

Fondamenti di Informatica, A.A. 2012-2013

12/02/2014

Prova Pratica

È dato un vettore di numeri reali ed un parametro di larghezza. Si chiede di produrre in uscita un istogramma, ossia un conteggio del numero di elementi in ingresso che cade in ciascuno degli intervalli della larghezza specificata, allineati allo 0. La funzione deve produrre in uscita il vettore delle numerosità delle classi ed il vettore con i limiti delle classi stesse; le classi devono essere determinate sulla base dei dati in ingresso. Ad esempio: se vale

```
v =[ 0.31991 -0.57178 0.72788 -0.57194]
```

allora

```
octave> [his ,xv]=histogram(v,0.4)
his =
```

```
2 0 1 1
```

```
xv =
```

```
-0.80000 -0.40000 0.00000 0.40000 0.80000
```

Se invece

```
v =[ 4.4862 6.9867 5.4798 5.8744]
```

allora

```
octave>[his ,xv]=histogram(v)
his =
```

```
1 2 1
```

```
xv =
```

```
4 5 6 7
```

Svolgimento

La soluzione del problema proposto richiede di determinare quali siano gli estremi delle classi che racchiudono i dati in oggetto. A tale proposito occorrerà identificare i valori minimo e massimo tra quelli presenti nell'array in ingresso, e poi calcolare la parte intera inferiore della divisione tra questi valori e la larghezza della classe; in questo modo avremo identificato l'estremo sinistro della prima e dell'ultima classe, con le istruzioni

```
vmin=min(v);  
vmax=max(v);  
  
kmin=floor(vmin/width);  
xmin=kmin*width;  
kmax=floor(vmax/width);  
xmax=kmax*width;
```

A questo punto possiamo calcolare quante classi sono presenti ed i loro estremi; aggiungendo anche l'estremo destro dell'ultima classe, i punti limite sono uno in più del numero delle classi.

Infine per produrre l'istogramma occorre identificare per ciascun elemento v_k del vettore di ingresso in quale classe vada a cadere; un semplice modo è di contare il numero di estremi sinistri delle classi che sono minori o uguali all'elemento stesso

```
kh=sum(xv<=vk);
```

Il codice completo, che rende anche opzionale l'ampiezza delle classi e l'uscita dei loro intervalli, è:

```
function [hist,xval]=histogram(v,wd)  
%  
% Compute a histogram from a vector v.  
% Optional wd: width of classes.  
%  
if (nargin >1)  
    width = abs(wd);  
else  
    width = 1;  
end  
  
vmin=min(v);    vmax=max(v);  
  
kmin=floor(vmin/width);  
xmin=kmin*width;  
kmax=floor(vmax/width);  
xmax=kmax*width;  
  
nclasses=kmax-kmin+1;  
hist=zeros(1,nclasses);  
xv=linspace(xmin,xmax+width,nclasses+1);  
  
for vk=v  
    kh=sum(xv<=vk);  
    hist(kh)=hist(kh)+1;  
end  
if (nargout>1)  
    xval=xv;  
end  
end
```