VTT TIETOTEKNIIKKA

TUTKIMUSRAPORTTI TTE1-2000-3

TEMPAT-projekti

Tapaustutkimus

Päällystyskoneen ylösajojen optimointi

Versio 1.1-3

Helmikuu 2000

Mikko Hiirsalmi
Jukka Kiviniemi
Jorma Kuha
Esa Rinta-Runsala
Antoni Wolski

Muutoshistoria

Versio	Pvm	Laatija(t)	Tarkastaja	Kuvaus
0.1-1	20.12.1999	Projektiryhmä		Alustava luonnos
0.2-1	22.12.1999	Projektiryhmä	anw	Toinen luonnos
1.0-2	26.1.1999	Projektiryhmä	anw	Ensimmäinen (JR) versio
1.1-1	11.2.2000	Projektiryhmä	anw	Projektiversio
1.1-2	22.2.2000	Projektiryhmä	anw	Jakeluversio
1.1-3	24.2.2000	projektiryhmä	anw	Ensimmäinen korjaus

Yhteystiedot

A. Wolski
VTT Tietotekniikka
PL 1201, FIN-02044 VTT
Katuosoite: Tekniikantie 4 B, Espoo
Puh. (09) 4561, fax (09) 456 6027
Sähköposti: antoni.wolski@vtt.fi

Raportti on saatavilla osoitteessa:
`http://www.vtt.fi/tte/projects/tempat/tapaustutkimus.html`

Viimeisin muutos 24.2. 2000
Tiedosto: T:\TEMPAT\Reports\tapaustutk-v11\raportti-v113c.doc

---

Tiivistelmä

Raportissa esitellään eräällä paperitehtaalla tehty tapaustutkimus, jonka tavoitteena on paperikoneen käyttökatkonjälkeisen ylösajon optimointi. Ylösajon aikana syntyy paperihylkyä, jonka määrä vaihtelee riippuen ylösajon vaikeasti ennakoitavasta kestosta. Tutkimuksen tavoitteena on tunnistaa ja luokitella eri tyyppisiä ylösajoja, sekä esittää tapa tuottaa optimaaliseen ylösajoon tarvittavat lähtöparametrit. Raportissa tutkitaan eri menetelmien soveltamismahdollisuuksia ja menetelmiä sovelletaan prosessista muutaman kuukauden ajan kerättyyn aineistoon. Tutkittuja menetelmiä ovat tilastollinen analyysi, hierarkkinen ryvästys, SOM-tekniikka, luokittelu päätöspuilla, diskriminanttianalyysi, lineaarinen mallintaminen, neuraaliverkot, Bayes-verkot ja muistiperustainen päättely (MBR). Parhaimmiksi menetelmiksi todetaan lineaarimalli ja muistiperustainen päättely. Parhaita menetelmiä testataan synteettisellä aineistolla ja lisäksi raportissa esitetään, miten menetelmiä voidaan soveltaa käytännön teollisuusympäristössä.

Abstract

A case study performed at a paper mill is presented, wherein the objective is to optimize after-break runups of a paper coater machine. During a runup, a certain amount of waste paper is produced, depending on the runup time that is difficult to predict. The objectives of the study are: to identify and classify different types of runups, and to propose the configuration of startup parameters that would lead to an optimal runup. Applicability of various methods is studied, and the methods are applied to real process data collected over a time period of few months. The methods covered are: statistical analysis, hierarchical clustering, SOM (self-organizing maps), decision tree classification, discriminant analysis, linear modelling, neural networks, Bayesian networks and memory-based reasoning. The most promising methods turned out to be linear modelling and memory-based reasoning. The successful methods are tested using a synthetic test data. Ways to implement process optimising tools in the mill environment are also proposed.

Sisällysluettelo

1 Johdanto

Tässä raportissa selostetaan vuonna 1999 suoritetun TEMPAT-projektin prosessitiedon analyysin tapaustutkimus. Tutkimuksen kohteena oli erään paperitehtaan päällystyskone, jonka ylösajoja oli tarkoitus optimoida niin, että ylösajojen yhteydessä syntyvän paperihylyn määrä olisi mahdollisimman vähäinen. On arvioitu, että optimoinnin tuomat säästöt olisivat noin 1 milj. mk vuodessa, jos jokaisen ylösajon yhteydessä hylkymäärä pienenisi keskimäärin yhdellä kilometrilla.

Tutkimukselle oli asetettu seuraavat kolmen tason tavoitteet:

1) tunnistaa ja luokitella eri tyyppisiä ylösajoja
2) löytää ylösajon kannalta tärkeitä tunnusmerkkejä ja formalisoida niitä (mallintaminen)
3) esittää tapa tuottaa ajotilanteessa optimaaliseen ylösajoon tarvittavat lähtöparametrit (päättely)

Tutkimukseen kuului useita vaiheita, jotka kuvataan omissa luvuissaan. Seuraavassa luvussa kuvataan päällystyskoneen prosessi ja siihen liittyvä tietoaineisto. Prosessitiedon kerääminen, esikäsittely, ylösajojen poiminta ja piirteiden poiminta kuvataan luvussa 3. Jatkoluvuissa kuvataan prosessiaineistoista poimittujen ylösajojen erityyppisiä analyysejä. Luku 4 on omistettu perinteiselle tilastolliselle analyysille. Luvussa 5 esitetään erilaisten ryvästystekniikoiden tuloksia. Luvussa 6 selostetaan automaattisen luokittelun tuloksia. Rinnakkainen kolmen mallinnustekniikan kokeilu on kuvattu luvussa 7. Tuloksia arvioidaan luvussa 8 ja luvussa 9 esitetään jatkotoimenpiteitä ja operatiivisen ratkaisun toteutustapa.

2 Tapaustutkimuksen lähtökohdat

Tapaustutkimus perustuu tietoihin, joita on kerätty prosessijärjestelmästä usean kuukauden ajan. Prosessin periaate ja ylösajon tapahtumat on kuvattu seuraavassa kappalessa. Sen jälkeen kuvataan prosessista kerätty tietoaineisto.

2.1 Päällystyskoneen toimintaperiaate

Päällystyskone on paperitehtaan itsenäinen prosessi, joka yhdistyy tuotannon muihin prosesseihin (edestä varsinainen paperikone ja takaa superkalanteri ja leikkuri) vain paperirullapuskurivarastojen kautta.

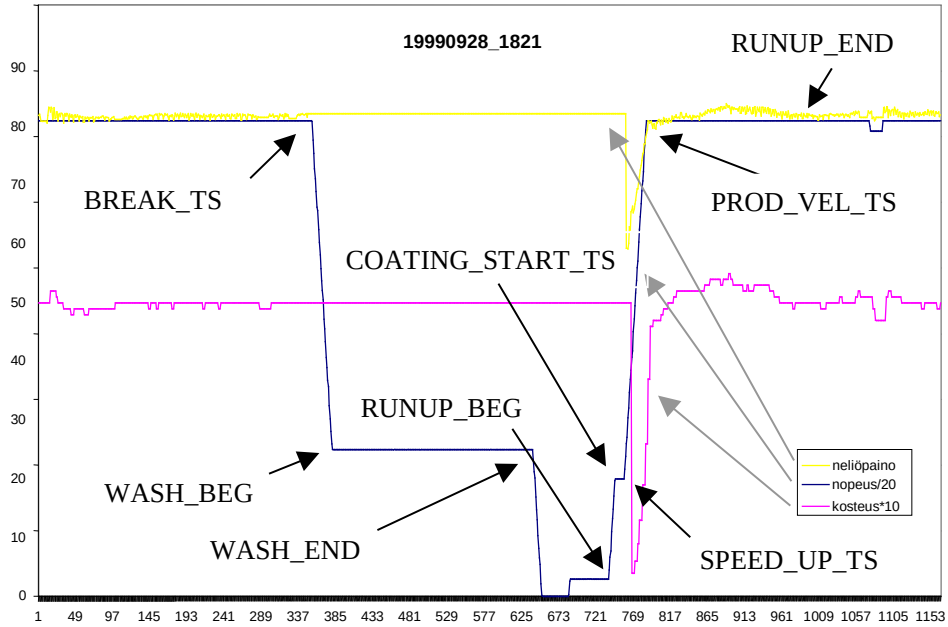
Päällystyskone ottaa raaka-aineeksi paperikoneella tuotettuja pohjapaperirullia ja tuottaa kaksipuoleisesti päällystettyjä paperirullia. Koneen toiminta perustuu neljään peräkkäiseen päällystysasemaan, joissa pinnoitetta levitetään ja kuivataan. Vaikka eri asemissa

käytetään erilaisia päällystysmassan levitystapoja, kuivatusmenetelmät ovat kaikissa asemissa samanlaisia: jokaisessa asemassa on kaksi infrapunakuivaajaa (nk. infrakuivaimet) kaksi puhalluskuivaajaa (nk. leijukuivaimet) ja lisäksi höyryllä lämmitettyjä sylintereitä (ylä- ja alasylinterit). Koneen kuivatusjärjestelmän säätö on ratkaiseva sekä jatkuvan ajon laadun ylläpitämisessä että ylösajon nopeuttamisessa.

Periaatteessa pällystyskone on rakennettu jatkuvaa toimintaa varten: uudet rullat pystytään ottamaan vastaan "lennosta" ja samalla tavalla vaihtuvat valmiin tuotteen rullat. Todellisuuteen kuuluvat kuitenkin odottamattomat tuotantokatkot ja niitä seuraavat ylösajot. Odottamattomat tuotantokatkot johtuvat prosessissa ilmenneistä virheistä, jotka vuorostaan johtuvat päällystysmassan joutumisesta koneen arkoihin paikkoihin. Päällystyskoneella katkoja tapahtuu useammin kuin paperikoneella, koska päällystyskoneen prosessi on "likaisempi" kuin paperikoneen prosessi. Siksi päällystyskoneen katkot ovat väistämättömiä muutaman tunnin välein. päällystyskoneella tapahtuu vuodessa noin 1000 katkoa, mikä tarkoittaa katkoa keskimäärin 8 tunnin välein.

2.2 Katkon anatomia

Kuva 2.1 esittää tyypillisen katkon ja sitä seuraavan ylösajon. Aikasarjakuvaajassa nähdään koneen nopeus, lopputuotteen neliöpaino (g/m^2) ja kosteus (%), jotka ovat tärkeimpiä laadun (paperilajin) parametreja.



Kuva 2.1. Tyypillisen katkon kuvaus.

Kun tuotantonopeudella 1600 m/min toimivassa koneessa havaitaan paperiradan katkeaminen (katkohetki, kohta BREAK_TS kuvassa 2.1), sitä seuraavat alla esitetyt prosessin vaiheet ja operaattorin toimenpiteet.

1) Automaatiojärjestelmä katkaisee päällystysasemien toiminnan (asemat "avataan") ja pudottaa koneen nopeuden nk. pesunopeudelle, joka on noin 500 m/min (kohta WASH_BEG)

2) Kone pestään koneellisesti, kunnes se on valmis ylösajoon. Tämä vaihe kestää noin tunnin. Sen jälkeen kone pysäytetään hetkeksi (kohta WASH_END).
3) Paperiradan pujotus aloitetaan pujotusnopeudella, joka on 60 m/min. Pujotusvaihe kestää noin 10 min.
4) Kun paperirata on pujotettu kiinnirullaukseen asti, kone on valmis ylösajoon. Tilanne on silloin seuraava:
   - automaatiojärjestelmä on asettanut ylösajon lähtöarvot (säätöarvot, jotka toimivat käynnistyshetkellä) paperilajin reseptin mukaan;
   - operaattori voi vaihtaa lähtöarvot, jos siihen on perustetta.
5) Operaattori käynnistää ylösajon (lähtöhetki, kohta RUNUP_BEG), joka sen jälkeen jatkuu automaattisella ohjauksella. Automaatiojärjestelmä kiihdyttää koneen alkunopeuteen (400 m/min) (kohta COATING_START_TS).
6) Kun kone käy alkunopeudella (noin 2 minuutin ajan), automaatiojärjestelmä käynnistää päällystysasemat (asemat "suljetaan").
7) Kun asemat on käynnistetty, automaatiojärjestelmä aloittaa loppukiihdytyksen (kohta SPEED_UP_TS), joka päättyy, kun kone on saavuttanut tuotantonopeuden (PROD_VEL_TS). Ylösajon käynnistämisestä (lähtöhetkestä) on kulunut 8 min.
8) Vaikka kone käy jo tuotantonopeudella, se tuottaa laatumäärittelyn mukaista paperia vasta, kun neliöpaino- ja kosteusarvot asettuvat "laatuputkeen" eli sallitun arvoalueen sisälle. Silloin ylösajo päättyy (laatuhetki, kohta RUNUP_END). Onnistuneessa ylösajossa laatuhetki saavutetaan 9-10 min kuluessa lähtöhetkestä.

Suurimmat hylkymäärät syntyvät hetkien PROD_VEL_TS ja RUNUP_END välillä, koska silloin kone käy huippunopeudella. Jos siihen kuluu vain pari minuuttia, hylkymäärä on alle 8 km ja tulos on hyvä. Tällä hetkelllä tyypillinen tulos on 8-10 km ja se on kohtuullinen. Joskus kuitenkin prosessi oskilloi pitkään. Tällaiset tapaukset ovat ei-toivottuja, koska hylkymäärät voivat noustaa kymmeniin kilometreihin. Jos hylkymäärä on yli 10 km, ylösajon tulos tulkitaan huonoksi.

Tämän tutkimuksen lopullisena tavoitteena on löytää tapa tai tapoja vaikuttaa ylösajoihin lähtöarvojen asetuksilla niin, että hylkymäärät keskimäärin pienenisivät.

2.3 Tutkimuksen lähtötietoja

Päällystyskoneen prosessista on kerätty tietoja usean kuukauden ajan. Niihin kuuluvat 10 s aika-askeleella poimitut muuttujien arvot, joita ovat sekä prosessin ulostulon mittausarvot (65 kpl) ja ohjausjärjestelmän asetusarvot (34 kpl). Sen lisäksi on saatu tehtaan järjestelmään talletettuja katkotietoja ja rullakohtaisia hylkytietoja. Aineistoa täydentävät eri paperilajien laatumäärittelyt.

Kaikki tehtaalta saadut tiedot on esikäsitelty ja syötetty analyysitietokantaan (siitä lisää seuraavassa luvussa). Kaikkien muuttujien lista ja niiden tietokantamäärittelyt on esitetty Liitteessä A. Tärkeimmät muuttujat on esitetty seuraavassa.

2.3.1 Mittaustietoja

Kaikki jatkuvasti mitatut muuttujat ovat mittaustietueiden attribuutteja ja ne on sijoitettu RAW_RECDS –tietokantatauluun. Tärkeimmät tutkimuksessa käytetyt tiedot ovat (oikealla puolella on tietokannan sarakenimi):

Koneen nopeus		VELOCITY

Lopputuotteen laatuparametrit:

Kosteus	4.asema	S4_MOIST
Neliöpaino	4.asema	S4_BASIS_W

Päällystysasemien säätöjen mittausmuuttujat:

Infrakuivain ’m’ teho	asema ’n’	Sn_IRDRm_P
Leijukuivain ’m teho	asema ’n’	Sn_AIDRm_P
Leijukuivain 'm lämpötila	asema 'n'	Sn_AIDRm_T
Leijukuivain 'm kuivatusilman nopeus	asema 'n'	Sn_AIDRm_F
Paine yläsäädin	asema 'n'	Sn_UPP_PRE
Paine alasäädin	asema 'n'	Sn_LOW_PRE

(jossa päällystysaseman numero n = 1, 2, ... 4; kuivaimen numero m = 1, 2, ... 8)

Massan levityksen mittausmuuttujat (23.11.1999 lähtien):

Kuormitusletkun paine (asemat 1 ja 2)	asema ’n’	Sn_TLOAD_PRE
Teräkulma (asemat 3 ja 4)	asema 'n'	Sn_TIP_ANG

Päällystysasemien säätöjen asetusmuuttujat (23.11.1999 lähtien)

Infrakuivain ’m’ teho	asema ’n’	SnS_IRDRm_P
Leijukuivain 'm' lämpötila	asema 'n'	SnS_AIDRm_T
Leijukuivain 'm kuivatusilman nopeus	asema 'n'	SnS_AIDRm_F
Paine yläsäädin	asema 'n'	SnS_UPP_PRE
Paine alasäädin	asema 'n'	SnS_LOW_PRE

Massan levityksen asetusmuuttujat (23.11.1999 lähtien):

Kuormitusletkun paine (asemat 1 ja 2)	asema ’n’	SnS_TLOAD_PRE

2.3.2 Ylösajojen ominaisuudet

Luvussa 3 kuvataan tarkemmin, miten ylösajot on tunnistettu ja miten niiden ominaisuudet on poimittu havaintomassasta. Numeerisesti tunnistetut ylösajot on kuvattu taulussa RUNUPS2. Jokaisen ylöajon yhteyteen on poimittu sitä kuvaavia ajanhetkiä (Kuva 2.1) ja mm. seuraavat ominaisuudet:

Lopullinen tuotantonopeus	PROD_VEL
Tuotetun rullan nro	REEL_ID_NEXT
Laskettu hylkymäärä	WASTE_APP
RUNUP_BEG - BREAK_TS (s)	TOTAL_INTVL
Laskettu lopullinen neliöpaino	BAS_W_NEW
Edellisen ajon neliöpaino	BAS_W_OLD
BAS_W_NEW - BAS_W_OLD	DELTA_BAS_W

2.4 Analyysin aineistot

Prosessista kerättyjen tietojen kokoelma muuttui kerran tutkimuksen aikana. Tämän johdosta tutkimuksessa on käytetty erilaisia aineistoja seuraavan kuvauksen mukaisesti.

2.4.1 Aineisto A

Aineisto A liittyy ensimmäisen tiedonkeruujaksoon 21.9. – 22.11.1999. Silloin kappaleessa 2.3. mainittuja asetusarvoja ei ollut kerätty, eikä myöskään massalevityksen tietoja. Aineiston A analyysissa on käytetty seuraavaa lähtöarvon määräytymissääntöä:

> ***Aineiston A lähtöarvon määräytymissääntö***
>
> Säädön ylösajossa käytetty lähtöarvo on sama kuin sitä edeltävällä katkohetkellä (BREAK_TS) oleva mittausmuuttujan arvo.

Tämän oletuksen perusteella, lähtöarvoiksi on poimittu seuraavien muuttujien arvot:

- kaikkien infrakuivainten tehot (8 kpl)
- kaikkien leijukuivainten lämpötilat (8 kpl)
- kaikkien leijukuivainten kuivatusilman nopeudet (8 kpl)
- ylä- ja alasylinterien hyörypaineet (8 kpl)

Kaikki nämä muuttujat on lueteltu oheisessa taulukossa (Taulukko 2.1)

Asema 1	Asema 2	Asema 3	Asema 4
S1_IRDR1_P	S2_IRDR3_P	S3_IRDR5_P	S4_IRDR7_P
S1_IRDR2_P	S2_IRDR4_P	S3_IRDR6_P	S4_IRDR8_P
S1_AIDR1_T	S2_AIDR3_T	S3_AIDR5_T	S4_AIDR7_T
S1_AIDR1_F	S2_AIDR3_F	S3_AIDR5_F	S4_AIDR7_F
S1_AIDR2_T	S2_AIDR4_T	S3_AIDR6_T	S4_AIDR8T
S1_AIDR2_F	S2_AIDR4_F	S3_AIDR6_F	S4_AIDR8_F
S2_UPP_PRE	S2_UPP_PRE	S3_UPP_PRE	S4_UPP_PRE
S2_LOW_PRE	S2_LOW_PRE	S3_LOW_PRE	S4_LOW_PRE

Taulukko 2.1. Aineiston A lähtöarvoiksi tulkitut muuttujat.

Leijukuivainten tehomuuttujia ei ole huomioitu, koska ne edustavat laskennallisia arvoja (lasketaan lämpötilan ja kuivatusilman nopeuden perusteella). Ottaen huomioon, että

päällystysasemaa kohti on kaksi infrakuivainta ja kaksi leijukuivainta, lähtöarvoja oli 32 jokaista ylösajoa kohti.

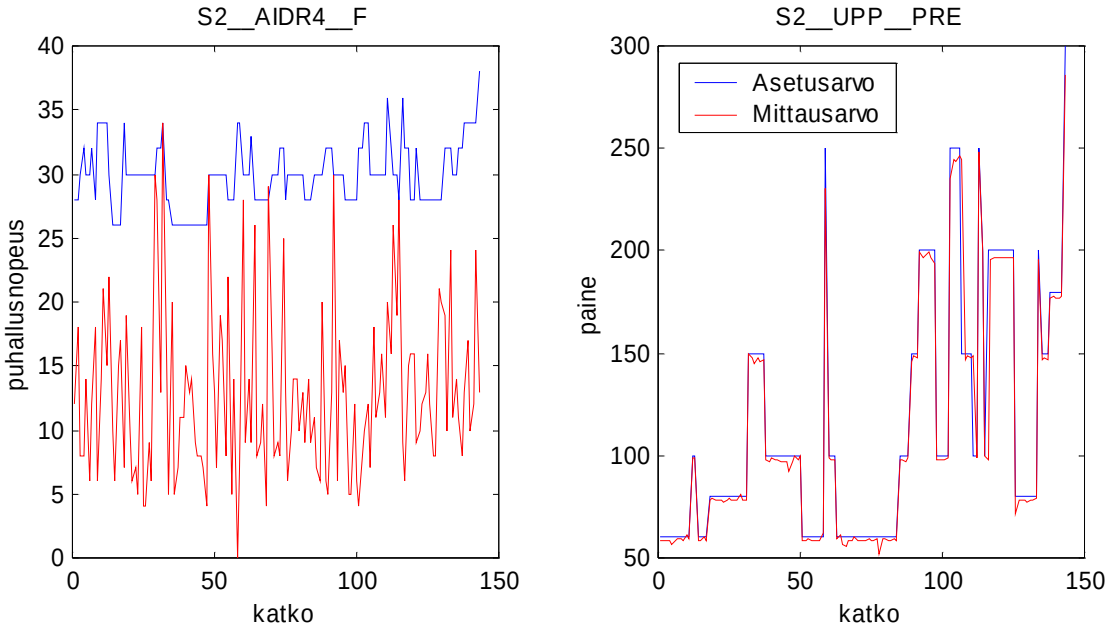
2.4.2 Aineisto B

Aineiston A käsittelyn aikana ilmeni, ettei kaikkien lähtöarvojen kohdalla edellisessa kappaleessa mainittu määräytymissääntö pitänyt paikkaansa. On päätetty laajentaa kerättyjen muuttujien joukkoa säätöjen asetusmuuttujilla ja johtaa lähtöarvot niistä.

Aineistoon B kuuluvat kaikki kappaleessa 2.3.1 mainitut muuttujat. Aineiston B kerääminen alkoi 23.11.1999. Aineiston B kohdalla käytetään seuraavaa lähtöarvon määräytymissääntöä:

> ***Aineiston B lähtöarvon määräytymissääntö***
>
> Muuttujan ylösajossa käytetty lähtöarvo on sama kuin loppukiihdytyksen alussa (SPEED_UP_TS-hetkellä) oleva asetusmuuttujan arvo.

Kuvasta (Kuva 2.2) nähdään, miten joidenkin muuttujien aineiston A säännön mukaan poimitut lähtöarvot (punainen, "Mittausarvo") eroavat selvästi aineiston B säännön mukaan poimittujen lähtöarvoista (sininen, "Asetusarvo"). Tällainen tapaus on esitetty vasemmanpuolisessa kuvaajassa. Oikeanpuoleisessa kuvaajassa näkyy tapaus, jossa määräytymissäännön muuttaminen ei aiheuttanut mertkittävää lähtöarvon muutosta.



Kuva 2.2. Kahden muuttujan mittausarvojen ja asetusarvojen käyttäytyminen.

Aineistossa B oli 34 asetusmuuttujaa. Tästä joukosta on sitten poistettu epäolennaiset muuttujat, eli sellaiset, joiden lähtöarvo oli joko vakio koko aineistossa tai melkein vakio. Näiden muuttujien arvot on vakioitettu jatkoanalyysia varten seuraavasti (Taulukko 2.1):

Asema 1	Asema 2	Asema 3	Asema 4
S1S_IRDR1_P=100 S1S_IRDR2_P=60 S1S_UPP_PRE=0 S1S_LOW_PRE=20 S1S_TLOAD_PRE =100	S2S_IRDR3_P=100 S2S_IRDR4_P=60 S2S_TLOAD_PRE =100	S3S_IRDR5_P=100	S4S_IRDR7_P=100

Taulukko 2.2. Ainestosta poistetut muuttujat ja niiden oletetut arvot.

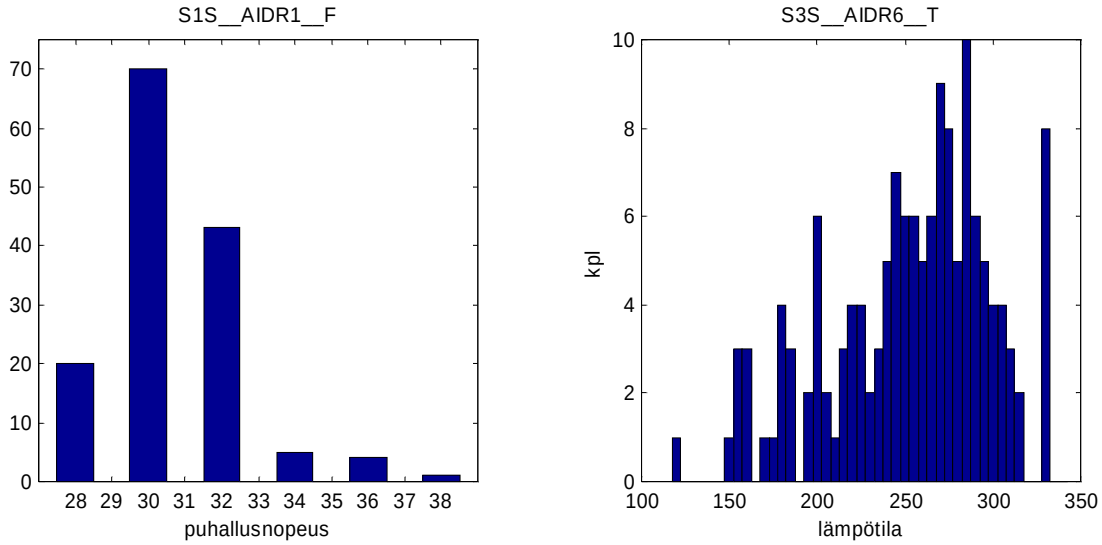
Tämän jälkeen joukkoon on jäänyt seuraavaa 24 asetusmuuttujaa (Taulukko 2.3):

Asema 1	Asema 2	Asema 3	Asema 4
S1S_AIDR1_T	S2S_AIDR4_T	S3S_IRDR6_P	S4S_IRDR8_P
S1S_AIDR1_F	S2S_AIDR3_T	S3S_AIDR5_T	S4S_AIDR7_T
S1S_AIDR2_T	S2S_AIDR3_F	S3S_AIDR5_F	S4S_AIDR7_F
S1S_AIDR2_F	S2S_AIDR4_F	S3S_AIDR6_T	S4S_AIDR8_T
	S2S_UPP_PRE	S3S_AIDR6_F	S4S_AIDR8_F
	S2S_LOW_PRE	S3S_UPP_PRE	S4S_UPP_PRE
		S3S_LOW_PRE	S4S_LOW_PRE

Taulukko 2.3. Aineiston B lähtöarvoiksi tulkitut muuttujat.

Yllä esitetystä joukosta, säädön lähtöarvoista 13 on muuttunut (suhteessa aineistoon A) ja 11 on pysynyt suunnilleen samana.

Lisäksi nämä 24 asetusmuuttujaa voidaan jakaa karkeasti kahteen ryhmään, diskreetti-jakaumaisiin ja jatkuvajakaumaisiin, joista esimerkki Kuva 2.3.



Kuva 2.3. Esimerkki diskreettijakaumaisesta (S1S_AIDR1_F) ja jatkuvajakaumaisesta (S3S_AIDR6_T) muuttujasta.

Taulukko 2.4 esittää tähän perustuvan muuttujien karkean jaottelun. Muuttujien jakaumat vaikuttavat joidenkin analyysien tuloksiin ja luotettavuuteen, joten tuloksia tulkittaessa ne on otettava huomioon.

Diskreettijakaumaiset muuttujat	Jatkuvajakaumaiset muuttujat
Mittausmuuttujat: S1S_AIDR1_T S2S_AIDR3_T S3S_AIDR5_T S4S_AIDR7_T S1S_AIDR1_F S1S_AIDR2_F S2S_AIDR3_F S2S_AIDR4_F S3S_AIDR5_F S3S_AIDR6_F S4S_AIDR7_F S4S_AIDR8_F S2S_UPP_PRE S2S_LOW_PRE S3S_UPP_PRE S3S_LOW_PRE S4S_UPP_PRE S4S_LOW_PRE S3S_IRDR6_P S4S_IRDR8_P	Mittausmuuttujat: S1S_AIDR2_T S2S_AIDR4_T S3S_AIDR6_T S4S_AIDR8_T Ylösajon ominaisuudet: TOTAL_INTVL BAS_W_NEW BAS_W_OLD WASTE_APP

Taulukko 2.4. Muuttujien ryhmittely jakauman perusteella.

3 Tietojen esikäsittely

Esikäsittelyssä on kaksi vaihetta:

- Aikasarjatiedon saaminen yhtenäiseen muotoon Oracle-tietokantaan tallettamista varten (kullekin aikaleimalle saadaan arvot kaikista muuttujista).
- Tarkempien katkotietojen poimiminen aikasarjatiedoista.

3.1 Tietoformaatin yhtenäistäminen

Aikaleimattua tietoa on saatu kolmessa eri muodossa:

- Aluksi (elo- ja syyskuun lopussa sekä lokakuun alussa) tietoa tuli Excel-taulukoina ja satunnaisesti – yhdellä taulukon rivillä kaikki yhden aikaleiman tiedot
- Sitten lokakuun puolestavälista lähes marraskuun loppuun tietoa tuli nk. lajittelemattomassa muodossa säännöllisesti, eli aikaleimattu tieto oli jaettu kuudelle peräkkäiselle riville kutakin aikaleimaa kohden. Aikaleimat saattavat kuitenkin "elää" ja rivejä saattaa puuttua, jolloin on tunnistettava stabiili jakso ja tulostettava ainoastaan se.
- Marraskuun lopusta alken lajittelemattomassa tiedossa on ollut mukana myös uusia, aiemmin poisjätettyjä muuttujia, kunkin aikaleiman tiedot vieden nyt 10 riviä.

Tietokantaan tallettamista varten tieto piti saada yhtenäiseen muotoon. Täksi muodoksi valittiin ascii-tiedosto. Excel-tiedostot konvertoitiin asciiksi kirjoittamalla Excel-makro joka konvertoi kaikki tiedostot. Lajittelemattoman tiedon käsittelyyn kirjoitettiin C++-ohjelma, joka

- tutkii, onko kysessä 6- vai 10-riviformaatin tiedosto, milloin aikaleimat ovat luotettavia, ja missä luotettavien aikaleimojen alueella alkaa ensimmäinen tulostettava rivi (tämä on pääteltävä älykkäästi, datassa ei ole muuttujien tunnisteita helpottamassa).
- poistaa turhaa dataa (ylimääräisiä aikaleimoja ja koskaan muuttumattomia "muuttujia" jotka ovat aina mukana datassa).
- muuttaa erotinmerkiksi sarkaimen pilkun sijaan.

Tämä C++-ohjelma konvertoi ainoastaan yhden tiedoston (kysyen konsolilta syöttö- ja tulostiedostojen nimet). Koska tehtaalta data saadaan lukuisissa tiedostoissa (zip-tiedoston sisälle talletettuina), talletetaan konvertoitavat tiedostot ensin kaikki samaan hakemistoon, ja kutsutaan .bat-komentotiedostoa joka ajaa konvertointiohjelman kaikille hakemistossa oleville tiedostoille. Tämän jälkeen tieto on valmis vietäväksi tietokantaan.

3.2 Katkotietojen poiminta

Aikasarjatiedosta halutaan tunnistaa

- katkot: aikaleimat, jotka kertovat katkon vaiheet koneen nopeuden muutoksien määritteleminä. On tiedossa "tyypillinen katko", mutta käytännössä voi esiintyä lukemattomia muunnelmia tästä

- aikaleima, joka kertoo milloin katkon jälkeen 4. aseman neliöpaino ja kosteus ovat "stabiloituneet"
- syntyneen hylyn määrä kilometreissä
- "stabiloituneet" neliöpainot ennen- ja jälkeen katkon.

Kaikki päättelyn syöttötieto (koneen nopeus, kosteus ja neliöpaino) on aikasarjaluonteista, ja koska on kyse mittaustiedosta, voidaan olettaa siinä olevan satunnaisia virheitä.

3.2.1 Katkon tunnistaminen

Tunnistamiseen voitaisiin käyttää esimerkiksi hahmontunnistusta, sumeita sääntöjä, aallokkeita, ohjatusti opetettuja neuroverkkoja yms. Tässä työssä kuitenkin päätettiin, että

1. Määritellään aakkosto, joka kuvaa katkon kannalta oleellista koneen nopeuden muutoksia.
2. Muunnetaan koneen nopeutta kuvaava aikasarja tuon aakkoston merkkijonoksi
3. Määritellään katkoa kuvaava merkkijono säännöllisenä lausekkeena.
4. Kirjoitetaan tuon säännöllisen lausekkeen tunnistava selaaja [ASU86], ja poimitaan katkot aikasarjasta.

Määritelty aakkosto on S ("stable"), B ("bottom"), I ("intermediate"), U ("up") ja D ("down"). Aikasarja muutetaan merkkijonoksi seuraavasti:

- Jos jonkun allaolevan säännön soveltaminen aikaansaisi kaksi perättäistä samaa aakkosta merkkijonoon, sääntöä ei sovelleta (eikä enää muitakaan sääntöjä kokeilla, vaan edetään aikasarjan tutkimisessa).
- Jos nopeus muuttuu edellistä edeltävästä aikaleimasta korkeintaan 1%, ja nopeus on yli 1000 metriä minuutissa, tulostetaan S
- muuten, jos nopeus on alle 66, tulostetaan B
- muuten, jos nopeus ei muutu enempää kuin 1% edellistä edeltävästä aikaleiman arvosta, tulostetaan I
- muuten, jos nopeuden muutos oli positiivinen, tulostetaan U
- muuten tulostetaan D.

Tyypillistä katkoa kuvaa merkkijono SDIBUIUS. Jos merkinnällä [!X]* tarkoitamme "nollaa tai useampaa mitä tahansa aakkoston alkiota paitsi X:ää", niin silloin haluamme tunnistaa merkkijonot SD[!S]*I[!S]*BUIU[!SB]*S[1]. Näiden merkkijonojen tunnistukseen kirjoitettiin yksinkertainen selaaja. Katkon löydyttyä tulostetaan katkon määrittelevien aakkosten aikaleimat.

Näin saadaan aikasarjoista poimittua kaikki muu katkoista tarvittava tieto paitsi "neljännen aseman oskilloinin stabiloituminen".

---

[1] Varsinaisesti säännöllisissä lausekkeissa merkintä * tarkoittaa "nollaa tai useampaa mitä tahansa merkkiä". Tässä dokumentissa se kuitenkin poikkeuksellisesti tarkoittaa "nollaa tai useampaa mitä tahansa merkkiä, paitsi tämän tähden jälkeen tulevaa akkosten sekvenssiä". Tämän dokumentin *:n voi muuntaa tavanomaiseksi esittelemällä uuden aakkosen jokaista tällaista osa-sekvenssiä kohti. Esimerkiksi lauseke [!S]*BUIU korjataan esittelemällä aakkonen X, joka generoidaan syötettä lukiessa silloin, kun syötteessä ovat merkit BUIU peräkkäin. Uudeksi lausekkeeksi tulee [!SX]*X.

3.2.2 Stabiloitumisen tunnistaminen

Päällystyskoneen säätöjärjestelmä nostaa koneen nopeuden ensin ajonopeuteen asti suoraviivaisesti, ja senjälkeen pyrkii saamaan paperin "laatuputkeen", eli paperin kosteuden ja neliöpainon pysymään halutuissa rajoissa. "Laatuputkeen" päästään tyypillisesti "värähtelyn" kautta. Ylösajon onnistuminen on määritelty laatuputkeen pääsemisen kautta: mitä vähemmän syntyy hylkyä (eli mitä nopeammin laatuputkeen päästään), sitä onnistuneempi ylösajo. Koska koko projektin tarkoitus on tutkia alkuparametrien vaikutusta ylösajon onnistumiseen, on stabiloitumisen luotettava tunnistaminen ensiarvoisen tärkeää koko projektin onnistumisen kannalta.

Tähänkin ongelmaan olisi voitu käyttää lukuisia menetelmiä (waveletit, Fourier-analyysi, Kalman-suodatin yms.), mutta lopulta päädyttiin yksinkertaiseen ratkaisuun:

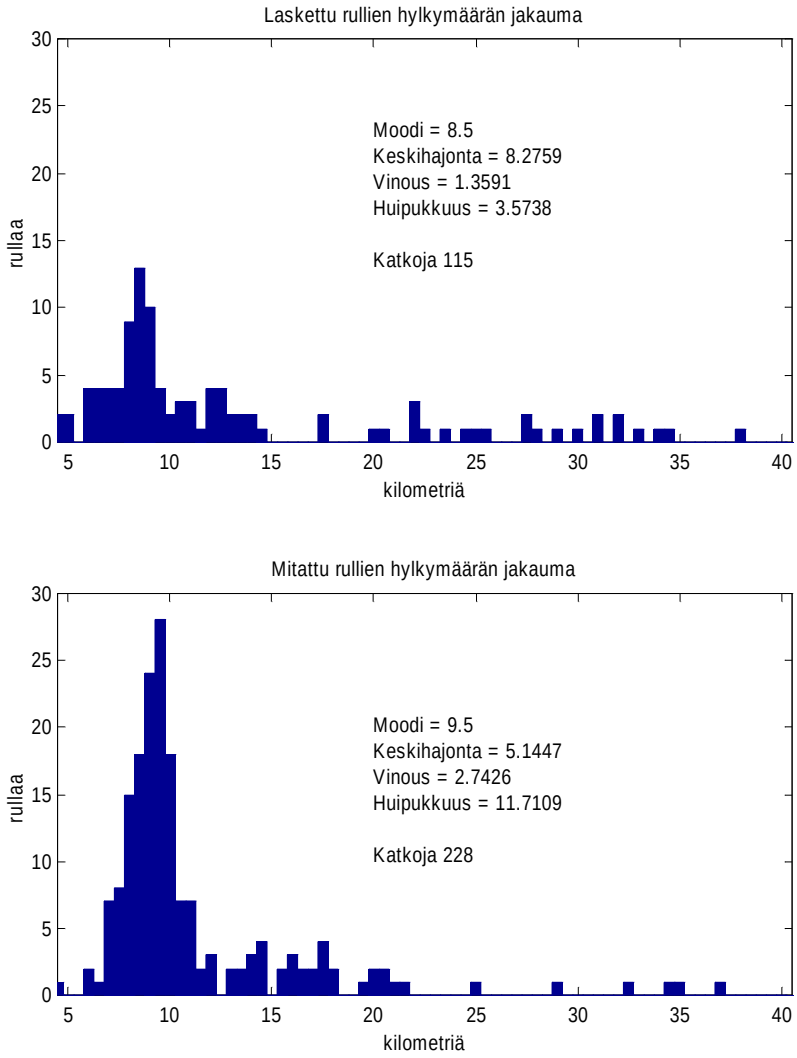
Määritellään päällystyskoneen olevan laatuputkessa hetkellä t, jos m minuutin ajan hetkestä t eteenpäin 4. aseman kosteuden vaihteluväli (suurimman ja pienimmän arvon erotus) on pienempi kuin k, ja 4. aseman neliöpainon vaihteluväli on pienempi kuin n. Tehtäväksi jää enää määritellä parametreille m, k ja n sopivat arvot.

Asiakkaalta saatiin historiatietoa rullissa havaituista hylkymääristä, jotka on kuvattu histogrammina (Kuva 3.1, kuvaaja "mitattu rullien hylkymäärän jakauma"). Tämän jälkeen kokeiltiin systemaattisesti yli 200 erilaista parametrien m, k ja n kombinaatiota, jotta saatiin mahdollisimman paljon havaittua hylkymäärää muistuttava jakauma (Kuva 3.1, kuvaaja "laskettu hylkymäärän jakauma"). Tässä tarkasteltiin jakaumien moodia, keskihajontaa, vinoutta ja huipukkuutta, järjestäen yritteet kunkin tunnusluvun mukaan paremmuusjärjestykseen, ja valiten sellaisen yritteen joka oli useimmalla tunnusluvulla mitaten parhaan 10% joukossa. Laskettu hylkymäärä "integroitiin" koneen nopeuden ja aikaleimojen erotusta käyttäen. Parhaat parametrien arvot olivat:

m: 8 minuuttia
k: 2*0,56
n: 2*30

Huomionarvoista näissä on, että tarkasteluputken pituus on sangen lyhyt, kosteuden vaihteluväli tiukahko, ja neliöpainon sallittu vaihteluväli laaja. Tästä ei voi suoraan päätellä, että päällystyskonetta käytännössä ajettaisiin näin – mielenkiintoisen, käytännön laatuputkea koskevan tutkimushypoteesin tästä voisi ehkä muodostaa (uutta tutkimusta varten).

Jakaumia verrataan siksi, että asiakkaalta saatua tietoa rullien hylkymääristä ei voitu täysin onnistuneesti yhdistää aikasarjatietoon – saattoi olla esimerkiksi kaksi rullaa, joilla oli sama aikaleima. Siksi pyrittiin saamaan jakaumat kohdalleen yksittäisissä ylösajoissa syntyneiden hylkymäärien sijaan.



Kuva 3.1. Mitatut ja lasketut hylkymäärät.

Laskettuun hylkymäärään jouduttiin lisäämään vakio, jotta jakaumien huiput saataisiin kohdalleen – tämä vakio voisi kuitenkin olla mielivaltainenkin, ja silti alkuparametrien hyvyysvaikutusta voitaisiin mielekkäästi tutkia. Myös kolmen ja viiden aikaleiman liukuvien keskiarvojen käyttöä kokeiltiin – ne "terävöittivät" jakaumaa, erityisesti vähentäen "oikeanpuoleisen hännän" paksuutta. Oleellista muutosta parempaan ei kuitenkaan havaittu.

Lisäksi jokaisesta katkosta poimittiin 4. aseman neliöpainot ennen ja jälkeen katkojen, katsomalla minkä vaihteluvälin sisällä se liikkuu 8 minuutin putkessa, ja valitsemalla vaihteluvälin keskipiste.

4 Tilastollinen analyysi

Tilastolliset analyysimenetelmät ovat klassillisia menetelmiä, joiden avulla pyritään tutkimaan aineiston muuttujien välisiä suhteta, testaamaan hypoteeseja aineiston avulla ja tekemään erilaisia johtopäätöksiä. Seuraavassa on sovellettu korrelaatioanalyysiä ja pääkomponenttianalyysiä aineiston laadun analysointiin ja muuttujien määrän vähentämiseen.

4.1 Korrelaatioanalyysi

4.1.1 Korrelaatioanalyysin periaatteista

Korrelaatioanalyysin [Coh95] tavoitteena on tutkia, miten aineiston eri muuttujat korreloivat keskenään. Korrelaation avulla pyritään etsimään muuttujia, joilla on vaikutusta toisiinsa. Korrelaatiota analysoitaessa tulee ottaa huomioon, että korrelaatio on vain yksi tapa ilmaista muuttujien välistä vaikutusta. Korrelaation avulla ei voi tulkita muuttujien riippumattomuutta eikä muuttujien välistä kausaalisuhdetta.

Korrelaatioanalyysissä kaikkien muuttujien välille lasketaan korrelaatiokerroin R (correlation coefficient) ja tunnusluku korrelaation merkittävyydestä (significance). Korrelaatiokerroin on välillä [-1,...1] missä 1 tarkoittaa täydellistä samansuuntaista ja -1 täydellistä vastakkaissuuntaista korrelaatiota sekä 0 sitä, että korrelaatiota ei ole. Perinteisesti kertoimen laskennassa käytetään Pearsonin korrelaatiota, mutta siinä on oletuksena muuttujien jaukautuminen normaalijakauman mukaisesti. Koska suuri osa muuttujista ei täytä tätä oletusta, analyysi on tehty käyttäen Spearmanin järjestyskorrelaatiota. Järjestyskorrelaatiossa korreloitavien muuttujien arvot lajitellaan kasvavaan järjestykseen ja korrelaatio lasketaan laskennassa saatujen sijalukujen (rank) perusteella.

Seuraavissa analyyseissä korrelaatio katsotaan huomattavaksi, kun korrelaatiokertoimen R itseisarvo $|R|>0.6$. Myös pienempiä korrelaatioita esitetään, jos ne ovat kokonaisuuden kannalta tärkeitä. Korrelaatioanalyysin tulosmatriisia ei matriisin laajuuden vuoksi esitetä tässä dokumentissa. Matriisista on poimittu havaintoja tärkeimmistä korrelaatioista.

4.1.2 Analyysi havaintoaineistolle A

Havaintoaineiston A korrelaatioanalyysissä on analysoitu mittausmuuttujien sekä muuttujien TOTAL_INTERVAL, WASTE_APP ja DELTA_BASIS_WEIGHT välistä korrelaatiota katkoaineistossa. Aineisto sisältää 192 katkoa. Muuttuja WASTE_APP (syntyvän hylyn määrä) ei korreloi minkään muun muuttujan kanssa merkittävästi. Muuttujalla on pieni korrelaatio (0.340) muuttujan DELTA_BASIS_WEIGHT (ylösajon yhteydessä ta-

pahtuva neliöpainon muutos). Muuttuja DELTA_BASIS_WEIGHT ei sen sijaan korreloi merkittävästi minkään muun muuttujan kanssa.

Eri asemien leijukuivainten ilmamääriä kuvaavat muuttujat (esim. S4_AID8F) korreloivat voimakkaasti keskenään (R > 0.850). Edellämainitut muuttujat korreloivat negatiivisesti infrakuivaimen tehojen (esim. S3_IRD5P) kanssa (R luokkaa –0.400). Lisäksi infrakuivaimien tehot korreloivat keskenään kertoimilla R luokkaa 0.400 – 0.550.

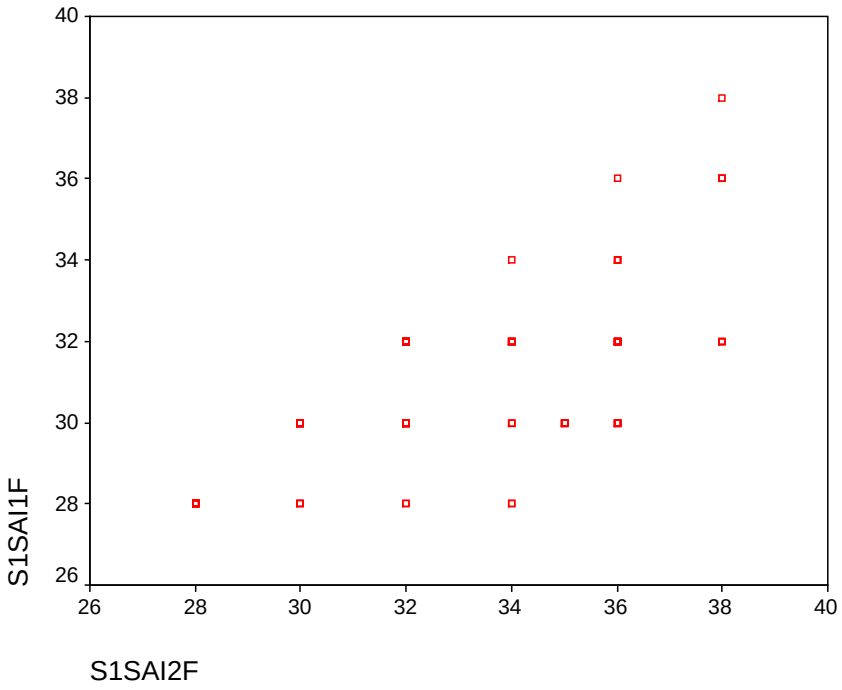
Korrelaatioanalyysissä havaitaan, että eri asemien ilmamääriä muuttujat korreloivat voimakkaasti keskenään. Lisäksi asemien tehomuuttujat korreloivat keskenään jonkin verran. Korrelaatioanalyysin tuloksena voidaan päätellä, että osa muuttujista voidaan poistaa mallista tai muuttujien havaintomäärää tulisi lisätä muuttujan arvojen varioinnin selvittämiseksi.

4.1.3 Analyysi havaintoaineistolle B

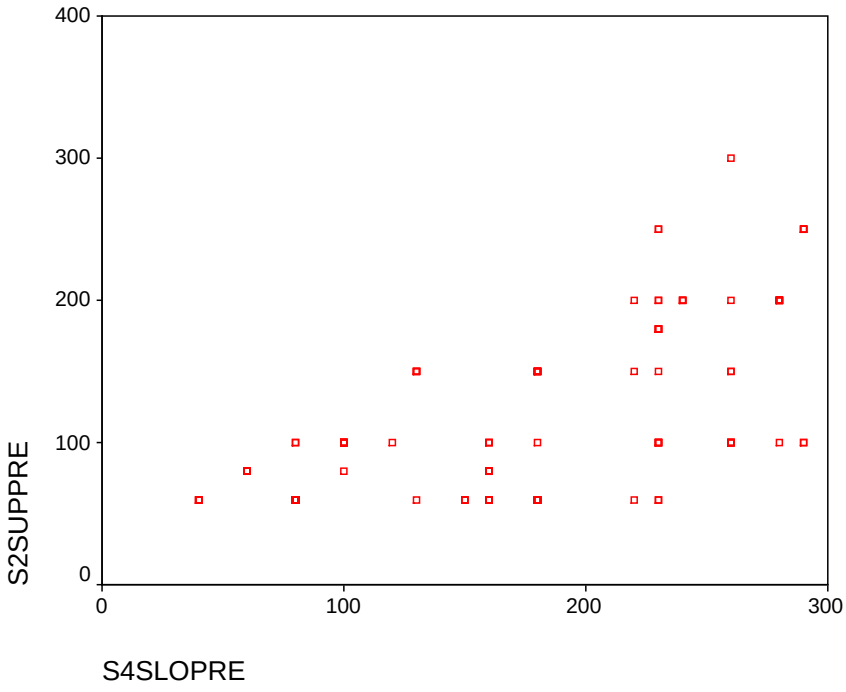
Havaintoaineiston B korrelaatioanalyysissä tarkastellaan kaikkien havaintoaineiston B muuttujien keskinäisiä korrelaatioita. Aineisto koostuu säätöarvoista ja se sisältää 143 katkoa. Kuten aineiston A analyysissä, muuttuja WASTE_APP ei korreloi merkittävästi minkään muun muuttujan kanssa. Yleisemminkin tarkasteltuna muuttujien väliset korrelaatiot ovat huomattavasti heikompia kuin havaintoaineistossa A. Havaintoaineiston B vähäisempi korrelaatioiden määrä havaintoaineistoon A verrattuna johtuu siitä, että havaintoaineistoon B on poimittu vain osa havaintoaineistoa A vastaavista muuttujista.

Eri asemien leijukuivainten ilmamääriä kuvaavat muuttujat, esim. S1S_AIDR2F (S1S_AIDR2_F) ja S1S_AIDR2F (S1S_AIDR2_F) korreloivat keskenään, mutta heikommin kuin havaintoaineistossa A (R on luokkaa 0.650, Kuva 4.1). Lisäksi eri asemien ylä- ja alapaineet, esim. S2S_UPP_PRE ja S4S_LOW_PRE korreloivat voimakkaasti keskenään (R luokkaa 0.67, Kuva 4.2).

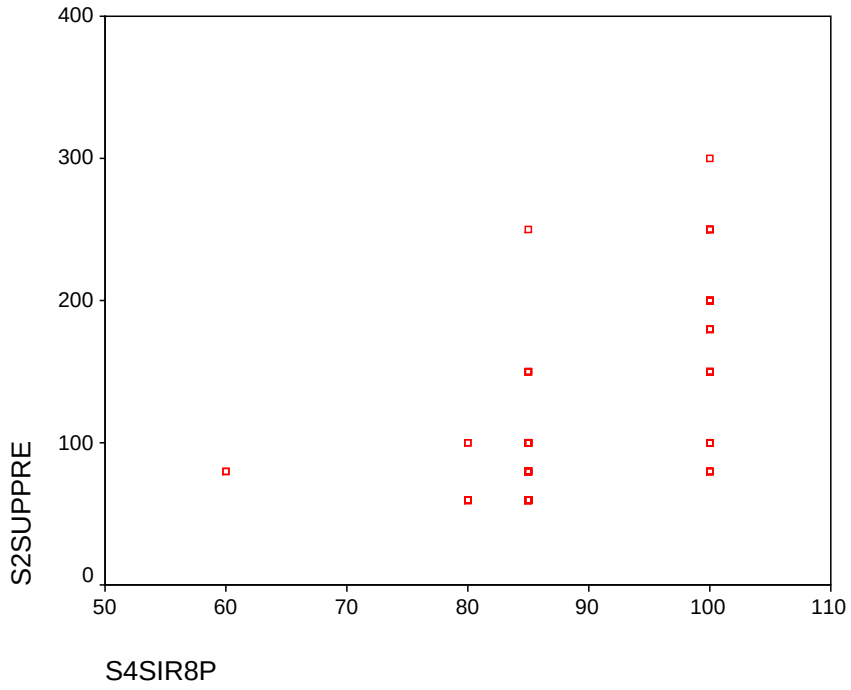
Negatiivista korrelaatiota infrakuivainten tehojen, esim. S4S_IRDR8P (S4S_IRDR8_P), ei sen sijaan ole havaittavissa. Uutena korrelaationa havaintoaineistossa B on havaittavissa infrakuivainten tehojen ja telan paineiden välinen voimakas korrelaatio, esimerkiksi muuttujien S4S_IRDR8P (S4S_IRDR8_P) ja S2S_UPP_PRE välinen korrelaatio on 0.716 (Kuva 4.3).



Kuva 4.1. Eri asemien ilmamäärien välinen korrelaatio.



Kuva 4.2. Eri asemien ala- ja yläpaineiden välinen korrelaatio.



Kuva 4.3. Eri asemien telapaineen ja infrakuivainten tehon välinen korrelaatio.

4.1.4 Yhteenveto

Korrelaatioanalyysissä tarkasteltiin, millä muuttujilla on vaikutusta toisiinsa. Analyysissä havaittiin, että hylyn määrä ei korreloi suoraan minkään toisen havaintoaineiston muuttujan kanssa. Tästä voidaan suurella todennäköisyydellä vetää se johtopäätös, että yksittäisten säätöparametrin arvoja optimoimalla ei voida vaikuttaa merkittävästi prosessien käyttäytymiseen ja pienentää syntyneen hylyn määrää. Korrelaatioanalyysissä havaittiin korrelaatiota eräiden muiden muuttujien välillä. Tätä tietoa voidaan käyttää hyväksi, kun valitaan muuttujia tai tarkastellaan aineiston kattavuutta jollain toisella menetelmällä laadittavaan prosessia kuvaavaan malliin.

4.2 Pääkomponenttianalyysi

4.2.1 Pääkomponenttianalyysin periaatteista

Pääkomponenttianalyysiä [Bish95] käytetään yleisesti havaintoaineiston esikäsittelyssä ennen varsinaista oppimista tai tiedon louhintaa. Tavoitteena on pienentää tarvittavien muuttujien määrää muodostamalla uudet alempiulotteiset kantavektorit, joilla aineisto voidaan kuvata riittävän tarkasti. Etuna on, että usein suppea mittausaineisto kattaa uudessa koordinaatistossa mahdolliset tilat paremmin ja yleistettävyys onnistuu paremmin.

Pääkomponenttianalyysi (myös Karhunen-Loeve muunnos) perustuu attribuuttivektorin lineaariseen projisointiin alempiulotteiseen avaruuteen siten, että syntyvä virhe muodostuu mahdollisimman pieneksi. Usein käytetty virhemitta on neliöllinen virhe, jolloin päädytään seuraavanlaiseen käytännön algoritmiin: Lasketaan aluksi havaintovektorien

keskiarvovektori ja vähennetään tämä havaintovektoreista. Lasketaan sitten näin skaalattujen havaintovektorien kovarianssimatriisi ja etsitään sille suurimmat ominaisarvot ja niitä vastaavat ominaisvektorit. Suurimpia ominaisarvoja vastaavista ominaisvektoreista saadaan uudet kantavektorit (pääkomponentit), joilla mittaukset voidaan kuvata. Kukin pääkomponentti muodostuu tyypillisesti suurimman residuaalivarianssin suuntaisesti.

Laskennassa ei huomioida tavoitekriteeriä, vaan kyseessä on eräänlainen ohjaamaton oppiminen. Ongelmaksi voi joskus muodostua, että esimerkiksi luokkien toisistaan erottamisen mielessä oleellista informaatiota katoaa, vaikka sellaisilla kriteereillä ei juuri olisikaan merkitystä mittausten esittämisen kannalta.

Myös autoassosiatiivisilla hermoverkoilla saadaan aikaiseksi samantyyppisiä projisointeja. Nelikerroksisilla MLP-verkoilla voidaan jopa muodostaa epälineaarisia pääkomponenttimuunnoksia.

4.2.2 Pääkomponenttianalyysin soveltaminen aineiston A tapauksessa

TEMPAT-testitapauksessa mittausaineisto esikäsiteltiin aluksi esittämällä prosessia kuvaavat mittaukset pääkomponentteinaan. Pääkomponenttianalyysi (PCA) suoritettiin kaikille lähtöarvoille ja ylösajonominaisuuksille Waste_APP (laatukriteeri), DELTA_BA (neliöpainon muutos), VELOCITY (nopeus), S4_MOIST, TOTAL_IN (katkon kesto).

Todettakoon, että tässä raportoiduissa tuloksissa on käytetty alunperin kerättyjä edellisen, katkenneen ajon parametreja seuraavan ylösajon ohjausparametreja mallintamassa. Oletuksena on ollut että edellisen ajon parametreja käytetään automaattisesti seuraavan ajon parametreina. Viime aikoina käyttöömme on tullut myös käynnistyksen säätöparametreja, jotka tuntuvat poikkeavan edellisen ajon parametreista.

Oheisessa taulukossa nähdään saatujen pääkomponenttien suhde alkuperäisiin muuttujiin. Kokonaisuudessaan 9 tärkeintä pääkomponenttia selittää melkein 81% kokonaisvarianssista.

Component Matrix[a]

	Component								
	1	2	3	4	5	6	7	8	9
S4_LOW_P	.160	.836	-.104	.135	-.163	-.285	3.006E-02	7.569E-02	-.142
S4_UPP_P	.174	.834	-.102	.133	-.161	-.281	3.087E-02	7.556E-02	-.147
S4_AID8F	.874	6.284E-02	.299	-7.14E-02	-.128	5.223E-02	-4.52E-02	7.890E-02	1.217E-02
S4_AID8T	2.770E-02	.115	.263	-.268	1.148E-02	-.581	-.193	-.332	.207
S4_AID7F	.892	2.062E-03	.297	-4.86E-02	-9.52E-02	8.670E-02	-5.46E-02	9.758E-02	-9.64E-03
S4_AID7T	-.244	.384	6.487E-02	-.212	.320	5.030E-03	-.554	.366	2.366E-02
S4_IRD8P	-.832	3.829E-02	.352	2.109E-02	-.213	5.621E-02	3.379E-02	3.538E-02	-4.23E-02
S4_IRD7P	-.600	.143	.449	6.694E-03	-.295	.208	-9.76E-02	-7.65E-02	4.662E-02
S3_LOW_P	-1.02E-02	.784	-.223	-.228	.121	.340	.211	-6.64E-02	4.834E-02
S3_UPP_P	-2.89E-02	.790	-.216	-.187	.125	.343	.212	-6.72E-02	5.358E-02
S3_AID6F	.869	3.273E-02	.337	-8.80E-02	-.104	.109	.111	2.352E-02	-1.97E-02
S3_AID6T	6.128E-02	4.969E-02	.496	-.219	.213	-.419	8.224E-02	-8.39E-02	.202
S3_AID5F	.875	-1.48E-02	.328	-7.12E-02	-7.38E-02	.105	7.096E-02	7.134E-02	-1.02E-02
S3_AID5T	-.133	.225	.272	-.406	.517	-5.48E-02	-.328	.277	2.704E-02
S3_IRD6P	-.677	.141	.354	-9.88E-02	-.247	2.571E-02	5.310E-02	.131	-8.18E-02
S3_IRD5P	-.573	.158	.475	-2.21E-02	-.298	.226	-.112	-.124	4.967E-02
S2_LOW_P	.109	.271	.227	.884	.148	-5.57E-02	-8.26E-02	2.431E-02	.146
S2_UPP_P	.121	.269	.227	.881	.155	-5.44E-02	-8.34E-02	2.525E-02	.148
S2_AID4F	.915	3.158E-02	.226	-3.29E-02	3.128E-02	.112	-1.17E-02	-9.42E-02	-4.48E-02
S2_AID4T	5.160E-02	.397	.430	-.204	.285	-6.24E-02	8.421E-02	-.465	-.184
S2_AID3F	.910	5.940E-02	.298	-2.91E-02	-9.17E-02	8.524E-02	-5.20E-02	5.193E-02	-1.94E-02
S2_AID3T	-.288	8.125E-02	9.416E-02	.188	.540	.423	5.364E-02	-.215	-6.36E-02
S2_IRD4P	-.889	1.056E-02	.270	-2.33E-03	-.176	1.339E-02	7.799E-02	7.683E-02	-4.69E-02
S2_IRD3P	-.725	9.896E-02	.507	-3.44E-02	-.115	.103	5.010E-03	-2.66E-02	-3.70E-02
S1_LOW_P	-3.51E-02	9.333E-02	6.970E-02	-.144	1.694E-02	-1.81E-04	.412	.182	.797
S1_UPP_P	-6.13E-02	2.659E-02	9.246E-02	-3.42E-02	.226	-.333	.543	.410	-.332
S1_AID2F	.887	-3.31E-02	.303	-1.72E-02	-.133	.129	3.874E-02	7.349E-02	-7.22E-02
S1_AID2T	2.277E-02	-.286	.377	.104	.366	-.192	.256	-.221	-.143
S1_AID1F	.890	-2.77E-02	.314	7.270E-03	-9.71E-02	.146	3.877E-03	.130	-3.37E-02
S1_AID1T	-.256	-.302	.238	3.418E-02	.464	.314	5.431E-02	.146	-1.72E-02
S1_IRD2P	-.864	4.459E-02	.248	1.170E-02	-.181	3.290E-02	9.972E-02	8.968E-02	-7.26E-02
S1_IRD1P	-.484	-.136	.379	8.587E-02	.111	-9.33E-02	9.131E-02	.227	-3.47E-02

Extraction Method: Principal Component Analysis.

a. 9 components extracted.

Taulukko 4.1 Pääkomponenttimatriisi.

Taulukosta 4.1 voidaan havaita, että 1. pääkomponentti korreloi positiivisesti vahvasti kaikkien leijukuivainten ilmamäärämittausten (AIDRn_F) kanssa ja vahvasti negatiivisesti infrakuivaimien tehosäätöjen (IRDn_P) kanssa. Tämä pääkomponentti selittää noin 33% kokonaisvarianssista.

Toinen pääkomponentti korreloi positiivisesti vahvasti kolmannen ja neljännen vaiheen minimi (LOW_P) ja maksimi (UPP_P) painesäätöjen kanssa. Tämä pääkomponentti selittää noin 11% kokonaisvarianssista.

Kolmas pääkomponentti korreloi positiivisesti melko vahvasti seuraavien kuivain lämpötilamittausten kanssa (S3_AID6_T, S2_AID4_T, S1_AID2_T) ja infrakuivaimen tehosäätöjen (IRDn_P) kanssa. Tämä pääkomponentti selittää noin 9 % kokonaisvarianssista.

Neljäs pääkomponentti korreloi positiivisesti vahvasti toisen vaiheen minimi (S2_LOW_P) ja maksimi (S2_UPP_P) tehosäätöjen kanssa ja melko vahvasti negatiivi-

sesti kolmannen vaiheen kuivainlämpötilan (S3_AID5_T) kanssa. Tämä pääkomponentti selittää noin 7 % kokonaisvarianssista.

Total Variance Explained

	Initial Eigenvalues			Extraction Sums of Squared Loadings		
Component	Total	% of Variance	Cumulative %	Total	% of Variance	Cumulative %
1	10.782	33.693	33.693	10.782	33.693	33.693
2	3.447	10.773	44.466	3.447	10.773	44.466
3	2.960	9.250	53.716	2.960	9.250	53.716
4	2.162	6.758	60.474	2.162	6.758	60.474
5	1.768	5.524	65.997	1.768	5.524	65.997
6	1.554	4.857	70.854	1.554	4.857	70.854
7	1.183	3.698	74.552	1.183	3.698	74.552
8	1.043	3.260	77.812	1.043	3.260	77.812
9	1.011	3.160	80.973	1.011	3.160	80.973
10	.903	2.821	83.794			
11	.704	2.199	85.993			
12	.684	2.138	88.131			
13	.616	1.924	90.055			
14	.557	1.742	91.797			
15	.536	1.674	93.471			
16	.411	1.284	94.754			
17	.352	1.100	95.855			
18	.299	.936	96.791			
19	.223	.698	97.489			
20	.194	.607	98.096			
21	.182	.569	98.665			
22	.160	.501	99.167			
23	7.340E-02	.229	99.396			
24	6.329E-02	.198	99.594			
25	4.749E-02	.148	99.742			
26	2.869E-02	8.966E-02	99.832			
27	2.249E-02	7.028E-02	99.902			
28	1.184E-02	3.699E-02	99.939			
29	1.007E-02	3.147E-02	99.971			
30	8.333E-03	2.604E-02	99.997			
31	6.159E-04	1.925E-03	99.999			
32	4.513E-04	1.410E-03	100.000			

Extraction Method: Principal Component Analysis.

Taulukko 4.2 Kokonaisvarianssin selitysasteet.

Muut pääkomponentit voidaan analysoida vastaavasti tutkimalla taulukoita 4.1 ja 4.2.

4.2.3 Pääkomponenttianalyysin soveltaminen aineiston B tapauksessa

TEMPAT-testitapauksen mittausaineisto B esikäsiteltiin aluksi esittämällä prosessia kuvaavat mittaukset pääkomponentteinaan. Pääkomponenttianalyysi (PCA) suoritettiin kaikille muille muuttujille paitsi WASTE_APP (laatukriteeri), BAS_W_NEW ja BAS_W_OLD (neliöpainon tuleva ja edellinen arvo), TOTAL_INT (katkon kesto). BAS_W_NEW ja BAS_W_OLD ja TOTAL_INT jätettiin pois PCA-analyysistä, koska ne toimivat syöteparametreina päätöksenteossa. WASTE_APP on laatukriteerinä ja sitä halutaan siis käyttää itsenäisenä.

Oheisessa taulukossa 4.3 nähdään saatujen pääkomponenttien suhde alkuperäisiin muuttujiin. Kokonaisuudessaan 7 tärkeintä pääkomponenttia selittää melkein 81% kokonaisvarianssista. Seuraavassa lyhyet tulkinnat pääkomponenteista. Tarkemmat tiedot saa taulukkoja tutkimalla.

Oheisessa taulukossa (Taulukko 4.3) on alleviivattu sellaiset arvot, joilla absoluuttinen arvo on yli 0.5. Voidaan havaita, että ensimmäinen pääkomponentti korreloi erityisen voimakkaasti S{432}_UPP_PRE ja S{432}_LOW_PRE säätöjen kanssa. Myös S{234}S_AI{357}DR_F muuttujien (leijukuivainten ilmamäärämittausten) kanssa on vahva korrelaatio.

Component Matrix[a]

	Component						
	1	2	3	4	5	6	7
S1SAI1T	.122	-6.49E-02	7.157E-02	**.677**	-.479	.326	-2.83E-02
S1SAI2T	-.128	6.230E-02	.448	.431	.360	.303	-2.71E-02
S1SAI1F	.323	**-.629**	.404	.166	-.114	.290	-3.57E-02
S1SAI2F	9.166E-02	**-.652**	**.577**	-.182	4.150E-02	.120	-6.92E-02
S2SAI3T	.344	**.538**	-.399	-4.46E-02	-.276	.273	-.295
S2SAI4T	.347	**.702**	.224	5.078E-02	.243	.189	8.728E-02
S2SAI3F	**.711**	.237	-.176	-.403	-7.89E-02	.325	-1.62E-02
S2SAI4F	.559	.331	5.341E-02	**-.563**	7.049E-02	.306	-4.59E-03
S2SUPPRE	**.714**	-.299	-.453	.136	5.464E-02	.270	5.019E-02
S2SLOPRE	**.714**	-.299	-.453	.136	5.464E-02	.270	5.019E-02
S3SIR6P	.438	-.214	-.135	-.115	.177	-9.83E-02	.743
S3SAI5T	**.518**	.388	9.727E-02	.266	-.477	-.304	.182
S3SAI6T	8.059E-02	**.568**	.340	.355	.235	-8.96E-03	.136
S3SAI5F	**.700**	9.639E-02	.424	-.106	-.352	1.750E-02	1.225E-02
S3SAI6F	**.558**	8.739E-02	**.637**	-.152	-.131	2.030E-02	9.824E-02
S3SUPPRE	**.863**	-4.93E-02	-3.26E-02	.109	.129	-.223	-.314
S3SLOPRE	**.863**	-4.93E-02	-3.26E-02	.109	.129	-.223	-.314
S4SIR8P	**.697**	-.175	-.148	7.729E-02	-6.75E-02	.207	.310
S4SAI7T	**.707**	.164	-4.37E-02	8.657E-02	-.297	-.298	.168
S4SAI8T	**.380**	.423	2.951E-02	.241	.422	.110	5.252E-02
S4SAI7F	**.821**	-9.12E-02	.166	-.169	-5.06E-02	-4.25E-02	-7.40E-02
S4SAI8F	**.733**	-1.90E-02	.366	-.225	7.868E-02	-.111	-9.66E-02
S4SUPPRE	**.871**	-.187	-.131	.169	.238	-.201	-8.30E-02
S4SLOPRE	**.871**	-.187	-.131	.169	.238	-.201	-8.30E-02

Extraction Method: Principal Component Analysis.

a. 7 components extracted.

Taulukko 4.3 Pääkomponenttimatriisi.

Tämä pääkomponentti selittää lähes 38% kokonaisvarianssista (Taulukko 4.4).

Toinen pääkomponentti korreloi kohtuullisen voimakkaasti S{23}S_AIDR{46}_T lämpötilamittausten kanssa sekä negatiivisesti voimakkaasti S1S_AIDR{12}_F leijukuivainten ilmavirtaussäätöjen kanssa. Tämä pääkomponentti selittää lähes 12% kokonaisvarianssista.

Kolmas pääkomponentti korreloi melko vahvasti S{13}S_AIDR{26}_F ilmavirtaussäätöjen kanssa. Tämä pääkomponentti selittää lähes 10 % kokonaisvarianssista.

Muut pääkomponentit voidaan analysoida vastaavasti tutkimalla taulukoita 4.3 ja 4.4.

Total Variance Explained

	Initial Eigenvalues			Extraction Sums of Squared Loadings		
Component	Total	% of Variance	Cumulative %	Total	% of Variance	Cumulative %
1	8.836	36.816	36.816	8.836	36.816	36.816
2	2.810	11.710	48.525	2.810	11.710	48.525
3	2.307	9.614	58.139	2.307	9.614	58.139
4	1.702	7.090	65.229	1.702	7.090	65.229
5	1.408	5.866	71.095	1.408	5.866	71.095
6	1.192	4.968	76.064	1.192	4.968	76.064
7	1.073	4.471	80.535	1.073	4.471	80.535
8	.911	3.795	84.330			
9	.654	2.726	87.057			
10	.562	2.340	89.397			
11	.427	1.781	91.178			
12	.369	1.539	92.717			
13	.348	1.449	94.166			
14	.331	1.381	95.547			
15	.268	1.119	96.665			
16	.255	1.063	97.728			
17	.160	.665	98.393			
18	.143	.596	98.989			
19	9.829E-02	.410	99.398			
20	8.236E-02	.343	99.741			
21	6.206E-02	.259	100.000			
22	1.523E-16	6.345E-16	100.000			
23	8.373E-18	3.489E-17	100.000			
24	-5.13E-16	-2.138E-15	100.000			

Extraction Method: Principal Component Analysis.

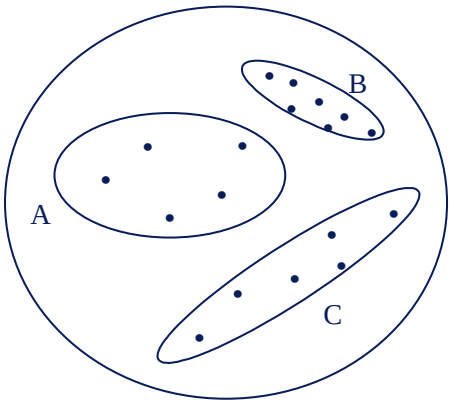
Taulukko 4.4 Kokonaisvarianssin selitysasteet.

4.2.4 Yhteenveto

Pääkomponenttianalyysi soveltuu aineiston esikäsittelyyn ennen varsinaista analysointia. Sen avulla aineisto voidaan kuvata pienemmässä ulottuvuudessa uusilla kantavektoreilla. Havaintoavaruuden supistaminen on suositeltavaa suppeilla havaintoaineistoilla ennen yleistystä. Näitä tuloksia on hyödynnetty luvun 5.3 AutoClass analyysissa ja luvun 7.4 Bayes-verkko analyysissä.

5 Ryvästys

Ryvästyksellä (clustering) etsitään aineistosta luokkia, joiden perusteella aineisto voidaan jakaa toisistaan eroaviin osiin [Eve77]. Ryvästys on luonteeltaan ohjaamatonta, eli ryvästystä ei suoriteta minkään yksittäisen muuttujan tai muuttujien suhteessa vaan ryvästyksellä pyritään identifioimaan luokkia pelkästään datan perusteella. Kuva 5.1 on esimerkki ryvästyksestä. Annetusta aineistosta muodostetaan kolme luokkaa. Syntyneitä luokkia voidaan kuvata luokkien keskipisteillä (cluster centers), luokkien välisillä etäisyyksillä ja luokan alkioiden välisillä etäisyyksillä. Tässä luvussa käydään läpi tärkeimpiä ryvästysmenetelmiä ja niiden soveltamista tapaustutkimuksessa.

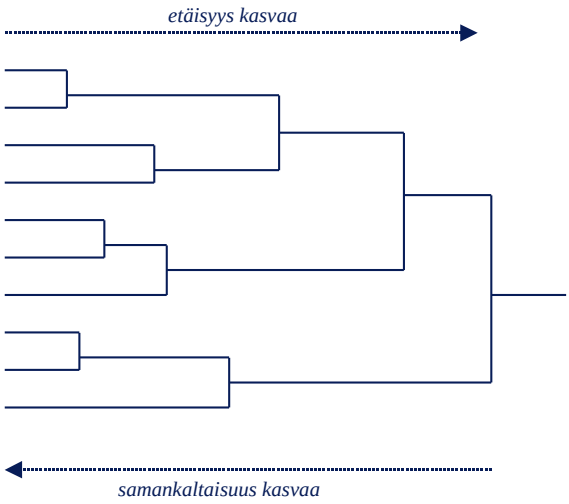


Kuva 5.1. Esimerkki ryvästyksestä.

5.1 Hierarkinen ryvästys

5.1.1 Hierarkisen ryvästyksen periaate

Hierarkinen ryvästys (hierarchical clustering) [Har75] muodostaa aineistosta puumaisen esityksen, jossa ryhmät voidaan tunnistaa puun eri tasojen sisällä olevista yhteyksistä. Kuva 5.2 on esimerkki hierarkisen ryvästyksen tulospuusta. Puun vasemmassa reunassa ovat yksittäiset ainestorivit ja puurakenne kuvaa, miten rivit yhdistetään toisiinsa. Hierarkisen ryvästyksen avulla saadaan helposti yleiskäsitys siitä, miten aineisto on jaettavissa luokkiin. Hierarkisen ryvästyksen etuna on, että luokkien määrää ei tarvitse kiinnittää etukäteen vaan ne voidaan päätellä tulospuusta. Käytännön tapauksissa puurakenteen tulkinta voi olla vaikeaa kun aineisto on laaja.



Kuva 5.2. Esimerkki hierarkisen ryvästyksen tulospuusta.

Ryvästysalgoritmin käytännön toteutuksessa lasketaan ryhmien välistä etäisyyttä ja yksittäisten alkioiden välistä etäisyyttä. Ryvästettäessä pyritään maksimoimaan ryhmien välinen etäisyys ja laskennan apuna käytetään alkioiden välistä etäisyyttä. Tyypillisiä esimerkkejä ryhmien välisen etäisyyden laskentamenetelmistä ovat eri luokkien lähimpien naapureiden vertailu (nearest neighbor), kauimpana olevien naapureiden vertailu (furthest neighbor) ja keskipisteiden vertailu (median clustering). Tyypillisiä esimerkkejä alkioiden välinen etäisyyden laskentamenetelmistä ovat alkioiden välinen suora etäisyys eli euklidinen etäisyys (Euclidean distance) ja Manhattan-etäisyys (Manhattan distance) jossa etäisyys määritellään ortogonaalisten vektoreiden perusteella. Ryvästyksellä saavutettavat tulokset riippuvat voimakkaasti menetelmästä, jota käytetään ryhmien ja alkioiden etäisyyksien arvioinnissa.

5.1.2 Aineisto ja käytetyt menetelmät

Hierarkisessa ryvästyksessä analysoidaan havaintoaineistoa A. Ryvästyksessä käytetty aineisto sisältää 212 katkoa. Aineiston muuttujat ovat mittaustiedot sekä lasketut muuttujat WASTE_APP (laskettu hylkymäärä), BAS_W_NEW (ylösajon stabiloitunut neliöpaino) ja TOTAL_INTERVAL (katkon pituus). Muuttujaa WASTE_APP ei käytetä ryvästettäessä, mutta ryvästyksen tuloksia arvoidaan sen suhteen. Aineisto skaalataan välille [0,1] siten, että jakauman muoto säilyy.

Analyysissä käytetty laskenta-algoritmi on hierarkinen ryvästys. Ryvästyksessä käytetty ryhmien välisen etäisyyden mitta on kauimpien naapureiden vertailu ja alkioiden välinen etäisyysmitta on Eulidinen etäisyys. Ryvästyksen tuloksena saadusta puurakenteesta poimitaan eri ratkaisuja, joissa luokkien määrä on välillä 2-10. Koska ryvästys on ohjaamaton toiminto, ei siinä voida asettaa erityistä painoarvoa tietyille muuttujille, joten kaikki muuttujat ovat analyysissä samanarvoisia. Analyysissä ja tulosten esittämisessä on käytetty SPSS tilasto-ohjelmistoa.

5.1.3 Ryvästyksen tulokset

Tässä tutkimuksessa on selvitetty, onko ryvästyksellä löydettävillä klustereilla mahdollista erottaa sellaisia tekijöitä, jotka selittäisivät laskettua muuttujaa WASTE_APP

(syntyneen hylyn määrä). Analyysissä tutkittiin ratkaisuja, jossa syntyneiden luokkien määrä on välillä 2-10 kpl. Seuraavassa esitetään luokkajaon tunnusluvut taulukoina. Ensimmäisen taulukon yhteydessä käydään esimerkkinä lävitse, miten tulostaulukoita tulkitaan. Tunnuslukuja ei esitetä kaikille mahdollisille luokkajaoille vaan tuloksista on poimittu ne, joiden on katsottu olevan ongelman kannalta mielenkiintoisia.

Taulukko 5.1 esittää, millaisia luokkia ryvästysalgoritmi on löytänyt, kun luokkien määrä on 2 kpl. Luokkia arvioidaan ja tunnusluvut lasketaan muuttujan WASTE_APP suhteen. Taulukosta nähdään, että luokassa 1 on 197 katkoa ja luokassa 2 15 katkoa. Luokan 1 hylkymäärän keskipiste laskettuna aritmeettisena keskiarvona (sarake mean) on 11.9 km ja hylkymäärän keskipisteen mediaani (sarake media) on 9.6 km. Lisäksi taulukosta havaitaan luokan 1 arvojen minimi- ja maksimiarvot. Vastaavat tunnusluvut esitetään myös 3, 5, 6 ja 10 luokan tapauksille(Taulukko 5.2 - Taulukko 5.5).

	N	Mean	Median	Minimum	Maximum
1	197	11.8990	9.6410	6.94	76.32
2	15	14.9726	14.1047	7.07	35.47
Total	212	12.1165	9.8432	6.94	76.32

Taulukko 5.1. Ryvästyksellä saatujen luokkien tunnusluvut kahden luokan ratkaisulle.

	N	Mean	Median	Minimum	Maximum
1	161	11.7256	9.5052	6.94	76.32
2	36	12.6749	11.1515	7.15	27.04
3	15	14.9726	14.1047	7.07	35.47
Total	212	12.1165	9.8432	6.94	76.32

Taulukko 5.2. Ryvästyksellä saatujen luokkien tunnusluvut kolmen luokan ratkaisulle.

	N	Mean	Median	Minimum	Maximum
1	128	11.5750	9.4673	6.94	76.32
2	33	12.3096	10.4777	7.49	25.68
3	36	12.6749	11.1515	7.15	27.04
4	8	15.7392	13.5081	7.07	35.47
5	7	14.0964	14.1047	9.50	18.95
Total	212	12.1165	9.8432	6.94	76.32

Taulukko 5.11. Ryvästyksellä saatujen luokkien tunnusluvut viiden luokan ratkaisulle.

	N	Mean	Median	Minimum	Maximum
1	113	11.7107	9.4918	6.94	76.32
2	33	12.3096	10.4777	7.49	25.68
3	15	10.5522	9.0682	7.76	20.34
4	36	12.6749	11.1515	7.15	27.04
5	8	15.7392	13.5081	7.07	35.47
6	7	14.0964	14.1047	9.50	18.95
Total	212	12.1165	9.8432	6.94	76.32

Taulukko 5.12. Ryvästyksellä saatujen luokkien tunnusluvut kuuden luokan ratkaisulle.

	N	Mean	Median	Minimum	Maximum
1	54	11.9127	9.2842	6.94	76.32
2	13	10.5148	9.4715	7.78	18.99
3	59	11.5259	9.6776	7.65	52.87
4	15	10.5522	9.0682	7.76	20.34
5	20	13.4763	11.4054	7.49	25.68
6	32	12.9242	11.5937	7.15	27.04
7	4	10.6801	10.9783	7.97	12.80
8	8	15.7392	13.5081	7.07	35.47
9	4	14.4968	13.8246	11.39	18.95
10	3	13.5626	14.1047	9.50	17.09
Total	212	12.1165	9.8432	6.94	76.32

Taulukko 5.5. Ryvästyksellä saatujen luokkien tunnusluvut kymmenen luokan ratkaisulle.

5.1.4 Yhteenveto

Ryvästyksen tuloksista havaitaan, että ryvästyksellä saadulla luokkajaolla ei pystytä selittämään muuttujaa WASTE_APP. Merkittävä osa havainnoista asettuu kaikissa ratkaisuissa luokkaan 1, jossa luokan minimi-ja maksimiarvot ovat samat kuin koko aineiston vastaavat arvot. Lisäksi luokkien keskipisteet ovat muuttujan WASTE_APP suhteen lähellä toisiaan, joten luokkajaon perusteella on vaikea löytää sellaista tekijää, jolla pystyttäisiin erottamaan ajotilanteet, jotka johtavat eri hylkymääriin.

Edellä mainituista syistä johtuen hierarkisella ryvästyksellä saatuja tuloksia ei voida soveltaa päällystyskoneen ylösajojen luokittelussa hylkymäärän suhteen. Ryvästyksen tuloksia on analysoitu ainoastaan yhden muuttujan suhteen. Jotta ryvästystä voitaisiin käyttää hyväksi prosessin tutkimisessa, tulisi ryvästyksellä löydettyjä luokkia analysoida myös muiden muuttujien suhteen ja pyrkiä päättelemään, mistä luokkien erot johtuvat. Tälläinen analyysi on kuitenkin tämän tutkimuksen ulkopuolella. Edellä mainituista syistä johtuen päätettiin, ettei hierarkista ryvästysmenetelmää sovelleta lainkaan havaintoaineistoon B.

5.2 SOM-tekniikka

5.2.1 Itseorganisoituvien karttojen periaatteet

Itseorganisoituvat kartat (Self-Organizing Maps, SOM) ovat yksi neuraalilaskennan tekniikka. SOM-menetelmässä moniulotteiseen havaintoaineistoon sovitetaan joukko prototyyppivektoreita, jotka levittäytyvät tietyn algoritmin mukaisesti muuttuja-avaruuteen kuvaamaan havaintojen jakaumaa [Koh95]. Tätä prosessia kutsutaan kartan opettamiseksi.

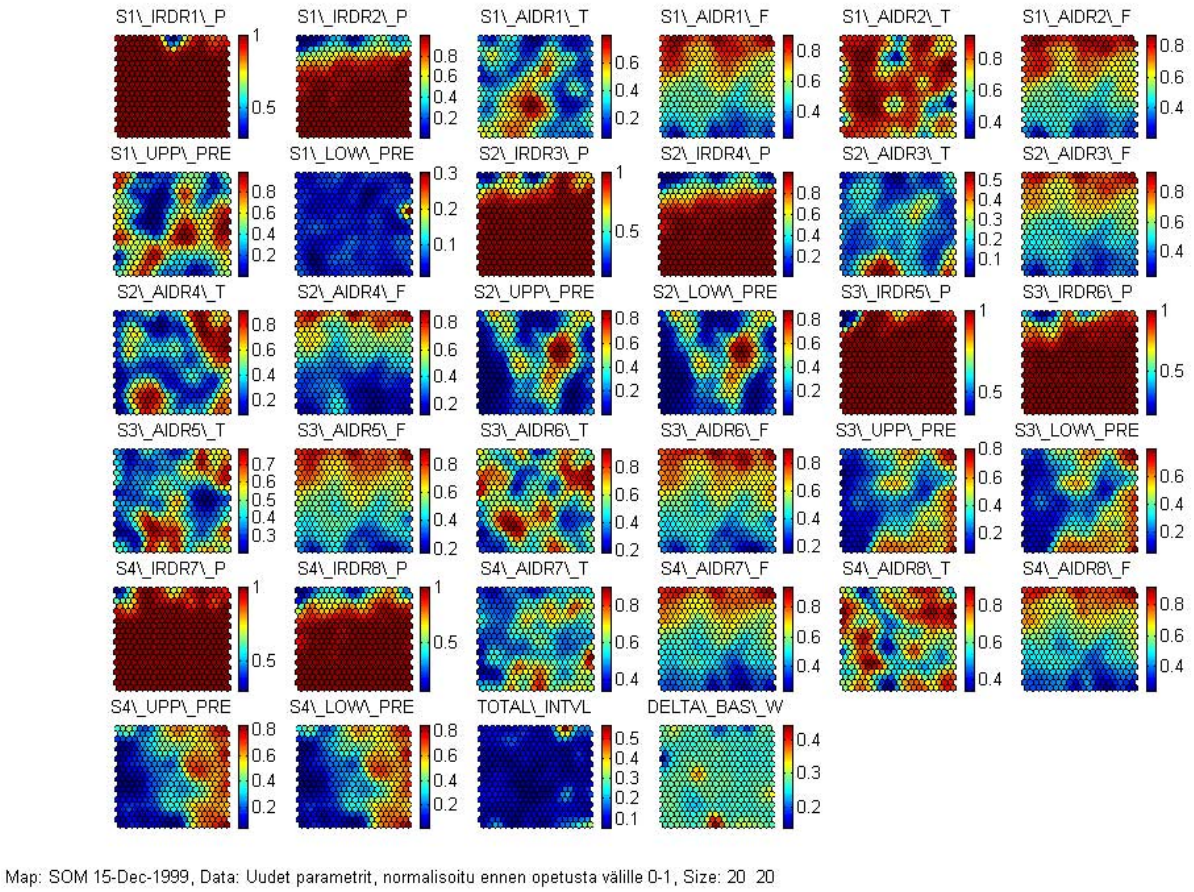
SOM-algoritmissa prototyyppivektoreita opetetaan ilman valvontaa. Opetus tapahtuu askeleittain siten, että jokaisella opetusaskeleella valitaan yksi havainto, ja sitä lähinnä vastaava prototyyppivektori (BMU, best matching unit). Tämän jälkeen BMU:n ja siitä tietyllä etäisyydellä olevien prototyyppivektorien arvoja päivitetään lähemmäs valitun havainnon arvoja. Muutoksen suuruuden määräävät havainnon ja prototyyppivektorin etäisyys, opetustahti ja prototyyppivektorin etäisyys BMU:sta. Kun opetustahtia vähitellen hidastetaan, stabiloituvat prototyyppivektorit kuvaamaan havaintoaineiston jakaumaa. Algoritmin hyvänä puolena on se, että yksittäiset poikkeavat havainnot eivät saa karttaa opetettaessa suurta painoa, jolloin ne karsiutuvat prototyyppivektorien jakaumasta [Ves99].

Piirtämällä opetetut prototyyppivektorit komponenteittain tasahilaisille tasoille siten, että komponenttiensa puolesta toisiaan lähellä olevat vektorit ovat lähellä toisiaan myös kartassa, saadaan ns. SOM-kartta (Kuva 5.3). Kullakin tasolla samanväriset alueet kuvaavat samaa vektorin komponentin arvoa. Esimerkiksi prototyyppivektorin ensimmäistä komponenttia, muuttujaa S1_IRDR1_P:tä kuvaavasta tasosta nähdään, että ko. komponentti saa useimmiten suuria arvoja. Tällä tavoin kartan perusteella voidaan tehdä päätelmiä komponenttien välisistä korrelaatioista, jaotella havaintoaineisto ryhmiin ja tehdä päätelmiä tiettyyn luokkaan kuuluvien havaintojen ominaisuuksista.

5.2.2 Havaintoaineisto A

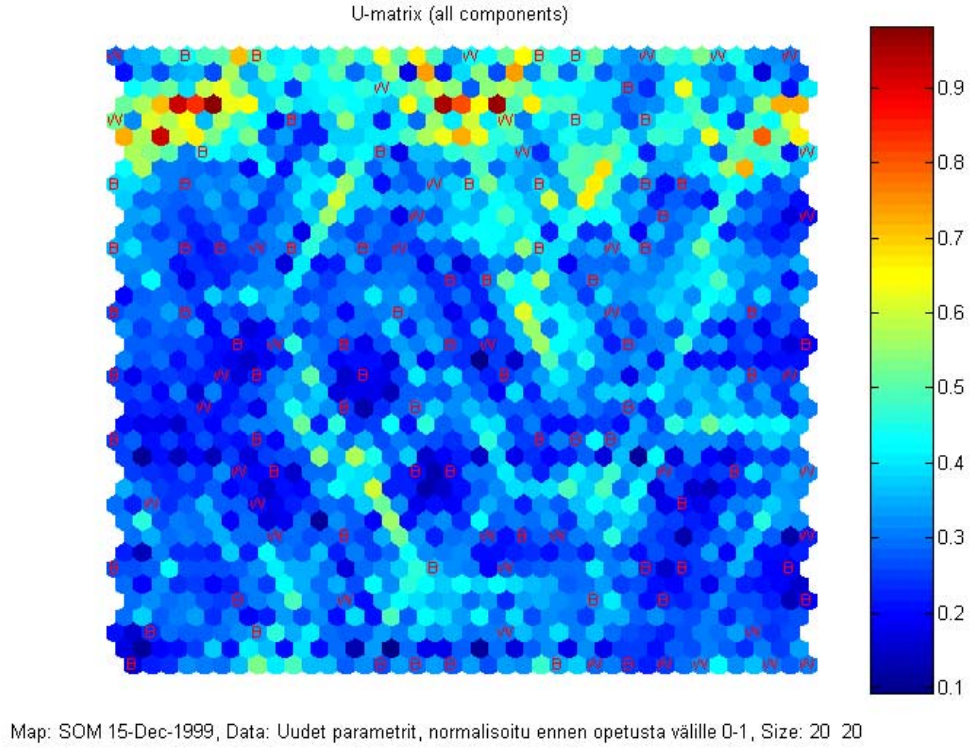
Päällystyskoneen ylösajojen SOM-analyysi toteutettiin käyttäen Matlab-ohjelmistoa sekä sille TKK:n Informaatiotekniikan laboratorion kehittämää SOM-toolbox –funktiokokoelmaa. Ensimmäisessä vaiheessa analyysissä käytettiin muuttujina prosessista saatuja mittausmuuttujia, eli havaintoaineistoa A. SOM-analyysiin otettiin mukaan 32 mittausmuuttujaa ja 2 ylöajon ominaisuutta, katkon pituus TOTAL_INTVL ja neliöpainon muutos DELTA_BAS_W. Analyysissä oletettiin, että mittausmuuttujat ovat samat kuin asetusmuuttujat, joilla ylösajo käynnistetään. Koska mittausmuuttujien arvoalueet eroavat toisistaan huomattavasti, skaalataan kaikki muuttujat ennen kartan opetusta välille 0-1. Näin mikään muuttuja ei saa huomattavaa painoa laskettaessa kartan prototyyppivektorien välisiä samankaltaisuuksia.

Opettamalla 212 ylösajolla 34 muuttujan 20×20 kokoinen SOM-kartta saadaan Kuva 5.3. Kuvan 34 karttaa kuvaavat prototyyppivektorien komponenttien, eli muuttujien arvoja. Karttoja vertailemalla havaitaan muuttujien välisiä korrelaatioita, kuten muuttujien S1_AIDR1_F, S1_AIDR2_F, S2_AIDR3_F, S3_AIDR5_F, S3_AIDR6_F, S4_AIDR7_F ja S4_AIDR8_F väliset.



Kuva 5.3. Pällystyskoneen 212 ylösajolla opetettujen SOM-prototyyppivektorien komponenttikartat.

Yhteenvetona kartan alkioiden samanlaisuudesta ja erilaisuudesta voidaan piirtää ns. U-matriisi, yksikköetäisyysmatriisi, jossa kartan alkiot piirretään erilaisuudestaan (prototyyppivektorien etäisyydestä) huolimatta yhtä kauas toisistaan ja erilaisuus ilmaistaan kartan kuusikulmioiden värien avulla. Kuvassa (Kuva 5.4) on kuvan (Kuva 5.3) karttojen prototyyppivektorien avulla laskettu U-matriisi. Mitä lähempänä U-matriisin kuusikulmion väri on asteikon punaista päätä, sitä enemmän se eroaa lähinaapureistaan. U-matriisiin on lisäksi merkitty "hyviä" ja "huonoja" ylösajoja vastaavat BMU:t B:llä (best, alle 9 km hylkyä) ja W:llä (worst, yli 13 km hylkyä). Koska prototyyppivektoreita on yhtä monta kuin kartan alkioita, eli 400 ja havaintoja vain 212, ei jokaiseen kuusikulmioon osu havaintoa. Havaintojen välissä olevat prototyyppivektorit toimivat interpolointipisteinä.

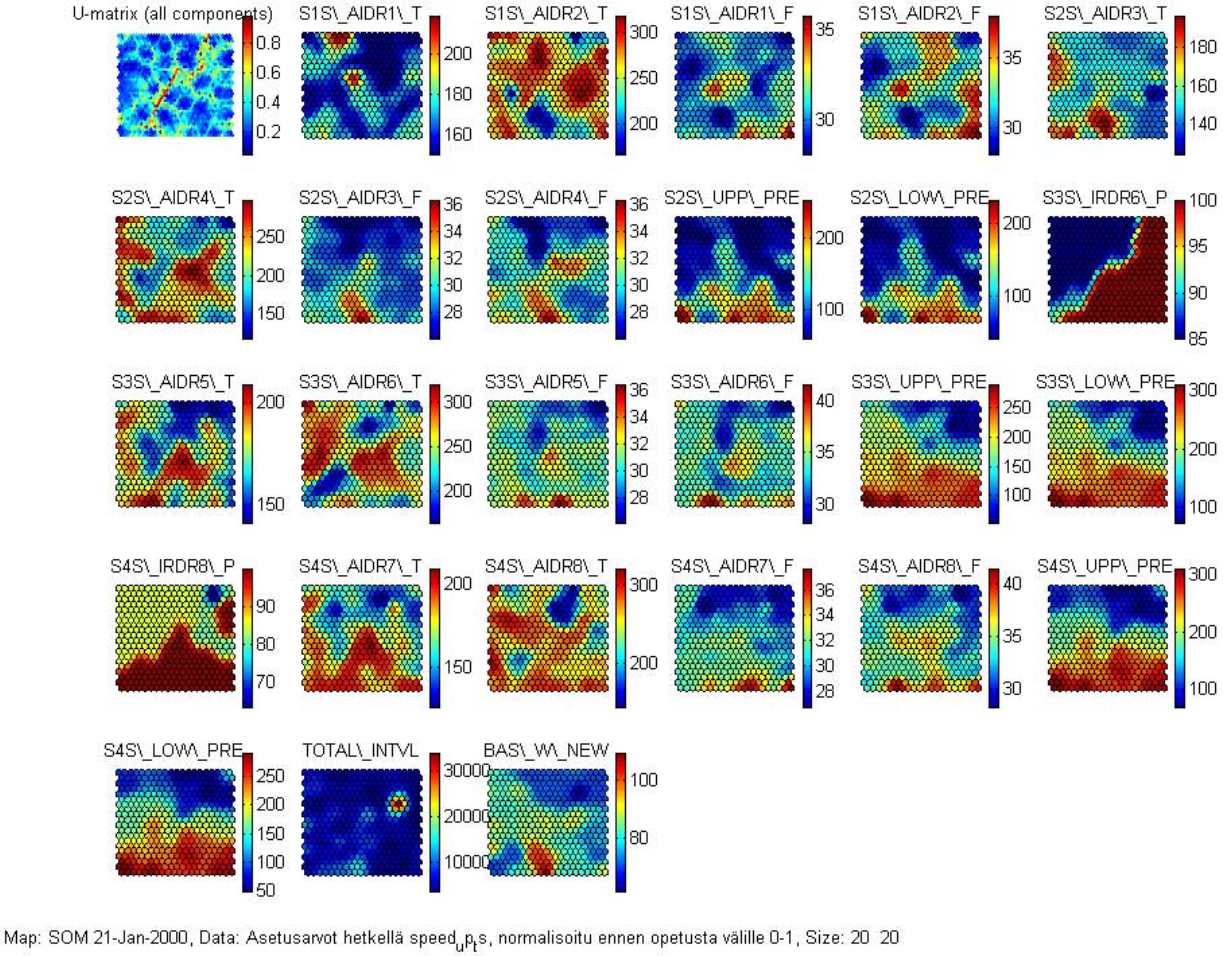


Kuva 5.4. 34 muuttujasta laskettu yksikköetäisyysmatriisi. Mitä punaisempi kuusikulmio on, sitä erilaisempi se on verrattuna lähinaapureihinsa.

U-matriisia käytetään ryhmien etsimiseen, jolloin siniset samankaltaiset prototyypit erottautuvat muista prototyypeistä punaisilla rajoilla. Tässä tapauksessa U-matriisi on värityksseltään niin tasainen, ettei selkeitä ryhmiä voida erottaa. Lisäksi hyvät ja huonot ylösajot jakautuvat heikosti näkyviin ryhmiin niin tasaisesti, ettei siitäkään voida päätellä mitään varmaa. Tämä viittaisi siihen, että tarkasteltavat muuttujat eivät joko riitä selittämään hylkymäärää tai hylkymäärän riippuvuus muuttujista on niin monimutkainen, ettei sitä pystytä SOM-kartasta löytämään.

5.2.3 Havaintoaineisto B

Tapaustutkimuksen toisessa vaiheessa käytettiin aineistona päällystyskoneen oikeita asetuksia, yhteensä 24 asetusmuuttujaa, jota kutsutaan havaintoaineistoksi B. Näillä muuttujilla havaintoja oli käytettävissä 143 kappaletta. SOM-analyysi tehtiin näiden 24 asetusmuuttujan ja 2 katkon ominaisuuden, katkon pituuden TOTAL_INTVL ja päällystettävän paperin neliöpainon BAS_W_NEW, perusteella. Muuttujien arvot normeerattiin välille 0-1, jotta ne saisivat analyysissä saman painoarvon.



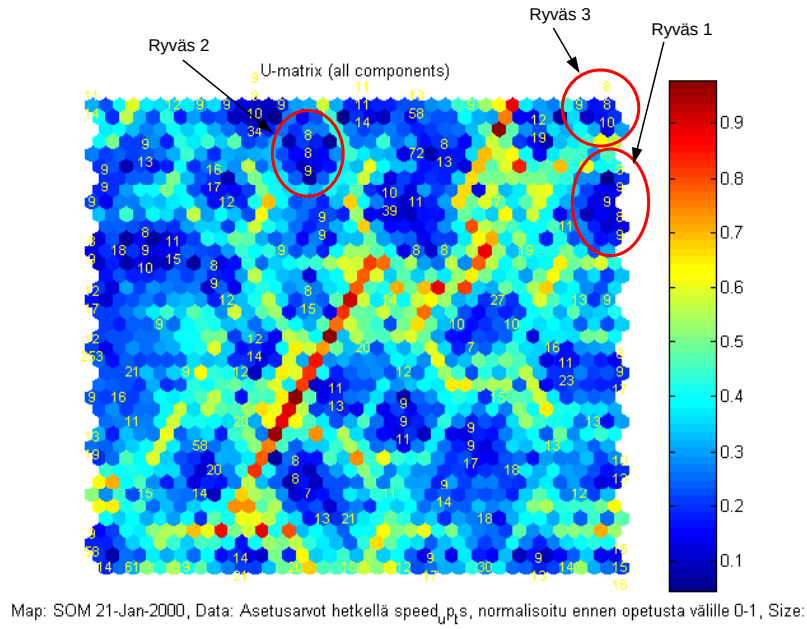
Kuva 5.5. Havaintoaineistolla B opetetun kartan komponenttitasot.

Aineistolla opetettu kartta päätyi kuvan (Kuva 5.5) mukaiseen tulokseen. Yhtä voimakkaita korrelaatioita kuin aineistolla A (Kuva 5.3) ei muuttujien kesken ole nähtävissä. Ainoastaan kuivausasemien paineiden SnS_UPP_PRE ja SnS_LOW_PRE keskenäiset korrelaatiot ovat selkeästi kartasta nähtävissä.

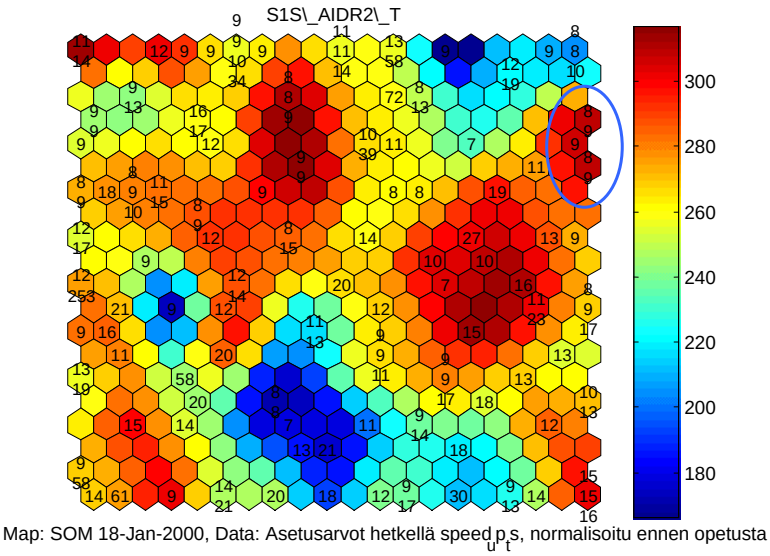
Piirretään tästäkin kartasta yksikköetäisyysmatriisi (Kuva 5.6) ja merkitään siihen ylösajojen BMU:t kunkin ylösajon tuottamalla hylkymäärällä. Tämä yksikkömatriisi on selvästi heterogeenisempi kuin aineistolla A saatu yksikkömatriisi (Kuva 5.4). Havaintoaineiston B matriisista voidaan erottaa muutama potentiaalinen vähäisen hylyn ryväs, jotka on ympyröity ja numeroitu kuvaan. Koska yhteen ryppääseen kuuluu maksimissaan 5 havaintoa, on tulos kuitenkin vain suuntaa-antava.

Komponenttitasoista (Kuva 5.5) voidaan lukea ryppäiden muuttujien arvot ryppään paikkaa vastaavalta kohdalta tasosta. Esimerkiksi kaikki ryppäät sijoittuvat komponenttitasossa BAS_W_NEW siniselle alueelle, joten ryppäisiin kuuluvien havaintojen neliöpainot ovat noin 70g/m^2 ja ryppään 1 muuttujan S1S_AIDR2_T arvot ovat hieman yli 300

(Kuva 5.7). Kaikkien kolmen ryppään muuttujien arvot on taulukoitu alla (Taulukko 5.14). Taulukon ryppäiden muuttujat on väritetty samalla värillä, mikäli muuttujan arvo on sama kaikilla ryppään havainnoilla.



Kuva 5.6. Havaintoaineisto B:n yksikköetäisyysmatriisi ja siitä löydetyt kolme vähäisen hylkymäärän ryvästä. Matriisiin merkityt numerot kertovat ko. prototyyppivektoriin liittyvien ylösajojen hylkymäärän kilometreissä.



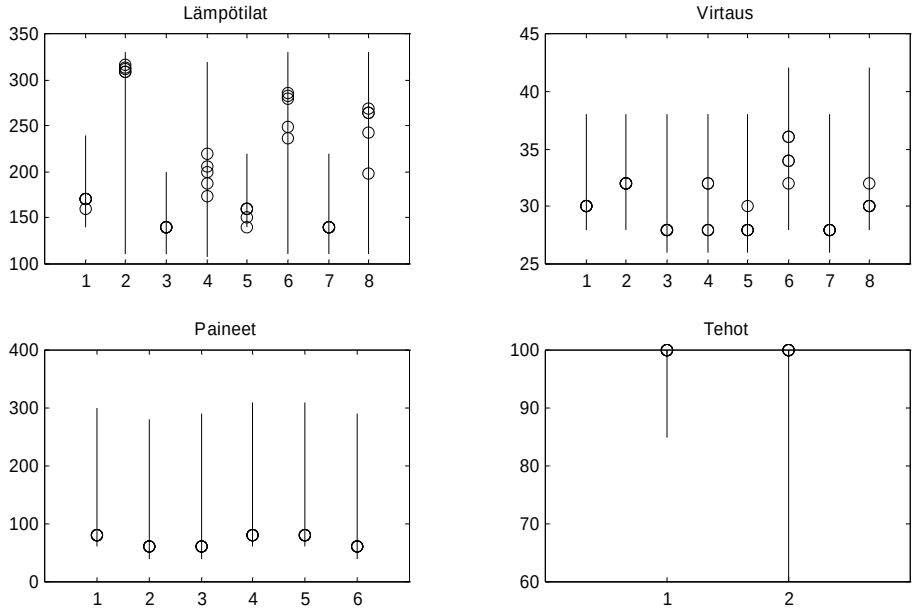
Kuva 5.7. Asetusmuuttujan S1S_AIDR2_T komponenttitaso. Kuvaan merkitty ylösajojen tuottamat hylkymäärät, kuten aikaisempaan yksikkömatriisiin (Kuva 5.6). Lisäksi ryväs 1 ympyröity.

Tarkastellaan tarkemmin ryvästä 1. Ryppään muuttujien arvojen sijoittuminen arvoavaruuteen nähdään piirtämällä ryppään havaintojen muuttujien arvot samaan kuvaan muuttujan vaihteluvälin kanssa (Kuva 5.8). Kuvasta nähdään, että myös ne muuttujien arvot, jotka eivät ole samoja, pysyvät lähellä toisiaan. Näin ollen ylösajon asetusarvoiksi voidaan suositella esimerkiksi ryppään parhaan ylösajon asetuksia. Tässä on kuitenkin muistettava, että ryppääseen kuuluu vain 5 havaintoa, joten asetusarvot ovat epävarmoja.

Ryppäiden asetusmuuttujat ja ylösajojen ominaisuudet

	Ryväs 1					Ryväs 2			Ryväs 3			
S1S_AIDR1_T	170	170	160	170	170	180	180	170	150	150	150	150
S1S_AIDR2_T	312	308	316	309	313	314	318	320	182	218	220	201
S2S_AIDR3_T	140	140	140	140	140	150	150	150	150	150	150	150
S2S_AIDR4_T	173	199	206	188	220	180	195	196	110	107	116	122
S3S_AIDR5_T	150	160	140	160	160	170	170	170	150	150	150	150
S3S_AIDR6_T	286	237	279	249	282	264	257	277	161	227	179	178
S4S_AIDR7_T	140	140	140	140	140	170	170	170	180	180	160	160
S4S_AIDR8_T	264	198	243	264	268	267	267	259	191	289	262	203
S1S_AIDR1_F	30	30	30	30	30	32	32	32	28	28	28	28
S1S_AIDR2_F	32	32	32	32	32	32	32	32	30	30	28	28
S2S_AIDR3_F	28	28	28	28	28	26	26	26	26	26	26	26
S2S_AIDR4_F	32	28	28	32	28	26	26	26	26	26	26	26
S3S_AIDR5_F	28	28	28	30	28	30	28	28	26	26	26	26
S3S_AIDR6_F	36	34	32	36	34	30	32	30	28	28	28	28
S4S_AIDR7_F	28	28	28	28	28	28	28	28	28	28	28	28
S4S_AIDR8_F	30	30	32	30	30	28	32	30	32	32	28	28
S2S_UPP_PRE	80	80	80	80	80	100	100	100	60	60	60	60
S2S_LOW_PRE	60	60	60	60	60	80	80	80	40	40	40	40
S3S_UPP_PRE	60	60	60	60	60	100	100	100	80	80	80	80
S3S_LOW_PRE	80	80	80	80	80	120	120	120	100	100	100	100
S4S_UPP_PRE	80	80	80	80	80	120	120	120	170	170	100	170
S4S_LOW_PRE	60	60	60	60	60	100	100	100	150	150	80	150
S3S_IRDR6_P	100	100	100	100	100	85	85	85	100	100	100	100
S4S_IRDR8_P	100	100	100	100	100	85	85	85	80	80	80	80
TOTAL_INTVL	2979	3113	4690	3110	3990	6735	3044	7923	5119	3620	4390	5180
BAS_W_NEW	71,34	71,26	71,10	71,27	70,97	70,58	71,89	71,71	70,81	71,44	71,73	71,16
BAS_W_OLD	72,23	71,16	71,59	70,35	71,97	71,96	71,91	71,78	71,27	72,01	71,36	71,77
WASTE_APP	7,84	8,47	8,62	8,81	9,28	8,17	8,48	8,63	7,96	7,96	8,62	9,51

Taulukko 5.14. Havaintoaineistolla B löydettyjen vähäisten hylkymäärien ryppäiden havaintojen muuttujien arvot. Muuttujien arvot on väritetty samalla värillä, mikäli muuttujalla on sama arvo kaikilla ryppään havainnoilla. Mikäli useammalla ryppäällä muuttujien arvot osuvat yksiin, on niillä kaikilla sama väri.



Kuva 5.8. Ryppään 1 havaintojen sijoittuminen havaintoavaruuteen. Ryppään havainnot on merkitty ympyrällä ja pystyviivat kertovat muuttujien vaihteluvälin. Mikäli yhden viivan kohdalla on alle 5 ympyrää, osuu useampi havainto samaan pisteeseen. Lämpötilat kuvassa ovat SxS_AIDRy_T:t, virtauksessa SxS_AIDRy_F:t, paineissa SxS_yyy_PRE:t ja tehoissa SxS_IRDRy_P:t.

5.2.4 Yhteenveto

SOM-menetelmän antama tulos hyvälle ylösajolle on hieman epäluotettava havaintojen vähäisen määrän takia. Kartasta löytyi hyviä ryppäitä, mutta vain vähän ja niihin kuului vain vähän havaintoja. Näin ollen menetelmää ei ainakaan tällä tasolla voi luotettavasti soveltaa ylösajojen optimointiin. Muiden menetelmien tulosten vertailuun SOM:in tuloksia sen sijaan kannattaa testata.

5.3 AutoClass-ryvästys

5.3.1 Yleistä AutoClass työkalusta

AutoClass on NASA:ssa kehitetty ohjaamattomaan oppimiseen perustuva luokittelujärjestelmä, joka pyrkii etsimään aineiston luonnolliset luokat käyttäen bayeslaista maksimitodennäköisyysperiaatetta. [CS96]

Ohjaamattomalla oppimisella tarkoitetaan prosessia, jossa opetusaineisto koostuu ominaisuusvektorista, mutta luokittelukriteeriä ei ole ennalta annettuna. Järjestelmän tehtävänä on tällöin etsiä sopivin luokkien lukumäärä ja näiden luokkien todennäköisimmät ominaisuusparametrit. Ohjatuissa oppimismenetelmissä oppimisaineisto on etukäteen luokiteltu annettuihin luokkiin ja tehtävänä on oppia luokitin, joka osaa ennustaa annetuilla kriteereillä uusien näytteiden luokan.

AutoClass tukee vektorimuotoisen mittaustiedon luokittelua. Sekä diskreettiarvoisia että jatkuva-arvoisia muuttujia voidaan käyttää mallinnuksessa. Muuttujat voidaan mallintaa toisistaan riippumattomiksi tai niiden keskinäinen korrelaatio voidaan huomioida. Diskreettejä muuttujia voidaan mallintaa multinominaalisella jakaumalla. Jatkuva-arvoisia normaalijakaumilla sekä riippumattomina että yhteisriippuvina.

Tapauksia ei allokoida suoraan luokkiin, vaan niille määritellään todennäköisyysperustainen luokkaan kuulumisaste. Tietoa mallinnetaan ehdollisesti riippumattomien luokkien yhdistelminä. Luokat määritellään todennäköisyysjakaumina attribuuttien metaavaruuden suhteen. Jatkuva-arvoisten muuttujien jakaumia mallinnetaan normaalijakaumilla ja diskreettejä Bernoulli-jakaumilla. AutoClass etsii sellaisen joukon luokkia, joka on mahdollisimman todennäköinen dataan ja määriteltyyn malliin nähden. Tuloksena saadaan luokkakuvaukset ja esimerkkien jäsenyysasteet luokissa.

AutoClass hyödyntää suhteellisen yksinkertaisia tilastollisia malleja. Siinä käytetään äärellisiä sekajakaumamalleja (finite mixture distribution model), joissa toisistaan riippumattomat attribuutit riippuvat latenteista luokista. Tällöin attribuuttien yhteistodennäköisyysjakauma voidaan esittää seuraavasti:

$$P(X_1,..., X_n) = \Sigma_y P(X_1,..., X_n \mid Y=y) P(Y=y)$$

$$= \Sigma_y P(Y=y) \Pi_i P(X_i \mid Y=y)$$

AutoClass työkalua käytettäessä käyttäjä määrittelee attribuuttien tyypin (multinominaalinen, normaalijakautunut vai alhaalta rajoitettu normaalijakautunut) ja mittausvirheen. Useampien vaihtoehtoisten mallien määrittely on mahdollista. Ratkaisun etsiminen perustuu 2-vaiheiseen EM-tyyppiseen (Expectation-Maximization) hakuun, jossa mallinhakuvaiheessa etsitään optimaalista luokkien lukumäärää (J) ja parhaita luokkamalleja (T_j). Parametriarvojen hakuvaiheessa haetaan MAP-mielessä (Maximum Posterior probability) parhaita parametreja luokkien mallin rakenteen ollessa kiinnitettynä.

AutoClass luokitinta on menestyksellisesti sovellettu astronomisten mittausten, DNA tiedon ja kaukokartoitustiedon luokitteluun.

5.3.2 AutoClass ryvästyksestä Aineistolla A

Sovelsimme AutoClass-ohjelmaa pääkomponenttianalyysillä (ks. tämän raportin luku 4.2) muodostettujen 9 pääkomponentin virittämän mittausavaruuden analysointiin. Mukaan otettiin lisäksi Waste_APP (laatukriteeri) ja DELTA_BA (neliöpainon muutos). Todettakoon, että tässä raportoiduissa tuloksissa on käytetty alunperin kerättyjä edellisen, katkenneen ajon parametreja seuraavan ylösajon ohjausparametreja mallintamassa. Oletuksena on ollut että edellisen ajon parametreja käytetään automaattisesti seuraavan ajon parametreina.

AutoClass työkalun soveltaminen edellyttää mittausaineiston muuntamisen attribuuttivektorityyppiseksi ja sen jakamisen opetus- ja testiaineistoon. Muuttujien tyypit tulee määritellä ja työkalulle tulee määritellä, millaisia oletuksia jakaumista tehdään (riippumattomat muuttujat ja keskenään korreloivat muuttujajoukot). Ohjelma hakee todennäköisyysperiaatteella lupaavimman määrän luokkia (käyttäjä antaa ohjeita luokkien määristä) ja tallettaa haun edetessä aina lupaavimman luokan talteen.

Tutkittaessa koko aineistoa AutoClass muodosti 6 luokkaa, joissa merkittävimmiksi erottaviksi kriteereiksi muodostuivat 2. pääkomponentti, neliöpainon muutos, 4. pääkomponentti, 8. pääkomponentti ja 3. pääkomponentti, 6. pääkomponentti ja vasta sitten arvioitu hylkymäärä. Loput kriteerit olivat jo varsin vähämerkityksisiä.

```
CLASS  0 - weight  65   normalized weight 0.341   relative strength 5.87e-004 *******
                        class cross entropy w.r.t. global class 1.54e+000 *******

   Model file:   D:\Tools\AutoclassC\data\tempat3\alku3.model  -  index = 0
   Numbers: numb/t = model term number; numb/a = attribute number
   Model term types (mtt): (single_normal_cn SNcn)


REAL ATTRIBUTE  (t = R)                                        |Mean-jk -
 numb t mtt   description           I-jk      Mean     StDev    Mean-*k|/     Mean      StDev
 t  a                                        -jk      -jk       StDev-jk     -*k       -*k

01 01 R SNcn  DELTA_BA ...........  0.386 (+5.59e-01 +2.88e+00) +2.50e-02 (+4.87e-01 +6.28e+00)
04 04 R SNcn  FAC2_1 .............  0.376 (+1.71e-01 +4.91e-01) +3.83e-01 (-1.67e-02 +1.04e+00)
02 13 R SNcn  Log S4_BASIS .......  0.283 (+3.56e+00 +2.90e-01) +8.38e-01 (+3.32e+00 +3.33e-01)
06 06 R SNcn  FAC4_1 .............  0.187 (+5.86e-01 +1.11e+00) +5.55e-01 (-2.94e-02 +1.03e+00)
07 07 R SNcn  FAC5_1 .............  0.093 (-2.87e-01 +8.06e-01) +3.54e-01 (-1.71e-03 +1.04e+00)
00 12 R SNcn  Log WASTE_AP .......  0.092 (+1.83e+00 +8.10e-01) +3.84e-01 (+1.52e+00 +7.52e-01)
05 05 R SNcn  FAC3_1 .............  0.041 (+2.11e-01 +9.08e-01) +2.79e-01 (-4.28e-02 +1.00e+00)
09 09 R SNcn  FAC7_1 .............  0.037 (+1.91e-01 +1.14e+00) +1.88e-01 (-2.32e-02 +1.01e+00)
03 03 R SNcn  FAC1_1 .............  0.029 (-2.12e-01 +1.11e+00) +1.68e-01 (-2.56e-02 +1.00e+00)
11 11 R SNcn  FAC9_1 .............  0.010 (-1.11e-01 +7.16e-01) +4.97e-02 (-7.53e-02 +6.56e-01)
08 08 R SNcn  FAC6_1 .............  0.009 (+1.28e-01 +9.81e-01) +1.30e-01 (+2.54e-05 +1.01e+00)
10 10 R SNcn  FAC8_1 .............  0.003 (-1.12e-01 +1.00e+00) +7.86e-02 (-3.37e-02 +1.00e+00)
```

Kuva 5.9. Esimerkki AutoClass:n löytämästä luokkamäärityksestä.

Oheisessa kuvassa on tulostettuna luokkaa 0 (luokista 0-5) vastaavat ominaisuudet. Tulostuksesta nähdään esimerkiksi, että DELTA_BA on merkittävin attribuutti ja sen keskiarvo on lievästi positiivinen (~0.56, koko aineistossa ~0,49).
Tutkittaessa luokkia tarkemmin havaittiin, että luokkarajojen välillä ei voida havaita selväpiirteisiä rajoja yksittäisten attribuuttiarvojen suhteen, vaan rajat ovat todennäköisyysperustaisia. Tulosraporteissa on kuitenkin raportoitu kunkin parametrin suhteen keskiarvo ja hajonta kyseisessä luokassa ja toisaalta koko aineistossa.

5.3.3 Yhteenveto

Tällä analyysillä pystytään tunnistamaan samantyyppisiä luokkia, mistä voi olla apua kohtuullisen tuntemattoman aineiston ominaisuuksia tutkittaessa. Mutta koska hylkymäärä ei valikoitunut merkittäväksi luokituskriteeriksi, ei ainakaan tällä aineistolla syntynyt suoraan pienihylkyistä luokkaa. Eräs hyödyntämistapa saattaisi kuitenkin olla, että luokitellaan aineisto tunnistettuihin klustereihin ja tutkitaan kuhunkin klusteriin luokittuvia tapauksia tarkemmin. Tätä ei ole kuitenkaan kokeiltu.

Eräs vaihtoehto jatkoanalyysiksi olisi tutkia syntyneiden luokkien jäseniä muilla ohjatun oppimisen menetelmillä ja pyrkiä tunnistamaan pienen hylkymäärän selittäjiä. AutoClass ohjelmalla tunnistettujen luokkien sisällä. Tätä ei ole kuitenkaan tehty.

Tätä menetelmää ei sovellettu uuteen aineistoon B.

6 Luokittelu

Luokittelun (classification) tavoitteena on jakaa aineisto luokkiin ennalta määriteltyjen päättelysääntöjen avulla. Luokittelussa on yleensä kaksi vaihetta: päättelysääntöjen määrittely ja luokittelu sääntöjen perusteella. Sääntöjen määrittely voidaan tehdä automaattisesti oppimalla datasta.

6.1 Päätöspuut

6.1.1 Päätöspuun periaate

Päätöspuu on kaavioesitys, jolla visualisoidaan luokittelussa käytettäviä päättelysääntöjä. Puuta muodostettaessa määritellään, minkä muuttujan arvoja luokitellaan. Päätöspuu esittää sellaiset muuttujat ja näiden muuttujien arvoalueet, joiden avulla luokiteltavan muuttujan arvot voidaan parhaiten erottaa toisistaan. Päätöspuun tuloksena saadaan joukko päättelysääntöjä, joita voidaan käyttää uuden aineiston luokittelussa.

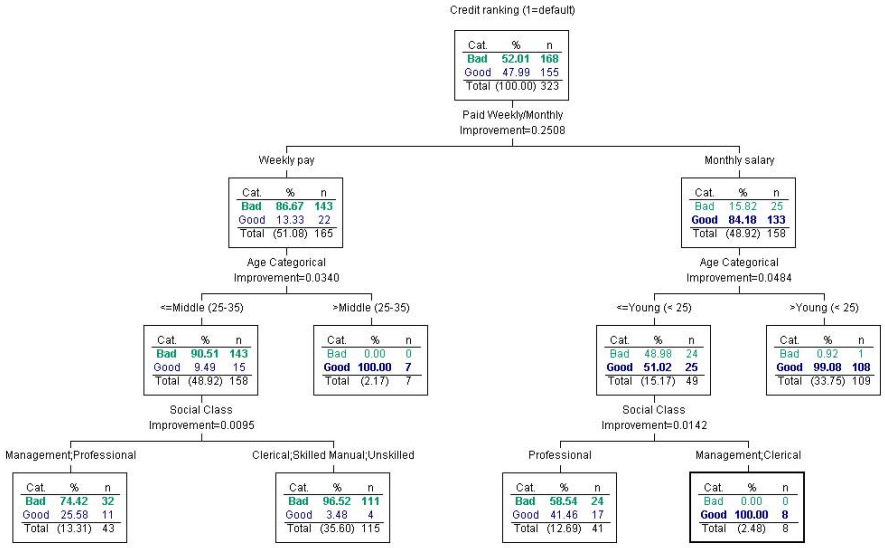
Päätöspuu alkaa juurisolmusta (root node), joka sisältää kaikki havainnot käsiteltävästä aineistosta. Juurisolmun muuttujaa kutsutaan analyysin kohdemuuttujaksi (target). Puuta alaspäin tarkasteltaessa havainnot jakautuvat täydellisesti poissulkeviin joukkoihin (mutually exclusive subset). Valitusta puunkasvatusalgoritmista riippuen jako voidaan tehdä kahteen tai useampaan haaraan. Jokaiseen jakautumiskohtaan liittyy muuttuja, jonka perusteella jako on tehty sekä muuttujan arvoalue kullekin haaralle. Puun alimpia solmuja kutsutaan päätössolmuiksi (terminal node). Päätössolmut kuvaavat, miten aineisto jakautuu luokkiin soveltamalla sääntöjä juurisolmun ja jokaisen päätössolmun välillä. Jos puu on täydellinen, päätössolmut kuvaavat aineiston yksikäsitteisiin luokkiin. Käytännössä puu ei yleensä ole täydellinen. Yksittäiset päätössolmut voivat kuvata aineiston useampaan eri luokkaan. Puun luokitteluvirhe voidaan kuvata esimerkiksi prosentuaalisena riskiestimaattina (risk estimate), joka kertoo, mikä osa aineistosta luokitellaan väärään luokkaan.

Kaikkien menetelmien avulla saaduista puista on mahdollista laskea puun luokitteluvirhe. Luokitteluvirhe lasketaan joko jakamalla aineisto opetus- ja validointijoukkoon tai ristiinvalidoinnin (cross validation) avulla. Ristiinvalidoinnissa aineisto jaetaan haluttuun määrään osia, esimerkiksi kymmeneen. Kun validointi suoritetaan, aineistosta rakennetaan tässä tapauksessa 10 eri päätöspuuta siten, että jokainen kymmenys käytetään vuorollaan tarkistusjoukkona. Luokitteluvirhe lasketaan eri validoinneilla saatujen virheiden keskiarvona. Ristiinvalidointi antaa paremman kuvan puun luokitteluvirheestä kuin jako opetus- ja validointijoukkoon, mutta se on laskennallisesti vaativampi.

Kuva 6.10 on esimerkki päätöspuusta, joka on laadittu SPSS Answer tree-ohjelmalla [SPSS98]. Puun solmut kertovat, miten kohdemuuttujan arvot jakautuvat kussakin solmussa luokkiin. Puun haarat kuvaavat muuttujia ja arvoja, joiden perusteella jako tehdään. Arvot voivat olla joko kategoristen muuttujien arvoja tai jatkuvien muuttujien tapauksessa arvoalueita. Haaran yhteydessä esitetään improvement-arvo, joka kertoo

kuinka paljon malli paranee kunkin haaran kohdalla. Esimerkkipuu on laadittu menetelmällä, joka jakaa aineiston aina kahteen luokkaan.

Kuvan esimerkkipuu on SPSS-ohjelman mukana tullut esimerkki ja siinä analysoidaan tekijöitä, jotka vaikuttavat henkilön luottokelpoisuuteen. Esimerkin kohdemuuttujana on asiakkaan luottokelpoisuusluokka, asiakkaat jakautuvat hyviin ja huonoihin. Kaikkein voimakkaimmin jakavaksi tekijäksi havaitaan, maksetaanko henkilölle palkka kuukausittain vai viikoittain. Seuraavaksi parhaaksi jakajaksi on molemmassa haarassa havaittu henkilön ikäluokka. Päätössolmuja tarkastelemalla havaitaan, että puu ei ole täydellinen, sillä päätössolmut eivät kaikissa tilanteessa jaa dataa yksikäsitteisiin luokkiin. Päätöspuun luokitteluvirheeksi on laskettu n. 13%. Tämä tarkoittaa sitä, millä todennäköisyydellä puun avulla muodostettavat luokittelusäännöt luokittelevat uuden kohdeaineiston virheelliseen luokkaan.



Kuva 6.10. Esimerkki päätöspuusta.

Päätöspuiden laskemiseen on kehitetty lukuisia algoritmeja. Algoritmit eroavat siinä, millä menetelmällä jako tehdään, millaisia muuttujia päätöspuissa voi käyttää ja millä tavalla algoritmit ottavat huomioon puuttuvia arvoja. Tässä projektissa on hyödynnetty kahta algoritmia, C&RT (classification and regression trees) [BFOS84] ja exhaustive CHAID (chi-squared automatic interaction detector) [BVS91]. Seuraavassa esitetään kummankin algoritmin tärkeimmät ominaisuudet. Lisätietoja algoritmeista on lähteessä [SPSS98, s. 190-195].

C&RT algoritmi jakaa puun aina kahteen haaraan. Jakava muuttuja valitaan siten, että jokainen lapsisolmu on puhtaampi (pure) kuin sen äitisolmu. Jotta solmu olisi täysin puhdas, kaikki solmuun kuuluvan kohdemuuttujan arvot ovat samat. Käytännössä puhtaus määritellään epäpuhtausmitan (impurity measure) avulla. Käytettävä epäpuhtausmitta riippuu siitä, onko kohdemuuttuja diskreetti vai jatkuva.

CHAID-algoritmissa puu jaetaan kahteen tai useampaan haaraan. Haarauttava muuttuja valitaan tilastollisen testin perusteella siten, että muuttuja on mahdollisimman merkittävä (significant) suhteessa kohdemuuttujaan. Käytettävä testi riippuu muuttujan mitta-asteikosta. exhaustive CHAID –algoritmi on muunnelma CHAID-algoritmista ja se antaa paremman tuloksen mutta vaatii enemmän laskenta-aikaa.

6.1.2 Aineistot

Havaintoaineiston A analyysiä varten tietokannasta poimitaan 192 katkoa. Aineistoon sisältyvät ennen katkoa voimassa olleet mittausmuuttujien arvot (poimittu aikaleiman BREAK_TS mukaan) ja katkon jälkeiseen ylösajoon liittyvät lasketut muuttujat DELTA_BASIS_WEIGTH, BASIS_WEIGTH_NEW, BASIS_WEIGTH_OLD ja TOTAL_INTERVAL. Laskettujen muuttujien arvot on poimittu aikaleiman RUNUP_END mukaan. Käytössä olleen SPSS-ohjelmiston rajoituksista johtuen muuttujanimistä on jouduttu käyttämään 8 merkkiin lyhennettyjä versioita.

Havaintoaineiston B analyysiä varten tietokannasta poimitaan 143 katkoa. Aineisto sisältää kaikki asetusmuuttujat ja lasketut muuttujat BAS_W_NEW ja TOTAL_INTVL. Asetusmuuttujien arvot poimitaan aikasarjasta aikaleiman SPEED_UP_TS mukaan ja lasketut muuttujat aikaleiman BREAK_TS mukaan.

Kaikkien seuraavassa kuvattujen analyysien kohdemuuttujana on laskettu muuttuja WASTE_DI. Muuttuja muodostetaan jakamalla muuttuja WASTE_APP luokkiin. Luokat määritellään aineiston perusteella siten, että kuhunkin luokkaan saadaan tietty osuus havainnoista. Käytetyt luokat ja niiden määrittelyperusteet on määritelty erikseen havaintoaineistolle A (Taulukko 6.1) ja havaintoaineistolle B (Taulukko 6.2).

Luokka	% havainnoista	waste_app
1	0-20%	< 8.5
2	20-50%	8.5 – 10
3	50-90%	10 – 35
4	90-100%	> 35

Taulukko 6.1. Muuttujan WASTE_APP diskeretoinnin rajat havaintoaineistolle A.

Luokka	% havainnoista	waste_app
1	0-45%	< 10.73
2	45-55%	10.73 – 12.18
3	55-100%	> 12.18

Taulukko 6.2. Muuttujan WASTE_APP diskeretoinnin rajat havaintoaineistolle B.

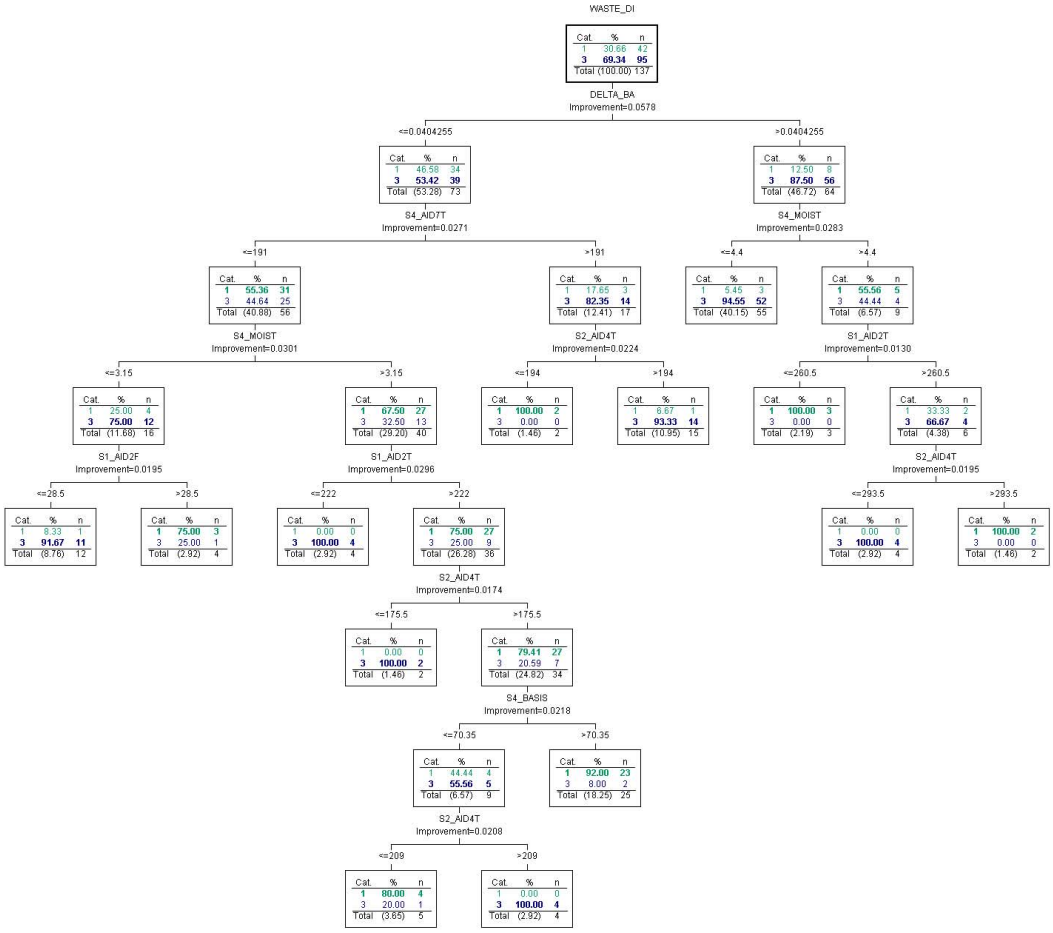
6.1.3 Analyysit havaintoaineistosta A

Kaikissa havaintoaineiston A analyyseissä kohdemuuttujana on muuttujan WASTE_APP diskretoitu muoto WASTE_DI. Analyysiä varten aineistosta poimitaan muuttujan WASTE_DI luokkien 1 ja 3 edustajat siten, että luokka 1 edustaa hyviä ja luokka 3 huonoja ylösajoja. Poiminta tehdään sen takia, että luokkien välille saadaan mahdollisimman suuret erot. Lisäksi luokan 4 aineisto sisältää runsaasti outliereitä (erittäin suuria arvoja, ilmeisesti huoltoseisokkeja).

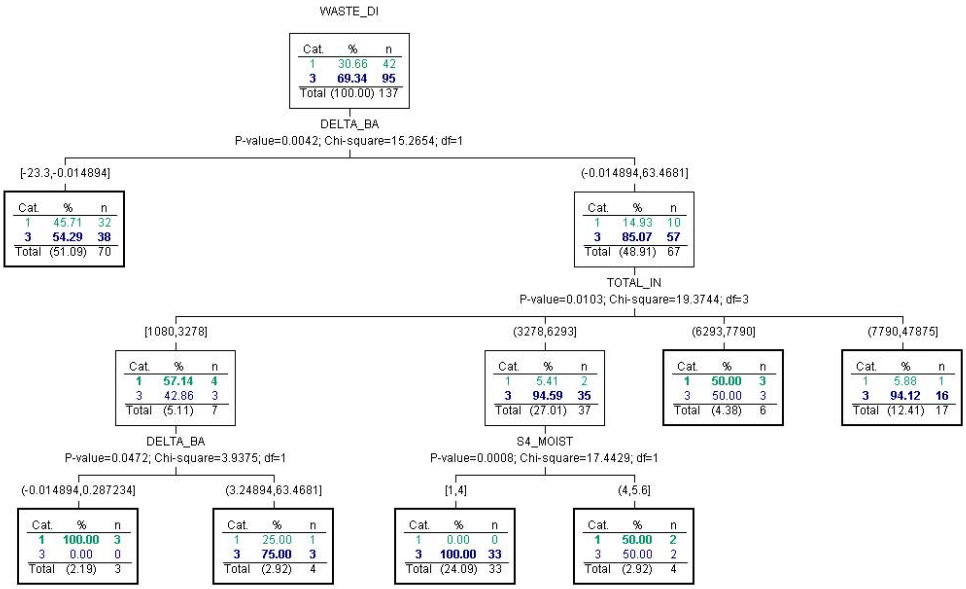
Analyyseissä 1.1 ja 1.2 käytetään puun kasvatuksessa lähdemuuttujina sekä mittausmuuttujia että laskettuja muuttujia DELTA_BASIS_WEIGHT, VELOCITY ja BASIS_WEIGHT_NEW. Käytetyn analyysiaineiston koko on 137 katkoa.

Analyysissä 1.1 käytetään C&RT algoritmia Gini-indeksillä. Puun haaran kokorajoitus on 5 aineistoriviä äitisolmussa ja 2 aineistoriviä lapsisolmussa ja puun maksimikorkeus (syvyysrajoite) on 10 solmua. Analyysin 1.1 luokitteluvirhe on 40% keskihajonnalla 4% (Kuva 6.11). Analyysi 1.2 tehtiin samalle aineistolle CHAID-menetelmällä. Puun rajoitteet olivat samat kuin analyysissä 1.1. Analyysin 1.2 luokitteluvirhe on 39% keskihajonnalla 4% (Kuva 6.12).

Molemmissa analyyseissä 1.1 ja 1.2 havaitaan, että tärkein jakaja on neliöpainon muutos, muuttuja DELTA_BASIS_WEIGTH. Jako tapahtuu likimain muuttujan nollakohdassa. Puuta eteenpäin tarkastellessa voidaan löytää muutamia lehtisolmuja, joilla hyvät ylösajot pystytään yksikäsitteisesti tunnistamaan. Koska neliöpainon muutos ei ole vapaa parametri (se on annettu ylösajon yhteydessä eikä sitä voida säätää), puun luokittelutarkkuus on hyvien ylösajojen suhteen heikko. Tämän takia päätettiin analysoida aineistoa ilman laskettuja muuttujia DELTA_BASIS_WEIGHT, VELOCITY ja BASIS_WEIGHT_NEW.



Kuva 6.11. Analyysin 1.1 tulospuu.



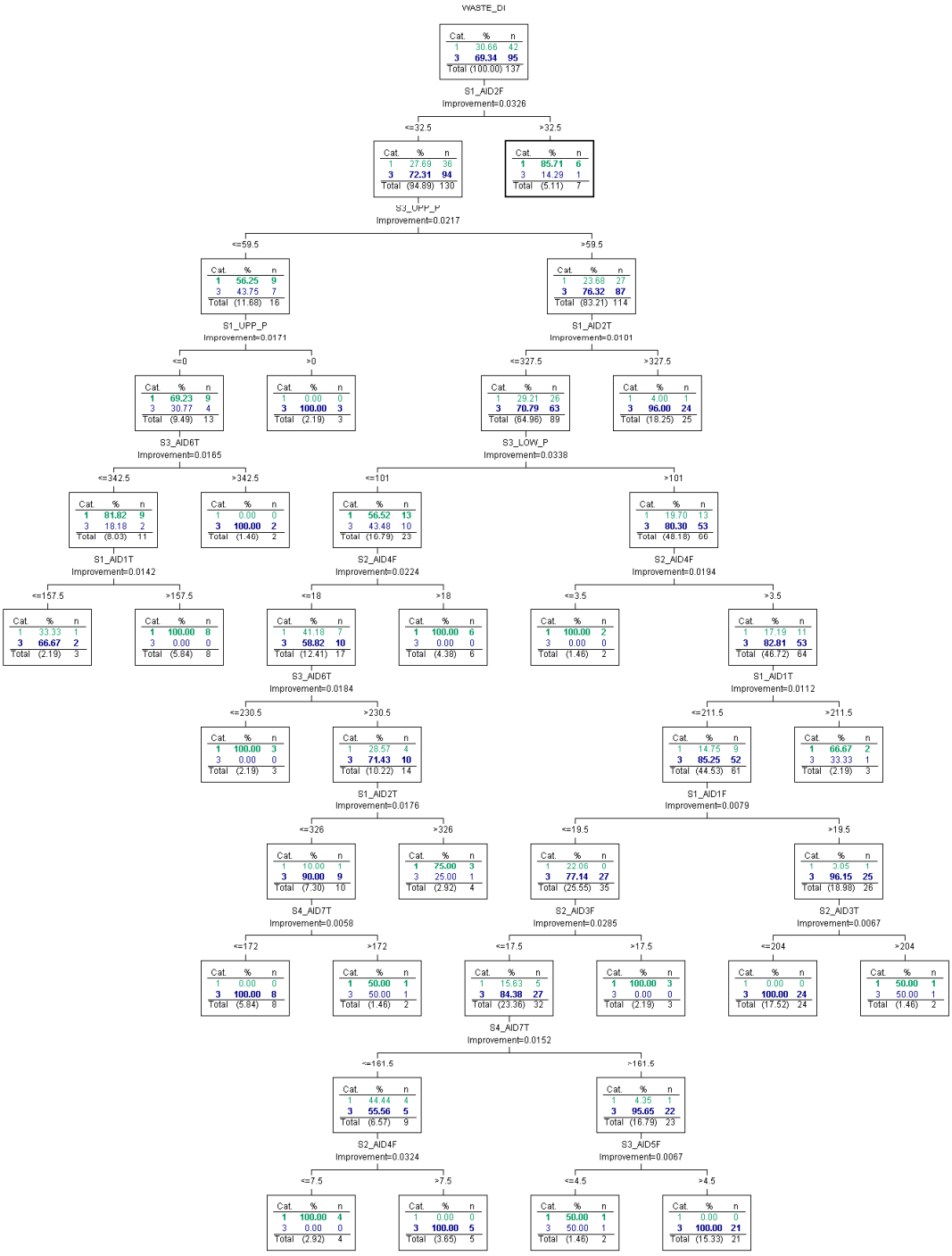
Kuva 6.12. Analyysin 1.2 tulospuu.

Analyyseissä 1.3 ja 1.4 käytetään puun kasvatuksessa lähdemuuttujina mittausmuuttujia kohdemuuttujan ollessa edelleen muuttuja WASTE_DI. Käytetyn analyysiaineiston koko on 137 katkoa.

Analyysissä 1.3 käytetään C&RT algoritmia Gini-indeksillä. Puun haaran kokorajoitus on 5 aineistoriviä äitisolmussa ja 2 aineistoriviä lapsisolmussa ja puun maksimikorkeus on 10 solmua. Analyysin 1.3 luokitteluvirhe on 33% keskihajonnalla 4% (

Kuva 6.13). Analyysi 1.4 tehtiin samalle aineistolle CHAID-menetelmällä. Puun rajoitteet olivat samat kuin analyysissä 1.3. Analyysin 1.4 tulospuu on kutistunut juurisolmuun, eli CHAID-menetelmä ei pystynyt löytämään yhtään sellaista muuttujaa, joka luokittelisi aineiston muuttujat WASTE_DI perusteella. Tämä johtuu menetelmän laskentatapojen eroista: CHAID-menetelmässä käytetään tilastollista testausta kun taas C&RT –menetelmässä informaatioteoreettista mittaa, joka on robustimpi häiriöille.

Analyysissä 1.3 merkittävin jakaja on muuttuja S1_AID2F (ensimmäisen aseman leijukuivaimen puhallusnopeus). Analyysissä löydetään 6 kappaletta lehtisolmuja, joilla hyvät ylösajot voidaan tunnistaa. Tunnistamisen kriteerinä on, että lehtisolmussa on luokkaan 1 kuuluvia alkioita vähintään 3 kpl ja luokkaan 3 kuuluvia alkioita korkeintaan 1 kpl.

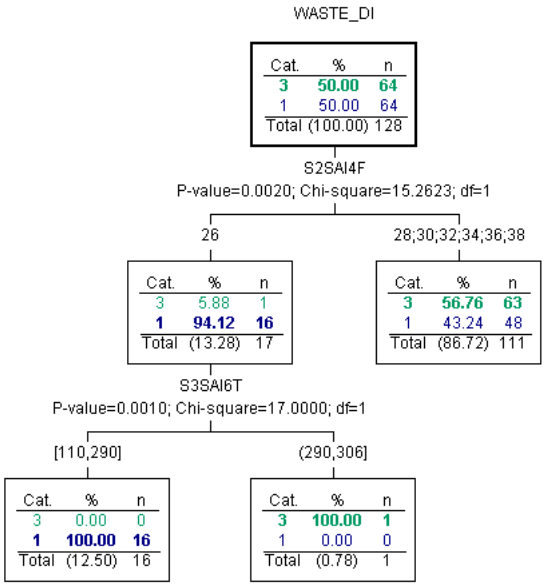


Kuva 6.*13. Analyysin 1.3 tulospuu.*

6.1.4 Analyysit havaintoaineistosta B

Havaintoaineiston B analyysissä kohdemuuttujana on muuttujan WASTE_APP diskretoitu versio WASTE_DI. Analyysiin valitaan kohdemuuttujan luokkien 1 ja 3 edustajat. Luokka 2 muodostaa 10% aineistosta ja se jätetään valitsematta, jotta luokkien 1 ja 3 välille saataisiin riittävät erot. Puun kasvatuksen lähdemuuttujina asetusmuuttujia. Lasketut muuttujat BAS_W_NEW ja TOTAL_INTVL jätetään mallinmuodostuksen ulkopuolelle. Analyyseissä käytetyn aineiston koko on 71 katkoa.

Analyysissä 2.1 käytetään C&RT algoritmia Gini-indeksillä. Puun haaran kokorajoitus on 1 aineistorivi äitisolmussa ja lapsisolmussa. Puun maksimikorkeus (syvyysrajoite) on 10 solmua. Analyysin 2.1 luokitteluvirhe on 41% keskihajonnalla 4%. Tämän analyysin tulospuuta ei esitetä puun laajuuden vuoksi, myöhemmin puun tuloskset esitetään sääntöjen avulla. Analyysi 2.2 tehtiin samoilla rajoitteilla kuin analyysi 2.1 mutta puun kasvatuksessa käytettiin CHAID-menetelmää. Analyysin 2.2 luokitteluvirhe on 44% (Kuva 6.14).



Kuva 6.14. Analyysin 2.2 tulospuu.

Analyysissä 2.1 merkittävin jakaja on muuttuja S4S_AIDR4_F (neljännen aseman leijukuivaimen ilmamäärä). Analyysissä 2.2 merkittävin jakaja on muuttuja S2S_AIDR4_F (toisen aseman leijukuivaimen ilmamäärä). Tämä tukee analyysin 2.1 tuloksia. Koska analyysin 2.1 luokittelukyky on parempi ja puun koko suurempi kuin analyysissä 2.2 päätettiin käyttää jatkotyössä analyysin 2.1 tuloksia.

Päätöspuuanalyysin tuloksena voidaan puun lehtisolmujen perusteella generoida säännöt, joiden avulla uusi aineisto voidaan luokitella. Perusoletuksena säännöt generoidaan kaikille lehtisolmuille, mutta tässä projektissa päätettiin ottaa vain ne lehtisolmut, joissa on luokkaan 1 kuuluvia opetusaineiston rivejä vähintään 3 kpl ja luokkaan 3 kuuluvia opetusaineiston rivejä 0 kappaletta. Analyysissä löydetään 5 kpl edelliset kriteerit täyttäviä lehtisolmuja.

Lehtisolmujen perusteella muodostetaan säännöt luokittelijan mielestä hyville ylösajoille (Taulukko 6.3). Käytössä olleen SPSS-ohjelman rajoitteiden takia säännöissä on jouduttu käyttämään muuttujanimistä 8 merkkiin lyhennettyjä versioita. Taulukon säännöt ovat toisensa poissulkevia, yhteen luokiteltavan uuden havaintoaineiston riviin voidaan soveltaa korkeintaan yhtä sääntöä. Jokainen sääntö koostuu ehto-osasta ja toiminto-osasta. Ehto-osassa määritellään luokiteltavan aineiston muuttujien arvorajat, joille kyseistä sääntöä sovelletaan. Toiminto-osan kohta node on viittaus alkuperäisen päätöspuun lehtisolmuun, jonka perusteella sääntö on generoitu. Kohta prediction kertoo, mihin luokkaan sääntö aineiston luokittelee. Kohta probability kuvaa säännön luokitteluvirhettä. Eräissä tapauksissa lehtisolmun perusteella voidaan luokitella aineistorivi useisiin eri luokkiin. Sääntöjen perusteella saatujen tulosten on kuitenkin oltava yksikäsitteisiä. Useiden mahdollisten luokkien tilanteessa valitaan se luokka, jota lehtisolmussa esiintyy eniten ja luokitteluvirhe kertoo, mikä osuus valitun luokan arvoista on kaikista mahdollista arvoista. Tässä analyysissä luokitteluvirhe on 1, eli luokittelu on täydellinen. Tämä johtuu sääntöjen generoinnille asetetuista ehdoista (ks. edellinen kappale).

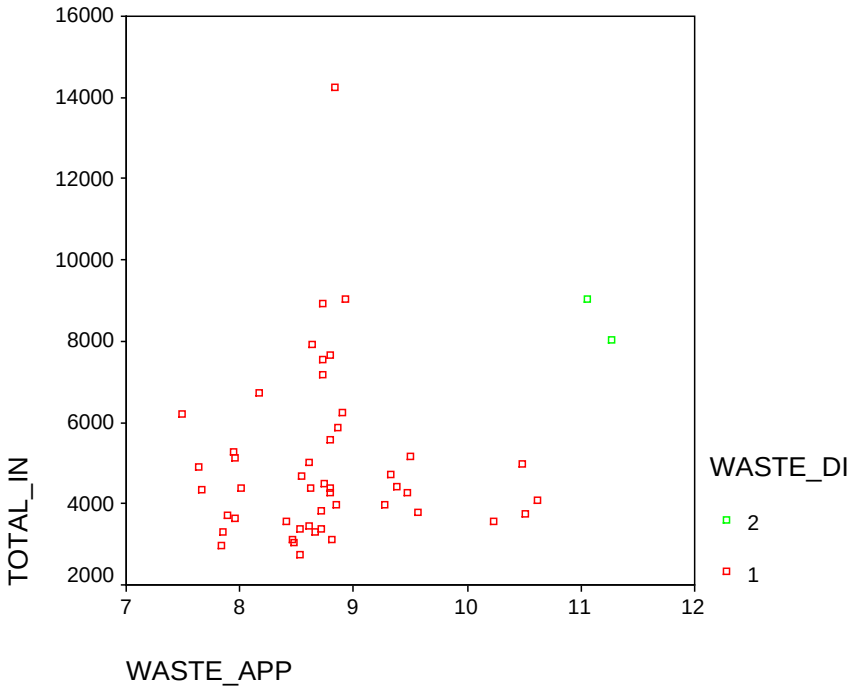
Analyysiin 2.1 perustuvilla säännöillä pystytään löytämään 78% hyvistä ylösajoista kun sääntöjä sovelletaan alkuperäiseen aineistoon. Seuraavassa tutkitaan tarkemmin sääntöjen soveltamisen onnistumista. Kuva 6.15 esittää, miten luokkaan 1 luokitellut ylösajot (WASTE_APP < 10.73 km) jakautuvat muuttujien WASTE_APP ja TOTAL_INTERVAL suhteen. Luokittelijan luokitteluvirhe on pieni, sillä luokkaan 1 on luokiteltu ainoastaan 2 kpl luokkaan todella kuulumattomia arvoja (muuttuja WASTE_DI). Kuvasta havaitaan, että 9 km pienemmille hylkymäärän arvoille löytyy runsaasti hyviksi luokiteltuja havaintoja laajalla katkon pituusalueella. Alle 8 km hylkymäärän arvoille ei löydy havaintoja, kun katkon pituus ylittää 5000 minuuttia.

Kuva 6.16 esittää vastaavaa tilannetta muuttujien WASTE_APP ja TOTAL_INTERVAL suhteen. Kuvasta voidaan havaita, että paperilaaduille 70 g/m^2 ja 80 g/m^2 on havaintoja suhteellisen laajalle syntyneen hylyn määrän arvoalueelle. Tuloksia arvioidaan vielä molempien ulkoisten muuttujien suhteen (Kuva 6.17). Kuvasta voidaan todeta sama kuin edellisestäkin kuvasta ja lisäksi se, että havaintoja on paperilaatukohtaisesti suhteellisen laajalle katkon pituusalueelle.

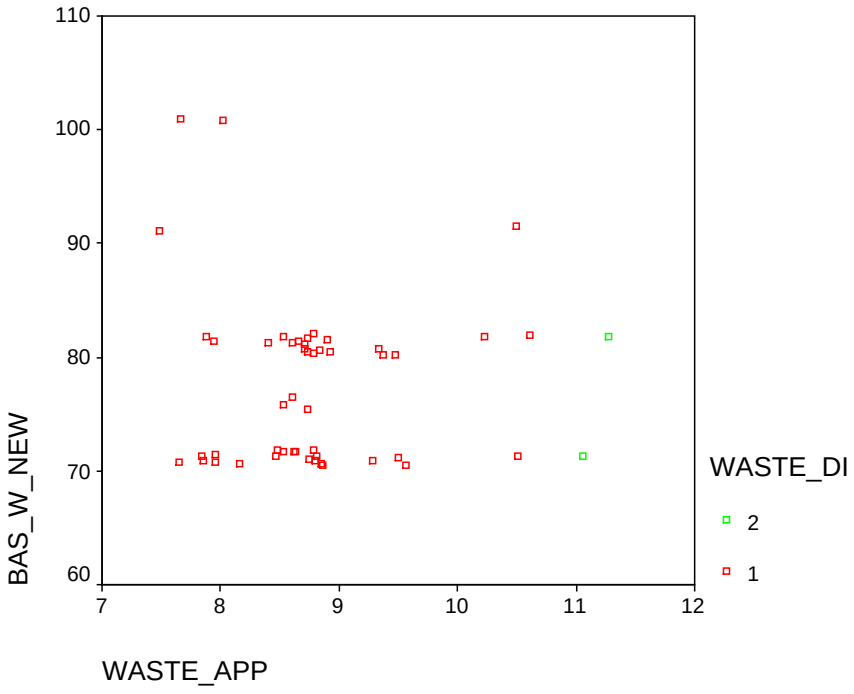
Edellä mainittujen seikkojen perusteella voidaan tehdä se johtopäätös, että projektissa käytetyn aineiston perusteella muodostetun päätöspuun avulla voidaan löytää säätömuuttujien alkuarvot ainakin että paperilaaduille 70 g/m^2 ja 80 g/m^2 kun katkon pituus on pienempi kuin 6000 minuuttia.

```
if s2sai4f <= 26 and s3sai6t <= 289
then
  node=2
  prediction='1'
  probability=1.000

if s2sai4f > 26 and s3suppre <= 180 and s3sai6f > 32 and s2sai3f <= 30 and s4sai8t
<= 241.5 and s4sai7t <= 150 and s2sai3t > 110 and s3suppre <= 130
then
  node=23
  prediction='1'
  probability=1.000

if s2sai4f > 26 and s3suppre <= 180 and s3sai6f > 32 and s2sai3f <= 30 and s4sai8t >
241.5 and s1sai2t <= 313.5 and s4sai8f > 28 and s3sai5t > 190 and s2sai4t <= 274
then
  node=42
  prediction='1'
  probability=1.000

if s2sai4f > 26 and s3suppre > 180 and s3sai5t > 180 and s3sai6t <= 293.5 and
s4suppre <= 280
then
  node=64
  prediction='1'
  probability=1.000

if s2sai4f > 26 and s3suppre <= 180 and s3sai6f > 32 and s2sai3f <= 30 and s4sai8t >
241.5 and s1sai2t <= 313.5 and s4sai8f > 28 and s3sai5t <= 190
then
  node=40
  prediction='1'
  probability=1.000
```

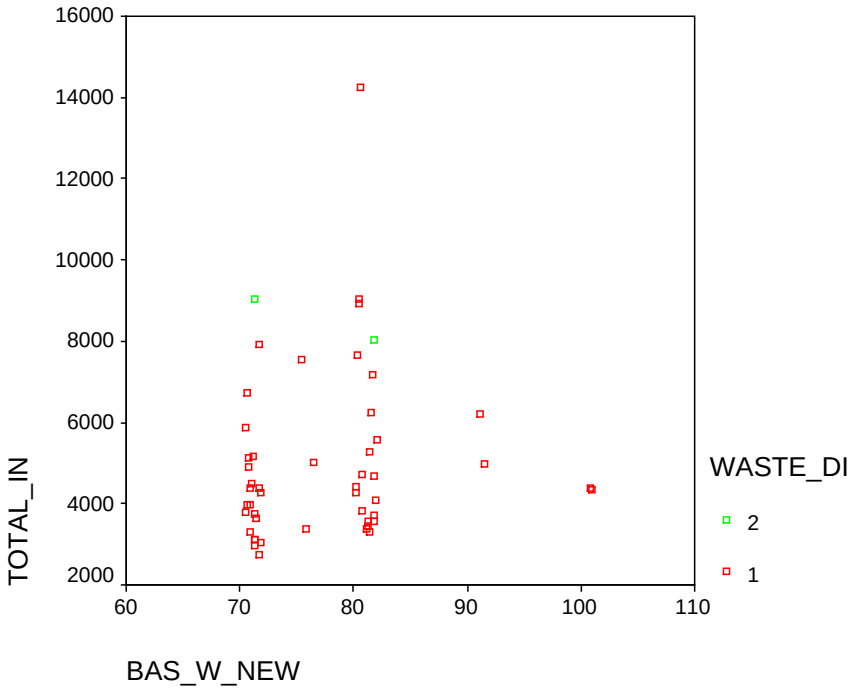
Taulukko 6.3. Analyysin 2.1 perusteella muodostetut luokittelusäännöt.



Kuva 6.15. Luokittelun tulosten arviointi katkon pituuden suhteen.



Kuva 6.16. Luokittelun arviointi paperilaadun suhteen.



Kuva 6.17. Luokittelun arviointi ulkoisten muuttujien välillä.

6.1.5 Yhteenveto tuloksista ja ehdotus jatkotoimenpiteistä

Päätöspuuta käytetään hyvien ylösajojen luokittelukriteerien löytämiseksi. Menetelmällä pyritään etsimään säännöt, joilla voidaan muuttujan WASTE_APP perusteella luokitella hyvien ylösajojen kriteerit. Analyyseissa saatujen alustavien kokemusten perusteella huomataan, että päätöspuumenetelmän avulla voidaan rakentaa kohtuullinen luokittelija, jonka avulla pystytään erottamaan hyvät ylösajot huonoista. Rakennettaessa päätöstukijärjestelmää alkuparametrien suositteluun eräänä lähestymistapana voisi olla seuraava menetelmä: Päätöspuuluokittelijasta tunnistetaan riittävän hyvin hyviksi luokittelevat lehtisolmut ja rakennetaan niiden tunnistamiseen sääntökanta. Tämän sääntökannan avulla leimataan opetusaineistossa olevien hyvien ylösajojen klusteriksi vastaava hyvän klusterin identifikaatio. Näille hyville klustereille lasketaan aineistosta tilastollisia tunnuslukuja (keskiarvo, keskipoikkeama, ylösajojen lukumäärä, luotettavuusarvo (montako prosenttia oikein luokiteltu). Näin saatuja hyviä klustereita hyödynnetään päätöksenteossa ylösajoa valmisteltaessa valitsemalla annettuja syötetietoja (uuden paperilaadun neliöpaino, neliöpainomuutos ja katkon pituus) parhaiten vastaavat hyvät klusterit ratkaisun pohjaksi.

6.2 Diskriminanttianalyysi

Diskriminanttianalyysin tarkoitus on löytää ne muuttujat, jotka "parhaiten" erottelevat ryhmiin jaetun aineiston ryhmät toisistaan. Näiden muuttujien perusteella voidaan tehdä luokittelusääntöjä, joiden avulla uudet havainnot sijoitetaan johonkin olemassa olevaan aineiston ryhmään.

6.2.1 Diskriminanttianalyysin periaate

Diskriminanttianalyysissä tarkastellaan havaintoaineistoa, joka on jaettu jollain tavoin ryhmiin. Ryhmät voivat perustua diskreetteihin suureisiin tai ne voidaan luoda diskretoimalla jatkuvia suureita. Diskriminanttianalyysissä pyritään löytämään "parhaat" erottelevat muuttujat, eli ne, joiden arvot vaihtelevat eniten eri ryhmiin kuuluvien havaintojen välillä.

Löydettyjen muuttujien avulla haetaan niiden lineaarikombinaationa uusi muuttuja tai indeksi, jonka avulla eri ryhmät pystytään erottamaan toisistaan. Käyttämällä löydettyjä erottelevia muuttujia tai niiden kombinaatiota luodaan lopulta sääntö, jonka avulla uudet havainnot pystytään luokittelemaan johonkin tarkastelluista ryhmistä. Sääntö uusien havaintojen luokitteluun voi perustua ryhmäkohtaisiin luokittelufunktioihin tai diskriminanttifunktioon, jonka saaman arvon perusteella uusi havainto luokitellaan.

Kahden ryhmän tapauksessa jaottelu voidaan tehdä diskriminanttifunktion tapauksessa yksinkertaisesti päättämällä diskriminanttifunktion arvolle jakopiste, jota pienemmän arvon saaneet havainnot luokitellaan ryhmään 1 ja suuremman arvon saaneet ryhmään 2. Toinen keino luokitella muuttujat on laskea kullekin ryhmälle havainnon muuttujien lineaarikombinaationa luokittelufunktio, ja liittää uusi havainto siihen ryhmään, jonka luokittelufunktio saa uuden havainnon tapauksessa suurimman arvon. Tällaisia ryhmäkohtaisia luokittelufunktioita kutsutaan Fisherin lineaarisiksi diskriminanttifunktioiksi [Sha96].

Diskriminanttianalyysi olettaa havaintojen olevan peräisin moniulotteisesta normaalijakaumasta. Mikäli ehto normaalijakautuneisuudesta ei toteudu, se vaikuttaa analyysissä tehtävien tilastollisten testien luotettavuuteen, sekä luokittelutuloksiin. Samaan tapaan luokittelussa käytettävien ryhmien kovarianssimatriisit oletetaan samoiksi, ja mikäli tämä ehto ei toteudu, vaikuttaa sekin analyysin tilastolliseen luotettavuuteen ja luokitteluun. Diskriminanttianalyysi on melko robusti menetelmä näiden ehtojen rikkomisille, mutta tuloksia analysoitaessa ehtojen rikkomisten mahdolliset vaikutukset on otettava huomioon [Sha96].

6.2.2 Havaintoaineisto A

Tehtäessä diskriminanttianalyysi päällystyskoneen ylösajoille jaetaan aineisto hyviin ja huonoihin ylösajoihin. Aineistona on 212 ylösajoa, ns. havaintoaineisto A, joista valitaan analyysiin kaksi ryhmää. Hyviin ylösajoihin lasketaan kuuluvaksi ne, joissa hylkyä syntyy alle 8,5 km (47 havaintoa, ryhmä 1) ja huonoihin ne, joissa hylkyä syntyy 11-20 km (69 havaintoa, ryhmä 2). Yli 20 km hylkymäärän tuottaneet ylösajot tulkitaan poikkeaviksi, ja ne jätetään huomiotta. Yhteensä kahteen ryhmään kuuluu 116 ylösajoa. Hyvien ja huonojen ylösajojen väliin jäävät "kohtalaiset" ylösajot jätetään analysin ulkopuolelle, jotta analysoitavat ryhmät erottuisivat toisistaan tarpeeksi selkeästi.

Ylösajojen hyvyyttä selittäviä muuttujia on analyysissä 34. Näistä 32 on katkon alkuhetkellä mitattuja säätöarvoja, ns. mittausmuuttujia ja 2 ylösajon ominaisuuksia, eli katkon pituus ja päällystettävän paperin neliöpainon muutos. Katkon alkuhetkenä mitattujen säätöarvojen oletettiin olevan samat, kuin lähtöarvot, joilla ylösajo suoritetaan.

Mittausmuuttujien ja ylösajon ominaisuuksien jakaumat poikkeavat selvästi normaalijakaumasta, eivätkä luokiteltavien ryhmien kovarianssimatriisit ole samoja. Nämä seikat vaikuttavat analyysin tilastollisten testien sekä löydettyyn diskriminanttifunktioon perus-

tuvan luokittelun luotettavuuteen. Diskriminanttifunktio ja analyysin mukaiset parhaat erottelevat muuttujat antavat kuitenkin jotain viitteitä ylösajojen ryhmittelyyn.

Ajamalla diskriminanttianalyysi SPSS-ohjelmalla saadaan luokittelufunktiot, jotka luokittelevat havaintoaineiston 78,4 prosenttisesti oikein (Taulukko 6.4). Tulos on optimistinen, koska mallia ei testattu uusiin havaintoihin. Realistisemman kuvan luokittelutehosta antaa ristiinvalidointi, jossa funktiot estimoidaan koko 116 ylösajon aineistolla yhtä havaintoa lukuunottamatta ja luokitellaan jäljelle jäänyt havainto saatujen luokittelufunktioiden avulla. Tällöin mallin luokittelutehoksi tulee 65,5 %. Huomattavaa on, että kummassakin tapauksessa huonot havainnot luokitellaan hyviä havaintoja helpommin oikeaan luokkaansa. Muuttuja S4_LOW_PRE jätettiin analyysistä pois, koska se korreloi liian voimakkaasti muiden muuttujien kanssa.

Classification Results[b,c]

			Predicted Group Membership		
		GROUP	1,00	2,00	Total
Original	Count	1,00	32	15	47
		2,00	10	59	69
		Ungrouped cases	45	51	96
	%	1,00	68,1	31,9	100,0
		2,00	14,5	85,5	100,0
		Ungrouped cases	46,9	53,1	100,0
Cross-validated[a]	Count	1,00	27	20	47
		2,00	20	49	69
	%	1,00	57,4	42,6	100,0
		2,00	29,0	71,0	100,0

a. Cross validation is done only for those cases in the analysis. In cross validation, each case is classified by the functions derived from all cases other than that case.

b. 78,4% of original grouped cases correctly classified.

c. 65,5% of cross-validated grouped cases correctly classified.

Taulukko 6.4. Havaintoaineiston A diskriminanttianalyysin luokittelun tulokset.

Aineiston luokittelu tehdään käyttäen Fisherin diskriminanttifunktioita (Taulukko 6.5). Havainto luokitellaan siihen luokkaan kuuluvaksi, jonka luokittelufunktio antaa suuremman arvon. Havaintoaineiston ryhmien erilaisuutta muuttujien suhteen voidaan arvioida ns. Wilksin lambdan avulla, joka on ryhmien sisäisten neliösummien ja kaikkien havaintojen neliösummien suhde. Wilksin lambda –tunnusluvun mukaan muuttujat ja siten myös analyysissä lasketut luokittelufunktiot erottelevat ryhmät toisistaan merkitsevyystasolla 0,037. Luokittelu on siis Wilksin lambdan mukaan perusteltua.

Eräs toinen diskriminanttianalyysiin liittyvä tunnusluku on kanonisen korrelaation neliö, joka saadaan ryhmien välisten neliösummien ja kaikkien havaintojen neliösummien suhteena. Kanonisen korrelaation neliö (CR^2) kertoo, kuinka suuren osan diskriminoivat muuttujat selittävät ryhmien välisestä vaihtelusta. Havaintoaineistolle A saatiin CR^2=0,39, joten tämän perusteella diskriminoivat muuttujat selittävät ryhmien vaihtelusta vain 39 %. Loput vaihtelusta selittää kohina sekä analyysin ulkopuoliset muuttujat, joista ei ole mittauksia.

On huomattava, että diskriminanttianalyysin oletuksien rikkomisen vuoksi tunnusluvut ovat tilastollisesti epävarmoja, eikä niiden perusteella voida tehdä varmoja johtopäätöksiä.

Classification Function Coefficients

	GROUP	
	1,00	2,00
S1_IR1P	4,992	4,827
S1_IR2P	,917	,971
S1_AI1T	,249	,206
S1_AI1F	12,717	12,692
S1_AI2T	,456	,475
S1_AI2F	-12,156	-12,243
S1_UPP	-2,597	-2,425
S1_LOW	58,505	58,693
S2_IR3P	-3,260	-3,172
S2_IR4P	1,120	1,130
S2_AI3T	,782	,764
S2_AI3F	,452	,407
S2_AI4T	-,324	-,316
S2_AI4F	7,687	7,587
S2_UPP	10,933	11,135
S2_LOW	-11,075	-11,271
S3_IR5P	5,508	5,362
S3_IR6P	,773	,831
S3_AI5T	-1,0E-02	-7,7E-03
S3_AI5F	-10,954	-10,803
S3_AI6T	-,234	-,228
S3_AI6F	8,112	7,674
S3_UPP	-5,903	-5,686
S3_LOW	5,900	5,693
S4_IR7P	1,968	1,869
S4_IR8P	-1,061	-1,164
S4_AI7T	,199	,223
S4_AI7F	4,641	4,912
S4_AI8T	,134	,135
S4_AI8F	-5,507	-5,318
S4_UPP	,351	,351
TOTAL_IN	-9,6E-05	-2,0E-05
DE_BAS_W	-,669	-,638
(Constant)	-1334,008	-1317,431

Fisher's linear discriminant functions

Taulukko 6.5. Havaintoaineistosta A lasketut Fisherin diskriminanttifunktiot.

6.2.3 Havaintoaineisto B

Diskriminanttianalyysi havaintoaineistolle B tehtiin samaan tapaan kuin aineistolle A. Aineiston vähyyden vuoksi tarkasteltaviksi ryhmiksi otettiin havaintoaineistosta A poiketen

alle 9 km hylkyä tuottaneet havainnot (49 havaintoa, ryhmä 1) ja 11-20 km hylkyä tuottaneet havainnot (61 havaintoa, ryhmä 2). Tällöin ryhmiin kuuluu yhteensä 110 havaintoa, joka on samaa suuruusluokkaa kuin havaintoaineistossa A. Analyysissä mukana olevat muuttujat ovat 24 asetusarvoa, paperilaadun uusi ja vanha neliöpaino sekä katkon pituus, yhteensä 27 muuttujaa.

Tässäkään havaintoaineistossa muuttujat eivät ole normaalijakautuneita, eivätkä ryhmien kovarianssimatriisit samoja. Analyysin tilastollisesta luotettavuudesta ei tästä johtuen pystytä sanomaan mitään varmaa.

Ajetaan diskriminanttianalyysi SPSS-ohjelmalla havaintoaineistolle B (Taulukko 6.6). Vaikka diskriminanttifunktion estimointiin käytettyjen havaintojen luokittelu onnistuukin havaintoaineistoa A paremmin, saadaan ristiinvalidoinnilla tulokseksi sama 65,5 % kuin aineistolla A. Muuttujat S2S_LOW_PRE, S3S_LOW_PRE ja S4S_LOW_PRE jätetään analyysistä pois, koska ne korreloivat liian selvästi muiden muuttujien kanssa. Tämä nähtiin jo esimerkiksi kohdan 5.2.3 kuvasta 5.5.

Classification Results[b,c]

			Predicted Group Membership		
		GROUP	1,00	2,00	Total
Original	Count	1,00	37	12	49
		2,00	11	50	61
		Ungrouped cases	18	15	33
	%	1,00	75,5	24,5	100,0
		2,00	18,0	82,0	100,0
		Ungrouped cases	54,5	45,5	100,0
Cross-validated[a]	Count	1,00	29	20	49
		2,00	18	43	61
	%	1,00	59,2	40,8	100,0
		2,00	29,5	70,5	100,0

a. Cross validation is done only for those cases in the analysis. In cross validation, each case is classified by the functions derived from all cases other than that case.

b. 79,1% of original grouped cases correctly classified.

c. 65,5% of cross-validated grouped cases correctly classified.

Taulukko 6.6. Havaintoaineistolle B tehdyn diskriminanttianalyysin luokittelutulokset.

Alla analyysin mukaiset Fisherin diskriminanttifunktioiden kertoimet (Taulukko 6.7). Wilksin lambda –tunnusluku antaa muuttujien erottelukyvylle merkitsevyystason 0,005, ja tämä viittaa siihen, että havaintoaineiston B muuttujat erottelevat ryhmät toisistaan havaintoaineistoa A paremmin. Näin ollen myös diskriminanttifunktioiden erottelukyky olisi aineistolla B parempi. Kanonisen korrelaation neliöksi saadaan kuitenkin vain 0,38, mikä johtuu aineistojen havaintojen ja muuttujien määrän eroista.

Edelleen muistutetaan, että tunnuslukuihin on suhtauduttava varauksella analyysin aineistolle tekemien oletusten rikkoutumisen vuoksi. Tämä epäluotettavuuden ja aineistojen

ryhmien erilaisen määrittelyn vuoksi ei aineistojen A ja B tuloksien hyvyyttä voida vertailla.

Classification Function Coefficients

	GROUP	
	1,00	2,00
S1S_AI1T	,220	,193
S1S_AI2T	,316	,316
S1S_AI1F	6,208	6,157
S1S_AI2F	2,613	3,092
S2S_AI3T	1,041	,990
S2S_AI4T	-,257	-,248
S2S_AI3F	-1,115	-,721
S2S_AI4F	1,630	1,772
S2S_UPP	-,194	-,195
S3S_IR6P	3,877	3,829
S3S_AI5T	,464	,475
S3S_AI6T	5,83E-02	6,61E-02
S3S_AI5F	7,355	7,088
S3S_AI6F	-6,633	-6,636
S3S_UPP	,166	,172
S4S_IR8P	1,093	1,045
S4S_AI7T	-,470	-,460
S4S_AI8T	,163	,165
S4S_AI7F	-2,570	-2,374
S4S_AI8F	6,735	6,311
S4S_UPP	-,466	-,456
TOTAL_IN	-2,2E-04	-6,2E-05
BAS_W_NE	1,749	1,894
BAS_W_OL	-,396	-,567
(Constant)	-603,773	-605,449

Fisher's linear discriminant functions

Taulukko 6.7 Havaintoaineiston B Fisherin diskriminanttifunktioiden kertoimet.

6.2.4 Yhteenveto diskriminanttianalyysin tuloksista

Diskriminanttianalyysin antamat tulokset viittaavat siihen, ettei kummallakaan havaintoaineistolla A tai B käytetyt muuttujat riitä selittämään tuloksia kovin suurella tarkkuudella. Tilastollisesti ryhmät eroavat toisistaan, mutta tämä johtuu lähinnä diskriminanttianalyysille verraten suuresta havaintojen määrästä, jolloin pienetkin erot tulevat merkitseviksi. Havaintoaineiston kasvattaminen ei siis diskriminanttianalyysin kannalta toisi välttämättä lisää olennaista informaatiota. Kummassakin tapauksessa ryhmien välisestä vaihtelusta jäi kuitenkin selittämättä n. 60 %. Tämä voi johtua joko hylkymäärään liittyvästä voimakkaasta kohinasta, joka peittää alleen muuttujien vaikutuksen, tai piilomuuttujista, jotka eivät ole mukana analyysissä. Jälkimmäinen tuntuu todennäköisimmältä vaihtoehdolta.

Tulosten luotettavuuteen vaikuttavat diskriminanttianalyysin oletusten rikkoutumiset. Analyysissä käytetyt muuttujat eivät ole normaalijakautuneita, eivätkä ryhmien kovarianssimatriisit ole samoja.

Näillä perusteilla diskriminanttianalyysin käyttöä jatkossa tämän tapaustutkimuksen yhteydessä ei suositella. Tuloksia voidaan käyttää korkeintaan vertailussa muiden menetelmien tuloksiin.

7 Mallintamis- ja päättelymenetelmät

Mallintamismenetelmissä muodostetaa ensin aineistosta malli, ja senjälkeen tutkitaa mallin ominaisuuksia, toivoen niiden vastaavan riittävän hyvin aineiston kuvaaman todellisuude ominaisuuksia. Tämän luvun lopun päättelymenetelmä ottavat niitä käytettäessa aina koko aineiston käyttöönsä, ja hakevat sen perusteella vastausta esitettyyn kysymykseen – toisin kuin aiempien lukujen luokittelu- ryvästys- yms. menetelmät, joissa on "esipäättelyvaihe", jonka perusteella varsinaista ongelmaa ratkotaan.

7.1 Lineaarinen mallintaminen

7.1.1 Johdanto

Funktio on lineaarinen, jos se lasketaan muuttujista ainoastaan kertomalla muuttujia (positiivisilla tai negatiivisilla) vakioilla, summaamalla näin saadut tulokset, sekä mahdollisesti lisäämällä vakio. Lähes kaikkien perinteisten tilastollisten menetelmien taustalla on oletus, että tutkittava ilmiö on lineaarinen luonteeltaan. Mielivaltaista funktiota voidaan approksimoida lineaarisella funktiolla, mutta yleisessä tapauksessa tämä approksimaatio on niin puutteellinen, ettei "ennustuksia" alkuperäisen funktion käyttäytymisestä voida tehdä kovin laajalla alueella.

Sensijaan lineaarinen approksimaatio saattaa olla hyvä malli, jos meillä on havaintoja funktion käyttäytymisestä jonkin suhteellisen pienen "kiinnostavan alueen" sisällä, ja haluamme tehdä "yhteenvedon" noista havainnoista. On myös helppoa nähdä, että kuinka hyvä approksimaatio lineaarinen malli on.

Mallin sovittamisen jälkeen on vielä haettava ne muuttujien arvot, joilla syntyy vähin määrä hylkyä. Tämä on optimointiongelma, mutta vaikka funktio onkin lineaarisen mallin sovituksen jälkeen yksinkertainen, on rajoitteiden eli "muuttujien sallittujen arvojen" määrittely erittäin haastava tehtävä.

Päällystyskoneen ylösajossa kaikille muuttujille voidaan periaatteessa asettaa mielivaltaiset arvot, paitsi katkon pituudelle sekä päällystettävän paperin neliöpainolle ennen ja jälkeen katkon. Tästä syystä olisi kaksi mahdollista tapaa lähestyä mallintamista:

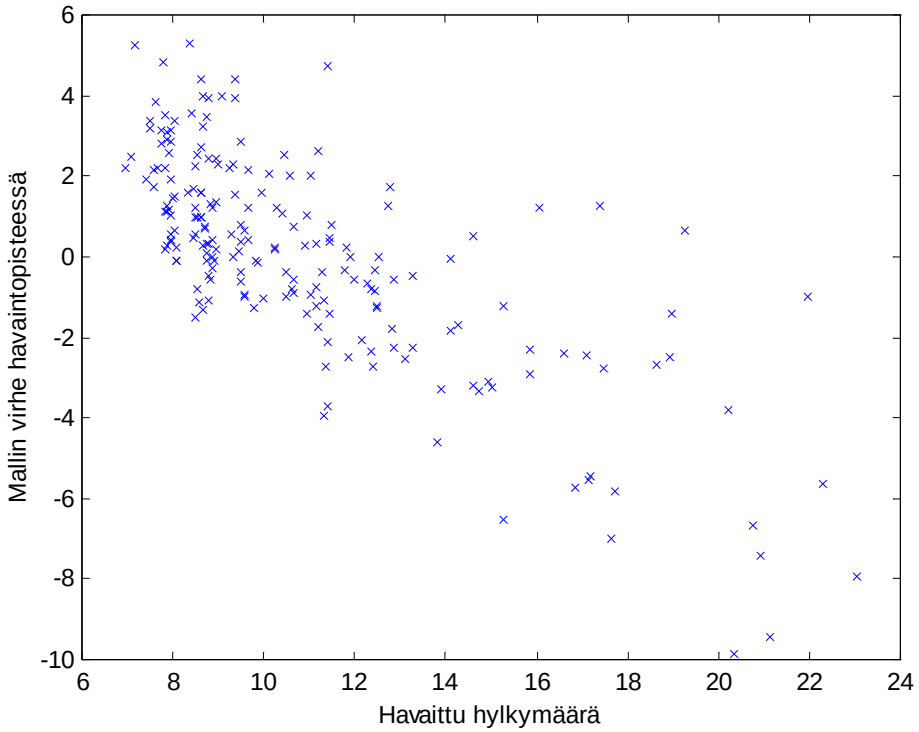
1. Jaetaan havaintoaineisto osiin, esimerkiksi neljään osaan katkon pituuden ja neliöpainon muutoksen mukaan. Tehdään jokaiseen osaan oma mallinsa.

2. Tehdään yksi ainoa malli, ja käsitellään katkon pituus ja neliöpainot rajoitteina haettaessa mallin antamaa minimi-hylkymäärää.

Edellisessä vaihtoehdossa kunkin mallin sovittamiseen käytettävissä oleva havaintopisteiden määrä vähenisi radikaalisti, ja mallien luotettavuus vähenisi vastaavasti. Lisäksi mallin antamat parhaat alkuparametrit "hyppäisivät" aivan erilaisiksi astuttaessa mallien rajapinnan yli. Tästä syystä valittiin jälkimmäinen vaihtoehto lähtökohdaksi.

7.1.2 Mallin sovittaminen aineistoon A

Kuvassa Kuva 7.1 on sovitettu lineaarinen malli muuten koko aineistoon A, paitsi 10 suurimman hylkymäärän katkoa on poistettu:



Kuva 7.1. Lineaarinen malli koko aineistolle sovitettuna.

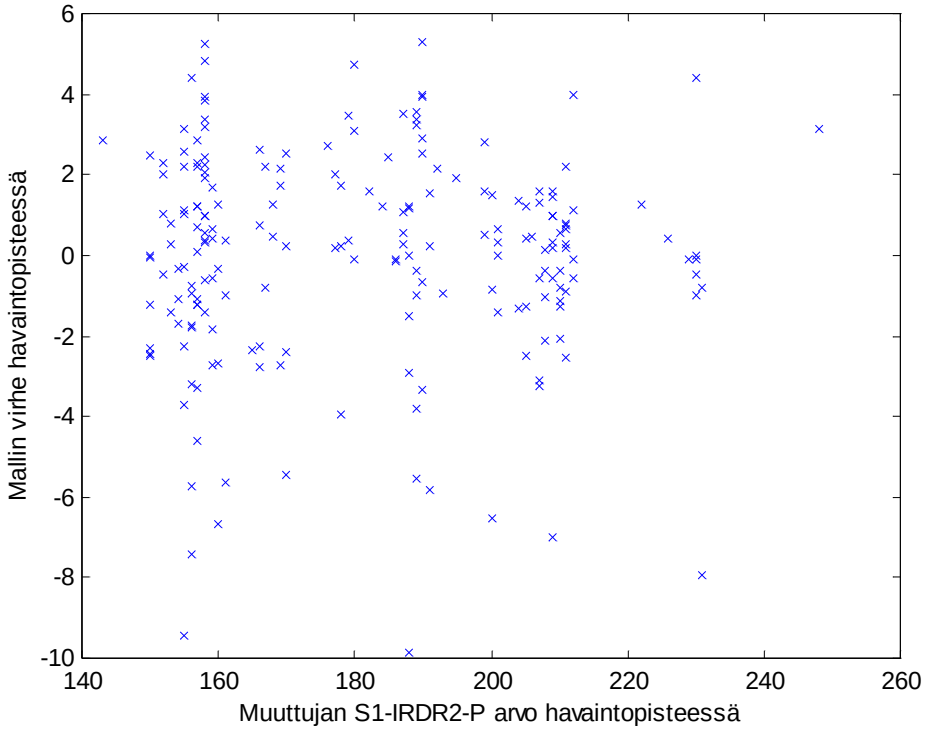
Kuvassa kaksi seikkaa kiinnittää huomiota:

1. Mallin virheen vaihteluväli on suuri.

2. Mallissa on jokin systemaattinen virhe, koska jos sovitetaan mallin virheeseen suora, niin suora on selvästi laskeva.

Molemmat seikat johtunevat siitä, että mallissa on niin paljon epälineaarisuutta, ettei lineaarinen malli "selitä" vaan "keskiarvoistaa".

Seuraava askel on pyrkiä löytämään epälineaarisuuden aiheuttavat muuttujat. Tämä tehtiin piirtämällä sirontakaaviot "muuttujan arvo havaintopisteessä / mallin virhe havaintopisteessä" kaikille muuttujille. Jos taas löytyisi jonkinlainen systemaattinen virhe, olisimme paikallistaneet epälineaarisuuden aiheuttavat muuttujat (ja voisimme tehdä epälineaarisen mallin). Tyypillinen kaavio on tällainen.

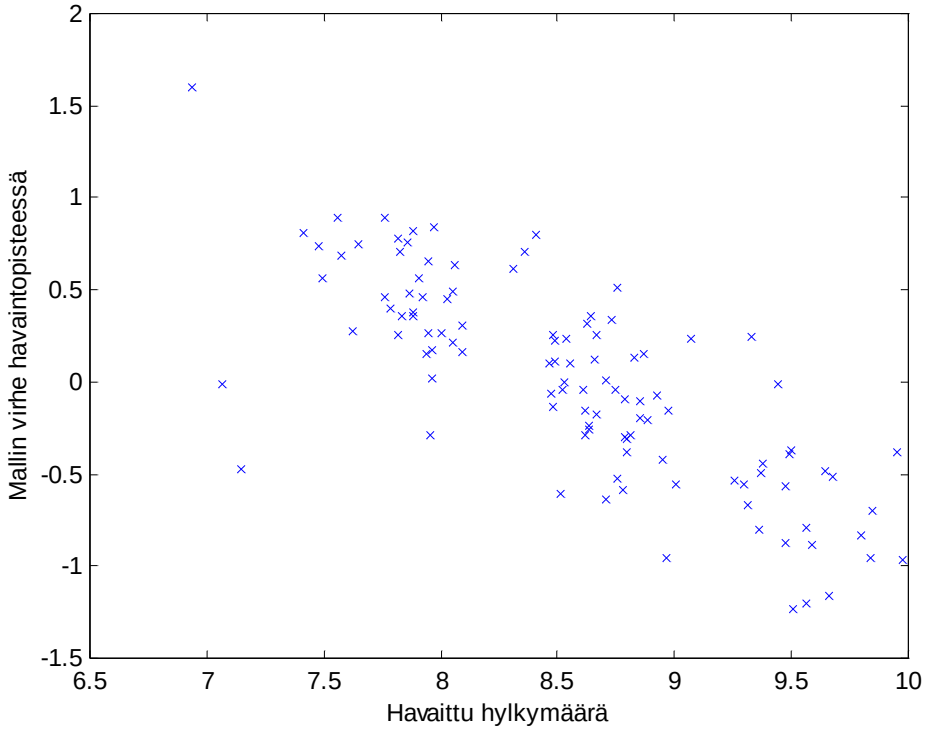


Kuva 7.2. Esimerkki sirontakaaviosta (muuttujan arvo, mallin virhe).

Valitettavasti kaikille muuttujille piiretyt sirontakaaviot osoittivat, että mallin virheessä ei ollut selvää epälineaarisuutta osoittavaa systemaattisuutta millekään muuttujalle – tämä tarkoittaa joko sitä, että

- Epälineaarisuuden aiheuttavat tekijät ovat edelleen "piilossa", ja niitä pitäisi tutkia lisää, tai
- Muuttujilla ei ole riittävää syy-seuraussuhdetta syntyvän hylyn määrään.

Seuraavaksi päätettiin yrittää mallintaa ainoastaan onnistuneita ylösajoja, eli niitä joissa hylkyä on syntynyt vähemmän kuin 10 kilometriä. Tämä on sikäli perusteltua, että "onnistunut" ylösajo noudattaa yleensä samoja lainalaisuuksia, mutta "epäonnistuminen" voi johtua mitä moninaisimmista, mallin ulkopuolisistakin syistä. Lisäksi työn tavoitteena on löytää nimenomaan onnistumisen edellytykset. Tuloksena oli tällainen sovitus:



Kuva 7.3. Lineaarinen malli "hyville" ylösajoille.

Nyt mallin sovitus onnistui jo paremmin. Systemaattisen virheen läsnäolo näkyy kuitenkin edelleen, ja itseasiassa kuvasta nähdään, että kutakin virhettä on "yhtä usein". Toisinsanoen, mallin virhe ei noudata normaalijakaumaa, vaan on tasaisesti jakautunut. Tämä on huono merkki. Mallin selitysaste on 34% (ts. malli selittää 34% havaitusta hylkymäärän variaatiosta) [MiA95, s. 494]. Tämä luku voisi olla parempikin, erityisesti kun meillä on hyvin vähän havaintopisteitä (109) muuttujien määrään (35) nähden.

Niinpä tätä mallia ei voi kutsua "selittäväksi malliksi" vaan "keskiarvoistavaksi malliksi". Malli voi silti olla käyttökelpoinen alkuperäiseen tarkoitukseensa, eli minimi-hylkymäärän antavien alkuparametrien hakemiseen. On kuitenkin muistettava, että "keskiarvoistavan" mallin ollessa kyseessä malli on luotettavimmillaan "havaintojoukon keskellä". Mutta koska tämä malli oli sovitettu pelkästään "hyville ylösajoille", niin tämä voi olla hyväksyttävää.

7.1.3 Mallin sovittaminen ainestoon B

Taulukko 7.1 kuvaa tammikuisen mallin sovittamisen säätöarvoille antamat tulokset (muuttujia oli 27 kpl). Malliin voidaan ottaa lähes kaikki katkot (alle 25 km hylkyä) tai ainoastaan "hyvät ylösajot". Nyrkkisääntönä katkoja tulisi olla vähintään 3* muuttujien määrä (koska muutoin on vaara, että mallinnetaan kohinaa). Tässä mielessä ainoastaan "alle 12 km hylkyä"-malli vaikuttaisi mahdollisesti käyttökelpoiselta.

Mitkä katkot mukana?	Katkojen lkm	Selitysaste
Alle 25 km hylkyä	133	38%
Alle 12 km hylkyä	74	48%
Alle 10 km hylkyä	56	57%
Alle 9 km hylkyä	50	72%

Taulukko 7.1. Lineaarimalli sovitettuna aineistoon B.

On kuitenkin muistettava, että kaikilla muuttujilla on äärellinen mittaustarkkuus. Taulukko 7.2 kuvaa, että mitkä muuttujat poikkeavat toisistaan yli 5% kahden eri mallin palauttamassa pisteessä (jos poikkeama on alle 5%, voidaan ajatella sen olevan mittaustarkkuuden rajoissa) (katkon pituutena ja neliöpainoina ennen ja jälkeen katkon on käytetty aineiston mediaaneja).

Ensimmäinen malli:	Toinen malli:	Poikkeavat muuttujat:
Alle 25 km hylkyä	Alle 12 km hylkyä	S2S_UPP_PRE, S2S_LOW_PRE, S3S_IRDR6_P, S3S_AIDR5_T, S4S_AIDR7_T
Alle 25 km hylkyä	Alle 10 km hylkyä	S2S_AIDR4_T, S2S_AIDR4_F, S3S_IRDR6_P, S4S_AIDR7_T, S4S_AIDR8_F, S4S_UPP_PRE, S4S_LOW_PRE
Alle 25 km hylkyä	Alle 9 km hylkyä	S2S_AIDR4_T, S3S_IRDR6_P, S4S_AIDR7_T, S4S_UPP_PRE, S4S_LOW_PRE
Alle 12 km hylkyä	Alle 10 km hylkyä	S2S_AIDR4_T, S2S_AIDR4_F, S2S_UPP_PRE, S2S_LOW_PRE, S3S_AIDR5_T, S4S_AIDR8_F, S4S_UPP_PRE, S4S_LOW_PRE

Alle 12 km hylkyä	Alle 9 km hylkyä	S2S_AIDR4_T, S2S_UPP_PRE, S2S_LOW_PRE, S4S_UPP_PRE, S4S_LOW_PRE
Alle 10 km hylkyä	Alle 9 km hylkyä	S4S_AIDR8_F, S4S_UPP_PRE, S4S_LOW_PRE

Taulukko 7.2. Eri mallien vastauksissa arvoiltaan eniten poikkeavat muuttujat.

Kymmenen muuttujan arvot eivät vaihdelleet lainkaan eri pisteissä. Taulukko 7.3 näyttää yhteenvedon vaihdelleista muuttujista.

Muuttuja	Kuinka monessa pisteessä oli poikkeava
S2S_AIDR4_T	4
S4S_UPP_PRE	4
S4S_LOW_PRE	4
S2S_UPP_PRE	3
S2S_LOW_PRE	3
S3S_IRDR6_P	3
S4S_AIDR7_T	3
S2S_AIDR4_F	2
S3S_AIDR5_T	2
S4S_AIDR8_F	2

Taulukko 7.3. Yhteenveto eniten poikkeavista muuttujista.

Muiden muuttujien arvot siis joko pysyvät samoina, tai eivät vaihtele 5 % enempää vastauspisteessä mallista toiseen.

7.1.4 Parhaiden alkuparametrien hakeminen

Oletetaan, että havaintopisteisiin on sovitettu lineaarinen malli. Kuinka löydetään piste, jossa malli antaa pienimmän vasteen (eli pienimmän hylyn määrän)? On helppoa löytää paras havaintopiste, mutta entäpä paras mallin antama piste "luotettavalta alueelta"?

Muuttujille on määriteltävä rajoitteet, eli sallittu etsintäalue. Ensimmäiseksi tulee mieleen käyttää jokaisen muuttujan arvoalueita havaintoaineistossa, jolloin etsintäalue olisi "hypersuorakulmio". Tämä ei kuitenkaan sovi: edellisen luvun parhaaseen malliin sovitettuna tämä tekniikka antaa pienimmäksi hylkymääräksi –10 kilometria! Havaintoaineisto on "hypersuorakulmion" keskialueella, ja minimin antava kärkipiste on niin kaukana siitä, ettei lineaarinen malli enää päde. Tämä ongelma vain pahenee ulottuvuuksien kasvaessa.

Seuraava tapa voisi olla määritellä "pienin hypertasoilla määritelty konveksi joukko, joka sulkee sisäänsä kaikki (tai ainakin kaikki luotettavat) havaintopisteet", ja hakea minimi tämän joukon sisällä. Valitettavasti joukon määrittelemisen aikavaatimus kasvaa eksponen-

tiaalisesti ulottuvuuksien suhteen [Cha93], ja käytännössä osoittautui mahdottomaksi ratkaista ongelmaa yli 11-ulotteisessa avaruudessa qhull-ohjelmalla (http://www.geom.umn.edu/software/qhull/). Muutamassa CPU-minuutissa onnistui korkeintaan 8-ulotteisen ongelman ratkaisu. Koska meillä on yli 30 muuttujaa, olisi ainoa mahdollisuus soveltaa tätä lähestymistapaa ongelman redusointi pienempiulotteiseksi esimerkiksi pääkomponenttianalyysillä. On kuitenkin kyseenalaista, että onnistuuko tämä aina (pääkomponenttianalyysi voisi antaa esimerkiksi 11 merkittävää komponenttia, ja niistä 8 tärkeimmän käyttäminen voisi redusoida ongelmaa liikaa).

Edelleen, voitaisiin yrittää määritellä "luotettava alue" siten, että jos mielivaltaisen pisteen etäisyys lähimpään havaintopisteeseen on jotain vakiota pienempi, niin tämä piste kuuluisi "luotettavaan alueeseen". Tässä on kuitenkin se ongelma, että "luotettavasta alueesta" tulee epäkonveksi joukko, ja lineaarisenkin funktion minimin hakeminen siitä on huomattavan vaikea ongelma.

Paras "perinteinen" ratkaisu olisi käyttää projektiopisteen [Rik79] käsitettä seuraavasti: määritellään ensin se arvo jota haetaan (esimerkiksi pienin havaintoarvo tai havaintoarvojen mediaani). Sitten lasketaan havaintoarvojen "keskipiste" (esimerkiksi kunkin muuttujan mediaanien avulla) ja haetaan tämän keskipisteen projektiopiste mallin sillä "hypertasolla", jolla mallin vaste on haluttu. Tämä lähestymistapa voisi muuten olla sopiva, mutta yksinkertaisten rajoitteiden käsittelyssä (kun halutaan esimerkiksi parin, kolmen muuttujan olevan vakioarvoisia) projektiopisteestä olisi liikuttava pois jotta saadaan rajoitteet voimaan, ja tässä on hankala ottaa huomioon "luotettavan alueen" vaatimusta.

Ongelmaan kehitettiin uusi ratkaisu. Se osoittautui yksinkertaiseksi, tehokkaaksi sekä luotettavaksi. Oletuksena on, että malli on sitä luotettavampi, mitä lähempänä aineiston "keskipistettä" (jonka määrittelevät kunkin muuttujan mediaanit) ollaan (voidaan kehittää patologisia aineistoja, joille tämä ehto ei päde, mutta tämän tutkimuksen aineistolle se pätee).

Algoritmi voidaan kuvata esimerkin avulla seuraavasti: oletetaan, että meillä olisi vain kaksi selittävää muuttujaa, x ja y, sekä yksi selitettävä f. On joukko havaintoja (x,y) sekä selitettävän f havaittuja arvoja seuraavasti:

(1,2): -1
(4,2): 4
(3,0): 1
(6,3): -2
(5,5): 0

joiden perusteella muodostetaan lineaarinen malli, esimerkiksi pienimmän neliösumman estimoinnilla. Oletetaan, että optimoinnin tuloksena on saatu malli F = x - y selittämään havaittua f:ää (esimerkki on hypoteettinen). Tiedetään pienin havaittu f, mutta halutaan tietää, että missä "luotettavassa" pisteessä malli F minimoituu?

Algoritmin on ideana lähteä kulkemaan muuttujien vaihteluväleistä saadusta minimipisteestä kohti "keskipistettä" muuttujien jakaumien sanelemia polkuja pitkin, kasvattaen F.n arvoa. Eteneminen lopetetaan siinä vaiheessa, kun F:n arvo on suurempi kuin havaittujen f:n mediaani. Ensiksi muodostetaan kokeilupisteet. Ne saadaan järjestämällä muuttujien arvot kasvavaan tai pienenevään järjestykseen sen mukaan, onko niiden kerroin mallissa positiivinen tai negatiivinen. Esimerkissämme kokeilupisteiksi tulevat:

(1,5)
(3,3)
(4,2)
(5,2)
(6,0)

Ensimmäisessä kokeilupisteessä malli estimoi F = 1-5=-3. Koska –3 on pienempi kuin havaittujen f:n mediaani 0, etenemistä jatketaan. Toisessa pisteessä saadaan F = 3-3 =0. Koska tämä on yhtä kuin havaintojen mediaani, algoritmi pysähtyy, ja tulostetaan vastauksena piste (3,3).

Jos halutaan "tarkkaa" vastausta, voidaan hakea piste, jossa mallin arvo on havaintojen mediaanin suuruinen algoritmin kahta viimeistä pistettä yhdistävältä suoralta. Havaintojen mediaanin sijaan referenssiarvona voitaisiin käyttää myös esimerkiksi pienintä havaittua arvoa, jos lineaarinen malli koettaisiin luotettavaksi.

Käyttäjät voivat haluta antaa muuttujille yhtäsuuruus-tyyppisiä rajoitteita, esimerkiksi "Anna parhaat alkuparametrit, kun katkon pituus on 30 minuuttia, neliöpaino ennen katkoa 80 ja katkon jälkeen 100". Nämä rajoitteet otetaan algoritmissa huomioon yksinkertaisesti pakottamalla kokeilupisteissä rajoitettujen muuttujien arvot halutuiksi heti kokeilupisteitä muodostettaessa.

Sekä lineaarisen mallin sovittaminen, että yo.algoritmi on toteutettu matlabin m-tiedostona. Se on hyvin yksinkertaisesti tuotteistettavissa itsenäiseksi C-ohjelmaksi, jotta asiakas ei tarvitsisi erillistä matlab-lisenssiä.

7.1.5 Parhaat alkuparametrit taulukoituna

Jotta saataisiin alustava vaikutelma parhaista lähtöarvoista, laskettiin ne systemaattisesti 50, 90 ja 130 minuutin mittaisille katkoille ja kaikille neliöpainojen kombinaatioille 70, 80 ja 90:stä – tuloksena oli kaikkiaan 27 laskettua lähtöarvoa. Käytetty lineaarimalli oli "alle 12 km hylkyä".

Mutta yllättäen 16:n muuttujan arvot eivät vaihdelleet lainkaan näissä pisteissä, ja loputkin muuttujat vaihtelivat tyypillisesti vähemmän kuin plus miinus 1%. Eniten vaihteli S3S_AIDR5_T, mutta sekin vain plus miinus 2,9%. Jos mittaustarkkuuden oletetaan olevat plus miinus 5%, ovat kaikki pisteet mittaustarkkuuden sisällä samoja.

Näinollen lineaarimalli suosittelee seuraavia lähtöarvoja kaikissa tilanteissa:

S1S_AIDR1_T: 170	S1S_AIDR2_T: 271,2	S1S_AIDR1_F: 30
S1S_AIDR2_F: 32	S2S_AIDR3_T: 15	S2S_AIDR4_T: 216,6
S2S_AIDR3_F: 28	S2S_AIDR4_F: 30	S2S_UPP_PRE: 80
S2S_LOW_PRE: 60	S3S_IRDR6_P: 85	S3S_AIDR5_T: 170
S3S_AIDR6_T: 259,8	S3S_AIDR5_F: 30	S3S_AIDR6_F: 34
S3S_UPP_PRE: 180	S3S_LOW_PRE: 200	S4S_IRDR8_P: 85
S4S_AIDR7_T: 170	S4S_AIDR8_T: 263,9	S4S_AIDR7_F: 30

S4S_AIDR8_F: 34 S4S_UPP_PRE: 179,2 S4S_LOW_PRE: 160

7.2 Epälineaarinen mallintaminen

Lineaarinen malli on usein paras lähtökohta, mutta joskus se jää epätyydyttäväksi. Voi esimerkiksi olla syytä olettaa kahdella tai useammalla muuttujalla olevan sellaista yhteisvaikutusta, joka jää lineaarisuuden vuoksi huomioonottamatta.

Tällöin voidaan asiantuntijatietämystä käyttäen lisätä malliin epälineaarisia termejä, mutta säilyttää silti mallin hakeminen lineaarisena tai pienimmän neliösumman optimointitehtävänä. Jos esimerkiksi edellisessä luvussa esitetty malli F osoittautuisi epätyydyttäväksi, voitaisiin tehdä hypoteesi, että ehkäpä malli

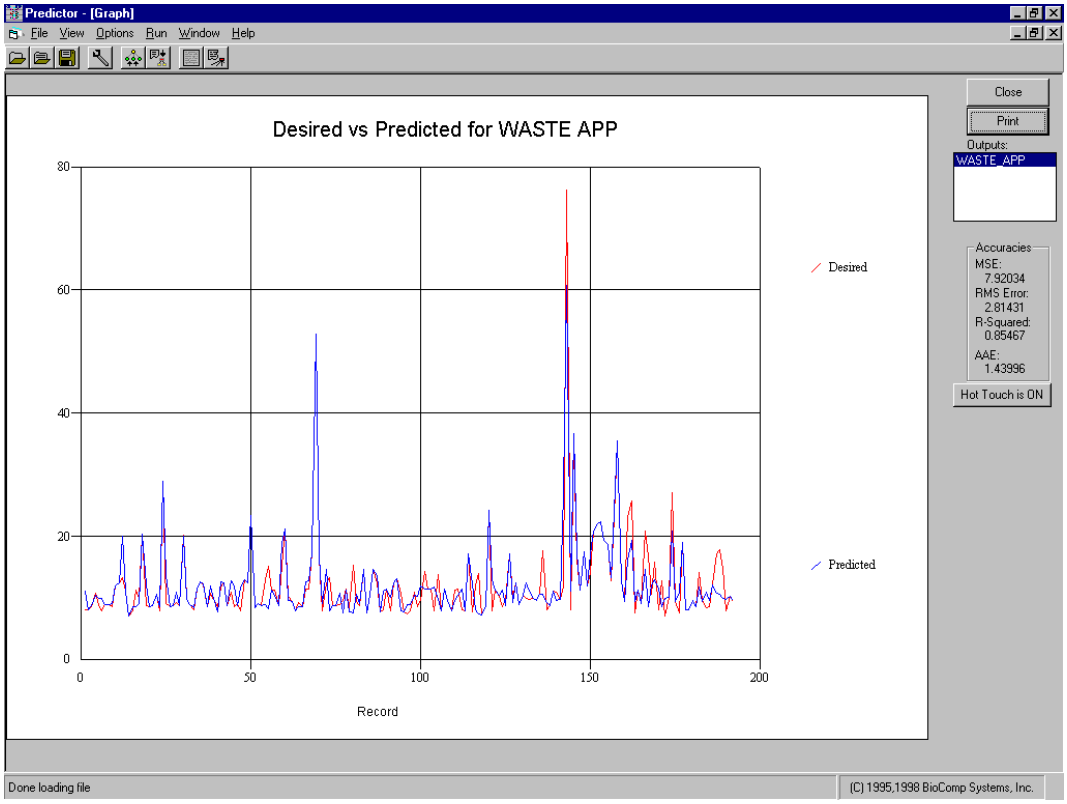
F = a*x + b*y + c + d*x*y

kuvaisi ongelmaa paremmin. Havaintojoukkoon "lisättäisiin" uutena "muuttujana" aiempien muuttujien tulo, ja estimoitaisiin samalla algoritmilla kuin aiemminkin tämä uusi malli. Mallin estimoija luulisi uuden muuttujan olevan riippumaton aiemminsta, mutta tämä ei haittaa. Parametrin d parhaiten sopivan arvon haku on edelleen lineaarinen optimointiongelma, samoin kuin muidenkin parametrien, vaikka malli onkin epälineaarinen. Ainoa ongelma on, kuinka suuriulotteisissa ongelmissa tehdään "älykkäitä arvauksia" uusiksi epälineaarisiksi termeiksi – jotkut tilastolliset menetelmät auttavat tässä.

Edellisessä luvussa kuvattu parhaiden alkuparametrien hakeminen toimii edelleen, mutta kokeilupisteitä muodostettaessa täytyy ottaa huomioon uusien muuttujien riippuvuus vanhoista muuttujista.

7.3 Neuroverkot

Ongelma annettiin neuroverkko-ohjelmiston mallinnettavaksi. BioComp Systems, Inc (http://www.biocompsystems.com/) myy NeuroGeneticOptimizer –ohjelmistoa, joka sekä hakee ongelmaan parhaiten sopivan neuroverkon (geneettisillä algoritmeilla), että opettaa sen. Verkolle annettiin syöttötietoina kaikki alkuparametrit, ja tavoitteena oli ennustaa niistä ylösajossa syntyvän hylyn määrä. Alla kuva koko aineiston pohjalta, kun puolet aineistosta käytettiin verkon opetukseen, ja puolet testaukseen – x-akseli kuvaa katkon numeroa, ja y-akseli ylösajossa syntynyttä(punainen) sekä ennustettua (sininen) hylyn määrää:



Kuva 7.4. Parhaan neuroverkon käyttäytyminen.

Huomiota tässä kiinnittää ennenkaikkea ihmeteltävän hyvä tulos. Se saattaa olla itse-asiassa jo liian hyvä: kyse voi olla jo ylioppimisesta ja "kohinan mallintamisesta".

Tämän jälkeen, jos/kun on löydetty aineiston hyvin oppinut neuroverkko, olisi löydettävä verkon antama minimi ja minimikohta. Tämä on hyvin haasteellinen ongelma yleisessä tapauksessa (globaali optimointiongelma). Onneksi kuitenkin BioComp Systems myy myös ExamiNeur- ohjelmistoa, joka nimenomaan tutkii opetetun neuroverkon vaste-pintaa, ja kertoo minimistä. Tämä ohjelmisto on tilattu VTT Tietotekniikkaan, mutta sitä ei ole vielä saatu käyttöön. On kuitenkin painotettava, että sama ongelma joka kohdattiin lineaarisen mallin tutkimisessa, eli mielekkään "luotettavan alueen" määritteleminen, on olemassa myös täällä, ja todennäköisesti paljon vaikeampana. Nähtäväksi jää, kuinka hy-vin ExamiNeur ratkoo tätä ongelmaa.

Jos saadaan hyvä verkko, niin tämän jälkeen verkko voitaisiin ottaa tuotantokäyttöön joko ActiveX-serverinä tai Exceliin upotettuna. Koska verkko on luonteeltaan "syntyvän hylyn määrää ennustava", niin sen käyttötapa voisi olla esimerkiksi sellainen, että erillinen optimointirutiini (mahdollisesti ExamiNeur-ohjelmisto) kysyisi neuroverkolta hylky-ennustetta erilaisissa pisteissä, ja lopulta tulostaisi parhaimman. Koska neuroverkko on luonteeltaan epälineaarinen, niin tämä optimointirutiinin tehtävä on kuitenkin huomatta-vasti haasteellisempi kuin lineaarisen funktion optimoinnissa – ongelman vaikeus riippuu oleellisesti opetetun verkon luonteesta, joka olisi täten syytä tutkia tarkkaan.

Jos verrataan neuroverkkoja lineaariseen malliin, niin neuroverkko tuo mukanaan epä-lineaarisuutta, joka auttaa mallin sovittamisessa. Toisaalta se tekee parhaiden alku-parametrien löytämisen huomattavasti vaikeammaksi. On myös vaikea tutkia, kuinka

hyvä verkosta tuli (muuten kuin kokeilemalla sitä uudella datalla) – lineaarisen mallin sovituksen yhteydessä saadaan aina myös selitysaste.

Neuroverkko voitaisiin myös opettaa alkuperäisten muuttujien sijaan käyttämällä esimerkiksi pääkomponenttianalyysissä saatuja "uusia muuttujia". Näin alkuperäinen suuriulotteinen ongelma redusoitaisiin pienempiulotteiseksi, ja yleensä ottaen tämä auttaa saamaan parempia tuloksia neuroverkoilla.

7.4 Bayes-verkot

7.4.1 Yleistä Bayes-verkoista

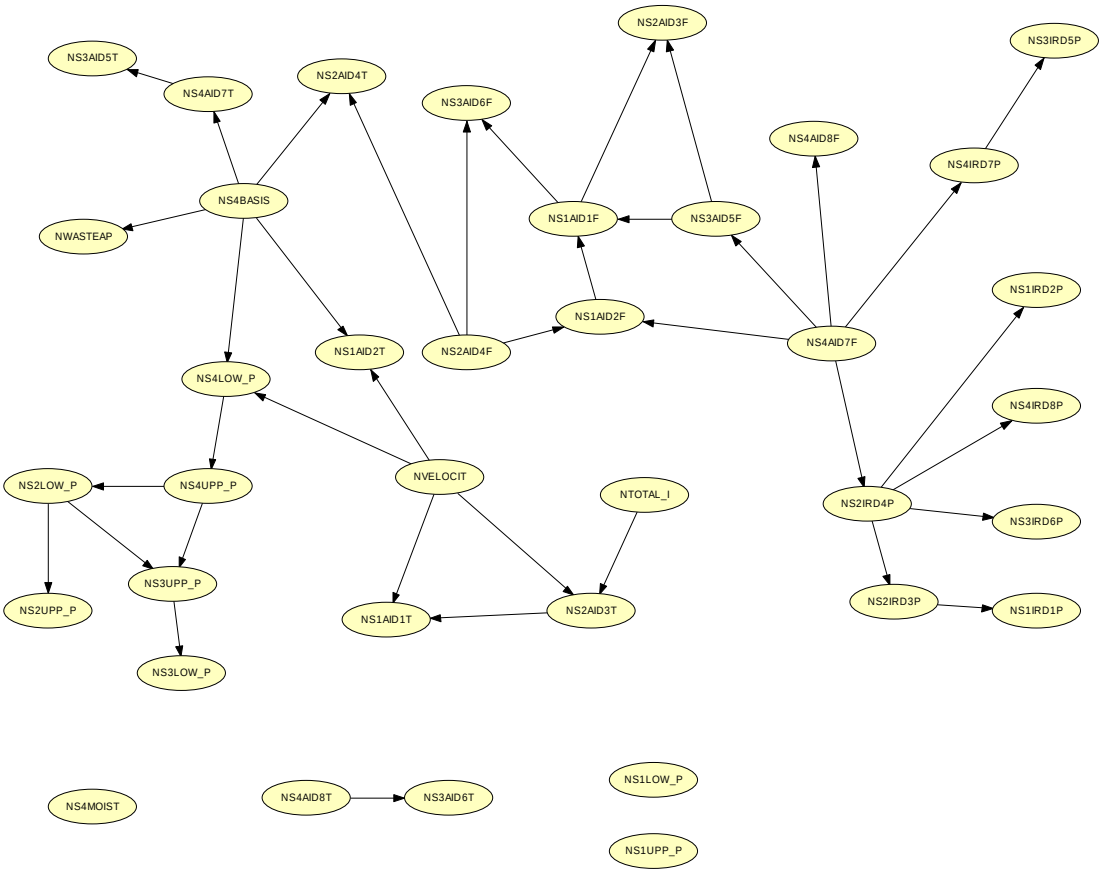
Bayes-verkoilla [Pea88] voidaan mallintaa muuttujien välisiä riippuvuuksia määrittelemällä mallin muuttujat suunnatun verkon solmuiksi ja näiden väliset mahdolliset riippuvuudet kvalitatiivisesti suunnatun verkon kaarien avulla. Kussakin solmussa määritellään paikallinen todennäköisyysjakauma vanhempiensa suhteen. Koko verkon yhteistodennäköisyysjakauma voidaan laskea näiden paikallisten todennäköisyyksien tulona. Marginalisoimalla yhteistodennäköisyysjakaumaa voidaan laskea muuttujien tai niiden ryhmien todennäköisyydet kaikilla mahdollisilla arvokombinaatioilla. Kokonaisen yhteistodennäköisyysjakauman ylläpitäminen ja sen käsittely on kuitenkin liian työlästä ja räjähtää nopeasti käsiin. Bayes-verkot tarjoavat keinoja esittää yhteisjakauma faktoroidusti ja ne tarjoavat tehokkaita päättelymenetelmiä havaintojen vaikutusten levittämiseen verkossa, jolloin voidaan laskea yksittäisen solmun todennäköisyysjakauma tai koko verkon tilojen todennäköisin konfiguraatio. Bayes-verkoissa ei ole erityisiä input ja output-muuttujia, kuten monissa hermoverkoissa, vaan mallin avulla voidaan laskea kaikkien muuttujien todennäköisyysjakaumat riippumatta tarvittavan päättelyn suunnasta. Useimmat nykyiset ratkaisumenetelmät on kehitetty verkoille, joissa käsitellään diskreettiarvoisia muuttujia. Jatkuva-arvoisten muuttujien käsittelyyn käytetään yleisesti normaalijakaumia tai näiden summia, mutta ei käytössämme olleilla työkaluilla.

Bayes-verkon kvalitatiivisen rakenteen määrittäminen on suhteellisen helppoa. Se voidaan tehdä graafisesti Bayes-verkkojen päättelytyökaluilla lisäämällä verkkoon solmuja ja määrittelemällä niille mahdolliset arvot. Suunnattuja kaaria lisäämällä määritellään kvalitatiivisia riippuvuuksia muuttujien välillä. Bayes-verkot toimivat parhaiten suhteellisen harvoilla verkoilla. Kytkentäasteen kasvaessa verkon faktorisoinnille jää vähemmän mahdollisuuksia ja tarvitaan parempia ratkaisumenetelmiä. Kun verkon kvalitatiivinen rakenne on saatu määritettyä, täytyy määritellä vielä kvantitatiivisesti paikalliset todennäköisyysriippuvuudet todennäköisyysmatriiseina (a priori todennäköisyydet juurisolmuille ja ehdolliset todennäköisyydet muille solmuille). Tämä vaihe on varsin työläs ja sitä tukemaan on kehitetty erilaisia työkaluja. Tässä työssä on käytetty hyväksi oppimismenetelmiä, jotka etsivät todennäköisimmät verkkomallit annetusta esimerkkiaineistosta ja estimoivat tarvittavat todennäköisyydet tästä esimerkkiaineistosta. Myös etukäteistietämystä muuttujien välisistä riippuvuuksista voidaan liittää mukaan. Olemme käyttäneet Belief Network PowerConstructor 2.0 –työkalua (BNPC2.0) Bayes-verkkojen rakenteen ja parametrien oppimiseen. On olemassa myös työkaluja parametrien oppimiseen esimerkeistä, kun verkon rakenne on anettu etukäteen.

BNPC2.0 [CBL98] tarkastelee havaintoavaruutta aluksi pareittaisten testien avulla ja arvioi informaatioteoreettisen mittarin avulla näiden muuttujien välistä riippuvuutta. Myös tilastollisia korrelaatiomittareita voidaan käyttää. Näin ollen tunnistetut paikalliset riippu-

vuudet heijastelevat lähinnä muuttujien välisiä lineaarisia riippuvuuksia. Epälineaarisia tai funktionaalisia riippuvuuksia ei pystytä välttämättä tunnistamaan.

7.4.2 Bayes-verkkokokeiluista aineistolla A



Kuva 7.5. Mitattujen muuttujien varaan rakennettu Bayes-verkko.

7.4.2.1 Koejärjestelyt

Bayes-verkkomallien oppimista varten jatkuva-arvoiset muuttujat jouduttiin diskretoimaan viiteen luokkaan siten, että kuhunkin luokkaan allokoitiin saman verran näytteitä (desiiliperuste). Oppiminen suoritettiin BNPC2.0 työkalulla, joka pystyi tunnistamaan suurimman osan verkon vaikutussuunnista, osa jouduttiin korjaamaan käsin verkkoeditorilla. Verkot talletettiin HUGIN-nimisen Bayes-verkko päättelytyökalun tiedostoformaatissa ja tuloksia tarkasteltiin tällä työkalulla. Kuvat on tuotettu Huginilla.

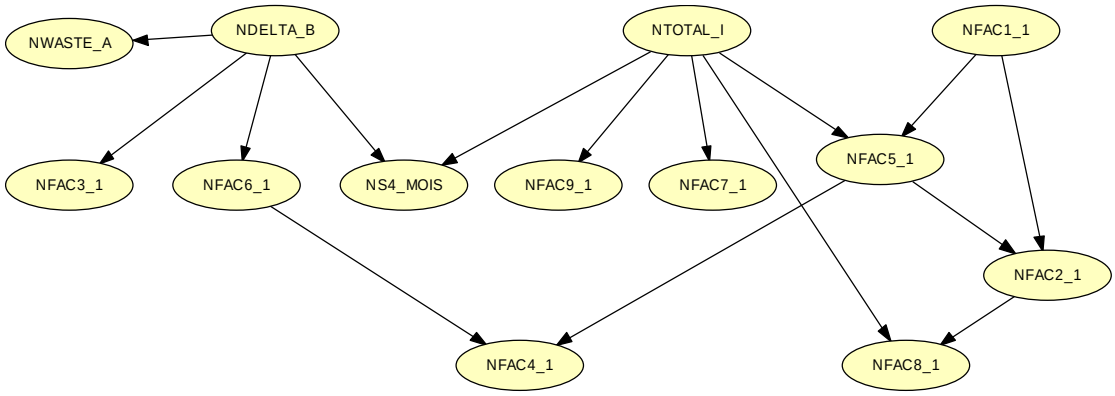
7.4.2.2 Mallinnus alkuperäisillä muuttujilla

Kokeilimme aluksi Bayes-verkon sovitusta suoraan mitattuihin aikasarjoihin ja saimme kuvan Kuva 7.5 mukaisen riippuvuuskaavion. Vasemmassa ylälaidassa näemme, että jätteen määrä (WASTE_APP) riippuu suoraan vain paperin neliöpainosta. Muiden muuttujien vuorovaikutuksia tarkasteltaessa havaitaan, että verkosta on havaittavissa samanlaisia riippuvuuksia kuin pääkomponenttianalyysillä ja korrelaatioanalyysillä: ilmamäärien mittaukset ovat läheisesti vuorovaikutuksessa keskenään ja myös infrakuivaimien tehot vuo-

rovaikuttavat toisiinsa. Kokonaisuudessaan voidaan todeta, että nykyisillä muuttujilla syntyvän hylyn määrää ei pystytä ennakoimaan kuin hyvin karkeasti tällä menetelmällä, sillä estimoidut todennäköisyysriippuvuudet tuottavat vain pieniä vaihteluita hylyn jakaumaan.

7.4.2.3 Mallinnus pääkomponenttianalyysillä tuotettujen muuttujien avulla

Toisessa testissä etsittiin ensin raakatietoaineiston mittausmuuttujien pääkomponentit pääkomponenttianalyysillä (ks. tämän raportin luku 4.1). Muuttujiin ei otettu mukaan seuraavia kenttiä: VELOCITY, WASTE_APP, TOTAL_IN. BNPC2.0 työkalulla laskettiin Bayes-verkkomalli saaduille pääkomponenttimuuttujille tarkoituksena mallintaa muuttujien välisiä riippuvuuksia.

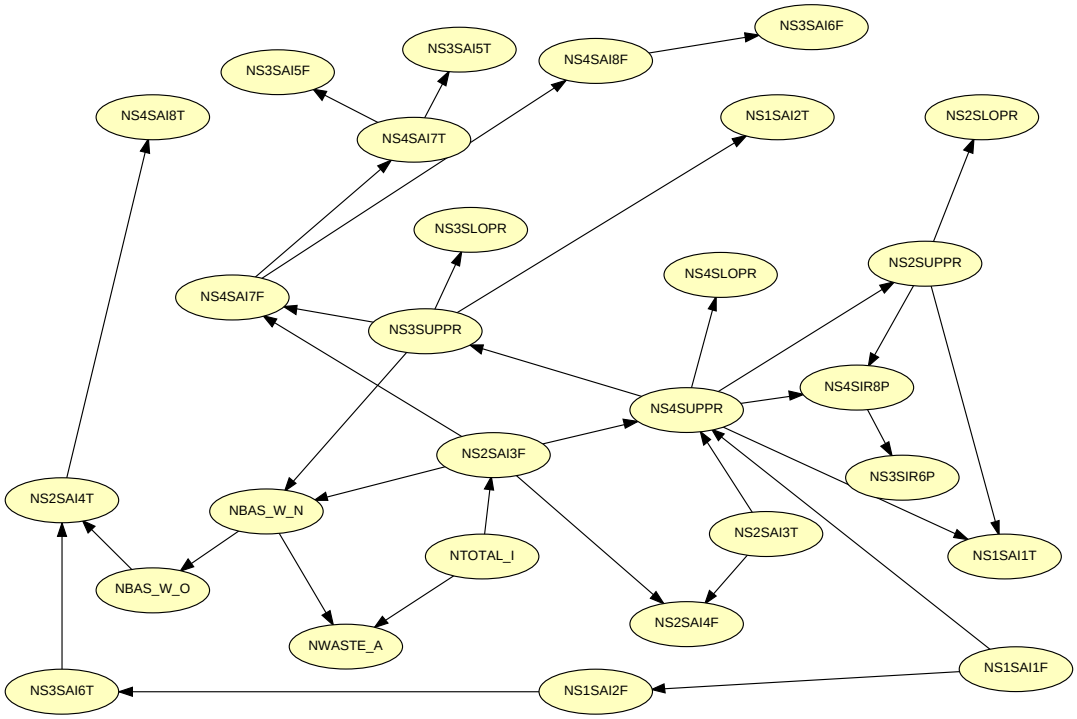


Kuva 7.6. Bayes-verkkomallin vaikutusgraafi valittujen muuttujien suhteen.

Neliöpainon muutos (NDELTA_B) ja katkon pituus (TOTAL_IN) mallinnettiin muista riippumattomiksi juurisolmuiksi ja esitettiin riippuvuudet niiden aiheuttamina. Syntyvän hylyn määrä (NWASTE_A) mallinnettiin lehtisolmuna, koska sen todennäköisyysjakaumaa on tarkoitus ennustaa muiden muuttujien suhteen. Kuvasta Kuva 7.6 nähdään, että hylyn määrä on suoraan riippuvainen vain neliöpainon muutoksesta (NDELTA_B). Todennäköisyysmatriisin mukaan jätteen määrä on keskimäärin alhaisempi, kun siirrytään ohuempaan paperiin. Pääkomponentit 3 (NFAC3_1) ja 6 (NFAC6_1) vaikuttavat myös epäsuorasti jätteen määrään, ellei neliöpainon muutosta ole annettuna.

Hugin-nimisellä Bayes-verkko -työkalulla kokeiltiin jakaumien muutoksia erilaisilla neliöpainon muutoksilla. Yleisesti voidaan todeta, että havaitut muutokset eivät ole kovin voimakkaita ja neliöpainon muutos peittää muiden muuttujien muutokset jätteen määrään, jos sen arvo tiedetään varmasti.

7.4.3 Bayes-verkkokokeiluista aineistolla B



Kuva 7.7. Mitattujen muuttujien varaan rakennettu Bayes-verkko aineistolla B.

7.4.3.1 Koejärjestelyt

Bayes-verkkomallien oppimista varten jatkuva-arvoiset muuttujat diskretoitiin viiteen luokkaan siten, että kuhunkin luokkaan allokoitiin saman verran näytteitä (desiiliperuste). Oppiminen suoritettiin BNPC2.0 työkalulla, joka pystyi tunnistamaan suurimman osan verkon vaikutussuunnista, osa jouduttiin korjaamaan käsin verkkoeditorilla. Verkot talletettiin HUGIN-nimisen Bayes-verkko päättelytyökalun tiedostoformaatissa ja tuloksia tarkasteltiin tällä työkalulla. Kuvat on tuotettu Huginilla.

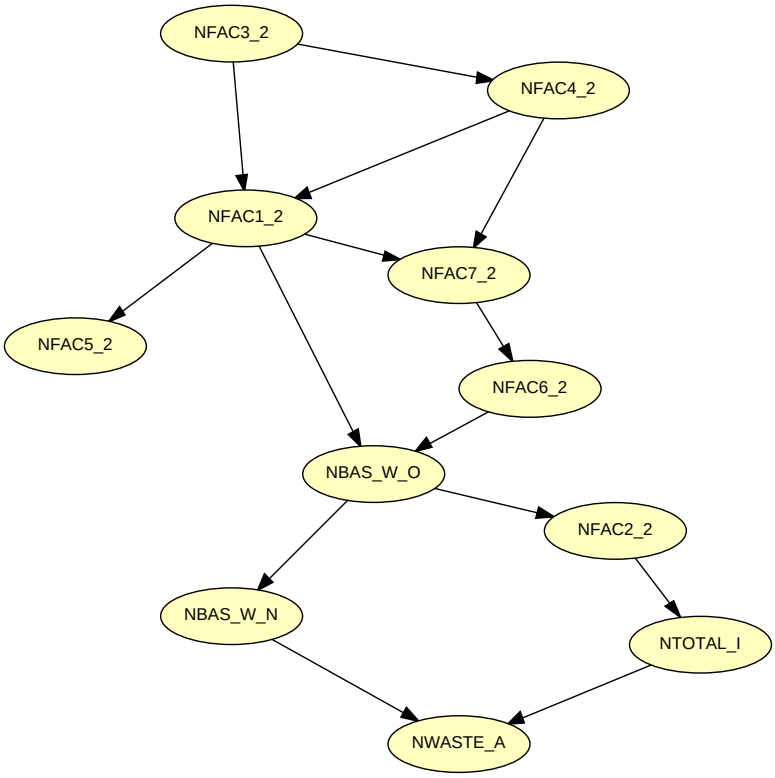
7.4.3.2 Mallinnus alkuperäisillä muuttujilla

Kokeilimme aluksi Bayes-verkon sovitusta suoraan mittausarvoilla ja saimme kuvan 7.3 mukaisen riippuvuuskaavion. Vasemmassa alalaidassa näemme, että jätteen määrä (NWASTE_A) riippuu suoraan vain seuraavan paperin neliöpainosta ja katkon pituudesta. Muiden muuttujien vuorovaikutuksia tarkasteltaessa havaitaan, että verkosta on havaittavissa samanlaisia riippuvuuksia kuin pääkomponenttianalyysillä ja korrelaatioanalyysillä: ilmamäärien mittaukset ovat läheisesti vuorovaikutuksessa keskenään ja myös infrakuivaimien tehot vuorovaikuttavat toisiinsa.

7.4.3.3 Mallinnus pääkomponenttianalyysillä tuotettujen muuttujien avulla

Toisessa testissä etsittiin ensin raakatietoaineiston mittausmuuttujien pääkomponentit pääkomponenttianalyysillä (ks. tämän raportin lukua 4.2). Pääkomponenttien lisäksi otettiin malliin mukaan ympäristömuuttujat (TOTAL_IN, BAS_W_NEW ja BAS_W_OLD)

sekä ennustettava tekijä WASTE_APP. Kaikki mittausarvot diskretoitiin SPSS-ohjelmalla viiteen luokkaan desiilimenetelmällä. BNPC2.0 työkalulla laskettiin Bayes-verkkomalli saaduille pääkomponenttimuuttujille tarkoituksena mallintaa muuttujien välisiä riippuvuuksia.



Kuva 7.8. Bayes-verkkomallin vaikutusgraafi valittujen muuttujien suhteen aineistolla B.

Syntyvän hylyn määrä (NWASTE_A) mallinnettiin lehtisolmuna, koska sen todennäköisyysjakaumaa on tarkoitus selittää muiden muuttujien arvoilla ja tähän suuntaan todennäköisyysmatriisin tulkinta on helpompaa. Kuvasta Kuva 7.64 nähdään, että hylyn määrä on suoraan riippuvainen vain tulevan paperilaadun neliöpainosta (NBAS_W_N) ja katkon pituudesta (NTOTAL_I). Tätä riippuvuutta kvantisoivan ehdollisen todennäköisyysmatriisin mukaan jätteen määrä on keskimäärin alhaisempi, kun päällystetään ohuempaa paperia ja kun katkon kesto on lyhyt. Päällystettäessä paksumpaa paperia ja katkon pituuden kasvaessa hylyn odotusarvo kasvaa. Uusi paperin neliöpaino on tilastollisesti riippuvainen aiemmasta (NBAS_W_O). Ehdollisesta todennäköisyysmatriisista nähdään, että yleensä pysytään samassa paperilaadussa, eikä kovin usein siirrytä ääripäiden välillä.

Hugin-nimisellä Bayes-verkko -työkalulla kokeiltiin jakaumien muutoksia erilaisilla ympäristömuuttujien arvoilla. Kyseisellä työkalulla pystyy havainnollisesti tarkastelemaan jakauman muutoksia eri tilanteissa ja mahdollisesti havaitsemaan ongelmatilanteita. Tilastolliset vaikutukset vaikuttavat selvemmiltä kuin aineistossa A.

7.4.4 Yhteenveto

Tässä luvussa oli aluksi lyhyt johdanto Bayes-verkkoihin ja sitten kuvattiin lyhyesti kahden testin tuloksia. Yhteenvetona voidaan todeta, että Bayes-verkkoja voidaan käyttää muuttujien välisen riippuvuusmallin oppimiseen annetusta esimerkkiaineistosta. Jo syntynyt graafinen, suunnattu verkkoesitys riippuvuuksista antaa havainnollisen kuvan riippuvuussuhteista. Estimoitujen todennäköisyysmatriisien avulla voidaan lisäksi suorittaa kvantitatiivisia What-If tarkasteluja opitulla mallilla asettamalla mallin muuttujille arvohavaintoja ja tarkastelemalla, miten kohdemuuttujien todennäköisyysjakaumat muuttuvat.

Aineiston B analysointimallien perusteella voidaan todeta, että sekä alkuperäisillä mittausarvoilla että pääkomponenteilla lasketut mallit tunnistavat hylyn selittäjiksi vain uuden paperilaadun neliöpainon ja katkon pituuden. Kun nämä tekijät tunnetaan, eivät muut arvot pääse vaikuttamaan syntyvän hylyn määrään. Tämän mallin mukaan vaikuttaa siis siltä, että ympäristömuuttujien käyttäytyminen vaikuttaa merkittävästi syntyvän hylyn määrään. Jos aineistoa olisi enemmän, olisi ollut mielenkiintoista vakioida ympäristömuuttujien arvot esimerkiksi tiettyyn paperilaatuun ja katkoalueeseen ja muodostaa malli sellaisessa pisteessä erikseen.

7.5 Muistiperustaisen päättelyn soveltaminen alkuparametrien valinnassa

Tässä tapaustutkimuksessa asiakkaan pääasiallisen kiinnostuksen kohteena on parantaa päällystyskoneen ylösajojen sujuvuutta, mikä ilmenee syntyvän hylyn määrän pienenemisenä. Ratkaisun ei välttämättä tarvitse olla optimaalinen, kunhan hylkymäärät saadaan keskimääräisesti pienenemään kokemuksen karttuessa. Parannusta uskotaan syntyvän asettamalla päällystyskoneelle ympäristöoloihin paremmin sopivat säätöparametrit. Merkittäviä ympäristömuuttujia ovat paperin seuraava ja mahdollisesti myös edellinen neliöpaino sekä katkon ajallinen kesto. Muistiperustaisen päättelyn soveltamiseen päädyttiin pohdittaessa, miten päätöspuiden tai muiden luokittimien tuottamia analyysituloksia voitaisiin hyödyntää valittaessa parhaita lähtöparametreja. Seuraavassa kuvataan mahdollisia ratkaisuvaihtoehtoja ja lopuksi kerrotaan näiden varaan rakennetun demonstraatio-ohjelmiston tuottamista tuloksista.

7.5.1 Muistiperustaisen päättelyn soveltamisvaihtoehdoista

Muistiperustainen päättely (MBR, memory-based reasoning) tarjoaa viitekehyksen yksinkertaiselle tapausperustaiselle oppimiselle, jossa tunnettuja tapauksia ei jalosteta eteenpäin tai tiivistetä mallimuotoon, vaan hyödynnetään jo kerättyjä tapauksia uuden ongelman ratkaisussa. Oletuksena on, että aiemmin hyviksi havaittuja ratkaisuja voidaan soveltaa myös uudessa tilanteessa. Tapausperustainen päättely (Case-Based Reasoning) on toinen termi samalle perusmenetelmälle. Siinä mennään pidemmälle ja käytetään edistyksellisempiä tapausten yhdistelymenetelmiä ja sitä on sovellettukin jopa suunnitteluongelmiin. MBR-menetelmää kutsutaan myös laiskaksi oppimiseksi (lazy learning), koska varsinaista mallinmuodostusvaihetta ei ole ja kaikki päättely tehdään vasta ongelmanratkaisuvaiheessa. Näin ollen suurille aineistoille MBR voi muodostua raskaaksi, ellei tapaustietokantaa ole organisoitu tehokkaasti. Menetelmää on kuitenkin sovellettu suuriin tapaustietokantoihin (1000-10000 tapausta).

MBR:ssä ongelman ratkaisu perustuu siihen, että tapaustietokannasta haetaan lähimmät tapaustiedot käyttämällä tapauskohtaisesti määriteltyä etäisyysmittaa ja saaduista ratkaisuista muodostetaan sitten uusi ehdotus jotakin tapauskohtaista menetelmää käyttäen. Etuna on, että perusmenetelmä on joustava tapausten esitystavan suhteen, mutta toisaalta ongelmien ratkaisu edellyttää ohjelmointia, jotta menetelmä voidaan sovittaa kuhunkin tapaukseen sopivaksi.

Tässä testitapauksessa esimerkkiaineisto on hyvin suppea (100-200 tapausta). Lähtöarvojen suosittelemiseksi voidaan tarjota parhaiten pärjänneen samantyyppisen esimerkkitapauksen ratkaisua, parhaiten pärjänneiden esimerkkitapausten kombinaatiota tai vaikkapa paikallisen regressiomallin tarjoamaa approksimaatiota annetuilla ympäristöparametreilla. Koska aineistoa ei juuri esikäsitellä, menetelmän luotettavuus (herkkyys esimerkkiaineiston kohinalle, virheille tai epänormaaleille havainnoille) saattaa tuottaa ongelmia, etenkin tarjottaessa vain parasta ratkaisua. Myös esimerkkiaineiston kattavuus voi olla riittämätön aineiston joissakin kohdissa. Tällöin ympäristörajoitteiden tarkka tyydyttäminen saattaa muodostua liian vaikeaksi käyttämättä approksimointeja.

Herkkyyttä aineiston poikkeamille voidaan helpottaa etsimällä aineistosta ryppäitä, jotka muistuttavat toisiaan parametriarvojensa suhteen (perinteiset klusterit) tai joissa hylkymäärät ovat keskimääräistä vähäisempiä ("hyvien tapausten klustereita" esimerkiksi luokittelun tuloksena). Ohjatun oppimisen menetelmillä, kuten päätöspuilla, voidaan rakentaa luokittimia, joilla aineisto voidaan jakaa hyviin ja huonoihin kohtuullisella tarkkuudella. Päätöspuiden lehtisolmut voidaan tulkita klustereiksi luokittelukriteerin suhteen. Hyvän klusterin ominaisuutena on riittävän suuri näytemäärä, pieni hylyn keskimääräinen määrä ja hylyn pieni varianssi. Syntynyttä tietoa hyvistä klustereista voidaan hyödyntää muistiperustaisessa päättelyssä valittaessa parhaimpia esimerkkitapauksia ratkaisun pohjaksi korostamalla luotettavan klusterin tapauksien hyvyyttä etäisyyttä laskettaessa. Myös interpolointi tai regressio tällaisen klusterin sisällä on perustellumpaa, koska kaikkien tapausten hylkymäärät ovat varsin tasalaatuisia, eikä poikkeamia juuri esiinny. Ongelmaksi muodostuu kuitenkin aineiston riittämättömyys.

7.5.2 Muistiperustaisen päättelyn kokeilutuloksia

Muistiperustaista päättelyä testattiin kehittämällä kolme samantyyppistä algoritmivaihtoehtoa ratkaisuvaihtoehdon valintaan perustuen olemassaolevaan tapaustietokantaan. Vertailua varten kehitettiin Matlabilla testiympäristö, jossa käyttäjä voi antaa haluamansa tavoitemuuttujan saraketunnisteen sekä rajoitemuuttujien tunnisteet ja tavoitearvon sekä painokertoimet kullekin tekijälle parhaan vaihtoehdon haluttavuuden laskemiseksi. Järjestelmä pyrkii sitten löytämään annettuja arvoja (käytetään aineiston mediaania, ellei tavoitearvoja ole annettu) parhaiten vastaavat suositukset pohjautuen aineistotietokantaan.

Yksinkertaisin vaihtoehto (menetelmä1) pohjautuu suoraan tapaustietokantaan ja valitsee K parhaiten nykytilaa vastaavaa vaihtoehtoa ja tarjoaa ratkaisuksi parasta näistä. Jonkinlainen yhdistely olisi myös mahdollista. Tällä hetkellä siis 1NN-ratkaisu (lähimmän naapurin menetelmä).

Toisessa vaihtoehdossa käytetään hyväksi luvussa 6.1.4 kuvatun päätöspuuanalyysin tuottamaa luokituspuuta ja tuotetaan siitä sääntötietokanta, jolla luokitellaan kukin esimerkkiaineiston tapaus kuuluvaksi johonkin päätöspuun lehtisolmuista. Ajatuksenamme on, että päätöspuun lehtisolmut sisältävät tuotetun hylkymäärän suhteen keskimääräistä homogeenisempia tapauksia. Näin ollen otaksumme, että etenkin paljon tapauksia sisältävä lehti-

solmu, jolla hylyn keskiarvo on pieni ja varianssikin pieni, soveltuu erityisen hyvin suosittelemaan alkuparametreja. Näin ollen aluksi arvioidaan kunkin lehtisolmuklusterin laatua arvofunktiolla. Vaihtoehdossa A (menetelmä2) valitaan parhaan lehtisolmun parhaat tapaukset ja keskiarvoistetaan niiden arvot, jolloin oletamme saavamme luotettavan pisteen ratkaisuksi. Toinen vaihtoehto olisi virittää paikallinen regressiomalli kyseisen solmun tapausten varaan, mutta näytteiden pieni määrä ei sallinut tätä. Vaihtoehdossa B (menetelmä3) parhaiden klusterien sisältämät esimerkkitapaukset arvotetaan etäisyysfunktiolla ja valitaan paras vaihtoehto ratkaisuksi ottaen arvotuksessa huomioon klusterin laatuarvio.

Nykyinen demonstraatiojärjestelmä tarjoaa käyttäjälle ratkaisuvaihtoehdot kunkin edellä kuvatun menetelmän avulla tuotettuna.

Demonstraatiojärjestelmän ratkaisuja on testattu varsinaisella aineistolla ja synteettisellä testiaineistolla, jolloin saatiin seuraavanlaisia tuloksia:

Järjestelmää testattiin varsinaisella aineistolla syöttämällä sille satunnaisia ympäristöparametreja 1000 kappaletta ja tuottamalla alkuparametriehdotukset annetuille tilanteille. Uuden neliöpainon arvot valittiin tasajakaumalla väliltä 65-105 g/m^2 ja katkon pituus väliltä 2700-7700 sekuntia. Vanhaa neliöpainoa ei huomioitu. Sekä tavoitemuuttujalle että rajoitteille sovellettiin samaa painoarvoa.

Yksinkertaisin menetelmä 1 tuotti keskimäärin parhaan tuloksen hylkymäärän suhteen (μ=7.82, δ=0.33, min=7.49, max=8.86). Uuden neliöpainon suhteen suhteellinen virhe tavoitteeseen oli (μ=6,6%, δ=5,1%, min=0%, max=32%). Katkon pituuden suhteen suhteellinen virhe tavoitearvoon oli (μ=6,5%, δ=5,1%, min=0%, max=30,4%).

Menetelmä 3 pärjäsi melkein yhtä hyvin hylkymäärän suhteen (μ=7.87, δ=0.37, min=7.49, max=8.86). Uuden neliöpainon suhteen suhteellinen virhe tavoitteeseen oli (μ=6,4%, δ=5,7%, min=0%, max=31%). Katkon pituuden suhteen suhteellinen virhe tavoitearvoon oli (μ=6,2%, δ=5,3%, min=0%, max=33,6 %).

Menetelmä 2 pärjäsi huonommin hylkymäärän suhteen, mutta kaikki tarjotut ehdotukset ovat kuitenkin edellisten tulosten vaihteluvälin sisällä (μ=8,71, δ=0.03, min=8,64, max=8,74). Uuden neliöpainon suhteen suhteellinen virhe tavoitteeseen oli (μ=12,2%, δ=6,9%, min=0%, max=25%). Katkon pituuden suhteen suhteellinen virhe tavoitearvoon oli (μ=17,3%, δ=16,4%, min=0.1%, max=77%).

Koko aineistossa alkuparametrien %-vaihtelu mediaaninsa ympärillä on seuraavanlainen muuttujille 1-24 sarakkeissa:

20.5882 38.9800 6.6667 12.5000 20.0000 48.5126 14.2857

13.3333 87.5000 116.6667 8.8235 14.7059 41.2758 11.6667

14.7059 62.5000 55.5556 11.7647 18.7500 41.6667 6.6667

15.6250 73.3333 84.6154

Menetelmällä 2 tuotettujen alkuparametriehdotusten vaihtelu on hyvin pientä. Sarakkeiden 1-24 muuttujien %-vaihtelu mediaaninsa ympärillä 1000 testitapauksella on seuraavalainen:

4.9029 2.1062 0.0000 0.0000 4.9029 5.4099 1.9029

1.9029 0.0000 0.0000 0.0000 0.0000 2.6230 0.0000

1.3746 0.0000 0.0000 0.0000 0.0000 2.2052 0.0000

0.6953 0.0000 0.0000

Menetelmillä 1 ja 3 tuotettujen alkuparametriehdotusten vaihteluvälit ovat selvästi edellistä suuremmat ja suunnilleen samansuuruiset keskenään. Ne ovat kuitenkin suppeammat kuin opetusaineistossa. Menetelmällä 1 sarakkeiden 1-24 muuttujien %-vaihtelu mediaaninsa ympärillä 1000 testitapauksella on seuraavanlainen:

9.3750 36.6782 7.1429 12.5000 13.3333 38.3858 14.2857

12.500 70.0000 87.5000 7.5000 16.6667 40.7407 10.0000

11.429 50.0000 45.0000 11.7647 20.0000 39.1304 7.1429

11.765 55.5556 62.5000

Menetelmällä 3 sarakkeiden 1-24 muuttujien %-vaihtelu mediaaninsa ympärillä 1000 testitapauksella on seuraavanlainen:

21.8750 36.6782 7.1429 12.5000 13.3333 38.3858 14.2857

12.5000 70.0000 87.5000 7.5000 16.6667 40.7407 10.0000

11.4286 50.0000 45.0000 11.7647 20.0000 40.4494 7.1429

11.7647 55.5556 62.5000

Luvussa 8 on kuvattu saavutettuja testituloksia synteettisellä aineistolla ja suoritettu vertailua lineaarimalleihin.

7.5.3 Yhteenveto

Muistiperustainen päättely toimi yllättävän hyvin alkuparametrien valintaan annetulla aineistolla ja synteettisellä testiaineistolla huolimatta aineiston laajuudesta. Menetelmän etuna on helppo toteutettavuus ja tapaustietokannan laajennettavuus. Paljon työtä tarvittaisiin kuitenkin luotettavuutta parantavien tarkistuspiirteiden kehittämiseen. Käytännön toimivuuden varmistaminen edellyttäisi kuitenkin käytännön testiajoja varsinaisella laitteistolla.

Toteutettujen vaihtoehtojen välillä ei juuri havaittu eroja, joten päätöspuun luokitustiedon hyödyntämisellä ei voitu osoittaa olevan vaikutusta saavutettuihin ratkaisuihin näin suppealla aineistolla. Aineistoa tarvittaisiin todella paljon enemmän (>10-kertaisesti)

8 Menetelmien ja tulosten arvointi

Tulosten arvioinnissa on kolme vaihetta:

1. Verrata menetelmien käyttäytymistä asiakkaalta saadulla datalla.
2. Verrata menetelmien käyttäytymistä keinotekoisella datalla.
3. Tutkia päällystyskoneen käyttäytymistä tulostemme pohjalta.

Kahta ensimmäistä vaihetta tarkastellaan seuraavassa, ja kolmatta vaihetta luvussa 9.

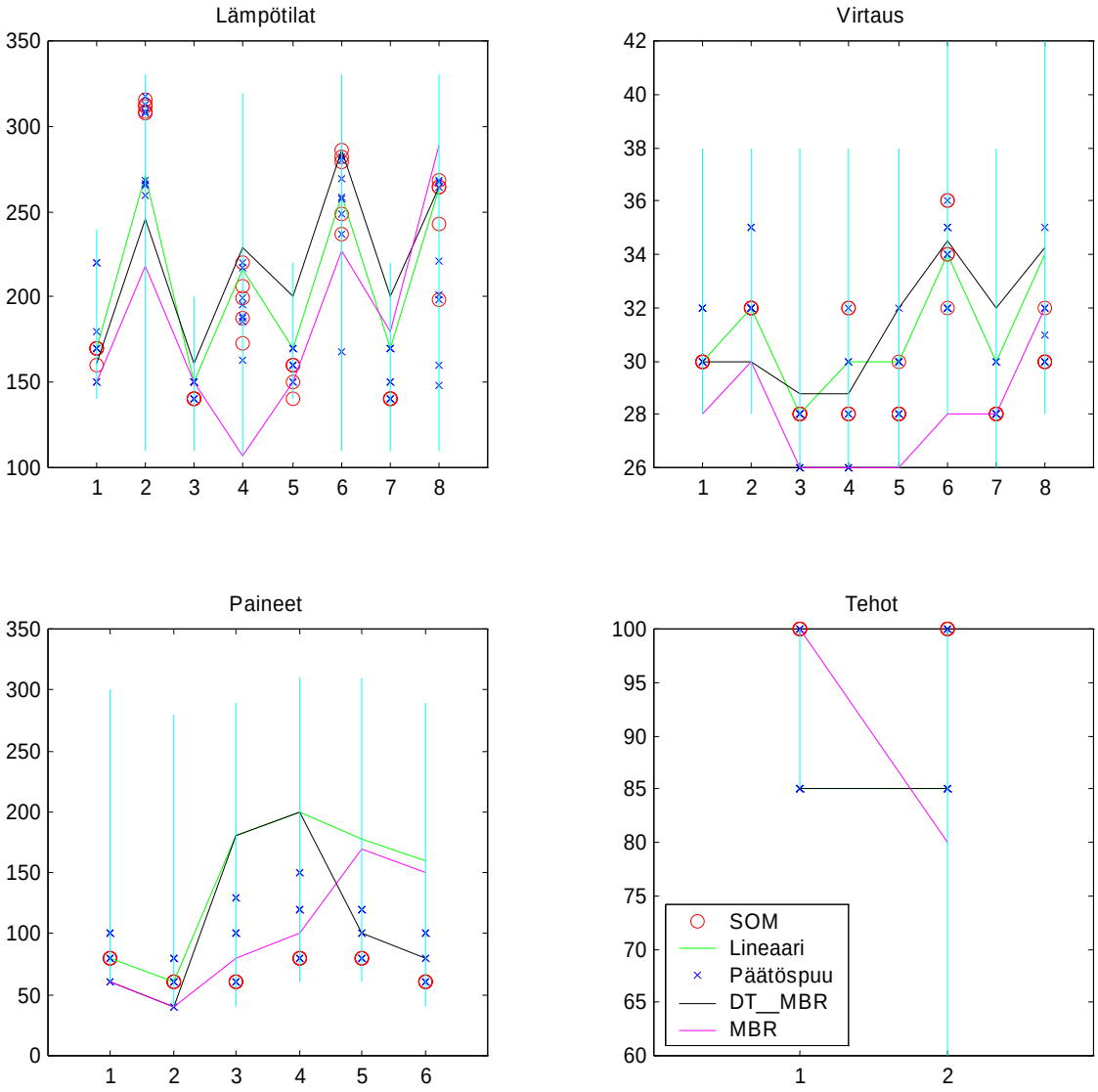
8.1 Menetelmien vertailu, aineisto B

Kuva 8.1 esittää eri menetelmien palauttamien lähtöarvojen eroja. Lämpötilojen- ja virtausten kuvissa x-akselin numero kertoo suoraan kuivaimen numeron, jolta arvo on mitattu. Paineiden kuvassa on asemien 2 – 4 lähtöarvoja alkaen 2. aseman yläsylinterin paineesta. ja päättyen 4. aseman alasylinterin paineseen.Tehojen kuvassa asemat ovat 6 ja 8. Käytetyt rajoitteet ovat neliöpainolle ennen ja jälkeen katkoa 71, ja katkon pituudelle n. 60 minuuttia. Punaiset pallot kuvaavat hyvässä SOM-klusterissa olevia kaikkia havaintopisteitä. Vihreä viiva kuvaa lineaarimallin palauttamaa pistettä. Rasti kuvaa kaikkia päätöspuun hyvissä solmuissa olevia havaintoja, joissa ollaan suhteellisen lähellä rajoitteita. Tumma viiva kuvaa "Decision Tree –Memory Based Reasoning"-menetelmää, jossa on muodostettu vastauspiste keskiarvoistamalla, ja punainen viiva "Memory Based Reasoning"-menetelmää, jossa ei ole keskiarvoistettu.

Kuvasta nähdään, että menetelmät antavat suhteellisen yhdenmukaisia vastauksia. Eniten vaihtelua on paineissa. Lineaarimallilla ei ole taipumusta mennä muuttujien vaihteluvälien reuna-alueille, toisin kuin monilla muilla menetelmillä. Reuna-alueiden ehdottaminen saattaa merkitä myös sitä, että vastaukset paranisivat muuttujien arvovälejä kasvattamalla.

8.2 Menetelmien vertailu, keinotekoinen aineisto

Menetelmien vertailussa alkuperäisellä aineistolla on se ongelma, että emme voi ilman kokeita tietää, kuinka hyvin päällystyskone toimisi ehdotetuilla parametreilla. Tämän vuoksi vertailimme kahta menetelmää myös keinotekoisella aineistolla. Muodostimme epälineaarisen ja stokastisen testifunktion, ja laskimme sen arvot kaikissa aineiston B pisteissä. Näin "testiaineiston pisteiden jakauma" oli täsmälleen sama kuin alkuperäisessä ongelmassa. Testifunktiossa oli kaksi epästokastista termiä: kaikkien muuttujien neliöiden summa, sekä kolmen muuttujan tulon sini, painotettuna siten, että näiden termien arvot olivat samaa suuruusluokkaa. Lisäksi stokastinen virhetermi oli välillä +- 3. Koko aineistoon sovellettuna tämän testifunktion arvot olivat välillä 4,9 – 37,4 – on kuitenkin huomattava, että testifunktion arvojen jakauma ei enää noudattanut alkuperäisen aineiston jakaumaa. Tämän jälkeen sovelsimme sekä lineaarisia malleja että luvun 7.5 (muistiperustainen päättely) menetelmiä. Sitten vertailimme testifunktion arvoa mallin palauttamassa pisteessä ilman stokastista virhetermiä. Neliöpainoina ennen ja jälkeen katkon sekä katkon pituutena käytettiin aineiston mediaaneja. Taulukko 8.1 kuvaa tulokset.



Kuva 8.1. Esimerkki eri menetelmien ehdottamista parhaista lähtöarvoista.

Menetelmä	Testifunktion arvo vastauksessa
Lineaarinen malli kaikille pisteille	11.3
Lineaarinen malli "hyville" pisteille (50% huonoimmista poistettu)	7.62
Luvun 7.5 menetelmä 1	5.94
Luvun 7.5 menetelmä 2 A	5.21
Luvun 7.5 menetelmä 2 B	5.21

Taulukko 8.1. Testitulokset keinotekoisella aineistolla.

Luvun 7.5 (muistiperustainen päättely) menetelmät toimivat selvästi parhaiten. Lineaarimalli toimii sitä paremmin mitä enemmän aineistosta poistetaan huonoja ylösajoja, mutta aineistoa ei ole tarpeeksi enempään rajaamiseen.

Painoitamme, että tämä testi kertoo ainoastaan että menetelmämme toimivat ainakin joskus. Niistä ei voi vetää johtopäätöstä, että ne toimivat päällystyskoneen tapauksessa, tai että kumpi menetelmä on siinä parempi. Tähän tarvitaan testausvaihe itse päällystyskoneella. Uskoa menetelmien soveltamiseen myös päällystyskoneeseen ne kuitenkin vahvistavat.

9 Toteutusehdotukset

Taulukko 9.2 näytää mitä lähtöarvoja luvun 7.5 menetelmä 1 (MBR) ja luvun 7.1 lineaarimallit ehdottavat käytettäväksi eri neliöpainoilla ja katkon pituuksilla silloin, kun neliöpaino säilyy samana katkon yli. Katkon pituus TOT_INTL on minuutteina. Lineaarimalli ehdottaa samoja lähtöarvoja näihin kaikkiin tilanteisiin. Taulukko 9.3 näyttää lähtöarvot myös silloin, kun katkon jälkeinen neliöpaino (BAS_W_NEW) on eri kuin katkoa edeltävä neliöpaino (BAS_W_OLD).

Seuraavassa taulukossa (Taulukko 9.1) esitetään vakioiksi oletettujen muuttujien arvot.

Asema 1	Asema 2	Asema 3	Asema 4
S1S_IRDR1_P=100 S1S_IRDR2_P=60 S1S_UPP_PRE=0 S1S_LOW_PRE=20 S1S_TLOAD_PRE =100	S2S_IRDR3_P=100 S2S_IRDR4_P=60 S2S_TLOAD_PRE =100	S3S_IRDR5_P=100	S4S_IRDR7_P=100

Taulukko 9.1. Vakiomuuttujat ja niiden oletetut arvot.

MBR-lähtöarvot:

BAS_W_NEW	TOT_INTL	S1S_AIDR1_T	S1S_AIDR2_T	S1S_AIDR1_F	S1S_AIDR2_F	S2S_AIDR3_T	S2S_AIDR4_T	S2S_AIDR3_F	S2S_AIDR4_F	S2S_UPP_PRE	S2S_LOW_PRE	S3S_IRDR6_P	S3S_AIDR5_T	S3S_AIDR6_T	S3S_AIDR5_F	S3S_AIDR6_F	S3S_UPP_PRE	S3S_LOW_PRE	S4S_IRDR8_P	S4S_AIDR7_T	S4S_AIDR8_T	S4S_AIDR7_F	S4S_AIDR8_F	S4S_UPP_PRE	S4S_LOW_PRE
70	50	170	312	30	32	140	173	28	32	80	60	100	150	286	28	36	60	80	100	140	264	28	30	80	60
70	90	150	182	28	30	150	110	26	26	60	40	100	150	161	26	28	80	100	80	180	191	28	32	170	150
70	130	170	320	32	32	150	196	26	26	100	80	85	170	277	28	30	100	120	85	170	259	28	30	120	100
80	50	170	312	30	32	140	173	28	32	80	60	100	150	286	28	36	60	80	100	140	264	28	30	80	60
80	90	170	273	30	32	150	296	28	32	80	60	85	150	292	30	36	180	200	85	150	289	30	36	180	160
80	130	160	252	30	32	180	174	30	30	100	80	85	180	330	32	34	180	200	85	180	261	32	32	180	160
90	50	170	312	30	32	140	173	28	32	80	60	100	150	286	28	36	60	80	100	140	264	28	30	80	60
90	90	150	295	28	32	150	282	28	34	100	80	100	150	306	30	36	180	200	80	150	276	28	36	200	180
90	130	150	295	28	32	150	282	28	34	100	80	100	150	306	30	36	180	200	80	150	276	28	36	200	180

Lineaarimallin lähtöarvot:

170	271	30	32	150	217	28	30	80	60	85	170	260	30	34	180	200	85	170	264	30	34	179	160		

Taulukko 9.2. Ehdotetut lähtöarvot neliöpainon säilyessä katkon yli samana.

MBR-lähtöarvot:

BAS_W_NEW	BAS_W_OLD	TOT_INTL	S1S_AIDR1_T	S1S_AIDR2_T	S1S_AIDR1_F	S1S_AIDR2_F	S2S_AIDR3_T	S2S_AIDR4_T	S2S_AIDR3_F	S2S_AIDR4_F	S2S_UPP_PRE	S2S_LOW_PRE	S3S_IRDR6_P	S3S_AIDR5_T	S3S_AIDR6_T	S3S_AIDR5_F	S3S_AIDR6_F	S3S_UPP_PRE	S3S_LOW_PRE	S4S_IRDR8_P	S4S_AIDR7_T	S4S_AIDR8_T	S4S_AIDR7_F	S4S_AIDR8_F	S4S_UPP_PRE	S4S_LOW_PRE
70	70	50	170	312	30	32	140	173	28	32	80	60	100	150	286	28	36	60	80	100	140	264	28	30	80	60
70	70	90	150	182	28	30	150	110	26	26	60	40	100	150	161	26	28	80	100	80	180	191	28	32	170	150
70	70	130	170	320	32	32	150	196	26	26	100	80	85	170	277	28	30	100	120	85	170	259	28	30	120	100
70	80	50	170	312	30	32	140	173	28	32	80	60	100	150	286	28	36	60	80	100	140	264	28	30	80	60
70	80	90	150	260	30	35	150	166	28	30	60	40	85	150	213	30	35	130	150	85	150	152	30	35	100	80
70	80	130	170	320	32	32	150	196	26	26	100	80	85	170	277	28	30	100	120	85	170	259	28	30	120	100
70	90	50	170	312	30	32	140	173	28	32	80	60	100	150	286	28	36	60	80	100	140	264	28	30	80	60
70	90	90	150	260	30	35	150	166	28	30	60	40	85	150	213	30	35	130	150	85	150	152	30	35	100	80
70	90	130	170	320	32	32	150	196	26	26	100	80	85	170	277	28	30	100	120	85	170	259	28	30	120	100
80	70	50	170	312	30	32	140	173	28	32	80	60	100	150	286	28	36	60	80	100	140	264	28	30	80	60
80	70	90	170	273	30	32	150	296	28	32	80	60	85	150	292	30	36	180	200	85	150	289	30	36	180	160
80	70	130	160	252	30	32	180	174	30	30	100	80	85	180	330	32	34	180	200	85	180	261	32	32	180	160
80	80	50	170	312	30	32	140	173	28	32	80	60	100	150	286	28	36	60	80	100	140	264	28	30	80	60
80	80	90	170	273	30	32	150	296	28	32	80	60	85	150	292	30	36	180	200	85	150	289	30	36	180	160
80	80	130	160	252	30	32	180	174	30	30	100	80	85	180	330	32	34	180	200	85	180	261	32	32	180	160
80	90	50	170	312	30	32	140	173	28	32	80	60	100	150	286	28	36	60	80	100	140	264	28	30	80	60
80	90	90	170	273	30	32	150	296	28	32	80	60	85	150	292	30	36	180	200	85	150	289	30	36	180	160
80	90	130	160	252	30	32	180	174	30	30	100	80	85	180	330	32	34	180	200	85	180	261	32	32	180	160
90	70	50	170	312	30	32	140	173	28	32	80	60	100	150	286	28	36	60	80	100	140	264	28	30	80	60
90	70	90	170	273	30	32	150	296	28	32	80	60	85	150	292	30	36	180	200	85	150	289	30	36	180	160
90	70	130	150	295	28	32	150	282	28	34	100	80	100	150	306	30	36	180	200	80	150	276	28	36	200	180
90	80	50	170	312	30	32	140	173	28	32	80	60	100	150	286	28	36	60	80	100	140	264	28	30	80	60
90	80	90	170	273	30	32	150	296	28	32	80	60	85	150	292	30	36	180	200	85	150	289	30	36	180	160
90	80	130	150	295	28	32	150	282	28	34	100	80	100	150	306	30	36	180	200	80	150	276	28	36	200	180
90	90	50	170	312	30	32	140	173	28	32	80	60	100	150	286	28	36	60	80	100	140	264	28	30	80	60
90	90	90	150	295	28	32	150	282	28	34	100	80	100	150	306	30	36	180	200	80	150	276	28	36	200	180
90	90	130	150	295	28	32	150	282	28	34	100	80	100	150	306	30	36	180	200	80	150	276	28	36	200	180

Lineaarimallin lähtöarvot:

170	271	30	32	150	217	28	30	80	60	85	170	260	30	34	180	200	85	170	264	30	34	179	160		

Taulukko 9.3. Ehdotetut lähtöarvot kaikille tapauksille.

Toteutusehdotuksemme on, että ensimmäisessä vaiheessa taulukoissa olevia lähtöarvoja kokeillaan päällystyskoneella, keräten tietoa syntyvän hylyn määrästä. Jos tämä vaihe tuottaa päätöksen jatkaa panostusta uuden projektin muodossa, niin silloin työvaiheet olisivat:

1. Lähtöarvojen mittaushetken validointi – lisätutkiminen, ottaako säätöjärjestelmä todellakin lähtöarvot hetkenä jona oletamme.

2. Automaattisen liitännän toteuttaminen prosessinohjausjärjestelmään tiedon keruuta ja syöttämistä varten (tiedon syöttö alkuvaiheessa vain ihmisen varmistuksen kautta).

3. Automaattisesti laskettavan syntyvän hylyn määrän (kuten luvussa 3 on kuvattu) hyödyntämisen tutkiminen esimerkiksi paperirullien merkkauksessa.

4. Adaptiivisuuden tutkiminen – olisiko tarvetta tehdä järjestelmästä automaattisesti sopeutuva prosessin muutoksiin siten, että se osaisi aina uusissa tilanteissa hakea parhaat lähtöarvot.

5. Vastaavan tutkimuksen tekeminen myös jollekin muulle kuin päällystyskoneelle.

Arviomme on, että tämän dokumentin tuloksien perusteella voimme toteuttaa todennäköisesti parannusta tuottavan järjestelmän.

Lähdeluettelo

[ASU86] A. Aho, R. Sethi, J. Ullman. Compilers – Principles, Techniques and Tools. Addison-Wesley, 1986.

[Bish95] C. Bishop. Neural Networks for Pattern Recognition, Oxford University Press, 1995.

[BFOS84] L. Breiman, J. Friedman, R. Olsen, C. Stone. Classification and regression trees. Belmont, California, 1984.

[BVS91] D. Biggs, B. de Ville, E. Suen. A method of choosing multiway partitions for classification and regression trees. Journal of applied statistics, 18, 1991, pp. 49-62.

[CBL98] J. Cheng, DA. Bell, W. Liu. Learning Bayesian Networks from Data: an Efficient Approach Based on Information Theory. A technical report, pp. 1-41, 1998.

[Cha93] B. Chazelle. An optimal convex hull algorithm in any fixed dimension. Discrete and Computational Geometry, 10, 1993, pp.377—409.

[Coh95] Paul R. Cohen: Empirical Methods for Artificial Intelligence. The MIT Press, Cambridge, 1995

[CS96] P. Cheeseman, J. Stutz. Bayesian Classification (AutoClass): Theory and results, in U. Fayyad. Advances in Knowledge Discovery and Data Mining, 1996.

[Eve77] B. Everitt. Cluster Analysis. Heinemann Educational Books Ltd, London, Great Britain, 1977.

[Har75] J. Hartigan. Clustering Algorithms. John Wiley & Sons, New York, USA, 1975.

[Koh95] T. Kohonen. Self-Organising Maps. Springer-Verlag, Berlin, 1995.

[MiA95] J. Milton, J. Arnold. Introduction to Probability and Statistics – Third Edition. McGraw-Hill, Singapore, 1995.

[Pea88] J. Pearl, Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, 1988.

[Rik79] H. Rikkonen. Matematiikan pitkä peruskurssi I – vektorialgebra ja analyyttinen geometria. Otakustantamo, Espoo, 1979.

[Sha96] S. Sharma. Applied Multivariate Techniques. John Wiley & Sons, New York, 1996.

[SPSS98] Answer tree 2.0 user's guide. SPSS Inc, USA, 1998.

[Ves99] J. Vesanto. SOM-based data visualization methods. Intelligent Data Analysis, 3, 1999, pp. 111-126.

LIITE A: Prosessitietojen kuvaus

A1. Taulu RAW_RECDS – jatkuvat prosessimittaukset

Positio nimi	Sijainti	Sarakkeen nimi	Engl.kielinen nimi		
		REC_ID	Record identifier		
		TS	Timestamp		
Koneen nopeus	PPK3	VELOCITY	Machine velocity		
Kosteus	Aukirullaus	UNW_MOIST	Unwind moisture		
Neliöpaino	Aukirullaus	UNW_BAS_W	Unwind basis weigth		
Päälystemäärä	1.asema	S1_COAT_W	Coating section 1:	coat weight	
Kosteus	1.asema	S1_MOIST		moisture	
Neliöpaino	1.asema	S1_BASIS_W		basis weigth	
Päälystemäärä	2.asema	S2_COAT_W	Coating section 2:	coat weight	
Kosteus	2.asema	S2_MOIST		moisture	
Neliöpaino	2.asema	S2_BASIS_W		basis weigth	
Päällystemäärä	3.asema	S3_COAT_W	Coating section 3:	coat weight	
Kosteus	3.asema	S3_MOIST		moisture	
Neliöpaino	3.asema	S3_BASIS_W		basis weigth	
Päälystemäärä	4.asema	S4_COAT_W	Coating section 4:	coat weight	
Kosteus	4.asema	S4_MOIST		moisture	
Neliöpaino	4.asema	S4_BASIS_W		basis weigth	
Kireys	Aukirullaus	UNW_TENS	Unwind web tension		
Infrakuivain 1 teho	1.asema	S1_IRDR1_P	Coating section 1:	IR dryer 1:	power
Infrakuivain 2 teho (profiloiva)	1.asema	S1_IRDR2_P		IR dryer 2:	power
Leijukuivain 1 teho	1.asema	S1_AIDR1_P		air dryer 1:	power
Leijukuivain 1 lämpötila	1.asema	S1_AIDR1_T			temperature
Leijukuivain 1 kuivatusilman määrä	1.asema	S1_AIDR1_F			air flow
Leijukuivain 2 teho	1.asema	S1_AIDR2_P		air dryer 2:	power
Leijukuivain 2 lämpötila	1.asema	S1_AIDR2_T			temperature
Leijukuivain 2 kuivatusilman määrä	1.asema	S1_AIDR2_F			air flow
1. Ryhmän paine yläsäädin	1.asema	S1_UPP_PRE		steam press.:	upper cylinder
1. Ryhmän paine alasäädin	1.asema	S1_LOW_PRE			lower cylinder
Infrakuivain 3 teho	2.asema	S2_IRDR3_P	Coating section 2:	IR dryer 3:	power
Infrakuivain 4 teho (profiloiva)	2.asema	S2_IRDR4_P		IR dryer 4:	power
Leijukuivain 3 teho	2.asema	S2_AIDR3_P		air dryer 3:	power
Leijukuivain 3 lämpötila	2.asema	S2_AIDR3_T			temperature
Leijukuivain 3 kuivatusilman määrä	2.asema	S2_AIDR3_F			air flow
Leijukuivain 4 teho	2.asema	S2_AIDR4_P		air dryer 4:	power
Leijukuivain 4 lämpötila	2.asema	S2_AIDR4_T			temperature
Leijukuivain 4 kuivatusilman määrä	2.asema	S2_AIDR4_F			air flow
2. Ryhmän paine yläsäädin	2.asema	S2_UPP_PRE		steam press.:	upper cylinder
2. Ryhmän paine alasäädin	2.asema	S2_LOW_PRE			lower cylinder
Infrakuivain 5 teho	3.asema	S3_IRDR5_P	Coating section 3:	IR dryer 5:	power
Infrakuivain 6 teho (profiloiva)	3.asema	S3_IRDR6_P		IR dryer 6:	power

Leijukuivain 5 teho	3.asema	S3_AIDR5_P		air dryer 5:	power
Leijukuivain 5 lämpötila	3.asema	S3_AIDR5_T			temperature
Leijukuivain 5 kuivatusilman määrä	3.asema	S3_AIDR5_F			air flow
Leijukuivain 6 teho	3.asema	S3_AIDR6_P		air dryer 6:	power
Leijukuivain 6 lämpötila	3.asema	S3_AIDR6_T			temperature
Leijukuivain 6 kuivatusilman määrä	3.asema	S3_AIDR6_F			air flow
3. Ryhmän paine yläsäädin	3.asema	S3_UPP_PRE		steam press.:	upper cylinder
3. Ryhmän paine alasäädin	3.asema	S3_LOW_PRE			lower cylinder
Infrakuivain 7 teho	4.asema	S4_IRDR7_P	Coating section 4:	IR dryer 7:	power
Infrakuivain 8 teho (profiloiva)	4.asema	S4_IRDR8_P		IR dryer 8:	power
Leijukuivain 7 teho	4.asema	S4_AIDR7_P		air dryer 7:	power
Leijukuivain 7 lämpötila	4.asema	S4_AIDR7_T			temperature
Leijukuivain 7 kuivatusilman määrä	4.asema	S4_AIDR7_F			air flow
Leijukuivain 8 teho	4.asema	S4_AIDR8_P		air dryer 8:	power
Leijukuivain 8 lämpötila	4.asema	S4_AIDR8_T			temperature
Leijukuivain 8 kuivatusilman määrä	4.asema	S4_AIDR8_F			air flow
4. Ryhmän paine yläsäädin	4.asema	S4_UPP_PRE		steam press.:	upper cylinder
4. Ryhmän paine alasäädin	4.asema	S4_LOW_PRE			lower cylinder

Asema 1 Kuormitusletkun p. mitt.	S1_TLOAD_PRE	Coating section 1	Tube loading	pressure
Asema 1Kuormitusletkun p. aset.	**S1S_TLD_PRE**			**press. setting**
Asema 2 Kuormitusletkun p. mitt.	S2_TLOAD_PRE	Coating Section 2	Tube loading	pressure
Asema 2 Kuormitusletkun p. aset.	**S2S_TLD_PRE**			**press. setting**
Asema 3 Teräkulma mittaus	S3_TIP_ANG	Coating Section 3	Tip	angle
Asema 3 Teräkuorma mittaus	S3_TIP_LOAD			load
Asema 4 Teräkulma mittaus	S4_TIP_ANG	Coating Section 4	Tip	angle
Asema 4 Teräkuorma mittaus	S4_TIP_LOAD			load
Infra 1 teho asetus	**S1S_IRDR1_P**	**Coating Section 1**	**IR dryer 1**	**power setting**
Infra 2 teho asetus	**S1S_IRDR2_P**		**IR dryer 2**	**power setting**
Infra 3 teho asetus	**S2S_IRDR3_P**	**Coating Section 2**	**IR dryer 3**	**power setting**
Infra 4 teho asetus	**S2S_IRDR4_P**		**IR dryer 4**	**power setting**
Infra 5 teho asetus	**S3S_IRDR5_P**	**Coating Section 3**	**IR dryer 5**	**power setting**
Infra 6 teho asetus	**S3S_IRDR6_P**		**IR dryer 6**	**power setting**
Infra 7 teho asetus	**S4S_IRDR7_P**	**Coating Section 4**	**IR dryer 7**	**power setting**
Infra 8 teho asetus	**S4S_IRDR8_P**		**IR dryer 8**	**power setting**
1.Ryhmä yläsylinterin paine asetus	**S1S_UPP_PRE**	**Coating Section 1**	**upper cylinder**	**press. setting**
1.Ryhmä alasylinterin paine asetus	**S1S_LOW_PRE**		**lower cylinder**	**press. setting**
2.Ryhmä yläsylinterin paine asetus	**S2S_UPP_PRE**	**Coating Section 2**	**upper cylinder**	**press. setting**
2.Ryhmä alasylinterin paine asetus	**S2S_LOW_PRE**		**lower cylinder**	**press. setting**
3.Ryhmä yläsylinterin paine asetus	**S3S_UPP_PRE**	**Coating Section 3**	**upper cylinder**	**press. setting**
3.Ryhmä alasylinterin paine asetus	**S3S_LOW_PRE**		**lower cylinder**	**press. setting**
4.Ryhmä yläsylinterin paine asetus	**S4S_UPP_PRE**	**Coating Section 4**	**upper cylinder**	**press. setting**
4.Ryhmä alasylinterin paine asetus	**S4S_LOW_PRE**		**lower cylinder**	**press. setting**

1.Leijukuivain lämpötila asetus.	**S1S_AIDR1_T**	**Coating Section 1**	**air dryer 1**	**I state setting**
2.Leijukuivain lämpötila asetus.	**S1S_AIDR2_T**		**air dryer 2**	**I state setting**
3.Leijukuivain lämpötila asetus.	**S2S_AIDR3_T**	**Coating Section 2**	**air dryer 3**	**I state setting**
4.Leijukuivain lämpötila asetus.	**S2S_AIDR4_T**		**air dryer 4**	**I state setting**
5.Leijukuivain lämpötila asetus.	**S3S_AIDR5_T**	**Coating Section 3**	**air dryer 5**	**I state setting**
6.Leijukuivain lämpötila asetus.	**S3S_AIDR6_T**		**air dryer 6**	**I state setting**
7.Leijukuivain lämpötila asetus.	**S4S_AIDR7_T**	**Coating Section 4**	**air dryer 7**	**I state setting**
8.Leijukuivain lämpötila asetus.	**S4S_AIDR8_T**		**air dryer 8**	**I state setting**
1.Leijukuivain puhallusnopeus asetus.	**S1S_AIDR1_F**	**Coating Section 1**	**air dryer 1**	**air speed setting**
2.Leijukuivain puhallusnopeus asetus	**S1S_AIDR2_F**		**air dryer 2**	**air speed setting**
3.Leijukuivain puhallusnopeus asetus	**S2S_AIDR3_F**	**Coating Section 2**	**air dryer 3**	**air speed setting**
4.Leijukuivain puhallusnopeus asetus	**S2S_AIDR4_F**		**air dryer 4**	**air speed setting**
5.Leijukuivain puhallusnopeus asetus	**S3S_AIDR5_F**	**Coating Section 3**	**air dryer 5**	**air speed setting**
6.Leijukuivain puhallusnopeus asetus	**S3S_AIDR6_F**		**air dryer 6**	**air speed setting**
7.Leijukuivain puhallusnopeus asetus	**S4S_AIDR7_F**	**Coating Section 4**	**air dryer 7**	**air speed setting**
8.Leijukuivain puhallusnopeus asetus	**S4S_AIDR8_F**		**air dryer 8**	**air speed setting**

A2. Taulu RUNUPS2 – numeerisesti poimittujen ylösajojen ominaisuudet

Sarakkeen nimi	Engl. selitys
RUP_ID	Runup identifier
REF_RUP_ID	The corresponding runup ID in table RUNUPS (if matching)
BREAK_TS	Timestamp of the web breakage
WASH_BEG	The starting point of the wash speed (500 m/min) period
WASH_END	The end point of the (first) wash speed period
RUNUP_BEG	Eventual start of automatic runup
COATING_START_TS	Coater sections are started at 400 m/min
SPEED_UP_TS	The final acceleration to production speed is started
PROD_VEL_TS	Time of reaching the production velocity
RUNUP_END	End of automatic runup - the quality grade of the recipe has been achieved
PROD_VEL	Production velocity attained (m/min)
REEL_ID_NEXT	Reel produced after the breakage
WASTE_APP	Approximate waste (m)

GRADE	Success level of runup (’A’, ’B’, 'C’, ’D’), A is the best
QI	General quality indicator : NULL - not processed 'BAD’ - not qualifying for analysis ’OK’ - good, qualifies
WASH_INTVL	WASH_END - WASH_BEG (s)
IDLE_INTVL	Total interval bewteen IDL_TS and SPEED_UP_TS (s)
TOTAL_INTVL	RUNUP_BEG - BREAK_TS (s)
BAS_W_NEW	Calculated target basis weigth after runup
BAS_W_OLD	Basis weigth before the breakage
DELTA_BAS_W	BAS_W_NEW - BAS_W_OLD