

VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
MATEMATINĖS STATISTIKOS KATEDRA

MAGISTRO DARBAS

KONTROLĖS PROCEDŪRŲ PARINKIMAS

Magistrantė INGA PILECKAJA

Darbo vadovas doc. P. VAITKUS

Vilnius

2006

1. Įvadas	4
2. Liejamojo formavimo įrenginys	7
2.1.1. Aprašymas.....	7
2.1.2. Funkcinių galimybių apžvalga.....	7
2.2.1. Ekranų apžvalga.....	9
2.3.1. SPV koncepcija.....	10
2.3.2. Kintamųjų kontrolė.....	11
2.3.3. Įrenginio ciklai ir atranka.....	11
2.3.4. Tendo ekranas ir SPV.....	12
2.3.5. Proceso kintamumas.....	13
2.3.6. Specifiškumo ribos ir standartai.....	14
2.3.7. Kokybės kontrolės koeficientas (Cpk).....	15
2.3.8. Valdymo diagramos.....	17
2.3.9. Avariniai SPV pranešimai.....	20
3. Logistinė regresija	21
3.1. Matematinis modelis.....	21
3.2. Logistinės regresijos statistiniai uždaviniai.....	24
3.3. Klasifikavimo tikslumas.....	28
4. BMPM (Biased Minimax probability Machine)	31
4.1. Įvadas į BMPM.....	31
4.2. Tiesinė optimalių pastumtų tikimybių sprendinių hiperplokštuma.....	32
4.3. Branduolio metodas.....	37
4.4. Pabaiga.....	41
5. Pracedūrų bandymai ir rezultatai	42
6. Išvados	51
Summary	52
Literatūros sąrašas	53
Priedas A	55
SAS'o skaičiavimai.....	55
Priedas B	63
ROC kreivės.....	63
Priedas C	65
MATLAB'o skaičiavimai.....	65

Priedas D	67
MATLAB pagrindinė programa.....	67

1. Įvadas

Produkcijos kokybės lygio reguliavimas ir gerinimas – vienas iš svarbiausių kiekvienos šiuolaikinės įmonės uždavinių, kad išleidžiama produkcija būtų konkurencinga pasaulinėje rinkoje.

Šiuolaikinėse modernių technologijų įmonėse saugomi ir nuolat papildomi gaminamos produkcijos ir technologinių procesų bankai. Dažnai duomenys nuskaitomi ir įvedami į kompiuterio atmintį automatizuotu būdu. Naudojant kompiuterinius tinklus ir duomenų bazių valdymo bei analizės sistemas, informacija realiu laiko masteliu gali būti perduodama ir panaudojama priimant sprendimus kiekviename tarpiniame bei baigiamosios kontrolės punkte. Taigi atsiranda galimybė priimant sprendimus naudoti informaciją ne tik iš pastarojo, bet ir iš ankstesnių kontrolės punktų. Tai gali labai pagerinti priimamų sprendimų tikslumą.

UAB “**NB Europe**” yra moderni ir sparčiai besiplečianti gamybos įmonė. Kompanija buvo įkurta 1993 metais kaip TŪB “Nemuno banga” ir užsiėmė cigarečių filtrų gamyba. PET preformų gamyba buvo pradėta 1997 metais, kai plastikinė pakuotė ėmė sparčiai populiarėti vietinėje rinkoje. 1998 metais kompanija tapo pagrindine tiekėja didžiausiems pasaulio gaiviųjų gėrimų ir alaus gamintojams Rytų Europoje. Nuosavų krovinių automobilių parkas padėjo praplėsti teikiamų paslaugų įvairovę ir užtikrino nepriekaištingą įsipareigojimų vykdymą. 2002 m. Lentvaryje pradėjo veikti PET butelių pūtimo cechasis. PET preformų gamyklos Melnike (Čekija) atidarymas 2004 metais buvo sparčios ir užtikrintos verslo plėtros rezultatas. Po kompanijos reorganizavimo 2003 metais **NB Europe** nebegamina cigarečių filtrų, tačiau sparčiai didina PET preformų gamybos bei PET butelių pūtimo apimtis. Šiuo metu “Nemuno bangos” grupė yra viena pirmaujančių PET preformų gamintojų Europoje, o tiekėjų ir klientų tarpe yra vertinama kaip patikimas ir lankstus verslo partneris.

VEIKLA: NB Europe yra PET pakuočių gamyboje savo veiklą koncentruojanti kompanija, kurios gamyklos veikia Lentvaryje, Klaipėdoje ir Melnike (Čekija). Daugiau kaip 65 proc. **NB Europe** užsakovų – tarptautinių prekės ženklų alaus, vaisvandenių ir aliejaus gamintojai. Apie 10 proc. produkcijos parduodama Lietuvoje. 2004 metais **NB Europe** užėmė daugiau nei 3 proc. Europos PET pakuočių rinkos. Siekdama patenkinti rinkos poreikius, **NB Europe** turi platų spektrą įvairių svorių ir skirtingo dizaino preformų.

Klientui yra siūlomos įvairiausios preformų ir butelių kaklelio, svorio ir spalvos modifikacijos:

- nealkoholiniams gėrimams (gazuotiems ir negazuotiems);
- mineraliniam vandeniui (gazuotam ir negazuotam);
- alui;
- aliejui;
- kitiems kliento pasirinktiems produktams.

PET PREFORMŲ DIAPAZONAS:

- **PCO** – 18,5g, 20g, 20,5g, 24g, 26.5g, 28g, 32g, 34g, 36g, 39g, 40g, 41g, 43g, 44g, 47g, 48g, 50g, 52g, 54g.
- **PCO daugiasluoksnis** – 26,5g, 28g.
- **BPF** – 42g, 47g, 48g.
- **Semi PCO** – 21g, 23g, 34,5g, 36g, 38g, 39g, 40g, 41,5g.
- **Ø 38 mm** – 18,5g, 21.5g, 28g, 35g.
- **Ø 38 mm Affaba Ferarri** – 21,5g, 24g, 24,5g, 28g.
- **OIL** – 18g, 21g, 22g.
- **Ø 44,7 mm** – 85 g, 145g.
- **Ø 60,0 mm** – 30g, 36g.

Šiame darbe bus naudojami UAB “NB Europe” duomenys, gauti gaminant buteliukų ruošinius liejamojo formavimo įrenginiu. Juos sudaro 37 kintamieji. Viena pagrindinių problemų, su kuria susiduria kokybės kontrolės skyriaus viršininkas, - kontroliuoti gaminių kokybę, kad būtų kuo mažiau reklamacijų. Bandymus jie atlieka iškart gamybiniu įrenginiu su 32 lizdais; eksperimentiniam įrenginiui įsigyti jie pinigų skirti nenori, nes mano, kad tai jiems nenaudinga, kadangi reikia ilgo laiko stebėjimų, kad pastebėtų sąryšius tarp eksperimentinio įrenginio ir masinės gamybos įrenginio gautų jau gerų gaminių ribų, nes vis tiek jos skirsis.

Gamyba vykdoma ištisai. Gaminių kokybės kontrolė atliekama kas tam tikrą laiką, dažniausiai kas pusvalandį, atrenkant iš didelės aibės kelis ruošinius. Visų gaminių kontrolė atliekama kartą per savaitę. Rezultatai užrašomi išmatavimų pavidalu, t.y. kokius gavo matavimus, tokius ir užrašo. Visų pagamintų ruošinių parametrus fiksuoja ir pats liejamojo formavimo įrenginys. Fiksavimai vyksta maždaug kas 15s. Taip pat jie atvaizduojami ekranuose grafikų pavidalu.

Turėdama liejamojo formavimo įrenginio duomenis iš įmonės ir pasitelkus į pagalbą matematiką, šiame darbe pabandyčiau palengvinti šios įmonės problemų sprendimą. Iš pradžių aprašysiu patį liejamojo formavimo įrenginį ir jo vykdomas funkcijas.

Kokybės kontrolei naudosiu logistinę regresiją ir naują metodą BMPM, pasiūlytą G.R. G. Lanckrieto ir išplėtotą Kaizhu Huang ir kitų mokslininkų disertacijoje “**Learning From Data Locally and Globally**”. Taip pat skaičiavimams atlikti pasinaudosiu jų sukurta programa, kuri vadinasi BMPM-1.0.

2. Liejamojo formavimo įrenginys

2.1.1 Aprašymas

2.1.1.1 Statistinis proceso valdymas (toliau vadinamas SPV) yra instrumentas, kuris procesų valdymui naudoja fundamentaliąją statistiką. SPV naudojimo privalumas yra tas, kad pagerėja kokybė ir diagnozavimas.

2.1.1.2 Kompanijos Chaski SPV funkcija yra instrumentas, kuris aprūpina "langu", tam, kad galima būtų pažvelgti į gamybos procesą. SPV proceso realizavimosi pranašumas tas, kad SPV informacija naudojama iškylančių problemų ieškojimui ir koregavimui.

2.1.2 Funkcinių galimybių apžvalga

2.1.1.3 SPV kontroliuoja gamybinį procesą ir nustato tuos atvejus, kada jis nukrypsta nuo standartinio technologinio proceso arba kada procesas daugiau nėra statistinės kontrolės būsenoje.

2.1.1.4 Procesas nuo standartinio nukrypsta tuo atveju, kai vienas ar daugiau kontroliuojamų SPV kintamų dydžių nėra nustatytame darbiname režime. Kiekvienam kintamam dydžiui operatorius gali nustatyti vieną ar daugiau sekančių veiksmų, kuriuos reikia padaryti, jeigu kintamas dydis yra už darbinio diapozono:

- a) įjungti SPV avarinės signalizacijos lemputę;
- b) sustabdyti liejamojo formavimo mašiną (įrenginį);
- c) įjungti papildomą išėjimą.

2.1.1.5 SPV programa gali ištirti standartinį technologinį procesą, grindžiamą liejamojo formavimo mašinos einamosiomis techninėmis charakteristikomis, kai operatorius nustato, kad mašina gamina gerus gaminius.

2.1.1.6 Operatorius gali :

- a) Išsirinkti iki 20 SPV kintamų dydžių, kurie turėtų būti kontroliuojami iš sąrašo, iki 60 kintamų proceso dydžių ir 6 kintamus gaminio dydžius.
- b) Peržiūrėti SPV kontroliuojamų kintamų proceso dydžių suvestinę, jų techninius apribojimus, jų einamąją gamybą procese (Cpk), o taip pat jų statistinės kontrolės būseną.

- c) "Išmokti" proceso specifiškumo(standarto) apribojimus, jeigu duoto proceso (gaminio) išėjimas perėjo kokybės tikrinimus.
- d) Peržiūrėti tendą, trunkantį 400 ciklų kiekvienam kintamam CPV kontroliuojamam dydžiui. Duotas trendas taip pat atvaizduoja vidutinį proceso dydį, "išsibarstymą", specifiškumo ribas ir Cpk.
- e) Peržiūrėti vidutinius proceso parodymus (vidutinių reikšmių diagramą), proceso pasikeitimus (R-diagramą) ir einamąsias valdymo ribas.
- f) Nustatyti bet kuriam kontroliuojamam kintamam dydžiui iš vidutinių reikšmių diagramos ir R-diagramos fiksuotas valdymo ribas.
- g) Statistinio valdymo tikrinimo pagalba gauti perspėjimą apie proceso pasikeitimus vidutinių reikšmių diagramoje (gaminio pavyzdžiai už valdymo ribų, trendo pasikeitimai, besikeičiantys duomenys).
- h) Kiekvienam SPV kontroliuojamam kintamam dydžiui nustatyti avarinį pranešimą ir veiksmus, kurie turėtų būti vykdomi, kai kintamasis yra avarinėje būklėje.
- i) Vienu metu visiems SPV kintamiems dydžiams peržiūrėti tendą vidutinių reikšmių diagramoje arba R/s diagramoje.
- j) Kiekvienam SPV kintamam dydžiui peržiūrėti tendą vidutinių reikšmių diagramoje arba R/s-informaciją iš paskutinių 10 000 ciklų. Šitie duomenys atvaizduojami kartu su įvykiais, parodančiais mašinos darbo režimą ir nustatytų parametrų pasikeitimus.
- k) Išrinkti kryptį, pagal kurią sudaromos įvairios SPV diagramos.
- l) Įvesti duomenis pagal gaminio kokybę, tokius kaip svoriai ar dydžiai, ir vykdyti tarp duomenų techninio proceso ir gaminio kokybės duomenų koreliaciją.
- m) Gamybiniame uždavinyje skaičiuoti gaminio defektus 12-koje defektų kategorijų.
- n) Peržiūrėti perspėjimų ir avarinių SPV pranešimų archyvo žurnalą.
- o) Persiųsti technologinio proceso duomenis ir gaminio kokybės duomenis į lanksčiuosius diskelius autonominei analizei.

2.2.1 Ekranų apžvalga

2.2.1.1.SPV programų paketas susidaro interfeise iš devynių ekranų. Priėjimas prie ekranų komplekto vykdomas mygtuko paspaudimu SPV

suvestinėje, apžvalgos meniu, o po to perėjimui nuo vieno ekrano prie kito naudojami viršutinės eilės mygtukai. Yra tokie ekranai:

- a) SPV suvestinės ekranas: šiame ekrane vaizduojama einamosios būsenos 20 SPV kontroliuojamų kintamų dydžių suvestinė. Jis parodo kiekvieno kintamo dydžio einamąją reikšmę, specifiškumo ribas ir avarinę būklę.
- b) SPV smulkios informacijos ekranas: šiame ekrane pavaizduojama vieno kontroliuojamo kintamo dydžio smulki informacija apie jo būklę. Jame parodyti trendiniai duomenys ir kintamojo valdymo diagramos ir pateikti mygtukai kintamo dydžio naujam nustatymui arba automatiškam specifiškumo ribų nustatymui.
- c) SPV sugrupuotas ekranas avarijos meniu: šiame ekrane vienu metu gali būti parodyti trendas, vidutinių reikšmių diagrama arba R/s-diagramos 20-čiai SPV kontroliuojamų kintamų dydžių.
- d) SPV archyvo ekranas: šiame ekrane gali būti atvaizduoti archyviniai trendas, diagrama bet kokiems 4-iems iš 60 arba daugiau procesų ir gaminių kokybės kintamų dydžių. Atvaizduojama informacija nuskaitoma iš duomenų žurnalo ir per vieną kartą galima peržiūrėti periodą iki 48 valandų.
- e) Rankinio duomenų įvedimo ekranas : operatorius naudoja šį ekraną, nustatydamas gaminio kokybės kintamuosius ir rankiniam jų duomenų įvedimui.
- f) Defektų įvedimo ekranas: operatorius naudoja šį ekraną rankiniam duomenų defektų įvedimui ir suminio defektų paskaičiavimo atvaizdavimui.
- g) SPV konfigūracijos ekranas: operatorius naudoja šį ekraną avarinių pranešimų konfigūracijos kontroliuojamų pranešimų nustatymams ir kitai antraeilei informacijai apie SVP programos konfigūraciją.
- h) Kintamojo išrinkimo ekranas : operatorius naudoja šį ekraną išrinkimui iki 20 SPV kintamųjų, kurie turi būti kontroliuojami iš 60 sąrašo arba daugiau procesų, o taip pat gaminių kokybės kintamų dydžių.
- i) Fiksuotų ribų įvedimo ekranas: operatorius naudoja šį ekraną įvedant fiksuotas ribas vidutinių reikšmių diagramoms arba konkrečius kontroliuojamus kintamus dydžius R/s-diagramoms.

2.3.1 SPV koncepcija

2.3.1.1. SPV programa gali aprūpinti statistinę analizę dviems kintamųjų tipams: technologinio proceso kintamiesiems ir gaminių kontrolės kintamiesiems.

2.3.1.2. Technologinio proceso kintamasis turi pasikeitimų duomenis, gautus liejamojo formavimo mašinoje po kiekvieno mašininio ciklo. Kintamieji proceso matavimai yra ciklo laikas, užspaudimo jėga arba “šneko” greitis.

2.3.1.3. Kai kuri informacija apie gaminį negali būti surinkta automatiškai (pvz., svoris arba gaminio matmenys). Gaminys turi būti ištrauktas iš mašinos ir išmatuotas, o duomenys, gauti tokiu būdu, turi būti įvedami rankiniu būdu. Šie duomenys įrašomi į kintamus gaminio kokybės dydžius.

2.3.1.4. Technologinio proceso ir gaminio kokybės kintamieji kartu vadinami SPV kintamaisiais.

2.3.2 Kintamųjų kontrolė

2.3.2.1. SPV programa gali “apdoroti” iki 60-ties gaminio kokybės proceso kintamųjų. Proceso kintamieji įvedami sistemos konfigūracijos metu ir negali būti pakeisti operatyviai.

2.3.2.2. Gaminio kintamieji nustatomi rankinio duomenų įvedimo ekrane pagalba. Gaminio kintamieji nustatomi tuo atveju, kai jiems suteiktas pavadinimas. Įvestieji kintamieji gali būti taip pat pavadinti, kaip ir sukonfigūruoti kintamieji.

2.3.2.3. SPV programa gali kontruoti iki 20-ties SPV kintamųjų, t.y. ji turi 20 laukų kintamiesiems, kiekvienas kurių išrenkamas iš sukonfigūruotų kintamųjų išrinkimo ekrano pagalba.

2.3.2.4. SPV kintamieji tiesiog nuskaitomi ir užrašomi į duomenų žurnalą. Jei kintamasis kontrolei išrenkamas kintamųjų ekrano pagalba, tai šiam kintamajam bus taikomos visos SPV funkcijos. Jo trendas bus atvaizduotas smulkios SPV informacijos ekrane, bus apskaičiuotos jo valdymo diagramos ir SPV programa gali pateikti avarinį pranešimą, jeigu šitas kintamasis "elgsis" neįprastai.

2.3.2.5. SPV kontrolei atrinktas kintamasis vadinamas atrinktuoju kintamuoju arba kontroliuojamu kintamuoju.

2.3.3 Įrenginio ciklai ir atranka

2.3.3.1. SPV programa tikrina procesų kintamuosius po kiekvieno mašininio ciklo. Vienas proceso kintamasis vadinamas parodymu. Kai kuriems paskaičiavimams programa grupuoja parodymus į imtis. Operatorius gali sukonfigūruoti imčių dydį ir dažnį.

2.3.3.2. Pavyzdžiui, imties dydis lygus 5, o imties periodas-15. SPV programa "supranta", kad reikia įtraukti į imtį 5 ciklus paeiliui, palaukti 10 ciklų, po to vėl įtraukti kitus 5 ciklus į kitą imtį.

2.3.3.3. Kitas pavyzdys: jeigu imties dydis lygus 5, o imties periodas 5, tai SPV programa į vieną imtį įtrauktų 5 ciklus paeiliui, po to kitus 5 ciklus į kitą imtį.

2.3.3.4. Kintamieji pagal gaminio kokybę grupuojami į imtį operatoriaus. Pvz.: imties dydžiui 5 operatorius gali paimti 5 gaminius iš 5 paeiliui mašininį ciklą ir užrašyti ciklą numerius, kuriuose buvo pagaminti šie gaminiai. Kiekvienam mašininiam ciklui priskiriamas numeris, kuris parodomas SPV suvestinės ekrane, smulkios informacijos SPV ekrane ir rankinio duomenų įvedimo ekrano lange. Po to operatorius turi išmatuoti šiuos gaminius ir

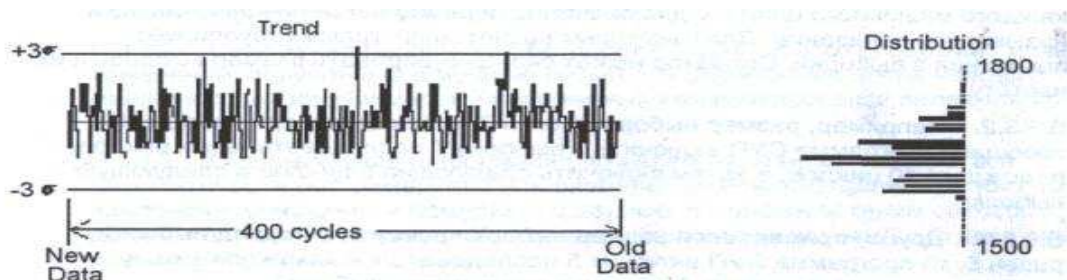
rezultatus įvesti, naudodamas rankinio duomenų įvedimo ekrano langą. Ciklo numeriai naudojami gaminio informacijos koreliacijai su atitinkama technologinio proceso informacija. Atrinkimo periodas nustatomas taip, kaip dažnai operatorius atrenka suformuotus ruošinius gamybos kokybės patikrinimo vykdymui.

2.3.4 Trendo ekranas ir SPV

2.3.4.1 Trendo diagramos SPV kintamojo reikšmes atvaizduoja grafiniu pavidalu paskutiniams 400 mašinos ciklų. Diagramos atvaizduoja parodymų “išsibarstymą” pagal ribų apskaičiavimus $+3\sigma$ ir -3σ . Išsibarstymas statistiškai parodo viršutinį ir apatinį trendo diapazoną. 99.7% parodymų turi būti tarp šių dviejų ribų. (žr. pav. B-1).

2.3.4.2. Pasiskirstymas suskaido SPV kintamųjų išmatavimų diapozoną į padiapazonius ir grafiškai atvaizduoja išmatavimų skaičių kiekviename padiapazonyje. Pasiskirstymo forma atsako į klausimus: “Kur patenka dauguma reikšmių šiame trende?”, “Sukoncentruoti jie nurodytoje juostoje ar pasiskirstę tolygiai”.

2.3.4.3. Pvz., pasižiūrėkim atkirtimo spaudimą, kuris matuojamas nuo 1500 iki 1800 svarų/kv.colyje diapazone 300svarų/kv.colyje. Rezultatų pasiskirstymas gali atrodyti kaip B-2 pav.



Trendo diagrama

Pav.B-1

Pasiskirstymo diagrama

Pav.B-2

2.3.4.4. SPV apskaičiavimų kiekis nurodo, kad kintamojo pasiskirstymas yra varpo formos, ir tai rodo normalųjį (Gauso) pasiskirstymą. Ši prielaida

korektiška daugeliu atveju ir gali būti lengvai patikrinta vizualia pasiskirstymo forma.

2.3.5 Proceso kintamumas

2.3.5.1. Kodėl SPV kintamųjų parodymai nekartoja vienas kito kiekviename mašinos cikle? Kiekvienas procesas pasireiškia kažkokiu pakitimo laipsniu. Pakitimai gali būti bendrų priežasčių arba kintamų priežasčių rezultatas.

2.3.5.2. Bendrosios priežastys negali būti pakeistos be gamybos proceso pakeitimo. Bendrų priežasčių pavyzdžiais gali būti matavimo metodai arba jų pačių, tokių kaip hidrauliniai vožtuvai, gaminio mechaninių komponentų apribojimai.

2.3.5.3. Kintamos priežastys gali būti aptiktos arba identifikuotos. Kintamų priežasčių pavyzdžiais gali būti medžiagos savybių pasikeitimas, aplinkos sąlygos ir prietaiso nusidėvėjimas.

2.3.6 Specifiškumo ribos ir standartai

2.3.6.1. Specifiškumo ribos apibrėžia norimą diapozoną technologinio proceso arba gaminio kokybės kintamajam. Operatorius gali nurodyti SPV programai, ką reikia daryti, jeigu proceso kintamasis yra už diapazono ribų. Operatorius gali išsirinkti perspėjantį šviesinės signalizacijos įjungimą ir/arba liejamojo formavimo mašiną ir/arba įjungti papildomą išėjimą.

2.3.6.2. Pavyzdžiui, skaičiavimai konkrečiai preso formai gali atspėti, kad teisingam gaminio formavimui atkirtimo spaudimas turi būti tarp 1500 svarų/kv.colyje ir 1800 svarų/kv.colyje. Jeigu duomenys yra už šių ribų, tai gaminyt laikomas blogu. Šiuo atveju proceso kintamojo "standartas" žinomas iš anksto, todėl operatorius gali išsirinkti mašinos sustabdymą ir gaminių patikrinimą, jeigu faktiniai atkirtimo duomenys yra už to diapazono ribų.

2.3.6.3. Dauguma atveju žinomos teisingos specifiškumo ribos tokiems gaminio kokybės kintamiesiems kaip svoris, bet ne tokiems proceso kintamiesiems kaip spaudimas ir laikas.

2.3.6.4. Paprastai operatorius tikslina įrenginį tol, kol gaminiai nepereis kokybės patikrinimo. Kai tik gaminys bus geras, operatorius gali nustatyti “standartines” specifiškumo ribas aukščiau ir žemiau einamojo darbinio proceso taško. Prielaida yra ta, kad jeigu procesas nesikeičia, tai nesikeičia ir gaminio kokybė. SPV programa turi “specifiškumo ribų ištyrimo” funkciją, kuri supaprastina standartinių reikšmių nustatymą.

2.3.6.5. Jeigu procesas skiriasi nuo standartinio, tai gali paveikti ir gali nepaveikti gaminio kokybės. Operatorius gali nustatyti, koku būdu proceso pasikeitimas paveiks gaminio kokybę ir ar reikia išplėsti ar susiaurinti specifiškumo ribas.

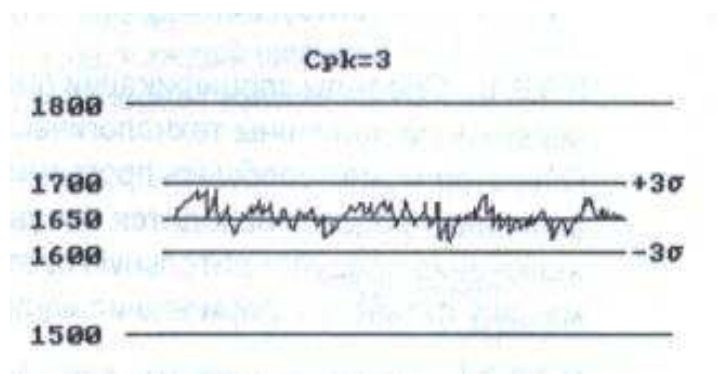
2.3.6.6. Išsagojimo ir pakartotinių preso formos nustatymų programa turi SPV specifiškumo ir spaudimo-formos duomenų konfigūracijos nustatymo ribas.

2.3.7 Kokybės proceso koeficientas (Cpk)

2.3.7.1. Kiek toli nukrypsta einamasis procesas nuo standartinio? Į šį klausimą atsako proceso galimybių indeksas Cpk. Kuo didesnis šis skaičius, tuo toliau išsidėsčiusios specifiškumo ribos nuo einamojo darbinio proceso diapozono.

2.3.7.2. Cpk apibrėžiamas kaip santykis atstumo nuo artimiausios specifiškumo ribos iki einamosios vidutinės proceso reikšmės ir proceso kintamojo ribos 3σ . Peržiūrėsime žemiau esančią spaudimo atkirtimo lentelę ir ją apibūdinančius paaiškinimus.

Proceso kintamojo išsibarstymas sudaro ± 50 svarų/kv.colį. Procesas gali nukrypti iki 100 svarų/kv.colį aukštin arba žemyn iki tol, kol kirs specifiškumo ribą. Koeficientas lygus 3.



$$C_{pk}=(1800-1650)/50=3$$

Proceso kintamojo išsibarstymas sudaro ± 50 svarų/kv.colį. Procesas yra ant specifiškumo ribos ir bet koks tolesnis vidutinės proceso reikšmės pasislinkimas aukštyn ar žemyn atves prie to, kad procesas išeis už ribų.

$$C_{pk}=(1550-1500)/50=1$$

Proceso kintamojo išsibarstymas sudaro ± 50 svarų/kv.colį. Procesas dalinai yra už specifiškumo ribų.

$$C_{pk}=(1525-1500)/50=0.5$$

Procesas visiškai už specifiškumo ribų.

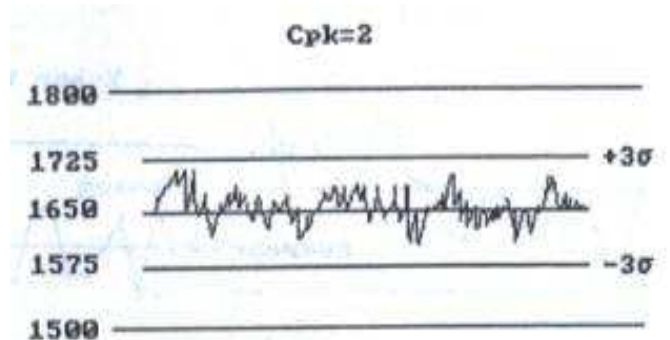
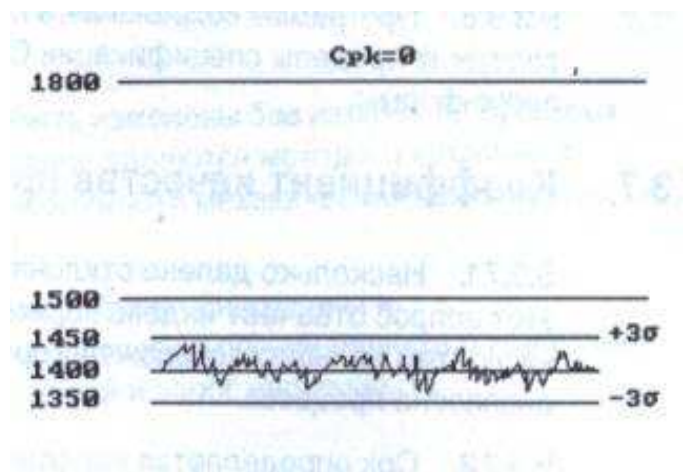
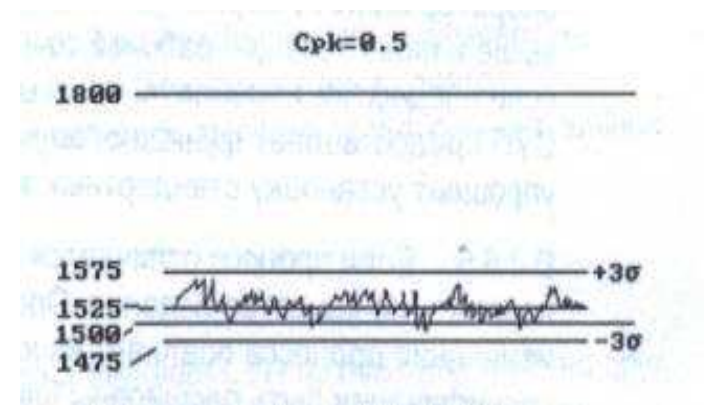
C_{pk} lygus 0.

$$C_{pk}=0$$

Proceso kintamojo išsibarstymas sudaro ± 75 svarų/kv.colį. Procesas gali nukrypti iki 75 svarų/kv.colį aukštyn arba žemyn tol, kol kirs specifiškumo ribą.

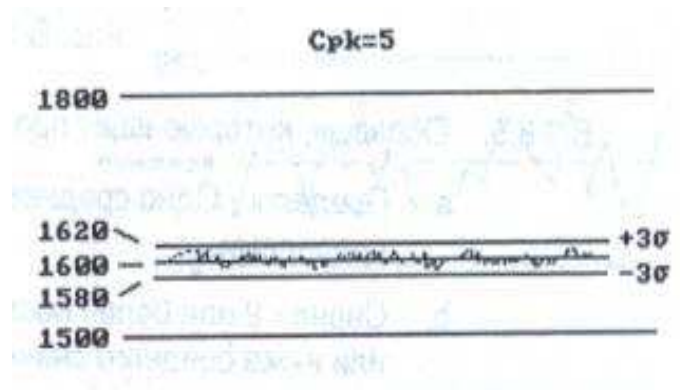
Koeficientas lygus 2.

$$C_{pk}=(1800-1650)/75=2$$



Proceso kintamojo išsibarstymas sudaro ± 20 svarų/kv.colį. Procesas gali nukrypti iki 80 svarų/kv.colį žemyn (iki 180 svarų/kv.colį aukštyn) tol, kol kirs specifiškumo ribą. Koeficientas lygus 5.

$$Cpk=(160-150/20)=5$$



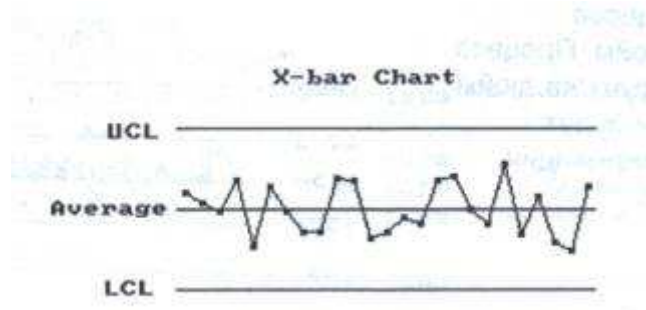
2.3.8 Valdymo diagramos

2.3.8.1. Valdymo diagramos parodo proceso vidutinę reikšmę ir kintamojo imties diapozoną. Vidutinė imties reikšmė yra parodymų imtyje suma, padalinta iš parodymų skaičiaus. Imties diapozonas lygus skirtumui tarp didžiausio ir mažiausio parodymo.

2.3.8.2. Diagramos su vidutinėmis reikšmėmis: diagrama su vidutinėmis reikšmėmis skirta nustatyti, ar procesas yra “statistinio valdymo būsenoje”. Procesas yra kontroliuojamas, jeigu imties ribos elgiasi atsitiktinai ir todėl labiausiai tikėtina, kad proceso nukrypimas vyksta dėl apribojimų, būdingų duotajam procesui (bendrų priežasčių).

2.3.8.3. SPV programa ieško pavyzdžių vidutinių reikšmių diagramoje, kurie rodo į tai, jog proceso pasikeitimas nėra atsitiktinis ir gali būti pasekmė problemų, kurios gali būti išaiškintos.

2.3.8.4. Viršutinė ir apatinė kontrolinės ribos nustato imties vidutinės reikšmės išsibarstymą. Jeigu proceso pasikeitimai yra atsitiktiniai, tai beveik visais atvejais (99.7%) vidutinė reikšmė turi patekti į kontrolinių ribų diapozoną.



Valdymo diagrama

Pav. B-3

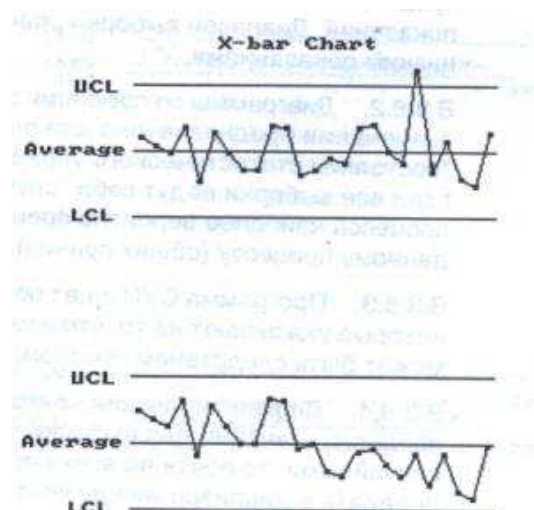
2.3.8.5. Pavyzdžiai, kurių ieško programa, yra tokie:

- Ribos – viena imties vidutinė reikšmė yra už kontrolinių ribų.
- Serija – 9 arba daugiau viena po kitos einančių vidutinių reikšmių yra aukščiau arba žemiau vidutinės vidutinių reikšmių diagramos.
- Trendas – 6 arba daugiau viena po kitos einančių vidutinių reikšmių turi tendenciją kilti aukštyn arba leistis žemyn.
- Kintami duomenys – 14 arba daugiau kintamų taškų vidutinių reikšmių diagramoje.

Ribos:

viena imties vidutinė reikšmė yra už kontrolinių ribų.

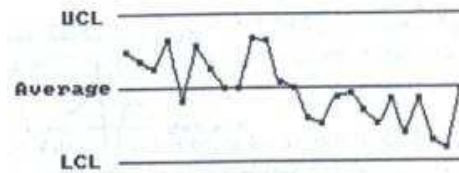
Serija :



9 arba daugiau viena po kitos einančių vidutinių reikšmių yra aukščiau arba žemiau vidutinės vidutinių reikšmių diagramos.

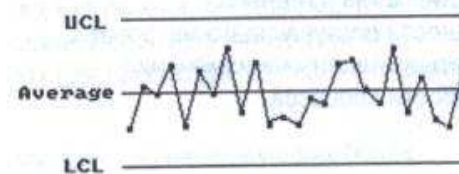
Trendas:

6 arba daugiau viena po kitos einančių vidutinių reikšmių, turi tendenciją kilti aukštyn arba leistis žemyn.



Kintami duomenys:

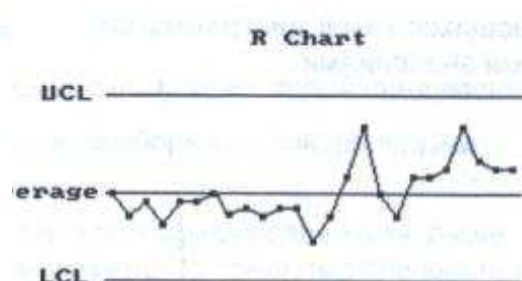
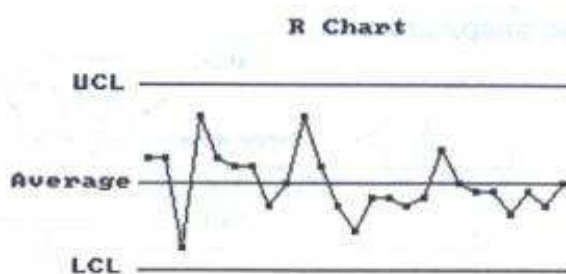
14 arba daugiau kintamų taškų vidutinių reikšmių diagramoje.



2.3.8.6. Diagramos R/s: Diagrama R/s skirta pavaizduoti proceso pasikeitimą pagal imties diapozoną. Jeigu imties dydis lygus 10 arba mažiau, tai programa apskaičiuoja R diapozoną kaip imties pasikeitimo matą. Jeigu imties dydis lygus 11 arba daugiau, tai diapozono skaičius nurodo pasikeitimo tiksliai ir todėl programa imties dydžio matu skaičiuoja didesnio tikslumo skaičių ir standartinę nuokrypį.

2.3.8.7. Imties diapozonas apibrėžiamas, kaip skirtumas tarp paties didžiausio ir paties mažiausio parodymo. Pvz.: 5 ciklų imties parodymams 1550, 1599, 1571, 1583, 1499 diapozonas bus lygus 100. Standartinis imties nuokrypis apibrėžiamas kaip kvadratinė šaknis iš parodymų sumos, atėmus vidutinę imties reikšmę, padalinus iš dydžio, kuris mažesnis už imties dydį, vienetu.

2.3.8.8. Jeigu skirtumas tarp viršutinės ir apatinės imties reikšmės keičiasi



ženkliai, tai proceso pakitimas pasikeičia ir gali iškilti problema. R/s diagramų imtys neturi kirsti viršutinės kontrolinės diagramos diapozono reikšmės.

Diagramos R/s pavyzdys: diagramoje R/s gaunamos medžiagos homogeniškumo pasikeitimai gali būti ištirti kaip proceso pakitimo pasikeitimai.

2.3.8.9. Reikia turėti omeny, kad R diagrama parodo proceso “išsibarstymą” ir jai neturi įtakos vidutinės proceso reikšmės pasikeitimas. Pvz.: jeigu vidutinė proceso reikšmė, kuri yra kontroliuojama, pasikeičia nuo 1500 svarų/kv.colį iki 1600 svarų/kv.colį, tai šis pasikeitimas bus parodytas trendo ir vidutinių reikšmių diagramose, bet R diagramoje - ne. Tačiau jei vidutinė reikšmė lieka tokia pati, bet pasikeičia “išsibarstymas”, tai trendo ir R diagramos atspindės tą pasikeitimą, bet vidutinių reikšmių diagrama - ne.

2.3.8.10. Valdymo diapazonas atvaizduoja iki 25 vidutinių imties reikšmių. Programa apskaičiuoja diagramos ribas ir tikrina tuos gaminių pavyzdžius kiekvieną kartą, kai gaunama nauja imtis. Diagramos atvaizduoja lange procesą iš 25 imčių ir pavaizduoja einamąją proceso būseną. Specifiškumo ribos atvirkščiai atvaizduoja fiksuotą etalono rėmą, su kuriuo gali susilyginti einamasis proceso “langas”.

2.3.9 Avariniai SPV pranešimai

2.3.9.1. SPV programa generuoja iki 4 avarinių pranešimų apie įrenginio būklę. Operatorius nustato, kurie iš kintamųjų turi būti kontroliuojami tuo metu, kai įvyksta avarija, ir ką SPV programa turi daryti, jeigu įvyko avarija.

2.3.9.2. Įvykus avarijai, avarijos ekrane rodomas pranešimas, detalesnis pranešimas užrašomas į įvykių žurnalą. SPV parametrų nustatymų pasikeitimas taip pat užrašomas įvykių žurnale.

2.3.9.3. Operatorius gali sukonfigūruoti avarijas SPV konfigūracijos ekrane. SPV suvestinės ekranas sumuoja avarinę būklę visų SPV kintamųjų.

2.3.9.4. Kiekvienam iš kontroliuojamų kintamųjų operatorius gali išrinkti įrenginio šviesos signalizaciją ir/arba įrenginio sustabdymą, ir/arba įjungti papildomą išėjimą, jeigu įjungti kokie nors avariniai pranešimai specifiškumo riboms.

2.3.9.5. SPV kintamasis (kintamieji) už specifiškumo ribų. SPV programa praneša operatoriui, kad bent vienas kintamasis yra už specifiškumo ribų. Po to operatorius gali persijungti į SPV suvestinės ekraną tam, kad peržiūrėtų SPV kintamųjų būklę.

2.3.9.6. Į įvykių žurnalą įrašomas vienas iš detalesnių pranešimų:
kintamojo pavadinimas – įjungta avarija “ taškas žemiau specifiškumo ribų”,
kintamojo pavadinimas – įjungta avarija “ taškas aukščiau specifiškumo ribų”.

2.3.9.7. Kai bus likviduotos avarijos įvykimo aplinkybės, tai į įvykių žurnalą bus įrašytas atitinkamas pranešimas apie avarijos “išjungimą”.

2.3.9.8. SPV kintamojo Cpk yra žemiau slenkstinės reikšmės. SPV programa informuoja operatorių apie tai, kad bent vieno kintamojo einamoji Cpk reikšmė yra žemiau, negu slenkstinė Cpk riba. Į įvykių žurnalą įrašomas toks pranešimas :

kintamojo pavadinimas – įjungta avarija “Cpk žemiau slenkstinio lygio”.

2.3.9.9. Kai bus likviduotos avarijos įvykimo aplinkybės, tai į įvykių žurnalą bus įrašytas atitinkamas pranešimas apie avarijos išjungimą.

2.3.9.10. Avarijos A ir B vadinamos specifiškumo ribų avarijomis, nes avarinės būklės nustatymui programa naudoja einamąsias specifiškumo ribas.

2.3.9.11. Specifiškumo ribų slenkstis nustato, kiek turi būti įvykdyta ciklą, kurių metu specifiškumo ribų avarija turi būti įjungta kintamajam paeiliui iki tol, kol SPV programa imsis veiksmų, nustatytų šiai avarijai.

2.3.9.12. Nevaldomi SPV kintamieji. Vidutinių reikšmių diagrama bent vienam kintamajam demonstruoja neatsitiktinius pavyzdžius ir todėl šis kintamasis įvardijamas kaip “statistiškai nevaldomas” parametras.

2.3.9.13. Į įvykių žurnalą įrašomas vienas iš detalesnių pranešimų:

kintamojo pavadinimas – įjungta avarija “už kontrolinių ribų vidutinės reikšmės diagramoje,

kintamojo pavadinimas – įjungta avarija “Serija”,

kintamojo pavadinimas – įjungta avarija “Trendas”,

kintamojo pavadinimas – įjungta avarija “Besikeičiantys duomenys”.

2.3.9.14. Kai bus likviduotos avarijos įvykimo aplinkybės, tai į įvykių žurnalą bus įrašytas atitinkamas pranešimas apie avarijos išjungimą.

2.3.9.15. SPV ciklo nutraukimas: įrenginys buvo sustabdytas dėl avarijos, susijusios su specifiskumo ribomis arba Cpk slenkstine reikšme.

3. Logistinė regresija

3.1 Matematinis modelis

Tarkime, norime prognozuoti kintamąjį Y , įgyjantį dvi reikšmes 0 ir 1, remiantis atsitiktinio vektoriaus $X = (X_1, \dots, X_k)$ matavimais. Tegu $Y=0$ jei gaminys geras, ir $Y=1$, jeigu gaminys defektinis. Jeigu Y skirstinys priklauso nuo X , tai sąlyginės tikimybės

$$\pi(x) = P\{Y = 1|X = x\}, \quad 1 - \pi(x) = P\{Y = 0|X = x\}$$

yra vektoriaus $x = (x_1, \dots, x_k)$ funkcijos. Kitaip tariant, a.d. Y skirstinys, kai $X=x$, yra binominis $B(1, \pi(x))$. Šį skirstinį galima užrašyti taip:

$$\begin{aligned} P\{Y = y|X = x\} &= [\pi(x)]^y [1 - \pi(x)]^{1-y}, \quad y = 0, 1; \\ E(Y|X = x) &= \pi(x) = \pi(x_1, \dots, x_k). \end{aligned} \tag{3.1}$$

Naudojamas toks funkcijos $\pi(x)$ priklausomybės nuo x modelis:

$$\pi(x) = \frac{e^{Z(x)}}{1 + e^{Z(x)}}, Z(x) = \beta_0 + \beta_1 x_1 + \dots + \beta_k x_k. \quad (3.2)$$

Šiame modelyje visais atvejais $0 \leq \pi(x) \leq 1$, kai $Z(x) \rightarrow +\infty$, tai $\pi(x) \rightarrow 1$; kai

$$Z(x) \rightarrow -\infty, \text{ tai } \pi(x) \rightarrow 0.$$

Sudarykime santykį

$$\gamma(x) = \frac{\pi(x)}{1 - \pi(x)} = e^{Z(x)}, \quad (3.3)$$

kuri vadiname įvykio $\{Y=1\}$ galimybės santykiu, kai $X=x$.

Galimybės santykį $\gamma(x)$ galima panaudoti klasifikavimo taisyklei konstruoti. Jeigu konkrečiam gaminiui $\gamma(x) > 1$, tai jis priskiriamas “įtartinų” kategorijai, jeigu $\gamma(x) \leq 1$, tai – gerų.

Nelygybė $\gamma(x) > 1$ ekvivalenti nelygybėms

$$\pi(x) > \frac{1}{2}, e^{Z(x)} > 1, Z(x) > \ln 1 = 0 \quad (3.4)$$

Tada klasifikavimo taisyklę galima suformuluoti taip:

- gaminys priskiriamas “įtartinų” kategorijai, jeigu $Z(x) > 0$;
- gaminys priskiriamas gerų kategorijai, jeigu $Z(x) \leq 0$.

P a s t a b a. Atsižvelgiant į praktinius poreikius, klasifikavimo ribą galima keisti. Pavyzdžiui, gali būti pageidaujama, kad į “įtartinų” gaminių kategoriją patektų, tarkime, ne daugiau kaip 1 procentas visos produkcijos. Tada klasifikavimo taisyklę (3.4) galima modifikuoti taip: “įtartinų” kategorijai priskirsime tuos gaminius, kurių galimybės santykis $\gamma(x)$ yra, pavyzdžiui, 10:1. Taigi bendruoju atveju klasifikavimo taisyklę galima suformuluoti taip:

- gaminys priskiriamas “įtartinų” kategorijai, kai

$$\pi(x) > c, e^{Z(x)} > \frac{c}{1-c}, Z(x) - \ln \frac{c}{1-c} > 0. \quad (3.5)$$

Nagrinèkime dvi a.v. x reikšmes x' ir x'' :

$$x' = (x_1, \dots, x_i + 1, \dots, x_k), \quad x'' = (x_1, \dots, x_i, \dots, x_k),$$

kurios skiriasi tik tuo, kad pirmajai i -oji koordinatè padidinta 1, visos kitos koordinatès tokios paèios. Apskaièiuokime galimybès santykio pokytì, pereinant nuo taško x' prie taško x'' :

$$\frac{\gamma(x')}{\gamma(x'')} = \frac{e^{Z(x')}}{e^{Z(x'')}} = e^{Z(x') - Z(x'')} = e^{\beta_i}. \quad (3.6)$$

Šis dydis parodo, kiek kartù pakinta (padidèja ar sumažèja) galimybès santykis, kai a.v. x i -oji koordinatè sumažèja vienetu, o visos kitos koordinatès nesikeièia.

3.2 Logistinės regresijos statistiniai uždaviniai

Modelyje (4.2) parametrai $\beta = (\beta_0, \beta_1, \dots, \beta_k)$ nežinomi, tad juos tenka vertinti iš statistinių duomenų.

Tarkime, turime tokius statistinius duomenis

$$(Y_i, X^{(i)}) = (Y_i, X_1^{(i)}, \dots, X_k^{(i)}), \quad i = 1, 2, \dots, N_0 + N_1; \quad (3.7)$$

èia Y_i ìgyja reikšmè 1, jeigu gaminys defektinis, ir reikšmè 0, jeigu geras; o

$$(X^{(i)}) = (X_1^{(i)}, \dots, X_k^{(i)}), \quad i = 1, 2, \dots, N_0 + N_1$$
 yra i -ojo gaminio parametrų reikšmès;

N_1 – defektinių gaminių skaièius, N_0 – gerų gaminių skaièius.

Sudarome didžiausiojo tikètino funkcijà $L(\beta)$; ji gaunama ìrašant ì skirstinį (3.1) paeiliui stebèjimus (3.7) ir sudauginant:

$$L(\beta) = L(\beta | Y_i, X^{(i)}) = \prod_{i=1}^{N_0 + N_1} [\pi(X^{(i)})]^{Y_i} [1 - \pi(X^{(i)})]^{1 - Y_i}, \quad i = 1, 2, \dots, N_0 + N_1. \quad (3.8)$$

Parametrų $\beta_0, \beta_1, \dots, \beta_k$ ìvertinius $\hat{\beta}_0, \hat{\beta}_1, \dots, \hat{\beta}_k$ randame didžiausiojo tikètino metodu, t.y. iš sąlygos

$$\ln L(\beta) \rightarrow \max. \quad (3.9)$$

Diferencijuodami $\ln L(\beta)$ pagal $\beta_0, \beta_1, \dots, \beta_k$ įvertiniamis rasti gauname didžiausiojo tikėtimumo lygčių sistemą

$$\frac{\partial \ln L(\beta)}{\partial \beta_j} = \sum_{i=1}^{N_0+N_1} X_j^{(i)} [Y_i - \pi(X^{(i)})] = 0, \quad j=0,1,\dots,k. \quad (3.10)$$

Tai netiesinių $k+1$ lygčių su $k+1$ nežinomuojų sistema, kurią tenka spręsti iteracijų metodais. Tokių lygčių sistemų sprendimas numatytas daugelyje matematinės statistikos taikomųjų programų paketu.

Gavę parametrų $\beta_0, \beta_1, \dots, \beta_k$ įvertinius $\hat{\beta}_0, \hat{\beta}_1, \dots, \hat{\beta}_k$, galime įvertinti tikimybę (3.2)

$$\hat{\pi}(x) = \frac{e^{\hat{Z}(x)}}{1 + e^{\hat{Z}(x)}}, \quad \hat{Z}(x) = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \dots + \hat{\beta}_k x_k,$$

galimybės santykį (3.3)

$$\hat{\gamma}(x) = \frac{\hat{\pi}(x)}{1 - \hat{\pi}(x)} = e^{\hat{Z}(x)},$$

galimybės santykio pokytį (3.6)

$$\frac{\hat{\gamma}(x')}{\hat{\gamma}(x'')} = \frac{e^{\hat{Z}(x')}}{e^{\hat{Z}(x'')}} = e^{\hat{Z}(x') - \hat{Z}(x'')} = e^{\hat{\beta}_i}$$

ir klasifikatoriaus funkciją (3.5)

$$\hat{Z}(x) - \ln \frac{c}{1-c} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \dots + \hat{\beta}_k x_k - \ln \frac{c}{1-c} > 0. \quad (3.11)$$

Pateiksime kriterijų hipotezei $H: \beta_1 = \dots = \beta_k = 0$ tikrinti. Ši hipotezė ekvivalenti tvirtinimui, kad $\pi(x) = P\{Y = 1 | X = x\} \equiv \pi, \forall x$, t.y. kad Y skirstinys nuo parametrų vektoriaus X nepriklauso ir Y prognozavimas pagal X neturi prasmės.

Pažymėkime

$$\hat{Y}_i = \hat{\pi}(X^{(i)}) = \frac{e^{\hat{z}(X^{(i)})}}{1 + e^{\hat{z}(X^{(i)})}}, i = 1, 2, \dots, N_0 + N_1$$

reikšmės Y_i prognozę, kai modelis (3.2) yra teisingas. Jeigu teisinga hipotezė H , tai tikimybės π įvertinys lygus aritmetiniam vidurkiui:

$$\hat{\pi} = \bar{Y} = \sum_{i=1}^{N_0+N_1} Y_i.$$

Didžiausio tikėtimumo funkcijos $L(\beta)$ maksimumas, kai teisingas modelis (3.2), yra

$$L(\hat{\beta}) = \prod_{i=1}^{N_0+N_1} \hat{Y}_i^{Y_i} (1 - \hat{Y}_i)^{1-Y_i}, \quad (3.12)$$

o kai teisinga hipotezė H ,

$$L(\hat{\pi}) = \prod_{i=1}^{N_0+N_1} \bar{Y}_i^{Y_i} (1 - \bar{Y}_i)^{1-Y_i}. \quad (3.13)$$

Jeigu hipotezė H teisinga, tai \hat{Y}_i ir \bar{Y} yra artimi, o kartu artimos funkcijų (3.11) ir (3.12) reikšmės. Įrodyta, kad esant dideliems stebėjimų skaičiams santykio

$$D_E = -\ln \frac{L_1(\hat{\pi})}{L(\hat{\beta})} \quad (3.14)$$

skirstinys artimas chi kvadrato skirstiniui su k laisvės laipsnių. Todėl hipotezė H atmetama, kai

$$D_E > \chi_\alpha^2(k);$$

čia $\chi_\alpha^2(k)$ yra chi kvadrato su k laisvės laipsnių α kritinė reikšmė.

Nagrinėkime hipotezę

$$H_{i_1, \dots, i_l} : \beta_{i_1} = \dots = \beta_{i_l} = 0, i_1 \neq i_2 = \dots = i_l, 1 \leq l < k,$$

kad parametrai X_{i_1}, \dots, X_{i_l} neturi įtakos prognozės tikslumui. Pažymėkime $D_E^{(k)} = D_E$ ir D_E^{k-l} statistiką (3.14) modeliui su visais parametrais $\beta_0, \beta_1, \dots, \beta_k$ ir modeliui, kai parametrai $\beta_{i_1}, \dots, \beta_{i_l}$ yra praleisti. Tada skirtumo $D_E^{(k)} - D_E^{k-l}$ skirstinys aproksimuojamas chi kvadrato skirstiniu su $k-(k-l)=l$ laisvės laipsnių. Hipotezė H_{i_1, \dots, i_l}

atmetama, kai

$$D_E^{(k)} - D_E^{k-l} > \chi^2_\alpha(l). \quad (3.15)$$

Parametrų β_j įvertinių $\hat{\beta}_j$ tikslumui apibūdinti apskaičiuojame Fišerio informacinę matricą $I(\beta)$ su elementais

$$I_{ls}(\beta) = -E \frac{\partial^2 \ln L(\beta)}{\partial \beta_l \partial \beta_s} = \sum_{i=1}^{N_0+N_1} X_l^{(i)} X_s^{(i)} \pi(X^{(i)}) (1 - \pi(X^{(i)}))$$

ir jos įvertinį $I(\hat{\beta})$ su elementais

$$I_{ls}(\hat{\beta}) = \sum_{i=1}^{N_0+N_1} X_l^{(i)} X_s^{(i)} \hat{\pi}(X^{(i)}) (1 - \hat{\pi}(X^{(i)})).$$

Tada atsitiktinio vektoriaus $\hat{\beta}$ skirstinys aproksimuojamas k -mačiu normaliuoju:

$$\hat{\beta} \sim N_{k+1}(\beta, I^{-1}(\hat{\beta})). \quad (3.16)$$

Atskiros vektoriaus $\hat{\beta}$ koordinatės $\hat{\beta}_s$ skirstinys aproksimuojamas vienmačiu normaliuoju $N(\beta_s, \hat{\sigma}_{ss})$ arba

$$\frac{\hat{\beta}_s - \beta_s}{\sqrt{\hat{\sigma}_{ss}}} \sim N(0,1); \quad (3.17)$$

čia $\hat{\sigma}_{sl}$; $s, l = 0, 1, \dots, k$, yra matricos $I^{-1}(\hat{\beta})$ elementai.

Tada hipotezė $H_s : \beta_s = 0$ atmetama, kai

$$\frac{|\hat{\beta}_s|}{\sqrt{\hat{\sigma}_{ss}}} > z_\alpha;$$

čia z_α - standartinio normaliojo skirstinio α kritinė reikšmė. Pažymėsime, kad šią hipotezę buvo galima tikrinti ir kriterijumi (3.15).

Naudojant aproksimacijas (3.16) ir (3.17) galima sudaryti parametru $\beta_s, s=0, 1, \dots, k$ arba jų funkcijų apytikslus pasiklovimo intervalus. Pavyzdžiui, iš (3.17) parametru β_s gauname pasiklovimo intervalą:

$$(\hat{\beta}_s - z_\alpha \sqrt{\hat{\sigma}_{ss}}, \hat{\beta}_s + z_\alpha \sqrt{\hat{\sigma}_{ss}})$$

kur pasiklovimo lygmuo $1-2\alpha$.

3.3 Klasifikavimo tikslumas

Tarkime, objektai (gaminiai) gali priklausyti vienai iš dviejų klasių. Pažymėkim Y a.d., kuris įgyja reikšmę 0, jeigu objektas priklauso pirmai klasei (gaminys yra geras), ir įgyja reikšmę 1, jeigu objektas priklauso antrai klasei (gaminys yra defektinis). Tegu η yra a.d., kuris įgyja reikšmę 0 arba 1, jeigu objektas priklauso pirmai arba antrai klasei. Klasifikavimo tikslumą galima apibūdinti tikimybėmis

$$\alpha_{ij} = P\{\eta = i | Y = j\}, \quad i, j = 0, 1 \quad (3.18)$$

jas surašysime į lentelę

1.lentelė. Klasifikavimo tikslumo tikimybės

$Y \backslash \eta$	0	1	Σ
0	α_{00}	α_{10}	1
1	α_{01}	α_{11}	1

Šioje lentelėje α_{00} ir α_{11} yra teisingų sprendimų tikimybės, o α_{01} ir α_{10} – klaidingų. Klasifikavimo taisyklė tuo geresnė, kuo didesnės tikimybės α_{00} ir α_{11} ir kuo mažesnės tikimybės α_{01} ir α_{10} . Klasifikuojant gaminius, tikimybė α_{10} vadinama gamintojo rizika, o tikimybė α_{01} – vartotojo rizika.

Praktiškai svarbesnės atvirkštinės tikimybės

$$\beta_{ji} = P\{Y = j | \eta = i\} = \frac{\alpha_{ij} w_j}{\alpha_{i1} w_1 + \alpha_{i0} w_0}, \quad i, j = 0, 1; \quad (3.19)$$

čia w_0 ir w_1 - klasių apriorinės tikimybės: $w_0 = P\{Y=0\}$, $w_1 = P\{Y=1\}$, $w_0 + w_1 = 1$.

Defektiniais pripažintų gaminių srauto dydis

$$Q = P\{\eta = 1\} = \alpha_{10} w_0 + \alpha_{11} w_1. \quad (3.20)$$

Jeigu pripažintų defektiniais gaminių dalis w_1 yra maža, tai įrašę formulėje (3.20) ir formulį (3.19) vardiklių $w_1=0$, $w_0=1$ gausime apytiksles formules

$$Q = \alpha_{10}; \quad \beta_{10} = w_1 \frac{\alpha_{01}}{\alpha_{00}} = w_1 \gamma; \quad \beta_{11} = w_1 \frac{\alpha_{11}}{\alpha_{10}} = w_1 \delta, \quad (3.21)$$

kuriomis galime apibūdinti klasifikavimo tikslumą, net ir nežinodami tikslų tikimybių w_0 ir w_1 reikšmių. Būtent pripažintų gerais gaminių srauto defektiškumo lygis apytiksliai sumažėja nuo w_1 iki $w_1 \gamma$, o pripažintų defektiniais gaminių srauto – padidėja nuo w_1 iki $w_1 \delta$.

Praktiškai tikimybės α_{ij} , $i, j = 0, 1$ nežinomos, jas reikia vertinti iš statistinių duomenų. Pritaikysime rastą klasifikavimo taisyklę statistiniams duomenims (3.7). Pažymėkime V_{00} ir V_{10} kaip gerais ir defektiniais pripažintų gaminių skaičių gerų gaminių aibėje

$(V_{00} + V_{10} = N_0)$; analogiškai V_{01} ir V_{11} - defektinių gaminių aibėje ($V_{01} + V_{11} = N_1$).

Naudodami šiuos duomenis, gauname tikimybių α_{ij} įvertinius

$$\hat{\alpha}_{ij} = \frac{V_{ij}}{N_j}, i, j = 0, 1. \quad (3.22)$$

Klasifikavimo rezultatus apibūdinantys skaičiai V_{ij} pateikti 2 lentelėje, o tikimybių

α_{ij} įvertiniai $\hat{\alpha}_{ij}$ - 3 lentelėje.

Turėdami įvertinius $\hat{\alpha}_{ij}$, gauname charakteristikų (3.21) įvertinius.

$$\hat{Q} = \hat{\alpha}_{10}; \gamma = \frac{\hat{\alpha}_{01}}{\hat{\alpha}_{00}}, \hat{\delta} = \frac{\alpha_{11}}{\alpha_{10}}. \quad (3.23)$$

2 lentelė. Klasifikavimo rezultatai

$Y \backslash \eta$	0	1	Σ
0	V_{00}	V_{10}	N_0
1	V_{01}	V_{11}	N_1

3 lentelė. Tikimybių α_{ij} įvertiniai

$Y \backslash \eta$	0	1	Σ
0	$\hat{\alpha}_{00} = \frac{V_{00}}{N_0}$	$\hat{\alpha}_{10} = \frac{V_{10}}{N_0}$	1
1	$\hat{\alpha}_{01} = \frac{V_{01}}{N_1}$	$\hat{\alpha}_{11} = \frac{V_{11}}{N_1}$	1

4. BMPM (Biased Minimax Probability Machine)

4.1 Įvadas į BMPM

Paslinktas klasifikatorius turi daug pritaikymų, įskaitant ir medicininę diagnostiką. Dviejų klasių paslinkto klasifikatoriaus konstravimo tikslas – vietoje bendrojo tikslumo svarbios klasės tikslumą padaryti kaip galima didesnį, kol mažiau svarbios klasės tikslumas palaikomas priimtina lygyje. Kai kuriems paslinktiems klasifikatoriams, pvz., svarbių vektorių metodui (angliškai vadinamam *the weighted Support Vector Machine*; žr. šaltinį [8]) sunku kokybiškai įvertinti, kaip svoris (weight) paveikia klasifikavimą. Neseniai naujas klasifikavimo modelis MPM (angliškai vadinamas *Minimax Probability Machine*; žr. šaltinį [4]) suteikė galimybę apskaičiuoti blogiausio atvejo ribą neteisingai būsimų duomenų tikimybei, patikimo įverčio vidurkiui ir duomenų apmokymo klasės kovariacijų matricai. Šis metodas remiasi bendru autorių Marshall ir Olkin tvirtinimu (žr. šaltinį [3])

$$\sup_{y \in \{\bar{y}, \Sigma_y\}} P \{y \in S\} = \frac{1}{1 + d^2}, \text{ kur } d^2 = \inf_{y \in S} (y - \bar{y})^T \Sigma_y^{-1} (y - \bar{y}),$$

kur supremumas imamas pagal visus skirstinius, kurie turi vidurkių vektorių \bar{y} ir kovariacijų matricą Σ_y (nebūtinai normalieji skirstiniai). Čia S iškila aibė. Toliau imama $S = \{a^T y \geq b\}$. Jo efektyvumas palyginamas su klasifikatoriumi SVM (angliškai vadinamu *the Support Vector Machine*; žr. šaltinį [12]). Viena pagrindinių šio metodo formuliu, kurios įrodymas bus paaiškintas kitame skyrelyje, yra tokia:

$$\sup_{y \in \{\bar{y}, \Sigma_y\}} P \{a^T y \geq b\} = \frac{1}{1 + d^2}, \text{ kur } d^2 = \inf_{a^T y \geq b} (y - \bar{y})^T \Sigma_y^{-1} (y - \bar{y}).$$

Paslinktas klasifikavimo (BMPM) modelis yra transformuotas į įgaubtą-išgaubtą trupmeninį programavimo (toliau-FP) (angliškai vadinamu concave-convex Fractional Programming; žr. šaltinį [10]) uždavinį arba pseudoįgaubtą uždavinį, kur kiekvienas lokalus minimumas yra globalus maksimumas. Šis modelis yra pirmasis kiekybinis metodas, kontruojuojantis, kaip sprendinių hiperplokštuma juda svarbesnės klasės klasifikavimo naudai.

4.2 Tiesinė optimalių pastumtų tikimybinių sprendinių hiperplokštuma

Apžvelgdami originalų MPM, pristatysime tiesinę pastumtą minimax schemą. Tarkime, du atsitiktiniai vektoriai atstovauja dviems duomenų klasėms su vidurkais $\bar{x}, \bar{y} \in R^n$ ir kovariacijų matricomis $\Sigma_x, \Sigma_y \in R^{n \times n}$. Patogumo dėlei taip pat x ir y naudojami reikšti atitinkamai X ir Y duomenų klasei. Dviejų duomenų klasių patikimais įverčiais $\{\bar{x}, \Sigma_x\}, \{\bar{y}, \Sigma_y\}$, naudojant MPM, pamėginsime apibrėžti hiperplokštumą $a^T z = b (a \neq 0, z \in R^n, b \in R)$, kuri gali atskirti dvi duomenų klases su maksimaliomis tikimybėmis. Modelio formulavimas aprašomas taip :

$$\max_{\alpha, \beta, a \neq 0} \alpha \quad \text{toks, kad} \quad \inf P \{a^T x \geq b\} \geq \alpha,$$

$$\inf P \{a^T x \leq b\} \geq \alpha,$$

Būsiami taškai z , kuriems $a^T z > b$, klasifikuojami kaip X klasė; priešingu atveju, juos vertinam kaip Y klasę. Ši gauta sprendinių hiperplokštuma pasiekama minimizuojant neklasifikuotą blogiausio atvejo (maksimumo) tikimybę arba klaidos dažnumą, klasifikuojant būsimus duomenis. Be to, MPM uždavinys gali būti transformuotas į išgaubtą optimizavimo uždavinį, tiksliau sakant, į uždavinį, angliškai vadinamą *Second Order Cone Programming* (žr. šaltinius [6], [7]).

Šiame modelyje dvi klasės įgyja nepastumtus svorius, t.y. priverčia X klasės ir Y klasės tikimybes būti lygias reikšmei α . Tačiau realiame pasaulyje dviejų klasių svarbumo vartojimas ne visada toks pats, tai reiškia, kad dvi atitinkamos tikimybės yra nebūtinai lygios. Atsižvelgiant į šį pastebėjimą, naudojamas toks BMPM formulavimas:

$$\max_{\alpha, \beta, a \neq 0} \alpha \quad \text{tokius, kad} \quad \inf P \{a^T x \geq b\} \geq \alpha, \quad (4.1)$$

$$\inf P \{a^T x \leq b\} \geq \beta, \quad (4.2)$$

$$\beta \geq \gamma, \quad (4.3)$$

kur γ yra iš anksto pasirinkta teigiama konstanta, kuri reiškia priimtina mažiau svarbios klasės tikslumo lygį.

Ši optimizacija maksimizuoja tikslumą (tikimybės α) paslinktai X klasei, kol kitai Y -kų klasei palaikomas apatinės ribos priimtino lygio tikslumas tokioje aplinkoje kaip (4.3). Hiperplokštuma $a^T z = b^*$ duoda šio uždavinio sprendimą, palankų svarbios klasės X klasifikavimo naudai prieš mažiau svarbią klasę Y ir labiau tinkantį pastumto klasifikavimo uždavinio panaudojimui.

Ši optimizavimo problema sprendžiama taip. Pirmiausia naudojama Lema 1 iš šaltinio [5].

Lema 1: Tarkim, $a \neq 0, b$ toks kad $a^T y \leq \beta$ ir $\beta \in [0,1)$ su sąlyga

$$\inf P \{a^T y \leq b\} \geq \beta,$$

tada ir tik tada jeigu $b - a^T \bar{y} \geq \kappa(\beta) \sqrt{a^T \Sigma_y a}$, kur $\kappa(\beta) = \sqrt{\frac{\beta}{1-\beta}}$.

Lema 1 gali būti įrodoma naudojant Lagranžo daugiklių metodą, teorija aiškinama šaltinyje [9]:

$$\sup_{y \in \{\bar{y}, \Sigma_y\}} P \{a^T y \geq b\} = \frac{1}{1+d^2}, \text{ kur } d^2 = \inf_{a^T y \geq b} (y - \bar{y})^T \Sigma_y^{-1} (y - \bar{y})$$

Paaiškinkime, kaip gaunama ši lema. Parodykime, kad d gaunamas taip:

$$d^2 = \inf_{a^T y \geq b} (y - \bar{y})^T \Sigma_y^{-1} (y - \bar{y}) = \frac{\max(b - a^T \bar{y}, 0)^2}{a^T \Sigma_y a}.$$

Panaudokime Lagranžo daugiklių metodą. Nagrinekime du variantus:

1) Jeigu $a^T \bar{y} \leq b$.

Pažymėkime $p^T = a^T \sum_y^{1/2}$, $g = \sum_y^{-1/2} (y - \bar{y})$, ir $q = b - a^T \bar{y}$, tą patį galime užrašyti taip $d^2 = \inf_{p^T w \geq q} g^T g$. Taip pat g galime gauti įvedę Lagranžo daugiklį: $\{g, \lambda\} = \arg \min_g \arg \max_{\lambda} \{g^T g + \lambda(q - p^T g)\}$, kur daugiklis $\lambda \geq 0$. Todėl gauname tokias lygybes:

$$g = \frac{\lambda p}{2}, q = p^T g.$$

kadangi $a^T \bar{y} \leq b$, nesunkiai gauname, kad $q \geq 0$. Toliau gauname:

$$\lambda = \frac{2q}{p^T p}, g = \frac{dp}{p^T p}.$$

Gauname tokią lygybę:

$$d^2 = \inf_{a^T y \geq b} (y - \bar{y})^T \sum_y^{-1} (y - \bar{y}) = \frac{(b - a^T \bar{y})^2}{a^T \sum_y a}.$$

2) Jeigu $a^T \bar{y} \geq b$.

Šiuo atveju gali būti tik $y = \bar{y}$. Todėl $d = 0$.

Sujungę abu atvejus ir gausime šios teoremos įrodymą. ■

Pasinaudojus Lema 1, gaunamas šis transformuotas optimizavimo uždavinys:

$$\max_{\alpha, \beta, \alpha \neq 0} \alpha \text{ toks, kad } -b + a^T \bar{x} \geq \kappa(\alpha) \sqrt{a^T \Sigma_x a}, \quad (4.4)$$

$$b - a^T \bar{x} \geq \kappa(\alpha) \sqrt{a^T \Sigma_x a}, \quad (4.5)$$

$$\beta \geq \gamma, \quad (4.6)$$

kur $\kappa(\alpha) = \sqrt{\frac{\alpha}{1-\alpha}}$, $\kappa(\beta) = \sqrt{\frac{\beta}{1-\beta}}$. (4.5) lygybė gaunama tiesiogiai iš (4.2)

panaudojus Lemą 1. Panašiai pakeitus $a^T x \geq b$ į $a^T (-x) \leq -b$, (4) gaunama iš (4.1).

Iš (4.4) ir (4.5), gaunama:

$$a^T y + \kappa(\beta)\sqrt{a^T \Sigma_y a} \leq b \leq a^T x + \kappa(\alpha)\sqrt{a^T \Sigma_x a}. \quad (4.7)$$

Jei iš šios nelygybės eliminuojama b , gaunama

$$a^T (\bar{x} - \bar{y}) \geq \kappa(\alpha)\sqrt{a^T \Sigma_x a} + \kappa(\beta)\sqrt{a^T \Sigma_y a}. \quad (4.8)$$

Matome, kad vektoriaus a norma (ilgis) neturi įtakos (4.8). Neprarandant bendrumo, galima pažymėti $a^T (\bar{x} - \bar{y}) = 1$. Kadangi $\kappa(\alpha)$ monotoniškai didėja su α , α maksimizavimas yra ekvivalentus $\kappa(\alpha)$ maksimizavimui. Taip uždavinys gali būti modifikuotas į

$$\max_{\alpha, \beta, a \neq 0} \kappa(\alpha) \quad \text{toks, kad} \quad 1 \geq \kappa(\alpha)\sqrt{a^T \Sigma_x a} + \kappa(\beta)\sqrt{a^T \Sigma_y a}, \quad (4.9)$$

$$a^T (\bar{x} - \bar{y}) = 1, \quad (4.10)$$

$$\kappa(\beta) \geq \kappa(\gamma), \quad (4.11)$$

kur (4.9) yra ekvivalentus (4.6). Tai funkcijos κ momotoniškumo savybė.

Lema 2: Maksimali $\kappa(\alpha)$ reikšmė apribota (4.9), (4.10), (4.11) yra pasiekiamą, kai dešinė (4.9) pusė yra griežtai lygi 1.

Paaiškinkime, kaip gaunama Lema 2. Tarkime, maksimumas pasiekiamas, kai $1 > \kappa(\alpha)\sqrt{a^T \Sigma_x a} + \kappa(\beta)\sqrt{a^T \Sigma_y a}$. Naujas sprendimas konstruojamas pagal didėjantį $\kappa(\alpha)$ su mažu teigiamu skaičiumi ir išlaikant $\kappa(\beta)$ nepakitusių, tenkinant apribojimus. Gausim tikslesnį atsakymą.

Be to, Σ_x ir Σ_y gali būti laikomos teigiamai apibrėžtomis matricomis (išsigimusio

atvejo nenagrinėsime). Todėl $\kappa(\alpha) = \frac{1 - \kappa(\beta)\sqrt{a^T \Sigma_y a}}{\sqrt{a^T \Sigma_x a}}$ pagal Lemą 2. Akivaizdu, ši

optimizavimo funkcija yra tiesinė funkcija dėl to, kad $\kappa(\beta)$ ir $\sqrt{a^T \Sigma_y a}$ yra teigiamos išraiškos; ši optimizavimo funkcija yra maksimizuojama, kai $\kappa(\beta)$ yra žemutinė

$\kappa(\gamma)$ riba. Tuo būdu BMPM optimizavimo uždavinys gali būti pakeistas ir perrašytas į taip vadinamąjį trupmeninį programavimą (FP) (Fractional Programming; žr. šaltinį [10]):

$$\max_{a \neq 0} \frac{f(a)}{g(a)}, \quad \text{toks kad } a \in A = \{a \mid a^T(\bar{x} - \bar{y}) = 1\}, \quad (4.12)$$

kur $f(a) = 1 - \kappa(\gamma)\sqrt{a^T \Sigma_y a}$, $g(a) = \sqrt{a^T \Sigma_x a}$. Toliau siūloma Lemos 3 pagalba parodyti, kad šis FP uždavinys yra išsprendžiamas.

Lema 3: Trupmeninio programavimo uždavinys (4.12) yra griežtai kvaziįgaubimo uždavinys ir todėl yra išsprendžiamas.

Paaiškinkime, kaip gaunama Lema 3. Lengva pastebėti, kad sritis A yra išgaubta forma erdvėje R^n , $f(a)$ ir $g(a)$ yra diferencijuojamos funkcijos srityje A ir $g(a)$ yra išgaubta funkcija srityje A . Tada $\frac{f(a)}{g(a)}$ yra įgaubimo-išgaubimo FP arba pseudoįgaubimo uždavinys. Iš čia tai yra griežtai kvaziįgaubta srityje A pagal šaltinį [10]. Todėl kiekvienas lokalus maksimumas yra ir globalus maksimumas (žr. šaltinį [10]). Kitaip tariant, šis trupmeninio programavimo uždavinys yra išsprendžiamas.

Daug metodų gali būti naudojami sprendžiant šį uždavinį. Pavyzdžiui, jungtinis gradientų metodas gali išspręsti šį uždavinį per n (duomenų didumas) žingsnių, jei tinkamai parinktas pradinis taškas (žr. šaltinį [1]).

Taip pat naudojamas Roso gradientų projekcijų metodas (žr. šaltinį [1]) šiam įgaubimo-išgaubimo FP uždavinio sprendimo radimui, kuris įrodomas konvergavimu į lokalų maksimumą su blogiausio atvejo tiesiniu konvergavimo greičiu. Be to, lokalus maksimumas šiame uždavinyje tiksliai bus globaliu maksimumu.

Iš Lemos 2 galima pamatyti, kad nelygybė (4.7) tampa lygybe prie maksimalaus taško. Optimalus b , žymimas b^* , gaunamas taip:

$$b^* = a^{*T} \bar{y} + \kappa(\beta^*)\sqrt{a^{*T} \Sigma_y a^*} = a^{*T} \bar{x} + \kappa(\alpha^*)\sqrt{a^{*T} \Sigma_x a^*}. \quad (4.13)$$

4.3 Branduolio metodas

Jo idėja yra tokia: originalūs duomenys atvaizduojami į didelio (begalinio) matavimo erdvę, kurioje randamas optimalus klasifikatorius. Panaudojant branduolių išskaidymo į skaliarinę sandaugą toje erdvėje idėją, grįžtama atgal į n -matę Euklidinę erdvę.

Tegu $\{x_i\}_{i=1}^{N_x}$ ir $\{y_j\}_{j=1}^{N_y}$ žymi apmokymo duomenis atitinkamai klasei X ir klasei Y ; pažymėkim taip: $x \rightarrow \varphi(x) \sim (\overline{\varphi(x)}, \Sigma_{\varphi(x)})$ ir $y \rightarrow \varphi(y) \sim (\overline{\varphi(y)}, \Sigma_{\varphi(y)})$, kur $\varphi: \mathbb{R}^n \rightarrow \mathbb{R}^f$ yra pažymėjimų funkcija. Atitinkamas tiesinis klasifikatorius erdvėje \mathbb{R}^f yra $a^T \varphi(z) = b$, kur $a, \varphi(z) \in \mathbb{R}^f$ ir $b \in \mathbb{R}$. Panašiai transformuotas FP optimizavimas BMPM gali būti užrašytas kaip:

$$\max_{a \neq 0} \frac{1 - \kappa(\gamma) \sqrt{a^T \Sigma_{\varphi(y)} a}}{\sqrt{a^T \Sigma_{\varphi(x)} a}} \quad \text{toks kad} \quad a^T (\overline{\varphi(x)} - \overline{\varphi(y)}) = 1. \quad (4.14)$$

Norint padaryti, kad branduolys veiktų, reikia pavaizduoti galutinę sprendimų hiperplokštumą ir optimizuoti branduolio formą $K(z_1, z_2) = \varphi(z_1)^T \varphi(z_2)$, būtent vidinę duomenų pažymėjimų sandaugos formą.

Optimizavimo ir sprendimų hiperplokštuma pertvarkoma į branduolio formą. Tegu $a = a_p + a_v$, kur a_p , yra a projekcija erdvėje, apimančioje visus apmokymo duomenis, t.y. $\{\varphi(x_i)\}_{i=1}^{N_x}$ ir $\{\varphi(y_j)\}_{j=1}^{N_y}$, ir a_v yra ortogonalūs a komponentai šioje apjungtoje erdvėje; komponentas a_v bus laikomas artėjančiu prie nulio optimizavime (4.14) naudojant $a_v^T \varphi(x_i) = 0$ ir $a_v^T \varphi(y_j) = 0$. Tai reiškia, kad optimalus a yra erdvėje, apimančioje apmokymų duomenis ir todėl gali būti aprašytas kaip tiesinė apmokymo duomenų sujungimo forma, t.y.

$$a = \sum_{i=1}^{N_x} \mu_i \varphi(x_i) + \sum_{j=1}^{N_y} \nu_j \varphi(y_j), \quad (4.15)$$

kur koeficientai $\mu_i, \nu_j \in \mathbb{R}$, $i = 1, \dots, N_x$ ir $j = 1, \dots, N_y$.

Pakeičiame (4.15) ir šiuos keturis įvedimo parametrų įverčius

$$\overline{\varphi(x)} = \frac{1}{N_x} \sum_{i=1}^{N_x} \varphi(x_i),$$

$$\overline{\varphi(y)} = \frac{1}{N_y} \sum_{j=1}^{N_y} \varphi(y_j), \quad \sum_{\varphi(x)} = \frac{1}{N_x} \sum_{i=1}^{N_x} (\varphi(x_i) - \overline{\varphi(x)})(\varphi(x_i) - \overline{\varphi(x)})^T,$$

$$\sum_{\varphi(y)} = \frac{1}{N_y} \sum_{j=1}^{N_y} (\varphi(y_j) - \overline{\varphi(y)})(\varphi(y_j) - \overline{\varphi(y)})^T$$

į optimizavimo uždavinį (4.14), tuomet gauname branduolizuotąją versiją:

$$\max_{w \neq 0} \frac{1 - \kappa(\gamma) \sqrt{\frac{1}{N_y} w^T \tilde{K}_y^T \tilde{K}_y w}}{\sqrt{\frac{1}{N_x} w^T \tilde{K}_x^T \tilde{K}_x w}} \quad \text{toks, kad } w^T (\tilde{k}_x - \tilde{k}_y) = 1. \quad (4.16)$$

F-lėje (16) $w = [\mu_1, \dots, \mu_{N_x}, \nu_1, \dots, \nu_{N_y}]^T$ ir $\tilde{k}_x, \tilde{k}_y \in R^{N_x + N_y}$, kai

$$[\tilde{k}_x]_i = \frac{1}{N_x} \sum_{j=1}^{N_x} K(x_j, z_i), \quad [\tilde{k}_y]_i = \frac{1}{N_y} \sum_{j=1}^{N_y} K(y_j, z_i), \quad \text{kur } z_i = x_i, i = 1, 2, \dots, N_x \text{ ir}$$

$z_i = x_{i-N_x}, i = N_x + 1, N_x + 2, \dots, N_x + N_y$. \tilde{K} aprašomas taip:

$$\tilde{K} = \begin{pmatrix} \tilde{K}_x \\ \tilde{K}_y \end{pmatrix} = \begin{pmatrix} \tilde{K}_x - 1_{N_x} \tilde{k}_x^T \\ \tilde{K}_y - 1_{N_y} \tilde{k}_y^T \end{pmatrix},$$

kur 1_{N_x} yra N_x -dimensijos vektoriaus stulpelis su visų elementų reikšmėmis, lygiomis 1; 1_{N_y} apibrėžiamas panašiai. N_x ir N_y yra duomenų taškų numeriai atitinkamai klasei X ir klasei Y . K_x ir K_y yra matricos, sudaromos pagal pirmą N_x eilutę ir paskutinę N_y eilutę Gramo matricos K , kuri apbrėžiama kaip

$$K_{ij} = \varphi(z_i)^T \varphi(z_j).$$

Taip pat optimalų b branduolizuotoje versijoje, atvaiztuotoje b*, galime gauti

$$b^* = w^{*T} \tilde{k}_y + \kappa(\beta^*) \sqrt{\frac{1}{N_y} w^{*T} \tilde{K}_y^T \tilde{K}_y w^*} = w^{*T} \tilde{k}_x + \kappa(\alpha^*) \sqrt{\frac{1}{N_x} w^{*T} \tilde{K}_x^T \tilde{K}_x w^*},$$

kur w^*, α^* ir β^* yra optimalios reikšmės duotos pagal anksčiau paminėtą optimizavimo procedūrą. Branduolizuota sprendimų hiperplokštuma gali būti užrašyta taip:

$$f(z) = \sum_{i=1}^{N_x} w_i^* K(z, x_i) + \sum_{i=1}^{N_y} w_{N_x+i}^* K(z, y_i) - b^*.$$

Po branduolizavimo kovariacijų matricos dimensija turi būti tokia pati, kaip ir duomenų taškų skaičius. Roso gradiento metodas yra netinkamas spręsti šį didelės apimties optimizavimo uždavinį, todėl parinktas parametrinis metodas (žr. šaltinį [10]) branduolizuoto trupmeninio programavimo (FP) uždavinio sprendimui. Tačiau naudojama tik nebranduolizuota versija algoritmo pavaizdavimui, kai (4.16) turi formą, panašią į nebranduolizuotą versiją (4.12). Pagal parametrinį metodą trupmeninė funkcija $\frac{f(a)}{g(a)}$ gali būti pakartotinai optimizuota dviem žingsniais:

Žingsnis 1: randam a pagal $f(a) - \lambda g(a)$ maksimizavimą strityje A , kur $\lambda \in R$ yra naujai įvestas parametras.

Žingsnis 2: atnaujinam λ pagal $\frac{f(a)}{g(a)}$.

Pagal šaltinį [10], λ maksimumas, būtent maksimumas FP uždavinio sprendimo, garantuoja konvergavimą per anksčiau nurodytas iteracijų serijas.

Toliau parinktas metodas spręsti maksimizavimo problemą pagal žingsnį 1. Pakeičiami $f(a)$ ir $g(a)$; optimizavimo problema išplečiama taip :

$$\max_{a \neq 0} 1 - \kappa(\gamma) \sqrt{a^T \Sigma_y a} - \lambda \sqrt{a^T \Sigma_x a} \quad \text{toks, kad } a^T (\bar{x} - \bar{y}) = 1. \quad (4.17)$$

Lygtis (17) yra ekvivalenti $\min_a \kappa(\gamma) \sqrt{a^T \Sigma_y a} + \lambda \sqrt{a^T \Sigma_x a}$ su tokiais pat apribojimais. Pagal užrašymą $a = a_0 + Fu$, kur $a_0 = (\bar{x} - \bar{y}) / \|\bar{x} - \bar{y}\|_2^2$ ir $F \in \mathbb{R}^{n \times (n-1)}$ yra ortogonali matrica, kurios stulpeliai apima ortogonalinių vektorių $\bar{x} - \bar{y}$ poerdvį, ir ekvivalenčią formą (trupmena $1/2$ su kiekvienu periodu sumažėja), pašalinus apribojimus galima gauti :

$$\min_{u, \eta > 0, \xi > 0} \eta + \frac{\lambda^2}{\eta} \left\| \sum_x^{1/2} (a_0 + Fu) \right\|_2^2 + \xi + \frac{\kappa(\gamma)^2}{\xi} \left\| \sum_y^{1/2} (a_0 + Fu) \right\|_2^2. \quad (4.18)$$

Ši optimizavimo forma yra labai panaši į MPM (žr. šaltinį [4]) ir gali būti taip pat sprendžiama panaudojus iteracinį mažiausių kvadratų metodą (žr. šaltinius [1], [4]).

MPM - tai neseniai pasiūlytas naujas klasifikatorius, numatantis blogiausio atvejo ribą klaidos tikimybei. Eliminavus prielaidą apie kiekvienos MPM klasės nepaslinktus svorius, išplėtojamas svarbus įrankis BMPM, kuris yra pirmas kiekybinis metodas, kontroliuojantis, kaip sprendimų hiperploštuma juda į svarbesnės klasifikavimo klasės pusę. Šis modelis yra transformuojamas į įgaubimo-išgaubimo trupmeninį programavimo uždavinį. Modelis iliustruojamas naudojant realius duomenis.

Pirma, modelis daug priklauso nuo vidurkio ir kovariacijų matricų įverčių, kaip tiksliai juos vertiname. Antra, ar yra efektyvesnių metodų spręsti trupmeninio uždavinio optimizavimo uždavinį? Ar galima dekompozicijos metodams naudoti Gramo matricą ir taip pat pagreitinti apmokymą mažiausių kvadratų metodu. Schemos pratęsimas daugelio klasių atžvilgiu yra taip pat vienas iš aktualiausių temų artimiausioje ateityje.

5. Procedūrų bandymai ir rezultatai

Aprašytos procedūros išbandytos su UAB “NB Europe” duomenimis. Jie gauti iš liejamojo formavimo įrenginio. Iš viso turime 4260 stebėjimų. Pradiniai duomenys sudaryti iš 37 kintamųjų:

- x1. Cycle Time [s]; (ciklo laikas)
- x2. Mold Closing Time [s]; (formos uždarymo laikas)
- x3. Clamp Tonnage [t]; (spaustuvo talpa)
- x4. Shot Size [mm]; (šovimo dydis)
- x5. Cushion [mm]; (pagalvėlė)
- x6. Shot Length [mm]; (šovimo ilgis)
- x7. Injection Time [s]; (įpurškimo laikas)
- x8. Transition Position [mm]; (pereinamoji padėtis)
- x9. Transition Pressure [bar]; (pereinamasis suspaudimas)
- x10. Max. Injection Pressure [bar]; (max. įpurškimo spaudimas)
- x11. Hold Pressure 1 [bar]; (išlaikymo spaudimas)
- x12. Hold Pressure 2 [bar];
- x13. Hold Pressure 3 [bar];
- x14. Screw Back Pressure 1 [bar]; (sraigto galinis suspaudimas)
- x15. Screw Back Pressure 2 [bar];
- x16. Screw Back Pressure 3 [bar];
- x17. Screw Speed [rpm]; (sraigto greitis)
- x18. Screw Run Time [s]; (sraigto veikimo laikas)
- x19. Transfer Time [s]; (perdavimo laikas)
- x20. Screw Recovery Time [s]; (sraigto regeneravimo laikas)
- x21. Shooting Pot Pack [bar]; (šaudymo puodynės pakelis)
- x22. Effective Cooling Time [s]; (efektyvus atvėsimo laikas)
- x23. Nozzle Adaptor [°C]; (purkštuko jungiklis)
- x24. Mold Open Time [s]; (formos atidarymo laikas)
- x25. Ejector Forward Time [s]; (išmetimo aparato paspartinimo laikas)
- x26. Oil [°C]; (tepalas, aliejus)
- x27. Resin [°C]; (plastinė masė)
- x28. Barrel Zone 1 [°C];
- x29. Barrel Zone 2 [°C];
- x30. Barrel Zone 3 [°C];
- x31. Barrel Zone 4 [°C];
- x32. Barrel Zone 5 [°C];
- x33. Barrel Zone 6 [°C]; (veleno zona)
- x34. Barrel Head Extension [°C]; (veleno galvutės pailginimas)
- x35. Shooting Pot 1 [°C]; (šaudymo puodynė)
- x36. Shooting Pot 2 [°C];
- x37. Distributor [°C]; (skirstytuvas)

Kadangi kintamieji, Transition Position, Screw Speed, Effective Cooling Time nesikeičia – jie konstantos, todėl jie į bandymus neįtraukiami.

Duomenų charakteristikas matome pavaizduotas lentelėje 1.

Lentelė 1.

kintamasis	vidurkis	Standart.	+3sigma	-3sigma
------------	----------	-----------	---------	---------

		nuokrypis		
x1	17,99999	0,030043	18,09012	17,9099
x2	16,99958	0,010974	17,0325	16,9667
x3	15,99998	0,001907	16,0057	15,9943
x4	14,99977	0,002726	15,00795	14,9916
x5	13,99999	0,053874	14,1616	13,8384
x6	13,00002	0,005507	13,0165	12,9835
x7	12,00004	0,011367	12,0341	11,9659
x8	9,999968	0,047411	10,1422	9,8577
x9	8,999996	0,041967	9,1259	8,8741
x10	8,000002	0,002716	8,0082	7,9919
x11	7,000022	0,005117	7,0154	6,9847
x12	5,999999	0,025353	6,0761	5,9239
x13	4,999995	0,028247	5,0847	4,9153
x14	2,999998	0,054936	3,1648	2,8352
x15	1,999999	0,0772	2,2316	1,7684
x16	0,999992	0,015565	1,0467	0,9533
x17	19	0,074403	19,2232	18,7768
x18	18	0,030055	18,0902	17,9098
x19	16,00002	0,019042	16,0571	15,9429
x20	14,99999	0,022158	15,0665	14,9335
x21	14,00002	0,428864	15,2866	12,7134
x22	13,00013	0,013452	13,0405	12,9598
x23	12	0,519756	13,5593	10,4407
x24	11,00002	0,011954	11,0359	10,9642
x25	10,00001	0,010052	10,0302	9,9699
x26	8,999998	0,010124	9,0304	8,9696
x27	8,000001	0,009733	8,0292	7,9708
x28	7	0,010081	7,0302	6,9698
x29	6,000007	0,009038	6,0271	5,9729
x30	4,999992	0,004645	5,0139	4,9861
x31	4,000007	0,00214	4,0064	3,9936
x32	2,99997	0,002089	3,0062	2,9937
x33	2,000018	0,003714	2,0112	1,9889
x34	1,000012	0,003183	1,0096	0,9905

Panaudojus randomizavimo procedūras, \bar{x} apmokymo duomenis atrinkta 350 stebėjimų, o \bar{x} testinius duomenis – 158. Taigi gavome duomenis, kuriuos panaudosime klasifikavimo procedūrose. Iš pradžių klasifikuosim patikrintu metodu - logistinė regresija, skaičiavimus atliksime SAS'o pagalba. Rezultatus matome pavaizduotus lentelėse.

Logistinės regresijos skaičiavimų rezultatai.

5.1. lentelė. Apmokymo duomenys. Klasifikavimo taisyklės slenkstis $p^*=0.1$

η	0	1	Σ
--------	---	---	----------

Y			
0	0.982	0.018	1
1	0.887	0.113	1

5.2. lentelē. Apmokymo duomenys. Klasifikavimo taisyklės slenkstis $p^*=0.2$

Y	η	0	1	Σ
0		0.971	0.029	1
1		0.803	0.197	1

5.3. lentelē. Apmokymo duomenys. Klasifikavimo taisyklės slenkstis $p^*=0.3$

Y	η	0	1	Σ
0		0.953	0.047	1
1		0.746	0.254	1

5.4. lentelē. Apmokymo duomenys. Klasifikavimo taisyklės slenkstis $p^*=0.4$

Y	η	0	1	Σ
0		0.939	0.061	1
1		0.620	0.380	1

5.5. lentelē. Apmokymo duomenys. Klasifikavimo taisyklės slenkstis $p^*=0.5$

Y	η	0	1	Σ
0		0.928	0.072	1
1		0.592	0.408	1

5.6. lentelē. Apmokymo duomenys. Klasifikavimo taisyklės slenkstis $p^*=0.6$

Y	η	0	1	Σ
0		0.896	0.104	1
1		0.521	0.479	1

5.7. lentelē. Apmokymo duomenys. Klasifikavimo taisyklės slenkstis $p^*=0.7$

Y	η	0	1	Σ
0		0.860	0.140	1
1		0.338	0.662	1

5.8. lentelė. Apmokymo duomenys. Klasifikavimo taisyklės slenkstis $p^*=0.8$

$Y \backslash \eta$	0	1	Σ
0	0.778	0.222	1
1	0.254	0.746	1

5.9. lentelė. Apmokymo duomenys. Klasifikavimo taisyklės slenkstis $p^*=0.9$

$Y \backslash \eta$	0	1	Σ
0	0.649	0.351	1
1	0.225	0.775	1

5.10. lentelė. Testiniai duomenys. Klasifikavimo taisyklės slenkstis $p^*=0.1$

$Y \backslash \eta$	0	1	Σ
0	0.922	0.078	1
1	0.941	0.059	1

5.11. lentelė. Testiniai duomenys. Klasifikavimo taisyklės slenkstis $p^*=0.2$

$Y \backslash \eta$	0	1	Σ
0	0.901	0.099	1
1	0.882	0.118	1

5.12. lentelė. Testiniai duomenys. Klasifikavimo taisyklės slenkstis $p^*=0.3$

$Y \backslash \eta$	0	1	Σ
0	0.887	0.113	1
1	0.882	0.118	1

5.13. lentelė. Testiniai duomenys. Klasifikavimo taisyklės slenkstis $p^*=0.4$

$Y \backslash \eta$	0	1	Σ
0	0.879	0.121	1
1	0.824	0.176	1

5.14. lentelė. Testiniai duomenys. Klasifikavimo taisyklės slenkstis $p^*=0.5$

$Y \backslash \eta$	0	1	Σ
0	0.858	0.142	1
1	0.706	0.294	1

5.15. lentelė. Testiniai duomenys. Klasifikavimo taisyklės slenkstis $p^*=0.6$

$Y \backslash \eta$	0	1	Σ
0	0.858	0.142	1
1	0.647	0.353	1

5.16. lentelė. Testiniai duomenys. Klasifikavimo taisyklės slenkstis $p^*=0.7$

$Y \backslash \eta$	0	1	Σ
0	0.844	0.156	1
1	0.647	0.353	1

5.17. lentelė. Testiniai duomenys. Klasifikavimo taisyklės slenkstis $p^*=0.8$

$Y \backslash \eta$	0	1	Σ
0	0.823	0.177	1
1	0.647	0.353	1

5.18. lentelė. Testiniai duomenys. Klasifikavimo taisyklės slenkstis $p^*=0.9$

$Y \backslash \eta$	0	1	Σ
0	0.773	0.227	1
1	0.588	0.412	1

Dabar panaudosime bandymams naują metodą BMPM. Iš pradžių lentelėse pavaizduosime tiesinį BMPM.

5.19. lentelė. Apmokymo duomenys. Blogiausio atvejo tikslumas=0.5, geriausio atvejo tikslumas=0.84

η	0	1	Σ

Y			
0	0.842	0.158	1
1	0.549	0.451	1

5.20. lentelė. Apmokymo duomenys. Blogiausio atvejo tikslumas =0.6, geriausio atvejo tikslumas =0.75

Y	η	0	1	Σ
0		0.785	0.215	1
1		0.409	0.591	1

5.21. lentelė. Apmokymo duomenys. Blogiausio atvejo tikslumas =0.7, geriausio atvejo tikslumas =0.63

Y	η	0	1	Σ
0		0.713	0.287	1
1		0.296	0.704	1

5.22. lentelė. Apmokymo duomenys. Blogiausio atvejo tikslumas =0.8, geriausio atvejo tikslumas =0.5

Y	η	0	1	Σ
0		0.555	0.445	1
1		0.226	0.774	1

5.23. lentelė. Apmokymo duomenys. Blogiausio atvejo tikslumas =0.9, geriausio atvejo tikslumas =0.5

Y	η	0	1	Σ
0		0.333	0.667	1
1		0.113	0.887	1

5.24. lentelė. Testiniai duomenys. Blogiausio atvejo tikslumas =0.5, geriausio atvejo tikslumas =0.84

Y	η	0	1	Σ
0		0.823	0.177	1
1		0.588	0.412	1

5.25. lentelė. Testiniai duomenys. Blogiausio atvejo tikslumas =0.6, geriausio atvejo tikslumas =0.75

$Y \backslash \eta$	0	1	Σ
0	0.681	0.319	1
1	0.529	0.471	1

5.26. lentelė. Testiniai duomenys. Blogiausio atvejo tikslumas =0.7, geriausio atvejo tikslumas =0.63

$Y \backslash \eta$	0	1	Σ
0	0.496	0.504	1
1	0.412	0.588	1

5.27. lentelė. Testiniai duomenys. Blogiausio atvejo tikslumas =0.8, geriausio atvejo tikslumas =0.5

$Y \backslash \eta$	0	1	Σ
0	0.418	0.582	1
1	0.353	0.647	1

5.28. lentelė. Testiniai duomenys. Blogiausio atvejo tikslumas =0.9, geriausio atvejo tikslumas =0.5

$Y \backslash \eta$	0	1	Σ
0	0.277	0.773	1
1	0.235	0.765	1

Dabar lentelėse pavaizduosim BMPM branduolio metodą.

5.29. lentelė. Apmokymo duomenys. Blogiausio atvejo tikslumas =0.6, geriausio atvejo tikslumas =0.89

$Y \backslash \eta$	0	1	Σ

Y			
0	0.885	0.115	1
1	0.437	0.563	1

5.30. lentelė. Apmokymo duomenys. Blogiausio atvejo tikslumas =0.7, geriausio atvejo tikslumas =0.85

Y	η	0	1	Σ
0		0.867	0.133	1
1		0.352	0.648	1

5.31. lentelė. Apmokymo duomenys. Blogiausio atvejo tikslumas =0.8, geriausio atvejo tikslumas =0.81

Y	η	0	1	Σ
0		0.828	0.172	1
1		0.268	0.732	1

5.32. lentelė. Apmokymo duomenys. Blogiausio atvejo tikslumas =0.9, geriausio atvejo tikslumas =0.73

Y	η	0	1	Σ
0		0.763	0.237	1
1		0.211	0.789	1

5.33. lentelė. Testiniai duomenys. Blogiausio atvejo tikslumas =0.6, geriausio atvejo tikslumas =0.89

Y	η	0	1	Σ
0		0.950	0.05	1
1		0.765	0.235	1

5.34. lentelė. Testiniai duomenys. Blogiausio atvejo tikslumas =0.7, geriausio atvejo tikslumas =0.85

Y	η	0	1	Σ
0		0.915	0.085	1
1		0.765	0.235	1

5.35. lentelė. Testiniai duomenys. Blogiausio atvejo tikslumas =0.8, geriausio atvejo tikslumas =0.81

$Y \backslash \eta$	0	1	Σ
0	0.858	0.142	1
1	0.765	0.235	1

5.36. lentelė. Testiniai duomenys. Blogiausio atvejo tikslumas =0.9, geriausio atvejo tikslumas =0.73

$Y \backslash \eta$	0	1	Σ
0	0.759	0.241	1
1	0.765	0.235	1

Panagrinėkime kelias lenteles. Paimkime po vieną lentelę iš kiekvieno metodo, pvz., tegul tai būna testiniai logistinės regresijos duomenys iš 5.16 lentelės, testiniai tiesinio BMPM duomenys iš 5.26 lentelės ir BMPM branduolio metodo testiniai duomenys iš 5.34 lentelės. Matome, kad logistinė regresija gerą gaminį atspėja su tikimybe 0.844, o blogą - su tikimybe 0.353. Gamintojo rizika šiuo atveju yra 0.156, o vartotojo rizika - 0.647. Tiesinis BMPM gerą gaminį atspėja su tikimybe 0.496, o blogą – su tikimybe 0.588. Šiuo atveju gamintojo rizika - 0.504, vartotojo – 0.412. BMPM branduolio metodas gerą gaminį atspėja su tikimybe 0.915, blogą gaminį atspėja su tikimybe 0.235. Gamintojo rizika - 0.085, o vartotojo – 0.765.

6. Išvados

Palyginę šiuos tris metodus matome, kad geriausiai gerus gaminius prognozuoja BMPM branduolio metodas, tačiau jis prastai atspėja blogus gaminius. Blogus gaminius geriausiai prognozuoja tiesinis BMPM metodas, tačiau jis prastai prognozuoja gerus gaminius. Logistinė regresija, palyginus su BMPM branduolio metodu, truputį prasčiau prognozuoja gerus gaminius, bet geriau - blogus, o su tiesiniu BMPM metodu palyginus, atvirkščiai, prasčiau prognozuoja blogus gaminius, bet

geriau – gerus. Taigi, jeigu norime, kad būtų kuo mažesnė gamintojo rizika, geriau naudoti BMPM branduolio metodą, jeigu norime sumažinti vartotojo riziką, tuomet geriau naudoti tiesinį BMPM metodą, o jeigu norime minimizuoti abi rizikas, tuomet geriausia naudoti patikrintą metodą - logistinę regresiją.

Summary

In this work the Biased Minimax Probability Machine model is used for quality control system improvement at the JSC “NB Europe”. The logistic regression method is used too. The comparison of these methods was done.

Literatūros sarakšas

- [1] D. P. Bertsekas. “**Nonlinear Programming**”. Athena Scientific, Belmont, Massashusetts, 2nd edition, 1999.

- [2] Kaizhu Huang, Haiqin Yang, Irwin King, Michael R. Lyu and Laiwan Chan. “**Biased Minimax Probability Machine for Medical Diagnosis**”. Department of Computer Science and Engineering The Chinese University of Hong Kong, 2004.
- [3] A. W. Marshall and I. Olkin. “**Multivariate chebyshev inequalities**”. Annals of Mathematical Statistics, 1960.
- [4] G. R. G. Lanckriet, L. E. Gheoui, C. Bhattacharyya, and M. I. Jordan. “**Minimax probability machine**”. In Advances in Neural Information Processing Systems (NIPS), 2001.
- [5] G. R. G. Lanckriet, L. E. Gheoui, C. Bhattacharyya, and M. I. Jordan. “**A robust minimax approach to classification**”. Journal of Machine Learning Research, 3:555-582, 2002.
- [6] M. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret. “**An analysis of bayesian classifiers**”. Applications of second order cone programming. Linear Algebra and its Applications, 284:193-228, 1998.
- [7] Y. Nesterov and Nemirovsky. “**Interior point polynomial methods in convex programming: Theory and applications**”. SIAM, Philadelphia, PA, 1994.
- [8] E. Osuna, R. Freund, and F. Girosi. “**Support Vektor Machines: Training and Applications**”. Technical Report AIM-1602, MIT, 1997.
- [9] I. Popescu and D. Bertsimas. “**Optimal innequalities in probability theory: A convex optimization approach**”. Technical Report TM62, INSEAD, 2001.
- [10] S. Schaible. “**Fractional programming**”. In R. Horst and P. M. Pardalos, editors, Handbook of Global Optimization, Nonconvex Optimization and Its Applications, pages 495-608. Kluwer Academic, Dordrecht-Boston-London, 1995.
- [11] B. Schölkopf and A. Smola. “**Learning with Kernels**”. MIT Press, Cambridge, MA, 2002.
- [12] V. N. Valpnik. “**The Nature of Statistical Learning Theory**”. Springer, New York, 2nd edition, 1999.
- [13] Руководство пользователя машины. Машина литьевого формования Index™ 90 серии G, 2001.
- [14] J. Kruopis, A. Vaišvila, R. Kalnius. “**Mechatronikos gaminių kokybė. Atrankinė kontrolė**” Vniaus universiteto leidykla, 2005.
- [15] Vytautas Kazakevičius. “**Logistinė regresija**”. Konspektai, 2005.
- [16] Matlab Help.

Number of Observations 350
 Model binary logit
 Optimization Technique Fisher's scoring

Response Profile

Ordered Value	y	Total Frequency
1	1	279
2	-1	71

Probability modeled is y=1.

Model Convergence Status

Convergence criterion (GCONV=1E-8) satisfied.

Model Fit Statistics

Criterion	Intercept Only	Intercept and Covariates
AIC	355.036	278.297
SC	358.894	413.325
-2 Log L	353.036	208.297

Testing Global Null Hypothesis: BETA=0

Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	144.7392	34	<.0001
Score	131.4713	34	<.0001
wald	71.5308	34	0.0002

The LOGISTIC Procedure

Analysis of Maximum Likelihood Estimates

Parameter	DF	Estimate	Standard Error	wald Chi-Square	Pr > ChiSq
Intercept	1	7178.9	10826.3	0.4397	0.5073
x1	1	20.5468	13.8897	2.1883	0.1391
x2	1	-31.0754	19.2401	2.6087	0.1063
x3	1	-136.5	104.5	1.7060	0.1915
x4	1	-818.3	561.1	2.1271	0.1447
x5	1	71.3661	53.8290	1.7577	0.1849
x6	1	780.4	560.7	1.9370	0.1640
x7	1	39.5600	29.8685	1.7542	0.1853
x8	1	-4.9918	10.1329	0.2427	0.6223
x9	1	-6.1519	12.4646	0.2436	0.6216
x10	1	-24.4252	76.9236	0.1008	0.7508
x11	1	86.4963	43.4213	3.9682	0.0464
x12	1	-6.9743	7.9047	0.7784	0.3776
x13	1	-2.4158	6.9130	0.1221	0.7267
x14	1	556.4	921.5	0.3646	0.5460
x15	1	-138.2	229.6	0.3621	0.5474
x16	1	7.7896	13.0455	0.3565	0.5504
x17	1	-405.4	670.3	0.3658	0.5453
x18	1	0.8773	7.0345	0.0156	0.9008
x19	1	28.0474	12.4378	5.0851	0.0241
x20	1	18.2431	15.7647	1.3391	0.2472
x21	1	-2.2451	0.5302	17.9299	<.0001
x22	1	-47.8870	16.2763	8.6562	0.0033
x23	1	-0.9895	1.4511	0.4650	0.4953
x24	1	23.2283	64.6899	0.1289	0.7195
x25	1	-8.4485	86.6984	0.0095	0.9224
x26	1	41.5307	89.6114	0.2148	0.6430
x27	1	-164.6	96.1973	2.9278	0.0871
x28	1	12.0847	91.6176	0.0174	0.8951
x29	1	108.1	66.1321	2.6704	0.1022
x30	1	148.1	35.5575	17.3495	<.0001
x31	1	114.3	89.0319	1.6469	0.1994
x32	1	140.1	112.5	1.5498	0.2132
x33	1	80.5710	64.3426	1.5681	0.2105
x34	1	60.9009	59.8793	1.0344	0.3091

Odds Ratio Estimates

Effect	Point Estimate	95% wald Confidence Limits	
x1	>999.999	0.001	>999.999
x2	<0.001	<0.001	760.877
x3	<0.001	<0.001	>999.999

The LOGISTIC Procedure

Odds Ratio Estimates

Effect	Point Estimate	95% Wald Confidence Limits	
x4	<0.001	<0.001	>999.999
x5	>999.999	<0.001	>999.999
x6	>999.999	<0.001	>999.999
x7	>999.999	<0.001	>999.999
x8	0.007	<0.001	>999.999
x9	0.002	<0.001	>999.999
x10	<0.001	<0.001	>999.999
x11	>999.999	4.024	>999.999
x12	<0.001	<0.001	>999.999
x13	0.089	<0.001	>999.999
x14	>999.999	<0.001	>999.999
x15	<0.001	<0.001	>999.999
x16	>999.999	<0.001	>999.999
x17	<0.001	<0.001	>999.999
x18	2.404	<0.001	>999.999
x19	>999.999	39.241	>999.999
x20	>999.999	<0.001	>999.999
x21	0.106	0.037	0.299
x22	<0.001	<0.001	<0.001
x23	0.372	0.022	6.389
x24	>999.999	<0.001	>999.999
x25	<0.001	<0.001	>999.999
x26	>999.999	<0.001	>999.999
x27	<0.001	<0.001	>999.999
x28	>999.999	<0.001	>999.999
x29	>999.999	<0.001	>999.999
x30	>999.999	>999.999	>999.999
x31	>999.999	<0.001	>999.999
x32	>999.999	<0.001	>999.999
x33	>999.999	<0.001	>999.999
x34	>999.999	<0.001	>999.999

Association of Predicted Probabilities and Observed Responses

Percent Concordant	90.7	Somers' D	0.814
Percent Discordant	9.2	Gamma	0.815
Percent Tied	0.1	Tau-a	0.264
Pairs	19809	c	0.907

The SAS System 19:03 Thursday, May
 The LOGISTIC Procedure
 Classification Table

Prob Level	Correct		Incorrect		Percentages				
	Event	Non-Event	Event	Non-Event	Correct	Sensitivity	Specificity	False POS	False NEG
0.100	274	8	63	5	80.6	98.2	11.3	18.7	38.5
0.200	271	14	57	8	81.4	97.1	19.7	17.4	36.4
0.300	266	18	53	13	81.1	95.3	25.4	16.6	41.9
0.400	262	27	44	17	82.6	93.9	38.0	14.4	38.6
0.500	259	29	42	20	82.3	92.8	40.8	14.0	40.8
0.600	250	34	37	29	81.1	89.6	47.9	12.9	46.0
0.700	240	47	24	39	82.0	86.0	66.2	9.1	45.3
0.800	217	53	18	62	77.1	77.8	74.6	7.7	53.9
0.900	181	55	16	98	67.4	64.9	77.5	8.1	64.1

Testiniai duomenys

The SAS System 19:03 Thursday, May
 The LOGISTIC Procedure
 Model Information

Data Set	WORK.BANDOM2	
Response Variable	y	y
Number of Response Levels	2	
Number of Observations	158	
Model	binary logit	
Optimization Technique	Fisher's scoring	

Response Profile

Ordered Value	y	Total Frequency
1	1	141
2	-1	17

Probability modeled is y=1.

Model Convergence Status

Convergence criterion (GCONV=1E-8) satisfied.

Model Fit Statistics

Criterion	Intercept Only	Intercept and Covariates
AIC	109.900	117.960
SC	112.963	225.151
-2 Log L	107.900	47.960

Testing Global Null Hypothesis: BETA=0

Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	59.9403	34	0.0039
Score	53.6306	34	0.0174
Wald	14.6026	34	0.9985

The LOGISTIC Procedure

Analysis of Maximum Likelihood Estimates

Parameter	DF	Estimate	Standard Error	wald Chi-Square	Pr > ChiSq
Intercept	1	46446.0	42024.5	1.2215	0.2691
x1	1	18.2254	45.1949	0.1626	0.6868
x2	1	87.6276	73.3757	1.4262	0.2324
x3	1	-268.4	454.4	0.3488	0.5548
x4	1	-1487.0	3466.6	0.1840	0.6680
x5	1	131.8	330.2	0.1594	0.6897
x6	1	1209.7	3358.7	0.1297	0.7187
x7	1	-102.1	124.1	0.6777	0.4104
x8	1	24.9082	35.1880	0.5011	0.4790
x9	1	-26.2832	49.0315	0.2873	0.5919
x10	1	514.7	356.3	2.0870	0.1486
x11	1	22.2524	135.3	0.0270	0.8694
x12	1	37.8289	35.8372	1.1142	0.2912
x13	1	-56.6979	31.0981	3.3240	0.0683
x14	1	3085.2	3686.9	0.7002	0.4027
x15	1	-766.8	911.2	0.7081	0.4001
x16	1	5.2213	45.7188	0.0130	0.9091
x17	1	-2243.8	2685.0	0.6984	0.4033
x18	1	47.0384	26.7926	3.0823	0.0791
x19	1	-408.2	183.5	4.9481	0.0261
x20	1	-32.3953	57.8196	0.3139	0.5753
x21	1	-5.0243	1.7879	7.8972	0.0050
x22	1	-85.2930	68.7205	1.5405	0.2145
x23	1	-29.4313	20.4023	2.0809	0.1491
x24	1	398.7	315.8	1.5942	0.2067
x25	1	-6.7743	357.9	0.0004	0.9849
x26	1	739.0	431.4	2.9344	0.0867
x27	1	-468.5	333.6	1.9723	0.1602
x28	1	-412.8	393.5	1.1005	0.2942
x29	1	-93.4089	299.0	0.0976	0.7548
x30	1	-38.3637	281.2	0.0186	0.8915
x31	1	-87.8183	423.2	0.0431	0.8356
x32	1	-339.4	469.8	0.5218	0.4701
x33	1	-1073.9	820.6	1.7127	0.1906
x34	1	324.4	306.9	1.1171	0.2905

Odds Ratio Estimates

Effect	Point Estimate	95% wald Confidence Limits	
x1	>999.999	<0.001	>999.999
x2	>999.999	<0.001	>999.999
x3	<0.001	<0.001	>999.999

The LOGISTIC Procedure

Odds Ratio Estimates

Effect	Point Estimate	95% Wald Confidence Limits	
x4	<0.001	<0.001	>999.999
x5	>999.999	<0.001	>999.999
x6	>999.999	<0.001	>999.999
x7	<0.001	<0.001	>999.999
x8	>999.999	<0.001	>999.999
x9	<0.001	<0.001	>999.999
x10	>999.999	<0.001	>999.999
x11	>999.999	<0.001	>999.999
x12	>999.999	<0.001	>999.999
x13	<0.001	<0.001	70.340
x14	>999.999	<0.001	>999.999
x15	<0.001	<0.001	>999.999
x16	185.180	<0.001	>999.999
x17	<0.001	<0.001	>999.999
x18	>999.999	0.004	>999.999
x19	<0.001	<0.001	<0.001
x20	<0.001	<0.001	>999.999
x21	0.007	<0.001	0.219
x22	<0.001	<0.001	>999.999
x23	<0.001	<0.001	>999.999
x24	>999.999	<0.001	>999.999
x25	0.001	<0.001	>999.999
x26	>999.999	<0.001	>999.999
x27	<0.001	<0.001	>999.999
x28	<0.001	<0.001	>999.999
x29	<0.001	<0.001	>999.999
x30	<0.001	<0.001	>999.999
x31	<0.001	<0.001	>999.999
x32	<0.001	<0.001	>999.999
x33	<0.001	<0.001	>999.999
x34	>999.999	<0.001	>999.999

Association of Predicted Probabilities and Observed Responses

Percent Concordant	93.1	Somers' D	0.898
Percent Discordant	3.3	Gamma	0.931
Percent Tied	3.5	Tau-a	0.174
Pairs	2397	c	0.949

The LOGISTIC Procedure

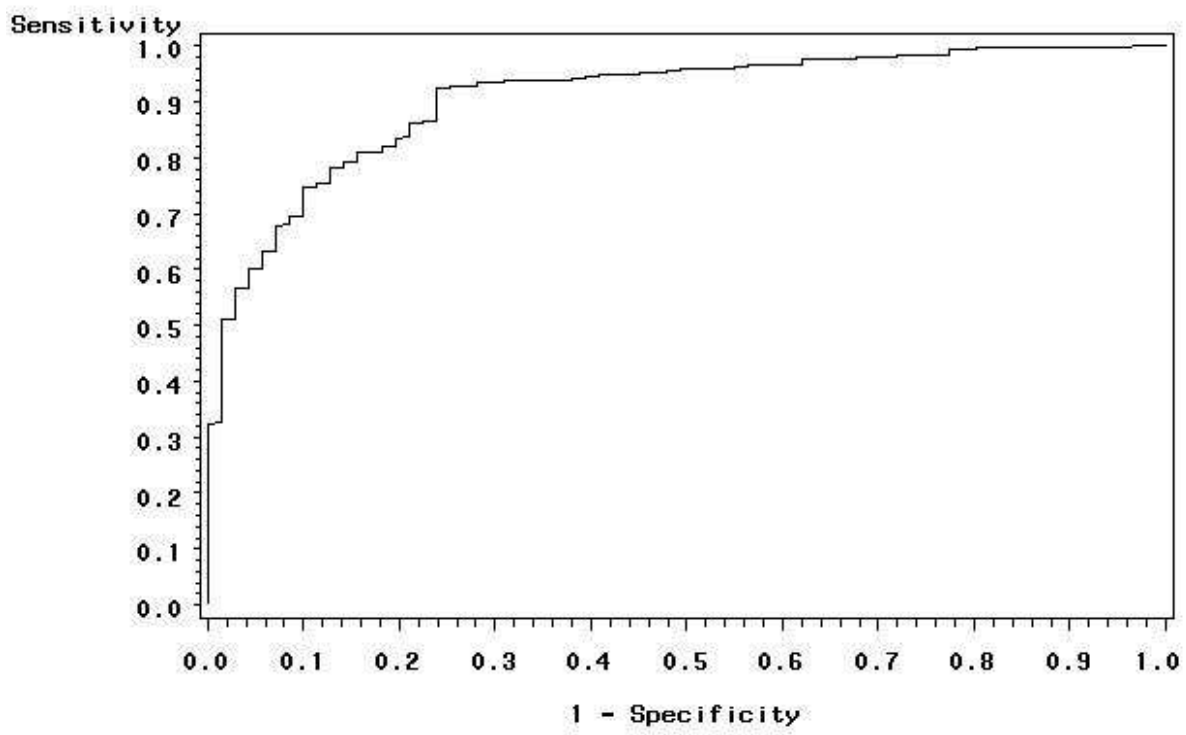
Classification Table

Prob Level	Correct		Incorrect		Correct	Percentages		False POS	False NEG
	Event	Non- Event	Event	Non- Event		Sensi- tivity	Speci- ficity		
0.100	130	1	16	11	82.9	92.2	5.9	11.0	91.7
0.200	127	2	15	14	81.6	90.1	11.8	10.6	87.5
0.300	125	2	15	16	80.4	88.7	11.8	10.7	88.9
0.400	124	3	14	17	80.4	87.9	17.6	10.1	85.0
0.500	121	5	12	20	79.7	85.8	29.4	9.0	80.0
0.600	121	6	11	20	80.4	85.8	35.3	8.3	76.9
0.700	119	6	11	22	79.1	84.4	35.3	8.5	78.6
0.800	116	6	11	25	77.2	82.3	35.3	8.7	80.6
0.900	109	7	10	32	73.4	77.3	41.2	8.4	82.1

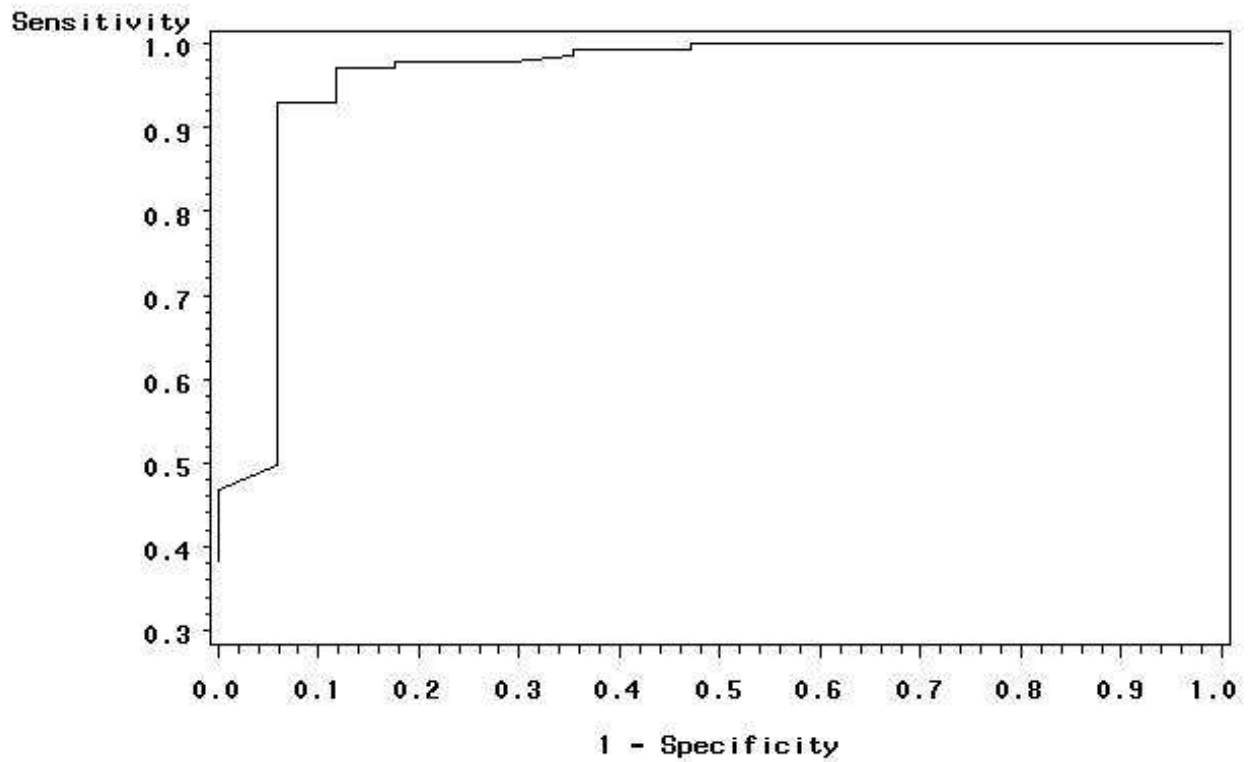
Priedas B**ROC kreivės**

Kad būtų vaizdžiau, klasifikavimo rezultatus galima pavaizduoti, nubrėžiant vadinamąją ROC kreivę (ROC – nuo angliško termino receiver operating characteristic).

Apmokymo duomenys



Testiniai duomenys



Priedas C

MATLAB'o skaičiavimai

beta0 yra 0.9, alfa yra 0.5

Underbound on correct classification of future data-Class 1: 50

Setting lower bound on correct classification of future data-Class 2: 90

Training set accuracy: 44.5714

Training set accuracy-Class 1: 33.3333
Training set accuracy-Class 2: 88.7324
Test set accuracy: 32.9114
Test set accuracy-Class 1: 27.6596
Test set accuracy-Class 2: 76.4706

beta0 yra 0.9, alfa 0.73(brand)

Underbound on correct classification of future data-Class 1: 73.9408
Setting lower bound on correct classification of future data-Class 2: 90
Training set accuracy: 76.8571
Training set accuracy-Class 1: 76.3441
Training set accuracy-Class 2: 78.8732
Test set accuracy: 70.2532
Test set accuracy-Class 1: 75.8865
Test set accuracy-Class 2: 23.5294

Kai beta0 yra 0.8, alfa yra 0.5

Underbound on correct classification of future data-Class 1: 50
Setting lower bound on correct classification of future data-Class 2: 80
Training set accuracy: 60
Training set accuracy-Class 1: 55.5556
Training set accuracy-Class 2: 77.4648
Test set accuracy: 44.3038
Test set accuracy-Class 1: 41.8440
Test set accuracy-Class 2: 64.7059

beta0 yra 0.8, alfa 0.81(brand)

Underbound on correct classification of future data-Class 1: 81.1279
Setting lower bound on correct classification of future data-Class 2: 80
Training set accuracy: 80.8571
Training set accuracy-Class 1: 82.7957
Training set accuracy-Class 2: 73.2394
Test set accuracy: 79.1139
Test set accuracy-Class 1: 85.8156
Test set accuracy-Class 2: 23.5294

beta0 yra 0.7, alfa 0.63

Underbound on correct classification of future data-Class 1: 63.7159
Setting lower bound on correct classification of future data-Class 2: 70
Training set accuracy: 71.1429
Training set accuracy-Class 1: 71.3262
Training set accuracy-Class 2: 70.4225
Test set accuracy: 50.6329
Test set accuracy-Class 1: 49.6454
Test set accuracy-Class 2: 58.8235

beta0 yra 0.7, alfa 0.85(brand)

Underbound on correct classification of future data-Class 1: 85.8293
Setting lower bound on correct classification of future data-Class 2: 70
Training set accuracy: 82.2857
Training set accuracy-Class 1: 86.7384
Training set accuracy-Class 2: 64.7887
Test set accuracy: 84.1772
Test set accuracy-Class 1: 91.4894
Test set accuracy-Class 2: 23.5294

beta0 yra 0.6, alfa 0.75

Underbound on correct classification of future data-Class 1: 75.1060
Setting lower bound on correct classification of future data-Class 2: 60
Training set accuracy: 74.5714
Training set accuracy-Class 1: 78.4946
Training set accuracy-Class 2: 59.1549
Test set accuracy: 65.8228
Test set accuracy-Class 1: 68.0851
Test set accuracy-Class 2: 47.0588

beta0 yra 0.6, alfa 0.89(brand)

Underbound on correct classification of future data-Class 1: 89.3663
Setting lower bound on correct classification of future data-Class 2: 60
Training set accuracy: 82
Training set accuracy-Class 1: 88.5305
Training set accuracy-Class 2: 56.3380
Test set accuracy: 87.3418
Test set accuracy-Class 1: 95.0355
Test set accuracy-Class 2: 23.5294

beta0 yra 0.5, alfa 0.84

Underbound on correct classification of future data-Class 1: 84.1681
Setting lower bound on correct classification of future data-Class 2: 50
Training set accuracy: 76.2857
Training set accuracy-Class 1: 84.2294
Training set accuracy-Class 2: 45.0704
Test set accuracy: 77.8481
Test set accuracy-Class 1: 82.2695
Test set accuracy-Class 2: 41.1765

Priedas D**MATLAB pagrindinė programa**

```
load poaibis1
data=poaibis1(find(poaibis1(:,36)==0),1:35);
% standartizuojam duomenis %
[eil,st]=size(data);
for i=1:st-1
    vid=mean(data(:,i));
    nuokr=std(data(:,i));
    for j=1:eil
        data(j,i)=(data(j,i)-vid)/nuokr;
```

```

        end
    end
    X=data(find(data(:,35)==1),1:34);
    Y=geriblogi(X);
    training_points = [1:350]';
    test_points = [351:508]';
    data_train = data(training_points,:);
    n_train = size(data_train,1);
    data_test = data(test_points,:);
    Xtrain = data_train(find(data_train(:,35)==1),1:34);
    Ytrain = data_train(find(data_train(:,35)==-1),1:34);
    Xtest = data_test(find(data_test(:,35)==1),1:34);
    Ytest = data_test(find(data_test(:,35)==-1),1:34);
    % tiesinis BMPM %
    xhat = mean(Xtrain)';
    Sx = cov(Xtrain,1);
    yhat = mean(Ytrain)';
    Sy = cov(Ytrain,1);
    beta0 = 0.5;
    gauss_assump = 1;
    [alpha_lin, a, b] =
    build_BMPM_lin_bi_RG(xhat,yhat,Sx,Sy,beta0,gauss_assump,0.1,1e
    -5,5000);
    [ng_tr,pg_tr,ng_t,pg_t,ng_trt,pg_trt,lab,pg_trx,pg_try,pg_tx,p
    g_ty,pg_trtx,pg_trty]=eval_lin_bi([data_train;data_test],n_tra
    in,a,b);
    disp('Underbound on correct classification of future data-
    Class 1:');
    disp(alpha_lin*100);
    disp('Setting lower bound on correct classification of future
    data-Class 2:');
    disp(beta0*100);
    disp('Training set accuracy:');
    disp(pg_tr);
    disp('Training set accuracy-Class 1:');
    disp(pg_trx);
    disp('Training set accuracy-Class 2:');
    disp(pg_try);
    disp('Test set accuracy:');
    disp(pg_t);
    disp('Test set accuracy-Class 1:');
    disp(pg_tx);
    disp('Test set accuracy-Class 2:');
    disp(pg_ty);
    %branduolio metodos%
    K =
    [data_train(:,1:34);data_test(:,1:34)]*[data_train(:,1:34);
    data_test(:,1:34)]';
    labels = [data_train(:,35);data_test(:,35)];
    beta0 = 0.5;
    gauss_assump = 1;
    [alpha_k, gamma_k, b_k] =
    build_BMPM_k_bi_PM(K(1:n_train,1:n_train),labels(1:n_train),be
    ta0,
    gauss_assump,1e-6,1e-5,50);
    [ng_tr,pg_tr,ng_t,pg_t,ng_trt,pg_trt,lab, pg_trx, pg_try,
    pg_tx, pg_ty, pg_trtx, pg_trty] =
    eval_k_bi(K,labels,gamma_k,b_k);
    disp('Underbound on correct classification of future data-
    Class 1:');
    disp(alpha_k*100);
    disp('Setting lower bound on correct classification of future
    data-Class 2:');
    disp(beta0*100);
    disp('Training set accuracy:');
    disp(pg_tr);

```

```
disp('Training set accuracy-Class 1:');  
disp(pg_trx);  
disp('Training set accuracy-Class 2:');  
disp(pg_try);  
disp('Test set accuracy:');  
disp(pg_t);  
disp('Test set accuracy-Class 1:');  
disp(pg_tx);  
disp('Test set accuracy-Class 2:');  
disp(pg_ty);
```