Pro gradu -tutkielma

Matematiikan aineenopettaja

Anna Karenina -periaate

Kohti matematiikan oppimisympäristössä toteutettua virheiden luokittelua

Hannu Tiitu

Kesäkuu 2016

Ohjaajat:	yliopistonlehtori Mika Koskenoja
	dosentti Jarmo Malinen
	dosentti Antti Rasila
Tarkastaja:	professori Juha Oikkonen

Helsingin yliopisto
Matemaattis-luonnontieteellinen tiedekunta
Matematiikan ja tilastotieteen laitos

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Tiedekunta/Osasto	Laitos
Matemaattis-luonnontieteellinen tiedekunta	Matematiikan ja tilastotieteen laitos

Tekijä

Hannu TIITU

Työn nimi

Anna Karenina -periaate. Kohti matematiikan oppimisympäristössä toteutettua virheiden luokittelua

Oppiaine

Matematiikan aineenopettaja

Työn laji	Aika	Sivumäärä
Pro gradu -tutkielma	Kesäkuu 2016	61

Tiivistelmä

Matemaattisten taitojen mittarina toimivat usein erilaiset tehtävät ja harjoitukset. Anna Karenina -periaatteen mukaisesti oikeat vastaukset eivät ole mielenkiintoisia, ne ovat kaikki samanlaisia. Sen sijaan väärä vastaus avaa mahdollisuuden oppimiselle, jos se osataan tulkita oikein. Opiskelijan käsitteenmuodostus ja taidot paljastuvat virheissä.

Sähköiset oppimateriaalit mahdollistavat oppimateriaaleja, joita on ollut mahdoton tuottaa painetussa muodossa. Niiden avulla on mahdollista luoda vuorovaikutteisia aineistoja, jotka reagoivat opiskelijan toimintaan. Myös tehtävien automaattinen tarkastaminen ja henkilökohtainen palaute on mahdollista. Tällainen oppimisympäristö on Stack, josta on pitkät kokemukset Aalto-yliopiston insinöörimatematiikan opetuksessa.

Sähköisten oppimisympäristöjen keräämää tietoa voidaan käyttää oppimisen analysointiin. Esimerkiksi virheluokittelua voidaan automatisoida. Tällöin luokittelumallin tulee olla luotettava. Jos mallin mukainen luokittelu ei onnistu ihmiseltä, ei kone suoriudu siitä sen paremmin.

Tutkielmassa käsitellään lääkelaskennan opetukseen laaditun 4 Cs -mallin mukaan tehtyä virheluokittelumallia, jossa virheet jaetaan neljään luokkaan: 1) laskuvirhe, 2) yksikkömuunnosvirhe, 3) käsitteellinen virhe ja 4) virhettä ei voida luokitella. Tämän mallin luotettavuutta selvitetään antamalla luokittelijoiden käyttää sitä opiskelijoiden lääkelaskentakokeissa tuottamiin virheisiin.

Luokitteluja analysoidaan k-means-klusterointialgoritmilla ja osoittautuu, että ne ovat lähellä toisiaan. Lisäksi luokittelu on stabiili, se toimii johdonmukaisesti vaikka käytössä olisi vain osa datasta. Näin ollen pedagoginen malli 4 Cs on mahdollinen perusta virheluokittelulle, joka voidaan toteuttaa esimerkiksi Stackin avulla. Tosin tässä on paljon haasteita, ja lisää tutkimusta ja kehitystä tarvitaan tämän kaltaisen tekoälyjärjestelmän rakentamiseksi.

Avainsanat

oppimisympäristö, virheluokittelu, automaattinen tarkastaminen, matematiikan opetus, lääkelaskenta, Stack, k-means, 4 Cs

Säilytyspaikka

Kumpulan kampuskirjasto, Gustaf Hällströmin katu 2 (PL 64), 00014 Helsingin yliopisto
E-thesis: http://hdl.handle.net/10138/163211

Muita tietoja

Tutkielman ohjasivat yliopistonlehtori Mika Koskenoja, dosentti Jarmo Malinen sekä dosentti Antti Rasila.

Vuokolle & Iskalle

Bertalle & Saulille

Hilkalle & Ollille

Alkulause

Tämä pro gradu -tutkielma on tehty Helsingin yliopiston matematiikan ja tilastotieteen laitokselle. Työn ohjaajina toimivat Helsingin yliopistolla yliopistonlehtori Mika Koskenoja sekä Aalto-yliopistolla dosentit Jarmo Malinen ja Antti Rasila. Suuret kiitokset heille neuvoista ja kannustavasta asenteesta tähän projektiin.

Kiitos myös Arcada-ammattikorkeakoulun Birgitta Dahl ja Tore Ståhl, teidän kanssanne oli mukava tehdä yhteistyötä! Tutkielmassa esitetyt lääkelaskentaan liittyvät virheluokittelut tehtiin Arcadan sairaanhoitajakoulutuksen myötävaikutuksella.

Haluan kiittää myös muita tietokoneavusteisen opetuksen parissa Aalto-yliopiston matematiikan ja systeemianalyysin laitoksella työskennelleitä: Matti Harjula, Linda Havola, Krista Linnoinen, Helle Majander, Kimmo Ojala ja Jarkko Savela. Eikä unohtaa sovi loppuja kahvihuoneen jengistä: Atte Aalto, Mikko Eronen, Janne Korvenpää, Eerno Niemenmaa, Tri Quach, Stratos Staboulis ja Lauri Viitasaari.

Lopuksi rakkaat kiitokset kotijoukoille: Karoliina, Lauri, Minea ja Kati.

Otaniemessä 5. kesäkuuta 2016

Hannu Tiitu

Sisältö

1

Johdanto

Tämä pro gradu -tutkielma on kirjoitettu Aalto-yliopiston matematiikan ja systeemianalyysin laitoksella. Kirjoittaja on työskennellyt tutkimusapulaisena Tietokoneavusteisen matematiikan opetuksen tutkimusryhmässä ja osallistunut erilaisiin matematiikan oppimisympäristöjä ja automaattisesti tarkastettavia tehtäviä käsitteleviin projekteihin.

Tämän tutkielman runkona toimivat seuraavat julkaisut I ja II, joiden sisältöjä on täydennetty ja yhdistetty kokonaisuudeksi. Artikkelissa II esitetty virheluokittelun luotettavuuden arviointi klusteroinnin avulla, joka on tämän tutkielman keskeinen aihe, on kirjoittajan oma idea ja toteutus.

I Rasila, A., Malinen, J. & Tiitu. H. 2015. On automatic assessment and conceptual understanding. Teaching Mathematics and its Applications 34 (3), 149–159. http://dx.doi.org/10.1093/teamat/hrv013

II Dahl, B., Ståhl, T., Malinen, J., Rasila, A. & Tiitu, H. 2014. Diagnosing nursing students' errors in medication calculation. Designing a method based on the 4 Cs teaching model for analysing mathematical proficiency. Teoksessa J. Viteli & A. Östman (toim.) Tuovi 12: Interaktiivinen tekniikka koulutuksessa 2014 -konferenssin tutkijatapaamisen artikkelit. TRIM Research Reports 12. Tampere: Tampereen yliopisto, 82–92. http://urn.fi/URN:ISBN:978-951-44-9561-8

2

Virheet hyödyksi matematiikan opetuksessa

Kaikki onnelliset perheet ovat toistensa kaltaisia, jokainen onneton perhe on onneton omalla tavallaan. (Толстой 1875/1963)

Tämä Tolstoin Anna Kareninan aloitus on yksi klassikkokirjallisuuden kuuluisimmista. Sen viestinä on, että ollakseen onnellinen, perheen tulee täyttää monia erilaisia vaatimuksia, kuten vaikka hyvä terveys ja turvallinen taloudellinen tilanne. Jos jokin monista edellisen kaltaisista ominaisuuksista puuttuu, perhe ei ole onnellinen. Koska perhe voi olla onnellinen vain yhdellä tavalla, eli täyttämällä kaikki onnellisuuden ehdot, voidaan ajatella, että on vain yhdenlaista onnellisuutta. Sen sijaan onnettomalle perheelle riittää, että yksikin ehdoista jää täyttymättä. Näin kukin onneton perhe voidaan nähdä onnettomaksi omalla yksilöllisellä tavallaan.

> Все счастливые семьи похожи друг на друга, каждая несчастливая семья несчастлива по-своему. – Л. Н. Толстой

Jared Diamond (1994; 1997) laajensi Tolstoin avauslauseen idean periaatteeksi, ja nimesi sen kirjan mukaan. Anna Karenina -periaatteesta puhutaan tarkasteltaessa monimutkaisten hankkeiden tai tapahtumaketjujen lopputulosta. Sellaisen positiivinen päätös vaatii hyvin monien osatekijöiden tai ehtojen suosiollista lopputulosta, kun taas epäonnistumiseen riittää yhdenkin osan pettäminen. Anna Karenina -periaatetta on käytetty havainnollistamaan ilmiöitä esimerkiksi kuluttajatutkimuksessa ja psykologiassa, joissa ihmisen päätöksenteon mallinnus on monimutkainen tehtävä.

Matemaattisten taitojen paljon käytettynä mittarina toimivat erilaiset tehtävät ja harjoitukset, joiden suoritusta opettajat arvioivat ja arvostelevat. Tämä on jokaiselle tuttua koulusta. Arvioidessaan suorituksia opettaja tulee tutustuneeksi opiskelijan tapaan hahmottaa matemaattisia käsitteitä ja, ja hänen erilaisiin taitoihinsa selviytyä matemaattisten ongelmien ratkaisemisesta.

Tehtävään oikein vastannut opiskelija on opettajan kannalta menestys. Valitut mallit ja oppimateriaalit ovat toimineet, ja opiskelija on pystynyt tuottamaan oikean vastauksen. Sille ei saada varmuutta, ovatko tehtävän sisältämät matemaattiset käsitteet tulleet oikein ymmärretyiksi, mutta muutakaan todistusta asiasta ei oikea vastaus anna. Tässä mielessä oikea vastaus ei siis olekaan opettajalle arvokas.

Sen sijaan väärä vastaus, vaikka onkin opiskelijan kannalta epätoivottu, voi olla opettajalle hyvinkin hyödyllinen. Väärän vastauksen avulla on mahdollista päästä tutustumaan opiskelijan ajatteluun, siihen millaisena matematiikka hänelle näyttäytyy. Anna Karenina -periaatteen mukaisesti virheellinen vastaus näyttää opiskelijan yksilöllisen matematiikan. Tätä tietoa voi olla mahdollista käyttää edelleen hyväksi opetuksen suunnittelussa.

Koulu ja opetus ovat vuosikymmenten saatossa rakentuneet virheitä välttäväksi, virhe käsitetään epäonnistumisen merkkinä. Tämä ajattelu ei ole kovin hedelmällinen, sillä toisaalta virheet ovat mahdollisuuksia oppia ja kehittyä – ja tämän lisäksi melko inhimillisiä.

Virheiden analysointi on perinteisesti ollut osa pedagogista tutkimusta. Nykyiset tietotekniset oppimisympäristöt mahdollistavat suuren tietomäärän keräämistä opiskelijoiden vastauksista sekä erilaisten analyysien automatisoinnin. Tulevaisuutta ovat sellaiset matematiikan oppimisympäristöt, joissa opiskelijat tekevät tehtäviä itsenäisesti, ja tehtävien tarkastaminen tapahtuu automaattisesti. Lisäksi tulevaisuuden oppimisympäristö analysoi opiskelijan etenemistä opinnoissa hänen suoritustensa perusteella, jolloin tähän on mahdollista reagoida antamalla opiskelijalle hyödyllisiä oppimateriaaleja.

Tämä tarjoaa mahdollisuuksia perinteiselle virheanalyysille, jonka tietokoneen nopeus ja täsmällisyys voi nostaa uudelle tasolle. Tällaisen tekoälyjärjestelmän toteuttaminen käytännössä on kuitenkin kaikkea muuta kuin yksinkertaista. Tässä tutkielmassa perehdytäänkin erääseen keinoon analysoida virheluokittelua itsessään, sillä luokittelun automatisoinnissa on järkeä vain sellaisessa tilanteessa, jossa luokittelu on luotettava.

Ensin luodaan katsaus matematiikan opetukseen ja oppimiseen. Mitä matematiikan taito on, millaisena matematiikka näyttäytyy opiskelijalle ja mikä on opettajan rooli. Lääkelaskenta esitellään esimerkkinä eri ammateissa tarvitusta matematiikasta.

Matematiikan kommunikaation traditio painetuista kirjoista moderneihin tietoteknisiin oppimisympäristöihin on muokannut opettamista. Nykypäivän vuorovaikutteiset oppimateriaalit mahdollistavat henkilökohtaisen opettamisen ja tehtävien automaattisen tarkastamisen. Tähän on mahdollista yhdistää oppimisanalytiikkaa, jolloin oppimisympäristö voi ohjata opiskelijan oppimista. Stack on esimerkki oppimisympäristöstä, jossa tämä on mahdollista.

Tutkielmassa hahmotellaan automaattisesti tarkastettavien tehtävien oheen toteutettua virheluokittelua, jonka avulla on mahdollista saada tietoa opiskelijan oppimisprosessin etenemisestä. Esimerkkitapaukseksi otetaan lääkelaskenta, jolle rakennetaan virheluokittelumalli erääseen lääkelaskennan opetusmalliin perustuen. Onko tämä luokittelumalli luotettava? Tätä tutkitaan opettajien tekemien luokittelujen avulla. Analyysissa käytetään klusterointia, jolla nähdään, ovatko opettajat yksimielisiä luokitteluissaan. Jos näin käy, on valittu pedagoginen malli hyvä perusta virheluokittelun tekemiselle lääkelaskennan opetuksessa.

3

Matematiikka käsitteiden rakennelmana

3.1 Matematiikka keskellämme

Ennen kuin matematiikan opettamista ja opiskelemista lähdetään enempää automatisoimaan, mietitään hetki millaista tietoa matematiikan nimellä meille oikeastaan opetetaan.

Matemaattisia käsitteitä ja ajattelua nähdään kaikkialla, eikä tämä ole katoamassa teknistyvästä yhteiskunnasta, päinvastoin. Matematiikan luonne on muuttunut. Erilaiset laskimet ja laskentaohjelmat ovat toisaalta vähentäneet tarvetta käsin laskemiselle, mutta ne ovat tuoneet mukanaan monille aloille uudenlaisia työskentelytapoja, joissa tarvitaan matemaattista ajattelua. Nykyisin ihmiset törmäävät formaaleihin konsepteihin enemmän kuin koskaan aikaisemmin. Nykyteknologian kanssa työskentely ja yhä kasvavat vaatimukset työn tehokkuuden nostamiselle tekevät mahdottomaksi palata menneisiin työtapoihin, jotka ovat nykyajattelun mukaan tehottomia.

Vaikka matematiikka voi näyttäytyä hankalana ja vaikeana oppia, siltä ei voi välttyä arkisessakaan elämässä. Huonoilla oppimistuloksilla ja matematiikan taidolla voi olla dramaattisia seurauksia. Esimerkiksi tästä käy tilanne, jossa vastasyntyneelle lapselle annetaan laskuvirheen seurauksena moninkertainen yliannostus lääkettä (Dekker 2007). Tällaisen surullisen tapahtuman syitä miettittäessä saattaa vastaan tulla jokapäiväisiä rutiineja suorittavan ammattilaisen kognitiivinen prosessi. Tilanteesta voidaan kysyä epämiellyttäviä kysymyksiä: miten tällainen tapahtumien kulku oli mahdollinen? Miksi desimaalipilkku oli väärässä paikassa? Miten lopputulos saattoi olla niin väärin, eikä asiaa huomattu, ennen kuin oli myöhäistä? Olisiko olemassa keinoja, joilla tällaiset tapahtumat voitaisiin estää?

Monien mahdollisten syiden joukossa selitys tapahtumille voi olla lääkintäammattilaisten puutteelliset tai jopa virheelliset käsitykset matemaattisista konsepteista, kuten yksikkömuunnokset tai suuruusluokkien arviointi. Voi myös olla, että sinänsä oikeaa tietoa ei vain sovellettu oikein syystä tai toisesta. Erehtyminen on inhimillistä, ja syyt virheeseen voivat olla yhtä monimutkaisia kuin inhimillinen ajattelu itse.

3.2 Monta näkökulmaa matematiikan taitoon

Matematiikan opetuksen tutkimuksessa matemaattisia taitoja on pyritty kategorisoimaan monilla eri tavoilla. Taustalla on pyrkimys pilkkoa laaja ja monimutkainen ajatus matemaattisesta osaamisesta pienempiin osiin, joita on yksinkertaisempi tarkastella. Opiskelijoiden erilaiset vahvuudet tukevat oppimisen prosesseja eri tavoin. Osat luovat eräänlaisen *osaamisavaruuden* dimensiot. Matemaattista osaamista syntyy Anna Karenina -periaatteen mukaisesti silloin, kun kaikki osaamisavaruuden komponentit "loksahtavat paikoilleen". Miten näitä eri komponentteja voisi havainnoida tai mitata? Miten paljon eri osa-alueet korreloivat keskenään? Nämä luonnolliset kymykset johdattavat suoraan laskuvirheiden analysointiin, josta lisää luvussa 7.

Seuraavassa muutamia esimerkkejä matemaattisen taidon jaottelusta eri osa-alueisiin. Haapasalon (2004) mukaan matemaattinen osaaminen voidaan jakaa kahteen osa-alueeseen, jotka täydentävät toisiaan. *Proseduraalisella tiedolla* henkilö selviytyy erilaisista matemaattiseen ongelmanratkaisuun liittyvistä tilanteista. Näihin kuuluvat erilaiset rutiinit, sekä miten hyvin henkilö pystyy yhdistämään erilaisia tekniikoita ja keinoja matemaattisten tehtävien ratkaisemiseksi. Jälkimmäisiin taitoihin liittyy läheisesti toinen osa-alue, *konseptuaalinen tieto*. Pystyäkseen käyttämään erilaisia työkaluja, henkilöllä tulee olla ymmärrys matemaattisen tiedon luonteesta, erilaisista käsitteistä ja niiden välisistä yhteyksistä. Matematiikan opetuksessa nämä osa-alueet kehittyvät usein vuorotellen. Kun matemaattinen käsite on opeteltu konseptuaalisella tasolla, sen hyödyntäminen muiden käsitteiden rakennuspalikkana edellyttää usein proseduraalisia taitoja.

Kilpatrick, Swafford ja Findell (2001) ehdottavat, että matematiikan taidot koostuvat seuraavista viidestä toisiinsa liittyvistä osioista:

Konseptuaalinen ymmärtäminen Matematiikka koostuu käsitteistä ja niiden välisistä operaatioista ja suhteista.

Proseduraaliset taidot Matemaattisten työkalujen ja menetelmien käyttö tiedon muokkaamiseksi ja uusien käsitteiden tuottamiseksi. Esimerkiksi yhtälön sieventäminen on proseduraalinen taito.

Strategiset taidot Matemaattisten ongelmien ja niiden ratkaisemisen lähestyminen erilaisten menetelmien avulla. Konseptuaaliset ja proseduraaliset kyvyt vaihtelevat.

Deduktiiviset kyvyt Looginen ajattelu ja esimerkiksi suoriutuminen matemaattisten lauseiden todistamisesta.

Mielenkiinto Matematiikka nähdään tarkoituksenmukaisena, tärkeänä ja järkevänä.

Näistä osa-alueista proseduraalisten taitojen hankkimisen tärkeys on korostunut suomalaisessa opetuskulttuurissa. Koulu- ja yliopistomatematiikka painottuvat sisällöiltään eri tavoin. Deduktiiviset kyvyt korostuvat yliopistomatematiikassa, jossa erilaisten todistusten osuus on merkittävä (Joutsenlahti 2005). Kolme jäljelle jäävää osa-aluetta ovat erityisen tärkeitä insinööreille ja luonnontieteilijöille, joiden tulee tunnistaa ja ratkaista erilaisia matemaattisia ongelmia, joista luonnontieteen ja teknologian maailma rakentuu. *Käsitteellinen ymmärtäminen* on matemaattisen ongelmanratkaisutaidon ydin, joka tekee siitä käyttökelpoisen kaikkien erilaisten matemaattisten tilanteiden käsittelemiseen.

3.3 Opiskelijan matematiikka

Toisin kuin monien muiden elämänalojen opiskelussa, matematiikan oppiminen vaatii paljon kommunikaatiota aiemmin opitun ns. alemman tason tiedon kanssa. Uuden tiedon rakentamisessa tarvittujen käsitteiden lisäksi tarvitaan huolellisesti valittuja esimerkkejä, joiden avulla uutta käsitettä voidaan tuoda tutuksi ja suorittaa harjoittelua. Näin opiskelija lähestyy uutta tietoa ottaen sen haltuunsa pienin askelin edeten. Piagetin (1929) mukaan tässä tapahtuu joko tiedon *assimilaatio* tai *akkomodaatio*. Assimilaatiossa uusi tieto liitetään osaksi aiemman tietämyksen muodostamaa tietorakennetta, kun taas akkomodaatiossa opiskelijan aiempi käsitys ei pysty selittämään uutta, jolloin tietorakenne itsessään joutuu mukautumaan voidakseen vastaanottaa uuden tiedon.

Esimerkkinä tästä on matemaattinen käsite luku. Lapsi oppii ensiksi luvun tarkoittavan lähinnä lukumäärää. Käsite kuitenkin syventyy tietojen karttuessa osaksi algebrallista järjestelmää. Tässä yhteydessä esitettynä käsite luku laajenee käsittämään abstraktimpia muotoja kuten negatiiviset luvut, päättymättömät desimaaliluvut, murtoluvut, reaaliluvut ja niin edelleen (Skemp 1976).

Edellä kuvattu skemaattinen oppimisprosessi on *konstruktiivinen* (eräänlaisena vastakohtana *behavioristiselle*), koska opiskelijan oman mentaalisen prosessin katsotaan olevan keskeinen osa oppimista. Behavioristisesta näkökulmasta oppiminen olisi vain ulkoisten ärsykkeiden aiheuttamaa opiskelijan toimiessa ainoastaan tiedon vastaanottajana.

Opettajan rooli konstruktiivisessa oppimisessa on toimia fasilitaattorina kehittäen oppimiselle suotuisat olosuhteet. On monia tapoja käsitteellistää ihmisen matemaattisten käsitteiden ymmärrystä. Tallin (2008) mukaan matematiikan käsittäminen on jaettu kolmeen maailmaan: *konseptuaalisesti ilmenevään*, *symbolisesti havainnolliseen* ja *aksiomaattis-formaaliin*. Skempillä (1976) on käsitteet *instrumentaalinen* ja *relationaalinen ymmärtäminen*, joista relationaalinen vertautuu edellä esitettyyn konseptuaaliseen ymmärtämiseen.

3.4 Lääkelaskenta on arjen matematiikkaa

Matematiikan opetuksesta puhuttaessa kiinnitetään usein huomiota joko pienten lasten opetukseen tai yliopistomatematiikkaan. Vähäisemmälle huomiolle jää moni sellainen ala, jonka luonteeseen matematiikkaa ei yleensä liitetä, mutta jonka arkisessa hallitsemisessa sillä on kuitenkin keskeinen asema. Hyvänä esimerkkinä tästä on edellä esitetty sairaanhoitajien harjoittama lääkelaskenta.

Useimmissa terveydenhuollon ammateissa tarvittu matematiikka ei ole kovin monimutkaista. Sairaanhoitajat eivät joudu tekemisiin derivaattojen ja integraalien kanssa, vaan peruslaskutoimitukset ja muu perusaritmetiikka ovat tarvittuja taitoja. Luonnollisesti hoitotyössä vastaan tulevien matemaattisten ongelmien ratkaisemisessa tarvitaan loogista päättelyä, deduktiota ja kriittistä arviointia.

Tällaisia taitoja opitaan jo peruskoulussa matematiikan opetuksen alkumetreiltä lähtien. Huhtalan (2000) mukaan terveydenhuoltoalalle suuntautuvat nuoret eivät useinkaan ole matemaattisesti orientoituneita. Sen sijaan matematiikan osaamisen todentamiseen usein liittyvät tiukat vaatimukset ja toistuvat kokeet saattavat hämmentää heitä.

Farmakoterapia on kehittynyt voimakkaasti viimeisten vuosikymmenten aikana. Lääkkeitä on yhä enemmän, ja osa niistä on tehokkaampia kuin aikaisemmin. Lääkkeiden kirjo on lisääntynyt uusilla biologisilla ja kemiallisilla lääkkeillä. Tämä kehitys asettaa uusia haasteita sairaanhoitajille ja muille terveydenhuollon ammattilaisille (Sosiaali- ja terveysministeriö 2006). Samaan aikaan sairaanhoitajien koulutus, kuten muutkin alat, joutuvat toimimaan ympäristössä, jossa opiskelijoiden matemaattiset taidot ovat yhä heikommat (Røykenes & Larsen 2010; Wright 2006).

Virheelliseen lääkitykseen liittyvät hoitovirheet ovat vakava ongelma Suomessa ja kansainvälisesti (Grandell-Niemi, Hupli, Leino-Kilpi & Puukka 2003; Grandell-Niemi, Hupli, Puukka & Leino-Kilpi 2006; McMullan, Jones & Lea 2009; Pasternak 2006). Nämä ongelmat ovat niin ilmeisiä, että ne ovat herättäneet suurenkin yleisön huolen (Rantanen 2013). Arviolta 700–1700 ihmistä kuolee Suomessa vuosittain hoitovirheeseen. Tätä lukua voidaan verrata esimerkiksi liikenteen 250 kuolleeseen (Tilastokeskus 2013). Tyypillinen hoitovirhe liittyykin lääkintään: potilaalle annetaan väärä määrä lääkettä, tai annettu lääke on väärä.

Virheellisen lääkinnän torjumisessa sairaanhoitajien koulutus on avainasemassa. Valitettavasti sekä koulutuksessa että opiskelijoiden ja valmiiden hoitajien matematiikan taidoissa on puutteita. Suomalaisen tutkimuksen mukaan vain pieni osa hoitajista tai opiskelijoista pystyi laskemaan sujuvasti MCS-lääkelaskentatestissä, joka mittaa lääkkeiden annostelussa tarvittavia laskemisen perustaitoja (Grandell-Niemi ym. 2006).

Suomessa ja ulkomailla tehdyt tutkimukset näyttävät, että sekä terveydenhoitajaopiskelijoilla että ammatissa toimivilla hoitajilla on parantamista matemaattisissa taidoissa (Grandell-Niemi ym. 2003; McMullan ym. 2009; Sheriff, Wallis & Burston 2011; Wright 2006).

Lääkelaskennan oppiminen ja opettaminen on haastavaa (Johnson & Johnson 2002). Lääkelaskennan opettajien tehtävänä onkin luoda opetusmenetelmät ja mahdollistaa oppimisen ympäristöt, joissa tämä haaste parhaiten kohdataan. Tavoitteena on saavuttaa sujuva taso lääkelaskennan mekaanisessa laskemisessa ja erilaisissa lääkelaskentaongelmissa tarvittavissa loogisissa päättelyketjuissa ja ajattelussa.

3.5 Yhteenveto

Matematiikka on läsnä kaikkialla yhteiskunnassa. Suurin huomio matematiikan opetuksessa kohdistetaan pienten lasten opettamiseen sekä yliopistotasolle. Yhteiskunnan kannalta yhtä tärkeää matematiikkaa löytyy myös monista ammateista esimerkiksi terveydenhuollon piirissä.

Matematiikan osaaminen voidaan jakaa monella tavalla osa-alueisiin. Näistä ehkä useimmin käytettyjä ovat konseptuaalinen ymmärtäminen ja proseduraaliset taidot. Koulumaailma kiinnittää usein paljon huomiota nimenomaan proseduraalisiin taitoihin, joita pyritään kehittämään tekemällä paljon toistoja.

Matematiikan oppiminen rakentuu aina aiemmin opitun päälle. Tässä matematiikan opiskelu eroaa paljon monista muista taidoista. Opettajan rooli onkin tärkeä, hän luo edellytykset oppimiselle valitsemalla opetusmenetelmät ja -välineet.

Tässä tutkielmassa luodaan katsaus perinteisen matematiikan opettamiseen pohjautuen Richard Skempin (1976) esseeseen. Tästä johtuen konseptuaalisen ymmärtämisen rooli on korostunut, eikä proseduraalisiin taitoihin kiinnitetä niin paljon huomiota. Digitaalisten oppimisympäristöjen roolia tulevaisuuden matematiikan opetuksessa tarkastellaan puolestaan Devlinin (2011) hengessä.

Matematiikan opettamista sähköisten oppimisympäristöjen avulla esitellään Aalto-yliopistossa saatujen kokemusten ja tehdyn tutkimuksen pohjalta (Sangwin 2013). Tutkielman lopussa hahmotellaan, miten opettaminen sähköisten apuvälineiden avulla peilautuu Skempin alkuperäisiin ajatuksiin, jotka on laadittu jo ennen digiaikaa.

4

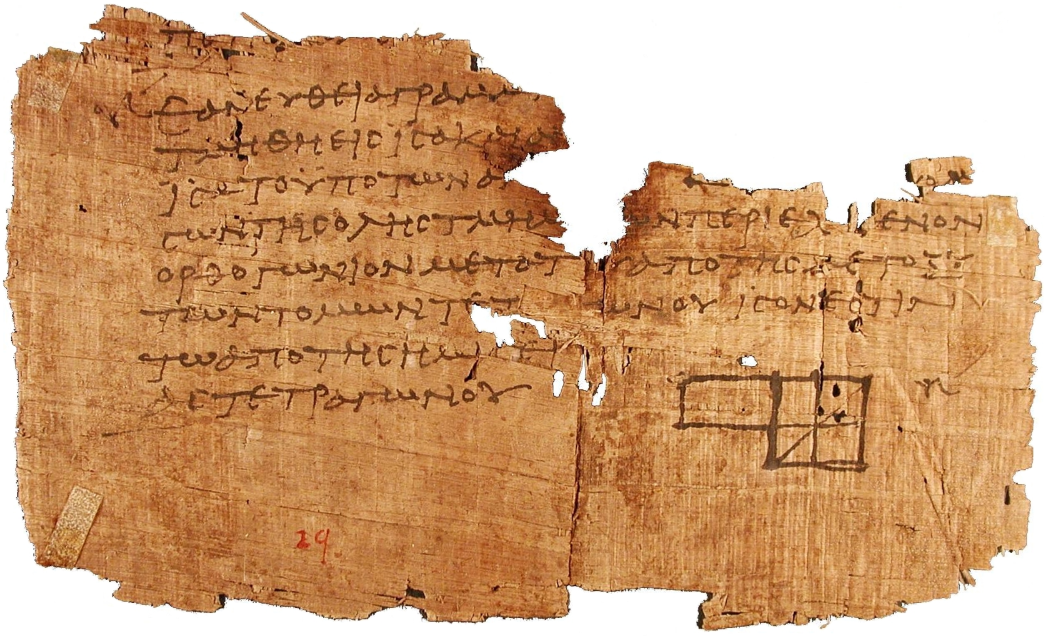
Matematiikan opettamisen välineet

4.1 Kirjan traditio matematiikan esittämisessä

Jos matematiikan oppiminen ei ole helppoa, niin ei ole sen opettaminenkaan. Tämän havainnon voi tehdä jokainen matematiikkaa opettanut luokka-asteesta ja tasosta riippumatta. Kuten edellisessä luvussa tuotiin esille, nyky-yhteiskunnassa tarvitaan matematiikan taitoja kaikkialla. Niinpä koulutuksessa tulee varoa sellaista kehitystä, jossa matematiikka ajatellaan jonkin pienen erityislahjakkaan ryhmän salatietona.

Matemaattinen tieto on perinteisesti esitetty painetuissa kirjoissa. Vaikka matemaattiset merkinnät ovat kehittyneet ja muuttuneet vuosisatojen aikana, kirjan konsepti on pysynyt hämmästyttävän samana (ks. kuvat 1 ja 2). Matemaattisen tiedon formaalin esittämisen ideaali on säilynyt muuttumattomana vuosisatoja. Kehityskaari voidaan johtaa aina varhaisimpiin säilyneisiin matemaattisiin teksteihin kuten Eukleideen Elementaan. Tosin Elementaa on kuvailtu pedagogisesta näkökulmasta jokseenkin onnettomaksi, sillä se ei sisällä johdantoja, sovelluksia, esimerkkejä tai viittauksia (Bochner 1981; Kutateladze 2006). Tiedon säilyttäjänä se on kuitenkin toiminut vuosituhansia.

Matematiikan opetuksen formalistinen filosofia näkee, että matematiikan esittäminen äärimmäisen *bourbakistisesti* on ainoa oikea tapa (ks. esim. Kutateladze 2006). Tämän näkemyksen heikkoutena on kuitenkin, että se keskittyy vain kahteen edellä esitetyistä viidestä matematiikan taidoista: proseduraalisiin taitoihin ja deduktiivisiin kykyihin. Monesti deduktiivisten kykyjen osuutta on kavennettu uusimmissa oppikirjoissa, jolloin proseduraalisten taitojen osuus kasvaa entises-

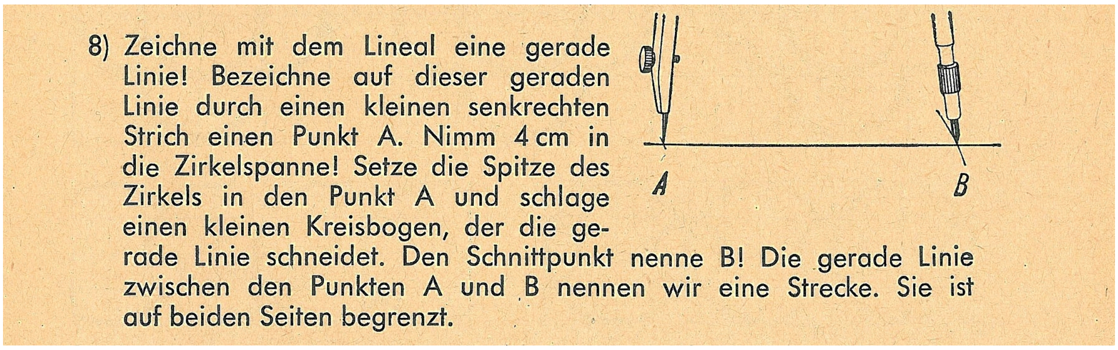


Kuva 1: Papyrus Oxyrhynchus 29 on yksi vanhimmista jäljellä olevista kopioista Eukleideen Elementasta. Papyrus kuuluu nykyisin Pennsylvanian yliopiston arkeologian ja antropologian museon kokoelmiin. Kuva W. Casselman. Alkuperäinen kuva löytyy osoitteesta http://www.math.ubc.ca/%7Ecass/Euclid/papyrus/.

tään.

Matematiikan esittämiselle ei ole vierasta, että kuvattaessa intuitiivisesti ymmärrettäviä ilmiöitä joudutaan käsittämättömään määritelmien viidakkoon. Tämä ei ole pelkästään huono asia, koska se mahdollistaa tiedon täsmällisen ja tehokkaan säilyttämisen. Viestinnällisesti se ei kuitenkaan ole paras tapa esittää tietoa, vaan sopii lähinnä matemaatikkojen keskinäiseen kommunikaatioon, jos siihenkään. Samalla tavalla insinöörien on osattava lukea ja tuottaa spesifikaatioita, protokollia ja datalehtiä, ja sama analogia voidaan tehdä lääketieteen ammattilaisille ja monille muille aloille, joissa käsitellään pitkälle kehittynyttä erityistietoa. Vaikka monimutkaisten määritelmien ja rakenteiden tulkitseminen tulee aina olemaan tärkeä osa matematiikkaa ja yksi päämäärä matemaattisten taitojen hankinnassa, ei se kuitenkaan ole optimaalinen viestinnän keino matematiikan opiskelijoille.

Jos matematiikka on kirjoitettu pelkästään kokoelmaksi sääntöjä, se edistää huonosti luonnollista tapaa ymmärtää käsitteitä opiskelijan omien havaintojen ja kokeilujen perusteella, yrityksen ja erehdyksen kautta. Tällöin opiskelija voi huo-



8) Zeichne mit dem Lineal eine gerade Linie! Bezeichne auf dieser geraden Linie durch einen kleinen senkrechten Strich einen Punkt A. Nimm 4 cm in die Zirkelspanne! Setze die Spitze des Zirkels in den Punkt A und schlage einen kleinen Kreisbogen, der die gerade Linie schneidet. Den Schnittpunkt nenne B! Die gerade Linie zwischen den Punkten A und B nennen wir eine Strecke. Sie ist auf beiden Seiten begrenzt.

Kuva 2: Janan konstruointi harpin avulla perusgeometrian oppikirjassa. Esimerkki on otettu kirjasta *Von 1 bis 1000*, Volk und Wissen Volkseigener Verlag, Berlin, 1965.

mata erilaisten alkutilanteeseen tehtyjen muutosten vaikutukset lopputuloksissa. Näin muodostuu kuva taustalla olevista käsitteistä ja siitä, miten ne suhtautuvat laajempaan matemaattiseen kokonaisuuteen. Sen sijaan on vaarassa, että opiskelijan huomio hukkuu määritelmien ja kaavojen viidakossa selviytymiseksi – ja samoin käy hänen motivaatiollensa.

Matemaattiset ideat on usein helpompi ymmärtää, kun niitä ei ole esitetty puhtaasti symbolisessa muodossa. Sen sijaan esimerkiksi spatio-visuaalisten tai motoristen esitystapojen käyttö saattaa tehdä käsitteistä helpommin ymmärrettäviä. Itse asiassa matemaatikot käyttävät usein tällaisia hyvinkin epäformaaleja keinoja puhuessaan uusista matemaattisista käsitteistä, vaikka ilmaisutapa ei olisikaan mitenkään täsmällistä tai käyttökelpoista käsitteiden todistamisen kannalta. Tämä havainto on ristiriidassa sellaisen väitteen kanssa, että formalismi olisi aina paras tapa viestiä tai opettaa matematiikkaa.

4.2 Kirjasta sähköisiin materiaaleihin

Matematiikan oppikirjat on usein laadittu siten, että ne esittävät matemaattista tietoa mahdollisimman johdonmukaisesti ilman tulkinnanvaraa, ja että niiden avulla olisi mahdollista opiskella mahdollisimman objektiivisesti. Kirjan on tarkoitettu toimivan vain tiedon välittäjänä ilman riskiä, että esimerkiksi tulkinta johtaisi tiedon muuttumiseen. Tästä saattaa olla myös haittaa pedagogisesta näkökulmasta. Jos aineisto on toteutettu tietokoneella, ongelma saattaa vielä korostua. Digitaalisten materiaalien luonne on usein tarkoituksella sellainen, että niitä voi käyttää itseopiskeluun ilman opettajan apua. Tästä seuraa luonnollisesti myös se, että opettaja ei ole läsnä silloinkaan, kun sille olisi tarvetta. Digitaaliselle oppimateriaalille onkin tältä kannalta suuremmat vaatimukset. Niinpä esimerkiksi automaattisen tarkastuksen sisältäviä oppimateriaaleja ei ole mahdollista laatia muuntamalla oppikirjoja paperista sähköiseen muotoon, vaan niiden sisältöjen ja tavan esittää asioita tulee olla erilaisia.

Painettuja kirjoja imitoivilla digitaalisilla oppimateriaaleilla on myös muita ongelmia. Perinteiset tavat esittää matematiikkaa ovat muovautuneet painotekniikan asettamien rajoitusten puitteissa. Kun tällainen materiaali muunnetaan sähköiseen muotoon, perii uusi esitystapa alkuperäiseltä medialta niin hyvät kuin huonotkin puolet. Ne toistavat helposti kaiken sen, mitä aiemmin voitiin helposti painaa paperille: kaavat, symbolit, kaksiulotteiset piirrokset ja sanalliset kuvaukset matemaattisista käsitteistä. Uuden tekniikan mahdollistamat dynaamiset sisällöt puuttuvat, ja asioiden suhteet on esitetty kaavojen tai algoritmien avulla, joskus täydennettynä esimerkein ja kuvin.

Viimeisten kymmenen vuoden aikana Keith Devlin Stanfordin yliopistosta on kehittänyt radikaalisti erilaisia näkemyksiä matematiikan esittämisestä nykyaikaisilla tietokoneilla ja oppimisympäristöissä. Yksi Devlinin ajatuksista on käyttää pelejä matemaattisten konseptien ja ideoiden esittämiseen. Hänen tutkimuksensa tavoitteena on ollut ymmärtää, miten uudet teknologiat voivat vaikuttaa matemaattisen osaamisen kehitykseen yleensä ja etenkin matematiikan oppimiseen ja opetukseen (Devlin 2011). Yksi Devlinin keskeisiä havaintoja on edellä esitetty, että perinteisessä matematiikan opetuksessa näkyy pitkä perinne, jossa käytetään välineinä painettuja kirjoja, symboleja, kaavoja ja määritelmiä. Vaikka niiden käyttö on toisinaan perusteltua, ei tätä tapaa esittää matematiikkaa kyseenalaisteta, vaan traditiota jatketaan myös sellaisissa tilanteissa, joissa nämä keinot eivät ole parhaita mahdollisia.

Dynaamisen sisällön esittäminen ei ole ongelma nykyisin käytössä olevalla tietotekniikalla. Digitaalisen median luomat mahdollisuudet antavat hyvät lähtökohdat kehittää oppimateriaaleja, jotka ovat intuitiivisia ja informatiivisia, ja joiden avulla saadaan tuotua oppimiseen muitakin matemaattisen tiedon esittämisen muotoja, kuin mihin on perinteisessä opetuksessa totuttu. Tämän lisäksi digitaalinen sisältö voi myös mahdollistaa älykkään vuorovaikutuksen opiskelijan ja oppimateriaalin välillä. Esimerkkinä tästä on automaattisesti tarkastettavat tehtävät, jotka kykenevät analysoimaan opiskelijan suoritusta luokittelemalla ja diagnosoimalla vastauksia, ja käyttämään tätä tietoa esimerkiksi henkilökohtaisen palautteen antamiseen tai opiskelijan opiskelusuorituksen ohjaamiseen. On todennäköistä, että tulevaisuuden oppimateriaalit näyttävät täysin muulta kuin kirjalta, ja ovat enemmänkin viihdetuotteen luonteisia kuten pelejä. Devlinin (2008) mukaan seuraava vallankumous matematiikassa ei koske niinkään sisältöjä, vaan tapoja ja keinoja esittää matemaattista sisältöä.

4.3 Yhteenveto

Matematiikkaa on vuosituhansia esitetty painetun kirjan muodossa. Kirjantekotaito on muokannut matemaattisia merkintöjä ja jopa sitä, millaista tietoa mielletään matematiikaksi. Matematiikan esittämisessä on usein vallalla formaaliuden ihanne, jolloin matematiikka nähdään kokoelmana täsmällisiä sääntöjä ja lauseita. Tällä voidaan nähdä yhteyksiä proseduraalisten taitojen korostumiseen sekä behavioristiseen opetukseen. Vaikka formaalilla esityksellä on oma sijansa, se ei ole optimaalinen keino viestittää matemaattisia käsitteitä opetustilanteessa.

Digitaalisten oppimateriaalien ongelmana on usein, että ne on tuotettu imitoimaan painettuja materiaaleja. Tällä on usein kahdenlaisia huonoja seurauksia. Toisaalta menetetään painetun kirjan hyvät puolet, mutta ei myöskään hyödynnetä tietoteknisen ympäristön luomia mahdollisuuksia matemaattisen tiedon esittämiseksi.

Sähköiset oppimisympäristöt mahdollistavat oppimateriaaleja, joita on ollut mahdoton tuottaa painetussa muodossa. Niiden avulla on mahdollista luoda vuorovaikutteisia aineistoja, jotka reagoivat opiskelijan toimintaan. Myös tehtävien tarkastaminen ja henkilökohtaisen palautteen antaminen on mahdollista sähköisillä oppimisympäristöillä. Tätä käsitellään tarkemmin seuraavassa luvussa.

5

Sähköiset oppimisympäristöt

5.1 Matematiikan esittämisen haaste

Viimeisen 40 vuoden aikana mikroelektroniikka on mullistanut maailman. Teknologian kehitys on luonut kokonaan uusia teollisuudenaloja etenkin tieto- ja viestintätekniikan piiriin, kun taas monet vanhat ammatit ovat tulleet tarpeettomiksi ja kadonneet. Muutosten voimakkuus on ollut niin suurta, että ne ovat kaataneet hallituksia ja järkyttäneet maailmantaloutta. Tietotekniikan kehitys on muokannut myös sitä perustaa, jolle teknologia on rakennettu, eli matematiikkaa itsessään. Sen opettaminen ja oppiminen on kokenut suuria murroksia.

Sähköisten oppimateriaalien hyöty voidaan nähdä tulevaisuudessa paljon laajemmin kuin edellä mainitut yksilöllisen harjoittelun mahdollistuminen sekä erilaisten rutiinien automatisoituminen. Yksinkertaisten harjoitusten lisäksi niiden avulla on mahdollista toteuttaa monimutkaisia tehtäviä, joilla on mahdollisesti monta vastausta. Erilaisten avointen tehtävien toteuttaminen tulee näin mahdolliseksi. Tämä auttaa opiskelijaa harjoittamaan kontekstuaalista tietoa, mikä on parannus behaviorististiseen ja proseduraaliseen opetukseen ohjaavien materiaalien käyttämiselle.

Sähköiset oppimisympäristöt ovat sopivia myös proseduraalisten taitojen opettamiseen, sillä niiden avulla on mahdollista tehdä toistoja tehokkaasti. Myös tehtävien tarkastaminen voidaan automatisoida. Sähköiset oppimisympäristöt ovatkin syrjäyttämässä kirjat ja muun painetun median opiskelun välineenä. Tähän liittyy myös matemaattisen tiedon esittämisen muuttuminen ja kehittyminen. Ovathan matematiikan merkinnät ja muu esittäminen voimakkaasti yhteydessä kirjapainotaitoon ja kirjaan tiedon esittämisen välineenä, kuten edellisessä luvussa todettiin.

Yleisesti ottaen sähköisten oppimisympäristöjen suunnittelijat ovat tietoisia edellä esitetyistä matematiikan esittämisen haasteista ja mahdollisuuksista. On myös tunnettua, että vuorovaikutus opiskelijan ja verkko-oppimisympäristön välillä johtaa (ainakin) kahteen erilaiseen yhteentörmäykseen, joista seuraavassa.

Ensiksi opiskelija voi kohdata vaikeuksia tuottaa matemaattista sisältöä oppimisympäristön käyttämällä syntaksilla. Matematiikan formaalia kieltä ei ole helppoa tuottaa tietokoneen näppäimistöltä. Useimmiten käytössä on lineaarinen syöte, jolloin koneelle annetaan vastauksena merkkijono (Sangwin & Ramsden 2007). Tällöin on käytössä oppimisympäristölle tyypillinen syntaksi, jolla matemaattiset merkinnät on muunnettu peräkkäin kirjoitettaviksi näppäimistön merkeiksi. Pahimmassa tapauksessa opiskelija voi joutua korjaamaan parittomia sulkumerkkejä ja puuttuvia tai vaadittavia kertomerkkejä muuten oikeaan vastaukseensa monta kertaa, ennen kuin syntaktisesti oikea vastaus on onnistuneesti annettu. Opiskelijat eivät yleensä ole tietokoneiden asiantuntijoita, joten moinen jumppa vain lisää turhautumista. Oppimisympäristö voi näin lisätä opiskelijan stressiä ja kääntää oppimismahdollisuuden merkityksettömäksi taisteluksi tietokoneen kanssa.

Toiseksi matemaattisen tiedon esittäminen tietokoneen näytöllä on jossain määrin ongelmallista. Hankaluus kumpuaa osittain oppikirjakeskeisestä näkökulmasta matemaattiseen oppimateriaaliin. Myös painetun materiaalin suora kopioiminen digitaaliseksi tuo ongelmia, mukana seuraa usein myös oppimateriaalikeskeisiä opetusmenetelmiä. Onneksi kuitenkin puhtaasti teknisistä rajoitteista ollaan pääsemässä eroon. Esimerkkinä tästä on MathJax-matematiikkamoottori (MathJax Consortium 2015).

Opettajan ja oppimateriaalin suunnittelijan näkökulmasta oppimisympäristön tarjoamat mahdollisuudet ja rajoitukset vaikuttavat niihin pedagogisiin ja teknisiin ratkaisuihin, joilla opiskelijan oppimisprosesseja aktivoidaan (Majander & Rasila 2011; Rasila, Havola, Majander & Malinen 2010). Oppimisympäristön käyttäjäkokemuksen olennainen osa on matemaattisen sisällön esittäminen. Sen pitäisi olla helposti ymmärrettävää myös ilman opettajan apua. Vuorovaikutus opiskelijan ja tietokoneen välillä tulee olla helppoa ja luonnollista (Tiitu & Rasila 2014).

Materiaalin kehittäjän on melko vaikeaa ennakoida, mitkä komponentit tai toiminnot lopulta aiheuttavat eniten negatiivisia käyttökokemuksia. Näitä kysymyksiä voidaan ratkaista nykyaikaisilla palvelumuotoilun keinoilla. Materiaalien kehitystyön tulee olla käyttäjälähtöistä jokaisessa vaiheessa (esim. Stickdorn & Schneider 2011). Lisäksi jotkut materiaalin ja järjestelmän ongelmakohdat voidaan havaita myöhemmin epäsuorasti perustuen opiskelijoiden antamiin harjoitusvastauksiin, joita oppimisympäristö kerää. Käyttäjäpalaute tulee kerätä talteen ja analysoida. Oppimisympäristö ja oppimateriaalit käyvät tyypillisesti läpi useita korjauskierroksia, jolloin tämä palaute voidaan huomioida.

Laadukkaiden sähköisten oppimateriaalien tuottaminen vaatii, että tuotantoryhmässä on sekä pedagogista että teknologista osaamista. Oppimateriaalin tuottaminen ei läheskään aina ole yksinkertainen prosessi. Jo teknologisten kysymysten ratkaiseminen esimerkiksi satunnaistetussa automaattisesti tarkastettavassa tehtävässä voi hyvinkin vaatia kymmeniä tunteja kehitystyötä.

Verkko-oppimisympäristöt tarjoavat mahdollisuuden tuottaa opetusta, jota ei helposti tai lainkaan olisi mahdollista toteuttaa ilman niitä. Perinteisillä opetusvälineillä on monia etuja. Niiden käyttö on hioutunut jopa vuosisatojen ajan, ja pedagogiikat ja opetus on muovautunut perinteisesti niiden ympärille. Onkin hyväksyttävä, että jossain tilanteessa vanha keino toimii paremmin kuin pussillinen uusia. Mutta tilanteissa, joissa sähköinen oppimismateriaali mahdollistaa mielenkiintoisen ja opiskelijan oppimista motivoivan ja stimuloivan oppimistilanteen, tulee niitä käyttää ennakkoluulottomasti. Usein tällaisessa oppimistilanteessa on läsnä jotain uutta ja jotain vanhaa. Opetusmenetelmiä ja -välineitä tulisikin käyttää toistensa tukena ennakkoluulottomasti ja antaa pienelle kokeilullekin sijansa.

5.2 Automaattisesta tarkastamisesta oppimisanalytiikkaan

Miten automaattisen arvioinnin järjestelmää voidaan käyttää kehittämään ja arvioimaan edistyneempiä oppimiseen liittyviä käsitteitä kuten konseptuaalista ymmärtämistä? Vaikka toisto ja harjoittelu ovat keskeisessä asemassa behavioristisessa oppimiskäsityksessä, on niillä sijansa myös konstruktiivisessa opetuksessa. Skempin (1976) mukaan uusi matemaattinen käsite tulee käydä läpi harjoitellen monta kertaa. On tärkeää, että perustalla olevat kognitiiviset prosessit automatisoituvat, jolloin opiskelijalla vapautuu kapasiteettia omaksua uusia käsitteitä syvemmin. Lopulta matemaattiset käsitteet irtautuvat niitä esittävistä symboleista ja esitystavoista, ja matemaattisen tiedon manipulointi tapahtuu symbolisesti ilman suoraa yhteyttä käsitteisiin merkintöjen takana. Tämä kaikki vaatii yleensä paljon harjoittelua. Vaikka edellä esitetyissä Skempin huomioissa ei ole mitään tietokoneavusteiseen opetukseen liittyvää sinänsä, voidaan sen avulla toteutettuja oppimateriaaleja käyttää helpottamaan ja mahdollistamaan edellä kuvatun kaltaista oppimisprosessia.

Miten on mahdollista tietää, paljonko harjoitusta matemaattisen käsitteen oppiminen vaatii? Tai miten voidaan tietää, onko opiskelijan tiedoissa ja taidoissa tapahtunut kehitystä? Yksi mahdollisuus on analysoida suuri määrä dataa, joka on tyypillisesti syntynyt sähköisen oppimisympäristön avulla opiskelemisesta. Esimerkki tällaisesta analyysista löytyy Linnoisen (2013) pro gradu -tutkielmasta, jossa Granger-kausaalisuuden avulla tutkittiin, johtaako kova työskentely hyvään menestykseen myöhemmissä matematiikan opinnoissa. Tässä tutkimuksessa tutkittiin ennemminkin konseptuaalista ymmärtämistä kuin proseduraalisia taitoja, koska tarkastellut opiskelijat oppivat täysin uusia asioita. Kykyä oppia näitä uusia asioita arvioitiin opiskelijan taustoja vastaan.

Skempin (1976; 1987) mukaan tärkein päämäärä matematiikan opetuksessa on opiskelijan kognitiivisten taitojen kehittäminen. Uuden tiedon esittäminen ja konstruoiminen ovat vain keinoja tämän päämäärän saavuttamiseksi. Opiskelijan motivaatio ja asenteet, käytettävissä oleva aika ja oppimisen järjestelyt kuten ryhmät, tilat ja aikataulutus saattavat muodostua esteiksi päämäärän saavuttamisessa. Opiskelija saattaa haluta vakuuttua oppimiensa asioiden tärkeydestä ja hyödystä nopeasti, ja pitkäjänteinen työ ja opeteltavien asioiden vaikeus turhauttavat häntä. Tästä syystä on hyödyllisempää antaa opiskelijoille suhteellisen yksinkertaisia tehtäviä, jolloin useimmat heistä motivoituvat, kuin haastavia tehtäviä, jotka olisivat mielenkiintoisia vain pienen joukon mielestä.

Opettajalle voi olla hyvin vaikeaa arvioida yleisellä tasolla oppimiseen tarvittavaa aikaa ja työmäärää, tai että mikä on opiskelijoille liian haastavaa. Tämä johtuu siitä, että yleensä opiskelijoiden välillä on hyvin suuria eroja, ja heidän henkilökohtaiset tarpeensa on mahdotonta ottaa huomioon yhdellä kertaa esimerkiksi luokkahuoneessa tapahtuvassa opetustilanteessa. Opetuksen alussa tilannetta voidaan yrittää kartoittaa erilaisilla lähtötasotesteillä, mutta opettajan kokemus on yleensä ainoa tekijä, jonka perusteella erilaiset opettamisen valinnat tehdään.

Sähköiset oppimisympäristöt voivat periaatteessa tuottaa personoituja harjoitustehtäviä, jotka ovat oikean tasoisia kullekin opiskelijalle. Oppimisympäristön avulla voidaan myös esittää täsmälliset oppimateriaalit kunkin opiskelijan tarpeisiin. Oppimisympäristö voi tallentaa suuren määrän tietoa opiskelijoiden vastauksista, jolloin siitä on mahdollista löytää yhdenmukaisuuksia tiedonlouhinnan keinoin. Oppimisympäristö tai siihen liitetty ohjelmisto voi suorittaa esimerkiksi virheluokittelua, jonka perusteella opiskelijoille voidaan valita satunnaistettu ja henkilökohtainen versio oppimateriaaleista. Vaikka tämä on kunnianhimoinen tavoite nykyisille oppimisympäristöille, on jo nähtävissä, että sähköisten oppimisympäristöjen kehitys tulee etenemään tähän suuntaan.

5.3 Yhteenveto

Sähköisillä oppimisympäristöillä voidaan automatisoida matematiikan laskuharjoittelua, ja tuottaa myös avoimen tyylisiä tehtäviä, jotka mahdollistavat konseptuaalisten taitojen harjoittamisen. Sähköisten oppimateriaalien tuottamisessa on kuitenkin haasteita, jotka liittyvät esimerkiksi matematiikan syöttämiseen päätelaitteella ja esittämiseen näytöllä. Oppimateriaalien kehittämisen tuleekin olla käyttäjälähtöistä, yleiset palvelumuotoilun menetelmät tuottavat myös hyviä matematiikan oppimisympäristöjä.

Tietokoneavusteisen opetuksen ja yleensä matematiikan esittämisen tietokoneella lisääntyessä myös itse matematiikka muuttuu. Seuraava suuri vallankumous matematiikassa saattaa olla seurausta tästä: matemaattiset sisällöt eivät itsessään muutu, mutta matemaattisen tiedon esitystapojen muuttuessa jopa käsityksemme matematiikan piiriin kuuluvasta tiedosta voi muuttua.

Sähköisten oppimisympäristöjen keräämää tietoa opiskelijoiden suorituksista voidaan käyttää oppimisen analysointiin. Esimerkiksi virheluokittelua voidaan automatisoida, jolloin on mahdollista käsitellä suuri määrä vastauksia ja saada tilastollisesti merkittäviä tuloksia. Tätä tietoa voidaan käyttää edelleen oppimisympäristön toiminnan säätämiseen, jolloin opiskelijalle voidaan tarjota personoituja tehtäviä, jotka ovat hänen oppimisensa vaiheeseen sopivia.

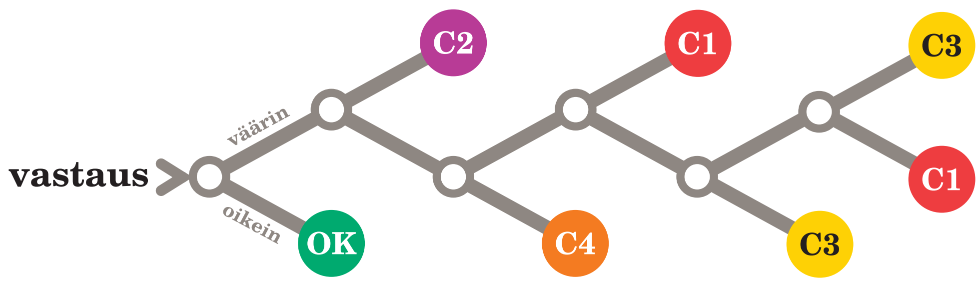
6

Matematiikan tehtävien automaattinen tarkastaminen

6.1 Oppimisympäristö Stack

Stack on GPL-lisensoitu avoimen lähdekoodin matematiikan oppimisympäristö, jonka kehittämisen aloitti Chris Sangwin Birminghamin yliopistossa. Stack on selainkäyttöinen verkko-oppimisympäristö, joka mahdollistaa automaattisen tarkastuksen lisäksi tehtävien satunnaistamisen. Stackin ytimessä on symbolisen laskennan ohjelmisto Maxima, jonka avulla tehtävien logiikka laaditaan ja vastaukset tarkistetaan. Myös Maxima on avoimen lähdekoodin ohjelmisto. Maxima pystyy aitoon symboliseen laskentaan, joten sen avulla on mahdollista tehdä hyvin monipuolisia tehtäviä, jopa hienostunutta simulointia ja mallinnusta. Lisäksi Stackillä on mahdollista tehdä avoimen tyylisiä tehtäviä, joilla on monta ratkaisua.

Stackin käyttö mahdollistaa monenlaisten pedagogisten menetelmien käytön. Stackin käyttöliittymä on nykyisessä versiossa Moodle-oppimisympäristö, jonka yksi tehtävätyyppi Stack-tehtävä on. Moodle on tietyistä käytettävyyteen liittyvistä rajoituksistaan huolimatta laajassa käytössä tällä hetkellä, joten valmius Stack-tehtävien käyttöönotolle on monessa koulussa olemassa entuudestaan tutun oppimisympäristön muodossa.



Kuva 3: Idelisoitu kuva Stackin tarkastuspuusta. Opiskelijan antamaa vastausta verrataan ensin opettajan antamaan mallivastaukseen. Mikäli vastaus on väärin, voidaan pelkän väärin-ilmoituksen sijaan vastausta prosessoida eteenpäin. Sitä voidaan verrata piirteisiin, jotka paljastavat virheluokittelumallin luokkien tyyppisiä ominaisuuksia. Virheluokittelumalli voi perustua esimerkiksi lääkelaskennan opetuksen 4 Cs -malliin, josta enemmän luvussa 7. Tätä analyysia voidaan käyttää palautteen luomiseksi sekä itse ympäristön toiminnan säätämiseen opiskelijan tarpeita vastaavaksi.

6.2 Automaattinen tarkastaminen toteutetaan ohjelmoimalla

Automaattinen tarkastus tehdään ohjelmoimalla tehtäviin logiikkaa, jolla arvioidaan opiskelijan antaman vastauksen oikeellisuutta ja tehdään tarvittaessa pisteytys. Matemaattisten ongelmien muuntaminen oppimisympäristön ohjelmaksi ei vaadi ainoastaan ohjelmointitaitoa, vaan myös käsitystä opiskelijoiden tekemistä virheistä sekä ajattelu- ja opiskeluprosesseista. Näiden asioiden hallitseminen on kaikkea muuta kuin helppoa. Laadukkaiden sähköisten opetusmateriaalien laatiminen kestää usein monta kehityskierrosta, joiden aikana materiaalia hiotaan paremmaksi. Vaikka kehityskierroksia on useita, oppimateriaalien tekijälle on usein hyvin vaikea saavuttaa sitä monimutkaisuuden tasoa, jolla opiskelijat virheitä tekevät.

Automaattisesti tarkastettavien tehtävien järjestelmä kerää paljon tietoa opiskelijan suorituksesta. Jos tämä tieto saadaan talteen, antaa se paljon mahdollisuuksia opiskelijan suorituksen analysoimiseksi. Keskeinen osa tätä tietoa ovat opiskelijan vastaukset, joista järjestelmä tyypillisesti pyrkii jaottelemaan oikeat vastaukset vääristä. Tämäkään perustehtävä ei läheskään aina ole yksinkertainen. Väärien vastausten luokitteleminen on pelkkää tarkastamista monimutkaisempi tehtävä, koska tähän liittyy jopa opiskelijan kognitiivinen ja emotionaalinen tila hänen ratkaistessaan tehtävää.

Tarkastusprosessi on yksinkertaisesti sarja loogisia testejä, joilla annettua vastausta testataan kuten kuvassa 3. Ensin kokeillaan vastaavuutta malliratkaisun kanssa, ja tämän jälkeen niin oikeaa kuin vääräákin vastausta on mahdollista verrata lausekkeisiin, joilla pyritään selvittämään lisäpiirteitä vastauksesta. Nämä lisäpiirteet voivat olla virheluokittelu, mutta myös esimerkiksi kohdennettua palautetta varten. On kuitenkin ilmeistä, että opiskelijan vastausdatassa on myös tietoa esimerkiksi opiskelijasta itsestään ja hänen opiskelutyylistään. Näitä tietoja on mahdollista tutkia tilastollisen analyysin ja tiedonlouhinnan keinoin. Tässä prosessissa käsiteltävät tietomäärät ovat niin suuria, että analyysia ei olisi mahdollista tehdä käsin, vaan se on tehtävä jollain automaattisella järjestelmällä.

6.3 Automaattisen tarkastamisen edut

Tässä tutkielmassa käytetään jatkossa Stackiä esimerkkinä satunnaistetut tehtävät ja automaattisen tarkastuksen mahdollistavasta oppimisympäristöstä. Stackin käyttämisestä on pitkä kokemus Aalto-yliopiston matematiikan ja systeemianalyysin laitoksella tekniikan alojen matematiikan yliopisto-opetuksessa. Samalla hahmotellaan tietokoneavusteisen opetuksen nykytilaa ja esitetään, millaiseen suuntaan sähköiset oppimisympäristöt ovat kehittymässä koulu- ja yliopistomatematiikan opettamisessa.

Stackin kaltaisen automaattisen tarkastuksen mahdollistavan oppimisympäristön käytöllä on seuraavia etuja.

Joustavuus Verrattuna perinteiseen lähiopetukseen, Stack tarjoaa enemmän joustavuutta opiskeluun ja säästää aikaa niin opiskelijoilta kuin opettajilta (Rasila, Harjula & Zenger 2007). Opettajat voivat käyttää säästyneen ajan opiskelijoiden ohjaukseen.

Personointi Stack mahdollistaa personoitujen tehtävien laatimisen. Tällöin opiskelijat saavat tehtävästä oman versionsa. Tämä mahdollistaa opiskelijoille ryhmätyön, jossa opiskelijat voivat keskustella tehtävän ratkaisemisesta jättäen kuitenkin jokaiselle lopullisen ratkaisemisen ilon. Yhdessä ratkaiseminen rohkaisee vertaisoppimiseen ja matemaattisista käsitteistä keskusteluun (Rasila ym. 2010).

Palaute Palaute ohjaa opiskelijan oppimisprosessia. Stack-tehtävien antama välitön palaute on opiskelijoiden keskuudessa suosittu ominaisuus (Sangwin 2013). Stack-tehtävien palaute on mahdollista tehdä niin yksityiskohtaiseksi, että opiskelija voi korjata virheellistä käsitteenmuodostustaan (Rasila ym. 2010). Hyvän palautteen rakentaminen on kuitenkin haastava tehtävä ja vaatii opiskelijan tehtävänratkaisuprosessien ymmärtämistä.

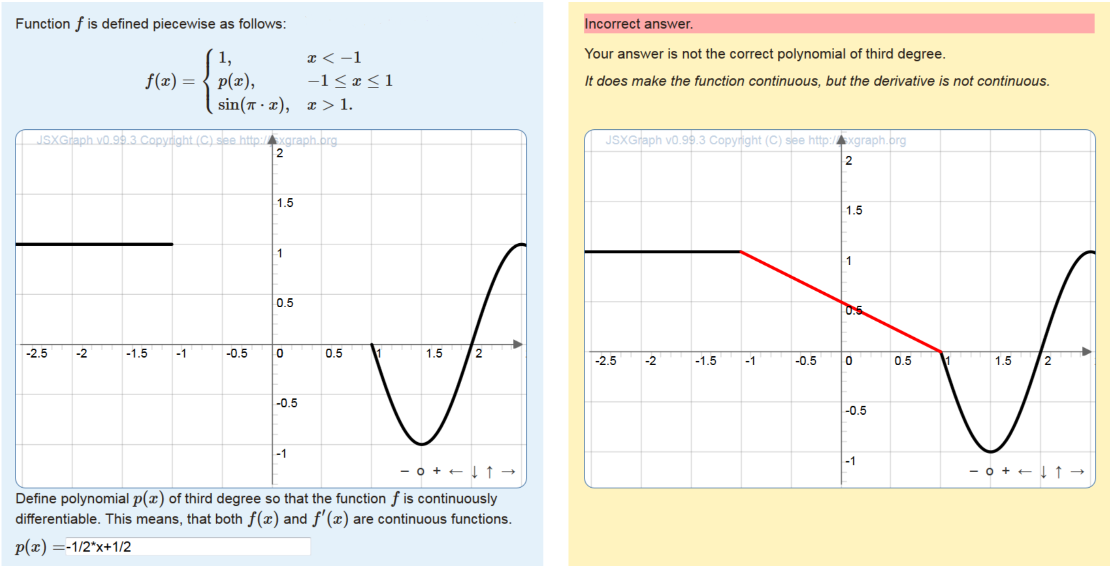
Jatkuva arviointi Stackin avulla arvostelu voidaan laatia jatkuvaksi koko kurssin suorituksia peilaavaksi. Tämä voi lisätä arvioinnin luotettavuutta ja reiluutta. Lisäksi jatkuvaa arviointia voidaan käyttää jakamaan työmäärää tasaisesti koko kurssin ajaksi (Majander & Rasila 2011).

Dynaaminen sisältö Stack-tehtävät voivat sisältää dynaamista sisältöä, esimerkiksi kuvia tai interaktiivisia visualisointeja. Näitä voi olla niin tehtävänannossa kuin palautteessa. Visuaalisen palautteen avulla käsiteltävät matematiikan käsitteet on helppo nostaa esille. Esimerkki tästä on kuvassa 4, jossa jatkuvan derivoituvuuden käsite on tuotu esille kuvan avulla. Opiskelijan oli pyydetty jatkaa annettua funktiota siten, että siitä tulee derivoituva koko annetussa alueessa. Visuaalinen palaute näyttää opiskelijan antaman vastauksen ja derivaatan epäjatkuvuus voidaan nähdä välin $[-1,1]$ päätepisteissä. Kuvan lisäksi on annettu sanallinen selitys.

Pelillisyys Stackiä voidaan käyttää myös ohjelmointiympäristönä yksinkertaisten oppimispelien tekemiseen. Niiden avulla oppimisesta on mahdollista tehdä motivoivaa ja interaktiivista. Esimerkiksi kuvassa 5 opiskelijaa pyydetään arvioimaan erilaisia riskiskenaarioita yritystoiminnalle epävarmoissa olosuhteissa toimimiseksi. Tehtävän viimeinen osa on muodostettu sen perusteella, mitä opiskelija on alussa valinnut. Näin tehtävän sisälle muodostuu tarina, jonka etenemiseen opiskelija vastauksillaan vaikuttaa.

Oppimisanalytiikka Opiskelijoiden määrällistä ja laadullista edistymistä voidaan analysoida tutkimalla heidän tekemiään virheitä. Stackin tietokantaan tallentuu tiedot opiskelijan tekemistä tarkistuksista ja vastauksista. Tämä voidaan tehdä pienelle tehtävämäärälle käsin tai automaattisesti käyttäen ennalta määriteltyä virheluokittelumallia. Mallin tulee tuottaa luotettavia tuloksia ja olla validoitu käyttäen tilastollisia menetelmiä.

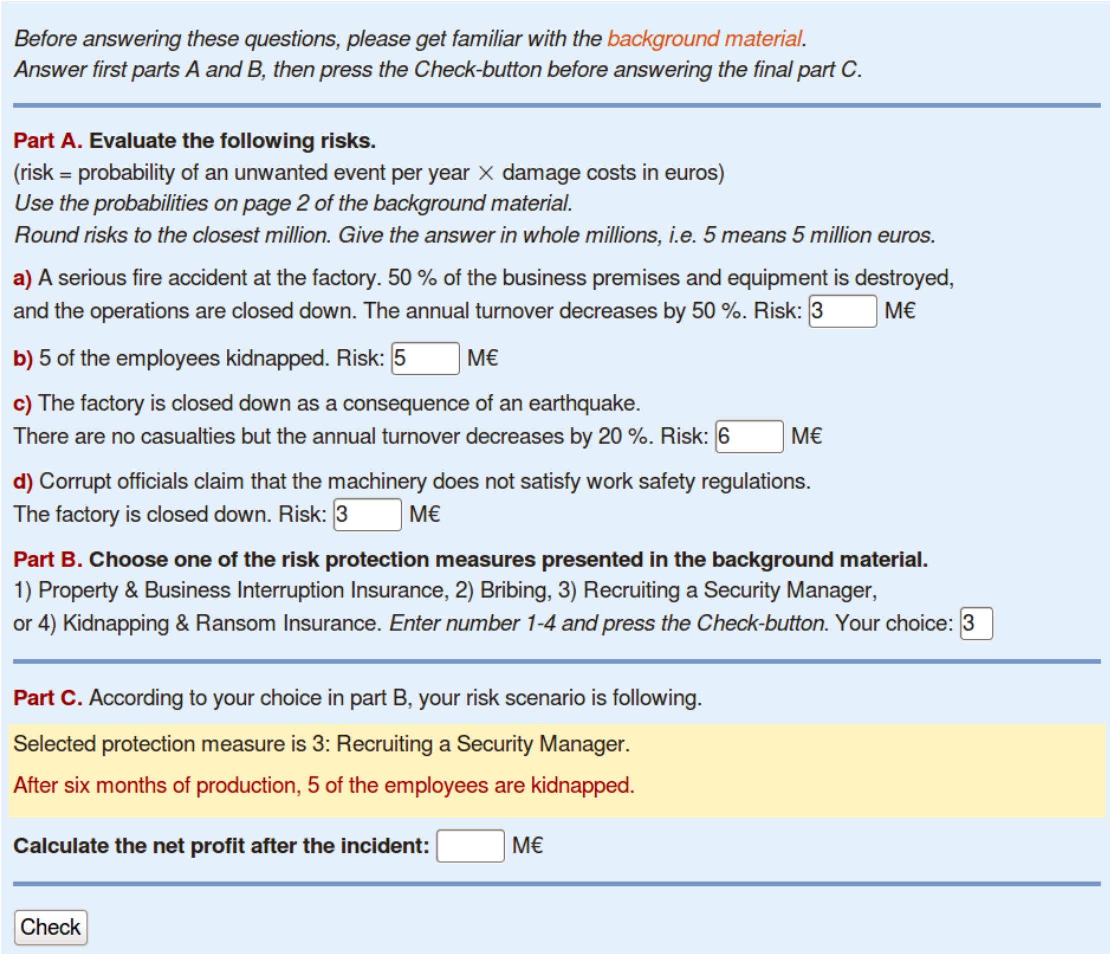
Mekaanista laskuharjoitusta tuottavien tehtävien lisäksi Stackillä voidaan toteuttaa yksinkertaisia pelin kaltaisia tehtäviä. Nämä tehtävät eivät kuitenkaan



Kuva 4: Vasemmalla Stack-tehtävässä dynaamisesti tuotettu kuva. Oikealla palautteena annettu opiskelijan vastauksen visualisointi.

nykyisen Stackin rajoitusten vuoksi pysty kommunikoimaan keskenään. Tehtävien ainut vuorovaikutus ulospäin on suorituksen tuottama pistemäärä. Nykyiset Stackin pelilliset tehtävät tuleekin käsittää loogisena sarjana harjoituksia, joiden suoritus etenee etukäteen laadittua tarinaa tai juonta seuraten.

Stackiä kehitetään edelleen mahdollistamaan myös sellaisia tehtäviä, joilla on keskinäistä vuorovaikutusta. Käytännössä tämä tarkoittaa, että järjestelmään rakennetaan ylätason muuttujat, jotka näkyvät myös yksittäisen tehtävän ulkopuolelle. Tällaisten muuttujien arvojen voidaan sanoa muodostavan tehtävien suorittamisen tilan. Ylätason muuttujien avulla on mahdollista rakentaa tehtäväsarjoja, joissa opiskelijan saamat tehtävät muodostavat tarinallisen kokonaisuuden. Tarinalla voidaan tässä käsittää peräkkäisiä tapahtumia, jotka seuraavat toisiaan riippuen opiskelijan tekemistä valinnoista eli vastauksista. Toisaalta tarina voi olla myös pedagoginen, jolloin opiskelijan käsitteenmuodostusta ohjataan vaihe vaiheelta antamalla sopivia osatehtäviä suoritettavaksi. Tietoa opiskelijan suorituksista voidaan käyttää joko tehtävien valinnassa tai tehtävien muokkaamisessa oppimistilanteeseen sopiviksi. Näin Stackin avulla saadaan tuotettua hyvin yksilöllistä opetusta.



Before answering these questions, please get familiar with the background material.
Answer first parts A and B, then press the Check-button before answering the final part C.

Part A. Evaluate the following risks.
(risk = probability of an unwanted event per year × damage costs in euros)
Use the probabilities on page 2 of the background material.
Round risks to the closest million. Give the answer in whole millions, i.e. 5 means 5 million euros.

a) A serious fire accident at the factory. 50 % of the business premises and equipment is destroyed, and the operations are closed down. The annual turnover decreases by 50 %. Risk: 3 M€

b) 5 of the employees kidnapped. Risk: 5 M€

c) The factory is closed down as a consequence of an earthquake.
There are no casualties but the annual turnover decreases by 20 %. Risk: 6 M€

d) Corrupt officials claim that the machinery does not satisfy work safety regulations.
The factory is closed down. Risk: 3 M€

Part B. Choose one of the risk protection measures presented in the background material.
1) Property & Business Interruption Insurance, 2) Bribing, 3) Recruiting a Security Manager,
or 4) Kidnapping & Ransom Insurance. *Enter number 1-4 and press the Check-button.* Your choice: 3

Part C. According to your choice in part B, your risk scenario is following.

Selected protection measure is 3: Recruiting a Security Manager.

After six months of production, 5 of the employees are kidnapped.

Calculate the net profit after the incident: M€

Check

Kuva 5: Pelin kaltainen Stack-tehtävä on jaettu kolmeen osaan A, B ja C. Viimeisen osan skenaario on tuotettu sen mukaan, mitä opiskelija on vastannut aiemmissa osissa.

6.4 Stack Aallon matematiikan opetuksessa

Kiinnostus matematiikan tietokoneavusteiseen opetukseen heräsi Teknillisessä korkeakoulussa (TKK, nykyisin osa Aalto-yliopistoa) Simo K. Kivelän MatTa-projektissa, joka käynnistyi 1993 (Kivelä & Spåra 2001). Projektin taustalla oli jo 1980-luvulla alkanut matemaattisten laskentaohjelmien kuten Matlab, Mathematica ja Maple käyttö matematiikan peruskursseilla.

Projektissa perehdyttiin erilaisiin menetelmiin, joilla olisi mahdollista tuottaa interaktiivisia oppimateriaaleja. Aikaan liittyi www:n tuleminen, ja *tehtävät* *nettiin* -ajattelu oli muodikasta. Kehitys oli teknologiapainotteista, koska tuolloin oli paljon teknisiä rajoitteita. Projektin tuotoksia ei otettu kovin laajaan käyttöön sen ulkopuolella.

Vuonna 2006 TKK:ssa arveltiin, että tehtävien automaattinen tarkastaminen on seuraava askel sähköisten oppimisympäristöjen kehityksessä. Aluksi oltiin kiinnostuneita käyttämään Mapleen pohjautuvia ohjelmistoja, mutta tämä todettiin epätyydyttäväksi. Ongelmaksi muodostui toimintojen muokkaaminen, joka oli välttämätöntä oman kehitystyön tekemiseksi. Erilaisten kokeilujen jälkeen ohjelmistoksi valikoitui Stack, jossa oli myös paljon puutteita, mutta se oli avoin ja muokattavissa.

Ensimmäinen kokeilu Stackin käytöstä oli syksyllä 2006 Matematiikan peruskurssilla KP 3 (Rasila 2008). Kurssin aiheena oli mm. kompleksianalyysi, Fourier-sarjat ja Laplace-muunnos. Opiskelijat olivat toisen vuosikurssin koneteekkareita. Stackiin kehitettiin tuolloin mm. Haka-autentikaatio ja LaTeX-integraatio, jonka avulla luotiin kuvia, joilla esitettiin matemaattisia kaavoja selainriippumattomasti. Aluksi tehtävät eivät olleet satunnaistettuja. Kokeilun jälkeen kerättiin opiskelijapalautetta, josta selvisi monia järjestelmän puutteita esimerkiksi käytettävyyden osalta. Yleisellä tasolla palaute oli kuitenkin positiivista, ja kokeilun lopputulosten perusteella Stackin käyttöä päätettiin jatkaa (ks. Harjula 2008).

Samaan aikaan Stackin kehitystyö eteni Britanniassa. Sangwinin kehitystiimi oli aloittanut Stack 2:n kehittämisen. Tämä synnytti ongelmia, sillä kehitystyö ei ollut yhteensopiva TKK:n kehitystyön kanssa. Tilannetta ei helpottanut, että asian selvitessä molemmat kehityshaarat olivat jo pitkällä. Kehitys päätettiin yhdistää, mutta tämä kesti monta vuotta. Versio 2 oli Stackin ensimmäinen Moodle-integraatio.

Aallon kehitystyö on vaikuttanut paljon siihen, millainen Stack tällä hetkellä on (Sangwin 2013). Ohjelmaan on lisätty paljon uusia ominaisuuksia etenkin ihmisen ja tietokoneen väliseen vuorovaikutukseen liittyvissä kysymyksissä. Näitä ovat esimerkiksi monet käyttöliittymään tehdyt parannukset, kuten monta syöttökenttää yhdessä tehtävässä, kaksiulotteiset eli matriisimuotoiset syöttökentät ja selainriippumaton matemaattisten kaavojen esittäminen (ks. Harjula 2008). Viimeisimpiä kehitysaskeleita ovat olleet muun muassa kysymystekstin ehdollinen tuottaminen if- ja foreach-rakenteiden avulla, muuttujien uudelleenmäärittäminen kysymystekstissä ja sisällön tuottaminen kysymyksiin ulkoisilla sovelluksilla perustuen kysymyksen parametreihin tai opiskelijan vastaukseen.

Stackistä on nykyisin käytössä versio 3, joka on asennettu Aallon MyCourses-palveluun, joka on koko yliopiston yhteinen Moodle-alusta. Stack-tehtäviä käytetään käytännössä kaikilla Aallon kandidaattitason insinöörimatematiikan kursseilla. Käytön laajuus esitetään taulukossa 1. Mainitut kurssit suoritetaan pääsääntöisesti kahden ensimmäisen opiskeluvuoden aikana. Stack on käytössä myös monilla syventävillä matematiikan kursseilla, kuten kompleksianalyysin ja diskreetin matematiikan kursseilla. Matematiikan lisäksi Stackiä käytetään fysiikan ja tuotantotalouden logistiikan kursseilla. Myös vanhempi versio 1 on ollut vielä viime vuosiin asti käytössä. Käyttäjiä eri versioilla lukuvuoden 2014–15 aikana oli 1629 ja 1156. Versio 1 poistuu syksyllä 2016.

Taulukko 1: Stack-tehtävien käyttö kandidaattivaiheen insinöörimatematiikan opetuksessa Aalto-yliopistossa lukuvuonna 2014–15. Stackin versiot 1 ja 3 ovat olleet käytössä.

Aine	Ver.	Kurs.	Harj.	Opisk.	Suor.
Matriisilaskenta	1	7	122	796	16506
	3	1	21	40	679
Yhden muuttujan analyysi	1	4	51	721	11427
Monen muuttujan analyysi	1	4	48	705	15536
Vektorianalyysi	3	3	42	226	2748
Todennäköisyyslaskenta	1	2	39	563	12691
ja tilastotiede	3	2	84	452	14294
Yhteensä		23	407	3503	73881

Aallon matematiikan kurssit rakentuvat yleensä luennoista, luokkahuoneessa tapahtuvista harjoituksista ja Stack-harjoituksista. Stack-harjoituksia tehdään joko opiskelijan omalla ajalla tai assistentin ohjaamissa harjoituksissa. Kurssit kestävät yleensä yhden opintoperiodin ajan, jolloin kurssin tyypillinen laajuus on 5 opintopistettä. Opiskelijalla saattaa olla viikoittain jopa kymmenen Stack-tehtävää ratkaistavana. Stack on apuna jatkuvan arvioinnin toteuttamisessa. Opiskelijoiden kurssiarvostelu koostuu perinteisistä kurssikokeista, mutta myös harjoituksissa erilaisilla kriteereillä ansaituista pisteistä. Stack-tehtävien pisteet ovat osana tätä.

Stackistä voidaan sanoa tulleen valtakunnallisen, kun yhteistyöprojekti matematiikan tietokoneavusteisen opettamisen kehittämiseksi alkoi vuonna 2015. Mukana projektissa on käytännössä kaikki matematiikkaa tai luonnontieteitä opettavat yliopistot Suomessa sekä useita ammattikorkeakouluja. Projektin osa on Stack-tehtävien materiaalipankki Abacus (https://abacus.aalto.fi). Projektissa on mukana myös ulkomaisia yliopistoja. Stackiä käyttävät maailmanlaajuisesti mm. Open University ja Loughborough University (Iso-Britannia), Ilias-konsortio (Saksa ja Sveitsi) sekä Leirian teknillinen korkeakoulu (Portugali) (Sangwin 2015).

6.5 Lääkelaskennan oppimisympäristö Sigma

Sähköisiä matematiikan oppimisympäristöjä on käytössä myös muilla aloilla kuin insinöörien tai luonnontieteilijöiden opetuksessa. Arcada-ammattikorkeakoulussa on kehitetty lääkelaskennan opettamiseen tarkoitettu oppimisympäristö *Sigma* osana MAQ-hanketta (*Medication Administration Qualification*) (Leikas, Granberg, Ståhl, Kurko, Antikainen, Airaksinen & Pohjanoksa-Mäntylä 2012). Hankkeen tavoitteena oli kehittää terveydenhuollon opiskelijoiden ja ammateissa toimivien lääkehoidon hallinnan ammattitaitoa ja varmuutta. Tällä on suuri merkitys, joka nähdään lisääntyneenä potilasturvallisuutena.

Sigma on verkossa selaimella käytettävä oppimisympäristö, jonka vahvuutena on oppisisältöjen autenttisuus. Lääkintätehtävä annetaan potilaan tilaan sidottuna, jolloin opiskelija joutuu huomioimaan kaikki vallitsevat olosuhteet lääkkeen annostelua laskiessaan. Tämä on todellinen tilanne, jossa sairaanhoitaja lääkelaskentaa suorittaa.

Sigman esimerkit on luokiteltu laskutavan, lääkkeen antotavan ja lääketieteellisen erikoisalan mukaan. Näin on mahdollista valita opiskelijalle sellaisia tehtäviä, jotka tukevat niitä lääkelaskennan osa-alueita, joissa hän tarvitsee apua.

Lääkelaskennassa on erittäin tärkeää, että esimerkit ja tehtävät ovat todellisia lääkintätilanteita. Opiskelijalle ei saa jäädä virheellistä kuvaa esimerkiksi lääkkeiden annostelun suuruusluokista jonkin kuvitteellisen harjoituksen perusteella.

6.6 Yhteenveto

Automaattisen tarkastamisen toteuttaminen on käytännössä ohjelmointia, jossa tehtävään rakennetaan sisäinen testausjärjestelmä. Opiskelijan antamaa vastausta voidaan verrata niin mallivastaukseen kuin muihinkin mahdollisiin piirteisiin. Näin voidaan etsiä erimerkiksi tyypillisiä opiskelijan tekemiä virheitä, jolloin niihin voidaan antaa sopiva palaute.

Automaattisen tarkastamisen mahdollistavan oppimisympäristön käytöllä on havaittu olevan monia etuja, kuten joustavuus perinteiseen lähiopetukseen verrattuna, personoitujen tehtävien laatiminen ja palautteen antaminen, jatkuvan arvioinnin mahdollisuus, tehtävien dynaamisen sisällön mahdollistaminen, oppimispelien tekeminen ja automatisoitu oppimisanalytiikka.

Stackiä on käytetty Aalto-yliopiston insinöörimatematiikan opetuksessa vuodesta 2006. Ympäristöä on myös kehitetty paljon Aallossa. Stackin matematiikkamoottorina on symbolisen laskennan ohjelmisto Maxima. Nykyinen Stack on integroitu Moodle-oppimisympäristöön. Tämä näkyy käytännössä siten, että Stack-tehtävä on yksi Moodlen tarjoama tehtävätyyppi. Tästä on seurauksena myös se, että Stackin käyttöliittymä tulee Moodlesta.

Sähköisiä oppimisympäristöjä on käytössä myös muiden alojen matematiikan opetuksessa. Sigma on lääkelaskennan opettamiseen kehitetty oppimisympäristö, joka mahdollistaa lääkelaskennan opiskelun autenttisten esimerkkien ja harjoitusten avulla.

7

Taustateoria

7.1 Tavoitteena luotettava automaattinen virheluokittelu

Edellä on luotu katsaus matematiikan opettamiseen ja oppimiseen sekä erilaisiin matematiikan opetuksessa käytettyihin välineisiin aina antiikin ajoista uusimpiin tietoteknisiin oppimisympäristöihin. Tietokoneiden avulla on mahdollista rakentaa oppimisympäristöihin sellaisia ominaisuuksia, joista aiemmin on voitu vain haaveilla. Nykytekniikalla oppimisympäristöstä voidaan rakentaa älykäs, jolloin se mukautuu opetuksen ja opiskelijan tarpeisiin tuottaen yksilöllistä opetusta.

Tässä tutkielmassa lähestytään oppimisanalytiikan rakentamista virheanalyysin kautta. Perusajatus on yksinkertainen: jos opiskelijan tekemä virhe voidaan automaattisesti kategorisoida, on mahdollista selvittää hänen oppimisprosessinsa tila. Yksittäisen opiskelijan tilanteen lisäksi tietoa virheistä voidaan kerätä myös suurelta joukolta opiskelijoita, jolloin erilaiset tilastolliset analyysit tulevat mahdollisiksi. Molemmissa tapauksissa lopputuloksia voidaan hyödyntää muokkaamaan oppimisympäristön toimintaa siten, että sen tuottama opetus on mahdollisimman hyödyllistä opiskelijoiden kannalta.

Tämä on kuitenkin vaativa tehtävä. Virheluokittelun ytimessä ovat virheluokat, joiden valinta voidaan tehdä enemmän tai vähemmän sofistikoituneesti. Parhaimmillaan ne heijastavat opetuksessa käytettyjä pedagogisia perusteita. Virheluokkien tulee ylipäätään olla sellaisia, että luokittelu niiden avulla on mah-

Kuva 6: Pedagoginen malli ohjaa opiskelijan oppimisprosessia, jota voidaan analysoida virheanalyysin keinoin. Virheanalyysi tuottaa virheiden kategorisointeja, joilla on tärkeää olla hyvä reliabiliteetti. Tämä ei aina ole itsestään selvää, vaan se on selvitettävä osana analyysia: onko pedagogisessa mallissa sellaisia piirteitä, joiden avulla voidaan luoda validi virheanalyysimalli, ja onko näin saatu malli luotettava? Tässä tutkimuksessa luotua virheluokittelumallia tutkittiin opettajien tekemien luokittelujen avulla. Kysymyksenä oli, että luokittelevatko opettajat samalla tavalla, onko virheanalyysi tällä tavoin konsistentti. Jos näin on, voidaan ajatella, että tietokone voisi tehdä saman luokittelun. Tällöin olisi mahdollista käsitellä suuri määrä dataa, ja löytää esimerkiksi piilomuuttujia. Näin saadun tiedon avulla on mahdollista palata takaisin kuvion alkuun: ohjata oppimisympäristön toimintaa, ja siten vaikuttaa opiskelijan oppimisprosessiin.

dollinen. Esimerkiksi luokkien lukumäärällä on vaikutusta siihen, miten helppoa luokittelujärjestelmää on käyttää.

Virheluokittelun tulee olla *validi*, eli sen tulee erotella oppimisprosessin kannalta olennaisia asioita toisistaan. Validius yleensä tarkoittaa menetelmän luotettavuutta, miten hyvin se toimii aiotussa tehtävässään (Tella & Lavonen 1995).

Virheluokittelun tulee olla *reliaabeli*, mikä tarkoittaa sen pysyvyyttä (Tella & Lavonen 1995). Eri luokittelijoiden tulee päätyä samaan tai samansuuntaiseen luokitteluun mallin avulla. Luokittelu ei siis saa antaa sattumanvaraisia tuloksia. Luokittelun tulee olla myös *stabiili*, sen tulee kestää satunnaisvirheitä.

Tässä tutkielmassa laaditaan virheluokittelumalli lääkelaskentatehtävien analysoimiseksi perustuen 4 Cs -opetusmalliin, josta lisää myöhemmin tässä luvussa. Mikäli luokittelumalli voidaan näyttää luotettavaksi, se voi olla käyttökelpoinen myös oppimisympäristössä toteutettavaksi. Tutkimuksen perusidea on esitetty kuvassa 6.

7.2 Virheluokittelu

Virhe syntyy, kun ihmisen toiminta poikkeaa aiotusta. Huhtalan (2000) mukaan matemaattisessa ongelmanratkaisussa syntyvät virheet voidaan jakaa karkeasti kahteen luokkaan. Virheet voivat olla *tahallisia*, jolloin opiskelija toimii kuten uskoo oikeaksi. Toisin sanoen hän toimii kuten aikoi, mutta toiminta sinänsä oli väärää. Matematiikassa tämä viittaa ongelmiin konseptuaalisessa tiedossa. Toinen virheiden luokka on *tahattomat*, jolloin toiminnan tavoitteena on ollut jotain muuta kuin mitä tapahtui. Näin käy esimerkiksi huolimattomuudesta tai proseduraalisten taitojen ollessa puutteellisia.

Mulhern (1989) on tarkastellut tutkimuksia opiskelijoiden tekemistä virheistä, ja määritellyt virheille seuraavia yleisiä piirteitä.

- Virheet ovat *yllättäviä*.
- Virheet ovat *pysyviä*, ne eivät muutu itsestään ja korjaaminen saattaa vaatia opiskelijan tietojen täydellistä uudelleen järjestäytymistä (eli luvussa 3 mainittua akkomodaatiota).
- Virheet voivat olla *systemaattisia* tai *satunnaisia*. Systemaattiset virheet ovat tehokkaampia ajatteluprosesseja selvitettäessä. Satunnaiset virheet taas kertovat huolimattomuudesta ja hetkittäisistä lipsahduksista.
- Virheet ovat usein *järjettömiä*, eli virheellinen vastaus on sellainen, että se ei mitenkään voisi olla tehtävän mahdollinen vastaus.

Mulhern (1989) listaa myös seuraavat tavat analysoida virheitä.

- Lasketaan virheellisten suoritusten määrä erityyppisissä laskutoimituksissa.
- Analysoidaan tehtyjen virheiden tyypit ja erot oikeista ratkaisuista.
- Analysoidaan virhemalleja. Tällaisella analyysilla voi paljastua systemaattisia tai satunnaisia virheitä.
- Tähdätään virheisiin, eli tutkija laatii tehtäviä siten, että opiskelijat tekevät yksilöllisiä virheitä.

Virheiden tutkimista luokittelemalla kutsutaan *virheanalyysiksi*. Matematiikan opetusta ja oppimista on tutkittu virheanalyysin avulla jo 1920-luvulta asti. Menetelmää käytettiin aluksi alkeismatematiikkaan (esim. Cox 1975 ja Radatz 1979). 1970-luvulla menetelmä oli hetken pedagogisen tutkimuksen valtavirrassa, ja sitä sovellettiin muihinkin matematiikan alueisiin kuten polynomi- ja todennäköisyyslaskentaan ja differentiaalilaskentaan. (Pehkonen 2009)

Virheanalyysin tutkimus on keskittynyt viiteen eri tavoitteeseen (Radatz 1980).

1. Kaikkien mahdollisten virhemenetelmien listaaminen,
2. virhemenetelmien frekvenssijakauman selvittäminen eri ikäryhmissä,
3. erikoisvaikeuksien analysointi,
4. yksittäisten virhemenetelmien pysyvyyden määrittäminen ja
5. virheiden luokittelu.

Virheluokittelun tekemisen keskeinen ongelma on kategorisointijärjestelmän määrittäminen. Lähes jokainen virheanalyysia luokittelemalla tehnyt tutkimus on päätynyt erilaiseen luokitteluun. Tähän vaikuttaa erityisesti luokittelun pohjaksi valittu näkökulma. Esimerkiksi Radatz (1979) luokitteli virhetyyppejä tiedonprosessoinnin kautta. Hänen lähtökohtanaan oli, että matematiikka on opiskelijalle kuin vierasta kieltä, jota hän joutuu tulkitsemaan päästäkseen ratkaisuun. Tämän kuten monien muidenkin luokittelujen heikkous on, että yksittäinen virhe saattaa johtua useammasta tekijästä (Pehkonen 2009).

Kritiikkiä virheluokittelua käyttävälle tutkimukselle voidaan osoittaa sikäli, että harva tutkimus ottaa minkäänlaista kantaa siihen, miten luotettava käytetty luokittelujärjestelmä on. Luokittelumalli yleensä vain ilmoitetaan ja otetaan käyttöön ilman analysointia. Pystyykö kaksi luokittelijaa saamaan mallin avulla samansuuntaisia tuloksia, eli onko malli reliaabeli. Kysymys jää usein kysymättä.

Esimerkkinä tässä suhteessa hyvästä tutkimuksesta on Movshovitz-Hadarin, Zaslavskyn ja Inbarin (1987) tekemä Israelin ylioppilaskirjoitusten virheiden analyysi. Tässä tutkimuksessa luokittelijoiden tuloksia verrattiin Kendallin järjestyskorrelaatiokertoimella. Tämä antoi tietoa siitä, vallitseeko luokittelijoiden mielestä yksimielisyyttä yleisimmästä virheestä, toiseksi yleisimmästä ja niin edelleen. Edes tämän kaltaista analyysia ei kuitenkaan virheluokitteluista yleensä löydy.

7.3 Lääkelaskennan pedagoginen malli 4 Cs

Johnson ja Johnson (2002) ovat kehittäneet sairaanhoidon teoriaan, sosiaaliseen oppimiseen ja kriittisen ajattelun teorioihin pohjautuvan opetusmetodin nimeltään *4 Cs*[1] lääkelaskennan opettamiseen. Nimi 4 Cs tulee kuvassa 7 esitetyistä neljästä lääkelaskentataidon osa-alueesta: 1) laske (*Compute*), 2) muunna (*Convert*), 3) ymmärrä (*Conceptualize*) ja 4) arvioi kriittisesti (*Critically evaluate*).

Mallin osassa *laske* opiskelija tutustuu peruslaskutoimituksilla laskemiseen. Tässä vaiheessa havaitaan yleensä ongelmia, jotka liittyvät peruslaskutaitojen ja laskurutiinin puutteeseen. Laskutoimitukset itsessään saattavat olla opiskelijalle vieraita.

Osassa *muunna* opiskelija joutuu käsittelemään erilaisia suuruusluokkia ja yksiköiden muuntamista toisiksi yksiköiksi. Opiskelija ei tarvitse vain laskutoimitusten hallintaa, vaan myös käsitystä suuruusluokista ja taitoa käyttää oikeita muuntokertoimia.

Opiskelijoiden tulee pystyä valitsemaan oikea matemaattinen ratkaisustrategia tai laskutapa annettuun ongelmaan Tämä on mallin osa *ymmärrä*. Tässä vaiheessa opiskelijoiden tulee myös osata päätellä, mitä tietoa tarvitaan käsillä olevan matemaattisen ongelman ratkaisemiseksi, ja miten vastaus kannattaa ilmoittaa yksiköiden ja tarkkuuden suhteen.

Neljännessä osa-alueessa *arvioi kriittisesti* opiskelijan tulee soveltaa ongelmanratkaisuprosessia arvioidakseen, onko tulokseksi saatu vastaus sovelias ratkaisu esitettyyn ongelmaan.

Mallin mukaan lääkelaskennan oppimiseen liittyy vielä koulun tarjoamat puitteet sekä tietyt opiskelijan ominaisuudet. Lääkelaskennan oppiminen on opiskelijan ja koulun vuorovaikutusta, jonka seurauksena taidot mainituilla neljällä eri osa-alueella kehittyvät.

4 Cs -mallin osat ovat hyvin samanlaisia kuin luvussa 3 mainittu luokittelu matemattisisten taitojen osa-alueiksi (Kilpatrick ym. 2001). Erilaiset laskemisen taidot vastaavat Kilpatrickin proseduraalisia taitoja. 4 Cs:n ymmärrä tarkoittaa konseptuaalista ymmärtämistä ja strategiset taidot liittyvät siihen, että opiskelija osaa valita oikean ratkaisumenetelmän. Lisäksi kun opiskelijan motivaatio rinnastuu Kilpatrickin mielenkiintoon, jää ainoastaan deduktiiviset kyvyt käsittelemättä. Niitäkin tarvitaan loogisen päättelyn osalta.

[1] Tässä tutkielmassa käytetään mallin nimenä sen englanninkielistä lyhennettä 4 Cs, jossa s-kirjain tulee monikosta *neljä C:tä*. Olisiko suomeksi kirjoitettuna esimerkiksi nimi 4C parempi?

Kuva 7: 4 Cs -opetusmalli (Johnson & Johnson 2002). Opiskelija ja koulun tarjoamat puitteet vuorovaikuttavat ja luovat perustan oppimiselle eli lääkelaskentataidon neljän osa-alueen kehittymiselle.

7.4 4 Cs -opetusmallista johdettu luokittelujärjestelmä

4 Cs -mallin neljästä C:stä kolme otetaan luokittelujärjestelmän luokiksi. Nämä ovat *laske*, *muunna* ja *ymmärrä*. Luokittelumalli tehdään *täydelliseksi* lisäämällä luokka *ei voida luokitella*. Tällöin jokainen virhe voidaan asettaa johonkin luokkaan. Tämä antaa luokittelijalle mahdollisuuden hylätä kolme varsinaista luokkaa. Luokittelumalli on *erillinen*, mikä tarkoittaa, että jokainen virhe kuuluu täsmälleen yhteen luokkaan.

4 Cs -mallin neljäs C, *arvioi kriittisesti*, jätetään pois kategorisointijärjestelmästä. Tämä tehdään siksi, että luokittelussa halutaan keskittyä vain opiskelijan

tekemään ensimmäiseen virheeseen. Mikäli opiskelija epäonnistuu vastauksensa kriittisessä arvioinnissa, on jo jonkinlainen muu virhe yleensä tapahtunut.

7.5 Klusterointi

Virheluokittelumallin reliabiliteettia tutkitaan *klusteroinnin* avulla. Se on signaalinkäsittelyyn kehitetty menetelmä, jolla pyritään etsimään datasta keskenään samankaltaisia tietoa. Klusteri on samankaltaiseksi miellettyjen vektorien tai datapisteiden kokoelma. Klusteria voidaan esittää edustajavektorin avulla, joka voidaan ajatella eräänlaiseksi klusterin keskipisteeksi tai -arvoksi.

Klusteroinnilla on paljon käytännön sovelluksia. Sitä käytetään tiedonlouhinnassa, tilastollisessa analyysissä, koneoppimisessa, hahmontunnistuksessa, hakukoneiden logiikassa, bioinformatiikassa, tiedon pakkaamisessa ja tietokonegrafiikassa.

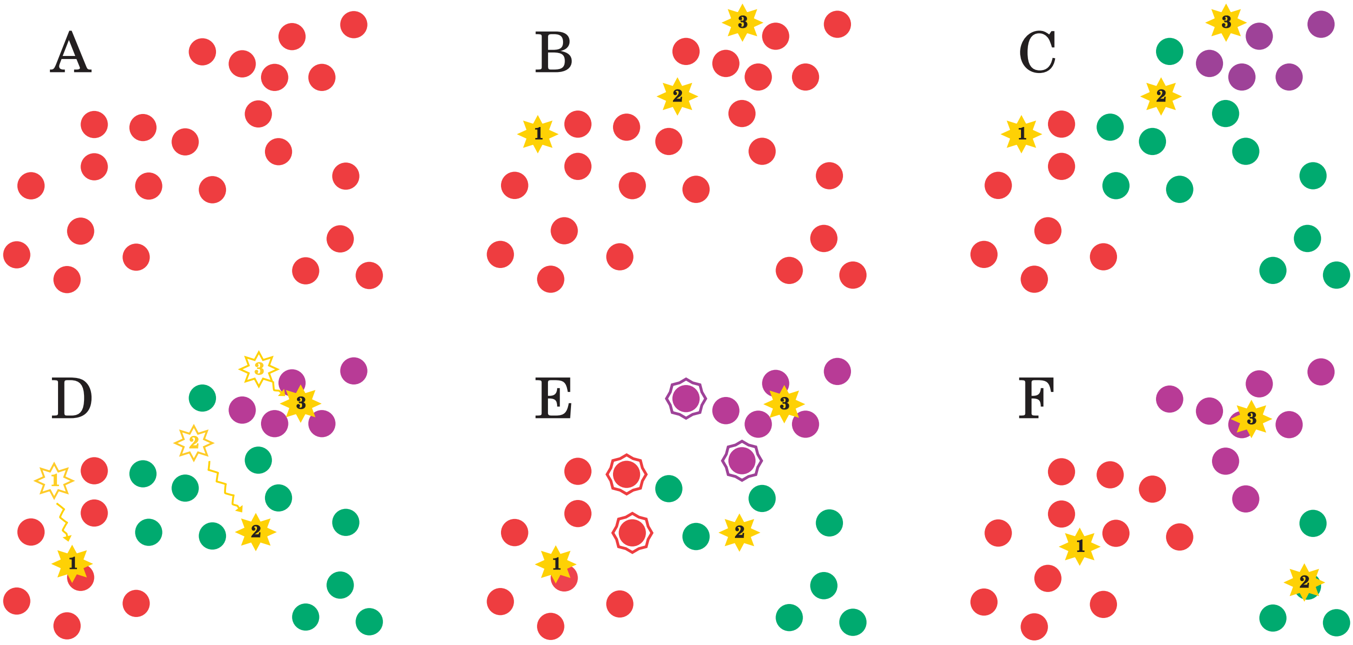
Klusterointi itsessään ei ole yksittäinen algoritmi, vaan ratkaistava ongelma. Klusterointiin on kehitetty monia erilaisia algoritmeja, jotka eroavat sen perusteella, miten ne päättelevät vektoreiden samankaltaisuutta, ja kuinka tehokkaasti ne toimivat. Datan laatu voi vaikuttaa suuresti algoritmin tehokkuuteen. Eri algoritmit muodostavat klustereita erilaisten kriteerien perusteella, kuten pisteiden väliset etäisyydet, pistejoukkojen tiheys tai pisteiden tilastollinen jakautuminen.

7.6 k-means-klusterointi

k-means-klusteroinnissa datajoukko jaetaan k:hon klusteriin, joissa kussakin on mahdollisimman paljon toisiaan muistuttavia alkioita. k-means käyttää samankaltaisuuden kriteerinä euklidista etäisyyttä, eli lähellä toisiaan olevat pisteet tulkitaan samankaltaisiksi.

Algoritmin iteratiivinen toiminta on esitetty kuvassa 8. Alkutilanteessa A on joukko punaisia pisteitä, joiden dimensio tässä tapauksessa voidaan ajatella olevan 2. Tämä joukko halutaan jakaa kolmeen osaan k-means-algoritmilla ($k = 3$).

Tilanteessa B algoritmi aloittaa toimintansa asettamalla joukkoon kolme ehdotusta klustereiden keskipisteiksi (numeroidut tähdet 1–3). Näiden pisteiden

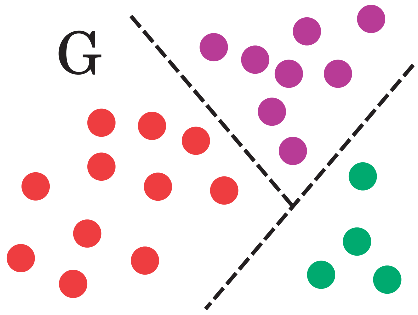


Kuva 8: k-means-algoritmin vaiheet. Joukko pisteitä (A) halutaan jakaa kolmeen klusteriin. Algoritmi aloittaa arpomalla joukkoon kolme klusterin keskipistettä (B). Tämän jälkeen kukin joukon piste väritetään eri värillä sen mukaan, mitä keskipistettä lähinnä se on (C). Keskipisteet siirretään keskelle oman värisiä pisteitään (D). Väritys tehdään uudestaan. Mikäli jokin piste vaihtaa väriä (ympyröidut kohdassa E), siirrytään takaisin kohtaan (D). Jos pisteet pysyvät saman värisinä uudelleenvärityksessä, klusterointi on löytynyt ja algoritmi pysähtyy (F).

optimaalisen sijainnin päättäminen ei ole yksinkertainen tehtävä, vaan algoritmi voi olla toteutettu siten, että ensimmäisten keskipisteiden sijainnit yksinkertaisesti arvotaan. Tätä vaihetta voidaan yrittää optimoida tilastollisin menetelmin tai käyttämällä hyväksi ennalta tiedettyjä ominaisuuksia datasta. Tässä esimerkissä ajatellaan, että ensimmäiset keskipisteet on arvottu.

Tämän jälkeen jokainen joukon piste väritetään sen mukaan, mitä keskipistettä lähimpänä se on. Kohdassa C on esitetty väritykset siten, että keskipistettä 1 lähinnä olevat pisteet on väritetty punaisiksi, keskipistettä 2 lähinnä olevat vihreiksi ja keskipistettä 3 lähinnä olevat pisteet violeteiksi. Nämä kolme väritystä ovat ensimmäiset ehdotukset halutuiksi kolmeksi klusteriksi.

Seuraavaksi keskipisteet 1–3 siirretään keskelle klusteriaan eli oman värisiä pisteitään. Tämä tehdään minimoimalla keskipisteen yhteenlaskettu etäisyys kaikista klusteriin kuuluvista pisteistä (kuvan kohta D). Tämän jälkeen pisteet väritetään uudelleen. Nyt on mahdollista, että joku tai jotkut pisteet vaihtavat väriä. Kuvan kohdassa E on ympyröity pisteet, joille on käynyt näin. Tämä on merkki siitä, että algoritmi jatkaa vielä toimintaansa, eli värityksen jälkeen keskipisteet 1–3 hakeutuvat jälleen muuttuneen joukkonsa keskelle ja uusi väritys tehdään. Kun uusi väritys ei muuta yhdenkään pisteen väriä, on algoritmi päässyt loppuun ja löytänyt jaon kolmeen klusteriin. Tämä nähdään kuvan 8 kohdassa F ja uudelleen kuvassa 9.



Kuva 9: k-kmeans algoritmin löytämät kolme klusteria.

Seuraavassa on esitetty edellä selitetyt k-meansin vaiheet kompaktimmin eli algoritmi pseudokoodilla kuvattuna. Algoritmissa esiintyy kustannusfunktio J, joka ilmaisee klusterin keskipisteen etäisyyden kaikista klusterin pisteistä. Algoritmin eteneminen pyrkii tämän funktion arvon minimoimiseen. Kuntannusfunktio määritellään seuraavasti.

$$J = \sum_{i=1}^{K} \sum_{x_i \in Q_k} \|x_i - c_k\|^2.$$

Algoritmi käy läpi seuraavat neljä vaihetta.

1. Valitse K kappaletta klusterien keskipisteitä c_k. Tämä voidaan tehdä esimerkiksi satunnaisesti.

2. Jaa data klustereihin Q_k siten, että kustannusfunktio J minimoituu.

3. Laske uudet klusterien keskipisteet

$$c_k = \frac{\sum_{x_i \in Q_k} x_i}{N_k},$$

missä N_k on pisteiden lukumäärä klusterissa Q_k.

4. Jos klusterit eivät muutu, lopeta. Muussa tapauksessa palaa takaisin kohtaan 2.

k-means-algoritmin antama jako riippuu useimmiten ensimmäisten keskipisteiden sijoittelusta. Se ei siis aina löydä samanlaista jakoa ajettaessa uudelleen samalle joukolle pisteitä. k-means-algoritmin yksityiskohdat löytyvät kirjallisuudesta kuten Steinhaus (1957), MacQueen (1967), Lloyd (1982) ja Bock (2008).

Tässä tutkimuksessa k-means-algoritmia käytettiin arvioimaan opettajen tekemien virheluokittelujen samankaltaisuutta. Virheluokitteluaineisto koodattiin vektorimuotoon, jota klusteroitiin. Tarkoituksena oli tutkia, osoittautuvatko luokittelut toistensa kaltaisiksi. Tästä voidaan päätellä virheluokittelun luotettavuutta. Myös luokittelun herkkyyttä satunnaiselle virheelle voidaan tutkia klusteroinnin avulla.

8

Tutkimusasetelma

Lääkelaskenta valikoitui tämän virheanalyysin kohteeksi sen merkittävyyden vuoksi. Lisäksi lääkelaskentaa on insinöörimatematiikan tavoin opetettu tietokoneavusteisesti jo aiemmin. Lääkelaskennalle ominaista pedagogiikkaa on ilmaistu tietoteknisin apuvälinein. Tämä tutkimus on ensimmäisten askelten ottamista kohti virheen syntymisen ymmärrystä lääkelaskennan peruslaskutoimituksissa. Syvempi tavoite on oppia ymmärtämään teoreettista taustaa virheen takana ja käytännöllisiä keinoja virheen synnyn analysoimiseksi. Yleinen tavoite on vähentää lääkelaskennassa tapahtuvia laskuvirheitä, mikä onnistuessaan johtaa parempaan potilasturvallisuuteen sairaaloissa. Tutkielmassa esitellään ja validoidaan tilastollisesti pitävä menetelmä, jolla luokitellaan sairaanhoitajaopiskelijoiden lääkelaskennan tehtävissä tekemiä virheitä.

Kategorisointijärjestelmän perustana pidetään lääkelaskennan 4 Cs -opetusmallia. Tämä malli on tehty erityisesti lääkelaskennan opettamista varten (Johnson & Johnson 2002). Tavoitteena on löytää sellainen luokittelumalli, joka voitaisiin toteuttaa automaattisesti sähköisessä oppimisympäristössä. Tällöin voitaisiin arvioida hyvin suuria määriä vastauksia ja analysoida tehtävien suorittamista laajoista vastaustietokannoista.

Tässä tutkielmassa virheluokittelua tarkastellaan inhimillisestä näkökulmasta. Lähtökohdaksi otetaan, että virheluokittelun tulee olla ihmisen tehtävissä kohtuullisella luotettavuudella, ennen kuin sen automatisointia kannattaa edes aloittaa. Tätä tutkitaan ottamalla joukko sairaanhoitajaopiskelijoiden virheellisiä tenttivastauksia, jotka luokitellaan 4 Cs -malliin perustuviin luokkiin. Jos virheluokittelu ei onnistu koeluokittelijoilta, on vähän toivoa siitä, että automaattinen järjestelmä tekisi tehtävän yhtään paremmin.

Tässä tutkimuksessa saadaan ensimmäiset tulokset siitä, antavatko 4 Cs -mallin mukaiset luokat perustan virheluokittelulle. Tämä on tärkeä tieto, jonka perusteella automaattista virheluokittelujärjestelmää voidaan lähteä kehittämään.

Tutkimus etsii vastauksia seuraaviin kysymyksiin.

1. Onko 4 Cs -mallin pohjalta tehty virheluokittelu luotettava ja stabiili?
2. Voiko 4 Cs -malliin perustuva virheluokittelu toimia sairaanhoitajaopiskelijoiden tekemien virheiden virheluokittelun perustana?
3. Toimiiko 4 Cs -malli virheluokittelun perustana sellaisenaan, vai tarvitseeko sitä muokata jotenkin?
4. Löytyykö aineistosta sellaisia virheitä, joita ei 4 Cs:n pohjalta voida luokitella?

Lisäksi tutkielman lopussa pohditaan, miten edellä kuvailtu automaattinen virheluokittelumalli voitaisiin toteuttaa Stack-oppimisympäristössä.

9

Tutkimuksen suorittaminen

9.1 Primäärinen aineisto

Tutkimuksessa käytetty aineisto kerättiin sairaanhoitajaopiskelijoiden tenttivastauksista. Käytössä oli kahden vuosikurssin koevastaukset ($N = 88$). Materiaali kerättiin Arcada-ammattikorkeakoulussa Helsingissä. Molempien vuosikurssien kokeessa oli 10 lääkelaskentatehtävää, ja aihepiirit käsittivät koko lääkelaskentakurssin.

Näistä vastauspapereista kerättiin kaikki virheelliset vastaukset. Tästä saatiin aineisto ($n = 90$), joka koostui käsin kirjoitetuista virheellisistä koevastauksista. Virheelliset vastaukset kopioitiin paperille siten, että yhdellä paperilla oli aina yksi virheellinen vastaus. Tehtävästä oli poistettu korjaavan opettajan merkinnät niin hyvin, kuin se oli mahdollista. Vastaukset oli numeroitu ja opiskelijan henkilöllisyys peitetty muuten kuin käsialan osalta. Tämä oli tutkimuksen *primäärinen aineisto*.

9.2 Luokittelijat

Primäärisen aineiston luokitteli yhdeksän matematiikan ja sairaanhoidon opettajaa ($m = 9$), joita kutsutaan jatkossa *luokittelijoiksi*. Heistä kaksi oli taustaltaan sairaanhoitajaopiskelijoiden lääkelaskennan opettajia, seitsemän muuta olivat joko matematiikan tai sairaanhoidon muiden aihepiirien opettajia. Kokeiden alkuperäinen arvostelija oli yksi luokittelijoista.

Luokittelijoiden ikä vaihteli 34 vuodesta eläkeikään. Luokittelijoista kahdeksan oli naisia ja yksi mies. Seitsemällä luokittelijalla oli sairaanhoitajan pätevyys. Kaikki heistä toimivat aktiivisesti opetustehtävissä joko toisella asteella (yksi luokittelija) tai ammattikorkeakoulussa (kuusi).

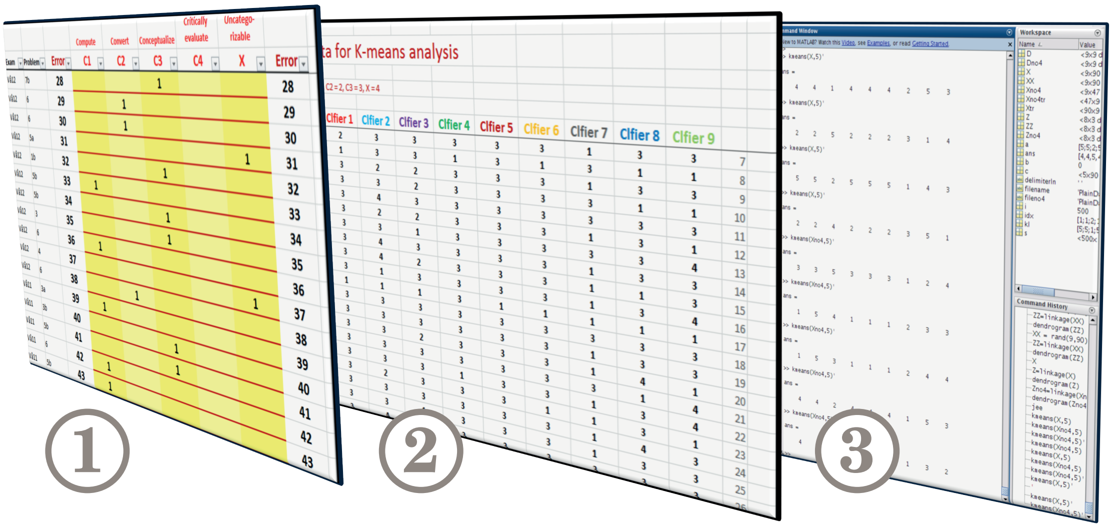
Seitsemän luokittelijaa oli opetustyönsä puolesta tiiviisti yhteydessä lääkityksen annostelemiseen joko lääkitykseen liittyvän sairaanhoidon tai siihen liittyvän matematiikan opettajina. Kaksi luokittelijoista oli lääkelaskennan opettajia. Kahden luokittelijan tausta oli matematiikan tai kemian opettajuus. Toisella heistä ei ole yhteyttä sairaanhoitajakoulutukseen, mutta toinen on ollut mukana niin perusmatematiikan opettamisessa kuin erilaisten tukiopetusten antamisessa hoitotyön opiskelijoille.

9.3 Primäärisen aineiston luokittelu

Primäärinen aineisto luokiteltiin luokittelijoiden subjektiivisena mielipiteenä 4 Cs -malliin perustuvan luokittelujärjestelmän soveltamisesta. Luokittelussa pyrittiin löytämään opiskelijan tekemä *ensimmäinen virhe* ja antaa sille luokka. Useinhan väärässä vastauksessa on monia virheitä, mutta tässä tarkastelussa oltiin kiinnostuttu vain siitä hetkestä, kun tehtävän ratkaisu lähtee menemään väärin.

Koska ensimmäisen virheen löytäminen saattaa olla hankalaa, luokittelijoille annettiin mahdollisuus ehdottaa myös muille virheille luokkaa. Näitä virheitä kutsutaan *jatkovirheiksi*. Mahdollisuus merkitä jatkovirheet ylös annettiin siksi, että tämä rohkaisisi merkitsemään ensimmäisen virheen myös tilanteessa, jossa ensimmäinen ja toinen virhe on hankala erottaa toisistaan. Tällöin luokittelija ehkä valitsisi *ei voida luokitella* -kategorian.

Luokittelutehtävä esiteltiin luokittelijoille sekä henkilökohtaisesti selittäen että kirjallisesti. Lisäksi luokittelijoilla oli luettavanaan artikkeli Johnson ja Johnson (2002) 4 Cs -mallista. Vain kahdella luokittelijalla oli aikaisempaa tietoa tästä mallista. Kaikki luokittelijat tekivät työnsä itsenäisesti, omaa tahtiaan ja valitsemassaan ympäristössä. Luokittelun kesto vaihteli 2 ja 8 tunnin välillä.



Kuva 10: Sekundäärisen aineiston tuottaminen ja analysointi. Luokittelijat luokittelivat jokaisen virheen etukäteen laadittuun Excel-taulukkoon (1). Luokittelut koodattiin luvuiksi 1–4, joista 1–3 kuvasi luokittelujärjestelmän luokkia *laske*, *muunna* ja *ymmärrä*. Luku 4 merkitsi luokkaa *ei voida luokitella* (2). Näin saatu numeerinen data analysoitiin Matlabin kmeans-funktiolla (3).

9.4 Sekundäärinen aineisto ja sen analysoiminen

Luokittelijat raportoivat työnsä valmiiseen pohjaan tehdyllä Excel-taulukolla, joka on esitetty kuvan 10 kohdassa 1. Luokittelut ovat tämän tutkimuksen *sekundäärinen aineisto*, jota analysoitiin tilastollisesti. Luokittelut koodattiin 90-ulotteisiksi vektoreiksi siten, että virhettä numero i vastasi vektorin kohta i. Tähän sijoitettiin luku 1–4 osoittamaan luokittelijan tekemää luokitusta (*laske* = 1, *muunna* = 2, *ymmärrä* = 3 ja *ei voida luokitella* = 4).

Näin saatiin yhdeksän vektoria ($m = 9$), joiden mahdollista samankaltaisuutta analysoitiin k-means-klusterointialgoritmilla, joka ajettiin MATLAB 8.1.0.604 (R2013a) -ohjelmiston kmeans-funktiolla.

Algoritmin metriikkana oli euklidinen etäisyys. Algoritmi ryhmittelee luokittelijat k:hon alaluokkaan, missä k voidaan valita arvojen $k = 1$ ja $k = m$ välillä. Käytössä oli arvo $k = \text{floor}(m/2) + 1$. Kun $m = 9$, tulee arvoksi $k = 5$. Tämä on suurin määrä luokkia, jolla yhden luokan on mahdollista muodostaa yksinkertainen enemmistö. Tässä tilanteessa kaikki muut luokat koostuvat vain yhdestä luokittelijasta.

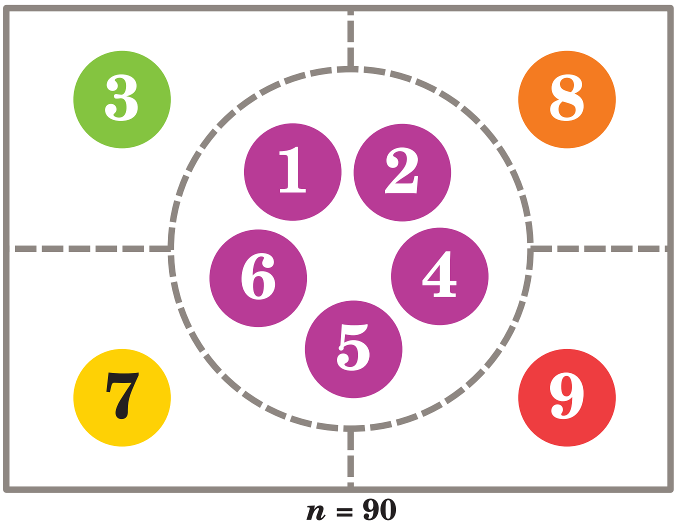
Kuten taustateoriassa mainittiin, k-means on iteratiivinen algoritmi, jonka lähtötilanteen Matlab valitsee satunnaisesti. Näin ollen klusterointi voi vaihdella, kun ajo suoritetaan monta kertaa. Tämä tarkoittaa, että enemmistö voi muodostua erilaiseksi eri klusterointikerroilla.

Kaikki tämän tutkimuksen klusteroinnit suoritettiin koko joukolle luokitteluja (n = 90). Klusterointien stabiiliutta arvioitiin suorittamalla klusterointeja, joissa hylättiin satunnaisesti 5 % (jolloin $n = 85$) ja 10 % ($n = 81$) datasta.

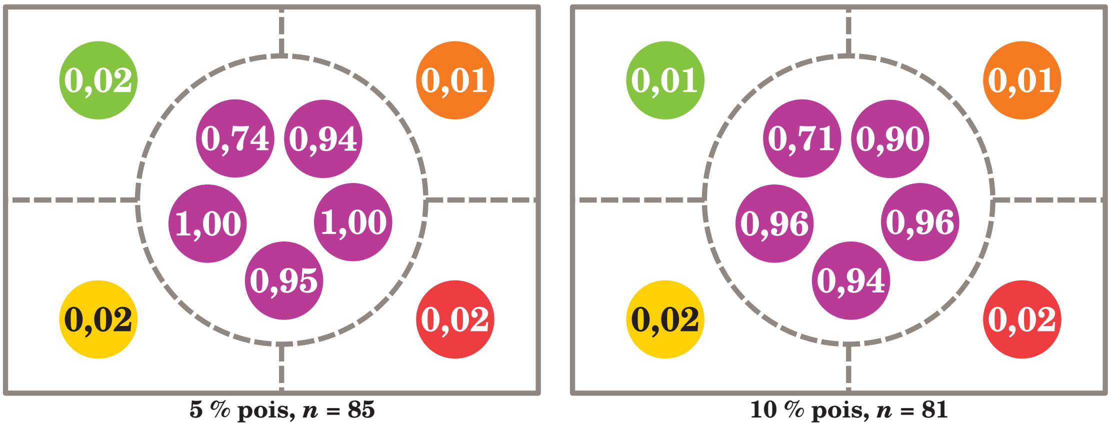
9.5 Tulokset

Ensimmäinen karkea havainto on, että luokittelijoiden tekemät virheluokittelut ovat kaukana satunnaisista. Eri luokittelijoiden tekemien luokittelujen välillä on kohtuullinen korrelaatio. Tätä ei kuitenkaan tässä tutkimuksessa selvitetty sen tarkemmin, vaan keskityttiin luokittelujen samankaltaisuuteen.

Luokittelun tulokset näyttävät, että aineistolla on voimakas pyrkimys muodostaa konsensusklusteri, joka muodostaa samalla yksinkertaisen enemmistön (kuva 11). Tämä on vakaa tulos siinä mielessä, että siihen ei vaikuta merkittävästi satunnainen 5 %:n tai jopa 10 %:n poistaminen tutkimuksen materiaalista.



Kuva 11: k-means-analyysi sekundääriseen dataan (k = 5). Lopputuloksena syntyy viiden luokittelijan konsensusklusteri, jolle ei synny kilpailevaa klusteria, vaan yli jäävät luokittelijat muodostavat omat klusterinsa. Kuva ei ole realistinen pisteiden sijaintien suhteen.



Kuva 12: Klusteroinnin stabiiliutta arvioitiin poistamalla aineistosta satunnaisesti 5 % ja 10 %. Todennäköisyydet on laskettu poistamalla aineistosta 500 kertaa satunnaisesti valittu osa ja tekemällä näille kullekin 100 kertaa k-means-klusterointi.

9.6 Stabiilius

Todennäköisyydet osoittavat, että mitä suurempi osa primääristä aineistosta on mukana luokittelussa, sitä todennäköisempää on alkuperäisen konsensusklusterin jäsenten kuuluminen konsensusklusteriin. Tuloksista näkyy myös, että luokittelija numero 1 on todennäköisimmin jättämässä konsensusklusterin ensimmäisenä.

Toinen havainto on, vaikkakin ilman tilastollista perustelua, että luokittelijoiden kaksi kokeneinta matematiikan opettajaa (numerot 4 ja 5) kuuluvat hyvin suurella todennäköisyydellä konsensusklusteriin. He päätyvät sinne, vaikka primääristä datasta olisi poistettu osa, kuten edellä on selitetty. Luokittelijat käyttivät luokkaa *ei voida luokitella* 4, 7, 21, 0, 8, 2, 0, 17 ja 14 kertaa (esitetty luokittelijoiden numerojärjestyksessä). Tästä nähdään, että konsensusklusteriin usein päätyvät luokittelijat käyttävät vähän tätä luokkaa.

Suorittamalla 500 erilaista valintaa, joissa kussakin jätetään pois satunnainen osa datasta, ja laskemalla kullekin valinnalle 100 kertaa k-means klusterointi, saadaan taulukossa 2 ja kuvassa 12 esitetyt todennäköisyydet luokittelijoiden kuulumiselle konsensusklusteriin.

Taulukko 2: Luokittelijoiden todennäköisyydet kuulua konsensusklusteriin, kun dataa on heikennetty poistamalla siitä satunnaisesti 5 % ja 10 %.

	Luokittelijat								
Pois	1	2	3	4	5	6	7	8	9
5 %	0,74	0,94	0,02	1,00	0,95	1,00	0,02	0,01	0,02
10 %	0,71	0,90	0,01	0,96	0,94	0,96	0,02	0,01	0,02

10

Diskussio

4 Cs -malli ilman osiota *arvioi kriittisesti* näyttäisi olevan toimiva perusta ensimmäisen vuoden sairaanhoitajaopiskelijoiden matematiikan kokeiden virheluokittelulle. Tätä johtopäätöstä tukee se, että näin laadittu luokittelumalli on luotettava.

Luokittelumallin validius saadaan siitä, että se on rakennettu erityisesti lääkelaskennan opetukseen laaditun pedagogisen mallin pohjalta. Reliabiliteettia puolestaan tukee, että k-means-luokittelussa syntyy tyypillisesti vakaa konsensusklusteri yksinkertaisella enemmistöllä. Luokittelujärjestelmä on myös stabiili. Tätä havaintoa tukee se, että konsensusklusteria ei saatu hajoitettua helposti edes poistamalla satunnaisesti 10 % datasta.

Tämä tulos on kuitenkin vain ensimmäinen havainto ja toteutus tällaisesta analyysista, joten paljon mielenkiintoisia kysymyksiä jää ilmaan niin analysoinnin kuin tulosten tulkitsemisen suhteen.

Luokittelijoille annettiin mahdollisuus ehdottaa opiskelijan tekemää jatkovirhettä. Tätä tietoa ei hyödynnetty tässä klusterointianalyysissa muuten, kuin että sen toivottiin vähentävän *ei voida luokitella* -luokan käyttöä. Tämä herättää kuitenkin kysymyksen toisensa leikkaavista virheluokista: jotkut virheistä voisi perustellusti luokitella useampaan kuin yhteen 4 Cs -luokkaan. Tällöin valinta ensisijaisen tai jatkovirheen välillä on täysin luokittelijan subjektiivinen mielipide. Tämä tilanne avaa mahdollisuuden kehittää luokittelujärjestelmää lisäämällä uuden luokan, jonka avulla kyseiset rajatapaukset saataisiin johdonmukaisemmin yhden luokan jäseniksi. Mallin muokkaamiselle voisi siis olla tarvetta.

Luokkien lisääminen ei kuitenkaan ole itsetarkoitus, sillä 4 Cs -luokittelun etuna on luokkien vähäinen määrä. Mikäli luokittelujärjestelmässä on paljon luokkia, se lisää vaikeutta löytää virheelle luokkaa. Luokittelujärjestelmästä on helppo tehdä liian monimutkainen käyttää ja ymmärtää.

Luokka *ei voida luokitella* on erityisen mielenkiintoinen. On odotettavissa, että sen syvempi tutkiminen johtaa erilaisten rakenteiden löytämiseen. Jatkotutkimus kuitenkin edellyttää lisää tutkimusaineistoa eli suurempaa luokittelijoiden lukumäärää m. Uuden luokan lisäämiselle tulee olla vahvat perustelut. Esimerkiksi seuraavissa tapauksissa uusi luokka voisi tulla kyseeseen: 1) opiskelija on suorittanut tehtävän oikein, mutta ei ole onnistunut poimimaan ratkaisun tiedoista oikeaa vastausta lopulliseksi vastaukseksi; 2) opiskelijan suoritus etenee täysin oikein, mutta se on lopetettu kesken ilman mitään näkyvää syytä; ja 3) vastauksessa ei ole mitään järkeä, mikään osa siitä ei liity annettuun tehtävään. Tässä koejärjestelyssä ei kuitenkaan esiintynyt sellaisia virheitä, joiden luokittelu olisi ollut mahdotonta käytetyllä kategorisointijärjestelmällä.

Koejärjestelyssä oli muutamia mahdollisia häiriötekijöitä. Primäärinen aineisto oli aitoja opiskelijoiden koevastauksia, jotka sisälsivät kokeen korjanneen opettajan merkintöjä. Vaikka merkinnät oli peitetty, kaikkea ei ole mahdollista poistaa peittämättä itse vastausta. Kaikki tehtäväpaperit olivat käsinkirjoitettuja, joten käsialan tulkinta saattoi olla haasteellista. Opiskelijoiden anonymiteetti oli kuitenkin suojeltu sikäli, että kaikki suorat viittaukset henkilöihin oli poistettu tai peitetty näkyvistä muuten kuin käsialan osalta.

Tässä tutkielmassa käsiteltiin perusteita, joilla arvioidaan 4 Cs -opetusmallin käyttämistä opiskelijoiden virheiden kategorisoinnin perustana. Tieto opiskelijoiden virheistä lääkelaskennassa antaa hyviä lähtökohtia käsitteellistää oppimisprosessia. Kirjalliset kokeet, kuten tutkimuksessa käytetty primääridata, voidaan kerätä opiskelijoilta harjoituksissa ja kokeissa, tai jopa oppimisympäristöjen tietokannoista.

Oppimisympäristöjen tulevaisuus tulee olemaan muuttuminen yhä enemmän ”oikean opettajan” kaltaisiksi. Ne arvioivat opiskelijoiden suorituksia henkilökohtaisesti ja yksilöllisesti, jolloin opetuksen etenemistä on mahdollista muuttaa ja säätää jokaiselle opiskelijalle erikseen. Tämän toteuttamiseen tarvitaan tekoälyn kaltaisia järjestelmiä, jotka käyttävät työkaluinaan esimerkiksi suoritusten ja virheiden luokittelua sekä suuresta tietomäärästä muodostettuja oppimisprofiileja.

Tässä tutkielmassa keskityttiin löytämään luokittelijoiden välistä yksimielisyyttä, eikä oltu kiinnostuttu missä ja millä tavoin luokittelijat olivat erimielisiä.

Hyvä aihe lisätutkimukselle on selvittää niitä kohtia aineistossa, jossa luokittelijoilla oli hankaluuksia löytää yksimielisyyttä, ja perehtyä erimielisyyden tilastollisiin rakenteisiin. Myös tähän tutkimukseen tarvitaan suurempi määrä dataa eli luokittelijoita. Yksimielisyyden puute voi olla merkki tarpeesta lisätä uusi luokka luokittelusääntöihin. Luokittelun lopullisen suunnittelun tulee ottaa huomioon päällekkäiset luokat. Tietoa näistä voidaan saada jatkovirheistä. Tätä ei tutkittu tässä tutkielmassa.

Automatisoitu virheluokittelu Stack-oppimisympäristössä tai vastaavassa edellyttää, että tehtävien rakenne ja tarkastuslogiikka on rakennettu siten, että myös ihminen pystyisi suoriutumaan tehtävän tarkastuksesta. Stackin tietokanta ei ole nykyisellään helposti käytettävissä luokittelun tarpeisiin. Tietojen esitystapa tuleekin suunnitella sellaiseksi, että sekä tietojärjestelmän että ihmisluokittelijan tuottaman aineiston tilastollinen analysointi on mahdollista samojen periaatteiden mukaisesti. Stack-harjoitusten tarkastusprosessi ei nykyisellään ole käyttökelpoinen 4 Cs -mallin mukaisen virheluokittelun toteuttamiseksi. Data nykyisin käytössä olevista Stack-harjoituksista ei sovellu 4 Cs -mallin mukaisen virheluokittelun automaattiseen toteuttamiseen.

Mallin neljäs C, *arvioi kriittisesti*, ei ollut mukana luokittelussa. Tämä kategoria voi olla varsin vaativa ihmisluokittelijalle, sillä se edellyttää tietämystä lääkkeiden annostuksesta. Vain lääkinnän ammattilainen voisi tehdä luokitteluja siihen luokkaan. Automaattinen luokittelu voisi tässä suoriutua ihmisluokittelijaa paremmin, sillä erilaisten lääkeaineiden vaaralliset annostukset ovat ennalta selkeästi tiedossa, ja ne voisi olla mahdollista ohjelmoida Stackillä toteutettuun automaattiseen luokittelujärjestelmään.

Osana 4 Cs -mallin opiskelijan toimintaan ja suoriutumiseen vaikuttavia tekijöitä on opiskelijan *minäpystyvyys* (Perceived Self-Efficacy, PSE), joka tulee Albert Banduran sosiaalisen oppimisen teoriasta (Johnson & Johnson 2002). PSE koostuu opiskelijan itseluottamuksesta ja käsityksestä omista kyvyistään sekä kyvystä oppia ja suorittaa tehtäviä. Lisäksi PSE kytketään oppimismotivaatioon ja kykyyn kysyä neuvoa vaikeissa tilanteissa. Motivoitunut opiskelija on valmis näkemään vaivaa ja tekemään työtä saavuttaakseen osaamista. PSE lisääntyy, kun oppilas kokee onnistumisen tunteita. Hodge (2002) havaitsi, että opiskelijoiden kyvyllä suoriutua lääkelaskentatehtävistä, minäpystyvyydellä matematiikassa ja tietokoneavusteisella opetuksella on merkittävä positiivinen yhteys. On myös muita viitteitä siitä, että nykyisen kaltainen tietokoneavusteinen opetus on opiskelijoita motivoivaa (Majander & Rasila 2011). Yhdistämällä näitä opetusmenetelmiä automatisoituun tulosten ja opiskelun etenemisen analysointiin voidaan luoda entistä motivoivampia oppimisympäristöjä.

Stack on soveltuva alusta monenlaisten matemaattisten tehtävien laatimista varten (Sangwin 2013; Rasila ym. 2007). Jo nykyisen kaltaisen Stackin käyttö opetuksessa parantaa oppimistuloksia (Rasila ym. 2010). Jos tähän lisätään esimerkiksi 4 Cs -malliin perustuvan virheluokittelun kaltainen analysointi, järjestelmällä on mahdollista toteuttaa erittäin käyttökelpoinen oppimisympäristö lääkelaskennan opettamiseen.

Kaikki opiskelijat ovat erilaisia matematiikan taitojen, oppimis- ja työskentelytyylien ja motivaation osalta. Kaikki voivat kuitenkin hyötyä innovatiivisista ja motivoivista opetus- ja oppimismenetelmistä, joita tieto- ja viestintätekniikan avulla on mahdollista toteuttaa. Opetuksen tulisi tunnistaa opiskelijoiden yksilölliset erot siinä, miten he ymmärtävät annetun tehtävän ja lähtevät etsimään oikeaa ratkaisua. Oppimisympäristön tulisi pystyä tukemaan opiskelijaa tässä prosessissa. Sigma ja Stack ovat järjestelmiä, joilla tällaisia oppimisympäristöjä on mahdollista rakentaa. Tutkimus ja kehitys niiden ympärillä onkin tällä hetkellä aktiivista (Leikas ym. 2012; Sangwin 2013).

Oppimisympäristöjen lisääntyvällä automatisoinnilla ei ole kuitenkaan mahdollista ratkaista kaikkia matematiikan opetukseen liittyviä kysymyksiä. Lääkelaskennassa pyritään täydelliseen osaamiseen. Miten voitaisiin verrata konseptuaalisia virheitä huolimattomuusvirheisiin, kun molemmat ovat tässä kontekstissa vakavia.

Entä jos opiskelija ei yksinkertaisesti halua oppia, tai panikoi ja haluaa vain päästä pois tilanteesta vastaamalla jotain. Ehkä oppimisympäristö voisi olla apuna tunnistamassa näitä tilanteita, jotta niistä päästäisiin eteenpäin esimerkiksi lisäämällä opettajan antamaa tukea.

Kun opiskelijoiden suorituksia on mahdollista analysoida automaattisesti sähköisten oppimisympäristöjen osana, on mahdollista koota hyvin laajoja tietokantoja opiskelijoiden suorituksista ja vastauksista. Tämä mahdollistaa erilaisten tiedonlouhinnan ja oppimisen analysointimenetelmien tehokkaan käytön. Tällöin nousevat esille myös erilaiset tietosuojaan liittyvät kysymykset. Opiskelijoiden on oltava hyvin perillä siitä, millaisia tietoja järjestelmä kerää ja mihin tietoja käytetään. Tietosuojaan liittyvä järjestelmän läpinäkyvyys onkin mietittävä huolellisesti, kun tulevaisuuden oppimisympäristöjä suunnitellaan. Tässä tutkielmassa tehty anonyymi tiedonkeruu on hyväksytty Arcadan eettisessä lautakunnassa. □

Lähteet

Bochner, S. 1981. The Role of Mathematics in the Rise of Science. Princeton: Princeton University Press.

Bock, H. H. 2008. Origins and extensions of the means algorithm in cluster analysis. Electronic Journ@l for History of Probability and Statistics 4 (2). http://www.jehps.net/Decembre2008/Bock.pdf

Cox, L. S. 1975. Diagnosing and remediating systematic errors in addition and substraction computations. Arithmetic Teacher 22 (2), 151–157.

Dekker, S. 2007. Just culture. Balancing Safety and Accountability. Aldershot: Ashgate Publishing.

Devlin, K. 2008. What will count as mathematics in 2100? Teoksessa B. Gold & R. Simmons (eds.) Proof and Other Dilemmas: Mathematics and Philosophy. Washington, DC: TheMathematical Association of America, 291–311.

Devlin, K. 2011. Mathematics Education for a New Era. Video Games as a Medium for Learning. Natick, MA: A K Peters.

Diamond, J. 1994. Zebras and the Anna Karenina principle. Natural History 103 (9), 4–10.

Diamond, J. M. 1997. Guns, germs, and steel: the fates of human societies. New York, NY: W. W. Norton.

Grandell-Niemi, H., Hupli, M., Leino-Kilpi, H. & Puukka, P. 2003. Medication calculation skills of nurses in Finland. Journal of Clinical Nursing 12 (4), 519–528.

Grandell-Niemi, H., Hupli, M., Puukka, P. & Leino-Kilpi, H. 2006. Finnish nurses' and nursing students' mathematical skills. Nurse Education Today 26 (2), 151–161.

Haapasalo, L. 2004. Pitääkö ymmärtää voidakseen tehdä vai pitääkö tehdä voidakseen ymmärtää? Teoksessa P. Räsänen, P. Kupari, T. Ahonen & P. Malinen (toim.) Matematiikka – näkökulmia opettamiseen ja oppimiseen. Jyväskylä: Niilo Mäki Instituutti.

Harjula, M. 2008. Mathematics exercise system with automatic assessment. Teknillinen korkeakoulu. Elektroniikan, tietoliikenteen ja automaation tiedekunta. Automaatio- ja systeemitekniikan diplomityö. http://urn.fi/URN:NBN:fi:aalto-201306116486

Hodge, J. E. 2002. The effect of math anxiety, math self-efficacy, and computer assisted instruction on the ability of undergraduate nursing students to calculate drug dosages. Väitöskirja. West Virginia University.

Huhtala, S. 2000. Lähihoitajaopiskelijan oma matematiikka. Väitöskirja. Helsingin yliopisto. Opettajankoulutuslaitoksen tutkimuksia 219.

Johnson, S. A. & Johnson, L. J. 2002. The 4 Cs: A model for teaching dosage calculation. Nurse Educator 27 (2), 79–93.

Joutsenlahti, J. 2005. Lukiolaisen tehtäväorientoituneen matemaattisen ajattelun piirteitä. Väitöskirja. Tampereen yliopisto. Acta Universitatis Tamperensis 1061.

Kilpatrick, J., Swafford, J. & Findell, B. (eds.) 2001. Adding It Up: Helping Children Learn Mathematics. Mathematics Learning Study Committee, Center for Education, Division of Behavioral and Social Sciences and Education, National Research Council. Washington, DC: National Academy Press.

Kivelä, S. K. & Spåra, M. 2001. Tietokoneet, ohjelmistot ja verkot matematiikan opetuksen välineinä. Teoksessa A. Ahtineva (toim.) Tutkimus kouluopetuksen kehittämisessä. Matematiikan ja luonnontieteiden opetuksen tutkimuksia. Turun yliopiston kasvatustieteiden tiedekunnan julkaisusarja C:17. Turku: Turun opettajankoulutuslaitos, 88–101.

Kutateladze, S. S. 2006. Apology of Euclid. Scientiae Mathematicae Japonicae/Notices from the ISMS 4 (5), 1–3.

Leikas, S., Granberg, L., Ståhl, T., Kurko, T., Antikainen, O., Airaksinen, M. & Pohjanoksa-Mäntylä, M. 2012. Sigma- lääkelaskennan oppimisympäristö: kehittämistarpeet ja hyödyntämismahdollisuudet farmasian peruskoulutuksessa. Farmaseuttinen aikakauskirja Dosis 28 (2), 106–117.

Linnoinen, K. 2013. Does practice make perfect? A study of the Granger-causal relationship between attempting to solve online exercises and mathematical proficiency. Helsingin yliopisto. Matemaattis-luonnontieteellinen tiedekunta. Soveltavan matematiikan pro gradu -tutkielma. http://hdl.handle.net/10138/42205

Lloyd, S. P. 1982. Least squares quantization in PCM. IEEE Transactions on Information Theory 28 (2), 129–137.

MacQueen, J. 1967. Some methods for classification and analysis of multivariate observations. Teoksessa Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, Volume 1. Berkeley, CA: University of California Press, 281–297.

Majander, H. & Rasila, A. 2011. Experiences of continuous formative assessment in engineering mathematics. Teoksessa H. Silfverberg & J. Joutsenlahti (toim.) Tutkimus suuntaamassa 2010-luvun matemaattisten aineiden opetusta. Matematiikan ja luonnontieteiden opetuksen tutkimuksen päivät Tampereella 14.–15.10.2010. Tampereen yliopisto. Kasvatustieteiden yksikkö. Tampere: Juvenes Print, 197–214.

MathJax Consortium. 2015. MathJax: Beautiful Math in All Browsers. https://www.mathjax.org [luettu 29.5.2015]

McMullan, M., Jones, R. & Lea, S. 2009. Patient safety: numerical skills and drug calculation abilities of nursing students and registered nurses. Journal of Advanced Nursing 66 (4), 891–899.

Movshovitz-Hadar, N., Zaslavsky, O. & Inbar, S. 1987. An empirical classification model for errors in high school mathematics. Journal for Research in Mathematics Education 18 (1), 3–14.

Mulhern, G. 1989. Between the ears: Making inferences about internal processes. Teoksessa B. Greer & G. Mulhern (eds.) New Directions in Mathematics Education. New York, NY: Routledge.

Pasternak, A. 2006. Hoitovirheet ja hoidon aiheuttamat haitat. Duodecim 122 (20), 2459–2470.

Pehkonen, E. 2009. Opetuksen arviointi ja kehittäminen. Luentomoniste. Helsingin yliopiston soveltavan kasvatustieteen laitos.

Piaget, J. 1929. The Child's Conception of the World. London: Routledge & Kegan Paul.

Radatz, H. 1979. Error analysis in mathematics education. Journal of Research in Mathematics Education 10 (3), 163–172.

Radatz, H. 1980. Students' errors in the mathematical learning process: a survey. For the Learning of Mathematics 1 (1), 16–20.

Rantanen, K. 2013. Numerot kiinni. Me tarvitsemme matematiikkaa. Mutta miten oppisimme sitä? Suomen Kuvalehti 97 (35), 44–49.

Rasila, A. 2008. Automaattisesti tarkastettavat tehtävät matematiikan opetuksessa. Teoksessa J. Viteli & S. Kaupinmäki (toim.) Tuovi 5: Interaktiivinen tekniikka koulutuksessa 2007 -konferenssin tutkijatapaamisen artikkelit. Tampereen yliopisto. Hypermedialaboratorion verkkojulkaisuja 15, 27–32. http://urn.fi/urn:isbn:978-951-44-7202-2

Rasila, A., Harjula, M. & Zenger, K. 2007. Automatic assessment of mathematics exercises: Experiences and future prospects. Teoksessa A. Yanar & K. Saarela-Kivimäki (toim.) ReflekTori 2007. Tekniikan opetuksen symposium 3.–4.12.2007. Teknillisen korkeakoulun Opetuksen ja opiskelun tuen julkaisu 1/2007, 70–80.

Rasila, A., Havola, L., Majander, H. & Malinen, J. 2010. Automatic assessment in engineering mathematics: evaluation of the impact. Teoksessa E. Myller (toim.) ReflekTori 2010. Tekniikan opetuksen symposium 9.–10.12.2010. Aalto-yliopisto. Teknillinen korkeakoulu. Dipoli-raportit B 2010:1, 37–45. http://urn.fi/URN:ISBN:978-952-60-3478-2

Røykenes, K. & Larsen, T. 2010. The relationship between nursing students' mathematics ability and their performance in a drug calculation test. Nurse Education Today 30 (7), 697–701.

Sangwin, C. 2013. Computer Aided Assessment of Mathematics. Oxford: Oxford University Press.

Sangwin, C. J. 2015. Who uses STACK? A survey of users of the STACK CAA system. Raportti. Loughborough University. https://dspace.lboro.ac.uk/2134/18540

Sangwin, C. J. & Ramsden, P. 2007. Linear syntax for communicating elementary mathematics. Journal of Symbolic Computation 42 (9), 920–934.

Sheriff, K., Wallis, M. & Burston, S. 2011. Medication calculation competencies for registered nurses: a literature review. Australian Journal of Advanced Nursing 28 (4), 75–82.

Skemp, R. R. 1976. Relational understanding and instrumental understanding. Mathematics Teaching 77, 20–26.

Skemp, R. R. 1987. The Psychology of Learning Mathematics. Expanded American Edition. New York, NY: Routledge.

Sosiaali- ja terveysministeriö. 2006. Turvallinen lääkehoito. Valtakunnallinen opas lääkehoidon toteuttamisesta sosiaali- ja terveydenhuollossa. Sosiaali- ja terveysministeriön oppaita 2005:32. http://urn.fi/URN:NBN:fi-fe201504226645

Steinhaus, H. 1957. Sur la division des corps matériels en parties. Bulletin de l'Académie Polonaise des Sciences 4 (12), 801–804.

Stickdorn, M. & Schneider, J. (eds.) 2011. This is Service Design Thinking: Basics – Tools – Cases. Hoboken, NJ: Wiley.

Tall, D. 2008. The transition to formal thinking in mathematics. Mathematics Education Research Journal 20 (2), 5–24.

Tella, S. & Lavonen, J. 1995. Tutkielma – oppimisen oiva osoitus. Opas tutkielman tekoon ja raportointiin. Helsingin yliopiston opettajankoulutuslaitos. Studia pædagogica 4. http://www.helsinki.fi/%7Etella/stpaed4.pdf

Tiitu, H. & Rasila, A. 2014. »Kieleni rajat ovat maailmani rajat». Oppimisympäristön käytettävyys luomassa matemaattista kieltä. Teoksessa A.-S. Röj-Lindberg, L. Burman, B. Kurtén-Finnäs & K. Linnanmäki (toim.) Spaces for learning: past, present and future. Proceedings of the FMSERA 30th annual symposium in Vaasa, November 6–8, 2013. Åbo Akademi. Report from the Faculty of Education No 36/2014. Vaasa: Arkmedia, 207–223.

Tilastokeskus. 2013. Tieliikenneonnettomuustilasto. Liitetaulukko 1. Liikenteessä kuolleet ja loukkaantuneet 1995–2012. Helsinki: Tilastokeskus. http://www.tilastokeskus.fi/til/ton/2012/ton_2012_2013-06-18_tau_001_fi.html [luettu 19.5.2014]

Толстой, Л. Н. 1963. Анна Каренина. Ленинград: Художественная литература. Alkuperäisjulkaisu 1875.

Wright, K. 2006. Barriers to accurate drug calculations. Nursing Standard 20 (28), 41–45.