

# PORTFOLIO

## Breast Cancer Dataset Classification



# Halo, saya Hilma Zahra Qorina



## Tentang saya

Saya seorang mahasiswa Universitas Negeri Surabaya, jurusan Teknik Informatika. Saya memiliki semangat untuk terus belajar dan mengembangkan diri di bidang Data Science, khususnya dalam analisis data dan pengembangan berbagai model berbasis Machine Learning.

Saya percaya bahwa data adalah aset berharga, sehingga saya terus mendalami eksplorasi, preprocessing, dan implementasi machine learning untuk menghasilkan model yang efektif dan andal. Melalui portofolio ini, saya ingin membagikan antusiasme saya terhadap Data Science.

# Dataset Breast Cancer

---

## Dataset

Dataset **load\_breast\_cancer** dari scikit-learn dirancang untuk mengklasifikasikan tumor payudara sebagai **malignant** (kanker) atau **benign** (tidak kanker). Dataset ini terdiri dari **569 contoh** dengan **30 atribut** yang menggambarkan berbagai fitur dari inti sel, seperti radius, tekstur, area, dan kelancaran. Dataset ini sering digunakan untuk mendemonstrasikan algoritma pembelajaran mesin dalam tugas **klasifikasi**.

## Tujuan

Tujuan utama dari dataset ini adalah untuk **memprediksi jenis tumor** berdasarkan fitur-fitur tersebut. Ini sangat penting dalam membantu deteksi dini dan diagnosis kanker payudara.

Dengan menggunakan dataset ini, para peneliti dan praktisi dapat mengembangkan model yang dapat membantu dalam pengambilan keputusan medis terkait kesehatan payudara.

# Tools

---



# Data Overview

`df.describe()`

Menampilkan 10 baris pertama dari dataset **breast cancer**.

	mean radius	mean texture	mean perimeter	mean area	mean smoothness	mean compactness	mean concavity	mean concave points	mean symmetry	mean fractal dimension	...	worst texture	worst perimeter	worst area	worst smoothness	worst concavity	worst concave points	worst symmetry	worst fractal dimension	target
0	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.30010	0.14710	0.2419	0.07871	...	17.33	184.60	2019.0	0.1622	0.7119	0.2654	0.4601	0.11890	0
1	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.08690	0.07017	0.1812	0.05667	...	23.41	158.80	1956.0	0.1238	0.2416	0.1860	0.2750	0.08902	0
2	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.19740	0.12790	0.2069	0.05999	...	25.53	152.50	1709.0	0.1444	0.4504	0.2430	0.3613	0.08758	0
3	11.42	20.38	77.58	386.1	0.14250	0.28390	0.24140	0.10520	0.2597	0.09744	...	26.50	98.87	567.7	0.2098	0.6869	0.2575	0.6638	0.17300	0
4	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.19800	0.10430	0.1809	0.05883	...	16.67	152.20	1575.0	0.1374	0.4000	0.1625	0.2364	0.07678	0
5	12.45	15.70	82.57	477.1	0.12780	0.17000	0.15780	0.08089	0.2087	0.07613	...	23.75	103.40	741.6	0.1791	0.5355	0.1741	0.3985	0.12440	0
6	18.25	19.98	119.60	1040.0	0.09463	0.10900	0.11270	0.07400	0.1794	0.05742	...	27.66	153.20	1606.0	0.1442	0.3784	0.1932	0.3063	0.08368	0
7	13.71	20.83	90.20	577.9	0.11890	0.16450	0.09366	0.05985	0.2196	0.07451	...	28.14	110.60	897.0	0.1654	0.2678	0.1556	0.3196	0.11510	0
8	13.00	21.82	87.50	519.8	0.12730	0.19320	0.18590	0.09353	0.2350	0.07389	...	30.73	106.20	739.3	0.1703	0.5390	0.2060	0.4378	0.10720	0
9	12.46	24.04	83.97	475.9	0.11860	0.23960	0.22730	0.08543	0.2030	0.08243	...	40.68	97.65	711.4	0.1853	1.1050	0.2210	0.4366	0.20750	0

10 rows × 31 columns

# Data Overview

## df.info()

- Dataset memiliki **569 baris** dan **31 kolom**.
- Semua kolom bertipe **float64** (numerik), kecuali kolom **target**.
- Kolom **target** memiliki tipe **int64**, karena hanya berisi angka **0** dan **1**.
- **Tidak ada missing values** karena jumlah non-null sama dengan jumlah baris.
- Dataset menggunakan 137.9 KB memori.

```
df['target'].unique()
```

```
array([0, 1])
```

Dalam dataset **breast cancer**, kolom **target** berisi dua kelas:

- **0** → Tumor ganas (malignant)
- **1** → Tumor jinak (benign)

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 569 entries, 0 to 568
Data columns (total 31 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   mean radius                           569 non-null    float64
1   mean texture                          569 non-null    float64
2   mean perimeter                        569 non-null    float64
3   mean area                            569 non-null    float64
4   mean smoothness                      569 non-null    float64
5   mean compactness                     569 non-null    float64
6   mean concavity                       569 non-null    float64
7   mean concave points                  569 non-null    float64
8   mean symmetry                        569 non-null    float64
9   mean fractal dimension                569 non-null    float64
10  radius error                          569 non-null    float64
11  texture error                         569 non-null    float64
12  perimeter error                      569 non-null    float64
13  area error                           569 non-null    float64
14  smoothness error                     569 non-null    float64
15  compactness error                    569 non-null    float64
16  concavity error                      569 non-null    float64
17  concave points error                 569 non-null    float64
18  symmetry error                       569 non-null    float64
19  fractal dimension error              569 non-null    float64
...
29  worst fractal dimension              569 non-null    float64
30  target                              569 non-null    int64
dtypes: float64(30), int64(1)
memory usage: 137.9 KB
```



# EDA

## (Exploratory Data Analysis)

### df.describe()

Untuk memahami karakteristik dataset sebelum melakukan pemodelan lebih lanjut, perlu dilakukan **statistik deskriptif**. Output dari df.describe() menampilkan berbagai metrik penting, seperti **mean** (rata-rata), **std** (standar deviasi), **min** & **max** (nilai minimum dan maksimum), serta **25%**, **50%**, dan **75%** yang mewakili persentil atau kuartil.

	mean radius	mean texture	mean perimeter	mean area	mean smoothness	mean compactness	mean concavity	mean concave points	mean symmetry	mean fractal dimension	...	worst texture	worst perimeter	worst area	worst smoothness	worst compactness	worst concavity	worst concave points	worst symmetry	worst fractal dimension	target
count	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000	...	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000
mean	14.127292	19.289649	91.969033	654.889104	0.096360	0.104341	0.088799	0.048919	0.181162	0.062798	...	25.677223	107.261213	880.583128	0.132369	0.254265	0.272188	0.114606	0.290076	0.083946	0.627417
std	3.524049	4.301036	24.298981	351.914129	0.014064	0.052813	0.079720	0.038803	0.027414	0.007060	...	6.146258	33.602542	569.356993	0.022832	0.157336	0.208624	0.065732	0.061867	0.018061	0.483918
min	6.981000	9.710000	43.790000	143.500000	0.052630	0.019380	0.000000	0.000000	0.106000	0.049960	...	12.020000	50.410000	185.200000	0.071170	0.027290	0.000000	0.000000	0.156500	0.055040	0.000000
25%	11.700000	16.170000	75.170000	420.300000	0.086370	0.064920	0.029560	0.020310	0.161900	0.057700	...	21.080000	84.110000	515.300000	0.116600	0.147200	0.114500	0.064930	0.250400	0.071460	0.000000
50%	13.370000	18.840000	86.240000	551.100000	0.095870	0.092630	0.061540	0.033500	0.179200	0.061540	...	25.410000	97.660000	686.500000	0.131300	0.211900	0.226700	0.099930	0.282200	0.080040	1.000000
75%	15.780000	21.800000	104.100000	782.700000	0.105300	0.130400	0.130700	0.074000	0.195700	0.066120	...	29.720000	125.400000	1084.000000	0.146000	0.339100	0.382900	0.161400	0.317900	0.092080	1.000000
max	28.110000	39.280000	188.500000	2501.000000	0.163400	0.345400	0.426800	0.201200	0.304000	0.097440	...	49.540000	251.200000	4254.000000	0.222600	1.058000	1.252000	0.291000	0.663800	0.207500	1.000000

8 rows × 31 columns

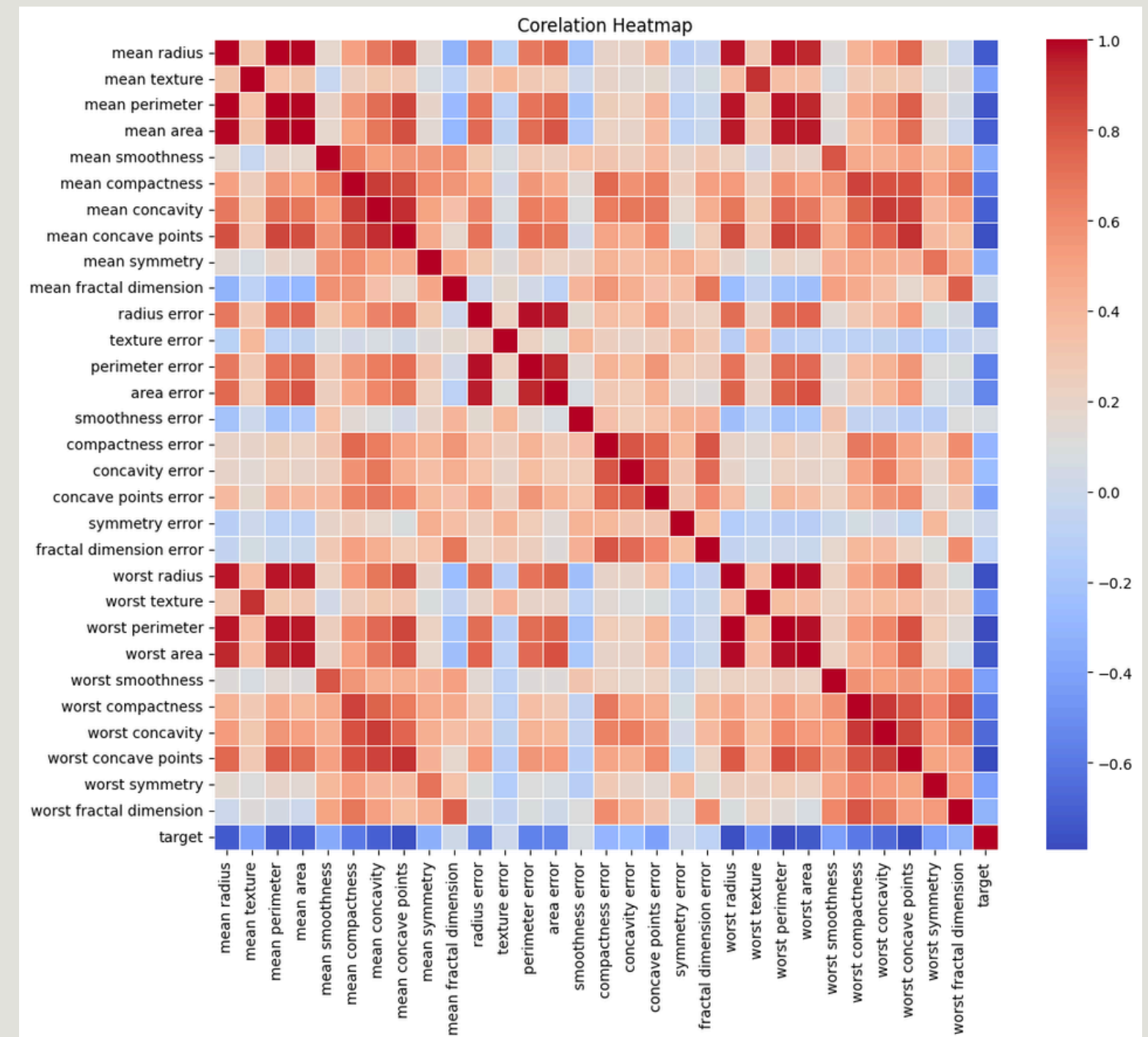
# Exploratory Data Analysis (EDA)

## Corelation Heatmap

Menampilkan hubungan **antar variabel** menggunakan warna untuk menunjukkan tingkat korelasi.

### Fungsi Corelation Heatmap

- Menemukan fitur yang berkorelasi tinggi.
- Mengetahui fitur yang paling berpengaruh terhadap target.
- Menganalisis hubungan antar variabel.





```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(df_X, df_y, test_size=0.2, random_state=42)
```

# Split Data

## Membagi dataset menjadi data latih (train) dan data uji (test)

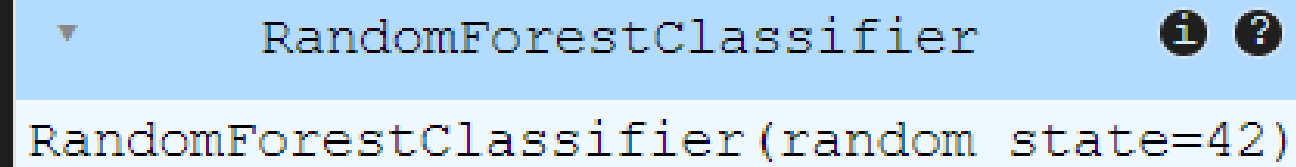
dalam machine learning agar model bisa diuji sebelum digunakan pada data baru. Penggunaan `random_state=42` memastikan bahwa hasil pembagian **tetap konsisten** setiap kali dijalankan.

# Modelling

---

```
from sklearn.ensemble import RandomForestClassifier

rf = RandomForestClassifier(n_estimators=100, random_state=42)
rf.fit(X_train, y_train)
```



▼ RandomForestClassifier ⓘ ?

```
RandomForestClassifier(random_state=42)
```

**Random Forest** adalah algoritma machine learning berbasis ensemble learning yang digunakan untuk **klasifikasi** dan **regresi**. Algoritma ini bekerja dengan membangun banyak pohon keputusan (decision trees) dan menggabungkan hasilnya untuk meningkatkan akurasi serta mengurangi overfitting.

## Mengapa Random Forest?

- **Memiliki akurasi yang lebih tinggi** dibandingkan decision tree tunggal karena menggabungkan **prediksi** dari banyak pohon keputusan.
- Dapat **mengatasi overfitting**.
- **Bekerja dengan baik** pada dataset yang memiliki **banyak fitur** dan **interaksi antar fitur yang kompleks**.
- Tetap dapat bekerja dengan baik meskipun terdapat **missing value** atau **outlier**.

### Laporan Klasifikasi:

	precision	recall	f1-score	support
0	0.98	0.93	0.95	43
1	0.96	0.99	0.97	71
accuracy			0.96	114
macro avg	0.97	0.96	0.96	114
weighted avg	0.97	0.96	0.96	114

### Confusion Matrix:

```
[[40  3]
 [ 1 70]]
```

Akurasi: 96.49%

Model memprediksi dengan benar **96.49%** dari total **114 sampel**.

- 0 = Malignant
- 1 = Benign

- **Precision** → Seberapa banyak prediksi positif yang benar?
  - Kelas 0 (Malignant) = 0.98 (98%) → Dari semua yang diprediksi sebagai Malignant, **98% benar**.
  - Kelas 1 (Benign) = 0.96 (96%) → Dari semua yang diprediksi sebagai Benign, **96% benar**.
- **Recall** → Seberapa banyak sampel yang sebenarnya positif berhasil ditemukan?
  - Kelas 0 (Malignant) = 0.93 (93%) → Dari semua yang benar-benar Malignant, model **mendeteksi 93%**.
  - Kelas 1 (Benign) = 0.99 (99%) → Dari semua yang benar-benar Benign, model **mendeteksi 99%**.
- **F1-score** → Rata-rata harmonis dari precision dan recall.
  - Kelas 0 = **0.95**
  - Kelas 1 = **0.97**

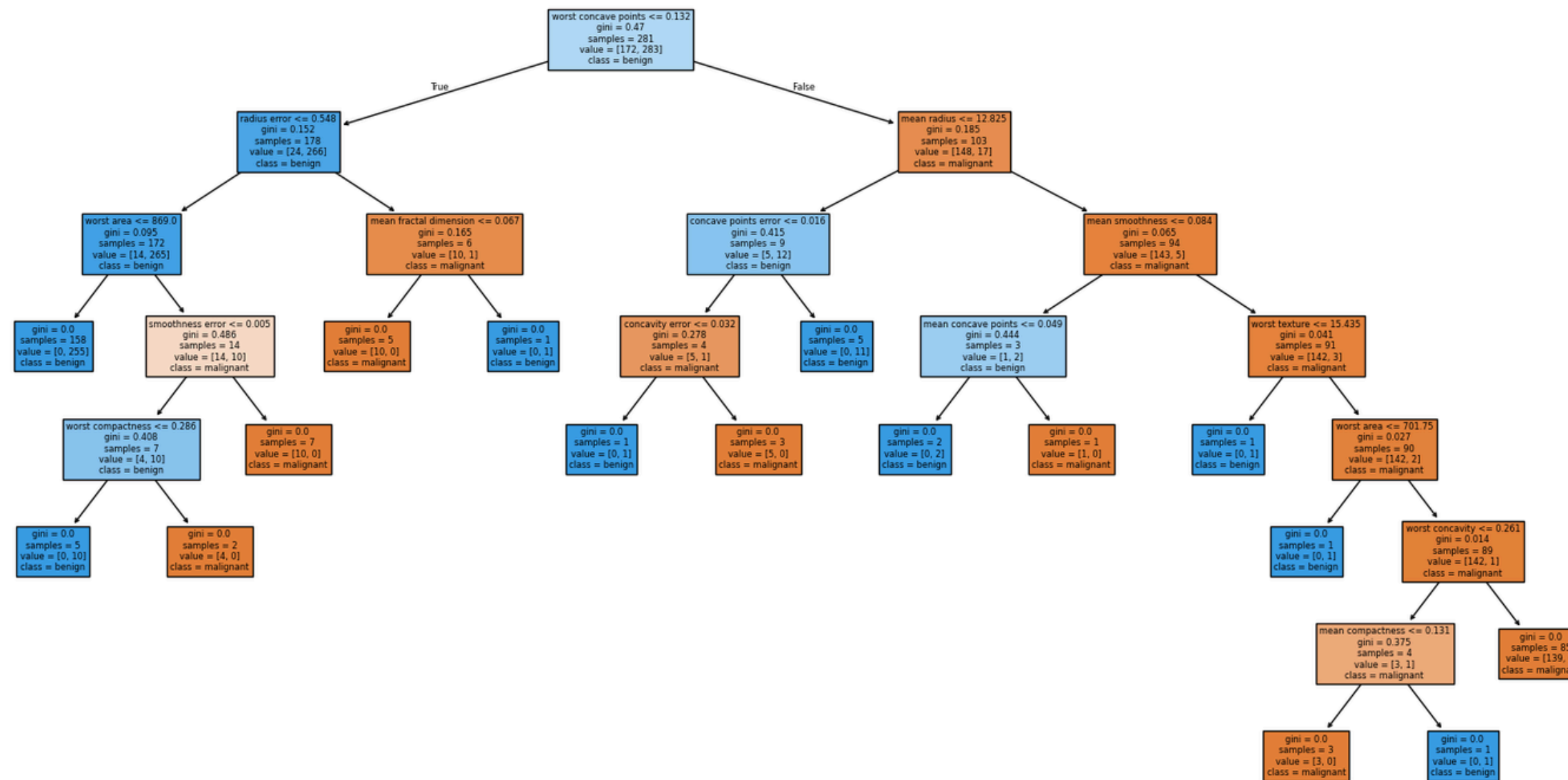
# Predict & Evaluate

# Visualization

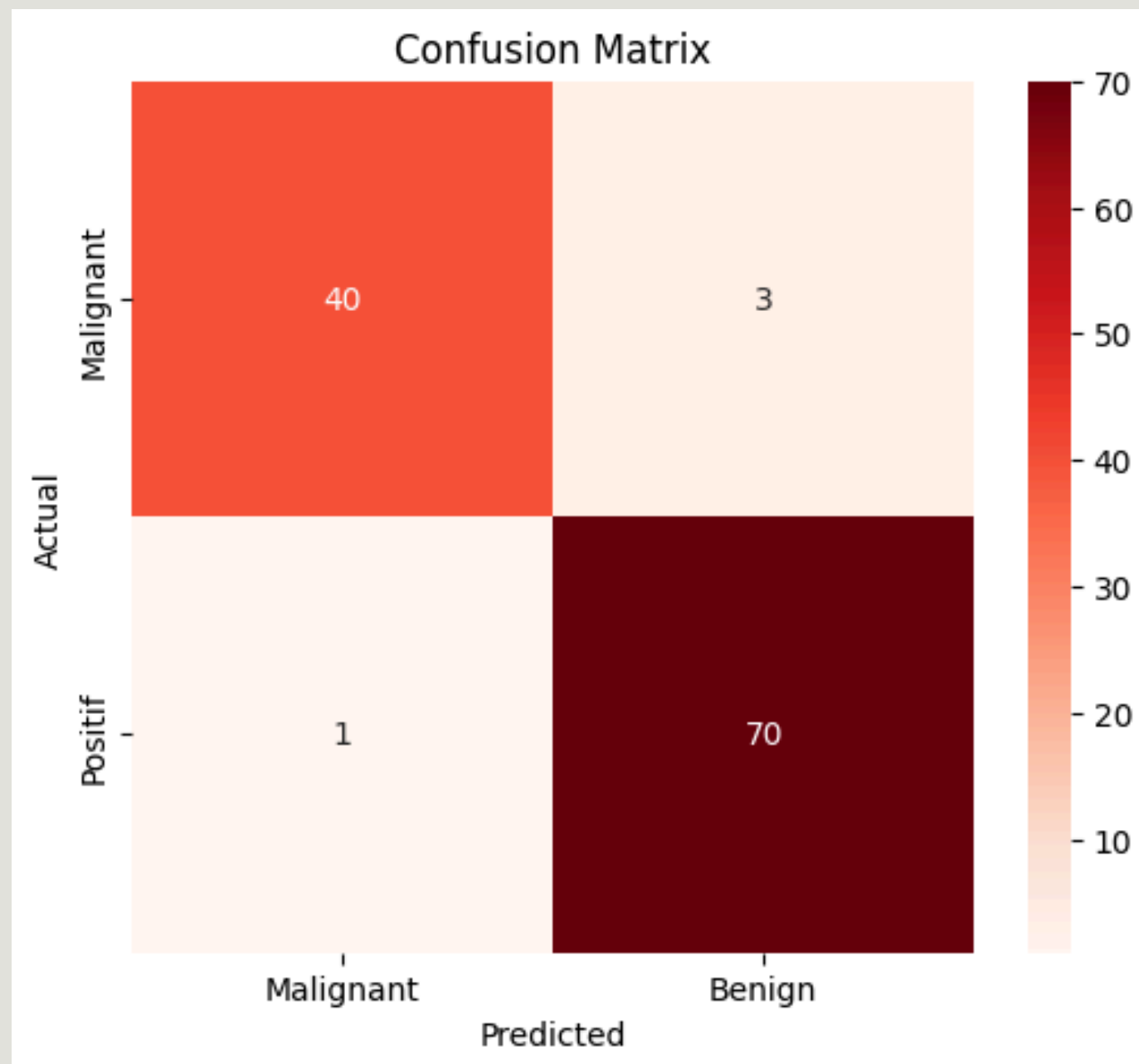
**Random Forest** adalah kumpulan dari banyak **Decision Tree**, di mana setiap pohon dilatih pada subset data yang berbeda untuk meningkatkan generalisasi model.

Dengan menampilkan **salah satu pohon** dalam Random Forest dapat membantu memahami bagaimana model membuat keputusan.

Menggunakan **estimators\_[0]** untuk mengambil **pohon pertama** dalam Random Forest.



# Visualization



**Confusion Matrix** digunakan untuk menampilkan **kinerja model klasifikasi** dengan cara membandingkan **hasil prediksi model** dengan **label asli**.

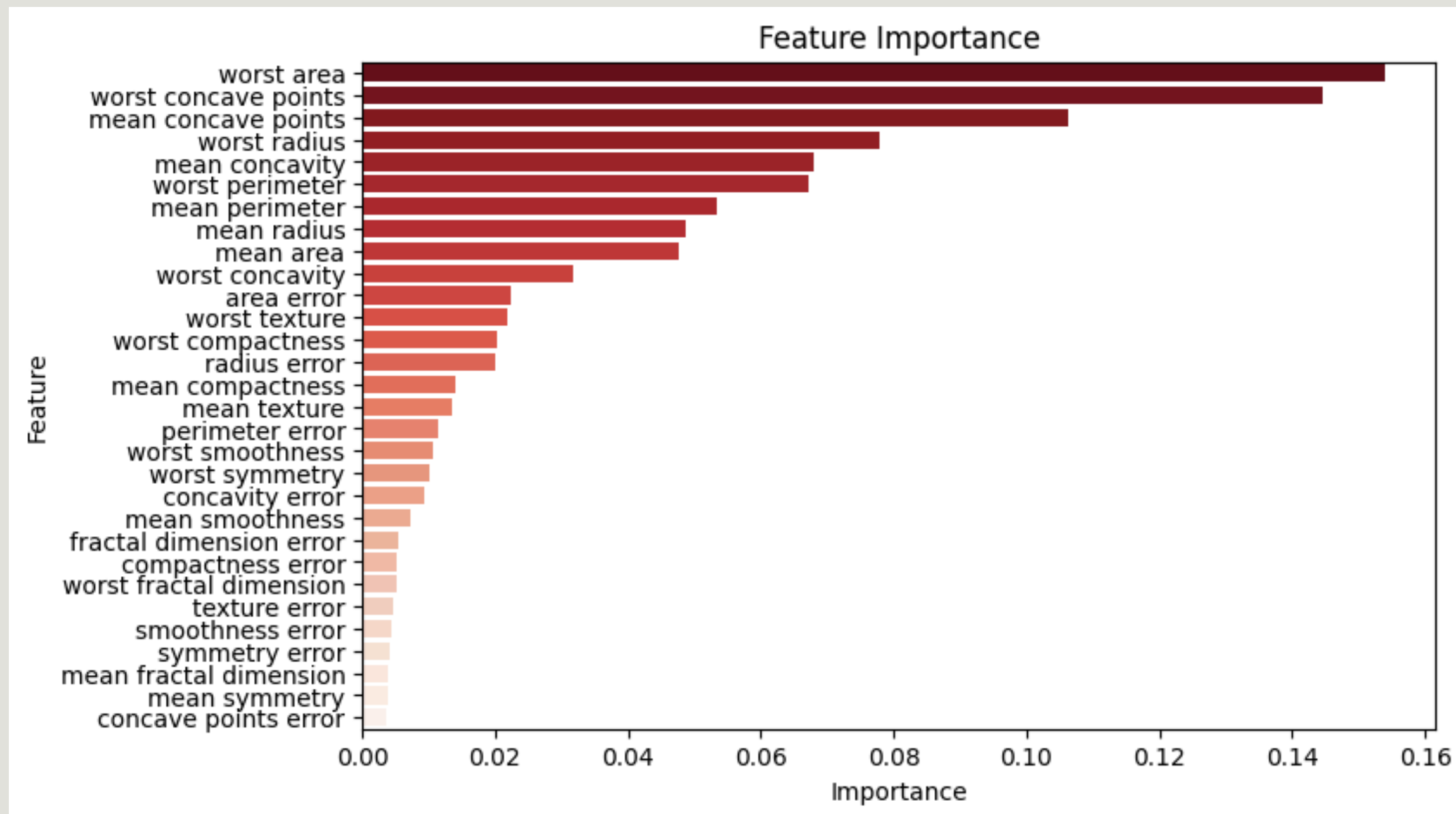
- **40 TN (True Negative)** → 40 pasien ganas diklasifikasikan dengan benar.
- **3 FP (False Positive)** → 3 pasien ganas salah diklasifikasikan sebagai jinak.
- **1 FN (False Negative)** → 1 pasien jinak salah diklasifikasikan sebagai ganas.
- **70 TP (True Positive)** → 70 pasien jinak diklasifikasikan dengan benar.

## Evaluasi Kesalahan Model

- **False Positive (FP) = 3** → Berbahaya, karena pasien mungkin tidak mendapatkan perawatan tepat waktu.
- **False Negative (FN) = 1** → Dapat menyebabkan tes tambahan atau pengobatan yang tidak perlu, tetapi lebih baik daripada salah mendeteksi kanker ganas sebagai jinak.




# Visualization



**Feature Importance** adalah ukuran seberapa **besar kontribusi setiap fitur** (variabel) terhadap prediksi model.

**Bar chart** menampilkan **perbandingan** nilai kepentingan (**importance**) dari **setiap fitur** dalam model (worst area, mean radius, worst concave points, dll.).

Terlihat bahwa **worst area** memiliki nilai importance **tertinggi**, yang berarti fitur ini sangat penting dalam model, dibandingkan dengan fitur seperti **mean symmetry** atau **concave points error** yang memiliki importance **lebih rendah**.

A solid dark grey horizontal bar.

# TERIMA KASIH

