Yeolwu Seong

HUID: 21123536

CSCIE63 Big Data Analytics

Assignment 01

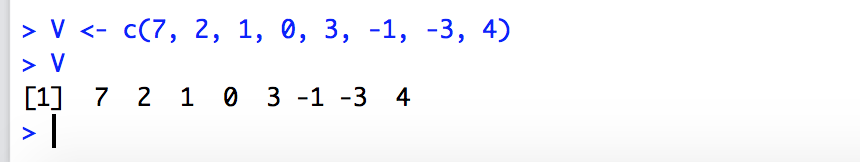
**Assignment 01 Solution**

Problem 1

Problem 1-1

“**Create a vector V with 8 elements (7, 2, 1, 0, 3, -1, -3, 4)”**

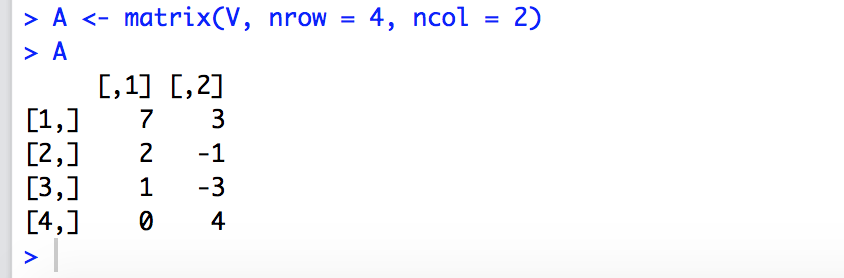
Solution: I create a vector V with the given 8 elements.



Problem 1-2

“**Transform that vector into a rectangular matrix A of dimensions 4X2 (4-row, 2-colums)”**

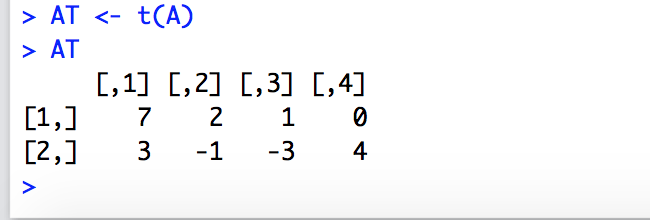
Solution: I create a 4X2 matrix A with the vector V by using matrix() with nrow and ncol parameters.



Problem 1-3

“**Create a matrix transpose to the above matrix A. Call that matrix AT”**

Solution: I create the matrix transpose by using t(A).

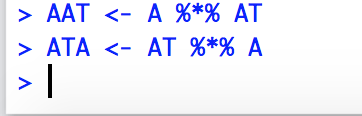


Problem 1-4

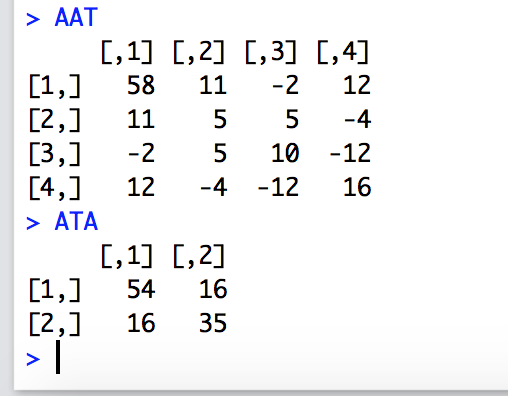
“**Calculate matrix products: A\*AT and AT\*A. Present the results. What are the dimensions of those two product matrices.”**

Solution:

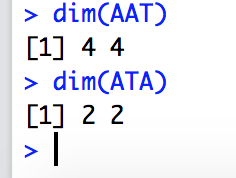
1. I calculate the matrix products by using %\*% operator and assigned the results to AAT and ATA respectively.



2. AAT is a 4X4 matrix and ATA is a 2X2 matrix.



3. By using dim() I prove the dimensions of the product matrices again.

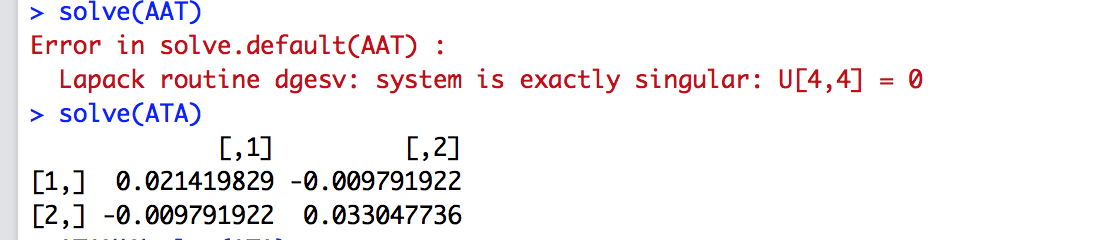


Problem 1-5

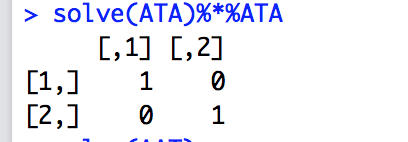
“**Square matrixes sometimes have an inverse matrix. Try calculating inverse matrices of above matrices A\*AT and AT\*A”**

Solution:

1. I try to calculate the inverse matrices of A\*AT and AT\*T by using solve().



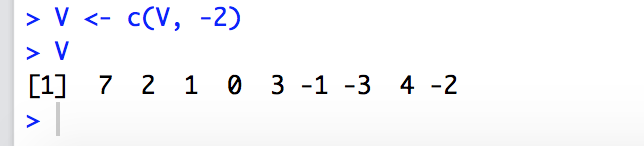
1. Because |AAT| is 0, AAT(A\*AT) is a singular matrix. So it cannot be inverted while I can get the inverse matrix of ATA(AT\*A). I prove it by the inverse matrix by AAT



Problem 1-6

“**Extend the above vector V with the ninth number of value -2. Do it elegantly by concatenating two vectors.”**

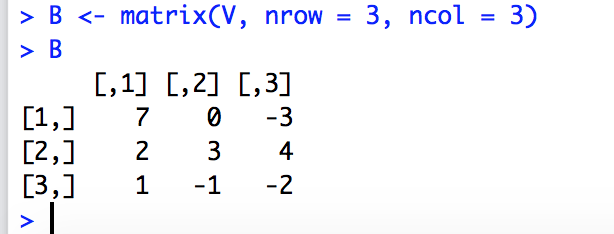
Solution: I extend V by concatenating V and a single number vector -2



Problem 1-7

“**Transform that extended vector into 3X3 matrix B”**

Solution: I create a 3X3 matrix B with the extended vector V by using matrix() with nrow and ncol parameters.

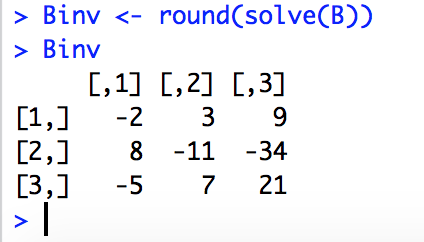


Problem 1-8

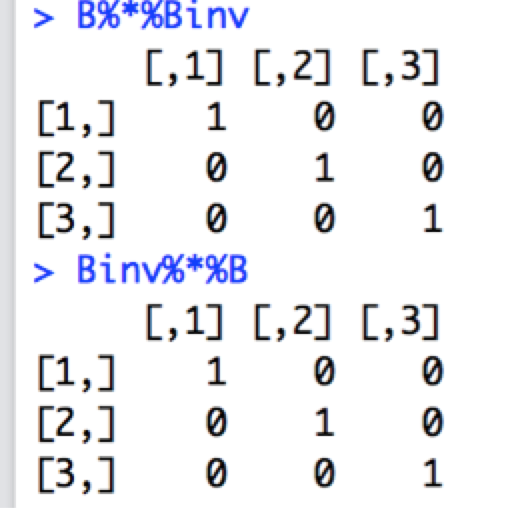
“**Calculate the inverse matrix of matrix B. Call it Binv. Demonstrate that the product of B and Binv is the same as the product of Binv and B and is equal to what?”**

Solution:

1. I calculate the inverse matrix (Binv) by using solve().



1. Then I calculate the product of B and Binv and the product of Binv and B.



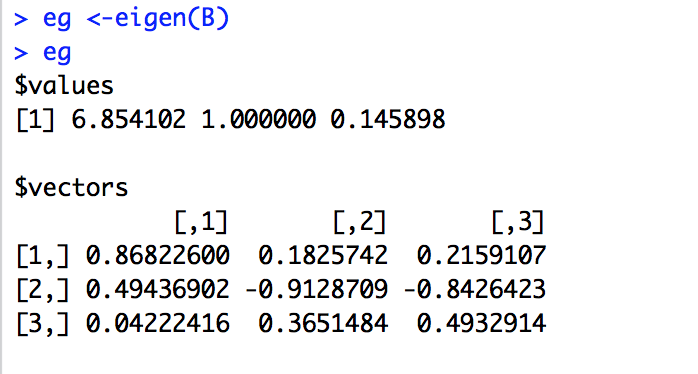
1. Both products are identical and are equal to a 3 X 3 identity matrix.

Problem 1-9

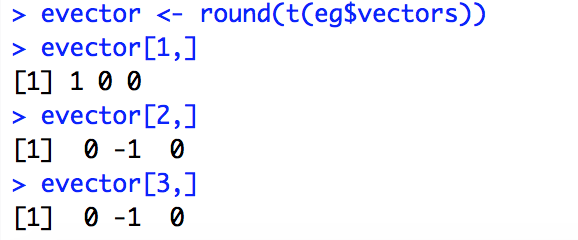
“**Determine the eigenvectors of matrixes B.”**

Solution:

1. I calculate the list of eigenvalues and eigenvectors of B by using eigen() and assign the list to eg.



1. Then determine the 3 eigenvectors with 3 columns of eg$vectors

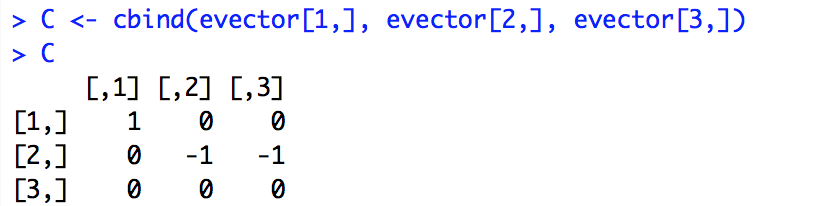


Problem 1-10

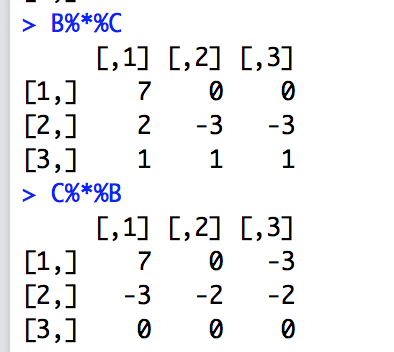
“**Construct a new matrix C which is made by using each eigenvector of matrix B as a column. Calculate the product of matrix C and matrix B and the product of matrix B and C. Is there any significance to the elements of the product matrixes.”**

Solution:

1. I construct the matrix C by binding the eigenvectors as columns.



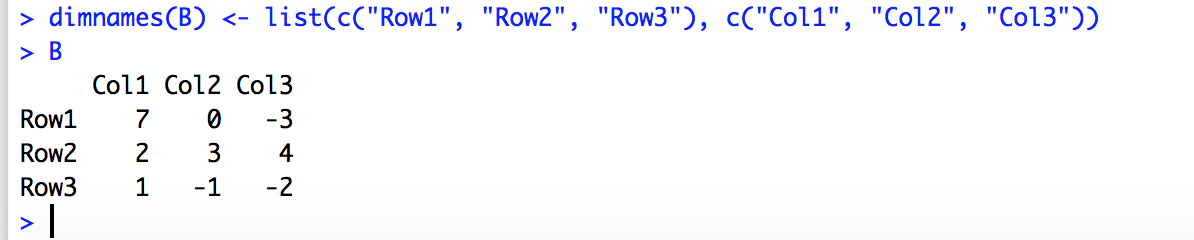
1. I calculate the product of C and B and the product of B and C



Problem 1-11

“**Transform matrix B into a matrix with names columns and named rows”**

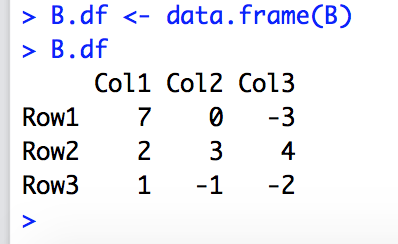
Solution: I name the rows and columns of matrix B by using dimnames()



Problem 1-12

“**Transformed that fully ‘named’ matrix into a data.frame”**

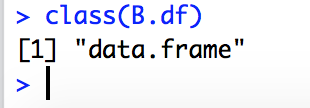
Solution: I transform the named matrix B into a data.frame B.df as below



Problem 1-13

“**Ask the object you just created what is its class”**

Solution: The type of the transformed object is “data.frame”

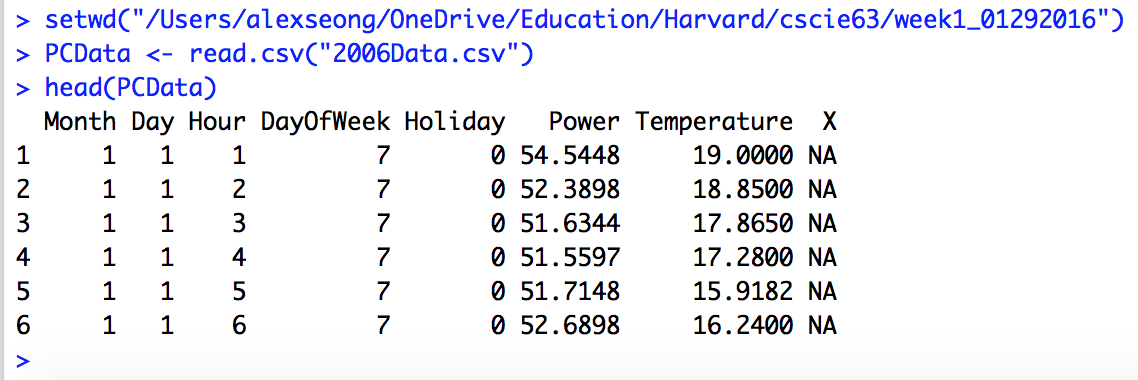


Problem 2

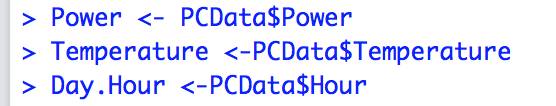
“**Consider file 2006Data.csv upload to the class site in Assignment 01 folder. File represents actual measurement of power consumption in a country somewhere in a California. Import data contained in that file into a data frame. You are expected to Google and find a function that will let you perform that import. Create a scatter plot of power consumption vs. temperature and power consumption vs. hour of the day. Subsequently create a boxplot with power on the vertical axis and hour of the day on the horizontal axis. The objective is to present the distribution (variation) of power consumption for every hour of the day.**”

Solution:

1. I change the working directory to the folder where the data file is located, then load the data from the csv file into PCData data.frame by using read.csv()

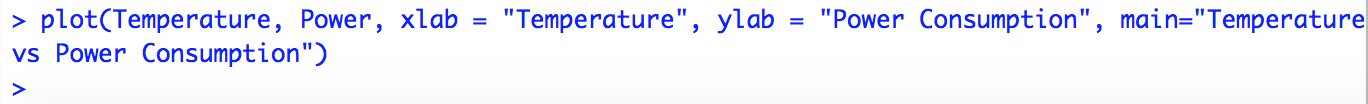


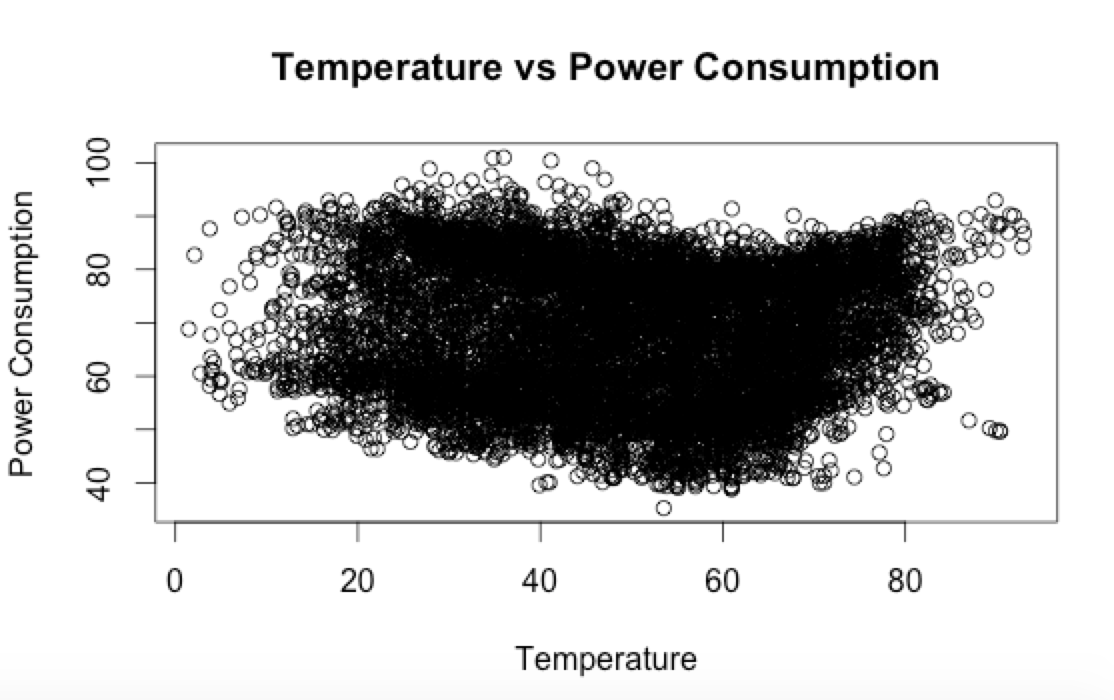
1. I get the 3 vectors from the data.frame



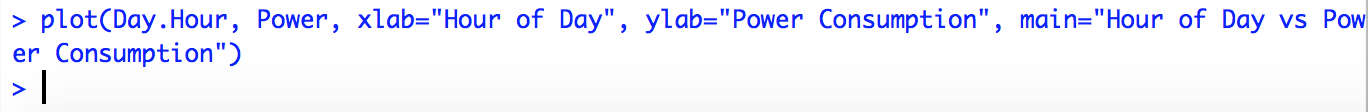
1. I create 2 scatter plots by using pot()

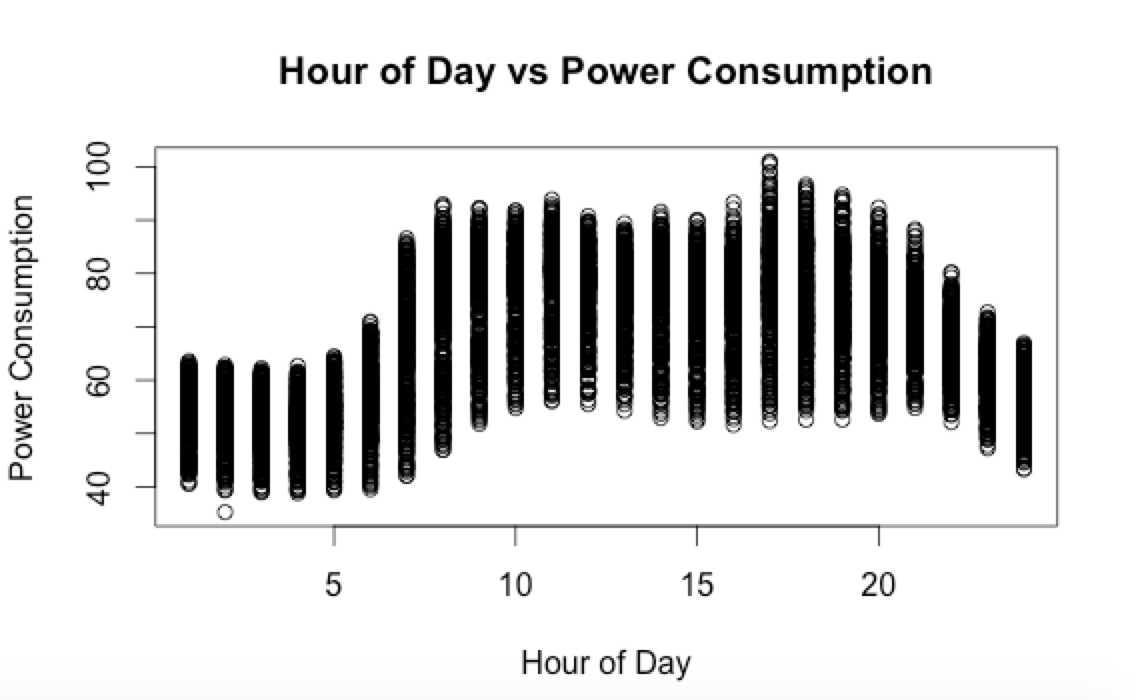
3-1 Temperature VS Power Consumption



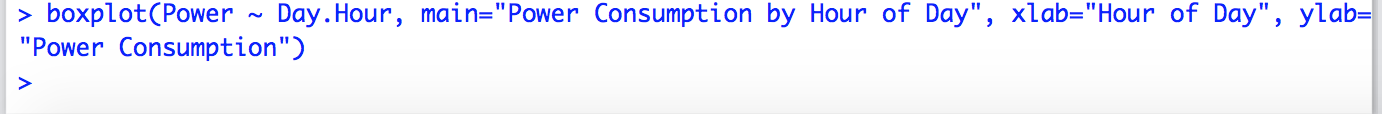


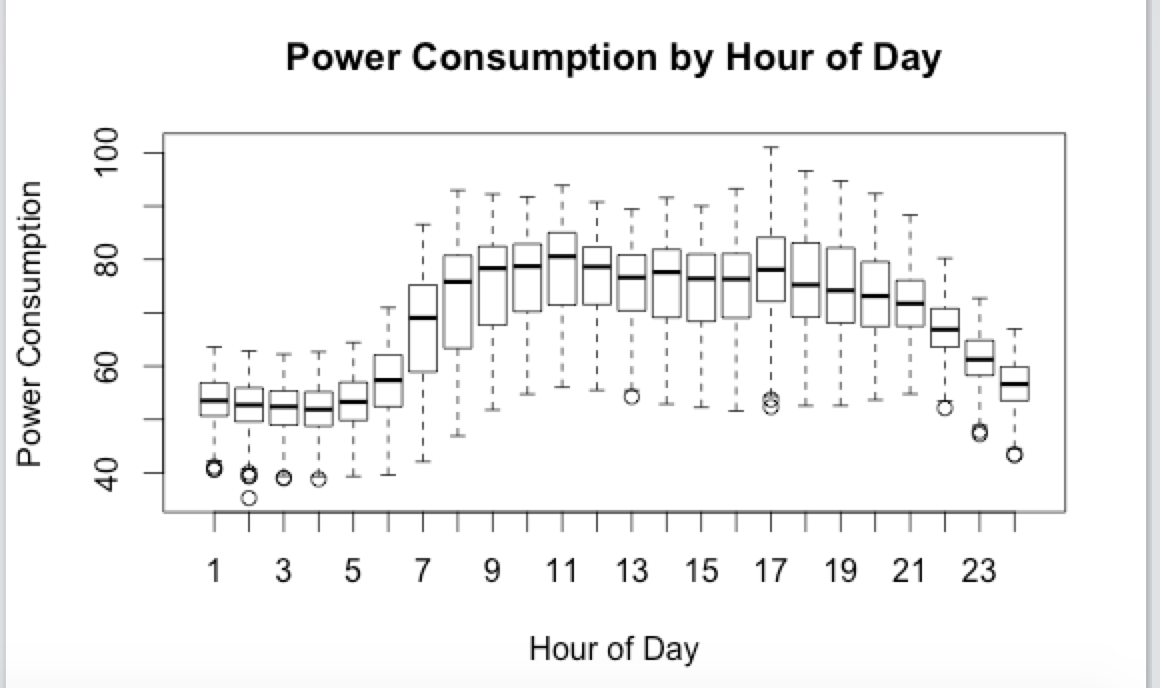
3-2 Hour of Day VS Power Consumption





1. I create a box plot with power on the vertical axis and hour of the day on the horizontal axis. The box plow shows the distribution of power consumption for every hour of day.



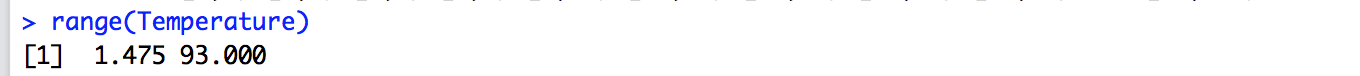


Problem 3.

**“Separate temperature scale in a reasonable number of intervals: 50 or 100. Calculate average power consumption, minimum power consumption and maximum power consumptions for every interval. Present those three sets of values on a single scatter graph (perhaps in different colours). Calculate three covariance matrixes between temperature and each of those power indicators (min, average, max)”**

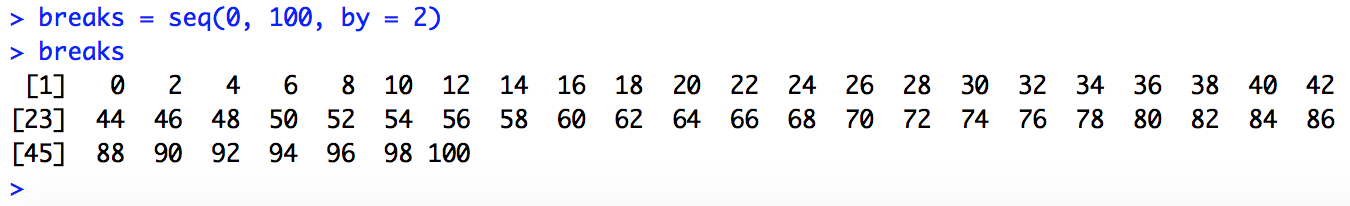
Solution:

1. I first find the range of vector Temperature.

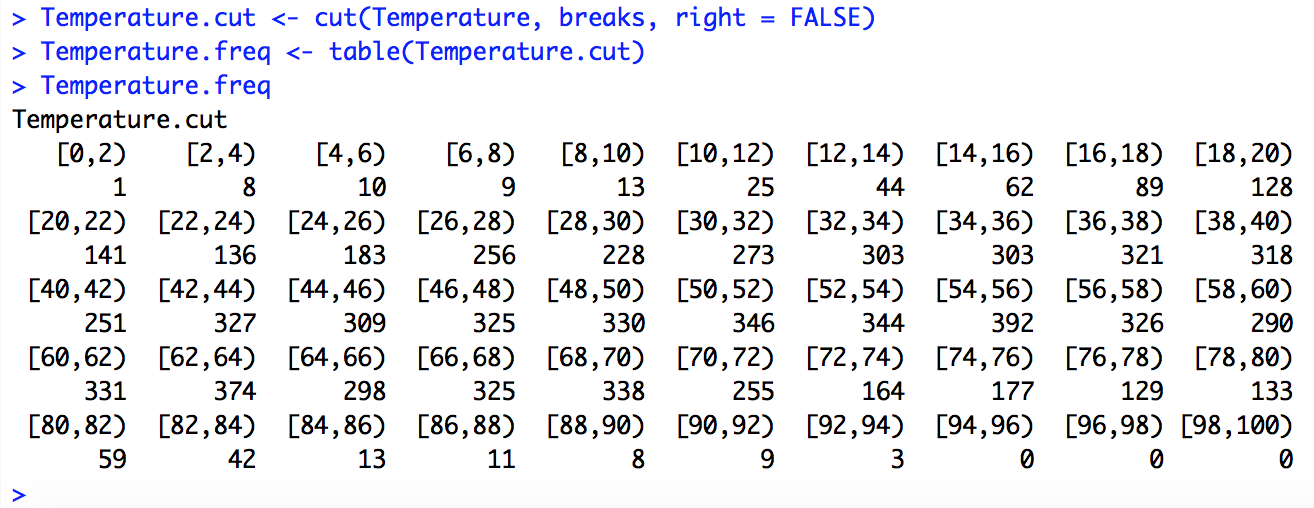


Observed temperatures are between 1.475 and 93.000

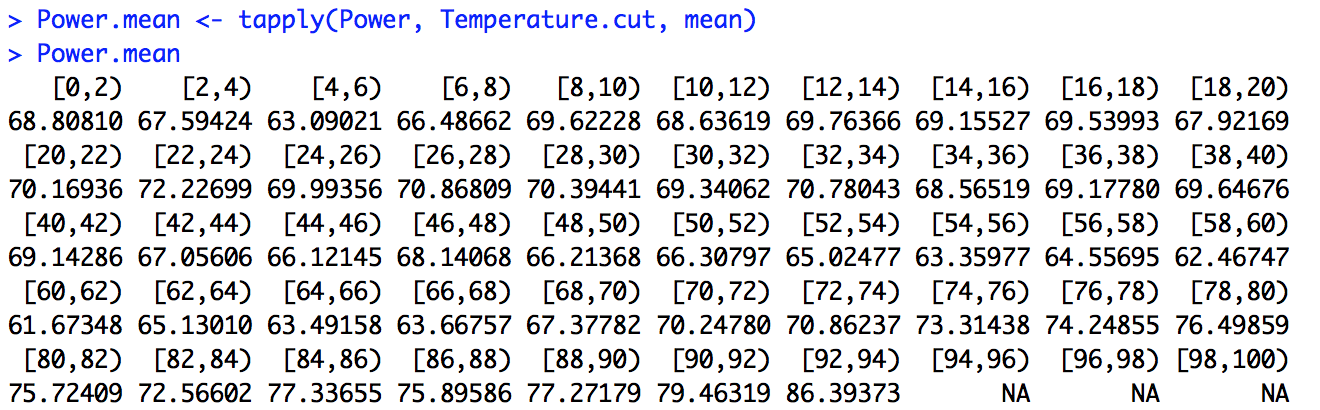
1. I set break points to get 50 equal distance intervals with the range of [0, 100]

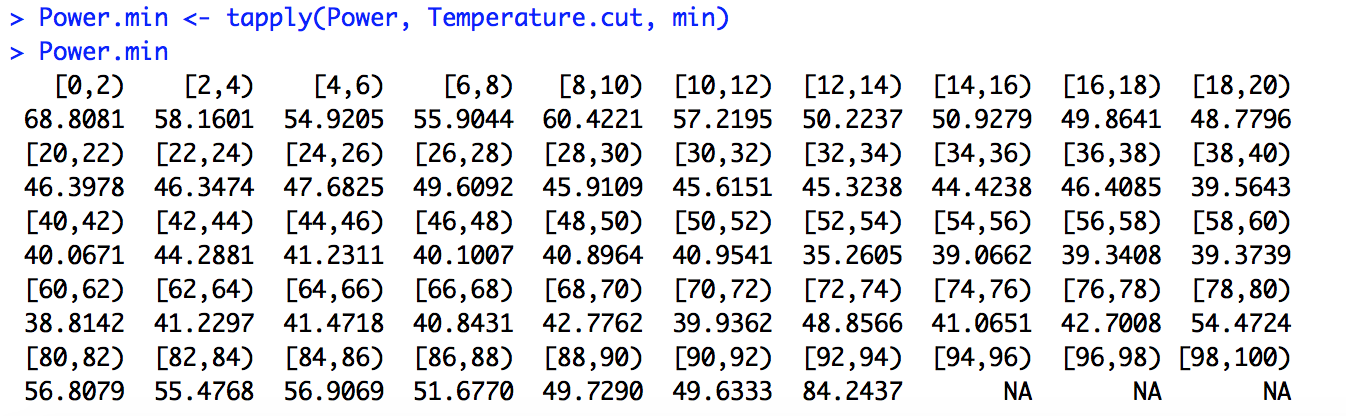


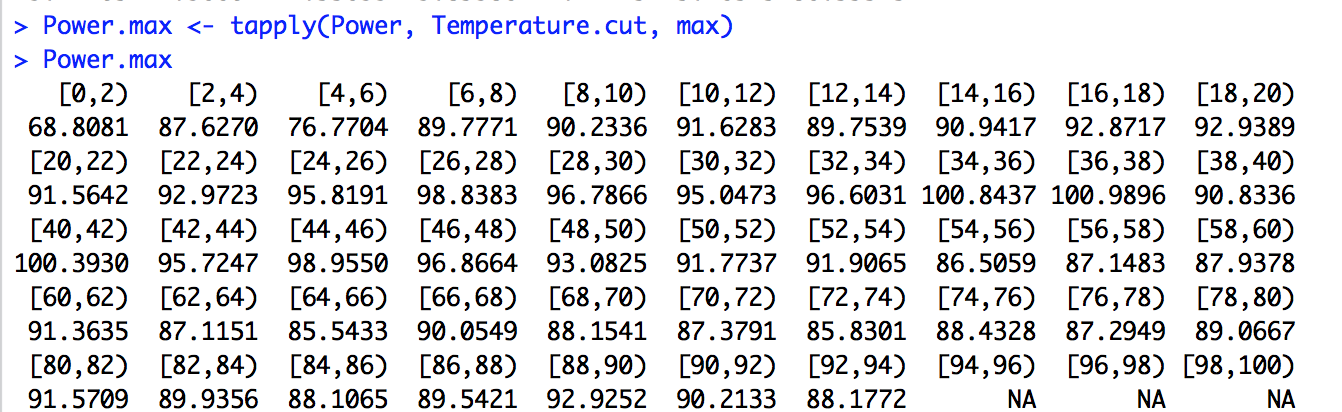
1. I assign values from vector Temperature to the intervals delimited by sequence breaks. The vector is converted to a factor Temperature.cut. Then I examine the frequency of Temperatures in each interval also.



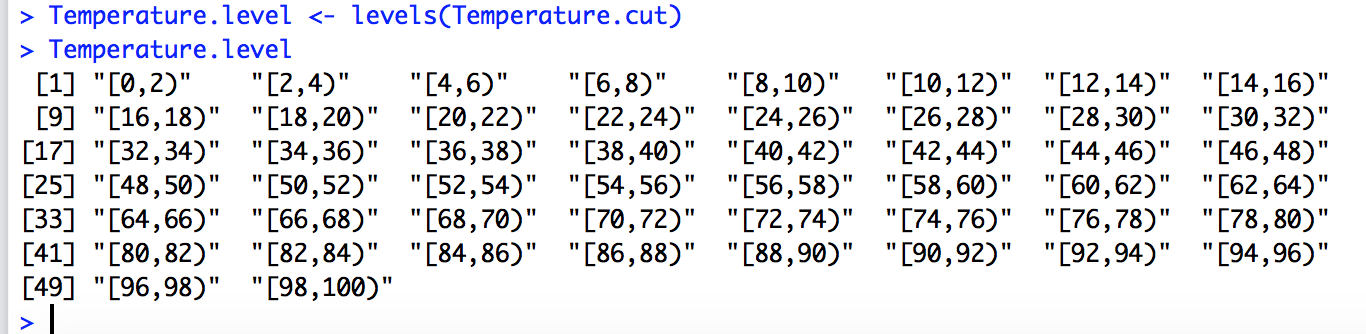
1. I calculate average power consumption, minimum power consumption and maximum power consumptions for every interval by using function tapply().







1. I get the levels of Temperature.cut to use for x-axis value.

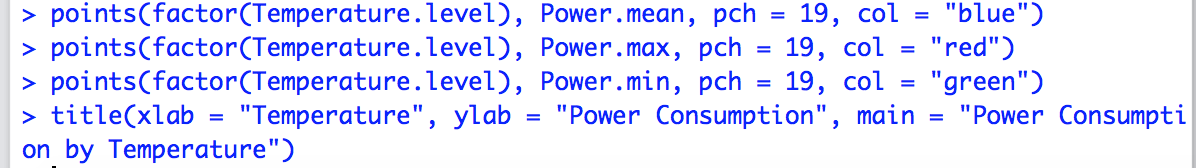


1. First I draw a plot with Powere.mean by Temperature.level. Because Temperature.level is text data, I use factor(Temperature.level) for the x-argument.

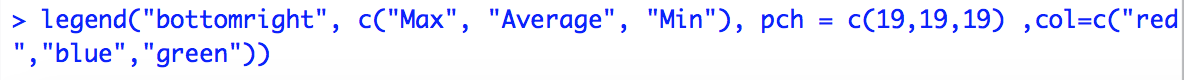
And I set ylim to c(0, 100) to put 3 points and a legend on the same space without any changes.

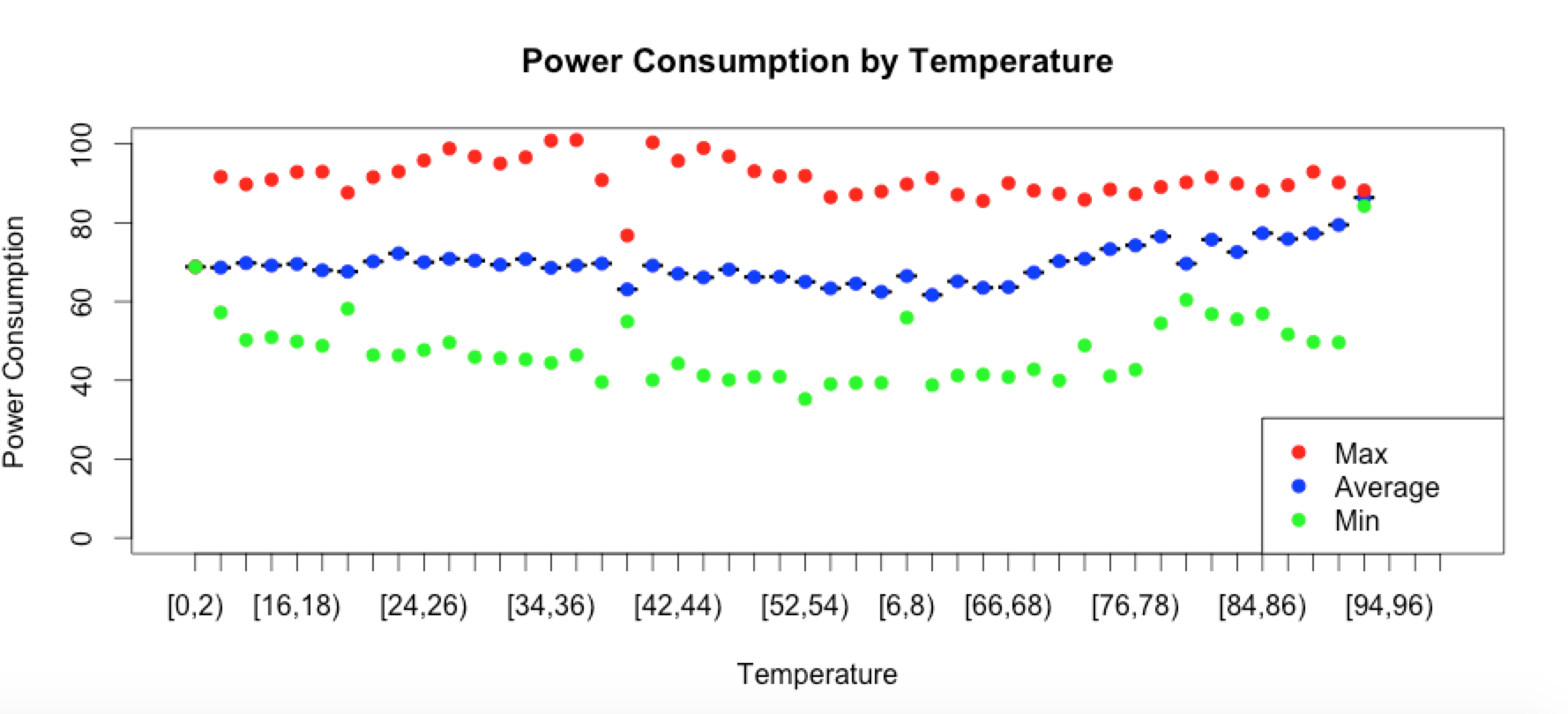


Then I put 3 points with different colors and titles and labels

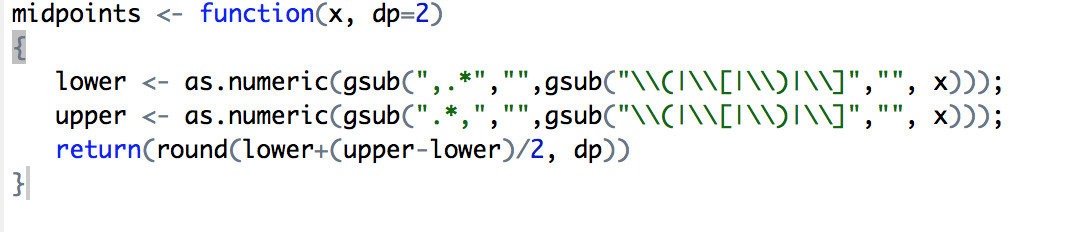


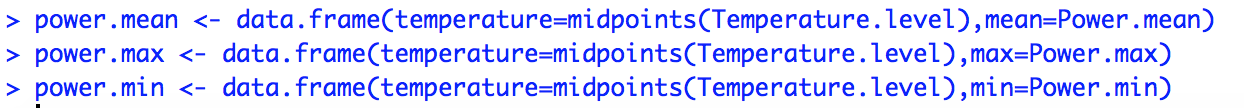
Finally I put a legend



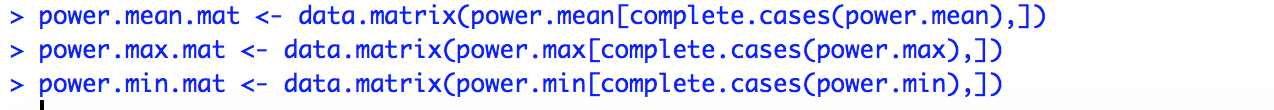


1. For the 3 covariance matrix, first I implement a function to get mid point from Temperature factor levels.



Then I create 3 data.frames(temperature midpoint with power mean, max, and min)

1. I make 3 matrixes from the 3 data frames after excluding the rows which have NA values by using complete.cases()



1. Then I calculate the 3 covariance matrixes by using the matrixes as arguments of cov()

