

Obrada spam mail-ova metodom klasifikacije

Tijana Todorov

26. avgust 2019

Pregled

- ① Uvod
- ② Upoznavanje sa podacima
- ③ Obrada podataka
- ④ Primenjeni algoritmi
 - Drveta odlučivanja
 - Najbliži susedi - KNN
 - Neuronske mreže
- ⑤ Metod potpornih vektora - SVM
- ⑥ Gausova klasifikacija
- ⑦ Zaključak
- ⑧ Hvala na pažnji

Uvod

- Spam poruke su zapravo neželjena pošta koja primaocu samo zatrpava sanduče
- Ona može biti nešto što primaoca ne zanima (reklama, online prodavnica...)
- Mogu predstavljati opasnost za primaoca ukoliko su zaražene virusom
- Cilj je napraviti dobar spam filter koji će odvajati očekivane i neočekivane poruke

Upoznavanje sa podacima

- Podaci se nalaze na https://web.stanford.edu/~hastie/CASI_files/DATA/SPAM.html pod nazivom **SPAM.csv**
- 4601 email - 1813 prijavljenih spam poruka
- 59 atributa
 - 57 numeričkih - najčešće korišćene reči
 - 2 kategorička binarna atributa - **spam** i **testid**
- nema null podataka

Podaci

- **spam** - označava da li je pošta neželjena ili ne
- **testid** - označava da li se instanca nalazi u trening ili test skupu
- 48 atributa - procenat pojavljivanja reči
 $100 * \text{broj_pojavljivanja_reči} / \text{ukupan_broj_reči}$
- 6 atributa (ch; , ch(, ch[, ch! , ch\$, ch#) - procenat pojavljivanja karaktera
 $100 * \text{broj_pojavljivanja_karaktera} / \text{ukupan_broj_karaktera}$
- **crl.ave** - označava prosečnu dužinu neprekidnih nizova velikih slova.
- **crl.long** - označava dužinu najduže sekvence velikih slova.
- **crl.tot** - ukupan broj velikih slova u email poruci.

Obrada podataka - SPSS

- Razlika u opsezima podataka
 - `crl.tot` - [1 - 15841]
 - `make` - [0 - 4.54]
- Izabrani neki atributi iz celog skupa (`all`, `remove`, `internet`, `mail`, `addresses`, `business`, `money`)
- Vrš se normalizacija pomoću čvora **`_Norm`**

Obrada podataka - Python

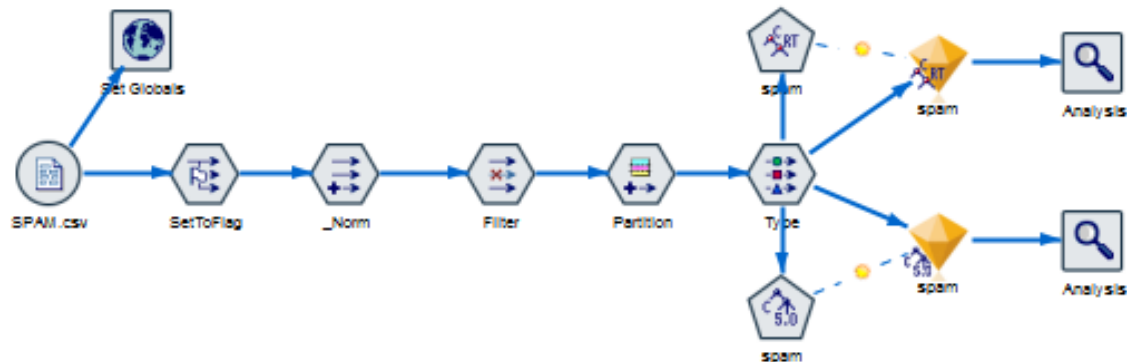
```
1 booleandf = df.select_dtypes(include=[bool])
2 booleanDictionary = {True: "tacno", False: "netacno"}
3
4 for column in booleandf:
5     df[column] = df[column].map(booleanDictionary)
6
7 features1 = df.columns[0]
8 features5 = df.columns[4]
9 features9 = df.columns[8]
10 features10 = df.columns[9]
11 features12 = df.columns[11]
12 features16 = df.columns[15]
13 features17 = df.columns[16]
14 features19 = df.columns[18]
15 features26 = df.columns[25]
16 features = [features5, features9, features10, features12,
17             features16, features17, features19, features26]
18 x_original = df[features]
19 x = pd.DataFrame(x原, MinMaxScaler().fit_transform(x_original))
```

Primenjeni algoritmi

- Drveta odlučivanja (SPSS, Python)
- Najbliži susedi - KNN (SPSS, Python)
- Neuronske mreže (SPSS, Python)
- Metod potpornih vektora - SVM (SPSS)
- Gausova klasifikacija (Python)

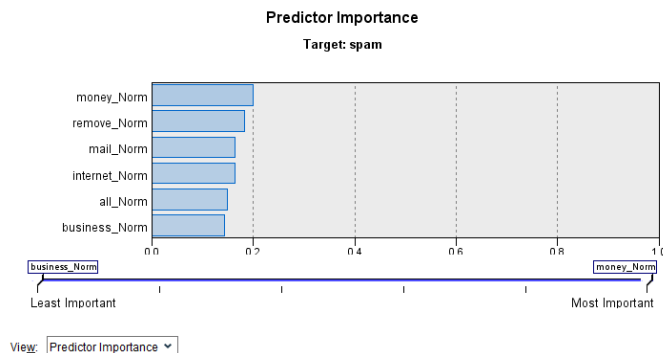
C5.0 i C&Rt

- Model se pravi pomoću čvora C5.0
- Model se pravi pomoću čvora C&Rt



Slika 1 : C5.0 i C&Rt u SPSS-u

C5.0



Results for output field spam

Comparing SC-spam with spam

'Partition'	1_Training	2_Testing
Correct	2,743 85.77%	1,197 85.32%
Wrong	455 14.23%	206 14.68%
Total	3,198	1,403

Coincidence Matrix for SC-spam (rows show actuals)

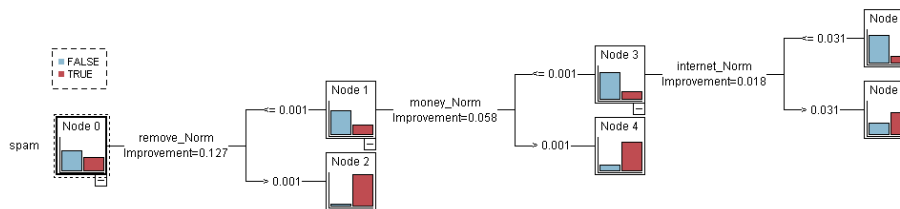
'Partition' = 1_Training		FALSE	TRUE
FALSE	FALSE	1,833	113
	TRUE	342	910
'Partition' = 2_Testing		FALSE	TRUE
FALSE	FALSE	783	59
	TRUE	147	414

Slika 2 : Rezultati C5.0 algoritma dobijeni Analyse čvorom u SPSS-u

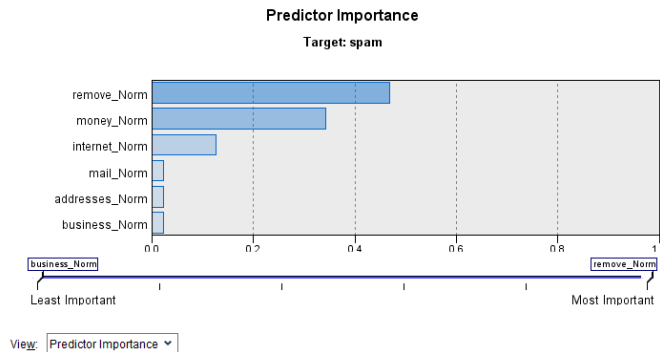
C&Rt

Drvo dobijeno primenom modela C&Rt sa sledećim karakteristikama:

- Dubina 3
- Broj instanci roditelja 4% a deteta 2%
- Mera nečistoće - Gini, 0.0001%



Slika 3 : Stablo dobijeno C&Rt algoritmom u SPSS-u



■ Results for output field spam

■ Comparing \$R-spam with spam

'Partition'	1_Training		2_Testing	
Correct	2,661	83.21%	1,188	84.68%
Wrong	537	16.79%	215	15.32%
Total	3,198		1,403	

■ Coincidence Matrix for \$R-spam (rows show actuals)

'Partition' = 1_Training		FALSE	TRUE
FALSE	FALSE	1,818	128
	TRUE	409	843
'Partition' = 2_Testing		FALSE	TRUE
FALSE	FALSE	793	49
	TRUE	166	395

Slika 4 : Rezultati C&Rt algoritma dobijeni Analyse čvorom u SPSS-u

Drveta odlučivanja - Python matrice konfuzije

```
1 #Skup  Trening
2 #Matrica konfuzije
3         netacno  tacno
4 netacno      1882    69
5 tacno        374   895
6
7 Preciznost  0.8624223602484472
8
9 #Skup  Test
10 #Matrica konfuzije
11         netacno  tacno
12 netacno      802    35
13 tacno        161   383
14
15 Preciznost  0.8580738595220855
```

Drveta odlučivanja - Python izveštaji klasifikacije

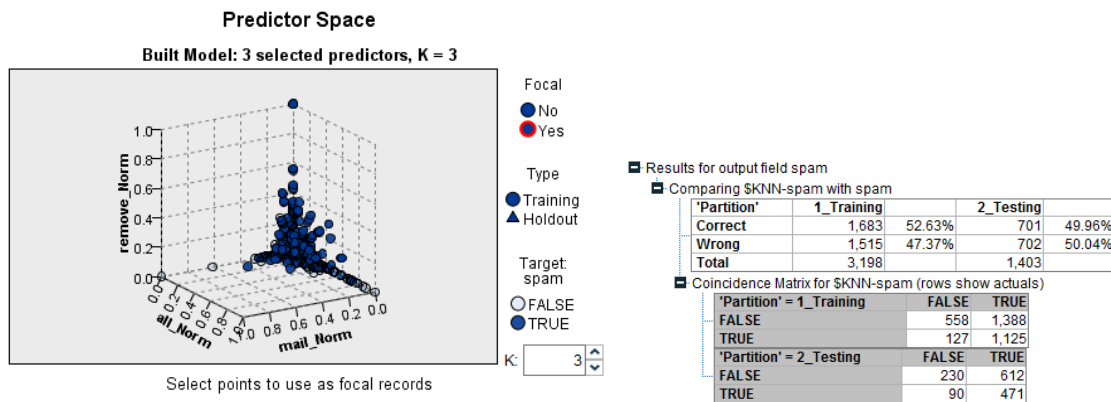
```
1 #Skup Trening
2 #Izvestaj klasifikacije
3         precision    recall  f1-score   support
4     netacno         0.83      0.96      0.89     1951
5     tacno           0.93      0.71      0.80     1269
6
7     micro avg       0.86      0.86      0.86     3220
8     macro avg       0.88      0.83      0.85     3220
9     weighted avg    0.87      0.86      0.86     3220
10 #Skup Test
11 #Izvestaj klasifikacije
12         precision    recall  f1-score   support
13     netacno         0.83      0.96      0.89      837
14     tacno           0.92      0.70      0.80      544
15
16     micro avg       0.86      0.86      0.86     1381
17     macro avg       0.87      0.83      0.84     1381
18     weighted avg    0.87      0.86      0.85     1381
```

Najbliži susedi - KNN

- Sređene podatke povezujemo sa *Partition* čvorom i generišemo model pokretanjem čvora *KNN*.
- Ciljni atribut je *spam*
- Minimalni broj k je 3 a maksimalan 5
- Udaljenost se računa Euklidskim rastojanjem



Najbliži susedi - KNN



Slika 6 : Grafikom prikazani rezultati KNN-a i Analize čvora u SPSS-u

Najbliži susedi KNN - Python

- Trening skup je postavljen na 70%.
- Najbolji rezultati dobijaju se za $k = 4$
- Euklidsko rastojanje
- Svi susedi imaju podjednak uticaj

Najbliži susedi KNN - Python

```
1 weights_values = ['uniform', 'distance']
2 #uniform
3 #Matrica konfuzije
4 [[793  44]
5  [209 335]]
6
7 Preciznost 0.8167994207096307
8 #Izvestaj klasifikacije:
9
10      precision    recall  f1-score   support
11
12 netacno      0.79      0.95      0.86      837
13 tacno        0.88      0.62      0.73      544
14
15 micro avg      0.82      0.82      0.82     1381
16 macro avg      0.84      0.78      0.79     1381
17 weighted avg      0.83      0.82      0.81     1381
```

Najbliži susedi - KNN

```
1
2 #distance
3 #Matrica konfuzije
4 [[770  67]
5  [163 381]]
6
7 Preciznost 0.833454018826937
8 #Izvestaj klasifikacije:
9               precision    recall  f1-score   support
10
11      netacno           0.83      0.92      0.87       837
12      tacno            0.85      0.70      0.77       544
13
14      micro avg          0.83      0.83      0.83      1381
15      macro avg          0.84      0.81      0.82      1381
16      weighted avg       0.84      0.83      0.83      1381
```

Neuronske mreže - SPSS

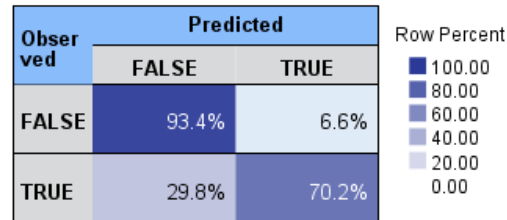


Slika 7 : Model Neuronske mreže u SPSS-u

Neuronske mreže - SPSS

Classification for spam

Overall Percent Correct = 84.3%



Slika 8 : Matrica konfuzije za neuronske mreže

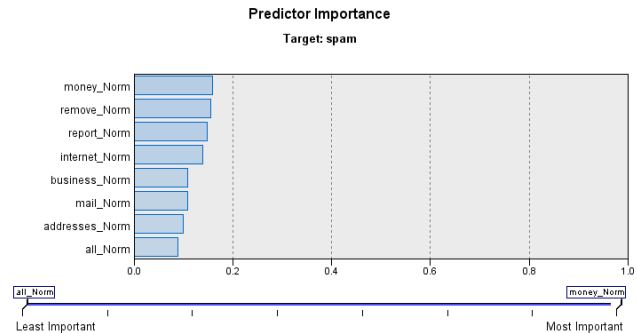
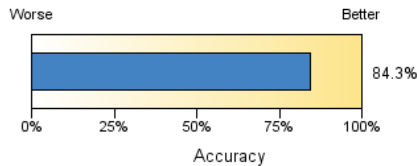
Uvod
 Upoznavanje sa podacima
 Obrada podataka
 Primljeni algoritmi
 Metod potpornih vektora - SVM
 Gausova klasifikacija
 Zaključak
 Hvala na pažnji

Drвета odlučivanja
 Najbliži susedi - KNN
 Neuronske mreže

Neuronske mreže - SPSS

Model Summary

Target	spam
Model	Multilayer Perceptron
Stopping Rule Used	Error cannot be further decreased
Hidden Layer 1 Neurons	4

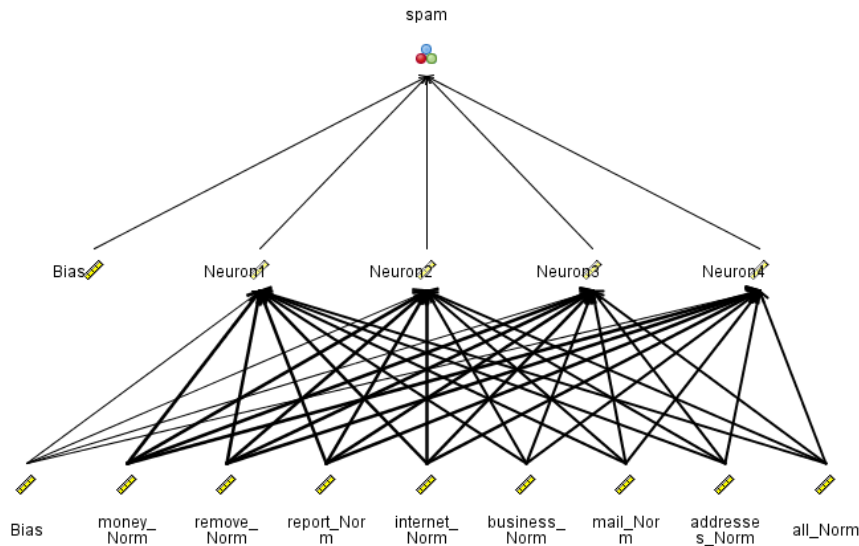


Slika 9 : Neuronska mreža

Uvod
Upoznavanje sa podacima
Obrada podataka
Primenjeni algoritmi
Metod potpornih vektora - SVM
Gausova klasifikacija
Zaključak
Hvala na pažnji

Drвета odlučivanja
Najbliži susedi - KNN
Neuronske mreže

Neuronske mreže - SPSS



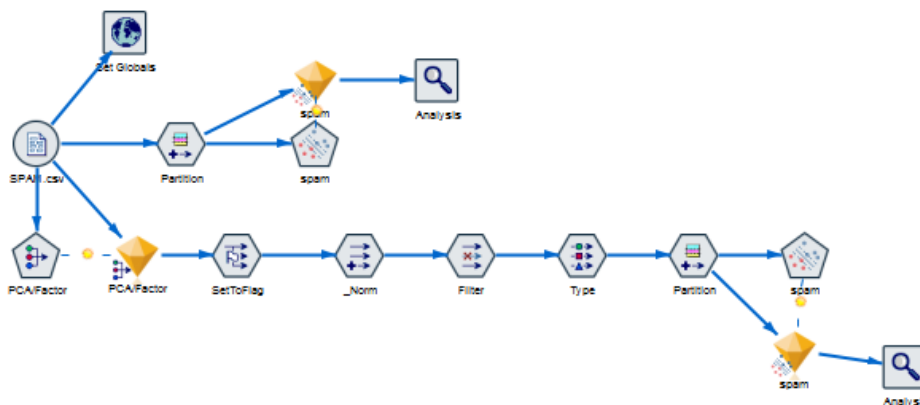
Slika 10 : Neuronska mreža

Neuronske mreže - Python

```
1 #Izvestaj za test skup:
2 #Matrica konfuzije
3 [[786  51]
4  [162 382]]
5 Preciznost 0.8457639391745112
6 #Izvestaj klasifikacije
7           precision    recall  f1-score   support
8   netacno           0.83      0.94      0.88         837
9   tacno             0.88      0.70      0.78         544
10
11   micro avg          0.85      0.85      0.85        1381
12   macro avg          0.86      0.82      0.83        1381
13   weighted avg       0.85      0.85      0.84        1381
14
15 Broj iteracija:  275
16 Broj slojeva:   4
```

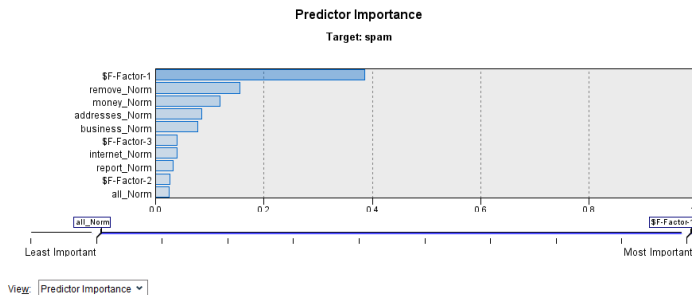

Metod potpornih vektora - SVM

- PCA čvor koristimo da bi smanjili skup podataka
- SVM čvor pokrećemo i rezultat analiziramo sa Analyse čvorom



Slika 11 : Metod potpornih vektora - SVM u SPSS-u

Metod potpornih vektora - SVM



Results for output field spam

Comparing \$S-spam with spam

'Partition'	1_Training		2_Testing	
Correct	2,555	79.89%	1,126	80.26%
Wrong	643	20.11%	277	19.74%
Total	3,198		1,403	

Coincidence Matrix for \$S-spam (rows show actuals)

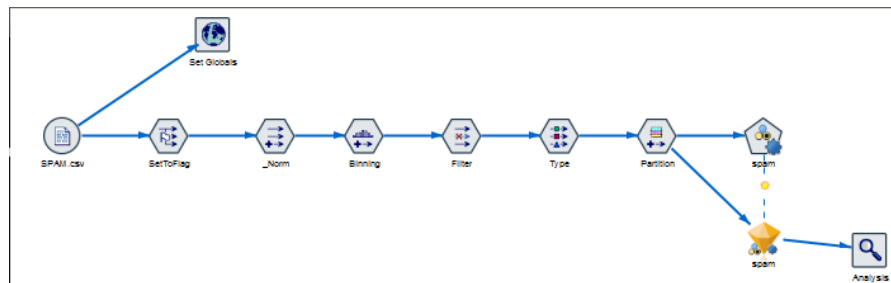
'Partition' = 1_Training	FALSE	TRUE
FALSE	1,816	130
TRUE	513	739
'Partition' = 2_Testing	FALSE	TRUE
FALSE	778	64
TRUE	213	348

Slika 12 : Rezultati SVM algoritma dobijeni Analyse čvorom u SPSS-u

Gausova klasifikacija





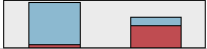



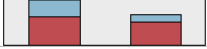

```
1 #Matrica konfuzije
2 [[808  29]
3  [306 238]]
4
5 Preciznost 0.7574221578566256
6 #Izvestaj klasifikacije
7           precision    recall  f1-score   support
8
9      tacno           0.73      0.97      0.83       837
10     netacno          0.89      0.44      0.59       544
11
12    micro avg          0.76      0.76      0.76      1381
13    macro avg          0.81      0.70      0.71      1381
14 weighted avg          0.79      0.76      0.73      1381
```

Zaključak



Slika 13 : Poređenje svih algoritama u SPSS-u pomoću čvora **Auto Classifier**

Zaključak

Use?	Graph	Model	Build Time (mins)	Overall Accuracy (%)	No. Fields Used
<input checked="" type="checkbox"/>		 C&R Tree 1	< 1	86.386	8
<input checked="" type="checkbox"/>		 Neural Net 1	< 1	85.460	8
<input checked="" type="checkbox"/>		 C5 1	< 1	85.317	7
<input checked="" type="checkbox"/>		 SVM 1	< 1	80.969	8
<input checked="" type="checkbox"/>		 KNN Algorithm 1	< 1	53.457	8

Slika 14 : Rezultat Analize čvora nad čvorom koji poredi sve metode

Uvod
Upoznavanje sa podacima
Obrada podataka
Primenjeni algoritmi
Metod potpornih vektora - SVM
Gausova klasifikacija
Zaključak
Hvala na pažnji

Hvala na pažnji!