KLASIFIKASI PENYAKIT DIABETES DENGAN MENGGUNAKAN ALGORITHMA MACHINE LEARNING

Muhammad Rizky Anugrah

Program Studi Fisika, Universitas Negeri Jakarta, Jalan Rawa Mangun Muka Raya No.11, RT.11/RW.14, Rawamangun, Kec. Pulo Gadung, Kota Jakarta Timur, Daerah Khusus Ibukota Jakarta 13220
Email : muhammadrizkyanugrah_1306620089@mhs.unj.ac.id

Abstrak – Diabetes adalah penyakit kronis yang berpotensi menyebabkan krisis perawatan kesehatan di seluruh dunia. Berdasarkan Federasi Diabetes Internasional 382 juta orang hidup dengan diabetes di seluruh dunia. Pada tahun 2035, ini akan menjadi dua kali lipat menjadi 592 juta. Diabetes melitus atau biasa disebut penyakit kencing manis adalah penyakit yang disebabkan karena peningkatan kadar glukosa darah. Berbagai metode tradisional, berdasarkan uji fisik dan kimia, tersedia untuk mendiagnosis diabetes. Namun, prediksi awal diabetes merupakan tugas yang cukup menantang bagi praktisi medis karena saling ketergantungan yang kompleks pada berbagai faktor karena diabetes mempengaruhi organ tubuh manusia seperti ginjal, mata, jantung, saraf, kaki, dll. Metode ilmu data memiliki potensi untuk bermanfaat bagi bidang ilmiah lainnya dengan memberikan pencerahan baru pada pertanyaan umum. Salah satu tugas tersebut adalah membantu membuat prediksi pada data medis. Machine Learning adalah sebuah bidang ilmiah yang muncul dalam ilmu data yang berurusan dengan cara mesin belajar dari pengalaman. Itu Tujuan dari proyek ini adalah untuk mengembangkan sistem yang dapat melakukan prediksi dini diabetes untuk pasien dengan akurasi yang lebih tinggi dengan menggabungkan hasil dari berbagai teknik machine learning. Eksperimen ini bertujuan untuk memprediksi diabetes melalui lima metode supervised machine learning yaitu: ANN, Naïve bayes, SVM, Decision tree dan Random forest. Eksperimen ini juga bertujuan untuk mengusulkan teknik yang efektif untuk deteksi dini penyakit diabetes.

Kata kunci : Diabetes; Machine Learning; Supervised; ANN; Naïve Bayes; SVM; Decision Tree; Random Forest.

I. PENDAHULUAN

1.1 Diabetes Mellitus

Diabetes Melitus (DM) merupakan penyakit kronis yang ditandai dengan hiperglikemia dan intoleransi glukosa yang terjadi karena kelenjar pankreas tidak dapat memproduksi insulin secara adekuat yang atau karena tubuh tidak dapat menggunakan insulin yang diproduksi secara efektif atau kedua-duanya (Evi, 2016).

Kadar gula (glukosa) dalam darah dikendalikan oleh hormon insulin yang diproduksi pankreas. Namun, pada penderita diabetes, pankreas tidak mampu memproduksi insulin sesuai kebutuhan tubuh. Tanpa insulin, sel-sel tubuh tidak dapat menyerap dan mengolah glukosa menjadi energi.

Glukosa yang tidak diserap sel tubuh dengan baik akan menumpuk dalam darah. Kondisi tersebut dapat menimbulkan berbagai gangguan pada organ tubuh. Jika tidak terkontrol dengan baik, diabetes dapat

menimbulkan komplikasi yang berisiko mengancam nyawa penderitanya.

Secara umum, diabetes dibedakan menjadi dua, yaitu diabetes tipe 1 dan tipe 2. Diabetes tipe 1 merupakan penyakit kronis yang paling banyak diderita oleh anak dan remaja di dunia. Diabetes tipe 1 adalah kelainan sistemik akibat terjadinya gangguan metabolisme glukosa yang ditandai oleh hiperglikemia kronik. Keadaan ini diakibatkan oleh kerusakan sel β pankreas baik oleh proses autoimun maupun idiopatik sehinga produksi insulin berkurang bahkan berhenti (Aloysia, 2017). Diabetes Mellitus Tipe 2 merupakan penyakit hiperglikemi akibat insensivitas sel terhadap insulin. Kadar insulin mungkin sedikitmenurun atau berada dalam rentang normal. Karena insulin tetap dihasilkan oleh sel-sel beta pankreas, maka diabetes mellitus tipe II dianggap sebagai non insulin dependent diabetes mellitus (Slamet, 2008). Diabetes Mellitus Tipe 2 adalah penyakit gangguan metabolik yang di tandai oleh kenaikan gula darah akibat penurunan sekresi insulin oleh sel beta pankreas dan atau ganguan fungsi insulin (resistensi insulin) (Restyana, 2015).

1.2 Machine Learning

Machine learning adalah digunakan untuk mengajari mesin cara menangani data lebih banyak efisien. Terkadang setelah melihat data, tidak bisa menafsirkan informasi ekstrak dari data. Dalam hal itu, menerapkan pembelajaran mesin. Tujuan dari machine learning adalah untuk belajar dari data. Banyak penelitian telah dilakukan tentang cara membuat mesin belajar sendiri tanpa diprogram secara eksplisit. Banyak matematikawan dan programmer menerapkan beberapa pendekatan untuk menemukan solusi dari masalah ini yaitu memiliki kumpulan data yang besar (Batta, 2018).

Algoritma machine learning adalah algoritma yang digunakan dalam proses machine learning, di mana sistem melakukan pembelajaran berdasarkan data. Teknik yang digunakan oleh Supervised Learning adalah metode klasifikasi di mana kumpulan data sepenuhnya diberikan label untuk mengklasifikasikan kelas yang tidak dikenal. Sedangkan teknik Unsupervised Learning sering disebut cluster dikarenakan tidak ada kebutuhan untuk pemberian label dalam kumpulan data dan hasilnya tidak mengidentifikasi contoh di kelas yang telah ditentukan (Thupae, 2018).

1.3 Supervised Learning

Metode supervised learning didasarkan pada kumpulan sampel data yang memiliki label. Kumpulan sampel digunakan untuk meringkas karakteristik distribusi ukuran perilaku dalam setiap jenis aplikasi sehingga membentuk model perilaku dari data (Amei, 2011). Supervised learning memiliki beberapa algoritma populer seperti Linear regression, Random Forest, Support Vector Machines, Naive Bayes, Decision Tree, Neural Network, Logistic Regression, dan Neural Network (Ahmad, 2019).

1.4 Support Vector Machine

Support Vector Machine (SVM) merupakan satu di antara banyak algoritma yang digunakan untuk klasifikasi dan termasuk dalam kategori supervised learning. Konsep kerja Support Vector Machine yaitu dengan mencari hyperplane atau garis pembatas paling optimal yang berfungsi untuk memisahkan dua kelas (Hendry, 2021). Keuntungan algoritma ini adalah cepat, efektif untuk ruang dimensi tinggi, akurasi yang bagus, powerful dan fleksibel, dan dapat digunakan di banyak aplikasi.

1.5 Random Forest

Random forest classifier merupakan metode klasifikasi yang terdiri dari kumpulan pohon keputusan yang nantinya akan dijadikan vote untuk mendapatkan hasil terakhir dari pendeteksian sarkasme dengan pendukung berupa data latih dan fitur acak yang independen dengan fitur yang berbeda-beda (Klyueva, 2019). Pohon keputusan dibuat dengan menentukan node akar dan berakhir dengan beberapa node daun untuk mendapatkan hasil akhir. Keuntungan dari algoritma ini adalah dapat digunakan u8ntuk

rekayasa fitur seperti mengidentifikasi fitur yang paling penting diantara semua fitur yang tersedia dalam dataset training, bekerja sangat baik pada database berukuran besar, sangat fleksibel, dan memiliki akurasi yang tinggi.

1.6 Decision Tree

Decision tree membangun model klasifikasi dan regresi dalam bentuk struktur pohon. Algoritma ini menguraikan kumpulan data menjadi himpunan bagian yang lebih kecil dan menghubungkannya menjadi pohon keputusan yang terkait. Tujuan utama dari algoritma decision tree adalah untuk membangun model pelatihan yang digunakan untuk memprediksi nilai variabel target dengan mempelajari aturan keputusan. Aturan ini disimpulkan dari data training yang sebelumnya telah diinput. Keuntungan algoritma ini adalah mudah dimengerti, mudah menghasilkan aturan, tidak mengandung hiper-parameter, dan model decision tree yang kompleks dapat disederhanakan secara signifikan dengan visualisasinya.

1.7 Neural Network

Neural Network merupakan model algoritma yang mencoba meniru otak manusia yang mampu memberikan stimulasi/rangsangan, melakukan proses, dan memberikan output untuk menemukan hubungan antara kumpulan data. Neural Network yang paling umum dikenal dengan Artificial Neural Network atau biasa disebut sebagai Jaringan Saraf Tiruan. Artificial Neural Network (ANN) atau Jaringan Saraf Tiruan merupakan salah satu pemodelan kompleks yang dapat memprediksi bagaimana ekosistem merespon perubahan variabel lingkungan dengan terinspirasi oleh cara kerja sistem saraf biologis, khususnya pada sel otak manusia dalam memproses informasi.

1.8 Naïve Bayes

Naive Bayes adalah algoritma klasifikasi yang mengadopsi prinsip dari Teorema Bayes. Algoritma ini menganggap bahwa kehadiran satu fitur tidak memengaruhi keberadaan fitur lain dalam probabilitas hasil yang diberikan, dan setiap prediktor memiliki efek yang sama pada hasil tersebut.

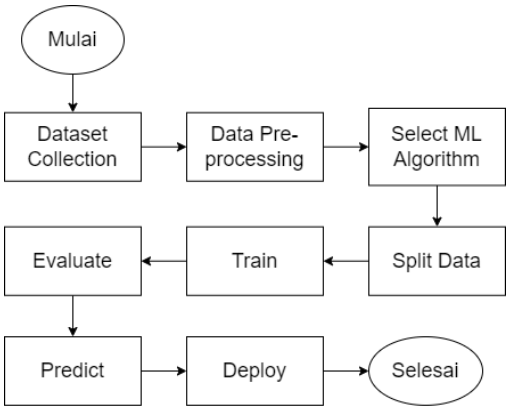
Tujuan melakukan penelitian ini yaitu untuk klasifikasi penyakit diabetes dengan metode supervised learning menggunakan algoritma machine learning seperti support vector machine, random forest, decision tree, neural network, naïve bayes dan lain sebagainya dan membandingkan hasil yang didapatkan dari algoritma yang digunakan.

II. METODE PENELITIAN

Pada penelitian kali ini yaitu memprediksi penyakit diabetes menggunakan algorithma machine learning berdasarkan klasifikasi yang telah ada. Algorithma yang digunakan yaitu Supervised Learning. Supervised learning adalah suatu metode untuk menciptakan artificial intelligence (AI).

Dalam supervised learning, algoritma computer dilatih dengan input data yang telah diberi label khusus sehingga menghasilkan output tertentu. Algoritma tersebut dilatih hingga dapat mengenali pola dan mendeteksi hubungan yang mendasari antara data input dan label output. Dengan begitu, nantinya algoritma dapat memberikan hasil pelabelan yang akurat meski data yang disajikan belum pernah ada sebelumnya. Supervised Learning merupakan sebuah pemodelan dimana algoritmanya dapat membangkitkan suatu fungsi yang memetakan input ke output yang diinginkan. Pada Supervised Learning dapat mengolah data yang memiliki label sehingga tujuan pengolahan tersebut adalah mengelompokkan data ke data yang sudah ada. Proses pengolahan data yang dilakukan jika menggunakan Supervised Learning juga memerlukan data training, data training sendiri digunakan dalam meprediksi maupun mengklasifikasi data. Supervised Learning dalam kehidupan sehari-hari bisa ditemukan pada kasus prediksi harga saham, klasifikasi pelanggan, klasifikasi gambar dan lain-lain.

Pada penelitian ini dapat dilihat urutan dan langkah-langkah dalam melakukan percobaan yaitu sebagai berikut.



2.1 Dataset Collection

Dataset Collection merupakan langkah awal dalam melakukan prediksi penyakit diabetes menggunakan algorithma machine learning. Dataset yang digunakan didapat dari Kaggle yaitu Pima Indians Diabetes Database.

2.2 Data Pre-processing

Fase model ini menangani data yang tidak konsisten untuk mendapatkan hasil yang lebih akurat dan presisi. Kumpulan data ini berisi nilai yang hilang. Jadi memasukkan nilai yang hilang untuk beberapa atribut yang dipilih seperti tingkat Glukosa, Tekanan Darah, Kulit Ketebalan, BMI dan Umur karena atribut ini tidak boleh memiliki nilai nol. Kemudian kami menskalakan dataset untuk dinormalisasi semua nilai.

2.3 Select ML Algorithm

Langkah selanjutnya adalah memilih algoritma yang cocok untuk menangani problem yang muncul. Algoritma yang digunakan yaitu Artificial Neural Network, Naïve Bayes, Support Vector Machine, Decision Tree dan Random Forest.

2.4 Split Data

Proses ini terjadi saat pembagian data agar memudahkan dalam proses selanjutnya. Dengan data yang dibagi ini model machine learning mudah untuk diidentifikasi. Variabel predictor medis sebagai independent variable. Outcome sebagai dependent variable (variable target).

2.5 Train

Proses train digunakan untuk melatih tubuh machine learning dengan mengisikan data yang sudah diproses. Mulai dari proses mengidentifikasi, mempersiapkan data, dan memilih algoritma yang tepat. Melakukan pembagian data yang sudah diproses. Mesin dilatih agar dapat menentukan data yang masuk sesuai dengan klasifikasinya.

2.6 Evaluate

Langkah selanjutnya adalah mengevalusai kembali machine learning. Memeriksa data yang telah dimasukan sudah benar dan juga sesuai Proses ini menentukan keakuratan prediksi atau hasil dari machine learning yang telah dibuat.

2.7 Predict

Melakukan prediksi akurasi yang didapatkan dari 5 algoritma yang digunakan beserta hasil confusion matrix dari masing masing algoritma.

2.8 Deploy

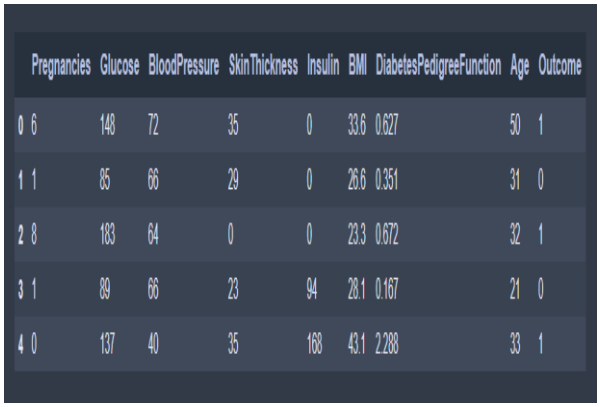
Mendapatkan hasil train accuracy dan test accuracy dari masing-masing algoritma yang digunakan dan juga hasil accuracy dari confusion matrix yaitu Accuracy, Miss Accuracy, Precison Score, Recall Score, False Positive Rate, Specificity, F1 Score dan ROC AUC Score. Membuat prediction system agar dapat memprediksi data input yang dimasukan merupakan pasien diabetes atau non diabetes. Beserta membandingkan hasil train accuracy, test accuracy, hasil confusion matrix dan juga prediction sistem yang telah dibuat.

III. HASIL DAN PEMBAHASAN

3.1 Dataset Collection

Data yang digunakan dalam penelitian ini, data yang diambil dari website Kaggle dengan jumlah 768 baris dan juga 9 column. Kumpulan data terdiri dari beberapa variabel prediktor medis (Independent) dan satu

variabel target atau hasil (Dependent). Variabel prediktor meliputi kehamilan. Glukosa. Tekanan Darah, Ketebalan Kulit, Insulin. BMI. Riwayat diabetes dalam keluarga (Diabetes Pedigree Function), usia, dan Hasil. Berikut merupakan gambaran data yang didapatkan.



	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1

3.2 Data Pre-processing

Mengganti nilai zero values dari setiap kolom dengan rata-rata yang dihasilkan oleh kolom yang sama.

3.3 Select ML Algorithm

Metode algoritma machine learning yang digunakan untuk melakukan klasifikasi penyakit diabetes ialah Artificial Neural Network, Naïve Bayes, Support Vector Machine, Decision Tree dan Random Forest.

Library yang digunakan ialah sebagai berikut.

1. Artificial Neural Network = MLPClassifier.

```
from sklearn.neural_network import MLPClassifier
```

2. Naïve Bayes = GaussianNB.

```
from sklearn.naive_bayes import GaussianNB
```

3. Support Vector Machine = SVC.

```
from sklearn.svm import SVC
```

4. Decision Tree = DecisionTreeClassifier

```
from sklearn.tree import DecisionTreeClassifier
```

5. Random Forest = RandomForestClassifier(criterion='entropy')

```
from sklearn.ensemble import RandomForestClassifier
```

3.4 Split Data

Melakukan split data frame menjadi X dan y. Variabel X merupakan variabel prediktor medis (Independent) dan y merupakan variabel target Outcome (Dependent). Variabel prediktor meliputi kehamilan. Glukosa. Tekanan Darah, Ketebalan Kulit, Insulin.BMI. Riwayat diabetes dalam keluarga (Diabetes Pedigree Function), usia, dan hasil atau outcome.

```
target_name = 'Outcome'
y = diabetes_dataset[target_name]
X = diabetes_dataset.drop(target_name, axis=1)
```

3.5 Train

Melakukan pembagian data yang sudah diproses. train/test split adalah salah satu metode yang dapat digunakan untuk mengevaluasi performa model machine learning. Metode evaluasi model ini membagi dataset menjadi dua bagian yakni bagian yang digunakan untuk training data dan untuk testing data dengan proporsi tertentu.

```
APPLY FEATURE SCALING
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
scaler.fit(X)
standardized_data = scaler.transform(X)
TRAIN TEST SPLIT
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(
    standardized_data, y, test_size = 0.2, random_state=4)
```

3.6 Evaluate

Melakukan evaluasi program yang telah dilakuka. Memeriksa kembali data yang telah dimasukan sudah benar dan juga sesuai. Memastikan agar tidak terdapat variabel yang bertabrakan. Melalukan semua tahapan

yang diperlukan sebelum mendeteksi data diabetes dengan klasifikasi algoritma machine learning. Proses ini menentukan berjalanya suatu program dan keakuratan prediksi yang dihasilkan.

3.7 Predict

Melakukan prediksi menggunakan klasifikasi algoritma machine learning yaitu Artificial Neural Network, Naïve Bayes, Support Vector Machine, Decisiom Tree dan Random Forest. Variabel yang digunakan untuk melakukan prediksi yaitu data training dan test yang didaptkan melalui training test split data. Source code untuk melakukan prediksi yaitu sebagai berikut.

1. Artificial Neural Network

```
mlp = MLPClassifier()
mlp.fit(X_train, y_train)
```

```
mlp_pred = mlp.predict(X_test)
```

2. Naïve Bayes

```
nb = GaussianNB()
nb.fit(X_train, y_train)
```

```
nb_pred = nb.predict(X_test)
```

3. Support Vector Machine

```
sv = SVC()
sv.fit(X_train,y_train)
```

```
sv_pred = sv.predict(X_test)
```

4. Decision Tree

```
dt = DecisionTreeClassifier()
dt.fit(X_train,y_train)
```

```
dt_pred = dt.predict(X_test)
```

5. Random Forest

```
rf = RandomForestClassifier(criterion='entropy')
rf.fit(X_train,y_train)
```

```
rf_pred = rf.predict(X_test)
```

Dari source code diatas akan mendapatkan accuracy dari data training dan juga data test.

Prediksi selanjutnya menggunakan confusion matrix.

Confusion Matrix

Confusion Matrix adalah tabel yang digunakan untuk menggambarkan kinerja masalah klasifikasi. Ini memvisualisasikan keakuratan classifier dengan membandingkan nilai prediksi dengan nilai sebenarnya. Istilah yang digunakan dalam Confusion Matrix adalah True positive (TP), true negative (TN). positif palsu (FP) dan negatif palsu (FN)

- True Positive: Hasil yang diprediksi adalah positif, meskipun diberi label positif.
- False Positive: Hasil yang diprediksi adalah positif, meskipun diberi label negatif. Itu juga menyebut Kesalahan Tipe I.
- False Negative: Hasil yang diprediksi adalah negatif, meskipun diberi label positif. Itu juga menyebut Kesalahan Tipe II.
- True Negative: Hasil prediksi negatif, sementara itu diberi label negatif.

Source code dan label confusion matrix menggunakan algoritma machine learning yaitu sebagai berikut.

1. Artificial Neural Network

```
from sklearn.metrics import classification_report,confusion_matrix
# Confusion Matrix of ANN
cm_mlp = confusion_matrix(y_test,mlp_pred)
cm_mlp
```

```
array([[82, 20],
       [17, 35]], dtype=int64)
```

2. Naïve Bayes

```
from sklearn.metrics import classification_report,confusion_matrix
# Confusion Matrix of Naive Bayes
cm_nb = confusion_matrix(y_test,nb_pred)
cm_nb
```

```
array([[81, 21],
       [18, 34]], dtype=int64)
```

3. Support Vector Machine

```
from sklearn.metrics import classification_report,confusion_matrix
# Confusion Matrix of Support Vector Machine
cm_sv = confusion_matrix(y_test,sv_pred)
cm_sv
```

4. Decision Tree

```
from sklearn.metrics import classification_report,confusion_matrix
# Confusion Matrix of Decision Tree
cm_dt = confusion_matrix(y_test,dt_pred)
cm_dt
```

```
array([[75, 27],
       [20, 32]], dtype=int64)
```

5. Random Forest

```
from sklearn.metrics import classification_report,confusion_matrix
# Confusion Matrix of Random Forest
cm_rf = confusion_matrix(y_test,rf_pred)
cm_rf
```

```
array([[84, 18],
       [20, 32]], dtype=int64)
```

Rumus Accuracy Menggunakan Confusion Matrix

1. Accuracy

Accuracy menggambarkan seberapa akurat model yang digunakan dalam klasifikasi.

Rumus :

$$\text{accuracy} = \left(\frac{(TP+TN)}{(TP+FP+TN+FN)}\right) \times 100\%$$

2. Miss Accuracy

Miss Accuracy menggambarkan kesalahan dalam accuracy dalam klasifikasi.

Rumus :

$$\text{miss accuracy} = 100 - accuracy$$

3. Precision

Precision menggambarkan akurasi antara data yang diminta dengan hasil prediksi yang diberikan oleh model.

Rumus :

$$\text{precision} = \left(\frac{TP}{(TP+FP)}\right) \times 100\%$$

4. Recall

Recall menggambarkan keberhasilan model dalam menemukan kembali informasi.

Rumus :

$$\text{recall} = \left(\frac{TP}{(TP+FN)}\right) \times 100\%$$

5. False Positive Rate

False Positive Rate menggambarkan proporsi mereka yang tes nya positif terhadap seluruh populasi yang tidak berpenyakit.

Rumus :

$$\text{false positive rate} = \left(\frac{FP}{(FP+TN)}\right) \times 100\%$$

6. Specificity

Merupakan kebenaran memprediksi negatif dibandingkan dengan keseluruhan data negatif.

Rumus :

$$\text{specificity} = \left(\frac{TN}{(TN+FP)}\right) \times 100\%$$

7. F1 Score

F-1 score menggambarkan perbandingan rata-rata precision dan recall yang dibobotkan.

Rumus:

$$f1 - \text{score} = \left(\frac{2 \times \text{precision} \times \text{recall}}{(\text{precision} + \text{recall})}\right) \times 100\%$$

8. ROC AUC Score

ROC (Receiver Operating Characteristics) adalah semacam alat ukur performance untuk classification problem dalam menentukan threshold dari suatu model.

Area yang berada dibawah kurva merupakan wilayah yang yang menunjukkan tingkat keakuratan dari model empirik dan dihitung dengan metode perhitungan yang disebut Area Under Curve (AUC). AUC merupakan daerah berbentuk persegi yang nilainya selalu berada diantara 0 dan 1.

ROC AUC Score didapatkan menggunakan library sebagai berikut.

```
from sklearn.metrics import accuracy_score, roc_auc_score, roc_curve
```

ROC AUC Score didapatkan dengan cara memanggil library accuracy_score dengan memasukan variabel y test dan juga predict dari masing masing algoritma.

Kurva ROC adalah salah satu metrik evaluasi penting yang harus digunakan untuk memeriksa kinerja model klasifikasi. Disebut juga kurva karakteristik operasi relatif, karena merupakan perbandingan dari dua karakteristik utama (TPR dan FPR). Ini diplot antara sensitivitas (alias ingat alias True Positive Rate) dan False Positive Rate (FPR 1-spesifisitas). = Kurva ROC (Receiver Operating Characteristic) memberi tahu tentang seberapa baik model dapat membedakan antara dua hal (mis. Jika pasien memiliki penyakit atau tidak).

ROC Curve didapatkan dengan cara memanggil library roc_curve dengan memasukan variabel y test dan juga predict dari masing masing algoritma.

3.8 Deploy

Mendapatkan hasil train accuracy dan test accuracy dari masing-masing algoritma yang digunakan dan juga hasil accuracy dari confusion matrix. Beserta melakukan tuning parameter model menggunakan test accuracy dan juga membuat sistem prediksi pasien diabetes.

Berikut merupakan hasil train accuracy dan test accuracy dari masing masing-lagoritma yang digunakan.

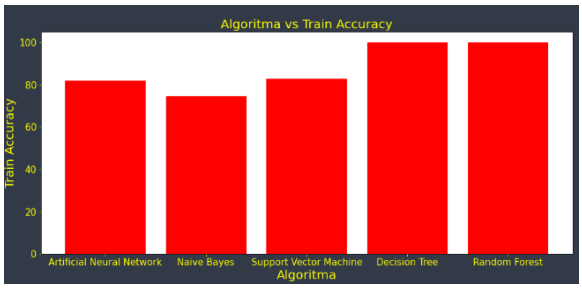
	Algoritma	Train Accuracy	Accuracy
1	Artificial Neural Network	81.758958	75.974026
2	Naive Bayes	74.592834	74.675325
3	Support Vector Machine	82.899023	76.623377
4	Decision Tree	100.000000	69.480519
5	Random Forest	100.000000	75.324675

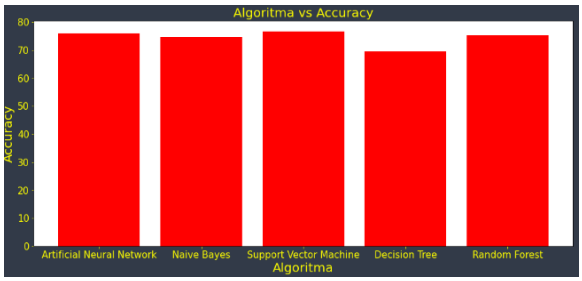
Berikut merupakan hasil accuracy menggunakan confusion matrix.

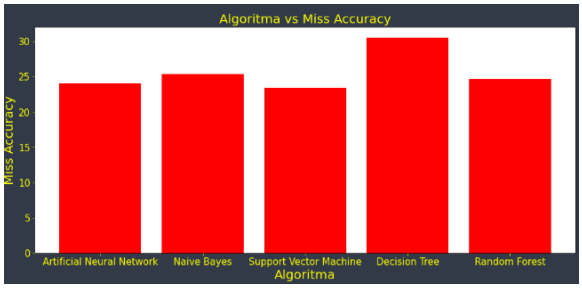
	Algoritma	Miss Accuracy	Precison	Recall
1	Artificial Neural Network	24.025974	63.636364	67.307692
2	Naive Bayes	25.324675	61.818182	65.384615
3	Support Vector Machine	23.376623	66.666667	61.538462
4	Decision Tree	30.519481	54.237288	61.538462
5	Random Forest	24.675325	64.000000	61.538462

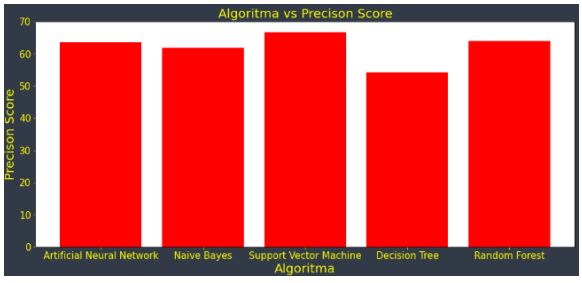
False Positive Rate	Specificity	F1 Score	ROC AUC Score
19.607843	80.392157	65.420561	0.738499
20.588235	79.411765	63.551402	0.723982
15.686275	84.313725	64.000000	0.729261
26.470588	73.529412	57.657658	0.675339
17.647059	82.352941	62.745098	0.719457

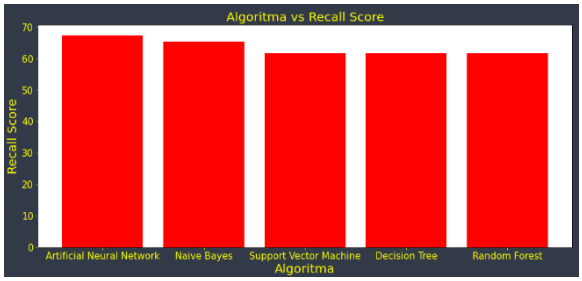
Berikut merupakan hasil grafik algoritma dengan accyracy yang dihasilkan.

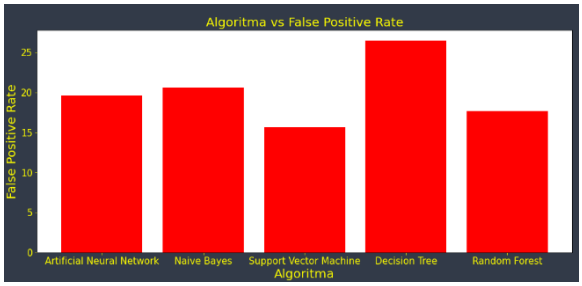


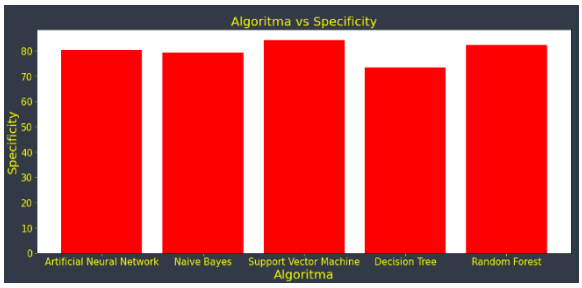


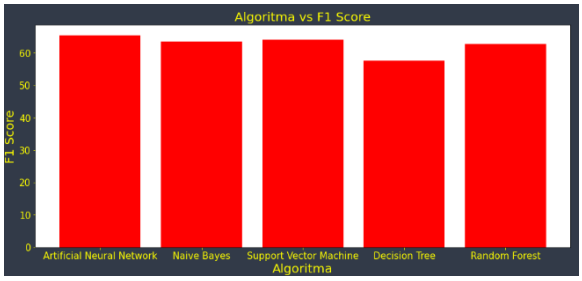


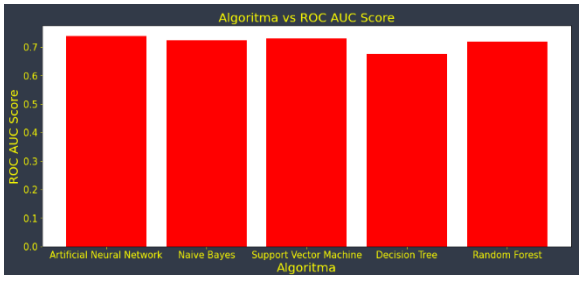




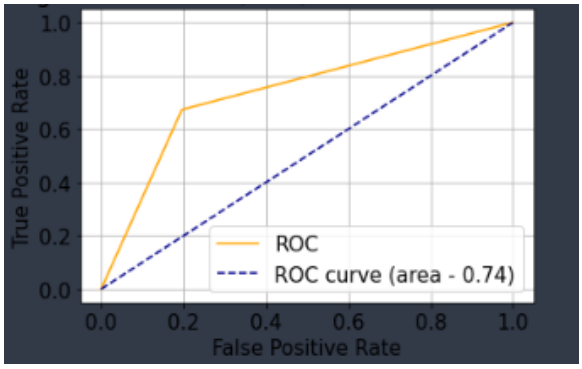




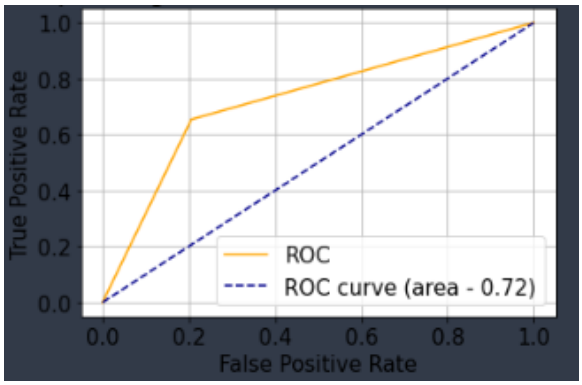




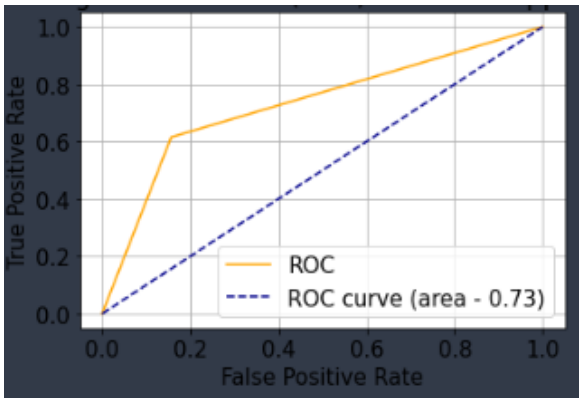
Selanjutnya merupakan hasil ROC Kurva yang didapatkan dari masing-masing algoritma.



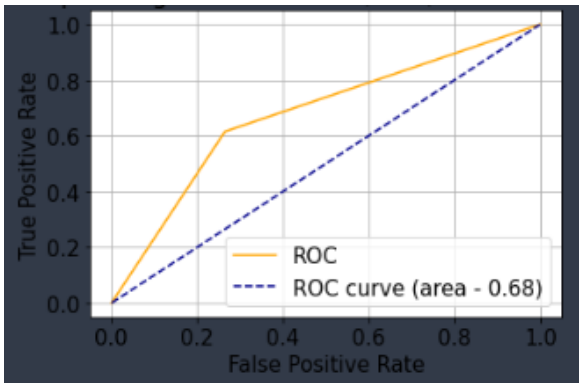
ROC Kurva ANN



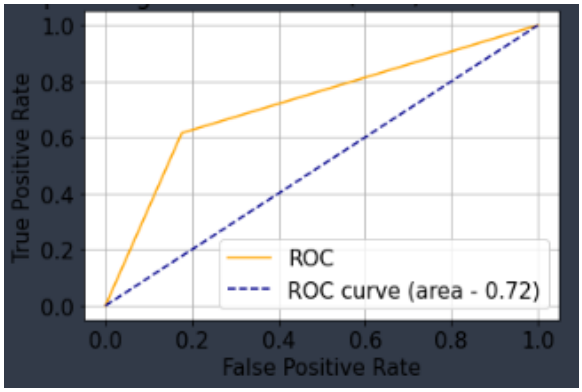
ROC Kurva Naïve Bayes



ROC Kurva SVM



ROC Kurva Decision Tree



ROC Kurva Random Forest

Selanjutnya melakukan tuning parameter model accuracy test dari masing-masing algoritma machine learning yang digunakan menggunakan library GridSearchCV sebagai berikut.

	model	best_score	best_params
0	ANN	0.805806	{'activation': 'tanh', 'hidden_layer_sizes': (...
1	naive_bayes_gaussian	0.766882	{}
2	svm	0.798925	{'C': 1, 'kernel': 'linear'}
3	decision_tree	0.700860	{'criterion': 'gini'}
4	random_forest	0.773548	{'n_estimators': 40}

Kemudian. membuat prediksi sistem pasien positif diabetes dan juga non diabetes dari masing-masing klasifikasi algoritma yang digunakan memiliki hasil yang sama yaitu sebagai berikut.

```
input_data1 = (5,166,72,19,175,25.8,0.587,51)
input_data2 = (1,85,66,29,0,26.6,0.351,31)
```

Untuk data 1 merupakan data pasien diabetes dan data 2 pasien non diabetes. Hasil perdiksi yang didapatkan yaitu sebagai berikut.

```
[1]
The person is diabetic
```

Data 1 menyatakan pasien tersebut diabetes.

```
[0]
The person is not diabetic
```

Data 2 menyatakan pasien tersebut non diabetes.

IV. KESIMPULAN

Berdasarkan penelitian mendeteksi penyakit diabetes menggunakan klasifikasi algoritrma machine learning yang berbeda dapat dilihat bahwa menghasilkan accuracy yang berbeda-beda. training accuracy terbaik diperoleh oleh decision tree dan juga random forest. test accuracy terbaik diperoleh oleh klasifikasi algoritma support vector machine. Jika ditinjau melalui confusion matrix hasil accuracy berdasarkan Accuracy, Miss Accuracy, Precison Score, Recall Score, False Positive Rate, Specificity, F1 Score dan ROC AUC Score algoritma machine learning lebih unggul. Berdasarkan grafik yang dihasilkan dapat terlihat bahwa algoritma machine learning dari support vector machine lebih banyak unggul sedangkan artificial neural network hanya unggul di train accuracy yang dihasilkan. Namun, jikalau dilakukan tuning parameter model accuracy test terlihat bahwa artificial neural network lebih unggul dari support vector machine. Sehingga dapat dikatakan setelah dilakukan tuning parameter artificial neural network dapat menandingin support vector machine yang dimana accuracy test sangat mempengaruhi accuracy confusion matrix yang dihasilkan. Walaupun SVM dan ANN lebih unggul dibanding yang lain, namun dapat terlihat bahwa dari prediksi siste yang dibuat berdasarkan data pasien diabetes dan non diabetes memiliki hasil yang sama dari masing-masing algoritma machine learning yang digunakan.

Dengan demikian klasifikasi algoritma machine learning yang bagus untuk digunakan pada penelitian ini saat sebelum dilakukan tuning parameter model yaitu support vector machine. Namun jikalau setelah dilakukan tuning parameter model artificial neural network dapat menandingi support vector machine. Namun tidak memperburuk juga dari algoritma naïve bayes, decision tree dan juga random forest karena masih dapat dikatakan bagus untuk digunakan karena memiliki hasil yang sama saat memprediksi data pasien diabetes dan non diabetes.

V. DAFTAR PUSTAKA

[1] Amei, W., Huailin, D., Qingfeng, W., & Ling, L. 2011. "A survey of application-level protocol identification based on machine learning". *2011 International Conference on Information Management, Innovation Management and Industrial Engineering*, Volume 3, pp. 201–204.

[2] Fatimah, Restyana Noor. 2015. "DIABETES MELITUS TIPE 2". *J MAJORITY,* Volume 4(5), pp. 93-101.

[3] Husada, Hendry Cipta. 2021. "Analisis Sentimen Pada Maskapai Penerbangan di Platform Twitter Menggunakan Algoritma Support Vector Machine (SVM)". *TEKNIKA*, Volume 10(1), pp. 18-26, doi: 10.34148/teknika.v10i1.311.

[4] Ispriantari Aloysia. 2017. "PENERIMAAN DIRI PADA REMAJA DENGAN DIABETES TIPE 1 DI KOTA". *Dunia Keperawatan*, Volume 5(2), pp. 115-120.

[5] I. Klyueva. 2019. "Improving Quality of the Multiclass SVM Classification Based on the Feature Engineering". *Proc. - 2019 1st Int. Conf. Control Syst. Math. Model. Autom. Energy Effic. SUMMA 2019*, pp. 491–494, doi: 10.1109/SUMMA48161.2019.8947599.

[6] Kurniawaty Evi. 2016. "Faktor-Faktor yang Berhubungan dengan Kejadian Diabetes Melitus Tipe II". Majority, Volume 5(2), pp. 27-31.

[7] Mahesh Batta.2018. "Machine Learning Algorithms - A Review". *International Journal of Science and Research (IJSR)*. Volume 9(1), pp. 381-386, doi: 10.21275/ART20203995.

[8] Roihan Ahmad. 2019. "Pemanfaatan Machine Learning dalam Berbagai Bidang: Review paper". *IJCIT*, Volume 5(1), pp. 75-82.

[9] Slamet S. 2008. *Diet pada diabetes Dalam Noer dkk.Buku ajar ilmu penyakit dalam*. Edisi III.Jakarta: Balai Penerbit FK-ill.

[10] Thupae, R., Isong, B., Gasela, N., & Abu Mahfouz, A. M. 2018. "Machine Learning Techniques for Traffic Identification and Classifiacation in SDWSN: A Survey". *IECON 2018 - 44th Annual Conference of the IEEE Industrial Electronics Society*, 4645–4650. https://doi.org/10.1109/IECON.2018.859 1178.