# 4. Punkthinnangute teooria

## Empiiriliste momentide meetod

*p*-järku moment punkti *a* suhtes $\int_{-\infty}^{+\infty} (x-a)^p f(x)dx$.

on avaldise $(x-a)^p$ keskväärtus. Vastav empiiriline moment on sama avaldise aritmeetiline keskmine üle valimi

$$\frac{\sum_{i=1}^{n}(x_i-a)^p}{n}.$$

Empiiriline esimest järku algmoment $m$ ($p = 1$, $a = 0$) on aritmeetiline keskmine. Intuitsioon ütleb, et see võiks sobida keskväärtuse hinnanguks. Teist järku tsentraalmomendi ehk dispersiooni hinnanguks võiks valida teist järku empiirilise tsentraalmomendi

$$s^2 = \frac{\sum_{i=1}^{n}(x_i-m)^2}{n}.$$

NB! Selles avaldises on probleemne keskväärtuse $\mu$ suvaline asendamine hinnanguga *m*.

Sageli arvutatakse lisaks veel kolmandat järku moment, mis iseloomustab tihedusfunktsiooni asümmeetriat ja neljandat järku moment, mis iseloomustab tihedusfunktsiooni ekstsessi.

## Hinnangufunktsiooni omadused

Valim $\boldsymbol{x} = \{x_1, x_2, \ldots, x_n\}$, hinnatav $\xi$, hinnangufunktsioon $T(\xi)$.

Hinnangu nihe $\mathrm{E}T(\boldsymbol{x}) - \xi$.

Hinnangu mõjusus ehk konsistentsus.

Hinnangu hajuvus ja efektiivsus $\mathrm{E}\{T(\boldsymbol{x}) - \xi\}^2$.

Hinnangu arvutamise kulud.

Parima hinnangufunktsiooni probleem.

## Suurima tõepära meetod

Kui kahe sõltumatu juhusliku sündmuse tõenäosused on $P_A$ ja $P_B$, siis nende koosjuhtumise tõenäosus on $P_A \times P_B$.

Kui kahe sõltumatu juhusliku suuruse tõenäostihedused on $f_A(x)$ ja $f_B(x)$, siis nende kombinatsiooni tõenäosustihedus on $f_A(x) \times f_B(x)$.

Kui juhusliku suuruse tõenäosustihedus on $f_x(x;\,\xi)$, siis sõltumatute komponentidega valimi tõenäosustihedus on $f_{\boldsymbol{x}}(\boldsymbol{x};\,\xi) = \prod_{i=1}^{n} f_x(x_i;\xi)$.

Konkreetse hinnangu tõepärasus $\mathrm{L}(\xi;\,\boldsymbol{x}) = f_{\boldsymbol{x}}(\boldsymbol{x};\,\xi)$ .

Meetod: valim $\boldsymbol{x}$ on antud, otsida $\xi$ mille puhul $L = L_{\max}$.

Võte: $l(\xi;\,\boldsymbol{x}) = \ln \mathrm{L}(\xi;\,\boldsymbol{x}) = \sum_{i=1}^{n} \ln f_x(x_i;\xi)$.

*Näide*: Poissoni jaotuse parameetri $\mu$ hinnang $\lambda$

$$l(\lambda;\mathbf{x}) = \sum_{i=1}^{n} \ln\left(\frac{\lambda^{x_i}}{x_i!} e^{-\lambda}\right) = \sum_{i=1}^{n} \left(x_i \ln\lambda - \ln(x_i!) - \lambda\right) ,$$

*Ülesanne*: lahendada võrrand $\mathrm{d}l/\mathrm{d}\lambda = 0$ .

## Vähimruutude meetod

*Näide*: Suuruse $\xi$ korduva mõõtmise tulemused on $\{x_1, x_2, \ldots, x_n\}$.

Üksikmõõtmise hälve $(x_i - \xi)$. Hälvete hulga mõõt $S = \sum_{i=1}^{n} (x_i - \xi)^k$

Antud $\boldsymbol{x}$ , leida $\xi$ nii, et $S = S_{\min}$ . Lahendada juhul $k = 2$ (miks 2?)

*Põhimõte*: hinnang leitakse nii, et hälvete ruutude summa $\Rightarrow$ min.

## Mõõtmistulemuste keskpunkti ja hajuvuse hindamine

Tsentraalne piirteoreem, keskmiseusk ja Cauchy hoiatused.

Keskväärtuse efektiivne hinnang $m$ normaaljaotuse korral
ja normaaljaotuse eriomadused. Hinnangu dispersioon.

Keskväärtuse efektiivne hinnang ühtlase jaotuse korral.

Dispersiooni ja standardhälbe hinnangud normaaljaotuse korral.

$\sigma(m)$ hinnang w/n ja n++ kompensatsioon.

# 5. Punkthinnangute arvutamine

Klassikaline lähenemisviis: keskmistamine vähendab juhuslikke mõõtmisvigu, järelikult tuleks mõõta nii palju kui jõuab ja kogunev andmestik hiljem keskmistamise teel kokku suruda.

Probleemid:

- müra normaaljaotus ei ole garanteeritud ja aritmeetiline keskmine võib osutuda halvaks hinnanguks,
- müra tegeliku jaotusseaduse kohta ei ole eelinformatsiooni,
- vaheandmete suur maht koormab arvutiressursse ja piirab sellega mõõtmissagedust.

Siit tulenevad soovitused:

- kasutada võimalikult robastseid hinnanguid,
- teostada eeltöötlus samaaegselt mõõtmistega ("on-line" meetod).

Hinnangu robastsuse tagavad:

- võimalike jaotuste hulga kohta tehtud eeldustele tuginevad meetodid,
- klassikalised suurte hälvete elimineerimise meetodid,
- mitteparameetrilised meetodid,
- totaalse tsensuuri meetodid.

## Klassikalised on-line hinnangud

Aritmeetilise keskmise arvutamisel pole tarvis vaheandmeid salvestada. Kui üksikmõõtmise tulemus on $x$, siis mõõtmiste käigus kogutakse $\Sigma x$ ja kas hiljem või kohe arvutatakse $\bar{x} = \Sigma x / n$ (vaata DEMO1 protseduurid Teine ja Kolmas).

Tõenäosusteooria käsiraamatutest võib leida kõigi tsentraalmomentide avaldised algmomentide kaudu. Iga algmomenti saab aga arvutada samuti kui aritmeetilist keskmist andmeid säilitamata.

*Ülesanne*: kuidas leida standardhälbe hinnang kolme summa $n = \Sigma 1$, $\Sigma x$ ja $\Sigma xx$ järgi? Lahendus leidub näites DEMO1. Mõõtmise ajal kogutakse kolm summat väärtustest 1, $x$, $xx$. Pärast mõõtmisi arvutatakse:

$$n = \Sigma 1, \quad \bar{x} = \Sigma x / n, \quad \hat{s}_x^2 = \Sigma xx / n - \bar{x}^2 .$$

(Kas see on ikka sama algoritm, kui programmis DEMO1?)

## Tsensuuri tehnoloogia

Massiivi minimaalse ja maksimaalse elemendi leidimise algoritm.
Üheastmeline ekstremaalväärtuste elimineerimine.
Hierarhiline ekstremaalväärtuste elimineerimine.

## Hinnangute testimine

Analüütiline meetod.
Monte Carlo meetod.

```
Program KESKMINE;
        {Monte Carlo meetodi demonstratsioon}

Const n = 25; {yksikmootmiste arv}
      m = 5;  {     sqrt (n)     }

Type andmed = array [1..n] of real;


Function lugem (karbsed : boolean) : real;
   {Karbeste puudumisel Gaussi jaotusega m = 0, s = 1 suurus,
    karbeste esinemisel lisandub 1% toenaosusega lisaviga,
    mis on yhtlaselt jaotatud vahemikus [-100..+100]}
   Var x : real; i : integer;
   Begin
   x := 0;
   for i := 1 to 12 do x := x + random;
   x := x - 6; {Gauss m = 0, s = 1}
   if karbsed and (random < 0.01) then x := x + 200 * random - 100;
   lugem := x;
   End;


Procedure mullsort (var x : andmed);
   Var OK : boolean;
        i : integer;
        a : real;
   Begin
   repeat OK := true;
      for i := 2 to n do if x [i-1] > x [i] then begin
         a := x [i-1]; x [i-1] := x [i]; x [i] := a;
         OK := false;
      end;
   until OK;
   End;
```

```
Procedure statistika (x : andmed;
                    var
                     km {mediaan},
                     k0 {totaalkeskmine},
                     k1 {valistatud ekstremaalvaartustega keskmine},
                     k2 {kvartiilidevaheline keskmine},
                     k5 {5-ste ryhmade tsenseeritud keskmiste
                                        tsenseeritud keskmine}
                        : real);

   Var          i, j, k : integer;
      a, b, s, min, max,
         vs, vmin, vmax :real;
   Begin

   {On-line rezhiimis teostatavad arvutused}

   {Totaalkeskmine}
   s := 0;
   for i := 1 to n do s := s + x [i];
   k0 := s / n;

   {Ekstreemumtsenseeritud keskmine}
   s := x [1]; min := s; max := s;
   for i := 2 to n do begin;
      s := s + x [i];
      if x [i] < min then min := x [i];
      if x [i] > max then max := x [i];
   end;
   k1 := (s - min - max) / (n - 2);

   {5*5 tsenseeritud keskmine}
   vs := 0; vmin := 1E33; vmax := -vmin;
   i := 1;
   for j := 1 to m do begin
      s := x [i]; i := i + 1; min := s; max := s;
      for k := 2 to m do begin;
         a := x [i];
         s := s + a;
         if a < min then min := a;
         if a > max then max := a;
         i := i + 1;
      end;
      b := (s - min - max) / (m - 2); {m mootmise tsenseeritud keskmine}
      vs := vs + b;
      if b < vmin then vmin := b;
      if b > vmax then vmax := b;
   end;
   k5 := (vs - vmin - vmax) / (m - 2);
   {ylesanne 1 : kirjutada viimane algoritm ratsionaalsemalt
                 kogudes kohe (m - 2)^2 arvu summa}
   {ylesanne 2 : kirjutada algoritm m^k mootmise k-astmelise
                 tsenseeritud keskmise arvutamiseks}
```

```
   {On-line rezhiimis mitteteostatavad arvutused}
   mullsort (x);

   {mediaan}
   km := x [13]; {mediaan NB! arvulised indeksid
                            demonstratsiooni lihtsuse huvides}

   {kvartiilidevaheline keskmine}
   s := 0;
   for i := 7 to 19 do s := s + x [i];
   k2 := s / 13;

   End;


Procedure MonteCarlo (kordi : integer; karbsed : boolean; var f : text);
   Var i, j : integer;
       km, k0, k1, k2, k5 : real;
       sm, s0, s1, s2, s5 : real;
       x : andmed;
   Begin
   randomize;
   sm := 0; s0 := 0; s1 := 0; s2 := 0; s5 := 0;
   for i := 1 to kordi do begin
      for j := 1 to 25 do x [j] := lugem (karbsed);
      statistika (x, km, k0, k1, k2, k5);
      sm := sm + km*km;
      s0 := s0 + k0*k0;
      s1 := s1 + k1*k1;
      s2 := s2 + k2*k2;
      s5 := s5 + k5*k5;
   end;
   sm := sm / kordi;
   s0 := s0 / kordi;
   s1 := s1 / kordi;
   s2 := s2 / kordi;
   s5 := s5 / kordi;
   writeln (f, sm:18:4, s0:9:4, s1:9:4, s2:9:4, s5:9:4);
   End;



VAR i : integer;
    f : text;

BEGIN
assign (f, '\b\keskmine.txt'); rewrite (f);
writeln (f, 'Karbesteta  mediaan  totaal  -minmax   7...19     5*5');
for i := 1 to 5 do MonteCarlo (10000, false, f);
writeln (f);
writeln (f, 'Karbestega  mediaan  totaal  -minmax   7...19     5*5');
for i := 1 to 5 do MonteCarlo (10000, true, f);
close (f);
END.
```

| Karbesteta | mediaan | totaal | -minmax | 7...19 | 5*5 |
|---|---|---|---|---|---|
| | 0.0645 | 0.0402 | 0.0414 | 0.0488 | 0.0532 |
| | 0.0630 | 0.0397 | 0.0407 | 0.0479 | 0.0529 |
| | 0.0647 | 0.0404 | 0.0416 | 0.0485 | 0.0538 |
| | 0.0625 | 0.0400 | 0.0410 | 0.0478 | 0.0528 |
| | 0.0634 | 0.0406 | 0.0417 | 0.0488 | 0.0531 |

| Karbestega | mediaan | totaal | -minmax | 7...19 | 5*5 |
|---|---|---|---|---|---|
| | 0.0637 | 1.3355 | 0.0924 | 0.0490 | 0.0534 |
| | 0.0646 | 1.3791 | 0.0868 | 0.0488 | 0.0525 |
| | 0.0638 | 1.3041 | 0.0926 | 0.0487 | 0.0538 |
| | 0.0646 | 1.3950 | 0.0955 | 0.0493 | 0.0541 |
| | 0.0643 | 1.3749 | 0.1020 | 0.0494 | 0.0536 |

*Ülesanne*: Kuidas realiseerida sügavamat hierarhilist tsensuuri, näiteks 5*5*5, 7*7*7*7 jne.?

*Lahendus:*

```
Function Keskmine (grupp, hierarhia : integer) : single;
   Var i : integer; x, s, min, max : single;
   Begin
      s := 0; min := 1E33; max := -min;
      for i := 1 to grupp do begin
         if hierarhia = 1
            then x := lugem
            else x := Keskmine (grupp, hierarhia-1);
         s := s + x;
         if x < min then min := x;
         if x > max then max := x;
      end;
      Keskmine := (s - min - max) / (grupp - 2);
   End;
```