PEHMOMALLEJA IHMISTIETEISIIN

ÄLYKKÄÄT JA OPPIVAT JÄRJESTELMÄT TUTKIJAN TUKENA

VESA A. NISKANEN

HELSINGIN YLIOPISTO
2010

ESIPUHE

Ihmistieteiden, kuten yhteiskunta- ja käyttäytymistieteiden, tutkimuksessa voidaan hieman pelkistäen erottaa kaksi menetelmällistä perinnettä, kvantitatiivinen ja kvalitatiivinen (laadullinen). Edellinen on nojautunut paljon tilastollis-matemaattisiin menetelmiin ja tietokonemallinnuksiin, kun taas jälkimmäisen osalta ei-numeeriset aineistot ja käsin tapahtuva työ ovat tyypillisiä. Näiden kahden lähestymistavan välillä on toisinaan esiintynyt jännitettä, ja usein tutkija on jopa joutunut tekemään joko-tai –valinnan siitä, kumpia menetelmiä työssään käyttäisi.

Tämä kirja pyrkii esittelemään näiden kahden tutkimusperinteen välimailla olevia lähestymistapoja hyödyntämällä oppivien ja älykkäiden järjestelmien menetelmiä, sillä nämä uudet metodit ovat tuottaneet viime vuosikymmeninä käyttökelpoisia tuloksia sekä niin sanotuissa ”insinööritieteissä” että monilla ihmistieteiden aloilla. Kirja on tarkoitettu erityisesti yliopistojen syventäviin ja sitä myöhempiin ihmistieteiden opintoihin sekä näiden tutkimusalojen uusista menetelmistä kiinnostuneille tutkijoille.

Oppivat ja älykkäät järjestelmät, jotka ovat usein analogisia luonnossa esiintyville ilmiöille, tunnetaan erityisesti tietokoneympäristössä myös nimellä *luonnolliset järjestelmät* (natural computing, computational intelligence), ja ne käsittävät muun muassa oppivat järjestelmät (neurocomputing), sumeat järjestelmät (fuzzy systems), evoluutiolaskennan (evolutionary computing), todennäköisyyspäättelyn (probabilistic reasoning), sosiaaliset järjestelmät (social computing) ja biologiset järjestelmät (biological computing). Esimerkkejä näistä järjestelmistä ovat vastaavasti neuroverkot, sumea logiikka, geneettiset algoritmit, Bayes-verkot, parviteoria ja soluautomaatit. Koska alan terminologia (erityisesti suomeksi) ja luokitteluperusteet ovat vielä osittain vakiintumattomia, vaihtoehtoisia termejä ja luokitteluja on kirjallisuudessa tarjolla.

Tarkastelemme tässä kirjassa muutamia keskeisiä tilastollisia mallinnustapoja sekä perinteisellä tavalla että soveltaen edellä mainittuja uusia menetelmiä, erityisesti sumeaa logiikka ja jonkin verran myös neuroverkkoja sekä geneettisiä algoritmeja. Pyrimme sumean logiikan perusajatuksen mukaisesti ihmismäiseen ja kielelliseen mallinnustapaan tietokoneympäristössä, joka toivottavasti on lukijalle ”pehmeämpi” lähestymistapa kuin vastaava perinteinen ”kova” mallinnus. Tällä tavoin pyritään rakentamaan käyttäjän kannalta helpom-

min ymmärrettäviä ja yksinkertaisia malleja, jotka kuitenkin ovat yhtä käyttökelpoisia, ja toisinaan jopa parempiakin, kuin perinteiset vastineensa. Niinpä tämän kirjan on tarkoitus sopia ihmistieteiden menetelmäoppaaksi myös vain vähän tai kohtalaisesti tilastollis-matemaattisista asioista kiinnostuneille lukijoille.

Kvantitatiiviseen tutkimukseen kirja esittelee melko yksinkertaisia ja käyttökelpoisia ei-parametrisia sekä ei-lineaarisia mallinnustapoja. Kvalitatiivisessa tutkimuksessa taas voidaan hyödyntää erityisesti kielellistä mallinnusta yleensä ja kirjan lopussa käsiteltyjä kognitiivisia karttoja. Kirjan tukena voi käyttää vaikkapa kirjaani ”Sumea logiikka - kirkasta älyä ja mallinnusta” (WSOY, 2003), jossa eräitä perusasioita on tuotu esille hieman laajemmin, ja myös filosofisemmin. Tilastotieteen oheislukemistona voin suositella vaikkapa lähteissä mainittuja Jokivuoren ja Hietalan, Metsämuurosen sekä Rannan, Ritan ja Koukin kirjoja, joista olen itse kerännyt ideoita.

Tarkoituksenani on, että tämä kirja omalta osaltaan monipuolistaa ihmistieteiden mallinnustapoja ja suo mahdollisuuden hyödylliseen tietokonemallinnukseen sellaisillekin opiskelijoille ja tutkijoille, joille tällainen tutkimustapa on perinteisten menetelmien avulla tuottanut vaikeuksia.

Esitän kiitokseni Suomen tietokirjailijat ry:lle saamastani taloudellisesta tuesta.

Helsingissä joulukuussa 2009

Vesa A. Niskanen
FT, käyttäytymistieteiden metodiikan dosentti
Helsingin yliopisto, taloustieteen laitos

SISÄLLYS

0. MIKSI PERINTEISET MENETELMÄT EIVÄT AINA RIITÄ?

Matemaattisella mallinnuksella on luonnontieteissä jo satoja vuosia jatkunut perinne, ja se on ollut keskeinen tekijä nykyisen teknologian kehityksessä. Ihmistieteissä, erityisesti yhteiskunta- ja käyttäytymistieteissä, näiden mallien käyttö yleistyi 2. maailmansodan jälkeen, ja tämä tutkimusperinne, eli kvantitatiivinen tutkimus, onkin pääasiassa keskittynyt numeeristen aineistojen tarkasteluun tilastollis-matemaattisin menetelmin.

Kvantitatiiviselle tutkimukselle vaihtoehtoisena, tai jopa kilpailevana, lähestymistapana on metodiselta kannalta pidetty kvalitatiivista eli laadullista tutkimusta, jossa tavallisesti käytetään ei-numeerisia aineistoja ja ei-laskennallisia menetelmiä. Kvalitatiivinen tutkimus oli ihmistieteissä tyypillistä ennen 2. maailmansotaa, ja se on uudestaan voimistunut viime vuosikymmeninä kun innostus kvantitatiiviseen tutkimukseen on vähentynyt.

Luonnontieteissä, varsinkin niin sanotuissa ”insinööritieteissä”, on kuitenkin jo havaittu, että perinteinen tilastollis-matemaattinen mallinnus ei ole aina tuottanut tarpeeksi käyttökelpoisia tuloksia. Tämä on ensinnäkin tietyissä tilanteissa johtunut matematiikan aseman ylikorostuksesta, jolloin reaalimaailman on oletettu olevan luonteeltaan matemaattinen tai jopa matemaattisesti täysin kuvailtavissa. Tähän on voinut myös liittyä sellainen lähestymistapa, että systeemejä on kehitetty liian teorialähtöisesti ilman riittäviä kytkentöjä havaintoihin tai reaalimaailman todellisiin olosuhteisiin. On nimittäin käytännön esimerkkejä matemaattisista malleista, jotka eivät toimi reaalimaailmassa, ja toisaalta reaalimaailmassa toimivia asioita, jotka eivät matemaattisten sekä muiden mallien mukaan ole toimintakelpoisia.

Toiseksi, käytetyt mallit ovat muun muassa laskennallisista syistä olleet toisinaan liian pelkistettyjä tai yksinkertaistettuja. Tyypillisesti on pyritty käyttämään lineaarisia malleja, vaikka reaalimaailman ilmiöt ovat usein ei-lineaarisia.

Kolmanneksi, tilastolliset mallit ovat perustuneet usein muuttujia koskeviin jakaumaoletuksiin, kuten normaalijakaumaan, ja jos näitä ehtoja ei ole voitu täyttää, ei välttämättä ole ollut vaihtoehtoisia menetelmiä tarjolla.

Kun sitten tietokoneita voitiin hyödyntää mallinnuksessa, erityisesti 1970-luvulta lähtien, perinteisen matemaattis-tilastollisen mallinnuksen rinnalle syntyi uusia menetelmätieteitä, jotka tunnetaan mm. nimellä oppivat ja älykkäät

järjestelmät (adaptive and intelligent systems, natural computing, computational intelligence, soft computing). Tietokoneiden suuren laskentatehon ansiosta voitiin nyt paremmin tarkastella reaalimaailman ilmiöitä, ja nämä uudet tutkimusalat käyttivät usein myös esimerkkinä sellaisia luonnonilmiöitä kuten ihmisen aivotoiminta ja päättely, evoluution periaatteet, solujen toiminta tai eliöyhteisöjen ja parvien käyttäytyminen. Myös tekoälytutkimus (artificial intelligence, AI) keskittyi näiden ilmiöiden, varsinkin inhimillisen päättelyn, tietokonemallinnukseen.

Oppivat ja älykkäät järjestelmät käsittävät muun muassa oppivat järjestelmät (neurocomputing), sumeat järjestelmät (fuzzy systems), evoluutiolaskennan (evolutionary computing), todennäköisyyspäättelyn (probabilistic reasoning), sosiaaliset järjestelmät (social computing) ja biologiset järjestelmät (biological computing). Esimerkkejä näistä järjestelmistä ovat vastaavasti neuroverkot (neural networks), sumea logiikka (fuzzy logic), geneettiset algoritmit (genetic algorithms), Bayes-verkot (Bayesian networks), parviteoria (swarm theory) ja soluautomaatit (cellular automata). Koska alan terminologia (erityisesti suomeksi) ja luokitteluperusteet ovat vielä osittain vakiintumattomia, vaihtoehtoisia termejä ja luokitteluja on kirjallisuudessa tarjolla.

Nykyään oppivia ja älykkäitä järjestelmiä käytetään jo laajasti insinööritieteissä, ja niiden perusteita opetetaan tekniikan oppilaitoksissa. Tyypillisiä sovelluskohteita ovat suurteollisuuden säätöjärjestelmät (control), kodinkoneet, kamerat, autot, päätöksentekojärjestelmät (decision making), robotiikka (robotics), Internet ja hahmontunnistus (pattern recognition).

Ihmistieteissä näiden järjestelmien sovellukset ovat olleet melko vähäisiä lääketieteen ja taloustieteiden sovelluksia lukuun ottamatta, ja sama koskee myös näiden järjestelmien opetusta. Näissä tieteissä kohdataan kuitenkin edellä mainittuja ongelmia, jos sovelletaan pelkästään perinteistä tilastollis-matemaattista mallinnusta. Niinpä oppivien ja älykkäiden järjestelmien käyttöä näissäkin tieteissä pitäisi edistää, ja tämä kirja pyrkii osaltaan tätä tehtävää täyttämään.

Keskitymme sumeiden systeemien sovelluksiin, koska ne ovat ehkä käyttäjäystävällisin tapa lähestyä näitä uusimuotoisia järjestelmiä. Näiden systeemien lisäksi hyödynnämme mallien rakentamisessa myös neuroverkkoja ja geneettisiä algoritmeja. Tietokonemallinnuksissa käytämme tilastotieteen sovelluksissa SPSS-ohjelmistoa ja oppivien ja älykkäiden järjestelmien osalta Mat-

lab'in perusohjelmistoa sekä sumean logiikan, neuroverkkojen ja geneettisten algoritmien työkaluja (toolbox). Nämä molemmat ohjelmistot ovat laajasti käytössä ja helppokäyttöisiä. Useinhan lopulliset mallitkin voivat myös olla perinteisten ja uusimuotoisten mallien yhdistelmiä, joten nämä kaksi lähestymistapaa mallinnukseen eivät missään nimessä ole toisensa poissulkevia, vaan toisiaan täydentäviä.

Kun lukijat sitten kokeilevat kirjan mallinnustapoja omiin tutkimusasetelmiinsa, he voivat tieteellisellä kriittisyydellä arvioida näiden uusien mallien käyttökelpoisuutta. Tässä kirjassa esitettävät ajatukset edellyttävät uutta ajattelutapaa, jossa ongelmia lähestytään käytännön ja ihmiselle ominaisen kielellisen päättelyn näkökulmasta. Usein nämä uusimuotoiset mallinnukset tuottavat hyviä tuloksia, ja toivottavasti tämä toteutuu lukijoidenkin tutkimustyössä.

1. RYHMITTELYANALYYSISTÄ EI-OHJATTUUN OPPIMISEEN JA TIEDON LOUHINTAAN

Aloitamme tarkastelumme ryhmittelyanalyysistä (cluster analysis), koska sillä on myöhemmin tärkeä asema mallinnuksessamme. Jos haluamme esittää tutkimusaineistomme tiivistetyssä muodossa tai etsiä siitä mahdollisia havaintoarvojen kasaumia eli rypäitä (cluster), voimme sopivilla menetelmillä kuvailla aineistomme jopa muutamalla lukuarvolla.

Jos vaikkapa tarkastelemme henkilöiden pituuksia, keskiluvut esittävät tiivistetyssä muodossa tämän aineiston keskimääräistä suuruutta. Jos kuitenkin pituudet ovat kasaantuneet useammaksi eri ryhmäksi, yksi keskiluku ei riitä, vaan meidän pitää laskea keskiluvut jokaisesta ryhmästä erikseen. Tilanne muuttuu vielä hankalammaksi, jos tarkastelemme useampaa muuttujaa yhtä aikaa.

Ryhmittelyanalyysin avulla etsitään aineiston havaintoarvojen tihentymiä tai kasaumia. Nämä kasaumat eli ryhmät käsittävät keskenään samantyyppisiä havaintoja, ja ne voidaankin sitten tarvittaessa nimetä sen mukaisesti, millaisia tämän ryhmän oliot ovat. Ryhmittelyanalyysiä voidaan käyttää itsenäisesti tai apuvälineenä tiedon louhinnassa (data mining) ja hahmontunnistuksessa (pattern recognition). Edellisessä tapauksessa pyritään löytämään suurista aineistoista tyypillisiä piirteitä tai toivotut ominaisuudet täyttävät havainnot, kun taas jälkimmäinen menetelmä pyrkii luokittelemaan tai tunnistamaan havaintoja annettujen piirteiden perusteella. Ryhmittelyanalyysin perusteella voidaankin hahmontunnistusta varten tehdä esimerkiksi erotteluanalyysejä (discriminant analysis), ja näitä asioita tarkastellaan 6.

1.1. Perinteinen ryhmittelyanalyysi

Kuvassa 1.1 on sirontakuvio kahden kuvitteellisen muuttujan, X ja Y, havaintoarvoista. Havaintopisteet näyttävät muodostavan kolme erillistä ryhmää, joista keskimmäisessä ryhmässä on vähiten ja oikeanpuoleisessa eniten havaintoja. Perinteisessä ryhmittelyanalyysissä siis pyritään löytämään sellaisia havaintojen ryhmiä, joissa ryhmissä olevat tilastoyksiköt ovat keskenään mahdollisimman samanlaisia ja toisaalta ryhmien tulisi olla keskenään mahdollisimman erillisiä. Jos käytämme varianssianalyysin kieltä, ryhmien sisäisten varianssien (hajonto-

jen) pitäisi olla mahdollisimman pieniä ja ryhmien välisten varianssien taas mahdollisimman suuria. Myös muuttujia on mahdollista ryhmitellä, mutta tällöin sovelletaan tavallisesti pääkomponentti- tai faktorianalyysiä.

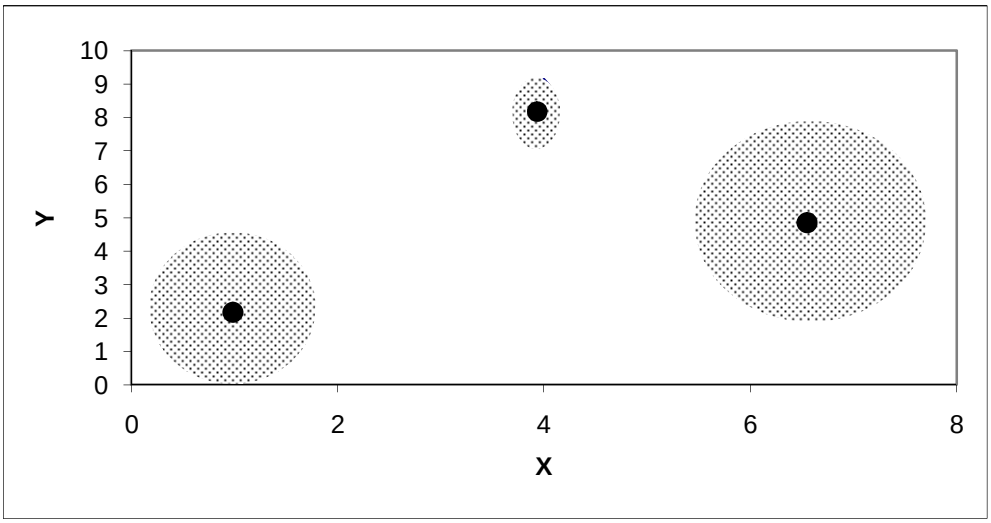
Käytännössä ryhmittelyanalyysissä lasketaan yleensä tilastoyksikköjen väliset etäisyydet toisistaan valittujen muuttujien muodostamassa avaruudessa ja näin saadun etäisyysmatriisin perusteella samaan ryhmään sijoitetaan aina ne yksiköt tai oliot, joiden keskinäiset etäisyydet ovat pieniä (eli jotka ovat lähellä toisiaan).

Ryhmien keskukset määritetään sitten vaikkapa laskemalla ryhmän havaintojen keskiarvot jokaisen muuttujan osalta (painopiste). Ryhmittelymenetelmiä ja hyvän ryhmittelyn (mm. sopivan ryhmien lukumäärän määrittäminen) kriteereitä on tarjolla useita alan kirjallisuudessa ja tilastollisissa ohjelmistoissa, mutta valitettavasti kaikissa niissä on omat puutteensa.

Kuvan 1.1 tapauksessa näyttää siltä, että parhaimman ratkaisun saa aikaan sellainen ryhmittelyanalyysi, joka tuottaa kolme ryhmää, ja näiden keskukset ovat

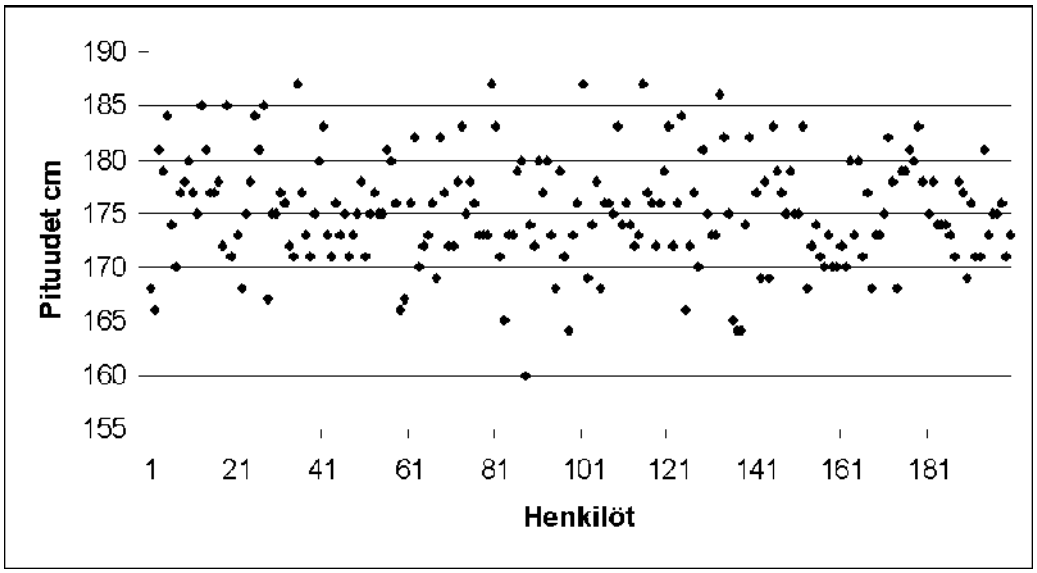
1. X=1 ja Y=2
2. X=4 ja Y=8
3. X=7 ja Y=5.

Kyseisten keskusten ajatellaan nyt olevan ryhmiensä tyypillisiä edustajia. Niinpä voimme pelkistetysti esittää tämän aineiston näiden kolmen keskuksen avulla.



Kuva 1.1. Kolme ryhmää kaksiulotteisessa avaruudessa ja niiden keskukset.

Tarkastellaan kuvitteellista, kuvassa 1.2 esitettyä aineistoa 200 miehen pituuksista kun nämä pituudet on esitetty kokonaisina senttimetreinä:



Kuva 1.2. Aineisto miesten pituuksista (200 havaintoa).

Taulussa 1.1 on esitetty eräitä keskeisiä aineistomme tunnuslukuja (SPSS-ohjelman tulosteita), jolloin esimerkiksi toteamme, että aritmeettinen keskiarvo on 175,21 cm ja keskihajonta 5,046 cm.

Taulu 1.1. Pituus-aineiston tunnuslukuja

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
Pituus_cm	200	27	160	187	175.21	5.046	25.463
Valid N (listwise)	200						

Jos haluamme nyt löytää tyypillisiä aineistomme ryhmiä ja määritämme ryhmien lukumääräksi viisi, SPSS:n k-means –ryhmittelymenetelmän mukaan (Analyze – Classify – K-means) näiden ryhmien keskukset ovat 165 cm, 170 cm, 175 cm, 181 cm ja 186 cm. Silloin ensimmäisessä ryhmässä on eniten miehiä (96), ja vähiten on toisessa ja viidennessä ryhmässä (11). Niinpä voimme ajatella, että nämä ovat viisi tyypillistä pituutta tässä aineistossa ja suurin ryhmä ovat noin 165-senttiset kun taas pienimpiä ryhmiä ovat noin 170- ja 186-senttiset.

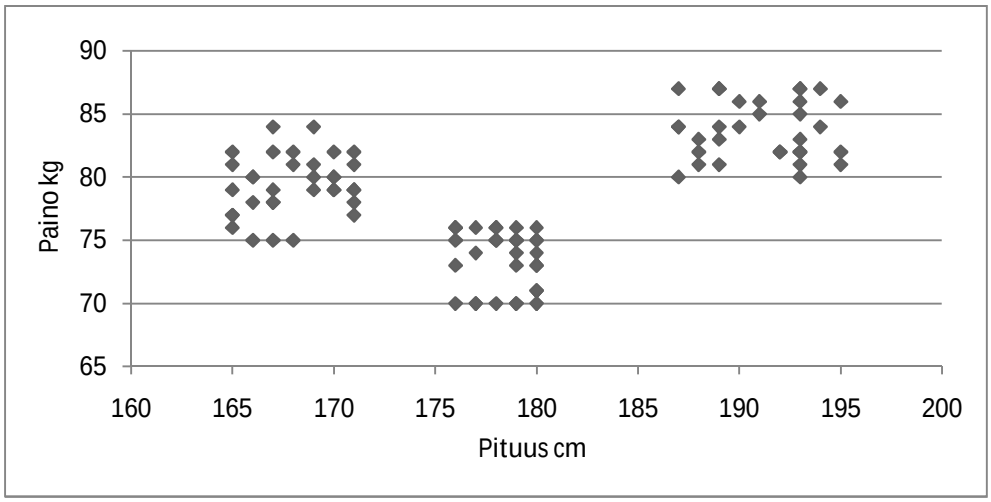
Ryhmittelyanalyysissä ryhmien etsiminen tapahtuu aineiston ja tiettyjen laskentaehtojen perusteella melko itsenäisesti ja vapaasti, joten se on siinä mielessä luonteeltaan exploratiivinen menetelmä. Tutkija siis pyrkii löytämään tyypillisiä ryhmiä vaikkapa omaa teorianmuodostustaan varten, ja exploratiivinen tutkimus onkin tavallista varsinkin uusien ilmiöiden tarkastelussa.

Ryhmittelyanalyysi käsittää seuraavat vaiheet:

1. Aineiston mahdollisen esikäsittelyn (feature analysis), jolloin havaintoarvoille voidaan tehdä matemaattisia muunnoksia tai tiettyjä arvoja voidaan poistaa. Esimerkiksi muuttujien arvot voidaan muuntaa vastaaviksi standardipisteiksi (z-pisteiksi) vähentämällä niistä aritmeettinen keskiarvo ja jakamalla erotus sitten keskihajonnalla.
2. Tarvittaessa poistetaan analyysin kannalta epäoleelliset muuttujat (feature extraction). Tämä voidaan tehdä muun muassa korrelaatiokertoimien tarkastelun perusteella. Muuttujia voidaan myös yhdistellä keskenään summamuuttujiksi esimerkiksi pääkomponentti-, faktori- ja osioanalyysin avulla.

3. Aineiston graafinen tarkastelu auttaa ryhmien rakenteen tarkastelussa ja ryhmien sopivan lukumäärän valinnassa. Jos muuttujia on korkeintaan kolme, näemme näitä asioita suoraan sirontakuvioista, muulloin tarkastelemme erikseen aina korkeintaan kolmea muuttujaa kerrallaan tai tutkimme aineistosta tuotettua etäisyysmatriisia. Jälkimmäisen avulla voimme tarkastella yksiköiden välisiä etäisyyksiä pareittain, ja tämä epäsuora tapa on useimmiten perustana ryhmittelyanalyysin laskutoimituksille.
4. Mielekkäiden (”luonnollisten”) ryhmien etsiminen aineistosta. Ryhmiä edustavat sitten monissa tutkimustilanteissa näiden ryhmien keskukset.
5. Ryhmittelyn hyvyyden arviointi. Perussääntö on, että ihannetapauksessa ryhmien sisällä tilastoyksiköt ovat mahdollisimman lähellä toisiaan, kun taas ryhmien väliset erot ovat mahdollisimman selvät. Ryhmien ”oikea” lukumäärä perustuu yleensä tähän ajatukseen. Lisäksi ryhmien ulkopuolelle ei saisi jäädä yksittäisiä tilastoyksiköitä. Tällainen ihannetilanne on esimerkiksi kuvan 1.3 tapauksessa, mutta todellisuus ei ole yleensä näin ruusuinen.

Tarkastellaan ryhmittelyanalyysiä hieman syvällisemmin kuvan 1.3 aineiston avulla, jossa on sadalta mieheltä mitattu pituus ja paino. Koska kyse on kahden muuttujan aineistosta, näemme sirontakuviosta suoraan, onko aineistosta löydettävissä ryhmiä, ja tässä tapauksessa ainakin kolme ryhmää näyttäisi olevan. Yleensä ryhmittelyanalyysit perustuvat kuitenkin ”sokkona” suoritettuihin laskutoimituksiin, koska monen muuttujan tapauksessa vastaavia, kokonaiskuvan antamia sirontakuvioita ei voida piirtää, joten kuvat ovat vain alustavia apuvälineitämme.



Kuva 1.3. Sadan miehen pituudet ja painot.

Tavallisesti lasketaan aluksi tilastoyksiköiden väliset etäisyydet avaruudessa, jonka ulotteisuus määräytyy muuttujien lukumäärän mukaan. Meidän tapauksessamme siis on kyse kaksiulotteisesta avaruudesta, joten jokaisen miehen sijainti tässä avaruudessa voidaan esittää pisteenä tai vektorina

(pituus,paino)

Jos siis tietyn miehen pituus on 180 cm ja paino 70 kg, hänen koordinaattinsa tässä avaruudessa ovat (180,70) (itse asiassa nämä vektorit tuottavat kuvan 1.3 mukaisia pisteitä).

Etäisyydet esitetään etäisyysmatriisina, jonka rivien ja sarakkeiden määrä on sama kuin yksiköiden määrä (meillä siis 100 miestä eli 100×100 matriisi). Kahden yksikön välinen etäisyys, vaikkapa henkilöt nro 23 ja 57, on silloin solussa, joka on 23:lla rivillä 57:ssä sarakkeessa. Koska yksiköiden 23 ja 57 välinen etäisyys on sama kuin yksiköiden 57 ja 23, on sama etäisyys myös 57:llä rivillä 23:ssa sarakkeessa. Matriisin lävistäjällä (so. solut, joissa rivi- ja sarakenumero ovat samat) on aina yksikön etäisyys itsensä kanssa eli niiden arvo = 0. Niinpä kahden yksikön välinen etäisyys on itse asiassa esitetty matriisin lävistäjän molemmilla puolilla. Taulussa 1.2 on esitetty etäisyysmatriisi aineistostamme kymmenen miehen osalta. Niinpä esimerkiksi miesten nro 3 ja 8 välinen etäisyys on noin 3,16.

Taulu 1.2. Osa 100 miehen aineiston etäisyysmatriisista										
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	0	2.2361	4	6.4031	5.099	1.4142	1	5.831	1.4142	3.1623
2	2.2361	0	5.3852	8.6023	6.7082	3.6056	3.1623	7.8102	3	5
3	4	5.3852	0	5	1.4142	3.1623	4.1231	3.1623	5.099	5.831
4	6.4031	8.6023	5	0	4.1231	5	5.6569	2.2361	6.4031	5.3852
5	5.099	6.7082	1.4142	4.1231	0	4	5	2	6	6.3246
6	1.4142	3.6056	3.1623	5	4	0	1	4.4721	2	2.8284
7	1	3.1623	4.1231	5.6569	5	1	0	5.3852	1	2.2361
8	5.831	7.8102	3.1623	2.2361	2	4.4721	5.3852	0	6.3246	6
9	1.4142	3	5.099	6.4031	6	2	1	6.3246	0	2
10	3.1623	5	5.831	5.3852	6.3246	2.8284	2.2361	6	2	0

Yksiköiden välinen etäisyys voidaan laskea monella tavalla, ja käytetty laskutapa eli metriikka voi vaikuttaa paljonkin ryhmittelyn lopputulokseen. Edellä on käytetty suosittua euklidista metriikkaa (joka perustuu Pythagoraan lauseeseen):

$$\text{etäisyys}(x,y) = +\sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \ \ldots \ + (x_m - y_m)^2}$$

kun *m* on muuttujien lukumäärä ja vektorit $x = (x_1, x_2, \ldots, x_m)$, $y = (y_1, y_2, \ldots, y_m)$ esittävät kahden tilastoyksikön osalta muuttujien arvot (usein käytännössä x ja y ovat havaintomatriisin rivejä, ainakin SAS-ohjelma käyttää tällaisesta kokonaisesta rivistä nimeä havainto (observation)).

Esimerkiksi edellä mainittujen miesten nro 3 ja 8 tapauksessa heidän pituutensa ja painonsa ovat havaintomatriisin perusteella vastaavasti vektoreita

$x = (166,80)$ ja $y = (165,77)$.

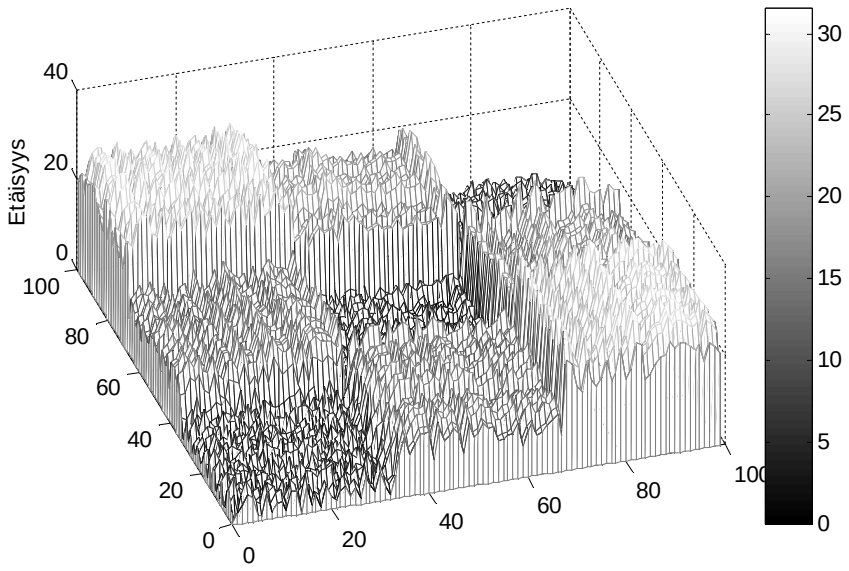
Niinpä euklidinen etäisyys on (vrt. taulu 1.2)

$$\text{etäisyys}(x,y) = \sqrt{(166-165)^2 + (80-77)^2} \ = \sqrt{10} \ = 3{,}16$$

eli lasketaan yhteen pituuksien ja painojen erotuksien neliöt, ja näiden summasta sitten lasketaan (positiivinen) neliöjuuri.

Koska etäisyydet lasketaan tavallisesti moniulotteisessa avaruudessa (eli käytämme useita muuttujia), tilastoyksiköitä on oltava niin paljon enemmän kuin muuttujia, että jokaiseen ryhmään saadaan riittävästi yksiköitä. Toisaalta on otettava huomioon, että suuret aineistot tai muuttujien lukumäärät vaativat paljon laskentatehoa.

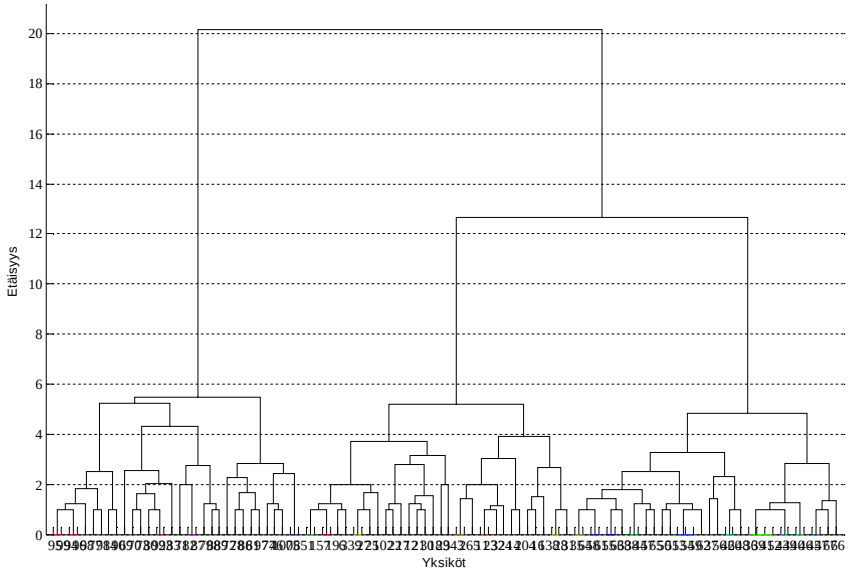
Kuvassa 1.4 ovat kuvan 1.3 aineiston euklidiset etäisyydet. Siinä ovat sattumalta toisiaan lähellä olevat yksiköt valmiiksi vierekkäin. Yleensä näin ei ole, mutta tarkastelun helpottamiseksi voidaan joskus tehdä tällainen yksiköiden uudelleensijoittelu. Toteamme kuvan perusteella (koska yksiköt siis sijoitettu edellä kuvatulla tavalla sopivasti), että aineistossa näyttää olevan pituuden ja painon perusteella kolme ryhmää (pienten etäisyyksien aluetta eli kolme tummaa aluetta kun lävistäjää ei oteta huomioon), ja että jokaisessa ryhmässä on noin kolmekymmentä yksikköä. Näin voimme siis ”graafisen etäisyysmatriisin” avulla tarkastella epäsuorasti ja suuntaa-antavasti ryhmittelyä hyvin monenkin muuttujan tapauksessa.



Kuva 1.4. Graafinen 100×100 etäisyysmatriisi miesten pituuksien ja painojen perusteella.

Dendrogrammit ovat yksi nopea tapa ryhmien muodostumisen seuraamiseksi. Nehän esittävät, kuinka etäisyyden kasvaessa aina lähekkäin olevista yk-

siköistä tai aikaisemmin muodostetuista ryhmistä muodostetaan yhä suurempia ryhmiä (eli tietyllä ”säteellä” toisistaan olevat ryhmät tai yksiköt yhdistetään). Esimerkiksi kuvan 1.4 perusteella ryhmien välisten etäisyyksien ollessa 12 (tässä tapauksessa on laskettu niiden keskusten väliset etäisyydet), voimme muodostaa kolme ryhmää, kun taas etäisyyden kasvaessa 14:ksi saamme kahden ryhmän ratkaisun. Tilastollisten ohjelmistojen avulla voidaan myös selvittää, mitä yksiköitä ryhmät sisältävät.



Kuva 1.4. Dendrogrammi kuvan 1.2 aineistosta. Pystyviivat ilmaisevat ryhmien määrän kullakin etäisyydellä.

Jos olemme tehneet päätöksen ryhmien lukumäärästä tai haluamme tarkastella erilaisilla ryhmien lukumäärillä tuotettuja malleja, voimme tilastollisten ohjelmistojen avulla saada selville muun muassa ryhmien keskukset ja niiden sisältävien yksiköiden lukumäärän. Jos vaikkapa päätämme etsiä edellä kuvatusta aineistosta kolme ryhmää ja haluamme myös tietää näiden ryhmien keskukset, SPSS:n k-means –menetelmä tuottaa seuraavat tulokset:

Final Cluster Centers

	Cluster		
	1	2	3
Pituus_cm	191	179	168
Paino_kg	84	73	79

Nämä ovat siis tässä tapauksessa kolme tyypillistä miestä aineistossamme. Ryhmien koot ovat vastaavasti 32, 32 ja 34 miestä.

K-means –algoritmi aloittaa ryhmittelyn määrittämällä ensin vaikkapa satunnaisesti ryhmien keskukset (k kpl) ja laskemalla sitten ryhmien väliset ja sisäiset varianssit. Sen jälkeen ryhmien keskuksien paikkoja muutetaan pienin askelin (eli iteroimalla), kunnes ryhmien väliset varianssit ovat maksimaaliset ja ryhmien sisäiset minimaaliset. SPSS tuottaa näiden varianssien F-suhteet jokaisen muuttujan osalta, ja tässä mielessä parhaalla ryhmittelyllä on korkeimmat F-suhteet. Tämän algoritmi, kuten moni muukin ryhmittelymenetelmä, toimii parhaiten kun ryhmät ovat ympyrän tai pallon muotoisia.

Samalla tavoin, jos haluamme vaikkapa löytää tyypillisiä lukion toisen luokan oppilaita aineistosta, jonka muuttujina ovat lukuaineet sekä eräät taustamuuttujat kuten sukupuoli ja kotipaikka, seuraavat ryhmät voisivat olla mahdollisia:

- Hyvin kieliä osaavat tytöt.
- Melko hyvin matematiikkaa hallitsevat pojat.
- Huonosti ruotsia osaavat itäsuomalaiset pojat.
- Hyvin uskontoa, biologiaa ja historiaa osaavat länsisuomalaiset tytöt.

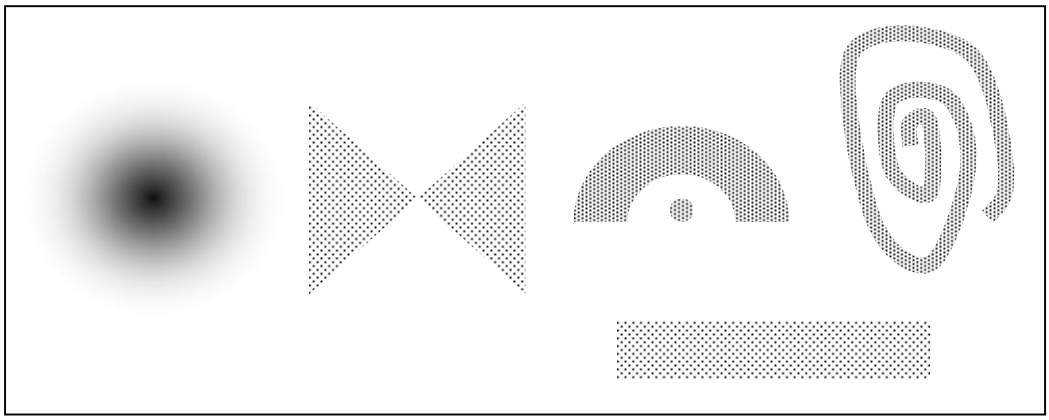
Näin siis perinteinen ryhmittelyanalyysi pääpiirteissään tapahtuu. Se on yleensä käyttökelpoinen, jos ryhmät selvästi erottuvat toisistaan, mutta jos aineistossa on ryhmien välillä rajatapauksia, se voi tehdä niiden osalta turhan yksioikoisia tai keinotekoisia sijoitteluja. Tätä ongelmaa varten, rajatapausten ja epätäsmällisten ryhmien tarkasteluun, on kuitenkin kehitetty uusia menetelmiä, sumea joukko-oppi ja päättely, ja näihin tutustumme seuraavaksi.

1.2. Sumeat joukot ja kielelliset muuttujat – lämmittelyä sumeaan ryhmittelyanalyysiin

Perinteiset ryhmittelymenetelmät ovat ongelmallisia tietyissä tilanteissa. Jos vaikkapa ryhmittelemme hedelmiä tiettyjen piirteiden perusteella omenoiksi, banaaneiksi, appelsiineiksi, persikoiksi ja luumuiksi, niin mihin ryhmään kuuluu nektariini, koska se on osittain persikka ja osittain luumu? Tämän vuoksi on kehitetty uusimuotoisia menetelmiä oppivien ja älykkäiden järjestelmien avulla.

Oppivien eli adaptiivisten järjestelmien (adaptive systems, esim. ”neuroverkot”) yhteydessä tällainen ryhmien etsintä liittyy tietokoneen avulla tapahtuvaan aineiston oppimiseen, ja koska järjestelmä suorittaa etsintää kohtalaisen itsenäisesti ja jopa ”älykkäästi”, tässä yhteydessä käytetään termiä ei-ohjattu oppiminen (unsupervised learning). Muita uusia menetelmiä ovat sumea ryhmittelyanalyysi sekä evoluutiolaskentaan perustuvat tekniikat.

Kuvassa 1.5 on esitetty viittä tyyppiä olevia ongelmallisia pistejoukkoja kaksiulotteisessa avaruudessa. Ensinnäkin niiden osalta on pohdittava, mitkä pisteet kuuluvat ryhmään ja mitkä eivät (varsinkin kaksi ensimmäistä tapausta vasemmalta). Toiseksi on arvioitava, monestako ryhmästä on itse asiassa kysymys (kolme oikeanpuoleisinta tapausta ylhäällä).



Kuva 1.5. Ongelmallisia pistejoukkoja ryhmittelyanalyysissä.

Kuvan 1.5 ensimmäinen ja toinen ryhmä vasemmalla ovat meillä jatkossa erityisen kiinnostuksen kohteena, koska tämäntyyppiset ongelmatilanteet johti-

vat uudenlaisen joukko-opin kehittämiseen. Kyse on niissä nimittäin perimmältään siitä, kuinka käsittelemme ryhmittelyssä ja luokittelussa rajatapauksia. Esimerkiksi vasemmalla olevan ryhmän tapauksessa meidän on päätettävä, kuinka etäällä keskuksesta pisteet saavat olla, jotta ne vielä kuuluvat ryhmään, sillä pistejoukko harvenee keskuksesta poispäin. Toinen ryhmä vasemmalla taas, eli niin sanottu ”perhonen”, voikin itse asiassa koostua kahdesta kolmion muotoisesta ryhmästä, ja meidän on silloin päätettävä, kumpaan ryhmään vaakatason keskikohdassa olevat pisteet kuuluvat.

Oikeassa alanurkassa oleva suorakulmio on taas useimmille ryhmittelymenetelmille sen vuoksi vaikea, että siihen eivät kunnolla sovellu pisteiden etäisyyksiä ryhmän keskuksesta laskevat menetelmät (jotka olettavat ryhmien olevan ympyröitä tai palloja), joten ryhmän tiheyden laskeminen on sen osalta parempi tapa, mutta tällaisia menetelmiä ei ole vielä paljon tarjolla.

Kielelliseltä kannalta ilmiöiden rajatapauksia voidaan lähestyä epätäsmällisten termien näkökulmasta. Jos me vaikkapa pohdimme, mitä tarkoittaa *nuori ihminen*, voimme epäröimättä todeta, että 15-vuotias on nuori kun taas 70-vuotias ei ole. Joukko-opin kannalta ajateltuna voimme tällöin sanoa, että edellinen henkilö kuuluu nuorten ihmisten joukkoon kun taas jälkimmäinen ei kuulu. Mutta entäpä sitten vaikkapa 25-vuotiaat, 30-vuotiaat tai 35-vuotiaat? He eivät ilmeisesti ole selvästi nuoria, mutta toisaalta heitä voidaan pitää ainakin jonkin verran nuorina. Niinpä voimme ajatella, että on olemassa erilaisia nuoruuden asteita, jolloin vaikkapa 15-vuotias on selvästi nuori, 25-vuotias melko selvästi nuori, 35-vuotias jonkin verran nuori ja 70-vuotias ei ole ollenkaan nuori. Näinhän ihmisen intuitio yleensä käsittelee tätä asiaa.

Nämä ongelmat alkavat siitä, että perinteinen länsimainen valtavirtamatematiikka ja –logiikka perustuvat intuitiomme vastaisesti kaksiarvoisuuteen, eli asiat käsitellään joko-tai –periaatteella. Väitteet ovat joko tosia tai epätosia, tai ihmiset ovat joko nuoria tai ei-nuoria. Tämä niin sanottu kolmannen poissuljetun laki (the law of the excluded middle) on hallinnut kaikkea teorianmuodostusta ja mallinnusta varsinkin kvantitatiivisessa tutkimuksessa. Niinpä ihmisen intuitio, joka hyväksyy asteittaiset muutokset, ja vallitseva kaksiarvoinen logiikka ovat olleet keskenään ristiriitaisia, ja tämä asia tulee selvästi esiin tunnetussa sorites- eli falakros-paradoksissa. Sovelletaan kaksiarvoista logiikkaa seuraavaan esimerkkiin:

1. Tänään 15-vuotta täyttänyt henkilö on nuori (totta).
2. Entä jos hän on päivän vanhempi, onko hän vielä nuori? Kyllä vai ei? Ilmeisesti vastaamme *kyllä* eli tosi väite.
3. Entä jos hän on kaksi päivää vanhempi? Ilmeisesti hän edelleen on nuori.
4. Entä jos kolme päivää vanhempi? Ilmeisesti edelleen nuori.
5. Jos jatkamme näin, niin jossakin vaiheessa, vaikkapa 30 ikävuoden kohdalla, meidän on todettava, että kyseinen henkilö ei ole enää nuori, koska ilman tällaista rajanvetoa 70-vuotiaskin olisi sitten nuori. Niinpä kaksiarvoinen ajattelu johtaa asteittaisten muutosten tapauksessa keinotekoiseen ja jyrkkään rajanvetoon, jolloin tietyn ikäinen henkilö on nuori, mutta jo päivän vanhempana hän ei enää ole nuori.
6. Siis jyrkkä rajanveto ei ole aina järkevää, mutta toisaalta ilman rajanvetoa kaikki ovat nuoria. Niinpä olemme sorites-paradoksiin johtaneessa tilanteessa.

Tällainen tiukka rajanveto, jonka seurauksena hyppäämme ”töksähtäen” joukosta toiseen, tuottaa monia ongelmia teorianmuodostuksessa ja mallien rakentamisessa.

Moniarvologiikassa (many-valued tai multi-valued logic) tämä paradoksi voidaan välttää, koska silloin ilmiöiden asteittaiset muutokset voidaan ottaa paremmin huomioon. Sumeassa joukko-opissa ja logiikassa (fuzzy set theory, fuzzy logic) on sovellettu moniarvologiikkaa erityisesti tietokonemallinnuksiin, ja tätä lähestymistapaa sovellamme paljon jatkossa.

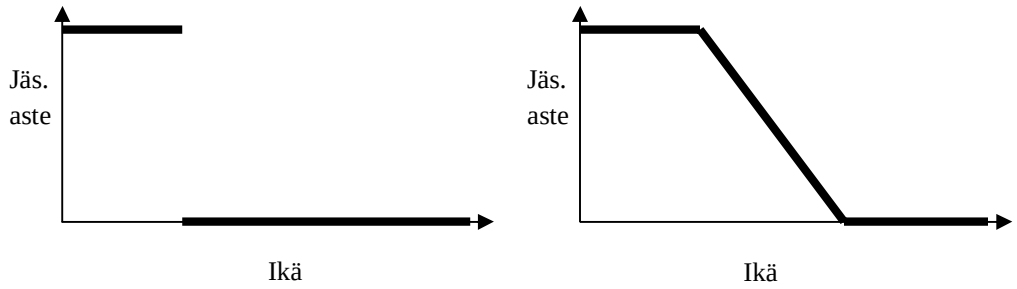
Berkeleyn yliopiston tietojenkäsittelytieteen professori Lotfi Zadeh esitti 1960-luvulla sumeiden joukkojen perusajatuksen, jonka mukaan oliot voivat myös osittain kuulua tiettyihin joukkoihin. Kun perinteinen kaksiarvoinen joukko-oppi edellyttää, että olio joko kuuluu tai ei kuulu tiettyyn joukkoon, sumeiden joukkojen tapauksessa oliot siis voivat kuulua joukkoihin myös vain osittain. Niinpä esim. 15-vuotias voi kuulua täysin ja 25- sekä 30-vuotiaat vain osittain nuorten ihmisten sumeaan joukkoon. 70-vuotias ei taas kuuluisi ollenkaan tähän joukkoon.

Hieman matemaattisemmin ilmaistuna, olion jäsenyysaste (degree of membership) tiettyyn sumeaan joukkoon voi vaihdella täydestä jäsenyysastees-

ta (full membership) ei-kuulumiseen (non-membership). Yleensä käytämme jäsenyysasteille numeroita välillä nollasta (ei-kuuluminen) yhteen (täysi jäsenyys). Perinteisten joukkojen tapauksessa taas jäsenyysaste on aina joko 0 tai 1, joten tässä mielessä perinteiset joukot ovat sumeiden joukkojen erikoistapauksia.

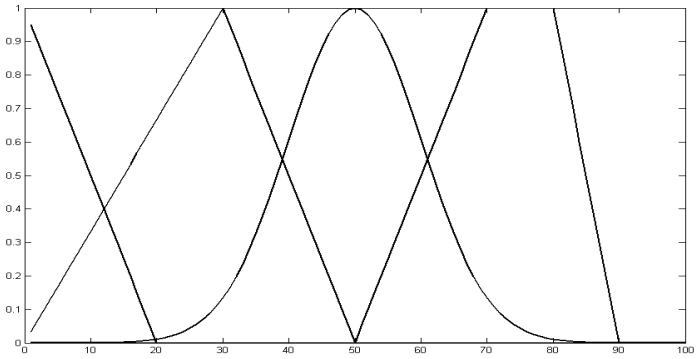
Nyt voidaankin edellä esitetty sorites-paradoksi ratkaista helposti: mitä vanhempi henkilö on, sitä pienemmällä jäsenyysasteella hän kuluu nuorten ihmisten joukkoon (ja yhtäaikaisesti, sitä suuremmalla jäsenyysasteella hän kuuluu vanhojen ihmisten joukkoon). Näin ollen sumeat joukot ottavat huomioon asioiden epätäsmällisyyden ja asteittaisen muutoksen.

Sumeat joukot esitetään tavallisesti niitä kuvaavien jäsenyysfunktioiden (membership function) avulla. Kuvassa 1.6 on esitetty nuorten ihmisten joukko perinteisen ja sumean joukon avulla. Perinteisessä joukossa kuulutaan nuorten joukkoon tiettyyn ikään asti täydellä jäsenyysasteella, ja sen jälkeen ei lainkaan kuuluta tähän joukkoon. Sumeassa joukossa jäsenyysaste nuorten ihmisten joukkoon taas alkaa hiljalleen pienentyä tietyn iän jälkeen.



Kuva 1.6. Jäsenyysasteet nuorten ihmisten perinteisessä (vas.) ja sumeassa joukossa.

Jäsenyysfunktion muoto ja arvot ovat tutkijan määritettävissä tai ne saadaan tutkimusaineiston perusteella. Jatkossa tarkastelemme tältä osin useita esimerkkejä. Muodollisesti jäsenyysfunktiot ovat yleensä jatkuvia kuvauksia muuttujan arvojen joukosta suljetulle reaalilukuvälille [0,1]. Kuvassa 1.7 on tyypillisiä jäsenyysfunktioita (yleensä nämä funktiot saavat arvon 1 ainakin yhdessä pisteessä eli ne ovat silloin normalisoituja (normalized)).



Kuva 1.7. Jäsenyysfunktioita: ensin vasemmalla kaksi kolmion muotoista (triangular), sitten kellokäyrän muotoinen (Gaussian tai bell-shaped) ja puolisuunnikas (trapezoidal).

Sumeat joukot (tai itse asiassa niiden jäsenyysfunktiot) kuvaavat erityisesti epätäsmällisiä asioita kahdella tavalla. Ensinnäkin eri tavoin määritetyille sumeille joukoille voidaan antaa sopivia nimiä niiden sijainnin ja muodon perusteella, esimerkiksi malliemme tuottamia sumeita joukkoja voidaan nimetä sopivasti (labeling). Toiseksi, kielelliset ilmaisut (lähinnä termit) voidaan esittää sumeina joukkoina tietokonemalleissa. Jos esimerkiksi tarkastelemme ihmisten osalta heidän ikäänsä, kyseinen muuttuja voi saada perinteisiä numeerisia arvoja, mutta voimme antaa sille myös kielellisiä (ja likimääräisiä) arvoja, ja jälkimmäisessä tapauksessa nämä arvot voivat perustua vaikkapa Osgoodin tai Likertin asteikkoihin. Niinpä voimme iän tapauksessa käyttää vaikkapa likimääräisiä arvoja

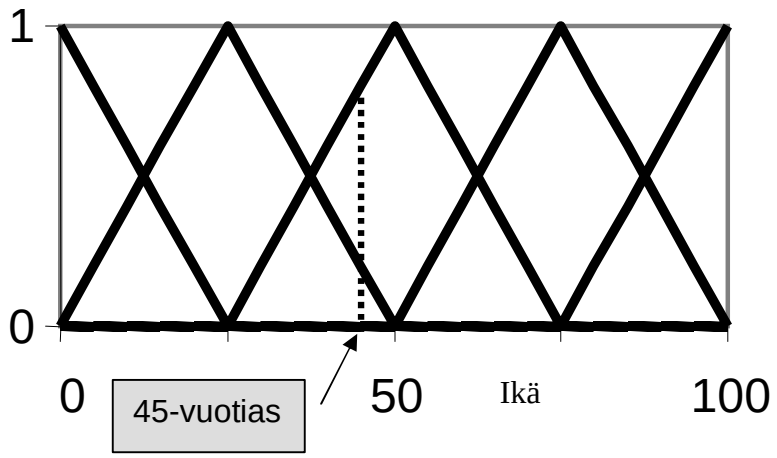
> nuori – melko nuori – ei nuori eikä vanha (tai keski-ikäinen) – melko vanha – vanha.

Likertin asteikon tapauksessa taas väitteelle *olen nuori* voidaan antaa arvoja

> täysin samaa mieltä – jokseenkin samaa mieltä – jokseenkin eri mieltä – täysin eri mieltä.

Edellisessä tapauksessa vastaavien sumeiden joukkojen jäsenyysfunktiot voivat olla kuvan 1.8 mukaisia. Tavallisesti valitut jäsenyysfunktiot ovat osittain toistensa päällä, mikä tarkoittaa, että oliot voivat kuulua osittain kahteen tai useam-

paan joukkoon yhtä aikaa (tämä on usein tärkeää mallinnuksen kannalta). Niinpä 45-vuotias näyttää kuuluvan yhtä aikaa pienellä jäsenyysasteella melko nuoriin ja aika suurella jäsenyysasteella keski-ikäisiin ihmisiin (katkoviiva kuvassa).



Kuva 1.8. Iän likimääräisiä arvoja sumeina joukkoina (vasemmalta oikealle: nuori, melko nuori, keski-ikäinen, melko vanha ja vanha)

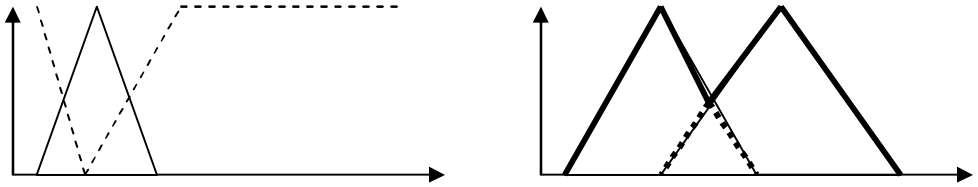
Sumeiden joukkojen avulla voimme siis käyttää tietokoneympäristössä kielellisiä muuttujia (linguistic variable), joiden arvot ovat kielellisiä ilmaisuja, yleensä epätäsmällisiä termejä. Tietyssä mielessä kielellisten ilmaisujen ”tulkinnat” ovat siis tietokoneympäristössä sumeita joukkoja, ja näin ollen voimme soveltaa kielellisiin malleihin yksinkertaista laskentaa. Tällaiset kielelliset muuttujat ovat monissa tilanteissa käyttökelpoisempia tai jopa ainoita vaihtoehtoja kuten myöhemmin tulemme toteamaan. Tyypillisiä esimerkkejä tällaisista arvoista ovat *melko nuori*, *hyvin pitkä*, *kohtalaisen matala*, *ei kovin uusi*, *kallis tai melko kallis*, *hyvin todennäköinen*, *ei lähellä*, *noin viisi* sekä *noin välillä kolmesta kuuteen*.

Näitä arvoja muodostetaan tavallisesti määrittämällä ensin muuttujan osalta vastakohtaiset primitiiviset termit (yleensä adjektiiveja) kuten iän tapauksessa *nuori* ja *vanha*. Sitten näiden termien avulla muodostetaan muita arvoja käyttämällä ensin sopivia adverbejä (modifiers tai hedges, hyvin, melko jne.). Tämän jälkeen voidaan käyttää myös ilmaisuja *ei*, *ja* sekä *tai* (sumeissa sään-

nöissä myös *jos-niin*). Niinpä sitten voimme käyttää vaikkapa arvoa *nuori ja ei hyvin nuori*. Käytännössä arvojen muodostus on melko vapaata, kunhan ne ovat mielekkäitä. Luonnollisesti voimme käyttää näiden muuttujien kanssa yhdessä myös perinteisiä kielellisiä ja numeerisia arvoja.

Kielellisten arvojen ”tulkinta” sumeina joukkoina perustuu siis sumean joukko-opin operaatioihin. Tavallisesti käytetään seuraavia operaatioita: Olkoon muuttujana ihmisten ikä, ja primitiivisinä arvoina *nuori* ja *vanha*.

1. Adverbejä liitettäessä siirto vaakasuunnassa on suositeltavin. Niinpä esimerkiksi arvoja *hyvin nuori* ja *melko nuori* edustavat jäsenyysfunktiot olisivat samanlaisia arvon *nuori* kanssa, mutta edellinen olisi *nuoren* vasemmalla ja jälkimmäinen oikealla puolella (kuva 1.8).
2. Ei-sanalla muodostetun arvon jäsenyysfunktio saadaan tavallisesti vähentämällä ykkösestä alkuperäisen arvon jäsenyysaste. Joukko-opillisesti kyse on tällöin joukosta ja sen komplementista (complement). Jos esimerkiksi 25-vuotias kuuluu jäsenyysasteella 0,7 nuorten ihmisten joukkoon, hän kuuluu (samanaikaisesti) jäsenyysasteella 1-0,7 = 0,3 ei-nuorten joukkoon (kuva 1.9).
3. Jos yhdistämme kaksi arvoa ja-sanalla, joukko-opillisesti kyse on vastaavien sumeiden joukkojen leikkauksesta (intersection). Yksi yleinen tapa tämän joukon määrittämiseksi on laskea alkuperäisten joukkojen jäsenyysasteiden minimit, mutta monia muitakin laskutapoja on tarjolla. Jos esimerkiksi henkilö kuuluu jäsenyysasteilla 0,7 ja 0,5 vastaavasti nuorten ja melko nuorten joukkoihin, hän kuuluu jäsenyysasteella min(0,7;0,5) = 0,5 joukkoon *nuori ja melko nuori* (kuva 1.9).
4. Jos yhdistämme kaksi arvoa tai-sanalla, joukko-opillisesti kyse on vastaavien sumeiden joukkojen yhdisteestä (union). Yksi yleinen tapa tämän joukon määrittämiseksi on laskea alkuperäisten joukkojen jäsenyysasteiden maksimit, mutta tässäkin monia muitakin laskutapoja on tarjolla. Jos esimerkiksi henkilö kuuluu jäsenyysasteilla 0,7 ja 0,5 vastaavasti nuorten ja melko nuorten joukkoihin, hän kuuluu jäsenyysasteella max(0,7;0,5) = 0,7 joukkoon *nuori tai melko nuori* (kuva 1.9).



Kuva 1.9. Vasemmalla sumea joukko ja sen komplementti. Oikealla kaksi kolmion muotoista sumeaa joukkoa ja niiden leikkaus (katkoviiva) sekä yhdiste (paksu viiva).

Voimme siis yhdistää kielelliset ilmaisut ja sumean joukko-opin operaatiot seuraavasti:

1. Adverbit (modifiers, hedges): joukkojen siirrot vaakasuunnassa.
2. Ei-sana eli negaatio (negation): joukon komplementti.
3. Ja-sana eli konjunktio (conjunction): joukkojen leikkaus.
4. Tai-sana eli disjunktio (disjunction): joukkojen yhdiste.

Näin olemme liittäneet toisiinsa kielellisten muuttujien arvoissa käytetyt kielelliset ilmaisut ja niiden matemaattiset tulkinnat sumeina joukkoina.

Edellä mainittujen puhtaasti kielellisten arvojen lisäksi voimme käyttää numeerisempiakin arvoja kuten *noin 5* tai *likimain välillä 4 – 6*. Perinteisten funktioiden ja relaatioiden lisäksi voidaan käyttää myös likimääräisiä funktioita (*y on likimain* x^2+4) tai relaatioita (*x on hieman suurempi kuin y*).

Seuraavaksi tarkastelemme sumeaa ryhmittelyanalyysiä, sillä sumeiden systeemien mallit perustuvat usein tähän tekniikkaan ja edellä kuvattuun ”sumeaan kieleen”.

1.3. Sumea ryhmittelyanalyysi

Sumeassa ryhmittelyanalyysissä (fuzzy clustering) määritetyt ryhmät ovat sumeita joukkoja, ja niinpä tilastoyksiköt voivat kuulua useampaan ryhmään yhtä aikaa eri jäsenyysasteilla. Tunnetuin menetelmä on James Bezdekin [8] kehit-

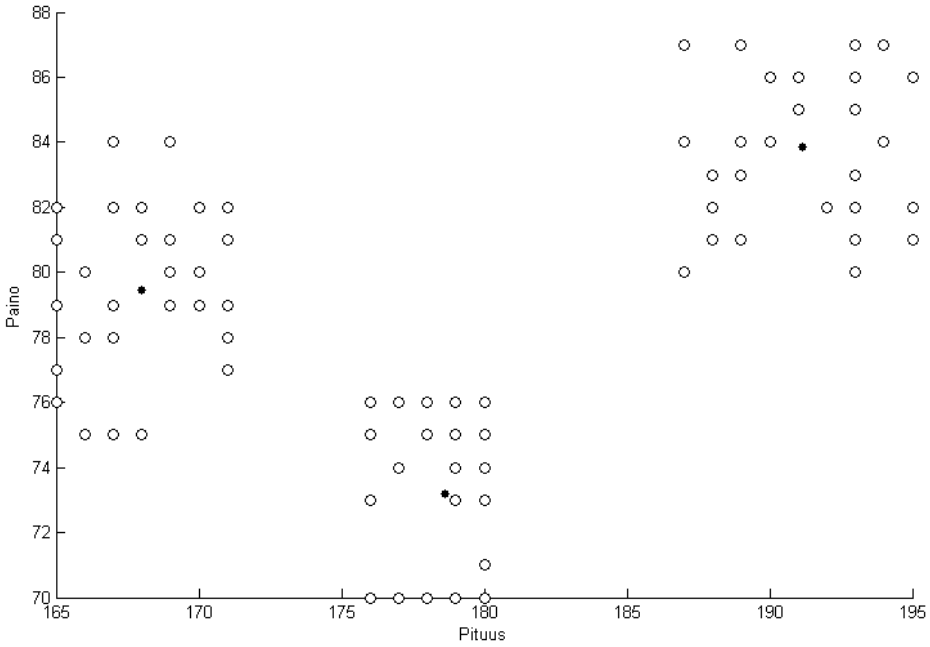
tämä fuzzy c-means –algoritmi (FCM), joka perustuu perinteisessä mallinnuksessa käytettyyn ISODATA-algoritmiin. SPSS:n k-means –ryhmittelyn perustana on myös ISODATA.

FCM-algoritmi etsii etukäteen annetun määrän ryhmiä siten, että tilastoyksiköille lasketaan jäsenyysasteet (jotka perustuvat yksiköiden etäisyyksiin ryhmien keskuksista) jokaisen ryhmän osalta. Aluksi se tuottaa satunnaisesti ryhmien keskukset ja pyrkii sitten optimoimalla muuttamaan niitä siten, että tilastoyksiköiden jäsenyysasteet ryhmiin ovat mahdollisimman suuria tai pieniä, eli rajatapauksia (jäsenyysaste noin 0,5) on mahdollisimman vähän. Satunnaisista alkuarvoista johtuen samalle aineistolle tuotetut ratkaisut eivät ole joka kerta täysin samoja. FCM myös määrittää jäsenyysasteet siten, että jokaisen tilastoyksikön osalta sen kaikkien jäsenyysasteiden summa on yksi. Me käytämme FCM:n tapauksessa jatkossa Matlab'in™ Fuzzy Logic Toolbox'in *fcm*-algoritmia (tarkempia tietoja Matlab-komennolla *help fcm*).

Tarkastellaan taas edellä mainittuja miesten pituuksia ja painoja, joihin sovellamme kolmen ryhmän FCM-ratkaisua (Matlab-komento *fcm(data,3),* kun *data* on havaintomatriisi). Silloin saamme ryhmien keskukset

	Ryhmä 1	Ryhmä 2	Ryhmä 3
Pituus cm	168,0	178,6	191,2
Paino kg	79,4	73,2	83,8

Kuvassa 1.10 on esitetty näiden ryhmien keskukset ja huomaamme, että ne voivat olla "teoreettisia" eli vastaavia pisteitä ei ole aineistossamme.



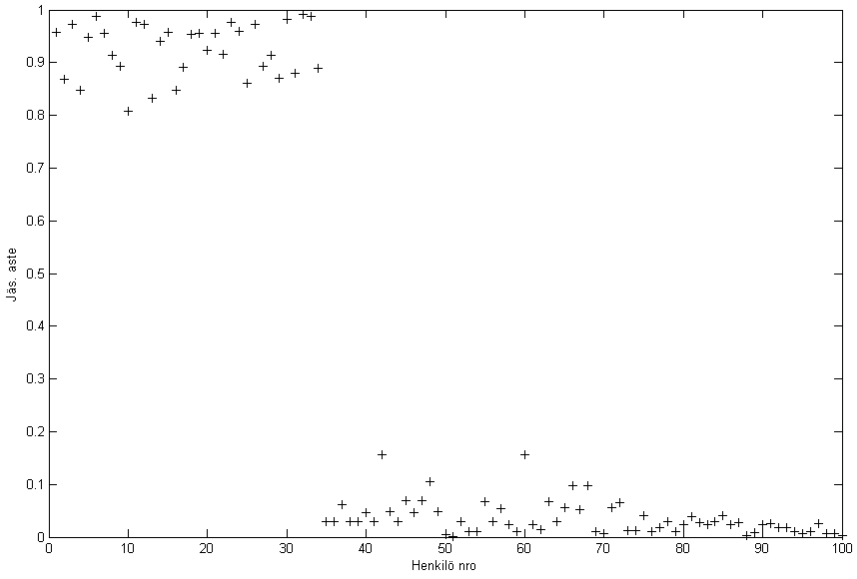
Kuva 1.10. FCM-algoritmin tuottamat ryhmien keskukset (●) sadan miehen aineistosta.

Kuvassa 1.11 on esitetty vastaavat tilastoyksiköiden (miesten) jäsenyysasteet kuvan 1.10 vasemmassa reunassa olevaan ryhmään. Huomaamme, että useimmilla kahteen muuhun ryhmään varsinaisesti kuuluvilla yksiköillä on pienet jäsenyysasteet myös kyseiseen ryhmään. Rajatapauksia ei kuitenkaan näytä esiintyvän (siis noin 0,5 olevia arvoja). Koska olemme tuottaneet kolmen ryhmän ratkaisun, jokaisen henkilön osalta saamme kolme jäsenyysastetta, yhden kuhunkin ryhmään. Lisäksi FCM siis tuottaa aina jokaiselle yksikölle ryhmiin sellaiset jäsenyysasteet, että niiden summa on yksi (tässä henkilön jäsenyysasteiden summa kolmeen ryhmään on aina 1).

Voimme suuntaa-antavasti tarkastella ryhmittelymme hyvyyttä testaamalla, poikkeavatko ryhmien jäsenyysasteet arvosta 0,5 esimerkiksi t-testin avulla, jolloin nollahypoteesin hylkääminen tarkoittaa vähäistä rajatapauksien määrää. Jos taas pyöristämme jäsenyysasteet nolliksi ja ykkösiksi, saamme perinteisen, ”kovan” (hard, crisp) ryhmittelyn, joka on analoginen SPSS:n k-means –ryhmittelyn kanssa. Tällöin ryhmien erillisyyttä voidaan arvioida jokaisen muuttujan osalta erikseen edellä kuvatulla tavalla F-testillä kuvittelemalla varianssianalyysin tapaan, että nämä ryhmät ovat ”käsittelyjä” (treatment). Isot F-

testisuureen arvot, eli kun ryhmien välisten varianssien ja ryhmien sisäisten varianssien välinen suhde on suuri, antavat aiheen olettaa, että ryhmät ovat homogeenisiä ja ne erottuvat hyvin toisistaan, siis toisin sanoen, ryhmittely on onnistunut hyvin.

Meidän aineistomme tapauksessa näiden testien perusteella rajatapauksia on vähän ja molemmat muuttujat ovat ryhmittelyn kannalta tärkeitä. Siis tässä mielessä meillä on hyvin onnistunut kolmen ryhmän ratkaisu.



Kuva 1.11. Miesten jäsenyysasteet ryhmään yhden ryhmän osalta.

Edellä olevan perusteella kuvan 1.5 kaksi vasemmanpuoleista ongelmatapausta voidaan tulkita sumeiksi joukoiksi, jolloin aivan vasemmalla olevassa ryhmässä pisteiden jäsenyysaste ryhmän keskukseen on sitä suurempi, mitä lähempänä ne ovat ryhmän keskusta (musta tiheä alue keskellä). Toinen ryhmä taas koostunee itse asiassa kahdesta ryhmästä, ja vaakasuunnassa keskellä olevat pisteet kuuluvat noin jäsenyysasteella 0,5 molempiin ryhmiin.

Sumean ryhmittelyanalyysin etuna on, että emme perinteisten tekniikoiden tavoin pakota tilastoyksiköitä kuulumaan täysin tai ei ollenkaan mihinkään ryhmään, vaan sallimme myös osittaisen kuulumisen, jopa useampaan ryhmään yhtä aikaa. Nämä osittain ryhmiin kuuluvat yksiköt usein sitten havahduttavat

tutkijan tekemään sellaisia lisätutkimuksia, jotka perinteisissä tapauksissa ehkä jäävät tekemättä. Jos ryhmittelyanalyysiä sovelletaan vaikkapa hahmontunnistuksessa potilaiden röntgenkuvien analyysiin, sumea ryhmittely voi löytää sellaisiakin patologisia tapauksia, joita muuten ei löydettäisi. Samoin vaikkapa pankin lainanhakijoiden joukosta sumea ryhmittely voi määrittää henkilön luotettavuudeltaan rajatapaukseksi, vaikka perinteisellä ryhmittelyllä hän olisi luotettava asiakas.

Sumeat ryhmät, kuten kaikki sumeat joukot, voidaan lopuksi tarvittaessa täsmällistää (defuzzify) eli pyöristää ryhmien jäsenyysasteet nolliksi ja ykkösiksi, jolloin saamme perinteisiä ”kovia” ryhmiä. Täsmällistäminen voi tarkoittaa myös sitä, että sumea joukko muunnetaan yksittäiseksi reaaliluvuksi. Tällöinkin tulokset ovat usein parempia kuin perinteisin menetelmin, varsinkin säädön ja päätöksenteon sovelluksissa. Tilanne on analoginen laskutoimitusten pyöristysten kanssa: jos pyöristämme tuloksia jokaisen välituloksen jälkeen, lopputulos voi olla hyvinkin virheellinen (perinteinen tekniikka). Toisaalta, jos pyöristämme vasta lopputuloksen, saamme paremmin oikeita tuloksia (sumeat systeemit).

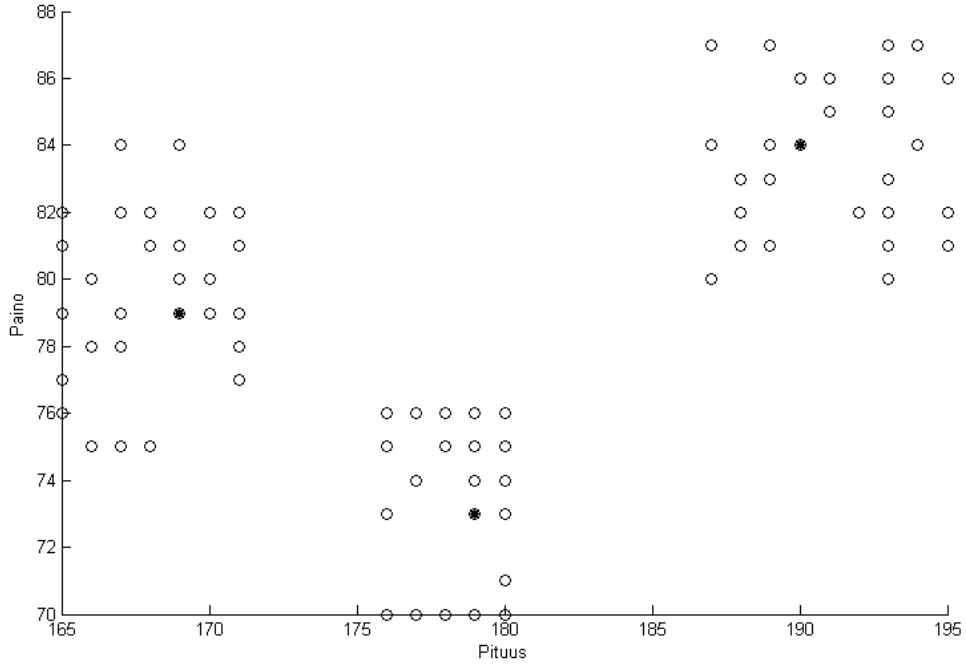
Toinen paljon käytetty sumea ryhmittelymenetelmä on alkuaan Ronald Yagerin [14] [66] kehittämä *subtractive clustering* -algoritmi. Se valitsee aina ryhmien keskuksiksi aineistossa olevia pisteitä eli havaintovektoreita ja ryhmien määrä voidaan valita vain epäsuorasti tutkijan antaman vaikuttavuussäteen (range of influence) perusteella.

Aluksi annetaan säteen pituudelle jokin arvo, ja sitten pistejoukkoja tarkastellaan aina tämän säteen määrittämässä muuttuja-avaruuden ympyröissä tai palloissa (säde on normeerattu välille nollasta ykköseen). Ensimmäisen ryhmän keskukseksi valitaan se havaintovektori, jonka ympärillä on kyseisellä säteellä tiheimmin muita havaintovektoreita. Seuraavan ryhmän keskus on samaa sädettä käytettäessä toiseksi tiheimmässä keskuksessa oleva vektori, mutta kuitenkin siten, että nämä keskukset ovat mahdollisimman etäällä toisistaan. Kolmas ryhmän keskus etsitään samalla periaatteella kolmanneksi tiheimmästä keskittymästä, mutta samalla sen taas pitää tietenkin olla mahdollisimman kaukana muista keskuksista. Tällä tavoin edetään kunnes kaikki pisteet ovat jonkin ryhmän keskuksen määrittämän pallon sisällä. Itse asiassa on siis taas tietyssä mielessä kyse ryhmien sisäisen varianssin minimoinnista ja ryhmien välisen varianssin maksimoinnista.

Jos sovellamme tätä algoritmia sadan miehen aineistoomme käyttämällä Fuzzy Logic Toolboxia (komento: *subclust(data,r)*, kun 0<r≤1, ks. *help subclust*), saamme seuraavan kolmen ryhmän ratkaisun säteen r arvolla 0,5 (haluttu määrä ryhmiä saadaan siis kokeilemalla säteen eri suhteellisilla pituuksilla):

	Ryhmä 1	Ryhmä 2	Ryhmä 3
Pituus cm	169	179	190
Paino kg	79	73	84

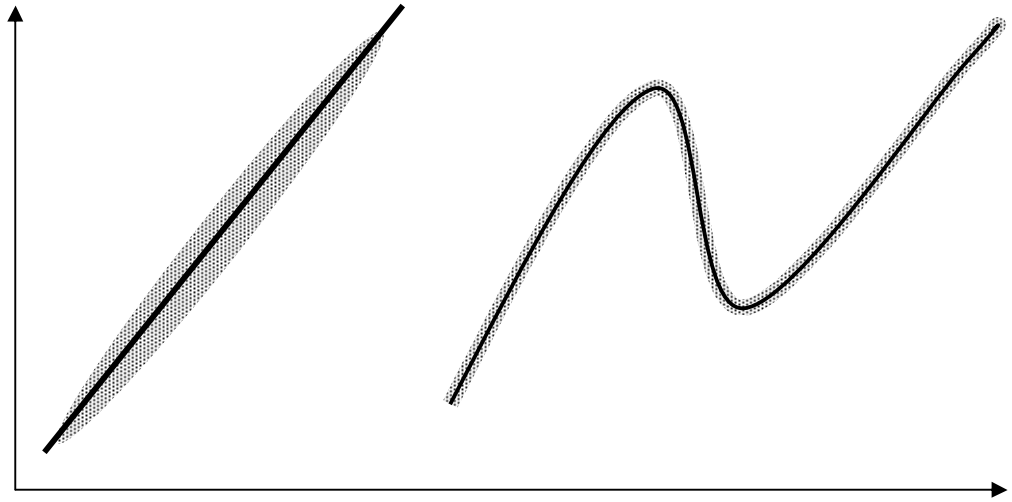
Nämä ovat nyt siis samalla aineistossamme olevia todellisia havaintovektoreita. Kuvassa 1.12 näkyvät kyseiset ryhmien keskukset. Matlab'in neuro-sumea *anfisedit*-mallinnusohjelma käyttää tätä ryhmittelytekniikkaa.



Kuva 1.12. Subclust-algoritmin tuottamat ryhmien keskukset (●) sadan miehen aineistosta.

Edellä ryhmien keskukset ovat olleet yksittäisiä pisteitä, mutta niiden sijasta keskuksina voivat olla myös erilaiset suorat ja käyrät. Tällöin on itse asiassa sovellettu niin sanottua switching regression –menetelmää, josta jo vuonna 1993 Richard Hathaway ja James Bezdek esittivät sumean version. Heidän

menetelmänsä avulla laskettiin pisteiden sijasta havaintoarvojen jäsenyysasteita ryhmien keskuksina oleviin suoriin tai käyriin. Nämä keskukset voivat sitten olla puhtaasti matemaattisia funktioita tai sumeiden mallien avulla tuotettuja käyriä (kuva 1.13). Tämä menetelmä soveltuu usein sellaisiin tapauksiin, joissa ryhmät eivät ole muodoltaan pallomaisia.



Kuva 1.13. Ryhmien keskukset voivat olla myös suoria tai käyriä, jotka perustuvat matemaattisiin funktioihin tai sumeisiin malleihin.

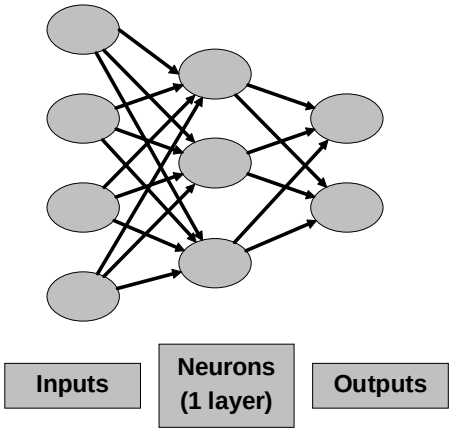
Samaa tekniikkaa voidaan sitten soveltaa suoraan myös regressioanalyysiin, jolloin yhden regressioyhtälön sijasta saadaan useampi yhtälö tai vastaava sovite (vrt. luku 5). Tämä menetelmä muistuttaa myös siinä mielessä perinteistä kovarianssianalyysiä, että ryhmien todellisempien erojen tarkastelemiseksi jokaisesta ryhmästä tuotetaan oma (lineaarinen) regressioyhtälönsä.

1.4. Ryhmittelyä adaptiivisten järjestelmien avulla

Adaptiiviset järjestelmät (adaptive systems) ovat sellaisia ”oppivia” järjestelmiä, jotka rakennetaan tai muokataan numeerisen aineiston perusteella. Neurolaskenta (neural computing) eli ”neuroverkot” (neural networks, jälkimmäistä, joskin hieman harhaanjohtavaa käsitettä, suuri yleisö käyttää enemmän) ovat yksi tämän sovellusalueen käyttökelpoinen lähestymistapa mallinnukseen ja,

toisin kuin vaikkapa sumeat systeemit, niissä ei järjestelmiin liittyviä asioita aina nimetä kielellisesti, vaan mallinnus perustuu pelkästään numeerisen aineiston käsittelyyn (mittaukset, havainnot jne.). Niinpä nämä mallit voivat olla käyttäjälle sellaisia ”mustia laatikoita”, jotka toimivat hyvin, mutta joiden yksityiskohtainen toimintaperiaate on niiden sisältämien lukuisten funktioiden vuoksi vaikea ymmärtää. Tyypillisiä sovelluskohteita ovat ryhmittelyyn ja regressiomalleihin liittyvät systeemit.

Neurolaskenta soveltaa ihmisaivojen toimintaa. Aivoissa hermosolut muodostavat monimutkaisen verkoston, ja tässä verkostossa tapahtuvien ärsykkeiden ja reaktioiden oletetaan olevan ihmisen ajattelun perustana. Tietokoneympäristössä hermosoluja vastaavat laskentaelementit, joita kutsutaan neuroneiksi (neuron) [22]. Jokaisella neuronilla on monta syötettä, mutta vain yksi vaste. Neuroverkko muodostuu neuroneja sisältävistä kerroksista (layer), ja uloimpina ovat syöte- ja vastekerros. Näiden välissä on yksi tai useampia piilokerroksia (hidden layer):



Jokainen neuroni käsittää tavallisesti funktion

$$y = \Sigma_i w_i \cdot x_i$$

missä w_i ovat painokertoimia, x_i neuronin syötteitä ja y on vaste.

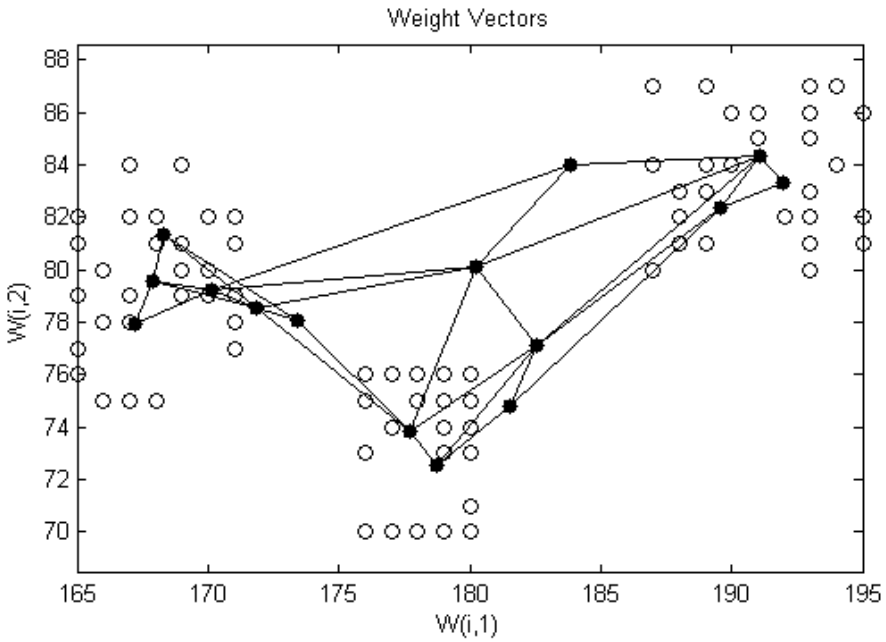
Syötekerroksen syötteinä ovat aineistomme syötearvot, ja tavoitteena on neuroverkon painojen arvoja muuttamalla tuottaa sellaisia vastekerroksen vas-

teiden arvoja, jotka ovat mahdollisimman samanlaisia aineistomme vastearvojen kanssa. Yleensä tämä optimointi tapahtuu siten, että esimerkiksi gradienttimenetelmää (gradient descent method) käyttäen muutamme painojen arvoja vähän kerrallaan kunnes tarpeeksi hyvä lopputulos on saavutettu. Näitä painojen muutoskertoja eli opetuskierroksia (learning epoch) voi olla jopa tuhansia ennen sopivaa lopputulosta. Tämäntyyppistä lähestymistapaa kutsutaan vastavirta-algoritmiksi (backpropagation algorithm), muita muitakin neuroverkkotyyppejä on kehitetty.

Adaptiivisten järjestelmien tapauksessa yksi tunnettu ryhmittelyyn sopiva menetelmä on akateemikko Teuvo Kohosen kehittämä itseorganisoituva kartta (self-organized map, SOM [30]), joka löytää hyvin ryhmien lisäksi myös näiden ryhmien muodostaman rakenteen.

Itseorganisoituvat kartat sijoittavat yleensä aluksi tutkijan valitseman määrän ryhmien keskuksia satunnaisesti muuttuja-avaruuteen, ja tämän jälkeen nämä alkupisteet (”neuronit”) pyrkivät liikkumaan kohti suurimpia havaintovektorien tihentymiä. Tällaisten järjestelmien haittapuolena on se, että tulosten tulkinta on usein vaikeaa monimutkaisen laskennan vuoksi. Onneksi tulkintaa voidaan usein näissä tapauksissa helpottaa sumean logiikan avulla esittämällä mallin vuorovaikutukset sumeina sääntöinä, ja tätä aihetta käsitellään myöhemmin tarkemmin. On myös olemassa sumeita itseorganisoituvia karttoja [10].

Kuvassa 1.14 SOM-algoritmi on tuottanut Matlab’in Neural Network Toolboxin *newsom*- ja *train*-komennoilla viidentoista neuronin avulla ”rautalankamallin” sadan miehen aineistostamme käyttämällä optimointiin 100 opetuskierrosta . Tällä tavoin voimme esittää pelkistetysti aineistomme perusrakenteen muutamien tyypillisten pisteiden ja niiden välistä etäisyyttä kuvaavien janojen avulla.



Kuva 1.14. SOM-algoritmilla tuotettu kartta sadan miehen aineistosta pituuksien ja painojen perusteella.

2. SUMEA PÄÄTTELY MALLIN RAKENNUKSEN PERUSTANA

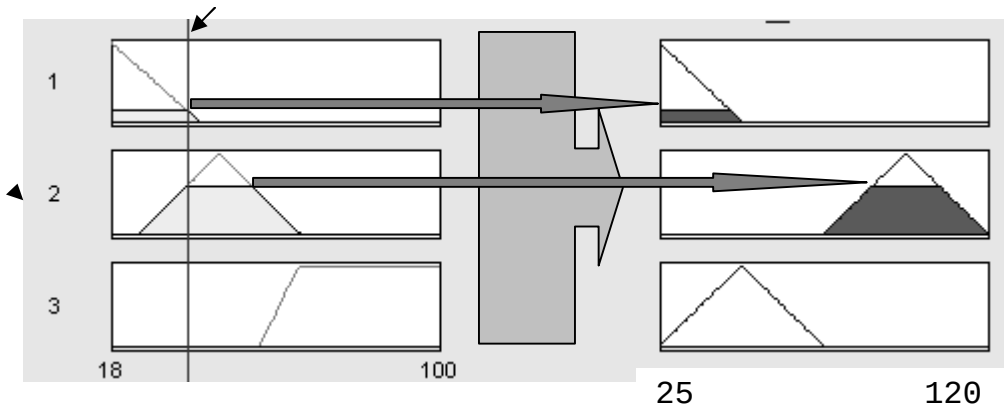
Edellä olemme jo tarkastelleet sumeaa ryhmittelyanalyysiä, koska se on perustana monissa muissa mallinnuksissa. Nyt tutustumme sumean päättelyn (fuzzy reasoning) perusteisiin, ja tämä päättelymenetelmä on hyvin sovelluskelpoinen monissa erityyppisissä malleissa. Rakennamme mallejamme vertaillen niitä samalla vastaaviin perinteisiin malleihin. Mallin rakentamisen ajattelemme tapahtuvan seuraavasti:

1. Määritämme malliin kuuluvat muuttujat. Usein osa muuttujista on syötemuuttujia (input variables) ja loput vastemuuttujia (output variables).
2. Pyrimme löytämään tai määrittämään muuttujien välisiä rakenteita tai yhteyksiä. Yhteydet muodostetaan ”asiantuntijoiden” tietojen (expertise) tai tutkimusaineiston (data) perusteella. Nämä yhteydet esitetään tavallisesti kielellisessä muodossa, esimerkiksi jos-niin –tyyppisinä sääntöinä.
3. Sumean päättelyn avulla tuotamme mallin tulosteita kohdan 2 tietojen perusteella. Tulosteet ovat reaalilukuja tai sumeita joukkoja.
4. Arvioimme saatuja tulosteita eli testaamme mallin hyvyyttä.
5. Tarvittaessa muokkaamme eli viritämme (tune) mallia paremmaksi.
6. Lopputuloksena pitäisi olla ”hyvä” malli.

Perinteinen kvantitatiivinen mallinnustapa pyrkii käyttämään sopivia matemaattis-tilastollisia funktioita, mutta tällöin voimme kohdata kaksi ongelmaa. Ensinnäkin sopivan funktion löytäminen voi olla hyvin vaikeaa. Toiseksi, jos sopiva funktio löytyy, se voi olla hyvin monimutkainen, vaikeasti ymmärrettävä (jopa ns. musta laatikko) tai laskennallisesti raskas. Lisäksi matemaattinen mallinnus edellyttää hyvät tiedot matematiikasta.

Sumeaa päättelyä soveltavissa malleissa taas ei näitä ongelmia useinkaan ole. Tässä tapauksessa muuttujat voivat saada myös epätäsmällisiä arvoja, muuttujien väliset yhteydet esitetään yleensä likimääräisesti jos-niin –tyyppisinä kielellisinä sääntöinä (implikaatioina) ja malleja voidaan tarvittaessa parantaa adaptiivisten järjestelmien ja geneettisten algoritmien avulla. Niinpä nämä mallit ovat yleensä käyttäjälle helposti ymmärrettävässä kielellisessä muodossa ja vastaavat tietokoneympäristössä automaattisesti toteutetut matemaattisetkin operaatiot soveltavat melko yksinkertaista laskentaa.

Kuva 2.1 luonnehtii tyypillistä sumeaan päättelyyn perustuvaa laskentaa kun käytämme kolmea sumeaa sääntöä. Tällöin laskemme ensin antamamme syötteen arvon samanlaisuusasteen (degree of similarity) erikseen jokaisen säännön etujäsenen kanssa, ja nämä arvot perustuvat itse asiassa syötteen etäisyyteen etujäsenistä siten, että mitä pienempi etäisyys, sitä suurempi samanlaisuuden aste. Näitä lukuja, eli sääntöjen ”laukaisuvoimakkuuksia” (firing strength) käytetään sitten sääntöjen vastaavien takajäsenien painoina. Lopuksi lasketaan mallin vasteen arvo sopivana yhdistelmänä näiden painojen perusteella painotetuista takajäsenistä. Tällainen sopiva yhdistelmä voi olla vaikkapa painotettujen takajäsenien yhdistäminen tai takajäsenien painotettu keskiarvo. Me siis oikeastaan sovellamme sääntöjen avulla interpolointia annetuille syöttearvoille.



Kuva 2.1. Kielelliset sumeat säännöt sumeiden joukkojen avulla esitettyinä. Etujäsenet vasemmalla. Syöte (tässä pystyviiva) on lähimpänä toisen säännön etujäsenen sumean joukon keskusta, joten sillä suurin laukaisuvoimakkuus (tumman alueen korkeus). Vastaavasti toisen säännön takajäsenellä on silloin suurin paino vastearvossa. Tässä vastearvo perustuu vain ensimmäiseen ja toiseen sääntöön, koska kolmannen säännön laukaisuvoimakkuus on nolla (vrt. kuva 2.3).

Seuraavaksi tarkastellaankin tarkemmin muuttujien välisten yhteyksien esittämistä sumeiden sääntöjen avulla. Niitä siis voidaan tehdä asiantuntemuksen, empiirisen aineiston tai näiden molempien perusteella.

2.1. Sumeiden sääntöjen muodostus ilman havaintoaineistoa

Jos havaintoaineistoa ei ole saatavilla, muuttujien väliset yhteydet kuvataan tavallisesti niin sanottujen asiantuntijoiden näkemysten (expertise) perusteella. Mallin hyvyys perustuu kuitenkin tässäkin tapauksessa sen käyttökelpoisuuteen. Yleensä tämä lähestymistapa on työläämpi kuin aineistoon perustuva mallinnus, ja varsinkin mallin hienosäätö voi viedä paljonkin aikaa.

2.1.1. Yksi syöte- ja vastemuuttuja

Tarkastellaan aluksi sellaista yksinkertaista mallia, jossa meillä on yksi syöte- ja vastemuuttuja, ja tarkastelemme näiden muuttujien välistä yhteyttä. Olkoon ne vastaavasti ihmisten ikä ja heidän asuntonsa pinta-ala, ja molemmat voivat saada epätäsmällisiä tai täsmällisiä arvoja. Sumeat sääntömme, jotka kuvaavat näiden muuttujien välistä yhteyttä, ovat silloin muotoa

Jos ikä on noin _ , niin asunnon koko on noin _.

Meidän on määritettävä itse tarvittavat suhteet muuttujien välille käyttämällä sopivia kielellisiä arvoja. Usein tässä voidaan hyödyntää Osgoodin tai Likertin asteikkoja. Esimerkiksi näin:

- Muuttujat: henkilön ikä (v) ja asunnon pinta-ala (m^2).
- Muuttujan *ikä* arvot: nuori, melko nuori, keski-ikäinen, melko vanha, vanha. Vaihteluväli 18 – 100 v. Muutama arvo muuttujaa kohti riittää.
- Muuttujan *pinta-ala* arvot: pieni, melko pieni, keskikokoinen, melko suuri, suuri. Vaihteluväli 25 – 120 m^2.
- Määritetään vastaavat sopivat sumeat joukot edustamaan muuttujien arvoja.
- Muodostetaan jos-niin –tyyppisiä sumeita sääntöjä kuvaamaan muuttujien välistä yhteyttä, joissa jos-osaa kutsutaan etujäseneksi (antecedent) ja niin-osaa takajäseneksi (consequent). Säännöt ovat siis implikaatioita logiikan näkökulmasta. Etujäsen liittyy yleensä syöte- ja takajäsen vastemuuttujiin.

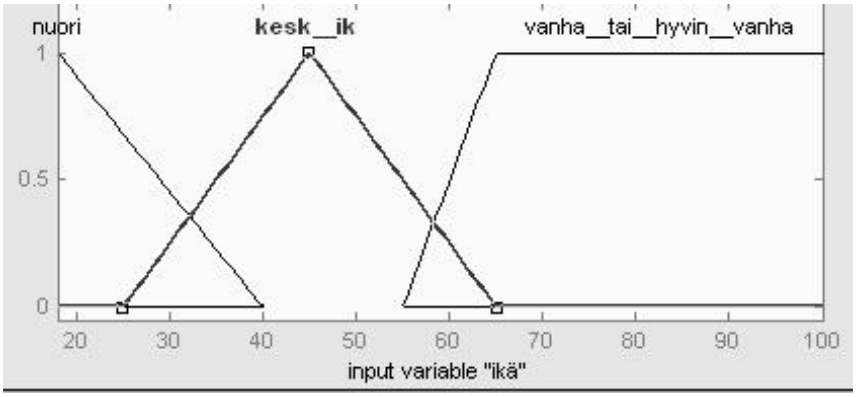
Olkoon asiantuntijoiden esittämä tieto tässä tapauksessa seuraavaa:

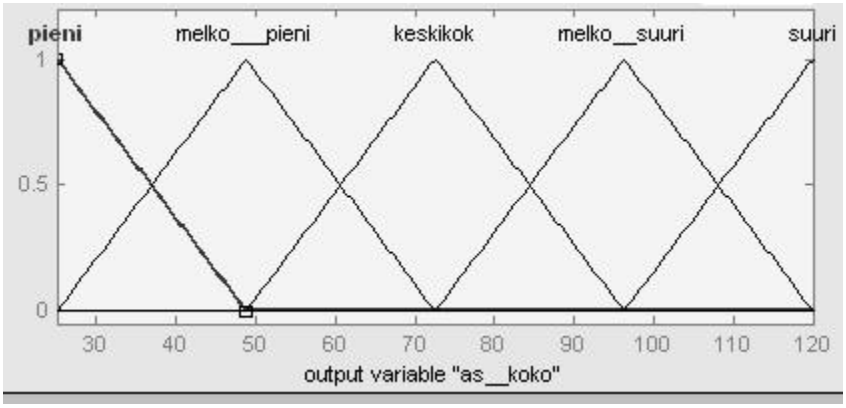
1. Nuoret ihmiset asuvat pienissä asunnoissa.
2. Keski-ikäiset asuvat melko suurissa asunnoissa.
3. Vanhat ihmiset asuvat taas melko pienissä asunnoissa.
4. Hyvin vanhat ihmiset asuvat melko pienissä asunnoissa.

Muutamme nämä väitteet sumeiksi säännöiksi, eli ”sumean logiikan kieleksi”, käyttämällä edellä kuvattuja ”kielioppisääntöjä” ja samalla hieman asioita pelkistäen:

1. Jos henkilö on nuori, hän asuu pienessä asunnossa.
2. Jos henkilö on keski-ikäinen, hän asuu melko isossa asunnossa.
3. Jos henkilö on vanha tai hyvin vanha, hän asuu melko pienessä asunnossa.

Kuvassa 2.2 on sitten yksi ehdotelma kielellisiä arvojamme edustavista sumeista joukoista, ja näiden joukkojen pitäisi yleensä olla osittain päällekkäin (yleensä noin 25 % joukon pinta-alasta yhteistä aluetta).





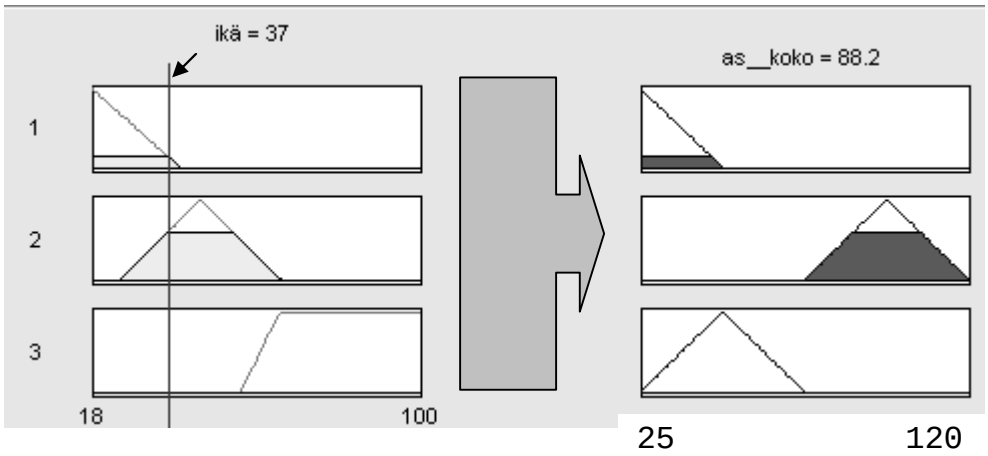
Kuva 2.2. Iän (ylh.) ja asunnon koon kielellisiä arvoja edustavat sumeat joukot.

Matlab'in *fuzzy*-komennon avulla voimme nyt rakentaa mallimme graafisen käyttöliittymän avulla kun syöte- ja vastemuuttujina ovat vastaavasti ikä ja asunnon koko. Jos käytämme kolmion ja puolisuunnikkaan muotoisia sumeita joukkoja muuttujien arvoille, edellä olevat säännöt voidaan esittää sumeina joukkoina kuvan 2.3 tapaan. Kielellisellä tasolla sumeat säännöt siis käsittävät muuttujan arvoina olevia, mahdollisesti epätäsmällisiä termejä, kun taas joukko-opillisella tasolla niissä on vastaavia sumeita joukkoja. Edellisten avulla voimme tulkita ja ymmärtää sääntöjä, ja jälkimmäisten avulla voimme puolestaan suorittaa erilaisia laskutoimituksia ja operaatioita tietokoneella. Kuvan 2.3 tapauksessa olemme numeeriselta kannalta määrittäneet säännöt:

1. Jos henkilön ikä on noin 18 vuotta, hänen asuntonsa pinta-ala on noin 25 m^2.
2. Jos henkilön ikä on noin 45 vuotta, hänen asuntonsa pinta-ala on noin 96 m^2.
3. Jos henkilön ikä on noin 65-100 vuotta, hänen asuntonsa pinta-ala on noin 49 m^2.

Niinpä *nuori* tarkoittaa tässä yhteydessä noin 18-vuotiasta, *keski-ikäinen* noin 45-vuotiasta, *vanha tai hyvin vanha* noin 65-100 –vuotiasta, *pieni asunto* noin 25 m^2 asuntoa jne. Olemme siis tietyssä mielessä operationalisoineet alkuperäiset sääntömme eli muuttaneet ne laskettavaan muotoon. Voimme tietysti käyttää myös pelkästään näitä numeerisempia sääntöjä, jos ne ovat jo sinänsä riittävän ymmärrettäviä.

Edellä valittujen muuttujien arvojen perusteella voimme muodostaa itse asiassa 3·5 = 15 erilaista sääntöä, mutta sumeita sääntöjä käytettäessä emme tarvitse niitä kaikkia, vaan valitsemme niistä vain muutaman tyypillisen säännön. Tämä periaate on hyvin keskeinen sumeissa malleissa, ja sen vuoksi näissä malleissa tarvitaan vain pienet sääntöjoukot. Pienten sääntöjoukkojen ansiosta taas mallit ovat yksinkertaisia ja laskennallisestikin keveitä. Niinpä perinteisten sääntöpohjaisten mallien tuhansiakin sääntöjä käsittävien kokoelmien sijasta voidaan käyttää muutamien tai yleensä korkeintaan muutamien kymmenien sääntöjen malleja.



Kuva 2.3. Kielelliset sumeat säännöt sumeiden joukkojen avulla esitettyinä. Etujäsenet vasemmalla. Muuttujina henkilön ikä ja asunnon koko. Syöte ikä=37 tuottaa vasteen, että asunnon koko=88,2 (vrt. kuva 2.1).

Kuvassa 2.3 on myös esitetty hieman sumean päättelyn käyttöä sääntöjen kanssa. Nämä säännöthän siis yleensä kertovat meille muutamia tyypillisiä asioita tarkastelun kohteena olevasta ilmiöstä. Näiden muutaman säännön avulla on kuitenkin pystyttävä mallintamaan ilmiö kaikissa tilanteissa eli koko syötemuuttujan vaihteluvälillä, ja tähän sumeat mallit käyttävät interpolointia (interpolation). Jos me vaikkapa haluamme edellä olevien sääntöjen perusteella tietää, minkä kokoisessa asunnossa 37-vuotias henkilö sitten asuu, päättelemme, että

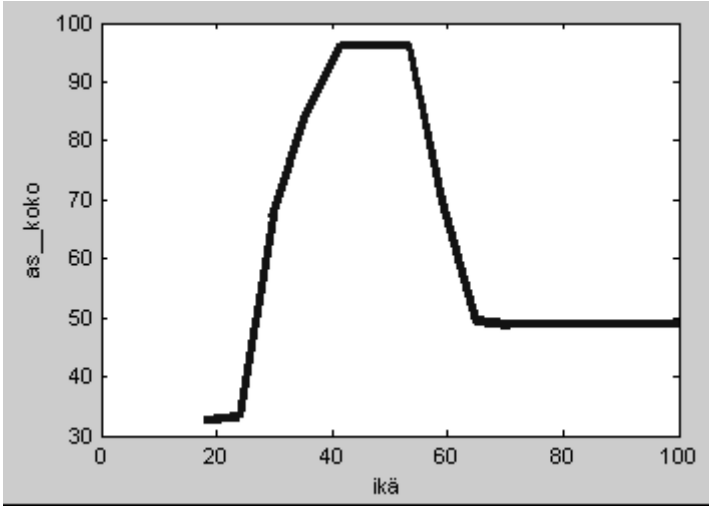
- tämä henkilö on lähinnä keski-ikäinen, mutta hän voi samalla olla myös jollakin pienellä jäsenyysasteella nuori, joten päättelyssämme on suurin painoarvo toisella säännöllä, mutta samalla myös pieni painoarvo ensimmäisellä säännöllä. Näin sen vuoksi, että arvo 37 vuotta on lähimpänä näiden sääntöjen etujäsenien arvoja, ja toisen säännön etujäsentä selvästi lähimpänä (eli tähän joukkoon 37-vuotiaalla on korkein jäsenyysaste). Kolmas sääntö näyttää taas olevan merkityksetön tässä tilanteessa.
- Voimme tämän jälkeen ensimmäisen ja toisen säännön takajäsenien perusteella päätellä, että 37-vuotias henkilö asuu pienessä tai melko suuressa asunnossa, ja näistä kahdesta jälkimmäisellä arvolla on selvästi suurempi paino.
- Lopputulos on siis arvojen pieni ja melko suuri välissä, joskin selvästi lähempänä arvoa melko suuri. Niinpä lopputulos on tietyssä mielessä näiden kahden takajäsenen painotettu keskiarvo eli olemme soveltaneet interpolointia. Näin sumea päättely malleissamme yleensä toimii, ja varsinaisia laskutoimituksia kommentoidaan tarkemmin myöhemmin. Ihmisen päättelyhän muuten toimii samalla tavalla.

Edellä kuvatun ajattelutavan ansiosta meidän ei siis tarvitse muodostaa kovin montaa sääntöä, koska me käytämme vastearvojen laskennassa edellä mainittua interpolointia. Näin sääntöjen määrä on selvästi pienempi kuin perinteisissä sääntöpohjaisissa järjestelmissä, joissa säännöt on tuotettava kaikkien syötemuuttujan tapausten osalta.

Kuvassa 2.3 on edellä esitetty päättelytapa nähtävillä sumeiden joukkojen tasolla, ja syötearvon *37 vuotta* paino on merkitty sumeissa joukoissa tummalla alueella (mitä korkeampi tumma alue, sitä suurempi paino). Takajäseniä on sitten tässä tapauksessa painotettu suoraan samalla tavalla. Syötearvomme tapauksessa mallimme päättely tuottaa takajäsenien painotettuna keskiarvona vastearvon 88,2 m^2 (tarkempi laskutapa kerrotaan siis myöhemmin).

Kuvassa 2.4 on laskettu mallimme tuottavat vastearvot koko syötemuuttujan vaihteluvälillä. Saamme ei-lineaarisen mallin helposti kolmen säännön avulla. Jos kyseiset vastearvot eivät vastaa odotuksiamme, voimme parantaa malliamme esimerkiksi muuttamalla sumeiden joukkojen muotoa ja sijaintia sekä lisäämällä sääntöjä. Valitettavasti tämä voi olla toisinaan ilman empiiristä

aineistoa työlästä, ja sen vuoksi havaintoaineiston kanssa onkin yleensä helpompi rakentaa malleja.



Kuva 2.4. Sumean mallin tuottamat vastearvot kolmen säännön perusteella kun muuttujina ikä ja asunnon koko.

2.1.2. Kaksi syötemuuttujaa ja yksi vastemuuttuja

Toisena pelkästään asiantuntemukseen perustuvana esimerkkinä tarkastellaan opiskelijan loppuarvosanan määrittämistä kahden tentin perusteella. Syötemuuttujina ovat näiden tenttien pistemäärät (0-100) ja vastemuuttujana loppuarvosana (0-5).

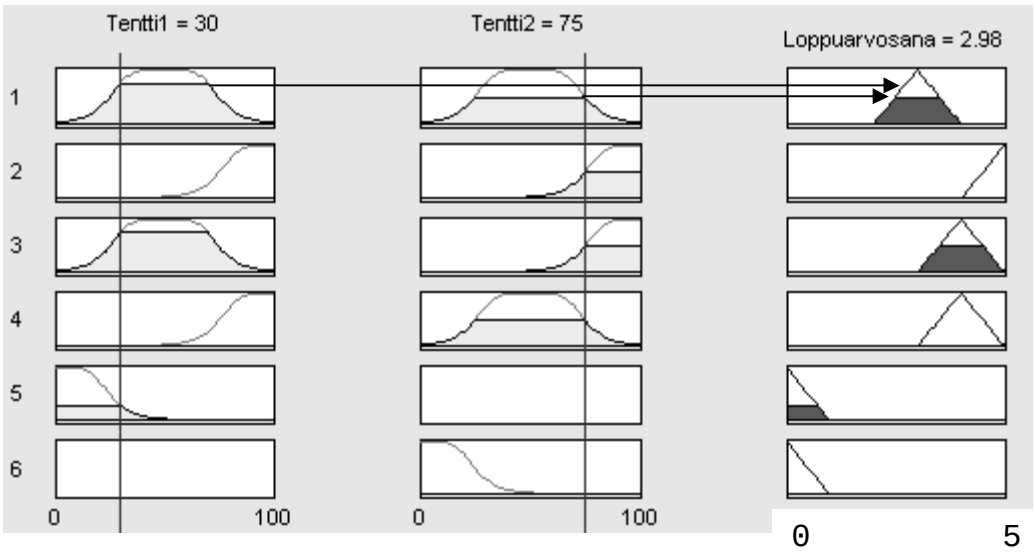
Edellä olevan perusteella meidän tarvitsee määrittää vain muutamia tyypillisiä esimerkkejä sumeiksi säännöiksi, ja näiden perusteella saamme sitten laskettua kaikki vastearvot. Käytämme vaikkapa seuraavia päättelysääntöjä:

1. Jos ensimmäisen tentin pistemäärä on noin 50 ja toisen tentin noin 50, niin loppuarvosana on noin 3.
2. Jos ensimmäisen tentin pistemäärä on noin 100 ja toisen tentin noin 50, niin loppuarvosana on noin 4.
3. Jos ensimmäisen tentin pistemäärä on noin 50 ja toisen tentin noin 100, niin loppuarvosana on noin 4.

4. Jos toisen tentin pistemäärä on noin 100 ja ensimmäisen tentin noin 50, niin loppuarvosana on noin 4.
5. Jos ensimmäisen tentin pistemäärä on noin 0 ja toisen mitä tahansa, niin loppuarvosana on noin 0.
6. Jos ensimmäisen tentin pistemäärä on mitä tahansa ja toisen tentin noin 0, niin loppuarvosana on noin 0.

Koska syötemuuttujia on nyt kaksi, sääntöjen etujäsenissä on myös kaksi muuttujan arvoa, ja ne on yhdistetty ja-sanalla (vaihtoehtoisesti voidaan toisissa tilanteissa käyttää tai-sanaa). Ja-sanaa käytetään eniten, koska sääntöjen ymmärtäminen on silloin helppoa. Kuvassa 2.5 nämä säännöt on esitetty graafisesti, ja niissä on käytetty sekä normaalijakauman että kolmionmuotoisia joukkoja.

Kun etujäsenessä on useampia kuin yksi muuttuja, kutakin niistä painotetaan ensin sen mukaisesti, kuinka lähellä ne ovat annettua syötearvoa (kuten luvussa 2.1.1). Tämän jälkeen lasketaan säännön koko etujäsenen paino sen komponenttien painojen perusteella. Ja-sanan tapauksessa paino on usein komponenttien painojen minimi, ja tai-sanan osalta komponenttien maksimi. Säännön etujäsenen paino on sitten aikaisemmin esitetyn mukaisesti myös suoraan takajäsenen paino, ja mallin vastearvo on takajäsenien painotettu yhdistelmä (josta myöhemmin tarkemmin).

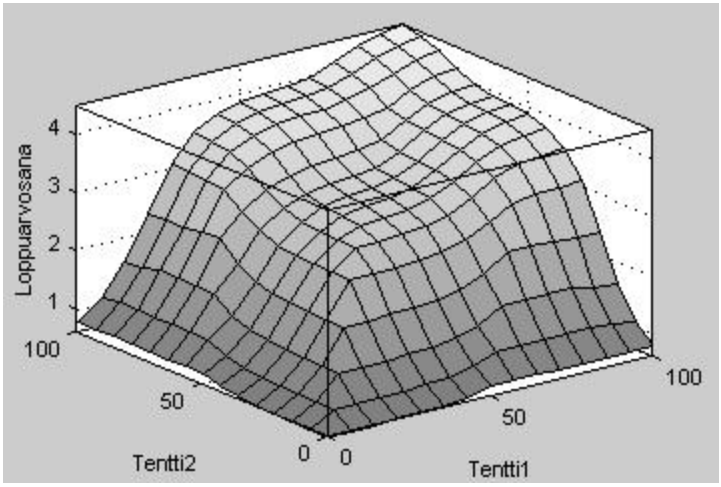


Kuva 2.5. Sumeat säännöt tenttien loppuarvosanan määrittämiseksi. Etujäsenissä kaksi muuttu-

jaa, jotka tässä yhdistetty ja-sanalla. Niinpä takajäsenien painot (tummien alueiden korkeudet oik.) määräytyvät vastaavien etujäsenien painojen minimin perusteella.

Kuvassa 2.5 on esitetty tilanne, jossa tenttien pistemäärät ovat 30 ja 75 eli syötevektorina on (30,75). Silloin säännön paino määräytyy siis aina pienimmän etujäsenen painon mukaan (tummennetut alueet etujäsenissä), jotka suoraan ovat vastaavien takajäsenien painoja. Sääntöjen 5 ja 6 tapauksessa vain yksi etujäsenen komponentti on mukana painon laskennassa, koska *mitä tahansa* käytännössä tarkoittaa, että niillä ei ole päätöksenteossa merkitystä. Niinpä säännöt 2,4,6 eivät ole tässä tapauksessa mukana laskennassa (painot nollia) ja säännöillä 1 sekä 3 on suurimmat painot. Sumea päättely tuottaa sitten vastearvoksi 2,98, ja tämä voidaan tarvittaessa pyöristää vaikkapa kokonaisluvuksi.

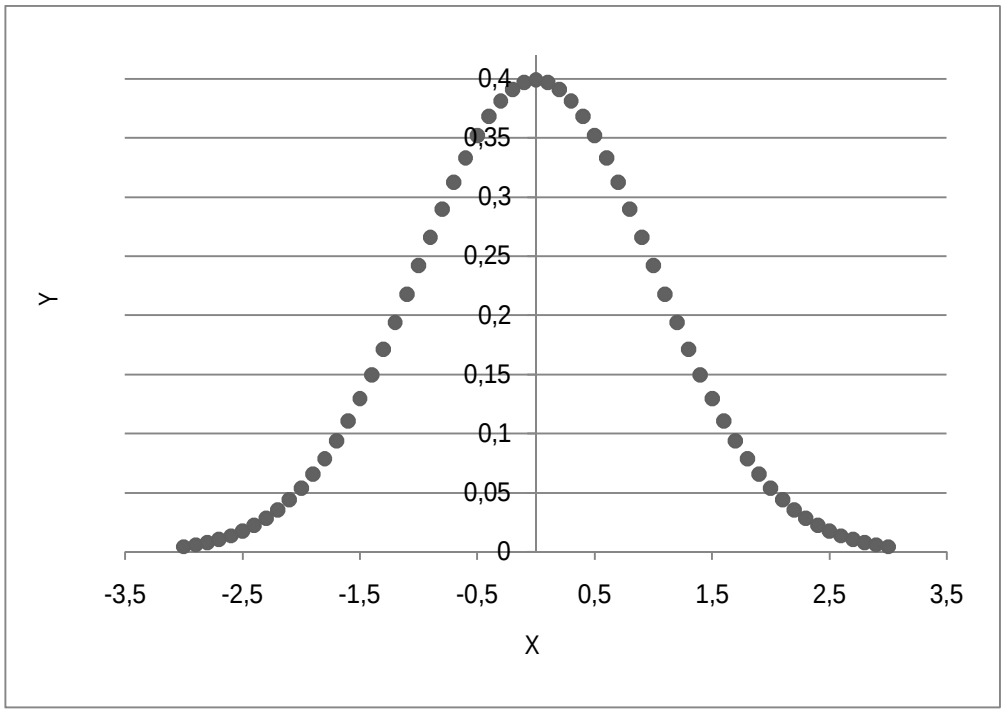
Kuvassa 2.6 on esitetty ei-lineaarisen mallimme kaikki vastearvot. Jos vasteet eivät tyydytä meitä, voimme virittää malliamme tarpeen mukaan edellä kuvatuilla tavoilla. Perinteisellä tavalla voimme esittää vastaavan mallin vaikkapa kahden muuttujan matemaattisen funktion avulla.



Kuva 2.6. Sumean mallin tuottamat vastearvot tentin loppuarvosanan määrittämiseksi.

2.2. Kun havaintoaineistoa on käytössä

Kun meillä on käytössämme (empiiristä) aineistoa, malleja on usein helpompi rakentaa. Silloin säännöt voidaan yleensä muodostaa havaintopisteiden ja ryhmittelyanalyysin avulla. Käytämme ensin kuvassa 2.7 esitettyä aineistoa, joka on tilastotiedettä opiskelleille varmasti tutun näköinen, koska kyse on standardoidun normaalijakauman tiheysfunktiosta.



Kuva 2.7. Mallinnusaineisto, jossa yksi syöte- ja vastemuuttuja.

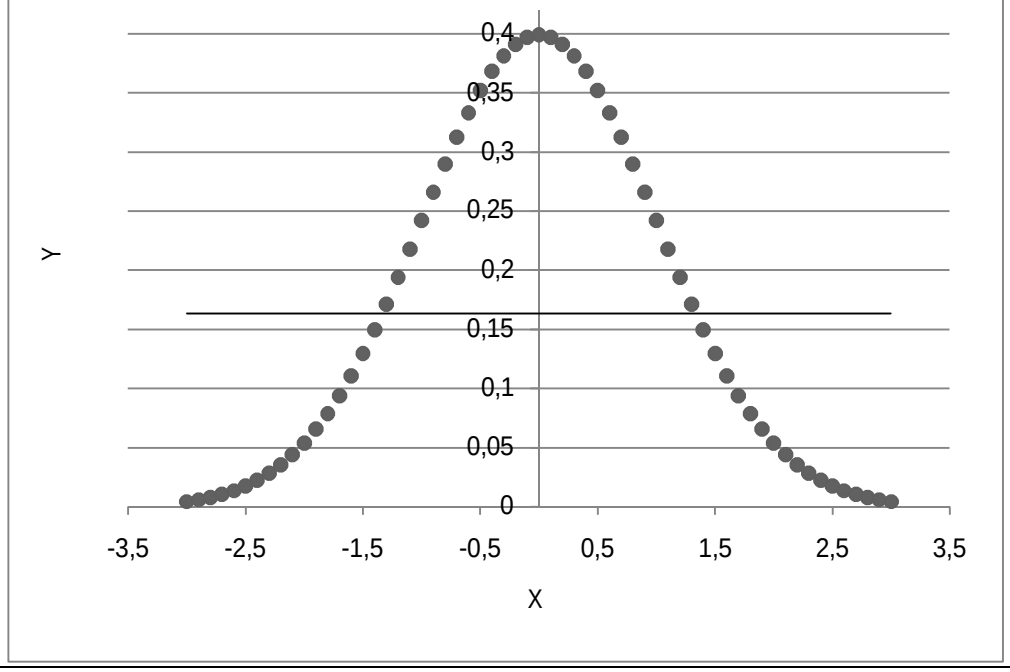
Kyseessä on siis malli, jossa on yksi syöte- ja vastemuuttuja (muuttujat X ja Y). Jos käytämme perinteistä matemaattista mallinnusta, SPSS (Analyze – Regression – Curve estimation) tai jopa MS Excel (trendikaaret) tuottavat jo ihan hyviä sovitteita (fitting).

Koetamme ensin löytää mahdollisimman yksinkertaisen mallin, mikä yleensä tarkoittaa lineaarista mallia. Niinpä määritämme sellaisen lineaarisen funktion, että vastaava sovite eli suora on mahdollisimman lähellä aineistomme pisteitä. Kuvassa 2.8 on Excelillä tuotettu sovite, ja kuten huomataan, lopputulos ei ole hyvä, sillä mallin (so. funktion) avulla laskettu sovite eli regres-

siosuora (regression line) poikkeaa paljon vastemuuttujan Y todellisista arvoista. Kyseinen mallimme on matemaattinen yhden muuttujan funktio

$$Y = 0{,}00 \cdot X + 0.16$$

ja siitäkin asiaan perehtynyt toteaa suoraan, että kyseessä on huono malli, koska X:n kerroin on nolla.



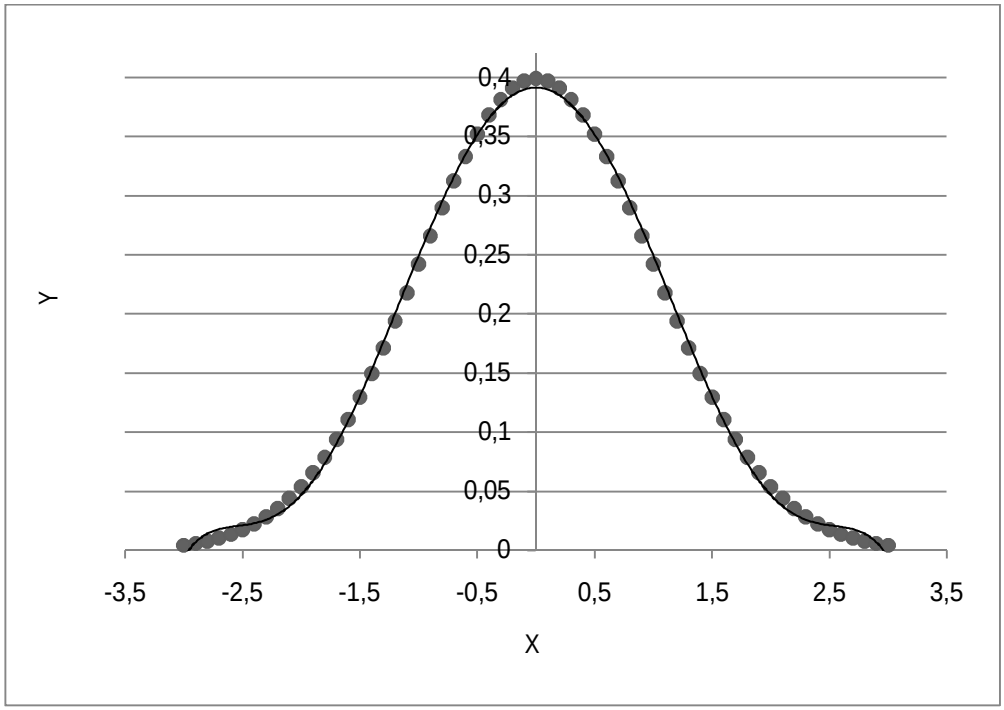
Kuva 2.8. Lineaarisen mallin sovite normaalijakaumalle.

Pyrimme seuraavaksi löytämään paremman mallin etsimällä sopivan ei-lineaarisen (non-linear) matemaattisen funktion, ja käytännössähän me silloin pyrimme löytämään sellaisen sovitteen, että se on mahdollisimman lähellä muuttujan Y arvoja. Ei-lineaarisessa mallissa tämä kuvaaja voi olla mikä tahansa käyrä, joten meillä on tarjolla paljon erilaisia vaihtoehtoja: se voi olla vaikkapa polynomi, eksponenttifunktio, logaritmifunktio, trigonometrinen funktio, potenssifunktio tai näiden yhdistelmä, ja meillä ei yleensä ole kunnollisia menetelmiä sopivan funktiotyypin löytämiseksi. On siis sovellettava pääasiassa yritystä ja erehdystä, ja tästä ne ei-lineaarisen mallinnuksen vaikeudet alkavat. Jos

sitten sopiva funktio löytyykin, sen ymmärtäminen voi olla vaikeaa, koska kyseiset funktiot voivat olla monimutkaisia. Toisaalta reaalimaailman ilmiöt näyttävät yleensä olevan luonteeltaan ei-lineaarisia.

Kuvassa 2.9 on yksi hyvältä vaikuttava ei-lineaarinen sovite, ja se perustuu Excelillä tuotettuun kuudennen asteen polynomiin. Ongelmana kuitenkin on, kuinka moni lukijoista pystyisi kyseisen funktion avulla tulkitsemaan tai selittämään, millainen yhteys on muuttujien X ja Y välillä (säästän lukijan turhilta kärsimyksiltä ja jätän tämän funktion esittämättä).

Toisin kuin tavallisesti, me tässä tapauksessa tiedämme pistejoukon muodostaneen oikean funktion, joka on eksponenttifunktio. Yrityksen ja erehdyksen avulla tämän funktion määrittäminen olisi kuitenkin työlästä.



Kuva 2.9. Kuudennen asteen polynomiin perustuva ei-lineaarinen sovite normaalijakaumalle.

Edellä on pelkistetyssä muodossa tuotu esille tilastollis- matemaattisen mallinnuksen perusongelmat:

- Lineaarisia malleja on suosittu, koska ne ovat matemaattisesti ja laskennallisesti melko yksinkertaisia, mutta niiden käyttökelpoisuus voi olla huono ja ymmärrettävyyskin ongelmallista.
- Ei-lineaariset mallit taas ovat käyttökelpoisempia, mutta sopivien mallien (funktioiden) löytäminen ja niiden ymmärtäminen voivat olla vaikeaa. Ne ovat usein myös laskennallisesti raskaita.

Oppivat järjestelmät (neuroverkot) ja evoluutiolaskenta (evolutionary computing) ovat uusi tapa löytää hyviä matemaattisesti esitettyjä ei-lineaarisia malleja, ja ne ovat luonteeltaan myös ei-parametrisiä menetelmiä, joten meidän ei tarvitse tehdä jakaumaoletuksia. Kuitenkin erityisesti adaptiivisten systeemien ymmärtäminen ja tulkinta ovat niissä esiintyvien lukuisten painokertoimien vuoksi myös vaikeaa, usein vaikeampaa kuin perinteisten matemaattisten mallien tapauksessa.

Sumeita systeemejä soveltavat mallit voivat tarjota ratkaisuja edellä kuvattuihin ongelmiin. Koska meillä on nyt aineisto käytettävissä, rakennetaan malli sen perusteella käyttämällä Matlab'in *anfisedit*-komentoa, joka perustuu Jangin neuro-sumeaan ANFIS-algoritmiin [32]. *Anfisedit* käynnistää meille graafisen mallinnusympäristön. Etenemme seuraavasti:

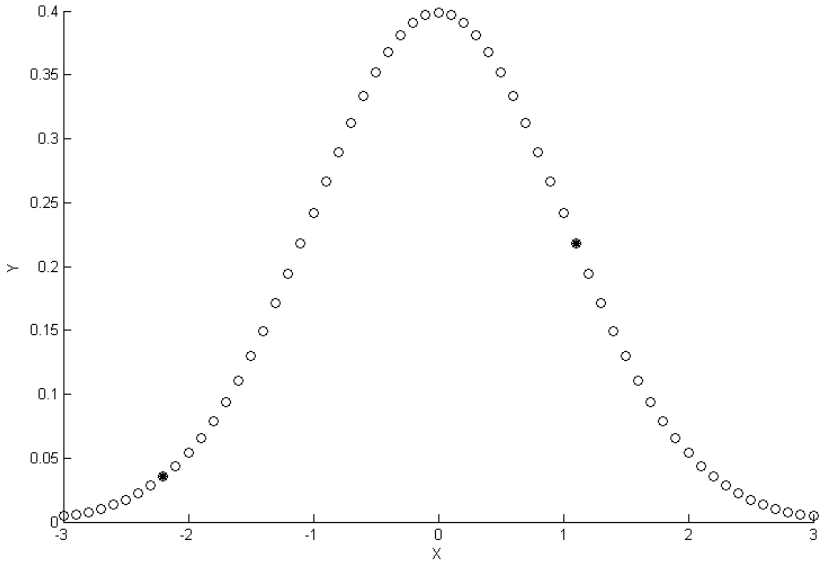
1. Etsimme ensin aineistostamme, joka käsittää sekä syöte- että vastemuuttujat, sumean ryhmittelyanalyysin avulla sopivia ryhmiä, ja sitten näiden ryhmien keskusten avulla muodostamme alustavat, muuttujien välisiä yhteyksiä kuvaavat sumeat säännöt.
2. Nämä säännöt ovat mallimme perusta, ja voimme niitä ja sumeaa päättelyä käyttäen laskea sitten samalla tavalla kuin luvussa 2.1 mallimme sovitteen arvoja.
3. Malliamme voidaan tarvittaessa parantaa adaptiivisten systeemien tai geneettisten algoritmien avulla. Edellisessä tapauksessa puhumme neuro-sumeista (neuro-fuzzy) ja jälkimmäisessä geneettis-sumeista malleista (genetic-fuzzy).
4. Pyrimme määrittämään sellaisen sovitteen, joka on mahdollisimman lähellä aineistomme vastemuuttujien todellisia pisteitä, mutta toisaalta pyrimme myös yksinkertaiseen malliin eli pieneen sääntöjen määrään ja jopa yleistyskelpoisuuteen. Niinpä me käytännössä kokeilemme erimuotoisilla jäse-

nyysfunktioilla ja erikokoisilla sääntöjoukoilla, ja tämä käy helposti esimerkiksi Matlab-ympäristössä.

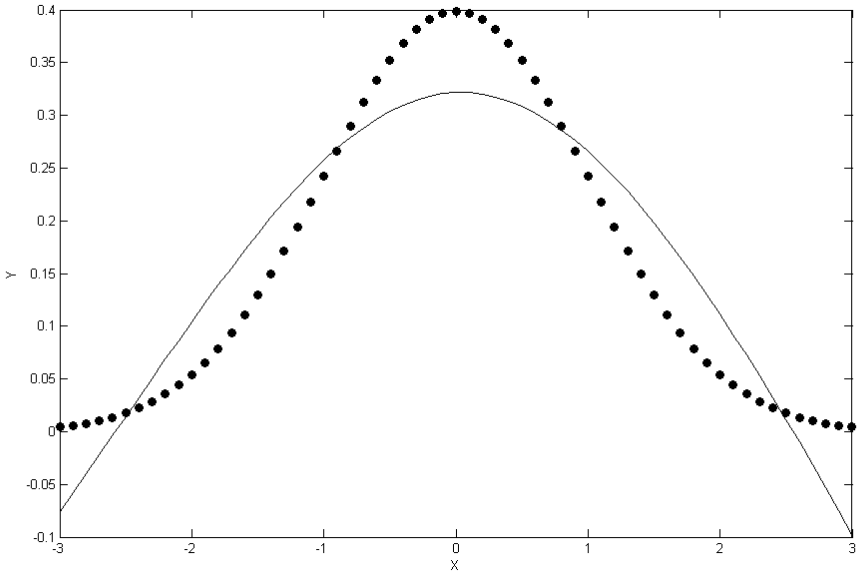
Tarkastelemme ensin kaksimuuttujaisen normaalijakauma-aineistomme osalta kahden säännön mallia, jossa on siis yksi syöte- ja vastemuuttuja. Jos nyt päätämme käyttää kahta sumeaa sääntöä, mallimme perustuu kahteen aineistostamme määritettyyn sumean ryhmän keskukseen. Koska Matlab'in graafinen *anfisedit*-mallinnusympäristö sallii vain *subclust*-ryhmittelyalgoritmin käytön, voimme määrittää ryhmien lukumäärän vain epäsuorasti vaikuttavuussäteen avulla (säde saa arvoja välillä 0 – 1, ja mitä lähempänä säteen arvo on ykköstä, sitä vähemmän ryhmiä määritetään). Käyttämällä säteen arvoa 0,99 saamme kaksi ryhmää, joiden keskukset ovat kuvassa 2.10, ja nämä keskukset tulkitaan sitten sumeiksi säännöiksi seuraavasti:

1. Jos X on noin -2,2, niin Y on noin 0,04.
2. Jos X on noin 1,1, niin Y on noin 0,22.

Käytämme siis säännöissämme suoraan ryhmien keskuksia. Nämä säännöt esittävät pelkistetysti aineistomme osalta, millainen on syöte- ja vastemuuttujan välinen yhteys, ja niiden avulla voidaan myöhemmin kuvatulla tavalla laskea helposti kaikki sovitteemme arvot syötemuuttujan X arvojen vaihteluväliltä. Kuvassa 2.11 on vastaava mallimme tuottama sovite, ja toteamme, että siinä on toivomisen varaa, joten rakennamme paremman mallin määrittämällä useampia sumeita ryhmiä (eli sääntöjä) aineistostamme.



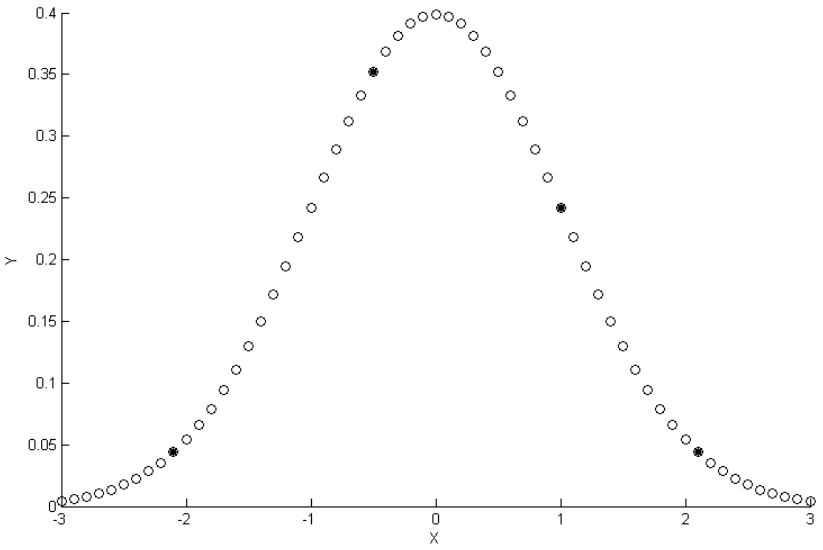
Kuva 2.10. Sumea ryhmittelyanalyysi normaalijakaumalle, ryhmien keskukset kahden ryhmän ratkaisussa.



Kuva 2.11. Kahteen sumeaan sääntöön perustuva sovite normaalijakaumalle.

Kun pienen kokeilun jälkeen löydämme sopivaksi vaikuttavuussäteeksi 0,55, saamme kuvan 2.12 mukaiset neljä ryhmän keskusta, jotka sitten tulkitaan sumeiksi säännöiksi

1. Jos X on noin -2,1, niin Y on noin 0,04.
2. Jos X on noin -0,5, niin Y on noin 0,35.
3. Jos X on noin 1,0, niin Y on noin 0,24.
4. Jos X on noin 2,1, niin Y on noin 0,04.



Kuva 2.12. Sumea ryhmittelyanalyysi normaalijakaumalle, ryhmien keskukset neljän ryhmän ratkaisussa.

Neljään sääntöön perustuva mallin sovite on kuvassa 2.13, ja tulos vaikuttaa hyvältä, jopa paremmalta kuin edellä mainitun matemaattisen polynomimallimme tapauksessa. Adaptiivisen systeemin avulla voimme vielä hieman tarvittaessa parantaa malliamme, jolloin käytännössä muokkaamme sumeiden joukkojen muotoa ja muutamme niiden sijaintia sopivasti. Samoin mallia voidaan parantaa sääntöjä lisäämällä.

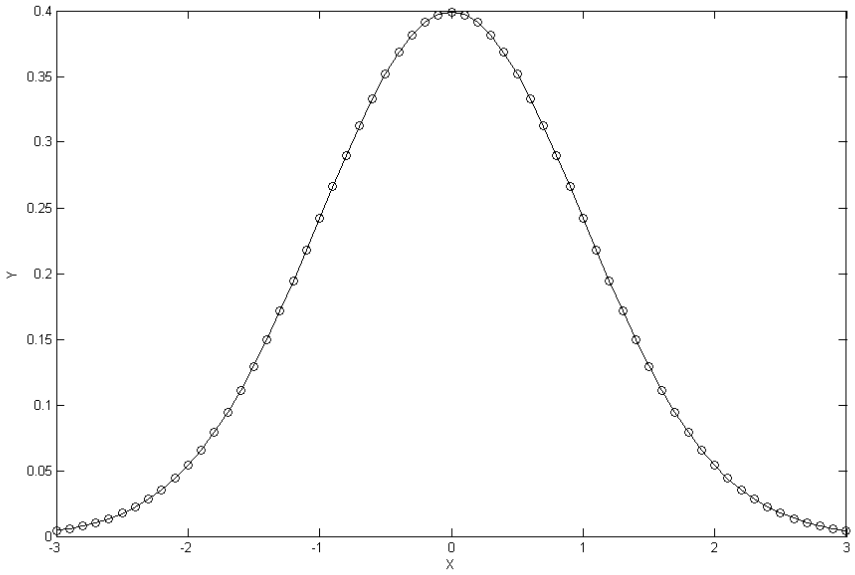
Mallin hyvyyttä voidaan arvioida vaikkapa laskemalla sovitteen etäisyys todellisista pisteistä eli yleensä näiden virhetermien neliöiden summa:

$$\text{kokonaisvirhe} = \Sigma_i(t_i - m_i)^2,\ (i=1,2,\ldots,n)$$

kun t_i ja m_i ovat vastaavasti todelliset ja mallin tuottamat arvot ja n on havaintojen lukumäärä. Mitä pienempi kokonaisvirhe, sen parempi malli. Jos tämä kokonaisvirhe (SS residuals) jaetaan havaintojen lukumäärällä ja sitten lasketaan osamäärän neliöjuuri, saamme virheiden kvadraattisen keskiarvon (root mean square of errors, *rmse*) eli

$$rmse = \sqrt{\text{kokonaisvirhe}/n}$$

Me käytämme jatkossa erityisesti jälkimmäistä tunnuslukua mallien hyvyyden arviointiin. Seuraavassa luvussa kerrotaan vielä tarkemmin, kuinka sumeat säännöt muodostetaan aineistosta ryhmittelyanalyysin avulla.



Kuva 2.13. Neljään sumeaan sääntöön perustuva sovite normaalijakaumalle.

2.3. Täsmällisempi kuvaus sumeiden sääntöjen muodostamiseksi

Jos meillä on käytössämme tutkimusaineisto, sumeat säännöt voidaan määrittää ryväs- ja ruudukkomenetelmällä. Ryväsmenetelmän avulla, jota käsittelemme ensin, etsimme sumean ryhmittelyanalyysin perusteella sopivat ryhmät annetussa muuttuja-avaruudessa, ja sitten mallin tuottamia vastearvoja voidaan puolestaan laskea näiden sääntöjen ja valitun sumean päättelyalgoritmin avulla.

Tarkastellaan kuvan 2.14 tilannetta, jossa on valittu neljä ryhmää edustamaan aineistoamme:

1. Tällöin vastaavat sumeat joukot voivat olla kuvan 2.14 tyyppisiä, vaikkapa kolmion muotoisia joukkoja.
2. Nämä sumeat joukot on muodostettu neljästä ryhmästämme siten, että sekä X- että Y-akselilla joukkojen lukumäärä määräytyy ryhmien keskuksien X- ja Y-koordinaattien perusteella. Huomattakoon, että saadut sumeat joukot eivät yleensä sijaitse akseleillaan tasavälisesti, koska ryhmittelyanalyysi määrittää niiden paikat.
3. X-akselilla on ryhmien keskuksilla neljä eri arvoa (-2,1, -0,5, 1 ja 2,1) ja Y-akselilla kolme (0,04, 0,25 ja 0,3). Näissä pisteissä, siis ryhmien keskuksissa, vastaavat sumeat joukot saavat maksimiarvonsa (tavallisesti 1).
4. Muiden havaintopisteiden jäsenyysasteiden arvot määräytyvät sen mukaan, kuinka etäällä ne ovat koordinaattiakselien suunnassa ryhmien keskuksista. Siis mitä lähempänä jokin (X,Y)-piste on jotakin ryhmän keskusta, sitä suurempi on sen jäsenyysaste tähän ryhmään. Niinpä esimerkiksi pisteellä (2, 0,05) on korkea jäsenyysaste sekä vaaka-akselin suunnassa oikeassa reunassa että pystyakselin suunnassa alimmassa sumeassa joukossa, koska sen X- ja Y-koordinaatit ovat vastaavasti lähellä näitä joukkoja vaaka- ja pystysuunnassa. Muihin joukkoihin tämä piste kuuluu pienemmillä jäsenyysasteilla. Jäsenyysfunktioiden muoto määräytyy siis jäsenyysasteiden perusteella, ja nämä vuorostaan havaintopisteiden muodostamien ryhmien mukaan. Matemaattisestihan näiden sumeiden joukkojen muodostamisessa on kyse ryhmien määrittämistä projektioista (”varjoista”) koordinaattiakseleilla.

5. Sumeat säännöt vuorostaan muodostetaan yhdistämällä kunkin ryhmän vaaka- ja pystyakselin joukot (merkitty kuvassa 2.14 nuolilla). Säännöt ovat näin ollen muotoa

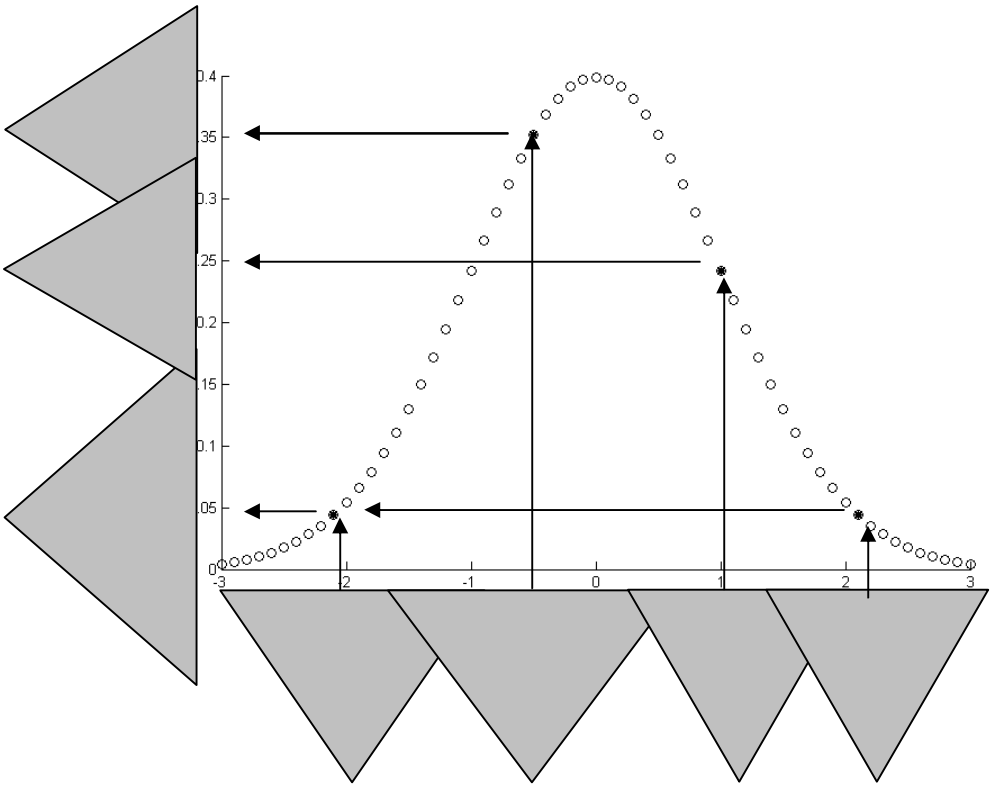
   Jos X on A, niin Y on B,

   kun A on vaaka- ja B vastaava pystyakselin sumea joukko. Nämä joukot sitten nimetään tarvittaessa sopivasti niiden muodon ja sijainnin perusteella. Niinpä esimerkiksi vasemman alanurkan joukkojen tapauksessa vaikuttaa mielekkäältä käyttää kielellis-matemaattista sääntöä

   Jos X on noin -2.1, niin Y on noin 0.04.

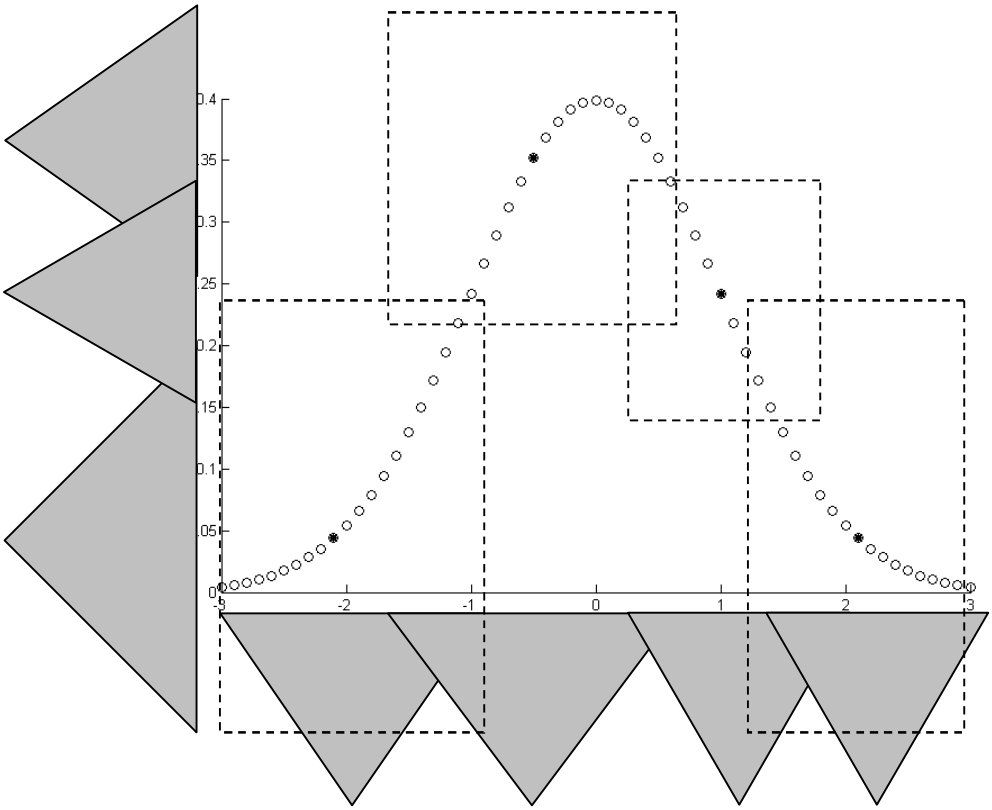
   Tilanne on analoginen perinteisen yhden muuttujan funktion y = f(x) kanssa:

   jos x = -2.1, niin y = 0,04.



Kuva 2.14. Sumeiden sääntöjen muodostaminen aineiston ryhmien perusteella.

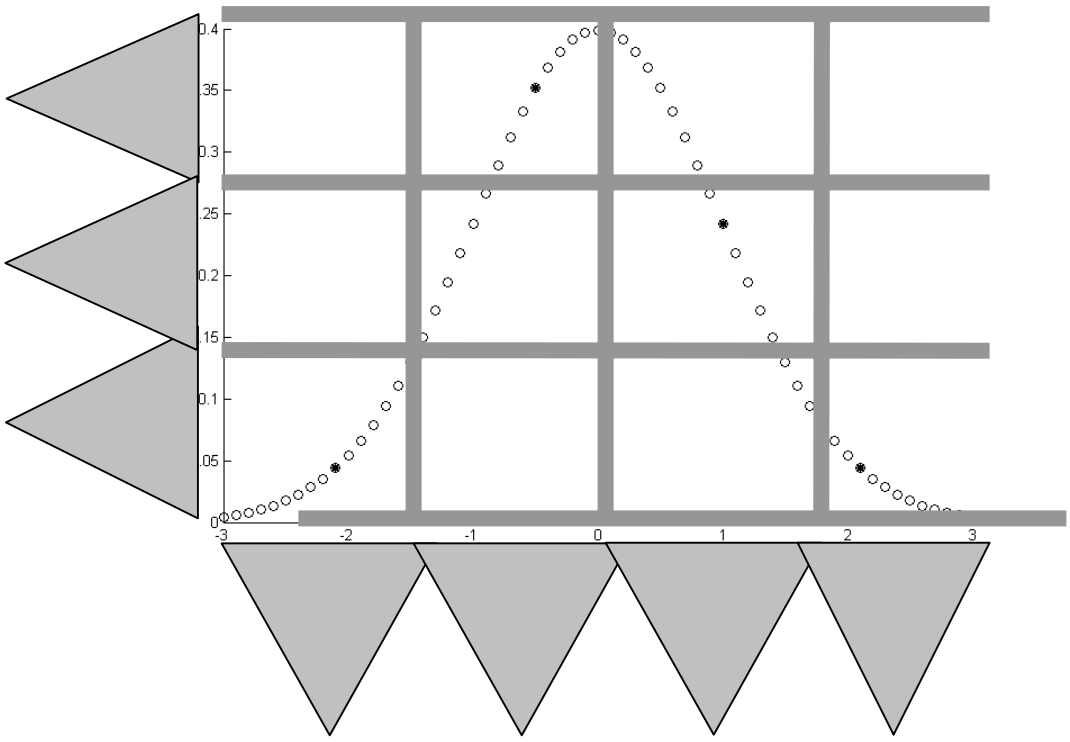
Kuvassa 2.15 on luonnosteltu vastaavat sumeat säännöt suorakulmioina, ja näiden sääntöjen tulisi olla lomittain, jotta sumeaa päättelyä voidaan täysipainoisesti hyödyntää.



Kuva 2.15. Sumeiden sääntöjen muodostaminen aineiston ryhmien perusteella.

Ruudukkomenetelmän (grid) tapauksessa jaamme muuttuja-akselit yhtä pitkiin väleihin, ja näin muodostuneet ruudut ovat sääntöjämme (aineiston ryhmittelyä ei siis tarvita). Tämä menetelmä on yleensä laskennallisesti raskaampi kuin ryväsmenetelmä, sillä se tuottaa enemmän sääntöjä. Jos muuttujia ja välejä on paljon, saamme todella suuria sääntömääriä. Niinpä esimerkiksi kolme muuttujaa, joiden vaihteluvälit on jaettu viiteen osaan, tuottavat jo $5^3 = 125$ mahdollista sääntöä. Toisaalta tämä menetelmä voi olla havainnollinen vaikkapa opetustilanteessa.

Normaalijakauma-aineistomme tapauksessa 4+3 väliä tuottavat siis 12 mahdollista sääntöä (eli ruutua) kuten kuvassa 2.16 on esitetty, ja näistä kuusi sääntöä (ei-tyhjät ruudut) tarvitaan mallin rakennukseen.



Kuva 2.16. Sumeiden sääntöjen muodostaminen aineistosta ruudukkomenetelmällä.

Me siis kuvailemme tälläkin menetelmällä taas aineistomme muutaman likimääräisen säännön avulla. Seuraavassa luvussa tarkastellaan lähemmin sumeaan päättelyyn perustuvaa laskentaa.

2.4. Mallin vastearvojen laskenta sumean päättelyn avulla

Sumeassa päättelyssä vastearvojen laskenta perustuu interpolointiin, ja voimme tässä yhteydessä soveltaa erilaisia päättelytapoja. Tunnetuimmat päättelymenetelmät ovat Mamdani- (tai Mamdani-Assilian) ja Takagi-Sugeno(-Kang) –päättely. Edellistä käytetään kun tutkimusaineistoa ei ole saatavilla ja jälkimmäistä etupäässä aineiston kanssa.

Tarkastellaan aluksi Mamdani-päättelyä kuvan 2.17 perusteella (joka on muuten miltei samanlainen kuin kuva 2.5) [9]:

1. Syötteiden ja sääntöjen etujäsenien perusteella lasketaan painot sääntöjen takajäseniä varten. Mitä lähempänä syöte on etujäsenen sumean joukon keskusta, sitä suurempi on painoarvo eli ”laukaisuvoimakkuus” (weight, firing strength). Koska painot käytännössä lasketaan vastaavien jäsenyysfunktioiden perusteella, ne saavat arvoja nollasta ykköseen.
   - Jos syötearvomme ovat vektorin (30,75) mukaiset, esimerkiksi kuvan 2.17 ensimmäisen säännön tapauksessa tentin 1 syötearvo (= 30) on melko lähellä sumean joukon *noin 50* keskusta, joten saamme melko korkean painoarvon (itse asiassa se on jäsenyysasteen arvo pisteessä syöte = 30, eli paino = 0,75).
   - Tentin 2 syötearvo = 75 taas tuottaa ensimmäisen säännön tapauksessa jäsenyysasteen arvon eli painon, joka on 0,5.
   - Samaan tapaan lasketaan muutkin painot jokaisen säännön etujäsenen joukkojen osalta kun syötevektorina on (30,75).
2. Sääntöjen etujäsenien painojen perusteella määritetään vastaavien takajäsenien painot. Jos etujäsenessä useampia kuin yksi sumea joukko, laskemme lopullisen painon yleensä seuraavasti:

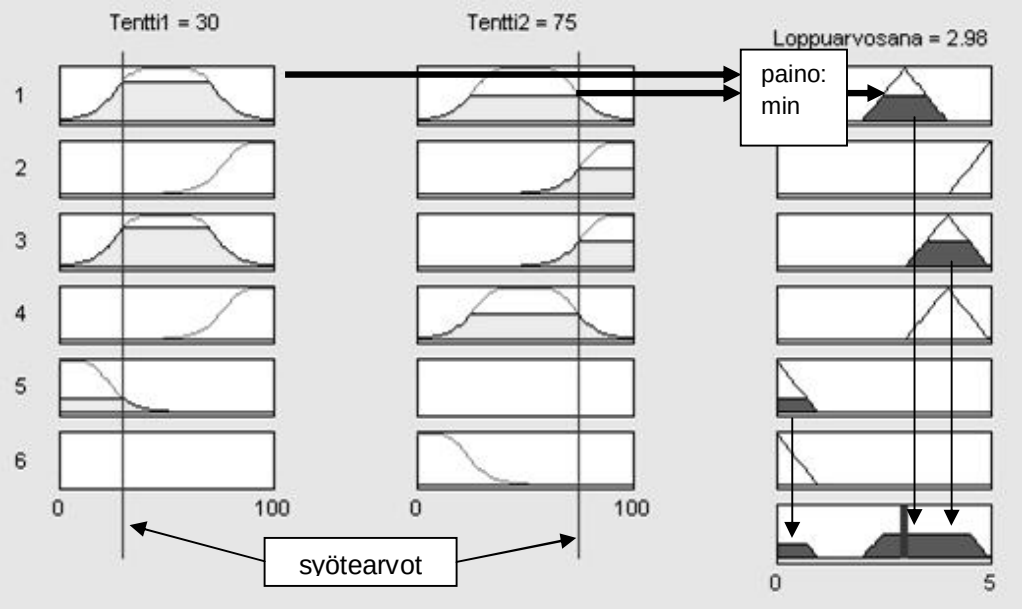
   - Etujäsen ja-tyyppiä eli
     jos X1 on A1 ***ja*** *X2 on A2* ***ja*** *…* ***ja*** *Xn on An*:
     painojen minimi tai tulo.
   - Etujäsen tai-tyyppiä eli
     jos X1 on A1 ***tai*** *X2 on A2* ***tai*** *…* ***tai*** *Xn on An*:
     painojen maksimi tai algebrallinen summa.
   - Siis ja- sekä tai-sanan käytön osalta on omat laskusääntönsä.
   - Kuvan 2.17 tapauksessa on käytetty minimi-sääntöä.

3. Laskettu paino määrittää säännön vastaavan takajäsenen tärkeyden vastearvon laskennassa. Kuvassa 2.17 ensimmäisen säännön paino on min(0,75; 0,5) = 0,5. Mamdani-päättelyssä takajäsenen sumeasta joukosta yleensä joko ”leikataan” huippu pois painon korkeudelta tai sitten painolla kerrotaan kaikki takajäsenen jäsenyysasteet. Kuvassa 2.17 on leikattu takajäsenien huiput pois ja vain tummia alueita käytetään lopputuloksen laskennassa.

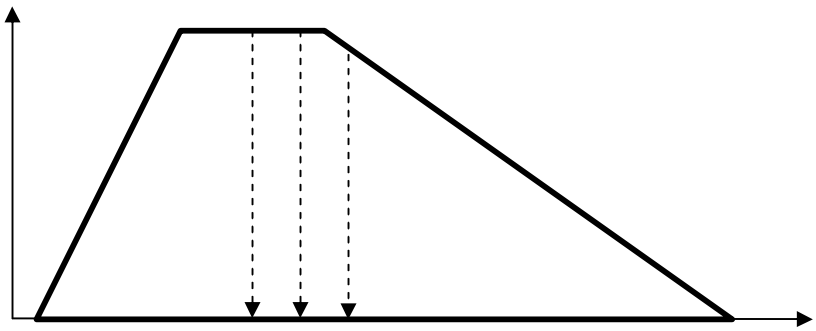
4. Lopullinen sumea vastearvo saadaan sitten yhdistämällä takajäsenien tummat alueet eli laskemalla niiden jäsenyysasteiden maksimiarvot. Jos tarvitsemme myös vastaavan täsmällisen vastearvon, saamme sen täsmällistämisen (defuzzification) avulla. Tämä täsmällinen arvo voi olla esimerkiksi (kuva 2.17)

   - vastaavan sumean joukon painopiste eli sen jäsenyysasteilla painotettu vaaka-akselin arvojen aritmeettinen keskiarvo (center of gravity),
   - ”mediaani” eli piste, jonka molemmin puolin on 50 % vastejoukon pinta-alasta
   - tai keskimmäinen niistä arvoista, joilla on maksimaaliset jäsenyysasteet vastejoukossa (mean of maxima, ”moodi”).

5. Kuvassa 2.17 on laskettu sumean vastearvon painopiste, joka on kuvan yläreunassa esitetty arvo 2,98 (myös paksu pystyviiva lopullisen arvon yhteydessä). Siis päättelymallimme ehdottaa tenttien 1 ja 2 pistemäärien (30 ja 75) perusteella loppuarvosanaksi 2,98 (eli tarvittaessa pyöristettynä = 3).
6. Samalla tavalla on vastearvo laskettu kuvan 2.19 tapauksessa, ja koska etujäsenissä on vain yksi komponentti, ne ovat samalla suoraan takajäsenien painoja.



Kuva 2.17. Sumea Mamdani-päättely: Syötearvojen etäisyys sääntöjen etujäsenien joukoista määrittää sääntöjen painot. Tällä tavoin painotettujen takajäsenien avulla saadaan vastearvo.



Kuva 2.18. Sumean joukon täsmällistäminen (vasemmalta oikealle): maksimien keskikohta, ”mediaani” ja painopiste.

Takagi-Sugeno –päättelyssä taas me tavallisesti muodostamme sumeat säännöt tutkimusaineiston perusteella, ja tällöin me sovellamme yleensä edellä

kuvattuja ryhmittelyanalyysin menetelmiä. Alustavat säännöt perustuvat siis datapisteiden ryhmien keskuksiin.

Tagaki-Sugeno –päättelyssä sumeiden sääntöjen painot lasketaan samoin kuin edellä Mamdani-päättelyssäkin vaiheissa 1-2, mutta takajäsenien osalta käytämme erilaista laskentaa. Tässä tapauksessa sääntömme ovat muotoa [65]

1. Jos … , niin $Y = w \cdot a$
2. Jos … , niin $Y = w \cdot (a_1 \cdot x_1 + a_2 \cdot x_2 + \ldots + a_n \cdot x_n + a_{n+1})$, kun n = etujäsenien komponenttien lukumäärä, $(x_1, x_2, \ldots, x_n)$ on syötevektori ja a_{n+1} on vakio.

eli takajäsenen muuttujan arvo on yksittäinen, painolla w painotettu reaaliluku tai 1. asteen painolla w painotettu lineaarinen yhtälö. Edelliset kuuluvat nollannen ja jälkimmäiset ensimmäisen kertaluvun malleihin (zero-order, first-order). Käytännössä takajäsenien arvot lasketaan nyt seuraavasti:

1. Nollannen kertaluvun tapauksessa se on siis syötearvon ja kyseisen säännön etujäsenien perusteella laskettu paino w kerrottuna reaaliluvulla a,
   $Y = w \cdot a$.
   Luku *a* taas saadaan määritettyä optimoinnin avulla.
2. Ensimmäisen kertaluvun tapauksessa muodostamme sopivan lineaarisen yhtälön jokaisen säännön osalta, ja sitten painotamme sitä edellä esitetyn mukaisesti. Lineaarisen yhtälön kertoimet, a, saadaan optimoinnin avulla. Yhtälön muuttujien lukumäärä on etujäsenen komponenttien lukumäärä.
3. Lopullinen vastearvo on molemmissa tapauksissa sitten takajäsenien painotettu keskiarvo, kun käytämme painoina edellä mainittuja arvoja w.

Kuvassa 2.19 on esitetty uudestaan luvussa 1.3 käsitelty aineisto miesten pituuksista ja painoista. Saimme silloin *subclust*-menetelmää ja kolmen ryhmän ratkaisua käyttäen nämä ryhmien keskukset:

	Ryhmä 1	Ryhmä 2	Ryhmä 3
Pituus cm	169	179	190
Paino kg	79	73	84

Nollannen kertaluvun Takagi-Sugeno –mallin tapauksessa sääntöjemme etujäsenet perustuvat siis näihin ryhmien keskuksiin, mutta takajäsenissä *a*:n

arvot lasketaan painojen *w* ja optimoinnin avulla. Niinpä sääntömme ovat muotoa:

1. Jos pituus on noin 169 cm, niin paino on $w_1 \cdot a_1$.
2. Jos pituus on noin 179 cm, niin paino on $w_2 \cdot a_1$.
3. Jos pituus on noin 190 cm, niin paino on $w_3 \cdot a_3$.

Lopullinen vastearvo on nyt näiden takajäsenien painotettu keskiarvo eli

$$(\Sigma_i w_i \cdot a_i) / \Sigma_i w_i \ (i=1,2,3)$$

Kun tässä yhteydessä laskemme arvot *a*, valitsemme niille optimoinnin avulla sellaiset arvot, että kaikki aineistomme eri syötearvojen perusteella lasketut vastearvot ovat mahdollisimman lähellä aineistomme ”todellisia” vastearvoja. Käytännössä me esimerkiksi pyrimme löytämään sellaiset a_i:n arvot, joita käytettäessä *rmse* on mahdollisimman pieni. Näistä optimointitekniikoista kerrotaan tarkemmin myöhemmin.

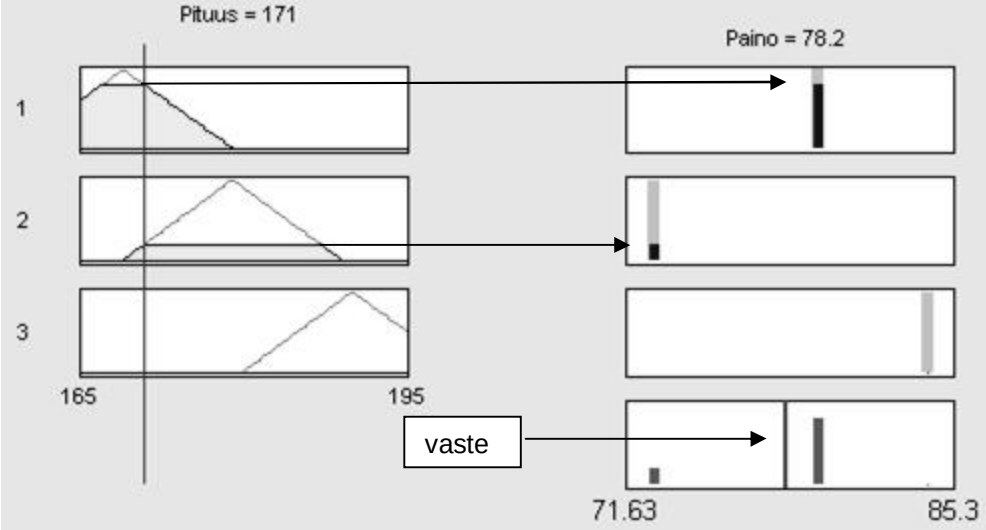
Kuvassa 2.19 on nollannen kertaluvun Takagi-Sugeno –malli aineistomme pituuksista ja painoista. Sääntömme ovat nyt optimoinnin jälkeen

1. Jos pituus on *noin 169 cm*, niin paino on 79,6 kg.
2. Jos pituus on *noin 179 cm*, niin paino on72,8 kg.
3. Jos pituus on *noin 190 cm*, niin paino on 84,2 kg.

Tässä tapauksessa siis etujäsenet perustuvat ryhmittelyanalyysissä saatuja ryhmien keskuksiin, mutta takajäsenet ovat edellä mainitun optimoinnin tuloksia, joskin ne ovat melkein samoja vastaavien ryhmien keskusten kanssa. Jos nyt esimerkiksi syötteen arvo on 171 cm, vaste lasketaan kaavalla

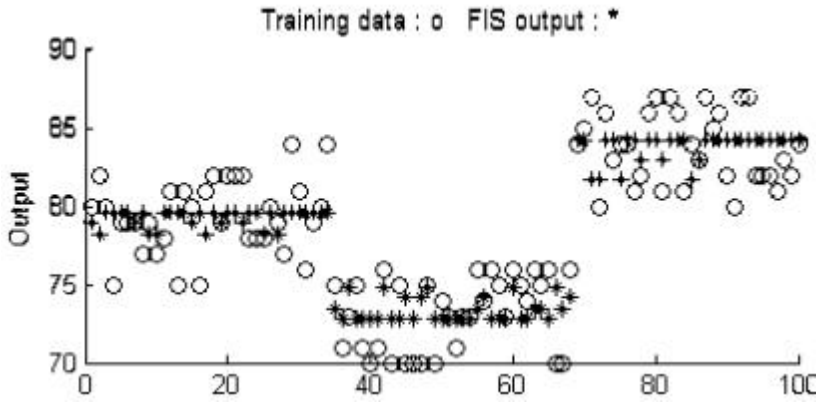
$$\text{paino} = (w_1 \cdot 79{,}6 \text{ kg} + w_2 \cdot 72{,}8 \text{ kg}) / (w_1 + w_2) = 78{,}2 \text{ kg}$$

sillä kolmannen säännön paino on nolla.



Kuva 2.19. Sumea 0. kertaluvun Takagi-Sugeno -päättely: Syötearvojen etäisyys sääntöjen etujäsenien joukoista määrittää sääntöjen painot. Takajäsenet reaalilukuja, ja vaste on näiden lukujen painotettu keskiarvo (takajäsenten painot pylväissä tummennettuna).

Kuvassa 2.20 ovat kaikki tällä tavoin lasketut vastearvot sekä vastaavat todelliset pituus-muuttujan arvot. Tämä, samoin kuin 1. kertaluvunkin menetelmä, antaa käytännössä hyviä tuloksia, vaikka takajäsenet eivät nyt olekaan sumeita joukkoja.



Kuva 2.20. Todelliset (o) ja 0. kertaluvun Takagi-Sugenon mallin (*) tuottamat arvot miesten pituus-aineistossa.

Jos sovellamme 1. kertaluvun Takagi-Sugeno -algoritmia tähän sadan miehen aineistoomme, sääntöjen etujäsenet ovat kuin edellä, mutta takajäsenet

ovat kuvan 2.21 tyyppisiä lineaaristen yhtälöiden määrittämiä ”regressiosuoria”, jotka on laskettu jokaisen kolmen ryhmän osalta erikseen. Sääntöjen takajäsenien arvot ovat syötteen (paksu nuoli) ja regressiosuorien leikkauspisteissä, ja niitä sitten painotetaan edellä esitetyllä tavalla. Vasteen arvo on edellä esitetyn mukaisesti taas takajäsenien painotettu keskiarvo. Matemaattiselta kannalta kyse on siis seuraavista funktioista kun luvut *w* ovat painoja ja luvut *a* regressiokertoimia:

1. Jos pituus on _ cm, niin paino on $w_1(a_{12} \cdot x + a_{13})$ kg.
2. Jos pituus on _ cm, niin paino on $w_2(a_{22} \cdot x + a_{23})$ kg.
3. Jos pituus on _ cm, niin paino on $w_3(a_{32} \cdot x + a_{33})$ kg.

Jos käytämme Fuzzy Logic Toolbox:in *anfisedit*-mallinnusta, sen *subclust*-menetelmä (eli ryväsmenetelmä) tuottaa kolmen säännön tapauksessa säännöt (kun vaikuttavuussäde = 0,5):

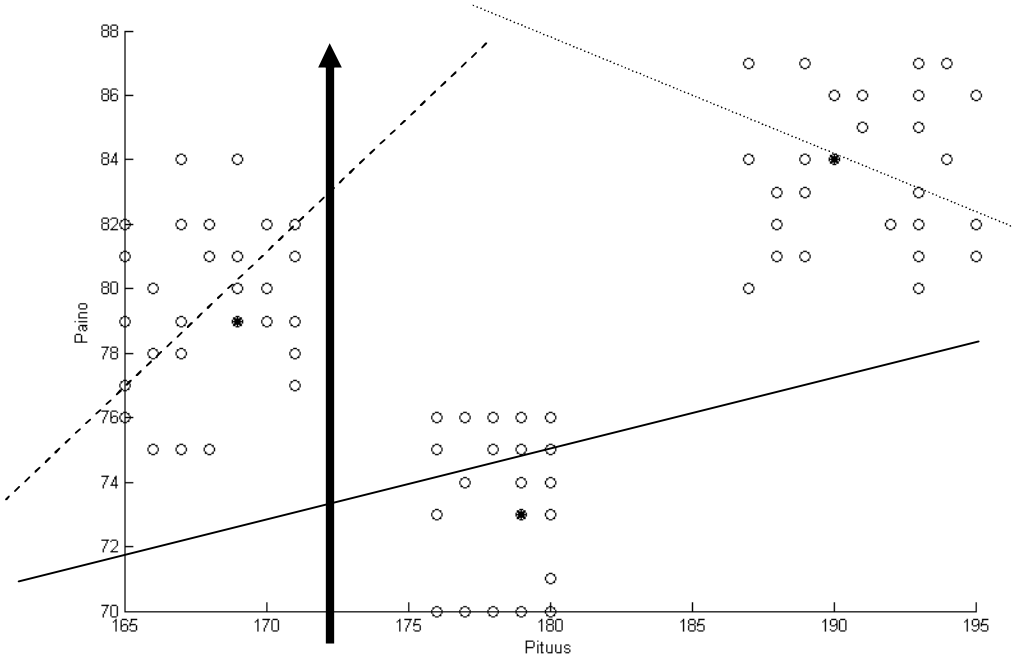
1. Jos pituus on noin 169 cm, niin paino on $w_1(0{,}9262x - 73{,}92)$.
2. Jos pituus on noin 179 cm, niin paino on $w_2(0{,}274x + 18{,}23)$.
3. Jos pituus on noin 190 cm, niin paino on $w_3(-0{,}5427x + 189{,}1)$.

Syötteen pituus = 171 cm tapauksessa vaste perustuu kuvan 2.22 perusteella vain kahteen ensimmäiseen sääntöön (koska kolmannen paino nolla), ja vaste saadaan näin ollen yhtälöstä

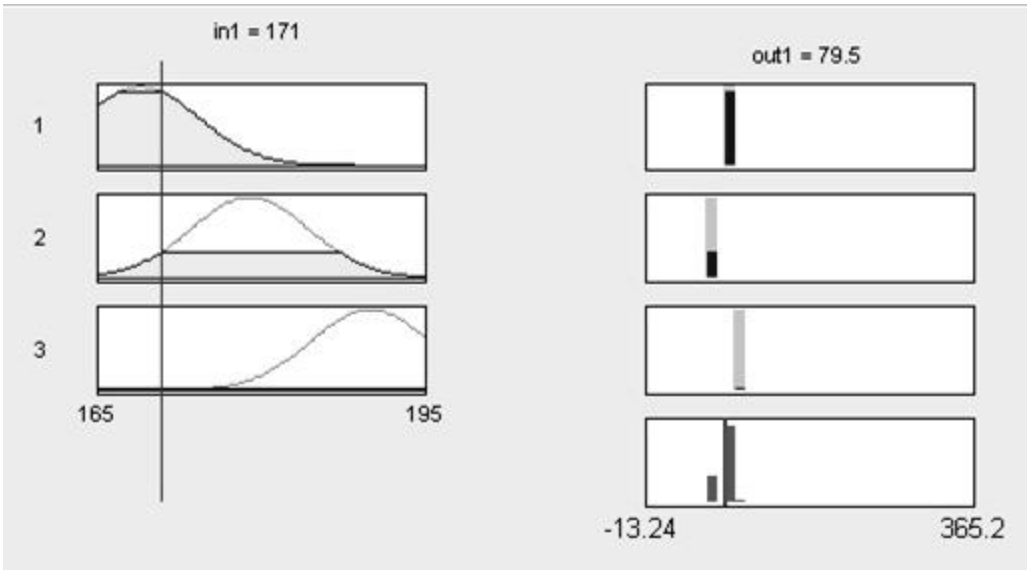
$$(w_1(0{,}9262 \cdot 171 - 73{,}92) + w_2(0{,}274 \cdot 171 + 18{,}23)\ ) / (w_1 + w_2) = 79{,}5 \text{ kg}.$$

Muut vasteiden arvot ovat kuvassa 2.23.

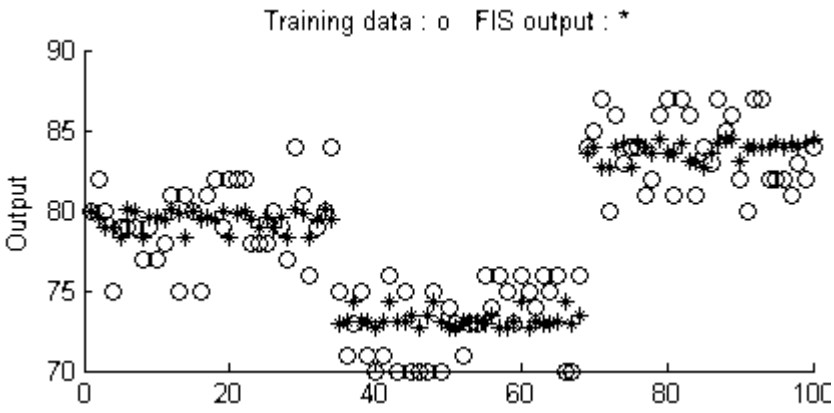
Tämä laskentatapa on analoginen kovarianssianalyysin kanssa, sillä me laskemme regressioyhtälöt ryhmittelyanalyysillä tuotetuissa ryhmissä (”käsittelyt”) syötemuuttujien perusteella (”kovariaatit”). Toinen tapa on kuvitella, että nyt sääntöjen kiinteiden takajäsenien sijasta me käytämme syötteiden perusteella laskettuja ”liukuvia” takajäsenten arvoja.



Kuva 2.21. 1. kertaluvun Takagi-Sugeno -mallissa sääntöjen takajäsenet ovat vastaaville ryhmille laskettuja regressiosuoria. Vaste on syötteen (kuvassa paksu nuoli) ja näiden suorien leikkauspisteiden painotettu keskiarvo.



Kuva 2.22. Sumea 1. kertaluvun Takagi-Sugeno -päättely: Syötearvojen etäisyys sääntöjen etu-jäsenien joukoista määrittää sääntöjen painot. Takajäsenet lineaarisia funktioita, ja vaste on näiden funktioiden arvojen painotettu keskiarvo.



Kuva 2.23. Todelliset (o) ja 1. kertaluvun Takagi-Sugenon mallin (*) tuottamat arvot miesten pituus-aineistossa.

Mamdani- ja Takagi-Sugeno –päättelyn etuja voidaan kuvata seuraavasti: Takagi-Sugeno –menetelmä on

- Laskennallisesti tehokas.
- Hyvä myös lineaarisissa malleissa.
- Käyttökelpoinen myös optimointimalleissa.
- Vastearvoissa ei yleensä epäjatkuvuuskohtia.
- Sopii hyvin tilastollis-matemaattiseen analyysiin.

Mamdani-menetelmä taas on

- Paljon ihmismäistä päättelyä muistuttava.
- Säännöt hyvin ihmiselle ymmärrettävässä muodossa.

Seuraavaksi tarkastellaan, kuinka sumean mallin viritys voidaan suorittaa neuroverkkojen ja geneettisten algoritmien avulla.

2.4.1. Sumean mallin viritys – neuroneja ja geenejä

Jos alkuperäinen sumea mallimme ei ole vielä tarpeeksi hyvä, voimme parantaa eli virittää (tune) sitä esimerkiksi neuroverkkojen ja geneettisten algoritmien avulla. Tavallisesti tämä viritys tarkoittaa, että muokkaamme mallimme sumeiden joukkojen sijaintia ja muotoa paremman lopputuloksen, esimerkiksi pienimmän mahdollisen *rmse*:n arvon saavuttamiseksi. Virittäminen voi liittyä myös mallin ristiinvalidointiin, jota käsitellään kirjan loppuosassa.

Sumeiden joukkojen perustyypit voidaan esittää yksikäsitteisesti muutaman arvo eli parametrin avulla. Esimerkiksi kolmion muotoinen joukko voidaan kuvata kolmella parametrilla, joukon maksimikohta ja ne rajapisteet, joiden välillä saadaan nollaa suurempia jäsenyysasteita. Tämän jälkeen voimme laskea joukon jäsenyysasteet kaikissa pisteissä. Tilanne on analoginen normaalijakaumien kanssa: keskiarvon ja keskihajonnan perusteella voidaan piirtää vastaava jakauma.

Mallin viritys perustuu näiden parametrien muuttamiseen. Neuroverkoissa nämä parametrit ovat neuroneita, ja neuroverkon toimintaperiaatetta on jo selostettu edellä. Valitettavasti perinteiset neuroverkot kykenevät löytämään vain lokaalisti parhaita ratkaisuja, sillä niiden optimointimenetelmät etsivät vain parametrien alkuarvojen lähettyvillä olevia ratkaisuja. Tilanne on periaatteessa samanlainen, kuin jos me vuoristossa etsimme syvimmän laakson sijasta alkupistettämme lähimpänä olevan laakson.

Evoluutiolaskennassa (evolutionary computing), joka nimensä mukaisesti soveltaa evoluutioteoriaa, pyritään löytämään globaalisti parhaita ratkaisuja. Sumeiden mallien virityksissä voidaan erityisesti hyödyntää sen yhtä osa-aluetta, geneettisiä algoritmeja (genetic algorithms, GA) [18] [23].

Geneettisiä algoritmeja sovellettaessa me oletamme, että edellä mainitut parametrit ovat ”geenejä” ja mallissa käytetyt parametrit muodostavat ”kromosomin”. Siis jos mallissamme on 50 optimoitavaa parametriä, vastaava kromosomi käsittää yhtä monta geeniä. Optimoinnin alussa määritämme vaikkapa satunnaislukujen avulla tietyn määrän näitä kromosomeja. Tämän jälkeen laskemme, millaisia vasteita nämä kromosomit tuottavat, ja parhaimmat vasteet tuottavat kromosomit pääsevät jatkoon kun taas muut karsitaan pois. Jatkoon valittuja kromosomeja (”vanhemmat”) muokataan sitten vaihtamalla satunnaisesti geenejä niiden välillä (”risteytys”, cross-over), jolloin saadaan seuraavan

sukupolven ”jälkeläisiä”. Lisäksi voidaan satunnaislukujen avulla tuottaa aivan uusiakin geenejä kromosomeihin (”mutaatio”, mutation).

Näihin uusiin parametri-kromosomeihin perustuvien mallin vasteiden perusteella karsitaan taas huonoimmat pois ja jäljelle jääneille kromosomeille suoritetaan taas risteytyksiä ja mutaatioita edellä kuvatulla tavalla. Näin jatketaan, kunnes mallimme on tarpeeksi hyvä, ja tällöin meillä on yleensä jäljellä vain yksi parametri-kromosomi, koska kaikki kromosomit yleensä lähestyvät asteittain samaa globaalia optimipistettä (jossa pienin mahdollinen *rmse* eli syvimmän laakson pohja).

Sumeat mallit voivat sisältää muitakin parametrejä kuten sääntöjen lukumäärä tai konnektiiveihin ja täsmällistämiseen liittyviä määrityksiä.

Näiden perustietojen jälkeen olemme valmiita hyödyntämään täysipainoisesti sumeaa päättelyä, ja yleisemmin älykkäitä ja oppivia järjestelmiä, mallien rakentamisessa.

3. KONTINGENSSITAULUJEN KIELELLISTÄ TULKINTAA

Kontingenssitauluiksi (contingency table) kutsutaan tavallisesti sellaisia kaksiulotteisia taulukoita eli ristiintaulukoita (cross table), joita käytetään kahden muuttujan välisen riippuvuuden tai korrelaation tarkastelussa. Tarkasteltavat muuttujat ovat luokittelu- tai järjestysasteikon tasolla tai sitten vain muutamaa eri arvoa saavia (yleensä luokiteltuja) mitta-asteikollisia muuttujia.

Tarkastellaan SPSS-ohjelman ohessa tulevaa Voter-nimistä esimerkkiaineistoa, jossa on 1847 amerikkalaisen äänestäjän tekemä valinta vuoden 1992 presidentinvaaleissa (ehdokkaina silloin Clinton, Bush sr. ja Perot) sekä eräitä taustamuuttujia. Keskitymme tässä vain kahteen muuttujaan, vastaajien ikään (luokiteltu neljään luokkaan) ja heidän koulutustasoonsa. Tarkastelun helpottamiseksi olemme käyttäneet suomenkielisiä muuttujien nimiä ja yhdistelleet koulutustason luokkia.

Meitä kiinnostaa tässä yhteydessä tietää, onko näiden muuttujien välillä riippuvuus, ja voimme intuitiivisin perustein myös päätellä, että jos riippuvuus todetaan, ikä-koulutustaso –parilla on syy-seuraus –suhde (ainahan syytä ja seurausta ei voida määrittää). Niinpä, kuten syy-muuttujan kanssa usein tapana on, ikä on taulukkomme sarake-muuttujana.

Riippuvuuden olemassaolohan testataan yleensä χ^2-riippumattomuus- tai likelihood ratio –testillä (eli G^2-testi). Molemmissa testeissä nollahypoteesina on, että riippuvuutta ei ole, ja vaihtoehtoisen hypoteesin mukaan muuttujat ovat siis ei-riippumattomia.

SPSS-ohjelman avulla tuotetun taulun 3.2 perusteella toteamme (Analyze – Descriptives – Cross-table), että sekä χ^2- että likelihood ratio –riippumattomuustestin perusteella voimme hylätä nollahypoteesin, koska molemmissa tapauksissa p-arvo (sigficance) on tarpeeksi pieni (esimerkiksi pienempi kuin 0,05 eli 5 % merkitsevyystaso, joka on meidän asettamamme raja jatkossa). Niinpä iän ja tutkinnon välillä on ilmeisesti tilastollinen riippuvuus (tai oikeammin: ne eivät ole riippumattomia).

Jos nollahypoteesi hylätään, meitä kiinnostaa selvittää vastaavan kontingenssitaulun avulla (taulu 3.1) tarkemmin, kuinka tämä korrelaatio ilmenee.

Taulu 3.1. Ylintutkinto * Ikä_luokiteltu, ristiintaulukointi, Voter-aineisto.

			Ikä_luokiteltu				
			alle35	35 - 44	45 - 64	65 +	Total
Ylintutkinto	akat. tutkinto	Count	26	51	97	19	193
		Expected Count	45,8	46,4	64,5	36,4	193,0
		% within Ikä_luokiteltu	5,9%	11,5%	15,7%	5,5%	10,4%
		Residual	-19,8	4,6	32,5	-17,4	
	keskiaste	Count	159	159	165	35	518
		Expected Count	122,8	124,5	173,0	97,6	518,0
		% within Ikä_luokiteltu	36,3%	35,8%	26,7%	10,1%	28,0%
		Residual	36,2	34,5	-8,0	-62,6	
	peruskoulu	Count	253	234	355	294	1136
		Expected Count	269,4	273,1	379,5	214,0	1136,0
		% within Ikä_luokiteltu	57,8%	52,7%	57,5%	84,5%	61,5%
		Residual	-16,4	-39,1	-24,5	80,0	
	Total	Count	438	444	617	348	1847
		Expected Count	438,0	444,0	617,0	348,0	1847,0
		% within Ikä_luokiteltu	100,0%	100,0%	100,0%	100,0%	100,0%

Taulu 3.2. Riippumattomuustestejä, Voter-aineisto.

	Value	df	Asymp. Sig. (2-sided)
Pearson Chi-Square	1,325E2	6	,000
Likelihood Ratio	142,352	6	,000
N of Valid Cases	1847		

a. 0 cells (.0%) have expected count less than 5. The minimum expected count is 36.36.

Taulun 3.1 soluissa on esitetty sekä aineistoon perustuvat havaitut (observed) että teoreettiset odotetut (expected) frekvenssit. Odotetut frekvenssit tunnetusti kertovat meille, millaisia havaittujen solufrekvenssien pitäisi olla, kun muuttujien välistä riippuvuutta ei ole. Niinpä, jos solujen havaitut ja odotetut frekvenssit poikkeavat riittävästi toisistaan, hylkäämme nollahypoteesin. Lisäksi me voimme kontingenssitaulun avulla tarkastella, minkätyyppisestä korrelaatiosta on kyse.

Itse asiassa esimerkiksi χ^2-testin tapauksessa me laskemme jokaisen taulukon solun osalta havaitun ja odotetun frekvenssin erotuksen neliön, ja tämä sitten jaetaan vielä odotetulla frekvenssillä. Lopuksi lasketaan näiden osamäärien summa, ja pieni summa tarkoittaa muuttujien välistä riippumattomuutta ja suuri riippuvuutta (tämä summahan on χ^2-testisuureen arvo). Lisäksi voidaan vielä tarvittaessa tarkastella lähemmin, mitkä osamäärät erityisesti vaikuttavat lopulliseen summaan.

Ryhmittelyanalyysin näkökulmasta me näiden testien osalta pohdimme, missä soluissa on enemmän havaintoja kuin riippumattomuuden kannalta pitäisi, eli missä soluissa on havaintojen odotettua suurempia ”kasaumia”. Niinpä korrelaation tyypin tarkastelussa me voimme silloin keskittyä vain niihin soluihin, joissa havaitut frekvenssit ovat odotettuja frekvenssejä suurempia. Voimme siis pelkistää kontingenssitaulumme tilanteen seuraavasti kun ”●” tarkoittaa tällaista kasaumaa:

	Alle 35 v	35-44 v	45-64 v	65+ v
Akat. tutkinto		●	●	
Keskiaste	●	●		
Peruskoulu				●

Toteamme nyt, että kyse on ei-lineaarisesta korrelaatiosta, ja monesti me voimme tässä yhteydessä kuvitella muuttujien muodostavan karkean x-y –koordinaatiston, jolloin korrelaation tyyppi on vielä helpompi hahmottaa. Tilastotieteessä voidaan laskea myös tähän yhteyteen sopivia korrelaatiokertoimia kuten kontingenssikerroin (contingency coefficient), joskin näiden kertoimien arvot ovat vain suuntaa-antavia johtopäätösten kannalta.

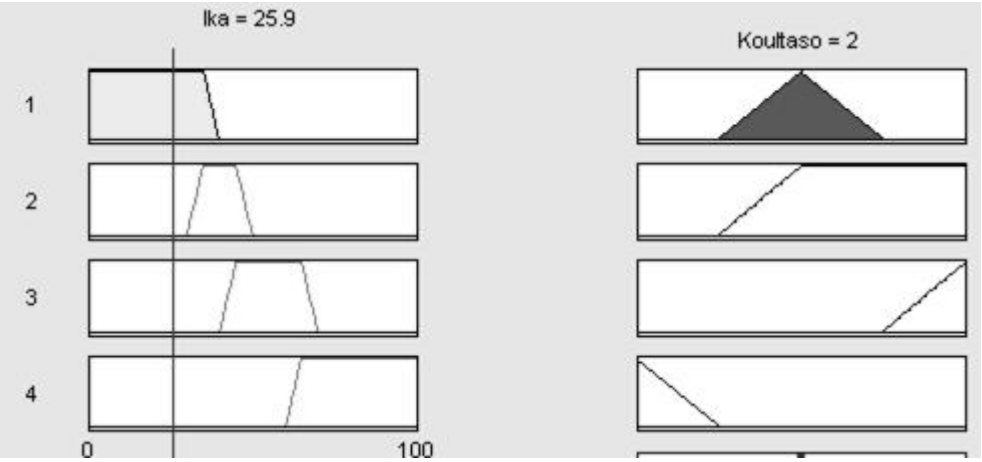
Lopullinen tulkinta voidaan esittää nyt myös kielellisinä sääntöinä soveltaen sumeiden sääntöjen määrittämisen tekniikkaa näihin ”havaintoklusterei-

hin”. Erona on kuitenkin se, että nyt kyseessä ovat täsmälliset joukot. Niinpä saamme säännöt

1. Jos on alle 35-vuotias/nuori, niin on korkeintaan keskiasteen koulutus.
2. Jos on 35-44 -vuotias/keski-ikäinen, niin on keskiasteen koulutus tai akateeminen tutkinto.
3. Jos on 45-64 -vuotias/melko vanha, niin on akateeminen tutkinto.
4. Jos on ainakin 65-vuotias/vanha, niin on korkeintaan peruskoulun käynyt.

Nämä säännöt sitten antavat meille pelkistetyn kokonaiskuvan siitä, millaisia tyypillisiä piirteitä kontingenssitaulumme (ja korrelaatiomme) käsittää.

Jos tutkimusaineistomme on edustava, voimme myös rakentaa sen perusteella sumean luokittelua tekevän mallin, jossa tässä tapauksessa ikä luokittelemattomana on syöte- ja koulutustaso vastemuuttujana. Esimerkiksi Mamdani-päättelyä ja edellä olevia kielellisiä tulkintoja käyttäen sumeat sääntömme ovat silloin kuvan 3.1 tyyppiset kun koulutustasojen koodit ovat 1 = peruskoulu, 2 = keskiaste ja 3 = akateeminen tutkinto.

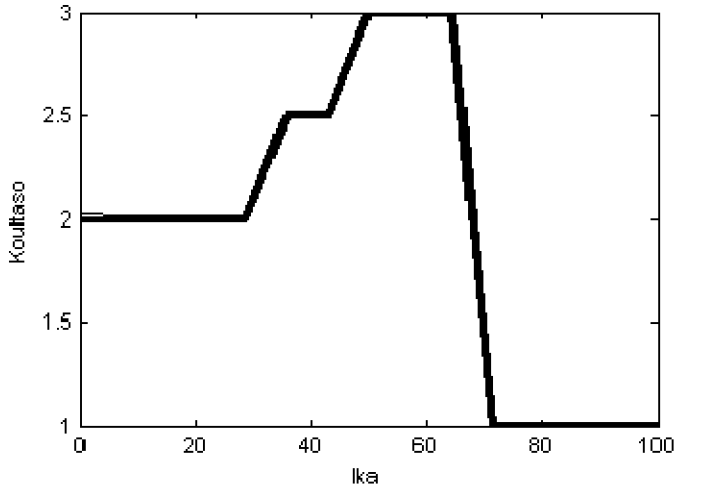


Kuva 3.1. Sumeat säännöt äänestäjien luokittelemiseksi koulutustasoihin iän mukaan Voter-aineiston kontingenssitaulun perusteella .

Vastaava mallin tuottama sovite on kuvan 3.2 mukainen, ja se on yhdenmukainen edellä esitetyn ”korrelaatiotaulun” kanssa eli korrelaation tyyppi, joka on siis tässä ei-lineaarinen, on samanlainen.

Tässä yhteydessä, kun toinen muuttujista on mitta-asteikollinen, voimme tilastotieteessä hyödyntää korrelaation laskemiseksi eta-kerrointa (eta coefficient), sillä sen avulla voidaan todeta myös ei-lineaarinen korrelaatio. Tällöin laskemme ensin ikävuosien osalta ryhmien väliset ja sisäiset varianssit koulutustasojen luokissa, ja sitten edellisen suhteen ikävuosien kokonaisvaihtelun kanssa. Tämän suhdeluvun neliöjuuri (joka siis on aina ei-negatiivinen) on sitten eta-kerroin. SPSS-ohjelma antaa tässä tapauksessa eta-kertoimen arvoksi 0,275. Tämän arvon perusteella, kuten kontingenssikertoimenkin tapauksessa, voidaan tehdä vain suuntaa-antavia johtopäätöksiä.

Nyt voisimme käyttää tätä mallia kun poimimme samasta perusjoukosta äänestäjiä ja pyrimme selittämään tai ennustamaan heidän koulutustasoaan iän perusteella. Mallin hyvyyttä voi sitten arvioida vaikkapa laskemalla, kuinka usein se tekee oikean luokittelun. Tällöin on sumean mallimme tapauksessa tarvittaessa pyöristettävä saadut vastearvot viime vaiheessa kokonaisluvuiksi. Voter-aineiston tapauksessa tällaista mallinnusta vaikeuttaa se, että 35 - 45 –vuotiaiden joukossa on sekä keskiasteen että akateemisen tutkinnon suorittaneita, joten mallin alkuperäinen vaste on heidän osaltaan näiden koulutustasojen välissä (arvo 2,5 kuvassa 3.2). Toisaalta näin saadaan esille useampaan kuin yhteen ryhmään kuuluvat tapaukset. Tähän mallinnusideaan palataan myöhemmin erotteluanalyysin yhteydessä.



Kuva 3.2. Äänestäjien luokittelu koulutustasoihin iän perusteella Voter-aineistossa Mamdani-päättelyä käyttäen.

Samalla tavalla voimme tarkastella myös vaikkapa 3-ulotteisia kontingenssitauluja kun haluamme soveltaa elaboraatioita. Voter-aineistossa myös sukupuolen ja koulutustason välillä on riippuvuus. Jos nyt haluamme vakioida (eli eliminoida) sukupuolen vaikutuksen edellä olevasta tarkastelusta, muodostamme kontingenssitaulun 3.3 SPSS-ohjelmalla tehtyjen χ^2- ja likelihood ratio –testien perusteella iän ja koulutustason riippuvuus on edelleen olemassa sekä miesten että naisten osalta, koska p-arvot ovat molemmissa tapauksissa hyvin pieniä.

Taulu 3.3. Ylintutkinto * Ikä_luokiteltu * Sukupuoli, Voter-aineisto

Sukupuoli			Ikä_luokiteltu				
			alle35	35 - 44	45 - 64	65 +	Total
nainen	Ylintutkinto akat. tutkinto	Count	12	27	51	3	93
		Expected Count	22,9	21,8	29,4	18,8	93,0
		Residual	-10,9	5,2	21,6	-15,8	
	keskiaste	Count	100	94	83	25	302
		Expected Count	74,4	70,9	95,6	61,1	302,0
		Residual	25,6	23,1	-12,6	-36,1	
	peruskoulu	Count	145	124	196	183	648
		Expected Count	159,7	152,2	205,0	131,1	648,0
		Residual	-14,7	-28,2	-9,0	51,9	
	Total	Count	257	245	330	211	1043
		Expected Count	257,0	245,0	330,0	211,0	1043,0
mies	Ylintutkinto akat. tutkinto	Count	14	24	46	16	100
		Expected Count	22,5	24,8	35,7	17,0	100,0
		Residual	-8,5	-,8	10,3	-1,0	
	keskiaste	Count	59	65	82	10	216
		Expected Count	48,6	53,5	77,1	36,8	216,0
		Residual	10,4	11,5	4,9	-26,8	
	peruskoulu	Count	108	110	159	111	488
		Expected Count	109,9	120,8	174,2	83,2	488,0
		Residual	-1,9	-10,8	-15,2	27,8	
	Total	Count	181	199	287	137	804

Taulu 3.3. Ylintutkinto * Ikä_luokiteltu * Sukupuoli, Voter-aineisto

Sukupuoli			Ikä_luokiteltu				
			alle35	35 - 44	45 - 64	65 +	Total
nainen	Ylintutkinto akat. tutkinto	Count	12	27	51	3	93
		Expected Count	22,9	21,8	29,4	18,8	93,0
		Residual	-10,9	5,2	21,6	-15,8	
	keskiaste	Count	100	94	83	25	302
		Expected Count	74,4	70,9	95,6	61,1	302,0
		Residual	25,6	23,1	-12,6	-36,1	
	peruskoulu	Count	145	124	196	183	648
		Expected Count	159,7	152,2	205,0	131,1	648,0
		Residual	-14,7	-28,2	-9,0	51,9	
	Total	Count	257	245	330	211	1043
		Expected Count	257,0	245,0	330,0	211,0	1043,0
mies	Ylintutkinto akat. tutkinto	Count	14	24	46	16	100
		Expected Count	22,5	24,8	35,7	17,0	100,0
		Residual	-8,5	-,8	10,3	-1,0	
	keskiaste	Count	59	65	82	10	216
		Expected Count	48,6	53,5	77,1	36,8	216,0
		Residual	10,4	11,5	4,9	-26,8	
	peruskoulu	Count	108	110	159	111	488
		Expected Count	109,9	120,8	174,2	83,2	488,0
		Residual	-1,9	-10,8	-15,2	27,8	
	Total	Count	181	199	287	137	804
		Expected Count	181,0	199,0	287,0	137,0	804,0

Silloin miesten osalta havaintoja on kasautunut seuraavasti:

	Alle 35	**35-44**	**45-64**	**65+**
Akat. tutkinto			●	
Keskiaste	●	●		
Peruskoulu				●

Naisten tapauksessa meillä on sama tilanne eli

	Alle 35	35-44	45-64	65+
Akat. tutkinto			●	
Keskiaste	●	●		
Peruskoulu				●

Kielellisinä sääntöinä tuloksemme ovat siis miesten tapauksessa muotoa

1. Jos on mies ja alle 35-vuotias, niin korkeintaan keskiasteen koulutus.
2. Jos on mies ja 35-44 -vuotias, niin korkeintaan keskiasteen koulutus.
3. Jos on mies ja 45-64 -vuotias, niin akateeminen tutkinto.
4. Jos on mies ja ainakin 65-vuotias, niin korkeintaan peruskoulun käynyt.

Naisten osalta saamme saman tuloksen, eli

1. Jos on nainen ja alle 35-vuotias, niin korkeintaan keskiasteen käynyt.
2. Jos on nainen ja alle 35-44 -vuotias, niin korkeintaan peruskoulun käynyt.
3. Jos on nainen ja alle 45-64 -vuotias, niin akateeminen tutkinto.
4. Jos on nainen ja ainakin 65-vuotias, niin korkeintaan peruskoulun käynyt.

Näitä sääntöjä voidaan tarvittaessa esittää sujuvammassakin muodossa, mutta tärkeintä tässä yhteydessä on oppia, kuinka voimme kyseisille tilastollisille tuloksille antaa helposti kielellisiä tulkintoja. Tilastotieteessähän tämäntyyppinen tarkempi analyysi tehdään elaboraation tapauksessa perinteisesti esimerkiksi log-lineaarisilla malleilla.

Edellä on koulutustasoa käytetty tietyssä mielessä selitettävänä muuttujana, ja tältä pohjalta on rakennettu yhden selittäjän käsittävä sumea mallikin. Koska selitettävä muuttuja on luokitteluasteikollinen, tilastotieteessä voidaan tässä tapauksessa rakentaa siihen tarkoitukseen sopivia malleja kuten logistiset (logistic) ja multinomiaaliset (multinomial) regressiomallit. Niiden tarkastelu kuitenkin tässä kirjassa sivuutetaan, ja rakennamme vain vastaavia neuro- ja geneettis-sumeita malleja.

Jos taas vaihdamme Voter-aineistossa koulutustason selittäjäksi ja iän selitettäväksi muuttujaksi, voimmekin perinteisesti soveltaa varianssianalyysiä. Tätä lähestymistapaa ja vastaavia sumeita malleja tarkastelemme seuraavaksi.