Metrik Evaluasi, Validasi Hasil & Analisis Statistik

Metodologi Penelitian Kuantitatif

Hendri Karisma, M.T.
Dosen Teknik Informatika, STMIK Tazkia, Bogor, Indonesia
VP Engineering, Jejakin.com
hendri@stmik.tazkia.ac.id — hendri.karisma@jejakin.com

Program Studi Teknik Informatika
STMIK Tazkia, Bogor, Indonesia

2026

Outline

Confusion Matrix: Fondasi Metrik Klasifikasi

Setiap prediksi model klasifikasi jatuh ke salah satu dari 4 kategori:

	Prediksi Positif	**Prediksi Negatif**
Aktual Positif	**TP** (True Positive)	**FN** (False Negative)
Aktual Negatif	**FP** (False Positive)	**TN** (True Negative)

- **TP:** Model benar memprediksi positif. *Email spam terdeteksi sebagai spam.*
- **TN:** Model benar memprediksi negatif. *Email normal tidak ditandai spam.*
- **FP (Type I Error):** Model salah memprediksi positif. *Email dosen masuk folder spam!*
- **FN (Type II Error):** Model salah memprediksi negatif. *Spam lolos ke inbox!*

Accuracy: Metrik yang Intuitif tapi Berbahaya

Rumus Accuracy

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

BAHAYA: Accuracy pada Dataset Imbalanced

Dataset fraud: 9.950 transaksi normal, 50 transaksi fraud.

Model yang **selalu memprediksi "normal"** mendapat Accuracy $= \frac{9950}{10000} =$ **99.5%**!

Padahal model tersebut **tidak mendeteksi satu pun fraud**. Accuracy 99.5% ini **menyesatkan**.

Kesimpulan: Jangan gunakan Accuracy sendirian pada dataset *imbalanced*!

Precision & Recall: Dua Sisi Mata Uang

Precision

$$\text{Precision} = \frac{TP}{TP + FP}$$

"Dari yang diprediksi positif, berapa % yang benar?"

Penting ketika FP mahal:

- Filter spam email
- Rekomendasi produk
- Prediksi saham

Recall (Sensitivitas)

$$\text{Recall} = \frac{TP}{TP + FN}$$

"Dari yang benar-benar positif, berapa % yang terdeteksi?"

Penting ketika FN mahal:

- Deteksi kanker
- Deteksi intrusi jaringan
- Deteksi malware

F1-Score: Keseimbangan Precision dan Recall

Rumus F1-Score (Harmonic Mean)

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Mengapa Harmonic Mean, bukan Arithmetic Mean?

Skenario	Precision	Recall	F1 vs Avg
Model seimbang	0.85	0.80	F1=0.824, Avg=0.825
Model tidak seimbang	1.00	0.01	F1=**0.020**, Avg=0.505

Harmonic mean **menghukum** ketimpangan! Model dengan Recall 0.01 mendapat F1 = 0.02 (bukan 0.505).

Gunakan F1-Score pada dataset *imbalanced* sebagai pengganti Accuracy.

AUC-ROC: Kurva dan Interpretasi

ROC Curve: Plot *True Positive Rate* (Recall) vs *False Positive Rate* untuk berbagai threshold.

Interpretasi AUC:

- AUC = 1.0: Model sempurna
- AUC = 0.5: Model = tebakan acak
- AUC < 0.5: Prediksi terbalik

Skala penilaian:

- > 0.9: Sangat Baik
- 0.8–0.9: Baik
- 0.7–0.8: Cukup
- 0.6–0.7: Kurang
- 0.5–0.6: Gagal

FPR (False Positive Rate)

$$FPR = \frac{FP}{FP + TN}$$

Keunggulan AUC-ROC:

- Tidak bergantung pada threshold tertentu
- Merangkum performa secara menyeluruh
- Baik untuk perbandingan antar model

Contoh Perhitungan Manual: Spam Detection

Model diuji pada 200 email:

	Pred. Spam	**Pred. Bukan Spam**
Aktual Spam	TP = 80	FN = 20
Aktual Bukan Spam	FP = 10	TN = 90

- Accuracy $= \frac{80+90}{200} =$ **85%**
- Precision $= \frac{80}{80+10} =$ **88.9%**
- Recall $= \frac{80}{80+20} =$ **80%**
- F1 $= 2 \times \frac{0.889 \times 0.80}{0.889+0.80} =$ **84.2%**
- FPR $= \frac{10}{10+90} =$ **10%**

Interpretasi

Precision tinggi (88.9%): jika ditandai spam, kemungkinan besar memang spam. Tapi 20 spam masih lolos (Recall 80%). Untuk keamanan, perlu tingkatkan Recall.

Metrik Regresi: MAE, MSE, RMSE

MAE (Mean Absolute Error)

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

Rata-rata selisih absolut. **Satuan sama** dengan variabel target. Robust terhadap outlier.

MSE (Mean Squared Error)

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Rata-rata kuadrat selisih. **Menghukum error besar** lebih keras. Sensitif terhadap outlier.

RMSE (Root Mean Squared Error)

$$RMSE = \sqrt{MSE}$$

Metrik Regresi: MAPE dan R^2

MAPE (Mean Absolute Percentage Error)

$$MAPE = \frac{100\%}{n} \sum_{i=1}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right|$$

Error dalam **persentase**. Tidak bergantung skala data.
$< 10\%$: Sangat akurat 10–20%: Baik 20–50%: Wajar $> 50\%$: Tidak akurat

R^2 (Koefisien Determinasi)

$$R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2}$$

Seberapa baik model menjelaskan variasi data dibanding rata-rata.

- $R^2 = 1$: Model sempurna
- $R^2 = 0$: Model = prediksi rata-rata

Metrik Performa Sistem: Running Time & Throughput

Running Time (Waktu Eksekusi)

- Satuan: milidetik (ms) atau detik (s)
- **Aturan fair:** Hardware sama, minimal 30 kali percobaan, laporkan **mean** $\pm$ **std**
- Hilangkan *cold start* (percobaan pertama)
- Python: gunakan `time.perf_counter()`, BUKAN `time.time()`
- Laporkan spesifikasi: CPU, RAM, GPU

Throughput

$$\text{Throughput} = \frac{\text{Jumlah request berhasil}}{\text{Waktu (detik)}} \quad \text{(RPS)}$$

- Penting untuk: Web API, microservices, database
- Tools: Apache JMeter, Locust, wrk, ab (Apache Bench)

Metrik Performa Sistem: Latency & Resource Usage

Latency (Response Time)

Waktu respons per *request*. Laporkan dalam persentil:

- **p50 (median):** 50% request selesai dalam waktu ini
- **p95:** 95% request selesai (penting untuk SLA)
- **p99:** 99% request selesai (deteksi *tail latency*)

CPU & RAM Usage

- Tools: `htop`, `psutil` (Python), Prometheus + Grafana
- Laporkan: penggunaan rata-rata dan puncak (*peak*)

Computational Complexity (Big-O)

Laporkan kompleksitas teoretis: $O(n \log n)$ vs $O(n^2)$

"Algoritma usulan $O(n \log n)$ terbukti 3.2x lebih cepat pada $n = 100.000$."

Metrik Khusus: NLP, Computer Vision, Keamanan

NLP:

- **BLEU:** Kualitas terjemahan (n-gram match)
- **ROUGE:** Kualitas ringkasan (R-1, R-2, R-L)
- **Perplexity:** Model bahasa (makin rendah, makin baik)
- **WER:** Speech recognition

Computer Vision:

- **IoU:** Tumpang tindih bounding box. Threshold: > 0.5
- **mAP:** Mean Average Precision. Standar: mAP@0.5
- **FID:** Fréchet Inception Distance (generatif)

Keamanan Siber:

- **Detection Rate:** % serangan terdeteksi
- **FPR:** False alarm rate
- **TTD:** Time to Detect
- **FAR:** Alarm palsu per waktu

Prinsip Utama

Gunakan metrik yang **standar di bidang Anda**. Cek paper terbaru di domain yang sama dan gunakan metrik yang sama agar hasil Anda dapat dibandingkan.

Decision Tree: Pemilihan Metrik yang Tepat

1. **Output model berupa kategori/kelas?** → Metrik Klasifikasi
   - Dataset seimbang? → Accuracy + F1-Score
   - Dataset *imbalanced*? → **F1, Precision, Recall, AUC-ROC**
   - Biaya FP tinggi? → Fokus **Precision**
   - Biaya FN tinggi? → Fokus **Recall**
2. **Output model berupa angka kontinu?** → Metrik Regresi
   - Robust terhadap outlier? → **MAE**
   - Hukum error besar? → **RMSE**
   - Error relatif (persentase)? → **MAPE**
   - Kemampuan model menjelaskan data? → $\mathbf{R^2}$
3. **Evaluasi performa sistem?** → Metrik Performa
   - Kecepatan? → Running Time, Throughput
   - Responsivitas? → Latency (p50, p95, p99)
   - Efisiensi? → CPU/RAM, Big-O

Validasi Model: Hold-out & K-Fold Cross Validation

Hold-out Validation

Bagi data menjadi Training (80%) dan Test (20%).
Kelebihan: Cepat, sederhana.
Kelemahan: Hasil bergantung pada pembagian data.
Kapan cukup: Dataset besar (> 100.000 sampel).

K-Fold Cross Validation

1. Bagi data menjadi k fold sama besar
2. Iterasi i: fold ke-i = test, sisanya = training
3. Hitung metrik tiap iterasi
4. Laporkan **mean** $\pm$ **std** dari k hasil

K=5: Lebih cepat, bias sedikit lebih tinggi
K=10: Lebih akurat (standar literatur), lebih lambat

Stratified K-Fold & LOOCV

Stratified K-Fold Cross Validation

Sama seperti K-Fold biasa, tetapi **setiap fold menjaga proporsi kelas** yang sama.

- **WAJIB** untuk dataset *imbalanced*!
- Contoh: Dataset 90% negatif, 10% positif $\rightarrow$ setiap fold juga 90:10
- Tanpa stratifikasi: fold bisa tidak memiliki sampel kelas minoritas

LOOCV (Leave-One-Out Cross Validation)

K-Fold dengan $k = n$ (jumlah sampel). Setiap iterasi: 1 sampel = test, sisanya = training.

- **Kelebihan:** Bias sangat rendah
- **Kelemahan:** Sangat lambat (n iterasi pelatihan!)
- **Gunakan hanya jika** $n < 100$ (dataset sangat kecil)

Uji Statistik: Paired T-Test

Tujuan: Membuktikan perbedaan 2 metode **signifikan secara statistik**, bukan kebetulan.

Fold	Metode A	Metode B	Selisih (d)
1	0.92	0.89	0.03
2	0.88	0.87	0.01
3	0.91	0.88	0.03
4	0.90	0.86	0.04
5	0.93	0.90	0.03

$$t = \frac{\bar{d}}{s_d/\sqrt{n}} = \frac{0.028}{0.011/\sqrt{5}} = 5.69 \quad \text{vs} \quad t_{\text{tabel}}(df = 4) = 2.776$$

$t_{hitung} > t_{tabel} \Rightarrow$ **Tolak** H_0: Metode A secara signifikan lebih baik!

ANOVA & Wilcoxon Signed-Rank Test

One-Way ANOVA: Bandingkan 3+ Metode

- T-Test berulang meningkatkan *Type I Error*: $1 - (1 - 0.05)^3 = 14.3\%$!
- ANOVA menguji "semua rata-rata sama" dalam **satu langkah**
- Jika signifikan $\rightarrow$ lanjut *Post-hoc* Tukey HSD
- Contoh: Bandingkan SVM vs RF vs CNN pada dataset yang sama

Wilcoxon Signed-Rank Test (Non-Parametrik)

Alternatif Paired T-Test ketika:

- Data **tidak** berdistribusi normal
- Sampel kecil ($n < 30$)
- Data berupa ordinal/ranking

Langkah: Hitung selisih $\rightarrow$ Ranking absolut $\rightarrow$ Beri tanda $\pm$ $\rightarrow$ Hitung statistik W

Interpretasi P-Value dan Effect Size

Interpretasi P-Value

- $p < 0.01$: **Sangat signifikan** (highly significant)
- $p < 0.05$: **Signifikan** (threshold standar)
- $p < 0.10$: Marjinal (umumnya tidak cukup di CS)
- $p \geq 0.05$: **Tidak signifikan** (gagal tolak H_0)

MISKONSEPSI

$p < 0.05$ **BUKAN** berarti "95% kemungkinan metode A lebih baik."
Artinya: "Jika H_0 benar, peluang mendapat hasil se-ekstrem ini $< 5\%$."

Cohen's d (Effect Size)

$d = \frac{\bar{X}_1 - \bar{X}_2}{s_{\text{pooled}}}$ $\quad |d| < 0.2$: Kecil $\quad$ 0.2–0.8: Sedang $\quad > 0.8$: Besar

Contoh: "$t(9) = 3.45$, $p = 0.007$, Cohen's $d = 1.23$ (efek besar)."

Contoh Analisis Hasil: Tabel Perbandingan

Studi Kasus: Deteksi malware — SVM vs Random Forest vs CNN (10-Fold CV)

Metrik	SVM	Random Forest	CNN
Accuracy	91.2% $\pm$ 1.3	93.8% $\pm$ 0.9	**95.1%** $\pm$ 0.7
Precision	90.5% $\pm$ 1.5	93.2% $\pm$ 1.1	**94.8%** $\pm$ 0.8
Recall	92.0% $\pm$ 1.8	94.5% $\pm$ 1.0	**95.5%** $\pm$ 0.9
F1-Score	91.2% $\pm$ 1.4	93.8% $\pm$ 0.9	**95.1%** $\pm$ 0.7
AUC-ROC	0.952 $\pm$ 0.01	0.975 $\pm$ 0.008	**0.987** $\pm$ 0.005
Waktu Train	**12.8s** $\pm$ 0.9	45.2s $\pm$ 3.1	234.5s $\pm$ 15.2

- Laporkan **mean** $\pm$ **std** (bukan hanya satu angka!)
- **Bold** untuk nilai terbaik per metrik
- Sertakan uji statistik untuk klaim "lebih baik"

Contoh Interpretasi Naratif untuk Bab 4

Contoh Penulisan yang Benar (Bab 4 Skripsi)

"Berdasarkan Tabel 4.1, CNN menghasilkan performa terbaik di seluruh metrik klasifikasi dengan accuracy **95.1%** (SD = 0.7%), precision 94.8%, recall 95.5%, dan F1-score 95.1%.

Uji One-Way ANOVA menunjukkan perbedaan yang signifikan ($F = 28.45$, $p < 0.001$). Uji *post-hoc* Tukey HSD mengkonfirmasi setiap pasangan berbeda signifikan ($p < 0.05$).

Namun, CNN membutuhkan waktu pelatihan 234.5 detik dibanding Random Forest 12.8 detik. Dalam skenario *real-time retraining*, RF dapat menjadi alternatif efisien."

Grafik yang wajib ditampilkan:

- Bar Chart metrik + error bar
- ROC Curve semua metode
- Box Plot distribusi per fold

Ringkasan Pertemuan 13

1. **Pilih metrik sesuai jenis penelitian:** klasifikasi, regresi, atau performa sistem
2. **Jangan hanya gunakan Accuracy** pada dataset *imbalanced*
3. **Laporkan mean $\pm$ std**, gunakan **bold** untuk nilai terbaik
4. **Validasi model:** K-Fold (k=10) adalah standar; Stratified untuk *imbalanced*
5. **Uji statistik WAJIB:** T-Test (2 metode), ANOVA (3+ metode)
6. **Laporkan p-value DAN effect size** (Cohen's d)
7. **Interpretasi naratif:** Jangan hanya menyajikan angka, jelaskan **maknanya**
8. **Grafik pendukung:** Bar chart, ROC curve, box plot

Pesan Utama

Angka tanpa uji statistik = **klaim tanpa bukti**. Angka tanpa interpretasi = **data tanpa makna**.

Tugas Pertemuan 13

Tugas (Dikumpulkan Pertemuan Berikutnya)

1. Tentukan **3–5 metrik evaluasi** yang akan Anda gunakan di skripsi. Jelaskan **mengapa** metrik tersebut dipilih (bukan metrik lain).
2. Buat **tabel rancangan hasil** (template kosong) yang akan Anda isi di Bab 4 nanti. Tabel harus memuat kolom: Metode, Metrik 1, Metrik 2, ..., Waktu.
3. Tentukan metode **validasi** yang akan digunakan (Hold-out / K-Fold / Stratified K-Fold). Jelaskan alasannya.
4. Tentukan **uji statistik** yang sesuai. Jika membandingkan 2 metode: T-Test atau Wilcoxon? Jika 3+: ANOVA?
5. **Bonus:** Jika sudah punya data awal, hitung salah satu metrik secara manual dan tunjukkan langkah perhitungannya.