Northeastern University
College of Science

Module 9 – Homework

Problem 1 (25 points)

On the Golub et al. (1999) data set, find the expression values for the GRO2 GRO2 oncogene and the GRO3 GRO3 oncogene. (Hint: Use grep() to find the gene rows in golub.gnames. Review module 2, or page 12 of the textbook on how to do this. Be careful to search *only in the column with gene names.*)

(a) Find the correlation between the expression values of these two genes.

(b) Find the parametric 90% confident interval for the correlation with cor.test(). (Hint: use ?cor.test to learn how to set the confidence level different from the default value of 95%.)

(c) Find the bootstrap 90% confident interval for the correlation.

Answer)

```
> # problem_1
> library("multtest")
> data(golub)

> # (a)
> GRO2<-golub[2714,]
> GRO3<-golub[2715,]
> cor(GRO2,GRO3)
[1] 0.7966283

> # (b)
> cor.test(GRO2, GRO3 , conf.level =0.90)

        Pearson's product-moment correlation

data:  GRO2 and GRO3
t = 7.9074, df = 36, p-value = 2.201e-09
alternative hypothesis: true correlation is not equal to 0
90 percent confidence interval:
 0.6702984 0.8780861
sample estimates:
      cor
```

```
0.7966283

> # (c)
> nboot <- 2000
> boot.cor<- matrix(0, nrow=nboot, ncol = 1)
> data<- cbind(GRO2,GRO3)
> for (i in 1:nboot){
+   dat.star <- data[sample(1:nrow(data), replace=TRUE),]
+   boot.cor[i,]<-cor(dat.star[,1], dat.star[,2])
+ }
> quantile(boot.cor[,1],c(0.025,0.90))
     2.5%       90%
0.5335614 0.8799417
```

Problem 2 (25 points)

On the Golub et al. (1999) data set, we consider the correlation between the Zyxin gene expression values and each of the gene in the data set.

(a) How many of the genes have correlation values less than **negative 0.5**? (Those genes are highly negatively correlated with Zyxin gene).

(b) Find the gene names for the top five genes that are most negatively correlated with Zyxin gene.

(c) Using the correlation test, how many genes are negatively correlated with the Zyxin gene? Use a false discovery rate of 0.05. (Hint: use cor.test() to get the p-values then adjust for FDR. Notice that we want a one-sided test here.)

Answer)

```
> #problem_2

> #(a)

> grep("Zyxin",golub.gnames[,2])
[1] 2124

> gene_zy <- golub[2124,]
> allgenes <- golub
> cor <- apply(allgenes, 1, function(x) cor(x, gene_zy))
```

```
> totalgenes<-sum(cor < (-0.5))
> totalgenes

[1] 85

> #(b)

> gene_5 <- (order(cor)[1:5])
> gene_5_names <- golub.gnames[gene_5,2]
> gene_5_names

[1] "Macmarcks"

[2] "Inducible protein mRNA"

[3] "C-myb gene extracted from Human (c-myb) gene, complete prim
ary cds, and five complete alternatively spliced cds"
[4] "Oncoprotein 18 (Op18) gene"

[5] "54 kDa protein mRNA"

> #(c)

> corr_test<- apply(golub, 1, function(x) cor.test(x, gene_zy))
> p.values <- sapply(corr_test, function(x) x$p.value)
> adj_val <- p.adjust(p=p.values, method="fdr")
> neg_gene <- sum(adj_val < 0.05)
> neg_gene

[1] 328
```

Problem 3 (25 points)

On the Golub et al. (1999) data set, regress the expression values for the GRO3 GRO3 oncogene on the expression values of the GRO2 GRO2 oncogene.

(a) Is there a statistically significant linear relationship between the two genes' expression? Use appropriate statistical analysis to make the conclusion. What proportion of the GRO3 GRO3 oncogene expression's variation can be explained by the regression on GRO2 GRO2 oncogene expression?

(b) Find an 80% prediction interval for the GRO3 GRO3 oncogene expression when GRO2 GRO2 oncogene is not expressed (zero expression value).

(c) Check the regression model assumptions. Can we trust the statistical inferences from the regression fit?

```
> # problem_3

> #(a)

> GRO2<-golub[2714,]
> GRO3<-golub[2715,]
> reg.fit_3<- lm(GRO3~GRO2)
> reg.fit_3

Call:
lm(formula = GRO3 ~ GRO2)

Coefficients:
(Intercept)         GRO2
    -0.8426       0.3582

> summary(reg.fit_3)

Call:
lm(formula = GRO3 ~ GRO2)

Residuals:
     Min       1Q   Median       3Q      Max
-0.78038 -0.10639 -0.00553  0.14225  0.96298

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.84256    0.05941 -14.182 2.62e-16 ***
GRO2         0.35820    0.04530   7.907 2.20e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3201 on 36 degrees of freedom
Multiple R-squared:  0.6346,    Adjusted R-squared:  0.6245
F-statistic: 62.53 on 1 and 36 DF,  p-value: 2.201e-09
```

```
> #(b)

> predict(reg.fit, data.frame(GRO2=0), interval="prediction", le
vel = 0.80)
        fit       lwr        upr
1 -0.842559 -1.267563 -0.4175553
>

> #(c)
> shapiro.test(resid(reg.fit_3))

        Shapiro-Wilk normality test

data:  resid(reg.fit_3)
W = 0.94779, p-value = 0.07532
```

- The residuals look like normally distributed because the p-value is 0.07532 and we can confirm this with qq-plot.

Problem 4 (25 points)

For this problem, work with the data set stack.loss that comes with R. You can get help on the data set with ?stackloss command. That shows you the basic information and source reference of the data set. Note: it is a data frame with four variables. The variable stack.loss contains the ammonia loss in a manufacturing (oxidation of ammonia to nitric acid) plant measured on 21 consecutive days. We try to predict it using the other three variables: air flow (Air.Flow) to the plant, cooling water inlet temperature (C) (Water.Temp), and acid concentration (Acid.Conc.)

(a) Regress stack.loss on the other three variables. What is the fitted regression equation?

(b) Do all three variables have statistical significant effect on stack.loss? What proportion of variation in stack.loss is explained by the regression on the other three variables?

Answer)
According to the output, the multiple R-squared is 0.9136, which indicates that the regression model explains for 91.36% of the variation in stack.loss. This shows that the model fits the data well and that there is a strong correlation between the predictor variables and stack.loss.
(c) Find a 90% confidence interval and 90% prediction interval for stack.loss when

Air.Flow=60, Water.Temp=20 and Acid.Conc.=90.

```
> # problem_4
> #(a)
> data("stackloss")
> stackloss_data <- data.frame(stackloss[,c('Air.Flow','Water.Te
mp','Acid.Conc.','stack.loss')])
> names(stackloss_data) <- c('Air.Flow','Water.Temp','Acid.Conc.
','stack.loss')
> reg <- lm(stack.loss ~ Air.Flow + Water.Temp + Acid.Conc., dat
a = stackloss)
> summary(reg)

Call:
lm(formula = stack.loss ~ Air.Flow + Water.Temp + Acid.Conc.,
    data = stackloss)

Residuals:
    Min      1Q  Median      3Q     Max
-7.2377 -1.7117 -0.4551  2.3614  5.6978

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -39.9197    11.8960  -3.356  0.00375 **
Air.Flow      0.7156     0.1349   5.307  5.8e-05 ***
Water.Temp    1.2953     0.3680   3.520  0.00263 **
Acid.Conc.   -0.1521     0.1563  -0.973  0.34405
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.243 on 17 degrees of freedom
Multiple R-squared:  0.9136,    Adjusted R-squared:  0.8983
F-statistic:  59.9 on 3 and 17 DF,  p-value: 3.016e-09

> # we can see that fitted regression equation is
> # stack.loss = -39+0.71Air.Flow+1.29Water.Temp-0.15Acid.Conc.



> #(b)
> #According to the output, the multiple R-squared is 0.9136, wh
ich indicates that the regression model explains for 91.36% of t
he variation in stack.loss. This shows that the model fits the
data well and that there is a strong correlation between the pre
dictor variables and stack.loss
```

```
> #(c)

> c_p_data <- data.frame(Air.Flow=60, Water.Temp=20, Acid.Conc.=
90)
> conf_interval <- predict(reg, c_p_data, interval="confidence",
 level=0.90)
> conf_interval
       fit      lwr      upr
1 15.23343 13.50069 16.96617
> pred_internal <- predict(reg, c_p_data, interval="prediction",
 level=0.90)


> pred_internal
       fit      lwr      upr
1 15.23343 9.331184 21.13568
```