"), value =c(as.Date("2018-03-02"),as
                           mainPanel(
                             tabsetPanel(type = "tabs",
                                         tabPanel("interval", plotOutput("plot_11"),h4(' In this projec
                             ))
                  )
                )
)
```

server part of shinny

```
knitr::opts_chunk$set(echo = TRUE)
url<-'https://www.alphavantage.co/query?function=DIGITAL_CURRENCY_DAILY&symbol=BTC&market=CNY&apikey=7U
data1 <- read_json(url,format="jsonl")

thedata<-lapply(data1$`Time Series (Digital Currency Daily)`,unlist)
thedata<-as.data.frame(thedata)
thedata<-t(thedata)
thedata<-thedata[,-c(2,4,6,8,10)]
col_names<-colnames(thedata)
col_names<-unlist(lapply(col_names,function(x){
  temp<-unlist(strsplit(x,"[.]"))
  temp[2]
}))
col_names<-gsub(" ","",col_names)
col_names<-gsub("[(]","_",col_names)
col_names<-gsub("[)]","",col_names)
time<-rownames(thedata)
time<-gsub("X","",time)
time<-as.Date(gsub("[.]","-",time),format="%Y-%m-%d")
mydata<-data.frame(thedata)
names(mydata)<-col_names
mydata$time<-time
mydata[,1:5]<-apply(mydata[,1:5],1,as.numeric)

server <- function(input, output,session) {


  updateSliderInput(session,"range", "Time interval:",min =time[length(time)],
                    max =time[1], value = c(time[50],time[1]))

  updateSelectInput(session, "var",
                    label = "Choose the varaible to display",
                    choices = col_names,
                    selected = "open_CNY")

  thedata<-reactive({
    date1<-input$range[1]
    date2<-input$range[2]
    select_col<-input$var
    temp<-subset(mydata,mydata$time>=date1&mydata$time<=date2)
    temp
  })

  output$plot_11<-renderPlot({
    if(!is.null(thedata())){
      ggplot(thedata(),aes_string(x='time',y=input$var))+geom_line()
    }
  })
}
```

run shinny app

```
knitr::opts_chunk$set(echo = TRUE)
shinyApp(ui = ui, server = server)
```

Shiny applications not supported in static R Markdown documents