Usporedba modela strojnog učenja za problem regresije

Medur, Patrik

SVEUČILIŠTE U RIJECI
TEHNIČKI FAKULTET
Preddiplomski sveučilišni studij računarstva

Završni rad

Usporedba modela strojnog učenja za problem regresije

Patrik Medur
0069087901

Rijeka, Lipanj 2023.

SVEUČILIŠTE U RIJECI
TEHNIČKI FAKULTET
Preddiplomski sveučilišni studij računarstva

Završni rad

Usporedba modela strojnog učenja za problem regresije

Mentor: doc. dr. sc. Goran Mauša

Rijeka, Lipanj 2023.

Patrik Medur
0069087901

Sadržaj

1. Uvod

Umjetna inteligencija je grana tehnologija sa velikim interesom na razvijanje i inovacije. Mnogo sustava današnjice koristi umjetnu inteligenciju npr. autonomna vožnja, predviđanje dionica, prikazivanje određenog sadržava korisnicima naprema njihovim osobinama, itd. Svi sustavi umjetne inteligencije imaju jednu stvar zajedničku, a to je da su na jedan način istrenirani sa mogućnošću predviđanja informacija kao ljudski mozak. Strojno učenje je svoja grana tehnologije unutar umjetne inteligencije koja spaja istrenirani model sa sustavom, te učinkovitost sustava ovisi o načinu treniranja i podacima nad kojima je treniran.

Ovim radom prikazat će se modeli strojnog učenja nad različitim podacima baza podataka: Vrijednosti će biti prikazane pomoću tablica i slika sa funkcijama pogreške, točnosti i RAPL vrijednostima koje prikazuju vrijeme i energiju potrošenu na stvaranje modela strojnog učenja, te odgovoriti na pitanje koji model je najefikasniji za rješavanje određenog problema.

2. STROJNO UČENJE

Strojno učenje (engl. *machine learning*) je grana računalnih znanosti koja razvija algoritme i modele, s kojim računalo nastoji imitirati način na koji ljudi uče nove informacije. Glavni cilj je naučiti sustav da pretpostavlja i misli isto kao i čovjek, a u pojedinim slučajevima i bolje od čovjeka. Zato strojno učenja spada u puno veću granu znanosti umjetne inteligencije (engl. *artificial intelligence*) koja pokušava emulirati razmišljanje čovjeka, dok strojno učenje identificira uzorke, stvara odluke i poboljšava se tim iskustvom.

2.1. Vrste strojnog učenja

Postoje četiri glavne vrste strojnog učenja:

- Nadzirano učenje (engl. *supervised learning*) koje trenira svoj model na temelju znanih ulaznih i izlaznih podataka
- Nenadzirano učenje (engl. *unsupervised learning*) koristi ne označene (engl. *unlabeled*) ulazne i izlazne podatke za treniranje
- Polunadzirano učenje (engl. *semi-supervised learning*) u kojem model uči iz kombinacije prijašnjih vrsta
- Ojačano učenje (engl. *reinforcement learning*) model uči iz pokušaja i pogreške uz dodijeljenu ocjenu programera

2.1.1. Nadzirano učenje

Ujedno i prvi korak unutar strojnog učenja. Nadzirano učenje se temelji na tome da nam je poznat točan izlaz, a modeli se uvježbavaju primjenom skupa ulaznih i izlaznih podataka. [1] Uvježbani model se koristi za predviđanje, a tu primjenu možemo vidjeti u postupcima predviđanja diskretnih odaziva ili klasifikacija (engl. *classification*) i predviđanje kontinuiranih odaziva ili regresija (engl. *regression*).

Klasifikacija koristi ulazne podatke razvrstane u kategorije s ciljem prepoznavanja obrasca i dobivanje rezultata koji se nalazi u jednom od mogućih stanja. Npr. imamo prepoznavanje brojeva sa slike, broj može biti između 0-9. Dobiveni rezultat će uvijek biti unutar tih granica samo postotak točnosti će se mijenjati.

Regresija se razlikuje u tome što podaci mogu biti realni brojevi $\mathbb{R}$. Rezultati predviđanja moraju biti drugačije mjereni jer je statistički gotovo nemoguće dobiti točnu vrijednost za rezultat, te

točnost mjerimo pomoću R^2 (engl. *R-squared*).

2.1.2. Nenadzirano učenje

U ovom modelu imamo nepoznate izlazne vrijednosti, tj. nema izlaza koji je mapiran s ulazom. Sustav sam uči od unosa podataka i otkriva skrivene uzorke. Najčešća tehnika je grupiranje (engl. *clustering*), pri čemu se traže skriveni obrasci ili grupe. [1] Npr. isti oblik, boja, cijena, veličina, itd. Ovaj pristup je korišten ako imamo veliki broj ne označenih podataka koji se mogu organizirati bez intervencije ljudi.

2.1.3. Polunadzirano učenje

Ovaj postupak se nalazi u sredini prijašnja dva načina. Razlika je u tome što koristi manji dio podataka kojem je poznat ulaz i izlaz kao kod nadziranog učenja, ali koji predvodi veći ne označeni dio podataka kao kod nenadziranog učenja. [2 Ovaj princip je jako koristan ako nemamo dovoljno poznatih podataka ili je preskupo staviti oznake na sve podatke.

2.1.4. Ojačano učenje

U ovoj vrsti učenja, algoritam uči pomoću mehanizma povratnih informacija i prošlih iskustava. [2] Algoritam ne uči prema predefiniranim podacima, nego uz pokušaje i pogreške. Niz uspješnih ishoda će biti ekstra nagrađen kako bi dobili najbolju preporuku, politiku za problem.

2.2. Algoritmi

Algoritmi su pristupi koje sustav upotrebljava za učenje i stvaranje modela kao rješenje problema. Ovim radom pokazane su učinkovitosti algoritama na problemima regresije, te korišteni su sljedeći algoritmi:

- Linearna regresija (engl. *linear regression*)
- Lasso (akronim od engl. *Least Absolute Shrinkage and Selection Operator*)
- Ridge regresija
- Elastic-Net regresija
- KNN (akronim od engl. *K-Nearest Neighbors*)
- SVR (akronim od engl. *Support Vector Regression*)
- MLPRegressor (*MLP* akronim od engl. *Multi-Layer Perceptron*)

2.2.1. Linearna regresija

Linearna regresij je prvi korišteni algoritam u ovom radu, a temelji se na predviđanju numeričkih vrijednosti linearnog odnosa varijabli. Nastali model predviđa linearnu vezu između nezavisnih (ulaznih) i zavisnih (izlaznih) varijabli, te pronalazi optimalnu liniju koja najbolje opisuje podatke.[3] Formula za više nezavisnih varijabli je prikazana izrazom 2.1:

$$y = w_0 + w_1 x_1 + w_2 x_2 + ... + w_n * x_n \tag{2.1}$$

gdje w_0 je odsječak na y-osi, a $w_1, w_2, ..., w_n$ su koeficijenti nagiba za svaku nezavisnu varijablu $x_1, x_2, ..., x_n$.

2.2.2. Lasso regresija

Lasso regresija je vrsta linearne regresije koja koristi regularizaciju za smanjenje podataka. Dovodom kaznenog izraza (L_1) funkciji linearne regresije je tehnika sprječavanja prekomjerene prilagodbe (engl. *overfitting*). Lasso model može biti jako koristan kada ga koristimo nad visokodimenzionalnim skupovima podataka (engl. *high-dimensional datasets*), u kojim broj varijabli ili značajki (engl. *features*) je isti ili veći od broja opažanja (engl. *observations*). [4][5] Korištenjem linearne regresije nad visoko-dimenzionalnim skupovima podataka, dobivamo model koji ima dobru performansu nad učenim podacima, ali dovodom novih ne viđenih podataka bi rezultiralo lošom performansom. Lasso regresija rješava problem nalaženjem najvažnijih značajki, te smanjenjem kompleksnosti modela. Lasso regresija je predstavljena izrazom 2.2:

$$\hat{\beta}^{lasso} = \underset{\beta}{argmin}\left\{\frac{1}{2}\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p} X_{ij}\beta_j)^2 + \lambda\sum_{j=1}^{p}|\beta_j|\right\} \tag{2.2}$$

Parametar podešavanja (λ) ili poznat kao parametar kažnjavanja (engl. *penalty parameter*) kontrolira jačinu upotrijebljene kazne. Ako $\lambda = 0$ niti jedan parametar neće biti elimiriran, povećavanjem λ povećava se broj koeficijenta postavljenih na nulu i njihovu eliminiciju, te ako $\lambda = \infty$ svi koeficijenti su eliminirani. Također možemo reći da povećavanjem λ povećavamo pristranost (engl. *bias*) dok smanjenjem λ povećavamo varijancu (engl. *variance*).

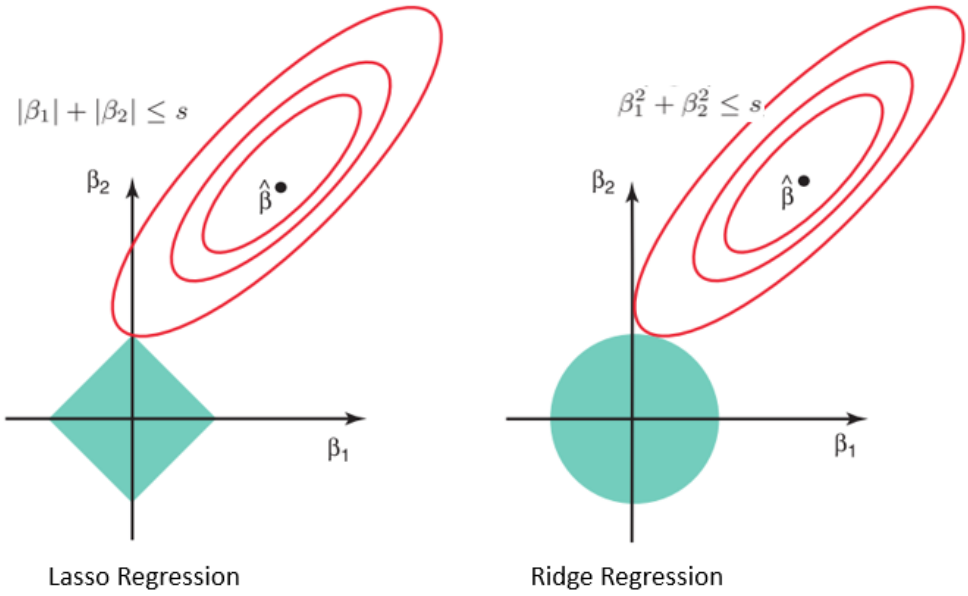
2.2.3. Ridge regresija

Ridge regresija je slična lasso regresiji. Ovaj algoritam također proizlazi od linearne regresije i ima ulogu sprečavanja prekomjerene prilagodbe, ali problem rješava na drugi način. Ridge također dovodi kazneni izraz (L_2). Razlika kaznenih izraza je što (L_1) uzima veličine koeficijenata (engl. *magnitude of the coefficients*), a (L_2) kvadrat koeficijenata. [5] Ovo možemo vidjeti na slici 2.1.

Ridge regresija je predstavljena izrazom 2.3:

$$\hat{\beta}^{ridge} = \underset{\beta}{argmin}\left\{\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p} X_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2\right\} \tag{2.3}$$

Parametar podešavanja (λ) je opet prisutan, te ako $\lambda = 0$ rezultat iznosi regresiji najmanjih kvadrata (engl. *least squares regression*). Ako $\lambda = \infty$ svi podaci se nalaze u nuli, za najbolje rezultate kazneni izraz je poželjno imati između dva ekstrema.



Slika 2.1: Razlika između Lasso i Ridge regresijskog algoritma, preuzeto iz [4] i preuređeno

2.2.4. Elastic-Net regresija

Elastic-Net regresija je vrsta linearne regresije treniran sa oba kaznena izraza (L_1) i (L_2). Ova kombinacija omogućuje učenje raspršenog modela (engl. *sparse model*) gdje je nekoliko težina različito od nule kao kod lasso, uz održavanje svojstva regulacije kao kod ridge-a.[3] Elastic-Net regresiju možemo prikazati izrazom 2.4:

$$\hat{\beta}^{enet} = \underset{\beta}{argmin}\left\{\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p} X_{ij}\beta_j)^2 + \lambda_2 \sum_{j=1}^{p} \beta_j^2 + \lambda_1 \sum_{j=1}^{p} |\beta_j|\right\} \tag{2.4}$$

Parametar podešavanja (λ) se nalazi između ili u 0 i 1. Ako $\lambda = 0$ onda naš rezultat je isti rezultatu lasso modela, te ako $\lambda = 1$ rezultat je isti rezultatu ridge modela. Za najbolje rezultate λ se mora nalaziti između ekstrema da bi imalo smisla koristiti ovaj model.

2.2.5. K-nearest neighbors (*KNN*) regresija

K-nearest neighbors je najkorišteniji model strojnog učenja. Model uči na principu neparametarske regresije (engl. *non-parametric regressionl*) tj. hipoteza nije eksplicitno definirana. KNN regresija koristi princip sličnosti - pronalazi K najbližih susjeda na temelju udaljenosti između značajki (atributa) primjera u skupu podataka. [6] Izračun predviđanja vrijednosti u KNN regresiji je prikazano izrazom 2.5:

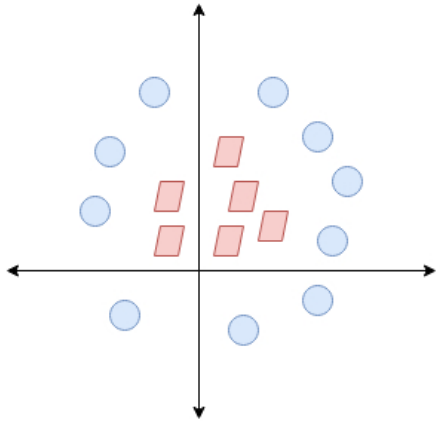
$$y_{pred} = (\frac{1}{K}) \sum_{j=1}^{k} y_{i_j} \tag{2.5}$$

Udaljenost između značajki može biti izračunan na više načina. Kod kontinuiranih varijabli imamo tri načina Euklidska distanca (engl. *Euclidean distance*), Manhattan distanca i Minkowska distanca. U nastavku svi podaci su trenirani Euklidanskom distancom koja je prikazana izrazom 2.6:

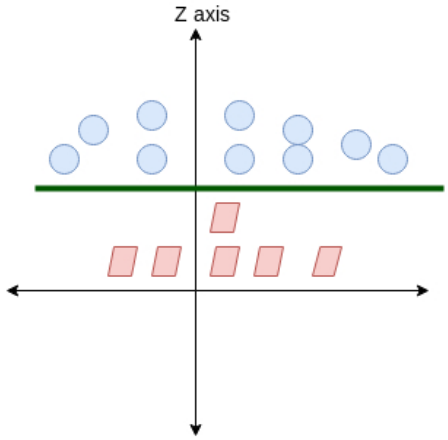
$$\sqrt{\sum_{i=1}^{k} (x_i - y_i)^2} \tag{2.6}$$

2.2.6. Support Vector Regression (*SVR*)

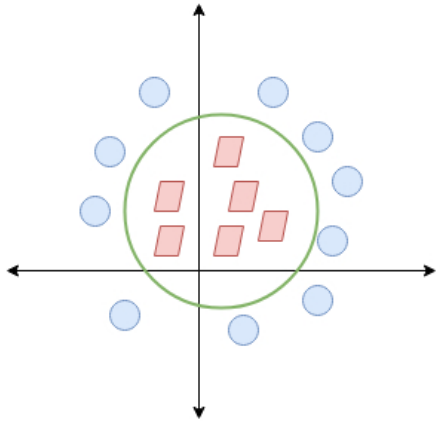
Support Vector Machine (*SVM*) je klasifikacijski algoritam koji razdvaja podatke u različite klase pomoću hiperravnine (engl. *hyperplane*). Linija / hiperravnina razdvajanja se često nalazi u n-dimenziji s kojom minimizira grešku unutar klasifikacije. [7] Ovo možemo vidjeti na slici 2.2 gdje dodavanjem z-osi dobivamo novu perspektivu i rješenje problema.



(a) Vrijednosti unutar xy kordinatnog sustava



(b) Dodavanje z osi za novu perspektivu



(c) Rješenje problema

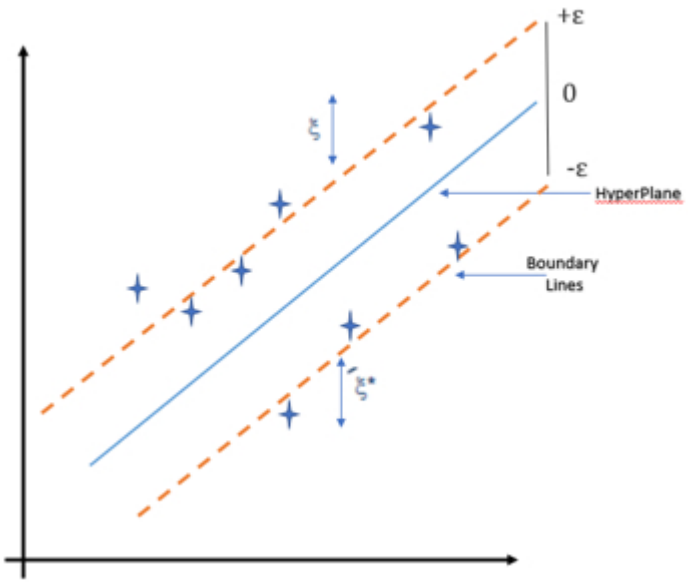
Slika 2.2: Prikaz načina rada SVM algoritma, preuzeto sa [7]

Support Vector Regression (*SVR*) je jako sličan (*SVM-u*), razlika je u kontinuiranim (engl. *continuous*) podacima naspram kategoričkim (engl. *categorical*) podacima. *SVR* algoritam nalazi hiperravninu koja prolazi kroz najveći broj podataka uz određenu distancu, poznata kao margina (engl. *margin*). [7][8] Margina ili ϵ se pokušava maksimizirati između hiperravnine i najbližih podataka uz minimiziranje pogreške. Svi podaci koji nisu unutar margine se zovu *Support Vector*, a udaljenost između vanjskog podatka i margine je označena sa ζ. ζ podaci pomažu u kreiranju margine jer pokazuju kolika greška je tolerirana. Kao što je prikazano na slici 2.3. Izračun hiperravnine je prikazano izrazom 2.7:

$$f(\vec{x}) = \vec{w} + \vec{x} + b \tag{2.7}$$

Ograničenja su prikazana izrazom 2.8:

$$\begin{aligned} y_i - wx_i - b &\leq \epsilon + \zeta_i \\ wx_i + b - y_i &\leq \epsilon + \zeta_i^* \\ \zeta_i, \zeta_i^* &\geq 0 \end{aligned} \tag{2.8}$$



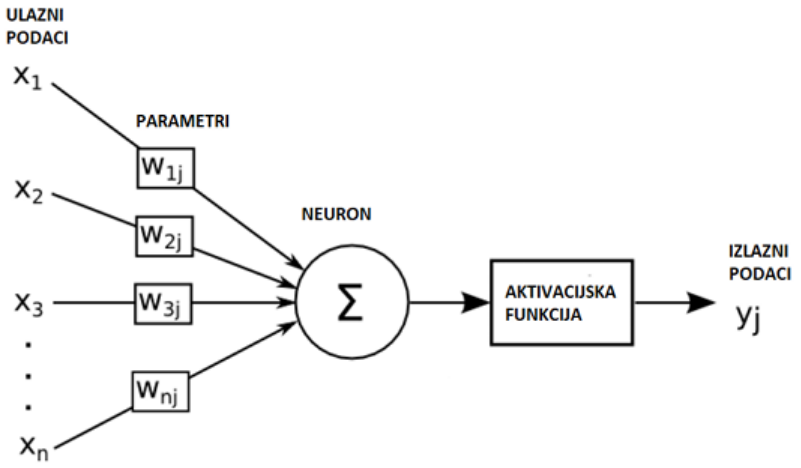
Slika 2.3: Prikaz SVR algoritma, preuzeto sa [8]

2.2.7. MLPRegressor

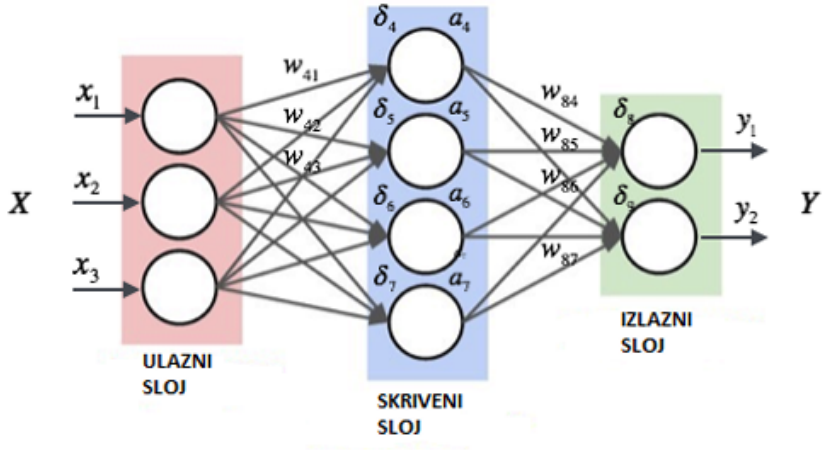
Umjetna neuronska mreža (engl. *Artificial neural network*) je algoritam koji pokušava imitirati biološke neuronske mreže ljudskog mozga s ciljem ostvarivanja umjetne inteligencije. Umjetna neuronska mreža se sastoji od neurona međusobno povezanih sa različitim težinama (engl. *weights*) koje određuju rezultat. Učenje umjetna neuronske mreže nije ništa drugo osim podešavanja težina dok ne dobijemo odgovarajući izlaz. Ovo možemo prikazati slikom 2.4 gdje vrijednost novog sloja neurona ovisi o sumi svih spojenih neurona prijašnje razine sa njihovim težinama. Izračun neurona je prikazan izrazom 2.9:

$$y = f(\sum_{i=1}^{n} x_i w_i) \tag{2.9}$$

Umjetna neuronska mreža se sastoji od više slojeva koje možemo razvrstati u tri sloja: ulazni sloj, skriveni sloj i izlazni sloj. Skriveni sloj je jedini sloj o kojem trebamo brinuti jer u njemu namještamo broj slojeva i neurona . Prilagođavanjem ovih parametara dovodi do boljih / lošijih rezultata. Višeslojnu neuronsku mrežu možemo vidjeti na slici 2.5.



Slika 2.4: Izračun neurona unutar neuronske mreže, preuzeto sa [9] i preuređeno



Slika 2.5: Izračun neurona unutar neuronske mreže, preuzeto sa [9]

MLPRegressor algoritam je vrsta umjetne neuronske mreže, omogućuje visoki stupanj točnosti u radu sa složenim nelinearnim skupovima podataka. Model MLPRegressora koristi širenje unazad (engl. *backpropagation*) tj. za vrijeme treniranja modela algoritam uzima stopu pogreške širenja prema izlazu i vrača tu vrijednost unatrag kroz slojeve neuronske mreže da dobijemo bolje podešavanje težina. Algoritmi umjetnih neuronskih mreža uvijek imaju parametre koji se mogu namjestiti za dobivanje boljih rezultata, pa MLPRegressor nije ništa drugačiji. [3] [9] Od mogućnosti namještanja različitih funkcija, rješavaća (engl. *solver*) za optimizaciju, do koliko skrivenih slojeva i neurona koristimo. Unutar ovog rada koristimo početne postavke više o njima kada prikažemo podatke, sa promijenjenim brojem skrivenih slojeva i neurona.

2.3. Prikaz informacija

Za razumijevanje dobivenih modela koristimo funkcije pogreške i točnosti za prikaz rezultata. Za bolje razumijevanje rezultati u nastavku su prikazani u slikovnom i brojčanom obliku. Postoje razne funkcije pogreške, a korištene su:

Mean Squared Error (*MSE*) je funkcija pogreške koja nam govori koliko je regresijska linija blizu skupu podataka. Ovu vrijednost dobivamo kvadriranjem razlike između podataka i regresijske linije. Kvadrat nam je potreban kako bismo uklonili sve negativne predznake, te također daje veću težinu razlikama tj. podaci koji su udaljeniji od regresijske linije su više penalizirani s obzirom na one s manjim odstupanjem. [10] MSE je prikazan izrazom 2.10:

$$MSE = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n} \tag{2.10}$$

gdje y_i je stvarna vrijednost, a $\hat{y}_i$ predviđena vrijednost.

Mean Absolute Error (MAE) je funkcija pogreške kojoj rješenje proizlazi iz apsolutnog prosjeka razlike između stvarnih i predviđenih vrijednosti. Najveća razlika naspram MSE je da MAE ne penalizira velika odstupanja više od manjih. [10] MAE je prikazan izrazom 2.11:

$$MAE = \frac{\sum_{i=1}^{n} |y_i - \hat{y}_i|}{n} \tag{2.11}$$

gdje y_i je stvarna vrijednost, a $\hat{y}_i$ predviđena vrijednost.

Root Mean Square Error (RMSE) je standardna devijacija pogrešaka predviđanja. RMSE funkcija pogreške mjeri udaljenost grešaka od regresijske linije, te pokazuje nam koliko su greške raspršene od idealne regresijske linije. [10] RMSE je prikazan izrazom 2.12:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (\hat{y}_i - y_i)^2}{n}} \tag{2.12}$$

gdje y_i je stvarna vrijednost, a $\hat{y}_i$ predviđena vrijednost.

Ovim radom svi modeli su korišteni na problem regresije, te prikaz točnog predviđanja nad takvim podacima nije moguć. Time koristimo drugačiji način prikaza parametara točnosti pomoću R Squared (R^2). R^2 je mjera koja predstavlja udio varijante za zavisni podatak koja je objašnjena nezavisnim podatkom u regresijskom modelu, tj. opisuje koliko dobro model odgovara podacima ili koliko ih dobro opisuje. [10][11] R^2 je prikazan izrazom 2.13:

$$R^2 = \frac{\sum_{i=1}^{n} (\hat{y}_i - \bar{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y}_i)^2} \tag{2.13}$$

gdje y_i je stvarna vrijednost, $\bar{y}_i$ je srednja vrijednost od y, a $\hat{y}_i$ je predviđena vrijednost.

Adjusted R Squared je modificirana verzija (R^2). Modificirana je na način da povećava svoju vrijednost kada novi podatak poboljša predviđanje modela, a smanjuje svoju vrijednost kada novi podatak ne poboljša predviđanje modela za dovoljnu vrijednost. Adjusted R^2 će zbog ovoga uvijek biti manji ili jednak od R^2.[10] Adjusted R^2 je prikazan izrazom 2.14:

$$Adjusted R^2 = 1 - \frac{(1 - R^2)(N - 1)}{N - p - 1} \tag{2.14}$$

gdje R^2 je rješenje formule 2.13, N je broj količine podataka, a p je broj nezavisnih varijabli.

2.4. Running Average Power Limit (*RAPL*)

Running Average Power Limit je sučelje za prikaz korištene energije i vremena za procesore. U ovoj svrsi, RAPL je knjižnica (engl. *library*) koja koristi Intel procesore u svrhu prikaza korištene

energije. Knjižnicu je moguće koristiti samo na Linux operativnim sustavima zbog mogućnosti pristupa procesoru. Parametri energije i vremena su iskorišteni za prikaz performansi različitih algoritama (spomenuti algoritmi pod poglavljem 2.2). [12] Primjer, ako imamo 2 modela strojnog učenja kojima su vrijednosti funkcije pogreške i točnosti slični, onda je u većini slučajeva bolje koristiti model koji za svoje treniranje koristi manje energije i kraće vrijeme treniranja.

2.5. Strojno učenje u Scikit Learn

Scikit Learn ili sklearn je jedna od najkorištenijih *open source* knjižnica za strojno učenje u Python programskom jeziku. Sklearn knjižnica sadrži razne algoritme strojnog učenja za probleme regresije, klasifikacije, grupiranja i smanjenju dimenzionalnosti. [3] Važno je napomenuti da glavna funkcija Scikit Learn knjižnice je stvaranje modela strojnog učenja, ali postoji i mogućnost manipulacijom nad podacima. Ova knjižnica nije najbolja za manipulaciju nad podacima, te za manipulacijom podataka korištene su druge bolje knjižnice, npr. NumPY i Pandas.

Razlog korištenja knjižnice Scikit Learn proizlazi iz jednostavnosti shvaćanja i korištenja algoritama strojnog učenja iz perspektive početnika. Ovo naravno ne znači da knjižnica nije namijenjena za veće projekte, nego da je lagano za shvatiti cijelu strukturu algoritama. Velika zajednica uzdržava i unapređuje knjižnicu od 2007. godine i popularnost raste izdavanjem novih verzija.

3. BAZE PODATAKA

3.1. UCI Machine learning repository

UCI Machine learning repository je kolekcija baza podataka kojima je glavna funkcija treniranje modela strojnog učenja. [13] UCI kolekcija je nastala 1987. godine, kao rješenje problema javno dostupnih baza podataka: Zbog jednostavnosti i raznolikosti u podacima, kolekcija je postala odličan alat za bilo kojeg studenta, edukatora i istraživača koji treba isprobati model strojnog učenja na neku pripremljenu baza podataka. Arhiva za sada sadrži 622 instanci baza podataka koji su uglavnom podijeljeni na zadanu zadaću (engl. *default task*), a zadaće su klasifikacija, regresija, grupiranje i ostalo.

3.2. Prikaz baza podataka

Za izradu ovog rada iskorištene su četiri baze podataka uzete iz arhive UCI Machine learning repositor. Svaka baza podataka ima zadanu zadaću regresije i minimum od 5000 instanci. Tri od četiri baze podataka imaju slični princip tražene vrijednosti, koja je energija potrebna za određenu tvorevinu. Dok zadnja ujedno i prva korištena baza podataka je Seoul Bike Sharing Demand Data Set. Ova Baza podataka traži vrijednost prirodnog broja koji predstavlja količinu slobodnih bicikli za najam u određenom trenutku, u glavnom Južno Korejskom gradu Seoulu. Baza podataka je prikazana slikom 3.1.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8760 entries, 0 to 8759
Data columns (total 14 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   date             8760 non-null   object
 1   bike_count       8760 non-null   int64
 2   hour             8760 non-null   int64
 3   temp             8760 non-null   float64
 4   humidity         8760 non-null   int64
 5   wind             8760 non-null   float64
 6   visibility       8760 non-null   int64
 7   dew_temp         8760 non-null   float64
 8   sunlight         8760 non-null   float64
 9   rain             8760 non-null   float64
 10  snow             8760 non-null   float64
 11  season           8760 non-null   object
 12  holiday          8760 non-null   object
 13  functioning_day  8760 non-null   object
dtypes: float64(6), int64(4), object(4)
memory usage: 958.2+ KB
```

Slika 3.1: Prikaz početne Seoul Bike Sharing Demand baze podataka

Atributi su: temperatura [°C], vlažnost (engl. *Humidity*) [%], brzina vjetra [$\frac{m}{s}$], vidljivost [m], dew point temperatura [°C], solarna radijacija [$\frac{MJ}{m^2}$], kiša [mm] i snijeg [cm] su vremenske informacije,

zatim imamo informacija o godišnjem dobu, praznicima, radni / neradni dan, datum, vrijeme, i broj iznajmljenih bicikli.

Druga baza podataka (Steel Industry Energy Consumption Data Set) prikazuje količinu potrošene energije za određeni dan i vrstu proizvoda. Podaci su arhivirani od strane južno korejske kompanije DAEWOO Steel Co. Ltd. [13] Više o ovoj bazi podataka možemo vidjeti na slici 3.2.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 35040 entries, 0 to 35039
Data columns (total 11 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   date                  35040 non-null  object
 1   Usage_kWh             35040 non-null  float64
 2   Lagging_Power         35040 non-null  float64
 3   Leading_Power         35040 non-null  float64
 4   CO2                   35040 non-null  float64
 5   Lagging_Power_Factor  35040 non-null  float64
 6   Leading_Power_Factor  35040 non-null  float64
 7   NSM                   35040 non-null  int64
 8   WeekStatus            35040 non-null  object
 9   Day_of_week           35040 non-null  object
 10  Load_Type             35040 non-null  object
dtypes: float64(6), int64(1), object(4)
memory usage: 2.9+ MB
```

Slika 3.2: Prikaz početne Steel Industry Energy Consumption baze podataka

Atributi su: kontinuirana potrošnja energije u industriji [*kWh*], kontinuirano kašnjenje struje jalove snage [*kVarh*], kontinuirano vodeća struja jalove snage [*kVarh*], kontinuiran tCO2 (CO2) [*ppm*], kontinuirano kašnjenje struje faktora snage [%], kontinuirano kašnjenje struje faktora snage [%], kontinuiran broj sekundi od ponoći [*s*], radni / neradni dan, dan u tjednu i vrsta proizvoda razvrstana na lagani, srednji i teški teret.

Treća baza podataka (Gas Turbine CO and NOx Emission Data Set) prikazuje dobivenu energiju plinskih turbina u turskoj. Osim dobivene energije ova baza podataka također pokazuje i dobivene plinove, koji su ugljični monoksid (CO) i dušikov oksid (NOX) kao loše strane plinskih turbina. [13] Više informacije je prikazano na slici 3.3.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7411 entries, 0 to 7410
Data columns (total 11 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   AT      7411 non-null   float64
 1   AP      7411 non-null   float64
 2   AH      7411 non-null   float64
 3   AFDP    7411 non-null   float64
 4   GTEP    7411 non-null   float64
 5   TIT     7411 non-null   float64
 6   TAT     7411 non-null   float64
 7   TEY     7411 non-null   float64
 8   CDP     7411 non-null   float64
 9   CO      7411 non-null   float64
 10  NOX     7411 non-null   float64
dtypes: float64(11)
memory usage: 637.0 KB
```

Slika 3.3: Prikaz početne Gas Turbine CO and NOx Emission baze podataka

Atributi su: sobna temperatura (AT) [°C], tlak okoline (AP) [mbar], vlažnost okoline (AH) [%], razina tlaka zračnog filtra [AFDP] [*mbar*], ispušni tlak plinske turbine (GTEP) [*mbar*], početna temperatura turbine (TIT) [°C], završna temperatura turbine (TAT) [°C], ispusni tlak kompresora (CDP) [*mbar*], dobivena energija (TEY) [*MWH*], Ugljični monoksid (CO) $[\frac{mg}{m^3}]$ i dušikov oksid (NOX) $[\frac{mg}{m^3}]$.

Zadnja baza podataka (Power consumption of Tetouan city Data Set) prikazuje dnevnu količinu energije za Tetouan graf u sjevernom Moroku. Ova baza podataka se sastoji od 3 atributa za predviđanje. Zbog nemogućnosti predviđanja nad više od jednog atributa sa većinom modela, predviđanje se vrši nad svakim atributom zasebno. Više informacija je prikazano slikom 3.4.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 52416 entries, 0 to 52415
Data columns (total 9 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   DateTime                   52416 non-null  object
 1   Temperature                52416 non-null  float64
 2   Humidity                   52416 non-null  float64
 3   Wind Speed                 52416 non-null  float64
 4   general diffuse flows      52416 non-null  float64
 5   diffuse flows              52416 non-null  float64
 6   Zone 1 Power Consumption   52416 non-null  float64
 7   Zone 2  Power Consumption  52416 non-null  float64
 8   Zone 3  Power Consumption  52416 non-null  float64
dtypes: float64(8), object(1)
memory usage: 3.6+ MB
```

Slika 3.4: Prikaz početne Power consumption of Tetouan city baze podataka

Atributi su: Vrijeme (svakih 10 minuta), temperatura [°C], vlažnost [%], brzina vjetra $[\frac{m}{s}]$, opći difuzni tokovi, difuzni tokovi i potrošena energija za 3 zone grada.

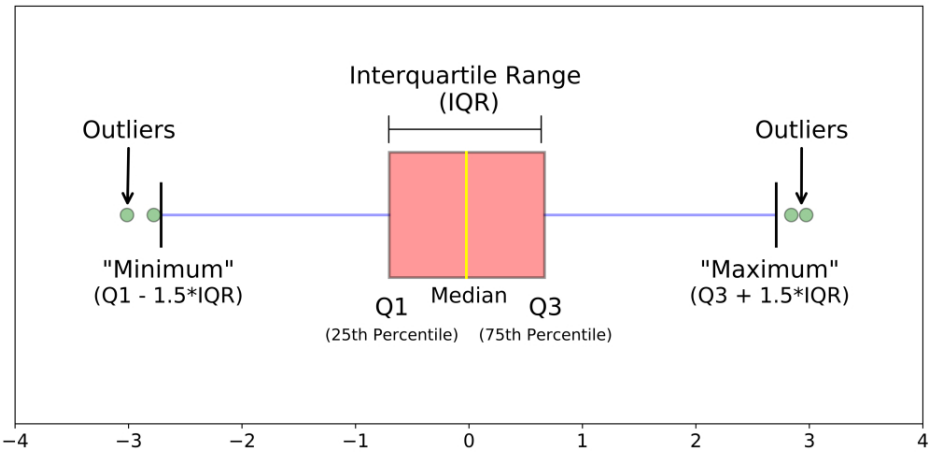
3.3. Pretprocesiranje

Za prikaz završnih podataka svaka baza podataka je prošla kroz par promjena koje su prikazane u nastavku. Promjene su napravljene za bolje rezultate, a točne promjene se razlikuju po potrebi samih podataka. Na kraju svega izračunata je energija i vrijeme potrebno za pripremanje podataka strojnog učenja.

3.3.1. Tehnike promjene i prikaza podataka

Tehnike promjene i prikaza podataka korištene ovim radom su uklanjanje ekstrema pomoću IQR (akronim za engl. *interquartile range*) i stavljanja granica (engl. *capping*), kodiranje novih atributa kao zamjena za stare, prikaz informacija nad toplinskim kartama i prikaz iskošenosti podataka, te moguća rješenja.

Interkvartil *IQR* je mjera raspršenja skupa podataka. Interkvartil je definiran u više koraka i uglavnom je prikazan kutijastim dijagramom (engl. *box-and-whisker plot*). Kutijasti dijagramom se sastoji od pravokutnika koji prikazuje podatke od donjeg (0.25) do gornjeg (0.75) kvartila. Crta po pravokutniku označava median. Donje i gornje horizontalne linije se nazivaju *whisker*. *Wisker* se najčešće predstavlja kao najmanji i najveći podatak koji se nalazi unutar 1.5 puta interkvartilnog raspona gledajući od donjeg, odnosno gornjeg kvartila. [14] Svi podaci koji se nalaze izvan 1.5 puta interkvartilnog raspona se nazivaju outlieri, te svaka vrijednost se crta kao zasebna točka. [14] Ova tehnika je prikazana slikom 3.5.



Slika 3.5: Prikaz interkvartila pomoću kutijastog dijagrama, preuzeto iz [15]

Stavljanje granice (engl. *capping*) je dodatak IRQ tehnici gdje stavljanjem granica rješavamo outliere koji nam daju nemoguče vrijednost, tj. vrijednosti koje su nastale greškom priliko mje-

renja ili zapisa podataka. Granice se uglavnom stavljaju iznad 99% ili ispod 1% kako bi se riješili pogrešaka.

Baze podataka su uglavnom napravljene od više vrsta podataka. Za bolje rezultate sve podatke stavljamo pod istom vrstom podataka, a to su numerički podaci. Svaki kategorički podatak zamjenjujemo sa brojem koji će ga reprezentirati i to ponovimo na sve podatke dok ne dobijemo potpunu zamjenu.

Toplinska karta je grafički prikaz podataka gdje su pojedinačne vrijednosti matrice prikazane bojama. Uglavnom boje idu od tamno plave koja predstavlja najnižu vrijednost ili 0 do tamno crvene koja prikazuje najvišu vrijednost ili 1. U ovom slučaju vrijednost između 0 i 1 prikazuju ovisnost podataka atributa, te ako 2 atributa jako ovise jedan o drugome to znači da povećavanjem vrijednosti jednog, za isti postotak povećavamo vrijednost drugog. Takve atribute želimo imati što manje te se pokušavamo riješiti jednog od dva atributa za poboljšanje rezultata.

Iskošenost podataka (engl. *skewness*) je mjera iskrivljenje simetrične distribucije ili asimetrije u skupu podataka. Distribucija može imati desnu (pozitivnu), lijevu (negativnu) ili nultu asimetriju. Desno zakrivljena distribucija duža je na desnoj strani svog vrha, a lijevo zakrivljena distribucija duža je na lijevoj strani svog vrha. [16] Rezultat iskošenosti podataka želimo da bude što bliže 0, a rezultat između -0.5 i 0.5 je jako dobar za upotrebu.

3.3.2. Prikaz završnih baza podataka

Kao što je prijašnje rečeno svaka baza podataka je prošla kroz neke ili sve tehnike promjene. Svaka baza je dobila preimenovanje svojih atributa za lakše čitanje i radnju s njima. Interkvartil *IQR* je iskorišten na traženim vrijednostima baza podataka pomoću kojeg smo riješili outliere za dobivanje boljih rješenja, te je na određenim atributima iskorištena tehnika stavljanja granice pomoću koje smo riješili svih pogrešaka. Svi atributi koji nisu numeričke vrste podataka su izmijenjeni da budu, tako da algoritam strojnog učenja nema problema sa razumijevanjem podataka. Nakon svih izmjena podaci su prikazani na toplinskoj karti za zadnju evaluaciju, gdje po potrebi su još izbačeni određeni atributi.box-and-whisker plot svih atributa različitih baza su prikazani slikom 7.1, a završne toplinske karte baza podataka se nalaze na slici 7.2. I na kraju iskošenosti podataka za svaku bazu je prikazano slikama 7.3, 7.4, 7.5, 7.6, 7.7, 7.8, te tablicom 3.1.

Tablica 3.1: Vrijednosti iskošenosti baza podataka

	Dostupnost bicikli	Željezna industrija	Plinska turbina	Potrošnja energije (1 zona)	Potrošnja energije (2 zona)	Potrošnja energije (3 zona)
orginalne vrijednosti	0.9834	1.1545	0.0508	0.2288	0.3271	0.6557
logaritamski transformirane vrijednosti	-1.8949	0.5191	-0.1781	-0.1548	-0.1990	-0.0278
korijenom transformirane vrijednosti	0.1531	0.7504	-0.0642	0.0362	0.0651	0.3268
kubično transformirane vrijednosti	-0.3980	0.6495	-0.1027	-0.0275	-0.0225	0.2116

Kao što je vidljivo iz podataka uzimanje vrijednosti najbliže 0 dobivamo dovoljno simetrične distribucije podataka za traženu vrijednost.

Izgled završnih baza podataka možemo vidjeti na slici 3.6. Završne vrijednosti potrošene energije u džul-ima i vremena u sekundama za pretprocesiranje podataka iznose:

- Potrošena energija: 3833.0 j
- Vrijeme trajanja: 139.18 s

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8760 entries, 0 to 8759
Data columns (total 14 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   temp                 8760 non-null   float64
 1   humidity             8760 non-null   int64
 2   wind                 8760 non-null   float64
 3   visibility           8760 non-null   int64
 4   sunlight             8760 non-null   float64
 5   rain                 8760 non-null   float64
 6   snow                 8760 non-null   float64
 7   holiday              8760 non-null   int64
 8   functioning_day      8760 non-null   int64
 9   weekdays_or_weekend  8760 non-null   int64
 10  timeshift            8760 non-null   int64
 11  Autumn               8760 non-null   uint8
 12  Spring               8760 non-null   uint8
 13  bike_count           8760 non-null   float64
dtypes: float64(6), int64(6), uint8(2)
memory usage: 838.5 KB
```

(a) Seoul Bike Sharing Demand baza podataka

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 35040 entries, 0 to 35039
Data columns (total 11 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   Lagging_Power         35040 non-null  float64
 1   Leading_Power         35040 non-null  float64
 2   CO2                   35040 non-null  float64
 3   Lagging_Power_Factor  35040 non-null  float64
 4   Leading_Power_Factor  35040 non-null  float64
 5   NSM                   35040 non-null  int64
 6   WeekStatus            35040 non-null  int64
 7   Light_Load            35040 non-null  uint8
 8   Maximum_Load          35040 non-null  uint8
 9   Medium_Load           35040 non-null  uint8
 10  Usage_kWh             35040 non-null  float64
dtypes: float64(6), int64(2), uint8(3)
memory usage: 2.2 MB
```

(b) Steel Industry Energy Consumption baza podataka

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7411 entries, 0 to 7410
Data columns (total 11 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   AT      7411 non-null   float64
 1   AP      7411 non-null   float64
 2   AH      7411 non-null   float64
 3   AFDP    7411 non-null   float64
 4   GTEP    7411 non-null   float64
 5   TIT     7411 non-null   float64
 6   TAT     7411 non-null   float64
 7   CDP     7411 non-null   float64
 8   CO      7411 non-null   float64
 9   NOX     7411 non-null   float64
 10  TEY     7411 non-null   float64
dtypes: float64(11)
memory usage: 637.0 KB
```

(c) Gas Turbine CO and NOx Emission baza podataka

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 52416 entries, 0 to 52415
Data columns (total 8 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   Temperature            52416 non-null  float64
 1   Humidity               52416 non-null  float64
 2   Wind Speed             52416 non-null  float64
 3   general diffuse flows  52416 non-null  float64
 4   diffuse flows          52416 non-null  float64
 5   Zone_1                 52416 non-null  float64
 6   Zone_2                 52416 non-null  float64
 7   Zone_3                 52416 non-null  float64
dtypes: float64(8)
memory usage: 3.2 MB
```

(d) Power consumption of Tetouan city baza podataka

Slika 3.6: Prikaz završnih atributa baza podataka

3.4. Skaliranje podataka

Podaci u bazi podataka mogu biti vrlo različiti, npr. imamo 2 atributa, prvi atribut ima vrijednosti od 0 - 20 dok drugi ima vrijednosti od 1000 - 200000. Ovakve velike razlike u vrijednostima može uzrokovati stvaranje lošijih modela strojnog učenja, te za rješavanje ovog problema koristimo funkcije skaliranja (engl. *scaler*). Funkcije skaliranja se koriste nad ulaznim podacima i postoje različite vrste. Najčešće korištene Funkcije skaliranja su: standardizacija (engl. *Standard Scaler*), skaliranje s najvećom vrijednošću (engl. *Min Max Scaler*) i skaliranje s najvećom apsolutnom vri-

jednošću (engl. *Max Abs Scaler*).

3.4.1. Standardizacija

Standardizacija je proces skaliranja podataka tako da je srednja vrijednost nula, a standardna devijacija je jedan. Ovaj način se uglavnom koristi za probleme klasifikacije. Standardizacije je prikazana izrazom 3.1:

$$x' = \frac{x - \mu}{\sigma} \tag{3.1}$$

Gdje je x vrijednost podatka, μ je srednja vrijednost, a σ je standardna devijacija.

3.4.2. Min Max Scaler

Min Max Scaler ili poznata kao normalizacija je proces skaliranja podataka u granice [0, 1]. Ova metoda se uglavnom koristi na problemima regresije, te je ovim radom korišten Min Max Scaler za skaliranje svih ulaznih podataka. Metoda također ima jedan nedostatak jer je osjetljiva na outliere. Normalizacije je prikazana izrazom 3.2:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{3.2}$$

Gdje je x vrijednost podatka, x_{min} je najmanja vrijednost, a x_{max} je najveća vrijednost.

3.4.3. Max Abs Scaler

Max Abs Scaler je proces gdje se svaka značajka skalira pomoću apsolutne maksimalne vrijednosti u granice [-1, 1]. Metoda je jako slična normalizaciji, te ima isti problem osjetljivosti na outliere i korištenje na problemima regresije. Max Abs Scaler je prikazan izrazom 3.3:

$$x' = \frac{x}{abs(x_{max})} \tag{3.3}$$

Gdje je x vrijednost podatka, a $abs(x_{max})$ je apsolutna maksimalna vrijednost.

4. PRIKAZ MODELA

Modeli strojnog učenja su trenirani na principu unakrsne validacije (engl. *k-fold cross-validation*). Unakrsna validacija je statistička metoda koja se koristi za procjenu modela strojnog učenja. Metoda koristi parametar nazvan 'K' koji predstavlja broj grupa na koje se baza podataka treba podijeliti, npr. ako uzmemo da 'K' iznosi 10 onda se baza podataka podijeli na 10 jednakih slučajno odabranih grupa. Model koji predstavlja završne vrijednosti je dobiven uzimanjem srednje vrijednosti 10 treniranih modela. Ova metoda jamči da rezultat našeg modela ne ovisi o načinu na koji smo odabrani podatke za treniranje i testiranje.

Algoritmi KNN i neuralne mreže imaju više argumenata koji utječu na završne vrijednosti. Za prikaz razlika oba algoritma su trenirana više puta sa istim podacima samo drugim parametrima. KNN algoritmu su uzete vrijednosti 15 različitih testiranja sa drugačijim brojem susjeda koji ide od 1 do 15 susjeda. Ovo je napravljeno za svaku bazu podataka i prikazano je grafovima. Naspram KNN algoritmu, neuralne mreže ovise o broju skrivenih slojeva i broja neurona, te za svaku bazu podataka napravljena su treniranja sa različitim brojem navedenih parametara. Navedeni parametri su uvijek isti i glase: 1 skriveni sloj sa 5 neurona (5), 1 skriveni sloj sa 10 neurona (10), 1 skriveni sloj sa 50 neurona (50), 2 skrivena sloja sa 5 i 5 neurona (5, 5), 2 skrivena sloja sa 10 i 10 neurona (10, 10), 2 skrivena sloja sa 50 i 50 neurona (50, 50), 3 skrivena sloja sa 5, 5 i 5 neurona (5, 5, 5) i 3 skrivena sloja sa 10, 10 i 10 neurona (10, 10, 10).

U nastavku su prikazani rezultati funkcija pogreške i točnosti, te RAPL informacije za spomenute algoritme pod poglavljem 2.2

4.1. Rezultati nad podacima Seoul Bike Sharing Demand

Rezultati prikazani tablicom 4.1 su za potražnju iznajmljivanja bicikli u glavnom Korejskom gradu Seolu. Funkcijama pogreške i rezultatom R^2 možemo vidjeti da model KNN-a (sa pet susjeda) najbolje predviđa rezultate. Za model neuralne mreže uzeti parametri su 2 skrivena sloja sa 6 i 6 neurona (6, 6). Također možemo usporediti korištenu energiju i vrijeme treniranja modela iz kojih dobivamo da najjeftiniji model za izradu je linearni dok najskuplji SVR. Najefikasniji model je prijašnje imenovan KNN koji dobiva najbolja predviđanja uz minimalnu potrošnju naspram drugim algoritmima.

Tablica 4.1: Rezultati modela Seoul Bike Sharing Demand baze podataka

ML model	R^2 [%]	MAE	RMSE	Adjusted R^2	E [j]	t [s]
Linear	67.03	259.463	379.244	0.575	0.751	0.011
Lasso	20.37	408.764	539.307	0.142	2.702	0.037
Ridge	67.03	259.627	379.439	0.575	1.243	0.017
Elastic Net	9.32	439.455	575.601	0.023	1.520	0.021
KNN	76.44	208.726	325.795	0.686	12.954	0.282
SVR	72.84	233.268	354.690	0.628	243.865	5.295
Neuralna mreža	68.64	247.934	364.038	0.605	128.930	2.468

Da bi vidjeli razliku koja dolazi uzimanjem drugih podataka za treniranje i testiranje, slikom 7.9 možemo uočiti box-and-whisker plot 10 k-fold unakrsne validacija gdje se nalaze vrijednosti za MAE, RMSE, adjusted R2, vrijeme trajanja i potrošena energija (PKG), svake od 10 grupe validacije. Ovisnost KNN modela o susjedima za tražene vrijednosti R^2, trajanje treniranja i potrošnje energije, te MAE i RMSE možemo vidjeti na slici 7.10. Možemo uočiti da KNN algoritmu dodavanjem susjeda ne poboljšavamo vrijednost u beskonačnost, već da postoji granica nakon koje model ne postaje bolji već lošiji. U oba slučaja to možemo vidjeti između 4 i 5 susjeda. Vrijednosti različitih parametara modela neuralnih mreža možemo vidjeti na slici 7.11, gdje je vidljivo da povećavanjem broja neurona i skrivenih slojeva ne poboljšava rezultate dovoljno da bi bilo vrjednije koristiti više energije i vremena za dobivanje gotovo istih rezultata.

4.2. Rezultati nad podacima Steel Industry Energy Consumption

Rezultati prikazani tablicom 4.2 su za potrošnju energije u tvornici željezne industrije. Tabličnim vrijednostima možemo vidjeti KNN model (sa pet susjeda) je ponovno najbolji u predviđanju, dok Linearni model ima najmanje vrijeme treniranja i potrošnje energije, a SVR ima najveće trijeme treniranja i potrošnju energije. Najefikasnije rješenje je i dalje KNN model. Velika razlika je u R^2 vrijednostima za Lasso i Elastic Net model kojima je vrijednost negativna, što znaci da model predviđa vrijednosti lošije od srednje vrijednosti ciljnih varijabli. Ovo nam dokazuje da neki algoritmi nisu dobri za predviđanje svih podataka, te ti podaci unutar tablice 4.2 su označeni sa 0. Vrijednosti modela neuralne mreže su dobiveni sa parametrima od 1 skrivenog sloja sa 3 neurona (3). Više informacija o zasebnim grupama modela za vrijednosti MAE, RMSE, adjusted R2, vrijeme trajanja i potrošena energija (PKG), možemo vidjeti na slici 7.12.

Tablica 4.2: Rezultati modela Steel Industry Energy Consumption baze podataka

ML model	R^2 [%]	MAE	RMSE	Adjusted R^2	E [j]	t [s]
Linear	88.68	1.354	2.601	0.870	0.846	0.010
Lasso	0	6.397	7.408	0	3.965	0.055
Ridge	88.68	1.354	2.602	0.869	2.011	0.028
Elastic Net	0	6.397	7.408	0	2.268	0.032
KNN	96.98	0.334	1.136	0.959	30.53	0.716
SVR	94.28	0.538	1.922	0.926	569.345	12.85
Neuralna mreža	94.87	0.721	1.827	0.935	10.935	3.138

[**??** slikom možemo vidjeti ovisnost susjeda KNN algoritma. R^2 vrijednost cijelim putem ostaje stabilna, dok PKG se pogoršava dodavanjem susjeda. MAE i RMSE Vrijednosti se ponašaju malo drugačije, već sa 3 susjeda dostignuta je najbolja vrijednost, te povećavanjem susjeda vrijednost se proporcionalno povećava. Ovakvo ponašanje modela može biti uzrokovano prevelikom udaljenošću vrijednosti, tj. vrijednosti su grupirane i udaljenost grupa je mnogo veća nego udaljenost unutar grupe. Ovo možemo vidjeti na slici 7.13. Sa modelima neuralnih mreža razlike u parametrima možemo vidjeti na slici 7.14. Možemo uočiti da povečavanjem skrivenih slojeva i neurona dobivamo stabilnije vrijednosti uz minimalno povećanje vremena treniranja i potrošnje energije.

4.3. Rezultati nad podacima Gas Turbine CO and NOX Emission

Rezultati prikazani tablicom 4.3 su za potrošnju energije i ispuštenih plinova plinske turbine. Kao što možemo vidjeti opet imamo problem sa modelima Lasso i Elastic net, ali zato po prvi put možemo vidjeti da KNN model nije jednoglasno najbolji. Linearni model jer još uvijek najbolji u vrijednostima trajanja treniranja modela i potrošnji energije, dok neuralna mreža sa 1 skrivenim slojem od 3 neurona (3) je najgora po vrijednostima trajanja treniranja modela i potrošnji energije. Kako vrijednosti KNN, SVR i Linearnog modela su slični, ja bi proglasio linearni algoritam kao najefikasniji za treniranje modela. Više informacija o zasebnim grupama modela za vrijednosti MAE, RMSE, adjusted R2, vrijeme trajanja i potrošena energija (PKG), možemo vidjeti na slici 7.15.

Tablica 4.3: Rezultati modela Gas Turbine CO and NOX Emission baze podataka

ML model	R^2 [%]	MAE	RMSE	Adjusted R^2	E [j]	t [s]
Linear	96.82	1.136	2.636	0.971	0.523	0.008
Lasso	0	12.016	15.931	0	1.550	0.022
Ridge	88.68	1.354	2.602	0.869	2.011	0.028
Elastic Net	0	12.016	15.931	0	0.889	0.012
KNN	97.77	0.990	2.243	0.979	11.417	0.236
SVR	97.45	1.058	2.364	0.976	8.012	0.179
Neuralna mreža	96.28	1.412	2.885	0.965	104.158	2.440

Ovisnost susjeda KNN algoritma je prikazano slikom 7.16. Vrijednost R^2 i PKG su stabilni cijelim vremenom treniranja, dok za MAE i RMSE vrijednosti imamo uobičajeno poboljšanje vrijednosti do određene granice, koja je u ovom slučaju 6 susjeda. Modeli neuralnih vrijednosti su uglavnom stabilni cijelim putem promjena vrijednosti skrivenih slojeva i neurona. Jedina razlika se može uočiti u promjeni potrošnje energije i vremena treniranja sa povećanjem neurona što je i za očekivati. Modeli neuralnih vrijednosti su prikazani slikom 7.17.

4.4. Rezultati nad podacima Power consumption of Tetouan city

Ova baza podataka ima 3 tražena atributa, ako koristimo bilo koji korišteni algoritam strojnog učenja osim neuralne mreže onda imamo problem. Neuralna mreža je jedini korišteni algoritam koji može raditi sa više traženih vrijednosti. Ovaj problem je riješen tako da podijelimo rješenje na 3 dijela, svaki model je istreniran sa jednom traženom vrijednosti i samo je neuralna mreža ekstra istrenirana sa sve 3 tražene vrijednosti.

4.4.1. Rezultati nad podacima Power consumption of Tetouan city 1, 2 i 3 zone

Rezultati prikazani tablicama 4.4, 4.5 i 4.6 su za potrošnju energije 1, 2 i 3 zone grada Tetouan. Usporedbom rezultata vidimo da svaki put KNN algoritam sa četiri susjeda je najbolji i najefikasniji, dok SVR troši najviše energije i vremena za treniranja. Iskorišteni parametri za treniranje algoritma neuralne mreže su 1 skriveni sloj sa 4 neurona (4). Više informacija o zasebnim grupama modela za vrijednosti MAE, RMSE, adjusted R2, vrijeme trajanja i potrošena energija (PKG), možemo vidjeti na slikama 7.18, 7.21 i 7.24.

Tablica 4.4: Rezultati modela Power consumption of Tetouan city 1 zone baze podataka

ML model	R^2 [%]	MAE	RMSE	Adjusted R^2	E [j]	t [s]
Linear	21.35	5197.304	6568.593	0.201	1.006	0.016
Lasso	9.91	5618.872	6786.371	0.093	3.673	0.049
Ridge	21.35	5197.278	6358.606	0.201	1.916	0.025
Elastic Net	2.86	5830.618	7042.412	0.079	5.837	0.079
KNN	44.00	3535.931	5341.776	0.437	21.742	0.501
SVR	23.30	4696.749	5308.566	0.216	637.181	151.395
Neuralna mreža	21.33	5197.091	6368.812	0.201	344.851	8.233

Tablica 4.5: Rezultati modela Power consumption of Tetouan city 2 zone baze podataka

ML model	R^2 [%]	MAE	RMSE	Adjusted R^2	E [j]	t [s]
Linear	17.41	3813.350	4748.118	0.164	1.415	0.023
Lasso	2.70	4180.976	5138.094	0.021	3.694	0.051
Ridge	17.41	3813.335	4748.127	0.164	2.119	0.029
Elastic Net	1.93	4196.079	5158.731	0.013	2.779	0.038
KNN	41.55	2689.338	3977.258	0.413	22.619	0.498
SVR	22.87	3530.843	4599.594	0.215	6632.299	153.343
Neuralna mreža	17.42	3813.562	4747.827	0.164	336.21	7.690

Tablica 4.6: Rezultati modela Power consumption of Tetouan city 3 zone baze podataka

ML model	R^2 [%]	MAE	RMSE	Adjusted R^2	E [j]	t [s]
Linear	19.53	4130.701	5176.633	0.199	1.349	0.021
Lasso	10.06	4317.372	5511.747	0.092	5.542	0.077
Ridge	19.53	4130.618	5176.710	0.199	2.665	0.036
Elastic Net	2.55	4486.597	5735.287	0.017	3.322	0.045
KNN	50.65	2732.087	4046.094	0.511	22.441	0.596
SVR	30.50	3697.728	4815.361	0.307	6972.332	152.960
Neuralna mreža	19.51	4131.583	5178.160	0.199	368.779	8.457

Slikama 7.19, 7.22 i 7.25 su prikazani rezultati ovisnosti susjeda KNN algoritma. Nad podacima triju modela možemo uočiti isti trend, a to je da R^2 počne stagnirati dok se vrijeme trajanja treniranja i PKG penje. MAE vrijednost dosegne minimum unutar prvih par susjeda i počne se linearno penjati, time da RMSE stagnira. Slikama 7.20, 7.23 i 7.26 vidimo rezultate različitih parametara neuralne mreže, te možemo uočiti da su rezultati dovoljno stabilni dodavanjem skrivenih slojeva i neurona. Vidimo po jedan outlier što dovodi do lošijih rezultata, ali to može biti zbog lošeg odabira podataka za tu grupu testiranja

4.4.2. Rezultati neuralne mreže nad podacima Power consumption of Tetouan city

Tablicom 4.7 su prikazani rezultati neuralne mreže sa 2 skrivena sloja, 3 i 3 neurona (3, 3). Više informacija možemo vidjeti na slici 7.27, gdje možemo vidjeti da dodavanjem skrivenih slojeva vrijednosti fluktuiraju.

Tablica 4.7: Rezultati modela Power consumption of Tetouan city sve zone baze podataka

ML model	R^2 [%]	MAE	RMSE	Adjusted R^2	E [j]	t [s]
Neuralna mreža	20.41	4329.567	5436.176	0.199	1003.436	23.880

5. ZAKLJUČAK

Strojno učenje je jako kompleksna i zahtjevna grana tehnologija. Koristi se u jako puno sustava današnjice, te nastavit će se razvijati i inovirati. Kao što je prikazano modeli strojnog učenja ovise o puno parametara, za dobivanje dobrog i efikasnog modela moramo brinuti o podacima, algoritmima i sustavu na kojem radimo. Računalo na kojem se radi treniranje i testiranje modela ima jako veliku ulogu, jer ovisno o performansama računala treniranje može izvršiti jako brzo ili sporo.

Ovim radom prikazana je učinkovitost različitih modela nad podacima 4 baze podataka, te njihove nedostatke u određenim trenucima. Istrenirani modeli nisu savršeni i imaju mjesta za unapređenje, dodavanjem novih parametara ili boljim pretprocesiranjem podataka. Ovo bi naravno bilo poželjno, ali niti jedan sustav temeljen naspram strojnom učenju nije savršen i to isto savršenstvo nije moguće ostvariti.

6. BIBLIOGRAFIJA

[1] Bolf, N.: "Osvježimo znanje: Strojno učenje", Kemija u industriji : Časopis kemičara i kemijskih inženjera Hrvatske, Vol. 70 No. 9-10, 2021, s interneta, https://hrcak.srce.hr/file/382926, 7. Svibnja 2023.

[2] Mahesh, B.: "Machine Learning Algorithms - A Review", s interneta, https://www.researchgate.net/profile/Batta-Mahesh/publication/344717762_Machine_Learning_Algorithms_-A_Review/links/5f8b2365299bf1b53e2d243a/Machine-Learning-Algorithms-A-Review.pdf?eid=5082902844932096, 29. Svibnja 2023.

[3] Scikit-learn: "About Scikit-learn", s interneta, https://scikit-learn.org/stable/about.html, 25. Listopad 2022.

[4] Ahmed Asim, S.: "L0 Regularization Based Neural Network Design and Compression", s interneta, https://arxiv.org/ftp/arxiv/papers/1905/1905.13652.pdf, 2. Veljače 2023.

[5] Ozgur, D. K.; Kamada, M.; Akutsu, T.; Knapp, E. W.: "Prediction using step-wise L1, L2 regularization and feature selection for small data sets with large number of features", s interneta, https://link.springer.com/article/10.1186/1471-2105-12-412, 2. Veljače 2023.

[6] Šnajder, J.: "Strojno učenje: 11. Neparametarske metode", s interneta, https://www.fer.unizg.hr/_download/repository/SU-2017-11-NeparametarskeMetode.pdf, 2. Veljače 2023.

[7] Sethi, A.: "Support Vector Regression Tutorial for Machine Learning", s interneta, https://www.analyticsvidhya.com/blog/2020/03/support-vector-regression-tutorial-for-machine-learning/, 3. Veljače 2023.

[8] Pedamkar, P.: "Introduction to Support Vector Regression", s interneta, https://www.educba.com/support-vector-regression/, 2. Veljače 2023.

[9] Židov, I.: "Uvod u neuronske mreže", s interneta, https://repozitorij.mathos.hr/islandora/object/mathos%3A256/datastream/PDF/view, 19. Travnja 2023.

[10] Brownlee, J.: "How to Choose Loss Functions When Training Deep Learning Neural Networks", s interneta, https://machinelearningmastery.com/howto-choose-loss-functions-when-training-deep-learning-neural-networks/, 30. Siječnja 2023.

[11] Ljubobratović, D.: "Interpretabilno strojno učenje", s interneta, https://www.inf.uniri.hr/images/studiji/poslijediplomski/kvalifikacijski/Dejan_Ljubobratovic_kvalifikacijski.pdf, 30. Siječnja 2023.

[12] Running Average Power Limit: "Reading RAPL energy measurements from Linux", s interneta, https://web.eece.maine.edu/ vweaver/projects/rapl/, 1. Studeni 2022.

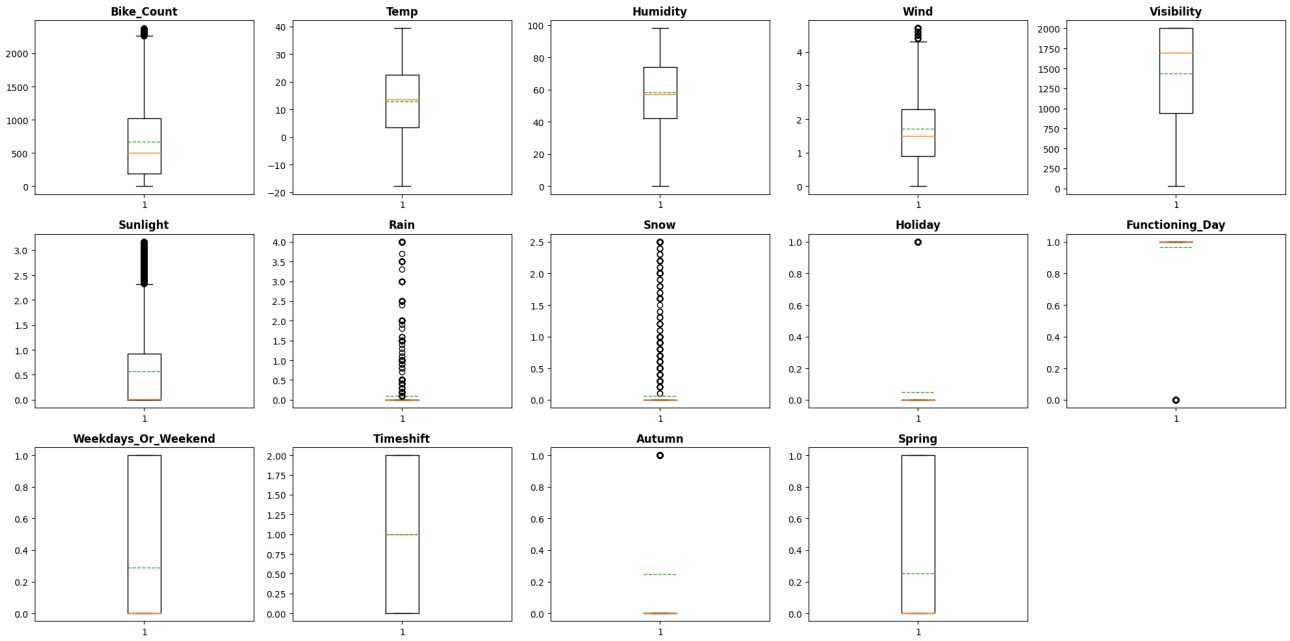
[13] UCI machine learning repository, s interneta, https://archive.ics.uci.edu/ml/index.php, 30.Siječnja 2023.

[14] Nuzzo, R. L.: "The Box Plots Alternative for Visualizing Quantitative Data ", s interneta, https://onlinelibrary.wiley.com/doi/pdf/10.1016/j.pmrj.2016.02.001, 30. Siječnja 2023.

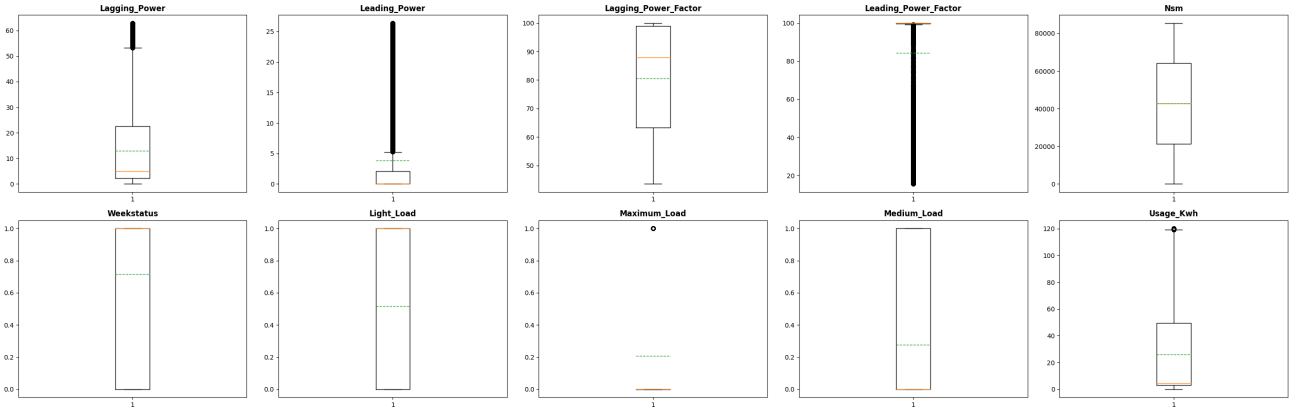
[15] PRIANCAASHARMA: "Box Plot", s interneta, https://datascienceunwind.wordpress.com/2019/10/03/box-plot/, 30. Siječnja 2023.

[16] Liu, W.; Chawla, S.: "A Quadratic Mean based Supervised Learning Model for Managing Data Skewness", s interneta, https://epubs.siam.org/doi/pdf/10.1137/1.9781611972818.17, 7. Veljače 2023.

7. DODATAK



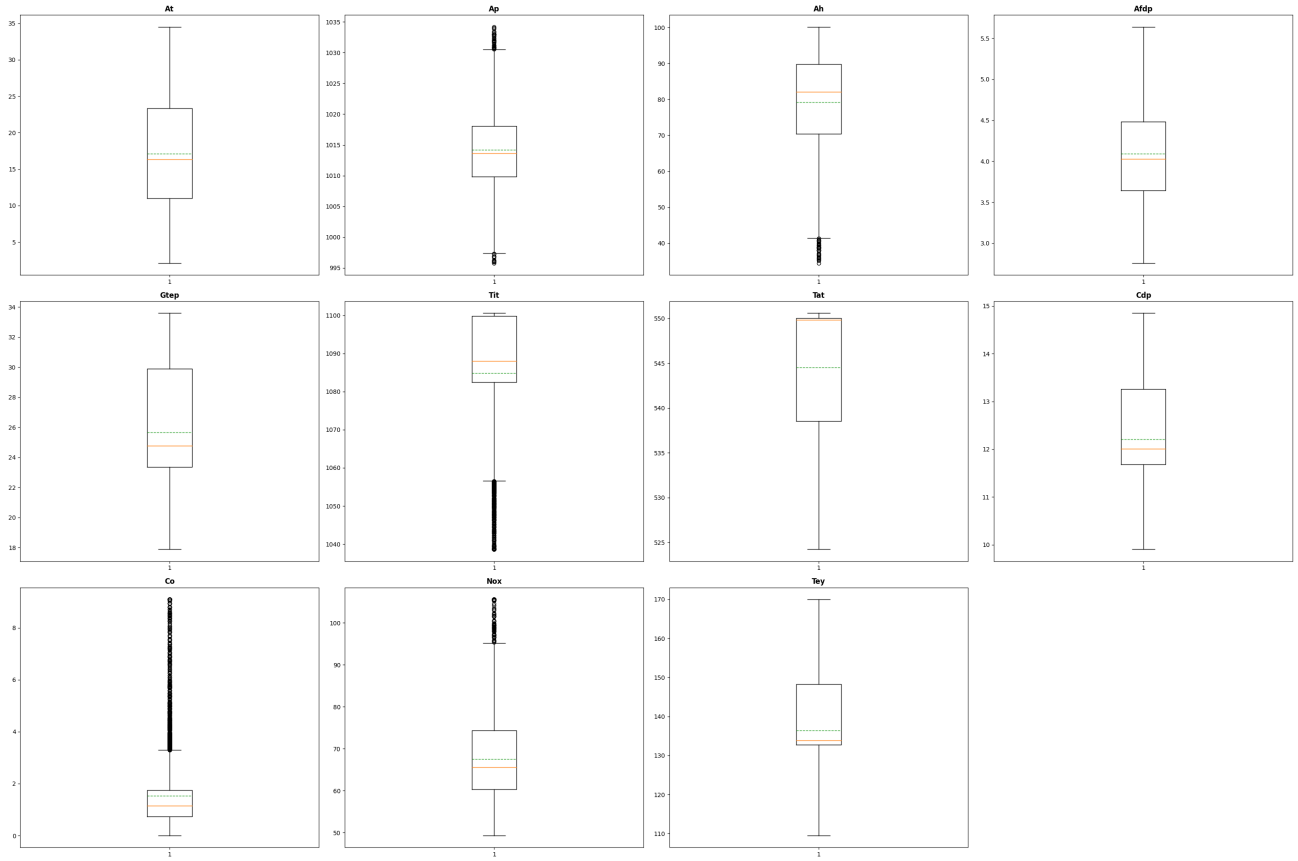
(a) Seoul Bike Sharing Demand baza podataka



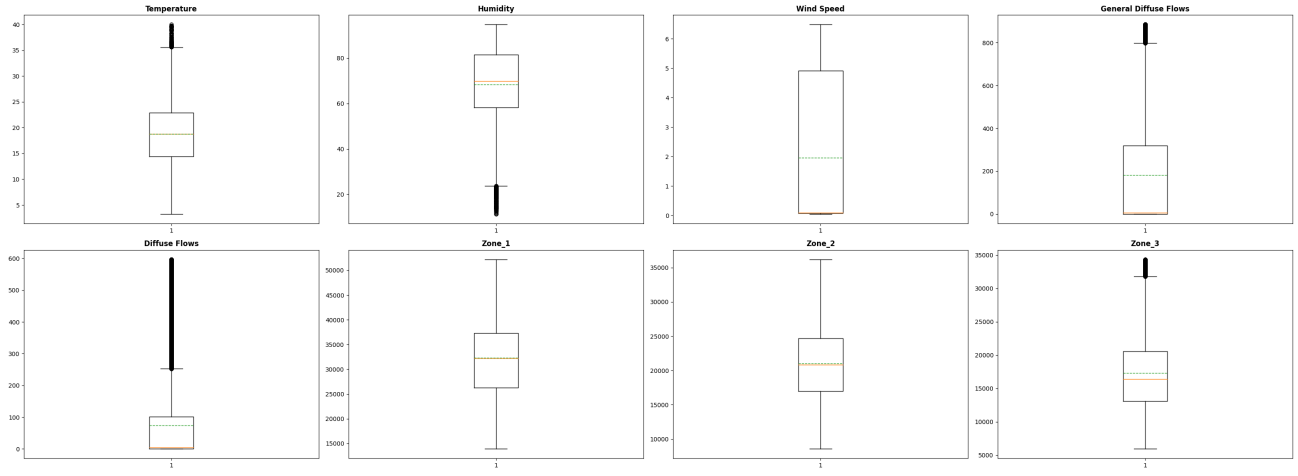
(b) Steel Industry Energy Consumption baza podataka

Slika 7.1: Box i whisker dijagram završnih vrijednosti baza podataka

(Nastavlja se na sljedećoj stranici)

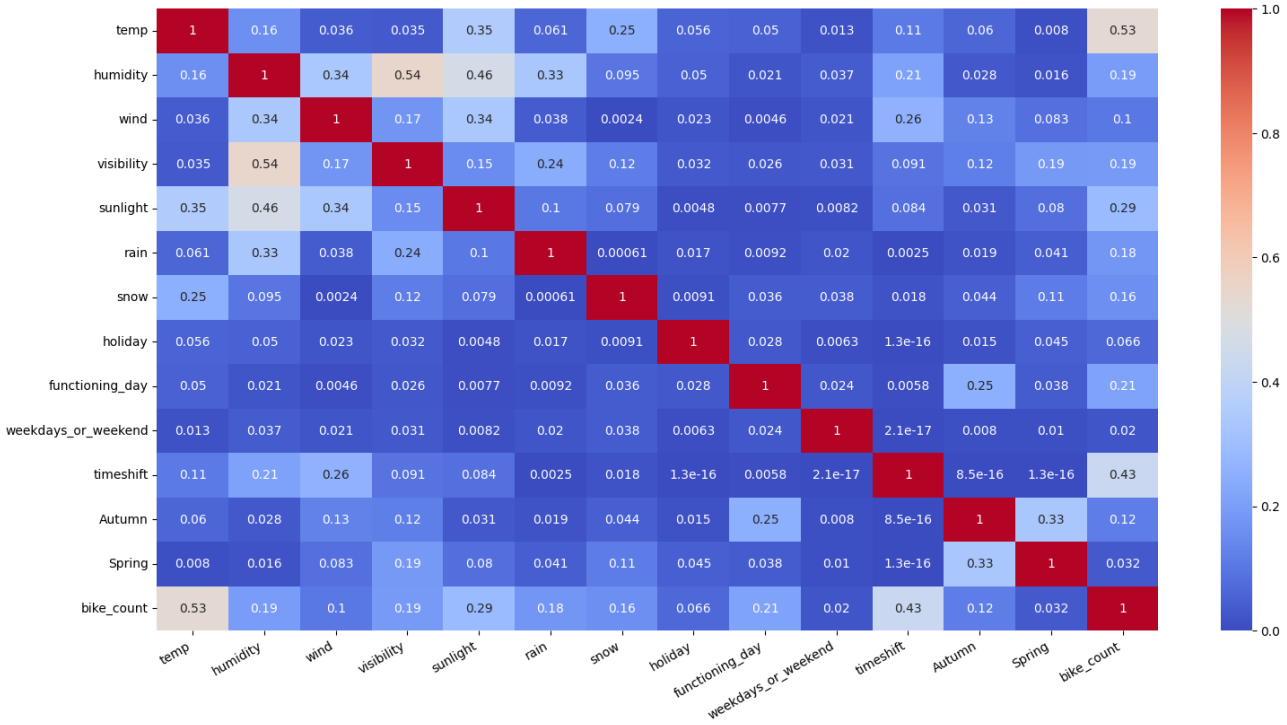


(c) Gas Turbine CO and NOx Emission baza podataka

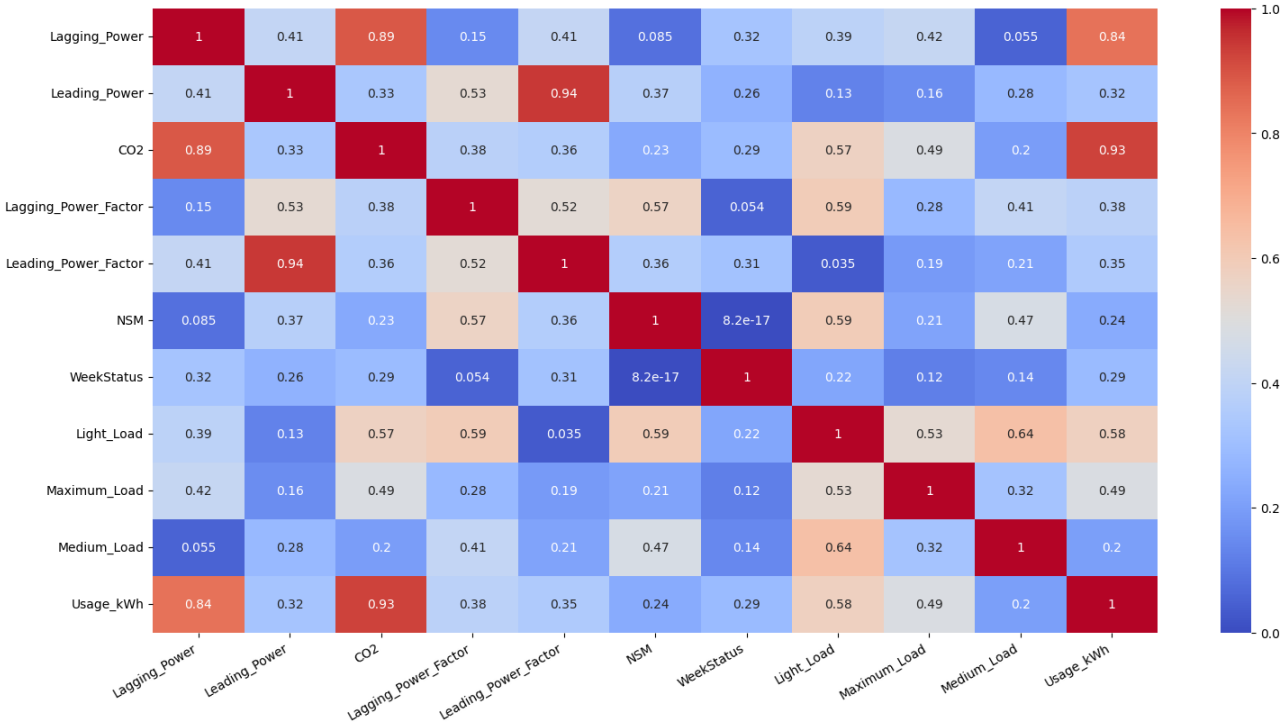


(d) Power consumption of Tetouan city baza podataka

Slika 7.1: Box i whisker dijagram završnih vrijednosti baza podataka
(Nastavak s prijašnje stranice)



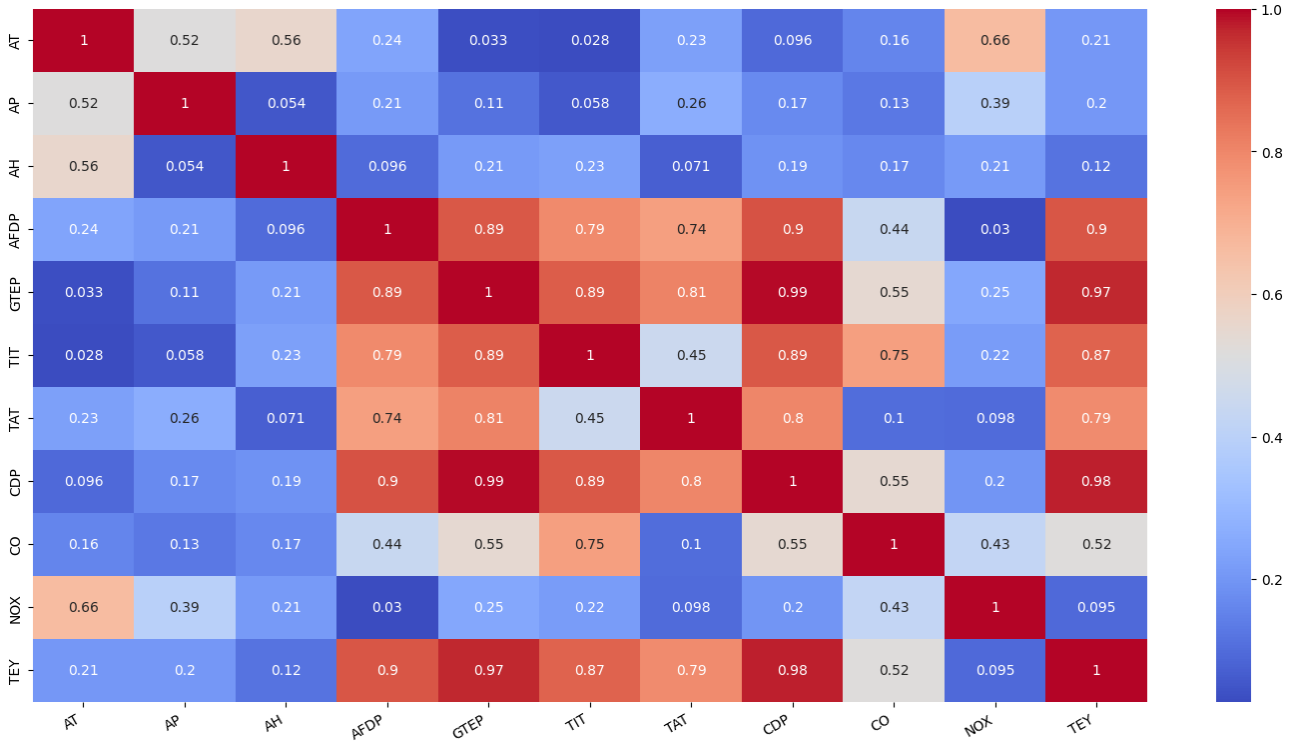
(a) Seoul Bike Sharing Demand baza podataka



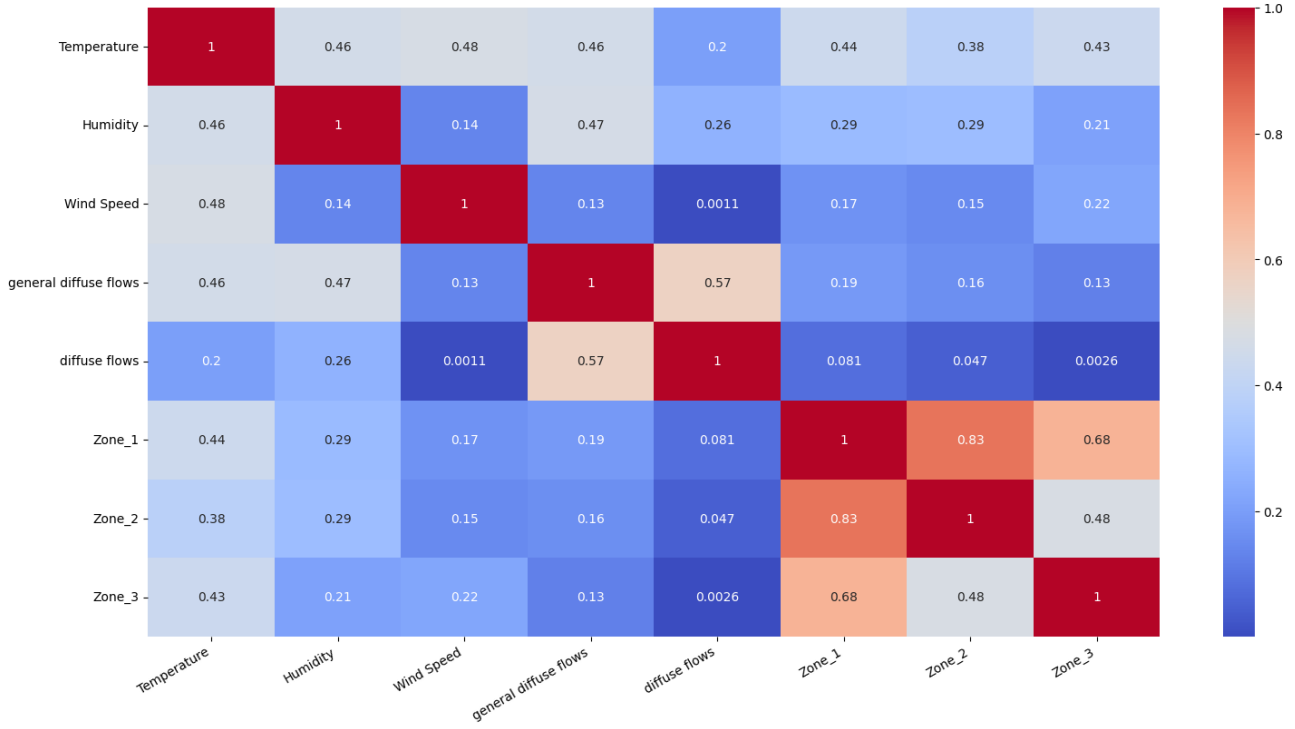
(b) Steel Industry Energy Consumption baza podataka

Slika 7.2: Toplinske mape završnih vrijednosti baza podataka

(Nastavlja se na sljedećoj stranici)

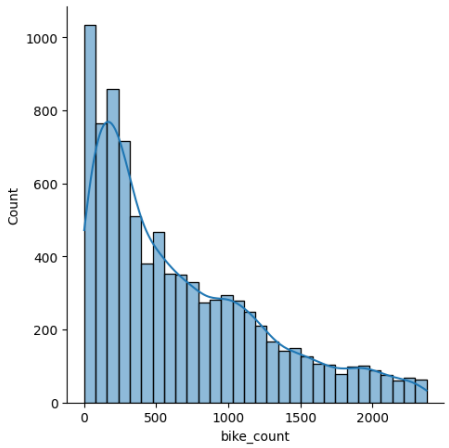


(c) Gas Turbine CO and NOx Emission baza podataka

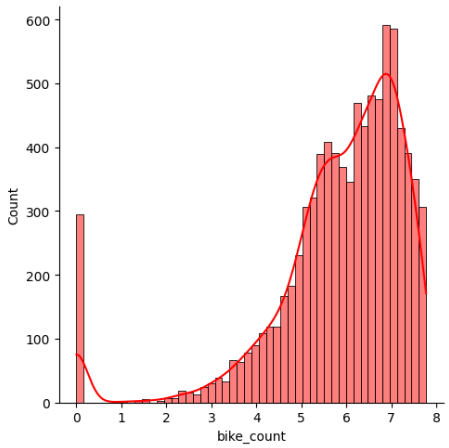


(d) Power consumption of Tetouan city baza podataka

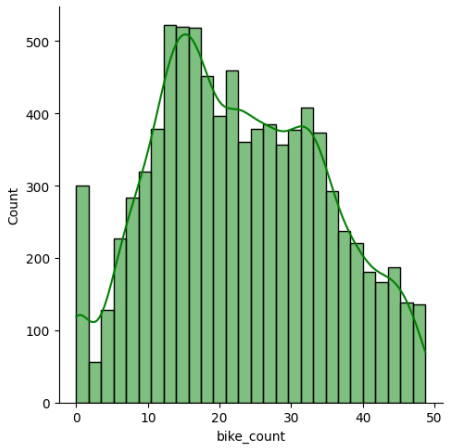
Slika 7.2: Toplinske mape završnih vrijednosti baza podataka
(Nastavak s prijašnje stranice)



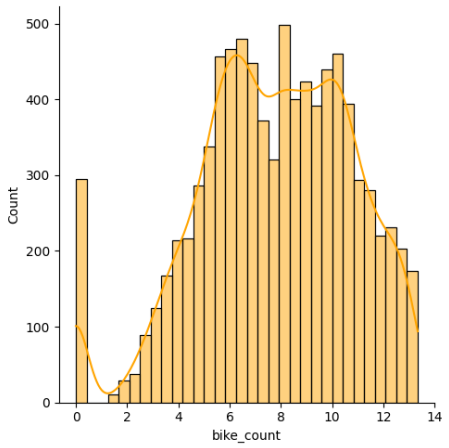
(a) Normalna iskrivljenost



(b) Logaritamski transformirana iskrivljenost

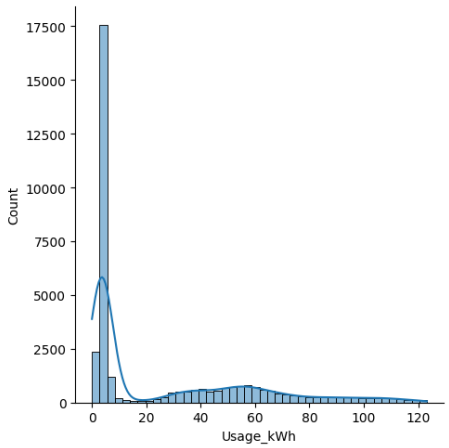


(c) Korijenom transformirana iskrivljenost

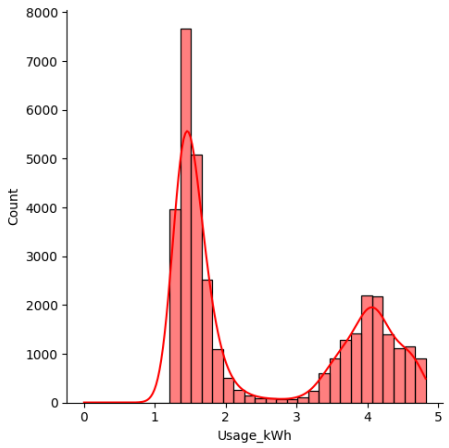


(d) Kubno transormirana iskrivljenost

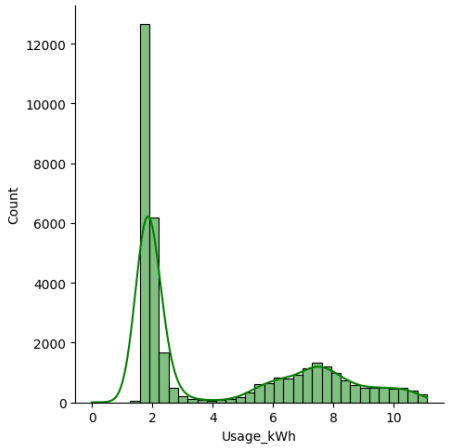
Slika 7.3: Iskrivljenosti Seoul Bike Sharing Demand baze podataka



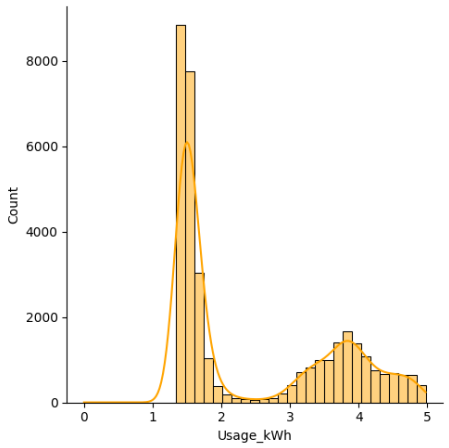
(a) Normalna iskrivljenost



(b) Logaritamski transformirana iskrivljenost

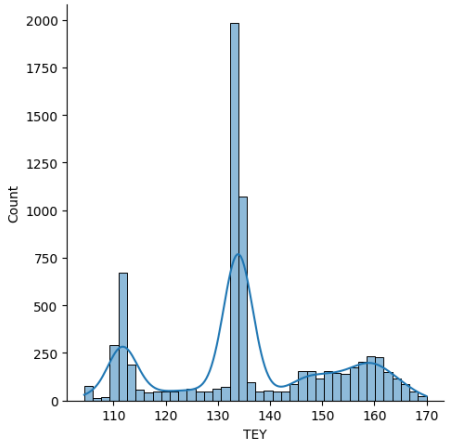


(c) Korijenom transformirana iskrivljenost

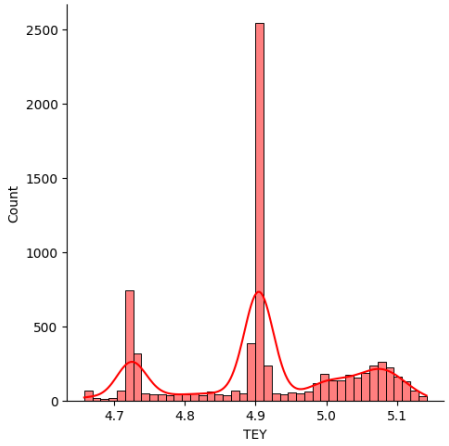


(d) Kubno transormirana iskrivljenost

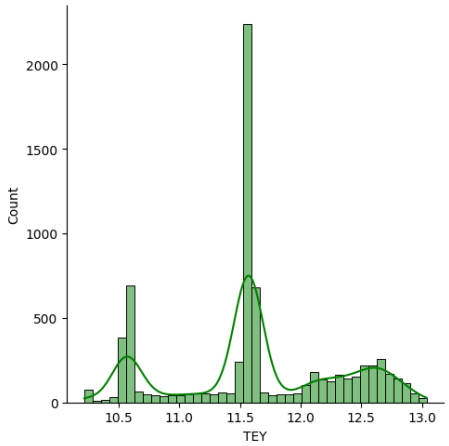
Slika 7.4: Iskrivljenosti Steel Industry Energy Consumption baze podataka



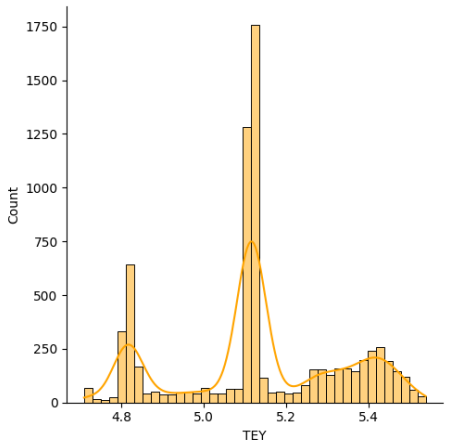
(a) Normalna iskrivljenost



(b) Logaritamski transformirana iskrivljenost

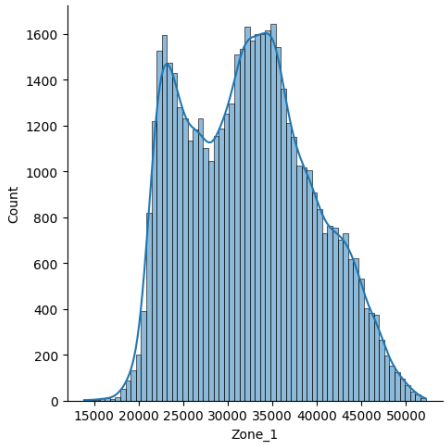


(c) Korijenom transformirana iskrivljenost

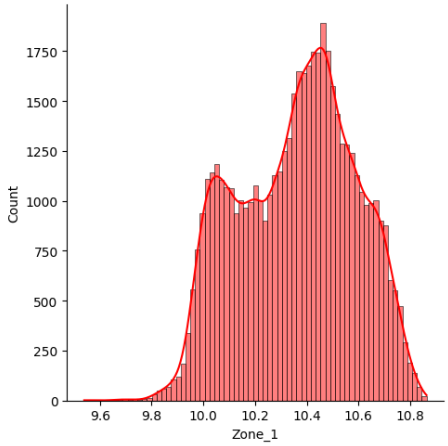


(d) Kubno transormirana iskrivljenost

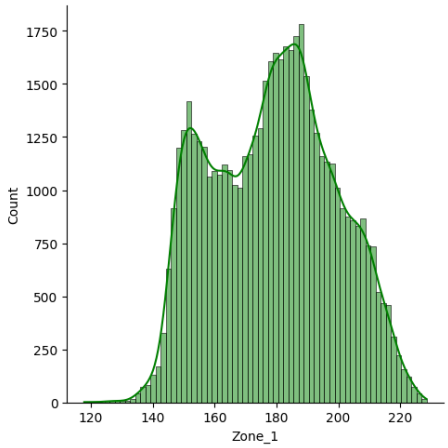
Slika 7.5: Iskrivljenosti Gas Turbine CO and NOX Emission baze podataka



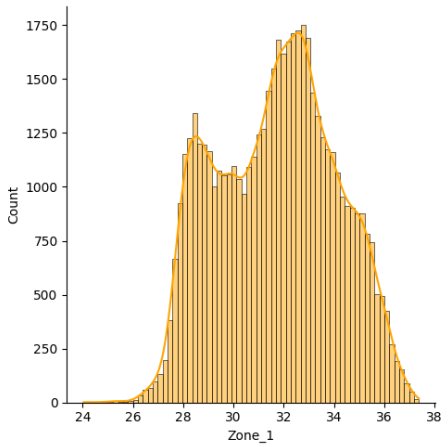
(a) Normalna iskrivljenost



(b) Logaritamski transformirana iskrivljenost

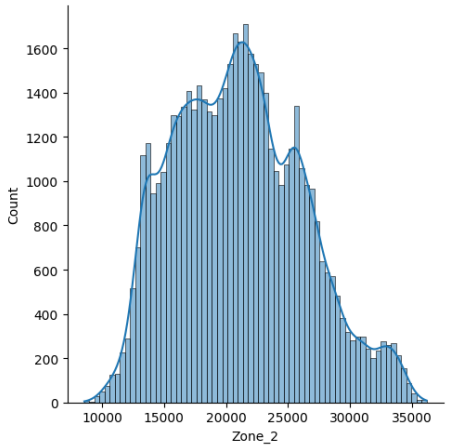


(c) Korijenom transformirana iskrivljenost

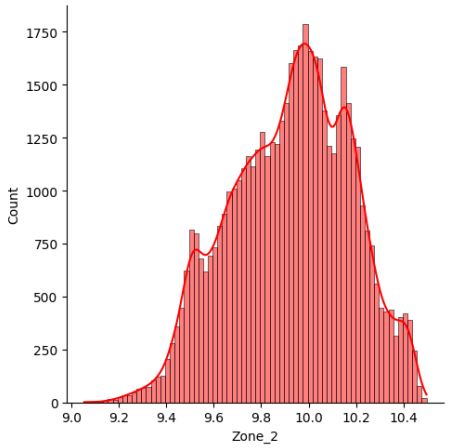


(d) Kubno transormirana iskrivljenost

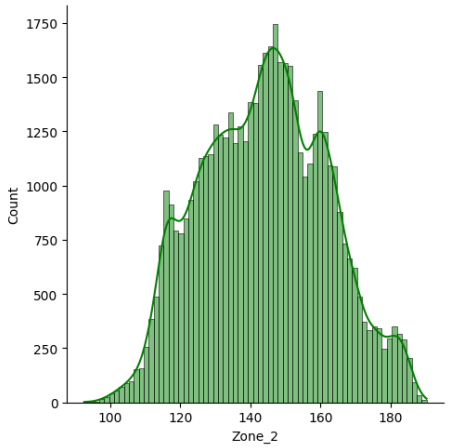
Slika 7.6: Iskrivljenosti Power consumption of Tetouan city baze podataka 1 zone



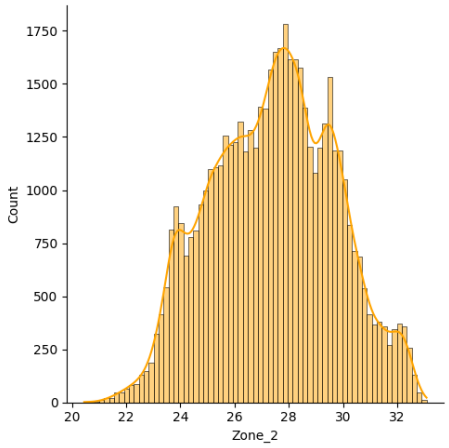
(a) Normalna iskrivljenost



(b) Logaritamski transformirana iskrivljenost

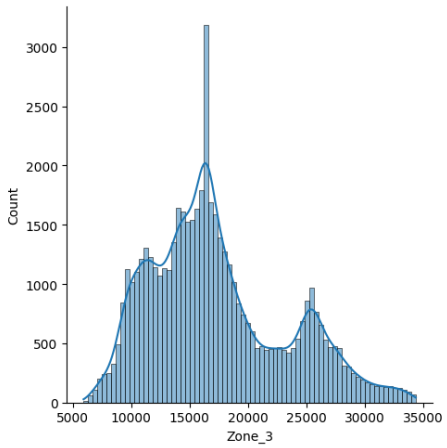


(c) Korijenom transformirana iskrivljenost

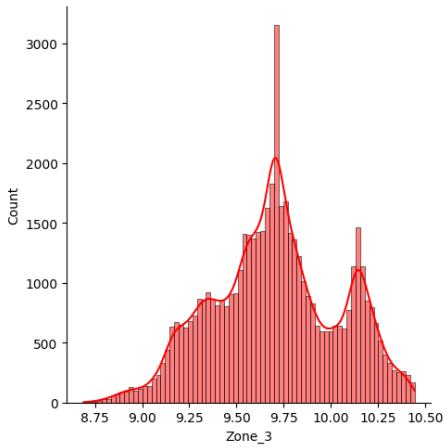


(d) Kubno transormirana iskrivljenost

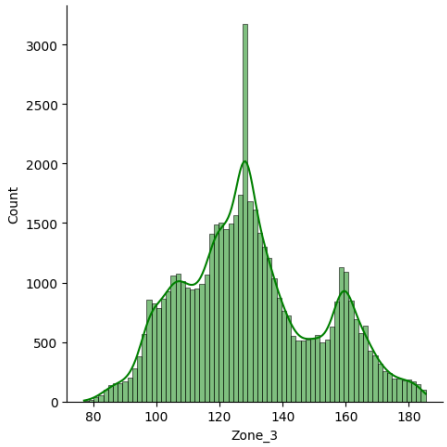
Slika 7.7: Iskrivljenosti Power consumption of Tetouan city baze podataka 2 zone



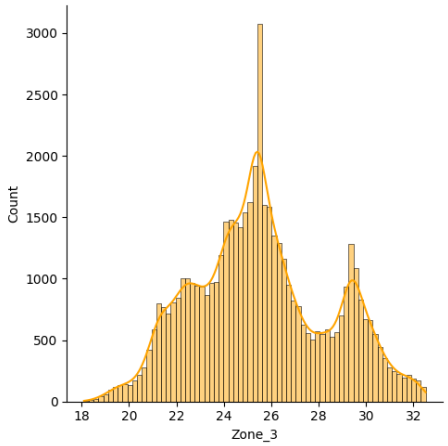
(a) Normalna iskrivljenost



(b) Logaritamski transformirana iskrivljenost

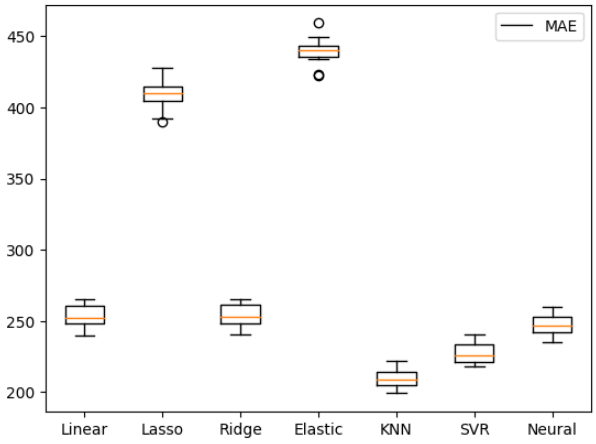


(c) Korijenom transformirana iskrivljenost

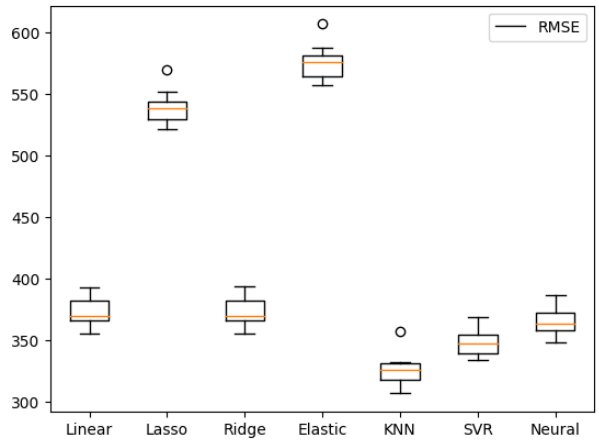


(d) Kubno transormirana iskrivljenost

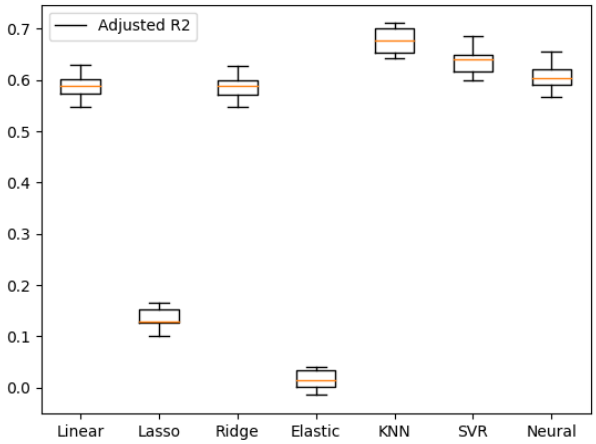
Slika 7.8: Iskrivljenosti Power consumption of Tetouan city baze podataka 3 zone



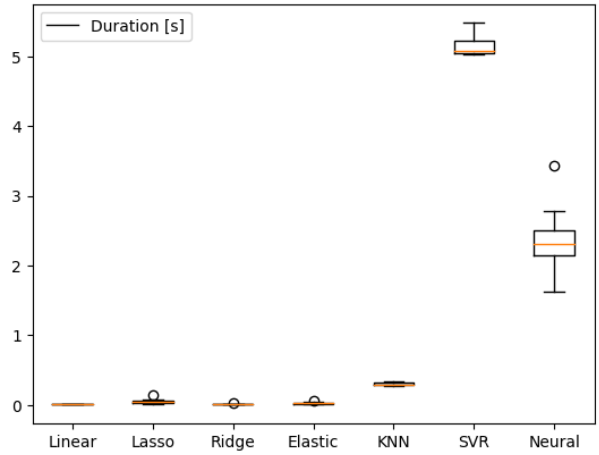
(a) MAE vrijednosti



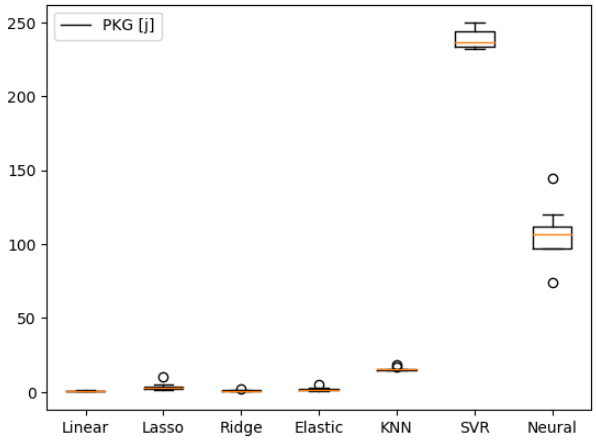
(b) RMSE vrijednosti



(c) Adjusted R^2 vrijednosti

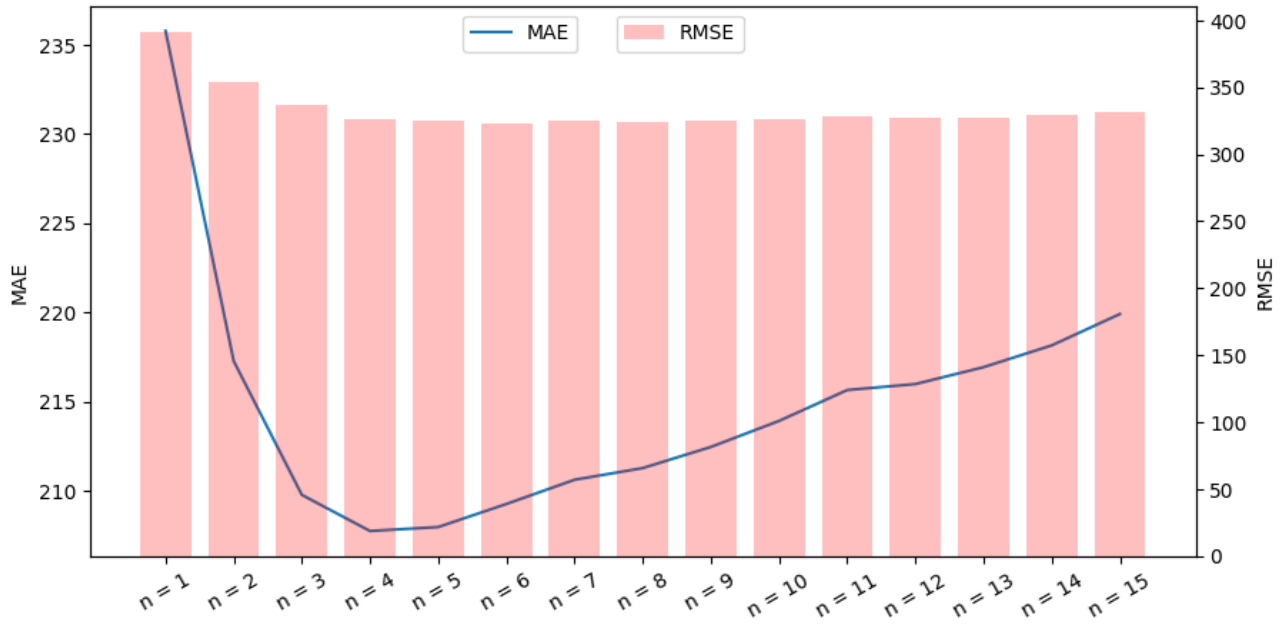


(d) Vrijednosti trajanja treniranja

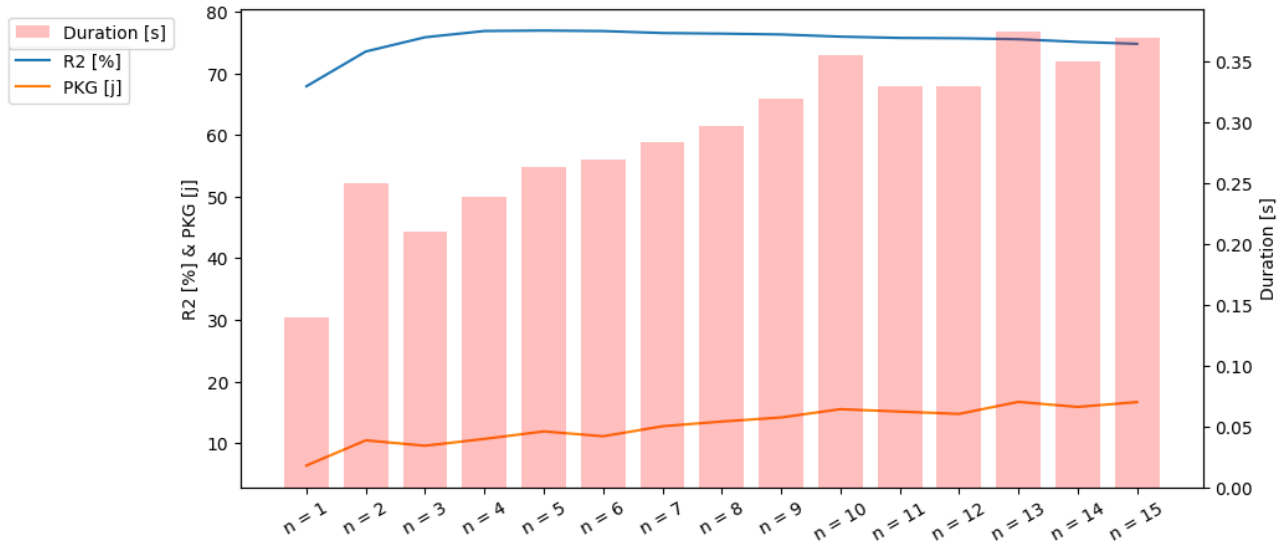


(e) PKG vrijednosti

Slika 7.9: Vrijednosti 10 grupa validacije za Seoul Bike Sharing Demand baze podataka

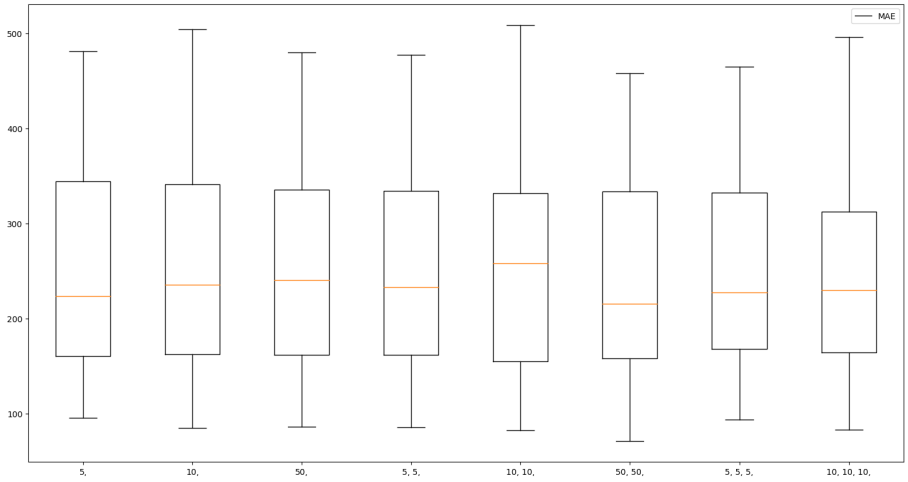


(a) MAE i RMSE vrijednosti

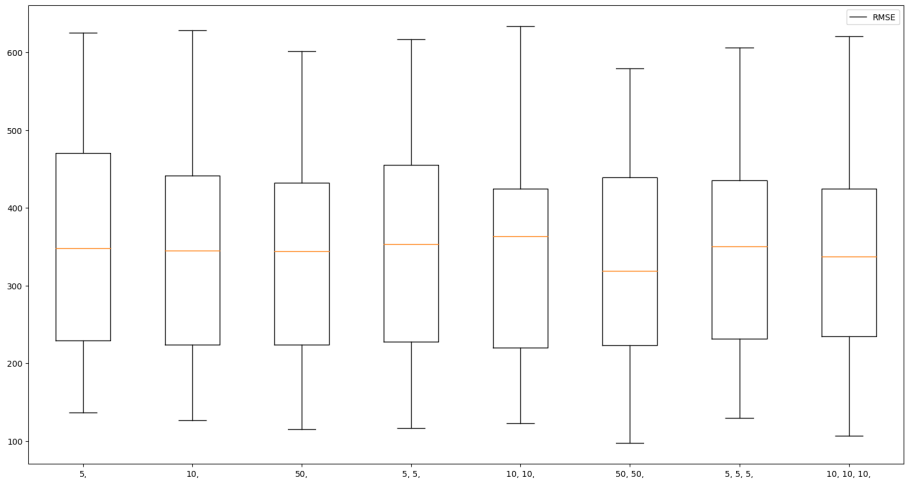


(b) R^2, vrijeme trajanja treniranja modela i PKG vrijednosti

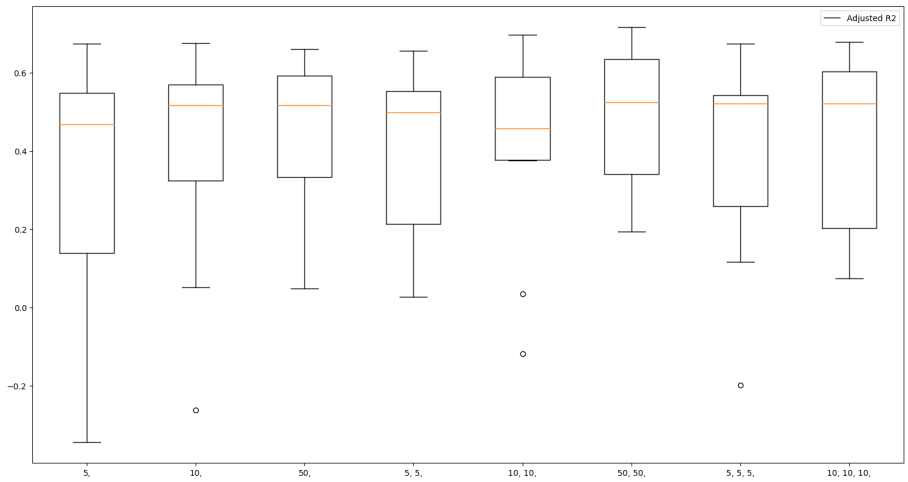
Slika 7.10: Vrijednosti KNN modela za različite parametre susjeda Seoul Bike Sharing Demand baze podataka



(a) MAE vrijednosti



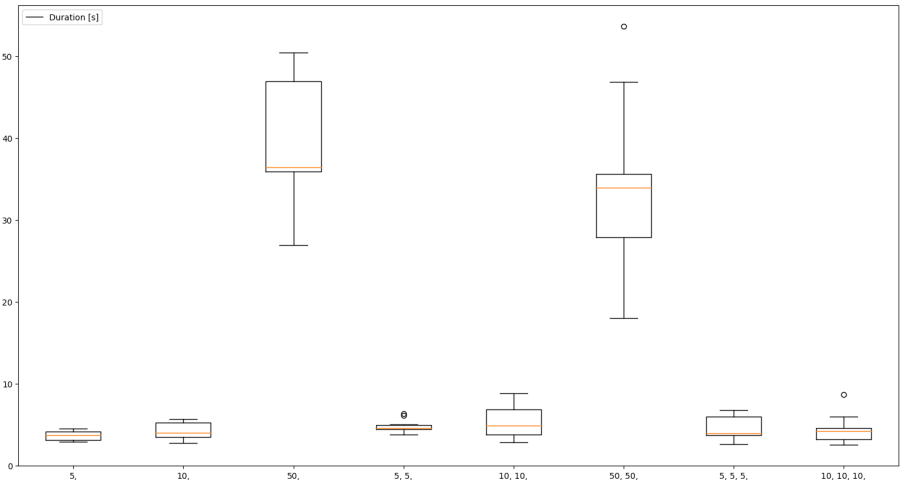
(b) RMSE vrijednosti



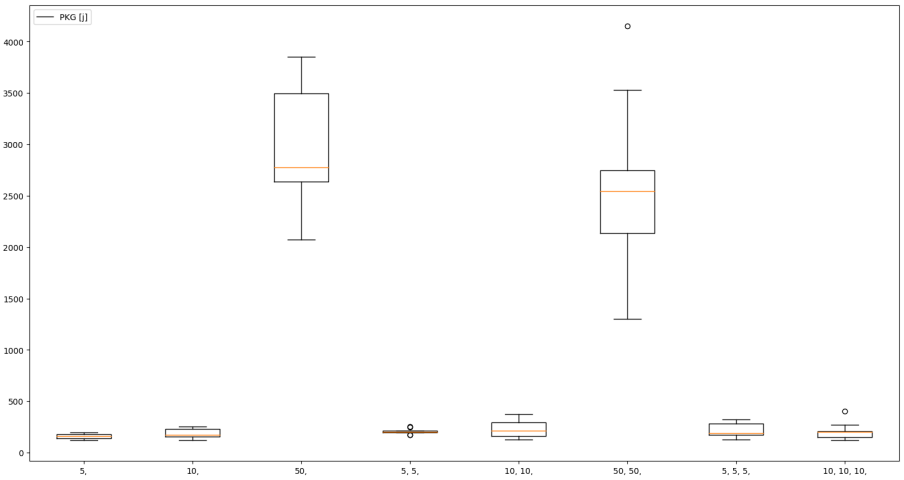
(c) Adjusted R^2 vrijednosti

Slika 7.11: Vrijednosti modela neuralne mreže za različite parametre neurona i skrivenih slojeva Seoul Bike Sharing Demand baze podataka

(Nastavlja se na sljedećoj stranici)



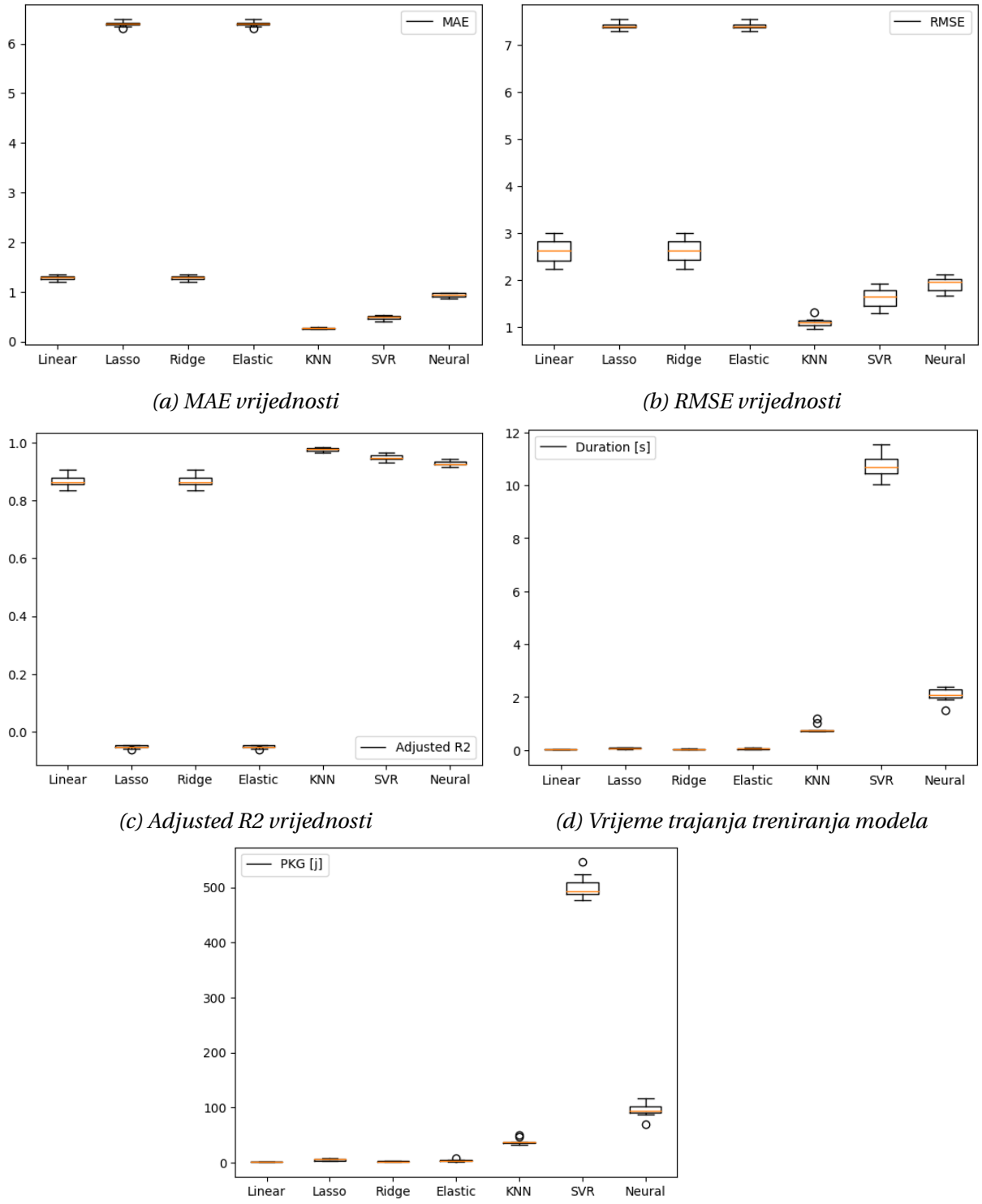
(d) Vrijeme trajanja treniranja modela



(e) PKG vrijednosti

Slika 7.11: Vrijednosti modela neuralne mreže za različite parametre neurona i skrivenih slojeva Seoul Bike Sharing Demand baze podataka

(Nastavak s prijašnje stranice)



(a) MAE vrijednosti

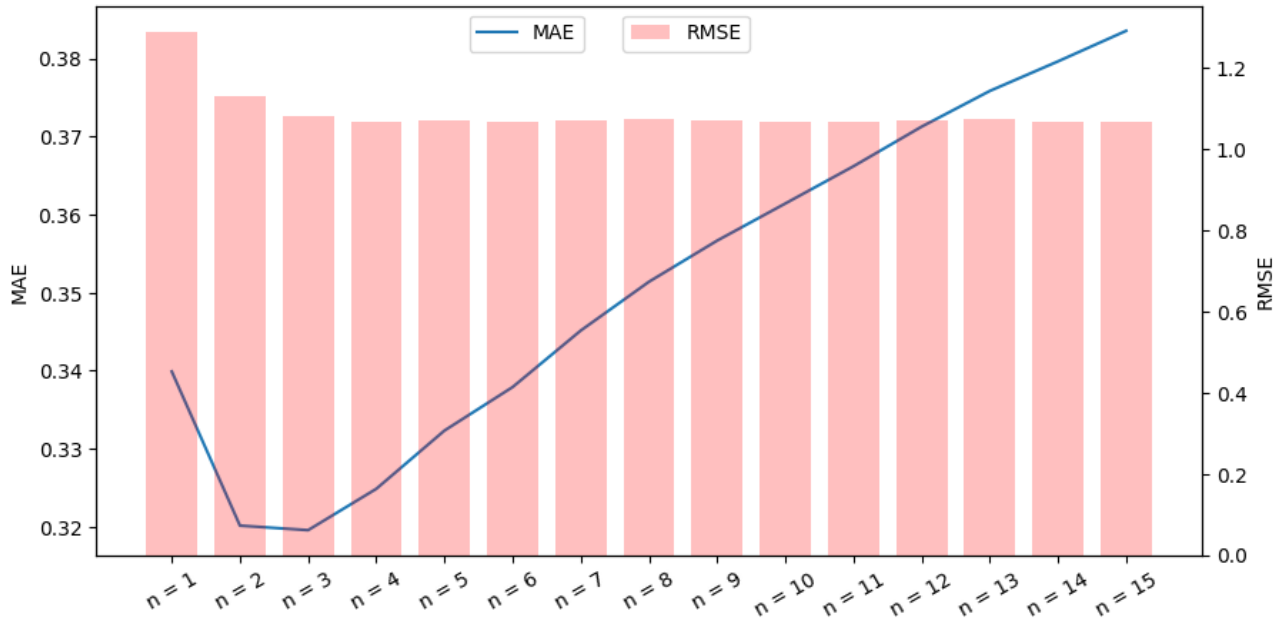
(b) RMSE vrijednosti

(c) Adjusted R2 vrijednosti

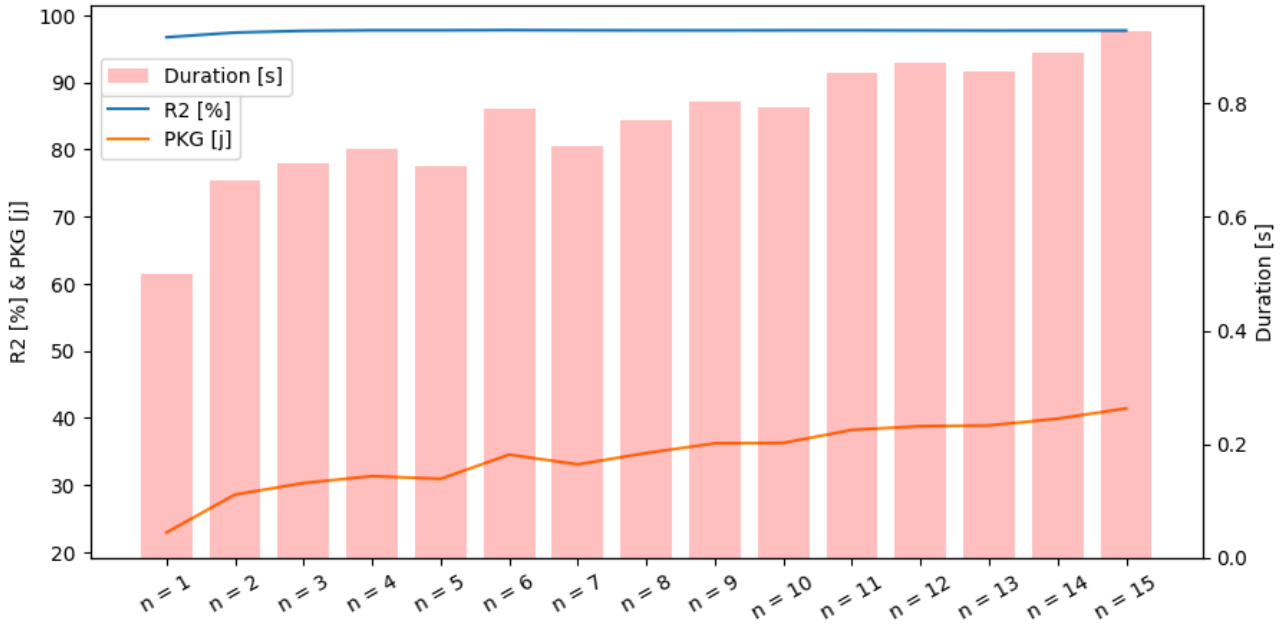
(d) Vrijeme trajanja treniranja modela

(e) PKG vrijednosti

Slika 7.12: Vrijednosti 10 grupa validacije za Steel Industry Energy Consumption baze podataka

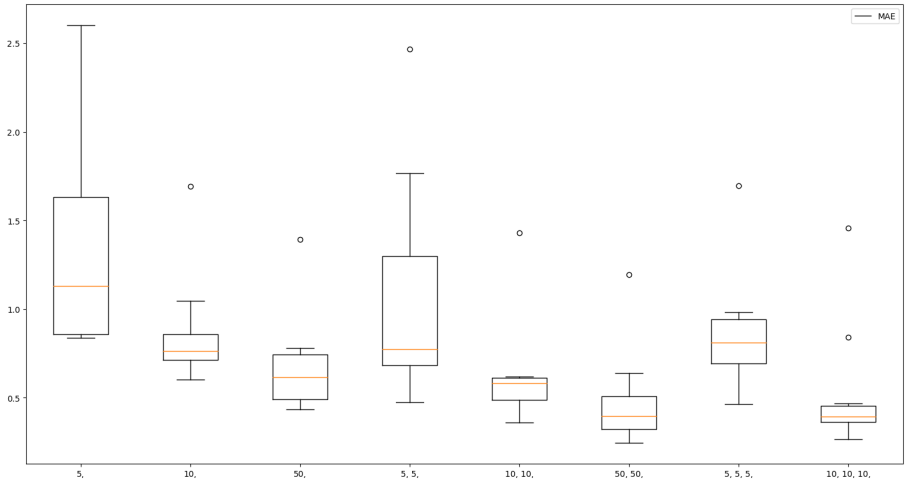


(a) MAE i RMSE vrijednosti

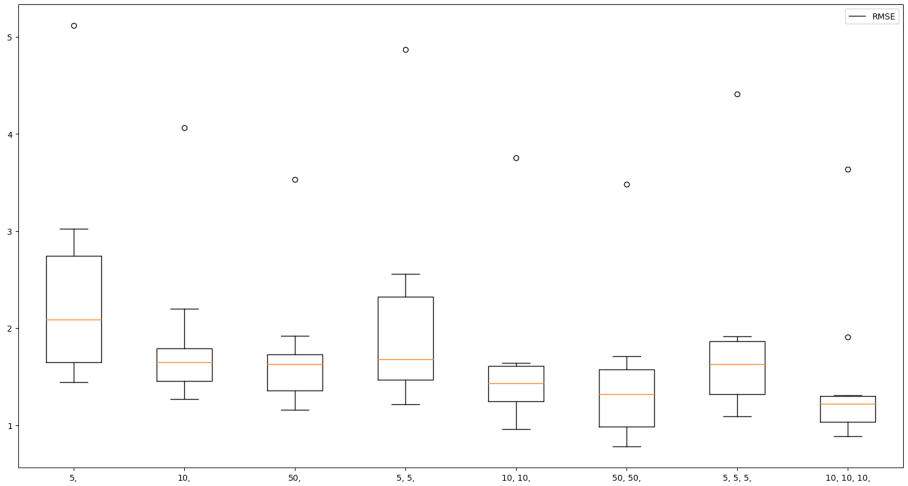


(b) R^2, vrijeme trajanja treniranja modela i PKG vrijednosti

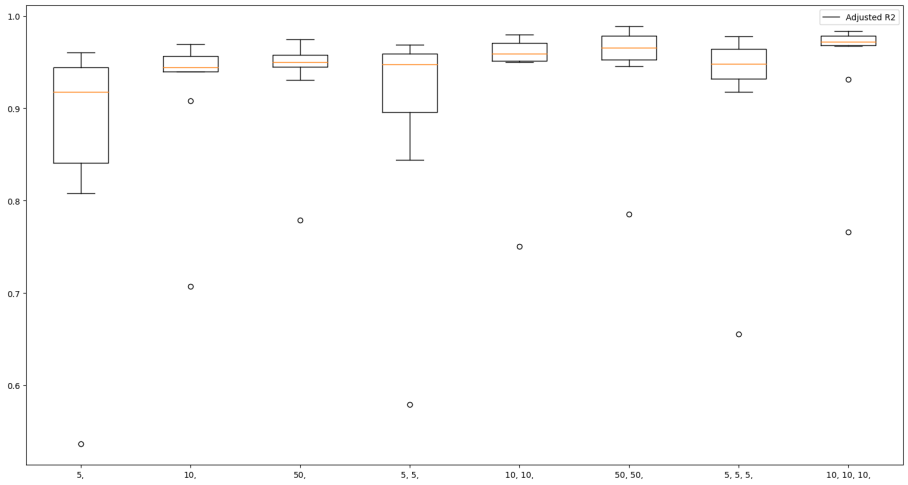
Slika 7.13: Vrijednosti KNN modela za različite parametre susjeda Steel Industry Energy Consumption baze podataka



(a) MAE vrijednosti



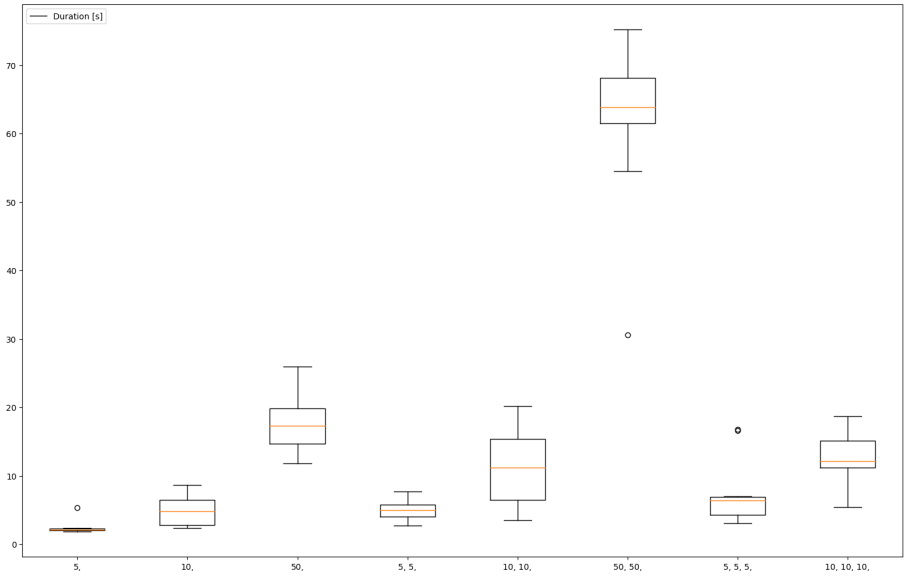
(b) RMSE vrijednosti



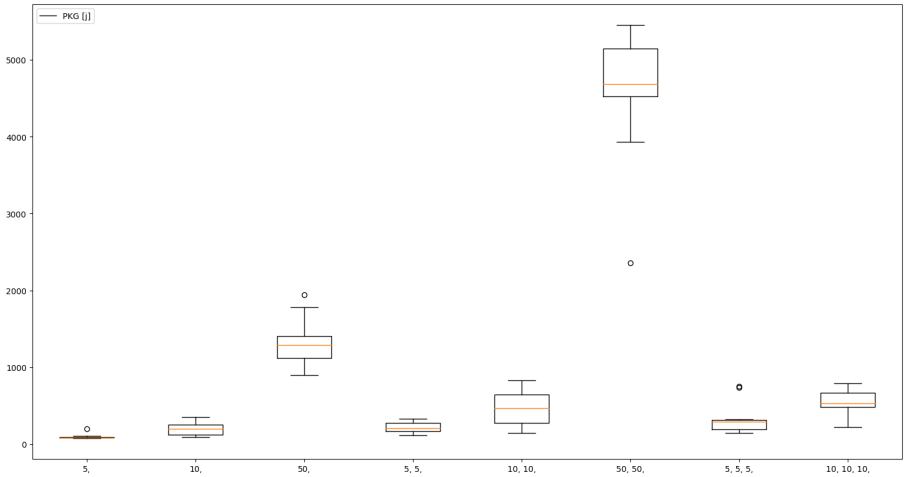
(c) Adjusted R^2 vrijednosti

Slika 7.14: Vrijednosti modela neuralne mreže za različite parametre neurona i skrivenih slojeva Steel Industry Energy Consumption baze podataka

(Nastavlja se na sljedećoj stranici)



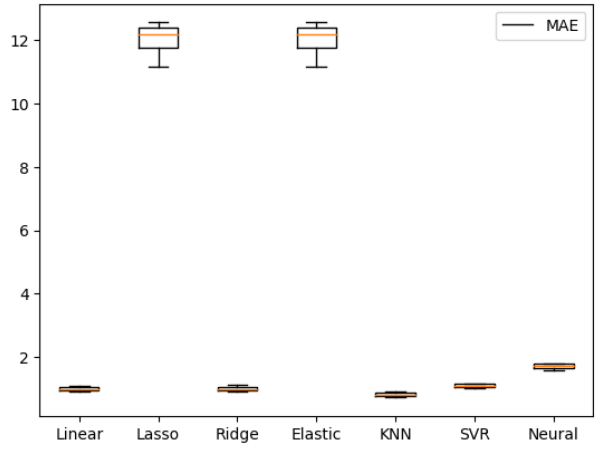
(d) Vrijeme trajanja treniranja modela



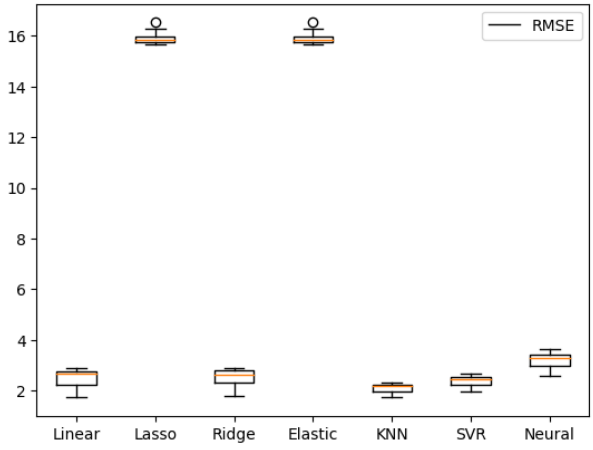
(e) PKG vrijednosti

Slika 7.14: Vrijednosti modela neuralne mreže za različite parametre neurona i skrivenih slojeva Steel Industry Energy Consumption baze podataka

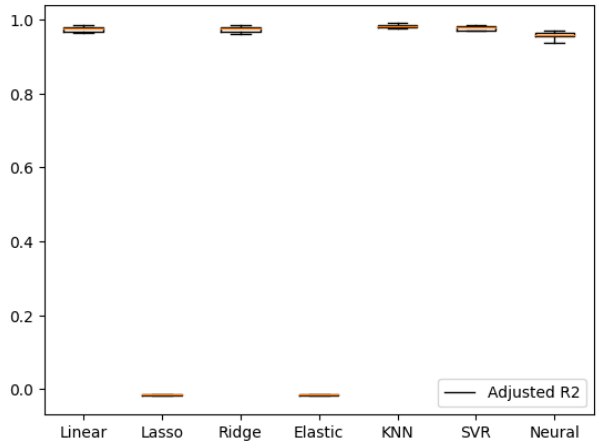
(Nastavak s prijašnje stranice)



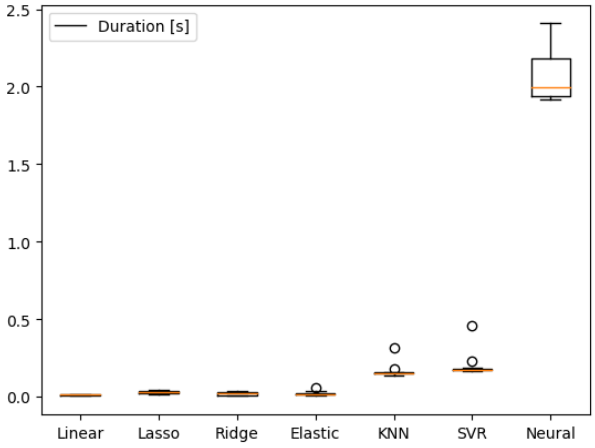
(a) MAE vrijednosti



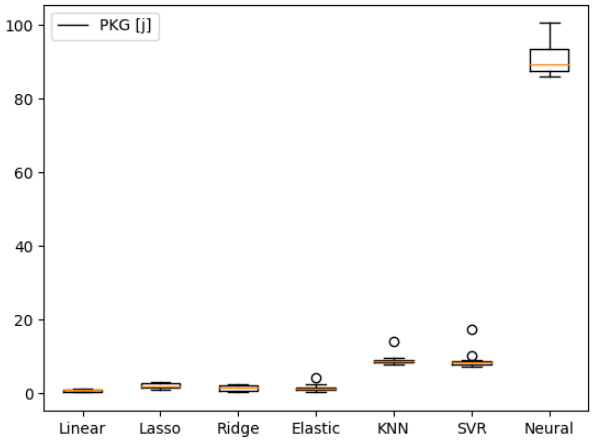
(b) RMSE vrijednosti



(c) Adjusted R^2 vrijednosti

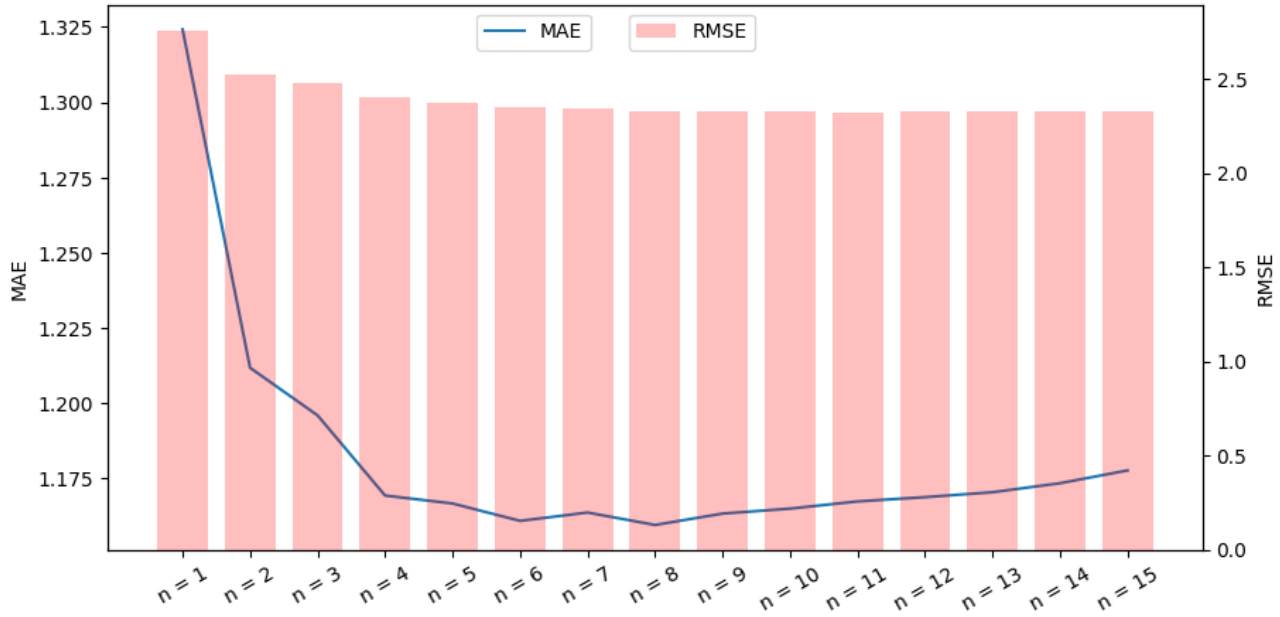


(d) Vrijeme trajanja treniranja modela

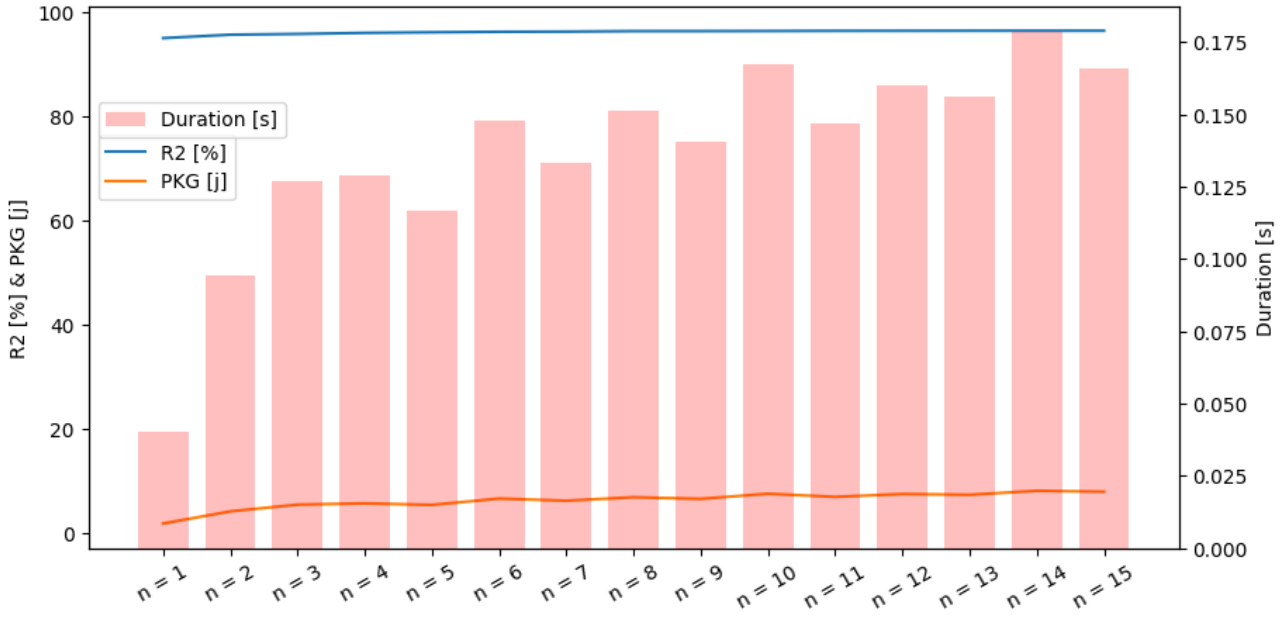


(e) PKG vrijednosti

Slika 7.15: Vrijednosti 10 grupa validacije za Gas Turbine CO and NOX Emission baze podataka

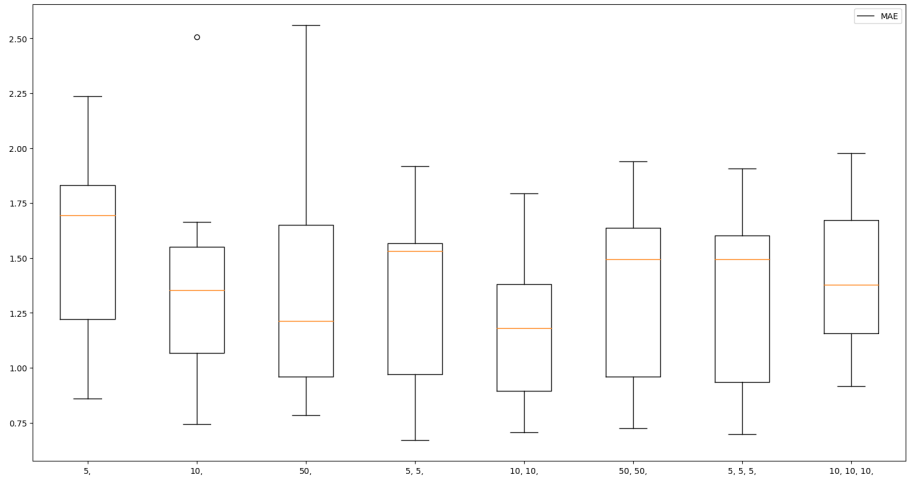


(a) MAE i RMSE vrijednosti

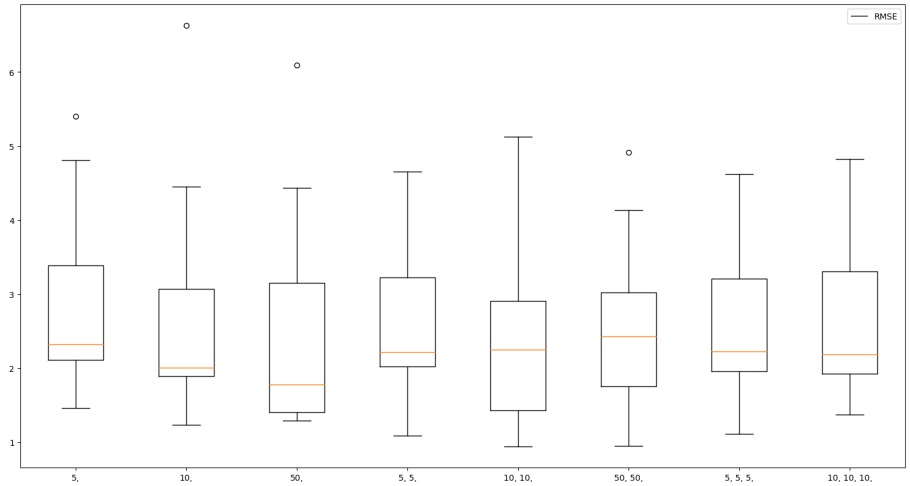


(b) R^2, vrijeme trajanja treniranja modela i PKG vrijednosti

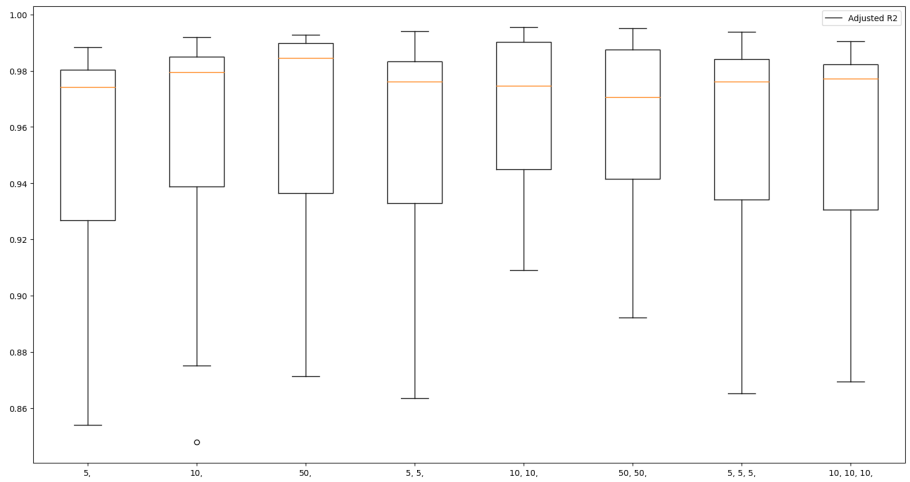
Slika 7.16: Vrijednosti KNN modela za različite parametre susjeda Gas Turbine CO and NOX Emission baze podataka



(a) MAE vrijednosti



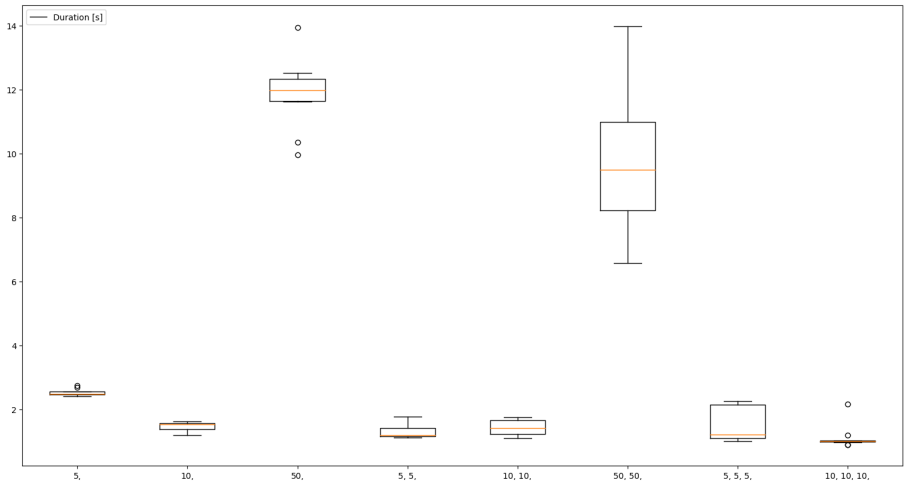
(b) RMSE modela



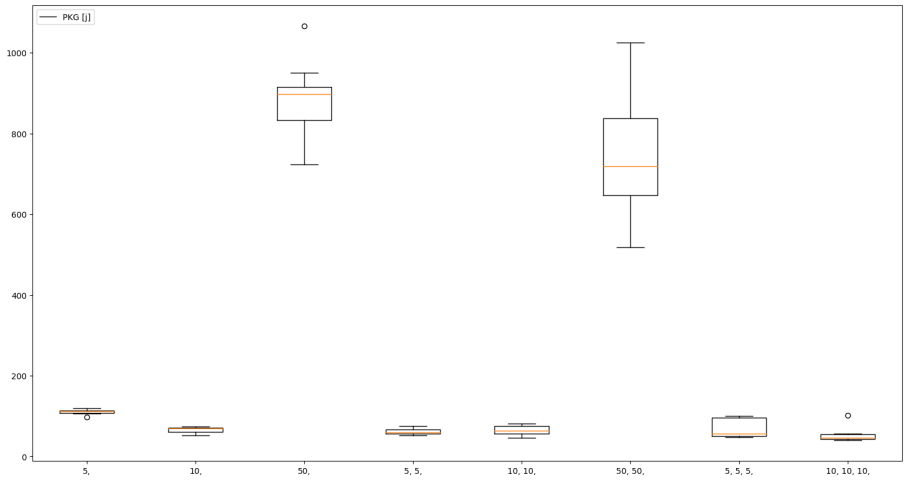
(c) Adjusted R^2 vrijednosti

Slika 7.17: Vrijednosti modela neuralne mreže za različite parametre neurona i skrivenih slojeva Gas Turbine CO and NOX Emission baze podataka

(Nastavlja se na sljedećoj stranici)



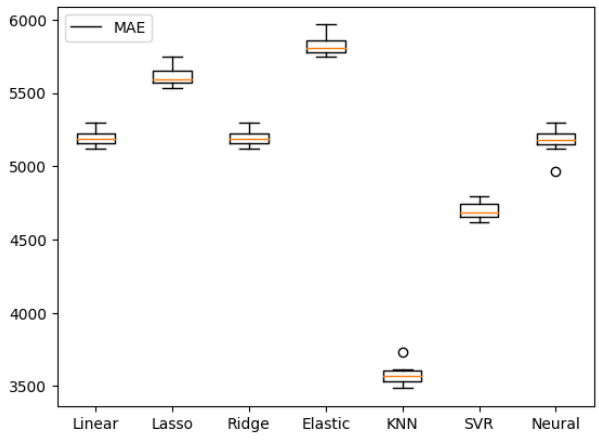
(d) Vrijeme trajanja treniranja modela



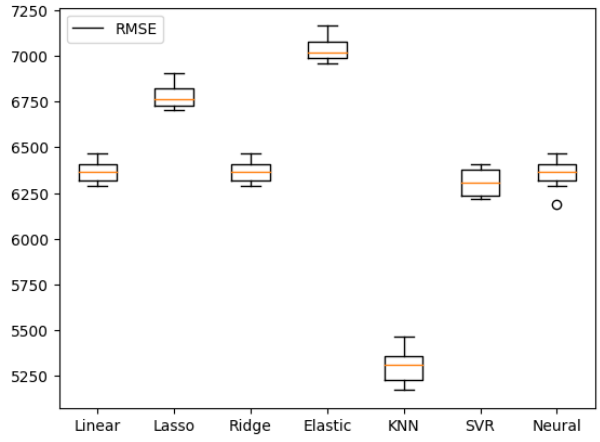
(e) PKG vrijednosti

Slika 7.17: Vrijednosti modela neuralne mreže za različite parametre neurona i skrivenih slojeva Gas Turbine CO and NOX Emission baze podataka

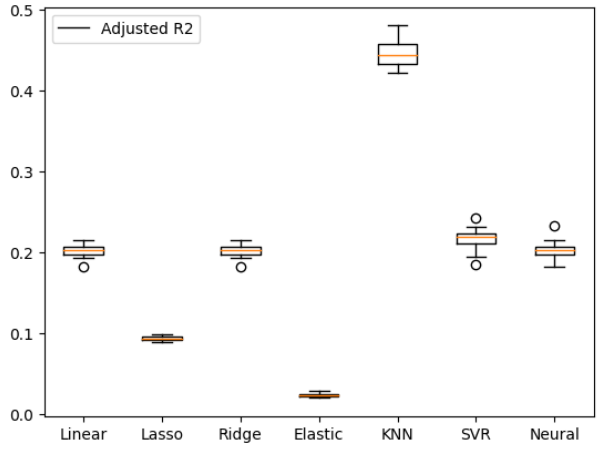
(Nastavak s prijašnje stranice)



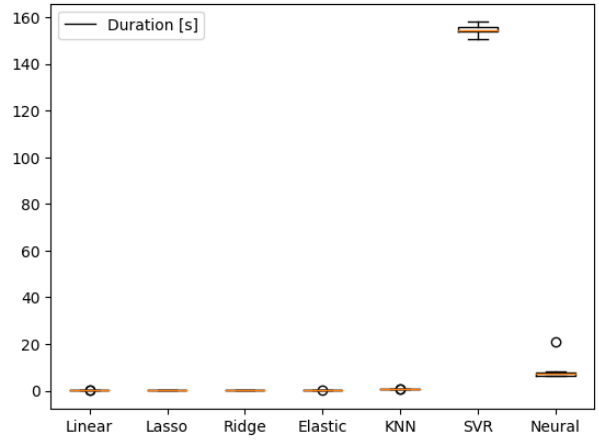
(a) MAE vrijednosti



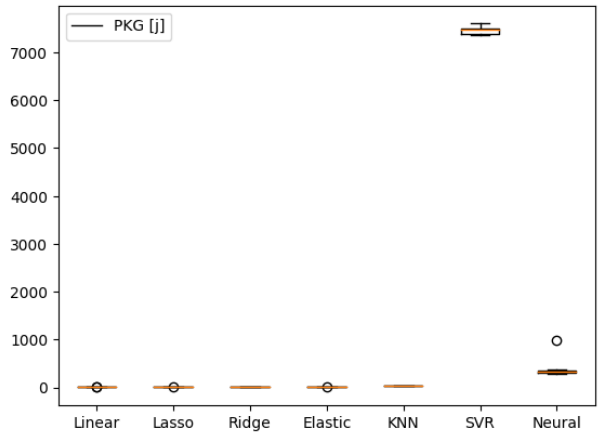
(b) RMSE vrijednosti



(c) Adjusted R^2 vrijednosti

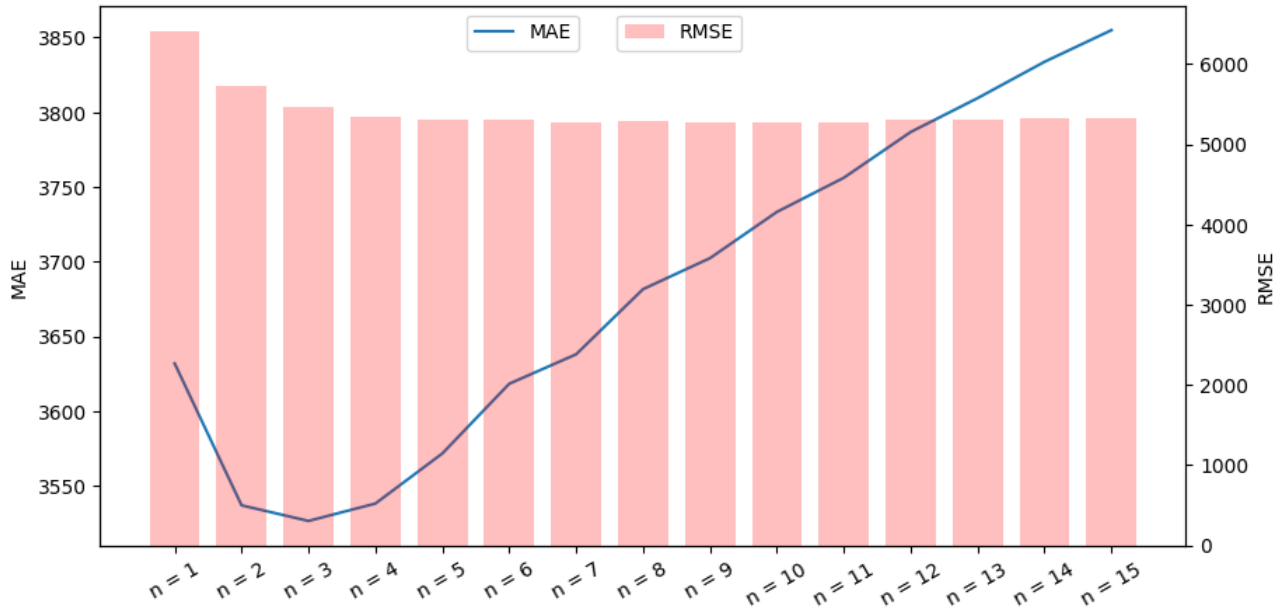


(d) Vrijeme trajanja treniranja modela

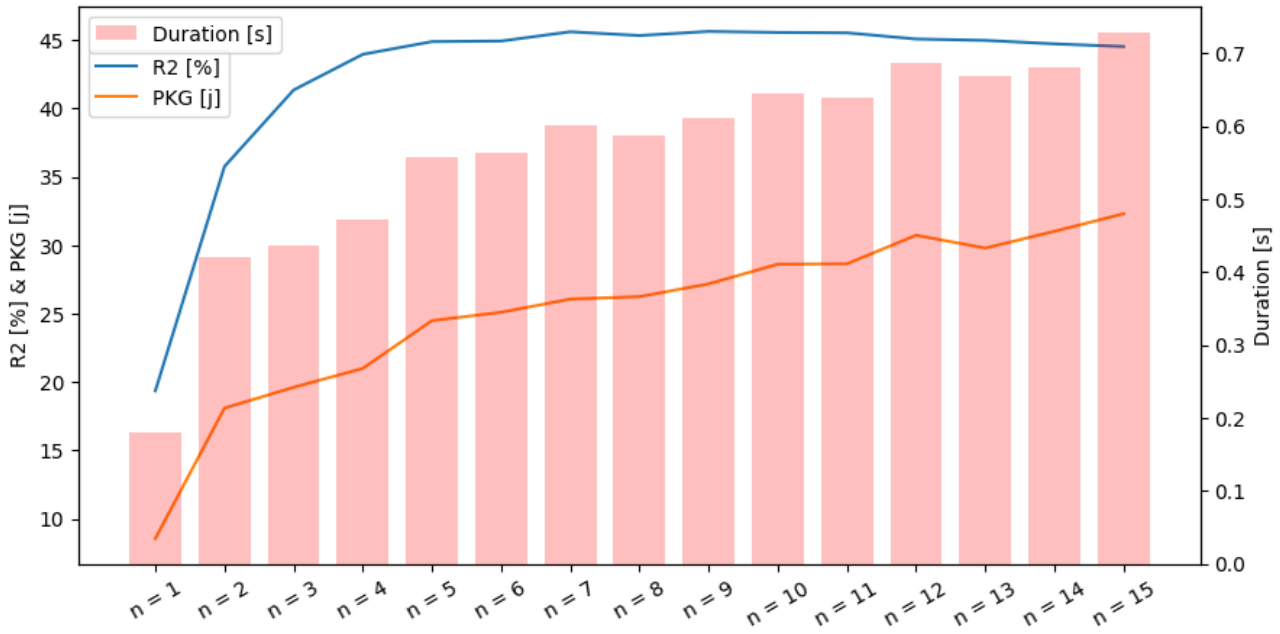


(e) PKG vrijednosti

Slika 7.18: Vrijednosti 10 grupa validacije za Power consumption of Tetouan city 1 zone baze podataka

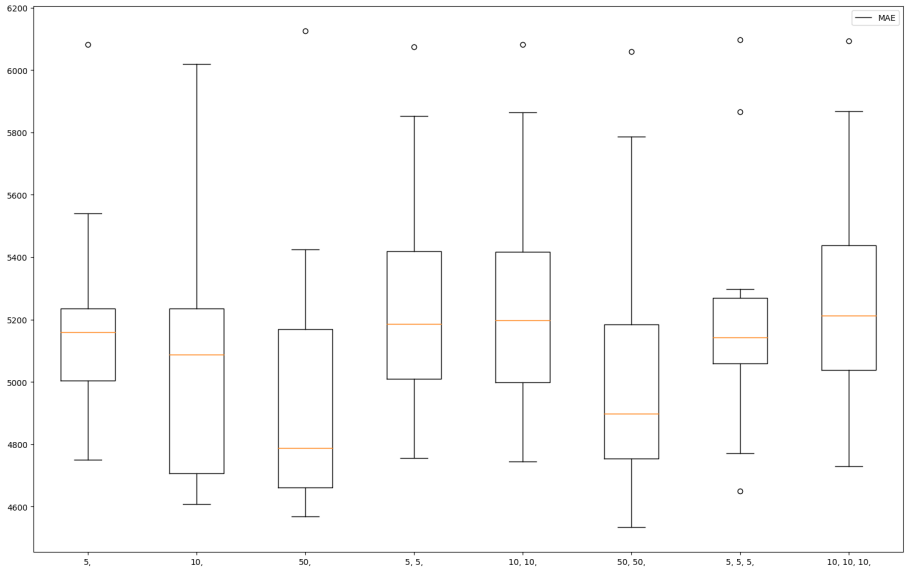


(a) MAE i RMSE vrijednosti

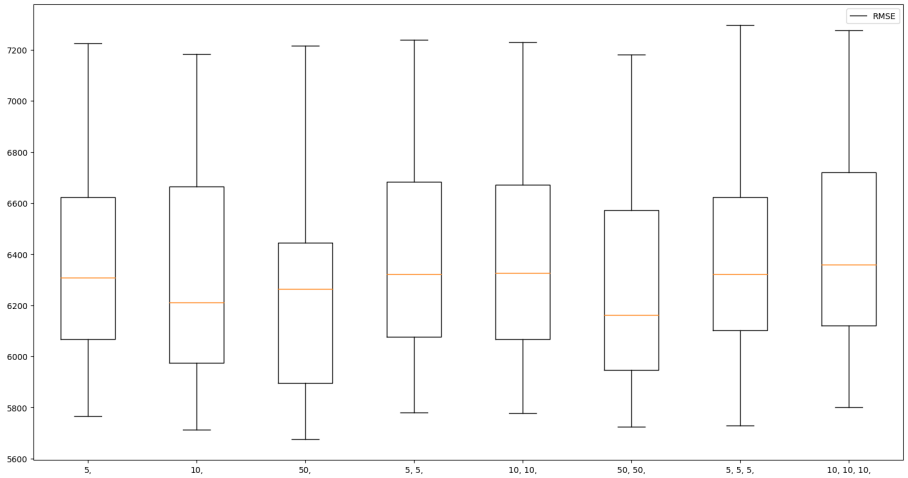


(b) R^2, vrijeme trajanja treniranja modela i PKG vrijednosti

Slika 7.19: Vrijednosti KNN modela za različite parametre susjeda Power consumption of Tetouan city 1 zone baze podataka



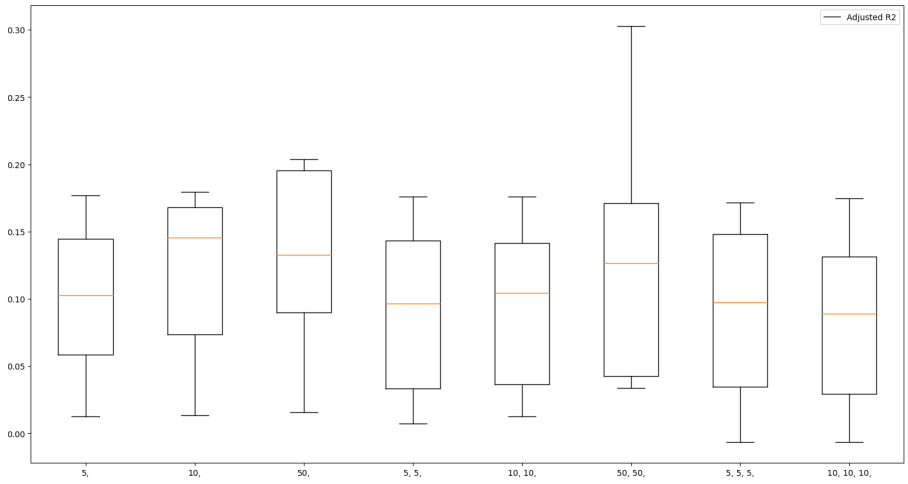
(a) MAE vrijednosti



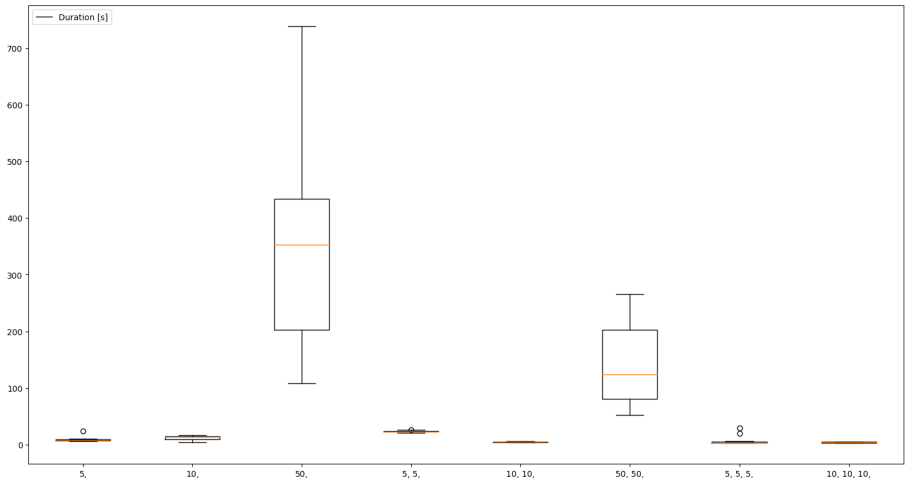
(b)RMSE vrijednosti

Slika 7.20: Vrijednosti modela neuralne mreže za različite parametre neurona i skrivenih slojeva Power consumption of Tetouan city 1 zone baze podataka

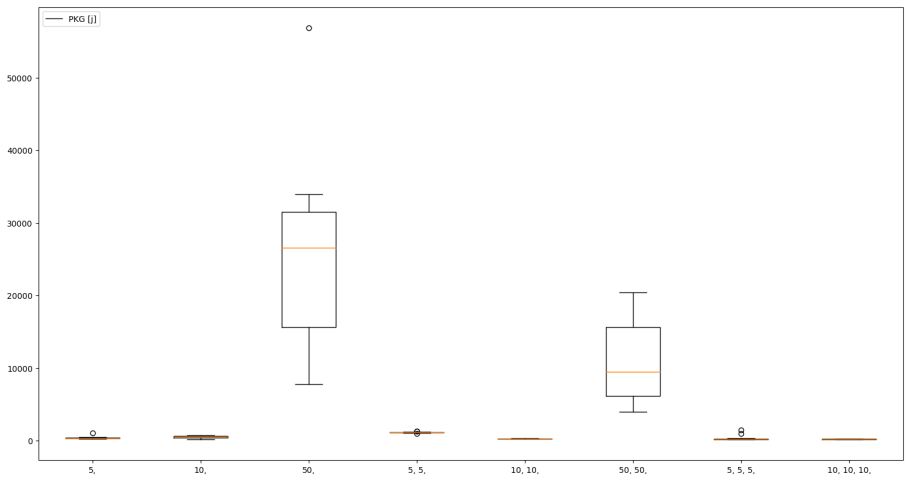
(Nastavlja se na sljedećoj stranici)



(c) Adjusted R^2 vrijednosti



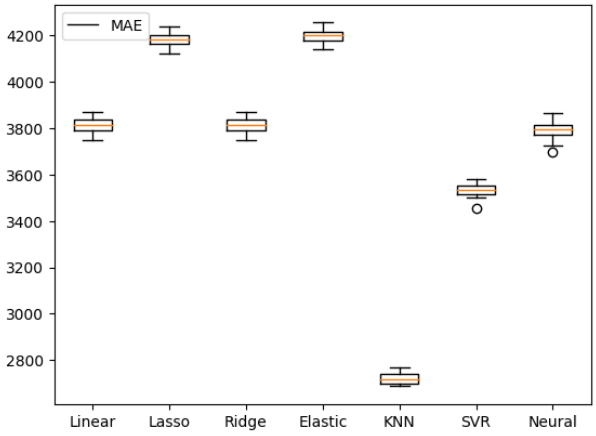
(d) Vrijeme trajanja treniranja modela



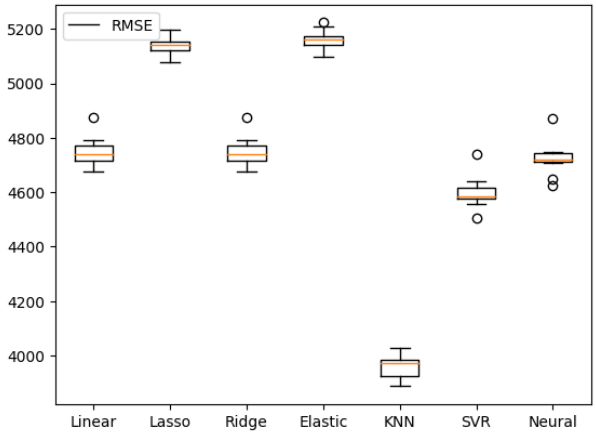
(e) PKG vrijednosti

Slika 7.20: Vrijednosti modela neuralne mreže za različite parametre neurona i skrivenih slojeva Power consumption of Tetouan city 1 zone baze podataka

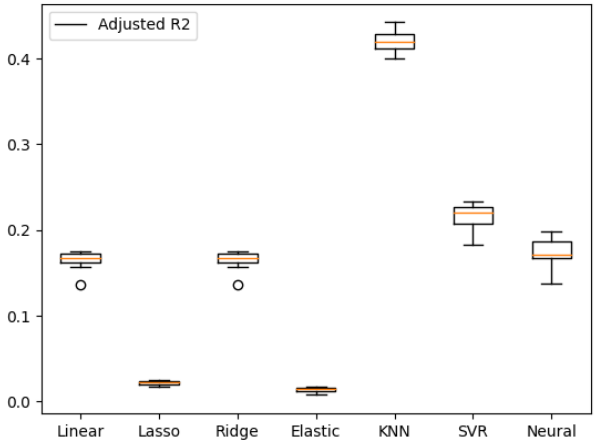
(Nastavak s prijašnje stranice)



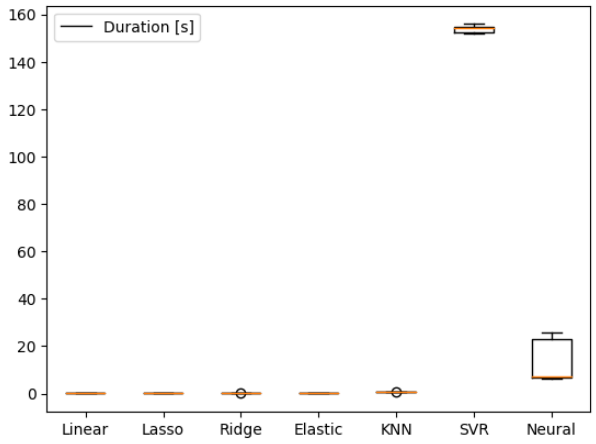
(a) MAE vrijednosti



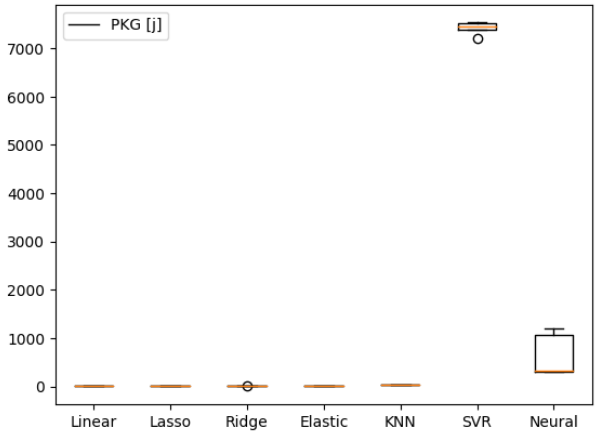
(b) RMSE vrijednosti



(c) Adjusted R^2 vrijednosti

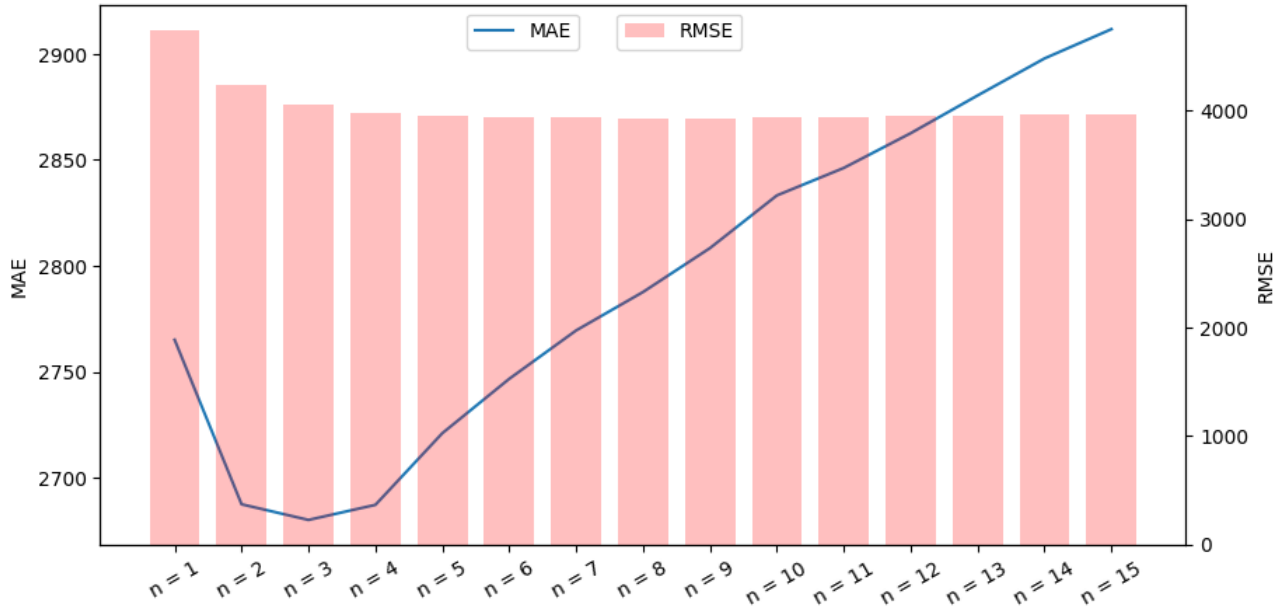


(d) Vrijeme trajanja treniranja modela

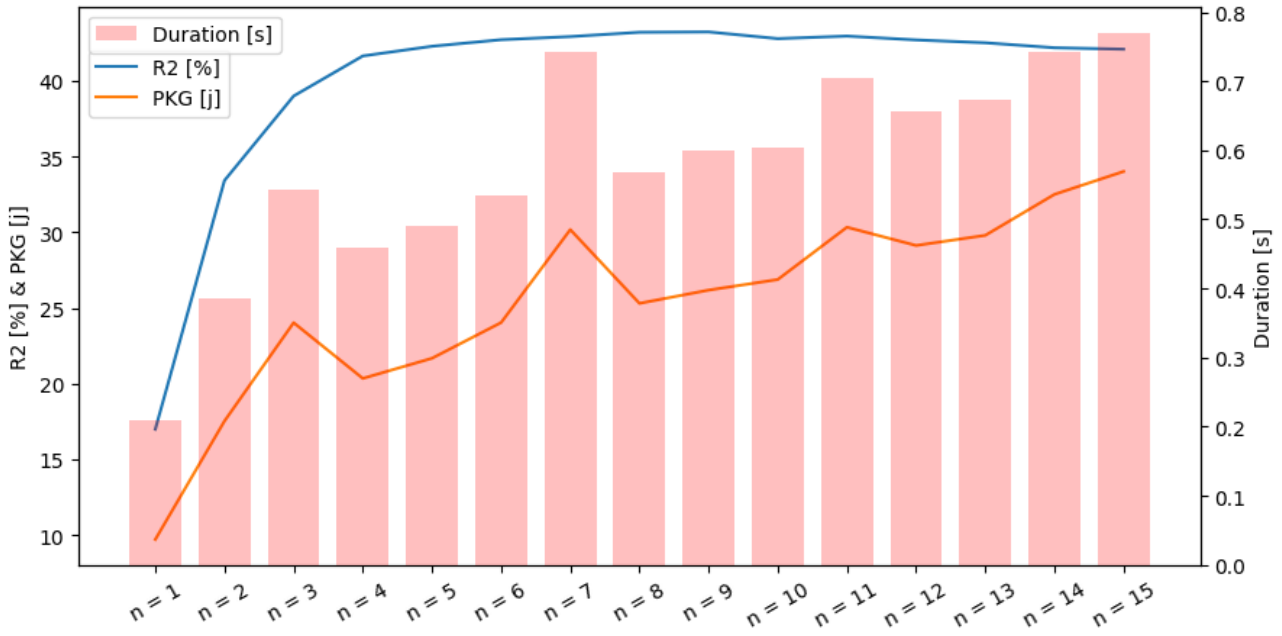


(e) PKG vrijednosti

Slika 7.21: Vrijednosti 10 grupa validacije za Power consumption of Tetouan city 2 zone baze podataka

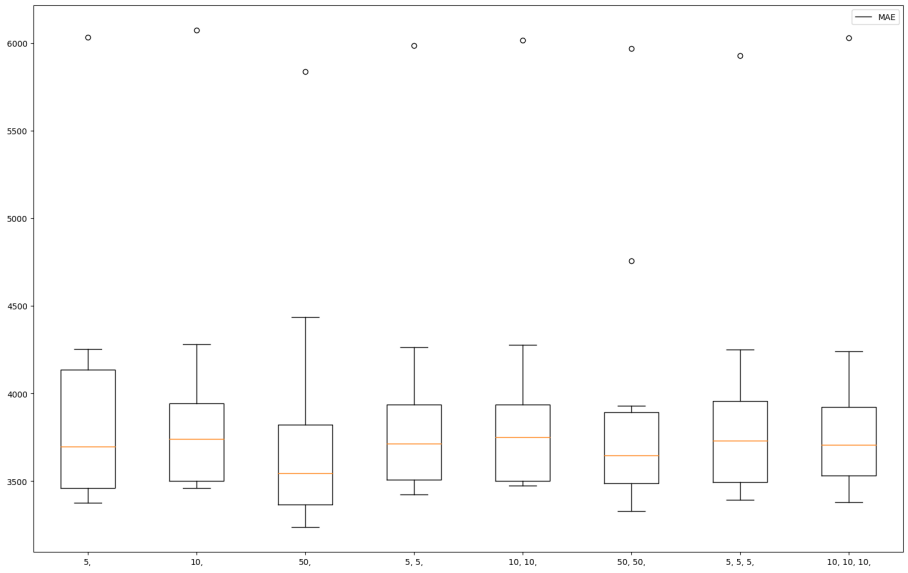


(a) MAE i RMSE vrijednosti

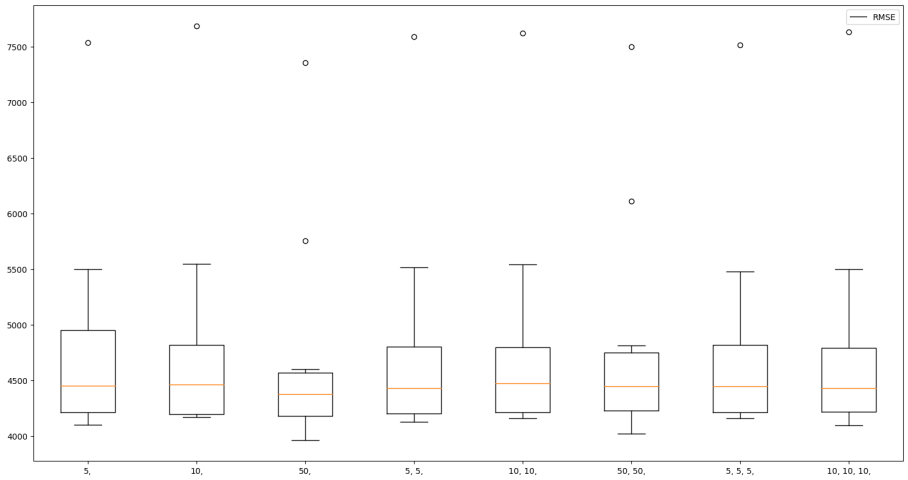


(b) R^2, vrijeme trajanja treniranja modela i PKG vrijednosti

Slika 7.22: Vrijednosti KNN modela za različite parametre susjeda Power consumption of Tetouan city 2 zone baze podataka



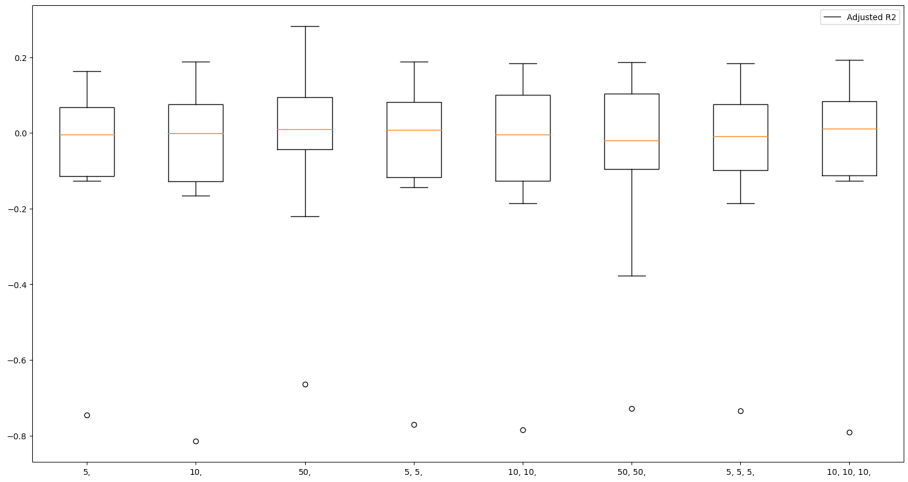
(a) MAE vrijednosti



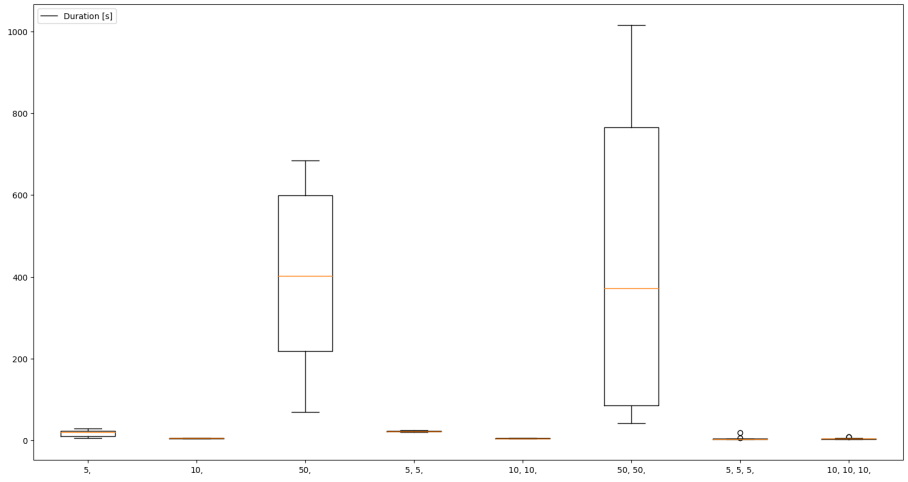
(b) RMSE vrijednosti

Slika 7.23: Vrijednosti modela neuralne mreže za različite parametre neurona i skrivenih slojeva Power consumption of Tetouan city 2 zone baze podataka

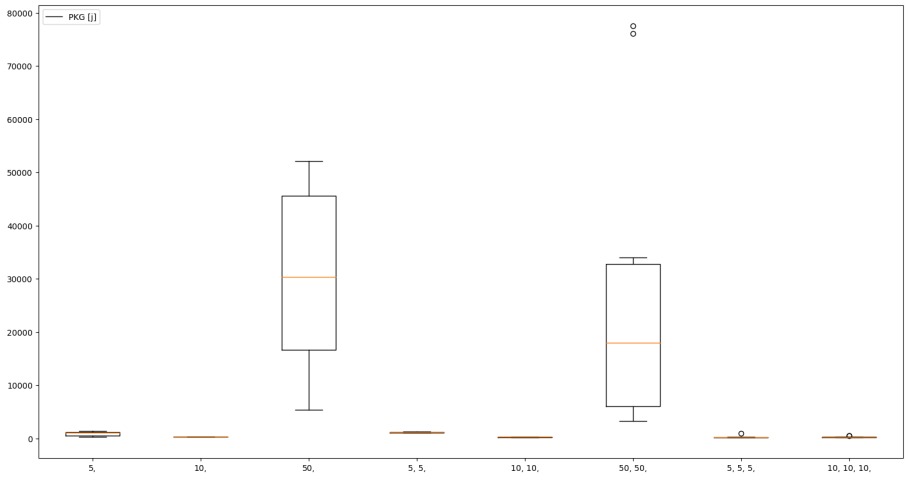
(Nastavlja se na sljedećoj stranici)



(c) Adjusted R^2 vrijednosti



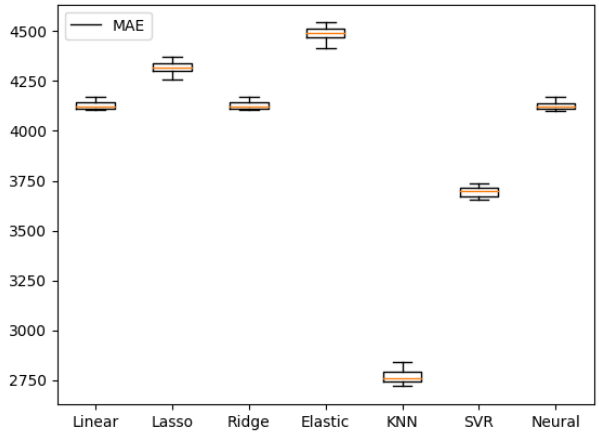
(d) Vrijeme trajanja treniranja modela



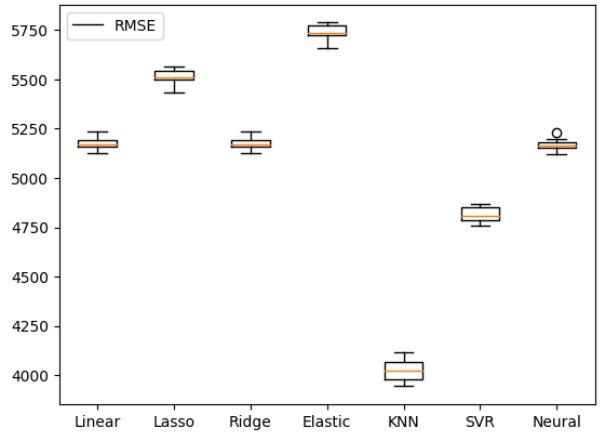
(e) PKG vrijednosti

Slika 7.23: Vrijednosti modela neuralne mreže za različite parametre neurona i skrivenih slojeva Power consumption of Tetouan city 2 zone baze podataka

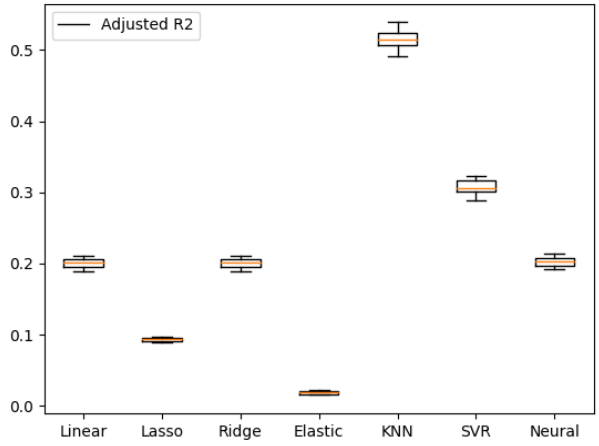
(Nastavak s prijašnje stranice)



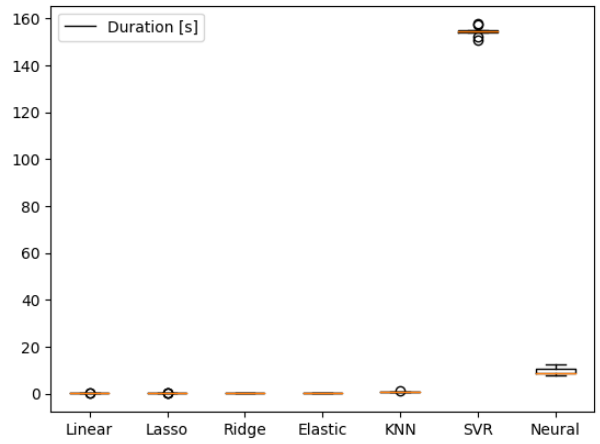
(a) MAE vrijednosti



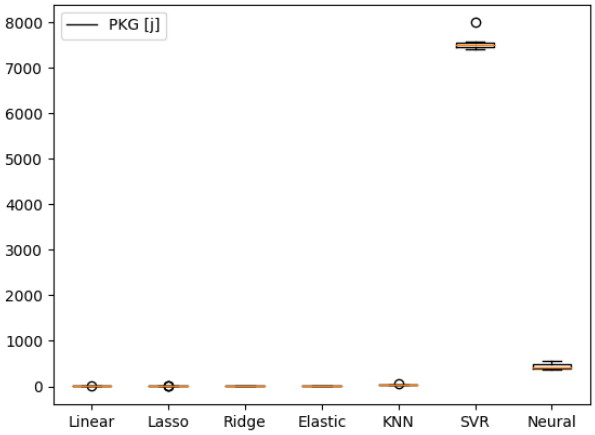
(b) RMSE vrijednosti



(c) Adjusted R^2 vrijednosti

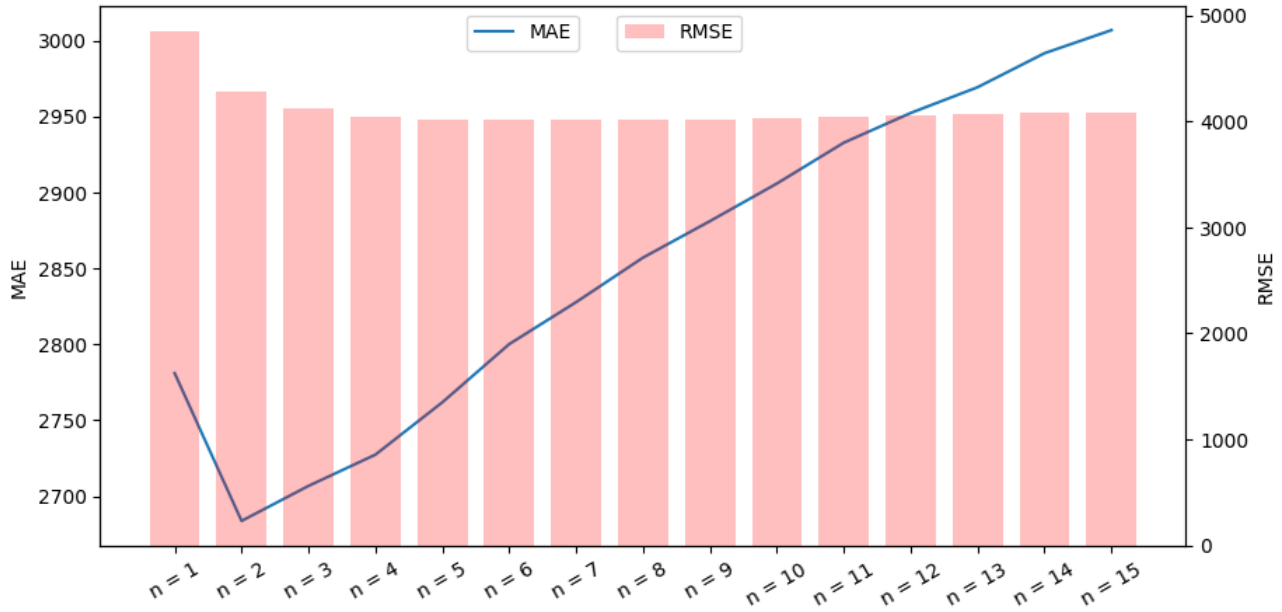


(d) Vrijeme trajanja treniranja modela

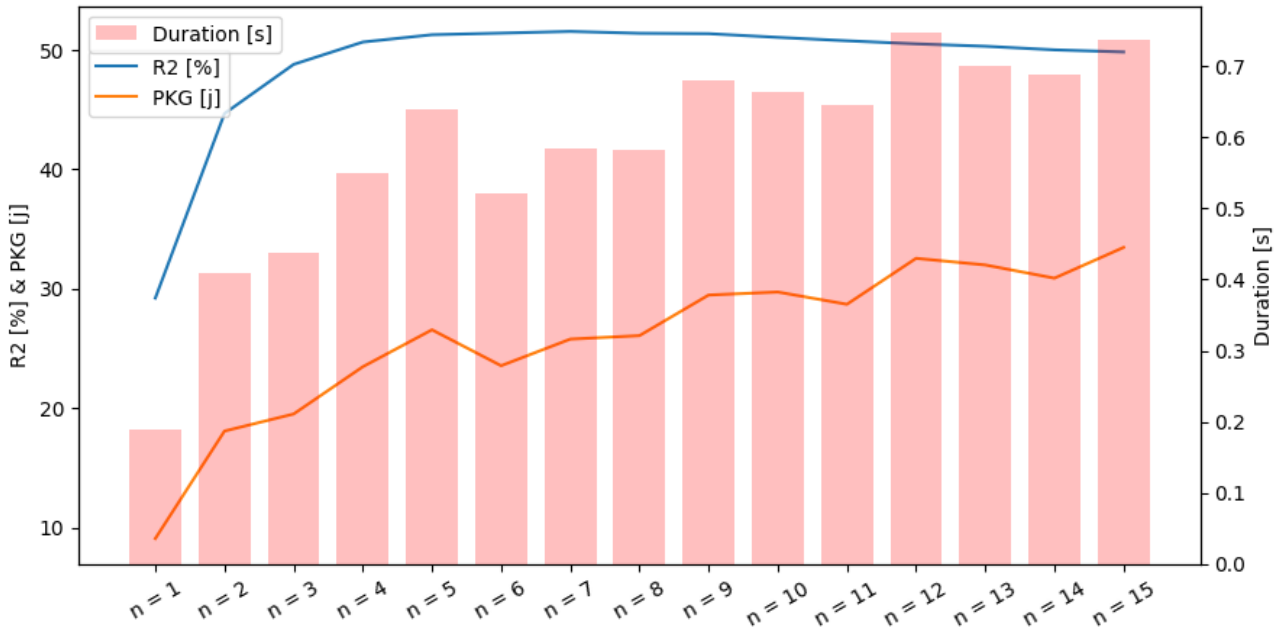


(e) PKG vrijednosti

Slika 7.24: Vrijednosti 10 grupa validacije za Seoul Power consumption of Tetouan city 3 zone baze podataka

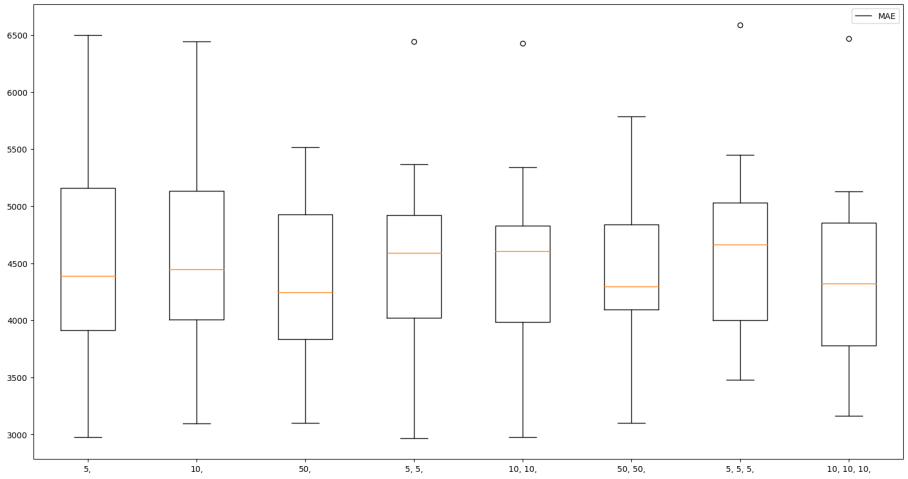


(a) MAE i RMSE vrijednosti

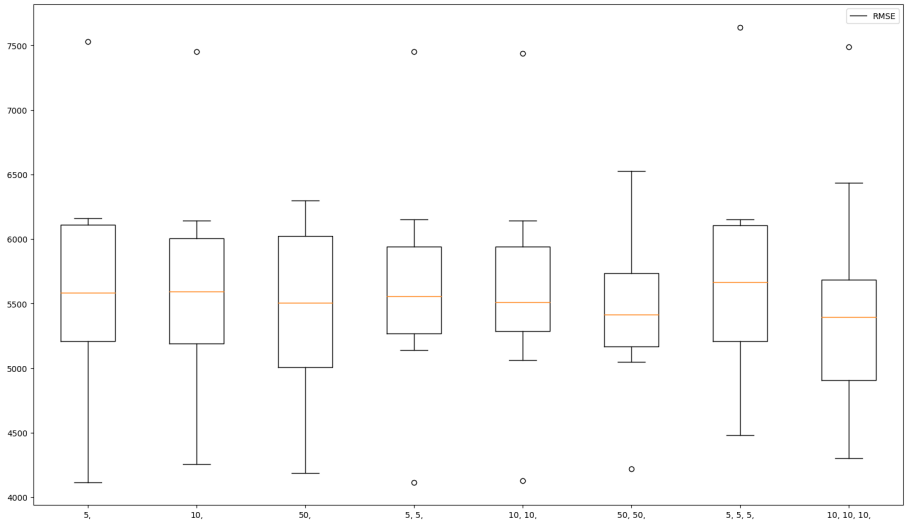


(b) R^2, vrijeme trajanja treniranja modela i PKG vrijednosti

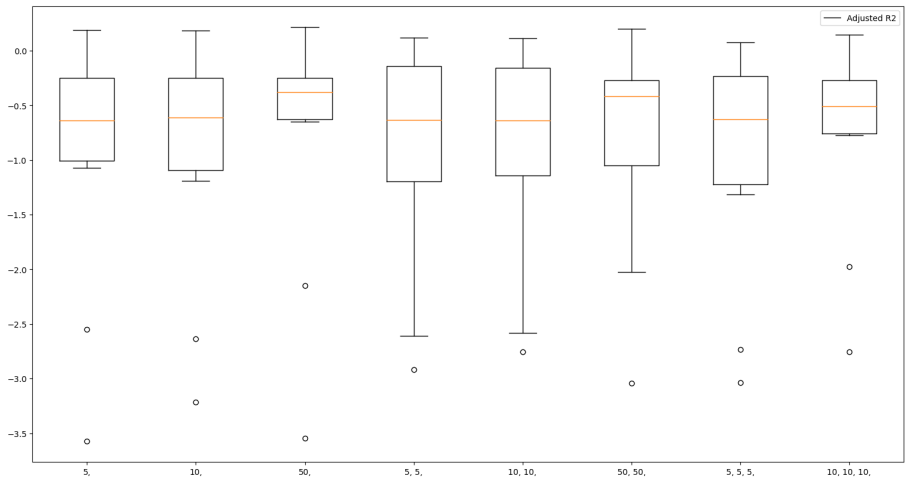
Slika 7.25: Vrijednosti KNN modela za različite parametre susjeda Power consumption of Tetouan city 3 zone baze podataka



(a) MAE vrijednosti



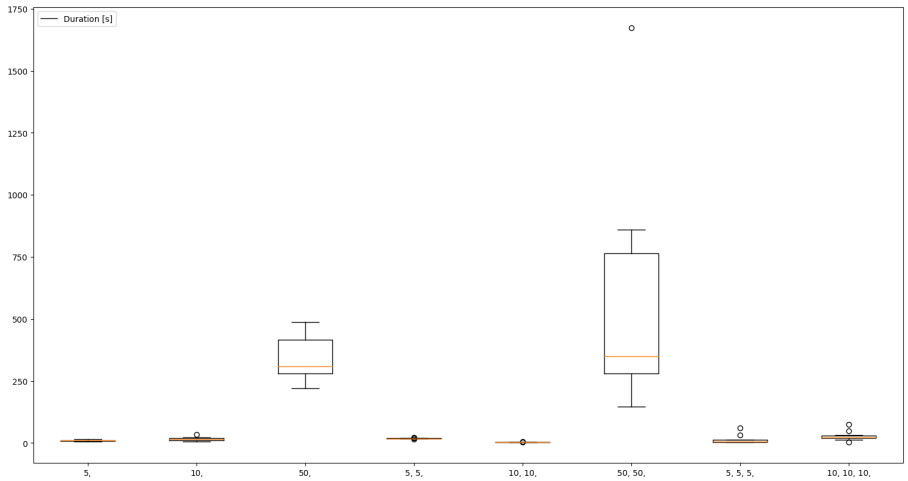
(b) RMSE vrijednosti



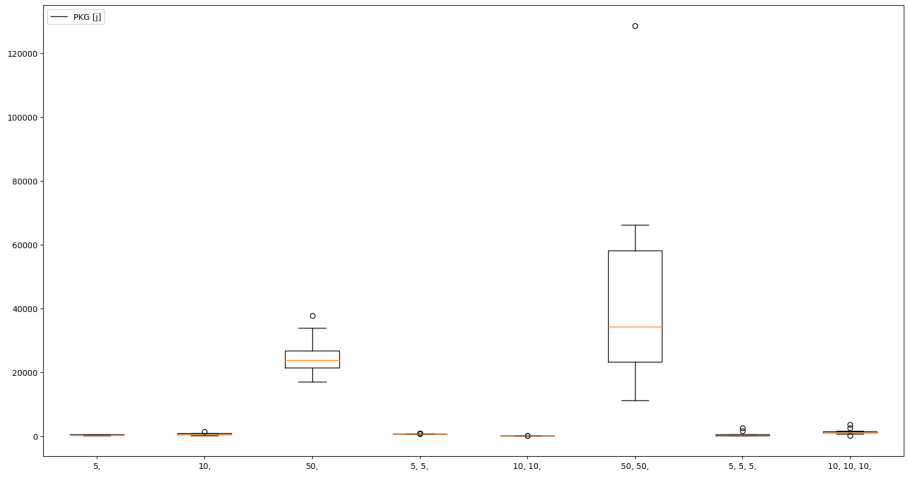
(c) Adjusted R^2 vrijednosti

Slika 7.26: Vrijednosti modela neuralne mreže za različite parametre neurona i skrivenih slojeva Power consumption of Tetouan city 3 zone baze podataka

(Nastavlja se na sljedećoj stranici)



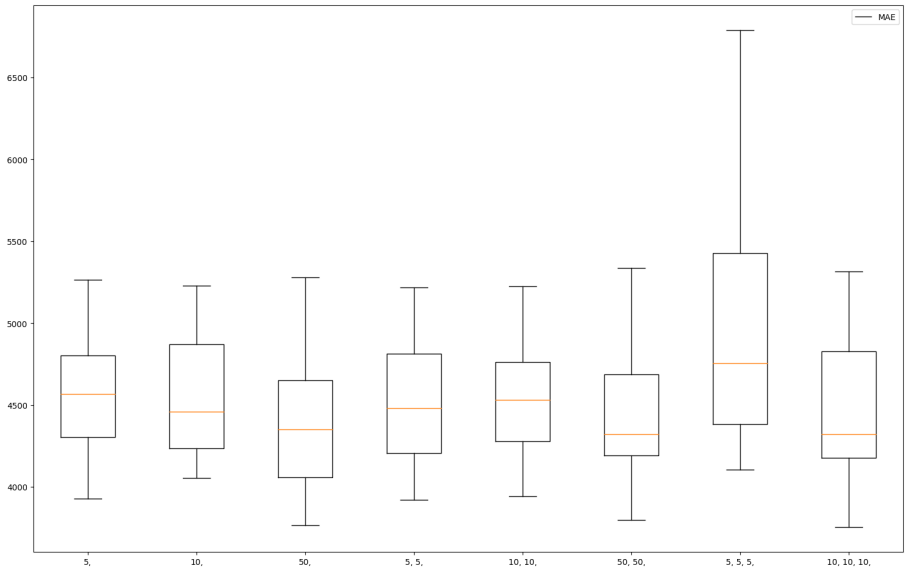
(d) Vrijeme trajanja treniranja modela



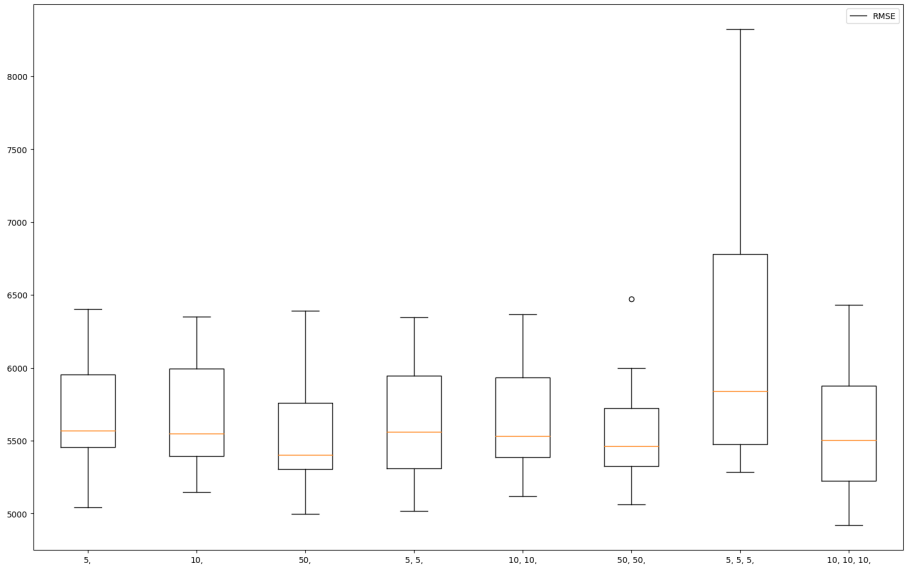
(e) PKG vrijednosti

Slika 7.26: Vrijednosti modela neuralne mreže za različite parametre neurona i skrivenih slojeva Power consumption of Tetouan city 3 zone baze podataka

Nastavak s prijašnje stranice



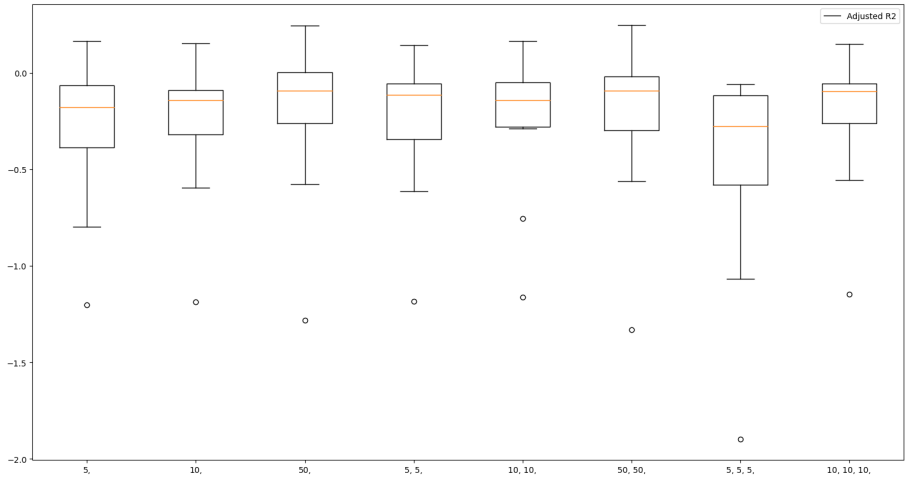
(a) MAE vrijednosti



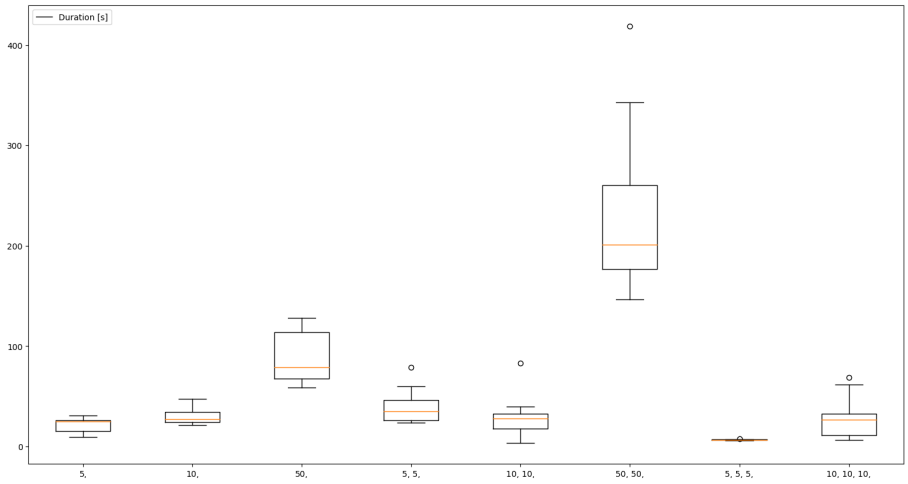
(b) RMSE vrijednosti

Slika 7.27: Vrijednosti modela neuralne mreže za različite parametre neurona i skrivenih slojeva Power consumption of Tetouan city sa 3 tražene vrijednosti baze podataka

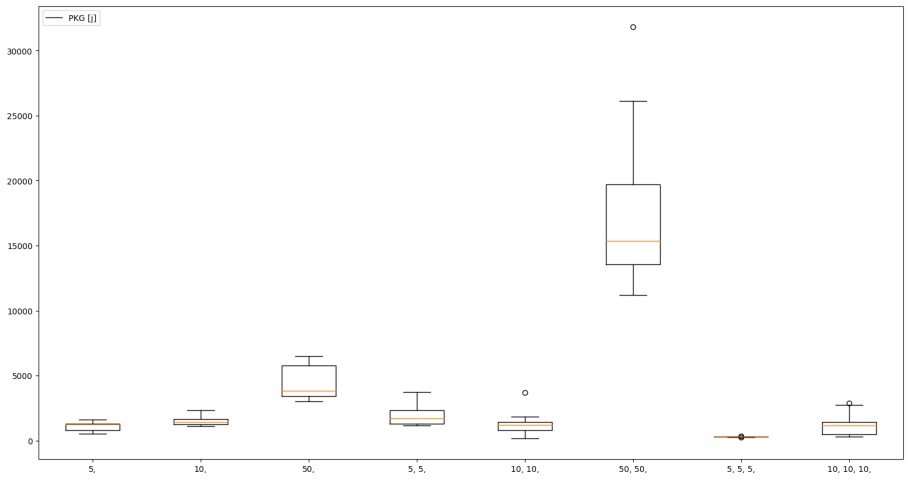
(Nastavlja se na sljedećoj stranici)



(c) Adjusted R^2 vrijednosti



(d) Vrijeme trajanja treniranja modela



(e) PKG vrijednosti

Slika 7.27: Vrijednosti modela neuralne mreže za različite parametre neurona i skrivenih slojeva Power consumption of Tetouan city sa 3 tražene vrijednosti baze podataka

(Nastavak s prijašnje stranice)