Introduction to multivariate distributions

Module I

David Raj Micheal

August 2018

Contents

This course materials contains is just a support to the lectures. This material will also explain the R codes which has been used to prepared this material. So, students are highly recommended to read the text books listed here and not to depend only on this material.

Text Books

- Rencher AC. Methods of multivariate analysis. John Wiley & Sons; 2003.
- Raykov T, Marcoulides GA. An introduction to applied multivariate analysis. Routledge; 2012
- Anderson TW. An introduction to multivariate statistical analysis. Wiley Eastern Private Limited; 1974
- Johnson RA, Wichern DW. Applied multivariate statistical analysis. New Jersey: Prentice-Hall; 2014.

1 What is multivariate data?

Consider the following data.

```
set.seed(1)
data = data.frame('Name' = c('Oliver', 'Noah', 'James','Benjamin','Logan'),
                  'Age(in yrs)' = sample(18:30, size = 5, replace = TRUE),
                  'Height in cm' = sample(160:180, 5),
                  'Weight in kg' = sample(45:70, 5)
                  )
data
```

Table 1: Example Data

Name	Age.in.yrs.	Height.in.cm	Weight.in.kg
Oliver	21	178	50
Noah	22	180	49
James	25	172	61
Benjamin	29	171	53
Logan	20	161	68
From the ab	ove data,		

- What does the first row tell us?

Yes. its Oliver, 21 years old with height 178cm and with weight 50kg.

Note that, the column headers 'Name', 'Age.in.yrs', 'Height.in.cm' and 'Weight.in.kg' are all random variables and together it represent some information about the observations. That is, the first observation can be seen as,

$$\begin{pmatrix} \text{Oliver} \\ 21 \\ 178 \\ 50 \end{pmatrix}$$

The above vector tells about a person's name, age, height and weight respectively. This setup is called **multivariate setup**. Note that, the headers of the above table, can be viewed as,

$$\begin{pmatrix} \text{Name} \\ \text{Age.in.yrs} \\ \text{Height.in.cm} \\ \text{Weight.in.kg} \end{pmatrix}$$

which is called as **Random Vector**.

1.1 Example of Multivariate Data.

Let us consider the following data from an experiment on the effect of diet on early growth of chicks. The body weights of the chicks were measured at birth and every second day thereafter until day 20. They were also measured on day 21. There were four groups on chicks on different protein diets. The description of the data:

- weight : a numeric vector giving the body weight of the chick (gm).
- Time : a numeric vector giving the number of days since birth when the measurement was made.
- Chick: an ordered factor with levels $18 < \cdots < 48$ giving a unique identifier for the chick. The ordering of the levels groups chicks on the same diet together and orders them according to their final weight (lightest to heaviest) within diet.
- Diet: a factor with levels $1, \ldots, 4$ indicating which experimental diet the chick received.

```
library(datasets)
chick_data = data.frame(ChickWeight)#Save the data as data.frame
head(chick_data) #To print only first 6 observations.
```

Table 2: Effect of diet on early growth of chicks.

weight	Time	Chick	Diet
42	0	1	1
51	2	1	1
59	4	1	1
64	6	1	1
76	8	1	1
93	10	1	1

Here each chick is identified by 4 different quantities such as its weight, and the number of days from its birth(Time), type of chick and the Diet. For example, the first chick can be viewed as,

$$\begin{pmatrix} 42 \\ 0 \\ 1 \\ 1 \end{pmatrix}.$$

In this case, the random vector is

$$\begin{pmatrix} \text{Weight} \\ \text{Time} \\ \text{Chick} \\ \text{Diet} \end{pmatrix}.$$

1.2 Visualizing multivarite data

There are many methods to visualize a multivariate data. We see some of them in this section.

1.2.1 Scatter Plots

2D Scatter Plot

When you have a bivariate data, you can easily visualize the relationship between the two variables by plotting a simple scatter plot. Consider the cross sectional data collected given below.

```
library(readxl)
CSdata = read_excel('crossSecData.xlsx')
head(CSdata)
```

Table 3: Cross Sectional Data

Id	Diabetes	Fat_intake	Height	Weight	SBP	DBP
1	0	25	162	57	134	90
2	0	150	151	102	110	70
3	0	34	150	56	124	70
4	0	24	150	59	130	80
5	0	50	156	53	126	80
6	0	50	180	63	120	80

The data is self explanatory with 1814 observations. The relationship between the variable 'height' and the variable 'SBP' can be plotted as follows.

```
library("lattice")
par(mfrow = c(1,2))
p1 =xyplot(Height~SBP,
       data = CSdata,
       xlab = "Height",
       ylab="SBP", main ="Scatter Plot") #Remember that plot(Height, SBP) does the same job.
p2 = xyplot(Height ~ SBP,
       group = Diabetes,
       data = CSdata,
       auto.key = TRUE,main ="Scatter Plot with Diabetes")
print(p1, split = c(1,1,2,1), more = TRUE)
print(p2, split = c(2,1,2,1),more = FALSE)
```

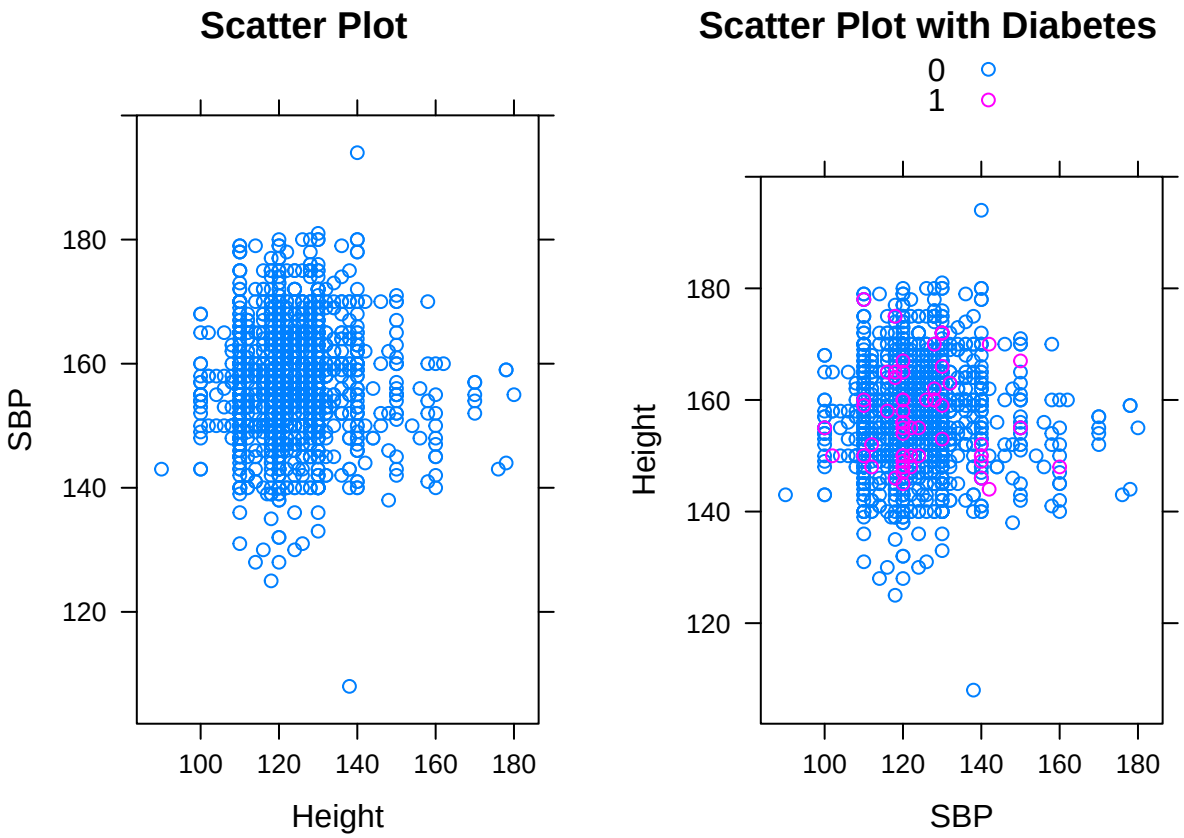


Figure 1: Scattor Plot of Height Vs SBP and Height Vs SBP Vs Diabetes

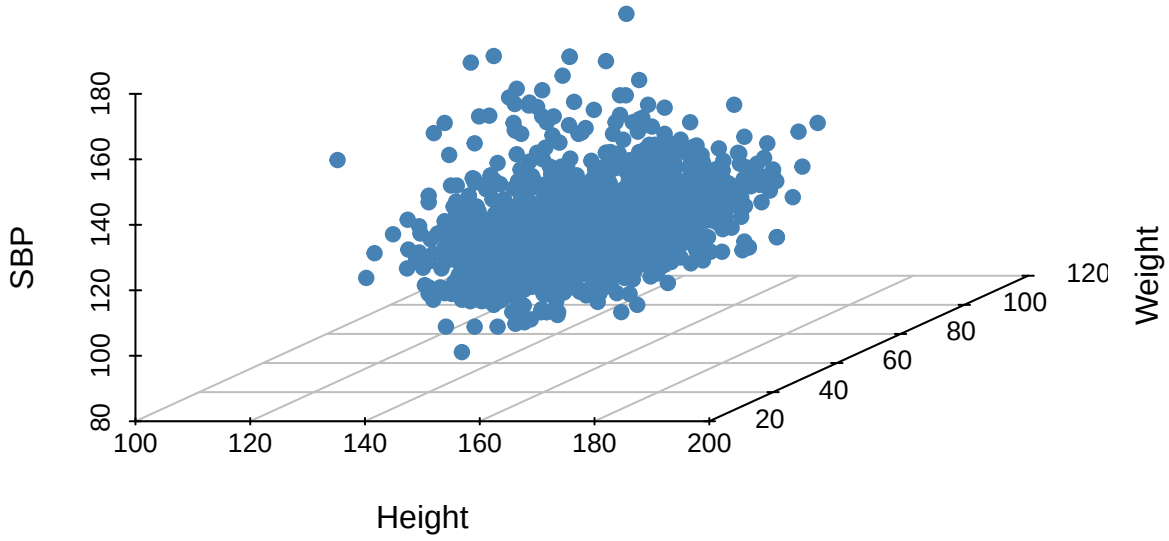


Figure 2: 3d Scatter plot

3D Scatter Plot

- Suppose we need to visualize the relationship between the variables 'Height', 'Weight' and 'SBP', what to do? Is there any way we can plot these three variables together?

Answer is yes. And that can be achieved in many ways. The immediate tourch will be on 3dscatter plot. Look at the following 3d scatter plot.

```
library(scatterplot3d)# to draw 3d scatter plot
scatterplot3d(
  CSdata[,4:6], pch = 19, color = "steelblue",
   grid = TRUE, box = FALSE
  )
```

```
## Warning: Unknown or uninitialised column: 'color'.
```

Scatter Plot Matrix

Fine. All good untill we want to visualize 3 variables together.

- Yeah. I got your question. What if the number of variable is more than 3?

This question is answered by many authors over the decade. One of them is scatter plot matrix. This method uses the 2d scatter plot pair wise. That is, it produces a pairwise comparison of multivariate data. For example, in our 'Cross Sectional Data', suppose if we wants to compare (visualize) all the variables (except patient id).

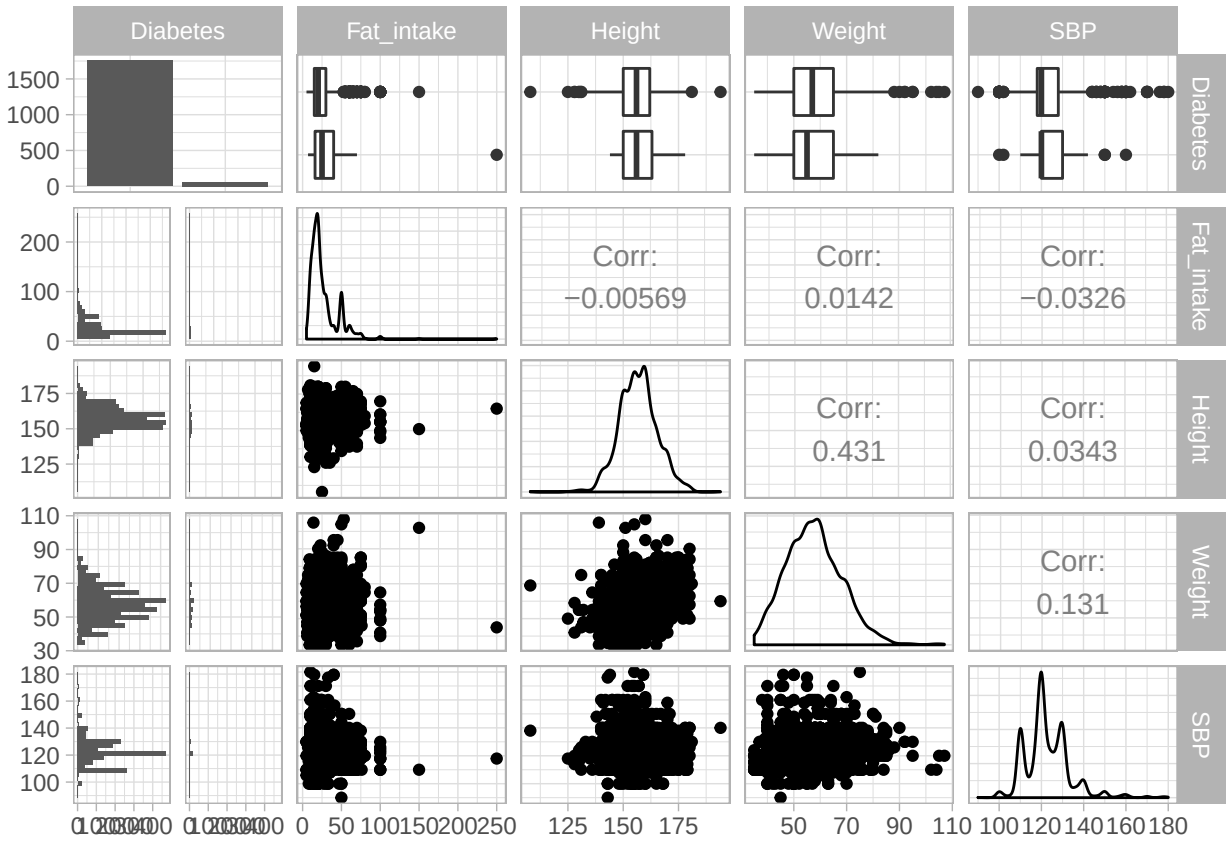


Figure 3: Scatter Plot Matrix

Here, I use the recent version (2016) of plotting scheme using the package `GGally`, an extension of `ggplot2` package.

```
library(GGally)
library(ggplot2)
CSdata$Diabetes = as.factor(CSdata$Diabetes)
ggpairs(CSdata[,c(2:6)])+
  theme_light()  #theme_light() is just to change the design of the graph. Feel free to play around
```

- What if the variables present in a data is more in number?

Obviously, we cant think of scatter plot matrix since the number of variables is proposional to the complexity in drawaing the plot. There are many other methods which over come this difficulty.

1.2.2 Chernoff Faces

(Chernoff 1973) depict each variable as a feature on a face, such as length of nose, size of eyes, shape of eyes, and so on. Flury and Riedwyl (1981) suggested using asymmetric faces, thus increasing the number of representable variables.

- The goal of Chernoff's faces is to show a bunch of variables at once via facial features like lips, eyes, and nose size.
- Most of the time there are better solutions, but the faces can be interesting to work with.

```
library(readxl)
cskdata = read_excel('csk.xlsx')
cskdata
```

PLAYER	Bt_Avg	SR	4s	6s	Ratio_inn_mat
Ambati Rayudu	43.00	149.75	53	34	1.0000000
MS Dhoni	75.83	150.66	24	30	0.9375000
Suresh Raina	37.08	132.44	46	12	1.0000000
Ravindra Jadeja	17.80	120.27	3	4	0.6250000
Deepak Chahar	16.66	172.41	1	4	0.3333333
Harbhajan Singh	9.66	80.55	3	1	0.2307692
Kedar Jadhav	0.00	109.09	1	2	1.0000000
Shardul Thakur	0.00	300.00	3	0	0.0769231

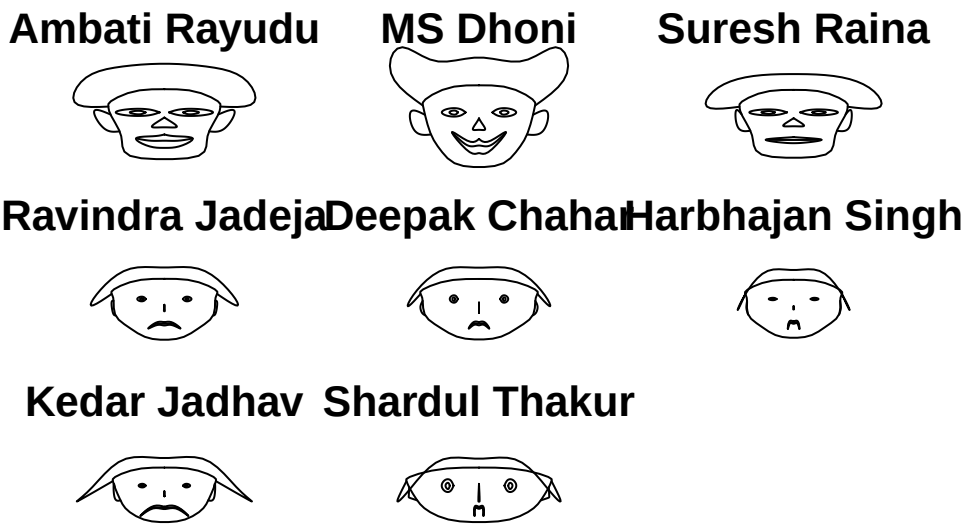
To create Chernoff faces these metrics were mapped to certain facial features as given in the table 5.

```
metric_dt = data.frame('Metrics' = c("Batting average",
                                     "Strike rate",
                                     "Number of fours ",
                                     "Number of sixers",
                                     "Ratio of Innings to total matched played"),
                       'Facial Features' = c("Height of face",
                                             "Curve of smile",
                                             "Width of eyes",
                                             "Height of eyes",
                                             "Width of face")
)
metric_dt
```

Table 5: Metric matchings for Chernoff Faces

Metrics	Facial.Features
Batting average	Height of face
Strike rate	Curve of smile
Number of fours	Width of eyes
Number of sixers	Height of eyes
Ratio of Innings to total matched played	Width of face

```
library(aplpack)
faces(cskdata[,2:6],labels=cskdata$PLAYER,face.type = 0)
```




```
## effect of variables:
##  modified item       Var
##  "height of face   " "Bt_Avg"
##  "width of face    " "SR"
##  "structure of face" "4s"
##  "height of mouth  " "6s"
##  "width of mouth   " "Ratio_inn_mat"
##  "smiling          " "Bt_Avg"
##  "height of eyes   " "SR"
##  "width of eyes    " "4s"
##  "height of hair   " "6s"
##  "width of hair   "  "Ratio_inn_mat"
##  "style of hair   "  "Bt_Avg"
##  "height of nose  "  "SR"
##  "width of nose   "  "4s"
##  "width of ear    "  "6s"
##  "height of ear   "  "Ratio_inn_mat"
```

As can be seen, the happiest face seems to be of Dhoni, no surprise there since he has the highest strike rate, a variable mapped to curve of the smile.

1.2.3 Glyphs

(Anderson 1960) Glyphs are circles of fixed size with rays whose lengths represent the values of the variables. Anderson suggested using only three lengths of rays, thus rounding the variable values to three levels.

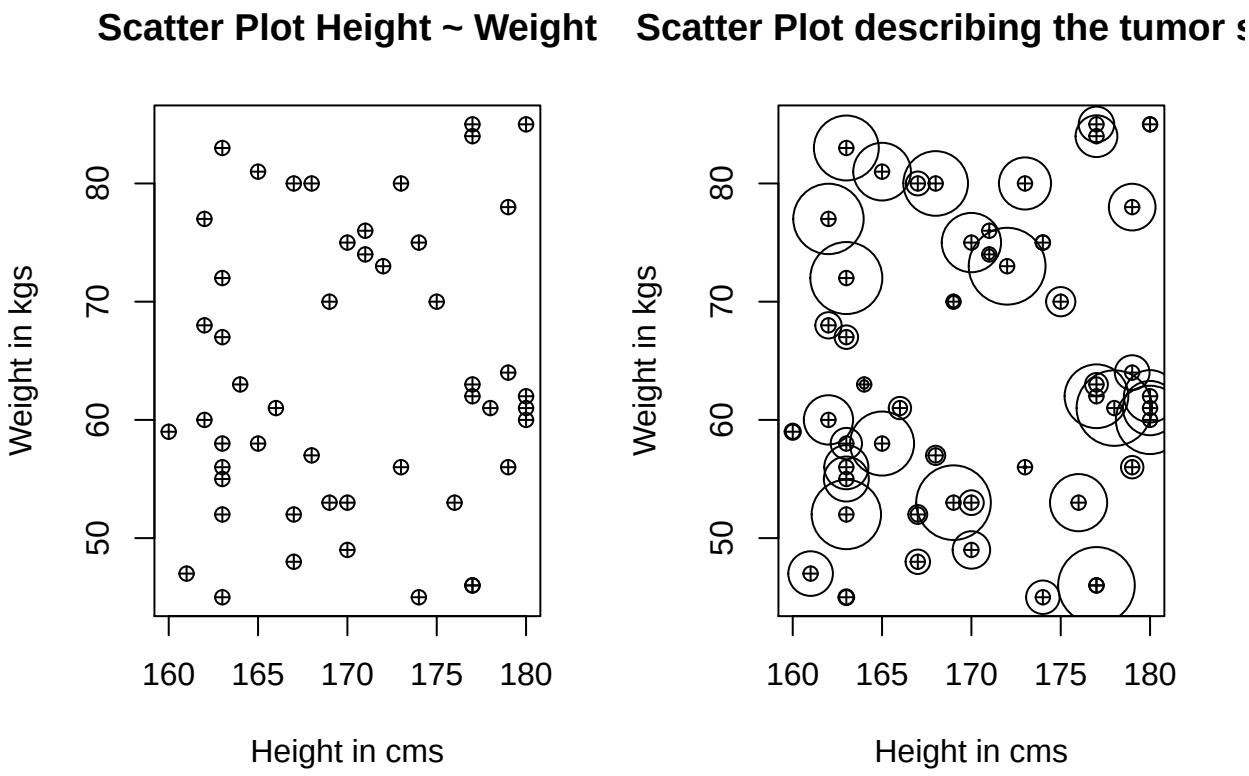
Consider the simulated data set where height, weight and the tumor size is been measured. The basic scatterplot can only display two variables.

```
set.seed(2)
height = sample(160:180, 50, replace = TRUE)
weight = sample(45:85, 50, replace = TRUE)
tumour_size = runif(50, 0,1)
df = data.frame(height, weight, tumour_size)
head(df)
```

height	weight	tumour_size
163	45	0.2009347
174	45	0.4276391
172	73	0.9806000
163	83	0.8289221
179	56	0.2869739
179	78	0.5959169

```
par(mfrow=c(1,2))
plot(weight~height,data=df,
     xlab="Height in cms",
     ylab="Weight in kgs",pch=10, main="Scatter Plot Height ~ Weight")

plot(weight~height,data=df,
     xlab="Height in cms",
     ylab="Weight in kgs",pch=10, main="Scatter Plot describing the tumor size")
with(df,symbols(height,weight, circles=tumour_size, inches=0.2,add=TRUE))
```



1.2.4 Profiles

Profiles represent each point by p vertical bars, with the heights of the bars depicting the values of the variables. Sometimes the profile is outlined by a polygonal line rather than bars.

1.2.5 Stars

Stars portray the value of each (normalized) variable as a point along a ray from the center to the outside of a circle. The points on the rays are usually joined to form a polygon.

```
library(randomNames)
set.seed(3)
Name = randomNames(50, which.names = 'first')
height = sample(160:180, 50, replace = TRUE)
weight = sample(45:85, 50, replace = TRUE)
tumour_size = runif(50, 0,1)
df = data.frame(Name, height, weight, tumour_size, rnorm(50, 10,3))

stars(df, cex = 0.55, labels = Name, col.stars = factor(Name))
```

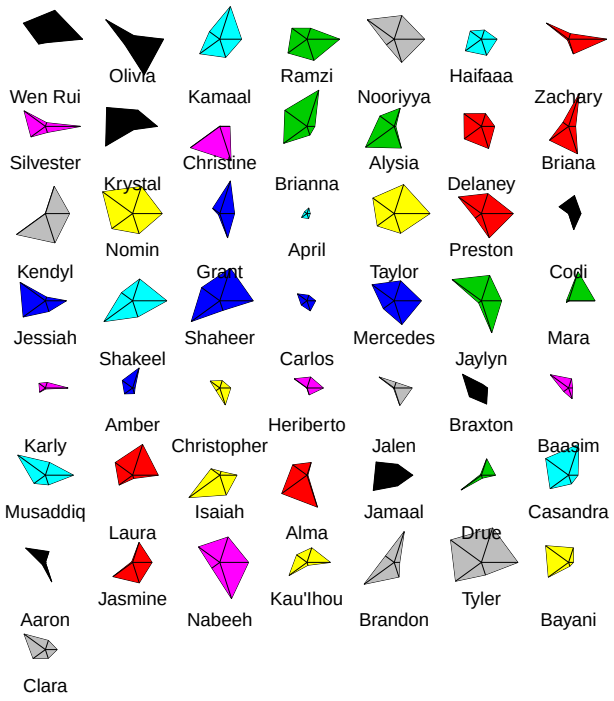


Figure 4: Visualization of Multivariate Data using stars

2 Excercises

In a baking competition, there are three judges who would sco re a cake created by each participant from zero (inedible) to ten (worth to be present ed in front of HM the Queen). There are ten contestants in the competition, and the matrix of scores are given in the following table.

```
judges_data = read.table(
  'https://raw.githubusercontent.com/davidrajdgl/MultivariteAnalysis/master/DataSets/3judges.txt',
  header = TRUE)
judges_data
```

Table 7: Scores of Three Judges

J1	J2	J3
7	7	7
6	7	7
5	6	4
6	7	6
7	8	7
7	8	7
5	7	4
6	8	5
5	5	5
7	7	6

Answer the following questions:

1. Which judge gives higher scores than the others?
2. What is the total and average score that each contestant received?
3. The host of the competition realises that the first judge had a cold and the score he gave is not entirely reliable. So the host suggests that the scores given by Judge 1 are weighted by 0.5, by Judge 2 are weighted by 1.25, and by Judge 3 are weighted by 1.25 as well. In this new scheme of scoring, what are the total scores that each individual receives?
4. udge 1 protested the new scheme and suggested a different scoring scheme. He sug- gested that the weights should be 0.5, 1.0, and 1.0 for Judge 1 to Judge 3 respectively. In this scheme, what is the average score per participant
5. Add some random names for the constestants using `randomNames` package.