Predviđanje količine prometa zasnovano na neuronskoj mreži

Ravnjak, Matea

Master's thesis / Diplomski rad

2015

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Transport and Traffic Sciences / Sveučilište u Zagrebu, Fakultet prometnih znanosti**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:119:749606

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-03-29**



Repository / Repozitorij:

Faculty of Transport and Traffic Sciences - Institutional Repository







SVEUČILIŠTE U ZAGREBU

FAKULTET PROMETNIH ZNANOSTI

Matea Ravnjak

PREDVIĐANJE KOLIČINE PROMETA ZASNOVANO NA NEURONSKOJ MREŽI

DIPLOMSKI RAD

Zagreb, 2015.

Sveučilište u Zagrebu
FAKULTET PROMETNIH ZNANOSTI
Vukelićeva 4, 10000 Zagreb
DIPLOMSKI STUDIJ

Diplomski studij: Promet
Zavod: Zavod za inteligentne transportne sustave
Predmet: Umjetna inteligencija

ZADATAK DIPLOMSKOG RADA

Pristupnik: Matea Ravnjak
Matični broj: 0135217342
Smjer: Informacijsko-komunikacijski promet

ZADATAK:

Predviđanje količine prometa zasnovano na neuronskoj mreži

ENGLESKI NAZIV ZADATKA:

Neural Network based Traffic Prediction

Opis zadatka:

Danas postoji velika potreba za transportom ljudi, dobara i podataka. U tu se svrhu koristi pripadna transportna mreža koje može biti vezana za cestovni, zračni, željeznički, vodni, pomorski i komunikacijski promet. Za optimalno korištenje postojeće transportne infrastrukture sve se više koriste napredne metode upravljanja iz područja umjetne inteligencije. Takve metode zahtijevaju kvalitetne podatke o trenutnom stanju mreže pri čemu se koristi i mogućnost predviđanja količine prometa. U radu je potrebno opisati probleme u predviđanju količine prometa, analizirati mogućnost primjene neuronskih mreža za predviđanje količine prometa, predložiti strukturu neuronske mreže te ju nakon učenja provjeriti na odabranom skupu podataka količine prometa komunikacijske i cestovne mreže.

Zadatak uručen pristupniku:
8. svibnja 2015.

Nadzorni nastavnik:

Predsjednik povjerenstva za diplomski ispit:

Djelovođa:

Sveučilište u Zagrebu

Fakultet prometnih znanosti

DIPLOMSKI RAD

PREDVIĐANJE KOLIČINE PROMETA ZASNOVANO NA NEURONSKOJ MREŽI

NEURAL NETWORK BASED TRAFFIC PREDICTION

Mentor: Doc. dr. sc. Eduard Ivanjko

Student: Matea Ravnjak, 0135217342

Zagreb, 2015

Sažetak

U ovom radu kreirana je neuronska mreža s ciljem predviđanja količine prometa na transportnoj mreži. Pri tome se misli na cestovni transport (broj vozila) te na telekomunikacijski promet, odnosno prijenos podataka. Na osnovu ulaznih varijabli prikupljenih tijekom određenog vremena, predviđa se broj vozila odnosno količina podatkovnog prometa za idući vremenski interval. Korištena je višeslojna struktura neuronske mreže, a rezultati pokazuju da je neuronska mreža sposobna prilikom učenja aproksimirati funkciju ulaznih i izlaznih varijabli uz visoku točnost.

Ključne riječi: umjetna inteligencija; neuronske mreže; predviđanje; transportni sustavi, višeslojna arhitektura, cestovna mreža, telekomunikacijska mreža

Summary

In this paper, neural network is created in order to predict the amount of traffic in the transport networks. This refers to road transport (number of vehicles) and telecommunication traffic, i.e. data transfer. Based on the input variables collected during a certain time, neutral network made predictions for number of vehicles and the amount of data traffic for the next time interval. Multilayer structure neutral network is used, and the results show that the neural network is capable to learn approximate function of input and output variables with high accuracy.

Key words: artificial intelligence; neural networks; prediction; transport systems; multi-layer architecture, road network, telecommunication network

Sadržaj:

1. UVOD 1
2. PROBLEMI U PREDVIĐANJU PROMETA 2
2.1. Karakteristike prometnog toka 2
2.2. Podjela modela kratkoročnih predviđanja 4
2.2.1. Parametarske tehnike 4
2.2.2. Neparametarski modeli 5
3. NEURONSKE MREŽE 7
3.1. Razvoj neuronskih mreža 7
3.2. Definicija i karakteristike neurona 8
3.3. Karakteristike neuronske mreže 10
3.4. Osnovne aktivacijske funkcije neurona 11
3.5. Postupak učenja mreže 13
3.6. Arhitektura i način rada neuronske mreže 16
4. OPIS STRUKTURE NEURONSKE MREŽE 19
4.1. Arhitektura neuronske mreže za cestovne podatke 19
4.2. Opis arhitekture neuronske mreže za telekomunikacijski promet 22
5. OPIS PODATAKA 25
5.1. Procedura učenja neuronskih mreža 25
5.2. Opis podataka za cestovnu mrežu 27
5.3. Opis podataka za telekomunikacijsku mrežu 29
6. REZULTATI 32
6.1. Analiza uspješnosti predviđanja za cestovnu mrežu 32
6.2. Analiza uspješnosti predviđanja za podatkovnu mrežu 35
7. ZAKLJUČAK 37
LITERATURA 38

Popis ilustracija 40

Popis tablica: 41

Prilog 1 42

Prilog 2 44

Prilog 3 47

Prilog 4 53

1. UVOD

U današnjem svijetu, brzih promjena, gdje se inovacije u različitim područjima događaju na dnevnoj bazi, problem čekanja u prometu i dalje ostaje neriješen. Pri tome se misli i na prijevoz ljudi i roba te na prijenos podataka. Novije generacije upravljačkih sustava oslanjaju se na tehnologiju koja im omogućava stvarnovremenski uvid u stanje sustava. Na osnovu tog uvida takvi sustavi podešavaju mrežu kako bi se osigurao optimalan protok. Dodatno se takvi upravljački sustavi mogu unaprijediti korištenjem predviđanja količine prometa. Jedno od mogućih tehnologija koje se mogu iskoristiti za predviđanje prometa jesu neuronske mreže. Upravo razvoj tehnologije i primjena neuronskih mreža za predviđanje količine prometa nude mogućnost rješenja problema prometnih zagušenja.

Neuronska mreža rastuća je paradigma sve popularnije discipline: umjetne inteligencije. Umjetna inteligencija implementira inteligentne osobine ljudskog ponašanja. Na primjeru neuronskih mreža, korištena je pojednostavljena struktura ljudskog mozga u svrhu procesiranja podataka. Neuronske mreže u sustavu transportnih mreža najviše se primjenjuje kako bi se na optimalan način iskoristila postojeća infrastruktura. Transportne mreže, pokazuju slično ponašanje bez obzira o kojoj vrsti prometa se radi. Tako primjerice, i cestovna i podatkovna mreža imaju manji promet preko noći, a porast prometa se dešava ujutro dok putnici odlaze na posao ili provjeravaju jutarnje novosti, prognozu za taj dan, preuzimaju elektroničku poštu ili tomu slično.

U ovom diplomskom radu obrađeno je kratkoročno predviđanje količine prometa na primjeru cestovne i komunikacijske mreže. Predviđanje je obavljeno pomoću neuronskih mreža i njihovog svojstva učenja. Na osnovu ulaznih podataka mreža je trenirana kako bi dala predviđanja, odnosno izlazne podatke. U drugom poglavlju opisuje se problematika predviđanja prometa, te nekoliko dosad korištenih metoda. Treće poglavlje sadrži analizu rada neuronskih mreža, njihov razvoj i strukturu. Četvrto poglavlje daje opis strukture neuronske mreže za predviđanje, dok peto poglavlje daje uvid u podatkovni skup korišten za učenje neuronske mreže. U šestom poglavlju, obrađeni su rezultati predviđanja. Diplomski rad završava zaključkom i prijedlozima za budući raad.

2. PROBLEMI U PREDVIĐANJU PROMETA

Kratkoročno predviđanje prometa je dugo smatrano kao kritičan modul za inteligentne transportne sustave. U slučaju upravljanja transportom ljudi i dobara predviđanje prometa je najčešće modul u sklopu upravljačkog centra odnosno unutar inteligentnog transportnog sustava [1]. U slučaju mreže za prijenos podataka modul za predviđanje podatkovnog prometa je dio poslužitelja koji upravljaju telekomunikacijskom mrežom te koji usmjeruju podatkovni promet. Kratkoročno predviđanje prometa u bliskog budućnosti je esencijalno za dinamičko (stvarnovremensko) upravljanje prometom i sustave navigacije radi smanjenja zagušenja. Međutim, složene prometne situacije, velikih prometnih mreža i karakteristike prometnih tokova dovode do teškoća pri dobivanju jasnih i točnih podataka kratkoročnih predviđanja svih perioda tijekom dana [2].

Postoje razni modeli za predviđanje te svaki od njih ima dobar učinak u određenom periodu. Dosadašnja istraživanja pokazala su da nijedan od jednostrukih prediktora nije razvijen dovoljno kako bi bio opće prihvaćen kao najbolji model efikasnog predviđanja za stvarnovremenske prometne aktivnosti. Stoga se nameće potreba za poboljšanim pristupom, gdje će se kombinirati ti jednostruki prediktori zajedno u više perioda. U ovom diplomskom radu, model neuronske mreže je predstavljen kao model koji objedinjuje predviđanje od jednostavne neuronske mreže do adaptivnih i heurističkih zadanih algoritama zasnovanih na teoriji uvjetnih vjerojatnosti i Bayesovog pravila [1].

2.1. Karakteristike prometnog toka

Prometni tokovi su rezultat individualnih vozača i njihovih vozila čija ruta na prometnoj mreži utječe na karakteristike prometnog toka. Njihov utjecaj je složen zbog velikog raspona promjenjivosti ponašanja vozača i promjenjivosti karakteristika samih vozila [3]. U komunikacijskom prometu, nastanak komutirane paketne mreže, kao i transformacija telefonskih mreža u višeuslužne sustave, pruža korisniku nove mogućnosti korištenja mreže. Višeuslužni sustavi promijenili su arhitekturu mreže, kao i statičku prirodu telekomunikacijskog prometa koji

je sada karakterističan po efektima samo-sličnosti i dugogodišnje ovisnosti [4]. Mrežna topologija postepeno postaje sve složenija te je sve više mrežnih zagušenja i neočekivanih situacija. Zbog toga, mrežna izvedba treba promatranje i predviđanje. Predviđanje mrežnog prometa može spriječiti i kontrolirati zagušenje mreže, poboljšati iskoristivost mreže [5]. Osim toga, predviđanje prometa predstavlja važnu ulogu u dizajnu, menadžmentu i optimizaciji modernih telekomunikacijskih sustava. Točne i pouzdane prognoze omogućavaju planiranje kapaciteta mreže i održavanje zahtijevane kvalitete usluge [4].

Korisne informacije o ponašanju prometnog toka mogu se dobiti promatranjem, odnosno mjerenjem i analizom karakteristika prometnog toka. Parametri koji opisuju karakteristike toka cestovne prometne mreže mogu se svrstati u dvije grupe:

- Makroskopski parametri (opisuju prometni tok u cijelosti);
- Mikroskopski parametri (opisuju ponašanje pojedinih vozila ili grupe vozila) [3].

Tri osnovna makroskopska parametra su: protok, srednja brzina i gustoća vozila. Temeljni mikroskopski parametri su: brzina pojedinačnoga vozila, praznina i razmak. Osim ovih parametara, često se promatra i vrijeme putovanja određenom rutom na prometnoj mreži.

Prometni tok može biti:

- slobodan protok (eng. *free flow*);
- maksimalni protok (eng. *capacity-flow*);
- zagušen tok (eng.*congested*);
- kreni-stani (eng. *stop-and-go*);
- zakrčen (eng. *jammed*).

Klasifikacija prometnog toka zasniva se na vrijednostima karakteristika prometnog toka i njihovih međusobnih utjecaja.

2.2. Podjela modela kratkoročnih predviđanja

Kratkoročnim predviđanjem transportnih sustava smatra se svaka procjena parametara prometnog toka do dva sata u budućnost. Modeli se koncentriraju na različite pristupe kako bi predvidjeli karakteristike prometnog toka. Pristupi predviđanju mogu se grupirati u tri kategorije:

- parametarske tehnike;
- neparametarske metode;
- naivne metode.

Parametarski modeli uključuju vremensko-serijske modele, model Kalmanovog filtriranja, i slično. Neparametarski modeli uključuju metodu najbližeg susjeda (eng. *k-nearest neighbor*), umjetnu neuronsku mrežu. Naivne metode nemaju nikakvo znanje o modelu i najčešće se koriste metoda povijesnog uprosječivanja (eng. *historical averages*) i metoda grupiranja (eng. *clustering*) [3] [6].

2.2.1. Parametarske tehnike

Obilježja parametarskih modela su predefinirana struktura i konačan skup parametara. Znanje o procesu koji se modulira može se ugraditi u strukturu modela i time poboljšati točnost predviđanja. Skup podataka je manji, usporedivši s neparametarskim metodama. Na osnovu tog skupa podataka utvrđuju se vrijednosti parametara. Prema području istraživanja modele parametarskih tehnika dijelimo na:

- simulaciju prometa (eng. *traffic simulation models*);
- vremenske serije (eng. *time series*).

Modele simulacije prometa možemo podijeliti prema:

- razini detalja na : makroskopske, mezoskopske i mikroskopske;
- skali nezavisnih varijabli na: kontinuirane, diskretne i polu-diskretne;

• predstavljanju procesa na deterministične i nedeterministične.

Modeli koji se zasnivaju na vremenskim serijama predviđaju vrijednost varijable na temelju povijesnih mjerenja. Modeliranje vremenskim serijama zahtijeva da je proces stacionaran. Taj uvjet prometni tok ne zadovoljava. Zbog toga je uveden pojam sezonalnosti (eng. *seasonality*). Umjesto teorije o prometnom toku korištena je statika. Modeli koji koriste vremenske serije za predviđanje su:

• linearna regresija (eng. *linear regression*);

• ARIMA (eng. *autoregressive integrated moving average*);

• Kalmanov filtar [3].

Tehnika koja je široko primijenjena za problem predviđanja prometa zasnovana je na metodi analize vremenskih serija. ARIMA model se smatra najznačajnijim statističkim modelom za predviđanje. Mnoge varijante ARIMA modela su predlagane kako bi se povećala točnost predviđanja. Stoga, u literaturi često se nalazi sezonska ARIMA (SARIMA), ARIMAX i STARMA [6].

2.2.2. Neparametarski modeli

Neparametarski modeli odlikuju time što su im potrebni podaci za određivanje infrastrukture i za određivanje vrijednosti parametara. Broj i definicija parametara u neparametarskom modelu nije unaprijed određena. Uspoređujući ih s parametarskim modelima, njihov glavni nedostatak je potrebna količina podataka. Neparametarskim modelima je potrebno mnogo više podataka. Prednost nad parametarskim modelima se krije u mogućnosti modeliranja dinamičnih i nestacionarnih procesa, poput prometnog toka. Rijetki i specifični događaji (poput nastanka incidenta) mogu prouzročiti grešku u predviđanju prometnog toka. Metode neparametarskih modela su:

• k-najbližih susjeda (eng. k-*Nearest Neighbour*);

• lokalna regresija s težinskim faktorom (eng. *locally weighted regression*);

- neizrazita logika (eng. *fuzzy logic*);
- Bayesove mreže (eng. *Bayesian networks*);
- neuronske mreže (eng. *Neural networks*).

Metoda k-najbližih susjeda zasnovana je na ideji pretraživanja povijesne baze podataka. Traži se k događaja najbliži trenutnome događaju. Rezultati se uprosječuju, a zatim se dodaju težine pojedinim događajima ovisno o udaljenosti od trenutnog događaja. Metoda neizrazite logike koristi skup ako-onda pravila (eng. *if-then rules*). Pravila zatim služe za predviđanje prometnog toka [3]. Bayesove mreže predstavljene su usmjerenim matematičkim grafom u kojem si prikazane uvjetne ovisnosti skupa slučajnih varijabli. U predviđanju određenog prometnog toka, koriste se podaci sa susjednih prometnica [7]. Neuronske mreže su matematički model inspiriran strukturom bioloških neuronskih mreža.

3. NEURONSKE MREŽE

Ljudski mozak sastoji od velikog broja živčanih stanica (neurona), koji pri obradi različitih vrsta informacija rade paralelno. Neurofiziološka istraživanja, koja su nam omogućila bolje razumijevanje strukture mozga, pa čak i kognitivna psihologija daju naslutiti da je modelu mozga najsličniji model u kojem brojni procesni elementi podatke obrađuju paralelno. Područje računarstva koje se bavi tim aspektom obrade informacija zovemo neuronsko računarstvo, a samu paradigmu obrade podatka umjetnom neuronskom mrežom (eng. *Artificial Neural Network,* ANN) [8]. Na konferenciji Dartmouth Summer Research Project on Artificial Intelligence, pedesetih godina ovog stoljeća, najavljeno je ostvarivanje vizije računalnog modela koji će u nekim temeljnim principima oponašati funkcioniranje mozga. Neuronsko računalstvo se pojavljuje kao alternativa strukturi Von Neumann-ovih računalima i nastoji simulirati paralelnu obradu informacija koju koristi ljudski mozak dok razmišlja, sjeća se ili rješava probleme [9].

3.1. Razvoj neuronskih mreža

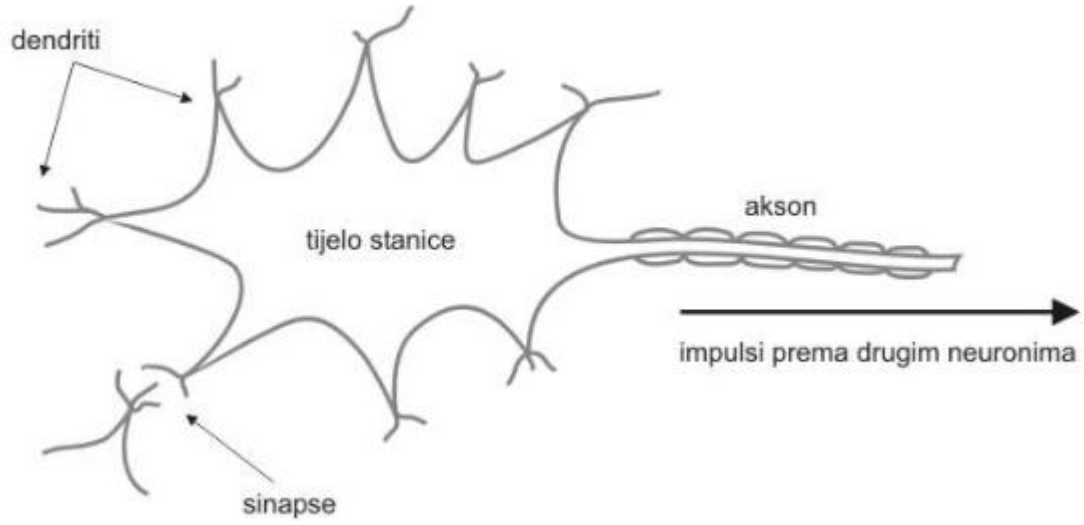
Ideja potiče još iz 1940. g, kada McCulloch i Pitts (Massachusetts Institute of Technology), istražujući neurofiziološke karakteristike bića, objavili matematički model neuronske mreže u okviru teorije automata. Dokazali su da atomi mogu imati 2 stanja (pobuđujuće i umirujuće) i da njihova aktivnost ovisi o nekom pragu vrijednosti. Međutim, procesna moć računala u to vrijeme nije još bila dorasla implementaciji umjetne neuronske mreže. Tek u kasnim pedesetima, pojavom snažnijih računala, dogodila su se i prva praktička ostvarenja. Idućih dvadesetak godina područje istraživanja neuronskih mreža gotovo da je bilo zaustavljeno. Umjetne neuronske mreže vraćaju se na scenu umjetne inteligencije 1990., kad je omogućeno aproksimiranje gotovo svih funkcija i rješavanje praktičnih problema. Danas su postale vodeći i gotovo nezaobilazan koncept pri razvoju inteligentnih sustava. Razvijeni su brojni algoritmi za učenje neuronske mreže, od kojih najveću komercijalnu upotrebu do danas ima algoritam "širenje unatrag" (eng. *Backpropagation*).

Dakle, teorijsko ishodište i inspiracija neuronskog računalstva je ljudski mozak [8] [9]. Premda su umjetne neuronske mreže po svojim sposobnostima obrade podataka daleko ispod mogućnosti obrade ljudskog mozga, pokazuju sposobnosti:

- učenja na temelju iskustva,
- poopćavanja,
- odjeljivanja bitnog od nebitnog,
- prepoznavanja uzoraka [10].

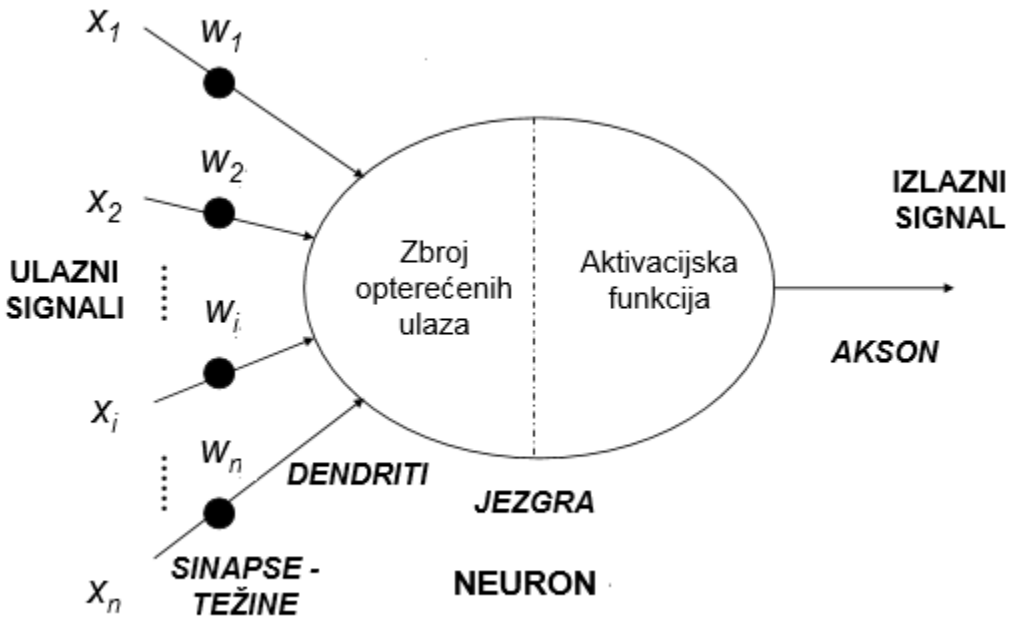
3.2. Definicija i karakteristike neurona

Ljudski mozak sastavljen je od oko 10^{11} neurona s više od 100 vrsta i koji su shodno svojoj funkciji raspoređeni prema definiranom rasporedu. U prosjeku, svaki je neuron povezan sa 104 drugih neurona. Neuron se sastoji od: tijela stanice (soma), skupa dendrita (ogranaka), aksona (dugačke cjevčice koje prenose električke poruke) i niza završnih članaka. Na idućoj slici prikazana je građa neurona.



Slika 1: Izgled biološkog neurona, [8]

Umjetni neuron ima više ulaza putem kojih prima informacije. Ulazi su analogni dendritima kod živih neurona. Ulazi se zbrajaju pomoću neke zbrojne funkcije i tako neuron stvara svoju internu aktivaciju. Zatim se, pomoću funkcije prijenosa mijenja taj zbrojni ulaz. Funkcija prijenosa može biti diskontinuirana funkcija skoka, ili kontinuirana funkcija, kao npr. sigmoida ili tangens-hiperbolna funkcija [9].



Slika 2: Umjetni neuron, [10]

Ulazne signale, njih ukupno n, označavamo koristeći x_1, x_2, x_3,... x_n. Ulazni signali su veličine kontinuiranih vrijednosti. Težine označavamo sa ω_1, ω_2, ω_3... ω_n. Težine su prilagodljive veličine pozitivnih ili negativnih vrijednosti. One pojačano (ekscitacijsko) ili oslabljeno (inhibicijsko) djeluju na ulazni signal. Jezgra predstavlja zbrajanje opterećenih ulaznih signala. U njoj se vrši aktivacijska funkcija (prijenosna funkcija) neurona. Suma težinske net vrijednosti računa se:

$$net = \omega_1 x_1 + \omega_2 x_2 + \ldots\ldots + \omega_n x_n - \theta \,. \qquad (1)$$

Dogovorno se uzima da je vrijednost praga $\theta = -\omega_0$ te se dodaje ulazni signal x_1 s fiksiranom vrijednošću 1, pa pišemo jednostavnije:

$$net = \omega_0 x_0 + \omega_1 x_1 + \omega_2 x_2 + \ldots\ldots + \omega_n x_n = \sum_{i=0}^{n} \omega_i x_i \,. \quad (2)$$

Izlaz y je rezultat prijenosne funkcije:

$$net = f(\sum_{i=0}^{n} \omega_i x_i) = f\,(net)\,. \quad (3)$$

3.3. Karakteristike neuronske mreže

Umjetne neuronske mreže su međusobno povezani procesni elementi čije je ponašanje slično ponašanju biološkog neurona. Drugim riječima, to je umjetna replika ljudskog mozga sa svrhom simuliranog postupka učenja. Učinjena su mnoga pojednostavljena umjetne neuronske mreže u odnosu na neurone biološkog sustava. Postoje još mnogi fenomeni živčanog sustava koji nisu modelirani umjetnim neuronskim mrežama, kao što postoje i karakteristike umjetnih neuronskih mreža koje se ne slažu s onima bioloških sustava. Neuronska mreža je mreža međusobno povezanih jednostavnih procesnih elemenata, jedinica ili čvorova, čije ponašanje sliči biološkom neuronu. Pri tome je moć mreže da vrši obradu podataka pohranjena u snazi veza između pojedinih neurona tj. težinama. Do težina se dolazi postupkom prilagodbe odnosno učenjem iz skupa podataka za učenje. Neuronska mreža obrađuje podatke distribuiranim paralelnim radom. Osobitosti neuronskih mreža u odnosu na konvencionalne načine obrade podataka su sljedeće:

- Vrlo dobra procjena nelinearnih odnosa uzoraka;
- Mogućnost rada s nejasnim (manjkavim) podacima tipičnim za podatke iz različitih senzora, poput kamera i mikrofona, i u njima raspoznavati uzorke;
- Robusnost na pogreške u podacima, za razliku od konvencionalnih metoda koje pretpostavljaju normalnu raspodjelu obilježja u ulaznim podacima;

- Neurobiološka analogija;
- Mogućnost rada s velikim brojem varijabli ili parametara;
- Prilagodljivost okolini;
- Mogućnost jednostavne implementacije u sklopovlju za paralelnu obradu podataka;
- Sposobnost formacije znanja učeći iz iskustva (primjera);
- Tolerantnost na greške (eng. fault tolerance) [8] [11].

Osim toga, neuronske mreže rješavaju probleme klasifikacije i predviđanja, odnosno općenito sve probleme kod kojih postoji odnos između prediktorskih (ulaznih) i zavisnih (izlaznih) varijabli, bez obzira na visoku složenost te veze (nelinearnost). Danas se neuronske mreže primjenjuju u segmentima poput medicine, bankarstva, strojarstva, geologije, fizike itd., najčešće za zadatke poput:

- raspoznavanje uzoraka;
- obrada slike;
- obrada govora;
- problemi optimizacije,
- nelinearno upravljanje;
- obrada nepreciznih i nekompletnih podataka;
- simulacije [8].

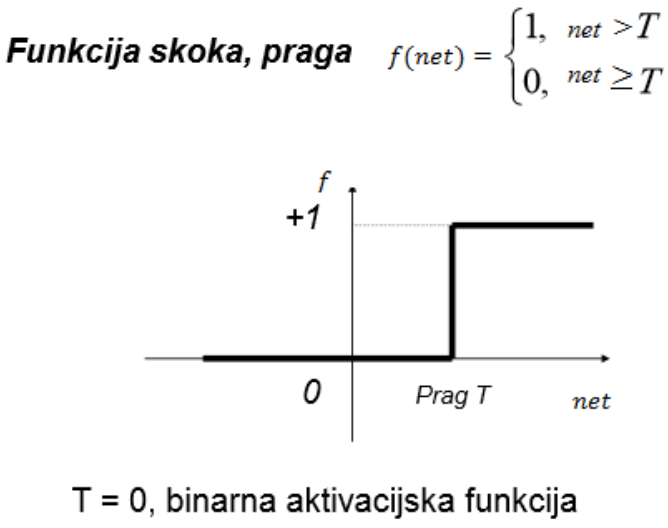
3.4. Osnovne aktivacijske funkcije neurona

Općeniti model umjetnog neurona prikazan na slici 2, u literaturi je dalje razmatran prema ugrađenoj prijenosnoj funkciji. Najjednostavnija moguća aktivacijska funkcija je:

$$f(net) = net. \tag{4}$$

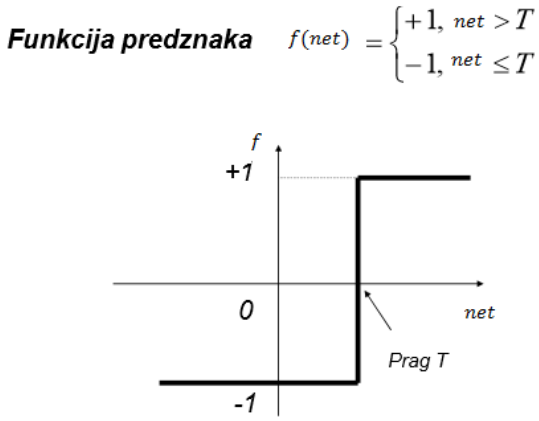
Ta funkcija svojstvena je modelu umjetnog neurona ADALINE (eng. *Adaptive Linear Element*). Izlaz iz takve jedinice je težinska suma njegovih ulaza. Druga mogućnost je korištenje

funkcije skoka ili praga (eng. *threshold function, hard-limiter*). Tom aktivacijskom funkcijom dobije se procesna jedinica koja daje Booleov izlaz (eng. *Threshold Logic Unit*, TLU):



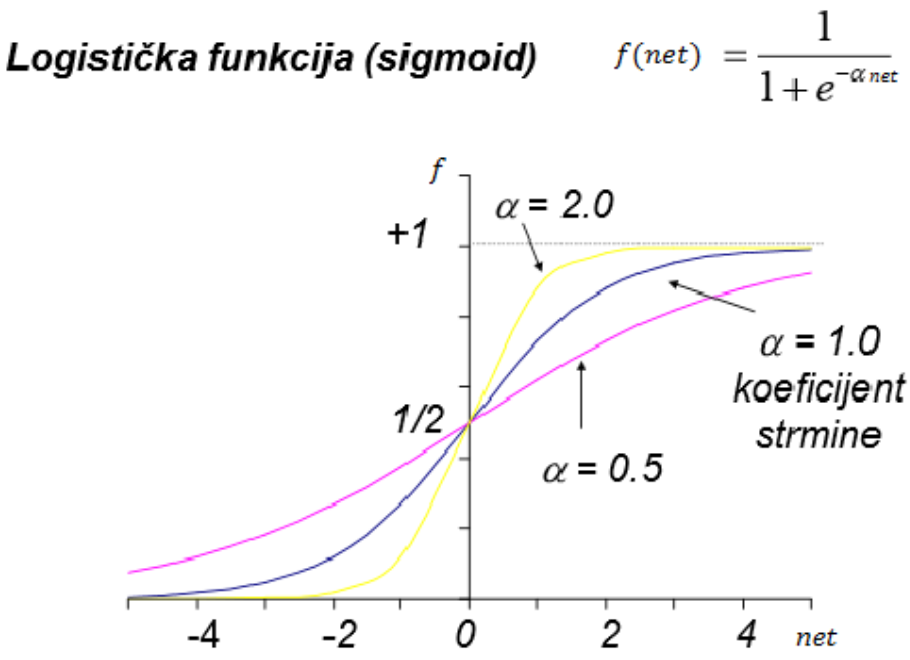
Slika 3: Prikaz funkcije skoka, izvor: [10]

Prijenosna funkcija može biti definirana i po dijelovima linearno. Ova funkcija zove se funkcija predznaka.



Slika 4: Prikaz funkcije predznaka, izvor: [10]

Najčešći oblik prijenosne funkcije jest logistička, odnosno sigmoidalna funkcija. Za razliku od prethodnih funkcija, ova je funkcija derivabilna. To je bitna prednost pri postupku učenja umjetne neuronske mreže. Sigmoidalna funkcija prikazana je na idućoj slici.



Slika 5: Logistička funkcija, [10]

Umjetne neurone možemo razvrstati i prema vrsti signala koje prosljeđuju (realni brojevi ili Booleove vrijednosti) te prema obliku integrirajuće funkcije (razmatranje težinske sume).

3.5. Postupak učenja mreže

Jednostavnije neuronske mreže moguće je izgraditi tako da obavljaju određeni zadatak. Ovo će biti moguće za mreže koje se sastoje od TLU perceptrona, i koje obavljaju unaprijed zadanu logičku funkciju, jer u tom slučaju može se nadzirati što i kako točno mreža radi. U slučaju korištenja složenih aktivacijskih funkcija poput logističke, ili uporaba realnih brojeva, gubi se mogućnost kontroliranja rada mreže. Takav slučaj zahtijeva prethodnu definiciju arhitekture mreže, a prije obrade podataka obavi se učenje ili treniranje mreže. Pod postupkom učenja

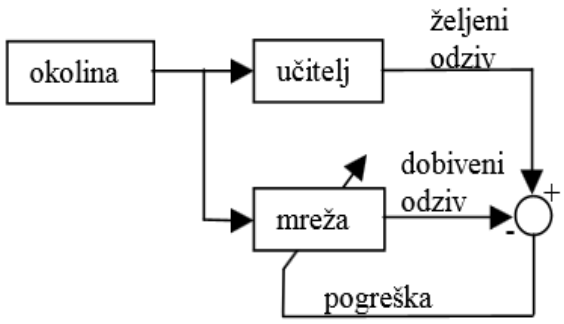
neuronskih mreža podrazumijeva se iterativan postupak predočavanja ulaznih primjera (uzoraka, iskustva) i eventualno očekivana izlaza.

U odnosu na ostale konvencionalne metode koje zahtijevaju da se obrada podataka analitički razloži na određeni broj algoritamskih koraka, kod složenijih funkcija takav algoritma ne postoji. Znanje o izlazu pohranjeno je u težinama veza između neurona. Težine se postupno prilagođavaju u postupku učenja mreže sve do trenutka kada je izlaz iz mreže zadovoljavajući. Učenje je postupak prilagođenja težina spojnica umjetne neuronske mreže s ciljem izračunavanja izlaznog vektora kao odziva na poticajni ulazni vektor.

Ovisno o tome da li nam je u postupku učenja poznat izlaz iz mreže, pa ga pri učenju mreže koristimo uz svaki ulazni primjer, ili nam je točan izlaz nepoznat, postoje idući načini učenja:

- Nadzirano učenje (eng. *supervised learning*) - učenje uspoređivanjem ciljnog i izračunatog izlaznog vektora na temelju pokazivanja niza parova ulazno izlaznih vektora od strane "učitelja";
- Nenadzirano, samouko učenje (eng. *unsupervised learning*) - izračunavanje odziva mreže se temelji na sposobnosti samoustrojstva mreže na nizove ulaznih vektora tvori se unutarnji prikaz značajki skupina ulaznih vektora bez "učitelja";
- Učenje stupnjevanjem dobrote naučenog- učenje procjenom sličnosti izlaznog vektora u odnosu na ulazni vektor , a pri tome "učitelj" ne pokazuje ciljni vektor [8][10].

Glavna karakteristika učenja pod nadzorom je prisutnost učitelja. Vanjski učitelj ima znanje o okolini u obliku parova ulaz-izlaz. Razlika između željenog i dobivenog odziva za neki ulazni vektor smatra se pogreškom. Parametri mreže mijenjaju se utjecajem ulaznih vektora i signala pogreške. Proces se iteracijski ponavlja sve dok mreža ne nauči imitirati učitelja. Nakon što je učenje završilo učitelj više nije potreban i mreža može raditi samostalno. Iduća slika opisuje proces učenja pod nadzorom.

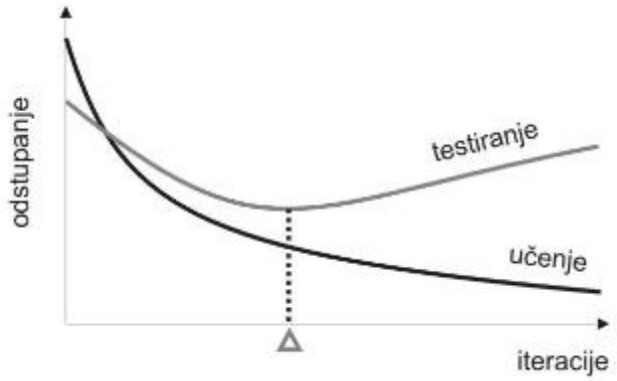


Slika 6: Učenje pod nadzorom, [11]

Primjeri algoritama za učenje pod nadzorom su: algoritam najmanjih kvadrata (engl. Least-Mean-Square) i algoritam s povratnom propagacijom pogreške (engl. Back-Propagation). Mana učenja pod nadzorom je ograničenost učenja, odnosno mreža bez učitelja ne može naučiti nove strategije koje nisu pokrivene primjerima koji su korišteni za učenje [11]. Skup primjera za učenje često se dijeli na:

- skup za učenje;
- skup za testiranje;
- skup za provjeru (validaciju).

Primjeri iz prvog skupa služe za učenje u užem smislu (podešavanje težinskih faktora). Uzorci iz drugog skupa koriste se za provjeru rada mreže s trenutnim težinskim faktorima tijekom učenja. U slučaju degradacije performanse mreže postupak učenja se zaustavlja. Točnost i preciznost obrade podataka moguće je provjeriti koristeći treći skup primjera – skupom za provjeru. Sljedeća slika opisuje primjer odnosa dobivenog i željenoga izlaza.



Slika 7: Odnos željenog i dobivenog rezultata, [8]

Uz pojam učenja umjetne neuronske mreže vezani su pojmovi iteracija i epoha. Iteracija je korak u algoritmu postupka za učenje u kojem se odvija podešavanje težinskih faktora, dok je epoha jedno predstavljanje cjelokupnog skupa za učenje. Ovisno o broju primjera predočenih mreži za trajanje jedne iteracije, postoji:

- pojedinačno učenje (eng. *on-line training*)– u jednoj iteraciji predočava se samo jedan primjer za učenje (tj. kod svakog primjera za učenje vrši se prilagodba težinskih faktora);
- grupno učenje (engl. *batch training*) –u jednoj iteraciji predočavaju se svi primjeri za učenje (tj. iteracije se podudaraju s epohama) [8].

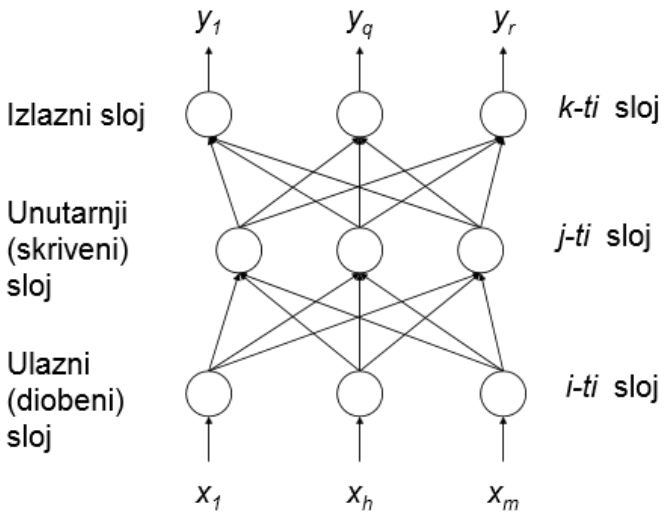
3.6. Arhitektura i način rada neuronske mreže

Arhitektura, odnosno topologija mreže određuje način na koji su neuroni međusobno povezani. Postoje 4 vrste neuronskih mreža:

- Jednoslojne mreže bez povratnih veza (eng. *single-layer feedforward networks*);
- Višeslojne mreže bez povratnih veza (eng. *multi-layer feedforward networks*);
- Mreže s povratnim vezama (eng. *recurrent networks*);
- Ljestvičaste mreže (eng. *lattice structures*).

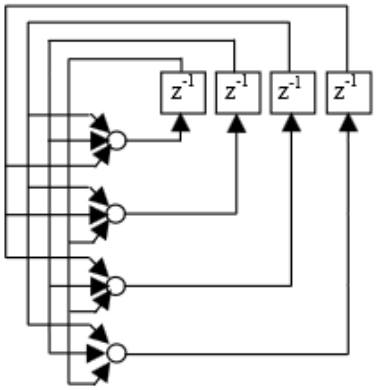
Jednoslojna mreža sastoji se od jednog sloja neurona- izlaznog sloja. U ovom slučaju, Ulazni sloj neurona se ne broji jer u njemu nema procesiranja. Ulazi u mrežu spajaju se na ulaze neurona čiji izlazi predstavljaju i izlaz mreže. Ne postoji povratna veza s izlaza na ulaz. Primjer jednoslojne mreže je linearna asocijativna memorija gdje ulazni vektor generira na izlazu asocijaciju.

Višeslojne mreže, osim ulaznog i izlaznog sloja, imaju i jedan ili više skrivenih slojeva neurona. Izlazi neurona iz n-tog sloja predstavljaju ulaze u neurone iz n+1-og sloja. U višeslojnim mrežama često se koristi pojam povezanosti. Mreža je potpuno povezana u slučaju kada je svaki neuron u svakom sloju povezan na sve neurone u sljedećem sloju. Mreža može biti i djelomično povezana, kada neke veze nedostaju [11]. Na idućoj slici oslikan je primjer mreže sa skrivenim slojem. Iako je u primjeru broj ulaznih neurona i broj neurona skrivenog sloja jednak, to nije nužno pravilo.



Slika 8: Arhitektura višeslojne mreže bez povratnih veza, [10]

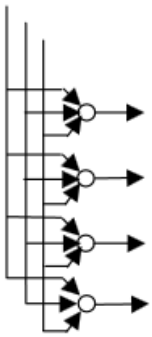
Višeslojne mreže s povratnom mrežom imaju bar jednu povratnu vezu. Mogu imati i skrivene neurone. Prisutnost povratnih veza daje bolju kvalitetu ovim mrežama, ali utječe i na složenost analize rada mreže. Kombinirajući s elementima za kašnjenje dobivamo nelinearne dinamičke sustave što je ključno za sposobnost memoriranja u mreži.



Slika 9: Arhitektura neuronske mreže s povratnom vezom, [11]

Na slici iznad nalazi se primjer mreže s povratnom vezom bez skrivenih neurona gdje svaki neuron dobiva na ulaz izlaze ostalih neurona osim samog sebe. Ovakva mreža je jedan nelinearni vremenski diskretni sustav.

Ljestvičaste mreže se sastoje od jednodimenzionalnog (dvodimenzionalnog, ili višedimenzionalnog) polja neurona sa skupom izvornih čvorova koji daju ulazne signale. Svaki ulaz spojen je na svaki neuron u polju. Ova mreža, zapravo je mreža bez povratne veze gdje su neuroni raspoređeni u polje.



Slika 10: Primjer jednodimenzionalne ljestvičaste mreže, [11]

4. OPIS STRUKTURE NEURONSKE MREŽE

Učenje je proces mijenjanja težina u mreži. Odvija se kao odgovor na podatke izvana koji su predstavljeni ulaznom sloju i u nekim mrežama izlaznom sloju. Podaci koji se predstavljaju izlaznom sloju su željene vrijednosti izlaznih varijabli. Ukoliko su one poznate, radi se o nadgledanom učenju.

U neuronskom mrežama, nije točno propisana arhitektura za određeni problem. Cilj je pronaći mrežu s najmanjom duljinom učenja i strukturu mreže koja daje najbolji rezultat na uzorku za testiranje. Dobiveni rezultat nakon faze testiranja uzima se kao konačno mjerilo uspješnosti mreže. Pravila za ovu podjelu nema, osim što se preporuča najveći dio podataka ostaviti za treniranje mreže, a manji dio podataka za testiranje (npr. 70% za treniranje, 30% za testiranje). Podaci se raspoređuju u poduzorke slučajno, osim kod vremenskih serija gdje treba poštovati vremenski slijed nastajanja promatranja, tj. trenirati mrežu na starijim, a testirati na novijim podacima [9].

Pri stvaranju strukture neuronske mreže korištena su načela Kolmogorovog teorema. Kolmogorov teorem glasi: „Postoji troslojna neuronska mreža s m neurona u ulaznom, n neurona u izlaznom i $2m + 1$ neuronom u unutarnjem sloju mreže koja ostvaruje navedeno preslikavanje, tj. rješava nelinearno odvojive zadaće. Jednostavnija mreža ne može izvesti navedeno preslikavanje“ [10]. Sukladno tome teoremu, u ovom diplomskom radu, korištena je što jednostavnija neuronska mreža koja može dovoljno točno raditi predviđanje prometnog toka.

4.1. Arhitektura neuronske mreže za cestovne podatke

Važna obilježja dizajna neuronskih mreža obuhvaćaju strukturu neuronske mreže, broj ulaznih varijabli, broj skrivenih slojeva, aktivacijsku funkciju i izbor algoritma za treniranje [9]. Višeslojna mreža korištena je za predviđanje prometnih podataka za pet radnih dana na osnovu dvadeset i pet prethodnih radnih dana. Odnosno, mreža je učena na osnovu 7.200 podataka kako bi

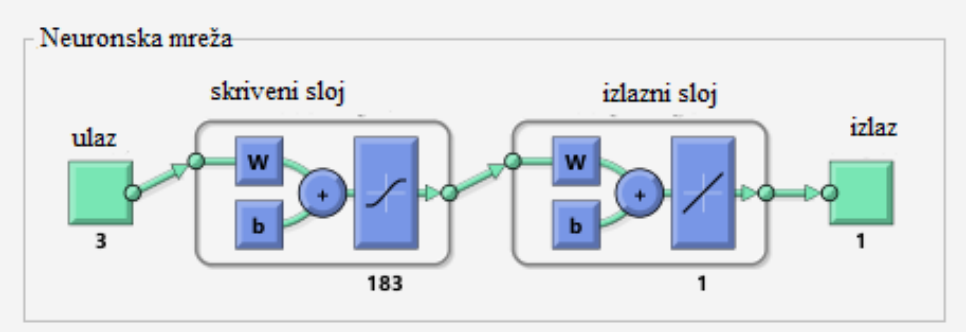
mogla predvidjeti prometni tok za svaki radni dan u tjednu. Mreža je trenirana na starijim, a testirana na novijim podacima.

Za prvi primjer, arhitektura mreže je aciklička. Aciklička mreža struktura bez povratnih veza između neurona pa signali koji krenu od ulaznih neurona nakon određenog broja prijelaza dolaze do izlaza mreže, odnosno propagacija signala je jednosmjerna. Zbog toga se koristi i engleski naziv *feedforward* mreže. Kod ovakve vrste mreža razlikujemo ulazni sloj neurona, izlazni sloj i skriveni sloj. Neuroni ulaznog sloja nemaju ulaznih signala, nemaju ni funkcionalnost neurona. Obično pod ulaznim slojem se podrazumijevaju podaci organizirani u vektor konkretnih vrijednosti [8]. U primjeru za prometni tok cestovne mreže, ulaz čine 3 podatkovna skupa: popis dana, popis sati i popis petominutnih intervala. Ta tri podatkovna skupa u kodu su nazvane kao ulazne serije, njima pripadni podaci za svaki interval nazvani su ciljne serije. Na slici 11 prikazano je nekoliko prvih varijabli ulaznih i ciljnih serija.

	1	2	3	4	5
1	1	1	5	9	
2	1	1	10	11	
3	1	1	15	2	
4	1	1	20	8	
5	1	1	25	6	
6	1	1	30	2	
7	1	1	35	3	
8	1	1	40	4	
9	1	1	45	10	
10	1	1	50	8	
11	1	1	55	1	
12	1	1	60	7	
13	1	2	5	2	
14	1	2	10	1	
15	1	2	15	3	
16	1	2	20	1	
17	1	2	25	1	
18	1	2	30	4	
19	1	2	35	4	
20	1	2	40	4	
21	1	2	45	3	

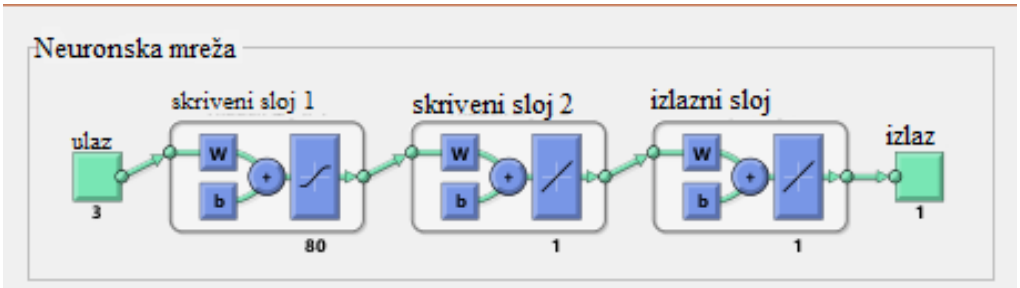
Slika 11: Ulazni podaci za predviđanje u tabličnom obliku

Slika 11 primjer je prvih dvadesetak ulaznih podataka. Prvi stupac označava dan. Ponedjeljak je označen brojem jedan, utorak brojem dva, sve do petka označenog brojem 5. Isti dan se uzastopno ponavlja 288 puta. Drugi stupac označava broj sata. Broj 1 označava prvi sat u određenom danu. Isti broj sata uzastopno se pojavljuje 12 puta. Posljednji podatkovni skup ulazne serije je treći stupac koji označuje petominutni interval unutar jednog sata. Četvrti stupac predstavlja ciljnu seriju, odnosno, označava broj vozila u petominutnom intervalu. Nakon učitavanja podataka, mreža počinje učenje. Za prvi primjer učenja mreže, u skrivenom sloju nalazi se 183 neurona.



Slika 12: Postavke arhitekture neuronske mreže s jednim skrivenim slojem

Za drugi primjer učenja neuronske mreže, uzeti su isti podaci, samo se arhitektura mijenjala s ciljem pronalaska arhitekture koja će što bolje naučiti danu karakteristiku. Na idućoj slici nalazi se arhitektura neuronske mreže s dva skrivena sloja. Prvi skriveni sloj ima 80 neurona, a drugi jedan.



Slika 13: Arhitektura neuronske mreže s dva skrivena sloja

4.2. Opis arhitekture neuronske mreže za telekomunikacijski promet

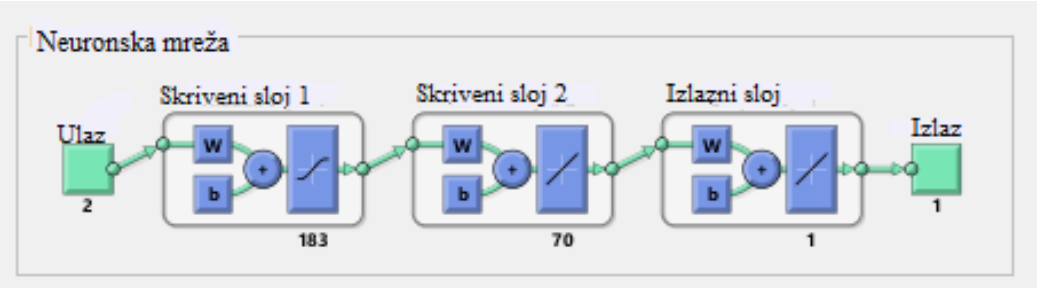
Prije samog učenja mreže, prikupljeni su podaci za predviđanje mreže. Podaci su podijeljeni u dva poduzorka. Prvi poduzorak se sastoj od 240 ulaznih varijabli (5 radnih dana) na osnovu kojih se neuronska mreže trenira. Drugi poduzorak se sastoji od 48 izlaznih varijabli, odnosno prometni tok šestog radnog dana. Pri raspoređivanju podataka u poduzorke, podaci nisu raspoređivani slučajno, nego je poštovan vremenski slijed.

Zbog ograničenog broja podataka, predviđanje za telekomunikacijsku mrežu, zahtijevalo je složeniju arhitekturu od jednog skrivenog sloja. Iskušano je više arhitektura te je za predviđanje podataka prometnog telekomunikacijskog toka te je odabrana je arhitektura koja je rezultirala najmanjom greškom. Ta arhitektura također je aciklička, kao i u primjeru za predviđanje cestovnog toka. Ulazni sloj u ovom primjeru čine dva podatkovna skupa. Zbog malog broja dana, oznaka dana nije bila potrebna. Prvi skup podataka čini oznaka sata. Drugi skup podataka su oznake minutnog intervala, odnosno da li se radi o početku sata ili o idućem 30-minutnom intervalu. Skupovi podataka iz ulaznog sloja označeni su kao ulazne serije, a pripadajući podaci telekomunikacijskog toka, izraženi u TB u kodu se vode kao ciljne serije. Na slici 14 prikazane su prve varijable ulaznih i ciljnih serija za predviđanje količine telekomunikacijskog toka.

0	1	3,3350444784E+16
0	2	7,0421257560E+16
1	1	7,6081109066E+16
1	2	7,4382230139E+16
2	1	6,6358231478E+16
2	2	6,5801415639E+16
3	1	8,0033252200E+16
3	2	1,1069623665E+17
4	1	5,7857435503E+16
4	2	5,5090757859E+16
5	1	6,2954486139E+16
5	2	4,7965850137E+16
6	1	6,1457454905E+16
6	2	4,4331712319E+16
7	1	5,4793279669E+16
7	2	3,6241488434E+16
8	1	2,5439856465E+16
8	2	2,0659646702E+16
9	1	1,4891469689E+16
9	2	1,0544172430E+16
10	1	1,2240507107E+16
10	2	6,8783579394E+15
11	1	6,2274210711E+15
11	2	4,3606635386E+15
12	1	6,3142848622E+15
12	2	5,1434731968E+15
13	1	1,2430477369E+16
13	2	2,1096058316E+16

Slika 14: Ulazni podaci za predviđanje količine telekomunikacijskog toka u tabličnom obliku

Na slici 14 prikazano je prvih dvadesetak varijabli za učenje neuronske mreže. Neuronska mreža ukupno uči pet radnih dana.Budući da se radi o manjem skupu podataka, gdje je lako moguće razlikovati dan, broj dana nema svoju oznaku. Prvi stupac predstavlja sat u određenom danu. Svaki sat u danu je ponovljen dva puta jer su podaci mjereni početkom i sredinom. Drugi stupac označuje o kojem dijelu sata se radi. 1 je oznaka za početak sata, a 2 za sredinu sata, odnosno 30 minuta svakog sata. Ostatak arhitekture čine dva skrivena sloja, prikazana na slici 15. U prvom sloju nalazi se 183 neurona, a u drugom 70 neurona



Slika 15: Arhitektura za predviđanje telekomunikacijske mreže

Na slici 15 prikazana je odabrana arhitektura, odnosno izabrani model za učenje, odnosno treniranje mreže na pripremljenim podacima telekomunikacijske mreže. Sama faza učenja je proces podešavanja težina u mreži, koje se odvija u više iteracija ili prolaza kroz mrežu. Jedna iteracija predstavlja učitavanje jednog promatranja iz podataka (jednog ulaznog i izlaznog vektora) [9]. Za primjer predviđanja telekomunikacijskog toka, najmanja greška je postignuta pri duljini učenja od 8.000 iteracija.

5. OPIS PODATAKA

Problem ovog diplomskog rada je praćenje ovisnosti varijabli koje utječu na količinu prometa, posebice ponašanje ljudi kao glavnih korisnika transportnih mreža. Pomoću neuronskih mreža predviđa se kretanje količine prometa unutar određenog perioda.

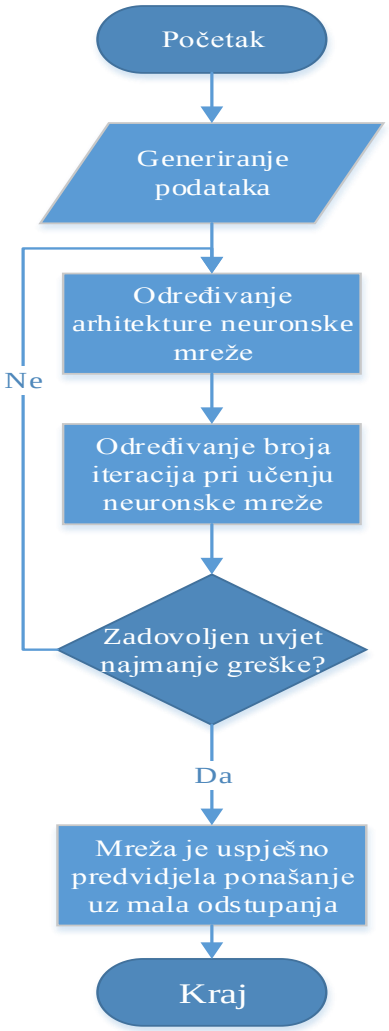
Prije samog učenja potrebno je definirati model, odnosno ulazne i izlazne varijable te prikupiti podatke iz prošlosti na kojima će se primijeniti mreža. Prikupljene podatke treba podijeliti u dva poduzorka: uzorak za treniranje i uzorak za testiranje [9]. Podaci korišteni u istraživanju nisu mijenjani, samo su prilagođavani za potrebe predviđanja u smislu stvaranja odgovarajuće podatkovne strukture.

5.1. Procedura učenja neuronskih mreža

Cijeli proces predviđanja budućih varijabli prometnog toka za narednih pet radnih dana (u cestovnom prometu) ili za naredni dan (u podatkovnom prometu), obavljen je unutar Matlab-a. Matlab je program koji služi za rješavanje različitih matematičkih problema, te čitav niz izračunavanja i simulacija vezanih uz obradu signala, regulaciju, upravljanje i identifikaciju sustava. Matlab je programski jezik za opća tehnička i znanstvena izračunavanja. Osim osnovnog sustava, Matlab sadrži i brojne programske pakete koji ga proširuju te pokrivaju gotovo sva područja inženjerske djelatnosti: obradu signala i slike, 2D i 3D grafičke prikaze, automatsko upravljanje, statističke obrade, analizu u vremenskoj i frekvencijskoj domeni, identifikaciju sustava, simboličku matematiku i druge. Jedan odvažnijih paketa je vizualni alat *Simulink*. On omogućuje simulaciju kontinuiranih i diskretnih sustava pomoću funkcijskih blok dijagrama, a pri tome ne zahtijeva od korisnika detaljno poznavanje sintakse nekog programskog jezika [12].

U ovom diplomskom radu korišten je alat *Neutral network toolbox* (skraćenica: NNToolbox). Neural Network Toolbox je ugrađen u Matlab okruženje sa svrhom simulacije rada umjetnih neuronskih mreža te njenom integracijom sa drugim funkcijama Matlab-a. Zasniva se na nizu gotovih funkcija za stvaranje, učenje i testiranje umjetnih neuronskih mreža. Ovaj alat pruža funkcije i aplikacije za modeliranje složenih nelinearnih sustava koje nije lako modelirati sa

jednadžbom zatvorenog oblika. NNToolbox podržava nadzirano učenje dinamične mreže. On također podržava učenje bez nadzora s samoorganizirajućim premještanjem i konkurentnim slojevima. Alat dopušta dizajn, treniranje, vizualizaciju i simuliranje neuronske mreže. Neutral Network Toolbox koristi se za aplikacije poput oblikovanje podataka, raspoznavanje uzoraka, predviđanje vremenskih serija te modeliranje i nadzor dinamičkih sustava [13]. Cijela procedura unutar Matlab alata, prikazana je na dijagramu toka prikazanog na slici 16.



Slika 16: Dijagram toka za proces učenja neuronske mreže za predviđanje količine prometa

5.2. Opis podataka za cestovnu mrežu

Istraživanje se zasniva na podacima prikupljenim u 2013-toj godini Ministarstva za infrastrukturni prostor, Republike Slovenije. Podaci su preuzeti s optičkog diska izdatog od Direkcije Republike Slovenije za ceste pod nazivom: „Analiza avtomatskih števcev prometa“. Na optičkom disku upisani su podaci pomoću automatskih brojila prometa za približno 650 različitih mjesta.

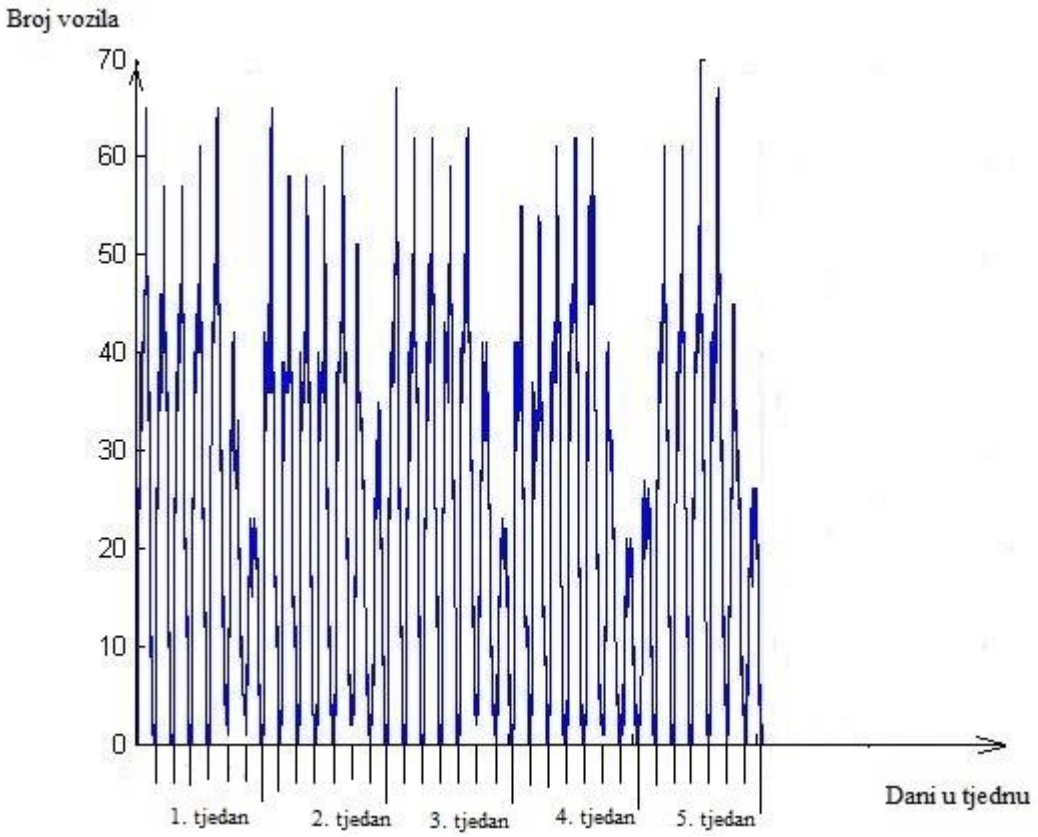
Svako mjesto, kao i prometni odsjek kojemu pripada ima svoj kod. Jedno mjesto ima pridajući tekstni dokument. U njemu se mogu pronaći izračuni broja vozila za cijelu 2013. godinu. Unutar tog tekstnog dokumenta pri samom vrhu nalaze se izračunati:

- prosječni godišnji dnevni promet ;
- prosječni dnevni promet za svaku vrstu vozila (motorno vozilo, osobno vozilo, autobusi i slično).

Nakon tih izračuna, opisan je broj vozila u svakom satu svakog dana 2013. godine. Svaki sat predstavlja jedan redak u tekstnom dokumentu. U stupcima se nalaze broj motora, osobnih vozila, lakih teretnih vozila, srednjih teretnih vozila, teških teretnih vozila, vozila s prikolicom i tegljača u oba smjera određene prometnice, kao i njihov zbroj [14].

Za potrebe istraživanja izdvojen je skup podataka za 42 radna dana. U Matlab alatu, nakon učitavanja tekstne datoteke s određenog mjesta, pomoću skripte brišu se podaci o prosječnom godišnjem i prosječnom dnevnom prometu. Rezultat skripte je tablica s podacima o danu (šifra dana), satu, i broju određene vrste vozila kao i ukupan broj vozila (pripadajućeg sata). Skripta za taj postupak opisana je u prilogu 1. Nakon toga, tablica se još mijenja kako bi se dobili podaci za petominutni interval. Uvid u te podatke postigao se drugom skriptom, opisanom u prilogu broj 2. Dakle, ona ukupan broj vozila u oba smjera u jednom satu raspodjeljuje po Gaussovoj krivulji na petominutne intervale. Rezultat druge skripte je tablica s oznakom intervala unutar sata (1-12) te pripadajućim brojem vozila u tom intervalu. Nakon toga, ta dobivena tablica se ubacuje u iduću skriptu opisanu pod prilogom broj 3. Ta skripta, najprije ubacuje još jedan stupac u tablicu, koji označuje broj dana. Poslije toga, iz skupa podataka, brišu se vikendi (dani s oznakom 6 i 7). Ostatak koda u skripti 3 odnosi se na učenje i predviđanje neuronske mreže. Najprije se odrede ulazne i ciljne serije, te se odredi broj unutarnjih slojeva i broj iteracija.

Na slici 17 prikazan je graf ulaznih podataka korištenih za učenje neuronske mreže za predviđanje prometa cestovne mreže.



Slika 17: Ulazni podaci količine cestovnog prometa

Na slici su prikazane vrijednosti tijekom šest tjedana. Svaki pojedini dan u grafu oslikan je s 288 vrijednosti. Graf je prikaz tablice koja sadrži petominutne intervale tijekom cijelog dana i pripadajuće zbrojeve vozila u tom intervalu. S lijeve strane grafa može se iščitati najveći broj automobila tijekom vršnog sata, koji ne prelazi brojku od 70. Na vodoravnoj osi svaka crtica označava kraj jednog dana tijekom perioda od pet tjedana.

5.3. Opis podataka za telekomunikacijsku mrežu

Podaci prikupljeni za predviđanje telekomunikacijske mreže rezultat su suradnje s Sveučilišnim računskim centrom (Srce). Srce je središnja infrastrukturna ustanova cjelokupnog sustava znanosti i visokog obrazovanja Republike Hrvatske koja izgrađuje, održava i daje podršku uporabi moderne računalne, komunikacijske, posredničke, podatkovne i informacijske infrastrukture (e-infrastrukture). Srce je, također i informacijski centar Sveučilišta u Zagrebu, nadležan za koordinaciju razvoja e-infrastrukture Sveučilišta u Zagrebu.

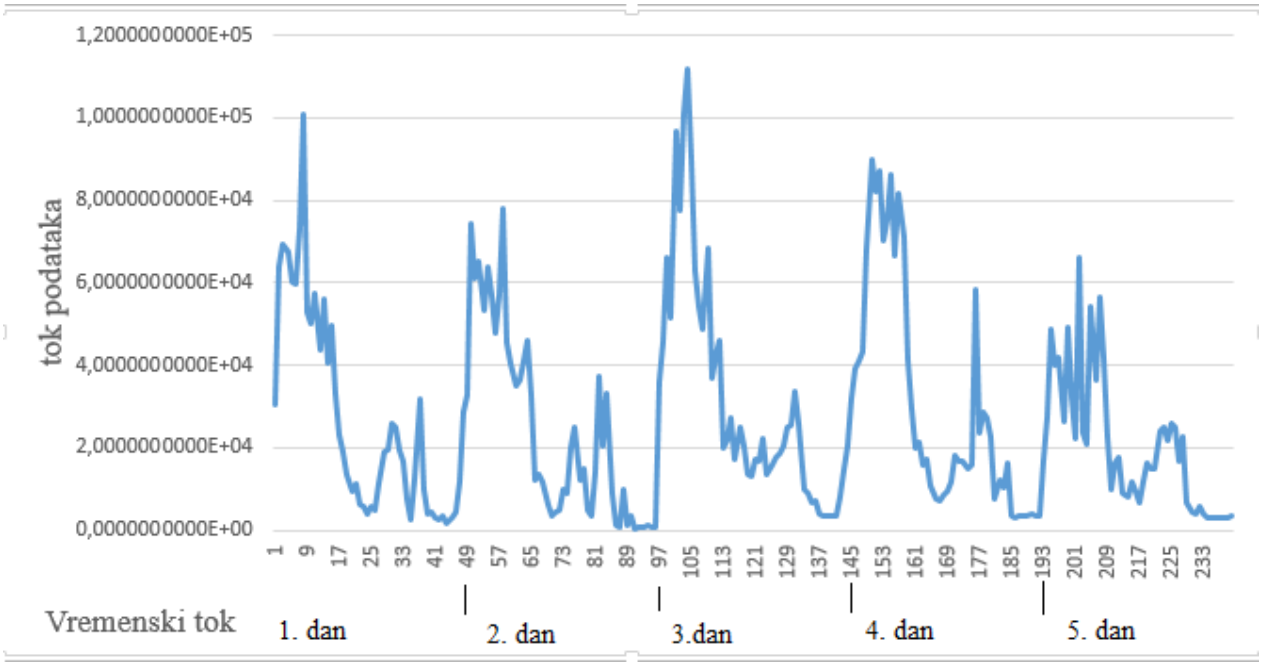
Dobiveni podaci bili su unutar rrd datoteke. Rrd je skraćenica od *Round Robin Database* i odnosi se na vrstu baze podataka koja ima svoju nepromjenjivu veličinu i pokazivače na zapisane vrijednosti. Nepromjenjiva veličina je postavljena prilikom kreiranja same baze, što omogućava lagano održavanje (nema potrebe za rotacijom) i sigurnost da neće doći do nestanka diskovnog prostora. RRDtool je koristan alat za bilježenje i prikazivanje podataka koji nastaju kao rezultat vremenskog praćenja određenih pojava [15].

Za prebacivanje datoteke u drugi format, bilo je potrebno instalirati Linux operativni sustav na virtualnom stroju (VMware). Unutar najnovije, 15.04 inačice Ubuntu-a, instalirani su RRDtool i pripadni Perl moduli. Pri naredbama za prebacivanje u xls format korištena je skripta (vidi prilog broj 4) preuzeta s Interneta [16]. Nakon pretvorbe u xsl format, svaki redak sadržavao je šifru mjesta na kojemu je mjereno, datum i vrijeme te količinu telekomunikacijskog prometa. Te tri varijable bile su odvojene zarezom. Skup podataka sadržavao je količinu telekomunikacijskog prometa mjerenog tijekom tri dana u petominutnom intervalu, zatim tijekom deset dana u intervalu od pola sata, te tijekom desetak dana mjereno u intervalu od dva sata.

Datoteke su složene prema pripadnosti. Datoteka korištena u primjeru predviđanja količine podataka sadrži sve vrijednosti okteta primljenih na ulaznom toku podataka uključujući i znakove okvira. Iduća datoteka naziva *InBroadCastPkts* sadrži brojeve paketa iz nižeg (pod)sloja prema višem (pod)sloju namijenjene završnoj (eng. *broadcast*) ili adresi. Nakon korištenja broadcast adrese, paket se kopira i šalje svim korisnicima na mreži. Datoteka pod nazivom *OutBroadcastPkts* sadrži ukupan broj paketa koji protokol više razine traži prijenos, a adresirani su na broadcast adresu na (pod)sloju, uključujući i one koje su odbačene ili se neposlane. Datoteka *InMulticastPkts* sastoji se od broja paketa koji se od (pod)sloja dostavljaju viši (pod)sloj, te se šalje na *multicast*

adresu (paket se šalje pojedinim čvorovima u mreži). *OutMulticastPkts* ima vrijednosti paketa (adresiranih na multicast adresu) koji se protokol više razine zahtijeva da se prenese, uključujući odbačene ili neposlane pakete. Datoteka naziva *InUcastPkts* omogućuje uvid u broj paketa, dostavljenih od (pod)sloja u viši (pod)sloj, a nisu bili namijenjeni za multicast ili broadcast adresu. *OutUcastPkts* sadrži broj paketa koji protokol više razine traži prijenos na niži (pod)sloj, a namijenjeni su samo za jedno odredište [17].

Dani skup podataka sadrži podatke prikupljene tijekom desetak dana, no za potrebe predviđanja izbačeni su vikendi te su uzeti samo radni dani. Podaci su prikupljani na Znanstveno-učilišnom kampusu Borongaj, gdje SRCE pruža usluge pristupa Internetu [18].



Slika 18: Prikaz ulaznih podataka korištenih za učenje neuronske mreže za previđanje telekomunikacijskog prometa

Na slici 18 prikazana je količina telekomunikacijskog toka podataka sredinom šestog mjeseca na Znanstveno- učilišnom kampusu Borongaj. Prikazani su podaci za pet radnih dana. Prvi dan je 17. lipnja 2015. Na osnovu tih podataka mreža je učena s ciljem predviđanja toka podataka šestog radnog dana. Svaki dan predočen je s 48 varijabli. Za svakih pola sata u danu stoji broj ostvarene

količine telekomunikacijskog prometa. Podaci su mjereni svakih pola sata u danu, počevši od ponoći. Vrijednosti predočene na lijevoj strani slike 14 izražene su u terabajtima (TB).

Budući da su podaci za telekomunikacijski promet manji skup nego podaci za telekomunikacijski promet, tablica s ulaznim podacima korištena je samo u jednoj skripti (prilog broj 3). Vikendi su već unaprijed uklonjeni, pa podaci su korišteni samo za učenje i predviđanje neuronske mreže.

6. REZULTATI

Sama faza učenja je proces podešavanja težina u neuronskoj mreži, koje se odvija u više iteracija ili prolaza kroz mrežu. Jedna iteracija predstavlja učitavanje jednog promatranja iz podataka (jednog ulaznog i izlaznog vektora), ali se zbog povećanja brzine učenja ponekad preporučuje učitati više promatranja odjednom. Pri tomu se broj promatranja koji se obrađuju u jednoj iteraciji zove epoha. U svakoj iteraciji računaju se nove težine, a kod nadgledanih algoritama i nova greška.

Najvažnije pitanje u fazi treniranja je koliko dugo trenirati mrežu kako bi ona dala što bolji rezultat, odnosno najmanju grešku. Ne postoje točno definirana pravila ni za ovu fazu. Dužina treniranja traži se vlastitim eksperimentiranjem ili primjenom optimizacijskih tehnika. Dobivena greška mreže na uzorku za validaciju je rezultat kojim se tumači uspješnost ili neuspješnost neuronske mreže i njezina korisnost u primjeni za predviđanje na budućim podacima. Najčešća greška koja se računa kod neuronskih mreža je srednja kvadratna greška (eng. *Root mean square error*), MSE (eng. *Mean Square Error*), MAE (eng. *Mean Absolute Error*), koeficijent korelacije i dr. U ovom radu su korišteni parametri MSE, MAE, relativna greška i koeficijent korelacije. Srednja kvadratna pogreška mjeri prosjek kvadrata pogreške, to jest, razliku između izračunatog i željenog izlaza iz mreže. Greška se uprosječuje prema broju izlaznih varijabli i u odnosu na broj slučajeva u uzorku. Srednja apsolutna pogreška (MAE) je veličina koja se upotrebljava za mjerenje koliko je blizu predviđanje eventualnim rezultatima [9]. Koeficijent korelacije pokazuje u kojoj su mjeri promjene vrijednosti jedne varijable povezane s promjenama vrijednosti druge varijable. Predznak koeficijenta korelacije govori nam o smjeru povezanosti [13].

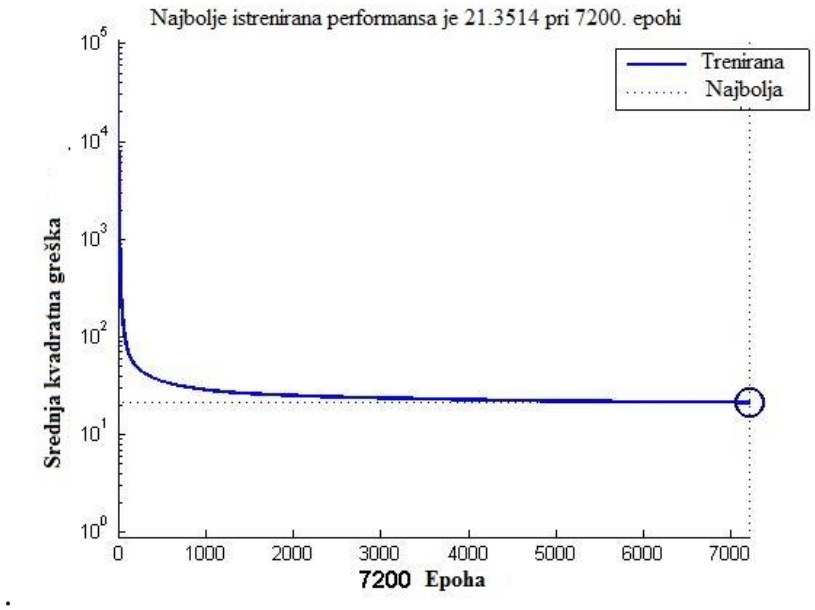
6.1. Analiza uspješnosti predviđanja za cestovnu mrežu

Na primjeru za cestovnu mrežu, u ovom istraživanju ispitano je kakav utjecaj na krajnji rezultat ima promjena broja slojeva i broja neurona unutra skrivenih slojeva. U idućoj tablici 1 prikazan je razlika grešaka u predviđanju između dvije različite arhitekture.

	Broj uzoraka za treniranje	Broj skrivenih slojeva (broj neurona)	Srednja kvadratna greška (MSE)	Apsolutna greška (MAE)	Relativna greška
Predviđanje toka za pet radnih dana	7200	1 (183)	21,76	202,01	13,27
Predviđanje toka za pet radnih dana	7200	2 (80, 1)	22,76	174,8	15,37

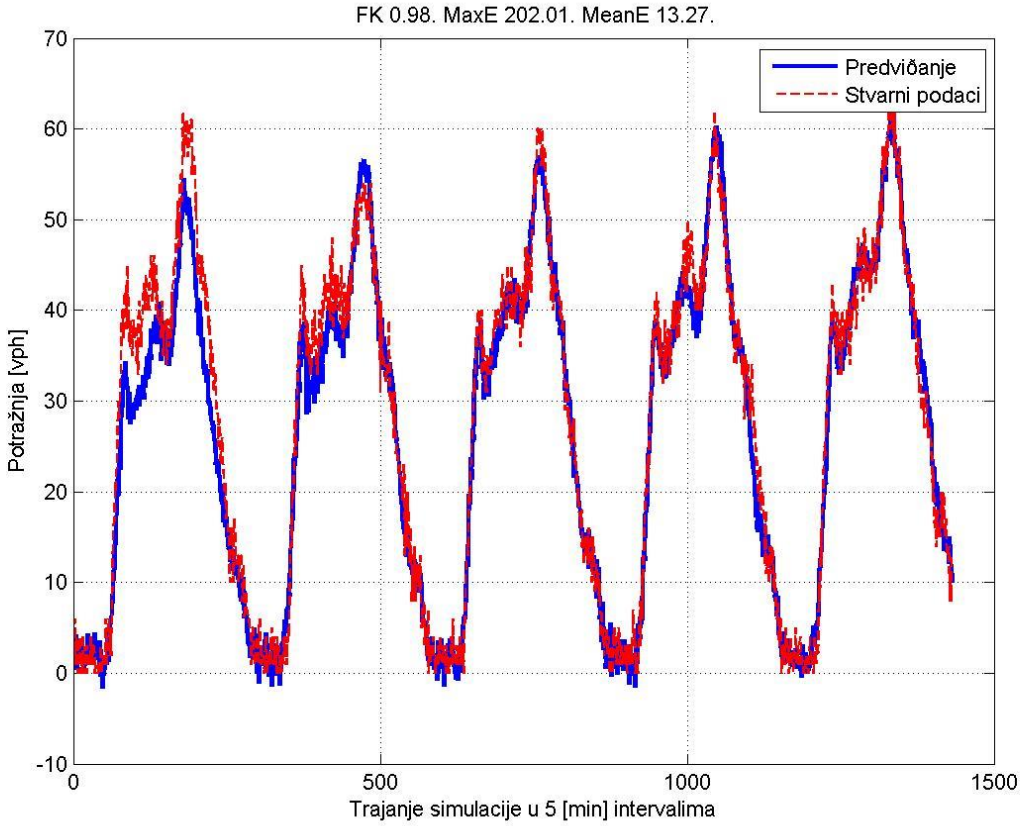
Tablica 1: Usporedba rezultata predviđanja prometnog toka različitim arhitekturama

Obično se mreža trenira u nekoliko tisuća iteracija. U ovom treniranju mreža prošla 7.200 epoha radi što boljeg rezultata. Na idućoj slici je graf odnosa greške učenja i epohe.



Slika 19: Prikaz odnosa performansi neuronske mreže i stupnja dobrote

Cilj treniranja neuronske mreže kod problema predviđanja je dobiti što manju grešku predviđanja. Stupanj pogreške mora postajati, te je poželjno da bude ispod 0,01, no i veće pogreške su prihvatljive, ukoliko se radi o većem skupu podataka. U ovom istraživanju, faktor korelacije između stvarnih i predviđenih podataka iznosi 0,98. To pokazuje da je neuronska mreže uspjela vrlo dobro naučiti danu karakteristiku prometnog toka.



Slika 20: Odnos predviđenog i stvarnog stanja cestovnog toka

Na slici 20 prikazan je odnos predviđanja i stvarnog stanja. Na nizu od 1440 podataka prometnog toka, jasno je oslikano pet radnih dana. Promatra li se slika kao koordinatni sustav, na y osi prikazan je broj vozila po satu, a na x osi prikazan je vremenski tok u petominutnim intervalima.

6.2. Analiza uspješnosti predviđanja za podatkovnu mrežu

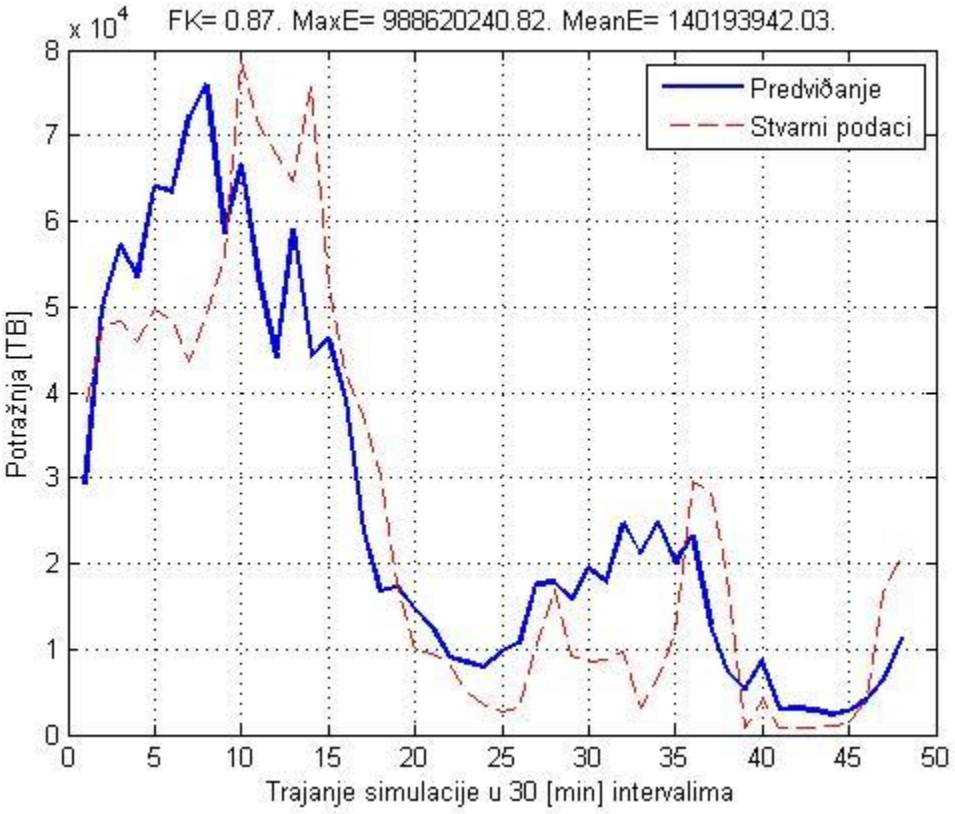
Kao na primjeru za cestovnu mrežu, za predviđanje telekomunikacijskog prometnog toka, isprobano je više arhitektura s mijenjanjem broja unutarnjih slojeva. U tablici 2 opisane su neke od arhitektura. Osim njih, isprobane su arhitekture s jednim unutarnjim slojem i sa 4 unutarnja sloja. Na kraju, izabrana je arhitektura s dva unutarnja sloja jer je imala najmanja odstupanja od stvarnog stanja. Za razliku od skupa podataka za cestovnu mrežu, skup podataka za telekomunikacijsku mrežu je dosta manji. Zbog toga, samo učenje mreže traje kraće i rezultati su lošiji.

	Broj uzoraka za treniranje	Broj skrivenih slojeva (broj neurona)	Srednja kvadratna greška (MSE)	Apsolutna greška (MAE)	Relativna greška
Predviđanje toka za naredni radni dan	240	2 (183, 70)	$1{,}17e^{8}$	988620240,82	140193942,03
Predviđanje toka za naredni radni dan	240	3 (180, 60, 50)	$2{,}47e^{8}$	1325192283,33	163850567,63

Tablica 2: Usporedba rezultata telekomunikacijskog toka pri korištenju različitih arhitektura neuronske mreže

U tablici su opisane arhitekture s različitim brojem skrivenih neurona, s različitim brojem neurona u tim skrivenim slojevima. Iz navedene tablice, može se zaključiti da više skrivenih slojeva, ne daje uvijek i bolji rezultat. Zbog toga je vrlo važno testirati i manje skrivenih neurona s različitim brojem neurona u pojedinom sloju. Osim toga, na krajnji rezultat može utjecati i broj iteracija. Za

iduću sliku, kao ciljani rezultat (predviđanje s najmanjom greškom), uzeta je prva arhitektura iz tablice, i broj iteracija je 7.000.



Slika 21: Omjer predviđenog i stvarnog niza podataka (telekomunikacijska mreža)

Na slici iznad prikazano je stvarno stanje i predviđanje neuronske mreže za jedan radni ran telekomunikacijskog toka. Budući da je mreža učena na malom skupu podataka, apsolutna i relativna greška veće su nego kod predviđanja za cestovni prometni tok. Faktor korelacije je 0,87. Iako je lošiji nego u primjeru za cestovni tok, on ipak pokazuje da je neuronska mreže uspjela vrlo dobro naučiti danu karakteristiku dnevnog telekomunikacijskog toka. Moguće je očekivati veću točnost uz korištenje većeg skupa podataka za učenje.

7. ZAKLJUČAK

U ovom diplomskom radu, neuronske mreže su korištene za predviđanje količine podataka. Na osnovu ulaznih varijabli, prikupljenih tijekom određenog perioda, predviđane su varijable narednog vremenskog intervala. Pri korištenju neuronske mreže, korištena je višeslojna mrežna arhitektura s jednim ili više skrivenih slojeva. Rezultati neuronske mreže s najmanjom greškom pokazuju relativno mala odstupanja od stvarnog stanja predviđanog vremenskog intervala. Pri tome je neuronska mreže uspješno naučila dnevnu karakteristiku promjene.

U ovom radu kao i u ostalim objavljenim istraživanjima do danas, neuronske mreže pokazale su se dobre rezultate predviđanja budućih varijabli. Iako su neuronske mreže relativno nova metoda, zbog svoje točnosti i brzine obrade podataka, ubrzo su postale konkurentne parametarskim metodama koje su do sada, u praksi, najviše korištene. Algoritmi neuronske mreže jednako dobro obrađuju linearne i nelinearne varijable. Upravo zbog toga neuronske mreže su primjenjive u više područja. Pogodne su za predviđanje stanja transportnih mreža, prepoznavanja uzoraka kod sortiranja poštanskih markica, segmentiranje kupaca u marketingu itd. Široki spektar upotrebe čini neuronske mreže nezamjenjivima.

Međutim, zbog velikog broja algoritama i širokog izbora arhitekture za predviđanje, neuronske mreže složenije je koristiti. Budući da je ovo metoda, koja se tek razvija, za određeni problem ili područje primjene, još uvijek ne postoji standardizirani pristup. Odnosno za svaki pokušaj predviđanja, potrebno je provjeriti više arhitektura kako bi se dobilo predviđanje s najmanjom pogreškom ili odstupanjem. Daljnjim istraživanjima i prikupljanjima podataka ulaznih varijabli, moći će se raditi na arhitekturi koja će postati model za predviđanje količine prometa sa manjom greškom od one dobivene u ovom radu.

LITERATURA

[1] Weizhong Zheng, Der-Horng Lee, M. Asce, Qixin Shi, Short-Term Freeway Traffic Flow Prediction: Bayesian Combined Neural Network Approach, Journal of Transportation Engineering, Vol. 132, No. 2, February 2006, pp. 114-121

[2] Caixia Li, Sreenatha Gopalarao Anavatti, Tapabrata Ray, Short-Term Traffic Flow Prediction Using Different Techniques, University of New South Wales, IEEE 2011

[3] Vedran Ivanac, Modeli kratkoročnog predviđanja karakteristika prometnog toka vozila, Telegra d.o.o., 2011

[4] Irina Klevecka, Short-term traffic forecasting with neural networks, Transport and Telecommunication, 2011, Volume 12, No. 2, pp. 20–27

[5] Sun Guang, Network Traffic Prediction Based on the Wavelet Analysis and Hopfield Neural Network, Internacional Journal of Future Computer and Communication, Vol. 2, No. 2, April 2013, pp. 101-105

[6] Yisheng Lv, Yanjie Duan, Wenwen Kang, Zhenxi Li, Fei-Yue Wang, Traffic Flow Prediction With Big Data: A Deep Learning Approach, IEEE Transactions on Intelligent Transportation Systems 2014

[7] Shiliang Sun, Changshui Zhang, Guoqiang Yu, A Bayesian Network Approach to Traffic Flow Forecasting, IEEE Transactions on Intelligent Transportation Systems, Vol. 7, No. 1, MARCH 2006, pp.124-132

[8] Dalbelo Bašić B., Čupić M., Šnajder J.: Umjetne neuronske mreže [Internet]. Fakultet elektrotehnike i računarstva, 2008. [zadnje uređivano: svibanj 2008; posjećeno: 23.lipanj 2015.] Dostupno na: http://www.fer.unizg.hr/_download/repository/UmjetneNeuronskeMreze.pdf

[9] Fakultet organizacije i informatike: Edukacijski repozitorij za inteligentne sustave: Neuronske mreže [Internet]. Varaždin, 2003. [zadnje uređivano: ožujak 2003, posjećeno: 23.lipanj 2015]. dostupno na : http://eris.foi.hr/11neuronske/nn-predavanje.html

[10] Ivanjko E., Gold H.: Predavanja za kolegij: Umjetna inteligencija. Fakultet prometnih znanosti, Zagreb, 2015.

[11] Lončarić S. Predavanja za kolegij: Neuronske mreže, Fakultet elektrotehnike i računarstva, Zagreb, 2012.

[12] Tomislav Petković, Kratke upute za korištenje Matlab-a, Zavod za elektroničke sustave i obradbu informacija Fakultet elektrotehnike i računarstva, Zagreb, 2005.

[13] Ivanjko E., Gregurić M. Upute za izradu laboratorijskih vježbi za kolegij: Umjetna inteligencija, Fakultet prometnih znanosti, Zagreb,2015.

[14] Republika Slovenija, Ministarstvo za infrastrukturni prostor. Analiza avtomatskih števcev prometa. Ljubljana, Direkcije Republike Slovenije za ceste, 2013.

[15] Dubravko Penezić U: RRDtool - od baze podataka do grafičkog prikaza u jednom alatu. [Internet]. Sistematic, Srce 2009. [zadnje uređivano: listopad 2009; posjećeno: 25. srpanj 2015.] dostupno na: http://sistemac.srce.unizg.hr/index.php?id=35&no_cache=1&tx_ttnews[tt_news]=356

[16] OpenNMS: Rrd2csv [Internet], [zadnje uređivano: rujan 2014; posjećeno: 23. srpanj 2015.] Dostupno na: http://www.opennms.org/wiki/Rrd2csv

[17] Gregurević I., Zvonko K. Predavanja za kolegij: Računalne mreže, Fakultet prometnih znanosti, Zagreb 2012.

[18] Goran Pasicek: Prikupljeni podaci za telekomunikacijsku mrežu na kampusu Borongaj. 2015

Popis ilustracija

Slika 1: Izgled biološkog neurona, [11] 8
Slika 2: Umjetni neuron, [10] 9
Slika 3: Prikaz funkcije skoka, izvor: [10] 12
Slika 4: Prikaz funkcije predznaka, izvor: [10] 12
Slika 5: Logistička funkcija, [10] 13
Slika 6: Učenje pod nadzorom, [11] 15
Slika 7: Odnos željenog i dobivenog rezultata, [8] 16
Slika 8: Arhitektura višeslojne mreže bez povratnih veza, [10] 17
Slika 9: Arhitektura neuronske mreže s povratnom vezom, [11] 18
Slika 10: Primjer jednodimenzionalne ljestvičaste mreže, [11] 18
Slika 11: Ulazni podaci za predviđanje u tabličnom obliku 20
Slika 12: Postavke arhitekture neuronske mreže s jednim skrivenim slojem 21
Slika 13: Arhitektura neuronske mreže s dva skrivena sloja 21
Slika 14: Ulazni podaci za predviđanje količine telekomunikacijskog toka u tabličnom obliku 23
Slika 15: Arhitektura za predviđanje telekomunikacijske mreže 24
Slika 16: Dijagram toka za proces učenja neuronske mreže za predviđanje količine prometa 26
Slika 17: Ulazni podaci količine cestovnog prometa 28
Slika 18: Prikaz ulaznih podataka korištenih za učenje neuronske mreže za previđanje telekomunikacijskog prometa 30
Slika 19: Prikaz odnosa performansi neuronske mreže i stupnja dobrote 33
Slika 20: Odnos predviđenog i stvarnog stanja cestovnog toka 34
Slika 21: Omjer predviđenog i stvarnog niza podataka (telekomunikacijska mreža) 36

Popis tablica:

Tablica 1: Usporedba rezultata predviđanja prometnog toka različitim arhitekturama 33
Tablica 2: Usporedba rezultata telekomunikacijskog toka pri korištenju različitih arhitektura neuronske mreže.....35

Prilog 1: Kod korišten u Matlab-u za učitavanje i obradu podataka kupljenih od Slovenske direkcije za ceste, Ministarstvo za promet Republike Slovenije

```
%  FILETOREAD1:  tekstualna datoteka sa podacima slovenskih brojila prometa%

%  vektor - vektor obradenih podataka u formatu

%  [mjerno_mjesto datum vrijeme_u_danu S1_motori S1_osobna_vozila S1_busevi S1_laki_kamioni S1_srednji_kamioni S1_teski_kamioni S1_kamioni_s_prikolicom S1_tegljaci ukupno_vozila_smjer1 S2_motori S2_osobna_vozila S2_busevi S2_laki_kamioni S2_srednji_kamioni S2_teski_kamioni S2_kamioni_s_prikolicom S2_tegljaci ukupno_vozila_smjer2 ukupno_vozila]

datoteka = fopen(fileToRead1);

% Procitaj prvih 25 linija datoteke

for i=1:25

   linija=fgetl(datoteka);

end

% citaj i radi obradu tako dugo dok se ne dode do oznake kraja podataka (znak '=')

% inicijaliziraj prvi redak vektora

podaci = textscan(linija,'%d %s %s %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d');

vektor = cell2mat(podaci(1));

datum = podaci(2);

vrijeme = podaci(3);

[~, ~, ~, sati, ~, ~]=datevec(cell2mat(vrijeme{1,1}));

vektor = [vektor datenum(cell2mat(datum{1,1}),'dd.mm.yy') sati];

for i=1:19

   vektor = [vektor cell2mat(podaci(i+3))];
```

```
end
linija=fgetl(datoteka);
% ucitaj ostale podatke
while ischar(linija)
    % provjera kraja datoteke
    if(linija(1) == '=')
        break
    end
    % obrada ucitanih podataka
    podaci = textscan(linija,'%d %s %s %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d %d');
    dodatniVektor = cell2mat(podaci(1));
    datum = podaci(2);
    vrijeme = podaci(3);
    [~, ~, ~, sati, ~, ~]=datevec(cell2mat(vrijeme{1,1}));
    dodatniVektor = [dodatniVektor datenum(cell2mat(datum{1,1}),'dd.mm.yy') sati];
    for i=1:19
        dodatniVektor = [dodatniVektor cell2mat(podaci(i+3))];
    end
    vektor = [vektor; dodatniVektor ];
    % ucitavanje sljedece linije
    linija=fgetl(datoteka);
end
```

Prilog 2: Kod za ekstrapolaciju podataka

```
function vph_eksp3 = genAllData( fileToRead, smjer, AADT, sigma, brojPetlje )

%vph_eksp3 = genAllData('AVTP_00006_2011_01_01__00_00_2011_12_31__23_50_A_U.TXT', 'A', 300, 7, 1)


% vph_eksp3 = genAllData( fileToRead, smjer, AADT, sigma, brojPetlje )

v = uvozSlovenija(fileToRead);

sat = v(:,3);

suma = v(:,12);

if smjer == 'B'

   suma = v(:,21);

end


kooeficijent = (AADT / mean(suma,1));


m1 = 1;

for i = 1:8760 %Svi dani

% for i = 1:24 %Dan s namanjim opterecenjem

%  for i = 7056:7080 %Dan s maksimalnim opterecenjem

%  for i = 4800:4825 %Dan s srednjim opterecenjem


   m2 = 1;
```

```
 for m = m1:(m1+12)
    distribucija = abs(normrnd(suma(i,1), sigma));
    vph_eksp3(m,brojPetlje+1) = distribucija/12;
       if brojPetlje == 1
           vph_eksp3(m,1) = m2;
           m2 = m2 + 1;
       end %if end petlja
    end
    m1 = m1 + 12;
end


% if brojPetlje == 1
%     k1 = 0;
%     m3  = 1;
%         while(true)
%           vph_eksp3(m3,1) = vph_eksp3(m3,1) + k1;
%           if mod(m3,24) == 0
%              k1 = k1 + 1;
%           end
%           if mod(m3,288) == 0
%              k1 = 1;
%           end
%           m3 = m3 + 1;
```

```
%          if m3 == 105120, break, end %105120
%        end
% end %if end petlja


end %end of particular function
```

Prilog 3: Kod za učenje mreže i predviđanje

```
clear all

%
[vektor]=uvozSlovenija('AVTP_00752_2013_01_01__00_00_2013_12_31__23_50_A_U.TXT');

%

% vph_eksp3 =
genAllData('AVTP_00752_2013_01_01__00_00_2013_12_31__23_50_A_U.TXT', 'B', 300, 7, 1
);

load Vektor_vph


PodaciZaUcenje= vektor(1488:2496,12);

daniV= vektor(1488:2496,2);


PodaciZaUcenjeData=vph_eksp3(17856:29951,2);

%get all data%


%1.Izvaditi vrijednost od vph_eksp1 vrijdnosti kako je izraèunato na konzulatacijama

%2.Izvaditi iz vektora (redak 2) matricu nx1 sa MATLAB kodovim dana (gdje je: n = 24*7*6)

%od poèetnog mjesta npr. 62 dana ili koliko smo veè proraèunali


dana = 42;%30  %upisati broj dana umjesto (*1 se moze izbrisat za sada)

 k=1;

for i = 1:24:(24*dana);

      a(k) = daniV(i,1);
```

```
  %uèiatavat ovdje iz matrice nx1 (izvaðena iz vektora)
     k = k + 1;
   end


k = 1;
for j = 1:length(a)
   for jj = 1:288
      c(k) = a(j); % varijabla c su vrijdnosti koje trebamo (kod dana) treba ih biti 288*7*6
      k    = k + 1;
   end
end



tablica1 =[c' PodaciZaUcenjeData];




podatciMinute1 = [];
for iii = 1:dana
   for i = 1:24 %dan 24 sata
       intervalPomoc = [ones(1,12)* i; 5:5:60];
```

```
    podatciMinute1 = [podatciMinute1 intervalPomoc]; %podatciMinuta1 je matrica nx2 sadrži sate i 5 minutne intervale
  end
end

%Dio koji se odnosi na ruèno raèunaje dana u tjednu prema MATLAB kodu dana weekday funkcijom
broj = 1;
for i = 1:length(c);
tjedan(broj) = weekday(c(i));
broj = broj + 1;
end
potraznjaSirovi = PodaciZaUcenjeData(1:length(tjedan),1); %ti podatci...Data idu umjest vph_eksp1...
Ukupno = [c' tjedan' podatciMinute1'];
[potraznjaSirovi2] = InterpolacijaCTMSIMdemand2( potraznjaSirovi, 2 );
Ukupno = [Ukupno potraznjaSirovi2];


% %brisanje vikenda
pod2 = Ukupno;
velicinaMatrice1 = size(pod2);
dani = pod2(:,2);
for i=velicinaMatrice1:-1:1;
```

```
    if dani(i,1) == 6 || dani(i,1) == 7
%       if matDat ==
      pod2(i,:) = [];
   end
end

pod2(:,1) = [];%Brisanje datuma u prvom stupcu

pod_ucenje      = double(pod2(1:end,:));
%OVAJ DIO ZA PREDVIĐANJE
p = [pod_ucenje(:,1) pod_ucenje(:,2) pod_ucenje(:,3)]';
t = [pod_ucenje(:,4)]';

BrojPredvidanja = 25*288+0; %vrijednost jednog dana 288
PP = 4*288+279; %ovoliko se koraka unaprijed predviđa

ulazneSerije = p(:,1:BrojPredvidanja);
ciljneSerije = t(1,1:BrojPredvidanja);

ulazneSerijeVri = con2seq(p(:,BrojPredvidanja+1:BrojPredvidanja + 1+PP));
ciljneSerijeVri = t(1,BrojPredvidanja + 1:BrojPredvidanja + 1+PP);
%ulazneSerijeVal = p(4,BrojPredvidanja+1:BrojPredvidanja + 1+PP);
ulazneSerijeVal = t(1,BrojPredvidanja+1:BrojPredvidanja + 1+PP);
```

```
%  p = con2seq(p);

p1 = con2seq(ulazneSerije);

t1 = con2seq(ciljneSerije);


net = newff(minmax(p1), t1, [183],{'tansig'},'trainrp');


net.trainParam.show = 50;

net.trainParam.epochs = 7200;

net.trainParam.showCommandLine = 1;

net.trainParam.max_fail = 1112;


[net,tr]=train(net,p1,t1);

Y = net(ulazneSerijeVri);



figure(1)

Pmatrica = cell2mat(p1);

Pmatrica(:,end) = [];

plot(cell2mat(Y(1,:)),'b','DisplayName','Predviđanje','LineWidth',2);

hold on

plot(ciljneSerijeVri,'r--','DisplayName','Stvarni podaci','LineWidth',1);
```

```
x = double((cell2mat(Y(1,:))));

yyy = double(ciljneSerijeVri);

faktor_korelacije = abs(corr(x',yyy'));

X = x;

Xapp = yyy;

D = abs(X-Xapp).^2;

MaxSE = max(D);

MeanSE = sum(D(:))/numel(X);


ylabel('Potražnja [vph]');

xlabel('Trajanje simulacije u 5 [min] intervalima');

legend('show'); %prikazi legendu

title(sprintf(' FK %.2f. MaxE %.2f. MeanE %.2f.',  faktor_korelacije, MaxSE, MeanSE));

grid on; %prikazi pomo?nu mrežu

grid on; %prikazi pomoænu mrežu
```

Prilog 4: Skripta korištena u Linuxu za prebacivanje datotete iz rrd u csv inačicu

```
#!/usr/bin/perl

package My::Handler;

use warnings;

use strict;

our $debug = 0;

sub new

{

	my $type = shift;

	return bless {}, $type;

}

sub start_element

{

	my ($self, $element) = @_;

	$self->{in}->{$element->{Name}} = 1;

}

sub end_element

{

	my ($self, $element) = @_;

	my $name =$element->{Name};


	if ($name eq 'row')

	{
```

```
		$self->_print_row();
	}
	$self->{in}->{$name} = 0;
}
sub characters
{
	my ($self, $characters) = @_;
	my $data = $characters->{Data};
	$data =~ s/^\s*(.*?)\s*$/$1/;
	print "characters: $data\n" if ($debug);
	if ($self->{in}->{row} and $self->{in}->{v})
	{
		$self->{value} = $data;
	}
	elsif ($self->{in}->{ds} and $self->{in}->{name})
	{
		$self->{dsname} = $data;
	}
	else
	{
		print "unknown characters: $characters->{Data}\n" if ($debug);
	}
}
```

```
sub comment
{
	my ($self, $comment) = @_;
	my $data = $comment->{Data};
	print "comment: $comment->{Data}\n" if ($debug);
	if ($data =~ /(\d\d\d\d)-(\d\d)-(\d\d) (\d\d)\:(\d\d)\:(\d\d) (\S+)/)
	{
		$self->{date} = "$2/$3/$1 $4:$5:$6";
	}
}

sub set_csv
{
	my ($self, $csv) = @_;
	$self->{csv} = $csv;
}

sub set_output
{
	my ($self, $handle) = @_;
	$self->{output} = $handle;
}
```

```
sub _print_row
{
	my $self = shift;
	$self->{csv}->combine($self->{dsname}, $self->{date}, $self->{value});
	print { $self->{output} } $self->{csv}->string, "\n";
}
1;
package main;
$| = 1; # unbuffered I/O
use warnings;
use strict;
use XML::SAX::ParserFactory;
use IO::Handle;
our $csv;
eval {
	require Text::CSV_XS;
	$csv = Text::CSV_XS->new();
};
if (not defined $csv)
{
	eval {
		require Text::CSV_PP;
```

```
        $csv = Text::CSV_PP->new();
    };
}

die "Unable to find the Text::CSV_PP or Text::CSV_XS perl modules.  Please install one of them.\n" unless (defined $csv);

my $file = shift || die "usage: $0 <rrd_file>";
my $outfile = shift;

die "$file does not exist\n" unless (-f $file);

chomp(my $rrdtool=`which rrdtool 2>/dev/null`);
die "unable to locate rrdtool, make sure it is in the path" unless (-x $rrdtool);

$file =~ s/\'/\\\'/gs;
my $input  = IO::Handle->new();
open($input, "rrdtool dump '$file' |") or die "unable to read from $file: $!";

my $output = IO::Handle->new();
if (defined $outfile)
{
    print STDERR "writing to $outfile\n";
```

```
        open($output, '>' . $outfile) or die "unable to write to $file: $!";
}
else
{
        print STDERR "writing to STDOUT\n";
        open($output, '>-') or die "unable to write to STDOUT: $!";
}

my $handler = My::Handler->new();
my $parser  = XML::SAX::ParserFactory->parser(Handler => $handler);
$handler->set_output($output);
$handler->set_csv($csv);

$csv->combine('Data Source', 'Date', 'Value');
print { $output } $csv->string, "\n";
$parser->parse_file($input);
close($input);
close($output);
sub usage
{
        print "usage: $0 <rrd_file> [output_file]\n";
        exit 1;
}
```

Sveučilište u Zagrebu
Fakultet prometnih znanosti
10000 Zagreb
Vukelićeva 4

METAPODACI

Naslov rada: Predviđanje količine prometa zasnovano na neuronskoj mreži

Autor: Matea Ravnjak
Mentor: doc. dr. sc. Eduard Ivanjko

Naslov na drugom jeziku (engleski):
Neural network based traffic prediction

Povjerenstvo za obranu:

- prof. dr. sc. Štefica Mrvelj , predsjednik
- doc. dr. sc. Eduard Ivanjko , mentor
- prof. dr. sc. Sadko Mandžuka , član
- prof. dr. sc. Hrvoje Gold , zamjena

Ustanova koja je dodjelila akademski stupanj: Fakultet prometnih znanosti Sveučilišta u Zagrebu
Zavod: Zavod za informacijsko komunikacijski promet
Vrsta studija: sveučilišni
Naziv studijskog programa: Promet
Stupanj: diplomski
Akademski naziv: mag. ing. traff.
Datum obrane diplomskog rada: **25.09.2015.**

Sveučilište u Zagrebu
Fakultet prometnih znanosti
10000 Zagreb
Vukelićeva 4

IZJAVA O AKADEMSKOJ ČESTITOSTI I SUGLASNOST

Izjavljujem i svojim potpisom potvrđujem kako je ovaj diplomski rad isključivo rezultat mog vlastitog rada koji se temelji na mojim istraživanjima i oslanja se na objavljenu literaturu što pokazuju korištene bilješke i bibliografija.

Izjavljujem kako nijedan dio rada nije napisan na nedozvoljen način, niti je prepisan iz necitiranog rada, te nijedan dio rada ne krši bilo čija autorska prava.

Izjavljujem također, kako nijedan dio rada nije iskorišten za bilo koji drugi rad u bilo kojoj drugoj visokoškolskoj, znanstvenoj ili obrazovnoj ustanovi.

Svojim potpisom potvrđujem i dajem suglasnost za javnu objavu diplomskog rada pod naslovom **Predviđanje količine prometa zasnovano na neuronskoj mreži**

na internetskim stranicama i repozitoriju Fakulteta prometnih znanosti, Digitalnom akademskom repozitoriju (DAR) pri Nacionalnoj i sveučilišnoj knjižnici u Zagrebu.

Student/ica:

U Zagrebu, 15.09.2015

(potpis)