HW 1 Due Tuesday Sept 6, 2016. Upload R file to Moodle with name: HW1_490IDS_YOURUNI.R

Do Not remove any of the comments. These are marked by

###Name: Zixin Ouyang

Load the data for this assignment into your R session

with the following command:

load(url("http://courseweb.lis.illinois.edu/~jguo24/SFTemps.rda"))

Check to see that the data were loaded by running:

objects()

This should show five variables: dates, dayOfMonth, month, temp, and year

Use the length() function to find out how many observations there are.

length(dates) # there are 5534 observations.

For the following questions, use one of: head(), summary(),

class(), min(), max(), hist(), quantile() to answer the questions.

1. (1) What was the coldest temperature recorded in this time period?

summary(temp)

 # Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

 # 38.30 53.00 57.00 56.96 60.80 79.60 36

 # so the coldest temperature recorded is 38.3 degrees Farenheit

2. (1) What was the average temperature recorded in this time period?

summary(temp)

 # so the average temperature recorded in this time period is 56.96 degrees Farenheit

3. (2) What does the distribution of temperatures look like, i.e.

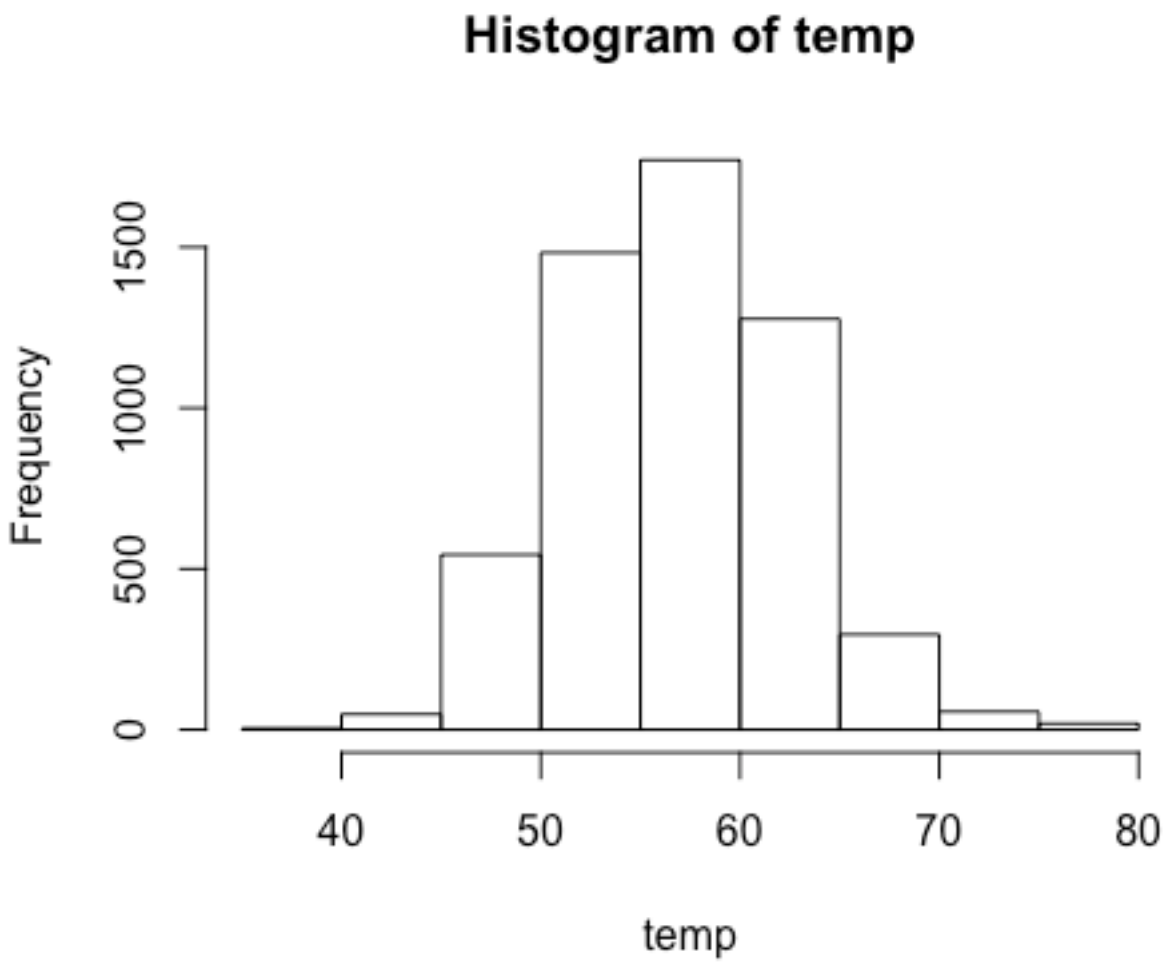
are there roughly as many warm as cold days, are the temps

clustered around one value or spread evenly across the range

of observed temperatures, etc.?

hist(temp)

 # it can be roughly as normal distribution based on the histogram of tempature.



4. (1) Examine the first few values of dates. These are a special

type of data. Confirm this with class().

 # the first few values of dates can be tested by head()

head(dates)

 # [1] "1995-01-01" "1995-01-02" "1995-01-03" "1995-01-04" "1995-01-05" "1995-01-06"

 # use class() function to exam the data type one by one.

```
class(dates)
# Date
class(dayOfMonth)
# integer
class(month)
# integer
class(temp)
# numeric
class(year)
# integer
# The special data type is the date type of dates variable.


# 5. (1) We would like to convert the temperature from Farenheit to Celsius.
# Below are several attempts to do so that each fail.
# Try running each expression in R.
# Record the error message in a comment
# Explain what it means.
# Be sure to directly relate the wording of the error message with the problem you find in the expression.
For Fahrenheit to Celsius: Celsius = (5 ÷ 9) × (Fahrenheit - 32)
(temp -32)
### Error message here
```

no error message shown in this expression

Explanation here

but this expression can’t change the Fahrenheit to Celsius for it’s lack of 5/9 times.

(temp - 32)5/9

Error message here

 # Error: unexpected numeric constant in "(temp - 32)5"

Explanation here

the expression can’t get a numerical result, and should change into (temp-32)*5

5/9(temp - 32)

Error message here

 # Error: attempt to apply non-function

Explanation here

there are also missing a multiply *, 5/9*(temp-32)

[temp - 32]5/9

Error message here

Error: unexpected '[' in "["

Explanation here

the sign “[“ is not recognized by R, it should express as “(“.

6. (1) Provide a well-formed expression that correctly performs the

calculation that we want. Assign the converted values to tempC.

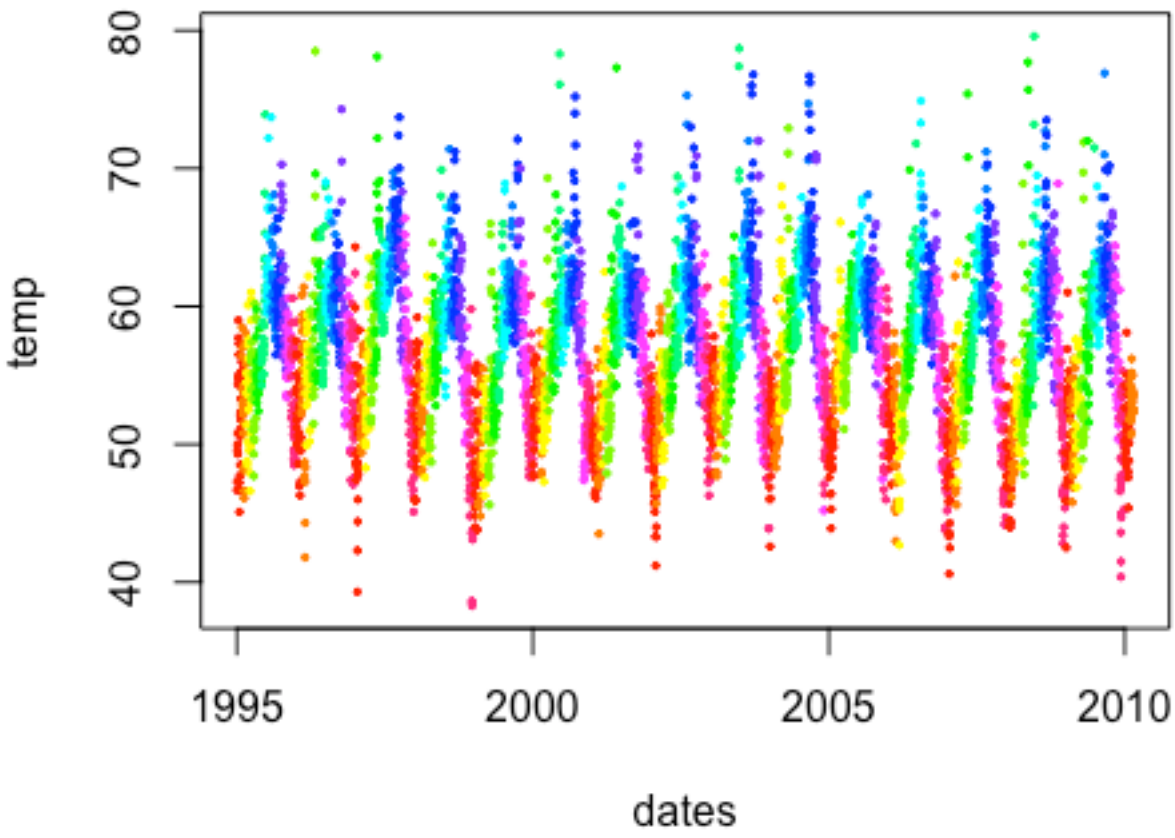
tempc=(5/9)*(temp - 32)

7. Run the following code to make a plot.

(don't worry right now about what this code is doing)

plot(temp~dates, col = rainbow(12)[month], type="p", pch=19, cex = 0.3)

the result is shown as follow:

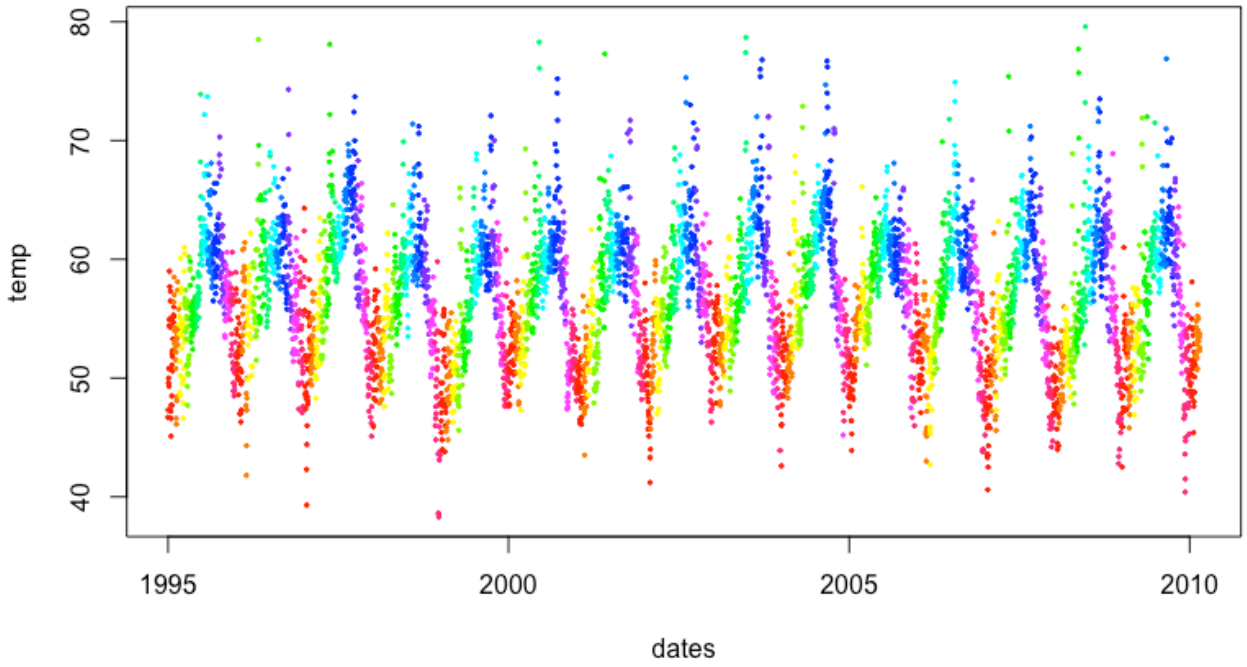


(1) Use the Zoom button in the Plots window to enlarge the plot.

Resize the plot so that it is long and short, so it is easier to read.

Include this plot in the homework your turn in.

###After click the zoom button, the plots is like



(1) Make an interesting observation about temp in the Bay Area

based on this plot (something that you couldn't see with

the calculations so far.)

Your answer goes here

The plot is wave-shaped. The temperature kept going up and going down.

(1) What interesting question about the weather in the SF Bay Area

would you like to answer with these data, but don't yet know

how to do it?

Your answer goes here

The trend of temperature in the SF Bay Area.

For the remainder of this assignment we will work with

one of the random number generators in R.

8. (5). Use the following information about you to generate

some random values:

#a. Use the day of the month you were born for the mean of the normal.

the day of the month of my birthday is 08, so the mean of the normal is 8

#b. Use your year of birth for the staZixndard deviation (sd) of the normal curve.

my year of birth is 1993, so the standard deviation of the normal curve is 1993.

#c. Generate 5 random values using the parameters from a and b.

rnorm(5, mean = 8, sd = 1993)

the result is [1] 611.8116 -529.7399 2526.3351 2219.8547 1243.0969

#d. Assign the values to a variable named with your first name.

Zixin=c(611.8116, -529.7399, 2526.3351, 2219.8547, 1243.0969)

#e. Provide the values generated.

Zixin

The values generated are 611.8116, -529.7399, 2526.3351,

2219.8547, 1243.0969

9. (1). Generate a vector called "normsamps" containing

100 random samples from a normal distribution with

mean 2 and SD 1.

```
normsamps = rnorm(100, 2, 1)
```

10. (1). Calculate the mean and sd of the 100 values.

```
mean(normsamps)

sd(normsamps)
```

The return values from your computation go here

the mean value is 2.064118

the sd value is 0.9672562

11. (1). Use implicit coercion of logical to numeric to calculate

the fraction of the values in normsamps that are more than 3.

```
coerced = 1 * normsamps

norm_frac = length(coerced[coerced > 3]) / length(coerced)

print(norm_frac)
```

The fraction is 0.17

12. (1). Look up the help for rnorm.

You will see a few other functions listed.

Use one of them to figure out about what answer you

should expect for the previous problem.

That is, find the area under the normal(2, 1) curve

to the right of 3. This should be the chance of getting

a random value more than 3. What value do you expect?

What value did you get? Why might they be different?

help(rnorm)

pnorm(3, mean = 2, sd = 1, lower.tail = FALSE, log.p = FALSE)

I expect the value will be 0.17, but I got 0.1586553.

I think this is because 0.17 is calculated by the 100 values in normsamps.

But 0.1586553 is calculated by all the values in rnorm.