



Departemen Statistika

Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Pertanian Bogor

PEMODELAN KLASIFIKASI

PERTEMUAN #4

**PENILAIAN KEBAIKAN MODEL, VALIDASI,
VALIDASI SILANG, DAN PENGANTAR KE
RELATIVE IMPORTANCE ANALYSIS**

Bagus Sartono

bagusco@ipb.ac.id

bagusco@gmail.com

Latihan: Data Kualitas Wine

Data ini berasal dari pengamatan terhadap sejumlah tipe wine putih (ada 4898 jenis) yang semuanya produksi Portugal.

Harga dari wine sangat tergantung pada kualitas rasa yang bersifat abstrak. Biasanya ada sejumlah ahli rasa yang diminta mencicipi wine dan kemudian memberikan penilaian. Antar penilai bisa jadi sangat bervariasi opininya.

Kualitas wine juga tergantung pada beberapa hasil physicochemical tests yang dilakukan di laboratorium dengan memeriksa antara lain acidity, pH level, kandungan gula, serta kandungan beberapa senyawa kimia lain.

Akan sangat menarik kalau hasil pengukuran kualitas di laboratorium memiliki keterkaitan dengan penilaian subjektif oleh pakar rasa.

Data yang ada hanya berisi data penilaian wine putih, yang meliputi 12 karakteristik hasil uji laboratorium serta hasil penilaian ahli raser pada skala 1 – 10, dengan 1 untuk kualitas rasa yang paling buruk dan 10 untuk rasa yang paling baik. Ada tiga penilai, dan nilai akhir diperoleh dari median ketiganya.

Latihan

Gunakan data “white_wine2.csv”

Gunakan hanya 2 variabel prediktor:

- DENSITY
- ALCOHOL

Kelas kualitas:

Jika skor kualitas $> 6 \rightarrow$ Baik

Skor kualitas $< 6 \rightarrow$ Kurang

Beberapa ukuran ketepatan prediksi

Confusion Matrix/Classification Table: A breakdown of predictions into a table showing correct predictions (the diagonal) and the types of incorrect predictions made (what classes incorrect predictions were assigned).

Precision: A measure of a classifiers exactness.

Recall: A measure of a classifiers completeness

F1 Score (or F-score): A weighted average of precision and recall.

Kappa (or [Cohen's kappa](#)): Classification accuracy normalized by the imbalance of the classes in the data.

ROC Curves: Like precision and recall, accuracy is divided into sensitivity and specificity and models can be chosen based on the balance thresholds of these values.

Classification Table

Predicted Class

		Predicted Class		
		0	1	
Actual Class	0	True Negative	False Positive	Actual Negative
	1	False Negative	True Positive	Actual Positive
		Predicted Negative	Predicted Positive	

Proporsinya harus tinggi

Accuracy / Correct Classification Rate

		Predicted Class		
		0	1	
Actual Class	0	True Negative	False Positive	Actual Negative (N_2)
	1	False Negative	True Positive	Actual Positive (N_1)
		Predicted Negative	Predicted Positive	

$$\text{Correct Rate} = (\text{True Positive} + \text{True Negative}) / N$$

Sensitivity / Recall

		Predicted Class		
		0	1	
Actual Class	0			
	1		True Positive	Actual Positive (N_1)
		Predicted Positive		

SENSITIVITY adalah **TRUE POSITIVE** dibagi dengan N_1

Classification Table

		Predicted Class		
		0	1	
Actual Class	0	True Negative		Actual Negative (N_2)
	1			
		Predicted Negative		

SPECIFICITY adalah **TRUE NEGATIVE** dibagi dengan N_2

Specificity

		Predicted Class		
		0	1	
Actual Class	0	True Negative		Actual Negative
	1			
		Predicted Negative		

SPECIFICITY adalah TRUE NEGATIVE dibagi dengan N_2

Tentu saja $1 - \text{Specificity}$ adalah 1 dikurangi nilai specificity-nya, atau diperoleh dari FALSE POSITIVE dibagi dengan N_2 .

Jika cutoff nilai prediksi peluang yang digunakan kecil, maka individu cenderung akan diprediksi memiliki kategori 1 (YES). Sehingga pada saat nilai cutoff kecil, maka $(1 - \text{specificity})$ ini akan besar (mendekati 100%). Secara umum nilai $(1 - \text{specificity})$ ini akan menurun jika nilai cutoff ditingkatkan. Namun secara ideal kita menginginkan untuk berbagai nilai cutoff, model memiliki $(1 - \text{specificity})$ yang rendah.

Beberapa ukuran ketepatan prediksi

Precision: A measure of a classifiers exactness.

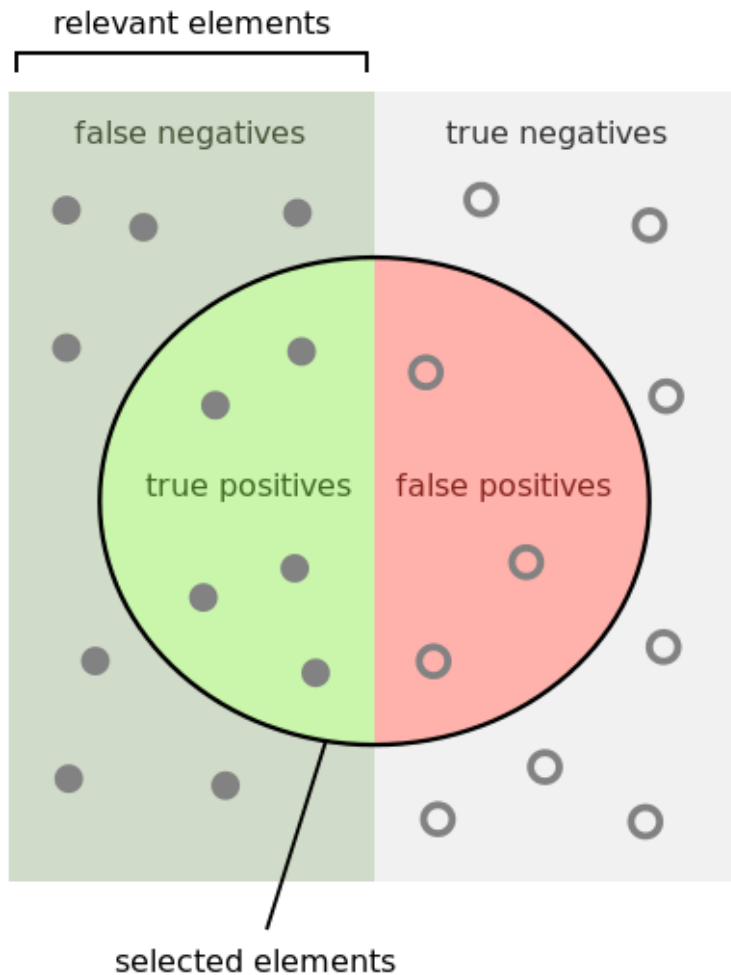
Recall: A measure of a classifiers completeness

F1 Score (or F-score): A weighted average of precision and recall.

Kappa (or Cohen's kappa): Classification accuracy normalized by the imbalance of the classes in the data.

		True condition			
Total population		Condition positive	Condition negative	$Prevalence = \frac{\sum \text{Condition positive}}{\sum \text{Total population}}$	$Accuracy (ACC) = \frac{\sum \text{True positive} + \sum \text{True negative}}{\sum \text{Total population}}$
Predicted condition	Predicted condition positive	True positive	False positive, Type I error	Positive predictive value (PPV), Precision = $\frac{\sum \text{True positive}}{\sum \text{Predicted condition positive}}$	False discovery rate (FDR) = $\frac{\sum \text{False positive}}{\sum \text{Predicted condition positive}}$
	Predicted condition negative	False negative, Type II error	True negative	False omission rate (FOR) = $\frac{\sum \text{False negative}}{\sum \text{Predicted condition negative}}$	Negative predictive value (NPV) = $\frac{\sum \text{True negative}}{\sum \text{Predicted condition negative}}$
<div>Click thumbnail for interactive chart:</div> <div><div><div>pop.</div><div>C+</div><div>C-</div><div>prev.</div><div>ACC</div></div><div><div>Pc+</div><div>T+</div><div>F+</div><div>PPV</div><div>FDR</div></div><div><div>Pc-</div><div>F-</div><div>T-</div><div>FOR</div><div>NPV</div></div><div><div>TPR</div><div>FPR</div><div>LR+</div><div>DOR</div></div><div><div>FNR</div><div>TNR</div><div>LR-</div><div>F1</div></div></div>		True positive rate (TPR), Recall, Sensitivity, probability of detection $= \frac{\sum \text{True positive}}{\sum \text{Condition positive}}$	False positive rate (FPR), Fall-out, probability of false alarm $= \frac{\sum \text{False positive}}{\sum \text{Condition negative}}$	$Positive \text{ likelihood ratio (LR+)} = \frac{TPR}{FPR}$	<div>Diagnostic odds ratio (DOR) $= \frac{LR+}{LR-}$</div> <div>$F_1 \text{ score} = \frac{2}{\frac{1}{Recall} + \frac{1}{Precision}}$</div>
		False negative rate (FNR), Miss rate $= \frac{\sum \text{False negative}}{\sum \text{Condition positive}}$	True negative rate (TNR), Specificity (SPC) $= \frac{\sum \text{True negative}}{\sum \text{Condition negative}}$	$Negative \text{ likelihood ratio (LR-)} = \frac{FNR}{TNR}$	

Sensitivity vs Precision



How many selected items are relevant?

$$\text{Precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}}$$

How many relevant items are selected?

$$\text{Recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}$$

The F1 Score is the $2 * ((\text{precision} * \text{recall}) / (\text{precision} + \text{recall}))$. It is also called the F Score or the F Measure.

Kappa

$$\kappa \equiv \frac{p_o - p_e}{1 - p_e} = 1 - \frac{1 - p_o}{1 - p_e},$$

where p_o is the relative observed agreement among raters (identical to [accuracy](#)), and p_e is the hypothetical probability of chance agreement, using the observed data to calculate the probabilities of each observer randomly seeing each category

$$p_e = p_{.1}p_{1.} + p_{.2}p_{2.}$$

Poor agreement = Less than 0.20

Fair agreement = 0.20 to 0.40

Moderate agreement = 0.40 to 0.60

Good agreement = 0.60 to 0.80

Very good agreement = 0.80 to 1.00

```
wine <- read.csv("white_wine2.csv")

data <- wine[,c("alcohol", "density", "quality")]

data$kelas.kualitas <- ifelse(data$quality > 6, 1, 2)

library(class)
prediksi <- knn(data[,1:2], data[,1:2], data[,4], k = 25)

library(caret)
confusionMatrix(prediksi, data[,4])
```

```
> confusionMatrix(prediksi, data[,4])
```

Confusion Matrix and Statistics

	Reference	
Prediction	1	2
1	913	86
2	147	3752

Accuracy : 0.9524

95% CI : (0.9461, 0.9582)

No Information Rate : 0.7836

P-Value [Acc > NIR] : < 2.2e-16

Kappa : 0.8568

Mcnemar's Test P-Value : 8.469e-05

Sensitivity : 0.8613

Specificity : 0.9776

Pos Pred Value : 0.9139

Neg Pred Value : 0.9623

Prevalence : 0.2164

Detection Rate : 0.1864

Detection Prevalence : 0.2040

Balanced Accuracy : 0.9195

'Positive' Class : 1

Validasi

Tidak semua amatan pada data digunakan sebagai data training

Sebagian disisihkan terlebih dahulu untuk dijadikan gugus data untuk validasi

Dilakukan pendugaan atau prediksi terhadap amatan pada gugus validasi, dan kemudian dinilai kemampuan prediksinya menggunakan data tersebut dengan membandingkan kelas hasil prediksi dan kelas yang sebenarnya

Validasi

```
set.seed(10)
acak <- sample(1:nrow(data), 4)
```

**Menentukan nomor baris
pembagian secara acak**

```
training <- data[-acak,1:2]
kelastraining <- as.factor(data[-acak,3])
```

**Membagi data
menjadi dua
bagian**

```
validasi <- data[acak,1:2]
kelasvalidasi <- as.factor(data[acak,3])
```

```
prediksi.validasi <- knn(training, validasi, kelastraining, k = 12)
prediksi.validasi
```

**Memprediksi
data validasi**

```
table(prediksi.validasi, kelasvalidasi)
```

```
mean(prediksi.validasi == kelasvalidasi)
```

**Membandingkan prediksi
dan kelas sebenarnya**



Lihat file program...

```
> confusionMatrix(prediksi, y.testing)
```

Confusion Matrix and Statistics

Prediction	Reference	
	1	2
1	366	32
2	32	1468

Accuracy : 0.9663

95% CI : (0.9571, 0.9739)

No Information Rate : 0.7903

P-Value [Acc > NIR] : <2e-16

Kappa : 0.8983

Mcnemar's Test P-Value : 1

Sensitivity : 0.9196

Specificity : 0.9787

Pos Pred Value : 0.9196

Neg Pred Value : 0.9787

Prevalence : 0.2097

Detection Rate : 0.1928

Detection Prevalence : 0.2097

Balanced Accuracy : 0.9491

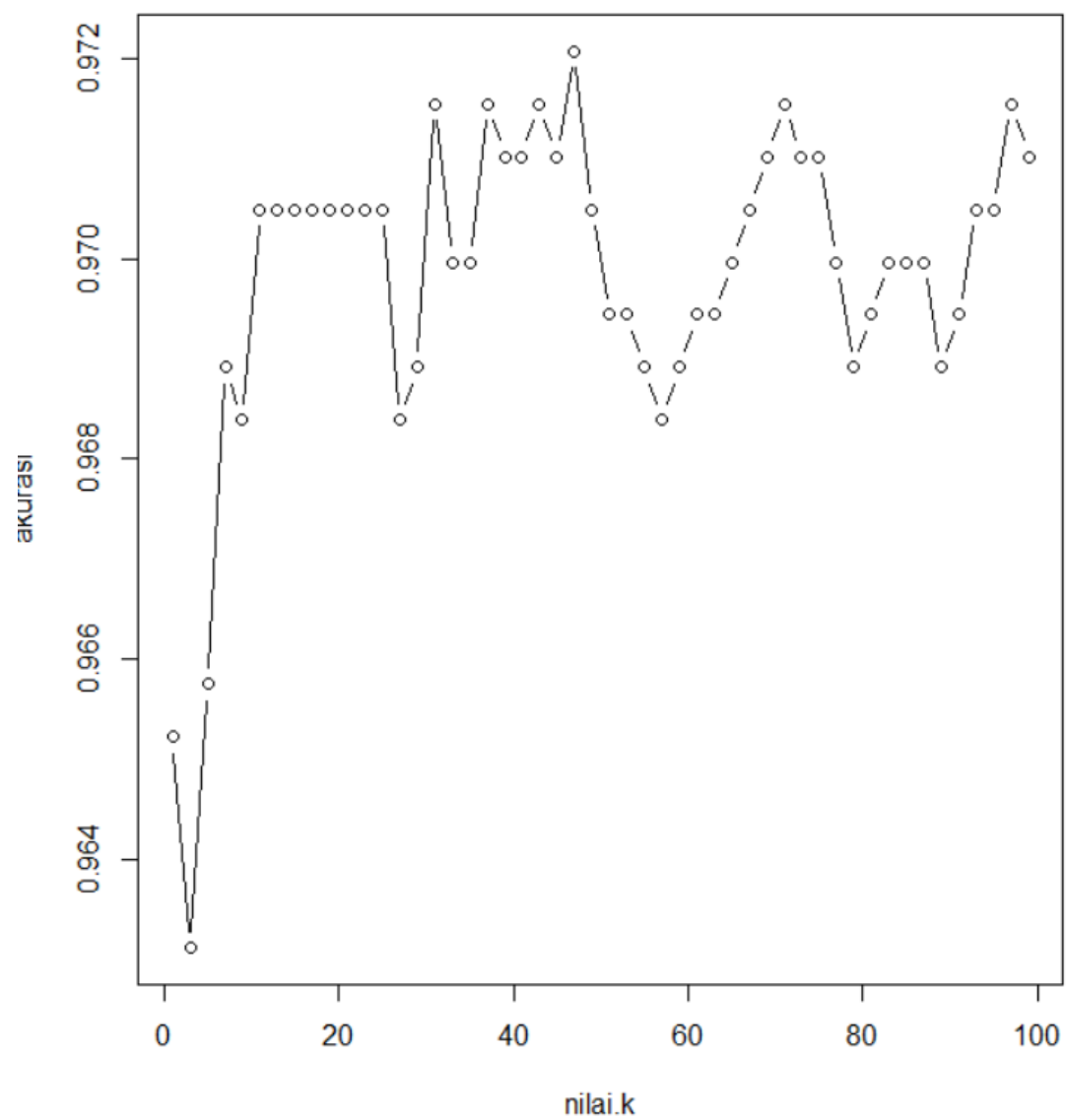
'Positive' Class : 1

Penentuan k-optimal dengan validasi

Jalankan algoritma k-nn untuk memprediksi kelas dari data testing

Kemudian ganti nilai k

Evaluasi nilai k berapa yang memberikan kinerja klasifikasi yang paling baik



Validasi Silang

Validasi vs Validasi Silang

- Pada proses validasi, setiap amatan hanya berperan sebagai anggota data latih/training saja atau sebagai anggota data validasi saja
- Pada proses validasi silang, memungkinkan suatu amatan berperan dalam dua jenis data tersebut di atas

LOO-CV (leave-one-out cross validation)

K-fold CV (k-fold cross validation)

Validasi Silang

LOO-CV (leave-one-out cross validation)

1. Andaikan terdapat n buah amatan
2. Sisihkan satu buah amatan sebagai gugus data validasi, sedangkan $(n-1)$ amatan lainnya sebagai gugus data training
3. Susun model analisis menggunakan data training, dan lakukan prediksi terhadap data validasi yang hanya berisi satu amatan tadi
4. Ulangi 2-3 dengan menggunakan satu amatan lainnya sebagai gugus validasi
5. Bandingkan hasil prediksi dengan nilai kelas aslinya

Validasi Silang

K-fold CV

1. Andaikan terdapat n buah amatan
2. Bagi gugus data menjadi k buah bagian yang berisi amatan dengan banyaknya yang sama (hampir sama)
3. Gunakan salah satu bagian sebagai gugus data validasi dan $k-1$ bagian lain sebagai gugus data training
4. Susun model analisis menggunakan data training, dan lakukan prediksi terhadap data validasi
5. Ulangi 3-4 dengan menggunakan satu bagian lainnya sebagai gugus validasi dan sisanya sebagai data training. Ulangi hingga seluruh k buah bagian pernah menjadi data validasi
6. Bandingkan hasil prediksi dengan nilai kelas aslinya

Biasanya orang menggunakan $k = 10$



Tugas Validasi Silang

susun program untuk melakukan validasi silang dalam menentukan k optimal