UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

DETEKCIA NEŠTANDARDNÝCH POISTNÝCH UDALOSTÍ POMOCOU METÓD STROJOVÉHO UČENIA

DIPLOMOVÁ PRÁCA

2018 Bc. Ján MURÍN

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

DETEKCIA NEŠTANDARDNÝCH POISTNÝCH UDALOSTÍ POMOCOU METÓD STROJOVÉHO UČENIA

DIPLOMOVÁ PRÁCA

Študijný program:	Ekonomicko-finančná matematika a modelovanie
Študijný odbor:	1114 Aplikovaná matematika
Školiace pracovisko:	Katedra aplikovanej matematiky a štatistiky
Vedúci práce:	doc. Mgr. Radoslav Harman, PhD.

Bratislava 2018 **Bc. Ján MURÍN**

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Bc. Ján Murín
Študijný program: ekonomicko-finančná matematika a modelovanie (Jednoodborové štúdium, magisterský II. st., denná forma)
Študijný odbor: aplikovaná matematika
Typ záverečnej práce: diplomová
Jazyk záverečnej práce: slovenský
Sekundárny jazyk: anglický

Názov: Detekcia neštandardných poistných udalostí pomocou metód strojového učenia
Detection of non-standard insurance claims using machine learning techniques

Anotácia: Každoročne na Slovensku nastanú tisícky poistných udalostí, za ktoré má poisťovňa Generali, a.s. povinnosť vyplatiť odškodné. Predstava ľahko zarobených peňazí priťahuje množstvo špekulantov, ktorí sa snažia skutočnosti ohľadom poistných udalostí sfalšovať. Poisťovňa sa snaží proti takýmto podvodníkom bojovať, no kvôli kvantite poistných udalostí nie je schopná každú manuálne skontrolovať. Množstvo kvalitných dát, ktorými poisťovňa disponuje, dáva priestor na využitie teórie strojového učenia pre zoradenie udalostí podľa rizikovosti.

Cieľ: Cieľom práce je navrhnúť systém automatickej detekcie neštandardných poistných udalostí, ktoré následne prejdú dodatočnou manuálnou verifikáciou zo strany poisťovne. Zámerom je využiť poznatky z teórie strojového učenia s dôrazom na špecifické požiadavky poistného prostredia a budúcu použiteľnosť systému v praxi.

Vedúci: doc. Mgr. Radoslav Harman, PhD.
Katedra: FMFI.KAMŠ - Katedra aplikovanej matematiky a štatistiky
Vedúci katedry: prof. RNDr. Daniel Ševčovič, DrSc.

Dátum zadania: 26.01.2017

Dátum schválenia: 27.01.2017 prof. RNDr. Daniel Ševčovič, DrSc.
garant študijného programu

.....
študent

.....
vedúci práce

Poďakovanie Chcel by som sa poďakovať spoločnosti Generali Poisťovňa, a.s. za poskytnutú príležitosť a vedúcemu práce doc. Mgr. Radoslavovi Harmanovi, PhD. za výbornú spoluprácu, dôveru a veľmi cenné rady. Ďakujem aj svojej rodine s priateľkou za trpezlivosť a veľkú podporu.

Abstrakt v štátnom jazyku

MURÍN, Ján: Detekcia neštandardných poistných udalostí pomocou metód strojového učenia [Diplomová práca], Univerzita Komenského v Bratislave, Fakulta matematiky, fyziky a informatiky, Katedra aplikovanej matematiky a štatistiky; školiteľ: doc. Mgr. Radoslav Harman, PhD., Bratislava, 2018, 78 s.

V diplomovej práci spolupracujeme s poisťovňou Generali Poisťovňa, a. s. a zaoberáme sa poistnými podvodmi v sektore neživotného poistenia. Poisťovňa na účely filtrovania rizikových poistných udalostí, ktoré by mohli byť podvodmi, používa systém založený na expertných skúsenostiach jej zamestnancov. Cieľom diplomovej práce je tento systém vylepšiť pomocou strojového učenia. Aby sme to dosiahli, v prvom rade sa venujeme samotnej problematike poistných podvodov, analyzujeme pôvodný systém a špeciálne požiadavky prostredia poisťovne. Po identifikovaní priestoru na aplikovanie strojového učenia diskutujeme o možnosti použitia zaužívaných metód, no okrem regresných stromov žiadna plne v tomto špecifickom prípade nevyhovuje. Z týchto dôvodov vytvárame ad-hoc metódu pomocou teórie Bayesovho klasifikátora a logistickej regresie šitú na mieru potrebám poisťovne. Na odhaľovanie opakujúcich sa podvodov dopĺňame regresné stromy a ad-hoc metódu zhlukovaním s nami vytvoreným postupom na priradenie rizikovosti jednotlivým skupinám. Jednotlivé metódy testujeme a porovnávame s pôvodným systémom. Na základe výsledkov testovania navrhujeme podobu nového systému, pričom dávame dôraz na to, aby bol systém ľahko implementovateľný a parametre v ňom dobre interpretovateľné. Nový systém vykazuje výrazné zvýšenie počtu skutočných podvodov medzi predikovanými najrizikovejšími poistnými udalosťami.

Kľúčové slová: poistné podvody, strojové učenie, rozhodovacie stromy, zhlukovanie, Bayesov klasifikátor

Abstract

MURÍN, Ján: Detection of non-standard insurance claims using machine learning techniques [Master's Thesis], Comenius University in Bratislava, Faculty of Mathematics, Physics and Informatics, Department of Applied Mathematics and Statistics; Supervisor: doc. Mgr. Radoslav Harman, PhD., Bratislava, 2018, 78 p.

In this master's thesis we cooperate with Generali Poisťovňa, a. s. insurance company to detect insurance fraud in the general insurance. The company uses a system based on the expert experience of its employees for the purpose of filtering risk-related insured events that could be fraud. The aim of this master's thesis is to improve the current system of fraud detection using machine learning. To do this, we primarily address the issue of insurance fraud, analyze the original system and special requirements of the insurance company's environment. After identifying the possibilities to apply machine learning, we discuss the possibility of using commonly used methods, but only regression trees fit the strict criteria. For these reasons, we create an ad-hoc method tailored to the needs of an insurance company using Bayesian classifier and logistic regression theory. In order to detect recurring frauds, we supplement the regression trees and the ad-hoc method with a clustering method using our procedure to approximate the risk of each group. We test and compare the different methods with the original system. Based on the test results, we propose a new system, emphasizing that the system is easy to implement and its parameters well interpretable. The new system shows a significant increase in the number of actual frauds between the predicted most risky insurance events.

Keywords: insurance fraud, machine learning, decision trees, clustering, Bayes classifier

Obsah

Zoznam obrázkov

Zoznam tabuliek

Úvod

Poisťovníctvo je dôležitou súčasťou finančného trhu. Vďaka širokej škále produktov ovplyvňuje takmer každú fyzickú a právnickú osobu vo vyspelých krajinách. Dnešný aktívny životný štýl ľudí spôsobuje, že každoročne na Slovensku nastanú tisícky poistných udalostí, za ktoré má povinnosť poisťovňa podľa poistnej zmluvy vyplatiť odškodné. Predstava ľahko zarobených peňazí priťahuje množstvo špekulantov. Škody, ktoré sa v skutočnosti nestali, prípadne sa stali v nižšej miere, ako je popísane v poistnej udalosti, sú relatívne bežným úkazom. Každá škodová udalosť je osobne preverovaná likvidátorom a podozrivé udalosti sú poslané na hlbšie preskúmanie. Nakoľko je poistných udalostí tak veľa, nie je v silách poisťovne každú podozrivú udalosť hlbšie preveriť. Čas zamestnancov vytvára veľké náklady, a keďže cieľom poisťovne je v prvom rade zisk, vytvára si rôzne poloautomatické a automatické systémy na zefektívnenie tohto procesu.

Strojové učenie je štatistická oblasť, ktorá si v dnešnej dobe rýchlo hľadá cestu do rôznych odvetví vďaka rastúcemu výkonu počítačov a zvyšujúcemu sa množstvu kvalitných dát. Poisťovne disponujú veľkými databázami o klientoch a poistných udalostiach, preto poskytujú ideálnu príležitosť na využitie metód strojového učenia. Na Slovensku zatiaľ nie sú tieto metódy v praxi zaužívané a neexistujú štandardizované postupy na ich aplikovanie pre odhaľovanie podvodov.

V tejto práci spolupracujeme s poisťovňou Generali Poisťovňa, a. s., ktorej základom odhaľovania podvodov sú expertné skúsenosti na danú oblasť poistenia orientovaných zamestnancov. Tento systém bol dlho postačujúcim, no v posledných rokoch vykazuje znižujúcu sa úspešnosť odhaľovania podvodov. Cieľom práce je aplikovať teóriu strojového učenia s veľkým dôrazom na špecifické potreby poisťovne a tým zlepšiť dnešný systém. Pomocou týchto metód využijeme informácie skryté v historických dátach na vyselektovanie podozrivých udalostí, ktoré následne môžu prejsť na manuálnu kontrolu.

Práca je rozdelená na tri kapitoly. Prvá slúži ako krátky úvod do poisťovníctva, stručne popisuje systém, ktorý v súčasnosti používa poisťovňa Generali Poisťovňa, a. s. na odhaľovanie podvodov, analyzuje jeho špecifické potreby a definuje nami zvolenú metodiku k jeho zlepšeniu. Druhá kapitola sumarizuje potrebné všeobecne známe poznatky z teórie a opisuje našu metódu vytvorenú špecificky pre potreby Generali

Poisťovňa, a. s.. V poslednej kapitole prezentujeme použitie jednotlivých metód a porovnávame ich úspešnosť odhaľovania podvodov s dnešným systémom.

1 Úvod do problematiky

1.1 Riziko a poistenie

Už len samotná existencia človeka alebo podnikateľského subjektu so sebou prináša veľké množstvo udalostí, ktorých výskyt ani priebeh sa nedá predvídať. Prítomnosť náhodných udalostí sa môže odzrkadliť na stratenom majetku alebo poškodenom zdraví. Podľa [1] všetky takéto náhodné a nechcené odchýlky od stanovených cieľov nazývame jednotným termínom - riziko. Podobne v [12] je riziko definované ako pojem, ktorý zjednoťuje:

1. neinformovanosť,
2. variabilitu výsledkov,
3. nebezpečenstvo odchýlok od cieľa,
4. nebezpečenstvo zlých rozhodnutí,
5. nebezpečenstvo nepriaznivých dopadov rozhodnutí.

Následky rizík majú veľkú variabilitu - od malých strát až po bankrot, od zdravotných nepríjemností až po smrť. Takéto nečakané straty sú samozrejme pre subjekt vysoko nežiaduce, no ich výskyt je niekedy až prekvapivo bežný. Modelových príkladov náhodných udalostí, ktoré v sebe zahŕňajú určité riziko, je mnoho:

- rodine vzniknú nečakané náklady na vymaľovanie bytu a odstránenie plesne po tom, ako ich vytopí sused,
- remeselný živnostník nie je schopný vykonávať činnosť a stratí na mesiac príjem kvôli zlomenej ruke po pošmyknutí sa na ľadovom chodníku,
- poľnohospodárovi zničí veľmi nepriaznivé počasie časť úrody, čím stratí zisk z jej predaja a prestane byť schopný splácať pôžičku od banky,
- malej prepravnej firme sa znehodnotí prevážaný náklad, znefunkční kamión a zamestnanec utrpí úraz po dopravnej nehode.

Prirodzeným správaním každého subjektu je hľadanie nástrojov na čo najväčšie zníženie svojho ohrozenia. Prvým spôsobom je zmena správania subjektu, ktorá má za následok zníženie pravdepodobnosti nastania rizika. Jednoduchým príkladom je zapnutie denného svietenia pre zníženie rizika autonehody. Druhým spôsobom je minimalizácia dopadov rizík, najčastejšie realizovaná pomocou poistenia, ktorým subjekt prenáša časť rizík na poisťovňu. Poisťovňa sa tým zaviaže, že v prípade nastania poistnej udalosti zahŕňajúcej určité riziko vyplatí poistencovi poistné plnenie. Jedná sa teda o elimináciu negatívnych finančných následkov. Vráťme sa k príkladu o poľnohospodárovi, ktorý investoval pri pestovaní úrody množstvo prostriedkov na vybavenie, pracovnú silu a podobne. Strata úrody mu môže zničiť celú podnikateľskú činnosť a toto riziko nemôže nijakým spôsobom úplne eliminovať. Po uzavretí poistnej zmluvy môžu nastať dva prípady - buď stratí úrodu a poisťovňa mu vyplatí poistné plnenie alebo bude počasie celý rok priaznivé a jemu sa len zvýšia náklady o výšku poistného.

Druhá strana mince je podrobne rozobratá v [1] - poisťovňa ako podnikateľský subjekt má za hlavný cieľ zisk. Výkon jej podnikateľskej činnosti podmieňuje nakladanie so zdrojmi poistenca pri krytí rizík ostatných subjektov. Svojou činnosťou v podstate rozkladá dopady rizík jednotlivca na celú skupinu. Zisk dosahuje nastavením výšky poistného pre poistencov za pomoci riadenia vlastných rizík. Za významný faktor najviac podmieňujúci solventnosť poisťovne označuje [1] práve správne nastavenie výšky poistného, čo dosahuje čo najpresnejším odhadom budúcich nákladov na poistné plnenia. Tu do hry vstupujú modely určenia rizík a ich závažnosti, ktoré si poisťovne vo väčšine prípadov samy interne vytvárajú.

Z matematického hľadiska sa podľa [2] ľudia rozhodujú o uzavretí poistenia porovnaním strednej hodnoty očakávaných strát. Pridelia svojmu bohatstvu w úžitok $u(w)$, kde u je ich neklesajúca úžitková funkcia. Následne z rôznych náhodných strát X a Y vyberú takú, ktorá má očakávanú stratu na úžitku najvyššiu, teda porovná $E[u(w-X)]$ a $E[u(w-Y)]$. Finálne rozhodnutie pre poistenie rizika spôsobujúceho danú najvyššiu stratu závisí od ponúkanej výšky poistného. Ak poistenec premýšľa nad poistením náhodnej straty X, podľa [2] vyrieši rovnicu rovnováhy:

$$E[u(w-X)] = u(w-P), \tag{1}$$

kde výsledné P^+ je výška celkového poistného, pri ktorom je poistenec indiferentný

voči poisteniu. Ak je poistné ponúkané poisťovňou nižšie ako P^+, rozhodne sa riziko si poistiť. Samozrejme takéto zmýšľanie je len hypotetické. V skutočnosti poistenci vo väčšine prípadov žiadne rovnice neriešia, no takýto postup je dobrým modelom pre racionálne rozhodovanie. Rovnicu (1) si pre vlastnú úžitkovú funkciu a inú hodnotu nákladov vyplývajúcich z náhodnej straty X vyrieši aj poisťovňa. Jej výsledkom je P^-, ktoré je minimum výšky celkového poistného, pri ktorom by poisťovňa riziko klientovi poistila. Kniha [2] túto časť uzatvára tým, že ak je výsledné ponúkané poistné menšie ako P^+ a väčšie ako P^-, obe strany sa rozhodnú uzatvoriť poistenie, lebo poisťovni aj poistencovi sa zvýši očakávaný konečný úžitok. Zo vzťahu medzi poistencom a poisťovňou vyplýva aj poistno-právna definícia rizika ako nebezpečenstva, ktoré môže viesť k vzniku poistnej udalosti (*§2 odst. 16 Zákona č. 8/2008 o poisťovníctve*). Aby riziko mohlo byť poistiteľné, potrebuje podľa [1] spĺňať nasledovné podmienky:

- identifikovateľnosť - jednoznačné určené príčiny výskytu udalosti, ktorá má za následok straty zahrnuté v poistnom krytí,
- únosnosť - riziko nesmie mať za následok straty, ktoré by poisťovňa nebola schopná vyrovnať,
- vyčísliteľnosť - finančná strata ako následok udalosti musí byť jasne vypočítateľná už pred samotným nastaním poistnej udalosti,
- náhodnosť - výskyt udalostí zahŕňajúcich dané riziko musí byť úplne náhodný, teda nesmie byť jasné či, alebo kedy udalosť nastane.

Poisťovne bežne ponúkajú široké spektrum poisťovacích produktov rozdelených do dvoch hlavných kategórií - životné a neživotné poistenie. Pre každý produkt je vymedzený jasný súbor kritérií, ktoré musí potencionálny klient pre získanie poistenia splniť. Takýto postup je ďalším spôsobom na obranu solventnosti poisťovne. Životné poistenie charakterizuje [1] ako rezervotvorné poistenie, ktorého obsahom je riziko dožitia alebo úmrtia. Na rozdiel od neživotného poistenia sa poistné plnenie vypláca v každom prípade, keďže jeho úlohou je hlavne finančná pomoc pozostalým, prípadne sporenie pre obdobie staroby. Životné poistenie sa delí na nasledovné odvetvia (*zákon č. 8/2008 Z. z. o poisťovníctve*):

1. poistenie pre prípad smrti, pre prípad dožitia alebo kombinácie týchto dvoch rizík,
2. poistenie vena alebo prostriedkov na výživu detí,
3. poistenie spojené s kapitalizačnými zmluvami,
4. poistenie podľa bodov 1 a 3 spojené s investičným fondom,
5. dôchodkové poistenie,
6. poistenie pre prípad úrazu alebo choroby, ak je pripoistením niektorého poistného odvetvia v bodoch 1 až 5.

Podobne [1] opisuje delenie neživotného poistenia. Poisťovne ponúkajú množstvo typov produktov stále v nových formách, pričom často kombinujú viacero typov podľa potrieb klienta. Neživotné poistenie sa delí na odvetvia (*zákon č. 8/2008 Z. z. o poisťovníctve*):

1. poistenie úrazov a chorôb,
2. poistenie motorových vozidiel,
3. námorné a dopravné poistenie,
4. letecké poistenie,
5. poistenie požiarov a iných majetkových škôd,
6. poistenie zodpovednosti za škodu,
7. poistenie úveru a kaucie,
8. všeobecné neživotné poistenie.

V tejto práci sa zaoberáme hlavne poistnými podvodmi v neživotnom poistení z oblasti poistenia motorových vozidiel a všeobecného neživotného poistenia.

1.1.1 Poistné podvody

Ak nerátame podvody na daniach, poistné podvody sú na základe [13] najčastejšou formou podvodu na svete. Táto skutočnosť plynie hlavne z podstaty podnikania v poisťovníctve, kde poisťovne generujú veľký a pravidelný tok peňazí (*angl. cash flow*) z platieb poistného. Podľa [1] bol počet podvodov v roku 2010 o 10-15% vyšší ako v predkrízovom období. Podvody v oblasti poistenia motorových vozidiel v roku 2011 tvorili až 80% odhalených podvodov v poisťovni Generali Poisťovňa, a. s..

V zmluve medzi poistencom a poisťovňou sú jasne uvedené podmienky pre vyplatenie poistného plnenia v prípade poistnej udalosti zahŕňajúcej poistené riziko. Poistný podvod vzniká vtedy, keď poistenec získa poistné plnenie bez splnenia poistných podmienok. Podvodník sa snaží o predstieranie uskutočnenia poistnej udalosti alebo o umelé navýšenie škôd namiesto toho, aby ostal poistený pre prípad náhodnej udalosti. Trestný zákon (*prvý odsek § 223 zákona číslo 300/2005*) vymedzuje trestný čin poistného podvodu ako vylákanie poistného plnenia uvedením do omylu v otázke splnenia podmienok na jeho poskytnutie a tým spôsobenie škody. Trest za trestný čin poistného podvodu sa líši v závislosti od veľkosti škôd - od 1 až 5 rokov za malé škody, až po 10 až 15 rokov za škody veľkého rozsahu. Kniha [13] rozlišuje 2 základné typy podvodov podľa závažnosti - ťažké (*angl. hard frauds*) a ľahké (*angl. soft frauds*). Podvody sú podľa nej ťažké, ak sa poistná udalosť uskutoční cielene, tým pádom nenáhodne. Poistenec si v týchto prípadoch môže napríklad úmyselne ublížiť na zdraví alebo podpáliť dom. Ľahšie podvody majú zveličené veľkosti škôd - bežným prípadom je udávanie vyššej výbavy auta, ako je v skutočnosti, čím sa umelo navýši jeho cena.

Poistné podvody majú mnoho foriem. Podrobne ich opisuje [1]. Pri životnom poistení opisuje [1] hlavne prípady úmyselného seba poškodzovania, účelové poistenie osôb so zdravotným stavom, ktorý to vylučuje, predkladanie falošných dokladov pri úrazoch alebo klamanie o dátume poistnej udalosti. Poistné podvody sú podľa [1] oveľa častejšie a zároveň vyskytujúce sa v rôznorodejších formách v neživotnom poistení. Špekulanti klamú hlavne pri dopravných nehodách, pričom ich často úplne zinscenujú, zatajujú skutočné informácie o vozidle, poisťujú ukradnuté vozidlá alebo spolupracujú so zamestnancom poisťovne. Pri majetkovom poistení je bežné úmyselné poškodzovanie nehnuteľnosti, fiktívne faktúry za opravy, viacnásobné nárokovanie na rovnaké škody

a podobne. Podobné praktiky sa uplatňujú aj v cestovnom poistení, živelnom poistení a mnoho ďalších.

Každý neodhalený podvod stojí poisťovňu finančné prostriedky a zároveň potenciálne podnecuje k jeho budúcemu opakovaniu. Podvody jednotlivcov vedia poisťovne odhaliť v oveľa vyššej miere ako podvody organizovaných skupín, ktoré sú mnohokrát sofistikovanejšie. Poisťovne na boj proti podvodníkom zamestnávajú celé oddelenia špecializovaných zamestnancov. Často pri tom spolupracujú medzi sebou pomocou výmeny informácií. Zamestnávanie toľkých ľudí ale stojí nemalé finančné prostriedky, preto je pre poisťovňu oveľa výhodnejšie namiesto podrobného preskúmania každej poistnej udalosti skúmať len určitú vytypovanú časť z nich. Kniha [1] opisuje viaceré metódy, ktoré si poisťovne vyvinuli na odhaľovanie podvodov:

1. značkovanie - jednoduchá no nie veľmi účinná metóda, pri ktorej pracovníci úmyselne vyhľadávajú udalosti na základe ich expertných skúseností,

2. vyhľadávanie v centrálnej databáze - správcovstvo slovenskej databázy zabezpečuje súkromná spoločnosť podporovaná štátom - Slovenská asociácia poisťovní (*skr. SLASPO*),

3. identifikácia odchýlok (*angl. outliers*) - metóda formou automatických reportov identifikuje poistné udalosti, ktoré sa vo vybraných štatistických ukazovateľoch vymykajú z normálu,

4. prediktívne modelovanie - nahrádza manuálne značkovanie, pomocou softvérových nástrojov a historických záznamov prideľuje mieru pravdepodobnosti toho, že poistná udalosť je podvodom,

5. link analýza - dôležitá metóda pre odhaľovanie organizovaných skupín, jej výstupom je *link graf*, v ktorom hrúbka čiary závisí od početnosti výskytu danej kombinácie faktorov.

Reálne používaných metód je na globálnom trhu samozrejme oveľa viac. Podľa [1] sa v dnešnej dobe využívajú už aj veľmi špecializované metódy ako napríklad detekcia stresu v hlase.

1.1.2 Pôvodný systém odhaľovania podvodov

Generali Poisťovňa, a. s. si rokmi vyvinula poloautomatický systém na odhaľovanie podvodov v neživotnom poistení. Systém najprv zoradil poistné udalosti podľa výšky podozrenia, že sú podvodmi a následne udalosti prešli od najrizikovejších na manuálnu kontrolu. Základom boli celé oddelenia zamestnancov, ktorí sa špecializovali na určité oblasti poistných produktov a podrobne v nich sledovali trendy v poistných podvodoch. Celý systém bol teda založený na expertíze a skúsenostiach jej zamestnancov a neobsahoval žiadne zložitejšie štatistické metódy. Z metód spomenutých v predošlej kapitole bolo prediktívne modelovanie jednou z úplne nevyužitých. Neživotné poistenie mala poisťovňa rozdelené do takzvaných scenárov, ktoré sú akýmisi skupinami podobných produktov. Pre každý scenár tím expertov volil a pravidelne revidoval určitú sadu $i = 1, \ldots, t$ identifikátorov, ktoré zohľadňovali pri predikcii, či bola udalosť podozrivá. Hodnoty každého identifikátora i boli rozdelené do $j = 1, \ldots, m_i$ disjunktných skupín („hodnôt") a každej skupine bola pridelená kladná celočíselná váha $w_{i,j}$. Váhy boli, rovnako ako identifikátory, určované a revidované odborníkmi na základe expertného odhadu na pravidelných stretnutiach. V Tabuľke 1 je uvedená štruktúra niektorých identifikátorov a váh pre „všeobecný scenár 1", ktorému sa v tejto práci venujeme. Konkrétne hodnoty identifikátorov a váh z bezpečnostných dôvodov neuvádzame.

Rizikovosť k-tej poistnej udalosti bola následne určená jednoduchou rovnicou:

$$\beta_k = \sum_{i=1}^{t} \sum_{j=1}^{m_i} I_{i,j}^k w_{i,j}, \tag{2}$$

kde $I_{i,j}^k = 1$ ak pre k-tu poistnú udalosť identifikátor i nadobúda hodnotu j, inak $I_{i,j}^k = 0$. Takto zadefinovaná rizikovosť β nepredstavuje pravdepodobnosť, ale má zmysel len ak rizikovosť určitej poistnej udalosti porovnáme s rizikovosťou ostatných udalostí. Z toho vyplýva, že samostatne nemá výpovednú hodnotu, no slúžila na zoradenie udalostí a tým určovala, ktoré udalosti boli prednostne vybraté na manuálnu kontrolu. Čím vyššia β_k, tým bola k-ta udalosť skôr vybratá.

Samotnú manuálnu kontrolu následne vykonával zamestnanec, ktorý sa podrobne pozrel na skutočnosti uvedené v poistnej udalosti a priložené fotografie. K poistencovi mohli vyslať technika, ktorý overil, či sa škody stali v uvedenom rozsahu a sám rozhodol o ďalšom postupe. Celý tento proces je obvykle časovo náročný a tým aj z fi-

Identifikátor	Hodnoty	Váha
Počet udalostí vodiča	hodnota 1	$w_{1,1}$
	hodnota 2	$w_{1,2}$
	hodnota 3	$w_{1,3}$
	hodnota 4	$w_{1,4}$
	hodnota 5	$w_{1,5}$
Záujmové VIN	hodnota 1	$w_{2,1}$
	hodnota 2	$w_{2,2}$
	hodnota 3	$w_{2,3}$
	hodnota 4	$w_{2,4}$
	hodnota 5	$w_{2,5}$
	hodnota 6	$w_{2,6}$
Priemerná škoda posledných 2 poistných udalostí	hodnota 1	$w_{3,1}$
	hodnota 2	$w_{3,2}$
Dlh na poistnom	hodnota 1	$w_{4,1}$
	hodnota 2	$w_{4,2}$

Tabuľka 1: Ilustračný príklad štruktúry identifikátorov pre scenár 1

nančného hľadiska nákladný, preto bolo dôležité, aby na manuálnu kontrolu prichádzalo čo najviac skutočných podvodov.

Pôvodná verzia tohto systému vykazovala rezervy. Podiel vyselektovaných najrizikovejších udalostí, ktoré sa naozaj ukázali ako podvodné dlhodobo klesal. Zamestnanci sa naučili, že veľké množstvo podvodov sa nachádza medzi udalosťami s najnižšou rizikovosťou β, preto ich často na manuálnu kontrolu uprednostňovali. Táto pôvodná metóda už nie je v dnešnej dobe čiastočne aj vďaka našej práci používaná.

1.2 Metodika použitia metód strojového učenia

1.2.1 Dáta

Na účely tejto práce nám Generali Poisťovňa, a. s. poskytla dáta zo scenárov neživotného poistenia. V prvom rade sme dostali tabuľky pre jednotlivé scenáre, ktoré určujú

zoznam identifikátorov a ich rozdelenie na jednotlivé hodnoty. Ku každej hodnote sme dostali pridelenú najaktuálnejšiu váhu používanú v pôvodnom systéme. Ďalej sme mali k dispozícii údaje o poistných udalostiach uskutočnených od januára 2016 po jún 2017. Každá poistná udalosť mala jednoznačne pridelené údaje pre každý identifikátor. Použitím tabuliek a hodnôt pre každý identifikátor sme vedeli poistným udalostiam prideliť rizikovosť β a podľa nej udalosti zoradiť. Dát bol pre každý scenár relatívny dostatok, všetky polia boli riadne vyplnené a nenachádzali sa v nich takmer žiadne chyby.

Tieto dáta ale neboli postačujúce na použitie klasifikačných metód, ktoré využívajú učenie s učiteľom (*angl. supervised learning* - podrobnejšie v Kapitole 2). Potrebovali sme jasné určenie, ktoré udalosti boli podvodmi, a ktoré nie. Táto skutočnosť je ale istá až po manuálnej kontrole, ktorej výsledkom je ušetrená finančná čiastka, teda rozdiel medzi udávanou a zistenou škodou. Rovnako poisťovňa nedisponuje databázou poistných udalostí, ktoré jednoznačne nie sú poistnými podvodmi, nakoľko automaticky nie je možné žiadnu poistnú udalosť s určitosťou vylúčiť. Museli sme teda použiť dáta už vybraté pôvodným systémom, ktoré boli manuálne prekontrolované, čo zapríčinilo viacero problémov.

V prvom rade manuálne prekontrolovaný bol len malý zlomok všetkých poistných udalostí. Tým sa nám výrazne znížil počet dát - z pôvodných tisícok na desiatky, v lepšom prípade stovky riadkov na scenár. Tento problém sa dal čiastočne vyriešiť zobratím starších dát, no metódy podvodníkov a aj celý poistný trh sa rýchlo a dynamicky menia, preto by ich použitie bolo kontraproduktívne. Výrazne najviac dát bol pre všeobecný scenár 1. Z toho dôvodu sme sa rozhodli po zvyšok práce venovať len tomuto scenáru. V prípade úspešného aplikovania metód strojového učenia na scenár 1 sa v budúcnosti už overené postupy môžu po zozbieraní väčšieho počtu dát aplikovať aj na ostatné scenáre. Druhým problémom bolo, že počet podvodov v dátach bol ovplyvnený úspešnosťou pôvodnej metódy expertného odhadu a nezobrazoval skutočnosť. Takmer polovica dát manuálne prekontrolovaných udalostí sa ukázala byť podvodom, čo je výrazne viac ako v realite. Množstvo podvodov v dátach na jednej strane ukazovalo na stále solídnu úspešnosť odhaľovania pomocou expertného odhadu. Na strane druhej mali manuálne skontrolované poistné udalosti hodnoty rizikovosti β na úrovni náhodného výberu, teda zamestnanci skutočne nevyberali udalosti na manuálnu kontrolu

len z tých najrizikovejších. Na základe skúseností s používaním systému boli zvolené naprieč celým spektrom rizikovosti použitím aj iných metrík.

Všetky metódy sme teda aplikovali len na scenár 1 a na priestore poistných udalostí vybratých pôvodným systémom, namiesto priestoru všetkých uskutočnených poistných udalostí.

1.2.2 Metodika - Klasifikácia

Hlavným nástrojom pri zlepšovaní pôvodného systému boli klasifikačné metódy natrénované pomocou učenia s učiteľom. Ich použitie je možné práve vďaka manuálne previereným udalostiam. Jednotlivci alebo skupiny podvodníkov sú pri svojej činnosti veľmi kreatívni, preto sa ich taktiky rýchlo vyvíjajú. Aj kvôli tejto dynamike nemusia stíhať tímy expertov jednotlivé taktiky odhaľovať, prípadne môžu niektoré taktiky ostať nezachytené. Klasifikačné metódy strojového učenia môžu len pomocou aktuálnych dát nájsť skryté spojitosti medzi určitými identifikátormi poistných udalostí a ich rizikovosťou. Zároveň pri pravidelnom trénovaní na najnovších dátach vedia potenciálne držať krok s organizovanými skupinami.

Keďže táto práca bola vypracovaná v spolupráci s poisťovňou Generali Poisťovňa, a. s., dávali sme veľký dôraz na použiteľnosť metód v praxi. Generali Poisťovňa, a. s. nemá tím dedikovaný na strojové učenie, no disponuje množstvom expertov, ktorí už dlhé roky s podvodmi v jednotlivých oblastiach neživotného poistenia bojujú. Z toho dôvodu je dôležité, aby výsledky boli pre expertov zrozumiteľné. Výsledné parametre a váhy, ktoré majú jasnú výpovednú hodnotu môžu v spojení so skúsenosťami expertov viesť k lepšej úspešnosti odhaľovania podvodov, ako by dosahovali jednotlivé svety samostatne. Zároveň musia byť metódy v praxi ľahko použiteľné, v prípade potreby upraviteľné a musia sa dať ľahko natrénovať na nových dátach.

Ako sme spomínali v predchádzajúcej kapitole, k dispozícii sme mali dáta o ušetrenej čiastke na manuálne prekontrolovaných poistných udalostiach. Výšku ušetrenej čiastky sme sa rozhodli pri trénovaní modelov nezohľadňovať, nakoľko aj malý podvod môže v sebe ukrývať pre expertov informácie o nových praktikách podvodníkov. Ak teda bola pri poistnej udalosti zachytená ušetrená čiastka, považovali sme ju jednoducho za podvod (1), v opačnom prípade sme podvod vylúčili (0).

Pri trénovaní sme sa snažili čo najviac eliminovať chybu druhého druhu. Označme si nulovú hypotézu:

$$H_0 : \textit{Daná poistná udalosť je podvodom},$$

chyba druhého druhu potom nastáva, ak vylúčime pri poistnej udalosti možnosť, že ide o podvod, no v skutočnosti je udalosť naozaj podvodom. Chyba prvého druhu by nastala, ak by sme poistnú udalosť prehlásili za podvod, no v skutočnosti by podvodom nebola. Pri chybe prvého druhu vznikajú náklady na čas zamestnancov, ktorí musia udalosť manuálne skontrolovať, aj keď to v skutočnosti nie je potrebné. Samotné zistenie, že poistná udalosť je v poriadku je oveľa rýchlejšie, ako zisťovanie veľkosti škôd, keď sa jedná o podvod. Pri chybe druhého druhu sa stane podvod úspešným a poisťovňa príde o finančné prostriedky, ktoré podvodníkovi vyplatila. Výšky podvodov sú pritom často v tisícoch až desaťtisícoch €. Zároveň úspešný podvod môže viesť k opakovaniu rovnakej schémy v budúcnosti. Z týchto dvoch modelových situácii je teda zjavné, že pri chybe druhého druhu prichádza poisťovňa o oveľa viac finančných prostriedkov. Všeobecne známe postupy, ktoré zaručia uprednostnenie minimalizácie chyby druhého druhu predstavujeme v Kapitole 2.

Posledným bodom, ktorý treba pri zadefinovaní metodiky spomenúť, je dôraz na poradie, nie klasifikáciu. Počet udalostí, ktoré stihnú zamestnanci za určité obdobie manuálne skontrolovať nie je predom daný. Závisí od mnohých faktorov, či už od zložitosti jednotlivých overovaní, prípadne od dovoleniek, počtu pracovných dní v mesiaci a podobne. Nie je prioritou označiť, ktoré udalosti sú potencionálnymi podvodmi, a ktoré nie, keďže akýkoľvek potencionálny podvod musí byť aj tak overený manuálne. Postačujúce je udalosti zoradiť podľa predikovanej pravdepodobnosti podvodu. Zamestnanci si, rovnako ako v pôvodnej metóde expertného odhadu, následne len zoradia udalosti zostupne podľa rizikovosti a začnú s manuálnou kontrolou. Úlohou nového systému teda bude, aby pomocou takéhoto výberu zamestnanci narazili na výrazne viac podvodov, ako len metódou náhodného výberu.

Dôraz na poradie sme zohľadnili aj pri testovaní, kde sme neostali len pri štandardných postupoch. Podrobnosti o testovaní sú uvedené v Kapitole 2.

1.2.3 Metodika - Zhlukovanie

Pozícia zhlukovania (*angl. clustering*) v nami vytvorenom systéme je úplne iná ako pozícia klasifikácie. Zatiaľ čo klasifikácia je základom a podstatou zlepšenia pôvodného systému, zhlukovanie je akýmsi doplnkom na dodatočné zvýšenie počtu odhalených podvodov. Zhlukovanie ako skupina metód strojového učenia zodpovedá trénovaniu bez učiteľa (*angl. unsupervised learning*). Slúži na rozdelenie záznamov do čo najpodobnejších skupín. Pre potreby tejto práce použijeme metódu K-means, ktorú bližšie opisujeme v Kapitole 2. Rovnako ako pri klasifikácii použijeme len expertmi vybraté identifikátory.

Podstatou použitia zhlukovania je nájdenie udalostí veľmi podobných udalostiam z minulosti, ktoré sú preukázanými podvodmi. Takéto udalosti majú potenciálne vyššiu šancu byť rovnako podvodmi. Aby sme dostali čo najmenší počet takých prípadov, rozdelili sme dáta do veľa malých skupín. Pre použitie zhlukovania v praxi bolo potrebné vymyslieť, ako by sa netradične dalo použiť na zoradenie udalostí podľa rizikovosti. Pre tento účel sme vytvorili nasledovný postup:

1. na historické dáta identifikátorov aplikujeme K-means zhlukovanie s cieľom vytvoriť veľký počet K malých skupín: $k = 1, \ldots, K$,
2. skupine k pridelíme rizikovosť β_{s_k} ako podiel počtu podvodov k celkovému počtu udalostí v skupine,
3. každú novú poistnú udalosť zaradíme do skupiny s čo najpodobnejšími záznamami (bližšie informácie v Kapitole 2.3),
4. ak nová poistná udalosť bola priradená do skupiny k, pridelíme jej rizikovosť β_{s_k},
5. nové poistné udalosti zoradíme podľa rizikovosti β a nasleduje manuálna kontrola opisovaná v predchádzajúcich kapitolách,
6. keď prídu ďalšie dáta, všetky predošlé manuálne overené zaradíme do historických a pokračujeme od bodu 1.

Takáto metóda by mala logicky umožniť rýchlo a jednoducho nájsť opakujúce sa podvody, prípadne v nich odhaliť určitý systém. Dôležité pre jej úspešnosť je, aby

principiálne rovnaké podvody mali aj veľmi podobné hodnoty čŕt, ktoré vybrali experti. Ako tento postup použiť v spojení s klasifikáciou rozoberáme v Kapitole 3.

2 Teória

2.1 Strojové učenie

V dnešnej dobe narážame na mnoho problémov, ktorých riešenie nie je natoľko jasné, aby sme jeho jednotlivé kroky vedeli klasickým spôsobom naprogramovať. Známe sú vstupy a požadované výstupy, ale ich vzájomný funkčný vzťah je veľmi komplexný a neznámy. Nedostatok vedomostí o riešení ale dokážeme nahradiť množstvom kvalitných dát. Príkladom uvedeným v [3] je predpovedanie zákazníckeho správania. Ak človek používa zákaznícku kartu alebo nakupuje online, obchodník zbiera bežne dáta o každej jeho transakcii - od dátumu, nakúpených položiek až po celkový objem minutých peňazí. Zákaznícke správanie ale nemusí byť nijakým spôsobom evidentné, vieme len, že sa v dátach nachádzajú určité súvislosti, keďže zákazníci nenakupujú úplne náhodne. Pri kúpe sladeného nápoja je väčšia šanca, že si kúpia čipsy, v lete je väčšia spotreba zmrzliny a podobne. V praxi takmer nikdy nepotrebujeme úplne odhaliť správanie zákazníka, postačujúca je určitá aproximácia. Od aproximácie požadujeme vysvetlenie časti dát a detekovanie závislostí, pomocou ktorých vieme s určitou presnosťou predikovať budúce správanie zákazníkov. Problémy ako tento sú ideálne na použitie metód strojového učenia. Ak si zameníme zákazníka za poistenca, dáta o nákupoch za dáta o minulých poistných udalostiach a namiesto správania jednotlivého zákazníka budeme chcieť predikovať, či je poistná udalosť podvodom, dostávame sa k predmetu tejto práce.

Pomocou [14] si zadefinujme základné pojmy strojového učenia:

- **črty** $x \in \mathbb{R}^n$ sú atribúty daného objektu alebo javu, pomocou ktorých chceme modelovať cieľovú hodnotu,
- **cieľová hodnota** $y = f(x), y \in \mathbb{R}^m$, je modelovaná premenná závislá od čŕt, črty a cieľové hodnoty medzi sebou mapuje neznáma funkcia $f : \mathbb{R}^n \to \mathbb{R}^m$,
- **hypotéza** $h : \mathbb{R}^n \to \mathbb{R}^m$ je funkcia, ktorá aproximuje f, teda $h(x) = \hat{f}(x) = \hat{y}$,
- **množina hypotéz** H obsahuje všetky hypotézy, z ktorých sa snažíme vybrať optimálnu; rôzne metódy strojového učenia môžu mať rôzne množiny hypotéz,

- **trénovacia množina** $T = (x^1, y^1), (x^2, y^2), \ldots, (x^r, y^r)$ je vzorka r dát reprezentujúca črty a funkčné hodnoty funkcie f,
- **chybová funkcia** $J(h(x), y) : \mathbb{R}^n \to \mathbb{R}$ je kľúčová pre proces trénovania, meriame pomocou nej kvalitu aktuálnej hypotézy (čím menšia funkčná hodnota, tým metóda lepšie aproximuje f v bodoch trénovacej množiny); tvar chybovej funkcie je potrebné zvoliť pred začiatkom trénovania,
- **optimálna hypotéza** h^* je najlepším odhadom funkcie f spomedzi všetkých hypotéz z množiny H, pričom slovo „najlepším" chápeme v zmysle vyššie definovanej chybovej funkcie.

Štandardne sa v literatúre vyskytuje aj iný, pravdepodobnostný spôsob zadefinovania základných pojmov strojového učenia, ktorý je potrebný v Kapitole 2.2.1. V tomto spôsobe aj ak majú záznamy i a j rovnakú hodnotu čŕt $x^i = x^j$, nemusí byť ich teoreticky správna klasifikácia rovnaká.

Hlavným predmetom strojového učenia je spomedzi všetkých hypotéz v priestore H nájsť takú, ktorá najlepšie aproximuje funkciu f. Celý tento proces sa bežne označuje ako „trénovanie". Predpokladom úspešného nájdenia aproximácie, teda úspešného trénovania, je dostatočná bohatosť množiny H. Ak berieme do úvahy len trénovaciu množinu, vieme nájsť h^{tr} spĺňajúcu:

$$h^{tr} := \arg\min_{h \in H} \sum_{x \in T} J(h(x), y). \tag{3}$$

Z (3) vyplýva, že h^{tr} skutočne najlepšie aproximuje v bodoch z T funkciu f. Našim cieľom je ale čo najlepšie aproximovať f aj v bodoch, pre ktoré nemáme dostupné dáta. Chceme teda využiť čo najviac informácií dostupných v množine T, no zároveň nechceme model príliš prispôsobiť náhodnému šumu, ktorý sa v dátach môže nachádzať. Informácie v T pritom využívame práve riešením úlohy (3), pretože jej optimalizáciou približujeme aproximáciu funkcie f k jej známym funkčným hodnotám. Ukazuje sa, že proces hľadania h^{tr} nás obvykle v prvej fáze trénovania približuje k h^*, ale od istého momentu nás môže od h^{tr} vzďaľovať. Tento problém je v teórii strojového učenia známy ako pretrénovanie (*angl. overfitting*). Na predídenie pretrénovania existuje množstvo metód a praktických postupov, z ktorých niektoré bližšie popisujeme v Kapitole 2.4. Ďalšou komplikáciou je, že nájdenie h^{tr} v takejto forme je obvykle veľmi

zložitý optimalizačný problém. Mnoho optimalizačných metód má na účelovú funkciu určité požiadavky, no my o vlastnostiach a tvare f často máme len málo informácií. V tejto práci sme pre trénovanie použili parametrické metódy, ktorými si problém zjednodušíme predpokladaním tvaru funkcie f a teda zmenšením priestoru H len na určitý druh funkcií. Ako bližšie popisuje [4], vďaka tomu sa úloha transformuje z hľadania aproximácie f na optimalizáciu k parametrov θ. Namiesto h^* a riešenia (3) teda hľadáme h^*_θ riešením optimalizačnej úlohy:

$$h^{tr}_\theta := \arg\min_{h_\theta \in H} \sum_{x \in T} J(h_\theta(x), y). \tag{4}$$

Zvolený tvar pritom môže byť veľmi jednoduchý, ako napríklad lineárny v prípade lineárnej regresie:

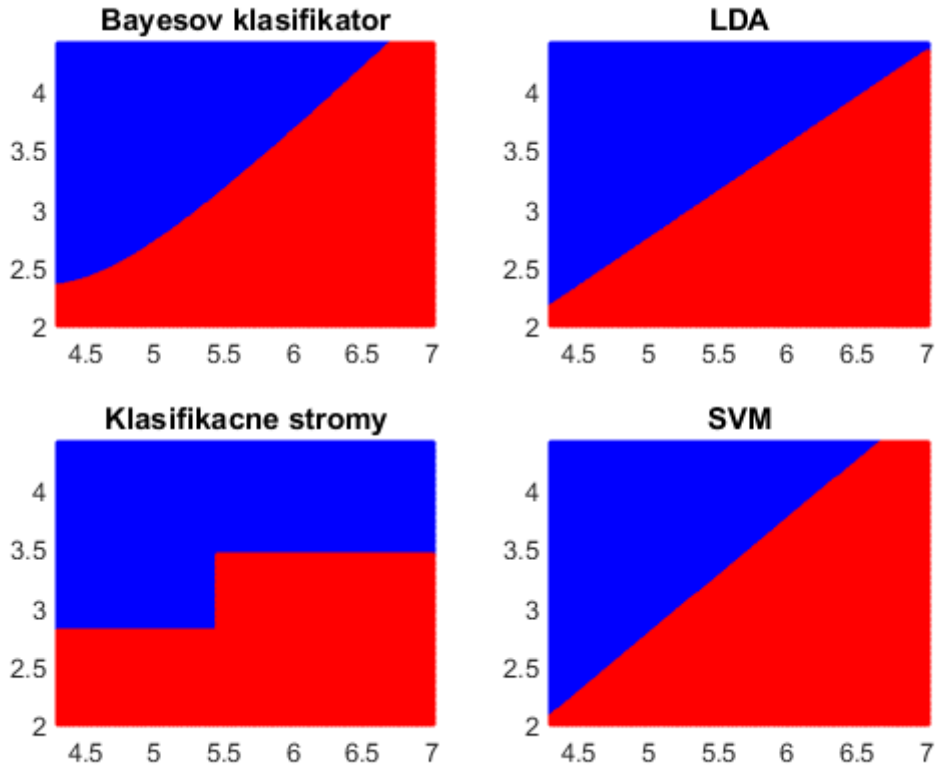
$$H := \{h_\theta; h_\theta(x) = \theta_0 + \sum_{i=1}^{k-1} x_i\theta_i;\ x \in \mathbb{R}^{k-1},\ \theta \in \mathbb{R}^k\}, \tag{5}$$

kde x_i predstavuje i-tu zložku vektora x a θ_i predstavuje $(i+1)$-vú zložku vektora θ. Tvar (5) umožňuje veľmi rýchle riešenie problému a jasne interpretovateľný vzťah medzi atribútmi x a cieľovou premennou y. Na druhej strane ale predpokladaný tvar funkcie f môže byť natoľko nepresný a tým pádom aj veľkosť H natoľko obmedzená, že aj h^*_θ bude stále nízkej kvality. Bez zmeny predpokladaného tvaru f jednoducho nemusí byť možné dostať chybu $J(h_\theta(x), y)$ pod určitú hodnotu. Rovnako ako v prípade pretrénovania, kde hľadáme h^{tr}_θ, no vo väčšine prípadov sa nechceme dostať do minima $J(h_\theta(x), y)$ úplne, bojujú aj v tomto prípade proti sebe dva prúdy. Na jednej strane chceme jednoduché a ľahko interpretovateľné modely, no na strane druhej nechceme obmedziť množinu H natoľko, že zabránime nájdeniu zložitejších a presnejších aproximácií f. Rôzne parametrické metódy strojového učenia sú práve definované predpokladaným tvarom f, respektíve množinou hypotéz H. Správne určenie metódy strojového učenia a správne zvolenie chybovej funkcie $J(h_\theta(x), y)$ sú pre úspešne vyriešenie problému kľúčové.

Ako sme načrtli v predošlej kapitole, metódy strojového učenia delíme aj na základe spôsobu učenia. Doteraz sme sa pri vysvetľovaní teórie zaoberali len učením s učiteľom. Tento názov je založený na povahe trénovacej množiny, nakoľko okrem hodnôt čŕt x v nej máme vždy prítomné aj prislúchajúce cieľové hodnoty y. Častým použitím učenia

s učiteľom je predikcia hodnôt y pre hodnoty čŕt x, ktoré sa nenachádzali v trénovacej množine. V našom prípade teda máme k informáciám o poistnej udalosti k dispozícií aj výsledky preverenia, či bola udalosť podvodom. Učenie s učiteľom sa najbežnejšie používa pri klasifikácii. Klasifikáciou označujeme problém, v ktorom cieľová hodnota y, a teda aj neznáma funkcia f nadobúda len určitý počet hodnôt, ktoré označujeme ako skupiny (*angl. classes*), pričom každý záznam patrí len do jednej skupiny. Našou úlohou je teda na základe príkladov z trénovacej množiny T aproximovať funkciu f tak, aby sme každý nový záznam s čo najväčšou presnosťou správne klasifikovali do určitej skupiny. V našom prípade máme, ako sme popísali v predošlej kapitole, dve skupiny - buď je udalosť podvodom $y = f(x) = 1$ alebo podvodom nie je $y = f(x) = 0$. Z tohto dôvodu sa ďalej v teórii obmedzíme len na binárnu klasifikáciu, pričom $y \in \mathbb{R}$, teda $m = 1$. Ako sme už zdôraznili, pre použitie metód strojového učenia v poisťovni Generali Poisťovňa, a. s. nie je kľúčové udalosti správne zaklasifikovať, ale ich zoradiť podľa pravdepodobnosti toho, že sú podvodom. Nástroje umožňujúce takéto zoradenie bývajú už priamo zabudované do klasifikačných metód pre dôvody spomenuté v [3]. V prvom rade po natrénovaní modelu vznikne v priestore čŕt x určitá rozhodovacia hranica, ktorá rozdeľuje priestor na disjunktné podpriestory, pričom každý z nich je jasne klasifikovaný do jednej zo skupín cieľovej hodnoty. Podpriestor klasifikovaný do $y = 1$ je množinou všetkých bodov x, pre ktoré je $h(x) > \lambda$, pričom zvyčajne $\lambda = 0.5$. Rovnako podpiestor klasifikovaný do $y = 0$ je množinou všetkých bodov x, pre ktoré je $h(x) < \lambda$. Rozhodovacia hranica má tvar odvodený od funkcie z množiny H, teda na základe zvolenej metódy strojového učenia. Na Obr. 1 vidíme príklady rôznych tvarov rozhodovacej hranice, ktorá od seba oddeľuje modrý a červený podpriestor. Rozhodovacia hranica metódy „LDA" a lineárnej verzie metódy „SVM" je lineárna, no aj po trénovaní na tých istých dátach má iný predpis.

Ak následne dostaneme nový záznam čŕt x_p, predikujeme jeho skupinu y na základe klasifikácie podpriestoru, do ktorého bod x_p spadá. Body nachádzajúce sa relatívne blízko rozhodovacej hranice následne považujeme za také, ktorými si metóda nie je úplne istá. Dôvodom je, že stačí aby sa jedna z čŕt daného bodu mierne zmenila a daný bod sa môže stať úplne inak klasifikovaným. Príkladom je poistná udalosť nachádzajúca sa pri rozhodovacej hranici klasifikovaná ako podvod, pričom stačí aby sa vek vodiča



Obr. 1: Príklady rôznych rozhodovacích hraníc pre $n = 2$

znížil o jeden rok a poistnú udalosť začneme považovať za bezproblémovú. Naopak bodom nachádzajúcim sa v strede podpriestorov sa môže akákoľvek črta zmeniť aj o väčšiu hodnotu a ostanú rovnako klasifikované, teda si je danou klasifikáciou metóda relatívne istá. Zvolením chybovej funkcie, ktorá zohľadní vzdialenosť klasifikovaného bodu od hranice podľa [3] zlepšíme zovšeobecnenie daného modelu. Zároveň rovnaký princíp môžeme uplatniť na cieľovú hodnotu. Namiesto diskrétnej cieľovej hodnoty hovoriacej o tom, či je udalosť podvodom, bude cieľová hodnota reálne číslo z intervalu $[0, 1]$. Hodnota $y_p = 0.5$ znamená, že model je úplne indiferentný a nevie rozhodnúť, do ktorej skupiny bod čŕt x_p patrí, je to teda bod najväčšej neistoty. Pre tento účel je dôležitá sigmoidálna funkcia:

$$g(t) = \frac{1}{1 + e^{-t}}, \tag{6}$$

ktorá je vhodná kvôli vlastnostiam:

1. spojitosť, diferencovateľnosť a rastúcosť na $\mathbb{R}$,
2. $g(0) = 0.5$,

3. $\lim_{t\to\infty} g(t) = 1$,

4. $\lim_{t\to-\infty} g(t) = 0$,

kde t predstavuje vzdialenosť od rozhodovacej hranice, pričom kladná hodnota znamená, že sa bod nachádza v podpriestore s klasifikáciou $y = 1$ a naopak záporná hodnota znamená, že sa bod nachádza v podpriestore s klasifikáciou $y = 0$. Čím ďalej sa bod nachádza od rozhodovacej hranice, tým sa hodnota $g(t)$ viac približuje ku klasifikácii daného podpriestoru.

Ďalším spôsobom učenia, ktoré pri hľadaní podvodných udalostí použijeme, je učenie bez učiteľa (*angl. unsupervised learning*). V tomto prípade máme dostupné len črty x bez cieľových hodnôt y. Ako opisuje [4], jedná sa o náročnú situáciu, v ktorej nemôžeme predikovať cieľovú hodnotu, pretože pracujeme naslepo. Jednou z možných analýz, ktorú vieme v tomto prípade spraviť a zároveň sa s ňou v práci zaoberáme, je zhlukovanie. Cieľom analýzy zhlukovania je pozrieť sa na hodnoty čŕt $x^1, x^2, \ldots, x^r$ a zistiť, či nespadajú do relatívne odlišných skupín. V našom prípade sa teda môžeme pozrieť, či určitá udalosť nespadá do skupiny, v ktorej bolo v minulosti objavených viacero podvodov. Bližšie sa fungovaniu konkrétnej metódy K-means venujeme v Kapitole 2.3.

2.2 Klasifikácia

2.2.1 Bayesov klasifikátor

Ako prvú zo známych klasifikačných metód strojového učenia predstavíme Bayesov klasifikátor. Ide o staršiu metódu zadefinovanú v polovici minulého storočia, no vysoký teoretický a praktický význam má dodnes. Rozhodovanie o klasifikácii pri Bayesovom klasifikátore je založené na Bayesovom pravidle, ktoré je zároveň hlavným dôvodom, prečo túto metódu uvádzame. Princípy použité v Bayesovom pravidle používame pri skladaní metódy na mieru šitej potrebám Generali Poisťovňa, a. s. uvedenej v Kapitole 2.2.6. Všetky všeobecne známe informácie o Bayesovom klasifikátore čerpáme z [7].

Ako spomíname v predošlej časti, predpokladať budeme už len binárne rozhodovanie medzi dvomi skupinami. Pre odvodenie Bayesovho pravidla potrebujeme najprv zadefinovať pár dodatočných označení a pojmov:

- X je spojitý náhodný vektor, ktorého pozorovania sme v predošlej sekcii zadefinovali ako črty x,
- y_1 a y_2 sú dve binárne skupiny ($y_1 = 1, y_2 = 0$), do ktorých chceme klasifikovať pomocou čŕt x,
- φ množina čŕt x z trénovacej množiny T, teda $\varphi = \{x^1, x^2, \ldots, x^r\}$,
- φ_i je podmnožina φ tvorená všetkými takými x, ktoré patria do skupiny y_i,
- p_i je pravdepodobnosť, že vektor čŕt x predstavujúci pozorovanie náhodného vektora X patrí do skupiny y_i, ide o celkový podiel tých objektov, ktoré patria do skupiny y_i, teda nezávisí na konkrétnych vektoroch čŕt x,
- c_{ij} je strata, ktorá je spôsobená zaradením x do skupiny y_i reprezentovanej polpriestorom φ_i, ak x v skutočnosti patrí do skupiny y_j, strata c_{ij} nezávisí na konkrétnych vektoroch čŕt x,
- $p_X(\cdot|y_i)$ je hustota podmienenej pravdepodobnosti náhodnej premennej X za podmienky, že x patrí do skupiny y_i.

Nakoľko rozhodujeme len medzi dvomi skupinami y_1 a y_2, obmedzíme indexy i a j na hodnoty z množiny $\{1, 2\}$. Takto zadefinovanými stratami c_{ij} pripúšťame pridelenie určitej penalizácie aj po správnom zaradení vektora čŕt x do príslušnej skupiny. Tento prípad nastáva pri c_{11} a c_{22}, no v praxi najčastejšie položíme obidve straty rovné 0. Aby malo zaradenie členov c_{11} a c_{22} zmysel, budeme požadovať, aby strata pri správnej klasifikácii bola nižšia ako strata pri nesprávnej klasifikácii, teda:

$$c_{11} < c_{21},\ c_{22} < c_{12}.$$

Dôležitou pri odvodzovaní Bayesovho pravidla je aj vlastnosť pravdepodobnosti p vyplývajúca zo samotnej definície pravdepodobnosti:

$$p_1 + p_2 = 1. \tag{7}$$

Na základe [7] pri Bayesovom klasifikátore minimalizujeme namiesto funkcie straty $J(\cdot)$ priemerné riziko zlej klasifikácie ρ. Našim cieľom je minimalizovať ρ a tým maximalizovať kvalitu klasifikácie. Toto riziko je sumárne pre celú množinu φ a je definované

vzťahom:

$$\begin{aligned}\rho =& c_{11}p_1 \int_{\varphi_1} p_X(x|y_1)dx + c_{22}p_2 \int_{\varphi_2} p_X(x|y_2)dx \\ &+ c_{21}p_1 \int_{\varphi_2} p_X(x|y_1)dx + c_{12}p_2 \int_{\varphi_1} p_X(x|y_2)dx.\end{aligned} \tag{8}$$

Prvé dva členy v rovnici (8) predstavujú strednú hodnotu straty v prípade správnej klasifikácie, tretí člen predstavuje strednú hodnotu straty v prípade, že nesprávne klasifikujeme do skupiny y_2 a posledný nesprávnu klasifikáciu do skupiny y_1. Stredná hodnota straty je súčinom straty c_{ij} a frekvencie, v ktorej sa daná strata vyskytuje. Napríklad frekvencia výskytu straty c_{12} je daná súčinom p_2 a pravdepodobnosti, že ak je bod z podpriestoru φ_1, tak je zo skupiny y_2. Skupiny y_1 a y_2 sú disjunktné, teda každý vektor čŕt x je jasne zadefinovaný do jednej z nich. Pre podpriestory φ_1 a φ_2 platí:

$$\varphi = \varphi_1 \cup \varphi_2. \tag{9}$$

Pomocou (9) prepíšeme (8) na tvar:

$$\begin{aligned}\rho =& c_{11}p_1 \int_{\varphi_1} p_X(x|y_1)dx + c_{22}p_2 \int_{\varphi-\varphi_1} p_X(x|y_2)dx \\ &+ c_{21}p_1 \int_{\varphi-\varphi_1} p_X(x|y_1)dx + c_{12}p_2 \int_{\varphi_1} p_X(x|y_2)dx,\end{aligned} \tag{10}$$

a rozdelíme integrály:

$$\begin{aligned}\rho =& c_{11}p_1 \int_{\varphi_1} p_X(x|y_1)dx + c_{22}p_2 \int_{\varphi} p_X(x|y_2)dx - c_{22}p_2 \int_{\varphi_1} p_X(x|y_2)dx \\ &+ c_{21}p_1 \int_{\varphi} p_X(x|y_1)dx - c_{21}p_1 \int_{\varphi_1} p_X(x|y_1)dx + c_{12}p_2 \int_{\varphi_1} p_X(x|y_2)dx.\end{aligned} \tag{11}$$

Pre ďalšiu úpravu rovnice (11) musíme uskutočniť pozorovanie:

$$\int_{\varphi} p_X(x|y_1)dx = \int_{\varphi} p_X(x|y_2)dx = 1, \tag{12}$$

ktoré vyplýva zo vzťahu (9). Inými slovami, (12) platí, lebo pravdepodobnosť výskytu bodu zaradeného do jednej zo skupín v množine φ je 1, nakoľko φ obsahuje φ_1 aj φ_2. Tým dostávame záverečnú podobu priemerného rizika uvedenú v [7] v tvare:

$$\begin{aligned}\rho =& c_{21}p_1 + c_{22}p_2 \\ &+ \int_{\varphi_1} [p_2(c_{12} - c_{22})p_X(x|y_2) - p_1(c_{21} - c_{11})p_X(x|y_1)]dx.\end{aligned} \tag{13}$$

Na prvý pohľad sú na (13) zaujímavé prvé dva členy, ktoré vôbec neobsahujú vektor čŕt x. Tieto členy predstavujú fixné straty. Nezáleží na tom, aké pozorovania x uskutočníme, členy predstavujúce fixné straty budú mať stále rovnakú hodnotu. Po úprave priemerného rizika ρ môžeme pristúpiť k jeho minimalizácii. Minimálnu hodnotu (13) dosiahneme, ak sa budeme držať nasledovných pravidiel:

1. všetky zložky vektora x, pre ktoré majú výrazy vo vnútri integrálu zápornú hodnotu, klasifikujeme do skupiny y_1, vďaka čomu znížime hodnotu celého integrálu a tým aj očakávanú stratu ρ,

2. naopak, všetky zložky vektora x, pre ktoré majú výrazy vo vnútri integrálu kladnú hodnotu, vylúčime z podpriestoru φ_1 tým, že ich klasifikujeme do skupiny y_2,

3. všetky zložky vektora x, pre ktoré majú výrazy vo vnútri integrálu nulovú hodnotu, môžeme voľne zaradiť, pretože nemajú vplyv na veľkosť ρ (pre jednoznačnosť ich zaradíme do skupiny y_2).

Sumarizáciou týchto pravidiel vieme jednoducho definovať **Bayesovo pravidlo**:

Ak platí $p_1(c_{21} - c_{11})p_X(x|y_1) > p_2(c_{12} - c_{22})p_X(x|y_2)$, potom klasifikujeme pozorovanie z vektora x do skupiny y_1, v opačnom prípade do y_2.

Nerovnica v tomto pravidle sa bežnejšie objavuje v upravenej verzii:

$$\Lambda(x) > \xi, \tag{14}$$
$$\Lambda(x) = \frac{p_X(x|y_1)}{p_X(x|y_2)}, \xi = \frac{p_2(c_{12} - c_{22})}{p_1(c_{21} - c_{11})},$$

pričom $\Lambda(x)$ sa nazýva pomer vierohodností (*angl. likehood ratio*) a ξ sa nazýva prah (*angl. threshold*). Autor v [15] pripomína, že ak predpokladáme rovnakú frekvenciu výskytu jednotlivých skupín $p_1 = p_2$ a zároveň straty c_{ij} zvolíme v rovnakej výške, ide o štandardizovaný prípad, v ktorom $\xi = 1$.

Dáta sa pri Bayesovom klasifikátore využívajú čisto na odhad pomeru vierohodností $\Lambda(x)$, čo vyžaduje predpokladanie pravdepodobnostného rozdelenia náhodnej premennej X. Pre urýchlenie výpočtov sa v praxi namiesto pomeru vierohodností využíva logaritmus pomeru vierohodností (*angl. log-likehood ratio*). Ten je možné použiť vďaka

kladnosti $\Lambda(x)$ a ξ a vďaka monotónnosti funkcie logaritmu. Bližšie podrobnosti ohľadom Bayesovho klasifikátora sú napríklad v [4], [5], [7] a [8].

2.2.2 Lineárna diskriminačná analýza

Jednou z najjednoduchších klasifikačných metód je lineárna diskriminačná analýza (skr. LDA). Táto metóda bola vyvinutá s cieľom aproximácie funkcie f pomocou lineárnych funkcií. Model predpokladá normálne rozdelenie náhodnej premennej X. Podľa [5] a [15] vychádza LDA z Bayesovho pravidla (14) (použitím logaritmu pomeru vierohodností $l(x)$):

$$l(x) = \log(\Lambda(x)) = \log\Big(\frac{p_X(x|y_1)}{p_X(x|y_2)}\Big). \tag{15}$$

Ak predpokladáme normalitu náhodnej premennej X, potom na základe [15] podmienené pravdepodobnosti $p_X(x|y_1)$ a $p_X(x|y_2)$ majú taktiež $(k-1)$-rozmerné normálne rozdelenie, teda:

$$\begin{aligned} p_X(x|y_1) &\sim N_{k-1}(\mu_1, \Sigma), \\ p_X(x|y_2) &\sim N_{k-1}(\mu_2, \Sigma), \end{aligned}$$

kde $x \in \mathbb{R}^{k-1}$ ako vyplýva zo vzťahu (5), $\mu_1 \in \mathbb{R}^{k-1}$ je stredná hodnota $p_X(x|y_1)$, $\mu_2 \in \mathbb{R}^{k-1}$ je stredná hodnota $p_X(x|y_2)$ a $\Sigma \in \mathbb{R}^{(k-1)\times(k-1)}$ je pozitívne definitná kovariančná matica. Predpokladáme rovnakú Σ pre obe skupiny. Hustota $p_X(x|y_i)$ má tvar:

$$p_X(x|y_i) = \frac{1}{(2\pi)^{\frac{k-1}{2}}\sqrt{\det\Sigma}} \exp(-\frac{1}{2}(x-\mu_i)^T\Sigma^{-1}(x-\mu_i)). \tag{16}$$

Do (15) dosadíme hustoty rozdelení podmienených pravdepodobností a rovnicu upravíme:

$$\begin{aligned} \log(\frac{p_X(x|y_1)}{p_X(x|y_2)}) =& \log\Big(\frac{\exp(-\frac{1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1))}{\exp(-\frac{1}{2}(x-\mu_2)^T\Sigma^{-1}(x-\mu_2))}\Big) \\ =& \frac{1}{2}(x-\mu_2)^T\Sigma^{-1}(x-\mu_2) - \frac{1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)) \\ =& (\mu_1-\mu_2)^T\Sigma^{-1}x + \frac{1}{2}(\mu_2^T\Sigma^{-1}\mu_2 - \mu_1^T\Sigma^{-1}\mu_1). \end{aligned} \tag{17}$$

Následne, ak podľa [15] označíme:

$$(\theta_1, \theta_2, \ldots, \theta_{k-1})^T = (\mu_1-\mu_2)^T\Sigma^{-1}, \tag{18}$$

$$\theta_0 = \frac{1}{2}(\mu_2^T\Sigma^{-1}\mu_2 - \mu_1^T\Sigma^{-1}\mu_1), \tag{19}$$

v konečnom dôsledku dostávame:

$$l(x) = \log\left(\frac{p_X(x|y_1)}{p_X(x|y_2)}\right) = \sum_{i=1}^{k-1} x_i\theta_i + \theta_0, \tag{20}$$

čo zodpovedá tvaru funkcií z množiny hypotéz lineárnej regresie (5). Teda skutočne, ak v Bayesovom pravidle predpokladáme normálne rozdelenie náhodnej premennej X a rovnakú Σ pre rozdelenia obidvoch skupín, potom je rozhodovacia hranica lineárna. Tvar (20) je na jednej strane veľmi žiaduci, lebo linearita zaručuje jednoduchosť modelu a ľahkú interpretovateľnosť, no na strane druhej nadobúda hodnoty z celého rozsahu reálnych čísel. Ak pre určitý nový záznam x_p je $p_X(x|y_1) \gg p_X(x|y_2)$, $l(x)$ nadobúda veľkú kladnú hodnotu a naopak, ak $p_X(x|y_1) \ll p_X(x|y_2)$, $l(x)$ nadobúda veľmi veľkú zápornú hodnotu. Našim cieľom je ale modelovať priamo $p_X(x|y_i) \in [0, 1]$. Postup na odvodenie výrazu pre $p_X(x|y_1)$ z (20) opisuje [5]. Ak si uvedomíme z vlastností pravdepodobnosti:

$$p_X(x|y_2) = 1 - p_X(x|y_1), \tag{21}$$

môžeme (20) upraviť nasledovne:

$$\begin{aligned}
\frac{p_X(x|y_1)}{p_X(x|y_2)} &= \exp(\sum_{i=1}^{k-1} x_i\theta_i + \theta_0), \\
\frac{p_X(x|y_1)}{1 - p_X(x|y_1)} &= \exp(\sum_{i=1}^{k-1} x_i\theta_i + \theta_0), \\
p_X(x|y_1)(1 + \exp(\sum_{i=1}^{k-1} x_i\theta_i + \theta_0)) &= \exp(\sum_{i=1}^{k-1} x_i\theta_i + \theta_0), \\
p_X(x|y_1) &= \frac{\exp(\sum_{i=1}^{k-1} x_i\theta_i + \theta_0)}{1 + \exp(\sum_{i=1}^{k-1} x_i\theta_i + \theta_0)}, \\
p_X(x|y_1) &= \frac{1}{1 + \exp(-(\sum_{i=1}^{k-1} x_i\theta_i + \theta_0))}.
\end{aligned} \tag{22}$$

Môžeme si všimnúť, že (22) je vlastne sigmoidálnou funkciou $g(\cdot)$:

$$p_X(x|y_1) = g(\sum_{i=1}^{k-1} x_i\theta_i + \theta_0). \tag{23}$$

Trénovanie je v prípade tejto metódy veľmi priamočiare. Predpoklady sú také silné, že na ich základe môžeme analyticky odvodiť vzťahy (18) a (19) pre hodnoty parametrov θ. Z tohto dôvodu nám stačí pre odhad optimálnych θ odhadnúť parametre μ_1, μ_2 a Σ z trénovacej vzorky dát.

LDA so sebou ale prináša mnoho nevýhod plynúcich hlavne z predpokladu normality X. Tento predpoklad by sa mal pred použitím metódy riadne otestovať na vzorke trénovacích dát, no v praxi sa testovanie často obchádza. Nesplnenie tohto predpokladu so sebou prináša skutočnosť, že použitie metódy nebude rigorózne, no v praxi je podstatná úspešnosť klasifikácie. Výsledky skúmaní v [8] potvrdzujú, že pri nesplnení predpokladu normality vykazuje LDA horšiu úspešnosť klasifikácie, ako iné alternatívne metódy. Normalita mnohých čŕt poistných udalostí poisťovne Generali Poisťovňa, a. s. je navyše na základe testov vylúčená, preto sme sa rozhodli LDA (a celú skupinu metód s rovnakým predpokladom) napriek jej ľahkej interpretácii a analytickému riešeniu nepoužiť.

2.2.3 Logistická regresia

Logistická regresia je metódou, ktorá prináša takmer všetky výhody LDA a zároveň eliminuje jej hlavnú nevýhodu. Zaujímavý spôsob pohľadu na logistickú regresiu opisuje [3]. Na rozdiel od LDA v nej nemodelujeme priamo hustoty pravdepodobností $p_X(x|y_1)$ a $p_X(x|y_2)$, ale ich pomer. Jediným predpokladom logistickej regresie je linearita logaritmu pomeru vierohodností:

$$\frac{p_X(x|y_1)}{p_X(x|y_2)} = \sum_{i=1}^{k-1} x_i\theta_i + \theta_0^0, \tag{24}$$

zatiaľ čo v prípade LDA táto linearita vyplývala z predpokladu normality X. Predpoklady sa teda stali menej striktnými. V prvom rade potrebujeme Bayesovu vetu, ktorá na základe [9] hovorí:

$$p(y = y_1|X = x) = \frac{p_1 p_X(x|y_1)}{p_X(x)}. \tag{25}$$

Logistická regresia sa v literatúre predstavuje aj priamo definovaním modelu, no k rovnakému výsledku sa vieme dostať aj odvodením z logaritmu pomeru vierohodnosti:

$$\begin{aligned}\log\Big(\frac{p(y_1|x)}{p(y_2|x)}\Big) &= \log\Big(\frac{\frac{p_1 p_X(x|y_1)}{p_X(x)}}{\frac{p_2 p_X(x|y_2)}{p_X(x)}}\Big) = \log\Big(\frac{p_1 p_X(x|y_1)}{p_2 p_X(x|y_2)}\Big) \\ &= \log\Big(\frac{p_X(x|y_1)}{p_X(x|y_2)}\Big) + \log\Big(\frac{p_1}{p_2}\Big) = \sum_{i=1}^{k-1} x_i\theta_i + \theta_0,\end{aligned} \tag{26}$$

kde $\theta_0 = \theta_0^0 + \log(\frac{p_1}{p_2})$. Ak v (26) preusporiadame členy rovnako ako v (20), dostávame sigmoidálnu funkciu:

$$p(y_1|x) = \frac{1}{1 + \exp(-(\sum_{i=1}^{k-1} x_i\theta_i + \theta_0))} = g(\sum_{i=1}^{k-1} x_i\theta_i + \theta_0),$$

ktorá je odhadom hustoty $p(y_1|x)$ rovnako ako pri LDA (23).

Trénovanie pri logistickej regresii, teda na nájdenie optimálnych $\theta_i, i = 0, \ldots, k-1$, môže prebiehať viacerými spôsobmi. Sledovať budeme postup uvedený v [3]. Nech $T = (x^1, y^1), (x^2, y^2), \ldots, (x^r, y^r)$ je trénovacia množina r dát, kde $y^j = y_1 = 1$, ak $x^j \in \varphi_1$ a $y^j = y_2 = 0$, ak $x^j \in \varphi_2$. Predpokladajme, že náhodná premenná y^j pri danom x^j má alternatívne rozdelenie s pravdepodobnosťou $p(y_1|x^j)$:

$$y^j|x^j \sim Alt(p(y_1|x^j)). \tag{27}$$

Tu môžeme podľa [3] vidieť ďalší rozdiel medzi logistickou regresiou a LDA, kde pri LDA sme modelovali $p(y_1|x)$, no pri logistickej regresii modelujeme priamo $y|x$. Výberová vierohodnostná funkcia má tvar:

$$l(x) = \prod_{i=1}^{r} p(y_1|x^i)^{y^i}(1 - p(y_1|x^i))^{(1-y^i)}. \tag{28}$$

Pre nájdenie optimálnych θ je potrebné funkciu vierohodnosti maximalizovať, no namiesto toho ju môžeme transformovať na funkciu straty a minimalizovať:

$$J(h_\theta(x), y) = -\sum_{i=1}^{r} y^i \log(p(y_1|x^i)) + (1 - y^i)\log(1 - p(y_1|x^i)). \tag{29}$$

Stratová funkcia typu (29) je známa ako I-divergencia. Na minimalizáciu funkcie straty použijeme čerpajúc z [3] gradientnú metódu s nasledovnými optimalizačnými krokmi:

$$\begin{aligned}\Delta\theta_j &= -\eta\frac{\partial J}{\partial\theta_j} = \eta\sum_{i=1}^{r}\Big(\frac{y^i}{p(y_1|x^i)} - \frac{1-y^i}{1-p(y_1|x^i)}\Big)p(y_1|x^i)(1-p(y_1|x^i))x_j^i \\ &= \eta\sum_{i=1}^{r}(y^i - p(y_1|x^i))x_j^i,\ j = 1, ..., (k-1), \qquad (30)\\ \Delta\theta_0 &= -\eta\frac{\partial J}{\partial\theta_0} = \eta\sum_{i=1}^{r}(y^i - p(y_1|x^i)). \qquad (31)\end{aligned}$$

Gradientná metóda teda v každej iterácii postupuje v smere opačnom ku gradientu stratovej funkcie a s dĺžkou kroku závislou od parametra $\eta \in \mathbb{R}$. Celý postup trénovania

môžeme zhrnúť do pseudokódu v Algoritme 1. Aby pseudokód mohol fungovať, musíme si dodefinovať pre každé x^i z trénovacej množiny $x_0^i = 1$. Optimalizované θ_j sú pred trénovaním inicializované napríklad náhodne z intervalu $[-0.01, 0.01]$. Premenná y v Algoritme 1 predstavuje klasifikáciu logistickej regresie pre x^i v danej iterácii, η musí byť na konvergenciu metódy dostatočne malé kladné reálne číslo.

Logistická regresia je teda skvelou začiatočnou voľbou na implementáciu do systémov Generali Poisťovňa, a. s.. Interpretácia parametrov θ je podobná ako v pôvodnom systéme, keďže β_k na základe (2) je tiež len akousi lineárnou kombináciou. Na druhej strane logistická regresia narozdiel od Bayesovho klasifikátora nevie štandardne uprednostniť minimalizáciu chyby druhého druhu. Kvôli týmto skutočnostiam sme sa rozhodli využiť základy z týchto dvoch metód na vytvorenie ad-hoc metódy na mieru šitej potrebám Generali Poisťovňa, a. s., ktorú predstavujeme v Kapitole 2.2.6.

Algoritmus 1 Logistická regresia - trénovanie

repeat
  for $j = 0, \ldots, k-1$ **do**
    $\Delta\theta_j \leftarrow 0$
  end for
  for $i = 0, \ldots, r$ **do**
    $suma \leftarrow 0$
    for $j = 0, \ldots, k-1$ **do**
      $suma \leftarrow suma + \theta_j x_j^i$
    end for
    $y \leftarrow g(suma)$
    for $j = 0, \ldots, k-1$ **do**
      $\Delta\theta_j \leftarrow \theta_j + (y^i - y)x_j^i$
    end for
  end for
  for $j = 0, \ldots, k-1$ **do**
    $\theta_j \leftarrow \theta_j + \eta\Delta\theta_j$
  end for
until konvergencia

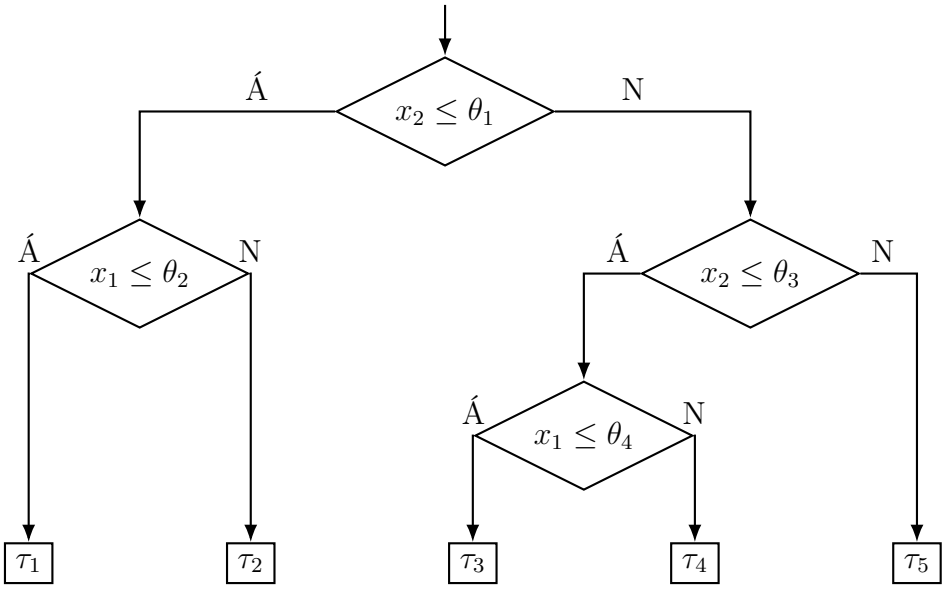
2.2.4 Klasifikačné stromy

Metódy rozhodovacích stromov (*angl. decision trees*) sú napriek svojej konceptuálnej jednoduchosti veľmi silným a bežne používaným nástrojom na klasifikáciu a regresiu. Vo svojej podstate sú v [8] opísané ako sekvencia otázok, kde v každom kroku záleží typ položenej otázky od predošlých odpovedí. Táto sekvencia sa ukončí jasným určením skupiny y_i pre daný bod čŕt v prípade klasifikačných stromov alebo predikovanou hodnotou spojitej modelovanej premennej v prípade regresných stromov. Koncepcia rozhodovacích stromov je najjednoduchšie pochopiteľná na grafe v Obr. 2 prevzatom z [8], zobrazujúcom hierarchický vzťah jednotlivých otázok. V grafe sú otázky zobrazené ako štvoruholníky a odpovede na dané otázky, ktoré zároveň určujú k akej ďalšej otázke daná odpoveď smeruje, ako šípky. Keďže analogická štruktúra sa v teórií grafov už dávno volá strom, všetky body sekvencie, v ktorých sa pýtame otázky, alebo v ktorých sekvencia končí, sú v literatúre nazývané aj uzly (*angl. nodes*) a odpovede na otázky sú označované vetvy (*angl. splits, branches*). Každá otázka musí byť položená tak, aby mala len binárne odpovede „Á''-áno alebo „N''-nie. Otázky na spojitú (kvantitatívnu) črtu sú typu porovnania jej hodnoty s parametrom θ_i. Ak je hodnota črty menšia ako θ_i, pokračujeme stromom cez vetvu „Á" a naopak, ak je väčšia ako θ_i, pokračujeme cez vetvu „N". Pri kategorických (kvalitatívnych) premenných s konečným počtom možných hodnôt sa pýtame, či črta x_i vektora čŕt x patrí do určitej množiny. Uzol pre črtu „značka vozidla"sa napríklad môže pýtať, či je značka vozidla z množiny $\{Volvo, Volkswagen\}$. Pre účely vysvetlenia princípov rozhodovacích stromov budeme objasňovať pre jednoduchosť len klasifikačné stromy a spojité črty.

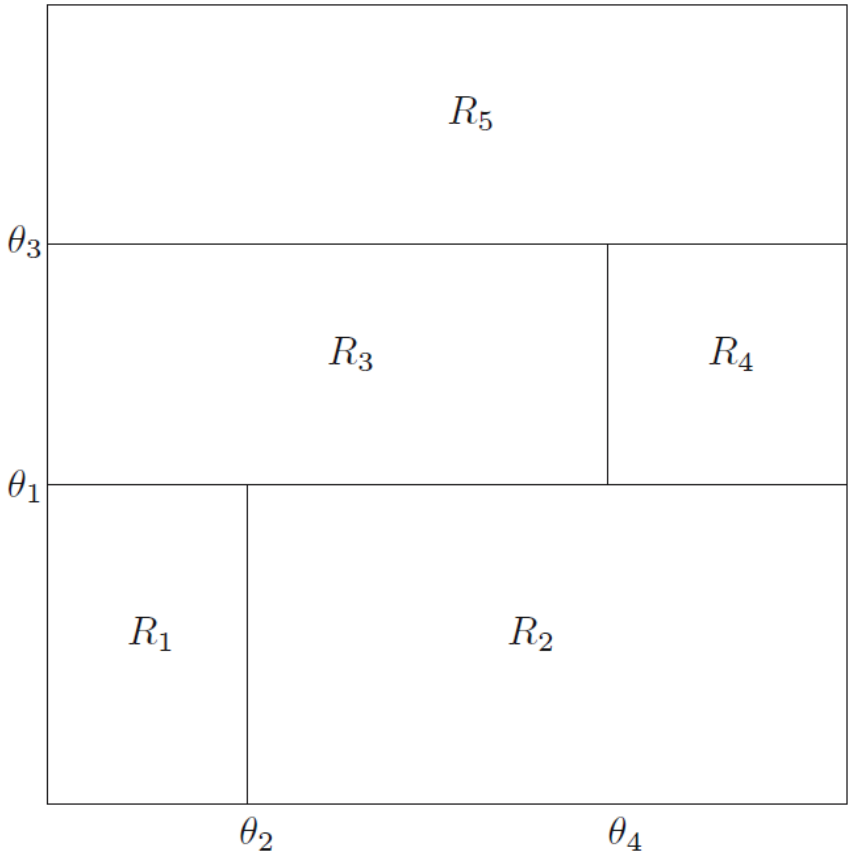
V súvislosti s rozhodovacími stromami sa v [8] spomínajú nasledovné bežne zaužívané pojmy:

- **koreňový uzol** (*angl. root node*) je uzol na vrchu rozhodovacieho stromu, teda uzol obsahujúci prvú zo sekvencie otázok, na ktorú potrebujeme poznať odpoveď pre každý bod čŕt z trénovacej množiny alebo každý nový bod, ktorý chceme klasifikovať,

- **terminálny uzol** τ (*angl. terminal node*) neobsahuje otázku, lebo sa v ňom sekvencia končí, teda sa už ďalej nerozdeľuje v grafe nerozdeľuje, každý takýto uzol



Obr. 2: Príklad stromového grafu klasifikačného stromu



Obr. 3: Príklad rozdelenia roviny na rozhodovacie regióny

musí byť jasne klasifikovaný do jednej zo skupín y_i, v strome sa môže nachádzať viac terminálnych uzlov klasifikovaných do jednej skupiny,

- **rodičovský uzol** v (*angl. parent node*) inak volaný aj neterminálny (*angl. non-terminal node*) obsahuje otázku a ďalej sa binárne rozdeľuje, konkrétny rodičovský uzol označujeme v_i, kde index i je zhodný s indexom parametra θ obsiahnutého v otázke,

- **dcérsky uzol** (*angl. daughter node*) môže byť každý taký uzol, či už terminálny alebo neterminálny, ktorý nie je koreňovým uzlom,

- **rozdelenie dát** (*angl. partition of the data*) je množina všetkých terminálnych uzlov.

Obr. 3, prevzatý z [8], zobrazuje rozdelenie $\mathbb{R}^2$ natrénovaným stromom v Obr. 2, pričom x-ová os predstavuje črtu x_1 a y-ová os predstavuje črtu x_2. Ako popisuje [6], rovina je rozdelená na regióny, kde každý región $R_1, \ldots, R_5$ zodpovedá určitému terminálnemu uzlu $\tau_1, \ldots, \tau_5$. Pre $n = 2$, teda v dvojrozmernom priestore, predstavujú regióny neprekrývajúce sa obdĺžniky, pričom na každý sa z pohľadu predikcie skupiny y môžeme pozerať ako na homogénny. Strany obdĺžnikov sú rovnobežné s osami priestoru čŕt.

Teraz si predstavme, že máme natrénovaný strom a chceme pomocou neho klasifikovať nový vektor čŕt x^p. V tom prípade začneme pri koreňovom uzle, postupne odpovedáme na otázky v rodičovských uzloch až dokým neprídeme k terminálnemu uzlu. Následne x^p klasifikujeme podľa skupiny, do ktorej je terminálny uzol zaradený. To, v akej sekvencii odpovedáme na otázky, a ku ktorému terminálnemu uzlu sa v konečnom dôsledku dostaneme záleží len na našich jednotlivých odpovediach „Á" alebo „N". Do koreňového uzla vstupujú všetky dáta a následne dáta vyhovujúce podmienke v otázke prechádzajú ďalej ľavou vetvou „Á", ostatné prechádzajú k ďalším uzlom pravou vetvou „N". Konkrétny opis klasifikácie zo stromu v Obr. 2 je uvedený v [8]. Tento strom je natrénovaný klasifikovať dvojrozmerné črty $x = (x_1, x_2)^T$. Všetky možnosti klasifikácie x^p sú pre jednotlivé uzly nasledovné:

- v_1: spýtame sa, či je $x_2 \leq \theta_1$, ak je odpoveď áno, pokračujeme ľavou vetvou „Á", ak je odpoveď nie, pokračujeme pravou vetvou „N",

- υ_2: ak je odpoveď na υ_1 „Á", spýtame sa, či je $x_1 \leq \theta_2$, ak odpovieme znova „Á", klasifikujeme x^p podľa klasifikácie uzla τ_1 s korešpondujúcim regiónom $R_1 = \{x_1 \leq \theta_2, x_2 \leq \theta_1\}$, ak odpovieme „N", klasifikujeme x^p podľa klasifikácie uzla τ_2 s korešpondujúcim regiónom $R_2 = \{x_1 > \theta_2, x_2 \leq \theta_1\}$,

- υ_3: ak je odpoveď na υ_1 „N", spýtame sa, či je $x_2 \leq \theta_3$, ak odpovieme „Á", pokračujeme do uzla υ_4, ak je odpoveď „N", klasifikujeme x^p podľa klasifikácie uzla τ_5 s korešpondujúcim regiónom $R_5 = \{x_2 > \theta_3\}$,

- υ_4: ak je odpoveď na υ_1 „N", a ak je odpoveď na υ_3 „Á", spýtame sa, či je $x_1 \leq \theta_4$, ak odpovieme „Á", klasifikujeme x^p podľa klasifikácie uzla τ_3 s korešpondujúcim regiónom $R_3 = \{x_1 \leq \theta_4, \theta_1 < x_2 \leq \theta_3\}$, ak odpovieme „N", klasifikujeme x^p podľa klasifikácie uzla τ_4 s korešpondujúcim regiónom $R_4 = \{x_1 > \theta_4, \theta_1 < x_2 \leq \theta_3\}$,

Otázkou teraz je, ako zostrojíme klasifikačný strom, ktorý čo najlepšie klasifikuje záznamy z trénovacej množiny a nové záznamy po trénovaní. Táto otázka v sebe zahŕňa výber čŕt x_i, na ktoré sa budeme pýtať, zvolenie θ_j, s ktorými budeme črty porovnávať, určenie, kedy uzol prehlásime za terminálny, zvolenie optimálnej sekvencie otázok a priradenie každého terminálneho uzla do skupiny. Procedúra konštrukcie stromu je podľa [15] nasledovná:

1. vytvor nový uzol, ktorý je buď koreňový, alebo je dcérsky uzol neterminálneho uzla,

2. rozhodni, či má byť tento uzol terminálny, alebo neterminálny,

3. ak je terminálny, priraď mu skupinu,

4. ak je neterminálny, priraď mu vhodnú otázku,

5. ak existuje neterminálny uzol bez dcérskych uzlov, opakuj postup.

Najťažšia časť tohto algoritmu je bod 4. V tomto bode je našou úlohou zo všetkých možných otázok v danom uzle tú najlepšiu. Nech y_1 a y_2 sú naše dve skupiny, teda buď je udalosť podvodom, alebo nie. Pre uzol υ (rovnako aj pre terminálne uzly τ)

definujeme podľa [8] funkciu nečistoty uzla ako:

$$i(\upsilon) = \phi(p(1|\upsilon), p(2|\upsilon)), \tag{32}$$

kde $p(i|\upsilon)$ je odhadom podmienenej pravdepodobnosti $p(x \in y_i|\upsilon)$, teda pravdepodobnosti, že ak vektor čŕt x spadá do uzla υ, je v skupine y_i. Od ϕ v (32) požadujeme, aby bola symetrická, mala minimum v bodoch najväčšej „čistoty" $(1, 0)$ a $(0, 1)$, a aby mala maximum v bode najväčšej „nečistoty" $(\frac{1}{2}, \frac{1}{2})$. Inak povedané, uzol označujeme za maximálne „čistý", ak všetky x^i, ktoré do neho spadajú, majú rovnakú skupinu. Naopak, uzol označujeme za maximálne „nečistý", ak polovica x^i, ktorá do neho spadá má skupinu y_1 a druhá polovica skupinu y_2. Bežne používaných funkcií nečistoty ϕ je na základe [8] a [15] viacero. Prvou takou funkciou je entropia (*angl. entropy function*), ktorá ma pre prípad dvoch skupín tvar:

$$i(\upsilon) = -p(1|\upsilon) \log p(1|\upsilon) - (1 - p(1|\upsilon)) \log(1 - p(1|\upsilon)). \tag{33}$$

Druhou často spomínanou funkciou je Giniho index (*angl. Gini diversity index*):

$$i(\upsilon) = 2p(1|\upsilon)(1 - p(1|\upsilon)). \tag{34}$$

Obe funkcie obsahujú len $p(1|\upsilon)$, lebo $1 = p(1|\upsilon) + p(2|\upsilon)$. Pre praktické použitie je medzi týmito dvoma funkciami len malý rozdiel, no častejšie sa v praxi využíva Giniho index. Ak sa do uzla υ rozhodneme dať otázku $x_i \leq \theta_j$, vytvoríme tým uzlu υ dva dcérske uzly υ_A, υ_N. Všetky x spadajúce do uzla υ, pre ktoré odpovieme na otázku $x_i \leq \theta_j$ áno, teda „Á", budú pokračovať ľavou vetvou do uzla υ_A, ostatné budú pokračovať pravou vetvou do uzla υ_N. Následne si vieme pomocou [8] spraviť Tabuľku 2, teda takzvanú kontingenčnú tabuľku klasifikácie (*angl. confusion matrix*):

	1	0	Suma
$x_i \leq \theta_j$	n_{11}	n_{12}	n_{1+}
$x_i > \theta_j$	n_{21}	n_{22}	n_{2+}
Suma	n_{+1}	n_{+2}	n_{++}

Tabuľka 2: Kontingenčná tabuľka klasifikácie pre uzol υ

Tabuľka 2 je kontingenčnou tabuľkou, ktorá sumarizuje rozdelenie x spadajúcich do uzla υ po aplikovaní otázky $x_i \leq \theta_j$. Počet takých x, ktoré patria do skupiny

$y_1 = 1$ a zároveň pre ne na otázku $x_i \leq \theta_j$ odpovedáme „Á", je v tabuľke označený ako n_{11}. Analogicky to platí pre n_{12}, n_{21} a n_{22}. Všetky n obsahujúce v dolnom indexe znak „+"sú súčtom počtov v riadku alebo stĺpci tabuľky. Napríklad počet všetkých x patriacich do skupiny y_1 a spadajúcich do uzla v označíme n_{+1} a počet všetkých x, pre ktoré odpovedáme „N" a spadajú do uzla v označíme n_{2+}. Nakoniec počet všetkých x spadajúcich do uzla v označíme n_{++}. Pomocou týchto počtov vieme vyčísliť odhady $p(1|v) = \frac{n_{+1}}{n_{++}}$ a $p(2|v) = \frac{n_{+2}}{n_{++}}$ a následne aj hodnotu funkcie nečistoty pre uzol v:

$$i(v) = -\left(\frac{n_{+1}}{n_{++}}\right) \log\left(\frac{n_{+1}}{n_{++}}\right) - \left(\frac{n_{+2}}{n_{++}}\right) \log\left(\frac{n_{+2}}{n_{++}}\right). \tag{35}$$

Nakoľko do dcérskeho uzla v_A spadajú všetky x z uzla v, pre ktoré platí $x_i \leq \theta_j$, je ich celkový počet n_{1+}. Vďaka tomu vieme analogicky pomocou tej istej Tabuľky 2 vyčísliť hodnotu funkcie nečistoty aj pre dcérske uzly:

$$i(v_A) = -\left(\frac{n_{11}}{n_{1+}}\right) \log\left(\frac{n_{11}}{n_{1+}}\right) - \left(\frac{n_{12}}{n_{1+}}\right) \log\left(\frac{n_{12}}{n_{1+}}\right). \tag{36}$$

$$i(v_N) = -\left(\frac{n_{21}}{n_{2+}}\right) \log\left(\frac{n_{21}}{n_{2+}}\right) - \left(\frac{n_{22}}{n_{2+}}\right) \log\left(\frac{n_{22}}{n_{2+}}\right). \tag{37}$$

Všetky tri rovnice (35), (36) a (37) využívajú entropiu. Pomocou týchto troch rovníc a [15] zadefinujeme, o čo sa zlepší čistota celého systému (*angl. goodness of split*), keď v rozdelíme na v_A a v_N:

$$\Delta i = i(v) - p_A i(v_A) - p_N i(v_N). \tag{38}$$

kde $p_A = \frac{n_{1+}}{n_{++}}$ je odhad pravdepodobnosti, že x spadajúci do v následne spadne aj do v_A a $p_N = \frac{n_{2+}}{n_{++}}$ je odhad pravdepodobnosti, že x spadajúci do v následne spadne aj do v_N. Najlepšou otázkou pre črtu x_i je teda tá s najväčšou hodnotou Δi. Keďže pre kvantitatívnu črtu x_i môže byť takýchto otázok nekonečne veľa, množinu hodnôt črty diskretizujeme na konečný počet n_i hodnôt. Následne vyberieme tú otázku spomedzi $x_i \leq \theta_j^1, \ldots, x_i \leq \theta_j^{n_i}$, ktorá má najvyššiu hodnotu Δi.

Bod 2 procedúry konštrukcie stromu, teda určenie, či je nový pridaný uzol koncovým, alebo nie, môže mať viacero prístupov. Prvým je jednoduché určenie maximálnej hĺbky stromu, teda maximálny počet vrstiev uzlov. Ak po vytvorení nového uzlu dosiahneme požadovaný maximálny počet uzlov, prehlásime všetky uzly, ktoré nie sú rodičovskými za terminálne. V druhom postupe zohľadňujeme len čistotu uzla. Buď uzol prehlásime

za terminálny, len ak je úplne čistý, teda ak do neho spadajú x patriace do jednej a tej istej skupiny, alebo ho prehlásime za terminálny, ak čistota prekročí určitú hranicu. Najefektívnejšou metódou sa ukazuje byť úplne iný prístup, ktorým je osekávanie (*angl. pruning*). Jedná sa o komplexnú metódu, preto pre účely tejto práce predstavujeme len jej myšlienky. Základnou myšlienkou osekávania je najprv vytvoriť strom obsahujúci čo najväčší počet uzlov T_{max} a následne z neho vybrať optimálny podstrom. Podstrom je pôvodný strom zmenšený o určitý počet uzlov smerom od terminálnych uzlov. Dôležitým pojmom pri osekávaní je chyba klasifikácie. Chyba klasifikácie $R(\cdot)$ predstavuje určitú metriku, nakoľko sme nespokojný s tým, ako uzol alebo celý strom klasifikuje črty x. Čím je $R(\cdot)$ menšia, tým je kvalita uzla alebo stromu vyššia. Resubstitučný odhad $R(\cdot)$ označíme $R^{re}(\cdot)$. Pre potreby orezávania nepoužívame $R^{re}(\cdot)$, ale nekvalitu stromu $R_\alpha(\cdot)$, ktorá okrem chyby klasifikácie zohľadňuje aj jeho komplexitu určenú počtom uzlov v strome. Nekvalitu stromu teda vypočítame pomocou vzťahu:

$$R_\alpha(T) = R^{re}(T) + \alpha n_T, \tag{39}$$

kde n_T predstavuje počet uzlov v strome T a α je voliteľný parameter nazývaný v literatúre parameter komplexity (*angl. complexity parameter*). Parameter α má veľký vplyv na to, ktorý podstrom je optimálny. Ak zvolíme $\alpha = 0$, penalizácia za veľkosť stromu sa stráca, teda optimálnym je stále T_{max}. Naopak, ak zvolíme za α dostatočne veľké α_m, optimálny bude podstrom T_0 obsahujúci len koreňový uzol. Pre akúkoľvek hodnotu medzi 0 a α_m parametra komplexity môže byť optimálnym iný podstrom. Algoritmus osekávania je podľa [8] a [15] nasledovný:

1. vytvoríme strom T_{max} s čo najväčším počtom uzlov,
2. zvolíme K hodnôt parametra α, $0 = \alpha_1 \leq \cdots \leq \alpha_K = \alpha_m$,
3. pre každú hodnotu $\alpha_1, \ldots, \alpha_K$ nájdeme optimálny podstrom $T_{max} = T_1^* \leq \cdots \leq T_K^* = T_0$, optimálny podstrom je ten, ktorý má pre určité α minimálnu $R_\alpha(\cdot)$,
4. spomedzi $T_1^* \leq \cdots \leq T_K^*$ vyberieme najlepší pomocou krížovej validácie, teda pomocou zohľadnenia jeho chyby klasifikácie na záznamoch čŕt, ktoré neboli použité na trénovanie.

Všetky ďalšie podrobnosti o osekávaní, o iných prístupoch k osekávaniu a celkovo o teórii rozhodovacích stromov sú napríklad v [8]. Pojem krížová validácia je objasnený v Kapitole 2.4.

Minimalizáciu chyby druhého druhu môžeme pri rozhodovacích stromoch dosiahnuť zmenením kontingenčnej Tabuľky 2. Každému záznamu $x^p, p = 1, \ldots, r$ z trénovacej množiny pridelíme váhu w^p, pričom pre naše potreby majú záznamy pochádzajúce z rovnakej skupiny i rovnakú váhu w_i. Počty záznamov $n_{11}, n_{12}, n_{21}, n_{22}$ zameníme za súčty váh záznamov $w_{11}, w_{12}, w_{21}, w_{22}$. Takto upravená Tabuľka 2 teda sumarizuje súčty váh w^p spadajúcich do uzla v po aplikovaní otázky $x_i \leq \theta_j$. Všetky w obsahujúce v dolnom indexe znak „+"definujeme rovnakou logikou ako v pôvodnej Tabuľke 2. Vážená funkcia nečistoty pre uzol v má potom tvar:

$$i^w(v) = -\left(\frac{w_{+1}}{w_{++}}\right) \log\left(\frac{w_{+1}}{w_{++}}\right) - \left(\frac{w_{+2}}{w_{++}}\right) \log\left(\frac{w_{+2}}{w_{++}}\right). \tag{40}$$

Na uprednostnenie minimalizácie chyby druhého druhu stačí nastaviť pomer váh $\frac{w_1}{w_2} > 1$, kde w_1 predstavuje váhu pridelenú všetkým poistným udalostiam, ktoré sú podvodmi $y^p = 1$ a w^2 predstavuje váhu pridelenú všetkým poistným udalostiam, ktoré podvodmi nie sú $y^p = 0$.

Pre zhrnutie, prvou veľkou výhodou rozhodovacích stromov ako celku je ich interpretovateľnosť. Logika, ktorou rozhodovacie stromy predikujú je veľmi podobná ľudskému zmýšľaniu, prípadne zmýšľaniu profesionálov v rôznych odboroch. Druhou nepochybne veľkou výhodou je použiteľnosť v praxi, kde na predikovanie z natrénovaného stromu niekedy stačí binárny graf stromu vytlačiť na papier. To platí samozrejme len pre menší počet nových dát, no pre rýchle určenie napríklad rizikovosti jednej poistnej udalosti stačí, ak pracovník poisťovne nasleduje sekvenciu otázok uvedenú na papieri. Z výpočtového hľadiska sú rozhodovacie stromy veľmi nenáročné a trénovanie aj na väčších počtoch dát trvá väčšinou maximálne pár sekúnd. Vďaka týmto dôvodom sme sa rozhodli použiť rozhodovacie stromy pre účely predikovania rizikovosti poistných udalostí v poisťovni Generali Poisťovňa, a. s.. Ich interpretovateľnosť je taká intuitívna, že napriek ich odlišnostiam od pôvodného systému opísanému v Kapitole 1.1.2, vie každý pracovník spojiť ich používanie so svojou expertízou. Všetky tieto výhody majú ale jeden háčik. Ak je uzlov v strome príliš veľa, ich interpretovateľnosť sa stáva otáznou rovnako ako ľahká použiteľnosť. Tento problém bližšie rozoberáme v Kapitole 3.

2.2.5 Umelé neurónové siete, metódy oporných bodov

Umelé neurónové siete (*angl. artificial neural networks*) a metódy oporných bodov (*angl. support vector machines, SVM*) spomenieme len stručne. Obe metódy, no najmä neurónové siete, dokážu riešiť veľmi zložité problémy a sú základným kameňom dnešného rozvoja umelej inteligencie. Napriek tomu, že SVM veľmi dobre fungujú s menším počtom dát a neurónové siete zažívajú v poslednej dobe veľký rozmach, sme sa rozhodli ich pre potreby Generali Poisťovňa, a. s. nepoužiť. Základným dôvodom je ich povaha takzvanej „čiernej skrinky". Takéto metódy obsahujú veľké množstvo parametrov, ktoré nie sú dobre interpretovateľné. Tieto parametre majú skôr inžiniersky charakter a ich konkrétna hodnota nás v podstate pri trénovaní, ani po ňom, nezaujíma. Ako sme v Kapitole 1 spomínali, Generali Poisťovňa, a. s. disponuje množstvom odborníkov z oblasti odhaľovania podvodov s dlhoročnými skúsenosťami, no nemá skúsenosti so strojovým učením. Našim cieľom je zlepšiť pôvodný systém aj tým, že strojové učenie a experti spolu dokážu spolupracovať ako tím pri hľadaní podvodov. Toto ale vieme dosiahnuť, len ak sú metódy strojového učenia dobre interpretovateľné a z hodnôt ich parametrov je jasné, prečo určitú udalosť označili ako rizikovú. Zároveň vďaka interpretovateľnosti existuje určitá forma kontroly tesne po natrénovaní modelu, či nevykazuje anomálie. Pri metódach, ktoré sú „čiernymi skrinkami" žiaľ toto nie je možné a treba počkať na výsledky reálneho testovania.

Neurónové siete sú univerzálnym aproximátorom, teda ich rozhodovacia hranica môže aproximovať akýkoľvek tvar. Na správne natrénovanie vyžadujú často väčší počet dát, ktoré ale, ako spomíname v Kapitole 1.2.1, nemáme k dispozícii. V konečnom dôsledku ich použitie do budúcnosti nevylučujeme, no je potrebné, aby si v prvom rade prostredie poisťovne zvyklo na použitie iných metód, ako expertného odhadu. V tom môže pomôcť namieru šitá ad-hoc metóda, ktorú predstavujeme v nasledujúcej Kapitole 2.2.6, alebo veľmi intuitívna metóda ako rozhodovacie stromy, ktoré predstavujeme v Kapitole 2.2.4.

2.2.6 Ad-hoc prístup

Ako spomíname v Kapitole 1, kvôli povahe poistného prostredia máme na použité metódy strojového učenia viaceré požiadavky. Najdôležitejšou je umožnenie spojenia

expertízy zamestnancov s interpretovateľnosťou pre maximálne zlepšenie pôvodného systému. Ďalšími, už viac technickými, sú jednoduché použitie v praxi aj bez zabehnutého oddelenia špecializovaného na strojové učenie a minimalizácia chyby druhého druhu. Doteraz sme v Kapitole 2 označili za najlepšiu metódu na aplikáciu do praxe rozhodovacie stromy. Napriek im nesporným výhodám si ale myslíme, že interpretovateľnosť a použiteľnosť vieme posunúť na ešte vyššiu úroveň vytvorením ad-hoc metódy presne šitej na mieru potrebám poisťovne.

Aby bol prechod na nový systém čo najhladší, táto metóda má za úlohu zachovať maximum z pôvodného systému a aplikovať naň metódy strojového učenia v čo najmenšej, no stále efektívnej miere. To dosiahneme zachovaním systému identifikátorov $i = 1, \ldots, t$ a hodnôt $j = 1, \ldots, m_i$ spomenutých v Kapitole 1.1.2. Kvôli tomu ad-hoc metóda v sebe stále zahŕňa určitú formu použitia expertných skúseností na výber čŕt x a na rozdelenie týchto čŕt do hodnôt. Zmena oproti pôvodnej metóde nastáva pri určovaní váh, ktoré namiesto expertného odhadu natrénujeme na trénovacej vzorke dát. Označenie váh podľa ich príslušnosti k identifikátoru a jeho hodnote je v rovnakej štruktúre ako v Tabuľke 1. Zároveň sčasti prepoužijeme výpočet rizikovosti (2) z pôvodného systému, ktorý sme pre k-ty záznam čŕt vypočítali ako:

$$\beta_p = \sum_{i=1}^{t} \sum_{j=1}^{m_i} I_{i,j}^p \theta_{i,j}. \tag{41}$$

Pripomenieme, že $I_{i,j}^p$ má hodnotu 1, ak i-ta zložka p-tého vektora čŕt x_i^p spadá do hodnoty j. Pod hodnotou j rozumieme pre kvantitatívnu črtu určitý interval a pre kvalitatívnu črtu množinu. Oproti rovnici (2) sme v 41 zmenili označenie váh z $w_{i,j}$ na $\theta_{i,j}$, aby sme reflektovali označenie z predošlých metód Kapitoly 2. Zatiaľ teda máme na optimalizáciu určených k parametrov $\theta_{i,j}$, no stále nemáme funkciu, ktorú by sme optimalizovali. Premennú β_p sme v Kapitole 1.1.2 označili za „rizikovosť", no tento pojem sa nezhoduje s rizikovosťou spomínanou v Kapitole 2, ktorá označuje pravdepodobnosť príslušnosti x^p do skupiny y^1, teda nadobúda hodnoty z intervalu $(0, 1)$. Ak chceme β transformovať na pravdepodobnosť, musíme najprv zaviesť nový parameter θ_0, ktorý je v tomto modeli vlastne rozhodovacou hranicou. Ak vektor čŕt x_p má hodnotu β_p v intervale $(-\infty, \theta_0)$, klasifikujeme ho do skupiny y_2, teda vylúčime, že je podvodom. Ak vektor čŕt x_p má hodnotu β_p v intervale $[\theta_0, \infty)$, klasifikujeme ho do skupiny y_1, teda ho označíme za podvod. Namiesto porovnania β_p s θ_0 následne porovnávame $(\beta_p - \theta_0)$

s 0, aby sme pomocou sigmoidálnej funkcie vedeli dostať výstupné hodnoty modelu do intervalu $(0, 1)$. Po týchto úpravách aproximujeme skupinu y^p vektora čŕt x^p ako:

$$\hat{y}^p = g(\beta_p - \theta_0) = \frac{1}{1 + \exp(\beta_p - \theta_0)}. \tag{42}$$

Pre skompletizovanie metódy potrebujeme vybrať chybovú funkciu $J(\cdot)$ a nájsť spôsoby, ktorými ju budeme optimalizovať. Pri chybovej funkcii je našou hlavnou požiadavkou uprednostnenie minimalizácie chyby druhého druhu. Tu si pomôžeme teóriou Bayesovho klasifikátora 2.2.1, teda konkrétne princípmi zostavovania priemerného rizika ρ. Pomocou nej v chybovej funkcii oddelíme chyby prvého a druhého druhu a pridelíme im koeficienty straty. Situáciu nám výrazne uľahčuje skutočnosť, že pracujeme len s dvomi skupinami $y_1 = 1$ a $y_2 = 0$. Ako vysvetľujeme v Kapitole 1.2.2, chyba prvého druhu znamená prehlásenie udalosti x^p za podvod, ak je v skutočnosti v poriadku a chyba druhého druhu znamená prehlásenie, že je udalosť x^p v poriadku no v skutočnosti je podvodom. Vektor čŕt x^p prehlásime za podvod, teda ho klasifikujeme do y_1, ak je $\hat{y}^p \geq \frac{1}{2}$. Vďaka trénovacej množine ale poznáme skutočnú hodnotu $y^p = \{0, 1\}$. Chyba prvého druhu teda jednoducho nastáva, ak $\hat{y}^p \geq \frac{1}{2}$ a $y^p = 0$ a chyba druhého druhu nastáva ak $\hat{y}^p < \frac{1}{2}$ a $y^p = 1$. Nakoniec metódu „potrestáme" za neistotu pri klasifikovaní, teda prípadu $y^p = 1$ a $\hat{y}^p = 0.6$ pridelíme väčšiu chybu ako $\hat{y}^p = 0.9$. Chybová funkcia ad-hoc metódy, ktorú nazývame asymetrický súčet štvorcov, má pre konkrétne hodnody $\theta = (\theta_{1,1}, \ldots, \theta_{t,m_i}, \theta_0)^T$ tvar:

$$J(\theta) = \frac{1}{r}(c_{12} \sum_{p, y^p < \hat{y}^p} (y^p - \hat{y}^p)^2 + c_{21} \sum_{p, y^p > \hat{y}^p} (y^p - \hat{y}^p)^2), \tag{43}$$

kde t je počet identifikátorov, teda počet zložiek vektora čŕt x, m_i je počet hodnôt identifikátora i a r je počet dát v trénovacej množine $T = (x^1, y^1), (x^2, y^2), \ldots, (x^r, y^r)$. Vďaka menovateľu obsahujúcemu r je chybová funkcia priemernou chybou na jeden bod trénovacej množiny, teda môžeme medzi sebou jednoduchšie porovnať inštancie modelu natrénované na rôznych trénovacích množinách. Voliteľné parametre c_{12} a c_{21} predstavujú spomínané koeficienty straty. Aby metóda uprednostnila minimalizáciu chyby druhého druhu, musí platiť:

$$c_{21} > c_{12} > 0, \tag{44}$$

pretože tým pádom všetky chyby druhého druhu budú prispievať k celkovej strate $\frac{c_{21}}{c_{12}}$-krát viac, ako chyby prvého druhu. V sumách sú štvorce zátvoriek z dôvodu vyhladenia

funkcie, čím dosiahneme lepšie fungovanie optimalizačných metód. Optimálne θ dostaneme minimalizáciou funkcie straty. Na minimalizáciu sme sa rozhodli použiť dvojicu známych optimalizačných metód. Najprv využívame metódu Nelder-Mead na rýchlu aproximáciu minima chybovej funkcie v zložitom 50+ rozmernom priestore. Následne aproximáciu doladíme kvázi-Newtonovou BFGS metódou. Nelder-Mead je simplexová metóda, ktorá využíva pri optimalizácii len funkčné hodnoty, preto je veľmi rýchla. Naopak BFGS je zlepšením klasickej Newtonovej metódy a v každej iterácii aproximuje Hessovu maticu, preto má presné, no pomalé kroky. Obe metódy sú často používané a všeobecne známe s bližším popisom napríklad v [10] a [11].

2.3 Zhlukovanie

Pre potreby poisťovne sme sa rozhodli použiť, z dôvodov spomenutých v Kapitole 1.2.3, aj metódy s učením bez učiteľa. Trénovacia množina T teda v prípade týchto metód neobsahuje dvojice (x^p, y^p), ale len príklady vektora čŕt x^p. Príslušnosti vektorov čŕt do skupín y_1 alebo y_2 sú následne použité až po aplikovaní zhlukovania. V postupe uvedenom v 1.2.3 uvádzame, že každú poistnú udalosť zaradíme do skupiny s čo najpodobnejšími záznamami. Na otázku, ktoré záznamy považujeme za najpodobnejšie, odpovedá každá metóda zhlukovania inak. Pre jednoduchosť, aby sme v prvom rade overili použiteľnosť postupu na odhaľovanie poistných podvodov, sme sa rozhodli použiť známu metódu K-means.

V K-means podľa [5] definujeme podobnosť záznamov ako ich druhú mocninu vzdialenosti v priestore. Podobnosť dvoch poistných udalostí s vektormi čŕt x^p a x^q teda hodnotíme podľa:

$$(x^p - x^q)^T(x^p - x^q) = \sum_{t=1}^{k-1}(x_t^p - x_t^q)^2, \tag{45}$$

kde x_t^p je t-ta zložka p-tého vektora čŕt $x^p \in \mathbb{R}^{k-1}$ z trénovacej množiny. Ak je hodnota (45) malá, poistné udalosti považujeme za podobné. Na vytvorenie algoritmu zhlukovania potrebujeme presnejšie zadefinovať, čo vlastne zhluk znamená. V prípade K-means reprezentuje každý zhluk dát určitý bod. Tento reprezentačný bod μ_i je vektorom rovnakej dimenzie ako x. Každý zhluk $i = 1, \ldots, K$ je reprezentovaný samostatným reprezentačným bodom μ_i, ktorý predstavuje ťažisko bodov v zhluku i. Polohu μ_i vieme

podľa [5] vypočítať ako:

$$\mu_i = \frac{\sum_j z_{ij} x^j}{\sum_j z_{ij}}, \tag{46}$$

kde z_{ij} predstavuje binárny identifikátor, ktorý nadobúda hodnotu 1, ak j-ta poistná udalosť spadá do zhluku i, inak $z_{ij} = 0$. Vo vzťahu (46) predpokladáme, že každá poistná udalosť môže spadať len do jedného zhluku, teda $\sum_i z_{ij} = 1$. Každý dátový bod x^p, respektíve poistná udalosť, je pridelená do najbližšieho zhluku i, teda do zhluku s minimálnou hodnotou vzdialenosti:

$$(x^p - \mu_i)^T (x^p - \mu_i). \tag{47}$$

Dostávame sa k opakujúcemu sa postupu, kde najprv zhlukom vyberieme reprezentačné body a následne pridelíme poistné udalosti do najbližších zhlukov. Ak poznáme zhluky $\mu_1, \ldots, \mu_K$, vieme do nich prideliť poistné udalosti, no bez počiatočného pridelenia poistných udalostí nevieme vytvoriť zhluky. Z tohto dôvodu v metóde K-means inicializujeme reprezentačné body zhlukov náhodne. Počet zhlukov K je potrebné zvoliť ešte pred začiatkom trénovania. Podľa [5] je myšlienka iterácie algoritmu zhlukovania metódou K-means nasledovná:

1. pre každú poistnú udalosť x^j nájdeme i, ktoré minimalizuje vzťah (47) a priradíme x^j do daného zhluku i, teda $z_{ij} = 1$ a $z_{lj} = 0$ pre $i \neq l$,
2. ak sú všetky priradenia od predošlej iterácie nezmenené, ukončíme výpočty,
3. aktualizujeme polohu reprezentačných bodov pomocou vzťahu (46),
4. vrátime sa na bod 1.

Takýto postup na základe [5] konverguje do lokálneho minima, no konvergencia do globálneho minima nie je garantovaná. Či metóda skonverguje do globálneho minima, alebo nie, je závislé na inicializácii reprezentačných bodov. Bežným spôsobom na obídenie tohto problému je natrénovanie viacerých K-means modelov s rôznymi inicializáciami a vybratie toho najlepšieho.

Metóda K-means so sebou prináša veľa výhod, pre ktoré sme sa ju rozhodli pre potreby poisťovne použiť. Najdôležitejšími sú ľahká implementácia a rýchly priebeh trénovania aj pri väčšom počte dát a identifikátorov. Všeobecnou nevýhodou je ťažké

určenie správneho počtu zhlukov K, no pre naše potreby je táto skutočnosť zanedbateľná. Ako popisujeme v Kapitole 1.2.3, našim cieľom je rozdeliť poistné udalosti do veľkého počtu zhlukov, aby sme naozaj spojili do rovnakých skupín len tie najpríbuznejšie. Počet zhlukov teda často, rovnako aj v našom prípade, vyplýva z potrieb praxe.

2.4 Testovanie kvality predikcie

Neoddeliteľnou súčasťou každého nami použitého modelu je testovanie prediktability. Ako opisujeme v Kapitole 2.1, pretrénovanie môže byť veľkou prekážkou v praktickom použití modelu. Ak model príliš prispôsobíme trénovacím dátam, kvalita klasifikácia na nových poistných udalostiach môže byť veľmi zlá. Na predídenie tohto problému existuje viacero štandardných metód.

Základom je nepoužiť všetky dostupné dáta na trénovanie, no rozdeliť ich na trénovaciu a testovaciu množinu. Toto rozdelenie nám zaručí, že na testovacej vzorke môžeme vždy overiť kvalitu modelu na dátach, ktoré pri trénovaní nemal k dispozícii. Často používaným spôsobom je dáta rozdeliť náhodne, pričom dodržíme predom zvolený pomer medzi počtom dát v trénovacej a testovacej množine. Trénovacia množina je tvorí štandardne okolo 80% všetkých dát. Dáta do trénovacej vzorky môžeme vybrať aj iným, menej rigoróznym, no viac výpovedným spôsobom. Jednoducho stanovíme dátum, do ktorého všetky poistné udalosti spadajú do trénovacej množiny. Takýto spôsob je akousi „simuláciou praxe", kde simulujeme trénovanie v určitom čase a testovanie výsledkov po pár mesiacoch, čo má pre pracovníkov poisťovne výbornú výpovednú hodnotu.

V našom prípade ale disponujeme menším počtom dát, preto by ďalšie zmenšenie trénovacej množiny mohlo mať veľmi zlé dôsledky na kvalitu modelov. Ako popisuje [4], tento problém čiastočne rieši napríklad K-násobná krížová validácia (*angl. K-fold cross-validation*). Pri nej chybu krížovej validácie CV odhadujeme nasledovným postupom:

1. náhodne rozdelíme dáta do K skupín, skupiny označíme $i = 1, \ldots, K$,
2. vyberieme skupinu i, ktorú sme v tomto bode ešte nevybrali a použijeme ju na testovanie, ostatné použijeme na trénovanie,

3. na trénovacej množine natrénujeme model, na testovacej množine vypočítame testovaciu chybu a označíme ju C_i,

4. ak neboli v bode 2 použité všetky skupiny, vrátime sa na bod 1, inak ukončíme výpočty.

Počet skupín K by nemal byť príliš veľký, bežne sa využíva $K = 5$ alebo $K = 10$. Jednu iteráciu postupu, teda jedno prejdenie bodov $1 - 4$ v postupe, nazývame „inštancia". Odhad testovacej chyby krížovej validácie vypočítame podľa [4] vzťahom:

$$CV = \frac{1}{K} \sum_{i=1}^{K} C_i. \tag{48}$$

k	**Riziko podvodu** β_k	**Je udalosť** k **podvodom?**
1	1	1
2	0.99	1
⋮	⋮	⋮
p	0.68	1
$p+1$	0.65	0
⋮	⋮	⋮
$n-1$	0.05	0
n	0	0

Tabuľka 3: Ideálny výsledok testovania metódy, ak je v n dátach p podvodov

Testovaciu chybu v jednej inštancii môžeme definovať rôznym spôsobom. Bežne sa na tieto účely používa chyba klasifikácie, teda percentuálny podiel zle klasifikovaných bodov. Dôležitejším pre výber najlepšieho modelu bolo pre nás, ako vysvetľujeme v predošlých kapitolách, porovnanie s ideálnym prípadom zoradenia udalostí, ktorý je uvedený v Tabuľke 3. Najprv sme udalosti zoradili zostupne podľa rizikovosti β_k, ktorú modelmi predikujeme. Pri n poistných udalostiach, ktoré reálne obsahujú p podvodov sme sledovali podiel podvodov medzi najrizikovejšími $p*10\%$, $p*25\%$ a $p*50\%$ prípadmi. V ideálnom prípade by všetky tieto tri skupiny obsahovali len skutočné podvody. Takýto spôsob testovania odzrkadľuje reálnu prácu zamestnancov pri manuálnej kontrole.

Jednoducho im odpovedá na otázky, koľko skutočných podvodov môžu očakávať medzi určitým počtom prvých manuálnych kontrol.

Za najkvalitnejší model teda považujeme ten, ktorý má testovaciu chybu krížovej validácie najmenšiu. Za testovaciu chybu v jednej inštancii považujeme pomer udalostí, ktoré nie sú podvodmi medzi prvými $p*10\%$, $p*25\%$ alebo $p*50\%$ prípadmi. Od metód so špecifickým použitím ako K-means zhlukovanie uprednostňujeme minimálnu chybu medzi prvými $p*10\%$ najrizikovejšími prípadmi, pri ad-hoc metóde alebo klasifikačných stromoch medzi prvými $p * 50\%$ prípadmi.

3 Aplikácia

V praktickej časti prechádzame k predstaveniu samotných výsledkov práce. Každú metódu sme naprogramovali, na získaných dátach otestovali a ich úspešnosť porovnali s pôvodným systémom. Na účely programovania zvolených metód sme použili programovací jazyk MATLAB. Tento jazyk sme zvolili hlavne vďaka jeho širokej škále knižníc a do budúcnosti aj rýchlej práci s väčším počtom dát. Dáta sme upravovali do požadovanej podoby v programe Microsoft Excel. Metódy sme aplikovali, ako spomíname v Kapitole 1.2.1, len na scenár 1 kvôli nedostatku dát pre ostatné scenáre. Niektoré identifikátory sme sa rozhodli nepoužiť ako črty na trénovanie. Jeden identifikátor obsahoval príliš veľa hodnôt, teda optimalizačný problém robil zbytočne náročnejším na výpočty. Pri príliš veľkom počte parametrov θ by bola množina hypotéz H zložitejšia, čím by sa hľadanie optimálnej hypotézy h^* sťažilo. Iné identifikátory obsahovali indikáciu, že v blízkej budúcnosti prestanú byť používané, preto sme sa ich tiež rozhodli vyradiť. Pre regresné stromy a zhlukovanie sme, ak bola črta číselná, použili samotnú hodnotu črty. Črty s nečíselnou hodnotou sme premenili na kategorické premenné. V prípade ad-hoc metódy sme premenili všetky črty na kategorické premenné. Črta mala taký počet kategórií, koľko hodnôt má jej identifikátor. Následne poistná udalosť dostala skupinu pre daný identifikátor podľa toho, do ktorej hodnoty identifikátora spadala. Pri ad-hoc metóde pracujeme s 875 dátami, pri regresných stromoch s 890 dátami. Tento rozdiel je spôsobený problémom pri konverzii určitých čŕt 15 udalostí na kategorické premenné. V prípade zhlukovania pracujeme s oveľa väčším počtom dát, čo bližšie špecifikujeme v Kapitole 3.1.3.

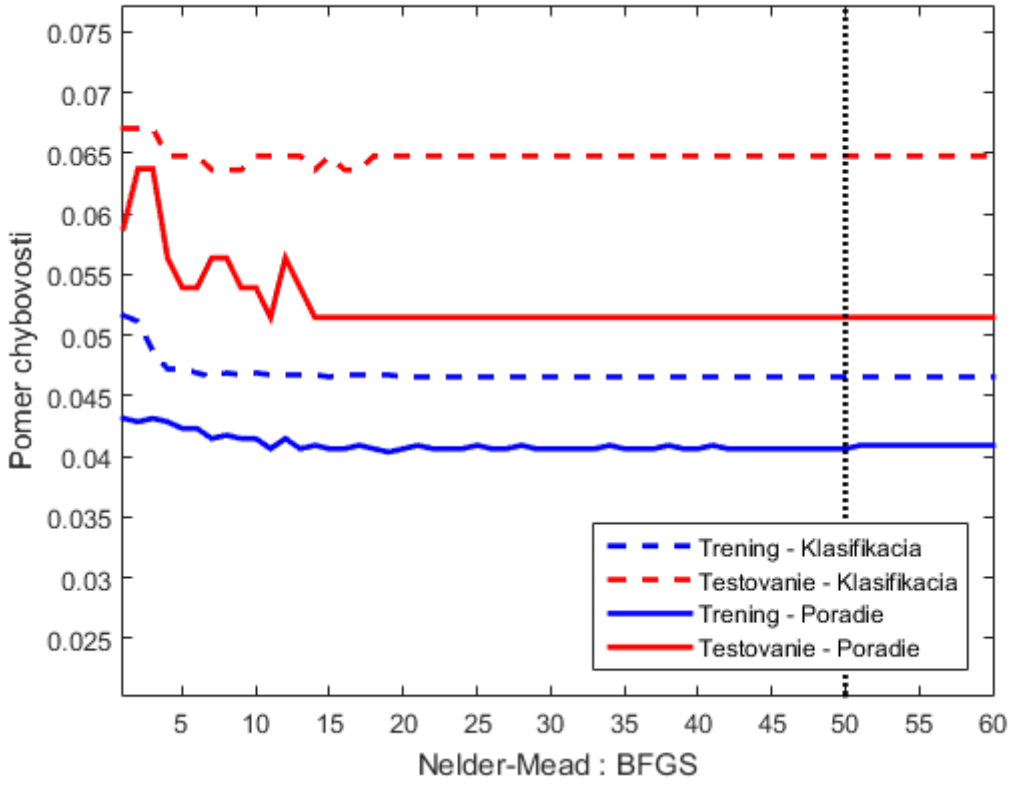
3.1 Výsledky

3.1.1 Ad-hoc metóda

Ako prvú predstavujeme implementáciu ad-hoc metódy špeciálne vytvorenej pre potreby poisťovne. Táto metóda je najjednoduchšou zmenou od pôvodného systému k systému automatickému, založenému sčasti na metódach strojového učenia. Ako spomíname v predošlej kapitole, ad-hoc metóda má rovnakú interpretovateľnosť parametrov ako pôvodný systém, teda je jej fungovanie najľahšie predstaviteľné aj pre expertov na od-

haľovanie podvodov bez znalostí fungovania strojového učenia. Optimalizačné metódy sme použili pomocou v MATLABE zabudovaných funkcií $fminsearch()$ a $fmincon()$. Pred trénovaním metódy volíme hodnoty niektorých parametrov. Pomer koeficientov strát $\frac{c_{21}}{c_{12}} = 2$, teda chyba druhého druhu má dvakrát vyššiu stratu ako chyba prvého druhu, čím sa snažíme dosiahnuť, aby metóda uprednostnila minimalizáciu chyby druhého druhu. Hlavným problémom pri trénovaní je správne nastavenie maximálneho počtu iterácii jednotlivých optimalizačných metód. Keďže Nelder-Mead používame ako hlavnú metódu na aproximáciu minima chybovej funkcie a kvázi-Newtonovskú metódu BFGS používame na doladenie výsledku, volíme maximálny počet iterácii pri Nelder-Mead výrazne vyšší. Počet iterácii najskôr nastavíme na vysokú hodnotu (30000 pre Nelder-Mead a 2000 pre BFGS) a následne sledujeme vývoj funkčnej hodnoty funkcie straty, trénovacích a testovacích chýb. Pre rozhodnutie o maximálnom počte iterácií potrebujeme preskúmať, či v modeli nedochádza k pretrénovaniu. K tomu nám pomáha pohľad na Obr. 4. Základným indikátorom pretrénovaného modelu je od určitej iterácie rastúca tendencia testovacej chyby napriek klesajúcej tendencii trénovacej chyby. Ani jedna optimalizačná metóda neskončila konvergenciou, ale dosiahnutím maximálneho počtu iterácií. Funkčná hodnota a chyby sú vyčíslené v prípade metódy Nelder-Mead v každej $\frac{1}{50}$ procesu trénovania, teda hodnoty $1, \ldots, 50$ na x-ovej osi až po zvislú čiernu čiaru pripadajú práve tejto metóde. Pre BFGS sú funkčná hodnota a chyby vyčíslené v každej $\frac{1}{10}$ trénovania, teda tejto metóde pripadajú hodnoty $51, \ldots, 60$ na x-ovej osi od zvislej čiernej čiary. Rovnaká logika zostrojenia x-ovej osi platí aj pre Obr. 5, Obr. 6 a Obr. 7. Obr. 4 síce nevykazuje rastúcosť testovacích chýb, no po približne 6000 iteráciách metódy Nelder-Mead, teda po čísle 10 na x-ovej osi, môžeme ďalší vývoj považovať za jemné dolaďovanie, keďže hodnoty všetkých chýb stagnujú.

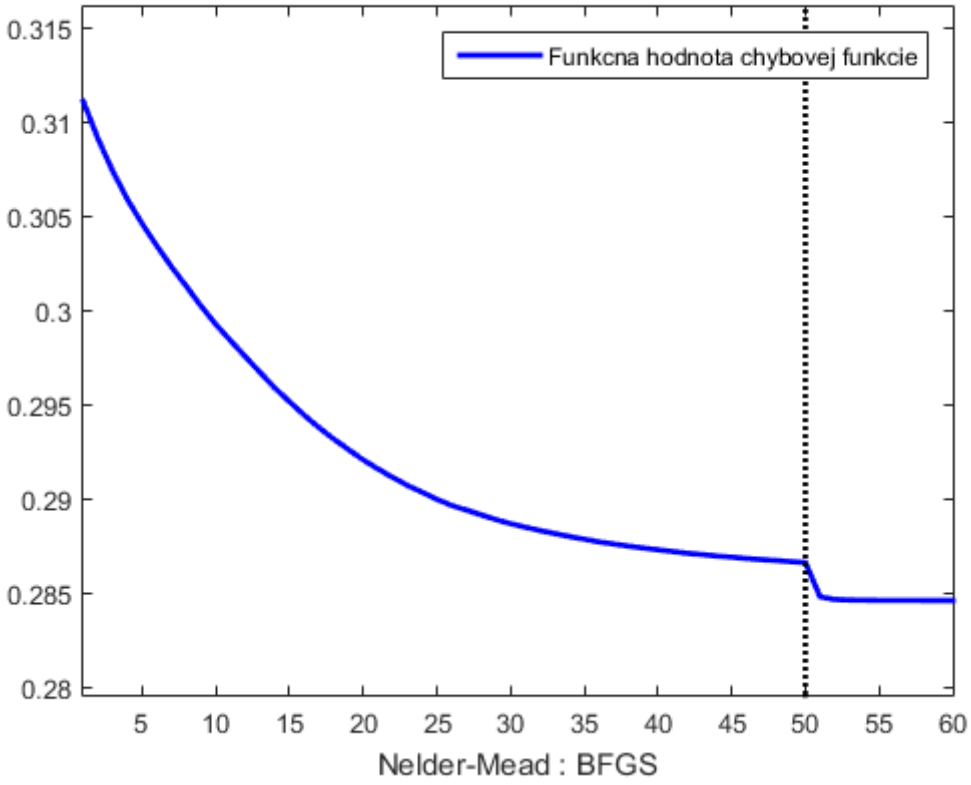
Dostávame sa teda ku maximálnemu počtu iterácií 6000 pre metódu Nelder-Mead a zachovaním pôvodného pomeru medzi týmito optimalizačnými metódami dostávame maximálny počet iterácií 400 pre BFGS. Obr. 5 popisuje vývoj funkčnej hodnoty funkcie straty v priebehu trénovania na trénovacej množine. Ako môžeme vidieť podrobnejšie na Obr. 5, procesom trénovania skutočne znižujeme funkčnú hodnotu nami definovanej funkcie straty. Krivka priebehu vývoja funkčnej hodnoty funkcie straty je pre zvyšujúci sa počet iterácií nestúpajúca, pretože metódy Nelder-Mead aj BFGS sú



Obr. 4: Vývoj trénovacích a testovacích chýb pri vysokom počte maximálnych iterácií optimalizačných metód

spádové. Použitie BFGS sa z pohľadu funkcie straty ukázalo ako veľmi prínosné, keď už v počiatočných iteráciách výrazne znížila jej funkčnú hodnotu.

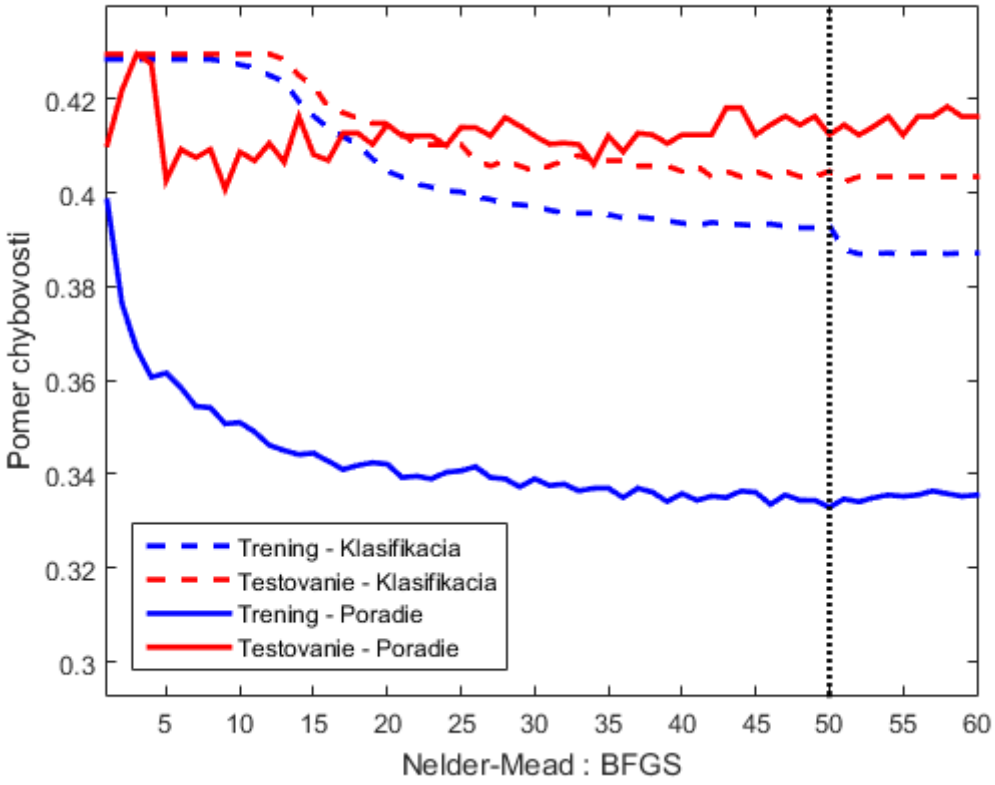
Na odhadnutie kvality natrénovaného modelu sa pozrieme na vývoj trénovacích a testovacích chýb. Pre čo najlepší odhad, vzhľadom na malý počet dát, použijeme 8-násobnú krížovú validáciu, teda na Obr. 6 a Obr. 7 už zobrazujeme priemer chýb z 8 rôznych inštancií modelu. Každá inštancia je natrénovaná na inej časti dát a otestovaná na zvyšku. Chyba klasifikácie, aj keď pre naše potreby menej podstatná, sa v priebehu trénovania, ako ukazuje Obr. 6, stabilne znižuje. Jediný problém nastáva pri testovacej chybe poradia, čo je zároveň pre nás najdôležitejšia metrika kvality modelu. Napriek klesajúcej funkčnej hodnote funkcie straty a klesajúcim klasifikačným chybám testovacia chyba poradia v priebehu trénovania rastie. Testovacia chyba poradia ale obsahuje najrizikovejších p udalostí, kde p je počet podvodov v dátovej vzorke. Z praktického hľadiska je p našej trénovacej vzorky stále priveľké na to, aby zamestnanci stihli všetkých p udalostí manuálne prezrieť. Obr. 7 sa bližšie pozerá na testovaciu



Obr. 5: Vývoj funkčnej hodnoty funkcie straty pri trénovaní

chybu poradia jej rozdelením na podskupiny podľa rizikovosti. Toto rozdelenie ukazuje, že prvých $p * 50\%$ udalostí predikovaných za najrizikovejšie v priebehu trénovania skutočne obsahuje viac a viac podvodov, teda trénovaním aj túto metriku zlepšujeme. Malou výnimkou je zhoršenie týchto chýb o približne 1% po aplikovaní BFGS. Trénovanie končí so všetkými chybami na úrovni 40%, čo pripisujeme obmedzenosti množiny hypotéz H, teda obmedzeniu samotnej ad-hoc metódy. Či sú chyby na takejto úrovni stále postačujúce pre použitie v praxi zistíme porovnaním ad-hoc metódy s pôvodným systémom.

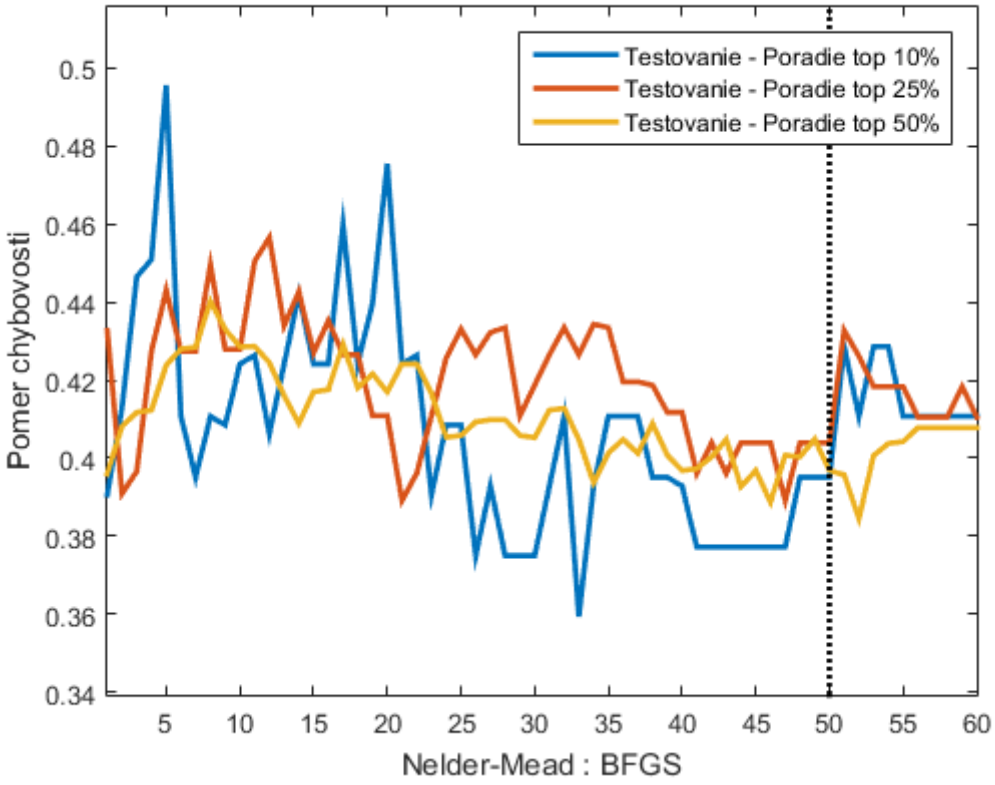
Odhad chyby poradia pomocou krížovej validácie je pre pôvodný systém uvedený v Tabuľke 4 a pre ad-hoc metódu v Tabuľke 5. Trénovaním a testovaním pri pôvodnom systéme myslíme aplikovanie pôvodných váh na predikciu rizikovosti pre trénovaciu a testovaciu vzorku. Pred začiatkom krížovej validácie sú záznamy pomiešané a rozdelené na trénovaciu a testovaciu vzorku. Obe metódy testujeme naraz pre tie isté záznamy. Analyzovaním tabuľky pre pôvodný systém zistíme, že úspešnosť zoradenia je na úrovni



Obr. 6: Vývoj trénovacích a testovacích chýb pri trénovaní

polovice, čo je zároveň aj pomer podvodov v dátach. Úspešnosť pôvodného systému pri zoraďovaní je teda na úrovni náhodného zoradenia. Veľmi optimistický je výsledok trénovania ad-hoc metódy, kde úspešnosť zoradenia medzi najrizikovejšími $p*50\%$ udalosťami je na úrovni takmer až 70% s ešte lepším výsledkom v prvých $p*10\%$ a $p*25\%$ udalostiach. Tento výsledok sa žiaľ nepotvrdil pri testovaní, no aj tak dosahuje ad-hoc metóda v porovnaní s pôvodným systémom zlepšenie na úrovni $1-5\%$. Najvýraznejšie zlepšenie 5.40% je v prvých $p*25\%$ poistných udalostiach s najvyššou predikovanou rizikovosťou, čo je v praxi dosiahnuteľná hodnota počtu manuálne skontrolovaných udalostí, preto sme s výsledkom relatívne spokojní. V prostredí, kde jeden odhalený poistný podvod ušetrí poisťovni aj tisícky eur, má aj takéto zlepšenie význam, no priestor na zlepšenie je veľký.

Klasifikačná chyba nie je pre naše účely podstatná. Keďže pôvodný systém na rozdiel od ad-hoc metódy nemá jasne určenú hranicu, od ktorej považuje poistnú udalosť za podvod, uvádzame len jednu kontingenčnú tabuľku klasifikácie v Tabuľke 6. Na



Obr. 7: Rozdelenie chyby poradia na podskupiny

Pôvodný systém	Trénovanie			Testovanie		
Top $p * 10\%$	24.5	44.63	54.90%	4.00	7.38	54.24%
Top $p * 25\%$	59.25	110.50	53.61%	9.13	16.88	53.55%
Top $p * 50\%$	124.13	220.13	56.38%	18.18	32.63	57.62%
Top p	243.86	437.75	55.70%	36.00	62.75	57.01%

Tabuľka 4: Odhad úspešnosti poradia pomocou krížovej validácie pre pôvodný systém

Ad-hoc metóda	Trénovanie			Testovanie		
Top $p * 10\%$	32.00	44.63	71.69%	4.38	7.38	58.93%
Top $p * 25\%$	79.13	110.50	71.60%	10.00	16.88	58.95%
Top $p * 50\%$	153.25	220.13	69.61%	19.38	32.63	59.22%
Top p	290.86	437.75	66.44%	36.75	62.75	58.38%

Tabuľka 5: Odhad úspešnosti poradia pomocou krížovej validácie pre ad-hoc metódu

základe tejto tabuľky vidíme, že ad-hoc metódu skutočne označuje udalosti za nerizikové len veľmi opatrne. Zo 766 poistných udalostí v trénovacej množine označila za nerizikové (0) len 42 z nich, z toho v priemere takmer 37 správne. Podobná situácia nastáva pri testovacej množine, teda účel minimalizácie chyby druhého druhu sa podarilo dosiahnuť. Ak metóda označí udalosť za bezrizikovú, vo veľkom percente prípadov sa nemýli.

Ad-hoc	Trénovanie		Testovanie	
metóda	0	1	0	1
0	36.75	291.38	4.5	42.75
1	5.13	432.63	1.63	61.13

Tabuľka 6: Odhad klasifikačnej chyby pomocou krížovej validácie pre ad-hoc metódu

Po skončení trénovania prichádza na rad dôležitá časť celého nového systému - revízia expertmi na odhaľovanie podvodov. Interpretovateľnosť natrénovaných váh považujeme za veľkú výhodu ad-hoc metódy. Rigorózne extrahovanie informácií z dát pomocou strojového učenia a bohaté skúsenosti expertov môžu spoločne dosahovať lepšie výsledky, ako jednotlivé prístupy osobitne. V pôvodnom systéme experti navrhovali váhy na základe svojich skúseností, no v novom systéme môžu skúmaním novo natrénovaných váh prísť na skutočnosti, ktoré predtým prehliadli. Tabuľka 7 ilustruje, ako môžu natrénované váhy vyzerať. Presné hodnoty neuvádzame z bezpečnostných dôvodov. Zamýšľanie sa nad hodnotami váh môže viesť napríklad týmito smermi:

- váha s najväčšou hodnotou spomedzi všetkých váh najviac prispieva k rizikovosti poistnej udalosti, teda hodnota identifikátora, ku ktorému táto váha prislúcha, môže byť medzi podvodmi veľmi častá,
- ak spadá poistná udalosť do hodnoty identifikátora so zápornou hodnotou, rizikovosť poistnej udalosti táto skutočnosť zníži,
- ak majú váhy pre jeden identifikátor veľmi podobnú hodnotu, stojí za zváženie vyradenie identifikátora z vektora čŕt,
- veľké zmeny v hodnotách určitých váh po opätovnom natrénovaní modelu s prí-

Identifikátor	Hodnoty	Váha
Počet udalostí vodiča	hodnota 1	1.45
	hodnota 2	1.87
	hodnota 3	0.98
	hodnota 4	1.54
	hodnota 5	1.75
Záujmové VIN	hodnota 1	−1.59
	hodnota 2	0.76
	hodnota 3	0.19
	hodnota 4	0.01
	hodnota 5	2.54
	hodnota 6	1.28
Priemerná škoda posledných 2 poistných udalostí	hodnota 1	0.75
	hodnota 2	0.78
Dlh na poistnom	hodnota 1	4.84
	hodnota 2	4.84
⋮	⋮	⋮

Tabuľka 7: Ilustračný príklad natrénovaných váh ad-hoc metódou

chodom nových záznamov môžu znamenať zmenu taktiky podvodníkov, respektíve zmeny v prostredí poistných podvodov.

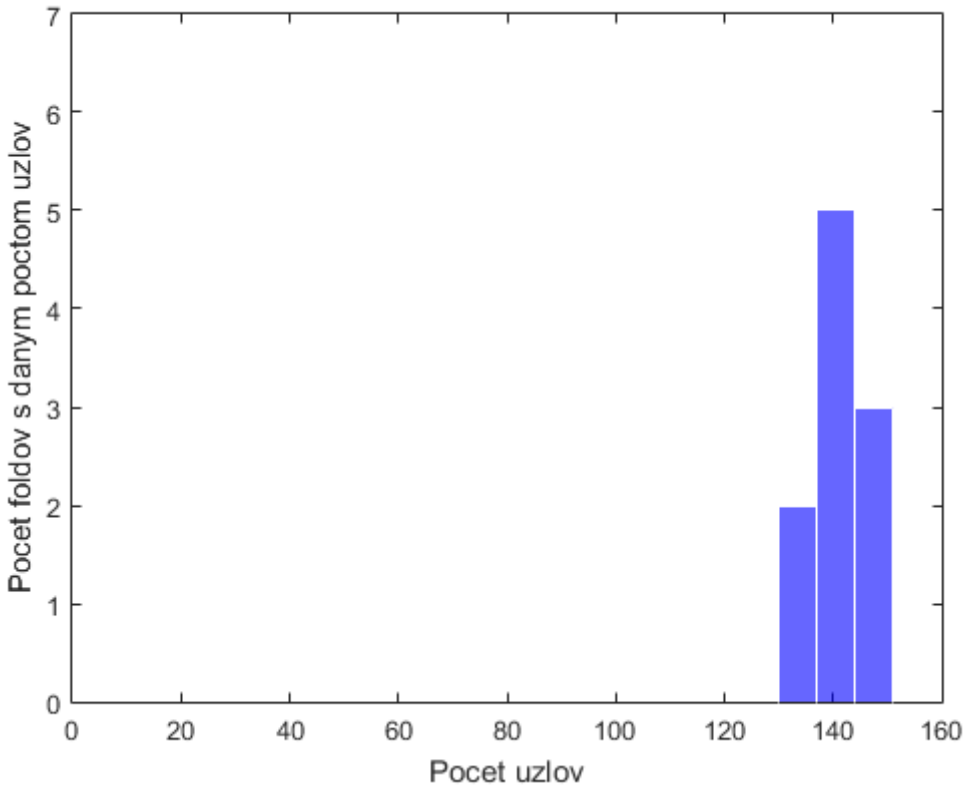
Ad-hoc metódu v konečnom dôsledku považujeme na základe nášho testovania za zlepšenie pôvodného systému, no či to tak naozaj je, ukáže až jej použitie v praxi. Napriek malému počtu dát vykazuje z testovania pomocou krížovej validácie menšiu chybu poradia, teda pracovníci môžu pri kontrole naraziť na o niečo vyšší počet podvodov. Tento výsledok sa môže ďalej zlepšovať ladením parametrov na základe výsledkov z praxe. Potenciál metódy vidíme aj pri použití spoločne s expertnými skúsenosťami pracovníkov poisťovne.

3.1.2 Klasifikačné stromy

Implementácia klasifikačných, teda konkrétne regresných stromov, je z programátorského hľadiska jednoduchšia ako implementácia ad-hoc metódy vďaka funkciám v MATLAB-e určeným na tieto účely. Pre regresné stromy sme použili funkciu $fitrtree()$. Pre uprednostnenie minimalizácie chyby druhého druhu existujú dva prístupy. Prvým, viac inžinierskym, je v prípade celočíselnosti $\frac{c_{21}}{c_{12}}$ zduplikovať záznamy v dátach, ktoré sú podvodmi $\frac{c_{21}}{c_{12}}$-krát. Ak nastavíme pomer $\frac{c_{21}}{c_{12}} = 2$, potom napríklad pre podvod x_i vytvoríme v dátach dvojníka x_j a následne akákoľvek chyba klasifikácie alebo nepresnosti v predikcii rizikovosti x_i, bude rovnaká aj pre x_j a teda započítaná 2-krát. Jednoduchšou metódou, fungujúcou aj pre neceločíselné $\frac{c_{21}}{c_{12}}$, je použitie váženej funkcie nečistoty spomenutej v Kapitole 2.2.4, ktorá je priamo zabudovaná do funkcie $fitrtree()$. Pomocou vstupného parametra „*weights*" vieme každému dátovemu záznamu x^p v trénovacej množine nastaviť váhu w^p, teda pre naše potreby stačí, ak všetkým skutočným podvodom nastavíme $w^i = \frac{w_1}{w_2} = \frac{c_{21}}{c_{12}} > 1$ a všetkým ostatným udalostiam $w^j = 1$.

Ako spomíname v Kapitole 2.2.4, interpretovateľnosť a rovnako aj použiteľnosť regresných stromov sa pri veľkom počte uzlov zmenšuje. Funkcia $fitrtree()$ tento problém v prípade potreby vie vyriešiť nastavením vstupného parametra „MaxNumSplits", čím limitujeme maximálny počet uzlov v natrénovanom strome. V prvom rade sa pozrieme, koľko uzlov je optimálnych bez ohraničenia jeho počtu. Toto zistenie prezentuje Obr. 8 pomocou histogramu počtu optimálnych uzlov pre jednotlivé inštancie v 10-násobnej krížovej validácii. Z histogramu vidíme, že optimálny počet uzlov sa stabilne pohybuje okolo 140, čo ďaleko presahuje praktické počty. Príkladom nevýhod takéhoto vysokého počtu uzlov sú debaty medzi expertmi, keď uzly pre nich nemajú takmer žiadnu výpovednú hodnotu už po pár vrstvách, prípadne znemožnenie predikcie z takéhoto stromu iným spôsobom, ako pomocou MATLAB-u. Pracovníci by si teda strom nemohli vytlačiť, prípadne ho naprogramovať do bežne používaných programov bez veľkej straty času.

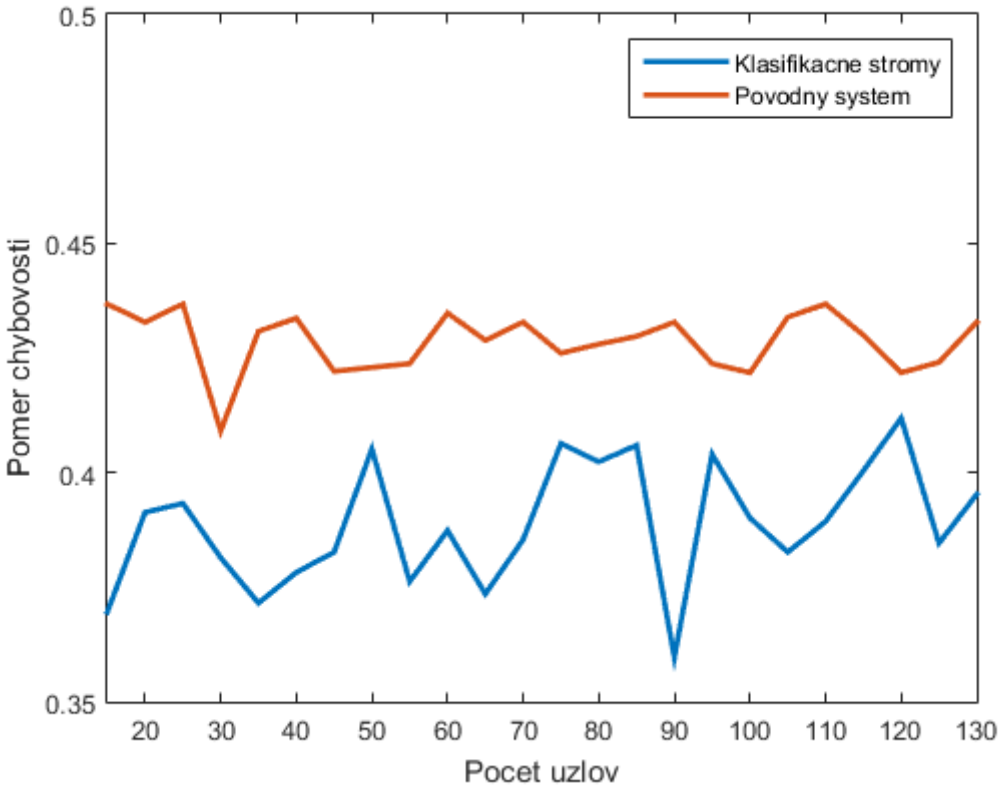
Naším úmyslom je teda rozumne maximálny počet uzlov obmedziť. Otázkou je, v akej veľkej miere môžeme takéto obmedzenie uskutočniť bez výraznej straty na kvalite modelu. Za kvalitu modelu opäť považujeme chybu poradia, nakoľko tá najlepšie odhaduje použiteľnosť modelu v prostredí spoločnosti Generali Poisťovňa, a. s.. Na



Obr. 8: Optimálny počet uzlov pre jednotlivé inštancie v krížovej validácii

Obr. 9 ukazujeme odhad chyby poradie pre rôzny počet uzlov, teda pre rôzne nastavenia vstupného parametra „MaxNumSplits". Chybu poradia pre každý počet uzlov odhadujeme pomocou 8-násobnej krížovej validácie. Spoločne s týmto odhadom chyby zobrazuje obrázok aj priemernú chybu poradia pôvodnej metódy. Ako môžeme vidieť, regresné stromy majú chybovosť nižšiu ako pôvodný systém pre všetky počty uzlov. Zároveň vidíme, že chyba poradia je zhruba konštantná pre rôzne počty uzlov, teda obmedzenie tohto počtu by pre naše potreby nemalo kvalitu modelu znížiť. Z praktického hľadiska sme sa teda rozhodli vybrať maximálny počet uzlov 15.

Tabuľky 8 a 9, rovnako ako pri ad-hoc metóde, predstavujú odhad chyby poradia pomocou krížovej validácie. Chybu poradia rozdeľujú na menšie skupiny a zoraďujú ich od tej pre nás najdôležitejšej v prvom riadku, až po najmenej dôležitú v riadku sumárnom, nakoľko v takom poradí sa s novými poistnými udalosťami stretávajú zamestnanci poisťovne. Úspešnosť správneho zoradenia udalostí podľa rizika je pochopiteľne pre regresné stromy väčšia v prípade trénovacej množiny, no rozdiel v úspešnostiach medzi



Obr. 9: Porovnanie chyby poradia krížovej validácie pre regresný strom s rôznym počtom uzlov a pôvodný systém

trénovacou a testovacou množinou nie je taký veľký ako v prípade ad-hoc metódy. Veľký rozdiel až 16.76% je len v prípade úplne najrizikovejších udalostí, teda pre skupinu $p * 10\%$. Táto skutočnosť ukazuje na lepšiu robustnosť regresných stromov oproti ad-hoc metóde, teda v ich prípade nedošlo na základe testovania k výraznejšej forme pretrénovania. Zatiaľ čo zlepšenie úspešnosti oproti pôvodnému systému je v prípade ad-hoc metódy najviac 5.40%, pri regresných stromoch je to 8.88% v skupine $p * 25\%$. Takéto zlepšenie už môže byť postrehnuteľné aj pre bežného zamestnanca pri manuálnej kontrole udalostí. Krížová validácia nám indikuje, že ak preveríme 17 udalostí s najvyššou predikciou rizikovosti, v priemere by sme mali naraziť na namiesto pôvodných 9.13 podvodov až na 10.75. Zlepšenia úspešnosti zoradenia sú celkovo pre regresné stromy v intervale $2 - 9\%$.

Klasifikačnú chybu vo forme priemernej kontingenčnej tabuľky klasifikácie z krížovej validácie uvádzame v Tabuľke 10. Tá nám ukazuje, rovnako ako pri ad-hoc metóde, že regresné stromy klasifikujú veľmi opatrne, aby minimalizovali chybu druhého druhu. Pri

Pôvodný systém	Trénovanie			Testovanie		
Top $p * 10\%$	26.38	45.50	57.94%	4.00	7.25	55.43%
Top $p * 25\%$	61.25	112.00	54.67%	9.13	17.13	52.64%
Top $p * 50\%$	128.38	223.13	57.52%	19.00	33.00	56.95%
Top p	249.13	443.63	56.15%	36.50	63.75	56.65%

Tabuľka 8: Odhad úspešnosti poradia pomocou krížovej validácie pre pôvodný systém

Regresné stromy	Trénovanie			Testovanie		
Top $p * 10\%$	35.63	45.50	78.22%	4.50	7.25	61.46%
Top $p * 25\%$	78.00	112.00	69.59%	10.75	17.13	61.52%
Top $p * 50\%$	153.75	223.13	68.89%	20.75	33.00	62.34%
Top p	299.63	443.63	67.52%	38.13	63.75	59.16%

Tabuľka 9: Odhad úspešnosti poradia pomocou krížovej validácie pre regresné stromy

regresných stromoch sme pomer $\frac{c_{21}}{c_{12}}$ pomocou vstupného vektora „*weights*" nastavili na hodnotu 3. Testovanie ukazuje, že pri tých udalostiach, ktoré regresné stromy označili ako bezrizikové, je táto klasifikácia správna v nižšom pomere prípadov ako pri ad-hoc metóde. Zatiaľ čo ad-hoc metóda sa mýlila len zhruba v $\frac{1}{3}$ prípadov, pri regresných stromoch je to viac než $\frac{1}{2}$. Táto skutočnosť ale pre naše potreby nie je dôležitá, ohľad berieme len na optimalizáciu chyby poradia.

Regresné	Trénovanie		Testovanie	
stromy	0	1	0	1
0	67.88	267.38	7.00	41.25
1	4.50	439.13	3.88	59.88

Tabuľka 10: Odhad klasifikačnej chyby pomocou krížovej validácie pre regresné stromy

Interpretovateľnosť regresných stromov nie je v súlade s interpretovateľnosťou pôvodného systému, ako to platí pri ad-hoc metóde. Nič to ale nemení na fakte, že je interpretovateľnosť stále vysoká a intuitívna. Pomocou natrénovaného stromu môžu experti vďaka svojim skúsenostiam stále prísť na niečo, čo by inak mohlo zostať nepovšimnuté. Použitie identifikátora v uzloch stromu ukazuje na jeho signifikantnosť.

Čím je otázka na daný identifikátor vyššie v strome a čím je identifikátor použitý vo väčšom počte otázok, tým je signifikantnejší. Rovnako môže byť veľmi výpovedný aj parameter θ_i, s ktorým sa črta poistnej udalosti v i-tom uzle porovnáva. V prípade kategorickej premennej sa pýtame, či hodnota črty poistnej udalosti pochádza z určitej množiny. Každá hodnota identifikátora v danej množine môže mať svoj interpretovateľný význam.

Regresné stromy teda na základe testovania chyby poradia pomocou krížovej validácie preukazujú výraznejšie zlepšenie pôvodného systému ako ad-hoc metóda. Ad-hoc metóda si drží miernu výhodu v interpretovateľnosti a v implementácii do pôvodného systému poisťovne. Aby sme jednoduchosť implementácie regresných stromov zlepšili, vytvorili sme a dodali poisťovni Generali Poisťovňa, a. s. súbor programu Microsoft Excel na predikciu z natrénovaného regresného stromu. Súbor umožňuje po zadaní parametrov stromu predikciu zo stromu bez použitia iných súborov alebo programov.

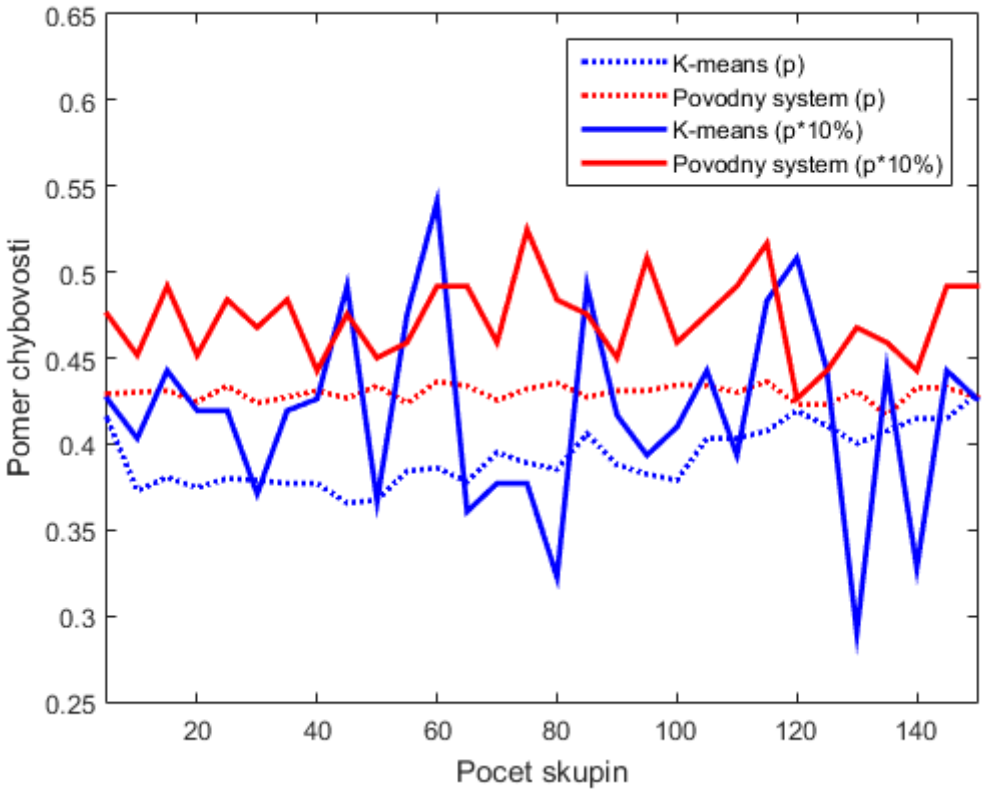
3.1.3 Zhlukovanie

Ako poslednú použitú metódu testujeme zhlukovanie, teda konkrétne metódu K-means. Použitie K-means, ako spomíname v Kapitole 1.2.3, je iné ako tomu bolo pri ad-hoc metóde alebo regresných stromoch. Zhlukovanie je doplnkovou metódou na nájdenie opakujúcich sa alebo veľmi podobných podvodov. Cieľom zhlukovania je dodatočne zlepšiť počet nájdených podvodov celého nového systému, teda zlepšiť úspešnosť ich nájdenia v určitom počte nových záznamov poistných udalostí s minimálnym pridaným úsilím pre pracovníka vykonávajúceho manuálnu kontrolu. Tento cieľ naplníme, ak K-means dosiahne vyššiu odhadovanú úspešnosť nájdenia podvodov medzi malým počtom $p * 10\%$ udalostí, označených podľa postupu uvedenom v 1.2.3 za najrizikovejšie. Vďaka tomu pracovník po preverení väčšieho množstva poistných udalostí na základe klasifikačných metód doplnkovo preverí pár zhlukovaním označených najrizikovejších udalostí. Veľkým problémom pri predošlých metódach bol nedostatok dát, no tento problém pri K-means máme len čiastočne. Keďže K-means používa učenie bez učiteľa, nepotrebujeme mať v trénovacej množine, okrem príkladov vektora čŕt x, nič iné. Tým sa nám trénovacia množina určená na vytvorenie skupín udalostí zväčší z pôvodných menej ako 900 na 62502. Označenie niektorých udalostí za podvod alebo bezrizikovú

udalosť potrebujeme ale pri postupe na predikciu rizikovosti, teda na pridelenie rizikovosti jednotlivým skupinám. Čím viac manuálne preverených udalostí máme, tým je označenie rizikovosti skupín presnejšie. V prípade K-means používame len spojité črty. Použitie kategorických čŕt by mohlo mať za následok to, že aj dve udalosti zhodné vo všetkých črtách okrem jednej kategorickej by s veľkou pravdepodobnosťou priradilo K-means do rôznych skupín. Uprednostnenie minimalizácie chyby druhého druhu v tomto doplnkovom postupe nezohľadňujeme.

Metóda K-means je rovnako ako regresné stromy všeobecne používaná, preto MATLAB obsahuje funkciu $kmeans()$ na jej použitie. Kľúčovým a zároveň jediným parametrom potrebným na odštartovanie trénovania na trénovacej množine je počet skupín K. Na základe nami navrhnutého postupu na predikovanie rizikovosti nechceme mať logicky počet skupín nízky. Maximálny počet hodnôt odhadu rizikovosti pre poistnú udalosť je v prípade zhlukovania rovný počtu skupín. Ak zvolíme počet skupín príliš malý, nepodarí sa nám dostatočne rozdeliť poistné udalosti tak, aby sme v konečnom dôsledku dostali malú skupinu najrizikovejších. Naopak, ak zvolíme počet skupín príliš veľký, bude kvalita rozdelenia pravdepodobne nízka, teda v konečnom dôsledku aj prakticky veľmi podobné udalosti by sme mohli považovať podľa K-means za rozdielne. Trénovanie K-means opakujeme pre každú inštanciu a pre každý počet skupín 5-krát pomocou „Replicates" vo funkcii $kmeans()$. Vďaka tomu znížime šancu, že trénovanie pre danú inštanciu a počet skupín skončí v lokálnom minime.

Nájsť vhodnú hodnotu K nám pomáha Obr. 10 zobrazujúci odhad chyby poradia pomocou 5-násobnej krížovej validácie pre rôzne počty skupín. Obrázok obsahuje chyby poradia pre najrizikovejších $p * 10\%$ udalostí pre pôvodný systém a zhlukovanie ako kľúčovú metriku výberu K. Tá je v 24 z 30 prípadov pri zhlukovaní lepšia ako v pôvodnom systéme. Najlepší výsledok vykazuje $K = 130$, na druhom mieste je $K = 80$, na treťom $K = 50$ a $K = 65$. Keďže chyba poradia $p * 10\%$ má pre zhlukovanie vzhľadom na počty skupín relatívne veľký rozptyl, rozhodneme medzi kandidátmi na najlepší počet skupín pomocou inej metriky. Tou je chyba poradia pre p najrizikovejších udalostí, ktorá lepšie popisuje celkovú kvalitu modelu. Na základe Obr. 10 hodnota chyby poradia p pre zhlukovanie s rastúcim počtom skupín takisto rastie. Rozdiel medzi chybou poradia p pre $K = 50$ a $K = \{65, 80\}$ je okolo 2%, pri $K = 130$ je tento rozdiel

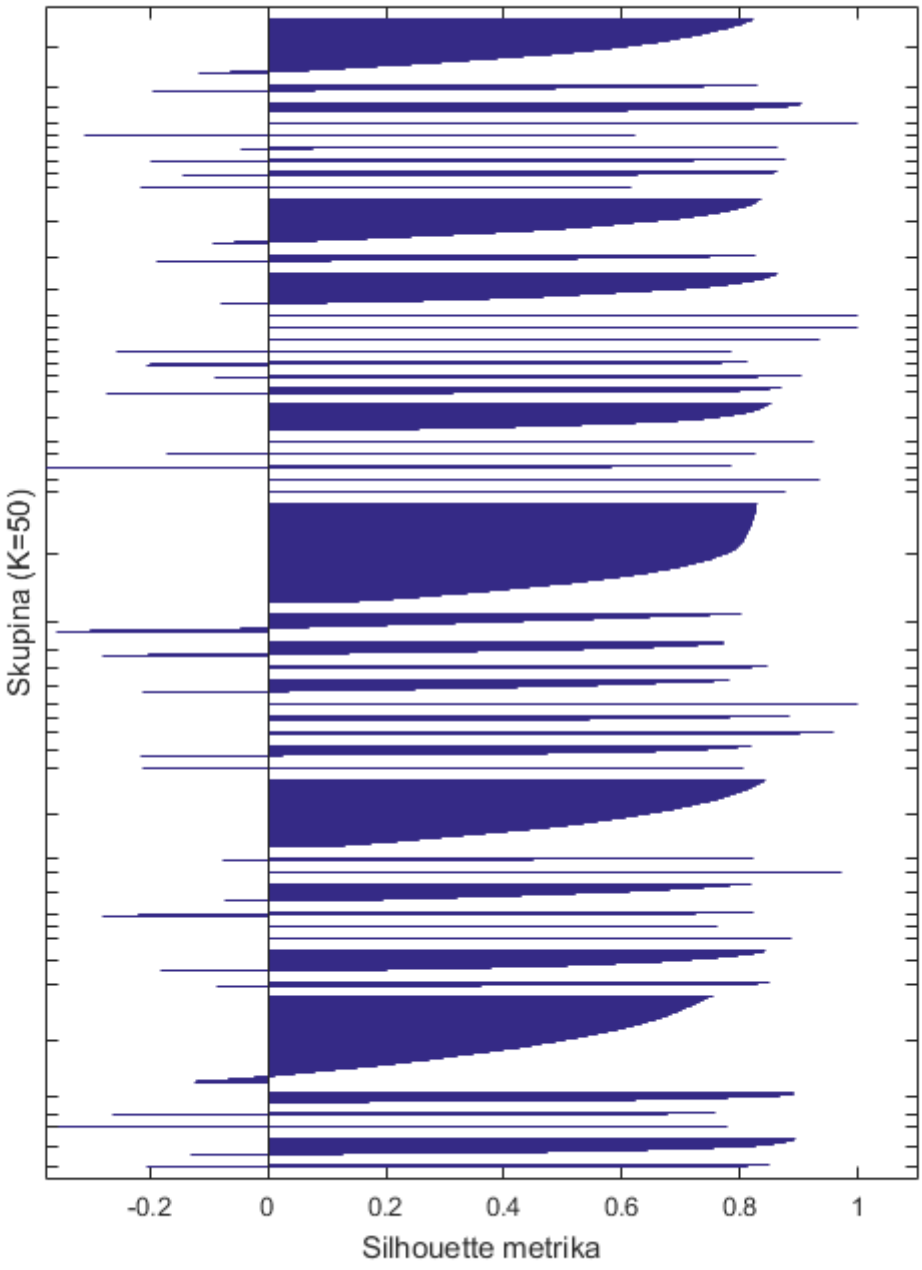


Obr. 10: Porovnanie chyby poradia pomocou krížovej validácie pre K-means s rôznym počtom skupín

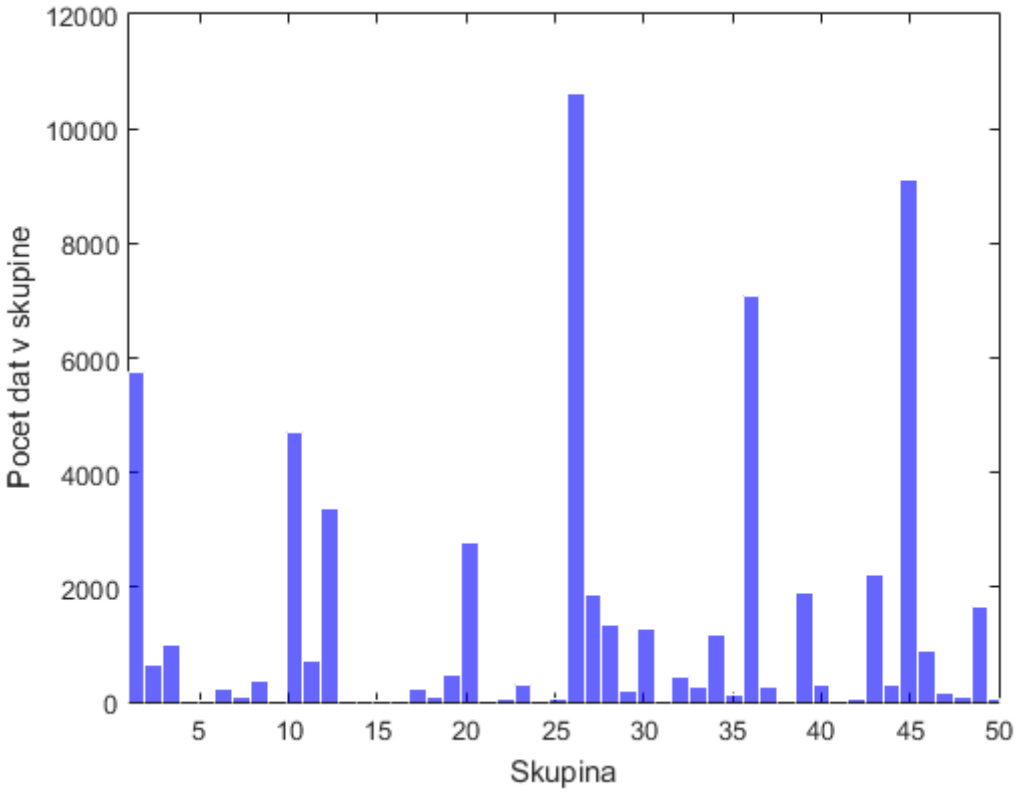
ešte oveľa výraznejší. Vzhľadom na malý rozptyl hodnôt tejto chyby považujeme tieto rozdiely za výrazné. Po týchto zisteniach vyberáme za optimálny počet $K = 50$ pre danú trénovaciu množinu.

Kvalitu rozdelenia dát do skupín pre zvolený počet $K = 50$ zanalyzujeme pomocou Obr. 11 a Obr. 12. Na Obr. 11 je takzvaný „Silhouette" graf, ktorý zobrazuje vzdialenosť každého bodu určitej skupiny od bodov susedných skupín. Táto vzdialenosť je zobrazená v metrike s hodnotami v intervale $[-1, 1]$. Hodnota metriky 0 predstavuje pozíciu na rozhodovacej hranici, hodnota 1 znamená, že bod je ďaleko od bodov susedných skupín. Záporné hodnoty ukazujú body s pravdepodobne zlým zaradením do skupiny. Metrika je zobrazená na x-ovej osi, y-ová os ukazuje rozdelenie bodov do skupín. Čím skupina zaberá väčší interval na y-ovej osi, tým obsahuje viac záznamov. V našom prípade máme 7 skupín s väčším počtom dát, ostatné sú menšie. Záznamov so zápornou metrikou alebo metrikou blízkou 0 je veľmi málo. S rozdelením sme teda spokojní, nakoľko ak menšie skupiny obsahujú skutočný podvod, je šanca, že sa nejedná o

náhodu. „Silhouette" graf dopĺňa histogram s presnejším zobrazením počtu dát v jednotlivých skupinách. Najväčšou je skupina 26 s takmer 11000 záznamami. Z najväčších 7 skupín obsahuje každá viac ako 3000 záznamov. Skupín s menej ako 1000 záznamami je 36.



Obr. 11: „Silhouette" graf pre K=12



Obr. 12: Histogram počtu dát v jednotlivých skupinách

Celkovú úspešnosť zhlukovania s 50 skupinami porovnáva s úspešnosťou pôvodného systému pomocou 5-násobného krížového validačného odhadu Tabuľka 11. Náš zámer je úspešný, lebo na odhad úspešnosti poradia pre najrizikovejších $p * 10\%$ udalostí je na úrovni až 64.06%, teda je skutočne vyšší ako pri pôvodnej metóde až o 11%. Aby použitie zhlukovania malo zmysel, je potrebné, aby bolo v tejto metrike lepšie ako ad-hoc metóda alebo regresné stromy. Táto skutočnosť sa potvrdila, keď dosiahnutých 64.06% je o 2.6% vyššie ako pri regresných stromoch a o 5.13% vyššie ako pri ad-hoc metóde. Prekvapujúcim zistením je, že zhlukovanie je najúspešnejšie aj v skupinách $p * 25\%$, $p * 50\%$ a p, keď ako jediné dosahuje hodnoty vo všetkých prípadoch nad 60%.

Celkovo sme úspešnosťou použitia zhlukovania a naším experimentálnym postupom na určenie rizikovosti pre jednotlivé skupiny veľmi spokojní. Zhlukovanie nielen plní náš prvotný účel, no je celkovo na základe testovania najúspešnejšou metódou. Použitie zhlukovania v praxi napriek týmto skutočnostiam nemeníme a ostáva len doplnkovou metódou k regresným stromom alebo ad-hoc metóde. Dôvodom je malá interpretova-

	Zhlukovanie			**Pôvodný systém**		
Top $p * 10\%$	5.13	8.00	64.06%	4.25	8.00	53.13%
Top $p * 25\%$	11.50	18.50	62.16%	10.13	18.50	54.73%
Top $p * 50\%$	22.63	36.00	62.85%	21.00	36.00	58.33%
Top p	44.00	69.38	63.42%	40.38	69.38	58.20%

Tabuľka 11: Odhad úspešnosti poradia pomocou krížovej validácie na testovacej množine pre zhlukovanie a pôvodný systém

teľnosť zhlukovania a veľmi experimentálna povaha metódy založená na predpoklade podobnosti podvodov alebo ich opakovaniu.

3.1.4 Simulácia použitia systému v praxi

Po aplikovaní a otestovaní jednotlivých metód pristúpime na záver k porovnaniu celých systémov. V takomto porovnaní nemôžeme predpovedať zistenia expertov z revízie natrénovaných modelov, preto nemôžeme zohľadniť vysokú podobnosť interpretovateľnosti ad-hoc metódy s pôvodným systémom. Za nový systém považujeme teda v tejto kapitole ten s najväčším potenciálom na základe testovania poradia, teda kombináciu regresných stromov a zhlukovania. Simulujeme trénovanie na záznamoch dostupných v určitom časovom bode, ostatné záznamy pridelíme do testovacej vzorky. Následne predpokladáme, že m z n poistných udalostí testovacej vzorky stihne prejsť na manuálnu kontrolu. V praxi by mali fungovať indikátory, či daná poistná udalosť už bola manuálne skontrolovaná, preto pri zhlukovaní vyberieme len udalosti neskontrolované po indikácii regresným stromom. Úlohou pôvodného a nového systému je, aby medzi prvými m poistnými udalosťami, ktorým daný systém predikuje najvyššiu rizikovosť, bolo čo najviac skutočných podvodov. Aby oba systémy skontrolovali rovnaký počet udalostí, ak pôvodný systém posunie na manuálnu kontrolu m udalostí, regresné stromy posunú $m * 80\%$ a zhlukovanie $m * 20\%$ udalostí.

Celý zoznam poistných udalostí, ktorých črty máme k dispozícii, zoradíme podľa dátumu. Vyberieme dátum D tak, aby rozdelil zoradené dáta na 7/8 do trénovacej vzorky a 1/8 do testovacej. Následne natrénujeme regresný strom na trénovacej vzorke s rovnakými parametrami ako v Kapitole 3.1.2. Zoberieme všetky dáta do dátumu

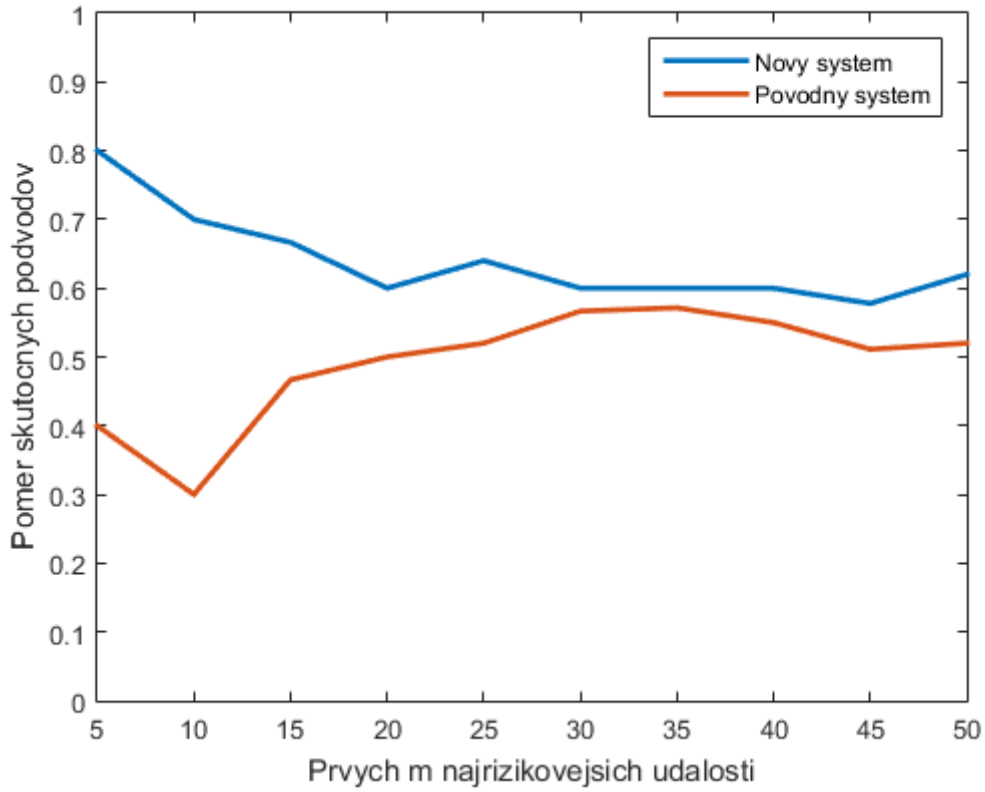
D o poistných udalostiach, ktoré neprešli na manuálnu kontrolu a teda nie je jasné, či sú podvodom a pridáme ich pre účely zhlukovania do trénovacej vzorky. Na tejto zväčšenej trénovacej vzorke natrénujeme metódu K-means s rovnakými parametrami ako v Kapitole 3.1.3 a následne pridelíme každej skupine rizikovosť pomocou postupu uvedeného v Kapitole 1.2.3. Rizikovosť udalostí v testovacej vzorke predikujeme osobitne pomocou regresných stromov, zhlukovania a pôvodnej metódy. Následne prvých m v predikcii najrizikovejších udalostí z testovacej vzorky imaginárne pošleme na manuálnu kontrolu. Rovnako pošleme na manuálnu kontrolu prvých m najrizikovejších udalostí podľa pôvodného systému. V ideálnom prípade všetky udalosti, ktoré sme na manuálnu kontrolu sa ukážu byť skutočnými podvodmi. Výsledky porovnania sú podrobnejšie uvedené v Tabuľke 12. Vizuálne porovnanie úspšností systémov zobrazuje Obr. 13.

	Nový systém			**Pôvodný systém**	
m	Reg. stromy	K-means	Úspešnosť	Počet	Úspešnosť
5	3	1	80.00%	2	40.00%
10	6	1	70.00%	3	30.00%
15	9	1	66.67%	7	46.67%
20	11	1	60.00%	10	50.00%
25	15	1	64.00%	13	52.00%
30	17	1	60.00%	17	56.67%
35	19	2	60.00%	20	57.14%
40	21	3	60.00%	22	55.00%
45	23	3	57.78%	23	51.11%
50	27	4	62.00%	26	52.00%

Tabuľka 12: Odhad úspešnosti poradia pomocou krížovej validácie na testovacej množine pre nový a pôvodný systém

Na grafe v Obr. 13 môžeme vidieť, že pri menej ako 15-tich odoslaných v predikcii najrizikovejších udalostiach má nový systém ďaleko väčšiu úspešnosť, ako pôvodný. Pre $m = 10$ je úspešnosť nového systému o až 40% lepšia. Od $m = 30$ je úspešnosť metód na podobnej úrovni, no v minime pre $m = 35$ je ich rozdiel stále 2.86%. Tabuľka 12

ukazuje, že regresné stromy odoslali pre každé m okrem $m = \{35, 40\}$ aspoň tak veľa skutočných podvodov ako pôvodná metóda napriek tomu, že odosielali $m * \frac{4}{5}$ udalostí. Metóda K-means odosielala vždy len $m * \frac{1}{5}$, no jej úspešnosť bola nižšia hlavne kvôli tomu, že mohla odoslať len udalosti nevybraté regresnými stromami. Spoločne tieto dve metódy nového systému aj v tomto teste predčili pôvodný systém.



Obr. 13: Simulácia odoslania m v predikcii najrizikovejších udalostí na manuálne preverenie

Záver

V práci sme uviedli čitateľa do problematiky podvodov vo svete neživotného poistenia. Zhrnuli sme štandardné techniky identifikácie podozrivých poistných udalostí a podrobne sme zanalyzovali tento systém v poisťovni Generali Poisťovňa, a. s.. Identifikovali sme požiadavky na potencionálny automatizovaný systém, zanalyzovali dostupné dáta a na základe nich sme našli priestor na použitie metód strojového učenia. Zadefinovali sme metodiku ich aplikovania a pomocou nej zvolili okruh metód, ktoré sme pomocou literatúry podrobne matematicky opísali. Keďže žiadna z metód úplne nevyhovovala všetkým požiadavkám, vytvorili sme novú ad-hoc metódu založenú na Bayesovom klasifikátore a logistickej regresii. Dbali sme hlavne na použiteľnosť v praxi a výpovednú hodnotu natrénovaných parametrov. Ako doplnok k predstaveným metódam sme vytvorili postup založený na zhlukovaní metódou K-means s cieľom zlepšiť úspešnosť celého systému, pomocou nájdenia opakujúcich sa podvodov. Zhlukovanie nám umožnilo použiť aj veľkú sadu dát, ktoré neboli manuálne preverené, keďže je založené na učení bez učiteľa.

Pokračovali sme zadefinovaním metodiky testovania, ktorou sme pôvodný systém založený na expertnom odhade porovnávali s nami navrhnutými metódami. Odhad chyby poradia pri regresných stromoch bol lepší ako pri ad-hoc metóde. Zhlukovanie metódou K-means spoločne s naším experimentálnym postup na určenie rizikovosti pre jednotlivé skupiny sa ukázalo byť až prekvapivo úspešné, no naďalej sme sa ho v novom systéme rozhodli použiť ako doplnkovú metódu. Potenciál ad-hoc metódy vidíme vo veľkej podobnosti s pôvodným systémom a spojení expertných skúseností zamestnancov s hodnotami natrénovaných parametrov. Vďaka tomuto spojeniu môže byť ad-hoc metóda aj z hľadiska jednoduchosti prechodu z pôvodného systému v konečnom dôsledku najúspešnejšou. Cieľ práce sa nám podarilo v testovaní splniť pomocou klasifikačných stromov aj ad-hoc metódy, nakoľko úspešnosť identifikovania podvodov bola v prípade všetkých metód oproti pôvodnému systému lepšia. Ad-hoc metóda odhadom pomocou krížovej validácie zlepšila pôvodný systém o zhruba $1.37\% - 5.4\%$, regresné stromy o $2.51\% - 8.88\%$. Doplnkové zhlukovanie zlepšilo zoraďovanie najrizikovejších udalostí až o 10.93%. Pôvodný a nový systém sme porovnali simulovaním prvých mesiacov jeho použitia. Simulácia potvrdila vyššiu úspešnosť zoradenia pri použití regresných stro-

mov ako hlavnej, a zhlukovania ako doplnkovej metódy rádovo o viac ako 10%. Tieto odhady môžu znamenať reálne ročné ušetrenie v rádoch tisícov až desaťtisícov eur.

S výsledkami práce sme spokojní, no zároveň vidíme potenciál na zlepšovanie. Napriek odhadovanému zlepšeniu je úspešnosť zoraďovania udalostí novým systémom stále len tesne nad úrovňou 60%. Najväčšou prekážkou sa ukázal byť nedostatok dát. Tie obsahovali priveľa podvodov a nezobrazovali dostatočne presne skutočné poistné prostredie, nakoľko šlo o dáta vybrané na manuálne overenie pomocou pôvodného systému. Sme presvedčení, že v prípade zlepšenia kvality dát je potenciál na zvýšenie úspešnosti odhaľovania podvodov oveľa vyšší.

Ďalší postup súvisí priamo s prekážkami, ktorým sme pri implementácii nového systému čelili. Odporúčame nájsť spôsoby, ktorými by bolo možné zvýšiť množstvo poistných udalostí, ktoré sú jasne identifikované ako podvody, alebo je v ich prípade možnosť podvodu vylúčená. Prvou možnosťou je zdieľanie dát vrámci skupiny, teda napríklad vďaka podobností poistných trhov medzi českou a slovenskou pobočkou. Druhou možnosťou je použitie dát o poistných udalostiach, ktoré sú hneď v prvých kolách automaticky vylúčené z procesu, nakoľko je v ich prípade možnosť podvodu veľmi nízka. Tieto dáta poisťovňa štandardne nezbiera, preto sme ich nemali k dispozícii. Po získaní kvalitnejších dát je možné pokračovať aplikovaním robustnejších modelov spájajúcich interpretovateľné modely s neinterpretovateľnými, ako napríklad zoraďovanie udalostí na základe lineárnej kombinácie predikcií jednotlivých metód. Vďaka tomu sa stanú použiteľnými aj neurónové siete a iné robustné metódy s vlastnosťami "čiernych skriniek".

Zoznam použitej literatúry

[1] Slepecký, J., Daňová, M.: *Poisťovníctvo*, Vysoká škola bezpečnostného manažérstva v Košiciach, 2014

[2] Kaas, R., et al.: *Modern Actuarial Risk Theory Using R*, Springer, 2008

[3] Alpaydin, E.: *Introduction to Machine Learning*, Prentice Hall of India, 2014

[4] James, G., Witten, D., Hastie, T., Tibshirani, R.: *An Introduction to Statistical Learning with Applications in R*, Springer, 2013

[5] Rogers, S., Girolami, M.: *A first course in machine learning*, Chapman and Hall, 2017

[6] Hastie, T., Tibshirani, R., Friedman, J.: *The Elements of Statistical Learning*, Springer, 2017

[7] Haykin, S.: *Neural Networks and Learning Machines*, Pearson, 2009

[8] Izenman, A. J.: *Modern multivariate statistical techniques*, Springer, 2008

[9] Murphy, K. P.: *Machine Learning a Probabilistic Perspective*, The MIT Press, 2012

[10] Spall, J. C.: *Introduction to Stochastic search and Optimization*, Wiley-Interscience, 2003

[11] Hamala, M., Trnovská, M.: *Nelineárne programovanie*, Epos, 2012

[12] Beranová, M.: *Metody manažerského rozhodování v riziku a nejistotě*, Univerzita Tomáše Bati ve Zlíně, 2007

[13] *Insurance Fraud Handbook*, Association of Certified Fraud Examiners, 2016, dostupné na internete (7.5.2018):
http://www.acfe.com/uploadedfiles/acfe_website/content/documents/insurance−fraud−handbook.pdf

[14] Vinař, T.: *Prednášky z predmetu Strojové učenie*, 2018, dostupné na internete (7.5.2018):
http://compbio.fmph.uniba.sk/vyuka/ml/handouts/

[15] Harman, R.: *Prednášky z predmetu Viacrozmerné štatistické analýzy 2*, 2017, dostupné na internete (7.5.2018):
$http://www.iam.fmph.uniba.sk/ospm/Harman/teaching.htm\#VSA$