Příklad 1

Budeme simulovat data náhodný výběr o rozsahu 30, 90 a 150 pocházející z dvojrozměrného normálního rozdělení s parametry $(0, 0)^T$ a $\begin{pmatrix} 1 & \sigma_{XY} \\ \sigma_{XY} & 1 \end{pmatrix}$ a s korelačním koeficientem ρ s hodnotami 0.05, -0.5 a 0.9.

nejprve zadefinujme rozsahy výběru, hodnoty parametrů a hodnoty korelačního koeficientu do příslušných proměnných

In[ ]:=
```
Clear["Global`*"]
```

In[ ]:=
```
n[1] = 30;
n[2] = 90;
n[3] = 150;
μ = {0, 0};
ρ[1] = 0.05;
ρ[2] = -0.5;
ρ[3] = 0.9;
σX = 1;
σY = 1;
pocetopakovani = 1000;
```

Generujme 9 náhodných výběrů. První složka argumentu funkce náhodného výběru určuje argument n pro rozsah výběru a druhá složka odpovídá argumentu ρ určující hodnotu korelačního koeficientu. Z nich sestavíme 95% intervaly spolehlivosti pro korelační koeficient pomocí metod popsaných v práci. Tedy pomocí Fisherovy z-transformace (označíme FZ), metodou zobecněných pivotů (označíme ZP), metodou empirické věrohodnosti (EV) a metodou empirické věrohodnosti s funkcí vlivu (EVFV). Pro každou metodu a pro každou volbu parametrů náhodného výběru postup opakujeme 1000-krát. Z těchto 1000 realizací pro každou volbu parametrů a každou metodu spočteme jak často zjištěný interval spolehlivosti obsahoval skutečný korelační koeficient a jaká byla průměrná délka intervalu spolehlivosti.

Zde generujeme 9 různých skupin náhodných výběrů, v každé skupině je 1000 náhodných výběrů se stejnými parametry, příkaz SeedRandom zajistí, že i při opakovaném volání dostaneme stejné hodnoty

```
In[•]:= SeedRandom[705];
vyber[1, 1] = RandomVariate[MultinormalDistribution[
     μ, {{1, ρ[1] * σX * σY}, {ρ[1] * σX * σY, 1}}], pocetopakovani * n[1]];
vyber[1, 2] = RandomVariate[MultinormalDistribution[μ,
     {{1, ρ[2] * σX * σY}, {ρ[2] * σX * σY, 1}}], pocetopakovani * n[1]];
vyber[1, 3] = RandomVariate[MultinormalDistribution[μ,
    {{1, ρ[3] * σX * σY}, {ρ[3] * σX * σY, 1}}], pocetopakovani * n[1]];
vyber[2, 1] = RandomVariate[MultinormalDistribution[μ,
    {{1, ρ[1] * σX * σY}, {ρ[1] * σX * σY, 1}}], pocetopakovani * n[2]];
vyber[2, 2] = RandomVariate[MultinormalDistribution[μ,
     {{1, ρ[2] * σX * σY}, {ρ[2] * σX * σY, 1}}], pocetopakovani * n[2]];
vyber[2, 3] = RandomVariate[MultinormalDistribution[μ,
     {{1, ρ[3] * σX * σY}, {ρ[3] * σX * σY, 1}}], pocetopakovani * n[2]];
vyber[3, 1] = RandomVariate[MultinormalDistribution[μ,
     {{1, ρ[1] * σX * σY}, {ρ[1] * σX * σY, 1}}], pocetopakovani * n[3]];
vyber[3, 2] = RandomVariate[MultinormalDistribution[μ,
     {{1, ρ[2] * σX * σY}, {ρ[2] * σX * σY, 1}}], pocetopakovani * n[3]];
vyber[3, 3] = RandomVariate[MultinormalDistribution[μ,
     {{1, ρ[3] * σX * σY}, {ρ[3] * σX * σY, 1}}], pocetopakovani * n[3]];
```

Fisherova z-transformace

Začněme Fisherovou z transformací. Zavedeme funkci se vstupem, kde první složka určuje opět zvolený rozsah výběru a druhá složka zvolený korelační koeficient, funkce dělí každý výběr na 1000 náhodných výběrů zvolené délky a jejím výstupem je seznam obsahující na první pozici odhad pravděpodobnosti pokrytí skutečného korelačního koeficientu spočtenými intervaly spolehlivosti a na druhé pozici průměrnou délku těchto intervalů.

```
u0.025 = Quantile[NormalDistribution[0, 1], 0.025];
FZfunkce[ir_, ikor_] := Module[{delka = 0,
    pocetpokryti = 0, r, i, B1, B2, levy, pravy},
   For[i = 1, i ≤ pocetopakovani, i++,
    r = Correlation[vyber[ir, ikor][[1 + (i - 1) * n[ir] ;; i * n[ir]]]][[1, 2]];
    B1 = (1 - r) / (1 + r) * Exp[(-2 / Sqrt[n[ir] - 3]) * u0.025];
    B2 = (1 - r) / (1 + r) * Exp[(2 / Sqrt[n[ir] - 3]) * u0.025]; levy = (1 - B1) / (1 + B1);
    pravy = (1 - B2) / (1 + B2);
    pocetpokryti = pocetpokryti + If[levy ≤ ρ[ikor] && pravy ≥ ρ[ikor], 1, 0];
    delka = delka + pravy - levy];
   {uspesnostpokrytiFZ = pocetpokryti / pocetopakovani // N ,
    prumernadelkaFZ = NumberForm[delka / pocetopakovani, {3, 3}]}]
```

Nyní funkci aplikujme na všech 9 kombinací voleb rozsahu a skutečného korelačního koeficientu. Sestavíme tabulku pro úspěšnost pokrytí a tabulku pro průměrnou délku intervalu, kde ve sloupcích jsou jednotlivé rozsahy výběru a řádky jsou rozděleny podle skutečné hodnoty korelačního koeficientu.

In[∘]:=
```
realizaceFZ = Table[Table[FZfunkce[rozsah, kk], {rozsah, 1, 3}], {kk, 1, 3}];
tabulkauspesnostipokrytiFZ = realizaceFZ[[All, All, 1]];
tabulkaprumernychdelekFZ = realizaceFZ[[All, All, 2]];
Grid[MapThread[Prepend, {Prepend[tabulkauspesnostipokrytiFZ, {n[1], n[2], n[3]}],
   {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[1], ρ[2], ρ[3]}}], Frame → All]
Grid[MapThread[Prepend, {Prepend[tabulkaprumernychdelekFZ, {n[1], n[2], n[3]}],
   {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[1], ρ[2], ρ[3]}}], Frame → All]
Clear[
  i];
```

Out[∘]=

ρ↓ | n →	30	90	150
0.05	0.944	0.96	0.943
−0.5	0.96	0.939	0.956
0.9	0.961	0.948	0.953

Out[∘]=

ρ↓ | n →	30	90	150
0.05	0.697	0.409	0.318
−0.5	0.550	0.311	0.241
0.9	0.153	0.081	0.062

Metoda zobecněných pivotů

Nejprve si vygenerujeme počet výběrů ve skupině krát 10 000 hodnot náhodných veličin U, U21 a Z. Poté Pro každý náhodný výběr spočteme 10 000 hodnot zobecněného pivotu R_ρ pomocí postupu popsaného v kapitole 2 práce, z empirického rozdělení těchto hodnot R_ρ zjistíme krajní body intervalu spolehlivosti každého z tisíce náhodných výběrů jako 0.025-tý a 0.975-tý kvantil.

In[•]:=
```
ZPfunkce[ir_, ikor_] :=
  Module[{delka = 0, pocetpokryti = 0, K = 10 000, aktualnivyber, cov, U, U21,
    Z, RσX, RσYpodmX, sse, b, Rβ, RσY, RσXY, Rρ, listR, levy, pravy, index},
   U = RandomVariate[ChiSquareDistribution[n[ir] - 1], K * pocetopakovani];
   U21 = RandomVariate[ChiSquareDistribution[n[ir] - 2], K * pocetopakovani];
   Z = RandomVariate[NormalDistribution[0, 1], K * pocetopakovani];
   Clear[i];
   For[i = 1, i ≤ pocetopakovani, i++,
    aktualnivyber = vyber[ir, ikor][[1 + (i - 1) * n[ir] ;; i * n[ir]]];
    cov = Covariance[aktualnivyber];
    sse = (n[ir] - 1) (cov[[2, 2]] - (cov[[1, 2]]^2) / cov[[1, 1]]);
    b = cov[[1, 2]] / cov[[1, 1]]; listR = {};
    Clear[j];
    For[j = 1, j ≤ K, j++, index = K * (i - 1) + j;
     RσX = (n[ir] - 1) * cov[[1, 1]] / U[[index]];
     RσYpodmX = sse / U21[[index]];
     Rβ = b - Z[[index]] * Sqrt[sse / (U21[[index]] * (n[ir] - 1) cov[[1, 1]])];
     RσY = (Rβ)^2 * RσX + RσYpodmX;
     RσXY = Rβ * RσX;
     Rρ = RσXY / Sqrt[RσX * RσY];
     listR = Join[listR, {Rρ}]
    ];
    levy = Quantile[listR, 0.025];
    pravy = Quantile[listR, 0.975];
    pocetpokryti = pocetpokryti + If[levy ≤ ρ[ikor] && pravy ≥ ρ[ikor], 1, 0];
    delka = delka + pravy - levy];
   {uspesnostpokrytiZP = pocetpokryti / pocetopakovani // N ,
    prumernadelkaZP = NumberForm[delka / pocetopakovani, {3, 3}]}]
```

In[•]:=
```
SeedRandom[905];
(realizaceZP = Table[Table[ZPfunkce[rozsah, kk], {rozsah, 1, 3}], {kk, 1, 3}]) //
 Timing
tabulkauspesnostipokrytiZP = realizaceZP[[All, All, 1]];
tabulkaprumernychdelekZP = realizaceZP[[All, All, 2]];
Grid[MapThread[Prepend, {Prepend[tabulkauspesnostipokrytiZP, {n[1], n[2], n[3]}],
   {DownArrow["metoda", ""] | RightArrow["n", ""], ρ[1], ρ[2], ρ[3]}}], Frame → All]
Grid[MapThread[Prepend, {Prepend[tabulkaprumernychdelekZP, {n[1], n[2], n[3]}],
   {DownArrow["metoda", ""] | RightArrow["n", ""], ρ[1], ρ[2], ρ[3]}}], Frame → All]
Clear[
  i];
```

Out[•]=
```
{3855.52, {{{0.945, 0.688}, {0.961, 0.407}, {0.939, 0.317}},
  {{0.959, 0.547}, {0.941, 0.311}, {0.955, 0.241}},
  {{0.961, 0.156}, {0.949, 0.082}, {0.954, 0.062}}}}
```

Out[•]=

metoda↓ | n →	30	90	150
0.05	0.945	0.961	0.939
-0.5	0.959	0.941	0.955
0.9	0.961	0.949	0.954

Out[•]=

metoda↓ | n →	30	90	150
0.05	0.688	0.407	0.317
-0.5	0.547	0.311	0.241
0.9	0.156	0.082	0.062

Metoda empirické věrohodnosti

Pro aktuálně zpracovávaný náhodný výběr spočteme výběrovou střední hodnotu, výběrovou kovarianční matici a výběrový korelační koeficient. Spočteme odhad pomocné konstanty A podle zavedení v práci. Proměnné r1 a r2 představují oba kořeny (krajní body intervalu spol.), nastavíme je nejprve na hodnotu mimo interval (-1,1) aby nemohlo dojít k potížím během výpočtu. Pomocné proměnné d, c, kk, dd jsou zde pouze pro účely změn v počátečních aproximacích. While cykly zajistí, že hledání kořenu bude probíhat dokud skutečný reálný kořen nebude nalezen a nebo dokud neuběhne stanovený počet opakování pro různé počáteční aproximace. V takovém případě program vypíše hlášku “chyba xxx”, kde xxx je číslo náhodného výběru (1 až 1000). Funkce FindRoot hledá kořen Newtonovou iterační metodou. Po nalezení kořene probíhá ověření, že nalezený “kořen” je reálný a že skutečně je kořenem. To činíme ze dvou důvodů. Zaprvé funkce FindRoot často najde komplexní kořen, takový musíme zavrhnout. Zadruhé, pokud je počáteční aproximace lambda příliš blízko 0 a program ji poté považuje za nulu, vzniká při Newtonově iterační metodě singulární Jakobián a funkce Findroot poskytne výstup, který je roven počáteční aproximaci a není kořenem. Při obou těchto problémech se někdy vyskytnou chybové hlášky (např. hláška, že FindRoot nenašel kořen za 100 iterací, nedokonvergoval na předepsanou přesnost ke konkrétnímu číslu za 100 iterací, nebo že se objevil singulární Jakobián). Funkce je napsána tak, že tyto chybové hlášky výstup funkce neovlivní. Jedinou chybovou hláškou, na kterou je třeba si dát pozor a v případě které je třeba předpis funkce (změny počátečních aproximací) upravit je zmiňovaná hláška “chyba xxx”, kde xxx je číslo náhodného výběru (1 až 1000).

```
In[•]:= chikvantil = Quantile[ChiSquareDistribution[1], 0.95];
EVfunkce[ir_, ikor_] :=
 Module[{aktualnivyberEV, z, σ0, σv, A, A1, A2, r1 = 0, r2 = 0, c, d, cislo,
   lambda, rho, delka = 0, levy = 0, pravy = 0, uspesnych = 0, strednihodnota,
   kovariancnimatice, cc, sumaA1, sumaA2, overeni, koreny, kk, dd},
  For[z = 1, z ≤ pocetopakovani, z++,
   aktualnivyberEV = vyber[ir, ikor][[1 + (z - 1) * n[ir] ;; z * n[ir]]];
   strednihodnota = Mean[aktualnivyberEV];
   kovariancnimatice = Covariance[aktualnivyberEV];
   cc = Correlation[aktualnivyberEV][[1, 2]];
   (*výběrový korelační koeficient*)
   A2[i_] := (aktualnivyberEV[[i, 1]] - strednihodnota[[1]]) *
     (aktualnivyberEV[[i, 2]] - strednihodnota[[2]]) /
      (Sqrt[kovariancnimatice[[1, 1]] * kovariancnimatice[[2, 2]]]);
   A1[i_] := A2[i] - (cc / 2) * (((aktualnivyberEV[[i, 1]] - strednihodnota[[1]]) ^2) /
         kovariancnimatice[[1, 1]] + ((aktualnivyberEV[[i, 2]] - strednihodnota[[2]]) ^
           2) / kovariancnimatice[[2, 2]]);
   sumaA2 = Sum[A2[k], {k, 1, n[ir]}] / n[ir];
   sumaA1 = Sum[A1[k], {k, 1, n[ir]}] / n[ir];
   σ0 = Sum[(A2[j] - sumaA2) ^2, {j, 1, n[ir]}];
   σv = Sum[(A1[j] - sumaA1) ^2, {j, 1, n[ir]}];
   A = σ0 / σv; (*odhad pomocné konstanty A*)
   r1 = 2;
   (*proměnné r1 a r2 představují oba kořeny (krajní body intervalu spol.),
   nastavíme na hodnotu mimo interval (-1,1) aby nemohlo dojít k potížím*)
   d = 0;
```

```
c = 0;
kk = 0;
dd = 0;
(*pomocné proměnné d,c,kk,
dd jsou zde pouze pro účely změn v počátečních aproximacích*)
While[r1 == 2, (*postup opakuj dokud skutečný kořen nenajdeme*)
 koreny = FindRoot[
   {Sum[Log[1 + lambda * (A2[i] - rho)], {i, 1, n[ir]}] - chikvantil / (2 * A) == 0,
    Sum[(A2[i] - rho) / (1 + lambda * (A2[i] - rho)), {i, 1, n[ir]}] == 0},
   {{rho, (-1)^kk * dd / 5 + cc}, {lambda, Round[Sum[(A2[i] - cc), {i, 1, n[ir]}] /
        Sum[(A2[i] - cc)^2, {i, 1, n[ir]}], 1 / 100000] + (-1)^c * d / 10
     }}, AccuracyGoal →
    5];
 (*hledá kořen soustavy rovnic Newtonovou iterační metodou
  při daných počátečních aproximacích*)
 overeni = Abs[N[Sum[Log[1 + koreny[[2, 2]] * (A2[i] - (koreny[[1, 2]]))],
      {i, 1, n[ir]}] - chikvantil / (2 * A)]];
 (*je nalezená dvojice čísel skutečně kořenem? Někdy (např. při počáteční
     aproximaci lambda=0) vzniká singulární jakobián a funkce FindRoot tak
  dá jako výstup právě počíteční aproximaci, která však není kořenem*)
 r1 = If[Head[N[cislo = koreny[[1, 2]]]] === Real &&
    Head[N[koreny[[2, 2]]]] === Real && overeni < 1 / 1000, cislo, 2];
 (*je nalezená dvojice čísel reálná a je skutečným kořenem? Pokud ano,
 našli jsme hodnotu r1 a while cyklus může skončit,
 pokud ne změníme proměnné, které mění počáteční aproximace*)c++;
 If[OddQ[c], d++, d = d];
 (*Systematicky měníme proměnné d,c,kk,dd*)If[d == 20, kk++, kk = kk];
 If[d == 20 && OddQ[kk], dd++, dd = dd];
 If[d == 20, d = 0, d = d];
 If[dd == 10, Print["chyba ", z];
  (*Ani pro stanovený počet změn počátečních aproximací
   nemáme řešení? Pak vypiš chybu s číslem výběru a zastav cyklus,
  bude třeba nastavení změn počátečních aproximací upravit*)Break[], dd = dd]];
r2 = 2;
d = 0;
c = 0;
kk = 0;
dd = 0;
While[r2 == 2, (*pro r2 identický postup jako pro r1,
 jen přibývá podmínka, že hledáme r2 které se nerovná r1
  (hledáme dva reálné kořeny, ne dvakrát ten stejný)*)
 koreny = FindRoot[{Sum[Log[1 + lambda * (A2[i] - rho)], {i, 1, n[ir]}] -
       chikvantil / (2 * A) == 0,
    Sum[(A2[i] - rho) / (1 + lambda * (A2[i] - rho)), {i, 1, n[ir]}] == 0},
   {{rho, (-1)^kk * dd / 5 + cc}, {lambda, Round[Sum[(A2[i] - cc), {i, 1, n[ir]}] /
        Sum[(A2[i] - cc)^2, {i, 1, n[ir]}], 1 / 100000] + (-1)^c * d / 10
     }}, AccuracyGoal → 5];
 overeni = Abs[N[Sum[Log[1 + koreny[[2, 2]] * (A2[i] - (koreny[[1, 2]]))],
      {i, 1, n[ir]}] - chikvantil / (2 * A)]];
 r2 = If[Head[N[cislo = koreny[[1, 2]]]] === Real && Head[N[koreny[[2, 2]]]] ===
     Real && overeni < 1 / 1000, If[Abs[N[cislo] - N[r1]] < 1 / 1000, 2, cislo], 2];
 c++;
 If[OddQ[c], d++, d = d];
```

```
      If[d == 20, kk++, kk = kk];
      If[d == 20 && OddQ[kk], dd++, dd = dd];
      If[d == 20, d = 0, d = d];
      If[dd == 10, Print["chyba ", z];
       Break[], dd = dd]];
     pravy = Max[r1, r2];
     levy = Min[r1, r2];
     delka = delka + pravy - levy;
     uspesnych = uspesnych + If[levy ≤ ρ[ikor] && pravy ≥ ρ[ikor], 1, 0];
    ];
    {uspesnostpokrytiEV = uspesnych / pocetopakovani // N,
     prumernadelkaEV = NumberForm[delka / pocetopakovani, {3, 3}]}]
```

In[∘]:=
```
(realizaceEV = Table[Table[EVfunkce[rozsah, kk], {rozsah, 1, 3}], {kk, 1, 3}]) // Timing
tabulkauspesnostipokrytiEV = realizaceEV[[All, All, 1]];
tabulkaprumernychdelekEV = realizaceEV[[All, All, 2]];
Grid[MapThread[Prepend, {Prepend[tabulkauspesnostipokrytiEV, {n[1], n[2], n[3]}],
    {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[1], ρ[2], ρ[3]}}], Frame → All]
Grid[MapThread[Prepend, {Prepend[tabulkaprumernychdelekEV, {n[1], n[2], n[3]}],
    {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[1], ρ[2], ρ[3]}}], Frame → All]
```

... FindRoot: The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

Out[∘]=
```
{695.313, {{{0.875, 0.669}, {0.936, 0.414}, {0.923, 0.320}},
  {{0.912, 0.500}, {0.919, 0.303}, {0.949, 0.239}},
  {{0.947, 0.125}, {0.946, 0.076}, {0.959, 0.059}}}}
```

Out[∘]=

ρ↓ | n →	30	90	150
0.05	0.875	0.936	0.923
-0.5	0.912	0.919	0.949
0.9	0.947	0.946	0.959

Out[∘]=

ρ↓ | n →	30	90	150
0.05	0.669	0.414	0.320
-0.5	0.500	0.303	0.239
0.9	0.125	0.076	0.059

Metoda empirické věrohodnosti s funkcí vlivu

Postup je ve funkci stejný jako pro metodu empirické věrohodnosti, jen neodhadujeme pomocnou konstantu a upravíme rovnosti stejně jako v práci. Informace o chybových hláškách platí stejné jako v případě empirické věrohodnosti.

In[∘]:=
```
chikvantil = Quantile[ChiSquareDistribution[1], 0.95];
EVFVfunkce[ir_, ikor_] :=
 Module[{aktualnivyberEV, z, r1 = 0, r2 = 0, Xzlomek, Yzlomek, c,
   d, koreny, lambda, rho, delka = 0, levy = 0, pravy = 0, uspesnych = 0,
   strednihodnota, kovariancnimatice, cc, overeni, cislo, kk, dd},
  For[z = 1, z ≤ pocetopakovani, z++,
   aktualnivyberEV = vyber[ir, ikor][[1 + (z - 1) * n[ir] ;; z * n[ir]]];
   strednihodnota = Mean[aktualnivyberEV];
   kovariancnimatice = Covariance[aktualnivyberEV];
   cc = Correlation[aktualnivyberEV][[1, 2]];
   Xzlomek[i_] := (aktualnivyberEV[[i, 1]] - strednihodnota[[1]]) /
```

```
    (Sqrt[kovariancnimatice[[1, 1]]]);
Yzlomek[i_] := (aktualnivyberEV[[i, 2]] - strednihodnota[[2]]) /
    (Sqrt[kovariancnimatice[[2, 2]]]);
r1 = 2;
d = 0;
c = 0;
kk = 0;
dd = 0;
While[r1 == 2,
  koreny = FindRoot[
     {Sum[Log[1 + lambda * (Xzlomek[i] * Yzlomek[i] - rho - (rho / 2) * (Xzlomek[i] ^2 -
                 1 + Yzlomek[i] ^2 - 1))], {i, 1, n[ir]}] - chikvantil / 2 == 0,
       Sum[ (Xzlomek[i] * Yzlomek[i] - rho - (rho / 2) * (Xzlomek[i] ^2 - 1 +
                 Yzlomek[i] ^2 - 1)) / (1 + lambda * (Xzlomek[i] * Yzlomek[i] -
                 rho - (rho / 2) * (Xzlomek[i] ^2 - 1 + Yzlomek[i] ^2 - 1))),
          {i, 1, n[ir]}] == 0}, {{rho, (-1) ^kk * dd / 5 + cc}, {lambda,
        Round[Sum[ (Xzlomek[i] * Yzlomek[i] - cc - (cc / 2) *
                 (Xzlomek[i] ^2 - 1 + Yzlomek[i] ^2 - 1)), {i, 1, n[ir]}] /
            Sum[ (Xzlomek[i] * Yzlomek[i] - cc - (cc / 2) * (Xzlomek[i] ^2 - 1 +
                    Yzlomek[i] ^2 - 1)) ^2, {i, 1, n[ir]}], 1 / 1000] + (-1) ^c * d / 10
      }}, AccuracyGoal -> 5];
  overeni =
   Abs[N[Sum[Log[1 + koreny[[2, 2]] * (Xzlomek[i] * Yzlomek[i] - (koreny[[1, 2]]) -
              (koreny[[1, 2]] / 2) * (Xzlomek[i] ^2 - 1 + Yzlomek[i] ^2 -
                 1))], {i, 1, n[ir]}] - chikvantil / 2]];
  r1 = If[Head[N[cislo = koreny[[1, 2]]]] === Real && Head[N[koreny[[2, 2]]]] ===
       Real  && overeni < 1 / 1000, cislo, 2];
  c ++;
  If[OddQ[c], d ++, d = d];
  If[d == 20, kk ++, kk = kk];
  If[d == 20 && OddQ[kk], dd ++, dd = dd];
  If[d == 20, d = 0, d = d];
  If[dd == 10, Print["chyba ", z];
   Break[], dd = dd]];
r2 = 2;
d = 0;
c = 0;
kk = 0;
dd = 0;
While[r2 == 2,
  koreny = FindRoot[
     {Sum[Log[1 + lambda * (Xzlomek[i] * Yzlomek[i] - rho - (rho / 2) * (Xzlomek[i] ^2 -
                 1 + Yzlomek[i] ^2 - 1))], {i, 1, n[ir]}] - chikvantil / 2 == 0,
       Sum[ (Xzlomek[i] * Yzlomek[i] - rho - (rho / 2) * (Xzlomek[i] ^2 - 1 +
                 Yzlomek[i] ^2 - 1)) / (1 + lambda * (Xzlomek[i] * Yzlomek[i] - rho -
                 (rho / 2) * (Xzlomek[i] ^2 - 1 + Yzlomek[i] ^2 - 1))), {i, 1, n[ir]}] == 0},
     {{rho, (-1) ^kk * dd / 5 + cc}, {lambda, Round[Sum[ (Xzlomek[i] * Yzlomek[i] -
               cc - (cc / 2) * (Xzlomek[i] ^2 - 1 + Yzlomek[i] ^2 - 1)), {i, 1, n[ir]}] /
            Sum[ (Xzlomek[i] * Yzlomek[i] - cc - (cc / 2) * (Xzlomek[i] ^2 - 1 +
                    Yzlomek[i] ^2 - 1)) ^2, {i, 1, n[ir]}], 1 / 1000] + (-1) ^c * d / 10
      }}, AccuracyGoal -> 5];
  overeni =
```

```
        Abs[N[Sum[Log[1 + koreny[[2, 2]] * (Xzlomek[i] * Yzlomek[i] - (koreny[[1, 2]]) -
                  (koreny[[1, 2]] / 2) * (Xzlomek[i]^2 - 1 + Yzlomek[i]^2 -
                     1))], {i, 1, n[ir]}] - chikvantil / 2]];
      r2 = If[Head[N[cislo = koreny[[1, 2]]]] === Real && Head[N[koreny[[2, 2]]]] ===
          Real && overeni < 1 / 1000, If[Abs[N[cislo] - N[r1]] < 1 / 1000, 2, cislo], 2];
      c++;
      If[OddQ[c], d++, d = d];
      If[d == 20, kk++, kk = kk];
      If[d == 20 && OddQ[kk], dd++, dd = dd];
      If[d == 20, d = 0, d = d];
      If[dd == 10, Print["chyba ", z];
        Break[], dd = dd]];
     pravy = Max[r1, r2];
     levy = Min[r1, r2];
     delka = delka + pravy - levy;
     uspesnych = uspesnych + If[levy ≤ ρ[ikor] && pravy ≥ ρ[ikor], 1, 0];
    ];
   {uspesnostpokrytiEVFV = uspesnych / pocetopakovani // N ,
     prumernadelkaEVFV = NumberForm[delka / pocetopakovani, {3, 3}]}]
```

In[•]:=

```
(realizaceEVFV = Table[Table[EVFVfunkce[rozsah, kk], {rozsah, 1, 3}], {kk, 1, 3}]) //
  Timing
tabulkauspesnostipokrytiEVFV = realizaceEVFV[[All, All, 1]];
tabulkaprumernychdelekEVFV = realizaceEVFV[[All, All, 2]];
Grid[MapThread[Prepend, {Prepend[tabulkauspesnostipokrytiEVFV, {n[1], n[2], n[3]}],
    {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[1], ρ[2], ρ[3]}}], Frame → All]
Grid[MapThread[Prepend, {Prepend[tabulkaprumernychdelekEVFV, {n[1], n[2], n[3]}],
    {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[1], ρ[2], ρ[3]}}], Frame → All]
```

... FindRoot: Encountered a singular Jacobian at the point {rho$308397, lambda$308397} = {0.0270912, 0.}. Try perturbing the initial point(s).

... FindRoot: Encountered a singular Jacobian at the point {rho$308397, lambda$308397} = {0.0270912, 0.}. Try perturbing the initial point(s).

... FindRoot: Encountered a singular Jacobian at the point {rho$308397, lambda$308397} = {-0.141231, 0.}. Try perturbing the initial point(s).

... General: Further output of FindRoot::jsing will be suppressed during this calculation.

... FindRoot: Failed to converge to the requested accuracy or precision within 100 iterations.

... FindRoot: The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

... FindRoot: The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

... FindRoot: The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

... General: Further output of FindRoot::lstol will be suppressed during this calculation.

... FindRoot: Failed to converge to the requested accuracy or precision within 100 iterations.

... FindRoot: Failed to converge to the requested accuracy or precision within 100 iterations.

General: Further output of FindRoot::cvmit will be suppressed during this calculation.

```
Out[]= {2076.22, {{{0.912, 0.629}, {0.949, 0.398}, {0.933, 0.312}},
  {{0.935, 0.496}, {0.922, 0.303}, {0.956, 0.239}},
  {{0.92, 0.137}, {0.942, 0.079}, {0.951, 0.061}}}}
```

Out[]=

ρ↓ | n →	30	90	150
0.05	0.912	0.949	0.933
-0.5	0.935	0.922	0.956
0.9	0.92	0.942	0.951

Out[]=

ρ↓ | n →	30	90	150
0.05	0.629	0.398	0.312
-0.5	0.496	0.303	0.239
0.9	0.137	0.079	0.061

Tabulky porovnání metod

Vytvořme tabulky pro úspěšnosti pokrytí a průměrnou délku intervalů spolehlivosti pomocí všech metod pro různé volby rozsahu výběru a korelačního koeficientu. Každou z tabulek si rovněž převedeme do zápisu pro LATEX.

```
In[]:= uradek1 = tabulkauspesnostipokrytiFZ // Flatten;
uradek2 = tabulkauspesnostipokrytiZP // Flatten;
uradek3 = tabulkauspesnostipokrytiEV // Flatten;
uradek4 = tabulkauspesnostipokrytiEVFV // Flatten;
dradek1 = tabulkaprumernychdelekFZ // Flatten;
dradek2 = tabulkaprumernychdelekZP // Flatten;
dradek3 = tabulkaprumernychdelekEV // Flatten;
dradek4 = tabulkaprumernychdelekEVFV // Flatten;
tabulka1usp =
  Prepend[MapThread[Prepend, {Prepend[{uradek1, uradek2, uradek3, uradek4},
      Table[n[If[Mod[g, 3] == 0, 3, Mod[g, 3]]], {g, 1, 9}]],
     {DownArrow["metoda", ""] | RightArrow["n", ""], "FZ", "ZP", "EV", "EVFV"}}],
   {RightArrow["ρ", ""], " ", ρ[1], " ", " ", ρ[2], " ", " ", ρ[3], " "}];
Row[{"Tabulka úspěšností pokrytí"}]
Grid[tabulka1usp, Frame → False]
tabulka1d = Prepend[MapThread[Prepend, {Prepend[{dradek1, dradek2, dradek3, dradek4},
      Table[n[If[Mod[g, 3] == 0, 3, Mod[g, 3]]], {g, 1, 9}]],
     {DownArrow["metoda", ""] | RightArrow["n", ""], "FZ", "ZP", "EV", "EVFV"}}],
   {RightArrow["ρ", ""], " ", ρ[1], " ", " ", ρ[2], " ", " ", ρ[3], " "}];
Row[{"Tabulka průměrných délek intervalů"}]
Grid[tabulka1d, Frame → False]
Row[{tabulka1usp // TeXForm, "\n", tabulka1d // TeXForm}]
```

Out[]= Tabulka úspěšností pokrytí

Out[]=

ρ →		0.05			-0.5			0.9	
metoda↓ | n →	30	90	150	30	90	150	30	90	150
FZ	0.944	0.96	0.943	0.96	0.939	0.956	0.961	0.948	0.953
ZP	0.945	0.961	0.939	0.959	0.941	0.955	0.961	0.949	0.954
EV	0.875	0.936	0.923	0.912	0.919	0.949	0.947	0.946	0.959
EVFV	0.912	0.949	0.933	0.935	0.922	0.956	0.92	0.942	0.951

Out[]= Tabulka průměrných délek intervalů

Out[∘]=

ρ →	0.05			−0.5			0.9		
metoda↓ | n →	30	90	150	30	90	150	30	90	150
FZ	0.697	0.409	0.318	0.550	0.311	0.241	0.153	0.081	0.062
ZP	0.688	0.407	0.317	0.547	0.311	0.241	0.156	0.082	0.062
EV	0.669	0.414	0.320	0.500	0.303	0.239	0.125	0.076	0.059
EVFV	0.629	0.398	0.312	0.496	0.303	0.239	0.137	0.079	0.061

Out[∘]=

```
\left(
\begin{array}{cccccccccc}
 \rho \rightarrow \text{} &  & 0.05 &  &  & -0.5 &  &  & 0.9 &  \\
 \text{metoda}\downarrow \text{}|\text{n}\rightarrow
   \text{} & 30 & 90 & 150 & 30 & 90 & 150 & 30 & 90 & 150 \\
 \text{FZ} & 0.944 & 0.96 & 0.943 & 0.96 & 0.939 & 0.956 & 0.961 & 0.948 & 0.953 \\
 \text{ZP} & 0.945 & 0.961 & 0.939 & 0.959 & 0.941 & 0.955 & 0.961 & 0.949 & 0.954 \\
 \text{EV} & 0.875 & 0.936 & 0.923 & 0.912 & 0.919 & 0.949 & 0.947 & 0.946 & 0.959 \\
 \text{EVFV} & 0.912 & 0.949 &
   0.933 & 0.935 & 0.922 & 0.956 & 0.92 & 0.942 & 0.951 \\
\end{array}
\right)
\left(
\begin{array}{cccccccccc}
 \rho \rightarrow \text{} &  & 0.05 &  &  & -0.5 &  &  & 0.9 &  \\
 \text{metoda}\downarrow \text{}|\text{n}\rightarrow
   \text{} & 30 & 90 & 150 & 30 & 90 & 150 & 30 & 90 & 150 \\
 \text{FZ} & 0.697 & 0.409 & 0.318 & 0.550 & 0.311 & 0.241 & 0.153 & 0.081 & 0.062 \\
 \text{ZP} & 0.688 & 0.407 & 0.317 & 0.547 & 0.311 & 0.241 & 0.156 & 0.082 & 0.062 \\
 \text{EV} & 0.669 & 0.414 & 0.320 & 0.500 & 0.303 & 0.239 & 0.125 & 0.076 & 0.059 \\
 \text{EVFV} & 0.629 & 0.398 &
   0.312 & 0.496 & 0.303 & 0.239 & 0.137 & 0.079 & 0.061 \\
\end{array}
\right)
```

Příklad 2

Budeme simulovat data náhodný výběr o rozsahu 30, 90 a 150 pocházející z dvojrozměrného normálního rozdělení s parametry $(2, 10)^T$ a $\begin{pmatrix} 4 & \sigma_{XY} \\ \sigma_{XY} & 9 \end{pmatrix}$ a s korelačním koeficientem ρ s hodnotami 0.05, -0.5 a 0.9.

In[∘]:=

```
n[4] = 30;
n[5] = 90;
n[6] = 150;
μ2 = {2, 10};
ρ[4] = 0.05;
ρ[5] = -0.5;
ρ[6] = 0.9;
p2σX = 2;
p2σY = 3;
p2varx = 4;
p2vary = 9;
pocetopakovani = 1000;
```

Opět generujme 9 náhodných výběrů. První složka argumentu funkce náhodného výběru určuje argument n pro rozsah výběru a druhá složka odpovídá argumentu ρ určující hodnotu korelačního koeficientu. Z nich sestavíme 95% intervaly spolehlivosti pro korelační koeficient pomocí metod popsaných v práci stejně jako v příkladu 1.

```
In[•]:= SeedRandom[1005];
vyber[4, 4] = RandomVariate[MultinormalDistribution[μ2,
    {{p2varx, ρ[4] * p2σX * p2σY}, {ρ[4] * p2σX * p2σY, p2vary}}], pocetopakovani * n[4]];
vyber[4, 5] = RandomVariate[MultinormalDistribution[μ2,
    {{p2varx, ρ[5] * p2σX * p2σY}, {ρ[5] * p2σX * p2σY, p2vary}}], pocetopakovani * n[4]];
vyber[4, 6] = RandomVariate[MultinormalDistribution[μ2,
   {{p2varx, ρ[6] * p2σX * p2σY}, {ρ[6] * p2σX * p2σY, p2vary}}], pocetopakovani * n[4]];
vyber[5, 4] = RandomVariate[MultinormalDistribution[μ2,
   {{p2varx, ρ[4] * p2σX * p2σY}, {ρ[4] * p2σX * p2σY, p2vary}}], pocetopakovani * n[5]];
vyber[5, 5] = RandomVariate[MultinormalDistribution[μ2,
    {{p2varx, ρ[5] * p2σX * p2σY}, {ρ[5] * p2σX * p2σY, p2vary}}], pocetopakovani * n[5]];
vyber[5, 6] = RandomVariate[MultinormalDistribution[μ2,
    {{p2varx, ρ[6] * p2σX * p2σY}, {ρ[6] * p2σX * p2σY, p2vary}}], pocetopakovani * n[5]];
vyber[6, 4] = RandomVariate[MultinormalDistribution[μ2,
    {{p2varx, ρ[4] * p2σX * p2σY}, {ρ[4] * p2σX * p2σY, p2vary}}], pocetopakovani * n[6]];
vyber[6, 5] = RandomVariate[MultinormalDistribution[μ2,
    {{p2varx, ρ[5] * p2σX * p2σY}, {ρ[5] * p2σX * p2σY, p2vary}}], pocetopakovani * n[6]];
vyber[6, 6] = RandomVariate[MultinormalDistribution[μ2,
    {{p2varx, ρ[6] * p2σX * p2σY}, {ρ[6] * p2σX * p2σY, p2vary}}], pocetopakovani * n[6]];
```

Fisherova z-transformace

```
In[•]:= u0.025 = Quantile[NormalDistribution[0, 1], 0.025];
realizaceFZp2 = Table[Table[FZfunkce[rozsah, kk], {rozsah, 4, 6}], {kk, 4, 6}];
tabulkauspesnostipokrytiFZp2 = realizaceFZp2[[All, All, 1]];
tabulkaprumernychdelekFZp2 = realizaceFZp2[[All, All, 2]];
Grid[MapThread[Prepend, {Prepend[tabulkauspesnostipokrytiFZp2, {n[4], n[5], n[6]}],
   {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[4], ρ[5], ρ[6]}}], Frame → All]
Grid[MapThread[Prepend, {Prepend[tabulkaprumernychdelekFZp2, {n[4], n[5], n[6]}],
   {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[4], ρ[5], ρ[6]}}], Frame → All]
Clear[
  i];
```

Out[•]=

ρ↓ | n →	30	90	150
0.05	0.944	0.949	0.94
-0.5	0.945	0.955	0.954
0.9	0.948	0.954	0.949

Out[•]=

ρ↓ | n →	30	90	150
0.05	0.697	0.409	0.318
-0.5	0.551	0.313	0.241
0.9	0.157	0.083	0.062

Metoda zobecněných pivotů

In[ ]:=
```
SeedRandom[1105];
(realizaceZPp2 = Table[Table[ZPfunkce[rozsah, kk], {rozsah, 4, 6}], {kk, 4, 6}]) //
 Timing
tabulkauspesnostipokrytiZPp2 = realizaceZPp2[[All, All, 1]];
tabulkaprumernychdelekZPp2 = realizaceZPp2[[All, All, 2]];
Grid[MapThread[Prepend, {Prepend[tabulkauspesnostipokrytiZPp2, {n[4], n[5], n[6]}],
   {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[4], ρ[5], ρ[6]}}], Frame → All]
Grid[MapThread[Prepend, {Prepend[tabulkaprumernychdelekZPp2, {n[4], n[5], n[6]}],
   {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[4], ρ[5], ρ[6]}}], Frame → All]
Clear[
  i];
```

Out[ ]=
```
{4061.52, {{{0.945, 0.687}, {0.95, 0.407}, {0.94, 0.317}},
  {{0.944, 0.547}, {0.954, 0.313}, {0.954, 0.241}},
  {{0.947, 0.160}, {0.957, 0.083}, {0.948, 0.062}}}}
```

Out[ ]=

ρ↓ | n →	30	90	150
0.05	0.945	0.95	0.94
-0.5	0.944	0.954	0.954
0.9	0.947	0.957	0.948

Out[ ]=

ρ↓ | n →	30	90	150
0.05	0.687	0.407	0.317
-0.5	0.547	0.313	0.241
0.9	0.160	0.083	0.062

Metoda empirické věrohodnosti

In[ ]:=
```
chikvantil = Quantile[ChiSquareDistribution[1], 0.95];
(realizaceEVp2 = Table[Table[EVfunkce[rozsah, kk], {rozsah, 4, 6}], {kk, 4, 6}]) //
 Timing
tabulkauspesnostipokrytiEVp2 = realizaceEVp2[[All, All, 1]];
tabulkaprumernychdelekEVp2 = realizaceEVp2[[All, All, 2]];
Grid[MapThread[Prepend, {Prepend[tabulkauspesnostipokrytiEVp2, {n[4], n[5], n[6]}],
   {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[4], ρ[5], ρ[6]}}], Frame → All]
Grid[MapThread[Prepend, {Prepend[tabulkaprumernychdelekEVp2, {n[4], n[5], n[6]}],
   {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[4], ρ[5], ρ[6]}}], Frame → All]
```

... FindRoot: The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

Out[ ]=
```
{697.484, {{{0.862, 0.677}, {0.916, 0.415}, {0.937, 0.323}},
  {{0.896, 0.503}, {0.946, 0.305}, {0.952, 0.238}},
  {{0.934, 0.129}, {0.962, 0.077}, {0.948, 0.060}}}}
```

Out[ ]=

ρ↓ | n →	30	90	150
0.05	0.862	0.916	0.937
-0.5	0.896	0.946	0.952
0.9	0.934	0.962	0.948

Out[ ]=

ρ↓ | n →	30	90	150
0.05	0.677	0.415	0.323
-0.5	0.503	0.305	0.238
0.9	0.129	0.077	0.060

Metoda empirické věrohodnosti s funkcí vlivu

```
In[•]:= chikvantil = Quantile[ChiSquareDistribution[1], 0.95];
(realizaceEVFVp2 = Table[Table[EVFVfunkce[rozsah, kk], {rozsah, 4, 6}], {kk, 4, 6}]) //
 Timing
tabulkauspesnostipokrytiEVFVp2 = realizaceEVFVp2[[All, All, 1]];
tabulkaprumernychdelekEVFVp2 = realizaceEVFVp2[[All, All, 2]];
Grid[MapThread[Prepend, {Prepend[tabulkauspesnostipokrytiEVFVp2, {n[4], n[5], n[6]}],
    {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[4], ρ[5], ρ[6]}}], Frame → All]
Grid[MapThread[Prepend, {Prepend[tabulkaprumernychdelekEVFVp2, {n[4], n[5], n[6]}],
    {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[4], ρ[5], ρ[6]}}], Frame → All]
```

... FindRoot: Encountered a singular Jacobian at the point {rho$420852, lambda$420852} = {−0.230319, 0.}. Try perturbing the initial point(s).

... FindRoot: Encountered a singular Jacobian at the point {rho$420852, lambda$420852} = {−0.230319, 0.}. Try perturbing the initial point(s).

... FindRoot: Encountered a singular Jacobian at the point {rho$420852, lambda$420852} = {0.0384151, 0.}. Try perturbing the initial point(s).

... General: Further output of FindRoot::jsing will be suppressed during this calculation.

... FindRoot: The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

... FindRoot: The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

... FindRoot: The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

... General: Further output of FindRoot::lstol will be suppressed during this calculation.

... FindRoot: Failed to converge to the requested accuracy or precision within 100 iterations.

... FindRoot: Failed to converge to the requested accuracy or precision within 100 iterations.

... FindRoot: Failed to converge to the requested accuracy or precision within 100 iterations.

... General: Further output of FindRoot::cvmit will be suppressed during this calculation.

```
Out[•]= {2142.56, {{{0.908, 0.634}, {0.936, 0.399}, {0.943, 0.314}},
  {{0.91, 0.499}, {0.933, 0.304}, {0.951, 0.238}},
  {{0.906, 0.141}, {0.944, 0.081}, {0.947, 0.062}}}}
```

Out[•]=

ρ↓ | n →	30	90	150
0.05	0.908	0.936	0.943
−0.5	0.91	0.933	0.951
0.9	0.906	0.944	0.947

Out[•]=

ρ↓ | n →	30	90	150
0.05	0.634	0.399	0.314
−0.5	0.499	0.304	0.238
0.9	0.141	0.081	0.062

Tabulky porovnání metod

Postup stejný jako v příkladu 1.

```
In[•]:= uradek1 = tabulkauspesnostipokrytiFZp2 // Flatten;
uradek2 = tabulkauspesnostipokrytiZPp2 // Flatten;
uradek3 = tabulkauspesnostipokrytiEVp2 // Flatten;
uradek4 = tabulkauspesnostipokrytiEVFVp2 // Flatten;
dradek1 = tabulkaprumernychdelekFZp2 // Flatten;
dradek2 = tabulkaprumernychdelekZPp2 // Flatten;
dradek3 = tabulkaprumernychdelekEVp2 // Flatten;
dradek4 = tabulkaprumernychdelekEVFVp2 // Flatten;
tabulka2usp =
  Prepend[MapThread[Prepend, {Prepend[{uradek1, uradek2, uradek3, uradek4},
      Table[n[3 + If[Mod[g, 3] == 0, 3, Mod[g, 3]]], {g, 1, 9}]],
     {DownArrow["metoda", ""] | RightArrow["n", ""], "FZ", "ZP", "EV", "EVFV"}}],
   {RightArrow["ρ", ""], " ", ρ[4], " ", " ", ρ[4], " ", " ", ρ[6], " "}];
Row[{"Tabulka úspěšností pokrytí"}]
Grid[tabulka2usp, Frame → False]
tabulka2d = Prepend[MapThread[Prepend, {Prepend[{dradek1, dradek2, dradek3, dradek4},
      Table[n[3 + If[Mod[g, 3] == 0, 3, Mod[g, 3]]], {g, 1, 9}]],
     {DownArrow["metoda", ""] | RightArrow["n", ""], "FZ", "ZP", "EV", "EVFV"}}],
   {RightArrow["ρ", ""], " ", ρ[4], " ", " ", ρ[5], " ", " ", ρ[6], " "}];
Row[{"Tabulka průměrných délek intervalů"}]
Grid[tabulka2d, Frame → False]
Row[{tabulka2usp // TeXForm, "\n", tabulka2d // TeXForm}]
```

Out[•]= Tabulka úspěšností pokrytí

ρ →		0.05			0.05			0.9	
metoda ↓ | n →	30	90	150	30	90	150	30	90	150
FZ	0.944	0.949	0.94	0.945	0.955	0.954	0.948	0.954	0.949
ZP	0.945	0.95	0.94	0.944	0.954	0.954	0.947	0.957	0.948
EV	0.862	0.916	0.937	0.896	0.946	0.952	0.934	0.962	0.948
EVFV	0.908	0.936	0.943	0.91	0.933	0.951	0.906	0.944	0.947

Out[•]= Tabulka průměrných délek intervalů

ρ →		0.05			-0.5			0.9	
metoda ↓ | n →	30	90	150	30	90	150	30	90	150
FZ	0.697	0.409	0.318	0.551	0.313	0.241	0.157	0.083	0.062
ZP	0.687	0.407	0.317	0.547	0.313	0.241	0.160	0.083	0.062
EV	0.677	0.415	0.323	0.503	0.305	0.238	0.129	0.077	0.060
EVFV	0.634	0.399	0.314	0.499	0.304	0.238	0.141	0.081	0.062

Out[ ]=
```
\left(
\begin{array}{cccccccccc}
 \rho \rightarrow \text{} &   & 0.05 &   &   & 0.05 &   &   & 0.9 &   \\
 \text{metoda}\downarrow \text{}|\text{n}\rightarrow
   \text{} & 30 & 90 & 150 & 30 & 90 & 150 & 30 & 90 & 150 \\
 \text{FZ} & 0.944 & 0.949 & 0.94 & 0.945 & 0.955 & 0.954 & 0.948 & 0.954 & 0.949 \\
 \text{ZP} & 0.945 & 0.95 & 0.94 & 0.944 & 0.954 & 0.954 & 0.947 & 0.957 & 0.948 \\
 \text{EV} & 0.862 & 0.916 & 0.937 & 0.896 & 0.946 & 0.952 & 0.934 & 0.962 & 0.948 \\
 \text{EVFV} & 0.908 & 0.936 &
   0.943 & 0.91 & 0.933 & 0.951 & 0.906 & 0.944 & 0.947 \\
\end{array}
\right)
\left(
\begin{array}{cccccccccc}
 \rho \rightarrow \text{} &   & 0.05 &   &   & -0.5 &   &   & 0.9 &   \\
 \text{metoda}\downarrow \text{}|\text{n}\rightarrow
   \text{} & 30 & 90 & 150 & 30 & 90 & 150 & 30 & 90 & 150 \\
 \text{FZ} & 0.697 & 0.409 & 0.318 & 0.551 & 0.313 & 0.241 & 0.157 & 0.083 & 0.062 \\
 \text{ZP} & 0.687 & 0.407 & 0.317 & 0.547 & 0.313 & 0.241 & 0.160 & 0.083 & 0.062 \\
 \text{EV} & 0.677 & 0.415 & 0.323 & 0.503 & 0.305 & 0.238 & 0.129 & 0.077 & 0.060 \\
 \text{EVFV} & 0.634 & 0.399 &
   0.314 & 0.499 & 0.304 & 0.238 & 0.141 & 0.081 & 0.062 \\
\end{array}
\right)
```

Příklad 3

Nyní budeme simulovat náhodné výběry dvojrozměrného Poissonova rozdělení s parametry 1, 1, λ_{12}, kde pro parametr λ_{12} budeme volit 1/9,1,9 a pro rozsah výběru hodnoty 30,90,150.

In[ ]:=
```
n[7] = 30;
n[8] = 90;
n[9] = 150;
μ = {0, 0};
λ12[1] = 1/9;
λ12[2] = 1;
λ12[3] = 9;
ρ[7] = λ12[1] / (1 + λ12[1]);
ρ[8] = λ12[2] / (1 + λ12[2]);
ρ[9] = λ12[3] / (1 + λ12[3]);
pocetopakovani = 1000;
```

In[∘]:=
```
SeedRandom[1205];
vyber[7, 7] = N[RandomVariate[
    MultivariatePoissonDistribution[λ12[1], {1, 1}], pocetopakovani * n[7]]];
vyber[7, 8] = N[RandomVariate[MultivariatePoissonDistribution[λ12[2], {1, 1}],
    pocetopakovani * n[7]]];
vyber[7, 9] = N[RandomVariate[MultivariatePoissonDistribution[λ12[3], {1, 1}],
    pocetopakovani * n[7]]];
vyber[8, 7] = N[RandomVariate[MultivariatePoissonDistribution[λ12[1], {1, 1}],
    pocetopakovani * n[8]]];
vyber[8, 8] = N[RandomVariate[MultivariatePoissonDistribution[λ12[2], {1, 1}],
    pocetopakovani * n[8]]];
vyber[8, 9] = N[RandomVariate[MultivariatePoissonDistribution[λ12[3], {1, 1}],
    pocetopakovani * n[8]]];
vyber[9, 7] = N[RandomVariate[MultivariatePoissonDistribution[λ12[1], {1, 1}],
    pocetopakovani * n[9]]];
vyber[9, 8] = N[RandomVariate[MultivariatePoissonDistribution[λ12[2], {1, 1}],
    pocetopakovani * n[9]]];
vyber[9, 9] = N[RandomVariate[MultivariatePoissonDistribution[λ12[3], {1, 1}],
    pocetopakovani * n[9]]];
```

Fisherova z-transformace

In[∘]:=
```
u0.025 = Quantile[NormalDistribution[0, 1], 0.025];
realizaceFZp3 = Table[Table[FZfunkce[rozsah, kk], {rozsah, 7, 9}], {kk, 7, 9}];
tabulkauspesnostipokrytiFZp3 = realizaceFZp3[[All, All, 1]];
tabulkaprumernychdelekFZp3 = realizaceFZp3[[All, All, 2]];
Grid[MapThread[Prepend,
  {Prepend[tabulkauspesnostipokrytiFZp3, {n[7], n[8], n[9]}], {DownArrow["ρ", ""] |
     RightArrow["n", ""], N[ρ[7]], N[ρ[8]], N[ρ[9]]}}], Frame → All]
Grid[MapThread[Prepend, {Prepend[tabulkaprumernychdelekFZp3, {n[7], n[8], n[9]}],
   {DownArrow["ρ", ""] | RightArrow["n", ""],
    N[ρ[7]], N[ρ[8]], N[ρ[9]]}}], Frame → All]
Clear[
  i];
```

Out[∘]=

ρ↓ | n→	30	90	150
0.1	0.946	0.934	0.941
0.5	0.934	0.93	0.932
0.9	0.93	0.937	0.944

Out[∘]=

ρ↓ | n→	30	90	150
0.1	0.692	0.406	0.315
0.5	0.547	0.312	0.241
0.9	0.158	0.083	0.062

Metoda zobecněných pivotů

In[ ]:=
```
SeedRandom[1405];
(realizaceZPp3 = Table[Table[ZPfunkce[rozsah, kk], {rozsah, 7, 9}], {kk, 7, 9}]) //
 Timing
tabulkauspesnostipokrytiZPp3 = realizaceZPp3[[All, All, 1]];
tabulkaprumernychdelekZPp3 = realizaceZPp3[[All, All, 2]];
Grid[MapThread[Prepend,
  {Prepend[tabulkauspesnostipokrytiZPp3, {n[7], n[8], n[9]}], {DownArrow["ρ", ""] |
     RightArrow["n", ""], N[ρ[7]], N[ρ[8]], N[ρ[9]]}}], Frame → All]
Grid[MapThread[Prepend, {Prepend[tabulkaprumernychdelekZPp3, {n[7], n[8], n[9]}],
   {DownArrow["ρ", ""] | RightArrow["n", ""],
    N[ρ[7]], N[ρ[8]], N[ρ[9]]}}], Frame → All]
Clear[
  i];
```

Out[ ]=
```
{3990.17, {{{0.947, 0.683}, {0.934, 0.404}, {0.942, 0.315}},
  {{0.933, 0.544}, {0.929, 0.312}, {0.933, 0.241}},
  {{0.929, 0.161}, {0.936, 0.083}, {0.944, 0.062}}}}
```

Out[ ]=

ρ↓ | n →	30	90	150
0.1	0.947	0.934	0.942
0.5	0.933	0.929	0.933
0.9	0.929	0.936	0.944

Out[ ]=

ρ↓ | n →	30	90	150
0.1	0.683	0.404	0.315
0.5	0.544	0.312	0.241
0.9	0.161	0.083	0.062

Metoda empirické věrohodnosti

In[ ]:=
```
chikvantil = Quantile[ChiSquareDistribution[1], 0.95];
(realizaceEVp3 = Table[Table[EVfunkce[rozsah, kk], {rozsah, 7, 9}], {kk, 7, 9}]) //
 Timing
tabulkauspesnostipokrytiEVp3 = realizaceEVp3[[All, All, 1]];
tabulkaprumernychdelekEVp3 = realizaceEVp3[[All, All, 2]];
Grid[MapThread[Prepend,
  {Prepend[tabulkauspesnostipokrytiEVp3, {n[7], n[8], n[9]}], {DownArrow["ρ", ""] |
     RightArrow["n", ""], N[ρ[7]], N[ρ[8]], N[ρ[9]]}}], Frame → All]
Grid[MapThread[Prepend, {Prepend[tabulkaprumernychdelekEVp3, {n[7], n[8], n[9]}],
   {DownArrow["ρ", ""] | RightArrow["n", ""], N[ρ[7]], N[ρ[8]], N[ρ[9]]}}], Frame → All]
```

... FindRoot: Encountered a singular Jacobian at the point {rho$518304, lambda$518304} = $\{1.14851\times10^{-17}, 0.\}$. Try perturbing the initial point(s).

... FindRoot: Encountered a singular Jacobian at the point {rho$518304, lambda$518304} = $\{1.14851\times10^{-17}, 0.\}$. Try perturbing the initial point(s).

... FindRoot: Encountered a singular Jacobian at the point {rho$518304, lambda$518304} = {0., 0.}. Try perturbing the initial point(s).

... General: Further output of FindRoot::jsing will be suppressed during this calculation.

... FindRoot: The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

Out[ ]=
```
{427.469, {{{0.875, 0.681}, {0.901, 0.421}, {0.93, 0.331}},
  {{0.886, 0.523}, {0.918, 0.326}, {0.931, 0.256}},
  {{0.934, 0.136}, {0.956, 0.083}, {0.954, 0.063}}}}
```

Out[ ]=

ρ↓ | n →	30	90	150
0.1	0.875	0.901	0.93
0.5	0.886	0.918	0.931
0.9	0.934	0.956	0.954

Out[ ]=

ρ↓ | n →	30	90	150
0.1	0.681	0.421	0.331
0.5	0.523	0.326	0.256
0.9	0.136	0.083	0.063

Metoda empirické věrohodnosti s funkcí vlivu

In[∘]:=
```
chikvantil = Quantile[ChiSquareDistribution[1], 0.95];
(realizaceEVFVp3 = Table[Table[EVFVfunkce[rozsah, kk], {rozsah, 7, 9}], {kk, 7, 9}]) //
 Timing
tabulkauspesnostipokrytiEVFVp3 = realizaceEVFVp3[[All, All, 1]];
tabulkaprumernychdelekEVFVp3 = realizaceEVFVp3[[All, All, 2]];
Grid[MapThread[Prepend,
  {Prepend[tabulkauspesnostipokrytiEVFVp3, {n[7], n[8], n[9]}], {DownArrow["ρ", ""] |
     RightArrow["n", ""], N[ρ[7]], N[ρ[8]], N[ρ[9]]}}], Frame → All]
Grid[MapThread[Prepend, {Prepend[tabulkaprumernychdelekEVFVp3, {n[7], n[8], n[9]}],
   {DownArrow["ρ", ""] | RightArrow["n", ""], N[ρ[7]], N[ρ[8]], N[ρ[9]]}}], Frame → All]
```

... FindRoot: Encountered a singular Jacobian at the point {rho$482617, lambda$482617} = {-0.110281, 0.}. Try perturbing the initial point(s).

... FindRoot: Encountered a singular Jacobian at the point {rho$482617, lambda$482617} = {-0.110281, 0.}. Try perturbing the initial point(s).

... FindRoot: Encountered a singular Jacobian at the point {rho$482617, lambda$482617} = {-0.0965812, 0.}. Try perturbing the initial point(s).

... General: Further output of FindRoot::jsing will be suppressed during this calculation.

... FindRoot: The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

... FindRoot: The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

... FindRoot: The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

... General: Further output of FindRoot::lstol will be suppressed during this calculation.

... FindRoot: Failed to converge to the requested accuracy or precision within 100 iterations.

... FindRoot: Failed to converge to the requested accuracy or precision within 100 iterations.

... FindRoot: Failed to converge to the requested accuracy or precision within 100 iterations.

... General: Further output of FindRoot::cvmit will be suppressed during this calculation.

Out[∘]=
```
{1474.16, {{{0.907, 0.633}, {0.926, 0.399}, {0.938, 0.318}},
  {{0.91, 0.514}, {0.926, 0.321}, {0.926, 0.253}},
  {{0.907, 0.149}, {0.942, 0.086}, {0.946, 0.065}}}}
```

Out[∘]=

ρ↓ | n →	30	90	150
0.1	0.907	0.926	0.938
0.5	0.91	0.926	0.926
0.9	0.907	0.942	0.946

Out[∘]=

ρ↓ | n →	30	90	150
0.1	0.633	0.399	0.318
0.5	0.514	0.321	0.253
0.9	0.149	0.086	0.065

Společné tabulky

In[∘]:=
```
uradek1 = tabulkauspesnostipokrytiFZp3 // Flatten;
uradek2 = tabulkauspesnostipokrytiZPp3 // Flatten;
uradek3 = tabulkauspesnostipokrytiEVp3 // Flatten;
uradek4 = tabulkauspesnostipokrytiEVFVp3 // Flatten;
dradek1 = tabulkaprumernychdelekFZp3 // Flatten;
dradek2 = tabulkaprumernychdelekZPp3 // Flatten;
dradek3 = tabulkaprumernychdelekEVp3 // Flatten;
dradek4 = tabulkaprumernychdelekEVFVp3 // Flatten;
tabulka3usp =
  Prepend[MapThread[Prepend, {Prepend[{uradek1, uradek2, uradek3, uradek4},
      Table[n[6 + If[Mod[g, 3] == 0, 3, Mod[g, 3]]], {g, 1, 9}]],
     {DownArrow["metoda", ""] | RightArrow["n", ""], "FZ", "ZP", "EV", "EVFV"}}],
   {RightArrow["ρ", ""], " ", N[ρ[7]], " ", " ", N[ρ[8]], " ", " ", N[ρ[9]], " "}];
Row[{"Tabulka úspěšností pokrytí"}]
Grid[tabulka3usp, Frame → False]
tabulka3d = Prepend[MapThread[Prepend, {Prepend[{dradek1, dradek2, dradek3, dradek4},
      Table[n[6 + If[Mod[g, 3] == 0, 3, Mod[g, 3]]], {g, 1, 9}]],
     {DownArrow["metoda", ""] | RightArrow["n", ""], "FZ", "ZP", "EV", "EVFV"}}],
   {RightArrow["ρ", ""], " ", N[ρ[7]], " ", " ", N[ρ[8]], " ", " ", N[ρ[9]], " "}];
Row[{"Tabulka průměrných délek intervalů"}]
Grid[tabulka3d, Frame → False]
Row[{tabulka3usp // TeXForm, "\n", tabulka3d // TeXForm}]
```

Out[∘]= Tabulka úspěšností pokrytí

Out[∘]=

ρ →		0.1			0.5			0.9	
metoda↓ | n →	30	90	150	30	90	150	30	90	150
FZ	0.946	0.934	0.941	0.934	0.93	0.932	0.93	0.937	0.944
ZP	0.947	0.934	0.942	0.933	0.929	0.933	0.929	0.936	0.944
EV	0.875	0.901	0.93	0.886	0.918	0.931	0.934	0.956	0.954
EVFV	0.907	0.926	0.938	0.91	0.926	0.926	0.907	0.942	0.946

Out[∘]= Tabulka průměrných délek intervalů

Out[∘]=

ρ →		0.1			0.5			0.9	
metoda↓ | n →	30	90	150	30	90	150	30	90	150
FZ	0.692	0.406	0.315	0.547	0.312	0.241	0.158	0.083	0.062
ZP	0.683	0.404	0.315	0.544	0.312	0.241	0.161	0.083	0.062
EV	0.681	0.421	0.331	0.523	0.326	0.256	0.136	0.083	0.063
EVFV	0.633	0.399	0.318	0.514	0.321	0.253	0.149	0.086	0.065

Out[ ]=
```
\left(
\begin{array}{cccccccccc}
 \rho \rightarrow \text{} &   & 0.1 &   &   & 0.5 &   &   & 0.9 &   \\
 \text{metoda}\downarrow \text{}|\text{n}\rightarrow
   \text{} & 30 & 90 & 150 & 30 & 90 & 150 & 30 & 90 & 150 \\
 \text{FZ} & 0.946 & 0.934 & 0.941 & 0.934 & 0.93 & 0.932 & 0.93 & 0.937 & 0.944 \\
 \text{ZP} & 0.947 & 0.934 & 0.942 & 0.933 & 0.929 & 0.933 & 0.929 & 0.936 & 0.944 \\
 \text{EV} & 0.875 & 0.901 & 0.93 & 0.886 & 0.918 & 0.931 & 0.934 & 0.956 & 0.954 \\
 \text{EVFV} & 0.907 & 0.926 &
   0.938 & 0.91 & 0.926 & 0.926 & 0.907 & 0.942 & 0.946 \\
\end{array}
\right)
\left(
\begin{array}{cccccccccc}
 \rho \rightarrow \text{} &   & 0.1 &   &   & 0.5 &   &   & 0.9 &   \\
 \text{metoda}\downarrow \text{}|\text{n}\rightarrow
   \text{} & 30 & 90 & 150 & 30 & 90 & 150 & 30 & 90 & 150 \\
 \text{FZ} & 0.692 & 0.406 & 0.315 & 0.547 & 0.312 & 0.241 & 0.158 & 0.083 & 0.062 \\
 \text{ZP} & 0.683 & 0.404 & 0.315 & 0.544 & 0.312 & 0.241 & 0.161 & 0.083 & 0.062 \\
 \text{EV} & 0.681 & 0.421 & 0.331 & 0.523 & 0.326 & 0.256 & 0.136 & 0.083 & 0.063 \\
 \text{EVFV} & 0.633 & 0.399 &
   0.318 & 0.514 & 0.321 & 0.253 & 0.149 & 0.086 & 0.065 \\
\end{array}
\right)
```

Příklad 4

V tomto příkladu budeme pracovat s náhodnými výběry z dvojrozměrného rovnoměrného rozdělení ze dvou rovnoměrných rozdělení R(0,1) s korelačními koeficienty opět ρ=0.1, 0.5, 0.9.

In[ ]:=
```
n[10] = 30;
n[11] = 90;
n[12] = 150;
ρ[10] = 0.1;
ρ[11] = 0.5;
ρ[12] = 0.9;
pocetopakovani = 1000;
```

```
In[•]:= SeedRandom[2148];
fcea[10] = -5/2 + (1/2) * Sqrt[(ρ[10] + 49) / (ρ[10] + 1)];
fcea[11] = -5/2 + (1/2) * Sqrt[(ρ[11] + 49) / (ρ[11] + 1)];
fcea[12] = -5/2 + (1/2) * Sqrt[(ρ[12] + 49) / (ρ[12] + 1)];
fcex[10] = RandomVariate[UniformDistribution[{0, 1}], pocetopakovani * n[10]];
fcex[11] = RandomVariate[UniformDistribution[{0, 1}], pocetopakovani * n[11]];
fcex[12] = RandomVariate[UniformDistribution[{0, 1}], pocetopakovani * n[12]];
fcev2[10] = RandomVariate[UniformDistribution[{0, 1}], pocetopakovani * n[10]];
fcev2[11] = RandomVariate[UniformDistribution[{0, 1}], pocetopakovani * n[11]];
fcev2[12] = RandomVariate[UniformDistribution[{0, 1}], pocetopakovani * n[12]];
fceu[10, 10] = RandomVariate[BetaDistribution[fcea[10], 1], pocetopakovani * n[10]];
fceu[10, 11] = RandomVariate[BetaDistribution[fcea[11], 1], pocetopakovani * n[10]];
fceu[10, 12] = RandomVariate[BetaDistribution[fcea[12], 1], pocetopakovani * n[10]];
fceu[11, 10] = RandomVariate[BetaDistribution[fcea[10], 1], pocetopakovani * n[11]];
fceu[11, 11] = RandomVariate[BetaDistribution[fcea[11], 1], pocetopakovani * n[11]];
fceu[11, 12] = RandomVariate[BetaDistribution[fcea[12], 1], pocetopakovani * n[11]];
fceu[12, 10] = RandomVariate[BetaDistribution[fcea[10], 1], pocetopakovani * n[12]];
fceu[12, 11] = RandomVariate[BetaDistribution[fcea[11], 1], pocetopakovani * n[12]];
fceu[12, 12] = RandomVariate[BetaDistribution[fcea[12], 1], pocetopakovani * n[12]];
fcey[10, 10] := Table[If[fcev2[10][[q]] < 0.5, Abs[fceu[10, 10][[q]] - fcex[10][[q]]],
    1 - Abs[1 - fceu[10, 10][[q]] - fcex[10][[q]]]], {q, 1, pocetopakovani * n[10]}];
fcey[10, 11] := Table[If[fcev2[10][[q]] < 0.5, Abs[fceu[10, 11][[q]] - fcex[10][[q]]],
    1 - Abs[1 - fceu[10, 11][[q]] - fcex[10][[q]]]], {q, 1, pocetopakovani * n[10]}];
fcey[10, 12] := Table[If[fcev2[10][[q]] < 0.5, Abs[fceu[10, 12][[q]] - fcex[10][[q]]],
    1 - Abs[1 - fceu[10, 12][[q]] - fcex[10][[q]]]], {q, 1, pocetopakovani * n[10]}];
fcey[11, 10] := Table[If[fcev2[11][[q]] < 0.5, Abs[fceu[11, 10][[q]] - fcex[11][[q]]],
    1 - Abs[1 - fceu[11, 10][[q]] - fcex[11][[q]]]], {q, 1, pocetopakovani * n[11]}];
fcey[11, 11] := Table[If[fcev2[11][[q]] < 0.5, Abs[fceu[11, 11][[q]] - fcex[11][[q]]],
    1 - Abs[1 - fceu[11, 11][[q]] - fcex[11][[q]]]], {q, 1, pocetopakovani * n[11]}];
fcey[11, 12] := Table[If[fcev2[11][[q]] < 0.5, Abs[fceu[11, 12][[q]] - fcex[11][[q]]],
    1 - Abs[1 - fceu[11, 12][[q]] - fcex[11][[q]]]], {q, 1, pocetopakovani * n[11]}];
fcey[12, 10] := Table[If[fcev2[12][[q]] < 0.5, Abs[fceu[12, 10][[q]] - fcex[12][[q]]],
    1 - Abs[1 - fceu[12, 10][[q]] - fcex[12][[q]]]], {q, 1, pocetopakovani * n[12]}];
fcey[12, 11] := Table[If[fcev2[12][[q]] < 0.5, Abs[fceu[12, 11][[q]] - fcex[12][[q]]],
    1 - Abs[1 - fceu[12, 11][[q]] - fcex[12][[q]]]], {q, 1, pocetopakovani * n[12]}];
fcey[12, 12] := Table[If[fcev2[12][[q]] < 0.5, Abs[fceu[12, 12][[q]] - fcex[12][[q]]],
    1 - Abs[1 - fceu[12, 12][[q]] - fcex[12][[q]]]], {q, 1, pocetopakovani * n[12]}];
vyber[10, 10] = {fcex[10], fcey[10, 10]} // Transpose;
vyber[10, 11] = {fcex[10], fcey[10, 11]} // Transpose;
vyber[10, 12] = {fcex[10], fcey[10, 12]} // Transpose;
vyber[11, 10] = {fcex[11], fcey[11, 10]} // Transpose;
vyber[11, 11] = {fcex[11], fcey[11, 11]} // Transpose;
vyber[11, 12] = {fcex[11], fcey[11, 12]} // Transpose;
vyber[12, 10] = {fcex[12], fcey[12, 10]} // Transpose;
vyber[12, 11] = {fcex[12], fcey[12, 11]} // Transpose;
vyber[12, 12] = {fcex[12], fcey[12, 12]} // Transpose;
```

Fisherova z-transformace

In[ ]:=
```
u0.025 = Quantile[NormalDistribution[0, 1], 0.025];
realizaceFZp4 = Table[Table[FZfunkce[rozsah, kk], {rozsah, 10, 12}], {kk, 10, 12}];
tabulkauspesnostipokrytiFZp4 = realizaceFZp4[[All, All, 1]];
tabulkaprumernychdelekFZp4 = realizaceFZp4[[All, All, 2]];
Grid[MapThread[Prepend, {Prepend[tabulkauspesnostipokrytiFZp4, {n[10], n[11], n[12]}],
   {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[10], ρ[11], ρ[12]}}], Frame → All]
Grid[MapThread[Prepend, {Prepend[tabulkaprumernychdelekFZp4, {n[10], n[11], n[12]}],
   {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[10], ρ[11], ρ[12]}}], Frame → All]
Clear[
  i];
```

Out[ ]=

ρ↓ | n →	30	90	150
0.1	0.942	0.933	0.945
0.5	0.906	0.891	0.892
0.9	0.574	0.591	0.612

Out[ ]=

ρ↓ | n →	30	90	150
0.1	0.691	0.405	0.315
0.5	0.540	0.310	0.240
0.9	0.141	0.080	0.062

Metoda zobecněných pivotů

In[ ]:=
```
SeedRandom[1326];
(realizaceZPp4 = Table[Table[ZPfunkce[rozsah, kk], {rozsah, 10, 12}], {kk, 10, 12}]) //
 Timing
tabulkauspesnostipokrytiZPp4 = realizaceZPp4[[All, All, 1]];
tabulkaprumernychdelekZPp4 = realizaceZPp4[[All, All, 2]];
Grid[MapThread[Prepend, {Prepend[tabulkauspesnostipokrytiZPp4, {n[10], n[11], n[12]}],
   {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[10], ρ[11], ρ[12]}}], Frame → All]
Grid[MapThread[Prepend, {Prepend[tabulkaprumernychdelekZPp4, {n[10], n[11], n[12]}],
   {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[10], ρ[11], ρ[12]}}], Frame → All]
Clear[
  i];
```

Out[ ]=
```
{4066.13, {{{0.944, 0.682}, {0.935, 0.403}, {0.945, 0.314}},
  {{0.907, 0.536}, {0.89, 0.309}, {0.894, 0.240}},
  {{0.564, 0.143}, {0.592, 0.080}, {0.609, 0.062}}}}
```

Out[ ]=

ρ↓ | n →	30	90	150
0.1	0.944	0.935	0.945
0.5	0.907	0.89	0.894
0.9	0.564	0.592	0.609

Out[ ]=

ρ↓ | n →	30	90	150
0.1	0.682	0.403	0.314
0.5	0.536	0.309	0.240
0.9	0.143	0.080	0.062

Metoda empirické věrohodnosti

In[∘]:=
```
chikvantil = Quantile[ChiSquareDistribution[1], 0.95];
(realizaceEVp4 = Table[Table[EVfunkce[rozsah, kk], {rozsah, 10, 12}], {kk, 10, 12}]) //
 Timing
tabulkauspesnostipokrytiEVp4 = realizaceEVp4[[All, All, 1]];
tabulkaprumernychdelekEVp4 = realizaceEVp4[[All, All, 2]];
Grid[MapThread[Prepend, {Prepend[tabulkauspesnostipokrytiEVp4, {n[10], n[11], n[12]}],
   {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[10], ρ[11], ρ[12]}}], Frame → All]
Grid[MapThread[Prepend, {Prepend[tabulkaprumernychdelekEVp4, {n[10], n[11], n[12]}],
   {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[10], ρ[11], ρ[12]}}], Frame → All]
```

... FindRoot: The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

Out[∘]=
```
{699.563, {{{0.901, 0.682}, {0.925, 0.411}, {0.945, 0.320}},
  {{0.907, 0.590}, {0.934, 0.357}, {0.944, 0.281}},
  {{0.776, 0.225}, {0.872, 0.162}, {0.91, 0.132}}}}
```

Out[∘]=

ρ↓ | n →	30	90	150
0.1	0.901	0.925	0.945
0.5	0.907	0.934	0.944
0.9	0.776	0.872	0.91

Out[∘]=

ρ↓ | n →	30	90	150
0.1	0.682	0.411	0.320
0.5	0.590	0.357	0.281
0.9	0.225	0.162	0.132

Metoda empirické věrohodnosti s funkcí vlivu

In[•]:=
```
chikvantil = Quantile[ChiSquareDistribution[1], 0.95];
(realizaceEVFVp4 =
   Table[Table[EVFVfunkce[rozsah, kk], {rozsah, 10, 12}], {kk, 10, 12}]) // Timing
tabulkauspesnostipokrytiEVFVp4 = realizaceEVFVp4[[All, All, 1]];
tabulkaprumernychdelekEVFVp4 = realizaceEVFVp4[[All, All, 2]];
Grid[
 MapThread[Prepend, {Prepend[tabulkauspesnostipokrytiEVFVp4, {n[10], n[11], n[12]}],
   {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[10], ρ[11], ρ[12]}}], Frame → All]
Grid[MapThread[Prepend, {Prepend[tabulkaprumernychdelekEVFVp4, {n[10], n[11], n[12]}],
   {DownArrow["ρ", ""] | RightArrow["n", ""], ρ[10], ρ[11], ρ[12]}}], Frame → All]
```

... FindRoot: Encountered a singular Jacobian at the point {rho$359089, lambda$359089} = {−0.136244, 0.}. Try perturbing the initial point(s).

... FindRoot: Encountered a singular Jacobian at the point {rho$359089, lambda$359089} = {−0.136244, 0.}. Try perturbing the initial point(s).

... FindRoot: Encountered a singular Jacobian at the point {rho$359089, lambda$359089} = {0.238586, 0.}. Try perturbing the initial point(s).

... General: Further output of FindRoot::jsing will be suppressed during this calculation.

... FindRoot: The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

... FindRoot: Failed to converge to the requested accuracy or precision within 100 iterations.

... FindRoot: The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

... FindRoot: The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

... General: Further output of FindRoot::lstol will be suppressed during this calculation.

... FindRoot: Failed to converge to the requested accuracy or precision within 100 iterations.

... FindRoot: Failed to converge to the requested accuracy or precision within 100 iterations.

... General: Further output of FindRoot::cvmit will be suppressed during this calculation.

Out[•]=
```
{2394.09, {{{0.932, 0.661}, {0.928, 0.405}, {0.948, 0.318}},
  {{0.92, 0.583}, {0.942, 0.357}, {0.95, 0.280}},
  {{0.74, 0.241}, {0.887, 0.171}, {0.916, 0.138}}}}
```

Out[•]=

ρ↓ | n →	30	90	150
0.1	0.932	0.928	0.948
0.5	0.92	0.942	0.95
0.9	0.74	0.887	0.916

Out[•]=

ρ↓ | n →	30	90	150
0.1	0.661	0.405	0.318
0.5	0.583	0.357	0.280
0.9	0.241	0.171	0.138

Společné tabulky

In[∘]:=
```
uradek1 = tabulkauspesnostipokrytiFZp4 // Flatten;
uradek2 = tabulkauspesnostipokrytiZPp4 // Flatten;
uradek3 = tabulkauspesnostipokrytiEVp4 // Flatten;
uradek4 = tabulkauspesnostipokrytiEVFVp4 // Flatten;
dradek1 = tabulkaprumernychdelekFZp4 // Flatten;
dradek2 = tabulkaprumernychdelekZPp4 // Flatten;
dradek3 = tabulkaprumernychdelekEVp4 // Flatten;
dradek4 = tabulkaprumernychdelekEVFVp4 // Flatten;
tabulka4usp =
  Prepend[MapThread[Prepend, {Prepend[{uradek1, uradek2, uradek3, uradek4},
       Table[n[9 + If[Mod[g, 3] == 0, 3, Mod[g, 3]]], {g, 1, 9}]],
      {DownArrow["metoda", ""] | RightArrow["n", ""], "FZ", "ZP", "EV", "EVFV"}}],
    {RightArrow["ρ", ""], " ", ρ[10], " ", " ", ρ[11], " ", " ", ρ[12], " "}];
Row[{"Tabulka úspěšností pokrytí"}]
Grid[tabulka4usp, Frame → False]
tabulka4d = Prepend[MapThread[Prepend, {Prepend[{dradek1, dradek2, dradek3, dradek4},
       Table[n[9 + If[Mod[g, 3] == 0, 3, Mod[g, 3]]], {g, 1, 9}]],
      {DownArrow["metoda", ""] | RightArrow["n", ""], "FZ", "ZP", "EV", "EVFV"}}],
    {RightArrow["ρ", ""], " ", ρ[10], " ", " ", ρ[11], " ", " ", ρ[12], " "}];
Row[{"Tabulka průměrných délek intervalů"}]
Grid[tabulka4d, Frame → False]
Row[{tabulka4usp // TeXForm, "\n", tabulka4d // TeXForm}]
```

Out[∘]= Tabulka úspěšností pokrytí

Out[∘]=

ρ →		0.1			0.5			0.9	
metoda↓ | n →	30	90	150	30	90	150	30	90	150
FZ	0.942	0.933	0.945	0.906	0.891	0.892	0.574	0.591	0.612
ZP	0.944	0.935	0.945	0.907	0.89	0.894	0.564	0.592	0.609
EV	0.901	0.925	0.945	0.907	0.934	0.944	0.776	0.872	0.91
EVFV	0.932	0.928	0.948	0.92	0.942	0.95	0.74	0.887	0.916

Out[∘]= Tabulka průměrných délek intervalů

Out[∘]=

ρ →		0.1			0.5			0.9	
metoda↓ | n →	30	90	150	30	90	150	30	90	150
FZ	0.691	0.405	0.315	0.540	0.310	0.240	0.141	0.080	0.062
ZP	0.682	0.403	0.314	0.536	0.309	0.240	0.143	0.080	0.062
EV	0.682	0.411	0.320	0.590	0.357	0.281	0.225	0.162	0.132
EVFV	0.661	0.405	0.318	0.583	0.357	0.280	0.241	0.171	0.138

Out[ ]=
```
\left(
\begin{array}{cccccccccc}
 \rho \rightarrow \text{} &   & 0.1 &   &   & 0.5 &   &   & 0.9 &   \\
 \text{metoda}\downarrow \text{}|\text{n}\rightarrow
   \text{} & 30 & 90 & 150 & 30 & 90 & 150 & 30 & 90 & 150 \\
 \text{FZ} & 0.942 & 0.933 & 0.945 & 0.906 & 0.891 & 0.892 & 0.574 & 0.591 & 0.612 \\
 \text{ZP} & 0.944 & 0.935 & 0.945 & 0.907 & 0.89 & 0.894 & 0.564 & 0.592 & 0.609 \\
 \text{EV} & 0.901 & 0.925 & 0.945 & 0.907 & 0.934 & 0.944 & 0.776 & 0.872 & 0.91 \\
 \text{EVFV} & 0.932 & 0.928 & 0.948 & 0.92 & 0.942 & 0.95 & 0.74 & 0.887 & 0.916 \\
\end{array}
\right)
\left(
\begin{array}{cccccccccc}
 \rho \rightarrow \text{} &   & 0.1 &   &   & 0.5 &   &   & 0.9 &   \\
 \text{metoda}\downarrow \text{}|\text{n}\rightarrow
   \text{} & 30 & 90 & 150 & 30 & 90 & 150 & 30 & 90 & 150 \\
 \text{FZ} & 0.691 & 0.405 & 0.315 & 0.540 & 0.310 & 0.240 & 0.141 & 0.080 & 0.062 \\
 \text{ZP} & 0.682 & 0.403 & 0.314 & 0.536 & 0.309 & 0.240 & 0.143 & 0.080 & 0.062 \\
 \text{EV} & 0.682 & 0.411 & 0.320 & 0.590 & 0.357 & 0.281 & 0.225 & 0.162 & 0.132 \\
 \text{EVFV} & 0.661 & 0.405 &
   0.318 & 0.583 & 0.357 & 0.280 & 0.241 & 0.171 & 0.138 \\
\end{array}
\right)
```

Příklad 5

V tomto příkladu budeme simulovat náhodné výběry nejprve dvou nezávislých náhodných veličin se spojitými rovnoměrnými rozděleními R(-1, 1) a R(0, 2), poté náhodné výběry dvou nezávislých náhodných veličin s chi-kvadrat rozděleními s 1 a s 2 stupni volnosti a nakonec náhodné výběry jedné náhodné veličiny s normovaným normálním rozdělením a jedné s Poissonovým rozdělení se střední hodnotou 1, které budou opět nezávislé. Pro každou tuto dvojici opět zvolíme rozsahy výběrů 30,90,150.

In[ ]:=
```
n[13] = 30;
n[14] = 90;
n[15] = 150;
ρ[13] = 0;
ρ[14] = 0;
ρ[15] = 0;
pocetopakovani = 1000;
```

In[∘]:=
```
SeedRandom[1305];
vyber[13, 13] = {RandomVariate[UniformDistribution[{-1, 1}], pocetopakovani * n[13]],
    RandomVariate[UniformDistribution[{0, 2}], pocetopakovani * n[13]]} // Transpose;
vyber[14, 13] = {RandomVariate[UniformDistribution[{-1, 1}], pocetopakovani * n[14]],
    RandomVariate[UniformDistribution[{0, 2}], pocetopakovani * n[14]]} // Transpose;
vyber[15, 13] = {RandomVariate[UniformDistribution[{-1, 1}], pocetopakovani * n[15]],
    RandomVariate[UniformDistribution[{0, 2}], pocetopakovani * n[15]]} // Transpose;
vyber[13, 14] = {RandomVariate[ChiSquareDistribution[1], pocetopakovani * n[13]],
    RandomVariate[ChiSquareDistribution[2], pocetopakovani * n[13]]} // Transpose;
vyber[14, 14] = {RandomVariate[ChiSquareDistribution[1], pocetopakovani * n[14]],
    RandomVariate[ChiSquareDistribution[2], pocetopakovani * n[14]]} // Transpose;
vyber[15, 14] = {RandomVariate[ChiSquareDistribution[1], pocetopakovani * n[15]],
    RandomVariate[ChiSquareDistribution[2], pocetopakovani * n[15]]} // Transpose;
vyber[13, 15] = N[{RandomVariate[NormalDistribution[0, 1], pocetopakovani * n[13]],
     RandomVariate[PoissonDistribution[1], pocetopakovani * n[13]]} // Transpose];
vyber[14, 15] = N[{RandomVariate[NormalDistribution[0, 1], pocetopakovani * n[14]],
     RandomVariate[PoissonDistribution[1], pocetopakovani * n[14]]} // Transpose];
vyber[15, 15] = N[{RandomVariate[NormalDistribution[0, 1], pocetopakovani * n[15]],
     RandomVariate[PoissonDistribution[1], pocetopakovani * n[15]]} // Transpose];
```

Fisherova z-transformace

In[∘]:=
```
u0.025 = Quantile[NormalDistribution[0, 1], 0.025];
realizaceFZp5 = Table[Table[FZfunkce[rozsah, kk], {rozsah, 13, 15}], {kk, 13, 15}];
tabulkauspesnostipokrytiFZp5 = realizaceFZp5[[All, All, 1]];
tabulkaprumernychdelekFZp5 = realizaceFZp5[[All, All, 2]];
Grid[MapThread[Prepend, {Prepend[tabulkauspesnostipokrytiFZp5, {n[13], n[14], n[15]}],
   {DownArrow["rozdělení", ""] | RightArrow["n", ""],
    "R(-1,1)  a R(0,2)", "χ1² a χ2²", "N(0,1)  a Po(1)"}}], Frame → All]
Grid[MapThread[Prepend, {Prepend[tabulkaprumernychdelekFZp5, {n[13], n[14], n[15]}],
   {DownArrow["rozdělení", ""] | RightArrow["n", ""],
    "R(-1,1)  a R(0,2)", "χ1² a χ2²", "N(0,1)  a Po(1)"}}], Frame → All]
Clear[
  i];
```

Out[∘]=

rozdělení↓ | n →	30	90	150
R(−1,1) a R(0,2)	0.954	0.955	0.957
χ_1^2 a χ_2^2	0.944	0.945	0.95
N(0,1) a Po(1)	0.938	0.94	0.951

Out[∘]=

rozdělení↓ | n →	30	90	150
R(−1,1) a R(0,2)	0.698	0.410	0.319
χ_1^2 a χ_2^2	0.697	0.410	0.319
N(0,1) a Po(1)	0.698	0.409	0.318

Metoda zobecněných pivotů

In[ ]:=
```
SeedRandom[1107];
(realizaceZPp5 = Table[Table[ZPfunkce[rozsah, kk], {rozsah, 13, 15}], {kk, 13, 15}]) //
 Timing
tabulkauspesnostipokrytiZPp5 = realizaceZPp5[[All, All, 1]];
tabulkaprumernychdelekZPp5 = realizaceZPp5[[All, All, 2]];
Grid[MapThread[Prepend, {Prepend[tabulkauspesnostipokrytiZPp5, {n[13], n[14], n[15]}],
   {DownArrow["rozdělení", ""] | RightArrow["n", ""],
    "R(-1,1)  a R(0,2)", "χ1^2 a χ2^2", "N(0,1)  a Po(1)"}}], Frame → All]
Grid[MapThread[Prepend, {Prepend[tabulkaprumernychdelekZPp5, {n[13], n[14], n[15]}],
   {DownArrow["rozdělení", ""] | RightArrow["n", ""],
    "R(-1,1)  a R(0,2)", "χ1^2 a χ2^2", "N(0,1)  a Po(1)"}}], Frame → All]
Clear[
  i];
```

Out[ ]=
```
{4015.88, {{{0.953, 0.689}, {0.955, 0.408}, {0.957, 0.318}},
  {{0.948, 0.688}, {0.946, 0.408}, {0.951, 0.318}},
  {{0.939, 0.688}, {0.941, 0.408}, {0.953, 0.318}}}}
```

Out[ ]=

rozdělení ↓ | n →	30	90	150
R(−1,1) a R(0,2)	0.953	0.955	0.957
χ_1^2 a χ_2^2	0.948	0.946	0.951
N(0,1) a Po(1)	0.939	0.941	0.953

Out[ ]=

rozdělení ↓ | n →	30	90	150
R(−1,1) a R(0,2)	0.689	0.408	0.318
χ_1^2 a χ_2^2	0.688	0.408	0.318
N(0,1) a Po(1)	0.688	0.408	0.318

Metoda empirické věrohodnosti

In[∘]:=
```
(realizaceEVp5 = Table[Table[EVfunkce[rozsah, kk], {rozsah, 13, 15}], {kk, 13, 15}]) //
  Timing
tabulkauspesnostipokrytiEVp5 = realizaceEVp5[[All, All, 1]];
tabulkaprumernychdelekEVp5 = realizaceEVp5[[All, All, 2]];
Grid[MapThread[Prepend, {Prepend[tabulkauspesnostipokrytiEVp5, {n[13], n[14], n[15]}],
    {DownArrow["rozdělení", ""] | RightArrow["n", ""],
     "R(-1,1)  a R(0,2)", "χ₁² a χ₂²", "N(0,1)  a Po(1)"}}], Frame → All]
Grid[MapThread[Prepend, {Prepend[tabulkaprumernychdelekEVp5, {n[13], n[14], n[15]}],
    {DownArrow["rozdělení", ""] | RightArrow["n", ""],
     "R(-1,1)  a R(0,2)", "χ₁² a χ₂²", "N(0,1)  a Po(1)"}}], Frame → All]
```

Out[∘]=
```
{748.719, {{{0.912, 0.679}, {0.948, 0.409}, {0.954, 0.318}},
  {{0.767, 0.599}, {0.857, 0.396}, {0.904, 0.314}},
  {{0.858, 0.674}, {0.913, 0.413}, {0.929, 0.326}}}}
```

Out[∘]=

rozdělení ↓ | n →	30	90	150
R(−1,1) a R(0,2)	0.912	0.948	0.954
χ_1^2 a χ_2^2	0.767	0.857	0.904
N(0,1) a Po(1)	0.858	0.913	0.929

Out[∘]=

rozdělení ↓ | n →	30	90	150
R(−1,1) a R(0,2)	0.679	0.409	0.318
χ_1^2 a χ_2^2	0.599	0.396	0.314
N(0,1) a Po(1)	0.674	0.413	0.326

Metoda empirické věrohodnosti s funkcí vlivu

In[ ]:=
```
(realizaceEVFVp5 =
   Table[Table[EVFVfunkce[rozsah, kk], {rozsah, 13, 15}], {kk, 13, 15}]) // Timing
tabulkauspesnostipokrytiEVFVp5 = realizaceEVFVp5[[All, All, 1]];
tabulkaprumernychdelekEVFVp5 = realizaceEVFVp5[[All, All, 2]];
Grid[
 MapThread[Prepend, {Prepend[tabulkauspesnostipokrytiEVFVp5, {n[13], n[14], n[15]}],
   {DownArrow["rozdělení", ""] | RightArrow["n", ""],
    "R(-1,1) a R(0,2)", "χ₁² a χ₂²", "N(0,1) a Po(1)"}}], Frame → All]
Grid[MapThread[Prepend, {Prepend[tabulkaprumernychdelekEVFVp5, {n[13], n[14], n[15]}],
   {DownArrow["rozdělení", ""] | RightArrow["n", ""],
    "R(-1,1) a R(0,2)", "χ₁² a χ₂²", "N(0,1) a Po(1)"}}], Frame → All]
```

... FindRoot: Encountered a singular Jacobian at the point {rho$78662, lambda$78662} = {-0.139036, 0.}. Try perturbing the initial point(s).

... FindRoot: Encountered a singular Jacobian at the point {rho$78662, lambda$78662} = {-0.139036, 0.}. Try perturbing the initial point(s).

... FindRoot: Encountered a singular Jacobian at the point {rho$78662, lambda$78662} = {-0.180591, 0.}. Try perturbing the initial point(s).

... General: Further output of FindRoot::jsing will be suppressed during this calculation.

... FindRoot: The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

... FindRoot: The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

... FindRoot: The line search decreased the step size to within tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the merit function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

... General: Further output of FindRoot::lstol will be suppressed during this calculation.

... FindRoot: Failed to converge to the requested accuracy or precision within 100 iterations.

... FindRoot: Failed to converge to the requested accuracy or precision within 100 iterations.

... FindRoot: Failed to converge to the requested accuracy or precision within 100 iterations.

... General: Further output of FindRoot::cvmit will be suppressed during this calculation.

Out[ ]=
```
{1615.97, {{{0.939, 0.658}, {0.953, 0.404}, {0.959, 0.315}},
  {{0.87, 0.546}, {0.902, 0.362}, {0.927, 0.292}},
  {{0.902, 0.629}, {0.93, 0.395}, {0.937, 0.315}}}}
```

Out[ ]=

rozdělení↓ | n →	30	90	150
R(-1,1) a R(0,2)	0.939	0.953	0.959
χ_1^2 a χ_2^2	0.87	0.902	0.927
N(0,1) a Po(1)	0.902	0.93	0.937

Out[•]=

rozdělení ↓ | n →	30	90	150
R(-1,1) a R(0,2)	0.658	0.404	0.315
χ_1^2 a χ_2^2	0.546	0.362	0.292
N(0,1) a Po(1)	0.629	0.395	0.315

Společné tabulky

In[•]:=

```
uradek1 = tabulkauspesnostipokrytiFZp5 // Flatten;
uradek2 = tabulkauspesnostipokrytiZPp5 // Flatten;
uradek3 = tabulkauspesnostipokrytiEVp5 // Flatten;
uradek4 = tabulkauspesnostipokrytiEVFVp5 // Flatten;
dradek1 = tabulkaprumernychdelekFZp5 // Flatten;
dradek2 = tabulkaprumernychdelekZPp5 // Flatten;
dradek3 = tabulkaprumernychdelekEVp5 // Flatten;
dradek4 = tabulkaprumernychdelekEVFVp5 // Flatten;
tabulka5usp =
  Prepend[MapThread[Prepend, {Prepend[{uradek1, uradek2, uradek3, uradek4},
      Table[n[12 + If[Mod[g, 3] == 0, 3, Mod[g, 3]]], {g, 1, 9}]],
     {DownArrow["metoda", ""] | RightArrow["n", ""], "FZ", "ZP", "EV", "EVFV"}}],
   {RightArrow["rozdělení", ""], " ", "R(-1,1) a R(0,2)", " ", " ",
    "χ₁² a χ₂²", " ", " ", "N(0,1) a Po(1)", " "}];
Row[{"Tabulka úspěšností pokrytí"}]
Grid[tabulka5usp, Frame → False]
tabulka5d = Prepend[MapThread[Prepend, {Prepend[{dradek1, dradek2, dradek3, dradek4},
      Table[n[12 + If[Mod[g, 3] == 0, 3, Mod[g, 3]]], {g, 1, 9}]],
     {DownArrow["metoda", ""] | RightArrow["n", ""], "FZ", "ZP", "EV", "EVFV"}}],
   {RightArrow["rozdělení", ""], " ", "R(-1,1) a R(0,2)", " ", " ",
    "χ₁² a χ₂²", " ", " ", "N(0,1) a Po(1)", " "}];
Row[{"Tabulka průměrných délek intervalů"}]
Grid[tabulka5d, Frame → False]
Row[{tabulka5usp // TeXForm, "\n", tabulka5d // TeXForm}]
```

Out[•]= Tabulka úspěšností pokrytí

Out[•]=

rozdělení →		R(-1,1) a R(0,2)			χ_1^2 a χ_2^2			N(0,1) a Po(1)	
metoda ↓ | n →	30	90	150	30	90	150	30	90	150
FZ	0.954	0.955	0.957	0.944	0.945	0.95	0.938	0.94	0.951
ZP	0.953	0.955	0.957	0.948	0.946	0.951	0.939	0.941	0.953
EV	0.912	0.948	0.954	0.767	0.857	0.904	0.858	0.913	0.929
EVFV	0.939	0.953	0.959	0.87	0.902	0.927	0.902	0.93	0.937

Out[•]= Tabulka průměrných délek intervalů

Out[•]=

rozdělení →		R(-1,1) a R(0,2)			χ_1^2 a χ_2^2			N(0,1) a Po(1)	
metoda ↓ | n →	30	90	150	30	90	150	30	90	150
FZ	0.698	0.410	0.319	0.697	0.410	0.319	0.698	0.409	0.318
ZP	0.689	0.408	0.318	0.688	0.408	0.318	0.688	0.408	0.318
EV	0.679	0.409	0.318	0.599	0.396	0.314	0.674	0.413	0.326
EVFV	0.658	0.404	0.315	0.546	0.362	0.292	0.629	0.395	0.315

Out[ ]=
```
\left(
\begin{array}{cccccccccc}
 \text{rozd$\unicode{011b}$len{\' \i}}\rightarrow
   \text{} &   & \text{R(-1,1) a R(0,2)} &   &   & \chi
   _1^2\text{ a }\chi _2^2 &   &   & \text{N(0,1) a Po(1)} &   \\
 \text{metoda}\downarrow \text{}|\text{n}\rightarrow \text{}
   & 30 & 90 & 150 & 30 & 90 & 150 & 30 & 90 & 150 \\
 \text{FZ} & 0.954 & 0.955 & 0.957 & 0.944 & 0.945 & 0.95 & 0.938 & 0.94 & 0.951 \\
 \text{ZP} & 0.953 & 0.955 & 0.957 & 0.948 & 0.946 & 0.951 & 0.939 & 0.941 & 0.953 \\
 \text{EV} & 0.912 & 0.948 & 0.954 & 0.767 & 0.857 & 0.904 & 0.858 & 0.913 & 0.929 \\
 \text{EVFV} & 0.939 & 0.953 & 0.959 & 0.87 & 0.902 & 0.927 & 0.902 & 0.93 & 0.937 \\
\end{array}
\right)
\left(
\begin{array}{cccccccccc}
 \text{rozd$\unicode{011b}$len{\' \i}}\rightarrow
   \text{} &   & \text{R(-1,1) a R(0,2)} &   &   & \chi
   _1^2\text{ a }\chi _2^2 &   &   & \text{N(0,1) a Po(1)} &   \\
 \text{metoda}\downarrow \text{}|\text{n}\rightarrow \text{}
   & 30 & 90 & 150 & 30 & 90 & 150 & 30 & 90 & 150 \\
 \text{FZ} & 0.698 & 0.410 & 0.319 & 0.697 & 0.410 & 0.319 & 0.698 & 0.409 & 0.318 \\
 \text{ZP} & 0.689 & 0.408 & 0.318 & 0.688 & 0.408 & 0.318 & 0.688 & 0.408 & 0.318 \\
 \text{EV} & 0.679 & 0.409 & 0.318 & 0.599 & 0.396 & 0.314 & 0.674 & 0.413 & 0.326 \\
 \text{EVFV} & 0.658 & 0.404 &
   0.315 & 0.546 & 0.362 & 0.292 & 0.629 & 0.395 & 0.315 \\
\end{array}
\right)
```