

Formulario Matlab

Probabilità e Statistica

Chiara Solito - Bioinformatica

II Semestre - A.A. 2020/2021

1 Basi

- %
Non compila e non esegue il testo (commenti)
- Mostrare qualcosa a display (output)
`disp()`
- Vettore
`a1 = [2 5 1]; %riga`
- Grafico a Torta
`a1 = [2; 5; 1]; %colonna`
- Appendere i vettori
`u = [r,w]`
- Operatore colonna
Genera vettori spaziati regolarmente
`u = m:q:n`
- Operatore linspace
Genera vettori riga spaziati linearmente
`linspace(x1,x2,n)`
- Matrice
- `a1 = [a b c; d e f; g h i]`
$$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$$
- Accedere alle matrici
`A(:,3)` denotes all the elements in the 3rd column of the matrix A
`A(:,2:5)` denotes all the elements in the 2nd through 5th columns of A
`A(2:3,1:3)` denotes all the elements in the 2nd and 3rd rows that are also in the 1st through 3rd columns
`v = A(:)` creates a vector v consisting of all the columns of A stacked from first to last
`A(end,:)` denotes the last row in A
`A(:,end)` denotes the last column in A

- Stringa
`month = 'August'`
- Cella
`a = cell(3,3)`
- Struttura
`student.year = 2; %come il c`

2 Funzioni per matrici

- Concatenare matrici
`B = [A A+32; A+48 A+16]`
- Creare matrici dai vettori
`c = [a b]; o c = [a; b];`
- Cancellare righe e colonne
`X = magic(4) e X(:,3)=[]`
- Creare matrici di zeri
`C = zeros(2,2)`
- Creare matrici di 1
`C = ones()`
- Creare matrici di numeri random
`C = rand()`
- Trovare un elemento diverso da zero
`[k,j] = find(A)`
- Trova la lunghezza (degli array) o il massimo tra altezza e lunghezza di una matrice
`length(A)`
- Trova il massimo
`max(A)`
- Trova il minimo
`min(A)`
- Trova la taglia
`size(A)`
- Somma gli elementi di ogni colonna
`sum(A)`

3 Operaioni per array e matrici

- Addizione
`+`
- Sottrazione
`-`
- Moltiplicazione elemento per elemento
`.*`
- Divisione elemento per elemento
`./`
- Divisione sinistra elemento per elemento
- Potenza elemento per elemento
`.^`
- Trasposta non coniugata di un array
`.'`

4 Grafi e Tavole di Frequenza

- Grafico a Bastoncini
`stem()`
- Grafico a Linee
`plot()`
- Grafico a Barre
`bar()`
- Grafico a Barre Orizzontali
`barh()`
- Grafico a Torta
`pie()` e `pie3()`
- Boxplot
`boxplot()`
- Subplot?
`subplot(A)`

5 Basic Plotting

```
x = 0:pi/100:2*pi;  
y = sin(x);  
plot(x,y)  
xlabel('x = 0:2 ylabel('sin(x)')  
title('Plot of the Sine Function')
```

6 Statistica Descrittiva - Funzioni

- Frequenza
`hist()`
- Tabella delle frequenze
`tabulate()`
- Media
`mean() %= sum(A)/n`
- Mediana
`median() %50th percentile`
- Moda
`mode() %most frequent value`
- Deviazione Standard
`std()`
- Varianza
`var()`
- Skewness
`skewness()`
- Kurtosi
`kurtosis`
- `(nanmean() %Mean, ignoring NaN values)`
- Range
`range()`
- Massimo
`max()`
- Minimo
`min()`
- Interquartile range
`iqr()`
- Percentile del data set
`prctile()`
- Quantile del data set
`quantile()`
- Coefficiente di correlazione
`corrcoef()`
- Grafico di Dispersione per gruppo
`gscatter()`

- Istogramma bivariato
`hist3()`
- Grafico di Dispersione con istogrammi marginali
`scatterhist()`

7 I Cicli

- FOR

```
for loopvar = range
    action
end
```

8 Elementi di Probabilità

- Permutazione
`perms()`
- Coefficiente Binomiale
`nchoosek()`

9 Distribuzioni di Probabilità

- Funzione di densità di probabilità
`pdf()`
- Funzione di distribuzione cumulativa
`cdf()`
- Inversa di cdf
`inv()`
- Generazione di numeri casuali
`rnd()`
- Generazione di numeri casuali uniformemente distribuito sull'intervallo (0,1)
`unifrnd()`
- Media e Varianza
`stat()`
- Testing di ipotesi
`test()`
- Funzione di errore
`erf()`

- Stima di Massima Verosimiglianza

```
erf()phat = mle(x,'distribution','dist')
```

returns the maximum likelihood estimate of the parameters that is assumed to be originated from the specified distribution by 'dist'.

```
phat = mle(means)
```

Distribution of means from repeated random samples of an exponential distribution

| Distribuzione | PDF | CDF |
|-----------------|------------|------------|
| Normale | normpdf() | normcdf() |
| Uniforme (cont) | unifpdf() | unifcdf() |
| Essponenziale | exppdf() | expcdf() |
| Chi-square | chi2pdf() | chi2cdf() |
| F | fpdf() | fcdf() |
| t | tpdf() | tcdf() |
| Binomiale | binopdf() | binocdf() |
| Poisson | poisspdf() | poisscdf() |
| Ipergeometrica | hygepdf() | hygecdf() |

10 Esercizi

Costruire un intervallo di confidenza al 95% per μ , l'intensità effettiva. Determinare, per gli stessi dati, un intervallo di confidenza al 99% per μ .

```
data = [17 21 20 18 19 22 20 21 26 29];
N = length(data);
sigma = 3;
alpha1 = 0.05;
z1 = norminv(1-alpha1/2);
x_bar = mean(data);
```

%calcolo del primo intervallo di confidenza

```
ConfidenceInterval95 = [x_bar - z1*sigma/sqrt(N) , x_bar + z1*sigma/sqrt(N)];
%[17.44 21.16]
```

%calcolo del secondo intervallo di confidenza

```
alpha2 = 0.01;
z2 = norminv(1-alpha2/2);
ConfidenceInterval99 = [x_bar - z2*sigma/sqrt(N) , x_bar + z2*sigma/sqrt(N)];
%[16.86 21.74]
```

Si suppone che l'intensità del segnale sia uguale a 20. Verificare se questa ipotesi è plausibile. Usare una significativa del 5%,

```
x_true = 20;
[h,p] = ztest(data, x_true, sigma);
% h = 0 lo ztest non rifiuta l'ipotesi nulla
%(H0: mu=20) ad un lvl del 5%
```

Costruire un modello (fit) di tipo lineare che descriva l'andamento degli incassi al botteghino in funzione del numero di weeken

```
%linear fit: least squares regression methos
b = polyfit(x,y,1);
```

Costruire una prima figura che rappresenti un diagramma a dispersione dei dati con sovrapposto il modello di tipo lineare (assegnare correttamente titolo e label alla figura)

%diagramma di dispersione dei dati e visualizzazione del modello lineare

```
figure(1)
scatter(x,y); xlabel('weekend'); ylabel('incassi')
hold on
plot(x, x*b(1)+b(2))
title('Diagramma di dispersione dei dati e retta di regressione (modello lineare y=bx+be)')
hold off
```

Determinare se il modello lineare è appropriato a) calcolando e visualizzando in una seconda figura i residui (assegnare correttamente titolo e label), b) calcolando il coefficiente di determinazione (R^2).

%calcolo dei residui e visualizzazione

```
res = polyval(b,x)-y;
figure(2)
stem(res)
title('Residui')
xlabel('weekend'); ylabel('residui')
```

%calcolo del coefficiente di determinazione

```
SSR = sum(res.^2); %residual sum of square
SYY = sum((y-mean(y)).^2); %devianza: total sum of squares
R2 = 1 -SSR/SYY; %coefficiente di determinazione
```

Calcolare la probabilità che al massimo 3 persone selezionate abbiano un determinato disturbo. $P((X \leq 3))$

```
p = 0.2; %probabilità dello 0.2 che una persona soffra di una malattia
n = 25; %seleziono 25 persone
cdf3 = binocdf(3,n,p); %P(X<=3)
```

Ottenere i valori della funzione di massa di probabilità (probability mass function) per $n=6$, $p = 0.3$, e successivamente per $n = 6$, $p = 0.7$ e visualizzare i risultati nella stessa figura (due subplot) usando la funzione bar. Assegnare correttamente titolo e label a ciascun subplot. *Nota: quando chiede la probability mass function lei dà come risposta la binopdf, la pdf binomiale*

```
x = 0:1:25;
```

```
n1 = 6;
p1 = 0.3;
pmf1 = binopdf(x, n1, p1); % pmf binomiale con parametri (n1,p1)
```

```
n2 = 6;
p2 = 0.7;
pmf2 = binopdf(x, n2, p2); %pmf binomiale con parametri (n2,p2)
```

```
figure(1);
subplot(2,1,1);
bar(pmf1);
title('pmf binomiale con parametri (n=6, p=0.3)')
xlabel('x');
ylabel('P(X=x)');
```

```
figure(1);
subplot(2,1,2);
bar(pmf2);
title('pmf binomiale con parametri (n=6, p=0.7)')
xlabel('x');
ylabel('P(X=x)');
```