

# Obrada spam mail-ova metodom klasifikacije

Tijana Todorov

27. avgust 2019

# Pregled

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

# 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](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 for column in booleandf:
4     df[column] = df[column].map(booleanDictionary)
5
6 features1 = df.columns[0]
7 features5 = df.columns[4]
8 features9 = df.columns[8]
9 features10 = df.columns[9]
10 features12 = df.columns[11]
11 features16 = df.columns[15]
12 features17 = df.columns[16]
13 features19 = df.columns[18]
14 features26 = df.columns[25]
15 features = [features5, features9, features10, features12,
16             features16, features17, features19, features26]
17 x_original = df[features]
18 x=pd.DataFrame(prepare.MinMaxScaler().fit_transform(x_original)
19               )
```

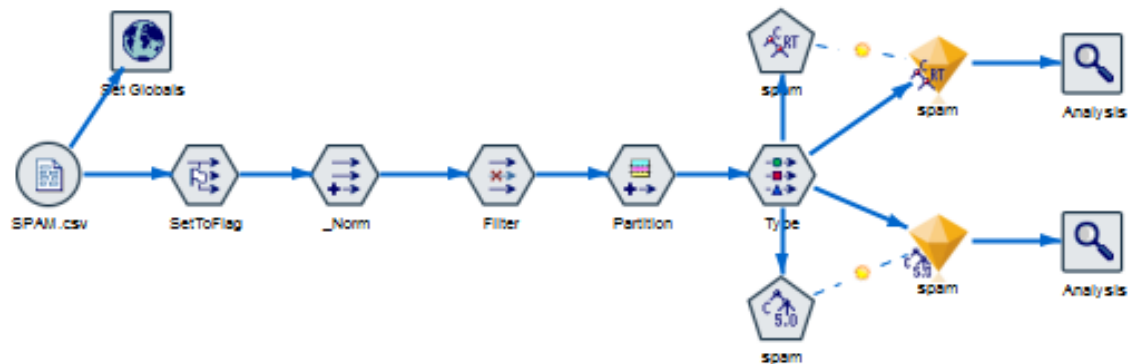
# Primenjeni algoritmi

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



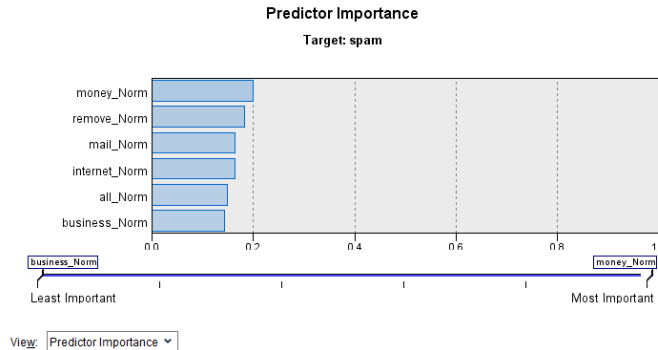
## C5.0 i C&Rt

- 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)

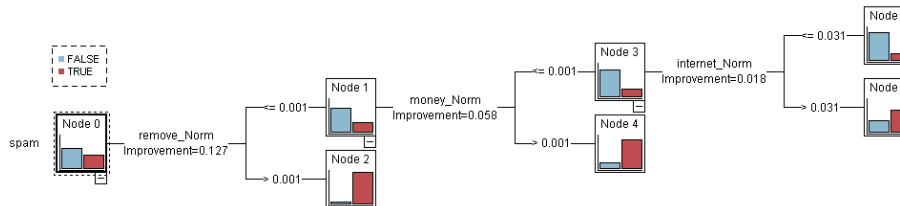
'Partition' = 1_Training	FALSE	TRUE
FALSE	1,833	113
TRUE	342	910
'Partition' = 2_Testing	FALSE	TRUE
FALSE	783	59
TRUE	147	414

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

# C&Rt

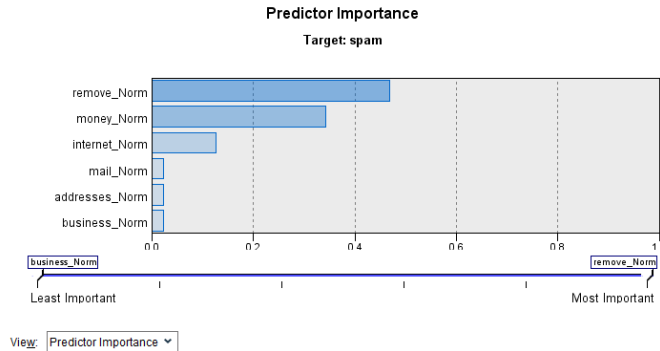
Drvo dobijeno primenom modela C&Rt sa sledećim karakteristikama:

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



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

# C&Rt



## 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		1,818	128
TRUE		409	843
'Partition' = 2_Testing		FALSE	TRUE
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 #Skup Test
8 #Izvestaj klasifikacije
9         precision    recall  f1-score   support
10  netacno         0.83      0.96      0.89      837
11  tacno          0.92      0.70      0.80      544
```

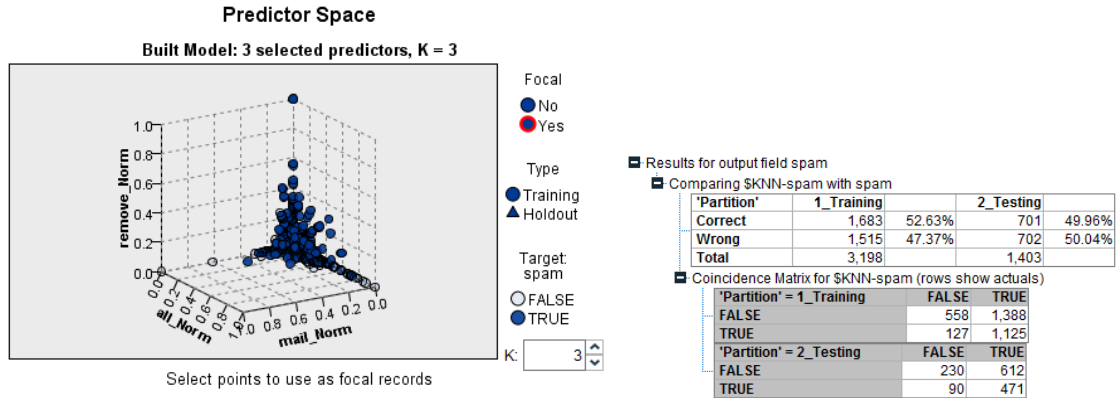
## Najbliži susedi - KNN

- 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



Slika 5 : Analiza modela k najbližih suseda - SPSS

# 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
```

# Najbliži susedi - KNN

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

# Neuronske mreže - SPSS



Slika 7 : Model Neuronske mreže u SPSS-u

## Neuronske mreže - SPSS


- Model se generiše pokretanjem čvora *Neural Net*
- Za cilj je odabrano kreiranje novog višeslojnog modela

### Classification for spam

Overall Percent Correct = 84.3%

Observed	Predicted	
	FALSE	TRUE
FALSE	93.4%	6.6%
TRUE	29.8%	70.2%

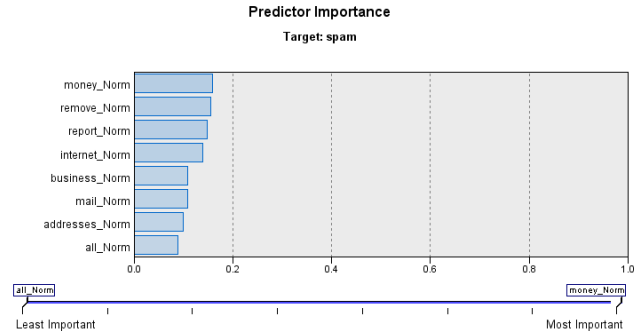
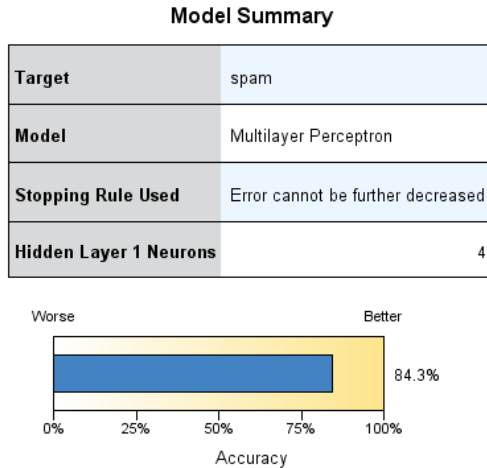
Row Percent



A vertical color scale legend for 'Row Percent' ranging from 0.00 (lightest blue) to 100.00 (darkest blue). The scale is marked at 0.00, 20.00, 40.00, 60.00, 80.00, and 100.00.

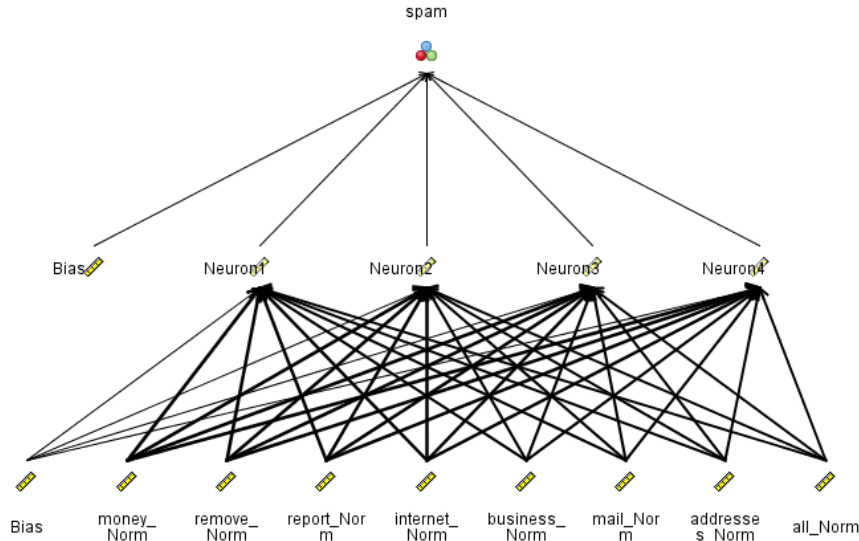
Slika 8 : Matrica konfuzije za neuronske mreže

# Neuronske mreže - SPSS



Slika 9 : Neuronska mreža

# Neuronske mreže - SPSS



Slika 10 : Neuronska mreža

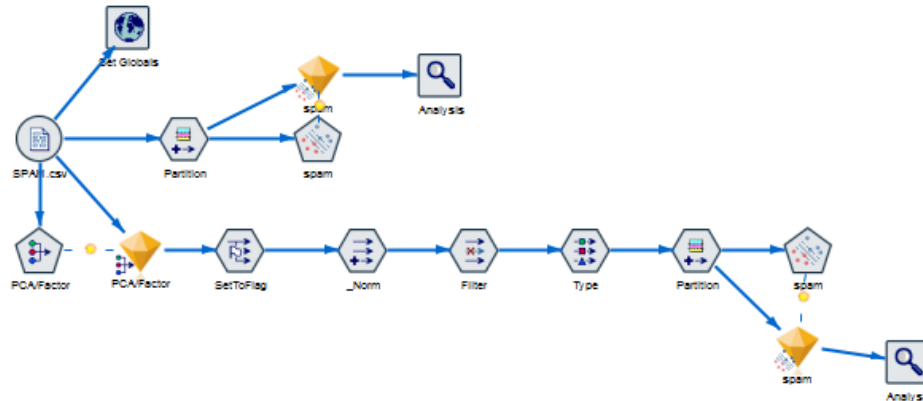
# Neuronske mreže - Python

```
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 Broj iteracija:  275
12 Broj slojeva:   4
```



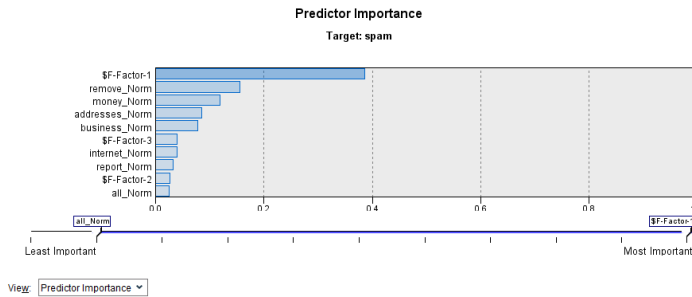
# Metod potpornih vektora - SVM

- 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)

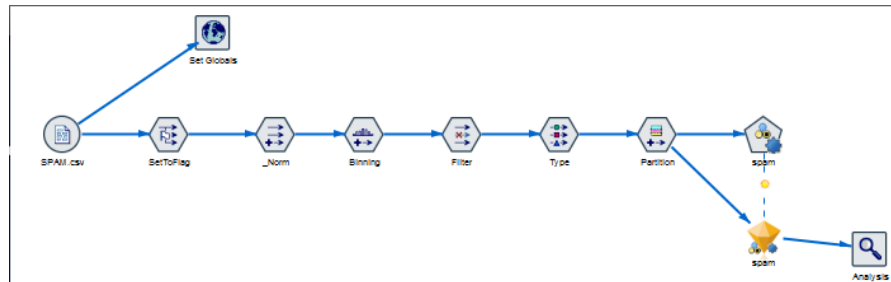
'Partition' = 1_Training	FALSE	TRUE
FALSE	1,816	130
TRUE	513	739
'Partition' = 2_Testing	FALSE	TRUE
FALSE	778	64
TRUE	213	348

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

# Gausova klasifikacija

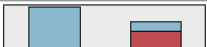



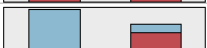





```
1 #Matrica konfuzije
2 [[808  29]
3  [306 238]]
4
5 Preciznost 0.7574221578566256
6 #Izvestaj klasifikacije
7               precision    recall  f1-score   support
8
9      tacno         0.73         0.97         0.83         837
10     netacno        0.89         0.44         0.59         544
```

# 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
✓		 C&R Tree 1	< 1	86.386	8
✓		 Neural Net 1	< 1	85.460	8
✓		 C5 1	< 1	85.317	7
✓		 SVM 1	< 1	80.969	8
✓		 KNN Algorithm 1	< 1	53.457	8

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

**Hvala na pažnji!**