An Copula Experiment in Geodesy

----- MOPI permanent observations ----- positive dependence ------ chopped (Xi,Yi) with extremal Yi's

---

Initial settings

```
(* ----- read in packages ----- *)

<< Statistics`ContinuousDistributions`
<< Statistics`MultiDescriptiveStatistics`
<< Statistics`MultinormalDistribution`

<< Graphics`Graphics`
<< Graphics`Graphics3D`

<< Statistics`NonlinearFit`
<< Statistics`DataManipulation`

<< Statistics`StatisticsPlots` (* Mathematica 5 *)

SetDirectory["d:\\math\\Analyza CR\\copula"];(* at home *)

    SetDirectory[d:\documents\phd\copula\program];(* at KMaDG *)

(* ----- settings ----- *)

SetOptions[ListPlot, PlotJoined → True, PlotRange → All, DisplayFunction → Identity];
SetOptions[{Histogram, Plot, QuantilePlot, ContourPlot, Plot3D},
  PlotRange → All, DisplayFunction → Identity];

Off[General::spell1];

(* ----- user functions ----- *)

fShow[plot__, options___] := Show[plot, DisplayFunction :→ $DisplayFunction, options];

fDShow[plot__] := Show[plot, DisplayFunction :→ Identity];

(* ----- read in the data ----- *)

XY = ReadList["data\\mopi_nt.dat", Table[Number, {2}]];

{X, Y} = Transpose[XY];

Y = -Y;
Y = Extract[Y, tmpOrdY = Partition[Drop[Drop[Ordering[Y], 1], 0], 1]];
X = Extract[X, tmpOrdY];
XY = Transpose[{X, Y}];
```

```
n = Length[XY]
```

727

First look

```
GraphicsArray[{ListPlot[X], ListPlot[Y]}] // fShow;
```



```
ListPlot[Transpose[{Y, X}], PlotJoined → False] // fShow;
```



```
{corr = Correlation[X, Y], ρ = SpearmanRankCorrelation[X, Y], KendallRankCorrelation[X, Y]} // N
```

{0.322441, 0.317775, 0.22424}

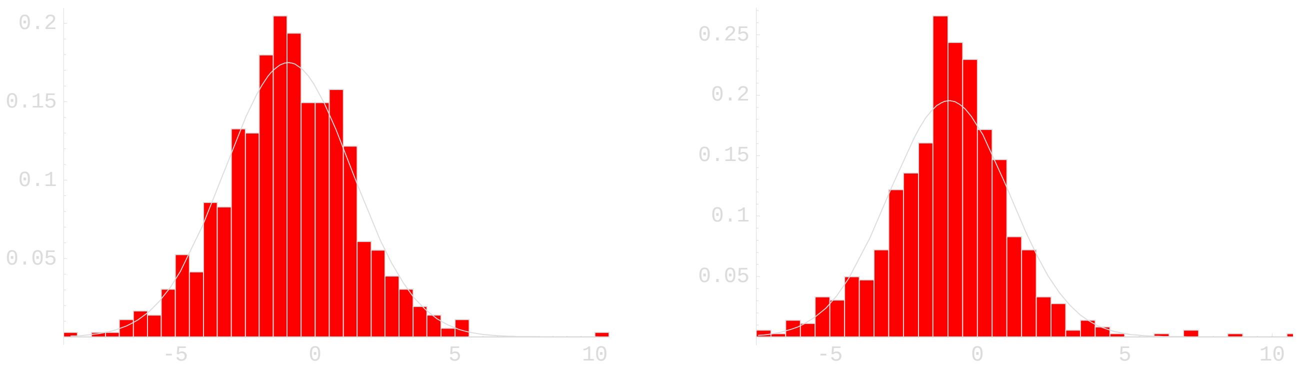
Usual nonparametric estimate of Kendall's tau = $\binom{n}{2}^{-1} \sum_{i<j} \text{Sign}[(X_i - X_j)(Y_i - Y_j)]$

```
τ = Binomial[n, 2]^(-1) Sum[Sum[Sign[(X[[i]] - X[[j]]) (Y[[i]] - Y[[j]])], {j, 1, i - 1}], {i, 1, n}] // N
```

0.232201

```
QuantilePlot[X, Y] // fShow; (* Mathematica 5 *)
```

```
GraphicsArray[
  Show[
         Histogram[#, HistogramScale → 1],
         Plot[PDF[NormalDistribution[pM = Mean[#], pSD = StandardDeviation[#]], x],
      {x, Min[#], Max[#]}],
         PlotLabel → StyleForm[{pM, pSD}, FontColor → RGBColor[0, 1, 0]]
    ] & /@ {X, Y}
 ] // fShow; Clear[pM, pSD]
```




```
pμ = pσ = Table[0, {2}];

{Median[X], Mode[X], pμ[[1]] = Mean[X],
 pσ[[1]] = StandardDeviation[X], Skewness[X], KurtosisExcess[X]}

{-0.975, {-2.37, -0.58}, -1.00533, 2.3343, -0.0378599, 0.852742}

{Median[Y], Mode[Y], pμ[[2]] = Mean[Y],
 pσ[[2]] = StandardDeviation[Y], Skewness[Y], KurtosisExcess[Y]}

{-0.9, -1.13, -1.00665, 2.13659, 0.0110652, 3.06612}
```

Nonparametric estimation of copula parameter

{ procedure by Genest&MacKey(1986,1993); described in Frees&Valdez(1998) and Abid&Naifar(2005) }

■ Nonparametric estimate Kn

```
(* unobserved variable Z=H (X,Y) *)
fZ[i_] := Sum[If[X[[j]] < X[[i]] && Y[[j]] < Y[[i]], 1., 0], {j, n}] / (n - 1)
Z = Table[fZ[i], {i, n}];

(* distribution function of Z *)
fKn[z_] := Sum[If[Z[[i]] ≤ z, 1, 0], {i, n}] / n
Kn = Table[fKn[z], {z, 0, 1, 1 / n}];
```

```
GraphicsArray[{ListPlot[Z], ListPlot[Kn]}] // fShow;
```


Parametric estimate Kϕ

$$K_\phi(z) = z - \frac{\phi(z)}{\phi'(z)} \quad \text{using relation} \quad \tau = 1 + 4\int_0^1 \frac{\phi(t)}{\phi'(t)}\, dt$$

Procedure: $\tau \to \theta \to \phi \to K_\phi$.

Independence copula

```
fKi[t_] := If[t ≠ 0, t (1 - Log[t]), 0];
Ki = Table[fKi[z], {z, 0, 1, 1 / n}];
```

Gumbel copula

```
NSolve[τ == (t - 1) / t];
θg = t /. %[[1]]
```

```
1.30242
```

```
fKg[t_] := If[t ≠ 0, t - t Log[t] / θg, 0];
Kg = Table[fKg[z], {z, 0, 1, 1 / n}];
```

Clayton copula

```
NSolve[τ == t / (t + 2)];
θc = t /. %[[1]]
```

```
0.604847
```

```
fKc[t_] := t - (t^(θc+1) - t) / θc;
Kc = Table[fKc[z], {z, 0, 1, 1 / n}];
```

Frank copula

```
fD1[x_] := 1 / x Integrate[t / (Exp[t] - 1), {t, 0, x}];
```

```
FindRoot[τ == 1 + 4 / t (fD1[t] - 1), {t, 2.2}];
θf = Re[t /. %]
```

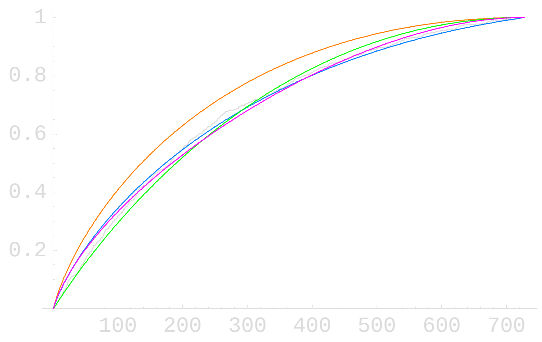
```
2.18657
```

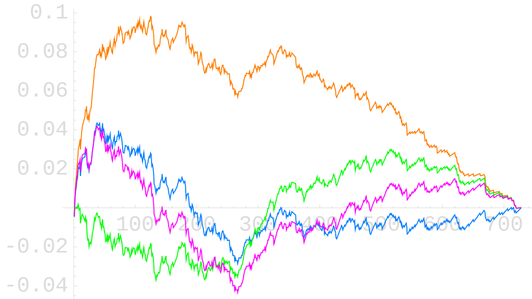
```
fKf[t_] := If[t == 0, 0, t - Log[(Exp[-θf t] - 1) / (Exp[-θf] - 1)] * (Exp[θf t] - 1) / θf]
Kf = Table[fKf[z], {z, 0, 1, 1 / n}];
```

▪ Comparing K's

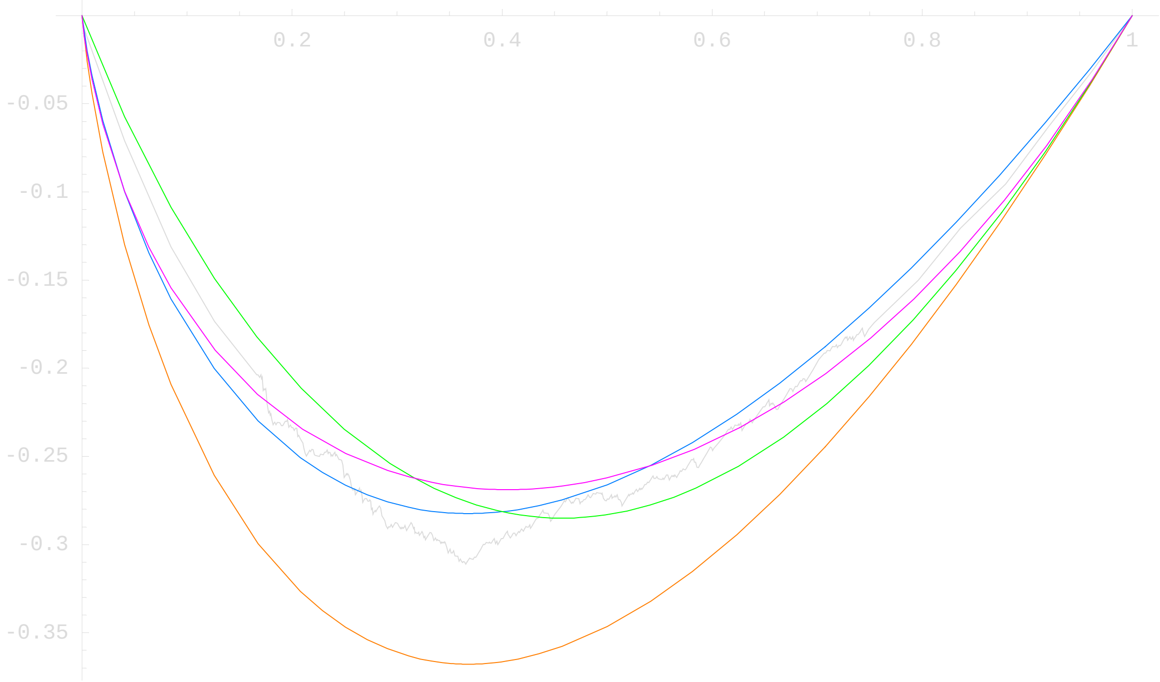
▫ Graphically

```
GraphicsArray[{
   Show[
    ListPlot[Kn],
    ListPlot[Ki, PlotStyle → RGBColor[1, 0.5, 0]],
    ListPlot[Kg, PlotStyle → RGBColor[0, 0.5, 1]],
    ListPlot[Kc, PlotStyle → RGBColor[0, 1, 0]],
    ListPlot[Kf, PlotStyle → RGBColor[1, 0, 1]]]
   ,
   Show[
    ListPlot[Ki - Kn, PlotStyle → RGBColor[1, 0.5, 0]],
    ListPlot[Kg - Kn, PlotStyle → RGBColor[0, 0.5, 1]],
    ListPlot[Kc - Kn, PlotStyle → RGBColor[0, 1, 0]],
    ListPlot[Kf - Kn, PlotStyle → RGBColor[1, 0, 1]]]
  }] // fShow;
```





```
Show[
  Plot[z - fKn[z], {z, 0, 1}],
  Plot[z - fKi[z], {z, 0, 1}, PlotStyle → RGBColor[1, 0.5, 0]],
  Plot[z - fKg[z], {z, 0, 1}, PlotStyle → RGBColor[0, 0.5, 1]],
  Plot[z - fKc[z], {z, 0, 1}, PlotStyle → RGBColor[0, 1, 0]],
  Plot[z - fKf[z], {z, 0, 1}, PlotStyle → RGBColor[1, 0, 1]]] // fShow;
```




```
GraphicsArray[{
    {QuantilePlot[Kn, Ki], QuantilePlot[Kn, Kg]},
    {QuantilePlot[Kn, Kc], QuantilePlot[Kn, Kf]}}] // fShow;
```


Numerically

```
Norm[# - Kn] & /@ {Ki, Kg, Kc, Kf} // N
```

```
{1.70604, 0.429077, 0.544439, 0.479107}
```

Semi-parametric estimation of copula parameter

■ Distribution functions

Empirical marginal distribution functions

$$\text{CDF}(x) = P(X \leq x)$$

```
(* rescaled empirical marginal distribution functions CDF=n/(n+1) CDF *)
fCDF1empir[x_] := Sum[If[X〚i〛 ≤ x, 1, 0], {i, 1, n}] / (n + 1)
fCDF2empir[y_] := Sum[If[Y〚i〛 ≤ y, 1, 0], {i, 1, n}] / (n + 1)
```

Empirical joint CDF

$$\text{CDF}(x, y) = P(X \leq x, Y \leq y)$$

```
Clear[i, j, k, odX, odY, n1, n2, freq, freqTab, ffreq]

odX = Transpose[Frequencies[X]]〚2〛;  (* ordered discrete values of X *)
odY = Transpose[Frequencies[Y]]〚2〛;

n0 = Length[dataX];
n1 = Length[odX];
n2 = Length[odY];

freq = Transpose[Reverse[Transpose[Frequencies[Transpose[{dataX, dataY}]]]]];
freqTab = Table[0, {i, n1}, {j, n2}];

Dimensions[freq]

{728, 2}

For[i = 1, i ≤ n1, i++,
  f1[odX〚i〛] = i];
For[j = 1, j ≤ n2, j++,
  f2[odY〚j〛] = j];

For[k = 1, k ≤ Length[freq], k++,
  freqTab〚 f1[freq〚k, 1, 1〛], f2[freq〚k, 1, 2〛]〛 = freq〚k, 2〛];

Clear[i, j, k]

(* ---!!!--- long computation (approx 2 h) ---!!!---- *)
cumuTab = Table[ Sum[ Sum[ freqTab〚ii, jj〛, {jj, j}], {ii, i}], {i, n1}, {j, n2}];

CDFempir = Table[{ odX〚i〛, odY〚j〛, cumuTab〚i, j〛 / n}, {i, n1}, {j, n2}];

Clear[i, j, k, odX, odY, n1, n2, freq, freqTab, ffreq, cumuTab]

    (* storing data - for activation change the cell style to 'Input' *)
    CDFempir >> CDFempir.txt
    Dimensions[CDFempir = (<< CDFempir.txt)]

{500, 461, 3}
```

```
(* margin of empirical joint distribution function *)
ListPlot[Transpose[Take[Transpose[CDFempir⟦ Length[CDFempir] ⟧], -2]]] // fShow;
```



```
Clear[i, j, k, odX, odY, n1, n2, freq, freqTab, ffreq]
```

Empirical copula

```
odU = Table[fCDF1empir[odX⟦i⟧], {i, 1, n1}];
odV = Table[fCDF2empir[odY⟦j⟧], {j, 1, n2}];
```

```
fCempir[listJoinedCDF_, fMarginalCDF1_, fMarginalCDF2_] :=
     Table[
          {fMarginalCDF1[listJoinedCDF⟦i, j, 1⟧],
           fMarginalCDF2[listJoinedCDF⟦i, j, 2⟧],
           listJoinedCDF⟦i, j, 3⟧},
              {i, 1, Dimensions[listJoinedCDF]⟦1⟧}, {j, 1, Dimensions[listJoinedCDF]⟦2⟧}
         ]
```

```
(* ---!!!--- long computation (approx 0.5 h) ---!!!---- *)
tmp = TimeUsed[];
Dimensions[Cempir = fCempir[CDFempir, fCDF1empir, fCDF2empir]]
(TimeUsed[] - tmp) / 60
Clear[tmp];
```

```
{500, 461, 3}
```

```
24.8392
```

```
(* storing and reloading data  *)
Cempir >> Cempir.txt
Dimensions[Cempir = (<< "Cempir.txt")]
```

```
(* one margin of the empirical copula *)
ListPlot[Transpose[Take[Transpose[Cempir[[Length[CDFempir]]]], -2]]] // fShow;
```


Archimedean copula

$$C(u, v) = \phi^{-1}[\phi(u) + \phi(v)]$$

```
fCg[u_, v_] = e^(-((-Log[u])^θ + (-Log[v])^θ)^(1/θ));
fCc[u_, v_] = (u^-θ + v^-θ - 1)^(-1/θ);
fCf[u_, v_] = -Log[((e^(-θ v) - 1) (e^(-θ u) - 1))/(e^-θ - 1) + 1]/θ;
```

Density functions:

```
fci[u_, v_] = D[fCi[u, v], u, v];
fcg[u_, v_] = Simplify[D[fCg[u, v], u, v]];
fcc[u_, v_] = D[fCc[u, v], u, v];
fcf[u_, v_] = D[fCf[u, v], u, v];
```

Multinormal distribution (estimated)

```
fCDFmultinorm[x1_, x2_] = CDF[MultinormalDistribution[Mean[XY], CovarianceMatrix[XY]], {x1, x2}]

CDF[MultinormalDistribution[{-1.00533, -1.00665},
  {{5.44896, 1.83046}, {1.83046, 4.56503}}], {x1, x2}]

(* ---!!!--- long computation (approx 1 h) ---!!!---- *)
tmp = TimeUsed[];
CDFmultinorm = Table[ {
                CDFempir[[i, j, 1]],
                CDFempir[[i, j, 2]],
                fCDFmultinorm[CDFempir[[i, j, 1]], CDFempir[[i, j, 2]]]},
           {i, 1, Dimensions[CDFempir][[1]]}, {j, 1, Dimensions[CDFempir][[2]]}];
(TimeUsed[] - tmp) / 60
Clear[tmp];

65.0357

(* storing data *)
CDFmultinorm >> CDFmultinorm.txt
Dimensions[CDFmultinorm = (<< CDFmultinorm.txt)]

{500, 461, 3}

CDFmultinorm = (<< "CDFmultinorm.txt");
```

```
ListPlot[Transpose[Take[Transpose[CDFmultinorm[[n1]]], -2]]] // fShow;
```



```
(* storing residuals *)
Flatten[Table[CDFempir[[i, j, 3]] - CDFmultinorm[[i, j, 3]], {i, 1, n1}, {j, 1, n2}]] >>
  residCDFmn.txt;
```

■ Fitting Archimedean to empirical copula

▫ Pseudo Log-Likelihood

{ procedure by Genest&Rivest(1995); described in Frees&Valdez(1998), Abid&Naifar(2005), Durrleman(2000) }

```
(* pseudo log-likelihood functions *)
fLg[θ_] = Sum[Log[fcg[ fCDF1empir[X[[k]]], fCDF2empir[Y[[k]]] ]], {k, 1, n}];
fLc[θ_] = Sum[Log[fcc[ fCDF1empir[X[[k]]], fCDF2empir[Y[[k]]] ]], {k, 1, n}];
fLf[θ_] = Sum[Log[fcf[ fCDF1empir[X[[k]]], fCDF2empir[Y[[k]]] ]], {k, 1, n}];
```

```
{pLLg = FindMaximum[fLg[θ], {θ, 1}],
  pLLc = FindMaximum[fLc[θ], {θ, 0.1}, AccuracyGoal → 7],
  pLLf = FindMaximum[fLf[θ], {θ, 0.1}, AccuracyGoal → 7]} // TableForm
```

52.8038	θ → 1.29935
50.7376	θ → 0.534041
45.435	θ → 2.29133

```
Transpose[{pLLg, pLLc, pLLf}];
{θ1g, θ1c, θ1f} = ((θ /. #) & /@ %[[2]]);
pAIC = -2 %%[[1]] + 2
```

{-106.235, -109.035, -90.7313}

```
(* storing residuals *)
```

```
pfCg[u_, v_] = fCg[u, v] /. θ -> θ1g;
Flatten[Table[Cempir[[i, j, 3]] - pfCg[odU[[i]] , odV[[j]] ], {i, 1, n1}, {j, 1, n2}]] >>
  residLLCg.txt;
Clear[
  pfCg];
```

```
pfCc[u_, v_] = fCc[u, v] /. θ -> θ1c;
Flatten[Table[Cempir[[i, j, 3]] - pfCc[odU[[i]] , odV[[j]] ], {i, 1, n1}, {j, 1, n2}]] >>
  residLLCc.txt;
Clear[
  pfCc];
```

```
pfCf[u_, v_] = fCf[u, v] /. θ -> θ1f;
Flatten[Table[Cempir⟦i, j, 3⟧ - pfCf[odU⟦i⟧ , odV⟦j⟧ ], {i, 1, n1}, {j, 1, n2}]] >>
  residLLCf.txt;
Clear[
  pfCf];
```

NonlinearRegress

Gumbel

```
NonlinearRegress[Flatten[Cempir, 1], ⅇ^-((-Log[u])^θ+(-Log[v])^θ)^(1/θ), {u, v}, {θ, 1.3},
 RegressionReport → {BestFitParameters, EstimatedVariance, ParameterCITable}]
θ2g = (θ /. (BestFitParameters /. %));
```

```
{BestFitParameters → {θ → 1.30308}, EstimatedVariance → 0.0000593832,
 ParameterCITable →     Estimate    Asymptotic SE    CI
                    θ   1.30308     0.000222035      {1.30264, 1.30351}}
```

```
(* storing residuals *)
NonlinearRegress[Flatten[Cempir, 1], ⅇ^-((-Log[u])^θ+(-Log[v])^θ)^(1/θ),
  {u, v}, {θ, 1.3}, RegressionReport → {FitResiduals}];
(FitResiduals /. %) >> residNRCg.txt;
```

```
1.3030765492605614`
```

Clayton

```
NonlinearRegress[Flatten[Cempir, 1], (u^-θ + v^-θ - 1)^(-1/θ), {u, v}, {θ, 0.61},
 RegressionReport → {BestFitParameters, EstimatedVariance, ParameterCITable}]
θ2c = (θ /. (BestFitParameters /. %));
```

```
{BestFitParameters → {θ → 0.559452}, EstimatedVariance → 0.0000738831,
 ParameterCITable →     Estimate    Asymptotic SE    CI
                    θ   0.559452    0.000454426      {0.558561, 0.560342}}
```

```
(* storing residuals *)
NonlinearRegress[Flatten[Cempir, 1], (u^-θ + v^-θ - 1)^(-1/θ),
  {u, v}, {θ, 0.5}, RegressionReport → {FitResiduals}];
(FitResiduals /. %) >> residNRCc.txt;
```

```
0.5594515322670758`
```

Frank

```
NonlinearRegress[Flatten[Cempir, 1], -Log[((ⅇ^(-θ v) - 1) (ⅇ^(-θ u) - 1))/(ⅇ^-θ - 1) + 1]/θ, {u, v}, {θ, 2.2},
 RegressionReport → {BestFitParameters, EstimatedVariance, ParameterCITable}]
θ2f = (θ /. (BestFitParameters /. %));
```

```
{BestFitParameters → {θ → 2.10342}, EstimatedVariance → 0.0000561715,
 ParameterCITable →     Estimate    Asymptotic SE    CI
                    θ   2.10342     0.00124743       {2.10097, 2.10586}}
```

```
(*  storing residuals *)
NonlinearRegress[Flatten[Cempir, 1], -Log[((e^(-θ v) - 1) (e^(-θ u) - 1))/(e^(-θ) - 1) + 1]/θ,
  {u, v}, {θ, 2.2}, RegressionReport → {FitResiduals}];
(FitResiduals /. %) >> residNRCf.txt;
```

2.1034184856358658`

Fitting linear convex combinations

```
θ2g = 1.3030765492605614;
θ2c = 0.5594515322670758;
θ2f = 2.1034184856358658;
```

Clayton-Gumbel

```
NonlinearRegress[Flatten[Cempir, 1], α * (u^-θ2c + v^-θ2c - 1)^(-1/θ2c) + (1 - α) * e^-((-Log[u])^θ2g+(-Log[v])^θ2g)^(1/θ2g),
 {u, v}, {α, 0.5}, RegressionReport → {BestFitParameters, EstimatedVariance, ParameterCITable}]
αcg = (α /. (BestFitParameters /. %));
```

{BestFitParameters → {α → 0.450664}, EstimatedVariance → 0.0000295381,

ParameterCITable →

	Estimate	Asymptotic SE	CI
α	0.450664	0.000933841	{0.448833, 0.452494}

}

```
NonlinearRegress[Flatten[Cempir, 1], α * (u^-θ2c + v^-θ2c - 1)^(-1/θ2c) + (1 - α) * e^-((-Log[u])^θ2g+(-Log[v])^θ2g)^(1/θ2g),
  {u, v}, {α, 0.5}, RegressionReport → {FitResiduals}];
(FitResiduals /. %) >> residNRCcg.txt;
```

Clayton-Frank

```
NonlinearRegress[Flatten[Cempir, 1],
 α * (u^-θ2c + v^-θ2c - 1)^(-1/θ2c) + (1 - α) * (-Log[((e^(-θ2f v) - 1) (e^(-θ2f u) - 1))/(e^(-θ2f) - 1) + 1]/θ2f), {u, v}, {α, 0.5},
 RegressionReport → {BestFitParameters, EstimatedVariance, ParameterCITable}]
αcf = (α /. (BestFitParameters /. %));
```

{BestFitParameters → {α → 0.371375}, EstimatedVariance → 0.0000466758,

ParameterCITable →

	Estimate	Asymptotic SE	CI
α	0.371375	0.00171498	{0.368013, 0.374736}

}

```
NonlinearRegress[Flatten[Cempir, 1],
  α * (u^-θ2c + v^-θ2c - 1)^(-1/θ2c) + (1 - α) * (-Log[((e^(-θ2f v) - 1) (e^(-θ2f u) - 1))/(e^(-θ2f) - 1) + 1]/θ2f),
  {u, v}, {α, 0.5}, RegressionReport → {FitResiduals}];
(FitResiduals /. %) >> residNRCcf.txt;
```

Frank-Gumbel

```
NonlinearRegress[Flatten[Cempir, 1],
 α * (-Log[((e^(-θ2f v) - 1) (e^(-θ2f u) - 1))/(e^(-θ2f) - 1) + 1]/θ2f) + (1 - α) * e^(-((-Log[u])^θ2g + (-Log[v])^θ2g)^(1/θ2g)), {u, v}, {α, 0.5},
 RegressionReport → {BestFitParameters, EstimatedVariance, ParameterCITable}]
αfg = (α /. (BestFitParameters /. %));
```

```
{BestFitParameters → {α → 0.554828}, EstimatedVariance → 0.000050367,
 ParameterCITable →     Estimate    Asymptotic SE    CI
                    α   0.554828    0.00273141       {0.549475, 0.560182}}
```

```
NonlinearRegress[Flatten[Cempir, 1],
 α * (-Log[((e^(-θ2f v) - 1) (e^(-θ2f u) - 1))/(e^(-θ2f) - 1) + 1]/θ2f) + (1 - α) * e^(-((-Log[u])^θ2g + (-Log[v])^θ2g)^(1/θ2g)),
 {u, v}, {α, 0.5}, RegressionReport → {FitResiduals}];
(FitResiduals /. %) >> residNRCfg.txt;
```

Frank-Independence

```
NonlinearRegress[Flatten[Cempir, 1], α * (-Log[((e^(-θ2f v) - 1) (e^(-θ2f u) - 1))/(e^(-θ2f) - 1) + 1]/θ2f) + (1 - α) * (u * v), {u, v},
 {α, 0.5}, RegressionReport → {BestFitParameters, EstimatedVariance, ParameterCITable}]
αfi = (α /. (BestFitParameters /. %));
```

```
{BestFitParameters → {α → 0.989971}, EstimatedVariance → 0.0000560856,
 ParameterCITable →     Estimate    Asymptotic SE    CI
                    α   0.989971    0.000534023      {0.988924, 0.991018}}
```

```
NonlinearRegress[Flatten[Cempir, 1], α * (-Log[((e^(-θ2f v) - 1) (e^(-θ2f u) - 1))/(e^(-θ2f) - 1) + 1]/θ2f) + (1 - α) * (u * v),
 {u, v}, {α, 0.5}, RegressionReport → {FitResiduals}];
(FitResiduals /. %) >> residNRCfi.txt;
```

Clayton-Independence

```
NonlinearRegress[Flatten[Cempir, 1], α * (u^(-θ2c) + v^(-θ2c) - 1)^(-1/θ2c) + (1 - α) * (u * v), {u, v},
 {α, 0.5}, RegressionReport → {BestFitParameters, EstimatedVariance, ParameterCITable}]
αci = (α /. (BestFitParameters /. %));
```

```
{BestFitParameters → {α → 0.984211}, EstimatedVariance → 0.0000736724,
 ParameterCITable →     Estimate    Asymptotic SE    CI
                    α   0.984211    0.00061499       {0.983006, 0.985417}}
```

```
NonlinearRegress[Flatten[Cempir, 1], α * (u^(-θ2c) + v^(-θ2c) - 1)^(-1/θ2c) + (1 - α) * (u * v),
 {u, v}, {α, 0.5}, RegressionReport → {FitResiduals}];
(FitResiduals /. %) >> residNRCci.txt;
```

Gumbel-Independence

```
NonlinearRegress[Flatten[Cempir, 1], α * E^(-((-Log[u])^θ2g + (-Log[v])^θ2g)^(1/θ2g)) + (1 - α) * (u * v), {u, v},
 {α, 0.5}, RegressionReport → {BestFitParameters, EstimatedVariance, ParameterCITable}]
αgi = (α /. (BestFitParameters /. %));
```

```
{BestFitParameters → {α → 0.983729}, EstimatedVariance → 0.0000591552,
 ParameterCITable →     Estimate     Asymptotic SE     CI
                    α   0.983729     0.000545987       {0.982659, 0.984799} }
```

```
NonlinearRegress[Flatten[Cempir, 1], α * E^(-((-Log[u])^θ2g + (-Log[v])^θ2g)^(1/θ2g)) + (1 - α) * (u * v),
  {u, v}, {α, 0.5}, RegressionReport → {FitResiduals}];
(FitResiduals /. %) >> residNRCgi.txt;
```

■ Comparing residuals

▫ Reading residuals from file

▫ L-2 norm

```
Norm[residCDFmn]
```

```
12.3631
```

```
Norm[#] & /@ {residLLCg, residLLCc, residLLCf}
```

```
{3.69999, 4.12752, 3.80591}
```

```
Norm[#] & /@ {residNRCg, residNRCc, residNRCf}
```

```
{3.6997, 4.12674, 3.59826}
```

```
Norm[#] & /@ {residNRCgi, residNRCci, residNRCfi}
```

```
{3.69259, 4.12085, 3.59551}
```

```
Norm[#] & /@ {residNRCcg, residNRCcf, residNRCfg}
```

```
{2.60931, 3.28005, 3.40728}
```

▫ L-1 norm

```
Norm[residCDFmn, 1]
```

```
4705.7
```

```
Norm[#, 1] & /@ {residLLCg, residLLCc, residLLCf}
```

```
{1290.88, 1480.61, 1436.27}
```

```
Norm[#, 1] & /@ {residNRCg, residNRCc, residNRCf}
```

```
{1287.48, 1471.8, 1335.41}
```

```
Norm[#, 1] & /@ {residNRCgi, residNRCci, residNRCfi}
```

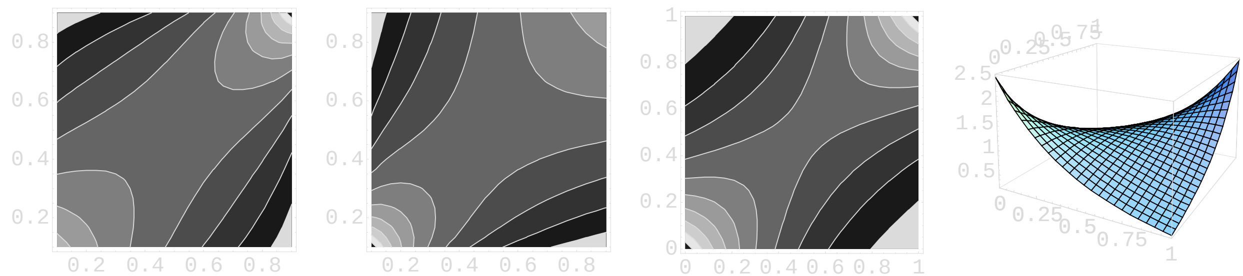
```
{1269.13, 1446.4, 1329.59}
```

```
Norm[#, 1] & /@ {residNRCcg, residNRCcf, residNRCfg}
```

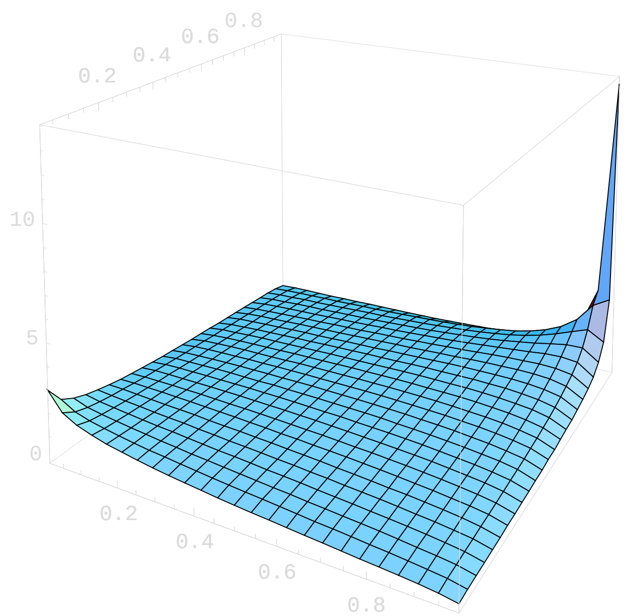
```
{990.576, 1247.53, 1191.22}
```

Visualization

```
GraphicsArray[{
    ContourPlot[fcg[u, v] /. θ → θg, {u, 0.1, 0.9}, {v, 0.1, 0.9}],
    ContourPlot[fcc[u, v] /. θ → θc, {u, 0.1, 0.9}, {v, 0.1, 0.9}],
    ContourPlot[fcf[u, v] /. θ → θf, {u, 0, 1}, {v, 0, 1}],
    Plot3D[fcf[u, v] /. θ → θf, {u, 0, 1}, {v, 0, 1},
         AspectRatio → 0.8, ViewPoint -> {1.3, -2.4, 0.6}]
   }] // fShow;
```



```
Plot3D[fcg[u, v] /. θ → θg, {u, 0.01, 0.99}, {v, 0.01, 0.99},
      AspectRatio → 1, ViewPoint -> {1.3, -2.4, 0.6}] // fShow
```



```
- SurfaceGraphics -
```

```
fCg[0.2, 0.2] /. θ → θg
```

```
0.0648072
```