Overview of Big Data & Analytics using R

Vinodh Krishnaraju

Agenda

Big Data in Industries

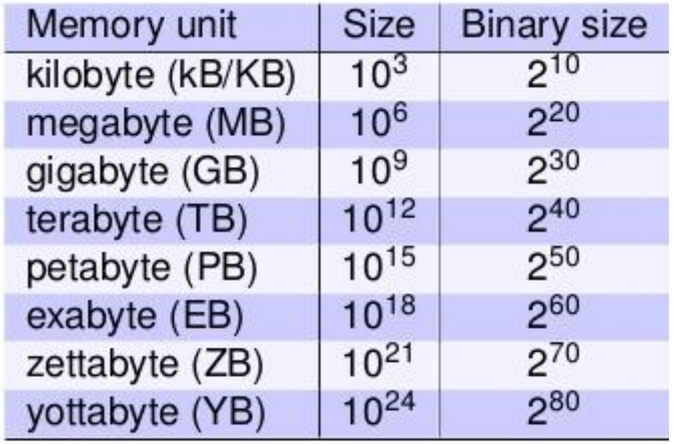
R as an Open source Analytics tool

Data mining using R

Visualisation using R

Simple to start

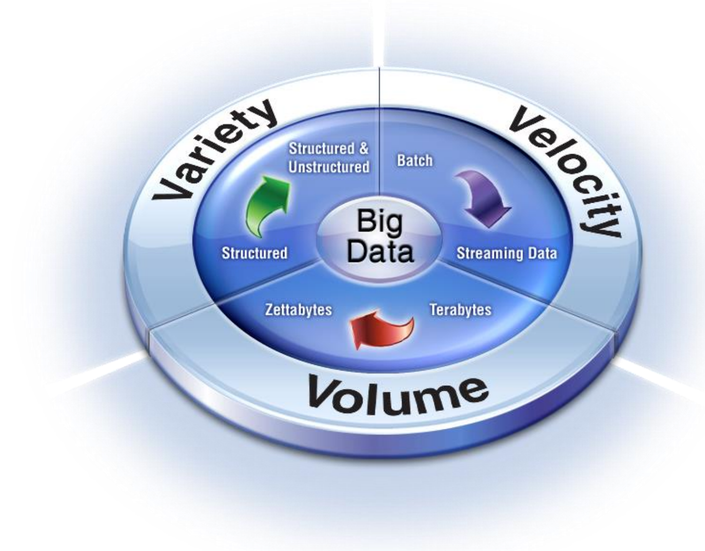
- What is the maximum file size you have dealt so far?
  - Movies/Files/Streaming video that you have used?
  - What have you observed?
- What is the maximum download speed you get?
- Simple computation
  - How much time to just transfer.



Memory unit	Size	Binary size
kilobyte (kB/KB)	10^3	2^{10}
megabyte (MB)	10^6	2^{20}
gigabyte (GB)	10^9	2^{30}
terabyte (TB)	10^{12}	2^{40}
petabyte (PB)	10^{15}	2^{50}
exabyte (EB)	10^{18}	2^{60}
zettabyte (ZB)	10^{21}	2^{70}
yottabyte (YB)	10^{24}	2^{80}

“***Big Data***” is data whose scale, diversity, and complexity require new architecture, techniques, algorithms, and analytics to manage it and extract value and hidden knowledge from it…

Big data is the realization of greater business intelligence by storing, processing, and analysing data that was previously ignored due to the limitations of traditional data management technologies.

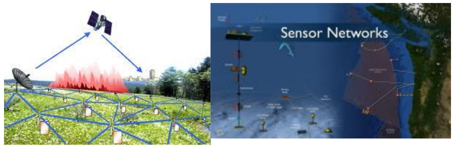


Who’s Generating Big Data

Social media and networks
(all of us are generating data)

Scientific instruments
(collecting all sorts of data)

Mobile devices
(tracking all objects all the time)



Sensor technology and networks
(measuring all kinds of data)

Big Data Everywhere!

Lots of data is being collected

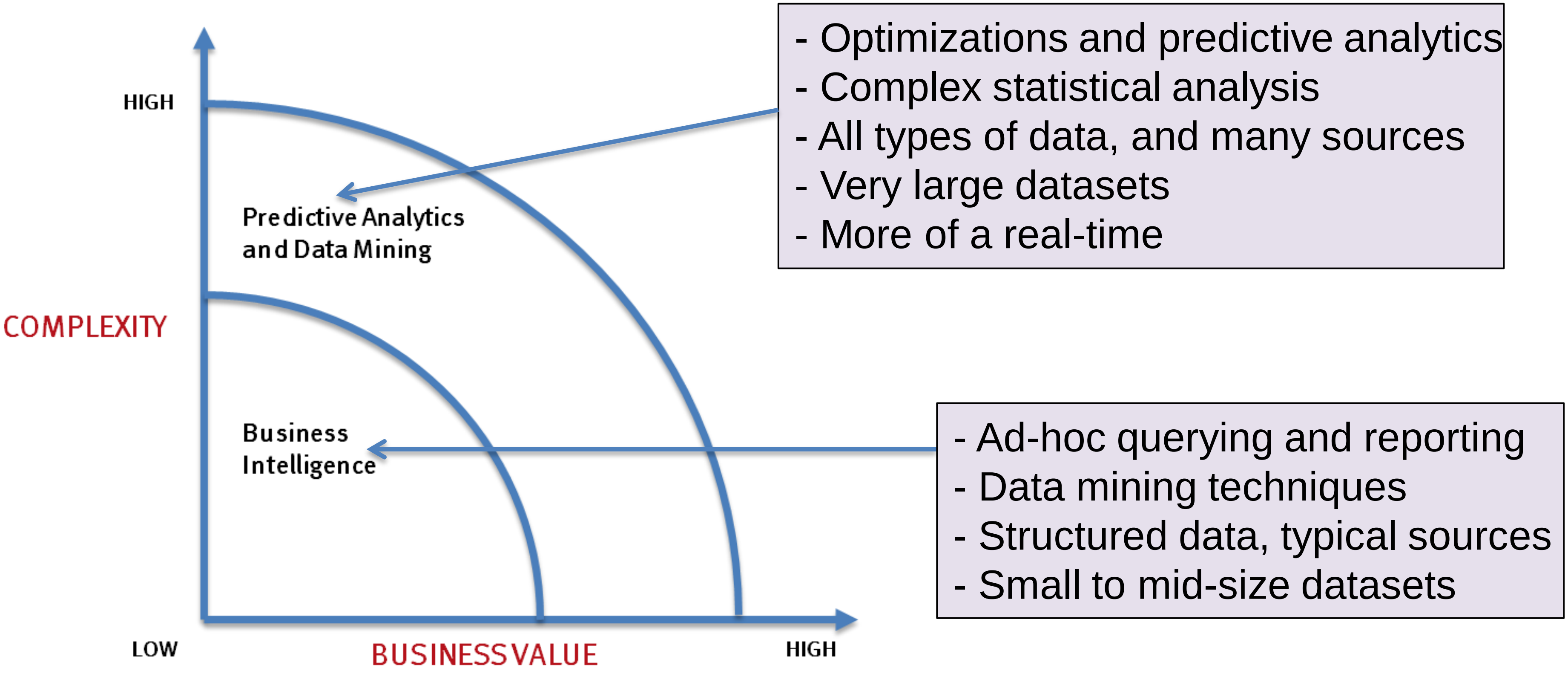
- and warehoused
  - Web data, e-commerce
  - purchases at department/
  - grocery stores
  - Bank/Credit Card
  - transactions
  - Social Network

How much data?

- Google processes 20 PB a day (2008)
- Wayback Machine has 3 PB + 100 TB/month (3/2009)
- Facebook has 2.5 PB of user data + 15 TB/day (4/2009)
- eBay has 6.5 PB of user data + 50 TB/day (5/2009)
- CERN’s Large Hydron Collider (LHC) generates 15 PB a year

640K ought to be enough for anybody.

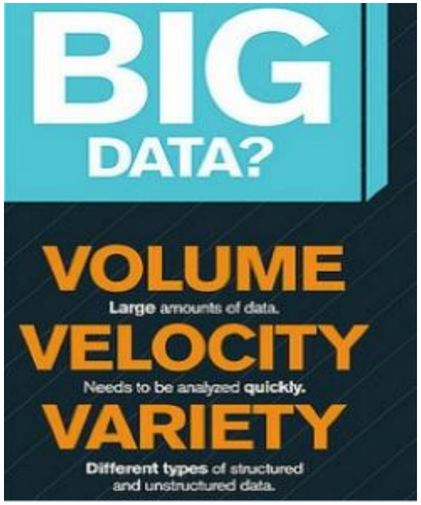
What’s driving Big Data

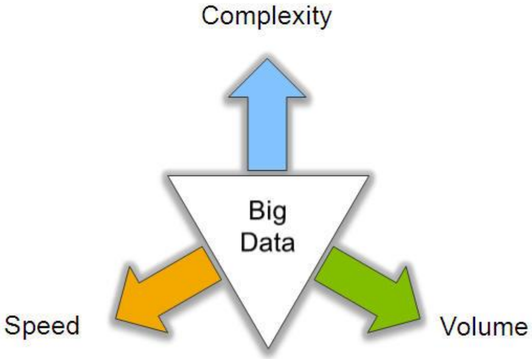


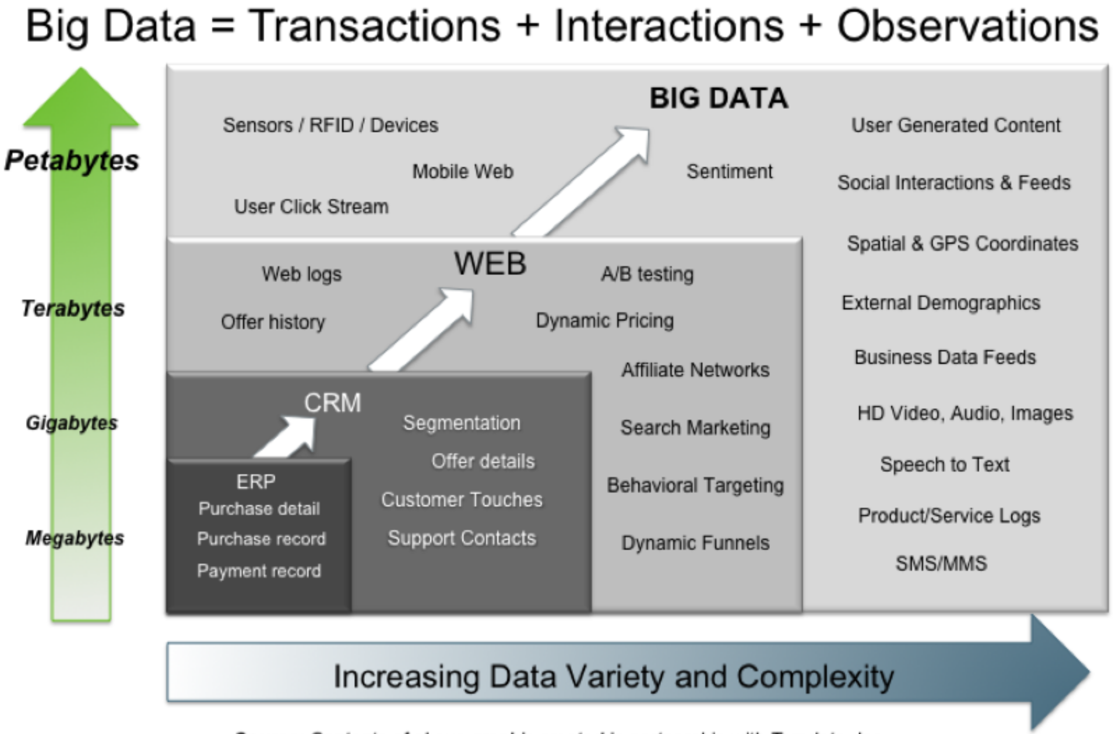
- Optimizations and predictive analytics
- Complex statistical analysis
- All types of data, and many sources
- Very large datasets
- More of a real-time

- Ad-hoc querying and reporting
- Data mining techniques
- Structured data, typical sources
- Small to mid-size datasets

Big Data: 3V's

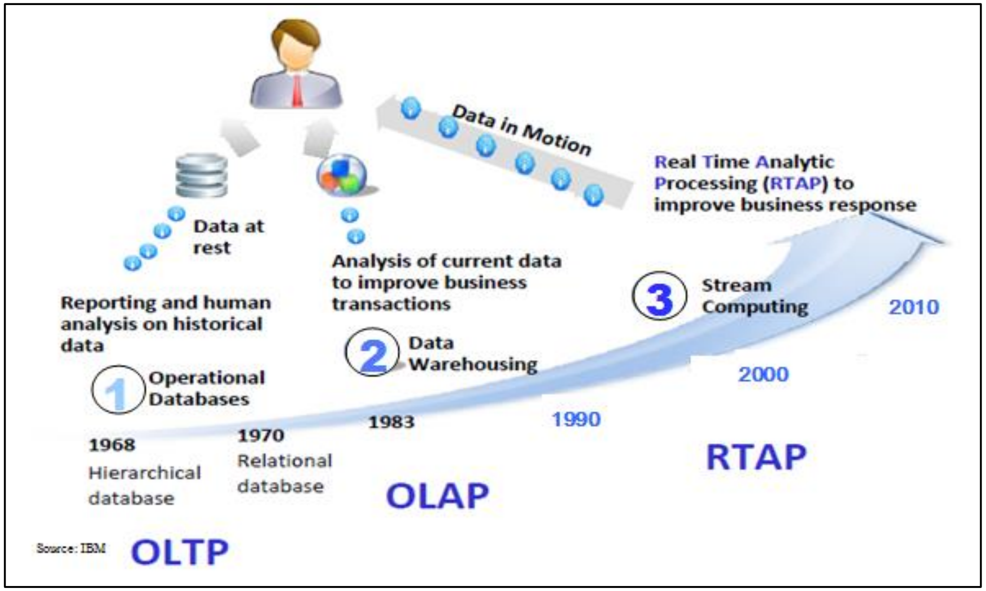

BIG
DATA?
VOLUME
Large amounts of data.
VELOCITY
Needs to be analyzed quickly.
VARIETY
Different types of structured and unstructured data.





Source: *Contents of above graphic created in partnership with Teradata, Inc.*

Harnessing Big Data


Data in Motion
Real Time Analytic Processing (RTAP) to improve business response
Data at rest
Analysis of current data to improve business transactions
Reporting and human analysis on historical data
3 Stream Computing
2010
2 Data Warehousing
2000
1 Operational Databases
1968 Hierarchical database
1970 Relational database
1983
1990
RTAP
OLAP
Source: IBM
OLTP

- **OLTP:** Online Transaction Processing (DBMSs)
- **OLAP:** Online Analytical Processing (Data Warehousing)
- **RTAP:** Real-Time Analytics Processing (Big Data Architecture & technology)

Big Data Landscape

Vertical Apps

- PREDICTIVE POLICING
- bloomreach GET FOUND.
- MYRRIX

Log Data Apps

- splunk>
- loggly
- sumologic

Ad/Media Apps

- rocketfuel
- collective[i]
- bluefin LABS
- Recorded Future
- LuckySort
- Media Science
- TURN
- DataXu Data. Insight. Action.

Data As A Service

- factual.
- kaggle
- knoema beta
- GNIP
- DATASIFT
- Windows Azure Marketplace
- INRIX
- LexisNexis
- SPACE CURVE
- LOQATE Everything Location

Business Intelligence

- ORACLE | Hyperion
- SAP
- Business Objects
- RJMetrics
- Microsoft Business Intelligence
- IBM
- COGNOS
- birst
- MicroStrategy
- Autonomy
- QlikView
- bime
- DOMO
- Chart.io
- GoodData

Analytics and Visualization

- tableau
- Palantir
- OPERA
- metaLayer
- METAMARKETS
- dataspora
- centrifuge
- TERADATA ASTER
- sas
- TIBCO
- KARMASPHERE
- panopticon Real-Time Visual Data Analysis
- pentaho
- Datameer
- platfora
- ClearStory DATA
- CIRRO
- alteryx
- visual.ly
- AYATA

Analytics Infrastructure

- Hortonworks
- VERTICA An HP Company
- MAPR TECHNOLOGIES
- cloudera
- INFOBRIGHT
- PARACCEL
- EMC²
- GREENPLUM.
- NETEZZA
- kognitio
- DATASTAX
- EXASOL
- calpont

Operational Infrastructure

- COUCHBASE
- 10gen | the MongoDB company
- TERADATA
- HADAPT
- TERRACOTTA
- VoltDB
- MarkLogic
- INFORMATICA

Infrastructure As A Service

- amazon web services
- Windows Azure
- infochimps
- Google BigQuery

Structured Databases

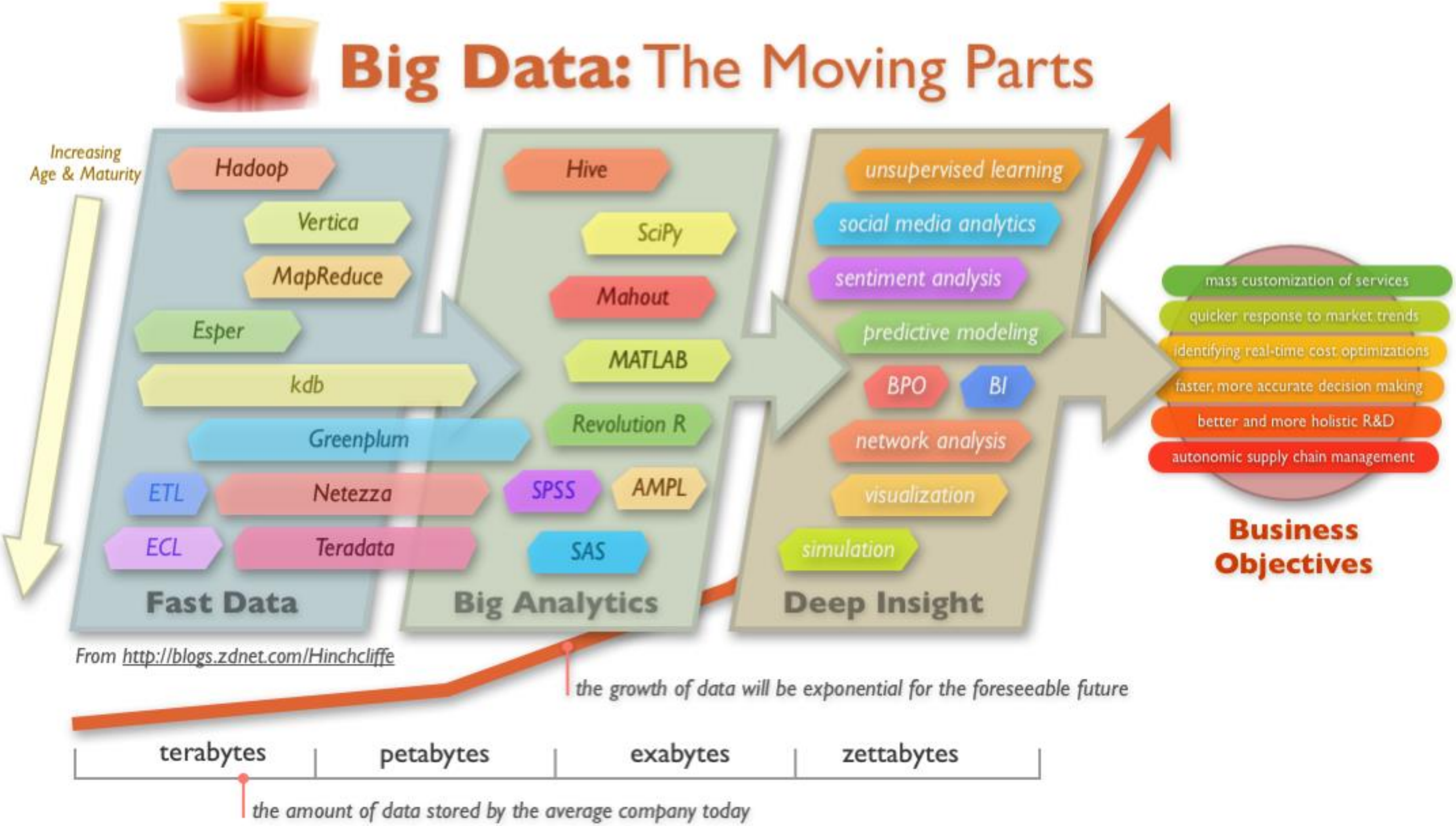
- ORACLE
- MySQL
- Microsoft SQL Server
- PostgreSQL
- IBM DB2.
- SYBASE
- memsql

Technologies

- hadoop
- hadoop Map Reduce
- mahout
- APACHE HBASE
- Cassandra

Copyright © 2012 Dave Feinleib dave@vcdave.com blogs.forbes.com/davefeinleib

Big Data Technology



Big Data: The Moving Parts

Increasing Age & Maturity
Hadoop
Vertica
MapReduce
Esper
kdb
Greenplum
ETL
Netezza
ECL
Teradata
Fast Data
Hive
SciPy
Mahout
MATLAB
Revolution R
SPSS
AMPL
SAS
Big Analytics
unsupervised learning
social media analytics
sentiment analysis
predictive modeling
BPO
BI
network analysis
visualization
simulation
Deep Insight
mass customization of services
quicker response to market trends
identifying real-time cost optimizations
faster, more accurate decision making
better and more holistic R&D
autonomic supply chain management
Business Objectives
the growth of data will be exponential for the foreseeable future
terabytes
petabytes
exabytes
zettabytes
the amount of data stored by the average company today

From http://blogs.zdnet.com/Hinchcliffe

What does Big Data trigger?

- From “Big Data and the Web: Algorithms for Data Intensive Scalable Computing”, Ph.D Thesis, Gianmarco

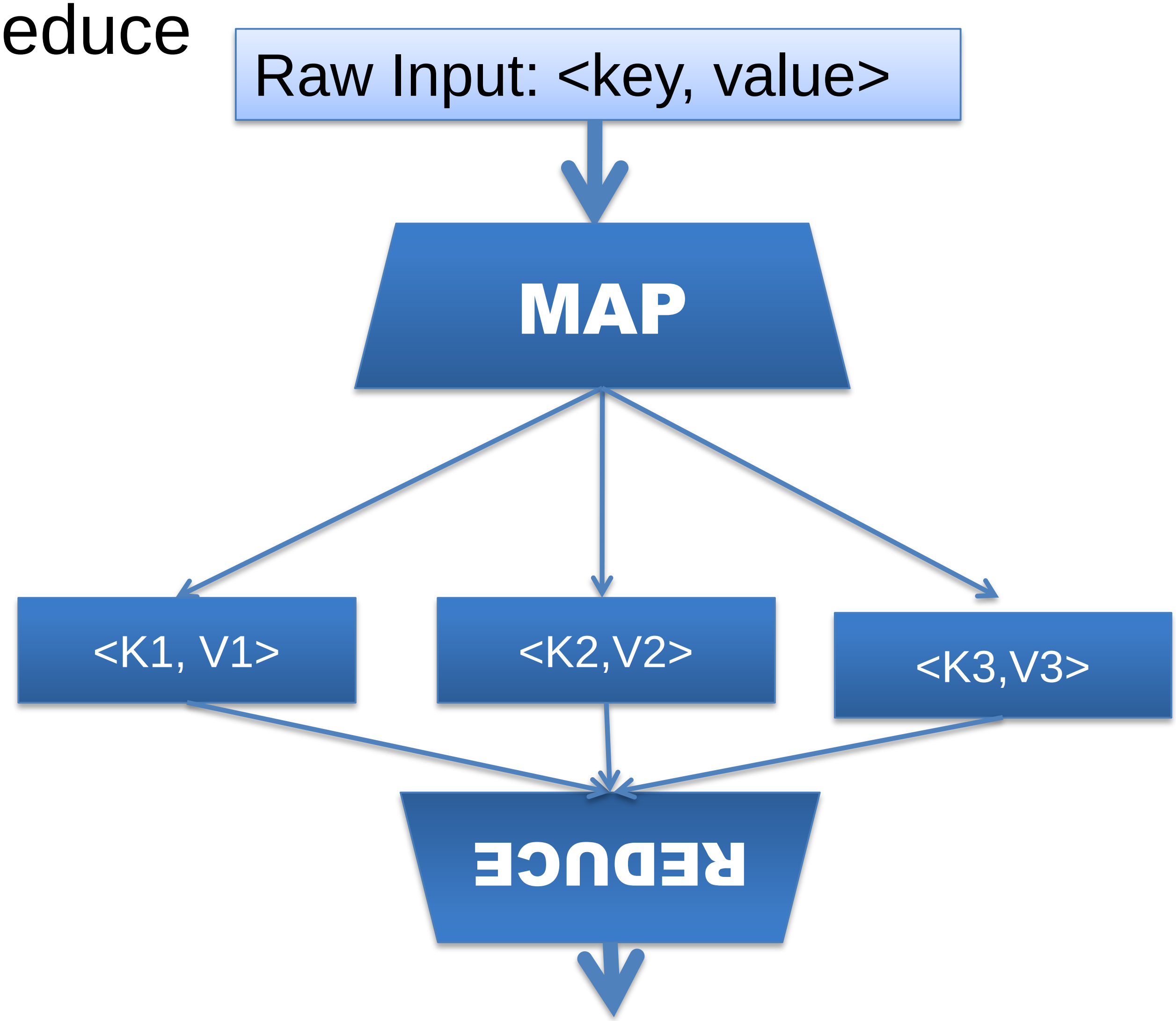
Implementation of Big Data

Platforms for Large-scale Data Analysis

- **Parallel DBMS technologies**
  - Proposed in late eighties
  - Matured over the last two decades
  - Multi-billion dollar industry: Proprietary DBMS Engines intended as Data Warehousing solutions for very large enterprises
- **Map Reduce**
  - pioneered by Google
  - popularized by Yahoo! (Hadoop)

Implementation of Big Data

MapReduce



Raw Input: <key, value>

MAP

<K1, V1>

<K2,V2>

<K3,V3>

REDUCE

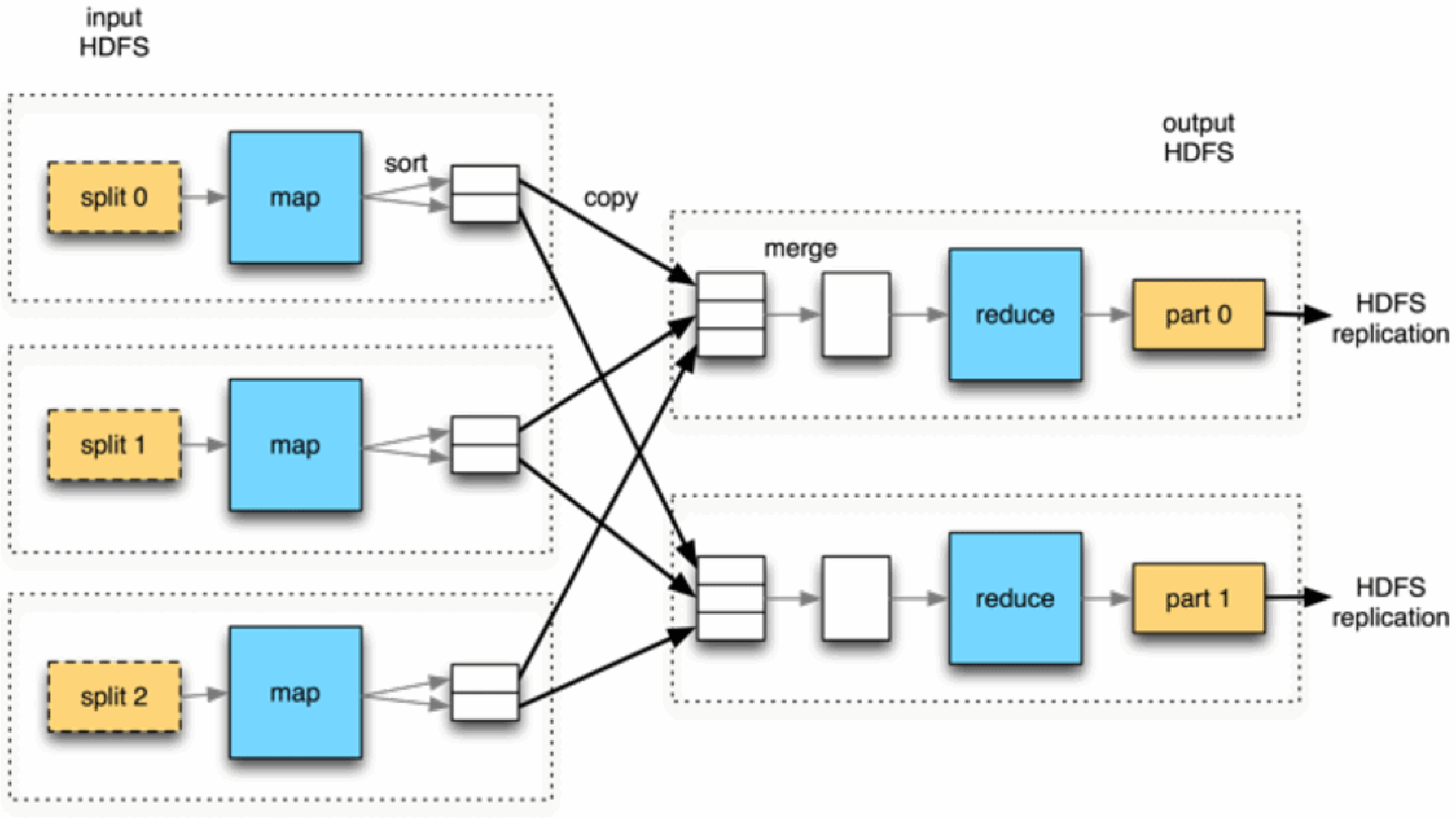
Hadoop

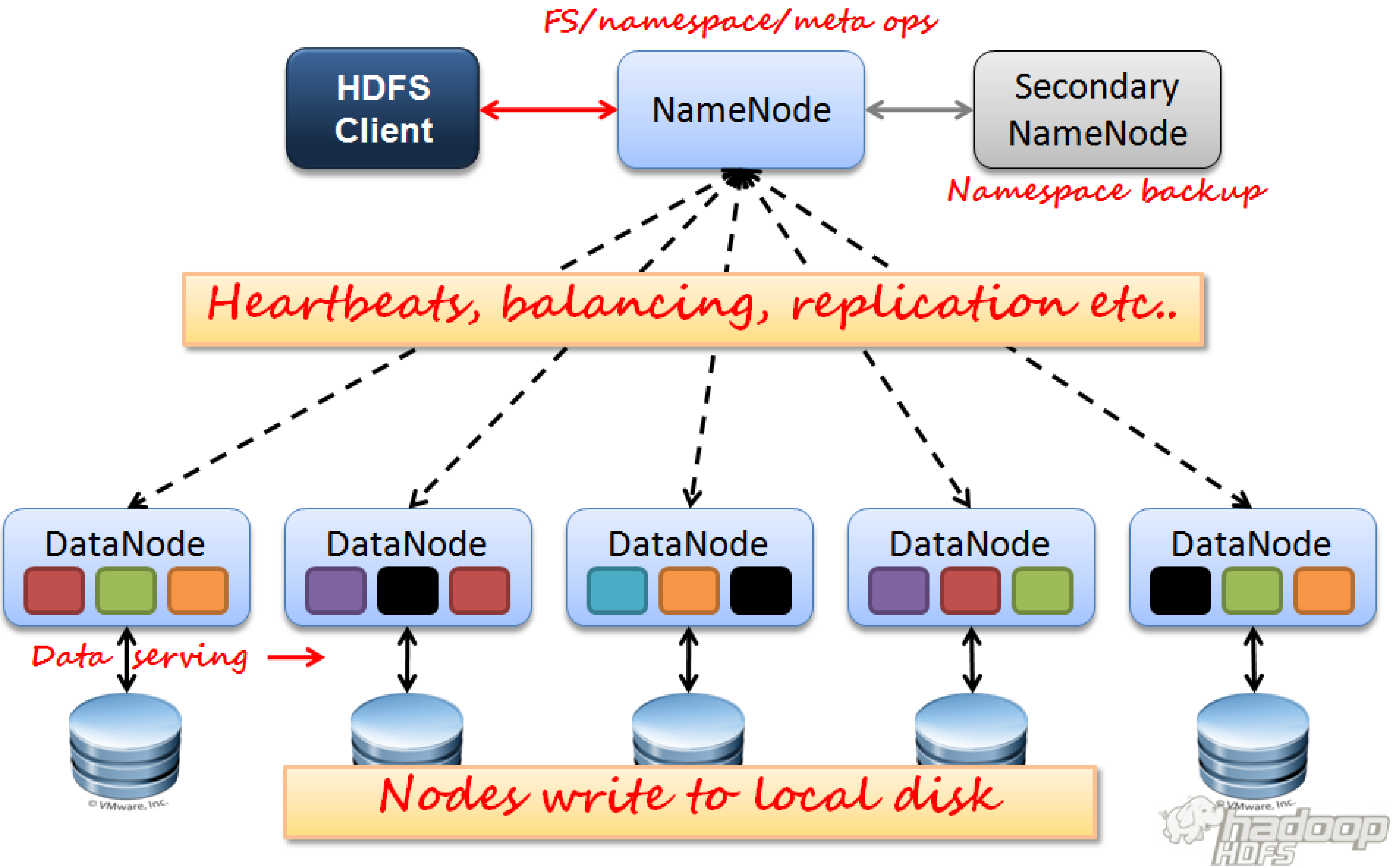
- Hadoop is a distributed file system and data processing engine that is designed to handle extremely high volumes of data in any structure.
- Hadoop has two components:

The Hadoop distributed file system (HDFS), which supports data in structured relational form, in unstructured form, and in any form in between

The MapReduce programing paradigm for managing applications on multiple distributed servers

- The focus is on supporting redundancy, distributed architectures, and parallel processing


input
HDFS
split 0
map
sort
copy
split 1
map
split 2
map
output
HDFS
merge
reduce
part 0
HDFS
replication
reduce
part 1
HDFS
replication



FS/namespace/meta ops

HDFS Client

NameNode

Secondary NameNode

Namespace backup

Heartbeats, balancing, replication etc..

DataNode

DataNode

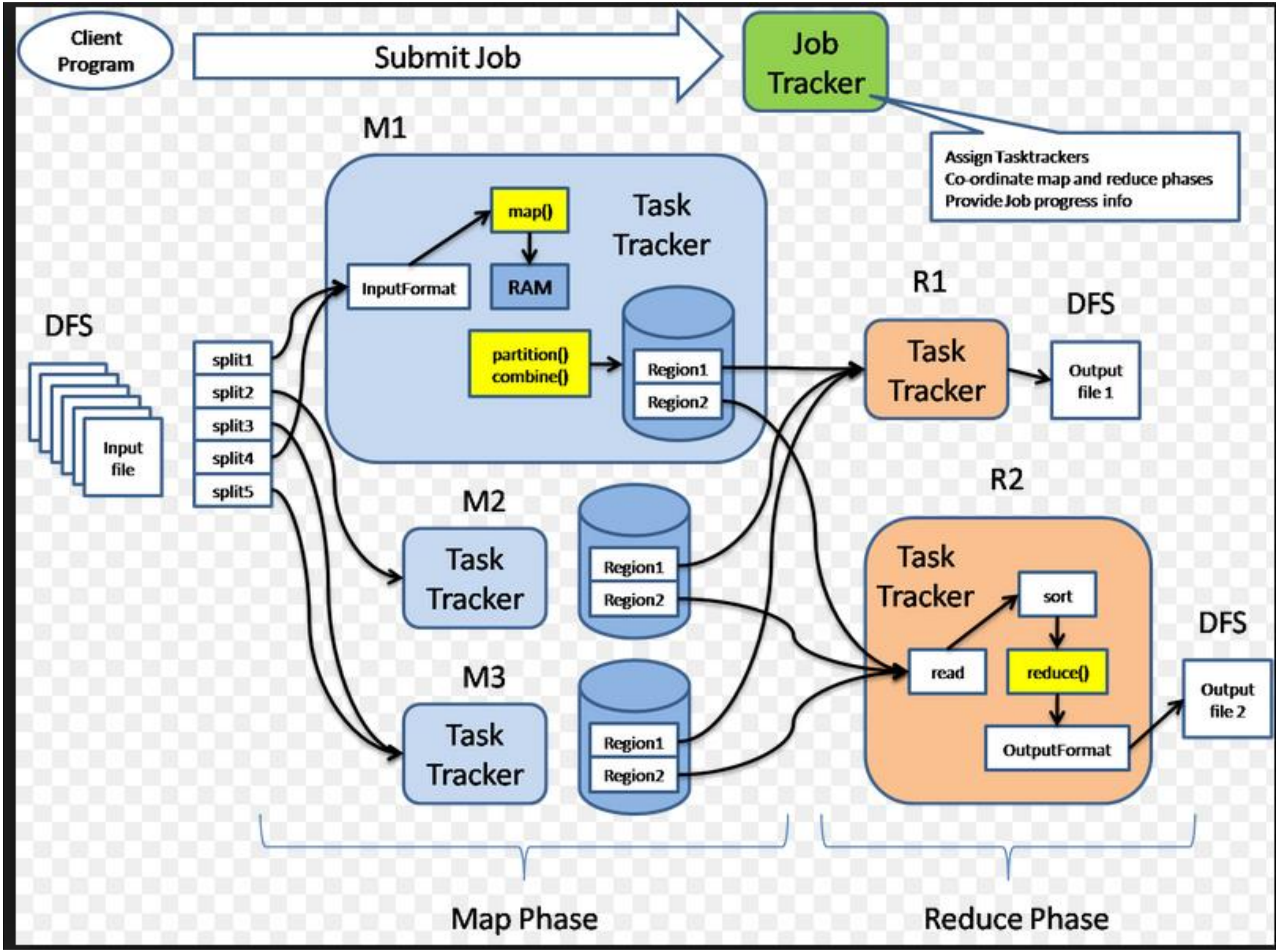
DataNode

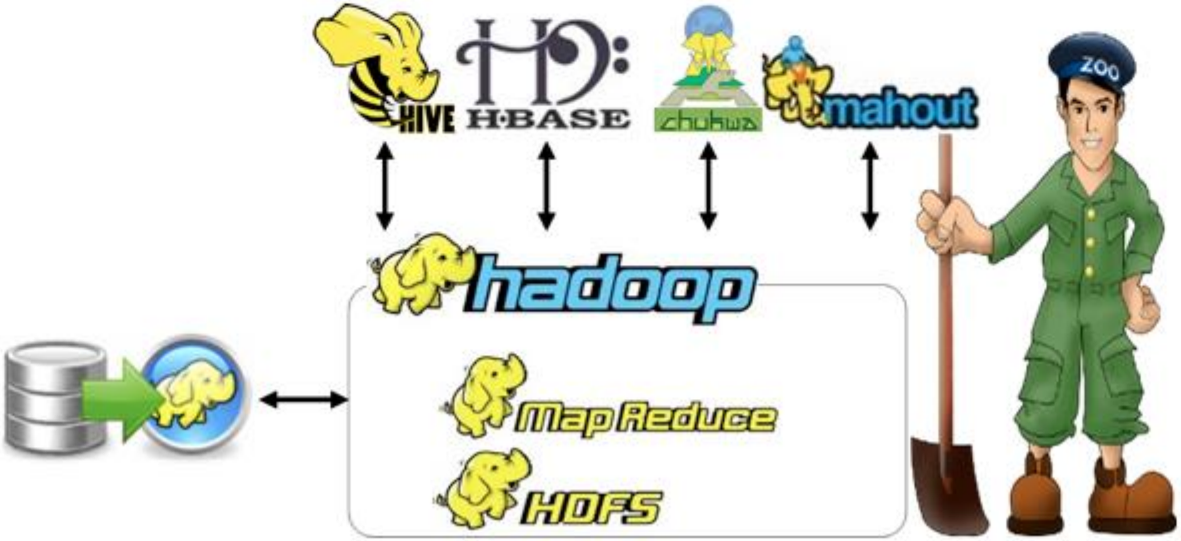
DataNode

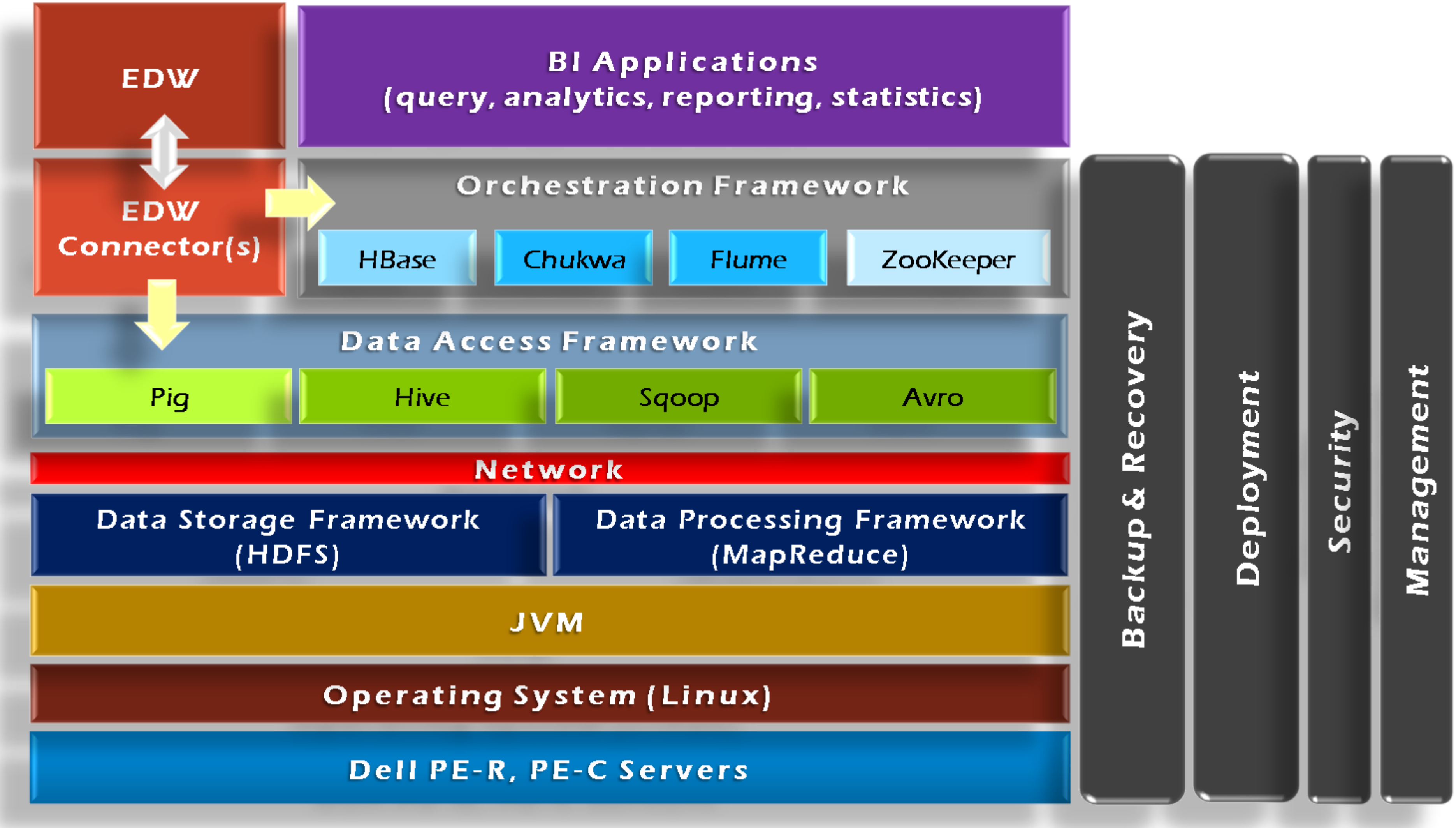
DataNode

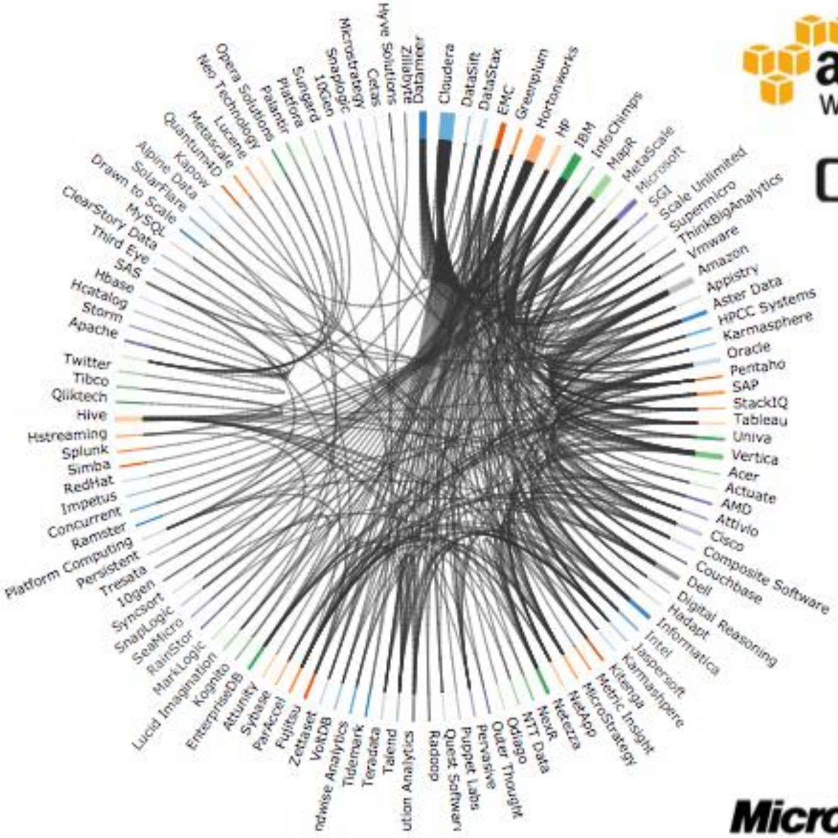
Data serving

Nodes write to local disk


Client Program
Submit Job
Job Tracker
Assign Tasktrackers
Co-ordinate map and reduce phases
Provide Job progress info
M1
Task Tracker
map()
InputFormat
RAM
partition()
combine()
Region1
Region2
DFS
Input file
split1
split2
split3
split4
split5
M2
Task Tracker
Region1
Region2
M3
Task Tracker
Region1
Region2
R1
Task Tracker
DFS
Output file 1
R2
Task Tracker
sort
read
reduce()
OutputFormat
DFS
Output file 2
Map Phase
Reduce Phase


HIVE
H·BASE
chukwa
mahout
ZOO
hadoop
Map Reduce
HDFS


EDW
EDW Connector(s)
BI Applications (query, analytics, reporting, statistics)
Orchestration Framework
HBase
Chukwa
Flume
ZooKeeper
Data Access Framework
Pig
Hive
Sqoop
Avro
Network
Data Storage Framework (HDFS)
Data Processing Framework (MapReduce)
JVM
Operating System (Linux)
Dell PE-R, PE-C Servers
Backup & Recovery
Deployment
Security
Management



amazon webservices™

cloudera

DataStax

EMC²
where information lives®

Hortonworks

IBM

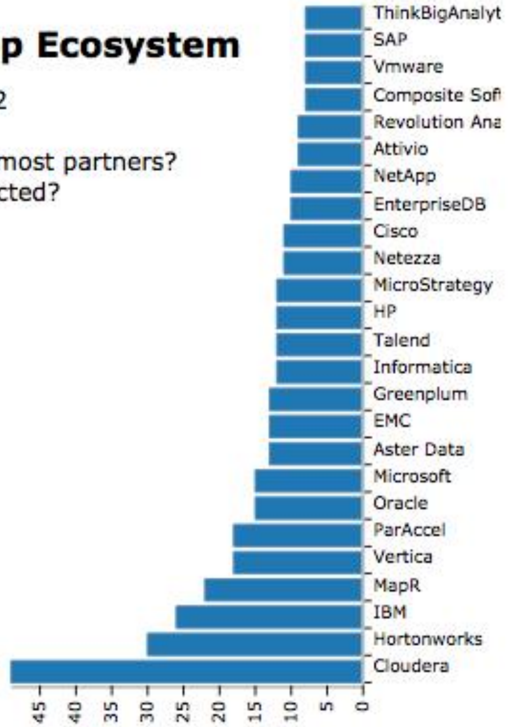
MAPR TECHNOLOGIES

Microsoft

The Hadoop Ecosystem

June 21, 2012

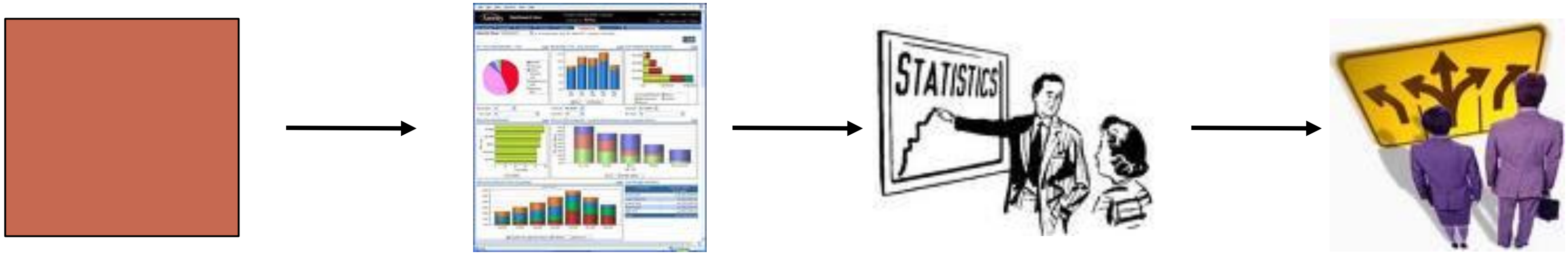
Who has the most partners?
Who is connected?



brought to you by Datameer
Powerfully Simple™

What is Business Intelligence?

Business Intelligence enables the business to make intelligent, fact-based decisions



Aggregate Data

Database, Data Mart, Data Warehouse, ETL Tools, Integration Tools

Present Data

Reporting Tools, Dashboards, Static Reports, Mobile Reporting, OLAP Cubes

Enrich Data

Add Context to Create Information, Descriptive Statistics, Benchmarks, Variance to Plan or LY

Inform a Decision

Decisions are Fact-based and Data-driven

Major BI Trends

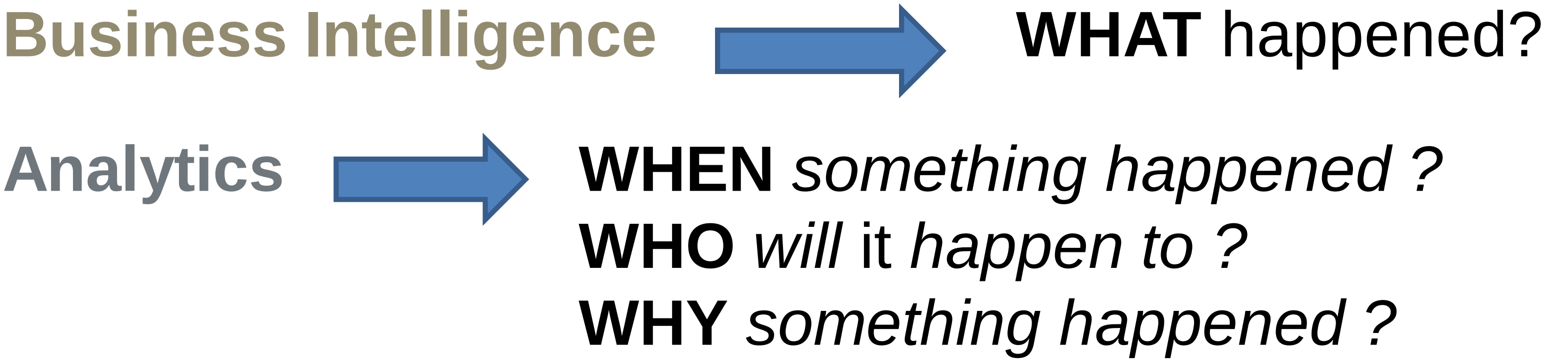
Mobile

•Cloud

•Social Media

•Advanced Analytics

Analytics and BI – the LINK



Business Intelligence → **WHAT** happened?

Analytics →

WHEN *something happened ?*

WHO *will* it *happen to ?*

WHY *something happened ?*

Analytics and BI – the LINK

- **Data: petabytes**
- **Reports: terabytes**
- **Excel: gigabytes**
- **PowerPoint: megabytes**
- **Analytics: bytes**

Business decision based on Analytics

Companies using Analytics

- Amazon
- Netflix
- Harrah
- FEDEX, UPS….
- Citibank, Bank of America, Barclays….
- American Airlines
- FBI, CIA, US Armed Forces…
- Walmart

Application Areas

Industry	**Application**
Finance	Credit Card Analysis
Insurance	Claims, Fraud Analysis
Telecommunication	Call record analysis
Transport	Logistics management
Consumer goods	promotion analysis
Data Service provider	Value added data
Utilities	Power usage analysis

Applications

- Banking: loan/credit card approval
- Customer relationship management:
- Targeted marketing
- Fraud detection: telecommunications, finance
- Manufacturing and production
- Medicine
- Molecular/Pharmaceutical
- Scientific data analysis:
- Web site/store design and promotion:

Relationship with other fields

- Analytics overlaps with data mining, machine learning, statistics, artificial intelligence, databases, visualization
- Stresses on
    - scalability of number of features and instances
    - stress on algorithms and architectures provided by statistics and machine learning.
    - automation for handling large, heterogeneous data

Analytics Tasks

- Classification [Predictive]
- Clustering [Descriptive]
- Association Rule Discovery [Descriptive]
- Sequential Pattern Discovery [Descriptive]
- Regression [Predictive]
- Deviation Detection [Predictive]
- Collaborative Filter [Predictive]

Introduction to R

- R is a statistical analysis package
- It has all of the standard statistical tests, models, and analyses, providing a comprehensive language for managing and manipulating data.
- R is free and open source software, allowing anyone to use and modify it.
- R has over 4800 packages available from multiple repositories.

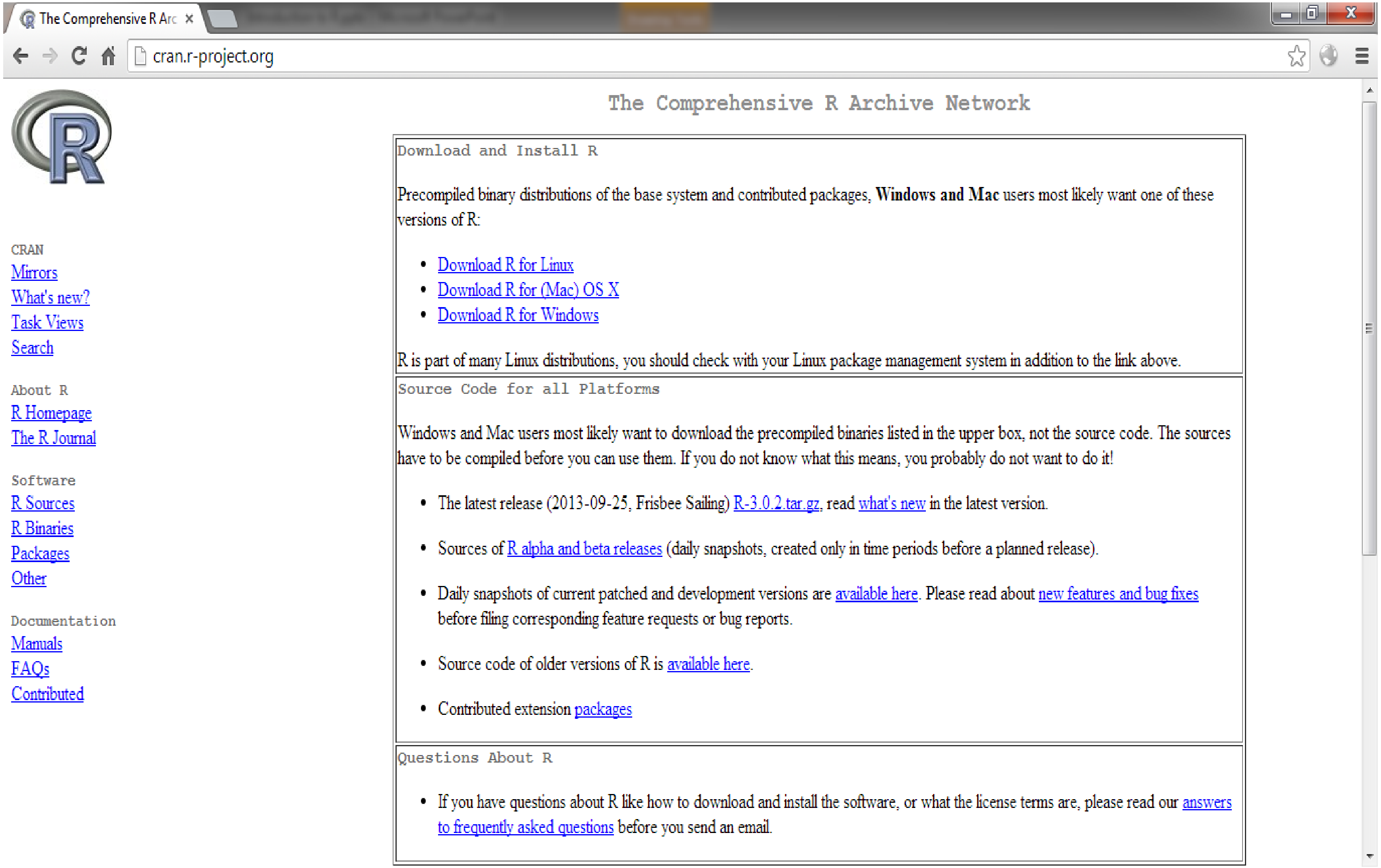
Introduction to R

- Topics like econometrics, data mining, spatial analysis, and bio-informatics.
- R is cross-platform and runs on many operating systems and different hardware.
- R has a vast community and many networking channels with support provided by the very people who developed the environment

Fundamentals of the R

- Base R and most R packages are available for download from   cran.r-project.org .
- Source codes for all platforms are available(Windows,Linux,Mac).

- Download Rstudio, an Integrated Development Environment.(IDE)



The Comprehensive R Archive Network

CRAN
Mirrors
What's new?
Task Views
Search

About R
R Homepage
The R Journal

Software
R Sources
R Binaries
Packages
Other

Documentation
Manuals
FAQs
Contributed

Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- Download R for Linux
- Download R for (Mac) OS X
- Download R for Windows

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

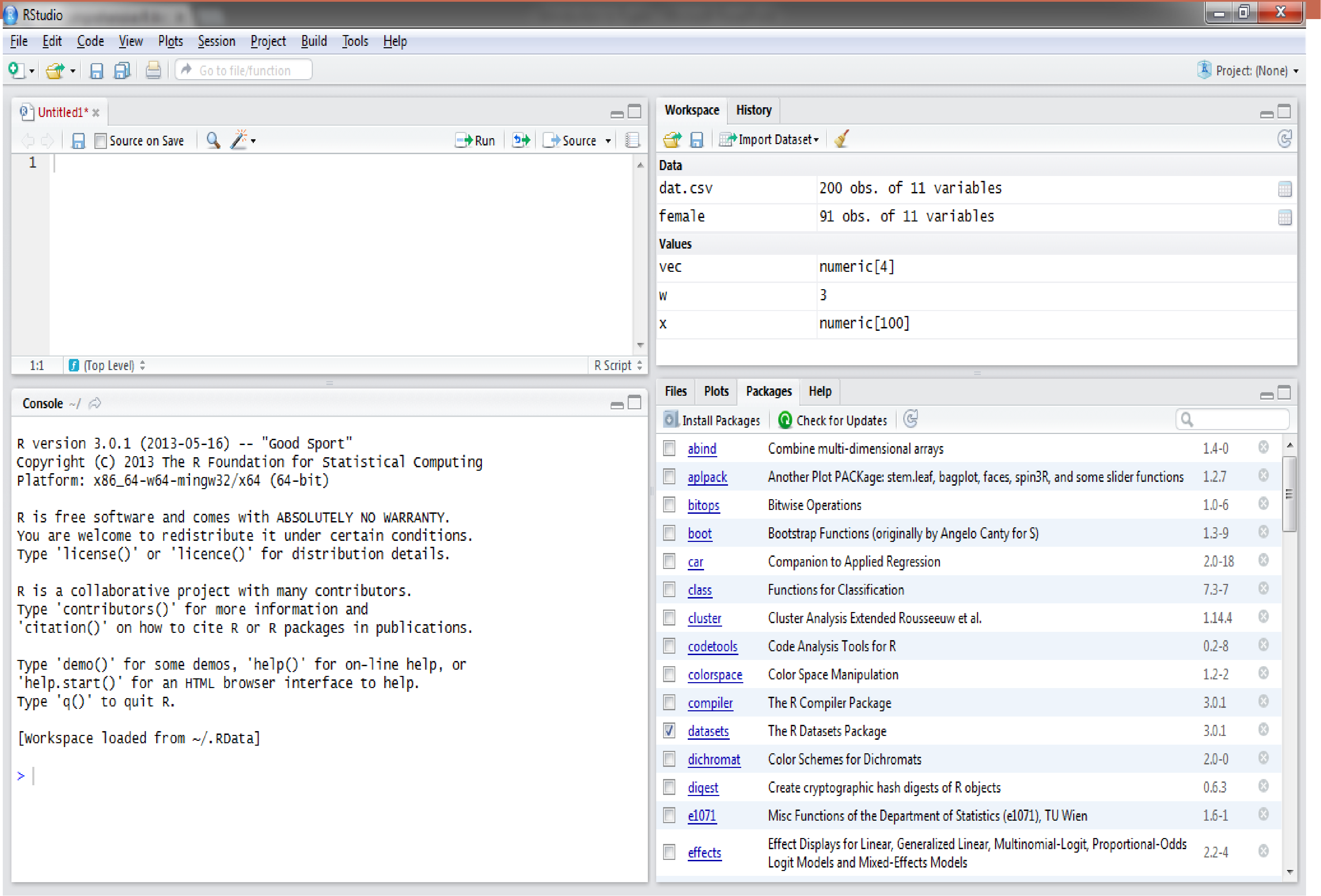
Source Code for all Platforms

Windows and Mac users most likely want to download the precompiled binaries listed in the upper box, not the source code. The sources have to be compiled before you can use them. If you do not know what this means, you probably do not want to do it!

- The latest release (2013-09-25, Frisbee Sailing) R-3.0.2.tar.gz, read what's new in the latest version.
- Sources of R alpha and beta releases (daily snapshots, created only in time periods before a planned release).
- Daily snapshots of current patched and development versions are available here. Please read about new features and bug fixes before filing corresponding feature requests or bug reports.
- Source code of older versions of R is available here.
- Contributed extension packages

Questions About R

- If you have questions about R like how to download and install the software, or what the license terms are, please read our answers to frequently asked questions before you send an email.



Data	
dat.csv	200 obs. of 11 variables
female	91 obs. of 11 variables
Values	
vec	numeric[4]
w	3
x	numeric[100]

```
R version 3.0.1 (2013-05-16) -- "Good Sport"
Copyright (C) 2013 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Workspace loaded from ~/.RData]

> 
```

☐	abind	Combine multi-dimensional arrays	1.4-0
☐	aplpack	Another Plot PACKage: stem.leaf, bagplot, faces, spin3R, and some slider functions	1.2.7
☐	bitops	Bitwise Operations	1.0-6
☐	boot	Bootstrap Functions (originally by Angelo Canty for S)	1.3-9
☐	car	Companion to Applied Regression	2.0-18
☐	class	Functions for Classification	7.3-7
☐	cluster	Cluster Analysis Extended Rousseeuw et al.	1.14.4
☐	codetools	Code Analysis Tools for R	0.2-8
☐	colorspace	Color Space Manipulation	1.2-2
☐	compiler	The R Compiler Package	3.0.1
☑	datasets	The R Datasets Package	3.0.1
☐	dichromat	Color Schemes for Dichromats	2.0-0
☐	digest	Create cryptographic hash digests of R objects	0.6.3
☐	e1071	Misc Functions of the Department of Statistics (e1071), TU Wien	1.6-1
☐	effects	Effect Displays for Linear, Generalized Linear, Multinomial-Logit, Proportional-Odds Logit Models and Mixed-Effects Models	2.2-4

R as a calculator

```
> (17*0.35)^(1/3)
[1] 1.812059

> log(10)
[1] 2.302585

> exp(1)
[1] 2.718282

> 3^-1
[1] 0.3333333
```

Vectors

- Typical operations on vectors include summary statistics (mean, var, range, max,…)

```
> y<-c(5,7,7,8,2,5,6,6,7,5,8,3,4)
> z<-13:1
> mean(y)
[1] 5.615385

> var(z)
[1] 15.16667
```

- Arithmetic with entire vectors, e.g. * operator. In R if two vectors are not the same length, the shorter vector is repeated as necessary, up to the length of the longer vector:

```
> y*6
[1] 30 42 42 48 12 30 36 36 42 30 48 18 24
```

- Join together two vectors using the concatenate function c() :

```
> c(y,z)
```

Obtaining Parts of Vectors

- Elements of vectors by subscripts in []:

```
> y[3]
```

- The third to the seventh elements of y:

```
> y[3:7]
```

- The third, fifth, sixth and ninth elements:

```
> y[c(3,5,6,9)]
```

- To drop an element from the array, use negative subscripts:

```
> y[-1]
```

- To drop the last element of the array without knowing its length:

```
> y[-length(y)]
```

Lists

- Lists are vectors with different classes of objects.
- Lists are subscribed like this `[[3]]`: list called “cars”, with
- three elements: “make”, “capacity” and “color”:

```
> cars<-list(c("Toyota","Nissan","Honda"),c(150,180,50))
>  cars[[1]]

[1] "Toyota" "Nissan" "Honda"

> cars[[2]]

[1] 150 180  50
```

- To extract one element of the sub-list:

```
> cars[[2]][[2]]

[1] 180
```

Matrices

Create a matrix: (Matrix creation is column wise by default)

```
>matrix(1:6,2,3)
```

```
 [,1] [,2] [,3]
[1,]   1   3   5
[2,]   2   4   6
```

Matrix from a vector:

```
>m2=matrix(1:3)        #R fills column wise
> m2
```

```
     [,1]
[1,]   1
[2,]   2
[3,]   3
```

```
>dim(m2)=c(1,3)        #Change dimensionality
>m2
```

```
     [,1] [,2] [,3]
[1,]   1   2   3
```

Data Frames

- It is similar to matrices but store different classes of objects.
- It is usually called with read.table().

Create a dataframe:

```
>d=data.frame(subjectID=1:5,gender=c("M","F","F","M","F")
,score=c(8,3,6,5,5))
>head(d)
```

```
   subjectID gender score
1        1      M     8
2        2      F     3
3        3      F     6
4        4      M     5
5        5      F     5
```

Number of rows:

```
>nrow(d)
[1] 5
```

Number if columns:

```
>ncol(d)
[1] 3
```

Data Frames

Check the attributes:

```
>attributes(d)
$names
[1] "subjectID" "gender"    "score"

$row.names
[1] 1 2 3 4 5

$class
[1] "data.frame"
```

Call a particular cell in the dataframe

```
>d[2,1]
[1] 2
>summary(d)
   subjectID gender     score
 Min.   :1   F:3    Min.   :3.0
 1st Qu.:2   M:2    1st Qu.:5.0
 Median :3          Median :5.0
 Mean   :3          Mean   :5.4
 3rd Qu.:4          3rd Qu.:6.0
 Max.   :5          Max.   :8.0
```

Display a dataframe:

>View(d)

Edit a dataframe :

>edit(d)

Getting help on a function:

>?functionname

(Eg) ?data.frame

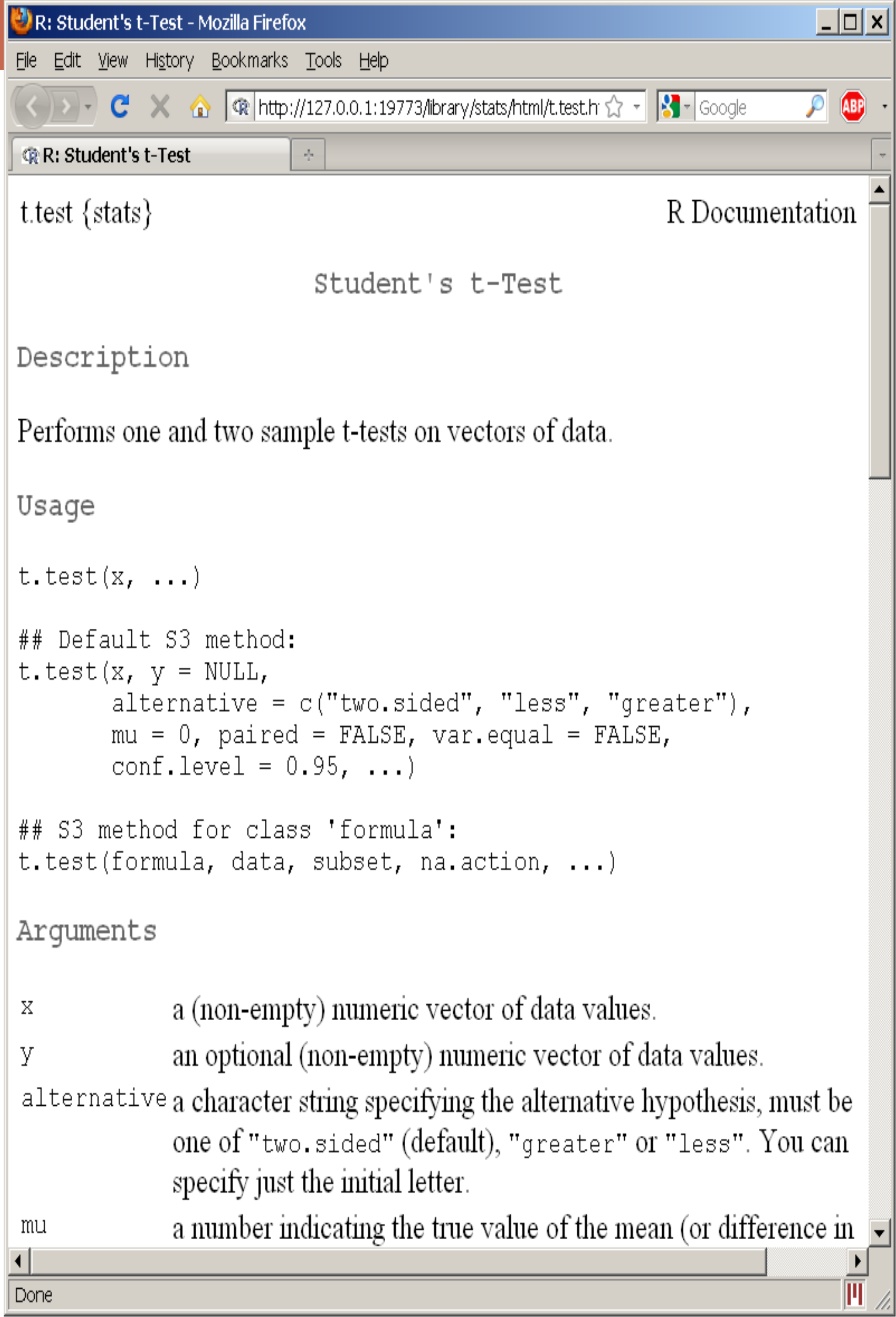
>help(t.test)

Save objects

>save(x,y,file=“xy.RData”)

>save.image()

You can save History File


R: Student's t-Test - Mozilla Firefox
File
Edit
View
History
Bookmarks
Tools
Help
http://127.0.0.1:19773/library/stats/html/t.test.h
Google
R: Student's t-Test
Done

t.test {stats} R Documentation

Student's t-Test

Description

Performs one and two sample t-tests on vectors of data.

Usage

```
t.test(x, ...)

## Default S3 method:
t.test(x, y = NULL,
       alternative = c("two.sided", "less", "greater"),
       mu = 0, paired = FALSE, var.equal = FALSE,
       conf.level = 0.95, ...)

## S3 method for class 'formula':
t.test(formula, data, subset, na.action, ...)
```

Arguments

`x` a (non-empty) numeric vector of data values.

`y` an optional (non-empty) numeric vector of data values.

`alternative` a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater" or "less". You can specify just the initial letter.

`mu` a number indicating the true value of the mean (or difference in

Installation of packages

Download and install packages:

```
>install.packages(“psych”)
```

-Need to specify the CRAN the first time.
-CRAN:- Comprehensive R Archive Network.

Load packages:

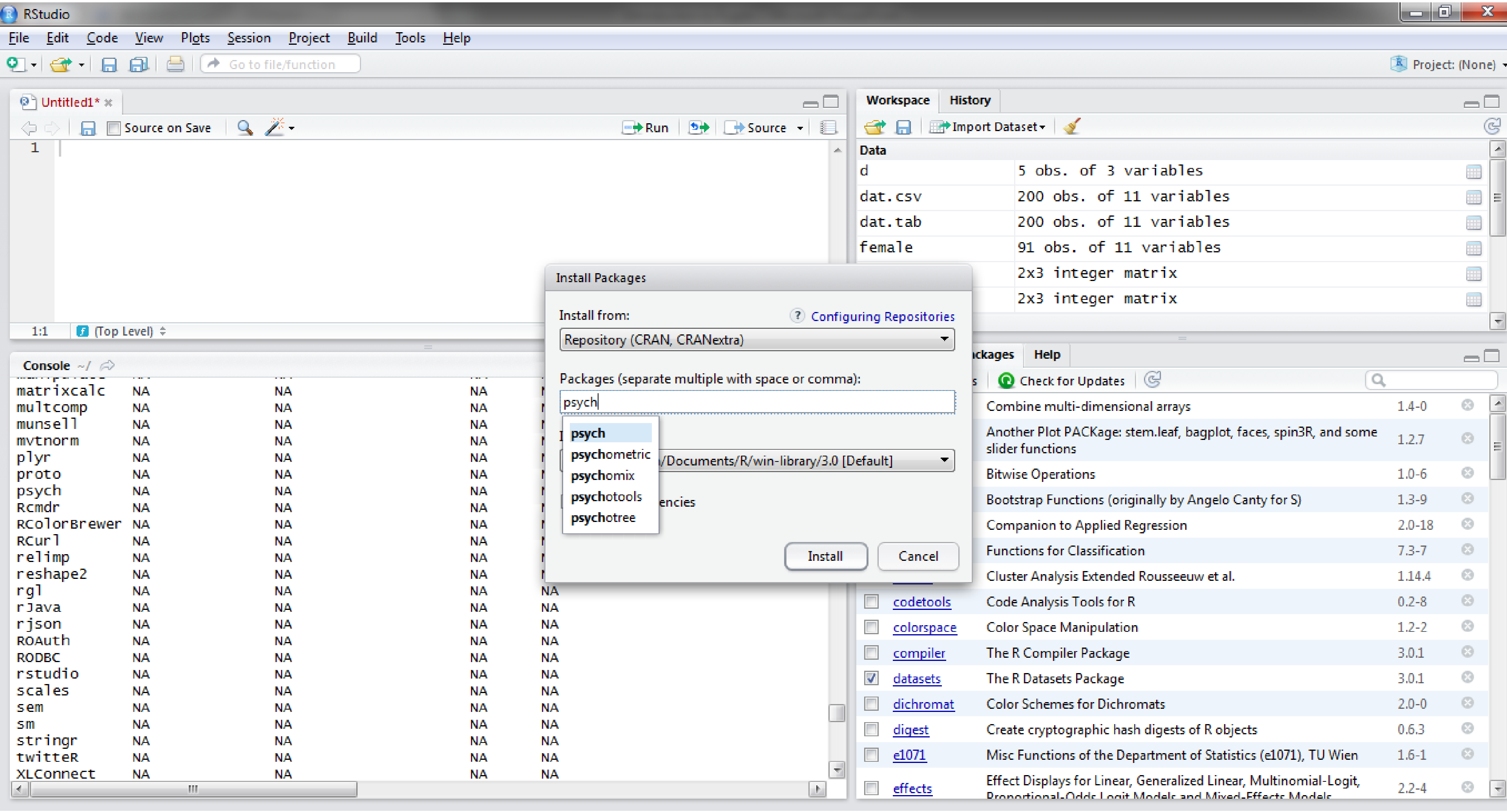
```
>library(psych)
```

To view the loaded packages in R:

```
>search()
```

or

```
>installed.packages()
```



```
1
```

1:1

Install Packages

Install from: Configuring Repositories

Repository (CRAN, CRANextra)

Packages (separate multiple with space or comma):

psych

- **psych**
- **psych**ometric
- **psych**omix
- **psych**otools
- **psych**otree

/Documents/R/win-library/3.0 [Default]

encies

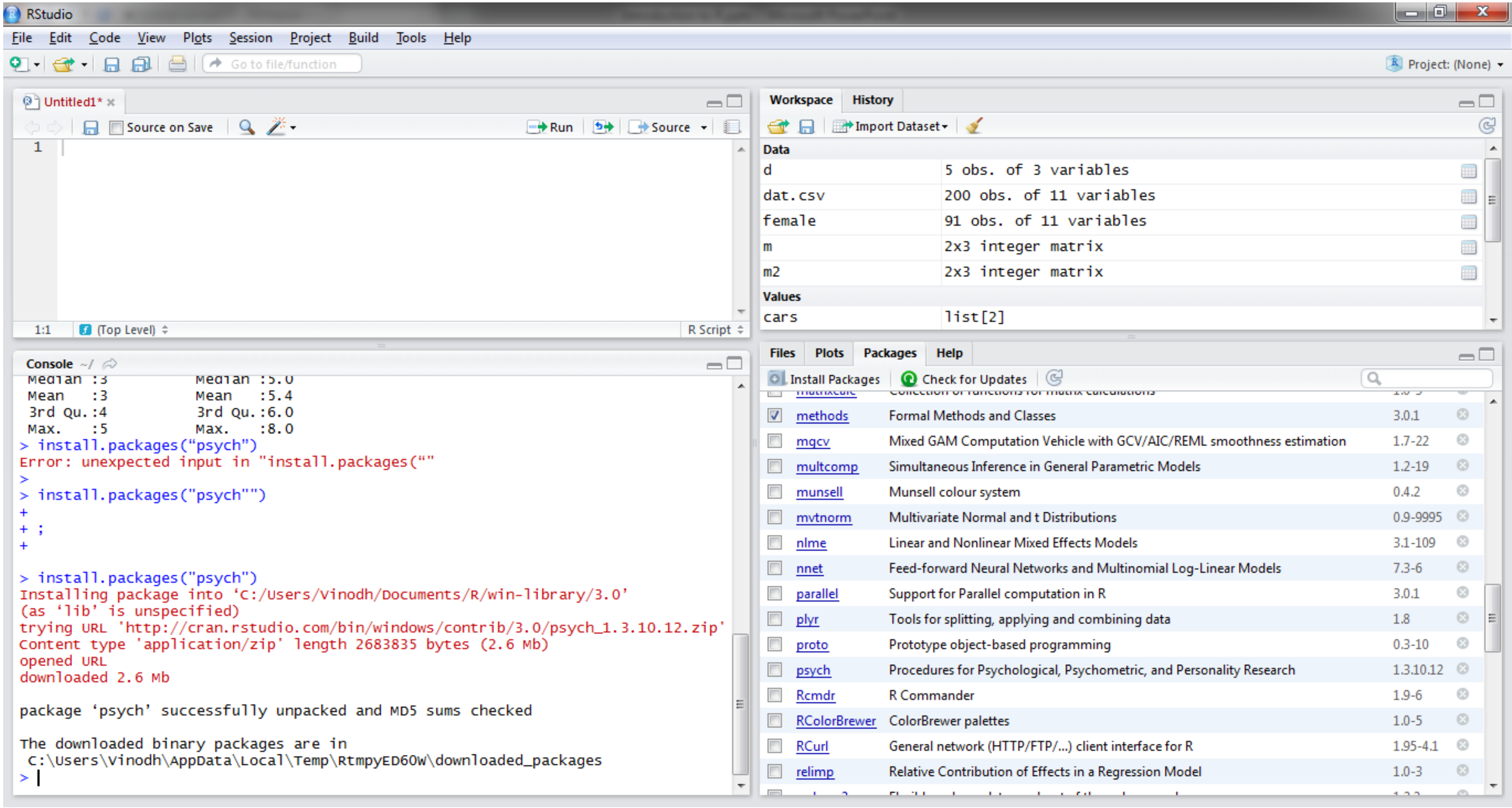
Install Cancel

Data

d	5 obs. of 3 variables
dat.csv	200 obs. of 11 variables
dat.tab	200 obs. of 11 variables
female	91 obs. of 11 variables
	2x3 integer matrix
	2x3 integer matrix

```
matrixcalc   NA          NA            NA        N
multcomp     NA          NA            NA        N
munsell      NA          NA            NA        N
mvtnorm      NA          NA            NA        N
plyr         NA          NA            NA        N
proto        NA          NA            NA        N
psych        NA          NA            NA        N
Rcmdr        NA          NA            NA        N
RColorBrewer NA          NA            NA        N
RCurl        NA          NA            NA        N
relimp       NA          NA            NA        N
reshape2     NA          NA            NA        N
rgl          NA          NA            NA        NA
rJava        NA          NA            NA        NA
rjson        NA          NA            NA        NA
ROAuth       NA          NA            NA        NA
RODBC        NA          NA            NA        NA
rstudio      NA          NA            NA        NA
scales       NA          NA            NA        NA
sem          NA          NA            NA        NA
sm           NA          NA            NA        NA
stringr      NA          NA            NA        NA
twitteR      NA          NA            NA        NA
XLConnect    NA          NA            NA        NA
```

		Combine multi-dimensional arrays	1.4-0
		Another Plot PACKage: stem.leaf, bagplot, faces, spin3R, and some slider functions	1.2.7
		Bitwise Operations	1.0-6
		Bootstrap Functions (originally by Angelo Canty for S)	1.3-9
		Companion to Applied Regression	2.0-18
		Functions for Classification	7.3-7
		Cluster Analysis Extended Rousseeuw et al.	1.14.4
□	codetools	Code Analysis Tools for R	0.2-8
□	colorspace	Color Space Manipulation	1.2-2
□	compiler	The R Compiler Package	3.0.1
☑	datasets	The R Datasets Package	3.0.1
□	dichromat	Color Schemes for Dichromats	2.0-0
□	digest	Create cryptographic hash digests of R objects	0.6.3
□	e1071	Misc Functions of the Department of Statistics (e1071), TU Wien	1.6-1
□	effects	Effect Displays for Linear, Generalized Linear, Multinomial-Logit, Proportional-Odds Logit Models and Mixed-Effects Models	2.2-4



Data

d	5 obs. of 3 variables
dat.csv	200 obs. of 11 variables
female	91 obs. of 11 variables
m	2x3 integer matrix
m2	2x3 integer matrix

Values

cars	list[2]

```
 Median :3          Median :5.0
 Mean   :3          Mean   :5.4
 3rd Qu.:4          3rd Qu.:6.0
 Max.   :5          Max.   :8.0
> install.packages("psych")
Error: unexpected input in "install.packages(""
>
> install.packages("psych"")
+
+ ;
+

> install.packages("psych")
Installing package into 'C:/Users/Vinodh/Documents/R/win-library/3.0'
(as 'lib' is unspecified)
trying URL 'http://cran.rstudio.com/bin/windows/contrib/3.0/psych_1.3.10.12.zip'
Content type 'application/zip' length 2683835 bytes (2.6 Mb)
opened URL
downloaded 2.6 Mb

package 'psych' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
 C:\Users\Vinodh\AppData\Local\Temp\RtmpyED60W\downloaded_packages
>
```

☑	methods	Formal Methods and Classes	3.0.1
□	mgcv	Mixed GAM Computation Vehicle with GCV/AIC/REML smoothness estimation	1.7-22
□	multcomp	Simultaneous Inference in General Parametric Models	1.2-19
□	munsell	Munsell colour system	0.4.2
□	mvtnorm	Multivariate Normal and t Distributions	0.9-9995
□	nlme	Linear and Nonlinear Mixed Effects Models	3.1-109
□	nnet	Feed-forward Neural Networks and Multinomial Log-Linear Models	7.3-6
□	parallel	Support for Parallel computation in R	3.0.1
□	plyr	Tools for splitting, applying and combining data	1.8
□	proto	Prototype object-based programming	0.3-10
□	psych	Procedures for Psychological, Psychometric, and Personality Research	1.3.10.12
□	Rcmdr	R Commander	1.9-6
□	RColorBrewer	ColorBrewer palettes	1.0-5
□	RCurl	General network (HTTP/FTP/...) client interface for R	1.95-4.1
□	relimp	Relative Contribution of Effects in a Regression Model	1.0-3

Getting started

Setting the working directory:

```
>getwd()
[1] "C:/Users/User1/Documents“
>setwd(“C:/Users/User1/Documents”)
```

Reading data into dataframe:

read.table(), read.csv(), read.xlsx()

Other database entries are possible.

RODBC package

```
>dat.csv <- read.csv("http://www.ats.ucla.edu/stat/data/hsb2.csv")
```

Tab separated values:

```
>dat.tab <- read.table("http://www.ats.ucla.edu/stat/data/hsb2.txt",
header=T, sep = "\t")
```

Get dimensions of dataframe:

```
>dim(dat.tab)
>nrow(dat.tab)
>ncol(dat.tab)
>edit(dat.tab)  --R editor opens.
```

Object types:

```
>class(dat.tab)
[1] "data.frame"
>class(dat.tab$math)
[1] “integer”
```

```
>attach(dat.tab)
>detach(dat.tab)
```

To see the header in the data file

```
>names(dat.tab)
```

```
[1] "id“  "female"  "race“  "ses“  "schtyp“  "prog"  "read"  "write“
 [9] "math“  "science“  "socst"
```

Exporting data:

```
>write.csv(dat.csv, file = “C:/Users/Vinodh/Desktop/filename.csv")
>write.table(dat.tab, file = "C:/Users/Vinodh/Desktop/filename.txt, sep="\t")
```

Programming Tools

If-else statement:

```
>w=3
>if( w < 5 )
        {d=2}   else
             {d=10 }
>d

[1] 2
```

For loop:

```
>h = seq(from = 1, to = 8)
>s = c()

>for(i in 2:10)
     {
          s[i] = h[i] * 10
     }
>s
[1] NA 20 30 40 50 60 70 80 NA NA
```

Creating functions

Pre-programmed R functions

```
>fun1 = function(arg1, arg2 )
  {
  w = arg1 ^ 2
  return(arg2 + w)
  }

>fun1(arg1 = 3, arg2 = 5)

>[1] 14
```

Statistics

Statistic: a quantity calculated from a sample of data

Average Age of students

Average Math grade

Standard deviation of M

Statistics

Population: the entire collection of cases to which we want to generalize

Sample: a subset of the population

We draw a random sample from the population, and compute appropriate *statistics* from the sample, that give estimates of the corresponding population parameters of interest.

Statistics

- Parameter: a numerical measure that describes a characteristic of a population
- Statistic: a numerical measure that describes a characteristic of a sample

Statistics

- Descriptive statistics: procedures used to summarize, organize, and simplify data.
- Inferential statistics: procedures that allow for generalizations about population parameters based on sample statistics

Types of Variables

- Nominal
- Ordinal
- Interval
- Ratio

Central Tendency

Measure of central tendency: A measure that describes the middle or center point of a distribution

– A good measure of central tendency is representative of the distribution

- Mean: the average, $M = (\Sigma X) / N$
- Median: the middle score (the score below which 50% of the distribution falls)
- Mode: the score that occurs most often

Variability

- A measure that describes the range and diversity of scores in a distribution
  - – Standard deviation (SD): the average deviation from the mean in a distribution
  - – Variance= SD^2

- Point estimate
- Interval estimate

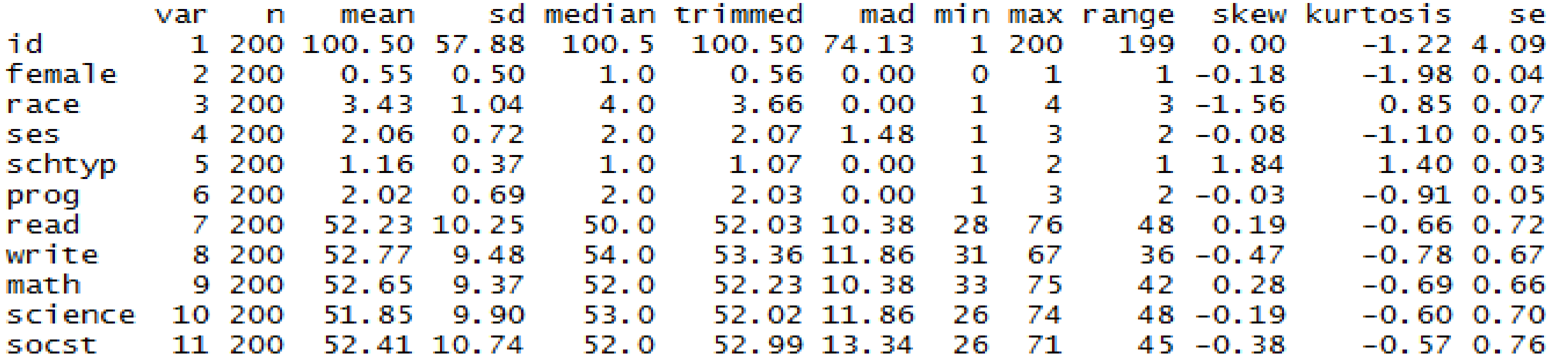
Summary statistics

Basic functions:

```
>install.packages("psych")
> library(psych)
>str(dat.tab)
>mean(dat.tab$science)
>sd(dat.tab$math)
```

Summary statistics

> describe(dat.tab)



```
        var   n   mean    sd median trimmed   mad min max range  skew kurtosis   se
id        1 200 100.50 57.88  100.5  100.50 74.13   1 200   199  0.00    -1.22 4.09
female    2 200   0.55  0.50    1.0    0.56  0.00   0   1     1 -0.18    -1.98 0.04
race      3 200   3.43  1.04    4.0    3.66  0.00   1   4     3 -1.56     0.85 0.07
ses       4 200   2.06  0.72    2.0    2.07  1.48   1   3     2 -0.08    -1.10 0.05
schtyp    5 200   1.16  0.37    1.0    1.07  0.00   1   2     1  1.84     1.40 0.03
prog      6 200   2.02  0.69    2.0    2.03  0.00   1   3     2 -0.03    -0.91 0.05
read      7 200  52.23 10.25   50.0   52.03 10.38  28  76    48  0.19    -0.66 0.72
write     8 200  52.77  9.48   54.0   53.36 11.86  31  67    36 -0.47    -0.78 0.67
math      9 200  52.65  9.37   52.0   52.23 10.38  33  75    42  0.28    -0.69 0.66
science  10 200  51.85  9.90   53.0   52.02 11.86  26  74    48 -0.19    -0.60 0.70
socst    11 200  52.41 10.74   52.0   52.99 13.34  26  71    45 -0.38    -0.57 0.76
```

>describeBy(dat.tab, dat.tab$female)

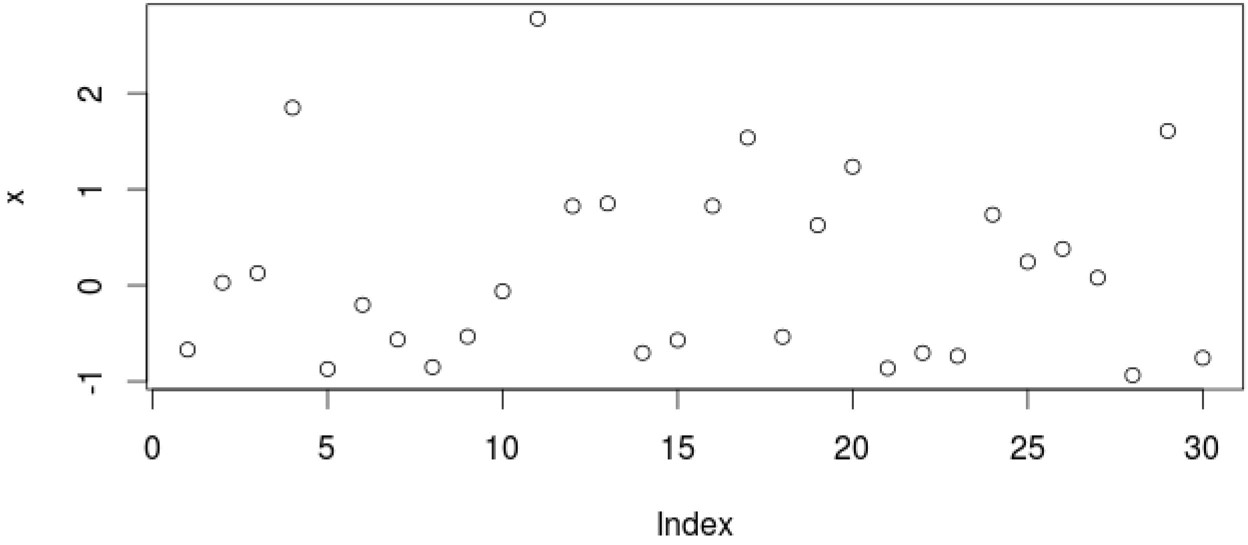
Subsetting data:

>female <-subset(dat.tab, dat.tab[,2]==0)
>male<-subset(dat.tab, dat.tab[,2]==1)

Plots

>help(rnorm) – it is an R function that creates random samples from a normal distribution.

```
>rnorm(5)
[1] 0.6867922  0.3659750  0.2918908  -2.5726535  1.0128191

>x <-rnorm(30)
>x
>plot(x)
```

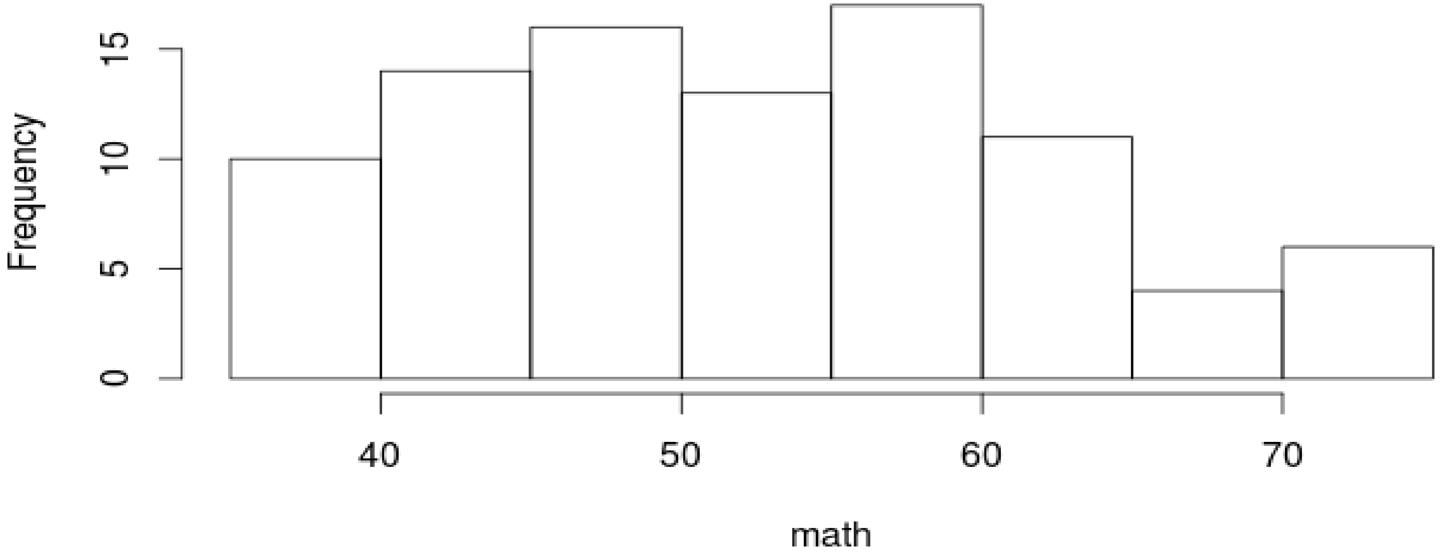


Histogram: (require a package called “psych”)

```
>par(mfrow = c(2,3))
>hist(female[,9],xlab="math",main="")
```

Or

```
>hist(female$math,xlab="math",main="")
```



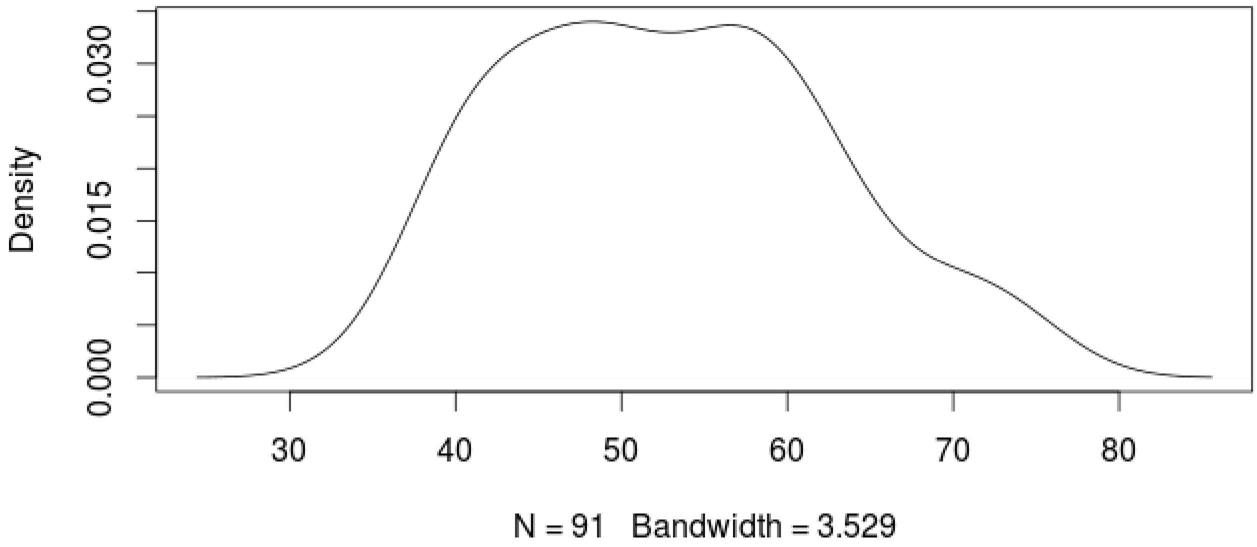
Density plots

Require “sm” package.

```
>install.packages("sm")
>library(sm)
>par(mfrow=c(1,2))
```

Density plot:

```
>plot(density(female[,9],xlab="math",main=""))
```



Correlation Analysis

Correlation refers to any statistical relationship between two random variables or two sets of data.

Packages required : psych, glus, rgl.

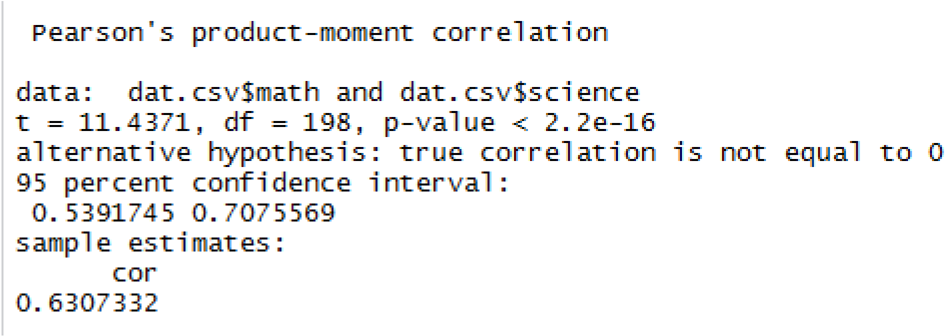
```
>cor(dat.tab[3:11])    -- Finding correlation among all variables.
```

```
>round(cor(dat.tab[3:11]) ,2)
```

```
  race  ses schtyp  prog  read write  math science socst
race     1.00 0.20   0.11 -0.05  0.24  0.22  0.20    0.32  0.19
ses      0.20 1.00   0.14  0.02  0.29  0.21  0.27    0.28  0.33
schtyp   0.11 0.14   1.00 -0.10  0.09  0.13  0.10    0.06  0.10
prog    -0.05 0.02  -0.10  1.00 -0.13 -0.18 -0.15   -0.19 -0.20
read     0.24 0.29   0.09 -0.13  1.00  0.60  0.66    0.63  0.62
write    0.22 0.21   0.13 -0.18  0.60  1.00  0.62    0.57  0.60
math     0.20 0.27   0.10 -0.15  0.66  0.62  1.00    0.63  0.54
science  0.32 0.28   0.06 -0.19  0.63  0.57  0.63    1.00  0.47
socst    0.19 0.33   0.10 -0.20  0.62  0.60  0.54    0.47  1.00
```

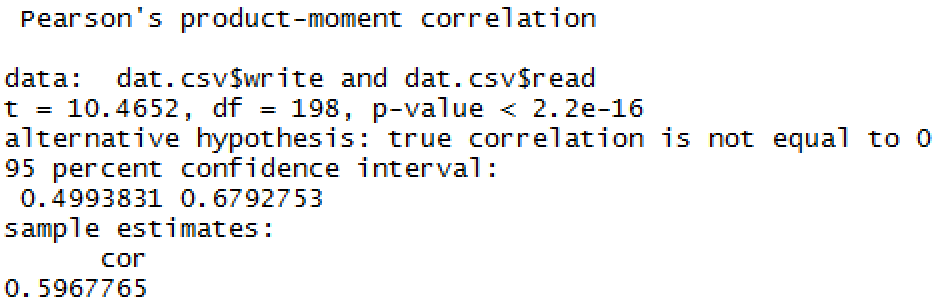
Correlation Analysis

>cor.test(dat.tab$math,dat.tab$science)



```
	Pearson's product-moment correlation

data:  dat.csv$math and dat.csv$science
t = 11.4371, df = 198, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.5391745 0.7075569
sample estimates:
      cor 
0.6307332 
```

>cor.test(dat.tab$write,dat.tab$read)

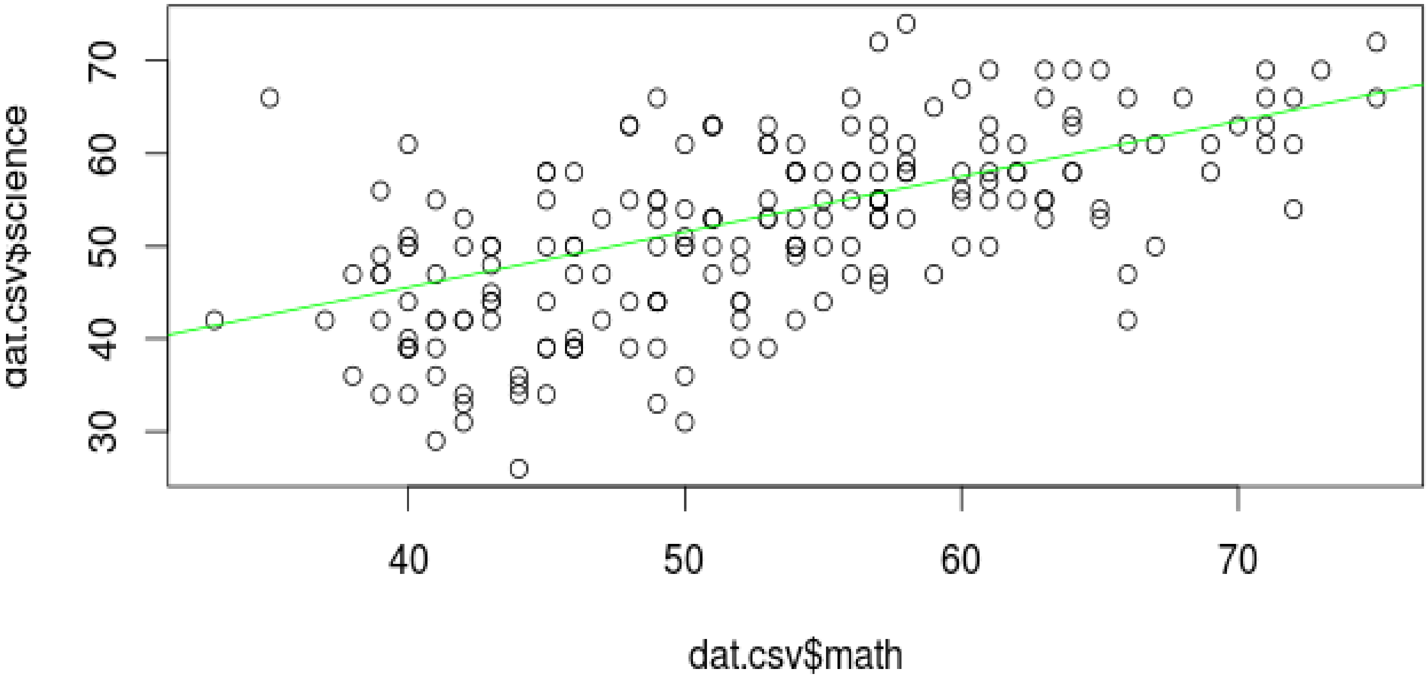


```
	Pearson's product-moment correlation

data:  dat.csv$write and dat.csv$read
t = 10.4652, df = 198, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.4993831 0.6792753
sample estimates:
      cor 
0.5967765 
```

Correlation Analysis

Standard Scatter plot: (requires ggplot2 package)

```
>plot(dat.tab$math,dat.tab$science)
>abline(lm(dat.tab$math~dat.tab$science),col="green")
```



Regression

- Important concepts & topics
- – Simple regression vs. multiple regression
- – Regression equation
- – Regression model

Regression

- Regression: a statistical analysis used to predict scores on an outcome variable, based on scores on one or multiple predictor variables
- – Simple regression: one predictor variable
- – Multiple regression: multiple predictors

Regression

- $Y = m + bX + e$
  - Y is a linear function of X
  - m = intercept
  - b = slope
  - e = error (residual)

- $Y = B0 + B1X1 + e$
  - Y is a linear function of X1
  - B0 = intercept = regression constant
  - B1 = slope = regression coefficient
  - e = error (residual)

Regression

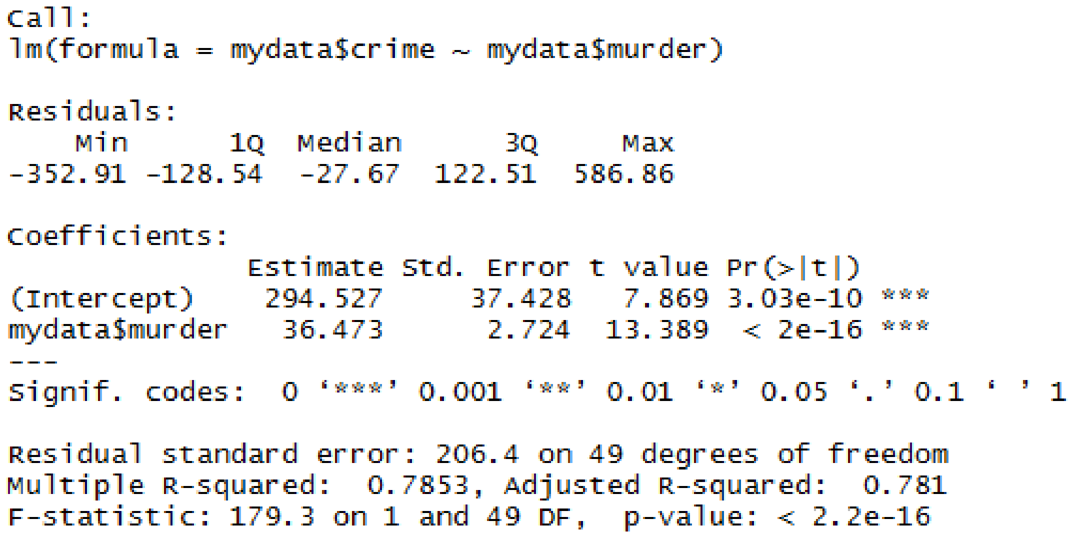
>mydata<-read.table("http://www.ats.ucla.edu/stat/data/crime.csv", header=TRUE,sep=",")
>str(mydata)

- Correlation plotting

>pairs(mydata[3:9],main="All combinations")

- Model Building

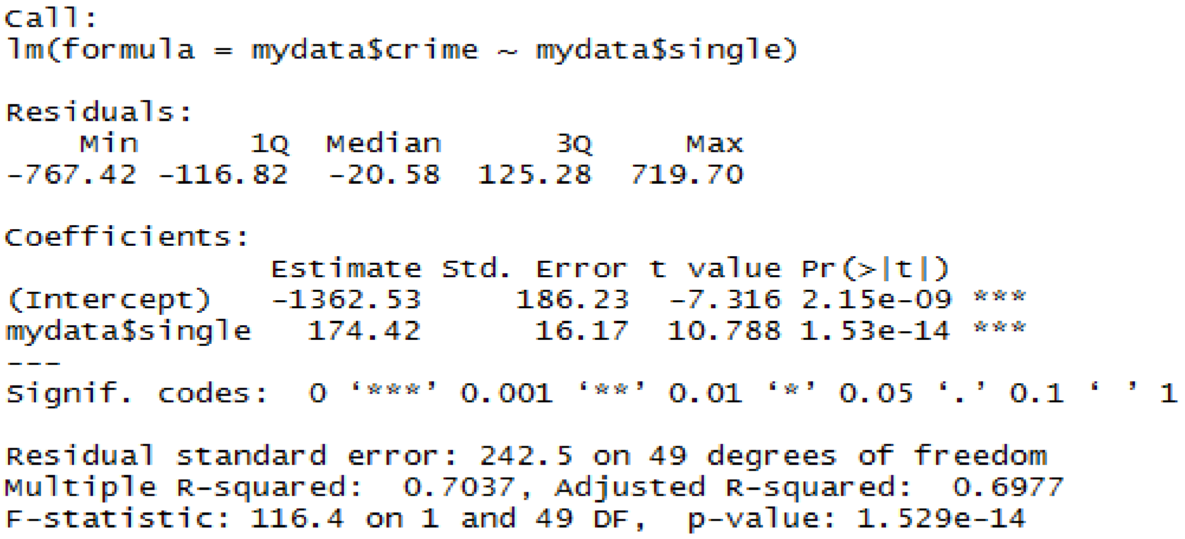
>model1<-lm(mydata$crime~mydata$murder)
> summary(model1)



```
Call:
lm(formula = mydata$crime ~ mydata$murder)

Residuals:
    Min      1Q  Median      3Q     Max 
-352.91 -128.54  -27.67  122.51  586.86 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)    294.527     37.428   7.869 3.03e-10 ***
mydata$murder   36.473      2.724  13.389  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 206.4 on 49 degrees of freedom
Multiple R-squared:  0.7853,	Adjusted R-squared:  0.781 
F-statistic: 179.3 on 1 and 49 DF,  p-value: < 2.2e-16
```

Regression

>model2<-lm(mydata$crime~mydata$single)
>summary(model2)

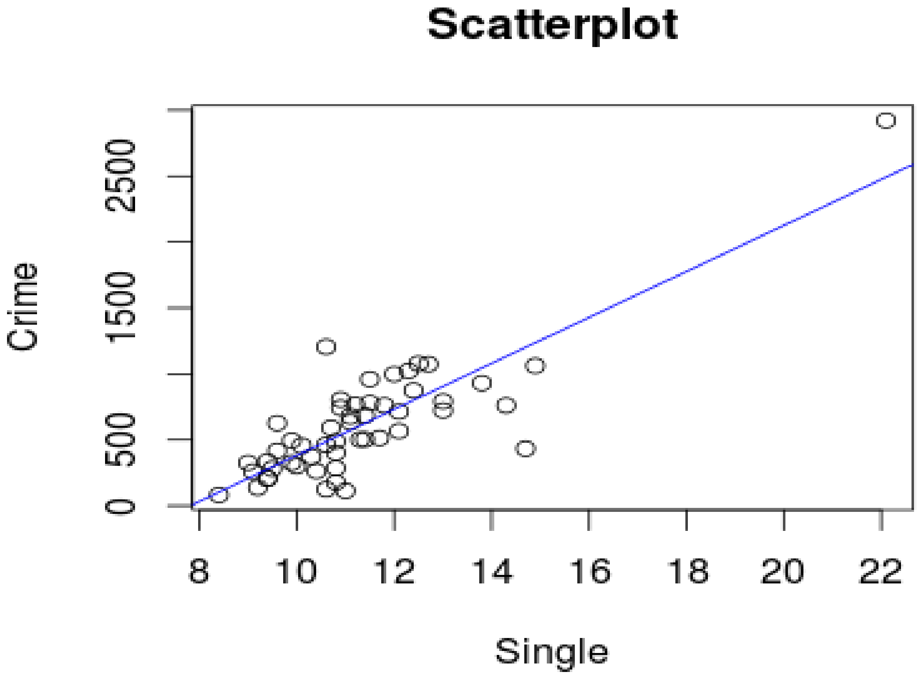


```
Call:
lm(formula = mydata$crime ~ mydata$single)

Residuals:
    Min      1Q  Median      3Q     Max 
-767.42 -116.82  -20.58  125.28  719.70 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)   -1362.53     186.23  -7.316 2.15e-09 ***
mydata$single   174.42      16.17  10.788 1.53e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 242.5 on 49 degrees of freedom
Multiple R-squared:  0.7037, Adjusted R-squared:  0.6977 
F-statistic: 116.4 on 1 and 49 DF,  p-value: 1.529e-14
```

R = multiple correlation coefficient
R2
– The percentage of variance in Y explained by the mode

Regression

```
>plot(mydata$crime~mydata$single, main = "Scatterplot", ylab =
"Crime", xlab = "Single")

>abline(lm(mydata
$crime~mydata$single), col="blue")

>model2.fit<-fitted(model2)
>model2.e<-resid(model2)
```



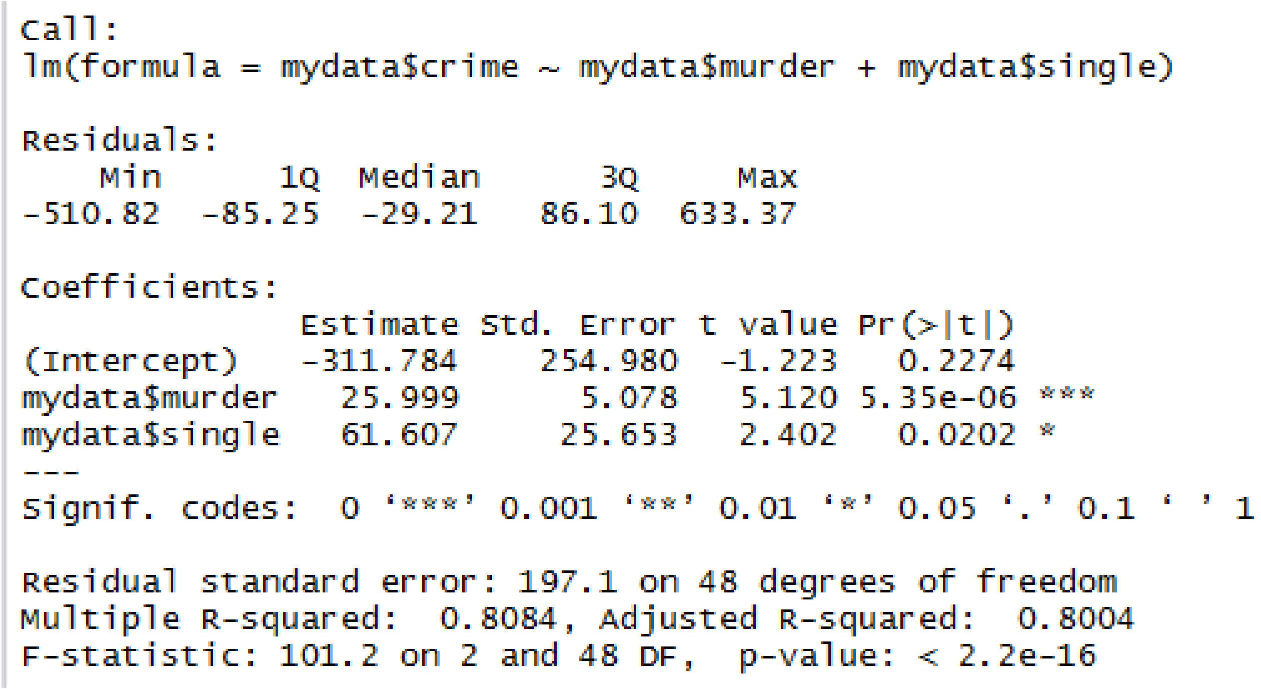
Regression

- Assumptions of linear regression
  - Normal distribution for Y
  - Linear relationship between X and Y
  - Homoscedasticity

  - Reliability of X and Y
  - Validity of X and Y
  - Random and representative sampling

Multiple Regression

```
>model3<-lm(mydata$crime~mydata$murder+mydata$single)
>summary(model3)
```



```
Call:
lm(formula = mydata$crime ~ mydata$murder + mydata$single)

Residuals:
    Min      1Q  Median      3Q     Max 
-510.82  -85.25  -29.21   86.10  633.37 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)   -311.784    254.980  -1.223   0.2274    
mydata$murder   25.999      5.078   5.120 5.35e-06 ***
mydata$single   61.607     25.653   2.402   0.0202 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 197.1 on 48 degrees of freedom
Multiple R-squared:  0.8084, Adjusted R-squared:  0.8004 
F-statistic: 101.2 on 2 and 48 DF,  p-value: < 2.2e-16
```

Multiple Regression

- There are three methods available for including variables in the regression equation:
  - the simultaneous method in which all independents are included at the same time
  - The hierarchical method in which control variables are entered in the analysis before the predictors whose effects we are primarily concerned with.
  - The stepwise method in which variables are selected in the order in which they maximize the statistically significant contribution to the model.

Logistic regression

- Logistic regression is used to analyze relationships between a dichotomous dependent variable and metric or dichotomous independent variables.

- Logistic regression combines the independent variables to estimate the probability that a particular event will occur

sale	custId	car	age	city	newCar
	c1	taurus	27	sf	yes
	c2	van	35	la	yes
	c3	van	40	sf	yes
	c4	taurus	22	sf	yes
	c5	merc	50	la	no
	c6	taurus	25	la	no

Logistic regression

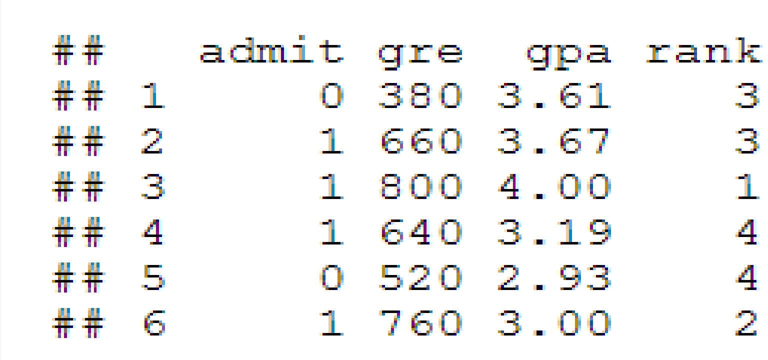
- Probability value between 0.0 and 1.0.
- Cut point (the default is 0.50), for membership
- Case based probability
- Logistic regression analysis requires that the dependent variable be dichotomous.
- Logistic regression analysis requires that the independent variables be metric or dichotomous.
- The minimum number of cases per independent variable is 10, using a guideline provided by Hosmer and Lemmeshow

Logistic regression

```
>install.package(“aod”)
>library(aod)

>mydata <- read.csv("http://www.ats.ucla.edu/stat/data/binary.csv")
```

## view the first few rows of the data

```
>head(mydata)
```

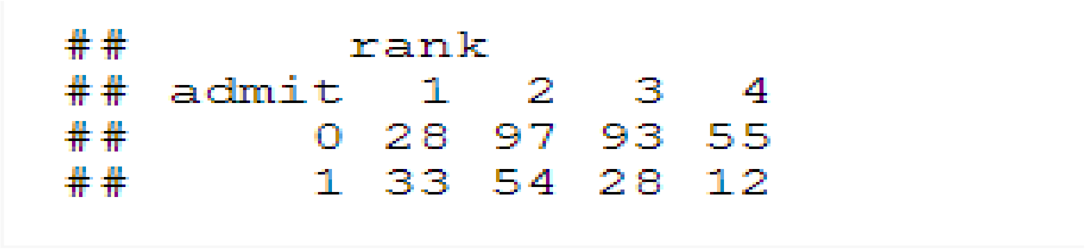


```
##   admit gre  gpa rank
## 1     0 380 3.61    3
## 2     1 660 3.67    3
## 3     1 800 4.00    1
## 4     1 640 3.19    4
## 5     0 520 2.93    4
## 6     1 760 3.00    2
```

```
>summary(mydata)
```

Logistic regression

>xtabs(~admit + rank, data = mydata)



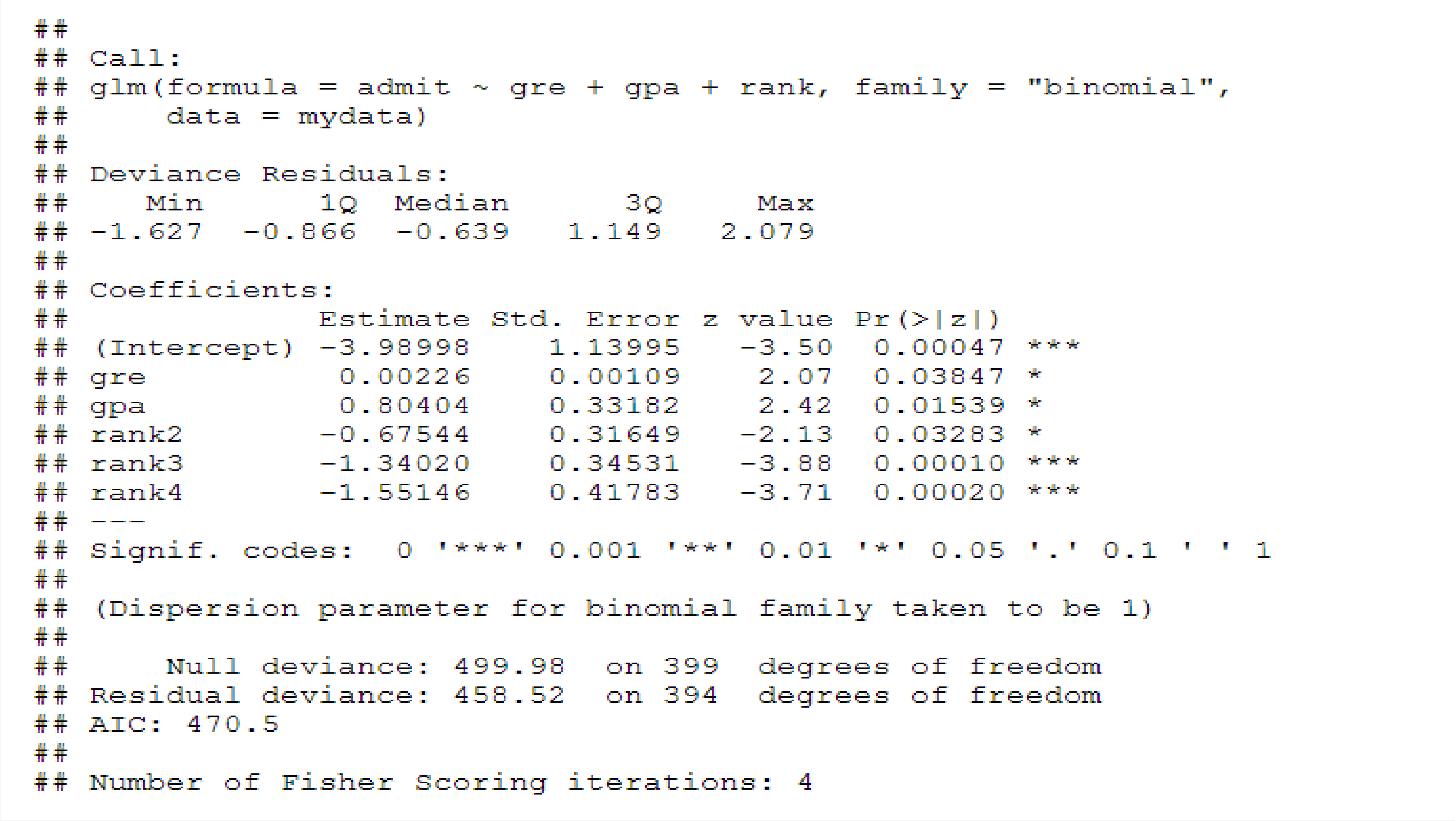
```
##        rank
## admit   1  2  3  4
##      0 28 97 93 55
##      1 33 54 28 12
```

>mydata$rank <- factor(mydata$rank)

>mylogit <- glm(admit ~ gre + gpa + rank, data = mydata, family = "binomial")

Logistic regression

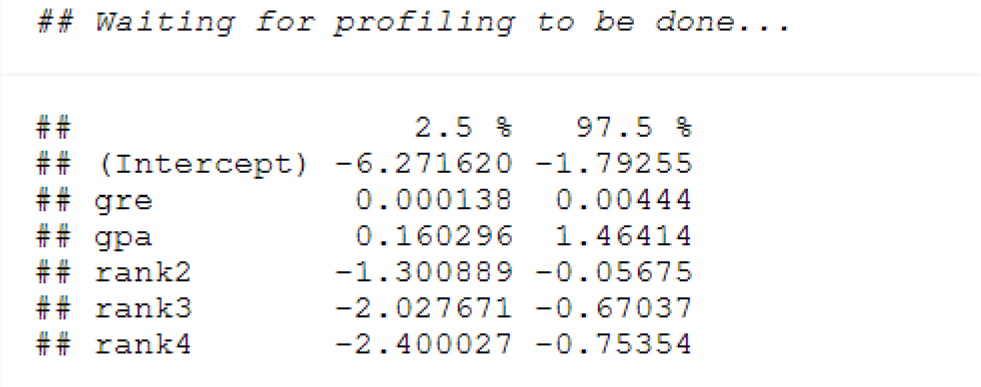
>summary(mylogit)



```
##
## Call:
## glm(formula = admit ~ gre + gpa + rank, family = "binomial",
##     data = mydata)
##
## Deviance Residuals:
##    Min       1Q   Median       3Q      Max
## -1.627  -0.866  -0.639   1.149   2.079
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.98998    1.13995   -3.50  0.00047 ***
## gre          0.00226    0.00109    2.07  0.03847 *
## gpa          0.80404    0.33182    2.42  0.01539 *
## rank2       -0.67544    0.31649   -2.13  0.03283 *
## rank3       -1.34020    0.34531   -3.88  0.00010 ***
## rank4       -1.55146    0.41783   -3.71  0.00020 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 499.98  on 399  degrees of freedom
## Residual deviance: 458.52  on 394  degrees of freedom
## AIC: 470.5
##
## Number of Fisher Scoring iterations: 4
```

Logistic regression

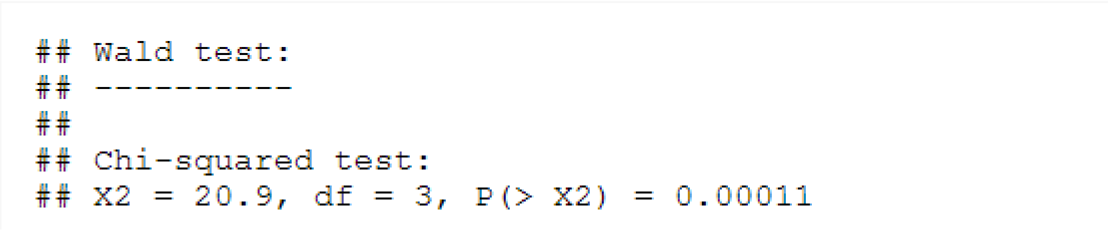
>confint(mylogit)



```
## Waiting for profiling to be done...
```

```
##                  2.5 %   97.5 %
## (Intercept) -6.271620 -1.79255
## gre          0.000138  0.00444
## gpa          0.160296  1.46414
## rank2       -1.300889 -0.05675
## rank3       -2.027671 -0.67037
## rank4       -2.400027 -0.75354
```

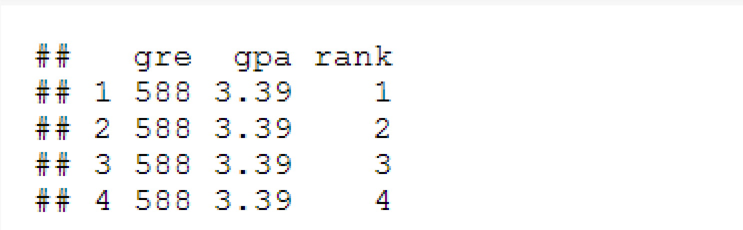
>wald.test(b = coef(mylogit), Sigma = vcov(mylogit), Terms = 4:6)



```
## Wald test:
## ----------
## 
## Chi-squared test:
## X2 = 20.9, df = 3, P(> X2) = 0.00011
```

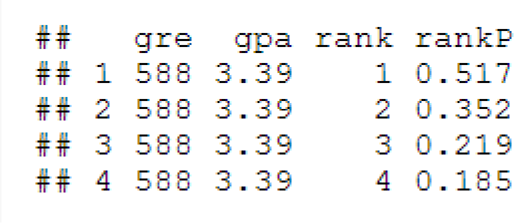
Logistic regression

>newdata1 <- with(mydata, data.frame(gre = mean(gre), gpa = mean(gpa), rank = factor(1:4)))
>newdata1



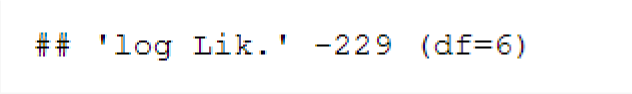
```
##   gre  gpa rank
## 1 588 3.39    1
## 2 588 3.39    2
## 3 588 3.39    3
## 4 588 3.39    4
```

>newdata1$rankP <- predict(mylogit, newdata = newdata1, type = "response")
>newdata1



```
##   gre  gpa rank rankP
## 1 588 3.39    1 0.517
## 2 588 3.39    2 0.352
## 3 588 3.39    3 0.219
## 4 588 3.39    4 0.185
```

>logLik(mylogit)



```
## 'log Lik.' -229 (df=6)
```

Goodness of Fit

- Often a model with intercept and predictors is compared to an intercept only model to test whether the predictors add over and above the intercept only. This is usually noted as $\chi 2=2[LL(B)-LL(0)]$

- Hosmer-and-lemeshow-goodness

Interpreting coefficients

- Each coefficient is evaluated using a Wald test (really just a Z-test)

$$W_j = \frac{B_j}{SE_{B_j}}$$

Class Imbalance

- Oversampling in one of the classes.

- ROSE package for oversampling

Model Building in R

- Attribute Listing
- Derived Attribute list
- Factor analysis
- Data Sampling
- Correlation
- Significance Test
- Choice of analysis
- Validation
- Contingency table
- Cross validation
- Compare models

Attribute Listing

- Manual assessment
- Domain Experience
- Data types of attributes
- Missing attribute calculation
- Naming the attribute list

Derived attribute list

- Some attributes are hidden
  - Mileage as a parameter for car performance
- Attributes derived from other attributes
  - Income level from car brand owned
- Naming the attributes

Factor Analysis

- Factor analysis is a statistical method used to describe variability among observed, correlated variables in terms of a potentially lower number of unobserved variables called factors.
- It is possible, for example, that variations in three or four observed variables mainly reflect the variations in fewer unobserved variables.
- Factor analysis searches for such joint variations in response to unobserved latent variables.

Factor Analysis

- Principal Component Analysis
  - library(psych)
  - fit <-principal(mydata, nfactors=5, rotate="varimax")
  - Fit
  - fit <- factanal(mydata, 3, rotation="varimax")
  - print(fit, digits=2, cutoff=.3, sort=TRUE)

Data Sampling

- Representative sample
- Random sample
- Stratified Sampling
- Minimum sample size
- Central limit theorem
- Training and Test Set

Correlation

- Correlation refers to any statistical relationship between two random variables or two sets of data.
- Formally, dependence refers to any situation in which random variables do not satisfy a mathematical condition of probabilistic independence.
- cor(mydata$var1,mydata$var2)

Significance test

- Test the significant effect of the predictor on response variable.
- Null Hypothesis Test for variables.
- Find p value

Choice of Analysis

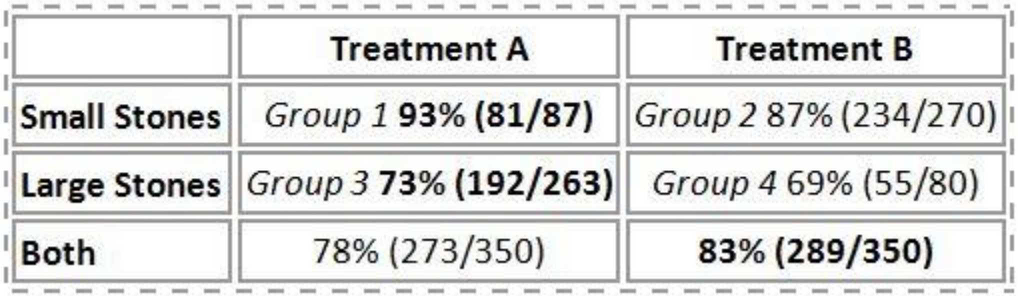
- Single or Mutiple Regression
- Classification
- Clustering
- Hybrid

Validation

- Run the model on test data.
- Find the predictive power of the model.

Contingency Table

- Truth table



	Treatment A	**Treatment B**
Small Stones	*Group 1* **93% (81/87)**	*Group 2* 87% (234/270)
Large Stones	*Group 3* **73% (192/263)**	*Group 4* 69% (55/80)
Both	78% (273/350)	**83% (289/350)**

Cross Validation

- Cross-validation for assessing how the results of a statistical analysis will generalize to an independent data set.
- One round of cross-validation involves partitioning a sample of data into complementary subsets

- k-fold cross-validation

Model Validation

- Prediction accuracy explains the stronger model.
- Variance explained in Rsquared is one sign
- Comparison of models with ANOVA.

What is Cluster Analysis?

Cluster: A collection of data objects
similar (or related) to one another within the same group
dissimilar (or unrelated) to the objects in other groups

Unsupervised learning: no predefined classes (i.e., *learning by observations* vs. learning by examples: supervised)
Typical applications
As a stand-alone tool to get insight into data distribution
As a preprocessing step for other algorithms

Applications of Cluster Analysis

Data reduction

Summarization: Preprocessing for regression, PCA, classification, and association analysis

Compression: Image processing: vector quantization

Prediction based on groups

Cluster & find characteristics/patterns for each group

Localizing search to one or a small number of clusters

Outlier detection: Outliers are often viewed as those “far away” from any cluster

Quality: What Is Good Clustering?

A good clustering method will produce high quality clusters

high intra-class similarity: cohesive within clusters

low inter-class similarity: distinctive between clusters

The quality of a clustering method depends on

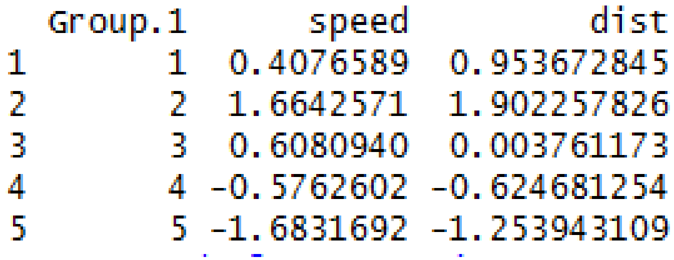
the similarity measure Its ability to discover some or all of the hidden patterns

Basic Steps to Develop a Clustering Task

Feature selection
Select info concerning the task of interest
Minimal information redundancy
Proximity measure
Similarity of two feature vectors
Clustering criterion
Expressed via a cost function or some rules
Clustering algorithms
Choice of algorithms
Validation of the results
Validation test (also, *clustering tendency* test)
Interpretation of the results
Integration with applications

Clustering

```
>library(datasets)
>data(cars)
>mydata<-cars
# K-Means Cluster Analysis
>fit <- kmeans(mydata, 5) # 5 cluster solution
# get cluster means
>aggregate(mydata,by=list(fit$cluster),FUN=mean)
```

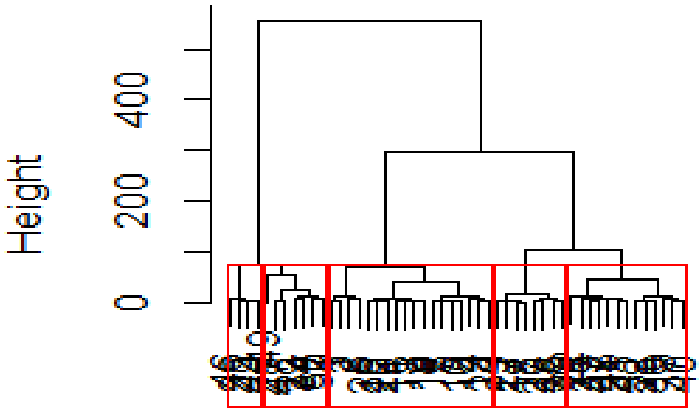


```
  Group.1      speed         dist
1       1  0.4076589  0.953672845
2       2  1.6642571  1.902257826
3       3  0.6080940  0.003761173
4       4 -0.5762602 -0.624681254
5       5 -1.6831692 -1.253943109
```

```
# append cluster assignment
>mydata <- data.frame(mydata, fit$cluster)
>mydata
```

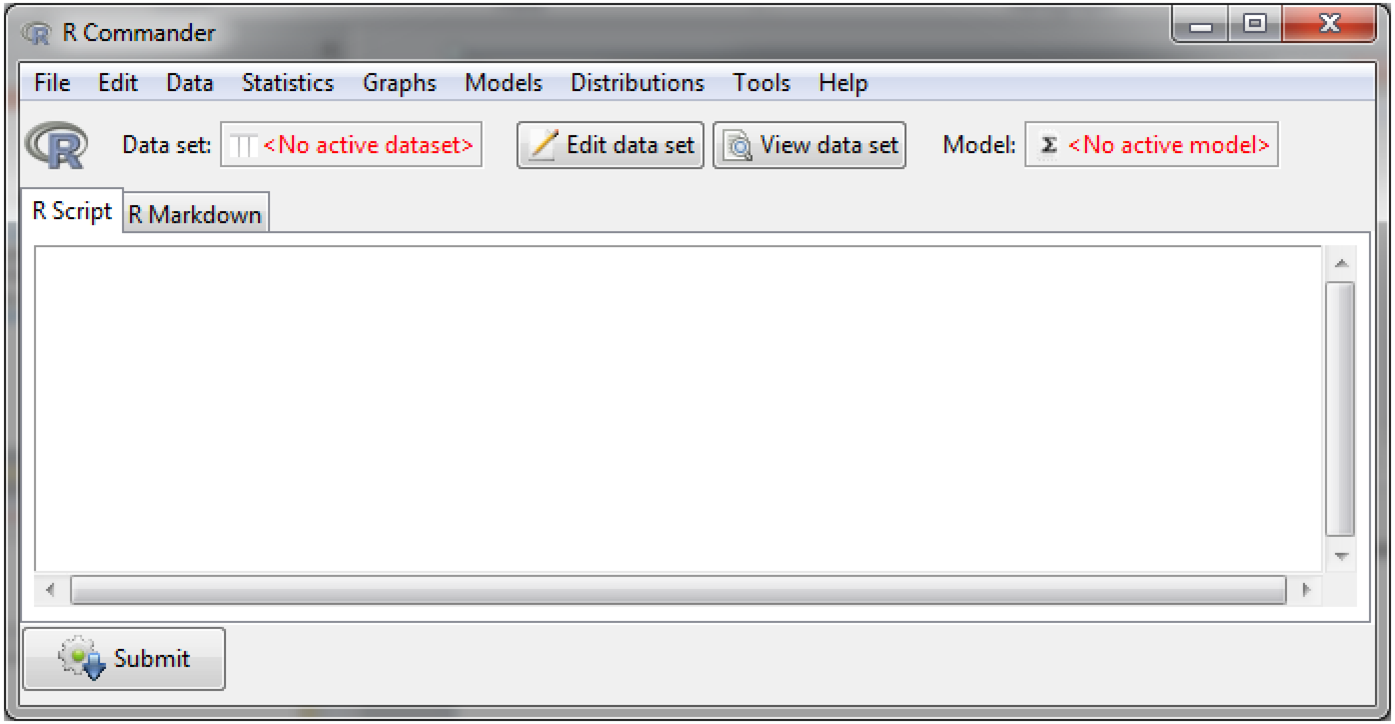
Clustering

```
>mydata<-cars
# Ward Hierarchical Clustering
d <- dist(mydata, method = "euclidean") # distance matrix
fit <- hclust(d, method="ward")
plot(fit) # display dendogram
groups <- cutree(fit, k=5) # cut tree into 5 clusters
# draw dendogram with red borders around the 5 clusters
rect.hclust(fit, k=5, border="red")
```



Rcmdr

- A platform-independent basic-statistics GUI (graphical user interface) for R, based on the tcltk package.



- Text Mining in twitteR package
- Support Vector Machine

Structural equation modeling

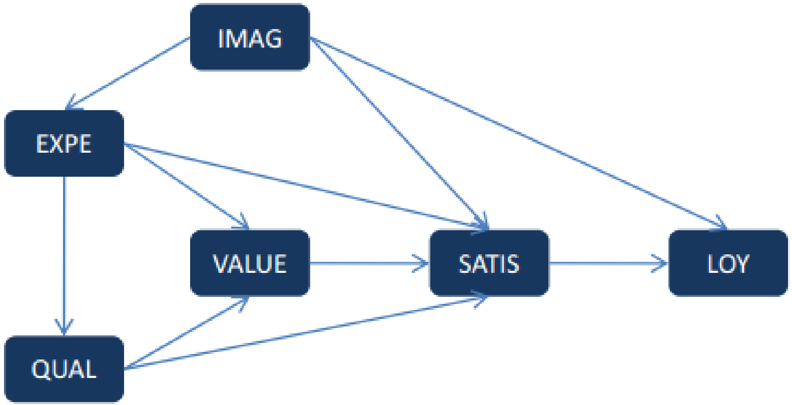
- Structural equation models (SEM) allow both confirmatory and exploratory modeling.

- They are suited to both theory testing and theory development.

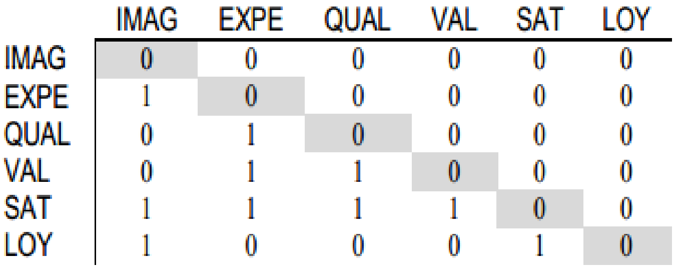
- Confirmatory modeling usually starts out with a hypothesis that gets represented in a causal model.

- The model is tested against the obtained measurement data to determine how well the model fits the data.

Structural equation modeling



Inner Design Matrix



	IMAG	EXPE	QUAL	VAL	SAT	LOY
IMAG	0	0	0	0	0	0
EXPE	1	0	0	0	0	0
QUAL	0	1	0	0	0	0
VAL	0	1	1	0	0	0
SAT	1	1	1	1	0	0
LOY	1	0	0	0	1	0

Structural equation modeling

```
>install.packages("plspm")
>library(plspm)
>data(satisfaction)
>satisfaction
>IMAG<-c(0,0,0,0,0,0)
>EXPE<-c(1,0,0,0,0,0)
>QUAL<-c(1,1,0,0,0,0)
>VAL<-c(0,1,1,0,0,0)
>SAT<-c(1,1,1,1,0,0)
>LOY<-c(1,0,0,0,1,0)
```

Structural equation modeling

```
>sat.mat<-rbind(IMAG,EXPE,QUAL,VAL,SAT,LOY)
>sat.sets<-list(1:5,6:10,11:15,16:19,20:23,24:27)
>sat.mod<-rep("A",6)
>res2<-plspm(satisfaction,sat.mat,sat.sets,sat.mod,scheme="factor",scaled=FALSE, boot.val=TRUE,plsr=FALSE)
>summary(res2)
```

Check C.alpha, DG.rho in BLOCKS UNIDIMENSIONALITY
paths value for Structural Model

Big Data in R

pbdR package enables high-level distributed data parallelism in R
it can easily utilize large HPC platforms with thousands of cores, making the R language scale

http://r-pbd.org/

Big Data in R

pbdDEMO
This package offers a comprehensive set of over 20 pbdR package demos, and a textbook-style vignette that can quickly help you take your programming from 1 to 10,000+ cores.
pbdMPI
This package provides an efficient interface to MPI
pbdPROF
This package provides access to MPI profiling

Big Data in R

I/O

pbdNCDF4

This package offers a friendly syntax to enable the management of NetCDF4

Computation

pbdDMAT, pbdBASE, and pbdSLAP

These packages offer high-level syntax for large scale, distributed matrix algebra and statistics operations.

Big Data in R

Application
pmclust
This package implements parallel model-based clustering, an unsupervised learning technique, for high dimensional and ultra large distributed data

Ggplot2 Package

- There are 3 options for producing graphics in R:

1)base graphs
2)lattice
3)ggplot2

- Ggplot2 package is the most popular package for creating customized and novel plots.
- Available from CRAN via *install.packages()*
- It is an implementation of the *Grammar of Graphics*, hence the name gg-plot.
- Each component is added to the plot as a layer and hence easy to customize.

Components of a plot

- Plots convey information through various aspects of their aesthetic
- Some aesthetics that plots use are:
- x position
- y position
- size of elements
- shape of elements
- color of elements

The elements in a plot are geometric shapes, like

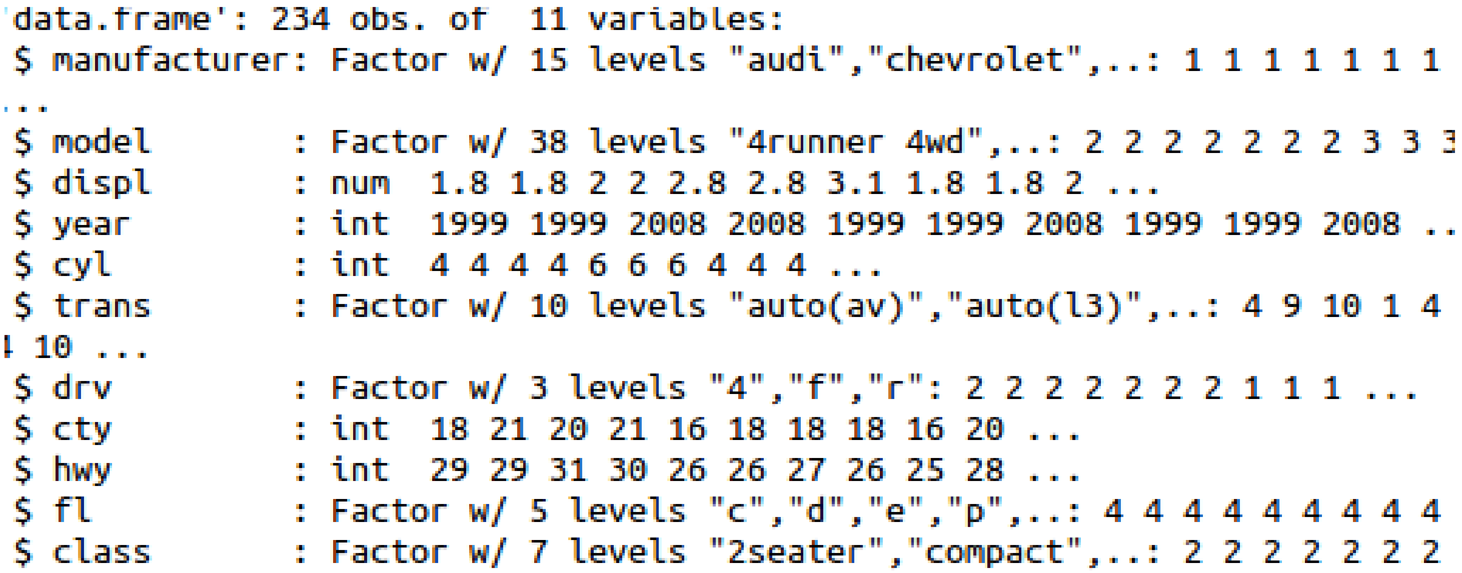
- points
- lines
- line segments
- bars
- text

The basics: qplot()

- The quick plotting function in the ggplot2 package.
- Most basic function.
- Plots contain 1) aesthetics :-size,shape,color
  2)geoms :- points,lines,bars

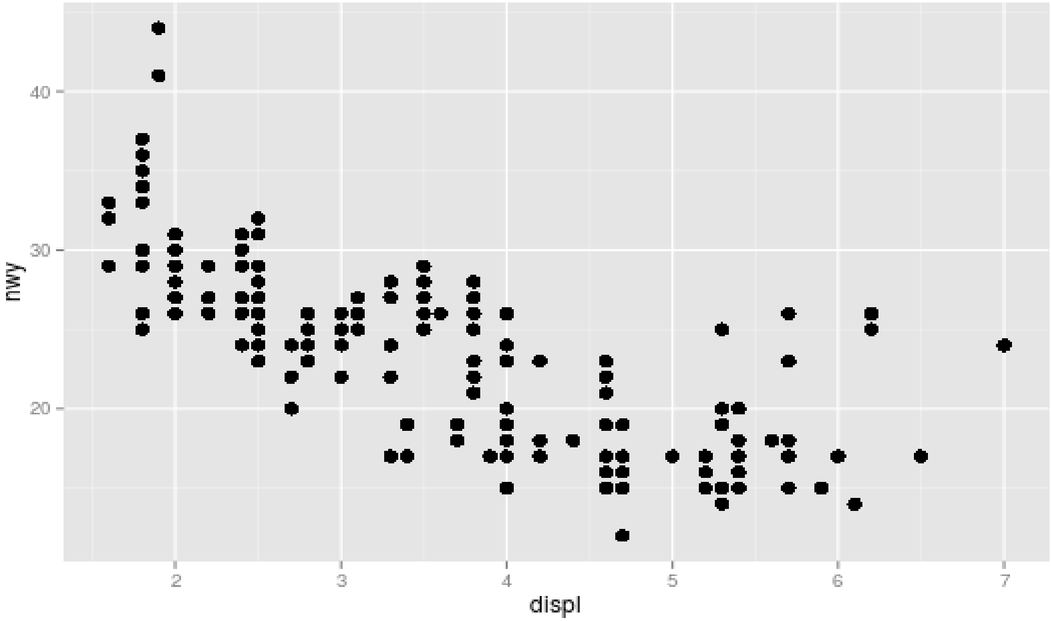
- Download and install the package:

```
>install.packages(“ggplot2”)
>library(ggplot2)
>str(mpg)
```



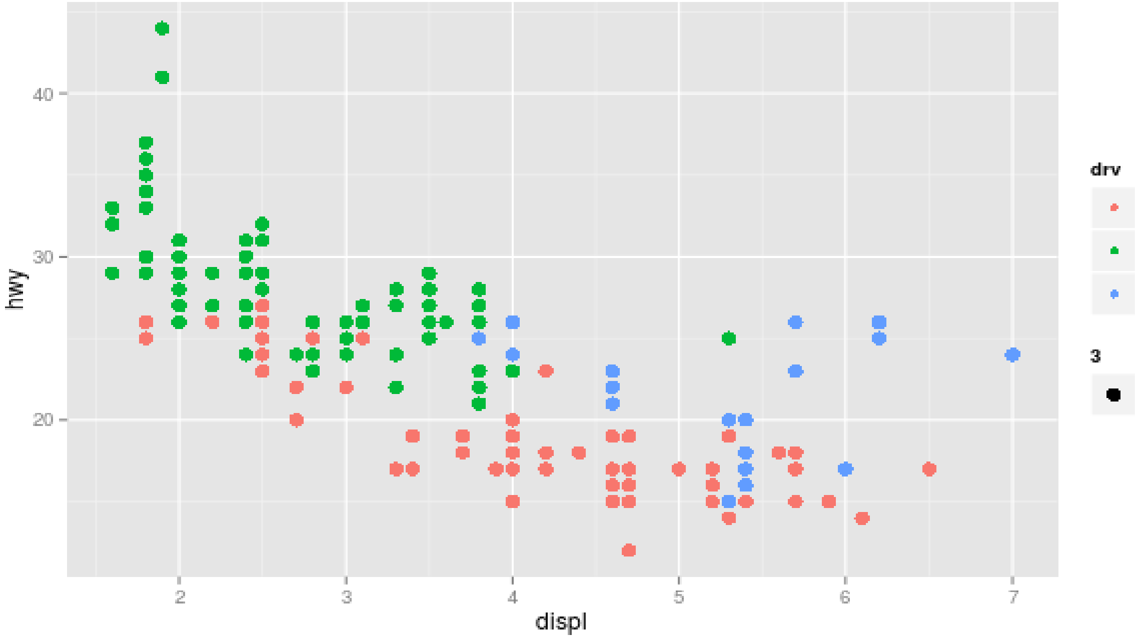
```
'data.frame':   234 obs. of  11 variables:
 $ manufacturer: Factor w/ 15 levels "audi","chevrolet",..: 1 1 1 1 1 1 1
...
 $ model       : Factor w/ 38 levels "4runner 4wd",..: 2 2 2 2 2 2 2 3 3 3
 $ displ       : num  1.8 1.8 2 2 2.8 2.8 3.1 1.8 1.8 2 ...
 $ year        : int  1999 1999 2008 2008 1999 1999 2008 1999 1999 2008 ..
 $ cyl         : int  4 4 4 4 6 6 6 4 4 4 ...
 $ trans       : Factor w/ 10 levels "auto(av)","auto(l3)",..: 4 9 10 1 4
4 10 ...
 $ drv         : Factor w/ 3 levels "4","f","r": 2 2 2 2 2 2 2 1 1 1 ...
 $ cty         : int  18 21 20 21 16 18 18 18 16 20 ...
 $ hwy         : int  29 29 31 30 26 26 27 26 25 28 ...
 $ fl          : Factor w/ 5 levels "c","d","e","p",..: 4 4 4 4 4 4 4 4 4
 $ class       : Factor w/ 7 levels "2seater","compact",..: 2 2 2 2 2 2 2
```

```
>qplot(displ,hwy,data=mpg)
```



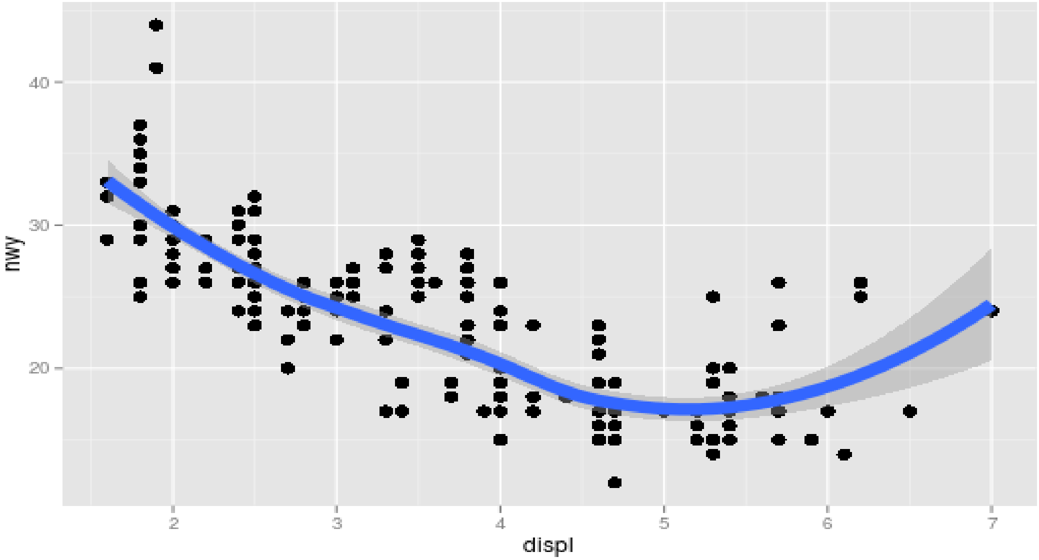
```
>qplot(displ,hwy,data=mpg,color=drv)
```

##Highlighting subgroups



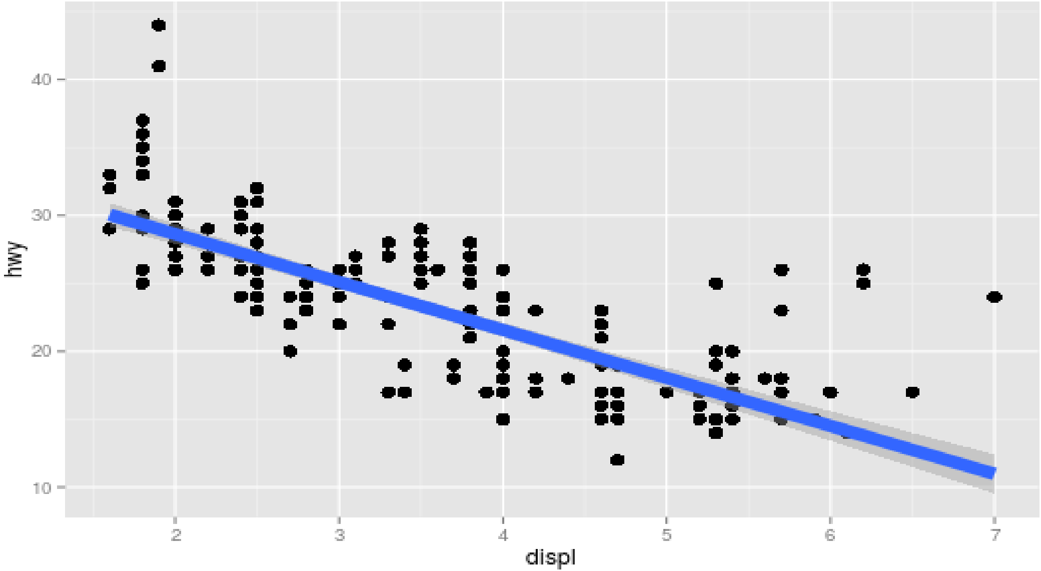
```
>qplot(displ,hwy,data=mpg, geom=c("point","smooth"))
```

## Add trend to the data



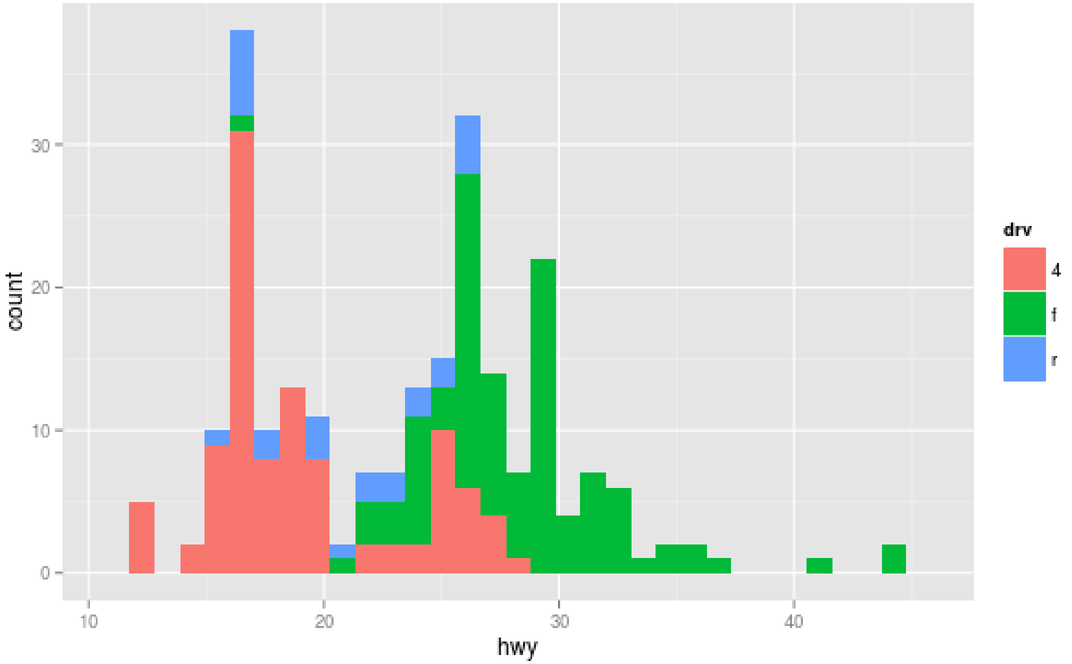
```
>qplot(displ, hwy,data=mpg, geom=c("point","smooth"), method=”lm”)
```

##Linear relationship
between the variables

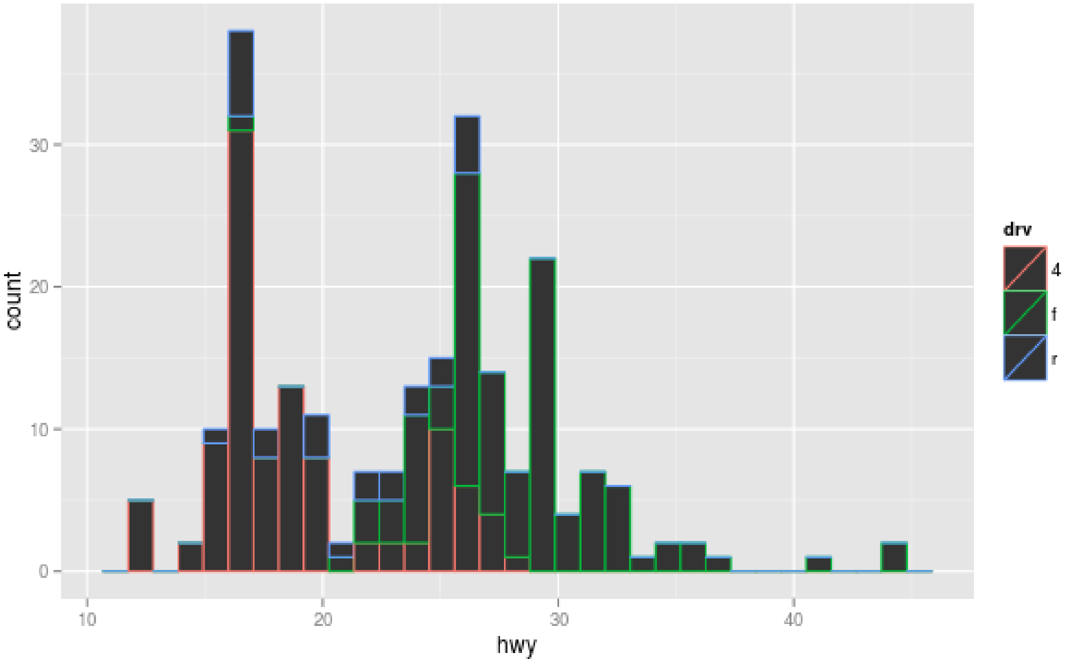


HISTOGRAMS

```
>qplot(hwy,data=mpg,fill=drv)
```



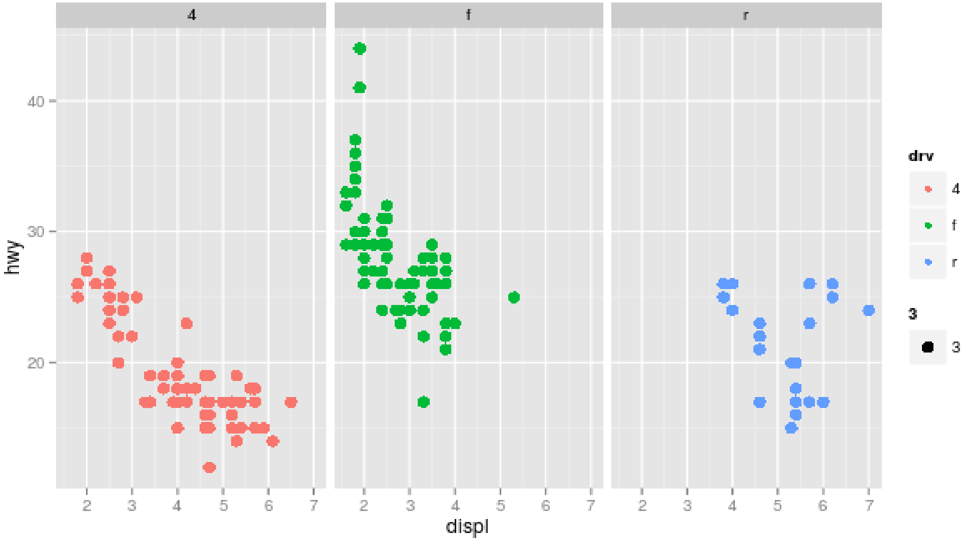
```
>qplot(hwy,data=mpg,color=drv)
```



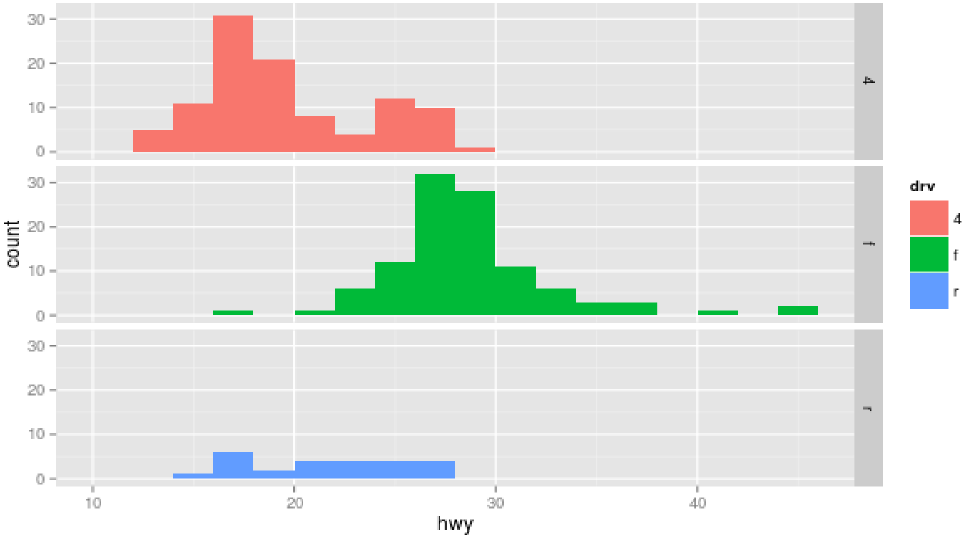
FACETS

>qplot(displ,hwy,data=mpg, *facets=.~drv* ,color=drv)

##Graphs in panels

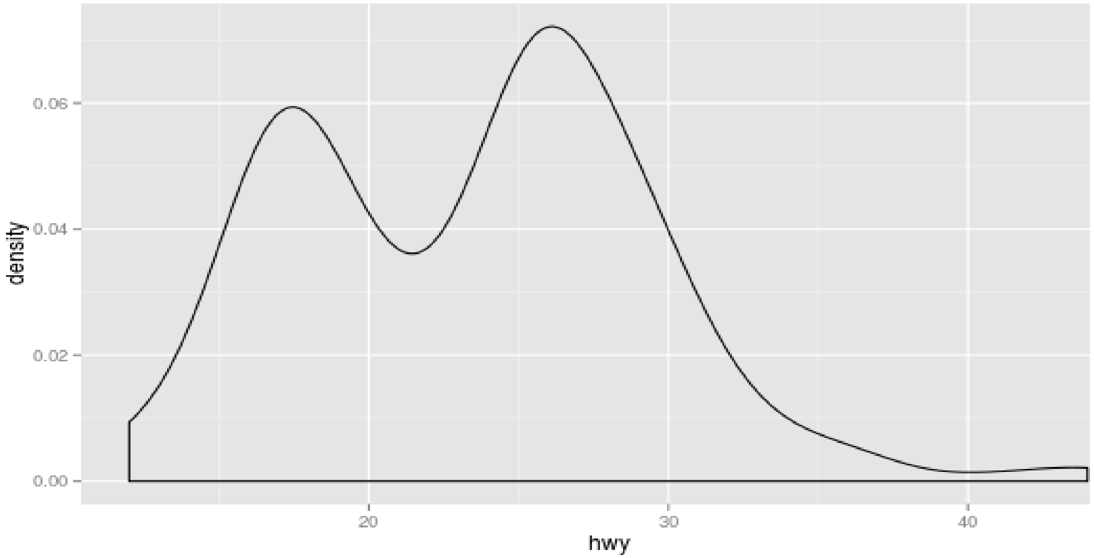


>qplot(hwy,data=mpg, *facets=drv~.* ,binwidth=2,fill=drv)

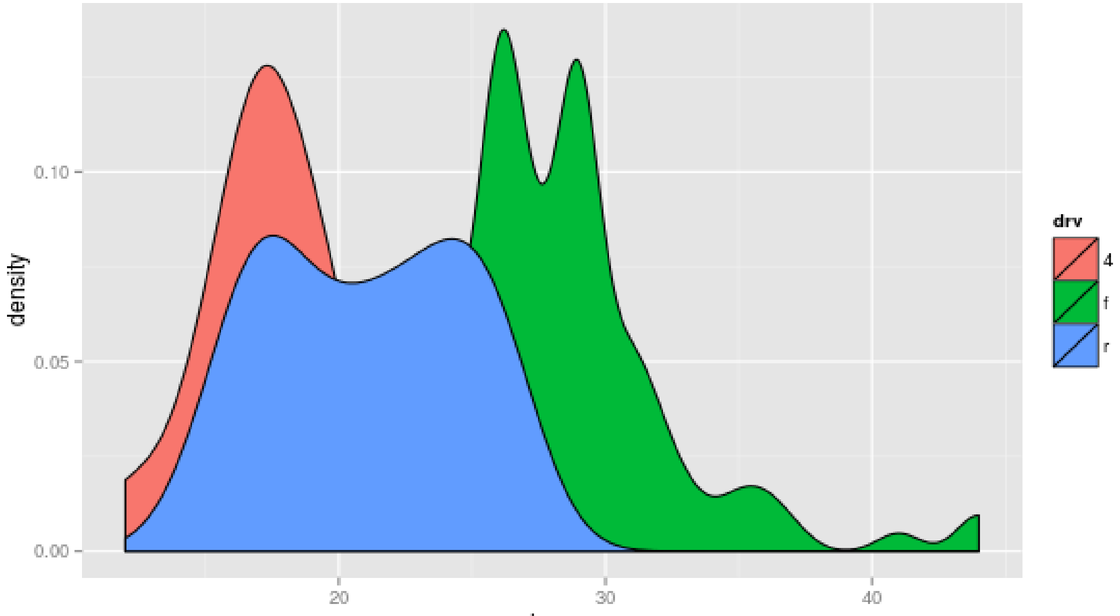


Density Smooth

```
>qplot(hwy,data=mpg,geom=”density”)
```



```
>qplot(hwy,data=mpg,geom=”density”,fill=drv)
```



ggplot()

- Ggplot() is the core function and allows more customisation
- Very flexible in doing things qplot() cant do.
- Start with the ggplot function call and then add things one by one,
- layer by layer.
- ggplot() takes two primary arguments:

  Data -The data frame containing the data to be plotted

  aes() - The aesthetic mappings to pass on to the plot elements

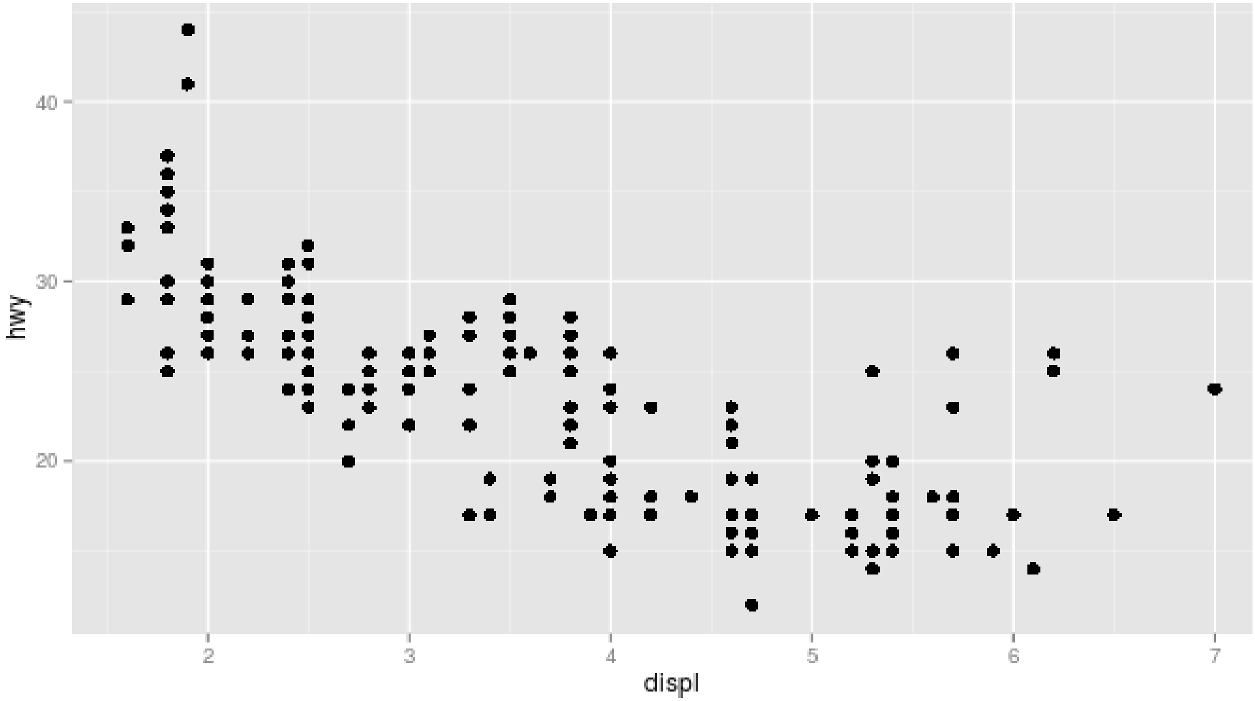
Basic components of ggplot

- *A data frame*
- *Aesthetic mappings* – how data are mapped to color and size
- *Geoms* – geometric objects like points,lines and shapes
- *Facets* – for dividing plots into panels
- *Stats* – for smoothing, a trend line

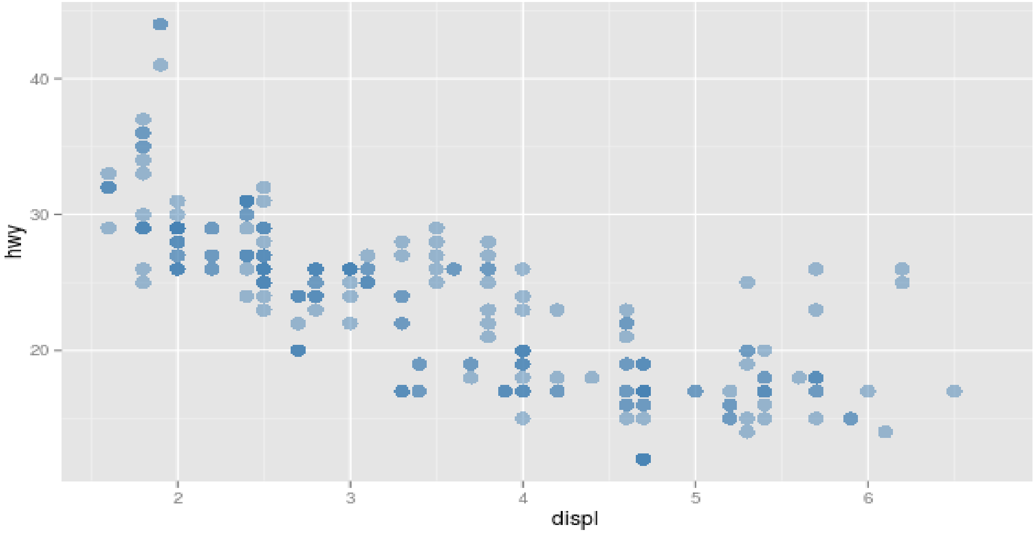
ggplot

- Ggplot() takes in a 1) data frame
  2)aes()
- Initial Function call:

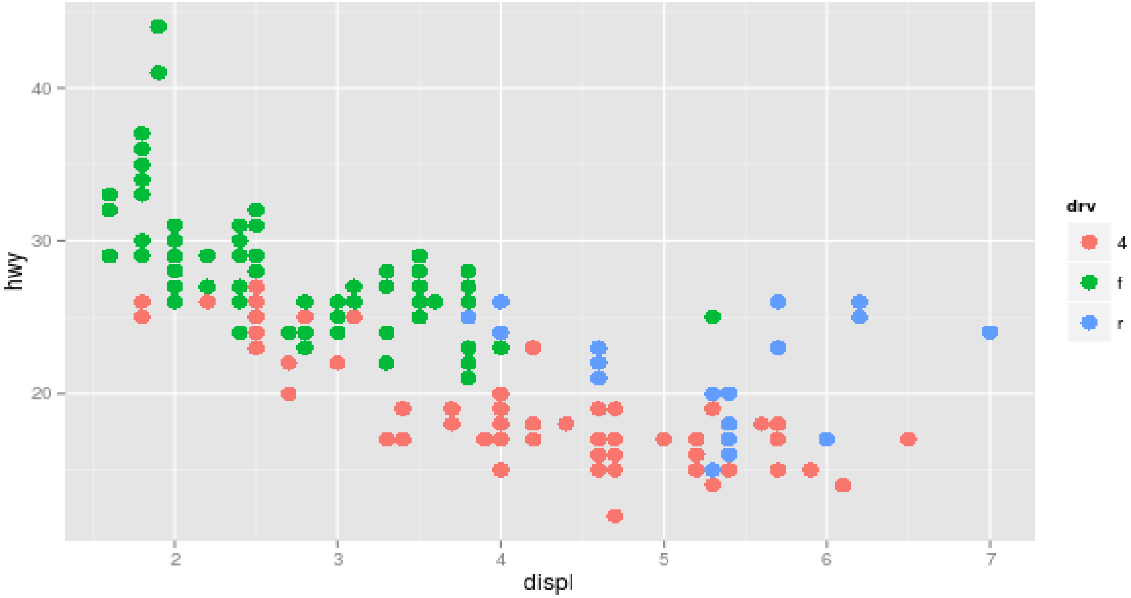
```
> p<- ggplot(mpg, aes(displ,hwy))
> p
Error: No layers in plot                ##Doesnt have enough information

> p +  geom_point()
```



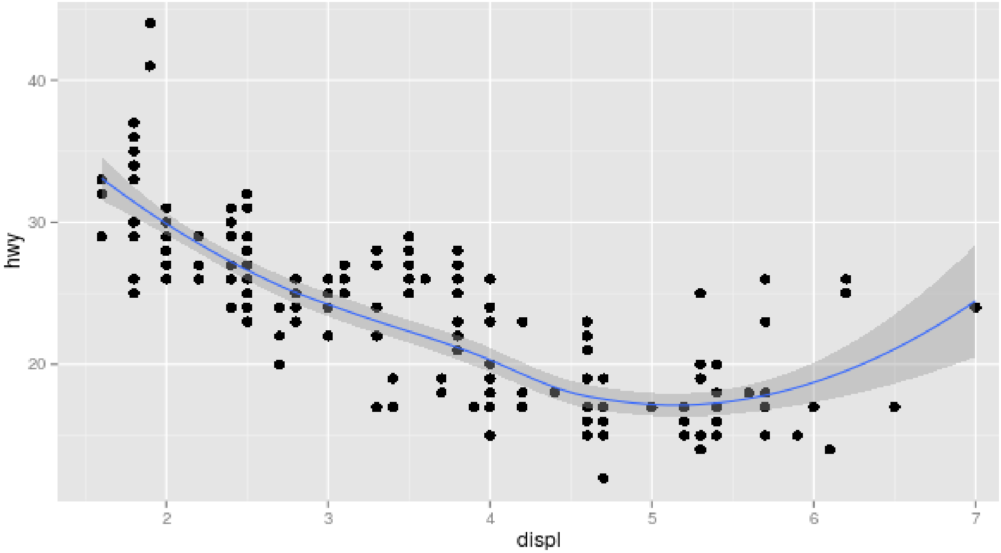
```
>p + geom_point(color=”steelblue”, size=4,alpha=1/2)
```



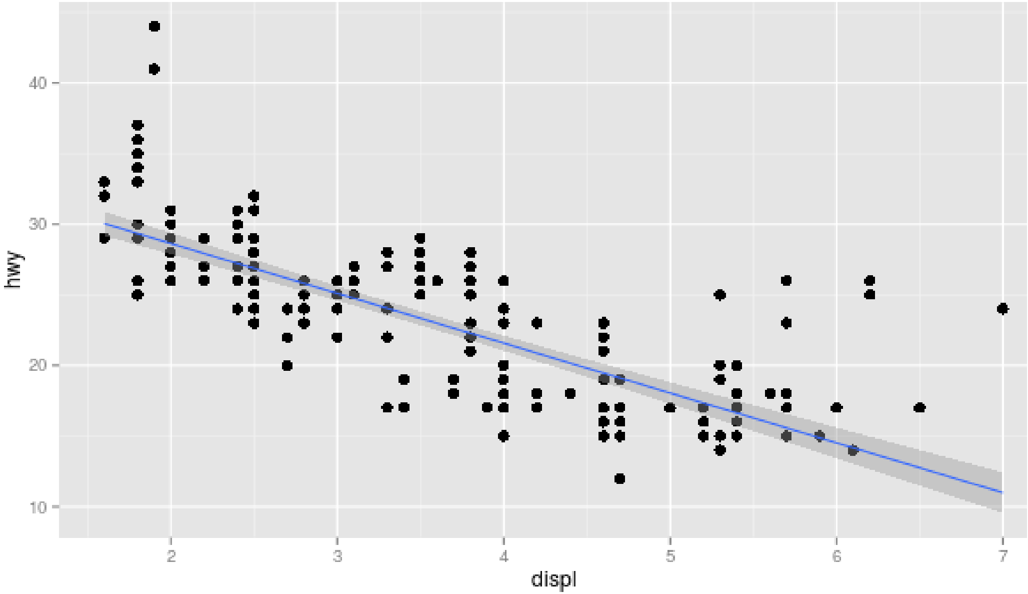
```
>p + geom_point(aes(color=drv), size=4,alpha=1)
```



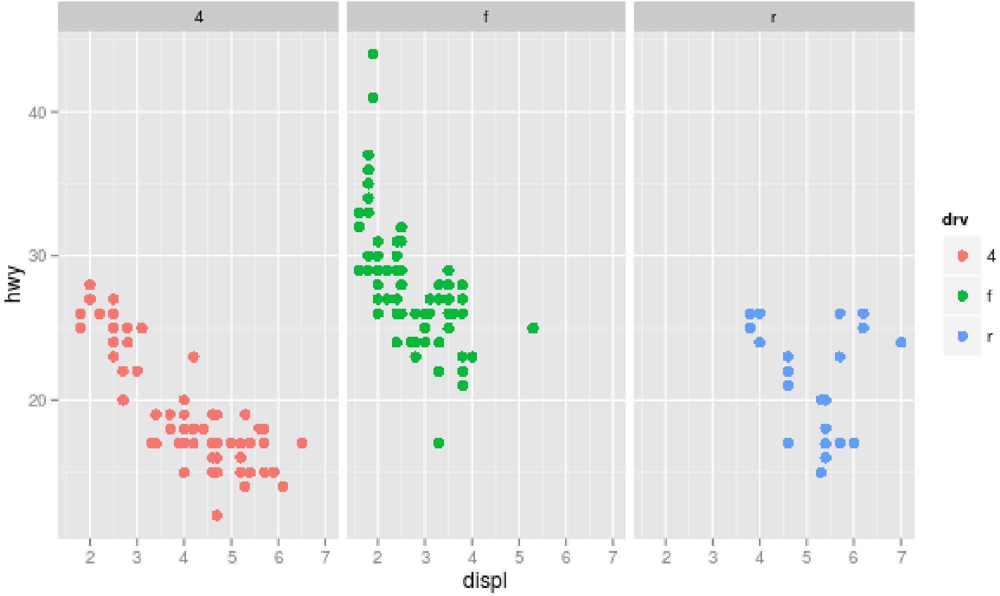
```
>p + geom_point() + geom_smooth()
```



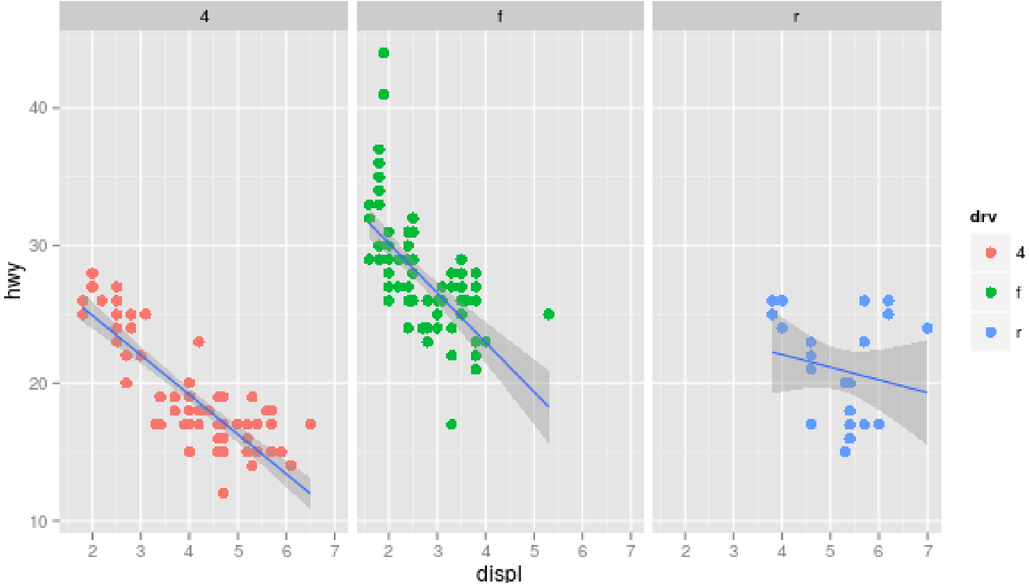
```
>p + geom_point() + geom_smooth(method = “lm”)
```



```
>p + geom_point( aes(color=drv)) + facet_grid(.~drv)
```

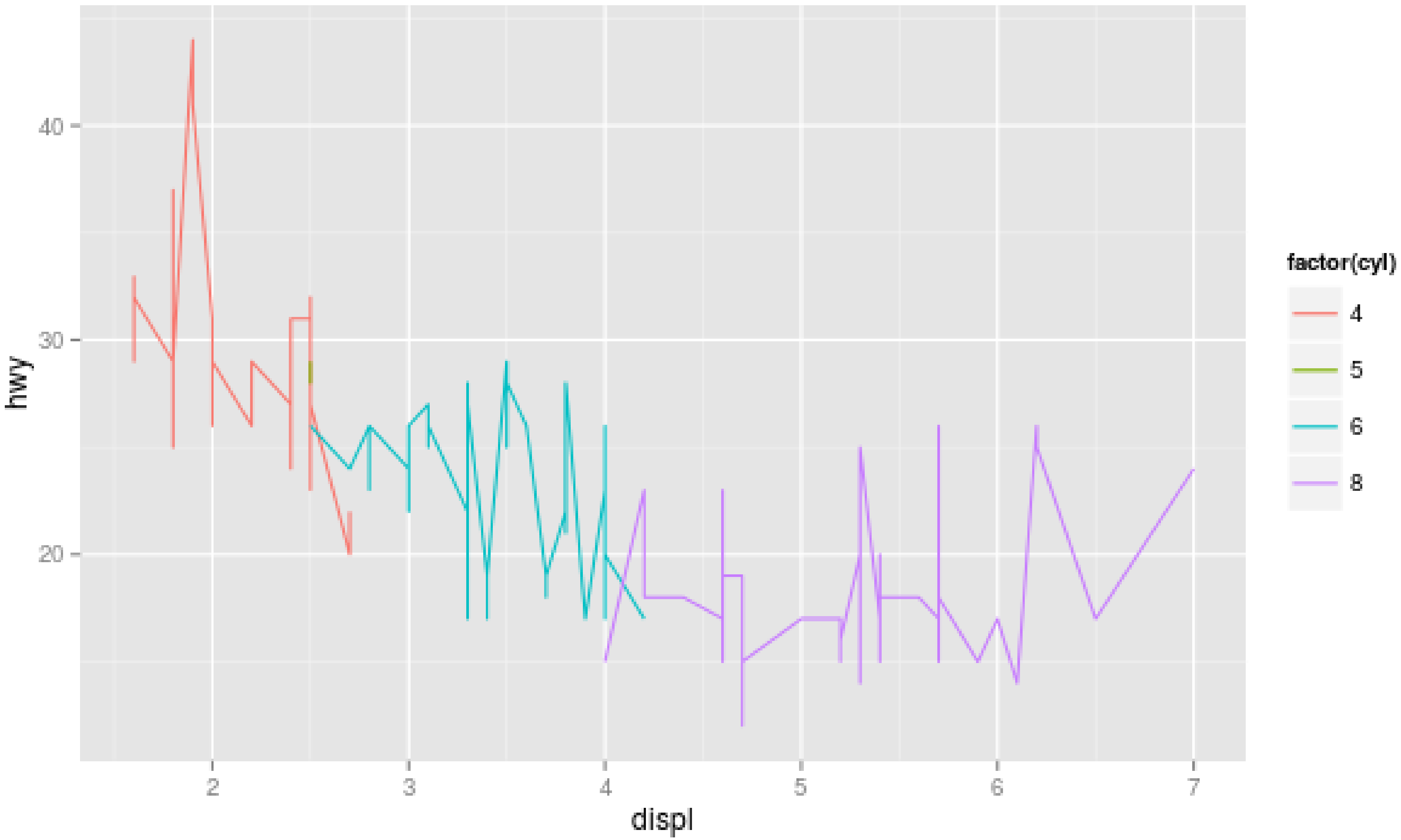


```
>p + geom_point() + facet_grid(.~drv) + geom_smooth(method=”lm”)
```



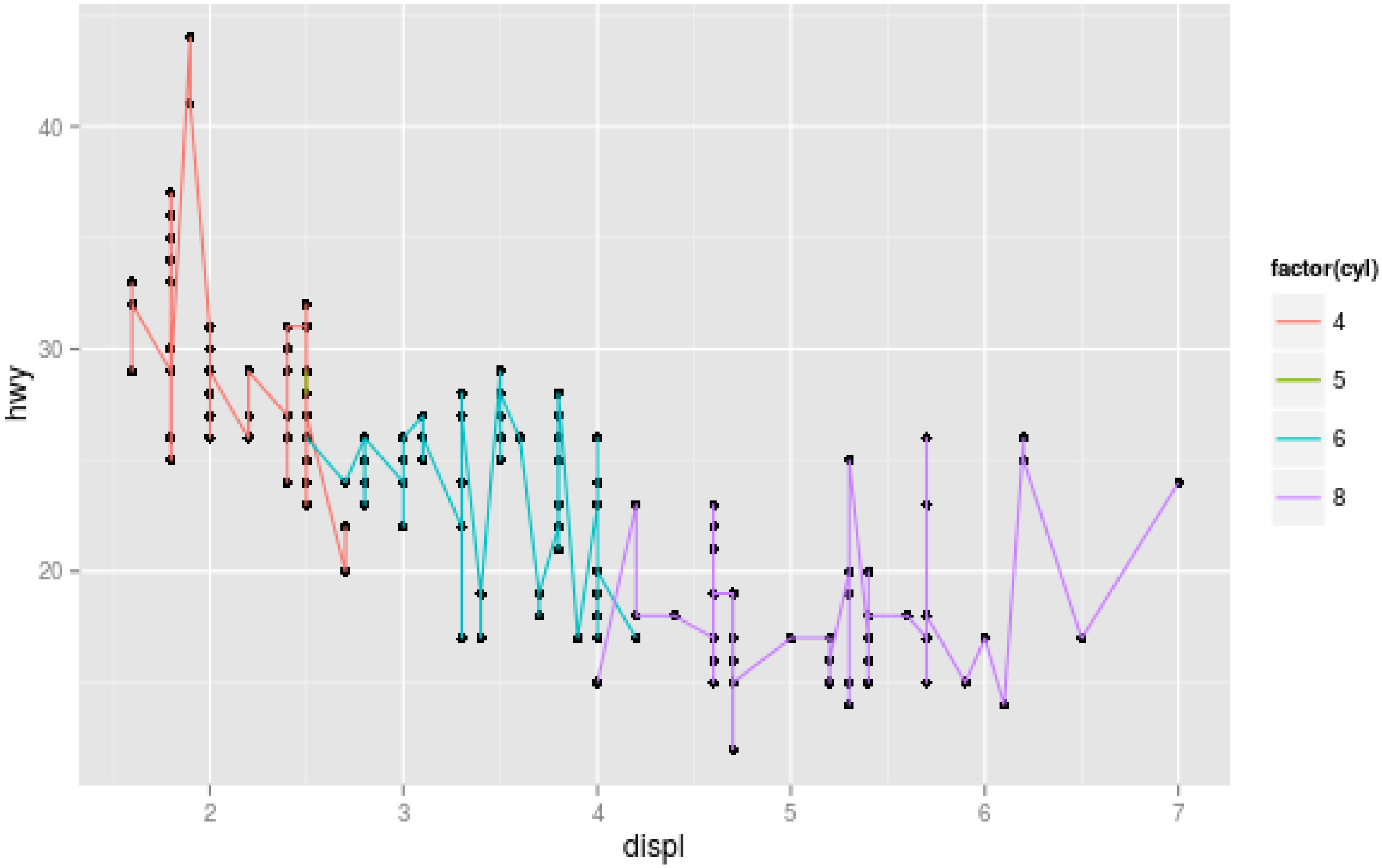
LINE PLOT

```
> p  + geom_line(aes(color=cyl))
```



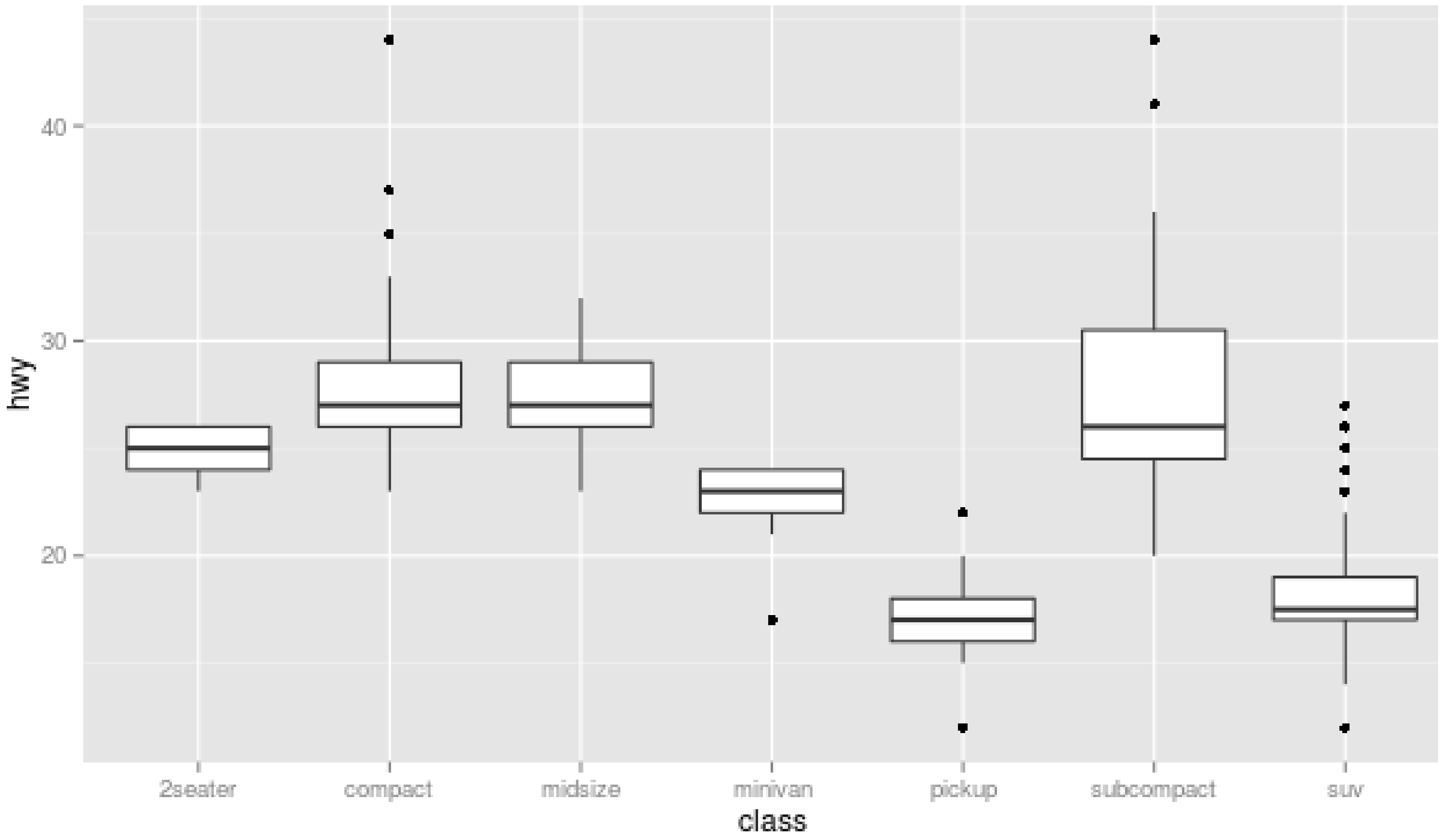
LINE PLOT

```
> p + geom_point() + geom_line(aes(color=factor(cyl)))
```



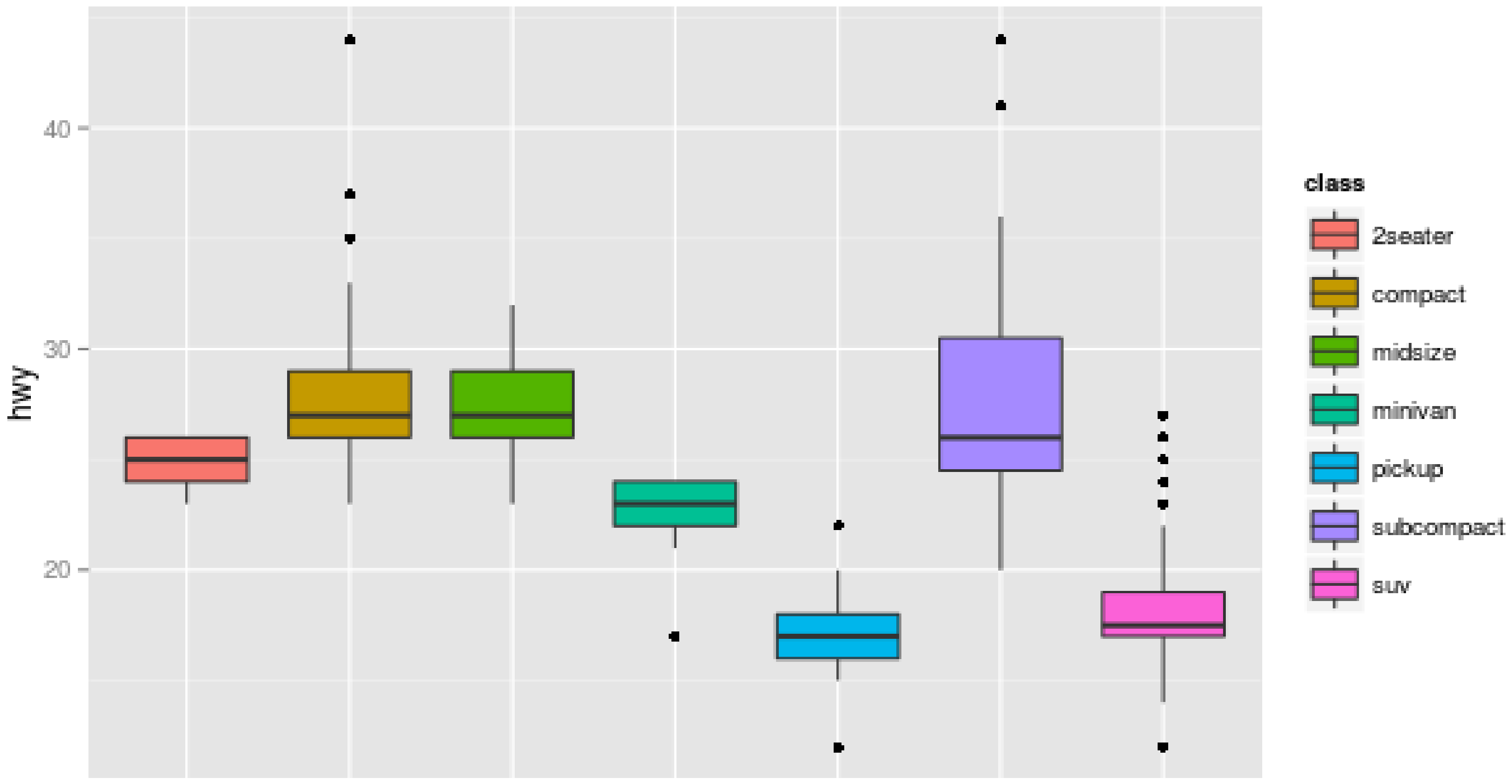
BOX PLOT

```
> ggplot(mpg,aes(class,hwy))+geom_boxplot()
```



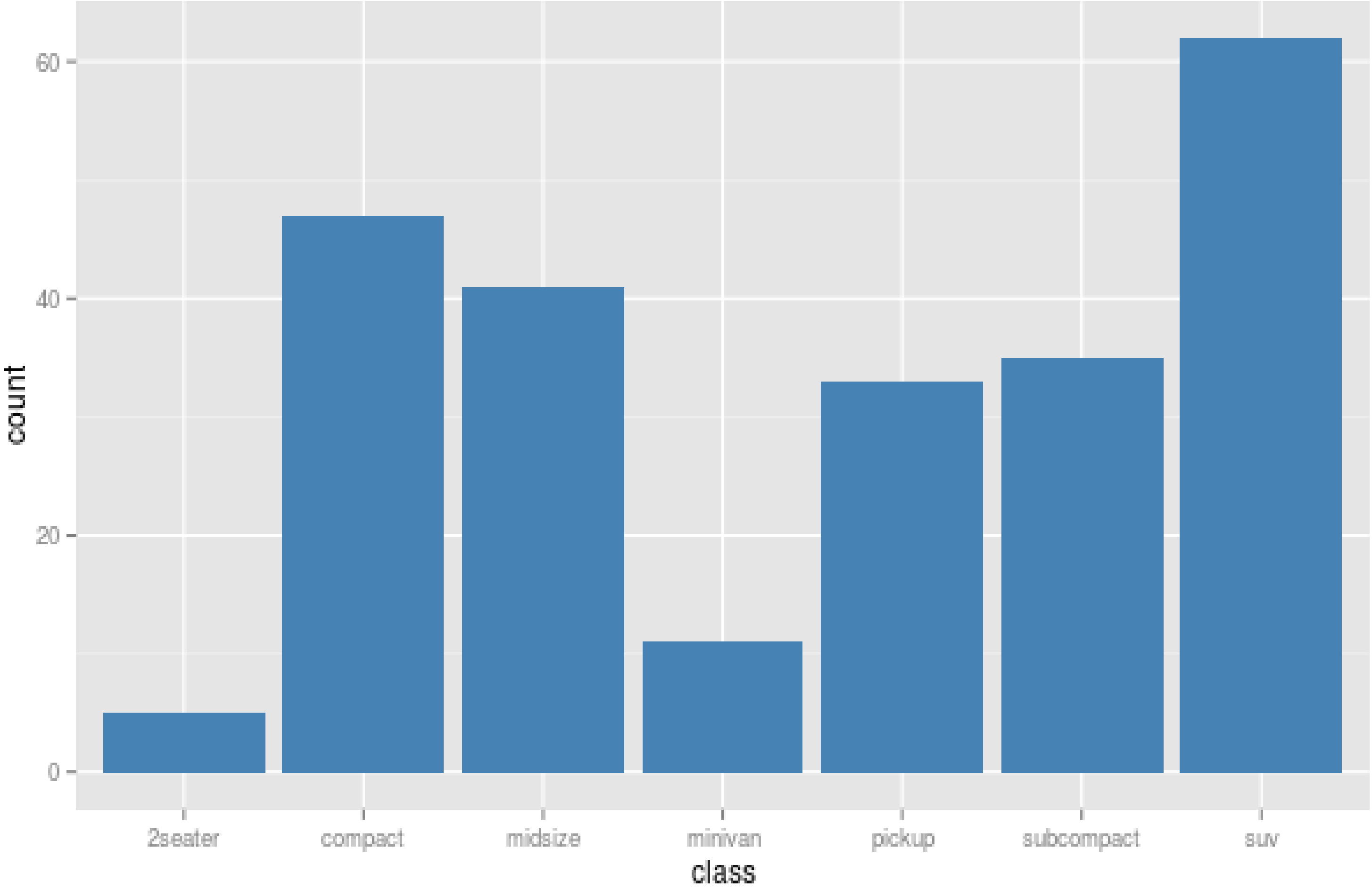
BOX PLOT

>ggplot(mpg,aes(class,hwy))+geom_boxplot(aes(fill=class))

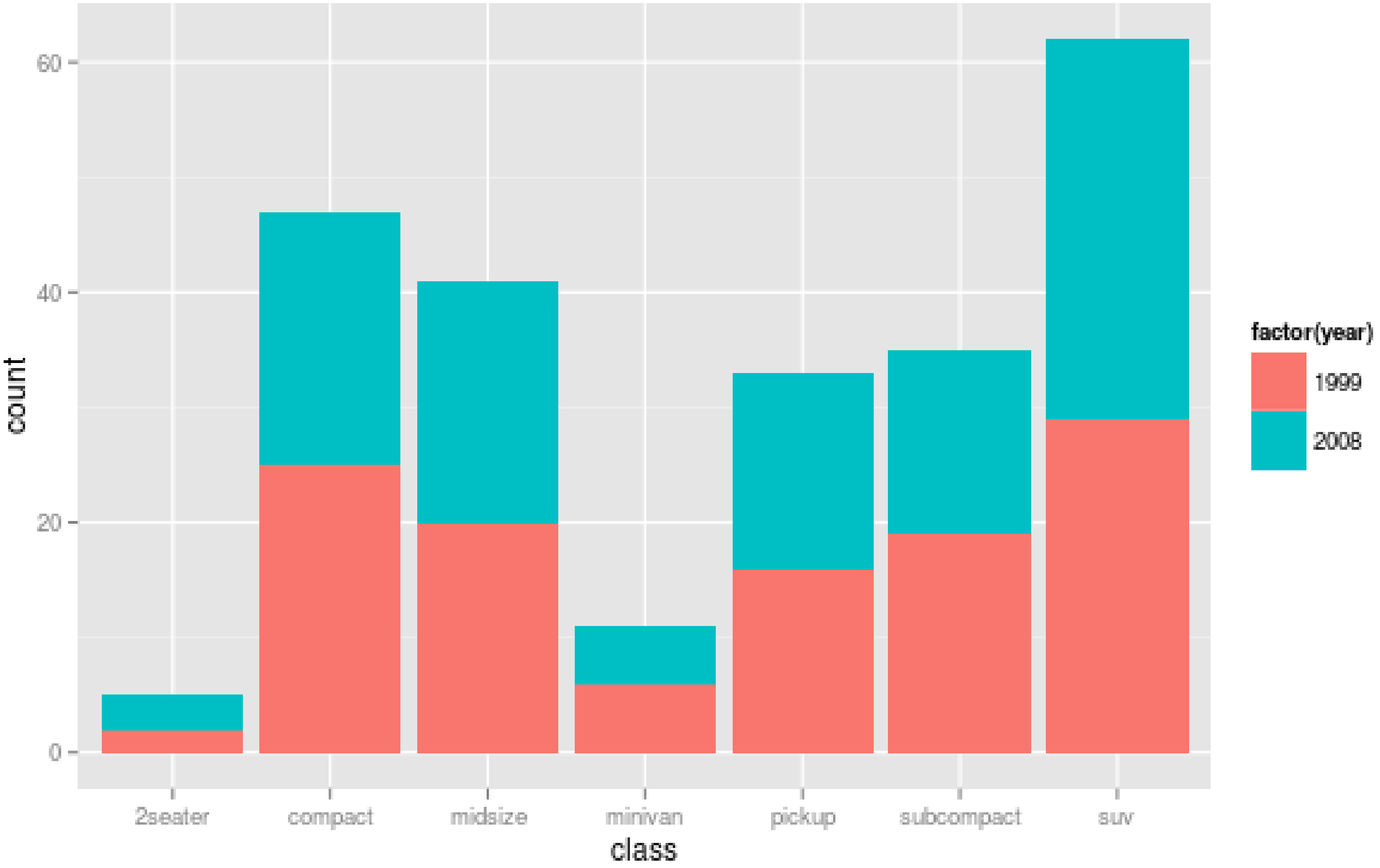


BAR CHART

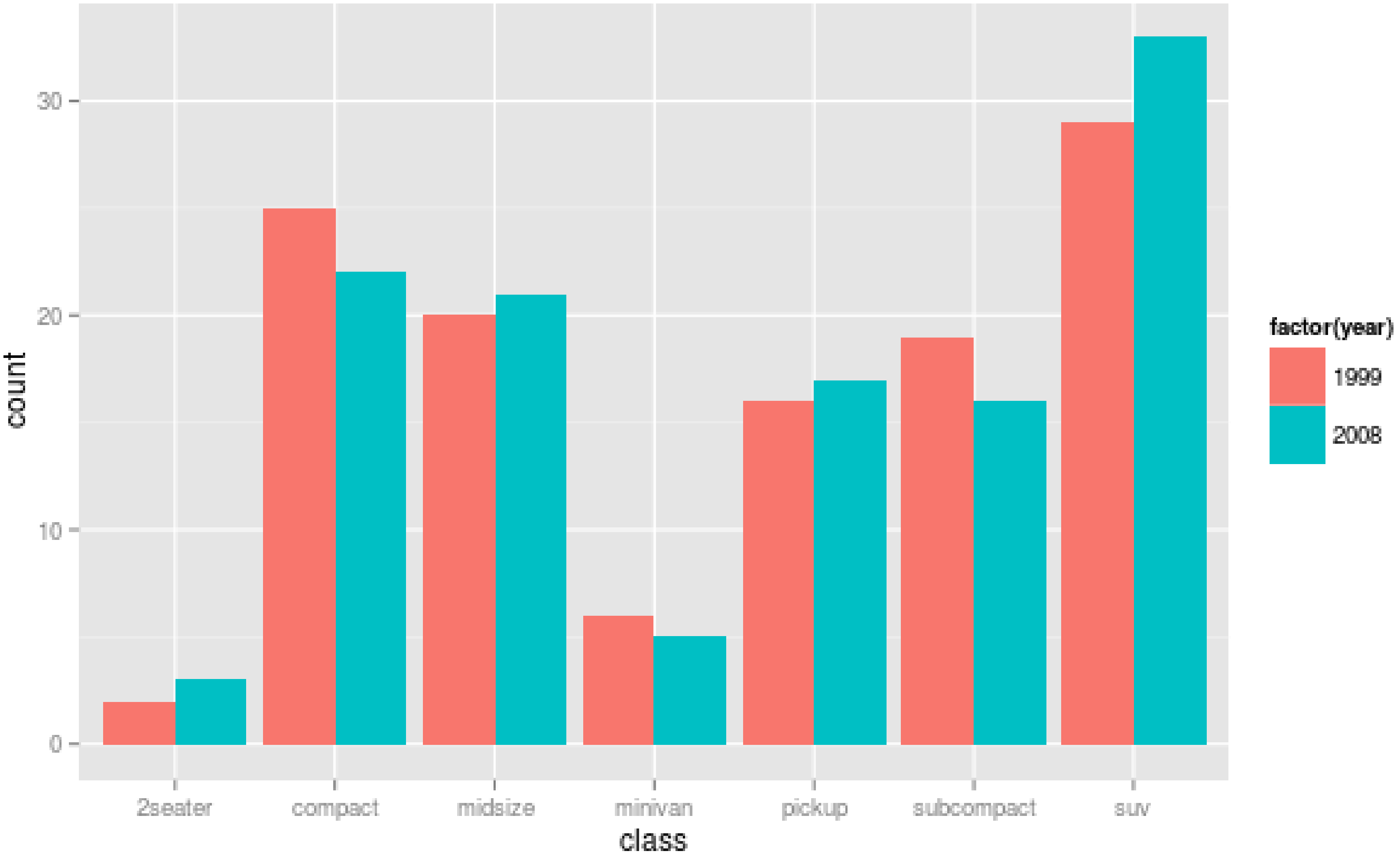
```
>ggplot(mpg,aes(class)) + geom_bar(fill="steelblue")
```



```
>ggplot(mpg,aes(class)) + geom_bar(aes(fill=factor(year)),position="stack")
```

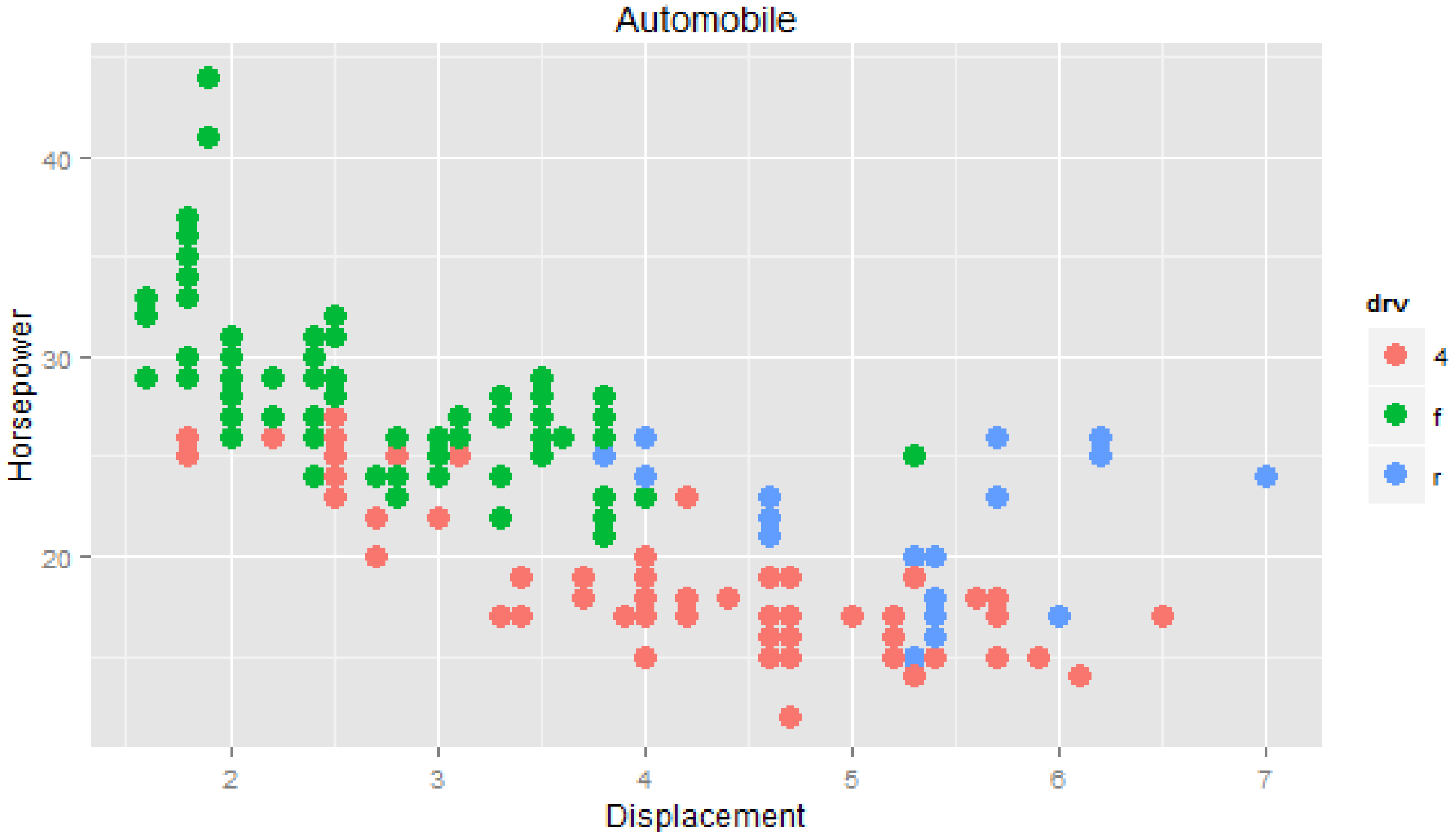


```
>ggplot(mpg,aes(class)) + geom_bar(aes(fill=factor(year)),position="dodge")
```



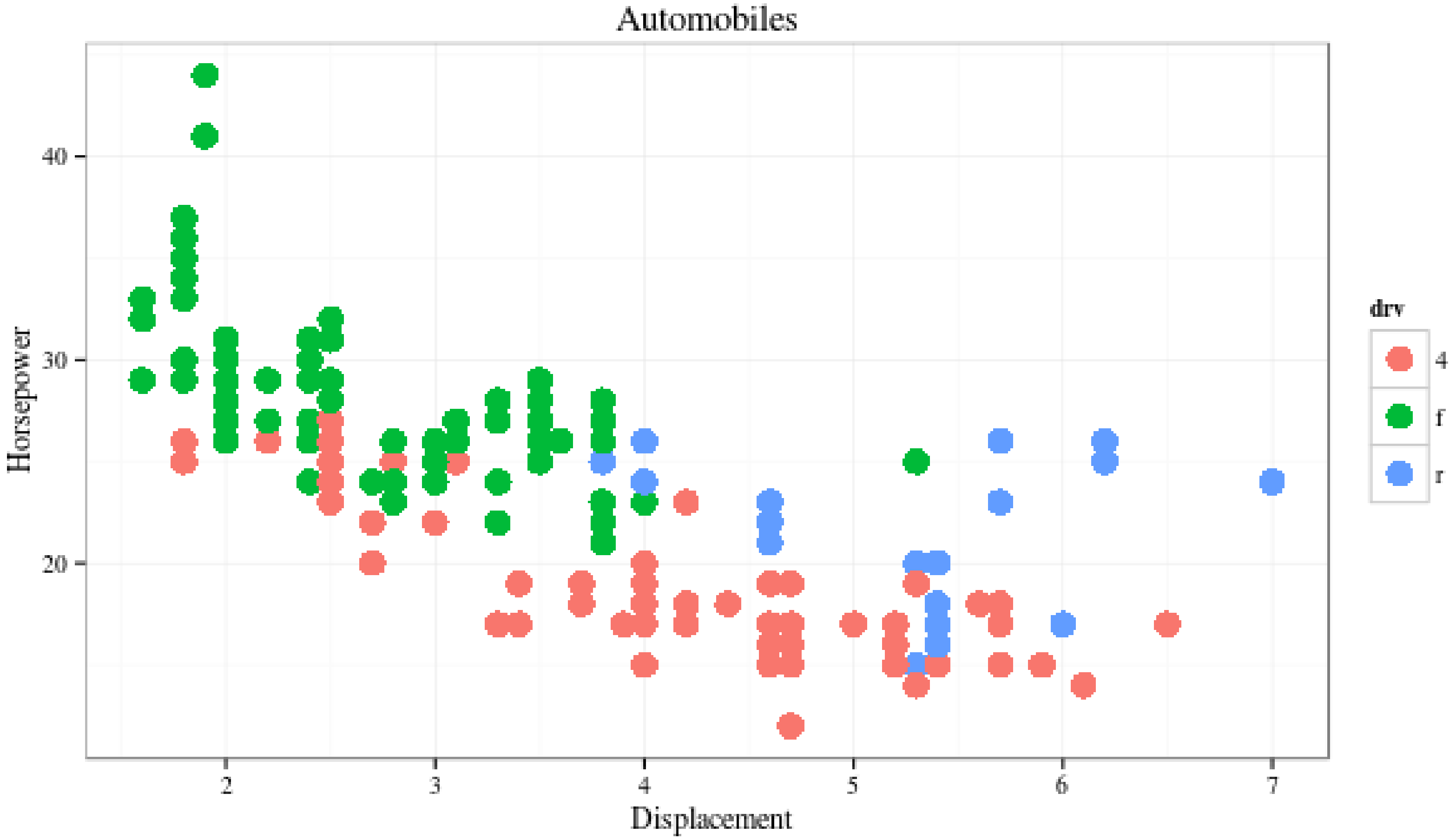
Modifying labels

```
p<- ggplot(mpg, aes(displ,hwy))
>p+geom_point(aes(color=drv)) + labs( title = “Cars”) + labs(x= “Displacement”)
+ labs(y=”Horsepower”)
```



Theme changing

```
> p + geom_point() + theme_bw(base_family = "Times")
```



Contact

- vinodh.krishnaraju@gmail.com
- www.vinodhkrishnaraju.com