

**Laporan Praktikum**  
**IF3270 Pembelajaran Mesin**



**Anggota :**

13521013	Eunice Sarah Siregar
13521018	Syarifa Dwi Purnamasari

**Teknik Informatika**  
**Sekolah Teknik Elektro dan Informatika**  
**Institut Teknologi Bandung**  
**2024**

## Hasil Analisis Data

Pada dataset yang diberikan, didapatkan bahwa dataset tersebut berisi 20 kolom dengan tipe data masing-masing kolom mayoritas adalah float64 dan terdapat sebuah target kolom dengan nama 'Diabetes'. Dataset ini berisikan kumpulan indikator individu yang diperoleh dari survei kasus diabetes. Dataset ini juga berfungsi untuk memprediksi penyakit diabetes pada suatu individu yang dapat diketahui memiliki risiko tinggi terkena diabetes atau tidak.

```
Data columns (total 20 columns):
#   Column                               Non-Null Count  Dtype
---  -
0   HighBP                               50736 non-null  float64
1   HighChol                             50736 non-null  float64
2   BMI                                  50736 non-null  float64
3   Smoker                               50736 non-null  float64
4   Stroke                               50736 non-null  float64
5   HeartDiseaseorAttack                 50736 non-null  float64
6   PhysActivity                         50736 non-null  float64
7   Fruits                               50736 non-null  float64
8   Veggies                              50736 non-null  float64
9   HvyAlcoholConsump                   50736 non-null  float64
10  AnyHealthcare                       50736 non-null  float64
11  GenHlth                             50736 non-null  float64
12  MentHlth                             50736 non-null  float64
13  PhysHlth                             50736 non-null  float64
14  DiffWalk                             50736 non-null  float64
15  Sex                                  50736 non-null  int64
16  Age                                  50736 non-null  float64
17  Education                           50736 non-null  float64
18  Income                              50736 non-null  float64
19  Diabetes                             50736 non-null  bool
dtypes: bool(1), float64(18), int64(1)
```

Terdapat beberapa *noise data* pada dataset, seperti adanya *value* yang duplikat, *value* yang hilang, *outlier*, dan jumlah distribusi data yang tidak seimbang. Berikut adalah hasil dari analisis noise data tersebut.

**Duplicated data:** 2329

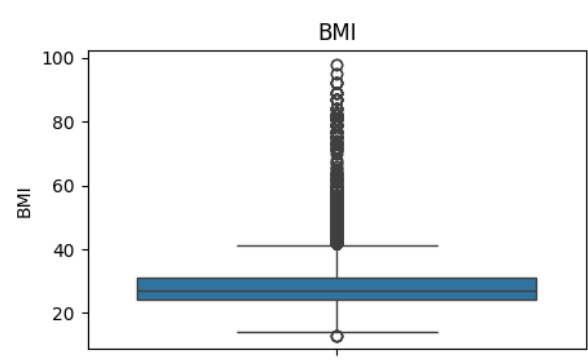
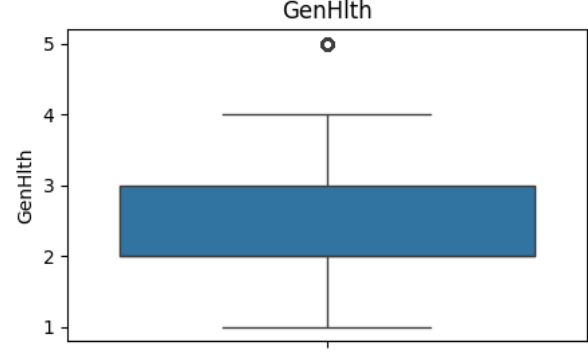
**Missing Value:**

HighBP	0
HighChol	0
BMI	0
Smoker	0
Stroke	0
HeartDiseaseorAttack	0
PhysActivity	0
Fruits	0
Veggies	0
HvyAlcoholConsump	0

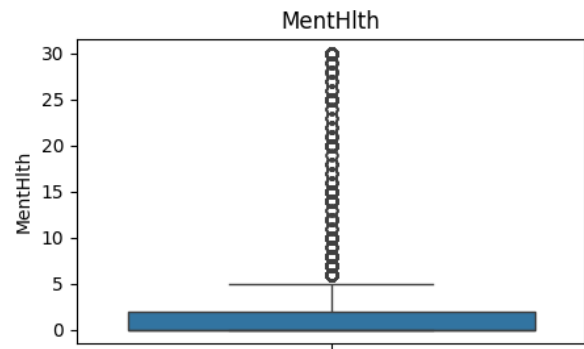
AnyHealthcare	0
GenHlth	0
MentHlth	0
PhysHlth	0
DiffWalk	0
Sex	0
Age	0
Education	0
Income	0
Diabetes	0

#### Imbalanced of Data:

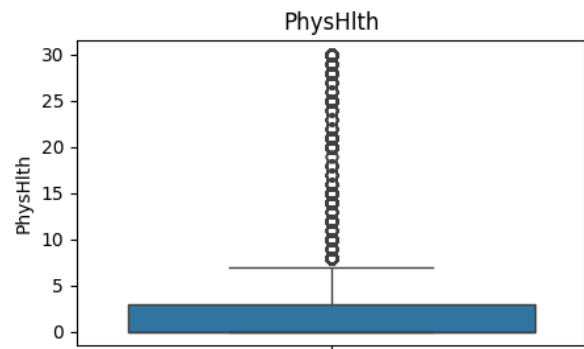
	Values	Count
0	False	43790
1	True	6946

Outlier	Box Plot
Feature: BMI Lower Bound: 13.5 Upper Bound: 41.5 Inside Range Outlier: 48757 Outside Range Outlier: 1979	 <p>The box plot for BMI shows a median around 28. The interquartile range (IQR) is from approximately 25 to 30. Whiskers extend from 13.5 to 41.5. There is a large number of outliers above the upper whisker, with values reaching up to 100.</p>
Feature: GenHlth Lower Bound: 0.5 Upper Bound: 4.5 Inside Range Outlier: 48371 Outside Range Outlier: 2365	 <p>The box plot for GenHlth shows a median around 2.5. The IQR is from approximately 2 to 3. Whiskers extend from 0.5 to 4.5. There is one outlier at the value 5.</p>

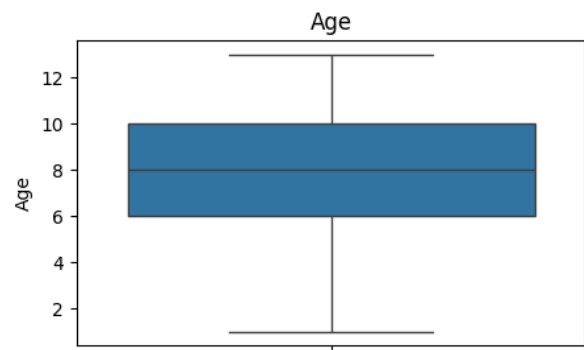
Feature: MentHlth  
Lower Bound: -3.0  
Upper Bound: 5.0  
Inside Range Outlier: 43428  
Outside Range Outlier: 7308



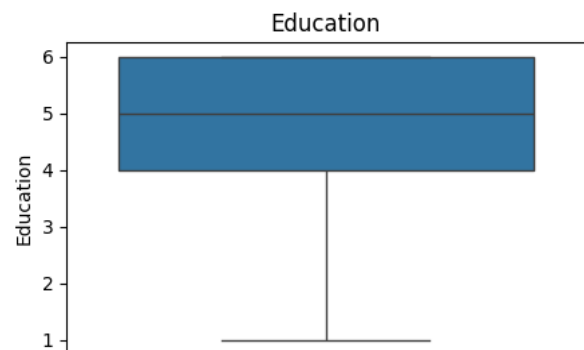
Feature: PhysHlth  
Lower Bound: -4.5  
Upper Bound: 7.5  
Inside Range Outlier: 42538  
Outside Range Outlier: 8198



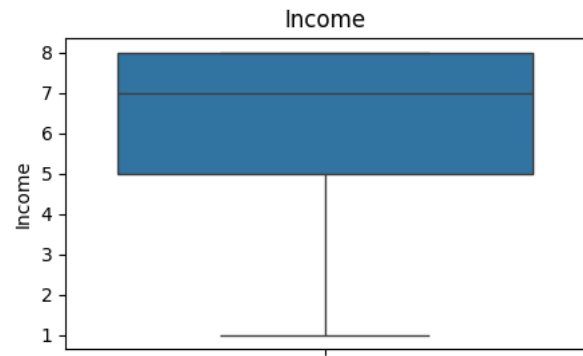
Feature: Age  
Lower Bound: 0.0  
Upper Bound: 16.0  
Inside Range Outlier: 50736  
Outside Range Outlier: 0



Feature: Education  
Lower Bound: 1.0  
Upper Bound: 9.0  
Inside Range Outlier: 50736  
Outside Range Outlier: 0



Feature: Income  
Lower Bound: 0.5  
Upper Bound: 12.5  
Inside Range Outlier: 50736  
Outside Range Outlier: 0



## Penanganan dari Hasil Analisis Data dan Justifikasi

Pada dataset yang diberikan, terdapat beberapa duplicate data, missing value, outlier, dan value data yang tidak balance. Berikut rencana penanganan yang dapat dilakukan untuk mengatasi masalah tersebut.

- Duplicate Data

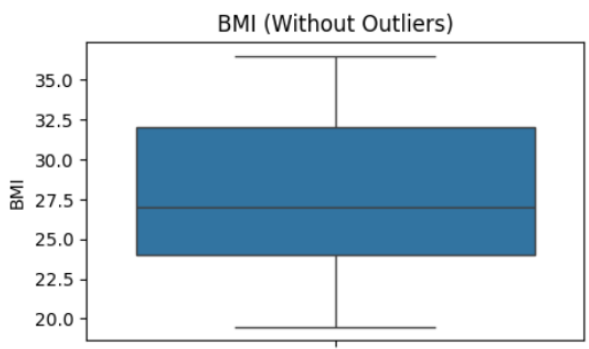
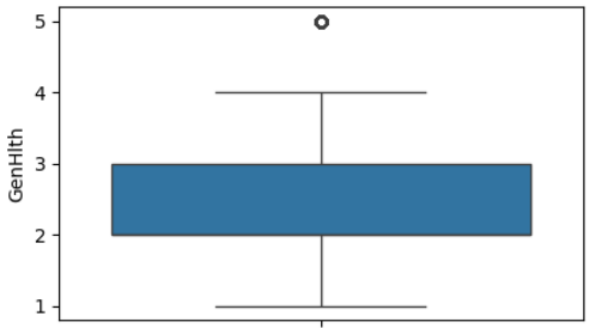
Untuk baris-baris yang memiliki value duplikat dapat diatasi dengan melakukan drop baris, sehingga baris tersebut tidak akan mengganggu perhitungan.

- Missing Value

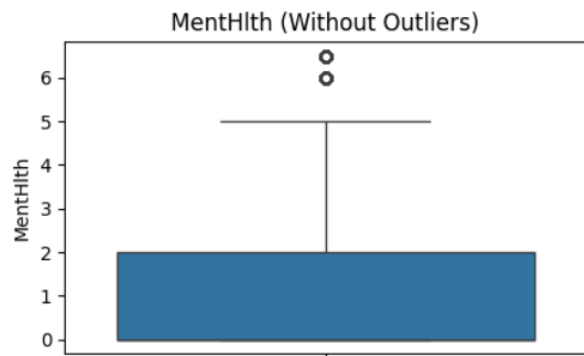
Pada dataset tidak ditemukan missing value, tetapi jika terdapat missing value dapat diatasi dengan melakukan impute untuk setiap nilai NaN tersebut dengan nilai mayoritas dari K data yang terdekat atau KNN (K-Nearest Neighbors).

- Outlier

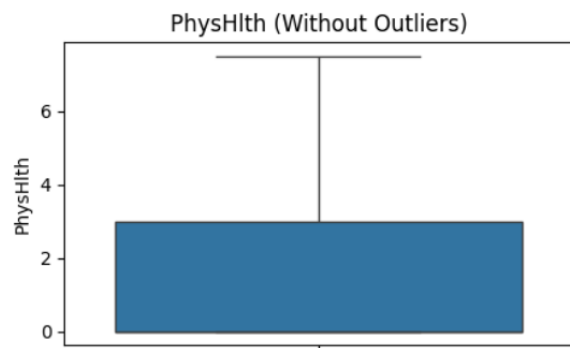
Pada dataset terdapat beberapa kolom yang memiliki outlier. Hal ini dapat diatasi dengan mengganti value untuk setiap baris menjadi nilai upper atau lower IQR.

Outlier	Box Plot
Feature: BMI Inside Range Outlier: 48407 Outside Range Outlier: 0	
Feature: GenHlth Inside Range Outlier: 48407 Outside Range Outlier: 0	

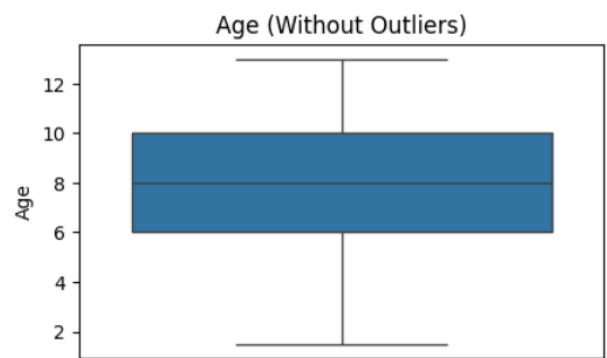
Feature: MentHlth  
Inside Range Outlier: 48407  
Outside Range Outlier: 0



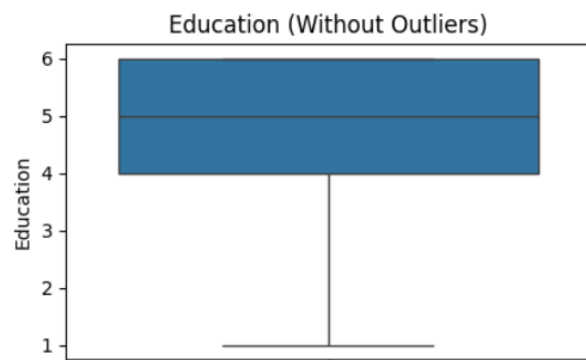
Feature: PhysHlth  
Inside Range Outlier: 48407  
Outside Range Outlier: 0



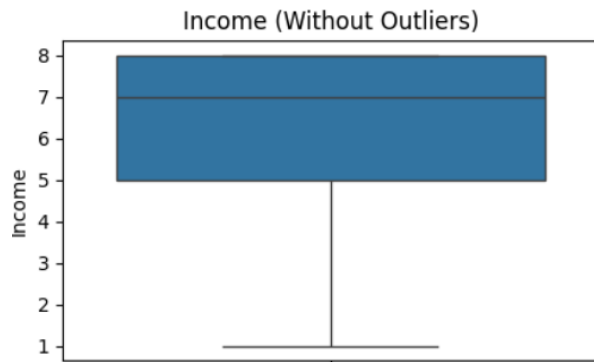
Feature: Age  
Inside Range Outlier: 48407  
Outside Range Outlier: 0



Feature: Education  
Inside Range Outlier: 48407  
Outside Range Outlier: 0



Feature: Income  
Inside Range Outlier: 48407  
Outside Range Outlier: 0



- Imbalance Data

Terdapat jumlah data yang tidak merata pada dataset, hal tersebut dapat diatasi dengan melakukan teknik resampling. Sebagai contoh dengan teknik oversampling. Teknik ini dilakukan dengan mengurangi jumlah dari sample minoritas untuk menyeimbangkan distribusi kelas training data.

### Undersampling

```
1 undersampling = RandomUnderSampler(sampling_strategy="majority", random_state=50)
2 undersampling_X, undersampling_y = undersampling.fit_resample(X, y)
3
4 undersampling_model = LogisticRegression(max_iter=1000, random_state=50)
5 undersampling_model.fit(undersampling_X, undersampling_y)
6
7 X_train, X_test, y_train, y_test = train_test_split(undersampling_X, undersampling_y, test_size=0.2, random_state=50)
8
9 undersampling_y_pred = undersampling_model.predict(X_test)
10
11 print(f"F1 Score: {f1_score(undersampling_y_pred, y_test)}")
```



F1 Score: 0.7398431931575196

### Oversampling

```
1 oversampling = RandomOverSampler(sampling_strategy="minority", random_state=50)
2 oversampling_X, oversampling_y = oversampling.fit_resample(X, y)
3
4 oversampling_model = LogisticRegression(max_iter=1000, random_state=50)
5 oversampling_model.fit(oversampling_X, oversampling_y)
6
7 X_train, X_test, y_train, y_test = train_test_split(oversampling_X, oversampling_y, test_size=0.2, random_state=50)
8
9 oversampling_y_pred = oversampling_model.predict(X_test)
10
11 print(f"F1 Score: {f1_score(oversampling_y_pred, y_test)}")
```



F1 Score: 0.7461488520332882



Pada perbandingan F1 Score menggunakan Undersampling dan Oversampling, didapatkan bahwa F1 Score pada Oversampling lebih besar sehingga dataset hasil data sampling yang akan digunakan ialah dataset hasil Oversampling. F1 Score lebih dilihat pada perbandingan ini dibandingkan Accuracy karena pada target kelas dataset yang dimiliki memiliki jumlah perbandingan False dan True yang tidak simetris.

## Desain, Hasil, dan Analisis Eksperimen

### Desain Eksperimen

- Tujuan Eksperimen

Mengembangkan model pembelajaran mesin klasifikasi biner untuk mendeteksi prediksi kasus diabetes berdasarkan beberapa tanda-tanda vital dan faktor-faktor pengaruh diabetes, dengan pemahaman yang mendalam terhadap proses analitik data, kualitas data, penanganan data, penentuan algoritma, hyperparameter, dan interpretasi hasil evaluasi model.

- Variable Dependen

Atribut Diabetes

- Variabel Independen

Atribut HighBP, HighChol, BMI, Smoker, Stroke, HeartDiseaseorAttack, PhysActivity, Fruits, Veggies, HvyAlcoholConsump, AnyHealthcare, GenHlth, MentHlth, PhysHlth, DiffWalk, Sex, Age, Education, dan Income

- Strategi Eksperimen

Eksperimen dimulai dengan tahap persiapan data yang komprehensif. Pertama, identifikasi dan tangani duplikasi data untuk mencegah bias dalam analisis. Selanjutnya, lakukan penanganan nilai yang hilang dengan melakukan imputasi menggunakan metode seperti penggantian dengan mean atau median kolom terkait. Setelah itu, fokus pada deteksi dan penanganan outlier dengan menggunakan teknik seperti Interquartile Range (IQR) atau z-score untuk menentukan tindakan yang tepat terhadap outlier yang terdeteksi. Dilanjutkan dengan penanganan ketidakseimbangan data, di mana teknik seperti oversampling atau undersampling dapat diterapkan untuk menyeimbangkan distribusi kelas dalam dataset. Setelah data dipersiapkan, langkah selanjutnya adalah membangun model klasifikasi. Pertama, bagi dataset menjadi data latih dan data uji dengan memperhatikan proporsi kelas yang seimbang setelah penanganan ketidakseimbangan data. Kemudian, terapkan Grid Search untuk menemukan kombinasi hyperparameter terbaik untuk model Logistic Regression dan beberapa teknik klasifikasi data lain seperti Random Forest, KNN, Naive Bayes, Decision Tree, menggunakan teknik cross-validation untuk evaluasi yang lebih andal. Setelah hyperparameter terbaik ditentukan, latih model menggunakan data latih dan uji performa model menggunakan data uji. Evaluasi performa model dilakukan menggunakan berbagai metrik yang relevan seperti akurasi, presisi, recall, dan F1-score, serta menginterpretasikan hasil evaluasi untuk memahami kekuatan dan kelemahan masing-masing model.

- Skema Validasi

Untuk melakukan validasi dapat menggunakan skema K-Fold Cross Validation dengan melakukan pembagian data yang akan ditrain kedalam beberapa variable. Dengan menggunakan F1 Score pada semua K-Fold, selanjutnya dapat menentukan model yang memiliki kinerja dan performa yang terbaik. Model yang sudah dipilih sebelumnya, akan dievaluasi dengan data testing.

## Hasil Eskperimen

Algoritma	Hasil
Random Forest	F1 Score: 0.9021530526671171 Accuracy: 0.9150846742602302 Confusion Matrix: [[7152 1131] [ 278 8032]]
Naive Bayes	F1 Score: 0.7012987012987013 Accuracy: 0.7158440306153198 Confusion Matrix: [[6343 1940] [2775 5535]]
Decision Tree	F1 Score: 0.7012987012987013 Accuracy: 0.9134574820707527 Confusion Matrix: [[6948 1335] [ 101 8209]]
K-Nearest Neighbors	F1 Score: 0.7761465388640879 Accuracy: 0.7744229494365094 Confusion Matrix: [[5755 2528] [1215 7095]]
Logistic Regression	F1 Score: 0.7468163428040285 Accuracy: 0.741336708250467 Confusion Matrix: [[5977 2306] [1986 6324]]

Setelah melakukan grid search menggunakan beberapa algoritma klasifikasi seperti Random Forest, Naive Bayes, Decision Tree, KNN, dan Logistic Regression, hasil evaluasi menunjukkan bahwa model Random Forest memiliki akurasi dan F1 score tertinggi dibandingkan dengan model-model lainnya. Akurasi dan F1 score yang tinggi menunjukkan bahwa model Random Forest mampu memberikan prediksi yang lebih akurat dan konsisten dalam mendeteksi kasus diabetes berdasarkan dataset yang digunakan. Oleh karena itu, Random Forest dipilih sebagai model terbaik untuk tugas klasifikasi ini.

## Data Training and Data Validation

F1 Score: 0.9170249355116079 Accuracy: 0.9127617899653457 Confusion Matrix: [[5717 246] [ 912 6399]]
<b>Data Testing</b>
F1 Score: 0.9182202905181287 Accuracy: 0.9138190803350811 Confusion Matrix: [[7135 1148] [ 282 8028]]

## Analisis Hasil

Berdasarkan hasil eksperimen yang telah dilakukan, terdapat beberapa faktor yang mendukung dan meningkatkan kualitas hasil dari eksperimen. Salah satu diantaranya adalah penggunaan teknik sampling. Setelah diperbandingkan antara oversampling dan undersampling, didapatkan nilai dari F1 score terbaik adalah oversampling. Teknik tersebut digunakan untuk menyamaratakan persebaran dari kelas minoritas. Dengan meningkatkan jumlah contoh dari kelas minoritas, model pembelajaran mesin memiliki lebih banyak informasi untuk belajar pola-pola yang mungkin ada dalam kelas tersebut, yang pada gilirannya dapat meningkatkan kinerja model dalam mengklasifikasikan kelas minoritas.

Selanjutnya dilakukan pembentukan model menggunakan Rain Forest Classification karena memiliki F1 Score terbaik. Lalu dilakukan data training, validation, dan testing untuk masing-masing model yang sudah dibuat

## Perubahan Jawaban

- Pada pengumpulan pertama, tertulis bahwa rencana penanganan outlier dengan cara melakukan drop. Namun, setelah dicoba didapatkan F1 score yang kurang memuaskan (sekitar 0.38), sehingga kami melakukan perubahan dengan menangani outlier dengan mengganti value outlier menjadi lower atau upper dari IQR dan didapatkan F1 score sebesar 0.74.
- Selain itu, terdapat perubahan pada penanganan imbalance data dengan yang awalnya dilakukan *undersampling*, tetapi setelah dicoba didapatkan F1 score *oversampling* bernilai lebih besar daripada *undersampling*.
- Terdapat perubahan pada strategi desain eksperimen. Pada awalnya dilakukan strategi dengan membuat model SVC, tetapi setelah melakukan research lebih lanjut, ditemukan beberapa model yang lebih cocok digunakan untuk dataset ini. Pada akhirnya, digunakan teknik klasifikasi dengan Random Forest, K-Nearest Neighbors, Naive Bayes, dan Decision Tree. Dari keempat model diatas, terpilih model terbaik yaitu Random Forest.

## Kesimpulan

Berdasarkan eksperimen yang telah dilakukan, dapat disimpulkan bahwa sangat penting untuk melakukan data cleaning sebelum melakukan eksperimen. Teknik-teknik yang dipilih untuk mengatasi noise data sangat berpengaruh pada hasil akhir F1 score yang didapatkan, sehingga diperlukan analisis dan banyak percobaan yang dilakukan untuk mendapatkan F1 score yang optimal. Semakin besar nilai F1 score yang didapatkan maka dapat disimpulkan bahwa model *machine learning* yang diterapkan sudah memiliki performa yang baik.

Pada eksperimen ini untuk mengatasi noise data dilakukan beberapa teknik, yaitu melakukan drop, mengganti nilai *noise data* menjadi *upper* atau *lower* dari IQR, dan memanfaatkan teknik *sampling* seperti *undersampling* dan *oversampling*. Setelah melakukan *data cleaning*, dilakukan optimalisasi model dengan beberapa jenis teknik, seperti Random Forest, KNN, Naive Bayes, dan Decision Tree untuk menentukan hyperparameter terbaik dan didapatkan bahwa Random Forest memiliki F1 Score terbaik sebesar 0.90 dan akurasi sebesar 0.91 dengan *best parameter* adalah *max\_depth* sebesar 20, *max\_features* sebesar log2, *min\_samples\_leaf* sebesar 4, dan *max\_feature* sebesar log2. Selanjutnya dilakukan data training dan validation berdasarkan hasil best parameter dengan memanfaatkan variable X\_val dan y\_val. Terakhir, dilakukan testing dengan hasil F1 Score sebesar 0.91 dan akurasi 0.91.

## Pembagian Tugas

NIM	Nama	Pembagian Tugas
13521013	Eunice Sarah Siregar	Data Preparation, Naive Bayes, Decision Tree, Data Training, Data Validation, Data Testing, Laporan
13521018	Syarifa Dwi Purnamasari	Data Preparation, Random Forest, KNN, Logistic Regression, Laporan