Data Analytics and Visualization through R Programming & Latex

Anji Reddy.Vaka

Associate Professor and Training & Placement Officer,

SPOC-NPTEL Courses by IIT-MADRAS,

Department Of Computer Science & Engineering

Lendi Institute of Engineering & Technology

Jonnada(V),Denkada(M),

Vizianagaram- 535005,Andhra Pradesh.

E-Mail id: anjireddy@lendi.org

Mobile Number : 8074680798

Domain Knowledge

- Virtual Machines
- Big Data
- Cloud Computing
- Data Base Management System

Evolution of R

- R was initially written by **Ross Ihaka** and **Robert Gentleman** at the Department of Statistics of the University of Auckland in Auckland, New Zealand. R made its first appearance in 1993.
- A large group of individuals has contributed to R by sending code and bug reports.
- Since mid-1997 there has been a core group (the "R Core Team") who can modify the R source code archive.

What is R

- Software for Statistical Data Analysis
- Based on S
- Programming Environment
- Interpreted Language
- Data Storage, Analysis, Graphing
- Free and Open Source Software

What R does and does not

- data handling and storage: numeric, textual
- matrix algebra
- hash tables and regular expressions
- high-level data analytic and statistical functions
- classes (“OO”)
- graphics
- programming language: loops, branching, subroutines

- is not a database, but connects to DBMSs
- has no graphical user interfaces, but connects to Java, TclTk
- language interpreter can be very slow, but allows to call own C/C++ code
- no spreadsheet view of data, but connects to Excel/MsOffice
- no professional / commercial support

Obtaining R

- **Current Version: R-2.0.0**
- **Comprehensive R Archive Network:**

  http://cran.r-project.org
- **Binary source codes**
- **Windows executables**
- **Compiled RPMs for Linux**
- **Can be obtained on a CD**

Installing R

- **Binary (Windows/Linux): One step process**
  - **exe, rpm (Red Hat/Mandrake), apt-get (Debian)**
- **Linux, from sources:**

```
$ tar –zxvf “filename.tar.gz”

$ cd filename

$ ./configure

$ make

$ make check

$ make install
```

Starting R

Windows, Double-click on Desktop Icon

Linux, type R at command prompt

Strengths and Weaknesses

- **Strengths**
  - **Free and Open Source**
  - **Strong User Community**
  - **Highly extensible, flexible**
  - **Implementation of high end statistical methods**
  - **Flexible graphics and intelligent defaults**
- **Weakness**
  - **Steep learning curve**
  - **Slow for large datasets**

Basics

- **Highly Functional**
  - **Everything done through functions**
  - **Strict named arguments**
  - **Abbreviations in arguments OK (e.g. T for TRUE)**
- **Object Oriented**
  - **Everything is an object**
  - **“< –” is an assignment operator**
  - **“X <- 5”: X GETS the value 5**

Getting Help in R

- **From Documentation:**
  - **?WhatIWantToKnow**
  - **help("WhatIWantToKnow")**
  - **help.search("WhatIWantToKnow")**
  - **help.start()**
  - **getAnywhere("WhatIWantToKnow")**
  - **example("WhatIWantToKnow")**
- **Documents: "Introduction to R"**
- **Active Mailing List**
  - **Archives**
  - **Directly Asking Questions on the List**

Data Structures

- Supports virtually any type of data
- Numbers, characters, logicals (TRUE/ FALSE)
- Arrays of virtually unlimited sizes
- Simplest: Vectors and Matrices
- Lists: Can Contain mixed type variables
- Data Frame: Rectangular Data Set

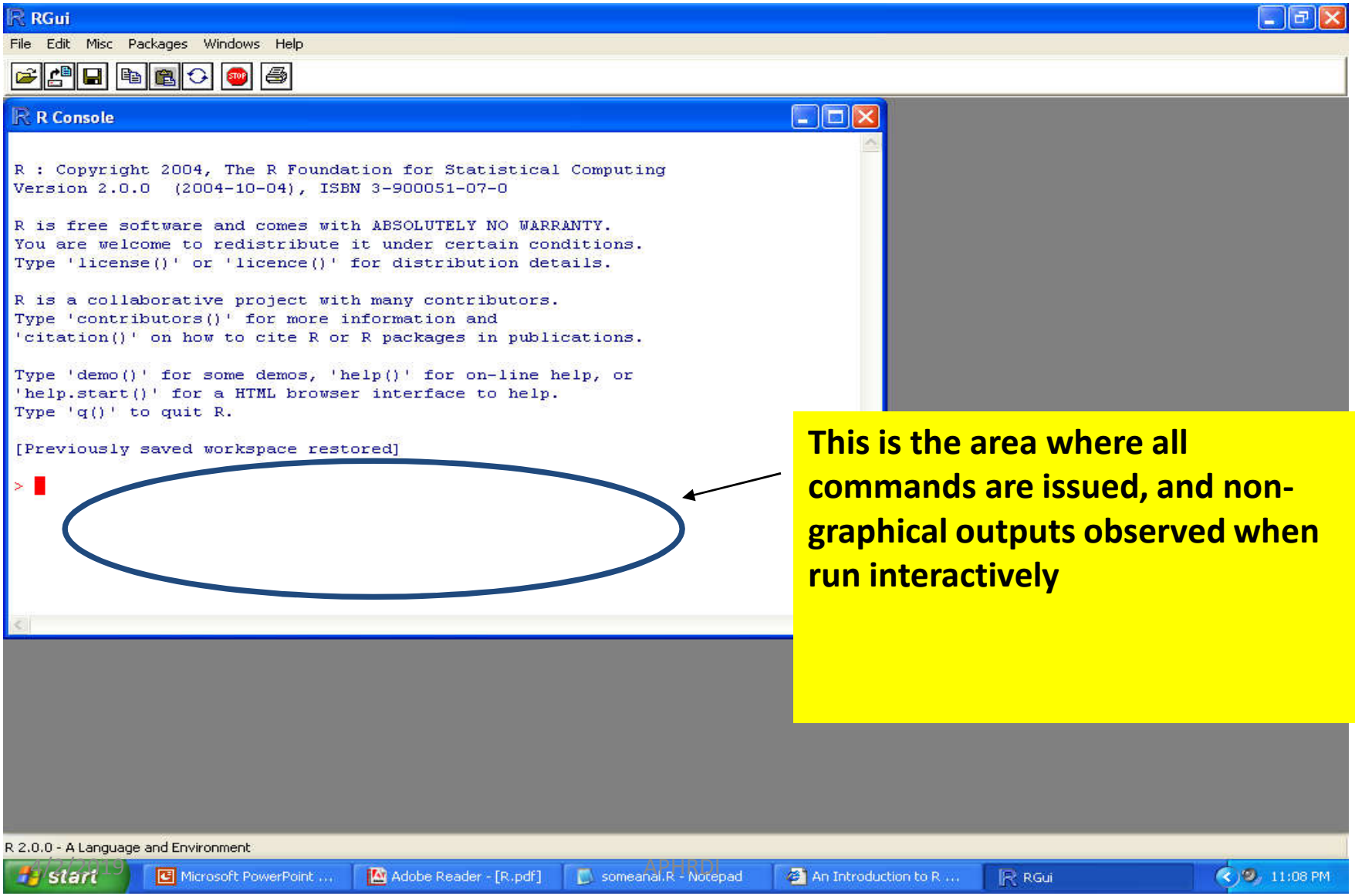
Data Structure in R

	Linear	Rectangular
All Same Type	VECTORS	MATRIX*
Mixed	LIST	DATA FRAME

Running R

- Directly in the Windowing System (Console)
- Using Editors
  - Notepad, WinEdt, Tinn-R: Windows
  - Xemacs, ESS (Emacs speaks Statistics)
- On the Editor:
  - source("filename.R")
  - Outputs can be diverted by using
    - sink("filename.Rout")

R Working Area



In an R Session…

- First, read data from other sources
- Use packages, libraries, and functions
- Write functions wherever necessary
- Conduct Statistical Data Analysis
- Save outputs to files, write tables
- Save R workspace if necessary (exit prompt)

Specific Tasks

- **To see which directories and data are loaded, type: search()**
- **To see which objects are stored, type: ls()**
- **To include a dataset in the searchpath for analysis, type: attach(NameOfTheDataset, expression)**
- **To detach a dataset from the searchpath after analysis, type: detach(NameOfTheDataset)**

Data Types

- There are different data types in R language

  Numeric

  Character

  Logical

  Integer

  Complex

  Raw

Vector

- When you want to create vector with more than one element, you should use **c()** function which means to combine the elements into a vector.

```
# Create a vector.
   apple <- c('red','green',"yellow")
   print(apple)
# Get the class of the vector.
   print(class(apple))
```

Lists

- A list is an R-object which can contain many different types of elements inside it like vectors, functions and even another list inside it.

```
# Create a list.
   list1 <- list(c(2,5,3),21.3,sin)
# Print the list.
   print(list1)
```

Data Frames

- Data frames are tabular data objects. Unlike a matrix in data frame each column can contain different modes of data. The first column can be numeric while the second column can be character and third column can be logical. It is a list of vectors of equal length.
- BMI <- data.frame(gender = c("Male", "Male","Female"), height = c(152, 171.5, 165), weight = c(81,93, 78), Age = c(42,38,26)) print(BMI)

Reading data into R

- R not well suited for data preprocessing
- Preprocess data elsewhere (SPSS, etc…)
- Easiest form of data to input: text file
- Spreadsheet like data:
  - Small/medium size: use `read.table()`
  - Large data: use `scan()`
- Read from other systems:
  - Use the library "foreign": `library(foreign)`
  - Can import from SAS, SPSS, Epi Info
  - Can export to STATA

Reading Data: summary

- Directly using a vector e.g.: `x <- c(1,2,3…)`
- Using `scan` and `read.table` function
- Using `matrix` function to read data matrices
- Using `data.frame` to read mixed data
- library(foreign) for data from other programs

Accessing Variables

- edit(<mydataobject>)
- Subscripts essential tools
  - x[1] identifies first element in vector x
  - y[1,] identifies first row in matrix y
  - y[,1] identifies first column in matrix y
- $ sign for lists and data frames
  - myframe$age gets age variable of myframe
  - attach(dataframe) -> extract by variable name

Subset Data

- **Using subset function**
  - **subset() will subset the dataframe**
- **Subscripting from data frames**
  - **myframe[,1] gives first column of myframe**
- **Specifying a vector**
  - **myframe[1:5] gives first 5 rows of data**
- **Using logical expressions**
  - **myframe[myframe[,1], < 5,] gets all rows of the first column that contain values less than 5**

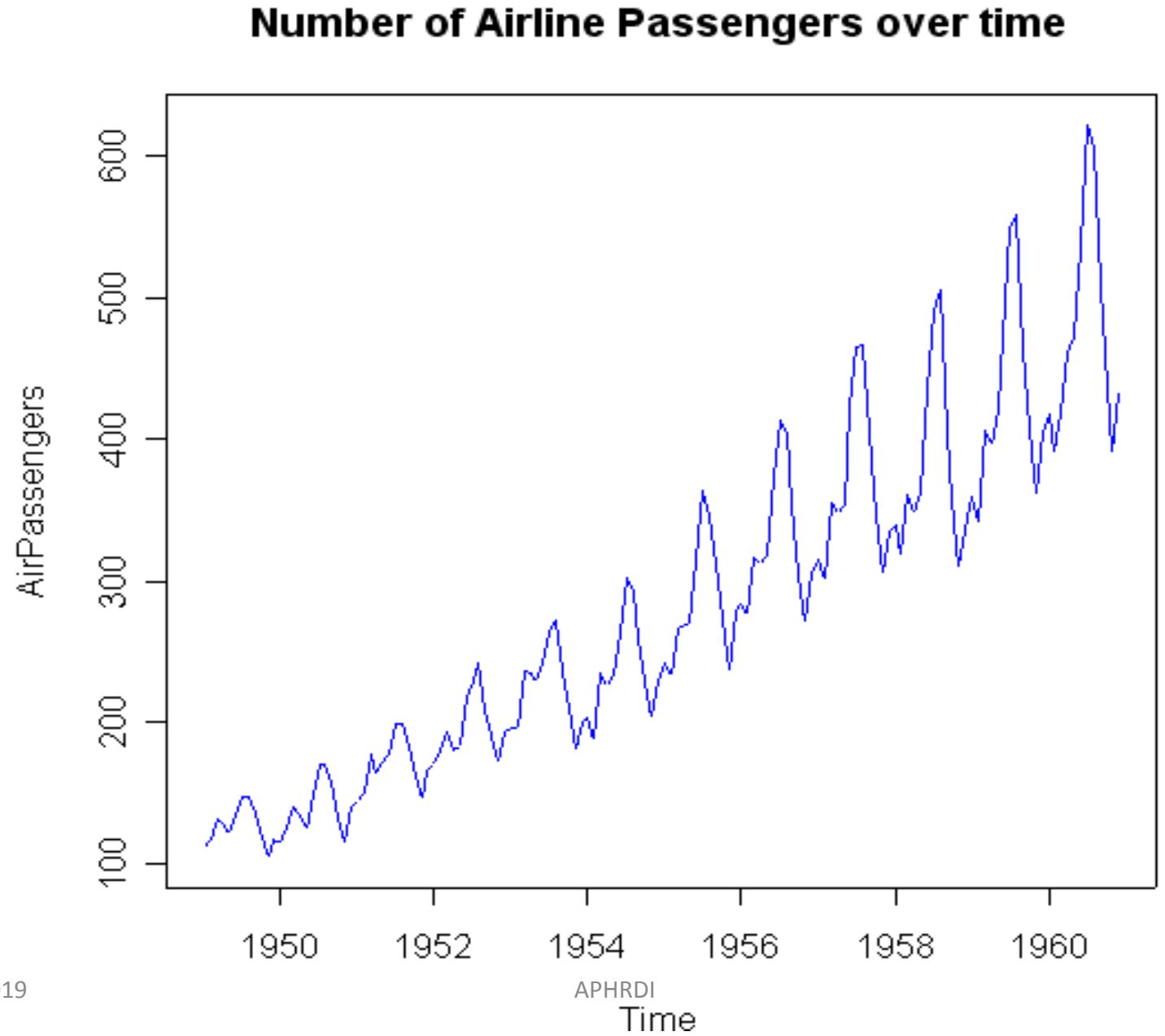
R statistics Examples

- R - Mean, Median & Mode
- R - Linear Regression
- R - Multiple Regression
- R - Logistic Regression
- R - Normal Distribution
- R - Binomial Distribution
- R - Poisson Regression
- R - Analysis of Covariance
- R - Time Series Analysis

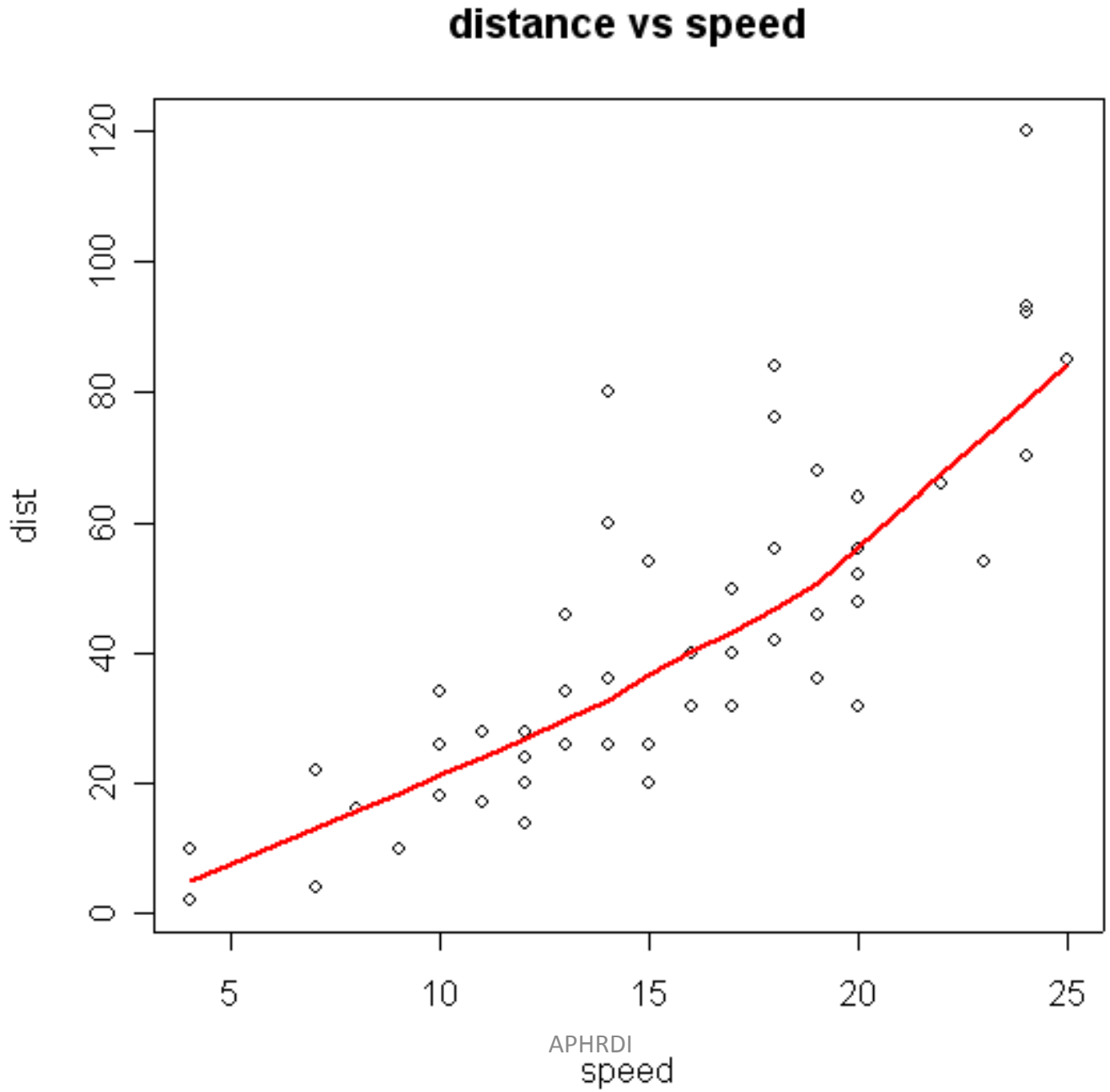
Graphics

- Plot an object, like: plot(num.vec)
  - here plots against index numbers
- Plot sends to graphic devices
  - can specify which graphic device you want
    - postscript, gif, jpeg, etc…
    - you can turn them on and off, like: dev.off()
- Two types of plotting
  - high level: graphs drawn with one call
  - Low Level: add additional information to existing graph

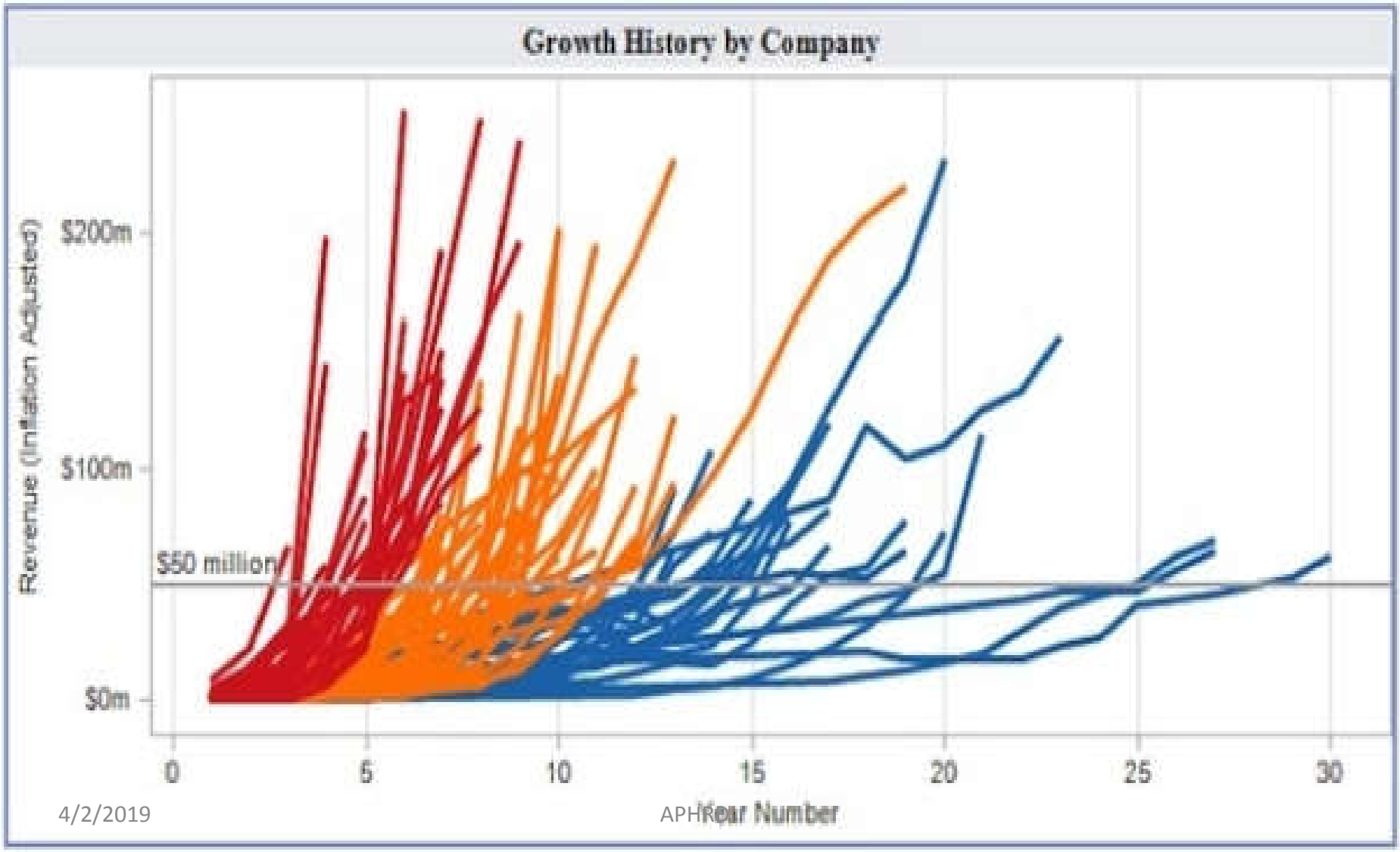
High Level: generated with plot()



Low Level: Scattergram with Lowess



Data Visualization



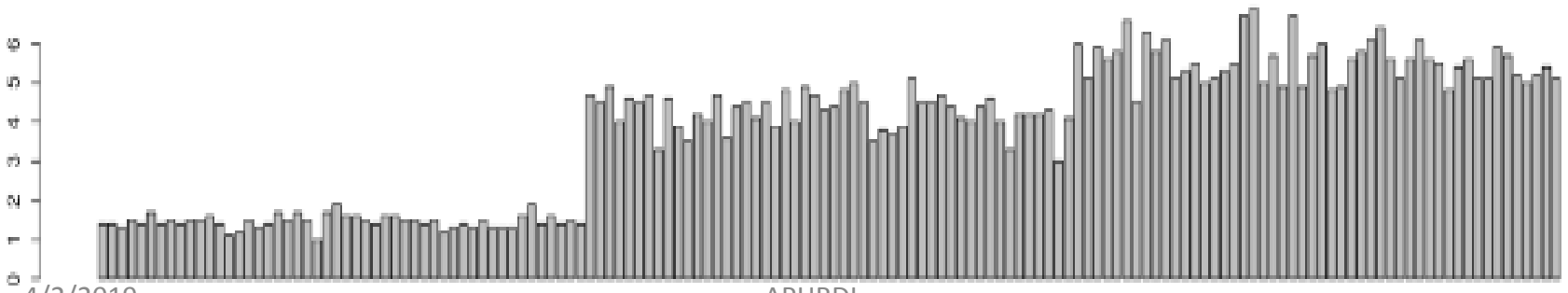
Bar chart

```
barplot(iris$Petal.Length) #Creating simple Bar Graph

barplot(iris$Sepal.Length,col  = brewer.pal(3,"Set1"))

barplot(table(iris$Species,iris$Sepal.Length),col  = brewer.pal(3,"Set1")) #Stacked

Plot
```



Programming in R

- **Functions & Operators typically work on entire vectors**
- **Expressions surrounded by {}**
- **Codes separated by newlines, “;” not necessary**
- **You can write your own functions and use them**

Statistical Functions in R

- Descriptive Statistics
- Statistical Modeling
  - Regressions: Linear and Logistic
  - Probit, Tobit Models
  - Time Series
- Multivariate Functions
- Inbuilt Packages, contributed packages

Descriptive Statistics

- Has functions for all common statistics
- summary() gives lowest, mean, median, first, third quartiles, highest for numeric variables
- stem() gives stem-leaf plots
- table() gives tabulation of categorical variables

Statistical Modeling

- Over 400 functions
  - lm, glm, aov, ts
- Numerous libraries & packages
  - survival, coxph, tree (recursive trees), nls, …
- Distinction between factors and regressors
  - factors: categorical, regressors: continuous
  - you must specify factors unless they are obvious to R
  - dummy variables for factors created automatically
- Use of data.frame makes life easy

How to model

- **Specify your model like this:**
  - **$y \sim x_i + c_i$, where**
  - **y = outcome variable, x_i = main explanatory variables, c_i = covariates, + = add terms**
  - **Operators have special meanings**
    - **+ = add terms, : = interactions, / = nesting, so on…**
- **Modeling -- object oriented**
  - **each modeling procedure produces objects**
  - **classes and functions for each object**

Synopsis of Operators

Operator	Usually means	In Formula means
+ or -	add or subtract	add or remove terms
*	multiplication	main effect and interactions
/	division	main effect and nesting
:	sequence	interaction only
^	exponentiation	limiting interaction depths
%in%	no specific	nesting only

Summarizing…

- Effective data handling and storage
- large, coherent set of tools for data analysis
- Good graphical facilities and display
  - on screen
  - on paper
- well-developed, simple, effective programming

R and statistics

- Packaging: a crucial infrastructure to efficiently produce, load and keep consistent software libraries from (many) different sources / authors
- Statistics: most packages deal with statistics and data analysis
- State of the art: many statistical researchers provide their methods as R packages

Overview

- Why LaTeX
- What Is LaTeX
- How LaTeX
- When/Where LaTeX

Why LATEX?

- LaTeX makes it very simple to handle equations, figures, bibliographies, indexes, etc. With LateX you focus on the content of the document and let the program handle how the output is formatted.

- Microsoft Word is `What You See Is What You Get' (WYSIWYG), this means that you see how the final document will look as you are typing.

What is LaTeX

- LaTeX (pronounced lay-tek) is a document preparation system for producing professional-looking documents, it is not a word processor.

- In 1978, Donald Knuth - arguably one of the most famous and well respected computer scientists - embarked on a project to create a typesetting system, called Tex (pronounced 'tech'), after being disappointed with the quality of his acclaimed *The Art of Programming* series.

What is LaTeX (Cont…)

- LaTeX is essentially a markup language. Content is written in plain text and can be annotated with various 'commands' that describe how certain elements should be displayed.

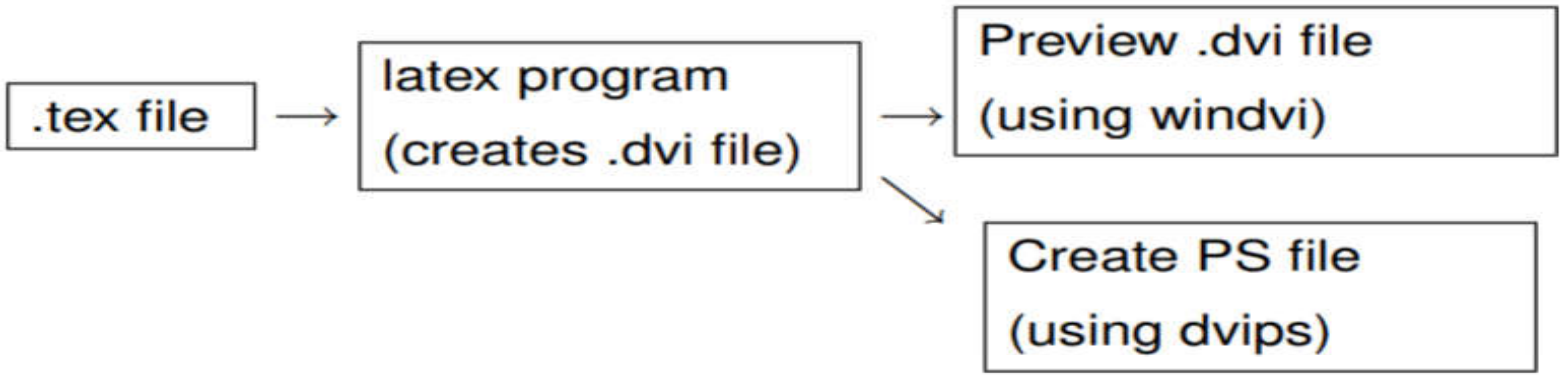
How LaTeX

- To use LAT EX, you first create a file using a plain text editor (such as WinShell or WinEdt on Windows) and give it a name ending with .tex.

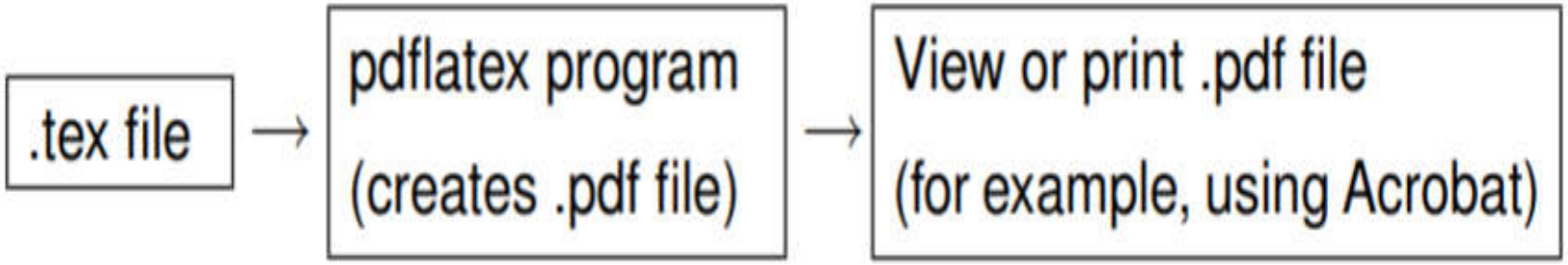
- In this file, you type both the text of your document and the commands to format it. Then there are two ways to process and print your .tex file:

Process your LaTeX Document

1.



2



Special indications

Special character in Latex

\	Escape character: masks special character and introduces commands
{ }	if arguments enclose, text blocks form etc.
%	Comment symbol: The remainder of the line is ignored
$	enclose mathematical formula in pairs inside of text
^ _	exponent and index in mathe
&	depending upon context - tabulator
~	Protected blank.

LaTeX Language Structure

```
\documentclass[a4paper,12pt]{article}
\begin{document}
A sentence of text.
\end{document}
```

Online compiler

https://www.tutorialspoint.com/online_latex_editor.php

Sample Example

```
% hello.tex – Hello world LaTeX example

\documentclass{article}
\begin{document}
 Hello World!
\end{document}
```

Title creation

- \documentclass{article}

- \title{My first document}
- \date{2013-09-01}
- \author{John Doe}

- \begin{document}
- \maketitle
- \newpage

- Hello World!
- \end{document}

Different Font Styles

Font Effects

There are LATEX commands for a variety of font eects:

\textit{words in italics} words in italics

- \textsl{words slanted} words slanted
- \textsc{words in smallcaps} words in smallcaps
- \textbf{words in bold} words in bold
- \texttt{words in teletype} words in teletype
- \textsf{sans serif words} sans serif words
- \textrm{roman words} roman words
- \underline{underlined words} underlined words

Colour model

- {\color{colour_name}text}

- Red, green, blue, cyan, magenta, yellow and white .

Lists

```
\begin{enumerate}
\item First thing
\item Second thing
\begin{itemize}
\item A sub-thing
\item Another sub-thing
\end{itemize}
\item Third thing
\end{enumerate}
```

Output:

1. First thing
2. Second thing
    - A sub-thing
    - Another sub-thing
3. Third thing

Tables

- \begin{tabular}{|l|l|}
- Apples & Green \\
- Strawberries & Red \\
- Oranges & Orange \\
- \end{tabular}

Apples	Green
Strawberries	Red
Oranges	Orange

Mathematical Formulas

- $$\sqrt{y^2}$$ produces: 
- $$\sqrt[x]{y^2}$$ produces: 

Eqations

- \documentclass{article}

- \begin{document}

- \begin{equation}
- f(x) = x^2
- \end{equation}

- \end{document}

Creating the table in LaTex

- \documentclass[12pt,twoside,a4paper]{article}
- \begin{document}
- \begin{tabular}{|c|c|c|}
- \hline
- A & B & C \\
- \hline
- 1 & 2 & 3 \\
- \hline
- 4 & 5 & 6
- \\
- \hline
- \end{tabular}
- \end{document}

Packages

- There are countless packages, all for different purposes in my tutorials I will explain some of the most useful. To typeset math, LaTeX offers (among others) n *environment* called *equation.* Everything inside this environment will be printed in *math mode*, a special typesetting environment for math.

Figures

%...

\begin{figure}[h!]

%...

- h (here) - same location
- t (top) - top of page
- b (bottom) - bottom of page
- p (page) - on an extra page
- ! (override) - will force the specified location

Hyperlink

- \usepackage{hyperref}
- \begin{document}
- This is my link: \href{http://www.latex-tutorial.com}{LaTeX-Tutorial}.

ANY QUESTIONS ?

THANK YOU