International Journal of Advance Research in Science and Engineering
Volume No.07, Special Issue No.04, March 2018
www.ijarse.com
IJARSE
ISSN: 2319-8354

R Software Based Graphical Visualisation of Multivariate Dataset

Immad Ahmad Shah[1], Owais Bhat[2], Showkat Yousuf[3]

[1]*Division of Agricultural Statistics, SKUAST-Kashmir, (India)*

[2]*Division of Agricultural Engineering, SKUAST-Kashmir, (India)*

[3]*Division of Soil Science, SKUAST-Kashmir, (India)*

ABSTRACT

Multivariate data are encountered in all aspects by researchers, scientists, engineers, manufacturers, financial managers and various kinds of analysts. Multivariate data visualization is hence strongly motivated by the many situations when they are trying to obtain an integrated understanding of the data distributions and investigate the inter-relationships between different data attributes. Such an effective visual display tool is demanded to facilitate users to identify, locate, distinguish, categorize, cluster, rank, compare, associate or correlate the underlying data. Graphics can effectively complement statistical data analysis in various ways. R software is used to visualise the data to enable users to explore the data space intuitively and interactively, as well as discriminating individual dimensions.

Keywords: Multivariate data, R software, Plots, Graphics, Packages

I. INTRODUCTION

Multivariate data visualization, as a specific type of information visualization, is an active research field with numerous applications in diverse areas ranging from science communities and engineering design to industry and financial markets, in which the correlations between many attributes are of vital interest. Due to the high dimensionality of multivariate data, we inevitably sacrifice the ability to show the details of each attributes [1] as we have fewer graphic attributes for encoding. While information is growing in an exponential way, our world is flooded with data which, we believe, should contain some kind of valuable information that can possibly expand the human knowledge. However, extracting the meaningful information is a difficult task when large quantities of data are presented in plain text or traditional tabular form. Effective graphical representations of the data thus enjoy popularity by harnessing the human's visual perception capabilities. Multivariate data is often of huge size and high dimensionality that will most likely result a dense structure. It is hence difficult to present such data in a single visual display, making it challenging to enable users to explore the data space intuitively and interactively, as well as discriminating individual dimensions. Dual view and distortion skills like fisheyes may be helpful to solve this problem. Furthermore, the ordering of dimensions has a major impact on the expressiveness of visualization. The terms multidimensional and multivariate are often used vaguely. Strictly

speaking, multidimensional refers to the dimensionality of the independent dimensions while multivariate refers to that of the dependent variables [2]. The more appropriate term for multivariate data visualization should be multidimensional multivariate data visualization [3]. Nevertheless, a set of multivariate data is in high dimensionality and can possibly be regarded as multidimensional because the key relationships between the attributes are generally unknown in advance.

II. MATERIALS AND METHODS

In the present study the data was obtained from DARS (Dryland Agriculture Research Station), SKUAST-Kashmir, comprising of 55 genotypes of maize. Five characters (Plant Height, Ear Height, Cob Length, Cob Diameter, Yield per Plant) were evaluated for each genotype. R was downloaded from **www.r-project.org** and was used to visualise the data and generate various plots.

III. RESULTS

The first thing, to analyse the multivariate data is to read it into R, and to plot the data. The data can read into R using the ***read.table()*** function.

```
> data=read.table("clipboard",header=T)
```

To read in the top portion of the data set the function head() is used which displays just the upper few observations from the entire data set.

```
> head(data)
```

The result of this function is shown below:

	Genotypes	PlantHt	Earhgt	CobLng	Cobdia	YPlant
1	G1	105	54	10.7	3.2	30
2	G2	120	60	9.2	3.0	44
3	G3	118	51	9.0	3.0	31
4	G4	100	42	8.0	2.9	29
5	G5	140	56	7.8	2.8	29
6	G6	138	50	8.7	3.3	42

This gives a clear understanding of how our dataset looks like. First column comprises of the various genotypes on which various parameters like Plant Height, Ear Height, Cob Length, Cob Diameter, Yield per Plant have been quantified. Once the multivariate data set is read into R, the next step is usually to make a plot of the data. One common way of plotting multivariate data is to make a “matrix scatterplot”, showing each pair of variables

plotted against each other. We can use the "scatterplotMatrix()" function from the "car" R package to do this. To use this function, the "car" R package is installed into R using the following command:

> install.packages("car")

Once the "car" R package is installed, load the "car" R package by typing:

> library("car")

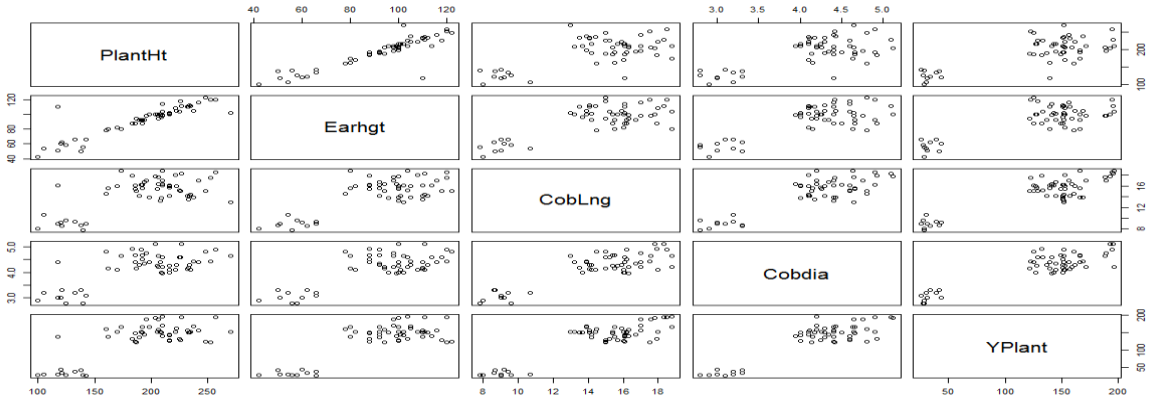
3.1) Scatter plots:

To make a matrix scatterplot of these 5 variables using the ***scatterplotMatrix()*** function we type:

> scatterplotMatrix(data[2:6])

The resultant plot is shown in Fig.1. We can easily observe patterns in the relationships between pairs of attributes from the matrix, but there may be important patterns in higher dimensions which are barely recognized in it [4].

Figure 1: Scatter Matrix plot.



To do anything beyond very simple graphs, it's generally better to switch to ***"ggplot2"*** package because ***"ggplot2"*** provides a unified interface and set of options, instead of the grab bag of modifiers and special cases required in base graphics. To load the ***"ggplot"*** package in R the following command is used:

> library("ggplot2", lib.loc="~/R/win-library/3.2")

To enlist the names of the variabvles in our data set the following command is used:

> names(data)

The output of this command is :

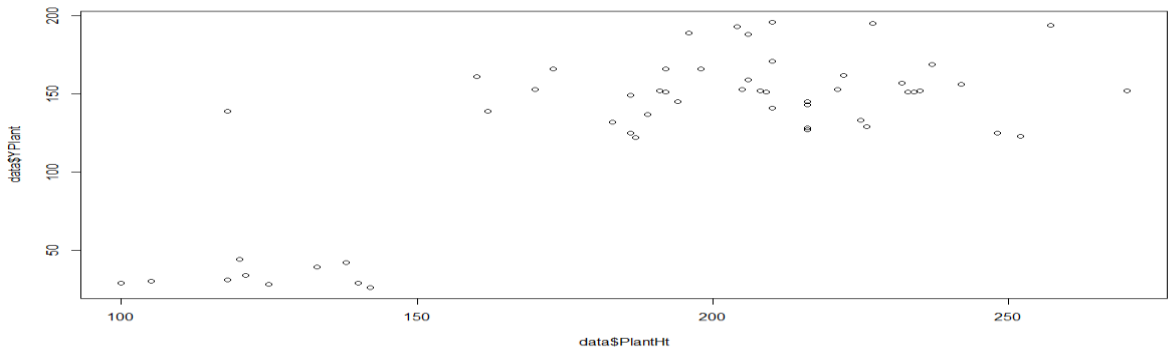
[1] "PlantHt" "Earhgt" "CobLng" "Cobdia" "YPlant"

To plot the Plant Height variable with the Yield per Plant variable the following command is used:

> plot(data$PlantHt,data$YPlant)

The below plot (Fig. 2) gives a clear understanding of how the two variables exist in association with each other.

Figure 2: Plot between Plant Height and Yield per Plant.

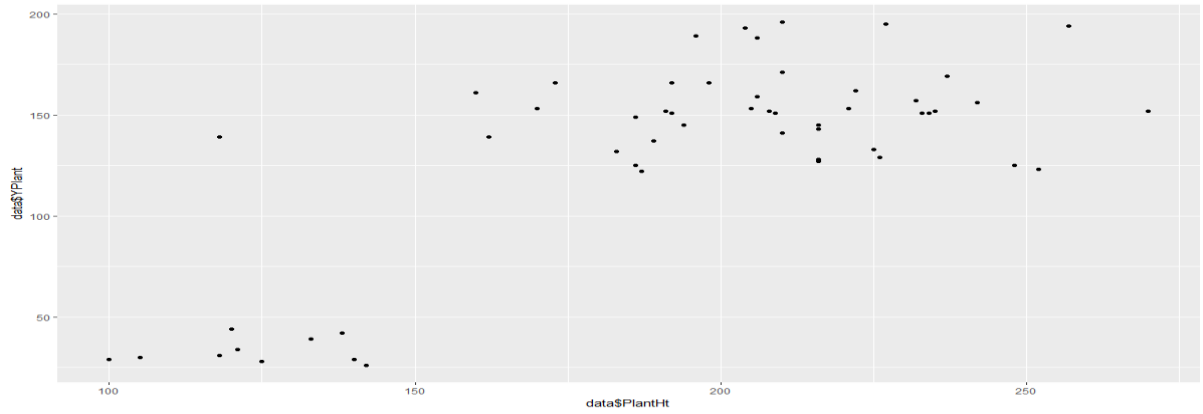


With the ***"ggplot2"*** package, we can get a similar result using ***qplot().***

> library("ggplot2", lib.loc="~/R/win-library/3.2")

> qplot(data$PlantHt,data$YPlant)

Figure 3: Plot using qplot function.



If the two vectors are already in the same data frame,we can also use the following syntax:

> qplot(PlantHt,YPlant,data=data)

This is equivalent to:

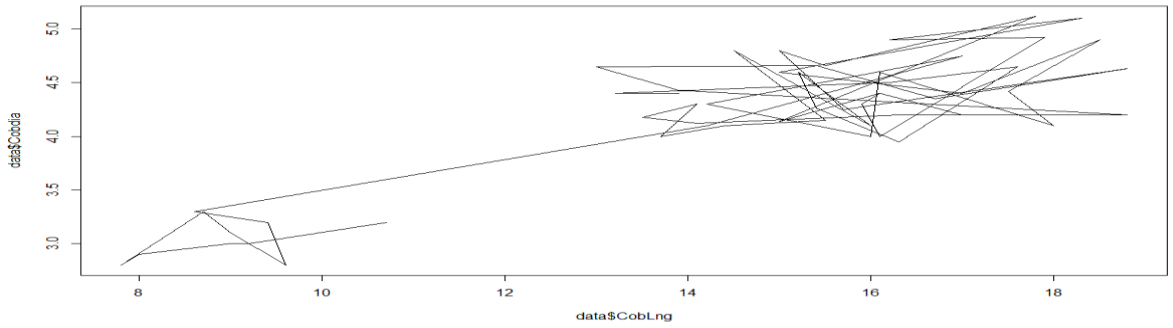
> ggplot(data, aes(x=PlantHt, y=YPlant)) + geom_point()

3.2) Line Plots:

Line graphs are typically used for visualizing how one continuous variable, on the yaxis, changes in relation to another continuous variable, on the x-axis. Often the x variable represents time, but it may also represent some other continuous quantity. To create a line graph pass it a vector of x values and a vector of y values, and use type="l" as shown in the command below and the resulting plot is shown in the Figure 4 for the variables Cob length and Cob Diameter.

> plot(data$CobLng,data$Cobdia, type="l")

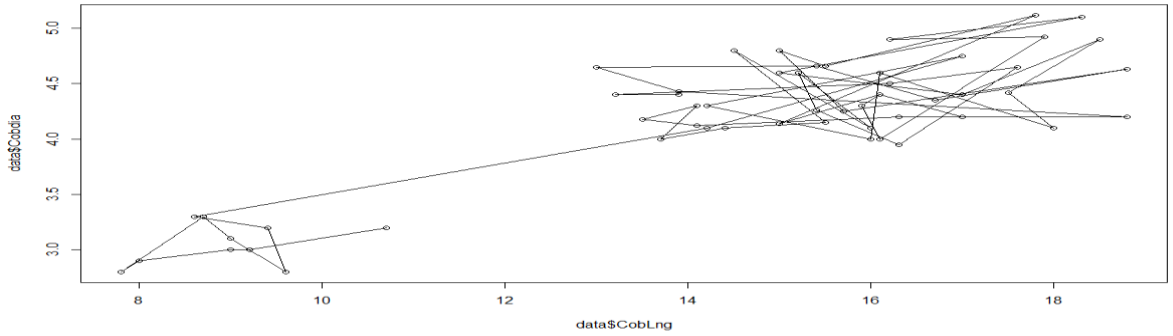
Figure 4: Line Plot of the variables Cob Length and Cob Diameter.



To add points and/or multiple lines, first call ***plot()*** for the first line, then add points with ***points()*** and additional lines with ***lines()***. The resulting plot is shown in Fig. 5.

> points(data$CobLng,data$Cobdia)

Figure 5: Line plot with points.

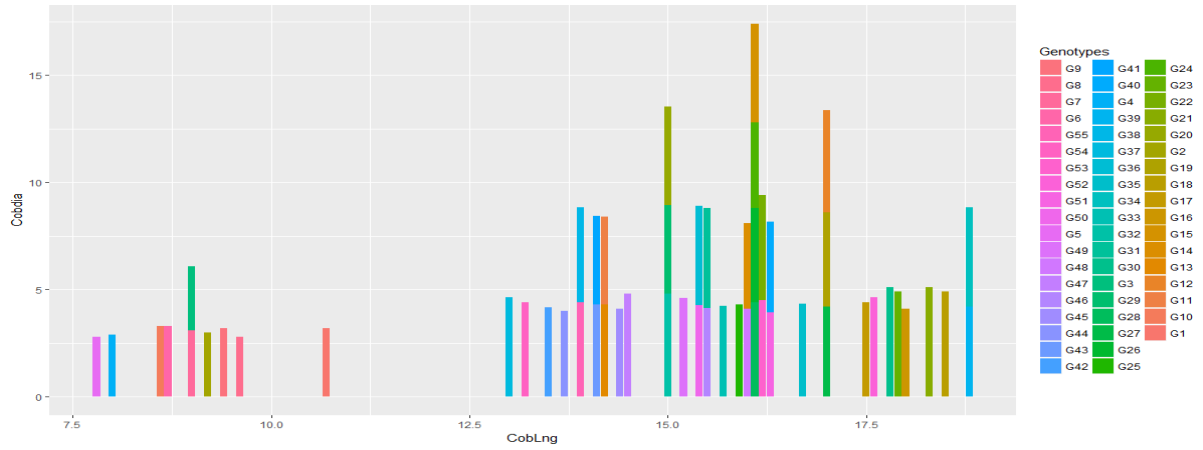


3.3) Stacked Bar Graph:

To understand how the graph is made, it's useful to see how the data is structured. There are 55 Genotypes of maize and for each genotype there is a value for Cob length and Cob diameter . To obtain a stacked bar graph the following command is used:

> ***ggplot(data, aes(x=CobLng, y=Cobdia, fill=Genotypes)) + geom_bar(stat="identity") + guides(fill=guide_legend(reverse=TRUE))***

Figure 6: Stacked Bar Plot.

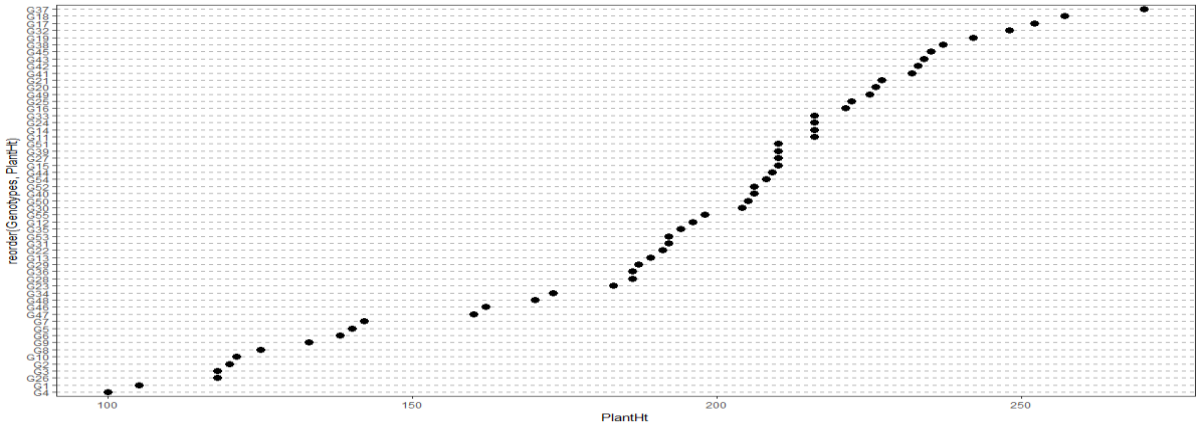


3.4) Basic dot plot:

Dot plots are sometimes used instead of bar graphs because they reduce visual clutter and are easier to read. The following command in R is used to obtain a dot plot. The dot plot is shown in Figure 7. Along its x axis is the variable Plant Height and along the y axis are the various genotypes.

> ggplot(data, aes(x=PlantHt, y=reorder(Genotypes, PlantHt))) + geom_point(size=3) +theme_bw() + theme(panel.grid.major.x = element_blank(),panel.grid.minor.x = element_blank(),panel.grid.major.y = element_line(colour="grey60", linetype="dashed"))

Figure 7: Dot Plot



IV.CONCLUSION

The primary goal of data visualization is to communicate information clearly and efficiently via statistical graphics and plots. Numerical data may be encoded using dots, lines, or bars, to visually communicate a quantitative message. Effective visualization helps users analyze and reason about data and evidence. It makes complex data more accessible, understandable and usable.We briefly outlined some of the general considerations for multivariate visualization and indicated some of the most popular current methods. There are many other methods that may also be helpful. The most important tools for revealing the structure of bivariate data are the scatterplot.

REFERENCES

[1.] C. Ware, *Information Visualization: Perception for Design* (Morgan Kaufmann Publishers,2004)

[2.] R. D. Bergeron, W. Cody, W. Hibbard, D. T. Kao, K. D. Miceli, L. A. Treinish and S. Walther. "*Database Issues for Data Visualization: Developing a Data Model", Proceedings of the IEEE Visualization '93 Workshop on Database Issues for Data Visualization, Lecture Notes in Computer Science, vol.871*, Springer-Verlag, 1994.3-15

[3.] P. E. Hoffman and G. G. Grinstein.*"A Survey of Visualizations for High-Dimensional Data Mining"*(Information Visualization in Data Mining and Knowledge Discovery, Morgan Kaufmann Publishers,2001,47-82.)

[4.] "Visualizing Higher Dimensional Data" from the MathWorks, available at: http://www.mathworks.com/products/demos/statistics/mvplotdemo.html, 2006.

[5.] R Core Team. R: A language and enivironment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/,2013.