



Direktorat Jenderal Pendidikan Tinggi, Riset, dan, Teknologi
Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi
Republik Indonesia



MICROCREDENTIAL: ASSOCIATE DATA SCIENTIST

01 November – 10 Desember 2021

Pertemuan ke-16

Membangun model 7: Evaluasi



[ditjen.dikti](#)



[@ditjendikt](#)



[ditjen.dikti](#)



Ditjen
Diktristek



<https://dikti.kemdikbud.go.id/>



Profil Pengajar: Nama Lengkap dan Gelar Akademik



Poto
Pengajar

Contak Pengajar:

Ponsel:

xxxxxx

Email:

xxxxxxx

Jabatan Akademik:

Latar Belakang Pendidikan:

- S1:
- S2:
- S3:

Riwayat/Pengalaman Pekerjaan:

- Dosen
- XXXX
- XXXX
- XXXX
- XXXX



Deskripsi Topik

KODE UNIT : J.62DMI00.014.1

JUDUL UNIT : Mengevaluasi Hasil Pemodelan

DESKRIPSI UNIT: Unit kompetensi ini berhubungan dengan pengetahuan, keterampilan, dan sikap kerja yang dibutuhkan dalam mengevaluasi hasil pemodelan.

ELEMEN KOMPETENSI	KRITERIA UNJUK KERJA
1. Menggunakan model dengan data riil	1.1 Data baru untuk evaluasi pemodelan dikumpulkan sesuai kebutuhan yang mengacu kepada parameter evaluasi . 1.2 Model diuji dengan menggunakan data riil yang telah dikumpulkan.
2. Menilai hasil pemodelan	2.1 Keluaran pengujian model dinilai berdasarkan metrik kesuksesan . 2.2 Hasil penilaian didokumentasikan sesuai standar yang berlaku.

BATASAN VARIABEL

1. Konteks variabel
 - 1.1 Data baru adalah data yang tidak terlibat dalam pembangunan model.
 - 1.2 Parameter evaluasi contohnya antara lain: akurasi, presisi, *recall*, *f1-score*, kohesi, *Mean Absolute Error* (MAE).
 - 1.3 Metrik kesuksesan adalah metrik yang digunakan sebagai acuan perhitungan untuk menentukan nilai kesuksesan hasil pemodelan
 - 1.4 Hasil penilaian adalah hasil analisis terhadap nilai kesuksesan model yang memberikan informasi pencapaian tujuan bisnis.



Deskripsi Topik

KODE UNIT : **J.62DMI00.015.1**

JUDUL UNIT : **Melakukan Proses Review Pemodelan**

DESKRIPSI UNIT: Unit kompetensi ini berhubungan dengan pengetahuan, keterampilan, dan sikap kerja yang dibutuhkan dalam melakukan proses *review pemodelan*.

ELEMEN KOMPETENSI	KRITERIA UNJUK KERJA
1. Menilai kesesuaian proses pemodelan	1.1 Proses pemodelan diperiksa kesesuaiannya berdasarkan tahapan yang ditentukan. 1.2 Tindak lanjut yang perlu diambil ditentukan berdasarkan hasil pengujian model hasil pemeriksaan kesesuaian.
2. Menilai kualitas proses pemodelan	2.1 Proses pemodelan dinilai berdasarkan rencana pelaksanaan proyek . 2.2 Hasil penilaian didokumentasikan sesuai standard yang berlaku.

BATASAN VARIABEL

1. Konteks variabel
 - 1.1 Tahapan adalah langkah-langkah atau fase-fase yang telah disesuaikan dengan rencana pelaksanaan proyek.
 - 1.2 Tindak lanjut adalah proses yang akan direncanakan atau dilaksanakan sesuai kebutuhan, misalnya ada tahapan yang perlu diulang atau bisa dilanjutkan ke proses berikutnya.
 - 1.3 Rencana pelaksanaan proyek mengandung setiap tahapan yang perludilakukan beserta durasi dan sumber daya yang diperlukan, masukan dan keluaran, serta ketergantungan lainnya. Diusahakan iterasi skala besar dideskripsikan secara eksplisit, misalnya repetisi modeling dan fase evaluasi. Rencana proyek juga perlu mengandung hasil analisis keterhubungan antara jadwal dan resiko, disertai dengan tindakan dan usulan penanganan apabila terjadi suatu resiko adalah metrik perhitungan yang menilai kualitas proses-proses yang telah dilaksanakan selama pemodelan.
- 1.4 Hasil penilaian adalah penilaian yang memberikan informasi penilaian proses pelaksanaan pemodelan dan keputusan terkait langkah selanjutnya.



Model yang diukur untuk:

Supervised Learning

- Klasifikasi
- Regresi

Unsupervised Learning

- Clustering



Pengantar

- Salah satu pekerjaan (task) inti dalam membangun model ML adalah mengevaluasi kinerjanya: sifat Fundamental, namun Tidak mudah/sulit/menantang
- Model ML harus memberikan prediksi akurat dalam rangka menghasilkan luaran yg berguna bagi organisasi
- Ketika model training dianggap sbg salah satu tahapan kunci, kita harus pastikan model prediksi yang dibangun dapat menggeneralisir data uji yang selalu baru dan berubah
- Pertanyaan terkait proyek model AI/DS/ML:
 - Bagaimana cara mengukur suksesnya suatu proyek?
 - Bagaimana kita tahu saat suatu proyek sukses?

Alasan Evaluasi Model

- Ada sejumlah tahapan dalam membangun model:
 - fase pertama adalah prototyping sebagai percobaan atas bbrp model yang berbeda utk mencari model terbaik (model selection)
 - setelah puas dengan model prototype, lanjut ke *deploy* produksi, utk menguji dengan *live* data

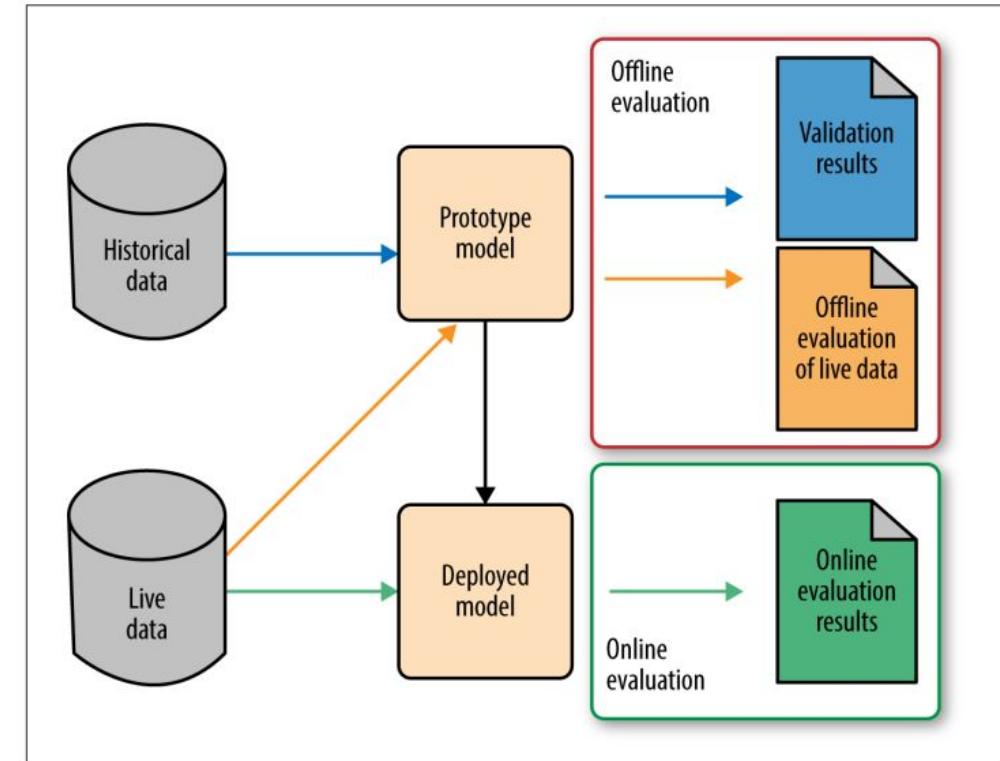


Figure 1-1. Machine learning model development and evaluation workflow

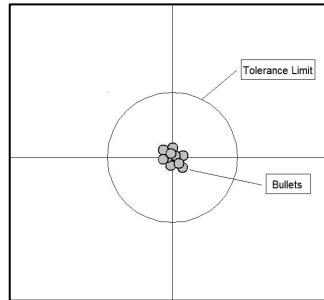


Alasan Evaluasi Model

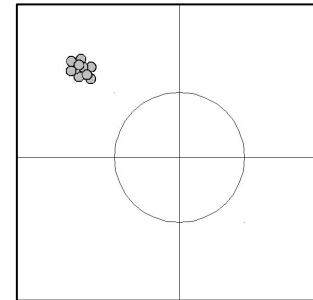
- Terdapat perbedaan antara data historikal vs data *live*, disebut: *distribution drift*
- data historikal, biasa diasumsikan normal
- data live, cenderung berubah
- Deteksi *distribution drift* adalah men-track kinerja model pada metrik validasi di data live
 - jika kinerja dapat dibandingkan dengan hasil validasi di model yg dibangun, maka model masih *fits* dgn data,
 - jika kinerja semakin menurun, kemungkinan distribusi data live menyimpang substantif dari data historikal

Bias dan Variansi

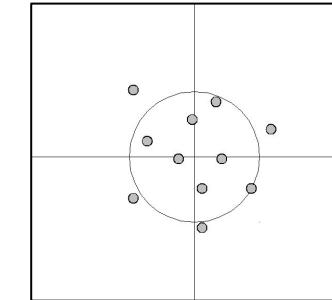
- Bias adalah asumsi penyederhanaan yang dibuat oleh model untuk membuat fungsi target lebih mudah didekati.
- Varians (var.) adalah jumlah estimasi fungsi target yang akan berubah jika diberikan data pelatihan yang berbeda.
- Trade-off adalah ketegangan antara error yang disebabkan oleh bias dan varians, yg menjadi masalah utama dalam supervised learning.



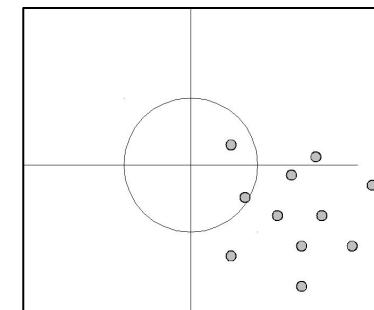
Bias rendah, Var. rendah



Bias tinggi, Var. rendah



Bias rendah, Var. tinggi



Bias tinggi, Var. tinggi



Bias dan Variansi

Berikut adalah bbrp cara utk menangani bias dan variansi:

- Gunakan insting & intuisi
- Bagging (Bootstrap Aggregating) and Resampling, utk mereduksi varians dalam model prediksi, Diantaranya dengan algoritma random forest
- Sifat asimtotik dari algoritma
- Pemahaman lebih dalam atas Bias dan Varians

$$\frac{dBias}{dComplexity} = -\frac{dVariance}{dComplexity}$$



Jenis error



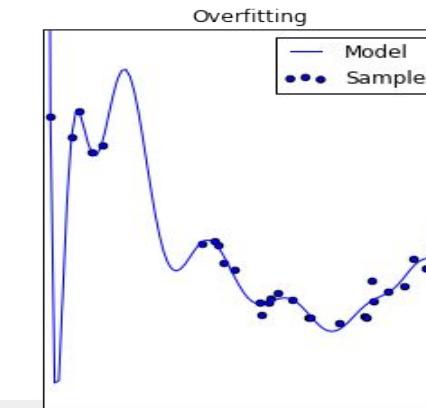
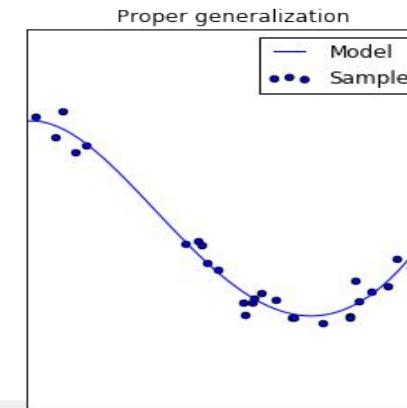
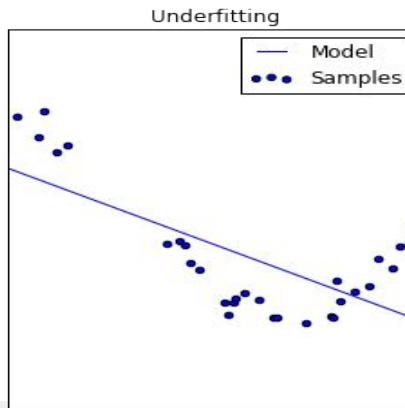
Underfitting:
oversimplifikasi/
menyepelekan



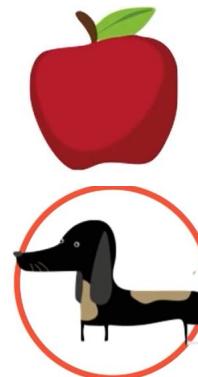
Overfitting:
melebih-lebihkan
permasalahan

Bias dan Variansi

- Underfitting: Model dengan bias tinggi kurang memperhatikan data yang disajikan
 - keadaan dimana model pelatihan data yang dibuat tidak mewakilkan keseluruhan data yang akan digunakan nantinya
- Overfitting terjadi karena model yang dibuat terlalu fokus pada training dataset tertentu, hingga tidak bisa melakukan prediksi dengan tepat jika diberikan dataset lain yang serupa.
 - memiliki low loss dan akurasi rendah.



Ilustrasi Pemilihan Model (klasifikasi)



VS

Bukan ANJING

ANJING

Bukan HEWAN

HEWAN

VS

apapun kecuali anjing
yg ekornya bergoyang

anjing dgn ekor
bergoyang

terlalu spesifik



terlalu sederhana



Ilustrasi Trade-off Bias vs Varians dlm Pemilihan Model (klasifikasi)

Tradeoff

High bias
(Underfitting)

Not animals

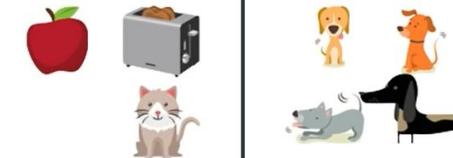


Animals



Just Right

Not dogs



Dogs

High variance
(Overfitting)

Not dogs who wag
their tail



Dogs who w
their tail



Bad on Training set

Bad on Testing set

Good on Training set

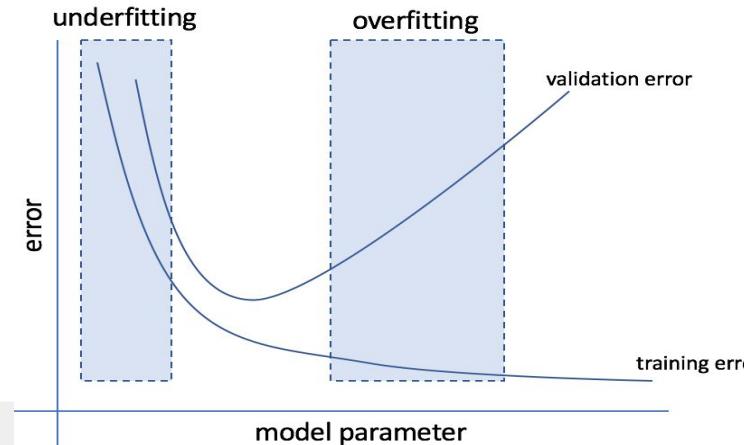
Good on Testing set

Great on Training set

Bad on Testing set

Kurva Validasi

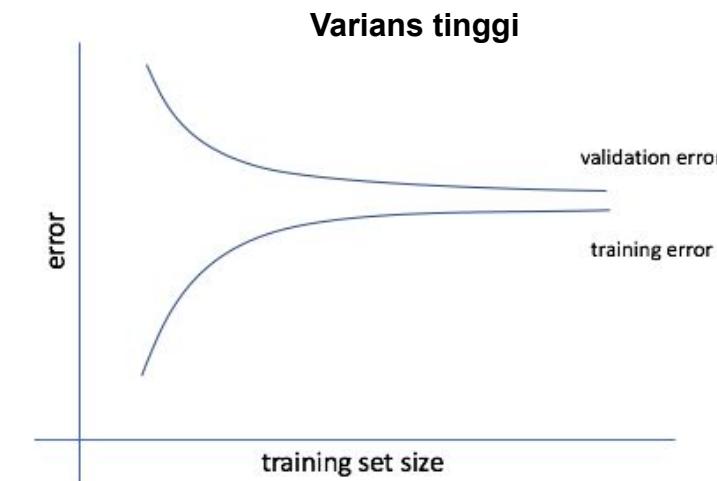
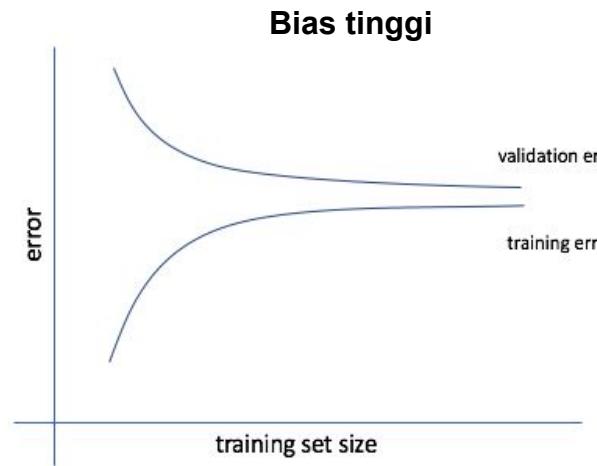
- Kurva validasi (silang) memungkinkan kita menemukan posisi terbaik (*sweet spot*) antara underfitting dan overfitting model untuk membangun model yang dapat digeneralisasi dengan baik.
- Kurva validasi tipikal adalah plot error model sebagai fungsi dari beberapa hyperparameter model yang mengontrol kecenderungan model untuk terlalu cocok atau tidak sesuai dengan data.
- Parameter yang dipilih bergantung pada model spesifik yang dievaluasi



<https://www.jeremyjordan.me/evaluating-a-machine-learning-model/>

Kurva Pembelajaran (Learning)

- Cara lain utk mendiagnosis bias dan varians dalam model
- memplot kesalahan model sebagai fungsi dari jumlah contoh training.
- Mirip dengan kurva validasi, kita akan memplot kesalahan untuk data pelatihan dan data validasi.





Model yang diukur untuk:

Supervised Learning

- Klasifikasi
- Regresi

Unsupervised Learning

- Clustering

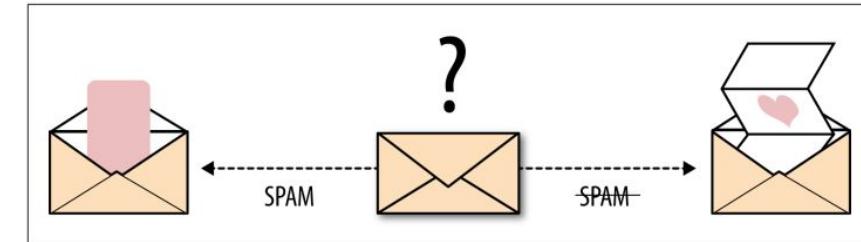


Figure 2-1. Email spam detection is a binary classification problem
(source: Mark Enomoto | Dato Design)



Akurasi

Menunjukkan persentase klasifikasi yang bernilai valid terhadap total klasifikasi yang dilakukan.

$$\text{Akurasi}(a) = \frac{\text{Prediksi Valid } (t)}{n} \times 100\%$$

dengan

a adalah akurasi dalam persen,

t adalah jumlah percobaan dengan prediksi valid, dan

n adalah jumlah percobaan

Contoh: Berapa akurasi
dari percobaan di samping ini?

Data aktual	Output model (prediksi)	Kesimpulan
mangga	mangga	valid
jeruk	apel	invalid
apel	jeruk	invalid
mangga	apel	invalid
jeruk	jeruk	valid



Contoh

Data aktual	Output model (prediksi)	Kesimpulan
mangga	mangga	valid
jeruk	apel	invalid
apel	jeruk	invalid
mangga	apel	invalid
jeruk	jeruk	valid

Jumlah percobaan valid (t) = 2

Jumlah percobaan invalid = 3

Total Percobaan (a) = 5

$$a = \frac{t}{n} \times 100\%$$

$$a = \frac{2}{5} \times 100\%$$

$$a = 40\%$$

Akurasi dapat digunakan sebagai ukuran awal mengevaluasi model, namun tidak cukup dengan akurasi saja. Terkadang akurasi tiap kelas perlu diketahui juga.



Confusion matrix

- Bukan metric, namun bermanfaat melihat **sebaran validitas percobaan**. atau untuk mengkuantifikasi biaya karena terjadinya kesalahan!
- Berguna utk memahami perbedaan antar kelas (**klasifikasi**)

		Kelas Prediksi			
		mangga	apel	jambu	pear
Kelas Aktual	mangga	19	3	2	1
	apel	1	22	1	1
	Jambu	2	2	21	0
	Pear	0	1	1	23

Karakteristik:

- Ada sumbu data aktual dan sumbu data prediksi (gunakan konvensi)
- Setiap kelas terpetakan satu sama lainnya
- Percobaan valid berada pada diagonal utama
- Matriks berbentuk bujur sangkar



Confusion matrix

		Kelas Prediksi			
		mangga	apel	jambu	pear
Kelas Aktual	mangga	19	3	2	1
	apel	1	22	1	1
	Jambu	2	2	21	0
	Pear	0	1	1	23

Pada area kotak merah:

terdapat 25 mangga yang di uji, dengan 19 mangga dikenali sebagai mangga (valid), 3 mangga dikenali sebagai apel (invalid), 2 mangga dikenali sebagai jambu (invalid), dan 1 mangga dikenali sebagai pear (invalid)



Confusion matrix

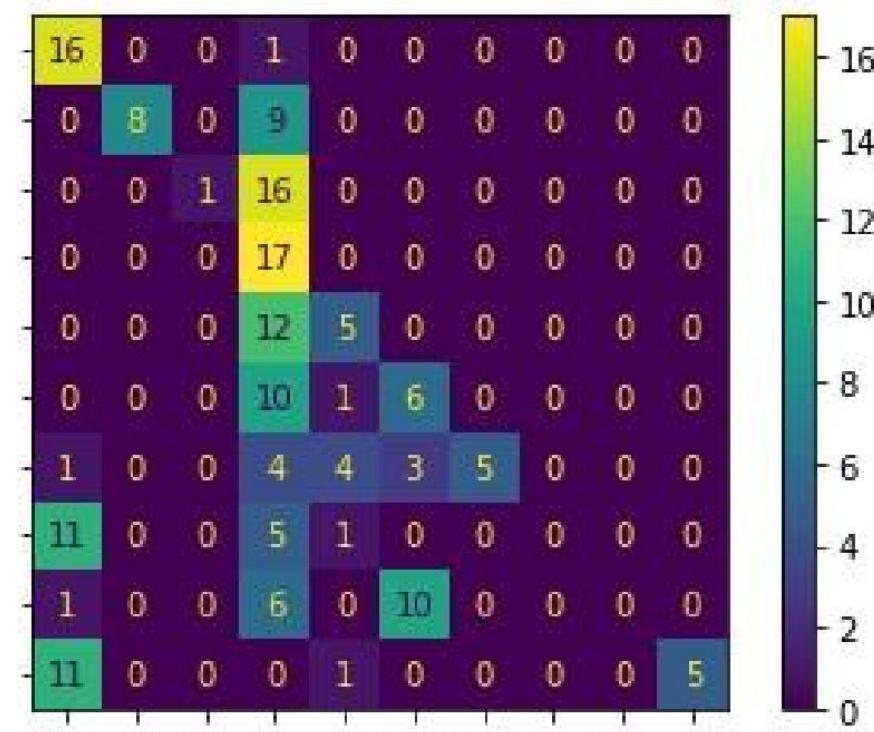
		Kelas Prediksi			
		mangga	apel	jambu	pear
Kelas Aktual	mangga	19	3	2	1
	apel	1	22	1	1
	Jambu	2	2	21	0
	Pear	0	1	1	23

Akurasi untuk pengujian kelas mangga adalah : $19/25 \times 100\% = 76\%$

Sedangkan akurasi total pengujian adalah : $(19+22+21+23)/(25+25+25+25) \times 100\% = 85\%$

Visualisasi confusion matrix

- Representasi dengan heat map lebih baik.
- Contoh di samping adalah confusion matrix pada suatu percobaan di epoch 1.
- Berapa jumlah kelasnya?
- Jumlah data uji tiap kelas?



Pertanyaan Umum: Metrik Terbaik ?

Jawab: Tergantung pada tujuan bisnis. lihat contoh ilustrasi brkt

- Filter Spam (kelas positif: spam). Optimasi pada presisi atau spesifikasi dimana lebih baik false negatif (spam masuk inbox) drpd false positif (non-spam masuk junk/folder spam)
- Detektor Transaksi Fraud (kelas positif: fraud). Optimasi pada sensitivitas krn false positif (transaksi normal ditandai fraud) lebih dapat diterima drpd false negatif (transaksi fraud tidak terdeteksi)

	Sent to Spam Folder	Sent to Inbox
Spam		
Not Spam		
False Positives		



Binary Classification

- Hanya ada kelas: 0 atau 1, valid atau invalid, true atau false, positif atau negatif, bagus atau tidak bagus, cantik atau tidak cantik, recommended atau tidak, lulus atau tidak, spam atau bukan spam, hoax atau fakta, dsb.
- Bentuk yang paling umum: satu kelas dinyatakan sebagai kelas **positif** (menjadi fokus dalam klasifikasi), dan satu kelas lainnya dinyatakan sebagai kelas **negatif**

Contoh dalam dunia medis : sampel cairan mukus yang mengandung virus Covid-19 dinyatakan sebagai kelas positif dan sampel yang tidak mengandung virus dinyatakan sebagai kelas negatif

Contoh dalam kebencanaan : gempa yang mengakibatkan tsunami sebagai kelas positif dan yang tidak mengakibatkan tsunami sebagai kelas negatif



Keterbatasan akurasi

Dalam kasus deteksi pasien positif Covid, sebuah detektor baru, sebut saja detektor X, diujikan pada 100 sampel. Sampel tersebut telah diuji dengan alat yang hasil deteksinya dijadikan acuan validitas (ground truth), yaitu PCR. Dari pengujian PCR sebelumnya diperoleh data bahwa 90 sampel adalah negatif dan 10 sampel adalah positif. Dengan menggunakan detektor X, ke 90 sampel negatif dideteksi negatif. Namun pada 10 sampel positif, diperoleh hasil bawah 5 sample dinyatakan sebagai positif dan 5 sampel sisanya negatif.

Berapa akurasi detektor X? $(90+5)/(90+10) \times 100\% = 95\%$!

Apakah akurasi 95% merupakan hasil yang baik?

Kesalahan detektor X yang hanya pada 5% dapat berakibat fatal. Apakah ada metrik lain yang menjelaskan kasus semacam ini?



Confusion Matrix untuk Klasifikasi Biner

		Nilai Prediksi	
		Positive	Negative
Nilai Aktual	Positive	True Positive	False Negative
	Negative	False Positive	True Negative

True Positive (TP): nilai sesungguhnya adalah positif dan diprediksi positif

False Positive (FP): nilai sesungguhnya adalah negatif namun diprediksi positif

True Negative (TN): nilai sesungguhnya adalah negatif dan diprediksi negatif, dan

False Negative (FN): nilai sesungguhnya adalah positif namun diprediksi negatif.

Klasifikasi yang bernilai **valid** adalah **TP** dan **TN**

Confusion Matrix untuk Klasifikasi Biner (Ilustrasi)

Prediction





Metrik pada Klasifikasi Biner

		Nilai Prediksi		
		Positive	Negative	
Nilai Aktual	Positive	True Positive (TP)	False Negative (FN)	Recall, Sensitivity, True Positive Rate $\frac{TP}{TP + FN}$
	Negative	False Positive(FP)	True Negative (TN)	Specificity, True Negative Rate $\frac{TN}{FP + TN}$ False Positive Rate $\frac{FP}{FP + TN}$
		Precision $\frac{TP}{TP + FP}$	Negative Predictive Value $\frac{TN}{TN + FN}$	Accuracy $\frac{TP + TN}{TP + TN + FP + FN}$



Interpretasi metrik Recall - Precision

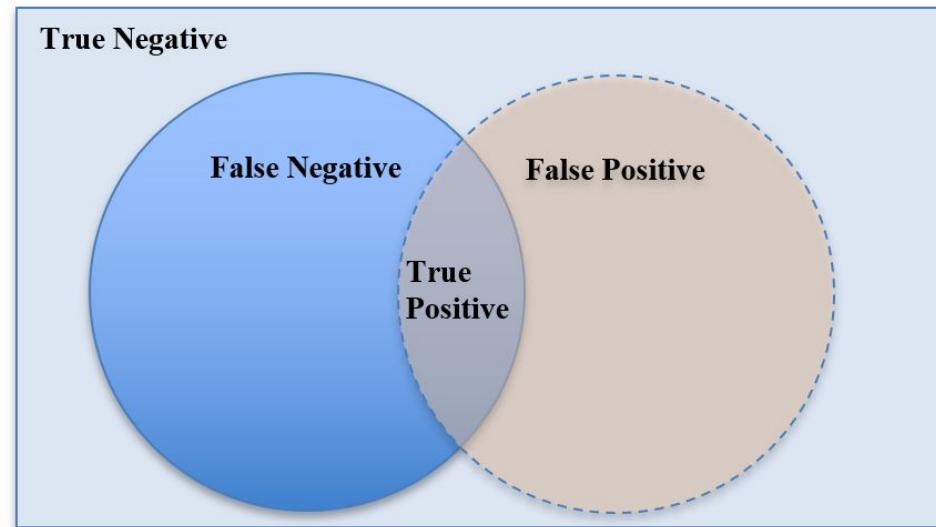
Recall dan *Precision* sering dihitung bersamaan untuk menggambarkan performansi model.

Kombinasi yang mungkin untuk keduanya:

- *Low recall low precision*
- *High recall low precision*
- *Low recall high precision*
- *High recall high precision*

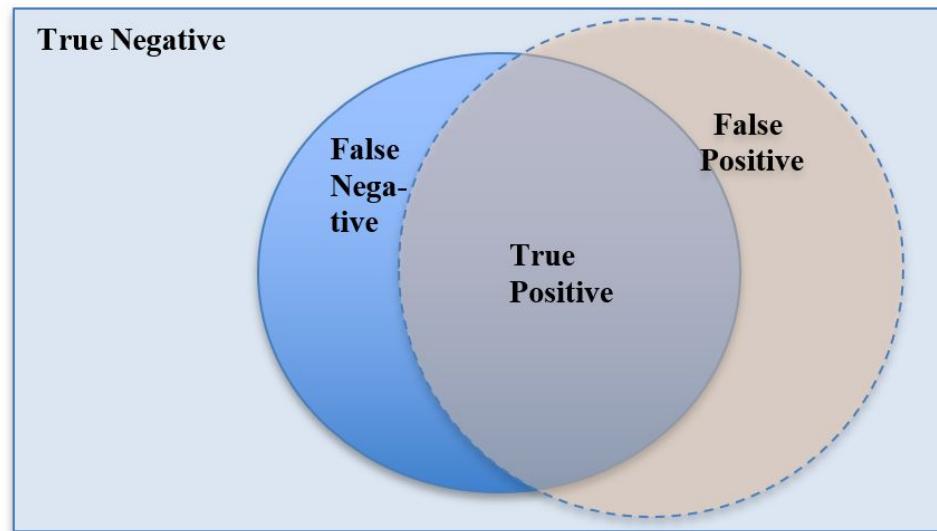
Low recall low precision

- Model berkinerja kurang baik
- Baik False negatif dan maupun false positif bernilai besar pada model ini



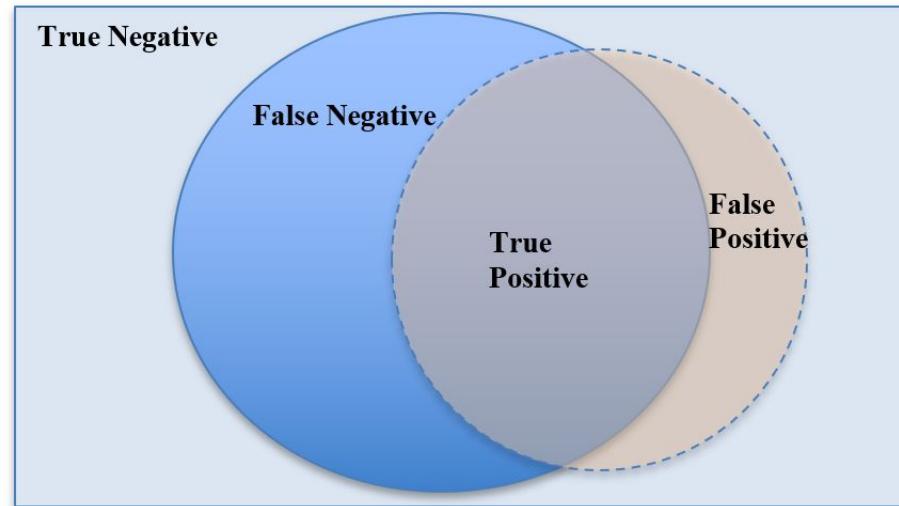
High recall low precision

- *Sebagian besar data positif dapat dikenali dengan baik (False Negative rendah)*
- *Tetapi banyak data negatif dikenali sebagai positif (False Positive tinggi)*





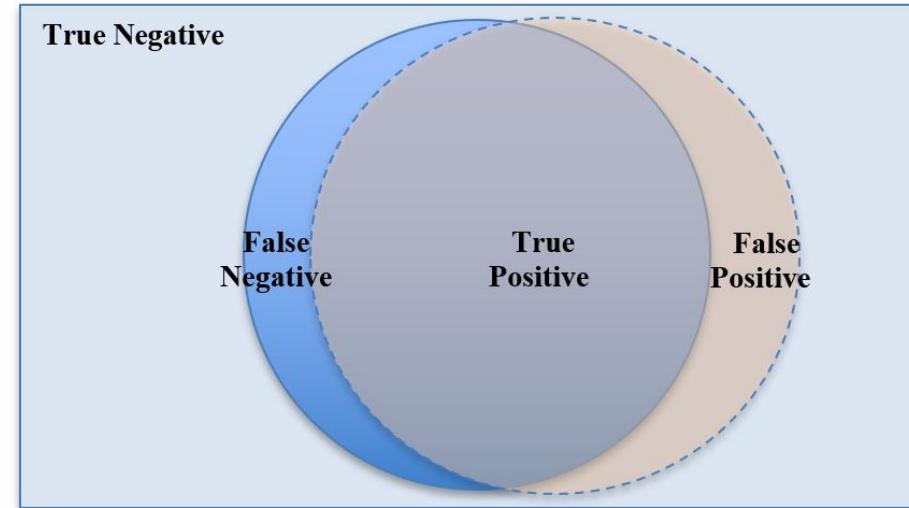
Low recall high precision



- *Banyak data positif yang teridentifikasi negatif (False Negative besar)*
- *Sebagian besar data yang teridentifikasi positif memang benar positif*



High recall high precision



Model memiliki kinerja baik

False Positive maupun False Negative rendah

True Positive dan True Negative tinggi



F Score

- Mengukur keseimbangan antara *Precision* – *Recall*
- Untuk model yang *Precision* dan *Recall* sama pentingnya digunakan *F-1 Score*, dinyatakan:

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = \frac{TP}{TP + \frac{1}{2}(FP + FN)}$$

- Untuk kasus *recall* lebih diutamakan dengan faktor β maka formula diperluas menjadi:

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{(\beta^2 \cdot \text{precision}) + \text{recall}}$$



F Score

Harmonic mean

$$\text{Arithmetic Mean} = \frac{x+y}{2}$$
$$\text{Harmonic Mean} = \frac{2xy}{x+y}$$

Precision = 1
Recall = 0
Average = 0.5
Harmonic Mean = 0

Contoh
Perhitungan

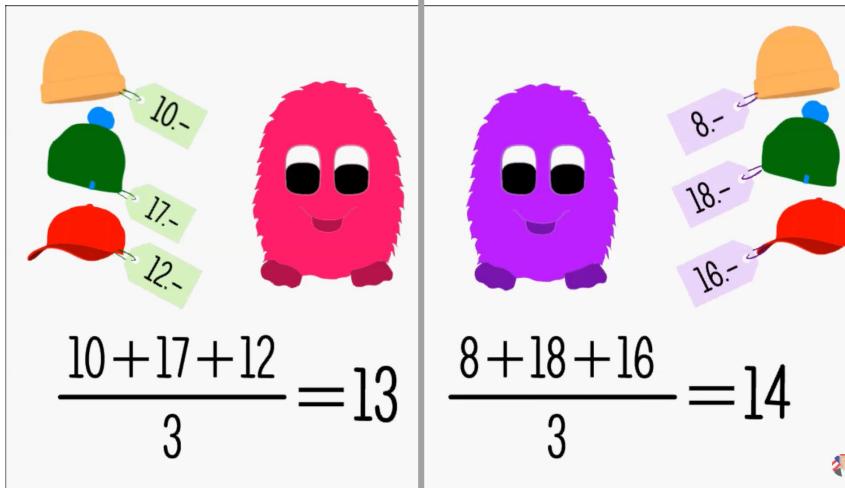
Precision = 0.2
Recall = 0.8
Average = 0.5
Harmonic Mean = 0.32

<https://www.youtube.com/watch?v=aDW44NPhNw0>

~~Arithmetic Mean(Precision, Recall)~~
F1 Score = Harmonic Mean(Precision, Recall)

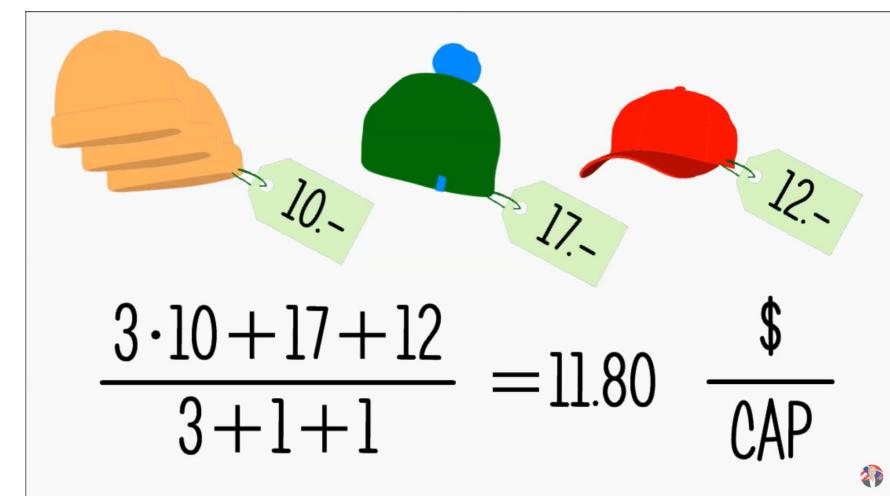
F Score: Harmonic Mean vs Aritmetik Mean

- *Aritmatik Mean*



jika beli 3 topi, harga
= $3 \times 13 = 39$
jika beli 5 topi, harga
= $5 \times 13 = 65$

- *Harmonic Mean*



jika beli 3 topi, harga
= $3 \times 11.8 = 35.4$
jika beli 5 topi, harga
= $5 \times 11.8 = 59$



F Score

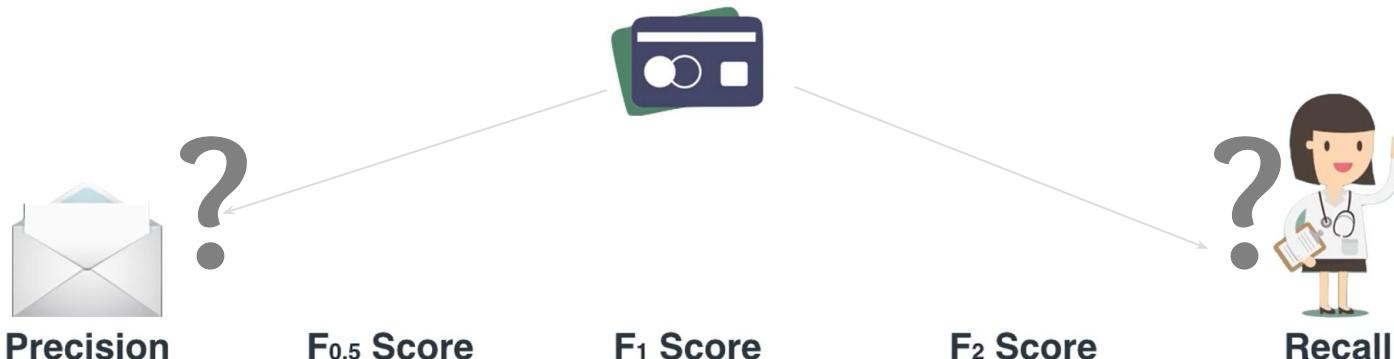
- *Contoh:*

Jika kita ingin mendeteksi transaksi fraud, metrik apa yang lebih optimal?:

- apakah cenderung pada Presisi, atau
- apakah cenderung pada Recall

F_{β} Score

<https://www.youtube.com/watch?v=aDW44NPhNw0>





F Score

- *Contoh:*

Jika kita ingin mendeteksi transaksi fraud, metrik apa yang lebih optimal?:

- cenderung pada Recall, karena lebih baik kita mendeteksi banyak transaksi dianggap fraud (kita dapat melakukan konfirmasi) daripada ada fraud yg lolos deteksi alias kebobolan!

F_{β} Score

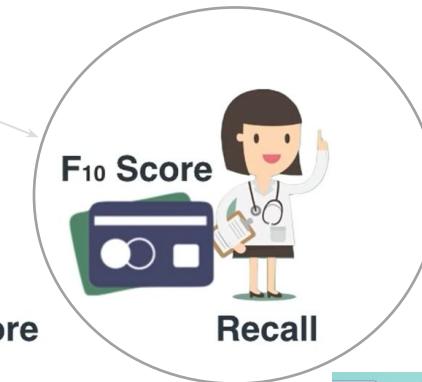


Precision

$F_{0.5}$ Score

F_1 Score

F_2 Score





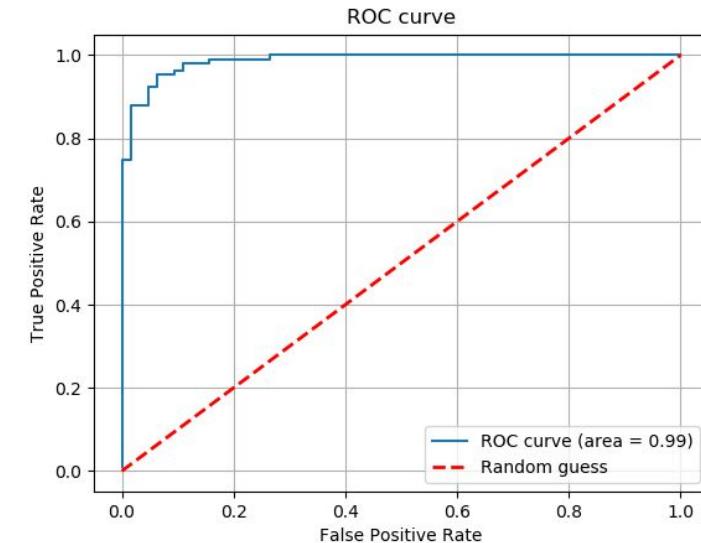
Evaluasi Model pada Probabilistik Model

- Model dalam memprediksi dengan menghasilkan nilai probabilitas $[0,1]$ untuk suatu label
- Nilai Probabilitas menunjukkan seberapa yakin terhadap suatu kelas/label
- Label ditentukan dengan menetapkan suatu threshold
- Evaluasi dapat dilakukan beberapa metode, diantaranya:
 - a. Receiver Operator Characteristic (ROC) (dan pengembangan ke PR-AUC)
 - b. Logarithmic loss function

<https://www.analyticsvidhya.com/blog/2020/10/quick-guide-to-evaluation-metrics-for-supervised-and-unsupervised-machine-learning/>

Evaluasi Model pada Probabilistik Model

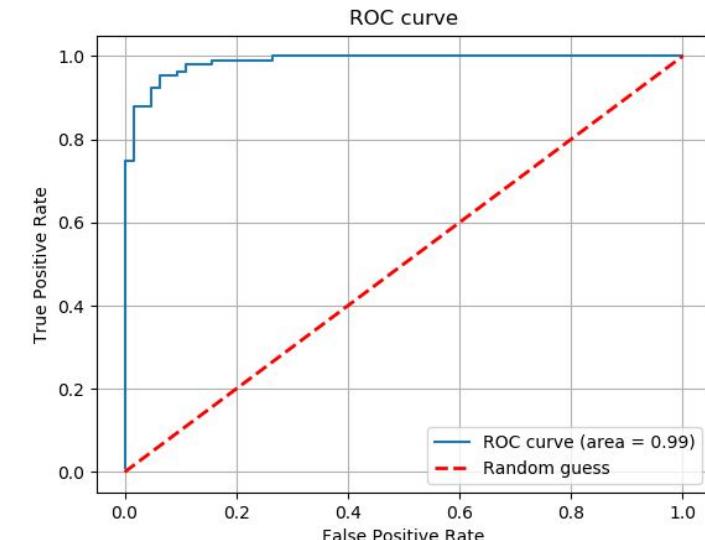
- Evaluasi dapat dilakukan metode Receiver Operating Characteristic (ROC) yaitu dengan mem-plot antara **Recall** (atau disebut juga **True Positive Rate**) sebagai sumbu -y dengan **False Positive Rate** sebagai sumbu-x untuk setiap threshold klasifikasi yang mungkin (threshold antara 0 hingga 1)
- Area yang diperoleh dari ROC dapat digunakan untuk analisis ROC – AUC (AUC: Area Under Curve),



<https://www.analyticsvidhya.com/blog/2020/10/quick-guide-to-evaluation-metrics-for-supervised-and-unsupervised-machine-learning/>

ROC vs Akurasi

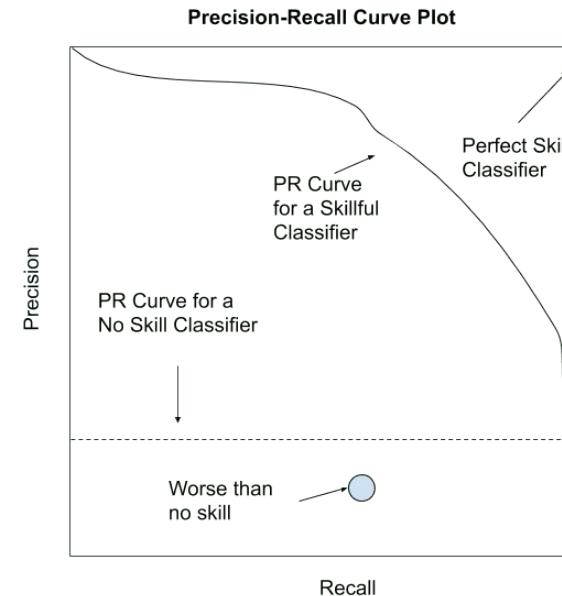
- Mengapa perlu ROC-AUC dan tidak cukup menggunakan akurasi?
- ROC-AUC lebih menggambarkan secara lengkap visualisasi untuk semua threshold klasifikasi yang mungkin (dari confusion matrix yg terbentuk dari perubahan threshold)
- Akurasi hanya merepresentasikan performansi pada satu nilai threshold
- **Diskusi:** apa arti garis putus berwarna merah (random guess) pada diagram ROC-AUC ?
- Hasil paling optimal adalah yang menghasilkan AUC paling luas



<https://www.analyticsvidhya.com/blog/2020/10/quick-guide-to-evaluation-metrics-for-supervised-and-unsupervised-machine-learning/>

Alternatif lain : PR - ROC

- Plotting antara Precision (sumbu -y) dan Recall (sumbu -x)
- Digunakan untuk kelas yang minoritas yang menjadi perhatian utama cukup kecil



<https://machinelearningmastery.com/tour-of-evaluation-metrics-for-imbalanced-classification/>



Logarithmic loss (log loss)

- Menunjukkan seberapa yakin pemberian label terhadap data yang diuji/diobservasi
- Untuk setiap sample, perlu dihitung probabilitas untuk semua label yang mungkin
- Nilai log loss berada di $[0, \infty)$ dan dinyatakan sebagai

$$\text{LogarithmicLoss} = \frac{-1}{N} \sum_{i=1}^N \sum_{j=1}^M y_{ij} * \log(p_{ij})$$

- N: jumlah sampel ,
- M : jumlah class ,
- y_{ij} menunjukkan apakah sample i adalah kelas j atau bukan
- p_{ij} menunjukkan probability sample i adalah kelas j
- Diskusi:** semakin baik prediksi, nilai log loss semakin besar atau semakin kecil?



Penerapan evaluasi model : Medis

Evaluasi model sangat bergantung kepada kasus dan distribusi data. Contoh model untuk medis:

- deteksi penyakit menular (misalnya alat detektor G-Nose untuk penyakit Covid-19)
- deteksi kanker (misalnya kanker ganas / jinak pada kanker payudara , kanker serviks, dsb)
- deteksi kehamilan
- deteksi kekurangan gizi

Ingin kembali detektor X untuk kasus Covid-19.

- Dari parameter evaluasi : TP, TN, FP, FN, parameter manakah yang perlu untuk ditekan?
- Dari parameter FP dan FN parameter mana yang paling krusial, dimana kesalahannya berakibat fatal bagi manusia?



Penerapan evaluasi model: Kebencanaaan

Contoh model untuk kebencanaan:

- *Deteksi gunung meletus*
- *Deteksi kemarau panjang*
- *Deteksi banjir*
- *Deteksi gempa bumi*
- *Deteksi tsunami*



Penerapan evaluasi model: Telekomunikasi

Contoh model telekomunikasi:

- *Deteksi spam (dipakai sebagai spam filter)*
- *Deteksi hoax*
- *Deteksi fraud*
- *Deteksi pembajakan akun*



False Negative vs. False Positive?

- Kesalahan prediksi berada di False Positive maupun false Negative.

Pada Kebencanaan, misalnya deteksi dini tsunami di tepi pantai (tsunami : +):

- **False Negative** : Diprediksi tidak ada tsunami, namun ternyata ada tsunami
- **False Positive** : Diprediksi ada tsunami, namun ternyata tidak tsunami

Pada case ini, Tsunami yang tidak terprediksi sebelumnya (FN) lebih membahayakan dan sangat merugikan, dibandingkan FP

Deteksi spam (dipakai sebagai spam filter, spam : +)

- **False Negative** : Email Spam masuk inbox
- **False positive** : Email normal masuk folder spam

Pada kasus ini, umumnya orang masih bisa menerima spam masuk inbox (FN) daripada email penting masuk spam (FP), bisa berakibat gagal kerja, gagal proyek, gagal sekolah, dsb.

Setiap permasalahan memiliki titik tekan parameter yang berbeda beda tergantung kasusnya!



Beberapa kasus aplikasi dengan Class Imbalance Problem

Isu dataset yang tidak seimbang muncul dalam berbagai persoalan dan menjadi bahasan tersendiri

Berikut contoh aplikasi dengan data kelas tidak imbang (sumber: Learning from Imbalanced Data Sets, Alberto F., et.al. Springer, 2018)

Applications of ML and DM where the class imbalance problem is present

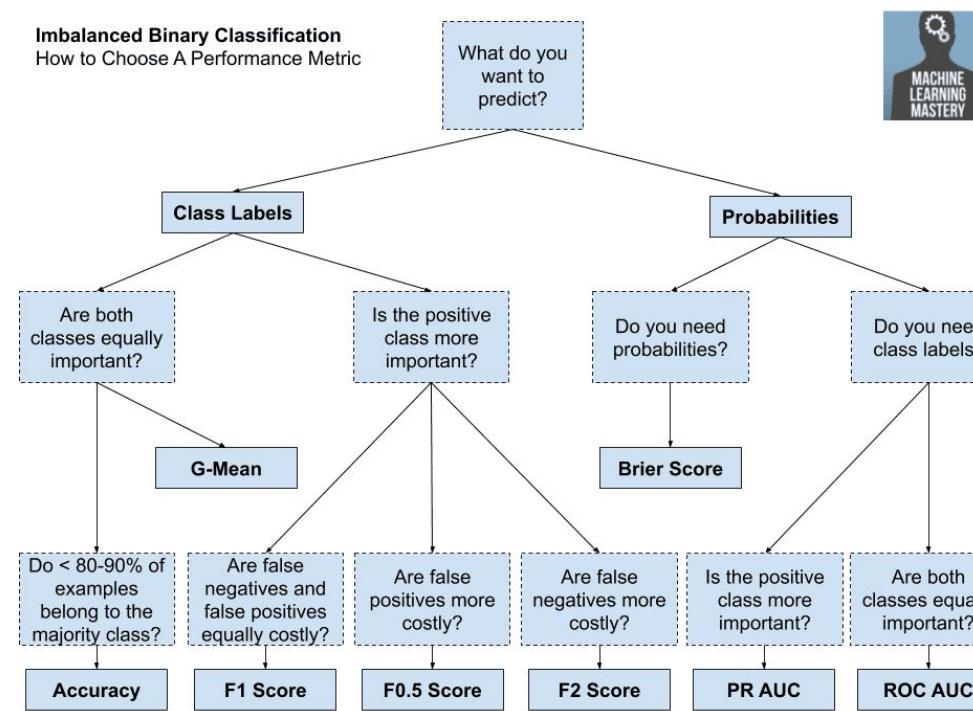
Year	Domain	Subcategory	Application	Data-level	Internal	Cost-sensitive	Ensemble
1997	Engineering	Satellite radar images	Detection of oil spills in satellite radar images		x		
1997	Engineering	Satellite radar images	Detection of oil spills in satellite radar images	x			
1998	Engineering	Satellite radar images	Detection of oil spills in satellite radar images	x	x		
2012	Information technology	Software	Software defect prediction	x		x	



Year	Domain	Subcategory	Application	Data-level	Internal	Cost-sensitive	Ensemble
2013	Bioinformatics	Protein identification	MicroRNA precursor classification	×		×	
2014	Medicine	Quality control	Prediction of the post-operative life expectancy in lung cancer patients		×	×	
2014	Bioinformatics	Protein identification	Five datasets that represent four different bioinformatics applications. These include miRNA identification, protein localization prediction, promoter identification from DNA sequences, kinase substrate prediction from protein phosphorylation profiling.	×			
2014	Information technology	Text mining	Text categorization	×		×	
2014	Bioinformatics	Cell recognition	Mitotic cells recognition in Hep-2 images	×		×	
2014	Medicine	Diagnosis	Lung nodule detection	×		×	
2014	Information technology	Software	Software defect prediction	×	×	×	
2014	Security	Video surveillance	Face re-identification	×		×	
2014	Information technology	Network analysis	Botnet traffic detection	×	×	×	
2014	Information technology	Network analysis	Network traffic classification			×	

2015	Medicine	Quality control	Prediction of long stay patients in emergency department			×
2015	Bioinformatics	Protein identification	Protein data classification			×
2015	Medicine	Diagnosis	Diagnosis of diabetes mellitus	×		
2016	Business management	Customer relationship management	Customer churn prediction	×		
2016	Medicine	Diagnosis	Breast cancer malignancy classification			×
2016	Medicine	Diagnosis	Bleeding detection in endoscopic video	×		×
2016	Education	High school	Early dropout detection	×	×	
2016	Security	Video surveillance	Face re-identification	×	×	×
2016	Engineering	Semiconductors	Fault detection in semiconductors	×	×	×
2016	Medicine	Diagnosis	Thyroid nodule classification	×		
2016	Medicine	Diagnosis	Breast cancer classification from Magnetic Resonance Images (MRIs)			×
2016	Security	Biometric authentication	Multimodal biometric authentication			×
2017	Engineering	Energy	Short-term voltage stability assessment	×	×	×
2017	Business management	Customer relationship management	Customer churn prediction	×	×	×
2017	Information technology	Network analysis	Mobile malware detection	×	×	
2017	Engineering	Semiconductors	Fault detection in semiconductors	×		×
2017	Medicine	Quality control	Prediction of the survival status of poly-trauma	×		

Alternatif Pilihan Metrik untuk Imbalanced Data Sets



© 2019 MachineLearningMastery.com All Rights Reserved.

<https://machinelearningmastery.com/tour-of-evaluation-metrics-for-imbalanced-classification/>

Pengayaan: Evaluasi Model untuk Segmentasi Citra

Untuk operasi klasifikasi pixel citra (untuk keperluan segmentasi), dimana citra akan dilabeli sebagai foreground dan background, maka kualitas segmentasi dapat diukur dengan beberapa metode diantaranya Jaccard Index dan Similarity Index.

Misalnya A ada adalah hasil segmentasi dan B adalah ground truth, maka

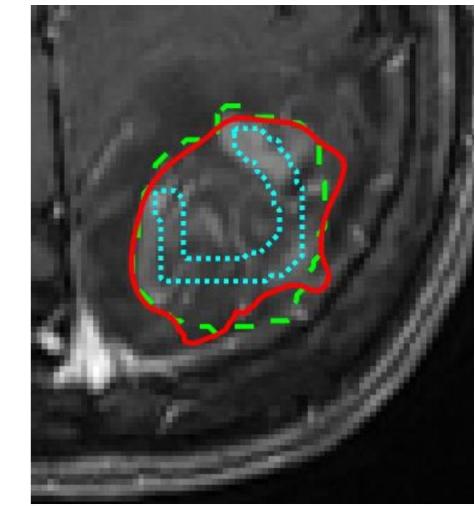
- *Jaccard Index (JI)*

$$JI = (A \cap B) / (A \cup B)$$

- *Similarity Index / Dice Coefficient (SI)*

$$SI = (2|A \cap B|) / (|A| + |B|)$$

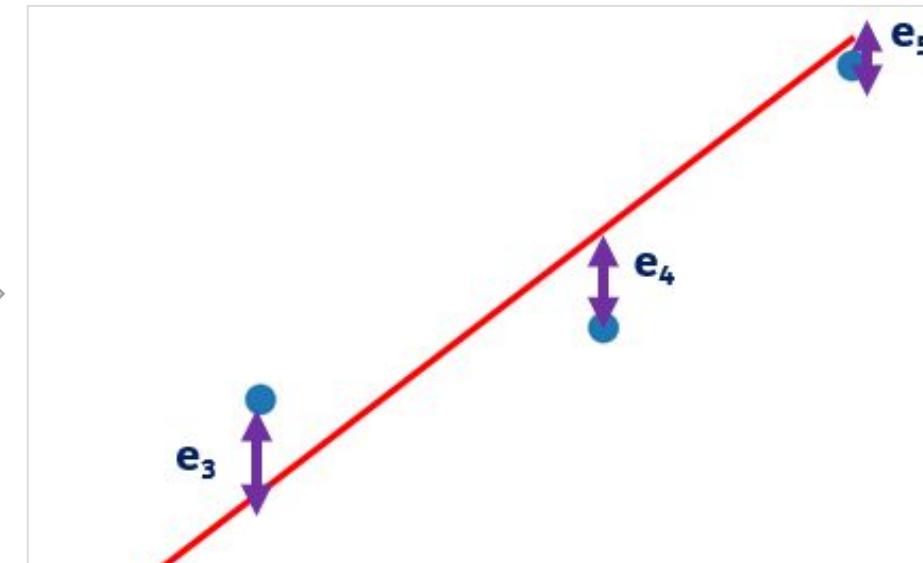
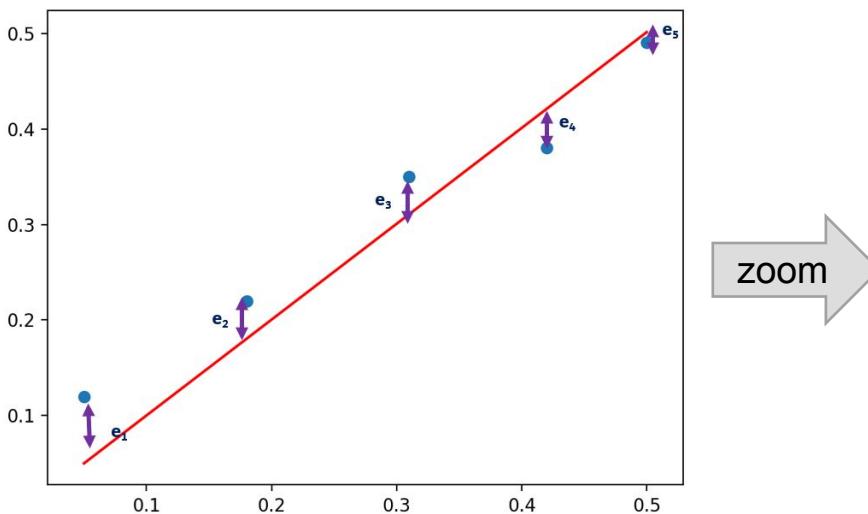
dimana $|.|$ menyatakan banyaknya elemen (dalam hal ini pixel)



Biru : inisiasi, Merah : Hasil segemntasi,
Hijau : Groundtruth

Evaluasi Model untuk Regresi

- Model memprediksi suatu nilai kontinyu (bilangan real), bukan nilai diskrit (berupa kelas/label)
- Contoh: prediksi harga rumah, suhu maksimum, kekuatan gempa, harga saham
- Error merupakan selisih dari nilai aktual dengan nilai prediksi (real)





Evaluasi Model untuk Regresi

Contoh pengukuran model untuk **regresi**:

- a. Mean Absolute Error (MAE)
- b. Relative Absolute Error (RAE)
- c. Mean Squared Error (MSE)
- d. Relative Squared Error (RSE)
- e. Root Mean Squared Error (RMSE)
- f. Mean Absolute Percentage Error (MAPE)
- g. Mean Percentage Error (MPE)
- h. R-squared

MAE dan RMSE akan diulas di slide berikut



Mean Absolute Error (MAE)

- Ide : Setiap selisih error diambil nilai mutlaknya untuk selanjutnya dijumlahkan (*Diskusi*: mengapa nilai mutlak?)
- Jumlah nilai mutlak semua error di bagi rata dengan banyaknya sampel sehingga diperoleh nilai rata rata error, karenanya disebut Mean Absolute Error, dinyatakan sebagai:

$$\text{MAE} = \frac{1}{n} \sum_{j=1}^n |y_j - \hat{y}_j|$$

- n : banyak sample
- y_j : nilai aktual untuk sample j
- \hat{y}_j nilai prediksi untuk sample j



Root Mean Squared Error (RMSE)

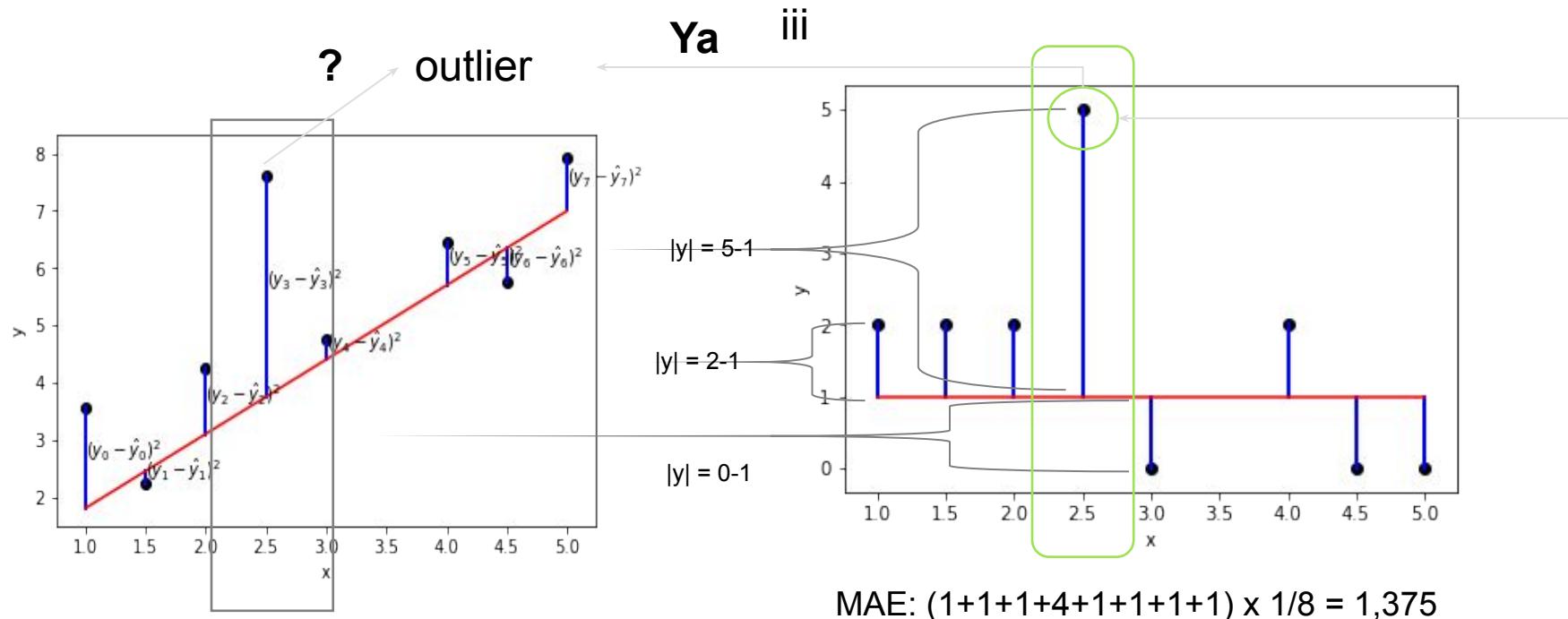
- Ide : Setiap selisih error diambil nilai kuadratnya untuk selanjutnya dijumlahkan
- Jumlah nilai kuadrat setiap error di bagi rata dengan banyaknya sampel sehingga diperoleh nilai rata rata kuadrat error, untuk kemudian ditarik nilai akarnya, karenanya disebut Root Mean Squared Error, dinyatakan sebagai:

- n : banyak sample
- y_j : nilai aktual untuk sample j
- \hat{y}_j nilai prediksi untuk sample j

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{j=1}^n (y_j - \hat{y}_j)^2}$$

- RMSE memiliki fungsi kuadratik yang bersifat kontinu dan dapat diturunkan (differentiable) dan menguntungkan untuk optimasi (**Diskusi: Mengapa?**).
- Lebih sensitive terhadap pencilan (outlier), **Diskusi: Mengapa?**

Root Mean Squared Error (RMSE)



$$MAE: (1+1+1+4+1+1+1+1) \times 1/8 = 1,375$$

$$RMSE: \sqrt{(1^2+1^2+1^2+4^2+1^2+1^2+1^2+1^2) \times 1/8} = 1,70$$



Perbandingan secara umum

Acronym	Full Name	Residual Operation?	Robust To Outliers?
MAE	Mean Absolute Error	Absolute Value	Yes
MSE	Mean Squared Error	Square	No
RMSE	Root Mean Squared Error	Square	No
MAPE	Mean Absolute Percentage Error	Absolute Value	Yes
MPE	Mean Percentage Error	N/A	Yes



Pengukuran Performansi pada Clustering

- Termasuk *unsupervised learning*, data tidak berlabel (tidak ada kelas)
- Tujuan : mengelompokkan data yang mirip sedekat mungkin dan memisahkan data yang tidak mirip sejauh mungkin
- Contoh pengukuran performansi untuk clustering :
 - *Silhouette Coefficient*
 - *Rand Index*
 - *Mutual Information*
 - *Calinski-Harabasz Index (C-H Index)*
 - *Davies-Bouldin Index*
 - *Dunn Index*



Silhouette Coefficient

- *Silhouette Coefficient dinyatakan sebagai*

$$s = \frac{b - a}{\max(a, b)}$$

s: silhouette Coefficient

a: rata-rata jarak sebuah sampel dengan sampel lainnya di cluster yang sama

b: rata-rata jarak sebuah sampel dengan sampel lainnya di cluster tetangga terdekat

- *Nilai berada diantara -1 dan +1*
- *Nilai -1 mengindikasikan clustering yang tidak tepat, di sekitar 0 mengindikasikan adanya overlapping clustering, dan +1 mengindikasikan clustering yang padat dan terpisah dengan baik*



Silhouette Coefficient

Silhouette Coefficient

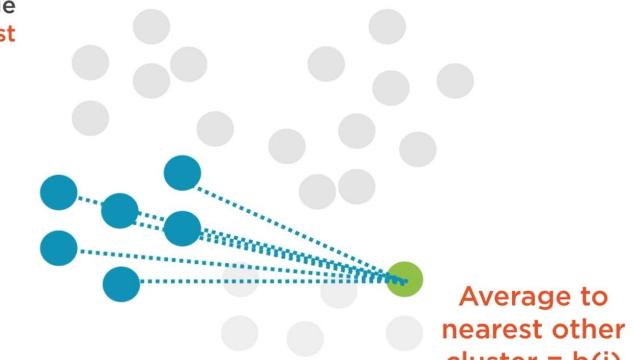
Find $a(i)$ = average distance of i to other points in **same cluster**



Average = $a(i)$

Silhouette Coefficient

Find $b(i)$ = average distance to **nearest other cluster**

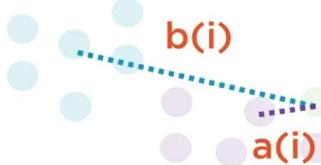


Average to
nearest other
cluster = $b(i)$

idealnya $a(i) \ll b(i)$
jika $a(i) > b(i)$ maka **misclassified**



Silhouette Coefficient



For any point i

$$s(i) = \frac{b(i) - a(i)}{\text{Larger of } b(i) \text{ and } a(i)}$$

$a(i)$ = Average distance inside cluster

$b(i)$ = Average distance to nearest other cluster

Ideally $s(i) = 1$

Ideally, $a(i) = 0$, $b(i) = \text{Infinity}$

$$s(i) = \frac{b(i) - a(i)}{\text{Larger of } b(i) \text{ and } a(i)} = 1$$

Worst-case $s(i) = -1$

Worst case, $a(i) = \text{Infinity}$, $b(i) = 0$

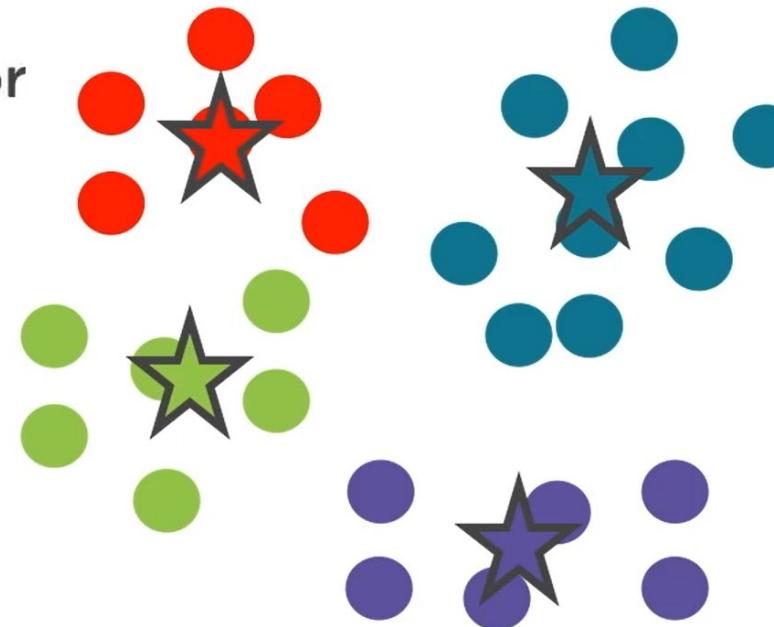
$$s(i) = \frac{b(i) - a(i)}{\text{Larger of } b(i) \text{ and } a(i)} = -1$$



Silhouette Coefficient

Silhouette Plot

Calculate $s(i)$ for each point





Membandingkan Model

- Model yang telah dibangun diharapkan memiliki akurasi yang lebih baik.
- Contoh:

Dua buah model, yaitu model 1 dan model 2, dilakukan pengujian terhadap 10 data dengan hasil seperti di samping. Dengan menghitung ratio jumlah percobaan valid terhadap jumlah percobaan diperoleh:

Akurasi model 1 : $6/10 = 60\%$

Akurasi model 2 : $5/10 = 50\%$

Data Uji	Model 1	Model 2
1	valid	tidak valid
2	tidak valid	tidak valid
3	valid	valid
4	valid	valid
5	tidak valid	tidak valid
6	valid	valid
7	valid	tidak valid
8	tidak valid	tidak valid
9	valid	valid
10	tidak valid	valid



Membandingkan Model

- Akurasi model yang lebih tinggi belum cukup untuk dapat diklaim bahwa model tersebut **secara statistik** signifikan berbeda (dan lebih baik) dari model lainnya.
- Untuk mendukung klaim bahwa model 1 lebih baik dari model 2 perlu pengujian secara statistik dengan membuat dua hipotesis yang berlawanan:
 H_0 : Kedua model memiliki akurasi yang sama
 H_1 : Kedua model memiliki akurasi yang berbeda
- Pengujian statistik yang sederhana dapat dilakukan dengan **McNemar's Test**
- Untuk pengujian lainnya yang lebih detail (5 cv test dsb.) silakan dilanjutkan ke pengayaan.



McNemar's Test

- Data pengujian disusun menjadi tabel contingency sebagai berikut (perhatikan pasangan dalam model 1/model 2)

	Model 2 valid	Model 2 tidak valid
Model 1 valid	valid/valid	valid/tidak valid
Model 1 tidak valid	tidak valid/valid	tidak valid/tidak valid

- Sehingga dari tabel sebelumnya diperoleh :

	Model 2 valid	Model 2 tidak valid
Model 1 valid	4	2
Model 1 tidak valid	1	3



McNemar's Test

- McNemar's test statistic dihitung dengan
- $S = (valid/tidak valid - tidak valid/valid)^2 / (valid / tidak valid + tidak valid/valid)$
- Hal penting dari S di atas adalah klaim statistik konsen kepada perbedaan valid dan tidak valid pada kedua model, bukan pada akurasi maupun tingkat error
- Melakukan perhitungan statistik lebih lanjut, perlu memperhatikan masing masing nilai dalam tabel contingency. Distribusi $\square X^2$ mengasumsikan nilai nilai besar untuk nilai elemen-elemen tabel contingency. Untuk nilai kecil, digunakan distribusi Binomial. Dalam praktikal, nilai S di atas dilakukan koreksi. Perhitungan detail statistik ini dapat dibaca di referensi.



Parameter penting dalam McNemar's Test

- Parameter dalam McNemar's Test, selain s adalah p
- Dalam penggunaan praktikal, dapat digunakan perintah (dalam python) untuk mendapatkan dua nilai ini, dengan memperhatikan apakah nilai elemen tabel contingency besar atau kecil
- Contoh : dari table contingency sebelumnya, dapat dituliskan:

$$T = [[4, 2], \\ [1, 3]]$$

- Untuk case nilai-nilai kecil (misalnya tabel contingency T di atas), dapat digunakan perintah:
 $s, p = mcnemar(T, exact=True)$
- Parameter lain adalah ambang batas p untuk threshold, yaitu $\square \alpha$, misalnya $\alpha = 0.05$



Penolakan / Penerimaan hipotesis

- Berdasarkan nilai p dan ambang α dapat ditentukan:
- Jika $p > \alpha$, hipotesis H_0 gagal untuk ditolak, kedua model secara statistik tidak ada perbedaan
- Jika $p \leq \alpha$, hipotesis H_0 ditolak, kedua model secara statistik secara signifikan ada perbedaan
- McNemar's adalah pengujian yang sederhana dan telah berkembang diantaranya 5xcv t-test beserta pengembangannya. Detail teori pengujian ini dapat dilihat di referensi.



Source Code

Import library

```
from statsmodels.stats.contingency_tables import mcnemar
```

Asumsi tabel contingency sudah tersedia

```
conti = [[4, 2],  
         [1, 3]]
```



```
[[4, 2], [1, 3]]
```



Source Code

Perhitungan mcnemar test dilakukan dengan fungsi mcnemar

```
retval = mcnemar(conti,  
exact=True)
```

Menampilkan nilai statistic dan p value

```
print('Nilai statistic =%.3f, \nNilai  
p-value =%.3f' % (retval.statistic,  
retval.pvalue))
```



**Nilai statistic =1.000,
Nilai p-value =1.000**



Source Code

Pengecekan nilai p-value, dengan mengambil sebuah nilai alpha

```
alpha = 0.01

if retVal.pvalue > alpha:
    print('Hipotesis H0 gagal ditolak, kedua
model memiliki peluang eror yang sama')
else:
    print('Hipotesis H0 ditolak, kedua model
memiliki peluang eror yang berbeda')
```



Hipotesis H0 gagal ditolak, kedua model
memiliki peluang eror yang sama



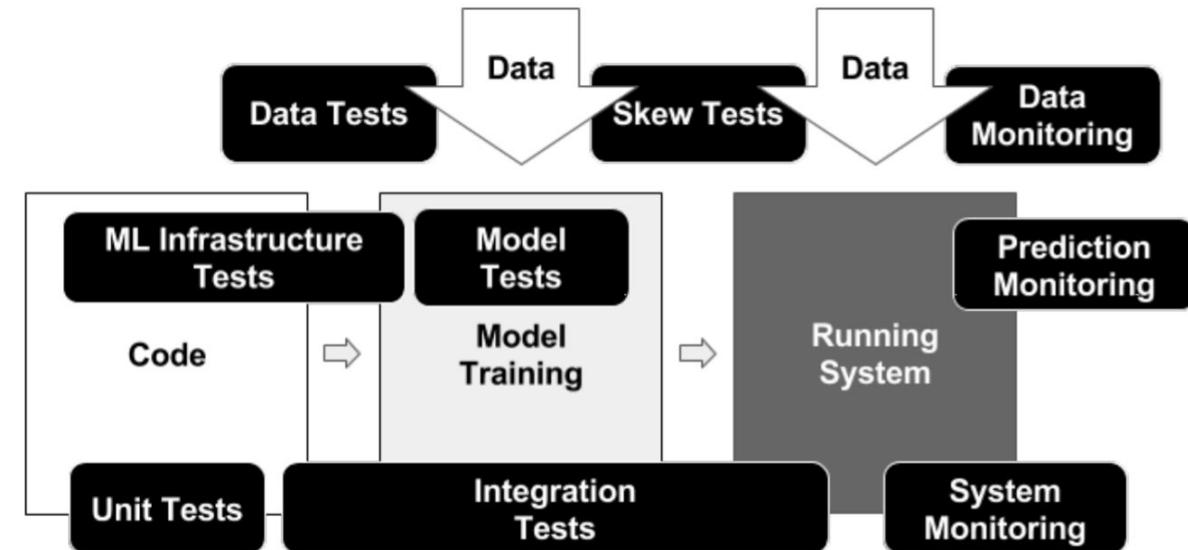
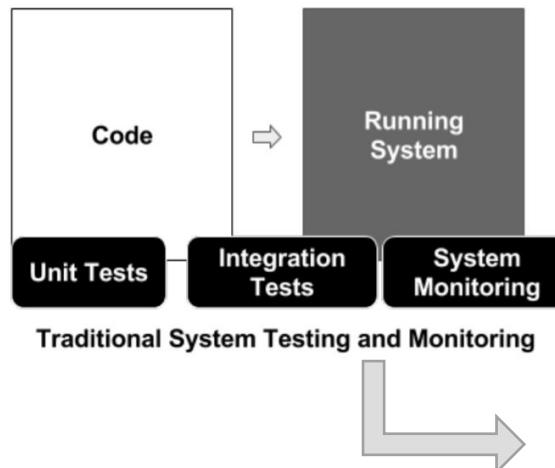
Sources

- *Klasifikasi*
- *Regresi:*
- <https://www.dataquest.io/blog/understanding-regression-error-metrics/>
- *Klasifikasi (ROC)*
- <https://www.youtube.com/watch?v=z5qA9qZMyw0>
- <https://www.youtube.com/watch?v=4jRBRDbJemM>
- <https://www.youtube.com/watch?v=4jRBRDbJemM&t=349s>
- *Regresi*
- *Clustering*
- *Evaluasi Model (Mc Nemar test dll.)*
- https://sebastianraschka.com/pdf/lecture-notes/stat479fs18/11_eval-algo_slides.pdf
- <https://www.youtube.com/watch?v=z5qA9qZMyw0>
- <https://machinelearningmastery.com/mcnemars-test-for-machine-learning/>
- Thomas G. Dietterich; Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. *Neural Comput* 1998; 10 (7): 1895–1923. doi: <https://doi.org/10.1162/089976698300017197>

•

Keunikan sistem berbasis ML

- Dipengaruhi data yang dinamis
- Dipengaruhi konfigurasi model



ML-Based System Testing and Monitoring



Google ML Test Score

Google ML Test Score menguji sistem berbasis ML melalui 28 kriteria yang secara umum dikelompokkan menjadi 4 :

- Memelihara semua fitur dalam skema, hanya menyimpan fitur yang penting dan tidak terlalu rumit, dapat digunakan tanpa melanggar privasi atau peraturan yang berlaku.
- Membuat model dalam lingkungan yang tercatat perkembangannya, mengoptimalkan parameter model, dan melakukan pemeriksaan rutin terhadap model dasar
- Membangun pipeline ML terintegrasi yang dapat didebug dengan mudah dan diuji sebelum diimplementasikan ke sistem produksi (setiap penambahan disertai alternatif rollback).
- Memantau ketidaktersediaan atau perubahan data input, inkonsistensi antara sub-bagian training dan scoring, penurunan kualitas statistik model, atau kecepatan keseluruhan sistem.



Data

1. *Ekspektasi fitur ditangkap dalam skema.*
2. *Semua fitur bermanfaat.*
3. *Tidak ada biaya fitur yang terlalu banyak.*
4. *Fitur mematuhi persyaratan tingkat meta.*
5. *Pipa data memiliki kontrol privasi yang sesuai.*
6. *Fitur baru dapat ditambahkan dengan cepat.*
7. *Semua kode fitur input diuji.*



Model

1. *Setiap spesifikasi model menjalani tinjauan kode dan diperiksa ke repositori.*
2. *Metrik proxy offline berkorelasi dengan metrik dampak online yang sebenarnya.*
3. *Semua hyperparameter telah disetel.*
4. *Dampak dari model staleness diketahui.*
5. *Model yang lebih sederhana tidak lebih baik.*
6. *Kualitas model cukup pada semua irisan data penting.*
7. *Model telah diuji untuk pertimbangan inklusi.*



Infra

1. Pelatihan dapat direproduksi.
2. Kode spesifikasi model diuji unit.
3. Pipeline ML lengkap diuji integrasinya.
4. Kualitas model divalidasi sebelum mencoba menyajikannya.
5. Model memungkinkan debugging dengan mengamati perhitungan langkah-demi-langkah pelatihan atau inferensi pada satu contoh.
6. Model diuji melalui proses kenari sebelum memasuki lingkungan penyajian produksi.
7. Model dapat dengan cepat dan aman dikembalikan ke versi penyajian sebelumnya.



Monitor

1. *Perubahan ketergantungan menghasilkan pemberitahuan.*
2. *Data invarian bertahan dalam pelatihan dan penyajian input.*
3. *Fitur pelatihan dan penayangan menghitung nilai yang sama.*
4. *Model tidak terlalu basi.*
5. *Model ini stabil secara numerik.*
6. *Model tidak mengalami regresi yang dramatis atau kebocoran lambat dalam kecepatan pelatihan, latensi penyajian, throughput, atau penggunaan RAM.*
7. *Model tersebut belum mengalami regresi kualitas prediksi pada data yang disajikan.*



Sources

- <https://static.googleusercontent.com/media/research.google.com/id//pubs/archive/aad9f93b86b7addfea4c419b9100c6cdd26cacea.pdf>
- <https://www.kaggle.com/discussion/217946>
- <https://medium.com/@rasmi/the-ml-production-readiness-of-teslas-autopilot-80acd03b3089>
- https://ckaestne.github.io/seai/S2020/slides/13_infrastructurequality/infrastucturequality.pdf
- <https://blog.dataiku.com/the-google-ml-test-score-measuring-your-sust-ai-nability>



Quiz / Tugas

Quiz dapat diakses melalui <https://spadadikti.id/>



Terima kasih