Izrada platforme za predviđanje proizvodnje solarnih elektrana u Hrvatskoj primjenom strojnog učenja

Radičević, Andro

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 649

IZRADA PLATFORME ZA PREDVIĐANJE PROIZVODNJE SOLARNIH ELEKTRANA U HRVATSKOJ PRIMJENOM STROJNOG UČENJA

Andro Radičević

Zagreb, lipanj 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 649

IZRADA PLATFORME ZA PREDVIĐANJE PROIZVODNJE SOLARNIH ELEKTRANA U HRVATSKOJ PRIMJENOM STROJNOG UČENJA

Andro Radičević

Zagreb, lipanj 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Zagreb, 4. ožujka 2024.

DIPLOMSKI ZADATAK br. 649

Pristupnik:	**Andro Radičević (0036522355)**
Studij:	Računarstvo
Profil:	Znanost o podacima
Mentor:	nasl. prof. dr. sc. Dubravko Sabolić
Komentor:	dr. sc. Bojan Franc

Zadatak: **Izrada platforme za predviđanje proizvodnje solarnih elektrana u Hrvatskoj primjenom strojnog učenja**

Opis zadatka:

Potrebno je načiniti platformu za predviđanje proizvodnje solarnih elektrana u Hrvatskoj. Osnovna funkcionalnost platforme je izrada i prikazivanje prognoze proizvodnje solarnih elektrana u Hrvatskoj dan unaprijed pomoću modela strojnog učenja na bazi povijesnih i meteoroloških podataka. Prognoze i realizirane vrijednosti, nakon što budu dostupne, trebaju biti prikazane prikladnom vizualizacijom i tablično, a također je potrebno omogućiti njihov izvoz u .csv formatu. Korisnik treba moći odabrati između prognoza različitih modela unutar platforme, te istovremeno uspoređivati rezultate raznih modela. Pomoću birača datuma trebaju se moći odabirati prošli datumi te se za njih trebaju vidjeti podaci o prognozi proizvodnje za taj dan, kao i podaci o realizaciji. Osim osnovne funkcionalnosti izrade i prikaza prognoza, platforma također treba omogućiti usporedbe performansi različitih modela. Korisnici će moći pomoću prikladnih vizualizacija i metrika usporediti točnost različitih modela kako bi mogli odabrati najprikladniji za svoje potrebe. Platforma će također trebati imati prikladan sustav registracije i prijave korisnika kako bi se korisnici autentificirali i autorizirali za pristup podacima koji će biti dostupni unutar platforme. Fokus zadatka za rad studenta na ovom diplomskom radu biti će izrada i testiranje različitih modela strojnog učenja za predviđanje proizvodnje solarnih elektrana u Hrvatskoj u programskom jeziku Python, koristeći pritom tipične Python biblioteke za strojno učenje kao što su Pandas, Numpy i Scikit-learn, te integracija tih modela s ostatkom platforme.

Rok za predaju rada: 28. lipnja 2024.

Zahvale

Hvala mojoj obitelji na stalnoj i velikodušnoj podršci tijekom studiranja.

Također, zahvaljujem svojem mentoru dr. sc. Dubravku Saboliću, komentoru dr. sc. Bojanu Francu te svim djelatnicima HOPS-a i FER-a koji su mi pomogli pri izradi ovog rada.

Sadržaj:

Uvod

Jedna od važnijih uloga operatora elektroenergetskog prijenosnog sustava, uključujući i Hrvatskog operatora prijenosnog sustava (HOPS) je održavanje konstantne frekvencije električne energije. Radi postizanja tog cilja od velike važnosti je osigurati da je u svakom trenutku proizvodnja električne energije jednaka potražnji, što se postiže kroz aktivacije ili deaktivacije kapaciteta elektrana namijenjenih za tu svrhu, tzv. rezerve. Ako se dogode prevelika odstupanja proizvodnje od potražnje, može doći do problema u elektroenergetskom sustavu, kako su uređaji koji koriste električnu energiju namijenjeni za rad na uskom pojasu frekvencije. (HOPS d.d., 2024) (ESO, 2024)

Ovaj problem dodatno kompliciraju obnovljivi izvori energije, čiju proizvodnju je teže unaprijed planirati kako ovisi o prirodnim pojavama kao npr. brzini i smjeru vjetra za vjetroelektrane ili jačini sunčevog zračenja i naoblaci za solarne elektrane. Ovi izvori energije su također često decentralizirani, što otežava upravljanje njima i nemaju svojstva inercije koja ublažuje promjene u frekvenciji. Kako su obnovljivi izvori već jako zastupljeni u elektroenergetskoj mreži, i očekuje se da im zastupljenost i važnost samo raste s vremenom, rješavanje navedenih problema bit će jako bitno za učinkovito upravljanje elektroenergetskim sustavom u budućnosti. (Muelaner, 2021) (ESO, 2024)

U ovome radu se istražuje mogućnost kratkoročnog predviđanja proizvodnje jedne solarne elektrane za tri dana unaprijed pomoću modela strojnog učenja na temelju podataka o njenoj proizvodnji iz prošlosti i vremenskoj prognozi. Modeliranje proizvodnje ove solarne elektrane može olakšati upravljanje i planiranje vezano za ovu elektranu i služiti kao primjer na temelju kojeg se mogu izraditi modeli za ostale solarne elektrane u budućnosti. Model čija je izrada opisana u ovom radu je samo jedna komponenta u sklopu planirane šire platforme za predviđanje proizvodnje solarnih elektrana u Hrvatskoj, koja bi trebala uključivati i modele za ostale elektrane.

U prvom poglavlju ovog rada je kratak teoretski prikaz polja strojnog učenja, te obrazloženja ključnih pojmova koji se koriste u radu. Zatim su prikazani podaci na temelju kojih su istrenirani modeli, te analiza i obrada tih podataka. Nakon analize podataka objašnjen je postupak odabira modela korištenih u ovom radu, teorija iza njih te su prikazani rezultati modela i ograničenja u modeliranju. Konačno, u diskusiji su komentirani rezultati u kontekstu ograničenja te priložene smjernice za daljnje istraživanje.

1. Strojno učenje

Strojno učenje je proces izrade algoritama koji se za funkcioniranje oslanjaju na prethodno postojeći skup primjera, odnosno skup podataka za učenje. Na temelju tih primjera model može dati korisne rezultate predviđanja na još neviđenim primjerima. (Burkov, 2019, poglavlje 1.1)

U ovom poglavlju nalazi se kratak pregled polja strojnog učenja i objasnit će se ključna načela i pojmovi s prikladnim ilustracijama i primjerima te s naglaskom na područja koja su relevantna za ovaj rad.

1.1. Podjela modela strojnog učenja

Modeli strojnog učenja se mogu dijeliti na više načina. Jedan od temeljnih načina podjele je prema vrsti ciljne značajke. Ako je ciljna značajka skup klasa ili oznaka (eng. *labels*) koje se pridružuju ostalim značajkama riječ je o problemu klasifikacije. Primjer ovoga problema može biti npr. razvrstavanje elektroničke pošte u željenu i neželjenu, ili dijagnosticiranje je li pacijent bolestan ili zdrav. Ako je ciljna značajka broj onda je riječ o problemu regresije. Bitno je napomenuti da se ovdje ne misli o broju kao o oznaci, nego kao uređenom skupu. Oznake za klasifikaciju također mogu biti brojevi, npr. 0 i 1, no ne podrazumijeva se uređenost između tih oznaka. Problem opisan u ovom radu, predviđanje proizvodnje solarne elektrane, spada u područje regresije zato što je ciljna značajka, proizvodnja solarne elektrane u MW, pozitivan realni broj. (Burkov, 2019, poglavlje 2.7.)

Strojno učenje se također može podijeliti i prema vrsti učenja. Ako je dostupan skup već označenih podataka $\{(x_i, y_i)\}_{i=1}^{N}$, te je cilj naučiti kako predvidjeti ciljnu značajku y_i na temelju odgovarajućih značajki x_i riječ je o nadziranom učenju (eng. *supervised learning*). Problem opisan u ovom radu spada u tu kategoriju, zato što su dostupni povijesni podaci o proizvodnji koji služe kao ciljna oznaka . Ako podaci nemaju neku oznaku koju nam je bitno predvidjeti, već su samo oblika $\{x_i\}_{i=1}^{N}$ riječ je o nenadziranom učenju (eng. *unspervised learning*). Tipičan primjer nenadziranog učenja je grupiranje, gdje se istražuje postoje li kakve pravilnosti u podacima u vidu značajki koje često dolaze zajedno, npr. možemo li grupirati kupce u trgovini u tipične grupe prema proizvodima koje kupuju. Također postoje i polunadzirano učenje (eng. *semi-supervised learning*) gdje su ulazni podaci kombinacija označenih i neoznačenih, te podržano učenje (eng. *reinforcement learning*) gdje je model postavljen u neku okolinu, može izvršavati radnje koje donose nagradu ili kaznu a cilj mu je naučiti optimalno birati radnje, tj. napraviti politiku (eng. *policy*) koju radnju da donese u kojim uvjetima da maksimizira očekivanu nagradu. (Burkov, 2019, poglavlje 1.2.)

1.2. Nadzirano učenje

Modeli nadziranog učenja, koji su najzanimljiviji za potrebe ovog rada, generalno funkcioniraju na sljedećem načelu: definira se model, koji na temelju poznatih značajki i parametara predviđa ciljnu značajku. Npr. jedan od jednostavnijih modela strojnog učenja, model linearne regresije modelira ciljnu značajku na način opisan izrazom (1):

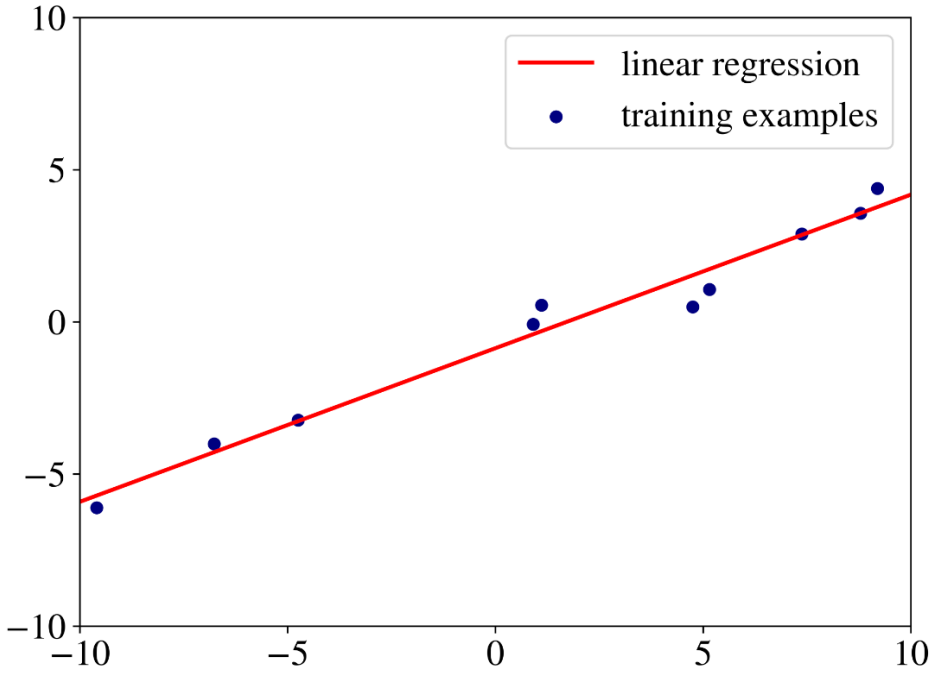
$$f_{w_1,w_2 \dots w_n,b}(x_1, x_2 \dots x_n) = w_1 x_1 + w_2 x_2 + \cdots + w_n x_n + b \tag{1}$$

Gdje je f funkcija čiji izlaz ovisi o parametrima $w_1, w_2 \dots w_n$, ulaznim značajkama nekog primjera $x_1, x_2 \dots x_n$ te nezavisnom parametru b (eng. *intercept*). Izlaz funkcije bi trebao biti što bliži ciljnoj značajki y za odgovarajući primjer. Sažetiji zapis opisan je izrazom (2):

$$f_{\boldsymbol{a},b}(x) = \boldsymbol{w}\boldsymbol{x} + b \tag{2}$$

U ovom zapisu $\boldsymbol{w}$ označava vektor parametara, $\boldsymbol{x}$ vektor ulaznih značajki, a ostale oznake su jednake kao i u prethodnoj definiciji (Burkov, 2019, poglavlja. 1.3. i 3.1.).

Ono što se jednadžbom linearne regresije postiže je aproksimiranje ciljne značajke linearnom kombinacijom ulaznih značajki. Ako imamo samo jednu ulaznu značajku, ovisnost će se aproksimirati pravcem. Za dvije značajke ovisnost će se aproksimirati ravninom, a za više od dvije hiperravninom. Jasno je da je ovo prilično jednostavan model, kako nisu sve ovisnosti u prirodi linearne te zato ovaj model nije prikladan za modeliranje kompleksnijih pojava. Vizualizacija linearne regresije prikazana je na slici Slika 1.1Slika 1.1



Slika 1.1 Primjer linearne regresije sa jednom ulaznom značajkom (Burkov, 2019, poglavlje 3.1.)

Zatim se definira funkcija greške, kojom se računa točnost modela u svrhu optimizacije. Za model linearne regresije to je srednja kvadratna pogreška (eng. *mean squared error*) koja je opisana izrazom (3):

$$MSE_{\boldsymbol{w},b}(x,y) = \frac{1}{N}\sum_{i=1\ldots N}(f_{\boldsymbol{w},b}(x_i) - y_i)^2 \tag{3}$$

Vidi se da je ovaj izraz veći što su veće razlike između predviđanja modela $f_{w,b}(x_i)$ i stvarnih vrijednosti y_i, a manji što su te razlike manje. Također možemo vidjeti da se razlika kvadrira. Razlog tomu je da bi se bez kvadrata u sumi greške u kojima je $f_{w,b}(x_i)$ veće od y_i, što čini cijeli rezultat unutarnjeg izraza pozitivnim međusobno poništavale s greškama u kojima je rezultat izraza negativan. S obzirom na to da želimo penalizirati obje vrste grešaka ovaj pristup se izbjegava. Također, kvadriranje će rezultirati time da su veće greške više penalizirane te se cijeli izraz greške još uvijek može derivirati, što nije slučaj kod npr. apsolutne vrijednosti. Iako je istina da bi druge funkcije greške također mogle imati slične karakteristike (npr. zamijeniti potenciju 2 s potencijom 4 u izrazu) navedeni izraz se najčešće koristi zbog navedenih pozitivnih karakteristika i jednostavnosti (Burkov, 2019, poglavlje 3.1.).

Konačno, optimizacijskim postupkom otkrivaju se parametri koji će dati optimalno rješenje na podacima za učenje mjereno odabranom funkcijom pogreške. U ovom slučaju funkcija greške je konkavna, što znači da ima jedan globalni minimum, i može se dobiti egzaktno rješenje tako da se funkcija greške derivira po parametrima $\boldsymbol{w}$ i $\boldsymbol{b}$, što rezultira brzom i lakom optimizacijom, no to nije slučaj kod svih modela i funkcija pogreške. Alternativno se optimizacija obavlja iterativnim postupkom kao što je npr. gradijentni spust, gdje se funkcija greške parcijalno derivira po svim parametrima te se parametri ažuriraju nakon svake iteracije u smjeru koji smanjuje grešku. Ovaj proces može naći zadovoljavajući lokalni minimum čak i ako funkcija greške nije konkavna. (Burkov, 2019, poglavlja 3.1. i 4.2.)

Također je bitno napomenuti da neki modeli imaju i hiperparametre. Hiperparametri su parametri koji se ne optimiziraju putem postupka optimizacije nego se ručno definiraju a mogu se optimizirati iscrpnom pretragom različitih opcija. Linearna regresija u ovom primjeru nema hiperparametara, no neki drugi modeli koji se koriste u ovom radu će ih imati. (Burkov, 2019, poglavlje 2.6.)

1.3. Inženjerstvo značajki

Inženjerstvo značajki (eng. *feature engineering*) je proces transformacije sirovih podataka u značajke korisne za predviđanje. Najčešće nisu sve dostupne značajke korisne za model, te je potrebno odabrati koje će se značajke koristiti na temelju relevantnosti za predviđanje ciljne značajke. Također, postoje razne transformacije nad podacima koje ih mogu učiniti korisnijim za model. (Burkov, 2019, poglavlje 5.1.)

Jedna od mogućih transformacija je normalizacija. Normalizacija pretvara raspon varijable u raspon od 0 do 1, gdje je 0 minimalna vrijednost, a 1 maksimalna vrijednost. Formula za normalizaciju MinMax korištenu u ovom radu dana je u izrazu (4):

$$\bar{x}_j = \frac{x_j - min_j}{max_j - min_j} \tag{4}$$

gdje je $\bar{x}_j$ normalizirana vrijednost primjera iz neke značajke, x_j originalna vrijednost, min_j minimalna vrijednost te značajke a max_j maksimalna. Normalizacija može biti korisna za neke modele, kao npr. model linearne regresije objašnjen u poglavlju 1.2. koji su osjetljivi na raspon ulaznih značajki. Općenito je poželjno izbjeći da veličina parametra pridijeljenog nekoj značajki ovisi o rasponu a ne o bitnosti parametra. Također, ako se model prilagođava značajkama različitih skala može doći do numeričkih problema u računalu u slučaju prilagođavanja velikim razlikama u skalama (eng. *overflow*). (Burkov, 2019, poglavlje 5.1.3.)

Alternativa normalizaciji je standardizacija, koja pretvara srednju vrijednost značajke u 0 i standardnu devijaciju u 1. Vrijednost značajke nakon normalizacije zapravo pokazuje koliko je standardnih devijacija vrijednost udaljena od sredine, i na koju stranu. Formula za standardizaciju je prikazana u jednadžbi (5):

$$\hat{x}_j = \frac{x_j - \mu_j}{\sigma_j} \tag{5}$$

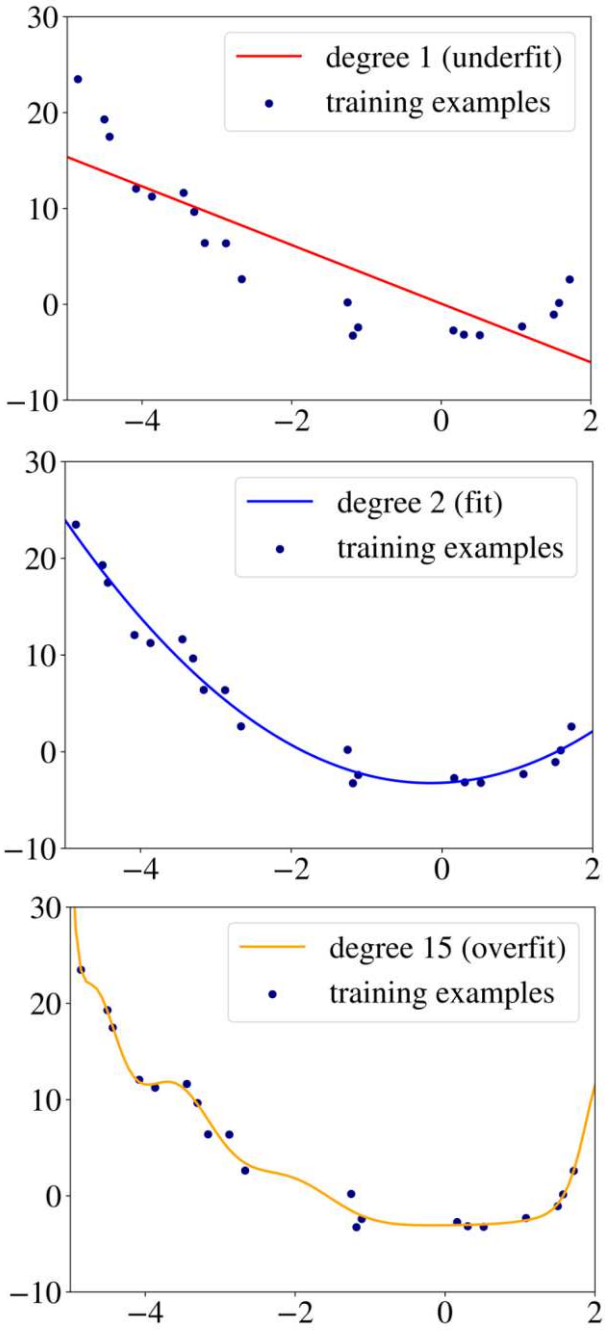
gdje je $\hat{x}_j$ standardizirana vrijednost primjera iz neke značajke x_j originalna vrijednost, μ_j srednja vrijednost te značajke a σ_j standardna devijacija te značajke. Standardizaciju je preporučljivo koristiti ako su značajke distribuirane slično normalnoj distribuciji, ili mogu imati ekstremno visoke ili niske vrijednosti, inače preferiramo normalizaciju. (Burkov, 2019, poglavlje 5.1.4.)

Još jedan oblik inženjerstva značajki relevantan za ovaj rad je dodavanje polinomijalnih značajki. Ova transformacija dodaje sve polinomijalne kombinacije ulaznih značajki stupnja manjeg ili jednakog definiranom. Npr. ako su dostupne značajke $\boldsymbol{a}$ i $\boldsymbol{b}$ i stupanj je 2 dodale bi se značajke $\boldsymbol{a^2}, \boldsymbol{b^2}$ i $\boldsymbol{ab}$. Na ovaj način može se povećati kompleksnost modela. Ako na primjer na linearnoj regresiji s jednom značajkom napravimo polinomijalnu transformaciju drugog stupnja modelirat će ovisnost u obliku kvadratne krivulje a ne pravca. (Scikit learn, 2024)

1.4. Prenaučenost i podnaučenost

Prenaučenost je čest problem koji se javlja kod modela strojnog učenja. Do prenaučenosti dolazi kada model nauči i šum u ulaznim podacima, osim samo generalnih pravilnosti. Šum može biti posljedica npr. ostalih čimbenika koji utječu na rezultat a nisu prisutni u ulaznim

značajkama, pogreška u mjerenju, slučajnosti itd. Prenaučeni model će imati skoro savršen rezultat predviđanja na istim podacima na kojima je naučen, no skoro sigurno loš rezultat na novim ulaznim podacima. Do prenaučenosti može doći pretjeranim povećanjem kompleksnosti modela, npr. dodavanjem polinomijalnih značajki vrlo visokog stupnja. Suprotnost prenaučenosti je podnaučenost, gdje model ne nauči ni generalne pravilnosti u podacima zbog npr. premale količine podataka ili nedovoljne kompleksnosti modela. Takav model ima loše rezultate i na ulaznim podacima na kojima je naučen i na novim podacima. Cilj u strojnom učenju je izraditi model koji nije ni podnaučen ni prenaučen. Za takav model kažemo da dobro generalizira, te bi trebao imati dobar rezultat predviđanja i na podacima na kojima je naučen i na novim podacima. Kako bi se detektirala prenaučenost najčešće se dio dostupnih podataka ne koristi za učenje modela, te se model testira na njima kako bi se vidjelo kakav rezultat daje na još neviđenim podacima iz perspektive modela. Vizualizacija podnaučenosti, prenaučenosti i dobrog modela prikazana je na slici Slika 1.2 (Burkov, 2019, poglavlje 5.4.) (Brownlee, 2019)



Slika 1.2 Vizualizacija podnaučenog, dobrog i prenaučenog modela (Burkov, 2019, poglavlje 5.4.)

1.5. Duboko učenje

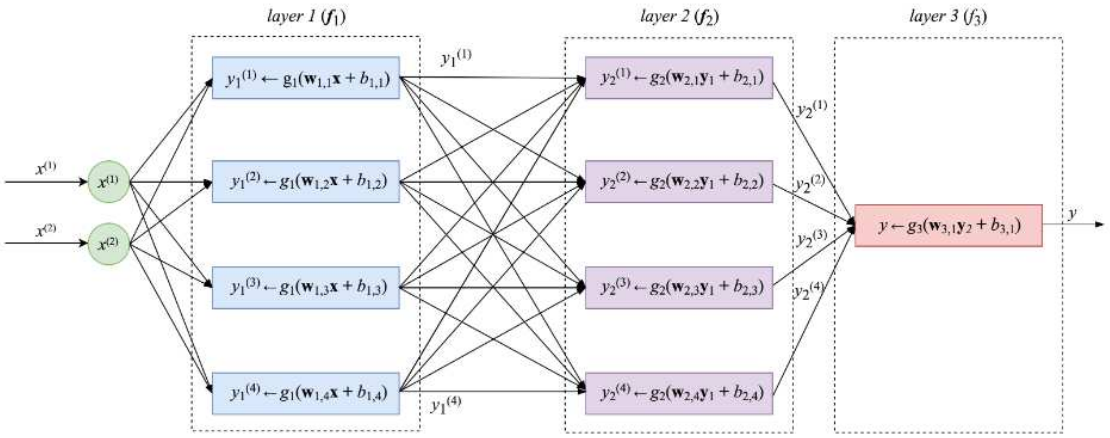
U ovome radu također se koriste neki modeli dubokog učenja. Duboko učenje naziva se tako zato što ima više od jednog sloja između ulaznih podataka i konačnog izlaza. To znači da za razliku od tipičnih modela strojnog učenja u kojima se izlaz može opisati kao funkcija ulaznih značajki i parametara $f_w(x)$, gdje su w parametri a x ulazne značajke, duboko učenje može se opisati kao ugniježđena (eng. *nested*) funkcija u izrazu (6):

$$f_{NN} = f_n(f_{n-1}(\dots f_1(x)))$$

(6)

Gdje f_{NN} označava funkciju dubokog učenja, a $f_1, \dots f_{n-1}, f_n$ slojeve između ulaznih značajki x i izlaza. (Burkov, 2019, poglavlje 6.1.)

Jednostavan primjer modela dubokog učenja je višeslojni perceptron (eng. *Multilayer Perceptron, MLP*). Primjer višeslojnog perceptrona prikazan je na slici Slika 1.3



Slika 1.3 Primjer višeslojnog perceptrona (Burkov, 2019, poglavlje 6.1.)

Ovdje se jasno vidi arhitektura slojeva i kako je ulaz za sljedeće slojeve izlaz prethodnog sloja. Slojevi se sastoje od jedinica (eng. *units*), te svaka jedinica daje jedan izlaz. Jedinice se mogu razlikovati po funkciji kojom računaju izlaz na temelju ulaza, u slučaju višeslojnog perceptrona vidimo da su jedinice slične modelu linearne regresije s dodatkom aktivacijske funkcije g. Aktivacijska funkcija unosi nelinearnost u model dubokog učenja, bez nje bi izlaz u ovom slučaju bio samo linearna kombinacija ulaznih varijabli te višeslojna arhitektura ne bi bila potrebna. Primjer aktivacijske funkcije je ReLU (eng. *Rectified Linear Unit*) funkcija definirana jednostavnim izrazom $f(x) = \max(0, x)$. (Burkov, 2019, poglavlje 6.1.)

Modeli dubokog učenja tipično se optimiziraju postupkom unazadne propagacije (eng. *backpropagation*). U ovom postupku parametri svakog sloja se optimiziraju metodom gradijentnog spusta slijedno od izlazne vrijednosti prema ulaznim značajkama, zbog ovog obrnutog puta u odnosu na predviđanje postupak se zove unazadna propagacija. Ovaj postupak je efikasan zato što se izbjegava ponovno računanje međukoraka za ažuriranje parametara slojeva udaljenijih od izlaza, rezultati za slojeve bliže izlazu mogu se iskoristiti kao međukorak. (Burkov, 2019, poglavlje 6.2.)

1.6. Objašnjivost modela

Još jedna značajna tema kod modela strojnog učenja je objašnjivost (eng. *explainability*) ili interpretabilnost (eng. *interpretability*). Kod nekih modela strojnog učenja moguće je da model postane toliko kompleksan da iako daje dobar rezultat, vrlo je teško razumjeti ili objasniti na koji je način model došao do tog rezultata. Ovo je često značajka modela dubokog učenja koji iako imaju visoke performanse u mnogim problemima su također i

slabo objašnjivi. Za razliku od tih modela, jednostavniji modeli kao npr. linearna regresija mogu davati lošije rezultate no ti rezultati su lako razumljivi iz samog modela npr. prema težinama pridruženim pojedinoj značajki. (Burkov, 2019, poglavlje 5.2.)

U ovome radu objašnjivost nije bila navedena kao zahtjev za izrađene modele. Također nisu prisutni razni uvjeti koji bi implicirali potrebu za objašnjivosti, kao odluke modela koje donose posljedice bitne za živote ljudi npr. u zdravstvu. Zato, na problem predviđanja solarnih elektrana primijenjeni su razni modeli koji se razlikuju prema objašnjivosti. Postupak odabira modela bit će opisan u kasnijim poglavljima.

2. Analiza i obrada podataka

Podatke na temelju kojih su izrađeni modeli za ovaj rad dao je HOPS, a sastoje se od povijesnih podataka o proizvodnji jedne solarne elektrane i povijesnih podataka o vremenskoj prognozi na lokaciji te elektrane. U poglavlju ispod opisat će se korištene tehnologije, značenja pojedinih stupaca podataka, tipovi podataka pojedinih stupaca, postupci analize provedeni nad podacima te konačni dobiveni oblik podataka.

2.1. Korištene tehnologije

Obrada i vizualizacija podataka napravljena je u programskom jeziku Python. Za obradu i spajanje podataka korištena je Python biblioteka **Pandas**. Za vizualizacije korištene su Python biblioteke **Matplotlib** i **Seaborn**.

2.2. Podaci o proizvodnji

Za potrebe izrade ovog rada bili su dostupni podaci o povijesnoj proizvodnji za jednu solarnu elektranu, čije se ime neće spominjati u ovom radu radi ugovora o povjerljivosti koji je autor ovog rada potpisao s HOPS-om. Elektrana se nalazi u sjeveroistočnoj Hrvatskoj. Proizvodnja se mjeri u 15 minutnoj rezoluciji, a vrijeme je UTC (eng. *coordinated universal time*). Originalni podaci sadržavali su sljedeće stupce i tipove podataka:

- **POWERPLANT_NAME**, tekst, ime elektrane čija se proizvodnja mjeri
- **DATE**, tekst, datum u kojem je obavljeno mjerenje u formatu yyyy-MM-dd
- **TIME**, tekst, vrijeme u kojem je obavljeno mjerenje u formatu hh:mm:ss
- **HOUR**, cijeli broj, sat u kojem je obavljeno mjerenje
- **QUARTER_NAME**, tekst, 15 minutni interval u kojem je obavljeno mjerenje, format npr. „00-15“ ili „30-45“
- **QTY**, realni broj, količina proizvodnje električne energije u MW

Najstariji podatak o proizvodnji koji je dostavljen bio je iz 2023-05-23, a najnoviji iz 2024-03-01. Analizom podataka također je utvrđeno da su u podacima na određenim mjestima bila prisutna dulja razdoblja od više tjedana u kojima je proizvodnja elektrana bila 0.

Konkretno, ta razdoblja su bila od prvog mjerenja do 7. mj. 2023. g. i tijekom cijelog 10. mj. 2023.g. Zaključeno je da su podaci u tim razdobljima posljedica greške te da nisu korisna informacija za učenje modela. Po ostalim karakteristikama podaci su bili visoke kvalitete i nisu im bila potrebna daljnja pročišćavanja.

Podaci o proizvodnji su također transformirani u oblik prikladniji za daljnju obradu, kako originalno imaju dosta redundantnih stupaca. Nakon obrade stupci iz originalnih podataka su svedeni na sljedeće stupce:

- **power_timestamp**, tekst, datum i vrijeme mjerenja u formatu yyyy-MM-dd hh:mm:ss
- **qty**, realni broj, količina proizvodnje električne energije u MW

2.3. Podaci o vremenskoj prognozi

Podaci o vremenskoj prognozi su izračuni WRF (eng. *Weather Research & Forecasting*) prognostičkog modela za područje solarne elektrane. WRF model simulira atmosferske prilike, te na temelju ulaznih podataka može proizvesti vremensku prognozu za određeno područje (National Center for Atmospheric Research, 2024). Prognoza se izrađuje svakih 6 sati, u 00:00, 06:00, 12:00 i 18:00 za sljedeća 72 sata u satnoj rezoluciji, vrijeme je također u UTC formatu. Pregled stupaca ovih podataka dan je ispod:

- **TOF**, tekst, datum i vrijeme izrade prognoze u formatu yyyy-MM-dd hh:mm:ss
- **vt**, tekst, datum i vrijeme cilja prognoze u formatu yyyy-MM-dd hh:mm:ss
- **barometer**, realni broj, prosječni tlak na razini mora u hPa
- **outtemp**, realni broj, temperatura zraka na visini 2m u °C
- **windspeed**, realni broj, brzina vjetra na visini 10m u m/s
- **winddir**, cijeli broj, smjer puhanja vjetra na visini 10m u °
- **rain**, realni broj, količina padalina tijekom 1h, u mm
- **radiation**, cijeli broj, jačina sunčevog zračenja u W/m^2
- **cloud_cover**, realni broj, pokrov oblaka u %

Ovi su podaci bili vrlo visoke kvalitete i analizom nisu uočeni propusti koji bi zahtijevali pročišćavanje ili promjenu formata podataka, osim rijetke negativne vrijednosti značajke **rain** koja je ispravljena na vrijednost 0. Najranije vrijeme izrade prognoze bilo je 2023-01-01 u 00:00:00 a najkasnije u 2024-03-26 00:00:00, što je omogućilo potpuno pokrivanje dostupnih podataka o proizvodnji za potrebe učenja modela.

2.4. Vremenski nizovi

Za ovaj rad značajna je i tema vremenskih nizova (eng. *time series*). Vremenski niz je oblik podataka u kojem se određeni uzorak prati regularno kroz vrijeme. Podaci u ovom obliku razlikuju se od tipičnog oblika podataka na kojima se treniraju modeli strojnog učenja, u kojima je svaki uzorak, odnosno redak, nezavisan te nema određenog poretka u kojem bi uzorci trebali biti poredani.

Ciljna značajka u ovom radu, proizvodnja električne energije solarne elektrane, je vremenski niz. Rad s podacima u obliku vremenskog niza pruža dodatne mogućnosti za analizu podataka i razvoj modela. Moguće je analizirati na koji način buduće vrijednosti ovise o prošlima, te iskoristiti tu vezu za predviđanje i modeliranje vrijednosti podataka u budućnosti.

2.5. Skup podataka za učenje modela – nezavisni redci

Prvi način spajanja podataka o vremenskoj prognozi s podacima o povijesnoj proizvodnji napravljen je za potrebe učenja modela koji ne zahtijevaju ulaz u obliku vremenskog niza. Cilj je bio dobiti tipičan skup podataka za nadzirano učenje gdje će značajke vremenske prognoze služiti kao značajke na temelju kojih se uči model, a povijesna proizvodnja biti će ciljna značajka. Skup podataka za učenje će biti u obliku individualnih redaka, te se neće iskoristiti karakteristike vremenskog niza ciljne značajke.

Ideja ovog spajanja podataka bila je napraviti unutarnje spajanje (eng. *inner join*) skupova podataka o povijesnoj proizvodnji po ključu **power_timestamp** i podataka o vremenskim prognozama po ključu **vt**. Na taj način svaka prognoza bi se povezala s svojom odgovarajućom ciljnom vrijednosti, te bi se sačuvale prognoze izrađene u različita vremena za istu ciljnu vrijednost.

Kako su rezolucije podataka o povijesnoj proizvodnji i vremenskoj prognozi različite, a zahtjev projekta je bio da prognoza treba biti u 15 minutnoj rezoluciji, bilo je potrebno

prebaciti podatke za vremensku prognozu iz satne u 15 minutnu rezoluciju. To je postignuto dodavanjem redaka s vrijednosti **vt** za svaki 15 minutni interval između dvije postojeće satne vrijednosti, npr. između redaka vrijednosti **vt** 00:00:00 i 01:00:00 dodani su reci s vrijednosti **vt** 00:15:00, 00:30:00 i 00:45:00, za svako vrijeme izrade prognoze **TOF** pojedinačno. Dodavanje novih redaka je rezultiralo „rupama" u skupu podataka ispunjenim nedostajućim vrijednostima između prethodno prisutnih vrijednosti za ostale značajke. Ove su „rupe" ispunjene linearnom interpolacijom, koja predstavlja da se nepoznate vrijednosti između dvije poznate točke kreću po pravcu koji spaja te poznate točke, te da su nepoznate vrijednosti jednoliko razmaknute. Kao alternativne metode razmatrale su se i kvadratna i kubna interpolacija, koje nastoje aproksimirati pojedine značajke krivuljom višeg stupnja, no odbačene su zato što se linearna interpolacija usprkos tome što je najjednostavnija pokazala kao podjednako dobra u usporedbi s ostalima u testovima na jednostavnijim modelima te olakšavala izvedbu jedne obrade rezultata koja će u kasnijim poglavljima biti detaljnije opisana.

Također, prilikom ovog spajanja dodane su dvije nove značajke izvedene iz vremenskih značajki. Prva je **num_minutes,** trenutni broj proteklih minuta u danu za stupac **vt**, npr. 75 za 01:15:00. Na taj način dobio se jednoznačni brojčani podatak koji opisuje trenutno vrijeme u danu, što je potencijalno korisna informacija kako solarne elektrane proizvode električnu energiju po danu a ne po noći. Druga dodana značajka je **month,** broj koji označava trenutni mjesec. Ova značajka je potencijalno korisna informacija kako se očekuje da solarne elektrane npr. u nekim zimskim mjesecima proizvode manje električne energije zbog čestog lošeg vremena, a u ljetnim više.

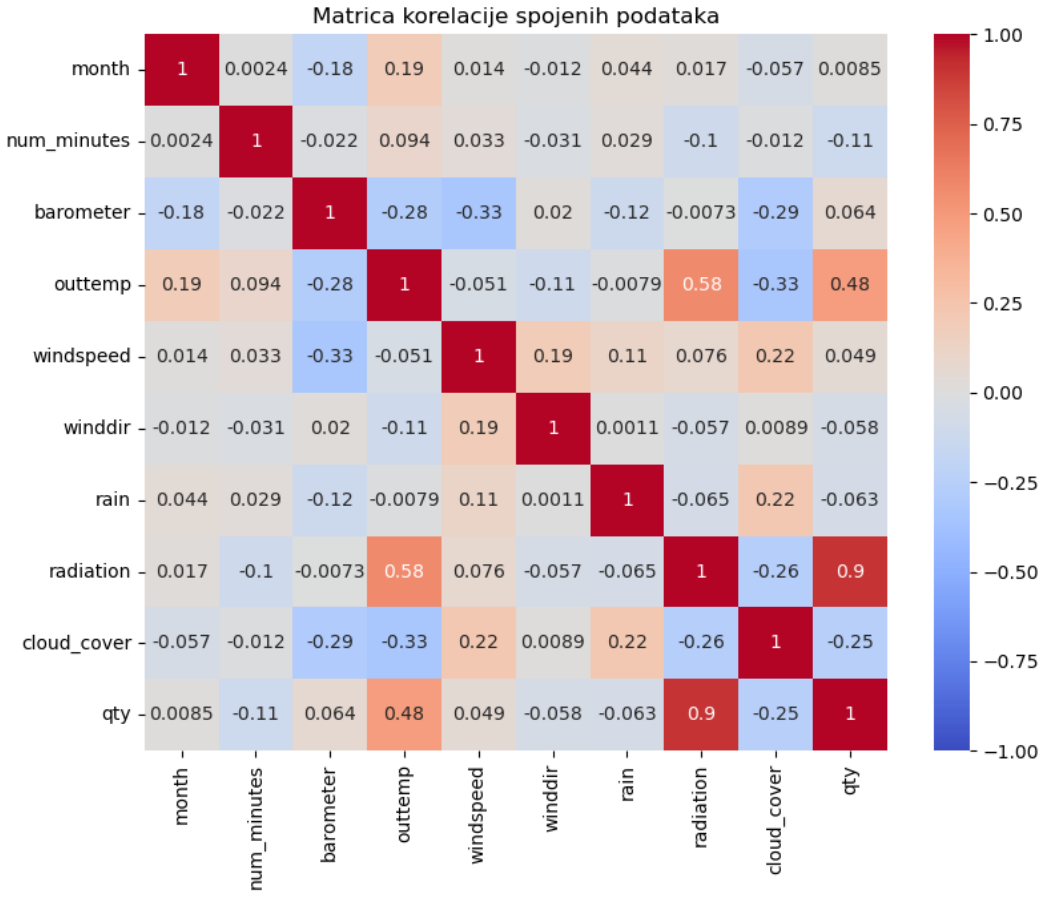
U sklopu ovog spajanja iz skupa podataka o povijesnoj proizvodnji izbačena su i prethodno spomenuta razdoblja kada je proizvodnja elektrane dulje vremena bila 0. Konačno, nakon samog spajanja radi urednosti skupa podataka izbačeni su podaci s vremenom izrade prognoze koji nisu imali odgovarajuće podatke o povijesnoj proizvodnji za cijelo razdoblje prognoze.

Nakon spajanja podataka bilo je moguće provesti analizu spojenog skupa podataka. Na temelju provedene analize bilo je omogućeno bolje razumijevanje podataka te odabir značajki za predviđanje ciljne značajke.

Prvi postupak u analizi podataka bio je izrada korelacijske matrice. U ovoj matrici prikazan je Pearsonov koeficijent korelacije između svake dvije varijable u skupu podataka. Pearsonov koeficijent korelacije računa se formulom (7):

$$r = \frac{\Sigma(x_i - x)(y_i - y)}{\sqrt{\Sigma(x_i - x)^2 \Sigma(y_i - y)^2}} \tag{7}$$

Raspon vrijednosti Pearsonovog koeficijenta korelacije je od -1 do 1, gdje vrijednosti između -1 do 0 označavaju negativnu linearnu povezanost, vrijednost 0 nedostatak linearne povezanosti a vrijednosti između 0 i 1 pozitivnu linearnu povezanost. Što je broj bliži 1 ili -1 to je povezanost jača. Također je bitno napomenuti da Pearsonov koeficijent korelacije prikazuje samo linearnu povezanost između dvije varijable. Ako postoji jasna povezanost koja nije linearna to ne mora biti reflektirano u Pearsonovom koeficijentu korelacije. Također, korelacija ne označava nužno uzročno posljedičnu vezu između dvije varijable nego samo njihovu međusobnu povezanost. Na slici Slika 2.1 prikazana je izračunata korelacijska matrica za spojeni skup podataka:

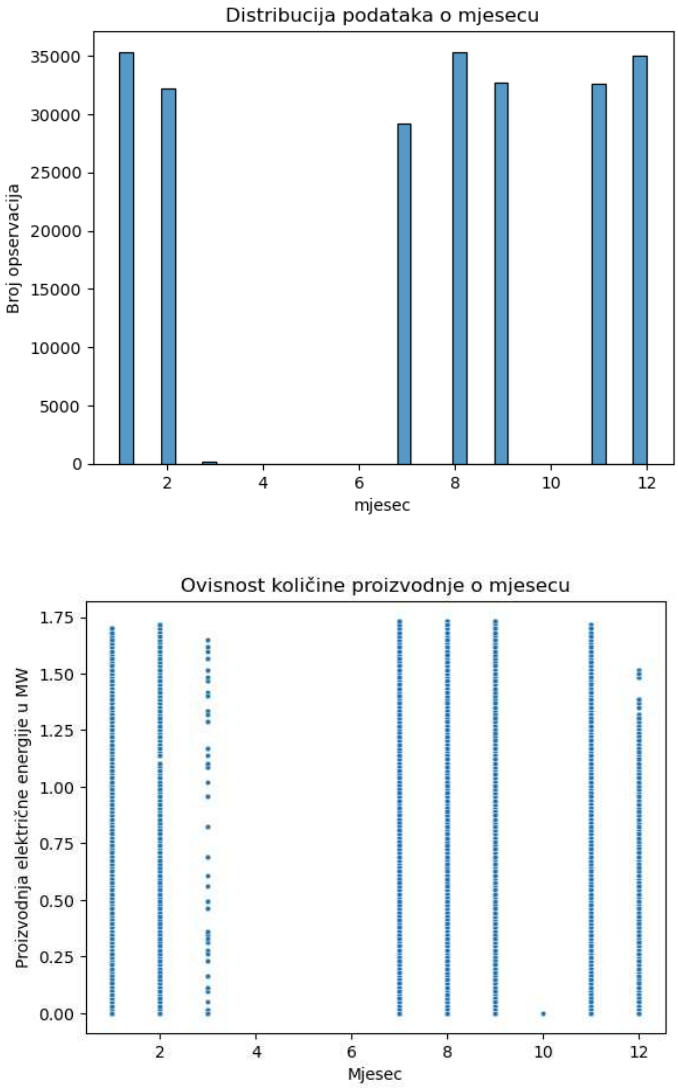


Slika 2.1 Korelacijska matrica spojenih podataka – nezavisni retci

Na ovoj slici najbitnije su povezanosti ciljne varijable **qty** i ostalih varijabli. Odmah se može uočiti vrlo visoka korelacija s varijablom **radiation.** Kako solarne elektrane proizvode električnu energiju s pomoću energije sunčevog zračenja, očekivano je da će ovo biti najvažnija značajka na temelju koje se predviđa proizvodnja solarne elektrane. Ostale varijable s većom (više od 0.2 apsolutne vrijednosti) povezanosti s ciljnom varijablom su **outtemp** i **cloud_cover**. Ove varijable su također povezane i s varijablom **radiation** na sličan način, te je očekivano da više sunčevog zračenja rezultira većom temperaturom, i da veća naoblaka smanjuje sunčevo zračenje i temperaturu kod površine.

Osim povezanosti s ciljnom varijablom iz matrice korelacije mogu se iščitati i razne veze među varijablama. Npr. varijable koje označavaju loše vrijeme kao **rain** i **cloud_cover** ili **windspeed** i **cloud_cover** su međusobno povezane što je i očekivano kako razne vremenske prilike često dolaze zajedno. Zato je moguće je da će uključivanje značajki koje nisu izravno povezane s ciljnom dati jasniju sliku o vremenskim prilikama i u konačnici poboljšati rezultat.

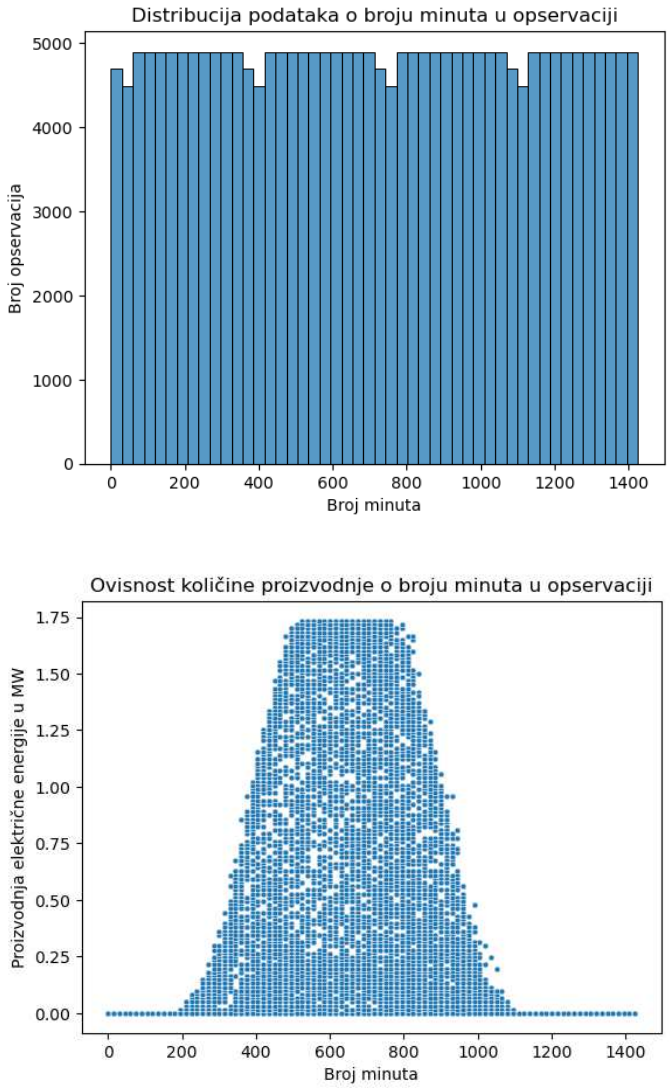
Osim analize s pomoću matrice korelacije, dodatno je provedena analiza pojedinih značajki. Za svaku značajku izrađen je histogram koji prikazuje distribuciju podataka, te raspršeni dijagram (eng. *scatterplot*) te značajke i ciljne. Informacija o distribuciji podataka je bitna zato što su neke transformacije na podacima korisne za određene vrste modela strojnog učenja, kao npr. standardizacija podataka (eng. *data standardization*) podrazumijevaju da su podaci normalno distribuirani što ne mora biti uvijek slučaj sa svim značajkama. Raspršeni dijagrami su bitni zato što se s pomoću njih mogu vizualizirati povezanosti između varijabli i ciljne značajke, te uočiti povezanosti koje nisu linearne te ih se stoga ne vidi u korelacijskoj matrici. Prvo će biti prikazana analiza za značajku **month** na slici Slika 2.2



Slika 2.2 Analiza značajke **month**

U analizi značajke **month** vidi se da u skupu podataka nisu prisutni svi mjeseci, što je bilo opisano prilikom analize ulaznih podataka. Također se vidi da su za prisutne mjesece podaci otprilike uniformno distribuirani, također prema očekivanjima. Konačno, na raspršenom dijagramu ne vide se jasne veze između mjeseca i proizvodnje električne energije, osim možda u 12. mjesecu Može se zaključiti da ova značajka u trenutnom skupu podataka nije iskoristiva. Ako se skupe podaci za sve mjesece kroz više godina, moguće je da će se na temelju toga moći u budućnosti uočiti neke pravilnosti.

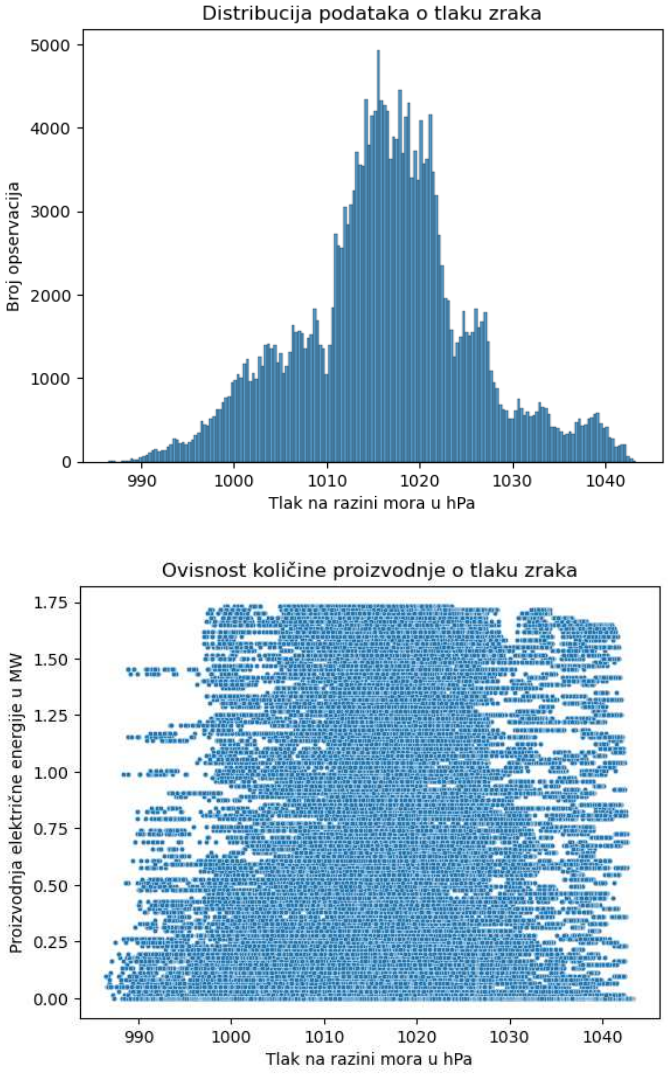
Sljedeća značajka koja se analizira je **num_minutes** na slici Slika 2.3



Slika 2.3 Analiza značajke **num_minutes**

Za značajku **num_minutes** također vidimo očekivanu uniformnu distribuciju podataka, kako se prognoze izrađuju u regularnim intervalima za isti broj sati unaprijed. Također vidimo vrlo jasnu i očekivanu vezu između broja minuta i proizvodnje, gdje nema proizvodnje po noći a ima po danu. Kako ova veza nije linearna, nije uočena u matrici korelacije što ilustrira jedan od nedostataka te metode. Na osnovi provedene analize može se zaključiti da značajka daje bitnu informaciju za predviđanje proizvodnje.

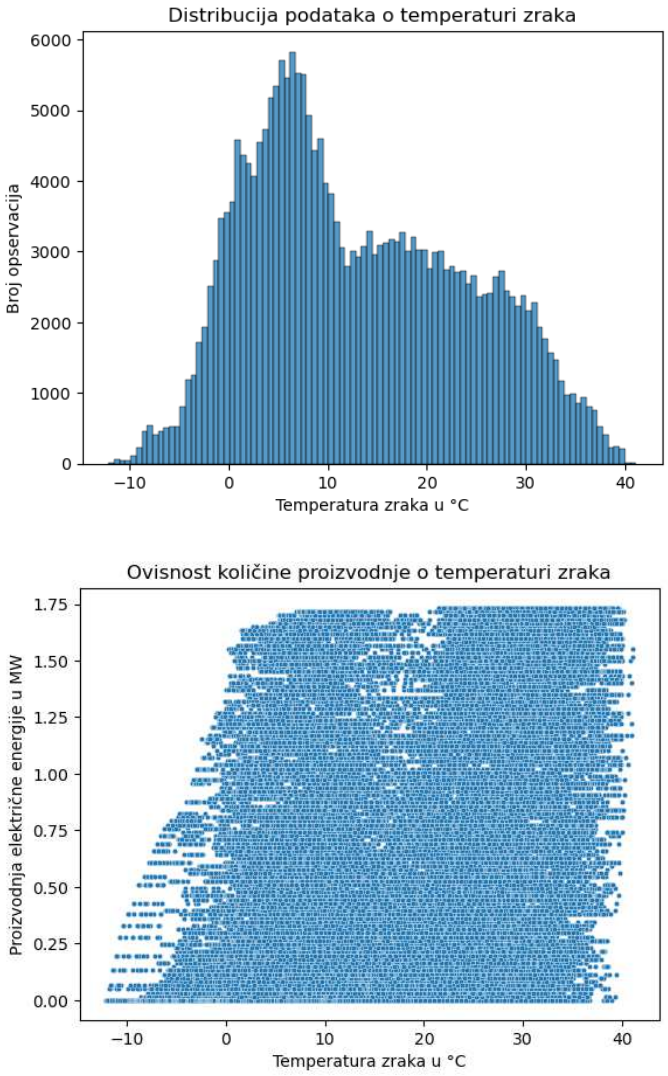
Sljedeće je prikazana analiza značajke **barometer** na slici Slika 2.4



Slika 2.4 Analiza značajke **barometer**

Za tlak zraka vidi se distribucija slična normalnoj, sa srednjom vrijednosti između 1010 i 1020 hPa. Iz grafa raspršenja vidimo da postoji ovisnost između tlaka zraka i proizvodnje električne energije. Kada je tlak zraka nizak, više vrijednosti proizvodnje su vrlo rijetke dok su prisutne kada je tlak prosječan ili visok. Razlog tomu je vjerojatno povezanost niskog atmosferskog tlaka s olujama, padalinama, ciklonama i ostalim turbulentnim vremenom (Rosenberg, 2020). Uzevši to u obzir, tlak zraka smatra se korisnom značajkom za predviđanje proizvodnje električne energije.

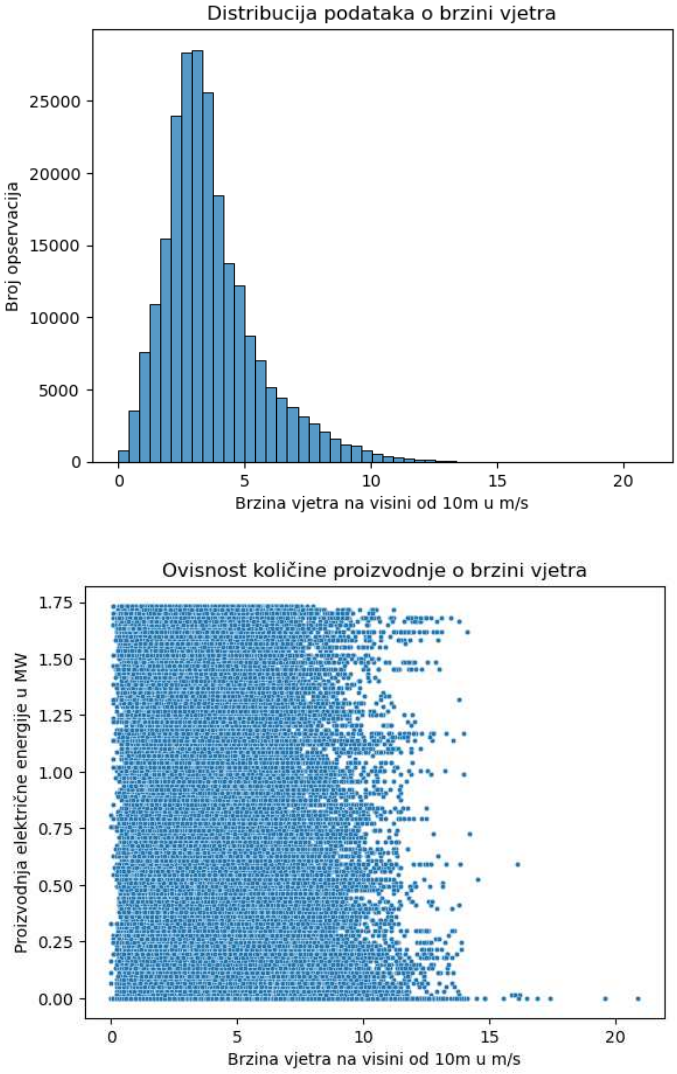
Ispod je prikazana analiza značajke **outtemp** na slici Slika 2.5



Slika 2.5 Analiza značajke **outtemp**

Za ovu značajku vidimo distribuciju sličnu normalnoj, no nagnutu (eng. *skew*) ulijevo, prema nižim temperaturama. Pretpostavka uzroka te distribucije je vjerojatno to što u skupu podataka nisu prisutna mjerenja iz cijele godine, te neki mjeseci pretežno u proljeće nisu zastupljeni. Po ostalim karakteristikama vrijednosti su unutar očekivanih. U cjelovitijem skupu podataka gdje su svi mjeseci zastupljeni očekivala bi se distribucija bliža normalnoj. Također, vidimo i jasnu ovisnost temperature zraka o proizvodnji zato što prilikom niskih temperatura uopće nema viših vrijednosti proizvodnje. Kako je ova značajka znatno povezana s ciljnom i u matrici korelacije, zasigurno će biti korisna u predviđanju.

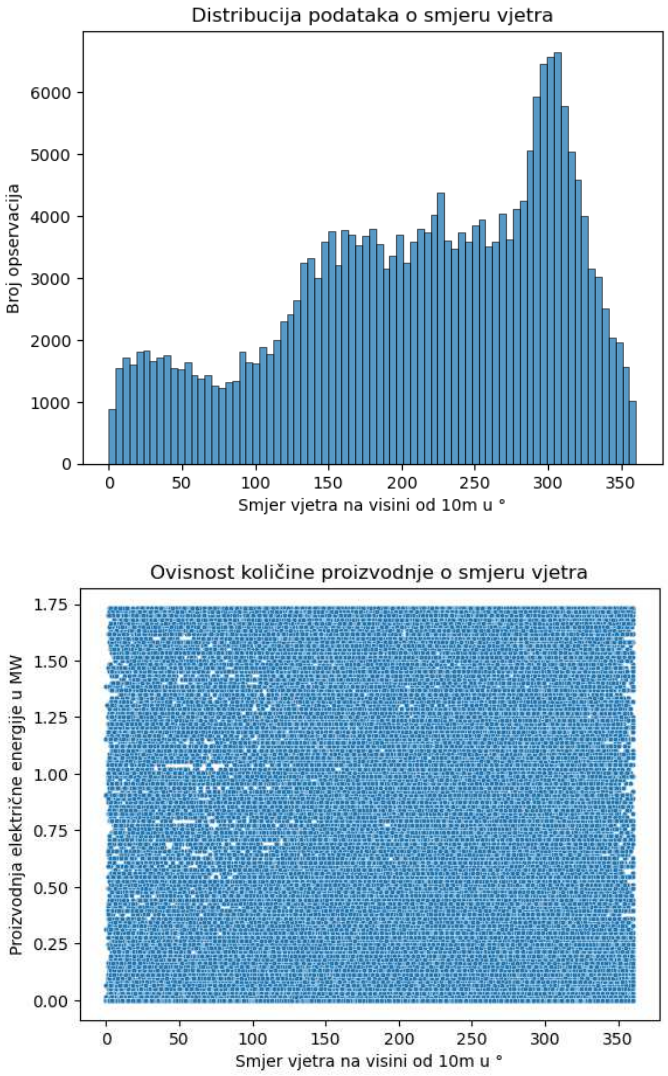
Sljedeće se analizira značajka **windspeed** na slici Slika 2.6



Slika 2.6 Analiza značajke **windspeed**

Značajka brzine vjetra također ima distribuciju sličnu normalnoj nagnutu ulijevo. To je unutar granica očekivanoga, kako je s jedne strane brzina vjetra ograničena nulom, a s druge su moguće razne velike brzine u rijetkim uvjetima. Raspon vrijednosti je također u skladu s očekivanjima. Po pitanju ovisnosti proizvodnje o brzini vjetra, ne može se uočiti jasna veza u većini slučajeva. No, vidi se da su velike iznimno velike brzine vjetra veće od 15 m/s skoro uvijek povezane s minimalnom proizvodnjom. Razlog tomu je vjerojatno što se takve brzine vjetra pojavljuju popraćene ekstremno lošim vremenom koje negativno utječe na proizvodnju solarnih elektrana. Ova značajka je također u matrici korelacije povezana s ostalima koje bi mogle imati utjecaja na ciljnu varijablu kao **barometer** ili **cloud_cover**. Sve u svemu, moguće je da će biti korisna za konačno predviđanje.

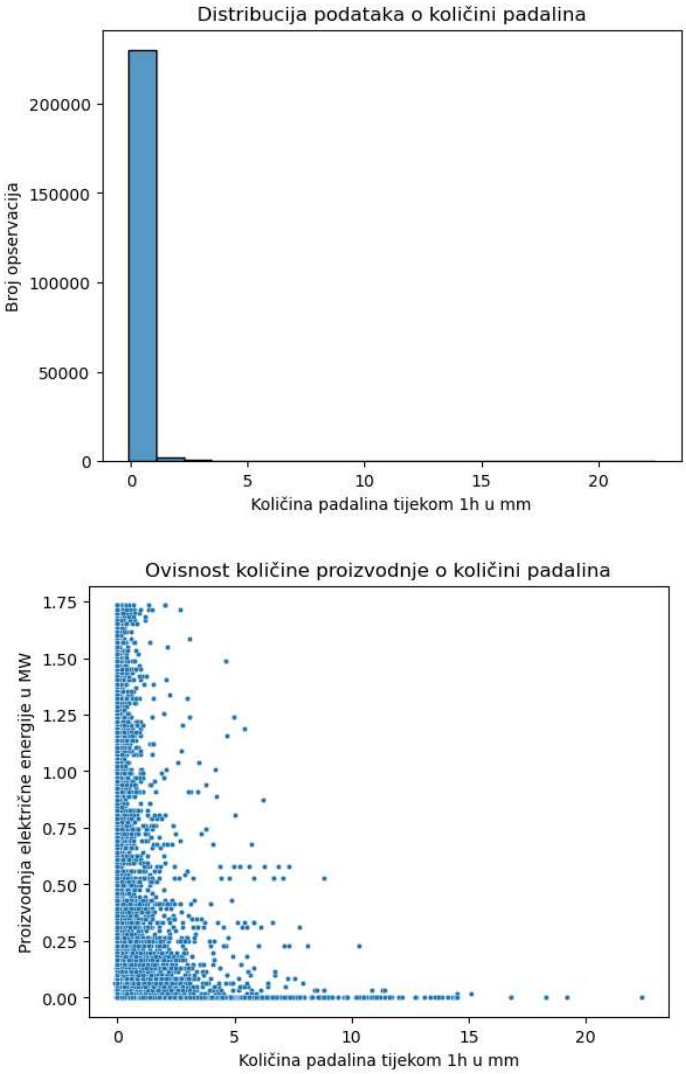
Značajka analizirana na sljedećoj slici, Slika 2.7, je značajka **winddir**.



Slika 2.7 Analiza značajke **winddir**

Distribucija ove značajke ne pokazuje jasnu distribuciju, no to nije očekivano kako vjetar puše iz različitih smjerova ovisno o lokaciji i ostalim vremenskim prilikama. Vidi se da su neki smjerovi, kao oni sjeverozapadno (oko 300̊), značajno zastupljeniji od ostalih dok su neki rijetki. Također na raspršenom dijagramu ne vidimo nikakvu ovisnost ciljne značajke o ovoj. Ova je značajka u matrici korelacije također slabo povezana sa svim ostalim značajkama, jedina nešto veća veza je sa značajkom **windspeed**. Uzevši sve ovo u obzir, ne očekuje se da će ova značajka biti korisna u predviđanju.

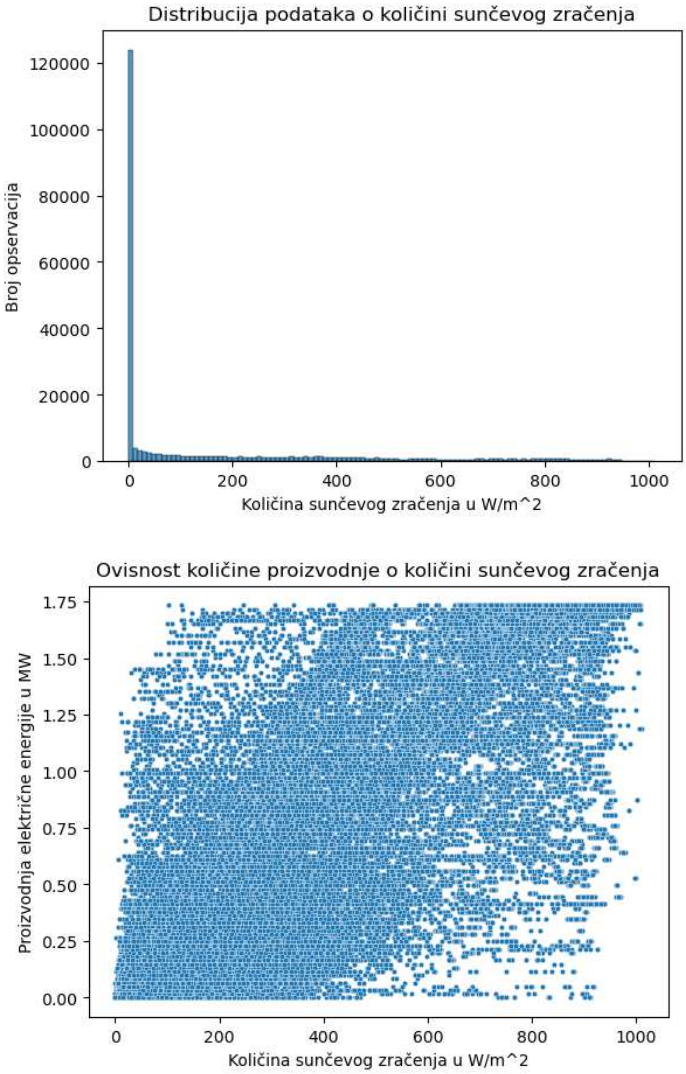
Nakon ove značajke prikazana je analiza značajke **rain** na slici Slika 2.8



Slika 2.8 Analiza značajke **rain**

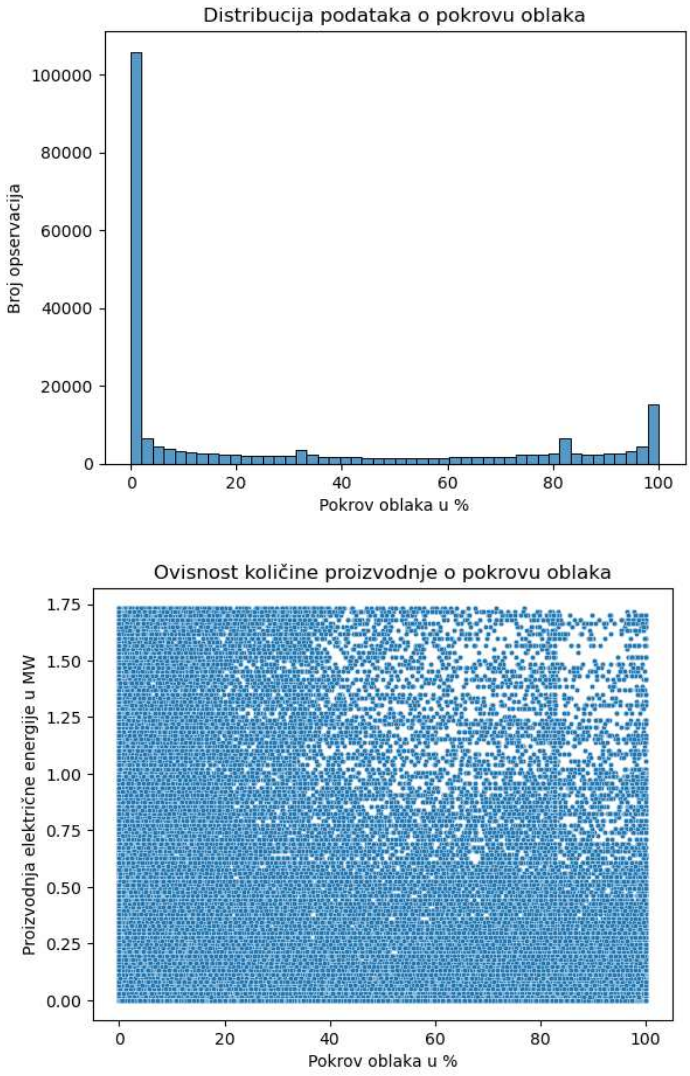
Značajka koja mjeri količinu padalina ima distribuciju vrlo nagnutu na lijevo, s velikom većinom podataka vrijednosti 0. To je očekivano, kako kiša nije stalna pojava u okolici analizirane elektrane te se relativno rijetko događa. Raspon vrijednosti je također unutar očekivanog, nema negativnih ili nerealno velikih vrijednosti. Ovisnost količine proizvedene električne energije o količini kiše slična je onoj za značajku **windspeed**, ali izraženija. Kada je vrijednost ove značajke manja ne možemo donijeti jasan zaključak o proizvodnji, no pri većim vrijednostima proizvodnja je skoro uvijek mala. To je očekivano, kako kiša ometa rad solarnih elektrana te često dolazi s ostalim vremenom nepogodnim za solarne elektrane kao što je velika naoblaka, što se može vidjeti i u matrici korelacije. Moguće je da će zbog toga ova značajka davati korisnu informaciju prilikom predviđanja.

Sljedeće je analizirana značajka **radiation** na slici Slika 2.9



Slika 2.9 Analiza značajke **radiation**

Histogram za značajku **radiation** pokazuje sličnu distribuciju kao i za značajku **rain**. Većina vrijednosti ove značajke je 0, čemu je uzrok to što preko noći nema sunčevog zračenja. Također je jasno vidljiva pozitivna korelacija između zračenja i proizvodnje električne energije. Uzevši u obzir i vrlo velik koeficijent Pearsonove korelacije u matrici korelacije između ciljne značajke i ove, jasno je kako će ova značajka obavezno biti uvrštena u modele te da će vjerojatno biti najbitnija značajka na temelju koje se predviđa. Konačno, prikazana je analiza značajke **cloud_cover** na slici Slika 2.10



Slika 2.10 Analiza značajke **cloud_cover**

Distribucija značajke **cloud_cover** također slijedi sličnu distribuciju kao i prethodne dvije značajke. Također se vidi da postoji ovisnost između pokrova oblaka i proizvodnje, pri većem pokrovu oblaka rjeđe su veće vrijednosti proizvodnje. Sve u svemu zbog postojanja većeg Pearsonovog koeficijenta korelacije u matrici korelacije i uočene veze pretpostavka je da će ova značajka biti korisna u predviđanju proizvodnje solarne elektrane.

Analizirane podatke s vremenskim značajkama možemo usporediti i s prosječnim vremenskim uvjetima u regiji gdje se nalazi elektrana. Tako se može provjeriti kvaliteta prognoze te steći bolje razumijevanje vremenskih prilika te regije. U ovom slučaju vrijednosti značajki su unutar očekivanja, te ovom provjerom nisu ustanovljeni problemi. (what's the weather like.org, 2024).

Čest problem koji se pojavljuje kod ovih značajki je taj da je proizvodnja solarne elektrane prvenstveno ovisna o količini sunčevog zračenja, a tek indirektno o ostalim značajkama. Npr. po noći će proizvodnja uvijek biti 0, bez obzira na to je li oblačno ili nije, te ima li kiše ili ne. To unosi šum u model te smanjuje korisnost ostalih značajki. Kako bi se ublažio ovaj

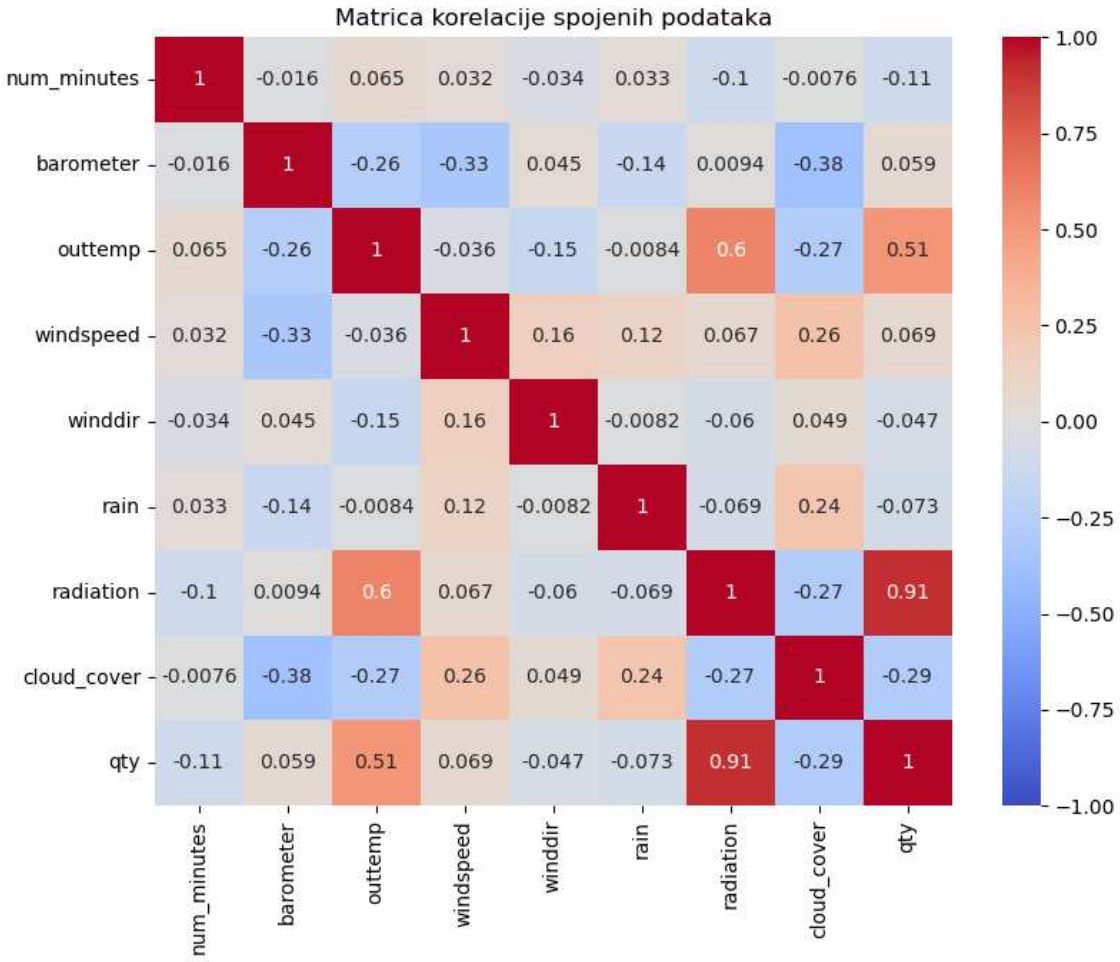
problem valja razmotriti metode koje proširuju značajke kao što su dodavanje interakcijskih značajki ili značajki višeg reda koje će biti detaljnije obrazložene u kasnijim poglavljima.

2.6. Skup podataka za učenje modela – vremenski niz

Cilj drugog načina spajanja podataka bio je dobiti neprekinuti vremenski niz podataka za modele koji očekuju takav ulaz. Problem kod ovog načina spajanja bila je struktura podataka o vremenskoj prognozi, koji za isto vrijeme imaju više prognoza izrađenih u različitim vremenskim trenutcima. Olakotna okolnost bila je da su podaci o proizvodnji već imali karakteristike vremenskog niza. Također je i ovdje bio prisutan problem razlike u rezolucijama između podataka (prognoze su u satnoj rezoluciji a proizvodnja u 15 minutnoj). Stoga se svaka točka iz podataka o proizvodnji koja se mogla spojiti s točkom iz podataka o vremenskoj prognozi, odnosno točke u punim satima, spojila s naj ažurnijom prognozom za tu točku tj. odabrala se prognoza s najvećim **TOF** dostupna za tu točku. Tako se nastojao dobiti vremenski niz s najtočnijim vremenskim prognozama, kako se očekuje da su prognoze točnije kada su napravljene za manje vremena unaprijed. „Rupe“ u podacima za značajke vremenske prognoze između točaka koje su već spojene s prognozom popunjene su linearnom interpolacijom, kao i za prethodno opisani skup podataka.

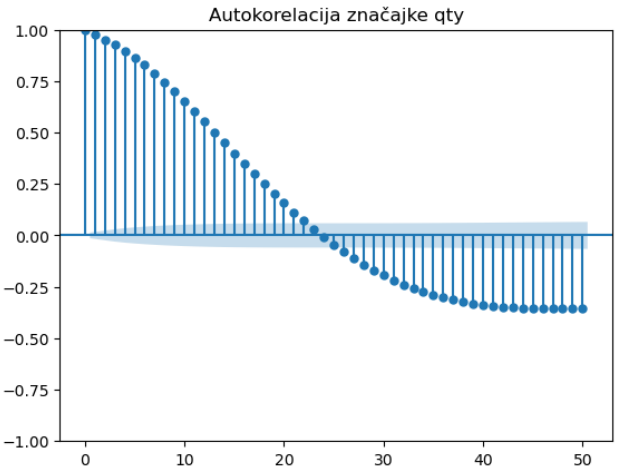
Ovom skupu podataka također su otklonjena prethodno spomenuta razdoblja gdje je proizvodnja dugo vremena bila 0. Iako uklanjanje ovih podataka u 10.mj. koji nisu na početku ili kraju niza unosi prekid, što narušava kvalitetu podataka, odlučeno da će se tako postupiti radije nego da se loši podaci zadrže te da će se 9. mjesec spojiti direktno s 11. kako bi se dobio kontinuirani vremenski niz. Također je podacima dodana i značajka **num_minutes** na isti način kao i u prethodnom skupu podataka.

Za ovaj skup podataka nije izrađena detaljna analiza značajki kao za prethodni skup. Pretpostavlja se da su pravilnosti i veze uočene na prošlom skupu slične kao one na novom, kako je novi skup podataka zapravo samo podskup prethodnog skupa. To se može pokazati prikazom matrice korelacije za drugi način spajanja na slici Slika 2.11, koja je vrlo slična prethodnoj prikazanoj na slici Slika 2.1:



Slika 2.11 Korelacijska matrica spojenih podataka – vremenski niz

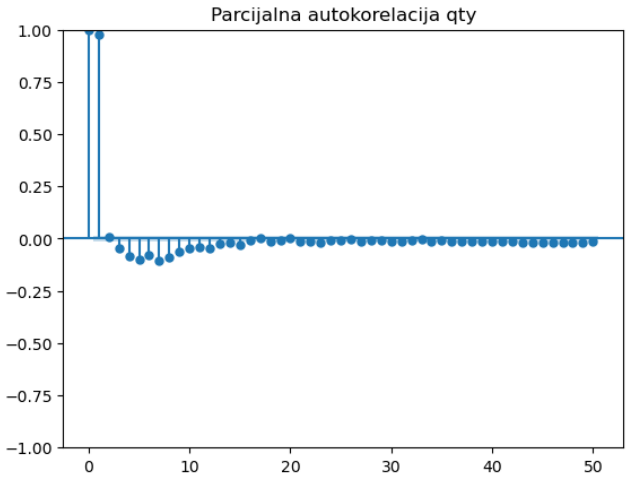
Podaci u obliku vremenskog niza također omogućuju druge načine analize specifične za vremenske nizove. Primjer takve analize je analiza autokorelacije. Autokorelacija je Pearsonov koeficijent korelacije vrijednosti s njenim prethodnim koracima u vremenskom nizu (eng. *lags*). Vizualizacija autokorelacije ciljne značajke s 50 svojih prošlih vrijednosti prikazana je na slici Slika 2.12:



Slika 2.12 Vizualizacija autokorelacije ciljne značajke

Iz ove vizualizacije možemo vidjeti jasnu autokorelaciju ciljne značajke s vrijednostima neposredno prije. To je očekivano, kako se vremenske prilike o kojima ovisi proizvodnja solarnih elektrana ne mijenjaju vrlo brzo a mjerenja su obavljana u kratkom vremenskom intervalu od 15 minuta. Također možemo vidjeti opadanje autokorelacije s kasnijim vrijednostima, te prelazak u negativnu autokorelaciju. To pokazuje sezonalnost u podacima, odnosno izmjenu dana u kojem tipično ima proizvodnje i noći u kojoj je proizvodnja 0.

Osim autokorelacije možemo prikazati i analizu parcijalne autokorelacije. Prilikom izračuna parcijalne autokorelacije između trenutne vrijednosti i određene točke u prošlosti izbacujemo utjecaj ostalih vrijednosti između te dvije. Tako možemo vidjeti koliko nove informacije donosi korištenje starijih vrijednosti koja nije sadržana u novijima. Graf parcijalne autokorelacije prikazan je na slici Slika 2.13



Slika 2.13 Vizualizacija parcijalne autokorelacije ciljne značajke

Iz ove vizualizacije vidljivo je da su gotovo sve informacije sadržane u značajki odmah iza trenutne, te da malo dodatne informacije dobivamo uključivanjem ostalih značajki. To je problematično za ovaj zadatak, kako je cilj predvidjeti više od jednog koraka unaprijed, te je zbog toga vrijednost proizvodnje odmah prije ciljne dostupna samo za prvi korak predviđanja. Zbog te činjenice možemo zaključiti da će predviđanje proizvodnje isključivo na temelju prošlih vrijednosti više od nekoliko koraka unaprijed vjerojatno davati loše rezultate, te da će značajke vremenske prognoze biti ključne za postizanje dobrih rezultata.

3. Implementacija modela strojnog učenja

Cilj ovoga rada je bio primijeniti razne modele strojnog učenja na prethodno opisane podatke radi dobivanja prognoze proizvodnje solarne elektrane. U ovom će se poglavlju opisati korištene tehnologije, postupak razdjeljivanja podataka u skup za testiranje i učenje, navesti koji su modeli strojnog učenja odabrani za rješavanje ovog problema te koji je njihov način funkcioniranja, koje značajke su odabrane i na koji način je izveden inženjering značajki za pojedini model, na koji način su evaluirani model te konačno prikazati i rezultati za sve modele.

3.1. Korištene tehnologije

Za implementaciju modela koriste se slične tehnologije kao i za obradu i analizu podataka. U ovom radu modeli su implementirani u programskom jeziku **Python**. Za učitavanje i rukovanje skupovima podataka korištena je Python biblioteka **Pandas**, dok je za neke operacije nad podacima korištena biblioteka **Numpy**. Za većinu operacija vezanih uz definiranje i treniranje modela, inženjering značajki i evaluaciju modela korištena je biblioteka **Scikit-learn**. Za definiranje i treniranje modela dubokog učenja korištena je biblioteka **Tensorflow**. Za vizualizacije rezultata korištena je biblioteka **Matplotlib**.

3.2. Ograničenja pri modeliranju

Prije opisa korištenja modela bitno je napomenuti ograničenja prisutna u ovom istraživanju, koja su utjecala na dobivene rezultate. Prvo bitno ograničenje je da dostupnih podataka o povijesnoj proizvodnji elektrane bilo relativno malo, to jest za razdoblje od 7 mjeseci. Drugo bitno ograničenje je da su bili dostupni podaci za samo jednu solarnu elektranu.

Posljedica kratkog vremenskog razdoblja iz kojeg su podaci dostupni vjerojatno će biti lošije performanse modela. Prilikom treniranja modela generalno je poželjno imati što više dostupnih podataka, kako bi model što bolje naučio generalizirati i prilagoditi se raznim mogućim ulaznim podacima. To je pogotovo bitno u podacima korištenim u ovom radu kako svako godišnje doba ima specifične vremenske uvjete. Ako podaci za određeno godišnje doba nedostaju u skupu podataka za treniranje, mogu se očekivati teškoće prilikom korištenja modela u tim vremenskim razdobljima.

Jasna posljedica ograničenja podataka samo na jednu elektranu bit će da je korištenje modela ograničeno samo na predviđanje proizvodnje te elektrane. Ovo nije nužno loše, zato što je moguće izraditi pojedinačni model za svaku solarnu elektranu čija se proizvodnja želi predviđati, što pogotovo vrijedi za modele koji se brzo treniraju. Također, prilikom analize rezultata rezultati će biti reprezentativni samo za elektranu na kojoj je napravljeno istraživanje, ne za predviđanje proizvodnje solarnih elektrana u Hrvatskoj općenito.

3.3. Podjela podataka

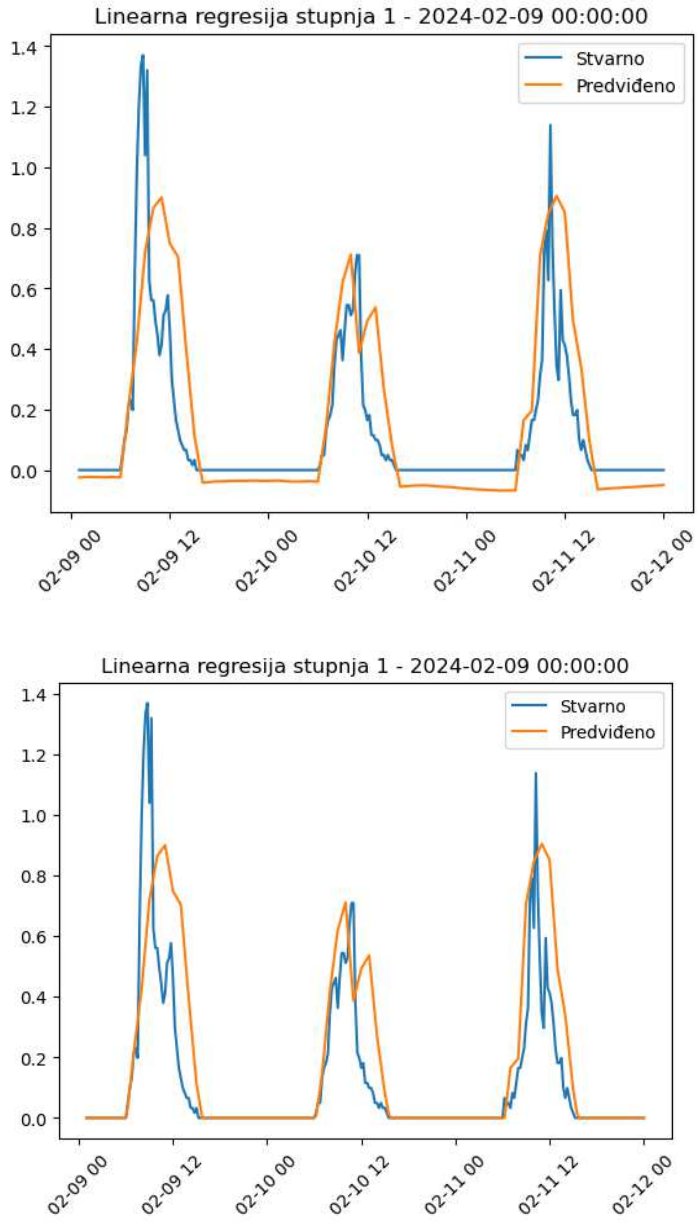
Kao što je prethodno spomenuto, proizvodnja električne energije je u ovome problemu ciljna značajka koja se predviđa na temelju ostalih značajki. Bitno je uzeti u obzir da bi modeli razvijeni u kontekstu ovog rada trebali funkcionirati u sklopu šire platforme za predviđanje proizvodnje solarnih elektrana. U takvom sustavu očekuje se da će u pojedinoj točki biti dostupni podaci iz prošlosti, a cilj biti predvidjeti proizvodnju u budućnosti. Zato su za skup podataka za testiranje uzeti podaci iz zadnjeg dostupnog mjeseca, veljača 2024, a podaci prije toga su korišteni za učenje modela umjesto da se iz podataka nasumično uzme određena količina podataka za učenje, i određena količina podataka za trening. Ovo je napravljeno za oba prethodno opisana skupa podataka.

Podaci za testiranje su naknadno filtrirani tako da ostaju samo one prognoze koje su izrađene na početku svakog dana u veljači 2024. Tako se simulirala izrada prognoze proizvodnje vjetroelektrane za sljedeća 3 dana, što je duljina vremenske prognoze iz jedne točke, svako jutro nakon što vremenska prognoza postane dostupna.

3.4. Obrada rezultata predviđanja

Bitno je napomenuti i da je za sve modele napravljena jednostavna obrada rezultata predviđanja modela u svrhu poboljšavanja kvalitete predviđanja. Ako je ulazna značajka **radiation** za neki primjer bila jednaka 0, ili je model predvidio negativnu vrijednost proizvodnje, onda je rezultat modela bio postavljen na 0 bez obzira na predviđanje modela. Pokazalo se da bez ovog postupka modeli nisu uspjeli precizno naučiti postaviti proizvodnju po noći na 0, te su nekada davali i besmislene negativne vrijednosti proizvodnje. Zbog toga je uveden ovaj postupak, koji je utemeljen na domenskom znanju i analizi podataka te se pokazalo da zamjetno poboljšava rezultate predviđanja. Primjer provedbe ovog postupka prikazan je na slici Slika 3.1

Mogućnost izvođenja ovog postupka također je razlog zašto je odabrana linearna interpolacija a ne kvadratna ili kubna. Ostale vrste interpolacija bi u značajki **radiation** između dvije vrijednosti 0 po noći često popunili vrijednosti nešto različite od 0, čak i besmislene negativne vrijednosti, u pokušaju prilagođavanja polinomijalne krivulje podacima. Linearna interpolacija je naprotiv popunjavala takve „rupe“ samo s vrijednosti 0. Vizualizacija ove obrade podataka prikazana je na slici Slika 3.1



Slika 3.1 Primjer predviđanja modela linearne regresije stupnja 1 prije (iznad) i nakon (ispod) obrade rezultata.

3.5. Evaluacija modela

U poglavlju 1. gdje je dan kratak pregled polja strojnog učenja, objašnjen je pojam funkcije pogreške i načina na koji se koristi prilikom učenja modela. Također je napomenuto da je izbor funkcije pogreške za optimiziranje donekle proizvoljan i često motiviran matematičkom jednostavnosti optimizacije. Za potrebe evaluacije rezultata modela takva su ograničenja manje bitna te je moguće na temelju vrijednosti koje je model predvidio i stvarnih vrijednosti izračunati različite metrike kojima se procjenjuje točnost predviđanja modela.

U ovome radu odabrane metrike su bile greške RMSE (eng. *Root Mean Squared Error*) i NRMSE (eng. *Normalized Root Mean Squared Error*). Formula za RMSE je samo korijen srednje kvadratne pogreške MSE, i opisana je u izrazu (8):

$$RMSE_w(x, y) = \sqrt{\frac{1}{N} \sum_{i=1...N} (f_w(x_i) - y_i)^2} \tag{8}$$

gdje je $f_w(x_i)$ predviđanje modela za neki primjer i, w parametri modela, x_i ulazne značajke primjera a y_i stvarna vrijednost ciljne značajke. Ova metrika je odabrana zato što se korjenovanjem izraza za MSE dobiva se jasniji uvid u to koliko je prosječno odstupanje predviđanja od stvarne vrijednosti bolje nego da se koristi kvadrirana vrijednost prosječnog odstupanja.

Greška NRMSE je greška RMSE podijeljena s konstantom, u ovom slučaju maksimalnom vrijednosti proizvodnje koja je iz podataka procijenjena na 1.75 MW, te pomnožena sa 100 da se pretvori iz udjela u postotak. Opisana je izrazom (9):

$$NRMSE_w(x, y) = RMSE_w(x, y)/1.75 \cdot 100 \tag{9}$$

Razlog korištenja ove metrike je da se da intuitivan uvid u obliku postotka koliko je velika greška modela u odnosu na raspon u kojem se kreću podaci. Npr. RMSE greška od 0.5 MW nije jednaka ako je maksimalni kapacitet elektrane 1 MW ili 100 MW.

Također, ove su greške za svaki model izračunate na dva načina. U prvom načinu su za svaku od prethodno spomenutih prognoza sa početka dana u skupu podataka za testiranje zasebno izračunate metrike RMSE i NRMSE. Na kraju se izračunao prosjek metrika NRMSE i RMSE od svih dana kako bi se dobila procjena performansi modela za više dana rada. U drugom načinu, za svaki 15 minutni interval od početka do kraja prognoze su se skupljali podaci zasebno za sve prognoze u testnom skupu. Zatim su se izračunale RMSE i NRMSE greške zasebno za svaki interval, te rezultati prikazali grafički krivuljom. Namjera iza ovog postupka je bila istražiti postoje li razlike u prosječnoj točnosti predviđanja između raznih perioda u danu, npr. po noći su prognoze uvijek 0 pa je model možda točniji u tim periodima i postoje li razlike u točnosti prognoza za manje ili više vremena unaprijed, npr. ako su vremenske prognoze preciznije u kraćem roku nego u dužem pa su posljedično i predviđanja proizvodnje solarnih elektrana točnija.

3.6. Odabir modela

U ovom radu su za implementaciju odabrani sljedeći modeli:

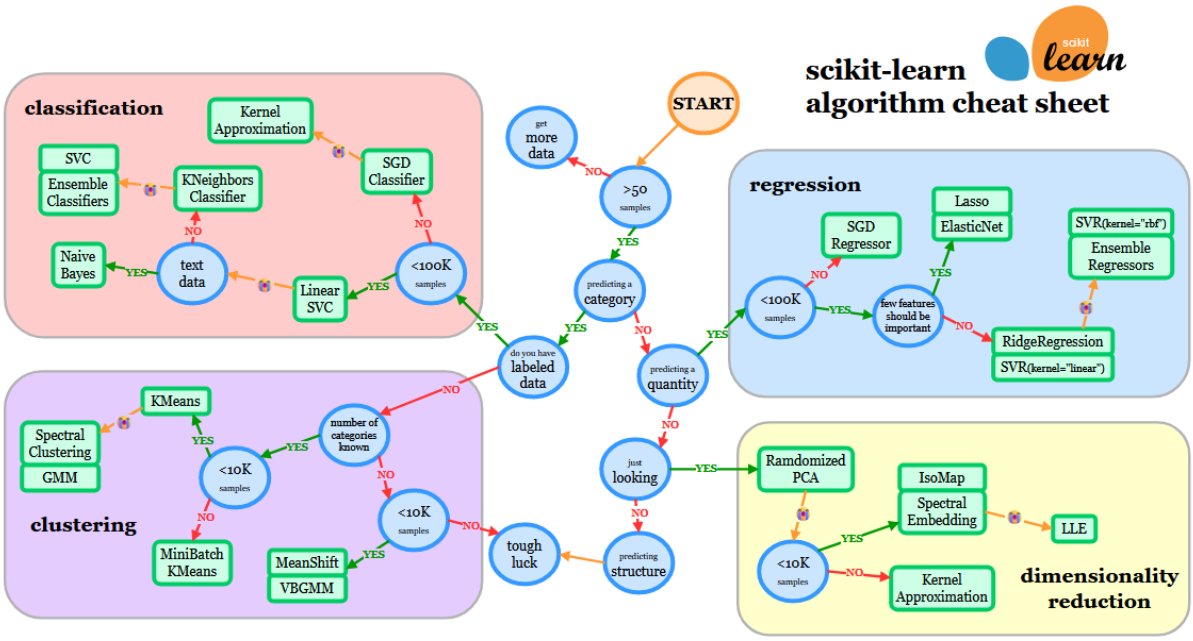
- Model linearne regresije, bez dodanih polinomijalnih značajki, te s dodanim polinomijalnim značajkama stupnja 2 i 3

- Model regularizirane linearne regresije (eng. *ridge regression*) bez dodanih polinomijalnih značajki, te s dodanim polinomijalnim značajkama stupnja 2 i 3. Provedena je optimizacija hiperparametara.
- Model **SVR** (eng. *Support Vector Regression*). Korištena jezgra je **rbf** (eng. *Radial Basis Function*), i provedena je optimizacija hiperparametara.
- Razne varijante modela dubokog učenja **LSTM** (eng. *Long Short Term Memory*)

Model linearne regresije, pogotovo linearne regresije bez dodanih polinomijalnih značajki, je odabran kao primjer jednostavnog regresijskog modela, te da se ostali kompleksniji modeli mogu usporediti s tim modelom te tako opravdati njihovo korištenje ako daju bolji rezultat.

Model regularizirane linearne regresije odabran je zato što je kompleksnije proširenje već odabranog modela linearne regresije. Na ovom modelu će se zato lako moći vidjeti posljedice povećanja kompleksnosti modela.

Model **SVR** i **rbf** jezgra su također odabrani kao primjer kompleksnijeg regresijskog modela prikladnog za ovaj problem. Odabir modela regularizirane regresije i modela SVR za ovaj problem također su podržani u dokumentaciji biblioteke **Scikit-learn**, vizualizacija procesa odabira modela prikazana je na slici Slika 3.2:



Slika 3.2 Proces odabira modela (Scikit-learn, 2024)

Konačno, model **LSTM** odabran je kao primjer modela dubokog učenja. Ovaj model je odabran na temelju znanstvenih radova koji su popisali metode strojnog učenja korištene za predviđanje proizvodnje solarnih elektrana u drugim znanstvenim radovima te njihove rezultate. U popisanim radovima često su korišteni modeli dubokog učenja različitih arhitektura, uključujući i **LSTM** te trenutno postižu dobre rezultate u tom području predviđanja. U nekim od navedenih radova, te u eksperimentu izvedenom u jednom od radova gdje su na isti skup podataka primijenjene razne metode strojnog učenja također se

pojavljuje i **SVR** model. (Bo Yang, 2023) (Yuan-Kang Wu, 2022) (Jwaone Gaboitaolelwe, 2023)

3.7. Model linearne regresije

Prvi model koji je u ovom radu bio upotrebljen je model linearne regresije. Način rada ovog modela objašnjen je u 1. poglavlju, kao primjer u svrhu objašnjenja načina rada modela strojnog učenja općenito.

Prije upotrebe ovog modela značajke su normalizirane korištenjem MinMax normalizacije. Od dostupnih značajki one korištene za model birane su kombinacijom uvida iz analize podataka prikazane u 2. poglavlju ovog rada te uspoređivanja rezultata pokretanja jednostavnijih modela, zbog kratkog vremena treniranja, s raznim kombinacijama značajki odabranim na testnom skupu podataka. Na kraju su odabrane značajke **num_minutes**, **barometer**, **outtemp**, **windspeed**, **rain**, **radiation**, **cloud_cover**.

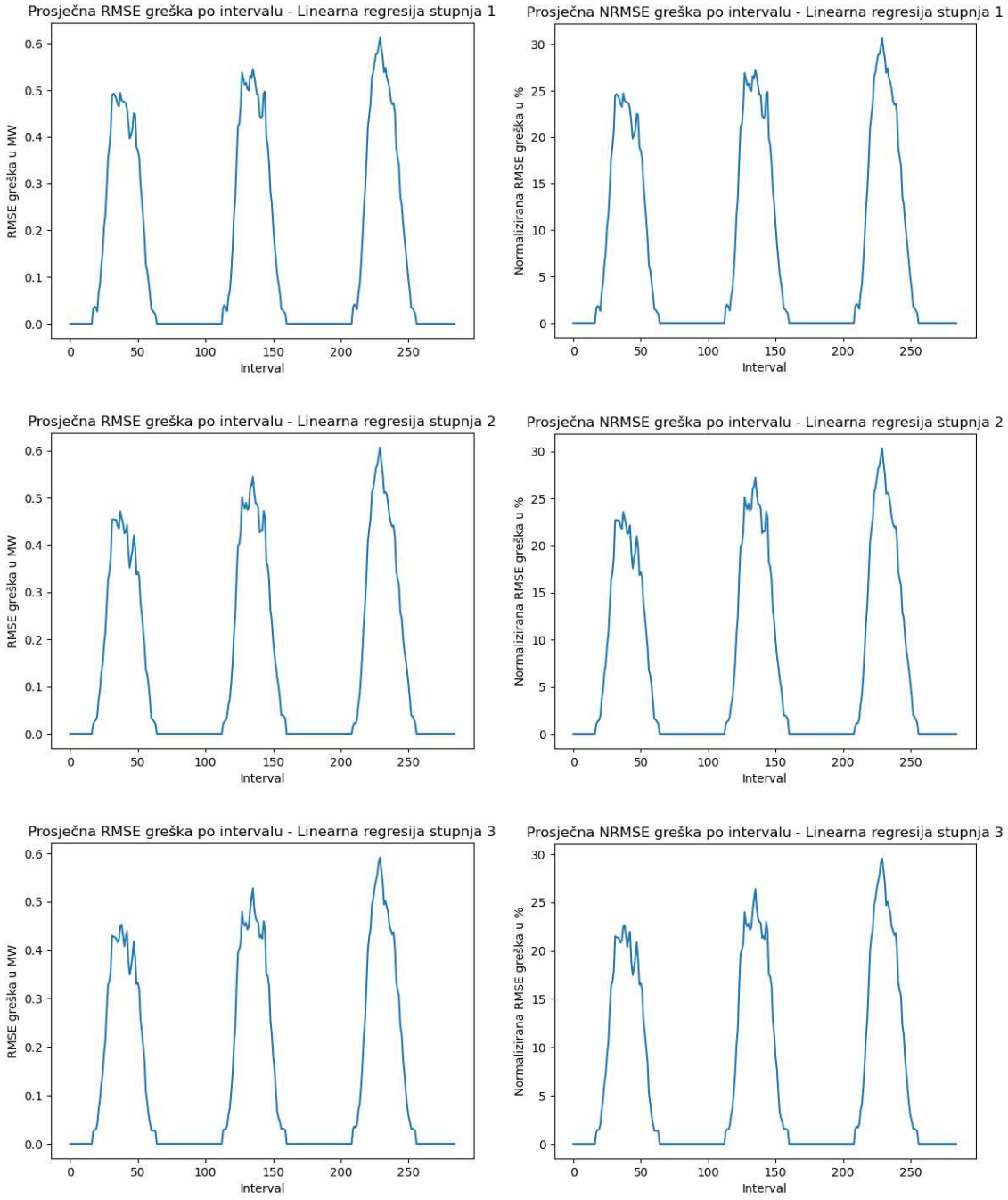
Odluka za ograničavanje dodavanja polinomijalnih značajki do 3. stupnja je bila motivirana isprobavanjem dodavanja polinomijalnih značajki raznih stupnjeva i praćenjem rezultata. Pokazalo se da nakon stupnja većeg od 3 rezultat na testnom skupu podataka počinje opadati, što je indiciralo prenaučenost modela.

Prosječna greška RMSE i NRMSE za modele linearnih regresija različitih stupnjeva prikazana je u tablici Tablica 3.1. Rezultati su zaokruženi na dvije decimale. Stupanj linearne regresije znači kojeg su stupnja značajke dodane, linearna regresija 1. stupnja je bez dodanih polinomijalnih značajki:

Tablica 3.1 Prosječni rezultati predviđanja za model linearne regresije

Model	RMSE	NRMSE
Linearna regresija 1. stupnja	0.25 MW	14.04%
Linearna regresija 2. stupnja	0.23 MW	13.34%
Linearna regresija 3. stupnja	0.23 MW	12.94%

Ovdje možemo vidjeti jasno poboljšanje nakon povećanja kompleksnosti modela dodavanjem polinomijalnih značajki do 3. stupnja. Na slici Slika 3.3 su prikazane i prosječne RMSE i NRMSE greške za ove modele po svakom intervalu:



Slika 3.3 Prosječni rezultati predviđanja po intervalu za model linearne regresije

Na ovoj slici možemo vidjeti zašto je bilo korisno podatke vizualizirati i na ovaj način. Jasno vidimo da model ima dosta dijelova tijekom noći gdje zbog obrade rezultata ima savršena predviđanja. U kombinaciji s ostalim dijelovima to rezultira naizgled boljim rezultatom NRMSE od oko 13-14%, dok se ovdje vidi da je ta greška u sredini dana, što je najbitnije za točno predvidjeti, obično veća od 20 %. Također se može potvrditi pretpostavka da su vremenske prognoze za manje vremena unaprijed zamjetno točnije. Na ovim se krivuljama jasno vidi rast greške u kasnijim danima predviđanja, ne računajući noći. Razlike u maksimalnoj grešci predviđanja između prvog i trećeg dana iznose oko 5 %. Također se i ovdje može vidjeti kako kompleksniji modeli većeg stupnja daju bolja predviđanja.

Na ovom primjeru može se i demonstrirati objašnjivost modela linearne regresije, pogotovo onog najjednostavnijeg bez dodanih polinomijalnih značajki. U tablici Tablica 3.2 prikazane su težine parametara pridijeljene pojedinoj značajki:

Tablica 3.2 Težine značajki linearne regresije bez dodanih polinomijalnih značajki

Značajka	Težina
num_minutes	-0.02
barometer	0.15
outtemp	-0.07
windspeed	-0.04
rain	0.01
radiation	1.85
cloud_cover	-0.02

Iz ovoga se jasno vidi kako je **radiation** najbitnija značajka, kao što je bilo i očekivano, s najvećom težinom te da je veza između proizvodnje i sunčevog zračenja snažna i pozitivna. Druga najbitnija značajka je **barometer** sa slabom pozitivnom vezom koja je također vizualizirana u analizi podataka, niski tlak je povezan s lošim vremenom. Ostale značajke imaju slabe veze, što znači da nisu jako relevantne za predviđanje ciljne značajke. Zanimljivo je da su značajke **cloud_cover** ili **outtemp** završile s malim koeficijentima u ovom modelu, iako se prilikom analize očekivalo da će igrati veću ulogu. No, ipak su zadržane prvenstveno radi modela veće kompleksnosti kako se prilikom treniranja pokazalo da nose bitnu informaciju i poboljšavaju rezultate ako su uključene.

Iako bi se teoretski mogli analizirati i objasniti parametri linearne regresije s dodanim polinomijalnim značajkama, zbog veće kompleksnosti modela i velikog broja značajki u ovom radu se nije provodila takva analiza.

3.8. Model regularizirane regresije

U ovom se radu koristi L2 regularizirana ili hrbatna regresija (eng. *ridge regression*). Model regularizirane regresije isti je kao i model linearne regresije, definiran je slijedećom jednadžbom (10):

$$f_{\boldsymbol{a},b}(x) = \boldsymbol{w}\boldsymbol{x} + b \tag{10}$$

Gdje su $\boldsymbol{w}$ i b parametri, $\boldsymbol{x}$ vektor ulaznih značajki a $f_{w,b}(x)$ izlaz modela.

Razlika između ova dva modela je u funkciji pogreške. Funkcija pogreške L2 regularizirane regresije sadrži regularizaciju kao dodatan element u funkciji pogreške, što se vidi u izrazu (11):

$$E_{w,b}(x,y) = \frac{1}{N}\sum_{i=1\ldots N}(f_{w,b}(x_i) - y_i)^2 + \alpha \cdot \sum_{j=1\ldots n} {w_i}^2 \tag{11}$$

Gdje su $\boldsymbol{w}$ parametri, , $\boldsymbol{x}$ vektor ulaznih značajki, y stvarna vrijednost ciljne značajke, $f_{\boldsymbol{w}}(x)$ izlaz modela, α koeficijent regularizacije i $E_w(x,y)$ rezultat funkcije pogreške. Samostalni parametar b se ne regularizira, kako ne počinju sve skale od 0 te to ne treba penalizirati.

Vidi se da će prilikom optimizacije ove funkcije pogreške ona biti što veća ako su težine parametara veće u apsolutnoj vrijednosti, a manja ako su manje. Također je prisutan i hiperparametar α, koeficijent regularizacije koji se kreće od 0 do 1 i utječe na snagu učinka regularizacije. Ako je koeficijent regularizacije 0 onda se regularizirana regresija pretvara u običnu linearnu regresiju. (Burkov, 2019, poglavlje 5.5.)

Ideja iza regularizacije je da bi prenaučen model koristio sve dostupne značajke, često s visokim apsolutnim vrijednostima parametara, da savršeno nauči šum u ulaznim podacima i postigne što veći rezultat na testnom skupu podataka pod cijenu lošije generalizacije. Ako penaliziramo težine parametara, model će pri treniranju morati napraviti kompromis između povećanja težine parametara i korištenja novih značajki i rezultata na skupu za učenje. Tako možemo smanjiti rizik od prenaučenosti modela. (Burkov, 2019, poglavlje 5.5.)

U postupku treniranja modela regularizirane regresije optimizirao se i hiperparametar α putem iscrpne pretrage (eng. *grid search*) između svih vrijednosti od 0.01 do 1 po koracima 0.01 (0.01, 0.02, …, 0.99, 1). Prilikom metode optimiziranja hiperparametara putem iscrpne pretrage podaci su se osim u skupove za učenje i testiranje dijelili i u skup za validaciju. Razlog korištenja skupa za validaciju je taj kako naučene modele raznih hiperparametara ne želimo testirati izravno na skupu podataka za testiranje zato što time riskiramo da bude odabran neki model koji slučajno ima bolje performanse na tom skupu, umjesto modela s najboljim hiperparametrima općenito. Zato se skup za testiranje koristi samo na kraju za prikaz konačnih rezultata na podacima koji ni na koji način nisu bili uključeni u učenje modela, a hiperparametri se optimiziraju na skupu za validaciju odvojenom iz podataka za učenje. Hiperparametri se optimiziraju tako da se odabere neka funkcija pogreške, te se uzme ona vrijednost hiperparametra koja daje najmanju pogrešku na skupu za validaciju. (Burkov, 2019, poglavlje 5.3.)

U ovom radu je također korištena unakrsna validacija (eng. *K-fold cross validation*). Ideja iza unakrsne validacije je da se za pojedinu vrijednost hiperparametra ne treba ograničiti na jednu podjelu dostupnih podataka na skup za učenje i skup za validaciju kako bi se procijenio rezultat koji daje. Umjesto toga, dostupni podaci mogu se podijeliti na više načina te kao konačni rezultat uzeti prosjek rezultata za sve podjele. Za unakrsnu validaciju dostupni podaci se nasumično rasporede te podijele u K podjednako velikih podskupa. Zatim se jedan

podskup odabere za validaciju, model se istrenira na ostalima i rezultat, koji je pogreška istreniranog modela na skupu za validaciju, se zabilježi. Postupak se ponavlja dok validacija nije završena na svakom podskupu. Također se nakon završetka ovog postupka mogu na drugi način nasumično rasporediti podaci i postupak ponoviti. Konačno se kao rezultat greške za određeni hiperparametar uzme prosjek grešaka od svakog skupa za validaciju napravljenog za taj hiperparametar. U ovom radu za svaku vrijednost hiperparametra broj podskupa koji se koriste (K) bio je 10, te se cijeli postupak ponavljao 3 puta. (Burkov, 2019, poglavlje 5.7.)

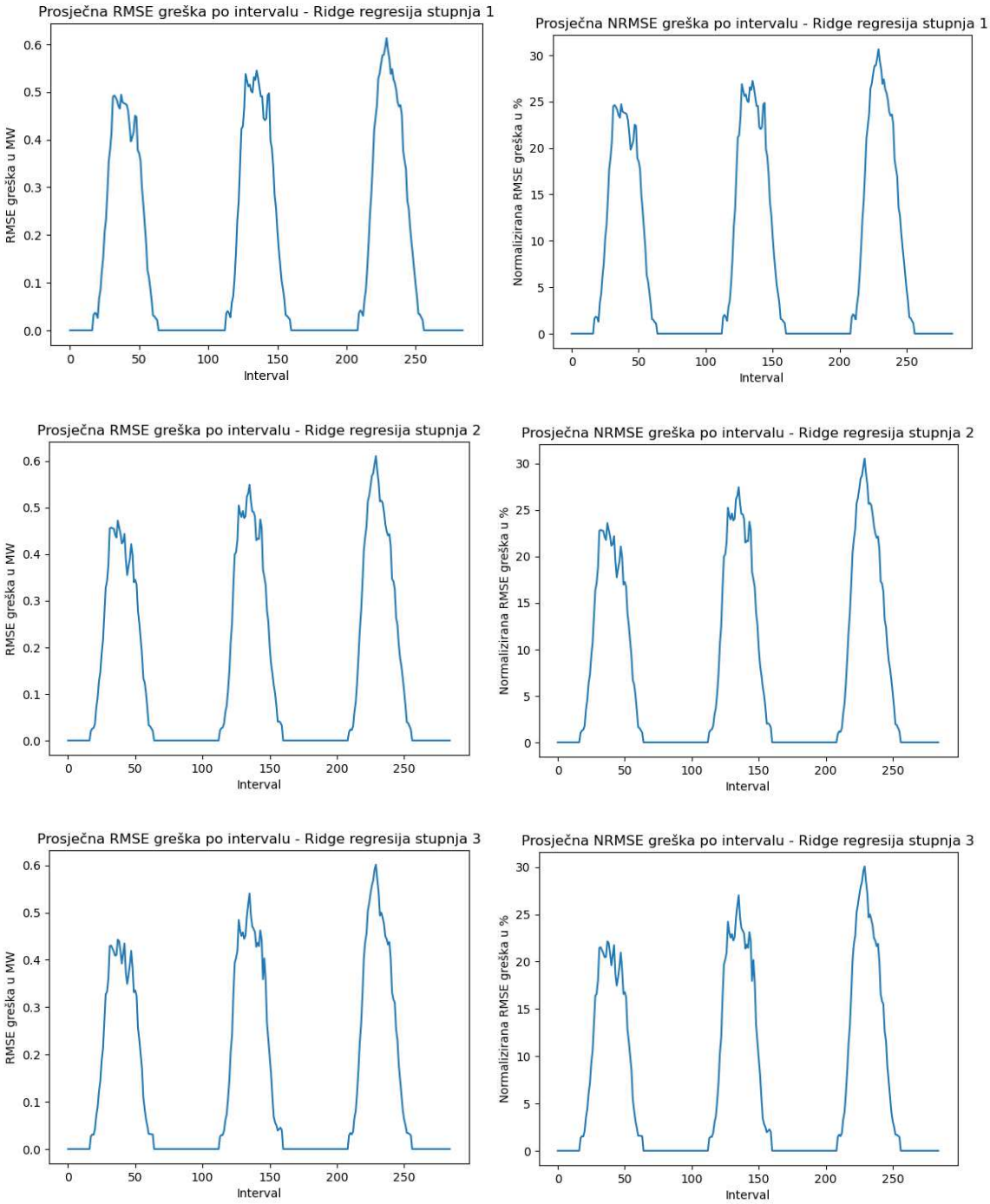
Za ovaj model također je korišten isti skup značajki kao i za model linearne regresije, **num_minutes**, **barometer**, **outtemp**, **windspeed**, **rain**, **radiation**, **cloud_cover**. Odabran je dodavanje polinomijalnih značajki 2. i 3.stupnja radi konzistentnosti s modelom linearne regresije, kako bi se vidjelo poboljšava li regularizacija rezultate. Ulazne značajke su normalizirane MinMax normalizacijom. U tablici Tablica 3.3 su prikazane prosječne greške RMSE i NRMSE za razne stupnjeve dodanih polinomijalnih značajki za ovaj model. Također je prikazan odabrani hiperparametar α:

Tablica 3.3 Prosječni rezultati predviđanja za model regularizirane regresije i odabrani hiperparametri

Model	RMSE	NRMSE	Odabrani α
Regularizirana regresija 1. stupnja	0.25 MW	14.04%	0.56
Regularizirana regresija 2. stupnja	0.23 MW	13.35%	0.01
Regularizirana regresija 3. stupnja	0.23 MW	12.89%	0.01

Isto kao i kod linearne regresije, možemo vidjeti poboljšanje u rezultatima kao posljedicu dodavanja polinomijalnih značajki. No, ne vidi se zamjetno poboljšanje rezultata u odnosu na linearnu regresiju. Također je zanimljivo da je odabrani hiperparametar α visok za regulariziranu regresiju 1. stupnja, a nizak inače. Iz analize težine značajki za linearnu regresiju 1. stupnja vidi se da ima dosta značajki koje su slabo korištene u predviđanju, što opravdava vrijednost ovog hiperparametra. No vrlo niska vrijednost, minimalna ponuđena i blizu 0, za ostale stupnjeve regularizirane regresije pokazuje da originalni model nije bio prenaučen, već je bio blizu optimalnom ili čak podnaučen.

Na slici Slika 3.4 su prikazane i prosječne greške po svakom intervalu za model regularizirane regresije:

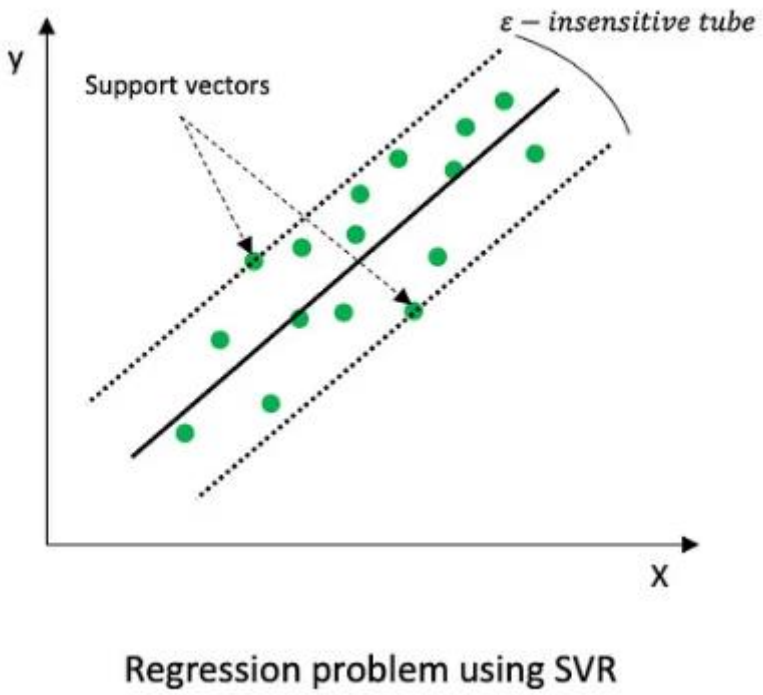


Slika 3.4 Prosječni rezultati predviđanja po intervalu za model regularizirane regresije

Kao što je i očekivano iz odabranih hiperparametara i prosječnih rezultata prikazanih u tablici Tablica 3.3, prosječne greške po intervalu su gotovo jednake za regulariziranu i linearnu regresiju. Sve u svemu možemo zaključiti da korištenje ovog modela nije donijelo značajno poboljšanje rezultata predviđanja u usporedbi s jednostavnijim modelom.

3.9. Model SVR

Model SVR funkcionira na nešto drugačiji načina od modela linearne regresije. Ideja iza ovog modela, za razliku od modela linearne regresije, je da se umjesto jedne linije koja se prilagođava podacima koristi „cijev“, odnosno marginu, s definiranim hiperparametrom širine ε oko središta. Svi primjeri koji spadaju unutar te cijevi se ne penaliziraju. Vizualizacija ideje iza ovog modela prikazana je na slici Slika 3.5 (Rasifagghihi, 2023)



Slika 3.5 Vizualizacija funkcioniranja modela SVR (Rasifagghihi, 2023)

Ne moraju svi primjeri biti unutar margine, kako bi to vjerojatno dovelo do prenaučenosti modela. Zato su dopuštene greške u kojima se primjer nalazi ispod i iznad margine. U optimizacijskom postupku se zatim balansira između toga da što više primjera u testnom skupu završi u margini i toga da smanjimo kompleksnost modela s pomoću L2 regularizacije na isti način kao u regulariziranoj regresiji. Ovaj model posljedično ima dva hiperparametra, širina margine ε i snaga regularizacije C. (Rasifagghihi, 2023)

Matematička formulacija modela za SVR ima zamjetne razlike u odnosu na model linearne regresije, te je dana u jednadžbi (12)

$$min_{w,b,\xi,\xi^*} \frac{1}{2}\sum_{j=1}^{n} w_j^2 + C\sum_{i=1}^{N}\xi_i + C\sum_{i=1}^{N}\xi_i^*$$

$$pod\ uvjetima:$$

$$\boldsymbol{w}\boldsymbol{\phi}(\boldsymbol{x_i}) + b - y_i \leq \varepsilon + \xi_i$$

$$y_i - \boldsymbol{w}\boldsymbol{\phi}(\boldsymbol{x_i}) - b \leq \varepsilon + \xi_i^*$$

$$\xi_i,\ \xi_i^* \geq 0, i = 1, \ldots N$$

(12)

gdje je $\boldsymbol{w}$ vektor parametara koji dolaze uz neku značajku, b samostalni parametar (eng. *intercept*), ξ_i i ξ_i^* greške redom veće i manje od područja margine, y_i prava vrijednost ciljne značajke, $\boldsymbol{\phi}(\boldsymbol{x_i})$ jezgrena funkcija te ε i C prethodno objašnjeni hiperparametri. (Lin, 2022):

Prvo valja objasniti što je to jezgrena funkcija. To je funkcija koja preslikava dane primjere i značajke u prostor veće dimenzionalnosti kako bi se dobilo više informacija i model mogao lakše prilagoditi podacima. Ovaj proces je na neki način sličan dodavanju polinomijalnih značajki koje je objašnjeno na prošlim modelima. Također, kako se ovdje značajke transformiraju jezgrenom funkcijom nema potrebe za dodavanjem polinomijalnih značajki. Korištena jezgrena funkcija **rbf** transformira podatke formulom (13):

$$K(X_1, X_2) = \exp\left(-\frac{d_{12}}{2\sigma^2}\right)$$

(13)

gdje je d_{12} kvadrirana euklidska udaljenost između točaka, odnosno primjera, X_1 i X_2, σ hiperparametar specifičan za rbf jezgrenu funkciju i $K(X_1, X_2)$ rezultat jezgrene funkcije. Objašnjenje ove formule je da točke iz skupa primjera za učenje zapravo postaju značajke. Nove točke za koje se treba predvidjeti se uspoređuju sa starima, te im se daje vrijednost ciljne značajke slična vrijednosti onih točaka kojima ta točka najsličnija. Vrijednost hiperparametra σ se u ovom radu nije optimizirala, već je uzeta automatski izračunata vrijednost metode u biblioteci Scikit-learn. (Sreenivasa, 2020)

Nakon objašnjenja pojma jezgrene funkcije može se lakše objasniti matematički opis ovog modela. Vidi se da u funkciju greške koja se minimizira ulazi element L2 regularizacije $\sum_{j=1}^{n} w_j^2$ koji penalizira velike vrijednosti parametara, te ostali elementi koji penaliziraju greške ξ_i i ξ_i^*. Uvjeti osiguravaju ostale objašnjene značajke modela. Bitno je napomenuti da ovdje za razliku od regularizirane regresije parametar C kontrolira snagu učinka grešaka ξ_i i ξ_i^*, ne težina parametara. No, svejedno manji C rezultira manje kompleksnim modelom kako se ne penaliziraju previše sva odstupanja od margine.

Zbog dugog vremena treniranja ovog modela optimizacija hiperparametara bila je manje detaljna. Za hiperparametar C ispitane su vrijednosti od 0.1 do 1 po koraku 0.1 (0.1, 0.2 …

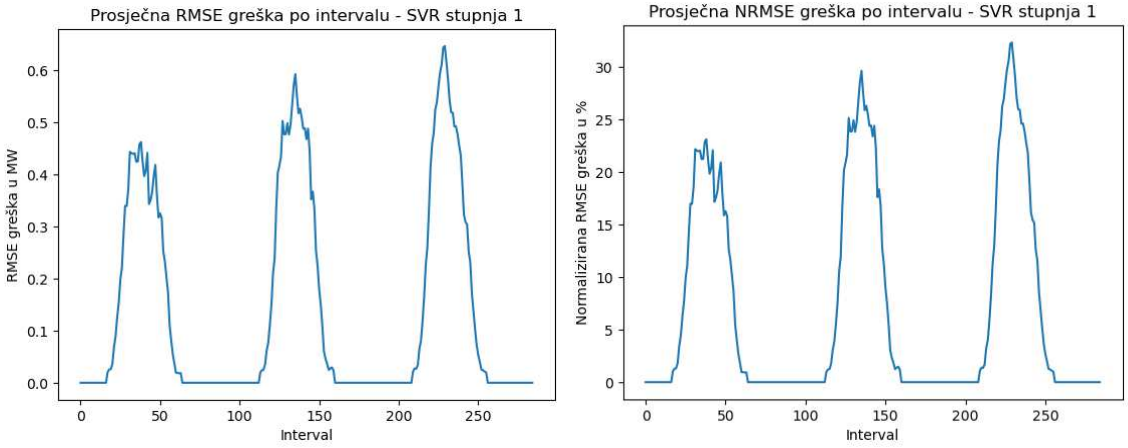
0.9, 1). Za hiperparametar ε ispitane su vrijednosti (0.05, 0.1, 0.15, 0.2) kako su se prema procjeni na temelju rezultata ostalih modela činile realnim marginama greške koje se mogu očekivati u ovom modelu. Također je korištena jednostavnija unakrsna validacija, gdje je skup podataka bio podijeljen na 5 podskupa bez ponavljanja postupka. Rezultati modela prikazani su u tablici Tablica 3.4:

Tablica 3.4 Prosječni rezultati predviđanja za model SVR i odabrani hiperparametri

Model	RMSE	NRMSE	Odabrani C	Odabrani ε
SVR	0.23	13.28%	0.1	0.05

Iz ovih rezultata vidimo da je model postigao dobar rezultat sličan ostalim kompleksnijim modelima, no ne i zamjetno bolji. Odabir hiperparametara je zanimljiv, s najužom marginom ponuđenom i najmanjim ponuđenim hiperparametrom C. Ovo je indikacija da je generalno pravilo prilično jasno zbog male margine no u podacima ima puno šuma kako se odstupanja od pravila slabo kažnjavaju.

Na slici Slika 1.1 može se vidjeti vizualizacija prosječne greške po intervalu za SVR model

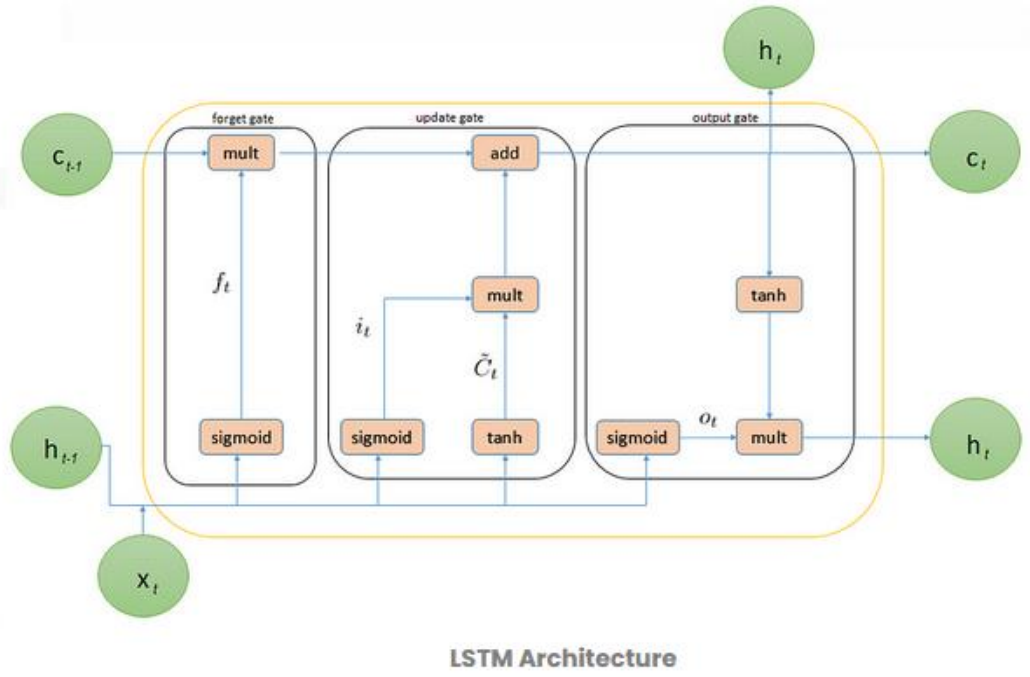


Slika 3.6 Prosječni rezultati predviđanja po intervalu za model SVR

Slično kao i s rezultatima, ne može se vidjeti ništa značajno bolje u usporedbi s ostalim modelima. Prosječna greška po intervalu je usporediva s ostalima te ne odudara od očekivanja.

3.10. Model LSTM

LSTM model je podvrsta **RNN** (eng. *Recurrent Neural Network*) modela dubokog učenja za potrebe predviđanja podataka u obliku vremenskih nizova. Glavna značajka LSTM modela je ta da koristi i dugoročnu i kratkoročnu memoriju u svrhu predviđanja ciljne značajke. LSTM modeli imaju relativno kompleksnu arhitekturu u usporedbi s modelom perceptrona objašnjenim u uvodnom poglavlju. Shema LSTM jedinice na kojoj će se protumačiti način rada ovog modela prikazana je na slici Slika 3.7:



Slika 3.7 Arhitektura LSTM jedinice (Towards AI, 2023)

Na ovoj shemi kratkoročna memorija iz prethodne jedinice je označena kao h_{t-1}, dugoročna memorija iz prethodne jedinice kao c_{t-1} i ulazni podatak kao x_t. Svaka jedinica odgovara jednom koraku vremenskog niza. Zatim se mogu vidjeti tri „vrata“ (eng. *gate*) kroz koje ti ulazni podaci prolaze. Prva vrata određuju koliko će se informacije od dugoročne memorije iz prethodne jedinice zadržati (eng. *forget gate*). Druga vrata određuju kako će se dugoročna memorija promijeniti u ovoj jedinici na temelju ulaznog podatka i kratkoročne memorije iz prethodne jedinice (eng. *update gate*). Konačno, zadnja vrata određuju kako će se kratkoročna memorija promijeniti na temelju ažurirane dugoročne memorije, ulaznog podatka i kratkoročne memorije prethodne jedinice (eng. *output gate*). Ta ažurirana vrijednost kratkoročne memorije smatra se izlazom jedinice LSTM modela. Sljedećoj jedinici u nizu se također na ulaz daju ažurirana kratkoročna i dugoročna memorija. (Towards AI, 2023)

Izrađeni modeli u ovom radu bili su relativno jednostavni. Ideja iza prvog modela je bila na temelju vremenskog niza samo vremenskih prognoza predvidjeti proizvodnju u sljedećem koraku. Za potrebu ovog modela izrađen je vremenski niz najažurnijih interpoliranih prognoza iz perspektive svake točke, odnosno početka dana u drugom mjesecu za koji se predviđa. Npr. ako se predviđa iz dana 2024-02-01 onda bi se u vremenskom nizu prognoza koristile najažurnije dostupne prognoze do tog dana, i prognoze izrađene u tom danu za ostalo. Za potrebe izrade ovih skupova podataka korišten je drugi izrađen skup podataka u kojem su podaci u obliku vremenskog niza. Konačno, za predviđenu vrijednost ciljne značajke koristio se sloj LSTM jedinica koji bi kao ulaz primalo vremenski niz od 288 15 minutna intervala, dakle 3 dana najažurnijih vremenskih prognoza iz perspektive točke iz koje se predviđa.

Drugi model je funkcionirao na ideji sličnoj prvom, no s dodatkom povijesne proizvodnje kao značajke. Kako ne postoje stvarne vrijednosti proizvodnje dostupne poslije trenutka iz kojeg predviđamo u stvarnim uvjetima, inače predviđanje uopće ne bi bilo potrebno, ovaj model je koristio rekurzivni pristup tom problemu. Predviđanja su u prvom koraku bila napravljena na temelju dostupne povijesne proizvodnje. Zatim, za korake nakon prvog prethodna predviđanja modela su se umetala u skup podataka i predviđanja su se dodatno obavljala nad njima. S pomoću ove metode nadalo se dobiti dodatnu informaciju iz prošlih vrijednosti ciljne značajke, osim samo korištenja značajki prognoze. Za potrebe ove metode koristila se ista arhitektura kao i a prethodnu, jedan LSTM sloj od 288 jedinica koji prima prethodne vremenske prognoze i povijesnu proizvodnju, stvarnu ili dodanu na temelju prethodnih predviđanja, ovisno o koraku predviđanja.

Za treniranje ova dva modela poslužili su svi kontinuirani vremenski nizovi duljine od 3 dana koji su se mogli napraviti iz skupa podataka u obliku vremenskog niza. Prilikom treniranja ovog modela bilo je potrebno odvojiti podatke i za skupa za validaciju da se može dobiti signal kada prestati s treniranjem modela, za što su poslužili svi nizovi u siječnju 2024. Za testiranje su korišteni podaci iz veljače 2024. kao što je objašnjeno u poglavlju 3.3.

Zadnja isprobana arhitektura koristila je sve dostupne prognoze iz jedne točke u pokušaju da nauči iz niza vremenskih prognoza predvidjeti cijeli niz proizvodnje odjednom, za razliku od prethodnih modela koji su predviđali proizvodnju korak po korak. Također se koristio jedan LSTM sloj duljine kao cjelokupna prognoza da je na temelju toga izračuna niz podataka za proizvodnju. Ovaj model treniran je nad svim dostupnim prognozama napravljenim u isto vrijeme do siječnja 2024, podaci iz siječnja 2024. korišteni su za validaciju a podaci iz veljače 2024. za testiranje.

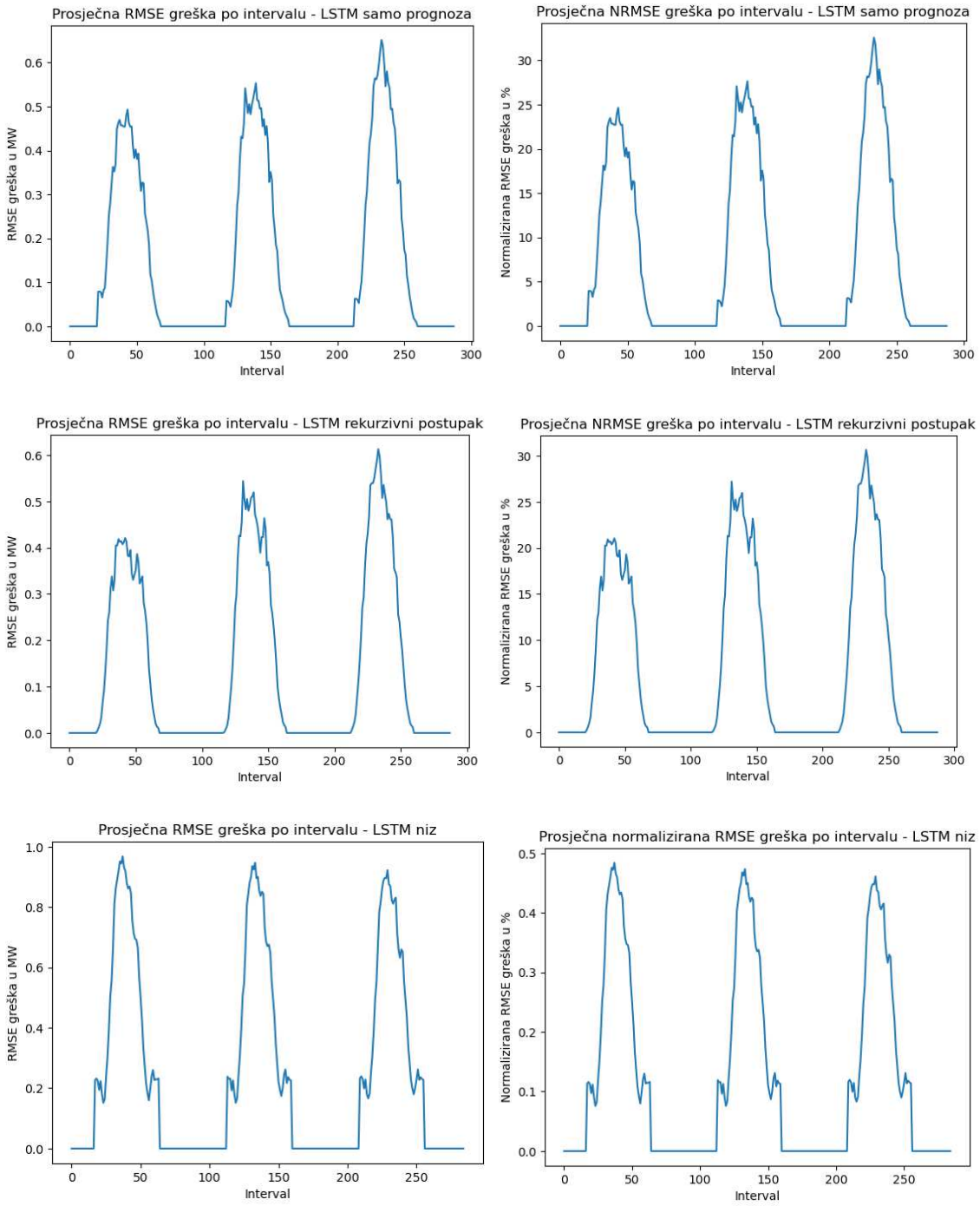
Rezultati za ova tri opisana modela redom su prikazani u tablici Tablica 3.5:

Tablica 3.5 Prosječni rezultati predviđanja za LSTM modele

Model	RMSE	NRMSE
LSTM samo prognoze	0.24 MW	13.75%
LSTM rekurzivni postupak	0.23 MW	13.16%
LSTM niz	0.47 MW	23.50%

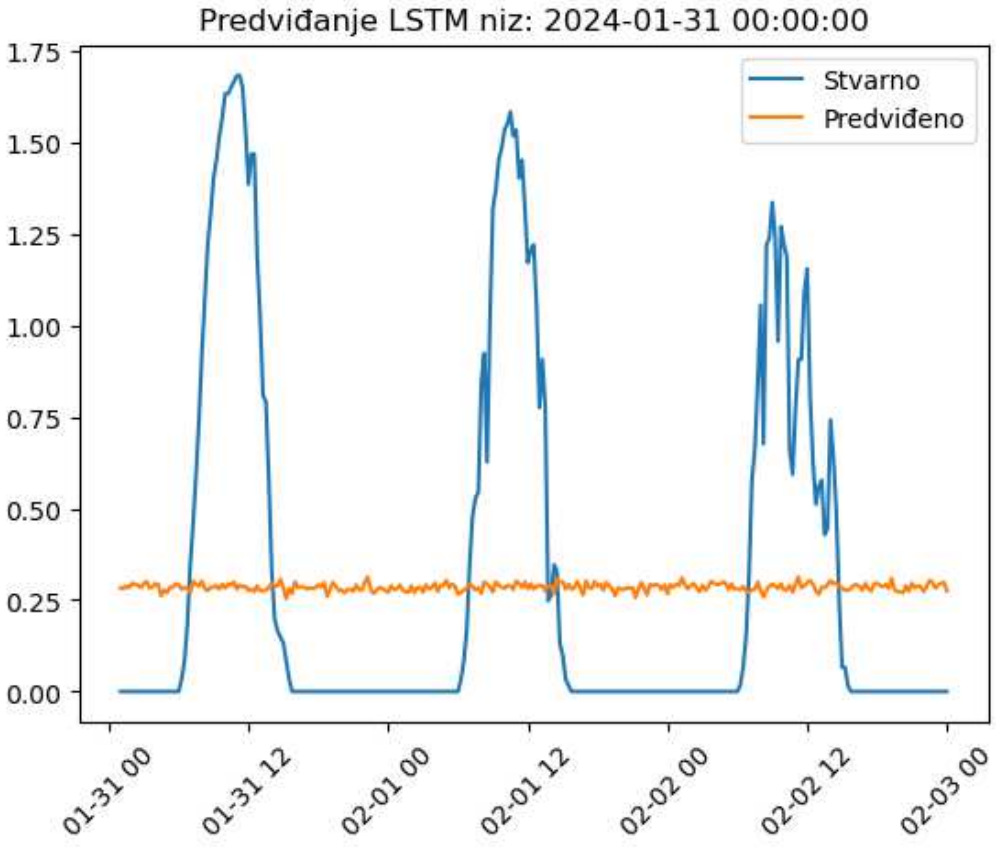
Iz rezultata se vidi da je prognoza samo s vremenskim prognozama i rekurzivno ostvarila slične rezultate. Rezultati su u skladu s ostalim kompleksnijim modelima i nisu ostvarili značajno poboljšanje. No, model koji predviđa cijeli niz odjednom ima rezultat zamjetno lošiji od svih dosadašnjih modela.

Slični se rezultati mogu vidjeti i u analizi prosječnih grešaka po intervalu prikazanoj na slici Slika 3.8:



Slika 3.8 Prosječni rezultati predviđanja po intervalu za model LSTM

Iz ovih se grafova mogu dobiti neki zanimljivi zaključci. Možemo vidjeti da model s rekurzivnim postupkom ostvaruje zamjetno bolje rezultate od modela samo s prognozama u ranijim predviđanjima oko prvog dana. Zatim, u drugom i trećem danu predviđanja počinju biti slične kvalitete. Razlog tomu je vjerojatno što se u ranijim danima mogu bolje iskoristiti prave povijesne vrijednosti, dok ih u kasnijim danima zamijene manje korisne rekurzivno dodane vrijednosti. Za model koji odjednom predviđa cijeli niz vidimo vrlo neobičan rezultat greške po intervalima Svi dani su podjednako loši i na početku dana greška naglo naraste, zatim padne pa opet naglo naraste. Razlog tome može se zaključiti iz ispod priložene slike predviđanja ovog modela, Slika 3.9, bez obrade rezultata:



Slika 3.9 Predviđanje modela LSTM niz

Vidi se da je model naučio samo prosječnu vrijednost predviđanja, nije se prilagodio obliku podataka dalje od toga. Zbog toga su rezultati vrlo loši i tek donekle ublaženi obradom. Razlog ovom lošem rezultatu učenja je vjerojatno nedostatak podataka, kako je predviđanje niza kompleksnije od predviđanja broja a nizova cijelih prognoza izrađenih u isto vrijeme koji su bili dostupni za treniranje je bilo puno manje nego nizova na kojima su istrenirani ostali modeli.

4. Diskusija

4.1. Analiza rezultata

U tablici Tablica 4.1 sažeto su prikazani rezultati svih modela, bez hiperparametara specifičnih za pojedini model Također su označeni i najbolji te najlošiji model prosuđeno prema rezultatu na skupu podataka za testiranje:

Tablica 4.1 Usporedba rezultata svih modela

Model	RMSE	NRMSE
Linearna regresija 1. stupnja	0.25 MW	14.04%
Linearna regresija 2. stupnja	0.23 MW	13.34%
Linearna regresija 3. stupnja	0.23 MW	12.94%
Regularizirana regresija 1. stupnja	0.25 MW	14.04%
Regularizirana regresija 2. stupnja	0.23 MW	13.35%
Regularizirana regresija 3. stupnja	0.23 MW	12.89%
SVR	0.23	13.28%
LSTM samo prognoze	0.24 MW	13.75%
LSTM rekurzivni postupak	0.23 MW	13.16%
LSTM niz	0.47 MW	23.50%

Iz ove tablice možemo vidjeti da se najboljim pokazao model regularizirane regresije s dodanim polinomijalnim značajkama 3. stupnja. Zanemarivo lošiji od tog modela bio je model linearne regresije 3. stupnja. Od lošijih modela valja istaknuti LSTM niz, koji se nje uspio pravilno prilagoditi podacima te najjednostavnije modele linearne i regularizirane regresije prvog stupnja.

Svi modeli imali su problema s nadilaženjem oko 13 % NRMSE greške. Čak ni zamjetna povećanja kompleksnosti modela, kao korištenje SVR ili LSTM modela, nisu rezultirala bitnim pomakom od te granice i bili su nešto lošiji od najboljih modela isprobanih prije. Precizna usporedba s rezultatima drugih radova na sličnu temu je otežana raznim faktorima kao razlikama u metodologiji, kvaliteti podataka, rezoluciji i koliko vremena unaprijed se predviđa. No mnogi slični radovi su ostvarili NRMSE grešku ispod ili oko 10 %, što pokazuje da rezultati ovog rada nisu na istoj razini kao najbolja dostupna rješenja. (Yuan-Kang Wu, 2022) (Jwaone Gaboitaolelwe, 2023) U eksperimentu provedenom u jednom od

radova također su ostvareni rezultati oko 10% NRMSE točnosti korištenjem nekih od sličnih metoda kao u ovom radu (SVR model). No, to predviđanje bilo je samo jedan dan unaprijed i u satnoj rezoluciji te je imalo više dostupnih podataka, od 2014.g. do 2017.g. Uzevši u obzir te činjenice, i da je demonstrirano da modeli u ovom radu bolje predviđaju prvi dan unaprijed nego ostale, rezultati dobiveni su usporedivi s rezultatima tog eksperimenta. (Jwaone Gaboitaolelwe, 2023).

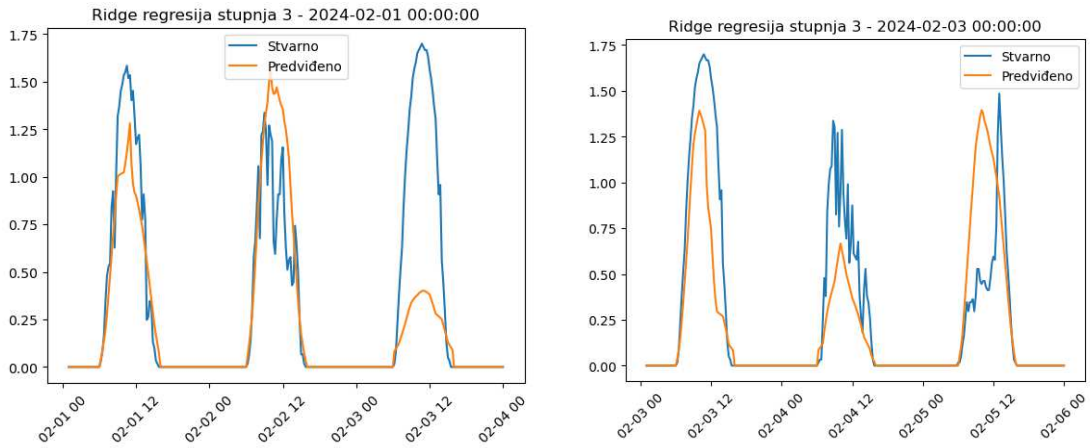
4.2. Utjecaj ograničenja na rezultate

Na dobivene rezultate utjecala su i navedena ograničenja male količine podataka i dostupnih podataka samo za jednu elektranu.

Prilikom analize uočeni su zamjetno niži rezultati u odnosu na slične radove na temu predviđanja proizvodnje solarnih elektrana. Moguće je da je to posljedica nedostatka podataka za treniranje, između ostalih problema.

Također ostaje nepoznanica kakvi bi bili rezultati za zadatak predviđanja proizvodnje solarnih elektrana u Hrvatskoj generalno. Iako su ovi rezultati i rezultati sličnih istraživanja indikacija da je strojno učenje obećavajuća metoda za rješavanje ovog problema te reda veličine greške, bez modela za druge elektrane ne može se napraviti preciznija usporedba.

Također se pouzdavanje na vremensku prognozu pokazalo kao bitan negativan utjecaj na rezultate. Na više mjesta pojavila se situacija u kojoj su za određeni ciljni trenutak predviđanja proizvodnje znatno varirala zbog razlika u vremenskoj prognozi izrađenoj u raznim trenutcima. Ovo je zamjetno povećavalo prosječne greške, i u konačnici negativno utjecalo na rezultat. Stoga, bilo kakva poboljšanja u vidu veće kvalitete vremenske prognoze i modela koji to radi bi imala pozitivan utjecaj na rezultate ovog modela. Vizualizacija ovog problema prikazana je na slici Slika 4.1. Vidi se kako za isti ciljni dan proizvodnje, krajnje desno na lijevom djelu slike, krajnje lijevo na desnom djelu slike, predviđena vrijednost znatno varira.



Slika 4.1 Razlika u predviđanju za isti ciljni stupac u ovisnosti o vremenskoj prognozi

4.3. Preporuke za budući rad

Preporuke za budući rad uglavnom su povezane s prethodno ograničenjima, te sa širim projektom izrade platforme za predviđanje proizvodnje solarnih elektrana u Hrvatskoj.

Ponovno treniranje modela nakon skupljanja veće količine podataka za istu elektranu analiziranu u ovom radu bi vjerojatno dalo bolji rezultat. Također, skupljanje podataka i izrada modela za ostale solarne elektrane bila bi nužna za izradu platforme za predviđanje koja je realno iskoristiva kao pomoć u planiranju, kako je elektrana opisana u ovom radu samo mali dio ukupnog kapaciteta koji imaju solarne elektrane u Hrvatskoj. Izrada modela za ostale elektrane bi također dala i jasniji rezultat o prosječnoj točnosti predviđanja na razini Hrvatske, a ne samo za jednu elektranu. Isto, mogli bi se isprobati drugačiji modeli prikladni za ovaj problem koji nisu isprobani u kontekstu ovog rada, kao npr. drugačije arhitekture LSTM modela.

Također, kako bi modeli opisani u ovom radu bili iskoristivi u platformi trebalo bi ih integrirati s ostalim komponentama platforme. U svrhu toga bilo bi potrebno preoblikovati kod te modele i neke elemente potrebne za transformaciju podataka kao npr. funkciju za MinMax normalizaciju bi trebalo spremiti u obliku datoteka koje se mogu pohraniti i po potrebi učitati. Za to bi bilo prikladno iskoristiti Python biblioteku **Pickle.**

Također se prema potrebi može izraditi programski kod koji bi automatizirao određene radnje vezane uz pojedini model. Npr. predviđanje proizvodnje bi se moglo automatski izvoditi kada novi podaci o vremenskoj prognozi postanu dostupni, te bi se redovno mogao obavljati ponovni trening i ažuriranje modela kada se nakupi zamjetna količina novih povijesnih podataka.

Zaključak

Uzevši u obzir rezultate rada, možemo zaključiti kako je strojno učenje prikladna metoda za predviđanje proizvodnje solarnih elektrana na temelju vremenske prognoze i povijesne proizvodnje. U sklopu ovog rada implementirani su razni modeli strojnog učenja i analizirani rezultati predviđanja, te se može zaključiti da modeli generalno daju obećavajuće rezultate, iako ima prostora za poboljšanje rezultata u usporedbi s ostalim radovima na sličnu temu.

Također je pokazano kao opravdano korištenje kompleksnijih modela i proširivanja originalnih značajki metodama kao što su dodavanje polinomijalnih značajki ili korištenjem jezgrenih funkcija u modelu SVR, zato što je to zamjetno poboljšalo rezultate u odnosu na jednostavnije korištene modele. No, također je demonstrirano da pretjerano povećanje kompleksnosti modela u vidu ovog zadatka ne daje bolje rezultate, čak i nešto lošije od modela koji su se pokazali optimalne složenosti za ovaj zadatak.

Kao neotklonjiv izvor greške pokazao se šum u ulaznim podacima. Ovi modeli su se pouzdali na izračune drugog modela, modela za vremensku prognozu, kako bi dali rezultate svojih predviđanja. Ako podaci o vremenskoj prognozi ne daju dobro predviđanje za određeni trenutak, vrlo je vjerojatno i da će rezultat predviđanja modela biti loš. Stoga, bilo kakva poboljšanja u kvaliteti ulaznih vremenskih prognoza bi se pozitivno odrazila na kvalitetu modela koje na temelju njih predviđaju proizvodnju solarnih elektrana.

Također se kao izvor problema pokazao nedostatak ulaznih podataka za treniranje. Bili su dostupni podaci od 7 cijelih mjeseci, što je smanjilo uzorak na temelju kojeg se može učiti model. Veća količina podataka, idealno iz više cijelih godina, bi dala veći uzorak iz kojeg se može bolje naučiti model i koji bolje opisuje koje su razine proizvodnje povezane s kakvim vremenom.

Sve u svemu, uzevši u obzir ograničenja koja su bila prisutna pri učenju ovih modela može se zaključiti da su rezultati zadovoljavajući i obećavajući za daljnje istraživanje. Za daljnji razvoj bilo bi preporučljivo ponoviti učenje modela kada se skupi više podataka o povijesnoj proizvodnji i pripadnim vremenskim prognozama, te napraviti modele i za druge solarne elektrane u Hrvatskoj. Također, kada se skupi više podataka možda će neke od kompleksnijih metoda isprobane u ovom radu davati bolje rezultate.

Literatura

[1] Bo Yang, T. Z. (2023). Classification and Summarization of Solar Irradiance and Power Forecasting Methods: A Thorough Review. *CSEE Journal of Power and Energy Systems*, 982 - 988.

[2] Brownlee, J. (12. kolovoz 2019). *Overfitting and Underfitting With Machine Learning Algorithms*. Dohvaćeno iz Machine Learning Mastery: https://machinelearningmastery.com/overfitting-and-underfitting-with-machine-learning-algorithms/; pristupljeno 28. lipnja 2024.

[3] Burkov, A. (2019). *The Hundred-Page Machine Learning Book.*

[4] ESO. *What is Frequency?* Dohvaćeno iz Electricity system operator for Great Britain (ESO): https://www.nationalgrideso.com/electricity-explained/how-do-we-balance-grid/what-frequency; pristupljeno 28. lipnja 2024.

[5] ESO. *What is inertia?* Dohvaćeno iz Electricity system operator for Great Britain (ESO): https://www.nationalgrideso.com/electricity-explained/how-do-we-balance-grid/what-inertia; pristupljeno 28. lipnja 2024.

[6] HOPS d.d. *Održavanje frekvencije*. Dohvaćeno iz HOPS: https://www.hops.hr/odzavanje-frekvencije; pristupljeno 28. lipnja 2024.

[7] Jwaone Gaboitaolelwe, A. M. (2023). Machine Learning Based Solar Photovoltaic Power Forecasting: A Review and Comparison. *IEEE Access*, 40831-40832, 40840.

[8] Lin, C.-C. C.-J. (2022). *LIBSVM: A Library for Support Vector Machines.* Taipei: National Taiwan University.

[9] Muelaner, J. (5. veljača 2021). *Grid Frequency Stability and Renewable Power*. Dohvaćeno iz engieneering.com: https://www.engineering.com/story/grid-frequency-stability-and-renewable-power; pristupljeno 28. lipnja 2024.

[10] National Center for Atmospheric Research. *Weather Research & Forecasting Model (WRF)*. Dohvaćeno iz Mesoscale & Microscale Meteorology: https://www.mmm.ucar.edu/models/wrf; pristupljeno 28. lipnja 2024.

[11] Rasifagghihi, N. (21. travanj 2023). *From Theory to Practice: Implementing Support Vector Regression for Predictions in Python*. Dohvaćeno iz Medium: https://medium.com/@niousha.rf/support-vector-regressor-theory-and-coding-exercise-in-python-ca6a7dfda927; pristupljeno 28. lipnja 2024.

[12] Rosenberg, M. (travanj 2020). *Air Pressure and How It Affects the Weather*. Dohvaćeno iz ToughtCo.: https://www.thoughtco.com/low-and-high-pressure-1434434; pristupljeno 28. lipnja 2024.

[13] Scikit learn. *PolynomialFeatures*. Dohvaćeno iz scikit-learn: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.PolynomialFeatures.html; pristupljeno 28. lipnja 2024.

[14] Scikit-learn *Choosing the right estimator*. Dohvaćeno iz Scikit-learn: https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html; pristupljeno 28. lipnja 2024.

[15] Sreenivasa, S. (12. listopad 2020). *Radial Basis Function (RBF) Kernel: The Go-To Kernel*. Dohvaćeno iz Medium: https://towardsdatascience.com/radial-basis-function-rbf-kernel-the-go-to-kernel-acf0d22c798a; pristupljeno 28. lipnja 2024.

[16] Towards AI. (20. lipanj 2023). *Demystifying the Architecture of Long Short Term Memory (LSTM) Networks*. Dohvaćeno iz Towards AI: https://towardsai.net/p/machine-learning/demystifying-the-architecture-of-long-short-term-memory-lstm-networks; pristupljeno 28. lipnja 2024.

[17] what's the weather like.org. *The climate of Slavonia (Croatia)*. Dohvaćeno iz what's the weather like.org: http://www.whatstheweatherlike.org/croatia/slavonia.htm ; pristupljeno 28. lipnja 2024.

[18] Yuan-Kang Wu, C.-L. H.-T.-Y. (2022). Completed Review of Various Solar Power Forecasting Considering Different Viewpoints. *energies*, 14.

Sažetak

Izrada modela strojnog učenja za predviđanje proizvodnje solarnih elektrana u Hrvatskoj

Tema ovoga rada bila je izrada modela za predviđanje proizvodnje solarne elektrane u Hrvatskoj za potrebe boljeg upravljanja elektroenergetskim sustavom. Na početku rada ukratko je objašnjena teorija iza strojnog učenja relevantna za rad. Zatim je provedena analiza podataka o vremenskoj prognozi i povijesnoj proizvodnji solarne elektrane dostupnih za treniranje modela.

Nakon toga su popisani modeli koji su korišteni u ovome radu, to jest modeli linearne regresije, regularizirane regresije, SVR i LSTM. Ukratko je objašnjena teorija iza tih modela te su prikazani rezultati predviđanja pomoću modela.

Na kraju, dan je komentar na rezultate, te smjernice za poboljšanje rada u budućnosti i daljnji razvoj modela.

Ključne riječi: Strojno učenje, Solarne elektrane, Znanost o podacima, Duboko učenje

Summary

Development of a machine learning model for forecasting the production of solar power plants in Croatia

The goal of this paper was development of a machine learning model for forecasting electrical energy production of a solar power plant in Croatia to better manage the electrical energy system. First, the theory of machine learning relevant for this paper was explained. Then the data about historical weather forecasts and historical production of the solar power plant on which the models are trained was performed.

After data analysis, the models used in this paper are listed and theory behind them is explained. The models selected were linear regression, Ridge regression, SVR and LSTM. Also, the results of all models are presented and analysed

Finally, the results of all models are discussed and suggestions for improving the results further are provided.

Key words: Machine learning, Solar power plants, Data science, Deep learning