#Ryan Allison

#Assignment 3

**INPUT**

#Problem 1

#a

dat <- read.table(file.choose())

dat

#b

# Seems wrong becasue we have variables for the first row.

#c

dat <- read.table(file.choose(), header=T)

dat

# This is better because the column header lists the appropriate variable names.

names(dat)

#Problem 2

x<-dat$x

x

y<-dat$y

y

dat.2<-list(x,y)

dat.2

dput(dat.2,file.choose())

rm(list = ls())

G<-dget(file.choose())

G

dfG<-data.frame(G)

dfG

#Problem 3

data()

data(PlantGrowth)

PlantGrowth

#There's two variables: 'weight' and 'group'.

?PlantGrowth

require(stats); require(graphics)

boxplot(weight ~ group, data = PlantGrowth, main = "PlantGrowth data",

ylab = "Dried weight of plants", col = "lightgray",

notch = TRUE, varwidth = TRUE)

anova(lm(weight ~ group, data = PlantGrowth))

#Problem 4

names(PlantGrowth)

weight <- PlantGrowth$weight

weight

rank<-rank(weight)

rank

#This tells us the position of the list of weights if sorted smallest to largest. The two '3.5' elements mean there are two 4.17 weights that would be the 3rd and 4th element in the list.

sort<-sort(weight)

sort

#This is the actual list of weights from smallest to largest. You can see the 4.17 are in the third and fourth position.

weight.table = data.frame(weight,rank,sort)

weight.table

**OUTPUT**

> dat <- read.table(file.choose())

> dat

V1 V2

1 x y

2 1.2 1.5

3 2.3 2.1

4 1.5 2.0

> dat <- read.table(file.choose(), header=T)

> dat

x y

1 1.2 1.5

2 2.3 2.1

3 1.5 2.0

> names(dat)

[1] "x" "y"

> x<-dat$x

> x

[1] 1.2 2.3 1.5

> y<-dat$y

> y

[1] 1.5 2.1 2.0

> dat.2<-list(x,y)

> dat.2

[[1]]

[1] 1.2 2.3 1.5

[[2]]

[1] 1.5 2.1 2.0

> dput(dat.2,file.choose())

> rm(list = ls())

> G<-dget(file.choose())

> G

[[1]]

[1] 1.2 2.3 1.5

[[2]]

[1] 1.5 2.1 2.0

> dfG<-data.frame(G)

> dfG

c.1.2..2.3..1.5. c.1.5..2.1..2.

1 1.2 1.5

2 2.3 2.1

3 1.5 2.0

> data()

> data(PlantGrowth)

> PlantGrowth

weight group

1 4.17 ctrl

2 5.58 ctrl

3 5.18 ctrl

4 6.11 ctrl

5 4.50 ctrl

6 4.61 ctrl

7 5.17 ctrl

8 4.53 ctrl

9 5.33 ctrl

10 5.14 ctrl

11 4.81 trt1

12 4.17 trt1

13 4.41 trt1

14 3.59 trt1

15 5.87 trt1

16 3.83 trt1

17 6.03 trt1

18 4.89 trt1

19 4.32 trt1

20 4.69 trt1

21 6.31 trt2

22 5.12 trt2

23 5.54 trt2

24 5.50 trt2

25 5.37 trt2

26 5.29 trt2

27 4.92 trt2

28 6.15 trt2

29 5.80 trt2

30 5.26 trt2

> ?PlantGrowth

> require(stats); require(graphics)

> boxplot(weight ~ group, data = PlantGrowth, main = "PlantGrowth data",

+ ylab = "Dried weight of plants", col = "lightgray",

+ notch = TRUE, varwidth = TRUE)

Warning message:

In bxp(list(stats = c(4.17, 4.53, 5.155, 5.33, 6.11, 3.59, 4.17, :

some notches went outside hinges ('box'): maybe set notch=FALSE

> anova(lm(weight ~ group, data = PlantGrowth))

Analysis of Variance Table

Response: weight

Df Sum Sq Mean Sq F value Pr(>F)

group 2 3.7663 1.8832 4.8461 0.01591 \*

Residuals 27 10.4921 0.3886

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

> names(PlantGrowth)

[1] "weight" "group"

> weight <- PlantGrowth$weight

> weight

[1] 4.17 5.58 5.18 6.11 4.50 4.61 5.17 4.53 5.33 5.14 4.81 4.17 4.41 3.59 5.87 3.83 6.03 4.89 4.32 4.69

[21] 6.31 5.12 5.54 5.50 5.37 5.29 4.92 6.15 5.80 5.26

> rank<-rank(weight)

> rank

[1] 3.5 24.0 17.0 28.0 7.0 9.0 16.0 8.0 20.0 15.0 11.0 3.5 6.0 1.0 26.0 2.0 27.0 12.0 5.0 10.0

[21] 30.0 14.0 23.0 22.0 21.0 19.0 13.0 29.0 25.0 18.0

> sort<-sort(weight)

> sort

[1] 3.59 3.83 4.17 4.17 4.32 4.41 4.50 4.53 4.61 4.69 4.81 4.89 4.92 5.12 5.14 5.17 5.18 5.26 5.29 5.33

[21] 5.37 5.50 5.54 5.58 5.80 5.87 6.03 6.11 6.15 6.31

> weight.table = data.frame(weight,rank,sort)

> weight.table

weight rank sort

1 4.17 3.5 3.59

2 5.58 24.0 3.83

3 5.18 17.0 4.17

4 6.11 28.0 4.17

5 4.50 7.0 4.32

6 4.61 9.0 4.41

7 5.17 16.0 4.50

8 4.53 8.0 4.53

9 5.33 20.0 4.61

10 5.14 15.0 4.69

11 4.81 11.0 4.81

12 4.17 3.5 4.89

13 4.41 6.0 4.92

14 3.59 1.0 5.12

15 5.87 26.0 5.14

16 3.83 2.0 5.17

17 6.03 27.0 5.18

18 4.89 12.0 5.26

19 4.32 5.0 5.29

20 4.69 10.0 5.33

21 6.31 30.0 5.37

22 5.12 14.0 5.50

23 5.54 23.0 5.54

24 5.50 22.0 5.58

25 5.37 21.0 5.80

26 5.29 19.0 5.87

27 4.92 13.0 6.03

28 6.15 29.0 6.11

29 5.80 25.0 6.15

30 5.26 18.0 6.31

>

