Labs: Group 3 & 4 code snippets, Heatmaps, Dimension Reduction Examples on PCA, Cross-Validation Labs

Assignment 5 Feedback

Project Updates

Check-in during the Lab

(One-on-One with the Instructor on Project Progress Updates)

Thilanka Munasinghe

Data Analytics

ITWS-4600/ITWS-6600/MATP-4450/CSCI-4960

Group 4, Lab 8, November 4th , 2022

Today…

Project Check-in during the Lab:

- Assignment 5 (Project Proposal Feedback) during the
- One-on-One session with the Instructor project progress updates during the Labs.

You need to meet with instructor and show your current progress and the development of your term project (Assignment6).

Rpart – recursive partitioning and Conditional Inference

Reminder to go over these code snippets…

group3/lab1_rpart1.R

group3/lab1_rpart2.R

group3/lab1_rpart3.R

group3/lab1_rpart4.R

Try rpart for “Rings” on the Abalone dataset

group3/lab1_ctree1.R

group3/lab1_ctree2.R

group3/lab1_ctree3.R

Today: Group 4: Cross-Validation Labs

Work on the Cross Validation code snippets …

They will NOT run as it is…

group4/lab1_cv1.R

group4/lab1_cv2.R

group4/lab1_cv3.R

...

...

...

group4/lab1_cv14.R

group4/lab1_cv15.R

…

Search before you ask! You might need to search your code errors online when you are debugging your code!

NOTE: you are allowed to work in small groups and discuss during this lab.

Cross Validation Code snippets are available (script fragments in R available) on:
https://rpi.box.com/s/b0jzxaeaqmhgx67y5jei9x1nv3d6wh11

Scripts – work through these

Reminder…

See in folder group2 and group3/
Labs

Go over the following scrips,

Lab3_ctree1.R

Lab3_ctree2.R

Lab3_ctree3.R

…..

And the remaining code snippets in group2/Lab 2 and Lab3

Search before you ask! You might need to search your code errors online when you are debugging your code!

NOTE: you are allowed to work in small groups and discuss during this lab.

Trees for the Titanic

Reminder to complete the following if you have not done yet…

After you complete this task, make sure to check-in with the TA (Show your work to the TA and get-checked)

data(Titanic)

rpart, ctree, hclust,RandomForest for:

Survived ~ .

Reminder to complete the In-Class work from the last lecture

```
# PCA with iris dataset
data("iris")
head(iris)
# creating another dataset from iris dataset that contains the columns from 1 to 4
irisdata1 <- iris[,1:4]
irisdata1
head(irisdata1)
principal_components <- princomp(irisdata1, cor = TRUE, score = TRUE)
summary(principal_components)
# in the summary you can see that it has four Principal Components it is becasue the input data has
# four different features.

# using the plot() function, we can plot the principal components.
plot(principal_components)

# plotting the principal_components using the a line in plot() functions
plot(principal_components, type = "l")

# using rhw biplot() function we can plot the components
biplot(principal_components)
```

Revisiting PCA …

- Today we will revisit the topic of Dimensionality reduction with PCA.
- In previous labs, we conducted PCA on Iris dataset.
- Today, we will use PCA on UCI Wine dataset, this dataset has higher number of variables compared to the iris dataset.
- During one of the previous labs, you learned how to use the heatmap() function in R, we will use the heatmap() function to show the correlations.

- **(Please complete the last week(s) lab if you have not done yet!)**

PCA on Wine data

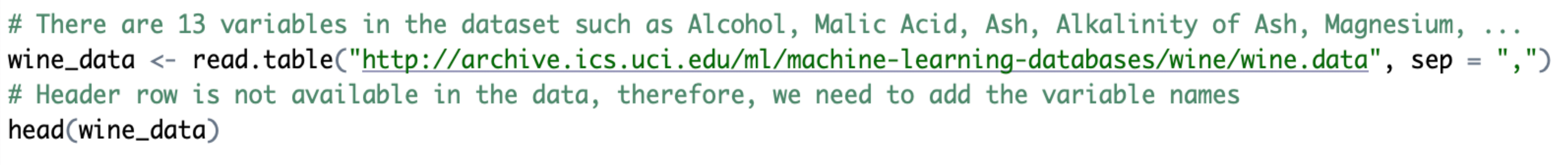
```
# PCA on Wine dataset from UCI
# Read the data using the read.table()
# Read the documentation for the UCI wine
dataset, in the documentation,
# Cvs stands for the "cultivars" (varieties) of the
class of the wine,
# cultivar are similar to wine classes Pinot
Noir,Shiraz,Muscat
# Goal is to identify the membership of the wine
in 1 of 3 cultivars.
```

There are 13 variables in the dataset such as Alcohol, Malic Acid, Ash, Alkalinity of Ash, Magnesium, ...

wine_data <- read.table("http://archive.ics.uci.edu/ml/machine-learning-databases/wine/wine.data", sep = ",")

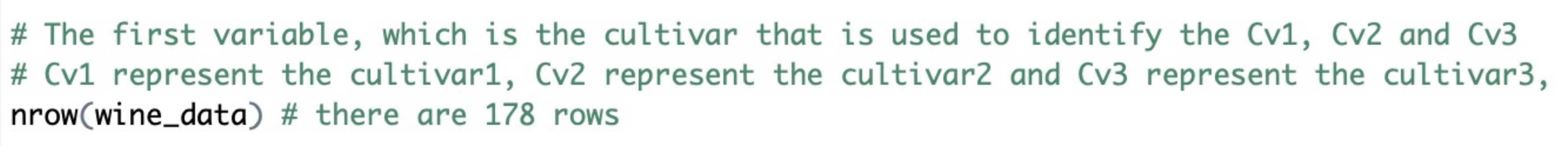
Header row is not available in the data, therefore, we need to add the variable names

head(wine_data)



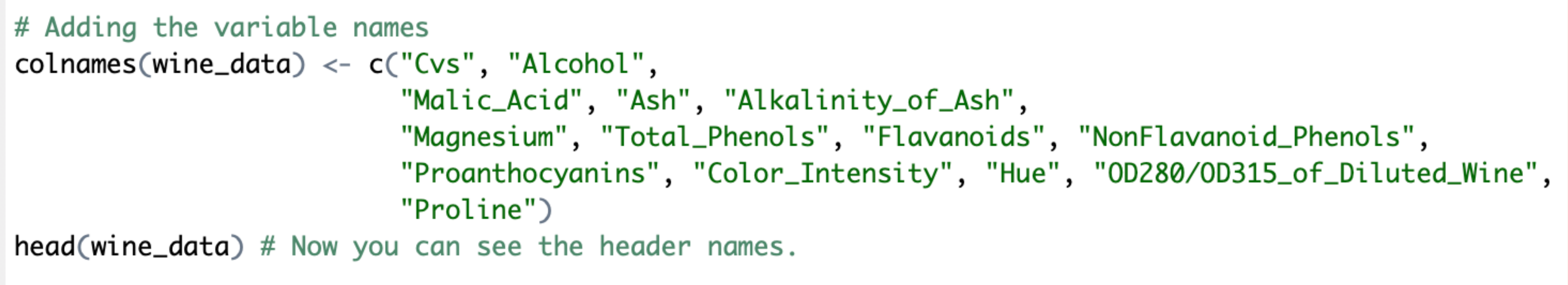
```
# There are 13 variables in the dataset such as Alcohol, Malic Acid, Ash, Alkalinity of Ash, Magnesium, ...
wine_data <- read.table("http://archive.ics.uci.edu/ml/machine-learning-databases/wine/wine.data", sep = ",")
# Header row is not available in the data, therefore, we need to add the variable names
head(wine_data)
```

```
# The first variable, which is the cultivar that is used to identify the Cv1, Cv2 and Cv3
# Cv1 represent the cultivar1, Cv2 represent the cultivar2 and Cv3 represent the cultivar3,
nrow(wine_data) # there are 178 rows
```



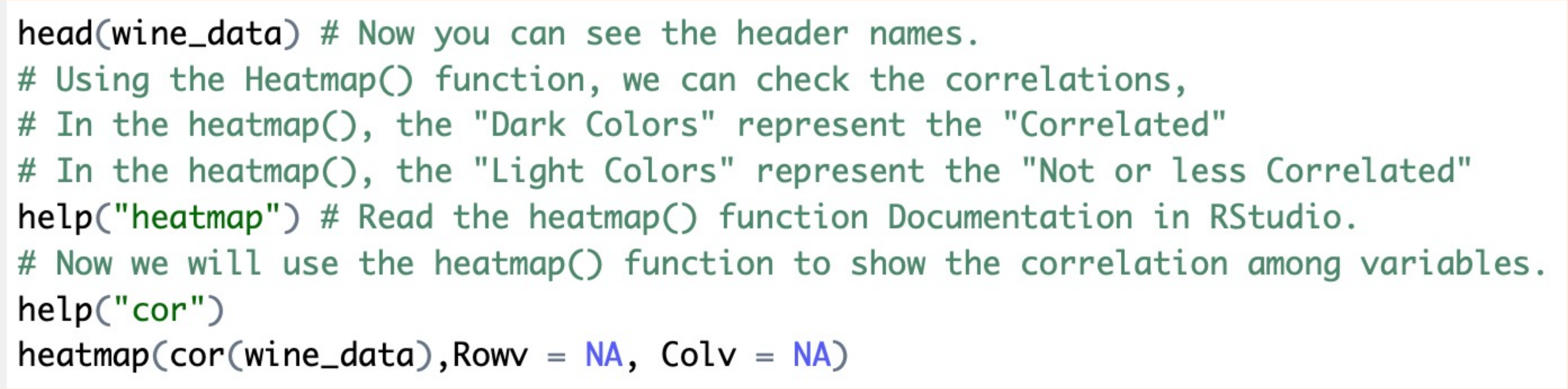
```
# The first variable, which is the cultivar that is used to identify the Cv1, Cv2 and Cv3
# Cv1 represent the cultivar1, Cv2 represent the cultivar2 and Cv3 represent the cultivar3,
nrow(wine_data) # there are 178 rows
```

```
# Adding the variable names
colnames(wine_data) <- c("Cvs", "Alcohol",
                    "Malic_Acid", "Ash", "Alkalinity_of_Ash",
                    "Magnesium", "Total_Phenols", "Flavanoids", "NonFlavanoid_Phenols",
                    "Proanthocyanins", "Color_Intensity", "Hue", "OD280/OD315_of_Diluted_Wine",
                    "Proline")
head(wine_data) # Now you can see the header names.
```



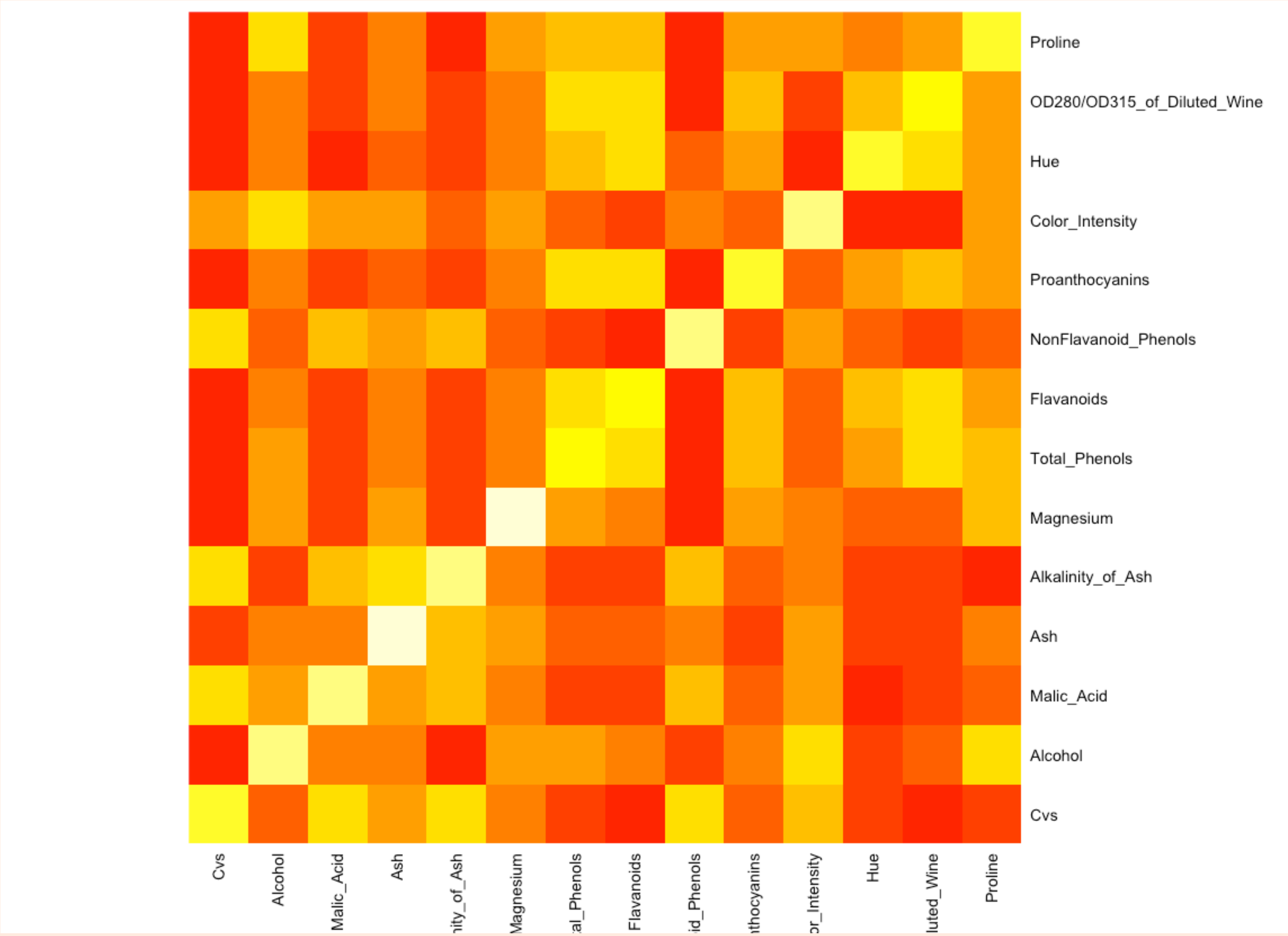
```
# Adding the variable names
colnames(wine_data) <- c("Cvs", "Alcohol",
                         "Malic_Acid", "Ash", "Alkalinity_of_Ash",
                         "Magnesium", "Total_Phenols", "Flavanoids", "NonFlavanoid_Phenols",
                         "Proanthocyanins", "Color_Intensity", "Hue", "OD280/OD315_of_Diluted_Wine",
                         "Proline")
head(wine_data) # Now you can see the header names.
```

```
# Using the Heatmap() function, we can check the correlations,
# In the heatmap(), the "Dark Colors" represent the "Correlated"
# In the heatmap(), the "Light Colors" represent the "Not or less Correlated"
help("heatmap") # Read the heatmap() function Documentation in RStudio.
# Now we will use the heatmap() function to show the correlation among variables.
heatmap(cor(wine_data),Rowv = NA, Colv = NA)
```



```
head(wine_data) # Now you can see the header names.
# Using the Heatmap() function, we can check the correlations,
# In the heatmap(), the "Dark Colors" represent the "Correlated"
# In the heatmap(), the "Light Colors" represent the "Not or less Correlated"
help("heatmap") # Read the heatmap() function Documentation in RStudio.
# Now we will use the heatmap() function to show the correlation among variables.
help("cor")
heatmap(cor(wine_data),Rowv = NA, Colv = NA)
```

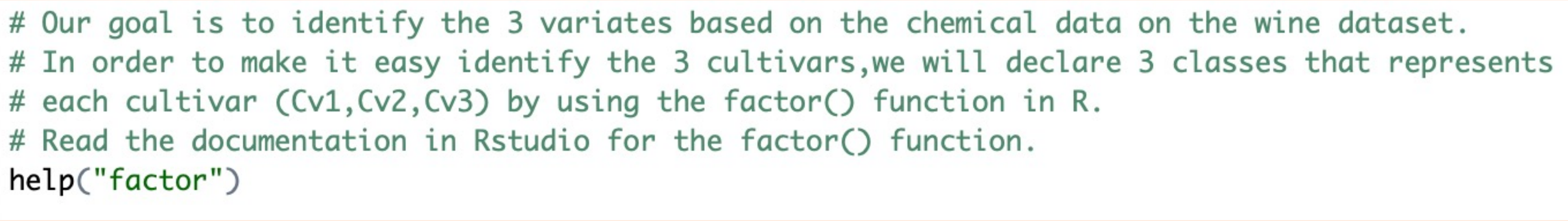
heatmap(cor(wine_data),Rowv = NA, Colv = NA)



Darker colors represents the strong correlations

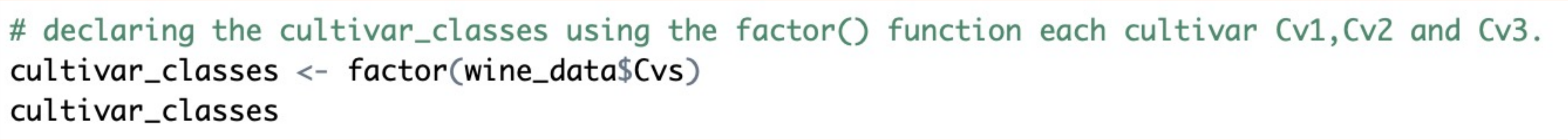
factor() function …

- Our goal is to identify the 3 variates based on the chemical data on the wine dataset.
- In order to make it easy identify the 3 cultivars, we will declare 3 classes that represents each cultivar (Cv1,Cv2,Cv3) by using the factor() function in R
- Read the documentation in Rstudio for the factor() function.
- help(factor)



```
# Our goal is to identify the 3 variates based on the chemical data on the wine dataset.
# In order to make it easy identify the 3 cultivars,we will declare 3 classes that represents
# each cultivar (Cv1,Cv2,Cv3) by using the factor() function in R.
# Read the documentation in Rstudio for the factor() function.
help("factor")
```

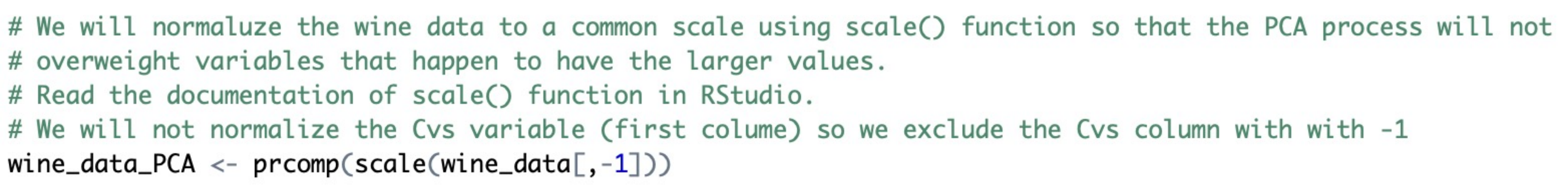
```
# declaring the cultivar_classes using the factor() function each cultivar Cv1,Cv2 and Cv3.
cultivar_classes <- factor(wine_data$Cvs)
cultivar_classes
```



```
# declaring the cultivar_classes using the factor() function each cultivar Cv1,Cv2 and Cv3.
cultivar_classes <- factor(wine_data$Cvs)
cultivar_classes
```

- Now we will use PCA,
- The default built in function in R for PCA is prcomp() function
- Read the documentation of prcomp() function in Rstudio
- Help(prcomp)

```
# We will normalize the wine data to a common scale using scale() function so that the PCA process will not
# overweight variables that happen to have the larger values.
# Read the documentation of scale() function in RStudio.
 help(scale)
# We will not normalize the Cvs variable (first colume) so we exclude the Cvs column with with -1
wine_data_PCA <- prcomp(scale(wine_data[,-1]))
```

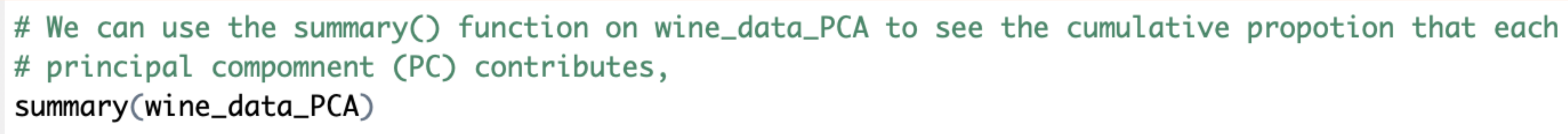


```
# We will normaluze the wine data to a common scale using scale() function so that the PCA process will not
# overweight variables that happen to have the larger values.
# Read the documentation of scale() function in RStudio.
# We will not normalize the Cvs variable (first colume) so we exclude the Cvs column with with -1
wine_data_PCA <- prcomp(scale(wine_data[,-1]))
```

We can use the summary() function on wine_data_PCA to see the cumulative proportion that each

# principal component (PC) contributes,

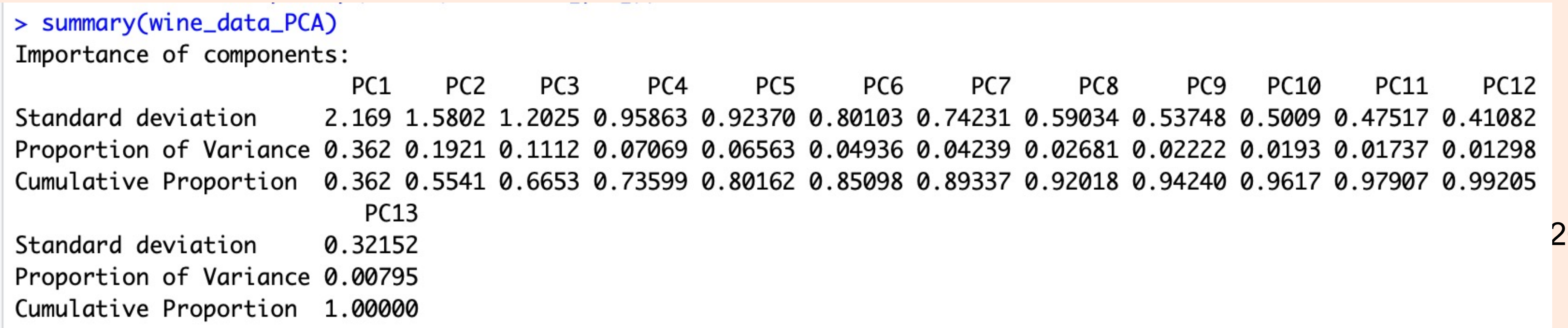
summary(wine_data_PCA)



```
# We can use the summary() function on wine_data_PCA to see the cumulative propotion that each
# principal compomnent (PC) contributes,
summary(wine_data_PCA)
```

Interpretation…

- summary(wine_data_PCA)
- We can see that PC1 gives the 36.2% cumulative contribution, which tells us that PC1 represents 36.2% variance of the data.



```
> summary(wine_data_PCA)
Importance of components:
                         PC1    PC2    PC3     PC4     PC5     PC6     PC7     PC8     PC9   PC10    PC11    PC12
Standard deviation     2.169 1.5802 1.2025 0.95863 0.92370 0.80103 0.74231 0.59034 0.53748 0.5009 0.47517 0.41082
Proportion of Variance 0.362 0.1921 0.1112 0.07069 0.06563 0.04936 0.04239 0.02681 0.02222 0.0193 0.01737 0.01298
Cumulative Proportion  0.362 0.5541 0.6653 0.73599 0.80162 0.85098 0.89337 0.92018 0.94240 0.9617 0.97907 0.99205
                          PC13
Standard deviation     0.32152
Proportion of Variance 0.00795
Cumulative Proportion  1.00000
```

Interpretation…

Principal Component (PC)	Cumulative Contribution
PC1	36.22%
PC2	55.41%
PC3	66.53%
PC4	73.59%
PC5	80.16%
PC6	85.09%
PC7	89.33%
PC8	92.08%

You can see that we can choose to have 8 variables from the total of 13 (choosing 8 out of 13) with only about 8% of loss of cumulative contribution value.

Reminder: Work through remaining code snippets in Group 2 & 3

These code snippets are available in the course repository:

(script fragments) in R available on:

https://rpi.box.com/s/2xx9ul1fmc6bf5ff8h4jreae69emikmf

Push your code to GitHub at the end of each lab, TA and I will check your code.

Next week: Support Vector Machine (SVM)

Please go over the reading material that available on our course webpage
 for Class 7 prior to next week..

Class 7 Reading Assignment available at:
https://tw.rpi.edu/classes/data-analytics-spring-2022

Next week we will go over the SVM lecture,
please go over these articles a preparation for the lecture.

https://rpi.box.com/s/2rp1gvuy2a9yjj0q4o53ox91d8jpgcuy

- https://rpi.box.com/s/nowbkb2ursqlsoj86lczmcjwjwl5tl9o
- https://rpi.box.com/s/v00ymfyy5bd4pfg165my5st3jjoz4nbu
- https://rpi.box.com/s/v00ymfyy5bd4pfg165my5st3jjoz4nbu

Reading Assignment on R-SVM

- http://www.stanford.edu/group/wonglab/RSVMpage/R-SVM.html
  - Read/ skim the paper
  - Explore this method on a dataset of your choice, e.g. one of the R built-in datasets

ggplot - line graph example

Poster Preparation…

Helpful Code Snippet to practice ggplot- line graphs…

Make sure to go over the chapter 4 of gcookbook…

```
# Chapter4_Line_Graphs_R_Graphics
library(gcookbook)
ggplot(BOD, aes(x=Time, y=demand)) + geom_line()
BOD
BOD1 <- BOD # make a copy of the dataset
BOD1$Time <- factor(BOD1$Time)
ggplot(BOD1, aes(x=Time, y=demand, group=1)) + geom_line()
ggplot(BOD, aes(x=Time, y= demand)) +geom_line() + ylim(0, max(BOD$demand))
ggplot(BOD, aes(x=Time, y=demand)) +geom_line() + expand_limits(y=0)
# Adding points to a line graph
ggplot(BOD, aes(x=Time, y=demand)) + geom_line() + geom_point()
library(gcookbook) # For the data set
ggplot(worldpop, aes(x=Year, y=Population)) + geom_line() + geom_point()
# same with log-y axis
ggplot(worldpop, aes(x=Year, y=Population)) + geom_line() + geom_point() + scale_y_log10()
```

Today…

Assignment 5 (Proposal Presentation) Feedback

Project Check-in during the Lab:

Remaining One-on-One with the Instructor on Project Progress Updates during the Labs. You Must check-in with the instructor before end of the class.

Show what you have completed so far…(Work in progress)

- You need to meet with instructor and show your current progress and the development of your term project (Assignment6).

Assignment 6 Available on LMS

- **Assignment 6 Due: Friday, 12/09/2022 by 11:59 pm ET on LMS**.
- Make sure to meet with the instructor regularly during lab times to update your progress and get the feedback.