

PERTEMUAN 8

TEKNIK KLASIFIKASI DALAM R

TUJUAN PRAKTIKUM

Mahasiswa akan dapat menggunakan teknik-teknik dasar klasifikasi menggunakan *tools* R

TEORI PENUNJANG

Pada pertemuan ini, teknik klasifikasi yang akan digunakan adalah *decision tree* dan *support vector machine* (SVM). Berikut adalah sedikit penjelasan mengenai kedua teknik tersebut

Decision Tree

Decision tree adalah *flow chart* seperti struktur *tree*, dimana setiap *internal node* (node bukan *leaf*) merupakan uji atribut, sedangkan masing-masing cabang menunjukkan keluaran dari hasil uji, dan masing-masing *leaf* (atau terminal node) merupakan label kelas. Node paling atas pada *tree* disebut *node root*.

Bagaimana *decision tree* digunakan untuk klasifikasi? Diberikan sekumpulan data, X , yang label kelasnya tidak diketahui, nilai atribut pada data diuji dengan *decision tree* yang ada. *Path* di-tracing dari *root* sampai *leaf*, yang merupakan kelas prediksi untuk data tersebut. *Decision tree* dapat secara mudah diubah ke aturan klasifikasi.

Mengapa *decision tree* sangat populer? Untuk membangun model *decision tree* tidak memerlukan domain pengetahuan atau pengaturan parameter khusus. Selain itu, *decision tree* dapat digunakan untuk menangani data dimensi banyak. Representasi pengetahuan yang diperoleh pada bentuk *tree* adalah intuitif dan mudah dipahami oleh manusia. Proses pembelajaran dan tahap klasifikasi dari *decision tree* cukup mudah dan cepat. Secara umum, *classifier* pada *decision tree* mempunyai akurasi yang tinggi.

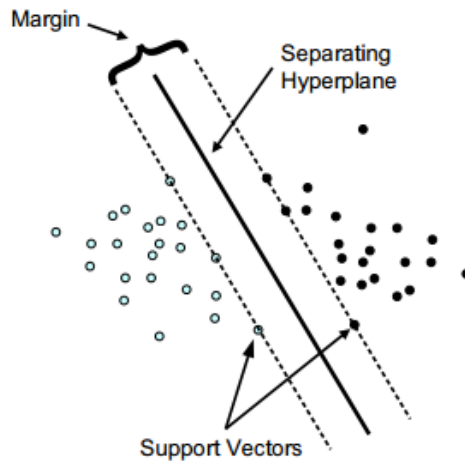
Untuk membentuk sebuah *decision tree* pemilihan atribut yang akan menjadi *internal node* sangat penting. Ada beberapa informasi yang dapat digunakan untuk melakukan pemilihan node. Beberapa contohnya adalah *information gain*, *gain ratio*, dan *gini index*.

Untuk beberapa kasus data yang sangat besar dengan atribut yang sangat besar pula, *tree* yang dihasilkan akan mungkin sangat rumit. Oleh karena itu, ada teknik untuk membuat bentuk *tree* menjadi lebih sederhana. Teknik ini disebut *pruning*. *Pruning* adalah teknik untuk memangkas *tree* sehingga lebih sederhana.

Support Vector Machine (SVM)

SVM pertama kali dibuat oleh Cortes dan Vapnik pada tahun 1995 untuk mengklasifikasikan data biner. Pendekatan yang digunakan mereka dapat dijelaskan secara singkat sebagai berikut

- Class separation: algoritma ini mencari *hyperplane* pemisah kelas yang optimal dengan memaksimalkan margin antar titik terdekat dengan kelasnya.



Dari gambar, dapat dilihat bahwa titik yang berada di pembatas disebut *support vector*, titik yang berada di tengah margin adalah *hyperplane* pemisah optimal.

- Overlapping classes: data yang berada pada “sisi yang salah” dari margin diskriminan, diboboti untuk mengurangi pengaruhnya (“soft margin”)
- Nonlinearity: ketika separator linear tidak dapat ditemukan, data akan diproyeksikan ke dimensi yang lebih tinggi sehingga data dapat dipisahkan secara linear.
- Problem solution: keseluruhan proses dapat diformulasikan kedalam optimasi masalah kuadratik.

LAPORAN PENDAHULUAN

1. Apa yang anda ketahui tentang klasifikasi?
2. Teknik apa saja yang dapat dipakai untuk melakukan klasifikasi?
3. Mengapa teknik *decision tree* banyak digunakan?
4. Apa kelebihan teknik SVM?

MATERI PRAKTIKUM

1. Klasifikasi dengan *Decision tree* (package: party)

Akan dilakukan klasifikasi pada data iris dengan fungsi *ctree()* dan *predict()* pada package party. *ctree()* digunakan untuk melakukan training pembentukan kelas, sedangkan *predict()* digunakan untuk melakukan prediksi kelas pada data yang belum ada label kelasnya (atau data uji).

Untuk melakukan training pada data set iris dapat dilakukan dengan tahapan sebagai berikut :

- a. Melakukan pembagian data ke dalam data training dan data testing

```
> str(iris)
> set.seed(1234)
> ind <- sample(2, nrow(iris), replace=TRUE, prob=c(0.7, 0.3))
> trainData<-iris[ind==1,]
> testData <- iris[ind==2,]
```

- b. Melakukan pemanggilan package party, dan melakukan training data

```
> print(iris_ctree)
> plot(iris_ctree)
> plot(iris_ctree, type="simple")
```

- c. Membandingkan hasil prediksi dan kelas sebenarnya

```
table(predict(iris_ctree), trainData$Species)
```

- d. Mencetak rule dan *tree* yang dihasilkan

```
> library(party)
> myFormula <- Species ~ Sepal.Length + Sepal.Width +
Petal.Length + Petal.Width
> iris_ctree <- ctree(myFormula, data=trainData)
```

- e. Melakukan prediksi kelas pada data testing

```
> testPred <- predict(iris_ctree, newdata = testData)
> table(testPred, testData$Species)
```

2. Klasifikasi dengan *Decision tree* (package: rpart)

Untuk melakukan training pada data set iris dapat dilakukan dengan tahapan sebagai berikut :

- a. Melakukan pembagian data ke dalam data training dan data testing

```
> str(iris)
> set.seed(1234)
> ind <- sample(2, nrow(iris), replace=TRUE, prob=c(0.7, 0.3))
> trainData<-iris[ind==1,]
> testData <- iris[ind==2,]
```

- b. Melakukan pemanggilan package party, dan melakukan training data

```
> library(rpart)
> myFormula <- Species ~ Sepal.Length + Sepal.Width +
Petal.Length + Petal.Width
> iris_rpart <- rpart(myFormula, data = trainData, control =
rpart.control(minsplit = 10))
```

c. Mencetak rule dan *tree* yang dihasilkan

```
> print(iris_rpart)
> plot(iris_rpart)
> text(iris_rpart, use.n=TRUE)
```

```
> iris_pred <- predict(iris_rpart, newdata = testData)
```

d. Melakukan prediksi kelas pada data testing

3. Klasifikasi dengan SVM (package: svm)

Untuk melakukan training SVM pada data set iris dapat dilakukan dengan tahapan sebagai berikut :

a. Melakukan pembagian data ke dalam data training dan data testing

```
> ind <- sample(2, nrow(iris), replace=TRUE, prob=c(0.7, 0.3))
> trainData<- iris[ind==1,]
> testData <- iris[ind==2,]
```

b. Melakukan pemanggilan package dan melakukan pemodelan

```
> library(e1071)
> library(rpart)
> svm.model <- svm(Type ~ ., data = trainData, cost = 100,
gamma = 1)
> svm.pred <- predict(svm.model, testData[,10])
```

```
table(pred = svm.pred, true = testset[,10])
```

c. Melakukan pemanggilan package dan melakukan pemodelan

DAFTAR PUSTAKA

1. Witten I.H & Frank E. 2011. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan-Kaufman. <http://www.cs.waikato.ac.nz/~ml/weka/book.html>. Accessed on 13 April 2014.
2. Bouckaert, R. R.. et al. 2013. *WEKA Manual for Version 3-6-9*. Edition of January 21, 2013. <http://jaist.dl.sourceforge.net/project/weka/documentation/3.6.x/WekaManual-3-6-9.pdf> . Accessed on 13 April 2014.