

## **TUGAS PERTEMUAN 3**

**Disusun Untuk Memenuhi Tugas Mata Kuliah Machine Learning**

**Dosen Pengampu : Uray Yufikar, Ir., MT.**



**Disusun Oleh :**

**NAMA : MILADYNA FAUZIA**

**NPM : 41155050210023**

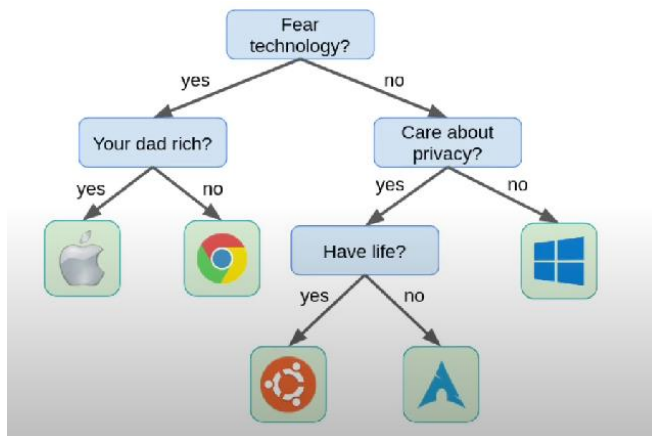
**KELAS : IF-A1**

**FAKULTAS TEKNIK PROGRAM STUDI INFORMATIKA**

**UNIVERSITAS LANGLANGBUANA**

**2024**

1. Lakukan praktik dari [https://youtu.be/5wwXKtLkyqs?si=fn88eveu\\_qbCC6b3](https://youtu.be/5wwXKtLkyqs?si=fn88eveu_qbCC6b3) , buat screenshot dengan nama kalian pada coding, kumpulkan dalam bentuk pdf, dari kegiatan ini:
  - 1.1. . Pengenalan komponen Decision Tree: root, node, leaf

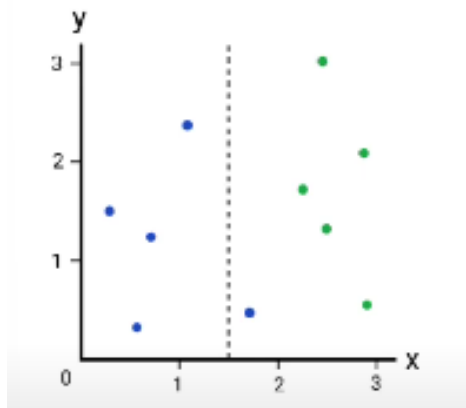


Algoritma Decision Tree Lainnya :

- ID3
- C4.5
- C5.0

## 1.2. Pengenalan Gini Impurity

Pengukuran ketidak murnian



Ruas Kiri :

$$\begin{aligned}
 G &= 1 - \sum_i^n p_i^2 \\
 &= 1 - P(biru)^2 \\
 &= 1 - \left(\frac{4}{4}\right)^2 = 0
 \end{aligned}$$

Ruas Kanan :

$$\begin{aligned}
 G &= 1 - \sum_i^n p_i^2 \\
 &= 1 - (P(biru)^2 + P(hijau)^2) \\
 &= 1 - \left(\left(\frac{1}{6}\right)^2 + \left(\frac{5}{6}\right)^2\right) = 0.278
 \end{aligned}$$

$$G = \frac{4}{4+6} \times 0 + \frac{6}{4+6} \times 0.278$$
$$= 0.1668$$

Gini Impurity = 0.6779

Average Gini Impurity = 0.1668

Information Gain = 0.6779 - 0.1668 = 0.51

	Color	Diameter	Label
0	Green	3	Apple
1	Yellow	3	Apple
2	Red	1	Grape
3	Red	1	Grape
4	Yellow	3	Lemon

Kemungkinan pertanyaan untuk splitting:

1. Color == Green?
2. Color == Yellow?
3. Color == Red?
4. Diameter <= 1
5. Diameter <= 3

Highest Information Gain

$$G = 1 - (P(\text{apple})^2 + P(\text{grape})^2 + P(\text{lemon})^2)$$

$$= 1 - ((\frac{2}{5})^2 + (\frac{2}{5})^2 + (\frac{1}{5})^2) = 0.63$$

[illegible]

## 1.6. Training model Decision Tree Classifier

```
[5]: from sklearn.tree import DecisionTreeClassifier

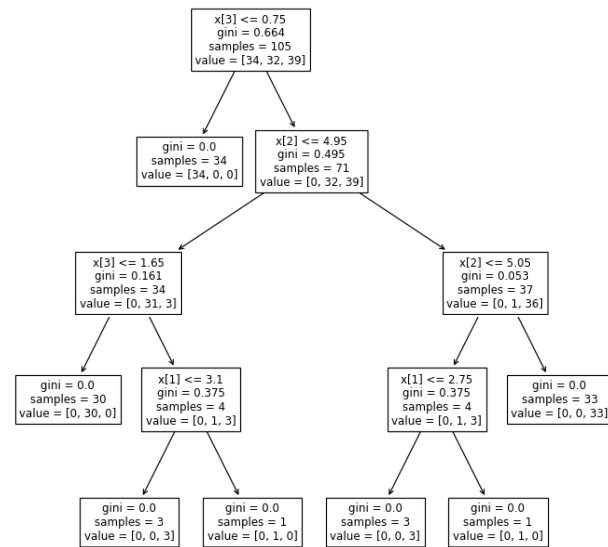
model = DecisionTreeClassifier(max_depth=4)
model.fit(X_train, y_train)

[5]: DecisionTreeClassifier
DecisionTreeClassifier(max_depth=4)
```

## 1.7. Visualisasi model Decision Tree

```
[7]: import matplotlib.pyplot as plt
from sklearn import tree

plt.rcParams['figure.dpi'] = 85
plt.subplots(figsize=(10, 10))
tree.plot_tree(model, fontsize=10)
plt.show()
```



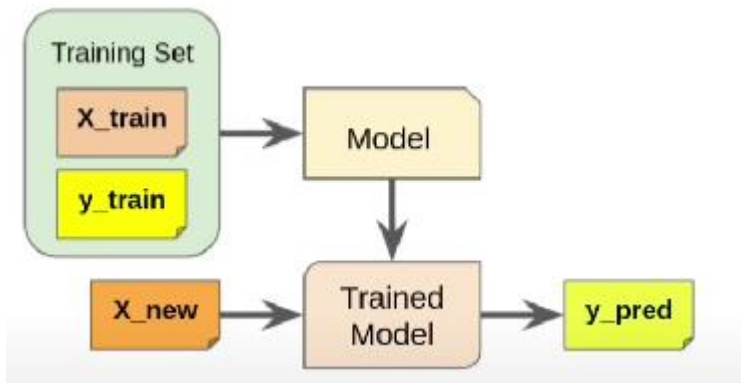
## 1.8. Evaluasi model Decision Tree

```
[9]: from sklearn.metrics import classification_report

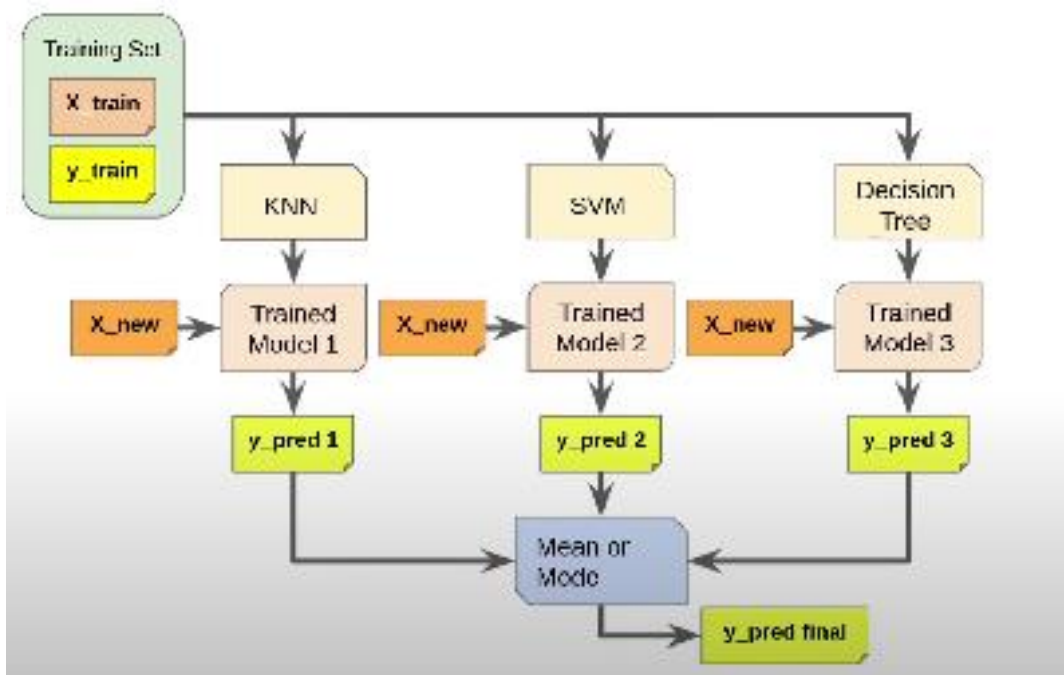
y_pred = model.predict(X_test)
print(classification_report(y_test, y_pred))
```

	precision	recall	f1-score	support
0	1.00	1.00	1.00	16
1	1.00	0.94	0.97	18
2	0.92	1.00	0.96	11
accuracy			0.98	45
macro avg	0.97	0.98	0.98	45
weighted avg	0.98	0.98	0.98	45

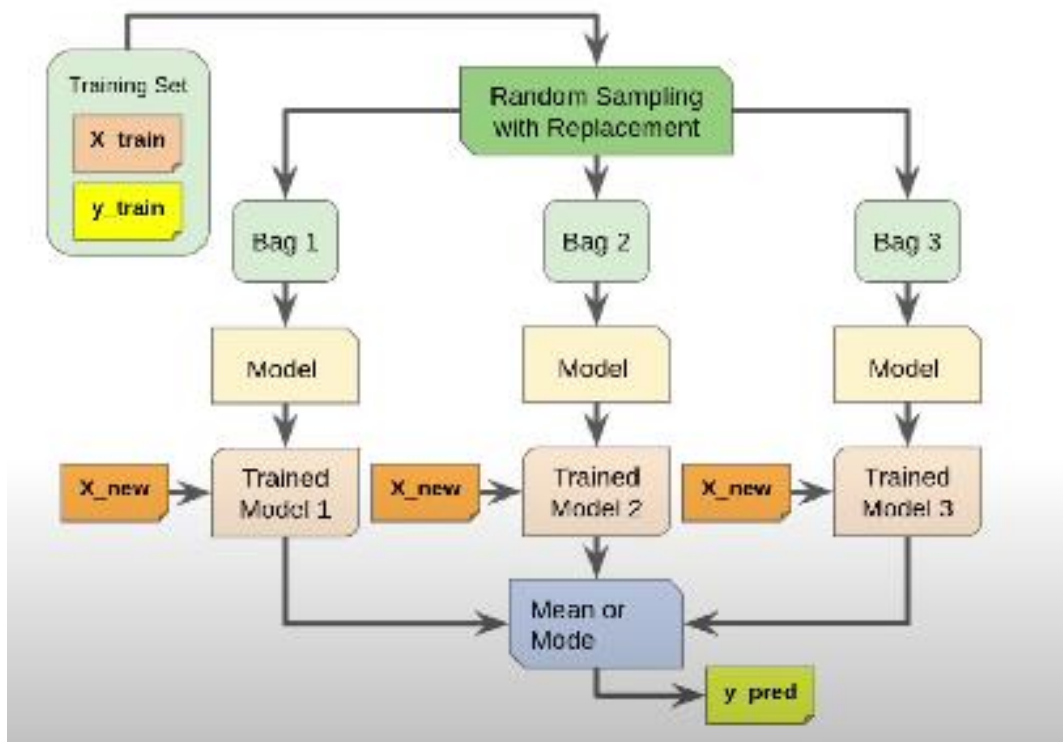
2. Lakukan praktik dari [https://youtu.be/yKovaQ6tyV8?si=HnHG6kcoCsDwvo\\_0](https://youtu.be/yKovaQ6tyV8?si=HnHG6kcoCsDwvo_0) , buat screenshot dengan nama kalian pada coding, kumpulkan dalam bentuk pdf, dari kegiatan ini:
- 2.1. Proses training model Machine Learning secara umum



2.2. Pengenalan Ensemble Learning

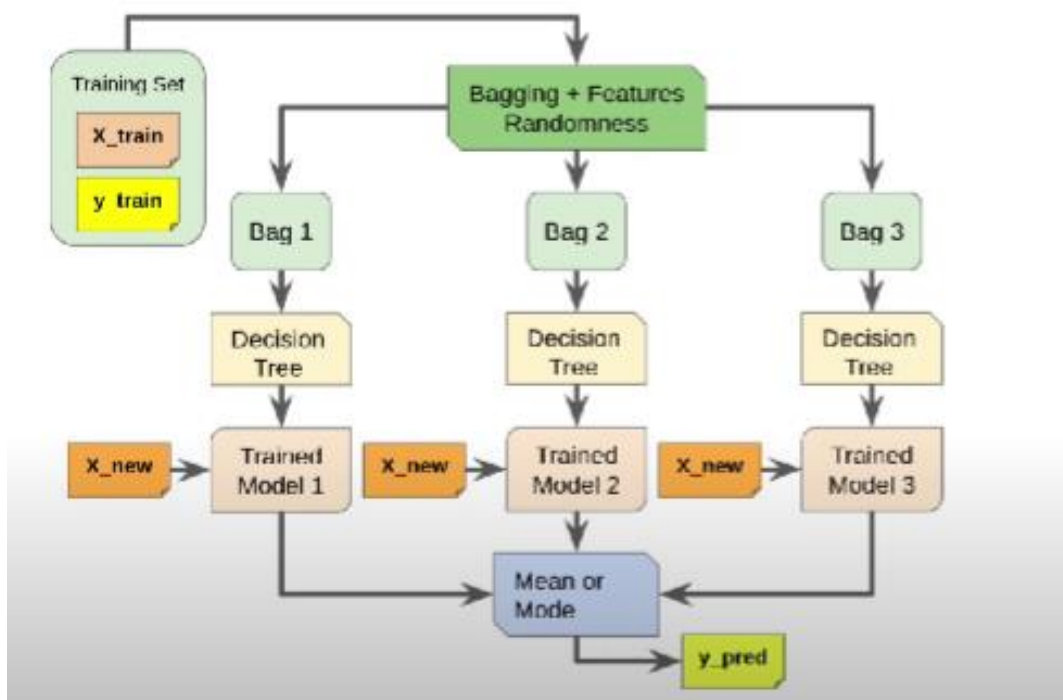


### 2.3. Pengenalan Bootstrap Aggregating | Bagging



### 2.4. Pengenalan Random Forest | Hutan Acak

Random Forest terbentuk dari sekumpulan decision tree atau pohon keputusan. Menerapkan Randomness karna setiap bag yang dihasilkan akan mengadopsi features yang secara acak dari training set sumbernya yang beragam dan akan menghasilkan trained yang berbeda juga, setiap train akan digunakan akan disatukan dan akan menjadi nilai prediksi final.



## 2.5. Persiapan dataset | Iris Flower Dataset

```
jupyter TP3_2 Last Checkpoint: 9 minutes ago
File Edit View Run Kernel Settings Help Trusted
+ % 
[1]: from sklearn.datasets import load_iris
X, y = load_iris(return_X_y=True)
print(f'Dimensi Featuter: {X.shape}')
print(f'Class: {set(y)}')
Dimensi Featuter: (150, 4)
Class: {0, 1, 2}

[3]: from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X,
                                                    y,
                                                    test_size=0.3,
                                                    random_state=0)
```

## 2.6. Implementasi Random Forest Classifier dengan Scikit Learn

```
[5]: from sklearn.ensemble import RandomForestClassifier
model = RandomForestClassifier(n_estimators=100,
                              random_state=0)
model.fit(X_train, y_train)

[5]: Random Forest Classifier
RandomForestClassifier(random_state=0)
```

## 2.7. Evaluasi model dengan Classification Report

```
[7]: from sklearn.metrics import classification_report
y_pred = model.predict(X_test)
print(classification_report(y_test, y_pred))
```

	precision	recall	f1-score	support
0	1.00	1.00	1.00	16
1	1.00	0.94	0.97	18
2	0.92	1.00	0.96	11
accuracy			0.98	45
macro avg	0.97	0.98	0.98	45
weighted avg	0.98	0.98	0.98	45