Lajitietojen automaattinen varmistus koneoppimismenetelmillä

Johdanto

Tässä dokumentissa esittelen neljä eri koneoppimismallia poikkeavien lajihavaintojen tunnistamiseen Suomen lajitietokeskuksen datasta. Yksikään menetelmistä ei ole toistaan parempi kaikkiin tilanteisiin.

Lähes kaikki mallit vaativat verrattain paljon opetusaineistoa, joka koostuu pääosin lajitietokeskuksen havainnoista, sekä erilaisista ympäristöä kuvaavista selittävistä / riippumattomista muuttujista, kuten lämpötilasta, CORINE-maankäyttöluokituksista, rantaviivan pituudesta ja niin edespäin. Aineiston käyttö vaihtelee malleittain ja aineistot saa suoraan sellaisinaan kysymällä osoitteesta alpo.turunen@helsinki.fi .

Tulevaisuutta ja mallien parantamista ajatellen uusia aineistoja tai CORINE-aineiston uudelleenluokitusta voi olla hyvä pohtia.

Koodit löytää GitHubista: https://github.com/AlpoTurunen/FinBIF_Outlier_Detection

Contents

Aineistot

Lajitietokeskuksen lajihavainnot (max 100 m tarkkuudella) rajapinnan sensitiivisen datan puolelta: https://api.laji.fi/v0/warehouse/private-query/unit/list?

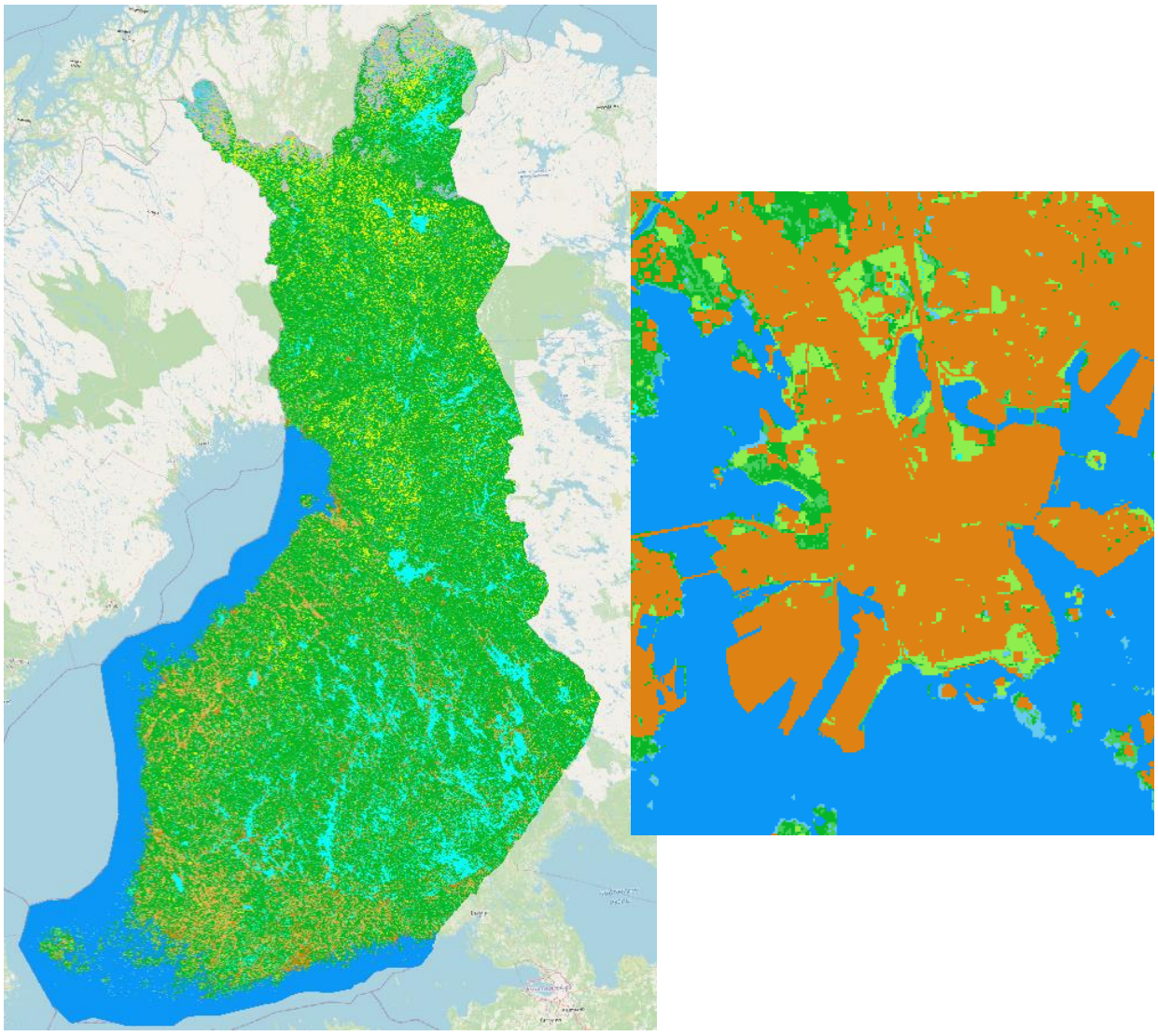
- Python-koodi hakee halutun lajin havainnot. Tarvitset vain access tokenin, ellet sitten käytä osittain karkeistettua dataa julkisen API:n kautta.

Lintuatlaksen havainnot YKJ-ruuduittain esim.: https://atlas-api.2.rahtiapp.fi/api/v1/grid/668:388/atlas

- Tätä varten on Python-skripti get_data_from_birdatlas_api.py
- Tarvitsee 10 km x 10 km YKJ-ruutuja pohjalle, jotka voi ladata shapefilena osoitteesta https://info.laji.fi/etusivu/paikkatieto/paikkatietotuotteet/yhtenaiskoordinaatisto-ykj/

CORINE maanpeite 2018, 25 ha: https://www.syke.fi/fi-FI/Avoin_tieto/Paikkatietoaineistot/Ladattavat_paikkatietoaineistot

- Tämä on uudelleenluokiteltu liitteenä olevan taulukon mukaan.



Kuva 1. CORINE-data

Maanmittauslaitoksen 25 m x 25 m korkeusmalli: https://paituli.csc.fi/download.html

Maanmittauslaitoksen hallintorajat 1:100 000: https://paituli.csc.fi/download.html

- Tästä käytetään vain Suomen ulkorajoja, joten kunnat voi sulauttaa toisiinsa

Ilmatieteenlaitoksen kuukauden keskilämpötilat vuosilta 1961-2023, 10 km x 10 km GeoTIFF: https://paituli.csc.fi/download.html

- Tästä on laskettu kaikkien 2000-luvun kuukausien keskiarvorasteri. Eli keskiarvo keskiarvoista.
- Lisäksi merialueille on interpoloitu lämpötila-arvoja, jottei rasterissa olisi reikiä mm. saariston kohdalla.
- Rasterin resoluutiota voi halutessaan vielä karkeistaa, sillä rasterin arvot ovat liukuvia

Ilmatieteenlaitoksen kuukauden keskisademäärä vuosilta 1961-2023, 10 km x 10 km GeoTIFF: https://paituli.csc.fi/download.html

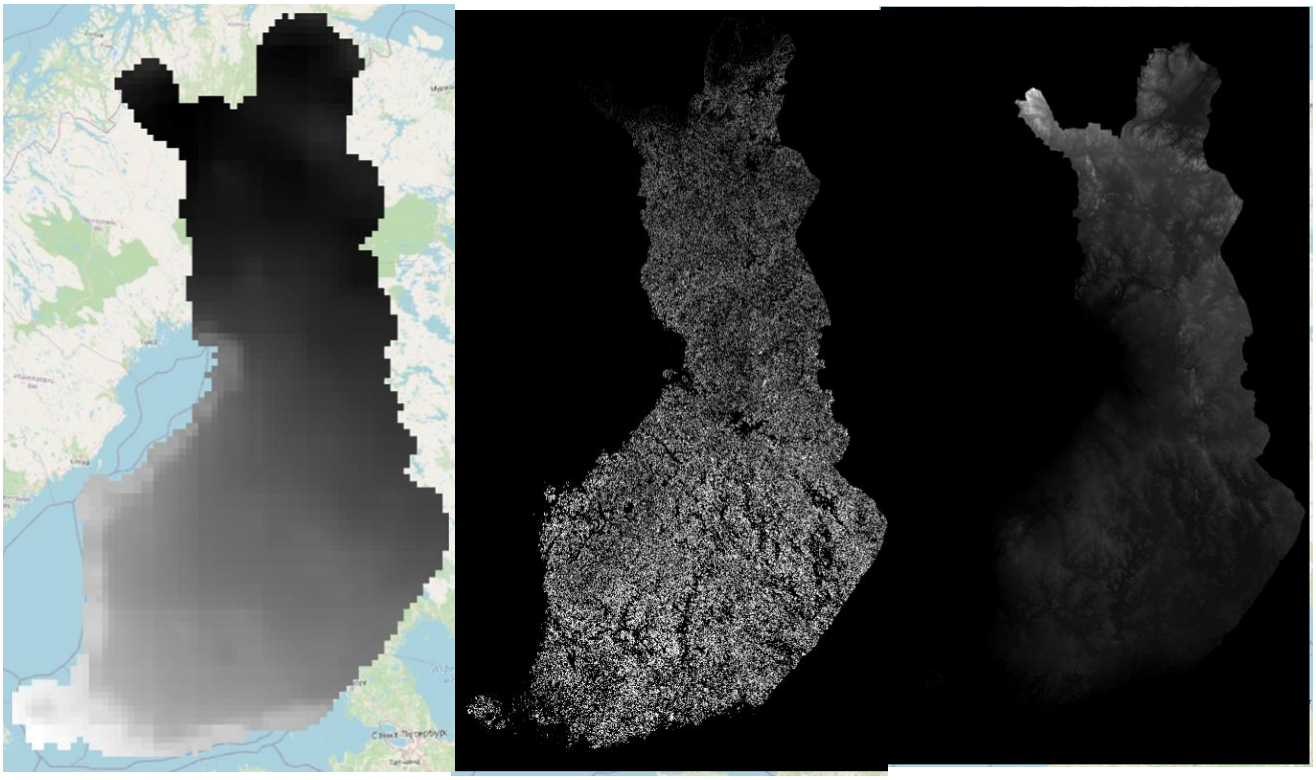
- Tästäkin on laskettu 2000-luvun kuukausien keskiarvorasteri. Eli keskiarvo keskiarvosta
- Lisäksi merialueille on interpoloitu arvoja, jotta rasterin kattavuus olisi parempi
- Rasterin resoluutiota voi halutessaan vielä karkeistaa, sillä rasterin arvot ovat liukuvia

Luonnonvarakeskuksen tilavuus, Puusto yhteensä 2021, (m3/ha), kaikki karttalehdet, GeoTIFF: https://kartta.luke.fi/opendata/valinta.html

- Nämä erilliset rasterit on sulautettu yhdeksi isoksi rasteriksi. Suoritustehoa varten rasterin resoluutiota voi karkeistaa, koska data aika iso

Rantaviivat pituudet jokaisessa YKJ-ruudussa laskettu skriptillä *add_coastline_lengths.py* seuraavista datoista:

- Ranta10 – järvet, Ranta10 – joet, Ranta10 – Meret ja merisaaret osoitteesta https://www.syke.fi/fi-FI/Avoin_tieto/Paikkatietoaineistot/Ladattavat_paikkatietoaineistot



Kuva 2. Lämpötila-, puun tilavuus- ja korkeusmallirasterit

Menetelmät:

Ohjaamattomat (unsupervised) mallit kaikelle datalle

Yleiskuvaus

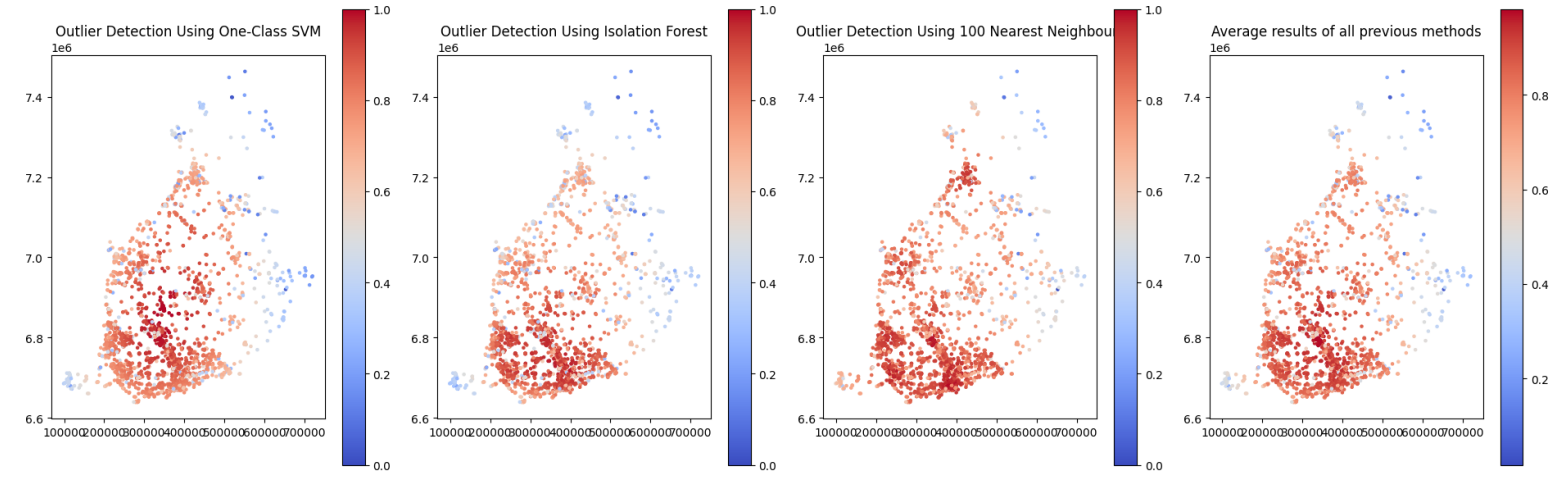
Yksi ensimmäisistä kokeilemistani malleista. Intuitiivinen käyttää kaikille mahdollisille lajeille, eikä tarvitse absence-dataa. Muuttujat voi valita itse. Vaatii erillistä dataa selittävien ympäristömuuttujien valintaan.

Linkki

https://github.com/AlpoTurunen/FinBIF_Outlier_Detection/blob/main/unsupervised_models.ipynb

Workflow

1. Aseta parametrit, kuten bufferien leveys ja muuttujat
2. Koodi lataa data lajitietokeskukselta
3. Koodi poistaa läheiset havainnot halutulta säteeltä
4. Koodi laskee havainnon ympäristön selittävät ympäristömuuttujat datoista annetulla säteellä. Esim. pisteen ympärillä 100 m säteellä merta 5 %, urbaania aluetta 50 %, metsää 15 % jne.
5. Koodi normalisoi arvot välillä 0–1
6. Koodi etsii parhaimmat hyperparametrit jokaiselle mallille. Lopulliseen tulokseen tulee parhaiten arvioitujen parametrien kombinaatiolla ajetut mallit.
    a. One-Class Support Vector Machine
    b. Isolation Forest
    c. K nearest Neighbours
7. Koodi normalisoi jokaisen pisteen todennäköisyydet välille 0–1 ja laskee niiden keskiarvon.
8. Koodi tallentaa tulokset pistemuotoisiksi tasoiksi. Pisteet voi halutessaan interpoloida jatkuvaksi tasoksi *interpolate_results.py* -skriptillä.



Tulokset

Vaikea arvioida, sillä ohjaamattomiin malleihin ei voi soveltaa samoja testejä, kuin ohjattuihin malleille. Tulokset riippuvat paljon valituista parametreista ja datoista taustalla.

Miinukset

- Tuloksien luotettavuutta vaikea arvioida, sillä ei ole vertailudataa samalla tavalla, kuin ohjatuissa malleissa.

- Datan lataamisessa ja rikastamisessa voi kestää kauan erityisesti, jos haluaa pyörittää mallit monelle eri lajille.

- Koska lajihavainnot ovat pääosin tarkkoja, voi usein olla pienestä kiinni, minkä todennäköisyyden malli antaa. Tätä voi toki hiukan korjata buffereiden koolla.

- Tarkoissa lajihavainnoissa on luonnollista vääristymää, joka korostuu tällaisissa tarkan mittakaavan malleissa. Esim. kaupungeissa ja teiden lähettyvillä lajihavainnot yliedustettuina.

- Rasteridatat eivät kata koko maailmaa, joten erityisesti Suomen ulkorajoilla tyhjiä arvoja. Myös Ahvenanmaa helposti outlier, koska on erillään muista.

Plussat

+ Helppo ja intuitiivinen käyttää.

+ Toimii kaikille lajeille

+ Mahdollistaa havainnollistamisen laajemmin ilman erillistä opetusdataa -> ei tarvitse miettiä datan splittauksia

+ Voi saada hyviä tuloksia mallia yksinkertaistamalla, kun käyttää vain muutamia parametreja (x, y ja vuodenaika).

+ Käyttää monia malleja yhtä aikaa, joka vähentää yksittäisten mallien vääristymiä.

+ Muuttujat voi valita lajien mukaan -> vaatii ymmärrystä lajista

Johtopäätös

Miellyttävä käyttää, sillä vähemmän säädettävää kuin ohjatuissa malleissa. Tulokset tosin herkkiä pienille eroille havainnon sijainnissa tai rastereissa.

Ohjattu Random Forest malli kaikelle datalle

Yleiskuvaus

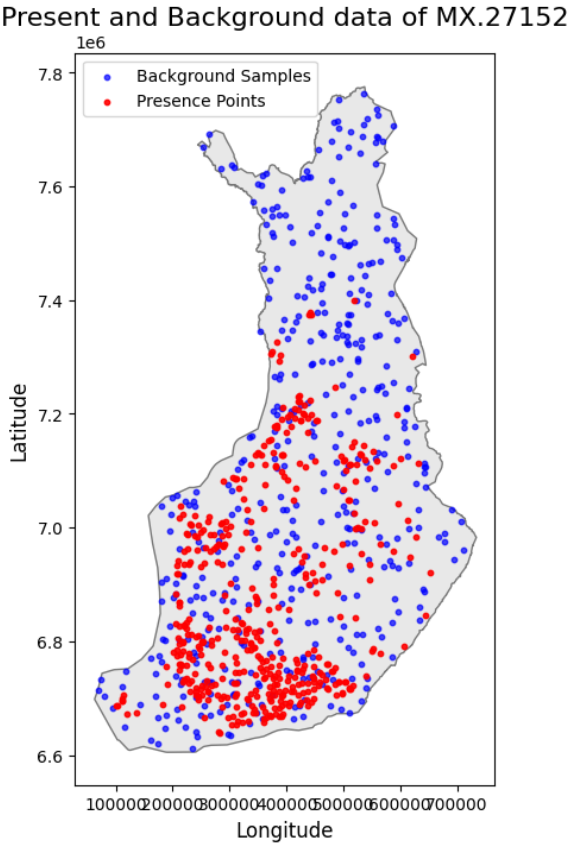
Kirjallisuudessa melko suosittu menetelmä, joka sopii kaikille lajitietokeskuksen havainnoille. Jakaa aineiston koulutus/harjoitusdataan, jolloin vain harjoitusdataa voi arvioida järkevästi. Toki lopputulos on mahdollista interpoloida jatkuvaksi tasoksi.

Linkki

https://github.com/AlpoTurunen/FinBIF_Outlier_Detection/blob/main/random_forest.ipynb

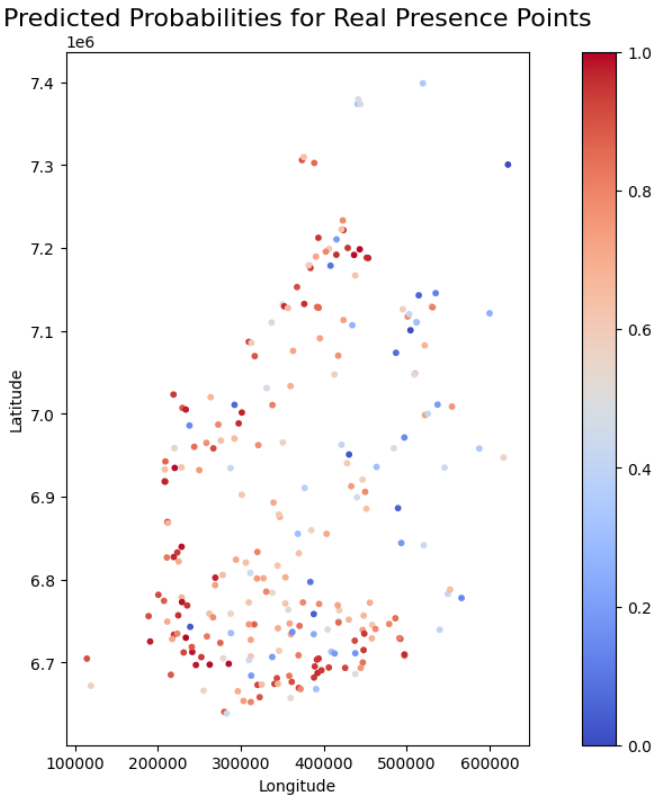
Workflow

1. Aseta parametrit, kuten bufferien leveys ja muuttujat
2. Koodi lataa datan lajitietokeskukselta
3. Koodi poistaa läheiset havainnot halutulta säteeltä
4. Koodi generoi automaattisesti background sample datan, joka jakautuu tasaisesti koko alueelle. Eli nämä siis tavallaan absence dataa, johon oikeiden havaintojen sijaintia verrataan. Esim:



5. Malli laskee havainnon ympäristön selittävät ympäristömuuttujat datoista annetulla säteellä. Esim. pisteen ympärillä 100 m säteellä merta 5 %, urbaania aluetta 50 %, metsää 15 % jne.
6. Koodi jakaa aineiston koulutus ja testausdatoihin shakkilaudan tyyppisellä jaolla. Tällöin spatiaalinen autokorrelaatio (vierekkäisten havaintojen samankaltaisuus) oletettavasti pienempi, kuin satunnaisjaolla.
7. Koodi normalisoi arvot välillä 0–1 ja laskee maantieteelliset painotukset mallille

8. Koodi etsii parhaimmat hyperparametrit Random Forest mallille ja tallentaa parhaimman kombinaation tulokset.
9. Malli tulostaa tilastoja, kuten tarkkuuden (accuracy) ja ROC AUC -arvon. Mitä lähempänä yhtä arvo on, sitä parempi se yleisesti ottaen on. Myös muita tilastoja näytetään.
10. Malli normalisoi tulokset, jolloin jokainen testausdatan piste saa todennäköisyysarvon väliltä 0–1.
11. Malli tallentaa tulokset pistemuotoisiksi tasoiksi. Pisteet voi halutessaan interpoloida jatkuvaksi tasoksi *interpolate_results.py* -skriptillä.



Tulokset

Tätä mallia on suht. helppo arvioida tunnuslukujen avulla. Yleisesti ottaen mallit, joiden ROC AUC tai Accuracy on lähellä yhtä, ovat toimivia. Tällöin malli osaa tunnistaa oikeat havainnot automaattisesti generoiduista havainnoista luotettavasti. Esim. Fasaanille (kuva yllä) tunnusluvut ovat seuraavanlaiset:

Accuracy: 0.84

ROC AUC: 0.94

TSS: 0.69

Miinukset

- Koska lajihavainnot ovat tarkkoja, voi usein olla muutamista metreistä kiinni, minkä todennäköisyyden malli antaa. Tätä voi toki hiukan korjata buffereiden koolla.

- Lajihavaintojen lataaminen ja rikastaminen laskennallisesti hidasta erityisesti, jos haluaa laskea monelle lajille peräkkäin -> koodia voisi toki optimoida

- Tarkoissa lajihavainnoissa on luonnollista vääristymää, joka korostuu tällaisissa tarkan mittakaavan malleissa. Esim. kaupungeissa ja teiden lähettyvillä lajihavainnot yliedustettuina.

- Rasteridatat eivät kata koko maailmaa, joten erityisesti Suomen ulkorajoilla tyhjiä arvoja. Myös ahvenanmaa helposti outlier, koska erillään muista.

- Enemmän säädettävää, kuin ohjaamattomissa malleissa. Esim. taustadatan generoiminen on vähän kyseenalaista ja siihen monta eri tapaa. Sama juttu aineistojen jakamisessa koulutus/testidatoihin. -> tarvitsisi oikeaa absence dataa, jota saatavilla vain muutamasta lajista.

- Osa havainnoista jää aina arvioimatta, sillä ei koulutusdatalla voi testata mallia.

Plussat

+ Toimii periaatteessa kaikille lajeille

+ Mallin hyperparametrit ja muuttujat voi optimoida automaattisesti.

+ Tuloksia helppo arvioida tunnuslukujen (esim. ROC AUC) perusteella.

Johtopäätös

Usein käytetty tutkimuskirjallisuudessa ja varmasti hyvä vaihtoehto, jos on lajikohtaista osaamista ja osaa säätää mallia. Ei kuitenkaan paras yhteisesti kaikille.

Ohjatut mallit lintuatlas datalle

Yleiskuvaus

Tämä on oma henkilökohtainen lempparini, mutta toimii ainoastaan lintuatlaksen linnuille. Tämän mallin käyttäminen tosin edellyttää paikkatieto-osaamista, sillä kaikki data ladataan ja rikastetaan käsin ennen mallien pyörittämistä.

Tämä menetelmä pohjautuu löyhästi Mikko Heikkisen aiempiin mallinnuksiin: https://www.biomi.org/2023/05/23/lintulajien-levinneisyysmallinnus-koneoppimisella/

Taustalla kolme eri mallia:

a. Random Forest Classified
b. Histogram Gradient Boosting Classifier
c. Logistic Regression (Eli käytännössä Maximum Entropy)

Linkki

https://github.com/AlpoTurunen/FinBIF_Outlier_Detection/blob/main/multiple_models_YKJ_squares.ipynb

Data

Tätä mallia varten tarvitsen 10 km x 10 km YKJ-ruudukon, jonka voi ladata ylhäällä olevasta linkistä. Ruudukko kattaa koko suomen ja sen voi täydentää lintuatlaksen pesimävarmuusindekseilel GitHubista löytyvällä skriptillä *get_data_from_birdlatlas_api.py*. Tämän jälkeen tiedostossa pitäisi olla n. 289 000 ruutua, joista suurin osa päällekkäisiä, yksi jokaiselle linnulle.

Eli jokaisen linnun pesimävarmuusindeksi lintuatlaksen rajapinnasta saatavilla 1–3816 YKJ-ruudussa riippuen linnun yleisyydestä ja kun jokaisen linnun ruutujen määrät summataan yhteen, saadaan n. 289 000 ruudun GeoPackage-tiedosto. Jokainen ruutu sisältää linnun nimen, id:n, atlasluokan arvon ja selvitysasteen. Matalat selvitysasteet on jätetty pois.

Tämän jälkeen jokaiselle ruudulle lasketaan arvot selittäville ympäristömuuttujille:

- Ruudun prosentuaaliset osuudet jokaiselle CORINE-maankäyttöluokalle. Voidaan laskea Zonal Statistics -työkalulla esim. QGIS:sä.
- Keskimääräinen lämpötila, maksimipuuntilavuus ja keskimääräinen korkeus.
- YKJ:n pohjois- ja itäkoordinaatit splittaamalla ”koordinaatit” sarake kahteen osaan.
- Rantaviivan pituus ruudussa skriptillä *add_coastline_lenghths.py*.

Ja lisäksi weight-sarakkeeseen laskin painotuksen sen perusteella, kuinka suuri osa ruudusta on Suomen rajojen sisällä. Jos esim. vain 1 % ruudusta on Suomessa, ruudun painoarvo on pieni.

Huom. On huomattavasti järkevämpää laskea nämä arvot vain kerran jokaiselle YKJ-ruudulle ilman päällekkäisyyksiä ja vasta sitten liittää lasketut arvot kaikkiin 289 000 ruutuun.

Lopuksi jokaisen ruudun arvot näyttävät about tältä:

```
coordinates                                                    665:331
species_id                                                    MX.26277
species_name                                                kyhmyjoutsen
atlas_class_value                                        Varma pesintä
Urban                                                             7207
Park                                                             11874
Rural                                                            28108
Forest                                                          114383
Open_forest                                                      21835
Fjell                                                                0
Open_area                                                         1393
Wetland                                                          11593
Open_bog                                                           274
Freshwater                                                        2127
Marine                                                           51206
ykj_n                                                              665
ykj_e                                                              331
temp                                                          0.737937
dem                                                           0.035957
coastline                                                132587.322547
weights                                                          100.0
activity_category                                          Erinomainen
tree_vol                                                    152.394457
geometry            MULTIPOLYGON (((319903.8897000002 6647211.8707...
```

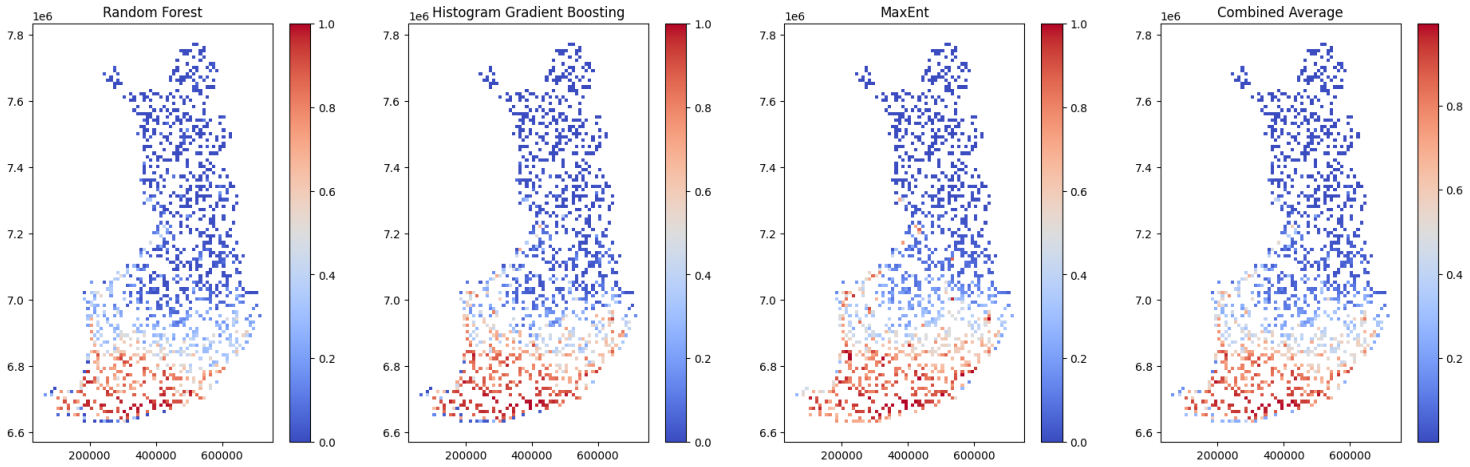
Lisäksi tarvitset samanlaiset 3816 ruutua samanlaisilla selittävillä ympäristömuuttujilla, mutta ilman mitään tietoja linnuista. Tämmöisen saa luotua esim. täydentämällä YKJ-ruutuihin tarvittavat tiedot, tai poistamalla isommasta tiedostosta kaikki duplikaattigeometriat ja tiedot linnuista. Tätä tiedostoa käytetään täydentämään niiden lintulajein ruutuja, joiden havainnot eivät kata kaikki ruutuja ennestään.

Data saatavilla mm. yliopiston verkkolevyltä (*P:\h978\public_admin\Lajitietokeskus\AI\YKJ_birds_and_env_to_models.zip*) tai kysymällä.

Workflow

2. Koodi lataa molemmat tiedostot (289 000 ruutua, joissa tiedot linnuista ja ympäristöstä, sekä 3816 ruutua, joissa tiedot vain ympäristöstä).
3. Koodi ryhmittelee koko datan linnun nimen perusteella ja suorittaa seuraavat askeleet jokaiselle linnulle erikseen
4. Koodi jakaa aineiston koulutus ja testausdatoihin pesimävarmuusindeksin perusteella. Presence-dataa on varma ja todennäköinen pesintä, kun taas absent-dataa on epätodennäköinen ja mahdollinen pesintä, sekä tyhjät arvot.
5. Koodi normalisoi kaikki arvot välillä 0–1.
6. Koodi etsii parhaimmat hyperparametrit seuraaville malleille ja tallentaa parhaimman kombinaation tulokset.
    a. Random Forest Classified
    b. Histogram Gradient Boosting Classifier
    c. Logistic Regression (Eli käytännössä Maximum Entropy)
7. Koodi pyöräyttää datan parhaimmiksi valituille malleille ja tulostaa tilastoja. Tunnusluvut tallentuvat myös CSV-tiedostoon.

8. Tulokset normalisoidaan ja tallennetaan tiedostoon. Koska vain testidatasta saa tuloksia, ne voi interpoloida jatkuvaksi tasoksi *interpolate_results.py* -skriptillä myöhemmin.



Tulokset

Tätä mallia on suht. helppo arvioida tunnuslukujen avulla. Yleisesti ottaen mallit, joiden ROC AUC tai Accuracy on lähellä yhtä, ovat toimivia. Tällöin malli osaa tunnistaa oikeat havainnot automaattisesti generoiduista havainnoista luotettavasti. Esim. kultarinnalle tunnusluvut ovat seuraavanlaiset:

Number of presence data: 663

Number of absent data: 3153

ROC AUC Random Forest: 0.9211225259489838

ROC AUC Histogram Gradient Boosting: 0.9185887152464225

ROC AUC MaxEnt: 0.9193323913436102

Miinukset

- Ennustaa käytännössä lintujen pesimätodennäköisyyttä, ei niinkään lajin havainnon luotettavuutta. Esim. muuttolintuja voi havaita Etelä-Suomessakin, vaikka pesisi vain lapissa. Lisäksi jakoa presence-absence datoihin voisi miettiä lisää. Kumpaan luokkaan esim. ’mahdollinen pesintä’ kuuluisi?

- Monissa lintuatlaksen lajeissa on joko liian vähän havaintoja, tai sitten ne kattavat koko Suomen, jolloin mallin antavat aina vääriä negatiivisia.

- Jos mallit aikoo pyöräyttää kaikille lintuatlaksen lajeille, siihen menee monta tuntia. Toki koodia helppo keventää jättämällä esim. Histogram Gradient Boostingin pois, joka tuottaa välillä outoja tuloksia. Tai sitten vähentämällä kokeiltavien hyperparametrien määrää tai asettamalla parametrin n_splits=5 pienemmäksi StratifiedKFoldissa.

- Voi yleistää liikaa, koska 10 km x 10 km ruudun sisälle mahtuu monenlaista habitaattia. Esim. saaristo leimautuu välttämättä merelliseksi alueeksi, vaikka saaressa olisikin suuri ja mukava metsäalue linnuille, jotka viihtyy metsissä.

Plussat

+ Kiva malli, koska yksittäisten havaintojen tarkkuus ei vaikuta niin paljoa 10 km x 10 km ruuduissa. Lisäksi lintuatlasdata on kerätty järjestelmällisemmin ympäri Suomea.

+ Hyperparametrit voi määrittää automaattisesti ja tuloksia voi helposti verrata tunnuslukujen avulla.

+ Malli pyörii automaattisesti kaikille lintuatlaksen linnuille. Mallin voi toki keskeyttää ja jatkaa myöhemmin haluamastaan linnusta lisäämällä muutaman rivin koodia.

+ Lintuatlaksen data on vähemmän vääristynyttä, kuin moni muu lajitietokeskuksen data. Lisäksi siinä on tavallaan absence-dataa mukana.

Johtopäätös

Yleisesti ottaen tämä malli antaa hyviä tuloksia (ROC AUC > 0.8) lähes kaikille linnuille, jos rajaa liian harvinaiset / yleiset pois. Lintuatlaksen data on lähtökohtaisesti parempaa mallinnuksiin, kuin muu lajitietokeskuksen data, jossa poissaolevia (absence) havaintoja ei juurikaan ole.

Tähän menetelmään voisi harkita Naive Bayesin lisäämistä erityisesti muuttujille, joissa ei ole autokorrelaatiota.

Muut kokeilut

Perhoset, joille luotu käsin sääntöjä

Lisäksi kokeilin Random Forest -mallia lajitietokeskuksen perhosaineistoille, joihin asiantuntijat ovat määritelleet käsin maantieteelliset levinneisyysalueerajat ja sallitut vuodenajat. Tällöin absence-datana käytettiin perhoshavaintoja, jotka eivät olleet sallittujen rajojen mukaisia ja presence-datana havaintoja, joiden maantieteellinen sijainti ja havainnon vuodenaika olivat sallittujen rajojen sisällä.

Menetelmä on muuten samankaltainen, kuin Random Forest -malli ylempänä. Tämän kehittely jäi aika lailla lasten kenkiin, sillä dataa kyseisistä perhosista oli melko vähän, eikä menetelmää voi sellaisenaan soveltaa muihin lajeihin. Toisaalta muut mallit voisi pyöräyttää ko. perhoslajeilla ja pohtia, päteekö tulokset näihin perhosasiantuntijoiden määrittelemiin rajauksiin.

Linkki koodiin:
https://github.com/AlpoTurunen/FinBIF_Outlier_Detection/blob/main/random_forest_with_absent_data.ipynb

Laji	Havlkm
purppurakenttämittari – *Xanthorhoe decoloraria*	1 106
lapinkenttämittari – *Xanthorhoe abrasaria*	520
tuhkapistesiipi – *Pelosia muscerda*	207
palsamikenttämittari – *Xanthorhoe biriviata*	139
metsäkenttämittari – *Xanthorhoe annotinata*	82
juovakenttämittari – *Xanthorhoe quadrifasiata*	16
kaaripistesiipi – *Pelosia obtusa*	4

Samankaltaisten lajien käyttö absence-datana

Kokeilin myös käyttää absence-datana samankaltaisten lajien havaintoja, joita sai skriptillä *get_similar_taxa_from_inat.py* kaiveltua iNaturalistin rajapinnasta. Tällöin esimerkiksi laulujoutsenen levinneisyyttä mallintaessa voisi käyttää kyhmy-, pikku- ja trumpettijoutsenen havaintoja absence-datana.

Tämä voisi sinänsä toimia tilanteessa, jossa tiedetään halutun lajin elävän jotenkin eri tavoin tai erilaisessa elinympäristössä, kuin lajit, joihin se usein sekoitetaan. Joutsenien tapauksessa idea ei ainakaan toiminut, sillä laulu- ja kyhmyjoutsen elää aika lailla samanlaisilla alueilla, eikä trumpetti- ja pikkujoutsenta oikein tavata Suomessa.

Jos jollakin biologilla olisi lajeista tarkempaa substanssiosaamista, tätä voisi pohtia lisää.

Yleisiä haasteita ja parannusideoita

Lajitietojen laatu vaihtelee, eikä monilla lajeilla ole riittävästi havaintoja puhumattakaan nollahavainnoista. Lisäksi lajihavainnoissa on paljon luontaista vääristymää (esim. teiden läheltä enemmän havaintoja, joitain lajeja vaikea tunnistaa, eläimet liikkuvat paljon jne...).

Yleisesti ottaen tarkkoja koordinaatteja on vain murto-osa, eikä esim. CORINE-aineistoa välttämättä kannata hyödyntää, jos sijainti on sinnepäin. Tätä ongelmaa ei ole YKJ-ruuduille tehtävissä mallinnuksissa.

Kaikki mallit löytävät aina vääriä outlier-havaintoja. Usein oikeasti poikkeavia havaintoja on vain muutamia, mutta mallit pyrkivät aina löytämään poikkeuksia myös 100 % oikeiden havaintojen joukosta.

Optimaalisimmassa tapauksessa lajeista olisi tarjolla paljon absence-dataa, johon oikeita, varmistettuja havaintoja voisi verrata. Lintuatlaksen kaltaiselle datalle olisi kysyntää myös muiden lajien kohdalla.

Yhteenveto

	Ohjaamattomat mallit	**Random Forest kaikelle datalle**	**Ohjatut mallit lintuatlas datoille**
Data	Kaikki laji.fi havainnot	Kaikki laji.fi havainnot	Lintuatlaksen pesimävarmuusindeksit YKJ-ruuduittain
Käyttötarkoitus	Antaa lajin kaikille havainnoille todennäköisyyden välille 0–1, jonka perusteella voi löytää poikkeavat havainnot valittujen muuttujien perusteella.	Antaa jokaiselle testidatan havainnolle todennäköisyyden välille 0–1, jonka perusteella voi löytää poikkeavat havainnot valittujen muuttujien perusteella.	Antaa jokaisen lintuatlaksen lajin jokaiselle YKJ-ruudulle todennäköisyyden linnun pesimiselle.
Hyödyt	Ei tarvitse erikseen koulutus- tai absence-dataa. Helppo käyttää halutuille muuttujille.	Toimii tutkimusten mukaan hyvin. Helppo arvioida luotettavuutta tunnuslukujen avulla.	Lintuatlaksen data hyvää dataa. Tuottaa selkeitä lopputuloksia. Helppo arvioida luotettavuutta tunnuslukujen avulla.
Haasteet	Mallien toimivuutta vaikea arvioida ilman vertailudataa. Herkkä parametrien valinnoille. Lajihavaintojen sijaintitarkkuus voi vaikuttaa paljon tuloksiin. Mallien välillä on eroja.	Herkkä parametrien valinnoille. Vaatii absence-datan generointia, sillä oikeaa absence-dataa ei ole saatavilla kuin perhosista.	10 km x 10 km ruutujen sisällä voi olla enemmän vaihtelua, kuin niiden välillä. Vaatii paljon datan esikäsittelyä. Ennustaa pesimätodennäköisyyttä, ei suoraan lajin havaittavuuden epäluotettavuutta. Mallien välillä on eroja.

Liite 1: CORINE-uudelleenluokittelu:

Uusi luokka	Vanhat luokat
URBAN 1	1 kaupunki, tiheä 2 kaupunki, asuinalue 3 kaupunki, julkiset rakennukset 4 teollisuus 5 tie, rautatie 6 hiekkakenttä, parkkipaikka, satama 7 lentokenttä 8 kaupunki, teollisuus? hiekkakuoppa 9 (rakennustyömaa) 10 kaatopaikka, maankaatopaikka 11 rakennustyömaa
PARK (2)	12 puistoniitty, puustoinen puisto 13 kosteikko (nyynäinen), mökkikylä (puolarmaari) 14 urheilukenttä 15 golfkenttä 21 rantaniitty, ruderaatti, puutarha
RURAL (3)	16 (hedelmäpuut) 17 pelto, voi olla myös rantaniitty (yyteri) 18 mansikkapelto, hedelmäpuut 19 (sekalainen pelto) 20 (sekalainen pelto)
FOREST (4)	22 (puuistutukset? agro-forest) 23 metsä, lehti?, ml. tunturikoivikko 24 (metsä, havu) 25 metsä, seka/havu? 26 (luonnollinen niitty) 27 (nummi) 28 metsä, havu? 29 (metsä-pusikko vaihettumisvyöhyke)
OPEN FOREST (5)	30 (dyynit) 31 (avokallio) 33 metsä (palanut alue) 34 metsä, avohakkuu (jäätikkö) 35 (inland marsh) 36 (peat bog) 37 (salt marsh)
FJELL (6)	32 tunturipaljakka
OPEN AREA (7)	38 hiekkaranta (saline) 39 avokallio, ml. tunturilaet (intertidal flat) 44 turvetuotanto
WETLAND (8)	40 tunturipaljakka, kostea? 41 kosteikko 42 kosteikko
OPEN BOG (9)	43 avosuo (torronsuo, isosuo...) (estuary)
WETLAND B	45 kosteikko 46 ruovikko
FRESHWATER (10)	47 joki 48 järvi
MARINE (11)	49 meri