

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 1257

**PROGNOZA VREMENSKIH SERIJA
KLIMATSKIH ELEMENATA I POKAZATELJA**

Branimir Škrlec

Zagreb, Lipanj 2019.

Zagreb, 8. ožujka 2019.

DIPLOMSKI ZADATAK br. 1257

Pristupnik: **Branimir Škrlec (0036484851)**
Studij: Informacijska i komunikacijska tehnologija
Profil: Telekomunikacije i informatika

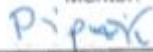
Zadatak: **Prognoza vremenskih serija klimatskih elemenata i pokazatelja**

Opis zadatka:

Vaš zadatak je proučiti i detaljno opisati tehnike strojnog učenja koje se koriste za prognozu vremenskih serija u radnim okvirima Keras/Tensorflow i/ili Spark MLlib. Pri tome analizirajte mogućnost primjene tih tehnika na prognozu vremenskih serija klimatskih elemenata i pokazatelja. Na studijskom slučaju podataka o Sjevernoatlanskoj i Arktičkoj oscilaciji primijenite i evaluirajte odgovarajuće tehnike strojnog učenja za prognozu klimatskih elemenata i pokazatelja u Hrvatskoj. Svu potrebnu literaturu, podatke i uvjete za rad osigurat će Vam Zavod za telekomunikacije.

Zadatak uručen pristupniku: 15. ožujka 2019.
Rok za predaju rada: 28. lipnja 2019.

Mentor:



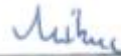
Izv. prof. dr. sc. Krešimir Pripužić

Djelovoda:



Doc. dr. sc. Marin Vuković

Predsjednik odbora za
diplomski rad profila:



Izv. prof. dr. sc. Miljenko Mikuc

*Zahvaljujem svima koji su pomogli u izradi ovog rada te također
zahvaljujem svojoj obitelji i prijateljima koji su mi bili podrška prilikom
studija.*

Sadržaj

Uvod	1
1. Prognoza vremenskih serija.....	2
1.1. Slučajni proces.....	2
1.2. Metode obrade vremenskih serija.....	3
1.3. Dijelovi vremenske serije	4
1.4. Podjela načina prognoziranja	5
2. Neuronske mreže	7
2.1. Aktivacijske funkcije.....	9
2.1.1. Tangens hiperbolni	9
2.1.2. Sigmoidalna funkcija.....	10
2.1.3. ReLU aktivacijska funkcija	11
2.2. Učenje neuronske mreže.....	12
2.3. Mreže s povratnim vezama.....	16
2.3.1. Dvosmjerne povratne neuronske mreže	16
2.3.2. Rekurzivne neuronske mreže.....	17
2.4. Mreža s dugom kratkoročnom memorijom	18
2.5. Ocjenjivanje performansi modela.....	20
3. Korištene tehnologije.....	22
3.1. Keras.....	22
3.2. Pandas	23
3.3. NumPy.....	24
4. Studijski slučaj prognoze vremenskih serija klimatskih elemenata i pokazatelja	25

4.1.	Sjeverno-atlantska oscilacija	25
4.2.	Opis podataka	27
4.3.	Korelacija podataka	30
4.4.	Opis modela	35
4.5.	Karakteristični dijelovi koda	38
4.6.	Modeli prognoze klimatskih elemenata mjernih postaja	41
4.6.1.	Bjelovar	41
4.6.2.	Ogulin	45
4.6.3.	Varaždin	47
4.6.4.	Zagreb Maksimir	49
5.	Zaključak	52
	Literatura	53
	Sažetak	55
	Summary	56
	Skraćenice	57
	Privitak A	58

Uvod

U zadnjih nekoliko godina sve vodeće zemlje svijeta stavljaju sve veći fokus na stalne i brze klimatske promjene uzrokovane globalnim zatopljenjem. Vidljivo je da se u zadnjih 50 godina mnogo toga promijenilo. Ako promatramo klimatske podatke o području Sredozemlja, jasno je da su promjene značajne. Maksimalna godišnja temperatura je u tih 50 godina porasla za približno 0.8 °C ([1]) dok se prosječna količina padalina tijekom čitave godine smanjila ([2]). Zime su postale blage i suhe, dok su ljeta iznimno vruća uz manje količine padalina.

Zbog nestabilnosti i nepoznavanja svih faktora koji mogu potencijalno utjecati na klimatske elemente, problem je uspješno predvidjeti što nam donosi sutra, prekosutra ili neka bliža budućnost.

Postoji i velik broj poznatih elemenata koji utječu na atmosfersko vrijeme. Kako bi se što preciznije mogli predvidjeti klimatski elementi, definirane su i dodatne mjere koje definiraju stanje atmosfere (Arktička oscilacija, Antarktička oscilacija...). U nekoliko dosadašnjih radova ([3],[4]) pokazano je kako jedna od tih mjera, indeks Sjeverno-Atlantske oscilacije utječe i na vrijeme u Europi, a time i na vrijeme u Republici Hrvatskoj. Upravo će u nastavku ovog rada biti provjereno postoji li poveznica između NAO indeksa i vrijednosti klimatskih elemenata u Hrvatskoj za nekoliko mjernih postaja s dovoljno dugom poviješću prikupljanja podataka. Za generiranje modela koji će pokušati utvrditi povezanost NAO indeksa i klimatskih elemenata koristit će se neuronska mreža s povratnim vezama.

1. Prognoza vremenskih serija

Vremenska serija je niz podataka koji osim vlastite mjerene vrijednosti sadrže i vremensku oznaku trenutka u kojima su podatci nastali. Vremenske serije mogu biti diskretne i kontinuirane. Najčešće, vremenske serije predstavljaju nizove koji nastaju prikupljanjem podataka u jednako razmaknutim trenutcima u vremenu, odnosno predstavljaju diskretni niz podataka. Vremenske serije se primjenjuju u statistici, raspoznavanju uzoraka, vremenskoj prognozi, predviđanju potresa, astronomiji i mnogim drugim područjima ljudskog života.

Matematički, vremenska serija je poredani niz podataka $x(t)$, $t \geq 0$ gdje je $x(t)$ tipično skalarna vrijednost, odnosno jedna realizacija slučajnog procesa ([17]).

1.1. Slučajni proces

Slučajni ili stohastički proces je familija slučajnih varijabli $X(t, \omega)$. Slučajni proces možemo interpretirati kao preslikavanje

$$X: T \times \Omega \rightarrow S \quad (1)$$

gdje je T vrijeme, Ω ishod eksperimenta a S skup stanja unutar kojih proces poprima vrijednosti. Realizacija ili trajektorija slučajnog procesa je preslikavanje

$$t \rightarrow X(t, \omega) \quad (2)$$

za neki fiksni ω . Slučajni proces $X(t, \omega)$ je (striktno) stacionaran ako za svaki pomak Δt slučajni vektori $(X(t_1), \dots, X(t_n))$ i $(X(t_1 + \Delta t), \dots, X(t_n + \Delta t))$ imaju jednaku zajedničku funkciju razdiobe, odnosno vrijedi

$$F_x(t_1)(x) = F_x(t_1 + \Delta t)(x) \quad (3)$$

Proces je slabo stacionaran ako vrijedi sljedeće:

$$E[X_t] = \mu = \text{const} ; t = 1, 2, 3, \dots \quad (4)$$

$$\gamma_X(t + h, t) = Cov(X_{t+h}, X_t) = E[(X_t - \mu)(X_{t+h} - \mu)]; neovisno o t \forall h \quad (5)$$

odnosno proces je slabo stacionaran ako se očekivanje i kovarijanca ne mijenjaju u vremenu tj. ako su konstantni.

Očekivanje ili moment prvog reda slučajnog procesa je

$$E[X(t)] = \int_{-\infty}^{+\infty} x f_X(t)(x) dx \quad (6)$$

gdje je $f_X(t)$ funkcija gustoće.

Kovarijanca je definirana kao aritmetička sredina umnožaka odstupanja varijabli od njihovih aritmetičkih sredina (7).

$$Cov_{xy} = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}) \quad (7)$$

Autokorelacijska funkcija (AKF) slučajnog procesa definirana je jednačbom (8) odnosno autokorelacijski koeficijent definiran je jednačbom (9). Autokovarijanca u trenutku $t=0$ predstavlja varijancu vremenske serije.

$$R_{x,x}(t_1, t_2) = E[X(t_1)X(t_2)] = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x_1 x_2 f_{x(t_1),x(t_2)}(x_1, x_2) dx_1 dx_2 \quad (8)$$

$$\rho_x(h) = \frac{\gamma_x(h)}{\gamma_x(0)} \quad (9)$$

1.2. Metode obrade vremenskih serija

Jedan od problema koji se nameću je što nam govore generirani podatci, odnosno što možemo zaključiti iz njihove obrade. Postoje dva područja razmatranja obrade podataka vremenskih serija: analiza i predikcija ([18]).

Analiza vremenskih serija predstavlja korištenje statističkih metoda nad vremenskom serijom kako bi se izračunali značajni statistički podatci i njihove karakteristike poput utjecaja vanjskih faktora, sezonskih trendova i sličnog. Analiza vremenskih serija uključuje i razvijanje modela koji najbolje opisuju promatranu vremensku seriju za bolje razumijevanje prikrivenih uzroka. Kvaliteta tako generiranih modela ovisi o tome koliko dobro model opisuje generirane podatke.

Predikcija vremenskih serija jest predviđanje jedne ili više sljedećih vrijednosti na temelju prethodno viđenih vrijednosti. Za razliku od analize, ovdje nije moguće koristiti podatke za generiranje modela koji opisuje vremensku seriju jer podatci još ne postoje. Umjesto tog, stvaraju se modeli na temelju prethodnih vrijednosti koji služe za aproksimaciju budućih vrijednosti. Drugim riječima, predikciju vremenskih serija možemo opisati kao predviđanje budućnosti shvaćanjem i analizom prošlosti.

Iz opisanog je vidljivo da je uspješnost predviđanja budućih vrijednosti ovisna o modelu koji opisuje podatke.

1.3. Dijelovi vremenske serije

Svaku vremensku seriju moglo bi se podijeliti na 4 dijela. Ti dijelovi su kretanje ili trend (*engl. trend*), sezonalnost (*engl. seasonality*), ciklus (*engl. cyclical*) i slučajne promjene ili šum (*engl. noise*).

Kretanje ili trend predstavlja povećanje ili smanjenje (može biti linearno, eksponencijalno ili neko drugo) ponašanja serije tijekom vremena, odnosno fluktuacije u vrijednosti podataka serije. Kretanje često ima smisla promatrati u dužem vremenskom periodu, a može biti rastuće (porast), padajuće (pad) i konstantno. Primjere kretanja podataka možemo pronaći posvuda, npr. broj stanovnika planeta Zemlje, broj kripto valuta koje imaju trend rasta te stopa mortaliteta koja ima padajući trend.

Sezonalnost predstavlja dio vremenske serije koji se ponavlja tijekom vremena, odnosno pravilno odstupanje. Na sezonalnost vremenske serije utječu vanjski faktori

poput klimatskih i vremenskih uvjeta, običaji, tradicije i slično. Kao vrlo dobar primjer sezonalnosti možemo promatrati broj turista u Republici Hrvatskoj tijekom godine, gdje je broj turista u ljetnim mjesecima mnogo viši nego tijekom ostatka godine.

Ciklus je pravilno odstupanje od prosjeka koje se događa unutar jediničnog razdoblja koje je dulje od godinu dana.

Šum predstavlja varijabilnost u promatranim podacima koja se ne može objasniti modelom. Izazvan je nepredvidljivim događajima. Neki od događaja koji mogu biti izvor šuma su ratovi, potresi, poplave i slično.

Uzimajući u obzir ova četiri dijela, postoje dva različita tipa modela za opis vremenske serije, multiplikativni i aditivni.

$$\text{Multiplikativni model: } X(t) = T(t) \times S(t) \times C(t) \times \check{S}(t) \quad (10)$$

$$\text{Aditivni model: } X(t) = T(t) + S(t) + C(t) + \check{S}(t) \quad (11)$$

U navedenim jednadžbama (jednadžbe (10) i (11)) $X(t)$ predstavlja vrijednost promatranog mjerenja u trenutku t , dok $C(t)$ predstavlja ciklus, $T(t)$ trend, $S(t)$ sezonalnost i $\check{S}(t)$ šum u trenutku t .

Multiplikativni model (10) polazi s pretpostavke da nisu svi dijelovi vremenske serije međusobno nezavisni te da mogu utjecati jedan na drugog, dok se kod aditivnog modela (11) pretpostavlja da su sva četiri dijela nezavisna jedan od drugog.

1.4. Podjela načina prognoziranja

Postoje brojni načini za predviđanje vremenskih serija. Ugrubo bi ih mogli podijeliti u tri skupine:

- prosudbene prognoze (engl. *judgemental forecast*) koje se zasnivaju na subjektivnoj procjeni, intuiciji, tajnim informacijama i slično. Ovaj način prognoziranja moguće je susresti gotovo bilogdje. Ona predstavlja donošenje zaključka o nekom skupu podataka bez ikakvih posebnih metoda za njihovu

analizu. Kao primjer bi se moglo navesti predviđanje sutrašnjeg vremena (npr. posljednja 3 dana pada kiša, pa pretpostavljamo da će padati i sutra)

- prognoze vremenskih serija s jednom varijablom (engl. *univariate methods*) kod kojih prognoze ovise o trenutnim i prethodnim vrijednostima samo jedne vremenske serije s jednom varijablom. Ovaj tip prognoze je čest kod vremenskih serija sa jednom varijablom koje ne ovise o nekim drugim elementima. Tipičan primjer ovakve serije je predviđanje idućeg broja u nizu ili slično.
- prognoze vremenskih serija s više varijabli (engl. *multivariate methods*) gdje predviđanje traženih vrijednosti ovisi, barem djelomice, o vrijednostima jedne ili više drugih varijabli vremenske serije. Ovo je najčešći oblik vremenskih serija. Primjeri ove prognoze su predviđanje cijene dionica na burzi, predviđanje uroda žitarica i slično.

U nastavku ovog rada će se razmatrati prognoza vremenske serije s više varijabli.

2. Neuronske mreže

Umjetna neuronska mreža je skup umjetnih neurona koji su međusobno povezani i interaktivni kroz operacije obrade signala. Uređena je po uzoru na ljudski mozak, odnosno kao i ljudski mozak neuronska mreža pokušava prepoznati pravilnosti i obrasce ulaznog skupa podataka te na temelju iskustva i prijašnjeg znanja pružiti generalizirani rezultat.

Neuroni (Slika 1) predstavljaju jednostavne procesne elemente koji vrše transformacije ulaza. Pojedinačni neuroni su, kao i slojevi, međusobno povezani vezama kroz koje idu signali ([19]). Svaka veza ima težinu koja određuje doprinos ulaznog podatka izlazu. Ulazne signale najčešće označavamo sa $x_1, x_2, \dots, x_n$ a težine veza sa $\omega_1, \omega_2, \dots, \omega_n$. Ulazni signali općenito su realni brojevi u intervalu $[-1,1]$, $[0,1]$ ili samo elementi iz $\{0,1\}$. Težinska suma ulaznih signala, net , definirana je sa (12).

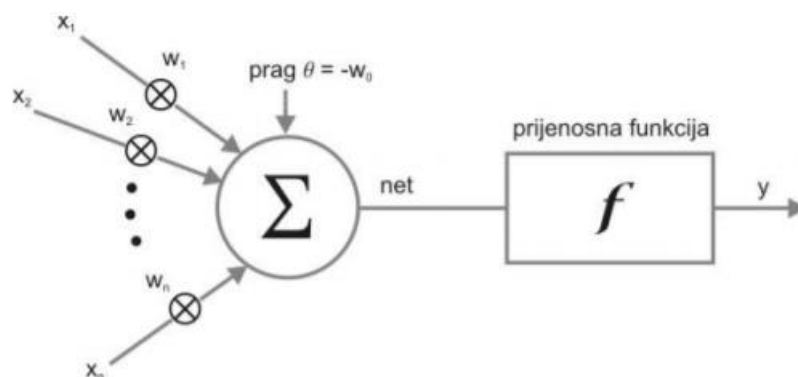
$$net = \omega_1 x_1 + \omega_2 x_2 + \dots + \omega_n x_n - \theta \quad (12)$$

Često se zbog kompaktnosti uzima da je vrijednost praga $\theta = -\omega_0$ te se dodaje ulazni signal x_0 sa fiksnom vrijednošću 1 pa pišemo jednostavnije (13).

$$net = \omega_0 x_0 + \omega_1 x_1 + \omega_2 x_2 + \dots + \omega_n x_n = \sum_{i=1}^n \omega_i x_i \quad (13)$$

y je izlaz iz prijenosne funkcije (češće se koristi naziv aktivacijska funkcija) i definiran je sa (14).

$$y = f\left(\sum_{i=1}^n \omega_i x_i\right) = f(net) \quad (14)$$

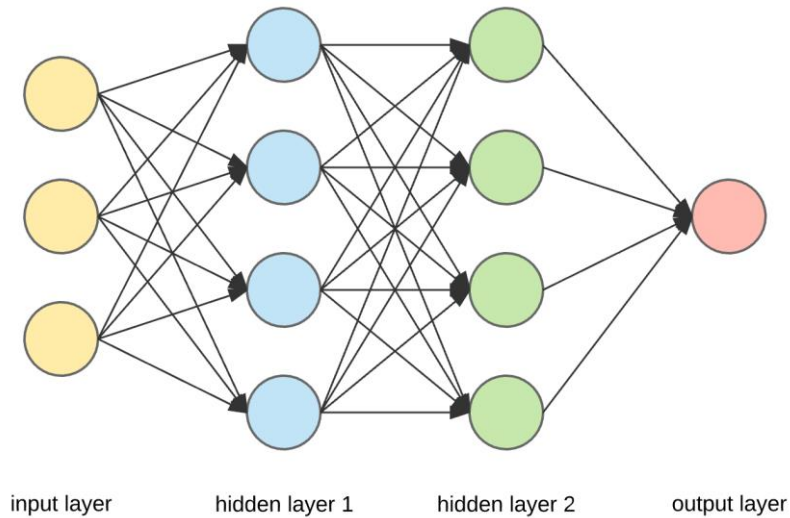


Slika 1 Umjetni neuron (preuzeto iz [19])

Odluku o tome koja je veza aktivna određuje aktivacijska funkcija (nešto više o njima u 2.1). Umjetne neuronske mreže se ističu kod predikcije vremenskih serija zbog nekolicine svojih značajki. Kao prvo, neuronske mreže su podatkovno orijentirane i adaptivne. To znači da nije potrebno definirati određen model ili neke a priori pretpostavke o distribuciji podataka. Željeni model se adaptivno formira na temelju značajki ulaznog skupa podataka te mogu raditi s nejasnim ili manjkavim podacima.. Kao drugo, neuronske mreže su inherentno nelinearne što ih čini praktičnijima i preciznijima u modeliranju složenih podatkovnih obrazaca u odnosu na tradicionalan linearan pristup. Još jedna važna značajka neuronskih mreža je ta da su one univerzalni aproksimatori funkcija, odnosno mogu aproksimirati bilo koju kontinuiranu funkciju do željene preciznosti.

Tipična neuronska mreža se sastoji od ulaznog sloja (dohvaća podatke), jednog ili više skrivenih slojeva (izvode obradu i donose zaključke) i izlaznog sloja (daje rezultat) (Slika 2).

Postoji mnogo različitih arhitektura neuronskih mreža no one nama najzanimljivije su mreže s povratnim vezama (engl. *Recurrent Neural Networks*, RNN) koje će biti detaljnije opisane u nastavku rada (2.3, 2.4).



Slika 2 Neuronska mreža sa dva skrivena sloja (preuzeto sa [20])

2.1. Aktivacijske funkcije

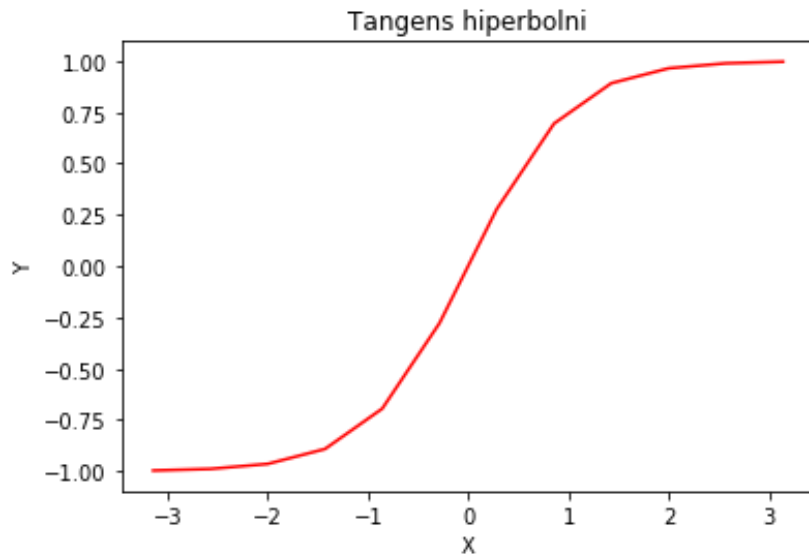
Kao što je već spomenuto, aktivacijske funkcije su funkcije koje odlučuju hoće li neki neuron biti „aktiviran“, tj. odlučuju o izlazu pojedinog neurona. Hoće li neuron biti „aktiviran“ ovisi o sumi umnoška vektora težina sa ulazima i pristranosti u tom sloju. Aktivacijske funkcije omogućuju pronalazak nelinearnih odnosa između podataka.

Postoji nekoliko često korištenih aktivacijskih funkcija koje su opisane u nastavku.

2.1.1. Tangens hiperbolni

Tangens hiperbolni(15) je funkcija koja je definirana na čitavom skupu realnih brojeva, a poprima vrijednosti u intervalu $(-1, 1)$. Ona je monotona i derivabilna na čitavoj domeni. Izgled funkcije tangens hiperbolni prikazuje Slika 3.

$$\tanh(x) = \frac{\sinh(x)}{\cosh(x)} = \frac{e^{2x} - 1}{e^{2x} + 1} \quad (15)$$

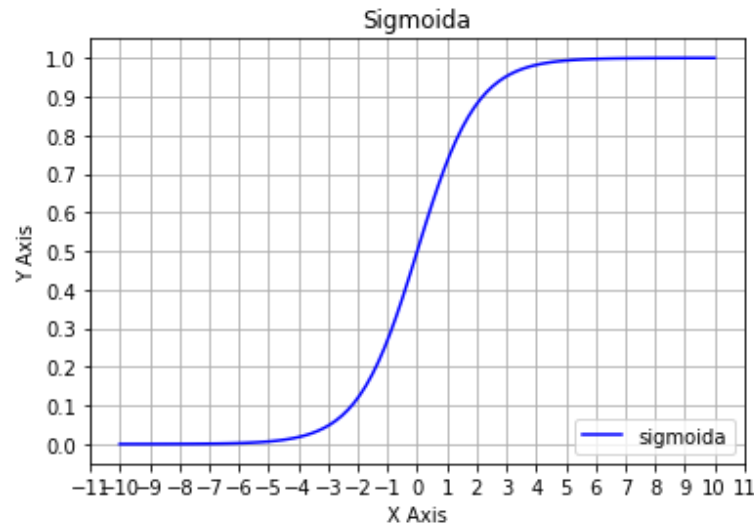


Slika 3 Tangens hiperbolni

2.1.2. Sigmoidalna funkcija

Sigmoidalna funkcija (sigmoida) (16) je također funkcija koja je definirana na čitavom skupu realnih brojeva, no za razliku od tangensa hiperbolnog ona poprima vrijednosti iz intervala (0, 1). Također je monotona i derivabilna na čitavoj domeni. Izgled sigmoide prikazuje Slika 4.

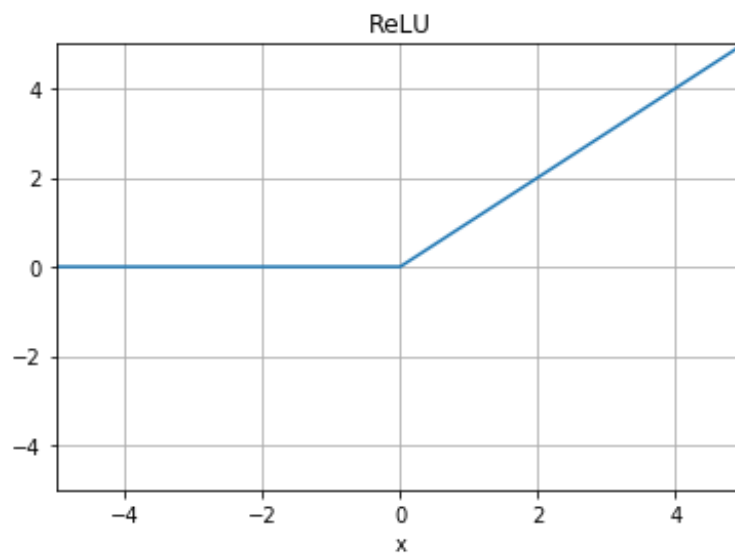
$$g(x) = \frac{1}{1 + e^{-x}} \quad (16)$$



Slika 4 Sigmoida

2.1.3. ReLU aktivacijska funkcija

ReLU (engl. *Rectified Linear Unit*) je funkcija koja je definirana na čitavom skupu realnih brojeva. Ova funkcija poprima vrijednost 0 za $x < 0$, a vrijednost x za $x > 0$. Ova funkcija je također monotona i derivabilna. Problem kod ove funkcije je taj što sve negativne vrijednosti poprimaju vrijednost 0 što smanjuje mogućnost modela da ispravno nauči. Izgled ReLU funkcije prikazuje Slika 5.



Slika 5 ReLU funkcija

2.2. Učenje neuronske mreže

Učenje mreže se sastoji od podešavanja težina veza između neurona sve dok izlaz iz mreže, provjeren na skupu podataka za testiranje, nije zadovoljavajući. Razlikujemo dva načina učenja, ovisno o tome je li nam izlaz poznat ili ne, a to su:

- nadzirano učenje (engl. *supervised learning*) – učenje mreže provodi se primjerima u obliku para (ulaz, izlaz)
- nenadzirano učenje (engl. *unsupervised learning*) – mreža uči bez poznavanja izlaza

Prilikom učenja znanje je pohranjeno u težine veza. Za uspješno shvaćanje algoritma učenja potrebno je poznavati pojmove gradijenta, pogreške i funkcije gubitka. Gradijent je vektor koji opisuje brzinu promjene neke funkcije u prostoru, a definira se kao parcijalna derivacija funkcije. Za funkciju $f(x)$, $x=(x_1, x_2, \dots, x_n)$ gradijent je definiran sa (17).

$$\nabla f = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \dots, \frac{\partial f}{\partial x_n} \right) \quad (17)$$

Na primjer, za funkciju sa 3 varijable $f(x,y,z)$, gradijent te funkcije bi bio definiran kao $\nabla f = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z} \right)$.

Pogreška predstavlja odstupanje dobivene vrijednosti od očekivane odnosno njihovu razliku.

Funkcija gubitka predstavlja funkciju koja služi za evaluaciju potencijalnih rješenja čiju vrijednost, ovisno o vrsti funkcije, neuronska mreža nastoji minimizirati ili maksimizirati. Funkcija gubitka opisuje sve karakteristike kreiranog modela jednom skalarnom vrijednošću, tj. brojem, čime je moguće međusobno uspoređivati rezultate i čije poboljšanje označava i poboljšanje modela. Funkcija gubitka se mora odabrati prilikom izračunavanja pogreške modela tijekom procesa optimizacije.

Algoritam učenja najlakše je objasniti kroz primjer. Slika 6 prikazuje jednostavnu neuronsku mrežu. Neka su x_1 i x_2 neki ulazni signali, a w_{11} , w_{12} , w_{13} , w_{14} , w_{21} i w_{22}

inicijalne težine veza između neurona (mogu biti određene slučajno) te izlaz y . Svaki neuron u ulaznom sloju (koji nije prikazan na slici jer samo propagira ulazne vrijednosti) prima ulazne signale koje zatim šalje skrivenom sloju pomnožene sa težinom veze kojom taj signal putuje te neuron u skrivenom sloju izračunava sumu svih takvih ulaznih signala prema formuli (13). U ovom slučaju za neurone u ulaznom sloju vrijedi (neka lijevi neuron u ulaznom sloju ima indeks $_1$ a desni $_2$):

$$net_1 = w_{11}x_1 + w_{13}x_2 \quad (18)$$

$$net_2 = w_{12}x_1 + w_{14}x_2 \quad (19)$$

Uz pretpostavku da je aktivacijska funkcija svih neurona skrivenog sloja sigmoida, moguće je izračunati izlaze (formula (14)) neurona sa indeksima $_1$ i $_2$ koristeći sumu ulaza iz jednadžbi (18) i (19). U ovom slučaju bi vrijedilo:

$$h_1 = \frac{1}{1 + e^{-net_1}} \quad (20)$$

$$h_2 = \frac{1}{1 + e^{-net_2}} \quad (21)$$

Gore izračunate izlaze ((20) i (21)) koristimo kao ulaze u neuron u izlaznom sloju. Suma ulaza u taj neuron definirana je kao:

$$net_3 = w_{21}h_1 + w_{22}h_2 \quad (22)$$

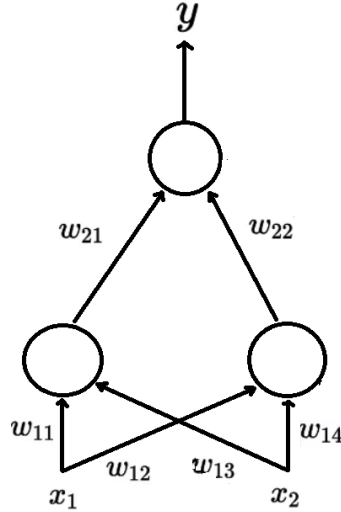
Izlaz ovog neurona (uz pretpostavku da je aktivacijska funkcija linearna) dobivamo kao:

$$y = net_3 \quad (23)$$

Kad poznajemo izlaz y neuronske mreže dobiven jednadžbom (23), moguće je izračunati pogrešku. Pogreška e (24) je definirana kao razlika dobivene vrijednosti y i očekivane vrijednosti $\hat{y}$.

$$e = y - \hat{y} \quad (24)$$

Nakon izračunavanja pogreške sljedeći korak u učenju mreže je prilagodba težina.



Slika 6 Jednostavna *feed-forward* neuronska mreža

Mreža težine veza uči koristeći algoritam propagacije pogreške unatrag (engl. *backpropagation*). Taj algoritam u općenitom slučaju koristi derivaciju funkcije gubitka po težinama ulaznih grana odnosno gradijent funkcije. U jednadžbi (25) θ^t predstavlja skup svih težina ulaznih veza u neki neuron u iteraciji t , α predstavlja brzinu učenja (parametar koji korigira podešavanje težina veza) a $E(X, \theta^t)$ predstavlja funkciju gubitka za ulazni skup uzoraka X i set težina θ .

$$\theta^{t+1} = \theta^t - \alpha \frac{\partial E(X, \theta^t)}{\partial \theta} \quad (25)$$

Najčešće se težine podešavaju koristeći gradijentni spust. Gradijentni spust je algoritam za pronalazak minimuma funkcije. Koristeći gradijentni spust se težine pomiču u smjeru negativnog gradijenta.

Jednadžbe za izračun gradijenta funkcije gubitka se razlikuje za izlazni (26) i skriveni sloj (27). U jednadžbi (26) $f_o'(net)$ predstavlja derivaciju aktivacijske funkcije izlaznog sloja, o_i^{m-1} predstavlja izlaz neurona i iz prethodnog sloja a ostale varijable su definirane ranije u ovom radu. Ova jednadžba nam govori kako podešavanje težina u posljednjem skrivenom sloju ovisi o izlazu neurona tog sloja, izlazu neurona izlaznog sloja i pogrešci modela.

U jednadžbi (27) r^{k+1} predstavlja broj neurona u $k+1$ sloju mreže, o_i^{k-1} predstavlja izlaz neurona i u sloju $k-1$, E predstavlja funkciju gubitka, w_{jl}^{k+1} predstavlja težinu veze od neurona j do neurona l u sloju $k+1$ a δ_l^{k+1} predstavlja izraženu pogrešku (engl. *error term*) za neuron l u sloju $k+1$, koja se za izlazni sloj izračunava pomoću jednadžbe (28) a za skrivene slojeve jednadžbom (29).

$$\frac{\partial E}{\partial w_{i1}^m} = (y - \hat{y}) f_o'(net) o_i^{m-1} \quad (26)$$

$$\frac{\partial E}{\partial w_{ij}^k} = f'(net^k) o_i^{k-1} \sum_{l=1}^{r^{k+1}} w_{jl}^{k+1} \delta_l^{k+1} \quad (27)$$

$$\delta_1^m = (y - \hat{y}) f'(a_1^m) \quad (28)$$

$$\delta_j^k = f'(net_j^k) \sum_{l=1}^{r^{k+1}} w_{jl}^{k+1} \delta_l^{k+1} \quad (29)$$

Poznavanjem parcijalne derivacije funkcije gubitka moguće je odrediti promjenu težine pojedine veze između neurona idući od izlaznog sloja prema ulaznom. Jednadžba (30) opisuje taj postupak.

$$\Delta w_{ij}^k = -\alpha \frac{\partial E(X, \theta)}{\partial w_{ij}^k} \quad (30)$$

Ovim skupom jednadžbi je definiran algoritam propagacije pogreške unazad i podešavanje težina veza između neurona. Algoritam će biti dodatno ilustriran na gore započetom primjeru koristeći opisane jednadžbe.

Error term izlaznog soja u gornjem primjeru odgovara pogrešci, tj. $\delta = e$. Izračun težina w_{21} i w_{22} u iteraciji 1 prikazuju jednadžbe (31) i (32).

$$w_{21}^1 = w_{21} - \alpha e h_1 \quad (31)$$

$$w_{22}^1 = w_{22} - \alpha e h_2 \quad (32)$$

U sljedećem koraku potrebno je izračunati *error term* za neurone u skrivenom sloju (jednadžbe (33) i (34)).

$$\delta_1 = h_1(1 - h_1)ew_{21} \quad (33)$$

$$\delta_2 = h_2(1 - h_2)ew_{22} \quad (34)$$

Nakon toga je moguće prilagoditi težine grana između ulaznog i skrivenog sloja koristeći gornje jednadžbe. Nakon što su prilagođene težine svih grana, završena je jedna iteracija učenja. Učenje se nastavlja dovođenjem novog ulaznog signala i ponavljanjem gore opisanih koraka.

Prilikom učenja se skup primjera dijeli na 3 odvojena skupa: skup za učenje, skup za testiranje, skup za provjeru. Prvi skup služi za učenje u širem smislu (podešavanje težinskih faktora). Primjeri iz drugog skupa služe za provjeru rada mreže s trenutnim težinskim faktorima kako bi se postupak zaustavio u trenutku degradacije performanse mreže. Umjetnu neuronsku mrežu moguće je „pretrenirati“ – nakon određenog broja iteracija mreža gubi svojstvo generalizacije i postaje stručnjak za obradu podataka iz skupa za učenje dok preostale podatke obrađuje loše. Treći skup podataka služi za provjeru preciznosti i točnosti obrade.

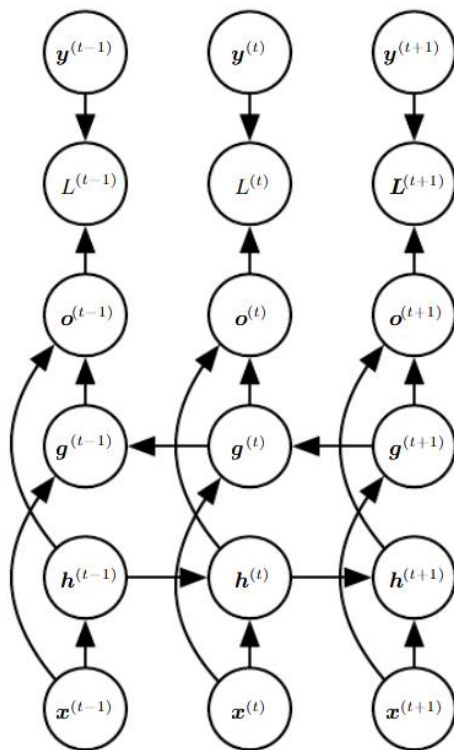
2.3. Mreže s povratnim vezama

Postoji više specifičnih arhitektura mreža s povratnim vezama, od kojih su neke češće korištene opisane u nastavku poglavlja. Prilikom izrade ovog rada je korištena neuronska mreža s dugom kratkoročnom memorijom koja je detaljno opisana u 2.4.

2.3.1. Dvosmjerne povratne neuronske mreže

U mnogim aplikacijama poželjno je ostvariti prognozu na temelju čitavog ulaznog niza, a ne samo prošlih vrijednosti. Na primjer, u raspoznavanju govora, ispravna interpretacija trenutnog glasa kao fonema može ovisiti o nekoliko sljedećih fonema ili sljedećih nekoliko riječi zbog jezičnih zavisnosti (ovo pretežno vrijedi za engleski jezik). Ako postoje dvije interpretacije trenutne riječi i obje jednako zvuče, potrebno je pogledati dalje u budućnost kako bi ih se raspoznalo. Isto načelo vrijedi i prilikom prepoznavanja rukopisa. Dvosmjerne povratne neuronske mreže (engl. *Bidirectional*

Neural Networks) su razvijene baš zbog gore opisanih problema. Dvosmjerne povratne neuronske mreže možemo predložiti kao kombinaciju dviju standardnih mreža s povratnim vezama osim što se jedna kreće kroz vrijeme od prošlosti prema budućnosti, a druga obrnuto. Izgled jedne takve mreže prikazuje Slika 7. Kao što je već spomenuto, ovakve mreže se najčešće koriste u prepoznavanju rukopisa i govora.

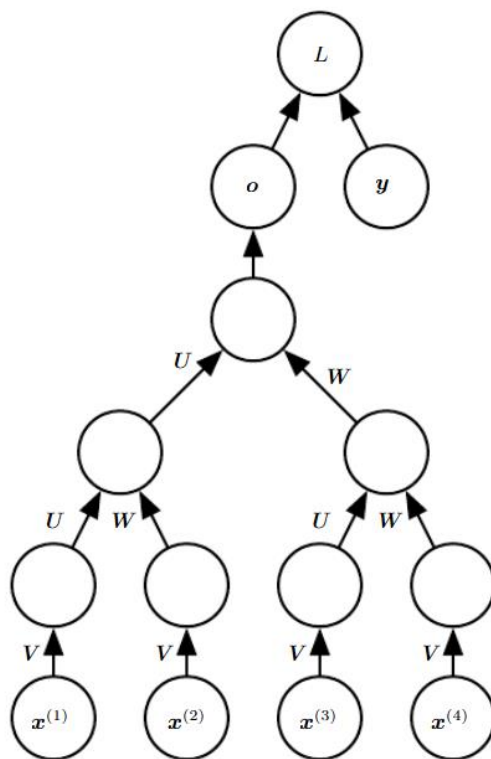


Slika 7 Dvosmjerna povratna neuronska mreža ([12])

2.3.2. Rekurzivne neuronske mreže

Rekurzivne neuronske mreže (engl. *Recursive Neural Networks*) predstavljaju još jednu generalizaciju povratnih neuronskih mreža. Način računanja težina i struktura ovakve mreže, koja nalikuje stablu, se razlikuje od arhitekture tipične mreže s povratnim vezama čija arhitektura nalikuje lancu. Jedna prednost rekurzivnih mreža je ta da se niz s varijabilnim brojem uzoraka može preslikati u reprezentaciju fiksne veličine tog niza, uz predodređen set parametara (matrica težina U, V, W). Slika 8

ilustrira slučaj učenja pod nadzorom u kojem je određen neki cilj y koji je povezan s cijelim ulaznim slijedom. Jedno od otvorenih pitanja kod rekurzivnih mreža je sam dizajn mreže (koja stablasta struktura najbolje odgovara traženom problemu). Rekurzivne neuronske mreže se koriste u obradi prirodnog jezika i računalnom vidu.



Slika 8 Rekurzivna povratna mreža ([12])

2.4. Mreža s dugom kratkoročnom memorijom

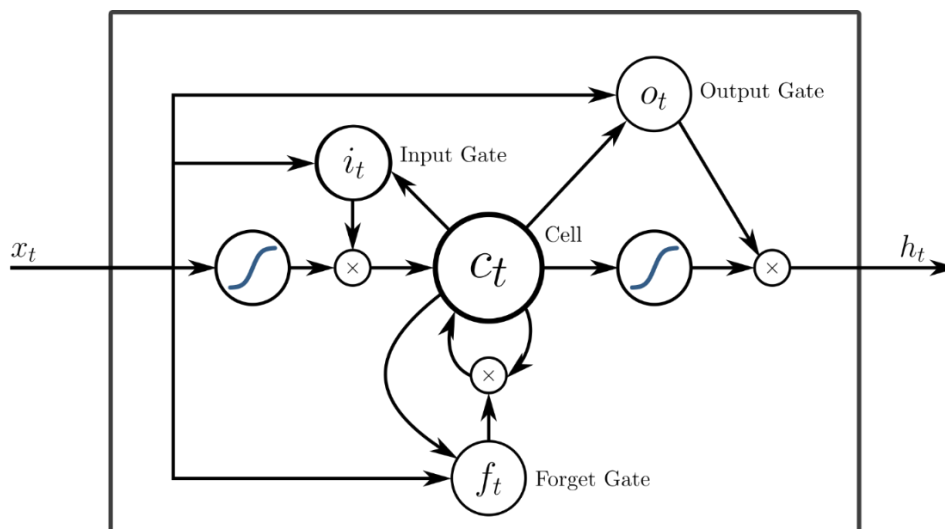
Mreža s dugom kratkoročnom memorijom (LSTM) je umjetna rekurentna neuronska mreža (*engl. Recurrent Neural Network, RNN*) koja se koristi u području dubokog učenja. Za razliku od standardnih *feed-forward* neuronskih mreža, LSTM mreža ima povratne veze koje daju neuronima mogućnost odluke na temelju ne samo trenutne već i prethodnih vrijednosti. Ne samo da može obraditi pojedinačne podatkovne točke (kao što su slike), već i cijele sekvence podataka (kao što su govor ili video). Na primjer, LSTM mreža je primjenjiva na zadatke kao što su prepoznavanje rukopisa ili prepoznavanje govora, te je pogodna za predviđanje vrijednosti vremenskih serija.

Osnovna građevna jedinica LSTM mreže je LSTM ćelija. Glavna ideja LSTM ćelije je modifikacija internog stanja. Stanje predstavlja skup informacija unutar ćelije. LSTM ćelija ima mogućnost dodavanja i oduzimanja informacija iz stanja putem tri "regulatora" (obično nazvanih vrata) toka informacija unutar LSTM jedinice: ulazna vrata, izlazna vrata i vrata za zaborav. Vrata za zaborav kao ulaz primaju vrijednost trenutnog ulaza i izlaz iz prošlog koraka. Njihova funkcija je brisanje informacija iz prethodnog stanja. Ulazna vrata odabiru koje vrijednosti ćemo ažurirati (stvara se lista kandidata koji bi mogli biti dodani u trenutno stanje). Novo stanje se računa na temelju ulaznih vrata i vrata za zaborav. Izlazna vrata predstavljaju izlaz ćelije koji je funkcija trenutnog stanja i trenutnog ulaza. Neke varijacije LSTM jedinice nemaju jednu ili više ovih vrata ili možda imaju druga vrata. Na primjer, povratne jedinice (*engl. Gated Recurrent Unit, GRU*) nemaju izlazna vrata. Intuitivno, stanica je odgovorna za praćenje zavisnosti između elemenata ulazne sekvence.

Laički, LSTM ćeliju bi mogli predložiti na sljedeći način. Ulazna vrata kontroliraju kako nova vrijednost dolazi u ćeliju, vrata za zaborav kontroliraju koliko vrijednost ostaje u ćeliji, a izlazna vrata kontroliraju stupanj do kojeg se vrijednost u ćeliji koristi za izračunavanje aktivacije izlaza LSTM jedinice.

Aktivacijska funkcija vrata LSTM često je logistička funkcija. Postoje veze u i i iz LSTM vrata, od kojih su neke povratne. Težine ovih veza, koje je potrebno naučiti tijekom treniranja, određuju kako vrata rade.

Izgled LSTM jedinice prikazan je u nastavku (Slika 9).



Slika 9 Shematski prikaz LSTM ćelije (preuzeto sa [24])

LSTM mreže su prikladne za klasificiranje, obradu i izradu predviđanja na temelju podataka vremenskih serija, budući da između važnih događaja u vremenskoj seriji mogu postojati zaostaci nepoznatog trajanja. Jedna od prednosti LSTM-a nad RNN-ovima je baš relativna neosjetljivost na duljinu razmaka između tih događaja. LSTM-ovi su razvijeni kako bi se bavili problemima gradijenta koji eksplodiraju i koji nestaju, a koji se mogu susresti pri treningu tradicionalnih RNN-ova.

U teoriji, klasični RNN-ovi mogu pratiti proizvoljne dugoročne ovisnosti u ulaznim sekvencama. Problem običnih RNN-ova je računalne prirode: kada treniramo RNN koristeći *back-propagation*, gradijenti koji su propagirani mogu "nestati" (to jest, mogu težiti nuli) ili "eksplodirati" (mogu težiti beskonačnosti), zbog računanja uključenih u proces, koji koriste brojeve konačnih preciznosti. RNN-ovi koji koriste LSTM jedinice djelomično rješavaju problem nestajanja gradijenta, jer LSTM jedinice dopuštaju da gradijenti ostanu nepromijenjeni. Međutim, LSTM mreže još uvijek mogu patiti od problema eksplodirajućeg gradijenta.

2.5. Ocjenjivanje performansi modela

Model koji je dobiven treniranjem neuronskih mreža potrebno je ocijeniti, odnosno usporediti njegovu kvalitetu s drugim modelima. Najčešće korišteni kriteriji za

ocjenjivanje modela su srednja apsolutna pogreška (*engl. Mean Absolute Error* - MAE), srednja kvadratna pogreška (*engl. Mean Squared Error* - MSE), korijen srednje kvadratne pogreške (*engl. Root Mean Squared Error* - RMSE) i srednja apsolutna postotna pogreška (*engl. Mean Absolute Percentage Error* - MAPE).

U ovom radu bit će korišten korijen srednje kvadratne pogreške (RMSE) kao ocjena kvalitete modela. Srednja kvadratna pogreška definirana je sa (35).

$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^n e_t^2} \quad (35)$$

Ova mjera kažnjava velike pogreške u predviđanju što uzrokuje kvadrat u formuli, odnosno velike pogreške su puno skuplje od malih. Srednja kvadratna pogreška nam ne govori ništa o smjeru pogreške, odnosno jesu li u većini slučajeva predviđene vrijednosti manje ili veće od stvarnih.

Za izračun svih gore opisanih vrijednosti definiramo y_t kao stvarnu vrijednost, f_t je predviđena vrijednost, $e_t = y_t - f_t$ je pogreška predviđanja a n je veličina skupa za testiranje.

3. Korištene tehnologije

Tehnologije koje su korištene za realizaciju ovog rada su Keras (uz TensorFlow kao *backend*), programske knjižnice Pandas i NumPy te programski jezik Python.

3.1. Keras

Keras je API za neuronske mreže, napisan u Pythonu te se izvodi povrh TensorFlowa, CNTK-a ili Theanoa. Fokus prilikom razvoja API je bio na mogućnosti brzog eksperimentiranja, bez potrebe za pisanjem vlastitih implementacija standardno korištenih algoritama. Keras omogućuje:

- Lagano i brzo stvaranje prototipa (kroz lakoću korištenja, modularnost i proširivost).
- Pruža podršku za stvaranje konvolucijskih i rekurentnih neuronskih mreža, kao i njihovu kombinaciju.
- Izvođenje je moguće i na CPU-u i na GPU-u.

Osnovna podatkovna struktura unutar Kerasa je model. Model predstavlja način organizacije slojeva. Najjednostavniji, i onaj koji je korišten za izradu ovog rada, je sekvencijalni model (*engl. Sequential*) koji predstavlja linearni stog slojeva. Dodavanje slojeva je jednostavno, model sadrži metodu `add()` koja kao parametar prima tip sloja (LSTM,Dense...). Kad smo zadovoljni kreiranom arhitekturom možemo podesiti njegov proces učenja metodom `compile()`. Pomoću ove metode možemo definirati funkciju pogreške (*engl. loss function*) i optimizacijski postupak. Funkcija pogreške predstavlja funkciju koju neuronska mreža nastoji optimizirati (najčešće minimizirati), a optimizacijski postupak predstavlja način prilagođavanja težina veza unutar neuronske mreže. Samo treniranje modela započinjemo funkcijom `fit()`. Ocjenu performansi modela moguće je dobiti koristeći metodu `evaluate()`. Više o načinu korištenja opisanih metoda bit će opisano u poglavlju 4.5.

Implementacija i korištenje je vrlo jednostavno i intuitivno. Osim gotovih implementacija mreža, moguće je kreirati i vlastite. Podržan je i velik broj algoritama učenja poput stohastičkog gradijentnog spusta, *Adagrad*, *Adam* i mnogi drugi. Keras sadrži velik broj funkcija za ocjenu kvalitete modela (srednja kvadratna pogreška, srednja apsolutna pogreška, srednja apsolutna postotna pogreška ...). Više o Kerasu¹ moguće je pronaći u dokumentaciji.

3.2. Pandas

Pandas je Python *open-source* programska knjižnica koja pruža alate i strukture za analizu podataka. Pandas sadrži strukture podataka koje su iznimno korisne za obradu raznih tipova podataka, a naročito vremenskih serija. One korištene u izradi ovog rada su `Series` i `DataFrame`.

`Series` predstavlja jednodimenzionalno označeno polje koje može sadržavati bilo koji tip podataka (cijeli broj, niz znakova,...). Primjer korištenja ove podatkovne strukture ilustrira sljedeća linija:

```
s = pandas.Series([0,1,2,3,4], index=['a', 'b', 'c', 'd', 'e'])
```

Gornja linija koda kreira novu strukturu tipa `Series` koja će imati zapise sa vrijednostima od 0 do 4 označene slovima od 'a' do 'e'.

`DataFrame` je dvodimenzionalna označena podatkovna struktura čiji stupci potencijalno mogu biti različitih tipova. `DataFrame` bi laički mogli predočiti kao tablicu.

Ove dvije strukture sadrže brojne metode koje se mogu koristiti za operacije s podacima, a više detalja o načinu upotrebe i primjerima moguće je pronaći u dokumentaciji².

¹ Keras, <https://keras.io>

² Pandas, <http://pandas.pydata.org/pandas-docs/stable>

3.3. NumPy

NumPy je osnovni paket za znanstveno računanje pomoću Pythona. Često se koristi jer, između ostalog, pruža strukturu koja predstavlja N -dimenzionalno polje, alate za integraciju s C++ kodom te brojne matematičke funkcije, poput Fourierovih transformacija, operacija linearne algebre i ostalog. Osim navedenog, NumPy se također može koristiti kao efikasan višedimenzionalan spremnik generičkih podataka jer, uz standardno definirane tipove podataka poput cijelih ili decimalnih brojeva, dopušta definiranje proizvoljnih tipova podataka.

Jedna od glavnih prednosti NumPy paketa je rad sa poljima. Osim lake konverzije lista i n-torki iz pythona u NumPy polja (funkcijom `array()`), jednostavno je i kreirati neke češće korištene oblike polja (polje nula naredbom `zeros()`, polje jedinica naredbom `ones()` i slično). NumPy omogućuje i slaganje polja horizontalno, tj. po stupcima naredbom `hstack()`.

Kako je NumPy paket prvenstveno namijenjen za računanje, postoji velik broj integriranih funkcija koje se mogu izvoditi nad njegovim tipovima podataka kao što su potenciranje (funkcija `power()`), korjenovanje, operacije sa matricama i slično. Više o NumPyu moguće je pronaći u dokumentaciji³.

³ NumPy, <https://www.numpy.org/devdocs/user/basics.html>

4. Studijski slučaj prognoze vremenskih serija klimatskih elemenata i pokazatelja

U sljedećem poglavlju biti će razmatran studijski slučaj prognoze vremenskih serija klimatskih elemenata i pokazatelja za meteorološke mjerne postaje u Republici Hrvatskoj. Za svaku postaju biti će kreiran prognostički model na skupu podataka za treniranje, te će svaki od njih biti evaluiran na skupu podataka za testiranje i uspoređen sa osnovnim prognostičkim modelom.

4.1. Sjeverno-atlantska oscilacija

Sjeverno-atlantska oscilacija (NAO engl. *North Atlantic Oscillation*) je jedan od najistaknutijih elemenata klimatske varijabilnosti na sjevernoj hemisferi. Ona je regionalna manifestacija cirkulacijskog polja poznatog kao Arktička oscilacija (engl. *Arctic Oscillation*). NAO pojava je najduže proučavani meteorološki fenomen za kojeg postoje dugogodišnji podatci. Definirana je kao simultano kolebanje atmosferskog tlaka između Islanda i Azora. Upravo razlika tlakova tih dvaju područja čini izvedenu mjeru koju nazivamo NAO indeks (36). Za izračun indeksa najčešće se koriste podatci postaja Lisabon, Portugal i Stykkisholmur/Reykjavik, Island zbog njihovog dugog niza mjerenja ([6],[11]).

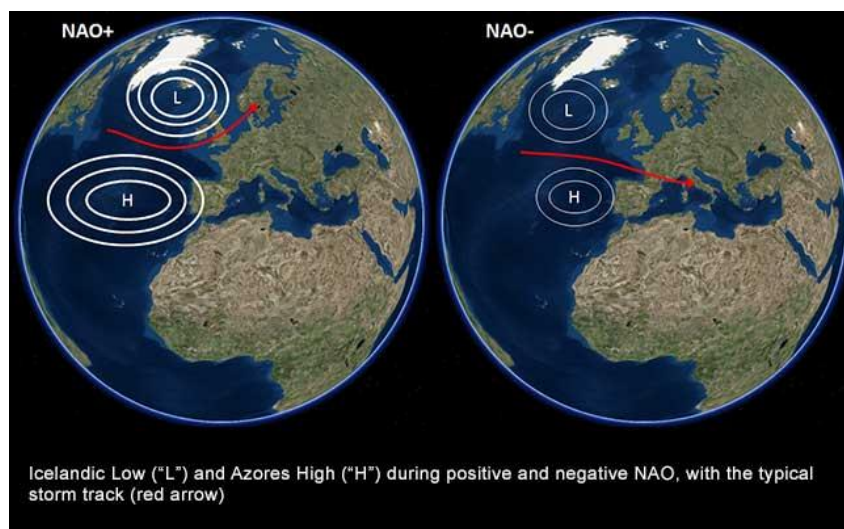
$$NAO = \frac{P_{south} - \bar{P}_{south_{30yrs}}}{std(P_{south})_{30yrs}} - \frac{P_{north} - \bar{P}_{north_{30yrs}}}{std(P_{north})_{30yrs}} \quad (36)$$

Jednadžba (36) se koristi za izračun mjesečnih vrijednosti NAO indeksa. P predstavlja srednju mjesečnu vrijednost tlaka na razini mora, P_{30yrs} predstavlja prosječnu srednju vrijednost tlaka na razini mora u proteklih 30 godina a std predstavlja standardnu devijaciju tlaka za gledani mjesec. Za izračun dnevnih NAO vrijednosti koristi se analiza empirijskih ortogonalnih funkcija primijenjena na polja tlaka na razini mora na području Europe i sjevernog Atlantika. Analiza empirijskih ortogonalnih funkcija je metoda za reduciranje dimenzija sustava (klimatološka polja se dobivaju usrednjavanjem a

karakterizira ih nelinearnost i multi dimenzioniranost te je stoga potrebno reducirati dimenzionalnost takvih sustava i izdvojiti najvažnija polja koja opisuju promjenjivost klime). Ona se temelji na određivanju glavnih vektora kovarijantne matrice ulaznih podataka s ciljem identificiranja dominantnih prostornih i vremenskih režima. Cilj analize je pronaći određeni broj međusobno nezavisnih varijabli pomoću kojih je moguće opisati početni skup podataka sa što manje gubitaka informacija ([7]).

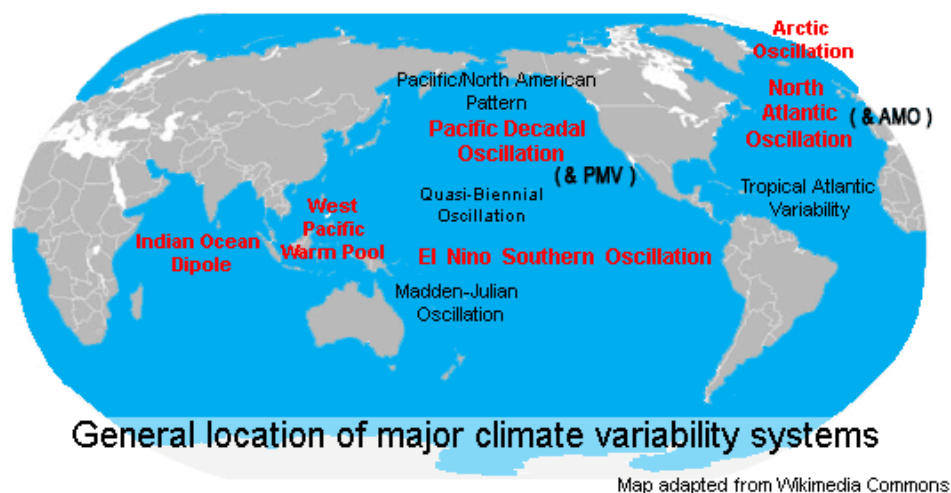
Postoje brojna istraživanja o utjecaju NAO indeksa na klimatske elemente i pokazatelje u Sjevernoj Americi, Europi i sjeveru Afrike. Mnoga od njih pokazuju kako zime s pozitivnim NAO indeksom karakterizira ispod prosječna količina oborine i niže temperature zraka u području mediteranskog dijela srednje i južne Europe, te iznad prosječne količine oborine i više temperature zraka u području sjeverne Europe, dok negativni NAO indeks djeluje suprotno (Slika 10). Također je pokazano i kako se utjecaj NAO pojave smanjuje prema istočnom dijelu Sredozemlja, ali taj utjecaj jača prema kraju zimske sezone ([20]).

U nastavku ovog rada pobliže će biti razmatran odnos NAO indeksa i klimatskih elemenata u kontinentalnom dijelu Republike Hrvatske (za razliku od priobalja Republike Hrvatske koje pripada području južne Europe, kontinentalni dio Republike Hrvatske se u pravilu ubraja u područje srednje Europe).



Slika 10 Kretanje zraka u fazama sa pozitivnim i negativnim NAO indeksom (preuzeto sa [7])

NAO nije jedina oscilacija koju je moguće koristiti za predviđanje klimatskih elemenata i pokazatelja (Slika 11). Izuzev njega postoje još i Arktička oscilacija (*engl. Arctic Oscillation - AO*), Antarktička oscilacija (*engl. Antarctic Oscillation - AAO*), Južna oscilacija (*engl. Southern Oscillation - SO*), Madden i Julian oscilacija (*engl. Madden and Julian Oscillation - MJO*) i brojne druge. Utjecaj ovih oscilacija neće biti razmatran u ovom radu, već su ovdje dane samo kao potencijalni podatci koji bi mogli dodatno poboljšati kreirani model.



Slika 11 Mapa sa svim oscilacijama (preuzeto sa [6])

4.2. Opis podataka

Podatci koje koristimo prilikom učenja, odnosno treniranja, modela su osim dnevnih vrijednosti NAO indeksa od 1. siječnja 1950. do 31. prosinca 2017. i dnevne vrijednosti temperature, tlaka i količine padalina za 4 mjerne postaje u Republici Hrvatskoj (Bjelovar, Varaždin, Ogulin, Zagreb Maksimir – ove postaje imaju potpune podatke od 1950. do danas). Podatke o dnevnim vrijednostima klimatskih elemenata u Republici Hrvatskoj za svrhu izrade ovog rada osigurao je Državni hidrometeorološki zavod (DHMZ). Podatci koji su izvorno dobiveni od DHMZ su u formi izvještaja, što znači da je podatke prije korištenja potrebno smisleno organizirati. Tablica 1, Tablica 2, Tablica 3 i Tablica 4 prikazuju format podataka koje koristimo za treniranje modela.

Tablica 1 Primjer NAO podataka

Vremenska oznaka[yyyy-mm-dd]	Vrijednost[NAO]
1950-01-01	0.365
1950-01-02	0.096
1950-01-03	-0.416

Tablica 2 Prva tri retka podataka o temperaturi za jednu od mjernih postaja

Vremenska oznaka [yyyy-mm-dd]	Vrijednost [°C]
1950-01-01	-2.5
1950-01-02	-3.0
1950-01-03	1.2

Tablica 3 Prva tri retka podataka o tlaku za jednu od mjernih postaja

Vremenska oznaka [yyyy-mm-dd]	Vrijednost [*10 Pa]
1950-01-01	9920
1950-01-02	9842
1950-01-03	9676

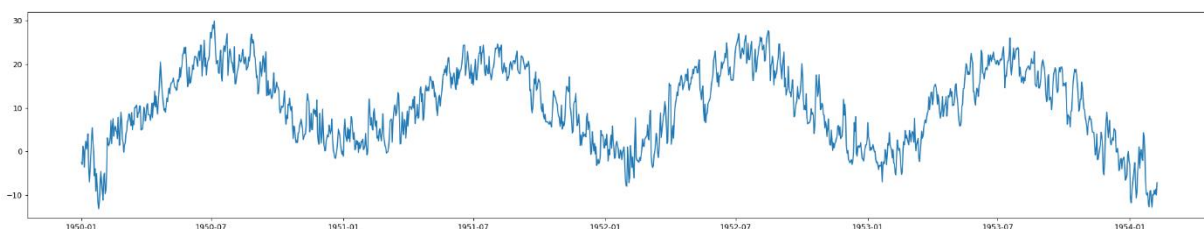
Tablica 4 Prva tri retka podataka o padalinama za jednu od mjernih postaja

Vremenska oznaka [yyyy-mm-dd]	Vrijednost [mm]
1950-01-01	0.0

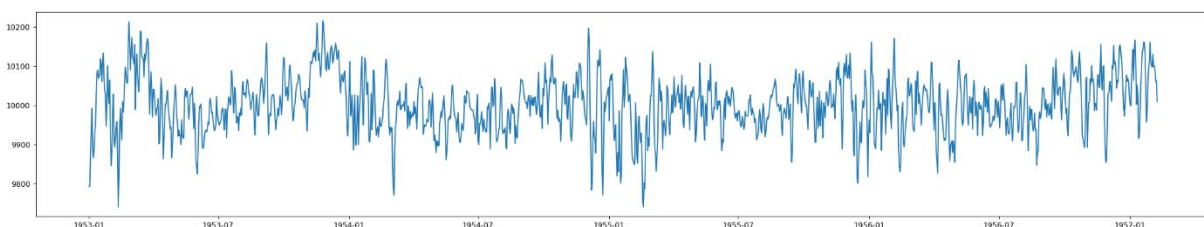
1950-01-02	0.0
1950-01-03	0.0

Dodatno, dobiveni su i podatci o istim klimatskim pokazateljima kao i za Republiku Hrvatsku za dvije mjerne postaje u Švicarskoj (Basel i Château-d'Oex). Konačni format podataka, nakon manje obrade, jednak je prethodno opisanom formatu.

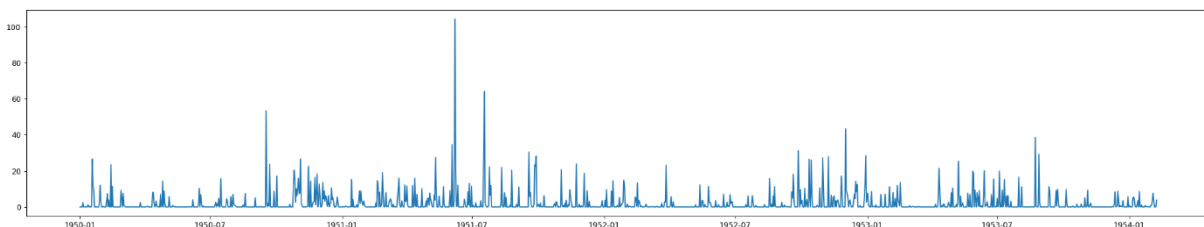
Prikaz vremenske serije temperature za mjernu postaju Bjelovar prikazuje Slika 12, vremenske serije tlaka Slika 13, vremenske serije oborina Slika 14 i vremenske serije NAO indeksa Slika 15.



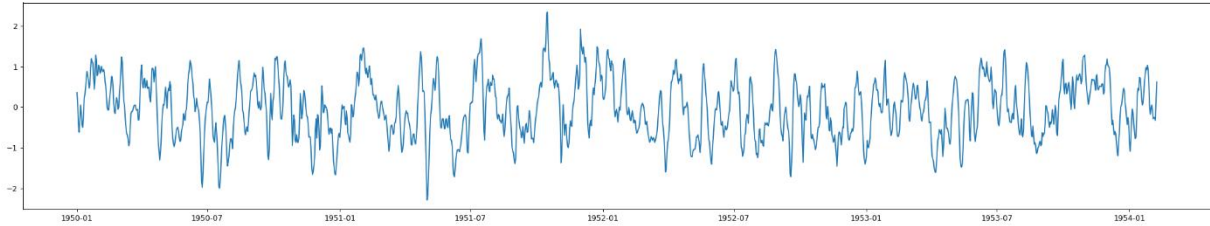
Slika 12 Vremenska serija temperature



Slika 13 Vremenska serija tlaka



Slika 14 Vremenska serija oborina



Slika 15 Vremenska serija NAO indeksa

4.3. Korelacija podataka

Prije stvaranja i treniranja modela pomoću neuronske mreže valjalo je provjeriti korelaciju vremenskih serija klimatskih elemenata i pokazatelja s NAO indeksom. Korelacija između vremenskih serija klimatskih elemenata i NAO indeksa je izračunata pomoću metode `corr()` koja je definirana nad `DataFrame`-om i koja kao parametar prima naziv koeficijenta korelacije kojeg računa (*pearson*, *spearman*, *kendall*).

Koeficijent korelacije izražava mjeru povezanosti između dvije varijable. Postoji više koeficijenata korelacije koji se koriste u različitim slučajevima. Dva najčešće korištena koeficijenta su Pearsonov i Spearmanov.

Pearsonov koeficijent korelacije koristi se u slučajevima kada između varijabli promatranog modela postoji linearna povezanost. Vrijednost Pearsonovog koeficijenta naziva se u intervalu $[-1, 1]$. Predznak koeficijenta upućuje na smjer korelacije. Za izračun Pearsonova koeficijenta korelacije potrebno je poznavati sumu kvadrata jedne varijable (npr. X), sumu kvadrata druge varijable (npr. Y) i sumu umnožaka tih varijabli.

$$S_X = \sum_{i=1}^n (X_i - \bar{X})^2 \quad (37)$$

Suma kvadrata varijable X (S_X) jednaka je sumi kvadrata odstupanja vrijednosti varijable X od njezine prosječne vrijednosti ($\bar{X}$) (37). Prosječna vrijednost varijable X definirana je sa (38).

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i \quad (38)$$

Na isti način se izračunava suma kvadrata varijable Y . Suma umnožaka varijabli X i Y jednaka je sumi umnožaka odstupanja vrijednosti varijabli od njihovih prosjeka (39).

$$S_{XY} = \sum (X_i - \bar{X})(Y_i - \bar{Y}) \quad (39)$$

Pearson koeficijent korelacije se izračunava pomoću jednadžbe (40).

$$r = \frac{S_{XY}}{\sqrt{S_X \cdot S_Y}} \quad (40)$$

Spearmanov koeficijent korelacije koristi se za mjerenje povezanosti između varijabli u slučajevima kada nije moguće primijeniti Pearsonov koeficijent korelacije. Mjeri se dosljednost povezanosti između poredanih varijabli a oblik povezanosti nije bitan (ne mora nužno biti linearan). Spearmanov koeficijent korelacije kao rezultat daje približnu vrijednost koeficijenta korelacije koji se tretira kao njegova dovoljno dobra aproksimacija. Prilikom korištenja Spearmanovog koeficijenta vrijednost varijabli potrebno je rangirati i na takav način svesti na zajedničku mjeru. Svakom uzorku iz skupa se pridjeljuje rang ovisno o vrijednosti varijable (najvećoj vrijednosti varijable se primjenjuje rang 1, drugoj najvećoj rang 2 itd.). Spearmanov koeficijent definiran je jednadžbom (41). U jednadžbi (41) d predstavlja razliku rangova vrijednosti varijabli.

$$r_s = 1 - \frac{6 \sum_{i=1}^n d_i^2}{n^3 - n} \quad (41)$$

Za ove podatke korišten je Pearsonov koeficijent korelacije za nekoliko zakašnjelih vrijednosti NAO indeksa (testirana je korelacija uz zakašnjele vrijednosti od jednog dana do 60 dana). Na slikama, danim u nastavku, koje prikazuju korelaciju zadnje slovo naziva svakog stupca/retka predstavlja mjesto mjerenja (B-Bjelovar, O-Ogulin, V-Varaždin, M-Zagreb Maksimir) dok prva riječ predstavlja o kojoj se vremenskoj seriji radi (TEMP-temperatura, TLAK-tlak, OBOR-oborine).

	TEMP_B	TEMP_O	TEMP_V	TEMP_M	TLAK_B	TLAK_O	TLAK_V	TLAK_M	OBOR_B	OBOR_O	OBOR_V	OBOR_M	NAO
TEMP_B	1	0.9761	0.9907	0.9926	-0.131	-0.1392	-0.215	-0.2374	-0.02568	-0.09172	0.001536	-0.004865	-0.03743
TEMP_O	0.9761	1	0.9771	0.9779	-0.1171	-0.1451	-0.2263	-0.2435	-0.04027	-0.1215	-0.01388	-0.0245	-0.03608
TEMP_V	0.9907	0.9771	1	0.9885	-0.1272	-0.1295	-0.2111	-0.2294	-0.02054	-0.09122	0.001877	-0.00369	-0.0312
TEMP_M	0.9926	0.9779	0.9885	1	-0.1306	-0.1315	-0.2065	-0.2304	-0.02133	-0.09079	0.004861	-0.005079	-0.03403
TLAK_B	-0.131	-0.1171	-0.1272	-0.1306	1	0.03088	0.03764	0.04358	-0.01019	-0.0003202	-0.01295	-0.008372	0.02226
TLAK_O	-0.1392	-0.1451	-0.1295	-0.1315	0.03088	1	0.9879	0.9908	-0.188	-0.2272	-0.1801	-0.2024	0.2027
TLAK_V	-0.215	-0.2263	-0.2111	-0.2065	0.03764	0.9879	1	0.9958	-0.18	-0.2071	-0.1713	-0.1935	0.2023
TLAK_M	-0.2374	-0.2435	-0.2294	-0.2304	0.04358	0.9908	0.9958	1	-0.1851	-0.2162	-0.1789	-0.1996	0.2006
OBOR_B	-0.02568	-0.04027	-0.02054	-0.02133	-0.01019	-0.188	-0.18	-0.1851	1	0.6117	0.6967	0.7251	-0.04469
OBOR_O	-0.09172	-0.1215	-0.09122	-0.09079	-0.0003202	-0.2272	-0.2071	-0.2162	0.6117	1	0.5822	0.6769	-0.04778
OBOR_V	0.001536	-0.01388	0.001877	0.004861	-0.01295	-0.1801	-0.1713	-0.1789	0.6967	0.5822	1	0.7094	-0.03711
OBOR_M	-0.004865	-0.0245	-0.00369	-0.005079	-0.008372	-0.2024	-0.1935	-0.1996	0.7251	0.6769	0.7094	1	-0.04333
NAO	-0.03743	-0.03608	-0.0312	-0.03403	0.02226	0.2027	0.2023	0.2006	-0.04469	-0.04778	-0.03711	-0.04333	1

Slika 16 Korelacija s vremenskim odmakom od jednog dana

Slika 16 prikazuje korelaciju pojedinih klimatskih elemenata za gore navedene postaje s NAO indeksom uz vremenski odmak od jednog dana. Vidljivo je da nema nikakve značajne korelacije između NAO indeksa i podataka o temperaturi i oborinama te tlaka zraka u Bjelovaru. Nešto veća korelacija postoji između NAO indeksa i tlaka za ostale mjerne postaje (oko 0.2). Zaključak je da ne postoji izražena korelacija između NAO indeksa i klimatskih elemenata.

	TEMP_B	TEMP_O	TEMP_V	TEMP_M	TLAK_B	TLAK_O	TLAK_V	TLAK_M	OBOR_B	OBOR_O	OBOR_V	OBOR_M	NAO
TEMP_B	1	0.9761	0.9907	0.9926	-0.131	-0.1392	-0.215	-0.2374	-0.02568	-0.09172	0.001536	-0.004865	-0.03207
TEMP_O	0.9761	1	0.9771	0.9779	-0.1171	-0.1451	-0.2263	-0.2435	-0.04027	-0.1215	-0.01388	-0.0245	-0.03121
TEMP_V	0.9907	0.9771	1	0.9885	-0.1272	-0.1295	-0.2111	-0.2294	-0.02054	-0.09122	0.001877	-0.00369	-0.02616
TEMP_M	0.9926	0.9779	0.9885	1	-0.1306	-0.1315	-0.2065	-0.2304	-0.02133	-0.09079	0.004861	-0.005079	-0.02921
TLAK_B	-0.131	-0.1171	-0.1272	-0.1306	1	0.03088	0.03764	0.04358	-0.01019	-0.0003202	-0.01295	-0.008372	0.02291
TLAK_O	-0.1392	-0.1451	-0.1295	-0.1315	0.03088	1	0.9879	0.9908	-0.188	-0.2272	-0.1801	-0.2024	0.1672
TLAK_V	-0.215	-0.2263	-0.2111	-0.2065	0.03764	0.9879	1	0.9958	-0.18	-0.2071	-0.1713	-0.1935	0.1674
TLAK_M	-0.2374	-0.2435	-0.2294	-0.2304	0.04358	0.9908	0.9958	1	-0.1851	-0.2162	-0.1789	-0.1996	0.1656
OBOR_B	-0.02568	-0.04027	-0.02054	-0.02133	-0.01019	-0.188	-0.18	-0.1851	1	0.6117	0.6967	0.7251	-0.04385
OBOR_O	-0.09172	-0.1215	-0.09122	-0.09079	-0.0003202	-0.2272	-0.2071	-0.2162	0.6117	1	0.5822	0.6769	-0.05122
OBOR_V	0.001536	-0.01388	0.001877	0.004861	-0.01295	-0.1801	-0.1713	-0.1789	0.6967	0.5822	1	0.7094	-0.03771
OBOR_M	-0.004865	-0.0245	-0.00369	-0.005079	-0.008372	-0.2024	-0.1935	-0.1996	0.7251	0.6769	0.7094	1	-0.04416
NAO	-0.03207	-0.03121	-0.02616	-0.02921	0.02291	0.1672	0.1674	0.1656	-0.04385	-0.05122	-0.03771	-0.04416	1

Slika 17 Korelacija s vremenskim odmakom od dva dana

Ako pogledamo korelaciju s vremenskim odmakom od dva dana (Slika 17), vidimo da i dalje ne postoji izražena korelacija između klimatskih elemenata i NAO indeksa (postaje sve manja).

	TEMP_B	TEMP_O	TEMP_V	TEMP_M	TLAK_B	TLAK_O	TLAK_V	TLAK_M	OBOR_B	OBOR_O	OBOR_V	OBOR_M	NAO
TEMP_B	1	0.9761	0.9907	0.9926	-0.131	-0.1392	-0.215	-0.2374	-0.02568	-0.09172	0.001536	-0.004865	-0.03117
TEMP_O	0.9761	1	0.9771	0.9779	-0.1171	-0.1451	-0.2263	-0.2435	-0.04027	-0.1215	-0.01388	-0.0245	-0.03817
TEMP_V	0.9907	0.9771	1	0.9885	-0.1272	-0.1295	-0.2111	-0.2294	-0.02054	-0.09122	0.001877	-0.00369	-0.03094
TEMP_M	0.9926	0.9779	0.9885	1	-0.1306	-0.1315	-0.2065	-0.2304	-0.02133	-0.09079	0.004861	-0.005079	-0.03515
TLAK_B	-0.131	-0.1171	-0.1272	-0.1306	1	0.03088	0.03764	0.04358	-0.01019	-0.0003202	-0.01295	-0.008372	-0.00644
TLAK_O	-0.1392	-0.1451	-0.1295	-0.1315	0.03088	1	0.9879	0.9908	-0.188	-0.2272	-0.1801	-0.2024	0.01765
TLAK_V	-0.215	-0.2263	-0.2111	-0.2065	0.03764	0.9879	1	0.9958	-0.18	-0.2071	-0.1713	-0.1935	0.01794
TLAK_M	-0.2374	-0.2435	-0.2294	-0.2304	0.04358	0.9908	0.9958	1	-0.1851	-0.2162	-0.1789	-0.1996	0.01776
OBOR_B	-0.02568	-0.04027	-0.02054	-0.02133	-0.01019	-0.188	-0.18	-0.1851	1	0.6117	0.6967	0.7251	-0.007135
OBOR_O	-0.09172	-0.1215	-0.09122	-0.09079	-0.0003202	-0.2272	-0.2071	-0.2162	0.6117	1	0.5822	0.6769	-0.00811
OBOR_V	0.001536	-0.01388	0.001877	0.004861	-0.01295	-0.1801	-0.1713	-0.1789	0.6967	0.5822	1	0.7094	-0.01032
OBOR_M	-0.004865	-0.0245	-0.00369	-0.005079	-0.008372	-0.2024	-0.1935	-0.1996	0.7251	0.6769	0.7094	1	-0.01007
NAO	-0.03117	-0.03817	-0.03094	-0.03515	-0.00644	0.01765	0.01794	0.01776	-0.007135	-0.00811	-0.01032	-0.01007	1

Slika 18 Korelacija s vremenskim odmakom od trideset dana

Slika 18 pokazuje korelaciju između NAO indeksa i klimatskih elemenata s vremenskim odmakom od mjesec dana. I dalje ne postoji izražena korelacija između vremenskih serija.

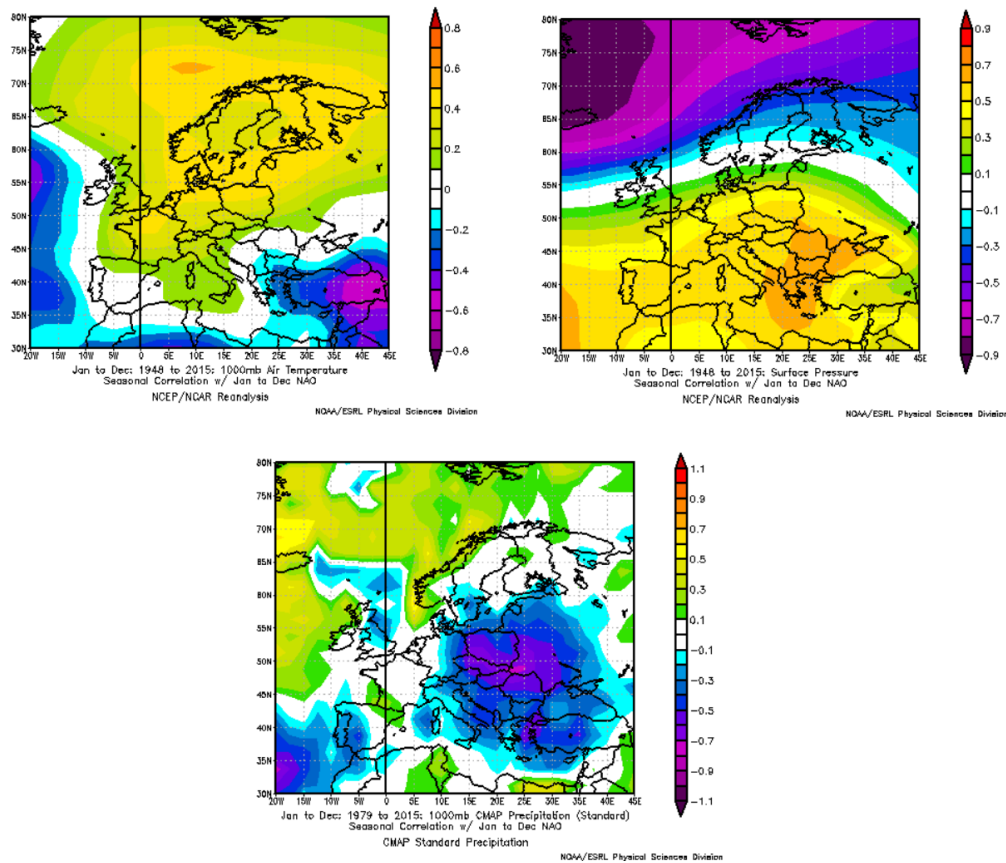
Kako bi provjerio slažu li se dobiveni rezultati s onima dosad pokazanima u mnogim radovima, korišten je *online* alat ESRL⁴ (*engl. Earth System Research Laboratory*) za izračun korelacije NAO indeksa s brojnim klimatskim elementima (Slika 19). ESRL računa korelaciju na prosječnim mjesečnim vrijednostima klimatskih elemenata, no dobiveni rezultat je sličan izračunu na temelju dnevnih vrijednosti.

Prva slika u gornjem retku prikazuje korelaciju NAO indeksa i temperature za cijelu Europu. Vidimo da je za Republiku Hrvatsku korelacija dobivena ovim alatom također oko 0, tj. ne postoji.

Druga slika u gornjem retku prikazuje korelaciju tlaka i NAO indeksa u Europi. Ona pokazuje malu korelaciju (0.2 – 0.5), što približno odgovara dobivenim rezultatima.

Slika u drugom retku prikazuje korelaciju NAO indeksa i padalina. Vidimo da se na slici koeficijent korelacije kreće između 0.1 i 0.5 za Republiku Hrvatsku što donekle odgovara dobivenim rezultatima (i dalje je korelacija slaba, no koeficijent korelacije je nešto veći od onog gore navedenog).

⁴ ESRL, <https://www.esrl.noaa.gov/psd/data/correlation>



Slika 19 Rezultati korelacije dobiveni online alatom (generirano pomoću [22])

Naime, vremenska serija količine padalina sadrži puno vrijednosti koje su jednake nuli. Isti datum u različitim godinama može poprimiti jako različite vrijednosti količine padalina. Zbog toga je izračunata i korelacija između prosječnih mjesečnih vrijednosti NAO indeksa i mjesečnih vrijednosti količina padalina. Kako današnji NAO indeks vjerojatno ne utječe na današnje padaline, sve vrijednosti NAO indeksa su zakašnjele za jedan dan. Korelacija u slučaju prosječnih mjesečnih vrijednosti je nešto veća, no i dalje je manja od 0.2 (Slika 20).

Kako su neki radovi pokazali da NAO indeks ima puno izraženiji utjecaj tijekom zimskih mjeseci (siječanj, veljača, ožujak), izračunata je i korelacija samo za te mjesece između klimatskih elemenata i NAO indeksa (Slika 21). Korelacija u zimskim mjesecima je nešto viša no i dalje nije značajna.

	OBOR_BJ	OBOR_OG	OBOR_VZ	OBOR_ZM	NAO
OBOR_BJ	1	0.6688	0.8145	0.8196	-0.176
OBOR_OG	0.6688	1	0.6402	0.7076	-0.1046
OBOR_VZ	0.8145	0.6402	1	0.8621	-0.1577
OBOR_ZM	0.8196	0.7076	0.8621	1	-0.1582
NAO	-0.176	-0.1046	-0.1577	-0.1582	1

Slika 20 Korelacija između mjesečnih srednjih vrijednosti

	TEMP_B	TEMP_O	TEMP_V	TEMP_M	TLAK_B	TLAK_O	TLAK_V	TLAK_M	OBOR_B	OBOR_O	OBOR_V	OBOR_M	NAO	MONTH
TEMP_B	1	0.9328	0.971	0.9806	-0.07678	-0.2292	-0.2897	-0.2873	-0.03699	-0.03104	0.0004408	0.004127	0.1465	0.4949
TEMP_O	0.9328	1	0.9325	0.938	-0.05414	-0.2306	-0.2978	-0.2882	-0.06278	-0.08042	-0.03172	-0.02938	0.1408	0.3712
TEMP_V	0.971	0.9325	1	0.965	-0.0624	-0.2029	-0.2708	-0.2622	-0.04153	-0.0464	-0.01516	-0.009216	0.1639	0.4604
TEMP_M	0.9806	0.938	0.965	1	-0.07658	-0.2261	-0.2847	-0.2843	-0.03711	-0.0368	-0.003099	-0.002694	0.154	0.48
TLAK_B	-0.07678	-0.05414	-0.0624	-0.07658	1	0.001154	0.002104	0.005864	0.001251	0.004749	-0.004638	-0.0003776	0.04251	-0.1275
TLAK_O	-0.2292	-0.2306	-0.2029	-0.2261	0.001154	1	0.991	0.9953	-0.2485	-0.3039	-0.2605	-0.28	0.2921	-0.1076
TLAK_V	-0.2897	-0.2978	-0.2708	-0.2847	0.002104	0.991	1	0.997	-0.2375	-0.2871	-0.2497	-0.271	0.2794	-0.1194
TLAK_M	-0.2873	-0.2882	-0.2622	-0.2843	0.005864	0.9953	0.997	1	-0.2441	-0.299	-0.2575	-0.2771	0.2779	-0.1317
OBOR_B	-0.03699	-0.06278	-0.04153	-0.03711	0.001251	-0.2485	-0.2375	-0.2441	1	0.6699	0.7597	0.8099	-0.08188	0.003999
OBOR_O	-0.03104	-0.08042	-0.0464	-0.0368	0.004749	-0.3039	-0.2871	-0.299	0.6699	1	0.653	0.742	-0.105	-0.004798
OBOR_V	0.0004408	-0.03172	-0.01516	-0.003099	-0.004638	-0.2605	-0.2497	-0.2575	0.7597	0.653	1	0.8157	-0.06796	0.02217
OBOR_M	0.004127	-0.02938	-0.009216	-0.002694	-0.0003776	-0.28	-0.271	-0.2771	0.8099	0.742	0.8157	1	-0.07451	0.01172
NAO	0.1465	0.1408	0.1639	0.154	0.04251	0.2921	0.2794	0.2779	-0.08188	-0.105	-0.06796	-0.07451	1	0.06781
MONTH	0.4949	0.3712	0.4604	0.48	-0.1275	-0.1076	-0.1194	-0.1317	0.003999	-0.004798	0.02217	0.01172	0.06781	1

Slika 21 Korelacija za zimske mjesec

4.4. Opis modela

Kao rezultat ovog rada kreirani su modeli koji pokušavaju predvidjeti temperaturu, tlak zraka i količinu padalina za postaje Bjelovar, Ogulin, Varaždin i Zagreb Maksimir gdje je svaki klimatski element predstavljen vlastitim modelom.

Za odabir parametara neuronske mreže kao što su broj skrivenih slojeva, broj neurona u skrivenim slojevima te koliko prethodnih vrijednosti neuronska mreža koristi za predviđanje sljedeće korišten je algoritam *grid search*. *Grid search* kao ulaze prima listu vrijednosti parametara neuronske mreže te nakon toga kreira model za sve moguće kombinacije tih parametara koje zatim evaluira. Kako je u ovom slučaju korištena srednja kvadratna pogreška kao mjera kvalitete modela, najbolji model je onaj koji ima najmanji RMSE (Tablica 5 i Tablica 6). Izvođenje ovog algoritma zahtjeva podosta vremena te je pretraživanje parametara izvedeno za model koji predviđa temperaturu i model koji predviđa mjesečnu količinu padalina ta manji broj epoha učenja (50 epoha).

Iz rezultata je vidljivo da najbolje karakteristike za predviđanje temperature ima mreža s 2 skrivena sloja s po 3 neurona, koja koristi vrijednosti od prethodna 3 dana za predviđanje današnjih, dok najgore karakteristike pokazuje mreža sa 3 skrivena sloja sa po jednim neuronom koja koristi vrijednosti prethodna 2 dana.

Tablica 5 Rezultati grid-search algoritma za model temperature

Broj skrivenih slojeva	Broj neurona u sloju	Broj prethodnih vrijednosti koje uzimamo u obzir				
		1	2	3	4	5
1	1	2,743892	2,856749	2,675020	2,958220	2,834081
	2	2,685199	2,679172	2,768156	2,646503	2,727446
	3	2,695986	2,898722	2,967479	2,650874	2,739048
2	1	3,270988	3,231839	3,107185	3,133479	3,177182
	2	2,941098	2,750001	2,736569	2,680858	2,742242
	3	2,953672	2,782441	2,631468	2,639401	2,662969
3	1	3,510905	3,687054	3,527185	3,103099	3,683493
	2	3,119703	3,038731	3,056706	2,861663	2,884615
	3	3,106767	3,022104	2,720567	2,799905	2,850699

Rezultati *grid search* algoritma za model koji predviđa količinu padalina pokazuju da najbolje rezultate daje model kojeg generira mreža sa 5 skrivenih slojeva sa po 3 neurona koristeći vrijednosti za proteklih 12 mjeseci a najgore mreža sa 5 skrivenih slojeva sa 6 neurona koja koristi vrijednosti za proteklih 12 mjeseci.

Tablica 6 Rezultati grid-search algoritma za model oborina

Broj skrivenih slojeva	Broj neurona u sloju	Broj prethodnih vrijednosti koje uzimamo u obzir	
		6	12
1	1	2,750967	2,721905
	2	2,661470	2,756776
	3	2,656235	2,839604
2	1	2,770719	2,730519
	2	2,730070	2,837953
	3	2,656093	2,609993
3	1	2,667062	2,707643
	2	2,680170	2,719287
	3	2,605624	2,782975
	6	2,656426	2,693188
	12	2,631996	3,062160
4	3	2,688855	2,833907
	4	2,618412	2,941307
	5	2,715935	2,485544
	6	2,637082	2,727496
	12	2,568916	2,518310
5	3	2,664630	2,472579

	4	2,628080	2,768094
	5	2,608972	2,630099
	6	2,518967	3,017222
	12	2,639209	-

4.5. Karakteristični dijelovi koda

Kako bi neuronska mreža mogla naučiti ovisnosti između ulaznih vremenskih serija, prije treniranja potrebno je vremensku seriju pretvoriti u klasičan problem nadziranog učenja ([13],[14],[15]) (Kod 1). Funkcija kao argumente prihvaća vrijednosti vremenske serije (polje ili `DataFrame`) i broj prethodnih vrijednosti koje treba uzeti u obzir. Funkcija vraća `DataFrame` koji sadrži retke s očekivanim (one vrijednosti koje bi neuronska mreža trebala dati na izlazu ako je dobro naučila) i ulaznim vrijednostima (vrijednosti koje služe za predviđanje) koje će biti korištene za treniranje mreže.

```
def transform_ts_to_supervised(values, lag):
    in_df = pd.DataFrame(values)
    out_df = in_df.shift(periods=1).fillna(0)
    for l in range(2, lag + 1):
        shifted_df = in_df.shift(periods=l).fillna(0)
        out_df = pd.concat([out_df, shifted_df], axis=1)
    out_df = pd.concat([in_df, out_df], axis=1)
    num_features = in_df.shape[1]
    new_names = np.array([[str(out_df.columns[j]) + '(t-' +
str(i) + ')'] for j in range(num_features)] for i in range(0,
lag+1)]).flatten()
    out_df.columns = new_names
    return out_df
```

Kod 1 Funkcija za pretvaranje vremenske serije u problem nadziranog učenja

Nakon pretvaranja vremenske serije u problem nadziranog učenja, potrebno je skup podataka podijeliti na skup za treniranje (otprilike 70% podataka) i skup za testiranje (otprilike 30% podataka).

Za ocjenjivanje kreiranog modela potrebno ga je usporediti s drugim modelom. Zbog toga se kreira osnovna (*engl. baseline*) prognoza. Osnovna prognoza je najčešće veoma jednostavna, bez ikakvih kompleksnih računanja ili treniranja neuronske mreže. U ovom slučaju model osnovne prognoze predviđa današnju vrijednost kao jučerašnju. Kod koji kreira osnovni model dan je u nastavku (Kod 2). Funkcija prihvaća skup za treniranje i skup za testiranje kao argumente, te vraća listu predviđenih vrijednosti i izračunatu srednju kvadratnu pogrešku.

```
def baseline_predictor(train, test):
    history_base = [x for x in train]
    predictions_base = []
    for i in range(len(test)):
        predictions_base.append(history_base[-1])
        history_base.append(test[i])

    rmse_bp = np.sqrt(mean_squared_error(y_true=test,
                                          y_pred=predictions_base))
    return predictions_base, rmse_bp
```

Kod 2 Kod za kreiranje osnovnog modela

LSTM očekuje da ulazni podatci budu unutar područja aktivacijske funkcije. Zbog toga je podatke prije dovođenja na ulaz potrebno skalirati unutar određenog intervala. To obavlja funkcija `scale()`. Za skaliranje koristi klasu `MinMaxScaler` iz paketa `sklearn.preprocessing`. Funkcija kao argumente prima vrijednosti skupova za treniranje i testiranje i raspon unutar kojeg će se nalaziti skalirane vrijednosti te vraća skalirane vrijednosti (Kod 3). Raspon je u ovom slučaju od -1 do 1 jer je korištena aktivacijska funkcija tangens hiperbolni.

```
def scale(train_values, test_values, feature_range=(-1, 1)):
    scaler = MinMaxScaler(feature_range=feature_range)
    scaler = scaler.fit(train_values)
```

```
scaled_train_values = scaler.transform(train_values)
scaled_test_values = scaler.transform(test_values)

return scaled_train_values, scaled_test_values, scaler
```

Kod 3 Funkcija za skaliranje

Nakon skaliranja podataka možemo kreirati neuronsku mrežu (Kod 4). Prvo se kreira `Sequential` model, odnosno model s linearnim stogom slojeva. Nakon toga modelu se dodaju skriveni slojevi metodom `add()`. U ovom slučaju dodaje se LSTM sloj koji kao argumente prima broj neurona, oblik ulaza, aktivacijsku funkciju, inicijalizator težina veza te oznaku vraća li zadnju vrijednost izlaza ili čitav niz izlaza. Na kraju se dodaje izlazni sloj s jednim neuronom.

```
model_lstm=Sequential()
model_lstm.add(LSTM(num_units, input_shape=(batch_size,
input_dim), activation='tanh',
kernel_initializer='glorot_normal', return_sequences=True))
model_lstm.add(LSTM(num_units, input_shape=(batch_size,
input_dim), activation='tanh',
kernel_initializer='glorot_normal', return_sequences=False))
model_lstm.add(Dense(1))
```

Kod 4 Kreiranje slojeva

Neuronsku mrežu konfiguriramo sljedećom linijom koda:

```
model_lstm.compile(loss='mean_squared_error',optimizer='adam')
```

Metoda `compile()` prima dva argumenta, funkciju gubitka koju nastoji minimizirati i optimizacijski postupak koji se koristi prilikom optimizacije.

Treniranje modela započnemo metodom `fit()`. Metoda kao argumente prima listu NumPy polja s ulaznim podacima, listu NumPy polja s očekivanim vrijednostima, broj epoha (iteracija) na kojima izvodi treniranje, količinu kontrolnog ispisa tijekom treniranja, dovode li se podatci na ulaz slučajnim redoslijedom i listu funkcija koje se pozivaju prilikom treniranja. Funkcija vraća objekt koji sadrži polje pogrešaka prilikom treniranja.

```
history_model_lstm=model_lstm.fit(train_X, train_Y,  
epochs=num_epochs, batch_size=batch_size, verbose=1,  
shuffle=False, callbacks=[early_stop])
```

Predviđanje sljedeće vrijednosti izvodi se metodom `predict()`. Ona kao argument prima skup za testiranje a kao rezultat vraća polje predviđenih vrijednosti.

```
y_pred_test_lstm=model_lstm.predict(test_X)
```

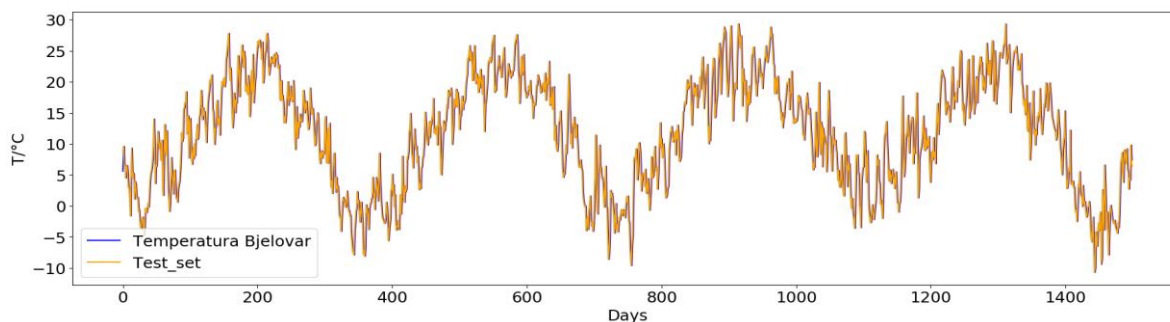
U nastavku će biti prikazani modeli za svaki pojedini klimatski element za svaku od četiri navedene postaje (osnovni model i model dobiven treniranjem neuronske mreže). Kriterij zaustavljanja treniranja neuronske mreže je broj iteracija, u ovom slučaju 100, ili ako vrijednost funkcije gubitka (*engl. loss function*) počne rasti. Model predviđa vrijednosti jedan dan unaprijed.

4.6. Modeli prognoze klimatskih elemenata mjernih postaja

U ovom poglavlju bit će prikazani kreirani modeli za 4 navedene mjerne postaje. Njihova kvaliteta biti će uspoređena sa osnovnim modelima temeljem korijena srednje kvadratne pogreške.

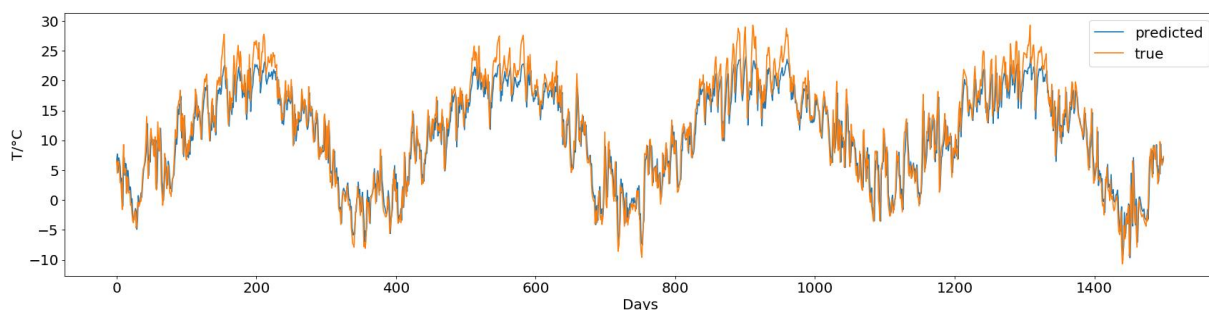
4.6.1. Bjelovar

Slika 22 predstavlja osnovni model za predviđanje temperature u Bjelovaru za prvu godinu predviđanja. Narančasta linija na grafu predstavlja stvarne vrijednosti, a plava linija predstavlja predviđene vrijednosti. U slučaju temperature ovaj model ne predstavlja pretjerano loše rješenje zbog toga što temperatura u Republici Hrvatskoj nema velike dnevne skokove (za razliku od nekih mjesta na svijetu gdje postoje velike dnevne razlike u temperaturi npr. pustinje, ovdje su razlike mnogo manje), pa je i prognoza dobra. RMSE osnovnog modela iznosi 2,575508.



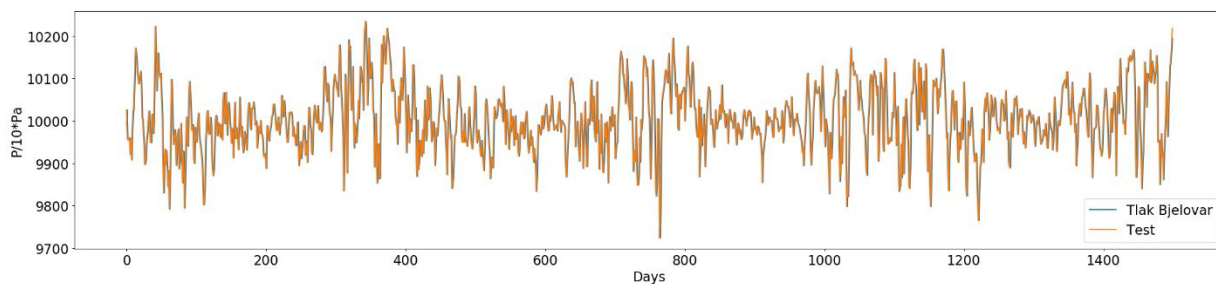
Slika 22 Osnovni model temperature u Bjelovaru

Model koji predviđa dnevne vrijednosti temperature za mjernu postaju Bjelovar za sljedeći dan prikazan je u nastavku (Slika 23). Prikazani su rezultati prognoze za prvih pet godina. Model uzima u obzir vrijednosti temperature i NAO indeksa za prethodna tri dana. Plava linija na slici predstavlja predviđene vrijednosti, a narančasta stvarne vrijednosti. Vidljivo je da model dobro predviđa tijekom zimskih mjeseci no prilikom ljetnih mjeseci malo podcjenjuje vrijednosti. Ako uzmemo u obzir prirodu NAO indeksa pokazanu u drugim radovima ([3]), odnosno činjenicu da NAO ima veći utjecaj na Republiku Hrvatsku zimi nego ljeti, jasno je zašto model bolje predviđa zimska razdoblja. RMSE ovog modela iznosi 2,669446. Osnovni model pokazuje nešto bolje performanse od kreiranog modela.



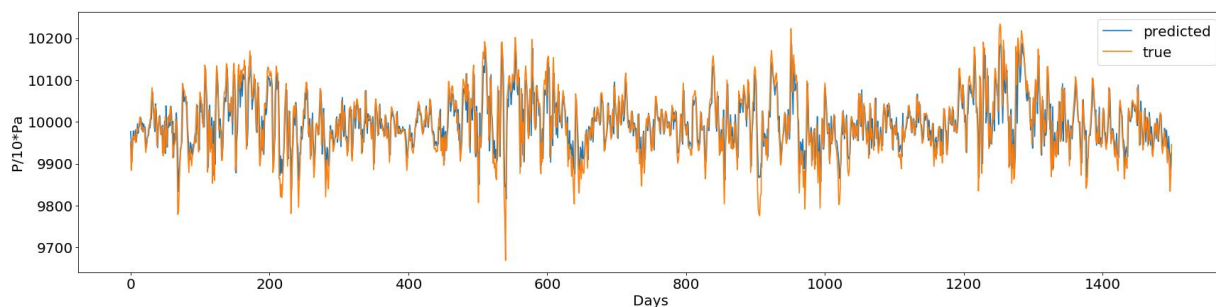
Slika 23 Model temperature za Bjelovar

U nastavku se nalazi osnovni model prognoze tlaka (Slika 24) za prvu godinu predviđanja. Način prognoziranja je istovjetan onom za temperaturu. Plava linija predstavlja predviđene vrijednosti a narančasta stvarne. RMSE ovog modela iznosi 46,321394.



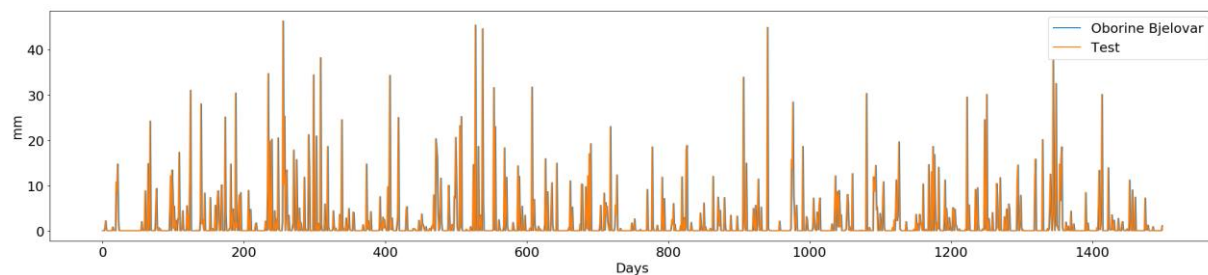
Slika 24 Osnovni model tlaka u Bjelovaru

Model prognoziranja dnevnih vrijednosti tlaka za mjernu postaju Bjelovar prikazuje Slika 25. Plava linija predstavlja predviđene vrijednosti, a narančasta stvarne. Kako je NAO indeks računat kao razlika u tlaku dalo bi se zaključiti kako će najveći utjecaj imati upravo na tlak u Republici Hrvatskoj. RMSE ovog modela iznosi 42,162026. Kreirani model daje nešto bolju prognozu od osnovnog modela.



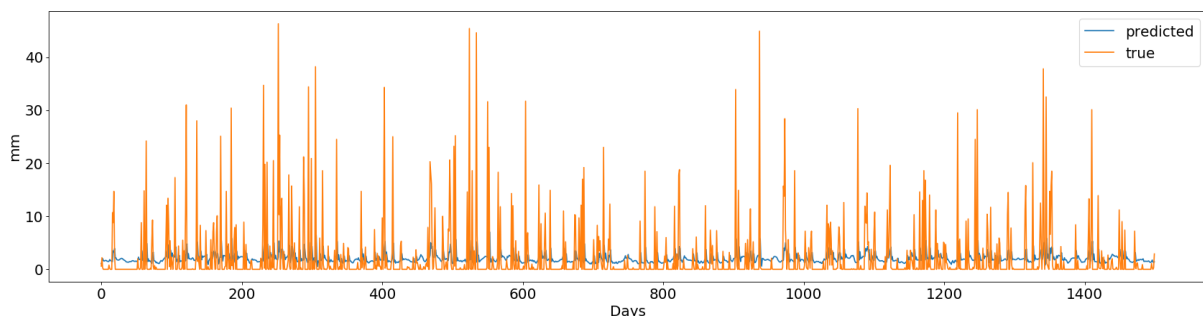
Slika 25 Model prognoze tlaka za Bjelovar

Osnovni model predviđanja dnevnih vrijednosti oborina prikazan je u nastavku (Slika 26) za prvu godinu predviđanja. Plava linija prikazuje predviđene vrijednosti, a narančasta stvarne. RMSE ovog modela iznosi 7,346129.



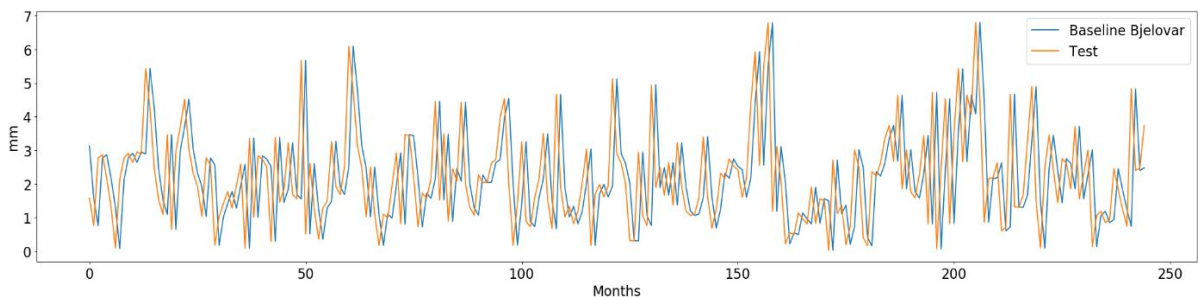
Slika 26 Osnovni model prognoze oborina

Model prognoziranja dnevnih vrijednosti oborina za mjernu postaju Bjelovar prikazuje Slika 27. Plava linija predstavlja predviđene vrijednosti, a narančasta stvarne. Vidljivo je da model ne prognozira dobro. Skup podataka o oborinama sadrži puno mjerenja čija vrijednost iznosi 0, te količina oborina u danima može biti izrazito različita. Neuronska mreža zbog toga teško može steći neko znanje kako bi predviđela oborine.



Slika 27 Model prognoze dnevnih vrijednosti oborina za Bjelovar

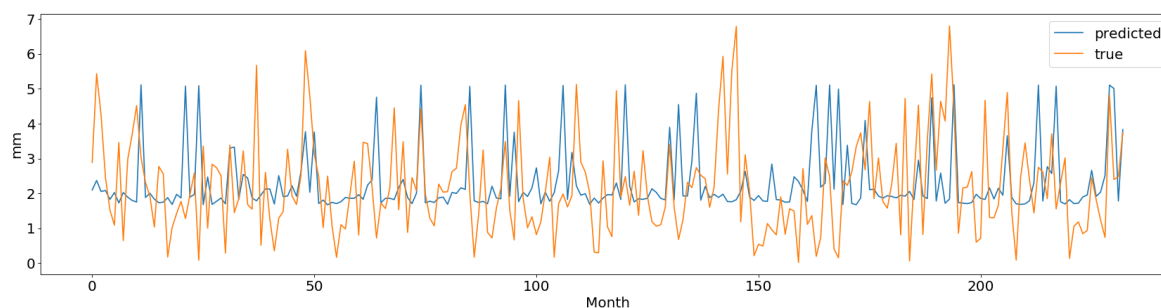
Zbog toga ima više smisla promatrati mjesečne prosječne vrijednosti količine padalina i srednje vrijednosti NAO indeksa. U odnosu na NAO indeks i prethodne vrijednosti količine padalina pokušavamo predvidjeti količinu padalina u sljedećem mjesecu. Slika 28 Prikazuje osnovni model prognoze. Taj model je kreiran na identičan način kao i gore opisani osnovni modeli a njegov RMSE iznosi 1,736111.



Slika 28 Osnovni model prognoze po mjesecima

Model dobiven treniranjem neuronske mreže prikazuje Slika 29. Neuronska mreža korištena za treniranje ovog modela ima 5 skrivenih slojeva, svaki s po 3 neurona, te koristi vrijednosti NAO indeksa i padalina za prethodnih 12 mjeseci. Plava linija prikazuje predviđene vrijednosti, a narančasta stvarne. Vidljivo je da model ne predviđa najbolje u slučaju mjeseci s izrazito niskom ili visokom količinom padalina, dok su ostale

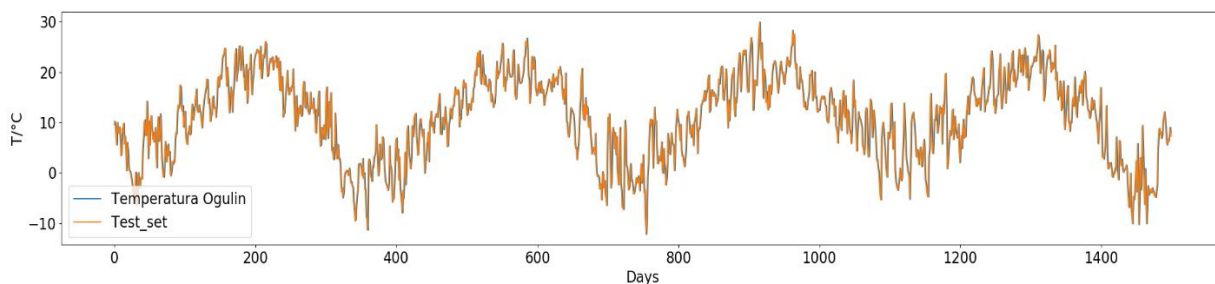
predviđene vrijednosti zadovoljivo dobre. RMSE ovog modela je 1,460221 što znači da kreirani model pokazuje bolje karakteristike od osnovnog modela.



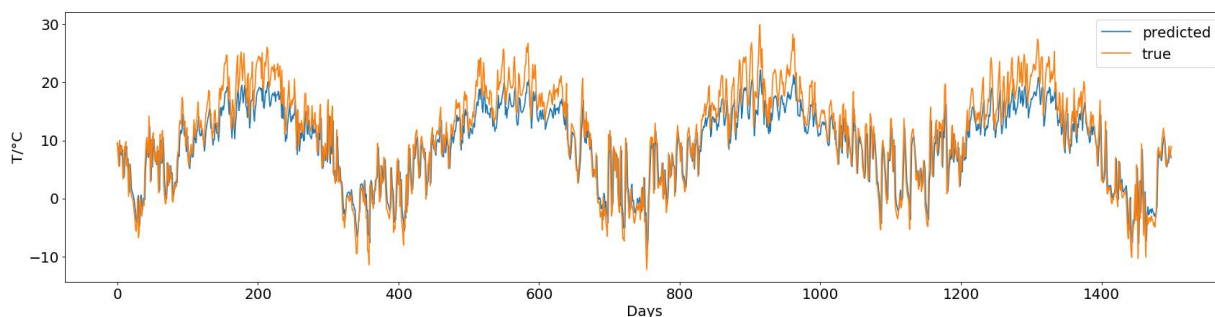
Slika 29 Model prognoze padalina po mjesecima

4.6.2. Ogulin

U nastavku su dani samo rezultati za mjernu postaju Ogulin. Arhitektura neuronske mreže identična je onoj za mjernu postaju Bjelovar. Slika 30 i Slika 31 prikazuju osnovni i kreirani model prognoze temperature. Plava linija na obje slike predstavlja predviđene vrijednosti, a narančasta stvarne. RMSE osnovnog modela iznosi 2,967644, a RMSE kreiranog modela iznosi 3,008457, odnosno, osnovni model je nešto bolji u odnosu na kreirani model.

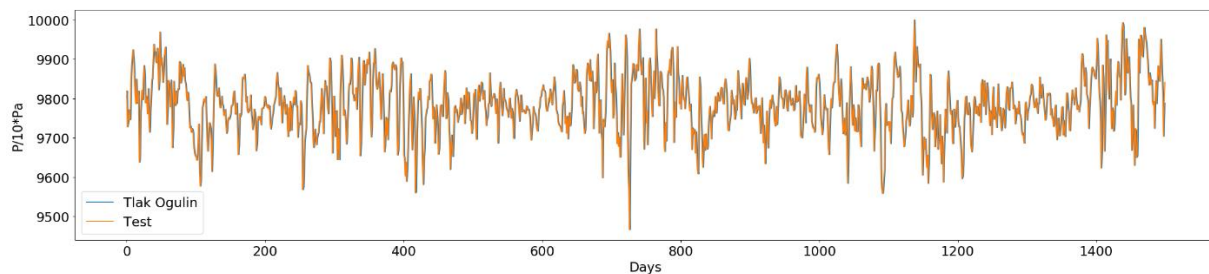


Slika 30 Osnovni model temperature za mjernu postaju Ogulin

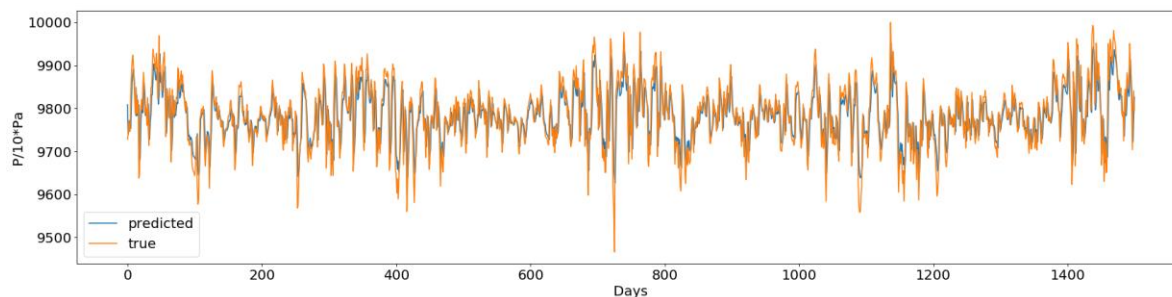


Slika 31 Model temperature za Ogulin

Modele (osnovni i kreirani) prognoze tlaka prikazuju Slika 32 i Slika 33. Ponovno, plava linija predstavlja predviđene vrijednosti, a narančasta stvarne. RMSE osnovnog modela iznosi 44,834838, a RMSE kreiranog modela 40,649936. Kreirani model pokazuje nešto bolje performanse od osnovnog modela.

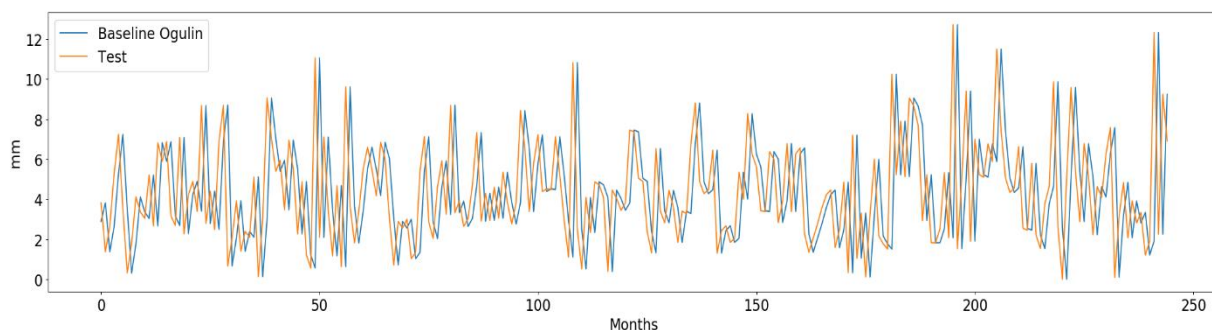


Slika 32 Osnovni model prognoze tlaka za Ogulin

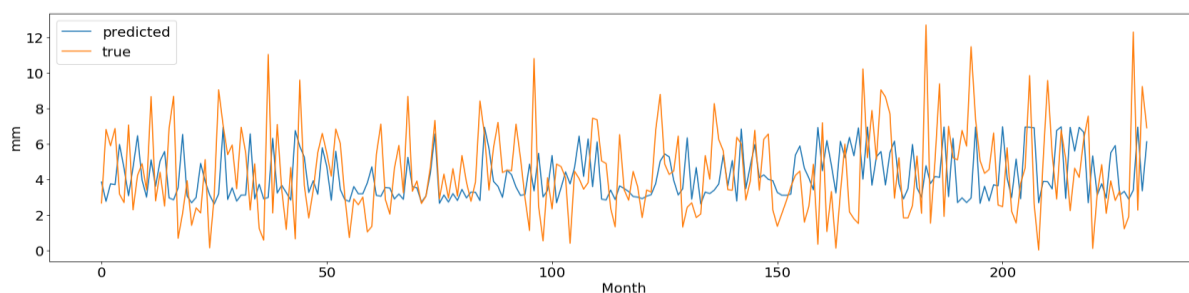


Slika 33 Kreirani model prognoze tlaka za Ogulin

Također je kreiran i model prognoze mjesečne količine padalina. RMSE osnovnog modela (Slika 34) iznosi 3,456581, a RMSE kreiranog modela (Slika 35) iznosi 2,709978.



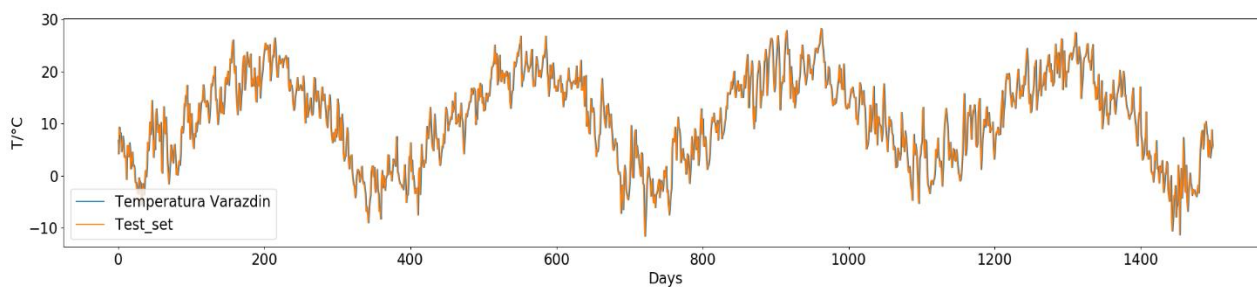
Slika 34 Osnovni model padalina za Ogulin



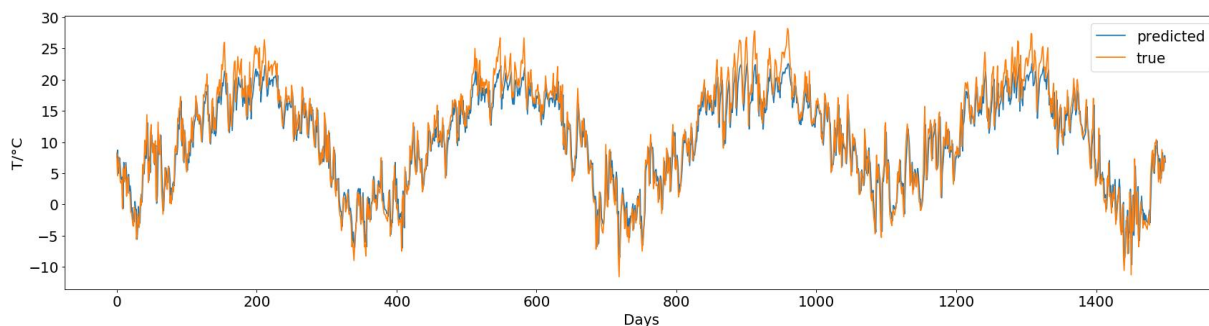
Slika 35 Kreirani model padalina za Ogulin

4.6.3. Varaždin

Osnovni i kreirani model prognoze temperature prikazuju Slika 36 i Slika 37. Plava linija predstavlja predviđene vrijednosti a narančasta stvarne. RMSE osnovnog modela iznosi 2,625346, a RMSE kreiranog modela iznosi 2,830756. Osnovni model pokazuje nešto bolje performanse od kreiranog modela.

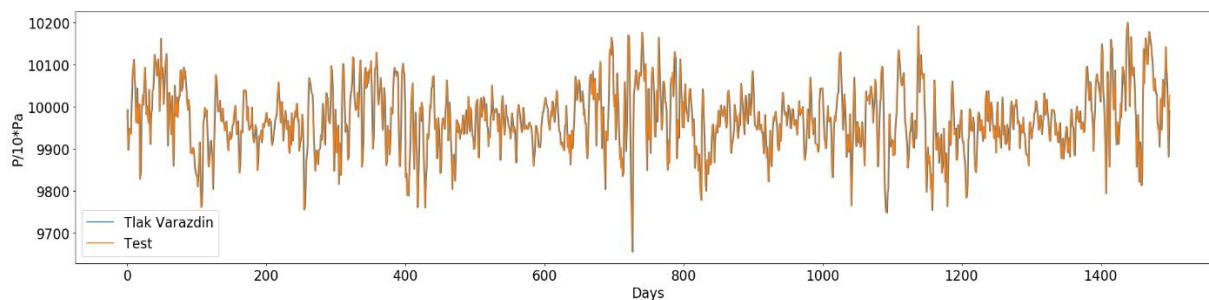


Slika 36 Osnovni model prognoze temperature za Varaždin

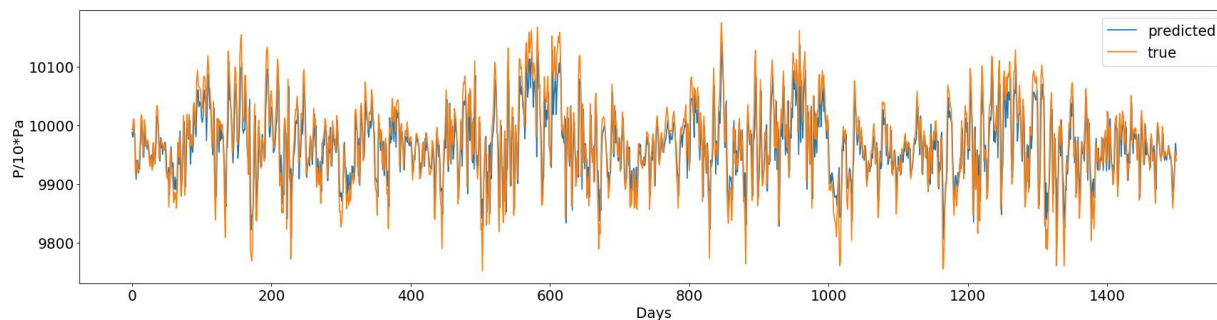


Slika 37 Kreirani model prognoze temperature za Varaždin

Osnovni model prognoze tlaka prikazuje Slika 38. Plava linija predstavlja predviđene vrijednosti, a narančasta stvarne. RMSE tog modela iznosi 46,649085. Slika 39 prikazuje kreirani model prognoze tlaka za mjernu postaju Varaždin. RMSE tog modela iznosi 42,581022. Kreirani model pokazuje nešto bolje performanse od osnovnog modela.



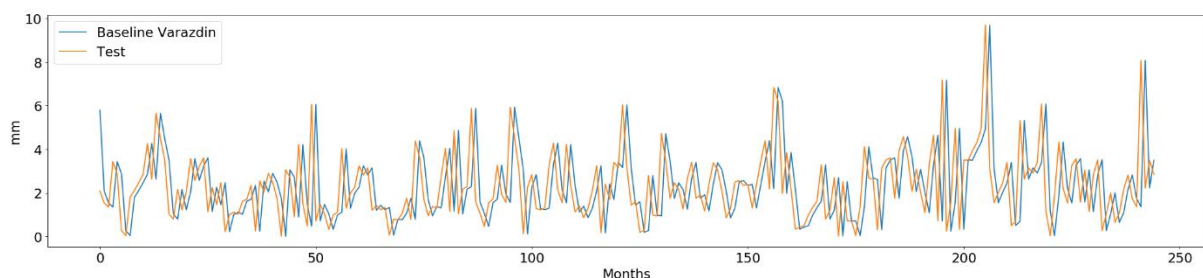
Slika 38 Osnovni model prognoze tlaka za Varaždin



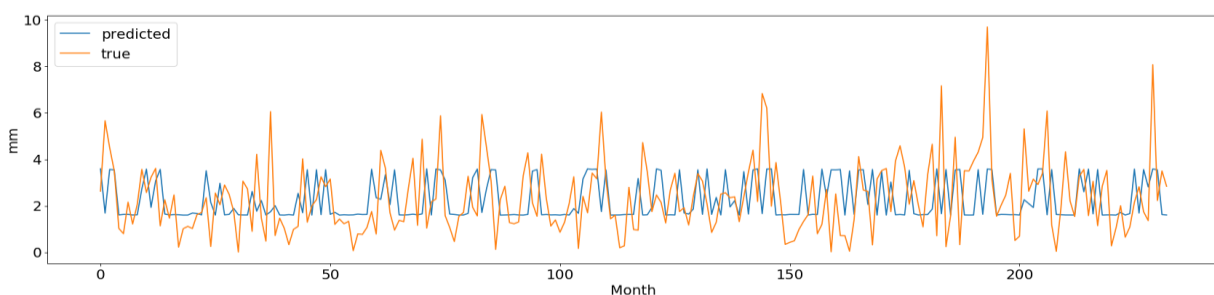
Slika 39 Kreirani model prognoze tlaka za Varaždin

Osnovni model prognoze mjesečne količine padalina prikazuje Slika 40. Plava linija prikazuje predviđene vrijednosti, a narančasta stvarne. RMSE osnovnog modela iznosi

2,022397. Slika 41 prikazuje kreirani model prognoze mjesečne količine padalina za mjernu postaju Varaždin. RMSE ovog modela iznosi 1,756721.



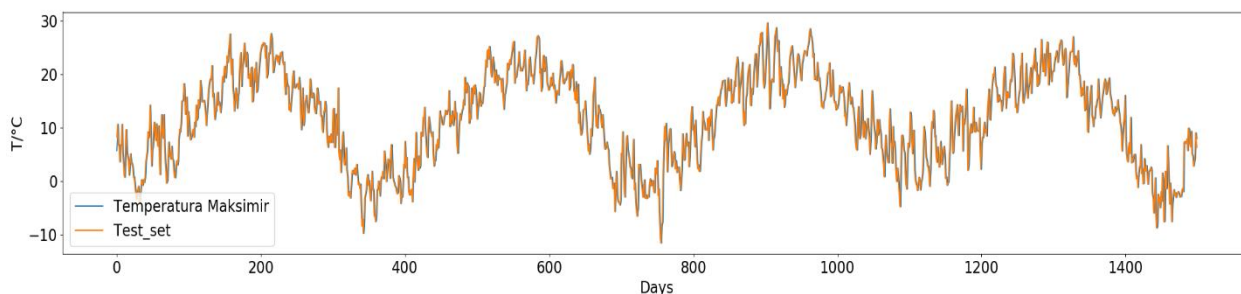
Slika 40 Osnovni model prognoze padalina za Varaždin



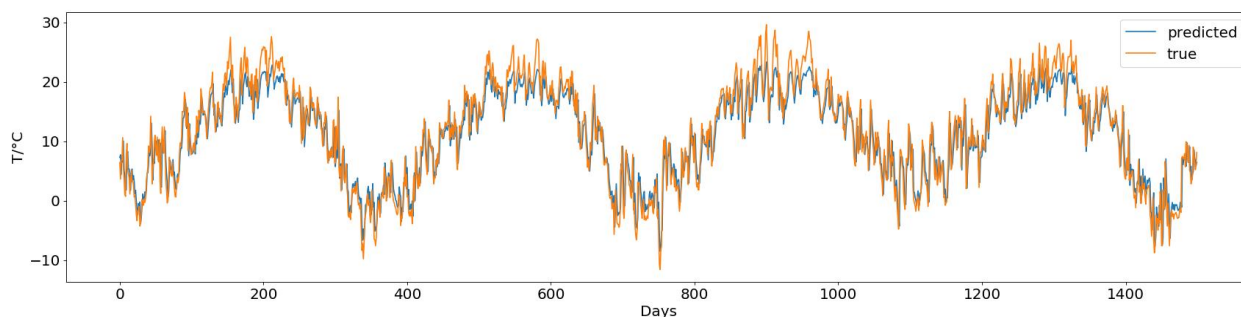
Slika 41 Kreirani model prognoze padalina za Varaždin

4.6.4. Zagreb Maksimir

Osnovni model prognoze temperature prikazuje Slika 42. Plava linija prikazuje predviđene vrijednosti, a narančasta stvarne. RMSE osnovnog modela iznosi 2,601042. Slika 43 prikazuje kreirani model prognoze temperature za mjernu postaju Zagreb Maksimir. RMSE tog modela iznosi 2,685509. Kreirani model ima nešto lošije performanse od osnovnog modela.

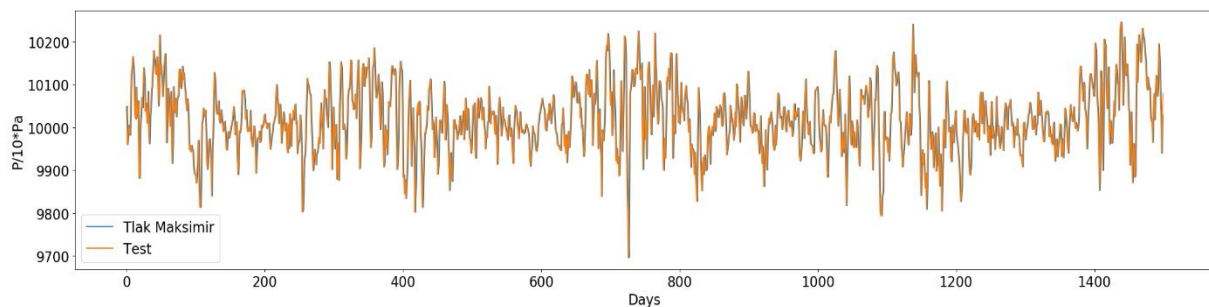


Slika 42 Osnovni model prognoze temperature za postaju Zagreb Maksimir

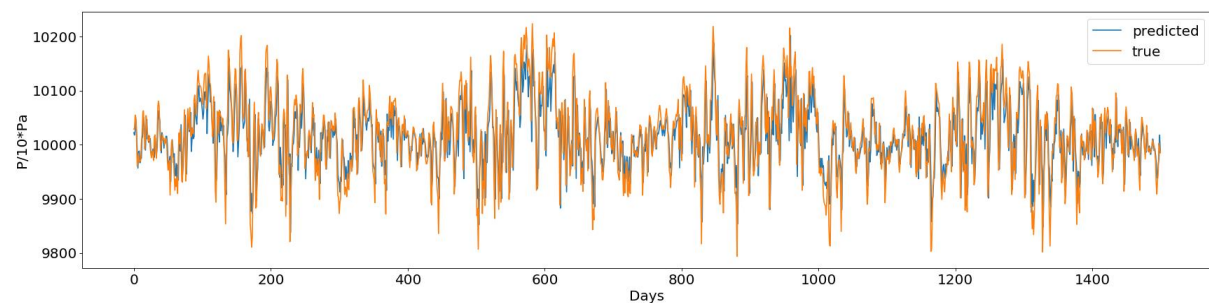


Slika 43 Kreirani model prognoze temperature za mjernu postaju Zagreb Maksimir

Slika 44 prikazuje osnovni model prognoze tlaka za mjernu postaju Zagreb Maksimir. Plava linija predstavlja predviđene vrijednosti, a narančasta stvarne. RMSE osnovnog modela iznosi 46,414833. Kreirani model prikazuje Slika 45. RMSE tog modela iznosi 41,986122.

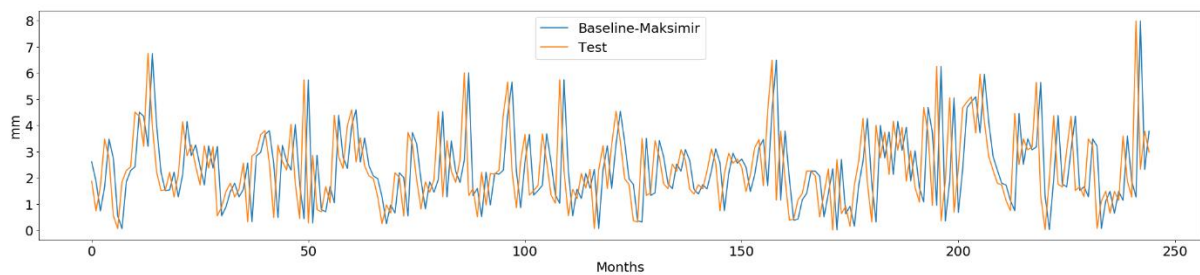


Slika 44 Osnovni model prognoze tlaka za mjernu postaju Zagreb Maksimir

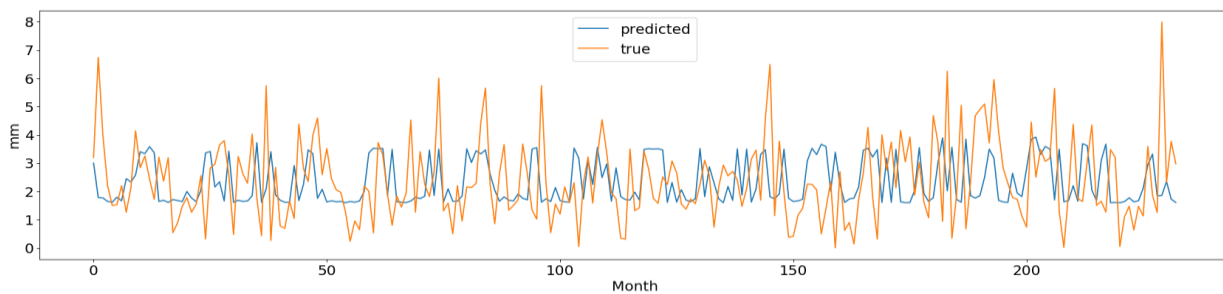


Slika 45 Kreirani model prognoze tlaka za mjernu postaju Zagreb Maksimir

Osnovni model prognoze mjesečne količine padalina za mjernu postaju Zagreb Maksimir prikazuje Slika 46. RMSE ovog modela iznosi 1,872176. Slika 47 prikazuje kreirani model za prognozu mjesečnih vrijednosti padalina. RMSE kreiranog modela iznosi 1,679157. Kreirani model pokazuje bolje performanse od osnovnog modela.



Slika 46 Osnovni model prognoze oborina za postaju Zagreb Maksimir



Slika 47 Kreirani model prognoze oborina za mjernu postaju Zagreb Maksimir

5. Zaključak

U ovom radu testirana je korelacija između vremenskih serija NAO indeksa i klimatskih elemenata u Republici Hrvatskoj (temperatura, tlak i padaline). Pokazano je da ne postoji značajna korelacija između NAO indeksa i vremenskih serija klimatskih elemenata. Korišteni su podatci o vremenskim elementima i NAO indeksu od prvog siječnja 1950. godine kao ulazni skup podataka kod treniranja neuronske mreže i stvaranja modela za prognoziranje za 4 mjerne postaje u Republici Hrvatskoj. Korištene su dvije arhitekture neuronskih mreža. Arhitektura mreže za pojedinu seriju određena je algoritmom *grid search*. Kod prognoziranja temperature i tlaka korištene su dnevne vrijednosti za prethodna 3 dana kao ulazni uzorci za treniranje mreže s 2 skrivena sloja s po 3 neurona. Modeli koji su kreirani pokazuju dobre performanse u odnosu na osnovne modele. Kreirani modeli za temperaturu i tlak pokazuju bolje performanse u zimskim mjesecima od ljetnih. Prognoza oborina koristi prosječne mjesečne vrijednosti i prosječnu vrijednost NAO indeksa. Neuronska mreža sadrži 5 skrivenih slojeva s po dva neurona te koristi prosječne vrijednosti proteklih 12 mjeseci kao ulaze. I ovaj model pokazuje bolje performanse od osnovnog modela, međutim greške ovog modela su nešto veće i nisu strogo vezane za jedno godišnje doba. Ovaj model također ne prognozira najbolje za mjesece sa malom prosječnom količinom padalina. Ako promatramo postaje, najbolje prognoze dobivamo za mjernu postaju Bjelovar, tj. za mjernu postaju Bjelovar dobivamo najmanje greške modela.

Modeli bi se mogli poboljšati korištenjem veće količine ulaznih podataka ili dodatnim vremenskim serijama pojava koje utječu na vrijeme u Republici Hrvatskoj. Valjalo bi razmotriti i utjecaje ostalih oscilacija na vrijeme kako bi prognoza potencijalno postala bolja, no i ovako su dobiveni zadovoljavajući rezultati.

Literatura

- [1] *Global and European temperature*, <https://www.eea.europa.eu/data-and-maps/indicators/global-and-european-temperature/global-and-european-temperature-assessment-3>, 04.06.2019.
- [2] *Mean precipitation*, <https://www.eea.europa.eu/data-and-maps/indicators/european-precipitation-2/assessment>, 16.06.2019.
- [3] Born K., Fink A., Knippertz P., I-5.2: Meteorological processes influencing the weather and climate of Morocco, 2010.
- [4] Turkes M., Erhat E., Climatological responses of winter precipitation in Turkey to variability of the North Atlantic Oscillation during the period 1930–2001, 2005., *Climatological responses of winter precipitation in Turkey to variability of the North Atlantic Oscillation during the period 1930–2001*, <https://link.springer.com/article/10.1007/s00704-004-0084-1> ,13.04.2019.
- [5] Cherry J., Cullen H., Visbeck M., Small A., Uvo C., Impacts of the North Atlantic Oscillation on Scandinavian Hydropower Production and Energy Markets, 2005., *Impacts of the North Atlantic Oscillation on Scandinavian Hydropower Production and Energy Markets*, <https://link.springer.com/article/10.1007/s11269-005-3279-z>, 15.04.2019
- [6] Lorenzo-Lacruz J., Vicente-Serrano S. M., López-Moreno J. I., González-Hidalgo J. C., Morán-Tejeda E., The response of Iberian rivers to the North Atlantic Oscillation, *Hydrol. Earth Syst. Sci.*, 15, 2581-2597, <https://doi.org/10.5194/hess-15-2581-2011>, 2011.
- [7] Muić I., UTJECAJ SJEVERNO-ATLANTSKE OSCILACIJE I EL NIÑO-JUŽNE OSCILACIJE NA KLIMATSKU VARIJABILNOST EUROPE, diplomski rad, Sveučilište u Zagrebu, Prirodoslovno-matematički fakultet, 2018.
- [8] https://sites.google.com/site/whythe2009winterissocold/_/rsrc/1263077412719/httpsitesgooglecomsitewhythe2009winterissocold/Climate-Variablity-PAV.gif.png, 04.06.2019.
- [9] https://blogwillis-zippykid.netdna-ssl.com/wp-content/uploads/2016/12/Windstorms-Figure-1_645x400.jpg, 01.06.2019
- [10] Hurrell J. W., Kushnir Y., Visbeck M., Ottersen G., 2003: An overview of the North Atlantic Oscillation., In: Hurrell J. W., Kushnir Y., Visbeck M., Ottersen G. (Eds.), *The North Atlantic Oscillation, climatic significance and environmental impact.*, AGU Geophysical Monograph 134: 1-35.
- [11] Marshall J., Kushnir Y., Battisti D., Chang P., Czaja A., Dickson R., Hurrell J., McCartney M., Saravanan R., and Visbeck M., 2001, 'North Atlantic climate variability: Phenomena, impacts, and mechanisms', *Int. J. Climatol.* 21, 1863–1898

- [12] Goodfellow I, Bengio Y., Courville A., Deep learning, Massachusetts Institute of Technology, 2016
- [13] Brownlee J., Time Series Forecasting with the Long Short-Term Memory Network in Python, 07.04.2017., *Time Series Forecasting with the Long Short-Term Memory Network in Python*, <https://machinelearningmastery.com/time-series-forecasting-long-short-term-memory-network-python/>, 25.05.2019
- [14] Brownlee J., Multi-step Time Series Forecasting with Long Short-Term Memory Networks in Python, 10.05.2017., *Multi-step Time Series Forecasting with Long Short-Term Memory Networks in Python*, <https://machinelearningmastery.com/multi-step-time-series-forecasting-long-short-term-memory-networks-python/>, 04.06.2019
- [15] Brownlee J., Multivariate Time Series Forecasting with LSTMs in Keras, 14.08.2017., *Multivariate Time Series Forecasting with LSTMs in Keras*, <https://machinelearningmastery.com/multivariate-time-series-forecasting-lstms-keras/>, 30.05.2019
- [16] Brownlee J., Time Series Forecasting as Supervised Learning, 14.08.2017., *Time Series Forecasting as Supervised Learning*, <https://machinelearningmastery.com/time-series-forecasting-supervised-learning/>, 01.06.2019
- [17] Brockwell P.J., Davis R.A., „Introduction to Time Series and Forecasting“, treće izdanje, Švicarska, Springer, 2016.
- [18] Adhikari R.; Agrawal R.K.; „An Introductory Study on Time Series Modeling and Forecasting“, 2013., <https://arxiv.org/abs/1302.6613>, 15.04.2019.
- [19] Bašić B.D., Čupić M., Šnajder J., „Umjetne neuronske mreže“, Zagreb; svibanj 2008
- [20] Stilinović T., Herceg-Bulić I., Vučetić V., UTJECAJ ZIMSKE SJEVERNO-ATLANTSKE OSCILACIJE NA PROLJETNE TEMPERATURE TLA U HRVATSKOJ, Hrvatski meteorološki Časopis - Croatian Meteorological Journal, 48/49, 2013/14, 37-45
- [21] Dertat, A., ‘Applied Deep Learning - Part 1: Artificial Neural Networks’, 8.8.2017., <https://towardsdatascience.com/applied-deep-learning-part-1-artificial-neural-networks-d7834f67a4f6>, *Applied Deep Learning - Part 1: Artificial Neural Networks*, 28.05.2019.
- [22] <https://www.esrl.noaa.gov/psd/data/correlation>, 08.06.2019.
- [23] Long short-term memory, https://en.wikipedia.org/wiki/Long_short-term_memory, *Long short-term memory*, 29.05.2019.
- [24] Backpropagation, *Brilliant.org*, <https://brilliant.org/wiki/backpropagation>, 19.06.2019.

Sažetak

Prognoza vremenskih serija klimatskih elemenata i pokazatelja

U sklopu ovog rada ukratko je objašnjeno što je točno vremenska serija te što su neuronske mreže. Pokazano je kakav utjecaj ima sjeverno atlantska oscilacija na vrijeme u Republici Hrvatskoj, odnosno postoji li korelacija između NAO indeksa i vrijednosti temperature, tlaka i padalina. Kreiran je model za predviđanje sutrašnjih vrijednosti pomoću LSTM neuronske mreže za 4 mjerne postaje u Republici Hrvatskoj: Bjelovar, Ogulin, Varaždin i Zagreb Maksimir. Pokazano je da kreirani model ima dobre karakteristike kod predviđanja dnevnih vrijednosti temperature i tlaka, dok model koji predviđa dnevnu količinu padalina ne daje dobre rezultate, te je zbog toga kreiran model koji koristi prosječne mjesečne vrijednosti.

Ključne riječi: NAO, LSTM, klimatski elementi, vremenska serija

Summary

Time series forecasting of climate elements and indicators

Within this paper, it is briefly explained what are time series and neural networks. The influence of North Atlantic Oscillation on weather in Republic of Croatia has been shown, ie whether there is a correlation between the NAO index and temperature, pressure and precipitation. A model for prediction of next day's values was created with LSTM neural network for 4 measuring stations in the Republic of Croatia: Bjelovar, Ogulin, Varaždin and Zagreb Maksimir. It has been shown that the model created has good characteristics when predicting daily values of temperature and pressure, while a model predicting daily precipitation amounts does not produce good results and therefore a model that uses average monthly values has been created.

Key words: NAO, LSTM, climate elements, time series

Skraćenice

NAO	<i>North Atlantic Oscillation</i>	sjeverno-atlantska oscilacija
LSTM	<i>Long Short-Term Memory</i>	duga kratkoročna memorija
RNN	<i>Recurrent Neural Network</i>	povratna neuronska mreža
NN	<i>Neural Network</i>	neuronska mreža
RMSE	<i>Root Mean Squared Error</i>	srednja kvadratna pogreška
MAE	<i>Mean Absolute Error</i>	srednja apsolutna pogreška
MSE	<i>Mean Squared Error</i>	srednja kvadratna pogreška
MAPE	<i>Mean Absolute Percentage Error</i>	srednja apsolutna postotna pogreška
CPU	<i>Central Processing Unit</i>	središnja procesorska jedinica (procesor)
GPU	<i>Graphics Processing Unit</i>	grafički procesor
AO	<i>Arctic Oscillation</i>	Arktička oscilacija
AAO	<i>Antarctic Oscillation</i>	Antarktička oscilacija
SO	<i>Southern Oscillation</i>	Južna oscilacija
MJO	<i>Madden and Julian Oscillation</i>	Madden i Julian oscilacija

Privitak A

Upute za pokretanje kreiranih *Jupyter* bilježnica

U procesu izrade ovog rada kreirano je nekoliko *Jupyter* bilježnica koje služe za kreiranje i iscrtavanje modela. Pokretanje kreiranih bilježnica moguće je korištenjem besplatne usluge *Google Colaboratory*. Nije potrebna dodatna instalacija nikakvih alata. Za korištenje je dovoljno posjedovati bilo koji Internet preglednik.

Prikaz modela odnosno rješenja moguć je putem sljedećih bilježnica:

- **Diplomski_corr.ipynb**
Bilježnica za izračunavanje korelacije i stacionarnosti vremenskih serija
- **Monthy.ipynb**
Bilježnica za iscrtavanje i kreiranje modela mjesečnih vrijednosti oborina
- **Grid_search.ipynb**
Bilježnica za izvođenje algoritma *grid search*
- **Bjelovar.ipynb**
Bilježnica za kreiranje modela za mjernu postaju Bjelovar
- **Ogulin.ipynb**
Bilježnica za kreiranje modela za mjernu postaju Ogulin
- **Varazdin.ipynb**
Bilježnica za kreiranje modela za mjernu postaju Varaždin
- **Maksimir.ipynb**
Bilježnica za kreiranje modela za mjernu postaju Zagreb Maksimir

Sve bilježnice se nalaze u direktoriju „/Code“ na priloženom mediju. Ulazni podatci nalaze se na istom mediju u direktoriju „/data“. Podatci o temperaturi se nalaze u direktoriju „/data/Temperature“, podatci o tlaku u direktoriju „/data/Pressure“, podatci o oborinama u direktoriju „/data/Precipitation“ i podatci o NAO indeksu u direktoriju „/data/NAO_index“.