Mengukur Kebaikan Model

Bagus Sartono
Departemen Statistika – IPB University

IPB University
Bogor Indonesia

Bagus Sartono

Pengalaman Kerja

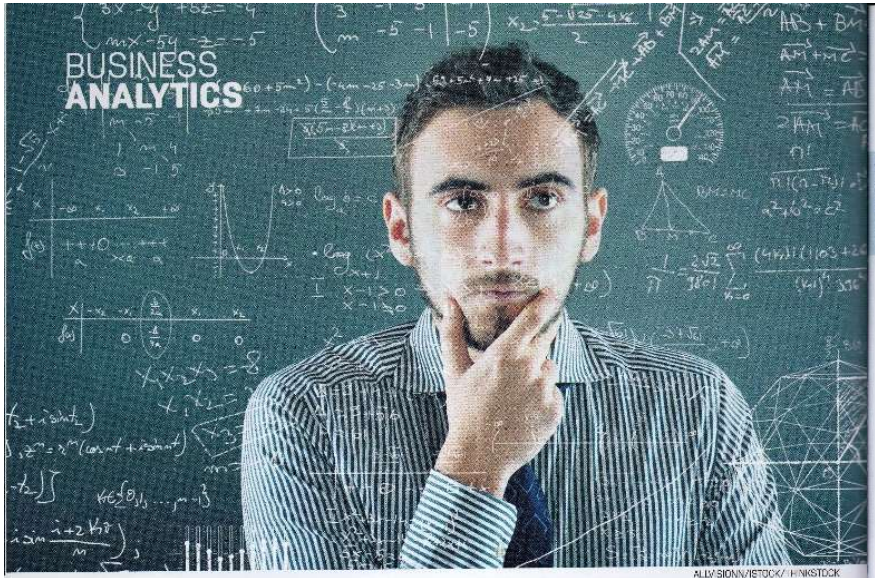
- 2000 – sekarang, Dosen di IPB (Departemen Statistika)
- 2002 – 2008, Statistician – PT Ganesha Cipta Informatika
- 2013, Technical Advisor – MarkPlus Insight
- 2013, Consultant – CIFOR
- 2014, Project Leader – PT Trans Intra Asia
- 2015 – 2017, Data Science Advisor – Starcore Analytics
- 2018, Consultant – IACCB

Pengalaman Lainnya

- Trainer bidang Statistika, Data Mining, Analitika di berbagai instansi, antara lain Bank Mandiri, Bank Syariah Mandiri, Bank Sinar Mas, Bank Danamon, Bank Indonesia, OJK, Telkomsel, Bank Permata, CIMB Niaga, LIPI, Kementerian Pertanian, Kementerian Keuangan
- Tenaga ahli di beberapa kegiatan kajian, antara lain Kementerian Pendidikan dan Kebudayaan, BPJS, Kementerian Keuangan, OJK, Bank Indonesia, LIPI, Astra Digital.
- Penulis rubrik Business Analytics di majalah InfoKomputer

Outline

- Kriteria Kebaikan Model
- Strategi Memperoleh Model yang Baik



ALLVISIONN/ISTOCK/THINKSTOCK

BAGIAN 1

Tentang Kriteria Pemilihan Model

Memilih suatu model tidaklah mudah. Setidaknya ada dua kriteria yang harus dipertimbangkan, yakni *goodness of fit* dan kesederhanaan.

PENULIS: 
BAGUS SARTONO
Dosen IPB University (Departemen Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam)


ALFIAN FUTUHUL HADI
Dosen Universitas Nahdlatul Wathan Mataram

BEBERAPA PEKAN yang lalu ada salah seorang teman yang menyampaikan pertanyaan tentang bagaimana memilih model. Dia mengatakan bahwa saat ini dia memiliki tiga buah kandidat model untuk dipilih dan tim di perusahaannya tidak sepakat untuk menentukan mana yang akan digunakan. Selanjutnya penulis menyarankan untuk mempertimbangkan tidak memilih salah satu yang dianggap paling baik, tetapi menggunakan ketiganya untuk melakukan prediksi dengan teknik *ensemble* (lihat *InfoKomputer* edisi Desember 2016). Saran tersebut tidak sepenuhnya dapat diterima oleh beliau dengan pertimbangan kemudahan, dan tetap pada permasalahan semula yaitu memilih satu dari beberapa model yang ada.

Model statistika pada prinsipnya adalah suatu upaya dari kita untuk mendekati proses dari suatu kejadian atau fenomena yang terjadi di sekitar kita. Munculnya kejadian tertentu diyakini atau diasumsikan dipengaruhi oleh hal-hal atau kondisi-kondisi tertentu.

Dalam pemodelan selanjutnya kejadian yang menjadi perhatian tersebut dijadikan sebagai target atau *output*, sedangkan kondisi-kondisi yang berpengaruh terhadap kejadian itu dikenal sebagai input. Padanan istilah *output-input* yang banyak digunakan adalah respons-*predictor* atau *dependent-independent*.

Model yang dibangun tentu saja diharapkan sangat mendekati proses yang sesungguhnya (yang sayangnya dalam banyak hal kita tidak pernah mengetahui dengan pasti). Karena ingin sangat mendekati yang sesungguhnya, jelas bahwa kriteria pertama yang dijadikan pertimbangan untuk menentukan model yang baik adalah ketepatan prediksi *output* yang dikenal dengan istilah *goodness of fit*.

Secara naluriah, manusia lebih menyukai hal-hal yang sederhana dan tidak rumit, sehingga muncul kriteria kedua yaitu kesederhanaan.

Kriteria Pertama: Goodness of Fit

Besaran ukuran *goodness of fit* berkenaan dengan seberapa mirip prediksi kejadian *output* yang dihasilkan oleh model dengan kejadian yang sebenarnya. Pada model-model dengan *output* numerik seperti regresi linear dikenal beberapa ukuran seperti R-squared (dikenal dengan nama koefisien determinasi), MAPE (*mean absolute percentage error*) dan MAD (*mean absolute deviation*).

Ukuran-ukuran tersebut pada dasarnya adalah selisih antara nilai aktual dengan nilai prediksi dari model. Makin kecil selisihnya maka model dikatakan makin baik. Pendekatan pemodelan yang menerapkan kriteria ini umumnya akan menggunakan metode yang didasarkan pada jarak antara nilai data dengan dugaan model, seperti metode *least squares* dengan berbagai macam variasinya. Model dengan R-squared besar, MAPE dan MAD kecil adalah model yang lebih disukai.

Sementara itu kriteria *goodness of fit* pada model-model dengan *output* berupa kelas (*output* yang bersifat kategorik) memiliki ukuran seperti akurasi, sensitivitas, *specificity*, *area under the ROC-curve*, serta beberapa ukuran lainnya. Semuanya didasarkan pada kesamaan antara kelas hasil prediksi dengan kelas yang sebenarnya. Makin besar persentase kesamaan antar keduanya maka modelnya disebut makin baik. Terdapat pula ukuran lain yang tidak berpikir hanya sama atau tidak sama antara prediksi dan aktual, tetapi didasarkan pada kejadian aktual dengan peluang terjadinya seperti ukuran *log loss*.

Pada pemodelan statistika atau *statistical learning* yang menggunakan pendekatan *maximum likelihood* juga dikenal ukuran seperti nilai dari *likelihood function*. Untuk kemudahan komputasi, sering yang digunakan adalah logaritma dari *likelihood function*. Ukuran ini pada prinsipnya ingin melakukan pencocokan distribusi sebaik mungkin dari model terhadap data yang dimiliki.

Secara umum (meskipun tidak selalu terjadi), model-model yang makin kompleks atau makin rumit bentuknya akan memberikan *goodness of fit* yang makin baik. Model polinomial dengan derajat tinggi dan nonlinear akan terlihat memiliki ukuran ketepatan prediksi yang lebih baik dibandingkan model linear.

Model yang melibatkan ratusan variabel *predictor* umumnya akan memberikan prediksi yang lebih baik dibandingkan model yang hanya menggunakan 5 atau 6 buah variabel *predictor*. Namun, secara naluriah, manusia lebih menyukai hal sederhana dan tidak rumit sehingga muncul kriteria kedua yaitu kesederhanaan.

Kriteria Kedua: Kesederhanaan

Jika bisa menggunakan model yang sederhana, buat apa yang rumit? Bukankah kalau sederhana biasanya lebih mudah dipahami? Kalau melibatkan sedikit variabel *predictor* kan lebih mudah dan lebih murah dalam mengumpulkan data?

Pertanyaan-pertanyaan di atas adalah pertanyaan yang menyiratkan bahwa secara naluriah manusia lebih menyukai yang sederhana (*parsimonious*). Sehingga kemudian muncul ukuran-ukuran seperti banyaknya parameter model, derajat bebas *error*, banyaknya *node* pada *tree*, dan sebagainya. Beberapa orang berpandangan tidak masalah kalaupun ukuran ketepatan prediksi berkurang sedikit, asalkan modelnya bisa jauh lebih sederhana.

Berbicara tentang prinsip kesederhanaan, banyak orang merujuk pada apa yang disebut *Occam's razor* yang dikemukakan oleh filsuf William of Ockham (1287–1347). Prinsipnya kira-kira berbunyi bahwa dari sekian banyak hipotesis yang harus diambil adalah yang memiliki paling sedikit asumsi. Dalam beberapa tulisan terdapat kutipannya "*Numquam ponenda est pluralitas sine necessitate*" dengan terjemahan bebas penulis adalah untuk apa menambah rumit jika tidak banyak membantu dan diperlukan.

Dengan dasar pemikiran dari kedua kriteria utama yang disebutkan di atas, kemudian berkembanglah kriteria-kriteria yang kalau ditelisik tidak lain adalah gabungan dari kedua kriteria yang telah disebutkan, yaitu memiliki ketepatan prediksi yang tinggi dengan tetap memperhatikan kesederhanaan bentuk model.

BAGIAN 2

Tentang Kriteria Pemilihan Model

Model yang baik tidak hanya harus memiliki ketepatan yang tinggi, melainkan juga bentuk yang sederhana.

PENULIS:

BAGUS SARTONO
Dosen di Departemen Statistika Institut Pertanian Bogor

ALFIAN FUTUHUL HADI
Dosen di Jurusan Matematika Universitas Jember

PADA BAGIAN pertama telah disebutkan bahwa ada dua kriteria utama yang digunakan orang dalam memilih model, yaitu *goodness of fit* yang merepresentasikan ketepatan model, dan kesederhanaan yang lebih mengutamakan model tak kompleks dengan sedikit parameter. Lalu apa selanjutnya?

Dengan dasar pemikiran dari kedua kriteria utama yang disebutkan di atas, kemudian berkembanglah kriteria-kriteria lain. Berbagai kriteria ini kalau ditelisik tidak lain adalah gabungan dari kedua kriteria utama yang telah disebutkan itu, yakni memiliki ketepatan prediksi yang tinggi dengan tetap memperhatikan kesederhanaan bentuk model.

Trade-off Kedua Kriteria Utama

Dalam diskusi-diskusi awal di model regresi linear misalnya, terdapat kriteria yang disebut sebagai *adjusted R-squared*. Nilai *R-squared* yang asli tidak dapat digunakan dengan mudah untuk memilih model karena tambahan variabel *predictor* dalam suatu model selalu meningkatkan nilai *R-squared*. Pemikiran kesederhanaan kemudian diadopsi dengan memasukkan derajat bebas *error* untuk mengoreksi dan menjadi pertimbangan sehingga terbentuklah ukuran *adjusted R-squared*.

Para ahli statistik juga mengusulkan penggunaan evaluasi terhadap perlu tidaknya tambahan variabel *predictor* melalui analisis seperti pengujian secara *sequential* semacam yang ditemukan pada *sequential sum of squares* dan *likelihood ratio test*. Ide dasarnya adalah membandingkan *goodness of fit*

FREEPIK.COM

Proses semacam *cross-validation* perlu juga dilalui sebagai bagian dari kehati-hatian dalam melakukan pemilihan model ini.

dari model yang rumit (memiliki lebih banyak parameter) dengan model yang sederhana (memiliki sedikit parameter).

Kriteria serupa juga ditemukan dalam banyak bentuk seperti *Mallows's* C_p, *Akaike's information criterion*, serta *Bayesian information criterion*.

Kriteria ini bekerja dengan memiliki model yang memaksimumkan ukuran ketepatan prediksi yang dipenalti oleh ukuran kekompleksan model. Dengan menerapkan kriteria tersebut maka model yang terbaik kira-kira adalah yang memiliki ketepatan yang tinggi (tetapi tidak harus sangat tinggi) dengan bentuk yang sederhana. Dengan kata lain, *trade-off* kedua kriteria utama.

Tidak hanya kriteria-kriteria di atas yang mempertimbangkan kedua aspek penilaian terhadap model. Telah dikembangkan pula kriteria yang memerlukan komputasi lebih sulit yaitu kriteria *minimum description length* dan *Bayes factor*. Keduanya relatif kurang populer digunakan saat ini, tetapi tidak mustahil akan lebih banyak digunakan di kemudian hari karena kemampuan komputasi makin baik dari waktu ke waktu.

Diskusi di atas menyiratkan bahwa proses penilaian kebaikan untuk memilih model dilakukan setelah kandidat-kandidat model terbentuk. Namun, saat ini kita juga dapat melakukan berbagai proses secara otomatis sehingga proses pemilihan model dapat dilakukan secara simultan dengan proses pemodelan itu sendiri. Bentuk-bentuk *penalized regression* dan penyusutan (*shrinkage*) terhadap koefisien telah juga berkembang luas seperti teknik LASSO dan SCAD.

Kriteria Ketiga: Masih Ada Lagi?

Di luar diskusi mengenai kriteria pemilihan model secara kuantitatif di atas, ada hal lain yang sering juga menjadi pertimbangan *modeler* dalam memilih model, yaitu masalah *interpretability*. Dengan mempertimbangkan kriteria ini, biasanya *modeler* akan melihat apakah modelnya memiliki nilai parameter/koefisien yang "masuk akal" dalam konteks tertentu.

Misalnya saja mereka akan melihat apakah tanda positif maupun negatif dari koefisien sesuai dengan keyakinan umum atau cocok dengan teori yang dipegang. Seorang *modeler* mungkin akan mengabaikan model yang secara *predictive* sangat baik, tetapi koefisien regresinya berlawanan arah dengan teori. Pasalnya dia khawatir tidak mudah menjelaskannya kepada orang lain.

Tidak hanya masalah tanda positif dan negatif, sebelum penyusunan model, seorang *modeler* juga memperkirakan bahwa koefisien yang satu akan lebih besar dibandingkan koefisien yang lain didasarkan pada teori atau pendapat tertentu. Koefisien ini umumnya merupakan representasi dari besar-kecilnya efek *predictor* terhadap *output*.

Sebagai contoh, pada pemodelan untuk memprediksi pertumbuhan pangsa pasar produk otomotif, banyak analis mengatakan bahwa faktor pertumbuhan ekonomi secara makro memiliki efek yang lebih besar dibandingkan faktor harga jual. Pada beberapa konteks, pertimbangan mengenai *magnitude* dari koefisien ini juga digunakan dalam pemilihan model. Akibatnya kalau urutan dari besaran efek suatu model terbolak-balik, model tersebut menjadi tidak disukai.

Sebagai penutup, memang proses pemilihan model ini sering menjadi satu permasalahan tersendiri dalam proses analitik. Meskipun tersedia banyak pilihan pendekatan dan kriteria yang dikembangkan di berbagai literatur telah diimpelementasikan dalam *software*, analis tetap perlu melakukan secara hati-hati proses ini agar didapatkan model yang benar-benar paling baik dari berbagai aspek.

Proses semacam *cross-validation* perlu juga dilalui sebagai bagian dari kehati-hatian dalam melakukan pemilihan model ini. Selain itu, diskusi dengan individu-individu di dalam perusahaan atau organisasi yang memahami konteks data dan permasalahan juga penting untuk ditempuh. Pemilihan model bukanlah suatu proses yang bersifat hitam putih yang didasarkan pada *hard criteria* tertentu saja.

Kriteria Kebaikan Model

- Ketepatan Prediksi (goodness of fit)
- Kesederhanaan Model (model parsimony)

- → AIC menggabungkan keduanya

Information Criteria

- **Information criterion** is a measure of the goodness of fit of an estimated statistical model.
- It is grounded in the concept of entropy,
  - offers a relative measure of the information lost
  - describes the tradeoff precision and complexity of the model.
- An IC is not a test on the model in the sense of hypothesis testing
- it is a tool for model selection.
- Given a data set, several competing models may be ranked according to their IC
- The model with the lowest IC is chosen as the “best”

Information Criteria

- IC rewards goodness of fit, but also includes a penalty that is an increasing function of the number of estimated parameters.
- This penalty discourages overfitting.
- The IC methodology attempts to find the model that **best explains the data with a minimum of free parameters**.
- IC judges a model by how close its fitted values tend to be to the true values.
- the AIC value assigned to a model is only meant to *rank* competing models and tell you which is the best among the given alternatives.

Akaike Information Criteria (AIC)

$$AIC = -2\log Lik + 2p$$

Akaike, Hirotugu (1974). "A new look at the statistical model identification". *IEEE Transactions on Automatic Control* **19** (6): 716–723..

Ketepatan Prediksi Model Klasifikasi

		Aktual		
		Yes	No	
Prediksi	Yes	TP	FP	
	No	FN	TN	
				N

ACCURACY = (TP + TN) / N

Sensitivity = TP / (TP + FN), recall, true positive rate

Specificity = TN / (FP + TN)

Precision = TP / (TP + FP), positive predictive value

F1 score = 2 x precision x sensitivity / (precision + sensitivity)

Penilaian Ketepatan Prediksi

- Gunakan data lain selain data training
- Best practice:
  - Pisahkan data menjadi dua bagian: data training + data testing. Proporsi umum
  - Susun model menggunakan data training
  - Lakukan pengukuran ketepatan prediksi menggunakan data testing

Ilustrasi

- Gunakan file program “ketepatan prediksi 01.R”
  - Data yang digunakan myopia.csv
  - Data dibagi menjadi dua bagian: 70% training set, 30% testing set
  - Model regresi logistik dibangun menggunakan data training
  - Ketepatan prediksi diukur berdasarkan hasil prediksi terhadap data testing

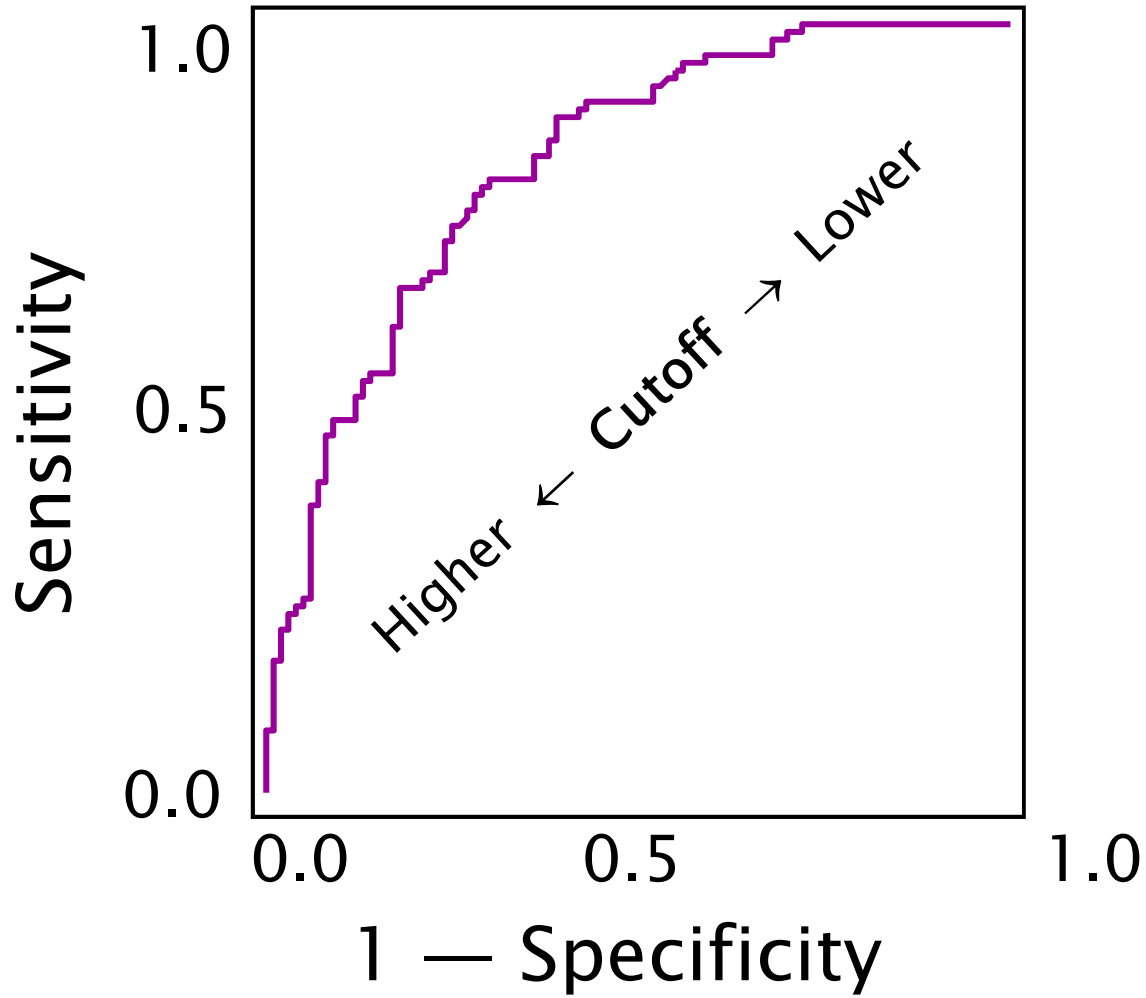
Kurva ROC

- Perubahan cut-off peluang dapat mengubah nilai ketepatan prediksi
  - Cut-off diperkecil → sensitivity naik, specificity turun

- Dari model yang sama, jika cut-off peluang diubah-ubah kita akan memperoleh nilai sensitivity dan specificity yang berbeda-beda

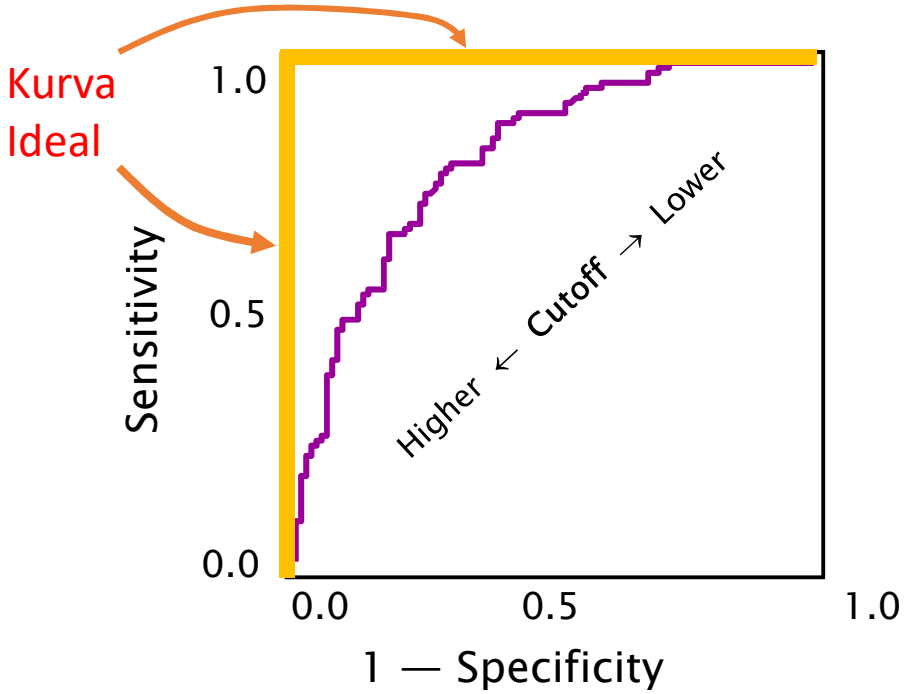
- Kurva ROC menggambarkan kedua nilai ketepatan prediksi tersebut untuk berbagai cut-off

ROC (receiver operating characteristic) Curve



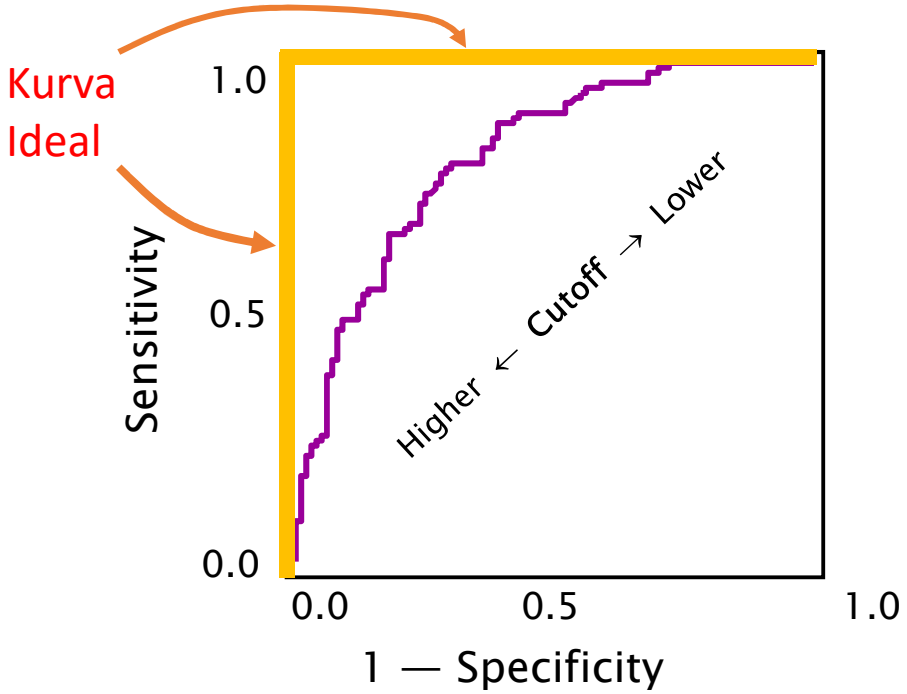
Kurva ROC

- Kondisi ideal
  - Sensitivity = 100%
  - Specificity = 100%, atau 1 – Specificity = 0



Kurva ROC

- Semakin mendekati kurva ideal, model semakin baik
- Ukuran yang digunakan → luas di bawah kurva (AUC = area under the curve)



Ilustrasi

- Gunakan file program “ketepatan prediksi 02.R”
- Konten:
  - Melihat dampak perubahan cut-off peluang terhadap nilai ketepatan prediksi
  - Menggambar plot kurva ROC
  - Menghitung nilai AUC

Latihan

- Lengkapi tabel berikut dengan menghitung nilai AIC dan AUC untuk model-model regresi logistik yang memuat variabel prediktor sesuai yang tercantum pada tabel.

Variabel Prediktor dalam Model	AIC	AUC
spheq + sporthr + readhr + mommy + dadmy		
spheq + sporthr + readhr + mommy + dadmy + diopterhr		
spheq + mommy + dadmy + diopterhr		

Strategi Memperoleh Model yang Baik

- Beberapa yang bisa dilakukan dalam memperoleh model regresi logistik yang baik:
  - Seleksi Variabel → Gunakan Information Value
  - Diskretisasi variabel prediktor numerik

Information Value

- Merupakan ukuran yang menggambarkan hubungan antara variabel prediktor (X) dengan variabel respon (Y) yang bersifat biner
- Information Value menggambarkan perbedaan antara distribusi data prediktor X pada dua kelas variabel respon
- IV yang semakin besar menunjukkan bahwa

Introduction

- These two concepts - weight of evidence (WOE) and information value (IV) evolved from the logistic regression technique.
- These two terms have been in existence in credit scoring world for more than 4-5 decades.
- They have been used as a benchmark to screen variables in the credit risk modeling projects such as probability of default.
- They help to explore data and screen variables.
- It is also used in marketing analytics project such as customer attrition model, campaign response model etc.

What is Weight of Evidence (WOE)?

- The weight of evidence tells the predictive power of an independent variable in relation to the dependent variable.
- Since it evolved from credit scoring world, it is generally described as a measure of the separation of good and bad customers.
- **"Bad Customers"** refers to the customers who defaulted on a loan. and **"Good Customers"** refers to the customers who paid back loan.

$$WOE = \ln\left(\frac{\text{Distribution of Goods}}{\text{Distribution of Bads}}\right)$$

- Many people do not understand the terms goods/bads as they are from different background than the credit risk.
- It's good to understand the concept of WOE in terms of **events and non-events**.
- It is calculated by taking the natural logarithm (log to base e) of division of % of non-events and % of events.

$$WOE = \ln\left(\frac{\%\ \text{of non-events}}{\%\ \text{of events}}\right)$$

Steps of Calculating WOE

- For a continuous variable, split data into 10 parts (or lesser depending on the distribution).
- Calculate the number of events and non-events in each group (bin)
- Calculate the % of events and % of non-events in each group.
- Calculate WOE by taking natural log of division of % of non-events and % of events

Note : For a categorical variable, you do not need to split the data (ignore Step 1 and follow the remaining steps)

Steps of Calculating WOE

Range	Bins	Non events	Events	% of Non-Events	% of Events	WOE
0-50	1	197	20	5.4%	5.9%	-0.0952
51-100	2	450	34	12.3%	10.1%	0.2002
101-150	3	492	39	13.4%	11.5%	0.1522
151-200	4	597	51	16.3%	15.1%	0.0774
201-250	5	609	54	16.6%	16.0%	0.0401
251-300	6	582	55	15.9%	16.3%	-0.0236
301-350	7	386	41	10.5%	12.1%	-0.1405
351-400	8	165	23	4.5%	6.8%	-0.4123
>401	9	184	21	5.0%	6.2%	-0.2123
	Total	3662	338			

Rules related to WOE

- Each category (bin) should have at least 5% of the observations.
- Each category (bin) should be non-zero for both non-events and events.
- The WOE should be distinct for each category. Similar groups should be aggregated.
- Missing values are binned separately.

Handle Zero Event/ Non-Event

- If a particular bin contains no event or non-event, you can use the formula below to ignore missing WOE. We are adding 0.5 to the number of events and non-events in a group.

  AdjustedWOE = ln (((Number of non-events in a group + 0.5) / Number of non-events)) / ((Number of events in a group + 0.5) / Number of events))

What is Information Value (IV)?

- Information value is one of the most useful technique to select important variables in a predictive model. It helps to rank variables on the basis of their importance. The IV is calculated using the following formula :

$$IV = \sum (\%\ \text{of non-events} - \%\ \text{of events}) * WOE$$

Rules related to Information Value

Information Value	Variable Predictiveness
Less than 0.02	Not useful for prediction
0.02 to 0.1	Weak predictive Power
0.1 to 0.3	Medium predictive Power
0.3 to 0.5	Strong predictive Power
>0.5	Suspicious Predictive Power (check once)

Hands-On

- Buka file “information value.R”

Diskretisasi

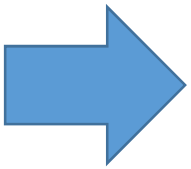
- Data terdiri atas banyak variabel dengan berbagai format/tipe:
  - Numerik diskret
  - Numerik kontinu
  - Kategorik ordinal
  - Kategorik nominal
- Variabel yang bertipe numerik dapat diubah menjadi kategorik (ordinal) → prosesnya dikenal sebagai **diskretisasi**, ada juga yang menyebut sebagai **binning**
- Diskretisasi ini sering membantu dalam pemodelan prediktif

Diskretisasi

Andaikan dataset berisi *N* observasi, proses diskretiasi terhadap variabel numerik *A* adalah mengubah nilai variabel tersebut menjadi *m* interval $D = \{[d_0, d_1], (d_1, d_2], \ldots, (d_{m-1}, d_m]\}$, dengan d_0 adalah nilai terkecil, d_m adalah nilai terbesar, dan $d_i < d_{i+i}$, untuk $i = 0, 1, \ldots, m-1$.

Diskretisasi

6.58
15.35
14.24
6.22
1.82
2.11
13.77
5.65
15.58
12.46
13.05
11.64
10.91
14.31
7.42

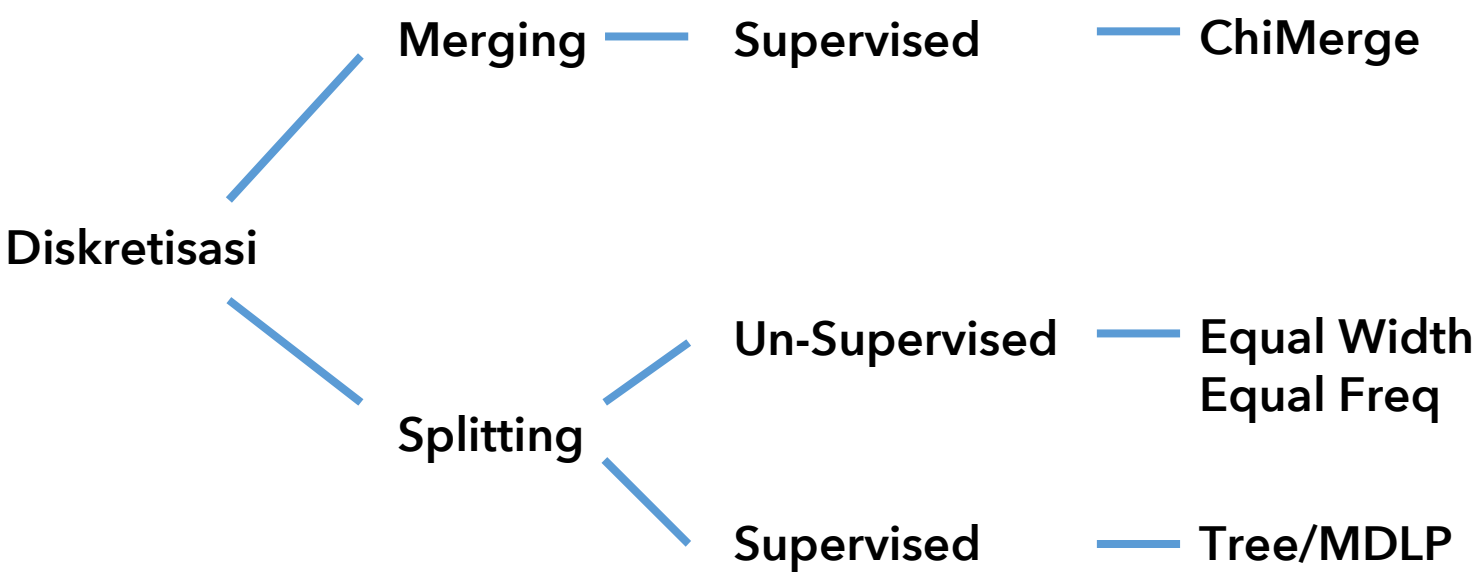


$X \leq 5$

$5 < X \leq 10$

$10 < X \leq 15$

$X > 15$

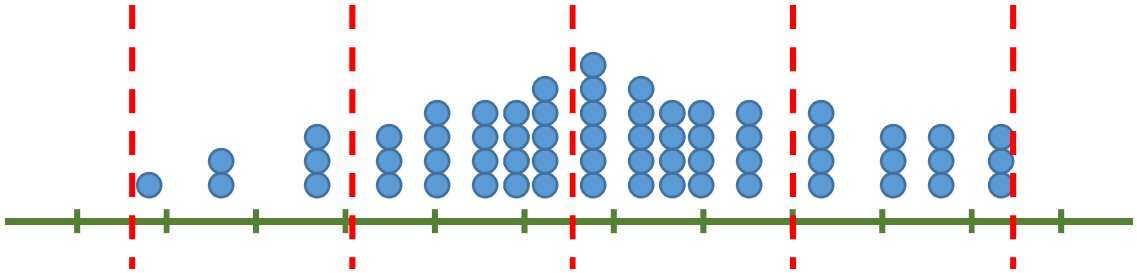

Diskretisasi
Merging
Supervised
ChiMerge
Splitting
Un-Supervised
Equal Width
Equal Freq
Supervised
Tree/MDLP

Equal Width dan Equal Frequency

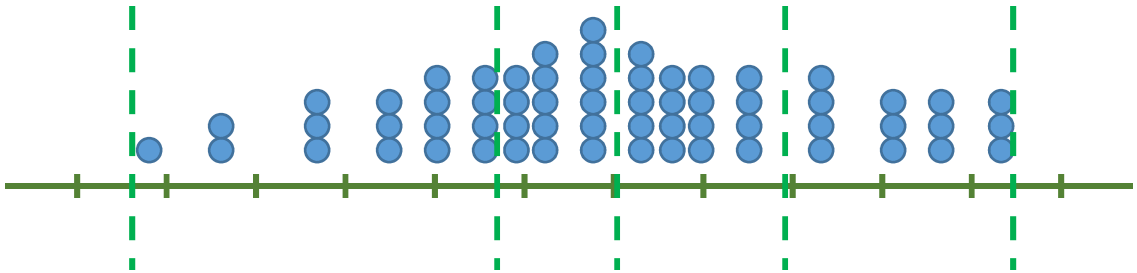
- In equal width, the continuous range of a feature is divided into intervals that have an equal width and each interval represents a bin. The arity can be calculated by the relationship between the chosen width for each interval and the total length of the attribute range.

- In equal frequency, an equal number of continuous values are placed in each bin. Thus, the width of each interval is computed by dividing the length of the attribute range by the desired arity.

Unsupervised Discretization: Equal Width Discretization



Unsupervised Discretization: Equal Freq Discretization



```
x <- c(15, 4, 21, 11, 16, 18, 24, 26, 28)
library(classInt)

#equal width
eqwid <- classIntervals(x, 4, style = 'equal')
eqwid$brks
```

```
> eqwid$brks
[1]  4 10 16 22 28
```

```
x.eqwid <- cut(x, breaks=eqwid$brks, include.lowest=TRUE)
cbind(x, x.eqwid)
```

```
> cbind(x, x.eqwid)
      x x.eqwid
 [1,] 15       2
 [2,]  4       1
 [3,] 21       3
 [4,] 11       2
 [5,] 16       2
 [6,] 18       3
 [7,] 24       4
 [8,] 26       4
 [9,] 28       4
```

```
#equal freq
eqfreq <- classIntervals(x, 4, style = 'quantile')
eqfreq$brks
```

```
> eqfreq$brks
[1]  4 15 18 24 28
```

```
x.eqfreq <- cut(x, breaks=eqfreq$brks, include.lowest=TRUE)
cbind(x, x.eqwid, x.eqfreq)
```

```
> cbind(x, x.eqwid, x.eqfreq)
      x x.eqwid x.eqfreq
 [1,] 15       2        1
 [2,]  4       1        1
 [3,] 21       3        3
 [4,] 11       2        1
 [5,] 16       2        2
 [6,] 18       3        2
 [7,] 24       4        3
 [8,] 26       4        4
 [9,] 28       4        4
```

Ilustrasi efek diskretisasi terhadap kualitas model prediktif

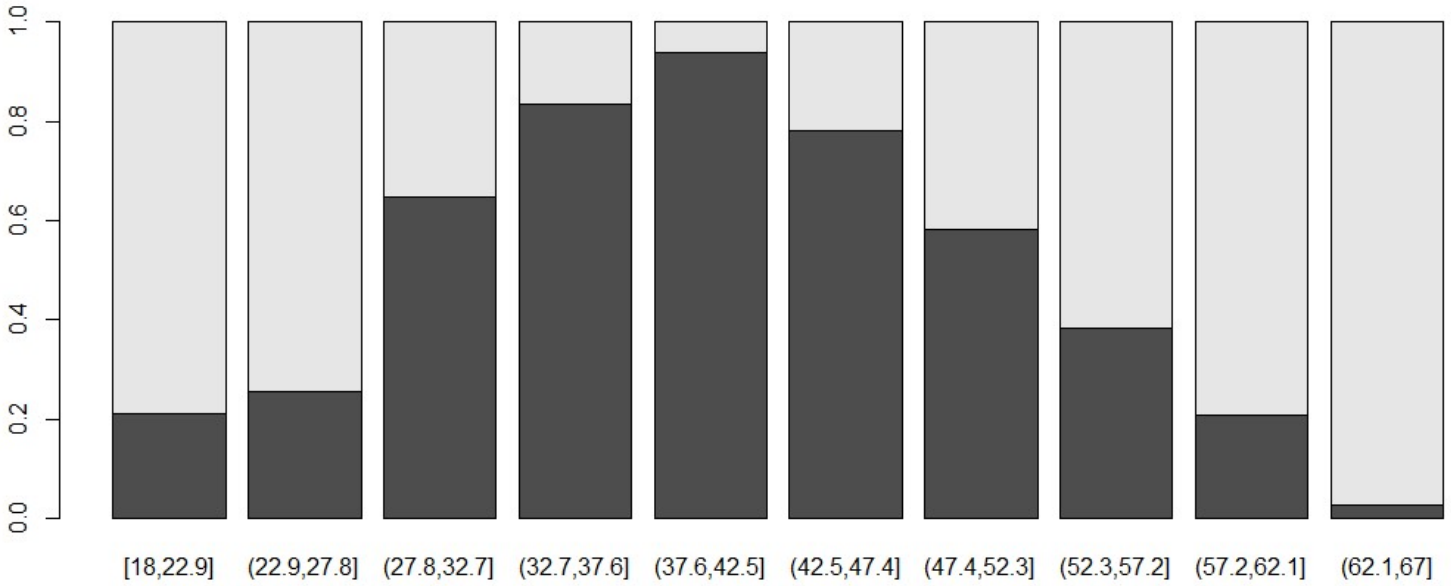
- Akan dipaparkan situasi dimana diskretisasi mampu memberikan peningkatan akurasi prediksi pada model regresi logistik
- Akan dibandingkan akurasi dua model dengan data yang sama
  - Model pertama menggunakan variabel prediktor asli
  - Model kedua menggunakan variabel prediktor yang telah didiskretkan

- **Buka file program “diskretisasi.R”**

```
> table(data.ts$class, prediksi.asli)
   prediksi.asli
      0   1
  0  93 134
  1  94 137
> mean(data.ts$class == prediksi.asli)
[1] 0.5021834
```

```
> table(data.ts$class, prediksi.disk)
   prediksi.disk
      0   1
  0 177  50
  1  73 158
> mean(data.ts$class ==
prediksi.disk)
[1] 0.731441
```

```
proporsi <- prop.table(table(data.tr$x.disk, data.tr$class), margin=1)
barplot(t(proporsi))
```



Description of Representative Methods

Mergin

Algorithm 2 Merging Algorithm

Require: S = Sorted values of attribute A
  procedure MERGING(S)
    if StoppingCriterion() == true **then**
      Return
    end if
    T = GetBestAdjacentIntervals(S)
    S = MergeAdjacentIntervals(S,T)
    Merging(S)
  end procedure

Description of Representative Methods

Merging

ChiMerge — χ^2 is a statistical measure that conducts a significance test on the relationship between the values of an attribute and the class. This statistic determines the similarity of adjacent intervals based on some significance level. Actually, it tests the hypothesis that two adjacent intervals of an attribute are independent

$$\chi^2 = \sum_{i=1}^{2} \sum_{j=1}^{c} \frac{(N_{ij} - E_{ij})^2}{E_{ij}}$$

Description of Representative Methods

Merging

ChiMerge —

$$
\begin{aligned}
c &= \text{number of classes} \\
N_{ij} &= \text{number of distinct values in the } i\text{th interval, } j\text{th class} \\
R_i &= \text{number of examples in } i\text{th interval} = \sum_{j=1}^{c} N_{ij} \\
C_j &= \text{number of examples in } j\text{th class} = \sum_{i=1}^{m} N_{ij} \\
N &= \text{total number of examples} = \sum_{j=1}^{c} C_j \\
E_{ij} &= \text{expected frequency of } N_{ij} = (R_i \times C_j)/N
\end{aligned}
$$

It is a supervised, bottom-up discretizer. At the beginning, each distinct value of the attribute is considered to be one interval. χ^2 tests are performed for every pair of adjacent intervals. Those adjacent intervals with the least χ^2 value are merged until the chosen stopping criterion is satisfied.

Description of Representative Methods

Splitting

Algorithm 1 Splitting Algorithm

Require: S = Sorted values of attribute A
  procedure SPLITTING(S)
    if StoppingCriterion() == true **then**
      Return
    end if
    T = GetBestSplitPoint(S)
    S_1 = GetLeftPart(S,T)
    S_2 = GetRightPart(S,T)
    Splitting(S_1)
    Splitting(S_2)
  end procedure

Description of Representative Methods

Splitting

MDLP — This discretizer uses the entropy measure to evaluate candidate cut points. Entropy is one of the most commonly used discretization measures in the literature. The entropy of a sample variable *X* is

$$H(X) = -\sum_{x} p_x \log p_x$$

where x represents a value of *X* and p_x its estimated probability of occurring.

Description of Representative Methods

Splitting

MDLP — Information is high for lower probable events and low otherwise. This discretizer uses the *Information Gain* of a cut point, which is defined as

$$G(A, T; S) = H(S) - H(A, T; S) = H(S) - \frac{|S_1|}{N} H(S_1) - \frac{|S_2|}{N} H(S_2)$$

where *A* is the attribute in question, *T* is a candidate cut point and *S* is the set of *N* examples. So, S_i is a partitioned subset of examples produced by *T*. The MDLP discretizer applies the *Minimum Description Length Principle* to decide the acceptation or rejection for each cut point and to govern the stopping criterion.

$$G(A, T; S) > \frac{\log_2(N-1)}{N} + \frac{\delta(A, T; S)}{N}$$

where $\delta(A, T; S) = \log_2(3^c - 2) - [c \cdot H(S) - c_1 \cdot H(S_1) - c_2 \cdot H(S_2)]$

Ilustrasi efek diskretisasi terhadap kualitas model prediktif

- Akan dipaparkan situasi dimana diskretisasi mampu memberikan peningkatan akurasi prediksi pada model regresi logistik
- Akan dibandingkan akurasi dua model dengan data yang sama
  - Model pertama menggunakan variabel prediktor asli
  - Model kedua menggunakan variabel prediktor yang telah didiskretkan

- **Buka file program "UNP diskretisasi.R"**

```
> table(data.ts$class, prediksi.mdlp)
   prediksi.mdlp
      0   1
  0 177  50
  1  73 158
> mean(data.ts$class == prediksi.mdlp)
[1] 0.731441
```

Catatan: akurasi dari model tanpa diskretisasi adalah 0.5021834

```
> table(data.ts$class, prediksi.chim)
   prediksi.chim
      0   1
  0 189  38
  1  77 154
> mean(data.ts$class == prediksi.chim)
[1] 0.7489083
```

Catatan: akurasi dari model tanpa diskretisasi adalah 0.5021834

terima kasih

bagusco@apps.ipb.ac.id

IPB University
Bogor Indonesia