Regresi Logistik Berganda (Multiple Logistic Regression)

Setelah mempelajari regresi linear berganda dan regresi logistik sederhana, sekarang kita akan menggabungkan kedua konsep tersebut dalam regresi logistik berganda

# Regresi Logistik Berganda

Regresi logistik berganda adalah perluasan dari regresi sederhana yang memungkinkan kita menggunakan lebih dari satu variabel independen (prediktor) untuk memprediksi variabel dependen biner (dua kemungkinan hasil).

# Persamaan Regresi Logistik Berganda

log(p/(1-p)) = b0 + b1x1 + b2x2 + ... + bnxn

di mana:

**p**: probabilitas peristiwa terjadi

**x1, x2, …, xn**: variabel independen

**b0**: intercept

**b1, b2, …, bn**: koefisien regresi

# Interpretasi Koefisien Regresi

Koefisien regresi dalam logistik berganda memiliki interpretasi yang sedikit berbeda dengan regresi linear berganda. Pada regresi logistik berganda, koefisien regresi menunjukkan perubahan log-odds dari variabel dependen ketika variabel independen meningkat satu unit, **dengan asumsi semua variabel independen lainnya tetap konstan.**

# Penerapan Regresi Logistik Berganda dengan Scikit-Learn

Kita dapat menggunakan LogisticRegression dari scikit-learn untuk membangun model regresi logistik berganda.

**Tugas Hari 23:**

Menggunakan dataset “Social Network Ads”

**Tugas**

1. **Membuat Model Regresi Logistik Berganda:**

* Gunakan LogisticRegression dari scikit-learn untuk membuat model.
* Gunakan Age, EstimatedSalary, dan Gender sebagai fitur, dan Purchased sebagai target.
* Latih model pada data latih.

1. **Prediksi dan Evaluasi:**

* Gunakan model untuk memprediksi label kelas pada data uji.
* Hitung akurasi, presisi, recall, dan F1-score dari model pada data uji.

1. **Interpretasi Koefisien:**

* Tampilkan koefisien regresi dari model.
* Interpretasikan arti dari setiap koefisien regresi.

**Contoh Kode (Scikit-Learn dan Matplotlib)**

# ... (kode untuk membaca dan mempersiapkan dataset Social Network Ads)

# Encoding fitur kategorikal Gender

df = pd.get\_dummies(df, columns=['Gender'], drop\_first=True)

# ... (kode untuk membagi data menjadi data latih dan data uji)

# ... (kode standardisasi fitur)

# Membuat model regresi logistik

model = LogisticRegression()

# Melatih model

model.fit(X\_train, y\_train)

# Prediksi dan evaluasi

y\_pred = model.predict(X\_test)

# ... (hitung akurasi, presisi, recall, dan F1-score)

# Tampilkan koefisien

print("Intercept:", model.intercept\_[0])

print("Koefisien:")

for feature, coef in zip(X.columns, model.coef\_[0]):

print(f"{feature}: {coef:.4f}")

**Catatan:**

* pastikan untuk menyesuaikan kode dengan dataset “Social Network Ads”
* Pastikan juga melakukan standarisasi fitur sebelum melatih model.

**Selamat Mengerjakan Tugas! 😂**

**Tugas:**

1. **Membuat Model Regresi Logistik Berganda:**

**# Encoding Fitur Kategorikal Gender**

**df = pd.get\_dummies(df, columns=['Gender'], drop\_first=True)**

**# Membagi Data Menjadi Data Latih dan Data Uji**

**# Standardisasi Fitur**

**# Membuat Model Regresi Logistik**

**model = LogisticRegression()**

**# Melatih Model**

**model.fit(X\_train, y\_train)**

1. **Prediksi dan Evaluasi:**

**y\_pred = model.predict(X\_test)**

**# Hitung Akurasi, Presisi, Recall, dan F1-score**

1. **Interpretasi Koefisien:**

**# Tampilkan Koefisien**

**print("Intercept: ", model.intercept\_[0])**

**print("Koefisien:")**

**for feature, coef in zip(X.columns, model.coef\_[0]):**

**print(f"{feature}: {coef:.4f}")**

**Penjelasan Kode:**

* **Encoding Gender:**
  + **pd.get\_dummies(X, columns=['Gender'], drop\_first=True): Melakukan one-hot encoding pada kolom Gender. Argument drop\_first=True digunakan untuk menghindari multikolinearitas dengan menghapus salah satu kolom dummy yang dihasilkan.**
* **Train-test split:**
  + **X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42): Membagi data menjadi 80% data latih dan 20% data uji.**
* **Standardisasi Fitur:**
  + **scaler = StandardScaler(): Membuat objek StandardScaler.**
  + **X\_train = scaler.fit\_transform(X\_train): Melakukan fitting dan transformasi pada data latih.**
  + **X\_test = scaler.transform(X\_test): Melakukan transformasi pada data uji dengan parameter yang sama dari data latih.**
* **Evaluasi Model:**
  + **Menghitung akurasi, presisi, recall, dan F1-score menggunakan fungsi-fungsi dari sklearn.metrics.**
* **Interpretasi Koefisien:**
  + **Mencetak intercept dan koefisien regresi untuk setiap fitur. Koefisien ini menunjukkan perubahan log-odds dari variabel dependen (Purchased) ketika variabel independen meningkat satu unit, dengan asumsi variabel independen lainnya tetap konstan.**

**Contoh Interpretasi Koefisien:**

**Jika koefisien untuk Age adalah 0.5, berarti setiap peningkatan satu tahun usia akan meningkatkan log-odds pembelian produk sebesar 0.5, dengan asumsi EstimatedSalary dan Gender\_Male tetap.**