

Perbandingan Algoritma *Extreme Learning Machine* dan *Multilayer Perceptron* Dalam Prediksi Mahasiswa *Drop Out*

Muhammad Ibnu Sa'ad

Program Studi Teknik Informatika, STMIK Widya Cipta Dharma, Samarinda, Indonesia

Email: saad@wicida.ac.id

Abstrak *Drop out* atau pemberhentian status mahasiswa adalah proses pencabutan status kemahasiswaan atas diri mahasiswa, yang disebabkan oleh hal -hal tertentu yang telah di tentukan oleh universitas yang bersangkutan. Tingginya jumlah mahasiswa *drop out* pada perguruan tinggi dapat diminimalisir dengan kebijakan dari perguruan tinggi untuk mengarahkan dan mencegah mahasiswa dari *drop out* bahwa mendeteksi mahasiswa berisiko pada tahap awal pendidikan sangat penting dilakukan untuk menjaga mahasiswa dari *drop out*.

Tujuan dari penelitian ini adalah untuk klasifikasi dan membandingkan algoritma *Extreme Learning Machine* dan *Multilayer Perceptron* dalam memprediksi Mahasiswa *drop out*. Penelitian ini menggunakan dua algoritma yaitu *Extreme Learning Machine* dan *Multilayer Perceptron* yang merupakan metode pembelajaran jaringan saraf tiruan *feedforward*.

Data yang digunakan sebanyak 110 data sesuai dengan jumlah mahasiswa dari angkatan 2012 s.d 2018. Data diambil dari sistem informasi akademik Doktor Manajemen Pendidikan. Dalam kasus ini bagaimana memprediksi mahasiswa *drop out* menggunakan variabel Jenis Kelamin, Status Bekerja, Status Berkeluarga, Usia, IP Semestrer 3, Ujian Komprehensif, Progres Disertasi, dan Publikasi. Hasil klasifikasi *Extreme Learning Machine* berdasarkan rasio 80:20 mendapatkan akurasi 95% dengan *hidden layer* 20 dan nilai *Mean Squared Error* sebesar 0.369. Sedangkan pada *Multilater Perceptron* dengan rasio yang sama mendapatkan hasil akurasi 91%. Dari kedua model yang digunakan menunjukkan bahwa kedua algoritma jaringan saraf tiruan ini dapat menghasilkan kinerja yang baik dalam memprediksi mahasiswa *drop out*.

Kata Kunci: *drop out, algoritma, extreme learning machine, multilayer perceptron*

Abstract - Determined by the university concerned. The high number of drop out students at tertiary institutions can be minimized by policies from tertiary institutions to direct and prevent students from dropping out that detecting at-risk students in the early stages of education is very important to do to keep students from dropping out.

The purpose of this study is to classify and compare the Extreme Learning Machine and Multilater Perceptron algorithms in predicting student drop out. This study uses two algorithms, namely Extreme Learning Machine and Multilater Perceptron which are feedforward artificial neural network learning methods.

The data used is 110 data according to the number of students from class 2012 to 2018. The data is taken from the Doctor of Education Management academic information system. In this case how to predict student drop out using the variables Gender, Working Status, Family Status, Age, Semester 3 GPA, Comprehensive Examination, Dissertation Progress, and Publications. The results of the Extreme Learning Machine classification based on a ratio of 80:20 get an accuracy of 95% with a hidden layer of 20 and a Mean Squared Error value of 0.369. Whereas the Multilater Perceptron with the same ratio gets 91% accuracy. From the two models used, it shows that the two artificial neural network algorithms can produce good performance in predicting drop out students.

Keywords: *drop out, algoritma, extreme learning machine, multilayer perceptron*

1. PENDAHULUAN

Prediksi *drop out* dapat dilakukan dengan serangkaian proses mendapatkan pengetahuan atau pola dari kumpulan data yang di sebut data mining [1],[2]. Data mining memecahkan masalah dengan menganalisis data yang telah ada dalam database. [3],[4] Beberapa algoritma klasifikasi data mining telah digunakan untuk memprediksi perilaku mahasiswa yang berpotensi *drop out* diantaranya decision tree, neural network, naïve bayes, instance-based learning, logistic regression, support vector machine, K-Nearest Neighbor, Rapidminer, dan Extreme Learning Machine. Prediksi adalah proses memperkirakan nilai atau kejadian di masa depan berdasarkan data yang tersedia dari masa lalu atau saat ini. Prediksi dapat dilakukan dengan menggunakan berbagai metode statistik dan pembelajaran mesin, seperti regresi linier, jaringan syaraf tiruan, dan decision tree. Prediksi merupakan usaha menduga atau memperkirakan sesuatu yang akan terjadi di waktu mendatang dengan memanfaatkan berbagai informasi yang relevan pada waktu-waktu sebelumnya (historis) melalui suatu metode ilmiah. [5].

Klasifikasi merupakan salah satu jenis tugas dalam pembelajaran mesin yang bertujuan untuk memprediksi label atau kelas pada suatu data yang belum diberi label berdasarkan informasi atau fitur yang tersedia. Dalam klasifikasi, model pembelajaran mesin dibangun dengan menggunakan data yang telah diberi label atau kelas untuk mempelajari pola atau aturan yang dapat digunakan untuk memprediksi kelas dari data yang belum diketahui. Salah satu tugas yang dapat dilakukan dengan data mining adalah pengklasifikasian. [6],[4].

Data mining merupakan Penemuan Pengetahuan dalam Basis Data (KDD) [7], adalah bidang penemuan informasi baru dan berpotensi berguna dari sejumlah besar data. Data mining telah diterapkan di banyak bidang, termasuk penjualan eceran, bioinformatika, dan kontra-terorisme. Dalam beberapa tahun terakhir, telah terjadi peningkatan minat dalam penggunaan data mining untuk menyelidiki pertanyaan ilmiah dalam penelitian pendidikan, bidang penyelidikan disebut data pendidikan pertambangan. Educational Data Mining (juga disebut sebagai "EDM") didefinisikan sebagai bidang

ilmiah penyelidikan berpusat di sekitar pengembangan metode untuk membuat penemuan dalam unik jenis data yang berasal dari pengaturan pendidikan, dan menggunakan metode tersebut untuk lebih memahami siswa dan latar tempat mereka belajar. [2][8]

Metode yang digunakan untuk memprediksi pada penelitian yang akan dilaksanakan yaitu menggunakan *Extreme Learning Machine* (ELM) [9]. [10]*Extreme Learning Machine* merupakan metode pembelajaran baru dari jaringan syaraf tiruan *feedforward* dengan single *hidden layer feedforward neural networks* (SLFNs). Jaringan syaraf tiruan (artificial neural network) adalah model matematika yang terinspirasi dari cara kerja sistem saraf manusia. Seperti sistem saraf manusia, jaringan syaraf tiruan terdiri dari banyak unit pemrosesan (neuron) yang terhubung dengan cara tertentu untuk memproses informasi. Suatu jaringan saraf tiruan memproses sejumlah besar informasi secara paralel dan terdistribusi, hal ini terinspirasi oleh model kerja otak biologis. [11] Pada jaringan syaraf [12], neuron-neuron akan dikumpulkan dalam lapisan-lapisan yang disebut dengan lapisan neuron. Biasanya neuron pada satu lapisan akan dihubungkan dengan lapisan sebelum atau sesudahnya terkecuali lapisan masukkan dan lapisan keluaran. Informasi yang diberikan pada jaringan syaraf akan dirambatkan dari lapisan ke lapisan, melalui dari lapisan masukkan sampai lapisan keluaran melalui lapisan tersembunyi. [13]. Sebagai algoritma pembelajaran baru, ELM memiliki waktu komputasi yang rendah persyaratan untuk melatih pengklasifikasi baru sejak bobot dan bias dari lapisan tersembunyi ditetapkan secara acak dan hasilnya bobot secara analitis ditentukan oleh matematika sederhana manipulasi [14].

Perbandingan metode lainnya untuk mengukur akurasi adalah MLP Multilayer Perceptron yang merupakan jenis jaringan saraf buatan (artificial neural network) yang paling banyak digunakan dalam pembelajaran mesin. MLP terdiri dari beberapa lapisan neuron, yaitu lapisan masukan (input layer), lapisan tersembunyi (hidden layer), dan lapisan keluaran (output layer) [15]. Multilayer Perceptron MLP adalah proses menyesuaikan koneksi bobot untuk mendapatkan perbedaan minimal antara jaringan output dan output yang diinginkan [16].

Solusi yang ditawarkan pada penelitian adalah untuk melihat akurasi terbaik dengan *hidden layer* terkecil dari kedua algoritma yang diuji, sehingga algoritma dengan akurasi terbaik bisa digunakan oleh program studi untuk memprediksi mahasiswa yang terindikasi *drop out* di masa yang akan datang.

2. METODOLOGI PENELITIAN

2.1 Metode Pengumpulan Data

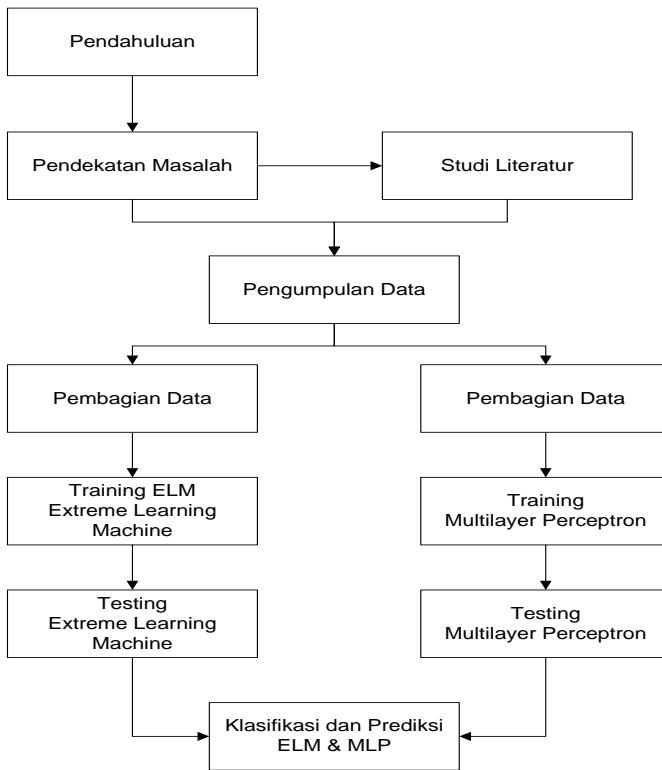
Data yang diperoleh dalam penelitian ini merupakan data mahasiswa baru yang didapatkan secara langsung dari sumber data bagian Akademik Prodi Manajemen Pendidikan Program Doktor FKIP Universitas Mulawarman, kemudian digunakan pula studi pustaka yang merupakan data sekunder baik itu berupa buku data mining, ebook, dan jurnal mengenai prediksi mahasiswa *drop out* [17]. Data yang digunakan adalah data mahasiswa baru tahun akademik 2012 sampai dengan 2018 Program Studi Doktor Manajemen Pendidikan FKIP Universitas Mulawarman.

2.2 Metode Analisa Data

Pada metode analisis data *Extreme Learning Machine* digunakan *Confusion Matrix* [18]. pada metode klasifikasi akan dilakukan evaluasi terutama pada bagian akurasi dari hasil klasifikasi [19]. Akurasi sebuah klasifikasi berpengaruh terhadap performa dari suatu klasifikasi. Untuk melakukan analisa dapat digunakan *Confusion Matrix* yaitu sebuah matrik dari prediksi yang akan dibandingkan dengan kelas yang asli dari data inputan. Sebagai contoh, sebuah test (i,j) dari sebuah *Confusion Matrix* adalah persentase dari waktu dari sebuah classifier yang melakukan identifikasi inputI sebagai pattern dari kelas j. Setiap kolom dari matriks berkorespondensi kepada classifier output dan setiap baris pada input. Akurasi sebuah klasifikasi dimana i=j menerangkan akurasi dari klasifikasi pada setiap kelas. Untuk menunjukkan kecepatan proses pembelajaran dan keakuratan Extreme Learning Machine maka kinerja Extreme Learning Machine akan dibandingkan dengan Algoritma Multilayer Perceptron (MLP) pada data yang sama yaitu dari Manajemen Pendidikan Program Doktor FKIP Universitas Mulawarman.

2.3 Tahapan Penelitian

Langkah-langkah pembuatan model prediksi mahasiswa *drop out* dengan menggunakan metode *Extreme Learning Machine* dan *Multilayer Perceptron* adalah: pengumpulan data, implementasi metode *Extreme Learning Machine* dan *Multilayer Perceptron* untuk prediksi melalui pembagian data, *training* dan *testing*, serta analisis hasil prediksi. Pada tahap awal penelitian dimulai dengan pendahuluan dan pendekatan masalah. Studi literatur dilakukan untuk memperdalam pemahaman mengenai cara kerja metode *Extreme Learning Machine* dan *Multilayer Perceptron* serta tahapan-tahapan apa saja yang diperlukan untuk prediksi dengan menggunakan metode *Extreme Learning Machine* dan *Multilayer Perceptron*. Selain itu, studi literatur juga dilakukan untuk mengetahui mahasiswa *drop out* agar pengumpulan data memiliki fokus yang lebih baik. Tahap ke dua dari penelitian ini adalah pengumpulan data. Tahap ke tiga adalah implementasi metode *Extreme Learning Machine* melalui pembagian data, *training* dan *testing*.[20]

**Gambar 1.** Tahapan Penelitian [20]

3. HASIL DAN PEMBAHASAN

Pada tahapan ini dilakukan proses implementasi dengan memanfaatkan data dari sistem informasi akademik doktor manajemen pendidikan. Pengujian menggunakan dua algoritma yaitu *Extreme Learning Machine* dan *Multilayer Perceptron*.

3.1 Implementasi

a. Skenario Data *Training* dan Data *Testing* *Extreme Learning Machine*

Pada tahapan ini mencari model yang paling akurat, dengan menggunakan data mahasiswa yang sudah berlalu. untuk mengukur akurasi *Extreme Learning Machine* menggunakan fungsi aktivasi Sigmoid dan fungsi Bipolar (*Symmetric Hard Limit*) dilakukan uji coba skenario data training dan testing. Setelah memprediksi keseluruhan data, kemudian dilakukan kombinasi variabel untuk menentukan akurasi terbaik dari variabel yang telah dikombinasikan.

b. Analisis *Extreme Learning Machine*

Proses uji coba dilakukan untuk menentukan keakuratan sistem dalam melakukan proses prediksi mahasiswa drop out. Dalam subbab ini menggunakan variabel Class DO dan NON DO dan menggunakan fungsi aktivasi Sigmoid. Pada Tabel 1 merupakan pembagian data untuk proses training dan testing.

Tabel 1. Skenario Data *Training* dan *Testing*

Skenario	Rasio	Data	Data <i>Testing</i>	Total
<i>Training</i>				
1	80:20	88	22	100%
2	70:30	77	33	100%

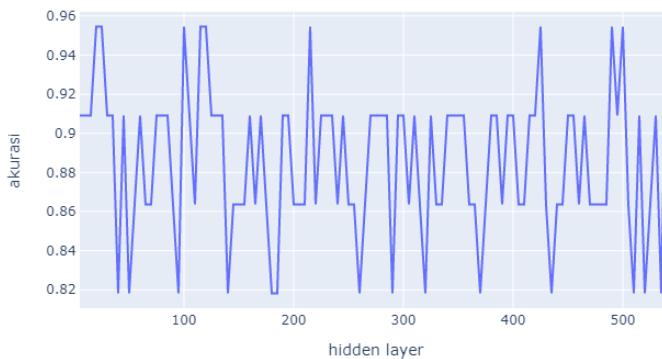
Data mahasiswa yang digunakan sebanyak 110 data. Untuk skenario 1 digunakan rasio 80:20 dengan data *training* 88 dan data *testing* 22, skenario 2 digunakan rasio 70:30 dengan data *training* 77 dan data *testing* 33 sehingga total keseluruhan masing-masing rasio adalah 110 data atau 100%.

Berikut ini merupakan hasil dari akurasi pada setiap skenario yang digunakan, dapat dilihat pada Tabel 2.

Tabel 2. Uji Coba dan Hasil Skenario

Skenario	Rasio	Akurasi	Hidden Layer	MSE
1	80:20	0.954	35	0.369
2	70:30	0.939	25	0.389

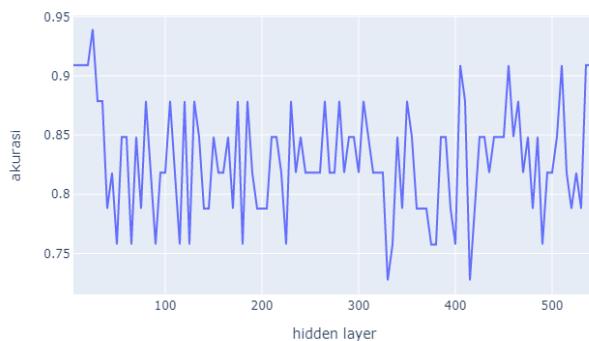
Akurasi dan Hidden Layer



Gambar 2. Grafik Akurasi dan *Hidden Layer* Rasio 80:20

Berdasarkan Tabel 2 dan Gambar 1, jika dilihat pada rasio 80:20 memiliki akurasi yang tinggi yaitu **0.954%** dengan *hidden layer* **35** dan nilai *Mean Squared Error* sebesar **0.369**. Pada Grafik dapat dilihat semakin banyak jumlah *hidden neuron* maka tidak semakin baik akurasi yang didapatkan karena semakin banyak proses yang dilakukan bisa menjadikan proses *Extreme Learning Machine* tidak optimal.

Akurasi dan Hidden Layer



Gambar 3. Grafik Akurasi dan *Hidden Layer* Rasio 70:30

Berdasarkan Tabel 2 dan Gambar 3, jika dilihat pada rasio 70:30 memiliki akurasi sebesar **0.939%** dengan *hidden layer* **25** dan nilai *Mean Squared Error* sebesar **0.389**. Pada grafik dapat dilihat semakin banyak jumlah *hidden neuron* maka tidak semakin baik akurasi yang didapatkan karena semakin banyak proses yang dilakukan bisa menjadikan proses *Extreme Learning Machine* tidak optimal, pada rasio 70:30 tidak cukup optimal dibanding rasio 80:20, karena *hidden layer* yang banyak digunakan untuk mendapatkan hasil akurasi yang tinggi dan nilai *error* yang didapatkan juga cukup besar dibanding rasio 80:20. Didalam pengujian ini, bila membandingkan nilai persentase akurasi dari kedua model yang digunakan, rasio 80:20 *Extreme Learning Machine* mempunyai persentase akurasi lebih besar dan nilai *error* yang kecil dibandingkan dengan rasio 70:30 serta *hidden layer* yang digunakan hanya sebanyak 20. Maka bisa disimpulkan pada pengujian ini, rasio 80:20 yang lebih akurat dalam memprediksi mahasiswa *drop out*. Dan pada jurnal Dong Xiao dikatakan bahwa jumlah *hidden layer* yang paling baik didapatkan dalam *range* 20-60.

c. Implementasi *Extreme Learning Machine*

1. Langkah pertama mengimport *library* yang dibutuhkan, dalam studi kasus ini menggunakan *library sklearn* yang telah disiapkan oleh python. Berikut ini kode yang digunakan:

```
import pandas as pd
pd.set_option('display.max_rows', 500)
import numpy as np
from sklearn_extensions.extreme_learning_machines.elm import ELMClassifier as elm
from sklearn.metrics import confusion_matrix
from sklearn.metrics import accuracy_score, confusion_matrix
from sklearn.metrics import mean_squared_error
import plotly.express as px
```

2. Langkah kedua adalah membaca data *input, dataset* yang digunakan adalah data mahasiswa Prodi Manajemen Pendidikan Program Doktor dari angkatan 2012 sampai 2018 yang sebelumnya telah dibuat dalam format excel.

```
data=pd.read_excel('tes tesis.xlsx')
data
```

Output:

	Jenis Kelamin	Status Bekerja	Status Berkeluarga		Usia	IP Semester 3	Ujian Komprehensif	Progres Disertasi	Publikasi
0	Perempuan	Ya	Ya	Tua	3.43		Tidak	Tidak	Tidak
1	Laki-Laki	Ya	Ya	Menengah	3.75		Ya	Selesai	Sudah
2	Laki-Laki	Ya	Ya	Menengah	3.58		Ya	Tidak	Tidak
3	Laki-Laki	Ya	Ya	Tua	3.40		Ya	Tidak	Tidak
4	Laki-Laki	Ya	Ya	Tua	4.00		Ya	Tidak	Tidak
5	Perempuan	Ya	Ya	Menengah	4.00		Ya	Selesai	Sudah
6	Laki-Laki	Ya	Ya	Tua	3.78		Ya	Selesai	Sudah
7	Laki-Laki	Ya	Ya	Menengah	4.00		Ya	Selesai	Sudah
8	Laki-Laki	Ya	Ya	Tua	3.33		Ya	Selesai	Sudah
9	Laki-Laki	Ya	Ya	Menengah	3.50		Ya	Selesai	Sudah
10	Laki-Laki	Ya	Ya	Menengah	4.00		Ya	Selesai	Sudah

3. Langkah ketiga melakukan preprocessing atau inisialisasi data. Proses ini digunakan untuk mengubah data yang berinisial huruf menjadi angka 0 dan 1. Berikut ini kode yang digunakan:

```
from sklearn import preprocessing
le = preprocessing.LabelEncoder()
data['Jenis Kelamin'] = le.fit_transform(data['Jenis Kelamin'].values)
data['Status Bekerja'] = le.fit_transform(data['Status Bekerja'].values)
data['Status Berkeluarga'] = le.fit_transform(data['Status Berkeluarga'].values)
data['Usia'] = le.fit_transform(data['Usia'].values)
data['Ujian Komprehensif'] = le.fit_transform(data['Ujian Komprehensif'].values)
data['Progres Disertasi'] = le.fit_transform(data['Progres Disertasi'].values)
data['Publikasi'] = le.fit_transform(data['Publikasi'].values)
data['Class'] = le.fit_transform(data['Class'].values)

print(data.info())
data.head()
```



BULLETIN OF INFORMATION TECHNOLOGY (BIT)

Vol. 4 No. 2 September 2022 H. 260 274
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 110 entries, 0 to 109
Data columns (total 9 columns):
 # Column Non-Null Count Dtype

 0 Jenis Kelamin 110 non-null int32
 1 Status Bekerja 110 non-null int32
 2 Status Berkeluarga 110 non-null int32
 3 Usia 110 non-null int32
 4 IP Semester 3 110 non-null float64
 5 Ujian Komprehensif 110 non-null int32
 6 Progres Disertasi 110 non-null int32
 7 Publikasi 110 non-null int32
 8 Class 110 non-null int32
dtypes: float64(1), int32(8)
memory usage: 4.4 KB
None

	Jenis Kelamin	Status Bekerja	Status Berkeluarga	Usia	IP Semester 3	Ujian Komprehensif	Progres Disertasi	Publikasi	Class
0	1	0		1	2	3.43	0	2	2
1	0	0		1	0	3.75	1	1	1
2	0	0		1	0	3.58	1	2	2
3	0	0		1	2	3.40	1	2	0
4	0	0		1	2	4.00	1	2	0

4. Langkah ke empat membagi data *training* dan *testing*, dalam penelitian ini menggunakan prosentase ukuran data *testing* 20% atau 0.2.

```
from sklearn.model_selection import train_test_split  
X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size = 0.2, random_state = 7)  
print ("Data Training:", len(X_train))  
print ("Data Testing:", len(X_test))
```

Output:

Data Training: 88
Data Testing: 22

5. Langkah kelima menguji kehandalan dari model Extreme Learning Machine. Range yang digunakan mulai dari 1 sampai 110 sesuai data mahasiswa, kemudian digunakan fungsi aktivasi *sigmoid* dengan *inputhidden layer* sebanyak 5 sehingga model bisa menghasilkan keluaran yang lebih akurat. pada fungsi sigmoid berapapun angka yang digunakan akan menghasilkan keluaran berupa angka 0 hingga 1. Berikut ini kode yang digunakan pada proses pengujian akurasi dan keluaran *hidden layer* pada model *extreme learning machine*:

```
hidden_layer=[]  
hasil=[]  
for i in range(1,110):  
    hiddenlayer=i*5  
    elm_clf = elm(n_hidden=hiddenlayer, activation_func='sigmoid').fit(X_train.values,  
y_train)  
    pred = elm_clf.predict(X_test)  
    hidden_layer.append(hiddenlayer)  
    hasil.append(accuracy_score(pred, y_test))  
data=hidden_layer,hasil  
hasil_akurasi=pd.DataFrame(data=data).transpose()  
hasil_akurasi.columns=['hidden layer','akurasi']  
hasil_akurasi
```

Output:

	hidden layer	akurasi
0	5.0	0.863636
1	10.0	0.909091
2	15.0	0.909091
3	20.0	0.863636
4	25.0	0.909091
5	30.0	0.863636
6	35.0	0.954545
7	40.0	0.727273
8	45.0	0.727273
9	50.0	0.681818
10	55.0	0.681818

Pada percobaan skenario 80:20 mendapatkan hasil akurasi **0.954** dengan jumlah keluaran *hidden layer* sebanyak **35**. Pada prosesnya jika melakukan percobaan beberapa kali, bisa jadi pada jumlah *hidden layer* mendapatkan hasil yang sedikit berbeda karena proses *training* dipilih secara acak pada saat proses pemilahan *training* dataset dan *testing* dataset. Pada beberapa kali dilakukan percobaan pada skenario **80:20** mendapatkan hasil akurasi sebesar **0.954** dan jumlah *hidden layer* pada rentang 20-60 yang artinya baik untuk digunakan dalam prediksi. pada studi kasus ini *hidden layer* yang didapatkan berada di range **20-35**. Dari percobaan ini telihat selisih perbedaan jumlah keluaran *hidden layer* untuk mendapatkan hasil akurasi yang sama.

6. Langkah keenam menghitung nilai *error* semua *output layer* dimana nilai *error* ini adalah nilai kesalahan prediksi. Pada percobaan ini menggunakan *mean squared error*. Berikut ini kode yang digunakan:

```
from sklearn.metrics import mean_squared_error
from math import sqrt
rmse = sqrt(mean_squared_error(y_test, pred))
print(rmse)
```

Output:

0.369

Pada percobaan menggunakan rasio **80:20** mendapatkan nilai *error* sebesar **0.369**. pada skenario ini merupakan jumlah kesalahan prediksi terkecil dibandingkan pada percobaan skenario **70:30** terdapat kesalahan prediksi sebesar **0.389**.

d. Analisis Multilayer Perceptron

Hasil perhitungan evaluasi *Precision*, *Recall*, dan *F-Score* pada data *training* model *Multilayer Perceptron* dapat dilihat pada tabel 3 berikut ini:

Tabel 3. Hasil Evaluasi *Training Precision*, *Recall*, dan *F-Score MLP*

	precision	recall	f1-score	support
0	1.00	0.89	0.94	35
1	0.93	1.00	0.96	53
accuracy		0.95	0.95	88
macro avg	0.96	0.94	0.95	88
weighted avg	0.96	0.95	0.95	88

Pada hasil evaluasi disimpulkan bahwa status mahasiswa *drop out* pada Prodi Doktor Manajemen Pendidikan memiliki nilai *precision* 1.00, *recall* 0.89, *F-Score* 0.94, dan memiliki support *training* sebesar 35%, sedangkan pada status mahasiswa *non drop out* terdapat nilai *precision* 0.93, *recall* 1.00, *F-Score* 0.96, dan memiliki support *testing* sebesar 53%.

Hasil perhitungan evaluasi *Precision*, *Recall*, dan *F-Score* pada data *testing* model *Multi Layer Perceptron* dapat dilihat pada Tabel 4 berikut ini:

Tabel 4. Hasil Evaluasi *Testing Precision*, *Recall*, dan *F-Score MLP*

	precision	recall	f1-score	support
0	0.91	0.91	0.91	11
1	0.91	0.91	0.91	11
accuracy		0.91	0.91	22
macro avg	0.91	0.91	0.91	22
weighted avg	0.91	0.91	0.91	22

Pada hasil evaluasi disimpulkan bahwa status mahasiswa *drop out* pada Prodi Doktor Manajemen Pendidikan memiliki nilai *precision* 0.91, *recall* 0.91, *F-Score* 0.91, dan memiliki support *training* sebesar 11%, sedangkan pada status mahasiswa *non drop out* terdapat nilai *precision* 0.91, *recall* 0.91, *F-Score* 0.91, dan memiliki support *testing* sebesar 11%. Hasil akurasi pada model *Multilayer Perceptron* dalam memprediksi mahasiswa *drop out* pada Prodi Doktor Manajemen Pendidikan adalah sebesar 91%.

4. KESIMPULAN

Dari teknik data mining yang dibangun, maka model yang memiliki tingkat akurasi yang terbaik dalam prediksi mahasiswa *drop out* adalah *Extreme Learning Machine*. berdasarkan skenario dan eksperimen yang dilakukan menggunakan fungsi aktivasi *sigmoid*, model *Extreme Learning Machine* memiliki tingkat akurasi sebesar 95% dengan *hidden layer* 35 dan nilai *mean squared error* sebesar 0.369 dengan uji skenario 80:20, sedangkan metode *Multilayer Perceptron & Support Vector Machine* memiliki akurasi yang sama yaitu sebesar 91%.

REFERENCES

- [1] S. Nurhayati and E. T. Luthfi, "Prediksi Mahasiswa Drop Out Menggunakan Metode Support Vector Machine," *Sisfotenika*, vol. 5, no. 1, pp. 82–93, 2015.
- [2] M. T. Anwar, L. Heriyanto, and F. Fanini, "Model Prediksi Dropout Mahasiswa Menggunakan Teknik Data Mining," *J. Inform. Upgris*, vol. 7, no. 1, 2021.
- [3] F. A. D. Aji Prasetya Wibawa, Muhammad Guntur Aji Purnama, Muhammad Fathony Akbar, "Metode-metode Klasifikasi," *Pros. Semin. Ilmu Komput. dan Teknol. Inf.*, vol. 3, no. 1, p. 134, 2018.
- [4] A. D. W. M. Sidik, I. Himawan Kusumah, A. Suryana, Edwinanto, M. Artiyasa, and A. Praditha Junfithrana, "Gambaran Umum Metode Klasifikasi Data Mining," *Fidel. J. Tek. Elektro*, vol. 2, no. 2, pp. 34–38, 2020, doi: 10.52005/fidelity.v2i2.111.
- [5] A. Wanto and A. P. Windarto, "Analisis Prediksi Indeks Harga Konsumen Berdasarkan Kelompok Kesehatan Dengan Menggunakan Metode Backpropagation," *J. Penelit. Tek. Inform. Sink.*, vol. 2, no. 2, pp. 37–43, 2017, [Online]. Available: <https://zenodo.org/record/1009223#.Wd7norlTbhQ>
- [6] Yuli Mardi, "Data Mining : Klasifikasi Menggunakan Algoritma C4 . 5 Data mining merupakan bagian dari tahapan proses Knowledge Discovery in Database (KDD) . Jurnal Edik Informatika," *J. Edik Inform.*, vol. 2, no. 2, pp. 213–219, 2019.
- [7] A. K. Pujari, *Data mining techniques*. Universities press, 2001.
- [8] M.-S. Chen, J. Han, and P. S. Yu, "Data mining: an overview from a database perspective," *IEEE Trans. Knowl. Data Eng.*, vol. 8, no. 6, pp. 866–883, Dec. 1996, doi: 10.1109/69.553155.
- [9] Y. Miche, A. Sorjamaa, and A. Lendasse, "OP-ELM: theory, experiments and a toolbox," in *International conference on Artificial Neural networks*, 2008, pp. 145–154.
- [10] M. I. S. K. M. S. Mustafa, "Student Prediction of Drop Out Using Extreme Learning Machine (ELM) Algorithm".
- [11] F. Pakaja, A. Naba, and Purwanto, "Peramalan Penjualan Mobil Menggunakan Jaringan Syaraf Tiruan dan Certainty Factor," *Eeccis*, vol. 6, no. 1, pp. 23–28, 2012.
- [12] T. A. Fetty and Y. P. Intan, "Jaringan Saraf Tiruan Dan Modifikasinya Menggunakan Supervised Learning." Indomedia Pustaka, 2019.
- [13] A. Sudarsono, "Jaringan Syaraf Tiruan Untuk Memprediksi Laju Pertumbuhan Penduduk Menggunakan Metode Bacpropagation (Studi Kasus Di Kota Bengkulu)," *J. Media Infotama*, vol. 12, no. 1, pp. 61–69, 2016, doi: 10.37676/jmi.v12i1.273.
- [14] S. Ding, X. Xu, and R. Nie, "Extreme learning machine and its applications," *Neural Comput. Appl.*, vol. 25, no. 3–4, pp. 549–556, 2014, doi: 10.1007/s00521-013-1522-8.
- [15] L. Noriega, "Multilayer perceptron tutorial," *Sch. Comput. Staff. Univ.*, pp. 1–12, 2005, [Online]. Available: <http://www.amno.moph.go.th/research/uploadfile/1365058846mlp.pdf>
- [16] H. Ramchoun, M. Amine, J. Idrissi, Y. Ghanou, and M. Ettaoui, "Multilayer Perceptron: Architecture Optimization and Training," *Int. J. Interact. Multimed. Artif. Intell.*, vol. 4, no. 1, p. 26, 2016, doi: 10.9781/ijimai.2016.415.
- [17] D. Setianingsih, F. Rosmi, G. Santoso, and A. Virginia, "Implementasi Pendidikan Karakter Dalam Pembelajaran PKn di Sekolah Dasar," *DIKDAS MATAPPA J. Ilmu Pendidik. Dasar*, vol. 3, no. 2, p. 279, 2020, doi: 10.31100/dikdas.v3i2.693.
- [18] S. Visa, B. Ramsay, A. Ralescu, and E. Van der Knaap, "Edited by Sofia Visa, Atsushi Inoue, and Anca Ralescu," *Maics*, vol. 710, pp. 120–127, 2011.
- [19] L. M. Jannah, "Metode penelitian kuantitatif," 2019.
- [20] H. Pratiwi and K. Harianto, "Perbandingan Algoritma ELM Dan Backpropagation Terhadap Prestasi Akademik Mahasiswa," *J-SAKTI (Jurnal Sains Komput. dan Inform.*, vol. 3, no. 2, p. 282, 2019, doi: 10.30645/j-sakti.v3i2.147.