UNIVERZA V LJUBLJANI
EKONOMSKA FAKULTETA

MAGISTRSKO DELO

UPORABA METOD PODATKOVNEGA RUDARJENJA PRI OCENJEVANJU KREDITNEGA TVEGANJA FIZIČNIH OSEB

Ljubljana, avgust 2017 PRIMOŽ PODOBNIK

IZJAVA O AVTORSTVU

Podpisani Primož Podobnik, študent Ekonomske fakultete Univerze v Ljubljani, avtor predloženega dela z naslovom Uporaba metod podatkovnega rudarjenja pri ocenjevanju kreditnega tveganja posameznikov, pripravljenega v sodelovanju s svetovalcem prof. dr. Igorjem Lončarskim

IZJAVLJAM

1. da sem predloženo delo pripravil samostojno;

2. da je tiskana oblika predloženega dela istovetna njegovi elektronski obliki;

3. da je besedilo predloženega dela jezikovno korektno in tehnično pripravljeno v skladu z Navodili za izdelavo zaključnih nalog Ekonomske fakultete Univerze v Ljubljani, kar pomeni, da sem poskrbel, da so dela in mnenja drugih avtorjev oziroma avtoric, ki jih uporabljam oziroma navajam v besedilu, citirana oziroma povzeta v skladu z Navodili za izdelavo zaključnih nalog Ekonomske fakultete Univerze v Ljubljani;

4. da se zavedam, da je plagiatorstvo – predstavljanje tujih del (v pisni ali grafični obliki) kot mojih lastnih – kaznivo po Kazenskem zakoniku Republike Slovenije;

5. da se zavedam posledic, ki bi jih na osnovi predloženega dela dokazano plagiatorstvo lahko predstavljalo za moj status na Ekonomski fakulteti Univerze v Ljubljani v skladu z relevantnim pravilnikom;

6. da sem pridobil vsa potrebna dovoljenja za uporabo podatkov in avtorskih del v predloženem delu in jih v njem jasno označil;

7. da sem pri pripravi predloženega dela ravnal v skladu z etičnimi načeli in, kjer je to potrebno, za raziskavo pridobil soglasje etične komisije;

8. da soglašam, da se elektronska oblika predloženega dela uporabi za preverjanje podobnosti vsebine z drugimi deli s programsko opremo za preverjanje podobnosti vsebine, ki je povezana s študijskim informacijskim sistemom članice;

9. da na Univerzo v Ljubljani neodplačno, neizključno, prostorsko in časovno neomejeno prenašam pravico shranitve predloženega dela v elektronski obliki, pravico reproduciranja ter pravico dajanja predloženega dela na voljo javnosti na svetovnem spletu preko Repozitorija Univerze v Ljubljani;

10. da hkrati z objavo predloženega dela dovoljujem objavo svojih osebnih podatkov, ki so navedeni v njem in v tej izjavi.

V Ljubljani, dne _____ Podpis študenta:_____

KAZALO

UVOD **3**

1 KREDITNO TVEGANJE V BANČNIŠTVU **7**

1.1 BASEL II in BASEL III 10

1.2 Ocenjevanje kreditnega tveganja 12

1.2.1 Diskriminantna analiza 13

1.2.2 Logistična regresija 14

1.3 Metode podatkovnega rudarjenja 15

1.3.1 Nevronske mreže 16

1.3.2 Metoda podpornih vektorjev 20

1.3.3 Odločitvena drevesa 23

1.3.4 K-najbližjih sosedov 25

1.3.5 Genetski algoritmi 26

2 PRIPRAVA PODATKOV ZA ANALIZO **28**

2.1 Vzorčenje 28

2.2 Določanje tipa podatkov 29

2.3 Vizualna in statistična raziskava podatkov 29

2.4 Manjkajoče vrednost 29

2.5 Ugotavljanje in reševanje napačnih vrednosti 30

2.6 Standardizacija podatkov 30

2.7 Kategorizacija 31

2.8 Določanje spremenljivk 31

2.9 Proces podatkovnega rudarjenja 32

3 ANALIZA PODATKOV **33**

3.1 Opisne statistike spremenljivk 35

3.2 Čiščenje in redukcija podatkov 37

3.3 Določanje spremenljivk 38

4 EMPIRIČNI REZULTATI **40**

4.1 Diskriminantna analiza 40

4.2 Logistična regresija 42

4.3 Nevronske mreže 43

4.4 Metoda podpornih vektorjev 46

4.5 Odločitvena drevesa 47

4.6 K-najbljižjih sosedov 49

4.7 Genetski algoritmi 51
SKLEP 53
LITERATURA IN VIRI 57
PRILOGE 1

KAZALO TABEL

Tabela 1: Razlaga pomena spremenljivk v podatkih in tip spremenljivke 34
Tabela 2: Korelacijska matrika spremenljivk 39
Tabela 3: Velikost efekta posamezne spremenljivke merjena s parametrom delni η^2 39
Tabela 4: Ocene koeficientov pri diskriminantni analizi 40
Tabela 5: Ocene koeficientov in njihove standardne napake pri logistični regresiji 43
Tabela 6: Opisne statistike posamezne spremenljivke v izbranem vzorcu 44
Tabela 7: Statistične karakteristike vzorca 46
Tabela 8: Osnovne statistične karakteristike vzorca 51
Tabela 9: Odvisnost natančnosti (v %) od velikosti generacije in verjetnosti mutacije 52
Tabela 10: Primerjava natančnosti metod pri enaki velikosti vzorca 54
Tabela 11: Najvišja natančnost posamezne metode in velikost vzorca, kjer je bila dosežena 54
Tabela 12: Najnižja natančnost posamezne metode in razpon natančnosti 55
Tabela 13: Primerjava potencialnih izgub pri uporabi različnih metod 56

KAZALO SLIK

Slika 1: Prikaz logistične regresije s pomočjo nevronskih mrež 16
Slika 2: MLP nevronska mreža 18
Slika 3: Razdelitev po metodi podpornih vektorjev v primeru popolne linearne ločljivosti 21
Slika 4: Primer odločitvenega drevesa 24
Slika 5: Odvisnost natančnosti od velikosti vzorca pri diskriminantni analizi 42
Slika 6: Odvisnost natančnosti logistične regresije od velikosti vzorca 43
Slika 7: Nevronska mreža 45
Slika 8: Primerjava natančnosti nevronskih mrež glede na velikost vzorca 45
Slika 14: Primerjava natančnosti metode podpornih vektorjev glede na velikost vzorca 47
Slika 9: Graf natančnosti modela odločitvenih dreves glede na število minimalnih delitev 48
Slika 10: Odločitveno drevo 48
Slika 11: Vpliv velikosti vzorca na natančnost odločitvenih dreves 49
Slika 12: Natančnost metode k-najbližjih sosedov pri različnih parametrih k 50
Slika 13: Odvisnost natančnost metode k-najbližjih sosedov od velikosti vzorca 51
Slika 15: Odvisnost natančnosti od velikosti vzorca pri genetskem algoritmu 53

UVOD

Tveganje je v bankah prisotno že od samega začetka in skozi celotno zgodovino bančništva. Eno bolj prisotnih tveganj je kreditno tveganje ali z drugimi besedami tveganje banke, da ne dobi povrnjeno posojenega. Obvladovanje kreditnih tveganj se je razvijalo skozi celotno zgodovino bančništva, saj banke z le-tem zmanjšajo potencialne izgube in posledične ekonomske in socialne krize. Na obvladovanje kreditnih tveganj lahko gledamo kot na proces, ki v grobem sestoji iz zaznavanja potencialnih tveganj, merjenja potencialnih tveganj, primernega razreševanja in dejanske implementacije modelov, ki ocenjujejo kreditno tveganje.

Kreditno tveganje lahko v grobem razdelimo na tri komponente: verjetnost neizpolnitve, izguba ob neizpolnitvi in izpostavljenost ob neizpolnitvi obveznosti. V nalogi bom preučeval natančnost različno izbranih metod pri ocenjevanju verjetnosti neizpolnitve.

Dandanes se zaradi velike količine dostopnih podatkov za analizo le-teh vse pogosteje uporablja t.i. metode podatkovnega rudarjenja, ki predstavljajo alternativo tradicionalnim metodam analize in ocenjevanja podatkov. Te metode lažje prepoznavajo vzorce v veliki količini podatkov, na podlagi katerih lahko potem pridemo do kakšnih zaključkov. V bančništvu zaradi strogih regulacij uporaba teh metod ni pretirano razširjena (Baesens & Van Gestel, 2009).

V magistrskem delu bom predstavil primerjavo natančnosti različnih metod pri ocenjevanju verjetnosti neizpolnjevanja obveznosti posameznikov. Primerjava je narejena med tradicionalnimi metodami kot je npr. logistična regresija in alternativnimi pristopi kot je npr. metoda podpornih vektorjev.

Do sedaj je bilo na področju podatkovnega rudarjenja razvitih in raziskanih že precej metod. S čedalje večjo količino podatkov pridejo le-te do izraza, saj nam omogočajo drugačen pogled na same podatke z odkrivanjem podobnih vzorcev, grupiranjem itd. V finančnem svetu sta zelo uporabni metodi nevronskih mrež in odločitvenih dreves, ki pripomoreta k dodatni finančni analizi. Po drugi strani uporaba metod podatkovnega rudarjenja zahteva več zgodovinskih podatkov kot standardne metode, prav tako pa jih je težje interpretirati. Nekateri avtorji (Chirani, Nashtaei, & Takyar, 2015; Delihodzic & Donko, 2013; Gouvea & Goncalves, 2007) so naredili primerjave metod podatkovnega rudarjenja s standardnimi modeli, kjer je bilo ugotovljeno, da se nekatere metode podatkovnega rudarjenja obnesejo bolje. Seveda se poraja tudi vprašanje, kako učinkovite dejansko metode podatkovnega rudarjenja so. Pokazano je bilo, da imajo nevronske mreže dobro sposobnost napovedovanja in predvidevanja takrat, ko so efektivno implementirane in preverjene (Angelini, Di Tollo, & Roli, 2008). Med strokovnjaki še vedno prevladuje mešano mnenje o njihovem prispevku in uporabi, saj imamo na eni strani modele

(logistična regresija), ki jih je lažje interpretirati, na drugi strani pa modele, ki imajo boljšo sposobnost napovedovanja (nevronske mreže).

Cilj magistrskega dela je primerjati alternativne metode ocenjevanja kreditnih tveganj (nevronske mreže, odločitvena drevesa, metoda podpornih vektorjev, genetski algoritmi, k-najbližjih sosedov), s tradicionalnimi metodami ocenjevanja kreditnih tveganj (diskriminantna analiza, logistična regresija). Nadalje bom v magistrski nalogi raziskal, kako različna velikost vhodnih podatkov vpliva na natančnost zgoraj naštetih metod in kako na natančnost vplivajo različno izbrani parametri teh metod. Nazadnje se bom še dotaknil vprašanja, kaj bi za banke pomenila izbira natančnejše metode.

Ključna vprašanja, na katera bom tekom magistrskega dela skušal podati odgovor, so naslednja:

1. Kateri od modelov se bolje obnese pri ocenjevanju posameznikovega kreditnega tveganja: diskriminantna analiza, nevronske mreže, logistična regresija, odločitvena drevesa, metoda podpornih vektorjev, k-najbližjih sosedov ali genetski algoritmi?
2. Kako vpliva velikost vzorca na natančnost metod? Kako različno izbrani parametri vplivajo na natančnost metod?
3. Ali je za banko smiselno uporabiti model, ki po eni strani bolje napove verjetnost neizpolnjevanja obveznosti, po drugi strani pa ga je težje interpretirati in implementirati?

Ocenjevanje kreditnega tveganja je eno najpomembnejših področij, s katerimi se ukvarja obvladovanje finančnih tveganj. Osnovne smernice obvladovanj kreditnega tveganja so predstavljene s t.i. baselskim kapitalskim sporazumom (angl. *Basel Capital Acorrd)*. Do sedaj so bili predstavljeni standardi Basel I, Basel II in Basel III. Na tem mestu velja poudariti, da se je implementacija standarda Basel III pričela leta 2013 in še ni končana. Banke se s kreditnimi tveganji soočajo na dnevni ravni. Zmožnost ločevanja med dobrimi in slabimi posojilojemalci je ena ključnih nalog, ki jih opravljajo. V ta namen banke uporabljajo različne metode ocenjevanja kreditnega tveganja posameznikov, ki želijo prejeti posojilo.

Prvi najden zapis posojila sega v antični Babilon. Lewis (1992) je v svoji študiji zapisal, da je na kamniti plošči stari okoli 5000 let napisano sledeče: "Mas-Schamach, sin Adadrimena si je od svečenice boga sonca Amat-Schamach, hčere Warad-Enlil, sposodil dva šekela srebra. Sončni svečenici bo Mas-Schamach plačal obresti. V času žetve bo vrnil izposojeno in do takrat nabrane obresti." Ta zapis dokazuje, da so se že takrat kmetje soočali s problemom denarnih tokov, ki so ga reševali z izposojo ob setvi in vrnitvijo z obrestmi ob žetvi. Samo ocenjevanje kreditnega tveganja pa ima precej krajšo zgodovino, njegovi začetki segajo v prvo polovico dvajsetega stoletja. Ocenjevanje kreditnega tveganja je dejansko problem razvrstitve populacije na dve podskupini, dobro in slabo. Prvi, ki je

reševal ta problem, je bil Fisher (1936), ki je v svoji študiji raziskoval razlikovanje različnih vrst irisa glede na velikost rastlin. Durand (1941) je kasneje prepoznal, da bi lahko tehnike iz Fisherjeve raziskave uporabili tudi pri razločevanju dobrih in slabih posojil. Med drugo svetovno vojno so imele finančne institucije zaradi pomanjkanja kadra veliko težav pri ocenjevanju kreditnega tveganja. Veliko odločitev so sprejemali na podlagi neutemeljenih ocen narejenih preko palca (Crook, Edelman, & Thomas, 2002).

Kmalu po koncu druge svetovne vojne se je pričela uporaba statističnih razvrstitvenih tehnik in statističnih modelov (Wonderlic, 1952). Uporabnost in razcvet ocenjevanja kreditnega tveganja pa je postala razvidna koncem šestdesetih let prejšnjega stoletja s pričetkom uporabe kreditnih kartic. Število prosilcev je skokovito naraslo, kar je privedlo do potrebe po avtomatizaciji odločanja o posojilih. Približno istočasno se je začela razvijati računalniška tehnologija, kar je omogočilo avtomatizacijo procesa. Myers in Forgy (1963) sta ugotovila, da je takrat novejši avtomatiziran način ocenjevanja kreditnega tveganja precej učinkovitejši od tradicionalnega, saj se je stopnja neizpolnjevanja obveznosti na posojilih znižala za več kot 50%. Nekateri tradicionalisti (Capon, 1982) so trdili, da surovo empirično ocenjevanje kreditnega tveganja žali tradicijo družbe. Zagovarjali so dejstvo, da bi morale biti kreditne ocene bolj odvisne od preteklih kreditov. Prav tako so zahtevali pojasnilo, zakaj so nekatere lastnosti uporabljene za ocenjevanje in druge ne. Ocenjevanje kreditnega tveganja s statističnimi metodami je bilo dokončno sprejeto po uveljavi aktov o enakih kreditnih priložnostih (angl. *Equal Credit Opportunity Acts*) v Združenih državah leta 1975 in 1976 (Crook et al., 2002).

V osemdesetih letih prejšnjega stoletja so banke pričele uporabljati modele ocenjevanja kreditnega tveganja tudi za osebna posojila, v devetdesetih letih prejšnjega stoletja pa so se te metode pričele uporabljati tudi za namene tržnih raziskav. V istem obdobju se je prav tako razširila uporaba logistične regresije in linearnega programiranja (Crook et al., 2002).

V zadnjih letih se je ob povečani količini podatkov v bančništvu pokazala potreba po drugačnih pristopih k upravljanju s kreditnimi tveganji. Alternativo tradicionalni diskriminantni analizi in logistični regresiji predstavljajo metode podatkovnega rudarjenja. Te za razliko od tradicionalnih metod iščejo vzorce v podatkih in ne predpostavljajo odvisnosti spremenljivk kot npr. pri logistični regresiji. To prednost lahko dandanes ob povečani računski moči in količini podatkov s pridom izkoristimo tudi pri obvladovanju kreditnih tveganj. Najpogosteje omenjene metode podatkovnega rudarjenja za ocenjevanje kreditnih tveganj so nevronske mreže in odločitvena drevesa, uporabljajo pa se tudi metode k-najbližjih sosedov, genetski algoritmi in metoda podpornih vektorjev (Crook et al., 2002).

Nevronska mreža je metoda za obdelavo informacij, ki deluje po vzoru človeških možganov. Osnovni gradniki metode so t.i. nevroni, ki imajo več različno uteženih vhodov, medsebojne povezave ter en izhod. Bistvo nevronskih mrež je v tem, da med

učenjem same ugotovijo povezavo med vhodnimi podatki in ciljno spremenljivko. Ko je nevronska mreža naučena, lahko rešuje tudi probleme, s katerimi v procesu učenja ni imela opravka.

Odločitveno drevo je metoda, ki sloni na grafični upodobitvi drevesa. Drevo sestavljajo koren, notranja vozlišča in listi. Odločitvena drevesa so zelo uporabna, ker jih je lahko interpretirati, po drugi strani pa lahko že zelo enostaven primer razvrščanja privede do časovno zelo zahtevnega postopka.

K-najbližjih sosedov je postopek za klasifikacijo primerov na podlagi najbližjih primerov iz učne množice vseh danih karakteristik. Parameter k je naravno število, ki ga poljubno izberemo na začetku postopka. Tipično je vrednost parametra majhna.

Osnovna naloga metode podpornih vektorjev je ločiti podana razreda glede na dane karakteristike (v primeru kreditnih tveganj ločiti dobre od slabih posojilojemalcev glede na dane karakteristike). Metoda je zelo primerna za učenje na velikih množicah z velikim številom nepomembnih karakteristik. Slaba stran metode je v njeni časovni zahtevnosti in interpretaciji odločitev.

Genetski algoritem je postopek optimizacije po vzoru biološke evolucije. Evolucijo lahko preprosto opišemo s tremi parametri: mutacijo, selekcijo in križanjem. Genetski algoritmi nam omogočajo iskanje rešitev v celotni populaciji, zato lahko najdemo več potencialnih rešitev hkrati. Slabost genetskih algoritmov je, da je pri mutaciji potrebna sreča za hitro in kvalitetno rešitev, torej so lahko zelo potratni.

Prvi je primerjal diskriminantno analizo z logistično regresijo Wiginton (1980). Uporabo odločitvenih dreves pri ocenjevanju kreditnega tveganja je prvi raziskoval Makowski (1985). Metodo k-najbližjih sosedov sta v kontekstu ocenjevanja kreditnega tveganja prvič uporabila Charttejee in Barcun (1970). Nekoliko kasneje sta uporabo metode najbližjih sosedov pri ocenjevanju kreditnega tveganja raziskovala tudi Hand in Henley. Metoda nevronskih mrež je bila pri ocenjevanju kreditnega tveganja največkrat uporabljena. Njeno uporabnost pri ocenjevanju posameznikovega kreditnega tveganja so raziskovali Malhotra, Malhotra in R.W.McLeod (1994). Nevronske mreže so prav tako uporabljali pri ocenjevanju kreditnega tveganja podjetij, in sicer so na to temo objavljali Altman, Marco in Varreto (1994), ki so raziskovali uporabo nevronskih mrež za napovedovanje propada podjetij v Italiji, Coats in Fant (1993) pa sta raziskovala uporabo nevronskih mrež za napovedovanje propada podjetij v ZDA. Metoda podpornih vektorjev pri ocenjevanju kreditnega tveganja se pri ocenjevanju kreditnega tveganja uporablja šele v zadnjih 15 letih. Verjetno prvi, ki je primerjal metodo podpornih vektorjev z ostalimi metodami ocenjevanja kreditnega tveganja, je Baesens, ki je v soavtorstvu z Van Gestelom, Viaene, Stepanovo, Suykens in J. Vanthienenom ugotovil, da metoda podpornih vektorjev ni vedno najboljša za ocenjevanje kreditnega tveganja.

V magistrskem delu nameravam primerjati različne metode podatkovnega rudarjenja s tradicionalnimi metodami ocenjevanja kreditnega tveganja.

Uvodno poglavje magistrskega dela predstavi problem kreditnega tveganja in na kratko osvetli razvoj ocenjevanja kreditnega tveganja skozi zgovodino. Skozi uvodna poglavja je podan tudi kratek opis metod podatkovnega rudarjenja, ki se bodo uporabila v nadaljevanju magistrske naloge.

Naslednje poglavje predstavlja smernice obvladovanja kreditnega tveganja v bančništvu, ki so predstavljene z mednarodnimi standardi Basel II in Basel III.

Magistrsko delo nadaljujem s podrobnejšim opisom metod ocenjevanja kreditnega tveganja. Podrobneje so opisane naslednje metode: diskriminantna analiza, logistična regresija, nevronske mreže, metoda podpornih vektorjev, metoda k-najbljižjih sosedov, odločitvena drevesa in genetski algoritmi. Za boljšo ponazoritev delovanja metod podatkovnega rudarjenja so pri nekaterih opisih podani tudi enostavni ilustrativni primeri.

Delo se nadaljuje s poglavjem Priprava podatkov za analizo, ki nam opiše proces obdelovanja podatkov in pripravo le-teh pred začetkom izvajanja metod ocenjevanja kreditnega tveganja. Kratko opišem vzorčenje, določanje tipa podatkov, vizualno in statistično raziskavo podatkov, obvladovanje manjkajočih in napačnih vrednosti, standardizacijo in kategorizacijo podatkov, določanje spremenljivk, končam pa z opisom procesa podatkovnega rudarjenja.

V nadaljevanju naloge sledi empirični del, ki sestoji iz analize podatkov, predstavitve empiričnih rezultatov in se zaključi s sklepom.

Na samem koncu magistrskega dela so kot priloge priložene še programske kode, ki sem jih uporabil za ocenjevanje verjetnosti neizpolnjevanja obveznosti posameznikov.

1 KREDITNO TVEGANJE V BANČNIŠTVU

Kreditno tveganje je oblika tveganja, ki je v bančništvu zaradi same narave poslovanja najbolj prisotna. Izmed vseh oblik tveganja v bančništvu, kreditno predstavlja potencialno največjo izgubo. Kreditno tveganje lahko definiramo kot tveganje, da gre posojilojemalec v stečaj in s tem ne izpolnjuje svojih obveznosti do banke v obliki odplačevanja dolga. Pojavi se takrat, ko posojilojemalci dolga niso zmožni odplačevati pravočasno ali pa ga sploh niso zmožni odplačevati. Najpogostejši razlog nezmožnosti odplačevanja dolga je finančna stiska posojilojemalca. Stečaji se pogosto pojavijo tudi zaradi nerazumevanja samega procesa odplačevanja dolga, ki se lahko pojavi zaradi tehničnih napak ali napak v informacijskih sistemih. Banke lahko utrpijo izgube tudi pri investiranju v posojilo

posojilojemalcu, ki zgleda kot dober, a se mu nato njegova kreditna ocena poslabša, kar potencialno lahko vodi do njegove nezmožnosti odplačevanja dolga. Pri postopku likvidacije se ustvarja neto izguba, saj je cena kredita na trgu nižja od cene kredita, ki ga je kupila banka. V primeru stečaja nasprotne strani je izguba banke odvisna od same narave kreditne pogodbe. Če so v samo pogodbo vključena tudi jamstva in garancije, izguba banke ni nujno tako visoka. Izguba v primeru stečaja pa je odvisna tudi od neposredne izpostavljenosti banke stečajnemu posojilojemalcu (Baesens & Van Gestel, 2009).

Pri ocenjevanju kreditnega tveganja je potrebno biti nekoliko pazljiv, saj merjenje kreditnega tveganja predstavlja nemalo težav. Tradicionalno se kreditno oceno dolgov meri z lestvicami (angl. r*ating*), ki predstavljajo zgolj številsko oceno dolga. Banke večinoma uporabljajo notranje lestvice, ki so prilagojene njihovim potrebam. Slaba stran lestvic je ta, da ne merijo tveganja celotnega portfelja, temveč merijo tveganje zgolj na individualnem posojilu. Merjenje kreditnega tveganja na celotnem portfelju je kritično za ugotavljanje potencialnih izgub in s tem povezanim kapitalom, ki je potreben za absorbcijo le-teh. Prav tako je težko meriti vpliv na celotnem portfelju, saj sta lahko tveganji dveh posameznih posojil korelirani, s čimer se tveganje njunih posojil poveča. Glavni problem, s katerim se tu soočajo razvijalci modelov za ocenjevanje tveganja, je pomanjkanje podatkov, s katerimi bi lahko ocenili tveganje na celotnem portfelju (Bessis, 2002).

Kreditno tveganje je sestavljeno iz tveganja pred poravnavo (angl. *pre-settlement risk*) in tveganja poravnave (angl. *settlement risk*).

Tveganje pred poravnavo se nanaša na potencialne izgube, ki nastanejo zaradi stečaja posojilojemalca med samim obdobjem transakcije (bodisi transakcije posojila, obveznic, izvedenih finančnih inštrumentov, itd.). Tveganje pred poravnavo traja od začetka veljave same pogodbe do konca poravnave (Baesens & Van Gestel, 2009).

Tveganje poravnave nastane zaradi posrednega odplačevanja posojila. Posojilojemalec lahko banki, ki mu je odobrila kredit, tega vrača preko transakcij druge banke, ki grejo lahko med samim potekom transakcij v stečaj. Ta oblika tveganje je prisotna od trenutka, ko druga banka od posojilojemalca prejme plačilo, do trenutka, ko prva banka prejme to transakcijo. Daljši kot je čas med obema transakcijama, višje je tveganje poravnave. Tej obliki tveganja so bolj izpostavljene višje transakcije in transakcije narejene v različnih časovnih pasovih (Baesens & Van Gestel, 2009).

Kreditno tveganje najlažje predstavimo s tremi komponentami in sicer s tveganjem neizpolnitve (angl. *default risk*), izgubo ob neizpolnitvi (angl. *loss given default,* v nadaljevanju LGD) in izpostavljenostjo ob neizpolnitvi (angl. *exposure at default,* v nadaljevanju EAD) (Baesens & Van Gestel, 2009).

Tveganje neizpolnitve je definirano kot verjetnost, da nasprotna stran ne izpolnjuje obveznosti, kar imenujemo tudi **verjetnost neizpolnitve** (angl. *probability of default*, v nadaljevanju PD). Iz same narave definicije verjetnosti sledi, da je verjetnost neizpolnitve številski izraz, ki leži med 0 in 1. Najpogostejša definicija neizpolnitve obveznosti je zamuda plačila za vsaj 3 mesece. Najpogosteje uporabljeni dejavniki, ki vplivajo na višino verjetnosti neizpolnitve so finančno stanje posojilojemalca, višina dolga in finančni prihodek. Slabo finančno stanje, velik dolg in nizek ter nestanoviten finančni prihodek vplivajo na višjo verjetnost neizpolnitve. Nekateri kvalitativni dejavniki, ki vplivajo na verjetnost neizpolnitve so kvaliteta upravljanja in različne informacije iz oddelkov. Na trgih, kjer je prisotna velika konkurenca, makroekonomska recesija in znižanje stopnje industrializacije, je pričakovana verjetnost neizpolnitve višja. Prejemanje sredstev iz drugih virov (torej ne zgolj samofinanciranje) zniža verjetnost neizpolnitve. Tveganje neizpolnitve je ocenjeno interno s strani bank običajno na podlagi sistema kreditnega ocenjevanja. Prav tako je lahko verjetnost neizpolnitve ocenjena neodvisno s strani bonitetnih agencij. Verjetnost neizpolnitve običajno merimo na posojilojemalcu, ne na produktih. Pri individualnih potrošnikih je opaziti višjo verjetnost neizpolnitve pri kreditnih karticah kot pri hipotekah. Posamezniki raje prenehajo izpolnjevati obveznosti na manj tveganem produktu kot na hipotekah, saj ne želijo trpeti stanovanjske krize. Dejansko izgubo banke ob neizpolnitve pa nam merita karakteristiki izguba in izpostavljenost ob nezpolnitvi (Baesens & Van Gestel, 2009).

Izguba ob neizpolnitvi ali LGD je določena kot delež izpostavljenosti v primeru neizpolnitve obveznosti. V primeru, ko ni izgub je izguba ob neizpolnitvi enaka 0, v primeru ko je banka polno izpostavljena pa je njena izguba ob neizpolnitvi enaka 100%. LGD ima lahko tudi negativen predznak, kar nakazuje na dobiček (navadno zaradi obrestnih mer ali kazenskih taks). LGD lahko tudi preseže 100% v primeru, visoke izpostavljenosti in dodatnih morebitnih stroškov, ki pri tem nastanejo (npr. stroški sodne poravnave). Vrednost LGD variira glede na naravo produkta. Vrsta neizpolnitve obveznosti znatno vpliva na dejanske izgube. Problem nastane, ker sama vrsta neizpolnitve morda ni znana v trenutku neizpolnitve in zagotovo ni znana v trenutku investicije. V primeru neizpolnitve posojilojemalcev imajo banke pravico do pravnega postopka. Dejavnike, ki opisujejo vrsto neizpolnitve in njegovo rešitev, lahko v grobem razdelimo na finančno zdravje podjetja ali posameznika, možnost prestrukturiranja in možnost likvidacije. V primeru, ko je finančno zdravje podjetja ali posameznika kmalu po neizpolnitvi popravljeno, banka ne utrpi znatne izgube, saj lahko podjetje ali posameznik nadaljuje z odplačevanjem dolga. Prav tako to nima znatnega vpliva na sam odnos banka-stranka. Samo prestrukturiranje dolga slabše vpliva na banko kot izboljšava finančnega zdravja posojilojemalca. Odnos banka-stranka je načet, vendar se načeloma ohrani, banka pa sprejme zmerne izgube, da bi se izognila večjim, ki nastanejo, če se zgodi postopek likvidacije ali neizpolnitve obveznosti. V primeru likvidacije banka zaseže posojilojemalčevo jamstvo. Odnos banka-stranka se konča in banko običajno prizadenejo visoke izgube. Sam tip razrešitve neizpolnitve je težko predvideti, preden se dejansko

zgodi. Predvideva se, da je likvidacija pogostejša med šibkejšimi strankami. V primeru visokega tveganja neizpolnitve ali izgube banke navadno zahtevajo jamstvo ali garancijo. Z jamstvom ali garancijo banko pokrijejo izgubo neplačanega dolga, ki nastane v primeru neizpolnitve obveznosti stranke. Vrednost parametra LGD je odvisna tako od vrednosti jamstva v času prodaje kot tudi od same pravne zmožnosti zasega in prodaje le tega. Vključitev garancij v pogodbo o dolgu bolje zaščiti banko, saj je porok načeloma manj tvegan in ni odvisen od dolžnikovega dolga. Banke, ki vlagajo v prednostni dolg, imajo ob morebitni neizpolnitvi obveznosti več pravic (Baesens & Van Gestel, 2009).

Izpostavljenost ob izgubi ali EAD ni vnaprej znana. Velikost izpostavljenosti je odvisna od same narave posojila in pri nekaterih vrstah ni vnaprej znana (npr. kreditne kartice), pri nekaterih pa je (obveznice, kredit). Tveganje izpostavljenosti nastane zaradi neznane velikosti izpostavljenosti pri morebitni neizpolnitvi obveznosti nasprotne strani. Običajno opazimo, da je nasprotna stran v finančni stiski takrat, ko ima visoke likvidnostne potrebe, kar se kaže kot uporaba kreditnih instrumentov do njihovega limita. Banka na drugi strani želi zaščito pred takim obnašanjem, kar doseže z dodajanjem pogodbenih klavzul v obliki znižanega kreditnega limita ali spremenitvi pogodbe ob nastopu določenega dogodka. Tem klavzulam pravimo tudi pogodbene zaveze ali materialne neželene klavzule (angl. *material adverse clauses*). Poleg samih lastnosti pogodbe in klavzul, je izpostavljenost tveganju odvisna tudi od karakteristik posojilojemalca in splošnega stanja gospodarstva. Izpostavljenost tveganju običajno izražamo v denarni enoti pogodbe (Baesens & Van Gestel, 2009).

Na vse tri oblike kreditnega tveganja (PD, LGD in EAD) vpliva tudi ročnost dolga. Daljša kot je ročnost, bolj negotova je pogodba in s tem tudi tveganje. Najbolj razvito ocenjevanje, obvladovanje in upravljanje kreditnega tveganja je ravno za pogodbe z enoletno ročnostjo. Nov bančni standard Basel posveča veliko pozornosti EAD in LGD tveganju. Za koherentno merjenje in upravljanje kreditnega tveganja potrebujemo natančne definicije spremenljivk PD, LGD in EAD. Spremenljivki EAD in LGD sta odvisni od same definicije neizpolnitve obveznosti, ki mora biti konsistentna in koherentna za pravilno izražanje kreditnega tveganja. Mednarodni bančni standard Basel II podaja prve smernice k enakomerni definiciji neizpolnitve obveznosti, kar nam zagotavlja tudi smernice za definicijo izpostavljenosti ob izgubi in izgubo ob neizpolnitvi (Baesens & Van Gestel, 2009).

1.1 BASEL II in BASEL III

Smernice obvladovanja kreditnega tveganja v bančništvu so predstavljene z mednarodnimi standardi Basel II in Basel III. V okviru standarda Basel II je predstavljen standardiziran postopek obvladovanja kreditnega tveganja v bankah. Bankam je dovoljeno uporabljati tudi svoj notranji postopek obvladovanja kreditnega tveganja.

Glede na standard Basel II lahko terjatev vključimo v regulatorni portfelj posameznikov, če izpolnjuje naslednje štiri pogoje:

- Orientacijski kriterij je opredeljen kot izpostavljenost banke individualni osebi ali osebam ali maloprodaji.
- Produktni kriterij je opredeljen kot izpostavljenost banke v eni od naslednjih oblik: obnovljiv kredit in kreditne linije (tu so vključene tudi kreditne kartice in prekoračitve stanja na računu), osebna dolgoročna posojila in najemi (avtomobilska posojila in najemi, študentska in druga posojila namenjena izobraževanju, osebne finance, obročna posojila) ter maloprodajni objekti in obveznosti. Iz te kategorije so posebej izključeni vrednostni papirji (obveznice, delnice) ter hipotekarna posojila, če se le-ta ne kvalificirajo kot terjatve, zavarovane s stanovanjskimi nepremičninami.
- Kriteriju razdrobljenosti se zadosti z zadostno razdrobljenostjo regulatornega portfelja posameznikov tako, da le-ta zmanjšuje tveganje v portfelju.
- Nizka vrednost posameznikove izpostavljenosti je opredeljena kot najvišja izpostavljenost eni maloprodajni nasprotni strani, ki v celoti ne sme presegati enega milijona evrov.

Če posameznika lahko vključimo v bančni regulatorni portfelj posameznikov, je njegova utež tveganja postavljena na 75%, razen v primeru, ko je posameznik v preteklosti že zamujal s plačili. Uteži tveganja morajo oceniti državne nadzorne oblasti glede na njihove pretekle izkušnje s tako vrsto izpostavljenosti. Nadzorniki lahko bankam priporočijo ustrezno prilagoditev uteži tveganja.

Glede na standard Basel II se uteži tveganja zapadlim posojilom dodelijo nekoliko drugače. Nezavarovanemu delu katerega koli posojila (razen kvalificiranega stanovanjskega hipotekarnega kredita), ki je zapadel več kot 90 dni, brez posebnih določb (vključeno delni odpis dolga), se dodeli uteži tveganja na naslednji način:

- 150%, če posebne določbe predstavljajo manj kot 20% neporavnanega dela dolga,
- 100%, če posebne določbe predstavljajo več kot 20% neporavnanega dela dolga,
- 100%, če posebne določbe predstavljajo manj kot 50% neporavnanega dela dolga, vendar se lahko utež tveganja z diskrecijsko pravico nadzornih organov zniža na 50%.

Zapadla posojila se izključijo iz celotnega maloprodajnega regulatornega portfelja za namene določanja uteži tveganja glede na kriterij razdrobljenosti. Stanovanjskim hipotekarnim posojilom, ki so zapadla več kot 90 dni, se dodeli utež tveganja 100% brez posebnih določb. Če so ta posojila zapadla in predstavljajo njihove posebne določbe manj kot 20% neporavnanega zneska, se lahko utež tveganja preostalega neporavnanega zneska zniža na 50% na državni presoji.

1.2 Ocenjevanje kreditnega tveganja

Ocenjevanje kreditnega tveganja je namenjeno predvidevanju kreditnega tveganja in ne njegovi razlagi. V zadnjem stoletju je bila glavna naloga ocenjevanja kreditnega tveganja predvideti, ali bo posojilojemalec prenehal izpolnjevati svoje obveznosti do banke.Banke z ocenjevanjem kreditnega tveganja ne skušajo oceniti zgolj verjetnosti posameznikovega neizpolnjevanja obveznosti, temveč skušajo predvideti tudi ali bo posameznik produkt dejansko uporabljal, ali bo posameznik med samim trajanjem pogodbe odšel k drugemu posojilojemalcu ter morebitne lažnive prijave za posojila. Ocenjevanje kreditnega tveganja je v bistvu predvidevanje tveganja na podlagi analize narejene z odločitvenimi modeli, samo ocenjevanje pa ne pojasnjuje razlogov za nastanek stečaja. Glavna pozitivna stran ocenjevanja kreditnega tveganja je njegova zdrava metodologija in empirična uporaba podatkov za izračun ocene kreditnega tveganja. Za ocenjevanje kreditnega tveganja uporabljamo tehnike odločitvenih modelov. Ti modeli nam pomagajo pri razločevanju med dobrimi in slabimi prosilci posojila, prav tako pa nam pomagajo določiti višino kredita, ki bi ga posameznik bil sposoben odplačati. Pri samem terminu kreditna sposobnost moramo biti nekoliko pazljivi, saj ni lastnost posameznika kot na primer višina, teža ali njegov prihodek. Kreditna sposobnost je posojilodajalčeva ocena posojilojemalca, ki izraža tudi posojilodajalčev pogled na verjetno prihodnje gospodarsko stanje. Zaradi tega bosta različna posojilodajalca drugače ocenila iste posojilojemalce. Dolgoročna slabost ocenjevanja kreditnega tveganja je ta, da teoretično lahko obstajajo posojilojemalci, ki dobijo posojilo pri vsakem posojilodajalcu in posojilojemalci, ki ne dobijo posojila pri nobenem posojilodajalcu. Posojilodajalec se mora odločiti, komu od novih prosilcev bo odobril kredit in kako upravljati s prosilci, ki že imajo pogodbo. Uporaba in robustnost metod odločitvenih modelov temelji na predpostavki, da imamo dobre zgodovinske podatke prosilcev kredita. Veliko je odvisnega od samega vzorca, na podlagi katerega nato ocenjujemo kreditno tveganje novega prosilca (Crook et al., 2002).

Ocenjevanje kreditnega tveganja temelji torej na predpostavki, da se bodo novi prosilci obnašali podobno kot pretekli prosilci s podobnimi karakteristikami. Ocena novih prosilcev je torej dobljena z obdelavo podatkov preteklih posojilojemalcev. V primeru, ko nimamo dovolj preteklih prosilcev, ali če ocenjujemo prosilca novejšega produkta, lahko uporabimo manjši vzorec podatkov ali vzorce podatkov podobnih produktov. Uporaba slednjega je slabša kot uporaba manjšega vzorca podatkov. V praksi je za uporabo modelov za ocenjevanje kreditnega tveganja prosilca potrebna katera koli lastnost posameznika, ki bi nam pomagala pri predvidevanjih. Večina uporabljenih lastnosti ima očitno povezave s tveganjem neizpolnitve obveznosti do banke. Nekatere kot so trajanje trenutne zaposlitve ali čas bivanja na trenutnem naslovu nam orišejo stabilnost prosilca, druge kot so imetje tekočega računa, imetje kreditnih kartic ali koliko časa je prosilec pri trenutni banki stranka, nam orišejo prosilčevo finančno stanje, tretje kot so status rezidentstva, zaposlitve ali zaposlitve partnerja, nam orišejo prosilčeve vire prihodka, nekatere kot so število otrok ali število oseb odvisnih od prosilčevega dohodka, pa nam

orišejo morebitne prosilčeve finančne odlive. V splošnem pa pri uporabi spremenljivk velja načelo, če pomaga pri predvidevanju, bi jo bilo pametno vključiti v model predvidevanja. Vendar pa vseh ni dovoljeno vključiti. Regulacije prepovedujejo vključitev spola, rase ali verskega prepričanja v ocenjevanje kreditnega tveganja. Chandler in Evert (v Crook et al., 2012, str. 5) sta v svoji študiji pokazala, da bi ob vključitvi spola v ocenjevanje kreditnega tveganja imelo več žensk kot moških odobren kredit. Nekatere posameznikove lastnosti kot so posameznikovo zdravstveno stanje ali število vozniških prekrškov niso vključene v kreditno ocenjevanje, ker so kulturno in socialno nesprejemljive. Nekateri posojilodajalci uporabljajo pri ocenjevanju kreditnega tveganja tudi uporabo zavarovanja dolga kreditne kartice. V splošnem pa je uporaba lastnosti pri ocenjevanju kreditnega tveganja subjektivnega značaja (Crook et al., 2002).

V osemdesetih letih prejšnjega stoletja so Nevin in Churchill (1979) ter Saunders (1985) v svojih delih zagovarjali ocenjevanje kreditnega tveganja, Capon (1982) pa mu je nasprotoval. Capon (1982) je v svojem delu trdil, da so modeli preveč subjektivni in odvisni od lastne presoje analitikov. Trdil je, da ocenjevanje kreditnega tveganja zmanjšuje potrebo po temelitejšem preiskovanju novih prosilcev, kar pa se v resnici ni dogajalo. Banke so tako ali tako zmanjševale preiskovanje novih prosilcev, saj jim je to omogočalo dajanje novih ponudb že obstoječim, ki so se jim zdele nesprejemljive. Uporaba kreditnih birojev je postajala pomembna predvsem pri zaznavanju prevar, saj so kreditni biroji preverjali informacije o prosilčevih preteklih kreditih. Zagovorniki ocenjevanja kreditnega tveganja pa so trdili, da ocenjevanje povečuje konsistentnost prosilcev, izboljšuje informacije dobljene iz računov in povečuje kvaliteto portfeljev. Nasprotniki uporabe ocenjevanja so predvsem napadali dejstvo, da ocenjevanje kreditnih tveganj ne pojasnjuje povezave med karakteristikami, ki so pomembne za ocenjevanje, in poznejšim izidom kreditne pogodbe. Trdili so, da obstaja kompleksna veriga vzajemno delujočih spremenljivk, ki povezujejo prvotne karakteristike in izid kreditne pogodbe. Nekateri modeli, kot je na primer grafično modeliranje, preiskujejo take verige spremenljivk. Prav tako je bila kritizirana uporaba statističnih metod zaradi pristranskosti uporabljenih vzorcev, ki ne vsebujejo že zavrnjenih prosilcev. Izpostavljeno je bilo tudi vprašanje o velikosti vzorca, ki ni natančno določena. Prav tako so kritizirali kolinearnost spremenljivk in njihovo nezveznost, ki jo grobo razvrščanje spremeni v zveznost (Crook et al., 2002).

1.2.1 Diskriminantna analiza

Diskriminantna analiza je poleg klasifikacijske metode edina metoda, ki se uporablja že od samih začetkov ocenjevanja kreditnega tveganja. Prednost diskriminantne analize je v dejstvu, da za namen ocenjevanja kreditnega tveganja uporabimo lastnosti vzorčnih cenilk, konstruiramo intervale zaupanja in testiramo želene hipoteze. Zaradi same statistične narave diskriminantne analize, je to metodo lažje interpretirati. Lahko govorimo o moči same diskriminacije in relativni pomembnosti različnih spremenljivk uporabljenih pri

oceni kreditnega tveganja. S pomočjo statistične analize lahko tako odstranimo nepomembne karakteristike iz modela. Originalno metodo diskrimantne analize je prvič uporabil Fisher (1936) za reševanje splošnih razvrstitvenih problemov. Po njegovi metodi pridemo do linearne ocene, ki temelji na Fisherjevi linearni diskriminantni funkciji. Linearna diskriminantno funkcijo lahko dobimo na tri načine: na podlagi odločitvene teorije, na podlagi ločevanja dveh podskupin in na podlagi formulacije linearne regresije. V magistrskem delu si bomo ogledali delovanje metode ločevanja dveh podskupin (Crook et al., 2002).

Kot smo že izpostavili, je cilj diskriminantne analize poiskati kombinacijo spremenljivk, ki najbolje ločujejo dve skupini glede na dane karakteristike. V primeru ocenjevanja kreditnega tveganja sta to podskupini, ki ju posojilodajalec razvrsti kot dobre in slabe, dane karakteristike pa so podrobnosti iz prošnje za odobritev kredita ter informacije kreditnih birojev. Označimo z $X = (X_1, X_2, \dots, X_p)$ množico karakteristik prosilcev kredita in z $Y = w_1X_1 + w_2X_2 + \cdots + w_pX_p$ neko linearno kombinacijo teh karakteristik. En možen pristop k ločevanju dobrih od slabih je iskanje povprečne vrednosti linearne kombinacije Y za dobre in slabe prosilce v vzorcu. Ideja je torej, da poiščemo pogojni matematični upanji E(Y|D) in E(Y|S) in izberemo uteži w_i tako, da se le-te seštejejo v 1. Ta način se izkaže za nekoliko naivnega, saj se lahko slabi zbirajo tudi okoli povprečja dobrih in obratno. Fisher je zato predlagal, da predpostavimo, da imajo dobri in slabi skupno vzorčno varianco. Dobra mera ločljivosti dobrih od slabih se potem lahko izrazi kot količnik med vzorčnima povprečjema dobrih in slabih ter skupno vzorčno varianco. Denimo, da je vzorčno povprečje dobrih μ_D, vzorčno povprečje slabih μ_S in S skupna vzorčna varianca. Nadalje predpostavimo, da je linearna kombinacija Y definirana kot zgoraj. Potem sledi, da je mera, ki loči dobre in slabe, označena z M, enaka

$$M = w' \frac{\mu_D - \mu_S}{(w'Sw)^{1/2}} \tag{1}$$

To drži, ker sta pogojna pričakovana vrednost dobrih in slabih enaki $w'\mu_D$ in $w'\mu_S$. Cilj je poiskati take uteži, ki bojo maksimirale razdaljo M. Zato torej razdaljo M odvajamo po parametru w in dani izraz izenačimo z 0. Reševanje tega optimizacijskega problema privede do rešitve za uteži w, ki je sorazmerna s faktorjem $S^{-1}(\mu_D - \mu_S)$. Ta rezultat drži ne glede na porazdelitev podatkov, saj v izrazu za razdaljo sredin uporabimo samo vzorčni povprečji in skupno varianco (Crook et al., 2002).

1.2.2 Logistična regresija

Pri logistični regresiji skušamo napovedati, kakšen bo prihodnji rezultat kategorične spremenljivke, natančneje binarne spremenljivke. Ta lahko zavzame samo vrednosti 0 ali

1. Pogojno matematično upanje te spremenljivke je izraženo kot $E(y|X) = P(y = 1|X)$. Poskušamo torej modelirati verjetnost nekega dogodka, torej pričakujemo rezultat med 0 in 1. Model logistične regresije povezuje predikate z verjetnostjo z naslednjo enačbo

$$p = P(y = 1|X) = \frac{exp(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k)}{1 + exp(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k)} \quad (2)$$

Verjetnost dogodka modeliramo z logistično porazdelitvijo. Količino $\frac{p}{1-p}$ imenujemo obeti (angl. *odds*) in povezuje verjetnost dobrega in slabega izida našega poizkusa.
Logistična regresija je najbolj pogosto uporabljen model za predvidevanje verjetnosti binarnega dogodka (Ledolter, 2013)

1.3 Metode podatkovnega rudarjenja

Družba je dandanes preplavljena z informacijami in podatki. Podatke lahko definiramo kot zapis dejstev, informacije pa kot nabor vzorcev ali pričakovanj, ki so osnova podatkom. Podatkovne baze vsebujejo ogromno informacij, ki so potencialno zelo pomembne, a še niso bile odkrite. Odkritje teh informacij nam omogočajo metode podatkovnega rudarjenja. Podatkovno rudarjenje bi lahko definirali kot črpanje neposrednih prej neznanih in potencialno uporabnih informacij iz podatkov. Osnovna ideja podatkovnega rudarjenja je narediti program, ki iz podatkovnih baz samodejno filtrira in išče vzorce ali regularnosti. Izrazitejši vzorci vodijo k natančnejšim predvidevanjem prihodnjih podatkov. Pri podatkovnem rudarjenju pa naletimo tudi na marsikateri problem. Veliko prepoznanih vzorcev je nezanimivih in banalnih, nekateri so lahko lažni, spet tretji pa odvisni od kakih naključij v določenem naboru podatkov. Prav tako so resnični podatki nepopolni, velikokrat so popačeni in manjkajoči. Vse kar lahko odkrijejo metode podatkovnega rudarjenja je lahko nenatančno. Vsako odkrito pravilo bo imelo izjeme, prav tako pa lahko naletimo na primere, za katere ne obstaja splošno pravilo. Zatorej morajo biti metode podatkovnega rudarjenja dovolj robustne, da so kos nepopolnim podatkom, in da izločijo vzorce in regularnosti, ki so nenatančni a uporabni (Frank, Hall, & Witten, 2011).

Strojno učenje predstavlja tehnično osnovo podatkovnega rudarjenja. Uporablja se za pridobivanje informacij iz podatkov, ki jih vsebujejo podatkovne baze. Osnovni proces strojnega učenja lahko opišemo kot zajem podatkov ter njihovo procesiranje in končno sklepanje o osnovnih strukturah podatkov. Strojno učenje lahko interpretiramo tudi kot pridobitev strukturnih opisov iz primerov. Najdeni opisi se lahko uporabljajo za napovedovanje, razlaganje in razumevanje določenih problemov (Frank et al., 2011).

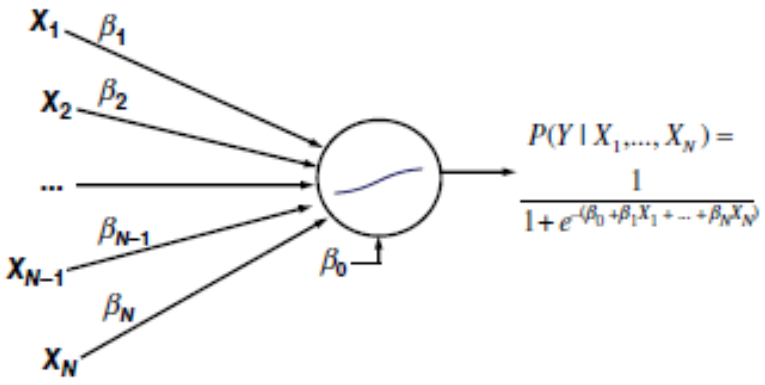
V grobem bi lahko metode podatkovnega rudarjenja razdelili v dve skupini: metode, ki se ukvarjajo z napovedovanjem in metode, ki se ukvarjajo z opisom struktur, na podlagi katerih lahko razvrstimo posamezne primere. Metode, ki se ukvarjajo z napovedovanjem, poskušajo napovedati, kaj se bo zgodilo v prihodnjih situacijah iz zgodovinskih podatkov,

najpogosteje z ugibanjem razvrščanja novih primerov. Metode, ki skušajo z učenjem priti do opisov struktur ponujajo poleg predvidevanja še razlago in razumevanje. Iskanje spoznanj pridobljenih s strani uporabnikov je zelo pomembno v praktični uporabi metod podatkovnega rudarjenja, kar je tudi glavna prednost uporabe teh metod. V nadaljevanju poglavja sledi opis nekaterih metod podatkovnega rudarjenja (Frank et al., 2011).

1.3.1 Nevronske mreže

Nevroni so sestavni del možganov, ki so odgovorni za hranjenje podatkov in prenašanje informacij. Kot že ime metode pove, je ta navdahnjena po delovanju nevronov v človeških možganih. Osnovni gradniki metode so t.i. nevroni, ki imajo več različno uteženih vhodov, medsebojne povezave ter en izhod. Najenostavnejše nevronske mreže imajo le en sloj, kompleksnejše pa jih lahko imajo več. Po medsebojnih povezavah nevroni pošiljajo drug drugemu signale. Ko je vsota vhodnih signalov dovolj velika, pride do vžiga nevrona in na izhodu se pojavi signal. Uteži vhodov posameznih nevronov, njihove medsebojne povezave ter prag oddajanja signala, se oblikujejo z učenjem. Parametri nevronske mreže se torej spreminjajo toliko časa, dokler ni ta zmožna optimalno rešiti problem. Metoda nevronske mreže je torej matematični model delovanja človeških možganov. Po drugi strani pa lahko na nevronske mreže gledamo kot na posplošitev obstoječih statističnih metod. Ta pogled si oglejmo na primeru logistične regresije.

Slika 1: Prikaz logistične regresije s pomočjo nevronskih mrež

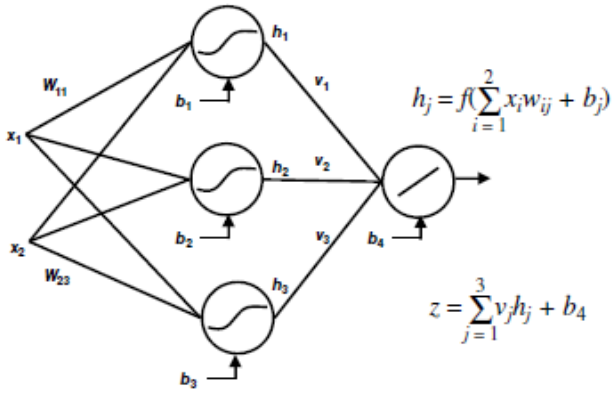


Vir: B.Baesens, Analytics in a Big Data World: The Essential Guide to Data Science and its Applicatios,, 2014, stran 48.

Na Sliki 1 element na sredini imenujemo nevron ali procesni element. Nevron ima dve nalogi, prva je procesiranje vhodnih podatkov in množenje teh z ustreznimi utežmi, kjer je

vključen tudi konstanten člen β_0 (v primeru nevronskih mrež ga imenujemo člen pristranskosti). Nevron nadalje uporabi nelinearno transformacijo, ki spremenljivke transformira v želeno obliko, v našem primeru v enačbo logistične regresije. Logistična regresija je torej nevronska mreža z enim nevronom. Podobno kot logistično lahko okarakteriziramo tudi linearno regresijo, ki je nevronska mreža z enim nevronom in linearno transformacijo (ali identiteto). Enostaven primer regresije lahko razširimo na večslojno perceptronsko nevronsko mrežo (v nadaljevanju MLP nevronska mreža) z dodajanjem nevronov (Baesens, 2014).

Slika 2: MLP nevronska mreža



Vir: B.Baesens, Analytics in a Big Data World: The Essential Guide to Data Science and its Applications, 2014, stran 49.

Slika 2 nam prikazuje MLP nevronsko mrežo z eno vhodno plastjo, eno skrito plastjo in eno izhodno plastjo. Skrita plast ima nelinearno transformacijo f, izhodna plast pa ima linearno transformacijo. Najpogosteje uporabljene transformacije pri nevronskih mrežah so (Baesens, 2014):

- Logistična, kjer vhodno spremenljivko transformiramo s funkcijo $f(z) = \frac{1}{1+e^{-z}}$, ki variira med 0 in 1,
- hiperbolična tangentna, kjer vhodno spremenljivko transformiramo s funkcijo $f(z) = \frac{e^{z}-e^{-z}}{e^{z}+e^{-z}}$, ki variira med -1 in 1,
- linearna, kjer vhodno spremenljivko transformiramo s funkcijo $f(z) = z$, ki variira med -∞ in ∞.

Optimizacija parametrov pri metodi nevronskih mrež je kompleksnejša od optimizacije parametrov enostavnih statističnih modelov (na primer linearne regresije). Za ocenjevanje uteži, ki so na povezavah uporabljamo navadno iterativne postopke. Na koncu algoritem optimizira še funkcijo, ki meri variabilnost izhodnih parametrov. Postopek optimizacije parametrov običajno začnemo z naključno določenimi utežmi, ki jih nato iterativno prilagajamo vzorcem, ki so dobljeni iz podatkov s pomočjo optimizacijskih algoritmov. Najpogosteje uporabljene metode optimizacije so metoda backpropagation, metoda konjugiranih gradientov in Levenberg-Marquardtova metoda. Glavni problem pri vseh metodah je določanje začetnih uteži. Ciljna funkcija optimizacijskega problema ni nujno konveksna in ima lahko več lokalnih minimumov. Če so začetne uteži izbrane neprimerno, lahko dobimo kot globalno rešitev problema v bistvu rešitev, ki je lokalni ekstrem.

Postopek optimizacije se po izboru začetnih uteži in iteraciji zaustavi ko je napaka dovolj majhna, ko se uteži ne spreminjajo več očitno ali ko smo prekoračili določeno število vnaprej predpisanih korakov iteracij. Če uporabljamo nevronsko mrežo z več kot enim nevronom, moramo določiti število skritih nevronov. Najenostavneje lahko ta postopek opišemo z naslednjimi koraki (Baesens, 2014):

- Podatke razdelimo na tri podmnožice: podmnožica namenjena treningu, podmnožica namenjena testiranju in podmnožica namenjena validaciji.
- Spreminjamo število skritih nevronov od 1 do 10 s korakom 1.
- Podmnožica podvrženo treningu izpostavimo učenju na nevronski mreži in izmerimo izvedbo na podmnožici namenjeni validaciji (tu lahko uporabimo več nevronskih mrež).
- Izberemo optimalno število skritih nevronov, ki ima najoptimalnejšo izvedbo.
- Izmerimo izvedbo na podmnožici namenjeni testiranju.

Nevronske mreže so navkljub njihovi splošnosti modeliranja pogosto opisane kot črna skrinjica, kjer so vhodni in izhodni parametri s kompleksnimi matematičnimi modeli. Za ocenjevanje kreditnih tveganj je pomembna tudi interpretacija matematičnega modela, zato je potrebno biti pri uporabi nevronskih mrež previden, saj nas ne zanimajo zgolj izhodni parametri, pač pa sta pomembna tudi vpogled in razumevanje osnovnih vzorcev. Za interpretacijo nevronskih mrež običajno uporabimo ekstrakcijo pravil ali dvostopenjsko modeliranje. Namen ekstrakcije pravil je posnemati obnašanje nevronskih mrež. Ena možnost je dekompozicija celotne mreže na posamezne dele glede na vrednosti uteži. To najpogosteje lahko storimo v naslednjih petih korakih:

- Treniramo nevronsko mrežo in zmanjšamo število povezav na najmanjše možno.
- Razvrstimo aktivne vrednosti skritih enot z uporabo grozdenja.
- Izluščimo pravila, ki opisujejo izhodne parametre mreže glede na razvrščene aktivne vrednosti skritih enot.
- Izluščimo pravila, ki opisujejo razvrščene aktivne vrednosti skritih enot glede na vhodne parametre mreže.
- Za direkten opis razmerja med vhodnimi in izhodnimi parametri združimo pravili pod točkama 3 in 4.

Druga možnost ekstrakcije pravil je uporaba tehnik pedagoške ekstrakcije pravil. Pri tem postopku podmnožico podatkov, ki je namenjena treningu umetno povečamo z dodajanjem opazovanj, ki so umetno ustvarjena (Baesens, 2014).

Za lažjo interpretacijo nevronskih mrež lahko uporabimo tudi dvostopenjsko modeliranje. Osnovna ideja je, da ocenjujemo model, ki ga je lahko interpretirati (na primer linearna regresija), nato pa v drugi stopnji modeliranja uporabimo nevronske mreže, ki

predvidevajo napake enostavnega modela z uporabo enake množice prediktorjev. Obe stopnji modeliranja nato seštejemo,kot je ilustrirano na naslednjem primeru:

$$\begin{aligned} Odvisna\ spremenljivka & \\ &= linearna\ regresija\ (Prediktorji\ X_1, \dots, X_n) \\ &+ nevronska\ mreža\ (Prediktorji\ X_1, \dots, X_n) \end{aligned} \tag{3}$$

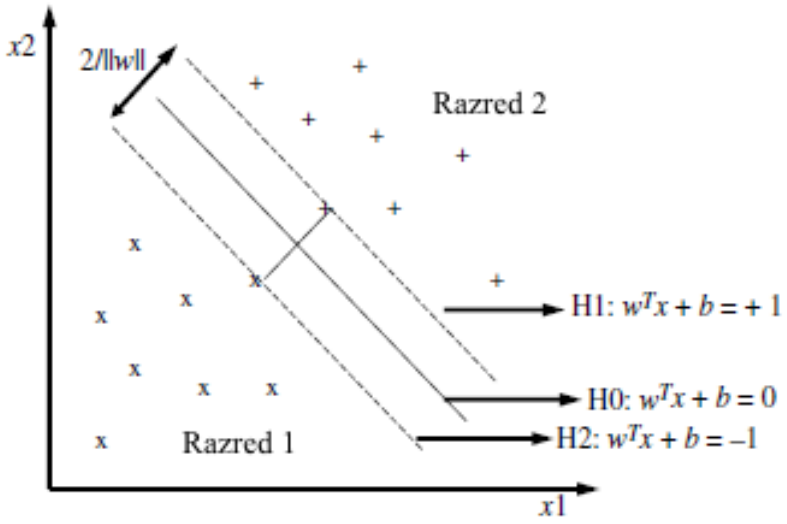
Na ta način sam model lažje interpretiramo in merimo njegovo izvedbo (Baesens, 2014).

1.3.2 Metoda podpornih vektorjev

Metoda nevronskih mrež ima dve ključni pomanjkljivosti: ciljna funkcija ima lahko več lokalnih ekstremov in težko je določiti število skritih nevronov. Metoda podpornih vektorjev teh problemov nima. Osnovna naloga metode podpornih vektorjev je ločiti podana razreda glede na dane karakteristike (v primeru kreditnih tveganj ločiti dobre od slabih posojilojemalcev glede na dan karakteristike). To metoda podpornih vektorjev opravi s postavitvijo optimalne hiperravnine. Optimalna hiperravnina je ravnina, ki je hkrati enako in najbolj oddaljena od najbližjih primerov obeh razredov. Najbližje primere obeh razredov imenujemo podporni vektorji, njihovi razdalji do hiperravnine pa pravimo rob. Optimalna hiperravnina je tista, ki ima optimalen rob. Ker pa med originalnimi razredi velikokrat ne obstaja linearna delitev, je potrebno najprej podatke nelinearno transformirati. Metoda je zelo primerna za učenje na velikih množicah z velikim številom nepomembnih karakteristik. Slaba stran metode je v njeni časovni zahtevnosti in interpretaciji odločitev.

Oglejmo si delovanje metode podpornih vektorjev na enostavnem primeru. Za ilustracijo naj nam služi Slika 3.

Slika 3: Razdelitev po metodi podpornih vektorjev v primeru popolne linearne ločljivosti



Vir: B.Baesens, Analytics in a Big Data World: The Essential Guide to Data Science and its Applications, 2014, stran 59.

Slika 3 nam prikazuje tri hiperravnine, dve robni H1 in H2 ter eno vmesno H0, ki služi kot razdelitvena meja. Razdalja med ravninama H1 in H0 je enaka $\frac{|b-1|}{\|w\|}$, kjer je $\|w\| = \sqrt{w_1^2 + w_2^2}$ običajna Evklidska norma. Podobno je razdalja med ravninama H2 in H0 enaka $\frac{|b+1|}{\|w\|}$. Razdalja med robnima ravninama H1 in H2 je torej $\frac{2}{\|w\|}$. Cilj metode podpornih vektorjev je maksimirati razdaljo med tema robnima ravninama. Problem lahko prevedemo na minimizacijo norme $\|w\|$, kar se nadalje lahko prevede v minimizacijo izraza $\frac{1}{2}\sum_{i=1}^{n} w_i^2$. V primeru popolne linearne ločljivosti z metodo podpornih vektorjev postopamo takole. Denimo,da imamo množico podatkov za urjenje modela (angl. *training set*) $\{x_k, y_k\}_{k=1}^{n}$, kjer je $x_k \in \mathbb{R}^N$ in $y_k \in \{-1, +1\}$. Dobri predstavniki množice podatkov za urjenje (razred +1) naj bi ležal nad ravnino H1, slabi predstavniki (razred -1) pa pod ravnino H2. Ta razdelitev nas pripelje do reševanja naslednjega optimizacijskega problema

$$Min \frac{1}{2}\sum_{i=1}^{N} w_i^2 \text{ pri pogojih} \qquad (4)$$
$$y_k(w'x_k + b) \geq 1 \text{ za } k = 1, \dots n$$

Ta problem kvadratične optimizacije rešimo s pomočjo metode Lagrangevih množiteljev. Ciljna funkcija v tem primeru je kvadratna brez lokalnega minimuma in z enim globalnim maksimumom. Točke iz množice podatkov za urjenje modela, ki ležijo na hiperravninah H1 in H2 se imenujejo **podporni vektorji**. Za nova opazovanja je potrebno preveriti ali

ležijo nad razdelitveno hiperravnino H0, kar jih uvršča kot dobra (+1) ali pod razdelitveno hiperravnino, kar jih uvršča kot slaba (-1) (Baesens, 2014).

Do sedaj smo predpostavljali, da je med dobrimi in slabimi vedno možna popolna linearna ločitev. V primeru, ko se razreda dobrih in slabih prekrivata lahko razširimo razvrstitev glede na metodo podpornih vektorjev, kar nas privede do naslednjega optimizacijskega problema

$$\begin{gathered} Min \frac{1}{2}\sum_{i=1}^{N} w_i^2 + C\sum_{i=1}^{n} e_i \text{ pri pogojih} \\ y_k(w'x_k + b) \geq 1 - e_k \text{ za } k = 1, \dots n \\ e_k \geq 0. \end{gathered} \tag{5}$$

Parameter e_k imenujemo **napaka razvrstitve**, parameter C v ciljni funkciji pa nam omogoča ustvariti ravnotežje med maksimiziranjem odmika hiperravnin in minimizacijo napake v podatkih. Visoka (nizka) vrednost parametra C nakazuje visoko (nizko) tveganje za prenasičenost (angl. *overfitting*) modela. Zgornji kvadratični optimizacijski problem ponovno rešimo s pomočjo metode Lagrangevih množiteljev (Baesens, 2014).

V primeru, ko delitev podpornih vektorjev ni linearna, vhodne podatke najprej transformiramo z neko preslikavo $\varphi(x)$, ki naredi delitev linearno. Optimizacijski problem se prevede v

$$\begin{gathered} Min \frac{1}{2}\sum_{i=1}^{N} w_i^2 + C\sum_{i=1}^{n} e_i \text{ pri pogojih} \\ y_k(w'\varphi(x_k) + b) \geq 1 - e_k \text{ za } k = 1, \dots n \\ e_k \geq 0. \end{gathered} \tag{6}$$

Ta optimizacijski problem ponovno rešimo z metodo Lagrangevih množiteljev (Baesens, 2014).

Izkaže se, da je zgornji optimizacijski problem lažje rešiti, če vpeljemo t.i. jedrno funkcijo $K(x_k, x_l) = \varphi(x_k)'\varphi(x_l)$. Z vpeljavjo jedrne funkcije lahko SVM klasifikator y_k izrazimo kot $y(x) = sign(\sum_{k=1}^{n} \lambda_k y_k K(x, x_k) + b)$, kjer λ_k predstavljajo Lagrangeeve množitelje, ki izhajajo iz optimizacijskega problema (6). Lagrangeevi množitelji podpornih vektorjev so neničelni, preostali so enaki nič. Za jedrno transformacijo lahko uporabimo več možnih funkcij, spodaj so naštete nekatere najbolj osnovne (Baesens, 2014):

- jedro $K(x, x_k) = x_k'x$ imenujemo tudi linearno jedro,
- jedro $K(x, x_k) = (1 + x_k'x)^d$ imenujemo tudi polinomsko jedro ter
- jedro $K(x, x_k) = e^{-\gamma\|x - x_k\|^2}$ imenujemo tudi radialna bazna fukncija (RBF).

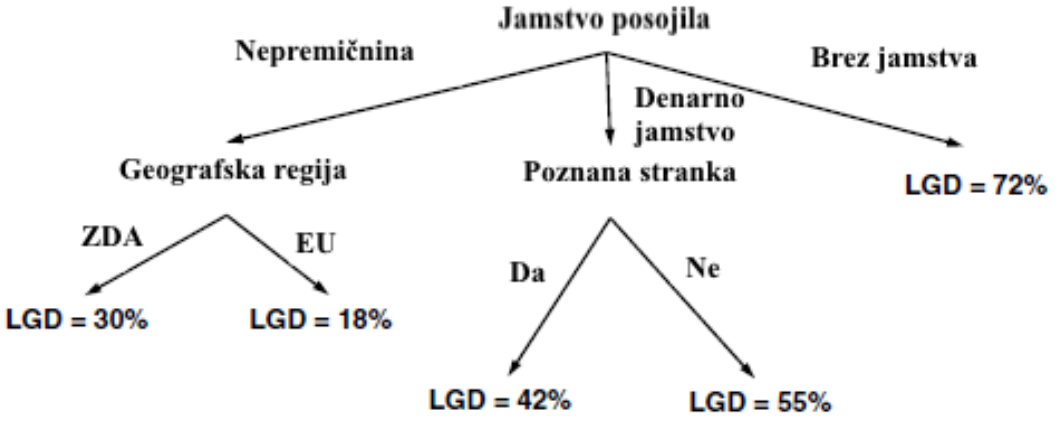
Empirične raziskave so pokazale, da se v povprečju najbolje obnese RBF jedro, vendar ta po drugi strani zahteva še iskanje optimalnega parametra γ. Parameter γ definira vpliv posameznega podpornega vektorja. Visoke vrednosti parametra nakazujejo finejšo delitev podatkov in so pokazatelj prenasičenosti modela. Nizke vrednosti parametra pa po drugi strani nakazujejo grobo delitev podatkov. V tem primeru verjetno ne opišemo kompleksnosti modela dovolj natančno, saj vsak podporni vektor vpliva na celotno množico podatkov(Burke & Felfernig, 2008).

Tako kot nevronske mreže ima tudi metoda podpornih vektorjev univerzalno lastnost aproksimacije, kar pomeni, da lahko poljubno zvezno funkcijo aproksimirajo poljubno natančno. Za razliko od nevronskih mrež, pri metodi podpornih vektorjev ni potrebno določiti števila skritih nevronov, prav tako pa je metoda karakterizirana s konveksno optimizacijo. Po drugi strani je uporaba metode podpornih vektorjev lahko zelo kompleksna pri problemih, ki zahtevajo interpretacijo modelov. Metodo podpornih vektorjev se lahko predstavi kot nevronsko mrežo, torej nam pri interpretaciji metode lahko pomaga eno od pravil, ki smo jih uporabili za interpretacijo nevronskih mrež (Baesens, 2014).

1.3.3 Odločitvena drevesa

Odločitvena drevesa sodijo v razred rekurzivnih razdelitvenih algoritmov. Ime izhaja iz drevesne strukture, ki nam predstavlja možne vzorce osnovnih podatkov. Drevo sestavljajo koren, notranja vozlišča in listi. V vsakem notranjem vozlišču se opravi preizkus, ki preveri eno ali več neodvisnih spremenljivk. Pot iz korena do listov nam predstavlja zaporedje delnih odločitev, ki jih sprejemamo pri razvrščanju vzorcev. Listi dreves predstavljajo rezultate. Novi primeri se klasificirajo po naslednjem postopku: začnemo pri korenu, potujmo navzdol v skladu z rezultati testov in na koncu dobimo odgovor v listu.
Na Sliki 4 je prikazan primer regresijskega odločitvenega drevesa.

Slika 4: Primer odločitvenega drevesa



Vir: B.Baesens, Analytics in a Big Data World: The Essential Guide to Data Science and its Applications, 2014, stran 47.

Odločitveno drevo sestavljajo koren, notranja vozlišča in listi. V primeru na Sliki 4 je koren spremenljivka Jamstvo posojila, listi so vrednosti pri spremenljivki LGD, notranja vozlišča pa so Nepremičnina, Denarno jamstvo, itd. V vsakem vozlišču opravimo preizkus, ki preveri vrednost ene ali več spremenljivk. Če ima preizkus dva izida, govorimo o **binarnih odločitvenih drevesih**. Vsak list vsebuje preizkus, ki odvisni spremenljivki (razredu) priredi kategorično ali številsko vrednost. Če je vrednost kategorična, govorimo o **razvrščevalnih odločitvenih drevesih**, če pa je vrednost številska, govorimo o **regresijskih odločitvenih drevesih**. Ko gradimo odločitveno drevo, je potrebno odgovoriti na tri ključna vprašanja. Kako in glede na kakšne vrednosti razdeliti spremenljivke (npr. starost < 40)? Kdaj ustaviti rast drevesa? Ali bodo na listih dobri ali slabi? Te tri naloge lahko imenujemo tudi razdelitvena odločitev, zaustavitveni kriterij in odločitev o nalogi. Odgovor na zadnje vprašanje je običajno najlažji. Najlažje je preveriti, ali so večinski rezultati v listih dobri ali slabi in se odločiti na podlagi te primerjave. Vprašanje, kako razdeliti podatke je odvisno od same kvalitete podatkov. Najpopularnejše mere, ki merijo kvaliteto podatkov so:

- Entropija, ki se izračuna kot $-d_G \log_2 d_G - d_B \log_2 d_B$, kjer je d_G delež dobrih v osnovnih podatkih in d_B delež slabih v osnovnih podatkih,
- Gini, ki se izračuna kot $2d_G d_B$,
- analiza hi-kvadrat.

Entropija in Gini imata najnižjo vrednost takrat, ko so v osnovnih podatkih vsi dobri ali slabi, in najvišjo vrednost takrat, ko je delež dobrih in slabih v osnovnih podatkih enak. Glede na kvaliteto podatkov, dobimo več možnih razdelitev. Na koncu se odločimo za

tisto, s katero največ pridobimo. Ostane nam še odgovor na vprašanje, kdaj ustaviti rast drevesa. Če je drevo preveč razcepljeno, dobimo veliko listov, ki vsebujejo zelo malo opazovanj. Ta lahko vsebujejo veliko šuma, kar potencialno vodi do prenasičenosti modela. Da se izognemo temu problemu razdelimo podatke na množico, ki je namenjena urjenju in množico, ki je namenjena preverjanju. Množica, ki je namenjena urjenju, bo uporabljena za odločitev o razdelitvi, množico, ki je namenjena preverjanju, pa uporabljamo za kontrolo napake napačnega razvrščanja (angl. *misclassification error*). Napaka na vzorcu množice, ki je namenjena urjenju se z večanjem števila vozlišč manjša, napaka na vzorcu množice, ki je namenjena preverjanju, pa se na začetku manjša, nato doseže minimum in se ponovno prične večati z večanjem števila vozlišč. Razlog je v čedalje finejši delitvi listov, kar privede do tega, da odločitveno drevo večkrat preveri podobne informacije. Optimalno je ustaviti rast drevesa v točki, kjer doseže napake množice, ki je namenjena preverjanju svoj minimum (Baesens, 2014).

V primeru, ko se odločamo o razdelitvi pri regresijskem drevesu, uporabimo kot mero kvalitete podatkov povprečno kvadratno napako (angl. *mean squared error*), ki se izračuna kot $MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \bar{Y})^2$, kjer je n število opazovanj v listu, Y_i je vrednost i-tega opazovanja in $\bar{Y}$ predstavlja povprečno vrednost vseh opazovanj v listu. Kot zaustavitveni kriterij v primeru regresijskih odločitvenih dreves lahko uporabimo mero R^2, povprečno kvadratno napako ali povprečno absolutno deviacijo. Odločitev o nalogi lahko naredimo glede na povprečno vrednost posameznega lista. Za vsak list lahko prav tako izračunamo interval zaupanja (Baesens, 2014).

Največja prednost odločitvenih dreves je, da jih je lahko interpretirati. Prav zaradi tega se jih lahko uporablja na številnih področjih (Baesens, 2014).

1.3.4 K-najbližjih sosedov

Metoda k-najbližjih sosedov je zelo uporabna takrat, ko želimo narediti razvrstitev določenih objektov (v primeru kreditnih tveganj bi radi razvrstili tiste, ki so se prijavili za posojilo na bolj in manj tvegane). Deluje tako, da nov objekt glede na želene rezultate postavi k tistim v množici, ki je namenjena urjenju, ki so mu najbolj podobni. Poenostavljeno povedano, ali si med dobrimi ali slabimi prosilci posojila odločajo tisti, ki so že razvrščeni, in sicer tisti, ki si jim bolj podoben. Učenja pri tej metodi skorajda ni, zato tej vrsti učenja rečemo tudi **leno učenje**. Metoda deluje po naslednjem postopku:

- Iz množice namenjene treningu vzamemo objekt, ki ima d lastnosti, ni pa še razvrščen v katero koli od skupin.
- Izračunamo razdaljo med novim objektom in vsakim objektom iz množice namenjene treningu, ki je že razvrščen.

- Preverimo k najbližjih razdalj v množici namenjeni treningu, preverimo njihovo razvrstitev in določimo najpogostejšo.

Najpogostejša razdelitev nato postane predvidena razvrstitev novega objekta. Če je pri klasifikaciji novega objekta izenačenih več najpogostejših možnih razdelitev, nov objekt naključno dodelimo eni od teh. V primeru, da je najbližji sosed en sam, nov objekt dodamo tej razvrstitveni skupini. Sosede vzamemo iz množice, ki je že pravilno razvrščena, kar načeloma razumemo kot množico, namenjeno urjenju, čeprav take množice metoda neposredno ne potrebuje. Faza urjenja pri tej metodi je namenjena shranjevanju in označevanju razredov posameznih vzorcev iz množice namenjene urjenju. V fazi razvrstitve najprej določimo konstanto k. Nov objekt je nato razvrščen med obstoječe tako, da ga označimo z oznako, ki je najpogostejša med k vzorci urjenja, ki so najbližje temu novemu objektu (Ledolter, 2013).

V primeru, ko razvrščamo številske lastnosti, razdaljo med njimi merimo z običajno Evklidsko normo. Če imamo le eno lastnost objektov, je njuna razdalja absolutna vrednost razlike. Ko razvrščamo kategorične lastnosti (spol, starost), uporabimo Hammingovo razdaljo. Hammingova razdalja za dva enako dolga niza znakov ustvari zaporedje enakih in različnih znakov na istoležnih mestih v nizih in nato prešteje število pojavitev različnih znakov. Razdalja je majhna, če je število različnih pojavitev majhno (Ledolter, 2013).

Slaba lastnost te enostavne a uporabne metode je ta, da razredi, ki imajo pogosteje uporabljene lastnosti dominirajo razvrstitev novega objekta. Ker se njihove lastnosti pojavijo večkrat, se pogosteje pojavijo med k–najbližjimi sosedi, ko se razvrsti nov objekt. Metoda je prav tako občutljiva na lokalno strukturo podatkov, saj nanjo precej vpliva prisotnost šumov ali nepomembne lastnosti. Kako najbolje izbrati število k, je odvisno od podatkov. Večje vrednosti parametra k po eni strani zmanjšajo šum razvrstitve novih objektov, po drugi strani pa se z večanjem tega parametra zabrišejo meje med sosedi. Če ima problem razvrstitve binarni izid (kar v primeru iskanja verjetnosti neizpolnjevanja obveznosti drži) izberemo k kot liho število, da se izognemo neodločenim izidom pri razvrščanju (Ledolter, 2013).

1.3.5 Genetski algoritmi

Genetski algoritem bi lahko opisali kot postopek sistematičnega iskanja skozi populacijo potencialnih rešitev določenega problema, kjer ima rešitev, ki pride bližje splošni rešitvi problema, večjo možnost ohranitve v množici kandidatov za rešitev od ostalih. Ideja samih genetskih algoritmov izhaja iz Darwinove teorije o evoluciji naravne selekcije (Darwin, 1859). Evolucijo lahko preprosto opišemo s tremi parametri: mutacijo, selekcijo in križanjem. Mutacija dednega zapisa je neusmerjen proces, ki je namenjen iskanju

alternativ. Naloga mutacije je preseči lokalni minimum. Križanje dednega zapisa ima vlogo iskanja rešitve med mutacijo in selekcijo. Selekcija določa, v katero smer se bo spreminjal dedni zapis. Brez motenj je selekcija deterministična, vendar ji to preprečujejo mutacije, ki vnašajo naključje v selekcijo. Genetski algoritmi nam omogočajo iskanje rešitev v celotni populaciji, zato lahko najdemo več potencialnih rešitev hkrati (Crook et al., 2002).

Recimo, da je funkcija $f(x_i) = a_1 x_{i1}^{b_1} + a_2 x_{12}^{b_2} + \cdots + a_p x_{ip}^{b_p} + c$, kjer so $x_{i1,} x_{i2}, \dots x_{ip}$ karakteristike i-tega prosilca kredita, enačba, ki ocenjuje kreditno tveganje prosilca i. Oceniti moramo parametre $a_1, a_2, \dots, a_p$ in $b_1, b_2, \dots, b_p$ ter razvrstiti i-tega prosilca glede na vrednost funkcije $f(x_i)$. Če je vrednost pozitivna, ga razvrstimo kot dobrega, če pa je negativna, ga razvrstimo kot slabega prosilca. Pri uporabi genetskega algoritma postopamo po naslednjih korakih. Najprej določimo populacijo možnih kandidatov za vse parametre a_j, b_j in c. Vsakemu kandidatu nato določimo binarno vrednost 0 ali 1, kar imenujemo tudi niz kromosomov. Množica rešitev računskega problema je binarna množica za vsakega od parametrov $a_1, \dots, a_p$, $b_1, \dots, b_p$ in c. Niz kromosomov vsebuje določene lastnosti ali gene. Vsak gen zavzame neko vrednost ali alel. Rešitev enačbe ocenjevanja kreditnega tveganja sestoji iz množic v vrstico urejenih genov, kjer vsak gen zavzame vrednost 0 ali 1, kjer se vsaka množica nanaša na kombinacijo lastnosti in pripadajočih parametrov. Celo vrstico imenujemo kromosom ali niz. V naslednjem koraku določimo število rešitev za vključitev v vmesno populacijo. Ta števila rešitev so lahko določena naključno. Da določimo člane vmesne populacije, izračunamo izid vsake rešitve v začetni populaciji. Ta izid pogosto imenujemo tudi fitnes. Pri ocenjevanju kreditnega tveganja, lahko fitnes določimo kot delež pravilno razvrščenih primerov, večkrat pa uporabimo normalizirano vrednost izida, ki jo izračunamo kot $p_j = \frac{f_j}{\sum_{j=1}^{n} f_j}$, ki nam pove delež izida j-te množice rešitev v množici celotnih izidov. Vmesna populacija je določena z naključnim izborom nizov prvotne populacije, kjer je p_j verjetnost izbora j-tega niza. S tem korakom ustvarimo vmesno populacijo, ne ustvarjamo pa novih kromosomov, ki jih ustvarimo v tretjem koraku. Te ustvarimo tako, da določimo število parov rešitev iz vmesne populacije, na katerih nato uporabimo genetske operacije. Genetska operacija je postopek spreminjanja vrednosti alelov določenega niza. Uporabimo lahko t.i. križanje ali mutacijo. Križanje je genetska operacija, kjer prvemu nizu zamenjamo prvih ali zadnjih n znakov z istoležnimi znaki drugega niza. Število znakov, ki jih zamenjamo, je določeno naključno. Demonstrirajmo križanje na primeru. Prvi niz naj bo 0110110, drugi pa 1101110. Denimo, da križanje izvajamo na zadnjih štirih znakih niza. Dobimo torej naslednja dva niza

011|0110 ->0111110
110|1110 ->1100110

Prvotna niza imenujemo starša, novonastala pa otroka. Otroka nato v vmesni populaciji zamenjamo s staršema. Mutacijo pa izvajamo tako, da v nizu naključno določimo znak in mu spremenimo njegovo vrednost (iz 0 v 1 in obratno). Ko na kromosomih vmesne populacije izvedemo križanja in mutacije dobimo novo populacijo. Drugi in tretji korak algoritma nato ponavljamo poljubno krat. Parametri, ki jih določi analitik so število kandidatov rešitev v populaciji, verjetnosti križanja in mutacij in število generacij (Crook et al., 2002).

2 PRIPRAVA PODATKOV ZA ANALIZO

Podatki so za namene analize problema ključnega pomena. Pravilo, ki naj bi se ga držali je nekako več kot je podatkov, bolje je. V realnosti, pa so podatki lahko zelo slabi zaradi morebitne nekonsistentnosti, nepopolnosti, podvajanja ali problemov, ki nastanejo pri združevanju podatkov. Zaradi tega je pred samo analizo podatkov potrebno opraviti proces filtracije podatkov (angl. *Data preprocessing*). Seveda nam sama filtracija ne pomaga, če imamo že v osnovi slabe podatke. Tu namreč velja načelo, da nam slabi podatki ne pomagajo pri uporabi analitičnih modelov (angl. *Garbage in garbage out*, v nadaljevanju GIGO).

Pod pojmom filtracija podatkov razumemo procese, ki jih opravimo za izboljšavo vhodnih podatkov za namene empirične analize problema. Ti procesi so lahko:

- Vzorčenje,
- določanje tipa podatkov,
- vizualna in statistična raziskava podatkov,
- manjkajoče vrednost,
- ugotavljanje in reševanje napačnih vrednosti,
- standardizacija podatkov,
- kategorizacija ,
- določanje spremenljivk.

V nadaljevanju bom podal kratek opis vsakega od zgoraj naštetih procesov (Baesens, 2014).

2.1 Vzorčenje

Namen vzorčenja je vzeti neko podmnožico zgodovinskih podatkov in jih uporabiti pri analizi. Pri tem procesu je potrebno paziti na to, da je vzorec reprezentativen. Prav tako se potrebno izogniti morebitnim pristranskostim kolikor je le možno, kar je pri zbiranju podatkov za ocenjevanje kreditnega tveganja zelo težko doseči. Denimo, da želimo oceniti tveganja bodočih prosilcev hipotekarnih posojil. Osnovni vzorec tu sestoji iz vseh, ki so se pri banki prijavili za kredit (angl. *Through-the-door population* v nadaljevanju *TTD*). Da bi

zgradili analitičen model za ocenjevanje kreditnega tveganja torej potrebujemo zgodovinske podatke vseh, ki so se pri banki prijavili za kredit. Banka pa je lahko v preteklosti že dajala posojila, ki ni bil odobren vsem. Iz tega sledi, da ima njihova TTD populacija dve podmnožici: stranke, ki so jim odobrili kredit in stranke, ki jim kredita niso odobrili. Ciljnih vrednosti za slednje ne poznamo, saj kredita niso dobili odobrenega. Ko torej zbiramo vzorec, lahko uporabimo le tiste, ki jim je banka odobrila kredit, kar pomeni, da imamo pristranskost v vzorčenju. Prav tako moramo pri zbiranju podatkov za ocenjevanje kreditnega tveganja biti pozorni na tiste, ki jim je bil kredit sicer odobren, a ga nato niso vzeli. Da bi imeli reprezentativen vzorec, moramo vanj vključiti tudi te (Baesens, 2014).

2.2 Določanje tipa podatkov

Pri določanju tipa podatkov ločimo dve možnosti. Podatki so lahko kvantitativni (numerični podatki) ali kvalitativni (kategorični podatki). Kvantitativni podatki so na primer teža v kilogramih, čas v sekundah, število sodelujočih v anketi itd. Kategorični podatki pa so na primer spol in odgovori na vprašanja z da ali ne. Kvantitativni podatki so lahko zvezni ali diskretni. Diskretni podatki vsebujejo končne vrednosti, ki jih lahko štejemo, zvezni pa tvorijo kontinuum iz neskončno korakov (npr. čas). Kategorične podatke lahko razdelimo na nominalne, ordinalne in binarne. Nominalni podatki so podatki, ki lahko zavzamejo končne vrednosti, kjer vrstni red ni pomemben. Primeri nominalnih kategoričnih podatkov so zakonski stan, poklic, namen posojila. Ordinalni podatki so podatki, ki lahko zavzamejo končne vrednoti, kjer je pomemben tudi vrstni red. Primer ordinalnega kategoričnega podatka je kreditna ocena. Binarni podatki so podatki, ki lahko zavzamejo samo dve vrednosti, običajno označeni z 0 in 1. Primer binarnega kategoričnega podatka je spol (Baesens, 2014).

2.3 Vizualna in statistična raziskava podatkov

Na vizualno raziskavo podatkov bi lahko gledali tudi kot na spoznavanje podatkov na neformalen način. Najpogosteje tu uporabljamo različne vrste grafov in histogramov. Pri statistični raziskavi podatkov poiščemo nekatere osnovne statistične karakteristike podatkov kot so povprečje, standardni odklon, minimum, maksimum, percentili in intervali zaupanja. Te karakteristike načeloma iščemo za vsak ciljni razred (Baesens, 2014).

2.4 Manjkajoče vrednost

Manjkajoče vrednosti so ena glavnih pomanjkljivosti podatkov, saj marsikatera tehnika ocenjevanja odpove, če te vrednosti podatki vsebujejo. Najpogosteje se pojavijo zaradi nerazkritja kakšnih informacij (stranka na primer ne želi razkriti njenega mesečnega prihodka) ali zaradi napake pri združevanju podatkov. Manjkajoče vrednosti lahko

nadomestimo, zbrišemo ali obdržimo. Pri zamenjavi manjkajočih vrednosti kvantitativne podatke najpogosteje nadomestimo s povprečjem znanih vrednosti, kategorične pa z modusom znanih vrednosti. Če se odločimo za izbris manjkajočih vrednosti, izbrišemo opazovanja, ki vsebujejo kako manjkajočo vrednost. Če pa se odločimo, da manjkajoče vrednosti obdržimo, je potrebno iskati njihovo interpretacijo. Na primer, če stranka ne želi dati podatka o njenem mesečnem prihodku, lahko interpretiramo kot trenutno nezaposlenost stranke. V praksi lahko najprej raziščemo povezavo med manjkajočimi vrednostmi in ciljno spremenljivko. Pri iskanju te povezave si lahko pomagamo s χ^2 testom. Če najdemo povezavo, lahko manjkajoče vrednosti obdržimo. Če povezave ni, lahko manjkajoče vrednosti spremenimo ali izbrišemo (Baesens, 2014).

2.5 Ugotavljanje in reševanje napačnih vrednosti

Napačne vrednosti so vrednosti, ki precej odstopajo od ostalih podatkov. Ločimo lahko dve vrsti napačnih vrednosti. Prva je veljavno opazovanje (npr. oseba ima 1 mio€ mesečnega prihodka), druga pa neveljavno opazovanje (npr. oseba je stara 400 let). To sta primera univariatnih napačnih vrednosti, poznamo pa tudi multivariatne napačne vrednosti (opazovanja, ki so napačna v več dimenzijah). Napačne vrednosti je potrebno zaznati in rešiti. Zaznamo jih lahko s pomočjo grafičnih metod, npr. risanjem histogramov ali grafov, izračunom minimalne in maksimalne vrednosti posameznih spremenljivk ali izračunom Z-vrednosti. Formula za izračun z-vrednosti je $z_i = \frac{x_i - \mu}{\sigma}$, kjer je μ povprečna vrednost posamezne spremenljivke in σ njen standardni odklon. Z-vrednost ima po definiciji povprečno vrednost 0 in standardni odklon 1. V grobem lahko rečemo, da imajo napačne vrednosti absolutno z-vrednost večjo od 3. Reševanje napačnih vrednosti pa je odvisno od tega, ali je napačna vrednost veljavno ali neveljavno opazovanje. Če je opazovanje neveljavno, lahko napačno vrednost rešimo po enakem postopku kot rešujemo manjkajoče vrednosti. Za veljavna opazovanja pa lahko uporabimo krajšanje ali omejevanje. To storimo tako, da spremenljivki postavimo spodnjo in zgornjo mejo in vsa opazovanja, ki so izven tega intervala ustrezno transformiramo (Baesens, 2014).

2.6 Standardizacija podatkov

Standardizacija podatkov je postopek transformiranja podatkov na enotno lestvico. Poznamo tri vrste standardiziranja podatkov. Prvi je min/max standardizacija, kjer uporabimo naslednjo formulo $X_{novi} = \frac{X_i - \min_{i=1,\dots,n} X_i}{\max_{i=1,\dots,n} X_i - \min_{i=1,\dots,n} X_i}$. V formuli vrednosti X_i označujejo vrednost posameznega opazovanja pred standardizacijo. Drugi možen postopek je standardizacija s pomočjo z-vrednosti, kjer vsem opazovanjem priredimo njihovo z-vrednost. Tretji postopek se imenuje decimalna standardizacija, kjer vrednosti opazovanj delimo z ustrezno veliko potenco števila 10, da dobimo decimalne vrednosti. Standardizacija podatkov je zelo uporabna pri uporabi regresijskih modelov, medtem ko jo nekatere druge metode kot npr. odločitvena drevesa ne potrebujejo (Baesens, 2014).

2.7 Kategorizacija

Kategorizacijo uporabljamo pri kategoričnih spremenljivkah za zmanjševanje števila kategorij, pri kvantitativnih spremenljivkah pa je uporabna za modeliranje nelinearnih efektov v linearnih modelih. Kategoriziramo lahko spremenljivke z različnimi metodami. Dve najosnovnejši sta kategoriziranje na enake intervale in kategoriziranje na enake frekvence. Pri kategoriziranju na enake intervale imajo vse kategorije enak razpon (npr. leta kategoriziramo kot 0-10,10-20,20-30,…), pri kategoriziranju na enake frekvence pa ima vsaka kategorija enako število opazovanj (Baesens, 2014).

2.8 Določanje spremenljivk

Pri selekcioniranju oz. določanju spremenljivk je cilj določiti spremenljivke, ki nam pomagajo pri napovedovanju ciljne spremenljivke. Tipična kreditna ocena vsebuje nekje 10 do 15 spremenljivk za ocenjevanje kreditnega tveganja. Pri določanju spremenljivk je glavna mera določanja korelacija med ciljno spremenljivko in spremenljivko. Kako bomo merili korelacijo, pa je odvisno od tipa ciljne spremenljivke in ostalih spremenljivk.
Če sta tako ciljna kot ostale spremenljivke kvantitativna uporabimo za mero Pearsonov korelacijski koeficient, ki ga izračunamo kot

$$\rho_P = \frac{\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i-\bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i-\bar{Y})^2}}. \quad (7)$$

Pearsonov korelacijski koeficient meri linearno odvisnost dveh spremenljivk in variira med -1 in +1. Običajno spremenljivke katerih Pearsonov korelacijski koeficient je po absolutni vrednosti večji od 0.5 dobro pojasnjujejo ciljno spremenljivko.

Če je ciljna spremenljivka kategorična in ostale spremenljivke kvalitativne (velja tudi obratno), merimo korelacijo s Fisherjevo oceno, ki jo izračunamo kot

$$\frac{|\overline{X_d}-\overline{X_s}|}{\sqrt{s_d^2+s_s^2}}, \quad (8)$$

kjer označuje vrednost $\overline{X_d}$ ($\overline{X_s}$) povprečje dobrih (slabih) v skupini, vrednosti s_d^2 in s_s^2 pa sta pripadajoči varianci skupin. Visoka vrednost Fisherjeve ocene pomeni dobro zmožnost napovedovanja ciljne spremenljivke. Fisherjeva ocena se lahko posploši v analizo variance (ANOVA), torej lahko za izračun korelacije med kategorično ciljno spremenljivko in kvantitativno spremenljivko uporabimo tudi metodo ANOVA (Baesens, 2014).

Če uporabimo metodo ANOVA, dejansko merimo velikost efekta posamezne neodvisne spremenljivke na ciljno spremenljivko. Mera efekta je izražena s parametrom η^2 (angl. *Eta*

squared), v primeru, če imamo eno neodvisno spremenljivko in s parametrom delni η^2 (angl. *Partial eta squared*) v primeru, če imamo več neodvisnih spremenljivk. Parameter delni η^2 izračunamo kot $\eta^2 = \frac{SS_{efect}}{SS_{efect}+SS_{error}}$, kjer je SS_{efect} variacija efekta in SS_{error} variacija napake. Vrednosti parametra do približno 0.01 nakazujejo majhen vpliv efekta spremenljivke, vrednosti parametra do približno 0.06 nakazujejo srednje velik vpliv efekta ina vrednosti parametra večje od 0.14 nakazujejo na velik vpliv efekta posamezne spremenljivke (Hullet & Levine, 2002).

Če so tako ciljna kot ostale spremenljivke kategorične pa merimo korelacijo z informacijo o vrednosti (angl. *Information value*), ki jo izračunamo kot

$$IV = \sum_{i=1}^{k}(Delež\ dobrih_i - Delež\ slabih_i) * WOE_i, \quad (9)$$

kjer k predstavlja število kategorij posamezne spremenljivke. Delež dobrih v posamezni kategoriji je izračunan kot količnik med vsemi dobrimi v tej kategoriji in vsemi dobrimi v celotnem vzorcu. Analogno dobimo delež slabih. Spremenljivko WOE za posamezno kategorijo izračunamo kot $WOE = \ln\frac{Delež\ dobrih}{Delež\ slabih}$. Glede na rezultat informacije o vrednosti nato ločimo, če posamezna kategorija ciljne ne predvideva (IV < 0.02), šibko predvideva (0.02 < IV < 0.1), srednje dobro predvideva (0.1 < IV < 0.3) ali dobro predvideva (IV > 0.3). Pri uporabi informacije o vrednosti vedno predpostavimo, da je bila spremenljivka kategorizirana (Baesens, 2014).

Največja slabost filtriranja je ta, da delujejo univariatno (za vsako dimenzijo posebej) in ne upoštevajo morebitne korelacije med dimenzijami. Pri selekciji spremenljivk moramo upoštevati tudi morebitne regulacije, ki določene karakteristike prepovedujejo. Za ocenjevanje kreditnega tveganja je na primer prepovedano uporabljati ločevanje glede na spol, raso ali vero (Baesens, 2014).

2.9 Proces podatkovnega rudarjenja

Na podatkovno rudarjenje je potrebno gledati kot na proces. Ko izvajamo samo rudarjenje je tako kot pri vseh statističnih analizah potrebno imeti jasne namene analize. Poleg samega razumevanja algoritmov podatkovnega rudarjenja je potrebno poznati tudi ciljno področje in ustvariti ustrezno strategijo modeliranja. Proces podatkovnega rudarjenja bi lahko opisali z naslednjimi medsebojno povezanimi koraki (Ledolter, 2013):

1. Učinkovito shranjevanje podatkov in njihova filtracija.
2. Določanje primerne ciljne spremenljivke in določanje števila spremenljivk, ki bodo pojasnjevale ciljno spremenljivko.

3. Pregled in zaznavanje manjkajočih ter napačnih vrednosti v podatkih, ter reševanje tega problema z eno od prej naštetih metod v podpoglavju o manjkajočih ter napačnih vrednostih.
4. Razdelitev podatkov na dve podmnožici: podmnožico namenjeno učenju in podmnožico namenjeno testiranju. Če imamo opravka z zelo veliko množico podatkov, je potrebno iz teh narediti reprezentativen vzorec, na katerem nato izvajamo metode podatkovnega rudarjenja.
5. Pred uporabo metod podatkovnega rudarjenja moramo raziskati podatke. To najlažje naredimo z risanjem črtnih grafov, grafov raztrosa (angl. *Scatter plot*), histogrami, grafov za prikazovanje korelacijskih matrik itd. Paziti moramo na uporabo pravilne skale, pravilno označevanje morebitne težave zaradi stratificiranja in združevanja.
6. Naredimo povzetek podatkov, ki običajno vsebuje zbirne statistike kot so povprečje, mediana in percentili, standardni odklon, korelacije in tudi morebitna naprednejša povzetja kot so npr. glavni sestavni deli (angl. *Principal components*).
7. Na podatkih izvršimo želeno metodo podatkovnega rudarjenja. Izbira metode je odvisna od same narave problema.
8. Ugotovitve dobljene z učenjem metode na podmnožici namenjeni za učenje potrdimo na testni podmnožici.
9. Na koncu je potrebno spoznanja dobljena z metodami podatkovnega rudarjenja še implementirati in pojasniti.

3 ANALIZA PODATKOV

Za namene primerjave modelov podatkovnega rudarjenja za ocenjevanje kreditnega tveganja bom uporabil podatke dostopne na spletni strani http://www.kaggle.com. Uporabil bom javno dostopno podatkovno bazo, ki so jo na spletni strani objavili za tekmovanje GiveMeSomeCredit. Podatkovna baza je na voljo na naslovu https://www.kaggle.com/c/GiveMeSomeCredit/data . Za potrebe magistrske naloge bom uporabil podatke, ki se nahajajo v datoteki cs-training.csv. V spodnji tabeli so prikazani ime spremenljivke, njen opis, njeno kodiranje, ki ga bom uporabil za lažje branje, ter tip posamezne spremenljivke.

Tabela 1: Razlaga pomena spremenljivk v podatkih in tip spremenljivke

Ime spremenljivke	**Opis spremenljivke**	**Kodirana vrednost**	**Tip spremenljivke**
SeriousDlqin2yrs	Posameznik v roku dveh let zamuja s plačilom dolga več kot 90 dni	x_1	Binarna
RevolvingUtilizationOfUnsecuredLoans	Skupni znesek na kreditnih karticah in osebnih kreditnih linijah, kjer so izvzete nepremičnine in obroki posojila (npr. avtomobilsko posojilo), deljen z vsoto vseh kreditnih limitov	x_2	Odstotek
Age	Starost prosilca posojila	x_3	Celoštevilska
NumberOfTime30-59DaysPastDueNotWorse	Kolikokrat je posojilojemalec v zadnjih dveh letih zamujal s plačili za 30-59 dni (slabše ni všteto)	x_4	Celoštevilska
DebtRatio	Razmerje mesečnega plačila dolga, preživnine, življenjskih stroškov in posameznikovega bruto dohodka	x_5	Odstotek
MonthlyIncome	Posameznikov mesečni prihodek	x_6	Realno število

Se nadaljuje

Nadaljevanje

Ime spremenljivke	Opis spremenljivke	Kodirana vrednost	Tip spremenljivke
NumberOfOpenCreditLinesAndLoans	Število odprtih posojil (hipoteka ali avtomobilsko posojilo) in kreditnih linij	x_7	Celoštevilska
NumberOfTimes90DaysLate	Kolikokrat je posojilojemalec v zadnjih dveh letih zamujal s plačili za več kot 90 dni	x_8	Celoštevilska
NumberRealEstateLoansOrLines	Število hipotek in nepremičninskih posojil, kamor so vključene tudi linije domačega lastniškega kapitala	x_9	Celoštevilska
NumberOfTime60-89DaysPastDueNotWorse	Kolikokrat je posojilojemalec v zadnjih dveh letih zamujal s plačili za 60-89 dni (slabše ni všteto)	x_{10}	Celoštevilska
NumberOfDependents	Število vzdrževanih družinskih članov, kjer je posojilojemalec izključen	x_{11}	Celoštevilska

3.1 Opisne statistike spremenljivk

Vsaka spremenljivka vsebuje 150000 opazovanj, torej imamo skupaj 1650000 vhodnih podatkov. V nadaljevanju bomo raziskali vsako spremenljivko posebej in podali nekatere opisne statistike. Za vsako spremenljivko bomo podali njeno maksimalno in minimalno vrednost, mediano, povprečje prvi in tretji kvartil ter število manjkajočih vrednosti. Pri

podajanju opisnih karakteristik spremenljivk uporabljam njihove kodirane vrednosti, ki so podane v Tabeli 1.

Minimalna in maksimalna vrednost spremenljivke x_1 sta 0 in 1. Mediana, prvi in tretji kvartil so vsi enaki nič. Manjkajočih vrednosti pri tej spremenljivki ni. Povprečna vrednost spremenljivke je 0.06684, kar pomeni da se od večine prosilcev, ki jim je bil odobren kredit pričakuje, da bodo tega odplačali.

Minimalna in maksimalna vrednost spremenljivke x_2 je 0 in 50708. Prvi kvartil je 0.03, tretji kvartil je 0.56. Mediana te spremenljivke je enaka 0.15, njeno povprečje pa je 6.05, kar pomeni da ima prosilec v povprečju približno šestkrat višjo bilanco na kreditnih karticah kot je njegov kreditni limit. Glede na vrednosti povprečja in mediane vidimo, da je porazdelitev populacije te spremenljivke izrazito asimetrična. Manjkajočih vrednosti tu ni, glede na maksimalno vrednost spremenljivke pa je precej verjetno, da imamo opravka z nepravilnimi vnosi.

Minimalna in maksimalna vrednost spremenljivke x_3 je 0 in 109. Prvi kvartil je 41, tretji kvartil je 63. Mediana spremenljivke je enaka 52. Povprečna starost prosilca, ki mu je bil odobren kredit je 52. Manjkajočih vrednosti pri tej spremenljivki ni, glede na minimalno in maksimalno vrednost pa ima spremenljivka verjetno nepravilne vnose.

Minimalna in maksimalna vrednost spremenljivke x_4 je 0 in 98. Njen prvi in tretji kvartil ter mediana so enaki 0. Povprečna vrednost spremenljivke je 0.421, manjkajočih vrednosti pri tej spremenljivki ni. Glede na vrednosti minimuma, maksimuma in vrednosti obeh kvartilov lahko pričakujemo tudi tu nepravilne vnose.

Minimalna in maksimalna vrednost spremenljivke x_5 je 0 in 329664. Njen prvi kvartil je 0.2, tretji kvartil pa 0.9. Mediana spremenljivke je 0.4, njena povprečna vrednost pa 353. Manjkajočih vrednosti pri tej spremenljivki ni, prav tako najverjetneje ni napačnih vnosov, vendar se moramo v to še prepričati z nadaljnjo analizo.

Minimalna in maksimalna vrednost spremenljivke x_6 je 0 in 3008750. Njen prvi kvartil je 3400, tretji kvartil pa 8249. Mediana spremenljivke je 5400, povprečna vrednost pa 6670. Glede na naravo spremenljivke bi morebitne napačne vrednosti lahko smatrali kot veljavna opazovanja. Pri tej spremenljivki imamo 29731 manjkajočih vnosov, kar predstavlja 19.82% podatkov.

Minimalna in maksimalna vrednost spremenljivke x_7 je 0 in 58. Njen prvi kvartil je 5, tretji kvartil pa 11. Mediana spremenljivke je 8, povprečna vrednost pa 8.453, kar lahko interpretiramo kot dejstvo, da ima prosilec, ki mu je bilo odobreno posojilo v povprečju 8 kreditnih linij. Manjkajočih vrednosti spremenljivka nima, prav tako najverjetneje ni napačnih vnosov.

Minimalna in maksimalna vrednost spremenljivke x_8 je 0 in 98. Njen prvi kvartil je 0, prav tako tudi tretji kvartil in mediana. Spremenljivka ima povprečno vrednost enako 0.266 in ne vsebuje manjkajočih vnosov. Glede na naravo spremenljivke lahko zaključimo tudi, da spremenljivka ne vsebuje napačnih vnosov, kar bomo preverili še grafično.

Minimalna in maksimalna vrednost spremenljivke x_9 je 0 in 54. Njen prvi kvartil je 0, tretji kvartil je 2, mediana pa 1. Povprečna vrednost spremenljivke je 1.018, manjkajočih vrednosti ni, morebitne napačne vnose bomo preverili nekoliko kasneje. V povprečju ima torej prosilec, ki mu je bilo posojilo odobreno eno hipotekarno posojilo.

Minimalna in maksimalna vrednost spremenljivke x_{10} je 0 in 98. Njen prvi in tretji kvartil ter mediana so enaki 0. Povprečna vrednost spremenljivke je 0.2404. Manjkajočih vrednosti spremenljivka nima, morebitnih napačnih vnosov zaradi same narave spremenljivke ni pričakovati.

Minimalna in maksimalna vrednost spremenljivke x_{11} je 0 in 20. Prvi kvartil in mediana sta enaka 0. Tretji kvartil spremenljivke je 1, njeno povprečje pa 0.757. Spremenljivka ima 3924 manjkajočih vrednosti, kar predstavlja 2,62% podatkov. Glede na naravo spremenljivke lahko zaključimo, da tu ni napačnih vnosov.

3.2 Čiščenje in redukcija podatkov

Kot že omenjeno po začetnem pregledu podatkov sledi njihovo čiščenje in morebitna redukcija. Ciljna spremenljivka v naši analizi je x_1. Ta nima manjkajočih in napačnih vrednosti, zato čiščenje pri njej ni potrebno. Prav tako čiščenje ne bo potrebno pri spremenljivki x_3. Na prvi pogled morda maksimalna vrednost te spremenljivke zgleda kot napačna, toda če upoštevamo dejstvo, da ima najstarejši Zemljan 122 let, lahko predpostavimo, da je možno, da ima prosilec 109 let. Zato torej te vrednosti ne bomo smatrali kot napačno. Ravno tako čiščenje ne bo potrebno pri spremenljivkah x_7 in x_9, saj nimamo nikjer nikakršne omejitve o številu odprtih kreditnih linij ali številu nepremičninskih posojil ali hipotek, ki jih posameznik lahko ima. Prav tako ni problematična spremenljivka x_{11}, saj ne moremo z gotovostjo trditi, koliko članov nek posameznik vzdržuje. Pri tej spremenljivki bomo manjkajoče vrednosti nadomestili z modusom, ki je enak 0. Spreminjali ne bomo niti spremenljivk x_4, x_{10} in x_8, saj nam vsaka od teh pove, kolikokrat je posameznik v preteklosti zamujal s plačili dolga. Čeprav se morda na prvi pogled zdijo maksimalne vrednosti teh spremenljivk velike, je možno da posameznik 98-krat zamudi s plačilom med 30 in 59 dnevi, 60 in 89 dnevi ali več kot 90 dni.

Pri spremenljivki x_2 so nekatere vrednosti glede na tip podatka (odstotek) vprašljive, saj presegajo 1. Toda tu je možno najti razlago. Demonstrirajmo jo na primeru. Denimo, da ima stranka na kreditni kartici 507.08€ neporavnanega dolga in denimo, da je stranka nato zaprla račun na kreditni kartici. Njen neporavnan dolg še vedno znaša 507.08€, njen limit na kreditnih karticah pa je po novem 0€. Pri računanju razmerja med zneskom skupnega dolga in vsoto vseh kreditnih linij tu naletimo na problem, saj pridemo do nedefiniranega izraza (deljenje z 0). Da se določi vrednost tega razmerja je torej potrebno zaokrožiti vsoto kreditnih linij na 0.01€. Tako pridemo do razmerja med bilanco skupnega dolga in vsote vseh kreditnih linij, ki v tem primeru znaša 50708. Ta razlaga je moža pri vseh vrednostih spremenljivke x_2, katerih vrednost presega 1.

Posebno pozornost pa bomo namenili spremenljivkam x_5 in x_6. Pri spremenljivki x_6 imamo veliko manjkajočih vrednosti. Te bomo nadomestili s povprečno vrednostjo znanih vrednosti, ki znaša 6670€. Posebno pozornost si zaslužijo vrednosti spremenljivke x_5 pri tistih opazovanjih, ki jim manjka vrednost x_6. Za izračun količnika x_5 je namreč potreben podatek o mesečnem bruto prihodku posameznika. Glede na to, da pri tej spremenljivki ni manjkajočih vrednosti, je banka verjetno nekako ocenila posameznikov mesečni prihodek. Prav tako so za namene izračuna spremenljivke x_5 verjetno uporabili ocenjen mesečni prihodek pri posameznik, ki so navedli, da je njihov mesečni prihodek enak 0. Pravega ozadja žal ne poznamo, saj ni dostopa do teh informacij, torej je vprašanje, ali je ta spremenljivka relevantna za nadaljnjo analizo. Iz podatkov in informacij, ki jih imamo je namreč nemogoče zaključiti ali pri izračunu spremenljivke x_5 banka smatra manjkajoče vrednosti mesečnega prihodka in nični mesečni prihodek na enak način.

Redukcijo podatkov bomo pri analiz uporabili po potrebi. Z redukcijo želimo zmanjšati računsko zahtevnost problema, saj nekatere metode (SVM, nevronske mreže) pri veliki količini podatkov potrebujejo veliko časa za končni izračun. Če bomo uporabili redukcijo, bo to omenjeno v nadaljevanju.

3.3 Določanje spremenljivk

V tem podpoglavju bom preveril korelacijo med spremenljivkami in podal interpretacijo le-te.

Preverimo najprej morebitno korelacijo med spremenljivkami, ki jih bom v empiričnem delu magistrske naloge uporabil za napovedovanje vrednosti spremenljivke x_1. Tu velja poudariti, da morebitna korelacija med njimi ne vpliva na pristranskost samih cenilk koeficientov, temveč poveča njihovo standardno napako. V primeru zaznavanja korelacije bomo torej še vedno vključili vse spremenljivke v model. Spodnja tabela nam prikazuje korelacijsko matriko spremenljivk.

Tabela 2: *Korelacijska matrika spremenljivk*

	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}
x_2	1	-0.01	0	0	0.01	-0.01	0	0.01	0	0
x_3	-0.01	1	-0.06	0.02	0.03	0.15	-0.06	0.03	-0.06	-0.22
x_4	0	-0.06	1	-0.01	-0.01	-0.06	0.98	-0.03	0.99	0
x_5	0	0.02	-0.01	1	-0.01	0.05	-0.01	0.12	-0.01	-0.04
x_6	0.01	0.03	-0.01	-0.01	1	0.08	-0.01	0.11	-0.01	0.06
x_7	-0.01	0.15	-0.06	0.05	0.08	1	-0.08	0.43	-0.07	0.07
x_8	0	-0.06	0.98	-0.01	-0.01	-0.08	1	-0.05	0.99	-0.01
x_9	0.01	0.03	-0.03	0.12	0.11	0.43	-0.05	1	-0.04	0.13
x_{10}	0	-0.06	0.99	-0.01	-0.01	-0.07	0.99	-0.04	1	-0.01
x_{11}	0	-0.22	0	-0.04	0.06	0.07	-0.01	0.13	-0.01	1

Spremenljivka x_4 ima skoraj popolno korelacijo s spremenljivko x_{10} (korelacijski koeficient je enak 0.99) in s spremenljivko x_8 (korelacijski koeficient je enak 0.98). Prav tako sta skoraj popolno korelirani spremenljivki x_{10} in x_8 (korelacijski koeficient je enak 0.99). Te spremenljivke so torej paroma linearno odvisne, kar pomeni, da se premik enega izraža v premiku drugega v isti smeri. Spremenljivki x_9 in x_7 imata srednje veliko korelacijo (korelacijski koeficient je enak 0.43). Ostale spremenljivke imajo medsebojno korelacijo skoraj nično.

Kot že omenjeno bom ocenjeval verjetnost posameznikovega neizpolnjevanja obveznosti v roku dveh let, torej bo ciljna spremenljivka x_1. Ker ima ta spremenljivka samo dva možna izida (1 za neizpolnitev in 0 za preživetje), jo kategoriziramo kot binarno kategorično spremenljivko. Ostale spremenljivke lahko smatramo kot kvantitativne. Za preverjanje efekta posazemzne spremenljivke na ciljno spremenljivko bom izmeril parameter delni η^2.

Tabela 3: *Velikost efekta posamezne spremenljivke merjena s parametrom delni* η^2

Spremenljivka	**Delni η^2**
x_2	0.0000048
x_3	0.0085000
x_4	0.0120000
x_5	0.0000120
x_6	0.0002000
x_7	0.0001800
x_8	0.0100000
x_9	0.0000092

Se nadaljuje

Nadaljevanje

Spremenljivka	Delni η^2
x_{10}	0.0270000
x_{11}	0.0004900

Glede na dobljene rezultate imata zgolj spremenljivki x_4 in x_8 srednje velik efekt na ciljno spremenljivko. Na tem mestu velja poudariti, da mera efekta ne pomeni nujno slabe zmožnosti napovedovanja modela (poudariti je potrebno, da bi bila v idealnem primeru vsota vseh delnih efektov blizu 1). Glede na omejitve podatkov, bi torej v empirični analizi še vedno bilo smiselno obdržati vse spremenljivke.

4 EMPIRIČNI REZULTATI

V tem poglavju predstavim metodologijo, s katero analiziram podatke. Za obdelavo podatkov sem uporabil programski jezik R, ki je zelo uporaben za statistično analizo ali uporabo metod podatkovnega rudarjenja. Algoritmi, ki jih v delu uporabljam za analizo podatkov, so vsi že vgrajeni v programskem jeziku R znotraj različnih paketov.

4.1 Diskriminantna analiza

Pri diskriminantni analizi sem uporabil vsa prečiščena opazovanja, vključenih je bilo vseh 11 spremenljivk. Za učenje sem uporabil 70% podatkov, za testiranje pa 30%.

Program nam vrne pogojni verjetnosti $P(y = 0|x)$ in $P(y = 1|x)$, kar v našem primeru pomeni verjetnost posameznikovega finančnega preživetja dobe dveh let glede na dane karakteristike in verjetnost posameznikovega neizpolnjevanja obveznosti v dobi dveh let glede na posameznikove karakteristike. Glede na dane podatke, je pogojna verjetnost posameznikovega preživetja dobe dveh let 0.9333 in pogojna verjetnost posameznikovega neizpolnjevanja obveznosti v dobi dveh let 0.0667. Program nam tudi izračuna ocene koeficientov posamezne karakteristike, kar je prikazano v Tabeli 4.

Tabela 4: Ocene koeficientov pri diskriminantni analizi

Koeficient	Ocena
x_2	$-5.78*10^{-5}$
x_3	$-2.81*10^{-2}$
x_4	$8.56*10^{-1}$
x_5	$-2.75*10^{-6}$
x_6	$-5.16*10^{-6}$

Se nadaljuje

Nadaljevanje

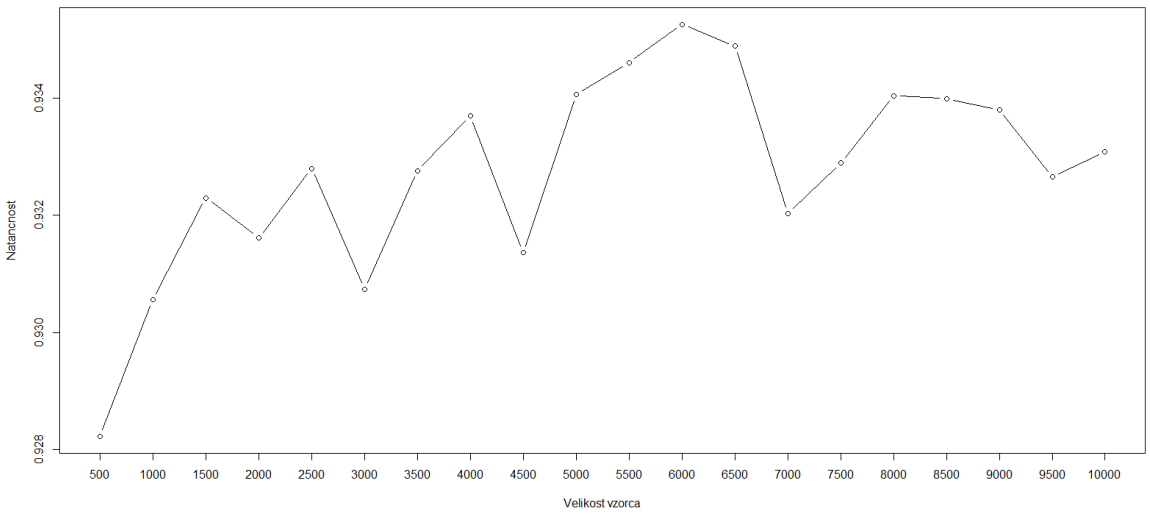
Koeficient	Ocena
x_7	-1.23*10^{-2}
x_8	8.92*10^{-1}
x_9	1.11*10^{-2}
x_{10}	-1.64
x_{11}	7.31*10^{-2}

Iz Tabele 4 vidimo, da ima najvišji mejni vpliv spremenljivka x_{10}. Negativen predznak nakazuje na dejstvo, da večkrat kot posameznik zamudi s plačilom obveznosti med 60 in 89 dni, bolj se posameznik nagiba k preživetju. Nekoliko presenetljiva sta predznaka pri spremenljivkah x_4 in x_8. Njun predznak je pozitiven, kar pomeni, da če posameznik zamuja s plačilom obveznosti od 30 do 59 dni ali več kot 90 dni, da verjetnost neizpolnjevanja obveznosti tega posameznika v roku 2 let naraste. Ravno tako posameznikova verjetnost neizpolnjevanja obveznosti narašča z večanjem števila hipotek in nepremičninskih posojil ter večanjem števila vzdrževanih članov (pozitivna predznaka pri ocenah koeficientov x_9 in x_{11}). Po drugi strani pa verjetnost posameznikovega neizpolnjevanja obveznosti pada z večanjem skupnega zneska na njegovih kreditnih karticah in osebnih kreditnih linijah (negativen predznak spremenljivke x_2), z večanjem posameznikove starosti (negativen predznak pri oceni spremenljivke x_3), z večanjem posameznikovega razmerja med mesečnim plačilom dolga in njegovim bruto dohodkom (negativen predznak spremenljivke x_5), višjim posameznikovim mesečnim dohodkom (negativen predznak pri oceni spremenljivke x_6) ter večanjem števila odprtih posojil (negativen predznak pri oceni spremenljivke x_7).

Dobljene ocene iz množice za učenje nato uporabimo še na množici za testiranje, kjer nato preverimo natančnost modela. Diskriminantna analiza ima v tem primeru 93.367% pravilno napovedanih rezultatov.

Nadalje sem analiziral vpliv spremembe velikosti vzorca na natančnost metode. Natančnost modela sem testiral na 20 različno velikih vzorcih velikosti med 500 in 10000. Za vsako velikost vzorca sem testiranje ponovil na 30 različnih vzorcih in za primerjavo vzel povprečje vseh 30 vzorcev pri vsaki velikosti. Postopal sem identično kot pri uvodnem primeru, torej 70% vzorca sem namenil učenju, 30% pa testiranju. Diskriminantna analiza se je najbolje obnesla pri vzorcu velikosti 6000, kjer je bilo pravilno napovedanih 93.53% rezultatov. Spodnja slika nam prikazuje odvisnost natančnosti diskriminantne analize od velikosti vzorca.

Slika 5: Odvisnost natančnosti od velikosti vzorca pri diskriminantni analizi



4.2 Logistična regresija

Pri logistični regresiji smo uporabili vseh 150000 opazovanj vseh 11 spremenljivk. Za analizo sem uporabil očiščene podatke. 70% podatkov sem uporabil za učenje, 30% pa za testiranje. V Tabeli 5 so prikazani izpis ocen koeficientov, njihove standardne napake in p-vrednosti.

Glede na ocene dobljene v Tabeli 5, lahko vidimo, da je mejni vpliv spremenljivk x_2, x_3, x_5, x_6, x_7 in x_{10} negativen, kar pomeni, da višje vrednosti teh spremenljivk nakazujejo višjo verjetnost posameznikovega neizpolnjevanja obveznosti v roku 2 let (konstanten člen je tu izvzet). Ocene ostalih koeficientov imajo pozitiven predznak, kar pomeni, da višje vrednosti ostalih spremenljivk nakazujejo na manjšo verjetnost posameznikovega neizpolnjevanja obveznosti v roku 2 let.

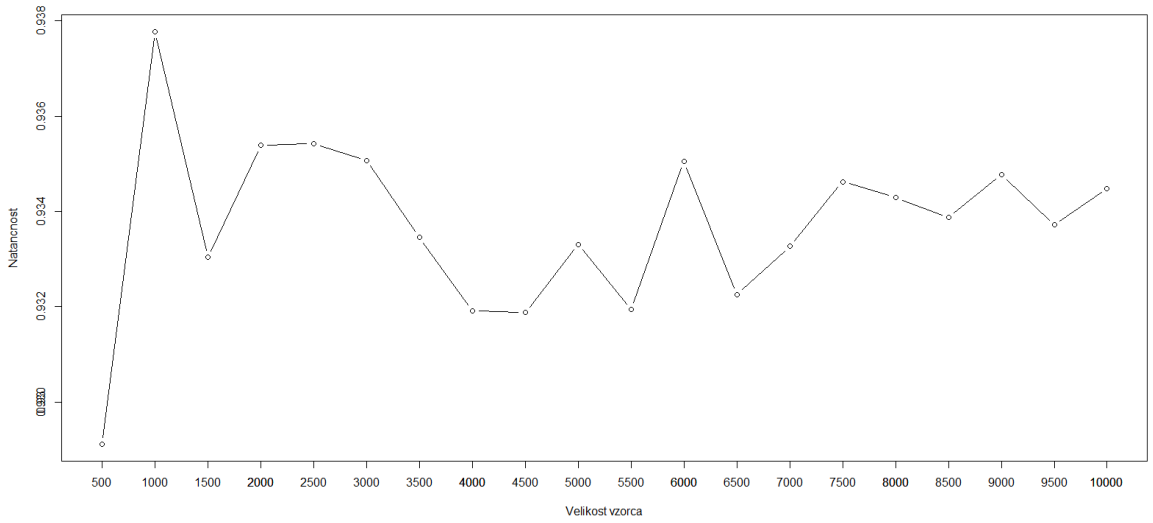
Dobljene ocene iz podatkov za učenje nato uporabimo za napovedovanje rezultatov na množici za testiranje. Na koncu še preverimo natančnost modela, kar storimo s primerjavo z uporabo nedoločene matrike (angl. *Confusion matrix*). Logistična regresija ima 92.84% pravilno napovedanih rezultatov.

Nadalje sem še preveril, kako na natančnost logistične regresije vpliva velikost vzorca. Iz očiščenih podatkov sem vzel 20 vzorcev velikosti med 500 in 10000 in na vsakem izvedel logistično regresijo na enak način kot je opisano zgoraj (torej 70% vzorca sem uporabil za učenje, 30% pa za testiranje in vsako vzorčenje ponovil tridesetkrat). Najvišja natančnost je bila dosežena pri vzorcu velikosti 1000, kjer je bilo pravilno napovedanih 93.78% rezultatov. Slika 6 prikazuje odvisnost natančnosti metode od velikosti vzorca.

Tabela 5: Ocene koeficientov in njihove standardne napake pri logistični regresiji

Koeficient	**Ocena**	**Standardna napaka**	**p-vrednost**
(Intercept)	-1.32	$5.01*10^{-2}$	10^{-16}
x_2	$-1.22*10^{-4}$	$1.12*10^{-4}$	$2.75 * 10^{-1}$
x_3	$-2.82*10^{-2}$	$9.91*10^{-4}$	10^{-16}
x_4	$5.03*10^{-1}$	$1.33*10^{-2}$	10^{-16}
x_5	$-5.14*10^{-6}$	$9.41*10^{-6}$	$5.85 * 10^{-1}$
x_6	$-3.37*10^{-5}$	$3.65*10^{-6}$	10^{-16}
x_7	$-7.62*10^{-3}$	$3.01*10^{-3}$	$1.12 * 10^{-2}$
x_8	$4.74*10^{-1}$	$1.83*10^{-2}$	10^{-16}
x_9	$6.09*10^{-2}$	$1.24*10^{-2}$	$8.31 * 10^{-7}$
x_{10}	$-9.46*10^{-1}$	$2.13*10^{-2}$	10^{-16}
x_{11}	$8.08*10^{-2}$	$1.09*10^{-2}$	$9.76 * 10^{-14}$

Slika 6: Odvisnost natančnosti logistične regresije od velikosti vzorca



4.3 Nevronske mreže

Nevronske mreže so računsko precej zahteven problem. Zaradi zmanjšanja časa izračuna metodo ne uporabimo na celotni prvotni populaciji, ampak na manjšem vzorcu prvotne populacije. Iz prvotnih podatkov naključno izberemo vzorec velikosti 7000 in na njem uporabimo metodo nevronske mreže. Nekatere osnovne opisne statistike posameznih spremenljivk izbranega vzorca so prikazane v Tabeli 6.

Tabela 6: Opisne statistike posamezne spremenljivke v izbranem vzorcu

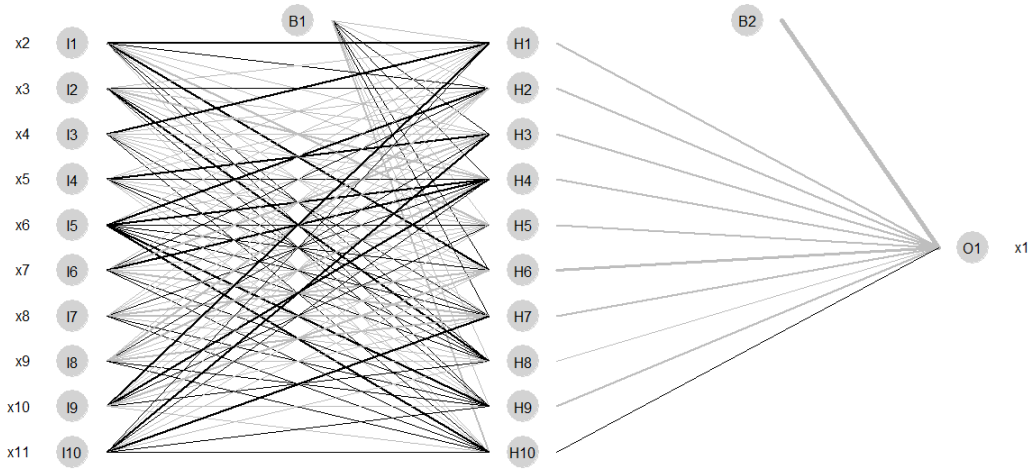
Spremenljivka	Prvi kvartil	Tretji kvartil	Mediana	Povprečna vrednost	Minimum	Maksimum
x_1	0	0	0	0.069	0	1
x_2	0.031	0.566	0.157	6.788	0	9340
x_3	41	63	52	52	21	103
x_4	0	0	0	0.413	0	98
x_5	0.173	0.889	0.375	357	0	60212
x_6	3833	7350	6500	6543	0	304000
x_7	5	11	8	8.52	0	49
x_8	0	0	0	0.24	0	98
x_9	0	2	1	1.027	0	13
x_{10}	0	0	0	0.224	0	98
x_{11}	0	1	0	0.735	0	8

Če primerjamo opisne statistike vzorca z opisnimi statistikami celotne populacije, podanimi v poglavju 4.1, vidimo, da na vzorcu ni bistvenih odstopanj, kar pomeni, da dobljeni vzorec dobro reprezentira celotno populacijo.

Podatke nato standardiziramo s pomočjo min/max standardizacije. Slika 7 prikazuje graf nevronske mreže.

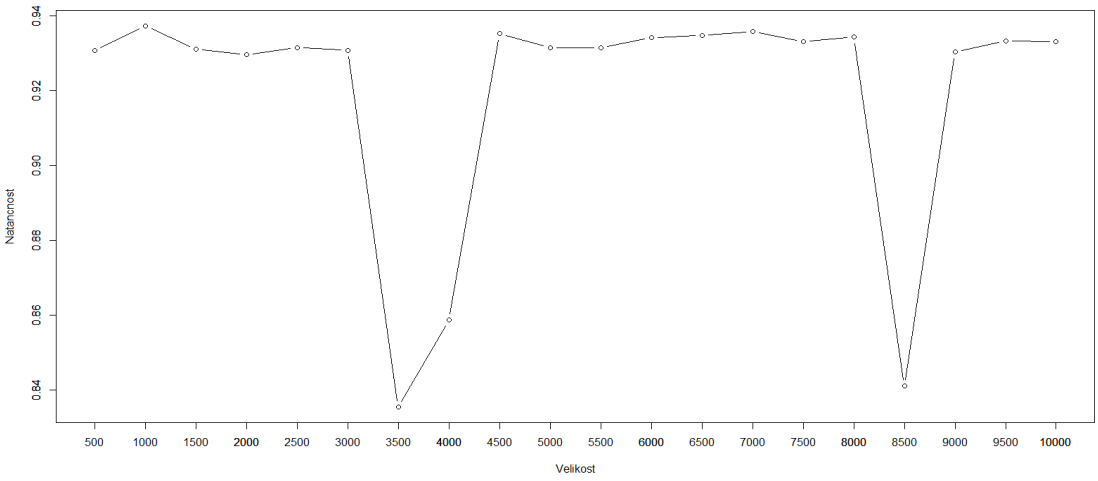
Za napovedovanje verjetnosti neizpolnitve obveznosti do banke je bilo uporabljenih vseh 10 spremenljivk. Nevronska mreža sestoji iz desetih vhodnih plasti, ene izhodne plasti ter desetih skritih plasti (na Sliki 7 označene kot H1,…H10) enega skritega nevrona. Metoda je skupno izračunala 121 uteži. Ko dobljene uteži uporabimo še na množici za testiranje, ugotovimo, da v tem primeru metoda nevronskih mrež pravilno napove 93.38% vzorčnih rezultatov.

Slika 7: Nevronska mreža



Zanimivo je tudi primerjati natančnost metode nevronskih mrež glede na velikost vzorca. Na Sliki 8 je prikazan graf natančnosti napovedanih rezultatov iz testne množice.

Slika 8: *Primerjava natančnosti nevronskih mrež glede na velikost vzorca*



Kot primer sem vzel 20 vzorcev različnih velikosti, za vsako velikost sem vzorčenje ponovil desetkrat in za primerjavo vzel povprečno vrednost pri vsaki velikosti. Pri tej metodi sem vzorčenje ponovil le desetkrat, ker je metoda računsko zelo potratna. Najmanjši vzorec vsebuje 500 opazovanj, največji pa 10000.

Za dane podatke je največja natančnost, 93.73% pravilno napovedanih rezultatov, dosežena pri vzorcu velikosti 1000.

4.4 Metoda podpornih vektorjev

Pri metodi podpornih vektorjev analizo izvajamo na vzorcu prvotnih podatkov zaradi zmanjšanja računskega časa metode. Iz prvotnih podatkov sem vzel vzorec velikosti 7000, 70% podatkov iz vzorca sem uporabil za učenje, 30% za testiranje. Tabela 7 prikazuje nekatere osnovne statistične karakteristike vzorca. Če primerjamo dobljene opisne statistike s tistimi opisanimi v poglavju 4.1, ki veljajo za celotno populacijo, vidimo, da bistvenih odstopanj od populacijskih opisnih statistik ni, torej dobljeni vzorec dobro predstavlja celotno populacijo.

Tabela 7: *Statistične karakteristike vzorca*

Spremenljivka	**Prvi kvartil**	**Tretji kvartil**	**Mediana**	**Povprečna vrednost**	**Minimum**	**Maksimum**
x_1	0	0	0	0.064	0	1
x_2	0	0.544	0.152	6.652	0	13930
x_3	41	62	51	52.04	21	97
x_4	0	0	0	0.43	0	98
x_5	0.18	0.87	0.37	373	0	307000
x_6	4000	7442	6666	6657	0	582369
x_7	5	11	8	8.502	0	54
x_8	0	0	0	0.274	0	98
x_9	0	2	1	1.012	0	17
x_{10}	0	0	0	0.245	0	98
x_{11}	0	1	0	0.749	0	20

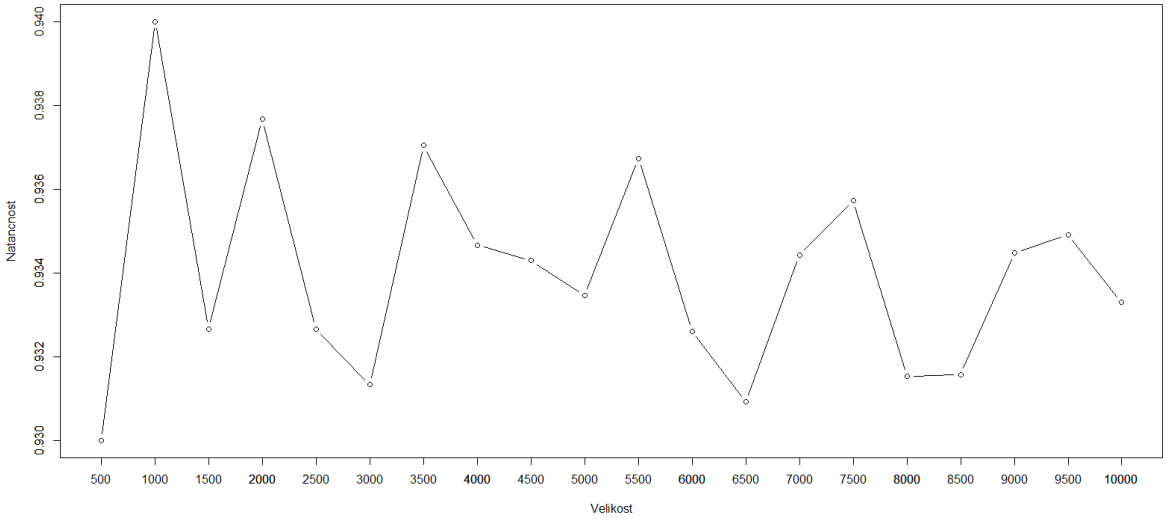
Standardizacije na podatkih ni potrebno izvajati, vzorec je vzet iz očiščenih podatkov. Pri izvajanju metode je potrebno določiti (angl. *tune*) parametra C in γ. Kot povedano, s parametrom omejimo območje podpornih vektorjev. Če je vrednost parametra prevelika, po nepotrebnem podaljšamo čas učenja. S parametrom γ omejujemo napako učenja. Če so vrednosti parametra visoke, bo metoda SVM skušala poiskati razdelitev, ki se eksaktno prilega podatkom, kar v praksi v primeru nelinearne delitve pomeni nepotrebno podaljšanje učenja. V našem primeru je optimalno razmerje med natančnostjo in časovno zahtevnostjo doseženo, če za C izberemo 1 in za γ 2. V tem primeru je delitev podatkov narejena s 1185 podpornimi vektorji. Enačba delitvene ravnine je torej izražena s 1185 podatki iz množice za učenje. Pri tako izbranih parametrih je pravilno napovedanih 93.35% razultatov iz množice za testiranje.

Tako kot pri metodi nevronskih mrež, je tudi pri metodi podpornih vektorjev zanimivo vprašanje, kako velikost vzorca vpliva na natančnost metode. Vpliv velikosti vzorca na

natančnost metode sem testiral na 20 različno velikih vzorcih, pri vsaki velikosti vzorca pa sem vzorčenje ponovil desetkrat in primerjal natančnost na povprečju desetih meritev. Tudi pri tej metodi sem podobno kot pri nevronskih mrežah vzorčenje ponovil manjkrat zaradi počasnosti izvedbe algoritma. Slika 14 prikazuje odvisnost natančnosti metode podpornih vektorjev od velikosti izbranega vzorca.

Najvišja natančnost metode je dosežena, ko vzamemo vzorec velikosti 1000. V tem primeru metoda podpornih vektorjev pravilno napove 94% rezultatov iz testne množice.

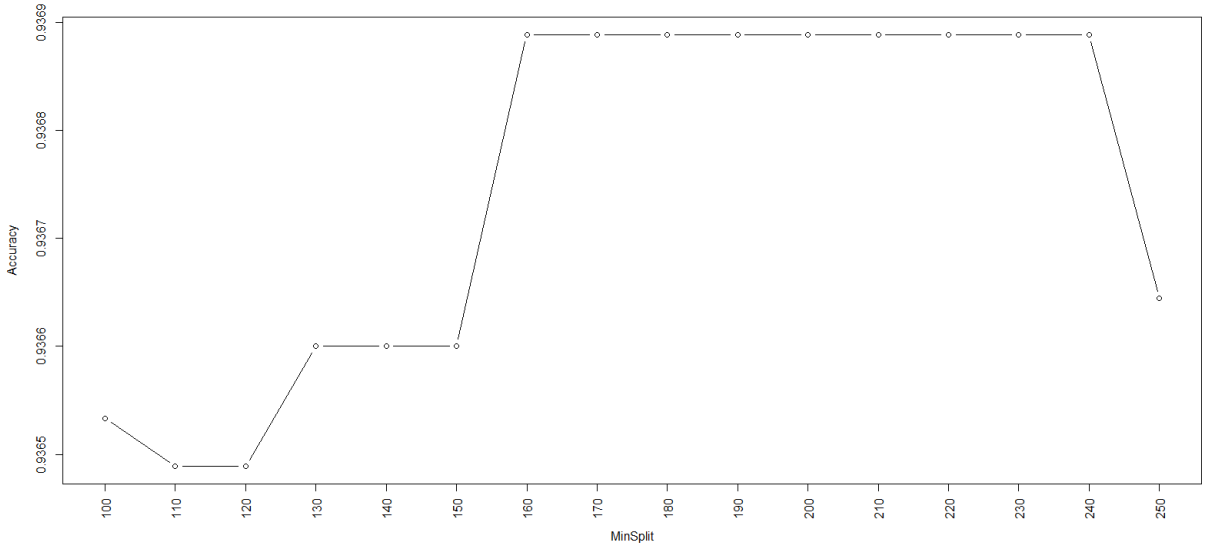
Slika 9: *Primerjava natančnosti metode podpornih vektorjev glede na velikost vzorca*



4.5 Odločitvena drevesa

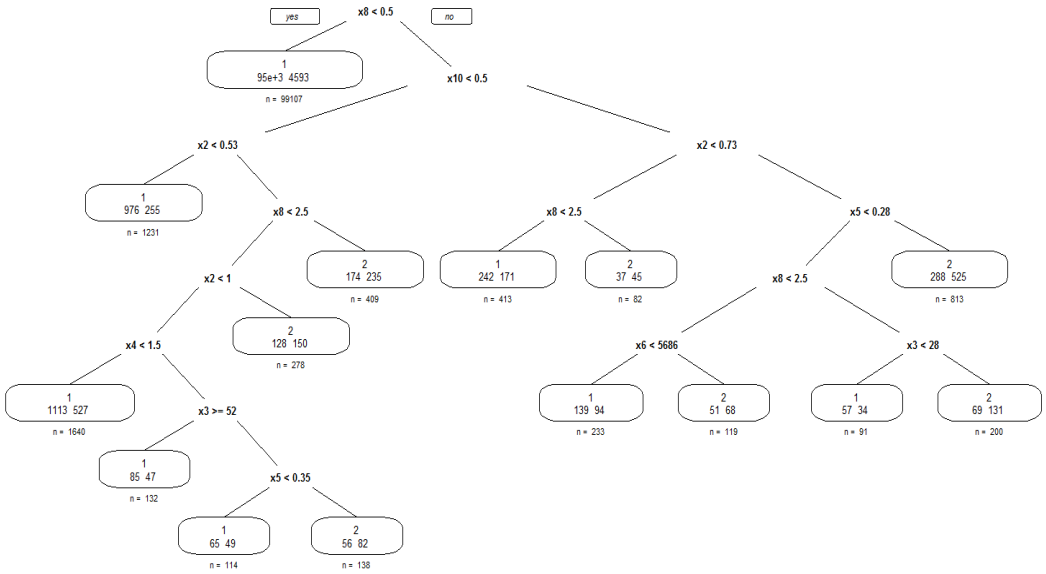
Pri metodi odločitvenih dreves sem uporabil vseh 150000 očiščenih podatkov. 70% teh podatkov sem uporabil za učenje, 30% pa za testiranje. Pri tej metodi je potrebno ugotoviti, kdaj je najbolje ustaviti rast drevesa. To določimo z dvema parametroma in sicer z minimalnim številom opazovanj spremenljivke, ki zadoščajo določenemu pogoju potrebnemu za nadaljnjo delitev drevesa in s parametrom, ki nam meri izboljšavo odločitve z nadaljnjo delitvijo (angl. *fit*). Slika 9 prikazuje natančnost napovedanih rezultatov glede na najmanjše število opazovanj določene spremenljivke. Vidimo, da je natančnost modela najvišja, ko se minimalno število opazovanj giblje med 160 in 240. Parameter, ki meri izboljšavo odločitve z nadaljnjo delitvijo je nastavljen na 0.001. Vsaka delitev, ki bi izboljšala odločitveno drevo za manj od zgornje vrednosti, ne bo izvedena.

Slika 10: *Graf natančnosti modela odločitvenih dreves glede na število minimalnih delitev*



V najboljšem primeru je natančnost odločitvenih dreves 93.69%. Na Sliki 10 je izrisano odločitveno drevo v najboljšem primeru.

Slika 11: *Odločitveno drevo*

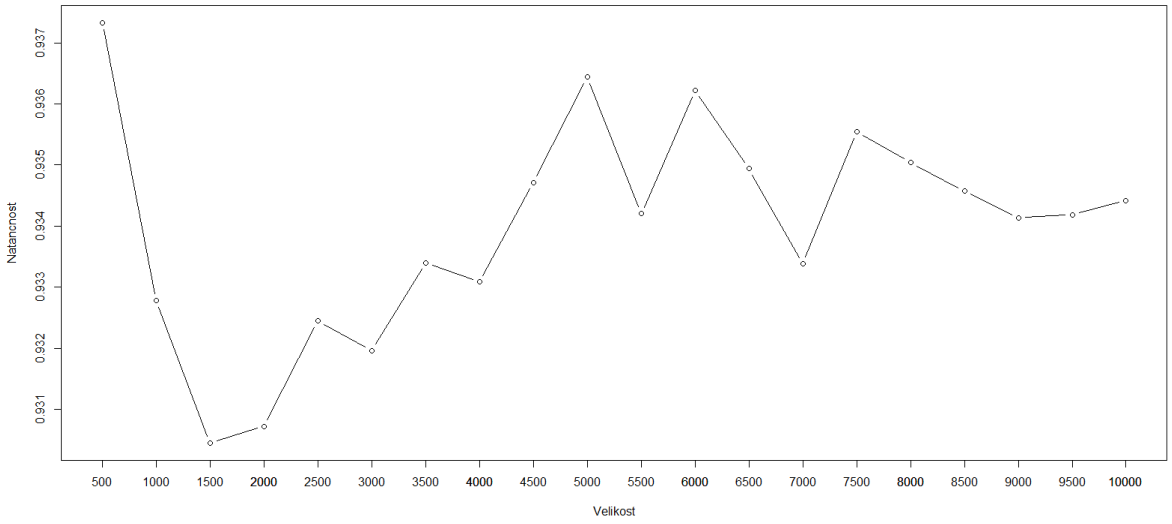


Kako interpretirati drevo na Sliki 10? Oglejmo si koren, kjer se nahaja spremenljivka x_8. Kriterij delitve pri tej spremenljivki je bila vrednost 0.5. 99107 opazovanj ima vrednost te spremenljivke manjšo od 0.5, 5893 pa večjo. Od teh 99107 jih je šlo dejansko 4593 v stečaj, ostalih 94514 jih je preživelo. 4593, ki jih je dejansko šlo v stečaj jih je bilo predvidenih napačno, saj glede na napoved drevesa še naprej izpolnjujejo obveznosti do banke. Oglejmo si še, kako pridemo do vrednosti pri listu $x_3 < 28$ (t.j. najbolj desni list). V

ta list je uvrščenih 200 posameznikov, ki izpolnjujejo naslednje kriterije: več kot 0.5-krat so v preteklosti že zamudili s plačilom obroka dolga, več kot 0.5-krat so v preteklosti s plačili zamujali med 60 in 89 dni, njihovo razmerje skupne bilance na kreditnih karticah in vsote vseh kreditnih linij je večje od 0.73, razmerje mesečnega plačila dolga in njihovega bruto dohodka je manjše od 0.28, v preteklosti so več kot 2.5-krat že zamudili s plačilom dolga za več kot 90 dni in so starejši od 28 let. Tem kriterijem zadošča 200 posameznikov, ki jim je drevo vsem napovedalo stečaj v roku dveh let. Od teh 200 jih je šlo v stečaj dejansko 69, 131 pa jih je preživelo.

Na koncu sem preveril še, kako velikost vzorca vpliva na natančnost metode odločitvenih dreves. Iz prvotnih očiščenih podatkov sem vzel 20 vzorcev velikosti med 500 in 10000 in na njih izvedel metodo na enak način, kot je opisano zgoraj (70% vzorca je namenjenega učenju, 30% pa testiranju, vsako vzorčenje sem ponovil tridesetkrat in za primerjavo natančnosti vzel povprečje pri vsaki velikosti vzorca). Metoda odločitvenih dreves je največjo natančnost, 93.73% pravilno napovedanih rezultatov, dosegla pri vzorcu velikosti 500. Slika 11 prikazuje spreminjanje natančnosti metode glede na velikost vzorca.

Slika 12: *Vpliv velikosti vzorca na natančnost odločitvenih dreves*



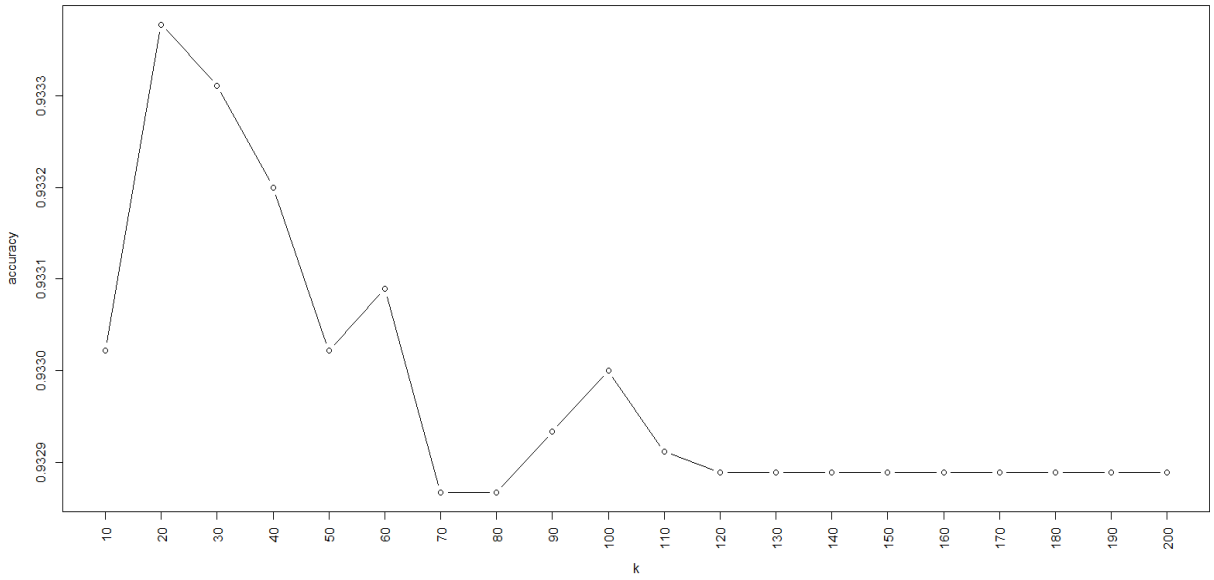
4.6 K-najbljižjih sosedov

Pri metodi k-najbližjih sem uporabil vseh 150000 očiščenih podatkov, katerih 70% sem uporabil za učenje in 30% za testiranje.

Ta metoda nam za razliko od logistične regresije in diskriminantne analize ne izpiše ocen koeficientov. Ob izvedbi programa metoda razporedi opazovanja iz množice za učenje v različne podskupine, hkrati pa se preveri tudi učinkovitost modela. Glavna naloga tu je določiti parameter k, ki ga določimo s poskušanjem. Slika 12 prikazuje gibanje pravilno

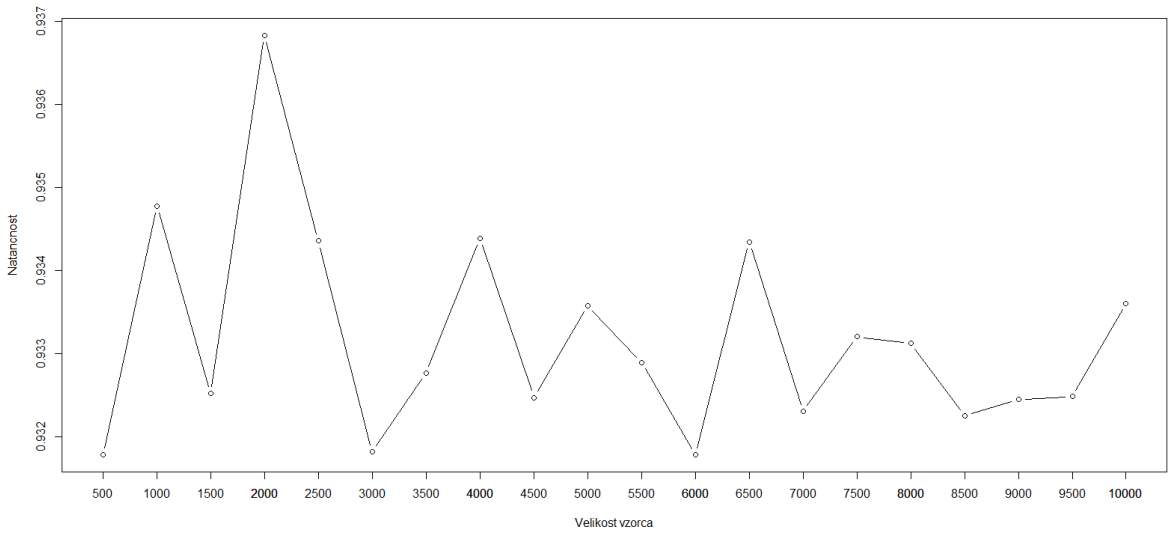
napovedanih rezultatov glede na različno izbrane parametre k. Medsebojno sem primerjal rezultate pri 20 različnih parametrih k, od 10 do 200 s korakom 10. Najvišja natančnost metode je dosežena, če za parameter k izberemo 20. V tem primeru metoda pravilno napove 93.34% rezultatov.

Slika 13: *Natančnost metode k-najbližjih sosedov pri različnih parametrih k*



Nadalje sem preveril še, kako se natančnost modela spreminja glede na velikost vzorca. Natančnost metode sem testiral na 20 vzorcih velikosti med 500 in 10000, postopal sem enako kot že navedeno (70% vzorca sem uporabil za učenje, 30% pa za testiranje, za vsako velikost sem vzorčenje ponovil tridesetkrat in vzel povprečje za primerjavo). Največ pravilno napovedanih rezultatov (93.68%) je bilo doseženih pri vzorcu velikosti 2000. Slika 13 prikazuje odvisnosti natančnosti metode od velikosti vzorca.

Slika 14: *Odvisnost natančnost metode k-najbližjih sosedov od velikosti vzorca*



4.7 Genetski algoritmi

Genetske algoritme lahko uporabljamo tudi za reševanje optimizacijskih problemov. Za namene magistrske naloge bom uporabil genetski algoritem za iskanje cenilke po metodi največjega verjetja. Cenilka, ki jo iščemo, zadošča naslednjem sistemu nelinearnih enačb

$$\sum_{i=1}^{N}\left(y_i - \Lambda(x_i'\beta)\right)x_i = 0, \tag{10}$$

kjer je $\Lambda(z) = \frac{e^z}{1+e^z}$ kumulativna porazdelitvena funkcija logistične porazdelitve. Za iskanje cenilke s pomočjo genetskega algoritma uporabimo vzorec velikosti 7000. V spodnji tabeli so predstavljene nekatere osnovne statistične karakteristike vzorca.

Tabela 8: Osnovne statistične karakteristike vzorca

Spremenljivka	**Prvi kvartil**	**Tretji kvartil**	**Mediana**	**Povprečna vrednost**	**Minimum**	**Maksimum**
x_1	0	0	0	0.069	0	1
x_2	0.031	0.583	0.158	2.192	0	5451
x_3	41	62	52	52.04	21	107
x_4	0	0	0	0.363	0	98
x_5	0.18	0.88	0.37	354	0	101320
x_6	3900	7208	6500	6541	0	237400
x_7	5	11	8	8.438	0	57

Se nadaljuje

Nadaljevanje

Spremenljivka	Prvi kvartil	Tretji kvartil	Mediana	Povprečna vrednost	Minimum	Maksimum
x_8	0	0	0	0.223	0	98
x_9	0	2	1	1.011	0	29
x_{10}	0	0	0	0.191	0	98
x_{11}	0	1	0	0.736	0	7

Če primerjamo dobljene opisne statistike s tistimi opisanimi v poglavju 4.1, ki veljajo za celotno populacijo, vidimo, da bistvenih odstopanj od populacijskih opisnih statistik ni, torej dobljeni vzorec ponovno dobro predstavlja celotno populacijo.

Za potrebe izvajanja genetskega algoritma ni potrebno narediti standardizacije podatkov. Za izračun cenilke bomo uporabili očiščene podatke.

Pri genetskem algoritmu lahko spreminjamo verjetnost mutacije in velikost vsake nadaljnje generacije. Seveda se tu pojavi vprašanje, če spreminjanje teh parametrov kako vpliva na natančnost metode. Za testiranje odvisnosti natančnosti od verjetnosti mutacije in velikosti nadaljnjih sem uporabil različne vrednosti: verjetnosti mutacije sem izbral kot 0.1, 0.15 in 0.2, velikosti nadaljnjih generacij pa so 100, 200 ali 300. Spodnja tabela nam prikazuje, kako se natančnost spreminja glede na različne vrednosti verjetnosti mutacije in velikosti nadaljnjih generacij.

Tabela 9: Odvisnost natančnosti (v %) od velikosti generacije in verjetnosti mutacije

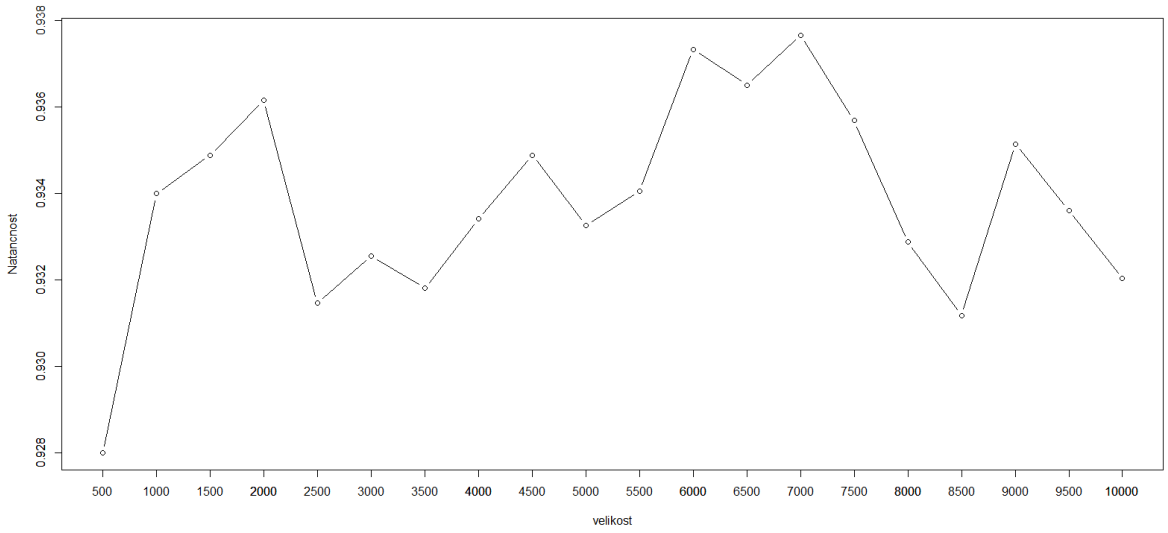
Verjetnost mutacije / Velikost generacije	0.1	0.15	0.2
100	94.24	94.24	94.24
200	94.24	94.24	94.24
300	94.24	94.24	94.24

Iz Tabele 9 vidimo, da je natančnost metode enaka ne glede na izbrano verjetnost mutacije ali velikost generacije. Za nadaljnjo empirično raziskavo izbira verjetnosti mutacije in velikosti nadaljnjih generacij ni bistvena, a je v tem primeru zaradi hitrejšega izračuna bolje vzeti manjšo generacijo, medtem ko verjetnost mutacije na rezultat ne vpliva. V vseh primerih nam metoda pravilno napove 94.24% rezultatov iz testne množice.

Nadalje nas zanima še, ali tudi velikost začetnega vzorca vpliva na natančnosti metode. Ali bo metoda pri različno velikih začetnih vzorcih vrnila različne rezultate? Da bi dobili

odgovor na to vprašanje, vzamemo iz prvotnih opazovanj različne velikosti vzorcev. Za namene testiranja odvisnosti natančnosti od velikosti vzorca sem uporabil 20 različnih vzorcev velikosti od 500 do 10000 in za vsako velikost vzorca ponovil vzorčenje desetkrat. Pri genetskih algoritmih sem podobno kot pri metodi podpornih vektorjev in pri nevronskih mrežah vzorčenje ponovil le desetkrat zaradi časovne potratnosti algoritma. Spodnja slika nam prikazuje odvisnost natančnosti metode od velikosti vzorca. Pri vsakem izvajanju genetskega algoritma sem tu vzel za verjetnost mutacije vrednost 0.1 in za velikost generacije 100, saj se natančnost po zgoraj pokazanem ne razlikuje za različne vrednosti teh parametrov. Iz spodnje slike je razvidno, da je najvišja natančnost modela, 93.77% pravilno napovedanih rezultatov, dosežena pri vzorcu velikosti 7000.

Slika 15: Odvisnost natančnosti od velikosti vzorca pri genetskem algoritmu



SKLEP

V tem poglavju predstavim rezultate empirične analize in opravim končni sklep. Rezultati so predstavljeni v več tabelah, kjer so po vrsti zapisani rezultati najboljše napovedi pri enakem vzorcu in najboljše napovedi pri poljubnem vzorcu. Na koncu še zaključim, kaj bi za banko pomenila izbira metode, ki bolje ocenjuje verjetnost neizpolnitve obveznosti posameznikov.

V Tabeli 10 so podani rezultati natančnosti metod pri vzorcu velikosti 7000.

Iz Tabele 10 je razvidno, da na danih podatkih pri enaki velikosti vzorca najbolje napovedujejo verjetnost neizpolnitve genetski algoritmi, ki pravilno napovejo 93.77% rezultatov. Razlike med pravilno napovedanimi so zelo majhne, saj najboljša metoda napove pravilno le 0.57% več kot najslabša metoda, v tem primeru diskriminantna analiza.

Tabela 10: Primerjava natančnosti metod pri enaki velikosti vzorca

Metoda	Natančnost (v %)
Diskriminantna analiza	93.20
Logistična regresija	93.33
Nevronske mreže	93.68
Odločitvena drevesa	93.38
k-najbližjih sosedov	93.23
Metoda podpornih vektorjev	93.44
Genetski algoritmi	93.77

Oglejmo si še, kakšna je najvišja natančnost posamezne metode in pri kateri velikosti vzorca je dosežena. Natančnost sem primerjal na 20 vzorcih velikosti med 500 in 10000. Pri vsaki velikosti sem vzorčenje ponovil desetkrat in primerjal povprečje pravilno napovedanih. Tabela 11 prikazuje najvišjo doseženo natančnost posamezne metode in velikost vzorca, kjer je bila ta natančnost dosežena.

Tabela 11: Najvišja natančnost posamezne metode in velikost vzorca, kjer je bila dosežena

Metoda	Najvišja natančnost (v %)	Velikost vzorca
Diskriminantna analiza	93.53	6000
Logistična regresija	93.78	1000
Nevronske mreže	93.73	1000
Odločitvena drevesa	93.73	500
k-najbližjih sosedov	93.68	2000
Metoda podpornih vektorjev	94.00	1000
Genetski algoritmi	93.77	7000

Najvišjo natančnost pri poljubni velikosti vzorca je dosegla metoda podpornih vektorjev, ki je pravilno napovedala 94% rezultatov na vzorcu velikosti 1000. Tudi najvišje dosežene natančnosti se med seboj ne razlikujejo precej, saj je med najvišjo in najnižjo vrednostjo razlika le 0.47%.

Nadalje se je pokazalo, da kljub temu, da so nekatere metode bolj natančne, pa imajo po drugi strani višji razpon natančnosti. Tabela 12 prikazuje najnižjo doseženo natančnost posamezne metode ter za vsako metodo prikaže rapon med najvišjo in najnižjo natančnostjo.

Če primerjamo Tabelo 11 in Tabelo 12, opazimo, da ima najmanjši razpon metoda k-najbljižjih sosedov. Pri tej metodi se najvišja in najnižja natančnost razlikujeta za 0.5 %. Najvišji razpon ima metoda nevronske mreže, pri kateri se najvišja in najnižja natančnost razlikujeta za 9.95%. Ta razlika bi lahko nakazovala na večjo občutljivost nevronskih mrež na osnovni vzorec, torej bi se uporaba te metode napram ostalim lahko izkazala za preveč nenatančno. Za večino metod velja, da je najnižja natančnost bila dosežena pri najmanjšem vzorcu, česar ne moremo trditi za nevronske mreže in odločitvena drevesa.

Tabela 122: Najnižja natančnost posamezne metode in razpon natančnosti

Metoda	**Najnižja natančnost (v %)**	**Velikost vzorca**	**Razpon (v %)**
Diskriminantna analiza	92.82	500	0.71
Logistična regresija	92.91	500	0.87
Nevronske mreže	83.78	3000	9.95
Odločitvena drevesa	93.04	1500	0.69
k-najbližjih sosedov	93.18	500, 3000 in 6000	0.5
Metoda podpornih vektorjev	93	500	1
Genetski algoritmi	92.8	500	0.97

Kaj za banko pomeni uporaba natančnejše metode za ocenjevanje verjetnosti neizpolnjevanja obveznosti v smislu pričakovanih izgube?

Denimo, da ima banka podatke o 10000 preteklih prosilcih, ki jim je bodisi odobrila ali zavrnila posojilo. Nadalje recimo še, da pri banki zaprosi za posojilo 1000 novih prosilcev, vsak želi posojilo v višini 100000€. Za ta navidezni portfelj predpostavimo še LGD parameter v vrednosti 0.95. LGD se v tem primeru lahko interpretira kot oportunitetna izguba dobrih strank, saj gre tu za stranke, ki so v resnici dobre, a so bile razvrščene kot slabe. Tabela 13 nam prikazuje pričakovane izgube banke pri uporabi različnih metod za ocenjevanje verjetnosti neizpolnjevanja obveznosti.

Iz Tabele 13 je razvidno, da je pri zgornjih pogojih najnižja pričakovana izguba pri uporabi nevronskih mrež in diskriminantne analize. V primeru, da banka uporablja namesto odločitvenih dreves genetske algoritme, bi pri pogojih kot so navedeni zgoraj bile njene pričakovane izgube za 5.88% višje. Če bi banka namesto odločitvenih dreves uporabljala diskriminantno analizo ali nevronske mreže, bi pri zgornjih pogojih bile njene pričakovane izgube za 3.03% višje. Če pa banka namesto odločitvenih dreves uporablja logistično regresijo, metodo k-najbljižjih sosedov ali metodo podpornih vektorjev, bi pri zgornjih pogojih njene pričakovane izgube bile za 4.48% višje.

Tabela 133: Primerjava potencialnih izgub pri uporabi različnih metod

Metoda	**Pravilno razvrščeni**	**Napačno razvrščeni**	**Pričakovana izguba (v €)**
Diskriminantna analiza	934	66	6 270 000
Logistična regresija	933	67	6 365 000
Nevronske mreže	934	66	6 270 000
Odločitvena drevesa	936	64	6 080 000
k-najbližjih sosedov	933	67	6 365 000
Metoda podpornih vektorjev	933	67	6 365 000
Genetski algoritmi	932	68	6 460 000

Z uporabo metod podatkovnega rudarjenja pri ocenjevanju kreditnega tveganja posameznika, bi banke torej glede na narejeno raziskavo lahko natančneje razvrstile nove prosilce in posledično znižale pričakovane izgube. Metode podatkovnega rudarjenja se po drugi strani slabše obnesejo pri interpretaciji vpliva posameznih spremenljivk na končni rezultat. Banke bi torej po eni strani natančneje razvrščale nove prosilce, po drugi pa bi proces razvrščanja z uporabo metod podatkovnega rudarjenja postal črna skrinjica, saj bi bilo težje interpretirati morebitne mejne vplive posameznih spremenljivk, ki bi napovedovale verjetnost posameznikove neizpolnitve obveznosti. Katere metode so primernejše za ocenjevanje kreditnega tveganja posameznikovega neizpolnjevanja obveznosti je torej odvisno predvsem od tega, ali si banka želi avtomatizirane procese, ali pa daje banka več poudarka na interpretaciji prispevka posameznih karakteristik.

LITERATURA IN VIRI

1. Altman, E.I., Marco, G., & Varetto, F. (1994). Corporate Distress Diagnosis: Comparisons Using Linear Discriminant Analysis and Neural Networks (The Italian Experience). *Journal of Banking and Finance, 18,* 505-534.
2. Angelini, E., Di Tollo, G., & Roli, A. (2008). A neural network approach for credit risk evaluation. *The Quarterly Review of Economics and Finance, 48(*4*),* 733-755.
3. Baesens, B. (2014, julij). *Analytics in a Big Data World: The Essential Guide to Data Science and its Applications*. New Jersey: Wiley.
4. Baesens, B., & Van Gestel, T. (2009). *Credit Risk Management, Basic concepts: financial risk components, rating analysis, models, economic and regulatory capital.* New York: Oxford University Press.
5. Banka za mednarodne poravnave. (2004, junij). *International Convergence of Capital Measurement and Capital Standards*. Basel: Banka za mednarodne poravnave.
6. Barcun, S., & Charttejee, S. (1970). A Nonparametric Approach to Credit Screening. *Journal of the American Statistical Association, 65(*329*),* 150-154.
7. Baesens, B., Stepanova, M., Suykens, J., Van Gestel, T., Vanthienen, J., & Viaene, S. (2003). Benchmarking state-of-the-art classification algorithms for credit scoring. *Journal of the Operational Research Society, 54(*6*),* 627-635.
8. Bessis, J. (2002). *Risk Management in Banking.* New York: Wiley.
9. Burke , R., & Felfernig A. (2008). Constraint – Based Recommender Systems: Technologies and Research Issues. *Proceedings of the 10th International Conference on Electronic Comerce (*1-10*).* Innsbruck, Avstrija.
10. Capon, N. (1982). Credit Scoring Systems: A Critical Analysis. *Journal of Marketing, 46(2),* 82-91.
11. Chandler, G.G., & Ewert, D.C. (1976). *Discrimination on Basis of Sex and the Equal Credit Opportunity Act* (Working Paper No. 8. Purdue University, Credit Research Center).
12. Chirani, E., Nashtaei, R.A., & Takyar, S.M.T. (2015). The Comparison of Credit Risk Between Artificial Neural Network and Logistic Regression Models in Tose-Taavon Bank in Guilan. *International Journal of Applied Operational Research, 5(*1*),* 63-72.
13. Churchill, G.A., Nevin, J.R., & Watson, R.R. (1977). The role of credit scoring in the loan decision. *Credit World, March,* 6-10.
14. Coats, P.K., & Fant, L.F. (1993). Recognizing Financial Distress Patterns Using a Neural Network Tool. *Financial Management, 22(*3*),* 142-155.
15. Crook, J. N., Edelman, D. B., & Thomas, L. C. (2002). *Credit Scoring and Its Applications (*4th ed.*).* Philadelphia: SIAM.
16. Donko, D., & Dzelihodžić, A. (2013). Data Mining Techniques for Credit Risk Assessment Task. *Zbornik mednarodne konference Recent Advances in Computer Science and Applications (str. 105-110).* Valencia, Španija.
17. Durand, D. (1941). *Risk Elements in Consumer Instalment Financing*. New York. NBER.
18. Forgy, E.W., & Myers, J.H. (1963). The Development of Numerical Credit Evaluation Systems. *Journal of the American Statistical Association, 58(*303*),* 799-806.
19. Fisher, R.A. (1936). The use of multiple measurements in taxonomic problems. *Ann. Eugenics, 7,* 179-188.
20. Frank, E., Hall, M. A., & Witten, I. H. (2011). *Data Mining: Practical Machine Learning Tools and Techniques (*3rd ed.*).* New York: Morgan Kaufmann.

21. Goncalves, E.B., & Gouvea, M.A. (2007). Credit Risk Analysis Applying Logistic Regression, Neural Networks and Genetic Algorithms Models. *Zbornik XVIII. letne konference POMS.* Dallas, ZDA.
22. Hullet, C.R. & Levine, T.R. (2002). Eta Squared, Partial Eta Squared, and Misreporting of Effect Size in Communication Research. *Human Communication Research, 28(*4*),* 612-625
23. Kaggle podatkovna baza, *Give Me Some Credit* (b.l.). Najdeno 18.8.2016 na spletni strani https://www.kaggle.com/c/GiveMeSomeCredit/data
24. Ledolter, J. (2013). *Data Mining and Business Analytics with R*. London: Wiley.
25. Lewis, E.M. (1992). *An Introduction to Credit Scoring*. San Rafael: Athena Press.
26. Makowski, P. (1985). Credit scoring branches out. *Credit World, 75,* 30-37.
27. Malhotra, D.K., Malhotra, R., & McLeod, R.W. (1994). Artificial Neural Systems in Commercial Lending. *The Bankers Magazine,* 40-44.
28. Saunders, J. (1985). This is credit scoring. *Credit management, september*, 23-26.
29. Wiginton, J.C. (1980). A note on the comparison of logit and discriminant models of consumer credit behaviour. *Journal of Finance and Quantitative Analysis, 15(*3*),* 757-770.
30. Wonderlic, E.F. (1952). An analysis of factors in granting credit. *Indiana Univ. Bull., 50,* 163-176.

PRILOGE

KAZALO PRILOG

Priloga 1: Programska koda za uporabo diskriminantne analize.....1
Priloga 2: Programska koda za uporabo logistične regresije.....1
Priloga 3: Programska koda za uporabo metode k-najbljižjih sosedov.....2
Priloga 4: Programska koda za uporabo odločitvenih dreves.....2
Priloga 5: Programska koda za uporabo genetskega algoritma.....3
Priloga 6: Programska koda za uporabo nevronskih mrež.....5
Priloga 7: Programska koda za uporabo metode podpornih vektorjev.....6

PRILOGA 1: Programska koda za uporabo diskriminantne analize

```
load("TrainingData.Rda")
library(MASS)
attach(TrainingData)

TrainSet <- TrainingData[1:105000,]
TestSet <- TrainingData[105001:150000,]

discriminant <- lda(Delinquent~.,TrainSet)
discriminant

#Testiranje modela
TestPred <- predict(discriminant,newdata = TestSet)

ConfusionMatrix <- table(TestPred$class,TestSet$Delinquent)
Accuracy <- sum(diag(ConfusionMatrix))/sum(ConfusionMatrix)
Accuracy
```

PRILOGA 2: Programska koda za uporabo logistične regresije

```
load("TrainingData.Rda")
velikost <- seq(500, 10000, by = 500)
k <- length(velikost)
Accuracy <- rep(0,k)
for(i in 1:k){
vzorec <- TrainingData[sample(nrow(TrainingData),velikost[i]),]
m <- velikost[i]
n <- 0.7*m
TrainSet <- TrainingData[1:n,]
TestSet <- TrainingData[(n+1):m,]
logit <- glm(Delinquent~., data = TrainSet, family = "binomial")

pred <- predict(logit,newdata = TestSet,type = "response")
pred[pred > 0.5] <- 1
pred[pred <= 0.5] <- 0

ConfMatrix <- table(TestSet$Delinquent, pred)
Accuracy[i] <- sum(diag(ConfMatrix))/sum(ConfMatrix)
}
plot(velikost,Accuracy,type = "b", ylab = "Natancnost")
```

PRILOGA 3: Programska koda za uporabo metode k-najbljižjih sosedov

```
library(class)
load("TrainingData.Rda")

velikost <- seq(500, 10000, by = 500)
k <- length(velikost)
Accuracy <- rep(0,k)

for(i in 1:k){
        print(i)
        vzorec <- TrainingData[sample(nrow(TrainingData),velikost[i]),]
        attach(vzorec)
        m <- velikost[i]
        n <- 0.7*m
        Data <- subset(vzorec, select = -Delinquent)
        TrainSet <- Data[1:n, ]
        TestSet <- Data[(n+1):m, ]

        Delinquent <- subset(vzorec, select = Delinquent)
        TrainLabel <- Delinquent[1:n, 1]
        TestLabel <- Delinquent[(n+1):m, 1]


        predikat <- knn(train = TrainSet, test = TestSet, cl = TrainLabel, k = 20)
        Accuracy[i] <- mean(predikat == TestLabel)
}

plot(velikost,Accuracy, type = "b", ylab = "Natancnost")
Accuracy
```

PRILOGA 4: Programska koda za uporabo odločitvenih dreves

```
library(rpart)

load("TrainingData.Rda")

velikost <- seq(500, 10000, by = 500)
k <- length(velikost)
Accuracy <- rep(0,k)

for (i in 1:k){
```

```
	vzorec <- TrainingData[sample(nrow(TrainingData),velikost[i]),]
	m <- velikost[i]
	n <- 0.7*m
	TrainSet <- TrainingData[1:n,]
	TestSet <- TrainingData[(n+1):m,]

	DecisionTree <- rpart(Delinquent~.,data = TrainSet, method ="class",control = rpart.control(minsplit = 180 , cp = 0.001))

	DecisionPred <- predict(DecisionTree,newdata = TestSet, method = "class")

	Predikcija <- numeric(length(TestSet$Delinquent))
	all <- length(Predikcija)

	for (j in 1:all){
		if (DecisionPred[j,][1] > 0.5){
			Predikcija[j] <- 0
		}
		else{
			Predikcija[j] <- 1
		}
	}

	pred.table <- table(Predikcija,TestSet$Delinquent)
	Accuracy[i] <- sum(diag(pred.table))/sum(pred.table)
}
plot(velikost,Accuracy, type = "b", ylab = "Natanenost")
Accuracy
```

PRILOGA 5: Programska koda za uporabo genetskega algoritma

```
library(GA)

load("TrainingData.Rda")
vzorec <- TrainingData[sample(nrow(TrainingData),7000),]
attach(vzorec)
x <- subset(vzorec, select = -Delinquent)
y <- subset(vzorec,select = Delinquent)
TrainX <- x[1:4900,]
TrainY <- y[1:4900,]
TestX <- x[4901:7000,]
TestY <- y[4901:7000,]
```

```
#Najprej izvedemo logistieno regresijo
MLE <- glm(Delinquent~.,data = vzorec,family="binomial")
se.coef <- sqrt(diag(vcov(MLE)))
min <- coef(MLE) - 3*se.coef
max <- coef(MLE) + 3*se.coef

estimator <- function(beta){
	x <- model.matrix(MLE)
	y <- model.response(model.frame(MLE))
	sum(t(x)%*%(y-exp(x%*%beta)/(1+exp(x%*%beta))))
}

Mutacija <- c(.1,.15,.2)
Populacija <- c(100,200,300)
Accuracy <- matrix(ncol=3,nrow=3,data=rep(0,9))
for(i in 1:3){
	for(j in 1:3){
		GAmle <- ga(type = "real-valued",fitness = estimator,min = min, max =
max, popSize=Populacija[i],pmutation= Mutacija[j],maxiter = 5000,run = 200)
		#summary(GAmle)

#Testiranje rezultatov
		beta <- GAmle@solution

		probability <- function(estimate){
			X <- as.matrix(TestX)
			x <- cbind(rep(1,length(X)/10),X)
			exp(x%*%estimate)/(1+exp(x%*%estimate))
		}


		Pred <- probability(t(beta))
		k <- length(Pred)
		Predikcija <- numeric(k)

		for (l in 1:k){
			if (Pred[l] > 0.5){
				Predikcija[l] <- 1
			}
			else{
				Predikcija[l] <- 0
			}
```

```
                }
                pred.table <- table(Predikcija,TestY)

                Accuracy[i,j] <- sum(diag(pred.table))/sum(pred.table)
        }
}

Accuracy
```

PRILOGA 6: Programska koda za uporabo nevronskih mrež

```
set.seed(500)
library(MASS)

load("TrainingData.Rda")

TrainingData$Delinquent <- as.numeric(TrainingData$Delinquent)

velikost <- seq(500,10000, by = 500)
k <- length(velikost)
Accuracy <- rep(0,k)
for(i in 1:k){
#Data preprocesing, normalizacija podatkov, vzoreenje
        vzorec <- TrainingData[sample(nrow(TrainingData),velikost[i]),]
        maksimi <- apply(vzorec,2,max)
        minimi <- apply(vzorec,2,min)

        scaled <- as.data.frame(scale(TrainingData, center = minimi, scale = maksimi-
minimi))

        l <- 0.7*velikost[i]
        m <- velikost[i]
        train <- scaled[1:l,]
        test <- scaled[(l+1):m,]
        Xtrain <- subset(train,select = -Delinquent)
        Ytrain <- subset(train,select = Delinquent)
        Xtest <-subset(test,select=-Delinquent)
        Ytest <- test$Delinquent

#Nevronske mreže
        library("nnet")
        Nevron <- nnet(Xtrain,Ytrain,size = 10,linout=FALSE,maxiter = 10^5)
```

```
        predikat <- predict(Nevron,Xtest)
        ConfMat <- table(Ytest,predikat)
        Accuracy[i] <- sum(diag(ConfMat))/sum(ConfMat)
}
Accuracy
plot(velikost,Accuracy,type = "b",ylab = "Natancnost")
```

PRILOGA 7: Programska koda za uporabo metode podpornih vektorjev

```
library("e1071")
set.seed(500)
load("TrainingData.Rda")

velikost <- seq(500,10000,by = 500)
k <- length(velikost)
Accuracy <- rep(0,k)
for (i in 1:k){
        m <- 0.7*velikost[i]
        n <- velikost[i]
        vzorec <- TrainingData[sample(nrow(TrainingData),n),]
        x <- subset(vzorec, select = -Delinquent)
        y <- subset(vzorec,select = Delinquent)
        TrainX <- x[1:m,]
        TrainY <- y[1:m,]
        TestX <- x[(m+1):n,]
        TestY <- y[(m+1):n,]

        model_svm <- svm(TrainX,TrainY)
        model_svm

        Predikcija <- as.numeric(predict(model_svm,TestX))
        Matrika <- table(TestY,Predikcija)

        l <- length(Predikcija)
        pred_test <- rep(0,l)
        for (j in 1:l){
                if (Predikcija[[j]] > 0.5){
                        pred_test[j] <- 1
                }
                else{
                        pred_test[j] <- 0
```

```
                }
        }
        ConfusionMatrix <- table(TestY,pred_test)
        Accuracy[i] <- sum(diag(ConfusionMatrix))/sum(ConfusionMatrix)
}
Accuracy

plot(velikost,Accuracy, type = "b", ylab = "Natancnost")
```