**Este documento contiene el código fuente del programa KDEN que es utilizado en la investigación desarrollada en el artículo: “Construcción de Kernels y Funciones de Densidad de Probabilidad”. El uso de este programa se describe en un manual resumido al final de este artículo.**

**Se incluyen el programa y doce funciones subyacentes en formato texto. El programa y cada función deben copiarse separadamente a ventanas de edición de MATLAB y guardarse con el nombre respectivo para su utilización.**

**------------------------------------------------------------------------------------------------------------------------------------**

**% Programa KDEN**

**% Estudio de kernels y funciones de densidad**

**% Luis Rodríguez Ojeda**

**% FCNM – ESPOL - 2013**

**format short**

**format compact**

**opc=0;**

**while opc~=4**

**mx(1);**

**try**

**opc=input('Elija una opción ');**

**end**

**if length(opc)==0**

**opc=0;**

**end**

**if isnumeric(opc)==0 | length(opc)>1**

**opc=0;**

**end**

**switch opc**

**case 1**

**try**

**tu=input('Datos de la muestra en un vector o entre [ ] ');**

**catch**

**disp('Error: Ingrese el nombre del vector o los datos entre [ ]');**

**break**

**end**

**if length(tu)==0**

**opc=0;**

**else**

**u=tu;**

**kdenpuntos(u);**

**end**

**case 2**

**try**

**tu=input('Datos de la muestra en un vector o entre [ ] ');**

**catch**

**disp('Error: Ingrese el nombre del vector o los datos entre [ ]');**

**break**

**end**

**if length(tu)==0**

**opc=0;**

**else**

**kb=0;**

**while kb==0**

**kb=1;**

**try**

**tbordes=input('Bordes de clases en un vector o entre [ ] ');**

**if length(tbordes)==0**

**kb=0;**

**end**

**catch**

**disp('Ingrese vector');**

**end**

**end**

**if length(tu)>0 & length(tbordes)>1**

**u=tu;**

**bordes=tbordes;**

**kdenhisto(u,bordes);**

**end**

**end**

**case 3**

**tk=0;**

**while tk~=12**

**mx(2);**

**try**

**tk=input('Elija Kernel ');**

**end**

**if length(tk)==0**

**tk=0;**

**end**

**if isnumeric(tk)==0 | length(tk)>1**

**tk=0;**

**end**

**switch tk**

**case 1**

**try**

**tu=input('Datos de la muestra en un vector o entre [ ] ');**

**catch**

**disp('Error: Ingrese el nombre del vector o los datos entre [ ]');**

**break**

**end**

**if length(tu)>1 & max(tu)>min(tu)**

**hopt=hstar(tu,tk);**

**disp(['Ancho de banda óptimo sugerido ',num2str(hopt)]);**

**end**

**if length(tu)==0**

**tu=0;**

**end**

**th=input('Ingrese el ancho de banda ');**

**if length(th)==0**

**th=1;**

**end**

**if length(tu)>0 & th>0**

**u=tu;**

**h=th;**

**kdenn(u,h);**

**end**

**case {2,4,5,6,7,8}**

**try**

**tu=input('Datos de la muestra en un vector o entre [ ] ');**

**catch**

**disp('Error: Ingrese el nombre del vector o los datos entre [ ]');**

**break**

**end**

**if length(tu)>1 & max(tu)>min(tu)**

**hopt=hstar(tu,tk);**

**disp(['Ancho de banda óptimo sugerido ',num2str(hopt)]);**

**end**

**if length(tu)==0**

**tu=0;**

**end**

**th=input('Ingrese el ancho de banda ');**

**if length(th)==0**

**th=1;**

**end**

**if length(tu)>0 & th>0**

**u=tu;**

**h=th;**

**kdent(u,h,tk);**

**end**

**case {3,9,10,11}**

**try**

**tu=input('Datos de la muestra en un vector o entre [ ] ');**

**catch**

**disp('Error: Ingrese el nombre del vector o los datos entre [ ]');**

**break**

**end**

**if length(tu)>1 & max(tu)>min(tu)**

**hopt=hstar(tu,tk);**

**disp(['Ancho de banda óptimo sugerido ',num2str(hopt)]);**

**end**

**if length(tu)==0**

**tu=0;**

**end**

**th=input('Ingrese el ancho de banda ');**

**if length(th)==0**

**th=1;**

**end**

**if length(tu)>0 & th>0**

**u=tu;**

**h=th;**

**kdenp(u,h,tk);**

**end**

**end**

**end**

**end**

**end**

**disp('FCNM - ESPOL');**

**function kdenn(u,h)**

**%Estudio del kernel normal**

**k='1/sqrt(2\*pi)\*exp(-x^2/2)';**

**s=0;**

**n=length(u);**

**u=sort(u);**

**mu=mean(u);**

**smuestral=std(u);**

**if n==1 | min(u)==max(u)**

**smuestral=1;**

**end**

**vs=[ ];**

**ts=[ ];**

**%Puntos de densidad**

**nk=n;**

**if nk<=2**

**np=200;**

**else**

**if nk<=10**

**np=100;**

**else**

**if nk<=50**

**np=20;**

**else**

**if nk<=100**

**np=10;**

**else**

**np=5;**

**end**

**end**

**end**

**end**

**for z=mu-4\*smuestral:h/np:mu+4\*smuestral**

**r=0;**

**for i=1:n**

**x=(z-u(i))/h;**

**r=r+eval(k);**

**end**

**ts=[ts,z];**

**vs=[vs,r/(n\*h)];**

**end**

**maxrg=max(vs);**

**opc=0;**

**while opc~=7**

**mx(3);**

**opc=input('Elija una opción ');**

**if length(opc)==0**

**opc=0;**

**end**

**switch opc**

**case 1**

**%gráfico de kernels**

**figure(3)**

**clf;grid on;hold on**

**%Puntos de kernels**

**for i=1:n**

**vk=[ ];**

**for z=mu-4\*smuestral:h/np:mu+4\*smuestral**

**x=(z-u(i))/h;**

**r=eval(k);**

**vk=[vk,r/(n\*h)];**

**end**

**plot(ts,vk);**

**end**

**maxrgk=max(vk);**

**for i=1:n**

**plot(u(i),0,'ko','LineWidth',2);**

**end**

**axis([mu-3.5\*smuestral,mu+3.5\*smuestral,0,1.2\*maxrgk]);**

**title(['Kernels Normal h = ',num2str(h)]);**

**pause;**

**case 2**

**%Gráfico de la función de densidad**

**figure(4);**

**clf;grid on;hold on**

**plot(ts,vs,'k');**

**for i=1:n**

**plot(u(i),0,'ko','LineWidth',2);**

**end**

**axis([mu-3.5\*smuestral,mu+3.5\*smuestral,0,1.2\*maxrg]);**

**title(['Función de densidad Normal h = ',num2str(h)]);**

**pause;**

**case 3**

**%gráfico de densidad y kernels**

**figure(5)**

**clf;grid on;hold on**

**%Puntos de kernels**

**for i=1:n**

**vk=[ ];**

**for z=mu-4\*smuestral:h/np:mu+4\*smuestral**

**x=(z-u(i))/h;**

**r=eval(k);**

**vk=[vk,r/(n\*h)];**

**end**

**plot(ts,vk);**

**end**

**for i=1:n**

**plot(u(i),0,'ko','LineWidth',2);**

**end**

**plot(ts,vs,'k');**

**axis([mu-3.5\*smuestral,mu+3.5\*smuestral,0,1.2\*maxrg]);**

**title(['Densidad y Kernels Normal h = ',num2str(h)]);**

**pause**

**case 4**

**clc;**

**disp('Medidas de densidad')**

**disp(' ');**

**[mu,vark,sesgo,rax,rbx,amp,med,q1d,q3d]=meden(u,ts,vs);**

**disp('Media');**

**disp(mu);**

**if n==1 | min(ts)==max(ts)**

**vark=1;**

**end**

**disp('Varianza')**

**disp(vark);**

**disp('Sesgo');**

**disp(sesgo);**

**disp('Mediana');**

**disp(med);**

**disp('Primer Cuartil');**

**disp(q1d);**

**disp('Tercer Cuartil');**

**disp(q3d);**

**disp(' ');**

**disp('Presione una tecla para continuar');**

**pause**

**case 5**

**r=prob(ts,vs);**

**disp(' ');**

**disp('Probabilidad');**

**disp(r);**

**disp(' ');**

**disp('Presione una tecla para continuar');**

**pause**

**case 6**

**clc;**

**disp('Función de densidad');**

**syms x**

**k=1/sqrt(2\*pi)\*exp(-x^2/2);**

**s=0;**

**n=length(u);**

**for i=1:n**

**p=subs(k,x,(x-u(i))/h)/(n\*h); % el dominio de todos los kernels es -inf a +inf**

**s=s+p;**

**end**

**digits(6);**

**s=vpa(s);**

**disp(s);**

**disp('-inf < x < inf');**

**disp(' ');**

**disp('Presione una tecla para continuar');**

**pause**

**end**

**end**

**function kdent(u,h,kt)**

**% Kernels definidos en un solo tramo**

**switch kt**

**case 2**

**k='0.5'; %Rectangular**

**tit='Rectangular';**

**case 4**

**k='3/4\*(1-x^2)'; %Epanechnikov**

**tit='Epanechnikov';**

**case 5**

**k='15/16\*(1-x^2)^2'; %Biweigth**

**tit='Biweigth';**

**case 6**

**k='pi/4\*cos(pi/2\*x)'; %Arco coseno**

**tit='Arco Coseno';**

**case 7**

**k='sqrt(1.145834-x^2)-0.381882'; %Arco circular**

**tit='Arco Circular';**

**case 8**

**k='(cos(pi\*x)+1)/2'; %Coseno**

**tit='Coseno';**

**end**

**n=length(u);**

**%Definición de tramos**

**clear a b**

**for i=1:n**

**a(i)=u(i)-h;**

**b(i)=u(i)+h;**

**end**

**% Construcción de puntos de la función de densidad**

**clear z ts vs**

**z=sort([a,b]);**

**z=eliminar(z); %puntos de quiebre de intervalos de kernels**

**nk=length(z);**

**vs=[ ];**

**ts=[ ];**

**if nk<=2**

**np=500;**

**else**

**if nk<=10**

**np=100;**

**else**

**if nk<=50**

**np=20;**

**else**

**if nk<=100**

**np=10;**

**else**

**np=5;**

**end**

**end**

**end**

**end**

**for kk=1:nk-1**

**dh=(z(kk+1)-z(kk))/np;**

**for j=0:np-1**

**t=z(kk)+j\*dh;**

**r=0;**

**for i=1:n**

**if a(i)<=t & t<=b(i)**

**x=(t-u(i))/h;**

**r=r+eval(k);**

**end**

**end**

**ts=[ts,t];**

**vs=[vs,r/(n\*h)];**

**end**

**end**

**t=z(nk);**

**y=0;**

**ts=[ts,t];**

**vs=[vs,y];**

**maxrg=max(vs);**

**opc=0;**

**while opc~=7**

**mx(3);**

**opc=input('Elija una opción ');**

**if length(opc)==0**

**opc=0;**

**end**

**switch opc**

**case 1**

**% Gráfico de kernels**

**figure(3)**

**clf; grid on,hold on**

**%Puntos de kernels**

**for i=1:n**

**tk=[ ];**

**vk=[ ];**

**for t=a(i):h/20:b(i)**

**tk=[tk,t];**

**x=(t-u(i))/h;**

**r=eval(k);**

**vk=[vk,r/(n\*h)];**

**end**

**plot(tk,vk)**

**end**

**maxrgk=max(vk);**

**for i=1:n**

**plot(u(i),0,'ko','LineWidth',2);**

**end**

**axis([min(a),max(b),0,1.2\*maxrgk]);**

**title(['Kernels ',tit,' h = ',num2str(h)]);**

**pause;**

**case 2**

**%Gráfico de la función de densidad**

**figure(4)**

**clf;hold on,grid on**

**plot(ts,vs,'k');**

**for i=1:n**

**plot(u(i),0,'ko','LineWidth',2);**

**end**

**axis([min(z),max(z),0,1.2\*maxrg]);**

**title(['Función de densidad ',tit,' h = ',num2str(h)]);**

**pause**

**case 3**

**% Gráfico de la función de densidad y kernels**

**figure(5)**

**clf; grid on,hold on**

**%Puntos de kernels**

**for i=1:n**

**tk=[ ];**

**vk=[ ];**

**for t=a(i):h/20:b(i)**

**tk=[tk,t];**

**x=(t-u(i))/h;**

**r=eval(k);**

**vk=[vk,r/(n\*h)];**

**end**

**plot(tk,vk)**

**end**

**for i=1:n**

**plot(u(i),0,'ko','LineWidth',2);**

**end**

**plot(ts,vs,'k');**

**axis([min(a),max(b),0,1.2\*maxrg]);**

**title(['Densidad y Kernels ',tit,' h = ',num2str(h)]);**

**pause;**

**case 4**

**clc;**

**disp('Medidas de densidad')**

**disp(' ');**

**[mu,vark,cs3,rax,rbx,amp,med,q1d,q3d]=meden(u,ts,vs);**

**disp('Media');**

**disp(mu);**

**disp('Varianza')**

**disp(vark);**

**disp('Sesgo');**

**disp(cs3);**

**disp('Rango')**

**disp([' [',num2str(rax),', ',num2str(rbx),']']);**

**disp('Amplitud');**

**disp(amp);**

**disp('Mediana');**

**disp(med);**

**disp('Primer Cuartil');**

**disp(q1d);**

**disp('Tercer Cuartil');**

**disp(q3d);**

**disp(' ');**

**disp('Presione una tecla para continuar');**

**pause**

**case 5**

**r=prob(ts,vs);**

**disp(' ');**

**disp('Probabilidad');**

**disp(r);**

**disp(' ');**

**disp('Presione una tecla para continuar');**

**pause**

**case 6**

**%Función de densidad por segmentos**

**syms x s**

**switch kt**

**case 2**

**k=0.5; %Rectangular**

**case 4**

**k=3/4\*(1-x^2); %Epanechnikov**

**case 5**

**k=15/16\*(1-x^2)^2; %Biweigth**

**case 6**

**k=pi/4\*cos(pi/2\*x); %Arco coseno**

**case 7**

**k=sqrt(1.145834-x^2)-0.381882; %Arco circular**

**case 8**

**k=(cos(pi\*x)+1)/2; %Coseno**

**end**

**n=length(u);**

**clear p;**

**for i=1:n**

**v=subs(k,x,(x-u(i))/h);**

**p{i}=v/(n\*h); %kernels en cada intervalo**

**end**

**%definición de tramos**

**clear a b**

**for i=1:n**

**a(i)=u(i)-h;**

**b(i)=u(i)+h;**

**end**

**% Construcción de la función de densidad**

**clear ps z c d**

**z=sort([a,b]);**

**z=eliminar(z); %puntos de quiebre de intervalos para sumar kernels**

**nk=length(z);**

**c=z(1:nk-1);**

**d=z(2:nk);**

**for ks=1:nk-1**

**vz=[ ];**

**m=(c(ks)+d(ks))/2;**

**for i=1:n**

**if a(i)<m & m<b(i)**

**vz=[vz, i]; %vector con los números de kernels a sumar**

**end**

**end**

**s=0;**

**for i=1:length(vz)**

**j=vz(i);**

**s=s+p{j};**

**end**

**ps{ks}=s;**

**end**

**clc;**

**disp('Función de densidad');**

**digits(6);**

**for i=1:nk-1**

**if ismember('x',char(ps{i}))==0**

**fps=ps{i};**

**else**

**fps=vpa(expand(ps{i}));**

**end**

**disp(fps);**

**fprintf('%20.4f <= x <= %8.4f\n',c(i),d(i))**

**end**

**disp(' ');**

**disp('Presione una tecla para continuar');**

**pause**

**end**

**end**

**function kdenp(u,h,kt)**

**%Kernels con dos tramos**

**switch kt**

**case 3**

**kp='1+x'; %Triangular**

**kq='1-x';**

**tit='Triángular';**

**case 9**

**kp='2/3\*(1+x^3)'; %Cúbico plano**

**kq='2/3\*(1-x^3)';**

**tit='Cúbico Plano';**

**case 10**

**kp='2/5\*(-x^3-3\*x^2+2)'; %Cúbico natural**

**kq='2/5\*( x^3-3\*x^2+2)';**

**tit='Cúbico Natural';**

**case 11**

**kp='-2\*x^3 - 3\*x^2 + 1'; %Cúbico sujeto**

**kq=' 2\*x^3 - 3\*x^2 + 1';**

**tit='Cúbico Sujeto';**

**end**

**n=length(u);**

**%definición de tramos**

**clear ap bp aq bq**

**for i=1:n**

**ap(i)=u(i)-h;**

**bp(i)=u(i);**

**aq(i)=u(i);**

**bq(i)=u(i)+h;**

**end**

**%Construcción de puntos de la función de densidad con polinomios segmentarios**

**clear z ts vs**

**z=sort([ap,bp,aq,bq]);**

**z=eliminar(z); %Puntos de quiebre de kernels**

**nk=length(z);**

**vs=[ ];**

**ts=[ ];**

**if nk<=2**

**np=500;**

**else**

**if nk<=10**

**np=100;**

**else**

**if nk<=50**

**np=20;**

**else**

**if nk<=100**

**np=10;**

**else**

**np=5;**

**end**

**end**

**end**

**end**

**for kk=1:nk-1**

**dh=(z(kk+1)-z(kk))/np;**

**for j=0:np-1**

**t=z(kk)+j\*dh;**

**r=0;**

**for i=1:n**

**if ap(i)<t & t<bp(i)**

**x=(t-u(i))/h;**

**r=r+eval(kp);**

**end**

**if aq(i)<=t & t<=bq(i)**

**x=(t-u(i))/h;**

**r=r+eval(kq);**

**end**

**end**

**ts=[ts,t];**

**vs=[vs,r/(n\*h)];**

**end**

**end**

**t=z(nk);**

**y=0;**

**ts=[ts,t];**

**vs=[vs,y];**

**maxrg=max(vs);**

**opc=0;**

**while opc~=7**

**mx(3);**

**opc=input('Elija una opción ');**

**if length(opc)==0**

**opc=0;**

**end**

**switch opc**

**case 1**

**% Gráfico de kernels**

**figure(3)**

**clf; grid on,hold on**

**%Puntos de kernels**

**for i=1:n**

**tkp=[ ];**

**vkp=[ ];**

**for t=ap(i):h/20:bp(i)**

**tkp=[tkp,t];**

**x=(t-u(i))/h;**

**r=eval(kp);**

**vkp=[vkp,r/(n\*h)];**

**end**

**plot(tkp,vkp)**

**tkq=[ ];**

**vkq=[ ];**

**for t=aq(i):h/20:bq(i)**

**tkq=[tkq,t];**

**x=(t-u(i))/h;**

**r=eval(kq);**

**vkq=[vkq,r/(n\*h)];**

**end**

**plot(tkq,vkq)**

**end**

**maxrgk=max(vkq);**

**for i=1:n**

**plot(u(i),0,'ko','LineWidth',2);**

**end**

**axis([min(ap),max(bq),0,1.2\*maxrgk]);**

**title(['Kernels ',tit,' h = ',num2str(h)]);**

**pause;**

**case 2**

**%Gráfico de la función de densidad**

**figure(4)**

**clf;hold on,grid on**

**plot(ts,vs,'k');**

**for i=1:n**

**plot(u(i),0,'ko','LineWidth',2);**

**end**

**axis([min(z),max(z),0,1.2\*maxrg]);**

**title(['Función de densidad ',tit,' h = ',num2str(h)]);**

**pause;**

**case 3**

**% Gráfico de la función de densidad y kernels**

**figure(5)**

**clf; grid on,hold on**

**%Puntos de kernels**

**for i=1:n**

**tkp=[ ];**

**vkp=[ ];**

**for t=ap(i):h/20:bp(i)**

**tkp=[tkp,t];**

**x=(t-u(i))/h;**

**r=eval(kp);**

**vkp=[vkp,r/(n\*h)];**

**end**

**plot(tkp,vkp)**

**tkq=[ ];**

**vkq=[ ];**

**for t=aq(i):h/20:bq(i)**

**tkq=[tkq,t];**

**x=(t-u(i))/h;**

**r=eval(kq);**

**vkq=[vkq,r/(n\*h)];**

**end**

**plot(tkq,vkq)**

**end**

**for i=1:n**

**plot(u(i),0,'ko','LineWidth',2);**

**end**

**plot(ts,vs,'k');**

**axis([min(ap),max(bq),0,1.2\*maxrg]);**

**title(['Densidad y Kernels ',tit,' h = ',num2str(h)]);**

**pause;**

**case 4**

**clc;**

**disp('Medidas de densidad')**

**disp(' ');**

**[mu,vark,sesgo,rax,rbx,amp,med,q1d,q3d]=meden(u,ts,vs);**

**disp('Media');**

**disp(mu);**

**disp('Varianza')**

**disp(vark);**

**disp('Sesgo');**

**disp(sesgo);**

**disp('Rango')**

**disp([' [',num2str(rax),', ',num2str(rbx),']']);**

**disp('Amplitud');**

**disp(amp);**

**disp('Mediana');**

**disp(med);**

**disp('Primer Cuartil');**

**disp(q1d);**

**disp('Tercer Cuartil');**

**disp(q3d);**

**disp(' ');**

**disp('Presione una tecla para continuar');**

**pause**

**case 5**

**r=prob(ts,vs);**

**disp(' ');**

**disp('Probabilidad');**

**disp(r);**

**disp(' ');**

**disp('Presione una tecla para continuar');**

**pause**

**case 6**

**syms x s**

**switch kt**

**case 3**

**kp=1+x; %Triangular**

**kq=1-x;**

**case 9**

**kp=2/3\*(1+x^3); %Cúbico plano**

**kq=2/3\*(1-x^3);**

**case 10**

**kp=2/5\*(-x^3-3\*x^2+2); %Cúbico natural**

**kq=2/5\*( x^3-3\*x^2+2);**

**case 11**

**kp=-2\*x^3 - 3\*x^2 + 1; %Cúbico sujeto**

**kq= 2\*x^3 - 3\*x^2 + 1;**

**end**

**n=length(u);**

**clear p q;**

**for i=1:n**

**vp=subs(kp,x,(x-u(i))/h);**

**vq=subs(kq,x,(x-u(i))/h);**

**p{i}=vp/(n\*h); %kernels en cada intervalo**

**q{i}=vq/(n\*h);**

**end %el dominio es diferente en cada uno**

**%definición de tramos**

**clear ap bp aq bq**

**for i=1:n**

**ap(i)=u(i)-h;**

**bp(i)=u(i);**

**aq(i)=u(i);**

**bq(i)=u(i)+h;**

**end**

**%Función de densidad con segmentos**

**clear ps z c d**

**z=sort([ap,bp,aq,bq]);**

**z=eliminar(z);**

**nk=length(z);**

**c=z(1:nk-1);**

**d=z(2:nk);**

**for ks=1:nk-1**

**vzp=[ ];**

**vzq=[ ];**

**m=(c(ks)+d(ks))/2;**

**for i=1:n**

**if ap(i)<m & m<bp(i)**

**vzp=[vzp, i];**

**end**

**if aq(i)<m & m<bq(i)**

**vzq=[vzq, i];**

**end**

**end**

**s=0;**

**for i=1:length(vzp)**

**j=vzp(i);**

**s=s+p{j};**

**end**

**for i=1:length(vzq)**

**j=vzq(i);**

**s=s+q{j};**

**end**

**ps{ks}=s;**

**end**

**clc;**

**disp('Función de densidad');**

**digits(6);**

**for i=1:nk-1**

**if ismember('x',char(ps{i}))==0**

**fps=ps{i};**

**else**

**fps=vpa(expand(ps{i}));**

**end**

**disp(fps);**

**fprintf('%20.4f <= x <= %8.4f\n',c(i),d(i))**

**end**

**disp(' ');**

**disp('Presione una tecla para continuar');**

**pause;**

**end**

**end**

**function mx(km)**

**%Rótulos**

**clc;**

**switch km**

**case 1**

**disp('Estudio de Kernels')**

**disp(' ');**

**disp('1) Gráfico de puntos');**

**disp('2) Histograma');**

**disp('3) Kernels');**

**disp('4) Salir');**

**disp(' ');**

**case 2**

**disp('Kernels disponibles ');**

**disp(' 1) Normal o Gaussiano');**

**disp(' 2) Rectangular');**

**disp(' 3) Triangular');**

**disp(' 4) Epanechnikov');**

**disp(' 5) Biweigth');**

**disp(' 6) Arco coseno');**

**disp(' 7) Arco circular');**

**disp(' 8) Coseno');**

**disp(' 9) Cúbico plano');**

**disp('10) Cúbico natural');**

**disp('11) Cúbico sujeto')**

**disp('12) Salir');**

**disp(' ');**

**case 3**

**disp('Opciones disponibles');**

**disp(' ');**

**disp('1) Gráfico de kernels');**

**disp('2) Gráfico de la función de densidad');**

**disp('3) Gráfico de la función de densidad y kernels');**

**disp('4) Estadísticas del estimador');**

**disp('5) Cálculo de probabilidad');**

**disp('6) Definición de la función de densidad');**

**disp('7) Salir');**

**disp(' ');**

**end**

**function kdenpuntos(u)**

**%Gráfico de puntos**

**figure(1)**

**clf;grid on;hold on**

**n=length(u);**

**for i=1:n**

**plot(u(i),0,'kd','LineWidth',2);**

**end**

**if n>1 & min(u)~=max(u)**

**du=(max(u)-min(u))/(n-1);**

**else**

**du=1;**

**end**

**axis([min(u)-du,max(u)+du,0,1]);**

**title('Puntos');**

**clc;**

**disp('Medidas muestrales')**

**disp(' ');**

**disp('Cantidad de datos');**

**disp(n);**

**disp('Media');**

**mu=mean(u);**

**disp(mu);**

**disp('Varianza');**

**v=var(u);**

**disp(v);**

**ax=min(u);**

**bx=max(u);**

**disp('Rango')**

**disp([' [',num2str(ax),', ',num2str(bx),']']);**

**disp('Amplitud');**

**disp(max(u)-min(u));**

**[med,q1,q3]=medm(u);**

**disp('Mediana');**

**disp(med);**

**disp('Primer cuartil');**

**disp(q1);**

**disp('Tercer cuartil');**

**disp(q3);**

**disp(' ');**

**disp('Presione una tecla para continuar');**

**pause**

**function kdenhisto(u,bordes)**

**%Histograma**

**figure(2);**

**clf;grid on;hold on**

**conteo=histc(u,bordes);**

**bar(bordes, conteo, 'histc');**

**title('Histograma')**

**colormap summer**

**clc;**

**disp('Medidas muestrales')**

**disp(' ');**

**disp('Cantidad de datos');**

**disp(length(u));**

**disp('Media');**

**mu=mean(u);**

**disp(mu);**

**disp('Varianza');**

**v=var(u);**

**disp(v);**

**ax=min(u);**

**bx=max(u);**

**disp('Rango')**

**disp([' [',num2str(ax),', ',num2str(bx),']']);**

**disp('Amplitud');**

**disp(max(u)-min(u));**

**[med,q1,q3]=medm(u);**

**disp('Mediana');**

**disp(med);**

**disp('Primer cuartil');**

**disp(q1);**

**disp('Tercer cuartil');**

**disp(q3);**

**disp(' ');**

**disp('Presione una tecla para continuar');**

**pause**

**function hopt=hstar(u,kn)**

**%Cálculo de h óptimo**

**hk=[1.000, 1.380, 2.212, 1.947, 1.985, 2.370, 1.781, 2.526, 1.953, 2.018, 2.515];**

**[med,q1,q3]=medm(u);**

**n=length(u);**

**s=std(u);**

**ri=q3-q1;**

**hg=0.9\*n^(-1/5)\*min(s,ri/1.349);**

**hopt=hg\*hk(kn);**

**function [med,q1,q3]=medm(u)**

**%Medidas muestrales**

**u=sort(u);**

**n=length(u);**

**if mod(n,2)==0**

**m=n/2;**

**med=0.5\*(u(m)+u(m+1));**

**if mod(m,2)==0**

**m1=m/2;**

**q1=0.5\*(u(m1)+u(m1+1));**

**m3=m+m/2;**

**q3=0.5\*(u(m3)+u(m3+1));**

**else**

**m1=fix(m/2)+1;**

**q1=u(m1);**

**m3=m+fix(m/2)+1;**

**q3=u(m3);**

**end**

**else**

**m=fix(n/2)+1;**

**med=u(m);**

**if mod(m-1,2)==0**

**m1=fix(m/2)+1;**

**q1=u(m1);**

**m3=m+fix(m/2);**

**q3=u(m3);**

**else**

**m1=fix(m/2);**

**q1=0.5\*(u(m1)+u(m1+1));**

**m3=m+fix(m/2)-1;**

**q3=0.5\*(u(m3)+u(m3+1));**

**end**

**end**

**function [mu,vark,sesgo,rax,rbx,amp,med,q1d,q3d]=meden(u,ts,vs)**

**% Medidas de densidad**

**%Media**

**mu=mean(u);**

**%Varianza**

**ns=length(ts);**

**vs2=[ ];**

**for i=1:ns**

**vs2(i)=(ts(i)-mu)^2\*vs(i);**

**end**

**vark=trp(ts,vs2,ts(ns));**

**if abs(vark)<0.0001**

**vark=0;**

**end**

**%Sesgo**

**vs3=[ ];**

**for i=1:ns**

**vs3(i)=(ts(i)-mu)^3\*vs(i);**

**end**

**sesgo=trp(ts,vs3,ts(ns));**

**if abs(sesgo)<0.0001**

**sesgo=0;**

**end**

**%Rango**

**rax=min(ts);**

**rbx=max(ts);**

**%Amplitud**

**amp=rbx-rax;**

**%Mediana**

**[q2,q1,q3]=medm(u);**

**ax=q2;**

**for i=1:6**

**r=trp(ts,vs,ax);**

**dx=(max(ts)-min(ts))/10^i;**

**if r>0.5**

**while r>0.5**

**ax=ax-dx;**

**r=trp(ts,vs,ax);**

**end**

**ax=ax+dx;**

**else**

**while r<0.5**

**ax=ax+dx;**

**r=trp(ts,vs,ax);**

**end**

**ax=ax-dx;**

**end**

**end**

**med=ax;**

**if abs(med)<0.0001**

**med=0;**

**end**

**%Cuartil 1**

**ax=q1;**

**for i=1:6**

**r=trp(ts,vs,ax);**

**dx=(max(ts)-min(ts))/10^i;**

**if r>0.25**

**while r>0.25**

**ax=ax-dx;**

**r=trp(ts,vs,ax);**

**end**

**ax=ax+dx;**

**else**

**while r<0.25**

**ax=ax+dx;**

**r=trp(ts,vs,ax);**

**end**

**ax=ax-dx;**

**end**

**end**

**q1d=ax;**

**%Cuartil 3**

**ax=q3;**

**for i=1:6**

**r=trp(ts,vs,ax);**

**dx=(max(ts)-min(ts))/10^i;**

**if r>0.75**

**while r>0.75**

**ax=ax-dx;**

**r=trp(ts,vs,ax);**

**end**

**ax=ax+dx;**

**else**

**while r<0.75**

**ax=ax+dx;**

**r=trp(ts,vs,ax);**

**end**

**ax=ax-dx;**

**end**

**end**

**q3d=ax;**

**function r=prob(ts,vs)**

**%Cálculo de probabilidad**

**nuloa=0;**

**nulob=0;**

**ns=length(ts);**

**disp(' ');**

**disp('Cálculo de probabilidad');**

**disp(' ');**

**ax=input('Izquierda ');**

**if length(ax)==0**

**ax=ts(1);**

**nuloa=1;**

**end**

**if ax<ts(1)**

**ax=ts(1);**

**end**

**bx=input('Derecha ');**

**if length(bx)==0**

**bx=ts(ns);**

**nulob=1;**

**end**

**if bx>ts(ns)**

**bx=ts(ns);**

**end**

**if bx<ax**

**bx=ax;**

**end**

**if nuloa==1 & nulob==1**

**ax=ts(1);**

**bx=ts(1);**

**end**

**r=trp(ts,vs,bx)-trp(ts,vs,ax);**

**function r = trp(x,y,b)**

**%Integración aproximada desde x(1) hasta b**

**n=length(x);**

**if b<x(1)**

**r=0;**

**return**

**end**

**if b>x(n)**

**r=1;**

**return;**

**end**

**for i=n-1:-1:1**

**if b>=x(i)**

**k=i;**

**break**

**end**

**end**

**switch k**

**case 1**

**d1=x(2)-x(1);**

**yp=y(1)+(y(2)-y(1))/d1\*(b-x(1));**

**r=(b-x(1))\*0.5\*(y(1)+yp);**

**case 2**

**d1=x(2)-x(1);**

**d2=x(3)-x(2);**

**s=d1\*0.5\*(y(1)+y(2));**

**yp=y(2)+(y(3)-y(2))/d2\*(b-x(2));**

**r=s+(b-x(2))\*0.5\*(y(2)+yp);**

**case 3**

**d1=x(2)-x(1);**

**d2=x(3)-x(2);**

**d3=x(4)-x(3);**

**s=d1\*0.5\*(y(1)+y(2));**

**s=s+d2\*0.5\*(y(2)+y(3));**

**yp=y(3)+(y(4)-y(3))/d3\*(b-x(3));**

**r=s+(b-x(3))\*0.5\*(y(3)+yp);**

**otherwise**

**s=0;**

**for i=2:k-2**

**di=x(i+1)-x(i);**

**s=s+di\*0.5\*(y(i)+y(i+1));**

**end**

**d1=x(2)-x(1);**

**dk1=x(k)-x(k-1);**

**dk=x(k+1)-x(k);**

**s=s + d1\*0.5\*(y(1)+y(2)) + dk1\*0.5\*(y(k-1)+y(k));**

**yp=y(k)+(y(k+1)-y(k))/dk\*(b-x(k));**

**r=s+(b-x(k))\*0.5\*(y(k)+yp);**

**end**

**function u=eliminar(v)**

**%Purgar un vector**

**n=length(v);**

**u=[ ];**

**for i=1:n**

**r=ismember(v(i),u);**

**if r==0**

**u=[u,v(i)];**

**end**

**end**