Classification Breast Cancer Wisconsin (Diagnostic) Data Set

Rya Meyvriska
Ilmu Komputer IPB - G64164008

Data Set

Breast Cancer Wisconsin (Diagnostic) Data Set (wdbc.data)
Alamat info data set :
https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)
Alamat data set : https://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/wdbc.data

Tujuan

Mendiagnosis kelas kanker payudara, kelas dibagi kedalam dua yaitu:

- B = benign (jinak)
- M = malignant (ganas)

Teknik yang Digunakan

kNN (k Nearest Neighbor)

Informasi Data Set

- Breast Cancer Wisconsin (Diagnostic) Data Set (wdbc.data) , alamat info data set : https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)
- Jumlah data adalah 569 data
- Jumlah fitur adalah 32 fitur
- Missing values 0 data
- 2 kelas (B = benign (jinak), M = malignant (ganas))
- fitur terdiri dari id, factor, dan 30 fitur bertipe num, fitur id tidak digunakan

Distribusi Kelas

- Benign: 357 (62.74%)
- Malignant: 212 (37.26%)

Informasi fitur

Pada data terdiri dari 32 fitur, fitur pertama berupa id tidak digunakan pada penelitian ini karena tidak ada hubungannya dengan analisis klasifikasi. fitur-fitur yang digunakan adalah :

2. Diagnosis
3. radius mean
4. texture mean
5. perimeter mean
6. area mean
7. smoothness mean
8. compactness mean
9. concavity mean
10. concave points mean
11. symmetry mean
12. fractal dimension mean
13. radius se
14. texture se
15. perimeter se
16. area se smoothness se
17. compactness se
18. concavity se
19. concave points se
20. symmetry se
21. fractal dimension se
22. radius worst
23. texture worst
24. perimeter worst
25. area worst
26. smoothness worst
27. compactness worst
28. concavity worst
29. concave
30. points worst
31. symmetry worst
32. fractal dimension worst

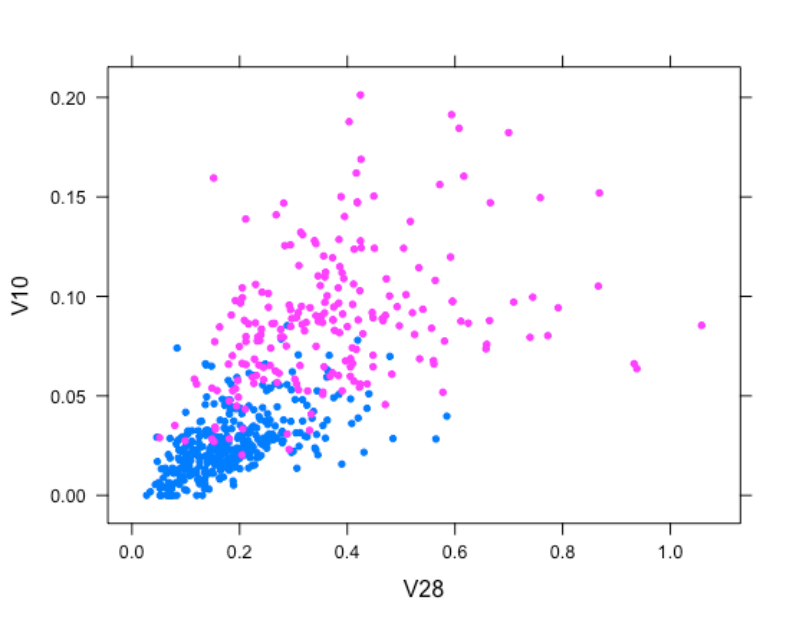
Alasan Memilih kNN

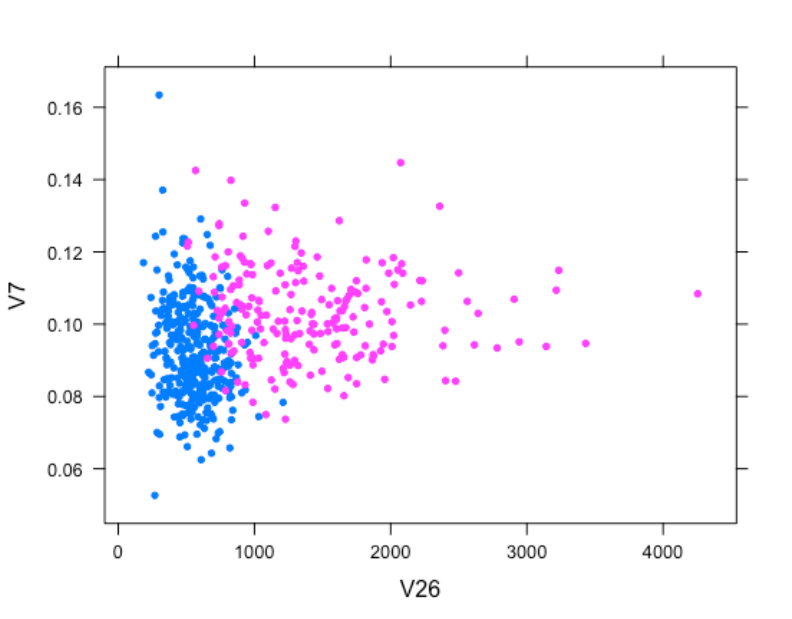
- Robust terhadap data yang noisy
- Efektif jika training data berjumlah banyak

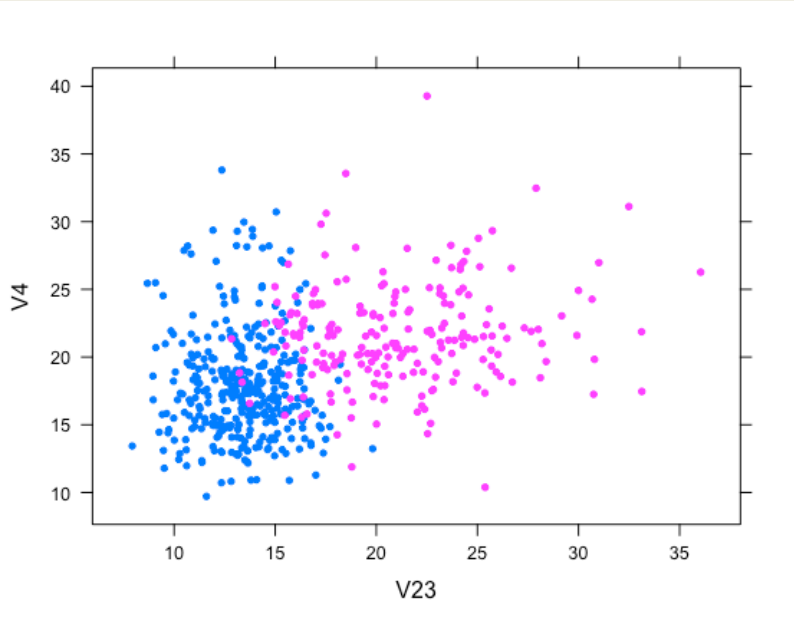
Alasan Menggunakan Semua fitur

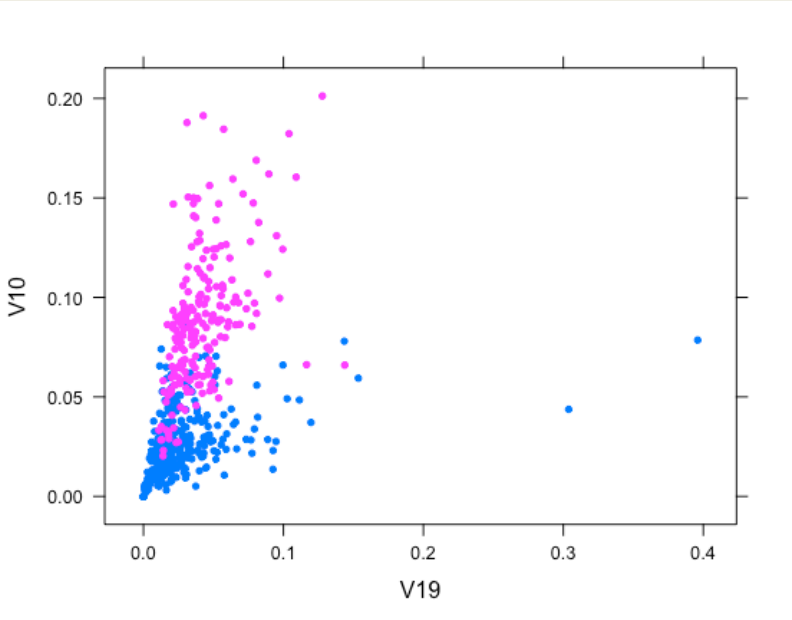
- Ingin mempertahankan informasi 100%
- Sumber daya yang ada mencukupi, baik pada segi peneliti, waktu, maupun komputasi

Beberapa Contoh Plot Data

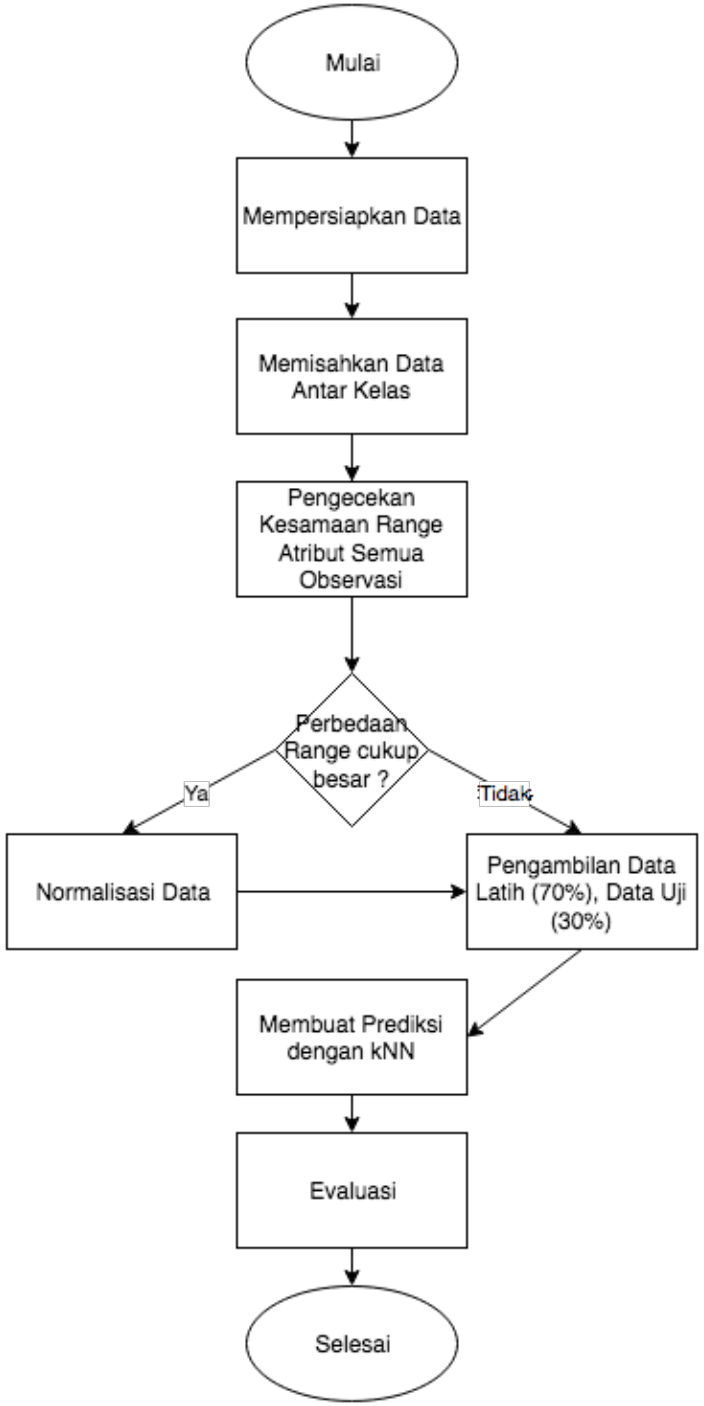








- Gambar plot data antara fitur 10 (concavity mean) dan fitur 28 (concavity worst)
- Gambar plot data antara fitur 7 (smoothness mean) dan fitur 26 (smoothness worst)
- Gambar plot data antara fitur 4 (texture mean) dan fitur 23 (texture worst)
- Gambar plot data antara fitur 10(concave points mean) dan fitur 19 (concave points se)



Tahapan Penelitian

Penelitian akan dilakukan dengan ±7 tahapan

Mempersiapkan Data

```
# 1. Mempersiapkan data
dt <- read.table("https://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/wdbc.data", sep = ',')
dt <- dt[,-1] #menghilangkan fitur v1, karena isinya hanya ID
```

- Pada tahap ini, diambil data set dari UCI yang bisa diakses dari https://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/wdbc.data
- Setelah itu, menghapus kolom ke 1 karena hanya berisi data id, dan pada penelitian ini tidak digunakan

Memisahkan Data Antar Kelas

```
# 2. Split data
dtClassB <- dt[which(dt$V2=='B'),]
dtClassM <- dt[which(dt$V2=='M'),]
```

- Pada tahap ini, data dipisah menjadi data kelas B dan kelas M yang nantinya akan digunakan untuk pemrosesan pengambilan data latih dan data uji

	V2	V3	V4	V5	V6	V7	V
42	M	10.950	21.35	71.90	371.1	0.12270	0.
43	M	19.070	24.81	128.30	1104.0	0.09081	0.
44	M	13.280	20.28	87.32	545.2	0.10410	0.
45	M	13.170	21.81	85.42	531.5	0.09714	0.
46	M	18.650	17.60	123.70	1076.0	0.10990	0.
47	B	8.196	16.84	51.71	201.9	0.08600	0.
48	M	13.170	18.66	85.98	534.6	0.11580	0.

Data

	V2	V3	V4	V5	V6	V7	V
20	B	13.540	14.36	87.46	566.3	0.09779	0.
21	B	13.080	15.71	85.63	520.0	0.10750	0.
22	B	9.504	12.44	60.34	273.9	0.10240	0.
38	B	13.030	18.42	82.61	523.8	0.08983	0.
47	B	8.196	16.84	51.71	201.9	0.08600	0.
49	B	12.050	14.63	78.04	449.3	0.10310	0.
50	B	13.490	22.30	86.91	561.0	0.08752	0.
51	B	11.760	21.60	74.72	427.9	0.08637	0

Data Kelas B

	V2	V3	V4	V5	V6	V7	V8
1	M	17.99	10.38	122.80	1001.0	0.11840	0.2
2	M	20.57	17.77	132.90	1326.0	0.08474	0.0
3	M	19.69	21.25	130.00	1203.0	0.10960	0.1
4	M	11.42	20.38	77.58	386.1	0.14250	0.2
5	M	20.29	14.34	135.10	1297.0	0.10030	0.1
6	M	12.45	15.70	82.57	477.1	0.12780	0.1
7	M	18.25	19.98	119.60	1040.0	0.09463	0.1
8	M	13.71	20.83	90.20	577.9	0.11890	0.1

Data Kelas M

```
> # 3. Cek Range
> summary(dt)
 V2           V3              V4              V5              V6              V7               V8
 B:357   Min.   : 6.981   Min.   : 9.71   Min.   : 43.79   Min.   : 143.5   Min.   :0.05263   Min.   :0.01938
 M:212   1st Qu.:11.700   1st Qu.:16.17   1st Qu.: 75.17   1st Qu.: 420.3   1st Qu.:0.08637   1st Qu.:0.06492
         Median :13.370   Median :18.84   Median : 86.24   Median : 551.1   Median :0.09587   Median :0.09263
         Mean   :14.127   Mean   :19.29   Mean   : 91.97   Mean   : 654.9   Mean   :0.09636   Mean   :0.10434
         3rd Qu.:15.780   3rd Qu.:21.80   3rd Qu.:104.10   3rd Qu.: 782.7   3rd Qu.:0.10530   3rd Qu.:0.13040
         Max.   :28.110   Max.   :39.28   Max.   :188.50   Max.   :2501.0   Max.   :0.16340   Max.   :0.34540
       V9               V10               V11              V12               V13              V14              V15
 Min.   :0.00000   Min.   :0.00000   Min.   :0.1060   Min.   :0.04996   Min.   :0.1115   Min.   :0.3602   Min.   : 0.757
 1st Qu.:0.02956   1st Qu.:0.02031   1st Qu.:0.1619   1st Qu.:0.05770   1st Qu.:0.2324   1st Qu.:0.8339   1st Qu.: 1.606
 Median :0.06154   Median :0.03350   Median :0.1792   Median :0.06154   Median :0.3242   Median :1.1080   Median : 2.287
 Mean   :0.08880   Mean   :0.04892   Mean   :0.1812   Mean   :0.06280   Mean   :0.4052   Mean   :1.2169   Mean   : 2.866
 3rd Qu.:0.13070   3rd Qu.:0.07400   3rd Qu.:0.1957   3rd Qu.:0.06612   3rd Qu.:0.4789   3rd Qu.:1.4740   3rd Qu.: 3.357
 Max.   :0.42680   Max.   :0.20120   Max.   :0.3040   Max.   :0.09744   Max.   :2.8730   Max.   :4.8850   Max.   :21.980
      V16               V17                V18               V19               V20                V21
 Min.   :  6.802   Min.   :0.001713   Min.   :0.002252   Min.   :0.00000   Min.   :0.000000   Min.   :0.007882
 1st Qu.: 17.850   1st Qu.:0.005169   1st Qu.:0.013080   1st Qu.:0.01509   1st Qu.:0.007638   1st Qu.:0.015160
 Median : 24.530   Median :0.006380   Median :0.020450   Median :0.02589   Median :0.010930   Median :0.018730
 Mean   : 40.337   Mean   :0.007041   Mean   :0.025478   Mean   :0.03189   Mean   :0.011796   Mean   :0.020542
 3rd Qu.: 45.190   3rd Qu.:0.008146   3rd Qu.:0.032450   3rd Qu.:0.04205   3rd Qu.:0.014710   3rd Qu.:0.023480
 Max.   :542.200   Max.   :0.031130   Max.   :0.135400   Max.   :0.39600   Max.   :0.052790   Max.   :0.078950
      V22                V23              V24              V25              V26              V27               V28
 Min.   :0.0008948   Min.   : 7.93   Min.   :12.02   Min.   : 50.41   Min.   : 185.2   Min.   :0.07117   Min.   :0.02729
 1st Qu.:0.0022480   1st Qu.:13.01   1st Qu.:21.08   1st Qu.: 84.11   1st Qu.: 515.3   1st Qu.:0.11660   1st Qu.:0.14720
 Median :0.0031870   Median :14.97   Median :25.41   Median : 97.66   Median : 686.5   Median :0.13130   Median :0.21190
 Mean   :0.0037949   Mean   :16.27   Mean   :25.68   Mean   :107.26   Mean   : 880.6   Mean   :0.13237   Mean   :0.25427
 3rd Qu.:0.0045580   3rd Qu.:18.79   3rd Qu.:29.72   3rd Qu.:125.40   3rd Qu.:1084.0   3rd Qu.:0.14600   3rd Qu.:0.33910
 Max.   :0.0298400   Max.   :36.04   Max.   :49.54   Max.   :251.20   Max.   :4254.0   Max.   :0.22260   Max.   :1.05800
      V29               V30               V31              V32
 Min.   :0.0000   Min.   :0.00000   Min.   :0.1565   Min.   :0.05504
 1st Qu.:0.1145   1st Qu.:0.06493   1st Qu.:0.2504   1st Qu.:0.07146
 Median :0.2267   Median :0.09993   Median :0.2822   Median :0.08004
 Mean   :0.2722   Mean   :0.11461   Mean   :0.2901   Mean   :0.08395
 3rd Qu.:0.3829   3rd Qu.:0.16140   3rd Qu.:0.3179   3rd Qu.:0.09208
 Max.   :1.2520   Max.   :0.29100   Max.   :0.6638   Max.   :0.20750
```

Cek range antar fitur

Untuk mengecek perbedaan range antar fitur

Normalisasi Data

Normalisasi data dilakukan pada fitur agar memiliki range yang sama, yaitu 0.0 hingga 1.0. Range atau rentan yang cukup besar dapat diatasi dengan normalisasi. Perbedaan range dapat membuat fitur yang memiliki range kecil seolah-olah tidak penting, padahal semua fitur digunakan dan dirasa penting.

```
> # 4. Normalisasi data
> normalize <- function(x) {
+   return ((x-min(x)) / (max(x) - min(x)))
+ }
> normalizedDtB <- as.data.frame(lapply(dtClassB[,2:31], normalize))
> normalizedDtM <- as.data.frame(lapply(dtClassM[,2:31], normalize))
```

```
> summary(normalizedDtB)
       V3                V4                V5                V6                V7                V8                V9
 Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.00000
 1st Qu.:0.3771   1st Qu.:0.2257   1st Qu.:0.3824   1st Qu.:0.2766   1st Qu.:0.2747   1st Qu.:0.1772   1st Qu.:0.04944
 Median :0.4802   Median :0.3187   Median :0.4857   Median :0.3711   Median :0.3442   Median :0.2734   Median :0.09029
 Mean   :0.4753   Mean   :0.3404   Mean   :0.4842   Mean   :0.3763   Mean   :0.3597   Mean   :0.2968   Mean   :0.11212
 3rd Qu.:0.5878   3rd Qu.:0.4170   3rd Qu.:0.5975   3rd Qu.:0.4803   3rd Qu.:0.4340   3rd Qu.:0.3822   3rd Qu.:0.14603
 Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.00000
```

Data Kelas B

```
> summary(normalizedDtM)
       V3                V4                V5                V6                V7                V8                V9
 Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
 1st Qu.:0.2404   1st Qu.:0.3096   1st Qu.:0.2302   1st Qu.:0.1607   1st Qu.:0.2860   1st Qu.:0.2123   1st Qu.:0.2124
 Median :0.3715   Median :0.3834   Median :0.3628   Median :0.2666   Median :0.4013   Median :0.2883   Median :0.3162
 Mean   :0.3795   Mean   :0.3884   Mean   :0.3728   Mean   :0.2883   Mean   :0.4112   Mean   :0.3312   Mean   :0.3396
 3rd Qu.:0.5035   3rd Qu.:0.4631   3rd Qu.:0.4976   3rd Qu.:0.3936   3rd Qu.:0.5242   3rd Qu.:0.4221   3rd Qu.:0.4445
 Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
```

Data Kelas M

Pengambilan Data Uji dan Data Latih

```
> # 5. Ambil training data (70%) dan testing data (30%)
> # Ambil 70% data dari kelas B
> nClassB <- nrow(normalizedDtB)
> boundClassB <- ceiling(nClassB * 0.7)
> trainingDtClassB <- normalizedDtB[1:boundClassB, ]
> testingDtClassB <- normalizedDtB[(boundClassB + 1):nClassB, ]
> trainingLabelsClassB <- dtClassB[1:boundClassB, 1]
> testingLabelsClassB <- dtClassB[(boundClassB + 1):nClassB, 1]
>
> # Ambil 70% data dari kelas M
> nClassM <- nrow(normalizedDtM)
> boundClassM <- ceiling(nClassM * 0.7)
> trainingDtClassM <- normalizedDtM[1:boundClassM, ]
> testingDtClassM <- normalizedDtM[(boundClassM + 1):nClassM, ]
> trainingLabelsClassM <- dtClassM[1:boundClassM, 1]
> testingLabelsClassM <- dtClassM[(boundClassM + 1):nClassM, 1]
>
> # Satukan dalam satu data
> trainingDt <- rbind(trainingDtClassB[,], trainingDtClassM[,])
> testingDt <- rbind(testingDtClassB[,], testingDtClassM[,])
> trainingLabels <- c(as.character(trainingLabelsClassB[]), as.character(trainingLabelsClassM[]))
> trainingLabels <- as.factor(trainingLabels)
> testingLabels <- c(as.character(testingLabelsClassB[]), as.character(testingLabelsClassM[]))
> testingLabels <- as.factor(testingLabels)
```

Di beberapa literatur, proporsi yang cukup sering digunakan adalah 70%:30% atau 80%:20%. Pada penelitian ini menginginkan data uji yang lebih besar, sehingga perbandingan yang digunakan adalah 70%:30%.

Membuat Prediksi dengan kNN

```
> # 6.Membuat prediksi dengan kNN
> library(class)
> startK <- ceiling(sqrt(nrow(trainingDt)))-1
> predictionLabels <- knn(train = trainingDt, test = testingDt, cl = trainingLabels, k = startK)
```

Menurut Duda *et all.* (2000), optimum k ada dikisaran √n. n atau jumlah data latih adalah 399 data, akar dari 399 mendekati 20. k yang digunakan adalah k±5(ganjil), yaitu 15, 17, 19, 21, 23, 25. Gambar diatas menampilkan contoh code untuk k-1 atau 19.

Evaluasi

```
> # 7. Evaluasi
> # Tabel Hasil
> confusionMatrix <- table(predictionLabels,testingLabels)
> accuracy <- (confusionMatrix[1,1] + confusionMatrix [2,2]) / (confusionMatrix[1,1] + confusionMatrix[1,2] + confusionMatrix[2,1] +confusionMatrix[2,2])
> recall <- (confusionMatrix[1,1]) / (confusionMatrix[1,1] + confusionMatrix[1,2])
> specificity <- (confusionMatrix [2,2]) / (confusionMatrix[2,1] +confusionMatrix[2,2])
> precision <- (confusionMatrix[1,1]) / (confusionMatrix[1,1] + confusionMatrix[2,1])
> # List Data Evaluasi
> evaluasiList <- list(confusionMatrix = confusionMatrix, accuracy = accuracy , recall = recall, specificity = specificity, precision = precision)
> evaluasiList
```

Perbedaan hasil antar k

- k = 15
- k = 17
- k = 19
- k = 21
- k = 23
- k = 25

```
> evaluasiList
$confusionMatrix
                testingLabels
predictionLabels   B   M
               B 103   7
               M   4  56

$accuracy
[1] 0.9352941

$recall
[1] 0.9363636

$specificity
[1] 0.9333333

$precision
[1] 0.9626168
```

```
> evaluasiList
$confusionMatrix
                testingLabels
predictionLabels   B   M
               B 105   6
               M   2  57

$accuracy
[1] 0.9529412

$recall
[1] 0.9459459

$specificity
[1] 0.9661017

$precision
[1] 0.9813084
```

```
> evaluasiList
$confusionMatrix
                testingLabels
predictionLabels   B   M
               B 105   6
               M   2  57

$accuracy
[1] 0.9529412

$recall
[1] 0.9459459

$specificity
[1] 0.9661017

$precision
[1] 0.9813084
```

```
> evaluasiList
$confusionMatrix
                testingLabels
predictionLabels   B   M
               B 105   7
               M   2  56

$accuracy
[1] 0.9470588

$recall
[1] 0.9375

$specificity
[1] 0.9655172

$precision
[1] 0.9813084
```

```
> evaluasiList
$confusionMatrix
                testingLabels
predictionLabels   B   M
               B 105   7
               M   2  56

$accuracy
[1] 0.9470588

$recall
[1] 0.9375

$specificity
[1] 0.9655172

$precision
[1] 0.9813084
```

```
> evaluasiList
$confusionMatrix
                testingLabels
predictionLabels   B   M
               B 105   7
               M   2  56

$accuracy
[1] 0.9470588

$recall
[1] 0.9375

$specificity
[1] 0.9655172

$precision
[1] 0.9813084
```

Accuracy : jumlah kanker jinak yang dipisahkan dengan benar / semua kanker
Recall : jumlah kanker jinak yang dipisahkan dengan benar/ semua kanker jinak sesungguhnya
Specificity : jumlah kanker ganas yang dipisahkan dengan benar/ jumlah yang diduga kanker ganas
Precision : jumlah kanker jinak yang dipisahkan dengan benar/ jumlah yang diduga kanker jinak

Perbandingan dengan Over-Under-Both Sampling

```
> prop.table(table(dt$V2))

        B         M
0.6274165 0.3725835
```

Menurut Brownlee (2015) , data dinyatakan imbalance jika perbandingannya mencapai 4:1 atau lebih. Data yang saya pakai perbandingannya hanya antara 2:1 hingga 3:2. Untuk lebih yakin, dicoba oversampling, undersampling, dan bothsampling untuk perbandingan. Ternyata hasil perhitungan matriks-nya tidak lebih baik dari sampling biasa. Berikut hasil dari uji oversampling, undersampling, dan bothsampling:

```
> # Oversampling pada trainingDt
> library(ROSE) #Random OverUnder Sampling Example
> dtOver <- cbind(trainingLabels,trainingDt)
> trainingDtOver <- ovun.sample(trainingLabels~., data = dtOver,
method = "over", N = 500)$data
> trainingDtOverLabel <- trainingDtOver$trainingLabels
> trainingDtOver <- trainingDtOver[,-1]
```

```
> evaluasiList
$confusionMatrix
                 testingLabels
predictionLabels  B  M
               B 95  3
               M 12 60

$accuracy
[1] 0.9117647

$recall
[1] 0.9693878

$specificity
[1] 0.8333333

$precision
[1] 0.8878505
```

```
> # Undersampling pada trainingDt
> library(ROSE) #Random OverUnder Sampling Example
> dtUnder <- cbind(trainingLabels,trainingDt)
> trainingDtOver <- ovun.sample(trainingLabels~., data = dtUnder,
method = "under", N = 298)$data
> trainingDtOverLabel <- trainingDtOver$trainingLabels
> trainingDtOver <- trainingDtOver[,-1]
```

```
> evaluasiList
$confusionMatrix
                 testingLabels
predictionLabels   B   M
               B 101   4
               M   6  59

$accuracy
[1] 0.9411765

$recall
[1] 0.9619048

$specificity
[1] 0.9076923

$precision
[1] 0.9439252
```

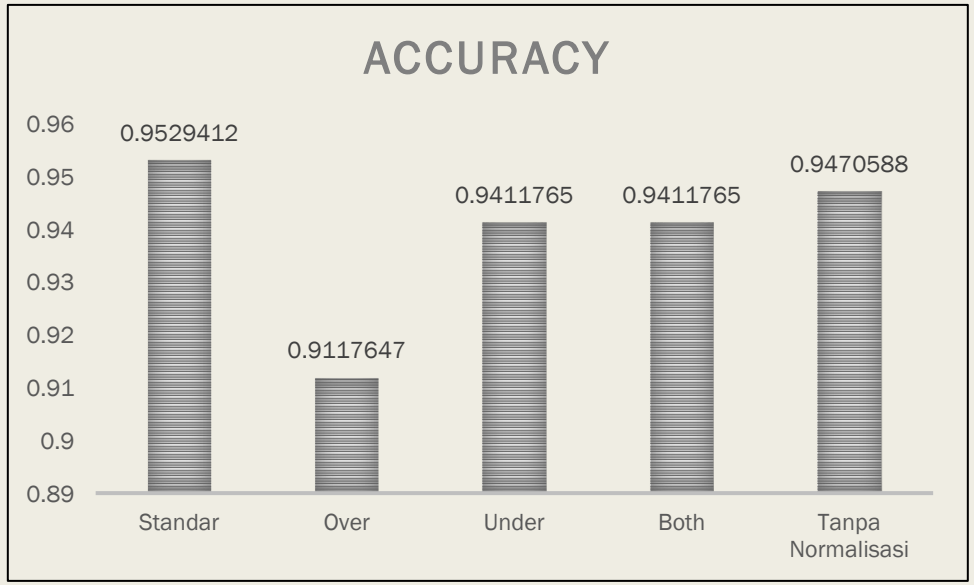
```
> # Bothsampling pada trainingDt
> library(ROSE) #Random OverUnder Sampling Example
> dtBoth <- cbind(trainingLabels,trainingDt)
> trainingDtOver <- ovun.sample(trainingLabels~., data = dtBoth,
method = "both", N = 399)$data
> trainingDtOverLabel <- trainingDtOver$trainingLabels
> trainingDtOver <- trainingDtOver[,-1]
```

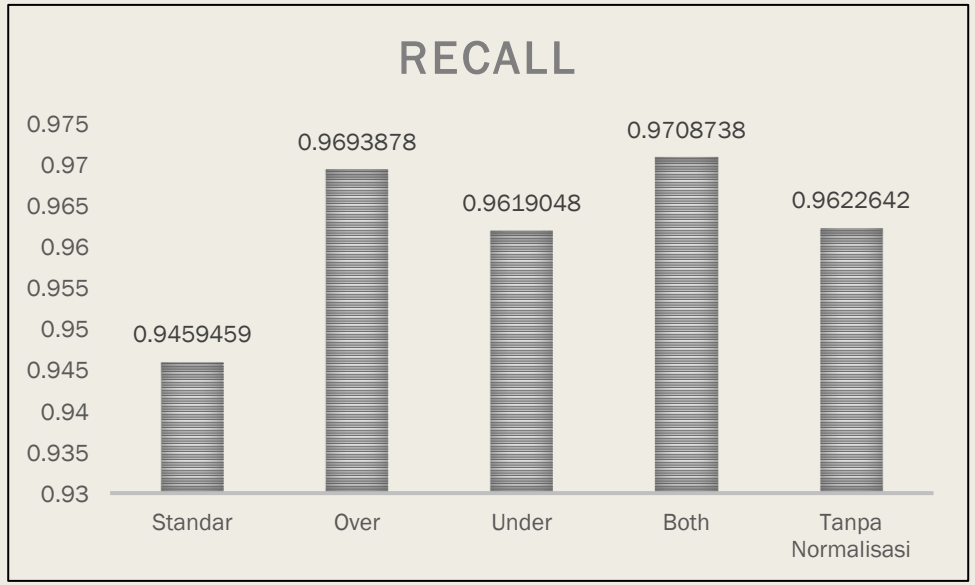
```
> evaluasiList
$confusionMatrix
                 testingLabels
predictionLabels   B   M
               B 100   3
               M   7  60

$accuracy
[1] 0.9411765

$recall
[1] 0.9708738

$specificity
[1] 0.8955224

$precision
[1] 0.9345794
```

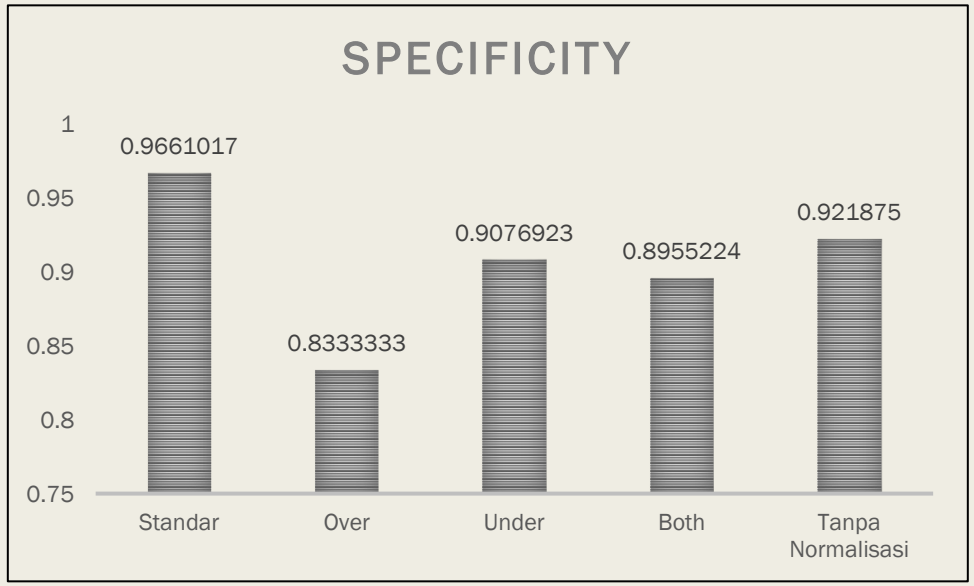
Perbandingan dengan data tanpa Normalisasi

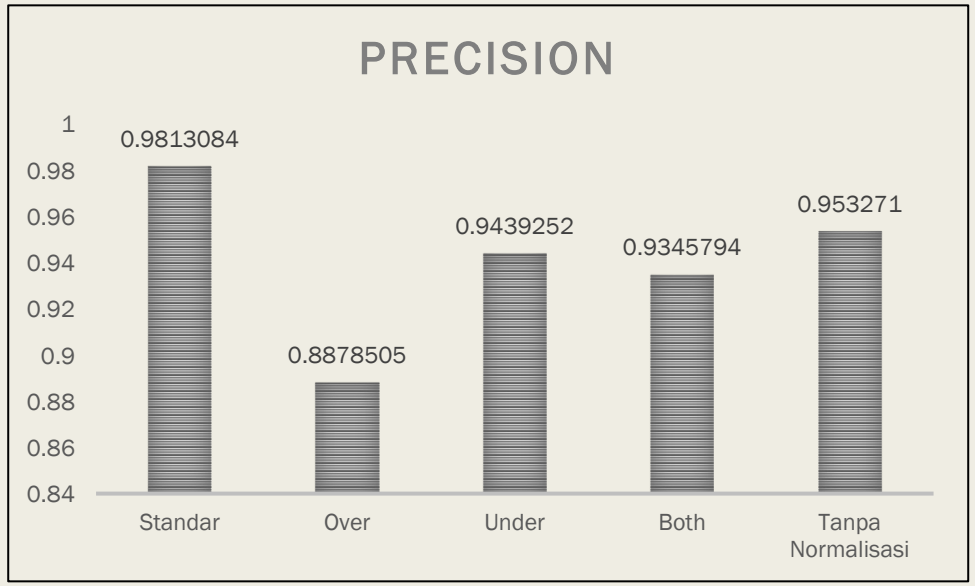
```
> evaluasiList
$confusionMatrix
                 testingLabels
predictionLabels   B   M
               B 102   4
               M   5  59

$accuracy
[1] 0.9470588

$recall
[1] 0.9622642

$specificity
[1] 0.921875

$precision
[1] 0.953271
```

Pada data yang tidak dinormalisasi memiliki recall yang lebih besar dengan perbedaan 0.9%. Namun, nilai accuracy, specificity, dan precision nya lebih kecil dibandingkan data yang dinormalisasi. Perbedaan accuracy adalah 0.59%, perbedaan specificity adalah 4.42%, dan perbedaan precision adalah 2.8%. Apabila di rata-rata, perbedaan data tanpa normalisasi dibandingkan data yang dinormalisasi adalah **2.19%**, dan data dengan normalisasi masih memiliki nilai yang rata-rata lebih tinggi.


ACCURACY
0.96
0.95
0.94
0.93
0.92
0.91
0.9
0.89
0.9529412
0.9117647
0.9411765
0.9411765
0.9470588
Standar
Over
Under
Both
Tanpa Normalisasi


RECALL
0.975
0.97
0.965
0.96
0.955
0.95
0.945
0.94
0.935
0.93
0.9459459
0.9693878
0.9619048
0.9708738
0.9622642
Standar
Over
Under
Both
Tanpa Normalisasi


SPECIFICITY
1
0.95
0.9
0.85
0.8
0.75
0.9661017
0.8333333
0.9076923
0.8955224
0.921875
Standar
Over
Under
Both
Tanpa Normalisasi


PRECISION
1
0.98
0.96
0.94
0.92
0.9
0.88
0.86
0.84
0.9813084
0.8878505
0.9439252
0.9345794
0.953271
Standar
Over
Under
Both
Tanpa Normalisasi

Kesimpulan

- Classification Breast Cancer Wisconsin (Diagnostic) Data Set menggunakan kNN setelah diuji k dengan evaluasi terbaik adalah 17 dan 19, karena 17 memiliki angka lebih kecil sehingga membutuhkan waktu komputasi/perhitungan lebih singkat maka k yang digunakan adalah 17. Pengujian dilakukan dengan perbandingan 70% data latih dan 30% data uji. Pengujian menghasilkan accuracy 95.28%, recall 94.59% , specificity 96.61% , dan precision 98.13%.
- Apabila data tidak dinormalisasi, nilai confusionMatriks lebih kecil kecuali pada recall saja. Perbedaan nilai confusionMatriks antara data normalisasi dan tidak dinormalisasi adalah 2.19%.
- Apabila diperlukan waktu komputasi yang lebih cepat, maka untuk data ini dapat dilakukan tanpa normalisasi, mengingat perbedaan dengan normalisasi hanya sekitar 2.19%.
- Setelah dibandingkan dengan over sampling, under sampling, dan both sampling ternyata pada kasus ini, sampling biasa masih memiliki nilai confusion matriks yang lebih baik.

Daftar Pustaka

Browniee Jason. 2015. *8 Tactics to Combat Imbalanced Classes in Your Machine Learning Dataset* [Internet]. [diakses 2017 Sep 3]. https://machinelearningmastery.com/tactics-to-combat-imbalanced-classes-in-your-machine-learning-dataset/

DataQ. 2013. *Perbedaan: precision, recall & accuracy* [Internet]. [diakses 2017 Sep 3]. https://dataq.wordpress.com/2013/06/16/perbedaan-precision-recall-accuracy/

Duda Richard O, Hart Peter E, Stork David G. 2000. *Pattern Classification Second Edition*.

Dzikrulloh Nihru Nafi, Indriati, Setiawan Budi Darma. 2017. Penerapan Metode K-Nearest Neighbor(kNN) dan Metode Weighted Product (WP) Dalam Penerimaan Calon Guru dan Karyawan Tata Usaha Baru Berwawasan Teknologi (Studi Kasus: Sekolah Menegah Kejuruan Muhammadiyah 2 Kediri). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*. 1(5):378-385.

Greenfield Yuval. 2012. *Precision, recall, sensitivity and specificity* [Internet]. [diakses 2017 Sep 3].https://uberpython.wordpress.com/2012/01/01/precision-recall-sensitivity-and-specificity/

Rahayu Dewi Sri. 2014. Klasifikasi *Naïve Bayes* Pada Data Tidak Seimbang Untuk Kasus Prediksi Resiko Kredit Debitur Kartu Kredit [skripsi]. Bogor(BGR): Institut Pertanian Bogor.

Sugianto Castaka Agus. 2015. Analisis Komparasi Algoritma Klasifikasi Untuk Menangani Data Tidak Seimbang Pada Data Kebakaran Hutan. *Techno.COM*. 14(4):336-342.