Tartu Ülikool

Loodus- ja täppisteaduste valdkond

Matemaatika ja statistika instituut

Mats Ploompuu

Mitmemõõtmelised meetodid puuliikide osakaalude prognoosimiseks satelliidiandmete põhjal

Matemaatika ja statistika õppekava

Matemaatilise statistika eriala

Magistritöö (30 EAP)

Juhendaja Märt Möls

Tartu 2019

Mitmemõõtmelised meetodid puuliikide osakaalude prognoosimiseks satelliidiandmete põhjal

Töös on rakendatud mitmeid mitmemõõtmelisi meetodeid Eesti metsade liigilise koosseisu prognoosimiseks satelliidiandmete põhjal. Parimad tulemused saadakse *K*-lähima naabri meetodit kasutades. Täpsemalt sobitatakse igale satelliidipildile eraldi *K*-lähima naabri mudel ning prognoositakse puuliikide osakaalud. Seejärel saadud prognoosid agregeeritakse. Töös on näidatud, et selliste prognooside agregeerimiseks on paremaid mooduseid kui aritmeetiline keskmine, näiteks Epanechnikovi tuumameetodiga hinnatud tiheduse mood.

Parima mitmemõõtmelise meetodi puuliikide osakaalude prognooside põhjal on koostatud näidiskaart.

Märksõnad: matemaatiline statistika, andmeteadus, tehisõpe, ruumiline statistiline analüüs, mitmemõõtmeline analüüs, metsakooslused

Teadusala kood ja nimetus: P160 statistika, operatsioonanalüüs, programmeerimine, finants- ja kindlustusmatemaatika

Multivariate methods for predicting tree species composition using satellite data

In this paper several multivariate methods are used for predicting tree species composition of Estonian forests employing satellite data. The best results are obtained by using *K*-Nearest Neighbor algorithm. More precisely independent KNN models are fitted to every single satellite image to predict tree species composition and afterwards those predictions are aggregated. In the paper better approaches compared to arithmetic mean to aggregate such predictions are demonstrated, such as Epanechnikov kernel density estimation's mode.

A sample map of tree species composition is constructed for a selected area using the predictions obtained by the best multivariate model.

Keywords: mathematical statistics, data science, machine learning, spatial statistics, multivariate analysis, forest communities

CERCS classification and code: P160 statistics, operation research, programming, actuarial mathematics.

Sisukord

1. Sissejuhatus

Kirjeldades Eesti metsi, räägitakse kaasikutest ja kuusikutest, haavikutest ja männikutest, metsastatistikas kasutatakse metsa iseloomustamiseks ka terminit peapuuliik. Selline kirjeldus on liiga ühekülgne, hõlmates vaid ühte puuliiki, mitte aga kooslust tervikuna. Metsakooslus on enamasti oma loomult keerukama, mitmemõõtmelise struktuuriga, ilma kindla klassikuuluvuseta.

Käesolevas töös määratakse metsa liigiline koosseis tervikuna: ei prognoosita mitte peamist puuliiki, vaid kooslusse kuuluvate puuliikide osakaalud. Prognoosid ja nende taga olevad meetodid on seega mitmemõõtmelised.

Liigilise koosluse prognoosimiseks kasutatakse satelliidiandmeid. Satelliidiandmed leiavad üha laiemat kasutust kaugseire valdkonnas, olles ühtlasi aina paremini kättesaadavad. Nii on ka töös kasutatavad satelliidiandmed kõigile vabalt kättesaadavad. Metsade kaugseire, nagu nimigi ütleb, võimaldab metsa uurida distantsilt. Satelliidipildid katavad kogu Eesti ja nii saab paljud mahukad välitööd asendada distantsilt kogutud andmete peale ehitatud mudelitega ning mudelite abil prognoosida seda, mida varem on pidanud käsitsi tegema – muuhulgas metsa liigilist koosseisu.

Töö eesmärgiks on satelliidiandmete põhjal mitmemõõtmeliste meetoditega prognoosida Eesti metsade liigilist koosseisu. Seejuures vaadatakse läbi erinevad satelliidiandmetele lähenemise võimalused, leitakse kõigi töös kasutatud meetodite korral täpseima prognoosini viivad mudelid ja nende parameetrid ning pakutakse välja võimalused teatud tüüpi prognooside agregeerimiseks. Mudeleid võrreldakse omavahel nende prognooside täpsuse alusel ning parimaks osutuva mitmemõõtmelise mudeli prognooside põhjal koostatakse puuliikide osakaalude kaart.

2. Ülevaade andmetest

Töös uuritavateks objektideks on alad, kus metsa hindaja ehk taksaator on statistilise metsainventuuri käigus kohal käinud ja teostanud mõõtmised. Neil 7–10 meetrise raadiusega aladel ehk takseeraladel on määratud puude liigid, kahjustused jms, mõõdetud on puude läbimõõt, kõrgus jms [1]. Töös huvi pakkuvaks ehk prognoositavaks tunnuseks on puuliikide osakaalude vektor. Puuliigi osakaalu all on mõeldud antud puuliigi tüvemahu ja kõigi takseeralal kasvavate puude tüvemahtude suhet. Takseerandmetes on kuue puuliigi osakaalud: mänd, kuusk, kask, haab, must lepp ja hall lepp, lisaks neile muude puuliikide osakaal. Töös on kitsendatud puuliikide arv kolmele: mänd, kuusk ja kask ning neile lisaks muud puuliigid.

Peamise osa seletavatest tunnustest moodustavad satelliidiandmed. Satelliidiandmed on pärid NASA ja USA Geoloogiateenistuse satelliidilt Landsat-8 ja Euroopa Kosmoseagentuuri Sentinel-2 satelliitidelt. Satelliidid mõõdavad erinevatele lainepikkustele vastavate spektraalkanalite heledusi, töös kasutatud kanalid on toodud tabelis 1. Edaspidi on ühele konkreetsele kuupäevale ja satelliidile vastavat andmehulka nimetatud ka satelliidipildiks.

Tabel 1. Sentineli ja Landsati kanalid [2].

Kanali nr. Sentinel	Kanali nr. Landsat			Lainepikkus (µm)	Resolutsioon (m) Sentinel	Resolutsioon (m) Landsat
2	2	*Blue*	Sinine	0.493 – 0.535	10	30
3	3	*Green*	Roheline	0.537 – 0.583	10	30
4	4	*Red*	Punane	0.646 – 0.685	10	30
5	-	*Vegetation Red Edge*	Vegetatsioonipunase serv	0.694 – 0.714	20	-
6	-	*Vegetation Red Edge*	Vegetatsioonipunase serv	0.731 – 0.749	20	-
7	-	*Vegetation Red Edge*	Vegetatsioonipunase serv	0.768 – 0.796	20	-
8a	5	*Near Infrared*	Lähisinfrapunane	0.848 – 0.881	20	30
11	6	*Shortwave Infrared*	Lühilaine infrapunane	1.539 – 0.1681	20	30
12	7	*Shortwave Infrared*	Lühilaine infrapunane	2.072 – 2.312	20	30

Sentinel-2 satelliite on kaks, kuid nende erinevused on väga väikesed. Tabelis 1 toodud lainepikkused on Setninel-2A ja Sentinel-2B keskmine. Satelliidiandmed on pärit

ajavahemikust 12.06.2015 kuni 19.09.2018, seejuures Sentineli satelliitide pilte on 15 ning Landsati pilte 9.

Takseeralad ja satelliidipildid on seotud takseerala keskpunkti koordinaatide põhjal. Paraku tuleb takseeralade ja satelliidipiltide iseärasustest tingituna umbes pooled alad kõrvale jätta. Takseeralasid maakategooriaga „mets“ on 935. Töös on kasutatud neid vaatlusi, kus tüvemahuhinnang on vähemalt 100 m^3 hektari kohta. Lisaks tüvemahupiirangule on kõik vaatlused läbinud manuaalse ülevaatuse Maa-ameti kaardil [3], ehk on kontrollitud

1. Takseerinfo vastavust tegelikkusele, peamiselt ega pärast takseerimist pole teostatud raieid.
2. Takseerala ja tema lähiümbruse homogeensust.

Esimese kontrolli puhul on paratamatu, et Maa-ameti pildid ei pruugi olla piisavalt ajakohased ja tegelikkuses on teostatud takseerimise ja satelliidipiltide tegemise vahel metsaraie, mida Maa-ameti kaart ei kajasta. Teine kontroll on küllaltki subjektiivne. Homogeensuse all on mõeldud võimalike häiringute puudumist takseerala läheduses. Häiringud on näiteks põld, maantee, raiesmik, veekogu vms. Sõltuvalt sellest, mitmel takseerala küljel asub midagi muud kui mets, on välja jäetud alad, kus häiringud on keskpunktile lähemalt kui 10–20 meetrit. Esiteks võib takseerala keskpunkti koordinaat olla ebatäpne – koordinaadid on üldjuhul teda täpsusega vähemalt 20 m [4] – teiseks on satelliidipiltide piksli suurus sõltuvalt kanalist 10–30 meetrise küljepikkusega. Seega võib heterogeensete piirkondade puhul 7–10 meetrise raadiusega takseerala keskpunktiga seotud piksel näidata mitte takseeralalt mõõdetud vastava kanali heledust, vaid näiteks kõrvalasuva ühtlaselt kollase õitsva rapsipõllu oma.

Pärast ebasobilike alade kõrvalejätmist jääb töös kasutada 455 takseerala.

Satelliidiandmetele on lähenetud kahel erineval viisil. Esimene lähenemine on kasutada töötlemata andmeid, st kasutada iga pildi informatsiooni eraldi. Teine lähenemine on luua koondandmestik, kus iga takseerala kohta on Sentineli kanalite väärtused kevade alguses, kevade lõpus ja sügisel. Koondandmestiku loomisel on kasutatud esmajärjekorras Sentineli andmeid, kuna Sentinelil on rohkem kanaleid kui Landsatil ning ka rohkem pilte: 15 vs 9 pilti. Lisaks Sentineli andmetele on iga Landsati pildi korral Landsati andmete põhjal prognoositud Sentineli kanalitega kattuvate kanalite väärtused (neid kanaleid on 6, tabel 1) ja kasutatud neid koondandmestiku loomisel. Selleks on iga kattuva kanali jaoks koostatud lineaarne mudel, kus pikslil võib olla juhuslik mõju, kuna Landsati ja Sentineli pikslite kattuvus maapinnal võib

pikslite lõikes olla erinev. Seejärel prognoositakse Sentineli kanalite väärtused kevade alguses, kevade lõpus ja sügisel, kasutades selleks kõiki Sentineli pilte ning Landsati piltide põhjal tehtud Sentineli kanalite prognoose. Selline lähenemine võimaldab kasutada informatsiooni fenoloogiliste erinevuste kohta, mis esimese lähenemise korral puudub.

Lisaks satelliidipiltidele on takseeralade kohta teada mullatüüp. Mullainfo on rühmitatud masinõppe jaoks sobilikuks [5] ning teisendatud seejärel 1–0 tunnusteks.

Kui ei ole teisiti öeldud, siis kõik töös kasutatavad andmed on projekti „KARU“ raames edastanud Tartu Ülikooli Tartu observatooriumi metsade kaugseire vanemteadur Mait Lang.

3. Puuliikide osakaalude prognoosimiseks kasutatavad meetodid

Puuliikide osakaalude prognoosimiseks kasutatakse töös mitmemõõtmelisi meetodeid ehk meetodeid, mis võimaldavad prognoosida osakaalude vektori tervikuna. See seab ka mõistliku piirangu kasutust leidvate meetodite arvule. Töös kasutatavad meetodid on:

- *K*-lähima naabri meetod (*K-nearest neighbors algorithm*)
- Regressioonipuu (*Regression tree*)
- Juhumets (*Random forest*)
- Multinomiaalne logistiline regressioon (*Multinomial logistic regression*)

K-lähima naabri meetod, edaspidi ka KNN, on metsade liigilise koosseisu prognoosimise valdkonnas üks levinumaid [6] [7] [8]. Seetõttu leitakse sel meetodil võrdluseks osakaalu vektori tervikuna prognoosimisele ka eraldi prognoosid puuliikide kaupa ning kombineeritakse neist prognoosidest puuliikide osakaalude vektori prognoos.

Takseeralade vähesus ei võimalda neid jagada treening- ja testandmeteks. Seetõttu on kõikide veahinnangud leidmisel kasutatud „jäta-üks-välja“ ristvalideerimist (*Leave-one-out cross-validation*).

3.1 K-lähima naabri meetod

Valitud positiivse täisarvu *K* ja ennustatava punkti x_0 korral tuvastab *K*-lähima naabri meetod esmalt *K* vaatlust treeningandmetes, mis on kõige lähemal punktile x_0. Olgu need vaatlused tähistatud N_0. Seejärel hinnatakse $f(x_0)$ kui kõikide hulgas N_0 olevate treeningvaatluste uuritava tunnuse *y* keskmine. Ehk

$$\hat{f}(x_0) = \frac{1}{K}\sum_{x_i \in N_0} y_i \text{ [9].} \qquad (1)$$

Antud töös ei ole andmestik jagatud treening- ja testandmeteks, seetõttu ei saa rääkida *K* vaatlusest treeningandmetes, vaid iga konkreetse vaatluse x_0 korral *K* lähimast vaatlusest kõigist ülejäänud vaatluste hulgast *X*, $x_0 \notin X$.

Lähimate punktide määramisel on võimalik kasutada erinevaid kaugusmõõte. Levinud on erinevate parameetri väärtusega Minkowski kauguse kasutamine [10]. Minkowski kaugus parameetriga *p* punktide x_i ja x_j tunnuste arvu *m* korral:

$$d_{Minkowski}(i,j) = \left(\sum_{h=1}^{m} \left|x_h^i - x_h^j\right|^p\right)^{1/p} \text{[10].} \quad (2)$$

Antud töö piirneb eukleidilise kauguse kasutamisega, ehk Minkowski kaugus parameetriga $p = 2$.

Kuna KNN baseerub intuitiivsel eeldusel, et distantsilt lähedased vaatlused on potentsiaalselt sarnased, siis on loogiline omavahel eristada *K* lähimat naabrit, teisisõnu lähematel punktidel *K* lähima naabri seas olgu vaadeldava punkti väärtuste prognoosimisel suurem mõju. Selle saavutamiseks tuleb igale lähimale naabrile anda kaal sõltuvalt tema suhtelisest kaugusest vaadeldavast punktist. [11]

Lähimate naabrite kaugused tuleb mingi eeskirja alusel teisendada kaaludeks. Olgu *u* kauguste vektor ning *w* kaalude vektor. Levinud funktsioon *i*-nda naabri kaalu leidmiseks on:

$$w_i = \frac{\frac{1}{u_i}}{\sum_{i=1}^{K} \frac{1}{u_i}} \text{[6].} \quad (3)$$

Veel on välja pakutud näiteks

$$w_i = \frac{exp(-u_i)}{\sum_{i=1}^{K} exp(-u_i)} \text{[11]} \quad (4)$$

või erinevad tuumafunktsioonid [10], näiteks Epanechnikovi või *tricube* tuumafunktsiooni korral vastavalt

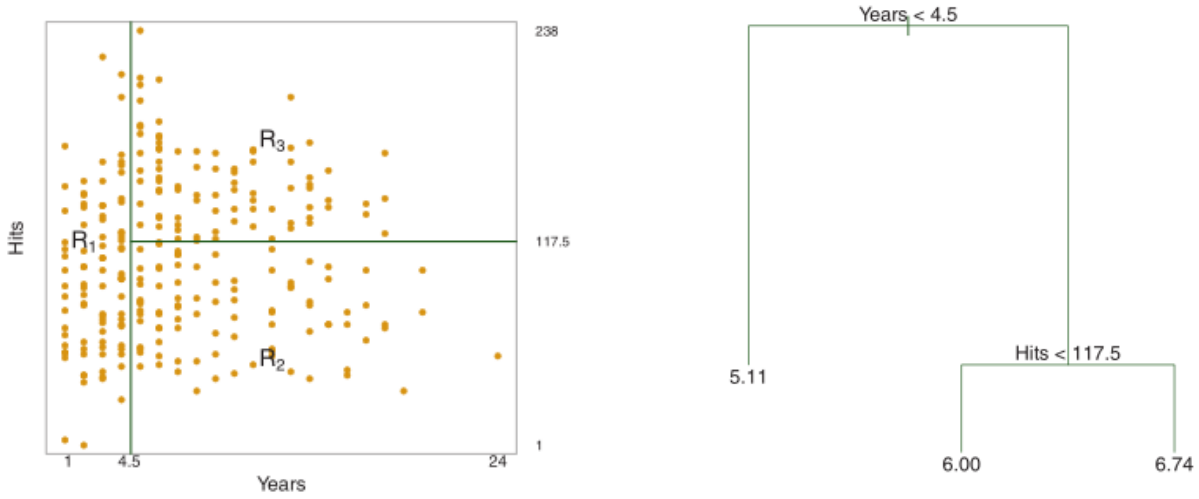
$$w_i = \frac{3}{4}(1 - u_i^2) \text{ ja} \quad (5)$$

$$w_i = \frac{70}{81}(1 - |u_i|^3)^3, \quad (6)$$

kus on nõutud, et maksimaalne lähima naabri kaugus on 1 [10]. Loogiline teisendus on jagada Eukleidiliste kauguste vektor u maksimaalse kaugusega: $u' = \frac{u}{max(u)}$. Lisaks tuleb arvesse võtta, et sellisel juhul suurima kaugusega naabri kaal on 0: selle vältimiseks tuleb K naabri kaalude leidmiseks kasutada K+1 naabri kauguste vektorit.

3.2 Regressioonipuu

Regressioonipuu korral jagatakse seletavate ehk sõltumatute tunnuste ruum $X_1, X_2, \dots, X_p$ esmalt J erinevaks lõikumatuks piirkonnaks $R_1, R_2, \dots, R_j$. Seejärel saab iga piirkonda R_j langev vaatlus sama prognoosi, milleks on lihtsalt antud piirkonda langenud vaatluste uuritavate tunnuste keskmine. [9]



Joonis 1. Näide kahe seletava tunnuse ruumi jagamine kolmeks lõikumatuks piirkonnaks R_1, R_2, R_3 *ning sama jagamine puu kujul.* [9]

Teoreetiliselt võiksid regioonid $R_1, R_2, \dots, R_j$ olla mistahes kujuga, kuid mudeli lihtsuse ja interpreteeritavuse huvides jagatakse sõltumatute tunnuste ruum kõrge dimensionaalsusega ristkülikuteks ehk kastikesteks. Eesmärgiks on leida sellised kastikesed, et prognooside jääkide ruutude summa (RSS) oleks minimaalne:

$$RSS = \sum_{j=1}^{J} \sum_{i \in R_j} \left(y_i - \hat{y}_{R_j}\right)^2, \qquad (7)$$

kus $\hat{y}_{R_j}$ on *i*-ndasse kastikesse jäävate vaatluste uuritavate tunnuste keskmine. Kahjuks ei ole arvutuslikult mõistlik kaaluda kõikvõimalikke tunnuste ruumi jaotusi J kastikeseks. Seetõttu kasutatakse lähenemist, mis on „ülevalt alla“ ja samas „ahne“, mis on tuntud ka kui rekursiivne binaarne poolitamine (*recursive binary splitting*). Lähenemine on „ülevalt alla“, kuna ta algab puu ülevalt osast ning seejärel järjest poolitab tunnuste ruumi. [9] Joonisel 1 paremal on kahe hargnemisega puu, kusjuures puu „tüvi“ asub ülaosas ja hargnemine toimub alla poole – vastupidiselt tavapärasele arusaamale puust. Regioone $R_1, R_2, \dots, R_j$ kutsutakse seejuures puu „lehtedeks“. Lähenemine on „ahne“, kuna iga puu konstrueerimise sammu korral tehakse parim poolitus just selle konkreetse sammu põhjal, mitte valides poolituse, mis eelseisvaid samme silmas pidades võiks kokkuvõttes viia parema puuni. [9]

Rekursiivse binaarse poolitamise puhul valitakse esmalt tunnus X_j ja selline lõikepunkt s, et tunnuste ruumi jagamise korral regioonideks $\{X|X_j < s\}$ ja $\{X|X_j \geq s\}$ väheneks RSS (valem 7) kõige enam. Teisisõnu vaadatakse läbi kõik tunnused $X_1, X_2, \dots, X_p$ ja kõikvõimalikud lõikepunkti s väärtused iga tunnuse korral ning valitakse selline tunnus ja lõikepunkt, et RSS oleks minimaalne. [9]

Protsessi korratakse nüüd juba mitte terve tunnuste ruumi, vaid eelmisel sammul leitud regioonide peal: vaadatakse läbi kõik tunnused X_j ja kõikvõimalikud lõikepunktid s kõikides eelmistel sammudel loodud regioonides. Protsess lõpetatakse mingi kriteeriumi alusel, näiteks kuni pole enam ühtegi regiooni, kuhu kuuluks enam kui viis vaatlust. [9]

Kokkuvõttes on ennustatava punkti x_0 prognoos

$$\hat{f}(x_0) = \sum_{j=1}^{J} \hat{y}_{R_j} \cdot 1_{(x_0 \in R_j)}. \qquad (8)$$

3.3 Regressioonipuu ja bagging

Bagging ehk *bootstrap* agregeerimine on üldist laadi protseduur, mis vähendab prognooside hajuvust [9]. *Bootstrap* valim on tagasipanekuga juhuvalim andmestikust, kusjuures valim on

sama suur kui algne andmestik. Seega on mõned vaatlused valimis esindatud korduvalt, mõned on aga välja jäänud. [12]

Bagging meetodi korral on punkti x_0 prognoosiks *bootstrap* valimite põhjal arvutatud prognooside $\hat{f}^b(x_0)$ keskmine, kus *B* on valimite arv:

$$\hat{f}_{bag}(x_0) = \frac{1}{B}\sum_{b=1}^{B} \hat{f}^b(x_0). \text{ [9]} \tag{9}$$

3.4 Juhumets

Juhumetsa meetod on väga sarnane meetodile, kus on ühendatud regressioonipuu ja *bagging.* Nagu ka eelmise meetodi puhul, siis juhumetsa korral luuakse teatud hulgale *bootstrap* valimitele regressioonipuud, ent neid puid ehitades ei otsida parimat poolitust mitte kõigi tunnuste seast, vaid iga poolituse korral otsitakse parimat tunnust *m* tunnuse seast, mis on juhuslikult valitud *p* kõigi tunnuse seast. [9] Selle kohta, milline peaks olema *m* ja *p* omavaheline suhe, on erinevaid arvamusi: levinud on $m \approx \sqrt{p}$ [9] ja $m \approx p / 3$, mida on soovitatud ka meetodi looja Breiman meetodit tutvustavas artiklis [13].

3.5 Multinomiaalne logistiline regressioon

Multinomiaalne logistiline regressioon on logistilise regressioon mitmemõõtmeline juht.

Olgu π mingi sündmuse toimumise või klassi kuulumise tõenäosus. Logistilise regressiooni korral sobitatakse lineaarne mudel sündmuse toimumise šansi logaritmile:

$$\log\left(\frac{\pi}{1-\pi}\right) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p. \tag{10}$$

Logistilise regressiooni mudeli korral jääb sündmuse toimumise tõenäosus alati 0 ja 1 vahele, mida lineaarse regressiooni korral ei ole võimalik tagada. [12]

Mudelist 10 saab avaldada tõenäosuse π*:*

$$\pi = \frac{\exp(\beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p)}{1 + \exp(\beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p)}. \quad (11)$$

Mitmemõõtmelisel juhul on sündmusi või klasse enam kui kaks. Olgu uuritava tunnuse *Y* klasside arv *c* ning klassi kuulumise tõenäosus $\pi_j,\ j = 1, \ldots, c,\ \sum \pi_j = 1$. Multinomiaalse logistilise regressiooni mudeli loomiseks fikseeritakse mingi baaskategooria, olgu selleks *c*, ning sarnaselt valemile 10 sobitatakse mudel šansside paaride logaritmidele:

$$\log\left(\frac{\pi_j}{\pi_c}\right) = \beta_{j0} + \beta_{j1} x_1 + \ \ldots + \beta_{jp} x_p, j = 1, \ldots, c - 1. \quad (12)$$

Seega on mudeli kohta $c - 1$ eraldi parameetritega võrrandit, mille parameetrid leitakse tarkvaraliselt samaaegselt. [14]

Tõenäosus π_j avaldub mudelist 12 seega järgnevalt:

$$\pi_j = \frac{\exp(\beta_{j0} + \beta_{j1} x_1 + \cdots + \beta_{jp} x_p)}{\sum_{h=1}^{c} \exp(\beta_{h0} + \beta_{h1} x_1 + \cdots + \beta_{hp} x_p)}, j = 1, \ldots, c. \text{ [14]} \quad (13)$$

4. Tulemused

Tulemused on *K*-lähima naabri meetodi ja multinomiaalse logistilise regressiooni korral arvutatud R-i keskkonnas ning regressiooni ja juhumetsa meetodite korral on kasutatud *Python*-it. Mudelite võrdluse aluseks on prognooside ruutkeskmine viga (*root mean square error* - RMSE), mille leidmisel on kasutatud jäta-üks-välja ristvalideerimist.

4.1 K-lähima naabri meetod

K-lähima naabri meetodi rakendamisel on kasutatud kahte erinevat lähemist takseerandmetes antud puuliikide tüvemahtude osakaaludele:

1. Kasutada *K*-lähima naabri puuliikide **osakaalude** vektorit ning prognoosida nende pealt huvipakkuva vaatluse puuliikide osakaalu vektor.
2. Kasutada *K*-lähima naabri puuliikide **tüvemahtude** vektorit. Esmalt prognoosida huvipakkuva vaatluse tüvemahtude vektor ning seejärel võtta osakaalude prognoosiks normeeritud tüvemahtude vektor.

Sisuliselt võtab 2. lähenemine arvesse takseeralade tüvemahtusid, andes suurema tüvemahuga aladele suurema kaalu. Takseerinfos on puuliikide tüvemahud alati sama suhtega, mis puuliikide osakaalud.

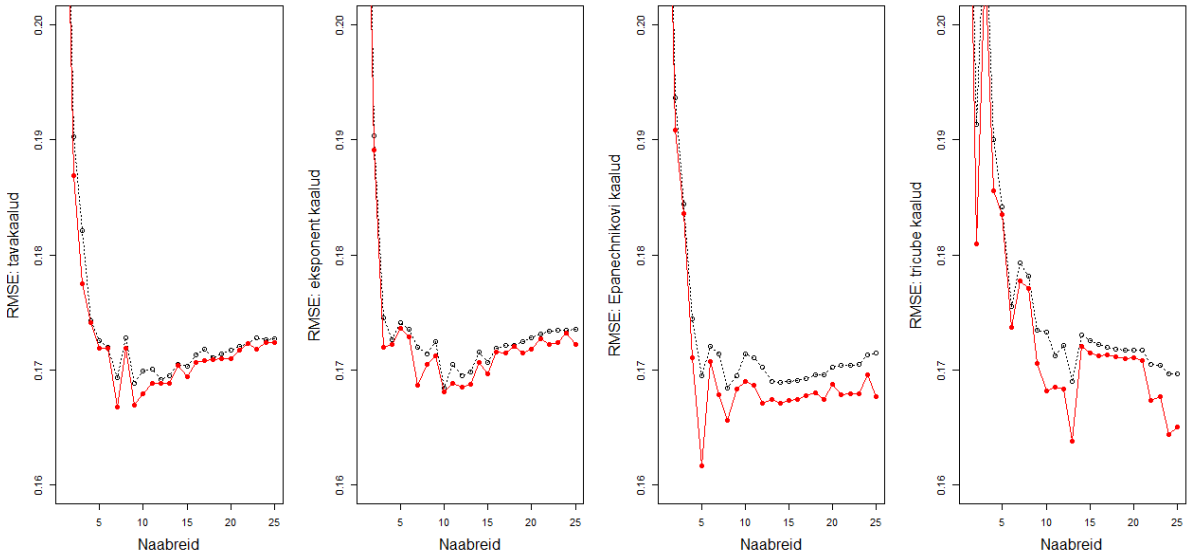
Mõlemal lähenemise korral on kasutatud nelja erinevat varianti naabrite kaalude leidmisel lähtuvalt nende eukleidilisest kaugusest:

1. Nn tavakaalud, ehk pöördvõrdeline eukleidilise kaugusega (valem 3)
2. Eksponentkaalud, ehk eksponent negatiivse märgiga eukleidilisest kaugusest (valem 4)
3. Epanechnikovi tuuma kaalud (valem 5)
4. *Tricube* tuuma kaalud (valem 6)

Iga naabrite arvu *K* korral leitakse esmalt tunnus, mille põhjal prognoosimine annab parima tulemuse ehk vähima RMSE. Seejärel lisatakse kõigi ülejäänud tunnuste seast tunnus, mille lisamisel RMSE kõige enam väheneb. Protsessi jätkatakse, kuni järgmise tunnuse lisamisel RMSE enam ei parane.

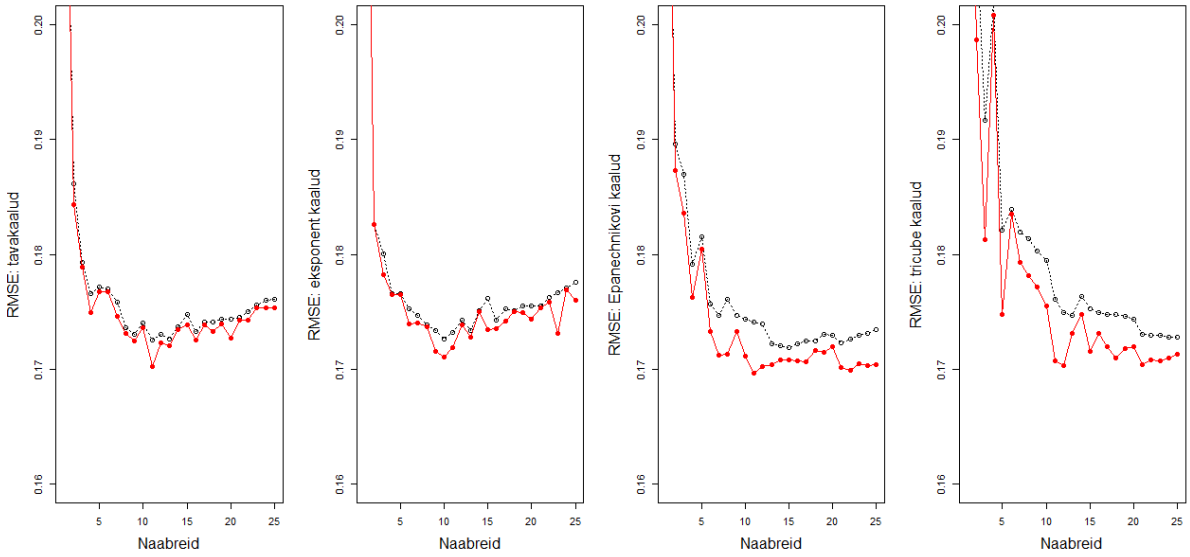
Seejärel optimeeritakse valituks osutunud tunnuste kaalud. Kuna KNN meetodi korral kasutatakse standardiseeritud andmestikku, siis kaalumine toimub läbi prognoosi parandavate

tunnuste korrutamise kaalude vektoriga. Kaalud on optimeeritud R-i funktsiooniga *optim* kasutades kvaasi-Newtoni meetodit, mis on küll aeglasem kui Nelder-Meadi meetod, kuid andis üldiselt parema tulemuse. Joonisel 2 on kujutatud RMSE käitumist sõltuvalt naabrite arvust erinevate kauguse kaalumismeetodite korral, kui prognoosid on arvutatud puuliikide tüvemahtude põhjal.



Joonis 2. Puuliikide osakaalude vektorite põhjal hinnatud mudelite ruutkeskmine viga sõltuvalt naabrite arvust. Kasutatud on nelja erinevat kauguse kaalumise meetodit (vasakult): 1. nn tavakaalud, eksponentkaalud, Epanechnikovi kaalud ja tricube kaalud. Must katkendjoon tähistab tulemust enne tunnuste kaalude optimeerimist, punane joon pärast optimeerimist.

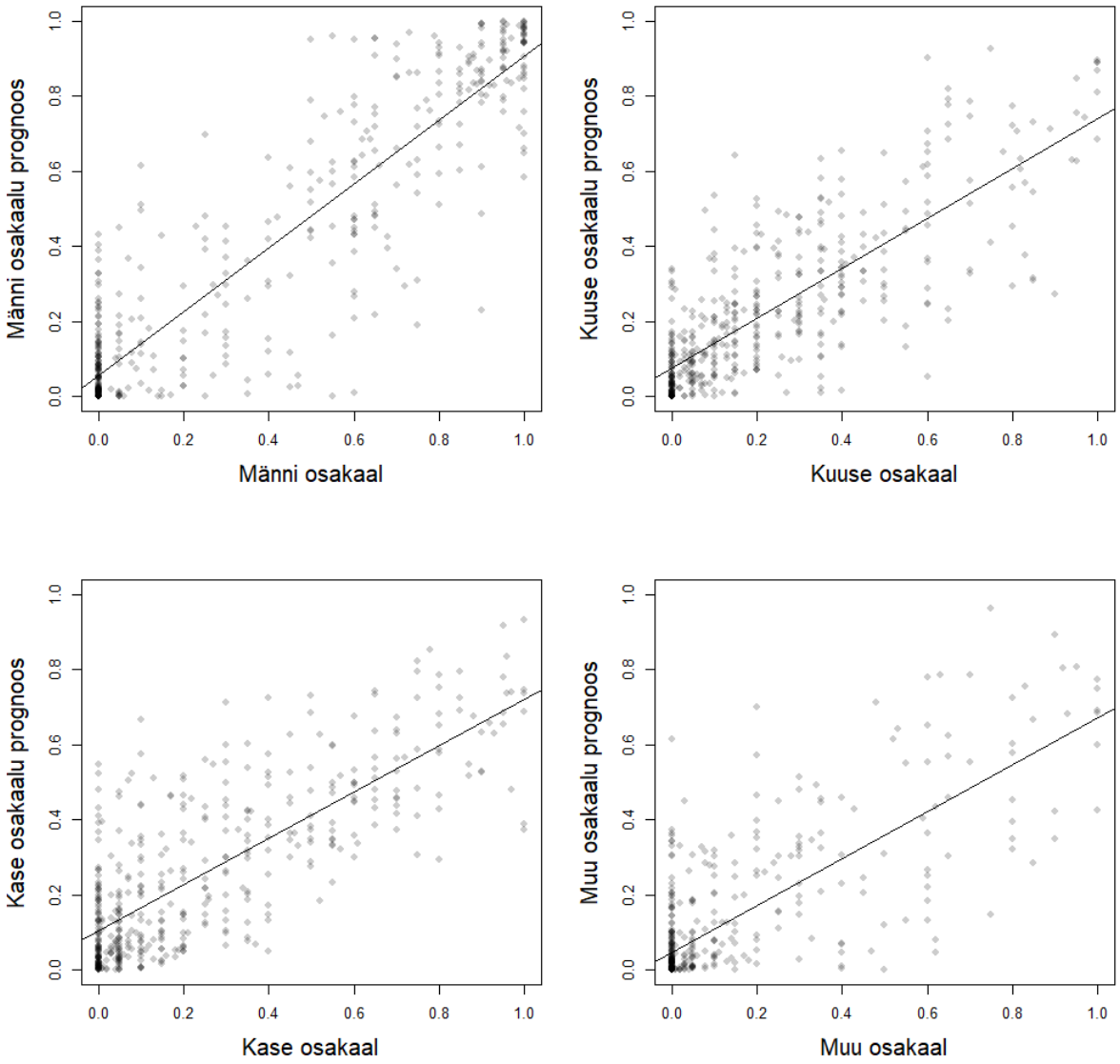
Tavakaalud ja eksponentkaalud käituvad üpris sarnaselt. Pärast tunnuste kaalude optimeerimist saavutatakse parim RMSE vastavat 7 (RMSE = 0.1668) ja 10 lähima naabri korral (RMSE = 0.1681). Epanechnikovi ja *tricube* kaalud töötavad paremini: Epanechnikovi kaale kasutades saavutatakse parim RMSE pärast tunnuste kaalude optimeerimist 5 lähima naabri korral (RMSE = 0.1616), kusjuures enne tunnuste kaalude optimeerimist annab parima tulemuse 8 naabri kasutamine. *Tricube* kaalude korral annab parima tulemuse 13 naabri kasutamine (RMSE = 0.1638). Satelliidipiltide põhjal KNN meetodiga metsanduslike näitajate prognoosimisel on soovitatud kasutada väikseimat K väärtust, mille korral RMSE ei ületa mingi etteantud protsendi võrra parimat tulemust [8] [15]. Antud juhul on selge, et valituks osutub Epanechnikovi kaaludega kaugus ja naabrite arv $K = 5$.



Joonis 3. Puuliikide tüvemahtude vektorite põhjal hinnatud mudelite ruutkeskmine viga sõltuvalt naabrite arvust. Kasutatud on nelja erinevat kauguse kaalumise meetodit (vasakult): 1. nn tavakaalud, eksponentkaalud, Epanechnikovi kaalud ja tricube kaalud. Must katkendjoon tähistab tulemust enne tunnuste kaalude optimeerimist, punane joon pärast optimeerimist.

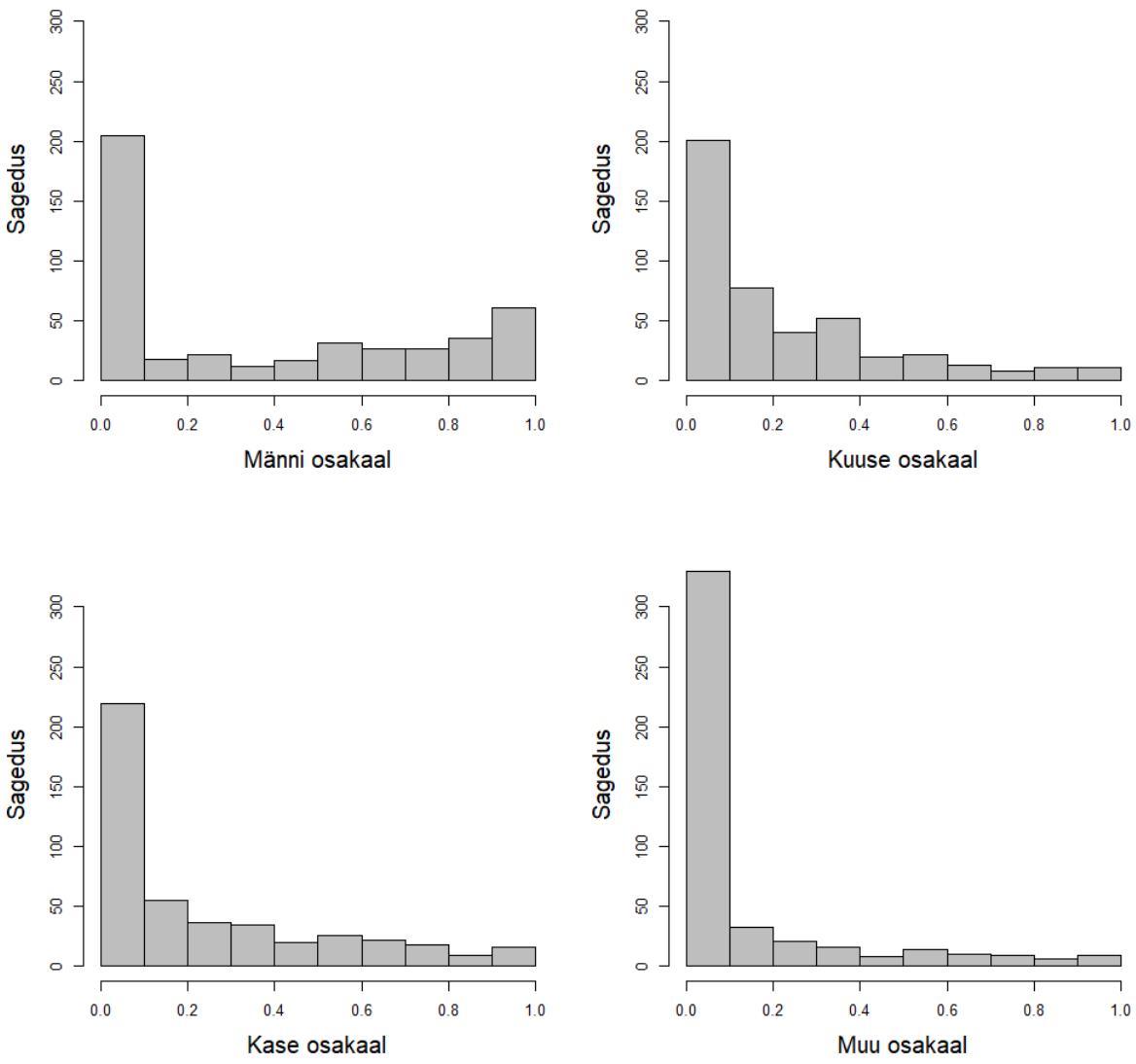
Puuliikide tüvemahtude põhjal prognoosides (joonis 3) käituvad erinevad kaalud üpris sarnaselt puuliikide osakaalude põhjal prognoosimisele, kuid kõikide kaalumismeetodite korral on tulemus kehvem. Seetõttu on töös edaspidi keskendutud puuliikide osakaalude prognoosimisele osakaalude põhjal.

Jooniselt 4 on näha, et kõikide liikide osakaalude prognoosimisel peale männi tekib suuremate osakaalude prognoosimisel süstemaatiline viga. Mänd on puuliikidest ainuke, mille puhul on arvestatav hulk vaatlusi, kus männi osakaal on 1 või selle lähedal (joonis 5), mis on arvatavasti hoiab ära suurema süstemaatilise vea. Süstemaatilisele veale ekstreemsete väärtuste (antud juhul 0 ja 1) prognoosimisel on ka varasemates töödes tähelepanu juhitud: kõik väikseimad väärtused on ülehinnatud ja kõik suuri väärtused alahinnatud, kuna paratamatult väikseima väärtusega vaatluse kõik naabrid on sama suured või suuremad ja vastupidi: suurima väärtusega vaatluse naabrid on sama suured või väiksemad [8].



Joonis 4. Puuliikide osakaalude prognoosid KNN meetodiga, K = 5. Epanechnikovi kaalud. RMSE = 0.1616.

Süstemaatiline viga ei tähenda, et ruutkeskmine viga oleks teistel liikidel kehvem kui männil. RMSE liigikaupa: mänd 0.1611. kuusk 0.1541, kask 0.1767, muu 0.1536, ehk kõige täpsemini on prognoositud hoopis muude liikide osakaalu. Põhjuseks võib olla „muu“ puuliigi kõige suurem nullide sagedus (joonis 5) – väärtus, mille täpse prognoosimisega saab meetod kõige edukamalt hakkama.



Joonis 5. Puuliikide osakaalude sagedused

Tabelis 2 on ära toodud parimas mudelis (K = 5, Epanechnikovi kaalud, hinnatud puuliikide osakaalude pealt) oluliseks osutunud tunnused ja nende kaalud pärast optimeerimist.

Tabel 2. KNN, K = 5 mudeli tunnused ja nende kaalud pärast optimeerimist.

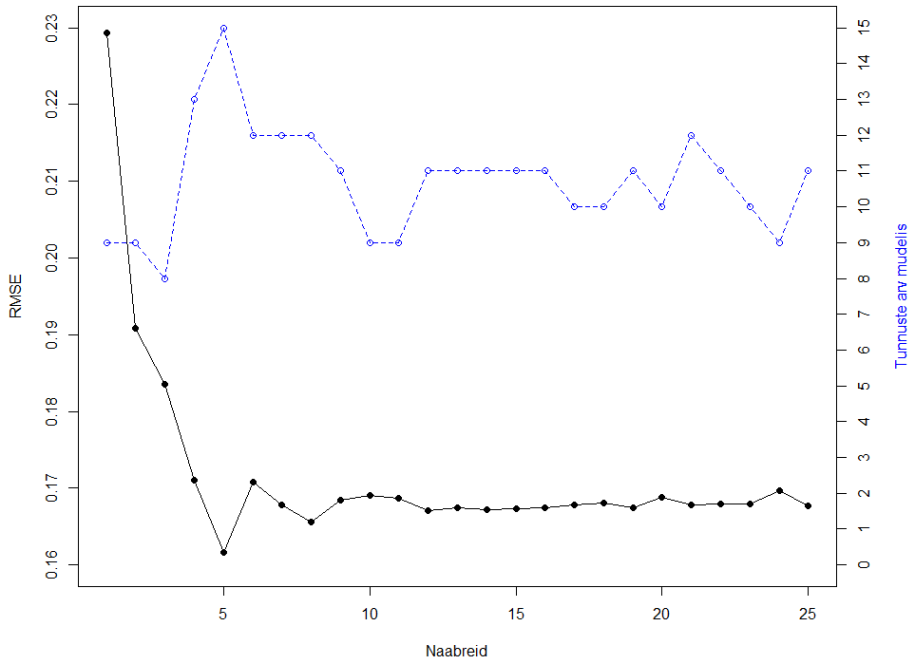
B08 kevadine erinevus	1.22	B11 kevadine erinevus	0.92	B04 sügis	1.30
B08 kevade lõpp	1.34	Muld 77	1.06	Muld 53	0.73
Muld 37	1.02	B04 kevade lõpp	1.24	B07 kevade algus	0.90
Muld 21	1.09	B11 sügis	0.90	B04 kevadine erinevus	0.82
B07 sügis	0.83	Muld 57	0.97	Muld 51	0.67

Satelliidikanalitest on esindatud B04 – punane, B07 – vegetatsioonipunase serv, B08 – lähisinfrapunane, B11 – lühilaine infrapunane. Satelliiditunnuseid on kokku 9, kuid peale erinevate punaste teisi satelliidikanaleid oluliste tunnuste seas ei ole.

„Muld 37“, „muld 77“ jne on masinõppe jaoks rühmitatud mullatüüpide indikaatortunnused. Rühmitatud mullatüüpe on mudelis 6.

Tunnuste kaalude optimeerimise alglähend on kõigi kaalude korral olnud 1. Kõige enam on üles kaalutud kevade lõpu lähisinfrapunane ning sügise punane, kõige väiksemate kaaludega on kaks mullatunnust. Kaalud jäid mingil määral sõltuma alglähendist.

Jooniselt 6 tuleb välja, et parimat prognoositäpsust näidanud viie lähima naabri mudelil on ühtlasi ka suurima tunnuste arvuga analoogiliste KNN mudelite seas.



Joonis 6. Tunnuste arv (sinine katkendjoon) Epanechnikovi kaaludega KNN mudelis erinevate naabrite arvu K korral. Võrdluseks RMSE samade naabrite arvude korral (must joon, pärast tunnuste kaalude optimeerimist).

4.2 K-lähima naabri meetod, pilt-haaval prognoosimine

Töös kasutatavas andmestikus on kokku 24 satelliidipilti. Neist 15 on pildistatud Sentinel-2 poolt ning 9 on pärit satelliidilt Landsat-8. Võrreldes koondandmestikuga on tunnused natukene erinevad: nimelt ei saa kasutada fenoloogiat ehk kanalite aastaajalisi erinevusi. Erinevad on ka Sentineli ja Landsati kanalid: Sentinel mõõdab kolme vegetatsioonipunase lainepikkust, mis Landsatil puuduvad.

Igale pildile on sobitatud eraldi KNN mudel: naabrite arv ja tunnused võivad piltide lõikes olla erinevad. Naabrite kaalumise meetod on kõikide piltide korral sama: eelmises peatükis parimaks osutunud Epanechnikovi kaalud.

Sentineli 15-st pildist 4 katavad ära kõik andmestikus olevat 455 takseerala, aga selles arvestuses kõige kehvem vaid 86 ala. Kõige rohkem on ühe takseerala kohta pilte 14, selliseid alasid on 41. 117 ala kohta on 13 pilti, 170 ala kohta 12 pilti jne. Kõige kehvemal juhul on aga ühe ala kohta vaid 8 pilti (tabel 3)

Tabel 3. Sentineli pilte takseerala kohta.

Pilte takseerala kohta	14	13	12	11	10	9	8
Takseeralasid	41	117	170	96	20	10	1

Landsati pilte on 9, neist parim katab ära 435 takseerala, kõige kehvem aga vaid 60. Kaks takseerala on kõigil piltidel, 36 ala kohta on 8 pilti jne. Kolme ala kohta on vaid 3 pilti (tabel 4).

Tabel 4. Landsati pilte takseerala kohta.

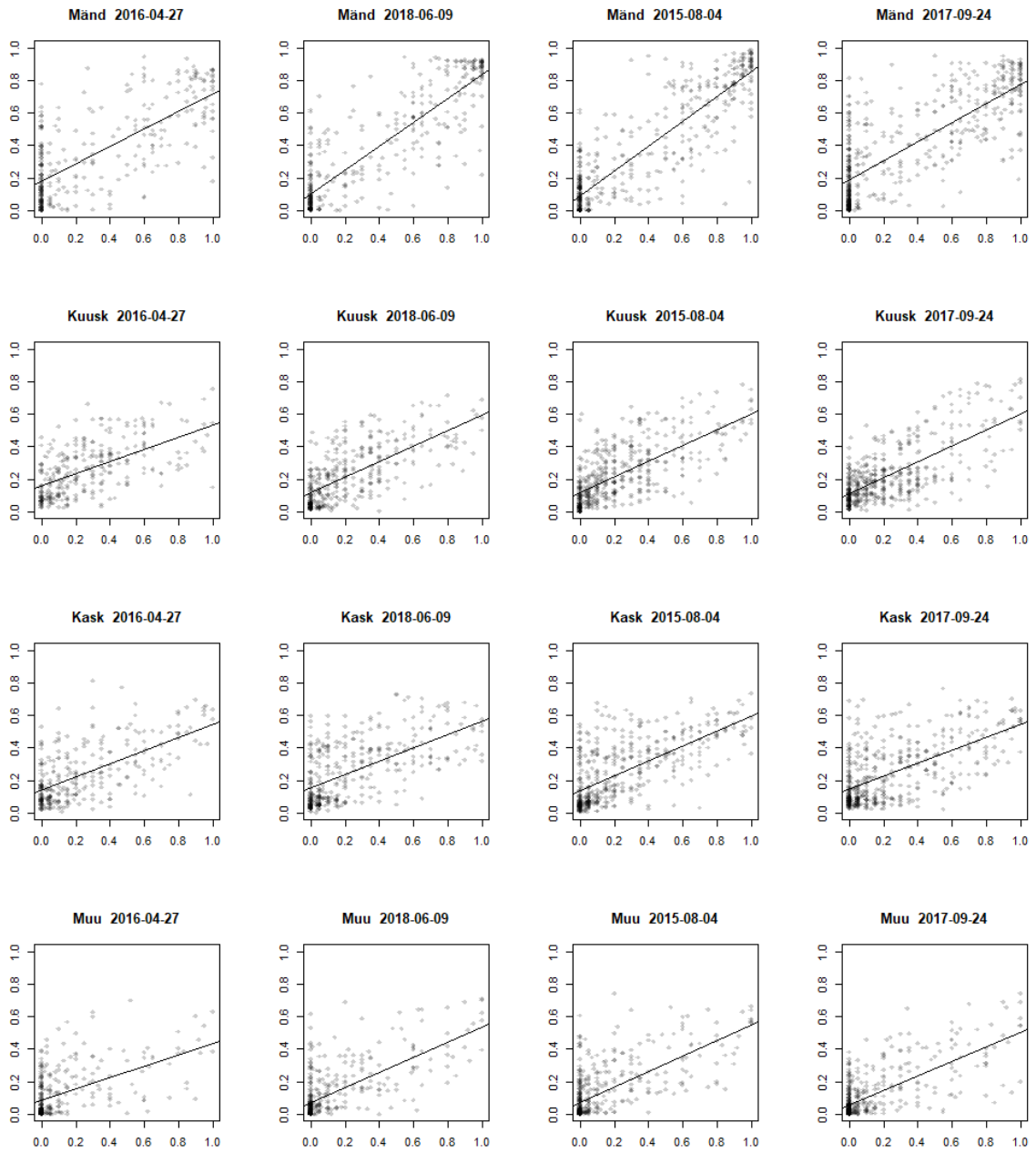
Pilte takseerala kohta	9	8	7	6	5	4	3
Takseeralasid	2	36	65	99	148	102	3

Kui vaadata Sentineli ja Landsati pilte ühiselt, siis parimal juhul on kahe takseerala kohta 22 pilti maksimaalsest võimalikust 24-st. Kõige kehvematel juhtudel on takseerala kohta 13 pilti (tabel 5).

Tabel 5: Sentineli ja Landsati pilte takseerala kohta.

Pilte takseerala kohta	22	21	20	19	18	17	16	15	14	13
Takseeralasid	2	22	41	68	92	113	67	39	6	5

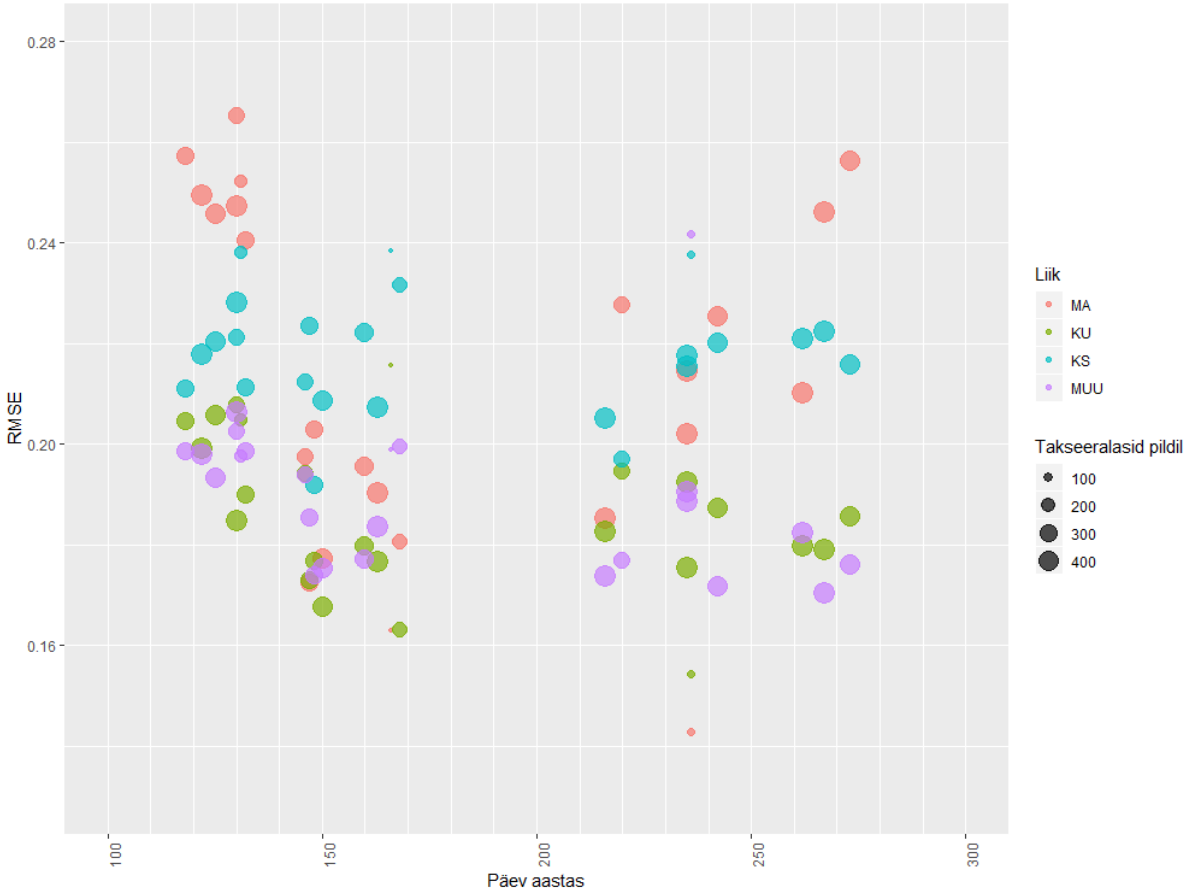
Joonisel 7 on ülevaade prognoosidest erinevatel kuupäevadel. Valitud on üks pilt varakevadel (27. aprill), üks pilt suve alguses (9. juuni), üks pilt suve lõpu poole (4. august) ja üks pilt sügisel (24. september). Naabrite arv nende piltide korral on vastavalt 10, 18, 17, 18, (20 on maksimaalne arv naabreid, mida on proovitud). Visuaalselt käituvad männi prognoosid kõige paremini, eriti suviste piltide korral. Kevadised ja sügisesed prognoosid on natukene kehvemad: võib eeldada, et näiteks varakevade ei ole lehtpuud veel piisavalt lehes, et neid omavahel eristada ja teiseks, et nad okaspuu-lehtpuu segapuistutes üldse visuaalselt nähtaval oleksid.



Joonis 7. KNN: pilt-haaval prognoosid erinevate puuliikide osakaaludele valitud kuupäevadel. Naabrite arv K vastavalt kuupäevadele 10, 18, 17 ja 18.

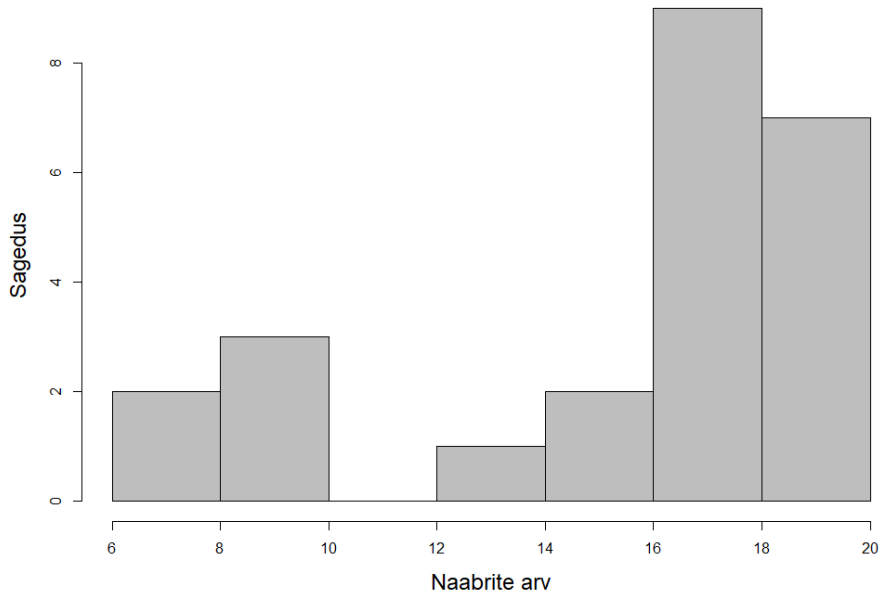
Erinevate puuliikide prognooside täpsuse sõltuvalt aastaajast on toodud joonisel 8. Väga teravalt jäävad silma mai alguse (~120. päev aastas) männi prognooside kehvem täpsus võrreldes kevade lõpu ja suvega. Männi puhul jääb silma ka sügiseste (~270. päev aastas) prognooside ebatäpsus. Mai alguses on ka kõikide teiste liikide prognooside täpsus kehvem kui

ülejäänud kuupäevadel, kuid erinevus ei ole nii terav kui männil. Ülejäänud kuupäevadel on teiste liikide prognooside täpsus üpris sarnane. Prognooside täpsuse sõltumist vastaval kuupäeval tehtud pildil olevate takseeralade arvust ei tule visuaalselt esile.



Joonis 8. KNN: pilt-haaval prognooside RMSE liikide kaupa sõltuvalt kuupäevast (mitmes päev aasta algusest).

Kui võtta pilt-haaval prognooside aritmeetiline keskmine, nõudes eelnevalt, et iga pildi korral oleks kasutatud sama arvu naabreid, siis kahe lähima naabri $K = 2$ korral RMSE = 0.1655. Erinevatel piltidel erineva arvu naabrite lubamine annab praktiliselt olematu võidu, seetõttu on edaspidi kasutatud kõikide piltide korral sama naabrite arvu. Jooniselt 9 on näha, et üksikute piltide korral on minimaalne naabrite arv 6 ning enamasti koguni 16–20, seega parima piltide aritmeetilise keskmise nimel tuleb kasutada kardinaalselt vähem naabreid kui üksikute piltide korral oleks optimaalne.



Joonis 9. KNN naabrite arv üksikute piltide korral.

Kindlasti ei ole aga aritmeetiline keskmine ainus võimalus pilt-haaval prognoose üheks terviklikuks prognoosiks agregeerida.

4.2.1 Pilt-haaval prognooside agregeerimine

Iga takseerala kohta on 13 satelliidipilti, kõige sagedamini 17–18 pilti. Seega on iga takseerala puuliikide osakaalude vektori kohta vähemalt 13 prognoosi, mis tuleb agregeerida üheks vektoriks. Töös on rakendatud sel eesmärgil 3 erinevat meetodit:

1. Aritmeetiline keskmine
2. Beta-jaotuse tiheduse maksimumpunkt ehk mood
3. Epanechnikovi tuumafunktsiooni abil leitud tiheduse mood

Aritmeetiline keskmine on lihtsasti (suurte andmemahtude korral kiiresti) arvutatav, olles baasiks, millega teiste meetoditega saadud tulemusi võrrelda.

Teine töös rakendatud meetod on kasutada agregeeriva funktsioonina pilt-haaval prognooside pealt hinnatud parameetritega beta-jaotuse tiheduse moodi. Beta-jaotus tuleb kasuks 0 ja 1 vahel asuvate pidevate juhuslike suuruste modelleerimisel, nagu näiteks osakaalud või protsendid [16], seega võiks sobida üksikute puuliikide osakaalude modelleerimiseks. Beta-

jaotust võibki mõista kui tõenäosuste jaotust, ehk ta representeerib vaadeldava tõenäosuse kõiki võimalikke väärtusi, kui me ei tea, mis see tõenäosus tegelikult on [17]. Beta-jaotuse tiheduse parameetrite hindamiseks on kasutatud funktsiooni *ebeta* R-i paketist *EnvStats*. Beta-jaotuse parameetrite hindamiseks on nõutud, et vaatluste väärtused jääksid vahemikku 0 kuni 1. Vältimaks 0 ja 1 esinemist andmetes, tuleb pilt-haaval saadud prognoosid teisendada: $y' = [y(N-1)+0.5]/N$ [18], kus *N* on vaatluste koguarv.

Antud töös on andmetele rakendatud ka Beta-jaotuse mitmemõõtmelist üldistust Dirichlet' jaotust, ent Beta-jaotus on töötanud käesoleva töö andmete korral paremini ja seega pole Dirichlet' jaotusega põhjalikult edasi mindud.

Tuumameetodi abil leitud tihedusfunktsiooni hinnanguks (*Kernel density estimator*) mingi tuumafunktsiooni *K* ja positiivse akna laiuse *h* korral kutsutakse funktsiooni

$$\hat{f}(x) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{h}K\left(\frac{x-X_i}{h}\right), \tag{14}$$

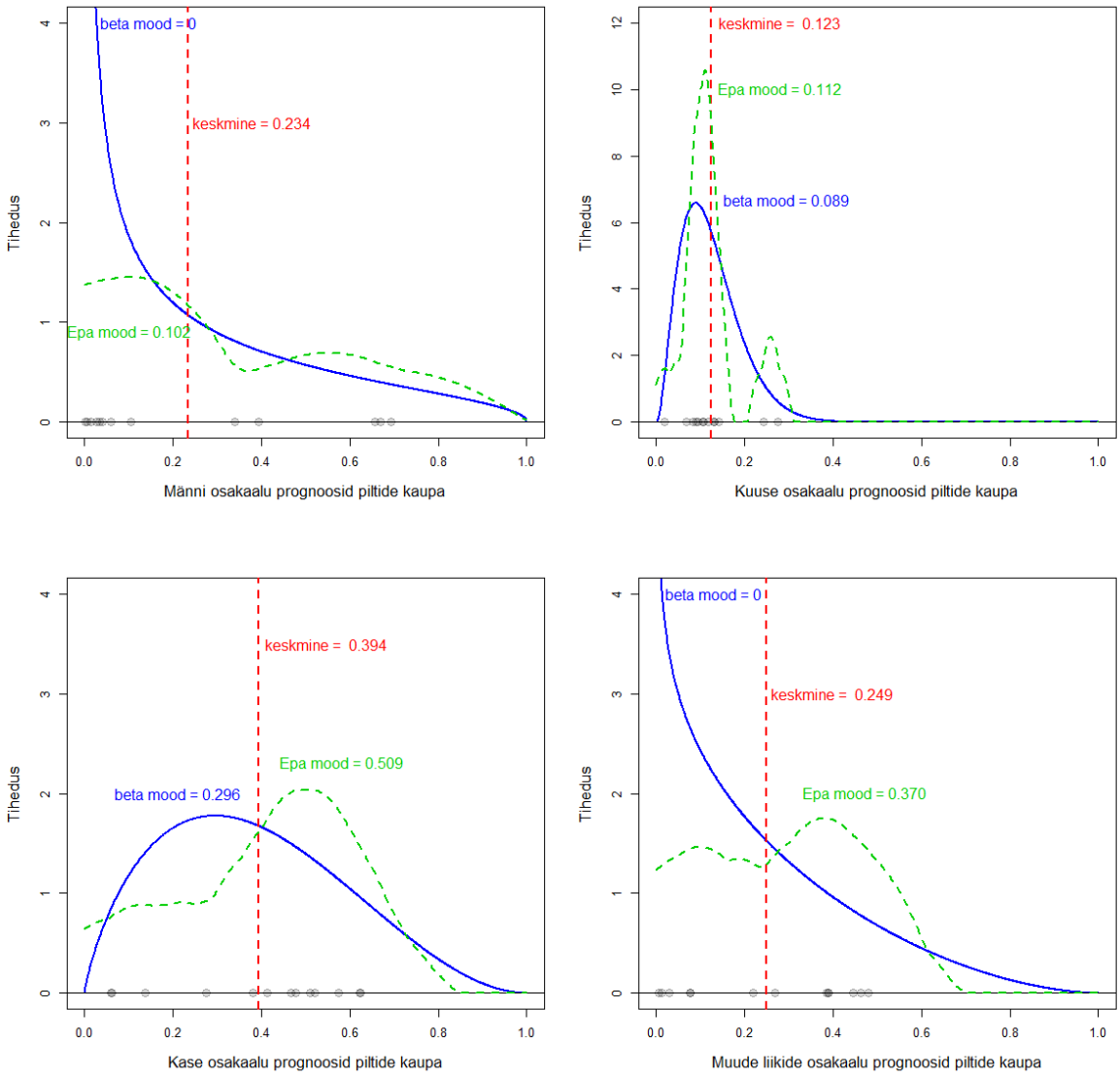
kus tuumafunktsioon *K* peab täitma tingimusi:

$$\int K(x)dx = 1, \quad \int xK(x)dx = 0 \quad ja \int x^2K(x)df = 0 \text{ [19].} \tag{15}$$

Antud juhul on tuumafunktsiooni rollis töös juba kasutust leidnud Epanechnikovi tuumafunktsioon (valem 5).

Kuna tihedusfunktsiooni moodi kasutamine agregeeriva funktsioonina ei taga, et puuliikide osakaalude prognooside summa oleks 1, tuleb sel viisil saadud prognooside vektor normeerida.

Agregeerimistulemused võivad olla erinevate meetodite korral väga erinevad. Joonisel 10 toodud näide on üks ekstreemsemaid juhte: ekstreemsus väljendub selles, et beta-funktsiooni tiheduse maksimumi põhjal agregeerides tuleb antud takseerala puuliikide osakaalude summaks enne osakaalude vektori normeerimist vaid 0.38.



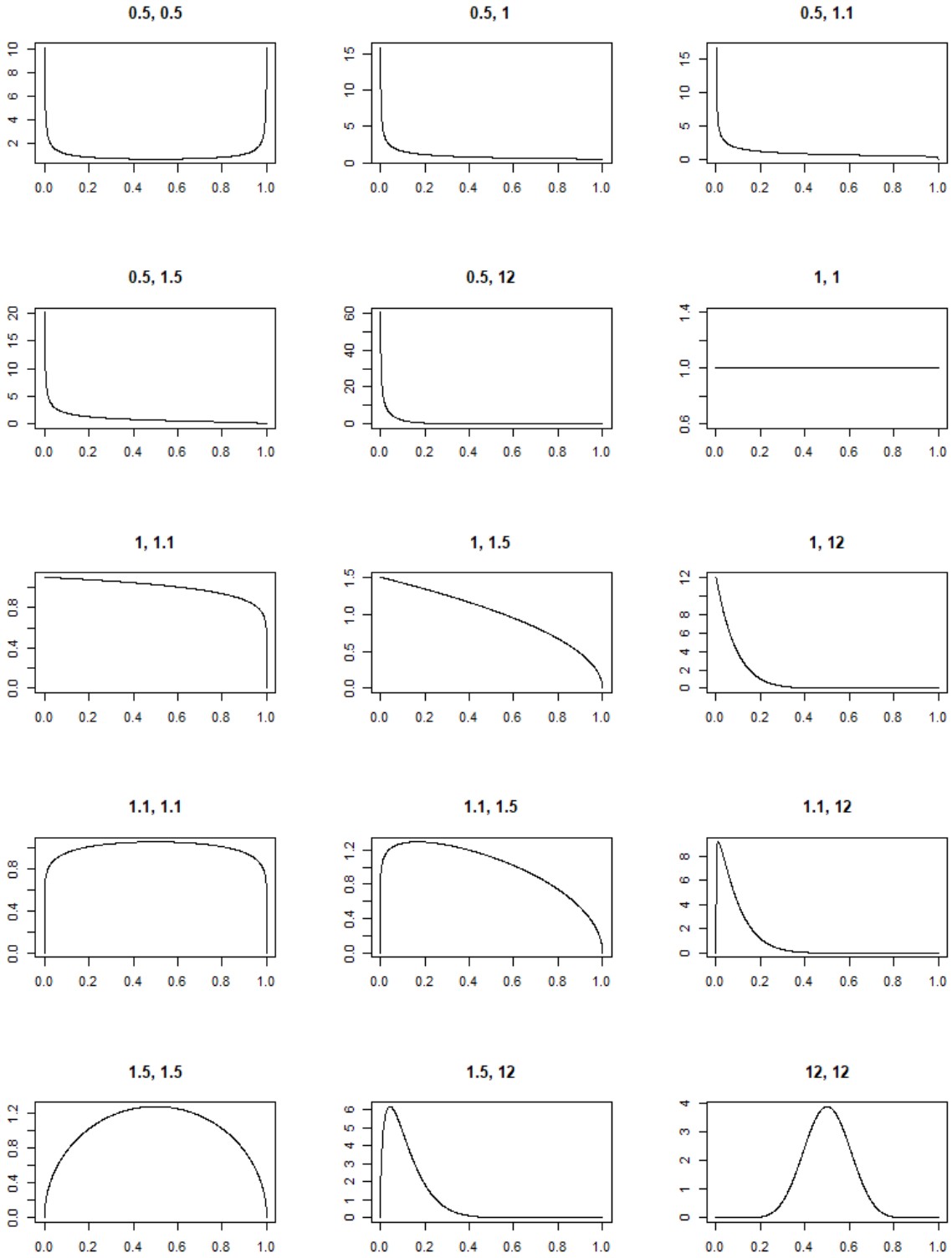
Joonis 10. Aritmeetiline keskmine, beta-funktsiooni tihedus ja Epanechnikovi tuumameetodil hinnatud tihedus ühe konkreetse takseerala pilt-haaval prognooside pealt hinnatuna.

Kuuse osakaalu prognoosid on enne osakaalude vektorite normeerimist üpris sarnased: aritmeetilise keskmise korral on antud takseerala kuuse osakaalu prognoosiks 0.123, Epanechnikovi tuumameetodil saadud tiheduse moodi korral (edaspidi epa-prognoos) on kuuse osakaalu prognoosiks 0.112 ning beta-funktsiooni tiheduse moodi korral (edaspidi beta-prognoos) 0.089. Pärast puuliikide prognooside vektori normeerimist jääb epa-prognoos peaaegu samaks (kuna puuliikide osakaalude summa $\cong$ 1), ent beta-prognoos = 0.089/0.38 = 0.234, mis on märkimisväärselt erinev nii aritmeetilisest keskmisest kui ka epa-prognoosist.

Teiste puuliikide prognoosid on erinevad nii enne kui ka pärast osakaalude vektori normeerimist. Peamine erinevus tuleb sellest, et männi ja muude liikide beta-funktsiooni tiheduse mood on andmete pealt hinnatud beta-funktsiooni kujuparameetrite korral 0. Beta-funktsiooni tiheduse mood käitubki halvasti teatud kujuparameetrite väärtuste korral. Joonisel 10 kujutatud beta-funktsioonide parameetrid on vastavalt 0.4 ja 1.5, 3.0 ja 21.0, 1.8 ja 3.0 ning 0.8 ja 2.7. Joonis 11 kujutab beta-funktsiooni tihedust erinevate kujuparameetrite väärtuste korral. Kui mõlema kujuparameetri väärtus on väiksem kui 1, siis on beta-jaotuse tihedus bimodaalne. Sellistel juhtudel on beta-prognoos asendatud aritmeetilise keskmisega (neid on aga vaid 2 juhtu 455 takseerala ja 4 liigi kohta, ehk ~0.1%). Kui ühe kujuparameetri väärtus on väiksem kui 1, nagu joonisel 10 männi ja muude liikide korral, siis on mood vastavalt kas 0 või 1. Kui mõlemad parameetrid on 1, siis on tegu ühtlase jaotusega. Kenasti esile kerkivat tippu ei ole ka neil juhtudel, kui mõlemad parameetrid ei ole oluliselt suuremad kui 1.

Neist puudustest hoolimata võiks pilt-haaval saadud prognooside põhjal hinnatud beta-jaotuse tiheduse moodi kasutamine sobida agregeerivaks funktsiooniks, kuid antud juhul töötab siiski kehvemini kui aritmeetiline keskmine: $RMSE_{keskmine} = 0.1655$, $RMSE_{beta} = 0.1702$. Seejuures saavutatakse beta-prognooside minimaalne RMSE suurema naabrite arvu K korral. Joonisel 12 tuleb hästi esile, et kui agregeeriva funktsioonina kasutada aritmeetilist keskmist, siis prognoos on täpsem väikeste K väärtuste korral: parim RMSE saavutatakse $K = 2$ korral, ent 2 kuni 6 naabri korral on tulemus üpris lähedane. Seevastu beta-prognooside korral saavutatakse parim RMSE tunduvalt suurema naabrite arvu $K = 9$ korral.

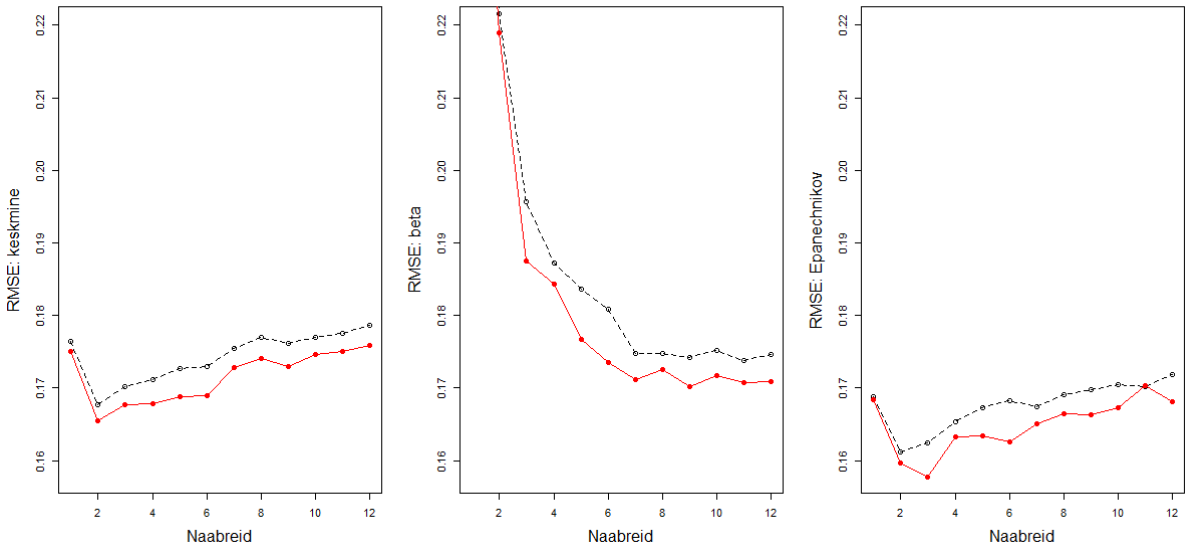
Jooniselt 12 on näha, et epa-prognoosid on täpsemad nii aritmeetilisest keskmisest kui ka beta-prognoosidest. Sarnaselt aritmeetilisele keskmisele on epa-prognoos täpsem väikeste K väärtuste korral: $RMSE_{epa} = 0.1578$ kui $K = 3$. Epanechnikovi meetodi eeliseks pilt-haaval hinnangute agregeerimisel võib olla tema tundetus erindite suhtes, mis tuleb hästi esile ka joonisel 10. Nii selles töös kasutatavate andmete korral kui ka mujal [5] on pilved ja nende varjud satelliidipiltidelt eraldatud käsitsi, st loodud manuaalne pilvemask. Tundetus erindite suhtes võib anda võimaluse loobuda pilvemaskidest või vähemalt usaldada nende loomine automaatsetele protseduuridele.



Joonis 11. Beta-jaotuse tihedus erinevate kujuparameetrite väärtuste korral.

Joonis 12 kontekstis tuleb üle rõhutada, et üksikute piltide prognoosid on sõltuvalt pildist täpsemaid enamasti 16-20 naabri kasutamise korral (joonis 9), kuid sõltumata

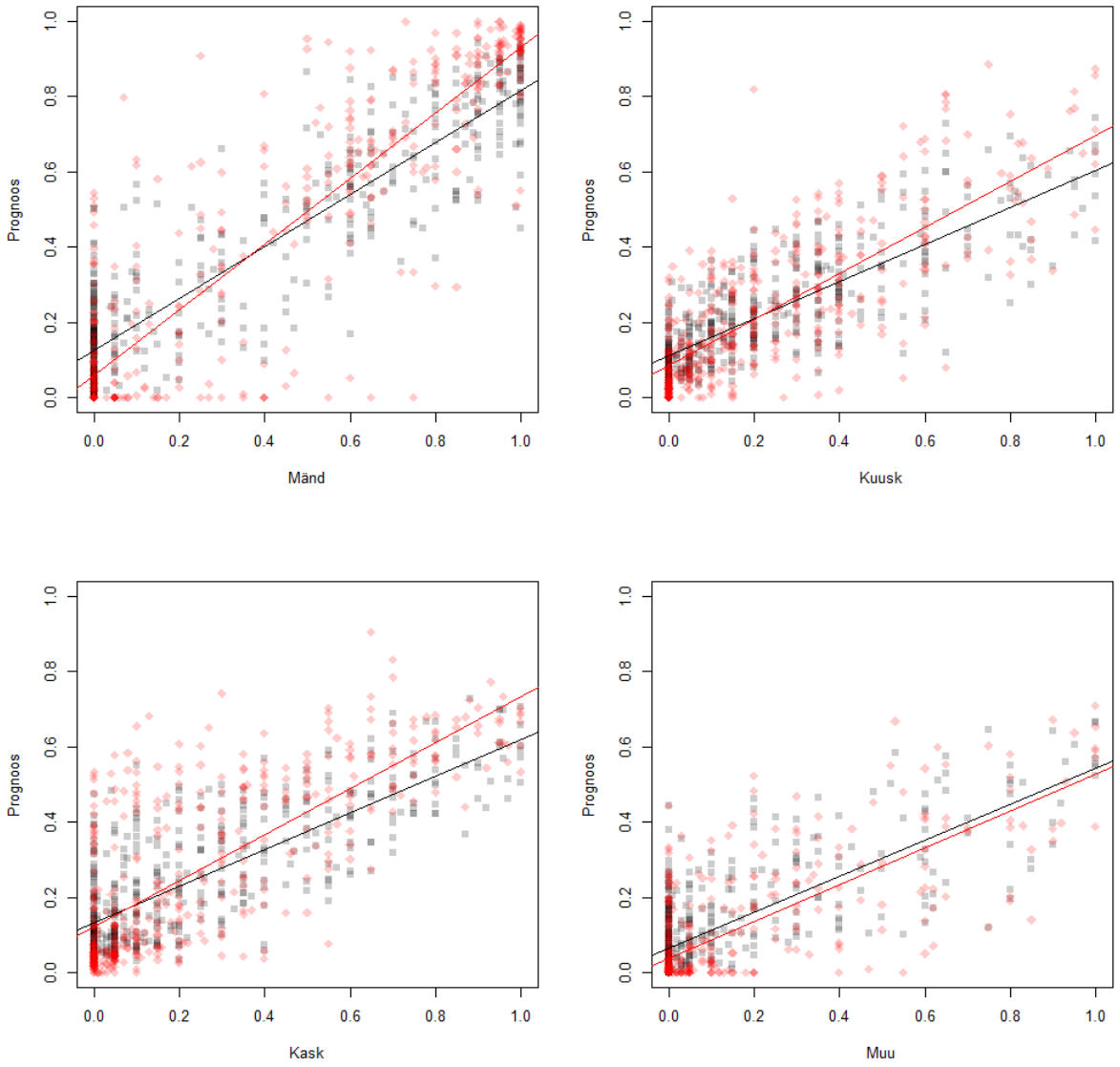
agregeerimismeetodist ja optimeerimisest saavutatakse parim üldine prognoos kindlasti väikesema arvu naabrite korral.



Joonis 12. KNN: agregeeritud pilt-haaval prognooside ruutkeskmine viga sõltuvalt naabrite arvust erinevate agregeerimismeetodite korral. Must katkendjoon tähistab tulemust enne tunnuste kaalude optimeerimist, punane joon pärast optimeerimist.

Kuigi aritmeetiline keskmine ja beta-prognoos on RMSE mõttes lähedased ($RMSE_{keskmine} = 0.1655$, $RMSE_{beta} = 0.1702$), siis vigade käitumises on olulisi erinevusi. Joonisel 13 on näha, et beta-prognooside trendijoon on palju parem (lähemal ideaalile) kui aritmeetilise keskmise abil saadud prognooside trendijoon. Peamiselt tuleneb see sellest, et proportsioonid, mille väärtus on 0, on prognoositud nullile lähemale kui aritmeetilise keskmise korral. Samuti on väga suurte proportsioonide prognoosid kõikide puuliikide korral täpsemad. Paraku on aga suuri (ühelähedasi) väärtusi vaid murdosa võrreldes nullilähedaste väärtustega (joonis 5).

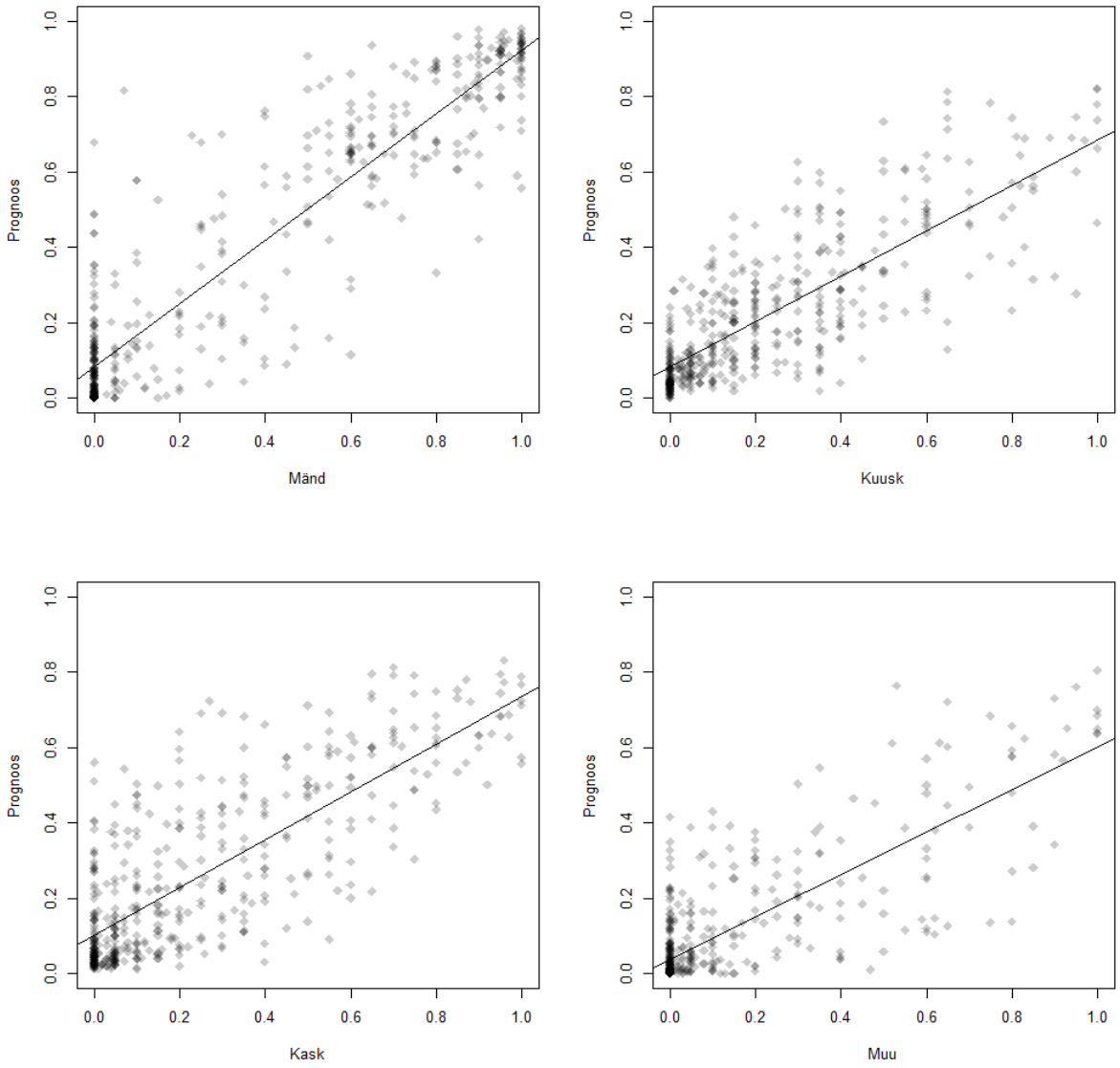
Beta-prognooside positiivsete külgede kõrval on loomulikult näha, et prognooside hajuvus on tunduvalt suurem kui aritmeetilise keskmise korral. Näiteks männi puhul on näha, et nulliks on prognoositud mitmed takseeralad, kus tegelikult on mändi isegi enam kui 50%. Keskmise-prognooside korral on nende takseeralade osakaalud täpsemini prognoositud.



Joonis 13. KNN: beta-funktsiooni tiheduse moodi abil agregeeritud pilt-haaval prognoosid (punased ruudud), RMSE = 0.1702 ja samade pilt-haaval prognooside aritmeetiline keskmine (mustad rombid), RMSE = 0.1655.

Epa-prognooside hajuvus trendijoone ümber ehk lineaarse mudeli standardviga (SE) on samuti suurem kui keskmise-prognooside korral. Näiteks männi korral $SE_{keskmine}$ = 0.1194, SE_{beta} = 0.1637 ja SE_{epa} = 0.1404.

Kuna aga epa-prognooside trendijooned on liikide kaupa praktiliselt samad, mis beta-prognooside trendijooned, seega paremad kui keskmise-prognooside korral, siis kokkuvõttes ongi epa-prognoos kõige täpsem: RMSE = 0.1578 (joonis 14).



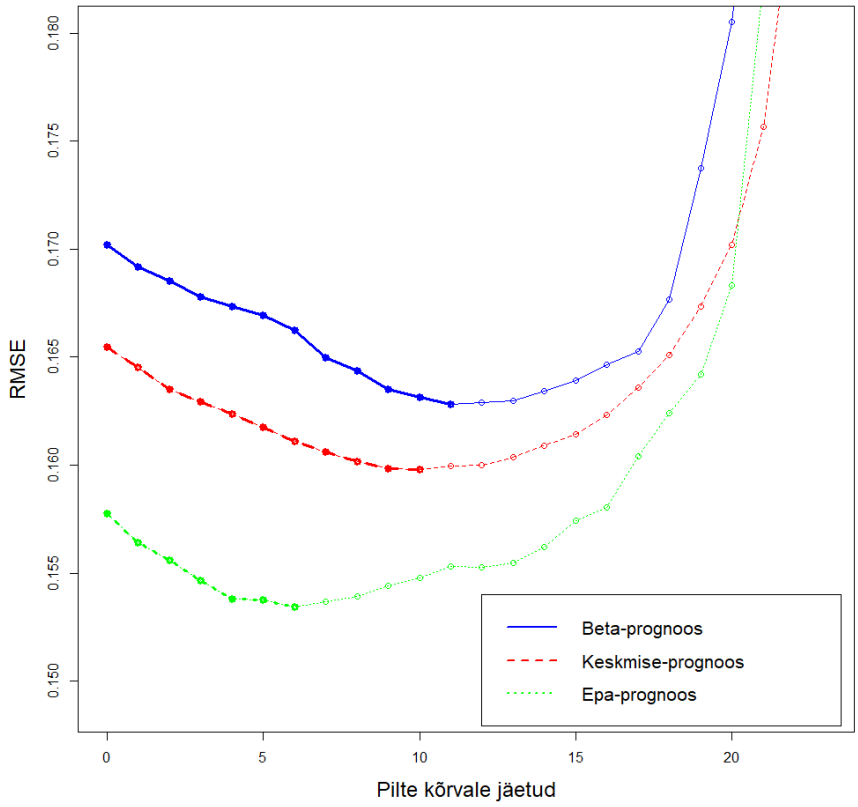
Joonis 14. KNN: Epanechnikovi tuumafunktsiooni tiheduse moodi abil agregeeritud pilt-haaval prognoosid. RMSE = 0.1578.

4.2.2 Prognoosid valitud piltide korral

Jooniselt 8 oli näha, et mõnel aastaajal tehtud piltide korral on prognoosid täpsemad, näiteks mai lõpus (~150. päev aastas) ja juuni alguses tehtud piltide RMSE on selgelt parem kui mai alguses (~ 120. päev aastas) tehtud piltidel. Mõne pildi peale on jäänud ka väga vähe takseeralasid. Eelneva põhjal võib eeldada, et mingitel alustel kehvemaid pilte välja jättes on võimalik üldist prognoositäpsust parandada.

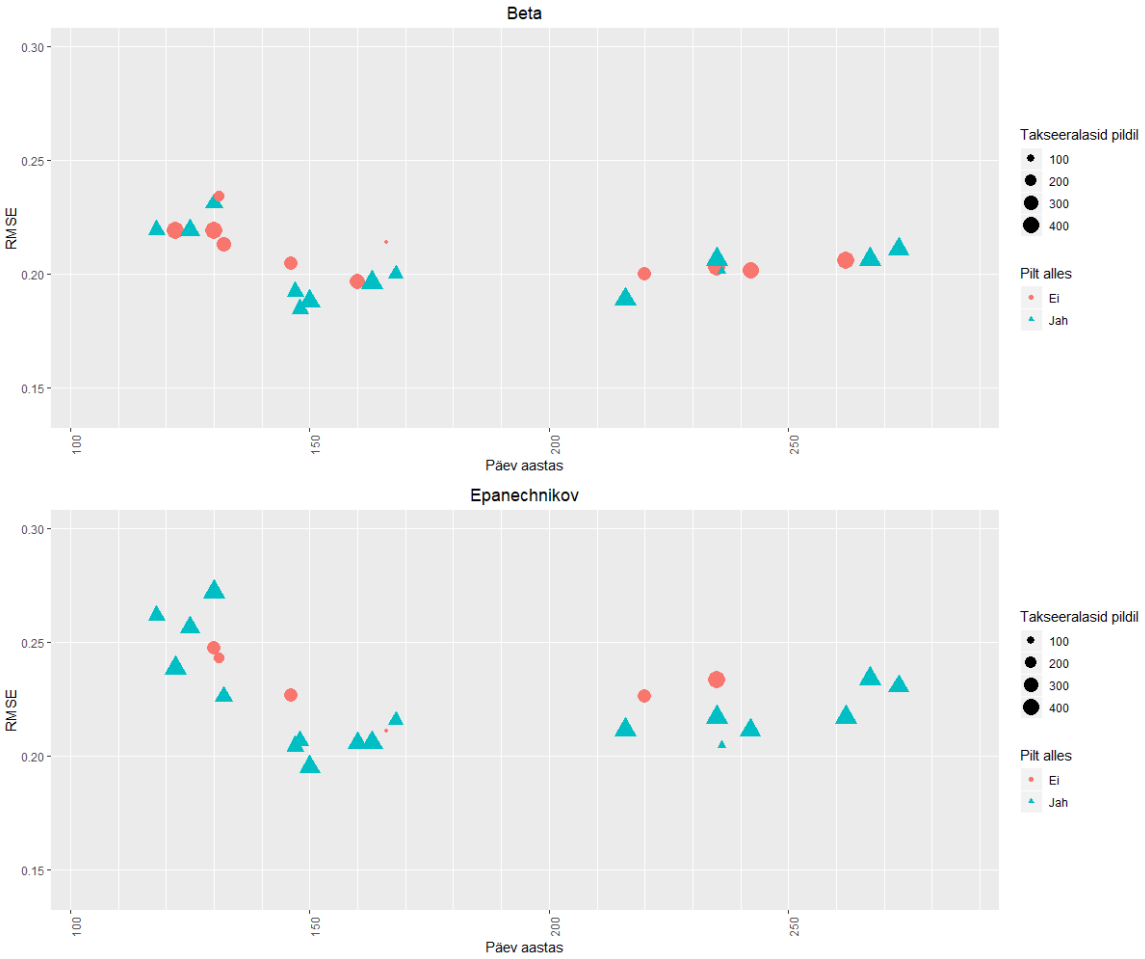
Pilte on välja jäetud samm-sammult: esmalt pilt, mille välja jätmisel RMSE paraneb kõige enam jne, kuni järgmise pildi kõrvale heitmine prognoosi enam ei paranda. Sel viisil prognooside paranemist kujutab joonis 15. Prognoos paraneb kõikide agregeerimismeetodite korral: keskmise-prognoosi parandatud RMSE = 0.1598, kui kõrvale on jäetud 10 pilti, beta-prognoosi RMSE = 0.1628, kui kõrvale on jäetud 11 pilti ning epa-prognoosi RMSE = 0.1534 kui kõrvale on jäetud 6 pilti.

Kõikide prognooside korral on kehvemate piltide väljajätmise mõju üsna ühtlane, st ei leidu mingi konkreetse pildi või piltide väljajätmisest tingitud suurt RMSE langust. Epa-prognoos on ka pärast kehvemate piltide kõrvale heitmist parim ning lisaks on välja visatud vähem pilte kui teiste meetodite korral. Eelneva põhjal on siiski prognooside koostamisel mingi konkreetse pildi kõrvale jätmist raske põhjendada.



Joonis 15. KNN: agregeeritud pilt-haaval prognooside paranemine valikuliselt kehvemaid pilte välja jättes.

Joonisel 16 on näha, et prognoosi parandamiseks on pilte välja visatud erinevatel aastaaegadel ning erinevate pildi kohta olevate vaatluste arvu korral. Piltide välja viskamine ei beta-prognoosi ega epa-prognoosi parandamiseks ei tundu alluvat mingile reeglile. Küll jääb aga silma, et beta-prognoosis kasutatud üksikute piltide prognoosid on täpsemad: neis prognoosides on kasutatud 9 lähima naabri infot, epa-prognoosi korral aga kolme naabri infot.

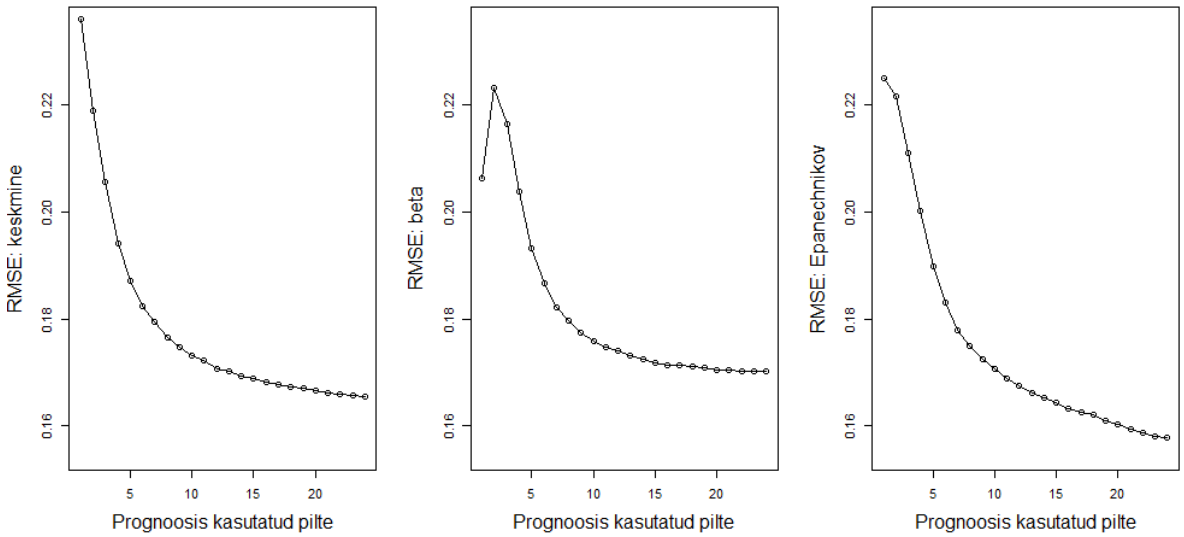


Joonis 16. KNN: agregeeritud pilt-haaval prognoosi parandamiseks välja visatud pildid.

4.2.3 Piltide arvu mõju agregeeritud pilt-haaval prognoosi täpsusele

Agregeeritud pilt-haaval prognoos on parem kui koondandmestiku põhjal saadud prognoos: prognoosi vead vastavalt RMSE = 0.1578 ja 0.1616. Agregeeritud pilt-haaval prognoosi puhul võib arvata, et piltide arvu kasvades prognoosi täpsus paraneb veelgi. Sama kehtib loomulikult ka koondandmestiku kohta, ent seal ei ole mõju nii hõlpsasti mõõdetav.

Sentineli ja Landsati pilte on kokku 24. Eesmärk on arvutada kõikide piltide arvu 1 kuni 24 korral iga võimaliku piltide kombinatsiooni korral eraldi prognoos ning keskmistada prognooside RMSE vastavalt piltide arvule. Paraku on sellisel juhul arvutusprotsess liiga ajamahukas, näiteks 12 pildi korral tuleb leida $C_{12}^{24} \cong 2.7\, miljonit\ prognoosi$, seetõttu on piltide arvu r korral kui $C_r^{24} > 500$ piltide arvule vastav veahinnang arvutatud 500 juhusliku vastava piltide arvuga komplekti pealt. Erinevate agregeerimismeetodite korral on kasutatud naabrite arvu, mis andis antud meetodi korral kõigi 24 pildi kasutamise korral parima tulemuse (joonis 12). Tulemused on joonisel 17: nagu eeldatud, paraneb prognoosi täpsus prognoosis kasutatud piltide arvu suurenedes, kusjuures kõige suuremas langustrendis on kõigi 24 pildi korral parimat tulemust näidanud epa-prognoos.



Joonis 17. KNN: prognoosi RMSE sõltuvalt prognoosi arvutamiseks kasutatud piltide arvust. Beta-jaotuse parameetrite hindamiseks on vaja vähemalt kahte unikaalset väärtust – ühe pildi korral on beta-prognoosi RMSE väiksem kui keskmise-prognoosi RMSE, kuna beta-prognoosi üksikud pilt-pilt-haaval prognoosid on täpsemad.

4.3 Ühemõõtmeline K-lähima naabri meetod

Kuna *K*-lähima naabri meetod on antud töö valdkonnas laialt levinud, siis võrdluseks mitmemõõtmelisele lähenemisele on puuliikide osakaalud prognoositud ka eraldiseisvalt. Ühemõõtmelise lähenemise korra hinnatakse igale liigile eraldi KNN mudel ning seejärel osakaalude liigikaupa prognooside vektor normeeritakse. Naabrite kaugused on teisendatud kaaludeks Epanechnikovi tuumafunktsiooni abil.

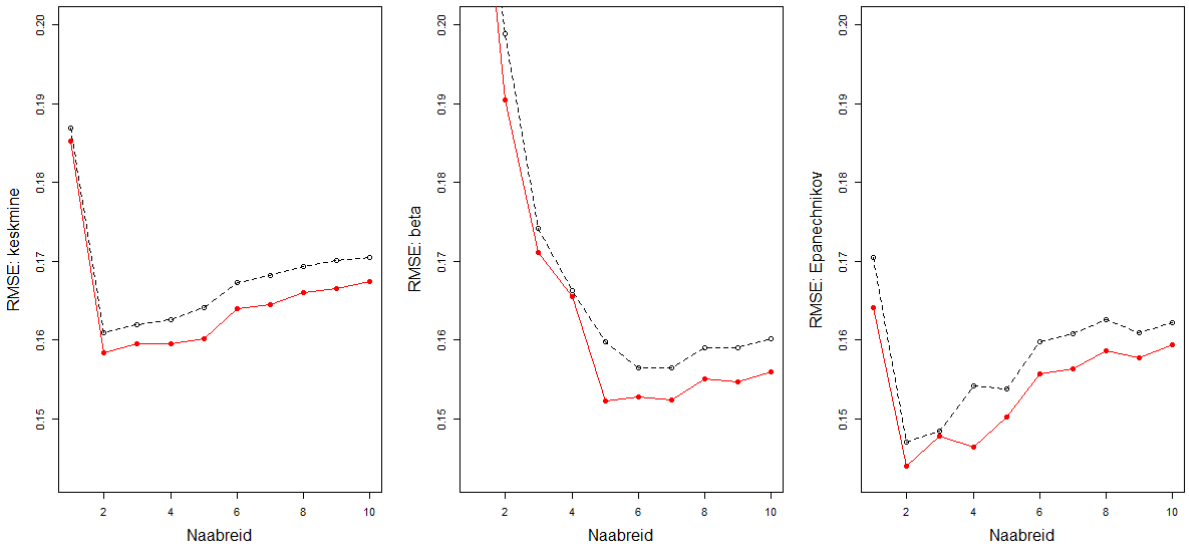
Koondandmestiku korral saavutatakse täpseim liigi osakaalu prognoos vastavalt naabrite arvu mänd 9; kuusk 16; kask 11 ja muu 11 puhul. Nõudes kõikide liikide korral sama naabrite arvu, saavutatakse $K = 11$ korral, pärast osakaalude vektorite normeerimist, RMSE = 0.1534. Erineva naabrite arvu lubamine annab marginaalse võidu, RMSE = 0.1523, kusjuures kõikide liikide korral ei ole naabrite arv sama, mis eraldi võttes parima prognoosi korral.

Mitmemõõtmelise lähenemise korral osutus pilt-haaval lähenemine täpsemaks ja seetõttu sel viisil saadud tulemused põhjalikumalt lahti kirjutatud. Lihtsuse huvides ja töö eelmist osa arvesse võttes on iga pildi korral kasutatud sama naabrite arvu. Samuti on kõikide liikide puhul naabrite arv sama.

Liigikaupa prognoositud osakaalude vektori normeerimiseks on kaks võimalust:

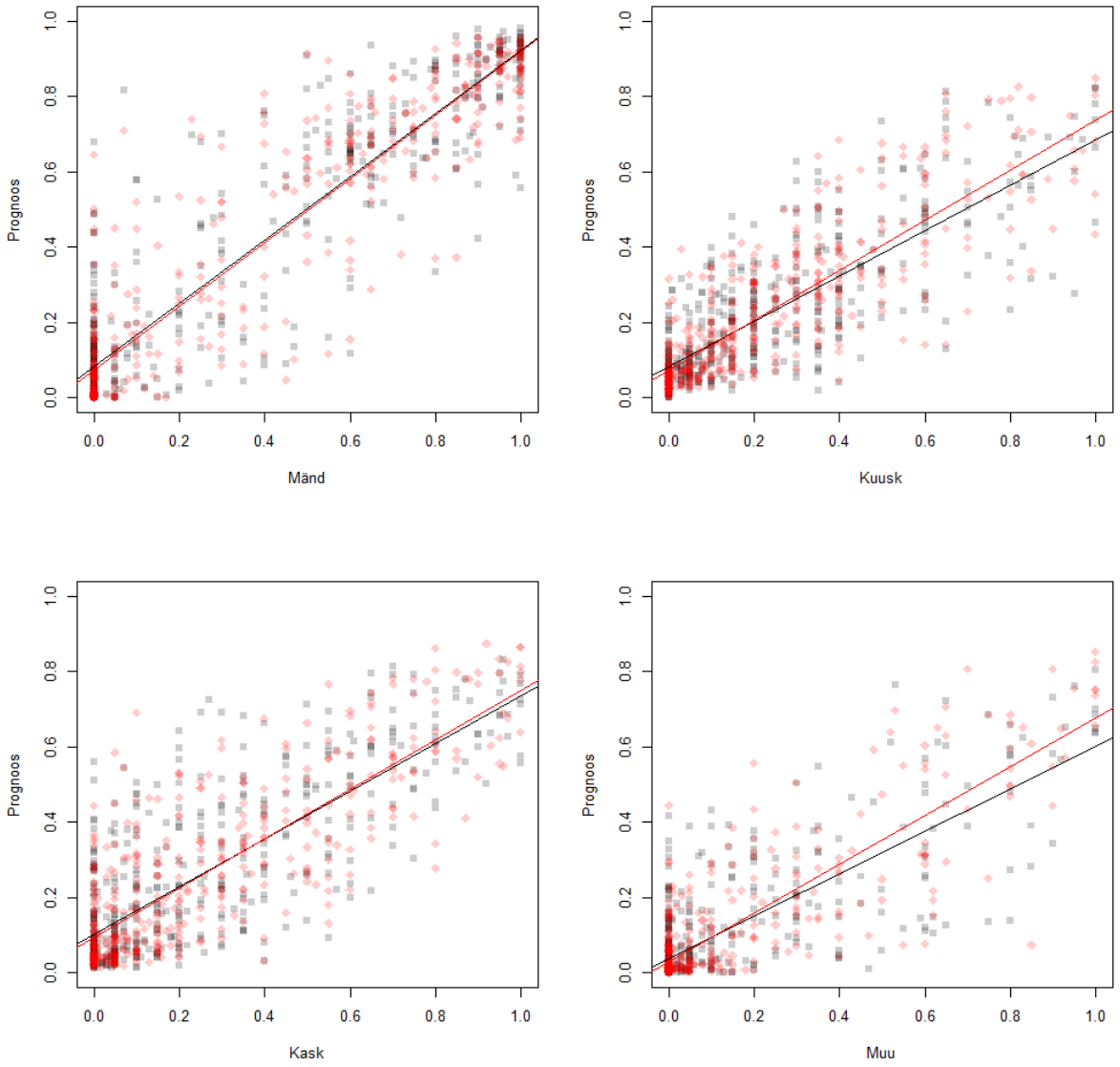
1. Normeerida iga takseerala puuliikide proportsioonide vektor iga pildi korral, seejärel agregeerida pilt-haaval prognoosid ning lõpuks normeerida uuesti iga takseerala kohta käiv puuliikide vektor.
2. Normeerida vektor ainult viimase sammuna.

Erinevad lähenemised annavad väga sarnase tulemuse parima keskmise-prognoosi korral: 0.1584 vs 0.1587 esimese meetodi kasuks ning samuti parima epa-prognoosi korral: 0.1440 vs 0.1437 teise meetodi kasuks. Suur erinevus on aga beta-prognooside korral: 0.1523 vs 0.1630 esimese meetodi kasuks. Jooniselt 18 on toodud esimesel meetodil arvutatud prognooside vead sõltuvalt naabrite arvust. Nagu näha, käituvad erinevad agregeerimismeetodid väga sarnaselt pilt-haaval prognooside mitmemõõtmelisele prognoosimisele (joonis 12): keskmise-prognoos ja epa-prognoos saavutavad väikseima RMSE väga väikese arvu naabrite korral, $K = 2$, beta-prognoos käitub paremini aga suurema arvu naabrite korral: $K = 5$. Erinevalt mitmemõõtmelisest prognoosimisest on beta-prognoosid antud juhul paremad kui keskmise-prognoosid.



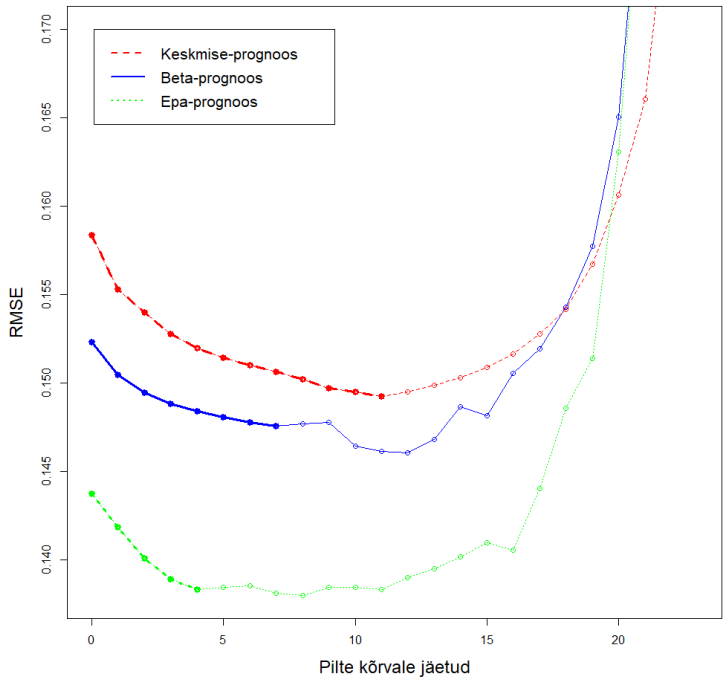
Joonis 18. Agregeeritud pilt-haaval prognooside ruutkeskmine viga sõltuvalt naabrite arvust erinevate agregeerimismeetodite korral. Igale liigile on hinnatud eraldi mudel. Must katkendjoon tähistab tulemust enne tunnuste kaalude optimeerimist, punane joon pärast optimeerimist.

Parima prognoosi korral RMSE = 0.1437, mis on parem kui mitmemõõtmeliste meetodite korral saadud parim prognoos. Joonisel 19 on näha, et üldiselt käituvad samal meetodil agregeeritud pilt-haaval prognoosid väga sarnaselt. Mõlemal juhul on tegemist epa-prognoosidega, mitmemõõtmelisel juhul $K = 3$, ühemõõtmelisel juhul $K = 2$. Kõikide liikide korral on trendijoon ühemõõtmelise KNN meetodi korral kergelt parem kui mitmemõõtmelise meetodi korral, ent mingit süstemaatilist erinevust ei tule esile.

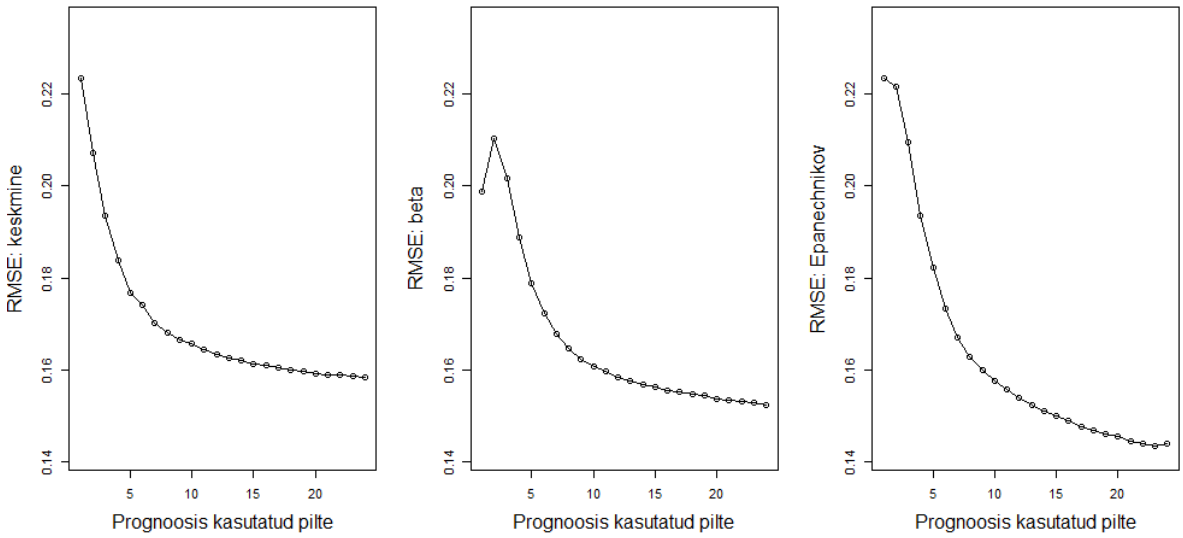


Joonis 19. KNN, ühemõõtmeline lähenemine: Epanechnikovi tuumafunktsiooni tiheduse moodi abil agregeeritud pilt-haaval prognoosid, RMSE = 0.1437 (punased rombid). Võrdluseks analoogne mitmemõõtmeline prognoos (mustad ruudud), RMSE = 0.1578 (joonis 14).

Jooniselt 20 on näha, et piltide valikuline agregeerimisest välja jätmine parandab prognoose. Pärast kehvemate piltide kõrvaldamist on epa-prognoosi RMSE = 0.1383 (4 pilti välja; varem RMSE = 0.1437). Beta-prognoosi korral on uus RMSE = 0.1475 (7; 0.1523) ning keskmise-prognoosi korral RMSE = 0.1492 (11; 0.1584). Pilte on välja visatud kuni esimese lokaalse miinimumi saavutamiseni, aga nagu jooniselt näha, siis lokaalne miinimum ei pruugi olla globaalne miinimum.



Joonis 20. KNN, ühemõõtmeline lähenemine: agregeeritud pilt-haaval prognooside paranemine valikuliselt kehvemaid pilte välja jättes.



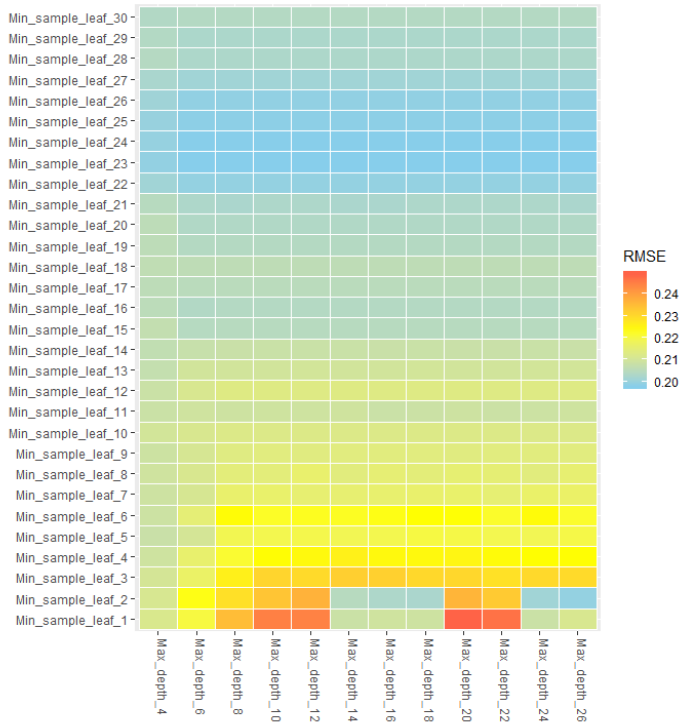
Joonis 21. KNN, ühemõõtmeline lähenemine: prognoosi RMSE sõltuvalt prognoosi arvutamiseks kasutatud piltide arvust

Vigade vähenemine sõltuvalt prognoosis kasutatud piltide arvust joonisel 21 on nagu varasemaltki arvutatud 500 juhusliku vastava piltidega arvuga kompleki pealt. Võrreldes mitmemõõtmelise juhuga (joonis 17) käitub beta-prognoos piltide arvu kasvades paremini.

Kokkuvõttes töötab *K*-lähima naabri meetodi korral pilt-haaval puuliikide osakaalude prognoosimine ning seejärel saadud tulemuste agregeerimine paremini kui koondandmestiku pealt osakaalude prognoosimine. Valikuline piltide kõrvale jätmine parandab tulemust veelgi. Parim agregeeriv funktsioon on Epanechnikovi tuumameetodil hinnatud tiheduse mood. Paraku tuleb tõdeda, et kuigi töö eesmärgiks on prognoosida puuliikide osakaalude vektorid tervikuna ehk kasutada mitmemõõtmelisi meetodeid, siis *K*-lähima naabri meetodi korral annab ühemõõtmeline lähenemine parema tulemuse.

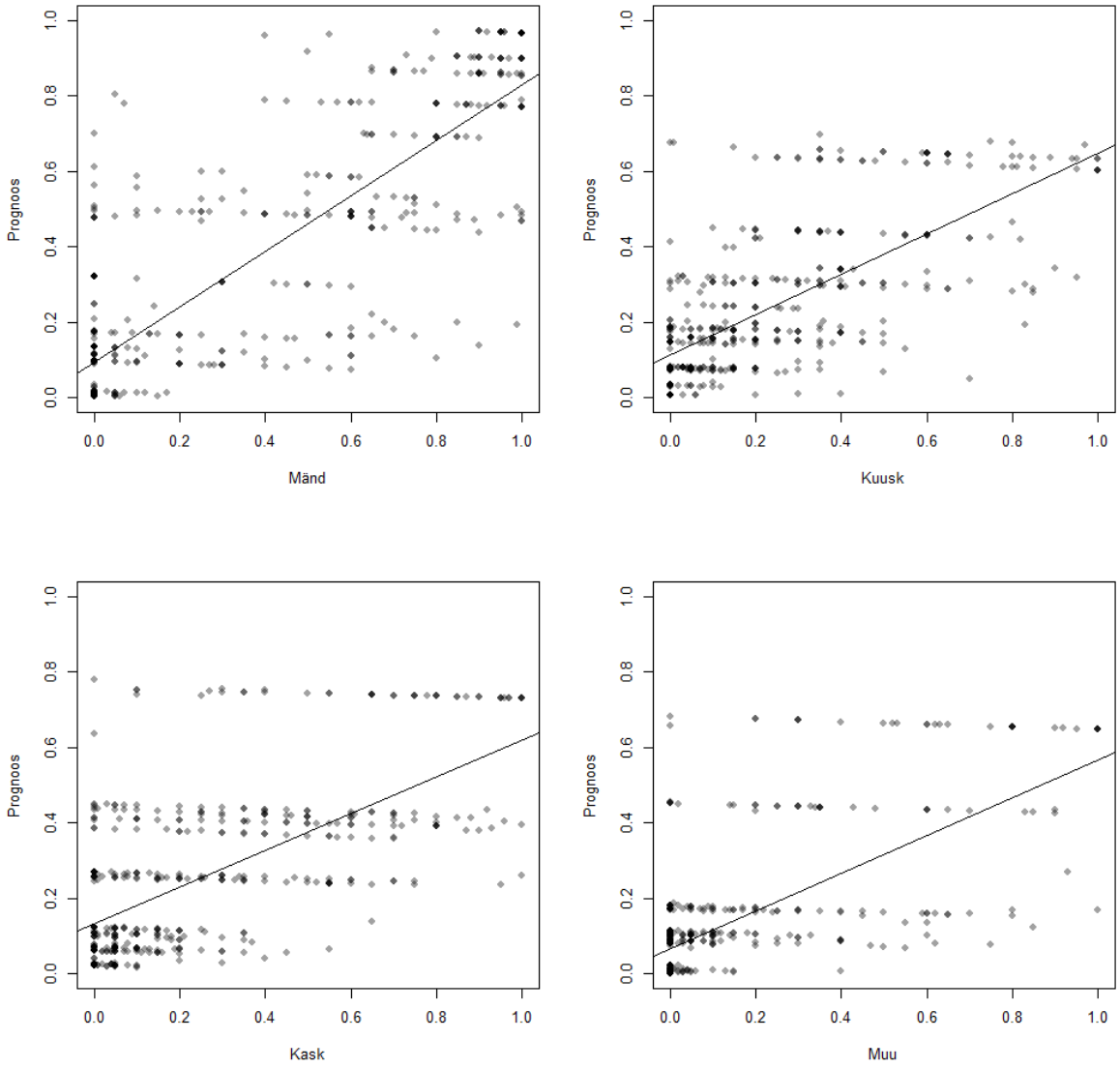
4.4 Regressioonipuu

Regressioonipuu ja juhumetsa meetodite korral on tulemused arvutatud *Python*-is kasutades teeki *scikit-learn*. Optimeeritud on kahte regressioonipuu hüperparameetrit: puu maksimaalset sügavust ehk maksimaalset teekonna pikkust puu tüvest lehtedeni (*maximum depth*) ja minimaalset nõutud vaatluste arvu lehe kohta (*minimum number of samples per leaf*, *MSL*). Nende parameetrite optimaalsete väärtuste otsingu võre on kujutatud joonisel 22.



Joonis 22. Regressioonipuu optimaalne maksimaalne sügavus (max depth) ja minimaalne vaatluste arv lehe kohta (min sample leaf).

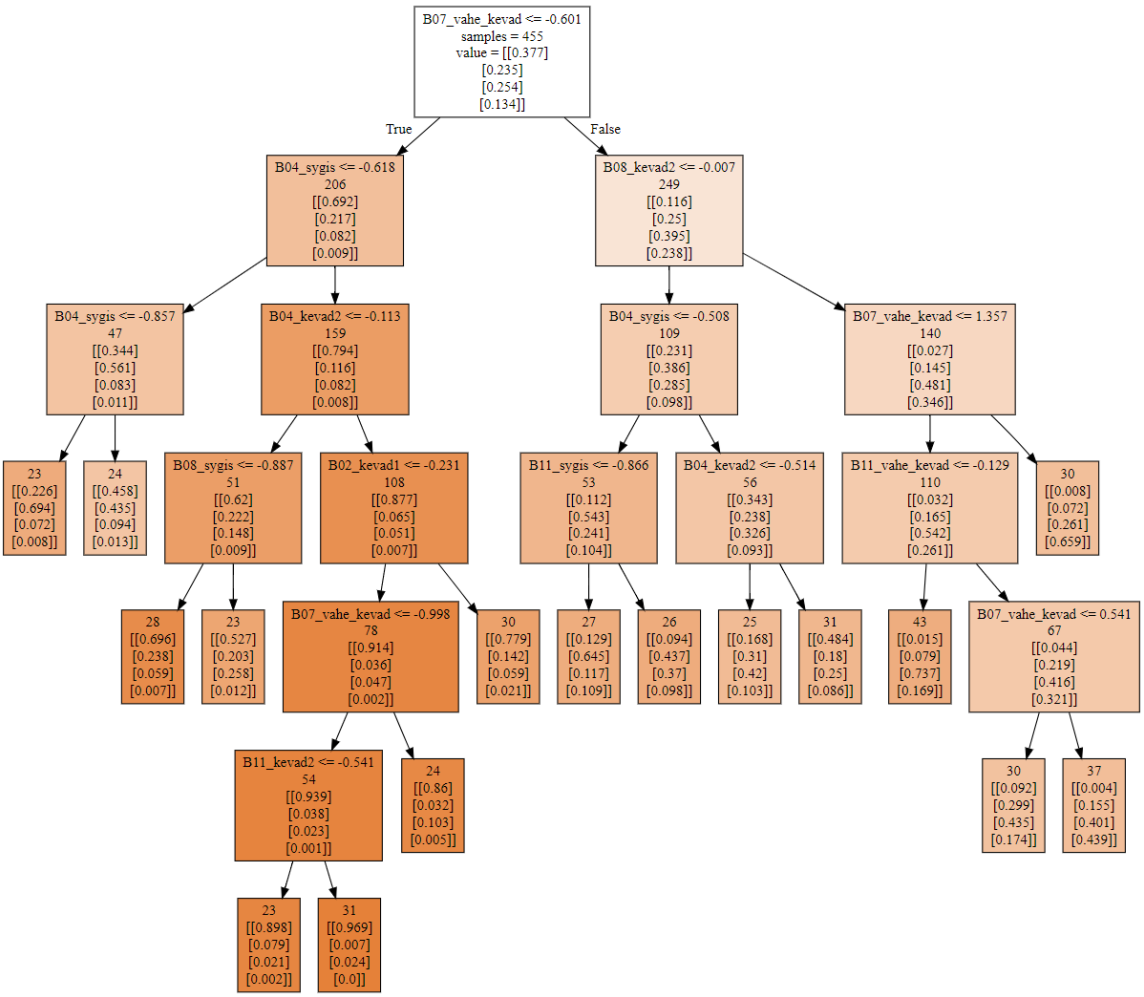
Parim RMSE saavutatakse, kui minimaalseks nõutud vaatluste arvuks lehe kohta on 23. Minimaalne vaatluste arv lehe kohta ja takseeralade arv seavad piiri maksimaalsele sügavusele, seetõttu ongi MSL = 23 korral, alates sügavusest 6, RMSE sama ehk 0.1979. Visuaalselt on tulemused kujutatud jooniselt 23. Prognoosid moodustavad tugevad horisontaalsed viirud ja tunduvad praktikas kasutamatud.



Joonis 23. Puuliikide osakaalude prognoosid regressioonipuu meetodiga. MSL = 23, maksimaalne sügavus 6. RMSE = 0.1979.

Joonisel 24 on näide regressioonipuu mudelist. Puul on 16 lehte, seega oleks joonisel 23 nähtud viirgude maksimaalne arv antud puu korral iga liigi puhul 16. Viirud joonisel 23 ei ole päris

sirged, kuna tulemused on saadud läbi jäta-üks-välja ristvalideerimise ehk iga vaatluse korral konstrueeritakse uus puu ja seega on prognoosid natuke erinevad. Joonisel 24 näiteks toodud puu on kasvatatud kõikide vaatluste pealt.

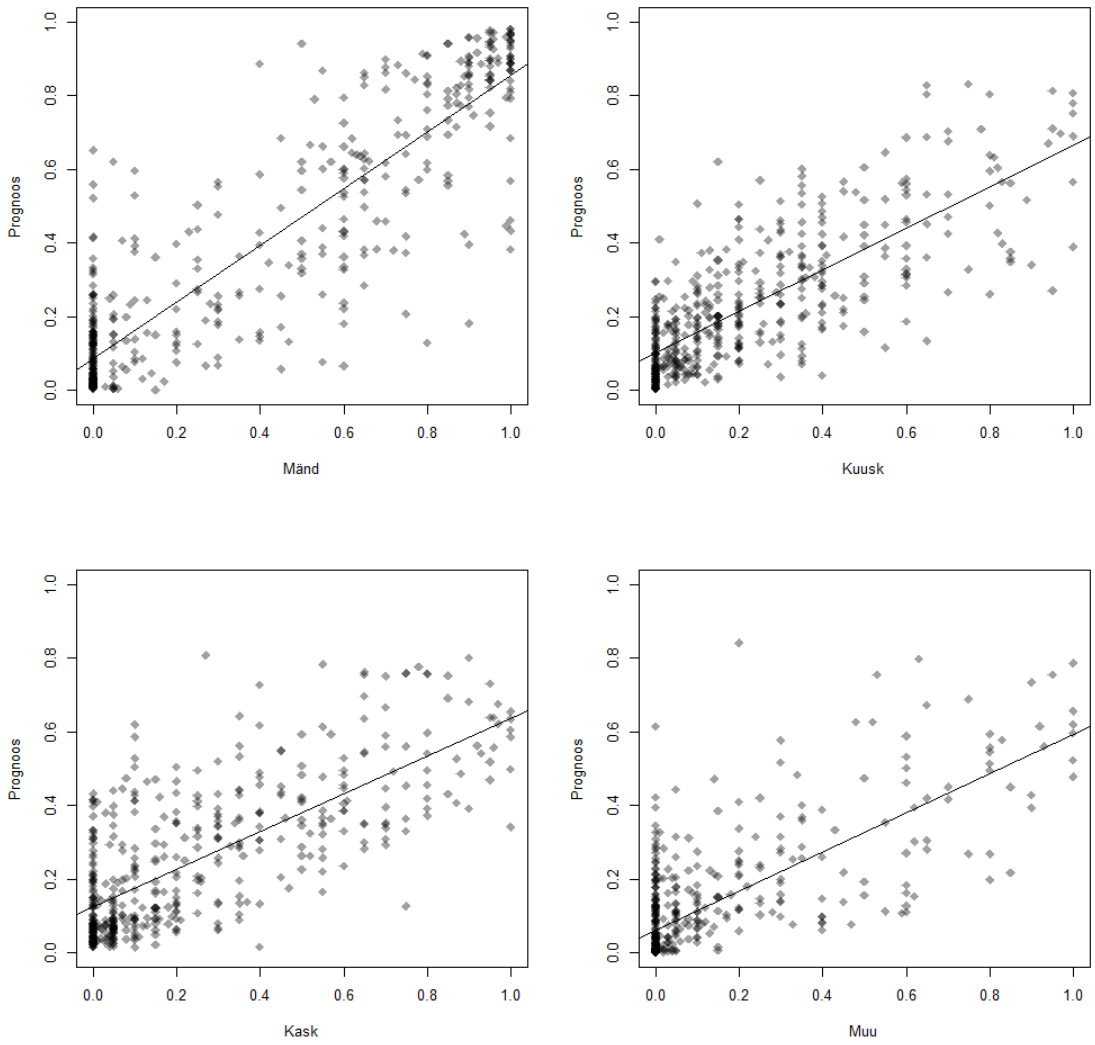


Joonis 24. Regressioonipuu näide. MSL = 23, maksimaalne sügavus 6. Iga lehe korral on näidatud vaatluste arv lehe kohta (minimaalne lubatud 23) ja uuritava tunnuse prognoos. Igal harul on lisaks näidatud ka hargnemise kriteerium. Haru või lehe tumedus vastab prognoosivektori [mänd, kask, kuusk, muu] suurimale väärtusele.

Regressioonipuu korral annab natuke parema tulemuse tunnuste eelvalik. Valides mudelisse ainult *K*-lähima naabri meetodis oluliseks osutunud 15 tunnust (tabel 2), paraneb prognoosi täpsus: RMSE = 0.1860 (MSL 21, suurim sügavus 6).

4.5 Regressioonipuu ja bagging

Erinevalt üksikust regressioonipuust kasvatatakse *baggingu* korral suure sügavusega puud [9]. Seda erinevust kinnitas ka hüperparameetrite võreotsing, mille põhjal *bootstrap* valimite (N = 250) peal ehitatud üksikute puude maksimaalne sügavus on 23 ning minimaalne vaatluste arv lehe kohta vaid 2. Tõenäoliselt piisavalt suure hulga *bootstrap* valimite korral (ehk kindlasti rohkem kui 250) võib üksikutel puudel lasta kasvada piiramatult, st piiramata maksimaalset sügavust ja minimaalset nõutud vaatluste arvu lehe kohta. Edaspidi pole puu kasvu kuidagi piiratud. Regressioonipuu ja *baggingu* korral RMSE = 0.1741. Tunnuste eelvalik antud meetodi korral võitu ei anna.

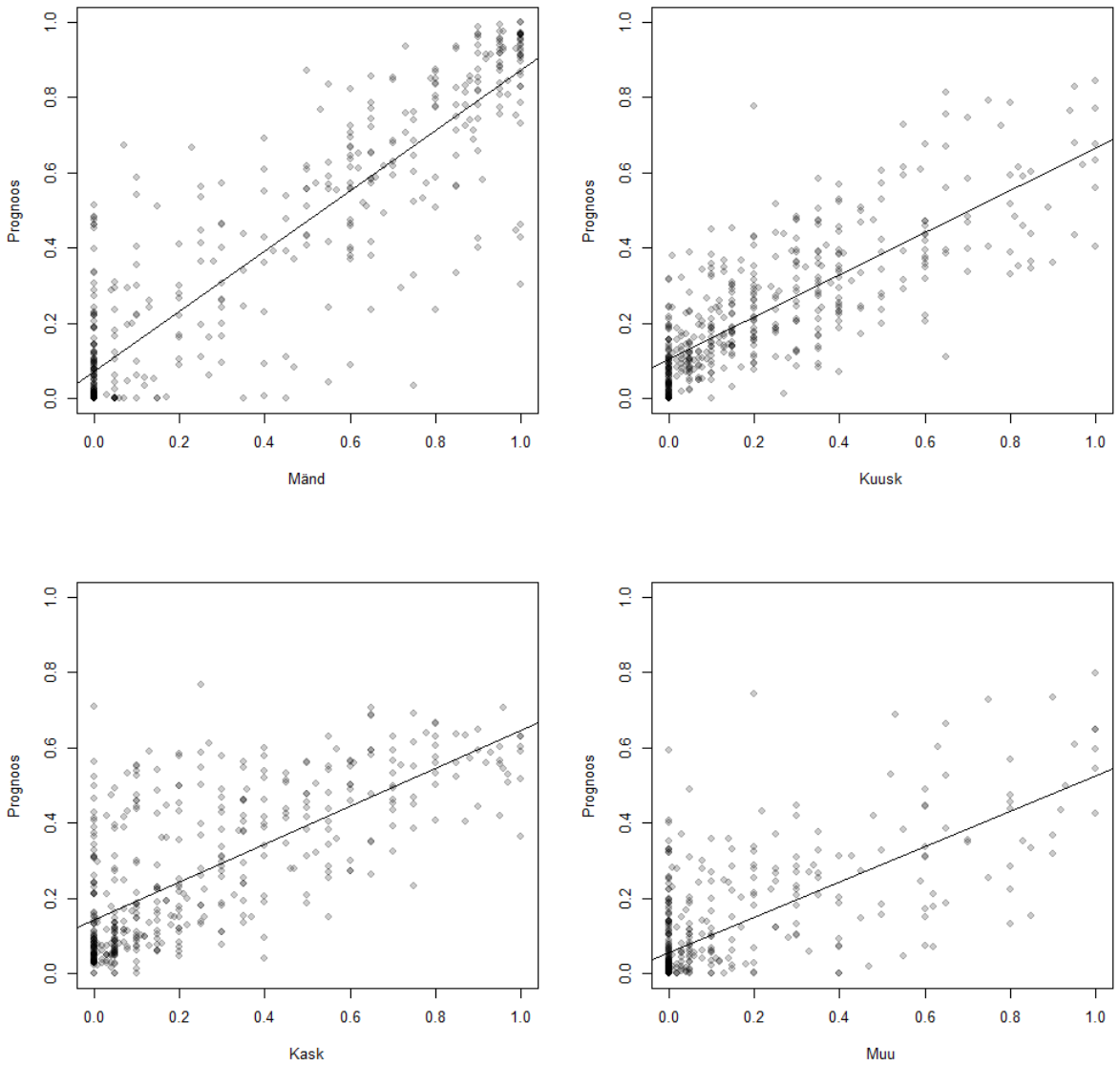


Joonis 25. Regressioonipuu ja bagging*: Minimaalne lehtede arv 2, maksimaalne sügavus 23. RMSE = 0.1741.*

4.6 Regressioonipuu ja bagging, pilt-haaval prognoosimine

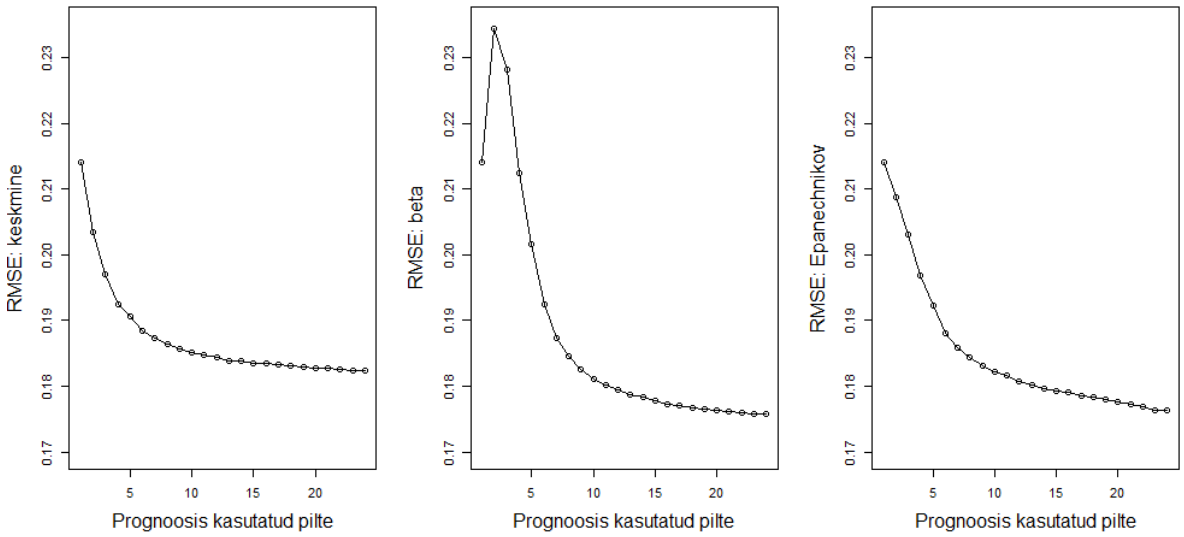
Kuna on näha, et ilma *baggingut* kasutamata töötab regressioonipuu kesiselt, siis pilt-haaval lähenemise korral on selle meetodi rakendamisest loobutud. Ka pole teostatud tunnuste eelvalikut, kuna on näha, et *baggingu* korral see võitu ei anna.

Iga pildi korral on juhumetsa kasvatamiseks kasutatud 250 üksikut puud, puude kasvu ei ole piiratud.



Joonis 26. Regressioonipuu ja bagging: *Beta-funktsiooni tiheduse moodi abil agregeeritud pilt-haaval prognoosid. RMSE = 0.1758.*

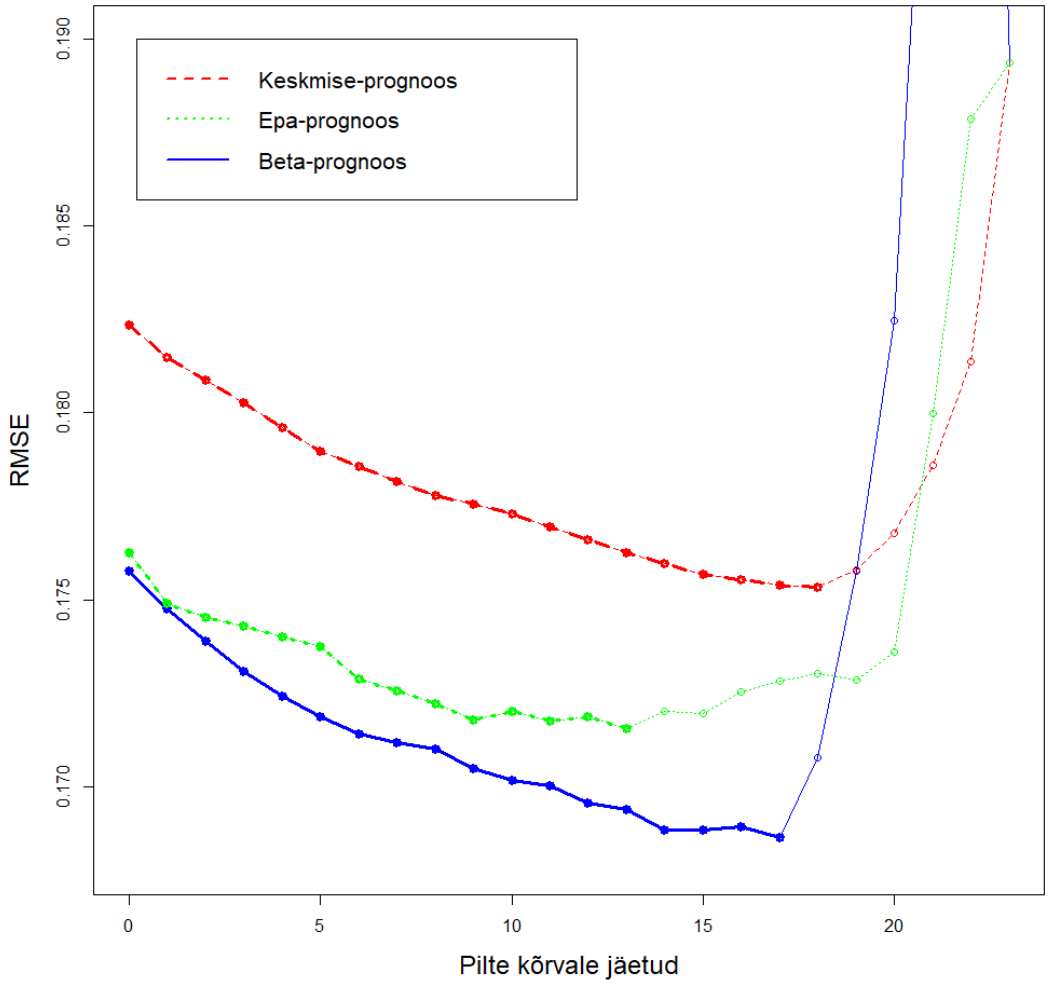
Pilt-haaval prognooside agregeerimisel on keskmise-prognoosi RMSE = 0.1824. Beta- ja epa-prognoosid on täpsemad: $RMSE_{beta}$ = 0.1758; $RMSE_{epa}$ = 0.1762 ehk praktiliselt sama, eriti arvestades mudeli juhuslikkuse faktorit. Küll jääb aga jooniselt 27 silma, et epa-prognoos käitub piltide arvu kasvades paremini kui beta-prognoos.



Joonis 27. Regressioonipuu ja bagging*: prognoosi RMSE sõltuvalt prognoosi arvutamiseks kasutatud piltide arvust.*

Pilte kõrvale heites annab jällegi beta-prognoos parima tulemuse. Jooniselt 28 on näha, et võrreldes KNN pilt-haaval lähenemisega (joonised 15 ja 20) visatakse kõikide meetodite korral oluliselt enam pilte välja: keskmise-prognoosi korral lausa 18 pilti ehk alles jääb 6 pilti; beta-prognoosi korral visatakse välja 15 pilte ning epa-prognoosi korral 9 pilti. Pärast piltide välja jätmist on parima prognoosi $RMSE_{beta}$ = 0.1688.

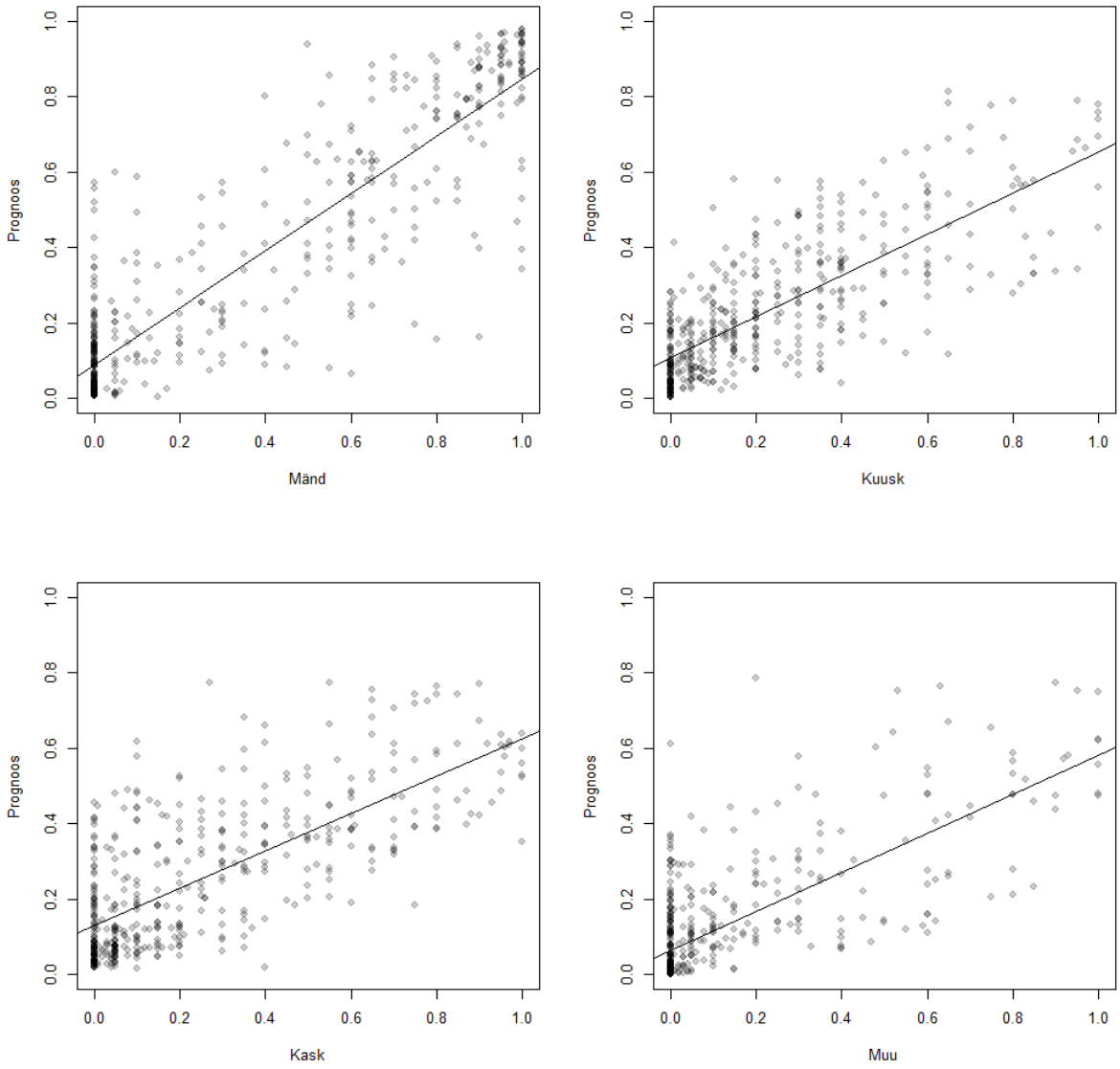
K-lähima naabri meetodi korral toimis pilt-haaval lähenemine paremini kui koondandmestiku pealt osakaalude prognoosimine. Regressioonipuu ja *bagging* korral on prognoosid praktiliselt samad ning meetodi edasise uurimise ja kasutamise entusiasmi vähendab ka asjaolu, et prognoosi parandamiseks välja visatud piltide arv on oluliselt suurem.



Joonis 28. Regressioonipuu ja bagging: *agregeeritud pilt-haaval prognooside paranemine valikuliselt kehvemaid pilte välja jättes.*

4.7 Juhumets

Regressioonipuu ja *baggingu* ning juhumetsa erinevus on kaunis väike: erinevus peitud selles, kui paljude tunnuste seast parimat hargnemist otsitakse. Näiteks Pythoni teegi *sklearn RandomForestRegressor* mudeli korral on vaikeväärtuseks *max_features = n_features* ehk maksimaalne arv tunnuseid, mille seast parimaid poolitusi otsitakse on kogu tunnuste arv – ehk tegu on regressioonipuu ja *bagging* meetodiga. Antud töös on parimaid poolitusi otsitud 1/3. *Bootstrap* valimite arv N = 1000. Visuaalne tulemus on näha joonisel 29. Tulemus nii visuaalselt kui ka numbriliselt väga sarnane regressioonipuu ja *bagging* meetodile joonisel 26.

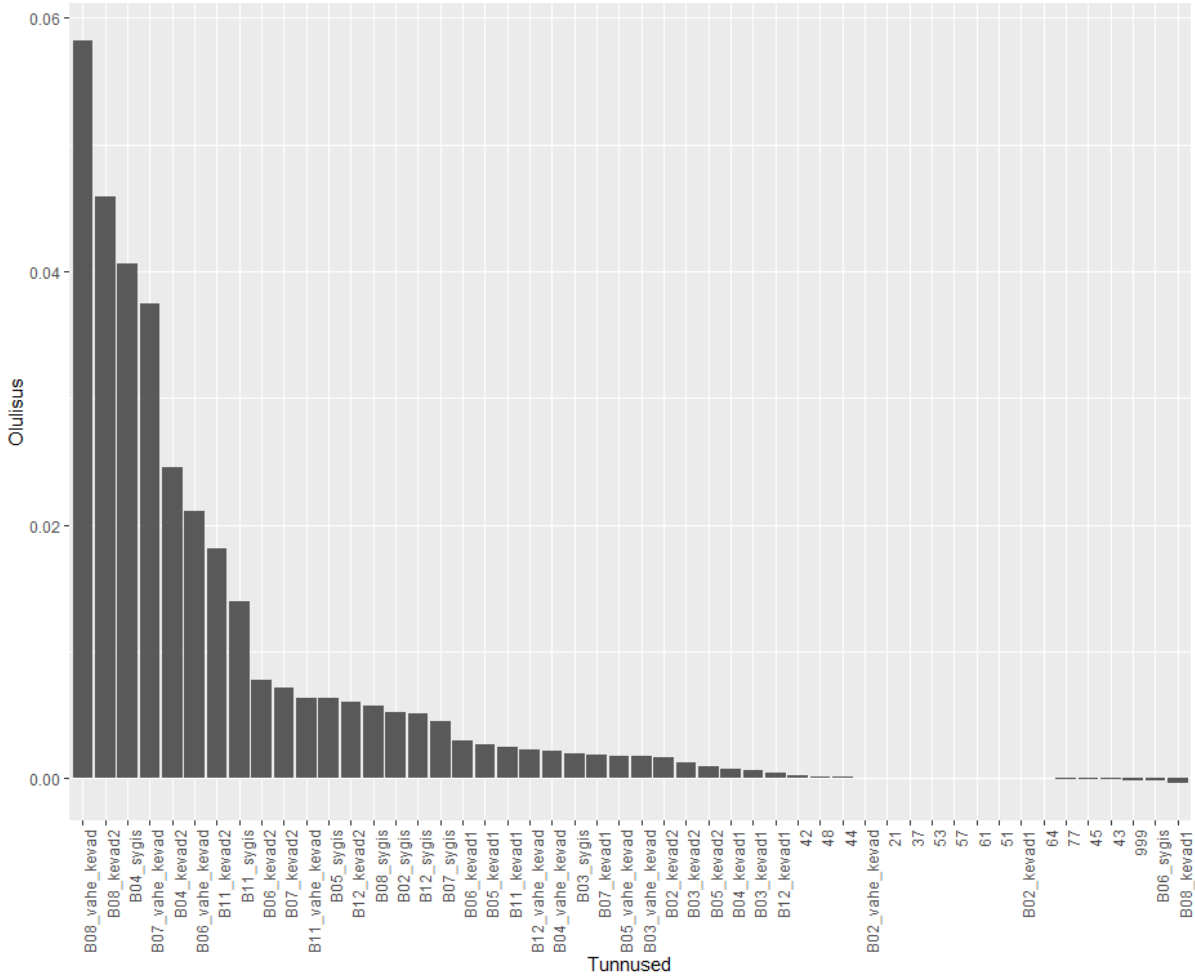


Joonis 29. Juhumets, N = 1000. RMSE = 0.1744.

Juhumetsa juures on huvitav vaadata, millised tulemused mudelis oluliseks on osutunud, seejuures on olulisuse mõõtmiseks mitu erinevat võimalust. Vaatame kahte võimalust:

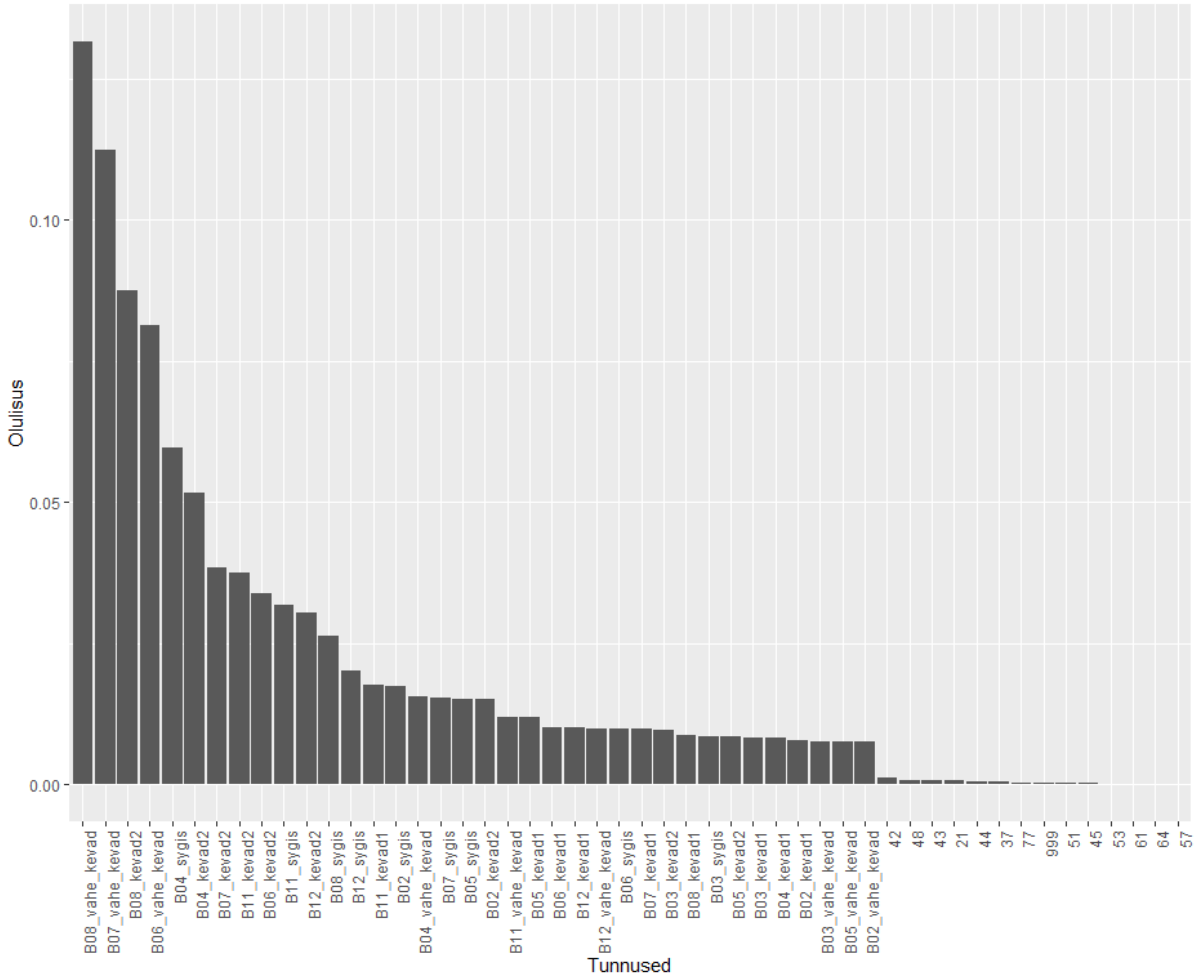
Üks võimalus on iga tunnuse *m* korral vaadata, kui palju selle tunnuse kasutamine kõikide puu hargnemiste peale kokku vähendab „ebapuhtust“. Ebapuhtusmeetodil saadakse vastava tunnuse *m* olulisus, liites kokku vastavad näitajad terves metsas. [20] Juhumetsa regressiooni korral on mudeli ebapuhtuse mõõdikuks prognooside hajuvus [21].

Teine võimalus tunnuste olulisuse leidmiseks on permutatsioonimeetod. Permutatsioonimeetodi korral järjestatakse iga tunnus juhuslikult ümber (permutatsioon) ja vaadatakse, kuidas see prognoosi täpsust mõjutab, täpsemalt: kui palju tunnuse ümberjärjestamine determinatsioonikordajat vähendab. [22]



Joonis 30. Juhumets: tunnuste olulisus permutatsioonimeetodil.

Permutatsioonimeetodil leitud kümne olulisima tunnuse seas (joonis 30) on kolm kevadist erinevust: koondandmestiku tekitamisega loodud fenoloogilised erinevused, mis puuduvad pilt-haaval lähemise korral, on ennast igati õigustanud. Kõik mullajanäitajad on kas väga väikese või sootuks negatiivse olulisusega. Permutatsioonimeetodi erinevalt ebapuhtusmeetodist suudab ära näidata ka need tunnused, mis on negatiivse olulisusega: ebapuhtusmeetodi korral on kõikide tunnuste olulisus positiivne, sest hargnemise eelduseks on ebapuhtuse ehk regressiooni korral jääkide ruutude summa vähenemine.

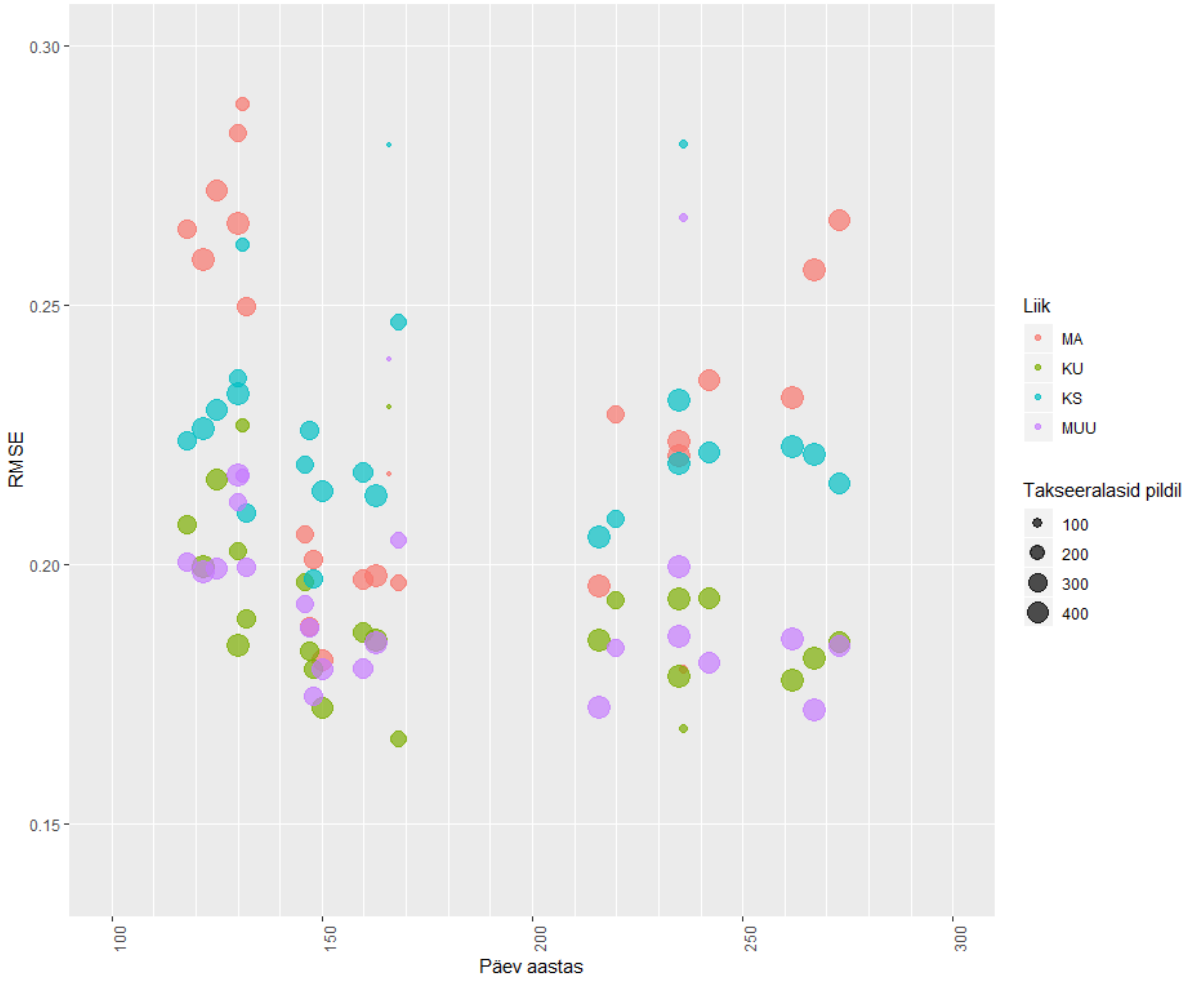


Joonis 31. Juhumets: tunnuste olulisus nende ebapuhtuse vähendamise alusel.

Ebapuhtusmeetodil saadud tunnuste olulisus on joonisel 31. Nagu näha, siis viis olulisimat tunnust on mõlema meetodi korral samad, ainult erinevas järjekorras, ning jällegi on mullatüüp pea olematu olulisusega. Mõlema meetodi korral osutus kõige olulisemaks tunnuseks kanali B8 ehk lähisinfrapunase kevadine erinevus.

4.8 Juhumets, pilt-haaval prognoosimine

Juhumetsa pilt-haaval lähenemise tulemused on väga sarnased regressioonipuu ja *bagging* tulemustele. Parimaks agregeerimismeetodiks osutus beta-funktsiooni tiheduse mood, RMSE = 0.1780, aga nagu ka regressioonipuu ja *bagging* puhul, siis epa-prognoosi RMSE on praktiliselt sama: RMSE = 0.1786. Kui vaadata vigasid piltide kaupa joonisel 32, siis muster on sarnane *K*-lähima naabri meetodi pilt-haaval prognooside vigadele (joonis 8): kevade alguses ja sügisel on männi prognoosid suurema veaga kui kevade lõpus ja suvel.



Joonis 32. Juhumets: pilt-haaval prognooside RMSE liikide kaupa sõltuvalt kuupäevast (mitmes päev aasta algusest).

Võttes regressioonipuu ja juhumetsa tulemused kokku, siis esiteks on tulemused kehvemad kui *K*-lähima naabri meetodite korral ja seda nii koondandmestiku pealt prognoosides kui pilt-haaval lähenemise korral. Kui KNN-i puhul annab pilt-haaval lähenemine parema tulemuse kui koondandmestiku kasutamine, siis regressioonipuu ja *baggingu* ja juhumetsa puhul on tulemus sama või pigem koondandmestiku korral parem. Regressioonipuu ja *bagging* on tulemuste poolest põhimõtteliselt eristamatud, küll aga on üksik regressioonipuu kahtlemata kehvem mudel puuliikide osakaalude modelleerimiseks kui regressioonipuu ja *bagging* või juhumets.

4.9 Multinomiaalne logistiline regressioon

Multinomiaalne logistiline regressioon on oma olemuselt puhas klassifitseerimismeetod, kuid teatud mööndustega on siiski võimalik puuliikide osakaalude vektoreid prognoosida: kui mingi puuliigi osakaal ületab teatud piiri, siis lugeda antud vaatlus vastava puuliigi klassi kuuluvaks. Töös on proovitud piire 0.5, 0.7 ja 0.8, neist kõige paremini töötas piir 0.8. Sellisel juhul kuulub tinglikku männi klassi 108 vaatlust, kuuse klassi 27 vaatlust, kase klassi 34 vaatlust ja muude liikide klassi 22 vaatlust. Mingisse tinglikku klassi kuuluvate vaatluste klassidesse kuulumise tõenäosused ehk puuliikide osakaalude prognoosid on saadud jäta-üks-välja ristvalideerimise teel. Ülejäänud 264 vaatlust on prognoositud ühe mudeliga, st mudeliga, mis on ehitatud klassidesse jagatud andmete peale.

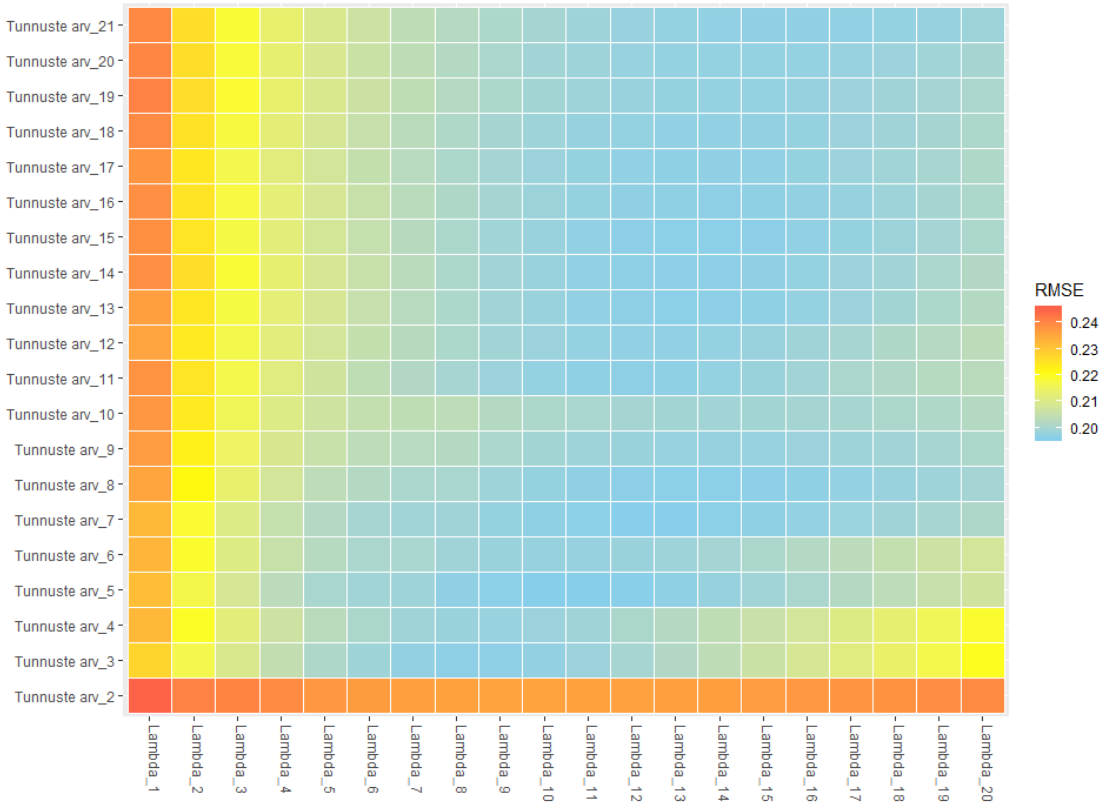
Multinomiaalne logistiline regressioon ei anna soovitud tulemust. Probleemiks võib olla eralduvus (*separation*): täielik eralduvus toimub, kui üks või mitu seletavat tunnust prognoosivad täpselt kõikide vaatluste klassid. Sellisel juhul on üks võimalik lahendus kasutada „karistatud" mudelit [23]. Eralduvus ei pruugi olla täielik, sellisel juhul prognoositakse täpselt mingisse klassi kuulumist või mittekuulumist [24].

R-is on karistatud multinomiaalse logistilise regressiooni mudel realiseeritu paketis *npmr* (*Nuclear Penalized Multinomial Regression*). Olgu c klasside arv ja p tunnuste arv ning β_0 ja $\mathbf{B} = (\beta_1, \dots, \beta_p)$ multinomiaalse logistilise regressiooni mudeli parameetrid nagu valemis (13). Karistatud mudeli parameetrid $(\beta_0^*, \mathbf{B}^*)$ leitakse kui

$$\underset{\beta_0 \in \mathbb{R}^c, \boldsymbol{B} \in \mathbb{R}^{p \times c}}{\arg\min} - \sum_{i=1}^{n} \log\left(\sum_{k=1}^{c} \frac{\exp(\beta_{j0} + \beta_{j1} x_1 + \cdots + \beta_{jp} x_p)}{\sum_{h=1}^{c} \exp(\beta_{h0} + \beta_{h1} x_1 + \cdots + \beta_{hp} x_p)} \mathbb{I}_{Y_i = k} \right) + \lambda \|\boldsymbol{B}\|_*, \tag{16}$$

kus λ on mingi positiivne arv ja $\|\mathbf{B}\|_*$ maatriksi $\mathbf{B}$ singulaarsete väärtuste summa [25].

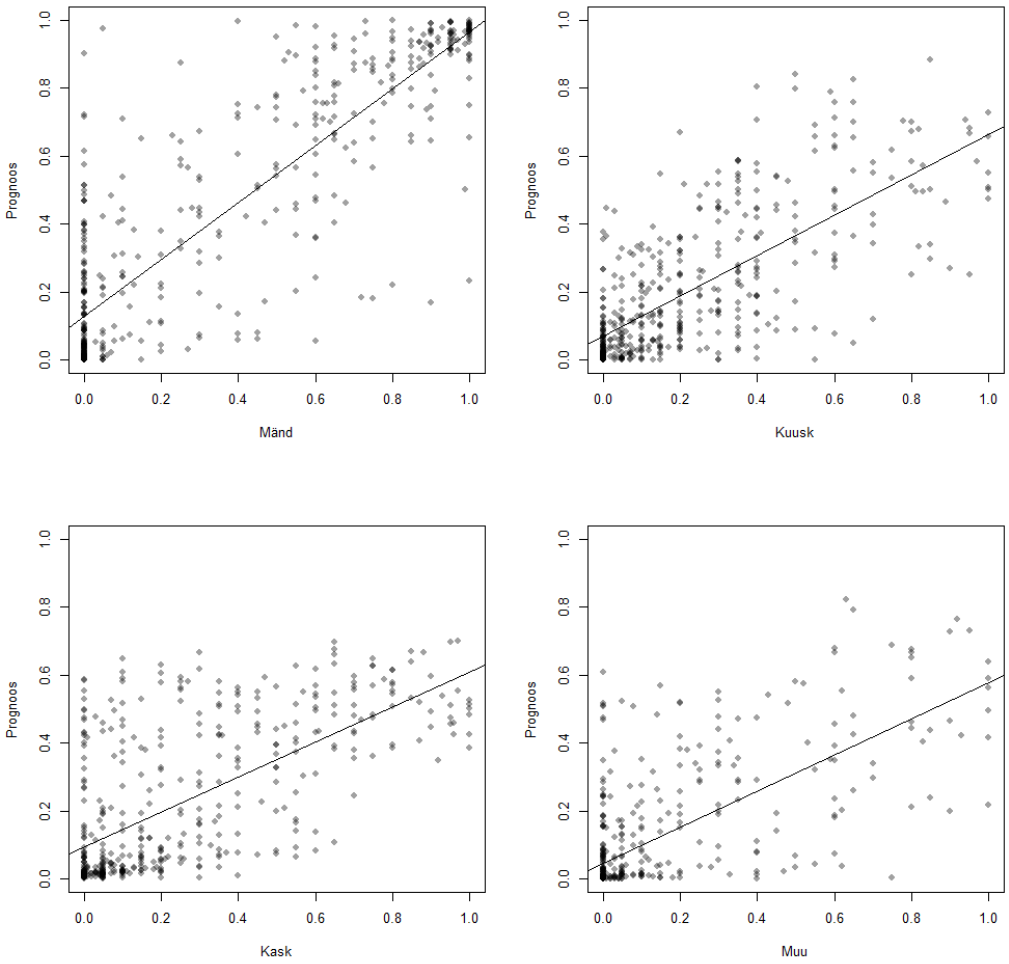
Lisaks karistusliikme kasutamisele on vähegi mõistlike prognooside saamise nimel arvesse võetud juhumetsa meetodi permutatsioonimeetodil leitud tunnuste olulisust. Joonisel 33 on näha, et karistatud multinomiaalse logistilise regressiooni korral tuleb parima tulemuse saavutamiseks kasutada viit kõige olulisemat tunnust ning karistusliikme väärtust $\lambda = 10$.



Joonis 33. Karistatud multinomiaalne regressioon. Karistusliikme väärtuse ja mudeli tunnuste arvu võreotsing. Tunnusteks on juhumetsa meetodi 20 olulisimat permutatsioonimeetodil leitud tunnust.

Jooniselt 34 on näha, et männi osakaalu prognoosimisel töötab multinomiaalne regressioon rahuldavalt, eriti ühelähedaste väärtuste prognoosimisel, samas paljud nullilähedaste väärtuste prognoosid on paljudel juhtudel suhteliselt kõrged. Teiste puuliikide osakaalude prognooside korral on pilt kehvem, samas on see nii kõikide meetodite korral.

Kokkuvõttes multinomiaalse logistilise regressiooni kasutamine end puuliikide osakaalude prognoosimisel ei õigusta ega vääri pikemat peatumist.



Joonis 34. Multinomiaalne logistiline regressioon: 5 olulisimat tunnust, karistusliige $\lambda = 10$. *RMSE = 0.1959.*

5. Puuliikide kaardi koostamine

Parimaks mitmemõõtmeliseks meetodiks puuliikide osakaalude prognoosimisel osutus pilt-haaval *K*-lähima naabri meetod, kus piltide prognoosid eraldivõetuna olid saadud naabrite arvu $K = 3$ korral. Prognooside agregeerimiseks on kasutatud Epanechnikovi tuumameetodil hinnatud tiheduse moodi.

Töös kasutatud meetoditega saadud prognooside põhjal saab koostada Eesti metsade liigilise koosseisu kaardi. Visuaalselt jäävad puuliikide osakaalude kaardil kõige paremini silma alad, kus ühe liigi osakaal on kõrge, seetõttu on alljärgnevalt toodud mõned parima mitmemõõtmelise meetodi täpsusnäitajad kõrge ühe puuliigi osakaaluga alade kohta. Tabelis 6 on toodud meetodi tundlikkus enimlevinud puuliigi tuvastamisel takseeraladel, kus enamuspuuliigi osakaal on vähemalt 75%. Kask tuvastatakse enimlevinud puuliigina kõikidel 35 juhul, männi korral ei tuvastata ühte juhtu 111 juhu kohta. Kuuse ja muude puuliikide tuvastamise täpsus on vastavalt 78.6% ja 77.3%.

Tabel 6. KNN: pilt-haaval lähenemine, K = 3, agregeeritud Epanechnikovi tuumameetodil hinnatud tiheduse moodi põhjal. Meetodi tundlikkus enimlevinud puuliigi tuvastamisel takseeraladel, kus enamuspuuliigi osakaal on vähemalt 75%. Keskmine täpsus 93.9%.

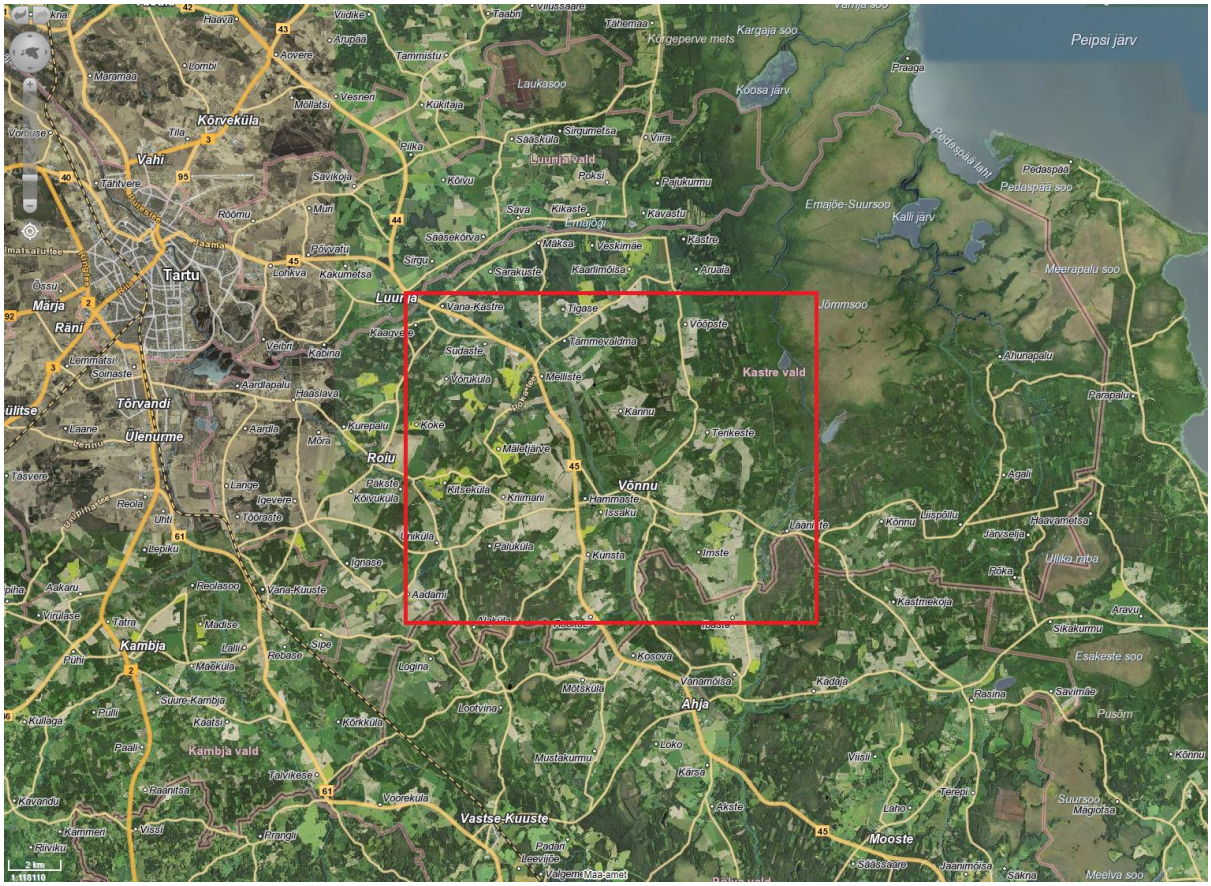
Suurim osakaalu prognoos	Takseerala enamuspuuliik, vähemalt 75%			
	Mänd	Kuusk	Kask	Muu
Mänd	110	4	0	0
Kuusk	1	22	0	0
Kask	0	2	35	5
Muu	0	0	0	17
Tundlikkus	99.1%	78.6%	100%	77.3%

Mudeli keskmine positiivne prognoosiväärtus enimlevinud puuliigi tuvastamisel, juhul kui prognoos on vähemalt 50%, on 88.8%. Tabelist 7 selgub, et kui männi prognoos on vähemalt 50%, siis 97.9% juhtudest on tegemist enimlevinud puuliigiga; kuuse puhul on 76.4% juhtudel tegu enimlevinud liigiga, kase puhul 86.3% juhtudel ning muude liikide puhul 65.7% juhtudel.

Tabel 7. KNN: pilt-haaval lähenemine, K = 3, agregeeritud Epanechnikovi tuumameetodil hinnatud tiheduse moodi põhjal. Enimlevinud puuliigi positiivne prognoosiväärtus takseeraladel, kus prognoosi järgi on üks puuliik enamuses (vähemalt 50%). Keskmine positiivne prognoosiväärtus 88.8%.

Osakaalu prognoos, vähemalt 50%	Takseerala enimlevinud puuliik				Positiivne prognoosiväärtus
	Mänd	Kuusk	Kask	Muu	
Mänd	184	2	2	0	97.9%
Kuusk	7	42	6	0	76.4%
Kask	3	6	69	2	86.3%
Muu	1	0	11	23	65.7%

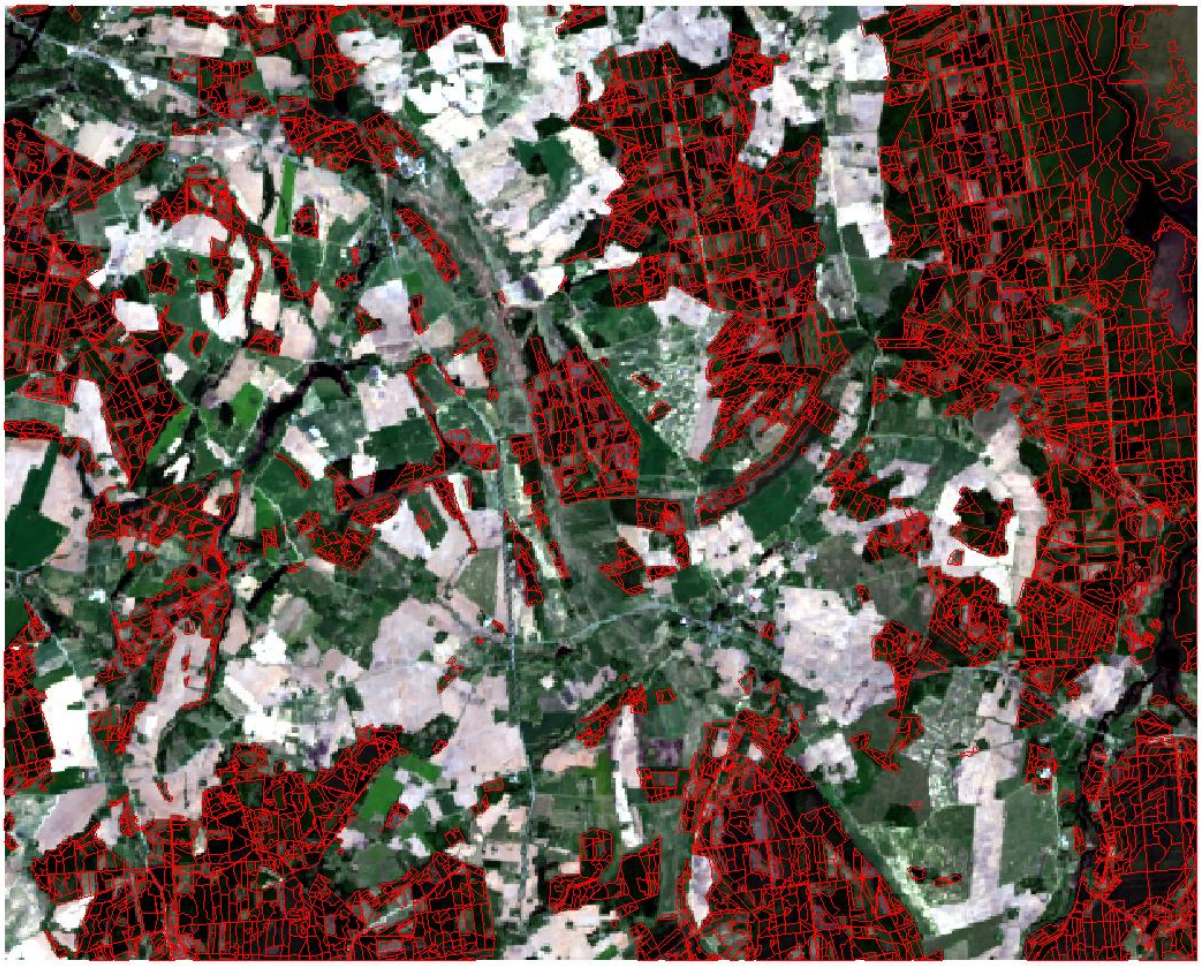
Et loodav kaart oleks paremini hoomatav, on välja valitud joonisel 35 kujutatud ristkülik ligikaudsete küljepikkustega 15 km ja 13 km. Valiku peamiseks aluseks on satelliidipiltide olemasolu antud ala kohta: nimelt isegi kui kõrvale jätta võimalikud pilvkattest tingitud häiringud, siis teatud osa Eestist on üles pildistatud mitme erineva ülelennutrajektoori korral, teatud osa aga ühe. Valitud ala jääb erinevatel trajektooridel tehtavate piltide ülekattuvale alale ja seetõttu on pilte rohkem. Kokku on kasutada 9 Sentineli ja 6 Landsati pilti. Ala asub valdavas osas Tartu maakonnas, Tartu linnast kagu suunas: ristküliku loodenurgas asub Luunja, keskosast natuke kagu suunas Võnnu alevik, kirdenurgas algab Emajõe-Suursoo ja selle ääreala soised metsad.



Joonis 35. Puuliikide osakaalude prognooskaardi koostamiseks valitud ristkülik Tartust kagus. Kuvapilt Maa-ameti kaardirakendusest [3].

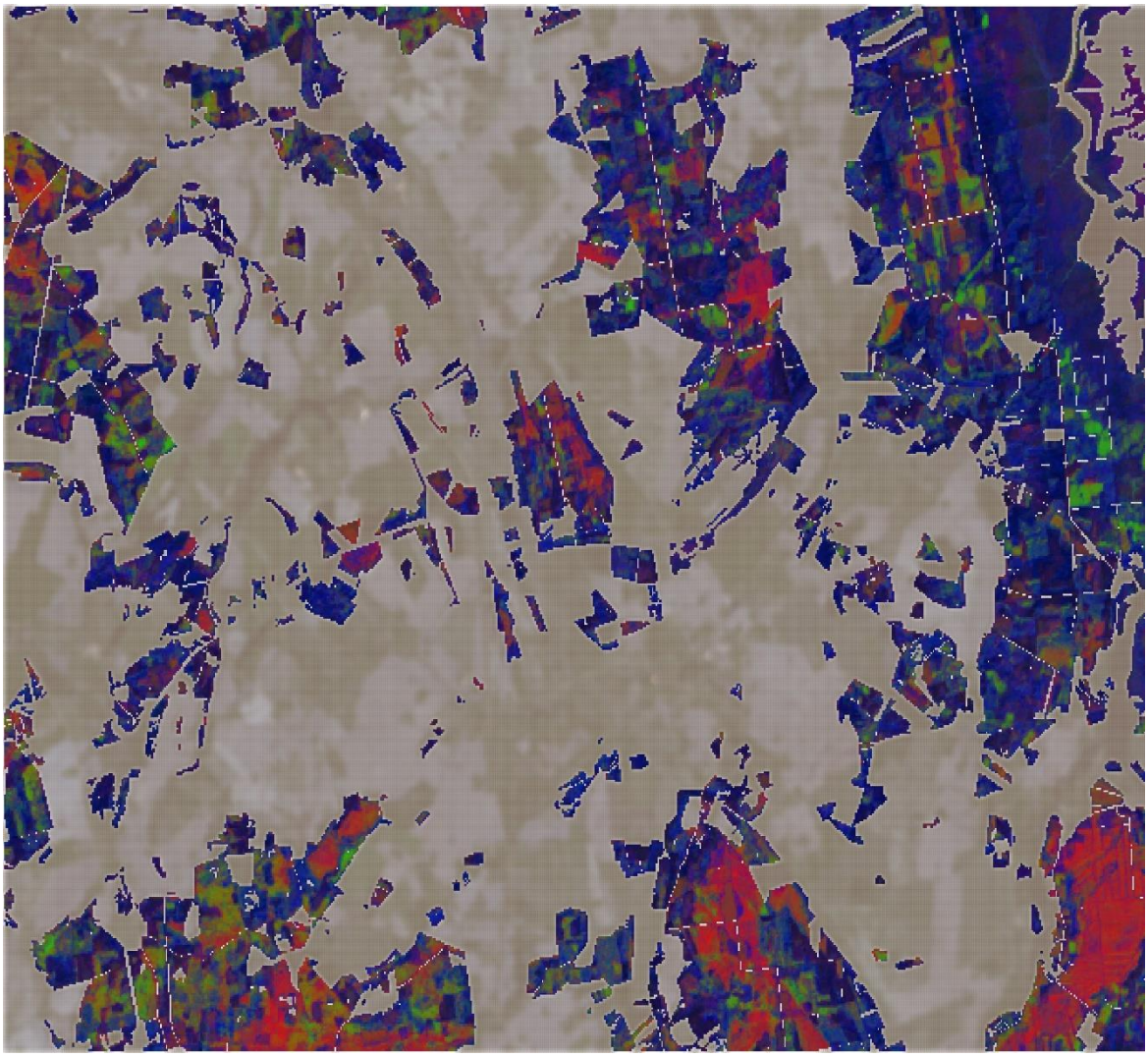
Kaardi koostamiseks on valitud ülalkirjeldatud parima mitmemõõtmeline meetod. Agregeeritud pilt-haaval prognooside arvutamiseks on kasutada 24 pilti, valitud ala kaardi koostamiseks on kasutada 15 pilti. Ka 15 pildi kasutamise korral osutus optimaalseks 3 naabri kasutamine, kuigi võiks eeldada, et mida vähem pilte, seda täpsemad peavad olema üksikute piltide prognoosid (üksikute piltide suurim täpsus saavutatakse enamasti 16–20 naabri korral, joonis 9). Kasutatud on kõiki olemasolevaid pilte, st prognooside täpsuse parandamise eesmärgil pole ühtegi pilti kõrvale jäetud.

Kuna mudel on koostatud metsade liigilise koosseisu määramiseks, siis tehisobjektid ja muu maastik tuleb metsast eraldada. Selleks on kasutatud metsaregistris olevaid eraldisi [26]. Paraku, nagu võib ka jooniselt 36 aimata, ei ole kogu mets eraldistena registrisse kantud.



Joonis 36. Metsaregistri eraldised valitud alal. Taustaks satelliidipilt.

Kaardi kujul prognoosid metsaeraldistele on joonisel 37. Puuliikide osakaalu kaart on RGB skaalal, kus punasele vastab mänd, rohelisele kuusk, sinisele kask ja nende puudumisele ehk mustale muud puuliigid. Ala kagunurgas, Läänistest lõunas, on näha suur kõrge männi osakaaluga metsamassiiv. Kõrged männi osakaalu väärtused on kõikide meetoditega üpris täpselt prognoositud, seega suurte punaste alade osas ei tohiks olla kahtlust, et tegemist on männimetsaga. Pildi kirdenurgas, Ahja jõe ääres, on suured kase enamusega alad, samas on seal ka hästi eristuvaid kõrge kuuse osakaaluga metsakvartaleid. Üldiselt on aga pilt üsna kirju: üheks põhjuseks võib olla asjaolu, et kuna kaardil on kõik metsakinnistud, siis on puuliikide osakaalude prognoos arvutatud ka raiesmikele.

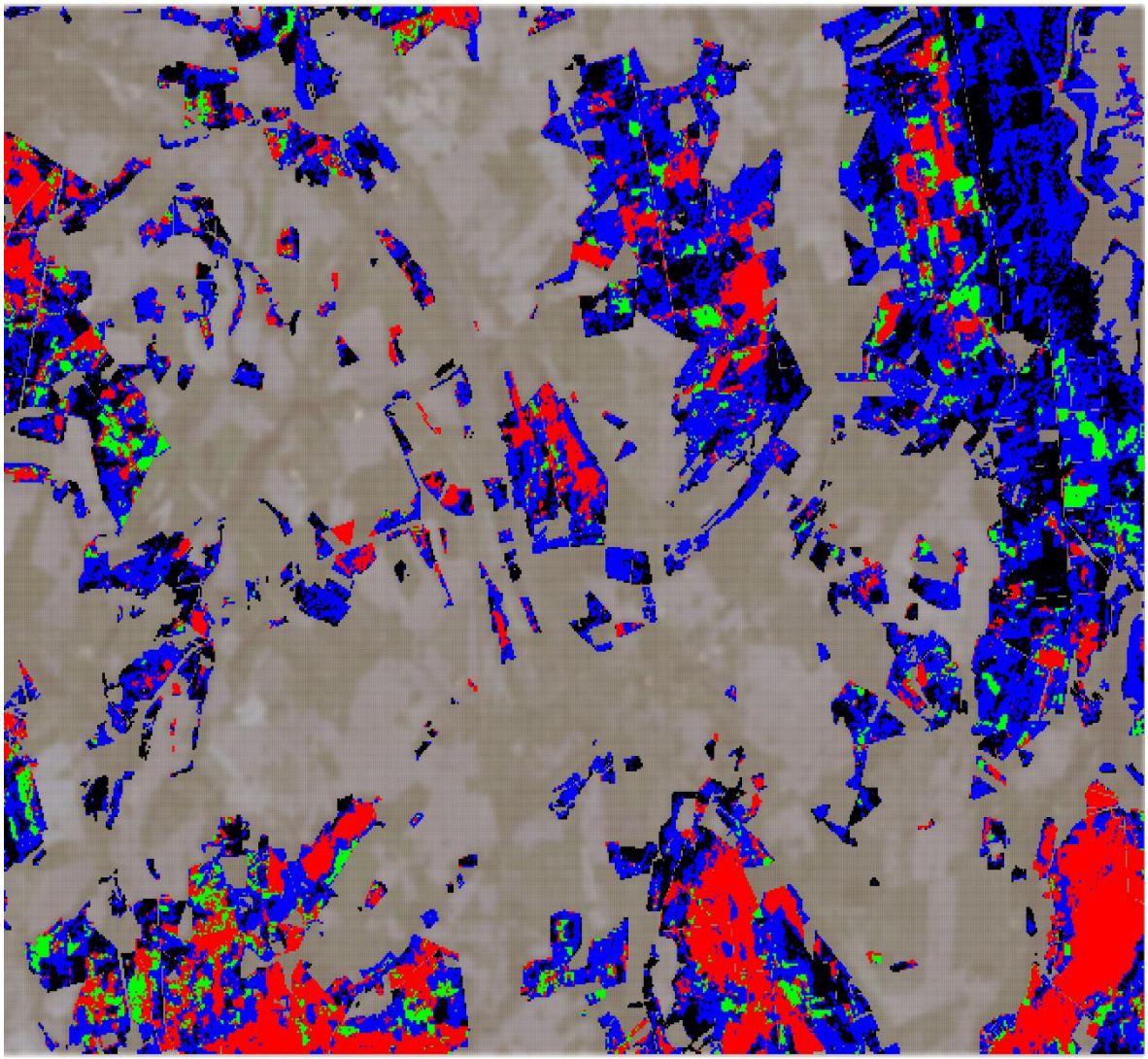


Joonis 37. Puuliikide osakaalude vektorite prognoosid metsaeraldistel RGB skaalal. Punane – mänd; roheline – kuusk; sinine kask; must – muu.

Pistelise kontrolli tulemusena võib öelda, et raiesmikud on pigem *sinised* ehk prognoositud kõrge kase osakaaluga metsaks. Kuna mudeli koostamisel on kasutatud vaid takseeralasid, mille korral tüvemahuhinnang hektari kohta on enam kui 100 m^3, siis peaks ka kaardilt kõrvaldama kõik väiksema tüvemahuhinnanguga pikslid. Tervet Eestit katvate aerolidari andmete põhjal on võimalik metsa tagavara prognoosida [27] ja seega välja selekteerida vaid piisava tüvemahuhinnanguga pikslid, aga see jääb antud töö raamidest välja.

Valitud ala puuliikide keskmised osakaalud on mänd: 23.1%, kuusk: 16.9%, kask: 33.9% ja muud 26.1%. Võrdluseks, Eesti metsamaa pindala jaotus peapuuliigi järgi on mänd 31.4%, kuusk 18.8%, kask 29.5% ja muud liigid 20.3% [28].

Joonisel 38 on kujutatud puuliikide osakaalude prognooside asemel enimlevinud puuliigid. Liikidest domineerivad mänd ja kask, samas erinevalt osakaalude vektorite kaardist on enimlevinu liikide kaardil tuvastatav ka „muud“ liigid (must), mille alla kuuluvad näiteks hall lepp ja must lepp.

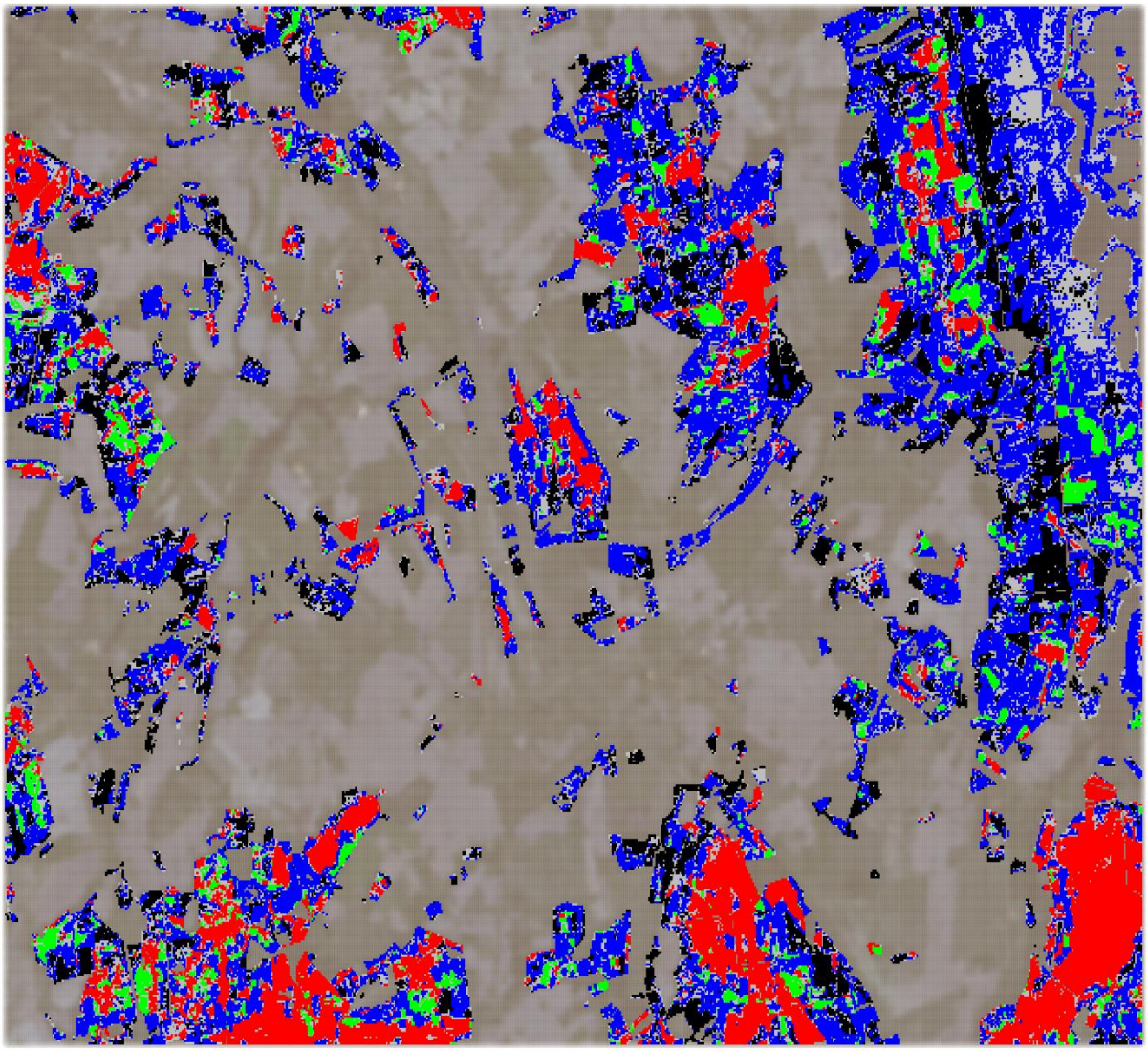


Joonis 38. Enimlevinud puuliigi prognoosid metsaeraldistel RGB skaalal. Punane – mänd; roheline – kuusk; sinine kask; must – muu.

Erinevate meetoditega tulevad prognoosid mõnevõrra erinevad. Joonisel 39 on kujutatud parima mitmemõõtmelise mudeli prognooside kattuvust paremuselt teise pilt-haaval mudeli prognoosidega. Paremuselt teine mudel on regressioonipuu ja *bagging*, RMSE = 0.1758, parima mudeli korral RMSE = 0.1578. Tabelist 8 on näha, et tulemused kattuvad 79.1% ulatuses. Liikidest kattuvad kõige enam männi prognoosid: 85.1% pikslitest, kus KNN meetodiga on prognoositud enimlevinud puuliigiks mänd, on sama prognoosi saanud ka

regressioonipuu korral. Teistpidine prognooside kattuvus on 81.4%. Kase prognoosid kattuvad keskmiselt umbes 80% ulatuses, kuuse prognoosid umbes 70% ulatuses ja muude liikide prognoosid umbes 75% ulatuses, kusjuures muude liikide prognoos on erinevuse korral mõlematpidi peaaegu alati kask.

Regressioonipuu prognoosid kaardi kujul on saadaval töö lisas.



Joonis 39. Regressioonipuu ja bagging *prognooside ja K-lähima naabri mudeli prognooside enamuspuuliigi kattuvus. Kattuvus 79.1%.*

Tabel 8. KNN ja regressioonipuu meetodil saadud enimlevinud puuliigi piksli kaupa prognooside kokkulangevus.

		Regressioonipuu prognoos				
		Mänd	Kuusk	Kask	Muu	
KNN		25426	7593	54087	29539	
Mänd	24321	20696	983	2629	13	85.1%
Kuusk	10818	2108	6391	2286	33	59.1%
Kask	57710	2393	219	45364	9734	78.6%
Muu	23796	229	0	3808	19759	83.0%
		81.4%	81.7%	83.9%	66.9%	79.1%

6. Kokkuvõte

Mitmemõõtmelistest meetoditest töötab takseeraladele vastavate satelliidiandmete ja mullainfo põhjal puuliikide osakaalude prognoosimisel kõige paremini *K*-lähima naabri meetod, mille korral on puuliikide osakaalude prognooside ruutkeskmine viga 0.1578. Mitmemõõtmeline *K*-lähima naabri meetod jääb siiski alla analoogilisele ühemõõtmelisele meetodile: liikide osakaalude eraldi prognoosimise ja pärastise saadud prognoosivektorite normeerimise korral on prognooside ruutkeskmine viga 0.1437.

Regressioonipuu ja *bagging* meetod ning juhumetsa meetod töötavad mõnevõrra kehvemini, kusjuures omavahel käituvad nad väga sarnaselt. Üksik regressioonipuu ja multinomiaalne logistiline regressioon töötavad proovitud meetoditest kõige kehvemini.

Pilt-haaval lähenemine annab töös kasutatud 24 satelliidipildi korral paremaid tulemusi, kui nende samade piltide pealt loodud koondandmestiku pealt prognoosimine. Pilt-haaval prognooside agregeerimiseks välja töötatud Epanechnikovi tuumameetodiga hinnatud tiheduse mood ning beta-funktsiooni tiheduse mood on ennast õigustanud, töötades paljudel juhtudel paremini kui aritmeetiline keskmine.

Piltide arvu kasvades prognoosid paranevad, kusjuures parimateks osutunud KNN meetodite – nii ühe- kui ka mitmemõõtmelise – puhul on paranemine kõige jõudsam, kui agregeerimisel kasutada Epanechnikovi tuumameetodiga hinnatud tiheduse moodi. Oskuslik üksikute piltide prognooside agregeerimine ning üha täienev pildibaas võivad anda võimaluse loobuda pilvemaskide manuaalsest koostamisest ning rakendada automaatseid protseduure agregeerimisel kasutatavate piltide valikuks. Juba 24 pildi korral ilmnevad selged aastaajalised mustrid üksikute piltide prognooside täpsuses, ent seos piltide agregeerimisest kõrvale jätmise ja üldise prognoositäpsuse vahel ei ole selge.

Täpsemate prognooside saamiseks peab kindlasti olema kasutada rohkem takseeralasid. Eriti on puudu takseeralasid, kus vähemlevinud puuliikidel oleks kõrge osakaal. Selliste alade lisandumine vähendab tõenäoliselt vähemlevinud puuliikide kõrgete osakaalude süstemaatilist alahindamist ning võimaldab prognoosida ka liike, mis antud töös kuuluvad asjakohaste andmete vähesuse tõttu muude puuliikide sekka. Takseeralade lisandumine koos olemasolevate takseeralade koordinaatide täpsustamisega ja pidevalt uueneva satelliidiinfoga annab võimaluse koostada Eesti metsade liigilise koosseisu kaart, millest on kasu nii teadlastele, metsamajandussektorile kui ka tavainimesele.

7. Kasutatud kirjandus

[1] Mis on statistiline metsainventuur ehk SMI. Kasutatud 13.05.2019, https://www.keskkonnaagentuur.ee/et/uudised/mis-statistiline-metsainventuur-ehk-smi

[2] Landsat 8 & Sentinel-2 Band Comparison. Kasutatud 13.05.2019, http://www.gisagmaps.com/landsat-8–sentinel-2-bands/

[3] Maa-ameti kaardirakendus. Kasutatud 13.05.2019, https://xgis.maaamet.ee/maps/XGis.

[4] M. Lang, Kaugseire meetoditega metsaressursi arvestamine: Tartu Ülikooli Tartu Observatooriumi tehtud tööde ülevaade, Reach-U, 2019.

[5] M. Lang, M. Kaha, D. Laarmann ja A. Sims, Construction of tree species composition map of Estonia using multispectral satellite images, soil map and a random forest algorithm, *Forestry Studies,* kd. 68, nr 1, lk. 5–24, 2018.

[6] H. Franco-Lopeza, A. R. Ek ja M. E. Bauer, „Estimation and mapping of forest stand density, volume, and cover type using the k-nearest neighbors method, *Remote Sensing of Environment,* 77, lk. 251–274, 2001.

[7] M. Lang, T. Arumäe, T. Lükk ja A. Sims, Estimation of standing wood volume and species composition in managed nemoral multi-layer mixed forests by using nearest neighbour classifier, multispectral satellite images and airborne lidar data, *Forestry Studies,* 61, lk. 47–68, 2014.

[8] R. E. McRoberts, „Estimating forest attribute parameters for small areas using nearest,“ *Forest Ecology and Management,* 272, lk. 3–12, 2012.

[9] G. James, D. Witten, T. Hastie ja R. Tibshirani, An introduction to statistical learning : with applications in R, New York: Springer, 2013.

[10] K. Hechenbichler ja K. Schliep, Weighted k-Nearest-Neighbor Techniques and Ordinal Classification, 2004. Kasutatud 13.05.2019, https://epub.ub.uni-muenchen.de/1769/1/paper_399.pdf

[11] S. B. Imandoust ja M. Bolandraftar, Application of K-Nearest Neighbor (KNN) Approach for Predicting Economic Events: Theoretical Background, *Journal of Engineering Research and Application,* 3(5), 2013.

[12] M. Kuhn ja K. Johnson, Applied Predictive Modeling, New York: Springer, 2013.

[13] L. Breiman, Random Forests, *Machine Learning,* 45(1), lk. 5–32, 2001.

[14] A. Agresti, An Introduction to Categorical Data Analysis, 3rd Edition, Wiley, 2019.

[15] R. E. McRoberts, M. D. Nelson ja D. G. Wendt, Stratified estimation of forest area using satellite imagery, *Remote Sensing of Environment,* 82, lk. 457–468, 2002.

[16] C. E. B. Owen, Parameter Estimation for the Beta Distribution, 2008. Kasutatud 13.05.2019, https://scholarsarchive.byu.edu/cgi/viewcontent.cgi?article=2613&context=etd

[17] D. Robinson, Understanding the beta distribution, 2014. Kasutatud 13.05.2019, http://varianceexplained.org/statistics/beta_distribution_and_baseball/

[18] M. Smithsen ja J. Verkuilen, A better lemon-squeezer? Maximum likelihood regression with beta-distributed dependent variables, *Psychological Methods,* 11(1), lk. 54–71, 2006.

[19] L. Wasserman, All of Nonparametric Statistics, New York: Springer, 2006.

[20] L. Breiman ja A. Cutler, Random Forests. Kasutatud 13.05.2019, https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm#varimp

[21] G. Louppe, L. Wehenke, A. S. ja P. Geurts, Understanding variable importances, *Advances in neural information processing systems*, 26, 2013.

[22] T. Parr, K. Turgutlu, C. Csiszar ja J. Howard, Beware Default Random Forest Importances. Kasutatud 13.05.2019, https://explained.ai/rf-importance/

[23] S. J. Cook, J. Niehaus ja S. Zuhlke, A warning on separation in multinomial logistic models, *Research and Politics,* 1(5), 2018.

[24] C. Rainey, Dealing with Separation in Logistic Regression Models, *Political Analysis,* 24, lk. 339–355, 2016.

[25] S. Powers, T. Hastie ja Robert, Nuclear penalized multinomial regression. Kasutatud 13.05.2019, https://web.stanford.edu/~hastie/Papers/Powers_baseball.pdf

[26] Metsaportaal. Kasutatud 13.05.2019, https://register.metsad.ee/

[27] T. Arumäe ja M. Lang, „Aerolidarilt puistu tüvemahu hindamise mudelid ning võrdlus takseeritud tagavaraga,“ *Metsanduslikud Uurimused,* 64, lk. 5–16, 2016.

[28] Keskkonnaministeeriumi metsastatistika. Kasutatud 13.05.2019, https://www.envir.ee/et/metsastatistika

8. Lisad

Prognooside ruutkeskmised vead (RMSE)

Tabel 9. Ülevaade meetodite prognooside ruutkeskmisest veast.

		Pilt-haaval hinnatud / pärast kehvemate piltide kõrvaldamist		
	Koondandmestik	Aritmeetiline keskmine	Beta-prognoos	Epa-prognoos
KNN, mitmemõõtmeline	0.1616	0.1655 / 0.1598	0.1702 / 0.1628	0.1578 / 0.1534
KNN, ühemõõtmeline	0.1534	0.1584 / 0.1492	0.1523 / 0.1475	0.1437 / 0.1383
Regressioonipuu / tunnuste eelvaliku korral	0.1979 / 0.1860			
Regressioonipuu ja *bagging*	0.1741	0.1824 / 0.1753	0.1758 / 0.1688	0.1762 / 0.1718
Juhumets	0.1744	0.1860 / 0.1775	0.1780 / 0.1713	0.1779 / 0.1767
Multinomiaalne logistiline regressioon	0.1959	0.1855	0.2192	0.2067

Näide koodist: K-lähima naabri meetod, Pilt-haaval lähenemine

```
require(FNN); require(dplyr); require(EnvStats)

#Andmete sisse lugemine: Skaleeritud Sentinel

data0 = read.csv("sentinel455sc.csv")
data0 = data0[,c(2:13,26:38)]
vars.sent = names(data0)[c(4:25)] #kirjeldavad tunnused
kps = unique(data0$kp)

#Andmete sisselugemine: takseerinfo
load(file = "taks_info.RData")

#vajalikud funktsioonid:
```

```
#funktsioon parimate tunnuste leidmiseks
fun_bestvars = function(k1, k2, sidxx, kernel, varsw = vars){
  list_return = vector("list", length = 2)
  var_return = vector("list", length = 1+k2-k1)
  rss_return = vector("list", length = 1+k2-k1)
  for(k in k1:k2){
    vars0 = vars
    vars1 = c()
    mx = 6969
    vahe = 696
    while(vahe > 0){
      mx0 = mx
      vars00 = c()
      rmse = c()
      for(j in 1:length(vars0)){
        vars00 = c(vars1,vars0[j])
        dex = data[,c("aproovitykk_id",vars00)]
        dex = dex[dex$aproovitykk_id %in% sidxx,]
        dex$cl = "cl"
        pred_puu = fun_agre_kernel(dex, data_puud, k = k, sid = sidxx,
kernel = kernel)
        rsdls = pred_puu[,1:4] - puud_true
        rmse[j] = sqrt((sum(rsdls**2))/(length(sidxx)*4))
      }
      mx0 = min(rmse)
      varmin = vars0[which.min(rmse)]
      vars0 = vars0[!(vars0 %in% varmin)]
      vars1 = c(vars1,varmin)
      vahe = mx-mx0
      mx = mx0
    }
    var_return[[1+k-k1]] =  vars1
    rss_return[[1+k-k1]] =  mx
  }
  list_return[[1]] = var_return;list_return[[2]] = rss_return
  list_return
}

#Funktsioon lähimate naabrite leidmiseks, nende kauguste teisendamine
kaaludeks
#ning kaalude ja naabrite puuliikide proportsioonide agregeerimine
prognoosiks
fun_agre_kernel = function(data, data_puud, k, sid, kernel = epa)
{
  sid00 = data$aproovitykk_id
  kk = k+1 #Epanechnikovi kaalud tahab ühe võrra pikemat kauguste vektorit
  dists = knn.cv(train = data[,2:(dim(data)[2]-1)], cl = data$cl, k = kk)
  dist1 = attr(dists,"nn.dist") #lähimate naabirte kaugused
  index1 = attr(dists,"nn.index") #neile vastavad indeksid
  props = apply(dist1, 1, kernel) #kauguste teisendamine kaaludeks
  props = t(props)
  indxprops = cbind(index1, props)
  tbr = t(apply(indxprops, 1, agre, data_puud))
  tbr = tbr / rowSums(tbr)
  tbr = data.frame(tbr); tbr$aproovitykk_id = sid00
  tbr
}
```

```
#Epanechnikovi kaalud:
epa = function(vec){
  if(any(is.na(vec))){
    return(NA);
  }
  props1 = c()
  if(vec[1] == 0){
    props1[1] = 1; props1[2:length(vec)] = 0
    return(props1)
  }
  props = 3/4*(1-(vec / vec[length(vec)])**2)
  props1 = props/sum(props)
  if(sum(props) == 0) {
    props1[1] = 1; props1[2:length(vec)] = 0
  }
  props1
}

#Kaalude ja naabrite puuliikide proportsioonide agregeerimine prognoosiks
agre = function(arg,data_puud){
  kk = length(arg) / 2
  indx = arg[1:kk]; props = arg[(kk+1):(2*kk)]
  colsums = colSums(data_puud[indx,]*props)
  colsums
}

#Funktsioon kaalude optimeerimiseks
fun_opti = function(w, k, vars, data, sidxx, kernel){
  dex = data[,c("aproovitykk_id",vars)]
  dex = dex[dex$aproovitykk_id %in% sidxx,]
  dex[,-1] = t((t(as.matrix(dex[,-1])))*w)
  dex$cl = "cl"
  pred_puu = fun_agre_kernel(dex, data_puud, k = k, sid = sidxx, kernel)
  rsdls = pred_puu[,1:4] - puud_true
  rmse = sqrt((sum(rsdls**2))/(length(sidxx)*4))
  rmse
}

#Funktsioonid pilt-haaval prognooside agregeerimiseks:
#Beta-funktsiooni tiheduse moodi põhjal:
bets_fun = function(vec, method = "mle"){
  if(any(is.na(vec))){
    stop("NA")
  }
  if(length(unique(vec)) < 2){
    max.d = mean(vec)
  }
  else{
    eb = ebeta(vec, method = method)
    a = eb$parameters[1]; b = eb$parameters[2]
    opt = optimize(interval = c(0,1), dbeta, shape1 = a, shape2 = b,
maximum = T)
    max.d = opt$`maximum`
    if((a < 1 & b < 1)){
      max.d = mean(vec)
    }
  }
  max.d
}
```

```
#Epanechnikovi tuumafunksiooniga hinnatud tiheduse moodi põhjal:
epa.kernel = function(vec, kernel = "epanechnikov"){
  if(length(vec) < 2){max = vec}
  else{
    d = density(vec, kernel = "epanechnikov", from = 0, to = 1)
    max = d$x[which.max(d$y)]
  }
  max
}

#Agregeeritud pilt-haaval prognoosid. RMSE leidmine sõltuvalt naabrite
arvust ja agregeerimismeetodist
tvmaht = 2 # 2 - prognoosid puuliikide proportsioonide pealt; 1 -
tüvemahtude pealt
RMSE.beta = c()
RMSE.epa = c()
RMSE.mean = c()
RMSE.beta.opt = c()
RMSE.epa.opt = c()
RMSE.mean.opt = c()
#Naabrite arv K = 1...20
for(k in 1:20){
  df.kp = data.frame("MA" = c(), "KU" = c(), "KS" = c(), "XX" = c(),
"aproovitykk_id" = c())
  df.kp.opt = data.frame("MA" = c(), "KU" = c(), "KS" = c(), "XX" = c(),
"aproovitykk_id" = c())
  #15 Sentineli pilti; analoogiline Landsati piltide korral
  for(i in 1:15){
    vars = vars.sent
    kp = kps[i]
    data = na.omit(data0[data0$kp == kp,])
    sid0 = data$aproovitykk_id
    #takseerinfost vajalikud read ja lahtrid
    data_puud = taks.info[taks.info$aproovitykk_id %in% sid0,]
    puud_true =  data_puud[,7:10]
    data_puud = data_puud[,(4*tvmaht-1):(4*tvmaht+2)] #vastavalt kas
proportsioonid või tüvemahud
    #parimate tunnuste leidmine
    bestvars.kp.k = fun_bestvars(k, k, sidxx = sid0, kernel = epa)
    best_vars = unlist(bestvars.kp.k[[1]]) #leitud parimad tunnused
    dex = data[,c("aproovitykk_id",best_vars)]
    dex = dex[dex$aproovitykk_id %in% sid0,]
    dex$cl = "cl"
    result.kp.k = fun_agre_kernel(dex, data_puud, k = k, sid = sid0, kernel
= epa)
    df.kp = rbind(df.kp, result.kp.k)
    #leitud tunnuste optimeerimine
    opt = optim(par = rep(1, length(best_vars)), fn = fun_opti, k = k, vars
= best_vars, data = data, sidxx = sid0,method = "BFGS", kernel = epa)
    wgt = opt$par #leitud kaalud
    dex = data[,c("aproovitykk_id",best_vars)]
    dex = dex[dex$aproovitykk_id %in% sid0,]
    dex[,-1] = t((t(as.matrix(dex[,-1])))*wgt)
    dex$cl = "cl"
    result.kp.opt = fun_agre_kernel(dex, data_puud, k = k, sid = sid0,
kernel = epa)
    df.kp.opt = rbind(df.kp.opt, result.kp.opt)
  }
```

```
  df.kp0 = df.kp
  df.kp0[,1:4] = df.kp0[,1:4] / rowSums(df.kp0[,1:4])
  #Pilt-haaval prognooside agregeerimine
  df.epa  = df.kp0 %>% group_by(aproovitykk_id) %>%
summarise_all(funs(epa.kernel))
  df.mean = df.kp0 %>% group_by(aproovitykk_id) %>%
summarise_all(funs(mean))
  df.beta = df.kp0 %>% group_by(aproovitykk_id) %>%
summarise_all(funs(bets_fun))
  #Prognooside vektorite normeerimine
  df.beta[,2:5] = df.beta[,2:5] / rowSums(df.beta[,2:5])
  df.epa[,2:5] = df.epa[,2:5] / rowSums(df.epa[,2:5])
  df.mean[,2:5] = df.mean[,2:5] / rowSums(df.mean[,2:5])
  #Prognooside sidumine takseerandmetega
  dp.b = merge(df.beta, taks.info, all.x = T, by = "aproovitykk_id")
  dp.e = merge(df.epa, taks.info, all.x = T, by = "aproovitykk_id")
  dp.m = merge(df.mean, taks.info, all.x = T, by = "aproovitykk_id")
  #RMSE arvutamine
  RMSE.beta[k] = sqrt(sum((dp.b[,11:14]-dp.b[,2:5])**2)/(dim(dp.b)[1]*4))
  RMSE.epa[k] = sqrt(sum((dp.e[,11:14]-dp.e[,2:5])**2)/(dim(dp.e)[1]*4))
  RMSE.mean[k] = sqrt(sum((dp.m[,11:14]-dp.m[,2:5])**2)/(dim(dp.m)[1]*4))

  #optimeeritud tulemuste pealt samad arvutused:
  df.kp = df.kp.opt
  df.kp[,1:4] = df.kp[,1:4] / rowSums(df.kp[,1:4])
  df.epa  = df.kp %>% group_by(aproovitykk_id) %>%
summarise_all(funs(epa.kernel))
  df.mean = df.kp %>% group_by(aproovitykk_id) %>%
summarise_all(funs(mean))
  df.beta = df.kp %>% group_by(aproovitykk_id) %>%
summarise_all(funs(bets_fun))
  df.beta[,2:5] = df.beta[,2:5] / rowSums(df.beta[,2:5])
  df.epa[,2:5] = df.epa[,2:5] / rowSums(df.epa[,2:5])
  df.mean[,2:5] = df.mean[,2:5] / rowSums(df.mean[,2:5])
  dp.b = merge(df.beta, taks.info, all.x = T, by = "aproovitykk_id")
  dp.e = merge(df.epa, taks.info, all.x = T, by = "aproovitykk_id")
  dp.m = merge(df.mean, taks.info, all.x = T, by = "aproovitykk_id")
  RMSE.beta.opt[k] = sqrt(sum((dp.b[,11:14]-
dp.b[,2:5])**2)/(dim(dp.b)[1]*4))
  RMSE.epa.opt[k] = sqrt(sum((dp.e[,11:14]-
dp.e[,2:5])**2)/(dim(dp.e)[1]*4))
  RMSE.mean.opt[k] = sqrt(sum((dp.m[,11:14]-
dp.m[,2:5])**2)/(dim(dp.m)[1]*4))
  print(k);print("mean; beta; epa")
  print(RMSE.mean.opt);print(RMSE.beta.opt);print(RMSE.epa.opt)
}

#RMSE sõltuvalt naabrite arvust
par(mfrow = c(1,3))
xx = c(1:20)
plot(xx, RMSE.mean, type = "o", xlab = "Naabreid", ylab = "RMSE: keskmine",
ylim = c(0.158,0.22), lty = 2, cex.lab = 1.5)
lines(xx, RMSE.mean.opt,  col = "red");points(xx, RMSE.mean.opt,  col =
"red", pch = 16)
plot(xx, RMSE.beta, type = "o", xlab = "Naabreid", ylab = "RMSE: beta",
ylim = c(0.158,0.22), lty = 2, cex.lab = 1.5)
lines(xx, RMSE.beta.opt, col = "red");points(xx, RMSE.beta.opt,  col =
"red", pch = 16)
```

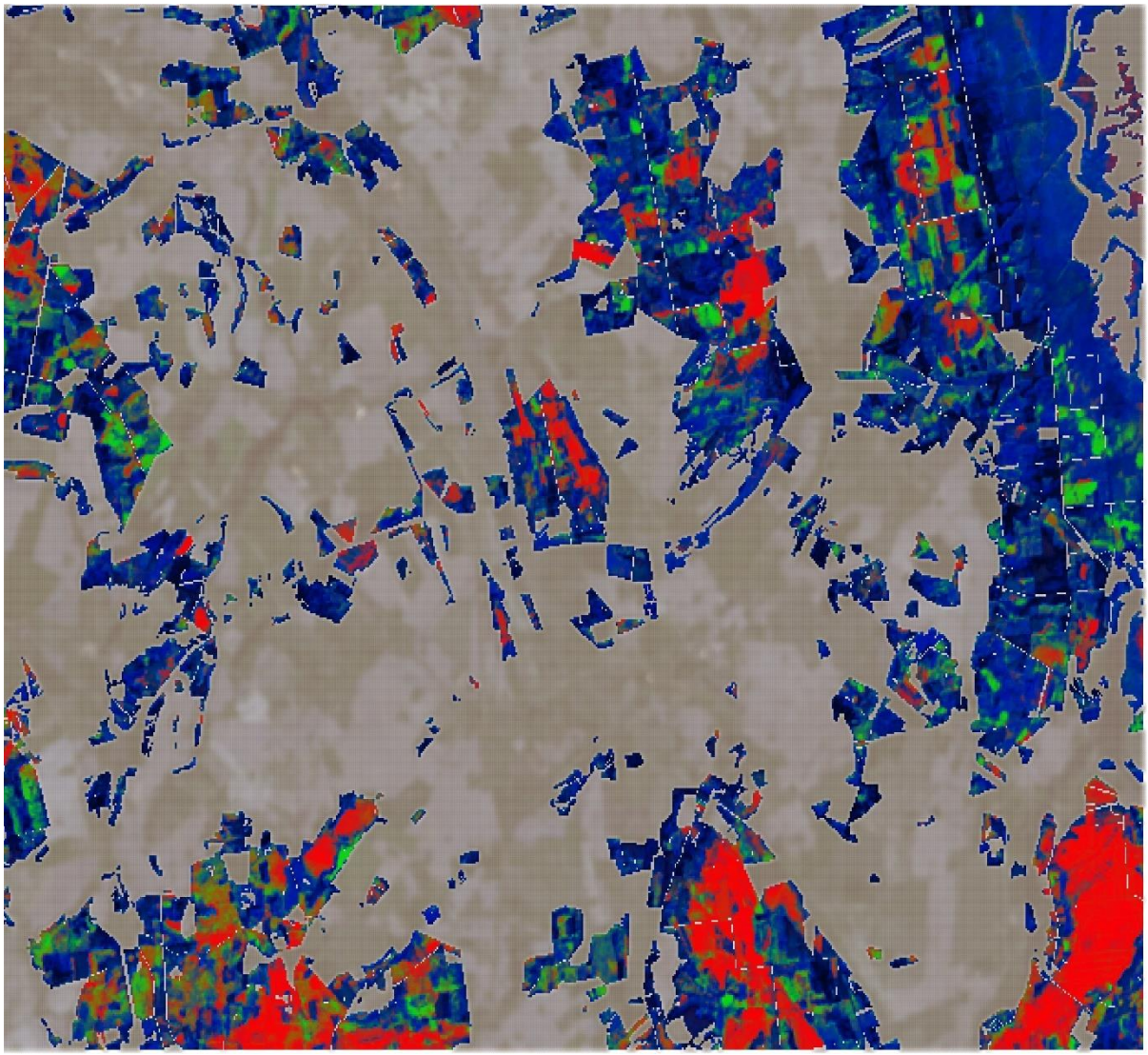
```
plot(xx, RMSE.epa, type = "o", xlab = "Naabreid", ylab = "RMSE:
Epanechnikov", ylim = c(0.158,0.22), lty = 2, cex.lab = 1.5)
lines(xx, RMSE.epa.opt, col = "red");points(xx, RMSE.epa.opt,  col = "red",
pch = 16)
```

Näide koodist: kaardi koostamine

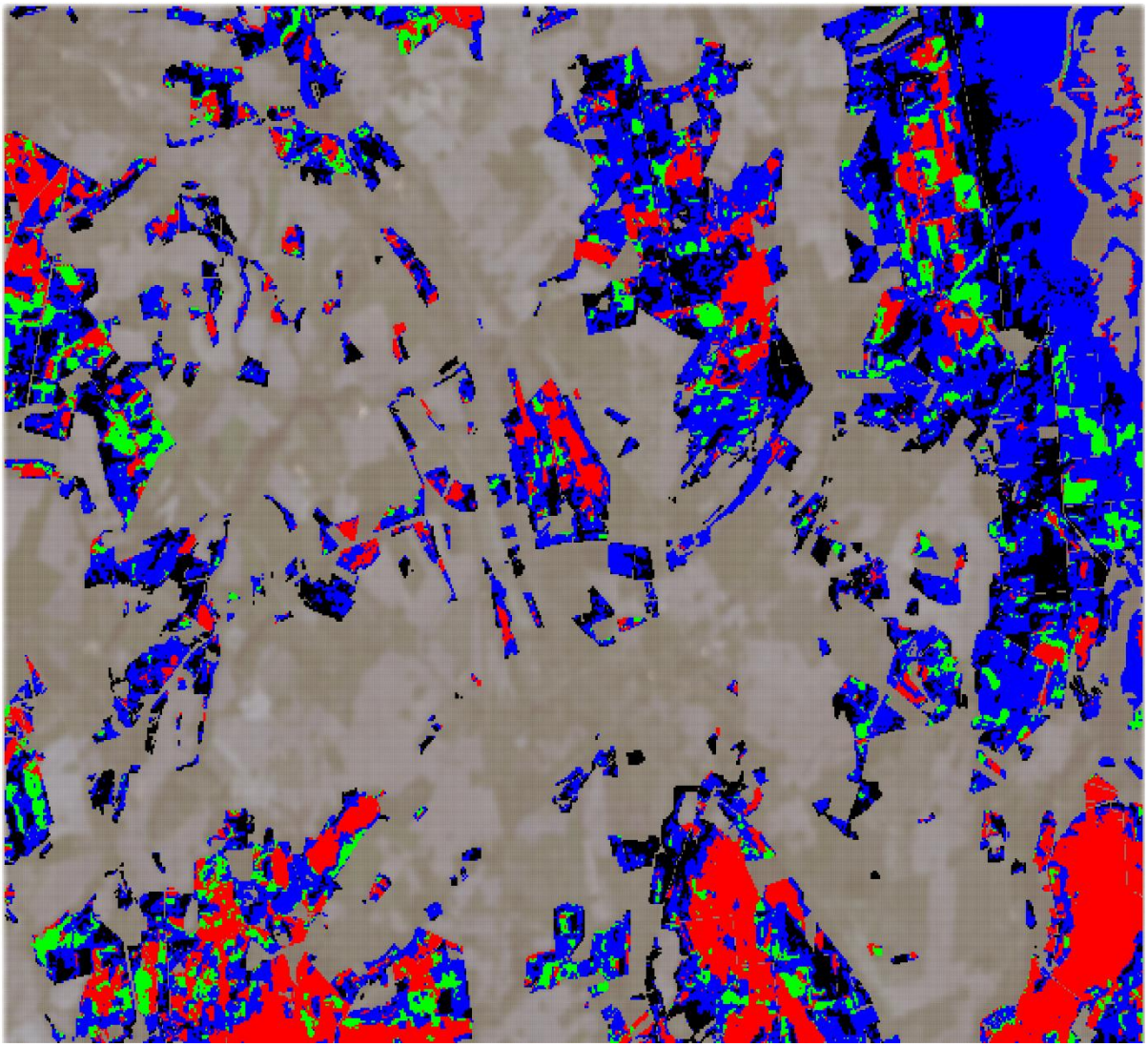
```
#Prognooskaardi koostamine
#Luunja-Võnnu ristkülik. Valitud, kuna satelliidi trajektoorid kattuvad,
seega pilte rohkem

#Agregeeritud pilt-haaval KNN prognoosid, K = 3
load(file = "kaart_sent_epaK3.RData")
load(file = "kaart_land_epaK3.RData")

kaart = rbind(sent.k3, land.k3)

require(EnvStats)
require(dplyr)
load("epa_kernel.RData")
#Agregeerimine Epanechnikovi tuumatiheduse moodi põhjal:
df.epa = kaart[,c(1:4,6,7)] %>% group_by(.dots=c("x","y")) %>%
summarise_all(funs(epa.kernel)) #epa.kernel: agregeeriv funktsioon
df.epa[,3:6] = df.epa[,3:6] / rowSums(df.epa[,3:6]) #vektorite pikkuse
normeerimine pärast agregeerimist

require(rgdal)
require(raster)
require(sp)
coordinates(df.epa) = ~ x + y
#koordinaatide referentssüsteem:
coord.ref = "+proj=lcc +lat_1=59.33333333333334 +lat_2=58
+lat_0=57.51755393055556 +lon_0=24 +x_0=500000 +y_0=6375000 +ellps=GRS80
+towgs84=0,0,0,0,0,0,0 +units=m +no_defs"
proj4string(df.epa) = coord.ref

#data.frame'ina:
df.epa = data.frame(df.epa)
#kõik punktid RGB skaalal; must seega muud liigid
require(ggplot2)
g = ggplot(data=df.epa, aes(x=x, y=y, col=rgb(MA,KU,KS))) + geom_point(size
= 0.4, shape = 15) + scale_color_identity()
g

#Metsakatastrid:
#metsaregistri andmed https://register.metsad.ee/#/
eraldis_era = readOGR(dsn = ".", layer = "eraldis")
eraldis_riik = readOGR(dsn = ".", layer = "eraldis_riik")
eraldis = rbind(eraldis_era, eraldis_riik)

#ruumiandmed punktkujule:
eraldis.points = spTransform(eraldis, CRS(coord.ref)) #landsati coord. ref
ext = c(669400,685000,6460000,6472500)
eraldis.ext = crop(eraldis.points, ext) #eraldised Luunja-Võnnu ristkülikus

eraldis.values = over(df.epa, eraldis.ext) #kontroll, kas piksel kuulub
mingisse eraldisse
eraldis.coords = cbind(coordinates(df.epa), eraldis.values)
```

```
coords_to_plot = eraldis.coords[!(is.na(eraldis.coords$ID)),][,1:2] #mõnda
eraldisse kuuluvad pikslid
coords_outof_plot = eraldis.coords[is.na(eraldis.coords$ID),][,1:2]
#eraldistest välja jäävad pikslid
coords_to_plot$sees = "jah";coords_outof_plot$sees = "ei"
coords = rbind(coords_to_plot,coords_outof_plot) #koordinaadid koos infoga,
kas kuulub mõnda metsaeraldisse

df = merge(df.epa, coords, by = c("x","y")) #prognoosid ja koordinaadid
df.in = df[df$sees == "jah",] #eraldisse kuulvad prognoosid
#Ainult katastritesse kuuluvad punktid plotina:
g1 = ggplot(data=df.in, aes(x=x, y=y, col=rgb(MA,KU,KS))) + geom_point(size
= 0.3, shape = 15) + scale_color_identity()
g1

#Taustaks üks (suvaline) Landsati pilt, kus pole pilvi või muid häiringuid
#setwd("A:/MAKA/KARU/pildid/Kagu-Eesti/2018/LC08_185019_20180512")
all_landsat <- list.files(pattern = ".jp2$",full.names = TRUE)
land1 <- readGDAL(all_landsat[1]); land2 <- readGDAL(all_landsat[2]); land3
<- readGDAL(all_landsat[3]) #RGB kanalid
land = cbind(land1,land2,land3)

land.sp = spTransform(land, CRS(coord.ref))
land.ext = crop(land.sp, ext)
df.land = data.frame(land.ext)
df.land.in.out = merge(df.land, coords, by = c("x","y")) #koos infoga, kas
piksel kuulub eraldisse või mitte
df.land.out = df.land.in.out[df.land.in.out$sees == "ei",] #eraldistest
välja jäänud osa
df.land.out$band4 = 0 #et oleks sama dimensionaalsus, mis prognoosil
names(df.land.out) = c("x","y","MA","KU","KS", "sees", "MUU"); #ja samad
veerunimed
df.land.out = df.land.out[c("x","y","MA","KU","KS", "MUU", "sees")] #... ja
sama järjekord
df.land.out$MA = df.land.out$MA / max(df.land.out$MA) #skaalateisendus RGB-
plottimiseks
df.land.out$KU = df.land.out$KU / max(df.land.out$KU)
df.land.out$KS = df.land.out$KS / max(df.land.out$KS)

df.in = df.in[,-7] #üleliigne veerg
df.inout = rbind(df.in, df.land.out) #prognoosid (eraldistes sees) ja
landsati pilt (eraldistest väljas)

#Prognoosid koos taustaga:
df.out = df.inout[df.inout$sees == "ei",]
g2 = ggplot(data = df.out, aes(x=x, y=y, col=rgb(MA,KU,KS))) +
geom_point(size = 2, alpha = 0.069) + scale_color_identity()
#salvestamine PNG formaadis:
png(filename="MA_KU_KS_MUU_prognoos.png", width = 3377, height = 2996,
units = "px", res = 300)
g2 + geom_point(data = df.in, aes(x=x, y=y, col=rgb(MA,KU,KS)), alpha = 1,
size = 0.3, shape = 15) + theme_bw() + theme(panel.border =
element_blank(), panel.grid.major = element_blank(), panel.grid.minor =
element_blank(), axis.line = element_line(colour = "black"))
dev.off()
```

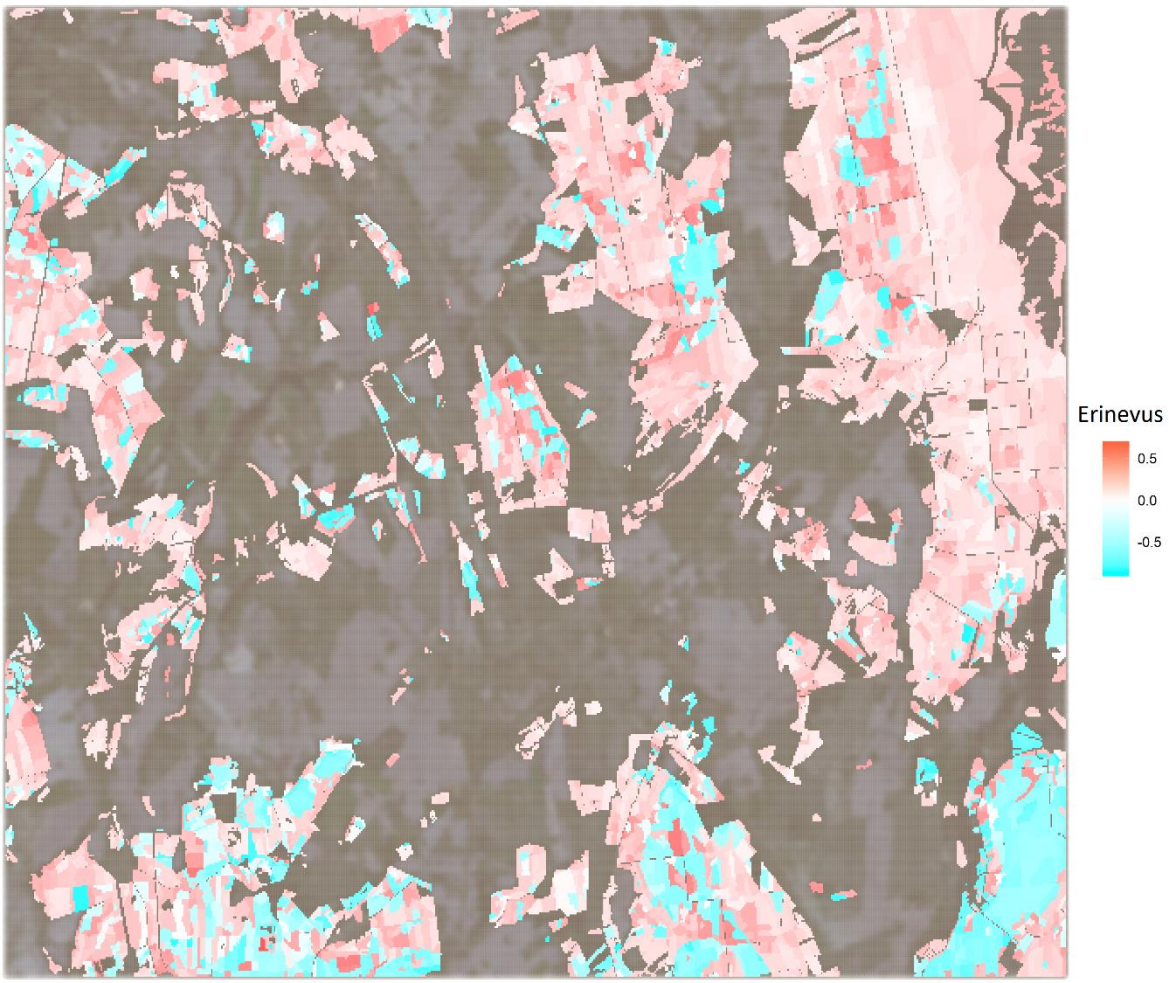
Regressioonipuu ja bagging *prognoosid kaardi kujul*



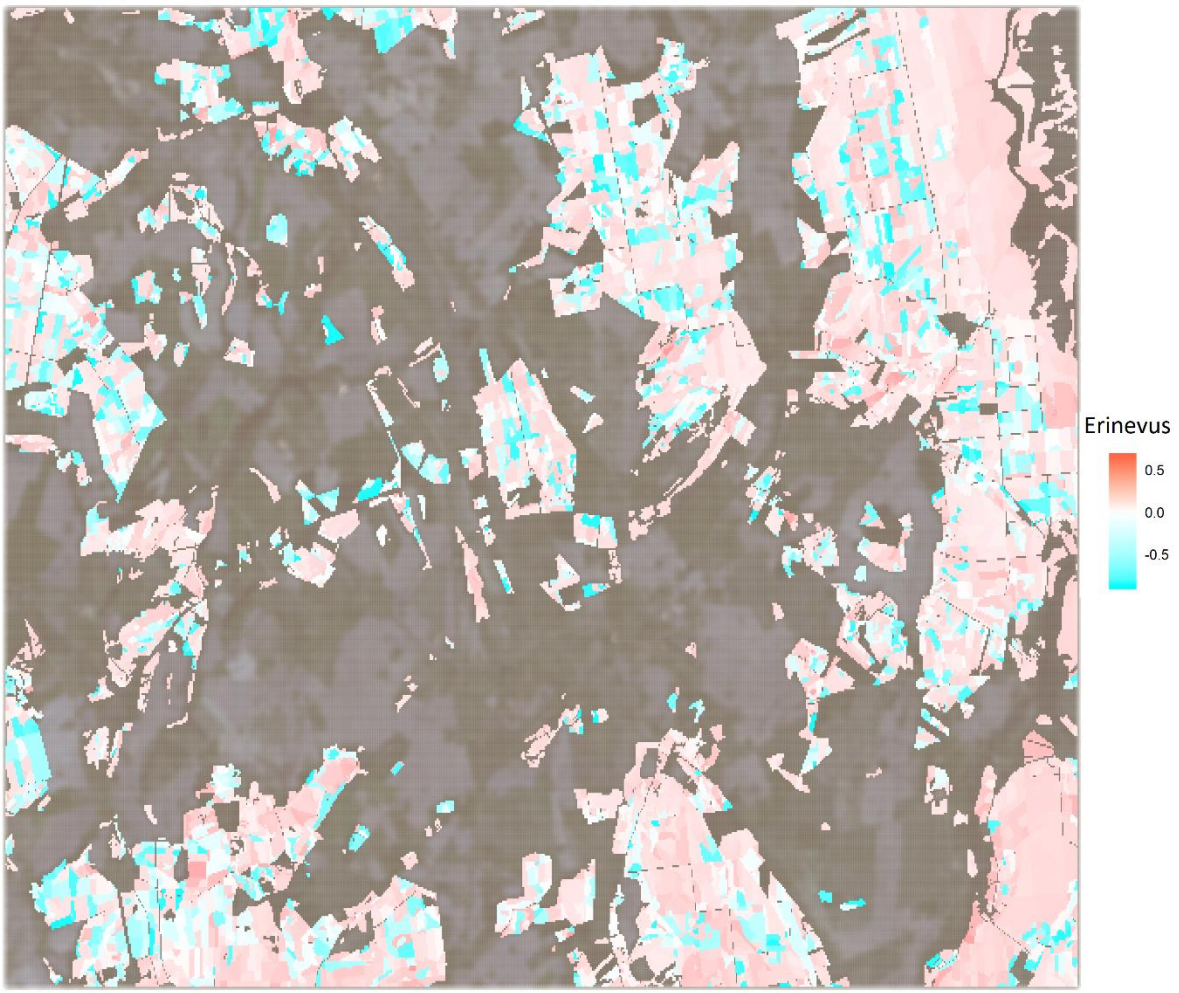
Joonis 40. Regressioonipuu ja bagging*. Puuliikide osakaalude vektorite prognoosid metsaeraldistel RGB skaalal. Punane – mänd; roheline – kuusk; sinine kask; must – muu*



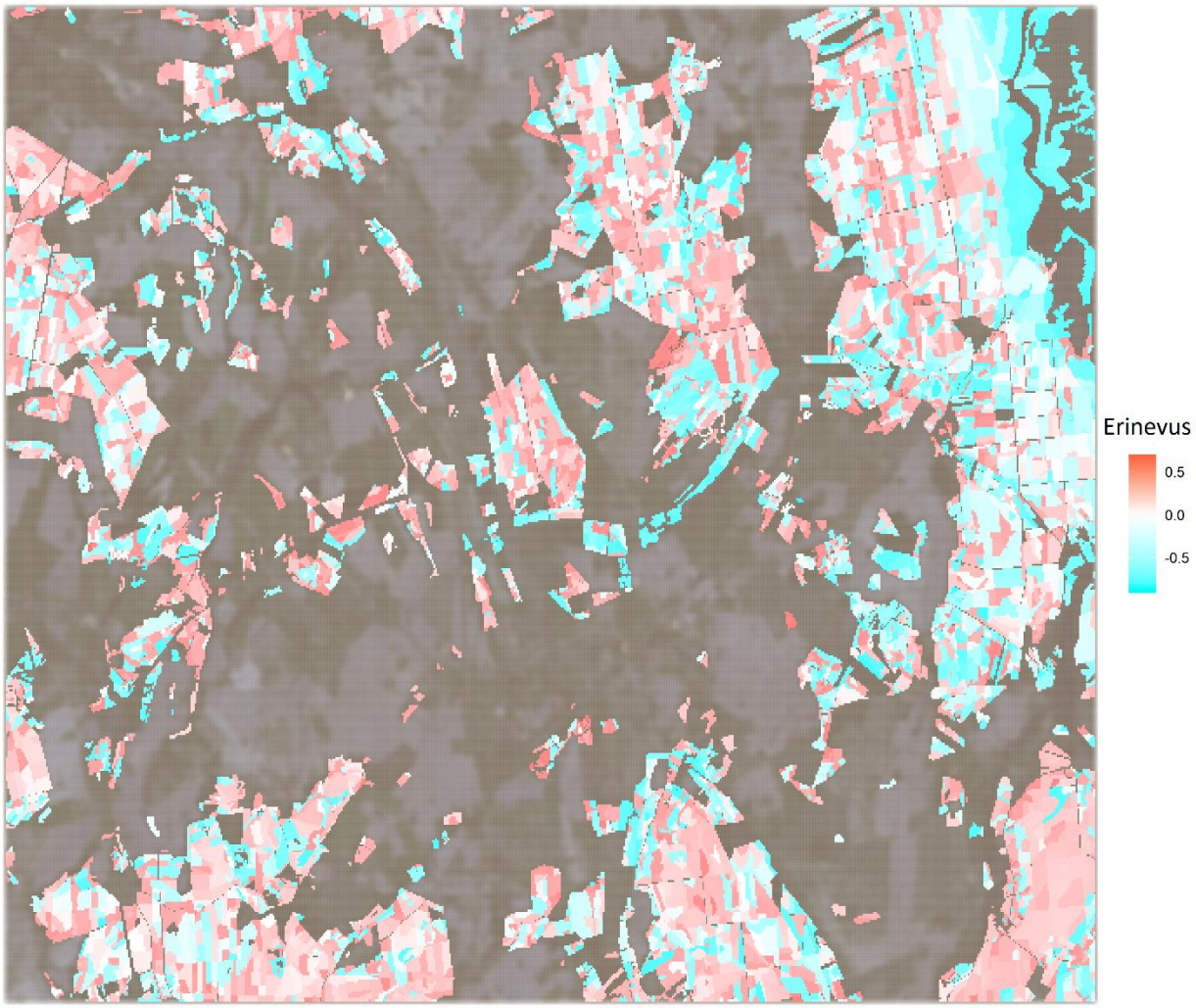
Joonis 41. Regressioonipuu ja bagging*. Enimlevinud puuliigi prognoosid metsaeraldistel RGB skaalal. Punane – mänd; roheline – kuusk; sinine kask; must – muu.*



Joonis 42. Männi osakaalu erinevus KNN pilt-haaval prognooside ja registriandmete vahel metsaregistri eraldiste kaupa. Positiivse erinevuse korral on prognoositud väärtus suurem kui registriandmete väärtus. Selgepiirilised intensiivsed tsüaniidsinised alad võivad olla raiesmikud.



Joonis 43. Kuuse osakaalu erinevus KNN pilt-haaval prognooside ja registriandmete vahel metsaregistri eraldiste kaupa. Positiivse erinevuse korral on prognoositud väärtus suurem kui registriandmete väärtus.



Joonis 44. Kase osakaalu erinevus KNN pilt-haaval prognooside ja registriandmete vahel metsaregistri eraldiste kaupa. Positiivse erinevuse korral on prognoositud väärtus suurem kui registriandmete väärtus. Selgepiirilised intensiivsed punased alad võivad olla raiesmikud.

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, Mats Ploompuu,

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose „Mitmemõõtmelised meetodid puuliikide osakaalude prognoosimiseks satelliidiandmete põhjal“, mille juhendaja on Märt Möls, reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada digitaalarhiivi DSpace kuni autoriõiguse kehtivuse lõppemiseni.

2. Annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi DSpace kaudu Creative Commonsi litsentsiga CC BY NC ND 3.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni.

3. Olen teadlik, et punktides 1 ja 2 nimetatud õigused jäävad alles ka autorile.

4. Kinnitan, et lihtlitsentsi andmisega ei riku ma teiste isikute intellektuaalomandi ega isikuandmete kaitse õigusaktidest tulenevaid õigusi.

Mats Ploompuu

15.05.2019