Data Analytics

Ivan Santagati

17 giugno 2025

Indice

1 Fondamenti di Statistica

La statistica rappresenta il fondamento dell'analisi dei dati. Per sviluppare competenze solide in Data Analytics, è necessario conoscere i concetti fondamentali legati a dati, popolazioni, campioni e frequenze.

1.1 Tipi di Dati

I dati analizzati solitamente rientrano in queste categorie:

- **Quantitativi**: numerici (es. altezza, reddito)
- **Qualitativi**: categorie (es. genere, regione)
- **Discreti**: valori interi e numerabili (es. numero figli)
- **Continui**: valori reali su un intervallo (es. tempo di risposta)

1.2 Concetti Chiave

Nel contesto statistico, si distinguono alcuni concetti essenziali:

- **Popolazione**: insieme completo degli individui osservabili
- **Campione**: sottoinsieme rappresentativo della popolazione
- **Variabile**: caratteristica misurata su ciascun individuo
- **Frequenza assoluta e relativa**: conteggio e proporzione dei valori

1.3 Esempio

Dato il seguente insieme di voti:

$$x = \{18, 21, 24, 24, 27, 30, 30, 30, 30, 30\}$$

Il valore 30 si presenta 5 volte. Quindi:

- Frequenza assoluta: 5
- Frequenza relativa: $\frac{5}{10} = 0.5$

2 Statistica Descrittiva con R

2.1 Misure di Tendenza Centrale

Per identificare il centro di un insieme di dati, si utilizzano:

- **Media**: media aritmetica dei valori
- **Mediana**: valore centrale in un insieme ordinato
- **Moda**: valore più frequente

2.2 Misure di Dispersione

Per misurare la variabilità dei dati, si considerano:

- **Range**: differenza tra valore massimo e minimo
- **Varianza**: media dei quadrati degli scarti dalla media
- **Deviazione standard**: radice quadrata della varianza

2.3 Analisi in R

Con un vettore di dati, i calcoli si effettuano in questo modo:

```
x <- c(18, 21, 24, 24, 27, 30, 30, 30, 30, 30)
mean(x)          # media
median(x)        # mediana
var(x)           # varianza
sd(x)            # deviazione standard
```

Nota: Quando sono presenti valori estremi, la mediana può fornire una misura più robusta rispetto alla media.

3 Visualizzazione dei Dati

3.1 Introduzione

La visualizzazione è uno strumento fondamentale per interpretare rapidamente la struttura dei dati. Permette di rilevare tendenze, gruppi, anomalie e relazioni tra variabili.

3.2 Principali Tipi di Grafici

- **Istogramma**: mostra la distribuzione di una variabile continua
- **Boxplot**: evidenzia valori centrali e outlier
- **Barplot**: rappresenta frequenze di categorie
- **Scatterplot**: evidenzia relazioni tra due variabili quantitative

3.3 Esempi in R

```
x <- rnorm(100)
y <- x + rnorm(100, sd = 0.5)

hist(x)                          # istogramma
boxplot(x)                       # boxplot
barplot(table(cut(x, 5)))        # barplot su intervalli
plot(x, y)                       # scatterplot
```

Oss: Una buona pratica consiste nel visualizzare i dati prima di applicare tecniche statistiche: spesso, un semplice grafico evidenzia aspetti non rilevabili da sole misure numeriche.

4 Concetti base

- Spazio campionario: insieme di tutti i risultati possibili
- Evento: sottoinsieme dello spazio campionario
- Probabilità: numero tra 0 e 1 associato all'evento
- Eventi indipendenti e mutualmente esclusivi

4.1 Regole fondamentali

$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)}$$

Esempio pratico:
Lancio di un dado:

$$P(\text{esce un numero pari}) = \frac{3}{6} = 0.5$$

5 Distribuzioni Discrete

Le distribuzioni di probabilità discrete si utilizzano per descrivere variabili aleatorie che possono assumere un numero finito o numerabile di valori distinti. Sono fondamentali per modellare situazioni in cui contiamo eventi (es. successi, errori, arrivi).

5.1 Variabile aleatoria discreta

Una variabile aleatoria discreta può assumere valori interi, ognuno con una certa probabilità:

$$\sum_i P(X = x_i) = 1$$

5.2 Distribuzione di Bernoulli

È la più semplice: una sola prova con due esiti (successo o insuccesso).

- Esiti: 1 (successo), 0 (fallimento)
- Parametro: $p = P(\text{successo})$
- Valori attesi: $\mathbb{E}(X) = p$, $\text{Var}(X) = p(1-p)$

Esempio in R:

```
# 10 prove di Bernoulli con p = 0.3
rbinom(10, size = 1, prob = 0.3)
```

5.3 Distribuzione Binomiale

Conta il numero di successi in n prove indipendenti di Bernoulli.

$$P(X = k) = \binom{n}{k} p^k (1-p)^{n-k}$$

- Parametri: n, p
- Media: np, Varianza: $np(1-p)$
- Appropriata quando ogni prova ha solo due esiti e probabilità costante.

Esempio in R:

```
# Distribuzione di probabilit  binomiale per n = 10 e p = 0.5
dbinom(0:10, size = 10, prob = 0.5)
```

5.4 Distribuzione di Poisson

Modella il numero di eventi che accadono in un intervallo (tempo, spazio, ecc.) se questi sono rari e indipendenti.

$$P(X = k) = \frac{\lambda^k e^{-\lambda}}{k!}$$

- Parametro: λ (media degli eventi per intervallo)
- Media e varianza: λ
- Utile per contare arrivi o errori (es. richieste a un server)

Esempio in R:

```
# Genera 100 valori dalla distribuzione di Poisson con media 4
rpois(100, lambda = 4)
```

5.5 Confronto Binomiale e Poisson

La distribuzione di Poisson può essere vista come il limite della binomiale per $n \to \infty$, $p \to 0$, con $np = \lambda$ costante.

Nota: Le distribuzioni discrete sono alla base di molti modelli inferenziali, in particolare nei contesti di conteggio, errori o classificazione binaria. Comprendere le loro proprietà aiuta ad applicarle con criterio nei modelli statistici o di machine learning.

6 Distribuzioni Continue

Le distribuzioni continue descrivono variabili aleatorie che possono assumere un numero infinito di valori reali all'interno di un intervallo. A differenza delle distribuzioni discrete, qui la probabilità che la variabile assuma un valore esatto è sempre pari a zero: si lavora con intervalli.

6.1 Funzione di densità di probabilità

Per una variabile continua X, la probabilità che cada in un intervallo $[a, b]$ si ottiene integrando la funzione di densità $f(x)$:

$$P(a \leq X \leq b) = \int_a^b f(x)\, dx$$

6.2 Distribuzione Normale

La distribuzione normale (o gaussiana) è una delle più importanti in statistica per il suo ruolo centrale nel Teorema del Limite Centrale.

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

- Parametri: μ (media), σ (deviazione standard)
- Simmetrica, a campana, media = mediana = moda
- Circa il 68% dei dati cade in $[\mu - \sigma, \mu + \sigma]$

Esempio in R:

```
x <- rnorm(1000, mean = 0, sd = 1)
hist(x, probability = TRUE)
curve(dnorm(x), add = TRUE, col = "blue")
```

6.3 Distribuzione Uniforme

La distribuzione uniforme continua assegna la stessa densità a tutti i valori in un intervallo.

$$f(x) = \frac{1}{b-a}, \quad x \in [a, b]$$

Esempio in R:

```
x <- runif(1000, min = 0, max = 1)
hist(x, probability = TRUE)
```

6.4 Funzione di distribuzione cumulativa (CDF)

La funzione di distribuzione cumulativa $F(x)$ rappresenta la probabilità che la variabile aleatoria assuma un valore minore o uguale a x.

Esempio in R:

```
pnorm(1.96)   # probabilit che una N(0,1) sia <= 1.96
```

Nota: Le distribuzioni continue sono fondamentali per i modelli inferenziali e predittivi. La distribuzione normale, in particolare, è la base per moltissimi test statistici e modelli lineari.

7 Introduzione a R

R è un linguaggio di programmazione open-source pensato per l'analisi statistica, la manipolazione dei dati e la visualizzazione grafica. È ampiamente usato da statistici, analisti e data scientist.

7.1 Concetti chiave

Quando si lavora con R, è importante comprendere questi elementi base:

- **Oggetti**: qualsiasi valore in R è memorizzato in un oggetto (es. vettori, liste, matrici, dataframe)
- **Assegnazione**: si usa `<-` per assegnare un valore a un oggetto
- **Funzioni**: operazioni o strumenti predefiniti che elaborano dati (es. `mean()`, `sum()`, `length()`)

7.2 Tipi di dati

I principali tipi di dato in R sono:

- `numeric` – numeri reali o decimali
- `integer` – numeri interi
- `character` – stringhe di testo
- `logical` – valori booleani: `TRUE`, `FALSE`
- `factor` – variabili categoriche

7.3 Strutture dati

- **Vettori**: una sequenza di elementi dello stesso tipo
- **Matrici**: tabelle bidimensionali con dati omogenei
- **Liste**: contenitori eterogenei
- **Data frame**: simili a tabelle con colonne di tipi diversi

7.4 Creazione oggetti in R

```r
# Vettore
v <- c(1, 2, 3)

# Matrice 2x3
m <- matrix(1:6, nrow = 2)

# Data frame
df <- data.frame(nome = c("A", "B"), eta = c(22, 25))
```

```r
# Lista
l <- list(v, m, df)
```

7.5 Comandi utili

- `class(x)` – tipo di oggetto
- `str(x)` – struttura interna dell'oggetto
- `summary(x)` – sintesi statistica
- `length(x)` – lunghezza vettore/lista
- `names(df)` – nomi delle colonne in un dataframe

7.6 Nota

R è particolarmente potente perché unisce capacità statistiche, funzioni grafiche e flessibilità nella manipolazione dei dati. Familiarizzare con i tipi di oggetti e i comandi base è il primo passo per usarlo in modo efficace.

8 Importazione e Manipolazione Dati

Per lavorare sui dati, è fondamentale saperli importare, visualizzare e trasformare. R offre funzioni efficienti per leggere dataset da file, esplorarli e modificarli.

8.1 Importazione dati

Il metodo più comune per importare un file CSV:

```r
# Lettura da file CSV
dati <- read.csv("file.csv", header = TRUE)
```

- `header = TRUE` indica che la prima riga contiene i nomi delle colonne
- Altri formati comuni: `read.table()`, `read.delim()`

8.2 Esplorazione dati

Per avere un'idea iniziale della struttura dei dati uso:

```r
head(dati)      # Prime righe
str(dati)       # Struttura delle colonne
summary(dati)   # Statistiche riassuntive
names(dati)     # Nomi delle colonne
```

8.3 Manipolazione di base

Posso filtrare, creare colonne, modificare elementi:

```
# Accesso a una colonna
dati$eta'

# Filtraggio righe
dati[dati$eta' > 25, ]

# Nuova colonna calcolata
dati$et 2 <- dati$et  * 2
```

Nota: Imparare a manipolare bene un dataframe è fondamentale per ogni analisi. È qui che inizia la vera pulizia e trasformazione dei dati.

9 Funzioni, Medie e Grafici in R

Una parte importante dell'analisi dei dati consiste nel riassumere informazioni con funzioni statistiche e rappresentarle con grafici chiari.

9.1 Funzioni statistiche di base

```
x <- c(10, 12, 15, 20)

mean(x)     # media
median(x)   # mediana
var(x)      # varianza
sd(x)       # deviazione standard
summary(x)  # riassunto statistico
```

9.2 Grafici semplici

```
# Istogramma
hist(x)

# Boxplot
boxplot(x)

# Scatterplot
plot(x, x + rnorm(4))
```

9.3 Personalizzazione dei grafici

```
hist(x, col = "skyblue", main = "Istogramma dei valori", xlab = "Valori")
```

Nota: Grafici ben costruiti sono strumenti potentissimi per spiegare i dati, anche a chi non ha una formazione tecnica.

10 Variabili Aleatorie in R

Le variabili aleatorie permettono di simulare comportamenti casuali attraverso distribuzioni note. In R, è possibile lavorare sia con distribuzioni teoriche sia con simulazioni pratiche.

10.1 Simulare distribuzioni discrete

```
# Binomiale: 100 prove, n=10, p=0.3
rbinom(100, size = 10, prob = 0.3)

# Poisson: lambda = 4
rpois(100, lambda = 4)
```

10.2 Simulare distribuzioni continue

```
# Normale standard
rnorm(1000, mean = 0, sd = 1)

# Uniforme continua tra 0 e 1
runif(100, min = 0, max = 1)
```

10.3 Visualizzazione dei risultati

```
x <- rnorm(1000)
hist(x, probability = TRUE)
curve(dnorm(x), add = TRUE, col = "red")
```

Nota: Le funzioni `r*`, `d*`, `p*`, `q*` permettono di simulare, calcolare densità, probabilità e quantili per ogni distribuzione supportata da R.

11 Regressione Lineare

La regressione lineare è una tecnica statistica utilizzata per modellare la relazione tra una variabile dipendente y e una o più variabili indipendenti x.

11.1 Regressione lineare semplice

Modello:

$$y = \beta_0 + \beta_1 x + \varepsilon$$

Dove:

- β_0: intercetta
- β_1: coefficiente di regressione
- ε: errore casuale

11.2 Esempio in R

```
x <- c(1, 2, 3, 4, 5)
y <- c(2, 4, 5, 4, 5)

modello <- lm(y ~ x)
summary(modello)

plot(x, y)
abline(modello, col = "blue")
```

12 Statistica Inferenziale

La statistica inferenziale permette di trarre conclusioni su una popolazione a partire da un campione.

12.1 Intervalli di confidenza

Un intervallo di confidenza fornisce un range di valori entro cui è probabile che si trovi il parametro della popolazione.

```
x <- c(24, 25, 23, 26, 22)
t.test(x)  # intervallo per la media
```

12.2 Test di ipotesi

Esempio di test su medie:

```
# Test t per media
t.test(x, mu = 25)

# Test per varianza
var.test(x, y)
```

13 Introduzione al tidyverse

Il pacchetto `tidyverse` raccoglie strumenti moderni per manipolare e visualizzare dati in R in modo efficiente e leggibile.

13.1 Funzioni chiave di `dplyr`

- `filter()`: filtra righe
- `select()`: seleziona colonne
- `mutate()`: crea nuove variabili
- `summarise()`: riassume variabili

- `group_by()`: raggruppa dati

```
library(dplyr)

dati %>%
  filter(et   > 30) %>%
  mutate(gruppo = ifelse(et   > 50, "senior", "adulto")) %>%
  group_by(gruppo) %>%
  summarise(media_voto = mean(voto))
```

13.2 Visualizzazione con `ggplot2`

```
library(ggplot2)

ggplot(dati, aes(x = et  , y = voto)) +
  geom_point() +
  geom_smooth(method = "lm")
```

14 PCA e Analisi Multivariata

L'analisi multivariata consente di esaminare più variabili contemporaneamente. PCA e clustering sono due strumenti potenti per esplorare strutture latenti.

14.1 PCA - Principal Component Analysis

Riduce la dimensionalità dei dati mantenendo la varianza più rilevante.

```
x <- scale(iris[, 1:4])
pca <- prcomp(x)
summary(pca)
biplot(pca)
```

14.2 Clustering con k-means

```
set.seed(1)
km <- kmeans(x, centers = 3)
plot(x, col = km$cluster)
```

15 Statistica Inferenziale

La statistica inferenziale permette di trarre conclusioni su una popolazione a partire da un campione.

15.1 Intervalli di confidenza

Un intervallo di confidenza fornisce un range di valori entro cui è probabile che si trovi il parametro della popolazione.

```
x <- c(24, 25, 23, 26, 22)
t.test(x)  # intervallo per la media
```

15.2 Test di ipotesi

Esempio di test su medie:

```
# Test t per media
t.test(x, mu = 25)

# Test per varianza
var.test(x, y)
```

16 Introduzione al tidyverse

Il pacchetto `tidyverse` raccoglie strumenti moderni per manipolare e visualizzare dati in R in modo efficiente e leggibile.

16.1 Funzioni chiave di `dplyr`

- `filter()`: filtra righe
- `select()`: seleziona colonne
- `mutate()`: crea nuove variabili
- `summarise()`: riassume variabili
- `group_by()`: raggruppa dati

```
library(dplyr)

dati %>%
  filter(et  > 30) %>%
  mutate(gruppo = ifelse(et  > 50, "senior", "adulto")) %>%
  group_by(gruppo) %>%
  summarise(media_voto = mean(voto))
```

16.2 Visualizzazione con `ggplot2`

```
library(ggplot2)

ggplot(dati, aes(x = et  , y = voto)) +
  geom_point() +
  geom_smooth(method = "lm")
```

17 PCA e Analisi Multivariata

L'analisi multivariata consente di esaminare più variabili contemporaneamente. PCA e clustering sono due strumenti potenti per esplorare strutture latenti.

17.1 PCA - Principal Component Analysis

Riduce la dimensionalità dei dati mantenendo la varianza più rilevante.

```
x <- scale(iris[, 1:4])
pca <- prcomp(x)
summary(pca)
biplot(pca)
```

17.2 Clustering con k-means

```
set.seed(1)
km <- kmeans(x, centers = 3)
plot(x, col = km$cluster)
```