1 praktikum – *StatVillage*'i tutvustus. Freimi moodustamine. Valimi võtmine *StatVillage*'ist *SAS*'i vahenditega

1.1 StatVillage - *http://people.stat.sfu.ca/∼cschwarz/StatVillage/*

StatVillage on hüpoteetiline asula Kanadas. Asula majad on paigutatud ristkülikukujuliste plokkidena, milles on 8 maja. Igale majale vastab ploki ja plokisisene majanumber:

1	2	3
4	**10**	5
6	7	8

Asulast on olemas kolm eri suurusega varianti: *Maximal*- (128 plokki), *Mini*- (60) (Mini varianti kasutame oma praktikumides) ja *Micro Village* (36).

Peresid (=leibkondi) saab uuringu jaoks valida kasutades kaarti, kus saab vajalikud majad märgistada. Andmed valitud perede kohta saab salvestada edasiseks kasutamiseks. Kasutamaks andmeid *SAS*is, on leheküljel ka kättesaadav programmijupp, mille abil saab salvestatud andmed teha *SAS*-andmestikuks. Eestikeelse *SAS*-faili jaoks saate programmi Moodles (fail **statvillage_eesti.pdf**).

Tegemist on reaalsete andmetega Kanada Rahvaloendusest 1991. Iga leibkonna kohta on mõõdetud 48 tunnust (täpsem kirjeldus ja kodeering on toodud *StatVillage*'i kodelehel):

- **Demograafilised tunnused** - pere suurus ja kooslus vanuseklasside ja soo järgi;
- **Sissetulekuid puudutavad tunnused** - investeeringud, riiklikud toetused, jne;
- **Hõivatus tööga**;
- **Eluaset puudutavad andmed** - tüüp, vanus, omanditüüp, väärtus, igakuised elamiskulud jne;
- **Andmed pere kuni kahe pea kohta** (täiskasvanud, kes vastutavad pere heaolu eest) - vanus, sugu, amet, emakeel, haridus, tööalane staatus jne.

Selles asulas paistavad suurema sissetulekuga elanikud olevat koondunud põhjaossa (üles) ning vaesemad lõunasse.

1.2 *StatVillage*'i freimi loomine

Tõenäosusliku valiku teostamiseks on vaja üldkogumi objektide loendit koos identifitseeriva infoga. Siin on objektiks leibkond, mida on võimalik identifitseerida aadressiga, kus ta elab. Kuna kokku on linnas 60 plokki ja igas plokis 8 maja, siis on kokku $N = 480$.

Loend, ka freim - ÜK elementide loend objekte identefitseerivate ja nende ülesleidmist

võimaldavate parameetritega (1:1 vastavusega). *StatVillage*'i puhul peaks see loend sisaldama leibkonna aadresse (ploki number ja maja number plokis).

Praktikumides vaatame 60-blokiga *MiniVillage*'it.

Freimi loomine:

```
/*Freim Mini Villagele*/
data freim;
number=0;
do plokk=1 to 60; /*üle kõigi plokkide*/
    do maja=1 to 8; /*igas plokis 8 maja*/
         number=number+1; /*järjekorratunnuse lisamine*/
         output;
    end;
end;
run;
```

Vaadata freimi *Explorer*'i aknas!

1.3 Valimi võtmine freimist

1.3.1 Mõisteid

Olgu $U = \{1, 2, ..., N\}$ üldkogum (st iga objekt on nummerdatud). Valimi esitamiseks kasutatakse valikuuringutes kolme erinevat varianti.

Hulkvalim, s on ÜK-i osahulk, $s \subseteq U$, kus elementide järjestus pole tähtis, kasutatakse TTA valikute korral. Näiteks: $s = \{2, 5, 3\}$

Järjestusvalim, $js = \{i_1, i_2, ..., i_n\}, i_k \in U$ on valim, kus elemendid on esitatud valimi võtmise järjekorras; võivad esineda ka kordused. Näiteks: $js = \{3, 1, 5, 1\}$. Saab kasutada nii TTA kui ka TGA disainide korral.

Vektorvalim koosneb N elemendist, $k = (k_1, k_2, ..., k_N)$, kus k_i on objekti i valikute arv. Juhul, kui $k_i = 0$, siis objekt i pole valimis. Saab kasutada nii TTA kui ka TGA disainide korral.
Näiteks, $k = (1, 0, 2, 0, ..., 3)$.
Realiseerunud valimimaht sel juhul on järgmine: $n = \sum_{i=1}^{N} k_i$.

Lihtne juhuslik valik: kõikidel ÜK objektidel on võrdne valimisse sattumise tõenäosus (n/N). Tulemuseks on etteantud mahuga (n) valim, mis peegeldab ÜK struktuuri.

Süstemaatiline valik: valimi elemendid valitakse ÜK-st fikseeritud sammu a tagant, kusjuures, 1. objekt valitakse juhuslikult.

Bernoulli valik: Iga ÜK elemendi kaasamistõenäosus on konstantselt π, $0 < \pi \leq 1$. ÜK elemendi kaasamise valimisse määrab juhuslik katse, mille oodatud tulemuse toimumise tõenäosus on π. Katsed on sõltumatud. Valimimaht on juhuslik, kusjuures

$n \sim Bin(N, \pi)$.

1.3.2 Bernoulli valiku teostamise viisid

ÜK maht $N = 480$; keskmine valimi maht $E\boldsymbol{n} = 50$ (Bernoulli valiku puhul on tegemist oodatava valimimahuga). Soovides, et $E\boldsymbol{n} = 50$, saame määrata $\pi = E\boldsymbol{n}/N = 50/480 \approx 0.104$.

Bernoulli valik 1
Kasutame ühtlast jaotust.

```
data valim1;    /*Loodav andmestik teeki WORK*/
set freim;    /*Algandmete andmestik teegist WORK*/
if uniform(0)<50/480 then output;
/*parameeter 0 ütleb, et kasutatakse juhuslikku seemet*/
/*positiivne parameeter annab sama juhusliku jada uuesti kasutamisel*/
run;
```

või alternatiivselt

```
data valim2;
set freim;
k=(uniform(0)<50/480);
run;
```

Missugune programm annab hulkvalimi, missugune vektervalimi?
Kirjutame realiseerunud valimimahud 1. valimis Moodlesse 1. nädala alla. Kas varieeruvad 50 ümbruses?
Leiame valimimahu dispersiooni. Kas saadud realisatsioonid on sellega kooskõlas?

Bernoulli valik 2
Kasutame Bernoulli jaotust.

```
data valim3;
set freim;
if ranbin(0,1,50/480)=1 then output;
run;
```

1.3.3 Süstemaatiline valik

Valimimaht määratakse eeskirjast $N = nm + c$, kus m on valiku samm, $0 \leq c < m$. Kui võtame valimimahtu $n = 53$, siis samm $m = 9$ ja $c = 3$. Nüüd valitakse juhuslik alguspunkt r nii, et $0 < r \leq m$. Realiseerunud valimimaht sõltub alguspunktist r:

$$n_s = \begin{cases} n+1, \text{ kui } r \leq c; \\ n, \text{ kui } r > c. \end{cases}$$

Igaüks valib oma $r \leq 9$. Võtame valimisse iga 9. elemendi alustades r-st.

```
data sys;
```

```
set freim;
    if mod(number-4, 9)=0 then output;
/*4 on juhuslikult valitud alguspunkt*/
run;
```

Võrrelda, kuidas muutub valimimaht sõltuvalt alguspunktist.

1.3.4 Lihtne juhuslik valik TTA

NB! See on fikseeritud mahuga disain.

Järjestusvaliku algoritm, ehk *order sampling*
∗ genereerida tunnus väärtustega mingist pidevast jaotusest
∗ sorteerida ja võtta valimisse 50 vähima väärtusega objekti.

Näiteks:

```
data lihtne;
set freim;
exp=ranexp(0); /*exp-jaotusega juhuslikud arvud*/;
run;

proc sort;
by exp;
run;

data lihtne1;
set lihtne;
if _n_ > 50 then stop;/*Võtab uute valimisse 50 esimest rida*/
run;
```

1.4 Kaheastmeline valik, SQL

1. astmel võtame valimisse 7 plokki kasutades Bernoulli valikut.
2. astmel igast valitud plokist valime 3 maja kasutades lihtsat juhuvalikut TTA.

Mis on lõplik valimimaht? Kas tuleb juhuslik või fikseeritud?

Teostame 1. astme valiku. Meil on juba olemas freim majade jaoks (nimega *freim*, sisaldab $60 * 8 = 480$ objekti). Kuid 1. astme valik toimub plokkide seast ja seepärast loome plokkide freimi eraldi:

```
/*plokkide freimi loomine*/
data pl_freim;
    do plokk=1 to 60;
         output;
    end;
```

```
run;
```

Nüüd Bernoulli valik oodatava mahuga 7 plokki:

```
/*1.  astme valimi moodustamine (E(n)=7)*/
data pl_valim;
set pl_freim;
    if ranbin(0,1,7/60)=1 then output;
run;
```

Milline tuli valimimaht? Valim sisaldab plokkide numbreid, peame lisama neile majade numbreid. Seda on lihtne teostada nn päringu keele SQL abil (*Structured Query Language*), mille üldsüntaks on väga lihtne:

```
CREATE TABEL uus_tabel AS
    SELECT tunnus1,tunnus2
    FROM vana_tabel
    WHERE tingimus;
```

Rohkem võimalusi ja seletusi saab vaatada nt Wikipeediast. Siin õppime jooksvalt. Paneme ka tähele kus on komad ja semikoolon!

```
/*Valimi ühendamine majadega*/
proc sql; /*SASis lülitab SQLi sisse*/
CREATE TABLE valim_aste1 AS /*Loome uue tabeli kausta WORK*/
        SELECT pl_valim.plokk, freim.maja /*Lisame sinna tunnused */
                                        /*olemasolevatest tabelitest*/
        FROM pl_valim,freim /*Kasutatud tabelite nimed*/
        WHERE pl_valim.plokk=freim.plokk; /*Tingimus valimisse sattunud plokkide*/
                                        /* saamiseks*/
quit;
```

Vaadata saadud tabelit!
Teisel sammul teostame majade valikut, kust igast valitud plokis saame 3 maja lihtsa juhuvaliku TTA abil. Selleks kasutame järjestusvaliku algoritmi.

```
/*2.  sammul LJV TTA igast klastrist*/
/*kasutades selleks järjestusvalikut U(0,1)*/
data abi1;
set valim_aste1;
    u=uniform(0);
run;
proc sort;
by plokk u; /*igas plokis järjestame u järgi*/
run;
```

Järgmisel sammul valime igast plokis 3 leibkonda väiksema u_i-ga uute valimisse. Seda saab teostada kahe programmi abil.

```
/*anname igale u-le aastaku*/
proc rank data=abi1
out=abi2;
by plokk;
var u;
run;
```

Vaata tulemust!

```
/*valime igast plokist 3 esimese aastakuga maja*/
proc sql;
create table Valim_aste2 as
select * /*valitakse kõik tunnused tabelist abi2*/
from abi2
where u<=3;
quit;
```

Nüüd peaks valim olema lõplikult võetud. Milline tuli valimimaht?

1.5 Valikuga määratud perede andmete saamine

Avada *StatVillage*'i koduleht. Klikkida lingil, mis viib Mini-linna kaardile. Võtta valimisse oma äranägemise järgi 5-10 peret, märkides need ära kaardil. Seejärel klõpsata nupul *Get the sample units* ja salvestada saadud andmed tekstifailina (**andmed.txt**) endale sobivasse kataloogi.

Andmete toomiseks *SAS*'i kasutada *SAS*-programmi, mille saab leida kas kodulehel (inglisekeelne versioon) või moodles (fail **statvillage_eesti.pdf**, eestikeelne versioon).

1.6 Iseseisev ülesanne

Teostada MiniVillages 2-astmeline valik. Esimesel astmel valida 10 plokki kasutades lihtsa juhuvalikut TTA, ja seejärel igast valitud plokist juhuvalik TTA mahuga 5 maja. Märgistada majad aadressilehel, saada andmeid ja salvestada valimiväärtused SASi failina oma kodukausta edaspidiseks tööks.