

Nama : Cynthia As Bahri  
NIM : 221810227  
No. Absen : 07  
Kelas : 3SD1  
Dosen : Ibnu Santoso, SST, MT  
Mata Kuliah : Data Mining dan Knowledge Management  
Hari / Tanggal Ujian : Selasa / 27 Oktober 2020

“Saya menyatakan bahwa ujian ini saya kerjakan dengan jujur sesuai kemampuan sendiri dan tidak mengutip sebagian atau seluruh pekerjaan orang lain. Jika suatu saat ditemukan saya melanggar ketentuan ujian, saya siap menerima konsekuensi yang berlaku.”



(Cynthia As Bahri)

## KLASIFIKASI : METODE SVM

Tahapan dalam klasifikasi menggunakan metode SVM dengan R :

### 1. Load library

```
1 title: "ujian"
2 author: "Cynthia As Bahri"
3 date: "10/27/2020"
4 output:
5   word_document: default
6   html_document: default
7 ---
8
9
10 # load library
11 {r}
12 library(tidyverse)
13 library(e1071)
14 library(caret)
15 }
```

Library yang digunakan adalah tidyverse, e1071, dan caret. Library tidyverse digunakan untuk plotting dan mengolah variable. Sedangkan library e1071 digunakan untuk melakukan pemodelan SVM. Dan terakhir library caret digunakan untuk membuat confusion matrix dan melihat akurasi model.

### 2. Load data

```
17 # load data
18 {r}
19 parkinsons <- read.csv("E:/KULIAH/UJIAN ONLINE/UTS SEM 5/DATMIN/parkinsons.data")
20 head(parkinsons)
21 }
```

	name <fctr>	MDVP.Fo.Hz. <dbl>	MDVP.Fhi.Hz. <dbl>	MDVP.Flo.Hz. <dbl>	MDVP.Jitter... <dbl>	MDVP.Jitter.Abs. <dbl>
1	phon_R01_S01_1	119.992	157.302	74.997	0.00784	0.00007
2	phon_R01_S01_2	122.400	148.650	113.819	0.00968	0.00008
3	phon_R01_S01_3	116.682	131.111	111.555	0.01050	0.00009
4	phon_R01_S01_4	116.676	137.871	111.366	0.00997	0.00009
5	phon_R01_S01_5	116.014	141.781	110.655	0.01284	0.00011
6	phon_R01_S01_6	120.552	131.162	113.787	0.00968	0.00008

6 rows | 1-7 of 24 columns

Dataset : parkinsons ( <https://archive.ics.uci.edu/ml/datasets/Parkinsons> )

3. Hilangkan variable id (pada data parkinsons terdapat variable name yang akan dihilangkan)

```
23 # hilangkan var pertama (nama objek yang diobservasi)
24 {r}
25 parkinsons <- parkinsons[-1]
26
27
```

Variabel pertama dihilangkan karena variable tersebut hanya untuk id atau nama dari objek yang diobservasi.

4. Lihat struktur data secara keseluruhan

```
28 # lihat struktur data
29 {r}
30 str(parkinsons)
31
```

```
'data.frame': 195 obs. of 23 variables:
 $ MDVP.Fo.Hz. : num 120 122 117 117 116 ...
 $ MDVP.Fhi.Hz. : num 157 149 131 138 142 ...
 $ MDVP.Flo.Hz. : num 75 114 112 111 111 ...
 $ MDVP.Jitter... : num 0.00784 0.00968 0.0105 0.00997 0.01284 ...
 $ MDVP.Jitter.Abs.: num 0.00007 0.00008 0.00009 0.00009 0.00011 0.00008 0.00003 0.00003 0.00006 0.00006 ...
 $ MDVP.RAP : num 0.0037 0.00465 0.00544 0.00502 0.00655 0.00463 0.00155 0.00144 0.00293 0.00268 ...
 $ MDVP.PPQ : num 0.00554 0.00696 0.00781 0.00698 0.00908 0.0075 0.00202 0.00182 0.00332 0.00332 ...
 $ Jitter.DDP : num 0.0111 0.0139 0.0163 0.015 0.0197 ...
 $ MDVP.Shimmer : num 0.0437 0.0613 0.0523 0.0549 0.0643 ...
 $ MDVP.Shimmer.db.: num 0.426 0.626 0.482 0.517 0.584 0.456 0.14 0.134 0.191 0.255 ...
 $ Shimmer.APQ3 : num 0.0218 0.0313 0.0276 0.0292 0.0349 ...
 $ Shimmer.APQ5 : num 0.0313 0.0452 0.0386 0.0401 0.0483 ...
 $ MDVP.APQ : num 0.0297 0.0437 0.0359 0.0377 0.0447 ...
 $ Shimmer.DDA : num 0.0654 0.094 0.0827 0.0877 0.1047 ...
 $ NHR : num 0.0221 0.0193 0.0131 0.0135 0.0177 ...
 $ HNR : num 21 19.1 20.7 20.6 19.6 ...
 $ status : int 1 1 1 1 1 1 1 1 1 ...
 $ RPDE : num 0.415 0.458 0.43 0.435 0.417 ...
 $ DFA : num 0.815 0.82 0.825 0.819 0.823 ...
 $ spread1 : num -4.81 -4.08 -4.44 -4.12 -3.75 ...
 $ spread2 : num 0.266 0.336 0.311 0.334 0.235 ...
 $ D2 : num 2.3 2.49 2.34 2.41 2.33 ...
 $ PPE : num 0.285 0.369 0.333 0.369 0.41 ...
```

5. Ubah struktur data status yang menjadi penentu klasifikasi apakah dia sehat atau memiliki penyakit parkinsons (1 untuk memiliki penyakit parkinsons dan 0 untuk sehat)

```
33 # ganti struktur data status menjadi factor
34 {r}
35 parkinsons$status <- as.factor(parkinsons$status)
36 str(parkinsons)
37
```

```
'data.frame': 195 obs. of 23 variables:
 $ MDVP.Fo.Hz. : num 120 122 117 117 116 ...
 $ MDVP.Fhi.Hz. : num 157 149 131 138 142 ...
 $ MDVP.Flo.Hz. : num 75 114 112 111 111 ...
 $ MDVP.Jitter... : num 0.00784 0.00968 0.0105 0.00997 0.01284 ...
 $ MDVP.Jitter.Abs.: num 0.00007 0.00008 0.00009 0.00009 0.00011 0.00008 0.00003 0.00003 0.00006 0.00006 ...
 $ MDVP.RAP : num 0.0037 0.00465 0.00544 0.00502 0.00655 0.00463 0.00155 0.00144 0.00293 0.00268 ...
 $ MDVP.PPQ : num 0.00554 0.00696 0.00781 0.00698 0.00908 0.0075 0.00202 0.00182 0.00332 0.00332 ...
 $ Jitter.DDP : num 0.0111 0.0139 0.0163 0.015 0.0197 ...
 $ MDVP.Shimmer : num 0.0437 0.0613 0.0523 0.0549 0.0643 ...
 $ MDVP.Shimmer.db.: num 0.426 0.626 0.482 0.517 0.584 0.456 0.14 0.134 0.191 0.255 ...
 $ Shimmer.APQ3 : num 0.0218 0.0313 0.0276 0.0292 0.0349 ...
 $ Shimmer.APQ5 : num 0.0313 0.0452 0.0386 0.0401 0.0483 ...
 $ MDVP.APQ : num 0.0297 0.0437 0.0359 0.0377 0.0447 ...
 $ Shimmer.DDA : num 0.0654 0.094 0.0827 0.0877 0.1047 ...
 $ NHR : num 0.0221 0.0193 0.0131 0.0135 0.0177 ...
 $ HNR : num 21 19.1 20.7 20.6 19.6 ...
 $ status : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 ...
 $ RPDE : num 0.415 0.458 0.43 0.435 0.417 ...
 $ DFA : num 0.815 0.82 0.825 0.819 0.823 ...
 $ spread1 : num -4.81 -4.08 -4.44 -4.12 -3.75 ...
 $ spread2 : num 0.266 0.336 0.311 0.334 0.235 ...
 $ D2 : num 2.3 2.49 2.34 2.41 2.33 ...
 $ PPE : num 0.285 0.369 0.333 0.369 0.41 ...
```

6. Buat model SVM

```
38
39 # model svm
40 ```{r}
41 modelSVM <- svm(status~., data=parkinsons)
42 summary(modelSVM)
43 ```
```

Call:  
svm(formula = status ~ ., data = parkinsons)

Parameters:  
SVM-Type: C-classification  
SVM-kernel: radial  
cost: 1

Number of Support Vectors: 92  
( 53 39 )

Number of Classes: 2

Levels:  
0 1

Pada model SVM ini didapatkan support vector sebanyak 92 dan bertipe radial dengan cost 1.

7. Lakukan prediksi pada model tersebut dan tentukan apakah model tersebut sudah baik

```
45 # prediksi
46 ```{r}
47 pred <- predict(modelSVM, parkinsons)
48 confusionMatrix(table(Predicted = pred, Actual=parkinsons$status))
49 ```
```

Confusion Matrix and Statistics

	Actual	
Predicted	0	1
0	29	0
1	19	147

Accuracy : 0.9026  
95% CI : (0.852, 0.9403)  
No Information Rate : 0.7538  
P-Value [Acc > NIR] : 1.082e-07

Kappa : 0.6971

Mcnemar's Test P-Value : 3.636e-05

Sensitivity : 0.6042  
Specificity : 1.0000  
Pos Pred Value : 1.0000  
Neg Pred Value : 0.8855  
Prevalence : 0.2462  
Detection Rate : 0.1487  
Detection Prevalence : 0.1487  
Balanced Accuracy : 0.8021

'Positive' class : 0

Dari prediksi model ini, dibantu dengan confusion matrix didapatkan tingkat akurasi sebesar 90,26%. Dimana dapat diartikan model ini sudah cukup baik, dikarenakan memiliki akurasi yang lebih tinggi dari 90%.

Pada matrix yang didapatkan :

- True positive = 29
- False positive = 0
- True negative = 147
- False negative = 19

Dan didapatkan sensitivity (recall) sebesar 60,42% dimana merupakan persen benar pada prediksi dibagi persen benar actual.

$$\text{Recall} = 29 / (29+19) = 0,6042$$

Untuk specificity didapatkan sebesar 100% dimana merupakan persen salah pada prediksi dibagi persen salah actual.

$$\text{Specificity} = 147 / (147 + 0) = 1$$

Untuk precision didapatkan sebesar 100% dimana merupakan persen benar actual dibagi benar prediksi.

$$\text{Precision} = 29 / (29 + 0) = 1$$

Untuk F-1 Score didapatkan sebesar 75.33% dimana merupakan rata-rata harmonic dari precision dan recall

$$\text{F-1 Score} = 2 \times [ (\text{precision} \times \text{recall}) / (\text{precision} + \text{recall}) ] = 0,75327$$

Namun, bagaimana dengan mencari model terbaik?

Lakukan Langkah-langkah berikut (model di tune) :

#### 1. Set data training dan testing


```
57 - ### Mencari Model Terbaik
58 - {r}
59 set.seed(123)
60 modelbaik <- tune(svm, status=., data=parkinsons,
61                 ranges = list(epsilon = seq(0,1,0.1),
62                             cost = 2^(2:9)))
63 modelbaik
64
```



	epsilon	cost
1 row	0	16

## 2. Hitung summary

```
66 ##### Summary
67 ```{r}
68 summary(modelbaik)
69 ```
```



epsilon <dbl>	cost <dbl>	error <dbl>	dispersion <dbl>
0.0	4	0.07236842	0.05713551
0.1	4	0.07236842	0.05713551
0.2	4	0.07236842	0.05713551
0.3	4	0.07236842	0.05713551
0.4	4	0.07236842	0.05713551
0.5	4	0.07236842	0.05713551
0.6	4	0.07236842	0.05713551
0.7	4	0.07236842	0.05713551
0.8	4	0.07236842	0.05713551
0.9	4	0.07236842	0.05713551

1-10 of 88 rows

Previous 1 2 3 4 5 6 ... 9 Next

## 3. Lakukan permodelan

```
70
71 Didapat model terbaik adalah model dengan epsilon 0 dan cost 16.
72 ```{r}
73 bestmodel <- modelbaik$best.model
74 summary(bestmodel)
75 ```
```

Call:  
best.tune(method = svm, train.x = status ~ ., data = parkinsons, ranges = list(epsilon = seq(0, 1, 0.1), cost = 2^(2:9)))

Parameters:  
SVM-Type: C-classification  
SVM-Kernel: radial  
cost: 16

Number of Support Vectors: 69  
( 45 24 )

Number of Classes: 2

Levels:  
0 1

#### 4. Prediksi model

```
77 ##### Confusion Matrix Best Model
78 ```{r}
79 pred <- predict(bestmodel, parkinsons)
80 confusionMatrix(table(Predicted = pred, Actual=parkinsons$status))
81 ```
```

##### Confusion Matrix and Statistics

	Actual	
Predicted	0	1
0	48	1
1	0	146

Accuracy : 0.9949  
95% CI : (0.9718, 0.9999)  
No Information Rate : 0.7538  
P-Value [Acc > NIR] : <2e-16

Kappa : 0.9863

Mcnemar's Test P-Value : 1

Sensitivity : 1.0000  
Specificity : 0.9932  
Pos Pred Value : 0.9796  
Neg Pred Value : 1.0000  
Prevalence : 0.2462  
Detection Rate : 0.2462  
Detection Prevalence : 0.2513  
Balanced Accuracy : 0.9966

'Positive' class : 0

Model yang telah di tune memiliki akurasi yang lebih baik yaitu 99,49%.