HW 5

SDS348 Spring 2021

Name: Santhosh Saravanan

EID: sks3648

This homework is due on Mar 12, 2021 at 8am. Submit a pdf file on Gradescope.

For all questions, include the R commands/functions that you used to find your answer (show R chunk). Answers without supporting code will not receive credit. Write full sentences to describe your findings.

### Question 1: (6 pts)

#### The dataset for this homework comes from the article:

#### *Tsuzuku N, Kohno N. 2020. The oldest record of the Steller sea lion Eumetopias jubatus (Schreber, 1776) from the early Pleistocene of the North Pacific. https://doi.org/10.7717/peerj.9709 (https://doi.org/10.7717/peerj.9709)*

#### Under the supplemental information, the data was retrieved from a word document into an excel document.

##### 1.1 (4 pts) Read the ***Abstract*** of the article and the section called *Results of Morphometric Analyses*. What was the goal of this study and what was the main finding?

The goal of the study is to explain the evolutionary tract of the earliest known sea lion fossil known to man (GKZ-N 0001) and compare it to the Stellar sea lion (E.jubatus). The main finding was that the oldest known sea lion fossil has been identified as having the same biological measurements as the Steller sea lion (E.jubatus).

##### 1.2 (2 pts) Import the dataset from Excel. How many rows and how many columns are in this dataset? What does a row represent? What does a column represent?

```
library(readxl)
library(tidyverse)
HW5 <- read_excel("~/Downloads/HW5.xlsx")
glimpse(HW5)
```

```
## Rows: 51
## Columns: 39
## $ ID <chr> "E. jubatus [m]1", "E. jubatus [m]2", "E. jubatus [m]3", "E. jubatu…
## $ A  <chr> "262", "285", "265.8", "244", "237", "228", "227", "226", "282.5", …
## $ B  <chr> "232", "242", "242.2", "212", "208.98", "201.06", "202.19", "190.07…
## $ C  <chr> "62.39", "64.52", "53.06", "44.88", "39.380000000000003", "39.52000…
## $ D  <chr> "31.12", "31.71", "30.16", "26.05", "26.09", "25.39", "24.85", "27.…
## $ E  <chr> "63.12", "70.48", "70.53", "55.94", "51.21", "51.19", "48.46", "48.…
## $ F  <chr> "59.01", "75.58", "60.28", "52.04", "49.44", "48.07", "49.25", "34.…
## $ G  <chr> "43.99", "44.33", "47.98", "38.46", "37.25", "36.39", "39.049999999…
## $ H  <chr> "46.83", "62.52", "50.82", "39.89", "37.93", "37.22", "39.119999999…
## $ I  <chr> "62.56", "63.13", "61.99", "51.77", "45.6", "62.98", "48.61", "50.3…
## $ J  <chr> "62.65", "63.5", "63.89", "55.91", "49.02", "49.68", "52.58", "50.1…
## $ K  <dbl> 57.82, 64.56, 63.62, 45.21, 41.41, 43.61, 43.64, 41.85, 67.95, 44.0…
## $ L  <chr> "87.09", "97.46", "99.17", "85.05", "83.41", "76", "81.2", "75.34",…
## $ M  <chr> "24.33", "14.71", "18.36", "19.8", "24.12", "17.2", "18.94000000000…
## $ N  <chr> "88.59", "108.81", "95.43", "76.150000000000006", "73.4300000000000…
## $ O  <chr> "85.49", "87.19", "95.89", "45.19", "41.05", "57.47", "54.13", "61.…
## $ P  <chr> "222", "233", "223.8", "199.66", "197.87", "180.22", "185.41", "186…
## $ Q  <chr> "73.790000000000006", "77.95", "73.23", "63.81", "64.90000000000000…
## $ R  <chr> "76.91", "83.65", "71.03", "49.81", "55.5", "51.61", "50.29", "58.0…
## $ S  <chr> "57.82", "64.28", "58.48", "43.19", "50.3", "40.31", "37.3800000000…
## $ T  <chr> "13.73", "16.37", "15.79", "6.07", "7.28", "8.43", "7.71", "10.74",…
## $ U  <chr> "25.05", "34.93", "26.59", "21.94", "33.47", "20.48", "18.62", "23.…
## $ V  <chr> "48.64", "53.21", "44.43", "37.200000000000003", "29.34", "32.96", …
## $ W  <chr> "28.4", "28.24", "30.35", "12.8", "10.44", "10.9", "12.69", "23.36"…
## $ X  <chr> "65.63", "73.08", "72.94", "63.53", "56.08", "54.24", "57.46", "54.…
## $ Y  <chr> "47.96", "51.58", "46.82", "41.33", "38.82", "35.75", "35.700000000…
## $ Z  <chr> "79.09", "68.22", "84.57", "70.239999999999995", "63.42", "55.26", …
## $ AA <chr> "55.51", "60.37", "63.67", "42.35", "44.04", "37.11", "40.5", "38.6…
## $ AB <chr> "8.89", "10.92", "11.04", "7.04", "7", "7.48", "6.87", "8.539999999…
## $ AC <chr> "112.38", "122.34", "118.02", "108.3", "104.39", "104.15", "99", "9…
## $ AD <dbl> 69.32, 76.88, 74.37, 69.13, 70.63, 67.12, 64.76, 64.23, 68.21, 63.3…
## $ AE <chr> "17.920000000000002", "18.600000000000001", "21.37", "16.8099999999…
## $ AF <chr> "27.7", "28.91", "31.61", "26.84", "25.27", "25.46", "23.19", "24.3…
## $ AG <chr> "9.7799999999999994", "9.34", "11.72", "8.66", "11.22", "9.24", "11…
## $ AH <chr> "11.62", "11.43", "13.41", "12.83", "13.01", "11.79", "13.23", "10.…
## $ AI <chr> "12.69", "14.57", "13.91", "14.5", "14.72", "10.5", "14.04", "10.94…
## $ AJ <chr> "12.45", "13.18", "12.93", "13.03", "12.51", "13.08", "12.03", "11.…
## $ AK <chr> "11.78", "9.9600000000000009", "12.5", "10.32", "11.6", "13.13", "9…
## $ AL <chr> "7.41", "6.94", "7.93", "4.54", "5.96", "3.88", "3.39", "3.72", "12…
```

There are 51 rows and 30 columns in this dataset. The rows refer to the 50 mandibles of fur seals and sea lions chosen for comparison with GKZ-N 00001. The last row corresponds to GKZ-N 00001. The columns must be the measurements taken for the different mandibles, I presume.

### Question 2: (7 pts)

#### Before we can analyze the data, let’s clean it.

##### 2.1 (1 pt) When importing this dataset into R Studio, which variables were considered numeric? Why are

some measurements not considered as numeric?

```
library(tidyverse)
head(HW5)
```

```
## # A tibble: 6 x 39
##   ID     A     B     C     D     E     F     G     H     I     J         K L
##   <chr>  <chr> <chr> <chr> <chr> <chr> <chr> <chr> <chr> <chr> <chr> <dbl> <chr>
## 1 E. ju… 262   232   62.39 31.12 63.12 59.01 43.99 46.83 62.56 62.65  57.8 87.09
## 2 E. ju… 285   242   64.52 31.71 70.48 75.58 44.33 62.52 63.13 63.5   64.6 97.46
## 3 E. ju… 265.8 242.2 53.06 30.16 70.53 60.28 47.98 50.82 61.99 63.89  63.6 99.17
## 4 E. ju… 244   212   44.88 26.05 55.94 52.04 38.46 39.89 51.77 55.91  45.2 85.05
## 5 E. ju… 237   208.… 39.3… 26.09 51.21 49.44 37.25 37.93 45.6  49.02  41.4 83.41
## 6 E. ju… 228   201.… 39.5… 25.39 51.19 48.07 36.39 37.22 62.98 49.68  43.6 76
## # … with 26 more variables: M <chr>, N <chr>, O <chr>, P <chr>, Q <chr>,
## #   R <chr>, S <chr>, T <chr>, U <chr>, V <chr>, W <chr>, X <chr>, Y <chr>,
## #   Z <chr>, AA <chr>, AB <chr>, AC <chr>, AD <dbl>, AE <chr>, AF <chr>,
## #   AG <chr>, AH <chr>, AI <chr>, AJ <chr>, AK <chr>, AL <chr>
```

```
glimpse(HW5)
```

```
## Rows: 51
## Columns: 39
## $ ID <chr> "E. jubatus [m]1", "E. jubatus [m]2", "E. jubatus [m]3", "E. jubatu…
## $ A  <chr> "262", "285", "265.8", "244", "237", "228", "227", "226", "282.5", …
## $ B  <chr> "232", "242", "242.2", "212", "208.98", "201.06", "202.19", "190.07…
## $ C  <chr> "62.39", "64.52", "53.06", "44.88", "39.380000000000003", "39.52000…
## $ D  <chr> "31.12", "31.71", "30.16", "26.05", "26.09", "25.39", "24.85", "27.…
## $ E  <chr> "63.12", "70.48", "70.53", "55.94", "51.21", "51.19", "48.46", "48.…
## $ F  <chr> "59.01", "75.58", "60.28", "52.04", "49.44", "48.07", "49.25", "34.…
## $ G  <chr> "43.99", "44.33", "47.98", "38.46", "37.25", "36.39", "39.049999999…
## $ H  <chr> "46.83", "62.52", "50.82", "39.89", "37.93", "37.22", "39.119999999…
## $ I  <chr> "62.56", "63.13", "61.99", "51.77", "45.6", "62.98", "48.61", "50.3…
## $ J  <chr> "62.65", "63.5", "63.89", "55.91", "49.02", "49.68", "52.58", "50.1…
## $ K  <dbl> 57.82, 64.56, 63.62, 45.21, 41.41, 43.61, 43.64, 41.85, 67.95, 44.0…
## $ L  <chr> "87.09", "97.46", "99.17", "85.05", "83.41", "76", "81.2", "75.34",…
## $ M  <chr> "24.33", "14.71", "18.36", "19.8", "24.12", "17.2", "18.94000000000…
## $ N  <chr> "88.59", "108.81", "95.43", "76.150000000000006", "73.4300000000000…
## $ O  <chr> "85.49", "87.19", "95.89", "45.19", "41.05", "57.47", "54.13", "61.…
## $ P  <chr> "222", "233", "223.8", "199.66", "197.87", "180.22", "185.41", "186…
## $ Q  <chr> "73.790000000000006", "77.95", "73.23", "63.81", "64.90000000000000…
## $ R  <chr> "76.91", "83.65", "71.03", "49.81", "55.5", "51.61", "50.29", "58.0…
## $ S  <chr> "57.82", "64.28", "58.48", "43.19", "50.3", "40.31", "37.3800000000…
## $ T  <chr> "13.73", "16.37", "15.79", "6.07", "7.28", "8.43", "7.71", "10.74",…
## $ U  <chr> "25.05", "34.93", "26.59", "21.94", "33.47", "20.48", "18.62", "23.…
## $ V  <chr> "48.64", "53.21", "44.43", "37.200000000000003", "29.34", "32.96", …
## $ W  <chr> "28.4", "28.24", "30.35", "12.8", "10.44", "10.9", "12.69", "23.36"…
## $ X  <chr> "65.63", "73.08", "72.94", "63.53", "56.08", "54.24", "57.46", "54.…
## $ Y  <chr> "47.96", "51.58", "46.82", "41.33", "38.82", "35.75", "35.700000000…
## $ Z  <chr> "79.09", "68.22", "84.57", "70.239999999999995", "63.42", "55.26", …
## $ AA <chr> "55.51", "60.37", "63.67", "42.35", "44.04", "37.11", "40.5", "38.6…
## $ AB <chr> "8.89", "10.92", "11.04", "7.04", "7", "7.48", "6.87", "8.539999999…
## $ AC <chr> "112.38", "122.34", "118.02", "108.3", "104.39", "104.15", "99", "9…
## $ AD <dbl> 69.32, 76.88, 74.37, 69.13, 70.63, 67.12, 64.76, 64.23, 68.21, 63.3…
## $ AE <chr> "17.920000000000002", "18.600000000000001", "21.37", "16.8099999999…
## $ AF <chr> "27.7", "28.91", "31.61", "26.84", "25.27", "25.46", "23.19", "24.3…
## $ AG <chr> "9.7799999999999994", "9.34", "11.72", "8.66", "11.22", "9.24", "11…
## $ AH <chr> "11.62", "11.43", "13.41", "12.83", "13.01", "11.79", "13.23", "10.…
## $ AI <chr> "12.69", "14.57", "13.91", "14.5", "14.72", "10.5", "14.04", "10.94…
## $ AJ <chr> "12.45", "13.18", "12.93", "13.03", "12.51", "13.08", "12.03", "11.…
## $ AK <chr> "11.78", "9.9600000000000009", "12.5", "10.32", "11.6", "13.13", "9…
## $ AL <chr> "7.41", "6.94", "7.93", "4.54", "5.96", "3.88", "3.39", "3.72", "12…
```

None of the variables are numeric. Some measurements are not considered numeric simply because the dataset may have been created as having string measurements. The measurements may have been originally keyed as characters, so we'll have to blame the people responsible for the recording of the measurements :).

2.2 (2 pts) Using `dplyr` functions, replace all "-" in the dataset by missing values *NA* then make sure all measurements are defined as numeric variables. What is the mean rostral tip of mandible C?

```
# your code goes here (make sure to add comments)
#HW5 = sapply(HW5, as.numeric )
HW5_clean = na_if(HW5,"-")
HW5_clean[-1] <- lapply(HW5_clean[-1], as.numeric) # convert all columns except the first one to numeric. we want to preserve the id's
print(mean(HW5_clean$C,na.rm = TRUE))
```

```
## [1] 34.86622
```

The mean rostral tip of mandible C is 34.86622mm.

##### 2.3 (2 pts) Using `dplyr` functions, only keep numeric variables that are not missing for the fossil specimen GKZ-N 00001 (hint: you can use `select_if()` on the condition that `HW5_clean[51,]` has *no* missing value with `is.na()`). Then remove the rest of the missing values. How many columns and how many rows are remaining in this dataset?

```
# your code goes here (make sure to add comments)
chungus <- HW5_clean %>% select_if(!is.na(HW5_clean[51,])) %>% drop_na()
nrow(chungus)
```

```
## [1] 42
```

```
ncol(chungus)
```

```
## [1] 23
```

There are 42 rows and 23 columns in this dataset.

##### 2.4 (2 pts) Using `dplyr` functions, only keep numeric variables and scale (also called standardize) each numeric variable. What should the mean of the scaled variable of the rostral tip of mandible C be?

```
chungus = chungus %>% mutate_if(is.numeric,scale)
print(mean(chungus$C))
```

```
## [1] 1.487009e-16
```

The mean of the scaled variable of the rostral tip of mandible C should be 1.487009e-16mm.

### Question 3: (6 pts)

#### Let's now perform PCA on the measurements available for the fossil specimen GKZ-N 00001.

##### 3.1 (2 pts) Using the function `prcomp()` , calculate the principal components (PCs) for the dataset obtained in Question 2.4. Find the percentage of explained variance for each PC. What is the cumulative percentage of explained variance for PC1 and PC2?

```
chungus$ID <- NULL
chungusPCA <- chungus %>% scale() %>% prcomp()
names(chungusPCA)
```

```
## [1] "sdev"     "rotation" "center"   "scale"    "x"
```

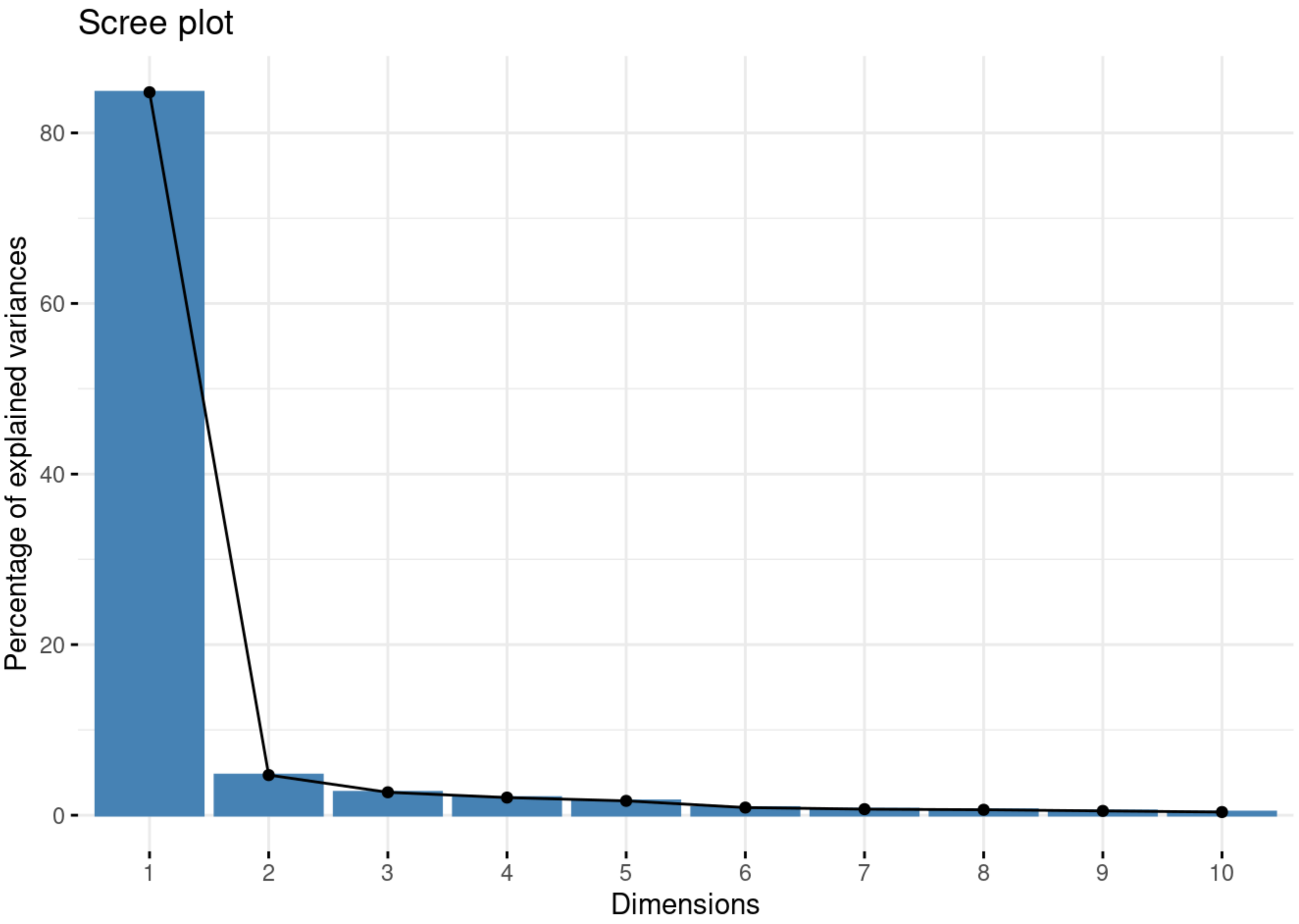
```
percent <- 100*(chungusPCA$sdev^2/sum(chungusPCA$sdev^2))
print(percent[1] + percent[2])
```

```
## [1] 89.46766
```

89.46766 is the cumulative percentage of explained variance for PC1 and PC2.

##### 3.2 (1 pt) Construct a scree plot using the package `factoextra` with the function `fviz_screeplot` and determine how many principal components should be considered.

```
library(factoextra)
fviz_screeplot(chungusPCA)
```



```
get_eig(chungusPCA)
```

```
##         eigenvalue variance.percent cumulative.variance.percent
## Dim.1  18.647912272     84.763237599                    84.76324
## Dim.2   1.034973770      4.704426229                    89.46766
## Dim.3   0.590645642      2.684752919                    92.15242
## Dim.4   0.454292083      2.064964013                    94.21738
## Dim.5   0.369099357      1.677724352                    95.89511
## Dim.6   0.196259283      0.892087650                    96.78719
## Dim.7   0.155109806      0.705044571                    97.49224
## Dim.8   0.139317998      0.633263627                    98.12550
## Dim.9   0.109640317      0.498365075                    98.62387
## Dim.10  0.080787829      0.367217403                    98.99108
## Dim.11  0.043901082      0.199550372                    99.19063
## Dim.12  0.042899138      0.194996080                    99.38563
## Dim.13  0.035196132      0.159982417                    99.54561
## Dim.14  0.021885421      0.099479186                    99.64509
## Dim.15  0.019607726      0.089126030                    99.73422
## Dim.16  0.014336052      0.065163874                    99.79938
## Dim.17  0.013353800      0.060699091                    99.86008
## Dim.18  0.010029168      0.045587127                    99.90567
## Dim.19  0.008812992      0.040059054                    99.94573
## Dim.20  0.006808475      0.030947613                    99.97667
## Dim.21  0.003145235      0.014296522                    99.99097
## Dim.22  0.001986422      0.009029193                   100.00000
```

2 principal components should be considered, as they both have eigenvalues greater than 1 and 1 and 2 form the crux of the elbow. Also, the cumulative proportion of variance is greater than 80%. PC1 and PC2 satisfy all three conditions of Kaiser’s rule.

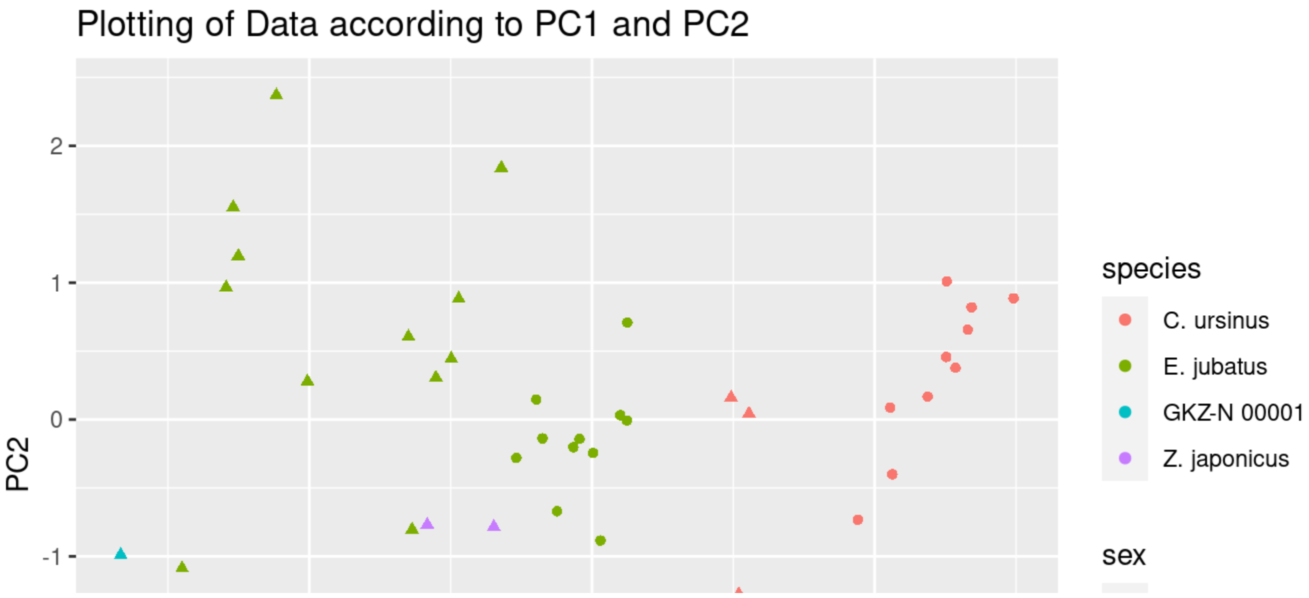
##### 3.3 (2 pts) Consider the matrix, x, of new data provided by the PCA, save it as a data frame, and add the ID variable from the dataset created in Question 2.3. Next, use the `ID` variable to create two variables `species` and `sex` by using the function `separate()` (hint: in the `ID` variable, what symbol separates the species from sex?).Note: you should get a warning because the fossil specimen is in a different format. The warning will not prevent your code from working. Finally, the article states that the fossil specimen has to be male. Replace the missing value of sex for the fossil specimen GKZ-N 00001 (hint: use the functions `mutate()` and `replace_na()`).

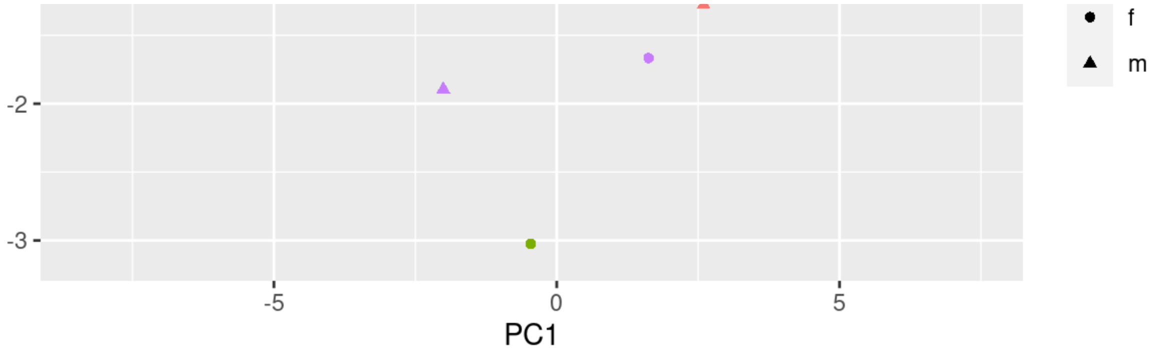
```
chungusPCAframe <- as.data.frame(chungusPCA$x)
chungus <- HW5_clean %>% select_if(!is.na(HW5_clean[51,])) %>% drop_na()
chungusPCAframe$ID <- chungus$ID
col_idx <- grep("ID", names(chungusPCAframe))
chungusPCAframe <- chungusPCAframe[, c(col_idx, (1:ncol(chungusPCAframe))[-col_idx])]
chungaIntermediateFrame <- chungusPCAframe %>% separate(ID, c('species', 'sex'), sep
="\\[|\\]")
chungaFinishedFrame <- chungaIntermediateFrame %>% mutate(sex= replace_na(sex, 'm'))
glimpse(chungaFinishedFrame)
```

```
## Rows: 42
## Columns: 24
## $ species <chr> "E. jubatus ", "E. jubatus ", "E. jubatus ", "E. jubatus ", "E…
## $ sex     <chr> "m", "m", "m", "m", "m", "m", "m", "m", "m", "m", "m", "m", "f…
## $ PC1     <dbl> -5.03223451, -5.58213261, -6.25543939, -3.24600375, -3.1806488…
## $ PC2     <dbl> 0.277943843, 2.369927479, 1.195698066, 0.605277959, -0.8040182…
## $ PC3     <dbl> -1.1718231, -0.2417352, -0.2735351, 0.2322060, 0.2592845, 0.20…
## $ PC4     <dbl> 0.37750592, 0.32460105, 0.30097950, -0.21757041, -0.45749756, …
## $ PC5     <dbl> 0.85505139, 0.69255798, 0.15303234, 0.70765225, 0.36323162, -0…
## $ PC6     <dbl> -0.1673534096, -0.4404413270, -0.0002656182, 0.4362657824, 0.8…
## $ PC7     <dbl> -0.27545293, 0.62913603, -0.69285074, -0.48256072, -0.38940535…
## $ PC8     <dbl> 0.049597467, 0.115464926, 0.035085699, -0.314346077, 0.0486537…
## $ PC9     <dbl> 0.30707650, -0.07666667, -0.78652761, 0.58035787, -0.13657382,…
## $ PC10    <dbl> -0.48388236, -0.39146522, -0.12059685, -0.05914516, -0.3505136…
## $ PC11    <dbl> 0.260337614, 0.099069746, 0.170380227, -0.371881862, -0.001305…
## $ PC12    <dbl> -0.521920138, 0.171488727, -0.126358906, 0.543441744, 0.243862…
## $ PC13    <dbl> 0.21587613, -0.13094477, -0.36738938, 0.23876617, -0.05245534,…
## $ PC14    <dbl> -0.09929164, 0.05502252, 0.21730143, -0.11015870, -0.22500510,…
## $ PC15    <dbl> -0.09811109, -0.32657352, 0.07854900, 0.03071431, 0.12620344, …
## $ PC16    <dbl> 0.27979899, -0.21953657, 0.02090820, -0.01629767, -0.02645930,…
## $ PC17    <dbl> -0.02437626, -0.23380003, 0.11311273, -0.08599059, 0.04567910,…
## $ PC18    <dbl> 2.894627e-02, 1.974919e-01, 2.486631e-03, 1.126851e-02, 9.6937…
## $ PC19    <dbl> 0.0635297777, -0.0001832198, 0.1162798417, -0.0030663753, -0.1…
## $ PC20    <dbl> 0.037829252, -0.101550399, -0.052932129, -0.029885699, -0.1207…
## $ PC21    <dbl> -5.068427e-02, 7.488273e-02, 2.071424e-02, -1.231091e-02, 5.95…
## $ PC22    <dbl> 0.034533186, 0.025206261, 0.013885209, 0.002352690, 0.06787679…
```

3.4 (1 pt) Using `ggplot` and the dataset created in the previous question, represent the observations along the new variables PC1 and PC2. In the aesthetics, color the observations by their species and shape the observations by their sex. The fossil specimen GKZ-N 00001 appears to be close to which species?

```
ggplot(chungaFinishedFrame, aes(x = PC1, y = PC2, color = species, shape=sex)) + geom
_point() + ggtitle("Plotting of Data according to PC1 and PC2")
```





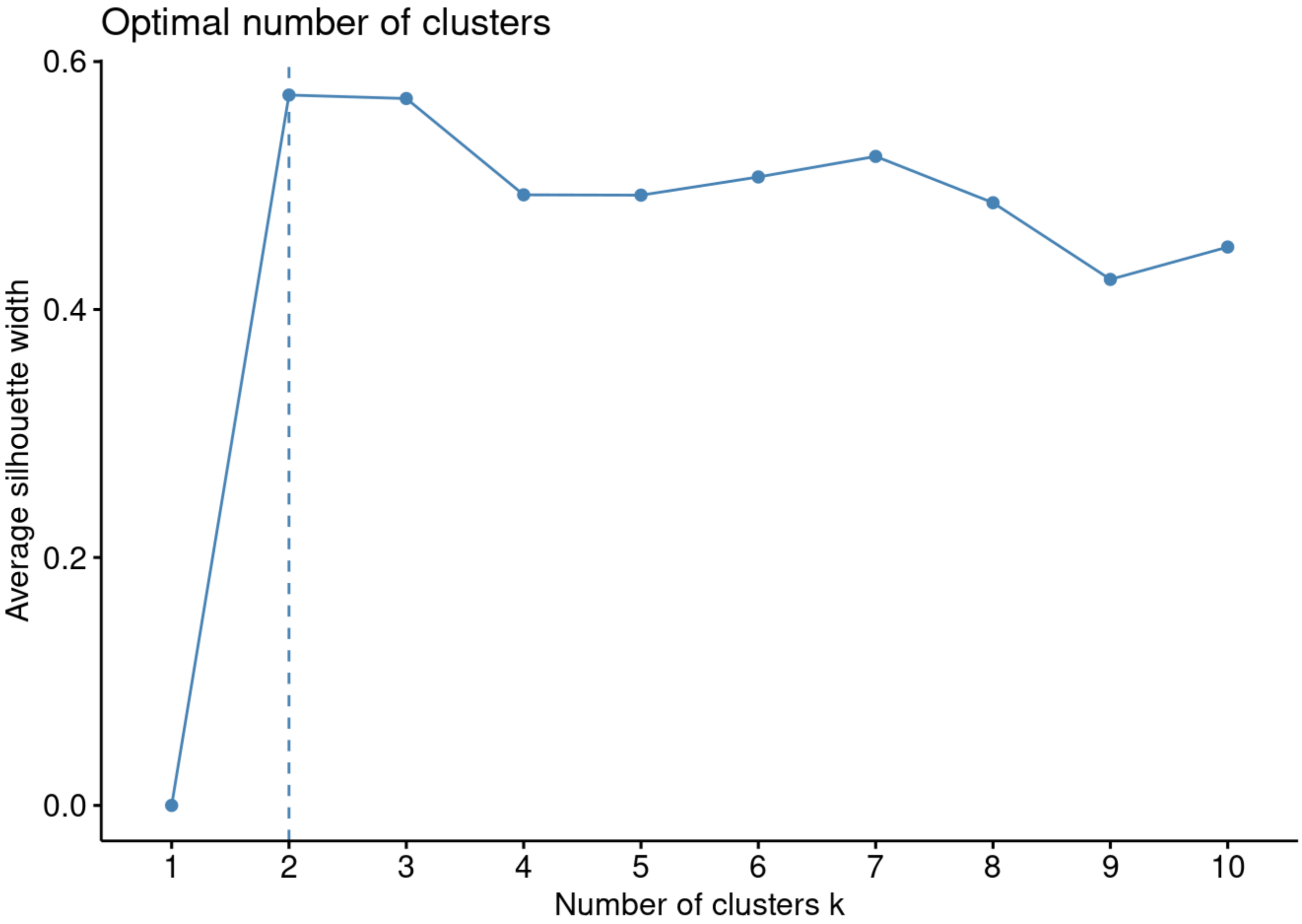
The fossil specimen GKZ-N 00001 appears to be close to a male E.jubatus species!

### Question 4: (6 pts)

#### Let’s now perform the partition around medoids (PAM) algorithm on the new variables to identify clusters of sea lions and determine which cluster the fossil specimen GKZ-N 00001 is likely to belong to.

##### 4.1 (2 pts) Using the function `pam()` from the library `cluster`, perform the PAM algorithm on the dataset obtained in Question 3.3. Make sure to only select the variables `PC1` and `PC2`. Add the identification of the cluster number to the dataset from Question 3.3 (hint: use one of the elements created through the PAM algorithm). How many clusters are we looking for if the goal is to (hopefully) recover the different species?

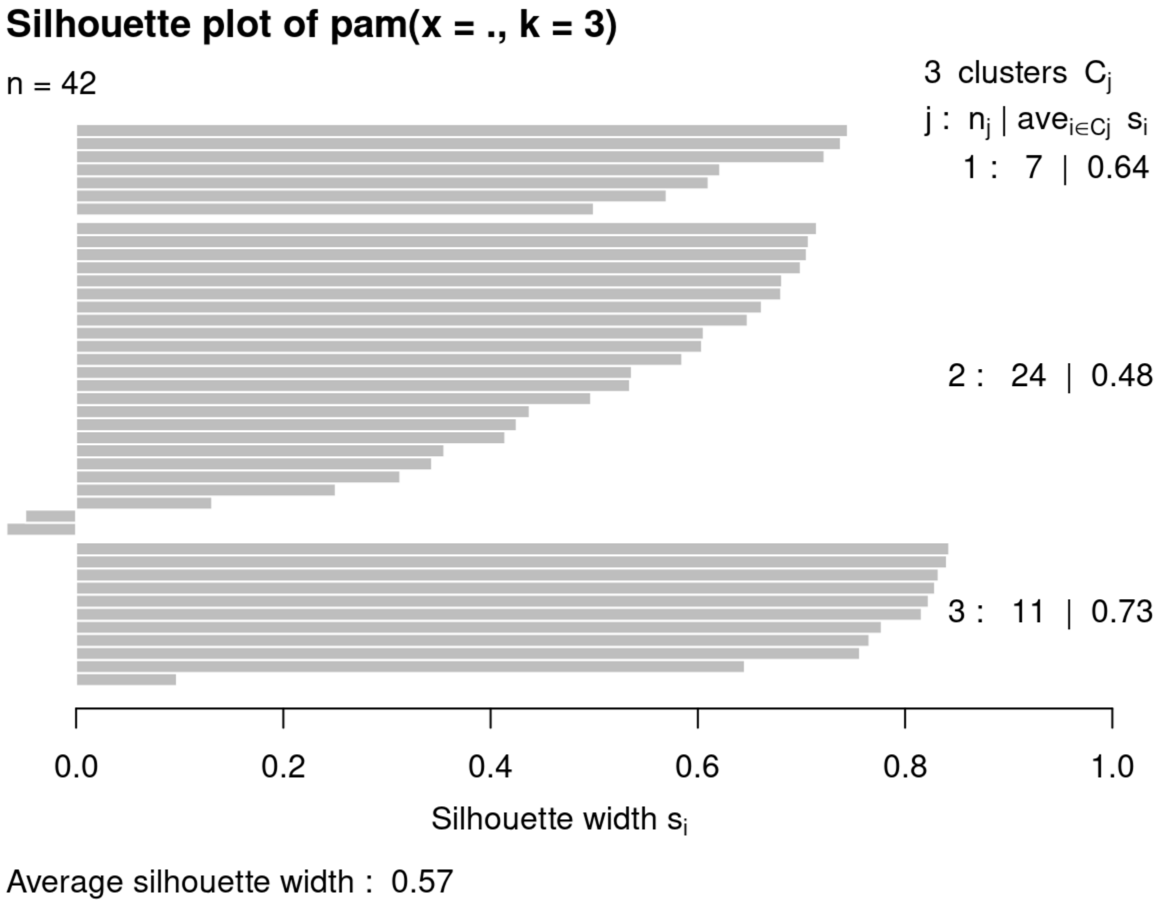
```
library(factoextra)
library(cluster)
fviz_nbclust(chungaFinishedFrame%>%select(3,4), FUNcluster = pam, method = "s")
```



```
pam1 <- chungaFinishedFrame%>%select(3,4) %>%
  pam(k=3)
pam1
```

```
## Medoids:
##      ID        PC1        PC2
## [1,] 12 -6.4705023  0.9650921
## [2,] 24 -0.8741844 -0.1380075
## [3,] 29  6.2656130  0.4571373
## Clustering vector:
##  [1] 1 1 1 2 2 2 2 2 1 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 3 2 3 3 3 3 3 3 3 3 3 3 2
## [39] 2 2 2 1
## Objective function:
##    build     swap
## 1.437508 1.435291
##
## Available components:
##  [1] "medoids"    "id.med"     "clustering" "objective"  "isolation"
##  [6] "clusinfo"   "silinfo"    "diss"       "call"       "data"
```

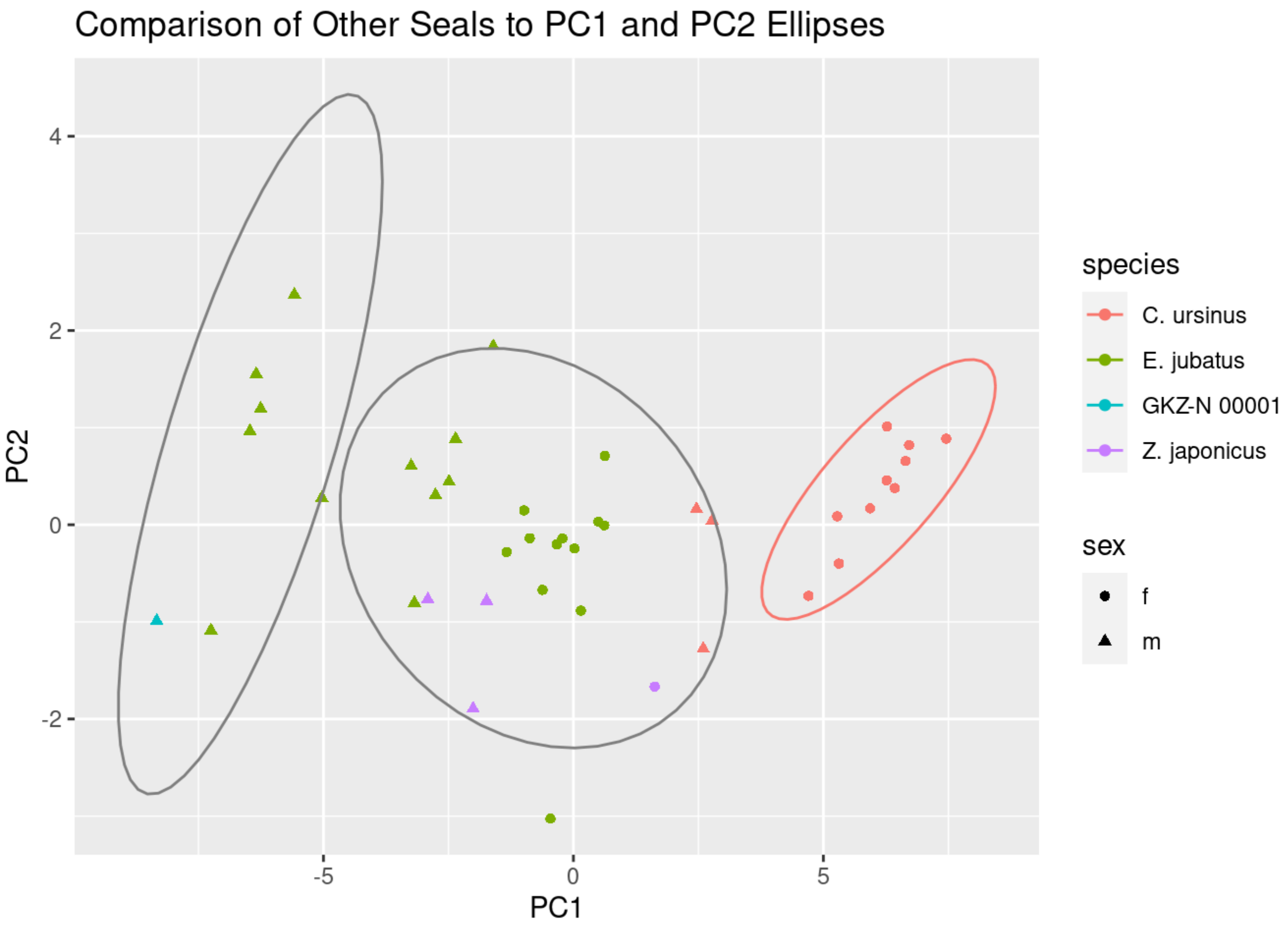
```
plot(pam1,which=2)
```



From this graph, it's quite obvious that we need 3 clusters if the goal is to recover the different species. The *silhouette width is also much greater than 0.26-0.50, where the structure is weak and could be artificial. The average silhouette width is 0.57, which indicates that a reasonable structure has been found. I picked 3 clusters because there wasn't a large difference between the optimal number of clusters (2 and 3).*

##### 4.2 (2 pts) Using `ggplot` and the dataset created in the previous question, let's create a scatterplot with the variables PC1 and PC2 to visualize the groups of species/sex and the clusters. In `geom_point`, specify the aesthetics of coloring by `species` and shaping by `sex`. Then add a layer called `stat_ellipse()` with the aesthetic of group by `cluster`. In the cluster containing the fossil specimen GKZ-N 00001, what species and sex are the other sea lions? What can you conclude about the species and sex of the fossil specimen GKZ-N 00001?

```
ggplot(chungaFinishedFrame, aes(x = PC1, y = PC2,color =species, shape=sex)) + geom_p
oint() + stat_ellipse(aes(group=pam1$clustering)) + ggtitle("Comparison of Other Seal
s to PC1 and PC2 Ellipses")
```



In the cluster containing the fossil specimen GKZ-N 00001, the other species and sex are male E.jubatus seals. The fossil specimen GKZ-N 00001 is closely related to the fossils of male E.jubatus seals.

##### 4.3 (2 pts) Putting it all together. Reflect on and summarize in 1-2 sentences the different steps taken through this assignment. Compare your conclusions to the findings discussed by the researchers in the article (cite their findings).

The steps taken in this assignments include determining the optimal number of clusters through eigenvalues, cumulative proportion of variation, and forming the crux of the elbow through Kaiser's rule. Next, you then run the PAM algorithm and confirm the number of clusters and change, if necessary. Then, find the number of *clusters and find what mystery fossils is most related to what other fossils. The conclusion I got seemed to match the findings by the researchers in the article.*

```
##                                       sysname 
##                                       "Linux" 
##                                       release 
##                            "5.4.0-66-generic" 
##                                       version 
## "#74-Ubuntu SMP Wed Jan 27 22:54:38 UTC 2021" 
##                                      nodename 
##                        "MechaChungus-linux64" 
##                                       machine 
##                                      "x86_64" 
##                                         login 
##                               "OmniLordSanta" 
##                                          user 
##                               "OmniLordSanta" 
##                                effective_user 
##                               "OmniLordSanta"
```