Chap3

Anjali Krishnan and Richard Troise

First, set working directory. 'data' is a table with three columns and same number of rows, and should be numeric. Columns have headers indicating the names of the variables. **User will also input desired variable names in double quotes**

```
data <- read.csv("chap3.csv", header = FALSE, skip = 1)
colnames(data) = c("LearnTrial", "InterpolateList", "CorrectRecall")
```

View the data.

```
knitr::kable(xtable(data))
```

LearnTrial	InterpolateList	CorrectRecall
2	2	35
2	4	21
2	8	6
4	2	40
4	4	34
4	8	18
8	2	61
8	4	58
8	8	46
2	2	39
2	4	31
2	8	8
4	2	52
4	4	42
4	8	26
8	2	73
8	4	66
8	8	52

Calculate the mean for all columns

```
summarise_all(data,mean)
```

```
##   LearnTrial InterpolateList CorrectRecall
## 1   4.666667        4.666667      39.33333
```

Calculate the standard deviation for all columns

```
summarise_all(data,sd)
```

```
##   LearnTrial InterpolateList CorrectRecall
## 1   2.566756        2.566756      19.11882
```

Replace 'var1', 'var2', and 'var3' to an appropriate first, second, and third column name using CTRL+F.

Only check off 'Match case' to avoid overwriting additional code.

```
column = colnames(data)
colnames(data) <- c("V1", "V2", "V3")
var1 = data$V1
var2 = data$V2
var3 = data$V3
```

We now perform an orthogonal multiple regression analysis on the data

```
multi_reg1=lm(var3~var1+var2,data=data)
```

We now compute the predicted values and the residuals, then print the results on a single table. Replace 'var3', the dependent variable, to an appropriate column name using CTRL+F. **Only check off 'Match case' to avoid overwriting additional code.**

```
Y_hat=predict(multi_reg1)
Residual=round(residuals(multi_reg1),2)
knitr::kable(xtable(data.frame(Y=var3,Y_hat,Residual)))
```

Y	Y_hat	Residual
35	34	1
21	26	-5
6	10	-4
40	46	-6
34	38	-4
18	22	-4
61	70	-9
58	62	-4
46	46	0
39	34	5
31	26	5
8	10	-2
52	46	6
42	38	4
26	22	4
73	70	3
66	62	4
52	46	6

We now compute the sum of squares of the residuals, then print the result

```
SS_residual=sum(Residual^2)
print(SS_residual)
```

```
## [1] 390
```

We now compute the correlation matrix between the variables. **User will also input desired variable names in double quotes**

```
colnames(data) = c("LearnTrial", "InterpolateList", "CorrectRecall")
r_mat=cor(data)
Corr=round(r_mat,4)
knitr::kable(xtable((Corr)))
```

	LearnTrial	InterpolateList	CorrectRecall
LearnTrial	1.0000	0.000	0.8055
InterpolateList	0.0000	1.000	-0.5370
CorrectRecall	0.8055	-0.537	1.0000

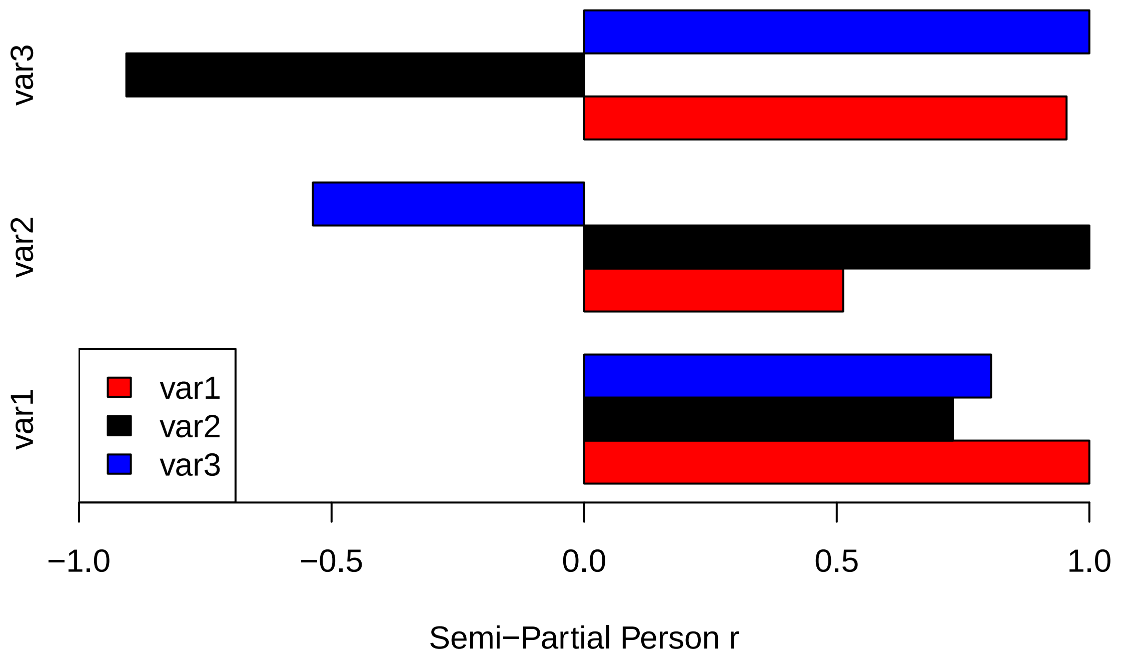
We now compute the semi-partial coefficients, then print the results. Replace ‘var1’, ‘var2’, and ‘var3’ to an appropriate first, second, and third column name using CTRL+F. **Only check off ‘Match case’ to avoid overwriting additional code.**

```
colnames(data) <- c("V1", "V2", "V3")
var1 = data$V1
var2 = data$V2
var3 = data$V3
semi_r = spcor(data)
semi_part=data.frame(var1=semi_r$estimate[3,1]^2, var2=semi_r$estimate[3,2]^2)
knitr::kable(xtable((semi_part)))
```

var1	var2
0.6488574	0.2883811

Plotting the semi-partial correlations

```
colnames(data) = c("var1", "var2", "var3")
semi_r = spcor(data)
barplot(semi_r$estimate, beside=TRUE, horiz=TRUE,
        col=c("red","black","blue"),
        xlab="Semi-Partial Person r", xlim=c(-1,1))
legend("bottomleft", legend=c("var1","var2","var3"), fill=c("red","black","blue"))
```



Multi-Regression Analysis

```
print(summary(multi_reg1))
```

```
##
## Call:
## lm(formula = var3 ~ var1 + var2, data = data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##   -9.0   -4.0    0.5    4.0    6.0
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  30.0000     3.3993   8.825 2.52e-07 ***
## var1          6.0000     0.4818  12.453 2.60e-09 ***
## var2         -4.0000     0.4818  -8.302 5.44e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.099 on 15 degrees of freedom
## Multiple R-squared:  0.9372, Adjusted R-squared:  0.9289
## F-statistic:   112 on 2 and 15 DF,  p-value: 9.609e-10
```

ANOVA analysis

```
print(anova(multi_reg1))
```

```
## Analysis of Variance Table
##
## Response: var3
##           Df Sum Sq Mean Sq F value    Pr(>F)
## var1       1   4032    4032 155.077 2.601e-09 ***
## var2       1   1792    1792  68.923 5.440e-07 ***
## Residuals 15    390      26
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```