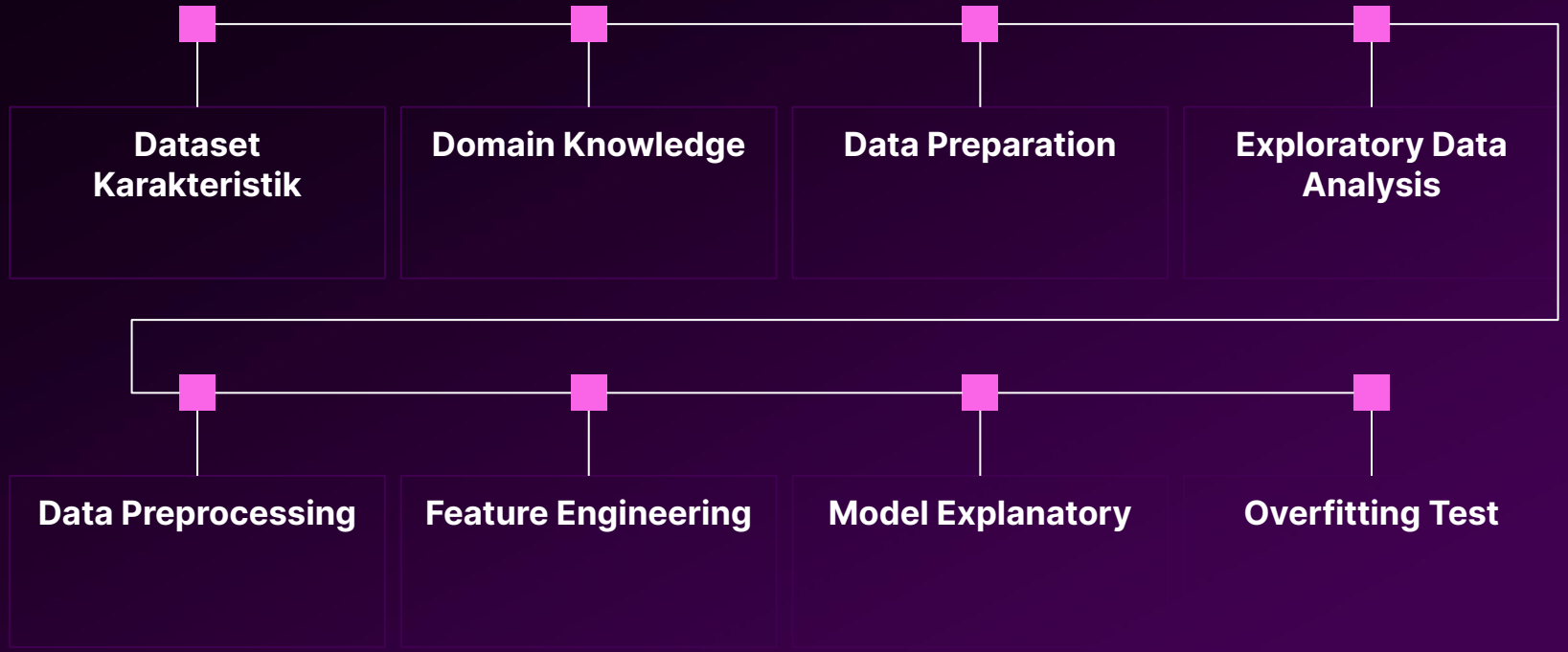
DIABETES PREDICTION

Presentation by

I Gusti Ayu Meliniarayani

PROJECT OVERVIEW



Background

Diabetes mellitus, yang sering disebut diabetes, adalah kelompok penyakit yang memengaruhi cara tubuh menggunakan gula (glukosa) sebagai sumber energi. Penyakit ini adalah masalah kesehatan global yang semakin umum dan berdampak besar pada individu, masyarakat, dan sistem perawatan kesehatan.

*Dataset ini berasal dari **National Institute of Diabetes and Digestive and Kidney Diseases.** Tujuan dari dataset ini adalah untuk memprediksi secara diagnostik apakah seorang pasien menderita diabetes atau tidak.*

Dataset Karakteristik

Secara khusus, **semua pasien di sini adalah perempuan yang berusia setidaknya 21 tahun dan keturunan Indian Pima**.

Keterangan Outcome :
1 diabetes, 0 tidak diabetes

Saya membangun sebuah **Mechine Learning Model** untuk memprediksi apakah pasien dalam dataset memiliki diabetes atau tidak.

- **Pregnancies** : Jumlah kehamilan
- **Glucose** : Konsentrasi glukosa dalam plasma 2 jam setelah tes toleransi glukosa oral
- **BloodPressure** : Tekanan darah diastolik (mm Hg)
- **SkinThickness** : Ketebalan lipatan kulit trisep (mm)
- **Insulin** : Insulin serum 2 jam (mu U/ml)
- **BMI** : Indeks massa tubuh (berat dalam kg/(tinggi dalam m)^2)
- **DiabetesPedigreeFunction** : Fungsi garis keturunan diabetes
- **Age** : Usia (tahun)
- **Outcome** : Variabel kelas (0 atau 1), 268 dari 768 adalah 1, yang lainnya adalah 0.

Domain Knowledge

Toleransi Glukosa

- Hasil Normal untuk Diabetes → Tingkat glukosa dua jam kurang dari 140 mg/dL
- Hasil Terganggu untuk Diabetes → Tingkat glukosa dua jam 140 hingga 200 mg/dL
- Hasil Abnormal (Diagnostik) untuk Diabetes → Tingkat glukosa dua jam lebih dari 200 mg/dL

BMI

- Kurang dari 18,5 → Kurang Berat Badan
- 18,5 - 24,9 → Berat Badan Normal atau Sehat
- 25,0 - 29,9 → Kelebihan Berat Badan
- 30,0 atau Lebih → Obesitas

Tekanan Darah

- Normal : Sistolik di bawah 120 dan diastolik di bawah 80
- Batas Normal : Sistolik 120–129 dan diastolik di bawah 80
- Hipertensi tahap 1: Sistolik 130–139 dan diastolik 80–89
- Hipertensi tahap 2: Sistolik 140 atau lebih dan diastolik 90 atau lebih
- Krisis hipertensi: Sistolik lebih tinggi dari 180 dan diastolik di atas 120

Triceps Skinfolds

Untuk orang dewasa, nilai normal standar untuk lipatan kulit trisep adalah:

18,0 mm (wanita)

Data Preparation

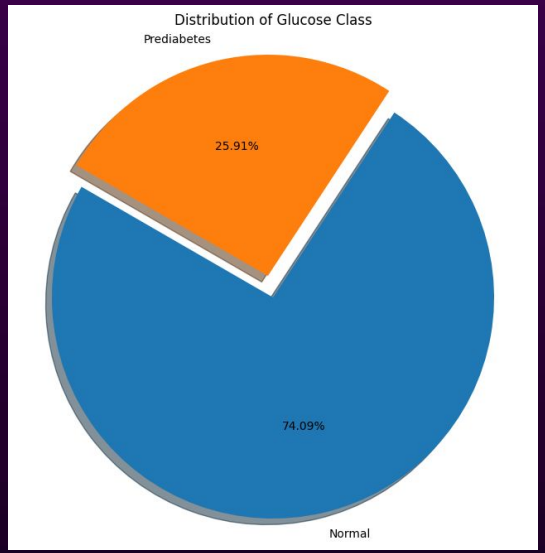
01 Load Dataset

02 Check Duplicate Values

03 Check Missing Values

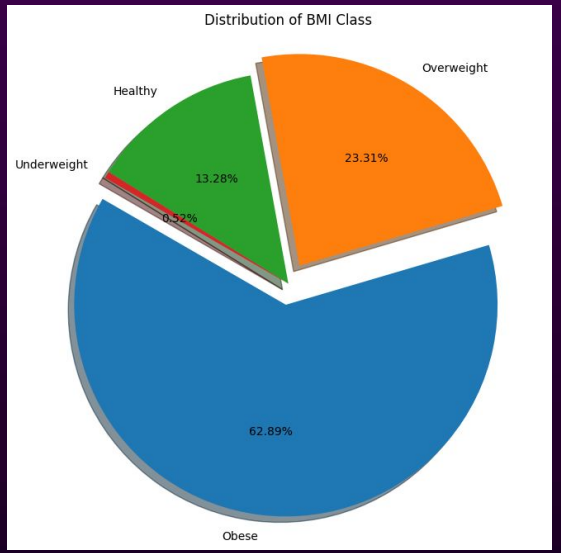
EDA

Distribution Glucose Level



74.09% orang memiliki kadar Glucose level Normal dan 25,91% memiliki kadar Glucose level Prediabetes

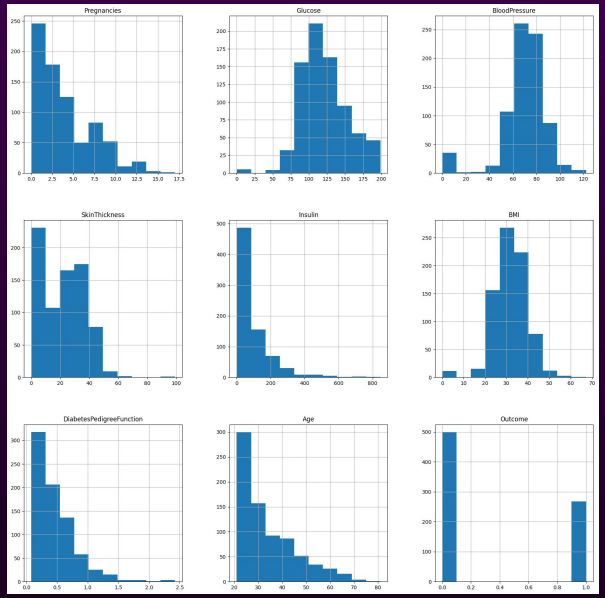
Distribution BMI Range



orang-orang pada dataset ini cenderung memiliki berat badan diatas BMI. Persentase orang yang obese adalah 62.98% dan Overweight 23.31%. Healthy 13.28% dan Underweight 0.52%

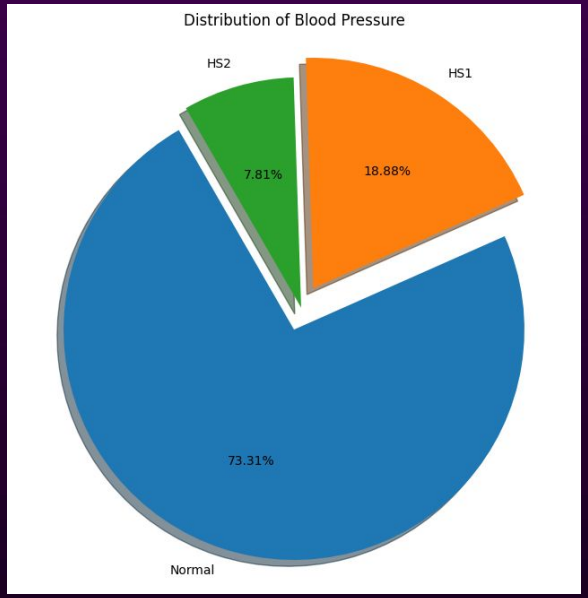
EDA

Data Distribution



Dari grafik histogram kita bisa melihat hampir semua data pada dataset berdistribusi tidak normal atau Skewed

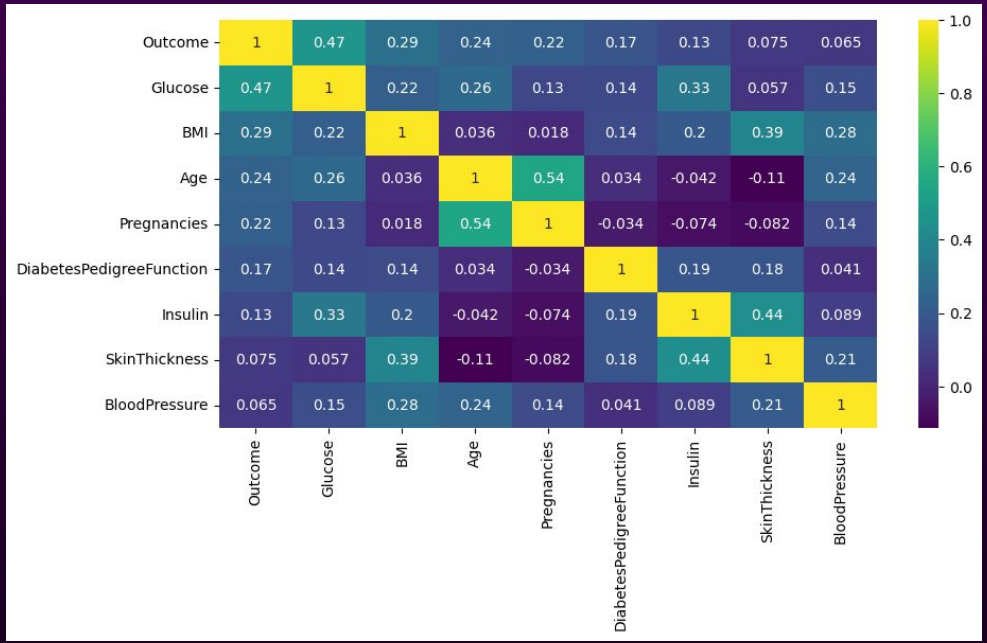
Distribution Tekanan Darah



Persentase Tekanan darah pada dataset ini didominasi oleh tekanan darah Normal 73.31%, Hipertensi awal 18.88% dan Hipertensi 7.81%

EDA

Correlation Matrix



Dari grafik hubungan antara variable dengan Outcome, Glucose (0.47), BMI (0.29), Age (0.24), Pregnancies (0.22), Diabetes Pedigree Function (0.17) dan Insulin (0.13) yang memiliki hubungan kolerasi yg cukup kuat.

Data Preprocessing

01 Fix Missing Value

```
# We can fill in NaN values with a median according to the target

for col in df.columns:
    outcome_0_median = df[df["Outcome"]==0][col].median()
    outcome_1_median = df[df["Outcome"]==1][col].median()
    df.loc[(df["Outcome"]==0) & (df[col].isnull()), col] = outcome_0_median
    df.loc[(df["Outcome"]==1) & (df[col].isnull()), col] = outcome_1_median
```

Column dengan Missing Value : Glucose, BloodPressure, SkinThickness, Insulin dan BMI

02 Fix Outlier

```
def replace_with_thresholds(dataframe, numeric_columns):
    for variable in numeric_columns:
        low_limit, up_limit = outlier_thresholds(dataframe, variable)
        dataframe.loc[(dataframe[variable] < low_limit), variable] = low_limit
        dataframe.loc[(dataframe[variable] > up_limit), variable] = up_limit
```

Menekan angka outlier ke IQR

Feature Engineering

01 Encoded dengan one hot Encoding

New_Glucose_Class_Prediabetes	New_BMI_Range_Healthy	New_BMI_Range_Overweight	New_BMI_Range_Obese	New_BloodPressure_HS1
1	0	0	1	0
0	0	1	0	0
1	1	0	0	0
0	0	1	0	0
0	0	0	1	0

Yang dimana dibuat Column baru : Glucose, BloodPressure, SkinThickness dan BMI yg berisi kategori dari range nilai threshold

02 Scalling dengan Robust

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	0.6	0.765432	0.000	1.000000	1.000000	0.170330	0.665359	1.235294	1.0
1	-0.4	-0.790123	-0.375	0.142857	0.000000	-0.598901	-0.056209	0.117647	0.0
2	1.0	1.629630	-0.500	0.571429	1.000000	-0.961538	0.783007	0.176471	1.0
3	-0.4	-0.691358	-0.375	-0.714286	-0.126866	-0.434066	-0.537255	-0.470588	0.0
4	-0.6	0.493827	-2.000	1.000000	0.977612	1.214286	4.121569	0.235294	1.0

Column yg discalling : Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin', BMI, DiabetesPedigreeFunction dan Age

Model Explanatory

	Model_Type	avg_accuracy	std_accuracy
0	LR	0.770728	0.058473
1	KNN	0.833442	0.038106
2	CART	0.850393	0.045907
3	RF	0.875051	0.027793
4	SVR	0.843797	0.033670
5	XGBM	0.882946	0.037853
6	GB	0.890755	0.037074
7	LightGBM	0.885543	0.035007

Before Tuning

Membandingkan beberapa model yg memenuhi nilai Recall dan Accuracy yg baik. Dari perbandingan model ini, saya memilih **LightGBM Classifier** sebagai Best Model kerena memiliki nilai Recall yang cukup optimal dan accuracy yg tinggi serta standard devisiasi yang rendah menunjukan kerja model yg stabil.

Untuk model terbaik yg terpilih. Saya juga melakukan pengecekan dengan model **RF, GB dan XGBM** sebagai perbandingan model terbaik setelah di tuning.

	Model_Type	Mean_Accuracy	Std_Accuracy	Mean_Recall	Std_Recall
0	RF	0.881630	0.029036	0.838454	0.051169
1	GBM	0.890755	0.032807	0.804725	0.043705
2	LightGBM	0.888124	0.029479	0.819827	0.054489
3	XGB	0.875103	0.032691	0.804631	0.057841

After tuning

Overfitting Test

	Model	Train_score	Test_score	Train_Recall	Test_Recall
0	Random Forest	0.895664	1.000000	1.000000	1.000000
1	LightGBM	0.903834	0.980519	0.995305	0.963636
2	XGBM	0.905473	1.000000	0.995305	1.000000
3	GB	0.893998	0.993506	0.981221	0.981818

Best Model LightGBM : nilai Mean Accuracy 0.888 sudah cukup baik untuk mendapatkan accuracy predict yg nanti akan dilakukan dengan nilai recall 0.819 ditambah dengan nilai Std Accuracy yang tidak terlalu tinggi (cukup rendah) menggambarkan model dapat konsisten dalam kinerjanya. Serta tidak mengalami Overfitting yang ekstream 3% untuk Recall dan 8% untuk Accuracy

THANKS!

Do you have any question?

igameliniarayani13@gmail.com

https://github.com/igamelinia

Linkedin
www.linkedin.com/in/igustiayumeliniarayani/