|  |
| --- |
| [Type the company name] |
| STAT 445 Assignment 5 |
| Kun Yang 301178299 |
|  |
|  |
|  |

|  |
| --- |
|  |

***> ### 1.Use the scan () function to enter these data as a matrix.***

> Data.matrix <- matrix(scan(n=24),nrow=6, byrow=TRUE)

1: 56.8000 -124.9000 680 1129

5: 55.5333 -122.4833 680 1231

9: 53.3667 -120.2500 771 1342

13: 51.2333 -116.6667 1143 1150

17: 57.0000 -122.3667 1204 842

21: 51.3000 -116.9667 787 1514

Read 24 items

***> ### 2.Label both the rows and columns of your matrix with meaningful, short labels.***

> colnames(Data.matrix)=c('Lat','Long','Elev','GDD')

> rownames(Data.matrix)=c('IP','PP','MN','YP','PM','GO')

> Data.matrix

Lat Long Elev GDD

IP 56.8000 -124.9000 680 1129

PP 55.5333 -122.4833 680 1231

MN 53.3667 -120.2500 771 1342

YP 51.2333 -116.6667 1143 1150

PM 57.0000 -122.3667 1204 842

GO 51.3000 -116.9667 787 1514

***> ### 3.Compute the correlation matrix for these data. Which variable pairs, if any, are now highly correlated?***

> round(cor(Data.matrix),4)

Lat Long Elev GDD

Lat 1.0000 ***-0.9565*** -0.0975 -0.6848

Long ***-0.9565*** 1.0000 0.3436 0.5046

Elev -0.0975 0.3436 1.0000 -0.6260

GDD -0.6848 0.5046 -0.6260 1.0000

**> ### Latitude and Longitude are highly correlated.**

***> ### 4.What do the entries in the correlation matrix suggest about which of these variables might be related to GDD?***

> GDD = Data.matrix[,4]

> Lat = Data.matrix[,1]

> Long = Data.matrix[,2]

> Elev = Data.matrix[,3]

**> full.model.fit = lm(GDD ~ Lat + Long + Elev)**

**> summary(full.model.fit)**

Call:

lm(formula = GDD ~ Lat + Long + Elev)

Residuals:

IP PP MN YP PM GO

6.832 -46.382 33.608 -28.343 15.338 18.947

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 10015.8894 2737.9207 3.658 0.0673 .

Lat 26.0439 52.9132 0.492 0.6713

Long 77.8763 44.6999 1.742 0.2236

Elev -0.9504 0.1837 -5.172 0.0354 \*

---

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

Residual standard error: 48.61 on 2 degrees of freedom

Multiple R-squared: 0.9815, Adjusted R-squared: 0.9538

F-statistic: 35.37 on 3 and 2 DF, p-value: 0.02762

**> drop1(full.model.fit)**

Single term deletions

Model:

GDD ~ Lat + Long + Elev

Df Sum of Sq RSS AIC

<none> 4725 48.013

Lat 1 572 ***5297*** 46.699

Long 1 7171 11896 51.553

Elev 1 63203 67929 62.007

**> ### *Drop latitude.***

**> full.model.fit1 = lm(GDD ~ Long + Elev)**

**> summary(full.model.fit1)**

Call:

lm(formula = GDD ~ Long + Elev)

Residuals:

IP PP MN YP PM GO

-3.838 -37.536 27.608 -40.424 24.953 29.236

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 8740.38869 763.88813 11.442 0.00143 \*\*

Long 56.15014 6.09180 9.217 0.00270 \*\*

Elev -0.87412 0.08538 -10.238 0.00199 \*\*

---

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

Residual standard error: 42.02 on 3 degrees of freedom

Multiple R-squared: 0.9793, Adjusted R-squared: 0.9654

F-statistic: 70.82 on 2 and 3 DF, p-value: 0.002987

**> drop1(full.model.fit1)**

Single term deletions

Model:

GDD ~ Long + Elev

Df Sum of Sq RSS AIC

<none> 5297 46.699

Long 1 150021 155319 64.969

Elev 1 185072 190369 66.190

***After we dropped the Latitude (the smallest RSS), there still are significant changes in RSS. So we cannot drop explanatory variables. We should try other method.***

***> ### 5.Evaluate the eigenvalues and eigenvectors of the correlation matrix for the x-variables,latitude, longitude, and elevation.***

> eigens = eigen(cor(Data.matrix[,c(1,2,3)]))

> (evals = eigens$values)

[1] 2.05040253 0.93831452 ***0.01128295***

> (evecs = eigens$vectors)

[,1] [,2] [,3]

[1,] 0.6591615 0.33289369 0.6743055

[2,] -0.6945234 -0.07432873 0.7156203

[3,] -0.2883458 0.94003034 -0.1822078

***> ### 6.Evaluate the sum of these eigenvalues. You should have found this sum to be an integer. Why does the sum equal this particular integer?***

> sum(evals)

[1] 3

*The sum of eigenvalues equal the numbers of variables, so it is an integer.*

***> ### 7.Show that one of the standard guidelines suggests that you can ignore the minor dispersion in the direction of the smallest eigenvalue.***

***> ### 8.Describe, with reference to the components of its eigenvector, the direction of variation associated with this eigenvalue.***

> (proportion = evals/sum(evals))

[1] 0.683467509 0.312771507 0.003760983

> evecs[,3]

[1] 0.6743055 0.7156203 -0.1822078

*And there are relatively large weights for the first and the second variable with same signs. (Latitude and Longitude) So these are the directions of increasing of the first and second variables.*

*And there is a relatively small weight for the third variable with negative sign. (Elevation) There is the direction of decreasing of the third variable.*

***> ### 9. Describe, with reference to the map of the sites, the reason why there is so little variation in this direction.***

*As we can see on the map, when latitudes increase (Northward), the longitudes also increase (westward). So the sites move toward northwest as the latitudes and longitudes decrease.*

*But as the latitudes and longitudes increase, the elevations decrease. The sites move out of Rocky Mountains areas.*

***> ### 10. Compute a matrix of standardized values for the three variables, latitude, longitude, and elevation.***

> Lat.std =(Lat-mean(Lat))/sd(Lat)

> Long.std =(Long-mean(Long))/sd(Long)

> Elev.std =(Elev-mean(Elev))/sd(Elev)

> StdMatrix<-cbind(Lat.std,Long.std,Elev.std)

> StdMatrix

Lat.std Long.std Elev.std

IP 0.9908083 -1.3073347 -0.8426984

PP 0.5070615 -0.5716298 -0.8426984

MN -0.3203529 0.1082435 -0.4544171

YP -1.1350883 1.1990911 1.1328427

PM 1.0671873 -0.5361337 1.3931192

GO -1.1096159 1.1077635 -0.3861479

***> ### 11.The variance-covariance for this matrix should match something you have already calculated.***

> round(cov(StdMatrix),4)

Lat.std Long.std Elev.std

Lat.std 1.0000 -0.9565 -0.0975

Long.std -0.9565 1.0000 0.3436

Elev.std -0.0975 0.3436 1.0000

*Covariance of the standardized data equal to correlation matrix of the unstandardized data*

***> ### 12.a. For the dimensions in the product to be compatible, which matrix should appear first in the product? ### b. Should you use the matrix or eigenvectors or its transpose?***

> PC = StdMatrix %\*% evecs

> eigen(cov(PC))

$values

[1] 2.05040253 0.93831452 0.01128295

$vectors

[,1] [,2] [,3]

[1,] 1.000000e+00 0.000000e+00 -3.196002e-16

[2,] -1.345981e-16 -1.000000e+00 1.110223e-15

[3,] -3.196002e-16 -1.110223e-15 -1.000000e+00

*We should use the eigenvectors.*

***> ### 13. Do a linear regression of GDD on all three principal component vectors.***

***> ### One of the principal components appears not to be useful in helping to predict GDD.***

***> ### Which one is this? To which direction does this correspond?***

> PC1=PC[,1]

> PC2=PC[,2]

> PC3=PC[,3]

**> PC.reg <- lm(GDD ~ PC1+PC2+PC3)**

**> summary(PC.reg)**

Call:

lm(formula = GDD ~ PC1 + PC2 + PC3)

Residuals:

IP PP MN YP PM GO

6.832 -46.382 33.608 -28.343 15.338 18.947

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 1201.33 19.84 60.541 0.000273 \*\*\*

PC1 -68.49 15.18 -4.512 0.045778 \*

PC2 -205.69 22.44 -9.166 0.011694 \*

PC3 269.64 204.64 1.318 0.318326

---

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

Residual standard error: 48.61 on 2 degrees of freedom

Multiple R-squared: 0.9815, Adjusted R-squared: 0.9538

F-statistic: 35.37 on 3 and 2 DF, p-value: 0.02762

**> PC.reg1 <- lm(GDD ~ PC1+PC2)**

**> summary(PC.reg1)**

Call:

lm(formula = GDD ~ PC1 + PC2)

Residuals:

IP PP MN YP PM GO

-23.88 -23.09 18.57 -59.00 37.48 49.92

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 1201.33 22.14 54.250 1.38e-05 \*\*\*

PC1 -68.49 16.94 -4.043 0.02723 \*

PC2 -205.69 25.04 -8.214 0.00378 \*\*

---

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

Residual standard error: 54.24 on 3 degrees of freedom

Multiple R-squared: 0.9654, Adjusted R-squared: 0.9424

F-statistic: 41.91 on 2 and 3 DF, p-value: 0.006424

***The third principal component (elevation) appear not to be useful in helping to predict GDD.***

***> ### 14. Do you feel that the principal component regression has been of some value here beyond the regression analyses that you performed first?***

Principal components regression can avoid model specification error, which cause by variable deletion.