## PAC - PROBABILITA' AL CALCOLATORE 2004
## GARFICI CON GNUPLOT

E' molto utile visualizzare graficamente i risultati di una simulazione. Qui di seguito illustriamo alcuni semplici esempi concreti[1]. Facciamo riferimento al programma `gnuplot`. Per una introduzione all'uso si può consultare il sito `http://www.duke.edu/~hpgavin/gnuplot.html`.

**Istogramma delle frequenze.** Nel programma qui riprodotto suddividiamo l'intervallo $[0,1]$ in un certo numero $d$ di intervalli di uguale lunghezza e visualizziamo la frequenza empirica di ciascun intervallo. Cioè su $n$ lanci indipendenti di una variabile uniforme in $[0,1]$, calcoliamo il numero di volte in cui la variabile cade in ciascun intervallo.

```
/*  n variabili indipendenti uniformi in [0,1] */
/* ******* Istogramma delle Frequenze ******* */

# include    <stdlib.h>
# include    <time.h>
# include    <stdio.h>
# include    <string.h>
# include    <math.h>

int main(void)
{
      double r;
      char comando[30];
      FILE *dati,*F;
      int   n,i,k,d,ord[100];

      srand48((unsigned)time(NULL));

      printf("numero variabili da simulare:");
      scanf("%d",&n);
      printf("numero colonne istogramma (max 100):");
      scanf("%d",&d);

      for (k=0; k < d; k++){
         ord[k]=0;}

      for (i=0; i < n; i++){
         r=drand48();
         for (k=0; k < d;k++){
            if(r>=(double)k/d && r<(double)(k+1)/d)
               ord[k]++;}
      }

      dati=fopen("esito","wt");
      for(i=0;i<d;i++){
         fprintf(dati,"%f \t %d \n",((float)(2*i+1)/(float)(2*d)),ord[i]);}
      fclose(dati);
      F=fopen("isto.p","wt");
      fprintf(F,"set title 'Istogramma frequenze' \n set xr [0.0:1.0] \n");
      fprintf(F,"plot 'esito' with histeps \n pause -1");
      fclose(F);
      sprintf(comando,"gnuplot > load 'isto.p'");
      system(comando);
      return 0;
}
```

Note del corso di PAC, per informazioni: Pietro Caputo, caputo@mat.uniroma3.it.

[1]Si ringraziano gli studenti Marco Caponigro e Federico Coglitore per la gentile concessione di alcuni codici C.

**Grafico media e varianza.** Il prossimo programma permette di visualizzare, su uno stesso grafico, media e varianza empirica.

```
/*  n variabili indipendenti uniformi in [0,1] */
/* ******* Grafico Media e Varianza Empirica ******* */

# include   <stdlib.h>
# include   <time.h>
# include   <stdio.h>
# include   <string.h>
# include   <math.h>

int main(void)
{
     double x,s,v,ord1[100],ord2[100];
     char comando[30];
     FILE *dati,*F;
     int   n,i,k,d,m;

     srand48((unsigned)time(NULL));

     printf("numero variabili da simulare:");
     scanf("%d",&n);
     printf("numero punti grafico (max 100):");
     scanf("%d",&d);

     for (k=0; k < d; k++){
       ord1[k]=0;
       ord2[k]=0;}
     x=0;
     s=0;
     v=0;
     m=(int)n/d;
     for(i=0;i<n;i++){
       x=drand48();
       s+=x;
       v+=x*x;
       for (k=0; k < d;k++){
         if(i==k*m){
           ord1[k]=s/(i+1);
           ord2[k]=v/(i+1) - ord1[k]*ord1[k];}
       }
     }

     dati=fopen("misure","wt");
     for(i=0;i<d;i++){
       fprintf(dati,"%d \t %f \t %f \n",i,ord1[i],ord2[i]);}
     fclose(dati);
     F=fopen("grafico.p","wt");
     fprintf(F,"set title 'Media e Varianza' \n
                set xlabel 'numero variabili (x n/d)' \n ");
     fprintf(F,"set ylabel 'm e v' \n
                plot 'misure' using 1:2 title 'Media' with lines \n ");
     fprintf(F,"replot 'misure' using 1:3 title 'Varianza' with lines \n
                pause -1");
     fclose(F);
     sprintf(comando,"gnuplot > load 'grafico.p'");
     system(comando);
     return 0;
}
```