Stroke Prediction

Presentation by I Gusti Ayu Meliniarayani

Overview

Background	Dataset Characteristic	Data Preparation
EDA	Data Preprocessing	Feature Engineering
Praprocess Modelling	Model Explanatory	Modelling
Overfitting Test	Save Model to Pickle	Conclucion

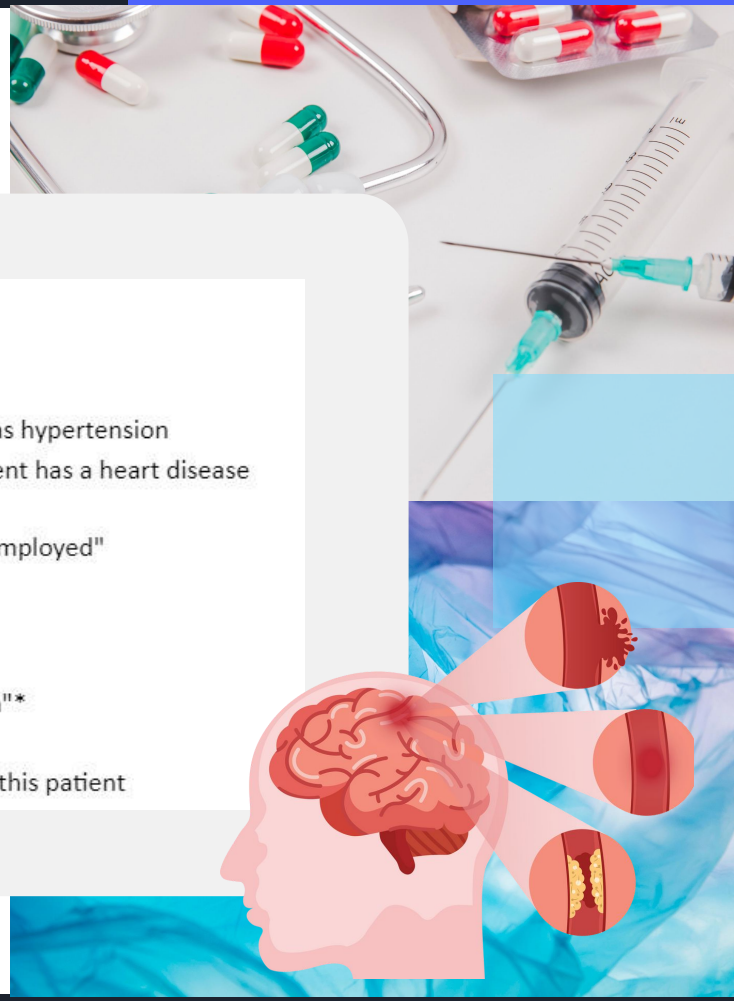
Background

Menurut Organisasi Kesehatan Dunia (WHO), stroke adalah penyebab kematian kedua terbesar di dunia, bertanggung jawab atas sekitar 11% dari total kematian. Penelitian ini bertujuan untuk memprediksi apakah seorang pasien berpotensi mengalami stroke berdasarkan pada parameter masukan / dataset yang tersedia.

Dataset Characteristic

1) id : unique identifier
2) gender : "Male", "Female" or "Other"
3) age : age of the patient
4) hypertension : 0 if the patient doesn't have hypertension, 1 if the patient has hypertension
5) heart_disease : 0 if the patient doesn't have any heart diseases, 1 if the patient has a heart disease
6) ever_married : "No" or "Yes"
7) work_type : "children", "Govt_jov", "Never_worked", "Private" or "Self-employed"
8) Residence_type : "Rural" or "Urban"
9) avg_glucose_level : average glucose level in blood
10) bmi : body mass index
11) smoking_status : "formerly smoked", "never smoked", "smokes" or "Unknown"*
12) stroke : 1 if the patient had a stroke or 0 if not

*Note: "Unknown" in smoking_status means that the information is unavailable for this patient



Data Preparation

1. Load Dataset

```
df = pd.read_csv('healthcare-dataset-stroke-data.csv')
df
```

index	id	gender	age	hypertension	heart_disease	ever_married	work_type	Residence_type	avg_glucose_level	bmi	smoking_status	stroke
0	9046	Male	67.0	0	1	Yes	Private	Urban	228.69	36.6	formerly smoked	1
1	51676	Female	61.0	0	0	Yes	Self-employed	Rural	202.21	NaN	never smoked	1
2	31112	Male	80.0	0	1	Yes	Private	Rural	105.92	32.5	never smoked	1
3	60182	Female	49.0	0	0	Yes	Private	Urban	171.23	34.4	smokes	1
4	1665	Female	79.0	1	0	Yes	Self-employed	Rural	174.12	24.0	never smoked	1
5	56669	Male	81.0	0	0	Yes	Private	Urban	186.21	29.0	formerly smoked	1
6	53882	Male	74.0	1	1	Yes	Private	Rural	70.09	27.4	never smoked	1
7	10434	Female	69.0	0	0	No	Private	Urban	94.39	22.8	never smoked	1
8	27419	Female	59.0	0	0	Yes	Private	Rural	76.15	NaN	Unknown	1
9	60491	Female	78.0	0	0	Yes	Private	Urban	58.57	24.2	Unknown	1
10	12109	Female	81.0	1	0	Yes	Private	Rural	80.43	29.7	never smoked	1
11	12095	Female	61.0	0	1	Yes	Govt_job	Rural	120.46	36.8	smokes	1
12	12175	Female	54.0	0	0	Yes	Private	Urban	104.51	27.3	smokes	1
13	8213	Male	78.0	0	1	Yes	Private	Urban	219.84	NaN	Unknown	1
14	5317	Female	79.0	0	1	Yes	Private	Urban	214.09	28.2	never smoked	1
15	58202	Female	50.0	1	0	Yes	Self-employed	Rural	167.41	30.9	never smoked	1
16	56112	Male	64.0	0	1	Yes	Private	Urban	191.61	37.5	smokes	1
17	34120	Male	75.0	1	0	Yes	Private	Urban	221.29	25.8	smokes	1
18	27458	Female	60.0	0	0	No	Private	Urban	89.22	37.8	never smoked	1
19	25226	Male	57.0	0	1	No	Govt_job	Urban	217.08	NaN	Unknown	1
20	70630	Female	71.0	0	0	Yes	Govt_job	Rural	193.94	22.4	smokes	1
21	13861	Female	52.0	1	0	Yes	Self-employed	Urban	233.29	48.9	never smoked	1
22	68794	Female	79.0	0	0	Yes	Self-employed	Urban	228.7	26.6	never smoked	1
23	64778	Male	82.0	0	1	Yes	Private	Rural	208.3	32.5	Unknown	1
24	4219	Male	71.0	0	0	Yes	Private	Urban	102.87	27.2	formerly smoked	1

Data Preparation

2. Check overview dataset

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5110 entries, 0 to 5109
Data columns (total 12 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   id                 5110 non-null   int64
 1   gender             5110 non-null   object
 2   age                5110 non-null   float64
 3   hypertension       5110 non-null   int64
 4   heart_disease      5110 non-null   int64
 5   ever_married       5110 non-null   object
 6   work_type          5110 non-null   object
 7   Residence_type     5110 non-null   object
 8   avg_glucose_level  5110 non-null   float64
 9   bmi                4909 non-null   float64
 10  smoking_status     5110 non-null   object
 11  stroke             5110 non-null   int64
dtypes: float64(3), int64(4), object(5)
memory usage: 479.2+ KB
```

```
df.describe()
```

	id	age	hypertension	heart_disease	avg_glucose_level	bmi	stroke
count	5110.000000	5110.000000	5110.000000	5110.000000	5110.000000	4909.000000	5110.000000
mean	36517.829354	43.226614	0.097456	0.054012	106.147677	28.893237	0.048728
std	21161.721625	22.612647	0.296607	0.226063	45.283560	7.854067	0.215320
min	67.000000	0.080000	0.000000	0.000000	55.120000	10.300000	0.000000
25%	17741.250000	25.000000	0.000000	0.000000	77.245000	23.500000	0.000000
50%	36932.000000	45.000000	0.000000	0.000000	91.885000	28.100000	0.000000
75%	54682.000000	61.000000	0.000000	0.000000	114.090000	33.100000	0.000000
max	72940.000000	82.000000	1.000000	1.000000	271.740000	97.600000	1.000000

```
df['stroke'].value_counts()
```

```
0    4861
1     249
Name: stroke, dtype: int64
```

Terjadi **imbalance dataset**

Data Preparation

3. Check Missing & Duplicated Values

```
df.isna().sum()
```

```
id                   0
gender               0
age                  0
hypertension         0
heart_disease        0
ever_married         0
work_type            0
Residence_type       0
avg_glucose_level    0
bmi                201
smoking_status       0
stroke               0
dtype: int64
```

```
df.duplicated().sum()
```

```
0
```

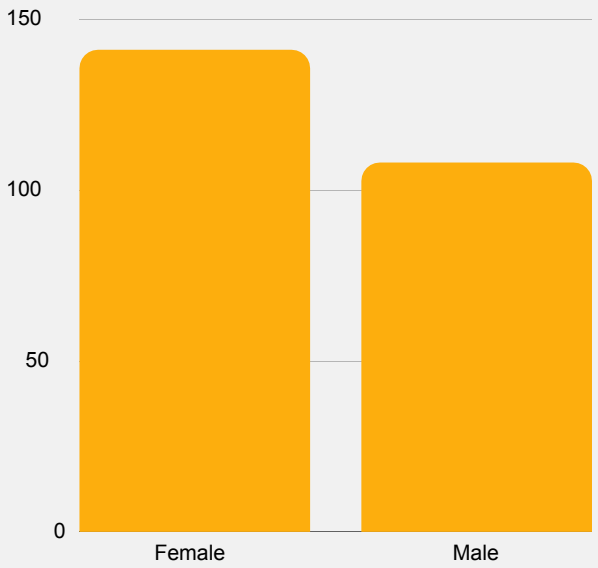
Tidak terdapat data yg duplikat

```
rows_with_different_ids = df[df['id'].duplicated(keep=False)]
print(rows_with_different_ids)
```

```
Empty DataFrame
Columns: [id, gender, age, hypertension, heart_disease, ever_married, work_type, Residence_type, avg_glucose_level, bmi, smoking_status, stroke]
Index: []
```

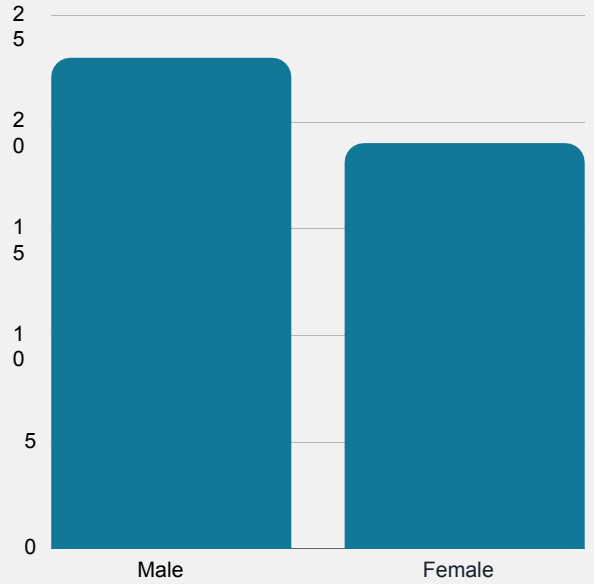
Terdapat **Missing Values pada kolom BMI 3.9%**

EDA



Stroke Count by Gender

jumlah kasus stroke tertinggi dengan gender "Female" sebanyak 141 (56.63%) kasus dibandingkan dengan "Male" berjumlah 108(43.37%) kasus

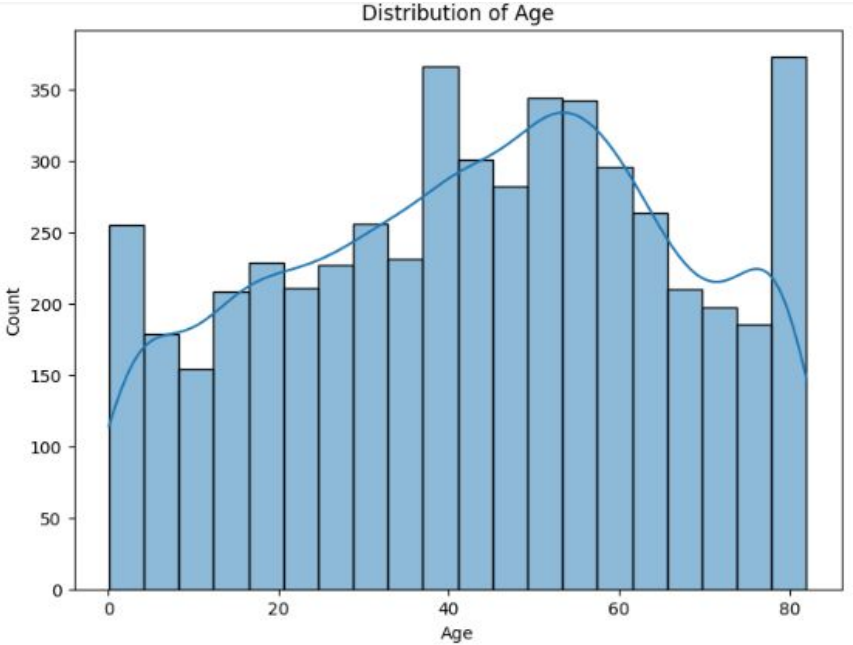


Stroke counts for Smokers by genders

Data menunjukkan bahwa jumlah kasus stroke pada individu yang merokok adalah Laki-laki (Male) memiliki 23 kasus stroke, dan Perempuan (Female) memiliki 19 kasus stroke

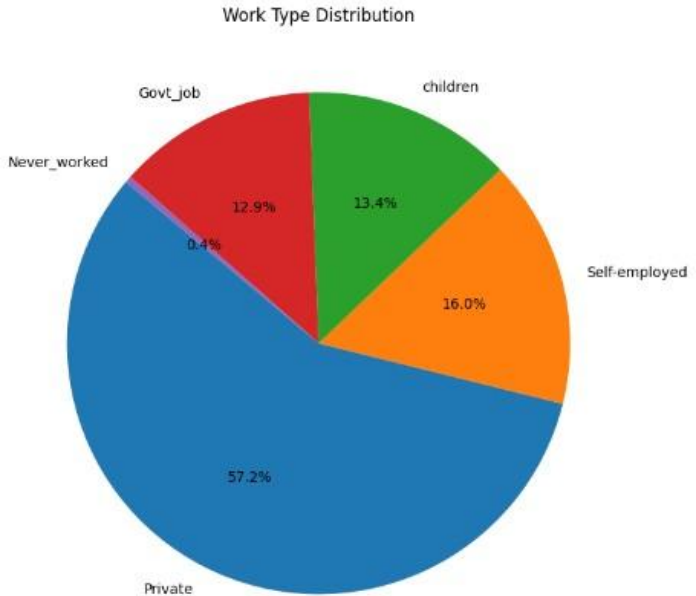
EDA

Distribution of Age



Rata-rata usia responden adalah sekitar 43 tahun, dengan rentang usia antara 0.08 hingga 82 tahun. Dengan median usia sekitar 45 tahun, distribusi usia cenderung simetris

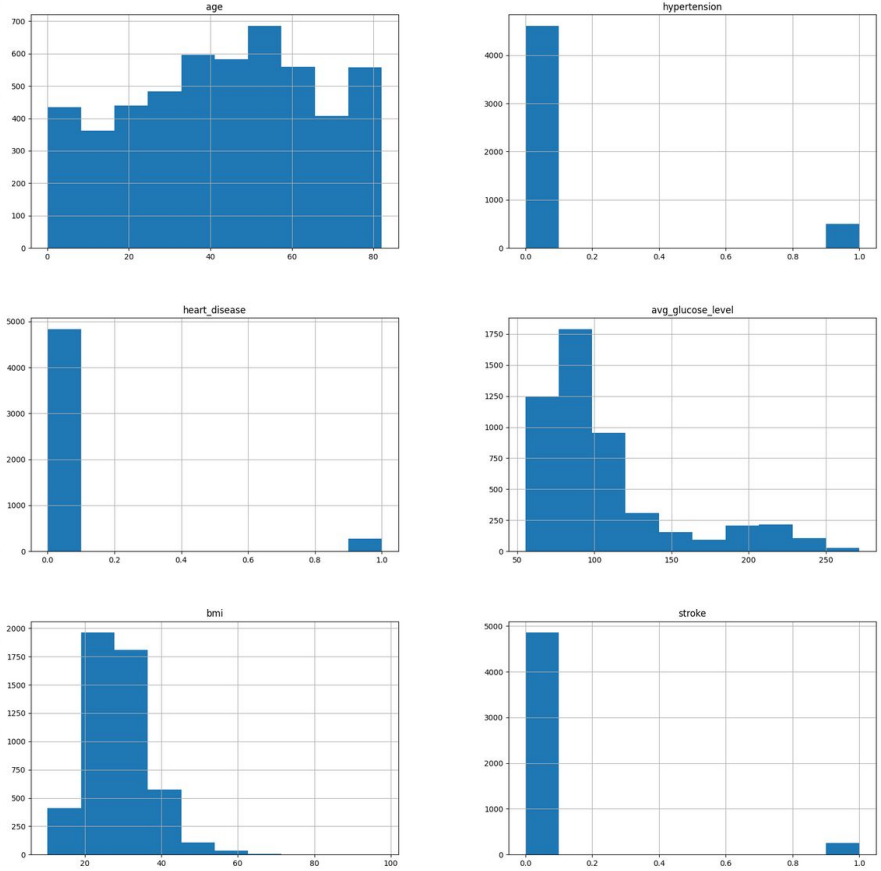
Work Type Distribution



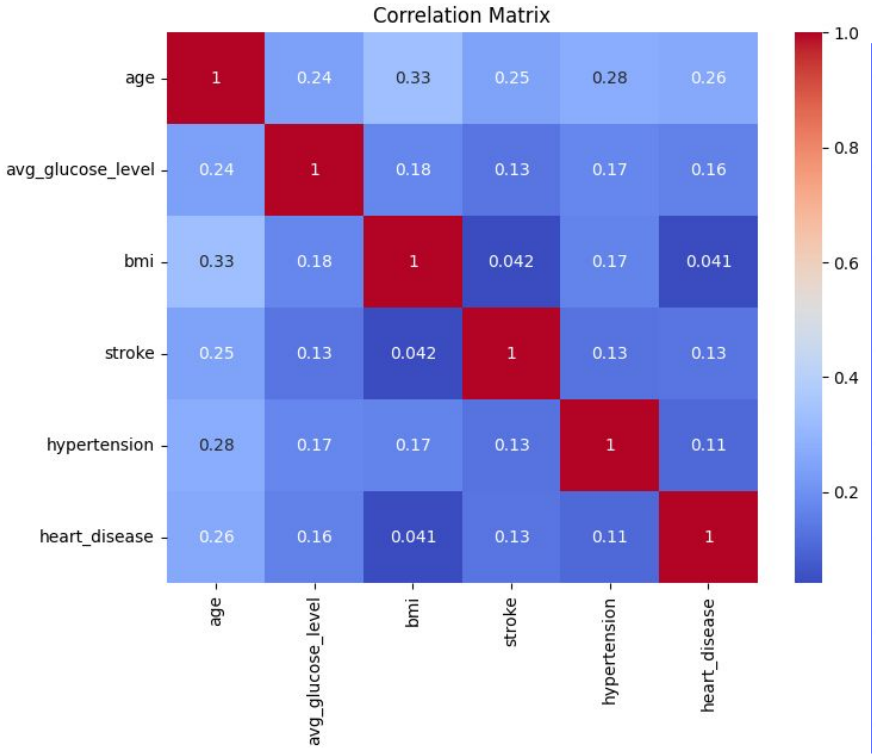
Data menunjukkan bahwa 57.2% orang bekerja sebagai pegawai swasta dan 0.4% yang tidak bekerja pada dataset ini

EDA

Data Distribution



Correlation Matrix



Stroke dengan Age, Hipertensi, Heart Disease dan Glucose Level memiliki hubungan yg cukup kuat untuk di dataset ini

Data Preprocessing

Fix Missing Values

```
# We can fill in NaN values with a median according to the target
stroke_0_median = df[df["stroke"] == 0]["bmi"].median()
stroke_1_median = df[df["stroke"] == 1]["bmi"].median()

df.loc[(df["stroke"] == 0) & (df["bmi"].isnull()), "bmi"] = stroke_0_median
df.loc[(df["stroke"] == 1) & (df["bmi"].isnull()), "bmi"] = stroke_1_median
```

```
] df.columns.isnull().sum()
```

```
0
```

Fix Outliers

```
#Menanggulangi Outlier dengan menekan nilai outliers
quartile1 = df['bmi'].quantile(0.10)
quartile3 = df['bmi'].quantile(0.90)
interquartile_range = quartile3 - quartile1
up_limit = quartile3 + 1.5 * interquartile_range
low_limit = quartile1 - 1.5 * interquartile_range

df.loc[(df['bmi'] < low_limit), 'bmi'] = low_limit
df.loc[(df['bmi'] > up_limit), 'bmi'] = up_limit
```

Feature Engineering

One Hot Encoding

```python
#Membuat fungsi Encoding
def one_hot_encode(dataframe, columns, drop_first=False, prefix_sep='_'):
    """
    Melakukan one-hot encoding pada kolom kategorikal dalam DataFrame.

    Args:
        dataframe (DataFrame): DataFrame yang akan diencode.
        columns (list): Daftar nama kolom kategorikal yang akan diencode.
        drop_first (bool): Jika True, akan menghapus kolom pertama dalam setiap kolom yang di-encode.
        prefix_sep (str): Pemisah untuk menambahkan prefiks pada nama kolom yang di-encode.

    Returns:
        DataFrame: DataFrame yang telah diencode dengan kolom-kolom baru.
    """
    df_encoded = pd.get_dummies(
        dataframe,
        columns=columns,
        drop_first=drop_first,
        prefix=columns,
        prefix_sep=prefix_sep)
    return df_encoded
```

Robust Scalling

```python
#Membuat fungsi Scalling
def robust_scale(dataframe, numeric_columns):
    """Melakukan robust scaling pada kolom numerik dalam DataFrame.

    Args:
        dataframe (DataFrame): DataFrame yang akan di-scale secara robust.
        numeric_columns (list): Daftar nama kolom numerik yang akan di-scale.

    Returns:
        DataFrame: DataFrame yang telah di-scale secara robust.
    """
    scaler = RobustScaler()
    dataframe[numeric_columns] = scaler.fit_transform(dataframe[numeric_columns])
    return dataframe
```

Praprocessing Modelling

Spliting Dataset

```
X = df.drop("stroke", axis=1)
y = df["stroke"]

# Split train-test data
X_train, X_test, y_train, y_test = train_test_split(X,
                                                    y,
                                                    test_size=0.20,
                                                    stratify=y,
                                                    random_state=42)
```

Imbalance Dataset SMOTEENN

```
# Define oversampling strategy
oversample = SMOTEENN()

# Fit and apply the transform
X_train_over, y_train_over = oversample.fit_resample(X_train, y_train)
```

```
y_train_over.value_counts(normalize=True)
```

```
1    0.53763
0    0.46237
Name: stroke, dtype: float64
```

SMOTENN adalah teknik resampling yang mengatasi ketidakseimbangan data dengan membuat sampel baru untuk kelas minoritas (SMOTE) dan kemudian membersihkan sampel sintetis yang dihasilkan (ENN) untuk menjaga kualitasnya.

Model Explanatory

Membandingkan beberapa model untuk menemukan model yang memberikan Recall terbaik. Berdasarkan perbandingan model, saya memilih **Random Forest Classifier** karena memberikan Recall terbaik serta metric lainnya.

Untuk memilih model terbaik, saya juga memeriksa model KNN, XGBM, dan LightGBM untuk dibandingkan.

	Model_Type	Mean_Accuracy	Mean_Recall	Mean_Precision	Mean_F1_Score
0	LR	0.883798	0.890541	0.893038	0.891713
1	KNN	0.966985	0.991646	0.949150	0.969912
2	CART	0.931083	0.946966	0.925352	0.936766
3	RF	0.978807	0.991721	0.970211	0.981204
4	SVR	0.910468	0.943153	0.895836	0.918855
5	XGBM	0.973185	0.985507	0.965277	0.975278
6	GB	0.920126	0.961683	0.897139	0.928261
7	LightGBM	0.972750	0.984449	0.965584	0.974913

Overfitting Test

Before Tuning

	Model_Type	Recall_Train	Accuracy_Train	Precision_Train	F1-Score_Train	Recall_Test	Accuracy_Test	Precision_Test	F1-Score_Test
0	KNN	0.994637	0.983997	0.976053	0.985257	0.46	0.792564	0.110577	0.178295
1	RF	1.000000	1.000000	1.000000	1.000000	0.40	0.853229	0.142857	0.210526
2	XGB	1.000000	1.000000	1.000000	1.000000	0.30	0.857143	0.119048	0.170455
3	LGBM	0.996782	0.995963	0.995714	0.996248	0.36	0.862035	0.141732	0.203390

Terjadi overfit yg signifikan
Perlu dilakukan penanggulangan
Dengan Hyperparameter Tuning

Random Forest Classifier After Tuning

```
Recall on Training Data (Best RF Model): 0.9557522123893806
Accuracy on Training Data (Best RF Model): 0.8533737024221453
Precision on Training Data (Best RF Model): 0.8070652173913043
F1-Score on Training Data (Best RF Model): 0.8751381215469614
Recall on Testing Data (Best RF Model): 0.82
Accuracy on Testing Data (Best RF Model): 0.6418786692759295
Precision on Testing Data (Best RF Model): 0.10301507537688442
F1-Score on Testing Data (Best RF Model): 0.1830357142857143
```

Fokus pada RF Classifier Score :
Training Recall 0.95
Test Recall 0.82

42% perfoma model meningkat
47% penurunan overfit

Save Model

```
with open('rf_tuned_model.pkl', "wb") as model_file:
    pickle.dump(best_rf_model, model_file)
```

Save model RF Classifier yang telah dituning dengan menggunakan pickle

Conclusion

- **Kolerasi antara Stroke dengan Age, Hipertensi, Heart Disease dan Glucose Level memiliki hubungan yg cukup kuat**
- **age distribution** : Rata-rata usia responden adalah sekitar 43 tahun, dengan rentang usia antara 0.08 hingga 82 tahun. Dengan median usia sekitar 45 tahun, distribusi usia cenderung simetris.
- **work type distribution** : Dalam sampel atau populasi yang direpresentasikan oleh pie chart, mayoritas individu (57%) bekerja di sektor swasta (private), yang menjadikannya tipe pekerjaan yang paling umum, dan Fakta bahwa sekitar 13.4% individu merupakan anak-anak menunjukkan bahwa sampel atau populasi juga mencakup anak-anak yang belum masuk ke dalam angkatan kerja. Meskipun hanya sekitar 0.4%, ada individu yang tidak pernah bekerja dalam populasi ini. Ini bisa mencakup individu yang masih belajar atau yang belum memulai karier mereka.
- **the gender that suffers the most strokes** : jumlah kasus stroke lebih tinggi pada individu dengan gender "Female" (Perempuan) dengan jumlah 141(56.63%) kasus dibandingkan dengan "Male" (Laki-laki) yang memiliki 108(43.37%) kasus.
- **stroke counts for smokers by gender** : Data menunjukkan bahwa jumlah kasus stroke pada individu yang merokok adalah Laki-laki (Male) memiliki 23 kasus stroke dan Perempuan (Female) memiliki 19 kasus stroke. Meskipun jumlah kasus stroke pada individu yang merokok adalah lebih tinggi pada laki-laki (23 kasus) dibandingkan perempuan (19 kasus), perbedaan ini tidak signifikan. Data ini menekankan pentingnya kesadaran akan risiko stroke yang dapat terkait dengan merokok. Terlepas dari perbedaan jumlah kasus, merokok dapat meningkatkan risiko stroke pada laki-laki dan perempuan.

Conclusion

correlation matrix :

- korelasi age terhadap rata-rata glukosa darah : Korelasi antara usia dan rata-rata glukosa darah adalah sekitar 0.24. Ini menunjukkan bahwa ada korelasi positif yang lemah antara usia dan rata-rata glukosa darah. Artinya, dengan bertambahnya usia, rata-rata glukosa darah cenderung sedikit meningkat, meskipun korelasinya lemah.
- korelasi age terhadap bmi : Korelasi antara usia dan BMI adalah sekitar 0.33. Ini menunjukkan bahwa ada korelasi positif yang sedang antara usia dan BMI. Dengan kata lain, dengan bertambahnya usia, BMI cenderung meningkat dalam tingkat korelasi yang lebih kuat daripada korelasi dengan glukosa darah.
- korelasi antara bmi terhadap rata-rata glukosa darah : Korelasi antara rata-rata glukosa darah dan BMI adalah sekitar 0.18. Ini menunjukkan adanya korelasi positif yang lemah antara rata-rata glukosa darah dan BMI. Artinya, individu dengan BMI yang lebih tinggi cenderung memiliki rata-rata glukosa darah yang sedikit lebih tinggi, meskipun korelasinya lemah.

Conclusion

- **Data Prepocesing** : Terdapat data missing value dan outlier di kolom BMI saya melakukan imputasi pada kolom BMI menggunakan Median, dan juga melakukan penekanan nilai outliers pada kolom BMI.
- **Feature Engineering** : Saya melakukan encoded pada data kategorik gender', 'ever_married', 'work_type', 'Residence_type', smoking_status dan scalling pada data numerik avg_glucose_level, bmi, age
- **Imbalance dataset** : Saya melakukan penanggulangan imbalance dataset dengan menggunakan **SMOTEENN**
- **Modeling** : Saya mencoba membuat model Classifier dengan beberapa algoritma dengan metrik recall_score, accuracy_score, precision_score dan F1_score. menggunakan validasi silang Kfold Best Model yang Saya pilih **Random Forest Classifier**
- **Test Overfitt** : Setelah membuat model saya mencoba melakukan test overfit pada model dengan hasil overfitt signifikan dan masih perlu dilakukan penanggulangan overfit. Training recall: 1.00 dan Testing recall: 0.40
- **Hyperparameter Tuning** : Untuk menanggulangi overfit tersebut saya melakukan Hyperparameter tuning dengan menghasilkan score **Training recall: 0.95 dan Testing recall: 0.82**. karena ini kasus saya adalah untuk memprediksi stroke atau tidak maka metrik utama yg saya lihat adalah Recall_score.
- Meskipun model masih terlihat overfit namun model yg telah di tuning ini merupakan hasil awal yg baik dan perlu dilakukan pengembangan model kembali untuk mengoptimalkan kerja model ini

Terimakasih

Presentation by I Gusti Ayu Meliniarayani