

# Razvoj modela softverskih senzora primjenom "bootstrapping" metode

---

**Cvetnić, Matija**

**Master's thesis / Diplomski rad**

**2015**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **University of Zagreb, Faculty of Chemical Engineering and Technology / Sveučilište u Zagrebu, Fakultet kemijskog inženjerstva i tehnologije**

*Permanent link / Trajna poveznica:* <https://urn.nsk.hr/urn:nbn:hr:149:102823>

*Rights / Prava:* [In copyright](#)/[Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2024-07-13**



*Repository / Repozitorij:*

[Repository of Faculty of Chemical Engineering and Technology University of Zagreb](#)



SVEUČILIŠTE U ZAGREBU  
FAKULTET KEMIJSKOG INŽENJERSTVA I TEHNOLOGIJE  
SVEUČILIŠNI DIPLOMSKI STUDIJ

Matija Cvetnić

DIPLOMSKI RAD

*Zagreb, srpanj 2015.*

SVEUČILIŠTE U ZAGREBU  
FAKULTET KEMIJSKOG INŽENJERSTVA I TEHNOLOGIJE  
SVEUČILIŠNI DIPLOMSKI STUDIJ

Matija Cvetnić

RAZVOJ MODELA SOFTVERSKIH SENZORA PRIMJENOM  
"BOOTSTRAPPING" METODE

DIPLOMSKI RAD

Voditelj rada:

Izv. prof. dr.sc. Nenad Bolf

Članovi ispitnog povjerenstva:

Izv. prof. dr. sc. Nenad Bolf

Doc. dr. sc. Igor Dejanović

dr.sc. Željka Ujević Andrijić, viši asistent

*Zagreb, srpanj 2015.*

## *Zahvala*

*Zahvalio bih se svom mentoru izv. prof. dr. sc. Nenadu Bolfu na svim savjetima vezanim za izradu diplomskog rada.*

*Puno hvala asistentici dr. sc. Željki Ujević Andrijić koja je u svakom trenutku izrade ovog rada bila na raspolaganju te je svojim znanjem, nesebičnom zalaganju i savjetima bila poticaj i podrška.*

*Također bih se želio zahvaliti dr.sc. Ivanu Mohleru i tehničaru Aleksandru Spieglu što su u svakom trenutku bili spremni pomoći.*

*Hvala mojim roditeljima, braći te djedu i baki što su mi sve omogućili i koji su tijekom cijelog studiranja u svim trenucima bili najveća podrška.*

## Sažetak

U rafinerijskim postrojenjima ključne procesne veličine kao što su sastavi procesnih struja i svojstva proizvoda mjere se pomoću procesnih analizatora. Budući da su procesni analizatori često nedostupni, ključna svojstva moraju se određivati rijetkim i dugotrajnim laboratorijskim analizama.

U radu je prikazan razvoj modela softverskih senzora za procjenu sadržaja benzena u lakom reformatu. Prikazana je usporedba više modela softverskih senzora razvijenih na temelju malog skupa eksperimentalnih podataka. Softverski senzori razvijeni su primjenom viševeličinske linearne regresije, viševeličinske adaptivne regresije (MARSpline) te primjenom neuronskih mreža.

Svi razvijeni modeli pokazali su dobro vladanje na skupu podataka za vrednovanje modela uz visoke korelacijske faktore i male pogreške modela. Kao najbolji modeli među razvijenima pokazale su se neuronske mreže. Nešto lošije, ali i dalje zadovoljavajuće vladanje dali su modeli viševeličinske adaptivne regresije te linearni model. Linearni viševeličinski model razvijen na generiranim podacima daje donekle lošije rezultate uz zadovoljavajuće vladanje pri određenim procesnim uvjetima.

### **Ključne riječi:**

softverski senzori, mali skupovi podataka, modeliranje procesa, neuronska mreža, *bootstrap* metoda

## **SUMMARY**

In refinery plants, key process variables like contents of process stream and fuel properties needs to be continuously monitored using adequate measuring devices. Measuring devices are often unavailable or malfunction, and laboratory assays, which are irregular and therefore not suitable for process control, are inevitable alternative.

In this study, soft sensors for the estimation of benzene content of light reformat are developed. This research shows a comparison of soft sensor models developed on the basis of a small data sets. Soft sensors have been developed using multivariable linear regression, multivariable adaptive regression splines (MARSpline) and neural networks.

Within the validation data set, all models have shown good results with high correlation factors and small errors. Best results are achieved by neural network-based models. Models developed on small data set by multivariable adaptive regression splines and by multivariable linear regression are satisfactory and still applicable for benzene content estimation while multivariable linear regression developed on resampled data set showed slightly worse behavior on particular data range.

### **Keywords:**

soft sensor, small data set, process modeling, neural network, bootstrapping

## **Sadržaj:**

### **Sažetak**

### **Summary**

<b>1. Uvod</b>	1
<b>2. Opći dio</b>	2
2.1. Softverski senzori	2
2.2. Postupak identifikacije procesa	5
2.2.1. Prikupljanje i predobrada podataka	7
2.2.2. Izbor ključnih veličina i strukture modela	10
2.3. Modeli softverskih senzora	12
2.3.1. Linearni viševeličinski model	12
2.3.2. MARSpline model	12
2.3.3. Neuronska mreža	15
2.3.4. Vrednovanje modela	20
<b>3. Metodika</b>	23
3.1. Opis procesa	23
3.1.1. Regulacija i mjerna oprema postrojenja	25
3.1.2. Analiza kvalitete proizvoda	27
3.2. Razvoj modela soft senzora za procjenu aromata u lakom reformatu	25
3.2.1. Prikupljanje i predobrada podataka	29
3.2.2. Generiranje novog skupa podataka bootstrap metodom	30
3.2.3. Razvoj modela i njegovo vrednovanje	31
3.2.3.1. <i>Linearni viševeličinski modeli</i>	33

3.2.3.2. <i>MARSpline model</i>	33
3.2.3.3. <i>Model neuronske mreže</i>	34
<b>4. Rezultati i rasprava</b>	<b>35</b>
4.1. Linearni modeli	35
4.2. MARSpline modeli	41
4.3. Razvoj modela primjenom neuronske mreže	47
4.3.1. Preliminarni razvoj neuronske mreže	47
4.3.2. Razvoj MLP neuronske mreže	49
<b>5. Zaključak</b>	<b>55</b>
<b>6. Literatura</b>	<b>56</b>
<b>Životopis</b>	<b>58</b>



# 1. Uvod

Zbog ubrzanog industrijskog razvoja, zahtjevima za što boljom kvalitetom i nižom cijenom proizvoda te zahtjevima zaštite okoliša i sigurnosti, procese kemijske industrije potrebno je optimalno voditi. Vođenje procesa zasniva se na regulaciji kontinuirano mjerenih veličina s ciljem ostvarivanja zadovoljavajuće kvalitete proizvoda uz minimalan utrošak energije i sirovina. Postoje brojni procesi u kemijskoj, petrokemijskoj i naftnoj industriji u kojima je vrlo teško ili uopće nije moguće kontinuirano mjeriti ključne veličine koje karakteriziraju kvalitetu produkta. Poznavanjem procesa i mjerenjem velikog broja procesnih veličina moguće je dovesti u vezu sekundarne, lako mjerljive veličine (kao što su protok, tlak, temperatura, razina) s primarnim, teško mjerljivim veličinama (kvaliteta i sastav proizvoda, točka tečenja, točka filtrabilnosti, temperatura početka i kraja destilacije i dr.).

Zbog zahtjevnih uvjeta u procesu (visoki tlakovi i visoke temperature) te korozivnosti medija procesni *on-line* analizatori često nisu dostupni zbog kvarova, redovitih održavanja i česte potrebe za umjeravanjem. Također, procesnim analizatorima je, kao i kod laboratorijskih analiza, potrebno određeno vrijeme kako bi se provela analiza, što otežava vođenje procesa.<sup>1</sup>

U industrijskim postrojenjima se za procjenu nemjerljivih veličina i kvalitete proizvoda mogu primjeniti softverski senzori.<sup>2</sup> Softverske senzore moguće je ugraditi u sustav kako bi paralelno radili s analizatorima. To omogućava detekciju kvara, a u slučaju kada je procesni analizator na održavanju ili kada je u kvaru, mogu ga privremeno zamijeniti i time održati automatsko vođenje procesa u funkciji.

U ovom radu razvijeni su i analizirani modeli softverskih senzora za procjenu sadržaja benzena u postrojenju za frakcionaciju reformata: dva modela viševereličinske linearne regresije, dva modela viševereličinske prilagodljive regresije (MARSpline) i dva modela temeljena na neuronskim mrežama. U okviru navedenih modela, jedan model razvijan je na temelju malog broja eksperimentalnih podataka, dok je drugi model razvijan na temelju generiranog skupa podataka *bootstrap* metodom te vrednovan na realnim podacima.

## 2. Opći dio

### 2.1. Softverski senzori

Naziv softverski senzor nastao je kao kombinacija riječi *software* jer su modeli zapravo računalni programi, te od riječi *sensors*, pošto modeli imaju uloge procesnih analizatora. Drugi uobičajeni nazivi za softverske senzore su i inferencijski senzori<sup>3,4</sup> ili virtualni online analizatori.<sup>5</sup>

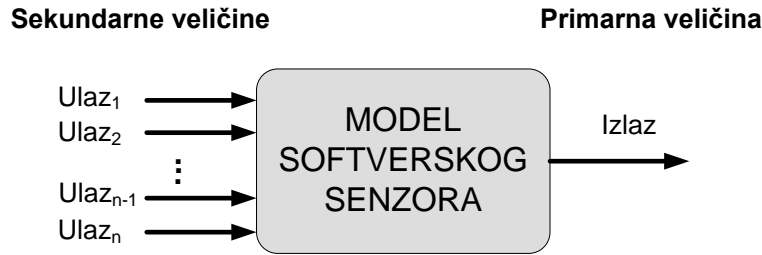
Neke ključne veličine u procesu ne mogu se mjeriti u stvarnom vremenu, odnosno dovoljno brzo da bi se koristile za automatsko vođenje procesa.<sup>6-8</sup> Za to postoji više razloga:

- Neka analitička mjerenja nisu dovoljno točna i pouzdana bez sudjelovanja ljudi u procesu. Rezultati ovakvih mjerenja dobivaju se samo povremeno iz laboratorija;
- Neka svojstva proizvoda se ne mogu određivati iz karakteristika poluproizvoda u pogonu. (Npr. kvaliteta goriva ili prehrambenih proizvoda ovisi o njihovoj krajnjoj primjeni i ne može se ispitati sve dok nisu potpuno izrađeni.);
- Trošak instalacije osjetila je prevelik da bi se opravdala ekonomska korist tog senzora. Troškovi standardnih senzora temperature, tlaka, protoka i razine obično nisu visoki, ali kod skupih analizatora koji imaju sustav za uzorkovanje i zahtijevaju velike troškove održavanja mogu biti ograničavajući faktor;
- Analizator ne može pravodobno pružiti informaciju zbog dugog vremena analize ili je smješten dalje od samog procesa;
- Nemogućnost izravnog mjerenja neke veličine, npr. konverzija stanovite kemijske reakcije.

Zbog čestih zadržki pri mjerenju ključnih veličina, automatsko vođenje procesa je otežano, ali to ne znači da ga nije uvijek nemoguće provesti.

Dobivanje dodatnih informacija iz lako mjerljivih sekundarnih veličina omogućavaju donošenje korisnih zaključaka vezanih za primarne, teško mjerljive veličine.<sup>6-8</sup>

Sekundarne veličine su ulaz u model procesa te se procjenjuje izlazna, odnosno primarna veličina, slika 1.



**Slika 1** Prikaz modela softverskog senzora

U industrijskim postrojenjima nalazi se veliki broj ugrađenih mjernih osjetila, *on-line* analizatora i digitalnih sustava čiji se mjerni podaci, u svrhu nadziranja i vođenja procesa, spremaju u bazu podataka.

Karakteristična područja primjene softverskih senzora su<sup>9</sup>:

- zamjena za mjerne uređaje;
- reduciranje potrebe za mjernom opremom;
- nadzor i vođenje procesa;
- otkrivanje pogrešaka i dijagnostiku procesa.

### **Softverski senzori kao zamjena za mjerne uređaje**

Pogreške pri radu sustava za vođenje mogu se javiti u obliku iznenadnih kvarova, ali i u obliku dugotrajnih teže uočljivih promjena statičke karakteristike koje mogu uzrokovati značajne probleme pri radu sustava za vođenje. Mjerne instrumente je, u takvom slučaju, potrebno zamijeniti. Najčešća primjena softverskih senzora upravo je kao trenutna zamjena za mjerne instrumente kako bi se omogućio nastavak rada postrojenja.

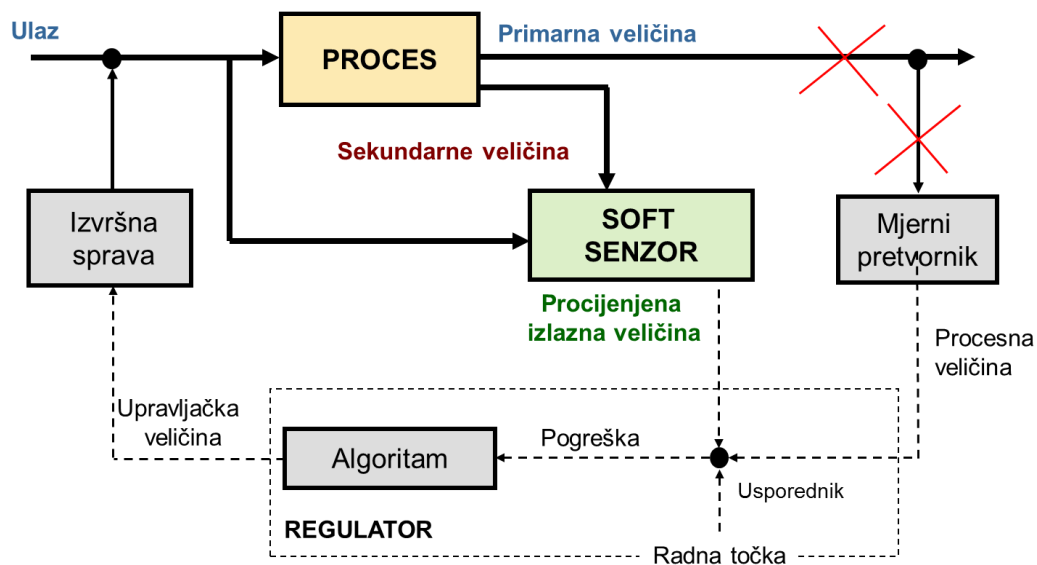
### **Reduciranje potrebe za mjernom opremom**

Primjena softverskog alata umjesto mjernog instrumenta dobra je osnova za moguće smanjenje troškova. Potrebno je provoditi i periodičku ocjenu valjanosti modela privremenom upotrebom mjernih instrumenata te ponovnim ugađanjem modela ako je to nužno. Periodička ocjena valjanosti softverskog senzora i njegovo ponovno ugađanje je glavni problem kod primjene.

## Softverski senzori za nadzor i vođenje procesa

Procjena varijabli sustava u stvarnom vremenu primjenom softverskih senzora, kad postoji zadržka iz mjernih uređaja, predstavlja najvažniju primjenu softverskih senzora. Svakom mjernom instrumentu potrebno je određeno vrijeme za provedbu mjerenja. Zbog visoke cijene nekih mjernih uređaja, o varijablama se ponegdje može zaključivati samo na temelju mjernog uređaja na udaljenom dijelu procesa. U tim slučajevima informacije dolaze s vremenskom zadržkom što onemogućuje učinkovito vođenje procesa.

U slučaju nemogućnosti mjerenja primarnih veličina ili nedostatka procesnog analizatora, vođenje procesa može se ostvariti primjenom modela koji procjenjuje izlaznu veličinu, slika 2.



Slika 2 Shematski prikaz regulacijskog kruga s softverskim senzorom

## Softverski senzori za otkrivanje pogrešaka i dijagnostiku procesa

Dio postupka nadzora suvremenih sustava u industriji su otkrivanje grešaka i dijagnostika procesa. U prošlosti, funkcija nadzora bila je ograničena na provjeru ključnih varijabli i na alarmiranje u slučaju izvanredne situacije. Danas se otkrivanje pogrešaka i dijagnostika provodi primjenom naprednih tehnika matematičkog modeliranja, obrade signala, metoda identifikacije, računalne inteligencije, itd.

## Ocjena valjanosti osjetila

Temeljni cilj ocjene valjanosti osjetila je dati informaciju o pouzdanosti provedenog mjerenja. Softverski senzori u ovom slučaju mogu biti sredstvo analitičke redundancije - pogreška se može otkriti usporedbom izlaza stvarnog osjetila i softverskog senzora.

## 2.2. Postupak identifikacije procesa

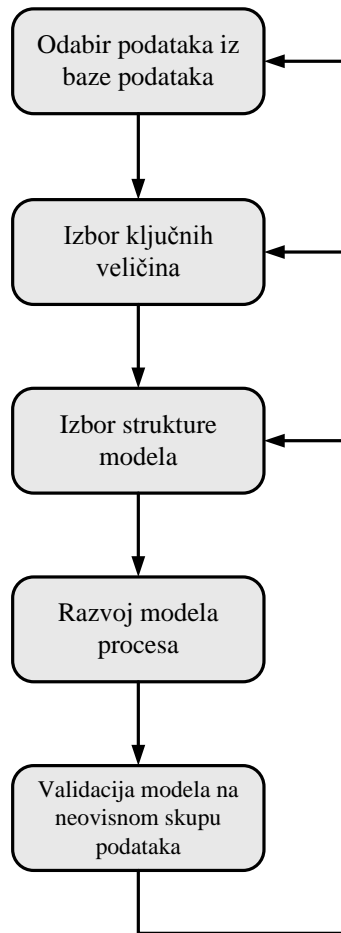
Pri razvoju modela softverskih senzora cilj je naći funkcijsku vezu između primarnih izlaznih veličina i sekundarnih ulaznih i izlaznih veličina.

Pri tome je moguće primijeniti jednu od metoda koje se uobičajeno koriste za razvoj modela procesa. Najvažniji i najčešće korišteni su matematički modeli procesa. Matematički modeli procesa mogu biti teorijski ili fundamentalni (engl. *white box*) modeli, eksperimentalni ili empirijski (engl. *black box*) modeli i kombinirani (engl. *gray box*) modeli.

Modeli razvijeni na temelju fizikalnih, kemijskih ili bioloških načela nazivaju se teoretski, odnosno fundamentalni modeli (engl. *first principles analysis*). Modeli koji su razvijeni empirijski, iz eksperimentalnih podataka, služeći se postupkom identificiranja nazivaju se modeli crne (engl. *black box*), odnosno sive kutije (engl. *gray box*). Model crne kutije je model sustava čija struktura nije poznata ili u čiju strukturu se ne ulazi. O svojstvima sustava zaključujemo na temelju promatranja ulaza i izlaza. Model sive kutije je model sustava čije karakteristike (značajke) poznajemo djelomično, odnosno elemente, strukturu i vladanje sustava možemo samo s određenom vjerojatnosti pretpostaviti. Model bijele kutije (engl. *white box*) je model sustava kojeg poznajemo u potpunosti.<sup>8</sup>

U slučaju industrijskih procesa, zbog njihove složenosti, razvoj fundamentalnih modela može biti vremenski vrlo zahtjevan, a u modelima se javlja velik broj parametara čije su vrijednosti nepoznate. Velika količina mjernih podataka u bazama podataka omogućava identificiranje nelinearnih modela sive ili crne kutije.<sup>10</sup>

Tipični koraci za razvoj softverskog senzora su prikazani slikom 3.<sup>11</sup>



**Slika 3** Dijagram toka pri razvoju softverskog senzora

### 2.2.1. Prikupljanje i predobrada podataka

U velikim industrijskim postrojenjima prikupljaju se i pohranjuju velike količine podataka iz procesa što pruža mogućnost njihove uporabe za identificiranje modela procesa. Pri identificiranju modela često se - javiti poteškoće prouzročene periodom uzorkovanja, nedostajućim podacima, ekstremnim vrijednostima (engl. *outliers*), radnim uvjetima, točnosti mjerenja, itd. Kako bi se moglo kritički odrediti podatke koji su reprezentativni potrebna je suradnja projektanta softverskog senzora i eksperata s postrojenja, operatora u kontrolnoj sobi i procesnih inženjera. Veoma vrijedni podaci za identifikaciju procesa dobivaju se iz *step* testova, odnosno skokomičnih promjena pobude procesa. Kako je u pogonskim uvjetima teško ili opasno izvoditi *step* testove, potrebno je prikupiti što veći broj podataka kroz što dulje vrijeme rada iz baze podataka. U tom slučaju softverski senzor je pouzdan samo u području rada za koju postoje podaci.<sup>12</sup>

Česta pojava u bazama podataka su nedostajuće vrijednosti čiji je najčešći uzrok kvar mjernog pretvornika ili njegovo održavanje. Isto tako, potrebno je ukloniti neželjene komponente iz mjernih uzoraka: ekstremne vrijednosti, „*offset*“, trend, klizanje, niskofrekvencijske periodičke smetnje, visokofrekventne smetnje.<sup>13</sup>

Osim automatskih postupaka predobrade podataka, podatke je potrebno i temeljito vizualno pregledati, jer neki neuobičajeni podaci, koji mogu izgledati kao znatna odstupanja, ponekad predstavljaju važnu karakteristiku dinamičkog vladanja.

Poznato je da je pri razvoju kvalitetnog modela potrebno imati što više podataka kroz što dulje vrijeme rada postrojenja. Budući da su procesni analizatori često nedostupni, ključna svojstva moraju se određivati rijetkim i dugotrajnim laboratorijskim analizama.

U takvim slučajevima zbog velikog vremena uzorkovanja od samo nekoliko analiza dnevno, dostupan je znatno manji broj podataka za razvoj model. Kod razvoja modela sa malim brojem podataka postoji velika mogućnost loše generalizacije jer pri razvoju modela nisu uzeti u obzir svi procesni uvjeti, pogotovo ako se ne posveti dovoljna pozornost pri izboru skupa podataka za razvoj modela, odnosno za vrednovanje.<sup>14</sup>

Kako bi se izbjegli navedeni problemi, u literaturi se predlažu neka rješenja. Predlaže se primjena višeslojne perceptronske mreže, MLP (engl. *Multi layer perceptrons*), temeljene na *bootstrap* metodi poduzorkovanja gdje se podaci za razvoj modela uzimaju nasumično s

moćnošću ponavljanja (tj. njihovog ponovnog korištenja) neograničen broj puta. Osim razvoja neuronskih mreža pomoću *bootstrap* metode poduzorkovanja, predlaže se i dodavanje šuma (engl. *Noise injection*) čime se izbjegava problem loše generalizacije te da bi se simulirala i dinamika koja inače nije obuhvaćena modelom. Predlaže se i generiranje novog skupa podataka dodavanjem Gaussovog šuma stalne varijance (engl. *Zero-mean fixed variance Gaussian noise*) ili *bootstrap* metodom generiranja (engl. *Bootstrap resampling method*) ili pak kombinacijom generiranja takvog novog skupa podataka sa dodavanjem šuma.<sup>15</sup>

### ***Bootstrap* metoda generiranja novog skupa podataka**

*Bootstrap* metodu predložio je Efron 1979. godine. To je metoda ponovnog uzorkovanja kojom se procjenjuju statističke distribucije bazirane na neovisnom promatranju. Metoda je veoma uspješna u mnogim slučajevima te je zato prihvaćena kao alternativa asimptotskim metodama. U nekim slučajevima daje bolje rezultate od pojedinih asimptotskih metoda, kao što su tradicionalna normalna aproksimacija (engl. *traditional normal approximation*) i Edgeworthova ekspanzija (engl. *Edgeworth expansion*). Međutim, postoje i slučajevi kada *bootstrap* metoda daje potpuno pogrešne rezultate.

Problem varijabilnosti procjene lokacije *bootstrap* metodom može se objasniti na sljedeći način. Ako su varijable  $x_1, x_2, \dots, x_n$  nezavisne slučajne varijable uobičajene funkcije distribucije,  $F$ , potrebno je odrediti varijabilnost i distribuciju uzorkovanja funkcije procjenjenih računom iz uzorka veličine  $n$ . Ako se procjena označi kao funkcija  $\hat{\theta}$ , ona je funkcija slučajnih varijabli  $X_1, X_2, \dots, X_n$  te stoga ima razdiobu vjerojatnosti i distribuciju uzorkovanja određenu sa  $F$  i  $n$ . Kod određivanja distribucije uzorkovanja, dva problema dolaze do izražaja:

1. kada funkcija  $F$  nije poznata;
2. čak i ako je funkcija  $F$  poznata, funkcija  $\hat{\theta}$  može biti jako složena te distribuciju nije moguće analitički odrediti.

U slučaju kada je poznata funkcija  $F$ , postavlja se pitanje kako odrediti funkciju  $\hat{\theta}$  bez korištenja iznimno složenih analitičkih proračuna. Pojavom računala, može se simulirati, odnosno generirati veliki broj uzoraka, određene veličine  $n$  funkcije  $F$  iz kojih se



preračunava  $\hat{\theta}$ . Empirijska distribucija je rezultat vrijednosti  $\hat{\theta}_1, \hat{\theta}_2, \dots, \hat{\theta}_B$ . Aproksimacija funkcije  $\hat{\theta}$  će biti zadovoljavajuća kada je  $B$  velik.

Kada funkcija  $F$  nije poznata, postoje dva različita slučaja. U prvom,  $F$  je nepoznat ali je poznat parametar  $\eta$ , funkcije tipa  $F(x|\eta)$ . Ideja ovog tipa *bootstrapa* (*parametarski bootstrap*) je u tome da se simuliraju podaci funkcije  $F(x|\hat{\eta})$  gdje je  $\hat{\eta}$  dobra procjena  $\eta$ . U drugom slučaju, struktura funkcije  $F$  je u potpunosti nepoznata. Ideja *neparametarskog bootstrapa* je simulacija podataka iz kumulativne distribucijske funkcije od  $F_n$ . Ovdje je  $F_n$  diskretna vjerojatnosna distribucija koja daje vjerojatnost  $1/n$  za svaku promatranu vrijednost  $x_1, x_2, \dots, x_n$ . Uzorak veličine  $n$  funkcije  $F_n$  je onaj uzorak koji je izvučen uz mogućnosti ponavljanja istog podatka iz skupa  $x_1, x_2, \dots, x_n$ . U tom slučaju standardna devijacija funkcije  $\hat{\theta}$  dana je formulom:

$$s_{\hat{\theta}} = \sqrt{\frac{1}{B} \sum_{i=1}^B (\theta_i - \bar{\theta})^2} \quad (1)$$

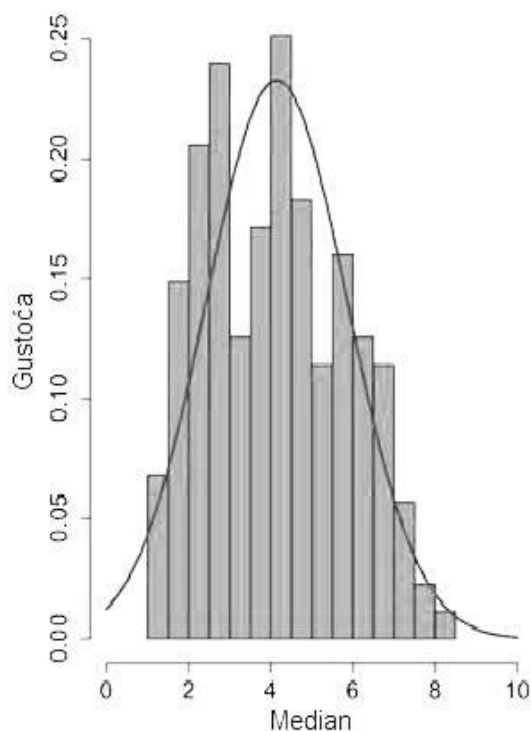
gdje su  $\hat{\theta}_1, \hat{\theta}_2, \dots, \hat{\theta}_B$  dobiveni iz  $B$  uzorka veličine  $n$  skupa  $x_1, x_2, \dots, x_n$ .

U slučaju velikog  $n$ , računanje distribucije se uvelike komplicira, te je stoga preporučeno jednostavno slučajno uzorkovanje (engl. *sampling*) za stvaranje *bootstrap* distribucije.

Promatranjem slučaja gdje se slučajnim odabirom stvora uzorak veličine  $n$  iz nespecificirane vjerojatnosne distribucije,  $F$ , neparametarski bootstrap razvija se kroz sljedeće osnovne korake<sup>16</sup>:

1. **korak:** Stvaranje empirijske vjerojatnosne distribucije,  $F_n$ , iz uzorka stvarajući vjerojatnost  $1/n$  za svaki slučaj  $x_1, x_2, \dots, x_n$  uzorka. Dobiva se empirijska funkcija distribucije koja je zapravo procjenjena maksimalna vjerojatnost distribucije populacije,  $F$ .
2. **korak:** Iz empirijske funkcije distribucije,  $F_n$ , uz mogućnost ponavljanja istog podatka slučajnim odabirom se stvara uzorak veličine  $n$ . Ovaj postupak se naziva ponovno uzorkovanje.
3. **korak:** Računaju se statistički parametri,  $T_n$ .

4. **korak:** Koraci 2 i 3 se ponavljaju  $B$  puta, kako bi se dobio  $B$  uzoraka, gdje je  $B$  veliki broj. U većini slučajeva,  $B$  je najmanje 1.000.
5. **korak:** Stvara se histogram relativnih frekvencija broja uzorka,  $B$  i nekog statističkog parametra,  $T_n$ , slika 4. Dobivena distribucija je rezultat *bootstrap* procjene distribucije uzorka  $T_n$ . Ova distribucija se sada može iskoristiti kao veza za dobivanje parametra  $\hat{\theta}$ .



**Slika 4** Usporedba bootstrap i normalne distribucije. Normalna distribucija prikazana je krivuljom<sup>17</sup>

### 2.2.2. Izbor ključnih veličina i strukture modela

Ovo područje kemijskog inženjerstva je veoma široko, te je opće rješenje za odabir ključnih veličina i strukture modela teško naći. Veoma bitan izvor informacija je iskustvo procesnih inženjera i operatora. Poznavanje informacija o izboru ulaznih veličina, reda sustava (engl. *system order*), radnog područja, vremenske zadržke, stupnja nelinearnosti,

perioda uzorkovanja i drugih informacija uvelike pomaže pri odabiru ključnih veličina i strukture modela procesa.

Prvo pitanje s kojim se susrećemo pri razvoju modela procesa je izbor nezavisnih veličina što utječu na izlaz iz modela. Ono se izvodi na sljedeće načine:

- Prvi i najintuitivniji pristup kojim se autori služe pri izboru varijabli je grafički prikaz odstupanja. Pored toga, rabe se kvantitativni kriteriji kao što su koeficijent korelacije i Mallowa statistika.<sup>18</sup>
- Drugi način je Kohonenovim mapama gdje se samoorganizirajuće mape rabe za preslikavanje podskupova ulaznih varijabli zajedno s izlaznim varijablama na mrežni izlazni prostor.<sup>19,20</sup>
- Kako bi se izbjegle numeričke poteškoće prilikom identifikacije u literaturi se predlaže upotreba tehnika temeljenih na preslikavanju poput PCA (engl. *Principal Component Analysis*) i PLS (engl. *Partial Least Squares*).<sup>21</sup>
- Druga u literaturi predložena tehnika za promjenjive procese je nezavisna analiza komponenata (engl. *Independent Component Analysis – ICA*).<sup>22</sup>

Kod odabira prikladne strukture modela potrebno je odabrati što jednostavniji model. To su najčešće autoregresivni modeli čije vladanje zadovoljava određene kriterije zbog njegove buduće implementacije u industriji. U slučajevima s velikim vremenom uzorkovanja izlaza, najčešće se razvijaju statički modeli, budući su prikladni dinamički modeli veoma složeni te zahtjevaju manje periode uzorkovanja. Pri odabiru vrste modela preporučljivo je početi s linearnim modelima, koji se zatim po potrebi generaliziraju u odgovarajuće nelinearne modele. Kod velikog broja industrijskih procesa, nelinearnost je blago izražena i vremenska promjenljivost dovoljno spora te se kod identifikacije procesa često može identificirati linearan statički proces. Kod procesa kod kojih je nelinearnost znatno izražena moraju se primijeniti nelinearni modeli te je u tom slučaju postupak identifikacije složeniji.<sup>13</sup>

## 2.3. Modeli softverskog senzora

### 2.3.1. Linearni viševereličinski model

Linearno viševereličinsko modeliranje je tehnika čiji je glavni cilj povezivanje i procjenjivanje nepoznatih vrijednosti izlaza iz više ulaznih, nezavisnih veličina koje se nazivaju prediktori. Linearnost regresijskog modela po pravilu se povezuje s dimenzijom varijabli i dimenzijom nepoznatih parametara. Model je linearan s obzirom na varijable ako svaka varijabla te parametar u modelu ima potenciju jednaku jedan. Model viševereličinske linearne regresije temeljni je regresijski model.

Opća jednadžba linearnog viševereličinskog modela (2) dana je u nastavku gdje su parametri  $b_i$  parametri pojedinog ulaza  $X_i$  analogni nagibu pravca te se nazivaju i regresijski koeficijenti. Parametar  $b_0$  označava odsječak na ordinati analogno onome kod pravca.

$$y = b_0 + \sum_{i=1}^n b_i \cdot X_i \quad (2)$$

$n$  je broj ulaznih veličina odnosno broj prediktora.

Pretpostavke pri razvoju linearnog viševereličinskog modela su:

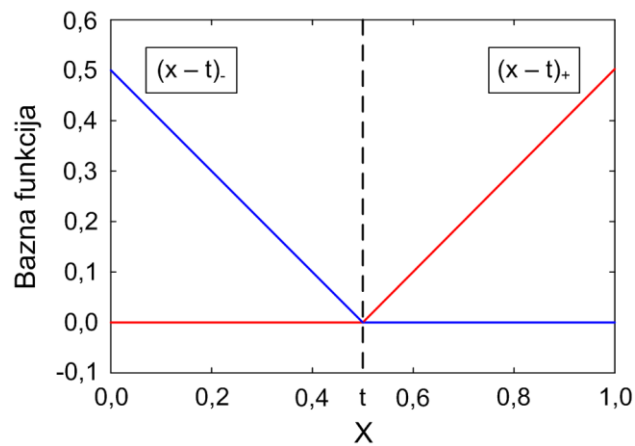
- Veza između izlaza  $y$  i svakog pojedinog ulaza  $X_i$  je linearna.
- Ulazne veličine nisu međusobno zavisne, što se jednostavno može odrediti računalnim određivanjem međusobnih korelacijskih koeficijenata.
- Uklonjeni su *outlieri* koji mogu znatno utjecati na razvoj modela.<sup>21</sup>

### 2.3.2. MARSpline model

Metodu viševereličinskog prilagodljivog regresijskog „*spline*“-a, MARSpline (engl. *Multivariate Adaptive Regression Splines*) predložio je Friedman 1991. godine. Namijenjena je za predviđanje vrijednosti zavisnih varijabli iz skupa nezavisnih varijabli. Veza između zavisne i nezavisnih varijabli kreira se iz skupa koeficijenata i tzv. baznih funkcija što se dobivaju na temelju eksperimentalnih podataka. Metoda se temelji na podijeli područja ulaznih varijabli od kojih svako ima svoju regresijsku jednadžbu, što ju čini pogodnu za rad s velikim brojem ulaznih varijabli.

Radi jednostavnije predodžbe, može se zamisliti da MARSpline radi poput viševeličinske po dijelovima linearne regresije u kojoj svaka točka prekida ili čvor procijenjen na temelju eksperimentalnih podataka, definira područje primjene za određenu jednadžbu linearne regresije. Pomoću ove metode moguće je pored nelinearne veze između zavisne i nezavisnih varijabli modelirati i njihove interakcije. Ova metoda postala je popularna u području „rudarenja“ podataka (engl. *data mining*) s obzirom da je primjenjiva u slučajevima kad je veza između zavisnih i nezavisnih varijabli nemonotona i teška za aproksimaciju pomoću parametarskih modela.<sup>11</sup>

Metoda koristi dvostrane odrezane (engl. *truncated*) funkcije kao bazne funkcije za razvoj modela, pomoću kojih se aproksimira veza između zavisne i nezavisnih varijabli.



**Slika 5** Bazne funkcije za metodu MARSpline

Na slici 5 prikazane su dvije bazne funkcije  $(t-x)_+$  i  $(x-t)_+$ . Parametar  $t$  je čvor baznih funkcija koji definira „dijelove“ linearne regresije. Čvor se također određuje iz podataka. Predznak „+“ uz članove  $(t-x)$  i  $(x-t)$  znači da se samo pozitivni rezultati odgovarajućih jednadžbi uzimaju u obzir, inače je vrijednost funkcije jednaka nuli.

U prvom koraku razvija se model i povećava njegova složenost dodavanjem baznih funkcija dok se ne dostigne unaprijed definirana maksimalna razina složenosti koju je definirao korisnik. Zatim se primjenjuje unatrag postupak kojim se uklanjaju najmanje značajne bazne funkcije iz modela, tj. one čije uklanjanje vodi do najmanjeg smanjenja kvalitete modela. Uklanjanjem najmanje značajnih baznih funkcija iz modela sprječava se „pretreniranje“ modela.

Razvoj modela ide ovim redom:

1. započeti s najjednostavnijim modelom koji sadrži samo konstantu bazne funkcije
2. pretraživati prostor baznih funkcija za svaku varijablu i sve moguće čvorove, kao i interakcije između varijabli i dodati onaj koji minimizira predikcijsku pogrešku
3. korak 2 rekurzivno primjenjivati dok model ne dostigne unaprijed definiranu maksimalnu razinu složenosti
4. u zadnjem koraku primjenjuje se postupak pročišćavanja kojim se uklanjaju one bazne funkcije koje najmanje doprinose ukupnoj kvaliteti modela

Bazne funkcije su definirane kao:

$$(x-t)_+ = \begin{cases} x-t & x > t \\ 0 & \text{inače} \end{cases} \quad (3)$$

MARSpline model za zavisnu varijablu  $y$  i  $M$  članova može se prikazati kao:

$$y = f(x) = \beta_0 + \sum_{m=1}^M \beta_m H_{km}(x_{v(k,m)}) \quad (4)$$

gdje se zbraja preko  $M$  članova modela, a  $\beta_0$  i  $\beta_m$  su parametri modela u vezi s čvorovima  $t$  za svaku baznu funkciju koji se također procjenjuju iz podataka. Funkcija  $H$  definira se kao:

$$H_{km}(x_{v(k,m)}) = \prod_{k=1}^K h_{km} \quad (5)$$

gdje je  $x_{v(k,m)}$   $k$ -ti prediktor  $m$ -tog produkta. Za red interakcija  $K = 1$  model je aditivan, a za  $K = 2$  model je interaktivan u parovima (engl. *pairwise interactive*).

Za vrijeme unaprijednih koraka u model se dodaju bazne funkcije u skladu s unaprijed definiranom razinom složenosti koja bi trebala biti znatno veća (najmanje dvostruko) od optimalne prema kriteriju najmanjih kvadrata (engl. *best least-squares fit*).

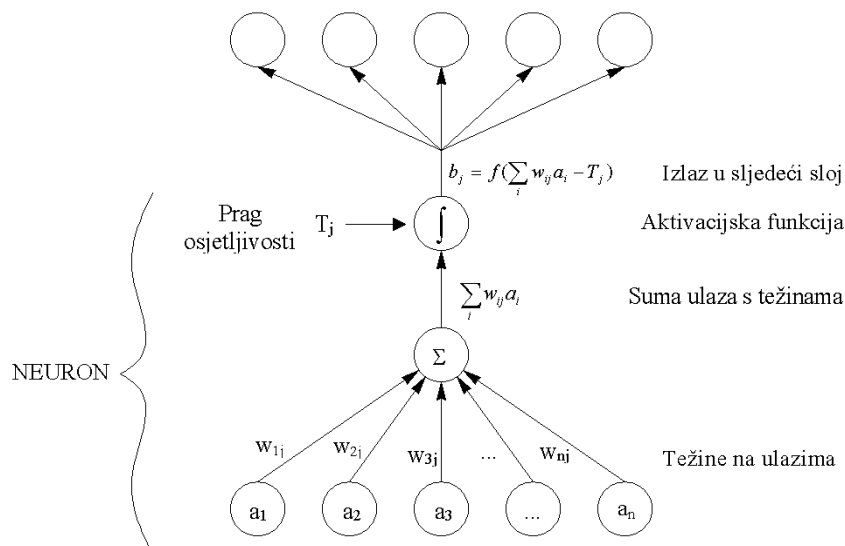
Nakon izbora baznih funkcija u unaprijednim koracima primjenjuje se unatražni postupak kojim se model pročišćava uklanjanjem onih baznih funkcija koje daju najmanji porast mjere prilagodbe u smislu najmanjih kvadrata. Stoga se računa funkcija pogreške najmanjih kvadrata (engl. *A least squares error function*). Tzv. pogreška generaliziranog unakrsnog provjeravanja (engl. *Generalized Cross Validation*) mjera je kvalitete prilagodbe koja uzima u obzir pogrešku i složenost modela. Dana je sljedećim izrazom:

$$GCV = \frac{\sum_{i=1}^N (y_i - f(x_i))^2}{\left(1 - \frac{C}{N}\right)^2}; \quad C = 1 + cd \quad (6)$$

gdje je  $N$  broj podataka,  $d$  je efektivan stupanj slobode koji je jednak broju nezavisnih baznih funkcija. Veličina  $c$  je najviša granica za dodavanje bazne funkcije.<sup>11</sup>

### 2.3.3. Neuronska mreža

Spoznaja o građi i načinu funkcioniranja ljudskog mozga i neurona kao njegovih građevnih jedinica potakla je istraživanje i razvoj umjetnih neuronskih mreža. Prve radove o umjetnim neuronskim mrežama objavili su McCulloh i Pitts (1943.), Pitts i McCulloh (1947.). Oni su koristili vrlo jednostavan model neurona koji, kao i biološki neuron, obrađuje signale putem sinaptičke i somatske operacije. Ovaj vrlo jednostavan model neurona nazvan je *perceptron*, slika 6. Sinaptička operacija predstavlja množenje svakog ulaznog signala  $a_i$  s težinskim koeficijentom  $w_i$ . Tako otežani ulazni signali se zbrajaju, a njihov zbroj uspoređuje se s pragom osjetljivosti neurona  $T_j$  (engl. *threshol*). Ako je zbroj otežanih signala veći od praga osjetljivosti neurona,  $\sum a_i w_i > T_j$ , nelinearna aktivacijska funkcija  $f$  generira izlazni signal neurona  $b_j$  iznosa jedan, a ako je manji, izlazni signal neurona je nula.



**Slika 6** Prikaz perceptrona

Početakom 60-tih dokazan je teorem o učenju perceptrona koji kaže da perceptron može naučiti sve što može predstaviti (Rosenblatt, 1962.). Predstavljanje se odnosi na sposobnost oponašanja stanovite funkcije. Pod učenjem se podrazumijeva postojanje sustavnog postupka za ugađanje parametara mreže u cilju ostvarivanja određene funkcije.

Sredinom 80-tih godina prošlog stoljeća otkriven je algoritam *povratnog prostiranja izlazne pogreške* (engl. *back propagation algorithm*) koji je omogućio jednostavno učenje višeslojnih neuronskih mreža (Rumelhartand McClelland, 1986.). Nakon otkrića ovog algoritma razvijeni su i brojni drugi, poboljšani, postupci učenja. Usporedno s razvojem algoritama istraživane su i nove strukture neuronskih mreža. Danas, zahvaljujući intenzivnom radu velikog broja istraživača na području razvoja teorije i praktične primjene neuronskih mreža, stoje nam na raspolaganju brojne strukture i algoritmi.<sup>22</sup>

Umjetna inteligencija pronalazi veliku primjenu u razmatranju kompleksnog financijskog tržišta jer ima svojstvo predviđanja s obzirom na dosadašnja iskustva, svemirskoj tehnologiji, učenju stranog jezika, ali i u svim područjima klasične znanosti jer je idealan alat za brzu obradu informacija, tj. podataka koji su međusobno povezani određenom logikom.

Neuronske mreže imaju svojstvo opće primjenjivosti. Njihova sofisticiranost omogućuje im univerzalnu primjenu, što ih čini još boljim i olakšava cjelokupnu primjenu.

Viševeličinski proces općenito se može prikazati kao:

$$y(k) = f(y(k-1), y(k-2), \dots, y(k-m), u(k-1), u(k-2), \dots, u(k-n)) \quad (7)$$

pri čemu  $y$  i  $u$  predstavljaju izlaze i ulaze, a indeks  $k$  označava vrijeme uzorkovanja,  $f$  predstavlja funkcionalni odnos između ulaza i izlaza. Kod tradicionalnih metoda identificiranja, potrebno je definirati oblik funkcije koja predstavlja model procesa, a također i odrediti parametre koji karakteriziraju taj oblik funkcije. Obično se uzima nekoliko pretpostavki o procesu kao što je linearnost, vremenska invarijantnost, itd., a rabe se i modeli smetnji da bi modeli obuhvatili i dinamiku koja inače nije obuhvaćena modelom.

Neuronska mreža ne zahtijeva eksplicitno definiranje funkcije. Potrebno je specificirati samo topologiju ili strukturu mreže. Specifikacija obuhvaća broj neurona u ulaznom, skrivenom i izlaznom sloju mreže. Broj neurona u ulaznom i izlaznom sloju određuje se prema broju bitnih ulaznih i izlaznih varijabli u procesu. Broj neurona u skrivenom sloju obično se određuje postupkom „pokušaja i pogreške“.



U svrhu modeliranja, obrađeni eksperimentalni podaci dijele se u tri skupa: skup za učenje, skup za provjeru i skup za testiranje. Podaci iz skupa za učenje stavljaju se u mrežu, a težinski koeficijenti se kontinuirano računaju dok se pogreška minimizira. Na kraju jedne iteracije (iteracija se odnosi na jedan prolaz svih podataka za učenje kroz mrežu), mreža predviđa vrijednosti skupom za provjeru. Težinski koeficijenti se mogu mijenjati samo za vrijeme učenja, a ne i za vrijeme provjere. Ako je pogreška na skupu za provjeru veća od specificiranog odstupanja, provodi se sljedeća iteracija. Proces se ponavlja sve dok pogreška ne bude manja od specificiranog odstupanja, ili je dosegnut maksimalni broj iteracija. Težinski koeficijenti dobiveni na kraju ovog postupka karakteriziraju model neuronske mreže.

Za provjeru dobivenog modela rabi se treći skup podataka – skup za vrednovanje, koji je nezavisan od podataka primijenjenih pri učenju.

Tipična neuronska mreža koja se poučava algoritmom unatragne propagacije uvijek ima jedan ulazni i izlazni sloj, te barem jedan skriveni sloj. Teorijski, nema ograničenja broja skrivenih slojeva, ali se u praksi većinom stavljaju jedan ili dva. Svaki sloj je u potpunosti povezan sa sljedećim slojem. Za vrijeme učenja informacija se također propagira unazad kroz mrežu, čime se ažuriraju težinski koeficijenti veza.<sup>26</sup>

Ulazi u neurone indeksirani sa  $i = 1, \dots, n$  primaju ulazne vrijednosti  $x_i$ . Ulazne vrijednosti neki su realni brojevi. Svaka ulazna vrijednost  $x_i$  množi se s težinskom vrijednošću  $w_i$ . Zbroj  $S$  svih veličina zove se interna aktivacija  $I$ . Ona je jednaka:

$$I_j^{[s]} = S = \sum w_j^{[s]} x_i^{[s]} \quad (8)$$

Tako dobiven zbroj  $S$  (ili interna aktivacija  $I$ ) obrađuje se pomoću prijenosne funkcije  $f$ . Izlaz

iz neuronske mreže jednak je  $x_j^{[s]}$ :

$$x_j^{[s]} = f\left(\sum_i w_{ji}^{[s]} \cdot x_i^{[s-1]}\right) = f(I_j^{[s]}) \quad (9)$$

pri čemu  $f$  može biti bilo koja derivabilna funkcija.

Uz pretpostavku da mreža ima definiranu globalnu funkciju pogreške  $E$  koja je derivabilna funkcija svih težinskih koeficijenta u mreži, za razumijevanje mehanizma

unatražne propagacije bitan parametar što se propagira unazad kroz slojeve definiran je izrazom:

$$e_j^{[s]} = -\partial E / \partial I_j^{[s]} \quad (10)$$

pri čemu je  $e_j^{[s]}$  mjera lokalne pogreške na elementu  $j$  u sloju  $s$ .

Primjenom *pravila niza* (engl. *chain rule*) uzastopno dva puta dobiva se odnos između lokalne pogreške na pojedinom procesnom elementu u sloju  $s$  i svih lokalnih pogrešaka u sloju  $s+1$ :

$$e_j^{[s]} = f'(I_j^{[s]}) \cdot \sum_k (e_k^{[s+1]} \cdot w_{kj}^{[s+1]}) \quad (11)$$

U jednadžbi (7), postoji sloj iza  $s$ , pa prema tome ona ne vrijedi za izlazni sloj.

Osnovni je mehanizam mreže s unatražnim propagiranjem unaprijedno prolaz ulaza kroz skrivene do izlaznog sloja, pa zatim propagiranje pogreške unazad kroz mrežu od izlaznog do ulaznog sloja prema jednadžbi (7).

Cilj postupka učenja je minimiziranje globalne pogreške sustava,  $E$ , promjenom vrijednosti težinskih koeficijenata. Na temelju trenutanih vrijednosti težinskih koeficijenata,  $w_{ij}^{[s]}$ , potrebno je odrediti njihovu promjenu tako da se globalna pogreška smanji. To se ostvaruje gradijentnom metodom spusta:

$$\Delta w_{ji}^{[s]} = -\eta \cdot (\partial E / \partial w_{ji}^{[s]}) \quad (12)$$

pri čemu je  $\eta$  koeficijent učenja. Drugim riječima, svi težinski koeficijenti mijenjaju se prema veličini i smjeru negativnog gradijenta na funkciju pogreške predočene površinom.

Parcijalna derivacija (12) može se izračunati direktno iz vrijednosti lokalnih pogrešaka jer se na temelju pravila niza i (11) dobiva:

$$\begin{aligned} \partial E / \partial w_{ji}^{[s]} &= \left( \partial E / \partial I_j^{[s]} \right) \cdot \left( \partial I_j / \partial w_{ji}^{[s]} \right) \\ &= -e_j^{[s]} \cdot x_i^{[s-1]} \end{aligned} \quad (13)$$

Uvrštavanjem (13) u (12) zajedno dobiva se

$$\Delta w_{ji}^{[s]} = -\eta \cdot e_j^{[s]} \cdot x_i^{[s-1]} \quad (14)$$

Funkcija globalne pogreške potrebna je da bi se definirale lokalne pogreške u izlaznom sloju tako da bi se mogle propagirati unatrag kroz mrežu. Pretpostavimo da vektor  $i$

predstavlja ulazni sloj mreže, vektor  $d$  željeni izlaz, dok stvarni izlaz što ga daje mreža uz trenutačne vrijednosti težinskih koeficijenata predstavlja vektor  $o$ . Kriterij pogreške dan je izrazom:

$$E = 0,5 \cdot \sum_k ((d_k - o_k)^2) \quad (15)$$

pri čemu indeks  $k$  označava komponente vektora  $d$  i  $o$ . Lokalna pogreška dana je s  $d_k - o_k$ . Prema jednadžbi (11) "lokalna pogreška" na svakom procesnom elementu izlaznog sloja definirana je ovako:

$$\begin{aligned} e_k^{(o)} &= -\partial E / \partial I_k^{(o)} \\ &= -\partial E / \partial o_k \cdot \partial o_k / \partial I_k \\ &= (d_k - o_k) \cdot f'(I_k) \end{aligned} \quad (16)$$

U jednadžbi (15)  $E$  predstavlja globalnu pogrešku mreže za određeni  $(i, d)$ . Ukupna funkcija globalne pogreške može se definirati kao suma svih uzorka funkcija pogrešaka. Za svaki pojedini par  $(i, d)$ , algoritam unatračne propagacije modificira težinske koeficijente da bi reducirao pojedinu komponentu funkcije ukupne pogreške.

Prema tome u standardnom algoritmu unatračne propagacije za dani ulazni vektor  $i$  željeni izlazni vektor,  $d$  izvode se slijedeće koraci:

1.  $i$  se dovodi ulazu sloja mreže i propagira kroz mrežu do izlaza da bi dobili izlazni vektor  $o$ ;
2. Kako se ova informacija propagira kroz mrežu, ona također određuje sumu svih ulaza  $I_j$  i izlazna stanja  $x_j$  svakog procesnog elementa u mreži;
3. Za svaki procesni element u izlaznom sloju mreže, računa se lokalna pogreška, prema jednadžbi (16), pa se zatim računa razlika težinskih koeficijenta jednadžbom (14);
4. Za svaki sloj  $s$ , počevši od sloja iza izlaznog i završivši sa slojem prije ulaznog, te za svaki procesni element u sloju  $s$ , računa se lokalna pogreška prema jednadžbi (15), pa se zatim računa razlika težinskih koeficijenta prema jednadžbi (14);
5. Svi težinski koeficijenti u mreži se ažuriraju dodavanjem razlika težinskih koeficijenta odgovarajućim prethodnim vrijednostima težinskih koeficijenta.

### 2.3.4. Vrednovanje modela

Nakon što su se odredili parametri modela procesa, model je potrebno vrednovati. Postupcima vrednovanja ocjenjuje se vladanje modela s vladanjem stvarnog procesa na neovisnom skupu podataka unutar pretpostavljenih radnih uvjeta. U slučaju da se uzeo isti skup podataka za identifikaciju i za vrednovanje, model se može dobro vladati s tim podacima, a loše s novim skupom podataka, javlja se pojava pretreniranja.

Ako se postupcima vrednovanja utvrdi da model ne zadovoljava, potrebno je otkriti uzroke i ponoviti odgovarajući dio postupka identificiranja. Na primjer, pogreška kod postupka identificiranja se može javiti zbog pogrešnog odabira podataka korištenih pri identificiranju modela, netočnog otkrivanja ekstremnih vrijednosti, neodgovarajućeg izbora utjecajnih varijabli, itd.

Kao opći pokazatelji kakvoće modela mogu se rabiti Pearsonov koeficijent korelacije dan izrazom (17) te koeficijent višestruke determinacije koji je dan izrazom (18):

$$R = \frac{\sum_{i=1}^n (y_i - \bar{y})(\hat{y}_i - \bar{\hat{y}})}{\sqrt{\sum_{i=1}^n (y_i - \bar{y})^2 (\hat{y}_i - \bar{\hat{y}})^2}} \quad (17)$$

$$R^2 = \frac{\sum_{i=1}^n (\hat{y}_i - \bar{\hat{y}})^2}{\sum_{i=1}^n (y_i - \bar{y})^2}, \quad 0 \leq R^2 \leq 1. \quad (18)$$

Pearsonov koeficijent je omjer suma umnoška razlika modela i eksperimentalnih podataka s korijenom njihovih kvadrata razlika.

Koeficijent višestruke determinacije je omjer zbroja kvadrata odstupanja protumačenog modelom i ukupnog zbroja kvadrata odstupanja. Osim koeficijenta višestruke determinacije računa se i korigirani koeficijent determinacije. Koeficijent je dan izrazom:

$$\bar{R}^2 = 1 - \frac{n-1}{n-(K+1)} \cdot (1-R^2), \quad \bar{R}^2 \leq R^2 \quad (19)$$

Pri računanju korigiranog koeficijenta uzima se u obzir broj stupnjeva slobode, koji ovisi o broju nezavisnih varijabli u modelu. Koeficijent determinacije monotono je neopadajuća funkcija broja nezavisnih varijabli pa mu se s povećanjem broja varijabli po pravilu povećava vrijednost. Model većih dimenzija ne mora biti i reprezentativniji mjeri li se njegova reprezentativnost procjenom regresijske varijance. Skup varijabli u regresijskom modelu s najmanjom procjenom varijance ima najveću vrijednost korigiranog koeficijenta determinacije. Zbog toga se u prosudbi reprezentativnosti modela, osim koeficijenta determinacije, rabi i korigirani koeficijent determinacije.

U slučaju numeričke procjene valjanosti modela najpopularnija mjera u praktičnim primjerima je korijen srednje kvadratne pogreške (engl. *Root Mean Square Error, RMS*) između procijenjenih i stvarnih vrijednosti izlazne veličine.

$$\text{RMS} = \sqrt{\frac{\sum_{i=1}^n (\hat{y}_i - y_i)^2}{n}} \quad (20)$$

U praktičnim primjerima koriste se još i sljedeći statistički pokazatelji:

- srednja apsolutna pogreška

$$e_{\text{MSE}} = \frac{1}{n} \cdot \sum_{i=1}^n \left| \hat{y}_i - y_{\text{exp},i} \right| \quad (21)$$

- aritmetička sredina pogreške

$$\bar{e} = \frac{\sum_{i=1}^n e_i}{n} \quad (22)$$

- standardna devijacija pogreške

$$\text{Error } \sigma = \sqrt{\frac{\sum_{i=1}^n (e_i - \bar{e})^2}{n}} \quad (23)$$

U rezultatima su još prikazane i aritmetička sredina eksperimentalnih podataka (izraz 24) kao i standardna devijacija podataka (izraz 25).

- aritmetička sredina eksperimentalnih podataka

$$\bar{y}_{\text{eksp}} = \frac{\sum_{i=1}^n y_{\text{eksp},i}}{n} \quad (24)$$

- standardna devijacija eksperimentalnih podataka

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (y_{\text{eksp},i} - \bar{y}_{\text{eksp}})^2}{n}} \quad (25)$$

### 3. Metodika

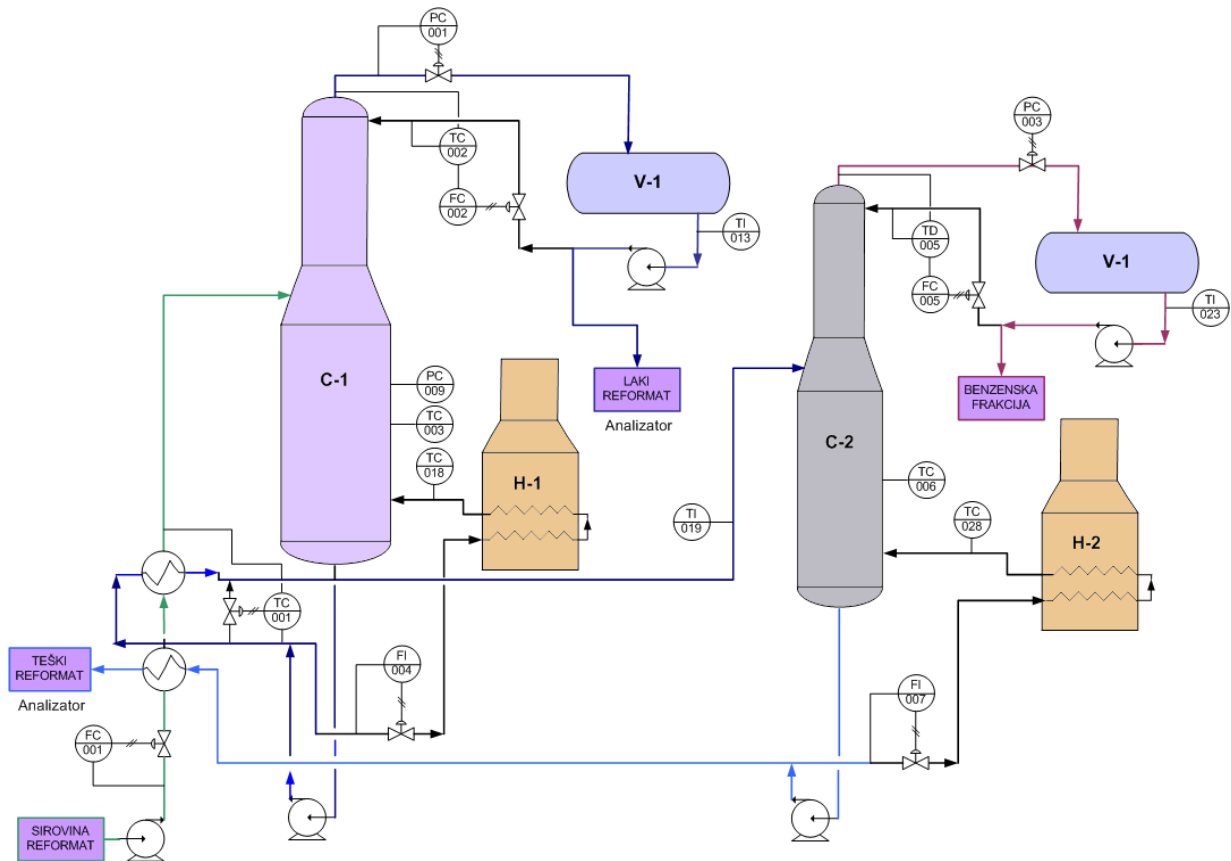
#### 3.1. Opis procesa

Proces katalitičkog reforminga je kemijski proces kojim se primarni benzin, niskog oktanskog broja, u nazočnosti katalizatora prevodi u visokooktanski reformat te je jedan od najvažnijih procesa u naftnoj industriji. Reformat je dominantna komponenta za namješavanje motornih benzina s udjelom višim od 30%. Procesom reforminga mijenja se struktura ugljikovodika, koji se iz ravnolančanih ugljikovodika (pretežito n-parafina) prevode u razgranate ugljikovodike s višim oktanskim brojem, kao i mijenja struktura nekih većih molekula.

U katalitičkom reformatu se od svih benzinskih komponenti (alkilat, izomerizat, FCC benzin, kreking benzini) nalazi najveća koncentracija ekološki nepoželjnog benzena (5-6 vol. %). Kako bi se udovoljile tehničke i zakonske norme benzina glede sadržaja benzena u gorivima do maksimalno 1 vol. %, benzen je potrebno ukloniti iz reformata. Iako je moguće kontrolirati nastajanje benzena predfrakcionacijom benzina podešavajući temperaturu kraja destilacije teškog benzina, odnosno povećavajući temperaturu kraja destilacije laganog benzina, nije moguće potpuno spriječiti njegovo nastajanje. Stoga je najbolje naknadno ukloniti benzen iz reformata postfrakcionacijom u spliteru reformata. To je zadaća postrojenja za frakcionaciju reformata. Ulazna sirovina postrojenja za frakcionaciju reformata je katalitički reformirani benzin (reformat).

Postrojenje za frakcionaciju reformata (slika 7) služi za izdvajanje lakih sastojaka reformata, koji se sastoji većinski od C<sub>5</sub> i C<sub>6</sub> ugljikovodika, okvirno raspoređenih kao frakcija C<sub>5</sub>-85 °C, tj. frakcija koja sadrži pentane i C<sub>6</sub> ugljikovodike s krajem destilacije kod oko 85°C na atmosferskom tlaku. Frakcija bogata benzenom, koja se po području vrelišta nalazi između lakog i teškog reformata, izvlači se iz katalitički reformiranih benzina.

Pored zahtjeva zakonske regulative, sadržaj benzena u lakom reformatu potrebno je ograničiti i zbog tehnoloških razloga. Naime, hidrogenacijom benzena pri procesu izomerizacije benzen se prevodi u cikloheksan, koji je zbog niskih oktanskih brojeva nepoželjan kao komponenta motornih benzina.



**Slika 7** Prikaz postrojenja za frakcionaciju reformata

Slika 7 prikazuje postrojenje za frakcionaciju reformata. Smjesa reformata se uvodi u kolonu C-1 (spliter) gdje se izdvaja laki reformat ( $n\text{-C}_6$ ) od smjese teškog reformata i benzenske frakcije. Kako benzen i  $n\text{-C}_6$  imaju bliska vrelišta i nisku relativnu hlapivost, njihovo razdvajanje može biti jako zahtjevno. Pošto je kolona C-1 veće visine i većeg promjera, razdvajanje benzenske frakcije od lakog reformata odvija se u njoj dok će se onda teški reformat i benzenska frakcija odvajati u C-2 koloni. Na taj način se postiže maksimalni mogući iscrpak  $n\text{-C}_6$  kao sirovinke komponente za proces izomerizacije, te je veća oštrina razdvajanja  $n\text{-C}_6$  od benzena. Sadržaj benzena u gornjem produktu kolone C-1, odnosno lakom reformatu ograničen je na manje od 2 vol %. Sav  $n\text{-C}_6$  koji zaostaje u preostalom reformatu, odnosno u produktu dna kolone C-1 predstavlja gubitak sirovine za izomerizaciju.

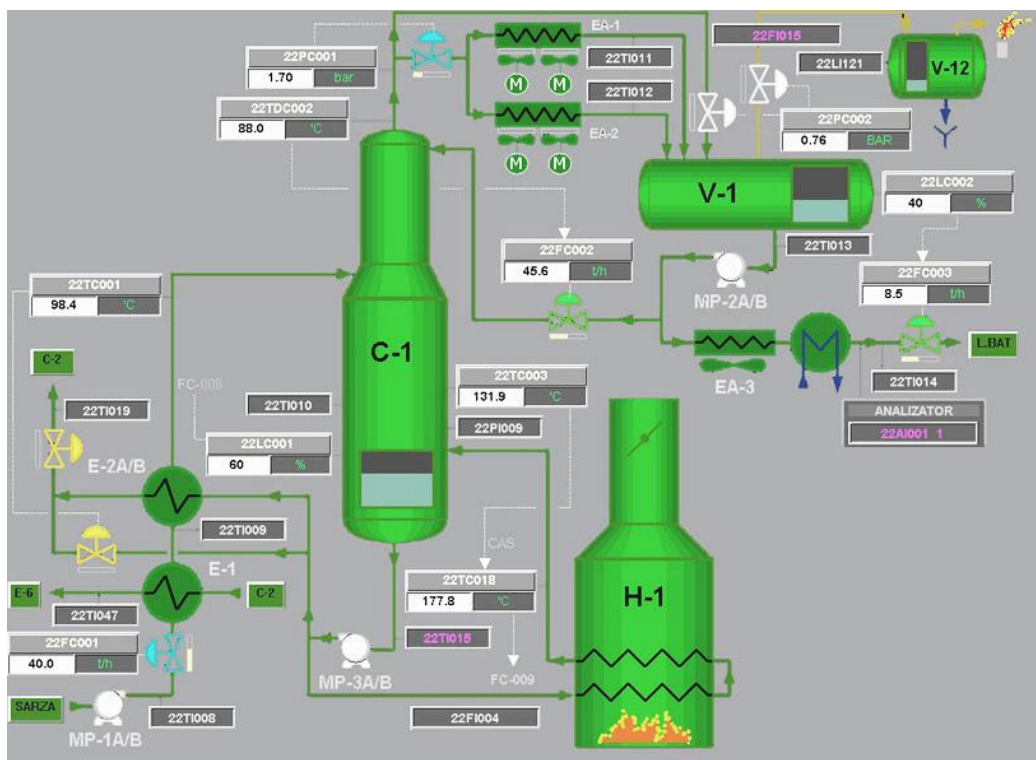
Produkt dna kolone C-1 je sirovina za kolonu C-2, čija je uloga razdvojiti benzensku frakciju od teškog reformata. Budući da su razlike vrelišta odnosno relativna hlapivost između benzena i teškog reformata ( $C_7$ -izomer-2,4 DM pentan) nešto veće nego za prethodni par ključnih komponenti ( $n\text{-C}_6$ /benzen) te zahvaljujući tome što benzenske frakcije kao destilata ima znatno manje, razdvajanje se može zadovoljavajuće provesti u koloni C-2. Potrebni



refluksni omjer je također manji. Dozvoljeni sadržaj benzena u teškom reformatu, kao produktu dna kolone C-2, mora biti manji od 1 vol %.

### 3.1.1. Regulacija i mjerna oprema postrojenja

Kako bi se odredile potencijalne utjecajne varijable na temelju kojih će se razviti model softverskog senzora za procjenu sadržaja benzena u lakom i teškom reformatu potrebno je detaljno analizirati vođenje motrenog postrojenja. Reformate se na postrojenje frakcionacije reformata doprema iz spremnika sirovinskom pumpom pri čemu se protok mjeri i regulira regulatorom protoka FC001. Sirovina se prije ulaska u kolonu C-1 predgrijava u izmjenjivačima topline s vrućim proizvodima dna kolona C-1 i C-2 na 100 °C te ulazi u kolonu C-1. Temperatura ulaza sirovine u C-1 regulira se s regulatorom temperature TC001, odnosno putem ventila na obilaznom vodu toplog toka iz dna kolone C-1. Vršne pare kolone C-1 pri temperaturi od 85,7 °C ulaze u paralelno spojene zračne kondenzatore, a izlazni kondenzat se sakuplja u prihvatnoj posudi pretoka i proizvoda u laki reformat V-1.



Slika 8 P&I dijagram kolone C-1 za izdvajanje lakog reformata

Tlak u vrhu kolone C-1 održava se između 2 i 7 bara i regulira pomoću regulatora uzvodnog tlaka. Tlak u sabirnoj posudi V-1 održava se regulacijom PC002. Protok proizvoda

lakog reformata regulira se i mjeri s regulatorom protoka FC003. Kaskadna veza regulatora temperature i protoka TDC002/FC002 omogućuje regulaciju protoka pretoka te održava potrebnu razliku temperatura pretoka i vršnih para kolone C-1. Slika 8 prikazuje i položaj *on-line* analizatora za kontinuiranu analizu sadržaja benzena u lakom reformatu. Laki reformat hladi se sa 65 °C na 40 °C u zračnom hladnjaku i odvodi u spremnike nakon čega se doručuje na postrojenju za izomerizaciju. Dnu kolone C-1 toplina se dovodi u reboilerskoj peći H-1. Temperatura izlaza iz peći, odnosno sadržaj topline izlaznog toka peći regulira se regulatorom TC-004/FC-004. Proizvod dna kolone C-1 odvodi se pumpom te se hladi u izmjenjivaču topline E-2 sa 161,8 °C na 125,1 °C, i ulazi u kolonu C-2.

Tablica prikazuje potencijalne ulazne varijable modela softverskog senzora za procjenu sadržaja benzena u lakom reformatu s pripadajućim oznakama iz P&I dijagrama.

Mjernim podacima može se pristupiti putem rafinerijskog PHD (engl. *Process History Database*) sustava. Ovaj sustav omogućuje kontinuirano praćenje i pohranu svih ključnih varijabli.

**Tablica 1** *Potencijalno utjecajne varijable na procjenu sadržaja benzena u lakom reformatu*

	<b>Ulazna veličina</b>	<b>Oznaka</b>	<b>Jedinica</b>
1.	Temperatura na ulazu u kolonu C-1	22TC001	°C
2.	Temperatura dna kolone (izlaz iz H-1)	22TC018	°C
3.	Temperatura na sredini C-1 kolone	22TC003	°C
4.	Tlak u sredini C-1 kolone	22PI009	bar
5.	Temperatura na vrhu C-1 kolone	22TC002	°C
6.	Tlak na vrhu C-1 kolone	22PC001	bar
7.	Refluks iz V-1	22FC002	t/h
8.	Protok ulaznog reformata u C-1	22FC001	t/h
9.	Ukupni protok kroz peć H-1	22FI004	t/h
10.	Temperatura izlaza iz V-1	22TI013	°C

### 3.1.2. Analiza kvalitete proizvoda

Benzen je iznimno otrovan u kontaktu sa živim organizmima. Udisanje visokih koncentracija benzenskih para može uzrokovati trenutnu smrt. Dugoročno, pak, djelovanje benzena u malim koncentracijama može uzrokovati leukemiju i više vrsta karcinoma. Osim toga, dugotrajna izloženost benzenu može rezultirati utjecajem na koštanu srž, anemijom, što posljedično izaziva krvarenja i slabljenje imunološkog sustava. Benzen predstavlja globalni zdravstveni problem, napada vitalne organe, te može uzrokovati štetu na kromosomima i nasljedna genetska oštećenja.<sup>27</sup>

Najčešći način kontaminacije benzenom nastaje udisanjem. Značajni udio onečišćenja zraka benzenom dolazi upravo iz vozila, pa je njegov sadržaj potrebno ograničiti u gorivima. Čak i s tehničkog gledišta, iako ima vrlo visok oktanski broj i visoku energetska vrijednost, benzen nije poželjna komponenta u lakom reformatu, odnosno sirovini za proces izomerizacije, jer se hidrira u cikloheksan, koji je nepoželjan zbog niskog oktanskog broja.

Prema europskim normama (EURO-III, EURO-IV i EURO-V standardu) za motorna vozila i ekvivalentnim ostalim normama (npr. MSAT – *Mobile Source Air Toxics*), čije vrijednosti Inina goriva moraju zadovoljavati, maksimalni dopušteni udio benzena u gorivima iznosi 1 vol %. Zbog toga je sadržaj benzena u rafinerijskim postrojenjima neophodno kontinuirano mjeriti.

Sadržaj benzena u lakom i teškom reformatu kontinuirano se određuje pomoću procesnog analizatora - kromatografskog uređaja u skladu s normom HRN EN 12177/AC:2002 *Tekući naftni proizvodi – Bezolovni motorni benzin – Određivanje količine benzena plinskom kromatografijom* (EN 12177:1998 AC/2000).

Sadržaj benzena laboratorijski se određuje i prati u skladu s normom EN HRN 238/A1:2005 - "Tekući naftni proizvodi - Benzin - Određivanje količine benzena pomoću infracrvene spektrometrije" (engl. EN 238:1996/A1:2003 *Liquid petroleum products - Petrol – Determination of the benzene content by infrared spectrometry*).<sup>12</sup>

### 3.2. Razvoj modela soft senzora za procjenu aromata u lakom reformatu

U postrojenjima za preradu nafte neophodno je kontinuirano pratiti sadržaj benzena u lakom i teškom reformatu. Kako je procesni analizator koji bi trebao kontinuirano procjenjivati sadržaj benzena često u kvaru, u ovom radu su razvijeni modeli softverskih senzora za kontinuiranu *on-line* procjenu sadržaja benzena u lakom reformatu. Modeli softverskih senzora razvijeni su pomoću softverskog paketa Statistica verzija 12.5.

Ideja za izradu ovog rada bila je utvrditi može li se dobiti zadovoljavajuća kvaliteta modela razvijenih na temelju malog skupa eksperimentalnih podataka. Modeli su razvijani na dva načina. Prvi način razvoja modela na temelju malog skupa eksperimentalnih, odnosno realnih podataka, obuhvatio je primjenu *bootstrap subsampling* metode za razvoj MLP. Kod drugog načina razvoja modela na temelju malog skupa podataka primijenilo se generiranje novih podataka na temelju *bootstrap* metode. *Bootstrap* metodom generiran je znatan broj novih podataka s manjim periodom uzorkovanja, koji su se nadalje koristili za razvoj linearnog, MARSpline te MLP modela neuronske mreže.

U nastavku ovog poglavlja detaljnije je objašnjen svaki od navedenih koraka istraživanja.

Na slici 9 shematski je prikazan model softverskog senzora s odabranim ulaznim veličinama, koje su detaljnije prikazane u tablici 2. Izlazna veličina je sadržaj benzena izražen u volumnim postotcima.



**Slika 9** Struktura modela softverskog senzora s odabranim ulaznim veličinama

**Tablica 2** *Ulazne veličine modela*

	<b>Ulazne veličine</b>	<b>Oznaka</b>	<b>Jedinica</b>
$U_1$	Temperatura na ulazu u C-1	TC001	°C
$U_2$	Temperatura dna C-1	TC018	°C
$U_3$	Temperatura sredine C-1	TC003	°C
$U_4$	Tlak u sredini C-1	PI009	Bar
$U_5$	Refluks iz V-1	FC002	t/h

### 3.2.1. Prikupljanje i predobrada podataka

Podaci se preuzimaju s PHD sustava koji pohranjuje sve kontinuirano mjerene veličine. S obzirom da se radi o proizvodnom postrojenju, kako se ne bi utjecalo na kvalitetu proizvoda, ulazno/izlazni signali mjere se za vrijeme normalnog rada procesa. Potrebno je prikupiti podatke iz što dužeg vremenskog razdoblja rada postrojenja, obuhvaćajući pri tom što je moguće više različitih režima rada. Veoma je važno od procesnih inženjera saznati detaljan opis procesa kao i dozvoljene i nedozvoljene radne uvjete. Također je bitno znati i svrhu primjene softverskog senzora, koja je u ovom slučaju zamjena procesnog analizatora kada je on u kvaru.

Iz baze podataka uzeti su podaci ulaznih varijabli izmjereni u neprekinutom nizu mjerenja od tri tjedna rada. Uzeto je približno 6.000 podataka s vremenom uzorkovanja od pet minuta. Uzorkovanje izlaza provedeno je laboratorijskim analizama svaka dva sata te je prikupljen 251 podatak izlaza.

Nakon prikupljanja podataka s postrojenja uslijedila je predobrada podataka.

Predobrada podataka je obuhvatila uklanjanje ekstremnih vrijednosti i njihovo nadomještanje te filtriranje podataka.<sup>12</sup>

Otkrivanje ekstremnih vrijednosti provedeno je metodom *3 sigma* te je uspoređeno s otkrivanjem ekstremnih vrijednosti pomoću Hampel identifikatora uz vizualni pregled podataka.<sup>12</sup>

Filtriranje je postupak kojim se nastoji smanjiti mjerni šum prisutan kod mjernih pretvornika. Svrha filtriranja je „izgladiti“ podatke i tako olakšati razvoj modela, tj. onemogućiti modeliranje šuma što rezultira nepotrebnom parametrizacijom modela. Ipak, filtriranje treba provoditi s mjerom jer u obrađenim podacima treba ostaviti dovoljno informacija. U ovom radu korišten je Loess filter (engl. *locally weighted scatter plot smoothing*). Loess filter je vrsta lokalne regresije kod koje se kombinira težinska linearna metoda najmanjih kvadrata i polinomski model 2. reda.

Za analizu utjecaja pojedinih ulaza na izlaze modela softverskog senzora provedena je analiza osjetljivosti. Broj ulaznih veličina u modelu mora biti dovoljno velik da dobro opisuje dinamiku procesa, ali opet ne prevelik zbog načela jednostavnosti. Analiza osjetljivosti za dane ulaze računa se tako da se odredi osjetljivost izlaza na promjene vrijednosti ulaza pri čemu se svaki puta jedan od ulaza izostavi. Osnovna mjera za osjetljivost je pogreška modela u odnosu na pogrešku modela kada su u obzir uzete sve potencijalne varijable. U istu svrhu provedena je i analiza korelacijskih koeficijenata te PCA i PLS analiza.<sup>12</sup>

Podaci su podijeljeni u dva skupa podataka, 80% podataka za razvoj modela te preostalih 20% za vrednovanje modela.

### 3.2.2. Generiranje novog skupa podataka *bootstrap* metodom

U programskom paketu *Statsoft Statistica 12.5. bootstrap* metodom generirano je iz 6.000 ulaza i 251 izlaza, 6.000 sparenih ulaza i izlaza. Slikom 10 prikazana je struktura *bootstrap* modela u *Statsoft Statistica 12.5.* gdje je broj ponavljanja,  $B$ , namješten na 1.000, a analiza podataka se provodila korelacijski.

Sintaksa *bootstrap* metode za generiranje novih podataka provodi se na način prikazan na slici 10. Prema sintaksi, svi ulazi se koreliraju međusobno, što je označeno sa [Ulaz  $X$ ] –  $z$  – [Ulaz  $Y$ ] gdje su  $X$  i  $Y$  oznake pojedinih ulaza (iznosi  $X$  i  $Y$  su od 1 do 5 pošto je 5 ulaznih varijabli, a  $z$  je oznaka broja korelacije) te se izlaz korelira sa svim ulazima [Izlaz] –  $z$  – [Ulaz  $X$ ]. Na kraju se generiraju novi ulazi i izlazi što se u sintaksi programa označava kao [Ulaz  $X$ ] – {1.} – [Ulaz  $X$ ], odnosno kod izlaza [Izlaz] – {1.} – [Izlaz].

```
Analysis syntax
[Ulaz2]-1-[Ulaz1]
[Ulaz3]-2-[Ulaz1]
[Ulaz3]-3-[Ulaz2]
[Ulaz4]-4-[Ulaz1]
[Ulaz4]-5-[Ulaz2]
[Ulaz4]-6-[Ulaz3]
[Ulaz5]-7-[Ulaz1]
[Ulaz5]-8-[Ulaz2]
[Ulaz5]-9-[Ulaz3]
[Ulaz5]-10-[Ulaz4]
[Izlaz]-11-[Ulaz1]
[Izlaz]-12-[Ulaz2]
[Izlaz]-13-[Ulaz3]
[Izlaz]-14-[Ulaz4]
[Izlaz]-15-[Ulaz5]
[Ulaz1]-{1.}-[Ulaz1]
[Ulaz2]-{1.}-[Ulaz2]
[Ulaz3]-{1.}-[Ulaz3]
[Ulaz4]-{1.}-[Ulaz4]
[Ulaz5]-{1.}-[Ulaz5]
[Izlaz]-{1.}-[Izlaz]
```

**Slika 10** Ispis sintakse Bootstrap metode

### 3.2.3. Razvoj modela i njegovo vrednovanje

U programskom paketu razvijeni su modeli softverskog senzora viševerižinskom linearnom regresijom (linearni viševerižinski model), viševerižinskom adaptivnom regresijom (MARSpline model) te neuronskim mrežama strukture MLP (engl. *Multi Layer Perceptrons*).

Modeli su razvijeni na dva načina:

- U prvom slučaju uzet je skup od 201 podataka za razvoj modela, dok se model vrednovao na 50 neovisnih slučajno odabranih podataka.
- U drugom slučaju, za razvoj modela, uzeto je 6.000 podataka ulaza i izlaza generiranih *bootstrap* metodom. Od 6.000 podataka, 80% je primjenjeno za razvoj modela, a model se vrednovao na preostalih 20%, odnosno 1.200 slučajno odabranih podataka. Dodatno vrednovanje modela provedeno na skupu od 251 laboratorijska podatka izlaza.

Vrednovanje modela predstavlja posljednji korak u postupku identifikacije. Usporedba razvijenih modela unutar određene skupine modela provedena je na temelju statističkih pokazatelja:

- Pearsonov koeficijent,  $R$ ;

- koeficijent determinacije,  $R^2$ ;
- korigirani koeficijent determinacije,  $\bar{R}^2$ ;
- srednja apsolutna pogreška,  $e_{MSE}$ , izražena u volumnim postotcima sadržaja benzena;
- korijen srednje kvadratne pogreške,  $RMS$ , izražen u volumnim postotcima sadržaja benzena;
- aritmetička sredina pogreške,  $\bar{e}$ , izražena u volumnim postotcima sadržaja benzena;
- standardna devijacija pogreške,  $Error \sigma$ , izražena u volumnim postotcima sadržaja benzena.

U rezultatima su navedeni i slijedeći podaci:

- aritmetička sredina podataka dobivenih modelom,  $\bar{y}$ , izražena u volumnim postotcima sadržaja benzena;
- standardna devijacija podataka dobivenih modelom,  $\sigma$ , izražena u volumnim postotcima sadržaja benzena;



### 3.2.3.1. *Linearni viševereličinski model*

U softver za razvoj linearnog modela unose se podaci od ulaznih veličina odnosno izlazna veličina. Linearnom regresijskom analizom procjenjuju se koeficijenti i „odsječak“ viševereličinske jednadžbe.

U tabličnom kalkulatoru se upiše dobivena jednadžba linearnog modela. Procijenjena izlazna veličina se računa iz jednadžbe modela uzimajući u obzir zadane ulazne veličine. U Statsoft Statistici određuju se Pearsonov koeficijent,  $R$ , koeficijent determinacije,  $R^2$ , korigirani koeficijent determinacije,  $\bar{R}^2$ , dok su u tabličnom kalkulatoru izračunati srednja kvadratna pogreška,  $RMS$ , i srednja apsolutna pogreška,  $e_{MSE}$ . Kod generiranog skupa podataka *bootstrap* metodom, model je dodatno vrednovan na realnim podacima te su na isti način kao u prethodnom slučaju određeni statistički pokazatelji.

### 3.2.3.2. *MARSpline model*

U softver za razvoj MARSpline modela se kao i kod linearnog modela odabiru ulazne odnosno izlazna veličina. Statsoft Statistica omogućava odabir maksimalnog broja čvorova, odnosno baznih funkcija, stupnja interakcija (engl. *degree of interactions*) između veličina te *penalty*. Kriterij *penalty* kao parametar izgladivanja (engl. *smoothing*) određuje „težinu“ svake bazne funkcije, odnosno što je kriterij *penalty*-ja veći, model će biti jednostavniji.

Kod razvoja modela na temelju 201 podatka, preliminarnim ispitivanjem je razvijeno više MARSpline modela s različitim parametrima. Pošto je razlika u njihovim statističkim pokazateljima valjanosti modela bila neznatna, odabran je najjednostavniji model s 8 baznih funkcija, odnosno 9 parametara modela, jednim stupnjem interakcija te kada kriterij *penalty* iznosi dva. Model je vrednovan na neovisnom skupu od 50 slučajno odabranih podataka koristeći iste kriterije valjanosti modela kao kod linearnog modela.

Kod razvoja modela na 80% bootstrapiranih podataka (4.800 podataka), odabran je model s 13 baznih funkcija, odnosno, model se sastoji od 14 parametara, jednim stupnjem interakcija te kada je kriterij *penalty* iznosio dva. Model je vrednovan na 20% (1.200 podataka) neovisnih slučajno odabranih podataka te dodatno vrednovan na realnih 251 podatak koristeći iste kriterije valjanosti modela kao kod prethodnih modela.

### 3.2.3.3. *Model neuronske mreže*

Za razvoj modela MLP neuronskih mreža, korišten je programski paket Statsoft Statistica 12.5. Kako bi se odredio optimalan broj neurona u skrivenom sloju i prijenosna funkcija za razvoj modela, preliminarno je razvijeno 1.000 neuronskih mreža, od kojih je softver po kriteriju kvadratne pogreške i koeficijentata korelacije odabrao 5 najboljih mreža. Broj skrivenih neurona je ograničen između minimalnih 3 do maksimalnih 8, što je određeno iskustvenim pravilom po kojemu broj skrivenih neurona može biti maksimalno 1,5 puta veći od broja ulaznih veličina.

Od prijenosnih (aktivacijskih) funkcija ispitivale su se eksponencijalna, sigmoidna, hiperboličko tangentna te linearna funkcija.

Ukupni skup podataka slučajnim se odabirom dijeli na 3 dijela: skup za učenje (engl. *train set*) koji sadrži 60 % podataka dok skup za provjeru podataka (engl. *testing set*) i skup za vrednovanje (engl. *validation set*) sadrže svaki po 20 % podataka. Skup podataka za učenje mreže služi za podešavanje parametara modela. Skup za vrednovanje služi kao nezavisni skup podataka za vrednovanje mreže, a u njemu se nalaze slučajno odabrani podaci koji nisu korišteni pri učenju mreže.

Iz kriterija izvedbi skupova odabrao se optimalni model u smislu kompromisa između složenosti (broja procijenjenih parametara) i kvalitete modela. Pri odabiru mreže, uzimalo se u obzir da su izvedbe u sva tri podskupa približno jednakih vrijednosti, što indicira da je vladanje neuronske mreže ujednačeno te da mreža ima veću sposobnost generalizacije. Pod generalizacijom se podrazumijeva dobra predikcija vrijednosti izlazne veličine kod podataka koji nisu korišteni pri razvoju modela, što može ukazivati na pouzdaniju primjenjivost modela u praksi.

Kod razvoja neuronske mreže na temelju 251 podatka, za odabir podataka za razvoj modela neuronske mreže u skupu za učenje korištena je *bootstrap* metoda poduzorkovanja. Bitno je napomenuti kako se kod *bootstrap* metode poduzorkovanja podaci za razvoj modela uzimaju nasumično s mogućnošću ponavljanja (tj. njihovog ponovnog korištenja) neograničen broj puta. Uobičajeno je da je skup podataka iste veličine kao zadani, ali s obzirom na prirodu metode, neće svi podaci biti izabrani.

Pri razvoju modela korištena je prijenosna funkcija i broj skrivenih neurona koji su određeni preliminarnim istraživanjem. Razvijeno je 1.000 novih mreža te je izdvojeno 5 najboljih. Prema kriterijima izvedbi i srednjim kvadratnim pogreškama pojedinih skupova, te kompromisa između složenosti i kvalitete modela odabrana je najbolja neuronska mreža.

Model je vrednovan na 20 % slučajno odabranih podataka, odnosno 50 podataka, te su rezultati dani koristeći iste kriterije valjanosti modela kao kod prethodnih modela.

Analogno, kod razvoja mreža na temelju *bootstrapiranih* podataka, razvijeno je 1.000 mreža te se od 5 najboljih prema kriteriju izvedbe i srednjim kvadratnim pogreškama te jednostavnosti odabire odgovarajuća mreža. Pri ocjeni kvalitete modela korišteni su isti statistički pokazatelji kao i pri vrednovanju prethodna dva modela, s time da je model vrednovan i na realnih 251 podatak izlaza. Računani su isti statistički parametri kao u prethodna dva tipa modela.

## 4. Rezultati i rasprava

### 4.1. Linearni modeli

Linearni viševereličinski model razvijen je na temelju 251 podatka od kojih je 201 podatak uzet za razvoj modela, a ostalih 50 za vrednovanje. Dobiveni linearni model za procjenu sadržaja benzena dan je sljedećom jednadžbom:

$$y = -5,545 - 0,019U_1 + 0,012U_2 + 0,097U_3 - 2,738U_4 + 0,004U_5 \quad (27)$$

Razvijen je i linearni model na temelju *bootstrapiranih* 6.000 podataka gdje se za razvoj modela koristilo 80% *bootstrapiranih* podataka, odnosno 4.800 podataka, a ostalih 1.200 podataka za vrednovanje. Model je dodatno vrednovan na realnih 251 podataka izlaza. Ovaj model je dan sljedećom jednadžbom:

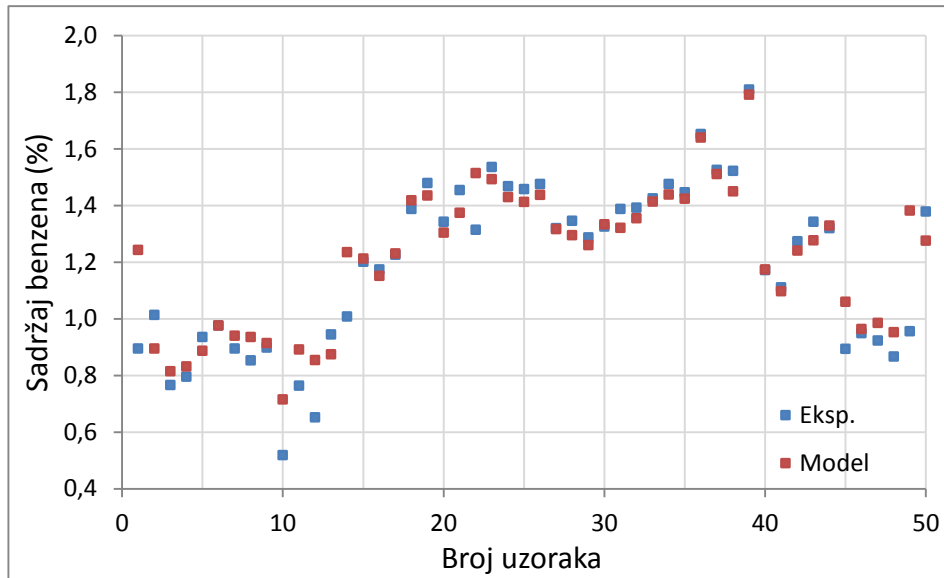
$$y = -9,644 - 0,028U_1 - 0,001U_2 + 0,094U_3 - 1,026U_4 + 0,004U_5 \quad (28)$$

**Tablica 3** Statistički pokazatelji valjanosti linearnog modela  
na malom skupu podataka

Stat. pokazatelj	Razvoj	Vrednovanje
$R$	0,881	0,928
$R^2$	0,777	0,862
$\bar{R}^2$	0,772	0,859
$e_{MSE}$ , vol. %	0,094	0,071
$RMS$ , vol. %	0,140	0,111
$e$ , vol. %	- 0,006	0,024
$Error \sigma$ , vol. %	0,140	0,110
$\bar{y}$ , vol. %	1,199	1,190
$\sigma$ , vol. %	0,285	0,289

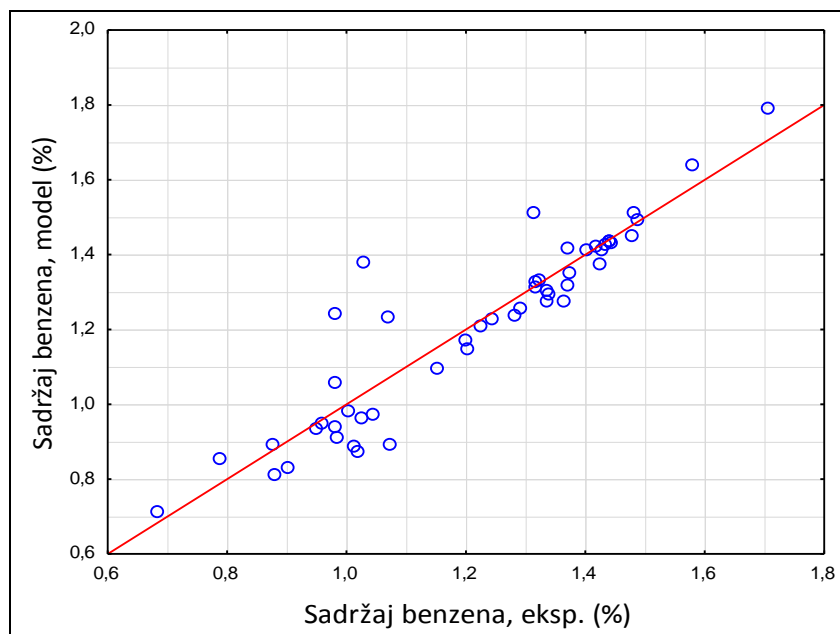
Statistički pokazatelji linearnog modela dani su u tablici 3. Prilično visoke vrijednosti korelacije te niske vrijednosti apsolutne i korijena kvadrata srednje pogreške pokazuju da se model dobro slaže s eksperimentalnim podacima.

Slika 11 prikazuje usporedbu između mjerenog izlaza i izlaza izračunatih linearnim modelom na skupu podataka za vrednovanje modela. Uočljivo je da izlaz modela dobro prati trend promjene mjerenog izlaza uz određena mala odstupanja. Ta mala odstupanja mogu se bolje vidjeti na slici 12 gdje je dana ovisnost izlaza modela i eksperimentalnih podataka. Poklapanje točaka sa pravcem  $y = x$  govori o relativno dobrom slaganju.

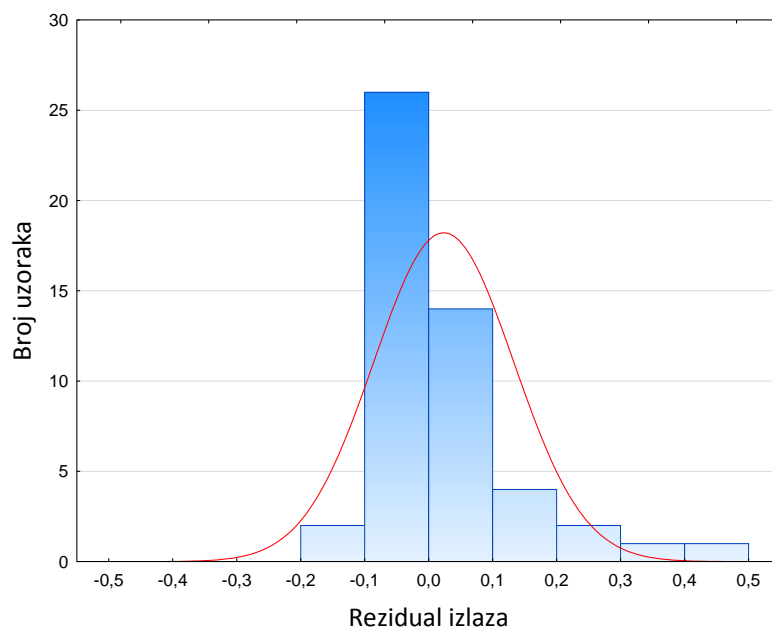


**Slika 11** Usporedba eksperimentalnih i vrijednosti dobivenih linearnim modelom na skupu za vrednovanje modela

Histogram na slici 13 daje uvid u relativne odnose izlaza modela i eksperimentalnih podataka. Najveći broj pogrešaka je između -0,1 i 0,1 vol. %, iz čega se može zaključiti da je model zadovoljavajuće prati laboratorijske rezultate te je uz statističke pokazatelje to potvrđeno i grafički. Histogram reziduala je uskog zvonolikog oblika i centriran oko nule.



**Slika 12** Odnos izlaza modela i eksperimentalnih podataka na podacima za vrednovanje modela

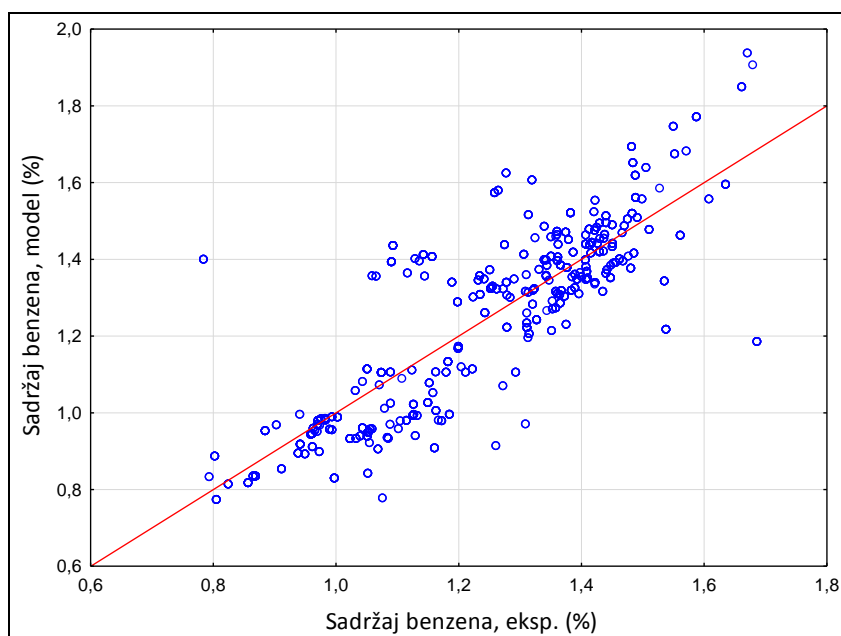


**Slika 13** Histogram reziduala izlaza linearnog modela (mali skup podataka)

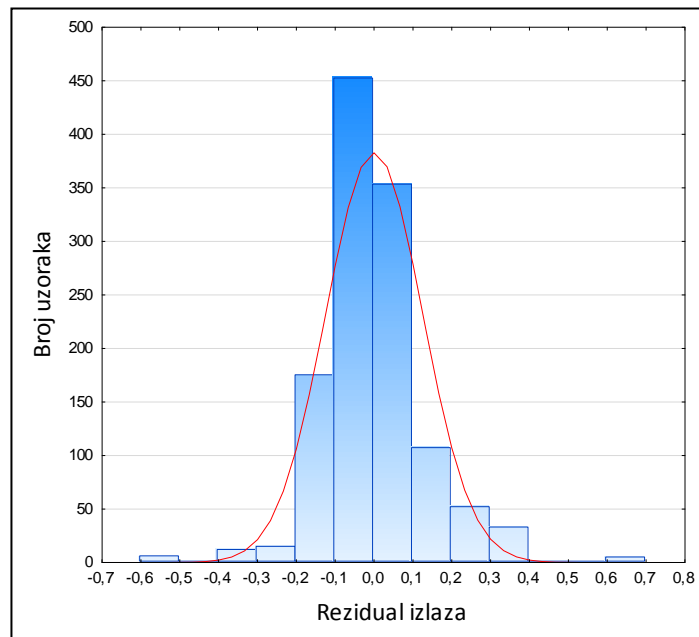
U tablici 4 dani su statistički pokazatelji valjanosti linearnog modela razvijenog i vrednovanog na 6.000 generiranih podataka *bootstrap* metodom te dodatno vrednovan na realnim podacima.

**Tablica 4** Statistički pokazatelji valjanosti linearnog modela razvijenog i vrednovanog na bootstrapiranim podacima

Pokazatelj	Razvoj	Vrednovanje	Realni podaci
$R$	0,837	0,839	0,838
$R^2$	0,700	0,704	0,702
$\bar{R}^2$	0,700	0,704	0,701
$e_{MSE}$ , vol. %	0,119	0,119	0,119
$RMS$ , vol. %	0,167	0,167	0,167
$e$ , vol. %	0,065	0,064	0,062
Error $\sigma$ , vol. %	0,154	0,154	0,156
$\bar{y}$ , vol.%	1,197	1,194	1,197
$\sigma$ , vol. %	0,281	0,283	0,285



**Slika 14** Odnos izlaza modela i eksperimentalnih podataka uspoređenih na podacima za vrednovanje (generirani skup podataka)

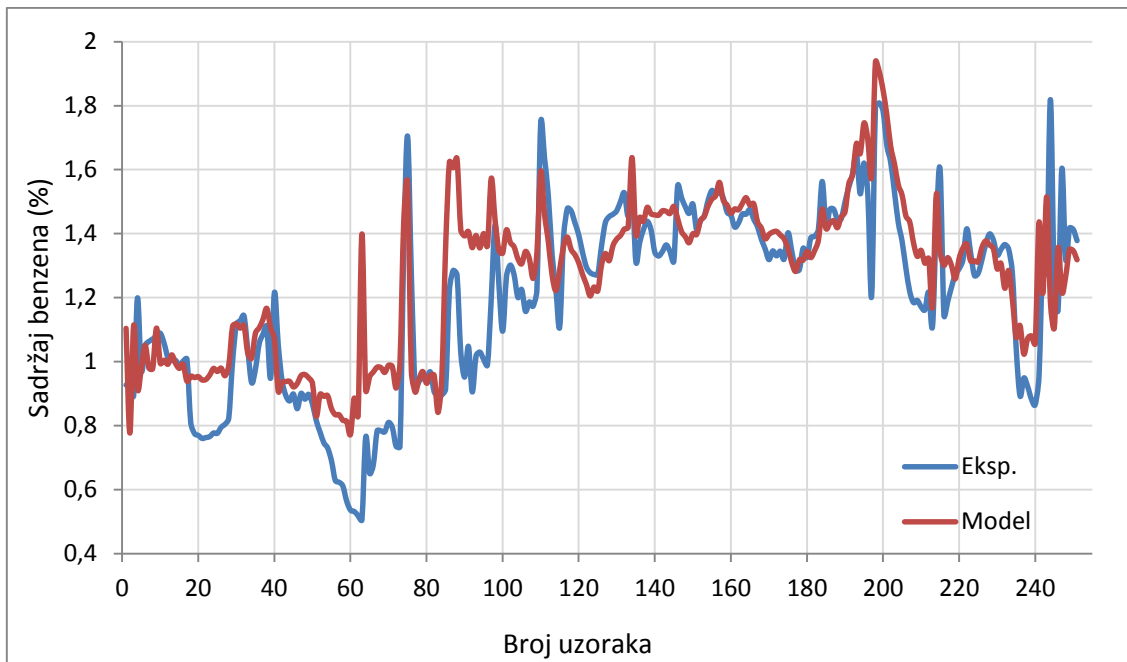


**Slika 15** Histogram reziduala izlaza linearnog modela (generirani skup podataka)

Linearni model razvijen na *bootstrapiranim* podacima pokazuje nešto lošije vladanje od prethodnog linearnog modela što se vidi i statističkih pokazatelja, tablica 4, kao i iz grafičkog prikaza ovisnosti izlaza modela i eksperimentalnih podataka na slici 14 koja pokazuje znatno veće rasipanje podataka oko pravca  $y = x$ . Na slici 15 prikazan je histogram pogreške izlaza te se to lošije vladanje potvrđuje većim udjelom pogreške manjih od -0,1 odnosno većih od 0,1 vol. % benzena. Bitno je naglasiti kako su statističke mjere kvalitete ipak vrlo slične za sva tri skupa podataka što je pokazatelj mogućnosti primjene modela, odnosno njegove generalizacije.

Na slici 16 prikazana je usporedba realnih eksperimentalnih podataka s izlazom linearnog modela razvijenog na *bootstrapiranim* podacima. Na slici se može vidjeti dobro slaganje kada je sadržaj benzena oko 1,4 vol. %. Na nižim vrijednostima sadržaja benzena model relativno dobro prati eksperimentalne vrijednosti uz određena odstupanja.





**Slika 16** Usporedba eksperimentalnih i vrijednosti dobivenih modelom višestruke linearne regresije na realnim podacima (generirani skup podataka)

## 4.2. MARSpline modeli

Pri razvoju oba modela metodom MARSpline-a primjenjeni su sljedeći kriteriji modela koji su dali optimalne rezultate s obzirom na kompleksnost modela:

- maksimalan broj čvorova = 21,
- stupanj interakcija = 1,
- *Penalty* = 2,

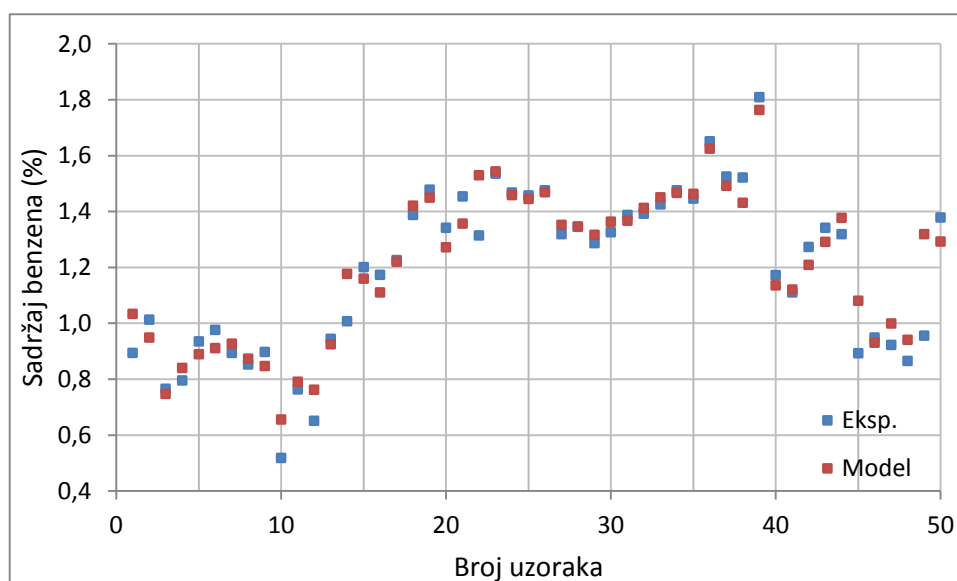
MARSpline model razvijen i vrednovan na malom skupu podataka (251 podatak) dan je jednadžbom:

$$\begin{aligned}
 y = & 0,789 + 0,185 \max(0; U_1 - 122,456) + 0,018 \max(0; 122,456 - U_1) \\
 & + 0,081 \max(0; U_3 - 125,409) - 0,135 \max(0; 125,409 - U_3) - 3,382 \max(0; U_4 - 1,995) \\
 & + 0,023 \max(0; U_5 - 41,643) - 0,591 \max(0; U_2 - 175,446) + 0,520 \max(0; U_2 - 174,998)
 \end{aligned} \quad (29)$$

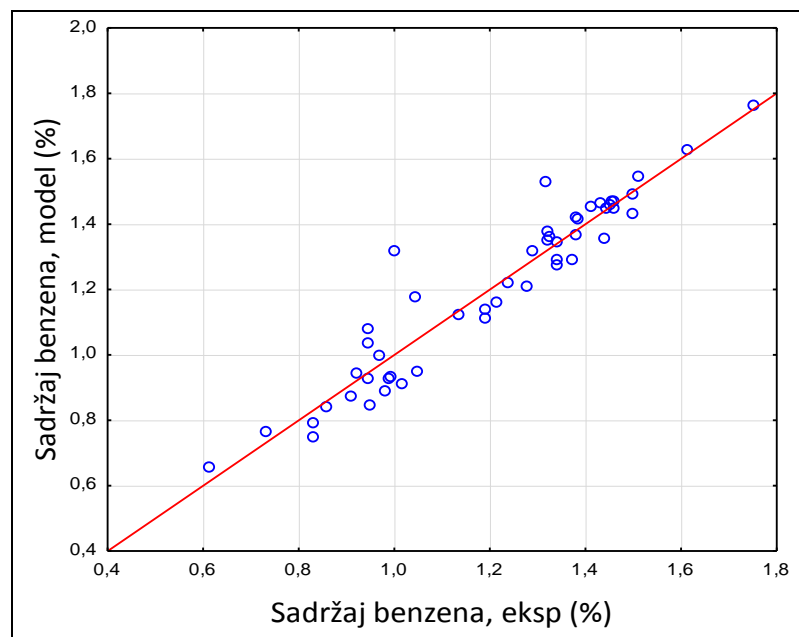
U tablici 5 dani su statistički pokazatelji valjanosti MARSpline modela (29). Iz tih pokazatelja i iz grafičkih prikaza (slike 17, 18 i 19) vidi se dobro slaganje modela s eksperimentalnim podacima.

**Tablica 5** Statistički pokazatelji valjanosti MARSpline modela razvijenog na malom skupu podataka

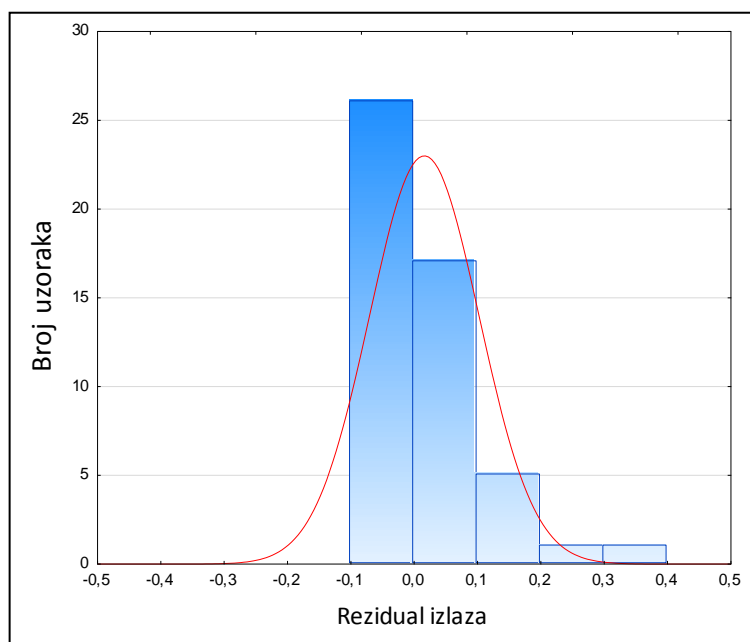
Stat. pokazatelj	Razvoj	Vrednovanje
$R$	0,896	0,954
$R^2$	0,803	0,911
$\bar{R}^2$	0,802	0,909
$e_{MSE}$ , vol. %	0,080	0,059
$RMS$ , vol. %	0,126	0,087
$e$ , vol. %	-0,004	0,016
$Error \sigma$ , vol. %	0,126	0,087
$\bar{y}$ , vol.%	1,199	1,190
$\sigma$ , vol. %	0,285	0,289



**Slika 17** Usporedba eksperimentalnih i vrijednosti dobivenih MARSpline modelom na skupu za vrednovanje modela (mali skup podataka)



**Slika 18** Odnos izlaza modela i eksperimentalnih podataka na podacima za vrednovanje (mali skup podataka)



**Slika 19** Histogram reziduala izlaza MARspline modela (mali skup podataka)

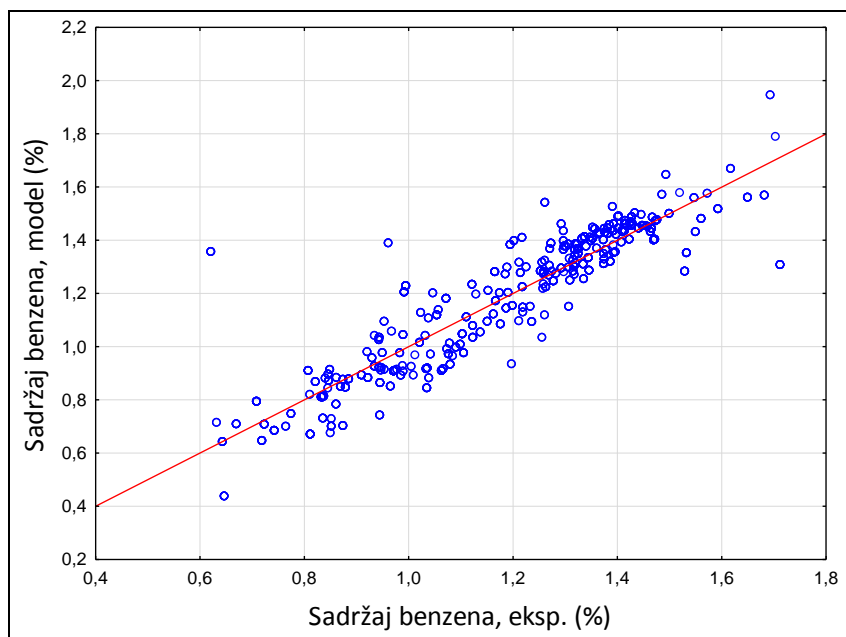
MARSpline model razvijen na generiranim podacima pomoću *bootstrap* metode dan je sljedećom jednačbom:

$$\begin{aligned}
 y = & 0,765 + 0,263 \max(0; U_5 - 47,990) + 0,008 \max(0; 47,990 - U_5) \\
 & + 2,189 \max(0; U_4 - 1,995) + 3,321 \max(0; 1,995 - U_4) - 0,043 \max(0; U_1 - 107,742) \\
 & - 0,006 \max(0; 107,742 - U_1) + 0,158 \max(0; U_3 - 121,532) + 0,127 \max(0; U_1 - 120,795) \\
 & - 0,441 \max(0; U_2 - 175,464) - 0,085 \max(0; U_3 - 125,417) + 0,153 \max(0; U_3 - 130,284) \\
 & - 0,265 \max(0; U_5 - 48,619) + 0,369 \max(0; U_2 - 174,952)
 \end{aligned} \tag{30}$$

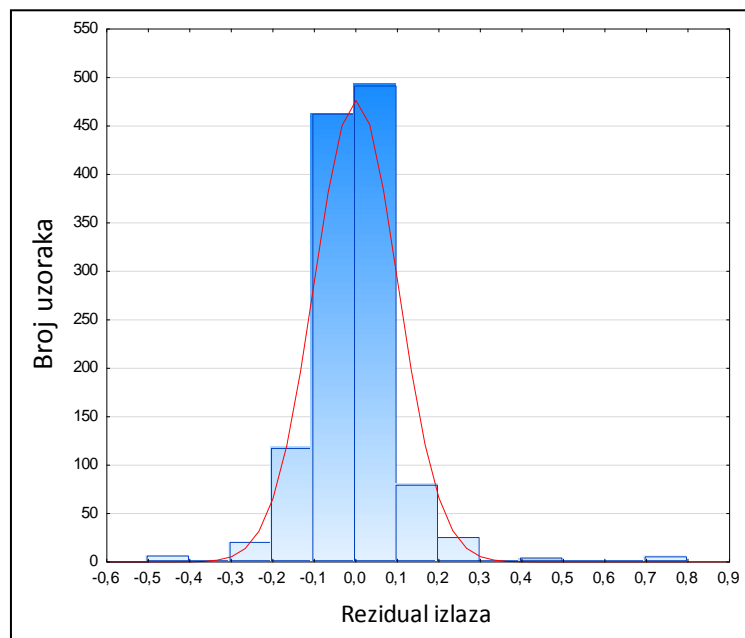
U tablici 6 nalaze se statistički pokazatelji valjanosti MARSpline modela razvijenog na generiranim podacima. Kao što je slučaj kod linearnog modela razvijenog na generiranim podacima, model u sva tri skupa podataka daje približno iste vrijednosti, ali u odnosu na linearni model, one su nešto bolje. Vrlo dobro slaganje modela i eksperimentalnih podataka može se vidjeti i na grafičkim prikazima, slika 20 i 21, na podacima za vrednovanje, odnosno na slici 22, gdje se nalazi usporedba modela i realnih eksperimentalnih podataka.

**Tablica 6** Statistički pokazatelji valjanosti MARSpline modela (generirani skup podataka)

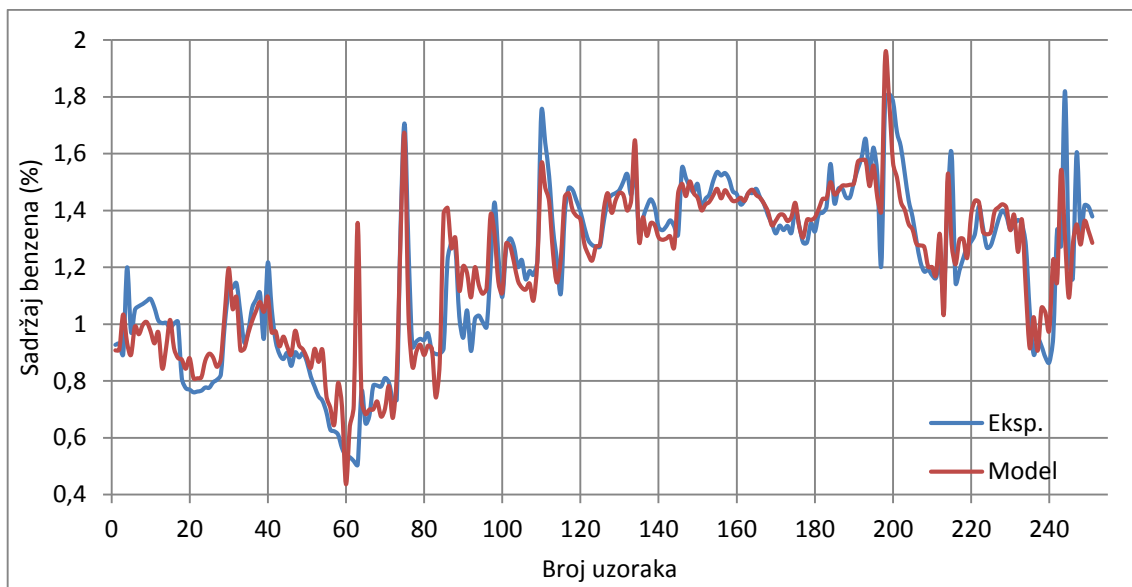
Pokazatelj	Razvoj	Vrednovanje	Realni podaci
$R$	0,915	0,918	0,915
$R^2$	0,838	0,843	0,836
$\bar{R}^2$	0,838	0,843	0,836
$e_{MSE}, vol. \%$	0,076	0,076	0,078
$RMS, vol. \%$	0,113	0,112	0,115
$e, vol. \%$	0,000	-0,001	0,000
$Error \sigma, vol. \%$	0,113	0,112	0,115
$\bar{y}, vol. \%$	1,197	1,194	1,197
$\sigma, vol. \%$	0,281	0,283	0,285



**Slika 20** Odnos izlaza MARSpline modela i eksperimentalnih podataka uspoređenih na podacima za vrednovanje (generirani skup podataka)



**Slika 21** Histogram reziduala izlaza MARSpline modela (generirani skup podataka)



**Slika 22** Usporedba eksperimentalnih i vrijednosti dobivenih MARSSpline modelom na realnim podacima (generirani skup podataka)

Usporedbom istih tipova struktura modela uočava se kako linearni i MARSSpline modeli razvijeni na temelju novo generiranih podataka *bootstrap* metodom imaju veću sposobnost generalizacije u odnosu na istovrsne modele razvijene na temelju malog skupa podataka.

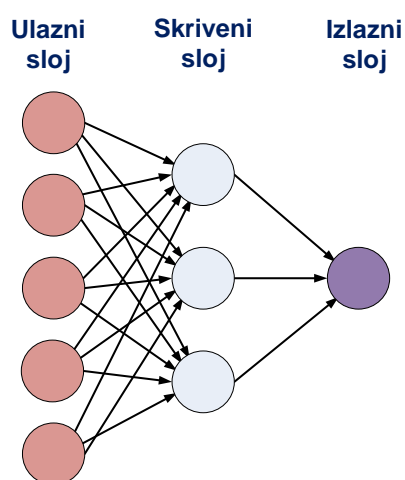
Usporedbom različitih tipova struktura modela, odnosno, linearnih modela sa MARSSpline modelima, uočava se kako MARSSpline modeli imaju užu razdiobu reziduala izlaza te bolje grafičko slaganje modela s eksperimentalnim podacima. Također su i statistički pokazatelji MARSSpline modela bolji od oba linearna modela, odnosno Pearsonovi koeficijenti i korelacijski koeficijenti su viši dok su pogreške modela niže.

### 4.3. Razvoj modela primjenom neuronske mreže

#### 4.3.1. Preliminarni razvoj modela neuronske mreže

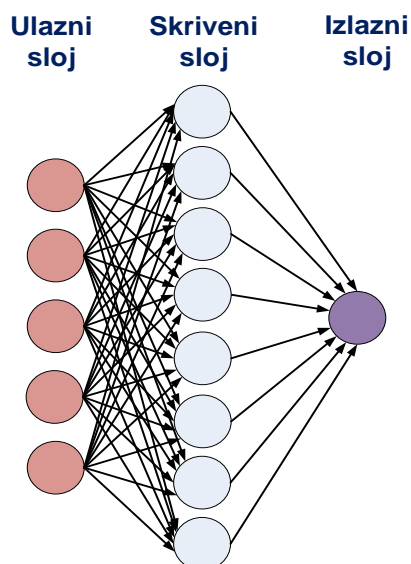
Pri razvoju modela za procjenu sadržaja benzena na malom skupu podataka (251 podatak) u preliminarnom ispitivanju odabrana je struktura mreže 5-3-1 (osjenčana u tablici 7). Ova mreža ima pet neurona u ulaznom sloju (5 ulaznih varijabli), tri u skrivenom te jedan u izlaznom sloju (izlazna varijabla). Struktura mreže dana je na slici 23.

Pri razvoju modela razvijenog na generiranih 6.000 podataka, odabrana je struktura mreže 5-8-1 (osjenčana u tablici 8). Struktura mreže prikazana je grafički na slici 24.



**Slika 23** Struktura MLP neuronske mreže za procjenu sadržaja benzena na temelju malog skupa podataka

Pri odabiru mreže uzimalo se u obzir da su izvedbe u sva tri podskupa približno jednake, što indicira da je vladanje neuronske mreže dobro te da mreža ima veću sposobnost generalizacije. U tablici 7 i 8 prikazane su 5 najboljih od 1.000 MLP mreža razvijenih u preliminarnim istraživanjima, za mali te za generirani skup podataka.



**Slika 24** Struktura MLP 5-8-1 neuronske mreže na temelju 6000 podataka

**Tablica 7** Preliminarno istraživanje za odabir strukture MLP mreže razvijene na malom skupu podataka

Struktura MLP mreže	Izvedba skupa za učenje	Izvedba skupa za provjeru	Izvedba skupa za vrednovanje	Kvadrat pogreške skupa za učenje	Kvadrat pogreške skupa za provjeru	Kvadrat pogreške skupa za vrednovanje	Akt. funk. skrivenog sloja	Akt. funk. izlaznog sloja
5-3-1	0,888	0,858	0,961	0,008	0,013	0,003	Tanh	Exp.
5-3-1	0,900	0,863	0,963	0,007	0,013	0,003	Tanh	Id.
5-3-1	0,897	0,860	0,965	0,007	0,013	0,003	Exp.	Id.
5-4-1	0,900	0,879	0,963	0,007	0,011	0,003	Exp.	Log.
5-3-1	0,885	0,857	0,965	0,008	0,013	0,003	Log.	Log.

**Tablica 8** Preliminarno istraživanje optimalne strukture MLP mreže razvijene na generiranim podacima

Struktura MLP mreže	Izvedba skupa za učenje	Izvedba skupa za provjeru	Izvedba skupa za vrednovanje	Kvadrat pogreške skupa za učenje	Kvadrat pogreške skupa za provjeru	Kvadrat pogreške skupa za vrednovanje	Akt. funk. skrivenog sloja	Akt. funk. izlaznog sloja
MLP 5-8-1	0,977	0,972	0,975	0,002	0,002	0,002	Log.	Log.
MLP 5-8-1	0,977	0,973	0,974	0,002	0,002	0,002	Tanh	Tanh
MLP 5-7-1	0,976	0,970	0,973	0,002	0,002	0,002	Log.	Tanh
MLP 5-6-1	0,972	0,967	0,971	0,002	0,002	0,002	Log.	Id.
MLP 5-8-1	0,979	0,975	0,978	0,002	0,002	0,002	Log.	Log.



### 4.3.2. Razvoj MLP neuronske mreže

Iz preliminarnih istraživanja na malom skupu podataka (251 podataka) odabrana mreža ima arhitekturu MLP 5-3-1 sa prijenosnom hiperboličko tangentnom funkcijom u skrivenom sloju. Nadalje, razvijeno je 1.000 novih mreža zadane topologije, od kojih je istaknuto 5 najboljih te je po prethodno navedenim kriterijima odabrana najbolja, tablica 9.

Iz preliminarnih istraživanja na podacima generiranim *bootstrap* metodom (6.000 podataka), odabrana je neuronska mreža arhitekture MLP 5-8-1 sa logaritamskim prijenosnim funkcijama u oba sloja. Nadalje i ovdje je razvijeno 1.000 novih mreža zadane topologije, od kojih je 5 najboljih istaknuto te je opet po istim kriterijima odabrana najbolja, tablica 10.

U tablici 11 dani su statistički pokazatelji valjanosti neuronske mreže razvijene na malom skupu podataka. Visoki korelacijski koeficijenti i male pogreške ukazuju na veoma dobro slaganje modela s eksperimentalnim podacima što pokazuju i grafički prikazi, slike 25, 26 i 27.

**Tablica 9** Odabir najbolje mreže razvijene na malom skupu podatka

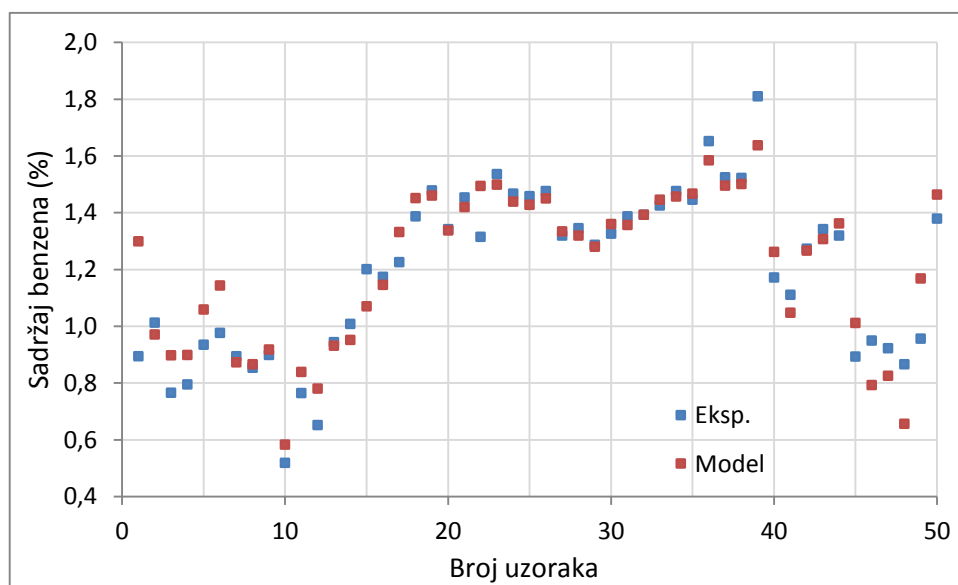
Struktura MLP mreže	Izvedba skupa za učenje	Izvedba skupa za provjeru	Izvedba skupa za vrednovanje	Kvadrat pogreške skupa za učenje	Kvadrat pogreške skupa za provjeru	Kvadrat pogreške skupa za vrednovanje	Akt. funk. skrivenog sloja	Akt. funk. izlaznog sloja
5-3-1	0,957	0,948	0,710	0,006	0,004	0,026	Tanh	Identity
5-3-1	0,944	0,922	0,934	0,007	0,005	0,005	Tanh	Identity
5-3-1	0,909	0,857	0,941	0,011	0,011	0,005	Tanh	Identity
5-3-1	0,965	0,892	0,821	0,004	0,008	0,018	Tanh	Identity
5-3-1	0,807	0,878	0,938	0,023	0,013	0,005	Tanh	Identity

**Tablica 10** Odabir najbolje mreže razvijene na generiranim podacima

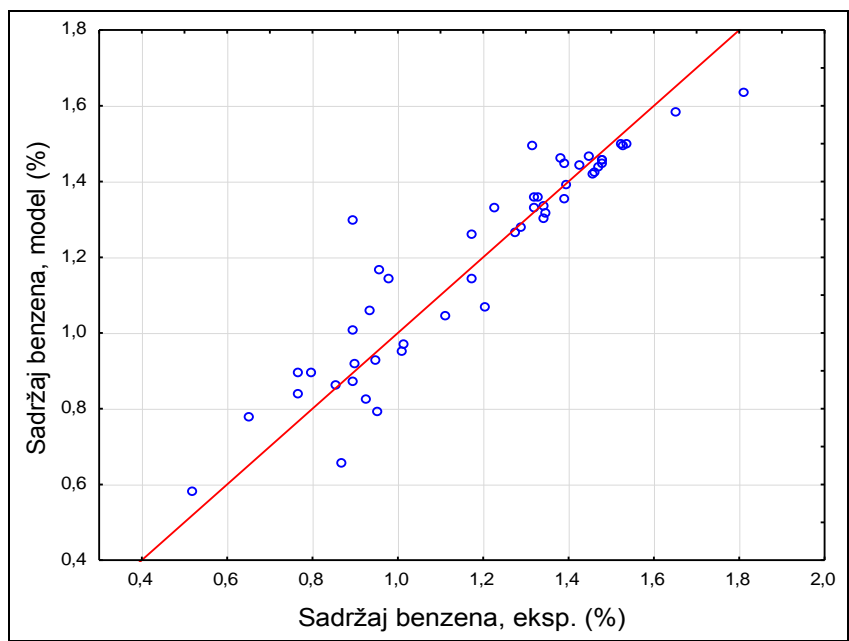
Struktura MLP mreže	Izvedba skupa za učenje	Izvedba skupa za provjeru	Izvedba skupa za vrednovanje	Kvadrat pogreške skupa za učenje	Kvadrat pogreške skupa za provjeru	Kvadrat pogreške skupa za vrednovanje	Akt. funk. skrivenog sloja	Akt. funk. izlaznog sloja
5-3-1	0,976	0,971	0,974	0,002	0,002	0,002	Log.	Log.
5-3-1	0,976	0,973	0,975	0,002	0,002	0,002	Log.	Log.
5-3-1	0,976	0,971	0,973	0,002	0,002	0,002	Log.	Log.
5-3-1	0,962	0,953	0,959	0,003	0,003	0,003	Log.	Log.
5-3-1	0,969	0,961	0,966	0,003	0,003	0,003	Log.	Log.

**Tablica 11** Statistički pokazatelji valjanosti modela neuronske mreže  
(mali skup podataka)

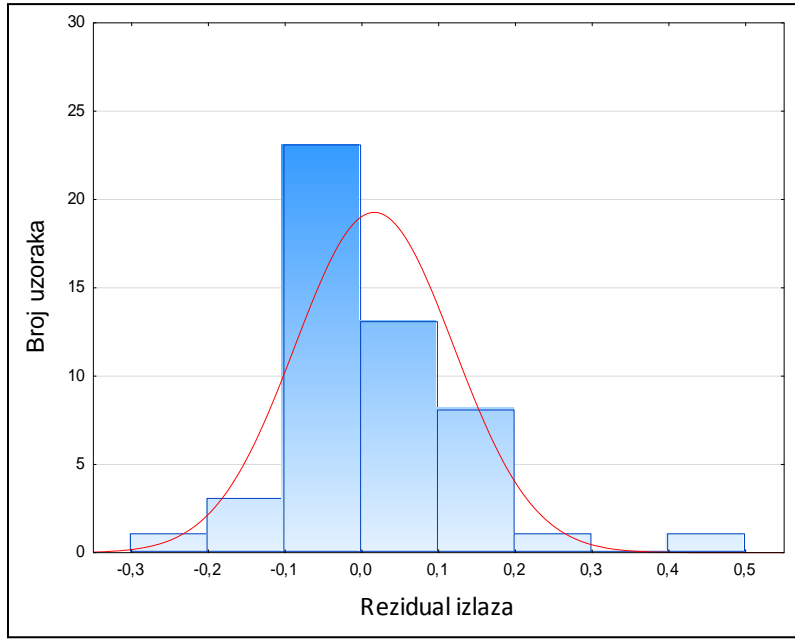
Stat. pokazatelj	Razvoj	Vrednovanje
$R$	0,932	0,934
$R^2$	0,868	0,872
$\bar{R}^2$	0,867	0,869
$e_{MSE}$ , vol. %	0,071	0,073
$RMS$ , vol. %	0,101	0,104
$e$ , vol. %	0,013	0,016
$Error \sigma$ , vol. %	0,101	0,103
$\bar{y}$ , vol. %	1,202	1,190
$\sigma$ , vol. %	0,279	0,289



**Slika 25** Usporedba eksperimentalnih i vrijednosti dobivenih modelom MLP neuronske mreže na skupu za vrednovanje modela (mali skup podataka)



**Slika 26** Odnos izlaza modela i eksperimentalnih podataka na podacima za vrednovanje (mali skup podataka)



**Slika 27** Histogram rezidual izlaza modela neuronske mreže (mali skup podataka)

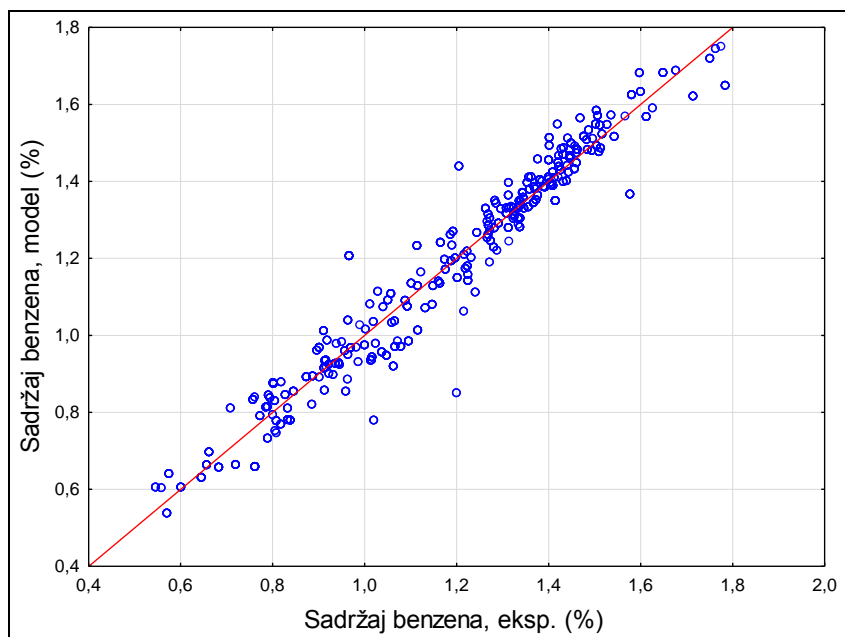
**Tablica 12** Statistički pokazatelji valjanosti MARSpline modela na generiranim podacima

Stat. pokazatelj	Razvoj	Vrednovanje	Realni podaci
$R$	0,975	0,975	0,973
$R^2$	0,951	0,951	0,947
$\bar{R}^2$	0,951	0,951	0,947
$e_{MSE}$ , vol. %	0,044	0,011	0,002
$RMS$ , vol. %	0,062	0,031	0,015
$e$ , vol. %	0,000	0,001	-0,002
$Error \sigma$ , vol. %	0,062	0,063	0,065
$\bar{y}$ , vol. %	1,197	1,194	1,197
$\sigma$ , vol. %	0,281	0,283	0,285

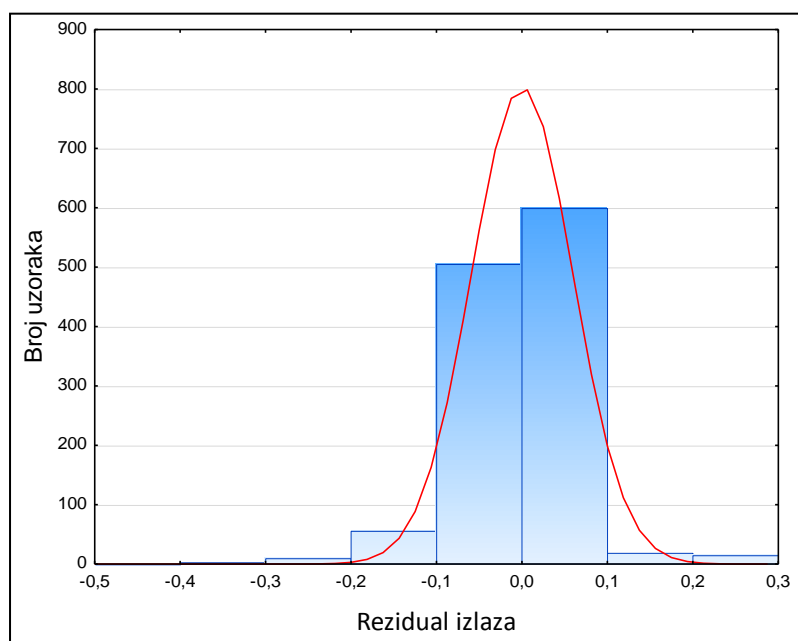
U tablici 12 dani su statistički pokazatelji valjanosti neuronske mreže razvijene na generiranim podacima. Veoma dobri statistički pokazatelji i sličnosti u vrijednostima ukazuju na to da je model veoma dobar te da je bolji od MLP modela razvijenog na malom skupu podataka.

Te pokazatelje potvrđuju i grafički prikazi, na slici 27, odstupanja vrijednosti od pravca  $y = x$  su minimalna. Na histogramu, slika 28, jasno se vidi kako se veći dio pogrešaka nalazi u rasponu od -0,1 do 0,1 vol. % benzena.

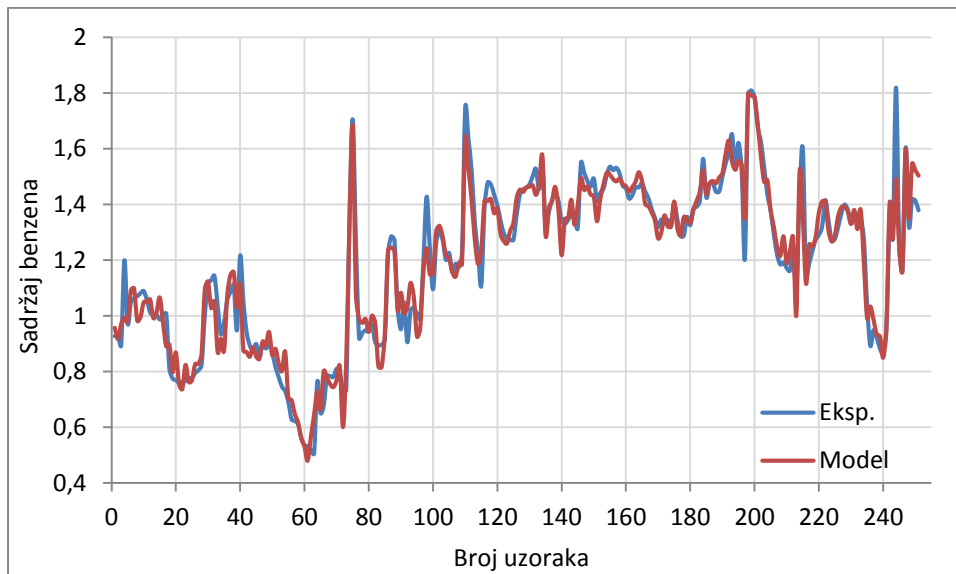
Slika 30 prikazuje usporedbu modela i eksperimentalnih podataka, na njoj se može vidjeti gotovo idealno slaganje na cijelom radnom području uz minimalna odstupanja.



**Slika 28** Odnos izlaza modela neuronske mreže i eksperimentalnih podataka na podacima za vrednovanje (generirani skup podataka)



**Slika 29** Histogram pogreške izlaza modela neuronske mreže (generirani skup podataka)



**Slika 30** Usporedba eksperimentalnih i vrijednosti dobivenih modelom neuronske mreže na realnim podacima (generirani skup podataka)

Kada se usporede svi razvijeni modeli uočava se kako najbolje rezultate daju modeli neuronskih mreža, kako onaj razvijen na malom skupu podataka tako i onaj razvijen na generiranim podacima pomoću *bootstrap* metode. Iz grafičkih prikaza histograma reziduala neuronskih modela uočava se uža raspodjela pogrešaka u odnosu na linearni i MARSpline model te oba modela neuronske mreže puno bolje prate eksperimentalne podatke.

Također, ovdje je važno uočiti veoma malu razliku Pearsonovog koeficijenta te korelacijskih koeficijenata između podataka za učenje i podataka za vrednovanje modela (za razliku od linearnog i MARSpline modela razvijenih na temelju malog skupa podataka) što govori o široj primjenjivosti neuronskih modela.

## 5. Zaključak

U ovom radu razvijeni su i uspoređeni modeli softverskih senzora za procjenu sadržaja benzena u lakom reformatu postrojenja za frakcionaciju reformata. Modeli su razvijeni na dva načina:

- Iz malog skupa eksperimentalnih podataka razvijeni su modeli viševerižinskom linearnom regresijom, metodom viševerižinske prilagodljive regresije te MLP neuronskim mrežama;
- Iz generiranog skupa podataka dobivenog iz malog skupa podataka *bootstrap* metodom. Razvijeni su modeli viševerižinske linearne regresije, modeli viševerižinske prilagodljive regresije te MLP neuronske mreže.

Prema statističkim pokazateljima, modeli razvijeni primjenom neuronskih mreža daju najbolje rezultate. Modeli razvijeni viševerižinskom linearnom regresijom te viševerižinskom prilagodljivom regresijom daju slične i isto tako zadovoljavajuće rezultate.

Usporedbom istih tipova modela razvijenih na malom skupu podataka sa modelima razvijenim na generiranim podacima, uočava se da model razvijen na malom skupu podataka pokazuje značajno različite statističke pokazatelje na skupu za razvoj modela od onih na skupu za vrednovanje modela. Time je taj model nešto manje pouzdan od svoje *bootstrapirane* inačice.

Vrednovanjem modela neuronskih mreža na realnim, a ne generiranim podacima izlaza, dokazano je da se *bootstrap* metoda može uspješno primijeniti za generiranje dodatnih podataka izlaza u svrhu dobivanja što kvalitetnijeg modela.

Rezultati istraživanja pokazali su da je razvijene softverske senzore moguće primijeniti za kontinuiranu analizu benzena u lakom reformatu i time za kvalitetniji nadzor i vođenje procesa.

## 6. Literatura

1. Bolf, N., Ivandić, M., Galinec, G.: **Soft sensors for Crude Distillation Unit Product Properties Estimation and Control**; *Proc. 16th Mediterranean Conference on Control and Automation*, Petton, R. J., Maquin, Didier (ur.), Ajaccio, IEEE, 2008., 1804-1809
2. Zamprogna, E., Barolo, M., Seborg, D. E.: **Development of a soft sensor for a batch distillation column using linear and nonlinear PLS regression techniques**; *Contr. Eng. Practice*, 12, 2004.a, 917–929
3. Kordon, A., Smits, G., Kalos, A., Jordaan, E.: **Robust soft sensor development using GP**; *Nature-Inspired Methods in Chemometrics*, R. Leardi, Ed. Amsterdam, The Netherlands: Elsevier, 2003., 69–108
4. Qin, S. J.: **Neural networks for intelligent sensors and control – Practical issues and some solutions**; *Neural Syst. Contr.*, 1997., 213–234
5. Han, C., Lee, Y.H.: **Intelligent integrated plant operation system for six sigma**; *Annu. Rev. Contr.*, 26 (1), 2002., 27–43
6. Martin, G.D.: **Consider Soft Sensors**, *Chem. Eng. Prog.* 7, 1997., 66-70
7. Quek, C. J., Balasubramanian, R., Rangaiah, G. P.: **Hydrocarbon Process.**; Int. Ed. 79, 2000, 101-106
8. Fortuna, L., Graziani, S., Rizzo, A., Xibilia, M. G.: **Soft Sensors for Monitoring and Control of Industrial Processes (Advances in Industrial Control)**, Springer, London, 2007.
9. Fortuna, L., Rizzo, A., Sinatra, M., Xibilia, M. G.: **Soft analyzers for a sulfur recovery unit**, *Control Eng. Pract.* 11, 2003., 1491-1500
10. Zahedi, G., Elkamel, A., Lohi, A., Jahanmiri, A., Rahimpour, M. R.: **Hybrid artificial neural network - First principle model formulation for the unsteady state simulation and analysis of a packed bed reactor for CO<sub>2</sub> hydrogenation to methanol**, *Chem. Eng. J.* 115, 2005., 113-120
11. Galinec, G.: **Razvoj softverskih senzora za identificiranje i inferencijsko vođenje rafinerijskih procesa**, doktorska disertacija, Fakultet kemijskog inženjerstva i tehnologije Sveučilišta u Zagrebu, Zagreb, 2010.
12. Ujević Andrijić, Ž.: **Softverski senzori za identifikaciju i vođenje nelinearnih procesa**, doktorska disertacija, Fakultet kemijskog inženjerstva i tehnologije Sveučilišta u Zagrebu, Zagreb, 2012.
13. Ljung, L.: **System Identification: Theory for the User**, 2nd ed., Prentice Hall, New Jersey, 1999.



14. Lanuette, R., Thibault, J., Valade, J. L.: **Process modeling with neural networks using small experimental datasets**, *Comput. Chem. Eng.* 23, 1997, 1167 - 1176
15. Di Bella, A., Fortuna, L., Graziani, S., Napoli, G., Xibilia, M. G.: **Development of a Soft Sensor for a Thermal Cracking Unit using a small experimental data set**, Intelligent Signal Processing, IEEE, WISP 2007.
16. <http://www.math.ntu.edu.tw/~hchen/teaching/LargeSample/notes/notebootstrap.pdf> (pristupljeno 19. svibnja 2015.)
17. <http://www.nature.com/articles/hortres20159/figures/3> (pristupljeno 19. svibnja 2015.)
18. Warne, K., Prasad, G., Rezvani, S., Maguire, L.: **Statistical and computational intelligence techniques for inferential model development: a comparative evaluation and a novel proposition for fusion**, *Eng. Appl. Artif. Intell.* 17, 2004., 871-885
19. Rallo, R., Ferre-Giné, J., Arena, A., Giralt, F.: **Neural virtual sensor for the inferential prediction of product quality from process variables**, *Comput. Chem. Eng.* 26, 2002., 1735-1754
20. Nagai, E. Y., Arruda, L. V. R.: **Soft sensor based on fuzzy model identification**, *Proc. 16th IFAC World Congress*, Prague, 2005, CD-ROM
21. Kourti, T.: **Process analysis and abnormal situation detection: from theory to practice**; *IEEE Contr. Syst. Mag.* 22, 2002., 10-25
22. Lee, J. M., Yoo, C., Lee, I. B.: **Statistical process monitoring within dependent component analysis**, *J. Process Control* 14, 2004., 467-485
23. <http://www.statsoft.com/textbook/multiple-regression> (pristupljeno 10. svibnja 2015.)
24. Baotić, M.: **Identifikacija i upravljanje nelinearnim vremenski promjenljivim procesima primjenom neuronskih mreža**, magistarski rad, Fakultet elektrotehnike i računarstva Sveučilišta u Zagrebu, Zagreb, 2000.
25. Bolf, N.: **Prilagodljivo koordinirano vođenje složenih procesa**, doktorska disertacija, Fakultet kemijskog inženjerstva i tehnologije Sveučilišta u Zagrebu, Zagreb, 2003.
26. **Neural Ware: Neural Computing - A Technology Handbook for Professional II / Plus and Neural Works Explorer**, Pittsburgh, PA: NeuralWareInc., 2003.
27. [www.hzt.hr](http://www.hzt.hr), stranice hrvatskog zavoda za toksikologiju i antidoping (pristupljeno 20. ožujak 2015.)

## ŽIVOTOPIS

Matija Cvetnić rođen je u Zagrebu 29. srpnja 1991. Pohađao je osnovnu školu Eugena Kumičića u Velikoj Gorici, a 2006. je upisao Prirodoslovno-matematičku gimnaziju u Velikoj Gorici. Tokom srednjoškolskog obrazovanja sudjelovao je na brojnim znanstvenim i sportskim natjecanjima.

2010. godine upisao se na preddiplomski studij *Kemijsko inženjerstvo* Fakulteta kemijskog inženjerstva i tehnologije. 2013. godine obranio je završni rad pod naslovom Natrijev acetat trihidrat kao niskotemperaturni fazno-promjenjivi materijal pod mentorstvom profesora dr. sc. Marka Rogošića.

2013. godine upisao se na diplomski studij Fakulteta kemijskog inženjerstva i tehnologije, smjer *Kemijsko inženjerstvo*. Te akademske godine sudjeluje na 10. *Susretu mladih kemijskih inženjera* sa temom: *Optimiranje rada sustava za vođenje procesa izmjene topline* pod mentorstvom profesora Nenada Bolfa. 2015. godine sudjeluje na 24. *Susretu kemičara i kemijskih inženjera* s radom *Razvoj modela za procjenu sadržaja aromata na temelju malih skupova podataka*. Tijekom diplomskog studija sudjeluje na više projektima pod vodstvom izv.prof. dr. sc. Nenada Bolfa i doc. dr. sc. Igora Dejanovića.