**Perbandingan Prediksi dari Dataset Breast Cancer menggunakan Random Forest Classifier dan K-Nearest Neighbour**

**Laporan Akhir Mata Kuliah IN317A MACHINE LEARNING**

Logo

Description automatically generated

**Disusun Oleh:**

Muhammad Faiq Adhitya Faqih ( 672020017 )

Adelliya Dewi Saputri ( 672020107 )

Krisna Adi Saputra ( 672020178 )

**Dosen Pembimbing:**

**Hindriyanto Dwi Purnomo, Ph.D.**

**Program Studi Teknik Informatika**

**Fakultas Teknologi Informasi**

**Universitas Kristen Satya Wacana**

**Salatiga**

**2022**

DAFTAR ISI

[DAFTAR ISI i](#_Toc121001701)

[DAFTAR GAMBAR ii](#_Toc121001702)

[BAB I PENDAHULUAN 1](#_Toc121001703)

[1.1. Latar Belakang 1](#_Toc121001704)

[1.2. Rumusan Masalah 2](#_Toc121001705)

[1.3. Tujuan Penelitian 2](#_Toc121001706)

[1.4. Manfaat Hasil Penelitian 2](#_Toc121001707)

[BAB II TINJAUAN PUSTAKA 3](#_Toc121001708)

[2.1. Breast Cancer Wisconsin 3](#_Toc121001709)

[2.2. K-Nearest Neighbors 4](#_Toc121001710)

[2.3. Analisis Random Forest 4](#_Toc121001711)

[2.4. Cara Kerja Random Forest 5](#_Toc121001712)

[BAB III METODOLOGI PENELITIAN 6](#_Toc121001713)

[3.1 Tahapan Penelitian 6](#_Toc121001714)

[3.2 Teknik Pengumpulan Data 7](#_Toc121001715)

[BAB IV HASIL DAN PEMBAHASAN 9](#_Toc121001716)

[4.1. Hasil Penelitian 9](#_Toc121001717)

[4.2. Pembahasan Pengumpulan Data 10](#_Toc121001718)

[4.3. Pembahasan Penelitian 11](#_Toc121001719)

[BAB V KESIMPULAN DAN SARAN 18](#_Toc121001720)

[5.1. Kesimpulan 18](#_Toc121001721)

[5.2. Saran 18](#_Toc121001722)

[DAFTAR PUSTAKA 19](#_Toc121001723)

DAFTAR GAMBAR

[Gambar 1 12](file:///C:\Users\Faiq\Downloads\otw%20jadi.docx#_Toc120999285)

[Gambar 2 12](file:///C:\Users\Faiq\Downloads\otw%20jadi.docx#_Toc120999286)

[Gambar 3 14](#_Toc120999287)

BAB I

PENDAHULUAN

1. **Latar Belakang**

Sekarang ini ilmu pengetahuan berkembang dengan pesat terutama perkembangan dalam bidang teknologi informasi. Perkembangan yang sangat pesat dalam bidang teknologi informasi membuat kehidupan kehidupan manusia semakin mudah. Akhir-akhir ini machine learning mengalami perkembangan yang cukup pesat karena banyak orang yang melakukan pengembangan dalam hal tersebut karena dapat membantu banyak software untuk berkembang lebih jauh lagi.

Kanker merupakan suatu penyakit yang ditandai pertumbuhan sel tubuh tertentu yang tidak terkendali yang berakibat merusak sel dan jaringan tubuh lainnya. Seseorang yang mengidap penyakit kanker memiliki kemungkinan terburuk yaitu kematian. Sel kanker terbagi menjadi dua yaitu sel kanker ganas dan sel kanker jinak. Terdapat beberapa penyebab yang dapat seseorang mengidap penyakit kanker yaitu, senyawa kimia (zat karsinogen), faktor fisika (bom atom atau radioterapi agresif), virus, dan hormon. Kanker memiliki banyak sekali jenis dan penamaan dari kanker biasanya berdasarkan dari bagian yang terkena kanker. Beberapa contoh kanker yang ada yaitu, kanker paru-paru, kanker lambung, serviks, payudara, dan masih banyak lagi jenis kanker yang lainnya. Kanker payudara merupakan salah satu kanker yang banyak diderita oleh wanita di dunia. menurut WHO pada tahun 2018 terdapat 58.256.000 kasus terjadi di negara berkembang dan menyebabkan kematian. Untuk itu dibutuhkan pengenalan kanker yang diderita apakah penderita kanker jinak atau ganas agar memiliki penanganan yang tepat. Dengan adanya prediksi menggunakan machine learning dapat mendeteksi kategori kanker payudara berdasarkan variabel-variabel yang sudah diolah pada machine learning sehingga dapat mengenali jenis kanker yang diderita.

Machine learning memiliki banyak sekali algoritma prediksi yang dapat digunakan. Pada penelitian ini kami menggunakan algoritma K-Nearest Neighbour (KNN) dan algoritma random forest. Pemilihan algoritma K-Nearest Neighbour (KNN) karena KNN memprediksi data berdasarkan kedekatan data yang sudah dilatih sedangkan random forest kami pilih karena menghasilkan kesalahan yang relatif rendah, kinerja klasifikasi yang baik, mampu menangani sejumlah besar data pelatihan secara efisien, serta metode yang efektif untuk mengestimasi missing data

1. Rumusan Masalah

Berdasarkan pada latar belakang yang telah dipaparkan, maka rumusan masalah dari penelitian ini adalah :

1. Mengetahui tingkat accuracy data tiap model
2. Perbandingan tinggkat accuracy data antara model KNN dan random forest
3. Mengetahui model terbaik dari data yang digunakan antara KNN dan random forest
4. Tujuan Penelitian

Tujuan yang ingin dicapai dari penelitian ini antara lain :

1. Mengetahui hasil akurasi tiap model Random Forest dan KNN
2. Mengetahui tingkat accuracy data antar 2 model2
3. mengetahui algoritma mana yang memiliki akurasi yang lebih baik antara algoritma K -Nearest Neighbour (KNN) dengan algoritma random forest untuk dataset breast cancer(kanker payudara)
4. Manfaat Hasil Penelitian

Dari penelitian ini kita bisa mengetahui algoritma mana yang memiliki akurasi prediksi paling tinggi untuk melakukan prediksi pada dataset breast cancer (kanker payudara). Dengan mengetahui algoritma mana yang memiliki akurasi prediksi yang paling tinggi membuat kemungkinan prediksi meleset semakin sedikit sehingga jika terdapat data baru kita dapat mengetahui seseorang tersebut mengidap penyakit breast cancer (kanker payudara) atau tidak

BAB II

TINJAUAN PUSTAKA

1. Breast Cancer Wisconsin

Breast Cancer Wisconsin (Diagnostic) Dataset adalah nama data set yang dipakai. Disini biasa disebut dengan istilah Kanker payudara. (Setyanto, Arief, 2021) Kanker payudara masih menjadi penyakit dengan angka kematian pada wanita yang cukup tinggi. Berdasarkan data dari WHO (*World Health Organization*) merupakan penyakit Kanker payudara dengan angka kematian sebesar 42,5% di dunia pada tahun 2018 dengan jumlah rata-rata kematian setiap tahunnya sebesar 9,3 wanita. Risiko terkena kanker payudara dipengaruhi oleh perbedaan geografis (Maulina, Inas Ulvy, 2015). Pada Negara berkembang jumlah kematian yang disebabkan oleh kanker payudara lebih tinggi karena pendeteksian kanker secara dini masih dalam tahap pengembangan [KIN-05]. Pada tahun 2006 di Amerika, terdapat 212.920 kasus baru kanker payudara pada wanita dan 1.720 kasus baru pada pria, dengan 40.970 kasus kematian pada wanita dan 460 kasus kematian pada pria. Di Indonesia, kanker payudara menempati urutan kedua setelah kanker leher rahim. Kejadian kanker payudara di Indonesia sebesar 11% dari seluruh kejadian kanker.

Data sumber penelitian yang digunakan dalam pengujian akurasi diagnosis kanker payudara tersebut sebagian besar menggunakan data publik dari UCI Machine Learning Repository dataset breast cancer. Dataset kanker payudara di repository UCI terbagi menjadi 6, yaitu breast cancer, breast Cancer Wisconsin (original) atau WBCD, Breast Cancer Winsonsin (Prognostic) atau WPBC, Breast cancer Winconsin (Diagnostic) atau WDBC dan Mammographic Mass dan Breast Cancer Coimbra [10]. Pada penelitian sebelumnya yang dilakukan oleh Wahyuni, menguji uji akurasi diagnosis kanker payudara menggunakan algoritma SMO, MLP, C4.5 dan Naive Bayes pada dataset WBCD dan menunjukkan bahwa SMO memperoleh akurasi diagnosis tertinggi sebesar 97.6574%

1. K-Nearest Neighbors

(Putra, M Ikhsan Perdana, 2019) Algoritma K-Nearest Neighbor (KNN) adalah [algoritma](https://www.trivusi.web.id/2022/06/mengenal-algoritma.html) machine learning yang bersifat non-parametric dan lazy learning. Algoritma k-nearest neighbor (KNN) merupakan algoritma clustering yang sangat sederhana dengan cara mengelompokkan data baru dengan K tetangga terdekat. Metode yang bersifat non-parametric memiliki makna bahwa metode tersebut tidak membuat asumsi apa pun tentang distribusi data yang mendasarinya. Dengan kata lain, tidak ada jumlah parameter atau estimasi parameter yang tetap dalam model, terlepas data tersebut berukuran kecil ataupun besar.

Algoritma non-parametric seperti KNN menggunakan sejumlah parameter yang fleksibel, dan jumlah parameter seringkali bertambah seiring data yang semakin banyak. Algoritma non-parametric secara komputasi lebih lambat, tetapi membuat lebih sedikit asumsi tentang data.Algoritma KNN juga bersifat lazy learning, yang artinya tidak menggunakan titik data training untuk membuat model. Singkatnya pada algoritma KNN tidak ada fase training, kalaupun ada juga sangat minim. Semua data training digunakan pada tahap testing. Hal ini membuat proses training lebih cepat dan tahap testing lebih lambat dan cenderung ‘mahal’ atau membutuhkan banyak cost dari sisi waktu dan memori.

1. Analisis Random Forest

Random Forest (RF) adalah algoritma berdasarkan pohon klasifikasi dan regresi, menggunakan metode pembelahan rekursif untuk mencapai node akhir dalam struktur pohon (Muntiari, Novita Ranti, 2022). Algoritma Random Forest telah menunjukkan beberapa keunggulan, termasuk mampu menghasilkan kesalahan yang relatif rendah, kinerja klasifikasi yang baik, mampu menangani sejumlah besar data pelatihan secara efisien, serta metode yang efektif untuk mengestimasi missing data.

Prediksi kanker payudara dilakukan dengan bantuan aplikasi ? Google Collab memanfaatkan Python dan library scikitlearn. Model Random Forest dilatih dengan data sebanyak 75% dan diuji dengan data sebanyak 25%. Random Forest memiliki beberapa pohon, dan attribute dipilih secara random dengan memakai bagging. Random Forest memakai sekumpulan pohon di mana setiap pohon tersebut memiliki nilai vektor arbitrary yang di sampel secara terpisah untuk semua pohon di Random Forest. Fase algoritma Random Forest dibagi menjadi 3 fase, yang pertama adalah pohon sebanyak i dibuat. Fase kedua menggabungkan semua pohon menjadi Random Forest. Yang terakhir adalah memberikan output. Output dari Random Forest dihasilkan melalui hasil voting dari setiap pohon.

1. Cara Kerja Random Forest

Random Forest bekerja dalam dua fase. Fase pertama yaitu menggabungkan sejumlah N decision tree untuk membuat Random Forest. Kemudian fase kedua adalah membuat prediksi untuk setiap tree yang dibuat pada fase pertama.

Cara kerja algoritma Random Forest dapat dijabarkan dalam langkah-langkah berikut:

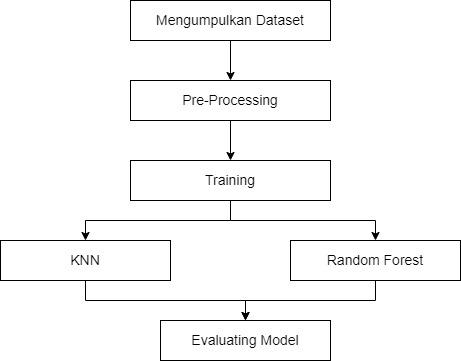
1. Algoritma memilih sampel acak dari dataset yang disediakan.
2. Membuat decision tree untuk setiap sampel yang dipilih. Kemudian akan didapatkan hasil prediksi dari setiap decision tree yang telah dibuat.
3. Dilakukan proses voting untuk setiap hasil prediksi. Untuk masalah klasifikasi menggunakan modus (nilai yg paling sering muncul), sedangkan untuk masalah regresi akan menggunakan mean (nilai rata-rata).
4. Algoritma akan memilih hasil prediksi yang paling banyak dipilih (vote terbanyak) sebagai prediksi akhir.

BAB III

METODOLOGI PENELITIAN

1. Tahapan Penelitian

Model *machine learning* dengan algoritma KNN dan *Random Forest* digunakan untuk menyelesaikan *mini project* berbagai permasalahan dalam melakukan analisis. Berikut tahapan-tahapan yang akan dilakukan :



Penjelasan dari tahapan pada Gambar 2 yang ditampilkan di atas sebagai berikut :

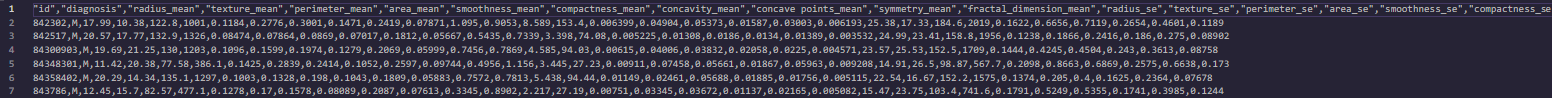
1. Mengumpulkan *dataset* yaitu mempersiapkan *dataset* yang akan diuji, data tersebut berasal dari *UCI Machine Learning Repository*.
2. *Pre-Processing* yaitu proses melihat data untuk menampilkan isi data yang akan diuji, Normalisasi dan Transformasi untuk melakukan pengecekan data yang bernilai null atau kosong atau *missing value* atau pembersihan data lalu melakukan normalisasi perbaikan data dengan rumus *min-max* yaitu persamaan *min-max* :

𝑁𝑜𝑟𝑚𝑎𝑙𝑖𝑠𝑎𝑠𝑖= 𝑑𝑎𝑡𝑎𝑥−𝑑𝑎𝑡𝑎𝑚𝑖𝑛

1. dan melakukan transformasi sesuai kebutuhan saat proses pengujian klasifikasi, dan menampilkan visualisasi data.
2. Training yaitu melakukan pengujian *testing data* dan *training data* yang ditentukan 75% sebagai *training data* dan 25% sebagai *testing data*.
3. KNN (*K-Nearest Neighbor*) yaitu proses klasifikasi pada data yang telah dilakukan testing pada tahap training kemudian akan diujikan dengan metode *K-Nearest Neighbor*. KNN juga termasuk bagian dari *supervised learning*. Menggunakan hasil yang datang dari *query instance* yang kemudian dikategorikan berdasarkan suara terbanyak dari kategori di KNN. cara kerja KNN adalah dengan melakukan *voting* dari jawaban sesuai dengan nilai k objek.
4. *Random Forest* yaitu proses klasifikasi pada data yang telah dilakukan testing kemudian akan diujikan dengan metode *Random Forest*. *Random Forest* memiliki beberapa pohon dan *attribute* dipilih secara *random*.
5. *Evaluating model* yaitu melakukan perhitungan dan menampilkan hasil *accuracy* atau *precision* atau *recall* dari data yang sudah di training. dan membandingkan tingkat *accuracy* antar dua metode berupa visualisasi dan tabel.
6. Teknik Pengumpulan Data

Data set yang digunakan, diperoleh dari *repository dataset* yang ada ada di website *UCI Machine Learning Repository*. *UCI Machine Learning Repository* adalah website yang berisi arsip dengan koleksi *dataset* mengelola 622 lebih sekumpulan data sebagai layanan untuk komunitas *machine learning*.

*Dataset* yang digunakan untuk melakukan analisa memilih *Breast Cancer Diagnosis* yang diunduh dalam bentuk format “.data” yang diberikan terakhir di unduh secara total 569 data dan 32 kolom pada tanggal 27 November 2022. Yang pada awal pengunduhan berupa format “.data” perlu dijadikan menjadi “.csv” agar bisa dapat diakses pada pengerjaan analisis dan dapat dibuka melalui *library* pandas pada python. Pada saat pemrosesan import ke python menjadi *dataframe* kolom berubah menjadi 33 kolom, Hal ini dikarenakan di file .csv pada string value untuk kolom terakhir yaitu “*fractal\_dimension\_worst*”, setelah string value tersebut muncul karakter koma. Yang membuat Python menganggap koma tersebut menandakan kolom yang baru tetapi tidak ada nama. Oleh sebab itu *python* memberikan nama kolom menjadi “*Unnamed:* 32” yang artinya tidak ada nama, angka 32 berasal dari indeks kolom yang dimulai dari 0 berikut gambaran datanya :



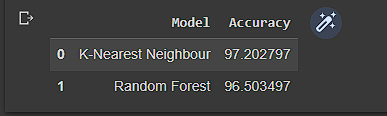
Maka *attribute* “*Unnamed:* 32” Tidak digunakan saat pemrosesan analisis, sehingga *attribute* tersebut dihilangkan atau dihapus. Selain itu ada *attribute* yang penting yang digunakan saat proses analisis seperti pada *attribute* kolom “*diagnosis*” dengan *value* “B” artinya *Benign* atau kanker jinak dan *value* “M” artinya *Malignant* atau kanker ganas.

BAB IV

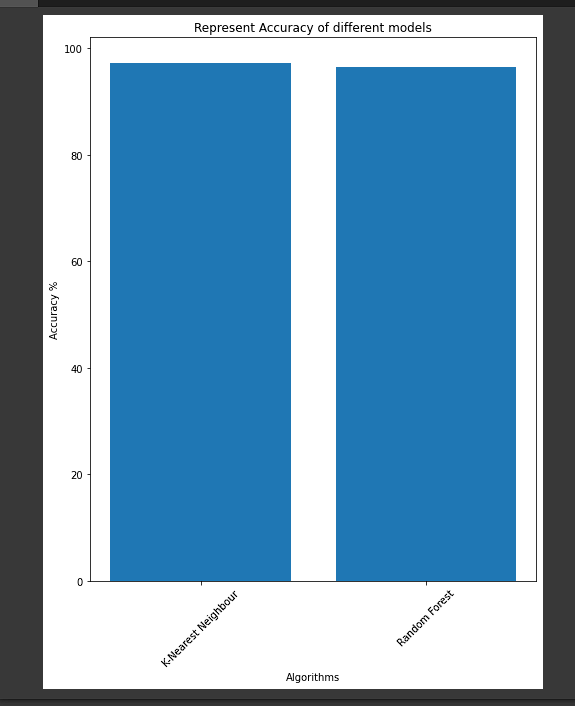
HASIL DAN PEMBAHASAN

1. Hasil Penelitian

Berikut merupakan hasil akurasi tabel dan diagram tiap model *Random Forest* dan KNN :



Gambar 1



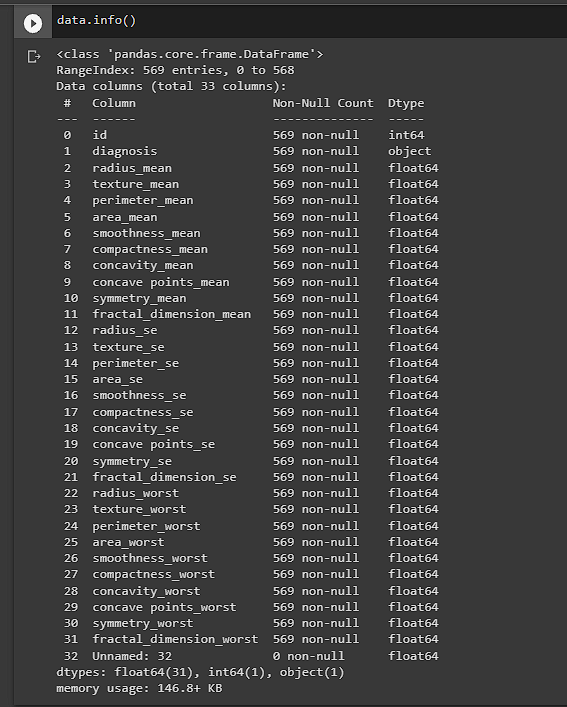
Gambar 2

Gambar 1 dan gambar 2 menunjukan hasil akurasi tiap model *Random Forest* dan KNN. Untuk nilai akurasi *Random Forest* memiliki akurasi sebesar 96.5034 dan untuk akurasi KNN memiliki akurasi sebesar 97.2027. Dari kedua nilai tersebut dapat disimpulkan bahwa pada kasus dataset kanker payudara (*Breast Cancer)* yang paling unggul adalah model KNN.

1. Pembahasan Pengumpulan Data

Setelah mengambil data dari web archive.ics.uci.edu, maka didapatkan *dataset* breast cancer (kanker payudara). *Data Set* disediakan dalam bentuk .txt lalu dijadikan CSV (*Comma Separated Values*). Gambar 3 merupakan informasi *dataset* kanker payudara yang diperoleh dari data.info() Gambar 3 memiliki 33 kolom. Berikut penjelasan terhadap kolom di gambar 3 :

1. Kolom pertama “#” adalah menyatakan nilai *index*. *Index* dimulai dari 0 sehingga kolom pertama memiliki nilai *index* 0.
2. Kolom kedua “*Coloumn”* adalah nama kolom menjadi *atribut* pada *dataset*.
3. Kolom ketiga “*Non-Null Count*”. di kolomtersebut semua *atribut* memiliki nilai 569 *non-null* kecuali pada *atribute* terakhir yaitu “*Unnamed: 32*”. Dari kolom tersebut dapat disimpulkan bahwa 569 data kolom tersebut memiliki nilai atau tidak kosong. Karena ada data set yang mempunyai nilai kosong seperti pada *attribute* “*Unnamed: 32”* maka nilai 0.
4. Kolom “*Dtype”* adalah memberi info tentang tipe data. *Int*64 yaitu integer. *object* yaitu *string*. *Float*64 yaitu angka yang memiliki nilai koma.



Gambar 3

1. Pembahasan Penelitian

Sebelum menguji model KNN dan *Random Forest* dengan *Scikit-learn*, *dataset* kanker payudara harus dibersihkan terlebih dahulu, agar *dataset* dapat dipakai tanpa kendala *error* ketika menjalankan *codingan* dan membuang *attribute* yang tidak terpakai dalam saat menganalisis atau dalam klasifikasi.

1. Data Cleaning, Normalisasi Data, dan Pembagian Data

*Dataset* kanker payudara yang dipakai memiliki 33 *attribute* dan dapat dilihat dari gambar 3. Untuk *data cleaning* hanya perlu menghapus kolom “*id*” dan “*Unnamed:* 32” . Serta melakukan visualisasi data dari kolom/*attribute* diagnosis. Setelah melakukan *cleaning data* selanjutnya adalah melakukan normalisasi data. Setelah normalisasi, *dataset* akan dilakukan *split function*, dataset akan dibagi menjadi *training data* sebesar 75% dan *test data* sebesar 25% dengan *random state* sebesar 42. *Random state* digunakan untuk pembagian data akan tetap sama sehingga *training data* dan *test data* menjadi konsisten jika *function* dijalankan lagi.

1. Membangun Model KNN dan *Random Forest Classifier*

Model KNN dan *Random Forest* dibangun dengan *library Scikit-learn*. Karena *Random Forest* memanfaatkan *random state* yang sebesar 42. Setelah model-model tersebut dibangun, model-model tersebut diberi *data training* dan *data testing* yang digunakan untuk melihat tingkat akurasi kedua model tersebut.

1. Implementasi di *Google Collab*

Pemakaian *platform Google Collab* untuk project mini memiliki tujuan untuk membangun model *machine learning*. Berikut tahapan memakai *Google Collab :*

1. Melakukan import untuk *library* yang dibutuhkan. Pada project mini ini *library Scikit-learn, Pandasm Matplotlib, Searbon* digunakan untuk keperluan *machine learning, dataset, graph* dan *statistic*.

import numpy as np *# linear algebra*

import pandas as pd *# data processing, CSV file I/O (e.g. pd.read\_csv)*

import seaborn as sns *# data visualization library*

import matplotlib.pyplot as plt

*#Machine Learning Libraries*

from sklearn.linear\_model import LogisticRegression

from sklearn.neighbors import KNeighborsClassifier

from sklearn.naive\_bayes import GaussianNB

from sklearn.svm import SVC

from sklearn.tree import DecisionTreeClassifier

from sklearn.ensemble import RandomForestClassifier

from sklearn.model\_selection import train\_test\_split

from sklearn.metrics import r2\_score,mean\_squared\_error

from sklearn.metrics import confusion\_matrix

from sklearn.metrics import accuracy\_score, confusion\_matrix, precision\_score, plot\_confusion\_matrix,f1\_score, recall\_score, classification\_report

1. Mengirimkan *dataset* ke dalam *data frame* dengan *library Pandas.* berikut adalah code yang dipakai.

data = pd.read\_csv("cancer.csv")

data

1. Melakukan *data cleaning*. Tahap ini menghilangkan atau penghapusan kolom dan transformasi kolom. berikut code untuk *data cleaning* dan memeriksa apakah data yang masih bernilai kosong atau *non-value*

data.drop(["id","Unnamed: 32"],axis=1,inplace=True)

data.isna().sum()

setelah melakukan *data cleaning,* melakukan normalisasi dan transformasi. berikut code untuk melakukan normalisasi dan transformasi data.

y = data.diagnosis.values

x\_data = data.drop(["diagnosis"],axis=1)

x = (x\_data - np.min(x\_data))/(np.max(x\_data)-np.min(x\_data))

x.head()

1. Membagi *data frame* menjadi *training data set* dan *testing data set*. *code* dibawah menunjukan cara untuk memakai *function train\_test\_split* dari *library Scikit-learn* untuk membagi data. Berisi parameter ukuran *testing data* sebesar 25% dan *random state* 42.

x\_train, x\_test, y\_train, y\_test = train\_test\_split(x,y,test\_size = 0.25,random\_state=42)

1. Membangun model M*achine Learning* dengan *Scikit-learn.* *code* di bawah caramembuat model *random Forest* dan KNN. Setelah membuat model dilatih dengan *training data.* Lalu melakukan tingkat akurasi tiap model yang sudah dibangun dengan *function predict*.

rf = RandomForestClassifier(n\_estimators=10,random\_state=1)

rf.fit(x\_train,y\_train)

# Accuray On Test Data

predictions = rf.predict(x\_test)

accuracy = accuracy\_score(y\_test, predictions)

print("Accuracy on Test Data Random Forest Classifier algo: ",rf.score(x\_test,y\_test))

rf\_acc\_score = rf.score(x\_test,y\_test)

# print(f"Accuracy on Test Data: .{accuracy\*100}%")

print(f"Precision Score: {precision\_score(y\_test, predictions)}")

print(f"Recall Score: {recall\_score(y\_test, predictions)}")

print(f"F1 Score: {f1\_score(y\_test, predictions)}")

plot\_confusion\_matrix(rf, x\_test, y\_test)

plt.title("Confusion Matrix for Test Data")

plt.show()

print(classification\_report(y\_test, predictions))

print()

# Accuray On Whole Data

predictions = rf.predict(x.values)

accuracy = accuracy\_score(y, predictions)

print(f"Accuracy on Whole Data: {accuracy}")

print(f"Precision Score: {precision\_score(y, predictions)}")

print(f"Recall Score: {recall\_score(y, predictions)}")

print(f"F1 Score: {f1\_score(y, predictions)}")

plot\_confusion\_matrix(rf, x.values, y)

plt.title("Confusion Matrix for Whole Data")

plt.show()

print(classification\_report(y, predictions))

knn = KNeighborsClassifier(n\_neighbors=10) #We are building our model

knn.fit(x\_train,y\_train) #We are training our model

print("Accuracy on Test Data K Neighbors Classifier algo: {}".format(knn.score(x\_test,y\_test)))

knn\_acc\_score = knn.score(x\_test,y\_test)

#Accuray On Test Data

predictions = knn.predict(x\_test)

accuracy = accuracy\_score(y\_test, predictions)

#print(f"Accuracy on Test Data: {accuracy\*100}%")

print(f"Precision Score: {precision\_score(y\_test, predictions)}")

print(f"Recall Score: {recall\_score(y\_test, predictions)}")

print(f"F1 Score: {f1\_score(y\_test, predictions)}")

plot\_confusion\_matrix(knn, x\_test, y\_test)

plt.title("Confusion Matrix for Test Data")

plt.show()

print(classification\_report(y\_test, predictions))

print()

# Accuray On Whole Data

predictions = knn.predict(x.values)

accuracy = accuracy\_score(y, predictions)

print(f"Accuracy on Whole Data: {accuracy\*100}%")

print(f"Precision Score: {precision\_score(y, predictions)}")

print(f"Recall Score: {recall\_score(y, predictions)}")

print(f"F1 Score: {f1\_score(y, predictions)}")

plot\_confusion\_matrix(knn, x.values, y)

plt.title("Confusion Matrix for Whole Data")

plt.show()

print(classification\_report(y, predictions))

1. Menampilkan hasil akurasi dalam bentuk visual dari *Random Forest* dan KNN

#Random Forest Accuracy Visual

y\_pred = rf.predict(x\_test)

y\_true = y\_test

cm = confusion\_matrix(y\_true, y\_pred)

#visualize

f, ax = plt.subplots(figsize=(5,5))

sns.heatmap(cm,annot = True, linewidths=0.5,linecolor="red",fmt = ".0f",ax=ax)

plt.xlabel("y\_pred")

plt.ylabel("y\_true")

plt.show()

#KNN Accuracy Visual

score\_list = []

for each in range(1,15):

knn2 = KNeighborsClassifier(n\_neighbors = each)

knn2.fit(x\_train,y\_train)

score\_list.append(knn2.score(x\_test,y\_test))

#visualize

plt.plot(range(1,15),score\_list)

plt.xlabel("k values")

plt.ylabel("accuracy")

plt.savefig('plot')

plt.show()

1. Melakukan evaluasi model dengan membandingkan akurasi tiap model antara *Random Forest* dengan KNN secara menampilkan angka dan visual nya.

model\_ev = pd.DataFrame({'Model': ['K-Nearest Neighbour','Random Forest'],

'Accuracy': [knn\_acc\_score\*100,rf\_acc\_score\*100]})

model\_ev

plt.figure(figsize=(8,10))

plt.title("Represent Accuracy of different models")

plt.ylabel("Accuracy %")

plt.xlabel("Algorithms")

plt.xticks(rotation=45)

plt.bar(model\_ev['Model'],model\_ev['Accuracy'])

plt.show()

BAB V

KESIMPULAN DAN SARAN

1. Kesimpulan

Berdasarkan percobaan yang sudah dilakukan dapat disimpulkan bahwa algoritma prediksi K-Nearest Neighbour memiliki tingkat akurasi prediksi lebih tinggi 0,7 dari pada algoritma prediksi Random Forest. Algoritma K-Nearest Neighbour memiliki nilai prediksi 97,2 sedangkan algoritma prediksi Random Forest memiliki nilai prediksi 96,5. Jadi apabila ingin melakukan prediksi terkait dataset breast cancer lebih baik menggunakan algoritma K-Nearest Neighbour karena memiliki tingkat prediksi yang lebih baik 0,7 dibandingkan dengan Random Forest. Dengan demikian kita dapat melakukan prediksi untuk dataset breast cancer dengan tingkat kebenaran nilai prediksi yang tinggi atau nilai dari kebenaran dari prediksi hampir dipastikan benar.

1. Saran

Pada penelitian selanjutnya terkait prediksi dengan dataset breast cancer agar mendapatkan algoritma yang memiliki nilai prediksi mendekati 100 disarankan agar menggunakan berbagai macam algoritma. Dengan berbagai macam algoritma prediksi dicoba pada dataset breast cancer maka dapat diketahui algoritma yang paling cocok atau memiliki nilai prediksi yang paling mendekati 100.

DAFTAR PUSTAKA

Maulina, Inas Ulvy. (2015). Klasifikasi Kanker Payudara Menggunakan Decision Tree Dengan Algoritma Iterative Dichotomizer-3 ( ID-3 ). 1-8.

Muntiari, Novita Ranti. (2022). Klasifikasi Penyakit Kanker Payudara Menggunakan Perbandingan Algoritma Machine Learning. *Jurnal Ilmu Komputer dan Teknologi, 3*(1), 1-6.

Setyanto, Arief. (2021). Sistem Klasifikasi pada Penyakit Breast Cancer dengan Menggunakan Metode Naïve Bayes. *Creative Information Technology Journal*, 109.

Putra, M Ikhsan Perdana. (2019). Implementasi Algoritma Modified K-Nearest Neighbor ( MKNN ) untuk Klasifikasi Penyakit Kanker Payudara Tugas Akhir diajukan untuk memenuhi salah satu syarat memperoleh gelar sarjana dari Program Studi S1 Ilmu Komputasi M Ikhsan Perdana Putra Program Studi. *E-proceeding of Engineering*, 2431-2441.

Raharjo, B. (2019). KUMPULAN SOLUSI PEMROGRAMAN PYTHON. Bandung: PENERBIT INFORMATIKA BANDUNG.

Angkasa, V., & Junifer Pangaribuan, J. (2022). KOMPARASI TINGKAT AKURASI RANDOM FOREST DAN KNN UNTUK MENDIAGNOSIS PENYAKIT KANKER PAYUDARA, 7.