



CORSO PREPARATORIO AGLI ESAMI DI STATO Seconda Sessione 2018

ETICA E PRATICA PROFESSIONALE DELL'INGEGNERE

9 – 10 gennaio 2019

SEDE ORDINE INGEGNERI DELLA PROVINCIA DI SALERNO

Relatore: Ing. Loredana Caruccio, Ph.D

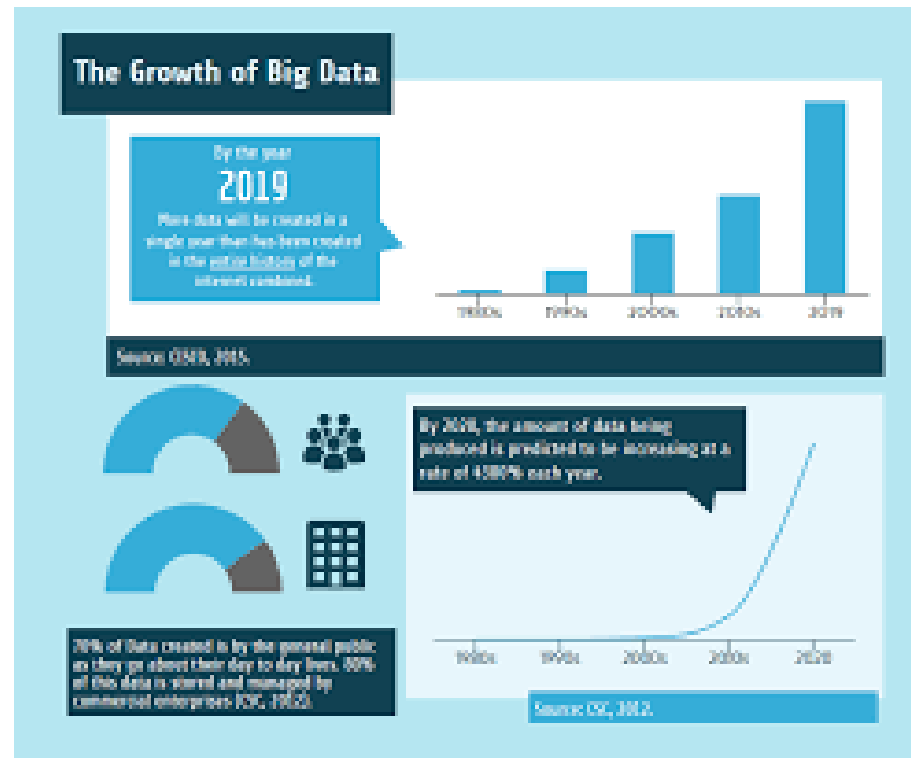
Agenda

- ▶ Big Data
 - ▶ Perché?
 - ▶ Le “V” dei Big Data
 - ▶ Applicazioni
 - ▶ Considerazioni
 - ▶ Visualizzazione dei dati
 - ▶ Creare grafici con R: Istogrammi, Scatter Plot, Bubble Blot
 - ▶ Grafici avanzati con R
 - ▶ Altri strumenti di visualizzazione
-



Perché?

- ▶ Ogni giorno vengono generati exabytes di dati sul Web
 - ▶ “Il tasso di crescita dei dati generati dal 2009 al 2020 è stimato a +4300%”



Perché?

- ▶ L'idea di base dietro la frase “Big Data” è che, tutto quello che facciamo, lascia delle tracce digitali
- ▶ Big Data si riferisce quindi alla nostra capacità di utilizzare questi crescenti volumi di dati

*NOBODY WILL
EVER NEED MORE
THAN 640K RAM.*

Bill Gates, 1981

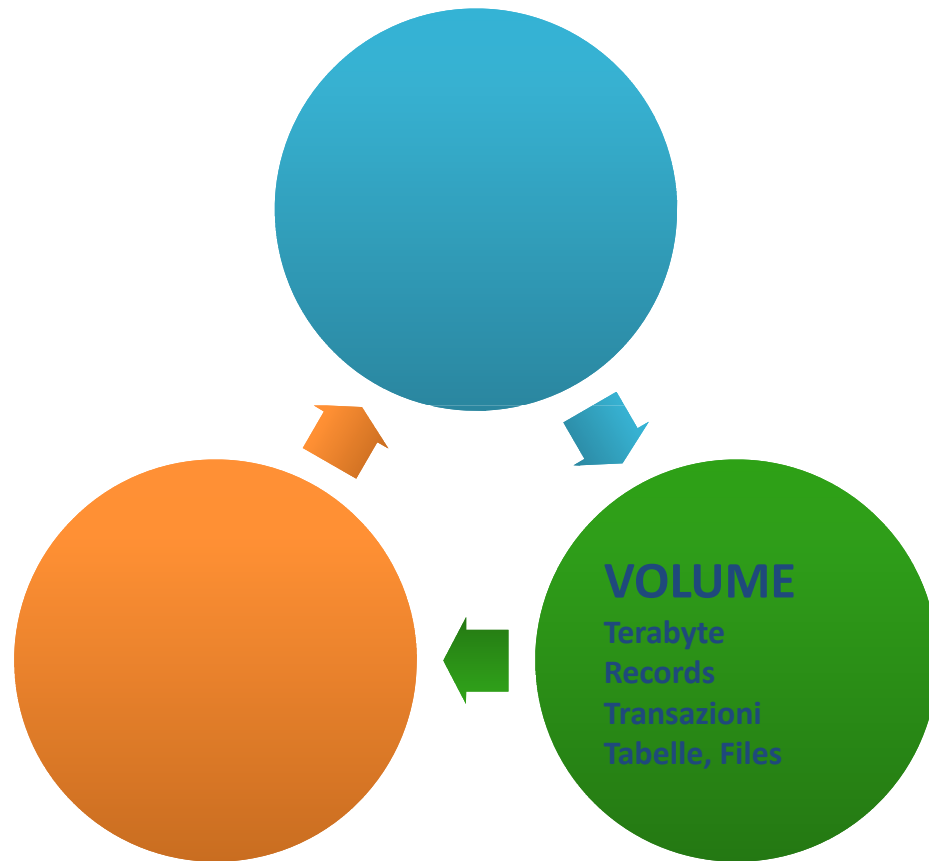


Attività che generano dati



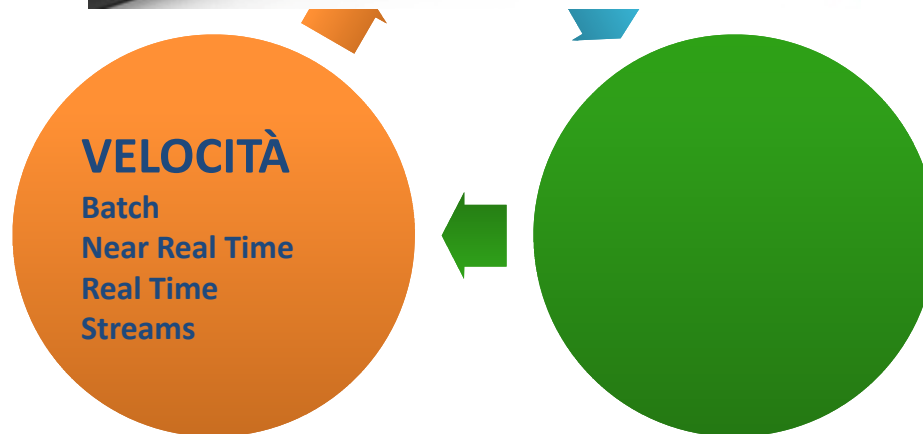
É difficile
immaginare
qualsiasi
attività che
non
genera dati

Le 3V dei Big Data



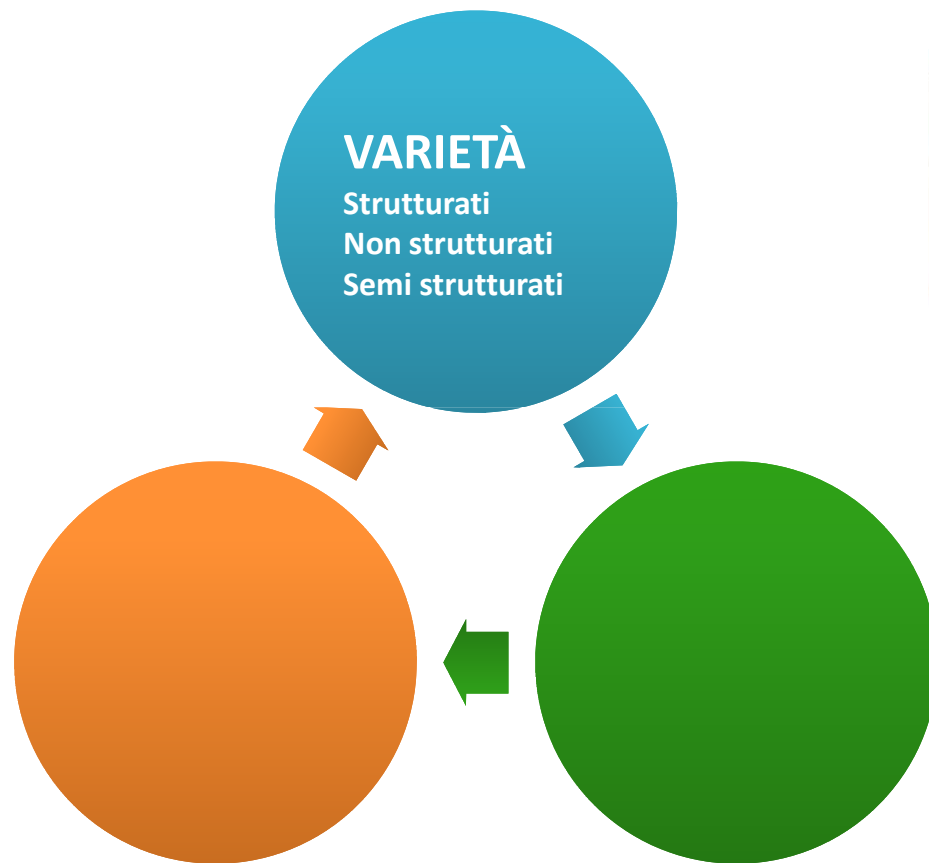
Kilobyte	KB
Megabyte	MB
Gigabyte	GB
Terabyte	TB
Petabyte	PB
Exabyte	EB
Zettabyte	ZB
Yottabyte	YB

Le 3V dei Big Data

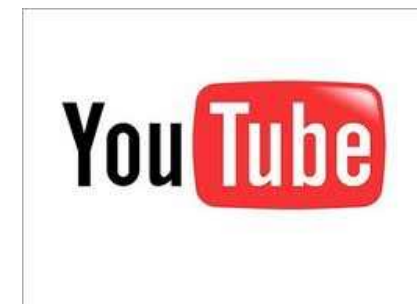


- ▶ Le aziende a dover prendere delle decisioni in tempi molto rapidi.
 - ▶ il monitoraggio del sentiment, con analisi dei messaggi sui social network che parlano del proprio brand
- ▶ Velocità di obsolescenza dell'informazione.
 - ▶ I dati invecchiano in fretta

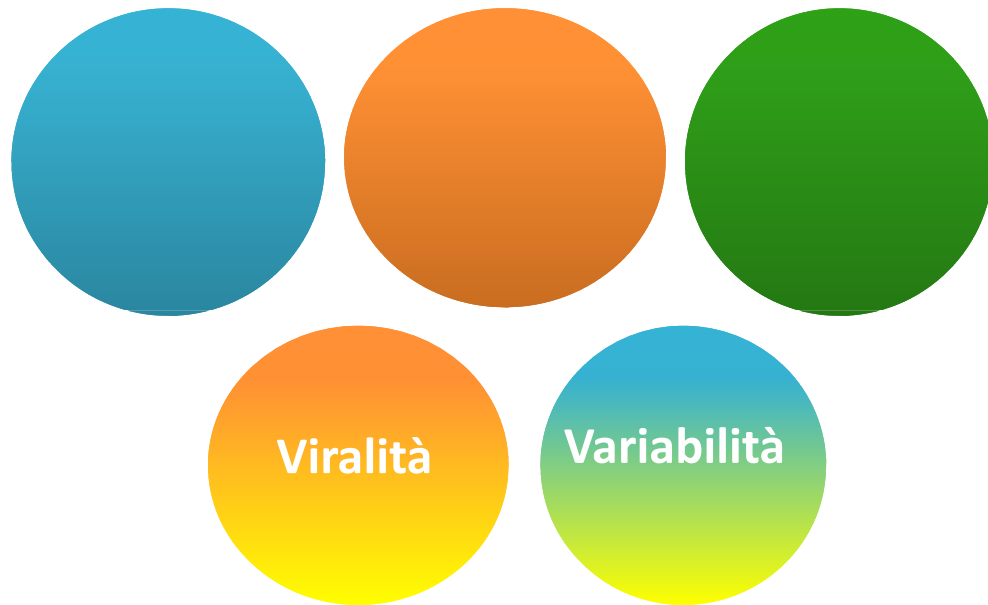
Le 3V dei Big Data



LinkedIn



.....nuove V dei Big Data...



Viralità: la grande quantità di dati e la velocità con cui vengono generati fa sì che le reazioni ad un evento si propaghino rapidamente e a grande distanza, con caratteristiche di viralità.

Variabilità: l'interpretazione di uno stesso dato può variare in funzione del contesto in cui questo viene raccolto ed analizzato.

Dati in Conversazioni

facebook

Search

Home

FAVORITES

News Feed

Messages

Other





Events

APPS

Pokes

Photos

Apps and Games

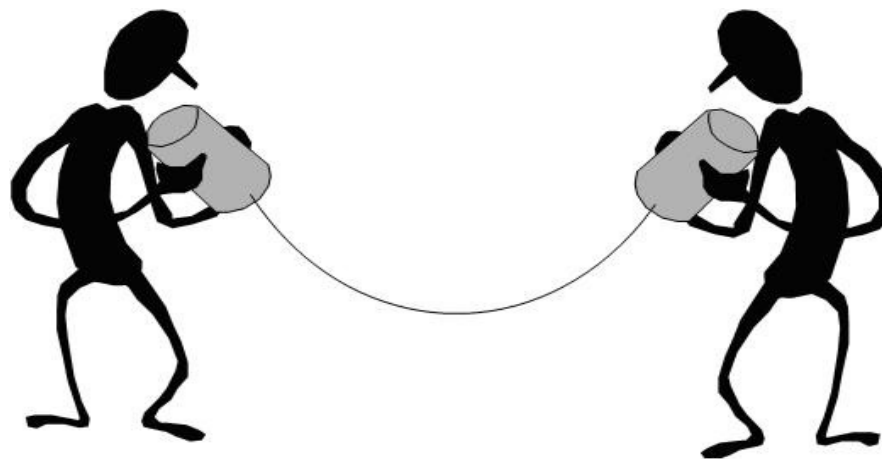


Name, Birthday, Family

Monetizable Intent

IOT: Internet of things

- L'Internet delle cose collega diversi dispositivi tra di loro.
- Ad esempio, i dati del traffico, catturati da sensori sulla strada, potrebbero essere inviati al vostro orologio di allarme per svegliarvi prima del previsto se la strada è bloccata.



Ulteriori applicazioni

Sanità



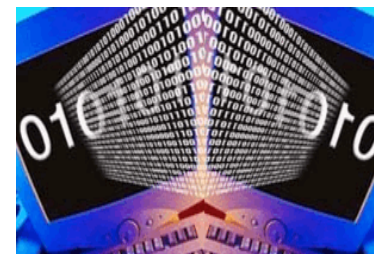
Vendita su più canali



Finanza



Analisi dei log



Sicurezza



Controllo del traffico



Telecomunicazioni



Controllo qualità



Settore manifatturiero



Trading Analytics



Frodi




Retail: Churn, NBO

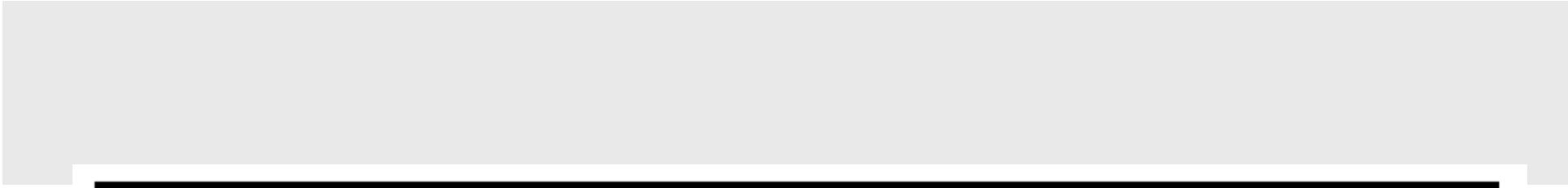





Ci sono alcune cose che sono così grandi che hanno implicazioni per tutti...

I Big Data sono una di quelle cose e, stanno trasformando completamente il modo di fare business...



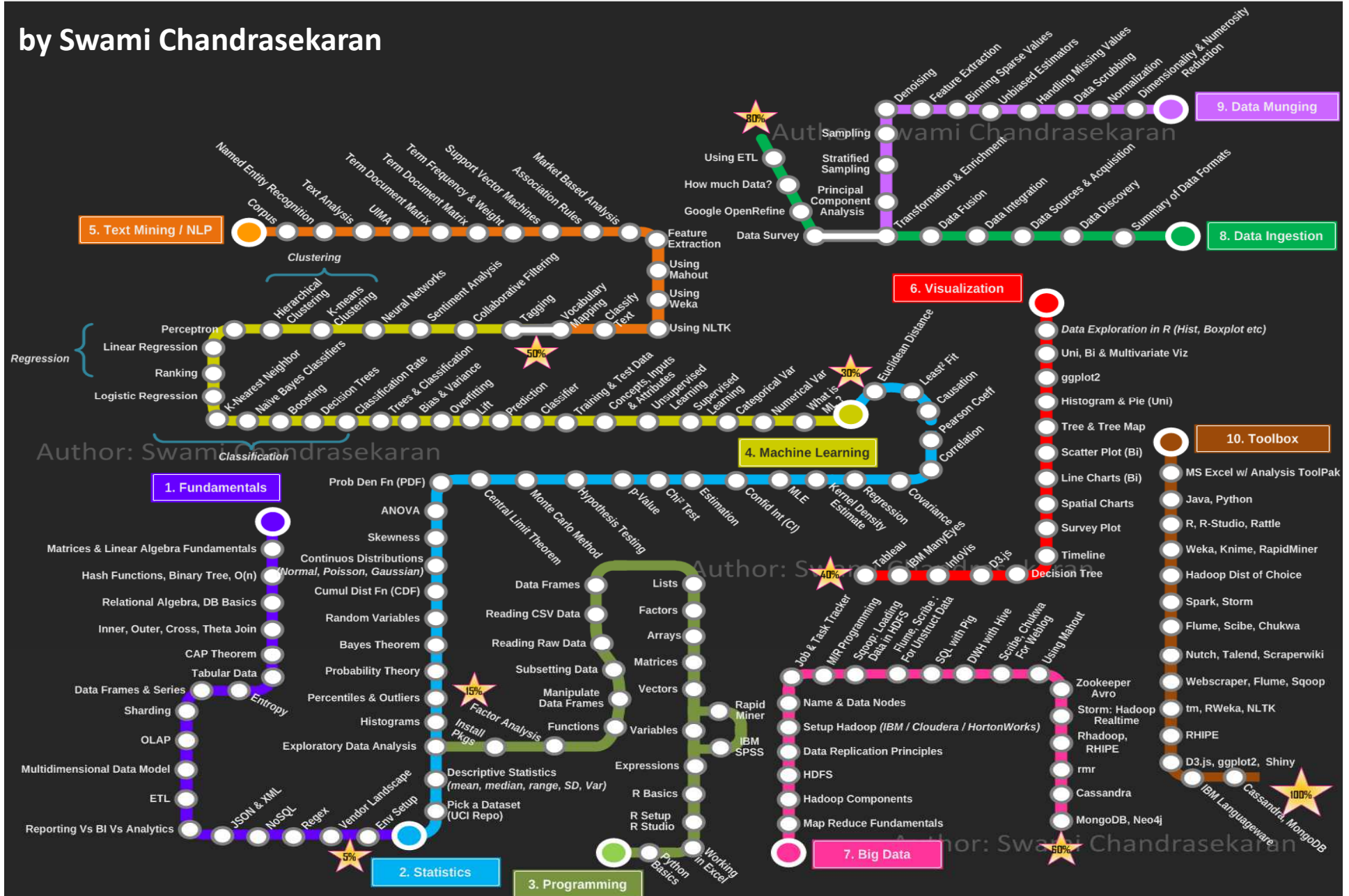


Il problema per le aziende è che questi dati sono utili solo se è possibile estrarre informazioni preziose ed agire sulla base di queste



Data Science

by Swami Chandrasekaran



Visualizzazione dei dati



Visualizzazione dei dati

*La **visualizzazione dei dati** permette di mappare dati numerici e non numerici in oggetti grafici che possono essere descritti in attributi percettibili*

- ▶ Forme: linee, punti, barre, ...
 - ▶ Colori: tonalità, trasparenze, saturazioni, ...
 - ▶ Localizzazione: all'interno di sistemi di coordinate, ...
-
- ▶ Precedentemente, la Data Visualization (visualizzazione dei dati) veniva chiamata **Computer Graphics**

Obiettivi della visualizzazione dei dati

- ▶ Diverse motivazioni per la visualizzazione dei dati dipendono dalla fase del processo **analisi dei dati** in cui è richiesta
 - ▶ Durante l'analisi dei dati
 - ▶ Comprendere i dati
 - ▶ Scoprire relazioni tra i dati
 - ▶ Contribuire alla costruzione di modelli di analisi
 - ▶ Alla fine dell'analisi
 - ▶ Comunicare i risultati
-

Strumenti per la visualizzazione dei dati

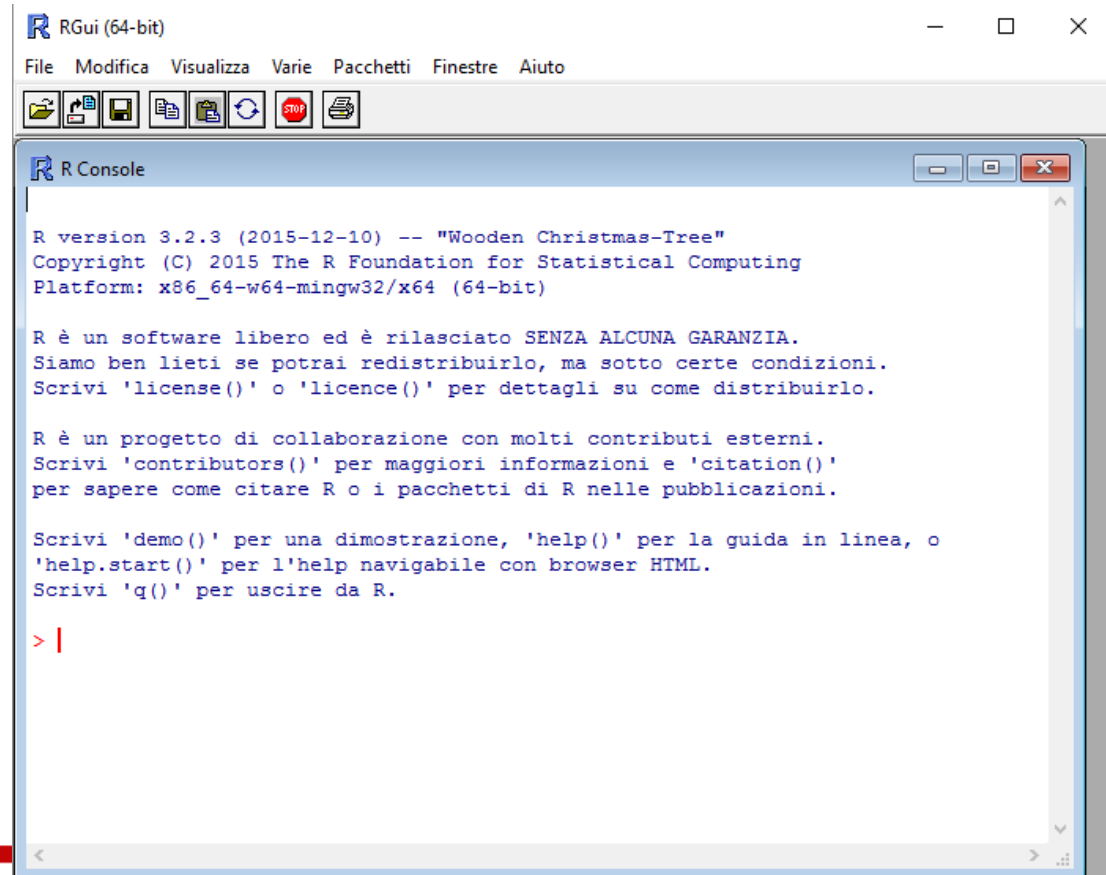
- ▶ Un data scientist sceglierà uno dei seguenti strumenti/software per effettuare data visualization
 - ▶ R
 - ▶ Un software free ed open source
 - ▶ È il software maggiormente utilizzato
 - ▶ Python
 - ▶ Un software free ed open source
 - ▶ È il maggiore competitor di R
 - ▶ SAS
 - ▶ Un software commerciale e costoso
-

Download di R

- ▶ R può essere scaricato attraverso il link

- ▶ <http://www.r-project.org>

- ▶ L'interfaccia principale di R è la
R console



The screenshot shows the RGui (64-bit) window. The title bar reads "RGui (64-bit)". The menu bar includes "File", "Modifica", "Visualizza", "Varie", "Pacchetti", "Finestre", and "Aiuto". Below the menu bar is a toolbar with icons for file operations and execution. The main window is the "R Console", which displays the following text:

```
R version 3.2.3 (2015-12-10) -- "Wooden Christmas-Tree"
Copyright (C) 2015 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R è un software libero ed è rilasciato SENZA ALCUNA GARANZIA.
Siamo ben lieti se potrai redistribuirlo, ma sotto certe condizioni.
Scrivi 'license()' o 'licence()' per dettagli su come distribuirlo.

R è un progetto di collaborazione con molti contributi esterni.
Scrivi 'contributors()' per maggiori informazioni e 'citation()'
per sapere come citare R o i pacchetti di R nelle pubblicazioni.

Scrivi 'demo()' per una dimostrazione, 'help()' per la guida in linea, o
'help.start()' per l'help navigabile con browser HTML.
Scrivi 'q()' per uscire da R.

> |
```

Package di R

- ▶ Le funzioni di R sono raggruppate in package (pacchetti)
 - ▶ Diverse funzioni di base sono caricate nella versione di base di R
 - ▶ Altre funzioni, chiamate **contributed function** devono essere scaricate dai server CRAN
 - ▶ I principali pacchetti per la data visualization sono
 - ▶ base R
 - ▶ lessR
 - ▶ ggplot2
 - ▶ **base R** rappresentano i pacchetti caricati in maniera automatica quanto si installa R
 - ▶ Essi richiedono uno sforzo di programmazione per ottenere una presentazione grafica di qualità
-

lessR

- ▶ **lessR (v3.5.5)** rappresenta il pacchetto fornito dagli autori del libro “R Data Analysis without Programming”
 - ▶ Non è un pacchetto grafico
 - ▶ utile per effettuare analisi dei dati
 - ▶ Vantaggi
 - ▶ Fornisce funzioni semplificate per effettuare analisi di base
 - ▶ Fornisce molta flessibilità sugli input
 - ▶ Svantaggi
 - ▶ La progettazione dell’output grafico dipende dalle opzioni messe a disposizione
 - ▶ Caricamento della libreria
 - ▶ **library(**lessR**)**
-

ggplot2

- ▶ **ggplot2 (v2.2.1)** rappresenta il pacchetto fornito da Hadley Wickham
 - ▶ È il pacchetto grafico maggiormente utilizzato dai data scientist, che fornisce molta flessibilità nel creare qualsiasi tipo di visualizzazione
 - ▶ Vantaggi
 - ▶ Fornisce un insieme di strumenti per creare virtualmente qualsiasi tipo di grafico
 - ▶ Permette di costruire grafici layer-by-layer
 - ▶ Caricamento della libreria
 - ▶ **library(tidyverse)**
-

Caricamento dei dati

- ▶ Per caricare i dati nel data frame (tabella) si utilizza la funzione Read

```
mydata<-Read("Employee", format="lessR")
```

- ▶ La funzione Read visualizzerà come output la descrizione delle caratteristiche dei dati caricati
- ▶ L'attributo format permette di specificare il formato del file da leggere
 - ▶ Il valore "lessR" significa che il file dei dati è incluso in lessR

Caricamento dei dati

- Per visualizzare i dati caricati in mydata basterà digitare il comando

mydata

	Years	Gender	Dept	Salary	Satisfaction	HealthPlan	Pre	Post
Ritchie, Darnell	7	M	ADMN	43788.26	med	1	82	92
Wu, James	NA	M	SALE	84494.58	low	1	62	74
Hoang, Binh	15	M	SALE	101074.86	low	3	96	97
Jones, Alissa	5	F	<NA>	43772.58	<NA>	1	65	62
Downs, Deborah	7	F	FINC	47139.90	high	2	90	86
Afshari, Anbar	6	F	ADMN	59441.93	high	2	100	100
Knox, Michael	18	M	MKTG	89062.66	med	3	81	84
Campagna, Justin	8	M	SALE	62321.36	low	1	76	84
Kimball, Claire	8	F	MKTG	51356.69	high	2	93	92
Cooper, Lindsay	4	F	MKTG	46772.95	high	1	78	91
Saechao, Suzanne	8	F	SALE	45545.25	med	1	98	100
Pham, Scott	13	M	SALE	71871.05	high	2	90	94
Tian, Fang	9	F	ACCT	61084.02	med	2	60	61
Bellingar, Samantha	10	F	SALE	56337.83	med	1	67	72
Sheppard, Cory	14	M	FINC	85027.55	low	3	66	73
Kralik, Laura	10	F	SALE	82681.19	med	2	74	71
Skrotzki, Sara	18	F	MKTG	81352.33	med	2	63	61
Correll, Trevon	21	M	SALE	124419.23	low	1	97	94
James, Leslie	18	F	ADMN	112563.38	low	3	70	70
Osterman, Pascal	5	M	ACCT	39704.79	high	3	69	70
Adib, Hassan	14	M	SALE	73014.43	med	2	71	69
Gvakharia, Kimberly	3	F	SALE	39868.68	med	2	83	79
Stanley, Grayson	9	M	SALE	59624.87	low	1	74	73
Link, Thomas	10	M	FINC	56312.89	low	1	83	83
Portlock, Ryan	13	M	SALE	67714.85	low	1	72	73
Langston, Matthew	5	M	SALE	39188.96	low	3	94	93
Stanley, Emma	3	F	ACCT	36124.97	high	2	86	84
Singh, Niral	2	F	ADMN	51055.44	high	2	59	59
Anderson, David	9	M	ACCT	59547.60	low	1	94	91
Fulton, Scott	13	M	SALE	77785.51	low	1	72	73
Korhalkar, Jessica	2	F	ACCT	62502.50	<NA>	2	74	87
LaRoe, Maria	10	F	MKTG	51961.29	high	2	80	86
Billing, Susan	4	F	ADMN	62675.26	med	2	91	90
Capelle, Adam	24	M	ADMN	98138.43	med	2	83	81

Grafici

- ▶ Vedremo come generare ed interpretare le seguenti tipologie di grafici
 - ▶ Istogrammi
 - ▶ Scatter Plot
 - ▶ Bubble Plot
-

Istogrammi: Base R

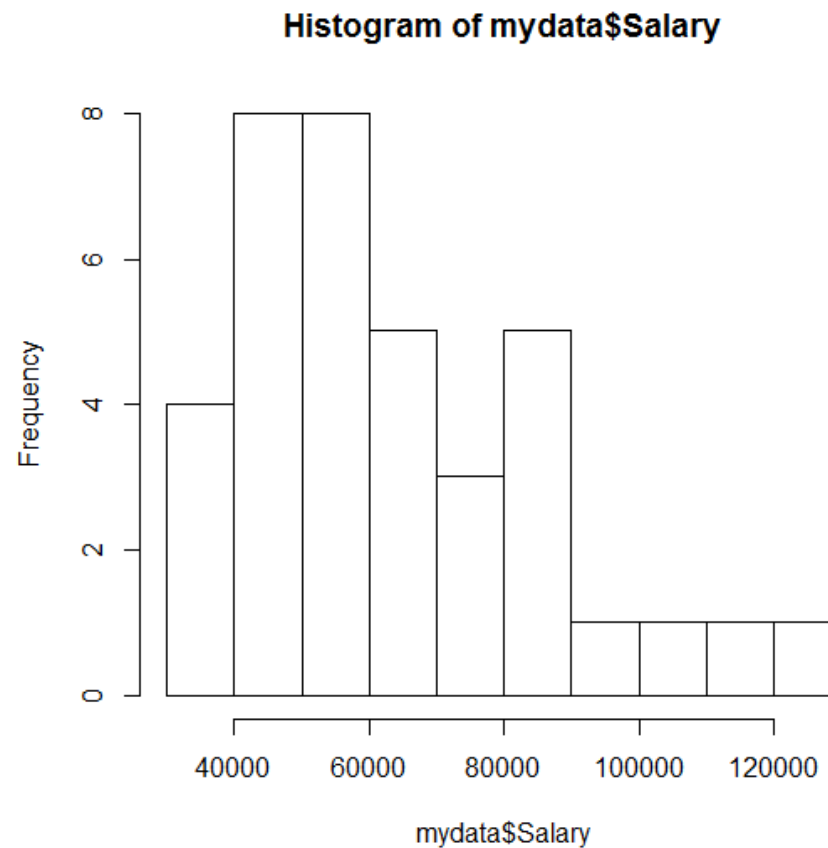
- ▶ Per ottenere un istogramma con la libreria base R si utilizza la funzione hist

```
hist(mydata$Salary)
```

- ▶ Crea l'istogramma della variabile Salary
 - ▶ Nota: è necessario specificare il nome del data frame che contiene la variabile
-

Istogrammi: Base R

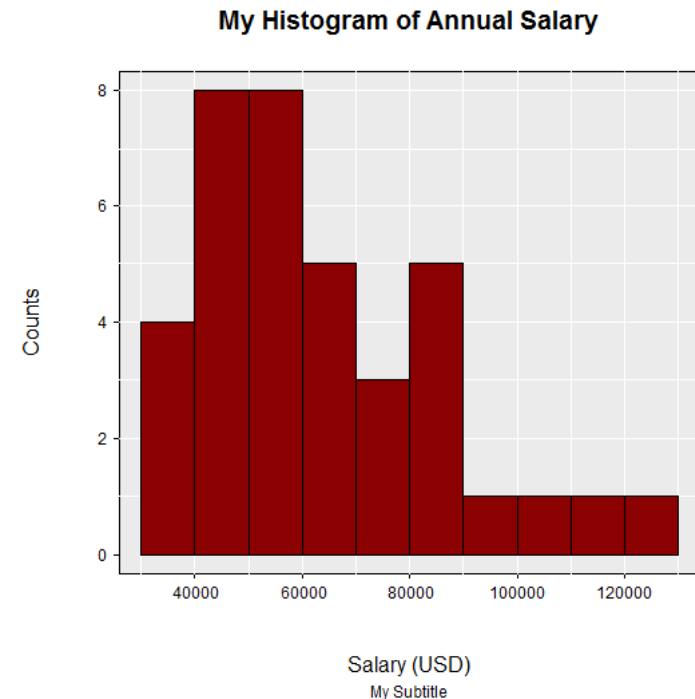
```
hist(mydata$Salary)
```



Istogrammi: Base R

- ▶ La grafica di default della libreria base R non è accettabile
 - ▶ È possibile migliorare l'aspetto del grafico

```
.fmt <- function(k, d=num.digits, w=0) {  
    format(sprintf("%.*f", d, k),  
           width=w, justify="right",  
           scientific=FALSE)}  
h <- hist(mydata$Salary, plot=FALSE)  
plot(h, border="transparent", axes=FALSE, ann=FALSE)  
axis(1, at=axTicks(1), labels=FALSE, tck=-.01)  
    text(x=axTicks(1), y=par("usr")[3],  
        labels=.fmt(axTicks(1), 0),  
        pos=1, xpd=TRUE, cex=0.75, col="black", srt=0,  
        offset=0.5)  
axis(2, at=axTicks(2), labels=FALSE, tck=-.01)  
    text(x=par("usr")[1], y=axTicks(2),  
        labels=.fmt(axTicks(2), 0),  
        pos=2, xpd=TRUE, cex=0.75, col="black")  
title(xlab="Salary (USD)")  
title(sub="My Subtitle", cex.sub=0.76)  
title(ylab="Counts")  
title(main="My Histogram of Annual Salary")  
usr <- par("usr")  
    rect(usr[1], usr[3], usr[2], usr[4],  
        col="gray92", border="black")  
vx <- h$breaks  
    abline(v=seq(vx[1], vx[length(vx)], vx[2]-vx[1]),  
          col="white", lwd=.5)  
vy <- pretty(h$counts)  
    abline(h=seq(vy[1], vy[length(vy)], vy[2]-vy[1]),  
          col="white", lwd=.5)  
plot(h, add=TRUE, col="darkred")
```



Istogrammi: lessR

- ▶ Per ottenere un istogramma con la libreria lessR si utilizza la funzione Histogram

Histogram(Salary)



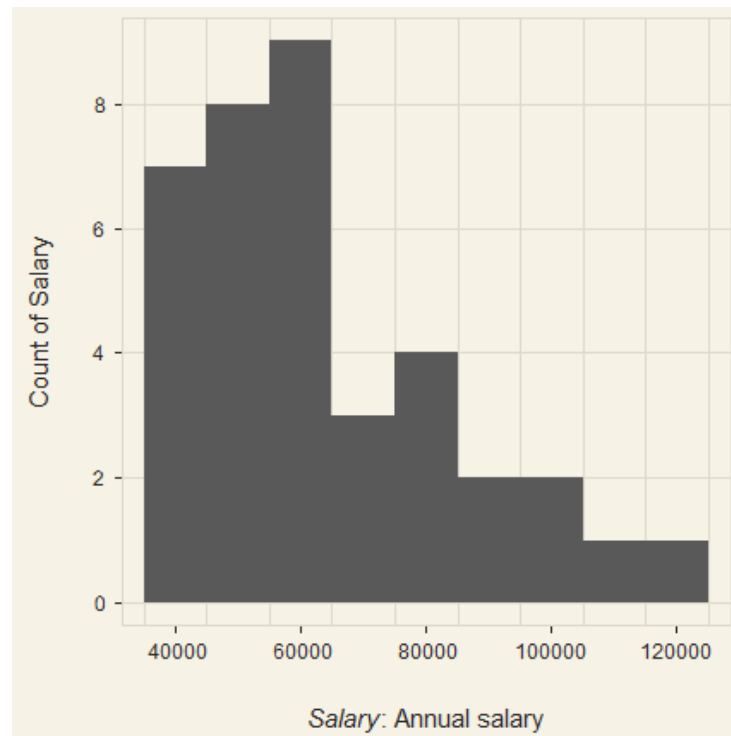
Istogrammi: lessR

- ▶ Anche se la grafica di default è abbastanza buona
 - ▶ È comunque possibile modificare alcune caratteristiche di visualizzazione
 - ▶ Modifica delle caratteristiche
 - ▶ Attributo fill -> permette di assegnare un colore alle barre
 - ▶ Attributo stroke -> permette di settare il bordo alle barre
 - ▶ Attributo bg.stroke -> permette di settare il bordo all'area del grafico
 - ▶ Attributo bin.start -> permette settare il primo valore dell'asse
 - ▶ Attributo bin.width -> permette di settare l'unità principale dei valori dell'asse
-

Istogrammi: lessR

► Modifichiamo alcune caratteristiche

```
Histogram(Salary, bin.start=35000, bin.width=10000, fill="gray35",  
color="off")
```



Istogrammi: ggplot2

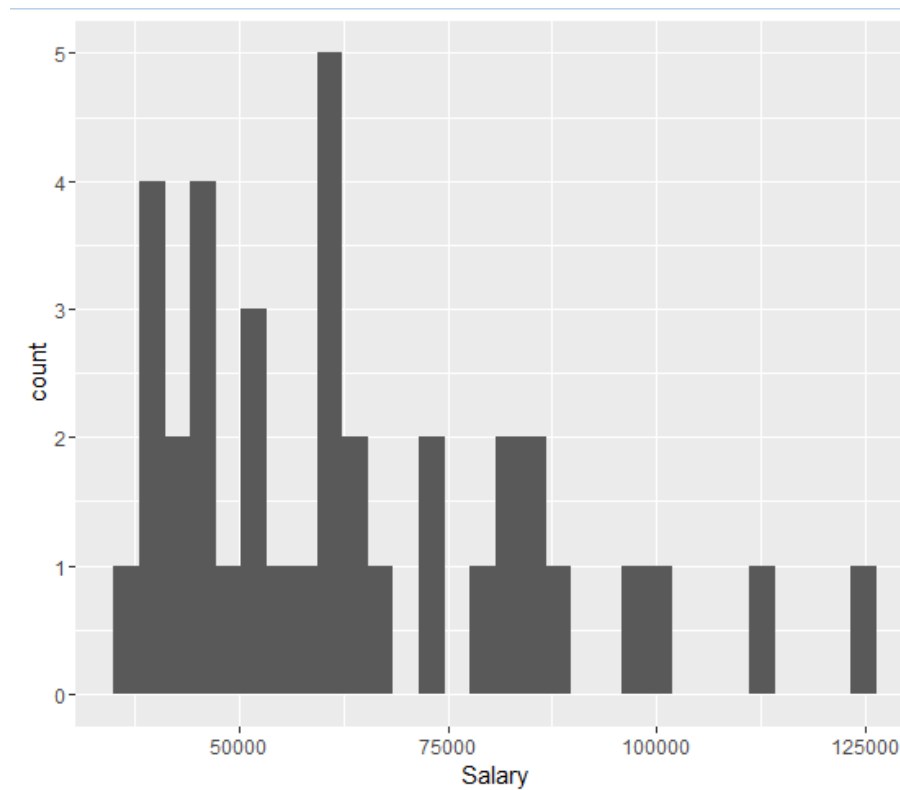
- ▶ Per ottenere un istogramma con la libreria ggplot2 si utilizza la funzione ggplot

```
ggplot(mydata, aes(Salary)) + geom_histogram()
```

- ▶ ggplot prende in input
 - ▶ il nome del data frame che contiene la variabile
 - mydata
 - ▶ la funzione aes (aesthetics) in cui insierire il nome della variabile
 - Salary
 - ▶ un Layer che specifica l'oggetto geometrico da visualizzare
 - un istogramma
-

Istogrammi: ggplot2

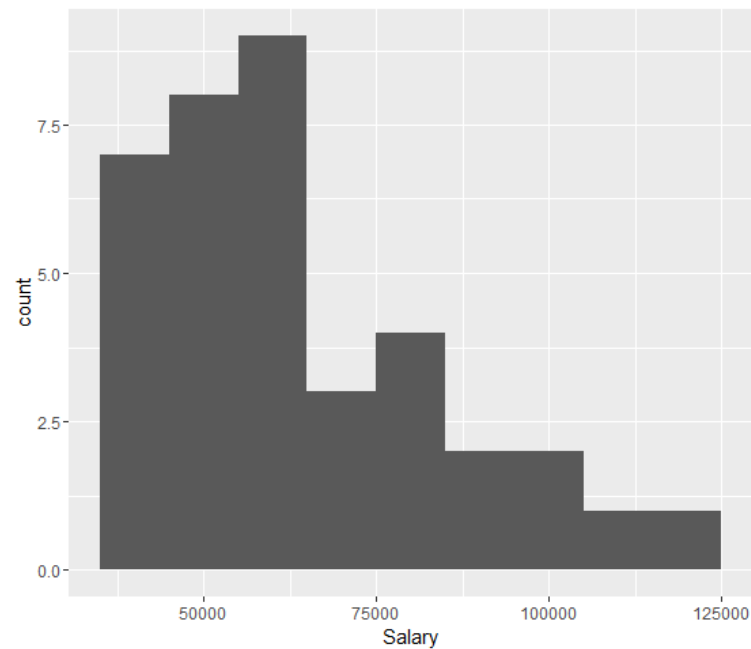
```
ggplot(mydata, aes(Salary)) + geom_histogram()
```



Istogrammi: ggplot2

- ▶ ggplot2 non utilizza l'unità principale di default di R
 - ▶ Bisogna settarla in maniera esplicita

```
ggplot(mydata, aes(Salary)) + geom_histogram(binwidth=10000)
```



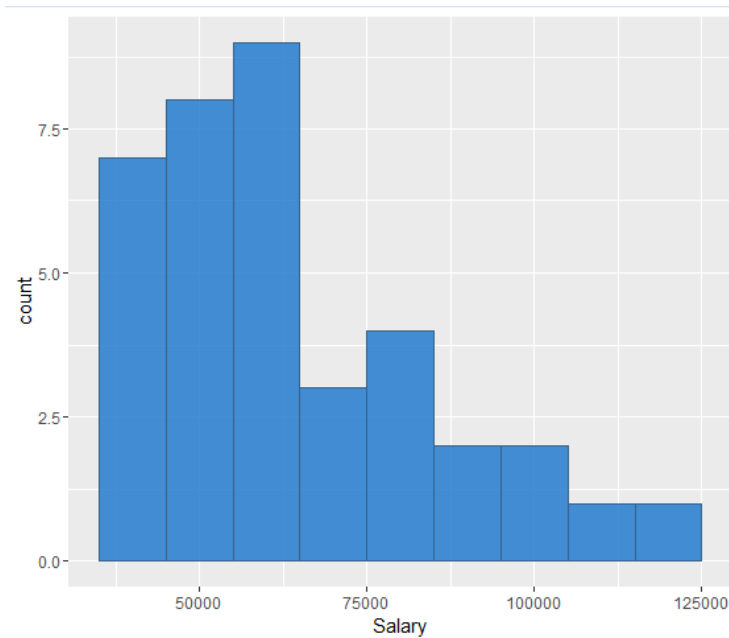
Istogrammi: ggplot2

- ▶ Anche se la grafica di default è abbastanza buona (simile a lessR)
 - ▶ È comunque possibile modificare molte caratteristiche di visualizzazione
 - ▶ Modifica delle caratteristiche
 - ▶ Attributo fill -> permette di assegnare un colore interno alle barre
 - ▶ Attributo color -> permette di assegnare un colore al bordo delle barre
 - ▶ Attributo alpha -> permette di settare la trasparenza del colore nel range [0,1]
 - ▶ Attributo size -> permette di specificare la larghezza dei bordi
-

Istogrammi: ggplot2

- Modifichiamo alcune caratteristiche

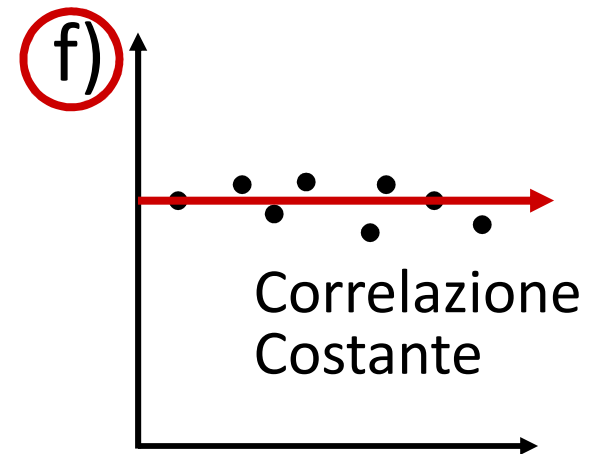
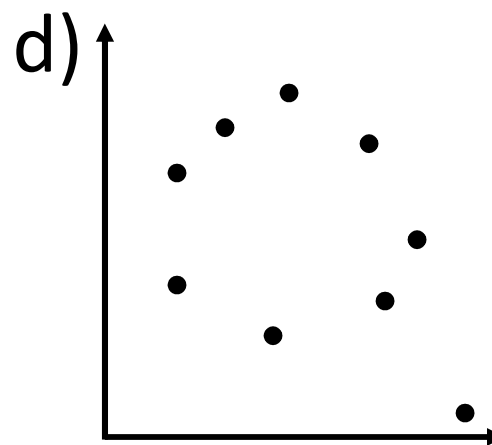
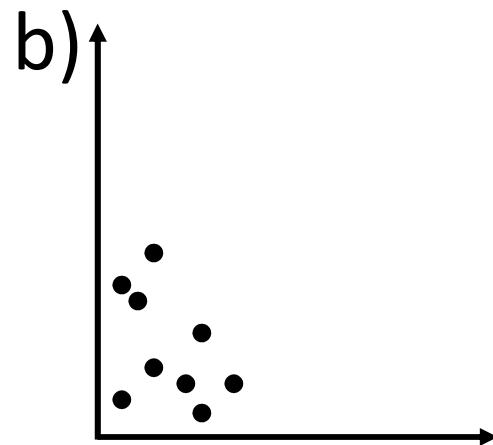
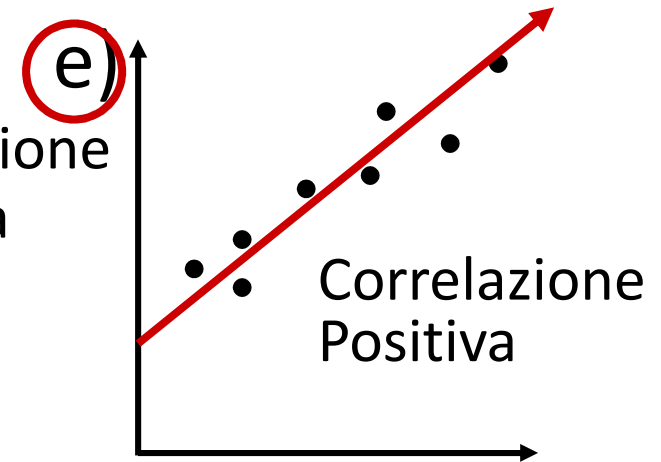
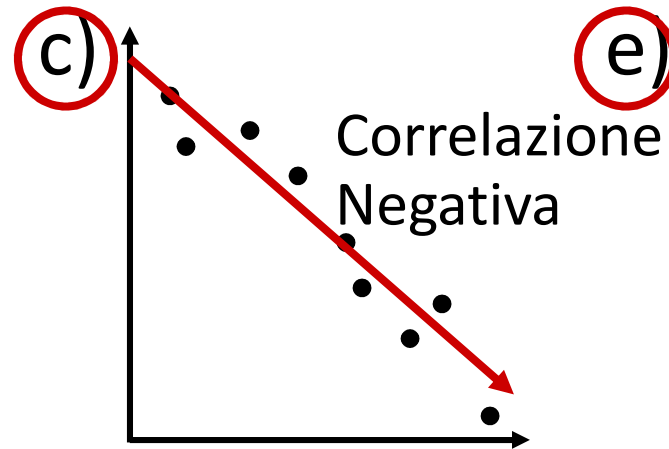
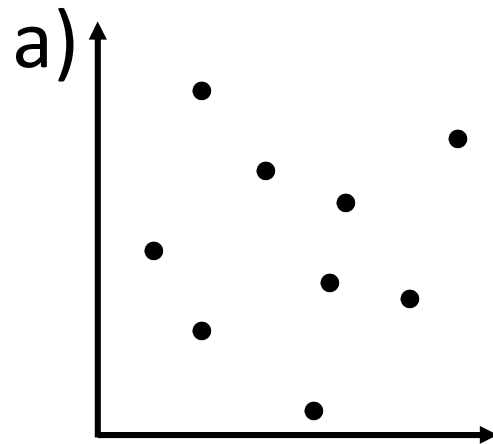
```
ggplot(mydata, aes(Salary)) +  
geom_histogram(binwidth=10000, fill="dodgerblue3",  
color="steelblue4", alpha=.8, size=.25)
```



Scatter Plot

- ▶ Uno scatter plot (diagramma a dispersione) rappresenta una collezione di coppie (x,y) rappresentate come punti su un asse cartesiano
 - ▶ Questo tipo di diagramma viene spesso utilizzato per
 - ▶ Visualizzare il grado di correlazione (la dipendenza lineare) tra due variabili
 - ▶ Uno scatter plot può suggerire vari tipi di correlazione tra variabili con un certo intervallo di confidenza
 - ▶ Correlazione positiva
 - ▶ Correlazione negativa
 - ▶ Correlazione nulla
-

Scatter Plot



Scatter Plot: lessR

- ▶ Con lessR gli scatter plot possono essere definiti semplicemente come combinazione sia di variabili numeriche (continue) e sia di variabili categoriche
- ▶ Si può utilizzare la funzione Plot

Plot(Years, Salary, fit=TRUE)



Scatter Plot: ggplot2

- ▶ Con ggplot gli scatter plot possono essere definiti semplicemente come combinazione sia di variabili numeriche (continue) e sia di variabili categoriche

- ▶ Si può utilizzare un Layer con l'oggetto `geom_point()`

```
ggplot(mydata, aes(Years, Salary)) + geom_point()
```

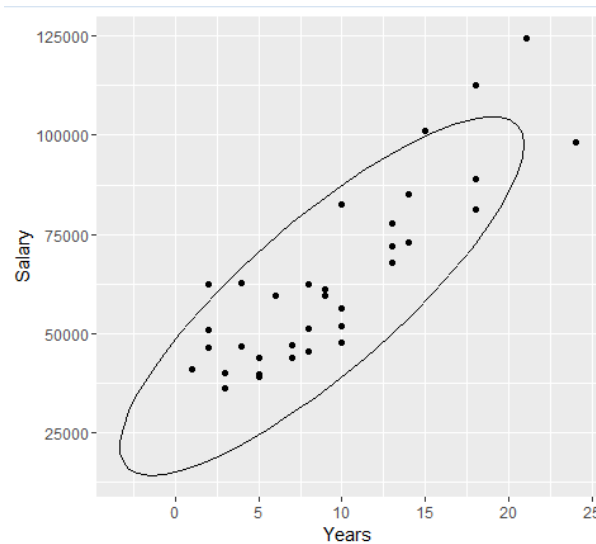
- ▶ A questo punto vediamo come si può
 - ▶ Aggiungere un'ellissi per analizzare il grado di correlazione tra le variabili
 - ▶ Aggiungere la linea di tendenza
-

Scatter Plot: ggplot2

- ▶ Per aggiungere un'ellissi memorizziamo il grafico nella variabile p e aggiungiamo un nuovo Layer

```
p <- ggplot(mydata, aes(Years, Salary)) + geom_point() +  
  stat_ellipse()
```

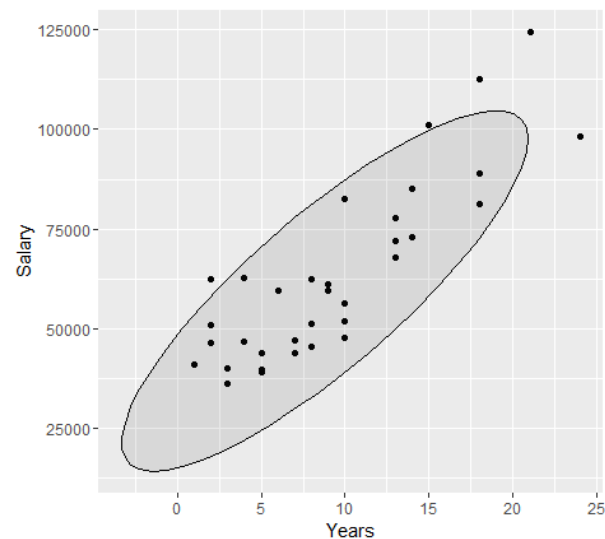
p



Scatter Plot: ggplot2

- ▶ Vediamo come settare alcuni parametri
 - ▶ Riempimento dell'ellisse e ampiezza dei bordi

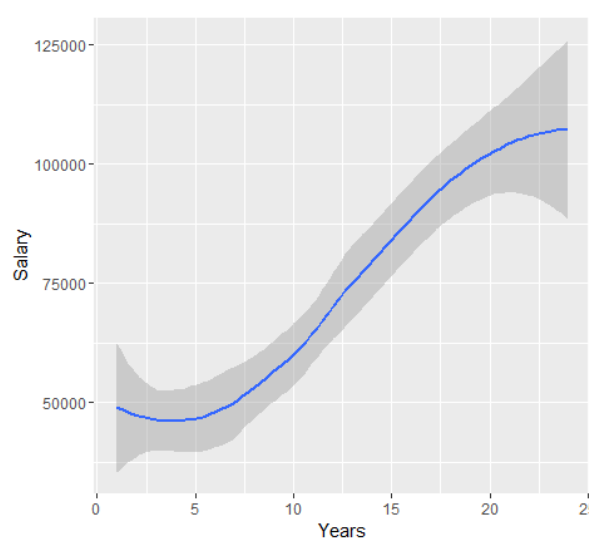
```
ggplot(mydata, aes(Years, Salary)) + geom_point() +  
  stat_ellipse(geom="polygon", color="black", size=.5,  
    alpha=.1)
```



Scatter Plot: ggplot2

- ▶ Per aggiungere la linea di tendenza aggiungiamo un nuovo Layer

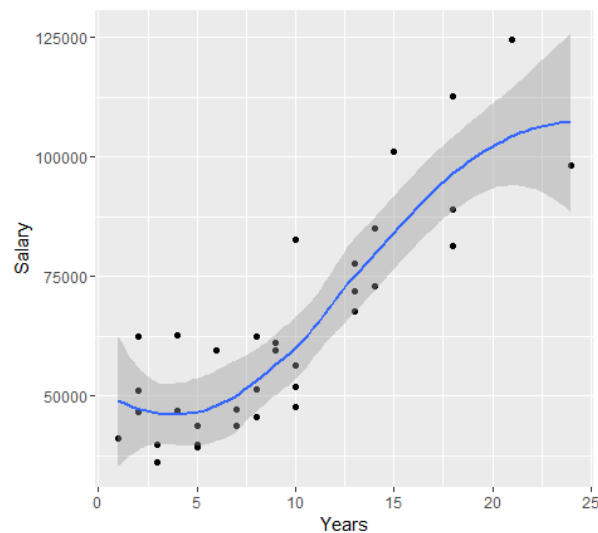
```
ggplot(mydata, aes(Years, Salary)) + geom_smooth()
```



Scatter Plot: ggplot2

- ▶ Naturalmente si possono aggiungere i punti dello scatter plot

```
ggplot(mydata, aes(Years, Salary)) + geom_point() +  
  geom_smooth()
```



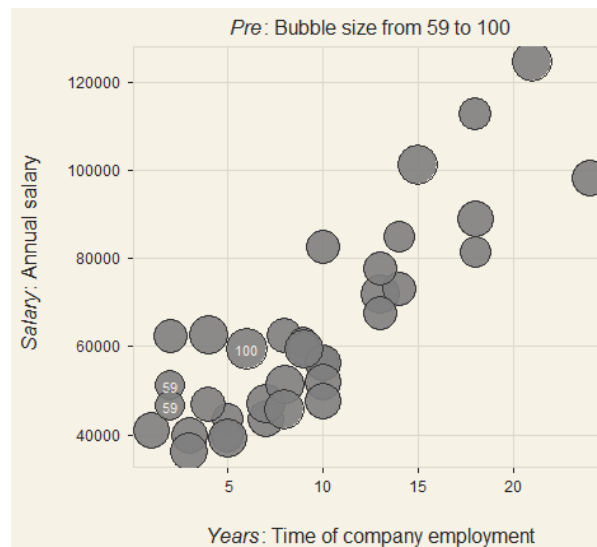
Bubble Plot

- ▶ Uno bubble plot (diagramma a bolle) rappresenta una collezione di triple di valori rappresentate come bolle su un asse cartesiano
 - ▶ La terza dimensione è visualizzata attraverso l'ampiezza delle bolle
 - ▶ Le entità rappresentate possono essere confrontate tra loro in base alla loro dimensione e alla loro posizione rispetto agli assi numerici
 - ▶ Un bubble plot può essere considerato come una variazione di uno scatter plot
 - ▶ Viene utilizzato soprattutto quando devono essere rappresentate più di due serie di valori
-

Bubble Plot: lessR

- ▶ Con lessR i bubble plot possono essere definiti semplicemente
- ▶ Si può utilizzare la funzione Plot

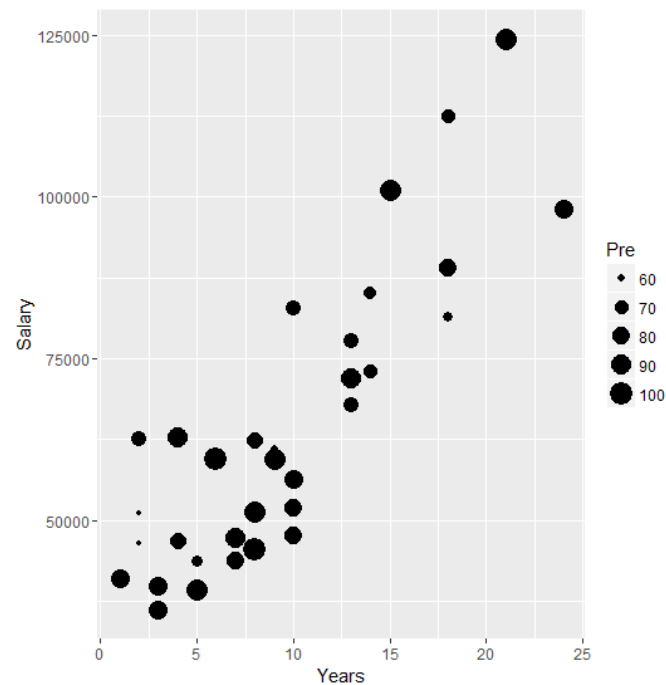
Plot (Years, Salary, size=Pre, radius=0.17)



Bubble Plot: ggplot2

- ▶ Con ggplot2 i bubble plot possono essere definiti semplicemente

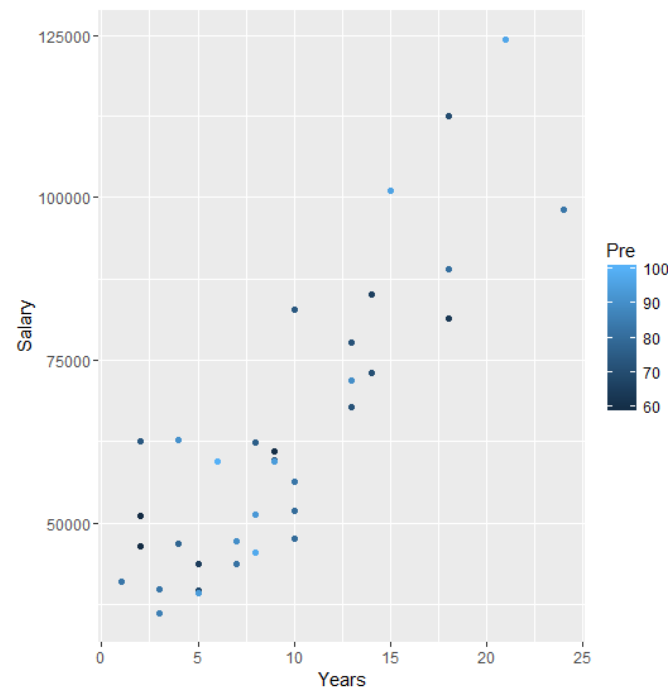
```
ggplot(mydata, aes(Years, Salary)) +  
  geom_point(aes(size=Pre))
```



Bubble Plot: ggplot2

- ▶ Con ggplot2 oltre a poter creare una scala sulla dimensione dei punti, si può creare una scala sui colori

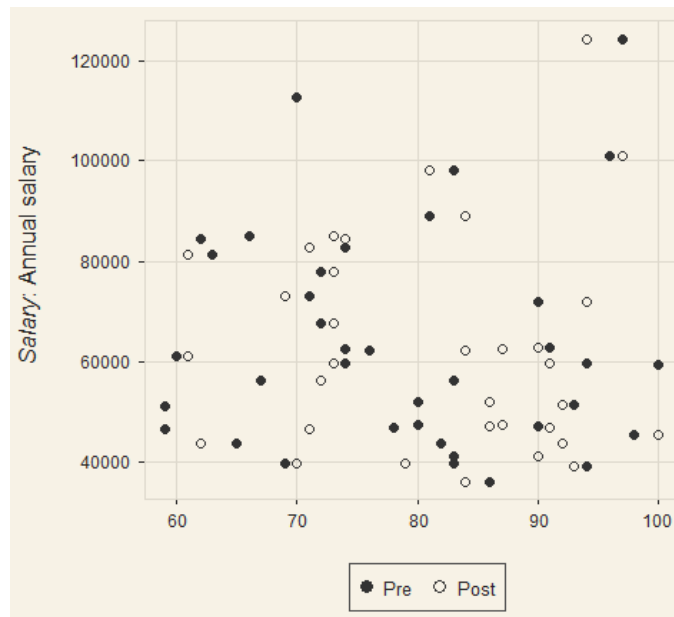
```
ggplot(mydata, aes(Years, Salary)) +  
  geom_point(aes(color=Pre))
```



Grafici Multipli: lessR

- ▶ È possibile costruire grafici multipli con lessR
 - ▶ Basta specificare o la variabile x o la variabile y come vettore

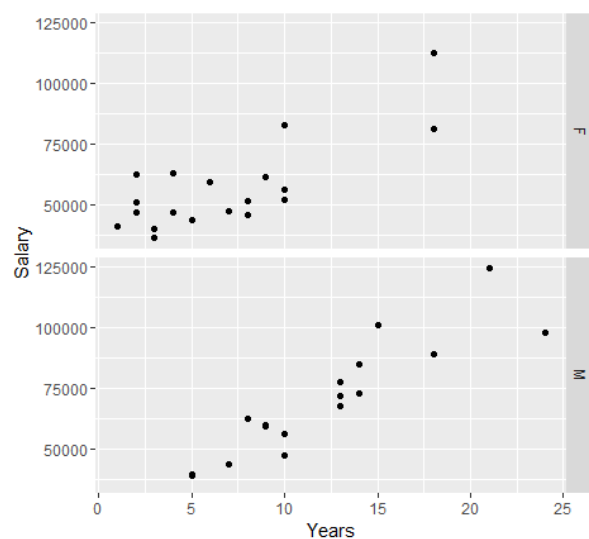
Plot(c(Pre, Post), Salary)



Grafici Multipli: ggplot2

- ▶ È possibile costruire grafici multipli con ggplot2
 - ▶ Separando il grafico sulla base dell'attributo Gender
 - ▶ Dovrà essere usato un nuovo Layer di tipo facet

```
ggplot(mydata, aes(Years, Salary)) + geom_point() + facet_grid(Gender ~ .)
```



Grafici Avanzati: ggplot2

- ▶ Vediamo come utilizzare i Layer di ggplot2 per costruire grafici avanzati
 - ▶ Vediamo un esempio di applicazione nella costruzione di mappe visuali
- ▶ Prima di tutto richiamiamo la libreria

```
library(tidyverse)
```

- ▶ Installiamo e carichiamo un nuovo package: “maps”
 - ▶ Permette di utilizzare le coordinate di latitudine e longitudine degli stati

```
library(maps)
```

Grafici Avanzati: ggplot2

- ▶ Caricamento i dati

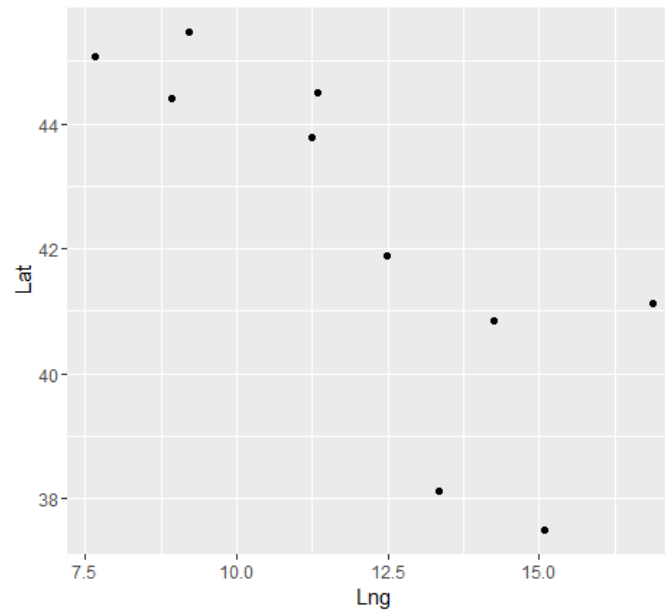
- ▶ Questa volta utilizzando la funzione read.csv

```
mydata <- read.csv(header=TRUE, text="
  City,Lat,Lng,Pop
  Rome,41.895956,12.483258,2318895
  Milan,45.469975,9.205009,1236837
  Naples,40.840025,14.245011,959470
  Turin,45.070387,7.669960,870456
  Palermo,38.125023,13.350027,648260
  Genoa,44.409988,8.930039,580223
  Bologna,44.500422,11.340021,366133
  Florence,43.780001,11.250000,349296
  Catania,37.499971,15.079999,290927
  Bari,41.114220,16.872758,277387")
```

Grafici Avanzati: ggplot2

- ▶ Costruiamo uno scatter plot
 - ▶ Usiamo gli attributi Lng e Lat

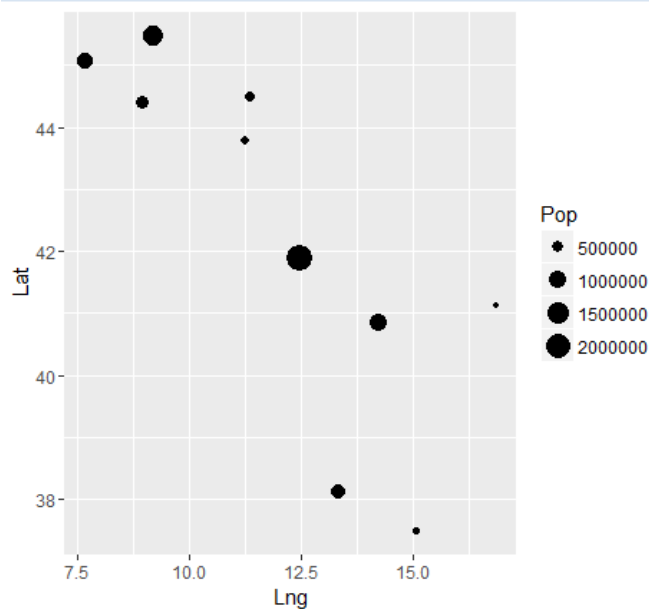
```
ggplot(mydata, aes(Lng, Lat)) + geom_point()
```



Grafici Avanzati: ggplot2

- Costruiamo un bubble plot
 - La taglia delle bolle dipenderà dalla popolazione

```
ggplot(mydata, aes(Lng, Lat)) + geom_point(aes(size=Pop))
```



Grafici Avanzati: ggplot2

- ▶ Vogliamo utilizzare i dati delle mappe
 - ▶ Usiamo la funzione “map_data”

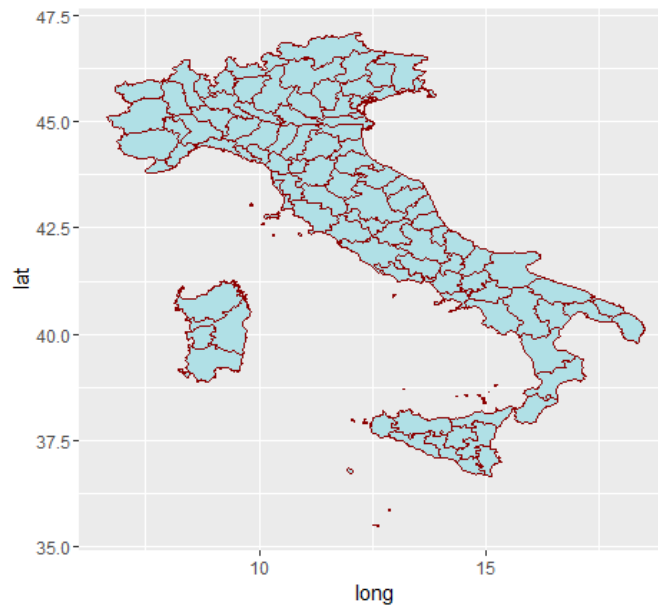
```
Italy <- map_data("italy")  
head(Italy)
```

```
      long      lat group order      region subregion  
1 11.83295 46.50011     1     1 Bolzano-Bozen      <NA>  
2 11.81089 46.52784     1     2 Bolzano-Bozen      <NA>  
3 11.73068 46.51890     1     3 Bolzano-Bozen      <NA>  
4 11.69115 46.52257     1     4 Bolzano-Bozen      <NA>  
5 11.65041 46.50721     1     5 Bolzano-Bozen      <NA>  
6 11.63282 46.48045     1     6 Bolzano-Bozen      <NA>
```

Grafici Avanzati: ggplot2

- ▶ Visualizziamo la mappa dell'Italia
 - ▶ Usiamo il Layer "geom_polygon"

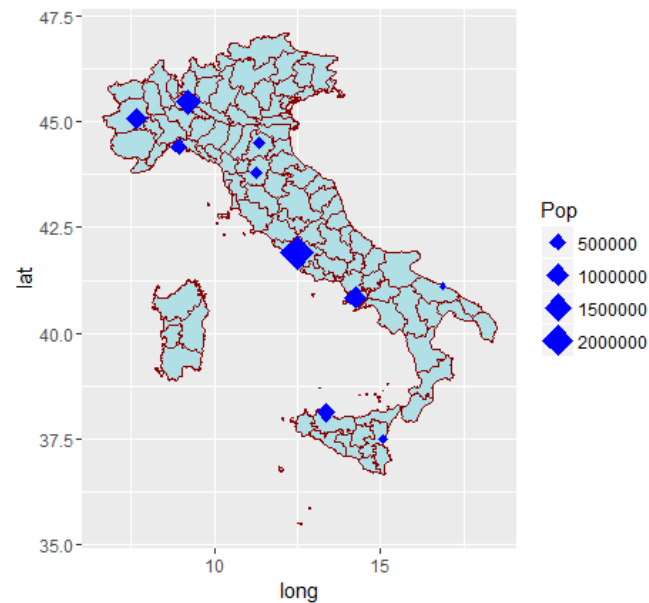
```
ggplot() +  
geom_polygon(data=Italy, aes(long, lat, group=group),  
color="darkred", fill="powderblue")
```



Grafici Avanzati: ggplot2

- Aggiungiamo un Layer per inserire i dati sulla popolazione

```
ggplot() +  
  geom_polygon(data=Italy, aes(long, lat, group=group),  
    color="darkred", fill="powderblue") +  
  geom_point(data=mydata, aes(Lng, Lat, size=Pop),  
    color="blue", fill="blue", shape=23)
```

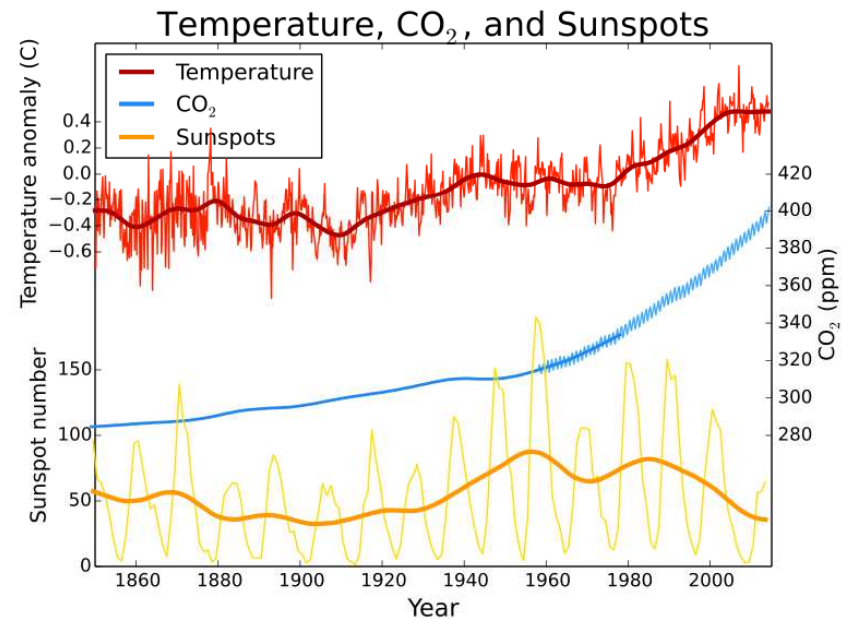


...Look At

- ▶ Esistono molti altri tool o librerie per effettuare una buona visualizzazione dei dati

- ▶ **matplotlib**

- ▶ Una libreria Python
 - ▶ Grafici interattivi



...Look At

- ▶ Esistono molti altri tool o librerie per effettuare una buona visualizzazione dei dati

- ▶ **d3.js**

- ▶ Una libreria Java Script
- ▶ Permette di creare grafici in
 - ▶ HTML
 - ▶ CSS
 - ▶ SVG
- ▶ Grafici interattivi

