

# KLASIFIKASI PENYAKIT STROKE MENGGUNAKAN ALGORITMA KNN DENGAN OPTIMASI TEKNIK *BAGGING*

Achmad Mujammil (1610651042)<sup>1</sup>, Hardian Oktavianto<sup>2</sup>,  
Ginanjar Abdurrahman<sup>3</sup>.

Program Studi Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Jember  
[achmadmujammil2@gmail.com](mailto:achmadmujammil2@gmail.com)<sup>1</sup>, [hardian@unmuhjember.ac.id](mailto:hardian@unmuhjember.ac.id)<sup>2</sup>,  
[abdurrahmanganjar@unmuhjember.ac.id](mailto:abdurrahmanganjar@unmuhjember.ac.id)<sup>3</sup>.

## Abstrak

Stroke merupakan suatu penyakit yang sangat serius dalam bidang kedokteran, dikarenakan dampak penyakit ini sangatlah fatal, karena dapat menyerang secara fisik dan mental dan lebih parahnya lagi hingga dapat menyebabkan kematian secara mendadak, penanganan Stroke ini perlu di tangani secara cepat dan tanggap karena dapat merusak pada inti otak sehingga kemungkinan dapat menimbulkan komplikasi. Pada penelitian ini peneliti ingin membuktikan bahwa dalam menentukan seseorang terjangkit penyakit stroke menggunakan algoritma klasifikasi *K-Nearest Neighbor* dengan optimasi *Bagging* ini bisa digunakan, dari prediksi tersebut dapat membantu untuk mempermudah pada bidang kesehatan dalam menentukan atau mengambil keputusan dengan cepat, dari hasil yang di dapatkan dari klasifikasi tersebut di dapatkan hasil Nilai akurasi, presisi dan recall yang tertinggi pada Algoritma *K-Nearest Neighbor* dengan optimasi *bagging* terdapat pada K-5 dengan nilai akurasi sebesar 99.93%, presisi sebesar 100.00% dan recall sebesar 98.25%.

Dari hasil pengujian, diketahui bahwa algoritma *K- nearest Neighbor* dengan optimasi *Bagging* ini sangat cocok digunakan untuk menentukan penyakit stroke.

**Kata Kunci:** *Klasifikasi, Bagging, K- Nearest Neighbour (KNN).*

## Abstract

*Stroke is a very serious disease in the field of medicine, because the impact of this disease is very fatal, because it can attack physically and mentally and even worse, it can cause sudden death, this stroke prevention needs to be handled quickly and responsively because it can damage the the core of the brain so that it is likely to cause complications. In this study the researcher wants to prove that in determining someone has a stroke using the K-Nearest Neighbor classification algorithm with Bagging optimization it can be used, from these predictions it can help make it easier for the health sector to determine or make decisions quickly, from the results obtained from this classification, the highest accuracy, precision and recall values are obtained in the K-Nearest Neighbor Algorithm with bagging optimization found in K-5 with an accuracy value of 99.93%, precision of 100.00% and recall before sar 98.25%.*

*From the test results, it is known that the K-nearest Neighbor algorithm with Bagging optimization it is very suitable for determining stroke.*

**Keywords:** *Classification, Bagging, K-Nearest Neighbor (KNN).*

## 1. PENDAHULUAN

stroke adalah salah satu penyakit yang mematikan di dunia dan di Indonesia sendiri berada peringkat nomor dengan angka kematian sekitar 21,1% kasus kematian dalam satu tahun terakhir, Maka dari itu diperlukan suatu penanganan atau tindakan pendeteksian secara dini agar risiko penyakit stroke bisa di atasi secara cepat sehingga dapat mengurangi tingkat resiko yang lebih parah, Pada penelitian sebelumnya yang dilakukan oleh Lishania dkk (2019) berjudul “Perbandingan Metode klasifikasi *Naive Bayes* Dan Metode *Decision Tree* Algoritma (J48) Pada Pasien Penyakit Stroke Di RSUD Abdul Wahab Sjahranie Samarinda “dengan kesimpulan algoritma *naive bayes* mendapat akurasi terbesar yaitu sebesar 81,25%. Adelina dkk (2018) melakukan penelitian berjudul “Klasifikasi Tingkat Resiko Penyakit Stroke Menggunakan *GA-Fuzzy Tsukamoto*” dengan ini dapat di simpulkan *Fuzzy Tsukamoto* juga dapat diterapkan pada kasus pendagnosisan risiko Stroke. Dari kedua penelitian tersebut dapat diketahui bahwa teknik data mining mampu untuk mengklasifikasikan penyakit stroke dengan baik Berdasarkan uraian di atas, maka dalam penelitian ini akan dilakukan optimasi algoritma KNN dengan teknik *Bagging* dalam mengklasifikasikan penyakit stroke dengan judul “Klasifikasi Penyakit Stroke Menggunakan Algoritma KNN dengan optimasi *Bagging*”

## 2. STROKE

Stroke adalah penyebab kematian ketiga di dunia angka kematian 18-37% untuk stroke pertama dan 62% untuk stroke ulang (Smeltzer, 2002), yang berarti pasien dengan stroke berulang memiliki risiko kematian ganda kali lebih besar dibandingkan dengan stroke, Tingginya angka kematian di Pasien dengan *stroke* atau *stroke* berulang memerlukan perhatian khusus karena diperkirakan 25% orang yang pulih dari stroke pertama mereka akan mengalami stroke berulang dalam 1-5 tahun (Jacob, 2001) Hal ini dibuktikan dengan studi pendahuluan yang dilakukan oleh para peneliti di pada tanggal 13 dan 15 Oktober 2011 di RSUP Dr. Kariadi Semarang, data diterima Pada tahun 2010, terdapat 1.009 pasien stroke yang mengalami stroke Rawat inap di dua bangsal saraf, yaitu Unit Stroke dan Saraf B1. 346 pasiendiantaranya menderita SH (*stroke hemoragik*) dan sisanya 663 penderita SNH (*stroke non hemoragik*). Tercatat juga bahwa 229 pasien memiliki *stroke* berulang, artinya 22,6% dari seluruh pasien *stroke* pernah menderita *stroke* Rawat inap dengan dr Kariadi merupakan pasien *stroke* berulang, Stroke dipicu oleh beberapa faktor risiko, semakin banyak faktor risiko yang dimiliki pasien, semakin tinggi kemungkinannya terjadinya *stroke* (Makmur, 2002). Stres adalah salah satu faktor risiko yang berada di urutan paling bawah sebagai faktor yang paling berpengaruh Terjadinya stroke (Utami, 2009), Fakta ini adalah salah satu alasan mengapa stres harus menjadi perhatian khusus setiap pasien stroke, Tingkat stres yang tinggi ini biasanya disebabkan Departemen Statistik menyatakan bahwa 31 juta orang atau 13,33% penduduk Indonesia hidup di garis kemiskinan per bulan di bawah Rp. 211.726.00 (Kementerian Kesehatan, 2009). Hanas dkk (dalam Fadlilah dkk. 2019) menjelaskan bahwa Stroke merupakan suatu penyakit yang dimana diakibatkan gangguan *vascular* dialami penderitanya. Dimana hal ini diakibatkan oleh pecahnya arteri pada bagian otak yang dapat menyebabkan gangguan fungsi otak pada lokasi pecahnya arteri tersebut, Gangguan fungsi otak sendiri terdiri dari kesadaran yang berkurang, gangguan persepsi, kesulitan berinteraksi, dan tingkat atensi penderita yang berkurang.

## 3. TINJAUAN PUSTAKA

### 1.1. DATA MINING

Data Mining adalah teknik dalam mendapatkan informasi baru, susunan tren data yang dibentuk pada banyaknya data dipelajari menggunakan teknik pengenalan pola serta ilmu statistik dan matematika, . Data mining merupakan rangkaian proses untuk menggali nilai tambah dari suatu kumpulan data berupa pengetahuan yang tidak diketahui secara manual. Dari definisi yang disajikan, hal-hal penting berkaitan dengan data.

1. Data mining adalah proses otomatis dari data yang ada.
2. Data yang akan diolah adalah data yang sangat besar.
3. Tujuan dari data mining adalah untuk mendapatkan kemungkinan hubungan atau pola

Menurut Utomo D,T (2020) Dalam data mining, ada istilah lain yang memiliki arti yang mirip dengan data mining, yaitu: Penemuan Pengetahuan dalam Basis Data (KDD). Data mining dan KDD memiliki tujuan yang sama yaitu: Menggunakan data yang tersedia dalam database dan kemudian mengolah data tersebut untuk memperoleh informasi baru yang berguna, banyak istilah lain yang memiliki arti serupa dengan penambangan data, misalnya Penemuan pengetahuan basis data, ekstraksi pengetahuan, analisis pola/data, arkeologi data, dan pengerukan data. Banyak orang menganggap penambangan data identik dengan istilah populer lainnya.

## 1.2. *K-Nearest Neighbor*

Algoritma *K-Nearest Neighbor* adalah algoritma yang digunakan dalam pengklasifikasian data berdasarkan data pembelajaran yang memiliki jarak ketetanggaan yang paling dekat dengan objek penelitian. Algoritma ini memiliki tujuan untuk mengklasifikasikan objek penelitian baru berdasar dari nilai dari masing – masing atribut (Puspitawuri dkk , 2019). Algoritma KNN (*K-Nearest Neighbor*) adalah algoritma yang digunakan untuk mengklasifikasi sebuah objek baru yang telah di pilih berdasarkan dari kumpulan atribut dan training samples, dimana dari hasil data sampel uji yang baru tersebut dilakukan pengklasifikasian berdasarkan mayoritas dari kategori pada *K-Nearest Neighbor*. proses pengklasifikasian ini, algoritma KNN (*K- Nearest Neighbor*) juga menggunakan sistem klasifikasi ketetanggaan sebagai nilai prediksi, Dimana jarak yang digunakan merupakan jarak *Euclidean Distance*. Atau jarak yang paling umum atau paling sering digunakan pada data numeric. (Febri Liatoni 2019).

Rumus pencarian jarak *Euclidean* adalah sebagai berikut:

$$\sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_n - y_n)^2}$$

Keterangan:

x = Data uji  
y = Data latih  
n = Data ke-n

## 1.3. *Bagging*

*Bagging* adalah teknik *ensemble aggregate bootstrap* awal mula teknik ini dikembangkan oleh Breiman yang di gunakan untuk meingkatkan tingkat akurasi klasifikasi, terhadap penerapan metode teknik bagging ini sendiri biasa dilakukan dengan menggabungkan pohon pelatihan lalu kemudian diset secara acak. Teknik ini sangat efektif karna telah dibuktikan oleh banyak para penelitian. Teknik ini sendiri memiliki beberapa kelebihan yaitu berguna untuk mengurangi varians dari algoritma dengan cara menyesuaikan antara estimasi dan hasil akhir yang diinginkan dari penambahan akurasi sebuah model. Menurut Sutton dkk (Prasetyo dan Prasetyo. 2020) tahapan bagging di bagi menjadi 2 yaitu:

### 1. Tahap *bootstrap*

- a. Mengambil random hasil data sampel dari data latih.
- b. Setelah mengambil data sampel, lalu susun *bag* terbaik.
- c. kemudian ulangi langkah a-b sebanyak n.

### 2. Tahap *Aggregating*

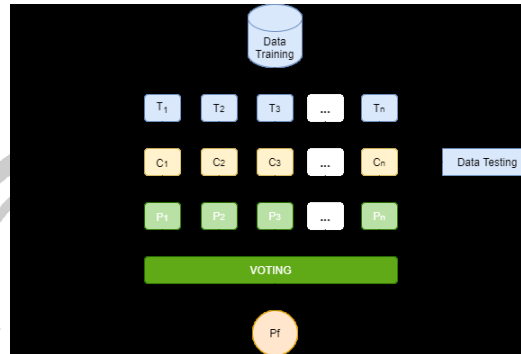
Pada tahapan ini dimana menggunakan teknik *majority vote* dimana dilakukannya tahap prediksi dari gabungan *bag* yang telah dibangun,

Algoritma Bagging ini merupakan suatu algoritma yang berasal dari teknik "*Bootstrap Aggregating*" yang dimana teknik ini sudah dilakukan penerapannya terhadap bidang lain, di antaranya biostatistik dan penginderaan secara jauh. Algoritma ini juga bisa diaplikasikan pada teknik kecerdasan buatan yang berguna untuk akuisisi informasi,

Algoritma ini bisa juga untuk dilakukan pengoptimalan terhadap kumpulan data pelatihan pada dimensi yang relatif lebih kecil, Di kutip dari Nugroho A dkk (2017) Teknik *Bagging* ini bekerja dengan cara membagi data latih untuk menjadi beberapa bagian, lalu dibuat model klasifikasidari tiap masing-masing bagian untuk mendapatkan hasil prediksi dari model tersebut. Hasil akhir yang diperoleh dengan voting,

Cara kerja bagging dapat dilihat pada ilustrasi pada Gambar di bawah ini :

Gambar Prinsip Kerja Algoritma Bagging



Pembahasan lebih jelasnya yaitu ada contoh data *training* kemudian dari data training tersebut dilakukan tahap *bootstrap resample* atau data di acak kemudian dari data acak tersebut dilakukan klasifikasi dengan menggunakan *tree* sehingga muncul muncul hasil prediksi pada masing-masing data yang yang telah di acak tadi lalu pada tahap terakhir kemudian dilakukan tahap *voting* dengan mencari nilai terbanyak yang akan dijadikan hasil prediksi akhir pada teknik *bagging*.

#### 1.4. **Klasifikasi**

Klasifikasi adalah salah satu teknik data mining. dan untuk klasifikasi itu sendiri merupakan suatu pengolahan menempatkan objek atau konsep tertentu kedalam rangkaian kategori yang ditampilkan berdasarkan objek yang digunakan. Ada beberapa teknik klasifikasi diantaranya jaringan syaraf tiruan, *naive bayes*, *support vector machine*, pohon keputusan dan *fuzzy*. Salah satu teknik klasifikasi yang paling umum digunakan adalah pohon keputusan. Klasifikasi itu sendiri dibagi menjadi dua fase, yaitu klasifikasi dan pembelajaran. Pada tahap pembelajaran, algoritma klasifikasi membuat model klasifikasi dengan menganalisis data pelatihan. Fase pembelajaran juga dapat dipandang sebagai fase membangun fungsi atau pemetaan  $y=f(x)$ , di mana  $y$  adalah kelas yang diprediksikan dan  $X$  adalah tupel yang ingin diprediksi oleh kelas tersebut.

#### 1.5. **Akurasi presisi dan recall**

Akurasi merupakan nilai rasio dari data benar yang terdeteksi di dalam data pengujian, presisi merupakan nilai sensitivitas dimana sensitivitas merupakan kemampuan tes untuk menunjukkan individu mana yang menderita sakit dari seluruh populasi yang benar-benar sakit, antara hasil yang didapat dari sistem untuk menentukan data negatif atau data positif. Sedangkan recall adalah nilai yang dapat menunjukkan tingkat spesifisitas dimana merupakan kemampuan tes untuk menunjukkan individu mana yang tidak menderita sakit dari seluruh populasi yang benar-benar sakit, untuk mengetahui kembali sebuah informasi dengan benar tentang kelas negatif atau pun positif dalam sebuah data (Azhari dkk. 2021). Rumus akurasi, presisi, dan *recall* di tunjukan pada tabel 2.1 berikut:

dan Recall

Tabel Rumus Akurasi, Presisi,

$$Akurasi = \frac{TP+TN}{TP+FP+FN+TN}$$

$$Recall = \frac{TP}{TP+FN}$$

$$Presisi = \frac{TP}{TP+FP}$$

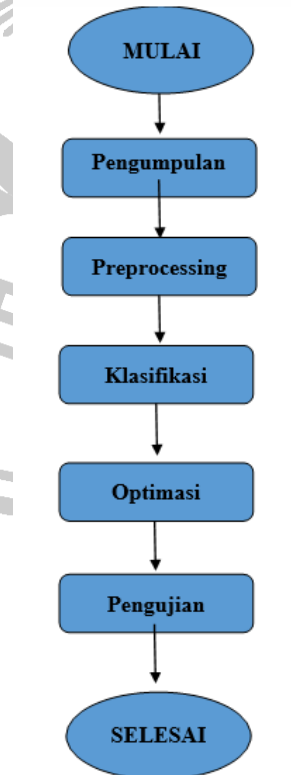
Keterangan  
 TP = True Positif  
 FP = False Positif  
 FN = False Negatif  
 TN = True Negatif

		Prediksi	
		Ya	Tidak
Aktual	Ya	TP	FN
	Tidak	FP	TN

#### 4. PENGUJIAN

##### 1. RANCANGAN PENELITIAN

Rancangan penelitian dalam penelitian ini yaitu pengumpulan data yang diambil dari website [www.kaggle.com](http://www.kaggle.com), klasifikasi menggunakan metode KNN, optimasi menggunakan teknik *Bagging* dan dilakukan tahap pengujian. Berikut gambar metode penelitian ditunjukkan pada gambar di bawah:



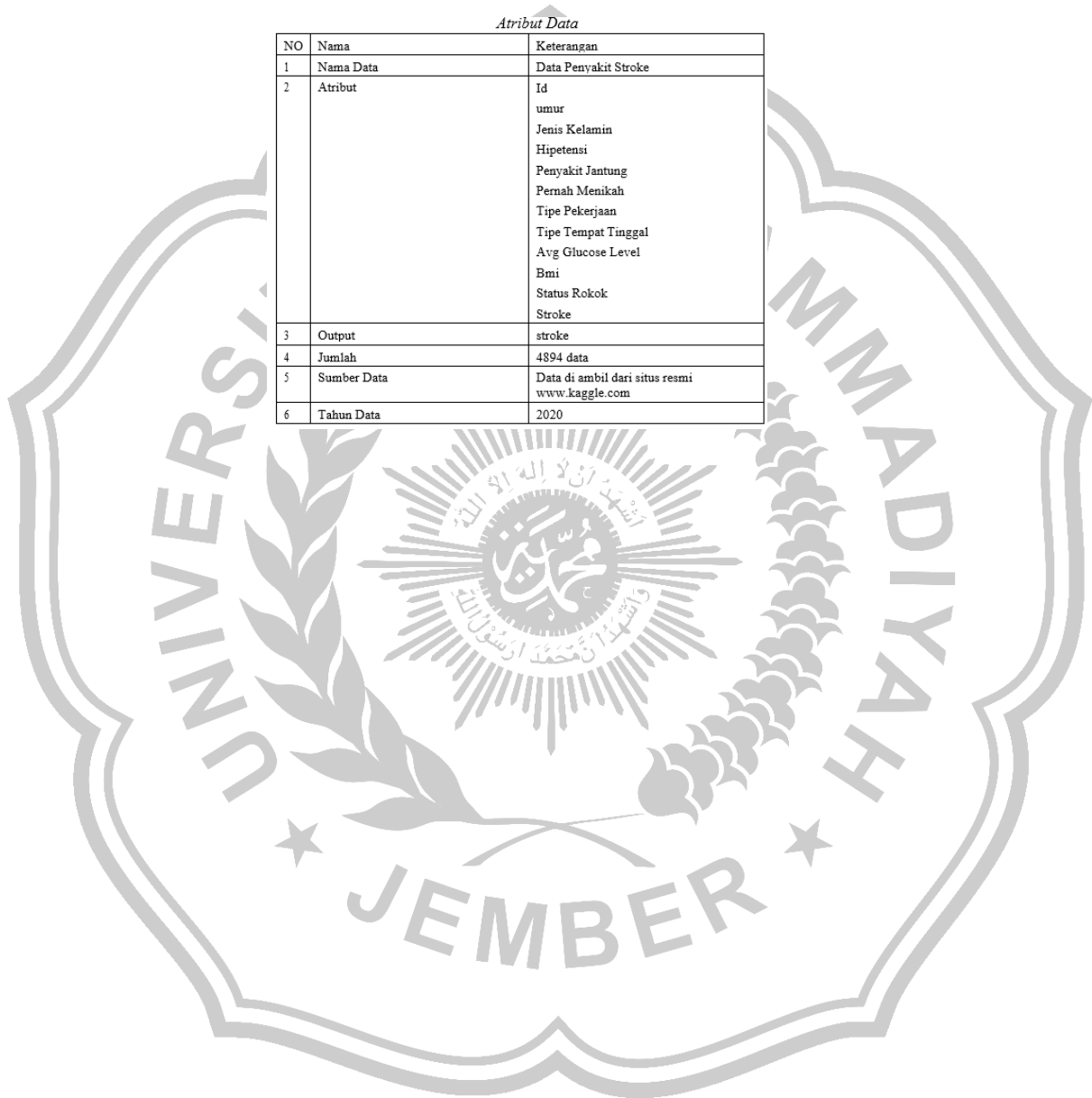
Gambar Alur Metode Penelitian

## 2. PENGUMPULAN DATA

Data pada penelitian penyakit Stroke ini di dapat langsung dari situs resmi [www.kaggle.com](http://www.kaggle.com) yang di akses di akses pada tanggal 2 agustus 2021, Tabel berikut merupakan data atribut data asli yang akan di proses mulai dari tahap *preprocessing*, Pembagian data uji dan data latih, implementasi metode KNN

Atribut Data

NO	Nama	Keterangan
1	Nama Data	Data Penyakit Stroke
2	Atribut	Id umur Jenis Kelamin Hipetensi Penyakit Jantung Pernah Menikah Tipe Pekerjaan Tipe Tempat Tinggal Avg Glucose Level Bmi Status Rokok Stroke
3	Output	stroke
4	Jumlah	4894 data
5	Sumber Data	Data di ambil dari situs resmi <a href="http://www.kaggle.com">www.kaggle.com</a>
6	Tahun Data	2020



### 3. PREPROCESSING

Setelah melewati tahap pengumpulan data kemudian berikutnya akan dilakukan pengolahan data terlebih dahulu. Berikut adalah proses yang akan dilakukan:

No	Gender	Age	Hyper tension	H_D	Married	Work	Residence	A_G_L	Bmi	Smoking	Stroke
1	Male	80	0	1	Yes	Private	Urban	68.53	24.2	smokes never smoked	1
2	Female	77	0	0	Yes	Private	Urban	105.22	31	never smoked	1
3	Female	74	0	0	Yes	Self-employed	Urban	74.96	26.6	never smoked	1
4	Female	20	0	0	No	Private	Rural	96.57	34.1	never smoked	0
5	Female	68	1	0	No	Self-employed	Urban	79.79	29.7	never smoked	1
6	Female	78	0	0	No	Self-employed	Urban	137.74	34.9	formerly smoked	0
7	Female	78	0	1	Yes	Govt_job	Urban	70.21	24.8	never smoked	0
8	Male	71	1	1	No	Govt_job	Rural	216.94	30.9	never smoked	1
9	Male	12	0	0	No	children	Urban	86.86	25.4	never smoked	0
10	Female	24	0	0	Yes	Private	Urban	75.23	29	never smoked	0

Tabel Data Sebelum *Preprocessing*

No	Gender	Age	Hyper tension	H_D	Married	Work	Residence	A_G_L	Bmi	Smoking	Stroke
1	0	80	0	1	1	0	0	68.53	24.2	2	1
2	1	77	0	0	1	0	0	105.22	31	0	1
3	1	74	0	0	1	1	0	74.96	26.6	0	1
4	1	20	0	0	0	0	1	96.57	34.1	0	0
5	1	68	1	0	0	1	0	79.79	29.7	0	1
6	1	78	0	0	0	1	0	137.74	34.9	1	0
7	1	78	0	1	0	2	0	70.21	24.8	0	0
8	0	71	1	1	0	2	1	216.94	30.9	0	1
9	0	12	0	0	0	3	0	86.86	25.4	0	0
10	1	24	0	0	1	1	0	75.23	29	0	0

Tabel Contoh Data Sesudah *Preprocessing*

Data pada tabel 4.1 dan 4.2 adalah data hasil dari reprosesing data dimana pada atribut gender 0 merupakan jenis kelamin male/laki laki sedangkan pada angka 1 merupakan jenis kelamin perempuan, hyper tension angka 0="Tidak" angka 1="ya", H\_D angka 0="Tidak" angka 1="ya", married angka 0="Tidak" 1="Ya", residence angka 0="Tidak" angka 1="Ya" dan stroke dimana angka 0="Tidak" dan 1 berarti "Ya" sedangkan pada atribut work 0= private, 1 = Self- employed, 2 = Govt\_job, 3 = children.

### 4. DATA UJI DAN DATA LATIH

Gambar 4.4 pembagian data uji dan data latih

```
print(X.shape, X_train.shape, X_test.shape)
```

```
(3424, 11) (2396, 11) (1028, 11)
```

Dari gambar di atas adalah pembagian antara data latih dan data uji, dimana data latih di ambil sebanyak 70% dan untuk data uji 30%, dari data 3424, data latihnya sebanyak 2396 dan untuk data uji sebanyak 1028. Hasil dari proses klasifikasi ini juga dilakukan penghapusan beberapa data yang tidak lengkap atau bisa di bilang normalisasi value dimana dari data sebanyak 4894 data ada beberapa yang di hapus, kemudian di temukan sebanyak 1470 data, sehingga data yang di dapat sebanyak 3424 data.

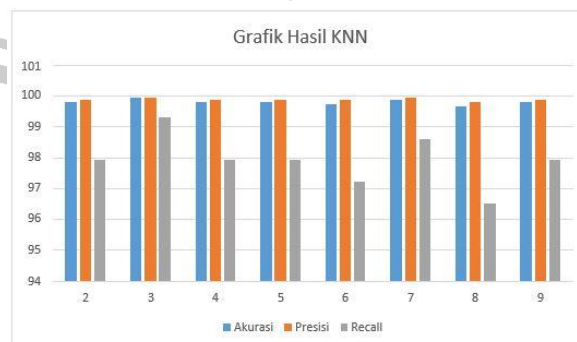
## 5. KLASIFIKASI

Pada langkah ini peneliti menggunakan rancangan K-NN yaitu pengukuran vector yang digunakan adalah euclidean distance dengan nilai k sama dengan 2, 3, 4, 6, 5, 7, 8, 9 peneliti pada implementasi ini menggunakan *tols jupiter* hasil pengukuran akurasi, presisi dan recall sebagai berikut:

Tabel data *klasifikasi* Hasil Kinerja KNN

No	Gender	Age	Hyper tension	H_D	Married	Work	Residence	A_G_L	Bmi	Smoking	Stroke
0	1.0	0.377441	0.0	0.0	1.0	0.50	1.0	0.223571	0.116838	0.333333	0.0
1	1.0	0.218750	0.0	0.0	0.0	1.00	0.0	0.125750	0.142039	1.000000	0.0
2	1.0	1.000000	0.0	1.0	1.0	0.25	1.0	0.805373	0.153494	1.000000	0.0
3	0.0	0.731445	1.0	0.0	0.0	0.00	0.0	0.040763	0.250859	0.000000	0.0
4	0.0	0.951172	0.0	0.0	1.0	0.00	1.0	0.115271	0.131730	0.000000	0.0
5	1.0	0.450684	0.0	0.0	1.0	0.00	0.0	0.236497	0.222222	0.000000	0.0
6	0.0	0.560547	0.0	0.0	1.0	0.00	0.0	0.223894	0.350515	1.000000	0.0
7	0.0	0.194336	0.0	0.0	0.0	0.00	1.0	0.126119	0.255441	1.000000	0.0
8	1.0	0.267578	0.0	0.0	0.0	0.75	1.0	0.216416	0.344788	0.666667	0.0
9	1.0	0.731445	0.0	0.0	1.0	0.00	0.0	0.049211	0.197022	1.000000	0.0
10	0.0	0.291992	0.0	0.0	0.0	0.00	1.0	0.669467	0.313860	0.666667	0.0
Dan seterusnya											
3411	1.0	0.121094	0.0	0.0	0.0	0.75	1.0	0.067953	0.038946	1.000000	0.0
3412	1.0	0.816895	1.0	0.0	1.0	0.25	1.0	0.031484	0.171821	0.666667	1.0
3413	1.0	0.707031	0.0	0.0	1.0	0.25	0.0	0.053504	0.474227	0.000000	0.0
3414	0.0	0.255371	0.0	0.0	0.0	0.00	1.0	0.216647	0.224513	0.000000	0.0
3415	1.0	0.755859	0.0	0.0	1.0	0.50	1.0	0.198597	0.363116	1.000000	0.0
3416	0.0	0.365234	0.0	0.0	1.0	0.00	0.0	0.192595	0.123711	1.000000	0.0
3417	0.0	0.157715	0.0	0.0	0.0	0.75	0.0	0.017265	0.075601	0.000000	0.0
3418	1.0	0.584961	0.0	0.0	1.0	0.50	1.0	0.923507	0.318442	0.333333	0.0
3419	1.0	0.096680	0.0	0.0	0.0	0.75	0.0	0.047087	0.092784	1.000000	0.0
3420	1.0	0.060059	0.0	0.0	0.0	0.75	0.0	0.167021	0.119129	1.000000	0.0
3421	1.0	0.633789	0.0	0.0	1.0	0.50	1.0	0.034807	0.316151	1.000000	0.0
3422	1.0	0.853516	0.0	0.0	0.0	0.25	0.0	0.199889	0.298969	0.333333	0.0
3423	0.0	0.816895	0.0	0.0	1.0	0.50	1.0	0.873234	0.218786	0.333333	0.0
3424	1.0	0.243164	0.0	0.0	0.0	0.00	0.0	0.096113	0.206186	0.000000	0.0

nk	Akurasi	Presisi	Recall
2	99.80%	99.89%	97.92%
3	99.93%	99.96%	99.31%
4	99.80%	99.89%	97.92%
5	99.80%	99.89%	97.92%
6	99.73%	99.86%	97.22%
7	99.86%	99.93%	98.61%
8	99.66%	99.82%	96.53%
9	99.80%	99.89%	97.92%



Gambar Hasil Pengujian KNN



Tabel di atas merupakan hasil keseluruhan skenario KNN yang menjelaskan bahwasannya nilai tertinggi terdapat pada k ke 3 di mana pada k tersebut memiliki tingkat akurasi sebesar 99.93%, dan untuk nilai presisinya yaitu sebesar 99.96%, dan untuk nilai *recall* nya sebesar 99.31% dan untuk hasil rata rata yang di dapat dari hasil uji Akurasi yaitu sebesar 99% dan untuk Presisi sama juga sebesar 99% dan untuk recall untuk rata rata 97% tertinggi 99% dan terendah 96%



## 6. PENGUJIAN

Gambar 4.5 Hasil Pengujian di tols *KNN* di jupiter

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=101)

knn = KNeighborsClassifier(n_neighbors = 3)
knn.fit(X_train, y_train)
y_pred_KNN = knn.predict(X_test)

tes = classification_report(y_test, y_pred_KNN, output_dict=True)

print()
print('Hasil KNN ')
print()
print('accuracy \t:', "{:.2f}".format((accuracy_score(y_test, y_pred_KNN))*100), '%')
print('presisi \t:', "{:.2f}".format((tes['macro avg']['precision'])*100), '%')
print('Recall \t\t:', "{:.2f}".format((tes['macro avg']['recall'])*100), '%')
print()

k_range = range(1, 26)
scores = []
for k in k_range:
    knn = KNeighborsClassifier(n_neighbors = k)
    knn.fit(X_train, y_train)
    y_pred = knn.predict(X_test)
    scores.append(accuracy_score(y_test, y_pred))
```

### Hasil KNN

accuracy	: 99.93 %
presisi	: 99.96 %
Recall	: 99.31 %

Dari gambar di atas merupakan pengujian di tols jupiter dimana hasil dari pengujiannya nilai akurasi presisi dan recall di *KNN* tertinggi terdapat pada *K*-3 hasil keseluruhan skenario *KNN* yang menjelaskan bahwasannya nilai tertinggi terdapat pada *k* ke 3 di mana pada *k* tersebut memiliki tingkat akurasi sebesar 99.93%, dan untuk nilai presisinya yaitu sebesar 99.96%, dan untuk nilai *recall* nya sebesar 99.31%.

## 6. OPTIMASI

Dari data training sebanyak 4894 kemudian dilakukan optimasi dengan teknik bagging yang di mulai melalui tahap bootstrap resample atau data di acak lalu dari data acak baru tersebut dilakukan klasifikasi dengan menggunakan tree sehingga muncul muncul hasil prediksi pada masing-masing data yang yang telah di acak tadi lalu pada tahap terakhir kemudian dilakukan tahap voting dengan mencari nilai terbanyak yang akan dijadikan hasil prediksi akhir pada teknik bagging.

```

from sklearn.ensemble import BaggingClassifier
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=101)

knn = KNeighborsClassifier(n_neighbors = 5)
knn.fit(X_train, y_train)

bagging = BaggingClassifier(base_estimator=knn, n_estimators=10, random_state=0).fit(X_train, y_train)
y_pred_baging = bagging.predict(X_test)

print('accuracy \t:', "{:.2f}".format((accuracy_score(y_test, y_pred_baging))*100), '%')
print('precision \t:', "{:.2f}".format((precision_score(y_test, y_pred_baging))*100), '%')
print('recall \t\t:', "{:.2f}".format((recall_score(y_test, y_pred_baging))*100), '%')

```

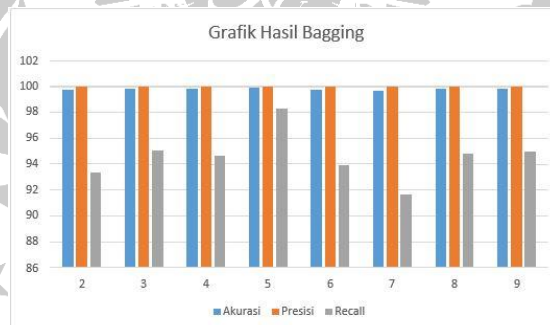
```

accuracy      : 99.93 %
precision     : 100.00 %
recall       : 98.61 %

```

nK	Akurasi	Presisi	Recall
2	99,73%	100.00%	93.33%
3	99.80%	100.00%	95.00%
4	99.80%	100.00%	94.64%
5	99.93%	100.00%	98.25%
6	99.73%	100.00%	93.94%
7	99.66%	100.00%	91.67%
8	99.80%	100.00%	94.83%
9	99.80 %	100.00%	95.00 %

Tabel Hasil Setelah Pengoptimasian *Bagging*



Tabel Hasil Pengujian *Bagging*

Tabel di atas merupakan hasil data dari klasifikasi KNN setelah dilakukan optimasi bagging dimana terjadi perubahan nilai sebelum dan sesudah, dapat dilihat pada tabel di atas bahwasanya nilai tertinggi terdapat pada k ke 5 di mana pada k tersebut memiliki tingkat akurasi sebesar 99.93%, dan untuk nilai presisinya yaitu sebesar 100.00%, dan untuk nilai *recall* nya sebesar 98.61% dan untuk rata-rata hasil nilai pengujian di mana di dapat untuk Akurasi rata-rata 99% dan presisi sebesar 100.00%, dan untuk nilai *recall* nya sebesar 98.61% dan untuk rata-rata hasil nilai pengujian di mana di dapat untuk Akurasi rata-rata 99% dan presisi

Berdasarkan hasil dari penelitian yang telah di lakukan dapat di simpulkan bahwa:

## 5. KESIMPULAN DAN SARAN

### 5.1 KESIMPULAN

Berdasarkan hasil dari penelitian yang telah di lakukan dapat di simpulkan bahwa:

1. Dari hasil uji nilai akurasi, presisi dan recall menyatakan bahwasannya hasil nilai tertinggi untuk Algoritma *K-Nearest Neighbor* terdapat pada K-3 dengan nilai akurasi sebesar 99.93%, presisi sebesar 99.96% dan recall sebesar 99.31%. maka dapat di simpulkan nilai yang di dapat sangat tinggi.
2. Untuk Nilai akurasi, presisi dan recall setelah di lakukan proses optimasi *bagging* terdapat pada K-5 dengan nilai akurasi sebesar 99.93%, presisi sebesar 100.00% dan recall sebesar 98.61%. begitupun dalam proses optimasi di dapat nilai tinggi pada K-5.
3. Nilai ,presisi dan recall mengalami perubahan dimana nilai prsisi semakin tinggi dan nilai recall mengalami penurunan.

### 5.2 SARAN

Berdasarkan hasil penelitian yang telah di lakukan di perlukan saran yang dapat untuk di kembangkan untuk penelitian selanjutnya sebagai berikut:

1. Metode yang telah di buat oleh penulis tidak terlalu berpengaruh terhadap hasil nilai yang di dapat karna hasil dari KNN saja sudah cukup baik walaupun tanpa di lakukan optimasi *bagging*
2. Karna hasil dari penelitian ini dapat di simpulkan hasil Akurasi presisi dan recall sudah sangat tinggi maka saran yang bisa di berikan yaitu mencoba menggunakan data lain serta menggunakan optimasi yang lain juga.

## 6. DAFTAR PUSTAKA

- Adelina V, Ratnawati, D.E, dan Fauzi , M.A. 2018. “Klasifikasi Tingkat Risiko Penyakit Stroke Menggunakan Metode *GA-Fuzzy Tsukamoto*”.
- Azhari,M, Situmorang ,Z, dan Rosnelly,R ,2021. “Perbandingan Akurasi, Recall, dan Presisi Klasifikasi pada Algoritma C4.5, Random Forest, SVM dan Naive Bayes” jurnal Media Informatika Budidarma Volume 5, Nomor 2, April 2021 ISSN 2614-5278 Universitas Katolik Santo Thomas,Medan.

- Fadlilah, M,S, Wihandika, R,C, dan Rahayudi,B,2019.” Klasifikasi Penurunan Fungsi Kognitif Pasien Stroke Menggunakan Metode Klasifikasi *Random Forest*” Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer Vol. 3, No. 3, Maret 2019, e-ISSN: 2548-964X Vol. 3, No. 3, Maret 2019, Universitas Brawijaya,Malang.
- Utomo D,P,2020 “Analisis Komparasi Metode Klasifikasi Data Mining danReduksi Atribut Pada Data Set Penyakit Jantung” Jurnal media informatikabudi darma Volume 4, Nomor 2, April 2020, Page 437-444 ISSN 2614-5278 Program Studi Teknik Informatika, STMIK Budi Darma, Medan,Indonseisa.
- Utama R..S, Hidayat, N, dan Santoso,E.2018. “Sistem Pakar Deteksi Dini Penyakit Stroke Menggunakan Metode *Naïve Bayes-Certainty Factor*” Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer Vol. 2, No.9, September 2018, e-ISSN: 2548-964X. Universitas Brawijaya, Malang.
- Koko ,H, 2016 “Penerapan Data Mining DalamMeningkatkan Mutu Pembelajaran PadaInstasi Perguruan Tinggi MenggunakanMetode *K-Means Clustering* (Studi Kasus DiProgram Studi TKJ Akademi KomunitasSolok Selatan)” Teknosi, Vol. 02, No. 03, Desember 2016. Universitas Putera Batam
- Lishania, I, Goejantoro, R, dan Novia Nasution.Y. 2019. “Perbandingan Klasifikasi Metode *Naive Bayes* dan Metode *Decision Tree* Algoritma (J48)pada Pasien Penderita Penyakit *Stroke* di RSUD Abdul Wahab Sjahranie Samarinda”. *Jurnal EKSPONENSIAL Volume 10, Nomor 2*. Universitas Mulawarman, Samarinda.
- Panjaitan I,L,K , Panggabean ,E, dan Sulindawaty ,2018. “Analisis PerbandinganMetode Dempster Shafer Dengan Metode *Certainty Factor* Untuk Mendiagnosa Penyakit *Stroke*”*Journal Of Informatic Pelita Nusantara* Volume 3 No. 1 Maret 2018 e-ISSN 2541-3724, STMIK Pelita Nusantara,Medan.
- Puspitawuri, A, Santoso,E, dan Dewi ,C,2019.“Diagnosis Tingkat Risiko Penyakit *Stroke* Menggunakan Metode *K-Nearest Neighbor* dan *Naïve Bayes*”Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer Vol. 3, No. 4, April 2019, e-ISSN: 2548-964X Universitas Brawijaya,Malang.
- Rahmawati E, Agustina C,2020.“Implementasi Teknik *Bagging* untuk Peningkatan Kinerja J48 dan *Logistic Regression* dalam Prediksi Minat Pembelian Online”Jurnal Teknologi Informasi dan Terapan (J-TIT) Vol. 7 No. 1 Januari-Juni 2020 ISSN: 2580-2291, Universitas Bina Sarana Informatika,Jakarta.
- Rohman ,R,S, Saputra ,R,A, Firmansaha, D,A, 2020. “Komparasi Algoritma C4.5Berbasi PSO Dan GA Untuk Diagnosa Penyakit *Stroke*” *Journal of Computer Engineering System and Science* , Vol. 5 No. 1 Januari 2020 p-ISSN :2502-7131, Fakultas Teknologi Informasi, Universitas Bina Sarana Informatika Jl. Cemerlang No. 8 Sukakarya Sukabumi, Jawa Barat
- Sulastri, Had iono,K, dan M,T, dan Anwar, 2019. “Analisis Perbandingan Klasifikasi Prediksi Penyakit Hepatitis dengan Menggunakan Algoritma *K-Nearest Neighbor*,*Naive Bayes* dan *Neural Network* Jurnal DINAMIK Volume 24, No. 2, Juli 2019 : 82-91 E-ISSN : 2623-1786 | P-ISSN : 0854- 9524, Universitas Stikubank,Semarang.Vol. 2, No. 11, November 2018, e-ISSN: 2548-964X Universitas Brawijaya,Malang.

- Widodo A,M, dkk 2021 “Komparasi Performansi Algoritma Pengklasifikasi *KNN*, *Bagging* Dan *Random Forest* Untuk Prediksi Kanker Payudara” Konferensi Nasional Ilmu Komputer (KONIK) 2021, P-ISSN : 2338-2899 Fakultas Ilmu Komputer, Universitas Unggul Esa Unggul, Jakarta Barat, 11510, Indonesia dll.
- Syahid,D, dkk 2016 “Sistem Klasifikasi Jenis Tanaman Hias Daun Philodendron Menggunakan Metode K-Nearest Neighbor (KNN) Berdasarkan Nilai Hue,Saturation,Value” JOIN | Volume I No. 1 | Juni 2016, ISSN 2527-9165 Fakultas



