Caracterización de Distribuciones Bivariadas Seleccionadas

Definición de funciones de densidad de probabilidad

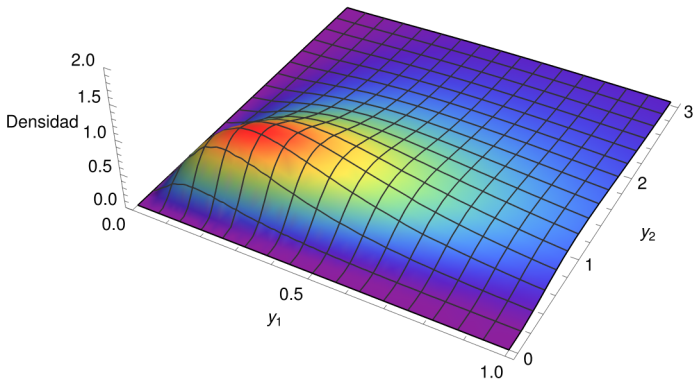
```
(*Definir los valores de los parámetros a,
b,c,d*){aVal, bVal, cVal, dVal} = {3, 6, 3, 6};
(*Definir los valores de los parámetros a,b,c,
d*){aVal2, bVal2, cVal2, dVal2} = {0.8, 6, 3, 0.2};
```

Combinación 1 (Beta para X_1 y Beta 4P para $X_2 \mid X_1$)

In[•]:=
$$f[x\_, y\_, a\_, b\_, c\_, d\_] := \frac{1}{\text{Beta}[a, b]\,\text{Beta}[c, d]}\,\frac{x^{a-1}\,y^{b-1}\,(x+y)^{d-(a+b)}}{(x+y+1)^{c+d}};$$

```
fx[x_] :=
  N[NIntegrate[f[x, y, a = aVal, b = bVal, c = cVal, d = dVal], {y, 0, Infinity}], 3];
fy[y_] := N[NIntegrate[f[x, y, a = aVal, b = bVal, c = cVal, d = dVal], {x, 0, Infinity}], 3];
Fx[t_] := N[NIntegrate[
    f[x, y, a = aVal, b = bVal, c = cVal, d = dVal], {y, 0, Infinity}, {x, 0, t}], 3];
Fy[t_] := N[NIntegrate[
    f[x, y, a = aVal, b = bVal, c = cVal, d = dVal], {x, 0, Infinity}, {y, 0, t}], 3];
Plot3df = Plot3D[f[x, y, aVal, bVal, cVal, dVal], {x, 0, 1},
  {y, 0, 3}, PlotRange → {0, 2}, AxesLabel → {y1, y2, Densidad},
  ColorFunction → "Rainbow", Boxed → False, PlotLabel → "Combinación 1"]
```

Out[•]=



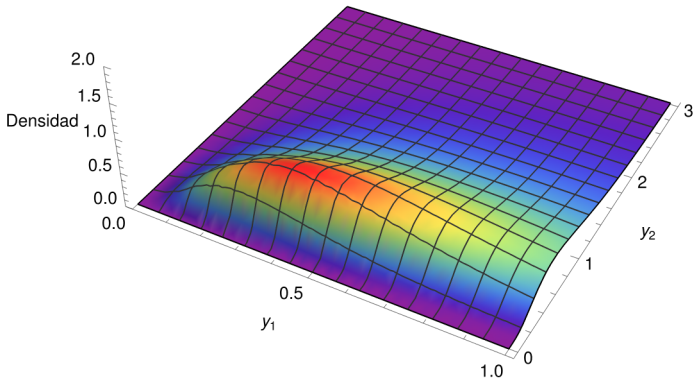
Combinación 6 (Kumaraswamy para X_1 y Beta 4P para $X_2 \mid X_1$)

In[•]:=
$$g[x\_, y\_, a\_, b\_, c\_, d\_] := \frac{a\,b}{\text{Beta}[c, d]} \frac{x^{a-1}\left((x+y)^a - x^a\right)^{b-1}}{(x+y)^{a\,b-d+1}\,(x+y+1)^{c+d}};$$

```
gx[x_] :=
  N[NIntegrate[g[x, y, a = aVal, b = bVal, c = cVal, d = dVal], {y, 0, Infinity}], 3];
gy[y_] := N[NIntegrate[g[x, y, a = aVal, b = bVal, c = cVal, d = dVal], {x, 0, Infinity}], 3];
Gx[t_] := N[NIntegrate[
    g[x, y, a = aVal, b = bVal, c = cVal, d = dVal], {y, 0, Infinity}, {x, 0, t}], 3];
Gy[t_] := N[NIntegrate[
    g[x, y, a = aVal, b = bVal, c = cVal, d = dVal], {x, 0, Infinity}, {y, 0, t}], 3];
Plot3dg = Plot3D[g[x, y, aVal, bVal, cVal, dVal], {x, 0, 1},
  {y, 0, 3}, PlotRange → {0, 2}, AxesLabel → {y1, y2, Densidad},
  ColorFunction → "Rainbow", Boxed → False, PlotLabel → "Combinación 6"]
```

Combinación 6

Out[•]=

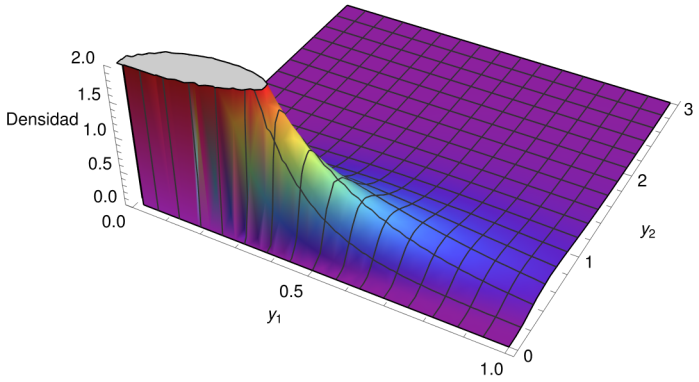


Combinación 8 (Kumaraswamy para X_1 y Triangular para $X_2|\, X_1$)

In[•]:=
```
h[x_, y_, a_, b_, d_] :=
 2 a b / d  x^(a-1) ((x + y)^a - x^a)^(b-1) / (x + y)^(a b) (x + y + 1)^2  Which[0 ≤ x y < d (x + y)^2 (x + y + 1), x y / ((x + y)^2 (x + y + 1)),
                                                                                cuál
  x y == d (x + y)^2 (x + y + 1), d, d (x + y)^2 < d (x + y)^2 (x + y + 1) ≤ x y, x y (x + y) d / ((x + y + 1) (x y - d (x + y)^2))];

hx[x_] := N[NIntegrate[h[x, y, a = aVal, b = bVal2, d = dVal2],
            · integra numéricamente
   {y, 0, Infinity}, Method → "Oscillatory"], 3];
              infinito          método

hy[y_] := N[NIntegrate[h[x, y, a = aVal, b = bVal2, d = dVal2],
            · integra numéricamente
   {x, 0, Infinity}, Method → "Oscillatory"], 3];
              infinito          método

Hx[t_] := N[NIntegrate[h[x, y, a = aVal, b = bVal2, d = dVal2],
            · integra numéricamente
    {y, 0, Infinity}, {x, 0, t}, AccuracyGoal → 4, PrecisionGoal → 4], 3];
              infinito                 objetivo de exactitud      objetivo de precisión

Hy[t_] := N[NIntegrate[h[x, y, a = aVal, b = bVal2, d = dVal2],
            · integra numéricamente
    {x, 0, Infinity}, {y, 0, t}, AccuracyGoal → 4, PrecisionGoal → 4], 3];
              infinito                 objetivo de exactitud      objetivo de precisión

Plot3dh = Plot3D[h[x, y, aVal, bVal2, dVal2], {x, 0, 1},
                 representación gráfica 3D
  {y, 0, 3}, PlotRange → {0, 2}, AxesLabel → {y1, y2, Densidad},
             rango de representación  etiqueta de ejes
  ColorFunction → "Rainbow", Boxed → False, PlotLabel → "Combinación 8"]
  función de color                rodeado···  falso     etiqueta de representación
```

Combinación 8

Out[•]=

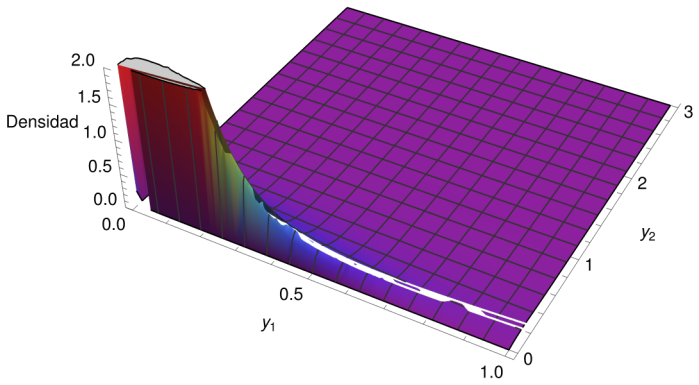


Combinación 11 (Triangular para X_1 y Beta 4P para $X_2 | X_1$)

In[•]:=
```
j[x_, y_, a_, c_, d_] := 2/(a Beta[c, d]) (x + y)^(d-2)/(x + y + 1)^(c+d)

    Which[0 ≤ x < a (x + y), x/(x + y), x == y/(1 - a), a, a (x + y) < x ≤ x + y, (y a)/((1 - a) (x + y))];
    cuál

jx[x_] := N[NIntegrate[j[x, y, a = aVal2, c = cVal2, d = dVal2], {y, 0, Infinity}], 3];
          · integra numéricamente                                                  infinito

jy[y_] := N[NIntegrate[j[x, y, a = aVal2, c = cVal2, d = dVal2], {x, 0, Infinity}], 3];
          · integra numéricamente                                                  infinito

Jx[t_] :=
  N[NIntegrate[j[x, y, a = aVal2, c = cVal2, d = dVal2], {y, 0, Infinity}, {x, 0, t}], 3];
   · integra numéricamente                                                  infinito

Jy[t_] :=
  N[NIntegrate[j[x, y, a = aVal2, c = cVal2, d = dVal2], {x, 0, Infinity}, {y, 0, t}], 3];
   · integra numéricamente                                                  infinito

Plot3dj = Plot3D[j[x, y, aVal2, cVal2, dVal2], {x, 0, 1},
                 representación gráfica 3D

  {y, 0, 3}, PlotRange → {0, 2}, AxesLabel → {y1, y2, Densidad},
             rango de representación  etiqueta de ejes

  ColorFunction → "Rainbow", Boxed → False, PlotLabel → "Combinación 11"]
  función de color           rodeado···  falso  etiqueta de representación
```

Combinación 11

Out[•]=



In[•]:=
```
Jy[4]
```

Out[•]=
```
0.956113
```

Combinación 14 (Triangular para X_1 y Gamma Truncada para $X_2 \mid X_1$)

In[•]:=
```
k[x_, y_, a_, c_, d_] := 2/(a d^c) x^c y^c/((x + y)^(2 c+1) (x + y + 1)^(c+1)) Which[0 ≤ x < a (x + y),
```
cuál

$$\frac{x}{x+y},\ x(1-a) == y,\ a,\ a(x+y) < x \le x+y,\ \frac{y\,a}{(1-a)(x+y)}\Big]\frac{\text{Exp}\left[-\frac{x\,y}{d\,(x+y)^2\,(x+y+1)}\right]}{\text{Gamma}\left[c,\ 0,\ \frac{x\,y}{d\,(x+y)^2}\right]};$$

```
kx[x_] := N[NIntegrate[k[x, y, a = aVal2, c = cVal2, d = dVal2], {y, 0, Infinity}], 3];
```
integra numéricamente · infinito

```
ky[y_] := N[NIntegrate[k[x, y, a = aVal2, c = cVal2, d = dVal2], {x, 0, Infinity}], 3];
```
integra numéricamente · infinito

```
Kx[t_] :=
  N[NIntegrate[k[x, y, a = aVal2, c = cVal2, d = dVal2], {y, 0, Infinity}, {x, 0, t}], 3];
```
integra numéricamente · infinito

```
Ky[t_] :=
  N[NIntegrate[k[x, y, a = aVal2, c = cVal2, d = dVal2], {x, 0, Infinity}, {y, 0, t}], 3];
```
integra numéricamente · infinito

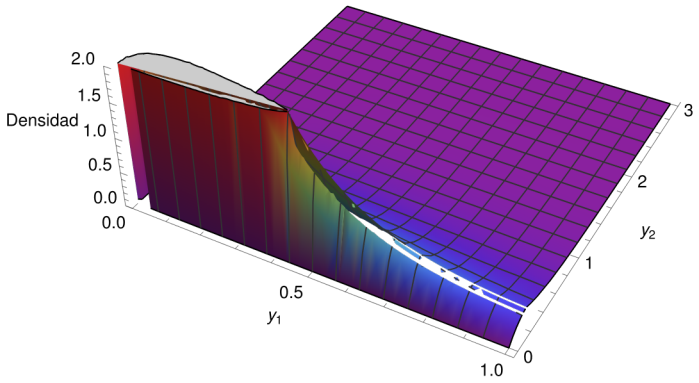
```
Plot3dk = Plot3D[k[x, y, aVal2, cVal2, dVal2], {x, 0, 1},
```
representación gráfica 3D

```
  {y, 0, 3}, PlotRange → {0, 2}, AxesLabel → {y1, y2, Densidad},
```
rango de representación · etiqueta de ejes

```
  ColorFunction → "Rainbow", Boxed → False, PlotLabel → "Combinación 14"]
```
función de color · rodeado··· · falso · etiqueta de representación

Combinación 14

Out[•]=



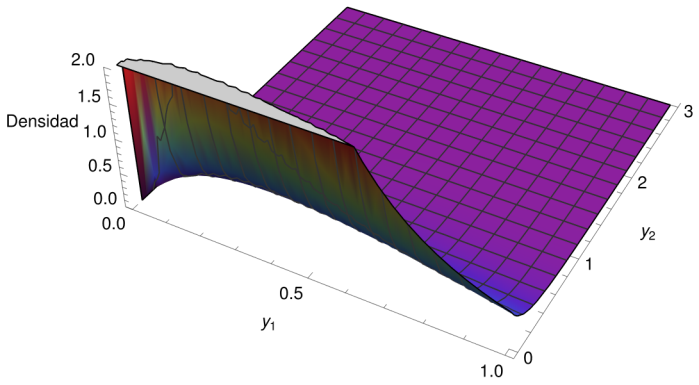
Combinación 19 (Gamma Truncada para X_1 y Gamma Truncada para

$X_2 \mid X_1)$

In[ ]:=
```
l[x_, y_, a_, b_, c_, d_] :=

  1                    x^(a+c-1) y^c                 Exp[-(x y)/(d (x+y)^2 (x+y+1)) - x/(b (x+y))]
-------- ------------------------------------ ;
b^a d^c   (Gamma[a, 0, 1/b])  (x + y)^(a+2 c) (x + y + 1)^(c+1)   Gamma[c, 0, (x y)/(d (x+y)^2)]

lx[x_] := N[NIntegrate[l[x, y, a = aVal, b = bVal, c = cVal, d = dVal], {y, 0, Infinity}], 3];
          · integra numéricamente                                                  infinito

ly[y_] := N[NIntegrate[l[x, y, a = aVal, b = bVal, c = cVal, d = dVal], {x, 0, Infinity}], 3];
          · integra numéricamente                                                  infinito

Lx[t_] :=  N[NIntegrate[
           · integra numéricamente

    l[x, y, a = aVal, b = bVal, c = cVal, d = dVal], {y, 0, Infinity}, {x, 0, t}], 3];
                                                                          infinito

Ly[t_] := N[NIntegrate[
           · integra numéricamente

    l[x, y, a = aVal, b = bVal, c = cVal, d = dVal], {x, 0, Infinity}, {y, 0, t}], 3];
                                                                          infinito

Plot3dl = Plot3D[l[x, y, aVal, bVal, cVal, dVal], {x, 0, 1},
                 representación gráfica 3D

  {y, 0, 3}, PlotRange → {0, 2}, AxesLabel → {y1, y2, Densidad},
                 rango de representación   etiqueta de ejes

  ColorFunction → "Rainbow", Boxed → False,  PlotLabel → "Combinación 19"]
  función de color                 rodeado···  falso      etiqueta de representación
```

Combinación 19

Out[ ]=

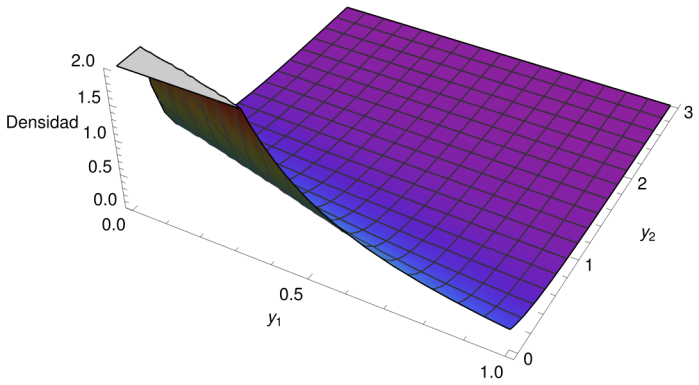


Combinación 25 (Normal Truncada para X_1 y Normal Truncada para $X_2 \mid X_1)$

In[•]:=
```
e[x_, y_, a_, b_, c_, d_] :=
 (1 / (((CDF[NormalDistribution[0, 1], (1 - a)/Sqrt[b]] - CDF[NormalDistribution[0, 1], (-a)/Sqrt[b]])
    2 Pi Sqrt[b d])))  (x y)/((x + y)^3 (x + y + 1)^2)

 Exp[-1/(2 d) ((x y)/((x+y)^2 (x+y+1)) - c)^2 - 1/(2*b) (x/(x+y) - a)^2] /
  CDF[NormalDistribution[0, 1], ((x y)/(x+y)^2 - c)/Sqrt[d]] - CDF[NormalDistribution[0, 1], (-c)/Sqrt[d]];

ex[x_] := N[NIntegrate[e[x, y, a = aVal, b = bVal, c = cVal, d = dVal], {y, 0, Infinity}], 3];

ey[y_] := N[NIntegrate[e[x, y, a = aVal, b = bVal, c = cVal, d = dVal], {x, 0, Infinity}], 3];

Ex[t_] := N[NIntegrate[
    e[x, y, a = aVal, b = bVal, c = cVal, d = dVal], {y, 0, Infinity}, {x, 0, t}], 3];

Ey[t_] := N[NIntegrate[
    e[x, y, a = aVal, b = bVal, c = cVal, d = dVal], {x, 0, Infinity}, {y, 0, t}], 3];

Plot3de = Plot3D[e[x, y, aVal, bVal, cVal, dVal], {x, 0, 1},
  {y, 0, 3}, PlotRange → {0, 2}, AxesLabel → {y1, y2, Densidad},
  ColorFunction → "Rainbow", Boxed → False, PlotLabel → "Combinación 25"]
```

Combinación 25

Out[•]=

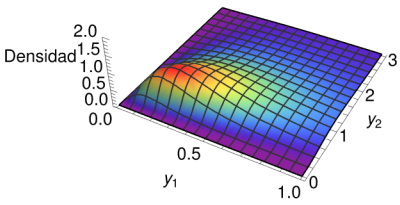


Superficies

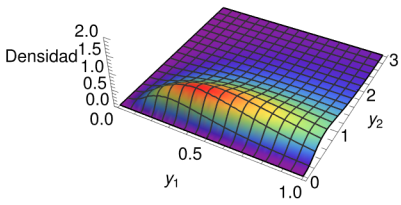
In[ ]:=
```
GraphicsGrid[{{Plot3df, Plot3dg}, {Plot3dh, Plot3dj}, {Plot3dk, Plot3dl}, {Plot3de,}}]
```
rejilla de gráficos

Out[ ]=

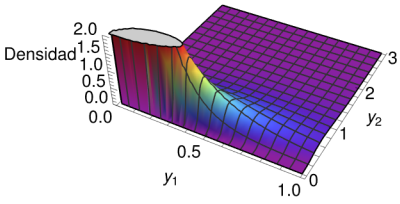
Combinación 1



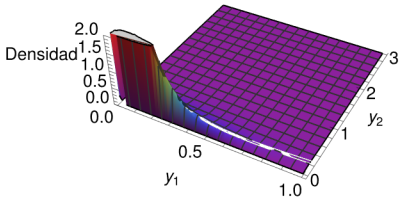
Combinación 6



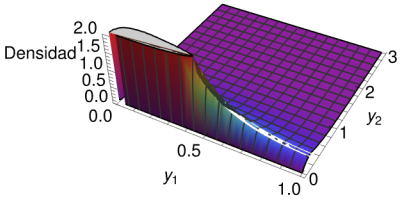
Combinación 8



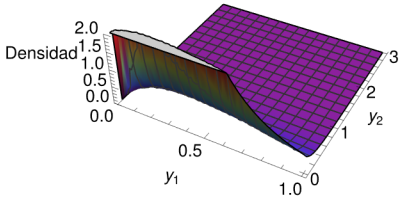
Combinación 11



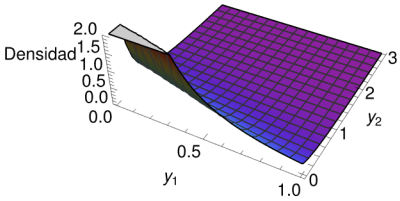
Combinación 14



Combinación 19



Combinación 25



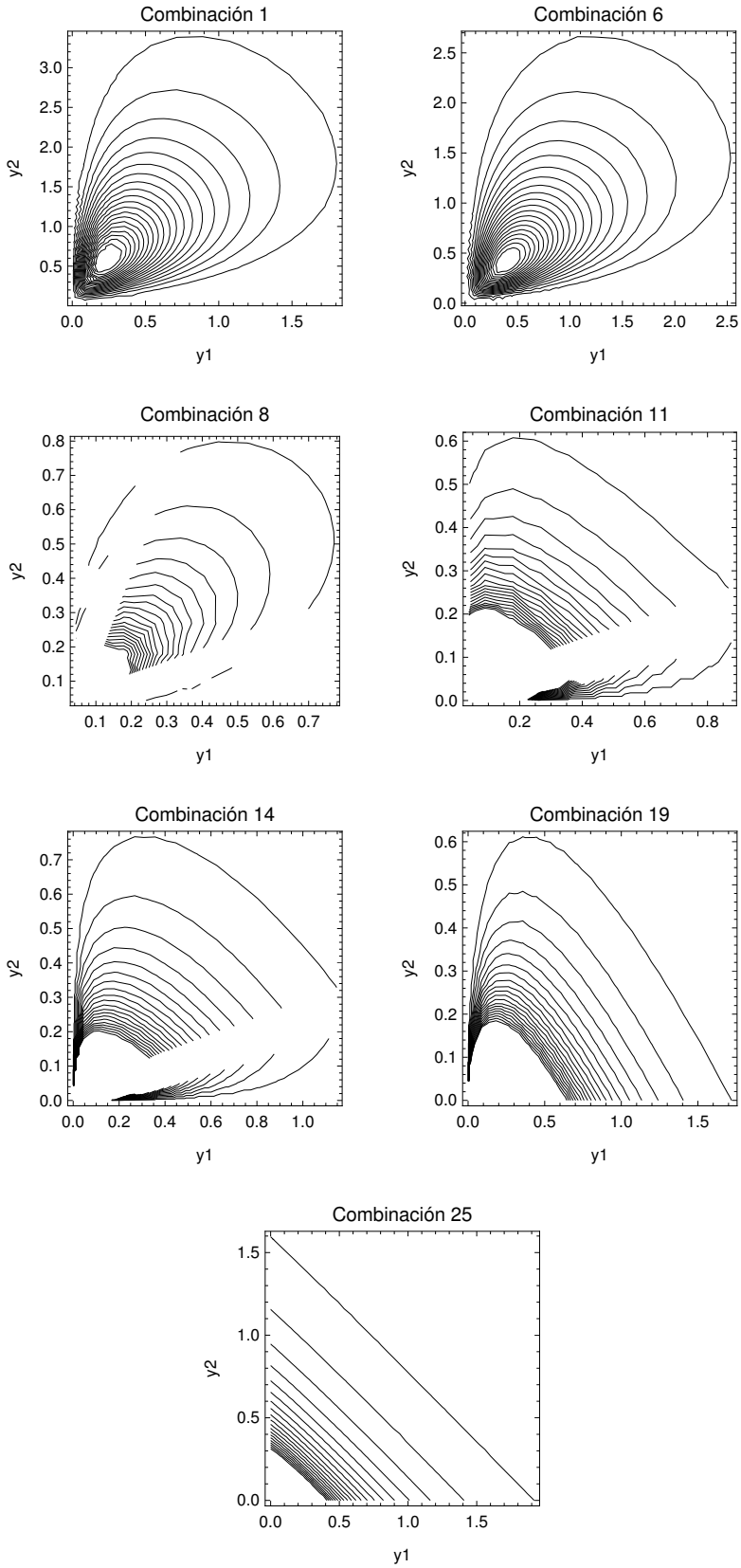
Curvas de Nivel

In[ ]:=
```
(*Generación de las curvas de nivel para las diferentes funciones*)
contourPlotF = ContourPlot[f[x, y, aVal, bVal, cVal, dVal],
```
representación de contornos

```
    {x, 0, 10}, {y, 0, 10}, Contours → 20, ContourShading → None,
                                contornos   sombreado de contornos  ninguno
    PlotRange → All, FrameLabel → {"y1", "y2"}, PlotLabel → "Combinación 1"];
    rango de repr···  todo  etiqueta de marco      etiqueta de representación
contourPlotG = ContourPlot[g[x, y, aVal, bVal, cVal, dVal],
                  representación de contornos
    {x, 0, 10}, {y, 0, 10}, Contours → 20, ContourShading → None,
                                contornos   sombreado de contornos  ninguno
    PlotRange → All, FrameLabel → {"y1", "y2"}, PlotLabel → "Combinación 6"];
    rango de repr···  todo  etiqueta de marco      etiqueta de representación
contourPlotH = ContourPlot[h[x, y, aVal, bVal2, dVal2], {x, 0, 10},
                  representación de contornos
    {y, 0, 10}, Contours → 20, ContourShading → None, PlotRange → All,
                  contornos   sombreado de contornos  ning···  rango de repr···  todo
    FrameLabel → {"y1", "y2"}, PlotLabel → "Combinación 8"];
    etiqueta de marco      etiqueta de representación
contourPlotJ = ContourPlot[j[x, y, aVal2, cVal2, dVal2], {x, 0, 10},
                  representación de contornos
    {y, 0, 10}, Contours → 20, ContourShading → None, PlotRange → All,
                  contornos   sombreado de contornos  ning···  rango de repr···  todo
    FrameLabel → {"y1", "y2"}, PlotLabel → "Combinación 11"];
    etiqueta de marco      etiqueta de representación
contourPlotK = ContourPlot[k[x, y, aVal2, cVal2, dVal2], {x, 0, 10},
                  representación de contornos
    {y, 0, 10}, Contours → 20, ContourShading → None, PlotRange → All,
                  contornos   sombreado de contornos  ning···  rango de repr···  todo
    FrameLabel → {"y1", "y2"}, PlotLabel → "Combinación 14"];
    etiqueta de marco      etiqueta de representación
contourPlotL = ContourPlot[l[x, y, aVal, bVal, cVal, dVal], {x, 0, 10},
                  representación de contornos
    {y, 0, 10}, Contours → 20, ContourShading → None, PlotRange → All,
                  contornos   sombreado de contornos  ning···  rango de repr···  todo
    FrameLabel → {"y1", "y2"}, PlotLabel → "Combinación 19"];
    etiqueta de marco      etiqueta de representación
contourPlotD = ContourPlot[e[x, y, aVal, bVal, cVal, dVal], {x, 0, 10},
                  representación de contornos
    {y, 0, 10}, Contours → 20, ContourShading → None, PlotRange → All,
                  contornos   sombreado de contornos  ning···  rango de repr···  todo
    FrameLabel → {"y1", "y2"}, PlotLabel → "Combinación 25"];
    etiqueta de marco      etiqueta de representación
(*Mostrar las gráficas juntas*)
```

```
GraphicsGrid[{{contourPlotF, contourPlotG}, {contourPlotH, contourPlotJ},
 rejilla de gráficos
  {contourPlotK, contourPlotL}, {contourPlotD, SpanFromLeft}}]
                                                 extiende desde la izquierda
```

Out[•]=


Combinación 1
y2
y1
Combinación 6
y2
y1
Combinación 8
y2
y1
Combinación 11
y2
y1
Combinación 14
y2
y1
Combinación 19
y2
y1
Combinación 25
y2
y1

Función de Densidad Marginal

In[•]:=
```
LaunchKernels[7];
```
lanza kernels
```
DistributeDefinitions[fx, fy, gx, gy, hx, hy, jx, jy, kx, ky, lx, ly, ex, ey];
```
distribuye definiciones
```
generatePlotfx :=
  Module[{pointsx, label}, pointsx = ParallelTable[{x, fx[x]}, {x, 0.1, 6, 0.1}];
```
módulo, tabla en paralelo
```
   label = "Combinación 1";
   ListPlot[pointsx, Joined → True,
```
representación de lista, unido, verdadero
```
    PlotRange → {{0, 6}, {0, 1}}, PlotLabel → label, PlotStyle → Blue];
```
rango de representación, etiqueta de representación, estilo de repre···, azul
```
generatePlotfy :=
  Module[{pointsy, label}, pointsy = ParallelTable[{y, fy[y]}, {y, 0.1, 6, 0.1}];
```
módulo, tabla en paralelo
```
   label = "Combinación 1";
   ListPlot[pointsy, Joined → True, PlotRange → {{0, 6}, {0, 1}},
```
representación de lista, unido, verd···, rango de representación
```
    PlotLabel → label, PlotStyle → Directive[Red, Dashed]];
```
etiqueta de representación, estilo de repre···, directiva, rojo, rayado
```
generatePlotgx :=
  Module[{pointsx, label}, pointsx = ParallelTable[{x, gx[x]}, {x, 0.1, 6, 0.1}];
```
módulo, tabla en paralelo
```
   label = "Combinación 6";
   ListPlot[pointsx, Joined → True,
```
representación de lista, unido, verdadero
```
    PlotRange → {{0, 6}, {0, 1}}, PlotLabel → label, PlotStyle → Blue];
```
rango de representación, etiqueta de representación, estilo de repre···, azul
```
generatePlotgy :=
  Module[{pointsy, label}, pointsy = ParallelTable[{y, gy[y]}, {y, 0.1, 6, 0.1}];
```
módulo, tabla en paralelo
```
   label = "Combinación 6";
   ListPlot[pointsy, Joined → True, PlotRange → {{0, 6}, {0, 1}},
```
representación de lista, unido, verd···, rango de representación
```
    PlotLabel → label, PlotStyle → Directive[Red, Dashed]];
```
etiqueta de representación, estilo de repre···, directiva, rojo, rayado
```
generatePlothx :=
  Module[{pointsx, label}, pointsx = ParallelTable[{x, hx[x]}, {x, 0.1, 6, 0.1}];
```
módulo, tabla en paralelo

```
    label = "Combinación 8";
    ListPlot[pointsx, Joined → True,
     PlotRange → {{0, 6}, {0, 1}}, PlotLabel → label, PlotStyle → Blue];

generatePlothy :=
  Module[{pointsy, label}, pointsy = ParallelTable[{y, hy[y]}, {y, 0.1, 6, 0.1}];
    label = "Combinación 8";
    ListPlot[pointsy, Joined → True, PlotRange → {{0, 6}, {0, 1}},
     PlotLabel → label, PlotStyle → Directive[Red, Dashed]];

generatePlotjx :=
  Module[{pointsx, label}, pointsx = ParallelTable[{x, jx[x]}, {x, 0.1, 6, 0.1}];
    label = "Combinación 11";
    ListPlot[pointsx, Joined → True,
     PlotRange → {{0, 6}, {0, 1}}, PlotLabel → label, PlotStyle → Blue];

generatePlotjy :=
  Module[{pointsy, label}, pointsy = ParallelTable[{y, jy[y]}, {y, 0.1, 6, 0.1}];
    label = "Combinación 11";
    ListPlot[pointsy, Joined → True, PlotRange → {{0, 6}, {0, 1}},
     PlotLabel → label, PlotStyle → Directive[Red, Dashed]];

generatePlotkx :=
  Module[{pointsx, label}, pointsx = ParallelTable[{x, kx[x]}, {x, 0.1, 6, 0.1}];
    label = "Combinación 14";
    ListPlot[pointsx, Joined → True,
     PlotRange → {{0, 6}, {0, 1}}, PlotLabel → label, PlotStyle → Blue];

generatePlotky :=
  Module[{pointsy, label}, pointsy = ParallelTable[{y, ky[y]}, {y, 0.1, 6, 0.1}];
    label = "Combinación 14";
```

```
    ListPlot[pointsy, Joined → True, PlotRange → {{0, 6}, {0, 1}},
    representación de lista  unido  verd···  rango de representación
      PlotLabel → label, PlotStyle → Directive[Red, Dashed]]];
      etiqueta de representación  estilo de repre···  directiva  rojo  rayado

generatePlotlx :=
  Module[{pointsx, label}, pointsx = ParallelTable[{x, lx[x]}, {x, 0.1, 6, 0.1}];
  módulo                              tabla en paralelo

    label = "Combinación 19";
    ListPlot[pointsx, Joined → True,
    representación de lista  unido  verdadero
      PlotRange → {{0, 6}, {0, 1}}, PlotLabel → label, PlotStyle → Blue]];
      rango de representación  etiqueta de representación  estilo de repre···  azul

generatePlotly :=
  Module[{pointsy, label}, pointsy = ParallelTable[{y, ly[y]}, {y, 0.1, 6, 0.1}];
  módulo                              tabla en paralelo

    label = "Combinación 19";
    ListPlot[pointsy, Joined → True, PlotRange → {{0, 6}, {0, 1}},
    representación de lista  unido  verd···  rango de representación
      PlotLabel → label, PlotStyle → Directive[Red, Dashed]]];
      etiqueta de representación  estilo de repre···  directiva  rojo  rayado

generatePlotex :=
  Module[{pointsx, label}, pointsx = ParallelTable[{x, ex[x]}, {x, 0.1, 6, 0.1}];
  módulo                              tabla en paralelo

    label = "Combinación 25";
    ListPlot[pointsx, Joined → True,
    representación de lista  unido  verdadero
      PlotRange → {{0, 6}, {0, 1}}, PlotLabel → label, PlotStyle → Blue]];
      rango de representación  etiqueta de representación  estilo de repre···  azul

generatePlotey :=
  Module[{pointsy, label}, pointsy = ParallelTable[{y, ey[y]}, {y, 0.1, 6, 0.1}];
  módulo                              tabla en paralelo

    label = "Combinación 25";
    ListPlot[pointsy, Joined → True, PlotRange → {{0, 6}, {0, 1}},
    representación de lista  unido  verd···  rango de representación
      PlotLabel → label, PlotStyle → Directive[Red, Dashed]]];
      etiqueta de representación  estilo de repre···  directiva  rojo  rayado
```

```
In[•]:= plots = {Show[generatePlotfx, generatePlotfy], Show[generatePlotgx, generatePlotgy],
                 muestra                                muestra
    Show[generatePlothx, generatePlothy], Show[generatePlotjx, generatePlotjy],
    muestra                               muestra
    Show[generatePlotkx, generatePlotky], Show[generatePlotlx, generatePlotly],
    muestra                               muestra
    Show[generatePlotex, generatePlotey], SpanFromLeft};
    muestra                               extiende desde la izquierda
  combinedPlot = GraphicsGrid[Partition[plots, 2], Spacings → {20, 20},
                              rejilla de gráficos  particiona  espaciados
   ImageSize → Large]
   tamaño de ima···  grande
  (* Añadir la leyenda *)
  Legended[combinedPlot, Placed[{"f(y1)", "f(y2)"}, Below]];
  con leyenda          colocado                      debajo
  CloseKernels[];
  cierra kernels
```

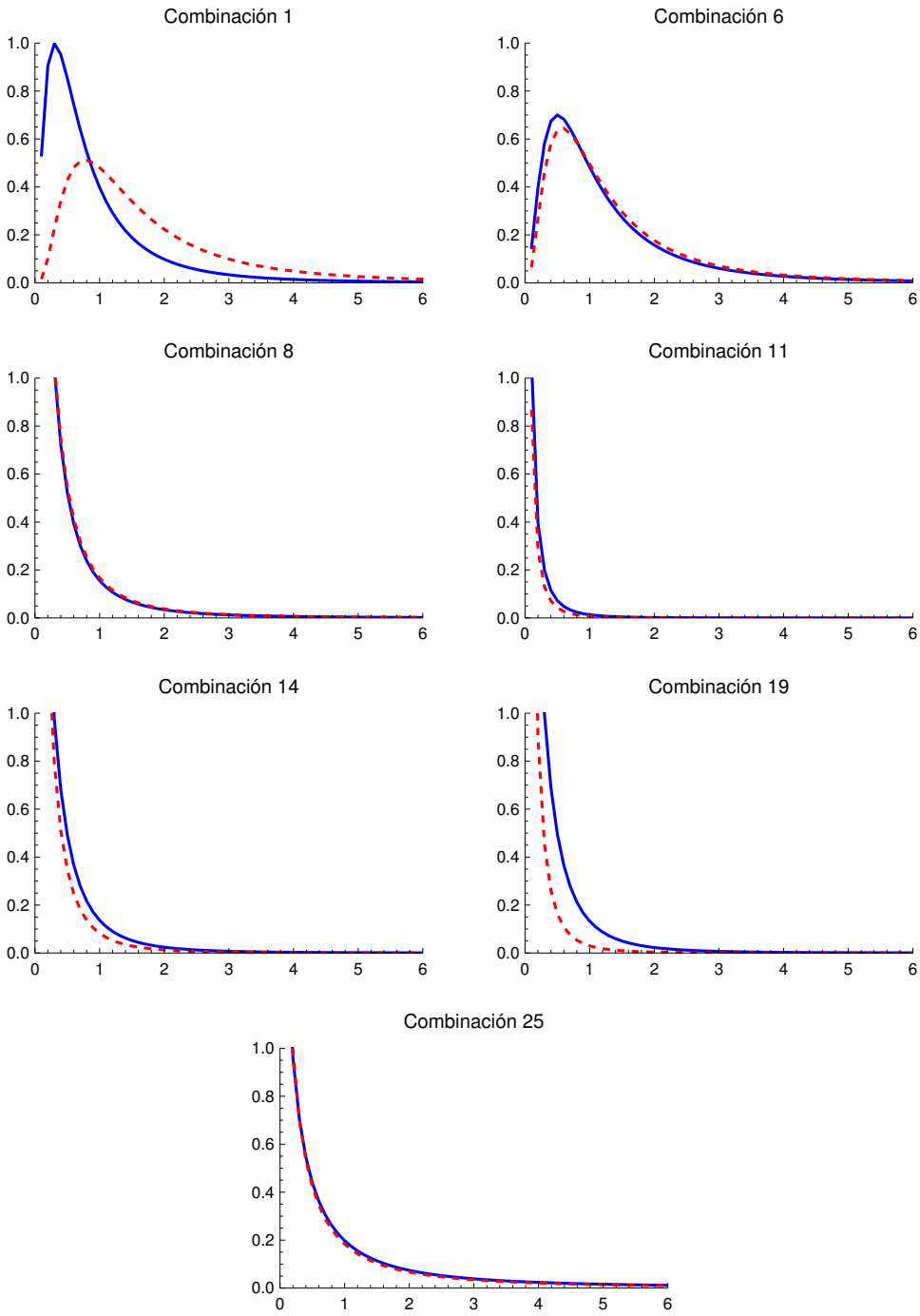
(kernel 43)

```
NIntegrate::ncvb :
 NIntegrate failed to converge to prescribed accuracy after 9 recursive
   bisections in y near {y} = {0.127773}. NIntegrate obtained 2.455246142585962`
   and 2.9939792368638543`*^-6 for the integral and error estimates.
```

(kernel 43)

```
NIntegrate::ncvb :
 NIntegrate failed to converge to prescribed accuracy after 9 recursive
   bisections in x near {x} = {0.127773}. NIntegrate obtained 2.368502436986138`
   and 3.282993534451745`*^-6 for the integral and error estimates.
```

Out[•]=



Distribución de Probabilidad Acumulada

In[•]:=
```
LaunchKernels[7];
```
lanza kernels
```
DistributeDefinitions[Fx, Fy, Gx, Gy, Hx, Hy, Jx, Jy, Kx, Ky, Lx, Ly, Ex, Ey];
```
distribuye definiciones

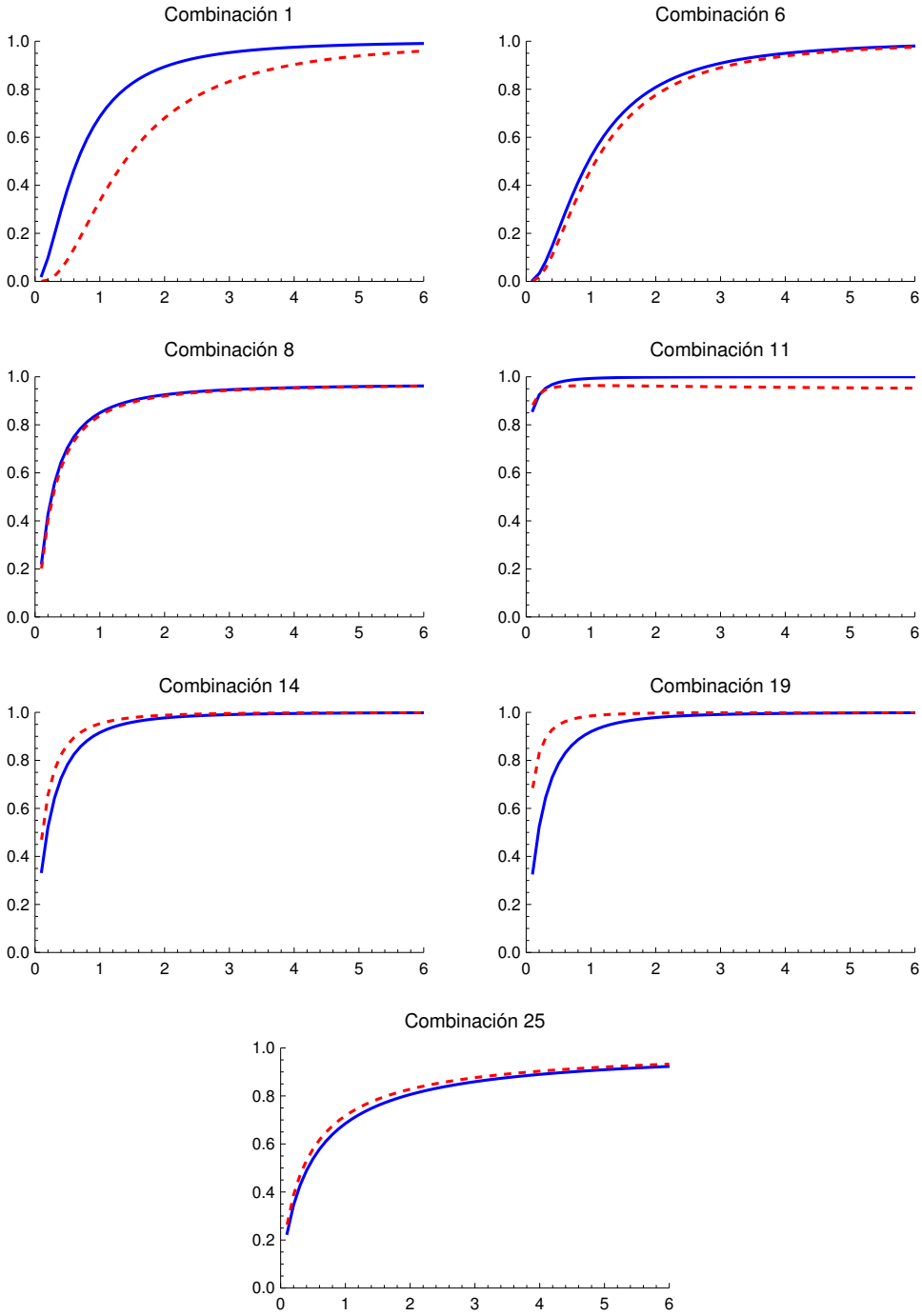
```
generatePlotFx :=
  Module[{pointsx, label}, pointsx = ParallelTable[{t, Fx[t]}, {t, 0.1, 6, 0.1}];
   label = "Combinación 1";
   ListPlot[pointsx, Joined → True,
    PlotRange → {{0, 6}, {0, 1}}, PlotLabel → label, PlotStyle → Blue]];
generatePlotFy :=
  Module[{pointsy, label}, pointsy = ParallelTable[{t, Fy[t]}, {t, 0.1, 6, 0.1}];
   label = "Combinación 1";
   ListPlot[pointsy, Joined → True, PlotRange → {{0, 6}, {0, 1}},
    PlotLabel → label, PlotStyle → Directive[Red, Dashed]]];

generatePlotGx :=
  Module[{pointsx, label}, pointsx = ParallelTable[{t, Gx[t]}, {t, 0.1, 6, 0.1}];
   label = "Combinación 6";
   ListPlot[pointsx, Joined → True,
    PlotRange → {{0, 6}, {0, 1}}, PlotLabel → label, PlotStyle → Blue]];
generatePlotGy :=
  Module[{pointsy, label}, pointsy = ParallelTable[{t, Gy[t]}, {t, 0.1, 6, 0.1}];
   label = "Combinación 6";
   ListPlot[pointsy, Joined → True, PlotRange → {{0, 6}, {0, 1}},
    PlotLabel → label, PlotStyle → Directive[Red, Dashed]]];

generatePlotHx :=
  Module[{pointsx, label}, pointsx = ParallelTable[{t, Hx[t]}, {t, 0.1, 6, 0.1}];
   label = "Combinación 8";
   ListPlot[pointsx, Joined → True,
    PlotRange → {{0, 6}, {0, 1}}, PlotLabel → label, PlotStyle → Blue]];
```

```
generatePlotHy :=
  Module[{pointsy, label}, pointsy = ParallelTable[{t, Hy[t]}, {t, 0.1, 6, 0.1}];
  módulo                             tabla en paralelo

   label = "Combinación 8";
   ListPlot[pointsy, Joined → True, PlotRange → {{0, 6}, {0, 1}},
   representación de lista  unido  verd···  rango de representación

    PlotLabel → label, PlotStyle → Directive[Red, Dashed]]];
    etiqueta de representación  estilo de repre···  directiva  rojo  rayado


generatePlotJx :=
  Module[{pointsx, label}, pointsx = ParallelTable[{t, Jx[t]}, {t, 0.1, 6, 0.1}];
  módulo                             tabla en paralelo

   label = "Combinación 11";
   ListPlot[pointsx, Joined → True,
   representación de lista  unido  verdadero

    PlotRange → {{0, 6}, {0, 1}}, PlotLabel → label, PlotStyle → Blue]];
    rango de representación  etiqueta de representación  estilo de repre···  azul

generatePlotJy :=
  Module[{pointsy, label}, pointsy = ParallelTable[{t, Jy[t]}, {t, 0.1, 6, 0.1}];
  módulo                             tabla en paralelo

   label = "Combinación 11";
   ListPlot[pointsy, Joined → True, PlotRange → {{0, 6}, {0, 1}},
   representación de lista  unido  verd···  rango de representación

    PlotLabel → label, PlotStyle → Directive[Red, Dashed]]];
    etiqueta de representación  estilo de repre···  directiva  rojo  rayado


generatePlotKx :=
  Module[{pointsx, label}, pointsx = ParallelTable[{t, Kx[t]}, {t, 0.1, 6, 0.1}];
  módulo                             tabla en paralelo

   label = "Combinación 14";
   ListPlot[pointsx, Joined → True,
   representación de lista  unido  verdadero

    PlotRange → {{0, 6}, {0, 1}}, PlotLabel → label, PlotStyle → Blue]];
    rango de representación  etiqueta de representación  estilo de repre···  azul

generatePlotKy :=
  Module[{pointsy, label}, pointsy = ParallelTable[{t, Ky[t]}, {t, 0.1, 6, 0.1}];
  módulo                             tabla en paralelo

   label = "Combinación 14";
   ListPlot[pointsy, Joined → True, PlotRange → {{0, 6}, {0, 1}},
   representación de lista  unido  verd···  rango de representación

    PlotLabel → label, PlotStyle → Directive[Red, Dashed]]];
    etiqueta de representación  estilo de repre···  directiva  rojo  rayado
```

```
generatePlotLx :=
  Module[{pointsx, label}, pointsx = ParallelTable[{t, Lx[t]}, {t, 0.1, 6, 0.1}];
  módulo                             tabla en paralelo

   label = "Combinación 19";
   ListPlot[pointsx, Joined → True,
   representación de lista  unido  verdadero

    PlotRange → {{0, 6}, {0, 1}}, PlotLabel → label, PlotStyle → Blue];
    rango de representación  etiqueta de representación  estilo de repre···  azul

generatePlotLy :=
  Module[{pointsy, label}, pointsy = ParallelTable[{t, Ly[t]}, {t, 0.1, 6, 0.1}];
  módulo                             tabla en paralelo

   label = "Combinación 19";
   ListPlot[pointsy, Joined → True, PlotRange → {{0, 6}, {0, 1}},
   representación de lista  unido  verd···  rango de representación

    PlotLabel → label, PlotStyle → Directive[Red, Dashed]];
    etiqueta de representación  estilo de repre···  directiva  rojo  rayado


generatePlotEx :=
  Module[{pointsx, label}, pointsx = ParallelTable[{t, Ex[t]}, {t, 0.1, 6, 0.1}];
  módulo                             tabla en paralelo

   label = "Combinación 25";
   ListPlot[pointsx, Joined → True,
   representación de lista  unido  verdadero

    PlotRange → {{0, 6}, {0, 1}}, PlotLabel → label, PlotStyle → Blue];
    rango de representación  etiqueta de representación  estilo de repre···  azul

generatePlotEy :=
  Module[{pointsy, label}, pointsy = ParallelTable[{t, Ey[t]}, {t, 0.1, 6, 0.1}];
  módulo                             tabla en paralelo

   label = "Combinación 25";
   ListPlot[pointsy, Joined → True, PlotRange → {{0, 6}, {0, 1}},
   representación de lista  unido  verd···  rango de representación

    PlotLabel → label, PlotStyle → Directive[Red, Dashed]];
    etiqueta de representación  estilo de repre···  directiva  rojo  rayado
```

```
In[•]:= plots = {Show[generatePlotFx, generatePlotFy], Show[generatePlotGx, generatePlotGy],
                 Show[generatePlotHx, generatePlotHy], Show[generatePlotJx, generatePlotJy],
                 Show[generatePlotKx, generatePlotKy], Show[generatePlotLx, generatePlotLy],
                 Show[generatePlotEx, generatePlotEy], SpanFromLeft};
        combinedPlot = GraphicsGrid[Partition[plots, 2], Spacings → {20, 20},
                       ImageSize → Large]

        (* Añadir la leyenda *)
        Legended[combinedPlot, Placed[{"F(y1)", "F(y2)"}, Below]];

        CloseKernels[];
```

Out[•]=



Momentos

In[•]:=
```
(*Valor esperado aproximado numericamente*)
Efx = NIntegrate[x f[x, y, a = aVal, b = bVal, c = cVal, d = dVal],
```
integra numéricamente

```
    {y, 0, Infinity}, {x, 0, Infinity}];
          infinito          infinito

Efy = NIntegrate[y f[x, y, a = aVal, b = bVal, c = cVal, d = dVal],
      integra numéricamente

    {x, 0, Infinity}, {y, 0, Infinity}];
          infinito          infinito

Egx = NIntegrate[x g[x, y, a = aVal, b = bVal, c = cVal, d = dVal],
      integra numéricamente

    {y, 0, Infinity}, {x, 0, Infinity}];
          infinito          infinito

Egy = NIntegrate[y g[x, y, a = aVal, b = bVal, c = cVal, d = dVal],
      integra numéricamente

    {x, 0, Infinity}, {y, 0, Infinity}];
          infinito          infinito

Ehx = NIntegrate[x h[x, y, a = aVal, b = bVal2, d = dVal2],
      integra numéricamente

    {y, 0, Infinity}, {x, 0, Infinity}, AccuracyGoal → 4, PrecisionGoal → 4];
          infinito          infinito      objetivo de exactitud  objetivo de precisión

Ehy = NIntegrate[y h[x, y, a = aVal, b = bVal2, d = dVal2],
      integra numéricamente

    {x, 0, Infinity}, {y, 0, Infinity}, AccuracyGoal → 4, PrecisionGoal → 4];
          infinito          infinito      objetivo de exactitud  objetivo de precisión

Ejx =
  NIntegrate[x j[x, y, a = aVal2, c = cVal2, d = dVal2], {y, 0, Infinity}, {x, 0, Infinity}];
  integra numéricamente                                    infinito          infinito

Ejy =
  NIntegrate[y j[x, y, a = aVal2, c = cVal2, d = dVal2], {x, 0, Infinity}, {y, 0, Infinity}];
  integra numéricamente                                    infinito          infinito

Ekx =
  NIntegrate[x k[x, y, a = aVal2, c = cVal2, d = dVal2], {y, 0, Infinity}, {x, 0, Infinity}];
  integra numéricamente                                    infinito          infinito

Eky =
  NIntegrate[y k[x, y, a = aVal2, c = cVal2, d = dVal2], {x, 0, Infinity}, {y, 0, Infinity}];
  integra numéricamente                                    infinito          infinito

Elx = NIntegrate[x l[x, y, a = aVal, b = bVal, c = cVal, d = dVal],
      integra numéricamente

    {y, 0, Infinity}, {x, 0, Infinity}, AccuracyGoal → 4, PrecisionGoal → 4];
          infinito          infinito      objetivo de exactitud  objetivo de precisión

Ely = NIntegrate[y l[x, y, a = aVal, b = bVal, c = cVal, d = dVal],
      integra numéricamente
```

```
    {x, 0, Infinity}, {y, 0, Infinity}, AccuracyGoal → 4, PrecisionGoal → 4];
          infinito        infinito      objetivo de exactitud  objetivo de precisión

Eex = NIntegrate[x e[x, y, a = aVal, b = bVal, c = cVal, d = dVal],
      integra numéricamente

    {y, 0, Infinity}, {x, 0, Infinity}, AccuracyGoal → 1, PrecisionGoal → 0.1];
          infinito        infinito      objetivo de exactitud  objetivo de precisión

Eey = NIntegrate[y e[x, y, a = aVal, b = bVal, c = cVal, d = dVal],
      integra numéricamente

    {x, 0, Infinity}, {y, 0, Infinity}, AccuracyGoal → 1, PrecisionGoal → 0.1];
          infinito        infinito      objetivo de exactitud  objetivo de precisión
```

In[•]:=

```
(*Transformaiones inversas de (Y1,Y2) a (X1,X2)*)

g1[x_, y_] := x / (x + y);

g2[x_, y_] := x y / ((x + y)^2 (x + y + 1));

(*Resultdos Numéricos obtenidos para los valores esperados*)

resultadosTabla = TableForm[
                  forma de tabla

  {{"1", Efx, Efy, g1[Efx, Efy], g2[Efx, Efy]}, {"6", Egx, Egy, g1[Egx, Egy], g2[Egx, Egy]},
   {"8", Ehx, Ehy, g1[Ehx, Ehy], g2[Ehx, Ehy]}, {"11", Ejx, Ejy, g1[Ejx, Ejy], g2[Ejx, Ejy]},
   {"14", Ekx, Eky, g1[Ekx, Eky], g2[Ekx, Eky]}, {"19", Elx, Ely, g1[Elx, Ely],
    g2[Elx, Ely]}, {"25", Eex, Eey, g1[Eex, Eey], g2[Eex, Eey]}}, TableHeadings →
                                                                  cabeceras de tabla

   {None, {"Combinación", "E[Y1]", "E[Y2]", "h1^-1(Y_1,Y_2)", "h2^-1(Y_1,Y_2)"}}]
    ninguno                 número e   número e
```

Out[•]//TableForm=

Combinación	E[Y_1]	E[Y_2]	$h_1^{-1}(Y\_1,Y\_2)$	$h_2^{-1}(Y\_1,Y\_2)$
1	1.	2.	0.333333	0.0555556
6	1.423	1.577	0.474335	0.0623353
8	0.534093	0.575118	0.481507	0.118366
11	0.06	0.04	0.6	0.218182
14	0.381234	0.260235	0.594314	0.146884
19	0.373937	0.129468	0.742815	0.127072
25	3.10711	2.65022	0.539679	0.0367638

In[•]:=

```
(*Valor esperado de X1 distribuida Beta*)
                                  beta

EBx1[a_, b_] := a / (a + b);

(*Valor esperado de X2|X1 distribuida Beta.4p*)
                                  beta
```

```
EB4Px2x1[c_, d_, u_] := c u / (c + d);

(*Valor esperado de X1 distribuida Kumaraswamy*)

EKx1[a_, b_] := b Beta[1 + 1 / a, b];

(*Valor esperado de X2 | X1 distribuida Kumaraswamy Truncada*)

EKTx2x1[c_, d_, u_] := c d NIntegrate[x^c (1 - x^c)^(d-1), {x, 0, u}] / (1 - (1 - (u)^c)^d);

(*Valor esperado de X1 distribuida Triangular*)

ETx1[a_] := (a + 1) / 3;

(*Valor esperado de X2 | X1 distribuida Triangular*)

ETx2x1[d_, u_] := (d + u) / 3;

(*Valor esperado de X1 distribuida GT*)

EGTx1[a_, b_] := NIntegrate[x^a Exp[-x/b], {x, 0, 1}] / NIntegrate[w^(a-1) Exp[-w/b], {w, 0, 1}];

(*Valor esperado de X2 | X1 distribuida GT*)

EGTx2x1[c_, d_, u_] :=
  NIntegrate[x^c Exp[-x/d], {x, 0, u}] / NIntegrate[w^(c-1) Exp[-w/d], {w, 0, u}];

(*Valor esperado de X1 distribuida NT*)

ENTx1[a_, b_] :=
  NIntegrate[x Exp[-1/(2 b) (x - a)^2], {x, 0, 1}] / ((CDF[NormalDistribution[0, 1], (1 - a)/Sqrt[b]] -
      CDF[NormalDistribution[0, 1], (-a)/Sqrt[b]]) Sqrt[2 Pi b]);

(*Valor esperado de X2 | X1 distribuida NT*)

ENTx2x1[c_, d_, u_] :=
  NIntegrate[x Exp[-1/(2 d) (x - c)^2], {x, 0, u}] / ((CDF[NormalDistribution[0, 1], (u - c)/Sqrt[d]] -
      CDF[NormalDistribution[0, 1], (-c)/Sqrt[d]]) Sqrt[2 Pi d]);
```

In[•]:=
```
(*Comparación entre resultados numéricos obtenidos
 y los valores perados marginales y condicionales*)
resultadosTabla =
 N[TableForm[{{"1", Efx, Efy, g1[Efx, Efy], g2[Efx, Efy], EBx1[aVal, bVal],
   EB4Px2x1[cVal, dVal, EBx1[aVal, bVal] (1 - EBx1[aVal, bVal])]},
  {"6", Egx, Egy, g1[Egx, Egy], g2[Egx, Egy], EKx1[aVal, bVal],
   EB4Px2x1[cVal, dVal, EKx1[aVal, bVal] (1 - EKx1[aVal, bVal])]},
  {"8", Ehx, Ehy, g1[Ehx, Ehy], g2[Ehx, Ehy], EKx1[aVal, bVal2],
   ETx2x1[dVal2, EKx1[aVal, bVal2] (1 - EKx1[aVal, bVal2])]},
  {"11", Ejx, Ejy, g1[Ejx, Ejy], g2[Ejx, Ejy], ETx1[aVal2],
   EB4Px2x1[cVal2, dVal2, ETx1[aVal2] (1 - ETx1[aVal2])]},
  {"14", Ekx, Eky, g1[Ekx, Eky], g2[Ekx, Eky],
   ETx1[aVal2], EGTx2x1[cVal2, dVal2, ETx1[aVal2] (1 - ETx1[aVal2])]},
  {"19", Elx, Ely, g1[Elx, Ely], g2[Elx, Ely], EGTx1[aVal, bVal],
   EGTx2x1[cVal, dVal, EGTx1[aVal, bVal] (1 - EGTx1[aVal, bVal])]},
  {"25", Eex, Eey, g1[Eex, Eey], g2[Eex, Eey], ENTx1[aVal, bVal],
   ENTx2x1[cVal, dVal, ENTx1[aVal, bVal] (1 - ENTx1[aVal, bVal])]}},
 TableHeadings → {None, {"Combinación", "E[Y1]", "E[Y2]", "h1^-1(E[Y1],E[Y2])",
   "h2^-1(E[Y1],E[Y2])", "E[X1]", "E[X2|X1=E[X1]]"}}, 5]
```

Out[•]//TableForm=

Combinación	$E[Y_1]$	$E[Y_2]$	$h_1^{-1}(E[Y_1],E[Y_2])$	$h_2^{-1}(E[Y_1],E[Y_2])$	$E[X_1]$	E[X
1	1.	2.	0.333333	0.0555556	0.33333	0.(
6	1.423	1.577	0.474335	0.0623353	0.47433	0.(
8	0.534093	0.575118	0.481507	0.118366	0.47433	0.1
11	0.06	0.04	0.6	0.218182	0.6	0.2
14	0.381234	0.260235	0.594314	0.146884	0.6	0.1
19	0.373937	0.129468	0.742815	0.127072	0.743663	0.1
25	3.10711	2.65022	0.539679	0.0367638	0.534431	0.1

In[•]:=
```
(*Covarianza aproximada numericamente*)
Cf = NIntegrate[x y f[x, y, a = aVal, b = bVal, c = cVal, d = dVal],
                integra numéricamente
   {y, 0, Infinity}, {x, 0, Infinity}, AccuracyGoal → 4, PrecisionGoal → 4];
          infinito      infinito      objetivo de exactitud  objetivo de precisión
Cg = NIntegrate[x y g[x, y, a = aVal, b = bVal, c = cVal, d = dVal],
                integra numéricamente
   {y, 0, Infinity}, {x, 0, Infinity}];
          infinito      infinito
Ch = NIntegrate[x y h[x, y, a = aVal, b = bVal2, d = dVal2],
                integra numéricamente
   {y, 0, Infinity}, {x, 0, Infinity}, AccuracyGoal → 1, PrecisionGoal → 4];
          infinito      infinito      objetivo de exactitud  objetivo de precisión
Cj = NIntegrate[x y j[x, y, a = aVal2, c = cVal2, d = dVal2],
                integra numéricamente
   {y, 0, Infinity}, {x, 0, Infinity}];
          infinito      infinito
Ck = NIntegrate[x y k[x, y, a = aVal2, c = cVal2, d = dVal2],
                integra numéricamente
   {y, 0, Infinity}, {x, 0, Infinity}];
          infinito      infinito
Cl = NIntegrate[x y l[x, y, a = aVal, b = bVal, c = cVal, d = dVal],
                integra numéricamente
   {y, 0, Infinity}, {x, 0, Infinity}];
          infinito      infinito
Ce = NIntegrate[x y e[x, y, a = aVal, b = bVal, c = cVal, d = dVal],
                integra numéricamente
   {x, 0, Infinity}, {y, 0, Infinity}, AccuracyGoal → 1, PrecisionGoal → 0.1];
          infinito      infinito      objetivo de exactitud  objetivo de precisión
```

In[•]:=
```
NIntegrate[x y e[x, y, a = aVal, b = bVal, c = cVal, d = dVal],
           integra numéricamente
  {x, 0, Infinity}, {y, 0, Infinity}, AccuracyGoal → 1, PrecisionGoal → 0.1]
        infinito      infinito      objetivo de exactitud  objetivo de precisión
```

Out[•]= 1.22186×10^{64}

In[•]:=
```
(*Covarianza*)
resultadosTabla =
 N[TableForm[{{"1", Efx, Efy, g1[Efx, Efy], g2[Efx, Efy], EBx1[aVal, bVal],
                                                    forma de tabla
    EB4Px2x1[cVal, dVal, EBx1[aVal, bVal](1 - EBx1[aVal, bVal])], Cf, Cf - Efx Efy},
   {"6", Egx, Egy, g1[Egx, Egy], g2[Egx, Egy], EKx1[aVal, bVal],
    EB4Px2x1[cVal, dVal, EKx1[aVal, bVal](1 - EKx1[aVal, bVal])], Cg, Cg - Egx Egy},
   {"8", Ehx, Ehy, g1[Ehx, Ehy], g2[Ehx, Ehy], EKx1[aVal, bVal2],
    ETx2x1[dVal2, EKx1[aVal, bVal2](1 - EKx1[aVal, bVal2])], Ch, Ch - Ehx Ehy},
   {"11", Ejx, Ejy, g1[Ejx, Ejy], g2[Ejx, Ejy], ETx1[aVal2],
    EB4Px2x1[cVal2, dVal2, ETx1[aVal2](1 - ETx1[aVal2])], Cj, Cj - Ejx Ejy},
   {"14", Ekx, Eky, g1[Ekx, Eky], g2[Ekx, Eky], ETx1[aVal2],
    EGTx2x1[cVal2, dVal2, ETx1[aVal2](1 - ETx1[aVal2])], Ck, Ck - Ekx Eky},
   {"19", Elx, Ely, g1[Elx, Ely], g2[Elx, Ely], EGTx1[aVal, bVal],
    EGTx2x1[cVal, dVal, EGTx1[aVal, bVal](1 - EGTx1[aVal, bVal])], Cl, Cl - Elx Ely},
   {"25", Eex, Eey, g1[Eex, Eey], g2[Eex, Eey], ENTx1[aVal, bVal],
    ENTx2x1[cVal, dVal, ENTx1[aVal, bVal](1 - ENTx1[aVal, bVal])], Ce, Ce - Eex Eey}},
  TableHeadings → {None, {"Combinación", "E[Y1]", "E[Y2]", "h1^-1(E[Y1],E[Y2])",
                  cabeceras de tabla   ninguno                número e  número e  núm··  número e
    "h2^-1(E[Y1],E[Y2])", "E[X1]", "E[X2|X1=E[X1]]", "E[Y1Y2]", "Cov(Y1,Y2)"}}, 5]
                          núm··  número e  número e  númer···  número e  número e
```

Out[•]//TableForm=

Combinación	$E[Y_1]$	$E[Y_2]$	$h_1^{-1}(E[Y_1],E[Y_2])$	$h_2^{-1}(E[Y_1],E[Y_2])$	$E[X_1]$	E[X
1	1.	2.	0.333333	0.0555556	0.33333	0.(
6	1.423	1.577	0.474335	0.0623353	0.47433	0.(
8	0.534093	0.575118	0.481507	0.118366	0.47433	0.1
11	0.06	0.04	0.6	0.218182	0.6	0.2
14	0.381234	0.260235	0.594314	0.146884	0.6	0.1
19	0.373937	0.129468	0.742815	0.127072	0.743663	0.1
25	3.10711	2.65022	0.539679	0.0367638	0.534431	0.1