

Nama : Satya Athaya Daniswara

NIM : 1103213152

## **TUGAS PERBAIKAN BAB 7**

### **1. Pengantar Regresi**

- Regresi adalah salah satu tugas utama dalam machine learning yang bertujuan untuk memprediksi nilai kontinu berdasarkan data input.
- Penulis menjelaskan perbedaan antara regresi linier dan regresi non-linier, serta situasi di mana masing-masing digunakan.

### **2. Algoritma Regresi**

- Regresi Linier:
  - Model yang paling sederhana, di mana hubungan antara variabel independen (fitur) dan variabel dependen (target) diasumsikan linier.
  - Penulis menjelaskan bagaimana model ini bekerja, termasuk konsep slope (kemiringan) dan intercept (titik potong).
  - Contoh penggunaan regresi linier sederhana dan regresi linier berganda (dengan beberapa fitur).
- Regresi Polinomial:
  - Memperluas regresi linier dengan menambahkan fitur polinomial untuk menangkap hubungan non-linier.
  - Penulis menjelaskan bagaimana memilih derajat polinomial yang tepat dan risiko overfitting.
- Support Vector Regression (SVR):
  - Menggunakan prinsip yang sama dengan Support Vector Machines untuk klasifikasi, tetapi diterapkan pada masalah regresi.
  - Penulis menjelaskan konsep margin dan epsilon-insensitive loss.
- Decision Tree Regressor:
  - Model yang membagi data menjadi subset berdasarkan fitur, membentuk struktur pohon untuk memprediksi nilai kontinu.
  - Penulis menjelaskan cara kerja algoritma ini dan bagaimana menghindari overfitting dengan pruning.
- Random Forest Regressor:
  - Ensemble method yang menggabungkan beberapa decision trees untuk meningkatkan akurasi dan mengurangi variabilitas.

- Penulis menjelaskan cara kerja Random Forest dan keunggulannya dibandingkan dengan decision tree tunggal.

- Gradient Boosting Regressor:

- Metode ensemble yang membangun model secara bertahap, di mana setiap model baru berusaha memperbaiki kesalahan model sebelumnya.

- Penulis membahas algoritma seperti XGBoost yang populer dalam kompetisi data science.

### 3. Implementasi Regresi dengan Scikit-learn

- Penulis memberikan contoh kode untuk menerapkan berbagai algoritma regresi menggunakan pustaka Scikit-learn.

- Contoh mencakup:

- Memuat dataset (misalnya, dataset Boston Housing atau dataset lain yang relevan).

- Memisahkan data menjadi set pelatihan dan pengujian.

- Melatih model regresi.

- Membuat prediksi dan mengevaluasi kinerja model menggunakan metrik yang telah dibahas sebelumnya (seperti Mean Absolute Error, Mean Squared Error, dan  $R^2$  score).

### 4. Evaluasi Model Regresi

- Penulis menekankan pentingnya evaluasi model regresi dengan menggunakan metrik yang sesuai.

- Diskusi tentang penggunaan metrik seperti:

- Mean Absolute Error (MAE): Rata-rata dari selisih absolut antara nilai yang diprediksi dan nilai aktual.

- Mean Squared Error (MSE): Rata-rata dari kuadrat selisih antara nilai yang diprediksi dan nilai aktual.

- $R^2$  Score: Mengukur proporsi varians dalam variabel dependen yang dapat dijelaskan oleh variabel independen.

### 5. Tuning Model

- Penulis menjelaskan pentingnya tuning hyperparameter untuk meningkatkan kinerja model regresi.

- Contoh penggunaan Grid Search dan Randomized Search untuk menemukan kombinasi hyperparameter terbaik.

### 6. Studi Kasus

- Di akhir bab, penulis sering menyertakan studi kasus atau contoh praktis yang menunjukkan penerapan algoritma regresi pada dataset nyata.
- Ini memberikan konteks dan pemahaman yang lebih baik tentang bagaimana algoritma bekerja dalam situasi dunia nyata.

## 7. Kesimpulan

- Bab ini memberikan pemahaman yang komprehensif tentang berbagai algoritma regresi, cara menerapkannya, dan pentingnya evaluasi serta tuning model.
- Pembaca diharapkan dapat memilih algoritma yang tepat berdasarkan karakteristik data dan tujuan analisis.

### Contoh Kode Regresi

Berikut adalah contoh kode sederhana yang menunjukkan penerapan Regresi Linier untuk memprediksi harga rumah menggunakan dataset Boston Housing:

```
```python
Importing necessary libraries

import numpy as np
import pandas as pd

from sklearn.datasets import load_boston
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

Load the Boston Housing dataset

boston = load_boston()

X = boston.data
y = boston.target
```

Split the dataset into training and testing sets

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Create a Linear Regression model

```
lin_reg = LinearRegression()
```

Fit the model to the training data

```
lin_reg.fit(X_train, y_train)
```

Make predictions on the test data

```
y_pred = lin_reg.predict(X_test)
```

Evaluate the model

```
print("Mean Absolute Error:", mean_absolute_error(y_test, y_pred))
```

```
print("Mean Squared Error:", mean_squared_error(y_test, y_pred))
```

```
print("R2 Score:", r2_score(y_test, y_pred))
```