

Univerza v Ljubljani  
Fakulteta za elektrotehniko

Gregor Černe

**Kratkoročno napovedovanje porabe  
električne energije z uporabo  
mehkih Takagi-Sugeno modelov**

Magistrsko delo

Mentor: prof. dr. Igor Škrjanc

Ljubljana, 2016



## Zahvala

Rad bi se iskreno zahvalil vsem, ki so mi pomagali tekom izdelave magistrskega dela. Predvsem bi se rad zahvalil mentorju, prof. dr. Igorju Škrjancu, ki mi je dal priložnost raziskovati zanimivo in praktično temo. Izpostavil bi tudi asistenta, dr. Dejana Dovžana, ki me je z direktnim in iskrenim pristopom vodil in usmerjal po poti raziskovanja. Hvala tudi Dejanu Banoviču za podrobno razlago energetskega trga ter Lini Boljka in njenim kolegom iz meteorološkega oddelka na Univerzi v Readingu za pomoč pri definiciji sončnega sevanja.

Poseben prostor bi namenil tudi zahvali kolegom avtomatikom, s katerimi smo s skupnimi močmi premostili vse prepreke magistrskega študija: poleg Klemna Grma, Martina Cimermana ter Dore Domanjko bi izpostavil Žiga Petriča za izredno timsko delo ter vzdušje pri vseh seminarskih nalogah, projektih ter tekmovanjih, kolega Davida Čuferja ter Emilijo Simić pa za medsebojno sodelovanje tekom izdelave magistrske naloge, tako praktično kot moralno.

Hvala tudi vsem prijateljem ter družini, ki so me podpirali in stali ob strani na poti do elektrotehniškega znanja, ki mi je navsezadnje prislužilo naziv magister inženir elektrotehnike.



# Vsebina

<b>1</b>	<b>Uvod</b>	<b>5</b>
<b>2</b>	<b>Problem kratkoročnega napovedovanja porabe električne energije</b>	<b>9</b>
2.1	Podatki in obdelovanje podatkov . . . . .	9
2.2	Opis problema . . . . .	19
2.3	Cilji . . . . .	20
<b>3</b>	<b>Teoretični okvir</b>	<b>21</b>
3.1	Identifikacija . . . . .	21
3.1.1	Mehka logika in model Takagi-Sugeno . . . . .	21
3.1.2	Deljenje prostora . . . . .	24
3.1.2.1	Metoda mehkih c-povprečij (FCM) . . . . .	24
3.1.2.2	Rojenje Gustafson-Kessel ter variacije . . . . .	27
3.1.2.3	Rekurzivno rojenje GK . . . . .	36
3.1.2.4	Omejitve kovariančne matrike v GK algoritmu . . . . .	37
3.1.2.5	Eliminacija vplivov nezaželenih vplivov normiranja . . . . .	40
3.1.2.6	Kombinacije računanja pripadnosti . . . . .	46
3.1.3	Gradnja lokalnih modelov . . . . .	49

3.1.3.1	Metoda uteženih najmanjših kvadratov (WLSE) . . . . .	50
3.1.3.2	Rekurzivna metoda uteženih najmanjših kvadratov . . . . .	52
3.1.3.3	Uteženi koeficient pozabljanja linearnega modela . . . . .	54
3.2	Napovedovanje porabe električne energije . . . . .	58
3.2.1	Metoda linearne regresije . . . . .	58
3.2.2	Metoda robustne linearne regresije . . . . .	60
3.2.3	Zlivanje med trenutno ter naslednjo napovedjo . . . . .	62
<b>4</b>	<b>Model napovedovanja . . . . .</b>	<b>65</b>
4.1	Uporabljeni podatki . . . . .	65
4.2	Razdelitev problema in struktura modela . . . . .	66
4.3	Splošni model povprečne porabe . . . . .	68
4.4	Splošni model standardne deviacije porabe . . . . .	70
4.5	Splošni model oblike porabe . . . . .	72
4.6	Modeli praznikov . . . . .	75
4.7	Model zlivanja napovedi . . . . .	76
<b>5</b>	<b>Rezultati in diskusija . . . . .</b>	<b>77</b>
5.1	Način ocenjevanja modela . . . . .	77
5.2	Rezultati modela . . . . .	78
5.3	Primerjava z obstoječimi metodami . . . . .	86
5.4	Diskusija . . . . .	90
5.5	Slabosti algoritma . . . . .	90
5.5.1	Ročna nastavitvev spremenljivke dneva . . . . .	91
5.5.2	Detekcija slabih podatkov le kot del GK algoritma . . . . .	91

5.5.3	Vključitev vzorcev s predstavitvijo ure . . . . .	91
<b>6</b>	<b>Zaključek</b>	<b>93</b>





## Seznam slik

2.1	Izmerjena poraba različnih dni tekom istega tedna stabilnega vremena . . . . .	11
2.2	Graf prikazuje povprečno moč v odvisnosti od temperature. Vsaka točka predstavlja povprečno temperaturo dneva, barva točke pa pomeni leto vzorca . . . . .	13
2.3	Graf povprečne moči v odvisnosti od padavin. Vsaka točka predstavlja en dan . . . . .	13
2.4	Graf profilov sevanja za različne tipične dni. Profil sevanja za sončni dan 9. 6. 2010, pretežno oblačni dan 17. 6. 2010 in deževni dan 21. 6. 2010 . . . . .	14
2.5	Graf povprečne moči v odvisnosti od povprečne hitrosti vetra. Vsaka točka predstavlja en dan . . . . .	15
2.6	Graf prikazuje povprečno moč v odvisnosti od povprečnega sevanja. Vsaka točka predstavlja en dan . . . . .	16
2.7	Graf povprečne porabe vzorca v odvisnosti od prejšnjega dne . . .	17
2.8	Graf standardne deviacije (v grafu STD) porabe vzorca v odvisnosti od prejšnjega dne . . . . .	17

3.1	Koncept mehkega modela za aproksimiranje nelinearne funkcije. Na zgornjem grafu je s črno prikazana nelinearna funkcija, z modrimi elipsami so obkrožena mehka področja, z rdečo pa izhodi linearnih modelov mehkih področij. Spodnji graf prikazuje funkcije pripadnosti rojem. . . . .	23
3.2	Prikaz rezultatov rojenja GK z različnimi razdaljami in rezultajočega modela . . . . .	30
3.3	Prikaz dobrega ter slabega rezultata rojenja GK z izhodnimi razdaljami . . . . .	33
3.4	Prikaz rezultatov rojenja GK z različnimi metodami računanja pripadnosti . . . . .	35
3.5	Graf pripadnostnih funkcij kvadratne funkcije po uporabljenem osnovnem GK rojenju (poglavje 3.1.2.2) s tremi roji. S puščicami so označeni vplivi oddaljenih rojev, ki tam ne bi smeli imeti vpliva. . . . .	41
3.6	Na sliki je prikazan princip skrivanja različnih rojev ( $\mathbf{v}_1, \mathbf{v}_2$ ter $\mathbf{v}_3$ ) za rojem s prototipom $\mathbf{v}_j$ pri vzorcu z spremenljivkami particioniranja $\mathbf{x}_p$ . Barva predstavlja vrednost koeficienta skrivanja $\alpha_{ij}$ . . .	42
3.7	Prikazana je primerjava pripadnostnih funkcij modeliranja kvadratne funkcije brez skrivanja (levo) ter s skrivanjem (desno). Pripadnosti so izračunane na primeru iz slike 3.2 . . . . .	43
3.8	Prikaz problema modeliranja globalne spremembe z večimi roji . .	55
3.9	Prikaz problema neutženega koeficienta pozabljanja . . . . .	56
3.10	Primerjava izhodov modelov neutženega koeficienta $\gamma_\theta$ (zgoraj) ter uteženega koeficienta $\gamma_\theta^{(\beta)}$ (spodaj) z različnimi vrednostmi . .	57
3.11	Graf primerjave izhoda modela pri prehodu med napovedmi pred in po zlivanjem . . . . .	63

---

4.1	Blokovni diagram zgrajenega modela. Z vijolično so predstavljeni zgrajeni modeli, z zeleno matematične operacije potrebne za povezavo med modeli, rdeča pa označuje področja že omenjenih podproblemov. . . . .	67
5.1	Histogram napake končne ocene . . . . .	79
5.2	Histogram napake skupne napovedi oblike . . . . .	79
5.3	Histogram napake skupne napovedi povprečne porabe . . . . .	80
5.4	Profil porabe na 17. 2. 2012 ter napoved modela kot primer zelo dobre napovedi porabe . . . . .	85
5.5	Profil porabe na 14. 2. 2012 ter napoved modela kot primer zelo slabe napovedi porabe . . . . .	85
5.6	Primerjava histogramov napake skupne porabe po metodah LR (levo) ter RLR (desno) . . . . .	88
5.7	Primerjava histogramov napake povprečne porabe po metodah LR (levo) ter RLR (desno) . . . . .	89
5.8	Primerjava histogramov napake oblike porabe po metodah LR (levo) ter RLR (desno) . . . . .	89
5.9	Primer napačnega podatka - izmerjeno na dan 10.3.2010. . . . .	92



## Seznam tabel

1	Veličine in simboli . . . . .	xv
2.1	Tabela podatkov ene ure . . . . .	10
5.1	Tabela prikazuje napako MAPD [%] različnih delov modela . . . .	78
5.2	Primerjava napake napovedi MAPD [%] različnih delov modelov glede na spremenljivko dneva D . . . . .	81
5.3	Primerjava napake napovedi MAPD [%] različnih delov modelov glede na dan v tednu . . . . .	82
5.4	Primerjava napake napovedi različnih delov modelov glede na letni čas . . . . .	83
5.5	Primerjava MAPD pri napovedovanju povprečja porabe glede na število rojev ter koeficienta pozabljanja . . . . .	83
5.6	Primerjava MAPD pri napovedovanju oblike porabe glede na šte- vilo rojev . . . . .	84
5.7	Primerjava napak MAPD različnih metod napovedovanja porabe .	86
5.8	Primerjava napak napovedi MAPD različnih metod pri različnih komponentah porabe . . . . .	86
5.9	Primerjava napak napovedi MAPD različnih metod pri različnih dnevih v tednu . . . . .	87

5.10 Primerjava napak napovedi MAPD različnih metod v različnih letnih časih . . . . .	88
--	----

## Seznam uporabljenih simbolov

V pričujočem zaključnem delu so uporabljeni naslednje veličine in simboli:

Veličina / oznaka	
Ime	Simbol
Pripadnost	$\mu$
Razdalja	$d$
Prostor particioniranja	$A_p$
Vektor spremenljivk v prostoru particioniraja	$\mathbf{x}_p$
Regresijski vektor	$\mathbf{x}$
Kovariančna matrika	$\mathbf{F}$
P-norma matrike	$\mathbf{A}$
Izmerjen izhod $k$ -tega vzorca	$\mathbf{y}(k)$
Vektor izhoda modela	$\hat{\mathbf{y}}$
Vektor parametrov linearnega modela	$\boldsymbol{\theta}$
Regresijska kovariančna matrika	$\mathbf{P}$
Prototip roja	$\mathbf{v}$

Tabela 1: Veličine in simboli

Pri čemer so vektorji in matrike napisani s poudarjeno pisavo. Natančnejši pomen simbolov in njihovih indeksov je razviden iz ustreznih slik ali pa je pojasnjen v spremljajočem besedilu, kjer je simbol uporabljen.





## Povzetek

Električni distributerji morajo za naslednji dan čim bolje napovedati porabo njihovih strank oziroma naročnikov električne energije, saj tako lahko vnaprej kupijo električno energijo po nižji ceni kot bi jo v trenutku porabe. Problem nastane v primeru napačne napovedi, saj to prinese dodatne stroške bodisi z nakupom dražje manjkajoče energije v trenutku porabe bodisi s kaznimi energetskega regulatorja zaradi preobremenitve omrežja. Opisana situacija je problem kratkoročnega napovedovanja porabe električne energije, ki so se ga z različnimi metodami lotili že mnogi znanstveniki ter gospodarske družbe.

Magistrsko delo se problema kratkoročnega napovedovanja porabe električne energije loti z uporabo najnovejših odkritij na področju modeliranja procesov z mehкими modeli. Ti se v literaturi za napovedovanje porabe električne energije že uporabljajo, vendar v pretežno neprilagodljivih konfiguracijah (ročno vnaprej določeni roji), kjer za dober model potrebujemo dobro znanje o samih podatkih ter dolgotrajen proces uglaševanja parametrov.

Magistrsko delo tako razišče možnosti uporabe adaptivnih mehkih modelov v namen poenostavitve načrtovanja modela napovedovanja porabe električne energije. Tekom dela so bile razvite tudi nadgradnje za particioniranje prostora, katere bi lahko bile uporabne tudi na drugih področjih uporabe mehkih modelov. Rezultati razvitega modela so se izkazali kot dobri in obetajoči za nadaljnje raziskovanje, predvsem se pa je model izkazal v primerjavi z že obstoječimi metodami.

**Ključne besede:** mehko modeliranje, napovedovanje, poraba električne energije, rojenje Gustafson-Kessel, metoda uteženih najmanjših kvadratov, skrivanje rojev, identifikacija

## Abstract

Because electricity distributors can pre-purchase electric energy in advance at the lower prices than at the time of consumption, they want to forecast their customer's demand for the following day in order to pre-purchase electric energy according to that forecast. A problem arises in the case of an inaccurate forecast, as it brings additional expenses either by buying more expensive energy at the time of consumption either by energy regulator's penalties for overloading the electric network. The described situation is the problem of short-term forecasting of electric energy consumption, which was already addressed by many scientists and corporations using various methods.

Master's thesis tackles the mentioned problem using the latest discoveries in the field of fuzzy modeling. In the literature, fuzzy modeling has been already used for forecasting electric energy consumption, but in the predominantly non-flexible configurations (for example manually predefined clusters), where a good knowledge of the given data and a time-consuming tuning of parameters is required.

Therefore, Master's thesis investigates the possibility of using adaptive fuzzy models for the purpose of simplifying modeling stage in electric energy consumption forecasting. During the research, we also developed upgrades for space partitioning, which could be effective also in other fields where fuzzy modeling is used. The results of the developed model proved to be good and promising for further research, especially when compared with existing methods.

**Key words:** fuzzy modeling, forecasting, electric energy consumption, Gustafson-Kessel clustering, weighted least mean square method, cluster hiding, identification

# 1 Uvod

Na prostem trgu električne energije kot posredniki med elektrarnami ter potrošniki delujejo distributerji električne energije – od elektrarn kupujejo električno energijo na debelo ter jo prodajo na drobno svojim strankam (potrošnikom). Zaradi specifičnosti blaga (proizvedena električna energija mora biti v vsakem trenutku enaka porabljeni) želijo elektrarne čim hitreje vedeti, koliko energije morajo proizvesti, saj se lahko na napovedano spremembo v porabi pripravijo. Zato ponujajo cenejšo električno energijo, če se jo kupi vnaprej – hitreje kot jo zakupi, cenejša je energija.

Zato želijo distributerji kupiti energijo čim hitreje, za kar pa potrebujejo napovedi o porabi njihovih strank v določenem trenutku. Ker pa je poraba odvisna od mnogih dejavnikov, distributer ne more napovedati natančne porabe za dalj časa vnaprej. Zato napovedi razdelijo na dolgoročne (celoletna poraba), srednjerodne (poraba za časovno obdobje meseca) ter kratkoročne (poraba preko dneva). Magistrska naloga se loteva problema kratkoročnega napovedovanja porabe električne energije, kar pomeni napoved porabe za naslednji dan.

Problem kratkoročnega napovedovanja porabe električne energije je možno rešiti na mnogo različnih načinov z različno uspešnostjo. V literaturi je pogosto omenjen model ARIMA [2][3], ki napoveduje z avtoregresijskim integrirnim drsečim povprečjem. Za napovedovanje porabe so uporabljeni tudi linearni modeli, identificirani z robustno linearno regresijo (metoda RLR) [1], ki za razliko od linearne regresije omeji vpliv slabih vzorcev (ang. outliers). Metoda napovedovanja s pomočjo podobnih dni [4] v zgodovini podatkov išče temu dnevu najbolj

podobne dni, na podlagi katerih nato sestavi napoved. Za napovedovanje so bile uspešno uporabljene tudi umetne nevronske mreže [5], ki v znanstvenih krogih že nekaj časa uživajo precejšnje zanimanje in pozornost. Pogosto se uporabljajo tudi mehki modeli, na katere se bo osredotočilo tudi magistrsko delo.

Od odkritja mehkega modeliranja z modeli Takagi-Sugeno (v nadaljevanju model TS) leta 1985 [6] je v literaturi omenjenih mnogo algoritmov in modelov, ki se kot osnove poslužujejo omenjenega modela. Hitro se je za določanje rojev začela uporabljati iterativna metoda mehkega rojenja s c-povprečji FCM [7], ki na podlagi Evklidove razdalje med vzorci najde suboptimalno razporeditev vnaprej določenega števila rojev hipersferične oblike. To je kmalu nadomestila metoda Gustafsson-Kessel [8], ki namesto hipersferičnih oblik rojev metode FCM išče roje v obliki hiperelips. Ti metodi sta za namen sprotne identifikacije časovno spremenljivih sistemov dobili rekurzivne različice [9][10][11], kasneje sta bili nadgrajeni z metodami avtomatskega določevanja števila rojev [12][13], nato so bile razvite še samorazvijajoče različice [14][15]. Model TS se lahko realizira tudi z nevronskimi mrežami [16].

Dosedanje raziskave uporabe mehkih modelov na področju kratkoročnega napovedovanja porabe električne energije so bile uspešne, vendar imajo razviti modeli določene omejitve: napovedovanje je strogo ločeno po mesecih in ne obstaja nobena povezava med njimi [17], ali pa je funkcija mehčanja vnaprej določena in toga (trikotna), kar pomeni, da mora načrtovalec sam določiti meje brez računalniške podpore [18].

Vse omenjene metode napovedovanja z mehкими modelmi so toge v smislu določanja področij mehčanja – ali moramo ročno nastaviti funkcije mehčanja ali trdo ločimo obdobja napovedovanja itd. Med tem pa so raziskave na področju mehkih modelov že napredovale do točke, ko algoritem večino parametrov in funkcij določi avtomatično.

V delu se bomo posvetili implementiranju najnovejših odkritij na področju mehkih modelov za kratkoročno napovedovanje porabe električne energije, s ka-

terimi želimo izboljšati trenutno uporabljene metode. To predvidevamo zaradi vpeljave sprotnih metod prilagajanja mehkih modelov, ki so se sposobni prilagajati na časovno spremenljiv sistem, kar poraba električne energije absolutno je. Zaradi modularne narave modela TS se lahko nadejamo tudi dobrega modeliranja spreminjanja porabe glede na letni čas.

V poglavju 2 bomo omenjeni problem podrobno razložili, hkrati pa bomo predstavili razpoložljive podatke ter cilje naloge. Tekom poglavja 3 bo razložena teorija, na kateri sloni razviti model napovedovanja. Konstrukcija razvitega modela je opisana v poglavju 4, rezultati in diskusija razvitega modela pa so navedeni v poglavju 5. Celotno delo je na koncu povzeto v zaključku.





## 2 Problem kratkoročnega napovedovanja porabe električne energije

V poglavju bomo najprej predstavili s kakšnimi podatki razpolagamo (podpoglavje 2.1), nato bomo tekom podpoglavja 2.2 razložili problem kratkoročnega napovedovanja ter in se z njem srečamo. Zaključili bomo s predstavitevijo ciljev magistrskega dela v podpoglavju 2.3.

### 2.1 Podatki in obdelovanje podatkov

Najprej bodo opisani podatki, ki so bili uporabljeni v magistrskem delu, na koncu pa še uporabljene metode obdelave podatkov, ki so nam koristile pri izgradnji modelov.

V nalogi raziskujemo podatke porabe električne energije ter meritve vremenskih parametrov za JZ Slovenijo od leta 2010 do vključno 2013, vendar za leto 2013 podatki o porabi niso podani. Meritve so pridobljene za vsako uro v dnevu. V tabeli 2.1 so zbrani podatki ene ure.

V našem primeru definiramo vzorec kot skupek meritev za en dan, tako da imamo v vsakem vzorcu 24 meritev za vsako spremenljivko in tip dneva (praznik, mesec in podobno).

Izhodna spremenljivka je urna dnevna poraba električne energije. Primeri so prikazani na grafih 2.1. Profili porabe se spreminjajo glede na letni čas, dan v tednu, praznik, kako blizu je praznik trenutnemu dnevu ipd.

Ime	Simbol	Enota ali nabor vrednosti
Porabljena električna energija	E	$MWh$
Povprečna temperatura	T	$^{\circ}C$
Povprečno sončno sevanje	$\Gamma$	$\frac{W}{m^2}$
Skupna količina padavin	R	$mm$
Povprečna hitrost vetra	W	$\frac{m}{s}$
Ura v dnevu (7 pomeni 6:00 do 7:00)	H	1 – 24
Dan v tednu (1 je ponedeljek)	F	1 – 7
Dan v mesecu	G	1 – 31
Mesec	M	1 – 12
Leto	Y	2010 – 2012
Praznik (1 je praznik, 0 pa ostalo)	P	0 ali 1

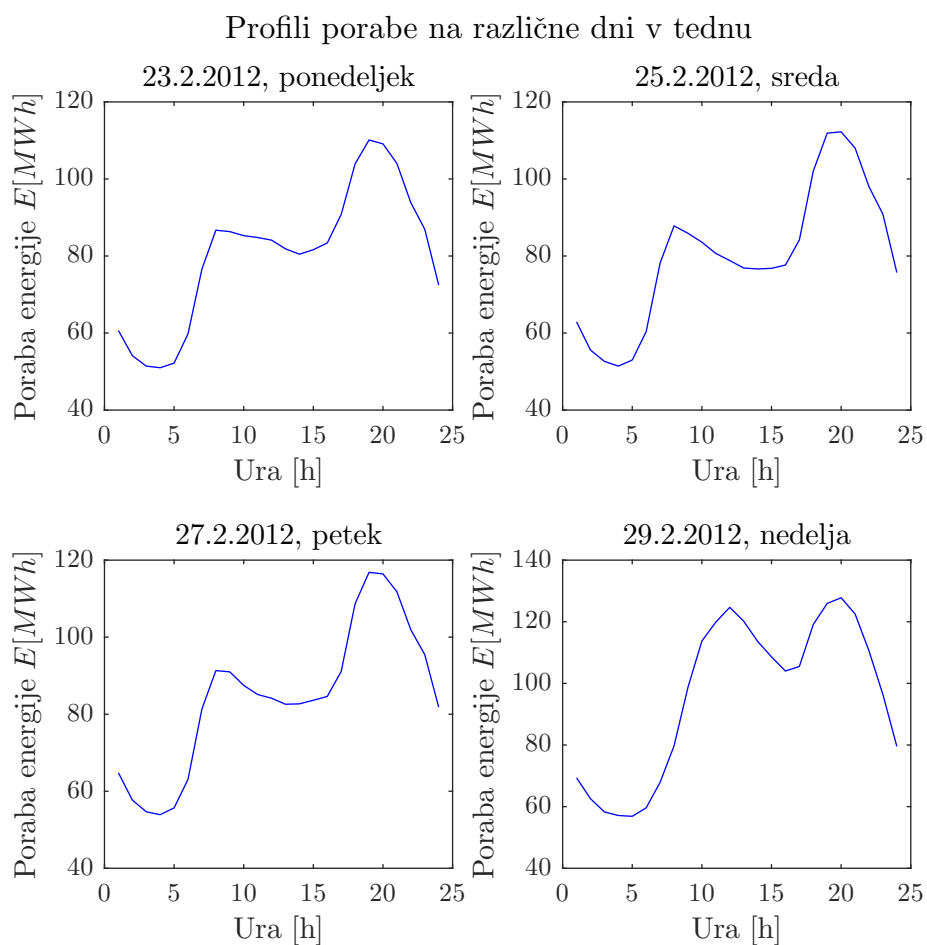
Tabela 2.1: Tabela podatkov ene ure

Na grafih 2.1 vidimo precejšnjo razliko v porabi med vikendi ter porabo med delavniki. Tudi med delavniki opazimo manjše razlike, predvsem zaradi vpliva vikenda na porabo v ponedeljek ter petek – to se predvsem vidi pri "ujetem delavniku", ko je delavnik med vikendom ter praznikom, saj si ponavadi ljudje tisti dan vzamejo dopust. Zato vpeljemo spremenljivko dneva  $D$ , s katerim želimo izraziti vpliv datuma na porabo.

Najprej z enačbo (2.1) definiramo spremenljivko posebnih dni  $D^*$ , ki ima vrednost 1 v primeru praznika ali vikenda, ter 0 pri vseh ostalih dneh. Z  $D^*$  sedaj lahko ločimo posebne dni (vikendi in prazniki) od ostalih dni.

$$D^*(k) = \begin{cases} 1, & F(k) \in \{6, 7\} \vee P(k) = 1 \\ 0, & \text{sicer} \end{cases} \quad (2.1)$$

Omenjena spremenljivka posebnih dni pa ne zajame vpliva vikenda na delav-



Slika 2.1: Izmerjena poraba različnih dni tekom istega tedna stabilnega vremena

nike (predvsem ponedeljek ter petek). Zato želimo, da na spremenljivko dneva  $D$  vpliva tudi vrednost spremenljivke posebnih dni  $D^*$  prejšnjega ter naslednjega dneva. Pri iskanju enačbe za  $D$  želimo pri analizi podatkov porabe upoštevati naslednje ugotovitve:

- Petkova poraba je bolj podobna porabi tekom vikenda kot ponedeljkova.
- Ujeti delavnik ima porabo izrazito bolj podobno vikendu kot delavniku.
- Spremenljivka posebnih dni  $D^*$  ima lahko tudi druge vrednosti kot 0 in 1,

ker nam omogoči fleksibilnost pri spreminjanju definicije  $D^*$ .

Z upoštevanjem omenjenih ugotovitev smo razvili enačbo (2.2) za izračun  $D$ . Če je  $k$ -ti dan poseben, potem  $D(k) = 1$ , drugače pa ga izračunamo glede na  $D^*$  naslednjega ter prejšnjega dne. Enačba ima 3 prispevke – prispevka naslednjega in prejšnjega dne ter kombinacija obeh. Zadnji prispevek smo dodali za primer ujetega delavnika, ko je  $D^*$  tako za prejšnji kot naslednji dan enak 1. S koeficienti  $g_+$ ,  $g_-$  ter  $g_{\pm}$  nastavljamo vpliv sosednjih dni, ki smo jih tekom magistrskega dela nastavili s poskušanjem ( $g_- = 0.125$ ,  $g_+ = 0.15$ ,  $g_{\pm} = 0.5$ ). Vsota koeficientov mora biti manjša kot 1 (2.3), saj ne obstaja dan, ki bi bil bolj poseben kot so vikendi oziroma prazniki.

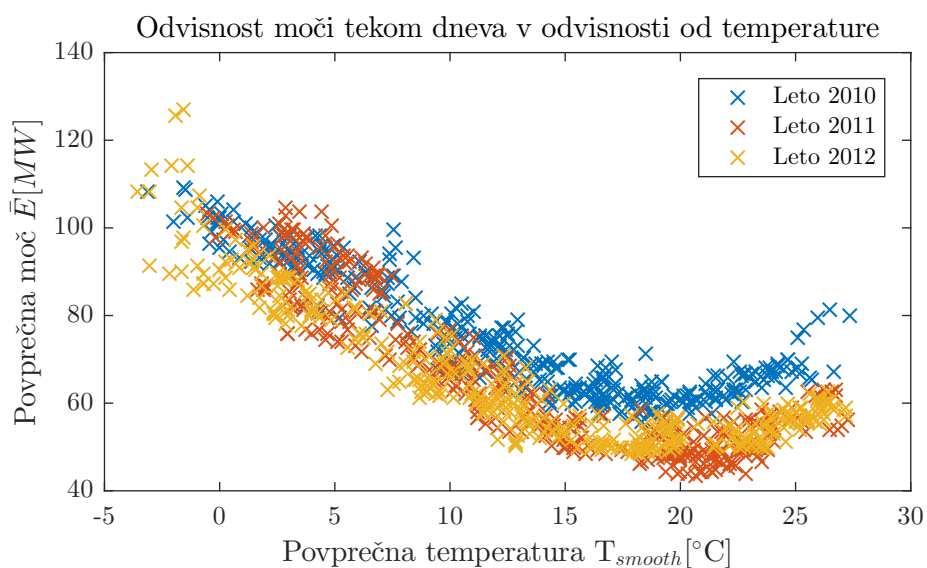
$$D(k) = \max \left[ g_+ D^*(k+1) + g_- D^*(k-1) + g_{\pm} D^*(k-1) D^*(k+1), \quad D^*(k) \right] \quad (2.2)$$

$$g_+ + g_- + g_{\pm} < 1 \quad (2.3)$$

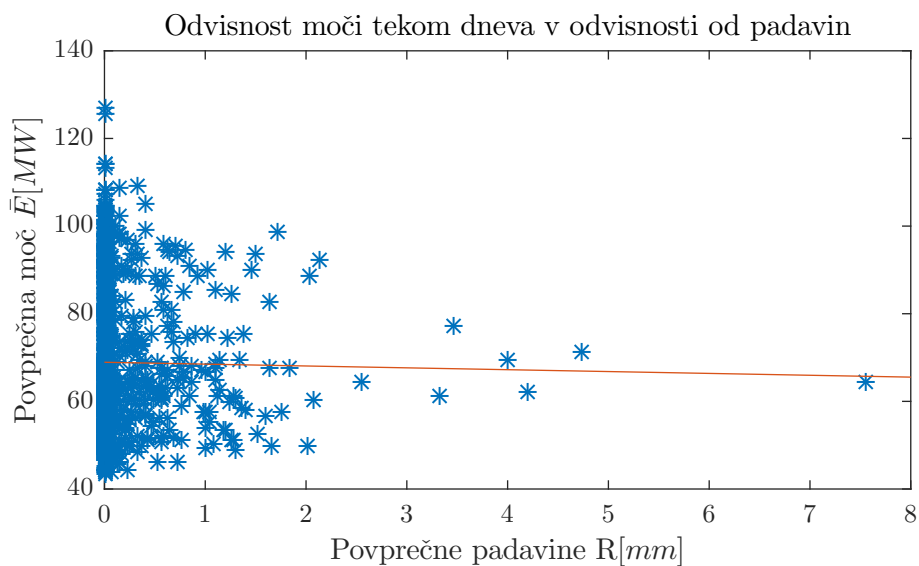
Ker precejšen del porabe predstavlja ogrevanje, je poraba odvisna od temperature, kar se dobro vidi na sliki 2.2. Ta odvisnost ni hipna, zato definiramo  $T_{smooth}$ , ki predstavlja izhod diskretnega linearnega modela 1. reda z vhodno spremenljivko izmerjene temperature  $T$ . Diskretni korak je ena ura. Koeficient  $\xi$  določimo z iskanjem minimuma pogreška izhoda mehkega modela pri vhodnih vektorjih  $\mathbf{x}_p(k) = \mathbf{x}(k) = [\overline{T}_{smooth}(k), 1]^T$  ter izhodom  $y(k) = \overline{E}(k)$ . Minimum smo iskali z ročnim nastavljanjem koeficienta  $\xi$ . Za  $\xi$  smo dobili vrednost 0.98 ter posledično časovno konstanto modela  $\tau = 50$  ur. Časovna konstanta je velika, kar nakazuje močan vpliv specifične toplote tal in objektov na ogrevanje.

$$T_{smooth}(k) = \xi T_{smooth}(k-1) + (1 - \xi) T(k), \quad \xi = 0.98 \quad (2.4)$$

Na sliki 2.2 opazimo tudi časovno spreminjanje procesa porabe, saj je tekom leta 2010 višja kot tekom let 2011 ter 2012.



Slika 2.2: Graf prikazuje povprečno moč v odvisnosti od temperature. Vsaka točka predstavlja povprečno temperaturo dneva, barva točke pa pomeni leto vzorca

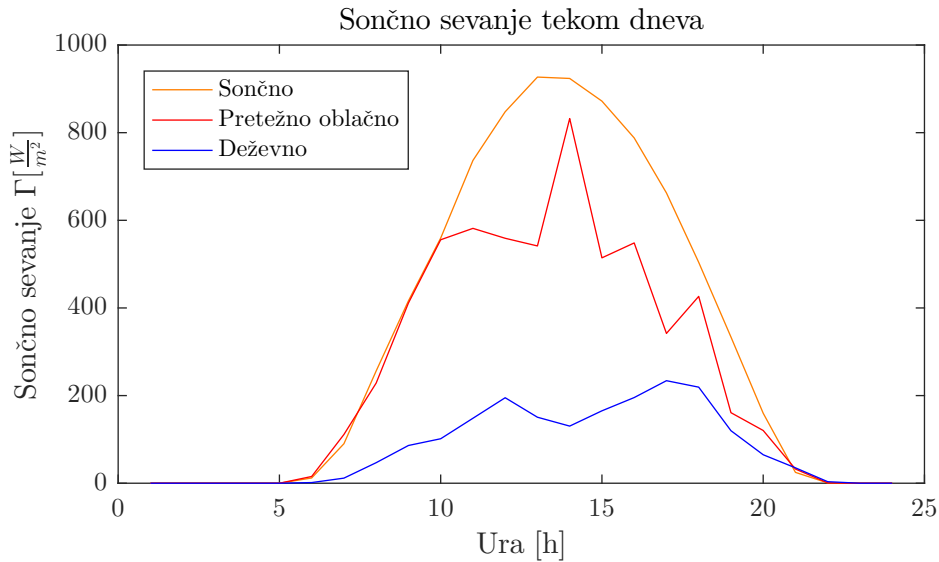


Slika 2.3: Graf povprečne moči v odvisnosti od padavin. Vsaka točka predstavlja en dan

Pri analizi podatkov deževnih dni smo opazili, da se poraba pri močnem deževju ne razlikuje bistveno od porabe pri šibkejšem (slika 2.3) – razlika se pa pokaže med deževnimi dnevi in dnevi brez dežja. Razlog smo našli v tem, da sam dež ne vpliva toliko na ogrevanje hiš, ampak na dejstvo ali se bodo ljudje raje zadrževali v hišah v slabem vremenu (in temu primerno bo poraba večja), ali pa v lepem šli na sprehod, izlet ipd.

Zato meritve dežja v eni uri navzgor omejimo na  $R_{max} = \frac{10}{24} \frac{mm}{h}$  (vrednost  $\frac{10}{24} \frac{mm}{h}$  je nastavljena tako, da je največja vsota omejenih padavin v enem dnevu  $10mm$ ), kar ponazarja enačba (2.5). Lahko bi vzeli le informacijo o tem, ali da je deževalo ali ne (0 ali 1), vendar metode rojenja (poglavje 3.1.2) ne operirajo dobro z binarnimi spremenljivkami.

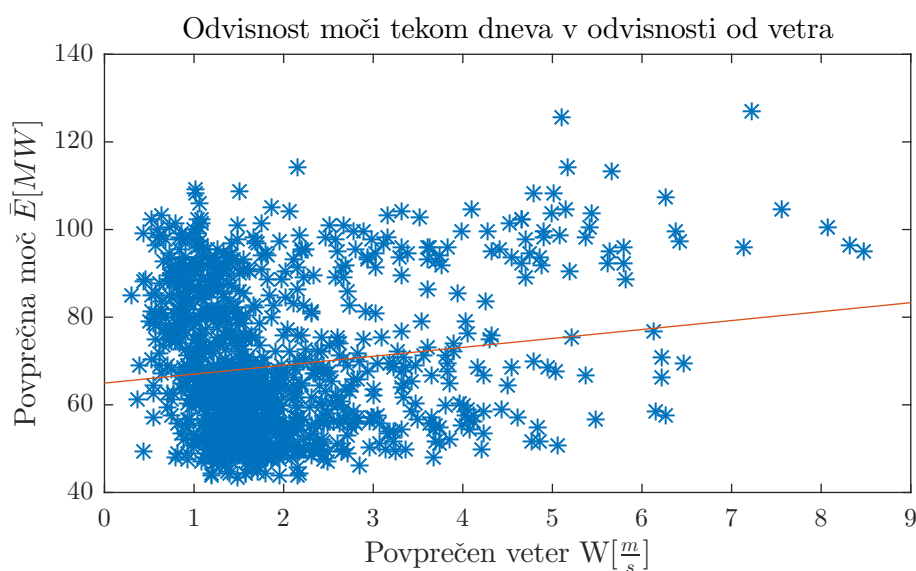
$$R(k)_l = \begin{cases} R(k)_l & R(k)_l \leq R_{max} \\ R_{max} & R(k)_l > R_{max} \end{cases} \quad (2.5)$$



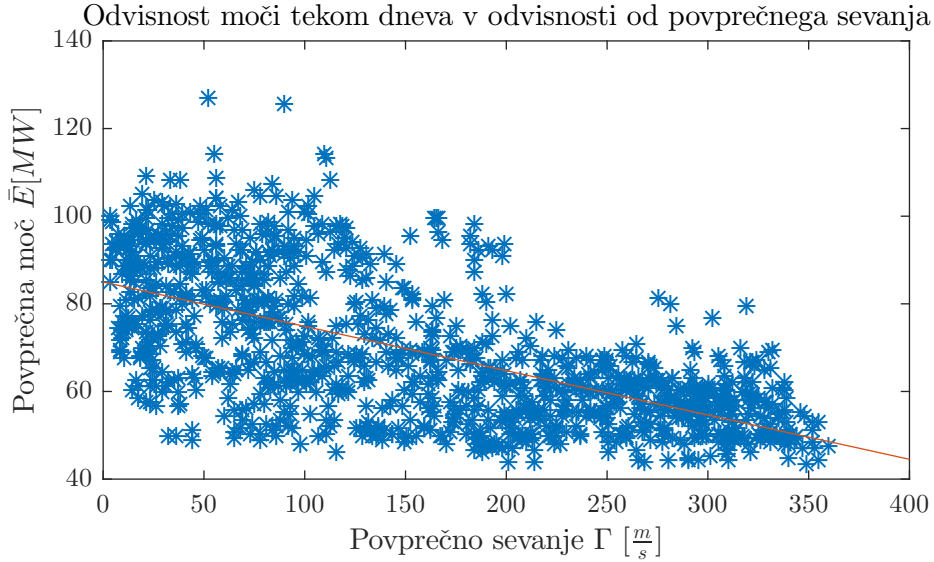
Slika 2.4: Graf profilov sevanja za različne tipične dni. Profil sevanja za sončni dan 9. 6. 2010, pretežno oblačni dan 17. 6. 2010 in deževni dan 21. 6. 2010

Na sliki 2.4 so prikazani profili sevanja za različne poletne dni. Vidimo lahko, da za deževno ter oblačno vreme sevanje močno niha, vendar ta nihanja ne vplivajo na porabo, pač pa nanjo vpliva povprečna vrednost sevanja v določenem dnevu. S tem razlogom sevanje v modelu upoštevamo s povprečno vrednostjo za tisti dan. Podobno lahko opazimo pri profilih padavin ter hitrosti vetra, zato tudi pri teh vremenskih podatkih vzamemo povprečje vrednosti.

Na slikah 2.5 ter 2.6 lahko kljub precejšnji razpršenosti povprečne porabe glede na povprečje vetra oz. sevanja opazimo šibko povezavo, zato povprečja vseeno upoštevamo v modelu.



Slika 2.5: Graf povprečne moči v odvisnosti od povprečne hitrosti vetra. Vsaka točka predstavlja en dan



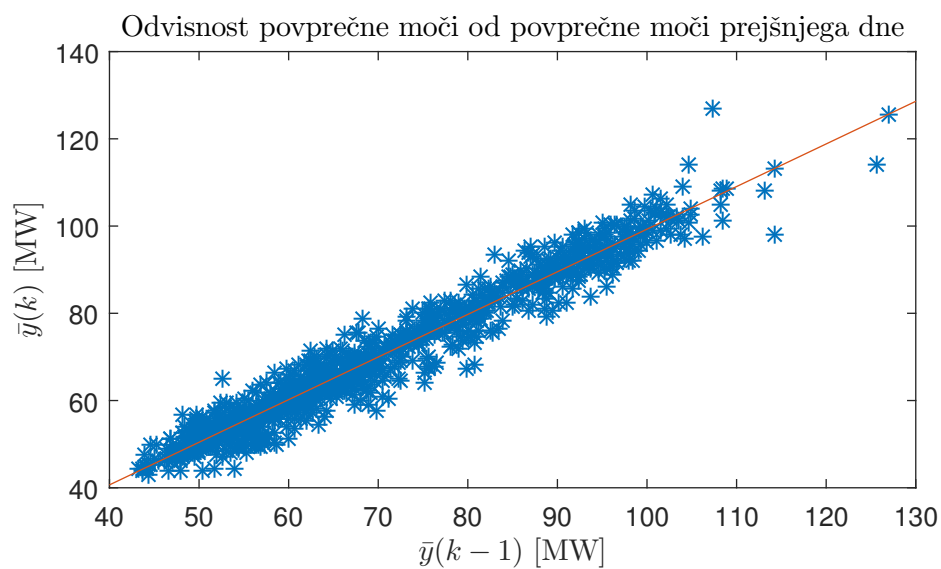
Slika 2.6: Graf prikazuje povprečno moč v odvisnosti od povprečnega sevanja. Vsaka točka predstavlja en dan

Oglejmo si še odvisnost povprečne porabe  $\bar{y}$  (enačba (2.6)) ter standardne deviacije  $\sigma$  (enačba (2.7)) porabe dneva glede vrednosti za prejšnji dan (sliki 2.7 ter 2.8). Vidimo lahko, da sta povprečje in standardna deviacija trenutnega dneva korelirani z vrednostjo prejšnjega dne. To nam omogoči napovedovanje porabe na podlagi porabe prejšnjih dni.

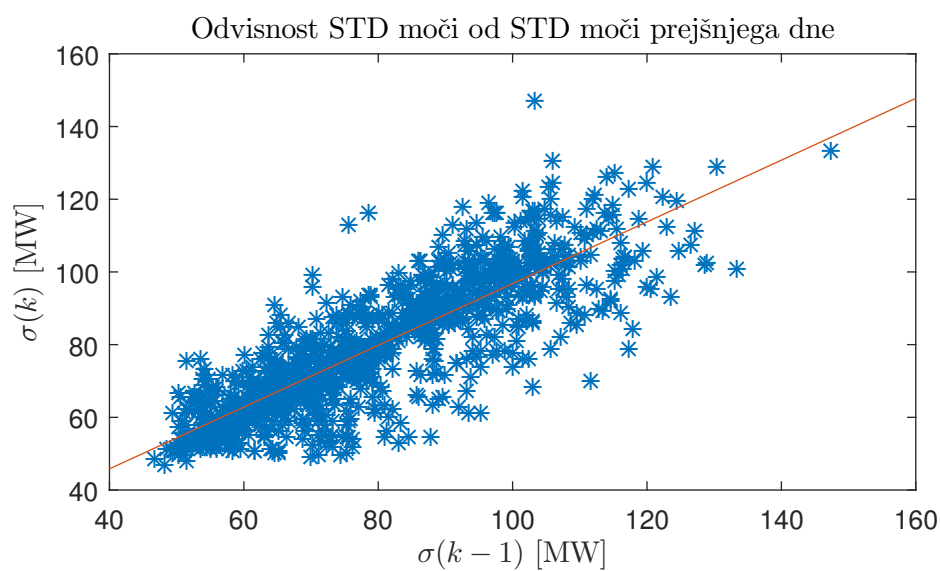
$$\bar{y}(k) = \frac{1}{24} \sum_{l=1}^{24} E_l(k) \quad (2.6)$$

$$\sigma(k) = \sqrt{\frac{1}{24} \sum_{l=1}^{24} (E_l(k) - \bar{y}(k))^2} \quad (2.7)$$





Slika 2.7: Graf povprečne porabe vzorca v odvisnosti od prejšnjega dne



Slika 2.8: Graf standardne deviacije (v grafu STD) porabe vzorca v odvisnosti od prejšnjega dne

V nadaljevanju smo meritve in izpeljane spremenljivke (padavine, povprečna poraba,  $T_{smooth}$  itd.) združili v prostor podatkov. Ker v algoritmu uporabimo tudi Evklidsko razdaljo, različni velikostni razredi podatkov ( $10^0$  pri spremenljivki dneva,  $10^1$  pri temperaturah) lahko pokvariyo kvaliteto deljenja tega prostora. Za primer podamo tri vzorce: vzorec srede ( $D_1 = 0$ ) pri  $T_1 = 20^\circ\text{C}$ , nato vzorec nedelje ( $D_2 = 1$ ) pri enaki temperaturi  $T_2 = T_1 = 20^\circ\text{C}$  in še vzorec srede ( $D_3 = 0$ ) pri  $T_3 = 22^\circ\text{C}$ . Če računamo Evklidsko razdaljo med vzorci, vidimo, da sta si najbližje vzorca 1 ter 3 – kar pa ni res v primerjavi porabe, saj se ta bolj spremeni od delavnika do vikenda kot pri spremembi temperature za par stopinj.

Zato vse podatke normiramo preden jih uporabimo kot vhodne spremenljivke v modelu. Normiramo jih z enačbami (2.8), (2.9) ter (2.10).

$$y(k) = \frac{x(k) - \bar{x}(k)}{\sigma(k)} \quad (2.8)$$

$$\bar{x}(k) = \frac{1}{N} \sum_{k=1}^N x(k) \quad (2.9)$$

$$\sigma(k) = \sqrt{\frac{1}{N} \sum_{k=1}^N (x(k) - \bar{x}(k))^2} \quad , \quad (2.10)$$

kjer je  $x(k)$  podatek  $k$ -tega vzorca pred normiranjem,  $N$  število vzorcev,  $y(k)$  pa normiran podatek, ki ga uporabimo kot spremenljivko v opisanih algoritmihi.

## 2.2 Opis problema

V električnem omrežju mora biti v vsakem trenutku poraba električne energije enaka proizvedeni, kar pomeni, da mora biti vsa energija iz elektrarn in ostalih virov energije enaka porabljeni energiji potrošnikov (kot so gospodinjstva, tovarne, poslovni objekti in ostali). Energijo v sistemu narekujejo porabniki, tako da je na plečih elektrarn, da sledijo trendom porabe in temu primerno proizvedejo zadostno količino električne energije.

Od uvedbe prostega trga na slovenskem energetskega trgu se je veriga dobave električne energije podaljšala za vmesni člen med elektrarnami ter potrošniki – z električnimi distributerji. Ti od elektrarn na prostem trgu kupujejo energijo, za katero računajo, da jo bodo njihovi naročniki porabili v določenem trenutku. S tem razbremenijo elektrarne napovedovanja ter dogovarjanja z mnogimi porabniki, te pa sedaj plačajo le eno položnico distributerju. Distributerji lahko kupujejo električno energijo tudi v tujini, s čimer omogočijo tujim elektrarnam vstop na slovenski trg. [19]

Ker pa imajo nekatere elektrarne določene fizične omejitve, ki jim preprečujejo nenadne spremembe proizvedene energije (primer Nuklearna elektrarna Krško, kjer je nenadna sprememba proizvedene energije nemogoča), želijo vedeti čim prej, koliko el. energije morajo proizvesti v določenem časovnem obdobju, da se lahko pripravijo na nenadne spremembe v porabi. To vzpodbudijo z nizkimi cenami vnaprej kupljene električne energije ter z višjimi v primeru nakupa v trenutku porabe. Poleg tega regulator finančno sankcionira dobavitelja, če kupi preveč energije, saj s tem dodatno obremenjuje omrežje.

Tako dobavitelj plača najmanj, če vnaprej napove porabo el. energije svojih potrošnikov popolnoma točno in jo v takih količinah tudi vnaprej kupi. Z vsako napako v napovedi (prevelika ali premajhna napoved) pa mora za zagotovitev dobave električne energije svojim potrošnikom plačati več kot bi lahko v primeru točne napovedi, posledično pa ima dobavitelj manj dobička.

Problem pri napovedovanju porabe električne energije nastane zaradi sistema porabe, saj je zelo odvisen od mnogih dejavnikov – od vremena, gospodarskih nihanj, novih uporabnikov, prenove stanovanj, enkratnih dogodkov in tako naprej. Že samo vreme s trenutno tehnologijo lahko napovemo le nekaj dni vnaprej, tako da je posledično tudi porabo (razen grobe ocene) nemogoče točno napovedati za več kot dan ali dva vnaprej.

Zaradi omenjenega razloga elektrarne ponujajo odkup električne energije v različnih konfiguracijah (pasovni, trapezni ter urni odkup), v različnih obdobjih (letni oz. dolgoročni, mesečni oz. srednjeročni ter dnevni/urni odkup oz. kratkoročni) ter različno glede na čas nakupa (leto, teden, dan vnaprej ali pa v trenutku porabe), vsaka s svojimi specifičnimi lastnostmi ter cenami. Distributer kombinira različne načine odkupa, da s kar se da nizko možno ceno nakupa električne energije zadosti povpraševanju njegovih naročnikov, pri čemer pazi, da ne zakupi ne preveč ne premalo energije.

Magistrsko delo se loteva problema kratkoročnega napovedovanja porabe električne energije, kar pomeni, da bomo na podlagi vremenske napovedi za naslednji dan ter na podlagi preteklih podatkov o porabi in vremenu napovedali dnevni potek porabe slovenskega distributerja za naslednji dan. Dnevni potek sestavljajo napovedi porabe za vsako uro v dnevu.

## **2.3 Cilji**

Cilj magistrskega dela je uporabiti nove metode na področju mehkih modelov za kratkoročno napovedovanje porabe električne energije ter rezultate napovedi primerjati z ostalimi metodami napovedovanja.

Preveriti želimo tudi sposobnost prilagajanja rekurzivnih mehkih modelov na časovno spremenljiv sistem, kar kratkoročno napovedovanje porabe zagotovo je.

## 3 Teoretični okvir

V tem poglavju je razložena teorija, na kateri je zasnovana izgradnja modela napovedovanja porabe električne energije. V podpoglavju 3.1 je razložen princip mehkega modela ter opis metod za določitev njegovih dveh glavnih komponent – particioniranje oz. deljenje prostora ter načrtovanje lokalnih modelov. Omenjene so tudi vse razvite nadgradnje že obstoječih metod. V naslednjem podpoglavju 3.2 sta predstavljeni metodi, s katerima bomo primerjali razvit algoritem, na koncu pa je predlagana metoda zlivanja med trenutno in naslednjo napovedjo.

### 3.1 Identifikacija

V nadaljevanju bo razložen uporabljen način identifikacije s pomočjo mehke logike. Najprej bo v podpoglavju 3.1.1 razložen koncept mehke logike za gradnjo modelov Takagi-Sugeno. Podpoglavje 3.1.2 je namenjeno razlagi metode rojenja Gustafsson-Kessel za deljenje prostora vključno z njenimi nadgradnjami. Nato pa v podpoglavju 3.1.3 podamo način izračuna lokalnih modelov na delih že razdeljenega prostora.

#### 3.1.1 Mehka logika in model Takagi-Sugeno

Večina realnih procesov je nelinearnih, zato je ponavadi gradnja njihovih matematičnih modelov zahtevna in dolgotrajna. Dobljeni modeli so lahko visokih redov ter težko razumljivi. Če deluje nelinearen proces le v manjšem delu območja delovanja okoli določene delovne točke, hkrati pa je proces v okolici te

delovne točke enostaven, poznamo pristope, s katerimi aproksimiramo delovanje nelinearnih procesov z enostavnejšimi matematičnimi modeli. En bolj razširjen pristopov je linearna aproksimacija. Rezultat linearne aproksimacije je linearni model, ki dobro opiše proces, a le na območju, za katerega je bil načrtan. Z večmodelnim pristopom to težavo dokaj učinkovito rešimo.

Pri večmodelnem pristopu nelinearni model zgradimo z množico linearnih modelov, ki v okolici svoje delovne točke vsak zadovoljivo opiše nelinearni proces. Na tej ideji slonijo tudi mehki modeli, ki omenjeni koncept nadgradijo z uporabo mehke logike. Za modeliranje z uporabo mehke logike se v literaturi največkrat uporablja model Takagi-Sugeno [6] (v nadaljevanju model TS), ki je uporabno in učinkovito orodje za modeliranje in vodenje kompleksnih sistemov.

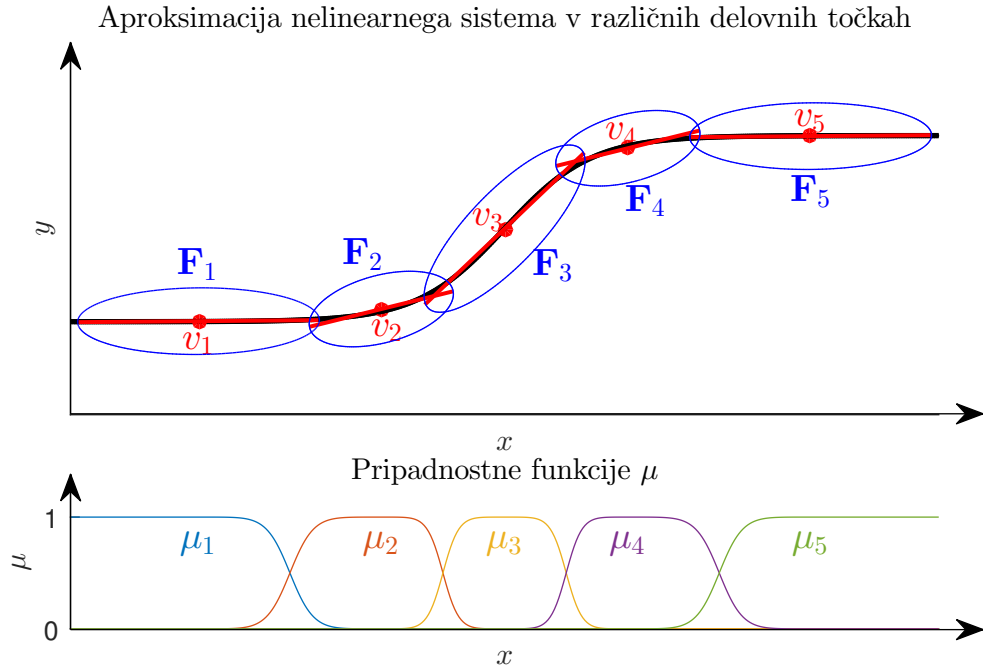
Ideja modelov TS je razdeliti področje delovanja nelinearnega procesa na področja, kjer je za vsakega definiran lokalni model, ki na svojem podpodročju aproksimira nelinearni proces. Z zlitjem izhodov posameznih lokalnih modelov tako aproksimiramo obnašanje nelinearnega sistema. Izhode zlivamo tako, da določimo razdaljo vzorca, za katerega iščemo izhod celotnega mehkega modela, od centra  $i$ -tega podpodročja. Večji vpliv ima izhod lokalnega modela  $i$ -tega mehkega področja, ki je bližje središču.

Primer je prikazan na sliki 3.1. Območje delovanja je razdeljeno na 5 mehkih področij (modra), v katerih aproksimiramo nelinearni proces (črna) z linearnimi modeli, ki jim pravimo tudi lokalni modeli (rdeča). Na spodnjem grafu je prikazano, koliko vpliva ima določen linearni model na izhod celotnega mehkega modela glede na vhod  $\mathbf{x}$  – funkcijam vpliva posameznih rojev pravimo pripadnostne funkcije  $\mu$ .

Izhod modela TS  $\hat{y}$  je utežena vsota izhodov lokalnih modelov  $\hat{y}_i$  (3.1)

$$\hat{y} = \sum_{j=1}^m \mu_i(\mathbf{x}_p) \hat{y}_i(\mathbf{x}) \quad , \quad (3.1)$$

kjer  $\hat{y}$  predstavlja izhod modela TS,  $\mathbf{x} = [1, x_1, \dots, x_{r-1}]^T$  regresijski vektor meritev,  $\mathbf{x}_p = [x_{p_1}, \dots, x_{p_q}]^T$  particijski vektor meritev,  $\mu_i(\mathbf{x}_p)$  vrednost pripa-



Slika 3.1: Koncept mehkega modela za aproksimiranje nelinearne funkcije. Na zgornjem grafu je s črno prikazana nelinearna funkcija, z modrimi elipsami so obkrožena mehka področja, z rdečo pa izhodi linearnih modelov mehkih področij. Spodnji graf prikazuje funkcije pripadnosti rojem.

dnostne funkcije  $i$ -tega roja pri spremenljivkah  $\mathbf{x}_p$  ter  $\hat{y}_i(\mathbf{x})$  izhod  $i$ -tega lokalnega modela pri regresijskem vektorju  $\mathbf{x}$ . Vsi možni regresijski vektorji sestavljajo regresijski prostor  $A_x$  velikosti  $\mathbb{R}^r$ , vsi partijski vektorji pa prostor particioniranja  $A_p$  velikosti  $\mathbb{R}^q$

Za izračun pripadnostnih funkcij  $\mu_i$  je potrebno določiti okolico lokalnega modela, v kateri velja. Okolico veljavnosti  $i$ -tega lokalnega modela definiramo v prostoru particioniranja  $A_p$  s funkcijo  $\mu_i(\mathbf{x}_p) : \mathbb{R}^r \rightarrow [0, 1]$ , s katero vsaki točki v prostoru  $A_p$  priredi realno število med 0 in 1, ki predstavlja pripadnost točke  $\mathbf{x}_p$   $i$ -temu lokalnemu modelu. Za pripadnostne funkcije mora veljati  $\sum_{j=1}^m \mu_j(\mathbf{x}_p) = 1$ .

Lokalni modeli so lahko poljubnega tipa, vendar se povečini zaradi enostav-

nosti računanja, načrtovanja in vodenja v literaturi ponavadi uporabi linearni model, ki ga smo uporabili tudi tekom magistrskega dela.

Naslednje podpoglavje 3.1.2 je posvečeno deljenju prostora particioniranja ter izračunu pripadnostnih funkcij  $\mu_i(\mathbf{x}_p)$ , podpoglavje 3.1.3 pa opiše iskanje lokalnega modela za izračun njegovega izhoda  $\hat{y}_i(\mathbf{x})$  ter njihov izračun.

### 3.1.2 Deljenje prostora

Za identifikacijo modela TS je potrebno identificirati mehka področja, na katerih nato identificiramo še lokalne modele. Mehka področja se lahko določijo na več načinov. Pri preprostih sistemih lahko to storimo ročno, pri zapletenejših pa se poslužujemo avtomatiziranih metod deljenja prostora (npr.: LOLIMOT [13], SUHICLUST [12], ICLA, metoda mehkih c-povprečij ali FCM [7], Gustafson-Kessel [8]). Cilj deljenja prostora je identificirati pripadnostne funkcije podpodročij.

Najprej je v podpoglavju 3.1.2.1 opisana osnovna avtomatizirana metoda deljenja prostora – metoda FCM. Nato je v podpoglavju 3.1.2.2 razložena njena razširitev za splošnejše oblike rojev imenovana metoda Gustafson-Kessel (metoda GK), poglavje kasneje pa še njegova rekurzivna različica. Metoda GK ima nekaj slabosti, ki se jih lotimo v naslednjih podpoglavjih: podpoglavje 3.1.2.4 predloži metode za izogib singularnosti kovariančne matrike, podpoglavje 3.1.2.5 se loti problemov, ki nastanejo zaradi normiranja pripadnosti, v podpoglavju 3.1.2.6 pa predstavimo različne kombinacije omenjenih izboljšav, vsako s svojo namembnostjo.

#### 3.1.2.1 Metoda mehkih c-povprečij (FCM)

Rojenje z metodo mehkih c-povprečij (v nadaljevanju FCM) je ena osnovnejših metod za deljenje prostora. Metoda razdeli prostor na vnaprej določeno število rojev  $m$ , ki so v prostoru  $A_p$  opisani s centri oz. prototipi roja  $\mathbf{v}_i = [v_1 \dots v_q]^T$  ter faktorjem mehкости  $\eta$ . Pripadnost je odvisna od Evklidske razdalje med vzorcem



ter centrom roja  $\mathbf{v}_i$ , s faktorjem mehkosti pa določamo mehkost prehoda med različnimi roji.

Algoritem temelji na minimizaciji kriterijske funkcije (3.2), kjer je  $d$  Evklidova razdalja med  $k$ -tim vzorcem v particijskem prostoru  $\mathbf{x}_p(k)$  ter prototipom  $i$ -tega roja  $\mathbf{v}_i$ ,  $\mu_i(k)$  je pripadnost  $k$ -tega vzorca  $i$ -temu roju,  $\eta$  pa je faktor mehkosti. Optimizacija upošteva omejitve za pripadnosti (3.3).

$$J(\mathbf{X}_p, \mathbf{U}, \mathbf{V}) = \sum_{i=1}^m \sum_{k=1}^N (\mu_i(k))^\eta d^2(\mathbf{x}_p(k), \mathbf{v}_i) \quad (3.2)$$

$$\sum_{i=1}^m \mu_i(k) = 1 \quad \forall k \quad (3.3)$$

$$\mathbf{U} = \begin{bmatrix} \mu_1(1) & \mu_2(1) & \dots & \mu_m(1) \\ \mu_1(2) & \mu_2(2) & \dots & \mu_m(2) \\ \vdots & \vdots & \ddots & \vdots \\ \mu_1(N) & \mu_2(N) & \dots & \mu_m(N) \end{bmatrix}, \quad (3.4)$$

kjer je  $\mathbf{X}_p = [\mathbf{x}_p(1) \mathbf{x}_p(2) \dots \mathbf{x}_p(N)]^T$  matrika spremenljivk prostora particioniranja,  $\mathbf{U}$  matrika pripadnosti ter  $\mathbf{V} = [\mathbf{v}_1 \dots \mathbf{v}_m]$  matrika prototipov. Optimizacijski problem je rešljiv s pomočjo Picardove iteracije. Rezultat sta enačbi za izračun centrov rojev (3.5) ter pripadnostnih funkcij (3.6), med katerimi iteriramo toliko časa, dokler razlika med  $\mathbf{V}$  ali  $\mathbf{U}$  sosednjih iteracij postane zadovoljivo majhna.

$$\mathbf{v}_i = \frac{\sum_{k=1}^N \mu_i(k)^\eta \mathbf{x}_p(k)}{\sum_{k=1}^N \mu_i(k)^\eta} \quad (3.5)$$

$$\mu_i(k) = \frac{d_i(k)^{\frac{-2}{\eta-1}}}{\sum_{j=1}^m d_j(k)^{\frac{-2}{\eta-1}}} \quad (3.6)$$

Povezavo med iteracijami napravimo pri računanju prototipov rojev (3.5), kjer računamo z vrednostmi pripadnosti prejšnje iteracije. Z vpeljavo iteracij v enačbi (3.5) ter (3.6) dobimo enačbe (3.7) ter (3.8). Evklidske dolžine  $d_i^{(q)}$  (3.9) izračunamo s prototipi rojev iste iteracije.

$$\mathbf{v}_i^{(q)} = \frac{\sum_{k=1}^N \mu_i^{(q-1)}(k)^\eta \mathbf{x}_p(k)}{\sum_{k=1}^N \mu_i^{(q-1)}(k)^\eta} \quad (3.7)$$

$$\mu_i^{(q)}(k) = \frac{d_i^{(q)}(k)^{\frac{-2}{\eta-1}}}{\sum_{j=1}^m d_j^{(q)}(k)^{\frac{-2}{\eta-1}}} \quad (3.8)$$

$$d_i^{(q)} = d_i^{(q)}(k) = \sqrt{(\mathbf{x}_p(k) - \mathbf{v}_i^{(q)})^T (\mathbf{x}_p(k) - \mathbf{v}_i^{(q)})} \quad (3.9)$$

Zaradi iterativne narave algoritma imamo več različnih možnosti inicializacije, ali apriori nastavimo vrednosti prototipov  $\mathbf{v}^{(0)}(k)$  ali pa vrednosti pripadnosti  $\mu_i^{(0)}(k)$ . V magistrski nalogi smo nastavili vrednosti pripadnosti (3.10). Nastavimo jih na naključne vrednosti v intervalu  $(0, 1]$  (odprti interval pri 0 je v izogib primeru vsote  $\sum_{j=1}^m \mu_j^{(0)*}$  enake nič), ki jih nato po vzorcih normiramo, da zadostimo pogoju (3.3).

$$\mu_i^{(0)} = \frac{\mu_i^{(0)*}}{\sum_{j=1}^m \mu_j^{(0)*}}, \quad \text{kjer je } \mu_i^{(0)*} \in (0, 1] \quad \forall i \quad (3.10)$$

Psevdokoda algoritma je podana v algoritmu 1. Najprej nastavimo parametre rojenja  $(\eta, m)$  ter inicializiramo pripadnostno matriko  $\mathbf{U}^{(0)}$ . Nato pa začnemo iterativni del, kjer izmenično računamo prototipe rojev ter pripadnosti vzorcev, dokler niso razlike med trenutno in naslednjo iteracijo dovolj majhne.

Rezultat algoritma so prototipi rojev, ki jih uporabimo za določitev lokalnih modelov (poglavje 3.1.3).

Problemi rojenja FCM so v togi obliki rojev (hipersfere), nezmožnosti modeliranja časovno spremenljivih sistemov, v velikem vplivu inicializacije na delitev prostora ter ročna nastavitve števila rojev. V kombinaciji z modeli TS rojenje FCM ne upošteva kvalitete lokalnih modelov pri deljenju prostora.

Vseeno pa zadovoljivo razdeli prostor particioniranja brez ročnega deljenja, kar je tudi cilj rojenja FCM.

**Algoritem 1** Rojenje FCM

- 
- 1: Nastavimo parametre  $\eta, m$
  - 2: Inicializiraj pripadnostno matriko  $\mathbf{U}^{(0)}$
  - 3:  $q = 0$
  - 4: **repeat**
  - 5:      $q = q + 1$
  - 6:     Posodobi prototipe rojev  $\mathbf{v}_i^{(q)}$
  - 7:     Izračunaj matriko Evklidovih razdalj vzorcev od posodobljenih rojev  $\mathbf{D}^{(q)}$
  - 8:     Izračunaj novo pripadnostno matriko  $\mathbf{U}^{(q)}$
  - 9: **until**  $\|\mathbf{U}^{(q)} - \mathbf{U}^{(q-1)}\| < 10^{-3}$
- 

**3.1.2.2 Rojenje Gustafson-Kessel ter variacije**

V delih [8] [20] je za razdelitev prostora na mehka področja uporabljena iterativna metoda rojenja Gustafson-Kessel (v nadaljevanju "rojenje GK"), prvotno objavljeno v delu [8]. Rojenje GK izhaja iz rojenja FCM [7], opisanega v poglavju 3.1.2.1.

Edina razlika med rojenjem FCM ter GK je v računanju razdalje med vzorci ter prototipi rojev. Namesto Evklidske razdalje se v GK rojenju uporablja Mahalanobisova razdalja (3.13). Mahalanobisova razdalja za razliko od Evklidske upošteva tudi obliko roja definirano z mehko kovariančno matriko  $\mathbf{F}_i$  (3.11), kar nam omogoča, da roje lahko opišemo tudi v obliki hiperelips. Ta bolj splošni opis rojev nam omogoča bolj realno deljenje območja delovanja procesa. Za določitev mehkega roja moramo sedaj poleg prototipa  $\mathbf{v}_i$  izračunati tudi mehko kovariančno matriko  $\mathbf{F}_i$ .

$$\mathbf{F}_i^{(q)} = \frac{\sum_{k=1}^N \mu_i^{(q)}(k)^\eta (\mathbf{x}_p(k) - \mathbf{v}_i^{(q)})(\mathbf{x}_p(k) - \mathbf{v}_i^{(q)})^T}{\sum_{k=1}^N \mu_i^{(q)}(k)^\eta}, \quad \forall i \quad (3.11)$$

$$\mathbf{A}_i^{(q)} = \rho_i \det(\mathbf{F}_i^{(q)})^{1/p} \mathbf{F}_i^{(q)-1} \quad (3.12)$$

$$d_i^{(q)}(k) = \sqrt{(\mathbf{x}_p(k) - \mathbf{v}_i^{(q)})^T \mathbf{A}_i^{(q)} (\mathbf{x}_p(k) - \mathbf{v}_i^{(q)})} \quad (3.13)$$

$\eta$  je že omenjeni faktor mehкости,  $\mathbf{F}_i$  je mehka kovariančna matrika roja  $i$ ,  $\mathbf{A}_i$  pa P-norma matrike roja  $i$ ,  $p$  je dimenzija matrike  $\mathbf{F}_i$ ,  $\rho_i$  pa velikost  $i$ -tega roja, ki ga ponavadi vsem rojem nastavimo na 1, ker predpostavljamo, da so roji enako veliki (le drugače oblikovani v prostoru). Psevdokoda rojenja GK je predstavljena v algoritmu 2.

---

**Algoritem 2** Rojenje GK
 

---

- 1: Nastavimo parametre  $\eta, m, \rho_i$
  - 2: Inicializiraj pripadnostno matriko  $\mathbf{U}^{(0)}$
  - 3:  $q = 0$
  - 4: **repeat**
  - 5:      $q = q + 1$
  - 6:     Posodobi prototipe rojev  $\mathbf{v}_i^{(q)}$
  - 7:     Posodobi matrike  $\mathbf{F}_i^{(q)}$  ter  $\mathbf{A}_i^{(q)}$
  - 8:     Izračunaj matriko Mahalanobisovih razdalj  $\mathbf{D}^{(q)}$
  - 9:     Izračunaj novo pripadnostno matriko  $\mathbf{U}^{(q)}$
  - 10: **until**  $||\mathbf{U}^{(q)} - \mathbf{U}^{(q-1)}|| > 10^{-3}$
- 

Rojenje GK generalizira obliko rojev v primerjavi z rojenjem FCM – ostale že omenjene slabosti rojenja FCM se pojavijo tudi pri rojenju GK. Zaradi generalizacije oblike pri rojenju GK lahko v nekaterih primerih (malo vzorcev v roju, močna korelacija vhodnih spremenljivk, odsotnost šuma) postane kovariančna matrika  $i$ -tega roja  $\mathbf{F}_i$  singularna, zaradi česar je nemogoče izračunati njen inverz. V tem primeru algoritem javi napako in se prekine.

**Upoštevanje kvalitete lokalnih modelov pri rojenju GK** Ena izmed slabosti rojenja GK je ne upoštevanje povratne informacije o kvaliteti izhoda modela, saj išče roje le na podlagi zgoščenosti podatkov v prostoru particioniranja  $A_p$ . Zaradi tega se lahko v istem roju znajdejo vzorci, ki so blizu v prostoru

particioniranja, a z drugačnimi lastnostmi. Najpogostejši primer tega je prikazan na sliki 3.2, ko imamo največ meritev in vzorcev v delu največje nelinearnosti – prehodu med dvema rojema.

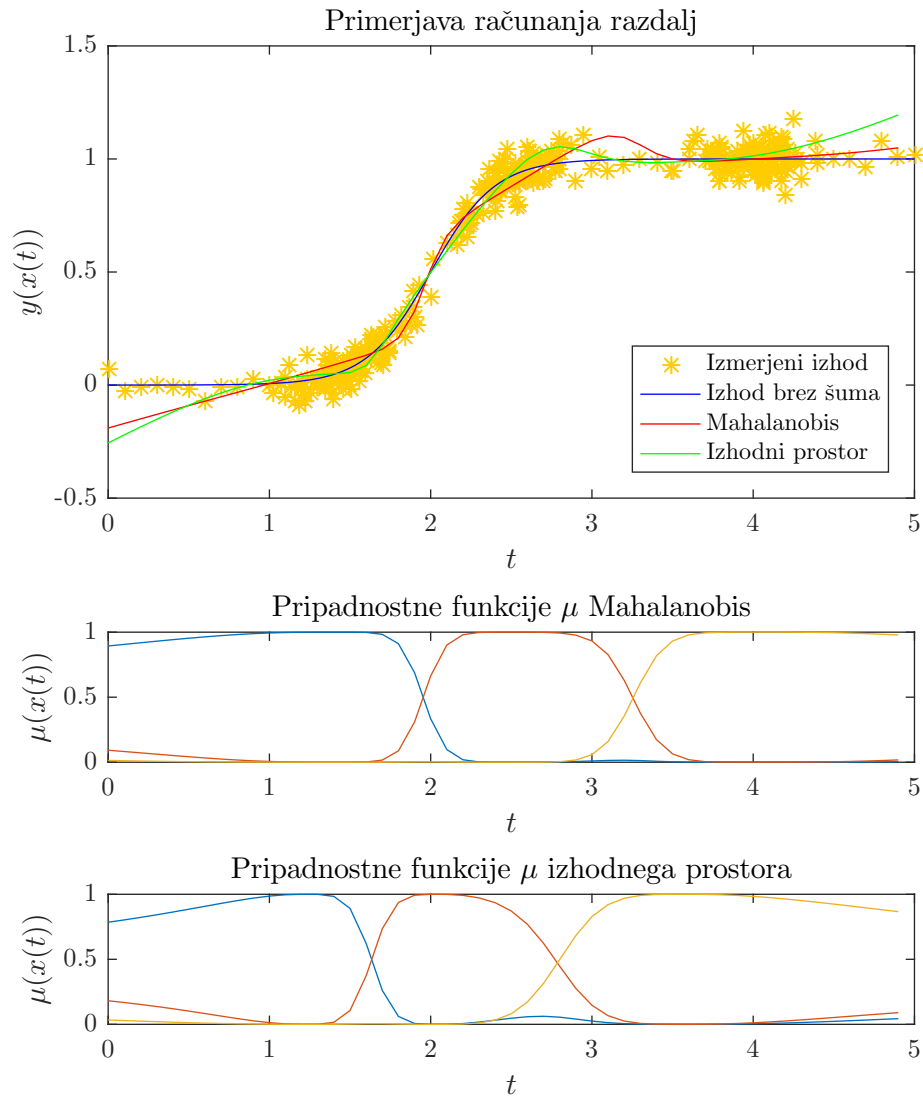
Rešitev je bila predlagana v [21] – zamenjava Mahalanobisove razdalje z razdaljo med izhodom  $i$ -tega modela ter izmerjenim izhodom, ki jo označimo z  $\zeta_i$  (3.14). S tem združimo vzorce s podobnimi lastnosti, ne glede na razdaljo med njimi. Da pa lahko za vsako iteracijo izračunamo izhode lokalnih modelov, moramo njihove parametre vsako iteracijo tudi izračunati - več o načrtovanju in računanju lokalnih modelov je opisano v podpoglavju 3.1.3.

$$\zeta_i^{(q)}(k) = f_y(\mathbf{y}(k), \hat{\mathbf{y}}_i^{(q-1)}(k)) \quad (3.14)$$

$$f_y(\mathbf{a}, \mathbf{b}) = \sum_{j=1}^Q |a_j - b_j| \quad , \quad (3.15)$$

kjer je  $\mathbf{y}(k)$  vektor izmerjenega izhoda  $k$ -tega vzorca,  $\hat{\mathbf{y}}^{(q-1)}(k)$  izhod  $i$ -tega lokalnega modela  $k$ -tega vzorca v  $q-1$  iteraciji,  $f_y$  pa funkcija razdalje v izhodnem prostoru, ki je tekom magistrske naloge izračunana kot vsota absolutnih razlik komponent (3.15).  $Q$  je število meritev izhoda. Tako enačbo smo vzeli zaradi narave problema napovedovanja električne energije, saj je strošek distributerja linearno odvisen od vsote absolutnih razlik med napovedjo ter dejansko porabo.

Pri napovedovanju  $\hat{\mathbf{y}}(k)$  pa izmerjenega izhoda  $\mathbf{y}(k)$  še nimamo, kar pomeni, da na podlagi izhodnih razdalj  $\zeta_i$  ne moremo napovedovati. Tako pri napovedovanju izhoda uporabimo Mahalanobisovo razdaljo  $d_i$  med  $\mathbf{x}_p(k)$  ter prototipom roja  $\mathbf{v}_i$  v prostoru particioniranja  $A_p$ . Ker pa  $\mathbf{v}_i$  iterativno ne računamo, ga izračunamo po končanem rojenju na podlagi matrike pripadnosti  $\mathbf{U}$  po enačbi (3.5). Izračunati moramo tudi kovariančne matrike mehkih rojev  $\mathbf{F}_i$  z enačbo (3.11). Postopek računanja je zapisan v algoritmu 3.  $\mathbf{Z}$  je matrika izhodnih razdalj (3.16).



Slika 3.2: Prikaz rezultatov rojenja GK z različnimi razdaljami in rezultajočega modela

$$\mathbf{Z} = \begin{bmatrix} \zeta_1(1) & \zeta_2(1) & \dots & \zeta_m(1) \\ \zeta_1(2) & \zeta_2(2) & \dots & \zeta_m(2) \\ \vdots & \vdots & \ddots & \vdots \\ \zeta_1(N) & \zeta_2(N) & \dots & \zeta_m(N) \end{bmatrix} \quad (3.16)$$

---

**Algoritem 3** Rojenje GK z izhodno razdaljo

---

- 1: Nastavimo parametre  $\eta, m, \rho_i$
  - 2: Inicializiraj pripadnostno matriko  $\mathbf{U}^{(0)}$
  - 3: Izračunaj izhodiščne parametre lokalnih modelov na podlagi  $\mathbf{U}^{(0)}$
  - 4:  $q = 0$
  - 5: **repeat**
  - 6:      $q = q + 1$
  - 7:     Izračunaj matriko izhodnih razdalj  $\mathbf{Z}^{(q)}$
  - 8:     Izračunaj novo pripadnostno matriko  $\mathbf{U}^{(q)}$
  - 9:     Posodobi parametre lokalnih modelov
  - 10: **until**  $\|\mathbf{U}^{(q)} - \mathbf{U}^{(q-1)}\| > 10^{-3}$
  - 11: Izračunaj prototipe rojev  $\mathbf{v}_i^{(q)}$  ter matriki  $\mathbf{F}_i^{(q)}$  in  $\mathbf{A}_i^{(q)}$
- 

Ta pristop sicer vzorce s podobnimi lastnosti roji skupaj, vendar se ti vzorci lahko nahajajo zelo daleč v prostoru particioniranja. To pri pretvorbi roja v opis z  $\mathbf{v}_i$  ter  $\mathbf{F}_i$  hitro povzroči nerealne oblike rojev, saj se prototip roja nahaja med zanj popolnoma netipičnimi vzorci. Ugotovimo, da ima rojenje z uporabo izhodne razdalje veliko lokalnih minimumov, ki pa so lahko precej slabši od globalnega. Da se približamo globalnemu minimumu, moramo rojenje večkrat ponoviti, kar podaljša čas gradnje modela. Primer slabega particioniranja z uporabo izhodne razdalje je prikazan na sliki 3.3.

Da se izognemo združevanju oddaljenih vzorcev, predlagamo, da poleg pripadnosti na bazi izhodne razdalje  $\mu_i(\zeta_i)$  upoštevamo tudi pripadnosti na bazi Mahalanovisove razdalje  $\mu_i(d_i)$ . S tem pri računanju pripadnosti  $\mu_i$  vpeljemo tudi informacijo o razdaljah v prostoru particioniranja  $A_p$ , s čimer preprečimo združevanje oddaljenih vzorcev v isti roj. Pri združitvi obeh pripadnosti želimo uresničiti sledeče zahteve:

- Če sta obe pripadnosti  $i$ -temu roju blizu 1 (vzorec je blizu prototipu in izhod modela je blizu izmerjenemu izhodu), potem želimo, da je tudi združena pripadnost  $i$ -temu roju enaka 1

- Če je pripadnost  $i$ -temu roju na bazi izhodne razdalje blizu 0 (kar pomeni da lokalni model za tisti vzorec slabo aproksimira obnašanje sistema), potem tega vzorca nočemo upoštevati pri gradnji lokalnega modela, kar pomeni da želimo skupno pripadnost blizu 0 ne glede na pripadnost na bazi Mahalanobisove razdalje
- Če je pripadnost  $i$ -temu roju na bazi Mahalanobisove razdalje blizu 0 (kar pomeni da je vzorec zelo oddaljen od prototipa  $i$ -tega roja), potem tega vzorca nočemo upoštevati pri računanju prototipa  $\mathbf{v}_i$  in kovariančne matrike  $\mathbf{F}_i$ , kar pomeni da tudi v tem primeru želimo skupno pripadnost blizu 0, ne glede na pripadnost na bazi izhodne razdalje.
- Če sta obe pripadnosti blizu 0.5, potem želimo tudi združeno pripadnost blizu 0.5.

Vse omenjene zahteve izpolnimo z enačbo (3.17), kjer korenimo zmnožek obeh pripadnosti. Dobljene pripadnosti nato še normiramo (3.18).

$$\mu_i^* = \sqrt{\mu_i(d_i) \cdot \mu_i(\zeta_i)} \quad (3.17)$$

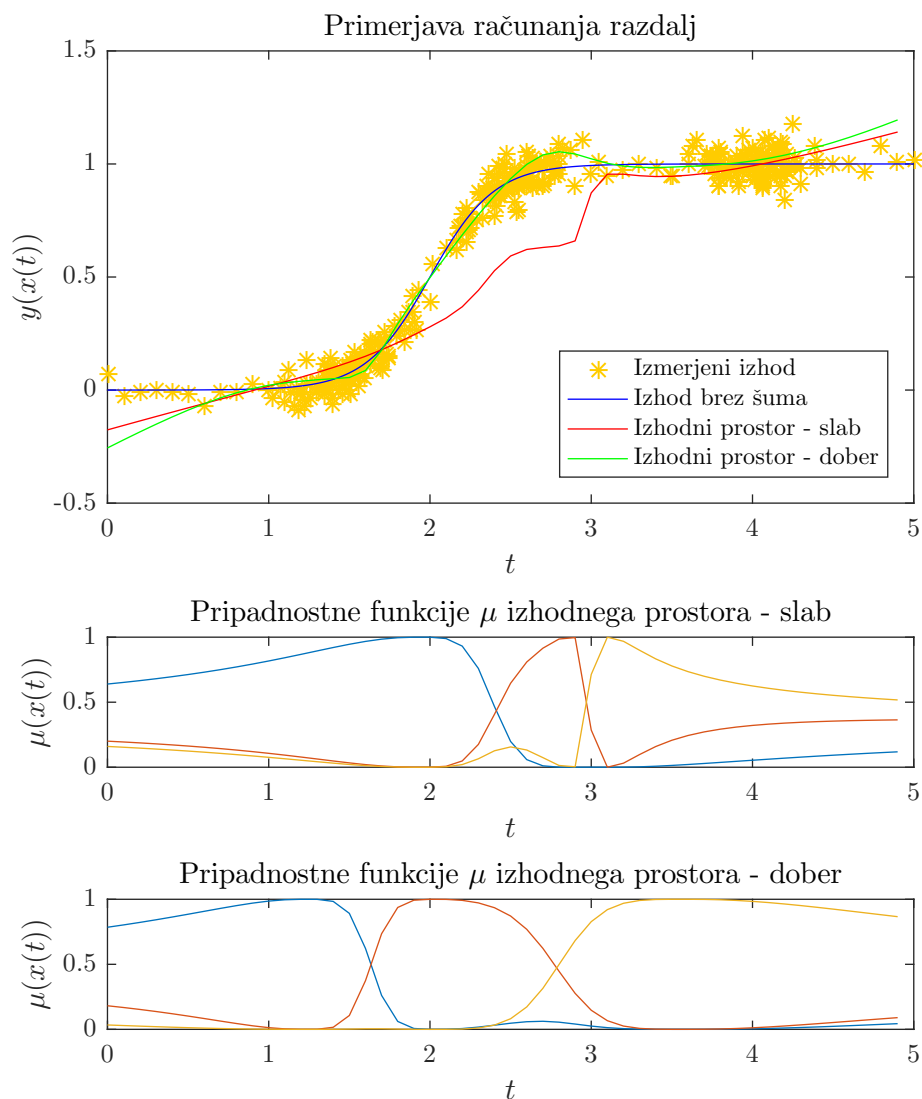
$$\mu_i = \frac{\mu_i^*}{\sum_{j=1}^m \mu_j^*} \quad (3.18)$$

S tem ko za pripadnost računamo obe razdalji (Mahalanobisovo  $d_i^{(q)}$  ter izhodno  $\zeta_i^{(q)}$ ), moramo vsak korak posodabljeni tako prototipe  $\mathbf{v}$  in matrike  $\mathbf{A}_i, \mathbf{F}_i$  mehkih rojev kot tudi parametre lokalnih modelov. Psevdokoda predlaganega algoritma je prikazana v algoritmu 4.

Zaradi uporabe izhodnih razdalj pri računanju pripadnosti predlaganega načina ne moremo uporabiti pri napovedovanju. Za napovedovanje smo primorani uporabiti pripadnost na podlagi Mahalanobisovih razdalj (3.13).

Primerjava vseh omenjenih razdalj med vzorci pri rojenju GK je prikazana na sliki 3.4. Vidimo, da najboljšo aproksimacijo dobimo z modelom, kjer smo pri ro-





Slika 3.3: Prikaz dobrega ter slabega rezultata rojenja GK z izhodnimi razdaljami

jenju uporabili pripadnosti na podlagi izhodne razdalje (algoritem 3), vendar smo za doseg dobrega rezultata potrebovali veliko ponovitev. Po kvaliteti rezultatih nato sledi rojenje s kombinirano pripadnostjo, najslabše rezultate pa dobimo z uporabo Mahalanobisove razdalje (če izpustimo slabe rezultate izhodnih razdalj).

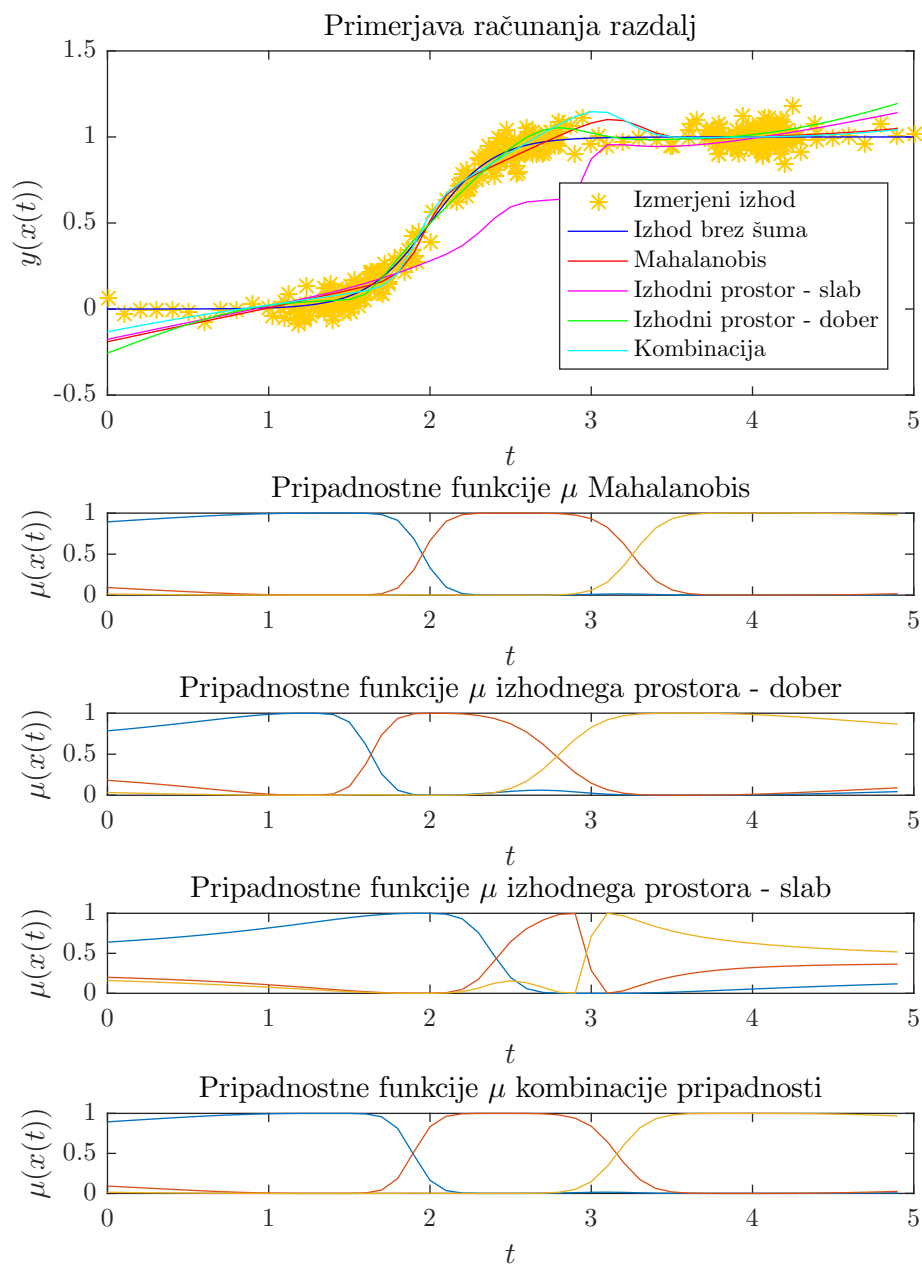
---

**Algoritem 4** Rojenje GK z združeno izhodno ter Mahalanobisovo razdaljo
 

---

- 1: Nastavimo parametre  $\eta, m, \rho_i$
  - 2: Inicializiraj pripadnostno matriko  $\mathbf{U}^{(0)}$
  - 3: Izračunaj izhodiščne parametre lokalnih modelov na podlagi  $\mathbf{U}^{(0)}$
  - 4:  $q = 0$
  - 5: **repeat**
  - 6:      $q = q + 1$
  - 7:     Posodobi prototipe rojev  $\mathbf{v}_i^{(q)}$
  - 8:     Posodobi matrike  $\mathbf{F}_i^{(q)}$  ter  $\mathbf{A}_i^{(q)}$
  - 9:     Izračunaj matriko izhodnih  $\mathbf{Z}^{(q)}$  ter Mahalanobisovih  $\mathbf{D}^{(q)}$  razdalj
  - 10:    Izračunaj novo pripadnostno matriko  $\mathbf{U}^{(q)}$
  - 11:    Posodobi parametre lokalnih modelov
  - 12: **until**  $\|\mathbf{U}^{(q)} - \mathbf{U}^{(q-1)}\| > 10^{-3}$
  - 13: Posodobi prototipe rojev  $\mathbf{v}_i^{(q)}$  ter matriki  $\mathbf{F}_i^{(q)}$  in  $\mathbf{A}_i^{(q)}$
- 

Slabost predlaganega načina je v vplivu slabih podatkov, saj sedaj vsi slabi podatki vplivajo na vse roje. Vseeno pa dobimo najboljše razmerje med rezultatom rojenja ter zanesljivostjo dobrega rezultata.



Slika 3.4: Prikaz rezultatov rojenja GK z različnimi metodami računanja pripadnosti

### 3.1.2.3 Rekurzivno rojenje GK

Rojenje GK se uporablja tudi za časovno spremenljive sisteme. Za prilagoditev rojenja GK časovno spremenljivim sistemom je bila razvita rekurzivna različica rojenja GK [9].

$$\mathbf{v}_i(k+1) = \mathbf{v}_i(k) + \frac{\mu_i(k+1)^\eta (\mathbf{x}_p(k+1) - \mathbf{v}_i(k))}{s_i(k+1)} \quad (3.19)$$

$$s_i(k+1) = \gamma_c s_i(k) + \mu_i(k+1)^\eta, \text{ kjer je } s_i(k) = \sum_{r=1}^k \mu_i(r)^\eta, \quad (3.20)$$

kjer je  $\gamma_c$  faktor pozabljanja, s katerim nastavimo za koliko imajo novejši vzorci večjo težo pri računanju parametrov rojev v primerjavi s starejšimi vzorci. Faktor lahko nastavimo na vrednost z intervala  $[0, 1]$ , pri čemer vrednost 1 pomeni, da model ne pozablja vzorcev, bolj kot pa faktor  $\gamma_c$  manjšamo, manjšo težo imajo starejši vzorci pri računanju parametrov rojev.

$$\mathbf{F}_i(k+1) = \gamma_c \frac{s_i(k)}{s_i(k+1)} \mathbf{F}_i(k) + \frac{\mu_i(k+1)^\eta}{s_i(k+1)} (\mathbf{x}_p(k+1) - \mathbf{v}_i(k+1)) \cdot (\mathbf{x}_p(k+1) - \mathbf{v}_i(k+1))^T \quad (3.21)$$

Za izračun matrike  $\mathbf{A}_i$  po enačbi (3.12) potrebujemo inverz kovariančne matrike  $\mathbf{F}_i$ , kar pomeni, da bi morali inverz izračunati vsak korak. Ker pa je inverz računalniško zahtevna operacija, uporabimo Woodburyjevo inverzno lemo (3.22) za rekurzivni izračun inverza matrike  $\mathbf{F}_i$

$$\mathbf{F}_i(k+1)^{-1} = A - \frac{B}{C} \quad (3.22)$$

$$A = \frac{1}{\gamma_c} \frac{s_i(k+1)}{s_i(k)} \mathbf{F}_i(k)^{-1} \quad (3.23)$$

$$B = \frac{1}{\gamma_c} \frac{s_i(k+1)}{s_i(k)} \mathbf{F}_i(k)^{-1} \frac{\mu_i(k+1)^\eta}{s_i(k+1)} (\mathbf{x}_p(k+1) - \mathbf{v}_i(k+1)) \cdot (\mathbf{x}_p(k+1) - \mathbf{v}_i(k+1))^T \frac{1}{\gamma_c} \frac{s_i(k+1)}{s_i(k)} \mathbf{F}_i(k)^{-1} \quad (3.24)$$

$$C = 1 + (\mathbf{x}_p(k+1) - \mathbf{v}_i(k+1))^T \frac{1}{\gamma_c} \frac{s_i(k+1)}{s_i(k)} \mathbf{F}_i(k)^{-1} \frac{\mu_i(k+1)^\eta}{s_i(k+1)} \cdot (\mathbf{x}_p(k+1) - \mathbf{v}_i(k+1)) \quad (3.25)$$

Z rekurzivno različico rojenja GK se lahko delitev prostora prilagaja časovno spremenljivemu sistemu. Ostale slabosti rojenja GK ostanejo.

#### 3.1.2.4 Omejitve kovariančne matrike v GK algoritmu

Osnovni GK algoritem opisan v poglavju 3.1.2.2 ima nekaj slabosti, ena izmed njih se izrazi v naslednjih primerih: kadar imamo v roju malo vzorcev, kadar so spremenljivke med seboj močno korelirane ali pa kadar v podatkih ni šuma. Takrat postane kovariančna matrika roja singularna in je posledično nemogoče izračunati njen inverz, zato algoritem vrne napako. V literaturi [20] sta omenjeni dve možni rešitvi problema.

Če ima kovariančna matrika lastno vrednost 0 ali je razmerje med največjo in najmanjšo vrednostjo ranga  $10^{20}$ , je matrika singularna ali skoraj singularna. Zato je v [20] predlagana omejitev razmerja med največjo in ostalimi lastnimi vrednostmi  $\xi_F$ , ki smo ga tekom modeliranja nastavili na  $\xi_F = 10^3$ .

$$\lambda_{i \max} = \max_j \lambda_{ij} \quad (3.26)$$

Najprej izločimo največjo lastno vrednost  $\lambda_{i \max}$  (3.26), potem pa poiščemo vse lastne vrednosti, ki so za koeficient  $\xi_F$  manjše kot  $\lambda_{i \max}$ , ter jih popravimo na vrednost  $\lambda_{i \max}/\xi_F$  (3.27)

$$\lambda_{ij} = \begin{cases} \lambda_{i \max}/\xi_F, & \lambda_{ij} < \lambda_{i \max}/\xi_F \\ \lambda_{ij}, & \text{sicer} \end{cases} \quad (3.27)$$

Nato s popravljenimi lastnimi vrednostmi  $\lambda_{ij}$  in njihovimi lastnimi vektorji  $\mathbf{V}_i$  rekonstruiramo kovariančno matriko  $\mathbf{F}_i$

$$\mathbf{F}_i = \mathbf{V}_i \mathbf{\Lambda}_i \mathbf{V}_i^{-1} \quad (3.28)$$

$$\mathbf{\Lambda}_i = \begin{bmatrix} \lambda_{i1} & 0 & \dots & 0 \\ 0 & \lambda_{i2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_{iq} \end{bmatrix}, \quad \mathbf{V}_i = [\phi_{i1} \quad \phi_{i2} \quad \dots \quad \phi_{iq}] \quad , \quad (3.29)$$

kjer je  $\mathbf{\Lambda}_i$  diagonalna matrika s popravljenimi lastnimi vrednostmi,  $\mathbf{V}_i$  pa matrika, katere stolpci so pripadajoči lastni vektorji  $\phi_{ij}$ .

Kljub popravku pa je lahko oblika roja ekstremno podolgovata v smeri največje lastne vrednosti, tako da takrat kovariančna matrika ne da prave informacije o realni razporeditvi podatkov. Za rešitev tega problema lahko še manjšamo razmerje med največjo ter ostalimi lastnimi vrednostmi  $\xi_F$ , lahko pa, kot je predlagano v [20], rešimo problem z upoštevanjem kovariančne matrike vseh vzorcev  $\mathbf{F}_0$  z enačbo:

$$\mathbf{F}_i^* = (1 - \gamma_F) \mathbf{F}_i + \gamma_F \det(\mathbf{F}_0)^{1/m} \mathbf{I} \quad , \quad (3.30)$$

kjer je  $\gamma_F$  uglasitveni parameter, s katerim nastavimo, v kakšni meri si bodo oblike rojev med seboj enake,  $m$  pa število rojev. Parameter  $\gamma_F$  nastavimo na vrednost v intervalu  $[0, 1]$ . Bližje kot je parameter vrednosti 1, bolj so kovariančne matrike rojev  $\mathbf{F}_i^*$  podobne enotini matriki pomnoženi z  $\det(\mathbf{F}_0)^{1/m}$  (pri  $\gamma_F = 1$  so kovariančne matrike vseh rojev enake  $\mathbf{F}_i^* = \det(\mathbf{F}_0)^{1/m} \mathbf{I}$ ). Bolj kot parameter  $\gamma_F$  manjšamo, manj vpliva ima  $\mathbf{F}_0$  na kovariančne matrike rojev  $\mathbf{F}_i^*$  (pri  $\gamma_F = 0$  se kovariančne matrike rojev ne spremenijo  $\mathbf{F}_i^* = \mathbf{F}_i$ ).

**Rekurzivna oblika omejitev** V [20] sta omejitvi predstavljeni v nerekurzivni obliki, za uporabo pri rekurzivnem rojenju GK pa ju je potrebno prirediti. Lastne vrednosti tako omejujemo po vsakem izračunu matrike  $\mathbf{F}_i$  (vsak korak) in ne

samo enkrat. Ker pa je iskanje lastnih vrednosti ter vektorjev računsko zahtevna operacija, lahko s predpostavko, da kovariančna matrika ne postane singularna v trenutku, lastne vrednosti omejujemo le na vsakih nekaj korakov. V našem primeru se omejitve lastnih vektorjev vseeno računajo vsak korak, saj so bile simulacije opravljene na dovolj močnem računalniku.

$$\lambda_{i\max}(k) = \max_j \lambda_{ij}(k) \quad (3.31)$$

$$\lambda_{ij}(k) = \begin{cases} \lambda_{i\max}(k)/\beta, & \lambda_{ij}(k) < \lambda_{i\max}(k)/\beta \\ \lambda_{ij}(k), & \text{sicer} \end{cases} \quad (3.32)$$

Z omejenimi lastnimi vrednostmi nato rekonstruiramo kovariančne matrike rojev (3.28).

Za upoštevanje oblike kovariančne matrike vseh vzorcev  $\mathbf{F}_0$  pa potrebujemo najprej njeno rekurzivno obliko. Izpeljava iz nerekurzivnih enačb je navedena z enačbami od (3.33) do (3.38). Poenostavitev v enačbi (3.37) lahko uporabimo pri velikem  $k$ , saj takrat  $\frac{k}{k+1} \approx 1$  in  $\frac{1}{k+1} \approx 0$  in posledično iz enačbe (3.35) dobimo  $\mathbf{v}_0(k+1) \approx \mathbf{v}_0(k)$ . Ker uporabimo  $\mathbf{F}_0$  le za namen omejitve kovariančnih matrik mehkih rojev  $\mathbf{F}_i$ , ta aproksimacija doprinese zanemarljivo majhno napako sistemu.

$$\mathbf{v}_0(k+1) = \frac{1}{k+1} \sum_{j=1}^{k+1} \mathbf{x}_p(j) = \frac{1}{k+1} \sum_{j=1}^k \mathbf{x}_p(j) + \frac{1}{k+1} \mathbf{x}_p(k+1) \quad (3.33)$$

$$\sum_{j=1}^k \mathbf{x}_p(j) = k \cdot \mathbf{v}_0(k) \quad (3.34)$$

$$\mathbf{v}_0(k+1) = \frac{k}{k+1} \mathbf{v}_0(k) + \frac{1}{k+1} \mathbf{x}_p(k+1) \quad (3.35)$$

$$\mathbf{F}_0(k+1) = \frac{1}{k+1} \sum_{j=1}^{k+1} (\mathbf{x}_p(j) - \mathbf{v}_0(k+1))(\mathbf{x}_p(j) - \mathbf{v}_0(k+1))^T =$$

$$\frac{1}{k+1} \sum_{j=1}^k (\mathbf{x}_p(j) - \mathbf{v}_0(k+1))(\mathbf{x}_p(j) - \mathbf{v}_0(k+1))^T + \quad (3.36)$$

$$\frac{1}{k+1} (\mathbf{x}_p(k+1) - \mathbf{v}_0(k+1))(\mathbf{x}_p(k+1) - \mathbf{v}_0(k+1))^T$$

$$\sum_{j=1}^k (\mathbf{x}_p(j) - \mathbf{v}_0(k+1))(\mathbf{x}_p(j) - \mathbf{v}_0(k+1))^T \approx k \cdot \mathbf{F}_0(k) \quad (3.37)$$

$$\mathbf{F}_0(k+1) \approx \frac{k}{k+1} \mathbf{F}_0(k+1) + \frac{1}{k+1} (\mathbf{x}_p(k+1) - \mathbf{v}_0(k+1)) \cdot$$

$$\cdot (\mathbf{x}_p(k+1) - \mathbf{v}_0(k+1))^T \quad (3.38)$$

Nato te iterativni matriki  $\mathbf{F}_0$  ter  $\mathbf{F}_i$  vstavimo v enačbo (3.30) in popravljeno matriko  $\mathbf{F}_i^*$  uporabimo v rekurzivnem rojenju GK.

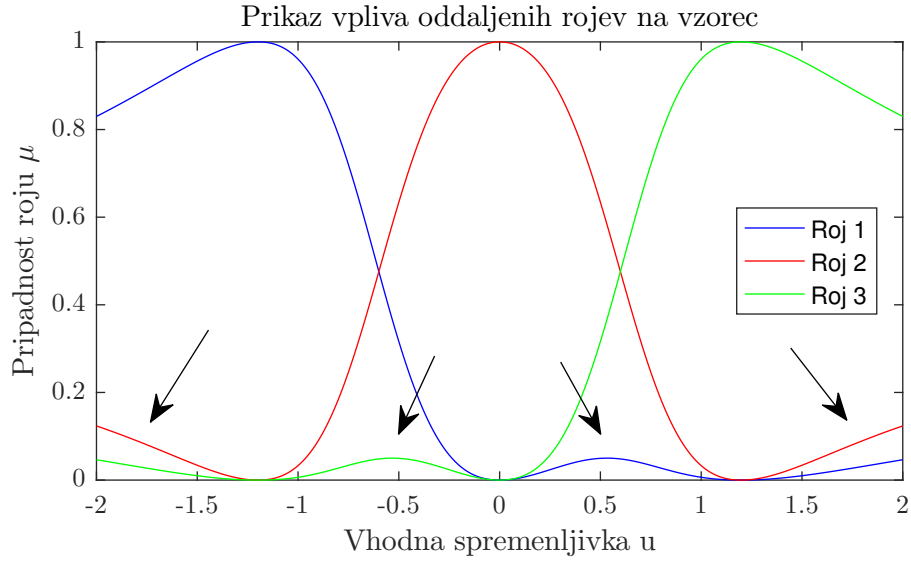
### 3.1.2.5 Eliminacija vplivov nezaželenih vplivov normiranja

Pri rojenju GK pripadnosti  $\mu_i$  vedno normiramo, saj s tem zagotovimo, da v procesu deljenja prostora upoštevamo vse vzorce. Zaradi normiranja pa pridemo do problema prikazanega na sliki 3.5. V območjih, ki niso v neposredni bližini enega izmed prototipov rojev, poleg vpliva bližnjih rojev opazimo tudi vpliv oddaljenih rojev, ki vpliva na omenjenih območjih sploh ne bi smeli imeti ( $\mu_i \neq 0$ ). Na sliki 3.5 za primer vzemimo vzorec, ki ima vhodno spremenljivko  $u = -0.5$ . Ta bi moral pripadati z določeno pripadnostjo rojema 1 ter 2, roj 3 pa ne bi smel vplivati nanj saj je 'skrit' za rojem 2, kar pa na sliki vidimo, da ni res.

Posamezni oddaljeni roj na vzorec sicer vpliva skoraj zanemarljivo malo (1 %–10 %), vendar če imamo takih oddaljenih rojev več, ta vpliv ni več zanemarljiv. Nezaželeni vplivi normiranja rastejo tudi s številom dimenzij vhodnega vektorja  $\mathbf{x}_p$ . Zato sta v sklopu magistrskega dela predlagani 2 metodi za odstranjevanje tega vpliva.

Prva metoda eliminira vpliv rojev, katerih prototip je v prostoru particioni-





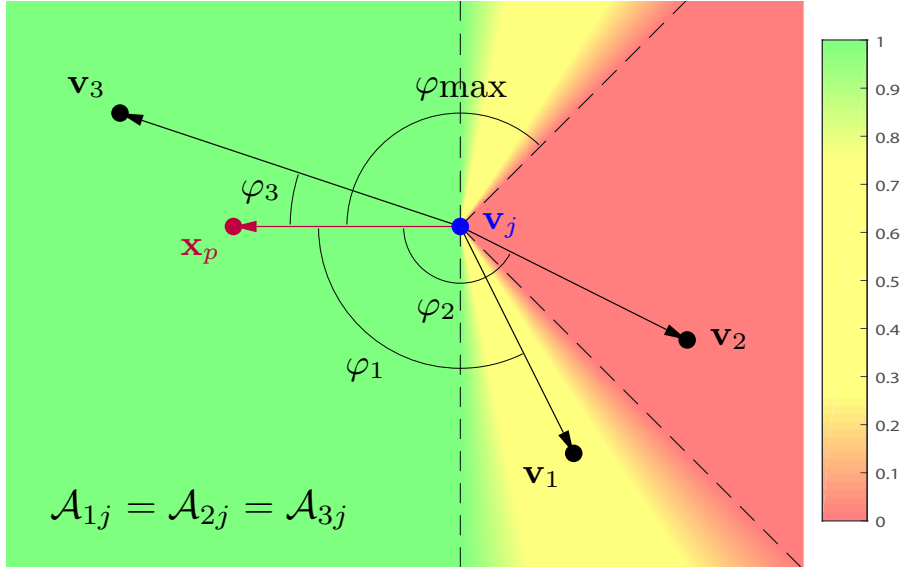
Slika 3.5: Graf pripadnostnih funkcij kvadratne funkcije po uporabljenem osnovnem GK rojenju (poglavje 3.1.2.2) s tremi roji. S puščicami so označeni vplivi oddaljenih rojev, ki tam ne bi smeli imeti vpliva.

ranja od vzorca skrit za vsaj enim drugim prototipom. Definicija skritosti:  $i$ -ti roj je skrit za  $j$ -tim rojem, če je kot  $\varphi = \angle \mathbf{v}_i \mathbf{v}_j \mathbf{x}_p$  na ravnini  $\mathcal{A}_{ij}$  topi oz. večji kot  $\pi/2$ .

Da zagotovimo zveznost pripadnostnih funkcij, definiramo koeficient vpliva skritosti  $i$ -tega roja za  $j$ -tim rojem  $\alpha_{ij}$ . Ta ima vrednost 1 ko je kot  $\varphi \leq \frac{\pi}{2}$ , nato med  $\frac{\pi}{2}$  in poljubno določenim kotom popolne eliminacije  $\varphi_{\max}$  pada sorazmerno s kotom  $\varphi$ , če pa je kot  $\varphi$  večji od kota popolne eliminacije, ima pa vrednost 0. Kakšne vrednosti ima koeficient skritosti  $\alpha_{ij}$  glede na položaj prototipov ter vzorca, je prikazano na sliki 3.6.

$$\cos \varphi = \frac{\overrightarrow{\mathbf{v}_j \mathbf{v}_i} \cdot \overrightarrow{\mathbf{v}_j \mathbf{x}_p}}{|\overrightarrow{\mathbf{v}_j \mathbf{v}_i}| |\overrightarrow{\mathbf{v}_j \mathbf{x}_p}|} \quad (3.39)$$

$$\varphi = \arccos \left( \frac{\overrightarrow{\mathbf{v}_j \mathbf{v}_i} \cdot \overrightarrow{\mathbf{v}_j \mathbf{x}_p}}{|\overrightarrow{\mathbf{v}_j \mathbf{v}_i}| |\overrightarrow{\mathbf{v}_j \mathbf{x}_p}|} \right) \quad (3.40)$$



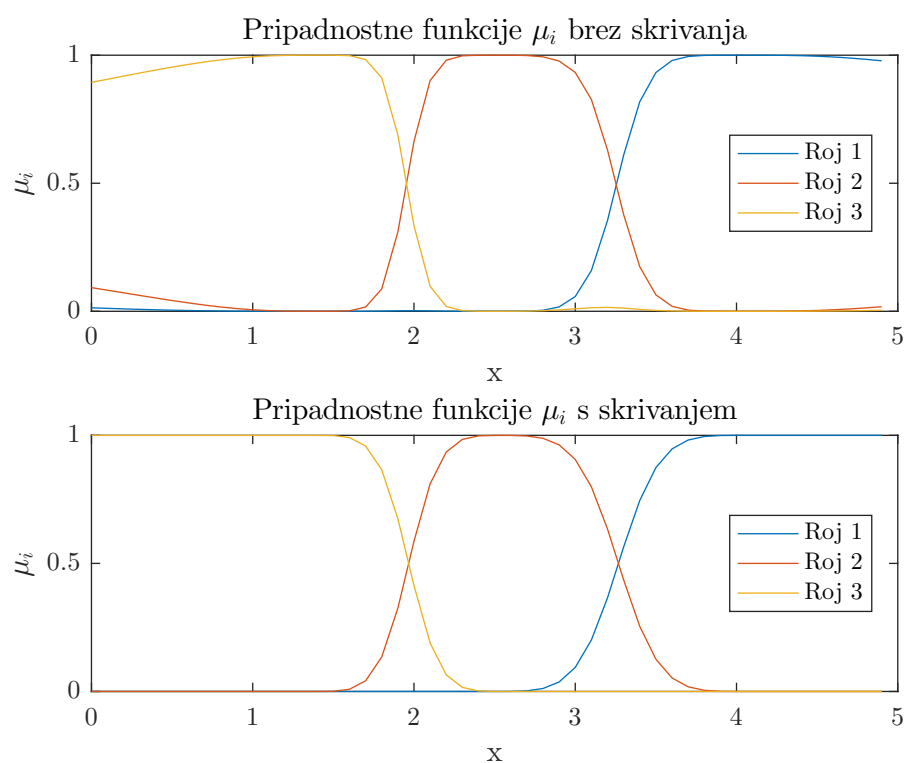
Slika 3.6: Na sliki je prikazan princip skrivanja različnih rojev ( $\mathbf{v}_1$ ,  $\mathbf{v}_2$  ter  $\mathbf{v}_3$ ) za rojem s prototipom  $\mathbf{v}_j$  pri vzorcu z spremenljivkami particioniranja  $\mathbf{x}_p$ . Barva predstavlja vrednost koeficienta skrivanja  $\alpha_{ij}$

$$\alpha_{ij} = \begin{cases} 1, & 0 \geq \varphi \geq \frac{\pi}{2} \\ 10^{-4} + (1 - 10^{-4}) \frac{\varphi_{\max} - \varphi}{\varphi_{\max} - \frac{\pi}{2}}, & \frac{\pi}{2} > \varphi \geq \varphi_{\max} \\ 10^{-4}, & \varphi < \varphi_{\max} \end{cases} \quad (3.41)$$

Skupni koeficient skritosti  $i$ -tega roja  $\alpha_i$  je produkt koeficientov skritosti  $i$ -tega roja pred vsemi ostalimi roji (3.42). Ta koeficient nato pomnožimo pripadnostim pred skrivanjem  $\mu_i^*$ , dobljene vrednosti pa še normiramo (3.43). Popravljenе pripadnosti  $\mu_i$  nato uporabimo v GK algoritmu.

$$\alpha_i = \prod_{j=1}^m \alpha_{ij} \quad (3.42)$$

$$\mu_i = \frac{\alpha_i \mu_i^*}{\sum_{j=1}^m \alpha_j \mu_j^*} \quad (3.43)$$



Slika 3.7: Prikazana je primerjava pripadnostnih funkcij modeliranja kvadratne funkcije brez skrivanja (levo) ter s skrivanjem (desno). Pripadnosti so izračunane na primeru iz slike 3.2

Predlagana metoda skrivanja rojev uspešno odstrani vpliv oddaljenih rojev, kot je prikazano na sliki 3.7. S skrivanjem izboljšamo tako delitev prostora, kot tudi posledično kvaliteto lokalnih modelov. Problem metode je potrebna računska moč, saj moramo po vsaki posodobitvi prototipov  $\mathbf{v}$  izračunati kote med vsemi pari centrov, kar pomeni da moramo izračunati  $2m^2 + m$  kotov. Za zmanjšanje računske moči predlagamo, da se primerja le tiste kote  $\angle \mathbf{v}_i \mathbf{v}_j \mathbf{x}_p$ , kjer je razdalja  $|\overrightarrow{\mathbf{v}_j \mathbf{x}_p}|$  krajša od  $|\overrightarrow{\mathbf{v}_i \mathbf{x}_p}|$ , saj je drugače kot  $\angle \mathbf{v}_i \mathbf{v}_j \mathbf{x}_p$  oster (manjši od  $\frac{\pi}{2}$ ).

Druga uporabljena metoda eliminira roje s premajhno utežjo pri določenem vzorcu, saj če je takih rojev mnogo, močno pokvarijo tako izhod TS modela kot tudi svoje lastne parametre modela. Zato predlagamo, da vse pripadnosti, ki ne dosežejo določene mejne pripadnosti  $\mu_{\min}$ , nastavimo na 0, preostale pa normiramo. Mejna utež mora biti dovolj velika, da odstrani nezaželene vplive, hkrati pa dovolj majhna, da v primeru najmanjših možnih pripadnosti ( $\mu_i \approx 1/m, \forall i$ ) ne odstranimo vseh vplivov. Zato smo se odločili, da mejno utež nastavimo na  $\frac{0.9}{m}$ , saj je dovolj velika za odstranitev manj vplivnih rojev, hkrati pa dovolj majhna, da ne eliminiramo vseh pripadnosti. Vrednost je bila nastavljena empirično. Da zagotovimo zveznost funkcije pripadnosti, predlagamo še maksimalno mejno utež  $\mu_{\max}$ , da med  $\mu_{\min}$  ter  $\mu_{\max}$  pripadnost linearno zmanjšujemo (3.47) in s tem vnesemo zveznost. Za  $\mu_{\max}$  predlagamo vrednost  $\frac{1}{m}$  iz že omenjenih razlogov.

$$\mu_{\min} = \frac{0.9}{m} \quad (3.44)$$

$$\mu_{\max} = \frac{1}{m} \quad (3.45)$$

$$(3.46)$$

$$\mu_i(k) = \begin{cases} \mu_i(k), & \mu_i(k) \geq \mu_{\max} \\ \mu_i(k) \frac{\mu_i(k) - \mu_{\min}}{\mu_{\max} - \mu_{\min}}, & \mu_{\min} \leq \mu_i(k) < \mu_{\max} \\ 0, & \mu_i(k) < \mu_{\min} \end{cases} \quad (3.47)$$

Pri testiranju predlagane metode smo ugotovili, da ni uspešna za majhno število rojev, saj je izračunana mejna pripadnost  $\mu_{\min}$  pri 2 rojih kar 0.45, s čimer izničimo velik del mehčanja prostora. V primeru manjšega števila rojev  $m$  želimo posledično tudi manjšo mejno pripadnost. Pri 2 rojih želimo mejno pripadnost 0, saj je namen predlagane metode odstraniti vpliv oddaljenih rojev večinoma v primerih velikega  $m$ , pri  $m > 10$  pa želimo mejno pripadnost enako že predlagani  $\mu_{\min} = \frac{0.9}{m}$ . Zato predlagamo od števila rojev odvisni koeficient  $k_\mu$ , izračunan po enačbi (3.48), ki ga uporabimo pri računanju mejne (3.49) ter maksimalne mejne (3.50) pripadnosti. Koeficient ima tako vrednost 0 pri 2 rojih ter  $k_\mu > 0.99$  pri 6 rojih ali več. Konstanta  $(\frac{1}{3})$  je določena poljubno, vendar jo lahko tudi poljubno spreminjamo glede na željo po hitrosti naraščaja vrednosti koeficienta  $k_\mu$  s številom rojev.

$$k_\mu = 1 - \left(\frac{1}{3}\right)^{(m-1)} \quad (3.48)$$

$$\mu_{\min} = \frac{0.9}{m} \cdot k_\mu \quad (3.49)$$

$$\mu_{\max} = \frac{1}{m} \cdot k_\mu \quad (3.50)$$

Po uporabi metod eliminiranja vpliva moramo novo izračunane pripadnosti  $\mu_j^{(0)*}$  normirati (3.51)

$$\mu_i = \frac{\mu_i^*}{\sum_{j=1}^m \mu_j^*} \quad (3.51)$$

Kot je bilo že omenjeno metoda dobro deluje le za veliko število rojev ( $m > 10$ ), saj šele pri takem številu prispevek oddaljenih rojev postane nezanemarljiv. Pri manjšemu številu rojev predlagana metoda ni tako učinkovita ravno zaradi majhnih nezaželenih vplivov.

### 3.1.2.6 Kombinacije računanja pripadnosti

Do sedaj so bile omenjene naslednje metode izboljšanja modela TS s spremembami računanja pripadnosti  $\mu_i(k)$ :

**Pripadnost v izhodnem prostoru** Pri gradnji upoštevamo kvaliteto lokalnega modela (poglavje 3.1.2.2).

#### Kombinacija pripadnosti v prostoru particioniranja ter izhodne pripadnosti

S tem kombiniramo vpliv bližine vzorcev v prostoru particioniranja ter vpliv kvalitete lokalnih modelov. Pripadnosti kombiniramo z enačbo (3.17).

**Metoda skrivanja rojev** V kolikor je roj za določen vzorec skrit za drugim vzorcem, njegov vpliv zmanjšano ali eliminiramo (poglavje 3.1.2.5).

**Eliminacija rojev z malim vplivom** V kolikor pri določenem vzorcu vrednost pripadnosti roju ne doseže mejne pripadnosti, to pripadnost zmanjšano oz. eliminiramo (poglavje 3.1.2.5).

Iz omenjenih metod lahko sestavimo različne kombinacije računanja pripadnosti, vsako s svojim namenov. Tako so bile sestavljene naslednje kombinacije računanja pripadnosti: pripadnost v prostoru particioniranja s skrivanjem rojev, pripadnost v prostoru particioniranja s skrivanjem rojev in eliminacijo rojev z malim vplivom, združitev pripadnosti v izhodnem prostoru ter pripadnosti v prostoru particioniranja s skrivanjem rojev ter združitev pripadnosti v izhodnem prostoru ter pripadnosti v prostoru particioniranja s skrivanjem rojev in eliminacijo rojev z malim vplivom.

**Pripadnost v prostoru particioniranja s skrivanjem rojev** (Algoritem 5) po izračunu Mahalabisove razdalje ter pripadajoče pripadnosti skrijemo roje. Ta izračun pripadnosti uporabimo pri napovedovanju (ne uporabljamo izmerjenega izhoda) z uporabo modela TS z malim številom dobro ločenih rojev.

**Pripadnost v prostoru particioniranja s skrivanjem rojev in eliminaciji majhnih vplivov** (Algoritem 6) razširi pripadnost v prostoru particioniranja

---

**Algoritem 5** Pripadnost v prostoru particioniranja s skrivanjem rojev

---

- 1: Izračun razdalj  $d_i$  po enačbi (3.13)
  - 2: Izračun pripadnosti  $\mu_i^*$  po enačbi (3.6)
  - 3: Metoda skrivanja rojev
  - 4: Normalizacija pripadnosti (3.51)
- 

s skrivanjem rojev z uporabo metode eliminacije majhnih vplivov pred normalizacijo. Ta način računanja pripadnosti je primeren za napovedovanje z uporabo modela TS z velikim številom slabo ločenih rojev.

---

**Algoritem 6** Pripadnost v prostoru particioniranja s skrivanjem rojev in eliminaciji majhnih vplivov

---

- 1: Izračun razdalj  $d_i$  po enačbi (3.13)
  - 2: Izračun pripadnosti  $\mu_i^*$  po enačbi (3.6)
  - 3: Eliminacija rojev z malim vplivom
  - 4: Metoda skrivanja rojev
  - 5: Normalizacija pripadnosti (3.51)
- 

**Združitev pripadnosti v izhodnem prostoru ter pripadnosti v prostoru particioniranja s skrivanjem rojev** (Algoritem 7) združi že razloženo pripadnost v prostoru particioniranja s skrivanjem rojev ter pripadnost v izhodnem prostoru z enačbo (3.17). Ta način je zaradi povratne informacije o kvaliteti modela primeren za rojenje ter posodabljanje parametrov modela, vendar ga zaradi istega razloga ne moremo uporabiti pri napovedovanju. Način je primeren za modele TS z majhnim številom rojev.

**Združitev pripadnosti v izhodnem prostoru ter pripadnosti v prostoru particioniranja s skrivanjem rojev in eliminaciji majhnih vplivov** (Algoritem 8) združi že razloženo pripadnost v prostoru particioniranja s skrivanjem rojev in eliminaciji majhnih vplivov ter pripadnost v izhodnem prostoru z enačbo (3.17). Ta način je zaradi povratne informacije o kvaliteti modela primeren za rojenje ter posodabljanje parametrov modela, vendar ga zaradi istega

---

**Algoritem 7** Združitev pripadnosti v izhodnem prostoru ter pripadnosti v prostoru particioniranja s skrivanjem rojev

---

**Izračun pripadnosti v prostoru particioniranja**

- 1:  $d_i$  po enačbi (3.13)
- 2:  $\mu_i^*(d_i)$  po enačbi (3.6)
- 3: Metoda skrivanja rojev na  $\mu_i^*(d_i)$
- 4: Normalizacija  $\mu_i^*(d_i)$

**Izračun izhodnih pripadnosti**

- 5:  $\zeta_i$  po enačbi (3.14)
- 6:  $\mu_i^*(\zeta_i)$  po enačbi (3.6)
- 7: Normalizacija  $\mu_i^*(\zeta_i)$

**Združitev omenjenih pripadnosti**

- 8:  $\mu_i^*$  po (3.17)
  - 9: Normalizacija pripadnosti  $\mu_i^*$
- 

razloga ne moremo uporabiti pri napovedovanju. Način je primeren za modele TS z velikim številom rojev.



---

**Algoritem 8** Združitev pripadnosti v izhodnem prostoru ter pripadnosti v prostoru particioniranja s skrivanjem rojev in eliminaciji majhnih vplivov

---

**Izračun pripadnosti v prostoru particioniranja**

- 1:  $d_i$  po enačbi (3.13)
- 2:  $\mu_i^*(d_i)$  po enačbi (3.6)
- 3: Metoda skrivanja rojev na  $\mu_i^*(d_i)$
- 4: Normalizacija pripadnosti  $\mu_i^*(d_i)$

**Izračun izhodnih pripadnosti**

- 5:  $\zeta_i$  po enačbi (3.14)
- 6:  $\mu_i^*(\zeta_i)$  po enačbi (3.6)
- 7: Normalizacija  $\mu_i^*(\zeta_i)$

**Združitev omenjenih pripadnosti**

- 8:  $\mu_i^*$  po (3.17)
  - 9: Normalizacija  $\mu_i^*$
- 

### 3.1.3 Gradnja lokalnih modelov

Ponovno si oglejmo poenostavljen zapis enačbe izhoda modela TS (3.52). V poglavju 3.1.2 je podrobno predstavljen postopek deljenja prostora na manjša območja ali mehke roje, ki nam služijo kot podlaga za izračun pripadnostne funkcije  $\mu_i(\mathbf{x}_p)$ . Za opis mehkega modela TS potrebujemo še njegovo drugo komponento, in to so njegovi lokalni modeli z izhodno funkcijo  $\hat{y}_i(\mathbf{x})$ , ki določijo obnašanje modela v njegovem področju.

$$\hat{y} = \sum_{i=1}^m \mu_i(\mathbf{x}_p) \hat{y}_i(\mathbf{x}) \quad (3.52)$$

Lokalni modeli so v splošnem lahko poljubnega tipa (linearni [7][8], nevronske mreže [16], tudi mehki modeli ipd.), vendar se pri modelih TS zaradi enostavnosti računanja, načrtovanja ter vodenja uporabljajo večinoma linearni modeli, ki so uporabljeni tudi v tem magistrskem delu. Podatke  $i$ -tega roja smo aproksimirali z linearnim modelom v obliki (3.53).

$$\hat{y}_i = \mathbf{x}^T \boldsymbol{\theta}_i, \quad (3.53)$$

kjer je  $\boldsymbol{\theta}_i = [\theta_{i1}, \dots, \theta_{ir}]^T$  vektor parametrov linearnega modela  $i$ -tega roja. Tekom tega poglavja smo prikazali metode ter načine računanja parametrov linearnih lokalnih modelov.

V podpoglavju 3.1.3.1 je predstavljena nerekurzivna metoda najmanjših kvadratov za izračun parametrov lokalnih modelov, ki upošteva delitev prostora na roje. Nato bo predstavljena rekurzivna različica, s katero omogočimo modelu prilagajanje na časovno spremenljivi sistem. Na koncu v podpoglavju 3.1.3.3 predlagamo uvedbo uteženega faktorja pozabljanja, s katerim nekoliko stabiliziramo rekurzivno metodo WLSE. Ta nam lahko posredno omogoči hitrejše prilagajanje na spremembe v sistemu.

### 3.1.3.1 Metoda uteženih najmanjših kvadratov (WLSE)

V splošnem se za določevanje parametrov linearnega modela iz množice vzorcev uporablja metoda najmanjših kvadratov (v nadaljevanju metoda LSE), ki poišče parametre modela z najmanjšo vsoto kvadratov razlik med izmerjenim izhodom ter izhodom modela vzorcev. Matrična enačba za določitev parametrov linearnega modela po metodi LSE je (3.54).

$$\boldsymbol{\theta} = [\mathbf{X}^T \mathbf{X}]^{-1} \mathbf{X}^T \mathbf{y} \quad (3.54)$$

$\mathbf{X} = [\mathbf{x}(k) \dots \mathbf{x}(N)]^T$  je regresijska matrika,  $\mathbf{y}$  pa je vektor izmerjenih izhodov. Slabost omenjene metode je, da posameznim vzorcem ne moremo določiti teže oziroma vpliva pri računanju parametrov. To pa nujno potrebujemo pri računanju lokalnih modelov TS modela, saj želimo, da imajo vzorci bližje lokalnemu prototipu tudi večji prispevek pri parametrih modela, vzorci dlje od prototipa pa manj.

Za upoštevanje različne teže vzorcev pri računanju parametrov se pojavi razli-

čica LSE metode, imenovana metoda uteženih kvadratov (v nadaljevanju WLSE). Pri metodi WLSE za vsak vzorec  $k$  določimo težo  $\beta(k)$ , s katero vplivamo, v kakšni meri se bo ta vzorec upošteval pri računanju parametrov linearnega modela.

$$\boldsymbol{\theta} = [\mathbf{X}^T \mathbf{W} \mathbf{X}]^{-1} \mathbf{X}^T \mathbf{W} \mathbf{y} \quad (3.55)$$

$$\mathbf{W} = \begin{bmatrix} \beta(1) & 0 & 0 & \dots & 0 \\ 0 & \beta(2) & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \beta(N) \end{bmatrix} \quad (3.56)$$

$\mathbf{W}$  je diagonalna matrika uteži vzorcev  $\beta(k)$ ,  $\mathbf{y}$  izmerjen izhod procesa,  $\mathbf{X}$  pa regresijska matrika.

Metodo WLSE sedaj uporabimo pri določevanju parametrov lokalnih modelov, kjer iščemo parametre linearnega modela  $\boldsymbol{\theta}_i$  za  $i$ -ti roj. Uteži vzorca  $\beta_i(k)$  definiramo glede na pripadnost  $i$ -temu roju. Ta pripadnost je lahko različna od pripadnosti pri deljenju prostora  $\mu_i$ , ki smo jo računali v poglavju 3.1.2. Kljub temu pa smo za namen napovedovanja porabe za uteži  $\beta_i$  uporabili kar vrednost pripadnostne funkcije  $\mu_i$ .

$$\beta_i(k) = \mu_i(k) \quad (3.57)$$

V primeru, da je izhod vzorca vektor, imamo opravka z matriko izhoda  $\mathbf{Y}(k) = [\mathbf{y}_1 \dots \mathbf{y}_l \dots \mathbf{y}_n]^T$ . Takrat za vsak  $l$ -ti element izhoda načrtamo na vsak  $i$ -ti roj svoj linearni model z vektorjem koeficientov  $\boldsymbol{\theta}_{il}$ . Pri vseh elementih izhoda računamo z istimi utežmi  $\beta_i(k)$ . Če definiramo matriko koeficientov  $i$ -tega roja  $\boldsymbol{\Theta} = [\boldsymbol{\theta}_1 \dots \boldsymbol{\theta}_n]$ , lahko izračunamo koeficiente za vse linearne modele roja po enačbi (3.58).

$$\boldsymbol{\Theta}_i = [\mathbf{X}^T \mathbf{W}_i \mathbf{X}] \mathbf{X}^T \mathbf{W}_i \mathbf{Y} \quad (3.58)$$

Z metodo WLSE ne moremo prilagajati model na časovne spremembe procesa. Še en problem metode je v močnem vplivu slabih podatkov (eng. outliers) na parametre modela, zato priporočamo njihovo ročno ali avtomatično odstranitev pred uporabo metode WLSE.

### 3.1.3.2 Rekurzivna metoda uteženih najmanjših kvadratov

Kot smo že omenili, z metodo ne moremo prilagajati modela na časovne spremembe procesa, zato se v literaturi uporablja rekurzivna metoda uteženih najmanjših kvadratov (v nadaljevanju rekurzivna metoda WLSE), s katero posodabljam parametre linearnega modela glede na trenutne vzorce.

Parametre linearnih modelov posodabljam z enačbo (3.59). Razlika  $y(k+1) - \mathbf{x}(k+1)\boldsymbol{\theta}(k)$  predstavlja napako med izmerjenim izhodom ter izhodom modela s parametri prejšnjega vzorca. V enačbi se pojavi regresijska kovariančna matrika  $\mathbf{P}(k)$ , s katero določamo, koliko vpliva ima nov vzorec na parametre modela. Večja kot je matrika, bolj se bodo parametri spremenili. Tudi  $\mathbf{P}(k)$  posodabljam z vsakim novim vzorcem. Posodobitev izračunamo po enačbi (3.60). S to enačbo vpeljemo koeficient pozabljanja linearnega modela  $\gamma_\theta \in (0, 1]$ . Večji kot je koeficient, počasneje se linearni model prilagaja spremembam v modelu, vendar je zaradi tega model bolj stabilen. V literaturi [22] se predlaga vrednost v intervalu  $\gamma_\theta \in [0.98, 1]$ .

$$\boldsymbol{\theta}(k+1) = \boldsymbol{\theta}(k) + \mathbf{P}(k+1)\mathbf{x}(k+1)\beta(k+1)(y(k+1) - \mathbf{x}(k+1)\boldsymbol{\theta}(k)) \quad (3.59)$$

$$\mathbf{P}(k+1) = \frac{1}{\gamma_\theta} \left( \mathbf{P}(k) - \frac{\beta(k+1)\mathbf{P}(k)\mathbf{x}(k+1)\mathbf{x}(k+1)^T\mathbf{P}(k)}{\gamma_\theta + \beta(k+1)\mathbf{x}(k+1)^T\mathbf{P}(k)\mathbf{x}(k+1)} \right), \quad (3.60)$$

V primeru ko vhodnih spremenljivk nimamo, matrike  $\mathbf{P}$  ne moremo izračunavati. Ker vemo, da je v tem primeru izhod linearnega modela le povprečje izmerjenih izhodov modela  $y$ , lahko metodo za rekurzivno računanje prototipov pri rekurzivnem rojenju GK (enačbi (3.20) ter (3.19)), ki sicer računa rekurzivno

povprečje spremenljivk v prostoru particioniranja, miselno uporabimo tudi pri problemu rekurzivnega računanja parametrov linearnega modela. Rezultat sta enačbi (3.61) ter (3.62).

$$\boldsymbol{\theta}(k+1) = \boldsymbol{\theta}(k) + \frac{\beta(k+1)(y(k+1) - \mathbf{x}(k+1)^T \boldsymbol{\theta}(k))}{s_{\theta}(k+1)} \quad (3.61)$$

$$s_{\theta}(k+1) = \gamma_{\theta} s_{\theta}(k) + \beta(k+1), \text{ kjer je } s_{\theta}(k) = \sum_{j=1}^k \beta(j) \quad , \quad (3.62)$$

kjer je  $s_{\theta}$  vsota uteži vzorcev.

Pri modeliranju z modelom TS se rekurzivna metoda WLSE uporabi za posodabljanje parametrov  $\boldsymbol{\theta}_i$  lokalnih linearnih modelov (oz.  $\boldsymbol{\theta}_{il}$  pri večizhodnem modelu). Uteži  $\beta_i(k)$  (utež vzorca  $k$  za računanje parametrov lokalnega modela  $i$ -tega roja) se definira enako kot pri nerekurzivni metodi WLSE s pripadnostmi  $\mu_i$ , vsakemu roju pa moramo računati svojo regresijsko kovariančno matriko  $\mathbf{P}_i$  oz. vsoto uteži  $s_{\theta_i}$ .

Za rekurzivno računanje potrebujemo tudi izhodiščne vrednosti parametrov modela  $\boldsymbol{\theta}_i(0)$  ter regresijske kovariančne matrike  $\mathbf{P}_i(0)$ . Tekom magistrskega dela za začetne vrednosti  $\boldsymbol{\theta}_i$  uporabimo rezultat nerekurzivne metode WLSE, uporabljeni na učni množini ter z utežmi, pridobljenimi z nerekurzivnim rojenjem GK. Za  $\mathbf{P}_0$  pa uporabimo kovariančno matriko vhodnih podatkov učne množice.

Z uporabljenno metodo se linearni model uspešno prilagaja na spremembe sistema. Problem metode je nastavljanje parametra pozabljanja  $\gamma_{\theta}$ , s katerim iščemo kompromis med hitrostjo prilagajanja ter variabilnostjo parametrov modela. Problem nastane tudi v primeru globalne spremembe sistema, saj moramo prilagoditi lokalne sisteme vsakega zase, s čimer dobimo napako napovedi pri prvih vzorcih vsakega roja po globalni spremembi sistema. Za ponazoritev vzemimo sistem s funkcijo (3.63).

$$y(x, t) = x^2(t) + H(t - t_0) \quad (3.63)$$

$$x(t) = \begin{cases} t & t \leq 2 \\ t - 4 & t > 2 \end{cases}, \quad (3.64)$$

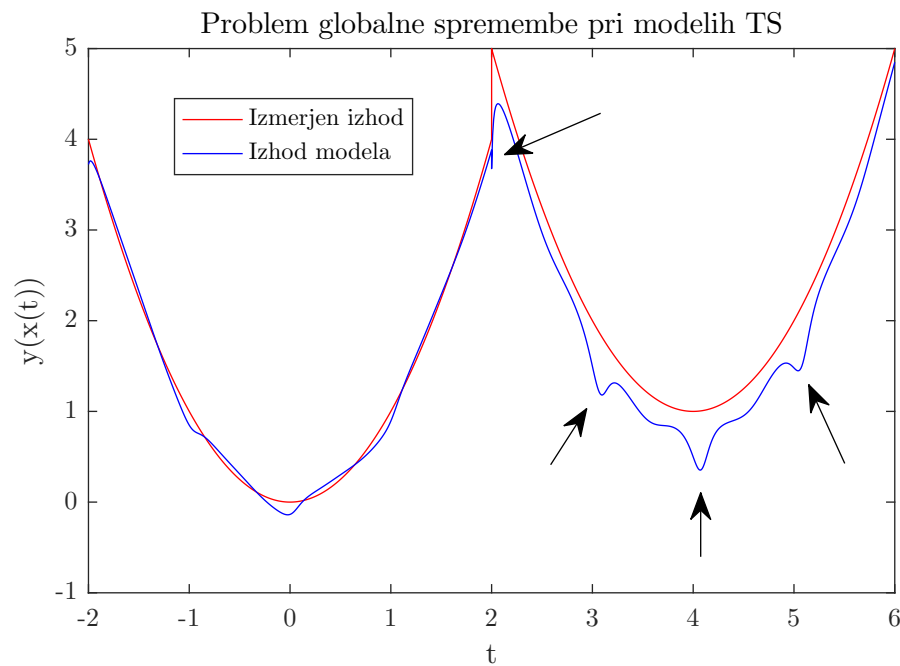
kjer je  $t$  čas,  $x(t)$  funkcija vhodne spremenljivke,  $H(t - t_0)$  enotina stopnica v trenutku  $t_0$  ter  $y(t)$  izhod sistema.

Če najprej v času  $-2 < t \leq 2$  nerekurzivno načrtamo model TS z rojenjem GK ( $\mathbf{x}_p(t) = x(t)$ ) ter metodo WLSE (diskretni korak  $\Delta t = 0.01$ ), dobimo model ki je dober približek kvadratni funkciji. Nato za definicijsko območje  $-2 < t \leq 6$  uporabimo rekurzivno metodo WLSE. Na sliki 3.8 je prikazan rezultat omenjenega modeliranja. Pri  $t_0 = 2$  se sistem spremeni (stopnica prišteje 1), takrat pričakujemo, da bo model imel napako, na katero se bo s časom prilagodil. Na primeru pa vidimo, da dobimo poleg povečane napake napovedi v času spremembe sistema tudi napako pri vsakem prehodu med roji (označeno s puščicami). To je zato, ker se vsak lokalni model posodablja le z informacijami iz svojega roja, nima pa informacije o spremembah izven lokalnega področja.

### 3.1.3.3 Uteženi koeficient pozabljanja linearnega modela

V prejšnjem poglavju smo omenili koeficient pozabljanja  $\gamma_\theta$ , s katerim manjšamo vpliv starejših vzorcev. Če so vzorci glede na čas v roje razporejeni približno naključno, imamo za vsak roj vedno dovolj sveže vzorce, s katerimi opišemo roj.

Problem nastane, ko so vzorci glede na čas močno ločeni, kar smo predstavili s funkcijo (3.63), ko jo razdelimo na 4 roje. Podrobno si pogledjmo roj v okolici  $x = 1.5$ . V časovnem obdobju  $-2 < t \leq -1$  vzorci večinoma pripadajo omenjenemu roju in takrat model dobro sledi sistemu. V intervalu  $-1 < t \leq 2$  pa roju ne pripada skoraj noben vzorec, takrat se njegova regresijska kovariančna matrika  $\mathbf{P}$  močno poveča (3.65), kar pomeni, da se bo linearni model zelo hitro prilagodil na naslednji pripadajoči vzorec. S tem se pa sistem pri ponovnem vzbujanju burno



Slika 3.8: Prikaz problema modeliranja globalne spremembe z večimi roji

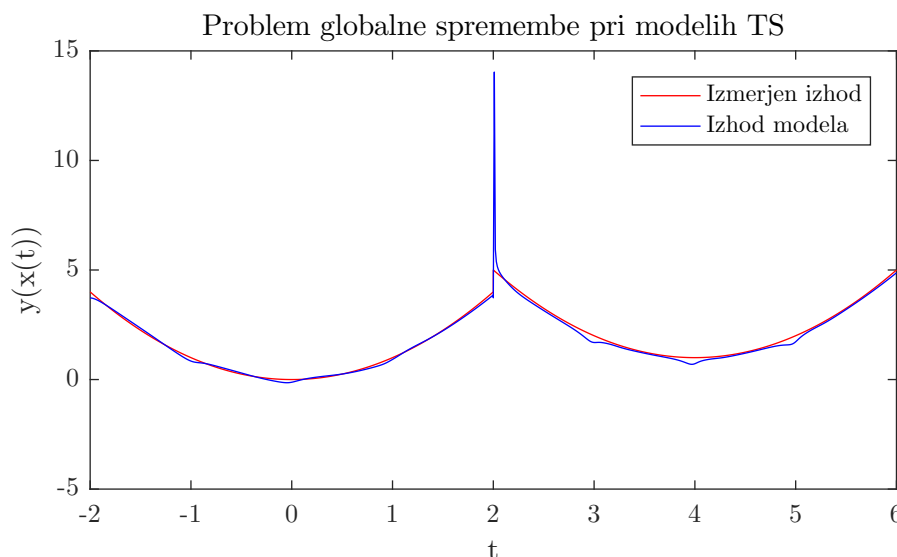
odzove na spremembe, kar je prikazano na sliki 3.9

$$P_1(t = -2) = \begin{bmatrix} 0.6012 & 0.8531 \\ 0.8531 & 1.2515 \end{bmatrix}, \quad P_1(t = 2) = \begin{bmatrix} 31.7438 & 34.6788 \\ 34.6788 & 49.8328 \end{bmatrix} \quad (3.65)$$

V izogib opisanemu problemu lahko manjšamo koeficient pozabljanja, s čimer manjšamo tudi prilagodljivost modela na spremembe v sistemu. Druga rešitev je omejevanje glede na vzbujanje, tako da lokalnega modela ne posodabljam, če ni dovolj vzbujan (predlagano v [23]). V magistrskem delu pa predlagamo uvedbo uteženega koeficienta pozabljanja linearnega modela  $\gamma_{\theta}^{(\beta)}(k)$ , računanega z enačbo (3.66).

$$\gamma_{\theta}^{(\beta)}(k) = 1 - \beta_i(k)(1 - \gamma_{\theta}) \quad , \quad (3.66)$$

kjer je  $\beta_i(k)$  utež  $k$ -tega vzorca za  $i$ -ti roj,  $\gamma_{\theta}$  pa koeficient pozabljanja linear-



Slika 3.9: Prikaz problema neuteženega koeficienta pozabljanja

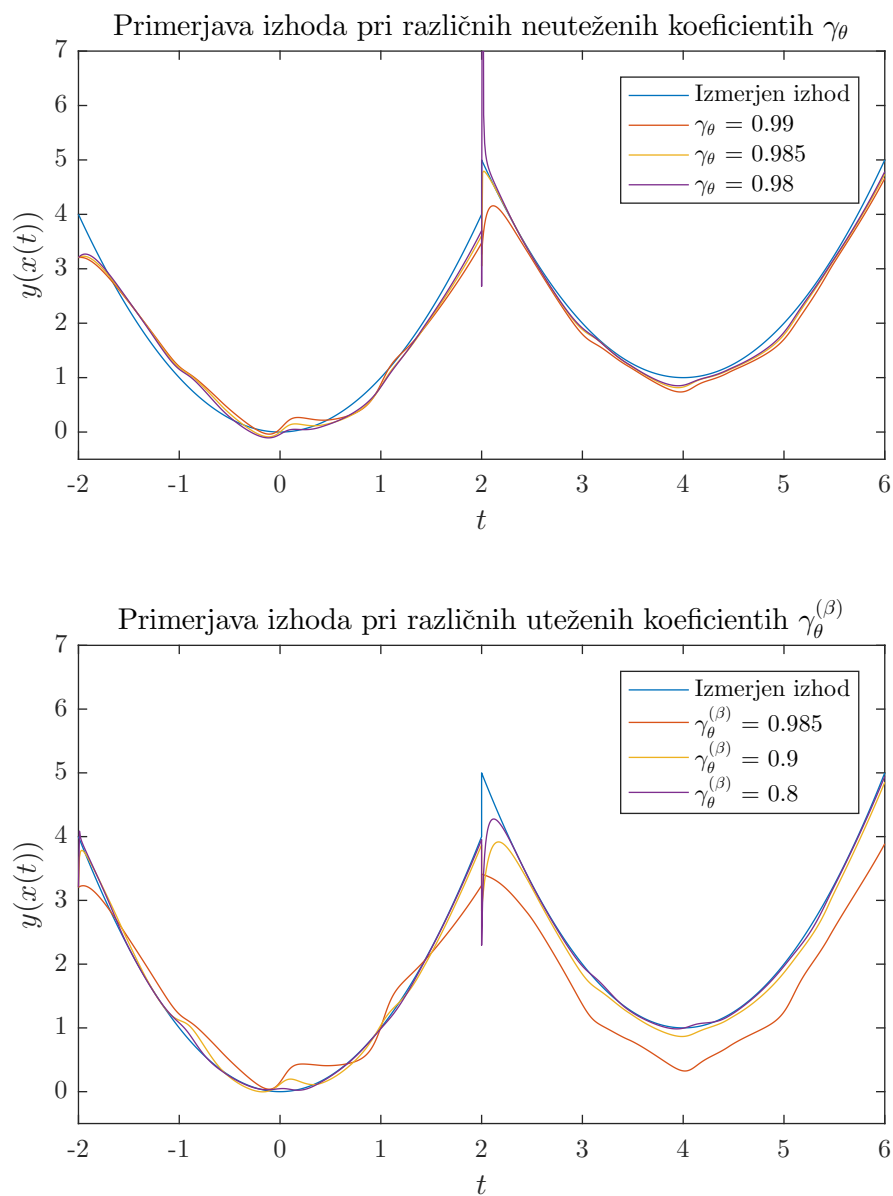
nega modela. Ta uteženi koeficient  $\gamma_{\theta}^{(\beta)}$  nato vstavimo v enačbe metode WLSE namesto neuteženega koeficienta  $\gamma_{\theta}$ .

Z omenjeno enačbo dosežemo, da v primeru, ko  $k$ -ti vzorec ne pripada  $i$ -temu roju  $\beta_i(k) = 0$ , se  $\mathbf{P}$  ne spremeni in bo s tem lokalni model enako dovzeten za spremembo pri naslednjem vzorcu, kot bi bil za trenutnega, če bi imel za utež  $\beta_i(k) > 0$ . Če pa vzorec pripada z določeno utežjo  $\beta_i(k) > 0$ , zaradi nove informacije nek del pozabimo, ker je  $\gamma_{\theta}^{(\beta)} < 1$ .

Hitro opazimo, da se  $\mathbf{P}$  pri neki določeni velikosti ne zmanjšuje več (tudi poveča ne), kar pomeni, da s parametrom  $\gamma_{\theta}$  posredno nastavimo relativno velikost matrike  $\mathbf{P}$ , ki je sicer še vedno odvisna od razporeditve vhodnih spremenljivk  $\mathbf{x}$  vzorcev.

Primerjavo uporabe uteženega ter neuteženega  $\gamma_{\theta}$  lahko vidite na sliki 3.10. Takoj vidimo, da se model z uteženim  $\gamma_{\theta}^{(\beta)}$  pri enaki vrednosti koeficienta slabše prilagaja globalni spremembi, zato moramo za hitrejšo prilagajanje koeficient ustrezno zmanjšati. Na grafih lahko tudi primerjamo občutljivost modela na koeficient pozabljanja - pri nastavitvi neuteženega koeficienta  $\gamma_{\theta}$  moramo zelo paziti,





Slika 3.10: Primerjava izhodov modelov neuteženega koeficienta  $\gamma_\theta$  (zgoraj) ter uteženega koeficienta  $\gamma_\theta^{(\beta)}$  (spodaj) z različnimi vrednostmi

saj ga lahko hitro nastavimo na takšno vrednost, da se model nezaželeno burno odzove (na sliki  $\gamma_\theta = 0.98$ ). Pri nastavljanju uteženega koeficienta pozabljanja  $\gamma_\theta^{(\beta)}$  pa tega problema nimamo, saj se model tudi v primeru majhnih vrednosti  $\gamma_\theta^{(\beta)} < 0.9$  na spremembo odzove brez burnega odziva (vseeno pa opazimo negativno špico pri  $t_0$ , za katero je pa kriv problem modeliranja funkcije (3.63) in ne direktno  $\gamma_\theta^{(\beta)}$ ).

Z uvedbo  $\gamma_\theta^{(\beta)}$  rešimo vpliv slabega vzbujanja na povečanje matrike  $\mathbf{P}$ , saj tudi dolge serije vzorcev, ki ne pripadajo določenemu roju oziroma lokalnemu modelu, ne povzročijo izgube informacije o roju, hkrati pa serija pripadajočih vzorcev lahko popolnoma spremeni parametre linearnega modela. V primeru premajhnega koeficienta pri močno dinamičnih sistemih se pa burni odzivi modela spet pojavijo.

## 3.2 Napovedovanje porabe električne energije

V uvodu so že bili predstavljeni načini in metode, ki so bile razvite za kratkoročno napovedovanje porabe električne energije. Tekom tega poglavja bosta predstavljeni metodi linearne regresije (naprej metoda LR) ter metoda robustne linearne regresije (naprej metoda RLR), s katerima bomo primerjali metodo razvito tekom magistrskega dela. Na koncu bomo predstavili še predlog zlivanja porabe med trenutno in naslednjo napovedjo, ki neodvisno napovedane sosednje vzorce zlije skupaj v zvezno porabo.

### 3.2.1 Metoda linearne regresije

Metoda LR je široko znana metoda za iskanje linearnih trendov v podatkih. Rezultat metode so parametri linearnega modela, izhod katerega lahko pri vhodnih podatkih za naslednji vzorec uporabimo za napovedovanje. Ker si lahko tudi pri problemu napovedovanja porabe pomagamo z linearnimi trendi, so metodo LR hitro posvojili in priredili tudi za ta problem.

V delu [1] je problem najprej razstavljen na 4 različne podprobleme: na posebne dneve (prazniki, dnevi s posebnimi profili porabe), na dneve, ki so od posebnih dni oddaljeni 1–3 dni, dneve, ki so oddaljeni 4–7 ter dneve, ki so oddaljeni več kot 7 dni.

Vse probleme nato razdelimo še po urah (24). Pri problemih, ki ne modelirajo posebnih dni, pa že po urah razdeljene probleme razdelimo še po dnevih v tednu (7). Skupaj rešujemo torej  $3 * 24 * 7 + 24 = 528$  podproblemov, kar pomeni, da moramo načrtati 528 linearnih modelov z uporabo metode LR.

Za podprobleme, ki modelirajo dni oddaljene več kot 7 dni od posebnih, iščemo model v obliki (3.67)

$$P_j = \beta_0 + \beta_1 T_j + \beta_2 T_j^2 + \beta_3 \phi_j + \beta_4 P_{(j-48)} + \beta_5 T_{(j-72)} \quad (3.67)$$

$P_j$  je izmerjena poraba električne energije. Indeks  $j$  predstavlja diskretno uro podatka oziroma napovedi,  $T_j$  je temperatura tekom  $j$ -te ure,  $\phi_j$  pa vlažnost.

Za podprobleme, ki so od posebnih dni oddaljeni med 4–7 dni, zaradi vpliva vikenda ne moremo več uporabiti podatka o porabi za 72 ur nazaj  $P_{(j-72)}$ . Zato za omenjene dni uporabimo model opisan v (3.68)

$$P_j = \beta_0 + \beta_1 T_j + \beta_2 T_j^2 + \beta_3 \phi_j + \beta_4 P_{(j-48)} \quad (3.68)$$

Za podprobleme, ki so od posebnih dni oddaljeni med 1–3 dni, zaradi vpliva vikenda ne moremo več uporabiti podatka o porabi niti za 72  $P_{(j-72)}$  niti za 48 ur nazaj  $P_{(j-48)}$ . Zato za omenjene dni uporabimo model opisan v (3.69)

$$P_j = \beta_0 + \beta_1 T_j + \beta_2 T_j^2 + \beta_3 \phi_j \quad (3.69)$$

Posebne dni modeliramo z enačbo (3.67). Pri tem pretekle podatke o porabi vzamemo iz množice vzorcev praznikov.

Za določitev parametrov linearnih modelov uporabimo metodo LSE z enačbo

(3.54), kjer je  $\boldsymbol{\theta} = [\beta_0 \dots \beta_r]^T$ ,  $\mathbf{X}$  matrika neodvisnih spremenljivk temperature, vlage ter preteklih porab,  $\mathbf{y}$  pa je vektor z elementi porabe v izbranih urah  $P_j$ .

Ker se porabe potrošnikov spreminjajo vsak dan, v želji, da bi načrtani modeli sledili spremembam v porabi potrošnikov, vsak dan na novo izračunamo parametre  $\boldsymbol{\theta}$  vseh modelov z vsemi podatki, ki so nam do sedaj na voljo.

Problem omenjene metode je občutljivost na napačne vzorce (eng. outliers). Slabost je tudi nezmožnost prilagajanja na časovne spremembe v porabi, saj pri računanju parametrov zastarele vzorce upoštevamo enako kot novejši in tako novejših trendov ne moremo modelirati.

### 3.2.2 Metoda robustne linearne regresije

Metoda RLR se loti problema linearne regresije glede občutljivosti na napačne vzorce oz. regresijske osamelce. Cilj metode je identificirati te regresijske osamelce in jim zmanjšati vpliv na računanje parametrov.

Ker imamo sedaj opravka z različnimi utežmi vzorcev, za določitev parametrov linearnih modelov uporabimo metodo WLSE (3.55). Uteži pri metodi RLR iščemo z minimizacijo kriterijske funkcije  $J(\boldsymbol{\theta})$  (3.71)

$$e_k = \mathbf{x}_k^T \boldsymbol{\theta} - y_k \quad (3.70)$$

$$J(\boldsymbol{\theta}) = \sum_{k=1}^N w_k e_k^2, \quad (3.71)$$

kjer je  $w_i$  utež pogreška,  $e_i$  pogrešek izhoda modela in  $N$  število dni, ki so vneseni v regresijski model. Rezultat minimizacije je iterativna metoda določevanja koeficientov, kjer za izračun uteži uporabimo metodo Tukeyevega bikvadrata (eng. Tukey's bisquare) (3.72).

$$w(e_k) = \begin{cases} \left[1 - \left(\frac{e_k}{\gamma_e}\right)^2\right]^2 & |e_k| \leq \gamma_e \\ 0 & |e_k| > \gamma_e \end{cases} \quad (3.72)$$

$$\mathbf{W} = \begin{bmatrix} w_1 & 0 & 0 & \dots & 0 \\ 0 & w_2 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & w_N \end{bmatrix} \quad (3.73)$$

Simbol  $\mathbf{W}$  predstavlja diagonalno matriko uteži,  $\gamma_e$  pa uglasitveno konstanto, definirano kot  $\gamma_e = \frac{4.685}{0.6745} \text{MAR}$ , kjer je MAR mediana absolutnih pogreškov.

Z dobljenimi utežmi po metodi WLSE (kjer matriko uteži nastavimo z enačbo (3.73)) izračunamo nove parametre modela in začnemo novo iteracijo z izračunov novih pogreškov  $e_k^{(q+1)}$ . Omenjeni iterativni algoritem ustavimo, ko kriterijska funkcija  $S$  pride v bližino minimuma, kjer velja pogoj (3.74).

$$|J(\boldsymbol{\theta}^{(q)}) - J(\boldsymbol{\theta}^{(q-1)})| < \delta \quad (3.74)$$

$J(\boldsymbol{\theta}^{(q)})$  je vrednost kriterijske funkcije v iteraciji  $q$ ,  $\delta$  pa vnaprej definirana vrednost zaustavitvenega pogoja.

Pri kratkoročnem napovedovanju porabe električne energije se metoda RLR pojavi v delu [1], kjer najprej problem napovedovanja razdelimo enako kot je razloženo v poglavju 3.2.1, nato pa namesto metode LR uporabimo metodo RLR.

Z metodo RLR tako izničimo vpliv regresijskih osamelcev na parametre modelov. Problem metode je v zahtevani procesorski moči, saj se v primerjavi z metodo LR poveča v povprečju za 5-krat (v vsaki iteraciji izračunamo vrednosti parametrov po metodi LR – in če potrebujemo vsakič 5 iteracij za zadostitev pogoja, pomeni, da potrebujemo 5-kratnik časa porabljenega za izračun po metodi RLR). Ta dodatna procesorska moč se močno pozna, kjer vsak korak računamo parametre 528 linearnih modelov. To lahko precej omilimo s tem, da posodabljam parametre le tistih modelov, ki veljajo za ta dan.

### 3.2.3 Zlivanje med trenutno ter naslednjo napovedjo

V magistrskem delu smo združili urne podatke o porabi v vektor, katerega definiramo kot izhod modela. Za ponazoritev problema tega združevanja predpostavimo, da za vsak dan napovedujemo le povprečje porabe za tisti dan, kar pomeni da je napoved porabe v vsaki uri enaka povprečni porabi za tisti dan. Seveda lahko hitro opazimo, da bo pri napovedi porabe med zadnjo uro nekega dne ter prvo uro naslednjega dne prišlo do nezveznosti porabe, kar pa se v realnosti ne dogaja (razen pri izpadih elektrike, ki pa jih ne moremo napovedati vnaprej).

Za rešitev omenjenega problema združevanja urne porabe v vzorce predlagamo zlivanje med trenutno ter naslednjo napovedjo, kar pomeni, da na začetne elemente napovedi določenega vzorca vplivajo elementi napovedi prejšnjega vzorca, na končne pa vplivajo elementi napovedi naslednjega vzorca. Zlivanje je realizirano z linearnim prehodom med istoležnimi elementi prejšnje oz. naslednje napovedi, izračunano z enačbami (3.75), (3.76) ter (3.77).

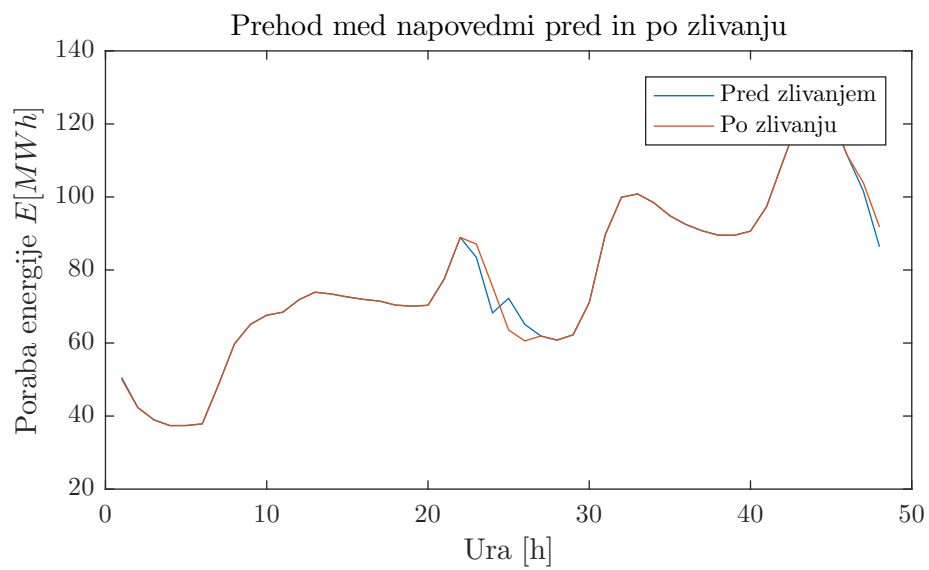
$$\Delta_-(l) = \frac{S + 1 - l}{2S + 1} \quad (3.75)$$

$$\Delta_+(l) = \frac{S + l - K}{2S + 1} \quad (3.76)$$

$$f_{\text{prel}}(x_l) = \begin{cases} (1 - \Delta_-(l))x_l(k) + \Delta_-(l) x_l(k - 1) & l \leq S \\ x_l(k) & S < l \leq (K - S) \\ (1 - \Delta_+(l))x_l(k) + \Delta_+(l) x_l(k + 1) & l > (K - S) \end{cases} \quad (3.77)$$

$\Delta_-$  ter  $\Delta_+$  sta uteži prejšnjega oz. naslednjega vzorca,  $K$  je število vhodnih spremenljivk, s parametrom  $S \in \{0, 1, 2, \dots, \lfloor \frac{K}{2} \rfloor\}$  pa določimo, koliko elementov izhodnega vektorja bo zlitih na začetku oz. na koncu vektorja napovedi.

S predlaganim zlivanjem rešimo problem nezveznosti med sosednimi napovedmi, je tudi enostavno za realizacijo, vpliv na celotno napako pa je precej majhen (slika 3.11). Linearno zlivanje sicer ni optimalno, saj ni zvezno odvedljivo.



Slika 3.11: Graf primerjave izhoda modela pri prehodu med napovedmi pred in po zlivanjem

V kolikor pa želimo imeti tudi zvezno odvedljivost, lahko za uteži uporabimo eksponentne funkcije.





## 4 Model napovedovanja

V poglavju je predstavljen model, ki je bil uporabljen za napovedovanje porabe električne energije za naslednji dan.

Najprej so v podpoglavju 4.1 predstavljeni uporabljeni podatki pri modeliranju, kot so informacije o učni ter validacijski množici ter o vseh in izhodih modela. Nato bo tekom poglavja 4.2 predstavljen način razdelitve napovedovanja na 3 glavne podprobleme: napovedovanje povprečja, standardne deviacije ter oblike. Model vsakega izmed teh podproblemov je predstavljen v svojem podpoglavju z istim imenom. Na koncu je predstavljen še popravek napovedi pri praznikih v poglavju 4.7.

### 4.1 Uporabljeni podatki

V učno množico so izbrani vzorci iz let 2010 ter 2011, validacija pa se izvaja na vzorcih iz leta 2012. Vzorci iz leta 2013 so izpuščeni, saj manjkajo podatki o porabi električne energije in zato nam ne koristijo ne pri gradnji, ne pri evalvaciji modela. Izpuščeni so tudi vzorci nedelj tekom katerih se zamakne ura, saj podatki niso skladni z ostalimi (23 oz. 25 podatkov v vzorcu namesto 24).

Izhod modela je vektor  $\hat{\mathbf{y}} = [\hat{E}_1(k), \dots, \hat{E}_l(k), \dots, \hat{E}_{24}(k)]^T$ , kjer  $\hat{E}_l(k)$  predstavlja oceno porabe v  $l$ -ti uri  $k$ -tega dneva. Za učenje modela uporabimo izmerjene podatke o porabi v  $k$ -tem dnevu  $\mathbf{E}(k) = [E_1(k), \dots, E_l(k), \dots, E_{24}(k)]^T$ .

Vhodi modela bodo v praksi napovedane vremenske (temperatura, dež, veter

ipd.) za naslednji dan in spremenljivko dneva. Vhodi so tudi pretekli podatki našteti v poglavju 2.1. Ker pa podatkov o vremenski napovedi nimamo, smo namesto napovedi privzeli dejanske izmerjene vremenske podatke tistega dne, tako da lahko v realnosti pričakujemo slabše rezultate, saj nam vsaka napoved vnaša v model dodatno napako.

## 4.2 Razdelitev problema in struktura modela

Problem kratkoročnega napovedovanja porabe električne energije je kompleksen proces, saj ima več vhodov in več izhodov in je nelinearen. Problema se zato lotimo tako, da ga najprej razdelimo na več manjših ter enostavnejših podproblemov, nato vsakega za sebe rešimo in načrtamo svoj model. Izhod celotnega modela nato sestavimo iz prispevkov modelov podproblemov.

Problem napovedovanja porabe električne energije smo tako razdelili na 3 podprobleme:

- napovedovanje povprečne porabe v vzorcu  $\bar{y}$  (op. a. tekom dneva),
- napovedovanje standardne deviacije porabe v vzorcu  $\sigma$ ,
- napovedovanje oblike porabe  $\underline{y}$ .

Vsak omenjen problem smo rešili s kombinacijo dveh modelov – splošnega modela  $(\hat{y}^*, \underline{\hat{y}}^*, \hat{\sigma}^*)$ , načrtanega po principu modela TS z rekurzivnim rojenjem GK in rekurzivno metodo WLSE, ter model praznikov  $(\hat{y}^p, \underline{\hat{y}}^p, \hat{\sigma}^p)$ , ki modelira napako splošnega modela pri praznikih. Zgrajen je z LSE metodo. Izhod prazničnega modela upoštevamo le na praznike, torej ko ima  $k$ -ti dan vrednost  $P = 1$ .

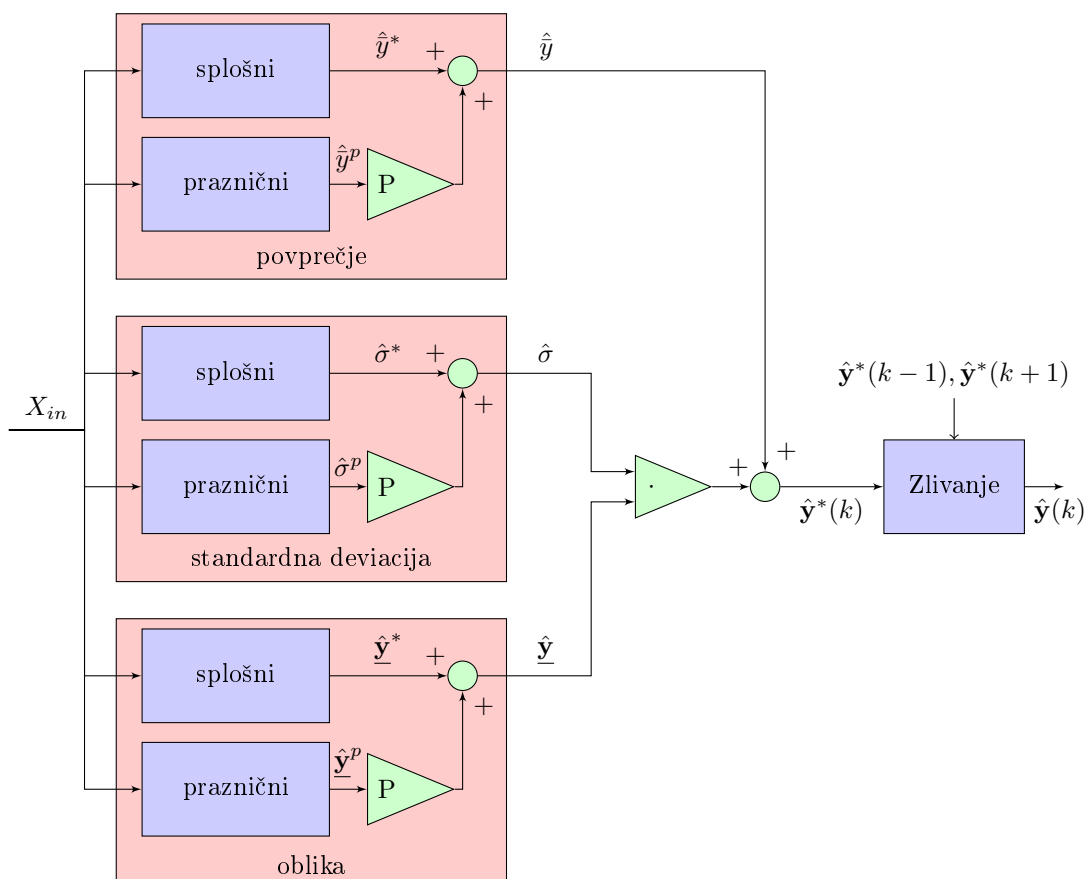
Izhode vseh omenjenih modelov združimo z enačbo (4.1) – napovedano standardno deviacijo zmnožimo z napovedanimi komponentami vektorja oblike, nato pa vsem komponentam prištejemo še napovedano povprečno porabo. Ker pa lahko pride do nezveznosti pri stiku vzorcev (med zadnjo uro vzorca ter prvo uro

naslednjega), na vrednosti  $\hat{\mathbf{y}}^* = [\hat{y}_1^* \dots \hat{y}_{24}^*]^T$  napravimo še zlivanje napovedi (4.2) opisano v poglavju 3.2.3.

$$\hat{\mathbf{y}}^* = ((\underbrace{\hat{\mathbf{y}}^*}_{\hat{\mathbf{y}}^*} + \mathbf{P}\hat{\mathbf{y}}^p) \cdot (\underbrace{\hat{\sigma}^*}_{\hat{\sigma}} + \mathbf{P}\hat{\sigma}^p)) + (\underbrace{\hat{\mathbf{y}}^*}_{\hat{\mathbf{y}}^*} + \mathbf{P}\hat{\mathbf{y}}^p) \quad (4.1)$$

$$\hat{y}_l = f_{pret}(\hat{y}_l^*), \quad \forall l \quad (4.2)$$

Celotna struktura modela kratkoročnega napovedovanja porabe je prikazana na sliki 4.1. V sledečih poglavjih bodo posamezni modeli podrobno predstavljeni.



Slika 4.1: Blokovni diagram zgrajenega modela. Z vijolično so predstavljeni zgrajeni modeli, z zeleno matematične operacije potrebne za povezavo med modeli, rdeča pa označuje področja že omenjenih podproblemov.

### 4.3 Splošni model povprečne porabe

S splošnim modelom povprečne porabe rešimo podproblem napovedovanja povprečne porabe  $\bar{y}$ , izračunana po enačbi (4.3). Povprečna poraba  $\bar{y}$  je tudi izhod, ki ga želimo napovedati z omenjenim modelom.

$$\bar{y}^*(k) = \bar{y}(k) = \frac{1}{24} \sum_{l=1}^{24} E_l(k) \quad (4.3)$$

Zaradi močne korelacije iz prejšnjega dne, prikazane na sliki 2.7, predvidevamo, da je povprečna poraba prejšnjega dne  $\bar{y}(k-1)$  najbolj zanesljiv podatek za napovedovanje. Zato smo se odločili, da napovedujemo razliko povprečne porabe  $\Delta\bar{y}(k) = \bar{y}(k) - \bar{y}(k-1)$ , ki jo nato prištejemo izmerjeni povprečni porabi prejšnjega dne  $\bar{y}(k-1)$ . Seštevek (4.4) vzamemo kot napoved povprečne porabe za naslednji dan.

$$\hat{\bar{y}}(k) = \bar{y}(k-1) + \Delta\hat{\bar{y}}(k) \quad (4.4)$$

Za poenostavitev modela napovedujemo direktno  $\bar{y}(k)$ ,  $\bar{y}(k-1)$  pa definiramo kot eno izmed vhodnih spremenljivk, saj pričakujemo, da bo njegov linearni koeficient  $\theta$  približno 1, in s tem se bo model razvil v enačbo (4.4). Za vhodne spremenljivke modela smo tako vzeli podatek o povprečni porabi za prejšnji dan  $\bar{y}(k-1)$  ter za napovedovanje razlike povprečne porabe še spremembo povprečij vremenskih podatkov (veter, sevanje, temperatura, dež) in spremembo spremenljivke dneva  $\Delta D(k)$ .

Vhodne spremenljivke:

- sprememba povprečja temperature  $\Delta\bar{T}(k) = \bar{T}(k) - \bar{T}(k-1)$ ,
- sprememba povprečja sevanja  $\Delta\bar{\Gamma}(k) = \bar{\Gamma}(k) - \bar{\Gamma}(k-1)$ ,
- sprememba povprečja padavin  $\Delta\bar{R}(k) = \bar{R}(k) - \bar{R}(k-1)$ ,
- sprememba povprečja hitrosti vetra  $\Delta\bar{W}(k) = \bar{W}(k) - \bar{W}(k-1)$ ,

- sprememba spremenljivke dneva  $\Delta D(k) = D(k) - D(k-1)$ ,
- povprečna poraba prejšnjega dne  $\bar{y}(k-1)$ .

Splošni model je tipa TS, ki ga računamo z rekurzivnim rojenjem GK z omejitvami kovariančne matrike ter rekurzivno metodo WLSE. Za razdelitev prostora particioniranja  $A_p$  na roje smo izbrali sledeče spremenljivke:

- povprečna temperatura  $\bar{T}(k)$ ,
- povprečna poraba prejšnjega dne  $\bar{y}(k-1)$ ,
- sprememba spremenljivke dneva  $\Delta D(k) = D(k) - D(k-1)$ ,
- povprečna hitrost vetra  $\bar{W}(k)$ .

Inicializacijo rekurzivnega rojenja GK smo naredili z nerekurzivnim rojenjem GK na učni množici. Parametri tako rekurzivnega kot nerekurzivnega rojenja so naslednji:

<b>Simbol izhoda modela</b>	—	$\hat{y}^*$
<b>Tip modela</b>	—	Model TS
<b>Število rojev <math>m</math></b>	—	3
<b>Koeficient mehčanja <math>\eta</math></b>	—	1.6
<b>Uglasitveni parameter</b>	—	0.1
$\gamma_F$		
<b>Način računanja pripadnosti za načrtovanje in posodabljanje</b>	—	Združitev pripadnosti v izhodnem prostoru ter pripadnosti v prostoru particioniranja s skrivanjem rojev (Algoritem 7)
<b>Način računanja pripadnosti za napovedovanje</b>	—	Pripadnost v prostoru particioniranja s skrivanjem rojev (Algoritem 5)
<b>Učna množica</b>	—	$P = 0 \wedge Y \in \{2010, 2011\}$

Ter parametri le rekurzivnega rojenja GK:

**Faktor pozabljanja**  $\gamma_c$  — 0.98

Inicializacijo rekurzivne metode WLSE smo naredili z nerekurzivno metodo WLSE na učni množici. Parametri tako rekurzivnega kot ne-rekurzivnega rojenja so naslednji:

**Način računanja uteži**  $\beta_i$  — Opisan v poglavju o metodi WLSE (poglavje 3.1.3.1)

Ter parametri specifični za rekurzivno metodo WLSE:

**Faktor pozabljanja**  $\gamma_\theta$  — 0.99

#### 4.4 Splošni model standardne deviacije porabe

S splošnim modelom standardne deviacije porabe rešimo podproblem napovedovanja standardne deviacije porabe  $\sigma$ , izračunano po enačbi (4.5). Standardna deviacija porabe  $\sigma$  je tudi izhod, ki ga želimo napovedati z omenjenim modelom.

$$\sigma^* = \sigma = \sqrt{\frac{1}{24} \sum_{l=1}^{24} (E_l - \bar{y})^2} \quad (4.5)$$

Zaradi močne korelacije prejšnjega dne, prikazane na sliki 2.8, predvidevamo, da je standardna deviacija porabe prejšnjega dne  $\sigma(k-1)$  najbolj zanesljiv podatek za napovedovanje. Zato smo se odločili, da napovedujemo razliko standardne deviacije porabe  $\Delta\sigma(k) = \sigma(k) - \sigma(k-1)$ , ki jo nato prištejemo izmerjeni standardni deviaciji porabi prejšnjega dne  $\sigma(k-1)$ . Seštevek (4.6) vzamemo kot napoved standardne deviacije porabe za naslednji dan.

$$\hat{\sigma}(k) = \sigma(k-1) +_{\Delta} \hat{\sigma}(k) \quad (4.6)$$

Za poenostavitev modela napovedujemo direktno  $\sigma(k)$ ,  $\sigma(k-1)$  pa definiramo kot eno izmed vhodnih spremenljivk, saj pričakujemo, da bo njegov linearni koeficient  $\theta$  približno 1, in s tem se bo model razvil v enačbo (4.6). Za vhodne spremenljivke modela smo tako vzeli podatek o standardni deviaciji porabe za prejšnji dan  $\sigma(k-1)$  ter za napovedovanje razlike standardne deviacije porabe še spremembo povprečij vremenskih podatkov (sevanje, temperatura, dež) in spremembo spremenljivk dneva  $\Delta D(k)$ . Za vhodno spremenljivko smo vzeli tudi povprečno porabo prejšnjega dne dneva.

Vhodne spremenljivke:

- sprememba povprečja temperature  $\Delta \bar{T}(k) = \bar{T}(k) - \bar{T}(k-1)$ ,
- sprememba povprečja sevanja  $\Delta \bar{\Gamma}(k) = \bar{\Gamma}(k) - \bar{\Gamma}(k-1)$ ,
- sprememba povprečja padavin  $\Delta \bar{R}(k) = \bar{R}(k) - \bar{R}(k-1)$ ,
- sprememba spremenljivke dneva  $\Delta D(k) = D(k) - D(k-1)$ ,
- povprečna poraba prejšnjega dne  $\bar{y}(k-1)$ ,
- standardna deviacija porabe prejšnjega dne  $\sigma(k-1)$ .

Splošni model je tipa TS, ki ga računamo z rekurzivnim rojenjem GK z omejitvami kovariančne matrike ter rekurzivno metodo WLSE. Za razdelitev na modele smo izbrali sledeče spremenljivke prostora particioniranja:

- povprečna temperatura  $\bar{T}(k)$ ,
- napoved povprečne porabe  $\hat{y}(k)$ ,
- sprememba spremenljivke dneva  $\Delta D(k) = D(k) - D(k-1)$ ,
- standardna deviacija porabe prejšnjega dne  $\sigma(k-1)$ .

Parametri tako rekurzivnega kot nerekurzivnega rojenja so naslednji:

<b>Simbol izhoda modela</b>	–	$\hat{\sigma}^*$
<b>Tip modela</b>	–	Model TS
<b>Število rojev <math>m</math></b>	–	3
<b>Koeficient mehčanja <math>\eta</math></b>	–	1.6
<b>Uglasitveni parameter</b>	–	0.1
$\gamma_F$		
<b>Način računanja pripadnosti za načrtovanje in posodabljanje</b>	–	Združitev pripadnosti v izhodnem prostoru ter pripadnosti v prostoru particioniranja s skrivanjem rojev in eliminacijo rojev z malim vplivom (Algoritem 7)
<b>Način računanja pripadnosti za napovedovanje</b>	–	Pripadnost v prostoru particioniranja s skrivanjem rojev in eliminacijo rojev z malim vplivom (Algoritem 5)
<b>Učna množica</b>	–	$P = 0 \wedge Y \in \{2010, 2011\}$

Inicializacijo rekurzivne metode WLSE smo naredili z nerekurzivno metodo WLSE na učni množici. Parametri tako rekurzivnega kot ne-rekurzivnega rojenja so naslednji:

<b>Način računanja uteži <math>\beta_i</math></b>	–	Opisan v poglavju o metodi WLSE (poglavje 3.1.3.1)
---	---	--

Ter parametri specifični za rekurzivno metodo WLSE:

<b>Faktor pozabljanja <math>\gamma_\theta</math></b>	–	0.99
--	---	------

## 4.5 Splošni model oblike porabe

S splošnim modelom oblike porabe rešimo podproblem napovedovanja oblike porabe  $\underline{y}$ , izračunano po enačbi (4.7). Oblika porabe  $\underline{y}$  je tudi izhod, ki ga želimo napovedati z omenjenim modelom.



$$\underline{\mathbf{y}}^* = \underline{\mathbf{y}}(k) = \frac{\mathbf{E}(k) - \bar{y}(k)}{\sigma(k)} \quad (4.7)$$

Pri obliki porabe smo za vhodne spremenljivke uporabili povprečja vremenskih podatkov, spremenljivko dneva  $D(k)$  ter napoved povprečja porabe  $\hat{y}(k)$ . Za tako izbiro smo se odločili zaradi narave problema oblike, saj oblika ni linearno odvisna od spremenljivk, se pa v odvisnosti od njih vseeno spreminja.

Vhodne spremenljivke:

- povprečna temperatura  $\bar{T}(k)$ ,
- povprečno sevanje  $\bar{\Gamma}(k)$ ,
- povprečne padavine  $\bar{R}(k)$ ,
- povprečna hitrost vetra  $\bar{W}(k)$ ,
- spremenljivka dneva  $D(k)$ ,
- povprečna poraba prejšnjega dne  $\bar{y}(k-1)$ .

Splošni model je tipa TS, ki ga računamo z rekurzivnim rojenjem GK z omejitvami kovariančne matrike ter rekurzivno metodo WLSE. Za razdelitev na modele smo izbrali sledeče spremenljivke prostora particioniranja:

- povprečna temperatura  $\bar{T}(k)$ ,
- povprečna poraba prejšnjega dne  $\bar{y}(k-1)$ ,
- sprememba spremenljivke dneva  $\Delta D(k) = D(k) - D(k-1)$ ,
- povprečna hitrost vetra  $\bar{W}(k)$ .

Inicializacijo rekurzivnega rojenja GK smo naredili z nerekurzivnim rojenjem GK na učni množici. Parametri tako rekurzivnega kot nerekurzivnega rojenja so naslednji:

---

<b>Simbol izhoda modela</b>	–	$\hat{\underline{y}}^*$
<b>Tip modela</b>	–	Model TS
<b>Število rojev <math>m</math></b>	–	3
<b>Koeficient mehčanja <math>\eta</math></b>	–	1.6
<b>Uglasitveni parameter</b>	–	0.1
$\gamma_F$		
<b>Način računanja pripadnosti za načrtovanje in posodabljanje</b>	–	Združitev pripadnosti v izhodnem prostoru ter pripadnosti v prostoru particioniranja s skrivanjem rojev in eliminacijo rojev z malim vplivom (Algoritem 7)
<b>Način računanja pripadnosti za napovedovanje</b>	–	Pripadnost v prostoru particioniranja s skrivanjem rojev in eliminacijo rojev z malim vplivom (Algoritem 5)
<b>Učna množica</b>	–	$P = 0 \wedge Y \in \{2010, 2011\}$

Ter parametri le rekurzivnega rojenja GK:

**Faktor pozabljanja  $\gamma_c$**  – 0.98

Inicializacijo rekurzivne metode WLSE smo naredili z nerekurzivno metodo WLSE na učni množici. Parametri tako rekurzivnega kot nerekurzivnega rojenja so naslednji:

**Način računanja uteži  $\beta_i$**  – Opisan v poglavju o metodi WLSE (poglavje 3.1.3.1)

Ter parametri specifični za rekurzivno metodo WLSE:

**Faktor pozabljanja  $\gamma_\theta$**  – 0.99

## 4.6 Modeli praznikov

Za podprobleme napovedovanja komponent porabe ( $\bar{y}$ ,  $\sigma$  ter  $\underline{y}$ ) smo opisali načrtane splošne modele, ki dobro opisujejo porabo tekom delavnikov ter vikendov, vendar pri napovedovanju porabe tekom praznikov model odpove. Ko smo analizirali porabo tekom praznikov, smo videli, da ima večina popolnoma drugačne profile, tako v primerjavi z delavniki oziroma prazniki kot tudi med seboj.

Za popravek spet razdelimo problem na tri dele – napovedovanje napak za povprečje, standardno deviacijo ter obliko, s to razliko, da napovedujemo napako med izhodom splošnega modela ter izmerjenim izhodom (enačbe (4.8), (4.9) ter (4.10))

$$\bar{y}^p = \bar{y} - \hat{y}^* \quad (4.8)$$

$$\sigma^p = \sigma - \hat{\sigma}^* \quad (4.9)$$

$$\underline{y}^p = \underline{y} - \underline{\hat{y}}^* \quad (4.10)$$

Zato smo tekom magistrske naloge za vsak praznik naredili svoj linearni model z metodo LSE. Ker pa imamo za vsak praznik le 2 vzorca v učni množici (za leti 2010 ter 2011), so vse linearne povezave z vhodnimi parametri statistično popolnoma nezanesljive. Zato v vhod linearnega modela ne peljemo nobene spremenljivke  $\mathbf{x} = [1]$  in tako izhod modela postane le povprečje prejšnjih vzorcev (poglavje 3.1.3.1). Kljub izredno majhni učni množici model prinese izboljšanje, saj vremenske vplive napovemo že s splošnim modelom, tako da v pogrešku splošnega modela prevladuje napaka zaradi specifičnosti praznika, ki pa jo lahko približno zajamemo že z dvema vzorcema. Seveda bomo v primeru obratovanja modela skozi daljše časovno obdobje imeli tudi večjo učno množico in posledično boljšo napoved.

Parametri specifični za linearne modele za  $i$ -ti praznik na dan  $G_i$  in mesec  $M_i$

<b>Tip modela</b>	– Linearni model
<b>Učna množica</b>	– $P = 1 \wedge G = G_i \wedge M = M_i$

Za praznike, ki niso vsako leto na isti datum (npr. velika noč) modelov tekom magistrskega dela ne naredimo.

Izhod prazničnega modela povprečne porabe označimo z  $\hat{y}^p$ , prazničnega modela standardne deviacije porabe z  $\hat{\sigma}^p$  ter prazničnega modela oblike porabe z  $\underline{\hat{y}}^p$

## 4.7 Model zlivanja napovedi

Ker napovedujemo za 24 ur hkrati (za vzorec 1 dneva), obstaja verjetnost, da je skok med porabo v zadnji uri nekega vzorca ter prvi uri naslednjega tako velika, da se v realnosti ne more zgoditi. Za zgladitev teh nezveznosti po združitvi vseh splošnih ter prazničnih modelov (napoved  $\hat{\mathbf{y}}^*$ ) napravimo še zlivanje napovedi trenutne ter naslednje napovedi (poglavje 3.2.3).

Parametri za model zlivanja:

<b>Tip modela</b>	– Model zlivanja napovedi
<b>Število zlitih vzorcev <math>S</math></b>	– 2

## 5 Rezultati in diskusija

Poglavje je namenjeno prikazu rezultatov načrtanega modela. Najprej je v poglavju 5.1 predstavljena napaka MAPD, s katero ocenjujemo uspešnost modelov. Nato so v poglavju 5.2 podani rezultati napovedovanja z razvitim modelom iz različnih zornih kotov. Te nato v naslednjem poglavju primerjamo še z ostalimi metodami. Na koncu povzamemo zaključke modeliranja v poglavju 5.4 ter izpostavimo probleme uporabljene metodologije in možne rešitve v poglavju 5.5.

### 5.1 Način ocenjevanja modela

Napaka  $e(k)$   $k$ -tega vzorca med izmerjeno  $\mathbf{y}(k)$  in ocenjeno  $\hat{\mathbf{y}}(k)$  porabo je izračunana kot vsota absolutne vrednosti razlike med urnimi komponentami porabe (5.1).

$$e(k) = \sum_{l=1}^n |y_l(k) - \hat{y}_l(k)| \quad , \quad (5.1)$$

kjer je  $n$  število elementov izhodnega vektorja. Glavni kazalec kvalitete modela je procent povprečne absolutne napake ali krajše MAPD (eng. Mean Absolute Percent Deviation), izračunanega z enačbo (5.2). MAPD smo uporabili zaradi finančnega vidika. Dodatni strošek je podoben, če napoved zgreši v  $+$  ali v  $-$ , hkrati pa strošek z absolutno vrednostjo linearno narašča. Kazalec uporabimo tudi pri vrednotenju napovedi komponent porabe  $(\hat{\underline{y}}, \hat{\sigma}, \hat{\underline{y}})$ .

$$\text{MAPD}[\%] = 100 * \frac{\sum_{k=1}^N |e(k)|}{\sum_{k=1}^N \sum_{l=1}^n |y_l(k)|} \quad (5.2)$$

## 5.2 Rezultati modela

Validacija je narejena na podatkih iz leta 2012 – skupaj 365 dni (en dodaten dan zaradi prestopnega leta ter en dan manj zaradi prestavitve ure).

Napake MAPD po komponentah modela so zbrane v tabeli 5.1. Komponente so združene po tipu modela. Skupna napaka MAPD je 3.66 %.

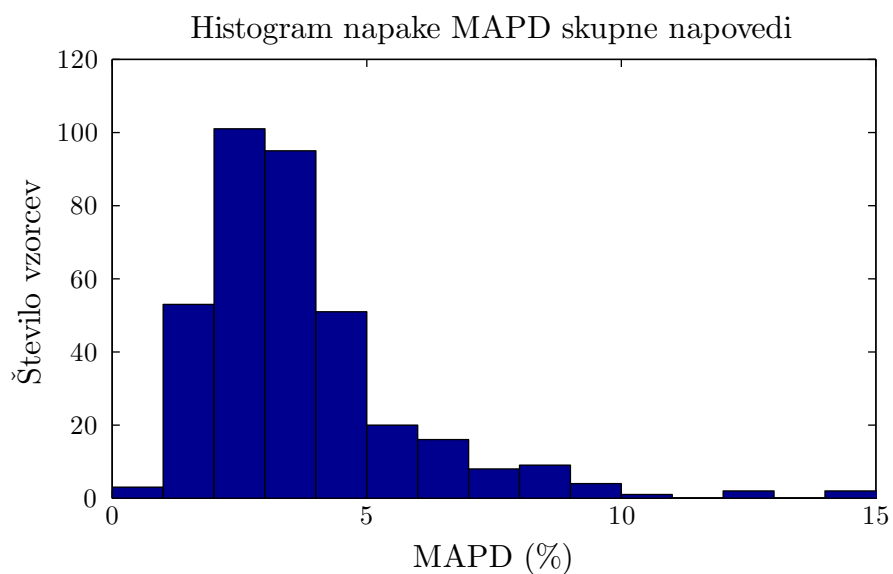
$$\begin{aligned}\sum_{k=1}^N \mathbf{y}(k) &= 572.2 \text{ GWh} \\ \sum_{k=1}^N e(k) &= 21.1 \text{ GWh} \\ \text{MAPD} &= 3.66 \text{ \%}\end{aligned}$$

	Model	MAPD [%]
Splošni modeli	$\hat{\mathbf{y}}^*$	2.31
	$\underline{\hat{\mathbf{y}}}^*$	13.73
	$\hat{\sigma}^*$	6.23
Skupni model (splošni + praznični)	$\hat{\mathbf{y}}$	2.25
	$\underline{\hat{\mathbf{y}}}$	13.45
	$\hat{\sigma}$	6.03
Skupna napoved	$\hat{\mathbf{y}}^*$	3.69
<b>Skupna napoved z zlivanjem</b>	$\hat{\mathbf{y}}$	<b>3.66</b>

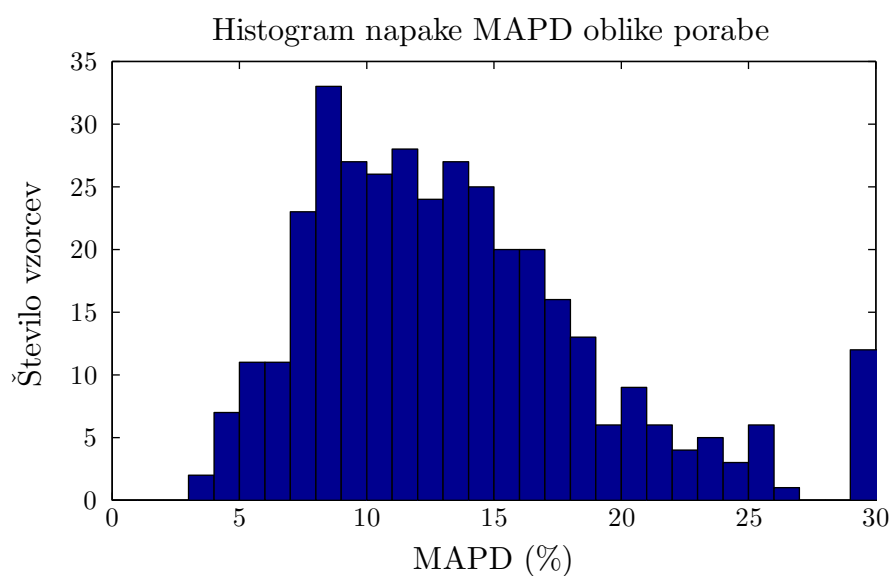
Tabela 5.1: Tabela prikazuje napako MAPD [%] različnih delov modela

Oglejmo si histograme napak napovedi skupne porabe, povprečne porabe ter oblike porabe za validacijsko leto 2012 na slikah 5.1, 5.3 ter 5.2. Na histogramu 5.1 se lepo vidi povprečje napake okoli 3.5 %. Le 5 napak skupne porabe manjših od 1% je posledica slabše napovedi oblike (slika 5.2). Vidimo tudi, da oblike nikoli ne napovemo točno. Razlog bi lahko našli v tem, da napovedujemo profil dneva

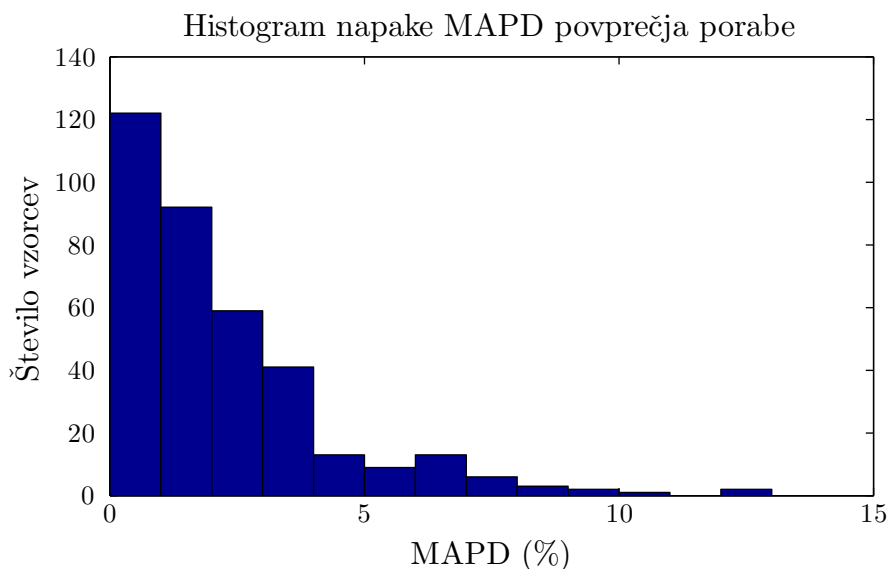
skupaj namesto za vsako uro posebej. Na histogramu povprečja pa vidimo, da ima več kot polovica napovedi napako med 0 % ter 2 % (skupni MAPD povprečne porabe je 2,25 %), kar pomeni, da smo za večino dni dobro napovedali.



Slika 5.1: Histogram napake končne ocene



Slika 5.2: Histogram napake skupne napovedi oblike



Slika 5.3: Histogram napake skupne napovedi povprečne porabe

Primerjava napovedi glede na spremenljivko dneva  $D$  je navedena v tabeli 5.2. Po pričakovanjih je napoved za posebne dneve ( $D > 0.5$ ) občutno slabša ( $\text{MAPD} = 4.09\%$  posebnih dni proti  $\text{MAPD} = 3.43\%$  delavnikov), je pa ta razlika zaradi vpeljave posebnih prazničnih modelov manjša kot bi bila drugače.

V tabeli 5.3 so zbrani rezultati napovedi po dnevih v tednu glede na korak algoritma. Najboljše rezultate dobimo v sredo, ki ji sledita napovedi za četrtek in torek. To ne preseneča, saj se v sredini tedna profil porabe najbolj stabilizira, ker izginje vpliv vikenda (nimamo "podaljšanih vikendašev", petkovih večerov in podobno). Presenetljivo pa vidimo, da je napaka skupne napovedi z zlitjem največja tudi v ponedeljek in ne samo med vikendi, kot smo predvidevali pri pregledu podatkov. Tam največji del prispeva napaka napovedovanja povprečne vrednosti. Problem pri napovedovanju povprečja za ponedeljek (in podobno tudi za soboto, kjer imamo približno enako napako) je vhodna spremenljivka povprečne porabe prejšnjega dne, ki je pa precej različna od porabe za napoved (nedelja  $\rightarrow$  ponedeljek, petek  $\rightarrow$  sobota). To sicer omilimo z vpeljavo spremembe spremenljivke dneva  $\Delta D$  v vhod modela, ampak precejšen del napake še vedno ostane.



	Korak	D > 0.5	D < 0.5
Splošni modeli	$\hat{\underline{y}}^*$	2.43	2.25
	$\underline{\hat{y}}^*$	14.67	13.22
	$\hat{\sigma}^*$	7.82	5.28
Skupni model (splošni + praznični)	$\hat{\underline{y}}$	2.25	2.25
	$\underline{\hat{y}}$	13.92	13.19
	$\hat{\sigma}$	7.28	5.28
Skupna napoved	$\hat{\underline{y}}^*$	4.11	3.46
<b>Skupna napoved z zlivanjem</b>	$\hat{\underline{y}}$	<b>4.09</b>	<b>3.43</b>

Tabela 5.2: Primerjava napake napovedi MAPD [%] različnih delov modelov glede na spremenljivko dneva D

Pri napovedi oblike ( $\underline{\mathbf{y}}^*$  ter  $\underline{\mathbf{y}}$ ) vidimo največje odstopanje v petek, ki mu sledita ponedeljek in nedelja. Napaki oblike pri petku in ponedeljku največ prispeva že omenjeni vpliv vikenda (na sliki 2.1), napak za nedeljo pa predvsem narava porabe tekom vikendov, ki se od tedna do tedna razlikuje.

Točnost napovedi se razlikuje tudi glede na letni čas, kar je prikazano v tabeli 5.4. Letni časi so razdeljeni po mesecih: zima zavzema december, januar in februar, pomlad marec, april in maj, poletje junij, julij in avgust, jesen pa september, oktober in november. Model najbolj napove porabo pozimi, tekom ostalih letnih časov pa približno enako.

Zanimivi so rezultati posameznih delov napovedovanja. Vidimo lahko, da je primaren razlog za napako tekom spomladanskih ter jesenskih mesecev napoved oblike porabe  $\underline{\hat{y}}$ , tekom poletnih pa napovedi amplitude ter v manjši meri povprečne porabe. Razlog za slabo napoved oblike v jesenskih ter spomladanskih mesecih je v naravi letnih časov, saj se oblika močno spreminja (tekom teh mesecev ogrevanje zamenja ohlajanje oziroma obratno), model pa zaradi majhnega

	Model	Pon	Tor	Sre	Čet	Pet	Sob	Ned
Splošni modeli	$\hat{\underline{y}}^*$	2.76	2.51	1.98	2.30	2.19	2.28	2.16
	$\underline{\hat{y}}^*$	14.85	12.98	13.21	12.81	14.37	13.04	14.77
	$\hat{\sigma}^*$	9.09	4.98	4.00	4.78	5.58	5.22	9.28
Skupni model (splošni + praznični)	$\hat{\underline{y}}$	2.75	2.50	1.80	2.25	2.17	2.28	1.99
	$\underline{\hat{y}}$	14.64	12.64	13.05	12.92	14.14	12.34	14.38
	$\hat{\sigma}$	8.67	4.73	3.95	4.71	5.59	5.22	8.72
Skupna napoved	$\hat{\underline{y}}^*$	4.16	3.66	3.10	3.35	3.49	3.79	4.18
<b>Skupna napoved z zlitjem</b>	$\hat{\underline{y}}$	4.10	3.64	3.08	3.31	3.49	3.75	4.16

Tabela 5.3: Primerjava napake napovedi MAPD [%] različnih delov modelov glede na dan v tednu

števila rojev trendu oblike ne sledi dovolj hitro. Razlog za poletna odstopanja pa so spremenljive oblike iz tedna v teden, saj je to čas dopustov in je težko napovedati, kdaj bo kdo na počitnicah.

Primerjali smo tudi rezultate glede na število rojev pri napovedovanju povprečne porabe  $\hat{\underline{y}}$  (tabela 5.5) ter oblike  $\underline{\hat{y}}$  (tabela 5.6)

Najbolj presenetljive rezultate smo dobili, ko smo primerjali napako napovedi oblike pri različnem številu rojev. V tabeli 5.6 lahko opazimo, da manj kot imamo rojev, boljše rezultate dobimo, kar pa ne sovpada z velikim številom različnih oblik v podatkih, ki bi morali povzročiti potrebo po večih rojih. Ta rezultat nam razkrije dejstvo, da se oblika s časom močno spreminja in le s hitro adaptacijo lahko sledi spremembam oblike porabe. Sicer smo prepričani, da obstajajo možnosti izboljšave za napovedovanje tekom pomladi ter jeseni. Zato predlagamo, da se razišče boljše načine particioniranja prostora, saj se oblike porabe tekom omenjenih letnih časov lahko napove boljše.

Na slikah 5.4 ter 5.5 sta prikazana primera zelo dobre ter zelo slabe napo-

	Model	Zima	Pomlad	Poletje	Jesen
Splošni modeli	$\hat{y}^*$	2.08	2.49	2.41	2.38
	$\underline{\hat{y}}^*$	11.64	15.38	12.29	15.84
	$\hat{\sigma}^*$	6.03	6.42	6.10	6.44
Skupni model (splošni + praznični)	$\hat{y}$	1.98	2.49	2.38	2.27
	$\underline{\hat{y}}$	11.00	15.05	12.15	15.84
	$\hat{\sigma}$	5.36	6.64	6.04	6.33
Skupna napoved	$\hat{y}^*$	3.09	4.11	3.68	4.18
<b>Skupna napoved z zlivanjem</b>	$\hat{y}$	3.07	4.07	3.65	4.15

Tabela 5.4: Primerjava napake napovedi različnih delov modelov glede na letni čas

Napovedovanje povprečja $\hat{y}^*$		
Število rojev	$\gamma_\theta$	MAPD[%]
1	0.97	2.71
1	0.99	2.62
2	0.97	2.61
2	0.99	2.48
3	0.97	2.43
3	0.99	2.30
4	0.97	2.45
4	0.99	2.29

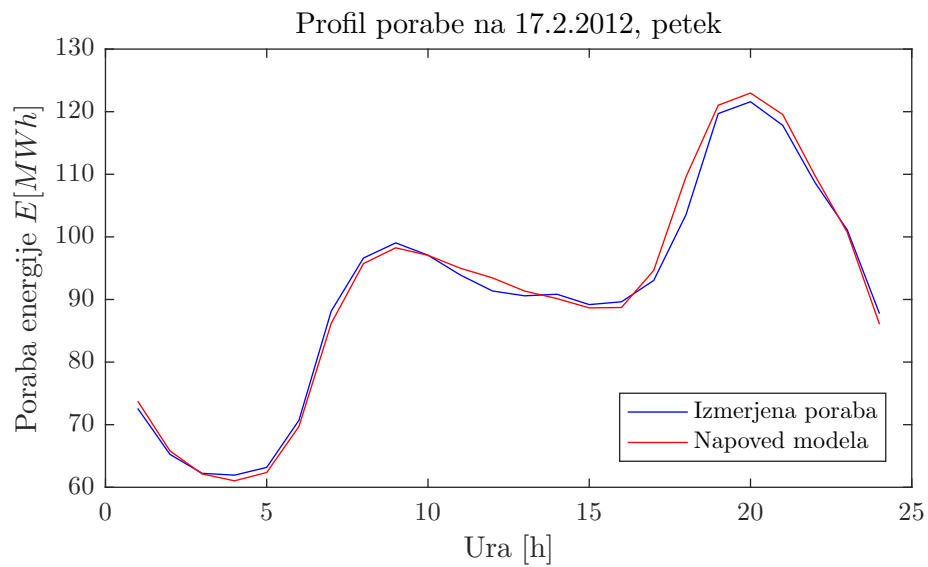
Tabela 5.5: Primerjava MAPD pri napovedovanju povprečja porabe glede na število rojev ter koeficienta pozabljanja

vedi. V primeru zelo slabe napovedi vidimo, da precejšen del napake prispeva napaka v napovedi povprečja porabe, kar drži tudi pri ostalih zelo slabih napovedih ( $\text{MAPD} > 8\%$ ). Bližje kot je MAPD vzorca povprečju (4%), bolj so si vplivi

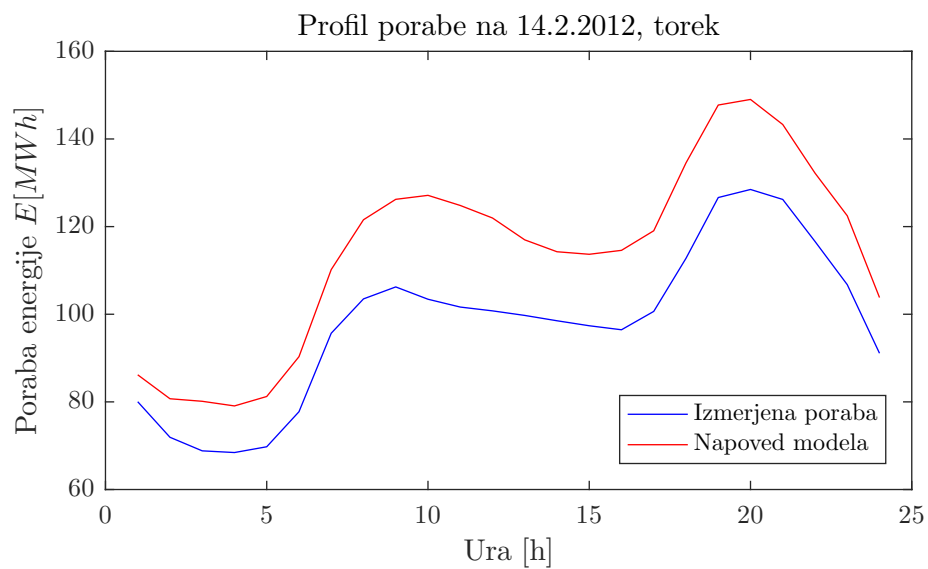
Napovedovanje oblike $\hat{\underline{y}}^*$	
Število rojev	MAPD[%]
1	13.43
2	13.73
3	13.92
4	14.00

Tabela 5.6: Primerjava MAPD pri napovedovanju oblike porabe glede na število rojev

napačne napovedi oblike, amplitude ter povprečja porabe enakovredni.



Slika 5.4: Profil porabe na 17. 2. 2012 ter napoved modela kot primer zelo dobre napovedi porabe



Slika 5.5: Profil porabe na 14. 2. 2012 ter napoved modela kot primer zelo slabe napovedi porabe

### 5.3 Primerjava z obstoječimi metodami

Algoritem smo primerjali z metodama LR (poglavje 3.2.1) ter RLR (poglavje 3.2.1), predstavljenima v [1]. V [1] je poudarek o napovedovanju za vsako regijo posebej, kar je upoštevano tudi v magistrski nalogi, saj gradimo model na podatkih iz JZ regije. Primerjava kazalcev MAPD je razvidna v tabeli 5.7. Model razvit v magistrskem delu napove porabo občutno bolje kot modela iz literature.

Metoda	MAPD
Nadgrajeni TS model z GK rojenjem	3.66
LR	5.22
RLR	5.20

Tabela 5.7: Primerjava napak MAPD različnih metod napovedovanja porabe

Ker načrtana metoda porabo razdeli na 3 dele (povprečna poraba  $\bar{y}$ , oblika  $\underline{y}$  ter standardna deviacija  $\sigma$ ), razdelimo tako tudi napovedi ostalih metod, da tako pridobimo možnost primerjave metod tudi po komponentah. Rezultati primerjav so prikazani v tabeli 5.8. Vidimo, da razvita metoda mnogo bolje napove vse komponente porabe, komponento povprečja porabe celo za skoraj 2-krat.

Metoda	MAPD $\bar{y}$	MAPD $\underline{y}$	MAPD $\sigma$
Nadgrajeni TS model z GK rojenjem	2.25	13.45	6.03
LR	4.34	14.04	6.43
RLR	4.33	14.06	6.27

Tabela 5.8: Primerjava napak napovedi MAPD različnih metod pri različnih komponentah porabe

Oglejmo si še rezultate metod glede na dneve v tednu v tabeli 5.9. Vidimo, da razvita metoda bolje napove porabo ne glede na dan. Predvsem je napoved

boljša za ponedeljke, torke ter srede, kjer v vhodnih spremenljivkah metod LR ter RLR nastopa poraba preko vikenda. Ker imamo tekom vikenda večji raztros porabe, se ta preko modela še multiplicira in tako pride do omenjenih odstopanj.

Korak	Pon	Tor	Sre	Čet	Pet	Sob	Ned
Nadgrajeni TS model z GK rojenjem	4.10	3.64	3.08	3.31	3.49	3.75	4.16
LR	5.68	6.64	5.16	4.40	4.89	4.92	4.92
RLR	5.68	6.55	5.19	4.35	4.92	4.89	4.89

Tabela 5.9: Primerjava napak napovedi MAPD različnih metod pri različnih dnevih v tednu

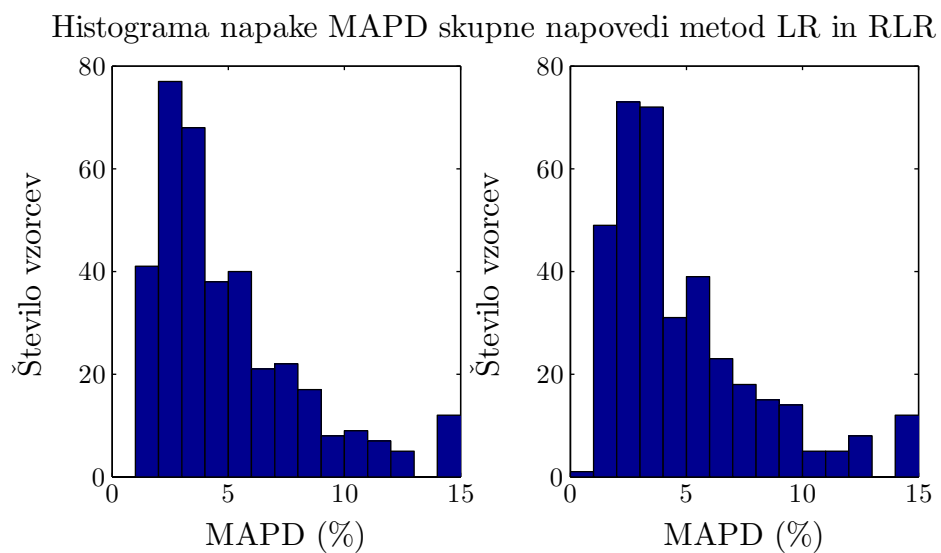
Najbolje se metodi LR ter RLR izkažeta, ko razdelimo rezultate po letnih časih, kjer za jesen napovedujeta približno tako dobro kot razvit algoritem. Močno pa zgrešita pri napovedi tekom zime, kjer skoraj 2-krat slabše napovesta porabo.

Razlog za takšno odstopanje je potrebno še raziskati, vendar menimo, da predpostavka o kvadratni odvisnosti porabe od temperature mogoče ni nujno najbolj primerna za modeliranje zimske porabe. Takrat ogrevanje, kot prevladujoči del porabe, teoretično raste linearno z nižanjem temperature (zakon o prevajanju toplote). Seveda linearna odvisnost v praksi zaradi različnih dejavnikov ni nujno resnična, zato puščamo možnost boljših modelov odvisnosti od temperature kot je linearni.

Oglejmo si še histograme napak obeh primerjalnih metod, LR ter RLR (Slike 5.6, 5.7 in 5.8). Vidimo, da imata metodi razporeditve napak med seboj zelo podobne, v primerjavi z razvitim algoritmom pa precej slabše. Predvsem bi izpostavili manjše število vzorcev, za katere napovemo popolnoma napačno obliko ( $\text{MAPD} > 27\%$ ), kar pomeni, da razvita metoda dobro modelira redke oblike.

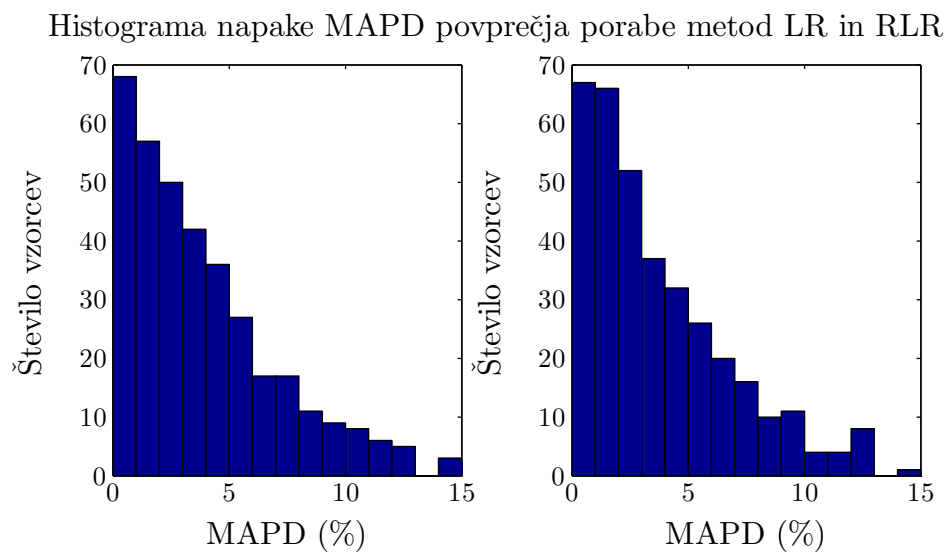
Metoda	Zima	Pomlad	Poletje	Jesen
Nadgrajeni TS model z GK rojenjem	3.07	4.07	3.65	4.15
LR	5.56	6.18	4.46	4.45
RLR	5.65	6.13	4.49	4.25

Tabela 5.10: Primerjava napak napovedi MAPD različnih metod v različnih letnih časih

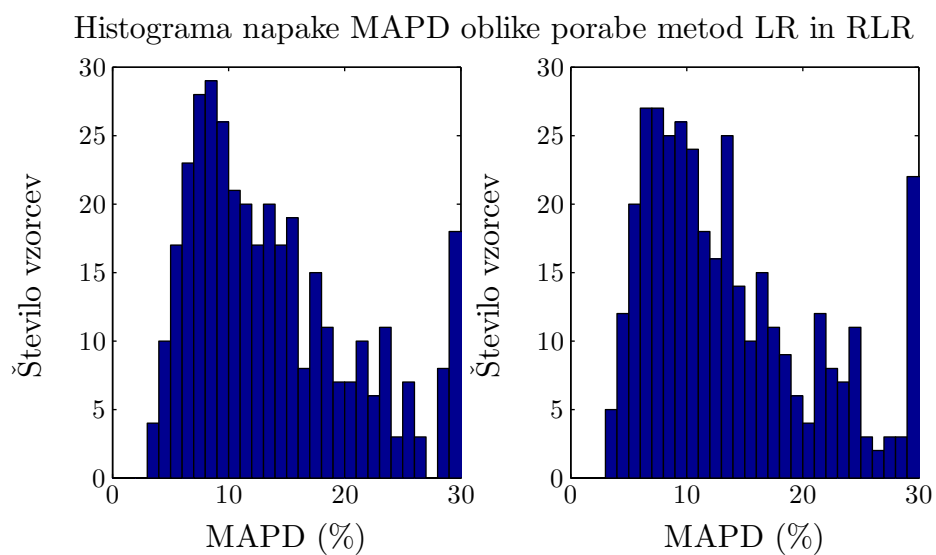


Slika 5.6: Primerjava histogramov napake skupne porabe po metodah LR (levo) ter RLR (desno)





Slika 5.7: Primerjava histogramov napake povprečne porabe po metodah LR (levo) ter RLR (desno)



Slika 5.8: Primerjava histogramov napake oblike porabe po metodah LR (levo) ter RLR (desno)

## 5.4 Diskusija

Mehki modeli so se izkazali za uporabne na področju kratkoročnega napovedovanja porabe električne energije, saj njihovo deljenje na področja sovpadata s problemom različnih profilov za različne dni ali letne čase. Časovno spremenljivost porabe pa dobro opišemo z rekurzivno obliko mehkega modela, saj ga tako sproti prilagajamo glede na tekoče podatke.

Model je relativno glede na povprečno porabo najboljši pozimi, najslabši pa poleti, kar je sicer posledica nespreminjajoče porabe v času zime zaradi ogrevanja.

Največjo napako dobimo pri napovedovanju profila porabe v ponedeljek ter petek. Vzrok za to je v podaljšanih vikendih, ki v profil porabe navadnega delavnika (primer v sredo) doda elemente profila porabe med vikendom. To smo poskusili modelirati z definiranjem spremenljivke dneva  $D$  v petek na vmesno vrednost med 0 in 1, s čimer smo sicer napoved izboljšali, a je vseeno ostala precejšnja napaka.

Še ena slabost metode je precejšnje število parametrov, kar pomeni več empiričnega testiranja za nastavljanje parametrov. Res pa je, da tudi s privzetimi parametri dobimo dobre rezultate.

V primerjavi z ostalimi metodami napovedovanja [1] da napoved z mehki modeli boljše rezultate, tako da lahko predvidevamo, da je metoda primerna za implementacijo v praksi.

## 5.5 Slabosti algoritma

Kljub pozitivni oceni pa ima algoritem nekaj slabosti, ki negativno vplivajo na točnost napovedi. Vse odkrite slabosti so našteje v naslednjih poglavjih, pri vsaki je tudi predlagana razširitev algoritma, ki bi določeno slabost odpravila.

### 5.5.1 Ročna nastavitev spremenljivke dneva

V uporabljeni metodi smo informacijo o dnevu v tednu ročno prevedli v spremenljivko dneva  $D$ , ki smo ga nato uporabili kot vhod v linearne modele. Problem je, ker poraba ni nujno linearno odvisna od spremenljivke dneva  $D$ . Petek ima po enačbi (2.2) vrednost  $D = 0.15$ , delavnik  $D = 0$  ter vikendi  $D = 1$ , poraba v omenjenih dneh ni nujno v enakih razmerjih, kot smo jih določili ročno.

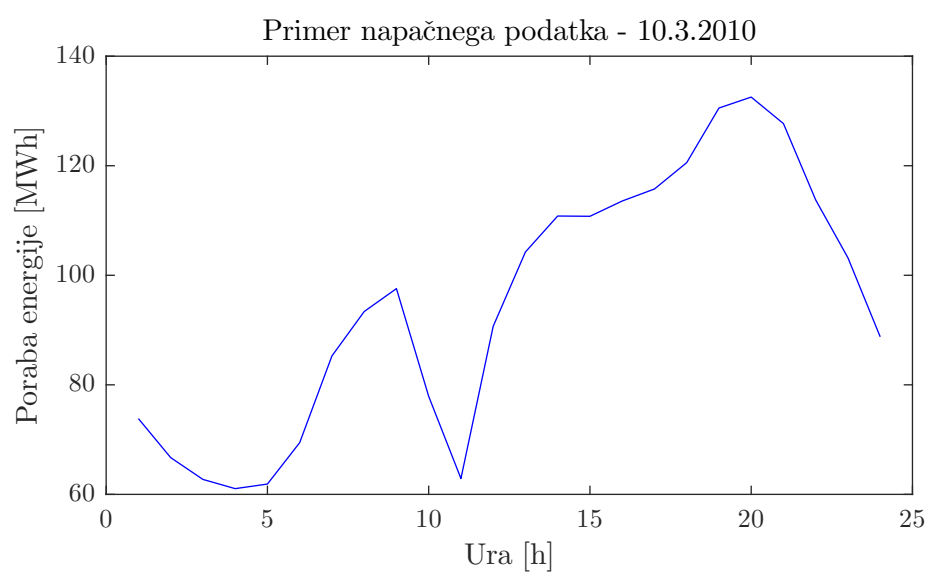
Tu predlagamo razmislek za nadaljnje raziskave, kako v spremenljivko dneva  $D$  avtomatično vpeljati znane informacije o porabi na različni dan, da vrednosti spremenljivke dneva ne iščemo s poskušanjem.

### 5.5.2 Detekcija slabih podatkov le kot del GK algoritma

Detekcija slabih podatkov, katerega primer je prikazan na sliki 5.9, je implementirana direktno v rojenje GK tako, da bolj kot se razlikuje izhod modela pri spremenljivkah določenega vzorca od izmerjenega izhoda istega vzorca, manj prispeva pri gradnji modela. To detekcijo bi lahko razširili že na začetno analizo podatkov, kjer bi z detekcijskimi algoritmi našli napačne podatke in jih odstranili iz učne množice.

### 5.5.3 Vključitev vzorcev s prestavitvijo ure

Vzorci dni, med katerimi se ura prestavi na poletni čas, so pred opisanim algoritmom odstranjeni iz učne množice, saj imamo tisti dan le 23 odčitkov podatkov. Te bi lahko s primernimi primerjalnimi funkcijami med popolnimi vzorci ter vzorci z manjkajočimi podatki vključili v algoritem



Slika 5.9: Primer napačnega podatka - izmerjeno na dan 10.3.2010.

## 6 Zaključek

V magistrskem delu smo se kratkoročnega napovedovanja porabe električne energije lotili z izgradnjo modela TS s pomočjo rekurzivnega rojenja Gustafson-Kessel ter rekurzivne metode uteženih najmanjših kvadratov. Porabo smo napovedovali za celoten dan hkrati. Napoved dnevnega poteka porabe smo razdelili na 3 komponente: povprečno porabo v dnevu, obliko porabe ter standardno deviacijo porabe. Za napovedovanje vsake komponente smo načrtali svoj model TS z različnimi vhodnimi spremenljivkami in parametri uporabljenih algoritmov.

Za napovedovanje porabe smo predstavili naslednje modifikacije rojenja GK: skrivanje rojev, eliminacija malih vplivov ter kombiniranje pripadnosti. Modifikacije izboljšajo delitev prostora. Modifikacije bi lahko uporabili tudi pri drugih aplikacijah kjer se uporabljajo tehnike rojenja, katerih pozitivni vpliv bi sicer morali še raziskati.

Zaradi specifične porabe tekom praznikov (vsak praznik ima rahlo različen profil porabe), za njih napovedujemo tudi razliko med izmerjeno porabo ter napovedjo omenjenega modela TS. To napoved nato prištejemo napovedi modela TS.

Razvit model TS z rekurzivnim rojenjem GK ter rekurzivno metodo WLSE je zelo uporaben za problem kratkoročnega napovedovanja, saj z roji dobro zajame letne čase oz. vikende, z rekurzivno naravo modela pa se hitro prilagodi na časovne spremembe v porabi.

Predstavljeni algoritem smo primerjali tudi z dvema metodama napovedova-

nja električne energije (linearna regresija ter robustna linearna regresija). Razviti model napove porabo v validacijskem letu z napako  $\text{MAPD} = 3.66 \%$ , s čimer znatno izboljšamo natančnost napovedovanja v primerjavi z ostalima metodama napovedovanja. Model se hitro in dobro odzove na časovne spremembe sistema (npr. povečanje števila naročnikov).

Problem razvitega modela je v prehodnih obdobjih (jesen, pomlad), kjer zaradi majhnega števila rojev pri napovedovanju oblike izgubimo informacije o oblikah v spreminjajočih obdobjih. Sicer se zaradi rekurzivnosti model prilagaja spremembam pri porabi, vendar je to prilagajanje počasno in posledično nam ta zamik prispeva k napaki napovedi. Temu bi se izognili z večjim upoštevanjem vzorcev iz zgodovine.

## Literatura

- [1] Š. Kunstelj, M. Rejc in M. Pantoš, “Kratkoročno napovedovanje porabe električne energije po regijah za območje Slovenije,” *1978 IEEE Conference on Decision and Control including the 17th Symposium on Adaptive Processes*, vol. 4, str. 222–228, 2014.
- [2] D. J. Bartholomew, “Review of time series analysis forecasting and control,” *Operational Research Quarterly (1970–1977)*, vol. 22, no. 2, str. 199–201, 1971.
- [3] M. Y. Cho, J. C. Hwang in C. S. Chen, “Customer short term load forecasting by using arima transfer function model,” v *Energy Management and Power Delivery, 1995. Proceedings of EMPD '95., 1995 International Conference on*, vol. 1, str. 317–322 vol.1, Nov 1995.
- [4] J. Feng, “A new method for ionospheric short-term forecast using similar-day modeling,” v *Antennas, Propagation EM Theory (ISAPE), 2012 10th International Symposium on*, str. 472–474, Oct 2012.
- [5] F. Rodrigues, C. Cardeira in J. Calado, “The daily and hourly energy consumption and load forecasting using artificial neural network method: A case study using a set of 93 households in portugal,” *Energy Procedia*, vol. 62, str. 220–229, 2014. 6th International Conference on Sustainability in Energy and Buildings, SEB-14.
- [6] T. Takagi in M. Sugeno, “Fuzzy identification of systems and its applications to modeling and control,” *IEEE Transactions on Systems, Man, and*

- 
- Cybernetics*, vol. SMC-15, str. 116–132, Jan 1985.
- [7] J. C. Bezdek, R. Ehrlich in W. Full, “Fcm: The fuzzy c-means clustering algorithm,” *Computers & Geosciences*, vol. 10, no. 2, str. 191–203, 1984.
- [8] D. Gustafson in W. Kessel, “Fuzzy clustering with a fuzzy covariance matrix,” *1978 IEEE Conference on Decision and Control including the 17th Symposium on Adaptive Processes*, vol. 17, str. 761–766, 1978.
- [9] D. Dovžan in I. Škrjanc, “Recursive clustering based on a gustafson–kessel algorithm,” *Evolving Systems*, vol. 2, no. 1, str. 15–24, 2010.
- [10] D. Dovžan in I. Škrjanc, “Recursive fuzzy c-means clustering for recursive fuzzy identification of time-varying processes,” *{ISA} Transactions*, vol. 50, no. 2, str. 159–169, 2011.
- [11] S. A. Soroosh in A. Kalhor, “Evolving takagi-sugeno model based on on-line gustafson-kessel algorithm and kernel recursive least square method,” v *Evolving and Adaptive Intelligent Systems (EAIS), 2014 IEEE Conference on*, str. 1–8, June 2014.
- [12] B. Hartmann, O. Banfer, O. Nelles, A. Sodja, L. Teslic in I. Škrjanc, “Supervised hierarchical clustering in fuzzy model identification,” *IEEE Transactions on Fuzzy Systems*, vol. 19, str. 1163–1176, Dec 2011.
- [13] O. Nelles in R. Isermann, “Basis function networks for interpolation of local linear models,” v *Decision and Control, 1996., Proceedings of the 35th IEEE Conference on*, vol. 1, str. 470–475 vol.1, Dec 1996.
- [14] P. Angelov in R. Buswell, “Evolving rule-based models: a tool for intelligent adaptation,” July 2001.
- [15] D. Dovžan, I. Škrjanc in F. Gomide, “Evolving fuzzy-madel-based on c-regression clustering,” v *Evolving and Adaptive Intelligent Systems (EAIS), 2014 IEEE Conference on*, str. 1–7, June 2014.



- 
- [16] J. Jang, C. Sun in E. Mizutani, *Neuro-fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*. MATLAB curriculum series, Prentice Hall, 1997.
  - [17] T. Mahmoud, D. Habibi, O. Bass in S. Lachowics, *Fuzzy Inference System in Energy Demand Prediction*. INTECH Open Access Publisher, 2012.
  - [18] A. Kazemi, A. F. A in M. Hosseinzadeh, “A multi-level fuzzy linear regression model for forecasting industry energy demand of iran,” *Procedia - Social and Behavioral Sciences*, vol. 41, str. 342–348, 2012. The First International Conference on Leadership, Technology and Innovation Management.
  - [19] D. Paravan, “Osnove trga z električno energijo,” 2010.
  - [20] R. Babuška, P. van der Veen in U. Kaymak, “Improved covariance estimation for Gustafson-Kessel clustering,” v *Fuzzy Systems, 2002. FUZZ-IEEE’02. Proceedings of the 2002 IEEE International Conference on*, vol. 2, str. 1081–1085, 2002.
  - [21] R. J. Hathaway in J. C. Bezdek, “Switching regression models and fuzzy clustering,” *IEEE Transactions on Fuzzy Systems*, vol. 1, str. 195–204, Aug 1993.
  - [22] K. J. Åström in B. Wittenmark, “A survey of adaptive control applications,” 1995.
  - [23] K. Åström in B. Wittenmark, *Adaptive Control*. Addison-Wesley series in electrical engineering : control engineering, Addison-Wesley, 1995.