TARTU ÜLIKOOL
Arvutiteaduse instituut
Informaatika õppekava

Karl Riis

Bayesi isotoonilise kalibreerimise algoritm ja selle optimeerimine

Bakalaureusetöö (9 EAP)

Juhendaja: Meelis Kull, PhD

Tartu 2019

Bayesi isotoonilise kalibreerimise algoritm ja selle optimeerimine

Lühikokkuvõte: Töö käigus kirjeldati detailselt Mari-Liis Allikivi ja Meelis Kulli loodud Bayesi isotoonilise kalibreerimise algoritm ning üritati seda optimeerida, kuna see oli suurtel andmehulkadel aeglane. Lisaks kirjeldati algoritmiga seotud matemaatilisi mõisteid ja võtteid ning lahendati nende seletamiseks näiteülesandeid. Algoritmi edasiarenduseks kasutati erinevaid tehnikaid, mis tegid algoritmi töö stabiilsemaks ja kiiremaks. Lõpuks analüüsiti optimeeritud Bayesi isotoonilist kalibreerimist ning võrreldi seda isotoonilise kalibreerimisega ja logistilise regressiooniga tehislikel andmetel.

Võtmesõnad:
Isotooniline kalibreerimine, Bayesi järeldamine, klassifitseerija kalibreerimine.

CERCS: P176

Bayesian isotonic calibration and its optimisation

Abstract: The work focused on describing and optimising the Bayesian isotonic calibration algorithm created by Mari-Liis Allikivi and Meelis Kull. The algorithm needed optimisation because it was slow on large datasets. Different mathematical concepts related to the algorithm are described in detail. The algorithm is then improved with various techniques which make it more stable and faster. Finally the algorithm is compared to isotonic calibration and logistic regression using synthetic data.

Keywords:

Isotonic calibration, Bayesian inference, classifier calibration.

CERCS: P176

Sisukord

1 Sissejuhatus

Masinõppe rakendamine erinevates valdkondades on hiljuti märkimisväärselt populaarsust kogunud. Paljud levinud masinõppemudelid annavad väljundiks ennustuse mingisse klassi kuuluvuse kohta ning enamus neist väljastavad ka mingi skoori, mille abil saab klassidesse kuulumise ennustusi järjestada. Näiteks tugivektormasinate puhul on selleks skooriks ennustuse kaugus klasse eraldavast sirgest. Tihti on kasulik skoor teisendada klassi kuulumise tõenäosuseks. See on vajalik näiteks juhul, kui mudelile on vaja teha mingit järeltöötlust [1]. Skooride tõenäosusteks teisendamiseks saab neile rakendada klasside tõenäosuste kalibreerimise algoritme. Juhul, kui mudeli tehtud vead on potentsiaalselt tõsiste tagajärgedega, on väga tähtis, et mudel oleks hästi kalibreeritud [2].

Üks algoritm klasside tõenäosuste kalibreerimiseks on Mari-Liis Allikivi ja Meelis Kulli loodud Bayesi isotooniline kalibreerimine [2]. Läbiviidud testide põhjal selgus, et 153 kalibreerimisülesande korral töötab antud algoritm paremini või vähemalt sama hästi kui teised kaasaegsed kalibreerimisalgoritmid, nagu näiteks isotooniline regressioon ja Platti skaleerimine. Lisaks tootsid teised algoritmid liiga enesekindlaid mudeleid, Bayesi isotooniline kalibreerimine seda aga ei teinud.

Bayesi isotoonilise kalibreerimise algoritmi praegune versioon muutub aeglaseks ja ebastabiilseks, kui andmete hulk ületab 3000 isendit. Antud uurimus püüab algoritmi täiendada nii, et see töötaks hästi ka suurte andmehulkade korral.

2 Klassifitseerija kalibreerimine

Teoreetilise osa eesmärk on selgitada lugejale järk-järgult Bayesi isotoonilise kalibreerimise jaoks tähtsamaid põhimõtteid, lõpetades sellega, kuidas kõik omavahel seondub ning kuidas algoritm toimib.

2.1 Kalibreerimise definitsioon

Järgnev definitsioon pärineb raamatust "Machine Learning: The Art and Science of Algorithms that Make Sense of Data" [3].

Klassifitseerija kalibreerimine on mudeli väljastatud skooride või kalibreerimata tõenäosuste teisendamine kalibreeritud tõenäosusteks.

Klassifitseerija on täiuslikult kalibreeritud kui kehtib võrrand

$$P(Y_i = 1 \quad | \quad \hat{p}(X) = (\hat{p}_1, ..., \hat{p}_i)) = \hat{p}_i \quad \text{iga} \quad i = 1, ..., k \quad \text{korral},$$

kus X on üks sisendväärtus, $\hat{p}(X)$ on X-i klassijaotuse tõenäosuste vektor, i on üks ennustatavatest klassidest ning $\hat{p_i}$ on i-ndasse klassi kuulumise tõenäosus.

Ehk üle kõikide andmepunktide, mille korral mudel ennustab jaotust $(\hat{p}_1, ..., \hat{p}_k)$, siis ka tegelik klassijaotus on $(\hat{p}_1, ..., \hat{p}_k)$.

2.2 Isotooniline regressioon

Isotooniline regressioon on levinud algoritm binaarse klassifitseerija kalibreerimiseks. See minimeerib treeningandmetel ruutkeskmise vea, ta sobitab neile kõige tõepärasema monotoonse ehk mittekahaneva kõvera [4].

Algoritmi tööpõhimõte [4]:

Olgu $(x_1, ..., x_n)$ treeningpunktide hulk, $g(x_i)$ õpitava funktsiooni väärtused, mis seatakse algoritmi töö algul võrdseks nende treeningpunktide märgenditega (0 või 1). Kui funktsioon g on juba mittekahanev, siis tagastatakse seesama funktsioon. Vastasel juhul leiduvad vähemalt kaks punkti $g(x_i)$ ja $g(x_{i+1})$, mille puhul ei ole monotoonsust ehk kehtib $g(x_i) > g(x_{i+1})$, sellisel juhul asendatakse $g(x_i)$ ja $g(x_{i+1})$ nende aritmeetilise keskmisega $\frac{g(x_i)+g(x_{i+1})}{2}$. Nüüd vastavad need kaks punkti isotoonsuse tingimusele. Sama protsessi jätkatakse kuni enam ei leidu ühtki punktipaari, mis tingimusele ei vasta.

Kui viimane treeningpunkt x_n on positiivne ehk $g(x_n) = 1$, siis isotooniline kalibreerimine tagastab iga testpunkti $x' \geq x_n$ korral ennustuseks 1. Teisisõnu ei jäeta mingit võimalust, et tegu ei ole positiivse punktiga. Võib siiski juhtuda, et tegelikkuses on x' negatiivne punkt. Seega võib isotooniline kalibreerimine toota liiga enesekindlaid mudeleid.

Üleliigse enesekindluse vältimiseks kasutame Bayesi lähenemist.

2.3 Bayesi lähenemine masinõppes

Bayesi lähenemine on matemaatilise statistika liik, mis on asjakohane selliste probleemide lahendamisel, mille korral peab tegema järeldusi mingi parameetri suhtes, mille kohta on eelnevalt teada vähe informatsiooni. Bayesi lähenemise aluseks on Bayesi teoreem [5].

Box ja Tiao kirjeldavad raamatus "Bayesian inference in statistical analysis"[5] Bayesi teoreemi järgmiselt:

Olgu $y = (y_1, ..., y_n)$ vektor n vaatlusest, mille tõenäosusjaotus $p(y|\theta)$ sõltub k parameetrist $\theta = (\theta_1, ..., \theta_k)$. Olgu θ ise tõenäosusjaotusega $p(\theta)$. Siis kehtib võrrand

$$p(y|\theta)p(\theta) = p(y, \theta) = p(\theta|y)p(y)$$

Sellest saame Bayesi teoreemi:

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)} \tag{1}$$

Võrrandis 1 märgib $p(\theta)$ olemasolevaid teadmisi parameetri θ kohta ilma vaatlusteta, $p(\theta)$ nimetatakse θ eeljaotuseks. Seevastu $p(\theta|y)$ näitab, mida teame θ kohta peale mingit vaatlust y, seda nimetatakse θ järeljaotuseks. Tõenäosus $p(y|\theta)$ näitab kui tõenäoline on vaatluse y toimumine kui tegelikkus on θ. Seda nimetatakse tõepäraks. Bayesi teoreemi järgi on järeljaotus proportsionaalne tõepära ja eeljaotuse korrutisega, sest vaatlusi y võib vaadelda konstantsena kui eesmärk on leida θ järeljaotus:

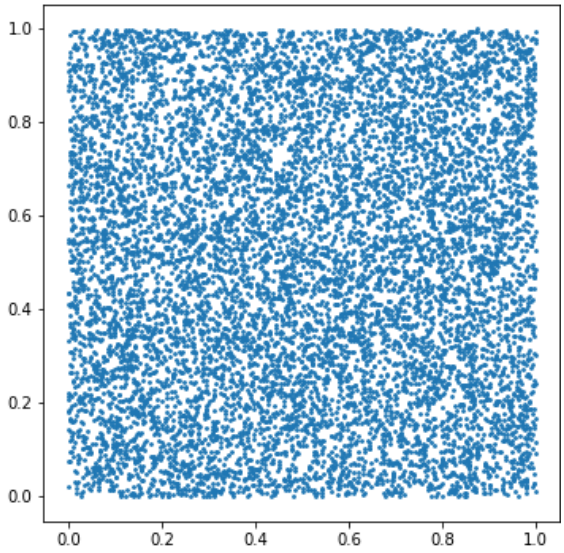
$$p(\theta|y) \propto p(y|\theta) \times p(\theta)$$

2.4 Ühtlase jaotuse keskväärtuse hindamine Monte Carlo meetodiga

Järgnevalt uurime keskväärtuse hindamise ülesannet ühe konkreetse kahemõõtmelise jaotuse puhul, mis osutub olema seotud Bayesi isotoonilise kalibreerimise ülesandega.

Olgu meil kaheelemendilist jaotust (p_1, p_2) genereeriv protsess, kus mõlemad arvud p_1 ja p_2 valitakse ühtlaselt juhuslikult vahemikust [0, 1]. Kui genereerime suure

hulga arve ning kujutame neid graafikul, kus telgedeks on p_1 ja p_2 väärtused, siis tekib kujuteldav ruut, mis on ühtlaselt täidetud punktidega (vt joonis 1).



Joonis 1: Kahemõõtmeline ühtlane jaotus üle piirkonna $[0, 1] \times [0, 1]$

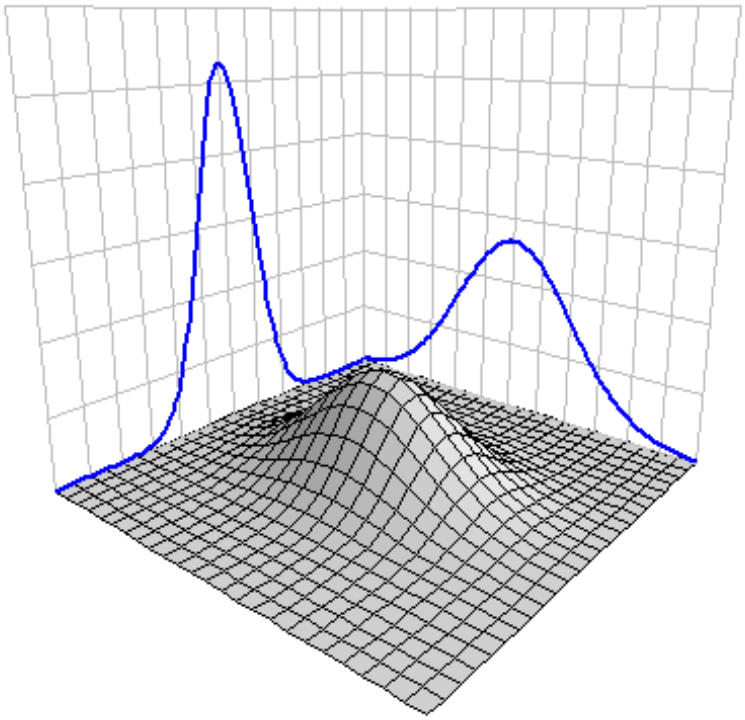
Hindame nüüd jaotuse keskväärtust $E[(P_1, P_2)]$. Selleks leiame mõlema suuruse jaoks eraldi keskväärtused, st leiame eraldi x ja y telje suunas keskväärtused.

Valem juhusliku suuruse X keskväärtuse leidmiseks on järgmine:

$$E(X) = \int_{-\infty}^{\infty} x f_X(x) dx, \tag{2}$$

kus $f_X(x)$ on juhusliku suuruse X tihedusfunktsioon [6].

Tihedust võib kaheelemendise jaotuse puhul kujutada ette kolmanda mõõtme lisandumisega. Tihedamad alad ulatuvad kõrgemale ja hõredamad alad jäävad madalamaks. Vaata näidet joonisel 2, kus on kujutatud kahest normaaljaotusest tekkiv tihedus. Ülesandes käsitletava ühtlase jaotusega ruudu puhul on tihedus igal pool ühtlane ehk tekib risttahukas.



Joonis 2: Tiheduse näide kahe muutuja puhul

Kuna käesoleva ruudukujulise näite puhul on jaotus ühtlane, siis selle tihedusfunktsioon on $f_{P_1,P_2}(p_1,p_2) = 1$.

Arvutame keskväärtused analüütiliselt valemi 2 järgi:

$$E[P_1] = \int\int p_1 f_{P_1,P_2}(p_1,p_2)dp_1dp_2 = \int_0^1\int_0^1 x f_{P_1,P_2}(x,y)dxdy =$$
$$\int_0^1\int_0^1 xdxdy = \int_0^1 x(\int_0^1 dy)dx = \int_0^1 xdx = \frac{x^2}{2}\Big|_0^1 = \frac{1}{2} - \frac{0}{2} = \frac{1}{2}$$

$$E[P_2] = \int\int p_2 f_{P_1,P_2}(p_1,p_2)dp_1dp_2 = \int_0^1\int_0^1 y f_{P_1,P_2}(x,y)dxdy =$$
$$\int_0^1\int_0^1 ydxdy = \int_0^1 y(\int_0^1 dx)dy = \int_0^1 ydx = \frac{y^2}{2}\Big|_0^1 = \frac{1}{2} - \frac{0}{2} = \frac{1}{2}$$

Saime, et jaotuse (P_1, P_2) keskväärtus on $(\frac{1}{2}, \frac{1}{2})$.

Antud näite puhul ei olnud keskväärtuse leidmine keeruline, kuid ülesanne läheb arvutuslikult palju keerukamaks kui jaotuse mõõde kasvab. Iga uue mõõtme kohta tuleb arvutusse üks integraal juurde. Integraalide arvutamise vältimiseks on võimalik ülesanne ligikaudse lähendiga lahendada ka lihtsa Monte Carlo meetodiga.

Integraali $I = \int_R g(x) f_X(x) dx$ ligikaudne väärtus I_n on saadud lihtsa Monte Carlo meetodiga kui

$$I_n = \frac{1}{n} \sum_{i=1}^{n} g(x_i),$$

kus $g(x)$ on jaotust genereeriv funktsioon, $f_X(x)dx$ on jaotuse X tihedusfunktsioon ning x_i on juhuslikud arvud jaotusest X [7]. Sisuliselt on tegu suurel hulgal suvaliste punktide võtmisega ja nende aritmeetilise keskmise arvutamisega.

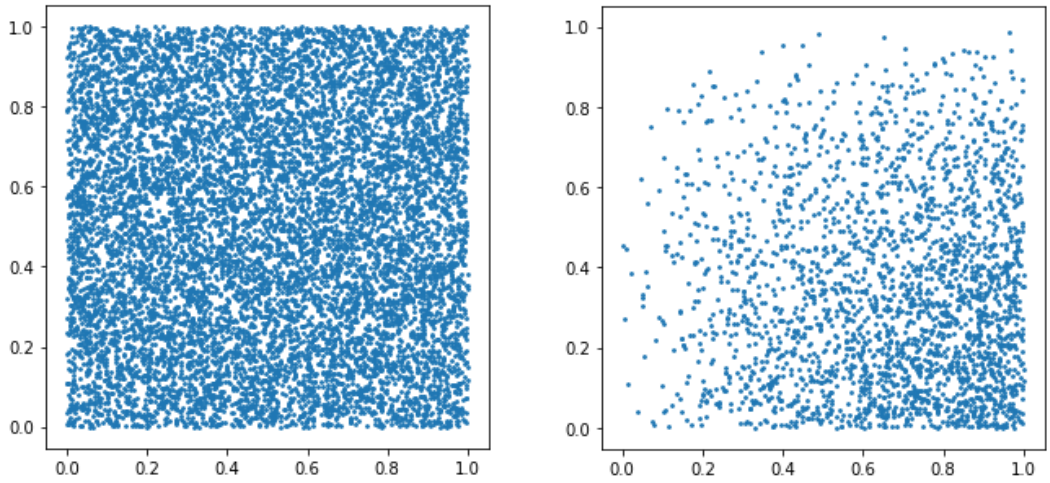
Kasutame lihtsat Monte Carlo meetodit eelnevalt kirjeldatud jaotuse keskväärtuse hindamiseks. Valime esialgu jaotusest kolm suvalist punkti, olgu nendeks (0.18, 0.49), (0.29, 0.78), (0.34, 0.81). Nende punktide aritmeetiline keskmine on (0.27, 0.6933). See tulemus erineb täpsest keskväärtusest üsna palju. Kui valime aga 100 suvalist punkti, siis on Monte Carlo meetodi tulemuseks (0.5185, 0.4926), mis on juba oluliselt lähedasem tulemus. Seega mida rohkem punkte valida, seda väiksem on erinevus tegelikust keskväärtusest.

2.5 Järeljaotuse keskväärtuse hindamine Monte Carlo meetodiga

Järeljaotuse keskväärtuse hindamist on lihtsam selgitada läbi näiteülesande. Olgu meil kaks ebaausat münti, mille puhul on vastavalt tõenäosus kull tulla p_1 ja p_2. Püüame peale ühe vaatluse nägemist ennustada järgmiste visete tulemusi.

Ilma ühegi vaatluseta pole meil midagi targemat teha, kui ennustada täiesti juhuslikult. Sisuliselt peame ennustama eeljaotuse põhjal, milleks on eelmises peatükis kirjeldatud ruudukujuline ühtlane jaotus, kuna p_1 ja p_2 väärtuseks võivad olla suvalised arvud vahemikust $[0, 1]$. Selle jaotuse puhul on x-telg esimese mündi tõenäosus tulla kull ning y-telg teise mündi tõenäosus tulla kull.

Oletame, et me saime nüüd ühe korra münte visata ning tulemuseks tulid kull ja kiri, märgime $Y_1 = 1$ ja $Y_2 = 0$. Selle põhjal saab nüüd järeljaotuse moodustada nii, et genereerime ühtlaselt juhuslikult hulga mündipaare (p_1, p_2), viskame ühe korra iga sellist mündipaari ning jätame alles vaid need paarid, mille korral on tulemuseks 1 ja 0. Kõikidele paaridele ja allesjäänud paaridele vastavad punktid on toodud vastavalt joonistel 3a ja 3b.



(a) Eeljaotus

(b) Järeljaotus

Joonis 3

Kuna viske tulemus oli $Y_1 = 1$ ja $Y_2 = 0$, siis on järeljaotusest näha, et olemasoleva info põhjal tulekski eelistada piirkonda, kus esimese mõndi tõenäosus p_1 on lähedane arvule 1 ja teise mündi tõenäosus p_2 on lähedane arvule 0 ehk alumise

parema nurga ümbrus järeljaotuse ruudus.

Arvutame ka järeljaotuse täpse keskväärtuse. Tähistame järeljaotuse tihedusfunktsiooni $f'_{P_1,P_2}(p_1,p_2)$. Kuna me ei tea milline järeljaotuse tihedusfunktsioon on, siis me ei saa kasutada valemit 2, mida kasutasime eeljaotuse puhul. Selle lahendamiseks on meil võimalik kasutada Bayesi reeglit ja kirjutada järeljaotus lahti eeljaotuse $f_{P_1,P_2}(p_1,p_2)$ ja tõepära $l(p_1,p_2)$ korrutisena. Kuna jaotus pole enam ühtlane, siis tuleb see läbi jagada ka normaliseeriva konstandiga, et tihedusfunktsiooni alune ruumala oleks 1.

$$f'_{P_1,P_2}(p_1,p_2) = \frac{l(p_1,p_2)\cdot f_{P_1,P_2}(p_1,p_2)}{z},$$

kus

$$z = \int_0^1 \int_0^1 l(p_1,p_2)\cdot f_{P_1,P_2}(p_1,p_2)dp_1dp_2$$

Tõepära saab arvutada järgmiselt:

$$\begin{aligned} l(p_1,p_2) &= P(Y_1=1,Y_2=0|P_1=p_1,P_2=p_2) = \\ &= P(Y_1=1|P_1=p_1)\cdot P(Y_2=0|P_2=p_2) = p_1(1-p_2) \end{aligned}$$

Ülaltoodu põhjal saame arvutada järeljaotuse keskväärtuse järgmiselt:

$$\begin{aligned} E[P_1|Y_1=1,Y_2=0] &= \int_0^1 \int_0^1 p_1 f'_{P_1,P_2}(p_1,p_2)dp_1dp_2 = \\ &= \int_0^1 \int_0^1 p_1 \frac{l(p_1,p_2)f_{P_1,P_2}(p_1,p_2)}{z} = \int_0^1 \int_0^1 p_1 \frac{p_1(1-p_2)\cdot 1}{0.25}dp_1dp_2 = 0.667 \end{aligned}$$

$$\begin{aligned} E[P_2|Y_1=1,Y_2=0] &= \int_0^1 \int_0^1 p_2 f'_{P_1,P_2}(p_1,p_2)dp_1dp_2 = \\ &= \int_0^1 \int_0^1 p_2 \frac{l(p_1,p_2)f_{P_1,P_2}(p_1,p_2)}{z} = \int_0^1 \int_0^1 p_2 \frac{p_1(1-p_2)\cdot 1}{0.25}dp_1dp_2 = 0.333 \end{aligned}$$

Saime, et $E[(P_1, P_2)|Y_1 = 1, Y_2 = 0]$ on (0.667, 0.333).

Nüüd proovime jälle sama tulemuseni lihtsa Monte Carlo meetodiga jõuda. Valime kolm suvalist punkti - (0.99, 0.01), (0.4, 0.29), (0.78, 0.19). Nende keskmine on (0.7233, 0.1633), mis on taas täpsest väärtusest kaugel. Saja suvalise punkti korral saame tulemuseks juba (0.6536, 0.3287), mis on lähedal tõesele väärtusele.

Jooniselt 3b on näha, et isotoonsuse tingimuse tõttu võib järeljaotus sisaldada eeljaotusega võrreldes väga vähe punkte. Väheste punktide põhjal ei pruugi me saada head hinnangut jaotuse keskväärtusele. Uurime järgmises peatükis, kuidas seda probleemi lahendada.

2.6 Olulise/kaalutud valimi meetod

Eelnevalt jätsime järeljaotuse tekitamiseks eeljaotusest välja punktid, mis vaatlusele ei vastanud. See tähendab seda, et osa arvutusressurssi, mis kasutati nende punktide eeljaotusesse tekitamiseks, oli raisatud. Olulise/kaalutud valimi meetodiga on siiski võimalik ka need punktid ära kasutada, mis eeljaotusest muidu eemaldataks.

Püüame jälle hinnata ühe järeljaotuse keskväärtust. Järgnev tuletus põhineb olulise valimi meetodil, mis on kirjeldatud Tõnu Kollo raamatus "Monte Carlo meetodid" [7].

Kirjutame järeljaotuse $f_X^{'}(x)$ taas Bayesi reegli abil lahti eeljaotuse $f_X(x)$ ja tõepära $l(x)$ korrutiseks, et saaksime järeljaotuse keskväärtust eeljaotuse põhjal hin-

nata.

$$E[X = x|Y = y] = \int x f_X^{'}(x)dx = \int x \cdot \frac{l(x)f_X(x)}{z}dx = \\ = \int \frac{x \cdot l(x)}{z} f_X(x)dx = \int g(x)f_X(x)dx \approx \frac{1}{n}\sum_{i=1}^{n} g(x_i),$$

kus $l(x)$ on väärtuse x tõepära, z on normaliseeriv konstant ja $g(x) = \frac{x \cdot l(x)}{z}$.

Avaldises 6 jõudsime lihtsa Monte Carlo meetodi abil selleni, et saame järeljaotust hinnata funktsiooni $g(x)$ põhjal. Tähistame normaliseeritud tõepära $\frac{l(x_i)}{z}$ ümber kaaluks w_i ehk kirjutame $g(x_i)$ lahti järgmiselt:

$$g(x_i) = \frac{l(x_i)}{z} \cdot x_i = w_i \cdot x_i$$

Seejuures saame kaalud arvutada tõepäradest renormaliseerides ehk $w_i = \frac{l(x_i)}{\sum_j l(x_j)}$.

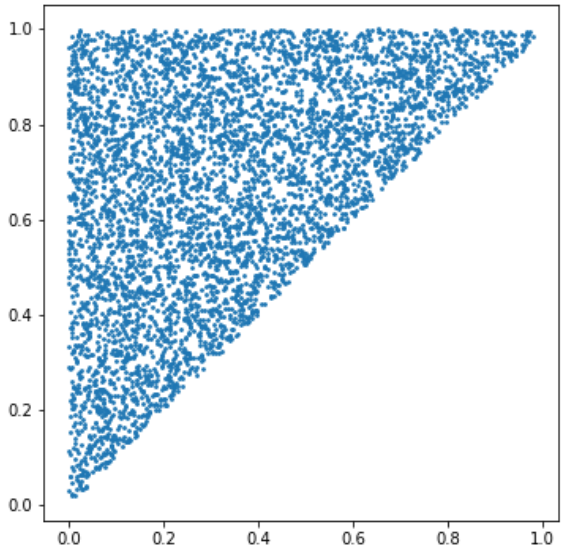
Jõudsime tulemuseni, et funktsioon $g(x)$, mille põhjal saime järeljaotust hinnata, on lihtsalt eeljaotusest punktide võtmine korrutatuna kaaludega. Seega ei lähe väikese tõepäraga punktid enam jaotusest kaduma, vaid neile omistatakse väike kaal, mistõttu saame need ikkagi ära kasutada.

2.7 Keskväärtuse hindamine kalibreerimisülesande puhul

Liigume nüüd kalibreerimisülesande juurde. Olgu meil mingi binaarse klassifitseerija poolt väljastatud skooride vektor $\boldsymbol{s} = (s_1, ..., s_n)$, mis vajab kalibreerimist. Kalibreerimise puhul tegeletakse järjestatud skooride vektoriga, kuna üldjuhul mida suurem on skoor, seda suurem on andmepunkti tõenäosus olla positiivne. Seega moodustatakse vektor nii, et kehtiks tingimus $s_1 \leq s_2 \leq ... \leq s_n$. Kalibreerimise jaoks on meil vaja teada nende punktide märgendeid, olgu need $y_1, ..., y_n \in \{0, 1\}$.

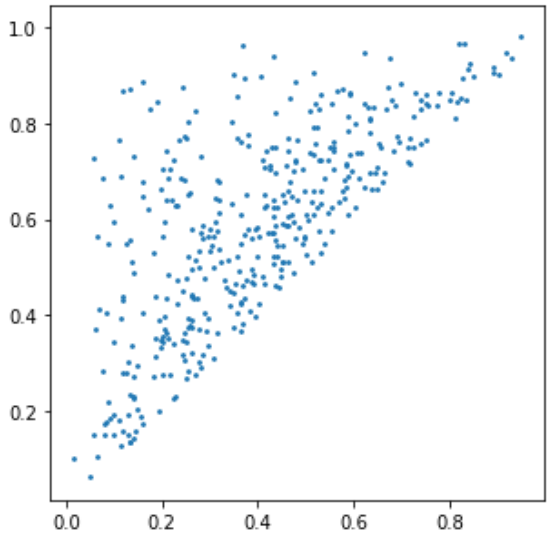
Nüüd tahame $y_1, ..., y_n$ põhjal ennustada punktide $s_1, ..., s_n$ kohta nende tõenäosusi olla positiivne. Seda saab vaadelda ülesandena n kallutatud mündist $p_1, ..., p_n$, kus eeldame, et $p_1 \leq ... \leq p_n$ ning meil on igast mündist üks vaatlus, st y_i.

Vaatame nüüd erijuhtu kahe mündiga. Olgu meil 2 kallutatud münti, mille tõenäosused kull olla on p_1 ja p_2. Kalibreerimisülesande raames paneme nüüd paika tingimuse $p_1 \leq p_2$ ehk tekitame isotoonsuse punktide vahel. Eeljaotust genereerides rakendame tagasilükkega valikut ehk jätame jaotusest välja andmepunktid, mille korral $p_1 > p_2$. Nii jääb ruudust punktidega täidetud osaks vaid sirge $p_1 = p_2$ ülessepoole jääv osa ehk uus jaotus on kujuteldav kolmnurgana (vt joonis 4). See on nüüd uus eeljaotus.



Joonis 4: Tagasilükkega valik

Oletame taas, et toimus vaatlus, mille korral visati kull ja kiri ($Y_1 = 1, Y_2 = 0$). Selleks, et järeljaotust kujutada, jätame jaotust genereerides jälle välja sellised müntide paarid, mille viskamisel ei saanud tulemust ($Y_1 = 1, Y_2 = 0$) (vt joonis 5).



Joonis 5: Isotoonse jaotuse järeljaotus vaatluse $Y_1 = 1, Y_2 = 0$ korral

Hinnates järeljaotuse keskväärtust lihtsa Monte Carlo meetodiga 100 punkti põhjal saame tulemuseks $E[P_1, P_2|Y_1 = 1, Y_2 = 0] = (0.4039, 0.5882)$.

2.8 Bayesi isotooniline kalibreerimine

Bayesi isotooniline kalibreerimine on kirjeldatud hetkel avaldamata artiklis "Non-parametric Bayesian Isotonic Calibration: Fighting Over-confidence in Binary Classification", mille autoriteks on Mari-Liis Allikivi ja Meelis Kull [2]. Käesoleva uurimistöö käigus algoritm täienes ning erineb mõne sammu poolest artiklis olevast versioonist. Järgnevalt kirjeldame täiendatud algoritmi.

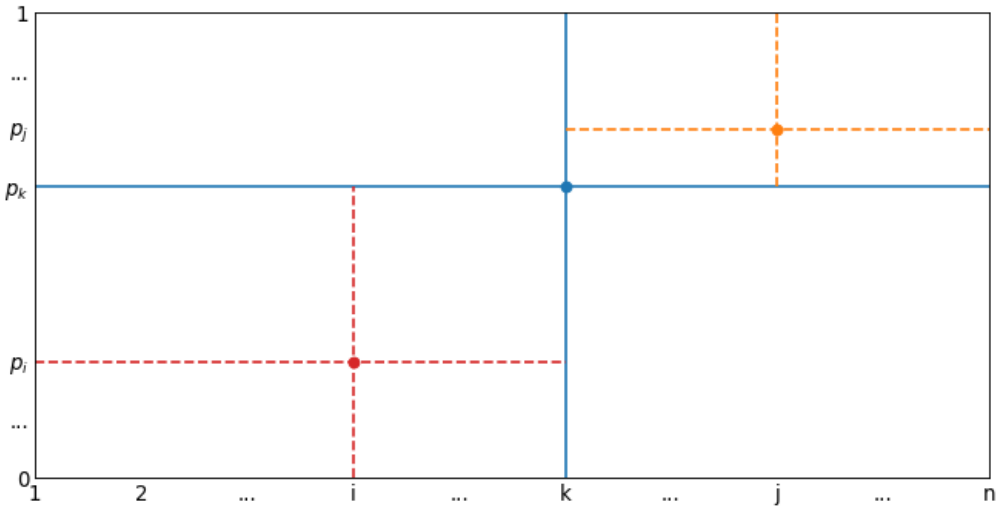
Bayesi isotooniline kalibreerimine defineerib eeljaotuse üle $p_1, ..., p_n$ läbi generatiivse protsessi, mis on järgmine:

Algoritm 1 Eeljaotuse genereerimine

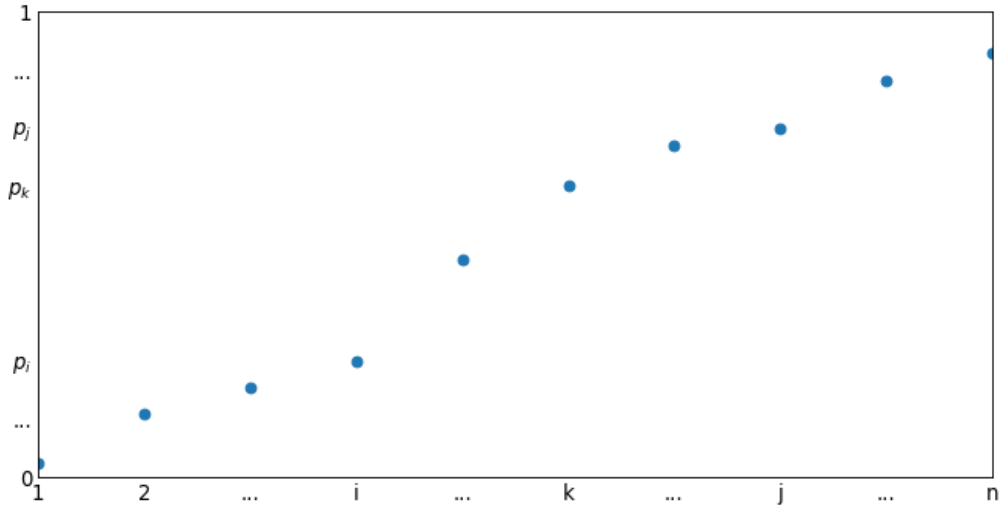
```
1: function GENEREERI_EELJAOTUS(k_min, k_max, p_min, p_max)
2:     k ← randint(k_min, k_max)
3:     p_k ← uniform(p_min, p_max)
4:     if k ≠ k_min then
5:         GENEREERI_EELJAOTUS(k_min, k − 1, p_min, p_k)
6:     end if
7:     if k ≠ k_max then
8:         GENEREERI_EELJAOTUS(k + 1, k_max, p_k, p_max)
9:     end if
10: end function
```

Algoritmis 1 on *randint* funktsioon juhuslike täisarvude valimiseks ning $uniform$ funktsioon juhusliku reaalarvu valimiseks.

Näide algoritmi tööst on joonisel 6. Näites valiti esimese sammuna täisarv k ja siis vahemikust $[0, 1]$ ühtlaselt juhuslikult p_k. Seejärel liiguti vasakule ja nüüd sai valiku teha ainult vasakpoolses alumises alamristkülikus (siniste piiridega). Seal valiti i ja p_i. Esialgsest valikust k liiguti ka paremale, kus uue valiku sai teha parempoolses ülemises alamristkülikus. Seal valiti j ja p_j. Joonisel 7 on kujutatud võimalik tulemus, kui algoritm on töö lõpetanud.



Joonis 6: Eeljaotuse loomise näide, 3 punkti valitud



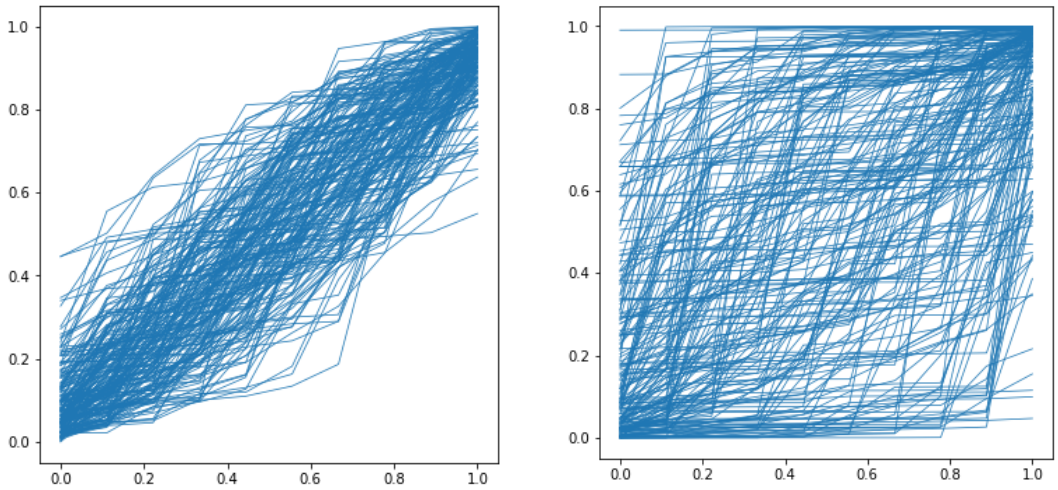
Joonis 7: Näide eeljaotuse algoritmi tulemusest

Joonisel 8b on kujutatud 10 andmepunkti kohta 200 eeljaotuse loomist Bayesi isotoonilise kalibreerimise meetodiga. Andmepunktid $1, ..., 10$ on x-teljel normaliseeritud vahemikku $[0, 1]$ ning y-teljel on tõenäosused vahemikus $[0, 1]$.

Alternatiiviks sellise valiku tegemisele oleks ühtlaselt juhuslikult n punkti valimine

vahemikust $[0, 1]$ ja nende sorteerimine. Sellisel juhul grupeeruksid kõik eeljaotused diagonaalile nagu on näidatud joonisel 8a. See oleks halb, kuna kui tõene kalibreerimiskõver paikneks diagonaalist eemal, siis oleks vaja eksponentsiaalselt rohkem andmeid, et sellele lähedaseid kõveraid tekitada.

Bayesi isotoonilise kalibreerimise viisil eeljaotuste loomine täidab kõverate ruumi paremini, kuna esimene punkt, mis valitakse, võib paikneda ükskõik kus. Kuhu iganes esimene punkt valitakse, seda kohta peab genereeritav eeljaotus ka läbima. Seetõttu saavad eeljaotused sattuda ükskõik kuhu saab üks suvaline punkt sattuda.



(a) Ühtlaselt juhuslik valik

(b) Bayes-Iso algoritmi valik

Joonis 8

Selliseid eeljaotusi luuakse sama generatiivse protsessiga algoritmi tarbeks palju. Kui eeljaotused on loodud, siis saab hinnata järeljaotuse keskväärtust $E(p_1, ..., p_n|y_1, ..., y_n)$ Monte Carlo meetodil:

$$E(p_1, ..., p_n|y_1, ..., y_n) = \frac{1}{m} \sum_{i=1}^{m} l(p_1^{(i)}, ..., p_n^{(i)}) \cdot (p_1^{(i)}, ..., p_n^{(i)}|y_1, ..., y_n)$$

3 Bayesi isotoonilise kalibreerimise optimeerimine

Selles peatükis kirjeldatakse eraldi suuremaid samme bayesi isotoonilise kalibreerimise algoritmi arengus.

3.1 Esialgse algoritmi implementeerimine

Esimese sammuna oli vaja programmeerida algoritm oma esialgses seisus, et sel-lest paremini aru saada ning, et oleks olemas lähtepunkt algoritmi edasiarendamiseks.

Üks tähtsaim osa algoritmist on eeljaotuste loomine. Algoritmi kirjeldus on peatükis 2.10. Eeljaotuste loomiseks on vaja ühte vaatluste vektorit, millest lähtuvalt eeljaotusi genereeritakse. Eeljaotustesse valitakse täpselt sama palju andmepunkte, kui on vaatluses. Kui eeljaotuses on kõik punktid valitud, siis moodustatakse nende põhjal kõver punktide interpoleerimise teel. Eeljaotusi luuakse niipalju kui argumendina ette määratakse. Mida rohkem eeljaotusi luuakse, seda täpsem on lõpptulemus, kuid samas algoritmi töö aeglustub kuna eeljaotuse loomine on ajakulukas.

Teine oluline osa on ühe eeljaotuse tõepära arvutamine mingi genereeritud eeljaotuse vektori $\boldsymbol{p}$ ja vaatluste vektori $\boldsymbol{y}$ põhjal. Tõepära l arvutatakse järgmise valemiga:

$$l(p_1, ..., p_n) = \prod p_i^{y_i}(1 - p_i)^{1-y_i}$$

Lühidalt on tõepära korrutis üle kõikide eeljaotuse elementide, kus korrutatavateks on p_i kui $y_i = 1$ ja $(1 - p_i)$ kui $y_i = 0$.

Viimane oluline osa on vaatluste põhjal kalibreeritud $(c_1, ..., c_n)$ väärtuste tagastamine ehk kalibreerimisfunktsiooni arvutamine. See käib järgmise valemiga:

$$(c_1, ..., c_n) = \frac{\sum l(p_1, ..., p_n) \cdot (p_1, ..., p_n)}{\sum l(p_1, ..., p_n)}$$

3.2 Tõepära piirang eeljaotuste leidmisel

Algoritmi töös on kõige ajakulukam osa eeljaotuste loomine. Suur osa ajast võib kuluda selliste eeljaotuste genereerimise peale, mille tõepära vaatluste põhjal on nullilähedane. Need ei mõjuta eriti vastust, kuna neile tekib hiljem väga väike kaal. Siit tuleb esimene optimeerimise samm – tuleb genereerida vaid selliseid eeljaotusi, mis ületavad mingi etteantud tõepära piiri t.

Nüüd lähendame järeljaotuse keskväärtust järgmiselt

$$E[(P_1, ..., P_n|y_1, ..., y_n)] \approx E[(P_1, ..., P_n|y_1, ..., y_n, l(P_1, ..., P_n) \geq t],$$

kus $l(P_1, ..., P_n)$ on jaotuse $(P_1, ..., P_n)$ tõepära.

Nimetame lävendi ületamise ümber sündmuse L toimumiseks. Mugavamaks lugemiseks lühendame eeljaotuse vektori $\boldsymbol{P} = (P_1, ..., P_n)$ ja vaatluste vektori $Y = (y_1, ..., y_n)$. Hindame nüüd järeljaotuse keskväärtust, kui toimub sündmus L:

$$E[(\boldsymbol{P}|Y, L)] = \int \boldsymbol{p} f_{P|Y,L}(\boldsymbol{p}) d\boldsymbol{p},$$

kus $f_{P|Y,L} = \frac{P_{Y|P,L} f_{P|L}}{z}$

Nüüd tahame esialgse generatiivse protsessi $f_{P|L}$ välja vahetada sellise generatiivse protsessi vastu, mis tagab, et sündmus L toimub - tähistame seda $f'_{P|L}$. Järgmises peatükis kirjeldame, kuidas uus generatiivne protsess töötab.

3.3 Tõepära piirang implementeeritud p_i valikuvahemiku tõkestamisega

Järgnevalt kirjeldatakse, kuidas tõepära piirang algoritmis realiseeriti.

Kõige suurema tõepäraga eeljaotus vaatluse jaoks on isotooniline regressioon vaatluspunktide põhjal [4]. Sellest tulenevalt võetigi ületatava tõepära piiriks isotoonilise regressiooni tõepära jagatud konstandiga, näiteks 1000-ga. Tähistame isotoonilise regressiooni tõepära lh_{max}, siis on piir $t = \frac{lh_{max}}{1000}$. Väikseim väärtus t jaoks on 0, siis on kõik kõverad lubatud. Suurim läve väärtus on isotoonilise regressiooni väljastatud tõepära, sellest suurema tõepäraga jaotusi pole võimalik genereerida.

Piiri kasutatakse eeljaotuse loomisel iga p_i genereerimisel. Kui algoritmi optimeerimata versioonis oli p_i leidmisel valikuvahemik rekursioonist kaasa tulnud alg- ja lõppväärtuse vaheline piirkond, siis nüüd vajaduse korral seda korrigeeritakse ehk valikuvahemikku tõkestatakse. Seda on vaja teha, sest kui valida punkt valikuvahemiku äärest, siis võib see sel hetkel genereeritava eeljaotuse lükata vaatlusest liiga kaugele ning seetõttu võib eeljaotus olla liiga väikese tõepäraga.

Alumise tõkke leidmiseks proovitakse p_i väärtuseks valida valikuvahemiku madalaim punkt ning seejärel arvutatakse hüpoteetilise jaotuse tõepära. Hüpoteetiline jaotus koosneb juba eelnevalt valitud punktidest ning kohtades, kus veel y-väärtust valitud ei ole, seatakse väärtuseks isotoonilise regressiooni poolt väljastatud väärtus sellel kohal. Nüüd kui jaotuse tõepära ületab piiri, siis jääbki alumiseks tõkkeks vahemiku algpunkt. Kui piiri ei ületatud, siis liigutakse mingi määratud ühiku jagu vahemikus edasi ning proovitakse punktiks valida see väärtus. Eelmist protsessi korratakse kuni leitakse punkt, mille korral tõepära ületab piiri ning alumiseks

tõkkeks saab viimati sobinud punkt. Analoogiline protsess läbitakse ülemise tõkke leidmisel, nüüd liigutakse lihtsalt valikuvahemiku kõrgeimaist punktist allapoole.

Tõkete kasutamisel ei raisata enam ressursse ebatõenäoliste eeljaotuste genereerimiseks, vaid leitakse rohkem suurema tõepäraga eeljaotusi. Seetõttu paraneb eeljaotuste arvu samaks jätmine algoritmi täpsust ning ilma tõketeta versioonile sarnase täpsuse saab kätte nüüd väiksema eeljaotuste arvuga.

3.4 Uus generatiivne protsess $f'_{P|L}$

Uus generatiivne protsess $f'_{P|L}$ loob vaid selliseid eeljaotusi, mille puhul toimub sündmus L ehk jaotuste tõepärad ületavad tõepära läve t.

$$\int P\frac{P_{Y|\boldsymbol{P},L}(\boldsymbol{P})\cdot f_{\boldsymbol{P}|L}(\boldsymbol{P})}{z}dP \approx \int P\frac{P_{Y|\boldsymbol{P}}(\boldsymbol{P})\cdot f_{\boldsymbol{P}|L}(\boldsymbol{P})}{z}dP =$$
$$= \int P\cdot\frac{l(\boldsymbol{P})}{z}\cdot\frac{f_{\boldsymbol{P}|L}(\boldsymbol{P})}{f'_{\boldsymbol{P}|L}(\boldsymbol{P})}\cdot f'_{\boldsymbol{P}|L}(\boldsymbol{P})dP = \frac{\sum w_i\cdot(p_1^{(i)},...,p_n^{(i)})}{\sum w_i}, \quad (3)$$

kus $w_i = l(p_1^{(i)},...,p_n^{(i)})\cdot\frac{f_{\boldsymbol{P}|L}}{f'_{\boldsymbol{P}|L}}$

Teisisõnu on ühe eeljaotuse kaaluks nüüd tema tõepära korrutatuna vana ja uue generatiivse protsessi suhtega. Algoritmi lõpptulemuseks on mingi suure arvu eeljaotuste kaalutud keskmine.

3.5 Eeljaotuste kaalud

Juhul kui y väärtuse valikuvahemikul rakenduvad eelmises peatükis kirjeldatud tõkked, siis tuleb arvesse võtta, et juhuslikult valitud arvu ei saanud valida täies

ulatuses, kitsendasime sellega juhuslikkust. Selle jaoks otsustasime kasutada kaalu kogu eeljaotuse jaoks, et vähendada selliste eeljaotuste tähtsust, mille puhul pidi valikuvahemikke kitsendama.

Eeljaotuse loomise alustamisel on ta kaal 1. Seejärel hakatakse iga punkti valikul korrutama kaalu läbi kitsendatud valikuvahemiku ja täieliku valikuvahemiku suhtega $\frac{y_{c2}-y_{c1}}{y_2-y_1}$, kus y_{c1} ja y_{c2} on vastavalt tõkestatud valikuvahemiku minimaalne ja maksimaalne väärtus ja y_1 ja y_2 esialgse tõkestamata vahemiku minimaalne ja maksimaalne väärtus.

3.6 Dünaamiline piir tõkete jaoks

Algoritmi optimeerimise käigus testiti tõkete jaoks kasutatava tõepära piiri jaoks mitmeid erinevaid väärtusi. Katsete käigus selgus, et piiriga väärtusega ühte äärmusesse minnes lähevad kaalud balansist välja - kui järjestada kõikide eeljaotuste kaalud, siis suurima kaalu osakaal üle kõikide teiste kaalude on ebaproportsionaalselt suur, tihti kümnete või sadade astmete jagu. Sellisel juhul hakkab suurima kaaluga eeljaotus teiste eeljaotuste üle domineerima, võib juhtuda, et kalibreerimine teostataksegi sisuliselt ainult ühe suurima kaaluga eeljaotuse põhjal. Samas kui piiri väärtusega liikuda liiga kaugele teisele poole, siis juhtub sama asi eeljaotuste tõepäradega ehk suurima tõepäraga eeljaotuse tõepära on tunduvalt suurem, kui teiste eeljaotuste tõepärad.

Probleemi lahendamiseks otsustati tagada tasakaal eeljaotuste kaalude suhete ja tõepärade suhete vahel. Selleks kontrolliti peale iga 100 eeljaotuse loomist suurima kaalu osakaalu teiste kaalude seas ning suurima tõepära osakaalu ülejäänud

tõepärade seas ning leiti nende suhe r:

$$r = \frac{\frac{w_{max}}{\sum_{i=1}^{n} w_i}}{\frac{lh_{max}}{\sum_{i=1}^{n} lh_i}} \tag{4}$$

Kui suhe r on suurem, kui mingi konstant (vaikimisi 2), siis nihutatakse piiri väärtust. Kui suhe on liiga suur ja suurim tõepära on ebaproportsionaalselt suur, siis muudetakse lh_{max}-ga jagatavat arvu 10 korda väiksemaks. Vastupidisel juhul kui suurim kaal on liiga palju suurem kui ülejäänud kaalud, siis muudetakse jagatavat arvu 10 korda suuremaks.

Nüüd on tagatud see, et ükski eeljaotus ei ole kalibreerimisel teistest eeljaotustest ebaproportsionaalselt olulisem.

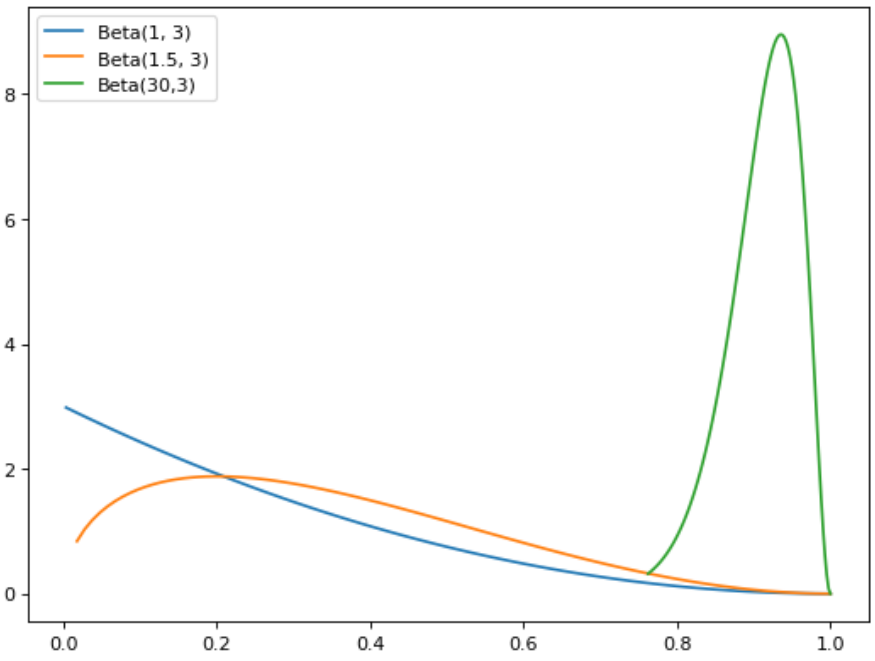
4 Katsed

Selles osas analüüsitakse täiendatud Bayesi isotoonilist kalibreerimist ning võrreldakse seda teiste levinud kalibreerimisalgoritmidega.

4.1 Andmed

Katsete jaoks kasutati tehislikke andmeid. Tehislike andmete peal on mugav kalibreerimismeetodeid võrrelda, sest sel juhul saab välja arvutada õige kalibreerimiskõvera, mille vastu saab erinevate meetodite väljundeid võrrelda.

Andmepunktide loomiseks kasutati beeta-jaotusi *scipy.stats* teegist. Kokku tehti kolm katset, 100, 1000 ja 10000 treeningpunktiga. Treeningpunktidest tähistasid pooled punktid negatiivset (0) väärtust ja pooled positiivset (1) väärtust. Negatiivsed andmepunktid valiti juhuslikult beeta-jaotusest argumentidega $\alpha = 1, \beta = 3$, pooled positiivsed andmepunktid valiti jaotusest argumentidega $\alpha = 1.5, \beta = 3$ ning ülejäänud pooled jaotusest argumentidega $\alpha = 30, \beta = 3$ (vt joonis 9).



Joonis 9: Beta jaotused

Treeningandmed sorteeriti kasvavasse järjekorda ning seejärel treeniti nende põhjal järgmised kalibreerimisalgoritmid: Bayesi isotooniline kalibreerimine, isotooniline kalibreerimine, logistiline regressioon. Viimased kaks valiti võrdluseks, kuna need on ühed levinumad kalibreerimisalgoritmid. Lisaks arvutati välja ka perfektne kalibreerimisfunktsioon:

$$f_p(x) = \frac{pos(x)}{pos(x) + neg(x)},$$

kus

$$pos(x) = 0.5 \cdot f_B(x, 1.5, 3) + 0.5 \cdot f_B(x, 30, 3)$$

$$neg(x) = f_B(x, 1, 3),$$

kus $f_B(x, \alpha, \beta)$ on beeta-jaotuse tihedusfunktsioon.

Samadest beeta-jaotustest loodi ka 10 000 testpunkti, mille põhjal ennustati treenitud mudelitega kalibreeritud väärtusi. Mudelite ennustuste põhjal arvutatud logaritmilised kaod on kujutatud tabelis 1.

Tabel 1: Kalibreerimisalgoritmide logaritmiline kadu erinevate andmehulkade korral

Kalibreerimisalgoritm	100 punkti	1000 punkti	10000 punkti
Bayes-Iso kalibreerimine	0.5120	0.4776	0.4736
Isotooniline kalibreerimine	1.4643	0.5433	0.4816
Logistiline regressioon	0.5098	0.5115	0.5095

Analüüsime tulemusi lähemalt ja võrdleme isotoonilist kalibreerimist ja logistilist regressiooni Bayesi isotoonilise kalibreerimisega eraldi.

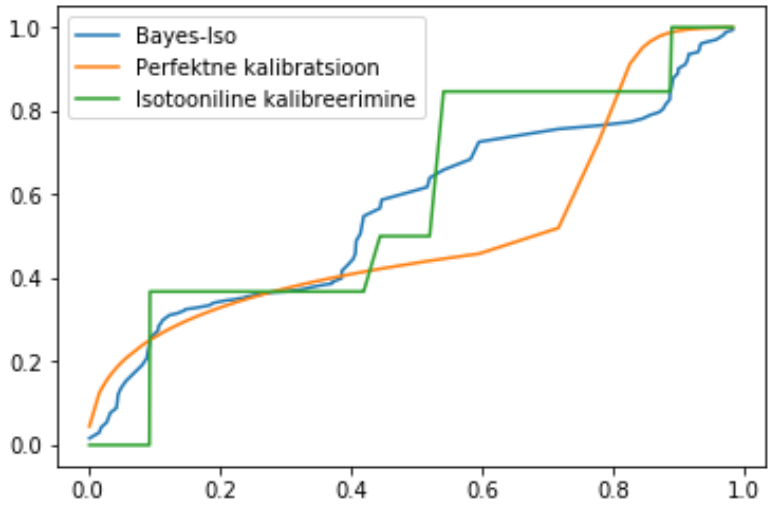
4.2 Võrdlus isotoonilise kalibreerimisega

Isotooniline kalibreerimine on üks levinumaid algoritme kahendklassifitseerijate kalibreerimiseks. Lisaks oli Bayesi isotoonilise kalibreerimise loomine sellest inspireeritud [2], seega tundub sobilik neid omavahel võrrelda.

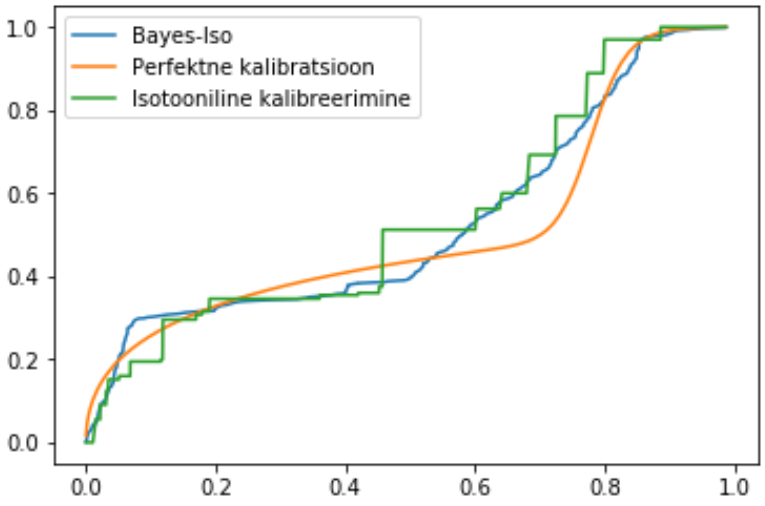
Vaadates katsete tulemusi tabelist 1 näeme, et Bayesi isotooniline kalibreerimine oli kõikide andmehulkade korral parem kui isotooniline kalibreerimine. Märkimisväärne on, et väga suur vahe oli väikse andmehulga korral ehk 100 andmepunktiga. Sel juhul oli Bayesi isotoonilise kalibreerimise logaritmiline kadu 0.5120 ja isotoonilise kalibreerimise oma 1.4643. Mida suuremaks testandmehulk läks, seda võrdsemaks muutusid algoritmide sooritused.

Joonistel 10, 11 ja 12 on kujutatud perfektne kalibreerimiskõver, Bayesi isotoonilise

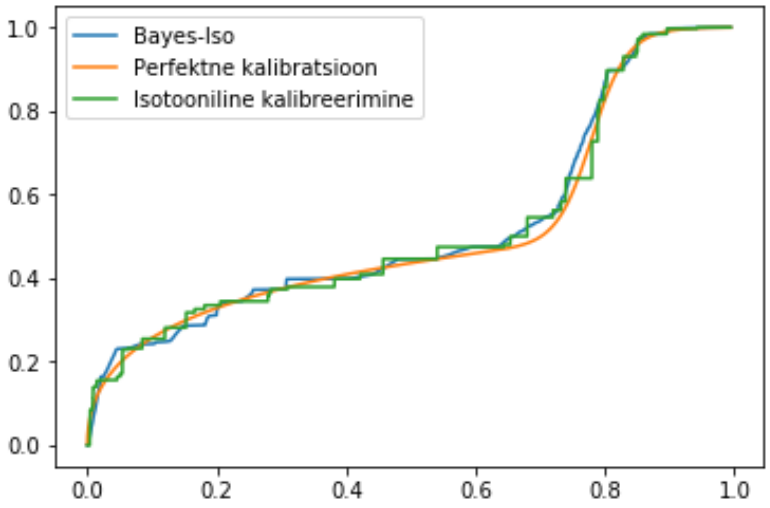
kalibreerimise kõver ning isotoonilise kalibreerimise kõver erinevate testandmehulkade korral. Joonistelt on näha, et isotoonilise kalibreerimise kõver on väga sakiline ehk hüpped ennustatavate väärtuste vahel on suured. Bayesi isotoonilise kalibreerimise kõver on seevastu palju siledam, selle tõttu võidetakse mitmes kohas täpsust juurde.



Joonis 10: Võrdlus isotoonilise kalibreerimisega 100 andmepunktiga



Joonis 11: Võrdlus isotoonilise kalibreerimisega 1000 andmepunktiga



Joonis 12: Võrdlus isotoonilise kalibreerimisega 10000 andmepunktiga

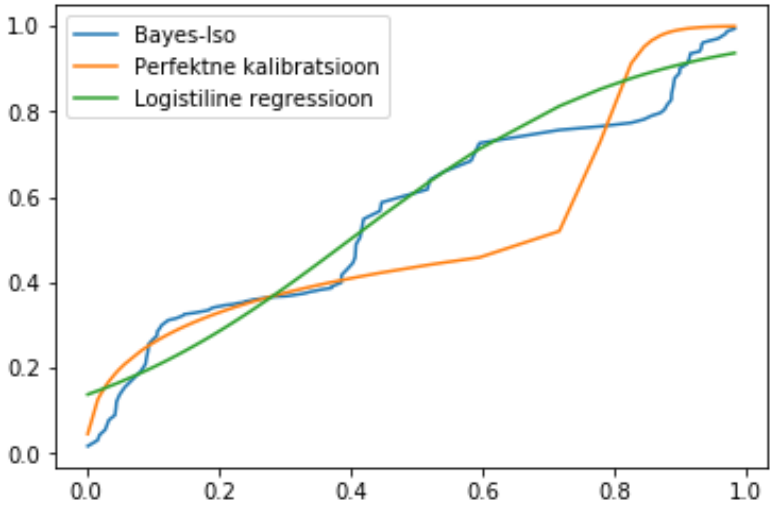
4.3 Võrdlus logistilise regressiooniga

Logistiline kalibreerimine on samuti levinud klassifitseerija kalibreerimise meetod. Tabelist 1 selgub, et Bayesi isotoonilise kalibreerimise ja logistilise regressiooni sooritus ei erine suuresti, kuid 1000 ja 10000 testpunkti korral on esimene algoritm siiski parem. 100 testpunkti korral on logaritmiline kadu peaaegu sama, kuid logistiline regressioon edastab veidi Bayesi isotoonilist kalibreerimist.

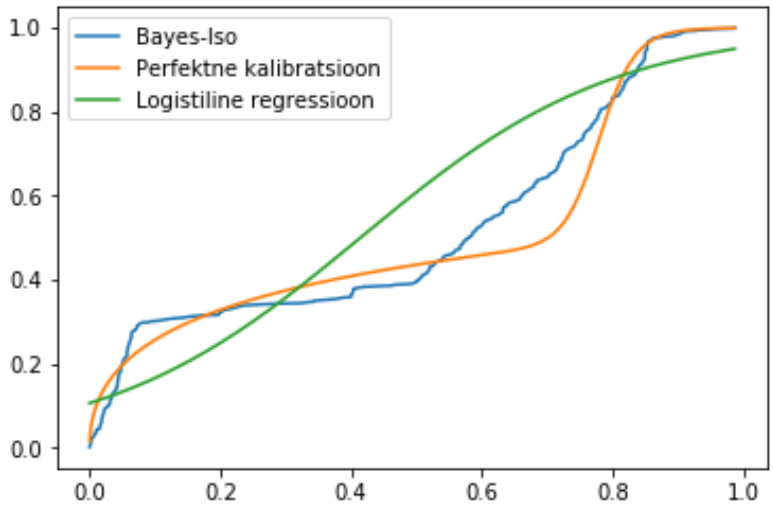
Joonistel 13, 14 ja 15 on kujutatud perfektne kalibreerimiskõver, Bayesi isotoonilise kalibreerimise kõver ning logistilise regressiooni kõver testandmetel.

Jooniselt 13 selgub, miks Bayesi isotooniline kalibreerimine tulemustega alla jäi. X teljel väärtuse 0.5 ümbruses teeb kalibreerimiskõver suure hüppe ja jääb perfektsest kalibreerimiskõverast liiga kaugele. Joonistel 14 ja 15 on aga Bayesi isotoonilise kalibreerimise kõver õigele jaotusele palju lähemal ning seetõttu saab ka parema tulemuse.

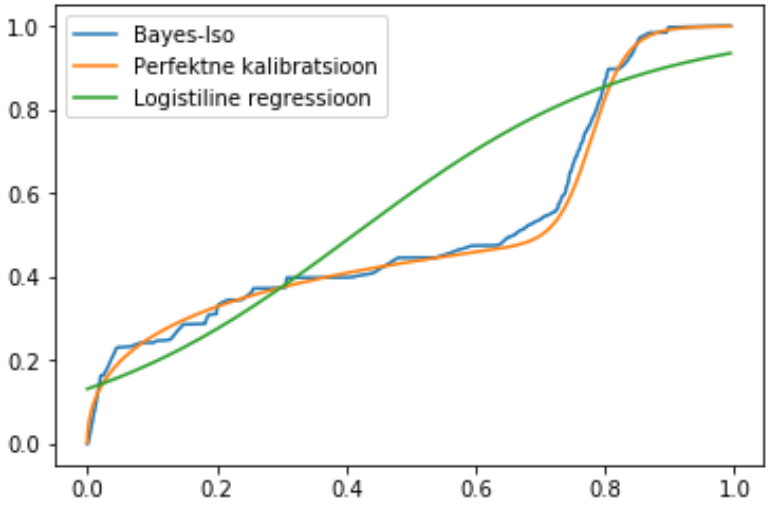
Kõikide jooniste pealt jääb aga silma, et logistilise regressiooni kõver on kohati perfektsest kalibreerimiskõverast kaugel ja ta kuju ei muutu andmehulga muutumisel. See juhtub seetõttu, et perfektne kalibreerimiskõver on vastupidise sigmoidi kujuga ning ta ei kuulu kalibreerimiskõverate logistilisse perekonda [8]. Kuna kõverat logistilises perekonnas ei ole, siis ei sobita logistiline regressioon isegi palju rohkemate andmetega tõesele kõverale lähemat kalibreerimiskõverat.



Joonis 13: Võrdlus logistilise regressiooniga 100 andmepunktiga



Joonis 14: Võrdlus logistilise regressiooniga 1000 andmepunktiga



Joonis 15: Võrdlus logistilise regressiooniga 10000 andmepunktiga

5 Kokkuvõte

Käesolevas töös kirjeldati detailselt Mari-Liis Allikivi ja Meelis Kulli loodud Bayesi isotoonilise kalibreerimise algoritmi ning optimeeriti seda. Algoritm vajas optimeerimist, kuna see muutus suurte andmehulkade korral ebastabiilseks ja aeglaseks.

Töös seletati lahti olulised põhimõtted, mida kasutatakse Bayesi isotoonilise kalibreerimise algoritmis. Nendeks on isotooniline kalibreerimine, Bayesi lähenemine ja Monte Carlo meetodid. Nendest paremini aru saamiseks kirjeldati näiteülesanded, mis kujutasid endast eeljaotuste ja järeljaotuste keskväärtuste hindamist. Näidati, kuidas saab neid ülesandeid lihstamatel juhtudel analüütiliselt lahendada ning kuidas Bayesi lähenemine ja lihtne Monte Carlo meetod aitavad ülesannetele lähen-did leida.

Töö käigus implementeeriti esialgu optimeerimata versioon Bayesi isotoonilisest kalibreerimisest ning seejärel täiendati seda erinevate optimeeringutega, mis muutsid algoritmi kiiremaks ning suurendasid arvutusjõudluse samaks jätmisel selle täpsust.

Täiendatud Bayesi isotoonilise kalibreerimisega viidi läbi katsed tehisandmetel. Võrdluseks tehti sama ka isotoonilise kalibreerimisega ja logistilise regressiooniga. Katsete tulemused näitasid, et enamus juhtudel on Bayesi isotooniline kalibreerimine teistest algoritmidest täpsem.

Edasi oleks võimalik uurida Bayesi isotoonilise kalibreerimise täpsust reaalsetel andmetel ning võrrelda seda rohkemate kalibreerimisalgoritmidega.

Viidatud kirjandus

[1] Martin Gebel ja Claus Weihs. Calibrating classifier scores into probabilities. In Reinhold Decker and Hans J. Lenz, editors, *Advances in Data Analysis*, pages 141–148, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[2] Mari-Liis Allikivi ja Meelis Kull. Non-parametric bayesian isotonic calibration: Fighting over-confidence in binary classification. (Avaldamisel), 4 2019.

[3] Peter Flach. *Machine Learning: The Art and Science of Algorithms That Make Sense of Data*. Cambridge University Press, New York, NY, USA, 2012.

[4] Bianca Zadrozny ja Charles Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02, pages 694–699, New York, NY, USA, 2002. ACM.

[5] George Box ja George Tiao. *Bayesian inference in statistical analysis*. Addison-Wesley Pub. Co, Reading, Massachusetts, 1973.

[6] Sheldon M. Ross. *Introduction to Probability Models, Ninth Edition*. Academic Press, Inc., Orlando, FL, USA, 2006.

[7] Tõnu Kollo. *Monte Carlo meetodid*. Tartu Ülikooli Kirjastus, 2004.

[8] Meelis Kull ja Telmo Silva Filho ja Peter Flach. Beta calibration: a well-founded and easily implemented improvement on logistic calibration for binary classifiers. pages 623–631, 01 2017.

Lisad

I. Koodi repositoorium

Lingil https://github.com/karlriis/bayes-iso asub repositoorium, mis sisaldab Bayesi isotoonilise kalibreerimise koodi.

II. Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, **Karl Riis**,

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose

   Bayesi isotoonilise kalibreerimise algoritm ja selle optimeerimine,

   mille juhendaja on Meelis Kull,

   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.

Karl Riis

10.05.2019