

02.06.2005: Asiakastapaaminen.

TARKENNUKSIA:

Päätulostetiedoston alkuun kommenttiriville (#):

- iteraatioiden lkm (total)
- sisäänajokierrosten määrä
- ohennus (kuinka harvoin tulostetaan, joka 10s, joka 100s, eri muuttujille erillinen)
- muuttujien järjestys-selite sarakkeille

Eri tulostetiedostoon halutaan hyväksy/hylätty-frekvenssit per muuttuja vaikkei kaikkia tulostettaisikaan (tai tämän voi kytkeä pois), tästä nähdään onko ehdotusjakauma kunnossa.

-mahdollisuus: Alkuarvot edellisen ajon lopusta seuraavan alkuun ilman burn-iniä, jotta ajoa voitaisiin jatkaa.

-jokin config-tiedosto johon nämä laitetaan ajon lopuksi, ja joka otetaan valittaessa käyttöön seuraavassa ajossa.

-kaikista löytymättömistä annetuista tiedostoista virheilmoitus.

-tulosteen muuttujat erotetaan spacella, riveillä iteraatioita, sarakkeilla muuttujia.

-tulosteen formatointi kentän leveyksillä. Liukuluvuille sopiva.

Laskettujen arvojen hyväksyminen tehdään aina samalla tavalla. Tietty kaava.

Päivitysstrategia:

Miten kandidaatteja generoidaan, yritetäänkö muuttaa yhtä muuttujaa kerrallaan, vai päivitetäänkö useampaa kerralla, eri blokkeja muuttujia halutaan ehkä päivittää yhtä aikaa, kuljetaanko muuttujajonoa järjestyksessä vai arvotaanko.

Joitakin muuttujia on hankala hyväksyä, jos se riippuu kaikista muista, hankala päätyä sopiviin arvoihin. Tähän olisi hyvä että joitakin voisi päivittää useasti.

Default: kaikkia päivitetään järjestyksessä yksi kerrallaan, tai kaikkia randomilla yksi kerrallaan.

Blokitus ei vaikuta ehdotusjakaumiin. Toisaalta blokin muuttujille halutaan usein valita sama ehdotusjakauma. Pitää myös olla mahdollista laittaa muuttujille eri ehdotusjakaumat.

(Emilin) Hatusta jotain blokitus-ehdotusrakennetta:

-Tällainen blokitusjärjestelmä kelpaa.

-Nämä ovat ehdotusjakaumia päivitystä varten (sim. tek)

-Kaikilla muuttujilla on oltava ehdotusjakauma.

Default Poisson(3)

1: {t1, t2} Poisson(3)

2: {t3} ...

3: {t4} ...

4: {t5, t6-t300}

t5 Poisson(2)

t6-t150 Poisson(6)

....

...

(Defaults assumed)

-Jakaumia: Norm, Poisson, Beta, Dirichlet, Gamma, Bin, Geom, Tasajakauma (diskr ja jatkuva), Bernoulli, Multinomial, Binormaali, LogNormaali. Käyttäjän pitää voida antaa oma fortran-funktio jakaumaksi.

Jakaumista tarvitaan:

-tiheysfunktio/pistetnfunktio, osasta tarvitaan myös random-piste jakaumasta.

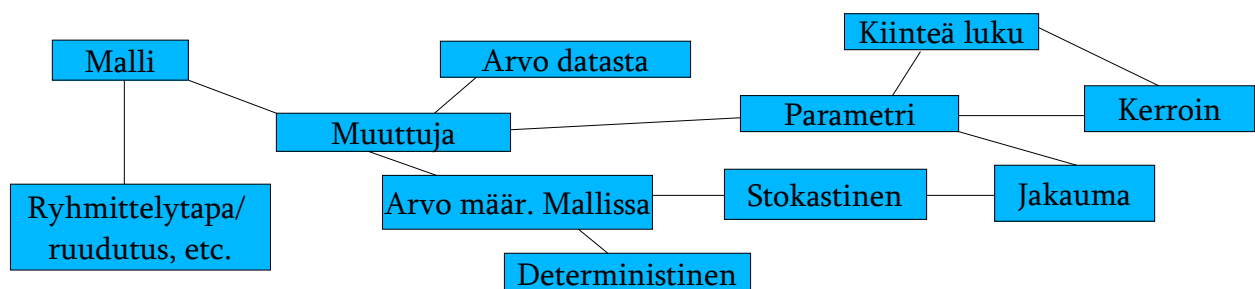
Normaalijakaumalle ainakin, ja tasaiselle.

Oltava monta numeroa kerralla tuottavat funktiot. (Jokin fortran-taulukkosyntaksia tukeva funktio)

Ehdotusjakauma:

Jakauma, josta muuttujille ehdotetaan arvoja. Kaikille muuttujille oltava tällainen. Joko default sim. tekissä tai annettuna erikseen sim. tekissä.

Käsitteellinen malli sovellusalueesta:



-Parametrin määr: arvo ei ole datasta.

Datalla on jakauma, jonka parametrissa päätellään taaksepäin. Viime kerran (30.05.2005) muistiinpanoissa μ_i on tällainen deterministinen muuttuja.

Esimerkki Bassistin kuvauskielestä:

Varatut sanat kursivoitu:

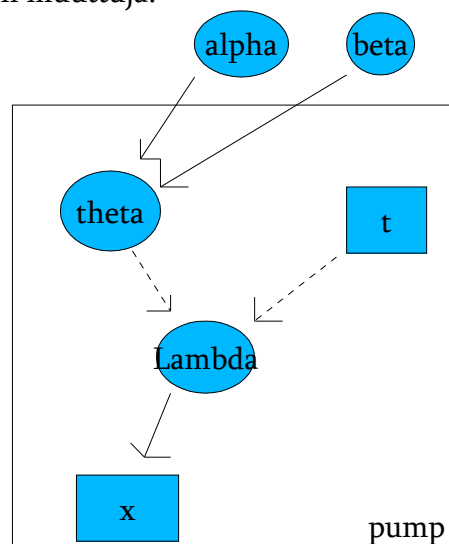
```

record pump{
var theta ~ gamma(alpha, beta)
var lambda = theta * t
var x ~ poisson (lambda)
};

```

alpha ~ exp(1, 0);

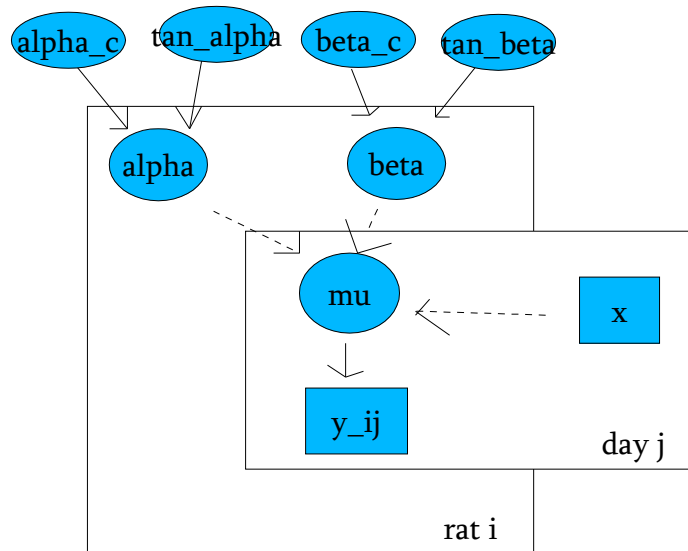
beta ~ gamma(0, 1);



t:tä ei ole annettu, se kuulemma aina tulee datasta. Tähän meille siis t from file jotakin sarakkeella jotakin...

Toinen esimerkki:

Jos tätä ei tule datasta, niin skuup.



```
Record day{  
var id;  
var x;  
};
```

```
record rat{  
var id;  
var alpha ~ norm(alpha_c, tan_alpha)  
var beta norm(beta_c, tan_alpha)  
};
```

```
record weight{  
ref rat as r;  
ref day;  
var mu = r.alpha + r.beta*(day.x - x_bar)  
var y ~ norm(mu, tan_c)  
};
```

Bassistissa tähän liittyvä data on muotoa:

```
rat r.0  
rat r.1  
day d.0: x= 8  
day d.1: x= 15
```

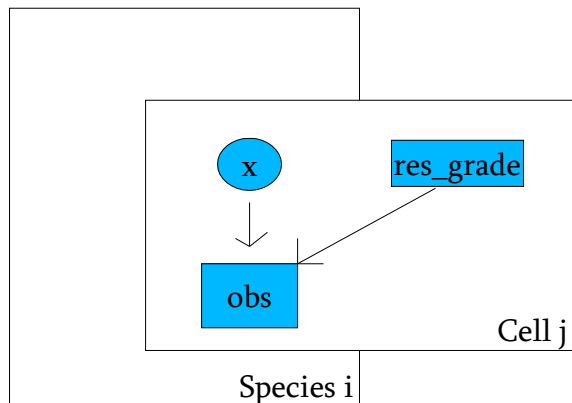
weight[r.0, d.0]: y = 151

Meidän malleissa toinen record on jokin spatiaalinen yksikkö, tyyliin:

Tässä spatiaalinen riippuvuus ei ole mukana.

Datassa on taulukko lajeista ja ruuduista

	cell1	Cell2	Cell3
Species1	2	0	1
Species2	3	2	1



Nelikulmiot kuvauskielessä recorderja (cell, species), näihin liittyy tiedosto. Recordin datasta haettava tavara voidaan yks. käs. määritellä tiedostonimen ja sarakenumeron perusteella.

```
record cell{
var id;
neighbours: vierusmatriisi.txt (täältä saadaan joukko viittauksia tietyn sarakkeen id solmun naapureiksi)
var nrneigh;
}
```

On tarjottava tietorakenne recordille, josta voidaan viitata kaikkiin muuttujien arvoihin ja ruudun naapureihin etc. kuin ohjelmointikielessä.

-Sovittu: globaaleja muuttujia ei tarvitse antaa datasta.