KAUNO TECHNOLOGIJOS UNIVERSITETAS

INFORMATIKOS FAKULTETAS

INFORMACIJOS SISTEMŲ KATEDRA

Tadas Vileiniškis

Intelektualios duomenų gavybos algoritmų taikymo tyrimas Microsoft SQL Server 2005 Data Mining Designer priemonėmis

Magistro darbas

Darbo vadovas: doc. dr. Vigintas Šakys

Kaunas 2009

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS

INFORMACIJOS SISTEMŲ KATEDRA

Intelektualios duomenų gavybos algoritmų taikymo tyrimas Microsoft SQL Server 2005 Data Mining Designer priemonėmis

Magistro darbas

Recenzentas:
doc. dr. Antanas Lenkevičius

2009 01

Darbo vadovas:
doc. dr. Vigintas Šakys

2009 01

Atliko:
IFM-3/4 gr. stud.
Tadas Vileiniškis

2009 01 05

Kaunas 2009

Turinys

Paveikslų turinys

Lentelių turinys

SUMMARY

Nowadays business companies have to face a lot of problems regarding exhaustive data analysis. The main problems occur because there are too many data stored in the databases and manual data analysis is very complex and inefficient. Bicycles selling company has to face these kinds of challenges, because they have a huge amount of detailed information in the database about sales, market and customers. This Master's work is dedicated to analyze company's database and find a solution for a quicker, more efficient and easier way of data analysis.

Data mining is the process of analyzing data from different perspectives and summarizing it into useful information – information that can be used to increase revenue, cuts costs, or both. Data mining software is one of a number of analytical tools for analyzing data. It allows users to analyze data from many different dimensions or angles, categorize it, and summarize the relationships identified. Technically, data mining is the process of finding correlations or patterns among dozens of fields in large relational databases.

The first part of the paper is dedicated to analysis of data mining concepts and business intelligence software. Further on the work discuses company's data base architecture and Microsoft's data mining algorithms. System realization is based on different data mining algorithms. Results provide useful information about customers in the way, which helps to increase sales, predict future sales and extract information about potential buyers. This information enables quicker and more reliable decision making for data analysts and marketing staff.

ĮVADAS

Šiuolaikinės konkurencingos verslo įmonės susiduria su daugybe problemų, bandydamos atliki išsamią turimų duomenų analizę. Problemų kyla ne dėl to, kad nepakanka duomenų, o kaip tik atvirkščiai – duomenų bazėse saugomas vis didesnis kiekis informacijos, ko pasėkoje duomenų analizė rankiniu būdu tampa itin sudėtinga ir neefektyvi. Su šia problema susiduriama ir dviračiais prekiaujančioje įmonėje, kuri yra pardavusi daugybę produkcijos įvairiose pasaulio šalyse bei duomenų bazėje turi sukaupusi daug detalios informacijos apie klientus. Problemos sprendimui išvystytos pažangios technologijos, panaudojančios kompiuterio skaičiuojamąją galią.

Viena iš technologijų yra vadinamoji intelektuali duomenų gavyba (angl. *data mining*). Šios technologijos dėka verslo atstovai gali atsakyti į tokio pobūdžio klausimus kaip: ką dažniausiai perka klientai, kokius produktus parduoti kartu, kokia buvusi rinkos padėtis ir numanoma situacija ateityje ir panašiai, taigi analitikai gali tiksliau bei sparčiau priimti sprendimus ir efektyviau išnaudoti turimus duomenis, kadangi sukurti modeliai gali išrinkti tik potencialiai naudingą informaciją. Dviračiais prekiaujančios įmonės analitikai nori suskirstyti produkciją įsigijusius klientus į grupes, pagal kurias galėtų prognozuoti būsimų klientų poreikius, taipogi siekiama nustatyti kuriem žmonėm siųsti reklaminius pasiūlymus bei norima nuspėti ateities pardavimus.

Duomenų gavybos tikslas – tam tikros naujos informacijos išgavimas iš didelių duomenų saugyklų. Technologijų taikymai versle prasidėjo prieš porą dešimtmečių, ir nuo tų laikų ši technologija išsivystė, pasidarė lankstesnė ir paprastesnė valdyme, bet diegimo, rezultatų interpretavimo ir integravimo sunkumai gerokai stabdo tolimesnę šios technologijos plėtrą organizacijose. Duomenų intelektualios gavybos sprendimai versle dažniausiai naudojami klientų elgesio modelių sukūrimui, prognozuojant ir tikrinant kainodaros strategijų, medicinoje – prognozuojant ligos vystimosi tempus, bei daugybėje kitų sričių, kur reikia apdoroti didelius informacijos kiekius.

Darbo metu realizuoti skirtingi Microsoft intelektualios duomenų gavybos algoritmai, kurių dėka nustatytos panašių klientų grupės, prognozuojami ateities pardavimai ir atrenkama informacija apie potencialiai įdomius klientus. Gautų rezultatų dėka analitikai gali greičiau, paprasčiau ir efektyviau manipuliuoti duomenimis, kas padidina parduotus produkcijos kiekius.

1. TYRIMO SRITIS, OBJEKTAS IR PROBLEMA

Tyrimo sritis – intelektualios duomenų gavybos algoritmų taikymo tyrimas Microsoft „Data Mining Designer“ priemonėmis.

Tyrimo objektas – potencialių klientų paieška ir pardavimų prognozavimas atsižvelgiant į turimus istorinius duomenis.

Tyrimo problema – prekyba užsiimanti įmonė siekia padidinti pardavimų apimtis ir nori išsiaiškinti, kokie klientai labiausiai linkę įsigyti kompanijos produkcijos. Taipogi norima numatyti galimus produkcijos pardavimus ateityje, kad vadybininkai galėtų užsakyti pakankamą kiekį reikalingos produkcijos. Kiekvienas pardavimas yra individualus – perkami skirtingai produktai ir jų priedai, atitinkantys klientų poreikius. Siekiama praplėsti vartotojų ratą: ieškoma naujų klientų ir stengiamasi išlaikyti lojalius klientus. Duomenų bazėje laikomi įrašai apie klientus, kurie praeityje yra įsigiję įmonės produktų, taipogi informacija surinkta apklausų ir viktorinų metu. Atsižvelgiant į istorinius duomenis ir tam tikrus specifinius kriterijus, reikia nuspėti, kokie klientai linkę įsigyti prekes.

2. TYRIMO TIKSLAS IR UŽDAVINIAI

Išanalizuoti duomenų bazėje esantį klientų sąrašą, patikrinti ar nėra klaidingų bei dalinai įvestų duomenų. Siekiama nuspėti galimus pardavimus ateityje, taipogi surinkti informaciją apie potencialiai „įdomius“ klientus, pritaikant intelektualios duomenų gavybos algoritmus.

3. TYRIMO PLANAS

Numatomas tyrimo planas:

- Literatūros analizė ir teorinių žinių gilinimas (informacijos paieška ir klasifikavimas).
- Reikalingos programinės įrangos diegimas (Microsoft SQL Server 2005), konfigūravimas, galimybių tyrimas ir testavimas.
- Duomenų bazės analizė, projektavimas ir realizacija Microsoft SQL Server 2005 priemonėmis.
- Intelektualios duomenų gavybos modelio kūrimas, algoritmų pritaikymas.
- Suprojektuoto modelio efektyvumo testavimas.
- Tyrimo išvadų formulavimas.

4. ANALIZĖS TIKSLAS

Visapusiškai bei nuodugniai išstudijuoti sprendžiamą problemą. Išnagrinėti literatūros šaltinius, susipažinti su dažniausiai kylančiais nesklandumais. Pasidomėti jau esamais sprendimais, jų privalumais ir trūkumais, pritaikant suformuluotos problemos sprendimui.

5. ANALIZĖS METODAI

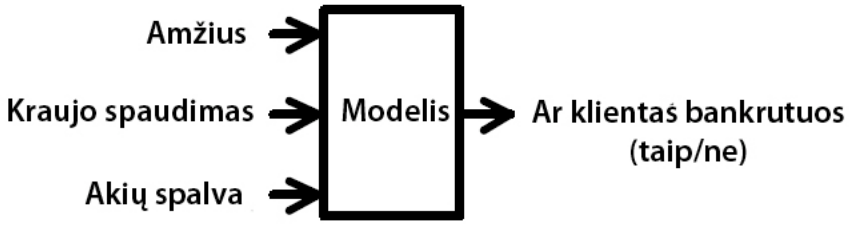
Sprendimo paieškai taikomi mokslinės literatūros analizės bei teorinės analizės ir apibendrinimo metodai.

6. INTELEKTUALI DUOMENŲ GAVYBA

Duomenų intelektuali gavyba yra dažnai apibūdinama, kaip procesas, išrenkant pagrįstą, patikimą ir veiksmingą informaciją iš didelių duomenų bazių. Duomenų intelektuali gavyba nustato neraiškias tendencijas (angl. *patterns*) ir prognozuojamas kryptis (angl. *trends*), kurios egzistuoja duomenyse. Šios tendencijos ir kryptys gali būti surinktos į vieną vietą ir apibrėžtos kaip intelektualios gavybos modelis.

Panašiai, kaip statistika, intelektuali duomenų gavyba nėra tik modeliavimas ir prognozė, bet ištisas problemų sprendimo procesas. Sėkmingam duomenų išgavimui svarbiausia supratimas, ko iš tikrųjų reikia probleminei sričiai, nes to įvertinti negali netgi patys moderniausi ir sudėtingiausi algoritmai. Dar vienas svarbus aspektas – duomenų kokybė, tik iš kokybiškų duomenų galima išgauti kokybiškus duomenis ir kokybiškai atlikti patį duomenų išgavimą. Ši sąlyga sunkiai įvykdoma, nes realūs duomenys retai būna paruošti duomenų gavybai, kadangi jie turi būti integruojami iš skirtingų duomenų šaltinių, turi klaidų arba neteisingų ar trūkstamų reikšmių. [3]

Pagrindinė duomenų gavybos idėja yra ta, kad reikalingų duomenų modelių ar taisyklių radimui galima panaudoti kompiuterį. Duomenų išgavimo technika ir algoritmai priklauso nuo pačių duomenų, jų kilmės, struktūros, užduoties ir kitų veiksnių (žr. 1 pav.).



1 pav. Duomenų intelektualios gavybos taikymo pavyzdys

6.1. Intelektualios duomenų gavybos panaudojimo galimybės

Duomenų intelektualios gavybos metodologija gali būti taikoma ten, kur sprendžiamos duomenų klasifikacijos ir ryšių tarp duomenų bei informacinių modelių identifikavimo problemos.

Duomenų intelektualiosios gavybos technologijų įgyvendinimo galimybės:

- Statistiniams skaičiavimams, sudarant hipotezes, ieškant tam tikrų modelių su netolygiais kintamaisiais.
- Optimizuojant sprendimus laiko ir sistemos išteklių aspektais.
- Komandų ir kontrolės įgyvendinime, siekiant sumažinti žmogiško faktoriaus klaidas.

Dažniausiai duomenų intelektualiosios gavybos technologijos naudojamos šiose pramonės srityse:

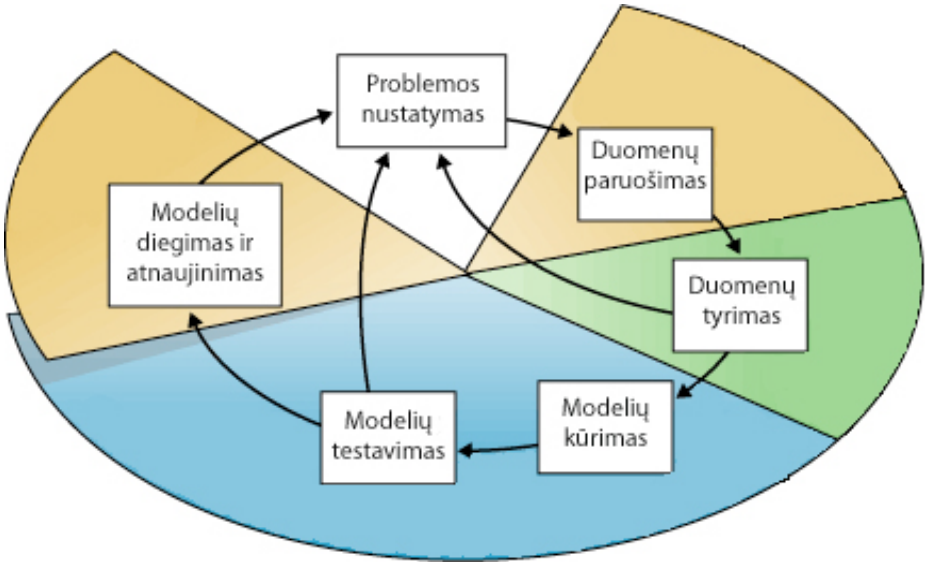
- Klientų vadybos fazėse: naujų klientų paieška, pajamų gavimas iš esamų klientų, lojalių klientų išsaugojimas. Tikslinių rinkų specifikavimo analizė pagal tam tikras charakteristikas.
- Medicininiai taikymai. Intelektualios duomenų gavybos metodologijos naudojamos nustatyti operacijų ir procedūrų efektyvumui, medicininių testavimų srityje. Farmacijoje, cheminės ir genetinės informacijos apdorojimui. Vaistų gamyboje, jų veiksmingumo numatyme, tam tikrų ligų gydymo procese. Cheminėje inžinerijoje.
- Mažmenininkų veikla. Nusprendžiant kokį produktą siūlyti rinkai. Numatant tam tikrų reklaminių kampanijų efektyvumą. Padeda verslo veiklos analizei, ieškant duomenų ryšių ir metodų, kurių teisingumas turi būti patvirtintas realaus pasaulio reiškiniais.
- Verslas ir finansai. Duomenų gavyba yra viena iš labiausiai besivystančių dirbtinio intelekto sričių, be to, plačiai naudojama stambių įmonių. Ši sfera leidžia analizuoti vartotojų praeities elgesį ir pagal tai atlikti strateginius ateities sprendimus. Bene plačiausiai duomenų gavyba naudojama ryšių su klientais valdymui. [2]

6.2. Intelektualios duomenų gavybos procesas

Intelektualios gavybos modelio kūrimas yra didesnio proceso dalis, kuriame yra viskas nuo pagrindinės problemos iki modelio dislokavimo į aplinką. Skiriami šeši pagrindiniai duomenų intelektualios gavybos proceso žingsniai:

1. Problemos nustatymas.
2. Duomenų paruošimas.
3. Duomenų tyrimas.
4. Modelių kūrimas.

5. Modelių testavimas.
6. Modelių dislokavimas ir atnaujinimas.



2 pav. Duomenų intelektualios gavybos proceso žingsniai

Nors duomenų intelektualios gavybos procesas atvaizduotas kaip ciklinis (žr. 2 pav.), bet kiekvienas žingsnis nebūtinai veda prie kito žingsnio. Taigi duomenų intelektualios gavybos modelio kūrimas yra dinaminis ir interaktyvus procesas.

Atlikus duomenų analizę, galima pastebėti, jog jie nėra pakankami norint sukurti tinkamus gavybos modelius, todėl reikia ieškoti daugiau duomenų. Gali nutikti ir taip, jog sukūrus keletą modelių, bus pastebėta, jog jie nepadeda išspręsti problemos. Taigi duomenų intelektualios gavybos modelio kūrimas yra procesas, kurio kiekvienas žingsnis gali būti kartojamas tol, kol pavyks sukurti tinkamą modelį.

Kompanijos Microsoft produktas SQL Server 2005 turi integruotą aplinką, pritaikytą intelektualios gavybos modelių kūrimui ir darbui su jais tai – verslo intelektualios plėtros studija (angl. *Business Intelligence Development Studio*). Į aplinką įeina duomenų intelektualios gavybos algoritmai ir priemonės, kurios palengvina parengti visapusiškus ir išsamius sprendimus daugeliui skirtingų projektų. [1]

6.2.1. Problemos nustatymas

Pirmasis duomenų intelektualios gavybos proceso žingsnis – aiškus verslo problemos nustatymas. Šiame žingsnyje reikia analizuoti verslo poreikius, apibrėžti problemos sritį, nustatyti matus, kuriais modelis bus įvertintas bei apibrėžti galutinius projekto tikslus. Problemos nustatymui turėtų būti keliami tokie klausimai:

- Ko tikimasi ir ieškoma?
- Kokius duomenų rinkinio atributus mėginama prognozuoti?

- Kokio tipo ryšius norima surasti?
- Ar norima padaryti prognozes iš intelektualios gavybos modelio, ar tik ieškoma dominančių neaiškių tendencijų ir asociacijų?
- Kokiu būdu paskirstyti duomenys?
- Kaip tarpusavy susiję stulpeliai arba kelių lentelių atveju lentelės?

Norinti atsakyti į šiuos klausimus, reikėtų atlikti duomenų savybių analizę, ištirti verslo vartotojų poreikius atsižvelgiant į galimus duomenis. Jei duomenys neatitinka vartotojų poreikių, gali prireikti iš naujo apibrėžti projekto problemą.

6.2.2. Duomenų paruošimas

Antrasis žingsnis duomenų intelektualios gavybos procese – pirmojo problemos žingsnio metu nustatytų duomenų paruošimas. Šio žingsnio darbams įvykdyti naudojamas Microsoft SQL Server 2005 Integration Services (SSIS) paketas, turintis visas reikiamas priemones.

Duomenys gali būti išsibarstę po visą kompaniją ir saugomi skirtingais formatais, arba gali turėti nesuderinamumų, tokių kaip neteisingi ar trūkstami pradiniai duomenys (pvz. vartotojas pirko produktą, kai dar nebuvo gimęs ir panašiai). Prieš pradedant modelio kūrimą reikia užfiksuoti tokias problemas. Dirbant su labai dideliu duomenų rinkiniu, fiziškai neįmanoma peržiūrėti kiekvienos transakcijos. Todėl duomenų tyrimams ir nesuderinamumų radimui reikia naudoti tam tikras automatizavimo formas, esančias Integration Services pakete.

6.2.3. Duomenų tyrimas

Trečiasis žingsnis duomenų intelektualios gavybos procese – parengtų duomenų tyrimas. Reikia suprasti duomenis, kad būtų galima priimti atitinkamus sprendimus kuriant modelius. Tyrimo technikos apima minimalių ir maksimalių reikšmių, vidurkių, standartinių nukrypimų apskaičiavimus bei duomenų pasiskirstymo priklausomybių paiešką. Atlikus duomenų tyrimą, galima spręsti, ar duomenų rinkinyje yra duomenų su defektais bei pasirinkti strategiją problemų nustatymui. Duomenų tyrimui atlikti naudojama Data Source View Designer programa, turinti keletą reikiamų priemonių.

6.2.4. Modelio kūrimas

Ketvirtasis žingsnis duomenų intelektualios gavybos procese – intelektualios gavybos modelių kūrimas. Prieš pradedant kurti modelį, reikia atsitiktinai atskirti parengtus duomenis į atskirus mokymo ir testavimo duomenų rinkinius. Modelio kūrimui naudojamas mokymo duomenų rinkinys, modelio tikslumo testavimui – testavimo rinkinys.

Intelektualios gavybos modelio nustatymui ir kūrimui naudojamos žinios, gautos iš duomenų tyrimo žingsnio. Tipiškai modelis turi įėjimo stulpelius, identifikavimo stulpelį ir

prognozuojamą stulpelį. Po to galima nustatyti šiuos stulpelius naujame modelyje naudojant Data Mining Extensions (DMX) programavimo kalbą arba Data Mining vedlį iš BI Development Studio paketo.

Nustačius intelektualios gavybos modelio struktūrą, jis kuriamas paskelbus tuščią struktūrą su neraiškiomis tendencijomis, kaip aprašyta modelyje. Šis procesas vadinamas modelio apmokymu. Neraiškios tendencijos randamos praleidžiant originalius duomenis per matematinį algoritmą. SQL Server 2005 turi skirtingą algoritmą kiekvieno tipo modeliui, kurį galima sukurti. Galima naudoti parametrus, kiekvieno algoritmo suderinimui. Intelektualios gavybos modelis yra apibrėžiamas kaip duomenų intelektualios gavybos struktūros objektas ir duomenų intelektualios gavybos algoritmas.

6.2.5. Modelio testavimas

Penktasis žingsnis duomenų intelektualios gavybos procese – sukurto modelio tyrimas ir efektyvumo testavimas. Nepatartina dislokuoti modelį į gamybos aplinką, neatlikus jo testavimo. Taipogi rekomenduotina sukurti keletą modelių ir nuspręsti, kuris modelis tinkamiausias. Jei nė vienas sukurtas modelis neveikia gerai, reikia grįžti į ankstesnį proceso žingsnį, ir iš naujo nustatyti problemą arba pakartotinai atrinkti duomenis į originalų duomenų rinkinį.

Galima testuoti, kaip gerai modelis kuria prognozes naudojant dizainerio priemones, pavyzdžiui, pakėlimo diagramas (angl. *lift chart*) ir klasifikavimo matricas (angl. *classification matrix*). Šios priemonės reikalauja testavimo duomenų, kurie buvo atskirti nuo originalaus duomenų rinkinio, modelio kūrimo žingsnyje.

6.2.6. Modelio dislokavimas

Paskutinis šeštasis žingsnis duomenų intelektualios gavybos procese – geriausiai veikiančių sukurtų modelių dislokavimas į gamybos aplinką. Kai duomenų intelektualios gavybos modeliai jau egzistuoja gamybos aplinkoje, galima atlikti daug užduočių, priklausomai nuo poreikių. Keletas užduočių pavyzdžių, kurias galima atliktis su duomenų intelektualios gavybos modeliais:

- Modelių naudojimas kuriant prognozes, kurias galima naudoti priimant verslo sprendimus. SQL Server turi DMX programavimo kalbą, kurią galima naudoti kuriant prognozės užklausas. Prediction Query Builder programa padeda sudaryti šias užklausas.
- Integration Services pagalba, sukurkite paketą, kuriame intelektualios gavybos modelis yra naudojamas intelektualiai atskirti įeinančius duomenis į daug lentelių. Pavyzdžiui, jei duomenų bazė pastoviai atnaujinama su potencialiais vartotojais, galima naudoti intelektualios gavybos modelį kartu su Integration Services programa dalinant

ateinančius duomenis vartotojams, kurie galėtų būti potencialūs klientai ir vartotojus, kurie tikriausiai nepirks produkto.

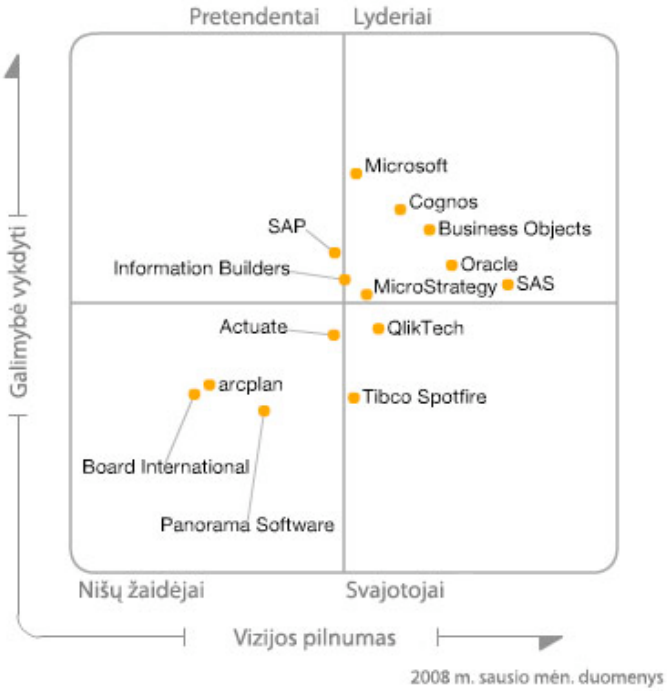
Modelio atnaujinimas yra dislokavimo strategijos dalis. Atsiradus daugiau duomenų, reikia iš naujo perkurti modelius, taip pagerinant jų efektyvumą.

7. VERSLO ĮŽVALGOS PLATFORMOS

Intelektualios duomenų gavybos uždavinių sprendimui sukurta įvairių gamintojų produktų, nuo mažų programinės įrangos paketų, iki daugybe funkcijų pasižyminčių verslo įžvalgos platformų.

Verslo įžvalgos (angl. *Business Inteligence*) platformų magiškasis kvadrantas (žr. 3 pav.) atspindi globalią „Gartner“ [4] nuomonę apie pagrindines kompanijas, teikiančias verslo įžvalgos programinius sprendimus. Rezultatai ir komentarai paremti šių trijų šaltinių duomenimis:

- klientų, naudojančių atitinkamos kompanijos verslo įžvalgos produktus nuomone, pareikšta „Gartner“ apklausų metu;
- gamintojų internete atliktos apklausos rezultatais;
- gamintojų atsakymais į klausimyną apie jų produkto verslo įžvalgos strategiją ir operacijas.



3 pav. Verslo įžvalgos platformų magiškasis kvadrantas

Didieji gamintojai pradeda dominuoti verslo įžvalgos įrankių rinkoje – anksčiau nei už vienų metų Microsoft, Oracle, SAP ir IBM užims daugiau nei du trečdalius rinkos.
Trečiame paveiksle esantys gamintojai suskirstyti į atitinkamas kategorijas:

Lyderiai:

Kategorijai priskiriami gamintojai, kurių produktai pasižymi dideliu funkcionalumu ir panaudojimo galimybėmis. Sprendimai gali būti nesudėtingai integruojami organizacijos aplinkoje, suteikdami skirtingų verslo įžvalgos strategijų palaikymą.

Pretendentai:

Gamintojų produktai pasižymi vidutiniu funkcionalumu ir yra konkurencingi. Galimi apribojimai, susiję su specifinėmis techninėmis aplinkomis ar programų domenais. Įvairių verslo įžvalgos platformų dalyse gali pasireikšti koordinacijos strategijos trūkumas. Trūksta geografinių aspektų ir specifinio, pramonei pritaikyto turinio, kurį siūlo lyderių kvadrante esantys gamintojai.

Svajotojai:

Kategorijai priskiriami gamintojai, turintys stiprią verslo įžvalgos platformos pristatymo viziją. Sprendimai išsiskiria programų architektūros atvirumu, lankstumu, ir siūlo gerą funkcionalumą pasirinktose srityse, tačiau turi spragų naudojant platesnėse srityse. Pasitaiko plėtojimo galimybių problemų.

Nišų žaidėjai:

Kategorijai priskiriami gamintojai, kurie pasižymi geru funkcionalumu specifiniuose verslo įžvalgos rinkos segmentuose (pavyzdžiui ataskaitų ruošime), arba turintys ribotas galimybes kurti inovatyvius sprendimus bei konkuruoti su kitais gamintojais.

1 lentelė. Verslo įžvalgos platformų palyginimas

Gamintojas	Privalumai	Trūkumai
Microsoft	• Sprendimo kaina ir integracija su gamintojo Office paketu bei SQL serverio produktais. • Didelis vartotojų ratas ir pagalbinės informacijos kiekis. • Programuotojams suteikiama infrastruktūra, kūrimo įrankiai, darbo sekos ir bendradarbiavimo galimybės, kurios yra pranašesnės, nei konkurentų teikiami analogai.	• Atsilikimas metaduomenų valdymo, ataskaitų rengimo ir „ad hoc“ užklausų galimybių srityse. • Pakankamai sudėtinga integracija organizacijų aplinkose, dirbančiose su kito gamintojo sprendimais.

	• Produktą naudojančių klientų manymu, Microsoft siūlo geriausią verslo įžvalgos sprendimą lyderių tarpe. Daugiau nei pusė respondentų teigė, kad programinė įranga veikė be klaidų. • Didžioji dalis verslo įžvalgos technologijų sukurta paties gamintojo.	
Oracle	• Verslo įžvalgos platformos ir analitinių programų kombinacija yra vienas iš geriausių sprendimų rinkoje. • Potencialas teikti operacines ir strategines verslo įžvalgos galimybes įvairiose aplinkose. • Sekos ir bendradarbiavimo galimybės, bei rafinuota vizualizacija. • Galingas OLAP varklis ir Hyperion Microsoft Office integracijos galimybės.	• Reikalingas geresnis verslo įžvalgos programų servisas ir palaikymas. • Nepakankama techninė ekspertizė. • Didelė sprendimo kaina.
SAS	• Dominavimas rinkoje pažangių analitinių sprendimų srityje. • Platus galimybių spektras ir pažangios analitinės funkcijos. • Stiprus servisas veikiantis daugumoje pasaulio šalių. • Integracija su JMP suteikia stiprias vizualizacijos galimybes.	• Sunkiai naudojamo produkto reputacija. • Dauguma duomenų manipuliavimo ir pažangios analizės užduočių reikalauja SAS programavimo kalbos. • Nepasižymi tradicinių ataskaitų sprendimais. • Trūksta internetinių ataskaitų, inkrementinių kubų atnaujinimo.
SAP	• Rimta ekspertizė ir gerai išplėtota infrastruktūra. • Didžiausias verslo įžvalgos platformų gamintojas (susijungta su kompanija „Business Objects“).	• Mažiau funkcionalesnė ir sunkiau organizacijos aplinkoje diegiama platforma, nei kitų gamintojų sprendimai. • Trūksta galimybės paprasčiau integruoti platformą kitose aplinkose, kuriose dirba ne SAP platforma pagrįstos programos. • Prasčiausias bendras vartotojų įvertinimo balas atsižvelgiant į funkcijas bei integravimo patirtį.
Cognos	• Platus vartotojų ratas. • Stiprus servisas, stabiliai veikianti sistema. • Duomenų integracija ir nestruktūrinės/teksto analizės galimybės.	• Prastai adaptuota analizės studija. • Tvirtesnės MOLAP pasiūlos trūkumas. • Menkas populiarumas tarp vartotojų naudojančių platformą OLAP stiliaus analizei didelėse reliacinėse duomenų bazėse. • Nuspėjamosios analizės ir

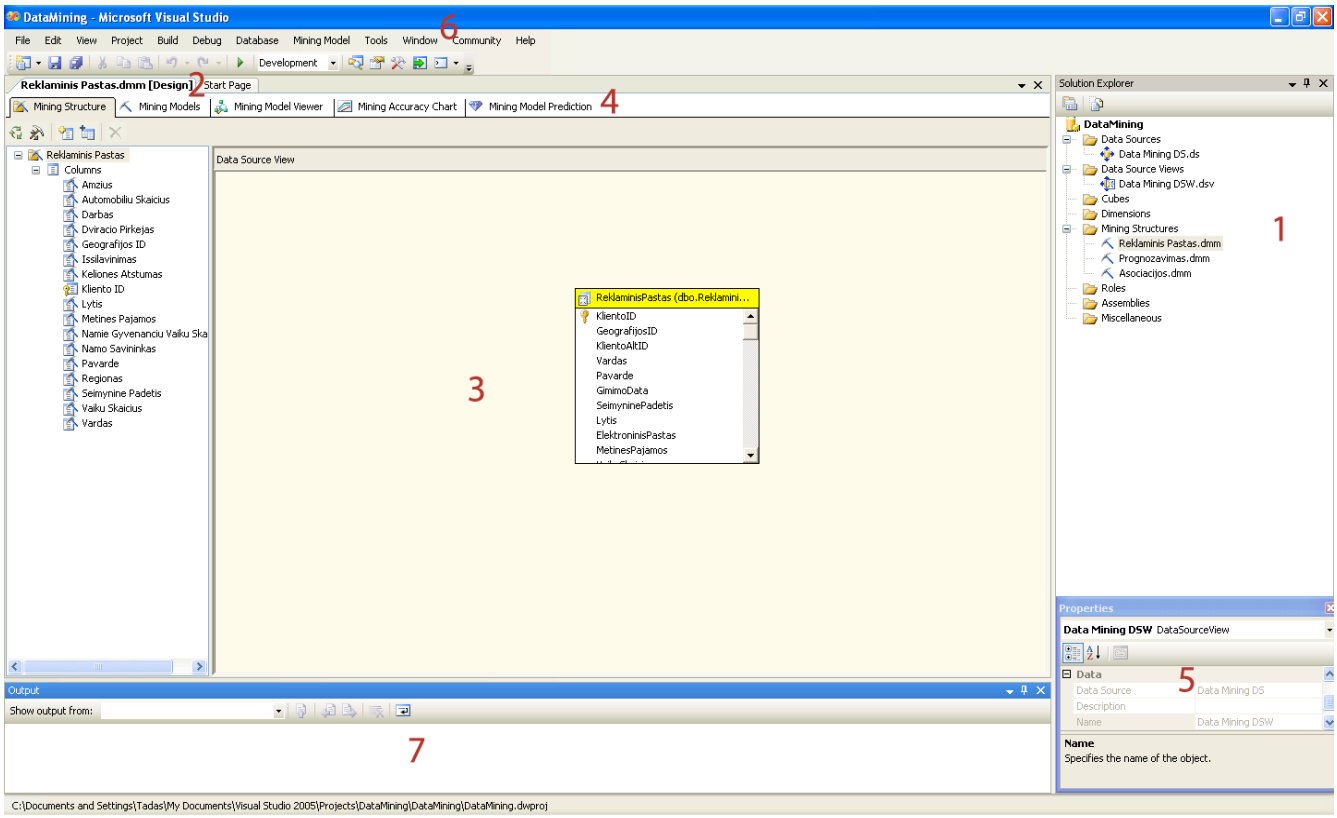
		duomenų gavybos galimybės gerokai silpnesnės, nei kitų pirmaujančių gamintojų. • Trūksta lankstesnių ataskaitų funkcionalumo, tokio kaip galimybės sukurti daugybinius informacijos blokus daugiamatėms dimensijoms.

7.1. Pasirinkta verslo įžvalgos platforma

Kompanijos „Microsoft“ teikiama „SQL Server Business Inteligence Development Sturio“ verslo įžvalgos platforma (žr. 4 pav.) turi dideles intelektualios duomenų gavybos technologijų galimybes, kuriomis lenkia kitų gamintojų siūlomus sprendimus. Internete ir vietinėje platformos pagalbos bibliotekoje pateikta aiški dokumentacija, nestinga pagalbinių priemonių. Rinkoje esantys nemokami (angl. *open-source*) sprendimai yra prastai dokumentuoti ir turi neištaisytų klaidų bei suderinamumo problemų. Kompanijos Microsoft platforma informatikos fakulteto studentams suteikiama nemokamai, kas suteikė labai didelį privalumą renkantis iš galimų sprendimų.

7.2. Microsoft SQL serverio verslo įžvalgos platformos aprašas

Didžioji dalis SQL serverio intelektualios duomenų gavybos užduočių atliekama verslo įžvalgos kūrimo studijoje (angl. *Business Intelligence Development Studio*). Ši aplinka integruota į Microsoft Visual Studio aplinką, kad užtikrintų pilną verslo įžvalgos operacijų palaikymą.



4 pav. Verslo įžvalgos kūrimo studija

7.3. Svarbiausios verslo įžvalgos kūrimo studijos platformos dalys

1. Sprendimo naršymo langas:

   Vieta, kurioje valdomi sprendimai ir projektai. Visi objektai kuriami ir redaguojami šiame lange. Norint projekte pridėti naują objektą pasirenkamas tinkantis elementas, ir dešiniu pelės mygtuko spragtelėjimu nurodoma funkcija “New”.

2. Langų meniu:

   Leidžia greitai išsirinkti norimą dizainerio langą. Kiekvienam atidarytam projektui sukuriamas naujas poskyris. Jeigu atidaryta daugiau objektų, negu gali atvaizduoti langų meniu, dešinėje pusėje atsiranda žemyn iškrentantis netelpančių projektų sąrašas.

3. Dizainerio langas:

   Vieta, kurioje analizuojami ir redaguojami objektai. Sukūrus naują ar sprendimo naršymo lange pasirinkus esantį objektą, atveriamas specifinis dizaineris (skiriasi priklausomai nuo objekto), kurio dėka manipuliuojama objektu.

4. Dizainerio meniu:

   Daugelis objektų turi skirtingus aspektus, kuriuos galima redaguoti ar peržiūrėti. Šiuos aspektus nurodo dizainerio meniu atsirandantys atitinkami punktai.

5. Nustatymų langas:

   Kontekstui jautrus parinkčių langas, atvaizduojantis pasirinkto elemento nustatymus. Tai bendras „Visual Studio“ konceptas, pasireiškiantis visose atliekamose operacijose. Pasirinkus elementą neturintį nustatymų, atvaizduojamas tuščias langas.

6. Verslo įžvalgos meniu:

   Pagrindiniame meniu tarp “Debug” ir “Tools” sričių esantys meniu laukai yra specifinio analizės serviso objekto meniu, kuris kinta priklausomai nuo pasirinkimo.

7. Informacijos išvesties langas:

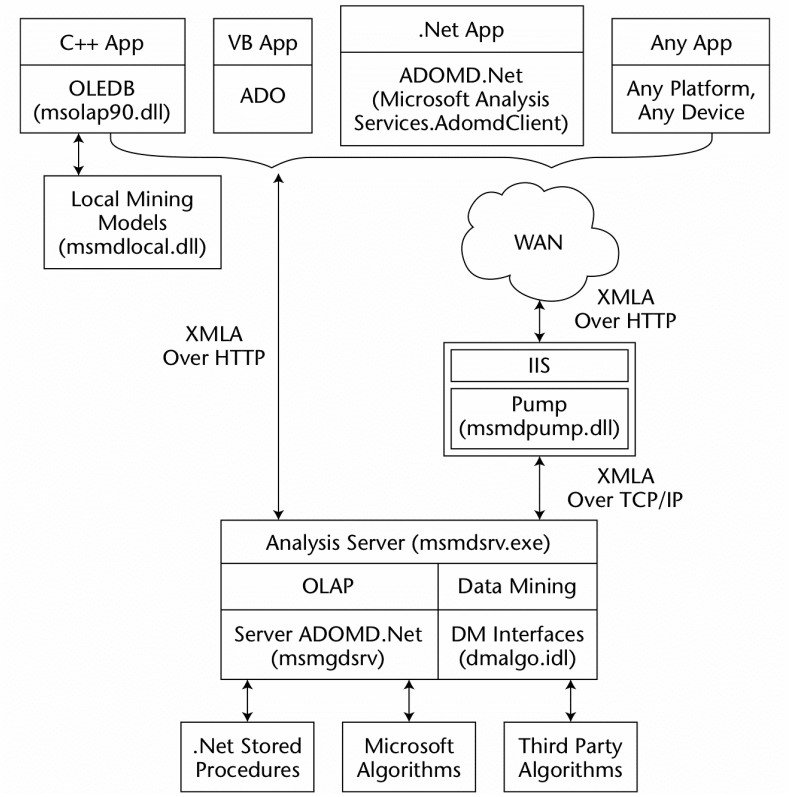
   Atvaizduojami pranešimai kompiliuojant ir paleidžiant projektą. Jeigu paleidimo metu aptinkama klaidų, jos atvaizduojamos šiame lange.

7.4. Analizės servisų architektūra

Analizės servisų veikimas pagrįstas paprasta kliento/serverio architektūra, leidžiančia klientams prisijungti TCP/IP arba HTML protokolais naudojantis ISS. Analizės servisai suteikia daugybę skirtingų klientų, tokių kaip OLE DB, ADOMD.Net, ir daugiau. Kiekvienas klientas komunikuoja su serveriu naudodamas pagrindinę analizės servisų sąsają, kuri vadinama XML analizei (angl. XMLfor Analysis).

Tradicinės duomenų pasiekimo technologijos reikalauja kliento komponentų, kurie yra tvirtai suporuoti su duomenų tiekėju diegimo. Šis poravimas sukuria apribojimus – priklausymas nuo kalbos ar platformos, ir skirtingų kliento ir serverio komponentų versijų suderinamumo problemos. XMLA sukurtas atvirų standartų HTTP, XML ir SOAP pagrindu, taigi nėra pririštas prie jokios specifinės kalbos ar platformos. XMLA apibrėžia komunikacijos mechanizmą su analitinių duomenų tiekėjais internetu. Technologinio lankstumo dėka, XMLA yra pagrindinis visų klientų ir analizės servisų serverių komunikacijos protokolas. Nepriklausomai nuo prisijungimui prie serverio naudojamo API, didžioji dalis užklausų ir atsakymų gaunami XMLA dėka.

Serveriui gavus užklausą analizės servisai nustato iš kurio mechanizmo ji (OLAP ar intelektualios duomenų gavybos) atsiųsta, ir atitinkamai ją nukreipia. Intelektualios duomenų gavybos užklausos aktyvuoja duomenų gavybos algoritmą, kuris gali būti sukurtas Microsoft (pridedamas su produktu), arba įdiegtas serveryje iš trečiųjų šalių. Papildomai gali būti inicijuojama vartotojo aprašyta užsaugota procedūra, naudojanti serverio pusės ADOMD.Net sąsaja, kad tiesiogiai pasiektų serverio objektus ir modelio turinį. Intelektualios duomenų gavybos kliento architektūros diagrama pateikta 5 paveiksle.



5 pav. Duomenų gavybos architektūra

7.5. Analizės servisų veikimo principas

Apdorojimas vyksta trimis lygiais:

1. Duomenų šaltiniui siunčiamos užklausos
2. Apibrėžiama neapmokyta statistika
3. Naudojantis modelio apibrėžimu ir intelektualios duomenų gavybos algoritmais apmokomi gavybos modeliai.

Norėdami apdoroti gavybos struktūrą ir ją sudarančius modelius, analizės servisai inicijuoja keletą užklausų šaltinio duomenų bazėje. Varikliukui užklausinėjant duomenų, sukuriamas

visų diskrečių ir diskretizuotų stulpelių sąrašas. Papildomai siunčiama užklausa siekiant nustatyti, ar yra tolydžias reikšmes turinčių stulpelių.
Visų užklausų tikslas – apdoroti specializuotą OLAP kubą, esantį gavybos struktūros viduje (vartotojui kubas nėra prieinamas). Asocijuotų dimensijų dėka kubas spartinančiojoje atmintyje indeksuoja ir talpina atvejo duomenis. Apdorojęs kubą serveris sukuria nepriklausomas gijas, kurios apmoko visus struktūroje esančius modelius. [15]

8. PROBLEMINĖS SRITIES APRAŠAS

Perspektyvi įmonė užsiima dviračių ir jų dalių prekyba, taipogi teikia papildomus montavimo ir remonto darbus. Įmonė turi savo komponentų gamybos liniją, tačiau dalis komponentų užsakoma iš partnerių, turinčių didesnę patirtį atitinkamojoje srityje. Priimami individualūs klientų užsakymai bei juridinių asmenų užsakymai. Atliekami produkcijos pritaikymo pagal klientų poreikius darbui bei, esant pageidavimui suteikiamos įvairios papildomos paslaugos (konsultuojama kaip prižiūrėti ir tvarkyti produkciją, teikiami įvairūs nenumatyti specifikacijose priedai, ir atliekamas senos produkcijos remontas bei patikra).

8.1. Pagrindinės veiklos valdymo funkcijos

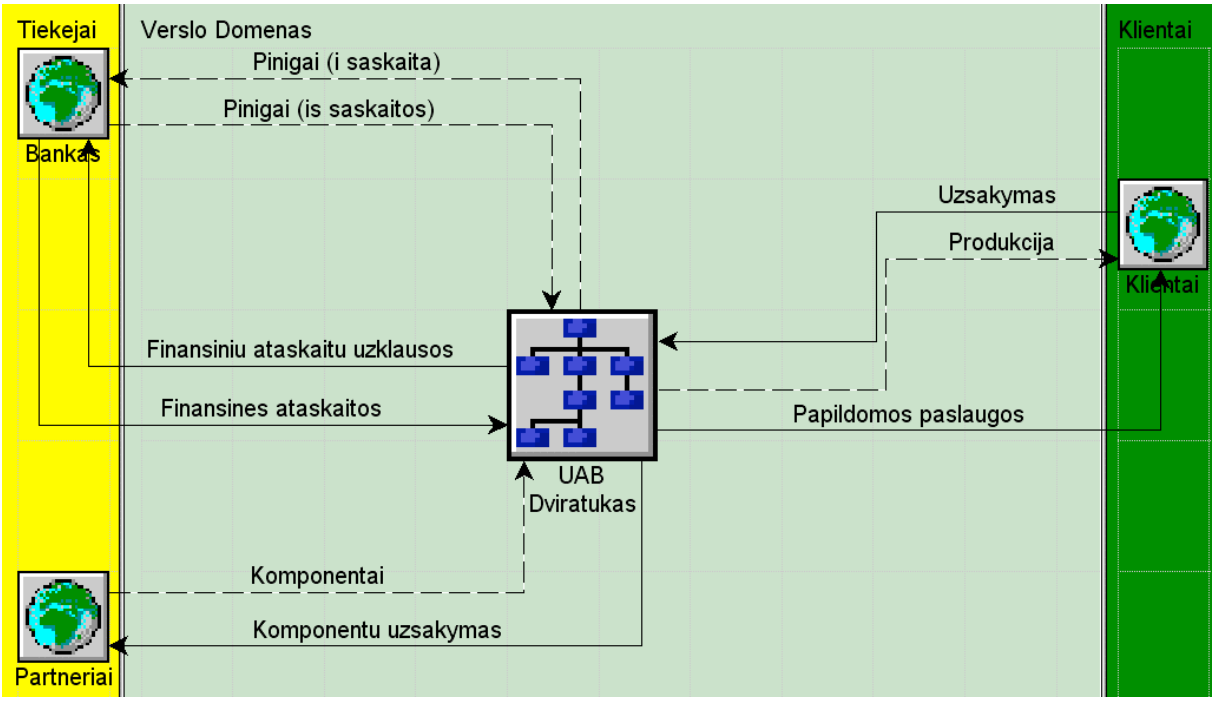
- *Kliento užsakymo priėmimas (pardavimas).*
  Visi klientų užsakymai saugomi duomenų bazėje (kliento kontaktinė informacija, šiek tiek informacijos apie kliento gyvenimą, užsakytos produkcijos aprašymai, produktų parametrai ir t.t.). Kiekvienam klientui priskiriamas unikalus identifikacijos numeris (ID), pagal kurį koduojamas užsakymas. Formuojamos ataskaitos bei nurodymai, kurie vėliau išsiunčiami gamybos skyriaus darbuotojams.
- *Produkto gamyba.*
  Gamybos skyriaus darbuotojai sulaukę informacijos apie užsakymą patikrina ar turimi visi reikalingi komponentai. Jeigu produktui surinkti trūksta komponentų, vykdomas komponentų užsakymas iš partnerių. Iš atskirų dalių surenkamas vartotojo poreikius atitinkantis produktas. Sumontuoti produktai siunčiami pristatymų poskyrio darbuotojams. Buhalterijai siunčiamos komponentų pirkimo ir produkto gamybos ataskaitos.
- *Produkto pristatymas.*
  Pristatymų poskyrio darbuotojai, gavę kliento kontaktinius duomenis, pristato užsakytą produktą į namus ir atlieka derinimo (bei papildomai užsakytus darbus).

Sumontavę produktą darbuotojai užpildo atitinkamą formą apie sėkmingai atliktą užsakymą (pardavimo ataskaitą) ir laukia naujų užsakymų.

8.2. Verslo sąveikos modeliai

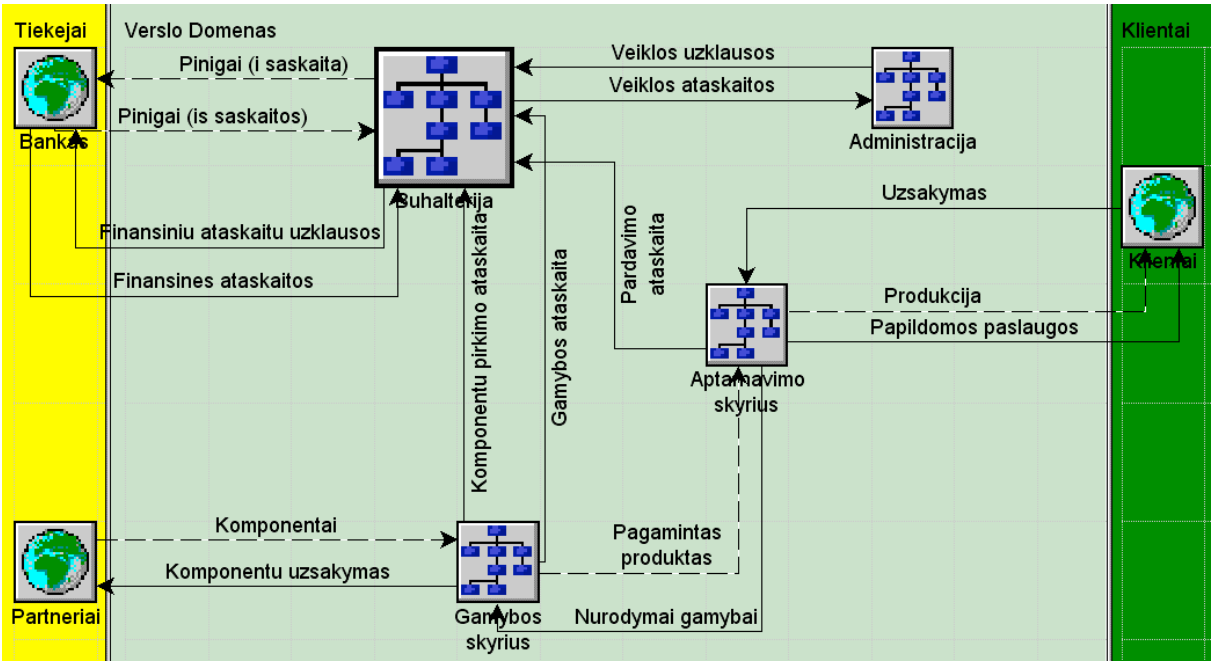
Šiuose modeliuose atvaizduojama organizacijos UAB „Dviratukas“ veiklos strategija ir sąveika su klientais bei tiekėjais. Modeliai suteikia bendrą organizacijos veiklos vaizdą.

6 paveiksle pateiktame modelyje organizacija yra tarytum „juoda dėžė“, neskaidant jos į atitinkamus skyrius ar padalinius, taipogi nesimato ryšių tarp vidinių organizacijos elementų. Matomi tik organizacijos sąveikos su išore srautai. Punktyrinė linija žymi materialius, ištisinė – informacinius srautus.



6 pav. Aukščiausio lygio verslo sąveikos modelis

7 paveiksle jau atvaizduojama organizacijos vidinė struktūrai ir tarpusavio srautai. Tai detalus įmonės veiklos strategijos atvaizdas.



7 pav. Nulinio lygio verslo sąveikos modelis

9. SIEKIAMAS SPRENDIMAS

Sukurti intelektualiosios duomenų gavybos modelį, pritaikytą atitinkama produkcija prekiaujančiai įmonei. Užtikrinti tinkamą modelio veikimą ir reikiamą funkcionalumą bei garantuoti modelio atnaujinimą, pasikeitus rinkos rodikliams ar verslo logikai. Sukurtas modelis palengvins analitikų darbą, padės jiems greičiau ir efektyviau priimti reikiamus sprendimus.

10. SISTEMOS REIKALAVIMŲ SPECIFIKACIJA

10.1. Nefunkciniai reikalavimai

Reikalingas kompiuteris, kuriame patalpinta duomenų bazė su informacija apie pardavimus, klientus ir produkciją. Jeigu tokios duomenų bazės nėra, jinai suprojektuojama ir realizuojama. Visi darbai atliekami „Microsoft SQL Server 2005 Developer Editon“ programinės įrangos pagalba. Šio paketo viduje esančio įrankio „Data Mining Designer“ pagalba sukuriami ir testuojami intelektualios duomenų gavybos modeliai.

Kompiuteris (32 bitų) turi atitikti žemiau pateiktus reikalavimus:

- 600 MHz Pentium III arba greitesnis procesorius; rekomenduojamas 1 (GHz) arba greitesnis procesorius.

- Microsoft Windows 2000 Serveris su Service Pack (SP) 4 ar naujesniu; Windows Server 2003 Standard Edition, Enterprise Editon arba Datacenter Editon su SP 1 ar naujesniu; Windows Small Business Server 2003 su SP 1 bei naujesniu.
- 512 megabaitų (MB) arba daugiau operatyviosios atmintinės (RAM); rekomenduojamas 1 gigabaitas (GB) arba daugiau.
- Apie 350 MB laisvos vietos standžiajame diske rekomenduojamam diegimo rėžimui. Apie 425 MB papildomos laisvos vietos standžiajame diske SQL Server internetinės knygoms, SQL Server Mobile internetinėms knygoms ir pavyzdinėms duomenų bazėms.
- Reporting Services funkcionalumui reikalinga Microsoft Internet Information Services (IIS) 5.0 arba naujesnė bei ASP.NET 2.0 arba naujesnė. [14]

11. MICROSOFT INTELEKTUALIOS GAVYBOS ALGORITMAI

11.1. Sprendimo medžių algoritmas

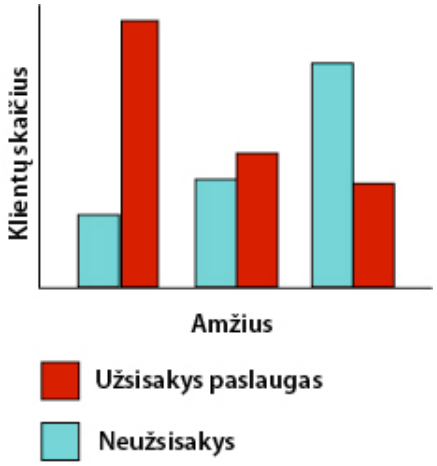
Microsoft sprendimo medžių algoritmas yra klasifikacijos ir regresijos algoritmas, naudojamas prognozavimų modeliavime su diskrečiais ir tolydžiais atributais. Diskretiems atributams algoritmas daro prognozes, remdamasis ryšiais tarp duomenų rinkinio įėjimo duomenų stulpelių. Jis naudoja stulpelių reikšmes ar būsenas nustatant prognozuojamų stulpelių būsenas. Specifiškai algoritmas identifikuoja įėjimo stulpelius, kurie yra susiję su spėjamu stulpeliu. Sprendimų medis padaro prognozes, paremtas jų tendencijomis per dalinius išėjimus. Tolydiems atributams algoritmas naudoja tiesinę regresiją nustatyti, kur sprendimų medis šakojasi. Jei daugiau kaip vienas stulpelis yra prognozuojamas, arba jei įėjimo duomenys turi įdėtinę lentelę prognozuojamame rinkinyje, algoritmas rengia atskirą sprendimų medį kiekvienam prognozuojamam stulpeliui. [7]

Šis algoritmas realizuoja intelektualios duomenų gavybos modelį, sukurdamas eilę medžio atsišakojimų, dar vadinamų mazgais Algoritmas prideda naują mazgą į modelį kiekvieną kartą, kai aptinkamas įvesties stulpelis, kuris glaudžiai siejasi su nuspėjamu stulpeliu. Būdai, kuriais algoritmas nulemia išsišakojimus skiriasi, priklausomai nuo to, ar bandomas nuspėti tolydinis, ar diskretus stulpelis. Sprendimų medžio modelis privalo turėti raktinį stulpelį, įvesties stulpelius ir vieną bandomą nuspėti stulpelį.

11.1.1. Diskrečių atributų prognozės

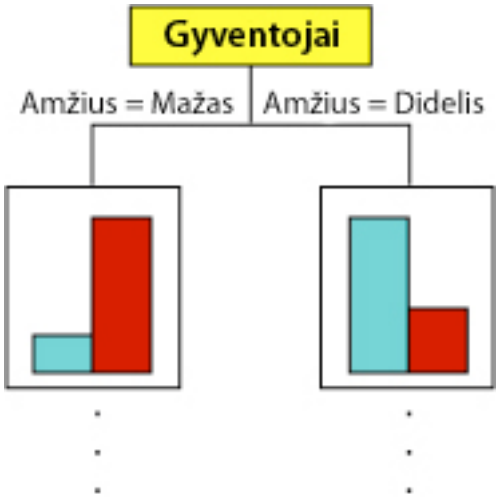
Būdas kuriuo Microsoft sprendimo medžių algoritmas sukuria medį, diskrečių duomenų atveju, gali būti iliustruojamas histogramos panaudojimu. Žemiau esančiame

paveikslėlyje pavaizduota histograma susiejanti bandomą nuspėti stulpelį „Paslaugas užsisakę klientai“ su įvesties stulpeliu „Amžius“. Iš histogramos matoma, kad asmens amžius padeda nustatyti, ar asmuo užsisakys paslaugas, ar ne.



8 pav. Paslaugų pardavimo spėjimo histograma

Histogramoje atvaizduotos sąsajos atveju, sprendimų medžio algoritmas sukurtų naują mazgą modelyje.

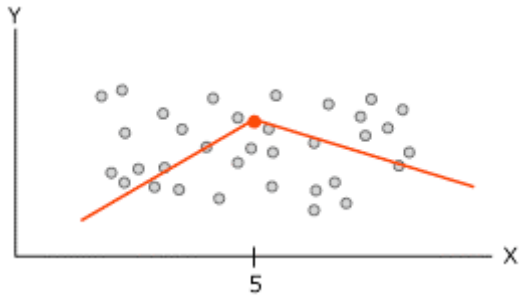


9 pav. Paslaugų pardavimo spėjimo modelis

Algoritmui pridėjus naujų mazgų į modelį, sukuriama medžio struktūra. Viršutinis medžio mazgas apibrėžia bandomo nuspėti stulpelio santykį su visa klientų populiacija. Modeliui augant, algoritmas atsižvelgia į visus stulpelius.

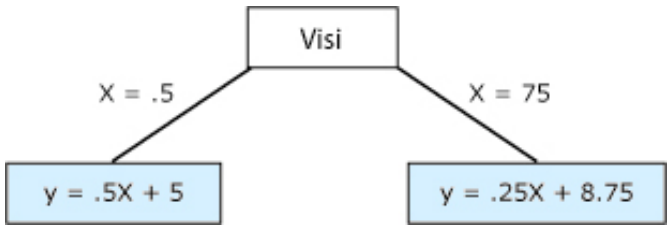
11.1.2. Tolydžių atributų prognozės

Tolydžių duomenų atveju, sprendimų medžio algoritmas sukuria medį, pagrįstą nenutrūkstamų bandomu nuspėti stulpeliu, tokiu atveju kiekvienas mazgas turi regresijos formuluotę. Atsišakojimas įvyksta nelinijiniame regresijos formuluotės taške.



10 pav. Tolydžių atributų prognozė

Diagramoje atvaizduoti duomenys, gali būti modeliuojami tiek panaudojus pavienes linijas, tiek dvi sujungtas linijas. Tačiau panaudojus pavienę liniją, duomenys būtų atvaizduojami ganėtinai prastai. Dviejų linijų atveju, modelis būtų labiau tinkamas duomenų aproksimacijai. Dviejų linijų susikirtimo taškas ir yra nelinijinis, šiame taške sprendimų medžio modelyje išsišakoja mazgai.



11 pav. Nelinijinis taškas ir išsišakojantys mazgai

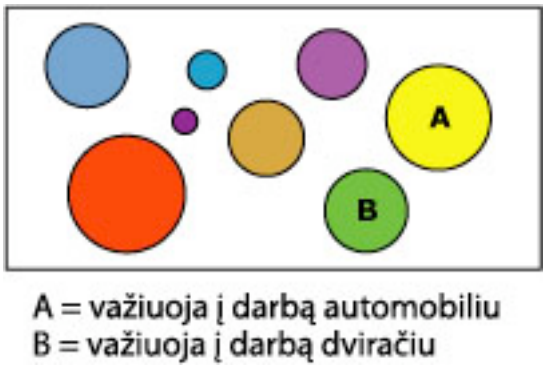
Dvi lygtys atvaizduoja dviejų linijų regresijos formuluotes.

11.2. Klasterių algoritmas

Tai yra grupavimo algoritmas, kurį suteikia Microsoft SQL Server 2005 Analysis Services (SSAS). Algoritmas naudoja iteratyvias metodikas, kad sugrupuotų duomenų rinkinius į klasterius, turinčius panašias charakteristikas. Šie grupavimai yra naudingi duomenų peržiūrai, duomenyse esančių anomalijų identifikacijai ir prognozių kūrimui.

Klasterių modeliai apibrėžia duomenų rinkinio tarpusavio ryšius, kuriuos nustatyti rankiniu būdu yra sudėtinga. Logiškai galima nustatyti, kad žmonės, važinėjantys į darbą dviračiais, dažniausiai gyvena netoli darbovietės. Algoritmas gali aptikti kitokias šių žmonių

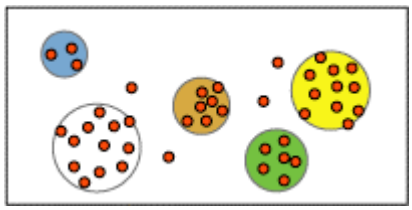
charakteristikas, kurios nėra aiškiai numatomos. 12 paveiksle klasteris A aprašo duomenis apie žmones važiuojančius į darbą automobiliais, o klasteris B duomenis apie žmones, važiuojančius į darbovietes dviračiu.



12 pav. Duomenų rinkinio grupės

Klasterių algoritmas skiriasi nuo kitų duomenų gavybos algoritmų: norint sukuri klasterių modelį, nereikia apibrėžti prognozuojamo stulpelio. Algoritmas griežtai apmoko modelį pagal duomenyse egzistuojančius ryšius ir pagal algoritmo atpažįstamus klasterius.

Algoritmas visų pirma identifikuoja duomenų rinkinių tarpusavio ryšius ir sugeneruoja eilę klasterių, atsižvelgdamas į šiuos ryšius. Vizualiam algoritmo duomenų grupavimo atvaizdavimui tinka išbarstytų taškų erdvė, pavaizduota 13 paveiksle. Išbarstytų taškų erdvė atvaizduoja visus duomenų rinkinio atvejus, kurių kiekvienas yra diagramos taškas. Klasteriai sugrupuoja diagramos taškus ir atvaizduoja algoritmo atpažįstamus ryšius.



13 pav. Klasterio algoritmo veikimo atvaizdavimas

Nustatęs klasterius, algoritmas apskaičiuoja atspindėtų taškų grupių efektyvumą ir bando iš naujo apibrėžti grupes, kad sukurtų geresnius klasterius. Proceso metu algoritmas kartojamas tol, kol nebegali pagerinti rezultatų. [6]

Sudarytų grupių efektyvumo skaičiavimams atlikti algoritmas naudoja du metodus: Galimybių Maksimizavimo (angl. *Expectation Maximization*) ir K-Reikšmių (angl. *K-Means*). Galimybių maksimizavimo atveju algoritmas naudoja tikimybinį metodą, kad suskaičiuotų tikimybę, ar duomenų taškas egzistuoja klasteryje. K-Reikšmių atveju algoritmas naudoja atstumo rodiklį, kad priskirtų duomenų tašką artimiausiam klasteriui.

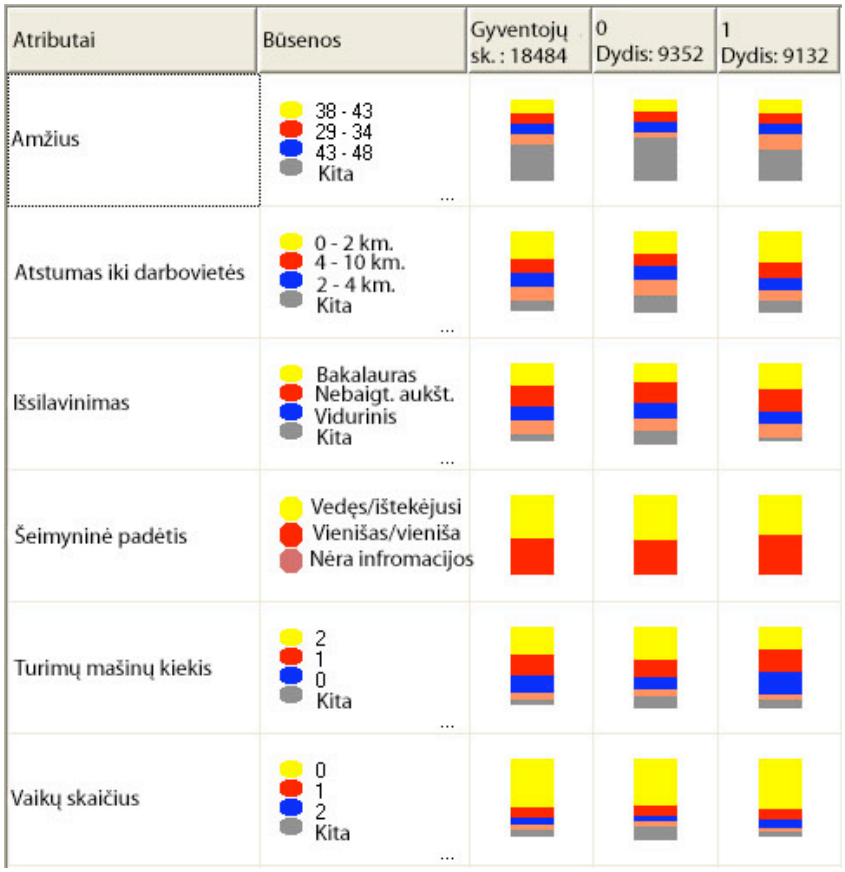
Klasterio modelis privalo turėti rakto stulpelį ir įvesties stulpelius. Galima apibrėžti įvesties stulpelius kaip nuspėjamus. Algoritmas leidžia gavybos modelių kūrimui naudoti Prognozuojančią Modelio Žymų kalbą (angl. *Predictive Model Markup language*).

11.3. Paprasto Atskyrimo algoritmas

Tai yra klasifikacijos algoritmas, kurį suteikia SSAS nuspėjamų modelių kūrimui. Algoritmas apskaičiuoja sąlyginę tikimybę tarp įvesties ir nuspėjamų stulpelių, ir daro prielaidą, kad šie stulpeliai yra nepriklausomi. Neieškoma galimų sąryšių tarp duomenų.

Paprasto atskyrimo algoritmas neatlieka tiek daug skaičiuojamojo darbo, kaip kiti Microsoft algoritmai, taigi jis greitai sugeneruoja intelektualios gavybos modelius, kad būtų galima atrasti ryšius tarp įvesties ir nuspėjamų stulpelių. Algoritmas naudotinas norint atlikti pradinę duomenų peržiūrą, o po to, pritaikius gautus rezultatus, sukurti papildomus gavybos modelius, pasitelkus kitus galingesnius bei tikslesnius algoritmus.

Algoritmas apskaičiuoja kiekvieno įvesties stulpelio visų būsenų tikimybę. Business Intelligence Development Studio įrankio pagalba galima sugeneruoti vizualų algoritmo apskaičiuotų būsenų paskirstymą.



14 pav. Klasterio algoritmo veikimo atvaizdas

Paskirstyme atvaizduojamas kiekvienas duomenų rinkinio įvesties stulpelis, matomas kiekvieno stulpelio būsenų pasiskirstymas. Vaizdinys gali būti panaudotas svarbių ir kintančių atitinkamose būsenose įvesties stulpelių nustatymui. 14 paveiksle esančiame stulpelyje „Atstumas iki darbovietės“ tikimybė, kad klientas nusipirks dviratį yra 0.387, jeigu atstumas nuo jo gyvenamosios vietos iki darbo yra nuo 2 iki 4 km., tuo tarpu tikimybė, kad jis nepirks dviračio yra 0.287. Algoritmas naudojasi skaitmenine informacija, gauta iš kliento charakteristikų , kad numatytų, ar jis gali įsigyti produkcijos. [10]

Paprasto atskyrimo modelis privalo turėti rakto stulpelį, įvesties stulpelius ir vieną prognozuojamą stulpelį. Visi stulpeliai turi būti diskretūs. Algoritmas gavybos modelių kūrimui nepalaiko Prognozuojančios Modelio Žymų kalbos.

11.4. Asociacijų algoritmas

Algoritmas naudingas rekomendacijų varikliukams, kurie rekomenduoja produktus klientams, priklausomai nuo to, kokius produktus jie jau yra nusipirkę, arba kuriais buvo susidomėję. Asociacijų algoritmas taipogi tinkamas rinkos krepšelio analizei.

Asociacijos modeliai kuriami duomenų rinkiniams, turintiems identifikatorius tiek atskiriem atvejams, tiek ir tuos atvejus sudarantiems elementams. Elementų grupė vadinama elementų rinkiniu. Asociacijos modelis yra sudarytas iš kelių elementų rinkinių ir taisyklių, apibrėžiančių kaip šie elementai grupuojami tarpusavyje. Taisyklės, kurias identifikuoja algoritmas, gali būti panaudotos nuspėjant galimus kliento pirkimus, remiantis elementais, esančiais kliento pirkimo krepšelyje.

2 lentelė. Elementų rinkinio taisyklės

Taisyklė
Kelioninio Buteliuko Dėklas = Yra, Dviratininko Šalmas = Yra -> Vandens Buteliukas = Yra
Mountain-200 = Yra, Kalnų Padanga = Yra -> HL Kalnų Padanga = Yra
Mountain-200 = Yra, Vandens Buteliukas = Yra -> Kalnų Buteliuko Dėklas = Yra
Touring-1000 = Yra, Vandens Buteliukas = Yra -> Kelioninio Buteliuko Dėklas = Yra
Road-750 = Yra, Vandens Buteliukas = Yra -> Kelioninio Buteliuko Dėklas = Yra
Turistinė Padanga = Yra, Sport-100 = Yra -> Turistinės Padangos Kamera = Yra

Asociacijų algoritmas potencialiai gali surasti daug taisyklių viename duomenų rinkinyje. Elementų rinkinių ir sukurtų taisyklių apibrėžimui naudojami du parametrai – palaikymas (angl. *support*) ir tikimybė (angl. *probability*). Tarkime, kad X ir Y apibūdina du prekių krepšelyje galinčius būti elementus. Palaikymo parametras yra duomenų rinkinio,

turinčio elementų X ir Y kombinacijas atvejų skaičius. Naudodamasis palaikymo parametro ir vartotojo apibrėžtų parametrų *MINIMUM_SUPPORT* ir *MAXIMUM_SUPPORT* kombinacijomis, algoritmas valdo kuriamų elementų rinkinių kiekį. Tikimybės parametras, (dar vadinamas tikrumo parametru) atspindi dalį duomenų rinkinio atvejų, turinčių elementus X, kurie turi elementus Y. Naudodamas tikimybės ir *MINIMUM_PROBABILITY* parametrų kombinaciją algoritmas valdo sukuriamų taisyklių skaičių.

Asociacijos algoritmas išanalizuoja duomenų rinkinį, kad surastų atskiruose atvejuose kartu esančius elementus. Visi susiję elementai sugrupuojami į rinkinius, pasirodančius parametre *MINIMUM_SUPPORT* apibrėžtame atvejų skaičiuje. Elementų rinkinys galėtų būti toks: "Mountain 200=Yra, Sport 100=Yra" ir galėtų turėti 710 palaikymą. Taisyklės sugeneruojamos iš elementų rinkinių. Jos yra naudojamos prognozuojant elemento egzistavimą duomenų bazėje, atsižvelgiant į kitų svarbių specifinių elementų egzistavimą. Taisyklė galėtų būti: jeigu Touring 1000=yra ir Kelioninio Buteliuko Dėklas=yra, tada Vandens Buteliukas=yra, ir galėtų turėti 0,812 tikimybę. Algoritmas identifikuoja, kad krepšelyje esant Touring 1000 produktui ir buteliuko dėklui, vandens buteliukas greičiausiai bus pirkinių krepšelyje. [5]

Asociacijos modelis privalo turėti rakto stulpelį, įvesties stulpelius ir vieną prognozuojamą stulpelį. Įvesties stulpeliai turi būti diskretūs. Dažniausiai asociacijos modelio įvesties duomenys būna dviejose lentelėse. Vienoje lentelėje gali būti kliento informacija, o kitoje lentelėje kliento pirkimų informacija. Šiuos duomenis galima įkelti į modelį, naudojant susietą (angl. *nested*) lentelę.

Algoritmas gavybos modelių kūrimui nepalaiko Prognozuojančios Modelio Žymų kalbos.

11.5. Sekų klasterių algoritmas

Tai yra sekų analizės algoritmas, kurį suteikia SSAS. Algoritmo pagalba galima analizuoti duomenis, turinčius keliais arba sekomis sujungiamus įvykius. Surandamos dažniausiai pasitaikančios sekos bei tarpusavyje sugrupuojamos identiškos sekos. Jos gali būti įvairių formų, įskaitant ir:

- duomenys, aprašantys vartotojo aplankomų tinklalapio dalių kelius.
- duomenys, aprašantys eiliškumą, kuriuo vartotojai įkelia prekes į pirkinių vežimėlį internetinėje parduotuvėje.

Algoritmas yra panašus į klasterių Algoritmą, tačiau, užuot ieškojęs atvejų klasterių, turinčių panašius atributus, surandami atvejų klasteriai, turintys panašias sekas.

Sukuriamas gavybos modelis turi dažniausiai duomenyse pasitaikančių sekų aprašymus, kuriuos galima panaudoti, prognozuojant kitą naujos sekos žingsnį. Algoritmas gali informuoti apie tiesiogiai su sekomis nesusijusius duomenų stulpelius, kuriuos galima panaudoti identifikuojant ryšius tarp sekose esančių duomenų bei jose nepasireiškiančių duomenų.

Klasterių ir jų sekų identifikavimui naudojamas Tikimybių Maksimizavimo (angl. *Expectation Maximization*) metodas, kad būtų nustatyta tikimybė, ar duomenų taškai egzistuoja klasteryje.

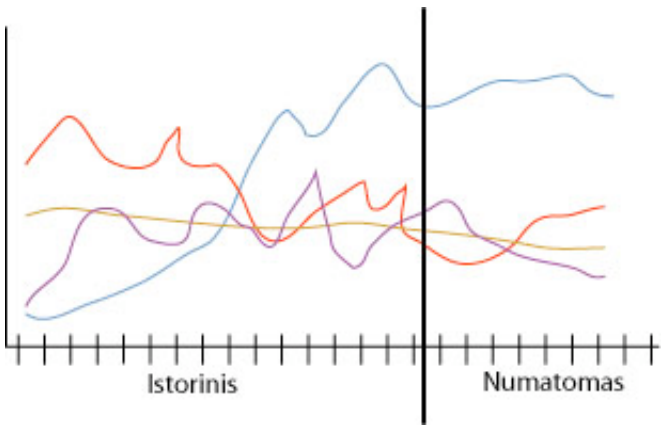
Vienas iš naudojamų įvesties stulpelių yra susieta lentelė, turinti sekų duomenis. Tai duomenų rinkinio individualaus atvejo perėjimų būsenų eilė. Nustatant kurie sekos stulpeliai bus naudojami kaip klasterių įvesties stulpeliai, matuojami skirtumai arba atstumai tarp visų galimų sekų, esančių duomenų rinkinyje. Nustatęs šiuos atstumus algoritmas gali naudotis sekos stulpeliu kaip Tikimybių Maksimizavimo metodo įvestimi. [12]

Klasterių modelyje privalo būti raktas, identifikuojantis įrašus, ir susieta lentelė, turinti susietus stulpelius, aprašančius sekos įvykius. Kiekvienai sekai galimas tiktai vienas susietas stulpelis. Kiekviename modelyje galimas tiktai vienas sekų tipas.

Algoritmas gavybos modelių kūrimui nepalaiko Prognozuojančios Modelio Žymų kalbos.

11.6. Laiko sekų algoritmas

Tai regresijos algoritmas naudojamas kuriant intelektualios duomenų gavybos modelius, siekiant nuspėti tolydžius stulpelius. Laiko sekų modelio prognozavimas pagrįstas tiktai tendencijomis, gautomis iš pradinio duomenų rinkinio, modelio kūrimo metu.



15 pav. Tipinis pardavimų prognozavimo modelis laiko atžvilgiu

Modelis sudarytas iš dviejų dalių: istorinės informacijos ir numatomos informacijos. Istorinė informacija yra toji, kurią algoritmas naudoja modelio kūrimui, tuo tarpu numatoma informacija atvaizduoja modelio sugeneruotas prognozes. Ties istorinės ir numatomos informacijos riba nubrėžta linija vadinama eilute (angl. *series*). Kiekvienas prognozių modelis privalo turėti atvejų eilutę, atskiriančią taškus eilutėje. Data yra šio atvejo eilutė, kadangi duomenys parodo istorinį ir numatomą prekių pardavimą kelių mėnesiu laikotarpyje.

Algoritmas turi galimybę atlikti kryžminį spėjimą. Apmokius algoritmą dviem skirtingomis, tačiau susijusiomis eilutėmis, galima naudoti gautą modelį, bandant nuspėti vienos eilutės elgesį atsižvelgiant į kitos eilutės elgesį. Stebimo objekto pardavimai gali įtakoti numatomus kito produkto pardavimus.

Algoritmas apmoko modelį naudodamasis Auto Regresyviu sprendimų medžiu (angl. *Auto Regressive decision tree*). Kiekvienas modelis turi raktinį stulpelį, apibrėžiantį laiko pjūvius, kuriuos turėtų nustatyti modelis. Algoritmas susieja įvairų kiekį praeities elementų su kiekvienu numatomu elementu. Įvesties duomenys apibrėžiami dviem būdais.

3 lentelė. Įvesties atvejų aprašymo pirmasis būdas

LaikoID	Prekė	Pardavimai	Kiekis
1/2001	A	1000	600
2/2001	A	1100	500
1/2001	B	500	900
2/2001	B	300	890

Laiko ID stulpelis yra identifikatorius, sudarytas iš dviejų įrašų kiekvienai dienai. Prekės stulpelis apibrėžia duomenų bazėje esantį produktą. Pardavimų stulpelis aprašo bendrą aprašyto produkto pelną vienos dienos laikotarpy, o kiekio stulpelyje talpinama informacija apie atitinkamo produkto likusį kiekį sandėlyje. Šiuo atveju modelis turi du nuspėjamus stulpelius: Pardavimai ir Kiekis.

4 lentelė. Įvesties atvejų aprašymo antrasis būdas

LaikoID	A_Pardavimai	A_Kiekis	B_Pardavimai	B_Kiekis
1/2001	1000	600	500	900
2/2001	1100	500	300	890

4 lentelėje pardavimų ir kiekio stulpeliai išskaidyti į du stulpelius, kurių kiekvienas išdėstomas pagal produkto pavadinimą. To pasėkoje LaikoID stulpelyje egzistuoja tiktai vienas kiekvienos dienos įrašas. Modelis turi keturis nuspėjamus stulpelius: A_Pardavimai, A_Kiekis, B_Pardavimai ir B_Kiekis.

Abu įvesties duomenų aprašymo metodai sugeneruoja tą pačią modelyje atvaizduojamą informaciją, tačiau įvesties atvejų formatas kinta, priklausomai nuo intelektualios duomenų gavybos modelio aprašymo.

Laiko sekų algoritmui reikalinga, kad numatomi stulpeliai būtų tolydūs. Kiekvienam modeliui galima tik viena atvejų eilutė. [13]

Algoritmas gavybos modelių kūrimui nepalaiko Prognozuojančios Modelio Žymų kalbos.

11.7. Neuroninių tinklų algoritmas

Neuroninių tinklų algoritmas sukuria klasifikacijos ir regresijos gavybos modelius, sukonstruodamas daugiasluoksnį neuronų tinklą. Apskaičiuojama kiekvieno įvesties atributo galimos būsenos tikimybė. Šios tikimybės vėliau gali būti panaudotos numatant atributų rezultatą, atsižvelgiant į įvesties atributus.

Neuroninių tinklų algoritmas naudingas analizuojant sudėtingus įvesties duomenis, kai turimas labai didelis kiekis apmokomų duomenų, bet taisyklės negali būti lengvai išvedamos, panaudojus kitus algoritmus.

Neuroninių Tinklų algoritmo naudojimo atvejai:

- marketingo ir reklamos analizės pasisekimo nustatymas.
- akcijų vertės nustatymas
- valiutos kurso svyravimų nustatymas
- nestabilios finansinės informacijos nustatymas, įvertinus istorinius duomenis
- pramonės ir gamybos procesų analizė.

Algoritmas naudoja daugiasluoksnį tinklą, sudarytą iš trijų neuronų arba perceptronų (angl. *perceptron*) sluoksnių. Šie sluoksniai yra: įvesties sluoksnis, nebūtinas paslėptas sluoksnis ir išvesties sluoksnis. Kiekvienas neuronas gauna vieną arba daugiau įvesčių ir pagamina vieną arba daugiau identiškų išvesčių. Kiekviena išvestis yra paprasta neurono įvesčių sumos netiesinė funkcija. Įvestys tiesiog pereina iš mazgų įvesties sluoksnyje, į mazgus, esančius paslėptame sluoksnyje, o galiausiai patenka į išvesties sluoksnį; tarp neuronų, esančių tame pačiame sluoksnyje nėra ryšių.

Neuroninių tinklų algoritmo pagalba sukurtas intelektualios duomenų gavybos modelis gali turėti daugybinius tinklus, priklausomai nuo įvesties ir spėjimams naudojamų stulpelių skaičiaus. Vieno gavybos modelio turimas tinklų skaičius priklauso nuo įvesties stulpeliuose ir nuspėjamuose stulpeliuose aprašytų būsenų skaičiaus.

Neuroniniame tinkle išskiriami trys neuronų tipai:

- Įvesties neuronai:

  duomenų gavybos modeliui suteikia įvesties atributo vertes. Diskrečių įvesties atributų atveju, neuronas dažniausiai aprašo vieną įvesties atributo būseną, įskaitant ir trūkstamas vertes. Dvejetainis (angl. *binary*) įvesties atributas sukuria vieną įvesties mazgą, aprašantį esamą arba trūkstamą būseną. Loginis (angl. *boolean*) įvesties atributas sukuria tris įvesties neuronus: vienas neuronas *true* reikšmei, vienas neuronas *false* reikšmei, ir vienas neuronas trūkstamai arba egzistuojančiai būsenai. Diskretus įvesties atributas, turintis daugiau nei dvi būsenas, sukuria vieną įvesties neuroną kiekvienai būsenai, ir vieną įvesties neuroną trūkstamai arba egzistuojančiai būsenai. Tolydus įvesties atributas sukuria du įvesties neuronus: vienas neuronas trūkstamai arba egzistuojančiai būsenai, kitas tolydžiojo atributo vertei. Įvesties neuronai suteikia įvestis vienam arba daugiau paslėptų neuronų.
- Paslėpti neuronai:

  gauna įvestis iš įvesties neuronų ir suteikia informaciją išvesties neuronams.
- Išvesties neuronai

  duomenų gavybos modelyje atspindi nuspėjamų atributų reikšmes. Diskretiems įvesties atributams, neuronas dažniausiai pateikia vieną nuspėjamą būseną, įskaitant ir trūkstamas reikšmes. Dvejetainis nuspėjamas atributas sukuria vieną išvesties mazgą, aprašantį trūkstamą arba egzistuojančią būseną. Loginis nuspėjamas atributas sukuria tris išvesties neuronus: vieną neuroną *true* reikšmei, vieną neuroną *false* reikšmei ir vieną neuroną trūkstamai arba egzistuojančiai būsenai. Diskretus nuspėjamas atributas, turintis daugiau nei dvi būsenas, sugeneruoja po vieną išvesties neuroną kiekvienai būsenai, ir vieną išvesties neuroną trūkstamai arba egzistuojančiai būsenai. Tolydūs nuspėjami stulpeliai sugeneruoja du išvesties neuronus: vieną neuroną trūkstamai arba egzistuojančiai būsenai ir vieną neuroną tolydžiojo stulpelio reikšmei. Jeigu peržiūrint nuspėjamų stulpelių rinkinius sugeneruojama daugiau nei 500 išvesties neuronų, Analysis Services sukuria naują tinklą, kad atvaizduotų papildomus išvesties neuronus.

Neuronas gauna kelias įvestis: įvesties neuronai gauna įvestis iš pradinių duomenų; paslėpti neuronai ir išvesties neuronai gauna įvestis iš kitų tinkle esančių neuronų išvesčių. Įvestys sukuria ryšius tarp neuronų, kurie tarnauja kaip keliai, specifinių įvykių atvejų analizėje.

Kiekviena įvestis turi jai priskirtą reikšmę, taip vadinamą svorį, aprašantį konkrečios įvesties svarbumą paslėptam ar išvesties neuronui. Kuo didesnis svoris priskirtas įvesčiai, tuo jis svarbesnis gaunančiam neuronui. Svoris gali būti ir neigiamas, kas reiškia, kad įvestis gali neleisti specifinio neurono.

Atitinkamai kiekvienas neuronas turi jam priskirtą paprastą netiesinę funkciją, vadinamą aktyvacijos funkcija. Ji apibrėžia konkretaus neurono svarbumą neuronų tinklo sluoksniui. Paslėpti neuronai naudoja hypertangento funkciją, tuo tarpu išvesties neuronai naudoja sigmoidinę funkciją. [11]

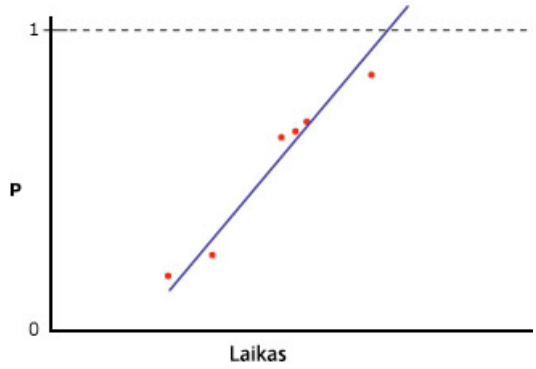
Neuroninių tinklų modelis privalo turėti rakto stulpelį, vieną arba daugiau įvesties stulpelių, ir vieną arba daugiau nuspėjamų stulpelių.

Neuroninių tinklų algoritmo pagalba sukurti modeliai nepalaiko „*drillthrough*“ bei duomenų gavybos dimensijų, kadangi mazgų struktūra nebūtinai tiesiogiai atitinka pamatinius duomenis.

11.8. Logistikos regresijos algoritmas

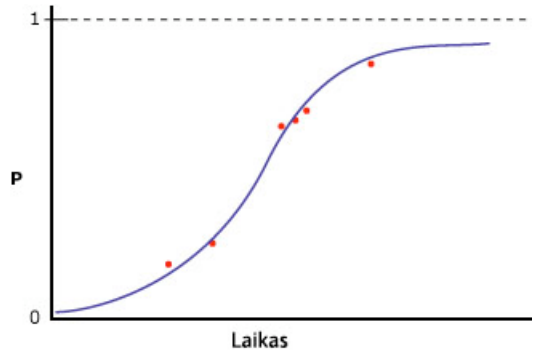
Tai neuroninių tinklų algoritmo variacija, kurioje parametro *HIDDEN_NODE_RATIO* reikšmė lygi nuliui. Šis nustatymas sukurs neuronų tinklo modelį, neturintį paslėpto sluoksnio, o tai yra ekvivalentu logistikos regresijai.

Nors nuspėjamas stulpelis turi tiktai dvi būsenas, vis tiek norima atlikti regresijos analizę, susiejančią įvesties stulpelius su tikimybe, kad nuspėjamas stulpelis turės specifinę būseną. 16 paveiksle pateikti rezultatai, kurie būtų gauti, priskyrus nuspėjamo stulpelio būsenoms reikšmes 0 ir 1, paskaičiavus tikimybę, kad stulpelis turi specifinę būseną ir atlikus linijinę įvesties kintamojo regresiją.



16 pav. Regresijos metu gaunami rezultatai

X ašis – įvesties stulpelio reikšmės, Y ašis – tikimybė, kad nuspėjamas stulpelis bus vienoje arba kitoje būsenoje. Linijinė regresija nesuformuoja situacijos, kad stulpelio reikšmė būtų tarp 0 ir 1, nors tai maksimali ir minimali stulpelio reikšmė. Šios problemos sprendimui reikia atlikti logistikos regresiją. Jos metu analizė sukuria "S" formos kreivę, turinčią maksimumo ir minimumo apribojimus.



17 pav. Logistikos regresijos metu gaunami rezultatai

Atkreipkite dėmesį, kad kreivė niekada neperžengia 1 ribos ir nenukrenta iki 0 ribos. Logistikos regresiją galima naudoti apibrėžiant kurie įvesties stulpeliai yra svarbūs, nustatant nuspėjamo stulpelio būseną.

Logistikos regresijos modelis privalo turėti rakto stulpelį, vieną arba daugiau įvesties stulpelių ir vieną arba daugiau nuspėjamų stulpelių. [9]

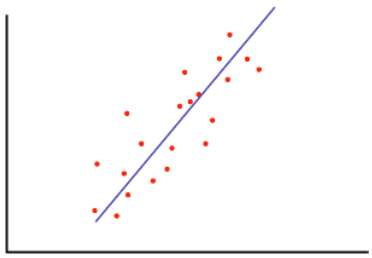
Logistikos regresijos algoritmo pagalba sukurti modeliai nepalaiko „*drillthrough*“ bei duomenų gavybos dimensijų, kadangi mazgų struktūra nebūtinai tiesiogiai atitinka pamatinius duomenis.

11.9. Tiesinės regresijos algoritmas

Tai yra sprendimų medžio algoritmo variacija, kurioje parametras *MINIMUM_LEAF_CASES* yra didesnis arba lygus suminiam duomenų rinkinyje esančių

atvejų skaičiui. Šiuo atveju algoritmas niekada nesukurs išsišakojimo, taigi gaunama tiesinė regresija.

Tiesinę regresiją galima naudoti, nustatant ryšius tarp dviejų tolydžių stulpelių. Ryšiai įgauna linijos lygties formą, geriausiai atvaizduojančią duomenų eilutes.



18 pav. Geriausias galimas duomenų tiesinis vaizdas

18 paveiksle esančios linijos lygtis įgauna paprastą formą y = ax + b. Kintamasis y reiškia išvesties kintamąjį, x reiškia įvesties kintamąjį, o a ir b yra pasirenkami koeficientai. Kiekvienas duomenų taškas turi neatitikimą, susietą su atstumu iki regresijos linijos. Formulėje esantys koeficientai a ir b reguliuoja regresijos linijos kampą ir vietą. Reguliuojant parametrus a ir b, kol su taškais susijusių neatitikimų suma pasiekia mažiausią skaičių, galima išgauti regresijos lygtį. [8]

Tiesinės regresijos modelis privalo turėti rakto stulpelį, įvesties stulpelius ir bent vieną prognozuojamą stulpelį.

11.10. Algoritmų palyginimas

5 lentelė. Algoritmų taikymo pavyzdžiai

Analitinė problema	Pavyzdžiai	Tinkami algoritmai
Klasifikacija: atvejų priskyrimas nustatytoms klasėms	Kredito rizikos analizė Klientų lojalumo skatinimas	Sprendimo medžių Paprasto atskyrimo Neuroninių tinklų
Segmentacija: panašių atvejų grupavimas	Vartotojo profilio analizė Informacijos skleidimo paštu kampanija	Sekų klasterių Laiko sekų klasterių
Asociacija: pažangus koreliacijų skaičiavimas	Prekių krepšelio analizė Pažangi duomenų analizė	Sprendimo medžių Asociacijų
Laiko serijų spėjimas: ateities numatymas	Pardavimų prognozavimas Akcijų vertės prognozavimas	Laiko sekų

Spėjimas: nuspėti naujo atvejo vertę, atsižvelgiant į panašius atvejus	Draudimo įmokų nustatymas Kliento pelno prognozavimas	Visi
Nukrypimų analizė: nustatymas kaip atvejis ar segmentas skiriasi nuo kitų	Naudojimosi kreditinėmis kortelėmis apgavysčių nustatymas Tinklo įkvėpimo (angl. *infusion*) analizė	Visi

12. DUOMENŲ BAZĖS MODELIS

Darbo realizacijai panaudoti duomenys iš kompanijos „Microsoft“ teikiamos mokomosios duomenų bazės , platinamos internete. Duomenų bazėje surinkti dviračiais prekiaujančios įmonės pardavimų rezultatai, informacija apie klientus ir produkcijos aprašymas. Pateikiami duomenys nėra sugeneruoti atsitiktiniu būdu, tai yra realūs duomenys, pateikti mokomaisiais tikslais. Atsitiktinis duomenų generavimas iškraipytų algoritmų dėka gaunamus rezultatus ir nesuteiktų analizei reikalingos informacijos.

Internete platinama duomenų bazė turi daug perteklinės informacijos, kuri nėra reikalinga intelektualios duomenų gavybos algoritmų pritaikymui, tad jinai buvo redaguojama. Išanalizuota visa duomenų bazė ir joje esanti informacija. Tai leido atrinkti nereikalingas lenteles ir jas pašalinti iš tolimesnės struktūros.

Atrinktos lentelės dar kartą peržiūrėtos, ir pašalinti nereikalingi lentelių laukai. Atlikus reikiamas duomenų bazės struktūros modifikacijas, sulietuvinta lentelėse esanti informacija. Sulietuvinti visi angliški įrašai išskyrus pavadinimus, vietovardžius ir žmonių vardus bei pavardes. Tokiu būdu išsamiau susipažinta su duomenų bazėje esančia informacija.

12.1. Duomenų bazės struktūra

Duomenų bazę sudaro aštuonios lentelės, kuriose talpinama informacija apie pardavimus, klientus ir produktus. Lentelėje „Geografija“ patalpinta geografinė informacija, identifikuojanti vietovę. Kiekvienas lentelėje esantis miestas priklauso tam tikrai pardavimų teritorijai. Lentelėje „PardavimuTeritorija“ talpinama informacija apie geografines vietoves, kuriose buvo įsigyta produkcijos. Informacija suskirstyta pagal šalis, regionus ir žemynus. Lentelėje „Klientas“ saugoma visa aktuali informacija apie produkciją įsigijusius klientus. Lentelėje „Pardavimai“ patalpinta informacija apie įvykusius pardavimus. Atitinkamai

lentelėje „Produktas“ aprašyta produkcija, kuria prekiauja kompanija. Lentelėje „Laikas“ saugoma astronominio laiko informacija. Lentelėje „ProduktoKategorija“ patalpinta produkcijos kategorijų informacija. Išskiriamos keturios kategorijos: dviračiai, jų dalys, specializuoti drabužiai ir aksesuarai. Lentelėje „ProduktoSubkategorija“ atitinkamai talpinami duomenys apie produkcijos subkategorijas.

Duomenų bazės struktūra aprašyta *6 lentelėje*. Duomenų bazės grafinis vaizdas pateiktas 19 paveiksle.

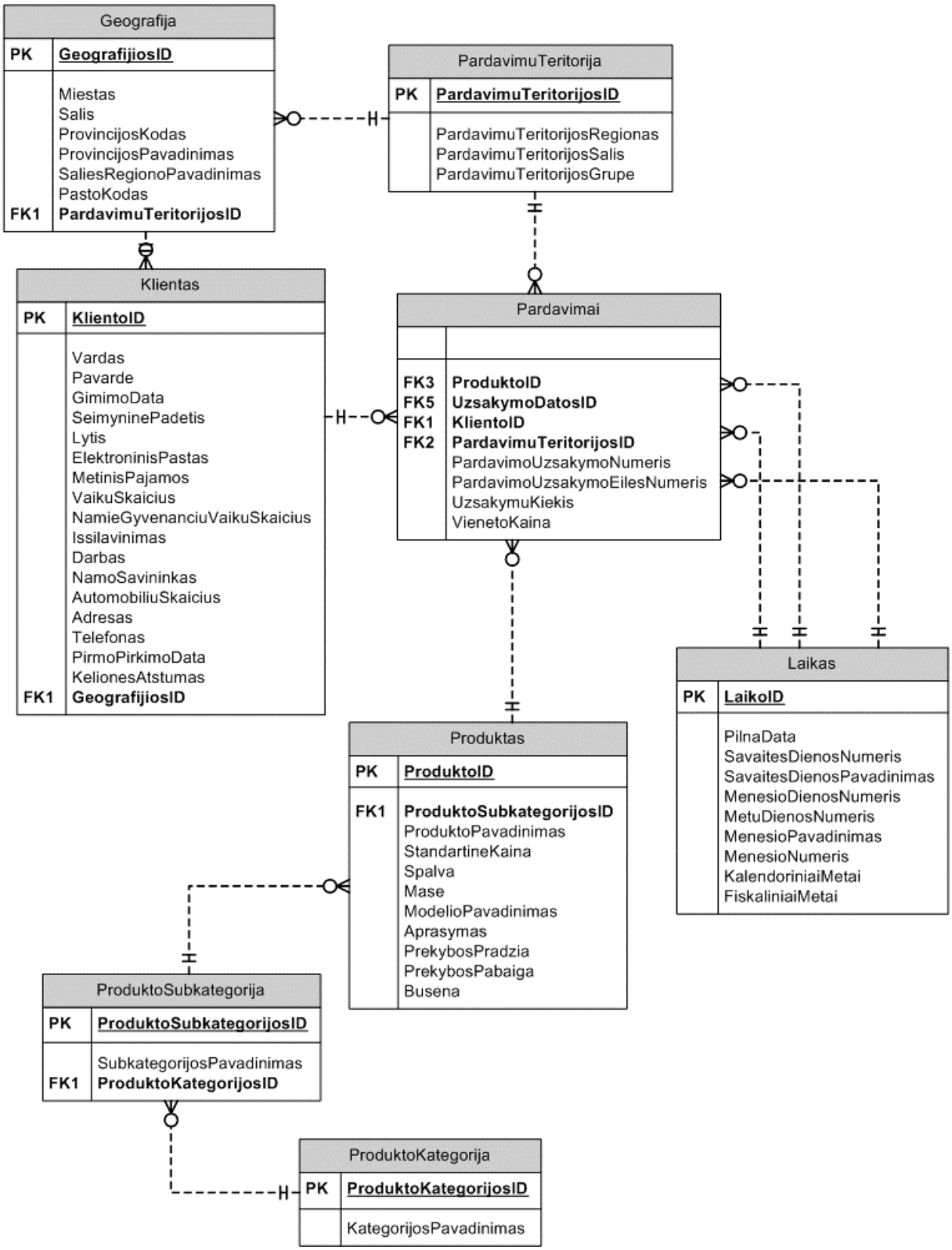
6 lentelė. Duomenų bazės struktūra

Lentelės pavadinimas	**Atributo pavadinimas**	**Atributo tipas**	**Aprašas**
Geografija	GeografijosID	int	Geografinės vietovės identifikatorius, raktas
	Miestas	nvarchar(30)	Miesto pavadinimas
	ProvincijosKodas	nvarchar(3)	Provincijos pavadinimo trumpinys
	ProvincijosPavadinimas	nvarchar(50)	Provincijos pavadinimas
	SaliesRegionoPavadinimas	nvarchar(50)	Šalies pavadinimas
	PastoKodas	nvarchar(15)	Vietovės pašto kodas
PardavimuTeritorija	PardavimuTeritorijosID	int	Vietovės, kurioje parduotas produktas identifikatorius, raktas
	PardavimuTeritorijosRegionas	nvarchar(50)	Regiono, kuriame parduotas produktas pavadinimas
	PardavimuTeritorijosSalis	nvarchar(50)	Šalies, kurioje parduotas produktas pavadinimas
	PardavimuTeritorijosGrupe	nvarchar(50)	Žemyno, kuriame parduotas produktas pavadinimas
Klientas	KlientoID	int	Kliento identifikatorius, raktas
	Vardas	nvarchar(50)	Kliento vardas
	Pavarde	nvarchar(50)	Kliento pavardė
	GimimoData	datetime	Kliento gimimo data
	SeimyninePadetis	nchar(1)	Kliento šeimyninė padėtis
	Lytis	nvarchar(1)	Kliento lytis
	ElektroninisPastas	nvarchar(50)	Kliento elektroninio pašto dėžutės adresas
	MetinesPajamos	money	Kliento metinio pajamų suma
	VaikuSkaicius	tinyint	Turimų vaikų skaičius
	NamieGyvenanciuVaikuSkaicius	tinyint	Namie gyvenančių vaikų skaičius
	Issilavinimas	nvarchar(40)	Kliento išsilavinimas
	Darbas	nvarchar(100)	Kliento darbo pobūdis
	NamoSavininkas	nchar(1)	Aprašymas, ar klientas yra gyvenamosios vietos savininkas
	AutomobiliuSkaicius	tinyint	Turimų automobilių skaičius

	Adresas	nvarchar(120)	Kliento namų adresas
	Telefonas	nvarchar(20)	Kliento kontaktinis telefonas
	PirmoPirkimoData	datetime	Data, kai klientas pirmą kartą įsigijo kompanijos produkcijos
	KelionesAtstumas	nvarchar(15)	Atstumas, kurį klientas įveikia keliaudamas iš namų į darbovietę
Pardavimai	UzsakymoDatosID	int	Produkto identifikatorius, raktas
	ProduktoID	int	Užsakyto produkto identifikatorius
	PardavimoUzsakymoNumeris	nvarchar(20)	Pardavimo užsakymo numeris
	PardavimoUzsakymoEilesNumeris	tinyint	Pardavimo užsakymo eilės numeris
	UzsakymuKiekis	smallint	Užsakymų kiekis
	VienetoKaina	money	Pardavimo vieneto kaina
	StandartineKaina	money	Standartinė kaina
Produktas	ProduktoID	int	Produkto identifikatorius
	ProduktoPavadinimas	nvarchar(50)	Produkto pavadinimas
	StandartineKaina	money	Standartinė kaina
	Spalva	nvarchar(15)	Produkto spalva
	Mase	float	Produkto masė
	ModelioPavadinimas	nvarchar(50)	Modelio pavadinimas
	Aprasymas	nvarchar(400)	Produkto aprašymas
	PrekybosPradzia	datetime	Prekiavimo produktu pradžios data
	PrekybosPabaiga	datetime	Prekiavimo produktu pabaigos data
	Busena	nvarchar(7)	Ar produktas dar yra prekyboje
Laikas	LaikoID	int	Laiko identifikatorius
	PilnaData	datetime	Data be laiko
	SavaitesDienosNumeris	tinyint	Savaitės dienos numeris
	SavaitesDienosPavadinimas	nvarchar(10)	Savaitės dienos pavadinimas
	MenesioDienosNumeris	tinyint	Mėnesio dienos numeris
	MetuDienosNumeris	tinyint	Metų dienos numeris
	MenesioPavadinimas	nvarchar(10)	Mėnesio pavadinimas
	MenesioNumeris	tinyint	Metų mėnesio numeris
	KalendoriniaiMetai	char(4)	Kalendoriniai metai
	FiskaliniaiMetai	char(4)	Fiskaliniai metai
ProduktoSubkategorija	ProduktoSubkategorijosID	int	Produkto subkategorijos identifikatorius
	SubkategorijosPavadinimas	nvarchar(50)	Produkto subkategorijos pavadinimas
ProduktoKategorija	ProduktoKategorijosID	int	Produkto kategorijos identifikatorius
	ProduktoKategorijosPa	nvarchar(50)	Produkto kategorijos pavadinimas

	vadinimas		

12.2. Duomenų bazės schema

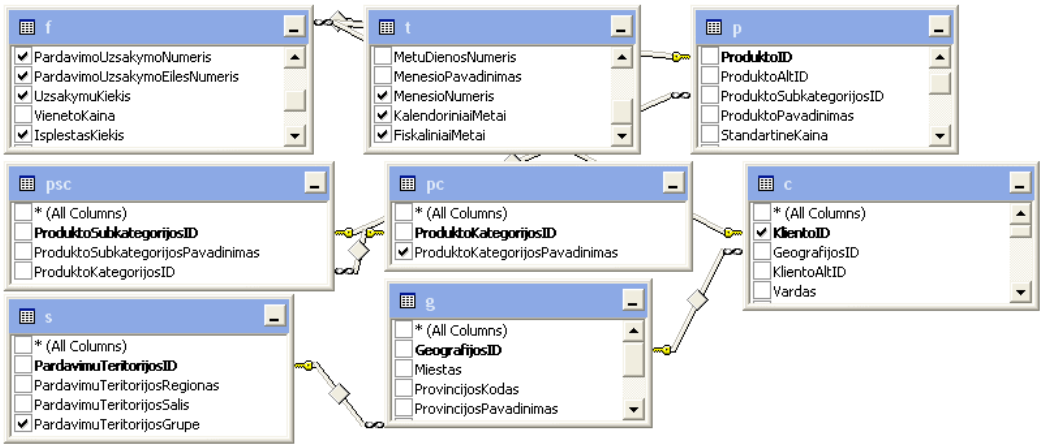


19 pav. Duomenų bazės schema

12.3. Naudojamų poschemių specifikacija

7 lentelė. Poschemės „DMParuosimas“ aprašas

Aprašas	Duomenų bazėje esančios informacijos atrinkimas intelektualios duomenų gavybos proceso realizacijai
Įeinančios lentelės	Pardavima, Laikas, Produktas, ProduktoSubkategorija, ProduktoKategorija, Klientas, Geografija, PardavimuTeritorija
Poschemės pavadinimas	DMParuosimas
Kodas	`SELECT pc.ProduktoKategorijosPavadinimas, COALESCE (p.ModelioPavadinimas, p.ProduktoPavadinimas) AS Modelis, c.KlientoID, s.PardavimuTeritorijosGrupe AS Regionas, CASE WHEN Month(GetDate()) < Month(c.[GimimoData]) THEN DateDiff(yy, c.[GimimoData], GetDate()) - 1 WHEN Month(GetDate()) = Month(c.[GimimoData]) AND Day(GetDate()) < Day(c.[GimimoData]) THEN DateDiff(yy, c.[GimimoData], GetDate()) - 1 ELSE DateDiff(yy, c.[GimimoData], GetDate()) END AS Amzius, CASE WHEN c.[MetinesPajamos] < 40000 THEN 'Mazos' WHEN c.[MetinesPajamos] > 60000 THEN 'Dideles' ELSE 'Vidutines' END AS PajamuGrupe, t.KalendoriniaiMetai, t.FiskaliniaiMetai, t.MenesioNumeris AS Menesis, f.PardavimoUzsakymoNumeris AS UzsakymoNumeris f.PardavimoUzsakymoEilesNumeris AS EilesNumeris, f.UzsakymuKiekis AS Kiekis, f.IsplestasKiekis AS Suma FROM dbo.Pardavimai AS f INNER JOIN dbo.Laikas AS t ON f.UzsakymoDatosID = t.LaikoID INNER JOIN dbo.Produktas AS p ON f.ProduktoID = p.ProduktoID INNER JOIN dbo.ProduktoSubkategorija AS psc ON p.ProduktoSubkategorijosID = psc.ProduktoSubkategorijosID INNER JOIN dbo.ProduktoKategorija AS pc ON psc.ProduktoKategorijosID = pc.ProduktoKategorijosID INNER JOIN dbo.Klientas AS c ON f.KlientoID = c.KlientoID INNER JOIN dbo.Geografija AS g ON c.GeografijosID = g.GeografijosID INNER JOIN dbo.PardavimuTeritorija AS s ON g.PardavimuTeritorijosID = s.PardavimuTeritorijosID`



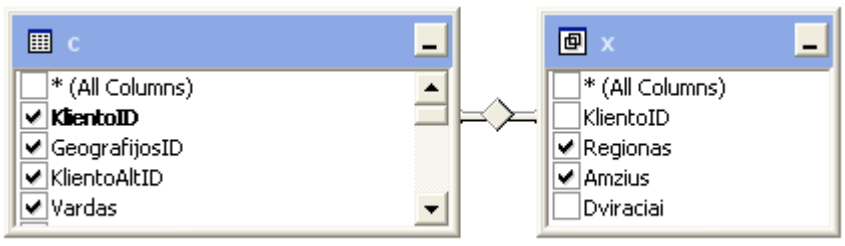
20. pav. Poschemės „DMParuosimas“ struktūra

ProduktoKateg...	Modelis	KlientoID	Regionas	Amzius	PajamuGrupe	KalendoriniaiMetai
Dviraciai	Mountain-100	11003	Okeanija	40	Dideles	2001
Dviraciai	Road-650	14501	Siaures Amerika	70	Dideles	2001
Dviraciai	Road-150	21768	Siaures Amerika	62	Dideles	2001
Dviraciai	Mountain-100	25863	Siaures Amerika	62	Vidutines	2001
Dviraciai	Mountain-100	28389	Europa	43	Mazos	2001
Dviraciai	Mountain-100	11005	Okeanija	43	Dideles	2001
Dviraciai	Mountain-100	11011	Okeanija	45	Vidutines	2001
Dviraciai	Road-150	16624	Okeanija	36	Dideles	2001
Dviraciai	Road-150	27645	Siaures Amerika	47	Dideles	2001
Dviraciai	Road-150	13513	Europa	67	Mazos	2001

21. pav. Poschemės „DMParuosimas“ rezultatai

8 lentelė. Poschemės „ReklaminisPastas“ aprašas

Aprašas	Atrenkama informacija apie klientus ir nustatoma, ar žmogus pirko dviratį
Įeinančios lentelės	Klientas
Poschemės pavadinimas	ReklaminisPastas
Kodas	`SELECT c.KlientoID, c.GeografijosID, c.KlientoAltID, c.Vardas, c.Pavarde, c.GimimoData, c.SeimyninePadetis, c.Lytis, c.ElektroninisPastas, c.MetinesPajamos, c.VaikuSkaicius, c.NamieGyvenanciuVaikuSkaicius, c.Issilavinimas, c.Darbas, c.NamoSavininkas, c.AutomobiliuSkaicius, c.Adresas, c.Telefonas, c.PirmoPirkimoData, c.KelionesAtstumas, x.Regionas, x.Amzius, CASE x.[Dviraciai] WHEN 0 THEN 0 ELSE 1 END AS DviracioPirkejas FROM dbo.Klientas AS c INNER JOIN (SELECT KlientoID, Regionas, Amzius, SUM(CASE [ProduktoKategorijosPavadinimas] WHEN 'Dviraciai' THEN 1 ELSE 0 END) AS Dviraciai FROM dbo.DMParuosimas GROUP BY KlientoID, Regionas, Amzius) AS x ON c.KlientoID = x.KlientoID`



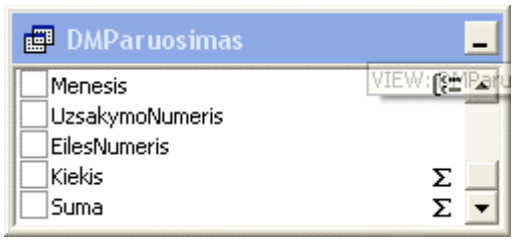
22. pav. Poschemės „ReklaminisPastas“ struktūra

KlientoID	GeografijosID	KlientoAltID	Vardas	Pavarde	GimimoData	SeimyninePadetis	Lytis
11000	26	AW00011000	Jon	Yang	1966.04.08 00:...	V	V
11006	8	AW00011006	Janet	Alvarez	1965.12.06 00:...	N	M
11007	40	AW00011007	Marco	Mehta	1964.05.09 00:...	V	V
11008	32	AW00011008	Rob	Verhoff	1964.07.07 00:...	N	M
11009	25	AW00011009	Shannon	Carlson	1964.04.01 00:...	N	V
11014	634	AW00011014	Sydney	Bennett	1968.05.09 00:...	N	M
11015	301	AW00011015	Chloe	Young	1979.02.27 00:...	N	M

23. pav. Poschemės „ReklaminisPastas“ rezultatai

9 lentelė. Poschemės „LaikoSerijos“ aprašas

Aprašas	Sukuriamas modelio regiono matmuo ir apskaičiuojami pardavimų kiekiai.
Įeinančios lentelės	DMParuosimas
Poschemės pavadinimas	LaikoSerijos
Kodas	`SELECT CASE [Modelis] WHEN 'Mountain-100' THEN 'M200' WHEN 'Road-150' THEN 'R250' WHEN 'Road-650' THEN 'R750' WHEN 'Touring-1000' THEN 'T1000' ELSE LEFT([Modelis], 1) + RIGHT([Modelis], 3) END + ' ' + Regionas AS ModelioRegionas, CONVERT(Integer, KalendoriniaiMetai) * 100 + CONVERT(Integer, Menesis) AS LaikoIndeksas, SUM(Kiekis) AS Kiekis, SUM(Suma) AS Suma FROM dbo.DMParuosimas WHERE (Modelis IN ('Mountain-100', 'Mountain-200', 'Road-150', 'Road-250', 'Road-650', 'Road-750', 'Touring-1000')) GROUP BY CASE [Modelis] WHEN 'Mountain-100' THEN 'M200' WHEN 'Road-150' THEN 'R250' WHEN 'Road-650' THEN 'R750' WHEN 'Touring-1000' THEN 'T1000' ELSE LEFT(Modelis, 1) + RIGHT(Modelis, 3) END + ' ' + Regionas, CONVERT(Integer, KalendoriniaiMetai) * 100 + CONVERT(Integer, Menesis)`



24. pav. Poschemės „LaikoSerijos“ struktūra

ModelioRegionas	LaikoIndeksas	Kiekis	Suma
00 Siaures Amerika	200212	43	88513,0078
M200 Europa	200306	42	86597,8380
R250 Europa	200211	32	73736,8125
R750 Okeanija	200305	18	14093,8200
R750 Okeanija	200405	42	22679,5800
R250 Okeanija	200204	40	143130,8000
R250 Europa	200203	30	107348,1000
M200 Okeanija	200309	46	106169,5400
R750 Siaures A...	200107	7	4893,6874
R250 Europa	200111	21	75143,6700
R250 Europa	200403	16	39093,6000
R250 Europa	200303	46	106111,2000
T1000 Europa	200311	35	83442,4500

25. pav. Poschemės „LaikoSerijos“ rezultatai

10 lentelė. Poschemės „AsocijuotuSekuUzsakymai“ aprašas

Aprašas	Išrenkama atitinkamų metų užsakymų informacija
Įeinančios lentelės	DMParu osimas
Poschemės pavadinimas	AsocijuotuSekuUzsakymai
Kodas	`SELECT DISTINCT UzsakymoNumeris, KlientoID, Regionas, PajamuGrupe FROM dbo.DMParuosimas WHERE FiskaliniaiMetai = '2004')`

26. pav. Poschemės „AsocijuotuSekUzsakymai“ struktūra

UzsakymoNumeris	KlientoID	Regionas	PajamuGrupe
SO51176	18239	Okeanija	Dideles
SO51177	27873	Okeanija	Mazos
SO51178	11245	Europa	Dideles
SO51179	22430	Europa	Vidutines
SO51180	16313	Europa	Mazos
SO51181	12132	Europa	Dideles
SO51182	22998	Siaures Amerika	Dideles
SO51183	20662	Siaures Amerika	Dideles
SO51184	11263	Siaures Amerika	Dideles
SO51185	27767	Europa	Mazos
SO51186	24339	Siaures Amerika	Vidutines

27. pav. Poschemės „AsocijuotuSekuUzsakymai“ rezultatai

11 lentelė. Poschemės „AsocijuotuSekuEilesElementai“ aprašas

Aprašas	Išrenkama atitinkamų metų užsakymų informacija
Įeinančios lentelės	DMParuosimas
Poschemės pavadinimas	AsocijuotuSekuEilesElementai

Kodas	`SELECT UzsakymoNumeris, EilesNumeris, Modelis`
	`FROM dbo.DMParuosimas`
	`WHERE (FiskaliniaiMetai = '2004')`

28. pav. Poschemės „AsocijuotuSekuElemtai“ struktūra

UzsakymoNumeris	EilesNumeris	Modelis
SO51178	1	Mountain-200
SO51178	2	Mountain Bottle ...
SO51178	3	Water Bottle
SO51184	1	Mountain-200
SO51184	2	HL Mountain Tire
SO51184	3	Mountain Tire Tube
SO51184	4	Sport-100
SO51181	1	Road-250
SO51181	2	Road Tire Tube
SO51181	3	HL Road Tire
SO51181	4	Sport-100
SO51188	1	Road-750
SO51180	1	Road-250

29. pav. Poschemės „AsocijuotuSekuElementai“ rezultatai

13. DUOMENŲ GAVYBOS MODELIO KŪRIMAS

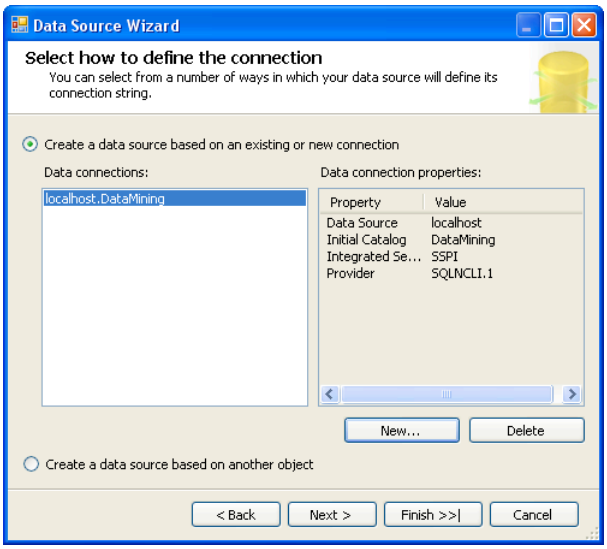
13.1. Reklaminių pasiūlymų siuntimas

Marketingo skyrius nori padidinti pardavimų skaičių. To bus siekiama atrinktiems klientams siunčiant produkcijos įsigijimo pasiūlymus. Analizuojant jau turimų klientų duomenis, siekiama atrasti šablonus, kuriuos vėliau būtų galima pritaikyti potencialių klientų

paieškai. Siekiama panaudoti atrastas tendencijas, spėjant kurie potencialūs klientai labiausiai linkę įsigyti kompanijos produkcijos. Taipogi ieškoma duomenų bazėje esančių klientų loginio grupavimo, kuris pagelbėtų nustatyti regione gyvenančių klientų poreikius. Verslo įžvalgos kūrimo studijos pakete sukuriamas naujas analizės servisų projektas.

13.1.1. Duomenų šaltinio kūrimas

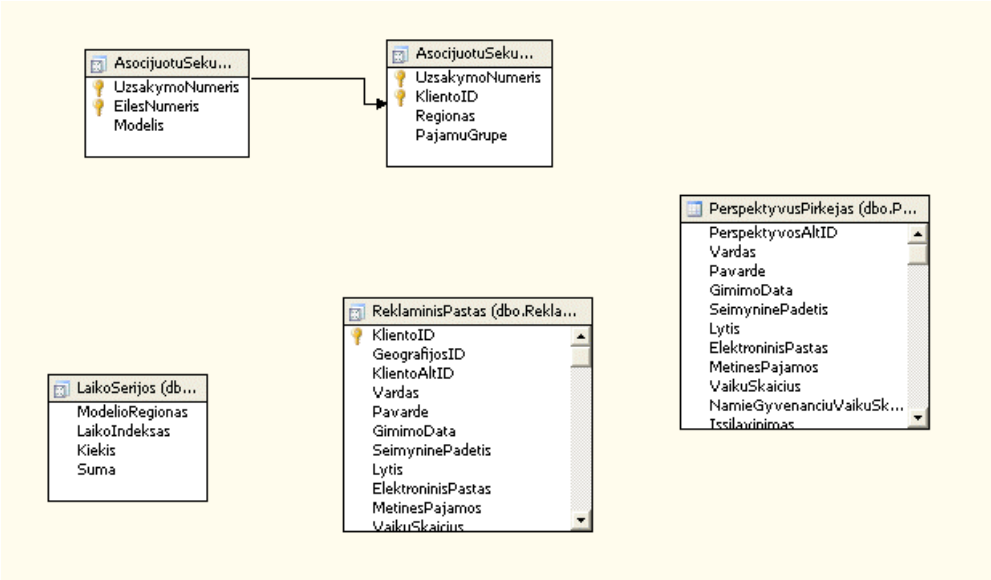
Duomenų šaltinis yra paprastas objektas, kuriame saugomi prisijungimo prie duomenų bazės parametrai bei papildoma informacija atspindinti kokiu būdu jungiamasi prie duomenų bazės. Kitaip negu dauguma intelektualios duomenų gavybos produktų, "SQL Server Data Mining" yra serverio architektūra grindžiamas sprendimas. Tai reiškia, kad nustatant duomenų šaltinį, jis turi būti pasiekiamas ne tik klientui, kuriame kuriamas modelis, tačiau ir serveriui, kuriame modelis bus apdorojamas.



30 pav. Duomenų šaltinis

13.1.2. Duomenų šaltinio poschemės kūrimas

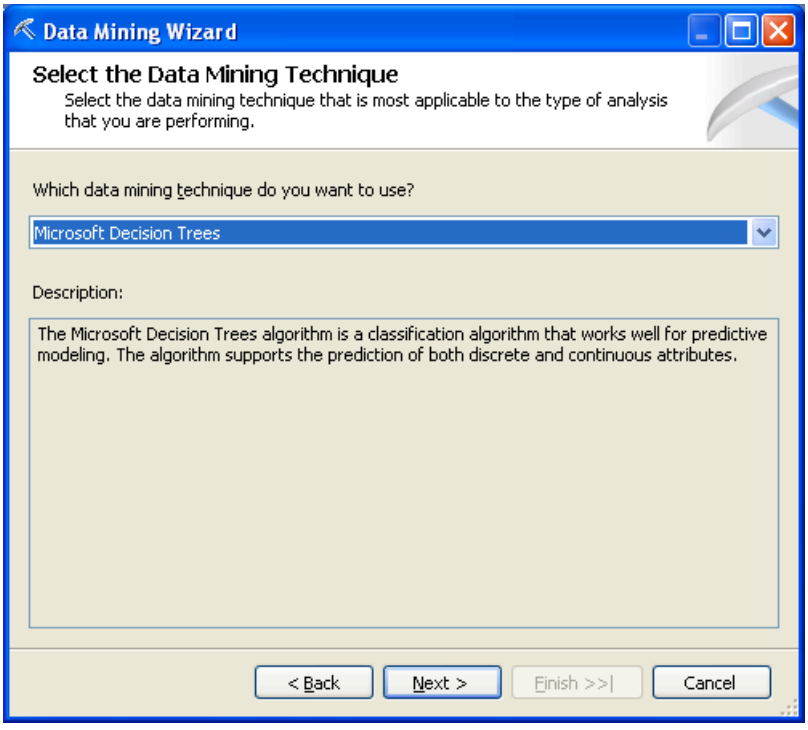
Duomenų šaltinio poschemė (angl. data source view) yra abstraktus duomenų atvaizdas kliento pusėje. Šioje vietoje prasideda modeliavimas: pasirenkami, tvarkomi ir peržiūrimi duomenų šaltinyje esantys duomenys. Duomenų šaltinio poschemė nurodo analizės servisams, kaip atvaizduoti duomenų šaltinyje esančią informaciją. Kuriant duomenų šaltinio poschemę intelektualios gavybos uždaviniams spręsti svarbiausia identifikuoti atvejo (angl. case) lentelę. Šioje lentelėje yra norimi analizuoti atvejai. Taipogi reikia nurodyti visas susijusias lenteles, kurios suteikia papildomos informacijos apie atvejus. Duomenų šaltinio poschemės dizaineris atvaizduoja duomenų šaltinio lenteles ir ryšius tarp jų.



31 pav. Duomenų šaltinio poschemė

13.1.3. Sprendimų medžio algoritmo pritaikymas

32 paveiksle pateikto intelektualios duomenų gavybos vedlio pagalba kuriamas gavybos modelis. Pasirenkamas norimas intelektualios gavybos algoritmas, nurodomi įvesties ir nuspėjami stulpeliai.

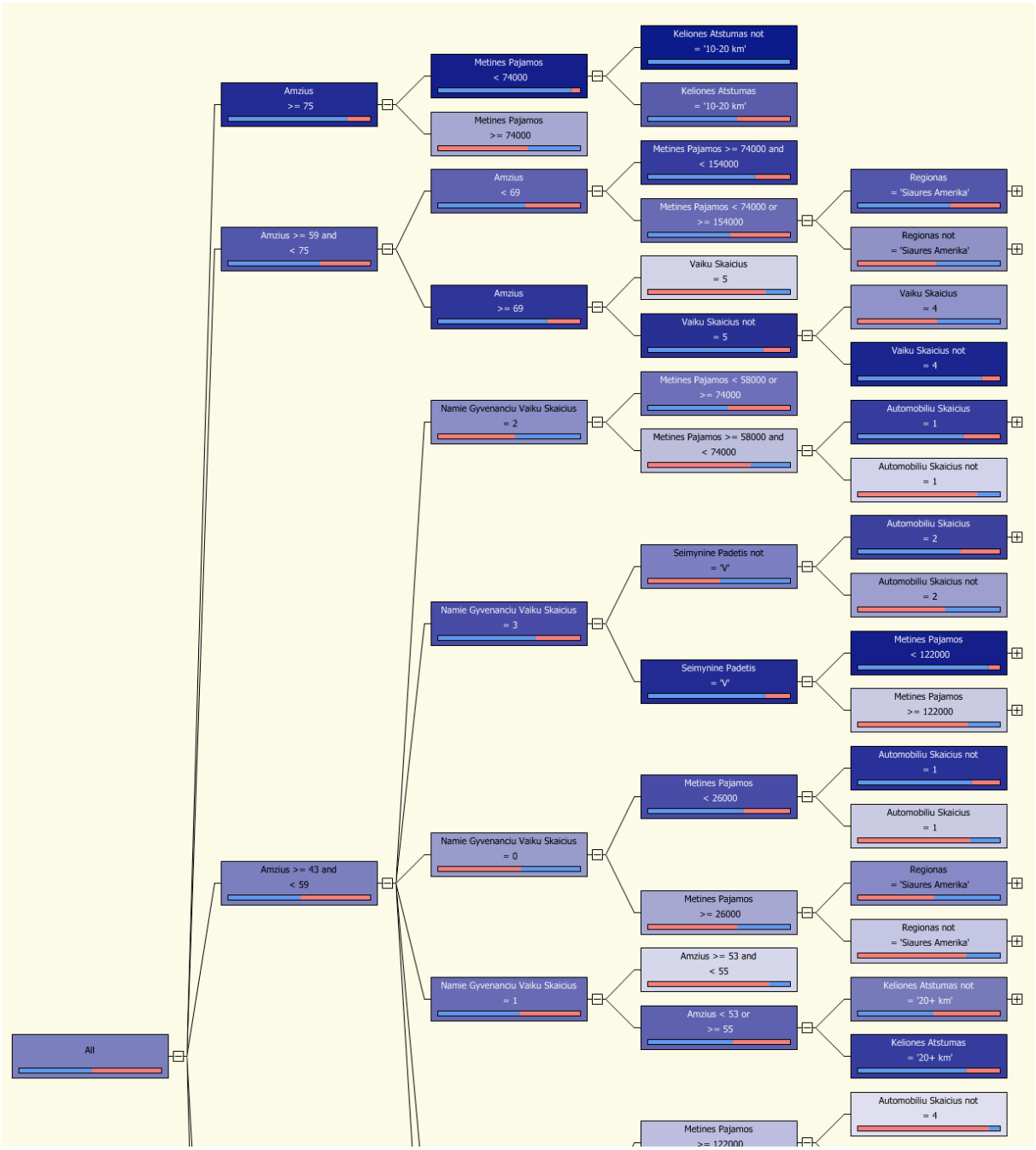


32 pav. Duomenų gavybos modelio kūrimo vedlys

Atlikus visus konfigūravimo darbus, paleidžiamas projektas. Atsiradusiame lange pasirenkama „Mining Model Viewer“ skiltis. Modelis sukurtas panaudojus sprendimų medžio

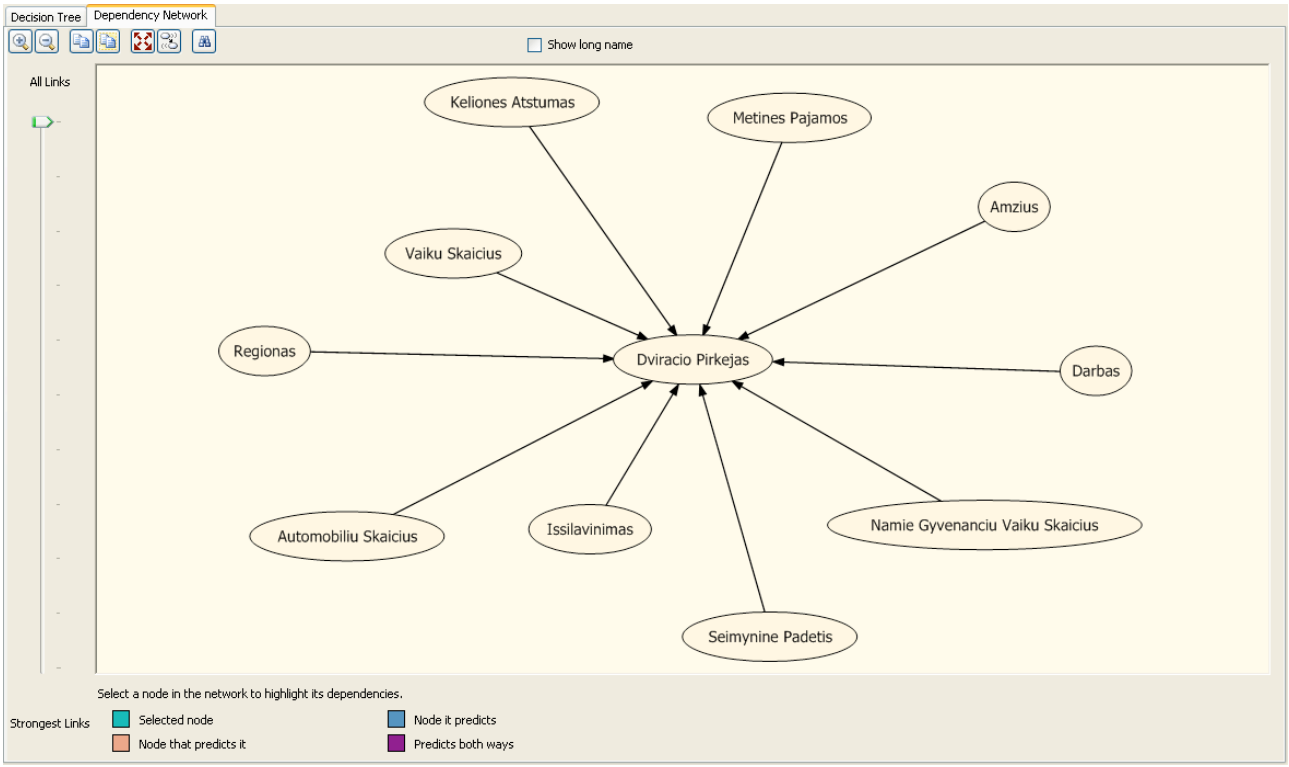
algoritmą grafiniame atvaizdavime turi dvi skiltis: sprendimų medį (žr. 33 pav.) ir priklausomybių grafiką. Kiekvienas sprendimų medyje esantis mazgas atvaizduoja šia informaciją:

- Būseną, kuri yra reikalinga, norint pasiekti mazgą iš prieš tai buvusiojo.
- Histogramą, apibrėžiančią būsenų pasiskirstymą pagal nuspėjamą stulpelį.
- Atvejų koncentraciją – kuo tamsesnis medyje esantis mazgas, tuo didesnė tikimybė, kad tos kategorijos klientas įsigys produkcijos.



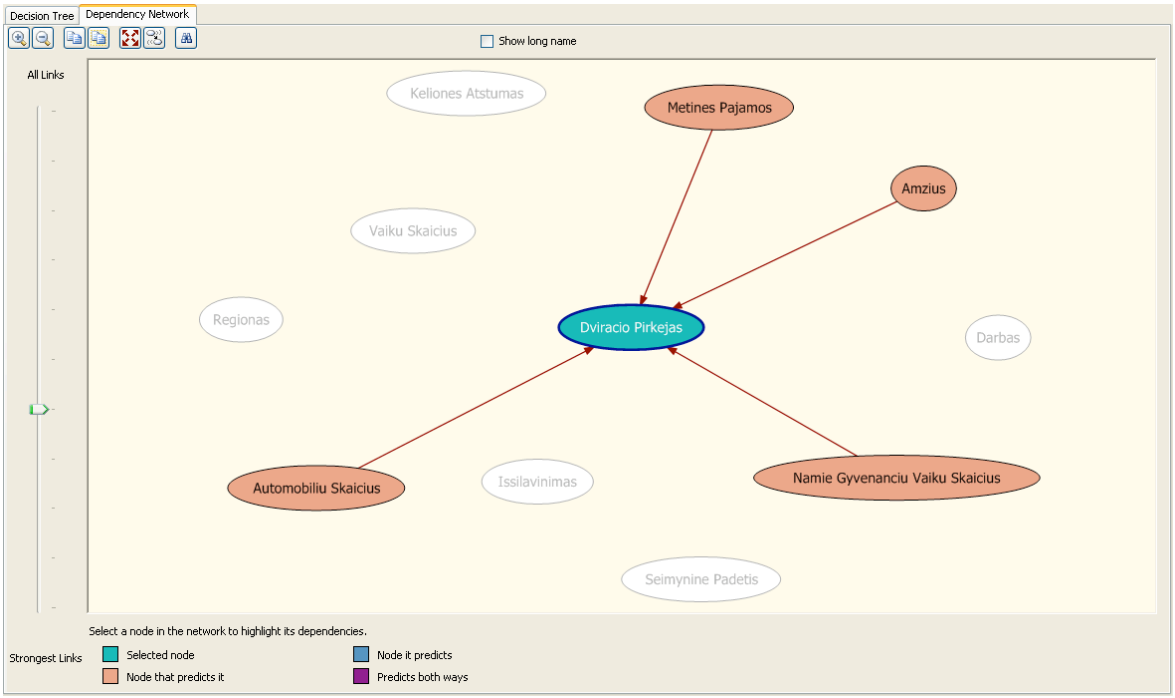
33 pav. Sprendimų medžio fragmentas

34 paveiksle pateiktame pilname priklausomybių tinkle matomi ryšiai tarp nuspėjamo stulpelio ir jį įtakojančių įvesties stulpelių. Vidurinis priklausomybių tinklo mazgas yra nuspėjamas stulpelis. Kiekvienas šalia esantis mazgas atvaizduoja atributus, įtakojančius numatomo stulpelio pardavimus.



34 pav. Pilnas priklausomybių tinklas

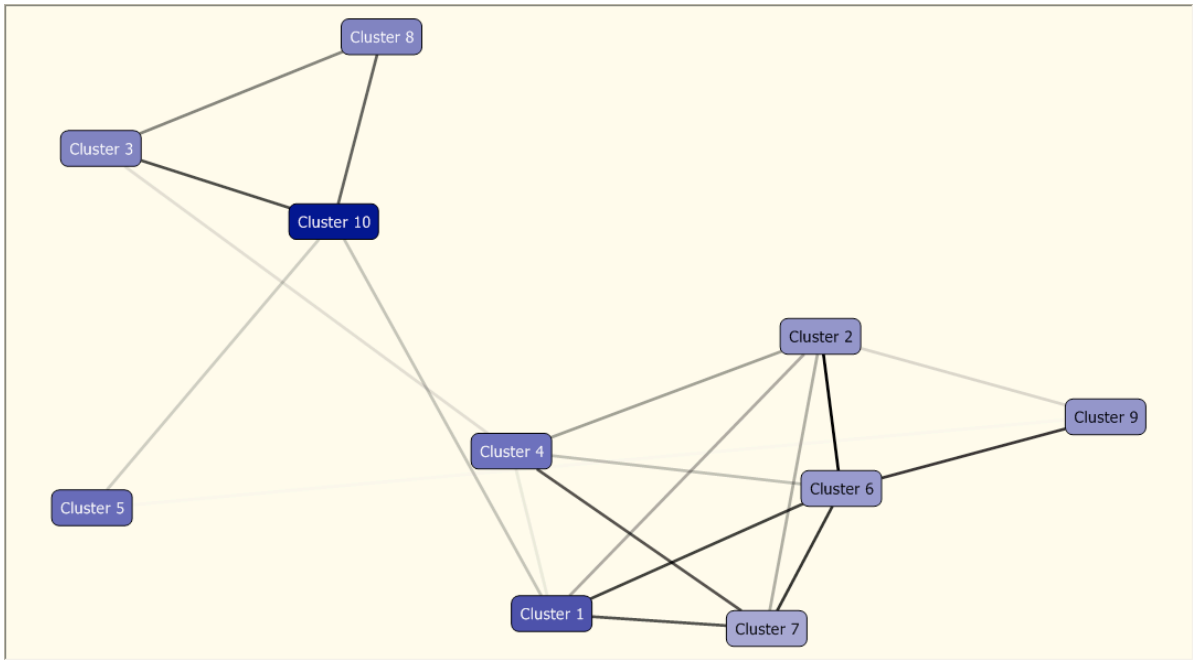
35 paveiksle atvaizduojami didžiausią įtaką produkto pardavimui turintys veiksniai – tai yra atfiltruojami tiktai stipriausi ryšiai. Nuspėjamas mazgas atvaizduotas žalsva spalva, o jį įtakojantys veiksniai ruda. Kaip matyti iš paveikslėlio pasirinkus stipresnius ryšius atkrenta dalis nuspėjamą stulpelį įtakojančių atributų. Didžiausią įtaką kliento apsisprendimui įsigyti dviratį turi metinės pajamos, asmens amžius, turimų automobilių skaičius ir namie gyvenančių vaikų skaičius.



35 pav. Sprendimą įsigyti produkciją labiausiai įtakojantys veiksniai

13.1.4. Klasterių algoritmo pritaikymas

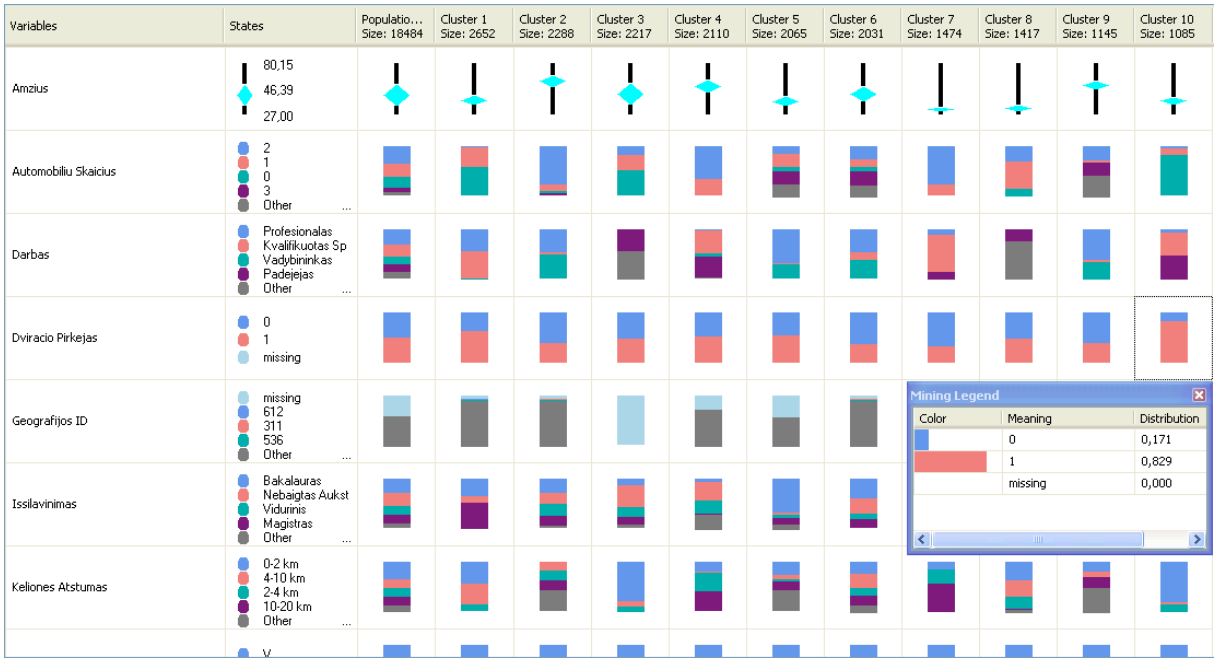
„Cluster Viewer“ skiltyje atvaizduojami visi gavybos modelyje esantys klasteriai. Linijos, jungiančios vieną klasterį su kitu, spalva atspindi klasterių panašumo stiprumą. Jeigu linija šviesi arba jos nėra, tai reiškia, kad klasteriai nėra panašūs. Kuo tamsesnis linijos atspalvis, tuo klasteriai yra panašesni.



36 pav. Ryšiai tarp klasterių

Pastebėta, kad 10 klasteryje yra didžiausias įsigyjančių produkciją žmonių skaičius – jis siekia 83 procentus, tuo tarpu mažiausiai produkcijos įsigyja žmonės esantys 7 klasteryje – tiktai 34%. Stipriausias ryšys egzistuoja tarp antrojo ir šeštojo klasterių.

Klasterių profilių sekcijoje pateikiamas bendras algoritmo sukuriamų klasterių vaizdas. Jame atvaizduojamas kiekvienas atributas ir jo pasiskirstymas kiekviename klasteryje. Pirmame stulpelyje atvaizduojami atributai, kurie yra susiję bent su vienu klasteriu, likusiuose stulpeliuose atributų būklių pasiskirstymas klasteriuose. Kiekvieno eilutėje esanti informacija atspindi pasiskirstymo statistiką, o kiekvieno stulpelio antraštėje atvaizduojamas klasterio dydis. Diskretūs atributai pateikiami spalvotų stulpelių pavidalų, o tolydūs deimanto atvaizdu. Kiekviename deimanto grafike atvaizduojamas klasterio atributo reikšmių vidurkis ir standartinis nuokrypis.

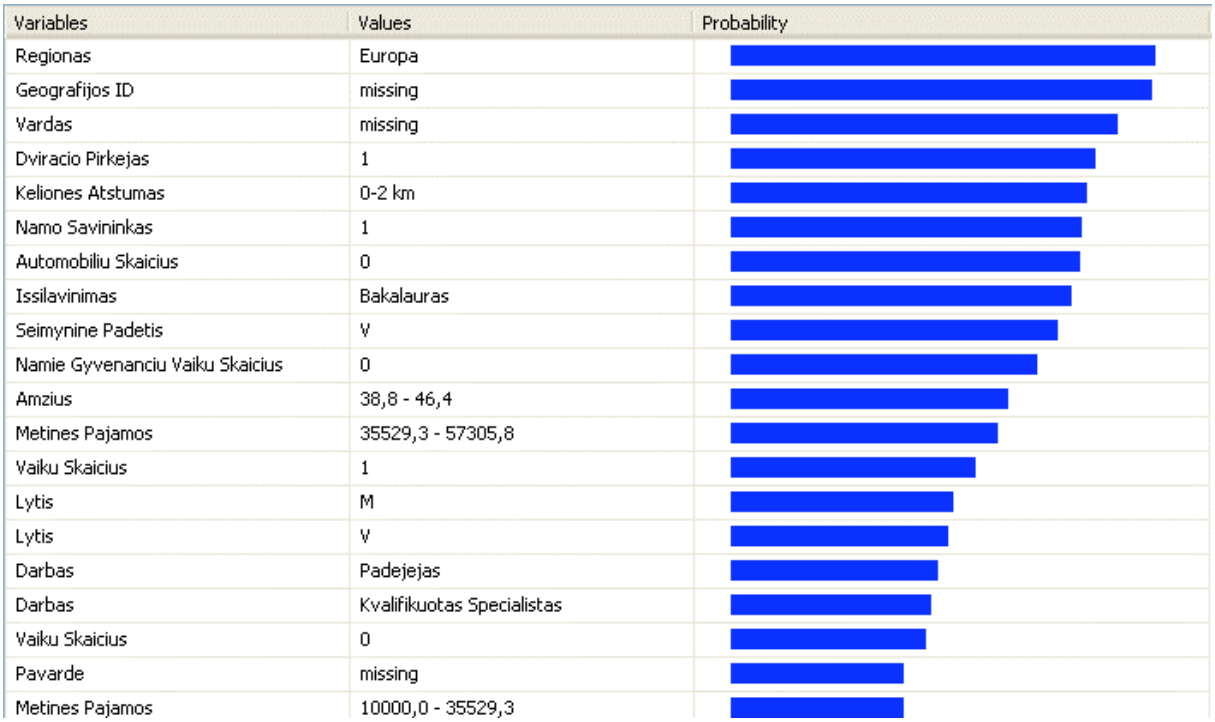


37 pav. Bendras klasterių vaizdas

Kiekvieno atributo būsenos atvaizduojamos „State“ stulpelyje. Kaip matoma paveiksle didžiausias yra pirmas klasteris – jo dydis 2652 klientai, tuo tarpu mažiausias dešimtas klasteris – 1085 klientai. Didžiausias produkcijos pirkėjų pasiskirstymas pastebimas dešimtame klasteryje: perkančių produkciją klientų yra 0,829, neperkančių – 0,171 procentų. Mažiausias produkcijos pirkimas pastebimas septintame klasteryje, perkančių – 0,338, neperkančių – 0,662 procentų. Pasirinkus bet kurį klasteryje esantį atributą „Mining Legend“ lange atvaizduojama atributo pasiskirstymo klasteryje skaitinė informacija. Detaliau panagrinėkime dešimtojo klasterio informaciją. Klientų amžiaus vidurkis siekia 40,61 metų,

didžioji dalis žmonių (79,1%) neturi nei vieno automobilio, turi bakalauro diplomą (77,2%), o atstumas nuo namų iki darbovietės daugumai žmonių tėra nuo nulio iki dviejų kilometrų (80,6%).

Klasterių charakteristikų sekcijoje iškrentančiame sąraše pasirenkamas norimas klasteris ir matomos jo charakteristikos. Klasterį sudarantys atributai išvardijami stulpelyje „Variables“, o jų būsenos šalia esančiame stulpelyje „Values“. Atributų būsenos išvardinamos pagal svarbumą, kurį atspindi tikimybė, kad šios būsenos pasireikš klasteryje. Tikimybė atvaizduojama kairiausiame stulpelyje „Probability“.

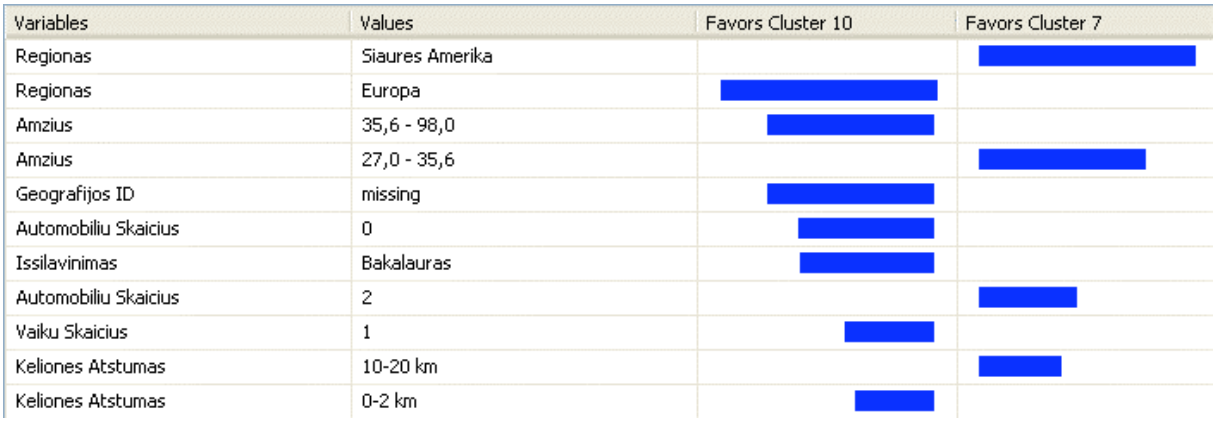


38 pav. Klasterio charakteristikos

38 paveiksle atvaizduotos dešimto klasterio charakteristikos (produkciją labiausiai perkančių klientų klasteris). Iš jo matome, kad klientai praeityje įsigiję dviratį pasižymi šiomis charakteristikomis: jie gyvena Europoje (tikimybė 96,15%), atstumas nuo namų iki darbovietės yra apie du kilometrus (tikimybė 80,64%), turimų automobilių skaičius lygus nuliui (tikimybė 79,13), jie yra vedę (tikimybė 74,1%) ir neturi namie gyvenančių vaikų (tikimybė 69,48%). Visos mažiau įtakos turinčios charakteristikos taipogi matomos paveiksle.

Klasterių diskriminacijos sekcijoje galima analizuoti dviejų klasterių charakteristikas, skiriančias vieną klasterį nuo kito. Iškrentančiame sąraše pasirinkus du klasterius, nustatomi ir atvaizduojami skirtumai tarp klasterių. Atvaizdavimas pradedamas nuo labiausiai besiskiriančių atributų.

Palyginimui pasirinkus septintą ir dešimtą klasterius suformuojamas 39 paveiksle pateiktas vaizdas. Kaip matyti dešimtame klasteryje esantys klientai gyvena Europoje, tuo tarpu septintame klasteryje esantys klientai gyvena Šiaurės Amerikoje. Produkciją perka vyresnio amžiaus žmonės (apytiksliai nuo 36 metų), tuo tarpu jaunesni (iki 36 metų) linkę neįsigyti produkcijos. Neperkantys produkcijos klientai turi du automobilius, tuo tarpu įsigyjantys dviračius klientai automobilių neturi. Neperkančių dviračių klientų atstumas nuo namų iki darbovietės yra apie 15 kilometrų, tuo tarpu įsigyjančių produkciją nuo nulio iki dviejų kilometrų.

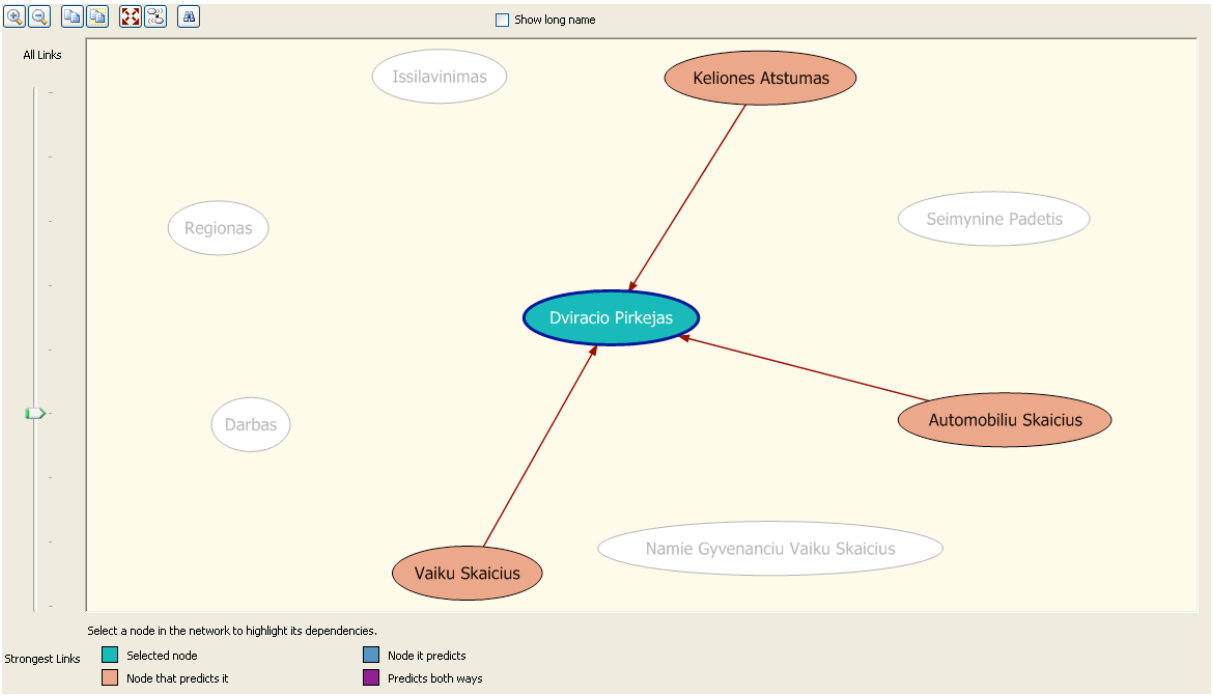


Variables	Values	Favors Cluster 10	Favors Cluster 7
Regionas	Siaures Amerika		
Regionas	Europa		
Amzius	35,6 - 98,0		
Amzius	27,0 - 35,6		
Geografijos ID	missing		
Automobiliu Skaicius	0		
Issilavinimas	Bakalauras		
Automobiliu Skaicius	2		
Vaiku Skaicius	1		
Keliones Atstumas	10-20 km		
Keliones Atstumas	0-2 km		

39 pav. Klasterių palyginimas

13.1.5. Paprasto atskyrimo algoritmo pritaikymas

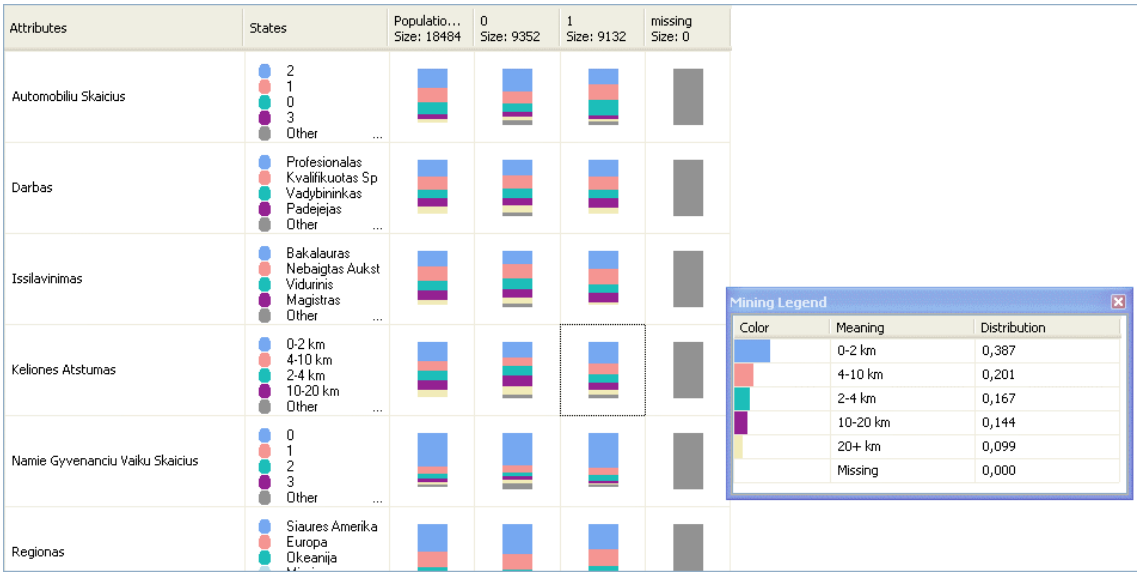
Priklausomybių tinklas atvaizduoja ryšius tarp nuspėjamų atributų ir įvesties atributų. Ši sekcija funkcionuoja analogiškai kaip ir sprendimų medžio algoritmo priklausomybių tinklas.



40 pav. Paprasto atskyrimo algoritmo priklausomybių tinklas

Paprasto atskyrimo algoritmu sudaryto priklausomybių tinklo rezultatai skiriasi nuo sprendimų medžio algoritmu grįsto priklausomybių tinklo. Labiausiai nuspėjamą stulpelį „Dviracio Pirkejas“ įtakojantys veiksniai šiuo atveju yra turimų automobilių skaičius, vaikų skaičius ir atstumas nuo namų iki darbovietės.

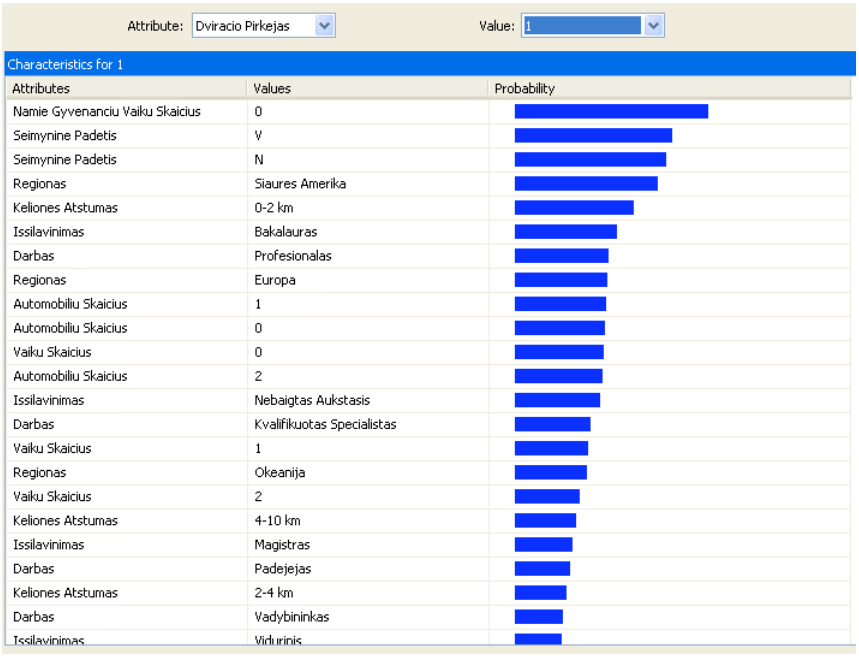
Atributų profilių sekcijoje apibrėžiama kaip skirtingos įvesties atributų būsenos įtakoja nuspėjamo atributo rezultatą. Kiekvienas diagramoje esantis stulpelis atspindi nuspėjamo atributo būseną.



41 pav. Atributų profilių sekcijos vaizdas

Panagrinėjus didžiausią įtaką pardavimui turinčius atributus pastebėta, kad kelionės atstumas yra nuo nulio iki dviejų kilometrų (38,7%) ir nuo keturių iki dešimties kilometrų (20,1%). Automobilių skaičiaus pasiskirstymas yra labai panašus: automobilių neturi apie 30 procentų, vieną automobilį turi taipogi apie 30 procentų potencialių klientų. Neturinčių vaikų klientų skaičius sudaro 29%, tuo tarpu vieną vaiką turi 23,8% klientų.

Atributų charakteristikų sekcijoje iškrentančiame sąraše pasirenkamas nuspėjamas atributas ir jo būsena. Tokiu būdu suformuojama informacija apie atributų būsenas, kurios yra susijusios su pasirinktu nuspėjamo atributo atveju. Atvaizduojami atributai surūšiuojami pagal svarbumą.



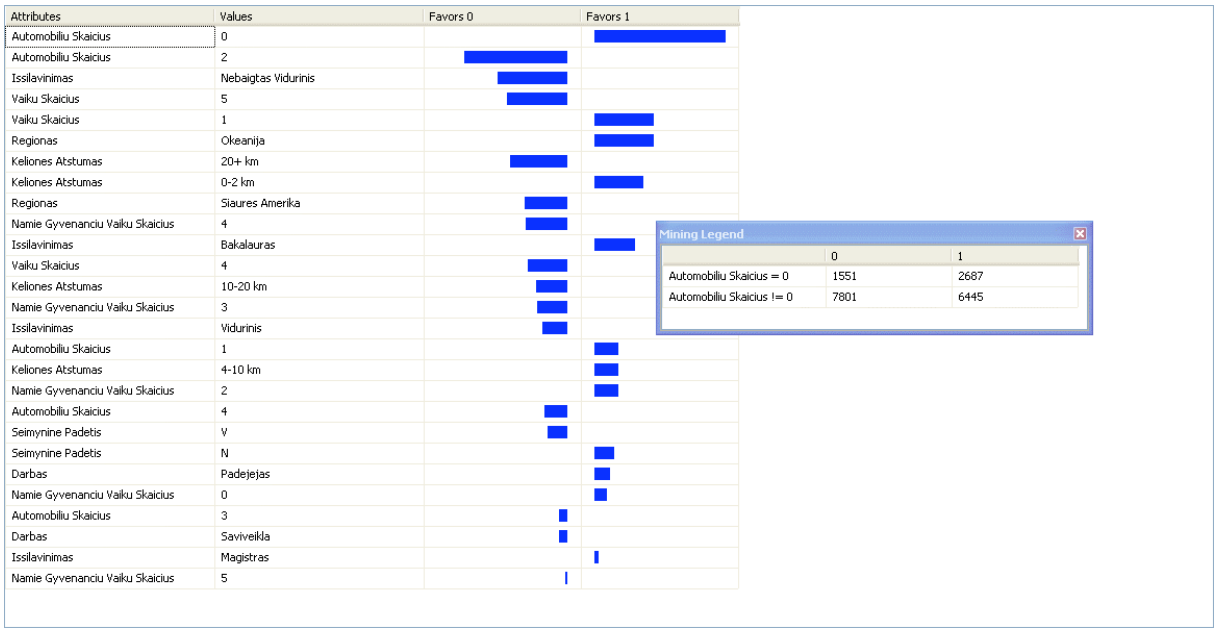
42 pav. Atributų charakteristikų pasiskirstymas

Kaip matyti 42 paveiksle dviračių pirkėjai neturi namie gyvenančių vaikų (tikimybė 62,74%), yra vedę (tikimybė 50,93%), gyvena Šiaurės Amerikoje (tikimybė 46,27%), o atstumas nuo namų iki darbovietės yra apie nuo nulio iki dviejų kilometrų (tikimybė 38,74%).

Atributų diskriminacijos sekcijoje pasirenkamas nuspėjamas atributas ir dvi jo būsenos. Šioje sekcijoje galima išnagrinėti ryšius tarp dviejų diskrečių nuspėjamo atributo reikšmių ir kitų atributų reikšmių. Atributų diskriminacijos sekcijoje atvaizduojama tokia informacija:

- Atributas: duomenų rinkinyje esantys kiti atributai stipriai įtakojantys vieną iš nuspėjamo atributo būsenų.
- Vertės: atributo stulpelyje nurodyto elemento būsena.

- Palankiai veikia <1 reikšmę>: stiprumas, kuriuo būsena palankiai įtakoja pasirinktos 1 reikšmės vertę.
- Palankiai veikia <2 reikšmę>: stiprumas, kuriuo būsena palankiai įtakoja pasirinktos 2 reikšmės vertę.

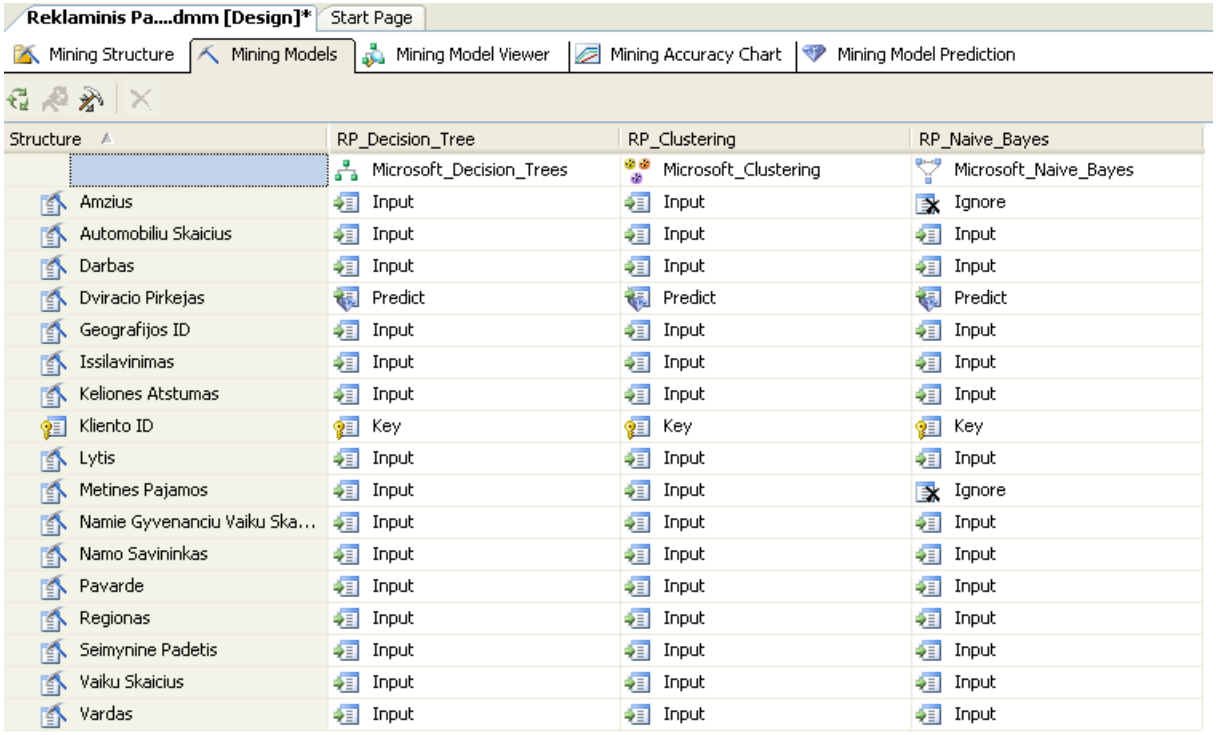


43 pav. Atributų diskriminacijos vaizdas

Kaip matyti dviračius perka žmonės neturintys automobilių, tuo tarpu asmenys turintys du automobilius neperka dviračių.

13.1.6. Modelių suvestinė

Norimam scenarijui įgyvendinti sukurti visi reikalingi modeliai. Struktūra pateikiama 44 paveiksle.



Structure	RP_Decision_Tree	RP_Clustering	RP_Naive_Bayes
	Microsoft_Decision_Trees	Microsoft_Clustering	Microsoft_Naive_Bayes
Amzius	Input	Input	Ignore
Automobiliu Skaicius	Input	Input	Input
Darbas	Input	Input	Input
Dviracio Pirkejas	Predict	Predict	Predict
Geografijos ID	Input	Input	Input
Issilavinimas	Input	Input	Input
Keliones Atstumas	Input	Input	Input
Kliento ID	Key	Key	Key
Lytis	Input	Input	Input
Metines Pajamos	Input	Input	Ignore
Namie Gyvenanciu Vaiku Ska...	Input	Input	Input
Namo Savininkas	Input	Input	Input
Pavarde	Input	Input	Input
Regionas	Input	Input	Input
Seimynine Padetis	Input	Input	Input
Vaiku Skaicius	Input	Input	Input
Vardas	Input	Input	Input

44 pav. Reklaminio pašto modelių struktūra

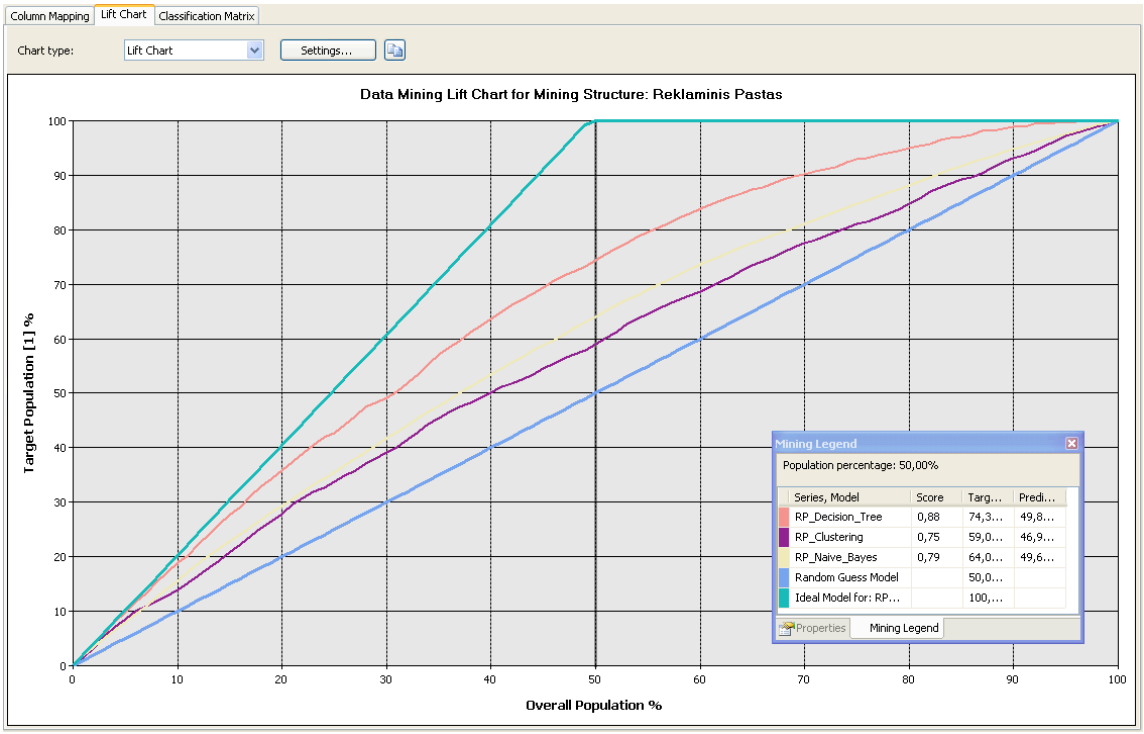
13.1.7. Modelių veikimo palyginimas

Sukurti modeliai testuojami, kad nustatyti, kaip gerai jie atlieka spėjimus, taipogi siekiama išsiaiškinti, ar kažkuris modelis duoda geresnius rezultatus už gaunamus kitais modeliais.

Gavybos tikslumo grafiko sekcijoje matomas sukurtų modelių tikslumas ir kokybė. Tikslumo grafikas atlieka modelių spėjimus ir palygina juos su turimais duomenimis, kurių atsakymas yra žinomas.

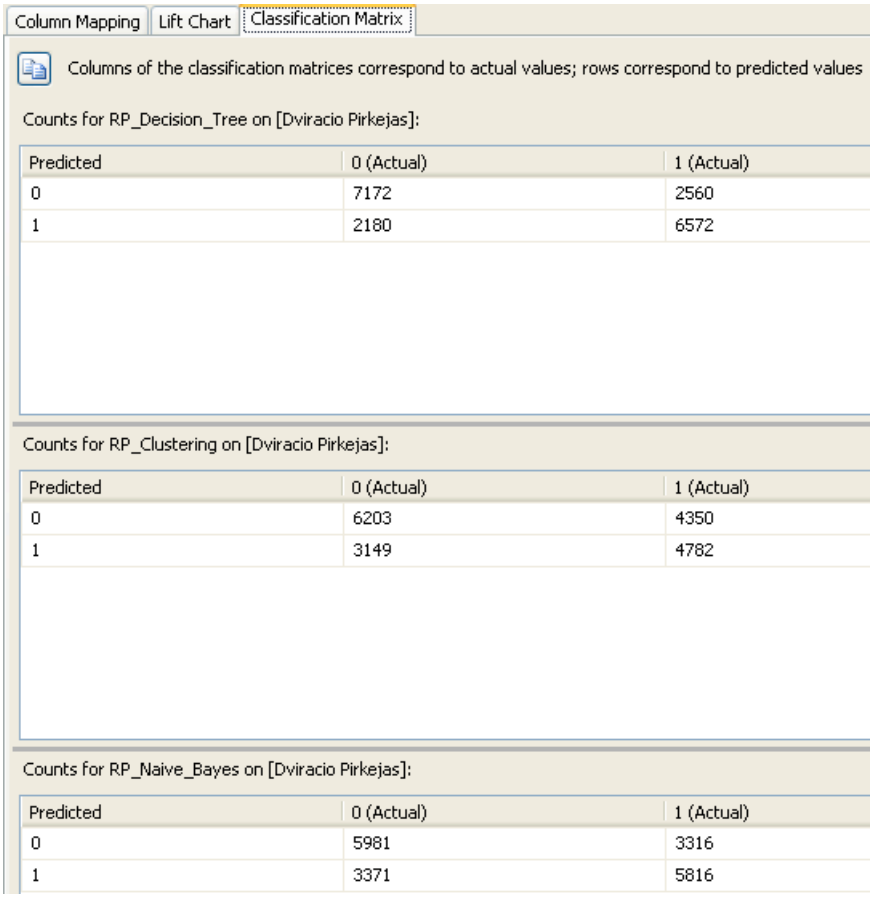
Grafike atvaizduojama po vieną liniją kiekvienam pasirinktam modeliui bei dvi papildomos linijos – ideali ir atsitiktinė. Kiekvieno linijos taško koordinatė parodo, kiek procentaliai būtų gaunama atsakymų, naudojant atitinkamą procentą modelio prognozuojamų rezultatų. Žemiau pateiktame paveiksle viršutinė linija atvaizduoja idealų modelį, kurio dėka 100% tikslo, būtų gaunama panaudojus 49% duomenų. Apatinė atsitiktinė linija visada yra 45 laipsnių linija grafike, nurodanti, kad atsitiktinai spėliojant kiekvieno atvejo rezultatus, gaunama 50% tikslo naudojant 50% duomenų. Kitos grafike esančios linijos atvaizduoja kiekvieno modelio rezultatus.

Kaip matyti paveiksle sprendimų medžio algoritmu sukurto modelio efektyvumas yra didžiausias – 90% tikslo būtų pasiekta panaudojus 70% duomenų. Paprasto atskyrimo atveju panaudojus 70% duomenų pasiekiama 81% tikslo, klasterių algoritmo atveju panaudojus 70% duomenų pasiekiama 77% tikslo.



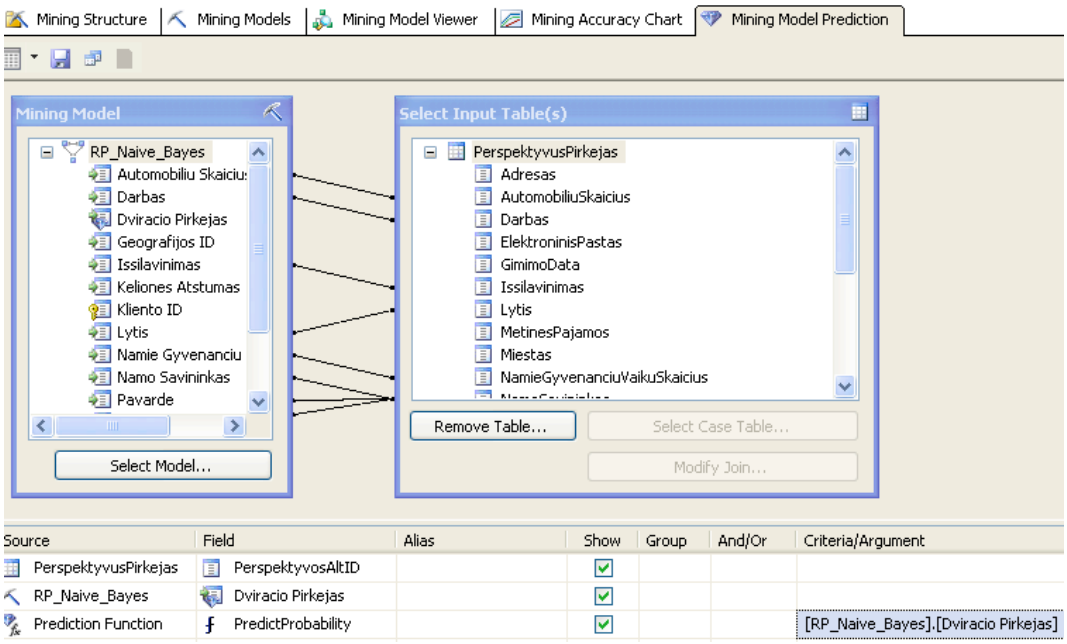
45 pav. Modelių palyginimo grafikas

Kitaip tariant turint 1000 klientų būtų gaunama 90% visų galimų atsakymų naudojantis modeliu, ir tiktai 50% atsakymų, jeigu pasiūlymai būtų siunčiami atsitiktinai. Tačiau tai nereiškia, kad 90% žmonių atsiųstų atsakymus. Anksčiau aprašyta, kad tik 49% žmonių atitinka tikslą, taigi naudojantis modeliu būtų gauta 90% atsakymų iš 49% žmonių, tai yra – 44,1%. Atsitiktinai spėliojant būtų gauta tiktai 22% atsakymų.



46 pav. Klasifikacijos matrica

Klasifikacijos matrica atvaizduoja kiek tiksliai kartų algoritmas teisingai nuspėja rezultatus, ir ką numato, kai rezultatas, būna klaidingas. Kaip matyti 46 paveiksle sprendimų medžio algoritmas nuspėdamas reikšmę 1 (klientas perkantis dviratį) tai atlieka teisingai 6572 atvejais, tačiau suklysta 2560 atvejais. Klasterių algoritmas spėdamas, kad nuspėjamo atributo reikšmė turėtų būti vienas, teisingai spėja 4782 atvejais, tačiau suklysta 4350 atvejais. Paprasto atskyrimo algoritmas prognozuodamas, kad klientas įsigys dviratį teisingai tai atlieka 5816 kartų, tačiau suklysta 3316 atvejų.



47 pav. Paprasto atskyrimo algoritmo spėjimų tikimybės modeliavimas

Stulpeliai PerspektyvosAltID, Dviracio Pirkėjas ir Expression identifikuoja potencialius klientus, nurodo, ar jie yra linkę įsigyti produkcijos, ir atvaizduoja tikimybę, kad spėjimas yra teisingas. Šie rezultatai gali būti naudojami nustatant, kuriem potencialiems klientams siųsti reklaminius pasiūlymus.

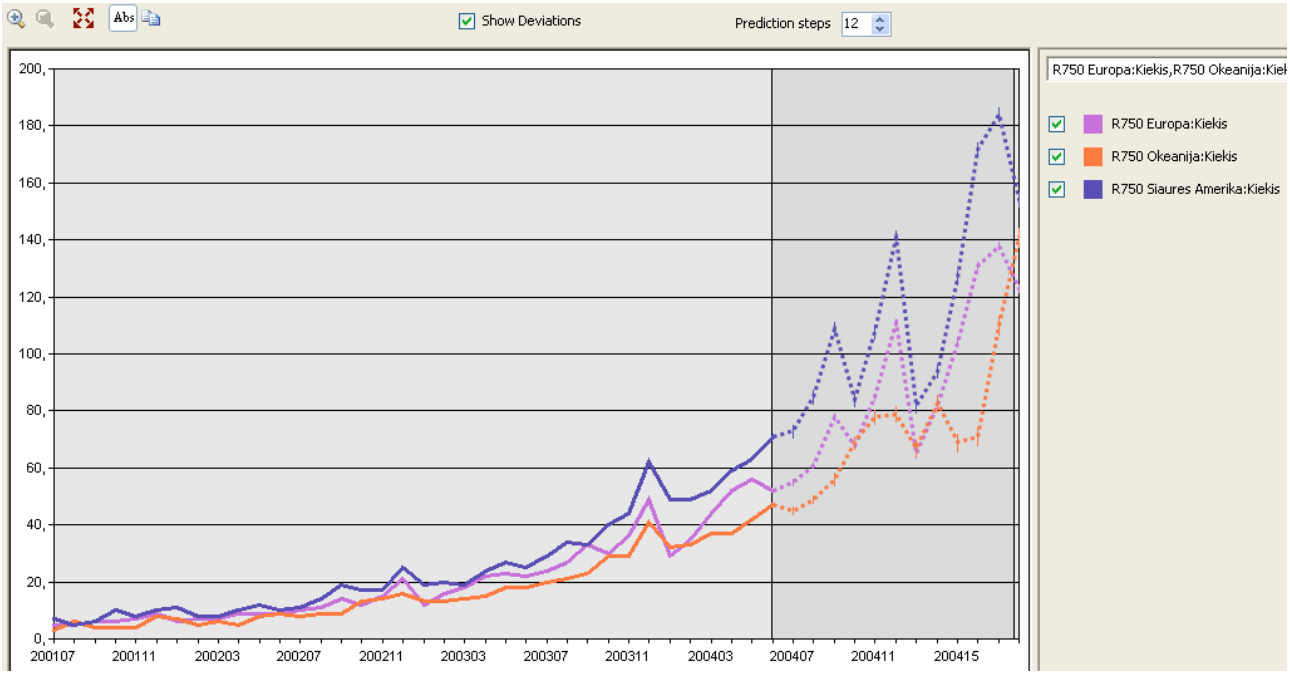
PerspektyvosAltID	Dviracio Pirkejas	Expression
827	1	0,67936574509...
833	0	0,57189182478...
844	0	0,65595479789...
832	0	0,79116239658...
53313373327	0	0,64798368694...
54107006788	1	0,53037905336...
53315894603	0	0,67067588230...
54037360548	1	0,78437229257...
15732240080	1	0,65241890221...
53469896316	0	0,55790933647...
15737550900	1	0,52517022035...
21596444800	0	0,57550732655...
75533120036	1	0,64516204208...

48 pav. Paprasto atskyrimo algoritmo spėjimų tikimybių rezultatų fragmentas

Iš gautų rezultatų galima pamatyti, kad klientas, kurio alternatyvus ID yra 827, pirks dviratį su tikimybe 67,94 %. Klientas, kurio alternatyvus ID 832 nepirks dviračio, ir šio spėjimo tikimybė yra 79,11%. Analogiškai galima pažiūrėti tikimybes kitiems klientams.

13.2. Produkcijos pardavimų prognozės

Įmonės pardavimų analitikui reikia numatyti individualaus dviračio modelio pardavimus ateinančiais metais skirtinguose regionuose. Tokiu būdu siekiama išsiaiškinti kokia produkcija bus paklausi atitinkamuose regionuose, ir pagal tai organizuoti dviračių ir jų priedų gamybą. Tiriami Europos, Šiaurės Amerikos ir Okeanijos regionų pardavimai. Prognozavimams atlikti panaudotas laiko serijų algoritmas, kuris kiekvienai duomenų rinkinyje egzistuojančiai serijai sukuria atskirą modelį. Kadangi duomenų rinkinyje pateikiami duomenys apie skirtingų regionų pardavimus, todėl algoritmas kiekvienam regionui sukuria atskirą laiko seriją.



49 pav. Produkcijos pardavimų prognozė

Dešinėje 49 paveikslo pusėje matomos grafike atvaizduojamos serijos. Varnele pažymėjus norimą seriją, ji atvaizduojama grafike, nužymėjus varnelę atitinkami duomenys nebėra rodomi. Grafike atvaizduoti tiek istoriniai, tiek nuspėjami ateities duomenys. Vertikali linija indikuoja ribą tarp istorinių ir ateities duomenų. Kaip matyti dviračio modelis R750 labiausiai perkamas Šiaurės Amerikoje: 2003 metų gruodžio mėnesį nupirkti 62 dviračiai, tuo tarpu Europoje tuo pačiu metu 49, Okeanijoje 41. Tyrinėjant istorinius duomenis matoma, kad produkcijos pardavimai nuolatos didėja, o gruodžio mėnesį pastebimas produkcijos pirkimo šuolis, kuris vėliau palaipsniui nuslopsta. Modelis prognozuoja, kad pardavimai ateityje augs, o gruodžio mėnesį vėlgi turėtų būti staigus pardavimų pagerėjimas: 2004 metų gruodžio

mėnesį Šiaurės Amerikoje 141, Europoje 111, tuo tarpu Okeanijoje 79 modelio R750 dviračių.

13.3. Pirkėjo krepšelio prognozavimas

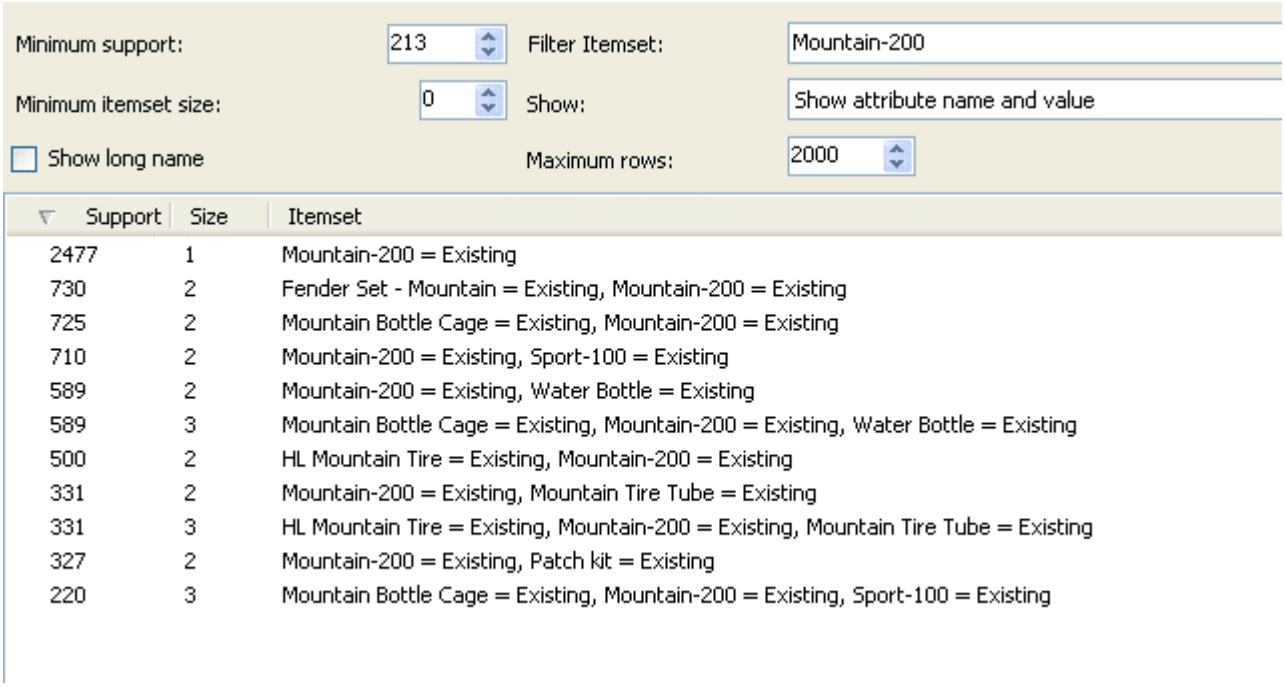
Kompanijos marketingo skyrius nori patobulinti internetinę produkcijos parduotuvę, kad galėtų sustiprinti kryžminių pardavimų (angl. *cross-selling*) strategiją.

Reikalingas modelis galintis nuspėti, kuriuos produktus klientai gali norėti įsigyti, atsižvelgiant į kitus produktus jau esančius internetinės parduotuvės kliento krepšelyje. Šie spėjimai taipogi padės marketingo skyriui paskirstyti produkciją, kuri gali būti parduodama kartu. Šiai užduočiai įgyvendinti panaudotas asociacijų algoritmas.

Elementų rinkinių sekcijoje atvaizduojamos trys svarbios informacijos dalys, kurias aptinka algoritmas:

- palaikymas – tai transakcijų skaičius, kuriame pasireiškia šis įrašų rinkinys
- dydis – tai yra elementų skaičius rinkinyje;
- aktuali elementų rinkinio sudėtis.

50 paveiksle pateikiama transakcijų informacija, kai buvo įsigytas dviračio modelis „Mountain-200“.



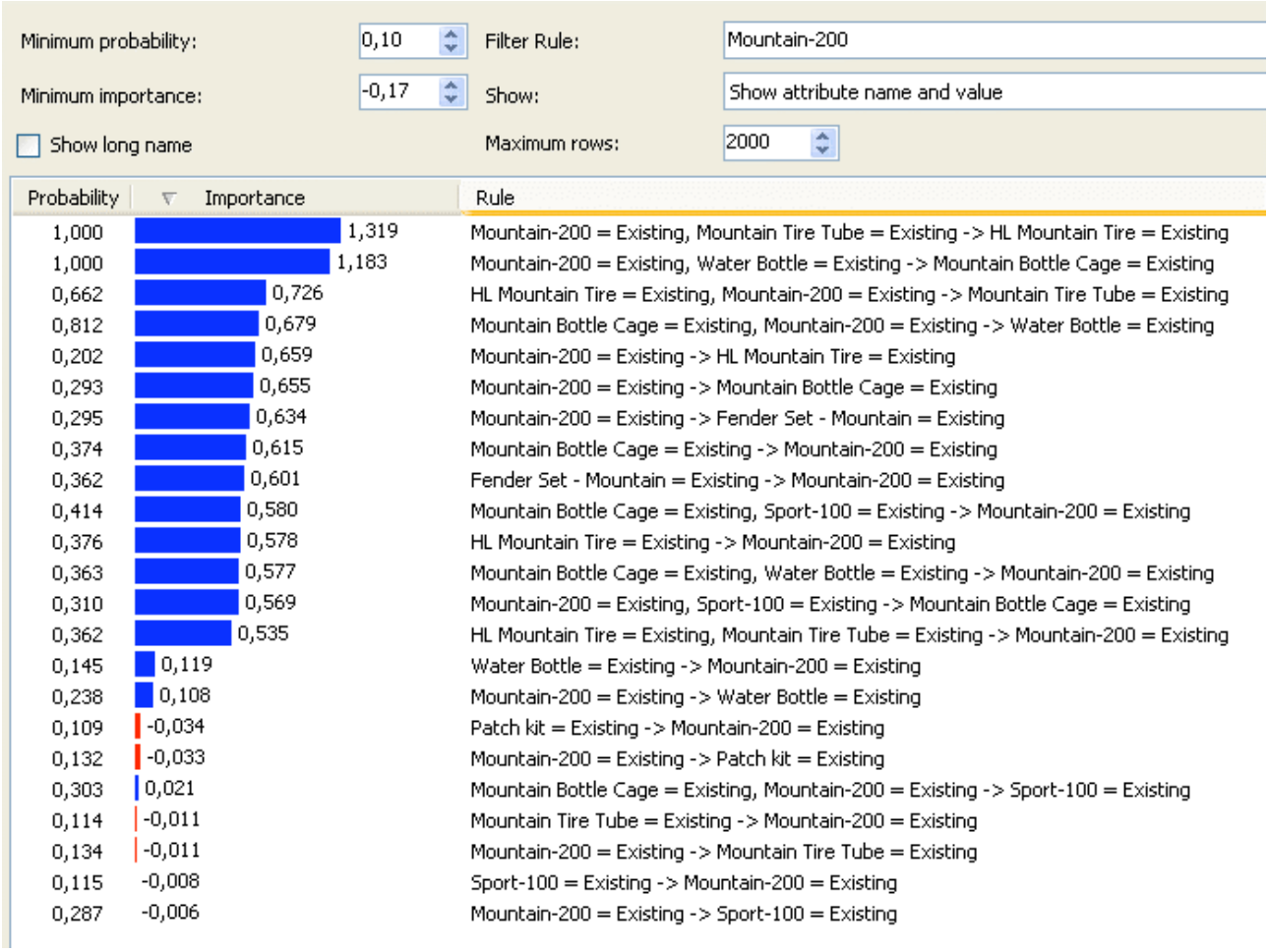
Support	Size	Itemset
2477	1	Mountain-200 = Existing
730	2	Fender Set - Mountain = Existing, Mountain-200 = Existing
725	2	Mountain Bottle Cage = Existing, Mountain-200 = Existing
710	2	Mountain-200 = Existing, Sport-100 = Existing
589	2	Mountain-200 = Existing, Water Bottle = Existing
589	3	Mountain Bottle Cage = Existing, Mountain-200 = Existing, Water Bottle = Existing
500	2	HL Mountain Tire = Existing, Mountain-200 = Existing
331	2	Mountain-200 = Existing, Mountain Tire Tube = Existing
331	3	HL Mountain Tire = Existing, Mountain-200 = Existing, Mountain Tire Tube = Existing
327	2	Mountain-200 = Existing, Patch kit = Existing
220	3	Mountain Bottle Cage = Existing, Mountain-200 = Existing, Sport-100 = Existing

50 pav. Dviračio „Mountain-200” pardavimo transakcijų informacija

Elementų rinkinys, kurio palaikymo reikšmė lygi 710 suteikia informaciją, kad iš visų dviračio „Mountain-200“ įsigijimo transakcijų (jų yra 2477), 710 kartų buvo taipogi įsigytas ir „Sport-100“ dviratis.

Taisyklių sekcijoje atvaizduojama tokia algoritmo aptinkama taisyklių informacija:

- Tikimybė: tikimybė, kad taisyklė pasitvirtins.
- Svarbumas: taisyklės naudingumo matmuo; kuo jis didesnis, tuo taisyklė geresnė. Vadovavimas vien tik tikimybėmis gali būti klaidingas. Pavyzdžiui, jeigu kiekvienoje transakcijoje yra elementas x, tada taisyklė y nuspėja, kad x tikimybė lygi vienetui, kas reiškia, kad x visada bus transakcijoje. Nors taisyklės tikslumas ir yra labai geras, tačiau jinai nesuteikia daug naudingos informacijos, kadangi kiekvienoje transakcijoje x egzistuoja, nepriklausomai nuo y.
- Taisyklė: taisyklės aprašymas.

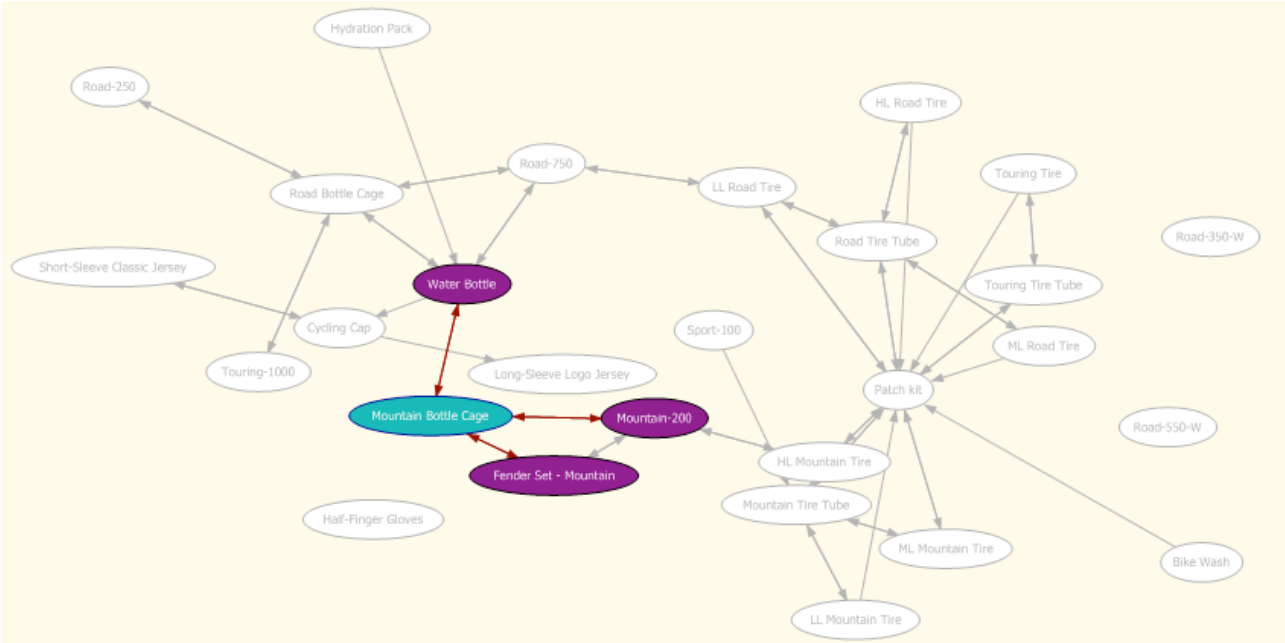


Probability	Importance	Rule
1,000	1,319	Mountain-200 = Existing, Mountain Tire Tube = Existing -> HL Mountain Tire = Existing
1,000	1,183	Mountain-200 = Existing, Water Bottle = Existing -> Mountain Bottle Cage = Existing
0,662	0,726	HL Mountain Tire = Existing, Mountain-200 = Existing -> Mountain Tire Tube = Existing
0,812	0,679	Mountain Bottle Cage = Existing, Mountain-200 = Existing -> Water Bottle = Existing
0,202	0,659	Mountain-200 = Existing -> HL Mountain Tire = Existing
0,293	0,655	Mountain-200 = Existing -> Mountain Bottle Cage = Existing
0,295	0,634	Mountain-200 = Existing -> Fender Set - Mountain = Existing
0,374	0,615	Mountain Bottle Cage = Existing -> Mountain-200 = Existing
0,362	0,601	Fender Set - Mountain = Existing -> Mountain-200 = Existing
0,414	0,580	Mountain Bottle Cage = Existing, Sport-100 = Existing -> Mountain-200 = Existing
0,376	0,578	HL Mountain Tire = Existing -> Mountain-200 = Existing
0,363	0,577	Mountain Bottle Cage = Existing, Water Bottle = Existing -> Mountain-200 = Existing
0,310	0,569	Mountain-200 = Existing, Sport-100 = Existing -> Mountain Bottle Cage = Existing
0,362	0,535	HL Mountain Tire = Existing, Mountain Tire Tube = Existing -> Mountain-200 = Existing
0,145	0,119	Water Bottle = Existing -> Mountain-200 = Existing
0,238	0,108	Mountain-200 = Existing -> Water Bottle = Existing
0,109	-0,034	Patch kit = Existing -> Mountain-200 = Existing
0,132	-0,033	Mountain-200 = Existing -> Patch kit = Existing
0,303	0,021	Mountain Bottle Cage = Existing, Mountain-200 = Existing -> Sport-100 = Existing
0,114	-0,011	Mountain Tire Tube = Existing -> Mountain-200 = Existing
0,134	-0,011	Mountain-200 = Existing -> Mountain Tire Tube = Existing
0,115	-0,008	Sport-100 = Existing -> Mountain-200 = Existing
0,287	-0,006	Mountain-200 = Existing -> Sport-100 = Existing

51 pav. Dviračio „Mountain-200” pardavimo taisyklės

Kiekviena taisyklė gali būti naudojama nuspėjant vieno produkto įsigijimą atsižvelgiant į įsigytus produktus. 51 paveiksle esanti antroji taisyklė suteikia informacijos, kad žmonės įsigiję Mountain-200 dviratį ir vandens buteliuką, nusipirks ir buteliuko laikiklį, šio fakto tikimybė lygi vienetui.

Priklausomybės tinklo sekcijoje atvaizduojama informacija, kaip sąveikauja skirtingi modelio elementai. Kiekvienas mazgas atspindi vieną elementą. Žymeklis susijęs su taisyklės tikimybe, nutempus jį žemyn paliekami tik stipriausi ryšiai.



52 pav. Asociacijų algoritmo priklausomybių tinklas

52 paveiksle matoma, kad buteliuko dėklas nuspėja ir yra nuspėjamas vandens buteliuko ir dviračio „Mountain-200“. Taigi tikėtina, kad šie elementai bus kartu vienoje transakcijoje. Kitaip tariant, jeigu klientas įsigyja dviratį, tai greičiausiai nusipirks ir buteliuką bei jo dėklą.

14. IŠVADOS

- Darbas realizuotas Microsoft Business Inteligence Development Studio įrankio pagalba, nes pastarasis pasižymi didelėmis intelektualios duomenų gavybos technologijų galimybėmis, turi aiškią dokumentaciją, ir yra suteikiamas nemokamai moksliniais tikslais.
- Darbo realizacijai panaudoti duomenys iš kompanijos „Microsoft“ teikiamos mokomosios duomenų bazės. Pateikiami realūs prekyba užsiimančios įmonės duomenys. Bandymas atsitiktiniu būdu sugeneruoti duomenis iškraipė gaunamus rezultatus ir nesuteikė analizei reikalingos informacijos.
- Intelektualios duomenų gavybos taikymas neatstoja profesionalaus analitiko, tačiau padeda jam optimaliau panaudoti turimus duomenis, tiksliau ir greičiau priimti sprendimus bei minimizuoti klaidos tikimybę.
- Darbo metu siekiant nustatyti potencialius klientus panaudoti skirtingi intelektualios duomenų gavybos algoritmai, kurių dėka nustatyti veiksniai turintys daugiausiai įtakos produkcijos pardavimams.
- Sprendimų medžio algoritmo pritaikymas atskleidė, kad didžiausią įtaką kliento apsisprendimui įsigyti dviratį turi metinės pajamos, asmens amžius, turimų automobilių skaičius ir namie gyvenančių vaikų skaičius.
- Klasterių algoritmo pritaikymo dėka duomenys sugrupuoti į grupes, kurios atskleidė, kad klientai praeityje įsigiję dviratį pasižymi šiomis charakteristikomis: jie gyvena Europoje (tikimybė 96,15%), atstumas nuo namų iki darbovietės yra apie du kilometrus (tikimybė 80,64%), turimų automobilių skaičius lygus nuliui (tikimybė 79,13), jie yra vedę (tikimybė 74,1%) ir neturi namie gyvenančių vaikų (tikimybė 69,48%).
- Realizavus paprasto atskyrimo algoritmą nustatyta, kad dviračių pirkėjai neturi namie gyvenančių vaikų (tikimybė 62,74%), yra vedę (tikimybė 50,93%), gyvena Šiaurės Amerikoje (tikimybė 46,27%), o atstumas nuo namų iki darbovietės yra nuo nulio iki dviejų kilometrų (tikimybė 38,74%).
- Palyginus visų reklaminio pašto siuntimo scenarijaus algoritmų veikimą nustatyta, kad sprendimų medžio algoritmu sukurto modelio efektyvumas yra didžiausias – 90% tikslo būtų pasiekta panaudojus 70% duomenų. Paprasto atskyrimo atveju panaudojus 70% duomenų pasiekiama 81% tikslo, klasterių algoritmo atveju panaudojus 70% duomenų pasiekiama 77% tikslo. Tai reiškia,

kad modelio dėka atrinkus žmones, kuriems reikėtų siųsti reklaminius pasiūlymus turėtų būti sulaukta atsakymų iš 44,1%. Atsitiktinai spėliojant būtų gauta tiktai 22% atsakymų.

- Realizavus laiko sekų algoritmą pastebėta, kad dviračio modelis R750 labiausiai perkamas Šiaurės Amerikoje: 2003 metų gruodžio mėnesį nupirkti 62 dviračiai, tuo tarpu Europoje tuo pačiu metu 49, Okeanijoje 41. Sukurtas modelis prognozuoja, kad pardavimai ateityje augs, o gruodžio mėnesį turėtų būti staigus pardavimų pagerėjimas: 2004 metų gruodžio mėnesį Šiaurės Amerikoje 141, Europoje 111, tuo tarpu Okeanijoje 79 modelio R750 dviračių. Pasitelkę šią informaciją įmonės vadybininkai gali užsakyti atitinkamą produkcijos kiekį.

15. LITERATŪROS SĄRAŠAS

1. Data mining concepts [žiūrėta 2007-12-05], prieiga internete: http://msdn2.microsoft.com/en-us/library/ms174949.aspx
2. Data mining overview [žiūrėta 2007-11-04], prieiga internete: http://www.microsoft.com/sql/technologies/dm/overview.mspx
3. Data Mining: What is Data Mining [žiūrėta 2007-11-03], prieiga internete: http://www.anderson.ucla.edu/faculty/jason.frand/teacher/technologies/palace/datamining.htm
4. Magic Quadrant for Business Intelligence Platforms, 2008 [žiūrėta 2008 09 27], prieiga internete: http://mediaproducts.gartner.com/reprints/microsoft/vol7/article3/article3.html
5. Microsoft Association Algorithm [žiūrėta 2008 02 03], prieiga internete: http://msdn.microsoft.com/en-us/library/ms174916.aspx
6. Microsoft Clustering Algorithm [žiūrėta 2008 01 30], prieiga internete: http://msdn.microsoft.com/en-us/library/ms174879.aspx
7. Microsoft Decision Trees Algorithm [žiūrėta 2008 03 07], prieiga internete: http://msdn.microsoft.com/en-us/library/ms175312.aspx
8. Microsoft Linear Regression Algorithm [žiūrėta 2008 03 09], prieiga internete: http://msdn.microsoft.com/en-us/library/ms174824.aspx
9. Microsoft Logistic Regression Algorithm [žiūrėta 2008 03 09], prieiga internete: http://msdn.microsoft.com/en-us/library/ms174828.aspx
10. Microsoft Naive Bayes Algorithm [žiūrėta 2008 01 30], prieiga internete: http://msdn.microsoft.com/en-us/library/ms174806.aspx
11. Microsoft Neural Network Algorithm (SSAS) [žiūrėta 2008 03 04], prieiga internete: http://msdn.microsoft.com/en-us/library/ms174941.aspx
12. Microsoft Sequence Clustering Algorithm [žiūrėta 2008 02 03], prieiga internete: http://msdn.microsoft.com/en-us/library/ms175462.aspx
13. Microsoft Time Series Algorithm [žiūrėta 2008 02 20], prieiga internete: http://msdn.microsoft.com/en-us/library/ms174923.aspx

14. SQL Server System Requirements [žiūrėta 2008 03 16], prieiga internete: http://www.microsoft.com/sql/editions/enterprise/sysreqs.mspx

15. ZhaoHui Tang, Jamie MacLennan Data Mining with SQL Server 2005. Inianapolis: Wiley Publishing, 2005.