Paikkatiedon laatutiedon automaattinen tuottaminen ja visualisointi

Esimerkkinä Digiroad-aineisto

Jenni Autere

Maantiede

Pro gradu -tutkielma

Laajuus: 30 op

Ohjaaja: Niina Käyhkö

28.5.2024

Turku

Pro gradu -tutkielma

Pääaine: Maantiede
Tekijä: Jenni Autere
Otsikko: Paikkatiedon laatutiedon automaattinen tuottaminen ja visualisointi – Esimerkkinä Digiroad-aineisto
Ohjaaja: Niina Käyhkö
Sivumäärä: 54 sivua + liitteet 21 sivua
Päivämäärä: 28.5.2024

Paikkatiedon kasvava saatavuus ja rinnakkaiset aineistot lisäävät yhä enemmissä määrin tarvetta paikkatiedon laadun arvioimiselle. Tietoaineistojen koon kasvaessa laadun manuaalinen arviointi käy yhä hankalammaksi, jolloin automaattisista menetelmistä muodostuu hyödyllinen osa arviointiprosessia. Tutkielmassa kehitetään automaattinen, Python-ohjelmointikieleen perustuva paikkatiedon laadun arviointi- ja visualisointimenetelmä, jota testataan joukkoliikenteen pysäkkitietoihin. Joukkoliikenteen pysäkkitiedot saadaan Digiroadista, joka on Väyläviraston ylläpitämä kansallinen tie- ja katuverkon tietojärjestelmä.

Tutkielmassa arvioidaan joukkoliikennepysäkkien sisällöllistä täydellisyyttä, alueellista täydellisyyttä, ajallista tarkkuutta sekä sijaintitarkkuutta. Alueellisen täydellisyyden ja sijaintitarkkuuden arvioimiseen käytetään referenssiaineistona avoimesta OpenStreetMap-paikkatietopalvelusta saatavia pysäkkitietoja. Laatutietojen tuottamisen jälkeen ne visualisoidaan tilastollisin kuvaajin sekä Suomen kuntajakoon pohjautuvasti interaktiivisella web-kartalla.

Pysäkkitiedon laatu on Suomessa lähtökohtaisesti hyvällä tasolla. Alueellista vaihtelua on erityisesti sisällöllisessä täydellisyydessä sekä ajallisessa tarkkuudessa. Digiroadissa olisi tärkeää kehittää tietojen sisältöä sekä tiedon laadusta viestimistä käyttäjän näkökulmasta hyödyllisellä tavalla. Esimerkiksi tarkat sijaintitiedot sekä aikaleima pysäkkitiedon viimeisimmästä tarkastusajankohdasta olisivat tarpeellisia kehityskohteita. Lisäksi Digiroadissa kannattaisi pyrkiä kehittämään erillisiä laatutietokuvauksia metatietokuvauksien ohelle huomioiden sekä ulkoiset että sisäiset laatutekijät.

Python-ohjelmointikieli soveltuu pistemäisen paikkatiedon automaattisen laadun arviointimenetelmän kehittämiseen hyvin, sillä se on nopea, monipuolinen ja helposti saavutettava kieli. Laadun arvioinnin automatisointi kokonaan tai osittain nopeuttaa arviointiprosessia, vähentää inhimillisten virheiden riskiä, ja mahdollistaa suurten tietoaineistojen kattavan laadun arvioinnin helposti.

Avainsanat: paikkatiedon laatu, laatutieto, automaattinen laadun arviointi, Python, Digiroad, pysäkkitieto

Master's thesis

Subject: Geography
Author: Jenni Autere
Title: Automating and visualising the assessment of the geospatial data quality – case Digiroad
Supervisor(s): Niina Käyhkö
Number of pages: 54 pages + appendixes 21 pages
Date: 28.5.2024

The increasing availability of geospatial data and parallel datasets are significantly raising the need for assessing the quality of geospatial data. In the era of big data, manual quality assessment becomes increasingly difficult, making automated methods a useful option to be utilized in the evaluation process. This thesis pursues to develop an automatic data quality assessment and visualization method based on the Python programming language. The method is then tested on public transportation stop data. The public transportation stop data is sourced from Digiroad, a national road and street network information system maintained by the Finnish Transport Infrastructure Agency.

Within this thesis the attribute completeness, spatial completeness, temporal accuracy, and positional accuracy of public transportation data is evaluated. For assessing spatial completeness and positional accuracy, reference data from the open geospatial service OpenStreetMap is utilized. After automatically generating the quality information, the quality information is then visualized using statistical charts and an interactive web map based on the Finnish municipal data.

In Finland, the quality of public transportation stop data is generally at a good level. However, there is some regional variation, particularly in attribute completeness and temporal accuracy of the stop data. It is crucial to develop the data content in Digiroad and communicate data quality better in a way that is useful from the user's perspective. For example, accurate positional information and timestamps of the most recent inspection of the stop information would be necessary areas for improvement. Additionally, there is a need to strive to develop separate quality descriptions alongside metadata in Digiroad, considering both external and internal quality factors.

The Python programming language is well-suited for developing an automated point-based geospatial data quality assessment method, as it is a fast, versatile, and easily accessible language. Automating quality assessment entirely or partially speeds up the evaluation process, reduces the risk of human-based errors, and makes it easy to assess the quality of datasets regardless of their size.

Key words: spatial data quality, quality information, automatic quality assessment, Python, Digiroad, public transport stop data

Sisällysluettelo

Johdanto

Paikkatietoa on yhä enemmissä määrin kaikkien saatavilla, ja sitä hyödyntävät niin julkisen vallan harjoittajat kuin yksityishenkilöt. Samalla paikkatietoa käytetään päätöksenteossa enemmän kuin aiemmin, mutta käyttäjät eivät välttämättä ole perillä käyttämänsä tiedon laadusta, mikä voi johtaa huonoihin tai harmillisiin päätöksiin. Siksi paikkatiedon soveltuvuuden arviointi erilaisiin käyttötarkoituksiin on tärkeää mahdollistaa. Arviointi kuitenkin on vaikeutunut samalla, kun tiedon tuottajien määrä on lisääntynyt ja tuotettu tieto on heterogeenisempää kuin aiemmin (Devillers ym. 2007). Laatutietojen tärkeys korostuu tiedon soveltuvuuden arvioinnissa. Hunter ym. (2009) listaavat paikkatiedon laaduntutkimuksen haasteiden liittyvän viiteen eri kokonaisuuteen: Laadun raportointiin, kuvailuun ja visualisointiin, virheen kasautumislaskentaan sekä laatutietojen käytännön soveltamiseen päätöksenteossa.

Metatiedot ovat osoittautuneet liian vähäpätöisiksi kertomaan käyttäjille paikkatiedon laadusta, joten laatutiedosta kommunikoimiseen tarvitaan parempia menetelmiä ja työkaluja. Metatiedot auttavat käyttäjiä arvioimaan tiedon soveltuvuutta, mutta ne koetaan käyttäjien toimesta usein hankaliksi ymmärtää, niiden rakenne on epäselväksi ja yhteydet niiden kuvaamaan tietoon puutteellisia, tai metatiedot saattavat puuttua osittain tai kokonaan (Devillers ym. 2007). Metatiedot ovat tärkeä osa tietoaineistoa ja siitä viestimistä käyttäjälle, mutta ne yksin eivät riitä tiedon soveltuvuuden arviointiin. Laatutietojen tuottaminen ja tietoaineiston laadusta viestiminen auttaa käyttäjiä ymmärtämään tietoaineiston rajoitteita paremmin, ja vielä suurempi hyöty saavutettaisiin silloin, jos laatutietoja hallittaisiin systemaattisesti ja yhdistäen paikkatietotyökaluihin (Devillers ym. 2005). Täysin automaattiset ratkaisut eivät kuitenkaan ole vastaus ongelmaan, sillä haasteet eivät ole ratkaistavissa pelkästään teknologialla, vaan tarvitaan myös syvempää ymmärrystä ja asiantuntijuutta paikkatiedon laadusta. Osittain automaattisille menetelmille on kuitenkin kasvava tarve, ja suurentuneet tietomassat ovat lisänneet tarvetta helppokäyttöisille ja saavutettaville web-pohjaisille työkaluille laatutietojen viestimisessä (Li ym. 2016).

Laadun käytettävyysnäkökulman myötä tiedon tuottajan odotetaan pikemminkin tarjoavan työkaluja käyttäjälle tiedon laadun arviointiin sen sijaan, että tiedon tuottaja vain kertoisi tiedon laadusta suhteessa ennalta määriteltyihin raja-arvoihin (Chrisman 2006: 25). Työkalut nähdään siksi tärkeinä, ettei nykyisessä digitaalisessa maailmassa ole mahdollista ennakoida kaikkia mahdollisia tiedon käyttötapoja.

Laatutietoja voidaan esittää esimerkiksi geovisualisoimalla, jolloin suuria määriä tietoa voidaan esittää tiiviissä ja käyttäjäystävällisessä muodossa. Visualisoimalla paikkatietoa ja siihen liittyviä tietoja, kuten laatutietoja, voidaan helpottaa esimerkiksi tiedon rakenteen tai sen sisältämien yhteyksien hahmottamista (Ge ym. 2008). Visuaaliset tiedon esitystavat palvelevat erityisesti muita kuin asiantuntijakäyttäjiä, joilla ei välttämättä ole tarpeeksi kyvykkyyttä esimerkiksi taulukkomuotoisen tiedon sisäistämiseen (Lush ym. 2012).

Joukkoliikenteen pysäkkitiedot ovat tärkeitä esimerkiksi joukkoliikennejärjestelmissä, joukkoliikenteen suunnittelussa sekä maksu- ja informaatiojärjestelmissä (Pysäkkitiedon hallinta Suomessa 2017). Lisäksi niitä hyödynnetään esimerkiksi joukkoliikenteen palvelutason määrittelyssä ja pysäkkien kunnossapidossa. Pysäkkitietojen käyttäjät ovat riippuvaisia sen laadusta, mutta laatua kuitenkin heikentävät muun muassa tiedon ylläpitäjien suuri määrä. Joukkoliikenteen pysäkkitiedot ovat olennainen osa matkatietoa, jonka kehittämistoimenpiteitä ohjaa kansallisen lainsäädännön lisäksi Euroopan Unionin asettama ITS-direktiivi tieliikenteen älykkäiden liikennejärjestelmien käyttöönoton sekä tieliikenteen ja muiden liikennemuotojen rajapintojen puitteista (2010/40/EY). Matkatiedon, eli siten myös pysäkkitietojen laatu on kytköksissä niin sosiaaliseen, ympäristön kuin taloudelliseenkin kestävyyteen matkatietoa hyödyntävien palveluiden kautta (Keski-Suomen laatupalvelupilotti 2022). Aiemmissa tutkimuksissa on huomattu heikkouksia muun muassa pysäkkien sijaintitiedoissa ja tietojen olemassaolossa (Keski-Suomen laatupalvelupilotti 2022). Matkahuollon toteuttamassa tutkimuksessa vuonna 2022 pysäkkitiedon laatua tutkittiin manuaalisesti tarkastelemalla, jolloin myös tutkimusalue oli rajoitettu Keski-Suomeen.

Tutkielman kohteena on Suomen kansallisen tie- ja katutietojärjestelmä Digiroadin sisältämä joukkoliikenteen pysäkki -aineisto, joka sisältää julkisen liikenteen käytössä olevat pysäkit Suomessa. Tutkielman tavoitteena on kehittää puoliautomaattinen menetelmä Digiroadin joukkoliikennepysäkkitiedon laadun arvioimiseen sekä laatutiedon tuottamiseen ja geovisualisointiin. Tutkimus pyrkii vastaamaan seuraaviin kysymyksiin:

1. Millainen on Digiroadin pysäkkien paikkatiedon laatu ja laadun alueellinen vaihtelu Suomessa?

2. Miten pysäkkitiedon laatutietoja voidaan arvioida automaattisesti ja esittää geovisualisoinnin keinoin?

3. Miten Digiroadin joukkoliikenteen pysäkkitiedon laatua voitaisiin parantaa?

1 Teoreettinen viitekehys

1.1 Paikkatieto ja tiedon laatu

1.1.1 Laadun moniulotteisuus

Tosimaailman täydellinen ja virheetön mallintaminen on tuskin koskaan mahdollista ihmisille, vaan luomamme mallit ja tietokokonaisuudet ovat aina alttiita virheille. Tästä syystä meillä on tarve tutkia tiedon laatua ja analysoida siinä olevia virheitä ja niiden juurisyitä. Yksinkertaisesti tiedon laatu voidaan käsittää sen ja tosimaailman välisenä eroavaisuutena (Devillers & Jeansoulin 2006: 33). Tiedon laatu on kuitenkin tosiasiallisesti paljon moniulotteisempi konsepti.

Paikkatiedon laadun tutkimus alkoi jo 1900-luvun alkupuolella, mutta noin 40 vuotta sitten laadun merkitys alkoi korostumaan kahdesta syystä: Paperikartoista alettiin siirtymään digitaalisiin karttoihin, ja samalla tieto avautui ja alkoi saavuttamaan uusia käyttäjäryhmiä internetin keksimisen myötä (Devillers & Jeansoulin 2006: 33). Paikkatieto sisältää kahta erilaista tietoa: Sijainti- ja ominaisuustietoa. Paikkatieto siis kuvaa asioita ja ilmiöitä, jotka sitoutuvat johonkin sijaintiin, ja näiden ominaisuuksia. Kuitenkin kehityksensä alkupuolella paikkatiedon laadun tutkimus oli yksinkertaista, ja se keskittyi lähes pelkästään sijaintitarkkuuden arviointiin paperisista kartoista (Chrisman 2006: 22). Myöhemmin tilastollisten menetelmien sisällyttäminen laatututkimuksiin alkoi laajentaa tutkimusalaa käsitteellisemmäksi ja moniulotteisemmaksi, ja lopulta digitaalisten karttojen syntyminen muutti alaa pysyvästi. Kun saatavilla olevan tiedon määrä kasvaa ja tiedon jakaminen helpottuu, myös mahdollisuudet tiedon väärinkäyttöön kasvavat. Tämän myötä laadun arvioinnin ja hallinnoinnin merkitys on korostunut.

Aluksi tiedon laadulla viitattiin lähinnä sisäiseen laatuun eli tiedon virheettömyyteen suhteessa tosimaailmaan, mutta myöhemmin rinnalle on vakiintunut myös toinen näkökulma, jossa laatu nähdään muuttuvaisena suhteessa käyttäjän laatimiin vaatimuksiin ja jossa laadulla pyritään kuvaamaan tiedon soveltuvuutta käyttäjän itse määrittelemään tarpeeseen (Devillers & Jeansoulin 2006: 37–39). Jälkimmäistä lähestymistapaa kuvataan usein englannin kielessä termillä *fitness for use*, jota terminä käytettiin ensimmäisen kerran 1970-luvulla (Juran ym. 1974). Nämä kaksi lähestymistapaa voidaan jakaa myös sisäiseen ja ulkoiseen laatuun, tai absoluuttiseen ja suhteelliseen laatuun (Devillers & Jeansoulin 2006: 36). Sisäinen ja ulkoinen laatu eivät kuitenkaan ole täysin toisistaan riippumattomia, vaan sisäinen laatu, eli virheiden

vähyys tietoaineistossa on osa ulkoista laatua. Sisäinen laatu voidaan myös kiteyttää siihen, kuinka hyvin tuotettu tieto vastaa tavoiteltua tietoa (Devillers & Jeansoulin 2006: 37). Ulkoinen laatu korostuu usein käyttäjille tärkeämpänä, mutta tiedon tuottajien nähdään keskittyvän enemmän sisäisen laadun varmistamiseen (Deitrick & Edsall 2008).

Laadulle on olemassa lukematon määrä erilaisia määritelmiä, mutta muutamat niistä ovat vakiintuneita. Määritelmissä usein ilmenee laadun kaksi näkökulmaa, sisäinen ja ulkoinen laatu, joista ensimmäisessä laadulla tarkoitetaan virheiden puuttumista ja jälkimmäisessä käyttäjätarpeiden täyttymistä. Esimerkiksi Devillers ym. (2005) määrittelevät laadun tietoaineiston ja käyttäjän määrittelemien tarpeiden yhteenpitävyytenä tiettyyn tarkoitukseen. Brus & Pechanecin (2014) määrittelevät laadun hyvin samaan tapaan: Laadukas tieto tarkoittaa tietoa, joka on oikeellista, luotettavaa ja riittoisaa käyttäjälle.

Usein tiedon laadusta puhuttaessa esille nousee myös epävarmuuden (engl. *uncertainty*) käsite. Epävarmuus ja laatu ovat kuitenkin eri asioita, ja epävarmuudella viitataan saavuttamattomaan tai ulottumattomissa olevaan tietoon esimerkiksi paikkatietoaineiston ja sen kuvaaman tosimaailman ristiriidoista (Deitrick & Edsall 2008). Epävarmuuteen vaikuttavat esimerkiksi monitulkinnallisuus, epäluotettavuus, epäjatkuvuus, heikko tarkkuus tai puutteelliset sisällöt (Brus & Pechanec 2014). Epävarmuus on käsitteenä laatua filosofisempi: Laadun käsite eriytyi epävarmuudesta samalla, kun digitaalinen edistyminen loi uudenlaiseen, käytännön tarpeen tiedon siirtämiselle (Fisher ym. 2006: 54). Epävarmuus, kuten laatukin, voi olla suhteellista ja vaihdella käyttäjän ja tiedon käyttötarkoituksen mukaan. Epävarmuus vaikuttaa laatuun ja sitä voidaan myös hyödyntää laadun arvioimisessa.

Nykyään sisäisen laadun katsotaan koostuvan pitkälti viidestä elementistä, joita ovat sisällöllinen tarkkuus, sijaintitarkkuus, ajallinen tarkkuus, täydellisyys sekä looginen eheys (Guptill & Morrison 1995). Nämä kriteerit ovat edelleen laajasti käytettyjä ja sisältyvät esimerkiksi paikkatiedon laatua ohjaavaan, kansainväliseen ISO 19157-standardiin (International Organization for Standardization 2013). Ulkoiselle laadulle ei ole vielä yhtä vakiintuneita kriteeristöjä, mutta esimerkiksi Bédard & Valliér (1995) tunnistavat kuusi ulkoisen laadun kriteeriä: Määritelmä, kattavuus, historiatieto, täsmällisyys, oikeellisuus ja saavutettavuus. Wang & Strong (1996) tunnistivat kyselytutkimuksensa avulla ulkoisen laadun elementeiksi myös historiatiedon, oikeellisuuden, täsmällisyyden ja saavutettavuuden, mutta myös uskottavuuden, objektiivisuuden, jalostusarvon, merkityksellisyyden, ajankohtaisuuden ja tulkinnallisuuden. Sijaintitarkkuus, historiatieto, semanttinen oikeellisuus, täydellisyys ja

looginen eheys muodostivat NCDCDS:n vuonna 1987 esittelemän määritelmän paikkatiedon laadulle (Moellering 1987). Laatukriteereihin lisättiin vielä ajallinen oikeellisuus sekä semanttinen eheys vuonna 1995.

Sijaintitarkkuudella viitataan paikkatiedon sisältämän sijaintitiedon oikeellisuuteen, eli siihen, kuinka hyvin sijainti vastaa todellisuudessa olevaa sijaintia. Sijaintitarkkuus on pisimpään tutkittu paikkatiedon laatuelementti. Sijaintitarkkuuteen vaikuttavat pääasiassa tiedon keruussa käytetyt menetelmät (Servigne ym. 2006: 188).

Semanttisella eli sisällöllisellä tarkkuudella tarkoitetaan ominaisuustietojen sekä niiden sisältämien yhteyksien oikeellisuutta. Semanttista tarkkuutta voidaan arvioida tutkimalla paikkatiedon ominaisuustietojen oikeellisuutta. Täydellisyydellä kuvataan paikkatietoaineiston sisältämiä ylimääräisiä ja puuttuvia kohteita, sekä kohteiden ominaisuustietojen puutteita. Lisäksi täydellisyydellä voidaan viitata paikkatiedossa esiintyvien yhteyksien olemassaoloon tai puutteellisuuteen (Servigne ym. 2006: 190).

Looginen eheys tarkoittaa tietorakenteen, ominaisuuksien ja niiden välisten suhteiden kykyä noudattaa loogisia sääntöjä. Looginen eheys jakautuu useille eri tasoille, joista keskeisin on topologia ja geometrinen eheys (Servigne ym. 2006: 192).

Paikkatiedon kuvaavat asiat ovat sitoutuneita paikan lisäksi aikaan. Paikkatiedon sisältämät sijainti- ja ominaisuustiedot muuttuvat ajassa, ja ajallisella tarkkuudella tarkoitetaan yleensä ajallisten ominaistietojen tarkkuutta sekä kohteiden ajallisten suhteiden oikeellisuutta (Yagoub 2017). Ajallisia ominaisuustietoja ovat esimerkiksi tiedon keräämisen tai julkaisun ajankohta, päivittämistiheys, viimeisin päivitysajankohta, tiedon ajantasaisuus sekä kohteiden väliset ajantasaisuustiedot (Fonte ym. 2017). Esimerkiksi aineiston sisältämä aikaleima kuvaa ajanhetkeä, jolloin tieto on vastannut todellisuutta, eli mitä vanhempi aikaleima, sitä huonompi ajankohtaisuus aineistolla voidaan sanoa olevan. Ajallinen tarkkuus on kytköksissä myös tiedon muihin laatuominaisuuksiin (Servigne ym. 2006: 194–195). Paikkatiedon ajallisuus voidaan jakaa kolmeen eri muotoon: Kaikilla kohteilla on tosiasiallinen aika, jolloin kohde on ollut olemassa todellisessa maailmassa. Tätä tietoa on usein vaikea kerätä, joten sen sijasta saatetaan ilmoittaa havaintoaika, joka kuvaa ajankohtaa, jolloin kohde havaittiin ensimmäisen kerran. Lisäksi joskus keskeistä voi olla toimintoaika, jolla viitataan ajankohtaan, jolloin tieto lisättiin tietokantaan tai tietoaineistoon. Käyttäjälle tärkein ajankohta on usein tosiasiallinen aika. Myös Krämer ym. (2007) määrittelee käyttäjän kannalta tärkeiksi ajallisiksi elementeiksi

aikamääreen tarkkuuden, viimeisimmän päivitysajankohdan, päivitystiheyden ja voimassaoloajan.

Aiemmin paikkatietoaineistoja tuottivat lähinnä viranomaiset. Nykyään viranomaisaineistojen ohelle on syntynyt uusia, rinnakkaisia paikkatietoaineistoja, joita tuottavat vapaaehtoiset yksityishenkilöt ja jotka kuvaavat samoja maantieteellisiä ilmiöitä tai kohteita viranomaisaineistojen kanssa. Käyttäjien on vaikeampi tulkita useista vaihtoehtoisista aineistoista se, mikä sopisi heidän käyttötarkoitukseensa parhaiten (Xavier ym. 2019, Devillers ym. 2007). Laatutietojen tärkeys on täten korostunut myös vapaaehtoisuuteen perustuvien paikkatietoaineistojen lisääntymisen myötä. Vapaaehtoisuuteen perustuvissa aineistoissa on seikkoja, jotka osaltaan saattavat parantaa tiedon laatua verrattuna viranomaisaineistoihin. Tällaisia ovat esimerkiksi paikallistuntemus, kun vapaaehtoiset henkilöt kartoittavat tuntemiaan alueita, sekä virheiden vähentyminen tilastollisessa mielessä, kun useat henkilöt keräävät tietoa samoilta alueilta. Vapaaehtoisaineistoissa onkin jo tunnistettu potentiaalia tukemaan viranomaisaineistojen laatua (esim. Sarretta & Mighini 2021, Khan & Johnson 2020, Peltonen 2016, Jackson ym. 2013). Vaihtoehtoiset paikkatietoaineistot ovat herättäneet keskustelua viranomaisaineistojen korkeista ylläpitokustannuksista, mutta Xavier ym. (2019) arvioi, että paikkatietoa tuottavat viranomaiset voisivat ottaa roolin laadun validoinnissa, tuottaen käyttäjille hyödyllistä ja standardoitua tietoa paikkatietoaineistojen laadusta. Myös viranomaispaikkatiedon laatua olisi tärkeää tutkia enemmän (Jackson ym. 2013).

1.1.2 Laadunhallinta ja standardit

Standardilla tarkoitetaan esimerkiksi yhteisiä toimintatapoja tiedon hankkimiselle, hallinnalle ja jakamiselle (Caprioli ym. 2003). Standardien kehittämisestä vastaavat yleensä kansainväliset ja kansalliset standardisointijärjestöt, jotka ovat kehittäneet useita erilaisia standardeja tukemaan paikkatiedon tuottamista ja paikkatietotuotteiden arvioimista. Paikkatiedon laatua ohjaavissa standardeissa on eroavaisuuksia, mutta pääpiirteittäin niiden sisältö koostuu samoista laadun kriteereistä. Standardeja hyödyntämällä paikkatiedon laatuarviointien tuloksista saadaan vertailukelpoisia, jolloin myös kaikista soveltuvin tietoaineisto voidaan helpommin löytää samankaltaisten aineistojen joukosta.

Yksi tunnetuimmista paikkatiedon laatua ohjaavista standardeista on kansainvälisen standardisointijärjestön (International Organization for Standardization, ISO) perustama ISO 19157 -standardi, jonka tarkoituksena on tarjota yhtenäinen menetelmä paikkatiedon laadun arviointiin (ISO 2013). ISO 19157 ei aseta paikkatiedon laadulle minimirajoja, vaan määrittelee

paikkatiedon laadun kuvaamisen osatekijät, tarjoaa erilaisia mittareita laadun arvioinnille, kuvailee millaisia toimintoja paikkatiedon laadun arviointi vaatii ja luo perustan paikkatiedon laadun raportoinnille. Kuten useimmat standardit, myös ISO 19157 on suositus. Chrisman (2006: 25) kritisoi ISO-standardeja siitä, että ne tarkastelevat laatua lähinnä tuottajan näkökulmasta, ja täten tiedon soveltuvuutta käyttötarpeeseen on vaikea arvioida niiden avulla. Tässä yhteydessä voidaan myös havaita kahtiajako tiedon tuottajien ja käyttäjien välillä: Tiedon tuottajat haluavat noudattaa standardeja, jotta he voivat saada tietotuotteilleen erilaisia sertifikaatteja, kun taas tiedon käyttäjät haluavat tiedon laadulta sitä, mitä heidän tavoitteensa ja tarpeensa vaativat (Servigne ym. 2006: 179).

Kansainvälisen standardisointijärjestön lisäksi Euroopan alueella toimii European Committee for Standardization (CEN). CEN toimii yhteistyössä ISO:n ja kansallisten standardisointijärjestöjen sekä muiden sidosryhmien kanssa, sillä standardisointia kehitetään ja toteutetaan nykyään pitkälti eri tahojen välisenä yhteistyönä (CEN and CENELEC s.a). CEN aloitti ensimmäisten virallisten paikkatiedon laatustandardien kehittämisen 1990-luvun alussa, mutta kehitys jäi ISO:n standardikehityksen alle ISO:n perustamisen myötä 1994 (Servegne et al. 2006: 199–200). Vuonna 2007 Euroopan unioni toimeenpani INSPIRE-direktiivin, jonka myötä myös CEN:in paikkatiedon laatustandardeja kehittänyt komitea elvytettiin. Elvytyksen tavoitteena oli vahvistaa ISO:n paikkatietostandardien asemaa ja lisätä käyttöä Euroopan unionissa. INSPIRE-direktiivin tarkoitus on ohjata Euroopan unionin paikkatietoinfrastruktuuria ja tukea erityisesti ympäristöön liittyvää päätöksentekoa sekä muita toimintoja (2007/2/EY). Lisäksi direktiivi tukee paikkatietoaineistojen yhdenmukaisuutta jäsenvaltioissa.

ISO:n ja CEN:n ohella paikkatietoon kytkeytyviä standardeja toteuttaa Open Geospatial Consortium eli OGC. OGC:n tavoitteisiin lukeutuvat yhteistyön lisääminen paikkatietoalalla sekä tiedon käytettävyyden lisääminen yli rajojen. Lisäksi OGC tavoitteisiin kuuluu paikkatietoteknologioiden kehityksen edistäminen, ja OGC:n standardeista useimmat keskittyvätkin erilaisiin paikkatiedon ja paikkatietojärjestelmien teknologioihin ja toteutusmuotoihin, kuten Web Map Service- ja Web Feature Service -rajapinnat (Servigne et al. 2006: 200). OGC:n standardit pohjautuvat ISO-standardeihin.

Suomessa paikkatietostandardeihin vaikuttaa standardien keskusjärjestö SFS ry, joka tekee yhteistyötä ISO:n ja CEN:in kanssa. SFS ry:n kautta suomalaiset voivat vaikuttaa kansainvälisiin standardeihin. Suomessa julkinen sektori pyrkii noudattamaan myös julkisen hallinnon suosituksia. Julkisen hallinnon suosituksista vastasi vuoteen 2020 asti julkisen hallinnon tietohallinnon neuvottelukunta JUHTA, mutta sittemmin neuvottelukunnan lakkauttamisen jälkeen Maanmittauslaitos on vastannut voimassa olevista paikkatietoalan kansallisista suosituksista (Paikkatietoalan standardit ja suositukset s.a). Samalla neuvottelukunnan lakkauttamisen yhteydessä lakkautettiin kansallinen paikkatiedon laatua ohjaava

suositus JHS 160 Paikkatiedon laadunhallinta, sillä sitä pidettiin vanhentuneena (Julkisen hallinnon tietohallinnon neuvottelukunnan työ päätökseen... 2019). Kuten useat muutkin paikkatietostandardit, myös JHS 160 perustui ISO-standardeihin (JHS 160 2006). Vuonna 2023 Valtionvarainministeriö lanseerasi julkishallinnon tiedon kansallisen laatukehikon, joka sisältää laadun arvioinnin kriteerejä ja mittareita ja jonka tarkoituksena on toimia tietoaineistojen laadun arvioinnin tukena (Tiedon laatukehikko s.a.). Kansallista laatukehikkoa voidaan soveltaa myös paikkatietoon.

Laadunhallinnan kannalta olisi tärkeää kehittää prosesseja ja työkaluja, jotka ottaisivat kantaa koko tiedon tuottamisen prosessiin (Du & Song 2015). Esimerkiksi laatukriteerien päällekkäisyys aiheuttaa sen, että joskus tiedossa esiintyvien virheiden luokittelu oikean kriteerin alle on haastavaa. Tällöin myöskään virheen lähdettä ei voida luotettavasti selvittää (Servigne ym. 2006: 196). Virheiden havaitseminen ja selvittäminen mahdollisimman varhaisessa vaiheessa vähentää niiden määrää lopullisessa tietoaineistossa ja vähentää samalla tiedon tuottamisprosessissa vaadittavien resurssien määrää (Westland 2002).

1.1.3 Metatieto laadunarvioinnissa

Metatieto ja laatutieto käsitetään joskus synonyymeinä, vaikka laatutieto voi pikemminkin olla osa metatietoa. Metatieto on tietoa tiedosta ja se kuvaa tietoaineiston ominaisuuksia, mutta ei välttämättä suoranaisesti sen laatua. Laatutiedoilla voidaan tarjota käyttäjälle paremmat mahdollisuudet arvioida aineiston käyttökelpoisuutta käyttäjän tarkoituksiin (Devillers ym. 2007). Laatutietojen merkitys on korostunut, kun paikkatiedon käyttäjäkunta on laajentunut asiantuntijoista ja tutkijoista myös muihin käyttäjiin, joilla ei välttämättä ole vaadittavaa asiantuntijuutta paikkatiedon laadun arviointiin. Hyödyn maksimoimiseksi laatutietoa on tärkeää kerätä ja säilyttää useilla eri tasoilla, aina kokonaisista aineistoista yksittäisiin tasoihin ja jopa kohteisiin (Devillers & Beard 2006: 237–242). Jos esimerkiksi tiedon sijaintitarkkuus vaihtelee suuresti aineiston sisällä, ei yksittäinen tieto sen tarkkuudesta ole riittävää tiedon soveltuvuuden arvioimiseksi. Käyttäjien kannalta on myös tärkeää, että käyttäjällä on mahdollisuus rajata ja suodattaa laatutietoja omien tarpeidensa ja vaatimustensa mukaan. Laatutiedot mahdollistavat sen, että käyttäjä kykenee helpommin tekemään tietoisen päätöksen käyttää tietoa hyväksyen sen epävarmuudet, tai vaihtoehtoisesti tulkita tiedon epäkelpoiseksi omaan tarkoitukseensa.

Käyttäjille tärkeää on kuvata myös laatutiedon laatua. Tätä kutsutaan metalaaduksi. Metalaatua ovat esimerkiksi tiedot laadunarvioinnin ajankohdasta, käytetyistä menetelmistä sekä perusjoukosta (Servigne ym. 2006: 185). Myös ISO 19157 -standardissa metalaatua kuvaaviksi

määreiksi on määritelty (käytettyjen arviointimenetelmien) luotettavuus, edustavuus (laatuarvioinnin laajuus) ja homogeenisuus (laatuarvioinnin tulosten yhdenmukaisuus). Homogeenisuuden arvioiminen vaatii sen, että paikkatiedon laatua on arvioitu otoksina, eli sitä ei voida arvioida silloin kun arviointi kohdistuu koko aineistoon.

Vaikka metatietojen on tarkoitus palvella käyttäjää, vain harva käyttäjä hakee paikkatietoja metatietojen perusteella (Timpf et al. 1996, Bielecka 2015). Metatietoja enemmän paikkatiedon etsimiseen ja soveltuvan aineiston valintaan hyödynnetään muiden käyttäjien suosituksia ja käyttäjälle itselleen tuttuja aineistontuottajia (Bielecka 2015). Metatiedot nähdään usein puutteellisina, jonka vuoksi käyttäjän on hankala arvioida paikkatietojen soveltuvuutta omaan käyttötarkoitukseensa niiden pohjalta. Bielecka (2015) kyselytutkimuksen perusteella käyttäjille tärkeimmiksi metatiedoiksi nousevat tiedon maantieteellinen sijainti ja kattavuus, temaattinen kattavuus, spatiaalinen resoluutio, jakelumuoto, aineiston saatavuus ja käyttörajoitukset sekä tiedosta vastaava organisaatio ja sen luotettavuus. Lisäksi historiatieto nähdään tärkeänä.

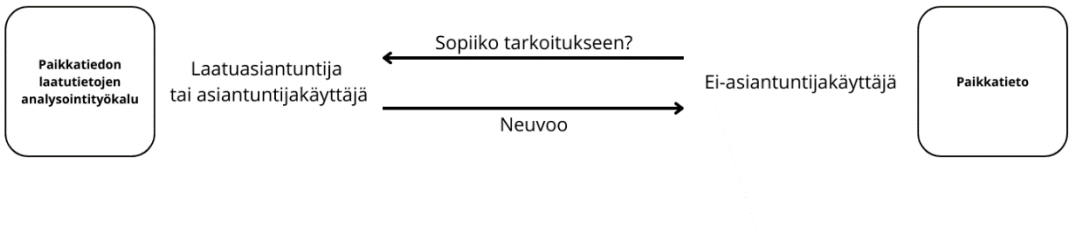
1.2 Digitaaliset ratkaisut paikkatiedon laadun tutkimuksessa

1.2.1 Laatuprosessien automatisointi

Automatisoinnin suosio on 2010-luvulta lähtien kasvanut geomatiikan aloilla, ja tiedon tuotantoketjua automatisoidaan yhä enenevissä määrin (Xavier ym. 2019). Oletettavaa on, että automatisointia tullaan hyödyntämään myös paikkatiedon laadunarvioinnissa ja -hallinnassa. Osittain automaattista laadunarviointia on testattu esimerkiksi WPS- eli Web Processing Service -pohjaisena (esim. Donaubauer ym. 2008, Mobasheri 2013, Xavier 2019). WPS on Open Geospatial Consortiumin standardisoima määritelmä rajapinnalle, jolla voidaan prosessoida paikkatietoa tietoverkon ylitse. Muita automaation kehittämisessä hyödynnettyjä työkaluja ovat esimerkiksi Python-ohjelmointikieli (esim. Sierra Requena 2023) ja PostgreSQL:n spatiaalista lisäosa PostGISiä (esim. Ďuračiová 2023).

Yksi varhaisimmista digitaalisista työkaluista on Devillers ym. (2007) kehittämä prototyyppi työkalusta, jolla voidaan hallita heterogeenisten paikkatietojen laatutietoja ja jonka toiminnoilla voidaan tukea asiantuntijoita aineistojen soveltuvuuden arvioinnissa niiden laatutietojen perusteella (kuva 1). Työkalun tarkoituksena on tukea ja osallistaa asiantuntijoita laadunarviointiprosessissa, jolloin he voivat paremmin tukea ei-asiantuntijoihin kuuluvia

loppukäyttäjiä, joilla ei välttämättä ole tarvittavia taitoja laadunarviointiin ilman asiantuntijatukea.



Kuva 1. Esimerkki laatutietotyökalun tai -järjestelmän asemasta aineiston soveltuvuuden arvioinnissa. Mukautettu Devillers ym. (2007).

Kasvavat tietomassat synnyttävät tarpeen laatuprosessien osittaiselle automaatiolle, sillä digitaalisetkin tietoaineistot ovat olleet pitkälti manuaalisen laadunarvioinnin ja laadunvarmistuksen varassa. Tiedon soveltuvuuden arvioimista käyttötarkoitukseen ei kuitenkaan ole mahdollista automatisoida täysin, sillä soveltuvuuden arviointi vaatii kognitiivisia taitoja ja syvempää ymmärrystä paikkatiedosta (Devillers ym. 2007). Automaatiota on kuitenkin mahdollista hyödyntää esimerkiksi sisäisen laadun arvioimisessa, joten automaatiolla on osansa laatutietojen tuottamisessa.

1.2.2 Geovisualisointi

Geovisualisoinnilla tarkoitetaan paikkatiedon visualisoimista erilaisin graafisin keinoin, kuten kartoilla tai kaavioilla. Paikkatiedolla on tärkeä asema päätöksenteossa, ja visualisointi nähdään hyvänä keinona paikkatiedon viestimiseen niin päätöksentekijöille kuin asiantuntijoille ja tutkijoille, sekä tavallisille kansalaisille, mikä tekee visualisoinnista tärkeän keinon paikkatiedon esittämiseen (Li ym. 2016). Paikkatiedon visualisoinnin merkitys on kasvanut samalla, kun tietomassat ovat suurentuneet. Suurten tietomassojen mukana ongelmaksi muodostuu usein se, etteivät perinteiset laskennalliset ja tilastolliset menetelmät riitä käsittelemään erittäin suuria tietomääriä (Li ym. 2016). Visuaalisissa menetelmissä laskennallisten menetelmien kyvykkyys yhdistyy ihmisen kyvykkyyteen tehdä visuaalisia tulkintoja, mikä luo perustan visualisoinnin hyödyllisyydelle. Visualisoinnissa ei siis ole kyse vain lopputuloksen viestimisestä muille, vaan se on myös tärkeä osa tiedon analysointivaihetta. Tietomassojen kasvaessa yhä suuremmiksi Li ym. (2016) näkevät visualisoinnin merkityksen kasvavan osana analyysia jopa suuremmaksi kuin se on tiedon viestinnässä.

Visualisoinnin avulla voidaan esittää suuria määriä moniulotteista tietoa, joten geovisualisointi nähdään hyvänä keinona laatutiedon esittämiseen (esim. Bielecka 2015, Ge ym. 2008, Goodchild & Clarke 2002). Ihmisillä on luontainen kyky hahmottaa nopeasti rakenteita ja yhteyksiä visuaalisesti havainnoimalla. Lisäksi kartat ja kaaviot ovat olennainen ja suora tapa esittää maantieteellistä tietoa. Visuaaliset menetelmät eivät kuitenkaan ole yksinkertainen ratkaisu suurten datamassojen käsittelyyn, sillä visualisoinnin haasteena on muun muassa informaatiotulva (Li ym. 2016). Informaatiotulvalla tarkoitetaan liiallisen tietomäärän esittämistä samanaikaisesti, jolloin ihmisen kognitiiviset kyvyt eivät enää riitä tiedon käsittelemiseen. Laatutietojen geovisualisoinnissa on myös graafisia haasteita, kuten laatutietojen ja niiden kuvaaman datan esittäminen rinnakkain (Devillers & Beard 2006: 242).

Laatutietojen visualisointi on haastavaa, mutta välttämätöntä käyttäjien kannalta, sillä visualisoinnilla voidaan helpottaa tiedon sisäistämistä sekä avustaa käyttäjää päätöksessään tiedon soveltuvuudesta käyttäjän tarkoituksiin (Lush ym. 2012). Visualisointi palvelee erityisesti muita kuin asiantuntijakäyttäjiä, sillä visuaalinen tieto nähdään usein helpommin ymmärrettävänä, kuin taulukkomuotoinen tieto. Laatutietojen visualisoinnin tutkimuksessa tulisi erityisesti hyödyntää loppukäyttäjien kokemuksia, jotta saadaan tietoa visualisoinnin vaikutuksista päätöksentekoon (Brus & Pechanec 2014). Visualisoinnin tutkimuksessa pyritään tunnistamaan uusia paikkatiedon graafisia esitystapoja ja niihin liittyviä epävarmuuksia sekä uusia yhteyksiä visualisoinnin ja paikkatiedon laadun välille.

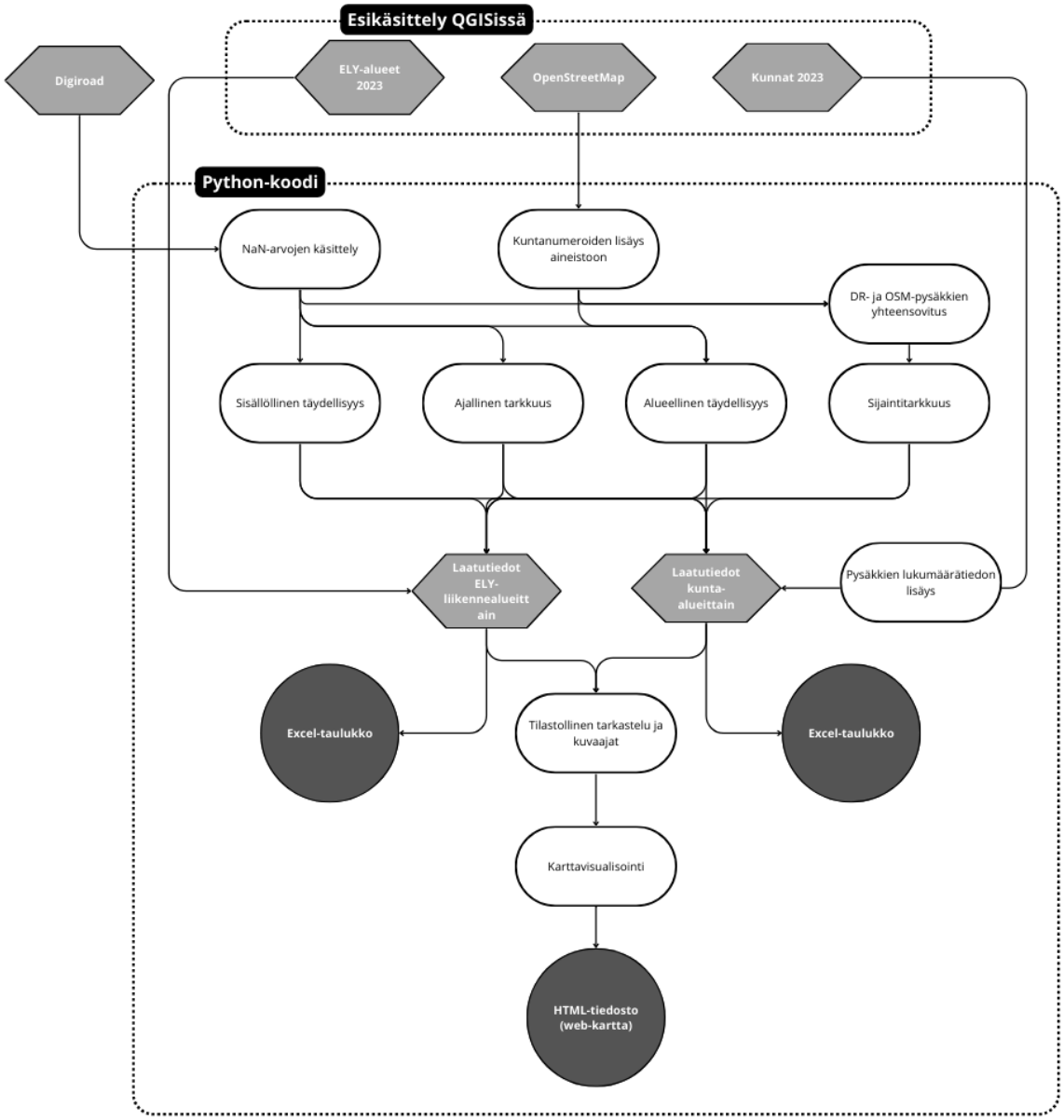
2 Aineistot ja menetelmät

2.1 Tutkimuksen asetelma

Tutkielmassa Digiroadin pysäkkiaineiston laatua tutkitaan automaattisella Python-ohjelmointikielellä tuotetulla laatuanalyysilla, joka koostuu neljästä laatutekijästä: Sisällöllinen täydellisyys, alueellinen täydellisyys, ajallinen tarkkuus ja sijaintitarkkuus. Nämä neljä laatutekijää on valittu siksi, että ne ovat korostuneet Väyläviraston tietojen käyttäjille tärkeiksi (Väylätiedon hyödyntäminen -selvitys 2021). Laatutekijöitä arvioidaan laatutietojen eli laatua kuvaavien arvojen avulla, jotka tutkielmassa tuotetaan ja visualisoidaan automaattisesti Python-ohjelmointikielellä. Tutkielman kohdeaineistona käytetään Joukkoliikenteen pysäkit -aineistoa, joka on saatavilla Väyläviraston ylläpitämässä Digiroad-tietojärjestelmästä.

Tutkittavista laatutekijöistä sijaintitarkkuuden ja alueellisen täydellisyyden tutkiminen vaatii referenssiaineistoa (Jackson ym. 2013). Referenssiaineistolla tarkoitetaan rinnakkaista tietoaineistoa, johon tutkittavaa aineistoa voidaan verrata. Tavanomaisesti viranomaisdataa on pidetty totuusaineistona, eli sitä usein käytetään referenssiaineistona erityisesti vapaaehtoisvoimin tuotetun paikkatiedon laadun tutkimuksessa (esim. Jackson ym. 2013, Giurres & Touya 2010). Koska rinnakkaista viranomaistietoa tai siihen rinnastettavaa tietoaineistoa ei ollut saatavilla tutkimukseen, käytetään referenssiaineistona OpenStreetMap-paikkatietopalvelun pysäkkitietoja. OpenStreetMapissa tieverkon muutokset voivat joskus näkyä aiemmin kuin muissa järjestelmissä (Keski-Suomen laatupalvelupilotti 2022).

Useissa aiemmissa paikkatiedon laadun tutkimuksissa visualisointi pohjautuu ruudukoihin, joilla voidaan esittää laadun alueellista vaihtelua (Borkowska & Pokonieczny 2022). Tässä tutkielmassa visualisointi kuitenkin pohjautuu Suomen kuntajakoon sekä ELY-keskusten liikenteen ja infrastruktuurin vastuualueisiin. Kunnat ja ELY-keskukset ovat pääsääntöisiä Digiroadin pysäkkitiedon tuottajia, jolloin hallinnollisiin rajoihin perustuva visualisointi on paremmin sovellettavissa laadun kehittämistyöhön (Pysäkkitiedon hallinta Suomessa 2017). Kaikki laatutiedot lasketaan tutkimuksessa samoin periaattein (kuva 2): Laskukaavat iteroidaan aineiston tai aineistojen läpi lisäten tulos aina listaan, ja lopullinen lista liitetään tilastointialueisiin, jolloin lopulta saadaan kaksi aineistoa: ELY-alueet 2023- sekä kunnat 2023, jossa jokaisella ELY-alueella ja kunnalla on niiden perustietojen lisäksi laatutietoja alueiden sisältämien joukkoliikenteen pysäkkitiedoista.



Kuva 2. Joukkoliikennepysäkkien laatutiedon puoliautomaattisen tuottamisen ja visualisoinnin yleistetty prosessi tutkielmassa.

Tutkielmassa laatutiedot tuotetaan ja visualisoidaan automaattisesti Python-ohjelmointikielellä (versio 3.10.12). Ohjelmointiympäristönä tutkielmassa käytetään Microsoft Visual Studio Code -ohjelmistoa (versio 1.81.1), joka on avoimen lähdekoodin tekstieditori. Automaattinen laatuanalyysi ja tietojen visualisointi toteutetaan Jupyter-työkirjana (.ipynb), jolloin Python-koodi voidaan rakentaa ja tarvittaessa myös suorittaa erillisissä osissa eli soluissa.

Python on yksi suosituimpia ohjelmointikieliä sen ollessa käytettävyydeltään laaja, mutta syntaksiltaan yksinkertainen ja helppo kieli. Sitä käytetään esimerkiksi data-analytiikassa, tehtävien automatisoinnissa sekä web-sovelluskehityksessä. Pythoniin sisältyy laaja

peruskirjasto, jonka moduuleja voidaan hyödyntää ohjelmoinnissa, minkä lisäksi Pythonille on kehitetty suuri määrä avoimen lähdekoodin paketteja ja moduuleja, jotka voidaan ottaa käyttöön asentamalla. Tässä tutkielmassa käytetään 8 eri pakettia ja moduulia (taulukko 1).

Taulukko 1. Tutkielmassa käytetyt Python-paketit ja -moduulit.

Kirjasto	Versio	Tarkoitus tutkielmassa
Pandas	2.2.0	Taulukkoaineistojen käsittely
Geopandas	0.14.3	(Paikkatieto-)taulukkoaineistojen käsittely
Shapely	2.0.3	Geometrian käsittely
Fiona	1.9.5	Paikkatiedon tiedostomuotojen käsittely
Numpy	1.26.4	Tiedon numeraalinen käsittely
Folium	0.15.1	Visualisointi
SciPy	1.12.0	Tilastollinen tarkastelu
Matplotlib	3.8.3	Kaavioiden luominen
Mapclassify	2.6.1	Luokitustyyppien määritys karttavisualisointiin

2.2 Digiroad-aineisto

Digiroad on Väyläviraston ylläpitämä kansallinen tie- ja katutietojärjestelmä, joka kuvaa Suomen tie- ja katuverkon keskilinjageometriaa sekä sen ominaisuustietoja (Digiroad: tietolajien kuvaus 2022). Digiroadin tietoaineistoista vastaavat Väyläviraston lisäksi useat eri toimittajat, kuten Maanmittauslaitos, ELY-keskukset, kunnat, tiekunnat ja muut toimivaltaiset viranomaiset. Digiroadin tietosisältö on avoimesti saatavilla ja hyödynnettävissä Creative Commons Nimeä 4.0 -lisenssillä. Digiroadin käyttäjiä ovat laajasti esimerkiksi viranomaiset, yritykset, järjestöt sekä kansalaiset. Digiroadin sisältämät tietoaineistot ovat tasokoordinaattijärjestelmässä ETRS-TM35FIN (Digiroad: tietolajien kuvaus 2022).

Tutkielmassa tutkimusaineistona käytetään Digiroadin sisältämää vektorimuotoista joukkoliikenteen pysäkit -aineistoa. Pistemäinen joukkoliikenteen pysäkit -aineisto koostuu linja-autopysäkeistä, virtuaalipysäkeistä sekä raitiovaunupysäkeistä, ja sisältää pysäkkien sijainti- ja ominaisuustietoja (liite 1). Pysäkkien sijaintikoordinaatit on Digiroadissa sidottu tien tai kadun keskilinjageometriaan, ja joillain kohteilla voi olla myös erilliset maastokoordinaatit, jotka kuvaavat pysäkin todellista sijaintia maastossa (Pysäkkitiedon hallinta Suomessa 2017). Joukkoliikenteen pysäkkejä on Suomessa noin 90 000. Joukkoliikenteen pysäkkien tiedoista vastaavat toimivaltaiset viranomaiset, joita ovat esimerkiksi kunnat ja ELY-keskukset.

Toimivaltaisia viranomaisia ovat myös seudullisen joukkoliikenteen toimijat, kuten Föli ja Helsingin seudun liikenne. Digiroadin pysäkkiaineistoa hyödynnetään useissa joukkoliikenteen digitaalisissa palveluissa, kuten valtakunnallisissa lippu- ja maksujärjestelmä Waltissa sekä liikennelupajärjestelmä Vallussa (Pysäkkitiedon hallinta Suomessa 2017). Lisäksi pysäkkiaineistoa hyödynnetään esimerkiksi pysäkkien kunnossapitoon ja joukkoliikenteen palveluiden suunnitteluun liittyvissä toiminnoissa. Pysäkkiaineiston käyttäjät ovat riippuvaisia tietojen laadukkuudesta ja jatkuvasta ylläpidosta.

Digiroadin pysäkkiaineisto on saatavilla useissa eri muodoissa. Väyläviraston Suomen Väylät -karttapalvelusta ajankohtaisin aineisto on ladattavissa Geopackage-, Shapefile-, CSV-, XLS- ja JSON-muodoissa, minkä lisäksi aineiston voi ladata Digiroadin neljästi vuodessa tehtävän erillisirroituksen mukana Geopackage- ja Shapefile-muodoissa. Lisäksi pysäkkiaineiston voi ladata erikseen GTSF-aineistona tai CSV-muodossa. Aineiston voi ottaa käyttöön myös WMS- ja WFS-rajapintojen kautta. Tässä tutkielmassa käytetään GeoPackage-muotoista aineistoa, joka on ladattu Suomen Väylistä (https://suomenvaylat.vayla.fi/) ELY-keskusten vastuualueittain 30.10.2023. Yhteensä aineistossa on 94 119 pysäkkikohdetta. Tutkielmaan ei sisällytetty Ahvenanmaan pysäkkejä.

2.3 OpenStreetMap -karttapalvelu

Tutkielmassa käytetään referenssiaineistona OpenStreetMapin joukkoliikenteen pysäkkitietoja. OpenStreetMap, lyhyemmin OSM on vapaaehtoiseen kartoittamiseen perustuva yhteisöprojekti, jonka tarkoituksena on tuottaa avointa ja saavutettavaa paikkatietoa esimerkiksi katuverkoilta. OpenStreetMapin aineistot ovat käytettävissä OpenStreetMap-säätiön omalla ODbL-lisenssillä. OpenStreetMapin pysäkkiaineisto ladattiin tutkielmaa varten QGISin QuickOSM-lisäosalla (versio 2.2.3). QuickOSM hakee OpenStreetMapin tiedot Overpass-rajapinnalta, josta pysäkkiaineisto tuodaan QGISiin siten, että aineisto sisältää pysäkkitiedot kokonaisuudessaan koko Suomen alueelta. QGISiin tuotu aineisto sisältää pistemäisten kohteiden lisäksi 13 aluemaista kohdetta sekä 21 viivamaista kohdetta, mutta manuaalisen tarkastelun jälkeen voidaan todeta, että näille kohteille löytyy vastaava kohde pisteaineistosta, eli kohteita ei tarvitse muuntaa pisteiksi ja ne voidaan poistaa tutkielman tarkoituksiin ladatusta aineistosta. Tutkielmassa käytetyt OSM-aineistot on ladattu 30.10.2023.

OSM-aineisto olisi mahdollista tuoda myös suoraan Jupyter Notebookiin OSMNX-paketilla, mutta aineisto on niin suuri, että algoritmin suorituskyky kärsii. Tästä syystä käytetään QGISin lisäosalla tuotua aineistoa.

2.4 Tilastoalueaineistot

Laatutietojen visualisoimiseksi tutkielmassa käytetään Suomen tilastointialueita, jotta laatutietoja voidaan koostaa alueittain. Tutkielmassa käytetään Tilastokeskuksen tilastointialueaineistoja ”ELY-alueet 2023 1:1 000 000” ja ”Kunnat 2023 1:1 000 000”. Aineistot ovat aluemaisia vektoriaineistoja koordinaattijärjestelmässä ETRS-TM35FIN. Kumpikin aineisto on ladattu Tilastokeskuksen karttapalvelusta (https://tilastokeskus-kartta.swgis.fi/) Shapefile-muodossa 13.7.2023. Osa ELY-keskuksista vastaa myös viereisten ELY-alueiden liikenteestä ja infrastruktuurista ja täten myös niillä alueilla sijaitsevista valtion omistamista joukkoliikenteen pysäkeistä, joten tutkielmassa huomioitavia ELY-alueita on 9 (taulukko 2).

Taulukko 2. Liikenteestä ja infrastruktuurista vastaavat ELY-keskukset ja niiden vastuulla olevat ELY-alueet.

ELY-keskus	Lyhenne	Vastuulla olevat ELY-alueet
Uusimaa	UUDELY	Uusimaa, Häme
Varsinais-Suomi	VARELY	Varsinais-Suomi, Satakunta
Pirkanmaa	PIRELY	Pirkanmaa
Kaakkois-Suomi	KASELY	Kaakkois-Suomi
Pohjois-Savo	POSELY	Pohjois-Savo, Etelä-Savo, Pohjois-Karjala
Keski-Suomi	KESELY	Keski-Suomi
Etelä-Pohjanmaa	EPOELY	Etelä-Pohjanmaa, Pohjanmaa
Pohjois-Pohjanmaa	POPELY	Pohjois-Pohjanmaa, Kainuu
Lappi	LAPELY	Lappi

2.5 Paikkatietoaineistojen esikäsittely

Ennen Python-ympäristöön tuomista OSM- ja tilastoalueaineistojen esikäsittely aloitetaan QGISissä. Osa esikäsittely tehdään QGISissä eikä Pythonilla siksi, että OSM-pysäkit ladataan käyttöön QGISin kautta, jolloin esikäsittelystä osa on helpompi toteuttaa samalla, ja tilastoalueaineistot esikäsitellään Pythonin ulkopuolella siksi, ettei Ahvenanmaan poisjättö ole pakollista algoritmin kannalta.

QuickOSM-haulla Suomeen rajatusta aineistosta poistetaan Ahvenanmaan pysäkit valitsemalla ensin Ahvenanmaan ELY-alue luvussa 2.4 Tilastoalueaineistot mainitusta ELY-alueet 2023 1:1 000 000-aineistosta ja rajaamalla tämän jälkeen poistettavat pysäkkipisteet OSM-aineistosta QGISin “Valitse sijainnin perusteella”-työkalua. Poiston jälkeen pistemäisiä kohteita on

aineistossa 93 310. Ennen tallentamista aineistolle tehdään projektiomuunnos QGISin ”Projisoi taso”-työkalulla OSM:n käyttämästä WGS84-koordinaattijärjestelmästä Digiroadin kanssa yhtenäiseen ETRS-TM35FIN-järjestelmään. Koordinaattijärjestelmä ei voi tutkimuksen tapauksessa olla WGS84, sillä WGS84 käyttää mittayksikkönä asteita, eikä tuota siten sovellettavaa mittatietoa esimerkiksi sijaintitarkkuutta tarkastellessa. (Jackson ym. 2013).

QuickOSM:n kautta QGISiin hakiessa OSM:n joukkoliikenneaineistoihin tulee useita kymmeniä attribuutteja, joissa suurimmassa osassa ei ole tietoja. Aineisto tallennetaan projektiomuunnoksen jälkeen GeoPackage-muotoon, ja tallentamisvaiheessa valitaan tallennettavaksi attribuuteiksi vain laatutietojen tuottamiseen käytettävät attribuutit: *full_id, osm_id, ref:findr, name:sv, name:fi* ja *name*. Tällöin tiedostojen lukeminen Pythonilla kestää huomattavasti vähemmän.

Tallentamisen jälkeen aineisto jaetaan ELY-aluekohtaisiin aineistoihin hyödyntämällä ELY-alueet 2023 1:1 000 000 –aineisto sen jälkeen, kun ELY-alueet on mukautettu vastaamaan tutkimuksen tarpeita. Aineisto jaetaan QGISin ”Valitse sijainnin perusteella”-työkalulla yhdeksään erilliseen tiedostoon, jotka sisältävät OSM:n joukkoliikennepysäkit ELY-keskusten liikennevastuualueittain. Tuloksena saadut ELY-aluekohtaiset aineistot tallennetaan GeoPackage-muotoon samalla tavoin kuin koko Suomen aineisto, eli tallennettaviksi attribuuteiksi valitaan *full_id, osm_id, ref:findr, name:sv, name:fi* ja *name*.

ELY-alueet 2023 1:1 000 000 -ainestoa käytetään ennen sen esikäsittelyä Ahvenanmaan pysäkkien poistamiseen OSM-aineistosta. Tämän jälkeen molemmista tilastoalueaineistoista poistetaan Ahvenanmaa QGISissä, sillä aineistot eivät olleet ladattavissa ilman Ahvenanmaata. Tämä muuttaa kuntien määrän kunta-aineistossa 309:stä 293:een. Lisäksi osa ELY-alueista yhdistetään toisiinsa (taulukko 2). Lopuksi aineistot viedään GeoPackage-muotoon tutkielman aineistojen yhtenäistämiseksi samaan tiedostomuotoon.

Loput esikäsittelyt tehdään kaikille aineistoille Pythonilla. Python-ympäristöön tuomisen jälkeen ELY-aluekohtaisille Digiroad-pysäkkiaineistoille lisätään tieto siihen liittyvästä ELY-alueesta (esim. liite 4, rivi 21), jonka jälkeen tuodut aineistot yhdistetään listaksi koodin tiivistämiseksi (liite 4, rivit 48–49). Tuoduissa Digiroad-aineistoissa tyhjät solut näyttäytyvät tyhjinä, mutta jotta ne voidaan huomioida esimerkiksi sisällöllisen täydellisyyden laskemisessa, tulee ne muuttaa NaN-muotoon (liite 4, rivit 51–56). Lisäksi aineistot yhdistetään yhdeksi erilliseksi taulukoksi (liite 4, rivit 58–62). Lopuksi kaikki Digiroad-pysäkit kattavasta

taulukosta muutetaan arvot 99 (”Ei tiedossa”) NaN-arvoiksi, jotta pysäkkien tietokohtaista sisällöllistä täydellisyyttä voidaan myöhemmin arvioida (liite 4, rivit 64–87).

Digiroad-pysäkkien jälkeen tuodaan myös tilastoalueaineistot, joihin analyysin tuloksena saatavat laatutiedot yhdistetään tietojen visualisoimiseksi. Tuomisen yhteydessä ELY-aineisto järjestetään ELY-keskuksen numeron perusteella, minkä jälkeen ELY-keskusten numerot ja kuntien kuntakoodit muutetaan kokonaisluvuiksi (liite 4, rivit 97-100). Lisäksi aineistoihin lisätään ELY-keskusten lyhenteet (liite 4, rivit 102–123). Lopuksi kunta-aineistoon lasketaan kunta-alueen sisältämien pysäkkien määrä Digiroad-aineistoista (liite 4, rivit 125–130), ja laatuanalyysin tarpeisiin luodaan lista kuntakoodeista kunta-aineiston pohjalta (liite 4, rivit 13–134).

OSM-pysäkkiaineistot tuodaan samalla tavoin kuin Digiroad-aineistot (liite 4, rivit 137–155). Kuten Digiroad-aineistot, myös OSM:n ELY-aluekohtaiset pysäkkitiedot yhdistetään listaksi. Lisäksi tuodaan koko Suomen pysäkkitiedot sisältävä OSM-aineisto.

OpenStreetMapin sisältämiin pysäkkitietoihin ei ole sisällytetty kuntakoodeja, kuten Digiroadissa. Koska kaikki kuntakohtaiset laatutietojen laskennat perustuvat tutkielmassa kuntakoodiin, tulee OSM-pysäkeille seuraavaksi lisätä kuntakoodit (liite 4, rivit 166–184). Pysäkit valitaan kunta-alueen geometrian perusteella, ja valitulle joukolle lisätään valinnan perusteena toimineen geometrian eli kunta-alueen kuntakoodi. Yhdistetty tieto lisätään uuteen taulukkoon *osmStops_munId*.

Sijainnin perusteella valitseminen jättää kuitenkin 40 kohdetta rajausten ulkopuolelle, sillä nämä 40 pysäkkiä eivät sisälly mihinkään geometriaan, eli pysäkki sijaitsee esimerkiksi vesistön ylittävällä sillalla. Ylimääräisille pysäkeille lisätään manuaalisesti kuntakoodi. Geometrioiden ulkopuolelle jäävät pysäkit etsitään avaamalla OSM-pysäkkiaineisto sekä Kunnat 2023-aineisto QGISissä, ja valitsemalla OSM-aineistosta ne pysäkkikohteet, jotka ovat erillään kunta-aineistosta käyttämällä ”Valitse sijainnin perusteella”-työkalua. Ylimääräisten pysäkkien kunta selvitetään pysäkin nimen perusteella hakemalla niitä Google Maps-palvelusta. Tämän jälkeen samaan kuntaan kuuluvien ylimääräisten pysäkkien valtakunnallinen ID (liite 2) lisätään listaan, ja käyttäen indeksointia ja ehtokomentoa ylimääräisille pysäkeille lisätään oikea kuntakoodi OSM-aineistoon (liite 4, rivit 192–234). Tämän jälkeen muokatut rivit lisätään uuteen, kuntakoodit sisältävään taulukkoon *osmStops_munId*. Lopuksi tarkastetaan, että kaikilla uuden taulukon pysäkeillä on kuntakoodi, ja muutetaan kuntakoodin

sisältävä sarake tyypiltään kokonaisluvuksi, jotta se on samassa muodossa muiden aineistojen sisältämien kuntakoodien kanssa (liite 4, rivit 241–245).

2.6 Laatuanalyysit

Digiroad-pysäkkiaineiston laatua tarkastellaan neljän laatutekijän perusteella, joita ovat sisällöllinen täydellisyys, alueellinen täydellisyys, ajallinen tarkkuus ja sijaintitarkkuus. Laatutekijöiden analysoimiseksi on valittu niitä kuvaavat laatumittarit (taulukko 3). Laatumittarit on valittu tutkielmaan niiden soveltuvuuden perusteella, eli ne soveltuvat pistemäisen paikkatiedon laadun tutkimiseen.

Taulukko 3. Laatuanalyysin rakenne, laatumittarit ja käytettävät menetelmät.

Laatutekijä	Mittari	Menetelmä	Esimerkkejä
Sisällöllinen täydellisyys	Attribuuttien täydellisyys	Tyhjien solujen suhde solujen lukumäärään	Bielecka (2015), Girres & Touya (2010)
Alueellinen täydellisyys	Ylimääräiset tai puuttuvat kohteet	Kohteiden lukumäärien vertailu suhdeluvuilla	Lee & Choi (2019), Jackson ym. (2013)
Ajallinen tarkkuus	Ajankohtaisuus	Viimeisimmästä muokkausajasta kulunut aika vuosina	Bielecka (2015)
Sijaintitarkkuus	Suhteellinen sijainti	Euklidinen etäisyys yhdisteltyjen pysäkkien välillä	Lee & Choi (2019), Girres & Touya (2010), Jackson ym. (2013)

Sisällöllinen täydellisyys ilmaisee aineiston sisältämien ominaisuustietojen kattavuutta. Tutkielmassa pysäkkikohteille lasketaan sisällöllinen täydellisyys, josta lasketaan keskiarvo kaikille kunta- ja ELY-alueille. Lisäksi tarkastellaan tietokohtaista sisällöllistä täydellisyyttä kaikkien Digiroad-pysäkkien osalta. Tietokohtaista tarkastelua ei tehdä alueellisesti tutkimuksen rajaamiseksi. Tietokohtainen tarkastelu tehdään vain sellaisille tiedoille, jotka eivät generoidu automaattisesti Digiroad-järjestelmässä (liite 1). Sisällöllistä täydellisyyttä arvioidaan joko puuttuvien tai täytettyjen tietojen suhteellisen osuuden perusteella kaikista tiedoista, eli jokaiselle kohteelle lasketaan täydellisyysprosentti kaavalla:

Sisällöllinen täydellisyys = 100 - tyhjät solut / kaikki solut * 100

tai

Sisällöllinen täydellisyys = solut, joissa tietoa / kaikki solut * 100

Jälkimmäistä laskutapaa käytetään tietokohtaisen sisällöllisen täydellisyyden laskemiseen teknisistä syistä. Sisällöllisen täydellisyyden laskemiseen ei tarvita referenssiaineistoa.

Sisällöllinen tarkkuus lasketaan ensin kaikille Digiroad-pysäkeille (liite 4, rivit 251–260). Tämän jälkeen saaduista sisällöllisen täydellisyyden arvoista lasketaan keskiarvot ELY-alueille (liite 4, rivit 263–277) sekä kunta-alueille (liite 4, rivit 280–301). Lopuksi lasketaan vielä tietokohtainen sisällöllinen täydellisyys koko Suomen kattavasta Digiroad-aineistosta (liite 4, rivit 304–323).

Tutkielmassa ajallista tarkkuutta tarkastellaan tiedon ajankohtaisuutena Digiroadin pysäkkiaineistoon sisältyvän ”Muokattu viimeksi”, eli *muokkauspv*-tiedon avulla. ”Muokattu viimeksi” -tieto sisältää tiedon kohteen edellisestä muokkauspäivämäärästä tai päivämäärästä, jolloin kohde on lisätty Digiroadiin (Digiroad: tietolajien kuvaus 2022). Päivämäärän pohjalta jokaiselle kohteelle lasketaan arvo, joka kuvaa viimeisimmästä muokkauksesta kulunutta aikaa nykyhetkestä vuoden tarkkuudella. Krämer ym. (2007) mukaan voimassaoloajan ilmoittaminen tukee ajankohtaisuuden arviointia yhdessä viimeisimmän päivitysajankohdan kanssa, mutta Digiroadissa pysäkkitiedoille ei ole annettu ennakoivaa voimassaoloaikaa, joten tutkielmassa tarkastellaan vain viimeisintä muokkausajankohtaa. Viimeisimmän muokkausajankohdan mukaan voidaan tehdä oletuksia paikkatiedon laadusta, sillä mitä vähemmän aikaa päivityksestä on kulunut, sitä todennäköisemmin tieto vastaa todellisuutta (Krämer ym. 2007). Ajankohtaisuuden arviointiin ei tarvita referenssiaineistoa. Ajallinen tarkkuus eli ajankohtaisuus lasketaan jokaiselle pysäkille erikseen vertaamalla kuluvan päivämäärän (liite 4, rivi 332) ja pysäkin *muokkauspv*-tiedon välistä erotusta, ja tulos muunnetaan vuosiksi (liite 4, rivi 334–345). Pysäkkikohtaisesta ajankohtaisuustiedosta lasketaan keskiarvot kunta-alueille (liite 4, rivit 348–362) ja ELY-alueille (liite 4, rivit 365–374).

Alueellinen täydellisyys ilmaisee aineiston kattavuutta, eli ylimääräisillä kohteilla tarkoitetaan kohteita, jotka eivät mahdollisesti ole olemassa maastossa, ja puuttuvilla kohteilla tarkoitetaan kohteita, jotka mahdollisesti sijaitsevat maastossa, mutteivat ilmene aineistossa. Alueellista täydellisyyttä kuvataan kohteiden lukumäärän avulla (esim. Girres & Touya 2010, Jackson ym. 2013). Alueellista täydellisyyttä kuvataan laskemalla Digiroad-aineistossa esiintyvät ylimääräiset ja puuttuvat kohteet suhteessa referenssiaineistoon, joka on tutkielman tapauksessa OpenStreetMapin pysäkkiaineisto. Digiroadin pysäkkien lukumäärää verrataan OpenStreetMapin pysäkkien lukumäärään kunta-alueittain kaavalla:

Alueellinen täydellisyys = Digiroad-pysäkit / OSM-pysäkit * 100

Tuloksena saadaan suhdeluku, joka ollessaan <100 prosenttia kertoo alueella olevan vähemmän Digiroad-pysäkkejä kuin OSM-pysäkkejä, ja ollessaan >100 prosenttia kertoo alueella olevan

enemmän Digiroad-pysäkkejä kuin OSM-pysäkkejä. Alueellinen täydellisyys lasketaan kunta-alueille valitsemalla kunta-alueella sijaitsevat pysäkit kummastakin pysäkkiaineistosta ja laskemalla näiden välinen suhde (liite 4, rivit 380–399). Tämän jälkeen ELY-alueille lasketaan keskiarvo kunta-alueiden alueellisesta täydellisyydestä (liite 4, rivit 402–415).

Sijaintitarkkuuden arvioinnissa tarkastellaan pysäkkien suhteellista sijaintia. Suhteellisella sijainnilla tarkoitetaan kohteiden mahdollista sijaintieroa referenssiaineistoissa esiintyviin vastaaviin kohteisiin. Suhteellista sijaintia voidaan siten tarkastella vain niiden pysäkkien osalta, jotka esiintyvät sekä Digiroadissa että OpenStreetMapissa. Pistemäisten vektoriaineistojen suhteellista sijaintitarkkuutta voidaan arvioida pisteiden välisellä euklidisella etäisyydellä (Jackson ym. 2013). Jotta euklidinen etäisyys lasketaan samaa pysäkkiä kuvaavien kohteiden välille, tulee kohteet yhdistää samaan taulukkoon yhteisen ominaisuustiedon mukaan siten, että samalla taulukon rivillä on sekä Digiroad-pysäkin että OpenStreetMap-pysäkin sijaintitieto (liite 4, rivit 421–437). Yhteisenä attribuuttina käytetään Digiroad ID -tunnusta, joka on Digiroad-aineistossa sarakkeessa *valtak_id* ja OpenStreetMap-aineistossa sarakkeessa *ref:findr*. Sarakkeiden nimi tulee vaihtaa samaksi (liite 4, rivi 424). Kaikilla OSM-pysäkeillä ei ole Digiroad ID -tunnusta, joten yhteensovittamista ei voida tehdä kaikille Digiroadin pysäkeille. Yhteensovitettuja pysäkkejä on 72 439.

Pysäkkien suhteellista sijaintia tarkastellessa on tärkeää huomioida rakenteelliset erot aineistojen välillä. Digiroadissa pysäkkien sijaintikoordinaatit sidotaan tien tai kadun keskilinjageometriaan, kun taas OpenStreetMapissa sijaintikoordinaatit ovat maastokoordinaatteja, eli ne kuvaavat pysäkin todellista sijaintia maastossa. Tämän vuoksi suhteellisen sijaintitarkkuuden arvioimisessa käytetään 30 metrin viitearvoa. Samaa pysäkkiä edustavat pisteet Digiroadissa ja referenssiaineistossa tulkitaan olevan samassa sijainnissa, jos niiden välinen etäisyys on alle 30 metriä. Sijaintitarkkuus lasketaan siis kaavalla:

> Sijaintitarkkuus = 100 – (yli 30 m toisistaan olevat pysäkkiparit / kaikki pysäkkiparit * 100)

Jokaiselle onnistuneesti yhteensovitetulle pysäkkiparille lasketaan kohteiden välinen euklidinen etäisyys (liite 4, rivit 440–450). Tämän jälkeen kunta-alueille lasketaan sijaintitarkkuusarvo valitsemalla alueella sijaitsevat pysäkkiparit, suodattamalla alle 30 metrin etäisyydellä toisestaan olevat pysäkkiparit ja laskemalla jäljelle jäävien pysäkkiparien määrä suhteessa alueen kaikkiin pysäkkipareihin (liite 4, rivit 457–473). Tuloksena saadaan

suhdeluku, joka kuvaa samassa sijainnissa olevien pysäkkien määrää kunta-alueella. Lopuksi ELY-alueille lasketaan keskiarvot kunta-alueiden sijaintitarkkuuksista (liite 4, rivit 476–488).

2.7 Geovisualisointi

Ennen laatutietojen karttavisualisointia lasketaan laatutiedoista tilastollisia tunnuslukuja sekä luodaan pylväs- ja laatikkokuvaajat. Tilastollisiksi tunnusluvuiksi laatutiedoista lasketaan pienin ja suurin arvo, keskiarvo, mediaani, vinouma ja huipukkuus (liite 4, rivit 548–559).

Lisäksi selvitetään laatutekijöiden mahdollisia keskinäisiä riippuvuuksia sekä pysäkkien alueellisen lukumäärän vaikutusta laatuun laskemalla laatutekijöiden sekä pysäkkien lukumäärän keskinäistä korrelaatiota. Korrelaation laskemiseen käytetään Kendallin korrelaatiokerrointa (liite 4, rivit 519–545). Kendallin korrelaatiokerrointa käytetään siksi, että laatutiedot eivät tilastollisten tunnuslukujen perusteella noudata normaalijakaumaa, joten siksi ei ole perusteltua käyttää Pearsonin korrelaatiokerrointa. Lisäksi otokset ovat pienehköjä ja aineistossa saattaa esiintyä samoja lukuja useita kertoja, jonka vuoksi Kendall sopii menetelmäksi Spearmanin korrelaatiokerrointa paremmin.

Tilastollisten tunnuslukujen jälkeen luodaan ELY-alueiden välistä vaihtelua kuvaavat vaakasuuntaiset laatikkokuvaajat jokaisesta laatutekijästä (liite 4, rivit 562–582). Lisäksi jokaista laatutekijää kuvaavasta laatutiedosta luodaan pylväskuvaajat, jotka kuvaavat laatutietojen jakautumista kunta-aineistossa (liite 4, rivit 585–732).

Tilastollisten tunnuslukujen ja kuvaajien jälkeen interaktiivinen karttanäkymä toteutetaan hyödyntäen Folium-moduulia (liite 4, rivit 735–854). Folium mahdollistaa interaktiivisen Leaflet-kartan luomisen Pythonilla, Leafletin ollessa JavaScript-ohjelmointikieleen pohjautuva kirjasto. Leaflet on avoimen lähdekoodin monipuolinen ja tehokas mutta helppokäyttöinen ja monille alustoille sopiva paikkatiedon visualisointityökalu. Foliumia käyttämällä voidaan tuotetut laatutiedot visualisoida suoraan samassa ympäristössä ja tarvittaessa suoraan laatutietojen laskemisen yhteydessä kanssa, ilman turhia välivaiheita.

Kartoissa laatutietojen arvojen luokittelutapana käytetään itse määriteltyä luokittelua, sillä tällöin luokkarajat saadaan määriteltyä haluttuihin tasalukuihin (esim. liite 4, rivit 753–754). Vaihtoehtoisesti olisi voitu käyttää myös Foliumin laskemaa tasavälistä luokittelua, mutta tällöin luokkarajat eivät olisi olleet tasalukuja, mikä vähentäisi kartan luettavuutta ja vaikuttaisi suoraan käyttäjäkokemukseen.

Lisäksi kartalle lisätään ELY-alueiden rajat (liite 4, rivit 823–835). Lopuksi kartalle lisättiin myös kaksi vaihtoehtoista taustakarttaa, karttatasojen valintaikkuna sekä hakukenttä, jolla voi hakea esimerkiksi kuntaa (liite 4, rivit 837–847).

3 Tulokset

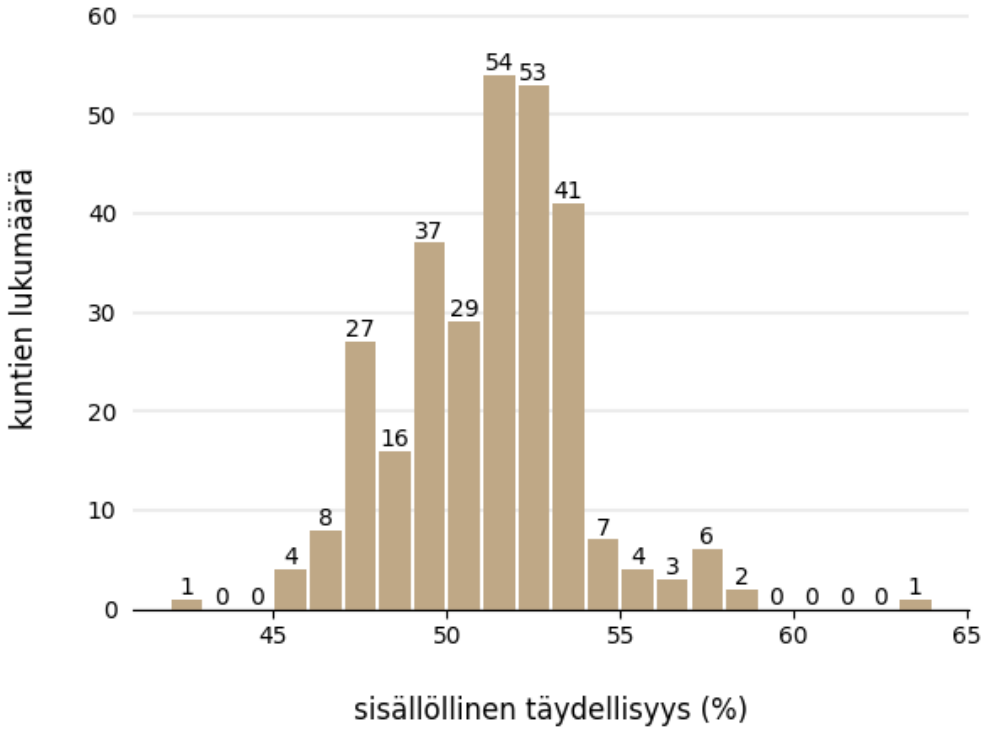
3.1 Joukkoliikenteen pysäkkitiedon laatu

Joukkoliikenteen pysäkkitiedon laatua mitattiin Digiroad-aineistosta neljän eri laatutekijän mukaisesti (taulukko 4).

Taulukko 4. Tilastollisia tunnuslukuja kunta-alueiden pysäkkitiedon laadusta.

	Sisällöllinen täydellisyys	**Alueellinen täydellisyys**	**Ajankohtaisuus**	**Sijaintitarkkuus**
Pienin arvo	42,39 %	80,00 %	0,97 vuotta	75,00 %
Suurin arvo	63,44 %	127,38 %	7,82 vuotta	100,00 %
Keskiarvo	51,21 %	100,07 %	4,13 vuotta	96,96 %
Mediaani	51,47 %	100,00 %	4,47 vuotta	98,26 %
Huipukkuus	1,67	17,29	-1,65	10,15
Vinouma	0,25	2,10	-0,04	-2,56

Suurimmassa osassa kunta-alueista sisällöllinen täydellisyys oli melko kattavaa sisällöllisen täydellisyyden ollessa 64–68 prosenttia (kuva 3). Sisällöllisen täydellisyyden keskiarvo oli 51,21 prosenttia ja mediaani 51,47 prosenttia (taulukko 4). Heikoin sisällöllinen täydellisyys oli Kemissä (42,39 prosenttia) ja korkein Järvenpäässä (63,44 prosenttia). Sisällöllisen täydellisyyden vinous on 0,25, eli sisällöllinen täydellisyys painottuu lievästi korkeisiin arvoihin. Lisäksi huipukkuus on 1,67, eli sisällöllinen täydellisyys on hyvin huiputtunut suhteessa normaalijakaumaan.



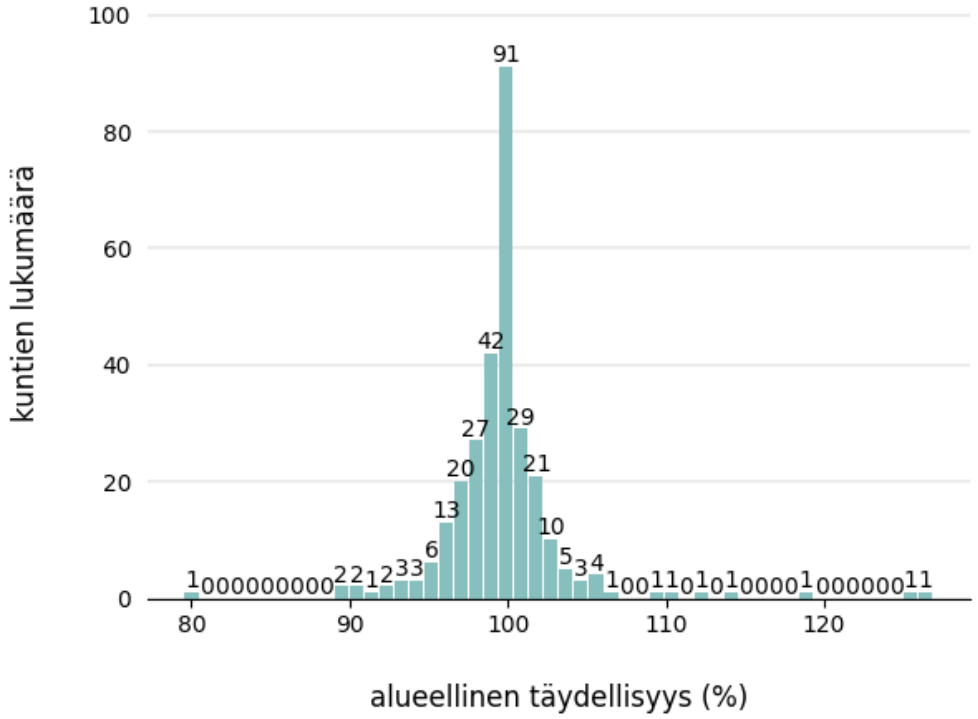
Kuva 3. Kunta-alueiden sisällöllisen täydellisyyden jakautuminen (n = 293).

Attribuuttikohtainen sisällöllinen täydellisyys kaikkien pysäkkien osalta on esitetty liitteessä 3. Pakollisista tiedoista tietojen ylläpitäjä ja pysäkin tyyppi -tiedoissa ei ollut puutteita, mutta pysäkin nimi suomeksi puuttui vähäiseltä määrältä pysäkkejä sisällöllisen täydellisyyden ollessa 99,63 prosenttia. Palvelutasoluokka-tiedon pakollisuus ei ole tiedossa, mutta kaikilla pysäkeillä oli kyseinen tieto. Vapaaehtoisten tietojen sisällöllisessä täydellisyydessä oli hieman vaihtelua. Yli puolelle kaikista Digiroadin pysäkeistä oli annettu vapaaehtoinen tieto pysäkin ensimmäisestä voimassaolopäivästä (97,07 prosenttia), pysäkin katoksesta (87,71 prosenttia), pysäkin korotuksesta (62,51 prosenttia) sekä pysäkin esteettömyydestä (85,72 prosenttia). Hieman vähäisemmin oli annettu tieto pysäkin nimestä ruotsiksi (27,25 prosenttia), ylläpitäjän tunnuksesta (28,41 prosenttia), matkustajatunnuksesta (24,13 prosenttia), ja liikennöintisuunnasta (16,11 prosenttia). Muilta osin vapaaehtoisten tietojen sisällöllinen täydellisyys oli alle 10 prosenttia.

Suurimmalla osalla kunta-alueista alueellinen täydellisyys oli hyvä (95–105 prosenttia) (kuva 4). Alueellisen täydellisyyden keskiarvo on 100,07 prosenttia ja mediaani 100,0 prosenttia, eli suurimmalla osalla kunta-alueista Digiroad-pysäkkejä oli saman verran kuin OSM-pysäkkejä (taulukko 4). Alueellisella täydellisyydellä vinouma on 2,10 ja huipukkuus 17,29, eli aineisto

on erittäin huiputtunut verrattuna normaalijakaumaan, sekä positiivisesti vino. Alueellinen täydellisyys siis painottuu siten, että Digiroad-pysäkkejä on enemmän kuin OSM-pysäkkejä (alueellinen täydellisyys > 100,00 prosenttia).

Alhaisin alueellinen täydellisyys eli 80,0 prosenttia oli Utsjoella, eli Digiroad-aineistossa pysäkkejä oli 20,0 prosenttia vähemmän kuin referenssiaineistossa. Korkein alueellinen täydellisyys oli Raumalla, jossa alueellinen täydellisyys oli 127,38 prosenttia, eli Digiroad-aineistossa oli 27,38 prosenttia enemmän joukkoliikenteen pysäkkejä kuin referenssiaineistossa. Alueellinen täydellisyys oli tasan 100,00 prosenttia yhteensä 51 kunnalla eli 17 prosentilla kunnista. Näissä kunnissa pysäkkien määrä oli aineistojen välillä yhtäläinen.



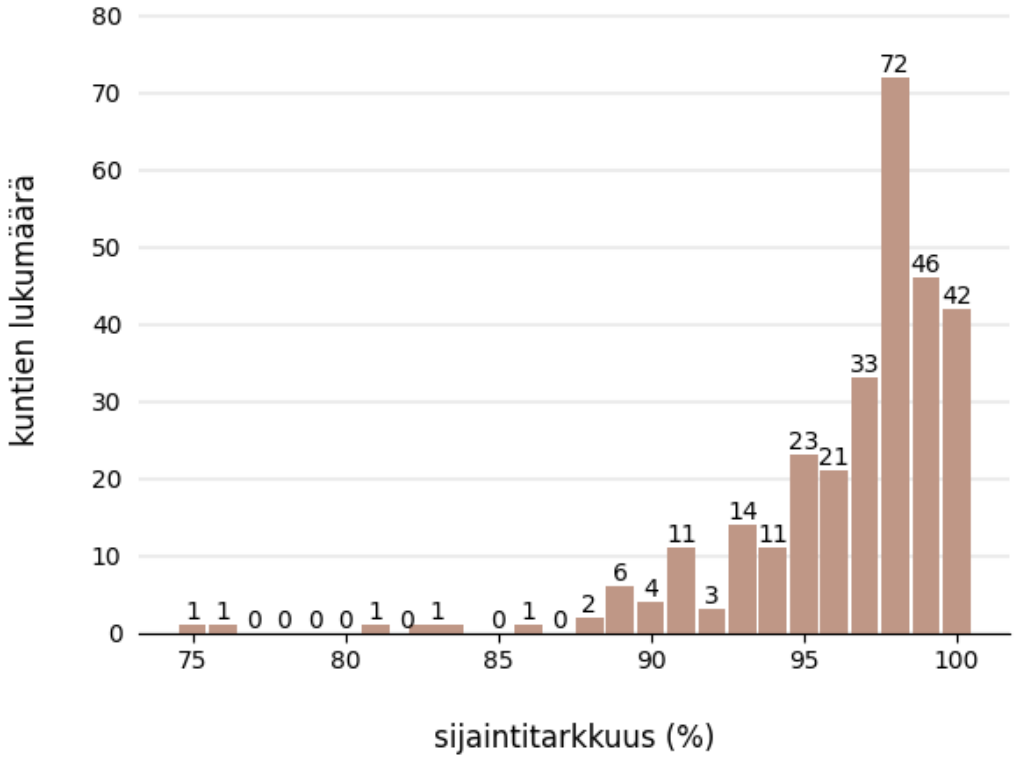
Kuva 4. Kunta-alueiden alueellisen täydellisyyden jakautuminen (n = 293).

Suurimmalla osalla kunta-alueista pysäkkitiedon ajankohtaisuus oli 1–2 vuotta (kuva 5). Lisäksi lähes 40 prosentilla kunta-alueista ajankohtaisuus oli 5–7 vuotta. Vähemmän kuin 1 vuosi oli vain 1,02 prosentilla kunta-alueista eli kolmella kunnalla. Ajankohtaisuuden keskiarvo on 4,08 vuotta ja mediaani 4,43 vuotta (taulukko 4). Korkein ajankohtaisuus on Hämeenkyrössä, jossa keskiarvoinen aika tietojen muokkaamisesta oli 0,97 vuotta. Heikoin ajankohtaisuus oli Lestijärvellä, jossa keskiarvoinen aika pysäkkitiedon muokkaamisesta oli 7,82 vuotta. Ajankohtaisuuden vinouma on -0,04 ja huipukkuus -1,65, eli jakauma on huiputon,

mutta myös lähes vinoumaton. Kunta-alueittain tarkasteltuna ajankohtaisuus siis painottuu kumpaankin päähän arvoja.

Kuva 5. Kunta-alueiden ajankohtaisuuden jakautuminen (n = 293). Laskentapäivä 15.3.2024.

Suurimmalla osalla kunta-alueista sijaintitarkkuus oli korkea eli yli 97,5 prosenttia (kuva 6). Yli 95 prosentin sijaintitarkkuuteen ylsi 80 prosenttia tarkastelluista kunta-alueista. Sijaintitarkkuuden keskiarvo kunta-alueilla oli 96,96 prosenttia ja mediaani 98,26 prosenttia (taulukko 4). Heikoin sijaintitarkkuus on Pirkkalassa (75 prosenttia). Täydelliseen sijaintitarkkuuteen ylsi 42 kunta-aluetta, eli 14,3 prosenttia tarkastelluista kunta-alueista. Sijaintitarkkuuden vinouma on -2,56 ja huipukkuus 10,15, eli aineisto on voimakkaasti huiputtunut sekä negatiivisesti vino, eli jakauma painottuu suuriin arvoihin.



Kuva 6. Kunta-alueiden sijaintitarkkuuden jakautuminen (n = 293).

Laatutiedot korreloivat keskenään vähäisesti (taulukko 5). Yli 0,1 korrelaatiokerroin voidaan havaita sisällöllisen täydellisyyden ja sijaintitarkkuuden välillä, muutoin laatutietojen välinen korrelaatiokerroin jää 0,0 ja 0,1 väliin. Korrelaatio on positiivista, paitsi ajankohtaisuus korreloi negatiivisesti alueellisen täydellisyyden ja sijaintitarkkuuden kanssa, mutta tämä korrelaatio ei ole tilastollisesti merkitsevää (p-arvo > 0,05). Lisäksi sisällöllinen täydellisyys ei korreloi merkitsevästi muiden laatutietojen kuin sijaintitarkkuuden kanssa. Muu laatutietojen välinen korrelaatio on tilastollisesti merkitsevää. Suurin tilastollinen merkitsevyys on alueellisen täydellisyyden ja sijaintitarkkuuden välillä (p-arvo 0,02). Laatutietojen välillä on siis havaittavissa keskinäisiä riippuvaisuuksia.

Pysäkkien alueellisen lukumäärän suhteen tilastollista merkitsevää korrelaatiota on sisällöllisen täydellisyyden (p-arvo 0,03) ja ajankohtaisuuden (p-arvo 0,01) kanssa (taulukko 5). Kumpikin korreloi lievän negatiivisesti pysäkkien lukumäärän kanssa, sisällöllinen täydellisyys hieman voimakkaammin kuin ajankohtaisuus. Kunta-alueilla, joissa on enemmän pysäkkejä, voi siis olla heikompi sisällöllinen täydellisyys, mutta korkeampi ajankohtaisuus.

Taulukko 5. Laatutietojen keskinäinen korrelaatio ja korrelaatio pysäkkien kunta-alueellisen lukumäärän mukaan.

	Sisällöllinen täydellisuus	**Alueellinen täydellisyys**	**Ajankohtai-suus**	**Sijaintitarkkuus**	**Pysäkkien lukumäärä**
Sisällöllinen täydellisyys	-	0.060 (p-arvo 0.12)	0.055 (p-arvo 0.15)	0.116 (p-arvo 0.03)	-0.083 (p-arvo 0.03)
Alueellinen täydellisyys	0.060 (p-arvo 0.12)	-	-0.017 (p-arvo 0.66)	0.092 (p-arvo 0.02)	-0.030 (p-arvo 0.44)
Ajankohtai-suus	0.055 (p-arvo 0.15)	-0.017 (p-arvo 0.66)	-	-0.017 (p-arvo 0.66)	-0.099 (p-arvo 0.01)
Sijaintitark-kuus	0.116 (p-arvo 0.03)	0.092 (p-arvo 0.02)	-0.017 (p-arvo 0.66)	-	-0.184 (p-arvo 3.28)

3.2 Pysäkkitiedon laadun puoliautomaattinen arviointimenetelmä

Tutkielman tuloksena syntyi 715 riviä Python-koodia, jonka suorittamalla Digiroadin joukkoliikennepysäkkiaineistosta saadaan ELY-alue- ja kuntatasoista laatutietoa (liite 4). Koko koodin suorittaminen Microsoft Visual Studio Codella kesti 7 minuuttia ja 48,2 sekuntia Asus Zenbook vuoden 2017 mallilla, jossa on Intel i5-8250U-suoritin. Eri vaiheiden suoritusajat on kuvattu taulukossa 6. Samoja lähtöaineistoja käyttäen koodi tuottaa aina saman lopputuloksen, lukuun ottamatta ajankohtaisuutta, jota mitataan suhteessa siihen ajankohtaan, jolloin koodi suoritetaan. Tutkielmassa luotu koodi on vapaasti käytettävissä MIT-lisenssillä ja se on saatavilla GitHub-palvelusta (https://github.com/jennikarro/DRStops-quality-assessment-tool).

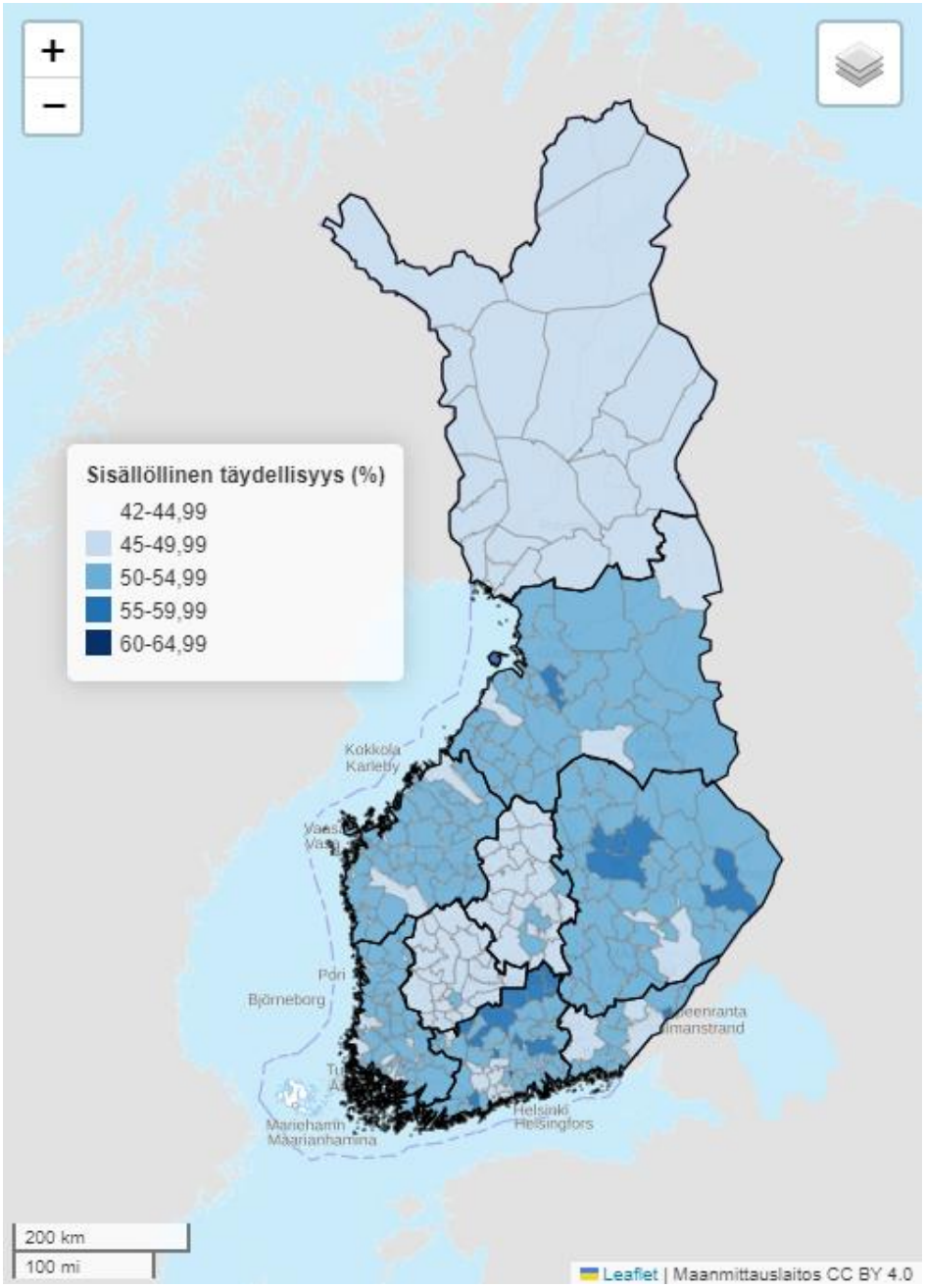
Tutkielmassa tuotettu interaktiivinen kartta tallennetaan HTML-tiedostoon, joka voidaan avata paikallisesti selaimessa (esim. Google Chrome) kartan katselua varten. Lisäksi algoritmi tallentaa laatutietoja sisältävät kunta- ja ELY-alueaineistot Excel-taulukkoon geometrian kanssa, jolloin aineistot voidaan tarvittaessa avata myös paikkatieto-ohjelmistossa, kuten QGISissä.

Taulukko 6. Laatutietojen tuottamiseen ja visualisointiin kuluvat suoritusajat.

Vaihe	Suoritusaika
Tarvittavien Python-pakettien tuonti	0 min 29,1 s
Lähtöaineistojen tuonti ja tarvittavat lähtöaineistojen muokkaukset (Digiroad)	3 min 30,3 s
Lähtöaineistojen tuonti ja tarvittavat lähtöaineistojen muokkaukset (Täydentävät aineistot)	0 min 16,6 s
Lähtöainestojen tuonti ja tarvittavat lähtöaineistojen muokkaukset (OpenStreetMap)	4 min 47,9 s
Sisällöllinen täydellisyys	0 min 41,8 s
Ajankohtaisuus	0 min 9,5 s
Alueellinen täydellisyys	0 min 12,4 s
Sijaintitarkkuus	0 min 22,8 s
Vienti .xlsx-tiedostoihin	0 min 1,2 s
Visualisointi (tilastolliset tunnusluvut ja kaaviot)	0 min 5,8 s
Visualisointi (Folium-kartta)	0 min 10,8 s

3.3 Joukkoliikenteen pysäkkitiedon alueellinen vaihtelu

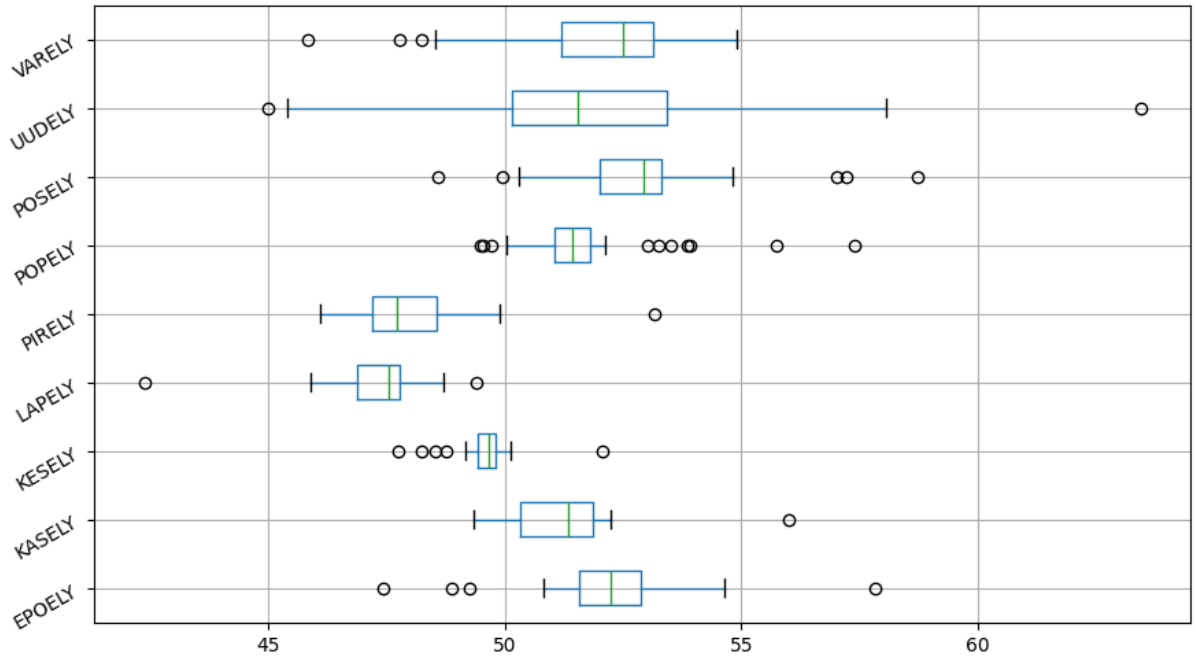
Pysäkkitiedon sisällöllisessä täydellisyydessä voidaan havaita selkeää maantieteellistä vaihtelua (kuva 7). Alhaisin sisällöllinen täydellisyys esiintyy Lapin ja Pirkanmaan ELY-keskusten alueella. Pohjois-Pohjanmaan, Keski-Suomen ja Kaakkois-Suomen ELY-keskusten alueella sisällöllinen täydellisyys on lähes kaikkien kuntien alueella 62–66 prosenttia, Pohjois-Savon ELY-keskuksen alueella suurimmalla osalla kunnista sisällöllinen täydellisyys on korkeampi, 66–70 prosenttia. Etelä-Pohjanmaan sekä Varsinais-Suomen ELY-keskusten alueella vaihtelua on enemmän, kuten myös Uudenmaan ELY-alueella. Uudenmaan ELY-alueella sijaitsevat lähes ainoat kattavaan sisällölliseen täydellisyyteen yltävät kunta-alueet (sisällöllinen täydellisyys yli 70 prosenttia). Uudenmaan ELY-alueen lisäksi yli 70 prosentin hyvään sisällölliseen täydellisyyteen yltää vain Pohjois-Pohjanmaan ELY-alueella sijaitseva Hailuoto.



Kuva 7. Sisällöllisen täydellisyyden maantieteellinen vaihtelu kunnittain Suomessa. ELY-alueiden rajat mustalla viivalla. Karttaa voi myös katsella selaimessa: https://jennikarro.github.io/DRStops-QI-map.html

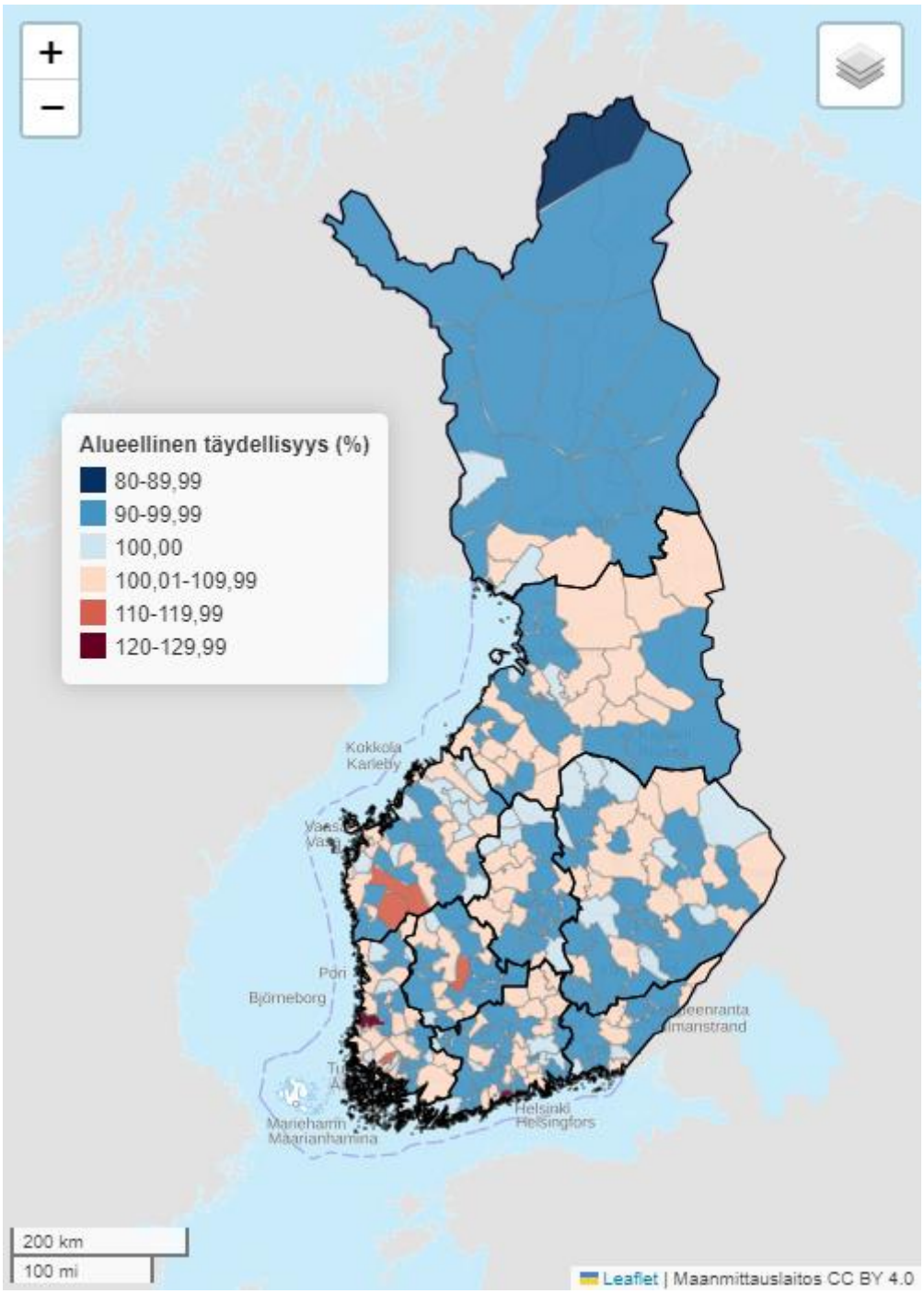
Sisällöllinen täydellisyys oli suurimmalla osalla ELY-alueista mediaaniltaan yli 50 prosenttia mutta alle 55 prosenttia (kuva 8). Tätä alempi mediaani oli vain Pirkanmaan, Lapin sekä Keski-Suomen ELY-alueilla. Suurin vaihtelevuus on Uudenmaan ELY-alueella, jossa huonoimman ja parhaimman sisällöllisen täydellisyyden välinen ero oli yli 10 prosenttia, kun lähes kaikilla muilla ELY-alueilla Varsinais-Suomea lukuunottamatta sama erotus ei ylittänyt 5 prosenttia. Kaikilla muilla ELY-alueilla oli minimistä poikkeavia arvoja paitsi Pirkanmaan ja Kaakkois-Suomen ELY-alueilla, ja lisäksi korkeimmasta arvosta poikkeuksia oli kaikilla muilla paitsi

Varsinais-Suomen ELY-alueella, eli kunta-alueiden sisällöllisessä täydellisyydessä oli hajanaisia ääripoikkeuksia.



Kuva 8. Sisällöllinen täydellisyys ELY-alueittain ryhmiteltynä.

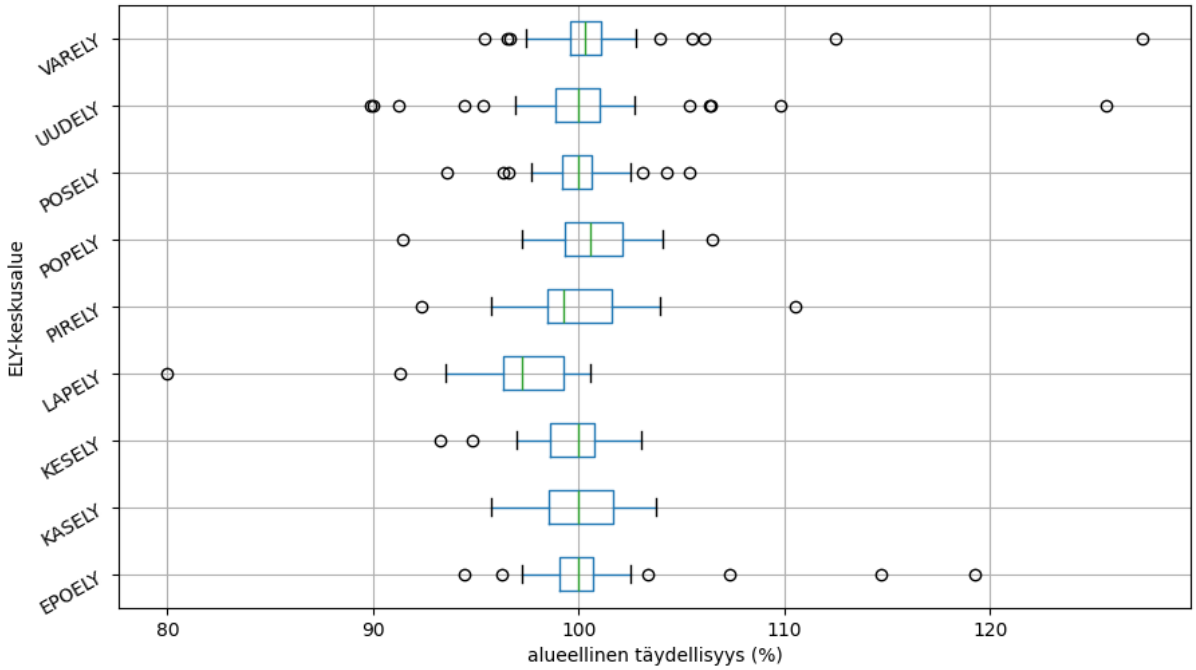
Alueellinen täydellisyys ei juurikaan vaihdellut maantieteellisesti (kuva 9). Lapin ja Kaakkois-Suomen ELY-keskusten alueilla esiintyi vähiten täydellisyyttä hipovaa alueellista täydellisyyttä (99–101 prosenttia). Lisäksi Lapin ELY-alueella suurimmalla osalla kunta-alueista oli vähemmän Digiroad-pysäkkejä kuin OSM-pysäkkejä. Alueellisen täydellisyyden ääripäät levittyivät eri puolelle Suomea, sillä enemmän Digiroad-pysäkkejä oli OSM-pysäkkeihin nähden enemmän Länsi-Suomessa, kun taas Digiroad-pysäkkien vähäinen määrä suhteessa OSM-pysäkkeihin korostui Lapissa Utsjoella.



Kuva 9. Alueellisen täydellisyyden maantieteellinen vaihtelu Suomessa. ELY-alueiden rajat mustalla viivalla. Karttaa voi myös katsella selaimessa: https://jennikarro.github.io/DRStops-QI-map.html

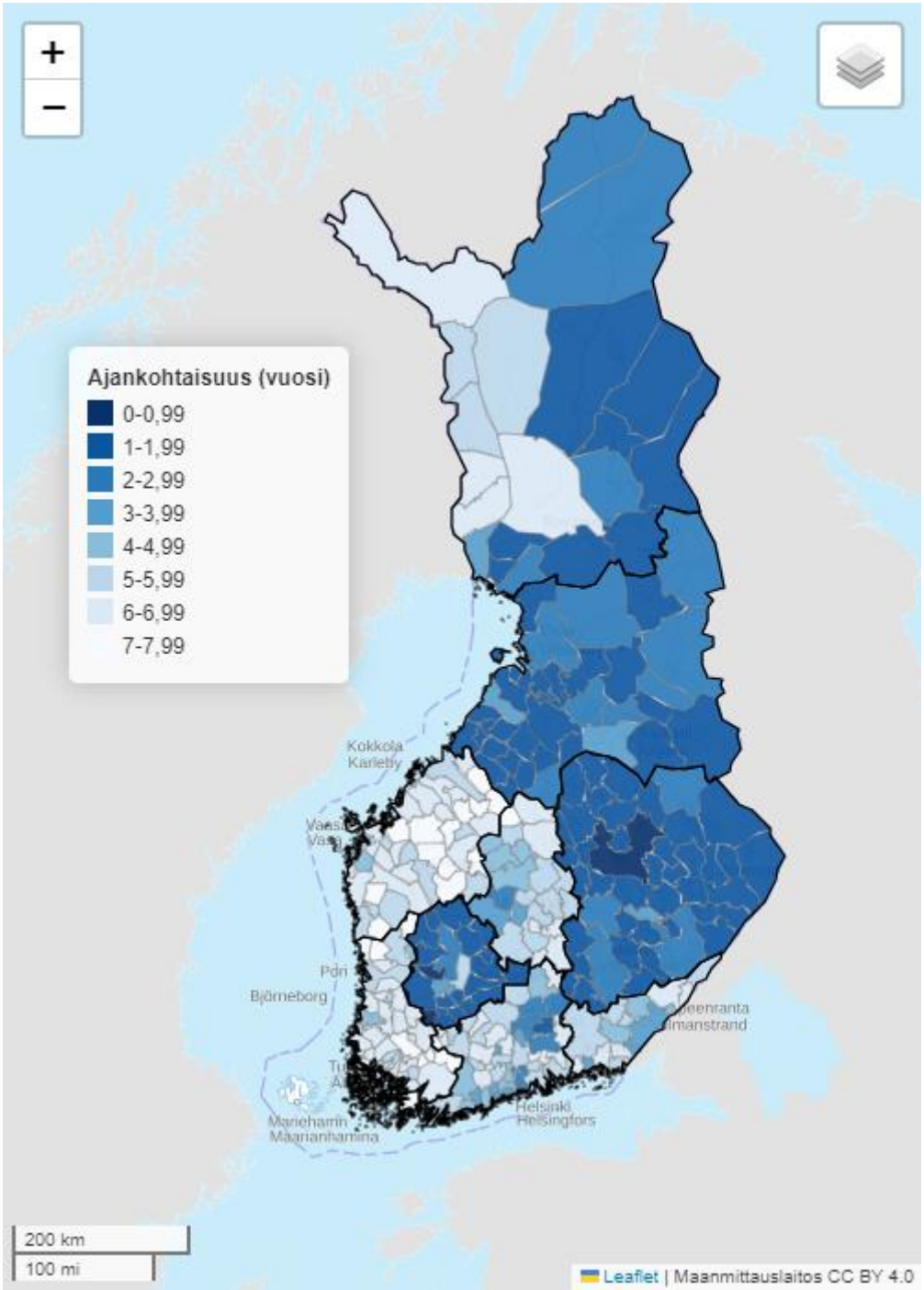
ELY-alueista Uudellamaalla, Pohjois-Savossa, Keski-Suomessa, Kaakkois-Suomessa ja Etelä-Pohjanmaalla alueellisen täydellisyyden mediaani oli 100 prosenttia (kuva 10). Varsinais-Suomen ja Pohjois-Pohjanmaan ELY-alueilla mediaani oli yli 100 prosenttia, eli näiden alueiden kunta-alueilla Digiroad-pysäkkejä esiintyi enemmän kuin OSM-pysäkkejä, kun taas Pirkanmaalla ja Lapissa mediaani jäi alle 100 prosentin, eli näillä alueilla oli enemmän kunta-alueita, joissa Digiroad-pysäkkejä on vähemmän kuin OSM-pysäkkejä. Kaakkois-Suomen

ELY-alueella ei ollut pienimmästä tai suurimmasta arvosta poikkeavia arvoja. Matalin vaihtelu alueellisessa täydellisyydessä oli Pohjois-Savon ELY-alueella, ja suurin vaihtelu Pirkanmaan ELY-alueella.



Kuva 10. Alueellinen täydellisyys ELY-alueittain ryhmiteltynä.

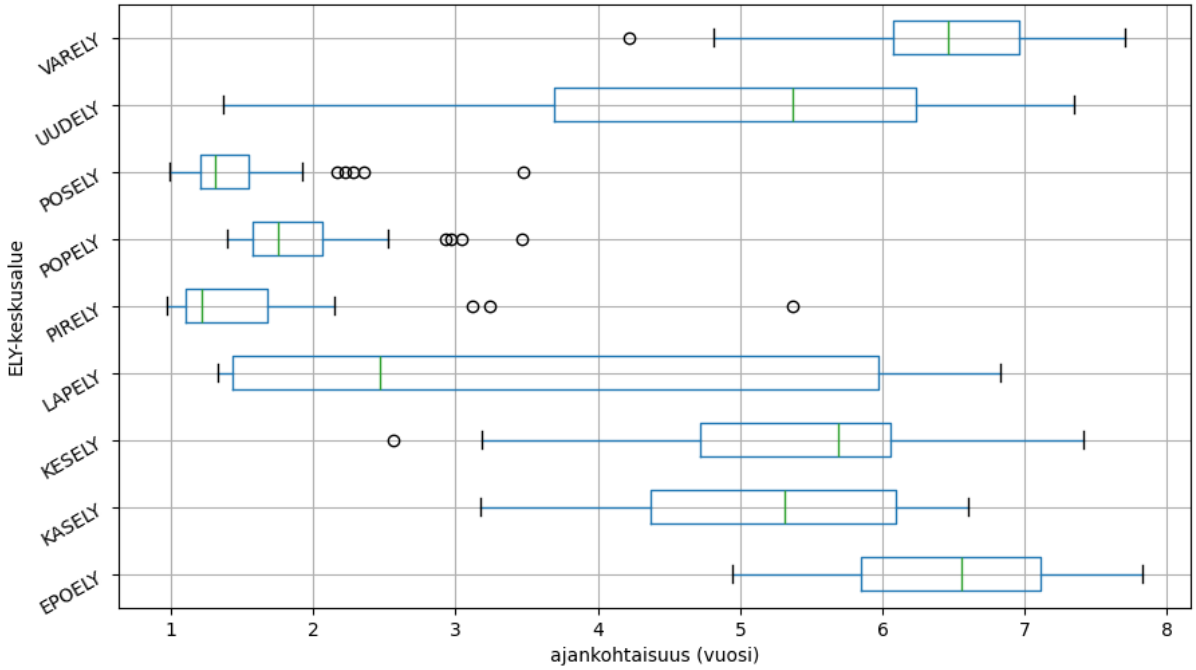
Pysäkkitiedon ajankohtaisuus oli selkeästi parempi Itä-Suomessa sekä Pirkanmaalla (kuva 11). Pohjois-Savon sekä Pohjois-Pohjanmaan ELY-alueilla pysäkkitietojen keskiarvoinen aika viimeisimmästä päivityksestä oli korkeintaan 4 vuotta. Myös Pirkanmaa erottui hyvällä ajankohtaisuudella. Kaakkois-Suomessa, Keski-Suomessa sekä Uudenmaan ELY-alueella vaihtelua oli enemmän, minkä lisäksi Lapin ELY-keskuksen alueella oli selkeä ero Itä- ja Länsi-Lapin välillä. Heikompi pysäkkitiedon ajankohtaisuus korostui erityisesti Etelä-Pohjanmaan sekä Varsinais-Suomen ELY-alueilla.



Kuva 11. Ajankohtaisuuden maantieteellinen vaihtelu Suomessa. ELY-alueiden rajat mustalla viivalla. Karttaa voi myös katsella selaimessa: https://jennikarro.github.io/DRStops-QI-map.html

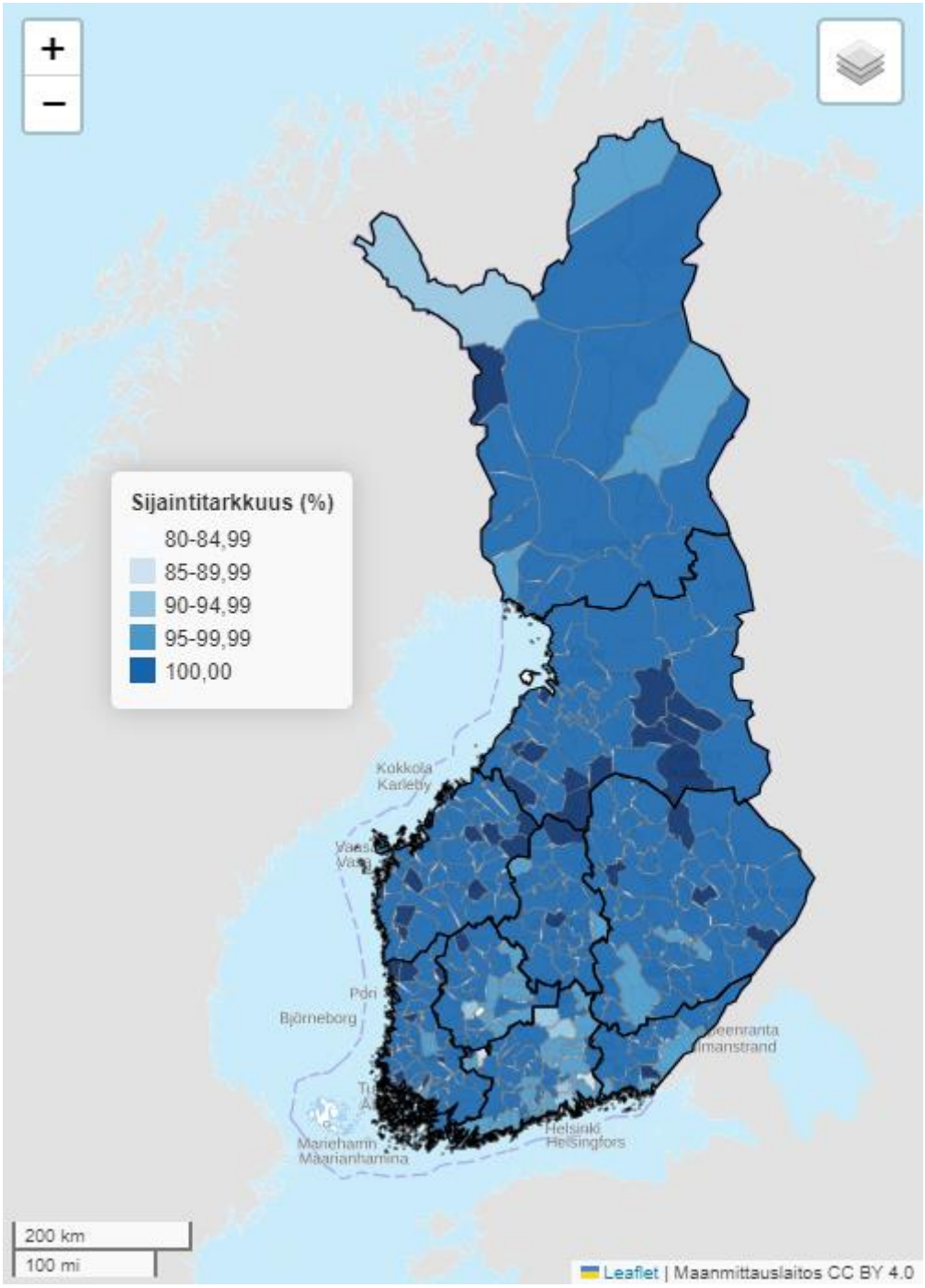
Sen lisäksi, että Pohjois-Savon, Pohjois-Pohjanmaan ja Pirkanmaan ELY-alueilla ajankohtaisuus oli hyvä, myös sen sisäinen vaihtelu oli vähäistä (kuva 12). Näillä ELY-alueilla oli kuitenkin jonkin verran maksimista poikkeavia arvoja. Suurin vaihtelu oli Lapin ELY-alueella, jossa alaneljännes oli 1–2 vuoden välissä ja yläneljännes kunta-alueista melkein 6 vuoden kohdalla. Ajankohtaisuudeltaan alimmat 25 prosenttia kunta-alueista olivat siis

ajankohtaisuudeltaan hyviä, alle 2 vuotta, ja ajankohtaisuudeltaan korkeimmat 25 prosenttia kunta-alueista olivat sellaisia, joiden pysäkkitieto on keskiarvoisesti vanhempaa kuin 6 vuotta. Uudenmaan ELY-alueella minimi- ja maksimiarvojen välinen erotus oli suurempi kuin Lapin ELY-alueella, mutta alaneljänneksen ja yläneljänneksen välinen ero ei ollut niin suuri kuin Lapissa, eli suurin osa alueella sijaitsevista kunta-alueista oli ajankohtaisuudeltaan lähellä toisiaan. Uudellamaalla ajankohtaisuus painottui huonompaan Lappiin verrattuna, sillä Lapin ELY-alueella ajankohtaisuus kuitenkin painottui matalempiin arvoihin mediaanin ollessa melkein 2,5 vuotta. Mediaani oli Pohjois-Savon, Pohjois-Pohjanmaan sekä Pirkanmaan ELY-alueilla 1–2 vuoden välissä. Uudenmaan, Keski-Suomen ja Kaakkois-Suomen ELY-alueilla mediaani oli 5–6 vuoden välissä, ja Varsinais-Suomen sekä Etelä-Pohjanmaan ELY-alueiden mediaani oli 6 ja 7 vuoden välissä.



Kuva 12. Ajankohtaisuus ELY-alueittain ryhmiteltynä.

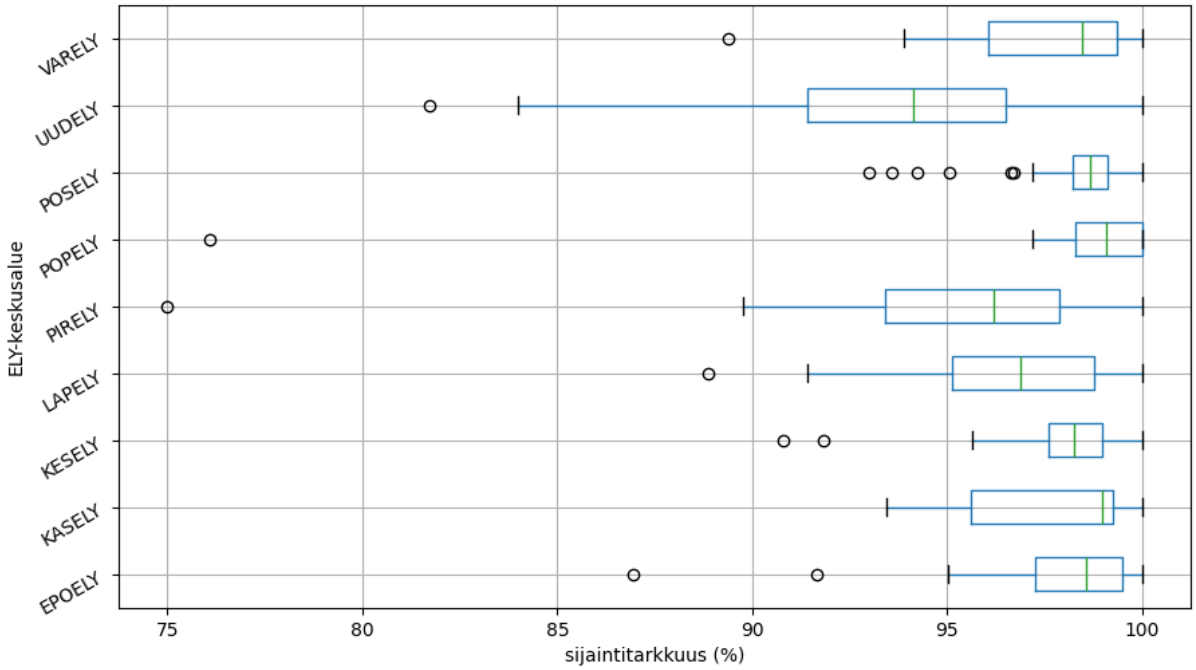
Sijaintitarkkuudessa maantieteellinen vaihtelevuus oli vähäistä, mutta hyvän sijaintitarkkuuden kunta-alueet painottuivat selkeästi itäiseen Suomeen sekä Pohjois-Pohjanmaan, Etelä-Pohjanmaan ja Keski-Suomen ELY-alueiden välisille rajaseuduille (kuva 13). Lisäksi Uudenmaan ELY-alue esiintyi selkeästi sijaintitarkkuudeltaan heikompana muihin ELY-alueisiin verrattuna.



Kuva 13. Sijaintitarkkuuden maantieteellinen vaihtelu Suomessa. ELY-alueiden rajat mustalla viivalla. Karttaa voi myös katsella selaimessa: https://jennikarro.github.io/DRStops-QI-map.html

Kaikilla ELY-alueilla sijaintitarkkuuden maksimi oli 100 prosenttia, eli kaikilla ELY-alueilla sijaitsee kunta-alueita, joiden sijaintitarkkuus oli täydellinen (kuva 14). Alle 97,5 prosentin mediaanin jäivät vain Uudenmaan, Pirkanmaan ja Lapin ELY-alueet. Matalin vaihtelu sijaintitarkkuudessa kuntien välillä oli Pohjois-Savon ELY-alueella, vaikkakin alueella oli jonkin verran poikkeusarvoja minimistä. Pohjois-Pohjanmaan ELY-alueella yläneljännes ja maksimi ovat sama arvo, eli 25 prosentilla sen alueella sijaitsevista kunnista oli täydellinen

sijaintitarkkuus. Suurin vaihtelu oli Uudenmaan ELY-alueella, jossa minimin ja täydellisen sijaintitarkkuuden erotus oli yli 15 prosenttia. Sijaintitarkkuudeltaan heikoimmat kunta-alueet olivat Pohjois-Pohjanmaan ja Pirkanmaan ELY-alueilla.



Kuva 14. Sijaintitarkkuus ELY-alueittain ryhmiteltynä.

4 Tulosten tarkastelu

4.1 Pysäkkitiedon laadun vaihtelu Suomessa

Koko Suomessa pysäkkitiedon laatu on pääsääntöisesti hyvällä tasolla. Esimerkiksi alueellisessa täydellisyydessä suurin osa kunta-alueista sijoittuu täydelliseen 100 prosenttiin tai lähelle täydellisyyttä, mikä tarkoittaa, että Digiroadin pysäkkien lukumäärä vastaa pitkälti referenssiaineiston pysäkkien lukumäärää. Kun pysäkkikohteiden määrät aineistoissa vastaavat toisiaan, vahvistaa se tiedon oikeellisuutta. Myös sijaintitarkkuus oli suurimmalla osalla pysäkeistä täydellinen tai lähes täydellinen. Koska Digiroadin pysäkkien koordinaatit on lähtökohtaisesti sidottu tien keskilinjageometriaan, voi tutkimuksessa mitattu sijaintitarkkuus heikentyä silloin, kun pysäkki sijaitsee kauempana tiestä, johon se on linkitetty. Virhearvioinnit voitaisiin minimoida parantamalla sisällöllistä täydellisyyttä, eli jos suurimmalla osalla Digiroad-pysäkeistä olisi saatavilla niiden todellinen sijainti maastokoordinaatteina. Tällä hetkellä tieto on vain 9,7 prosentilla pysäkeistä (liite 3).

Pysäkkitietojen sisällöllisessä täydellisyydessä sekä ajankohtaisuudessa oli enemmän selkeää maantieteellistä vaihtelua kuin sijaintitarkkuudessa tai alueellisessa täydellisyydessä. Sisällöllisessä täydellisyydessä tulisi luonnollisesti tavoitella 100 prosentin täydellisyyttä, mutta kunta-alueiden keskiarvoinen sisällöllinen täydellisyys jäi noin 51 prosenttiin. Keskiarvolla ei ole yhteyttä pakollisten tai automaattisesti generoituvien tietojen määrään Digiroadissa, joiden osuus kaikista tiedoista on noin 34 prosenttia (liite 1). Mitä enemmän myös vapaaehtoisesti kirjattavia tietoja joukkoliikenteen pysäkeillä olisi, sitä enemmän lisäarvoa tiedosta voitaisiin tuottaa. Tietokohtaisesti kaikkia Digiroad-pysäkkejä tarkastellessa pakolliset tiedot oli täytetty muutoin, mutta suomenkielinen nimi puuttui 345 pysäkiltä. Näiden pysäkkien lähempi tarkastelu ei antanut ilmi syytä nimitietojen puuttumiselle. Varustetiedot (aikataulu, katos ja mainoskatos, penkki, pyöräteline, sähköinen aikataulunäyttö) olivat monilta osin vähäiset, mutta ne voivat olla arvokkaita pysäkkien käyttäjille, joten tietojen määrää olisi syytä pyrkiä lisäämään. Esteettömyystiedot oli annettu suurimmalle osalle pysäkeistä, mutta saavutettavuuden vuoksi tieto olisi tärkeää löytyä kaikilta pysäkeiltä. Esteettömyystietojen keräämiseen kuitenkin liittyy joitakin ongelmia, kuten yhteisen määrittelyn puuttuminen matkatiedon kontekstissa (Alinikula ym. 2021).

Ajankohtaisuus vaihteli kunta-alueiden välillä suuresti, osalla pysytellen lähellä 1–2 vuotta ja osalla ajankohtaisuuden noustessa yli viiteen vuoteen. Ajankohtaisuuteen vaikuttaa se, ettei

Digiroadin pysäkkitiedossa oli muuta lähdettä tiedon viimeisimmälle tarkastuspäivälle, kuin Viimeisin muokkausajankohta -tieto. Kuitenkin jos pysäkin tietoja ei ole ollut tarpeen muuttaa, ei päivitystä ole tehty eikä tällöin tietoon myöskään jää aikaleimaa siitä, että tieto on tarkastettu oikeaksi. Koska viimeisin muokkausajankohta muuttuu myös kun uusi pysäkki perustetaan Digiroadiin (Digiroad: tietolajien kuvaus 2022), voi ajankohtaisuus olla parempi sellaisilla kunta-alueilla, joissa on perustettu useita uusia pysäkkejä viime vuosina.

Laatutekijät korreloivat suurimmaksi osin keskenään lievän positiivisesti, jolloin voidaan päätellä, että heikot arvot laatutekijöissä kasautuvat samoille pysäkeille ja niiden kunta-alueille. Tämä pätee myös ajankohtaisuuden korrelointiin alueellisen täydellisyyden ja sijaintitarkkuuden kanssa, mutta korrelointi on negatiivista, eli ajankohtaisuuden ollessa hyvä myös sijaintitarkkuus tai alueellinen täydellisyys ovat parempia. Tämä korrelointi ei kuitenkaan ollut tilastollisesti merkitsevää.

Pysäkkien lukumäärä voi mahdollisesti vaikuttaa pysäkkitiedon laatuun, eli pysäkkien lukumäärän voidaan katsoa jollain tapaa vaikuttavan pysäkkitietojen ylläpitoon. Alueellisen täydellisyys ja sijaintitarkkuus eivät vaikuta olevan yhteydessä pysäkkien lukumäärään. Sen sijaan sisällöllinen täydellisyys saattaa olla heikompi niillä kunta-alueilla, joissa on enemmän pysäkkejä. Mielenkiintoista on, että ajankohtaisuus vaikuttaa olevan sitä parempi, mitä enemmän pysäkkejä kunta-alueella on. Tämä voi johtua esimerkiksi siitä, että suurempi pysäkkimäärä lisää tarvetta tietojen aktiivisemmalle ylläpidolle. Lisäksi pysäkkejä todennäköisesti esiintyy enemmän kunta-alueilla, joissa on kattavat julkisen liikenteen palvelut ja kehittyviä kaupunkikeskuksia, jolloin uusia pysäkkejä saatetaan perustaa Digiroadiin useammin, kuin esimerkiksi maaseutukeskuksissa tai periferisissä kunnissa. Uusien pysäkkien perustaminen parantaa ajankohtaisuutta tutkimuksen perustuessa ”Viimeisin muokkausajankohta”-tietoon, joka päivittyy myös uuden pysäkin Digiroadiin luonnin yhteydessä.

Kaikissa laatutekijöissä ei ollut yhtä paljon maantieteellistä vaihtelua kuin toisissa. Sijaintitarkkuus ja alueellinen täydellisyys eivät vaihdelleet alueellisesti paljon, mutta sisällöllisessä täydellisyydessä ja ajankohtaisuudessa oli huomattavaa vaihtelua etenkin ELY-alueiden kesken. Alueellista vaihtelua selittää se, että Digiroad-aineistossa joukkoliikenteen pysäkeillä on pääsääntöisesti kaksi ylläpitäjää: Kunta tai ELY-keskus. ELY-keskukset vastaavat alueensa maanteillä sijaitsevista, valtion omistamista joukkoliikennepysäkeistä ja kunnat vastaavat omistamistaan joukkoliikennepysäkeistä (Pysäkkitiedon hallinta Suomessa

2017). Tutkielmassa käytetyssä Digiroad-aineistossa 69,4 prosentilla pysäkeistä on ylläpitäjänä ELY-keskus, joten tämä myötävaikuttaa alueellisiin eroihin, jotka noudattavat ELY-alueiden rajoja.

4.2 Laatutietojen automatisoinnin hyödyt ja haasteet

Laatutietojen tuottamisen ja visualisoinnin automatisoinnilla on useita etuja. Huomattavin näistä on nopeus, kuten voidaan huomata tutkielmassa tuotetun koodin suoritusajoista: Koko koodin suorittamiseen kului tutkielman yhteydessä alle 10 minuuttia, ja yli minuutin kestäneet vaiheet liittyivät aineistojen tuontiin ja esikäsittelyyn, siinä missä tietojen käsittely, laatutietojen laskeminen ja niiden tilastollinen tarkastelu sekä visualisointi kestivät lähtökohtaisesti korkeintaan minuutin per vaihe. Koodi suoriutuu nopeasti myös vanhemmalla, ei-ammattikäyttöön tarkoitetulla kannettavalla tietokoneella. Automatisointi on tehokkaampi ja nopeampi tapa tuottaa ja visualisoida tietoa verrattuna käyttöliittymän kautta tehtyyn aineiston käsittelyyn (Xavier ym. 2019). Tutkielman tapauksessa voidaan katsoa, että kyseisten laatutietojen tuottaminen ja visualisointi esimerkiksi QGISilla lähtöaineistoista vaatisi useita eri työkaluja. Laatutietojen tuottaminen ja visualisointi Pythonilla vähentää myös manuaaliseen työhön liittyviä virheitä koodin tuottaessa aina saman lopputuloksen samoja lähtöaineistoja käyttäessä. Tästä poikkeuksena on ajankohtaisuus, joka lasketaan suhteessa suoritushetken ajankohtaan, joten ajankohtaisuuden arvot muuttuvat vaikka koodiin ei tehtäisi muutoksia.

Tulosten epävarmuuteen vaikuttavat lähtöaineistojen väliset erot erityisesti pysäkkien sijainneissa. OSM-pysäkeille jouduttiin lisäämään tarvittavat kuntanumerot, jotka pääteltiin pysäkin sijainnista suhteessa Tilastokeskuksen Kunnat 2023 -aineiston sisältämiin geometrioihin. Koska kuntanumerot lisättiin valitsemalla kuntarajojen sisälle osuvat OSM-pysäkit, voi jollakin pysäkillä olla eri kuntanumero OSM-aineistossa kuin Digiroadissa. Lisäksi kunta-alueiden geometrian perusteella pysäkkien valitseminen on siitäkin syystä heikko menetelmä, että esimerkiksi rannikkoalueilla silloilla sijaitsevat pysäkit jäävät valitsematta, koska kunta-alueen geometria kattaa vain maa-alueet. Lisäksi mikäli lähtöaineistoja muutetaan, on mahdollista, ettei koodi enää toimi toivotulla tavalla. Esimerkiksi jos OpenStreetMapiin lisätään uusia joukkoliikenteen pysäkkejä ja lähtöaineistot päivitetään, tulee tämä huomioida myös koodissa, jottei aineistoon jää OSM-pysäkkejä ilman kuntanumeroa.

Laatutietojen visualisointi interaktiiviselle kartalle on haastavaa, sillä useimmat Pythoniin soveltuvat karttavisualisointityökalut eivät palvele tutkielman tarkoitusta tarpeeksi. Laatutietojen katselua ja vertailua varten olisi hyödyllistä, mikäli tietoja olisi mahdollista

suodattaa kartalla. Tutkielmassa harkittiin aluksi Kepler.gl-työkalun käyttöä, sillä se on moderni avoimen lähdekoodin paikkatiedon visualisointityökalu Pythonille, ja sen ominaisuuksiin sisältyvät datasuodattimet. Kepler.gl-pakettia ei kuitenkaan päädytty hyödyntämään tässä tutkielmassa, sillä Kepler.gl-työkalu ei salli luokitellun aineiston luokkarajojen muokkaamista halutunlaisiksi, mikä nähtiin tärkeäksi kartan luettavuuden kannalta. Tutkielmassa päädyttiin käyttämään Foliumia, sillä se on ominaisuuksiltaan enemmän muokattavissa kuin Kepler.gl, mutta Foliumissa ei ole valmiutta laadukkaisiin suodattimiin.

Visuaalisin keinoin voidaan esittää lähinnä kvantitatiivisia laatutietoja, jotka kuvaavat paikkatiedon sisäistä laatua (Vasseur ym. 2006: 263–264). Sisäinen laatu on osa ulkoista laatua, mutta käyttäjien kannalta olisi tärkeää kuvata myös kvalitatiivisia laatutietoja, sillä paikkatiedon laatu riippuu sekä sisäisestä että ulkoisesta laadusta (Harding 2006: 142) Kvalitatiivisten laatutietojen visualisointimahdollisuuksia tulisi tutkia lisää.

Laatutietojen automatisointia voisi tehostaa kehittämällä automatisointiprosessista palvelun, jossa laatutiedot laskettaisiin aina uusimmista saatavilla olevista aineistoista. Tämän tutkielman rajoissa lähtöaineistoina käytettiin paikalliseksi ladattuja aineistoja, jotta tulokset pysyvät mahdollisimman stabiileina tutkimuksen ajan. Käyttäjää voisi kuitenkin palvella enemmän se, että aineistot haetaan koodiin tiedon tuottajien tarjoamilta rajapinnoilta, jolloin analyysin kohteena olisi mahdollisimman tuore tieto. Lisäksi laatutietojen tuottamisen asetelmaa voisi muuttaa siten, että tuloksena on vertailu eri aineistojen laatutiedoista. Käyttäjien saatavilla on nykyään useita rinnakkaisia aineistoja, ja Bieleckan (2015) kyselytutkimus osoittaa, että käyttäjät kokevat rinnakkaisten paikkatietoaineistojen laadun vertailun tukevan päätöksentekoa. Laatutietojen automatisointia voisi kehittää esimerkiksi WPS- eli Web Processing Service -muodossa, sillä sen on todettu soveltuvan paikkatiedon automatisoituun laatuanalyysiin (Xavier ym. 2019).

4.3 Digiroadin pysäkkitiedon kehitystarpeet

Digiroad-aineistoon ei ole tuotettu erillisiä laatutietoja, eli käyttäjä joutuu tukeutumaan metatietoihin etsiessään sopivaa aineistoa. Erillisistä laatutiedoista on kuitenkin hyötyä tiedon tuottajan lisäksi myös käyttäjille (Devillers ym. 2007), joten erillisten laatutietojen tuottamista metatietojen yhteyteen voitaisiin nähdä tarpeellisena.

Joukkoliikenteen pysäkkien sijaintitiedon luotettavaksi arvioimiseksi tulisi arviointiin voida käyttää tien keskilinjageometriaan sidotun sijainnin sijasta maastokoordinaatteja, jotka

ilmaisevat pysäkin todellista sijaintia. Tätä tietoa ei ollut tutkielmassa käytetyssä Digiroad-aineistossa kuin 9,7 prosentilla aineiston pysäkeistä. Maastokoordinaattitietojen määrää olisi tärkeä lisätä aineistossa, sillä niitä hyödynnetään esimerkiksi reitittämiseen ja opastamiseen pysäkille (Keski-Suomen laatupalvelupilotti 2022).

Ajankohtaisuuden mittaamisen epävarmuustekijäksi muodostui se, että ajankohtaisuutta mitattiin tiedon muokkauspäivämäärästä. Pitkä aikaväli nykyhetken ja tiedon muokkausajan välillä ei suoraan tarkoita sitä, että tieto olisi väärin, sillä pysäkin ominaisuudet ovat voineet pysyä pitkään muuttumattomina maastossa. Pitkä aikaväli muokkauksesta kuitenkin lisää epävarmuutta tiedon laadusta (Servigne ym. 2006: 183). Temporaaliset tiedot ovatkin käyttäjälle yksi tärkeimmistä tiedoista, jotka tukevat tiedon soveltuvuuden arviointia käyttäjän tarpeeseen, ja siksi Digiroad-aineistossakin temporaalisten tietojen keruuta olisi tärkeä ylläpitää ja kehittää. Pysäkkitiedon ylläpitoohjeistuksen mukaan kuntien tulisi tarkastaa pysäkkitietonsa vähintään kerran vuodessa (Pysäkkitiedon hallinta Suomessa 2017), mutta jos tiedot pysyvät muuttumattomina, ei tarkastamisestä jää aikaleimaa pysäkin tietoihin.

Digiroadin tiedonlaadun parantamiseksi olisi tärkeää kartoittaa käyttäjien tarpeita lisää etenkin laatutietojen tuottamisen näkökulmasta. Myös laatutietojen visualisointiin tulisi osallistaa käyttäjiä, jotta voitaisiin kehittää parempia menetelmiä paikkatiedon laadun visualisoimiseksi (Brus & Pechanec 2015). Yang ym. (2013) teettämän käyttäjille kohdistetun kyselytutkimuksen tulosten perusteella käyttäjät saattavat arvostaa myös niin kutsuttua ”pehmeää tietoa” (engl. *soft knowledge*) tiedon laadusta, eli esimerkiksi tiedon tuottajan antamaa sanallista laatukuvausta. Sanallinen kuvaus tiedon laadusta voisi olla hyödyllistä myös Digiroadin käyttäjille. Tässä tutkielmassa Digiroadin joukkoliikenteen pysäkkitiedon laatua tutkittiin kvantitatiivisilla mittareilla, mutta kattavan laatutiedon saavuttamiseksi olisi tärkeää tehdä kvalitatiivista lisätutkimusta tiedon tuotantoon liittyvien prosessien vaikutuksista laatuun (Wu & Buttenfield 1994). Mikäli erillisiä laatutietoja tuotettaisiin Digiroadiin metatietojen ohelle, olisi tärkeää huomioida myös kvalitatiiviset laatutiedot, joita tarvitaan yhtäläisesti kvantitatiivisten laatutietojen kanssa. Tällöin käyttäjällä on mahdollisuus tehdä kattavia päätelmiä tiedon soveltuvuudesta omiin käyttötarpeisiinsa.

5 Johtopäätökset

Digiroadin pysäkkitiedon laatu on hyvällä tasolla, mutta vaihtelevaa. Pysäkkitietojen laadussa on myös jyrkkiä eroja ja paljon poikkeuksia. Pysäkkitiedon laatu vaihtelee Suomessa myös maantieteellisesti. Vaihtelu on selkeintä eri ELY-alueiden välillä, mikä selittyy ELY-keskusten vastuulla olevien maanteiden pysäkkien suuresta määrästä aineistossa. Joukkoliikenteen pysäkkitiedon laatutiedot soveltuvat hyvin geovisualisoitavaksi kunta-aluejakoon perustuvalla koropleettikartalla.

Joukkoliikenteen pysäkkitiedon laatua voidaan arvioida tavanomaisilla pistemäisten vektorikohteiden laatumittareilla, mutta arviointia vaikeuttaa sopivan referenssitiedon puute. Laatutietojen tuottamisen automatisointi pysäkkitiedon kontekstissa voidaan toteuttaa yksinkertaisilla Python-komennoilla nopeasti. Tämän vuoksi myös täysin automatisoidun laatutietopalvelun tuottaminen olisi mahdollista esimerkiksi Web Processing Service:nä.

Digiroadin joukkoliikenteen pysäkkitiedoissa tulisi erityisesti kehittää pysäkkitiedon sisällöllistä täydellisyyttä maastosijainti-, esteettömyys- ja varustetietojen osalta. Ajallisten tietojen keruuta olisi myös tarpeen kehittää, sillä tällä hetkellä käyttäjä ei pysty tulkitsemaan aineistosta milloin se on viimeksi tarkistettu. Digiroad-tietojärjestelmällä on suuri merkitys esimerkiksi reititämisen ja matkatietopalveluiden kannalta, ja laatutietojen tuottamista ja jakamista käyttäjille olisi syytä kartoittaa ja kehittää.

Kiitokset

Kiitos Väylävirastolle tuesta graduprosessissa ja erityisesti Minna Huoviselle ja Jani Lehenbergille tuesta tutkielman aloittamisessa ja aiheen määrittelyssä.

Lähteet

2010/40/EY = Euroopan parlamentin ja neuvoston direktiivi 2010/40/EU, annettu 7 päivänä heinäkuuta 2010, tieliikenteen älykkäiden liikennejärjestelmien käyttöönoton sekä tieliikenteen ja muiden liikennemuotojen rajapintojen puitteista. Hyödynnetty 8.4.2024. http://data.europa.eu/eli/dir/2010/40/2023-12-20

2007/2/EY = Euroopan parlamentin ja neuvoston direktiivi 2007/2/EY, annettu 14 päivänä maaliskuuta 2007, Euroopan yhteisön paikkatietoinfrastruktuurin (INSPIRE) perustamisesta. Hyödynnetty 15.5.2024. http://data.europa.eu/eli/dir/2007/2/oj

Alinikula, P, Kivi, M., Somerpalo, S. & Tamminen, T. (2021). Selvitys liikkumispalvelujen esteettömyystietojen määrittelyn, saatavuuden ja tuottajien tietoisuuden parantamisesta. *Liikenne- ja viestintäministeriön julkaisuja* 2021: 18. http://urn.fi/URN:ISBN:978-952-243-600-9

Bédard Y. & Valliére D. (1995) *Qualité des données à référence spatiale dans un contexte gouvernemental*. Laval University, Quebec.

Bielecka, E. (2015) Geographical data sets fitness of use evaluation. *Geodetski Vestnik* 59(2), 335–348. https://doi.org/10.15292/geodetski-vestnik.2015.02.335-348

Borkowska, S. & Pokonieczny, K. (2022). Analysis of openstreetmap data quality for selected counties in Poland in terms of sustainable development. *Sustainability* 14(7), 3728. https://doi.org/10.3390/su14073728

Brus, J. & Pechanec, V. (2014). The user centered framework for visualization of spatial data quality. *Lecture notes in Geoinformation and Cartography*, 325–338. https://doi.org/10.1007/978-3-319-07926-4_25

Caprioli, M., Scognamiglio, A., Strisciuglio, G., & Tarantino, E. (2003) *Rules and standards for spatial data quality in GIS environments*. Proceedings of the 21st International Cartographic Conference. Durban, South Africa.

CEN and CENELEC (s.a.) Hyödynnetty 15.5.2024. https://www.cencenelec.eu/european-standardization/cen-and-cenelec/

Chrisman, N. (2006). Development in the treatment of spatial data quality. *Teoksessa* Devillers, R. & Jeansoulin, R. (toim): *Fundamentals of spatial data quality*, 21–30. Wiley, New York.

Deitrick, S. & Edsall, S. (2009). Making uncertainty usable: approaches for visualizing uncertainty information. *Teoksessa* Dodge, M., McDerby, M. & Turner, M.: *Geographic visualization: Concepts, tools and applications*. Wiley, New York.

Devillers, R., Bédard, Y., & Jeansoulin, R. (2005). Multidimensional management of geospatial data quality information for its dynamic use within gis. *Photogrammetric Engineering & Remote Sensing* 71(2), 205–215. https://doi.org/10.14358/pers.71.2.205

Devillers, R. & Beard, K. (2006). Communication and use of spatial data quality information in GIS. *Teoksessa* Devillers, R. & Jeansoulin, R. (toim.): *Fundamentals of spatial data quality*, 237–253. Wiley, New York.

Devillers, R., Bédard, Y., Jeansoulin, R. & Moulin, B. (2007). Towards spatial data quality information analysis tools for experts assessing the fitness for use of spatial data. *International Journal of Geographical Information Science* 21(3), 261–282. https://doi.org/10.1080/13658810600911879

Devillers, R. & Jeansoulin, R. (2006). Spatial data quality: concepts. *Teoksessa* Devillers, R. & Jeansoulin, R. (toim). *Fundamentals of spatial data quality*, 31–42. Wiley, New York.

Digiroad: tietolajien kuvaus 4/2022 (2022). Väylävirasto, Helsinki. https://ava.vaylapilvi.fi/ava/Tie/Digiroad/Aineistojulkaisut/latest/Julkaisudokumentit/FI_Tietolajien_kuvaus_4_2022.pdf

Ďuračiová, R. (2003). An aggregated shape similarity index: a case study of comparing the footprints of OpenStreetMap and INSPIRE buildings. *ISPRS Int. J. Geo-Inf.* 12(495). https:// doi.org/10.3390/ijgi12120495

ELY-keskukset ja niiden tehtävät (2023). Elinkeino-, liikenne- ja ympäristökeskus. Hyödynnetty 31.7.2023. https://www.ely-keskus.fi/ely-keskukset

Fisher, P., Comber, A. & Wadsworth, R. (2006). Approaches to uncertainty in spatial data. Teoksessa Devillers, R. & Jeansoulin, R. (toim.): *Fundamentals of spatial data quality*, 43-59. Wiley, New York.

Fonte, C. C., Antoniou, V., Bastin, L., Estima, J., Arsanjani, J. J., Bayas, J.-C. L., See, L. & Vatseva, R. (2017). Assessing vgi quality. *Teoksessa* Foody, G., See, L., Fritz, S., Mooney, P., Olteanu-Raimond, A.-M., Fonte, C. C. & Antoniou, V. (toim.): *Mapping and the citizen sensor*, 137–164. Ubiquity Press, London. https://doi.org/10.5334/bbf

Ge, Y., Hexiang, B. and Li, S. (2008). Geo-spatial data analysis, quality assessment and visualization. *Computational Science and Its Applications* – ICCSA 2008, 258–267. https://doi.org/10.1007/978-3-540-69839-5_20

Girres, J.-F. & Touya, G. (2010). Quality assessment of the French openstreetmap dataset. *Transactions in GIS* 14(4), 435–459. https://doi.org/10.1111/j.1467-9671.2010.01203.x

Goodchild, M.F. & Clarke, K. (2002) Data quality in massive datasets. *Teoksessa* Abello, J., Pardalos, P.M. & Resende, M. G. C. (toim): *Handbook of massive datasets*, 643–659. Springer, Berlin.

Guptill, S. C. & Morrison, J. (1995) *Elements of spatial data quality*. Elsevier Science, New York.

Harding, J. (2006). Vector data quality: a data provider's perspective. *Teoksessa* Devillers, R. & Jeansoulin, R. (toim): *Fundamentals of spatial data quality*, 141–159. Wiley, New York.

Hunter, G. J., Bregt, A. K., Heuvelink, G. B. M., De Bruin, S. & Virrantaus, K. (2009). Spatial data quality: problems and prospects. *Lecture Notes in Geoinformation and Cartography*, 101–121. https://doi.org/10.1007/978-3-540-88244-2_8

ISO = International Organization for Standardization (2013). *Geographic information – Data Quality* (ISO 19157). ISO/TC 211. 146 s.

Jackson, S., Mullen, W., Agouris, P., Crooks, A., Croitoru, A. & Stefanidis, A. (2013). Assessing completeness and spatial error of features in volunteered geographic information. *ISPRS International Journal of Geo-Information* 2(2), 507–530. https://doi.org/10.3390/ijgi2020507

JHS 160 = Julkisen hallinnon suositus 160 – Paikkatiedon laadunhallinta (vanhentunut). Julkisen hallinnon tiedonhallinnon neuvottelukunta, Helsinki. 22.6.2006.

Juran, J.M., Gryna F.M. & Bingham, R.S. (1974) *Quality control handbook*. 3. p. McGraw-Hill, New York

Keski-Suomen laatupalvelupilotti – multimodaalisen joukkoliikenteen olennaisten tietojen laatupalvelu ja koonti (2022). Projektin loppuraportti. Matkahuolto, Helsinki.

Khan, Z. T. & Johnson, P. A. (2020). Citizen and government co-production of data: Analyzing the challenges to government adoption of VGI. Canadian Geographies / Géographies Canadiennes 64(3), 374–387. https://doi.org/10.1111/cag.12619

Krämer, M., Haist, J. & Reitz, T. (2007). Methods for spatial data quality of 3d city models. *Teoksessa* Amicis, R. D. & Conti, G. (toim.): *Eurographics Italian chapter conference*, 167-172.

Li, S., Dragicevic, S., Castro, F. A., Sester, M., Winter, S., Coltekin, A., Pettit, C., Jiang, B., Haworth, J., Stein, A. & Cheng, T. (2016). Geospatial big data handling theory and

methods: a review and research challenges. *ISPRS Journal of Photogrammetry and Remote Sensing* 115, 119–133. https://doi.org/10.1016/j.isprsjprs.2015.10.012

Lush, V., Bastin, L. & Lumsden, J. (2012). Geospatial data quality indicators. *Teoksessa* C. Vieira, V. Bogorny & Aquino, A. R. (toim.): *Proceedings of the 10th international symposium on spatial accuracy assessment in natural resources and environmental sciences,* 121-126.

Moellering, H. (toim.) (1987). A draft proposed standard for digital cartographic data. *Open-File Report* 87-308. https://doi.org/10.3133/ofr87308

Paikkatietoalan standardit ja suositukset (s.a.). Hyödynnetty 15.5.2024. https://geoforum.fi/paikkatietoalan-standardit-ja-suositukset/

Peltonen, T. (2016). Digiroad- ja OpenStreetMap-aineistojen yhteiskäyttö joukkoliikennepysäkeissä. Liikennevirasto, hankehallintaosasto, opinnäytetyö 3/2016. https://urn.fi/URN:ISBN:978-952-317-225-8

Pysäkkitiedon hallinta Suomessa: pysäkkitietojen ylläpito ja vastuut (2017). Liikenneviraston ohjeita 29/2017. Helsinki. https://urn.fi/URN:ISBN:978-952-317-224-1

Sarretta, A., & Minghini, M. (2021). Towards the integration of authoritative and OpenStreetMap geospatial datasets in support of the european strategy for data. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLVI-4/W2-2021, 159–166. https://doi.org/10.5194/isprs-archives-xlvi-4-w2-2021-159-2021

Servigne, S., Lesage, N. & Libourel, T. (2006). Quality components, standards, and metadata. *Teoksessa* Devillers, R. & Jeansoulin, R. (toim): *Fundamentals of spatial data quality*, 179–210. Wiley, New York.

Tiedon laatukehikko (s.a). Hyödynnetty 15.5.2024. https://stat.fi/org/tiedon-laatukehikko/index.html

Timpf S, Raubal M, Kuhn W (1997) Experiences with metadata. Proceedings of the 7th International Symposium on Spatial Data Handling

Timpf, S., M. Raubal, & Kuhn, W. (1996). Experiences with metadata. *Proceedings of Symposium on Spatial Data Handling, Advances in GIS Research II*, s. 31–43.

Vasseur, B., Jeansoulin, R., Devillers, R. & Frank, A. (2006). External quality evaluation of geographical applications: an ontological approach. *Teoksessa* Devillers, R. & Jeansoulin, R. (toim): *Fundamentals of spatial data quality*, 255–270. Wiley, New York.

Väylätiedon hyödyntäminen -selvitys: projektiraportti (2021). Väylävirasto, Helsinki. Hyödynnetty 8.4.2024. https://vayla.fi/documents/25230764/35601500/V%C3%A4yl%C3%A4tietojenHy%C3%B6dynt%C3%A4minen_Projektiraportti_NettisivuVersio_2021_04_21.pdf/3235c0eb-80ed-7a76-6a87-d1da2037b52a/V%C3%A4yl%C3%A4tietojenHy%C3%B6dynt%C3%A4minen_Projektiraportti_NettisivuVersio_2021_04_21.pdf?t=1620221613470

Wang, R. Y. & Strong, D. M. (1996). Beyond accuracy: what data quality means to data consumers. *Journal of Management Information Systems* 12, 5–34.

Westland, J. C. (2002). The cost of errors in software development: evidence from industry. *Journal of Systems and Software* 62(1), 1–9. https://doi.org/10.1016/S0164-1212(01)00130-3.

Xavier, E. M., Ariza-López, F. J. & Ureña-Cámara, M. A. (2019). Automatic evaluation of geospatial data quality using web services. *Revista Cartográfica* 98, 59–73. https://doi.org/10.35424/rcarto.i98.141

Yagoub, M. M. (2017). Assessment of openstreetmap (OSM) data: The case of Abu Dhabi City, United Arab Emirates. *Journal of Map & Geography Libraries* 13(3), 300–319. https://doi.org/10.1080/15420353.2017.1378150

Yang, X., Blower, J. D., Bastin, L., Lush, V., Zabala, A., Masó, J., Cornford, D., Díaz, P. & Lumsden, J. (2013). An integrated view of data quality in earth observation. *Phil Trans R Soc A:* 371:20120072. https://doi.org/10.1098/rsta.2012.0072

Liitteet

Liite 1. Digiroadin Joukkoliikenteen pysäkki -aineiston tietokuvaus. Yhdistelty Digiroad: tietolajien kuvaus (2022) ja Pysäkkitiedon hallinta Suomessa (2017).

Selite	Tyyppi	Pakollinen	Generoituu automaattisesti
Koordinaatti X	Desimaaliluku	-	Kyllä
Koordinaatti Y	Desimaaliluku	-	Kyllä
Linkin Link-ID	Merkkijono	-	Kyllä
M-arvo	Desimaaliluku	-	Kyllä
Vaikutussuunta	Kokonaisluku (koodiarvo)	Kyllä	Ei
Muokattu viimeksi	Merkkijono	-	Kyllä
Valtakunnallinen ID	Kokonaisluku	-	Kyllä
Nimi suomeksi	Merkkijono	Kyllä	Ei
Nimi ruotsiksi	Merkkijono	Ei	Ei
Tietojen ylläpitäjä	Kokonaisluku (koodiarvo)	Kyllä	Ei
Ylläpitäjän tunnus	Merkkijono	Ei	Ei
Livi-tunnus	Merkkijono	-	Kyllä
Matkustajatunnus	Merkkijono	Ei	Ei
Maastokoordinaatti X	Merkkijono	Ei	Ei
Maastokoordinaatti Y	Merkkijono	Ei	Ei
Maastokoordinaatti Z	Merkkijono	Ei	Ei
Liikennöintisuunta	Merkkijono	Ei	Ei
Liikennöintisuuntima	Luku	-	Kyllä
Ensimmäinen voimassaolopäivä	Merkkijono	Ei	Ei
Viimeinen voimassaolopäivä	Merkkijono	Ei	Ei
Pysäkin tyyppi	Merkkijono (koodiarvo)	Kyllä	Ei
Irti geometriasta	Kokonaisluku (koodiarvo)	-	Kyllä
Vyöhyke	Merkkijono	-	Ei
Palvelutasoluokka	Kokonaisluku (koodiarvo)	-	-
Aikataulu	Kokonaisluku (koodiarvo)	Ei	Ei
Katos	Kokonaisluku (koodiarvo)	Ei	Ei
Mainoskatos	Kokonaisluku (koodiarvo)	Ei	Ei
Penkki	Kokonaisluku (koodiarvo)	Ei	Ei
Pyöräteline	Kokonaisluku (koodiarvo)	Ei	Ei
Sähköinen aikataulunäyttö	Kokonaisluku (koodiarvo)	Ei	Ei

Valaistus	Kokonaisluku (koodiarvo)	Ei	Ei
Saattomahdollisuus henkilöautolla	Kokonaisluku (koodiarvo)	Ei	Ei
Esteettömyys liikuntarajoitteisille	Merkkijono	Ei	Ei
Liityntäpysäköintipaikkojen lkm	Merkkijono	Ei	Ei
Liityntäpysäköinnin lisätiedot	Merkkijono	Ei	Ei
Pysäkin omistaja	Merkkijono	Ei	Ei
Palauteosoite	Merkkijono	Ei	Ei
Lisätiedot	Merkkijono	Ei	Ei
Kuntanumero	Kokonaisluku	-	Kyllä

Liite 2. OpenStreetMap-pysäkit, joille lisätään kuntakoodi manuaalisesti

Kunta	**Geometrioiden ulkopuoliset pysäkit (valtakunnallinen ID)**
Espoo	60761224, 1398792779, 6092621568
Helsinki	370307691, 408775467, 475714793, 475714796, 488284845, 3050055694, 7008571115, 11037074729
Pori	897518664, 897518931, 6271081682, 6271081683
Naantali	6259555671, 6259555680, 6259556238
Turku	6090192118, 6090192158
Parainen	2764596229, 2961579645, 2961579646, 6259556851, 6259556852, 9766537877
Kaarina	6259558401, 6259561483
Kotka	6270886981, 6270887644
Vöyri	6270969691, 6270969726
Oulu	1815107856, 1079073483
Hailuoto	6271025210
Kustavi	6259556831
Luoto	6270971243
Salo	6111500407
Kokkola	4859616741
Inkoo	4927890428

Liite 3. Digiroad-pysäkkien tietokohtainen sisällöllinen täydellisyys. Sisällöllinen täydellisyys on laskettu vain sellaisille ominaisuustiedoille, jotka eivät muodostu automaattisesti Digiroadissa (Pysäkkitiedon hallinta Suomessa 2017).

Ominaisuustieto	**Selite**	**Pakollinen**	**Sisällöllinen täydellisyys**
nimi_su	Nimi suomeksi	Kyllä	99,63 %
nimi_ru	Nimi ruotsiksi	Ei	27,25 %
yllapitaja	Tietojen ylläpitäjä	Kyllä	100,00 %
yllap_tunnus	Ylläpitäjän tunnus	Ei	28,41 %
matk_tunnus	Matkustajatunnus	Ei	24,13 %

maast_x	Maastokoordinaatti X	Ei	9,70 %
maast_y	Maastokoordinaatti Y	Ei	9,70 %
maast_z	Maastokoordinaatti Z	Ei	0,00 %
liik_suunta	Liikennöintisuunta	Ei	16,11 %
ens_vo_pv	Ensimmäinen voimassaolopäivä	Ei	97,07 %
viim_vo_pv	Viimeinen voimassaolopäivä	Ei	0,79 %
pys_tyyppi	Pysäkin tyyppi	Kyllä	100 ,00 %
aikataulu	Aikataulu	Ei	7,99 %
katos	Katos	Ei	87,71 %
penkki	Penkki	Ei	7,50 %
mainoskat	Mainoskatos	Ei	6,78 %
pyoratelin	Pyöräteline	Ei	7,18 %
s_aikataul	Sähköinen aikataulunäyttö	Ei	6,31 %
korotettu	Korotettu	Ei	62,51 %
estettomyy	Esteettömyys liikuntarajoitteisille	Ei	85,72 %
saattomahd	Saattomahdollisuus henkilöautolla	Ei	1,02 %
liit_lkm	Liityntäpysäköintipaikkojen lkm	Ei	0,04 %
liit_lisat	Liityntäpysäköinnin lisätiedot	Ei	0,74 %
pys_omist	Pysäkin omistaja	Ei	1,44 %
palaute_os	Palauteosoite	Ei	0,14 %
lisatiedot	Lisätiedot	Ei	7,65 %
palvelutasoluokka	Palvelutasoluokka	-	100,00 %

Liite 4. Python-koodi laatutietojen automaattiseen tuottamiseen ja visualisoimiseen Digiroadin pysäkkitiedoista.

```
# %%
# Import needed Python modules
import geopandas as gp
import numpy as np
import pandas as p
import matplotlib.pyplot as plt
import datetime
import folium
import folium.plugins
import scipy.stats
import mapclassify
from xyzservices import TileProvider

# %% [markdown]
# LOADING AND PREPROCESSING THE DATA

# %%
# Load the Digiroad stop data by ELY Centre area and insert the ELY ID to the data
dr_uudely = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\Graduaineistot_DR\uudely.gpkg")
```

```
dr_uudely.insert(0, 'ely', 1)
dr_varely = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\Graduaineistot_DR\varely.gpkg")
dr_varely.insert(0, 'ely', 2)
dr_pirely = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\Graduaineistot_DR\pirely.gpkg")
dr_pirely.insert(0, 'ely', 5)
dr_kasely = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\Graduaineistot_DR\kasely.gpkg")
dr_kasely.insert(0, 'ely', 6)
dr_posely = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\Graduaineistot_DR\posely.gpkg")
dr_posely.insert(0, 'ely', 8)
dr_kesely = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\Graduaineistot_DR\kesely.gpkg")
dr_kesely.insert(0, 'ely', 10)
dr_epoely = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\Graduaineistot_DR\epoely.gpkg")
dr_epoely.insert(0, 'ely', 11)
dr_popely = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\Graduaineistot_DR\popely.gpkg")
dr_popely.insert(0, 'ely', 13)
dr_lapely = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\Graduaineistot_DR\lapely.gpkg")
dr_lapely.insert(0, 'ely', 15)

# Create a list of the DR stop data by county
dr_stops_list = [dr_uudely, dr_varely, dr_pirely, dr_kasely, dr_posely, dr_kesely,
dr_epoely, dr_popely, dr_lapely]

# Create a function to replace missing values in data
def replaceWithNan(dr_stops_list):
    return dr_stops_list.infer_objects(copy=False).replace(r'^\s*$', np.nan,
regex=True)
# Use the function to replace missing values with NaN
dr_stops_list = [replaceWithNan(dr_stops_list) for dr_stops_list in dr_stops_list]

# Merge the separated DR stop data to a single GeoDataFrame
dr_stops_all = p.concat([dr_stops_list[0], dr_stops_list[1], dr_stops_list[2],
dr_stops_list[3], dr_stops_list[4], dr_stops_list[5], dr_stops_list[6],
                         dr_stops_list[7], dr_stops_list[8]],
ignore_index=True)

# Replace 99 values with Nan
for i in range(len(dr_stops_all)):
    if dr_stops_all.at[i,'aikataulu'] == 99:
        dr_stops_all.at[i,'aikataulu'] = np.nan
    if dr_stops_all.at[i,'penkki'] == 99:
        dr_stops_all.at[i,'penkki'] = np.nan
```

```
    if dr_stops_all.at[i,'katos'] == 99:
        dr_stops_all.at[i,'katos'] = np.nan
    if dr_stops_all.at[i,'s_aikataul'] == 99:
        dr_stops_all.at[i,'s_aikataul'] = np.nan
    if dr_stops_all.at[i,'korotettu'] == 99:
        dr_stops_all.at[i,'korotettu'] = np.nan
    if dr_stops_all.at[i,'pyoratelin'] == 99:
        dr_stops_all.at[i,'pyoratelin'] = np.nan
    if dr_stops_all.at[i,'mainoskat'] == 99:
        dr_stops_all.at[i,'mainoskat'] = np.nan
    if dr_stops_all.at[i,'saattomahd'] == 99:
        dr_stops_all.at[i,'saattomahd'] = np.nan
    if dr_stops_all.at[i,'valaistus'] == 99:
        dr_stops_all.at[i,'valaistus'] = np.nan
    if dr_stops_all.at[i,'yllapitaja'] == 99:
        dr_stops_all.at[i,'yllapitaja'] = np.nan
    if dr_stops_all.at[i,'pys_tyyppi'] == 99:
        dr_stops_all.at[i,'pys_tyyppi'] = np.nan

# %%
# Load municipality (mun) and ELY Centre (ely) polygon data. These dataframes will
host the quality information.
ely_QI = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\Taustaaineistot\elyalueet2023.gpkg")
municipality_QI = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\Taustaaineistot\kuntajako2023.gpkg")
# Sort ELY polygon data to be in order by ELY Id
ely_QI = ely_QI.sort_values(by='ely', ascending=True, ignore_index=True)
# Convert ELY and municipality Id numbers to int64
ely_QI = ely_QI.astype({'ely':'int64'})
municipality_QI = municipality_QI.astype({'kunta':'int64'})

# Add ELY Centre name abbreviations to the ELY polygon data
ely_id = ['UUDELY', 'VARELY', 'PIRELY', 'KASELY', 'POSELY', 'KESELY', 'EPOELY',
'POPELY', 'LAPELY']
ely_QI['lyhenne'] = "" # Create empty column for abbreviation
i = 0
while i < len(ely_QI):
    ely_QI.iloc[i,6] = ely_id[i]
    i += 1

# Add ELY Centre name abbreviations to municipality polygon data
municipality_QI['ely'] = "" # Create empty column for abbreviation

i = 0
while i < len(ely_QI):
    geometry = ely_QI.geometry[i] # Select a ELY geometry
    k = 0
    while k < len(municipality_QI):
```

```
        if municipality_QI.iloc[k,5].within(geometry):
            municipality_QI.iloc[k,6] = ely_QI.iloc[i,6] # Add ELY abbreaviation
if municipality geometry within ELY geometry
        k += 1
    i += 1

# Calculate the number of bus stops in a munincipality
stop_count = dr_stops_all.groupby('kuntakoodi').size()
stop_count = stop_count.reset_index()
stop_count.rename(columns={0:'pysakkien_maara','kuntakoodi':'kunta'},
inplace=True)
municipality_QI = municipality_QI.merge(stop_count, how='outer', on=['kunta'])

# Create a list of the munincipal codes in DR stop data and sort it
municipal_id_dr = dr_stops_all.kuntakoodi.unique()
municipal_id_dr.sort()

# %%
# Load OSM stop data separated by ELY Centre area
osm_uudely = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\OSM-aineistot\osm_uudely.gpkg")
osm_varely = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\OSM-aineistot\osm_varely.gpkg")
osm_pirely = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\OSM-aineistot\osm_pirely.gpkg")
osm_kasely = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\OSM-aineistot\osm_kasely.gpkg")
osm_posely = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\OSM-aineistot\osm_posely.gpkg")
osm_kesely = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\OSM-aineistot\osm_kesely.gpkg")
osm_epoely = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\OSM-aineistot\osm_epoely.gpkg")
osm_popely = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\OSM-aineistot\osm_popely.gpkg")
osm_lapely = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\OSM-aineistot\osm_lapely.gpkg")

# Create list from OSM geodataframes
osm_stops_list = [osm_uudely, osm_varely, osm_pirely, osm_kasely, osm_posely,
osm_kesely, osm_epoely, osm_popely, osm_lapely]

# Load the OSM data including all stops in Finland
osm_stops_all = gp.read_file(r"C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\OSM-aineistot\kokosuomi.gpkg")

# %%
# OSM data doesn't have munincipal Id numbers, so they need to be added to the
dataframes
```

```
168
169 # Create new dataframe for OSM stops with munincipal Id
170 osmStops_munId = p.DataFrame(columns = ['full_id', 'osm_id', 'osm_type',
171 'highway', 'ref:findr', 'name:sv', 'name:fi', 'name', 'geometry', 'kunta'])
172 municipal_id_int = municipality_QI.kunta.unique() # Save munincipal Id numbers
173 from polygon data to a variable
174
175 i = 0
176 while i < len(municipal_id_dr):
177     geometry = municipality_QI.geometry[i] # Select municipality geometry
178     osmSubset = osm_stops_all[osm_stops_all.within(geometry)] # Create subset from
179 OSM stops by municipality geometry
180     osmSubset.loc[slice(None), 'kunta'] = str(municipality_QI.iloc[i,0]) # Add
181 municipality Id from geometry
182     osmStops_munId = p.concat([osmStops_munId, osmSubset], axis=0,
183 ignore_index=True, copy=False) # Add OSM stops to the new dataframe
184     i = i + 1
185
186 # Some OSM stops are outside of municipality geometries, so they need to have
187 municipality Id numbers added manually
188 # Create a new column for municipality Id to the original OSM dataframe
189 osm_stops_all['kunta'] = np.nan
190
191 # Add missing munincipal Id numbers to stops
192 missingStops_espoo = ['60761224', '1398792779', '6092621568']
193 osm_stops_all.loc[osm_stops_all['osm_id'].isin(missingStops_espoo), 'kunta'] =
194 '49'
195 missingStops_helsinki = ['370307691', '408775467', '475714793', '475714796',
196 '488284845', '3050055694', '7008571115', '11037074729']
197 osm_stops_all.loc[osm_stops_all['osm_id'].isin(missingStops_helsinki), 'kunta'] =
198 '91'
199 missingStops_pori = ['897518664', '897518931', '6271081682', '6271081683']
200 osm_stops_all.loc[osm_stops_all['osm_id'].isin(missingStops_pori), 'kunta'] =
201 '609'
202 missingStops_parainen = ['2764596229', '2961579645', '2961579646', '6259556851',
203 '6259556852', '9766537877']
204 osm_stops_all.loc[osm_stops_all['osm_id'].isin(missingStops_parainen), 'kunta'] =
205 '445'
206 missingStops_oulu = ['1815107856', '10790734838']
207 osm_stops_all.loc[osm_stops_all['osm_id'].isin(missingStops_oulu), 'kunta'] =
208 '564'
209 missingStops_naantali = ['6259555671', '6259555680', '6259556238']
210 osm_stops_all.loc[osm_stops_all['osm_id'].isin(missingStops_naantali), 'kunta'] =
211 '529'
212 missingStops_turku = ['6090192118', '6090192158']
213 osm_stops_all.loc[osm_stops_all['osm_id'].isin(missingStops_turku), 'kunta'] =
214 '853'
215 missingStops_kaarina = ['6259558401', '6259561483']
```

```
osm_stops_all.loc[osm_stops_all['osm_id'].isin(missingStops_kaarina), 'kunta'] =
'202'
missingStops_kotka = ['6270886981', '6270887644']
osm_stops_all.loc[osm_stops_all['osm_id'].isin(missingStops_kotka), 'kunta'] =
'285'
missingStops_voyri = ['6270969691', '6270969726']
osm_stops_all.loc[osm_stops_all['osm_id'].isin(missingStops_voyri), 'kunta'] =
'946'
osm_stops_all.loc[osm_stops_all['osm_id'] == '6271025210', 'kunta'] = '72' #
Hailuoto
osm_stops_all.loc[osm_stops_all['osm_id'] == '6270971243', 'kunta'] = '440' #
Luoto
osm_stops_all.loc[osm_stops_all['osm_id'] == '6259556831', 'kunta'] = '304' #
Kustavi
osm_stops_all.loc[osm_stops_all['osm_id'] == '6111500407', 'kunta'] = '734' # Salo
osm_stops_all.loc[osm_stops_all['osm_id'] == '4859616741', 'kunta'] = '272' #
Kokkola
osm_stops_all.loc[osm_stops_all['osm_id'] == '4927890428', 'kunta'] = '149' #
Inkoo

# Add the manually filled stops to the dataframe
osmStops_munId = p.concat([osmStops_munId,
osm_stops_all[osm_stops_all['kunta'].notna()]], axis = 0, ignore_index=True,
copy=False)

# Check that all OSM stops have a munincipal Id
print(osmStops_munId['kunta'].isna().sum())

# Convert column type to integer
osmStops_munId['kunta'] = osmStops_munId['kunta'].astype('int64')

# %% [markdown]
# ATTRIBUTE COMPLETENESS

# %%
# Calculate attribute completeness for every DR bus stop
attrCompletenessList = []
i = 0
while i < len(dr_stops_all):
    completeness = 100 - dr_stops_all.iloc[i].isna().sum().sum() /
len(dr_stops_all.columns) * 100 # Count the completeness value
    attrCompletenessList.append(completeness) # Add to the empty list
    i = i + 1
# Insert the list filled with completeness values to the stop data
dr_stops_all.insert(0, 'sisällöllinenTäydellisyys', attrCompletenessList)

# %%
# Calculate attribute completeness for ELY Centre areas from DR stop data
elyCompletenessList = []
```

```
elyId = ely_QI.ely.unique() # Save ELY Id number as a variable 'elyId'

i = 0
while i < len(dr_stops_list):
  elySubset = dr_stops_all.loc[dr_stops_all['ely'] == elyId[i]] # Select stops by
ELY Id
  completeness = elySubset['sisällöllinenTäydellisyys'].mean() # Calculate mean
complenetess for ELY area
  elyCompletenessList.append(completeness) # Add to list
  i = i + 1

# Add mean completeness values to ELY polygon data
ely_QI.insert(0, 'sisällöllinenTäydellisyys', elyCompletenessList)

# %%
# Calculate mean attribute completeness for munincipalities
municipalCompletenessList = []
i = 0
while i < len(municipal_id_dr):
    municipalSubset = dr_stops_all.loc[dr_stops_all['kuntakoodi'] ==
municipal_id_dr[i]]
    completeness = municipalSubset['sisällöllinenTäydellisyys'].mean()
    municipalCompletenessList.append(completeness)
    i = i + 1

# Sort the municipality polygon data by munincipal Id
municipality_QI.sort_values(by=['kunta'])

# Create new dataframe from munincipal attribute completeness and municipality Id
municipalCompletenessList_df = p.DataFrame(list(zip(municipal_id_dr,
municipalCompletenessList)), columns =['kunta', 'sisällöllinenTäydellisyys'])
# Convert municipality Id to integer in the polygon data
municipality_QI['kunta'] = municipality_QI['kunta'].astype('int64')
# Merge attribute completeness dataframe to municipality polygon data by
municipality Id
municipality_QI = municipality_QI.merge(municipalCompletenessList_df, how='outer',
on=['kunta'])


#%%
# Calculate attribute completeness
noNanCells = dr_stops_all.count(axis=0) # Count no-NA cells
attributeCompleteness_df = p.DataFrame(data={'noNanCellsCount':noNanCells}) #
Create dataframe for attribute completeness
attributeCompleteness_df = attributeCompleteness_df.reset_index() # Reset
index to move attribute names into column
attributeCompleteness_df =
attributeCompleteness_df.rename(columns={'index':'attribute'}) # Rename
attribute names columns
```

```
attributeCompleteness_df['completeness'] = 0.0 # Create empty column for completeness values

for i in range(len(attributeCompleteness_df)):
    completeness = attributeCompleteness_df.at[i,'noNanCellsCount'] / len(dr_stops_all) * 100
    attributeCompleteness_df.at[i,'completeness'] = completeness

attributeCompleteness_df = attributeCompleteness_df.round({'completeness':2})
# Round completeness to 2 decimals
attributeCompleteness_df.head(50)


# %% [markdown]
# TIMELINESS

# %%
# Create empty list for stop timeliness information
stopTimeliness = []
# Set current datetime
currentDate = p.to_datetime('today').normalize()

# Calculate timeliness for stops
i = 0
while i < len(dr_stops_all):
    stopSubset = dr_stops_all['muokkauspv'][i] # Select stop modifying time
    stopSubset_date = datetime.datetime.strptime(stopSubset, "%d.%m.%Y %H:%M:%S")
# Convert datetime
    stopSubset_difference = (currentDate - stopSubset_date).days / 365 # Convert to years
    stopTimeliness.append(stopSubset_difference) # Add to list
    i = i + 1
# Insert timeliness values to DR stop data
dr_stops_all.insert(0, 'ajankohtaisuus', stopTimeliness)

# %%
# Calculate mean timeliness values for munincipalities
munTimeliness = []
i = 0
while i < len(municipal_id_dr):
    municipalSubset = dr_stops_all.loc[dr_stops_all['kuntakoodi'] == municipal_id_dr[i]]
    munTimelinessMean = municipalSubset['ajankohtaisuus'].mean()
    munTimeliness.append(munTimelinessMean)
    i = i + 1
# Create dataframe from timeliness values and municipality Id
munTimeliness_df = p.DataFrame(list(zip(municipal_id_dr, munTimeliness)), columns =['kunta', 'ajankohtaisuus'])
# Merge timeliness values to municipality polygon data by munincipalit Id
```

```
361 municipality_QI = municipality_QI.merge(munTimeliness_df, how='outer',
362 on=['kunta'])
363
364 # %%
365 # Calculate mean timeliness values for ELY area
366 elyTimeliness = []
367 i = 0
368 while i < len(dr_stops_list):
369     elySubset = dr_stops_all.loc[dr_stops_all['ely'] == elyId[i]]
370     elyTimelinessMean = elySubset['ajankohtaisuus'].mean()
371     elyTimeliness.append(elyTimelinessMean)
372     i = i + 1
373 # Add mean timeliness to ELY polygon data
374 ely_QI.insert(0, 'ajankohtaisuus', elyTimeliness)
375
376 # %% [markdown]
377 # DATA COMPLETENESS
378
379 # %%
380 # Calculate data completeness for munincipalities into a empty list
381 municipalDataCompleteness_list = []
382 municipal_id_osm = osmStops_munId.kunta.unique() # Save municipality Id numbers
383 from OSM data into a variable
384 i = 0
385 while i < len(municipal_id_osm):
386     municipalSubset_dr = dr_stops_all.loc[dr_stops_all['kuntakoodi'] ==
387 municipal_id_dr[i]] # Subset DR stops by municipality Id
388     municipalSubset_osm = osmStops_munId.loc[osmStops_munId['kunta'] ==
389 municipal_id_osm[i]]# Subset OSM stops by municipality Id
390     completeness = len(municipalSubset_dr) / len(municipalSubset_osm) * 100 #
391 Calculate spatial completeness
392     municipalDataCompleteness_list.append(completeness)
393     i = i + 1
394 ###Luodaan tyhjä taulukko alueelliselle täydellisyydelle kunnittain
395 municipalDataCompleteness_df = p.DataFrame(list(zip(municipal_id_dr,
396 municipalDataCompleteness_list)), columns =['kunta', 'alueellinenTäydellisyys'])
397 ###Lisätään luvut kunta-aineistoon
398 municipality_QI = municipality_QI.merge(municipalDataCompleteness_df, how='outer',
399 on=['kunta'])
400
401 # %%
402 # Calculate mean data completeness values for ELY regions
403 elyDataCompleteness = []
404 i = 0
405 while i < len(ely_QI):
406     geometry = ely_QI.geometry[i] # Select geometry of Ely region
407     elyRegionMunincipalities = municipality_QI[municipality_QI.within(geometry)] #
408 Select mean data completeness values of munincipalities in ELY region
```

```
    completeness = elyRegionMunincipalities['alueellinenTäydellisyys'].mean() #
Calculate mean data completeness for ELY region
    elyDataCompleteness.append(completeness) # Append to a list
    i = i + 1
# Insert data completeness values in ELY regions to the ELY quality information
data
ely_QI.insert(0, 'alueellinenTäydellisyys', elyDataCompleteness)

# %% [markdown]
# POSITIONAL ACCURACY

# %%
# In order to asses positional accuracy, the DR and OSM stops need to be data
matched
# Rename some od the column in order to match the column names in both dataframes
osmStops_munId.rename(columns = {'ref:findr':'valtak_id'}, inplace=True)
# Change national stop id to a string in both dataframes
osmStops_munId = osmStops_munId.astype({'valtak_id':'str'})
dr_stops_all = dr_stops_all.astype({'valtak_id':'str'})
# Create subsets of both dataframes to remove unnecessary columns
dr_stops_all_subset = dr_stops_all.loc[:,['valtak_id', 'nimi_su', 'nimi_ru',
'yllapitaja', 'kuntakoodi', 'geometry']].copy()
osm_stops_all_subset = osmStops_munId.loc[:,['valtak_id', 'name:sv', 'name:fi',
'kunta', 'geometry']].copy()
# Perform a join based on the national stop id number. Not all OSM stops have this
information, so the new dataframe will have less features than the original DR
dataframe.
dr_osm_merged = p.merge(dr_stops_all_subset, osm_stops_all_subset,
on=['valtak_id'], suffixes=(None, '_osm'))

# %%
# Calculate the relative distance between matched stops
distances = []
i = 0
while i < len(dr_osm_merged):
    dr_coordinates = dr_osm_merged.iloc[i,5]
    osm_coordinates = dr_osm_merged.iloc[i,9]
    distance = dr_coordinates.distance(osm_coordinates)
    distances.append(distance)
    i = i + 1
# Insert distance values (metres) into the dataframe
dr_osm_merged.insert(0, 'sijaintitarkkuus_m', distances)

# %%
# Sort merged stop data by municipality id
dr_osm_merged.sort_values('kuntakoodi', inplace=True)

# Calculate positional accuracty for stops in munincipalities
munPositionalAccuracy = []
```

```
i = 0
while i < len(municipal_id_dr):
    muncipalSubset = dr_osm_merged.loc[dr_osm_merged['kuntakoodi'] ==
municipal_id_dr[i]] # Create subset of bus stops in municipality
    distanceOver30m = muncipalSubset.loc[muncipalSubset['sijaintitarkkuus_m']
> 30] # Create another subset of the municipality's bus stops that have over 30 m
distance between DR and OSM datasets
    positionalAccuracy = 100 - len(distanceOver30m) / len(muncipalSubset) * 100
# Calculate the percent of positionally accurate stops (distance under 30 m) in
the municipality
    munPositionalAccuracy.append(positionalAccuracy) # Append to list
    i = i + 1

# Insert positional accuracy values in to the municipality qualtiy information
data
municipality_QI.insert(0, 'sijaintitarkkuus', munPositionalAccuracy)

# %%
# Calculate mean positional accuracy values for ELY regions
elyPositionalAccuracy = []
i = 0
while i < len(ely_QI):
    geometry = ely_QI.geometry[i]
    positionalAccuracySubSet = municipality_QI[municipality_QI.within(geometry)] #
Create a subset of the munincipalities' positional accuracy in ELY region
    elyPositionalAccuracyMean =
positionalAccuracySubSet['sijaintitarkkuus'].mean() # Calculate mean
    elyPositionalAccuracy.append(elyPositionalAccuracyMean) # Append mean
positinal accuracy to list
    i = i + 1
ely_QI.insert(0, 'sijaintitarkkuus', elyPositionalAccuracy)

# %% [markdown]
# EXPORTING THE QUALITY INFORMATION

# %%
ely_QI.to_excel(r'C:\Users\Jenni Autere\Documents\UTU\FM-vaihe\Gradu_hommia\IPY-
kirjastot\gradu_loppuaineisto_elyt.xlsx', engine='xlsxwriter')
municipality_QI.to_excel(r'C:\Users\Jenni Autere\Documents\UTU\FM-
vaihe\Gradu_hommia\IPY-kirjastot\gradu_loppuaineisto_kunnat.xlsx',
engine='xlsxwriter')

# %% [markdown]
# VISUALIZATION

# %%
# Calculate Kendall's tau correlation coefficient between quality indicators in
municipalities
```

```
DC_AC = scipy.stats.kendalltau(municipality_QI['alueellinenTäydellisyys'],
municipality_QI['sisällöllinenTäydellisyys'])
DC_TI = scipy.stats.kendalltau(municipality_QI['alueellinenTäydellisyys'],
municipality_QI['ajankohtaisuus'])
DC_PA = scipy.stats.kendalltau(municipality_QI['alueellinenTäydellisyys'],
municipality_QI['sijaintitarkkuus'])
AC_TI = scipy.stats.kendalltau(municipality_QI['sisällöllinenTäydellisyys'],
municipality_QI['ajankohtaisuus'])
AC_PA = scipy.stats.kendalltau(municipality_QI['sisällöllinenTäydellisyys'],
municipality_QI['sijaintitarkkuus'])
TI_PA = scipy.stats.kendalltau(municipality_QI['ajankohtaisuus'],
municipality_QI['sijaintitarkkuus'])

# Calculate Kendall's tau correlation coefficient between quality indicators and
the bus stop count in municipalities
DC_BC = scipy.stats.kendalltau(municipality_QI['alueellinenTäydellisyys'],
municipality_QI['pysakkien_maara'])
AC_BC = scipy.stats.kendalltau(municipality_QI['sisällöllinenTäydellisyys'],
municipality_QI['pysakkien_maara'])
TI_BC = scipy.stats.kendalltau(municipality_QI['ajankohtaisuus'],
municipality_QI['pysakkien_maara'])
PA_BC = scipy.stats.kendalltau(municipality_QI['sijaintitarkkuus'],
municipality_QI['pysakkien_maara'])

print('The Kendalls Tau between data and attribute completeness is:', DC_AC)
print('The Kendalls Tau between data completeness and timeliness is:', {DC_TI})
print('The Kendalls Tau between data completeness and positional accuracy is:',
{DC_PA})
print('The Kendalls Tau between attribute completeness and timeliness is:',
{AC_TI})
print('The Kendalls Tau between attribute completeness and positional accuracy
is:', {AC_PA})
print('The Kendalls Tau between timeliness and positional accuracy is:', {TI_PA})
print('The Kendalls Tau between data completeness and bus stop count is:',
{DC_BC})
print('The Kendalls Tau between attribute completeness and bus stop count is:',
{AC_BC})
print('The Kendalls Tau between timeliness and bus stop count is:', {TI_BC})
print('The Kendalls Tau between positional accuracy and bus stop count is:',
{PA_BC})


# %%
# Aggregate statistics for quality indicators in munincipalities
municipality_QI.agg(
    {
        "sisällöllinenTäydellisyys": ["min", "max", "mean", "median", "skew",
"kurt", "std"],
```

```
        "alueellinenTäydellisyys": ["min", "max", "mean", "median", "skew",
"kurt", "std"],
        "ajankohtaisuus": ["min", "max", "mean", "median", "skew", "kurt", "std"],
        "sijaintitarkkuus": ["min", "max", "mean", "median", "skew", "kurt",
"std"],
    }
)

# %%
# Create box plots by ELY regions and set axis labels and title
bp1 = municipality_QI.boxplot(column=['sisällöllinenTäydellisyys'], by="ely",
figsize=[10,6], rot=30, vert=False)
bp1.set_xlabel('sisällöllinen täydellisyys (%)')
bp1.set_ylabel('')
bp1.set_title('')
bp2 = municipality_QI.boxplot(column=['alueellinenTäydellisyys'], by="ely",
figsize=[10,6], rot=30, vert=False)
bp2.set_xlabel('alueellinen täydellisyys (%)')
bp2.set_ylabel('')
bp2.set_title('')
bp3 = municipality_QI.boxplot(column=['ajankohtaisuus'], by="ely", figsize=[10,6],
rot=30, vert=False)
bp3.set_xlabel('ajankohtaisuus (vuosi)')
bp3.set_ylabel('')
bp3.set_title('')
bp4 = municipality_QI.boxplot(column=['sijaintitarkkuus'], by="ely",
figsize=[10,6], rot=30, vert=False)
bp4.set_xlabel('sijaintitarkkuus')
bp4.set_ylabel('')
bp4.set_title('')


# %%
# Create histograms of quality indicators in municipalities
ax = municipality_QI.hist(column='ajankohtaisuus',
                          bins=[0,1,2,3,4,5,6,7,8], # Set bins
                          rwidth=0.9, # Bar witdh
                          color='#86bf91', # Bar color
                          grid=False, # Hide grid
                          zorder=2) # Set bars to be in front of horizontal axis
lines
bx = municipality_QI.hist(column='sijaintitarkkuus',
                          bins=[75,76,77,78,79,80,81,82,83,85,86,87,88,89,90,91,92
,93,94,95,96,97,98,99,100,101], # Set bins¨
                          align = 'left',
                          rwidth=0.9,
                          color='#bf9786',
                          grid=False,
                          zorder=2)
```

```
cx = municipality_QI.hist(column='alueellinenTäydellisyys',
                          bins=50,
                          rwidth=0.9,
                          align='left',
                          color='#86bfbd',
                          grid=False,zorder=2)
dx = municipality_QI.hist(column='sisällöllinenTäydellisyys',
                          bins=[58,59,60,61,62,63,64,65,66,67,68,69,70,71,72],
                          rwidth=0.9,
                          color='#bfa886',
                          grid=False,zorder=2)

# Additional formatting and parameters for hidtograms
ax = ax[0]
for x in ax:

    # Despine
    x.spines['right'].set_visible(False)
    x.spines['top'].set_visible(False)
    x.spines['left'].set_visible(False)

    # Switch off ticks
    x.tick_params(axis="both", which="both", top=False, left=False, right=False,
labelleft=True)

    # Draw horizontal axis lines
    vals = x.get_yticks()
    for tick in vals:
        x.axhline(y=tick, alpha=0.4, color='#d1d1d1', zorder=1)

    # Show values
    for p in x.containers:
        x.bar_label(p, label_type='edge')

    # Remove title
    x.set_title("")

    # Set x-axis label
    x.set_xlabel("pysäkkitiedon keskiarvoinen ikä vuosina", labelpad=20, size=12)

    # Set y-axis label
    x.set_ylabel("kuntien lukumäärä", labelpad=20, size=12)
bx = bx[0]
for x in bx:

    # Despine
    x.spines['right'].set_visible(False)
    x.spines['top'].set_visible(False)
    x.spines['left'].set_visible(False)
```

```
650
651         # Switch off ticks
652         x.tick_params(axis="both", which="both", top=False, left=False, right=False,
653     labelleft=True)
654
655         # Draw horizontal axis lines
656         vals = x.get_yticks()
657         for tick in vals:
658             x.axhline(y=tick, alpha=0.4, color='#d1d1d1', zorder=1)
659
660         # Show values
661         for p in x.containers:
662             x.bar_label(p, label_type='edge')
663
664         # Remove title
665         x.set_title("")
666
667         # Set x-axis label
668         x.set_xlabel("sijaintitarkkuus (%)", labelpad=20, size=12)
669
670         # Set y-axis label
671         x.set_ylabel("kuntien lukumäärä", labelpad=20, size=12)
672
673     cx = cx[0]
674     for x in cx:
675
676         # Despine
677         x.spines['right'].set_visible(False)
678         x.spines['top'].set_visible(False)
679         x.spines['left'].set_visible(False)
680
681         # Switch off ticks
682         x.tick_params(axis="both", which="both", top=False, left=False, right=False,
683     labelleft=True)
684
685         # Draw horizontal axis lines
686         vals = x.get_yticks()
687         for tick in vals:
688             x.axhline(y=tick, alpha=0.4, color='#d1d1d1', zorder=1)
689
690         # Show values
691         ##x.bar_label(x.containers[0])
692         for p in x.containers:
693             x.bar_label(p, label_type='edge')
694
695         # Remove title
696         x.set_title("")
697
698         # Set x-axis label
```

```
699         x.set_xlabel("alueellinen täydellisyys (%)", labelpad=20, size=12)
700 
701         # Set y-axis label
702         x.set_ylabel("kuntien lukumäärä", labelpad=20, size=12)
703 
704     dx = dx[0]
705     for x in dx:
706 
707         # Despine
708         x.spines['right'].set_visible(False)
709         x.spines['top'].set_visible(False)
710         x.spines['left'].set_visible(False)
711 
712         # Switch off ticks
713         x.tick_params(axis="both", which="both", top=False, left=False, right=False,
714     labelleft=True)
715 
716         # Draw horizontal axis lines
717         vals = x.get_yticks()
718         for tick in vals:
719             x.axhline(y=tick, alpha=0.4, color='#d1d1d1', zorder=1)
720 
721         # Show values
722         for p in x.containers:
723             x.bar_label(p, label_type='edge')
724 
725         # Remove title
726         x.set_title("")
727 
728         # Set x-axis label
729         x.set_xlabel("sisällöllinen täydellisyys (%)", labelpad=20, size=12)
730 
731         # Set y-axis label
732         x.set_ylabel("kuntien lukumäärä", labelpad=20, size=12)
733 
734     # %% [markdown]
735     # MAP VISUALIZATION
736 
737     # %%
738     # Import base map as raster tiles from the National Land Survey of Finland. See
739     README for more information.
740     mml_basemap = TileProvider(
741         name="Maanmittauslaitoksen taustakartta",
742         url="https://avoin-
743     karttakuva.maanmittauslaitos.fi/avoin/wmts/1.0.0/taustakartta/default/WGS84_Pseudo
744     -Mercator/{z}/{y}/{x}.png?api-key=c506db28-9e2f-4415-98f1-e4b4241e86e9",
745         attribution="Maanmittauslaitos CC BY 4.0"
746     )
747 
```

```
# %%
# Create a Folium map object
m = municipality_QI.explore(
    column="ajankohtaisuus",  # Data (column) being visualized
    cmap = "Blues_r", # Colourmap
    scheme="userdefined",  # Set classification scheme
    classification_kwds=dict(bins=[1,2,3,4,5,6,7,8]),
    legend=True,  # Show legend
    tiles=None, # Hide defaul base map tiles
    zoom_start=5, # Set zoom start
    tooltip=False, # Hide tooltip
    popup=["nimi", "ajankohtaisuus"],  # Show popup (on-click)
    legend_kwds=dict(colorbar=False,caption="Ajankohtaisuus (vuosi)",labels=['0-
0,99','1-1,99','2-2,99','3-3,99','4-4,99','5-5,99','6-6,99','7-7,99','8-
8,99']),  # Set colorbar caption and labels
    style_kwds=dict(stroke=True, color='grey',weight=0.5, fillOpacity=0.9), # Add
stroke and add fill opacity
    name="Ajankohtaisuus, kunnat",  # Name of the layer in the map
)

# Create other layers to be passed to the map object
municipality_QI.explore(
    m=m,  # Pass the map object
    column="sijaintitarkkuus",
    scheme="userdefined",
    classification_kwds=dict(bins=[80,85,90,95,99.99,100]),
    legend=True,
    show=False, # Hide layer on start
    cmap = "Blues",
    tooltip=False,
    popup=["nimi", "sijaintitarkkuus"],
    legend_kwds=dict(colorbar=False,caption="Sijaintitarkkuus (%)",labels=['80-
84,99','85-89,99','90-94,99','95-99,99','100,00']),
    style_kwds=dict(stroke=True, color='grey', weight=0.5, fillOpacity=0.9),
    name="Sijaintitarkkuus, kunnat",
)

# Create the rest of the layers as the one above
municipality_QI.explore(
    m=m,
    column="alueellinenTäydellisyys",
    show=False, # Hide layer on start
    scheme="userdefined",
    classification_kwds=dict(bins=[89.99,99.99,100.1,109.99,119.99,130]),
    legend=True,
    cmap = "RdBu_r", # Use diverging colormap for visualizing data completeness
    tiles=None,
    tooltip=False,
    popup=["nimi", "alueellinenTäydellisyys"],
```

```
    legend_kwds=dict(colorbar=False,caption="Alueellinen täydellisyys (%)",
labels=['80-89,99','90-99,99','100,00','100,01-109,99','110-119,99','120-
129,99']),
    style_kwds=dict(stroke=True, color='grey',weight=0.5,fillOpacity=0.9),
    name="Alueellinen täydellisyys, kunnat",
)

municipality_QI.explore(
    m=m,
    column="sisällöllinenTäydellisyys",
    cmap = "Blues",
    scheme="userdefined",
    classification_kwds=dict(bins=[45,50,55,60,65]),
    legend=True,
    show=False, # Hide layer on start
    tiles=None,
    tooltip=False,
    popup=["nimi", "sisällöllinenTäydellisyys"],
    legend_kwds=dict(colorbar=False,caption="Sisällöllinen täydellisyys
(%)",labels=['42-44,99','45-49,99','50-54,99','55-59,99','60-64,99']),
    style_kwds=dict(stroke=True, color='grey',weight=0.5,fillOpacity=0.9),
    name="Sisällöllinen täydellisyys, kunnat",
)

# Add ELY region borders as a layer
ely_QI.explore(
    m=m,
    column="nimi",
    legend=False,  # Hide legend
    fill=False, # Remove fill
    tooltip=False,  # Hide tooltip
    style_kwds=dict(stroke=True, color='black',weight=1.2,fillOpacity=1), # Adjust
stroke to visualize the borders on top of the municipality quality information
    map_kwds=dict(z_index=300), # Set z-index to set the layer to keep in front
    name="ELY Liikenne -alueet",  # Name of the layer in the map
)

# Add base maps to map object
folium.TileLayer(mml_basemap, show=True).add_to(m) # Add NLSF's base map as
default
folium.TileLayer("CartoDB Voyager", show=False).add_to(m) # Alternative built-in
base map by CartoDB

# Add geocoder (search box) to map object
folium.plugins.Geocoder().add_to(m)

# Add layer control to map object
folium.LayerControl().add_to(m)

```

```
# Export map as HTML file
m.save(r'C:\Users\Jenni Autere\Documents\UTU\FM-vaihe\Gradu_hommia\IPY-
kirjastot\kartta.html')

# Show map
m
```