

BAB IV HASIL DAN PEMBAHASAN

4.1 Pengumpulan Data

Data diperoleh dari Kepala Pusat Data dan Sistem Informasi (PDSI) Universitas Nurul Jadid yang kemudian disederhanakan sehingga dapat digunakan. Dalam penelitian ini menggunakan 70 data mahasiswa Universitas Nurul Jadid angkatan 2018 yang terdiri dari 35 mata kuliah selama semester 1 sampai semester 6 dan hanya mengambil data dari mahasiswa program studi Teknik Informatika. Berikut adalah contoh sampel data yang digunakan, ditunjukkan pada tabel 4.1 dibawah ini.

Tabel 4.1 Sampel data awal

No	Nim	Thn. Akademik	Semester	Mata Kuliah			Nilai			Lulus
				Kode	Nama	Sks	Akhir	Huruf	Angka	
1	1842100123	2018/2019 Ganjil	1	MKU121403	Dasar Pemrograman	3	69,00	C+	2,50	Ya
2	1842100123	2018/2019 Ganjil	1	MKD121410	Fisika Teknik	3	64,00	C	2,00	Ya
3	1842100123	2018/2019 Ganjil	1	MKD121409	Kalkulus	3	74,00	B	3,00	Ya
4	1842100123	2018/2019 Ganjil	1	MKD121401	Pendidikan Agama Islam	2	79,00	B+	3,50	Ya
5	1842100123	2018/2019 Ganjil	1	MKU121401	Pengantar Informatika	2	100,00	A	4,00	Ya
6	1842100123	2018/2019 Ganjil	1	MKD121414	Probabilitas dan Statistika	3	100,00	A	4,00	Ya

Tabel 4.2 Sampel data awal lanjutan

No	Nim	Thn. Akademik	Semester	Mata Kuliah			Nilai			Lulus
				Kode	Nama	Sks	Akhir	Huruf	Angka	
7	1842100123	2018/2019 Ganjil	1	MKU121402	Sistem Informasi Manajemen	3	100,00	A	4,00	Ya
8	1842100123	2018/2019 Genap	2	MKU121405	Algoritma dan Struktur Data	3	74,00	B	3,00	Ya
9	1842100123	2018/2019 Genap	2	MKD121411	Aljabar Linier	3	74,00	B	3,00	Ya
10	1842100123	2018/2019 Genap	2	MKD121407	Bahasa Inggris I	2	74,00	B	3,00	Ya
11	1842100123	2018/2019 Genap	2	MKU121404	Basis Data	3	74,00	B	3,00	Ya
12	1842100123	2018/2019 Genap	2	MKU121406	Organisasi dan Arsitektur Komputer	3	64,00	C	2,00	Ya
13	1842100123	2018/2019 Genap	2	MKU121407	Perencanaan Berorientasi Objek	3	69,00	C+	2,50	Ya
14	1842100123	2018/2019 Genap	2	MKD121402	Pendidikan Pancasila	2	100,00	A	4,00	Ya
15	1842100123	2019/2020 Ganjil	3	MKU121409	Animasi Desain Grafik	3	69,00	C+	2,50	Ya

Tabel 4.3 Sampel data awal lanjutan

No	Nim	Thn. Akademik	Semester	Mata Kuliah			Nilai			Lulus
				Kode	Nama	Sks	Akhir	Huruf	Angka	
16	1842100123	2019/2020 Ganjil	3	MKU121411	Grafika Komputer	3	39,00	E	0,00	Tidak
17	1842100123	2019/2020 Ganjil	3	MKU121412	Komputasi Numerik	3	74,00	B	3,00	Ya
18	1842100123	2019/2020 Ganjil	3	MKD121412	Matematika Diskrit	3	74,00	B	3,00	Ya
19	1842100123	2019/2020 Ganjil	3	MKU121410	Pemrograman Desktop	3	69,00	C+	2,50	Ya
20	1842100123	2019/2020 Ganjil	3	MKD121403	Pendidikan Kewarganegaraan	2	100,00	A	4,00	Ya
21	1842100123	2019/2020 Ganjil	3	MKU121408	Sistem Digital	3	74,00	B	3,00	Ya
22	1842100123	2019/2020 Genap	4	MKU121418	Jaringan Komputer	3	100,00	A	4,00	Ya
23	1842100123	2019/2020 Genap	4	MKU121417	Kecerdasan Buatan	3	100,00	A	4,00	Ya
24	1842100123	2019/2020 Genap	4	MKD121417	Kuliah Kerja Nyata	2	100,00	A	4,00	Ya
25	1842100123	2019/2020 Genap	4	MKU121415	Pemrograman Perangkat Bergerak I	3	74,00	B	3,00	Ya
26	1842100123	2019/2020 Genap	4	MKU121414	PPL	3	100,00	A	4,00	Ya

Tabel 4.4 Sampel data awal lanjutan

No	Nim	Thn. Akademik	Semester	Mata Kuliah			Nilai			Lulus
				Kode	Nama	Sks	Akhir	Huruf	Angka	
27	1842100123	2019/2020 Genap	4	MKU121416	Sistem Operasi	3	100,00	A	4,00	Ya
28	1842100123	2019/2020 Genap	4	MKU121413	Teori Graf	3	79,00	B+	3,50	Ya
29	1842100123	2020/2021 Ganjil	5	MKU121411	Grafika Komputer	3	77,00	B+	3,50	Ya
30	1842100123	2020/2021 Ganjil	5	MKU121421	Keamanan Informasi dan Jaringan	3	80,00	A-	3,75	Ya
31	1842100123	2020/2021 Ganjil	5	MKU121420	Kecerdasan Komputasional	3	82,00	A-	3,75	Ya
32	1842100123	2020/2021 Ganjil	5	MKP121401	Manajemen Proyek Perangkat Lunak	3	92,00	A	4,00	Ya
33	1842100123	2020/2021 Ganjil	5	MKD121413	Metode Penelitian	2	85,00	A	4,00	Ya
34	1842100123	2020/2021 Ganjil	5	MKU121419	Pemrograman Web I	3	64,00	C+	2,50	Ya
35	1842100123	2020/2021 Ganjil	5	MKD121415	Riset Operasional	3	68,00	B-	2,75	Ya
36	1842100123	2020/2021 Ganjil	5	MKP121402	Game	3	84,00	A-	3,75	Ya

Tabel 4.5 Sampel data awal lanjutan

No	Nim	Thn. Akademik	Semester	Mata Kuliah			Nilai			Lulus
				Kode	Nama	Sks	Akhir	Huruf	Angka	
37	1842100123	2020/2021 Genap	6	MKP121404	Data Mining	3				
38	1842100123	2020/2021 Genap	6	MKL121401	Jaringan Saraf Tiruan	3				
39	1842100123	2020/2021 Genap	6	MKD121406	Kepesantrenan	2	92,00	A	4,00	Ya
40	1842100123	2020/2021 Genap	6	MKL121405	Pemrograman Perangkat Bergerak II	3				
41	1842100123	2020/2021 Genap	6	MKP121403	Pengolahan Citra Digital	3				
42	1842100123	2020/2021 Genap	6	MKD121418	Praktek Kerja Lapangan	3				
43	1842100123	2020/2021 Genap	6	MKL121406	Sistem Informasi Geografis	3				
44	1842100123	2020/2021 Genap	6	MKL121402	Sistem Pendukung Keputusan	3				

Selanjutnya data diolah menggunakan Microsoft Excel, dimana data mata kuliah dan nilai mahasiswa yang awalnya berupa kolom menjadi baris. Seperti pada **tabel 4.6** dibawah ini.

Tabel 4.6 Sampel Data Sesudah Diolah

Nim	Dasar Pemrograman	Fisika Teknik	Kalkulus	Pendidikan Agama Islam	Pengantar Informatika	Probabilitas dan Statistika	Sistem Informasi Manajemen
1842100124	2,50	2,50	3,00	3,50	4,00	4,00	4,00
1842100125	2,00	3,50	3,00	3,00	4,00	4,00	3,00
1842100128	3,00	3,00	4,00	3,50	4,00	4,00	3,00
1842100129	2,00	2,50	3,00	3,50	4,00	4,00	3,00
1842100132	2,00	2,00	3,00	2,50	4,00	3,00	3,00
1842100135	3,00	3,00	4,00	3,50	4,00	3,50	4,00
1842100136	2,50	3,00	3,00	3,50	4,00	4,00	4,00
1842100138	3,00	3,00	2,00	0,00	4,00	3,00	3,00
1842100141	2,50	2,50	2,50	4,00	4,00	4,00	4,00
1842100143	2,50	3,00	2,00	3,50	4,00	3,00	2,50
1842100144	2,00	2,50	3,00	3,50	4,00	3,00	4,00
1842100151	4,00	3,00	3,00	3,50	4,00	4,00	3,50
1842100152	3,50	3,00	3,50	3,50	4,00	3,50	3,00
1842100154	2,50	3,00	4,00	3,50	4,00	4,00	4,00
1842100160	2,00	2,50	3,00	3,50	4,00	4,00	3,50

Kemudian data tersebut diolah lagi menjadi format csv untuk mempermudah ketika pengolahan dengan pustaka-pustaka perangkat lunak yang ada pada *python*. Kemudian data yang dibutuhkan hanyalah mata kuliah, nilai mata kuliah setiap mahasiswa serta jurusan yang berisi label dari bidang keilmuan yang diambil, serta membuat mata kuliah menjadi setiap kolom seperti pada **Tabel 4.6** diatas.

4.2 Pre-processing

Hal pertama yang dilakukan adalah menyambungkan *google collab* dengan *google drive* serta mengimpor pustaka yang dibutuhkan untuk

pengolahan data seperti yang terlihat pada **segmen 4.1**. Setelah dataset dipersiapkan maka selanjutnya dataset tersebut dibaca pada *google colab* menggunakan library *pandas*, seperti yang terlihat pada **segmen program 4.2** dibawah ini.

Segmen Program 4.1 Koneksi *Google Colab*

```
#koneksi google drive
from google.colab import drive
drive.mount('/content/drive')
```

Segmen Program 4.2 Menampilkan dataset

```
# impor pustaka
import pandas as pd
import numpy as np

# Baca dan menampilkan dataset
df = pd.read_csv('/content/drive/MyDrive/Skripsi 2021/Dataset
nilai mahasiswa angkatan 2018.csv', sep=',', decimal=',')
df.head()
```

Adapun hasil dari membaca dataset ditunjukkan pada **gambar 4.1** dibawah ini.

	Nim	Dasar Pemrograman	Fisika Teknik	Kalkulus	Pendidikan Agama Islam	Pengantar Informatika	Probabilitas dan Statistika	Sistem Informasi Manajemen	Algoritma dan Struktur Data	Aljabar Linier	Bahasa Inggris I	Basis Data	Organisasi dan Arsitektur Komputer	Pemrograman Berorientasi Objek
0	1842100124	2.5	2.5	3.0	3.5	4.0	4.0	4.0	3.0	3.0	3.0	3.0	2.0	2.5
1	1842100125	2.0	3.5	3.0	3.0	4.0	4.0	3.0	3.0	3.0	0.0	2.5	3.0	2.5
2	1842100128	3.0	3.0	4.0	3.5	4.0	4.0	3.0	3.0	3.5	3.5	3.0	4.0	3.0
3	1842100129	2.0	2.5	3.0	3.5	4.0	4.0	3.0	3.0	3.0	3.0	4.0	1.0	2.5
4	1842100132	2.0	2.0	3.0	2.5	4.0	3.0	3.0	3.0	3.0	3.0	2.5	4.0	2.5

Gambar 4.1 Hasil membaca dataset

Pre-processing dilakukan untuk mempersiapkan dataset agar mendapatkan hasil yang baik, dataset sangat mempengaruhi hasil saat dilakukan proses *training* dengan model, dataset yang bagus terhindar dari adanya *missing value*, proses pre-processing juga dilakukan untuk melihat dataset apakah telah seragam atau masih belum. Seperti pada **gambar 4.2**,

dari setiap mata kuliah telah memiliki tipe data yang sama yaitu *float* dan pada kolom jurusan bertipe data *object* karena nilai daripada kolom jurusan berupa *string*. Hal ini memperlihatkan bahwa dataset sudah seragam dan sesuai.

```

RangeIndex: 70 entries, 0 to 69
Data columns (total 37 columns):
 #   Column                                          Non-Null Count  Dtype
---  -
 0   Nim                                           70 non-null     int64
 1   Dasar Pemrograman                            70 non-null     float64
 2   Fisika Teknik                                70 non-null     float64
 3   Kalkulus                                       70 non-null     float64
 4   Pendidikan Agama Islam                       70 non-null     float64
 5   Pengantar Informatika                        70 non-null     float64
 6   Probabilitas dan Statistika                  70 non-null     float64
 7   Sistem Informasi Manajemen                  70 non-null     float64
 8   Algoritma dan Struktur Data                  70 non-null     float64
 9   Aljabar Linier                                70 non-null     float64
10  Bahasa Inggris I                             70 non-null     float64
11  Basis Data                                    70 non-null     float64
12  Organisasi dan Arsitektur Komputer           70 non-null     float64
13  Pemrograman Berorientasi Objek               70 non-null     float64
14  Pendidikan Pancasila                          70 non-null     float64
15  Animasi Desain Grafik                        70 non-null     float64
16  Grafika Komputer                             70 non-null     float64
17  Komputasi Numerik                            70 non-null     float64
18  Matematika Diskrit                           70 non-null     float64
19  Pemrograman Desktop                          70 non-null     float64
20  Pendidikan Kewarganegaraan                   70 non-null     float64
21  Sistem Digital                               70 non-null     float64
22  Jaringan Komputer                            70 non-null     float64
23  Kecerdasan Buatan                           70 non-null     float64
24  Kuliah Kerja Nyata                           70 non-null     float64
25  Pemrograman Perangkat Bergerak I             70 non-null     float64
26  Perancangan Perangkat Lunak                  70 non-null     float64
27  Sistem Operasi                               70 non-null     float64
28  Teori Graf                                    70 non-null     float64
29  Keamanan Informasi dan Jaringan              70 non-null     float64
30  Kecerdasan Komputasional                     70 non-null     float64
31  Manajemen Proyek Perangkat Lunak             70 non-null     float64
32  Metode Penelitian                            70 non-null     float64
33  Pemrograman Web I                            70 non-null     float64
34  Riset Operasional                            70 non-null     float64
35  Teknik Pengembangan Game                     70 non-null     float64
36  Jurusan                                       70 non-null     object
dtypes: float64(35), int64(1), object(1)
memory usage: 20.4+ KB

```

Gambar 4.2 Info dataset

Pada pustaka *pandas* ada beberapa fungsi untuk menganalisis dataset dengan secara cepat dan juga efektif salah satunya *isnull()*. Fungsi ini dapat melihat dataset apakah ada nilai *missing value* pada setiap kolomnya. Seperti pada **gambar 4.3**, hasil setiap kolom terutama pada mata kuliah mendapat informasi *False* yang artinya tidak ada sebuah *missing value* pada setiap kolomnya.


```
df.isnull().any()
Nim False
Dasar Pemrograman False
Fisika Teknik False
Kalkulus False
Pendidikan Agama Islam False
Pengantar Informatika False
Probabilitas dan Statistika False
Sistem Informasi Manajemen False
Algoritma dan Struktur Data False
Aljabar Linier False
Bahasa Inggris I False
Basis Data False
Organisasi dan Arsitektur Komputer False
Pemrograman Berorientasi Objek False
Pendidikan Pancasila False
Animasi Desain Grafik False
Grafika Komputer False
Komputasi Numerik False
Matematika Diskrit False
Pemrograman Desktop False
Pendidikan Kewarganegaraan False
Sistem Digital False
Jaringan Komputer False
Kecerdasan Buatan False
Kuliah Kerja Nyata False
Pemrograman Perangkat Bergerak I False
Perancangan Perangkat Lunak False
Sistem Operasi False
Teori Graf False
Keamanan Informasi dan Jaringan False
Kecerdasan Komputasional False
Manajemen Proyek Perangkat Lunak False
Metode Penelitian False
Pemrograman Web I False
Riset Operasional False
Teknik Pengembangan Game False
Jurusan False
dtype: bool
```

Gambar 4.3 Dataset tidak ada *missing value*

Kemudian setelah menganalisis dataset dari *missing value* maka selanjutnya perlu dilakukan pemisahan data yaitu data masukan (*input x*) dan data target (*output y*) karena *neural network* menerima sebuah inputan *x* dan menghasilkan target *output y*.

1. Pemisahan data (*input x*)

Pemisahan data latih pada variabel *x* hanya menampung data yang akan menjadi sebuah inputan yaitu nilai mata kuliah, maka pada nilai setiap kolom mata kuliah akan diambil dan ditampung pada variabel *x*.

Segmen Program 4.3 Pemisahan data input X

```
# ambil hanya data nilai saja dari tiap mata kuliah dan
# tampung pada variabel x
#pemisahan data input x
x = df.iloc[:,1:36].values
print(x[0])
```

seperti pada **gambar 4.4** ketika di tampilkan isi dari variabel *x* hanyalah sebuah angka yaitu nilai setiap mata kuliah.

```
[2.5 2.5 3. 3.5 4. 4. 4. 3. 3. 3. 3. 2. 2.5 4.
4. 2. 4. 3.5 3. 4. 3. 4. 4. 4. 3. 4. 4. 3.
3.75 3.75 4. 4. 3.75 3.75 3.75]
```

Gambar 4.4 Hasil input X

2. Pemisahan data target (*output y*)

Pemisahan data target pada variabel *y* hanya menampung data dari kolom jurusan yang merupakan label daripada mata kuliah pilihan.

Segmen Program 4.4 Pemisahan data target Y

```
# ambil data hanya pada kolom jurusan saja dan
# tampung pada variabel y
y = df.iloc[:,36].values
print(y)
```

Seperti pada **gambar 4.5** ketika ditampilkan isi daripada variabel *y* adalah 2 label pada kolom jurusan yaitu komputasi bergerak dan komputasi cerdas.

pembagian dilakukan dengan persentase dari total dataset yaitu 80% untuk data latih dan 20% untuk data tes.

Segmen Program 4.7 Data latih dan data tes

```
# pustaka pembagi data
from sklearn.model_selection import train_test_split

# pisahkan data training dan testing
# 80% data latih | 20% data test
X_train, X_test, Y_train, Y_test = train_test_split(X_scaled,
y, test_size=0.2, random_state=0)
```

Pada gambar 4.8 diketahui bahwa data latih berjumlah 56 baris dan data tes berjumlah 14 baris.

```
isi x_train : [0.75      0.66666667 1.         0.875      1.         1.
1.         1.         1.         1.         0.875      1.
1.         0.         1.         1.         1.         1.
1.         1.         0.875      1.         1.         1.
1.         1.         1.         1.         0.875      0.75
1.         1.         0.875      0.75      0.9375     ]
-----
ukuran x_train : (56, 35) |
-----
isi X_test : [1.         0.66666667 1.         0.875      1.         0.75
1.         1.         1.         0.75      1.         0.83333333
1.         0.         1.         1.         0.875      0.83333333
0.75      1.         0.75      1.         1.         1.
0.75      1.         0.875      1.         0.9375     0.9375
1.         1.         1.         1.         0.9375     ]
-----
ukuran X_test : (14, 35) |
-----
isi Y_train : [0 1 1 1 1 0 1 1 0 1 0 0 1 1 1 0 1 1 1 1 1 0 0 1 0 0 1 1 0 0 0 0 0 0 1 0 0 0 1
0 1 1 1 1 1 0 1 0 1 0 0 0 1 0 0 1 1 1]
-----
ukuran Y_train : (56,) |
-----
isi Y_test : [0 0 1 0 0 1 0 1 0 1 1 0 0 1]
-----
ukuran Y_test : (14,) |
-----
```

Gambar 4.8 Hasil pembagian data latih dan data tes

4.3 Implementasi Metode

Pada tahapan ini dilakukan dengan cara melatih data dengan model *neural network* yang nantinya model dapat mengklasifikasikan jurusan atau mata kuliah pilihan serta dapat memprediksi mahasiswa yang hendak mengambil mata kuliah pilihan berdasarkan nilai yang diperoleh mahasiswa dan mengklasifikasikan dalam bidang komputasi bergerak atau komputasi cerdas.

Segmen Program 4.8 Arsitektur model NN

```
# arsitektur model nn
model = Sequential([
    Dense(32, activation='relu', input_shape=(3
5,)),
    Flatten(),
    Dense(32, activation='relu'),
    Dropout(0.2),
    Dense(64, activation='relu'),
    Dropout(0.2),
    Dense(1, activation='sigmoid'),])
model.compile(optimizer='adam', loss='binary_crossentropy',
metrics=['accuracy'])
model.summary()
```

Pada **segmen gambar 4.9** diatas merupakan model *neural network* yang digunakan pada penelitian ini. Pada lapisan input terdapat 32 *neuron* menggunakan fungsi aktivasi *relu* dan diisi dengan banyaknya data mata kuliah yaitu 35 dan dilakukan *flattening*. Kemudian pada lapisan selanjutnya juga menggunakan 32 *neuron* dan fungsi aktivasinya *relu*. Setiap lapisan *neural network* dengan fungsi aktivasi *relu* akan dilakukan *dropout* sebesar 20% dimana setiap lapisan akan diberhentikan sementara secara acak *dropout* juga dapat mencegah terjadinya *overfitting* pada model dan juga mempercepat proses pelatihan. Pada lapisan selanjutnya menggunakan lapisan dengan 64 *neuron* dan fungsi aktivasi *relu* dan juga dilakukan *dropout*. Pada lapisan output hanya ada 1 *neuron* dan menggunakan fungsi aktivasi *sigmoid* karena kasus penelitian ini adalah *binary classification* atau klasifikasi dua kelas. Optimasi model menggunakan *adam* dimana algoritma ini merupakan parameter *default* dari *keras*. Penghitung nilai kerugian menggunakan *binary_crossentropy* untuk masalah klasifikasi dua kelas dan metrik *accuracy* untuk mengevaluasi performa model.

Untuk melihat hasil dari model *Neural Network* bisa dilihat pada **gambar 4.9** dibawah ini.

```
Model: "sequential"
```

Layer (type)	Output Shape	Param #
dense (Dense)	(None, 32)	1152
flatten (Flatten)	(None, 32)	0
dense_1 (Dense)	(None, 32)	1056
dropout (Dropout)	(None, 32)	0
dense_2 (Dense)	(None, 64)	2112
dropout_1 (Dropout)	(None, 64)	0
dense_3 (Dense)	(None, 1)	65

```
Total params: 4,385
Trainable params: 4,385
Non-trainable params: 0
```

Gambar 4.9 Hasil model NN

Segmen Program 4.9 Latih model

```
# latih model
history = model.fit(
    X_train, Y_train,
    validation_data=(X_test, Y_test),
    epochs=100,
    verbose=2)
```

Setelah perancangan model selesai maka model siap dilakukan proses *training*. Seperti pada **gambar 4.10** dibawah merupakan proses pelatihan terhadap model dengan data latih dan data uji dimana *epoch* model 100 yaitu model akan belajar selama 100x. Pada **gambar 4.10** terlihat hasil akurasi pada *epoch* 100 mendapatkan 83% dan validasi akurasi mendapatkan 78%, nilai *loss* 41% dan nilai validasi *loss* 62%.

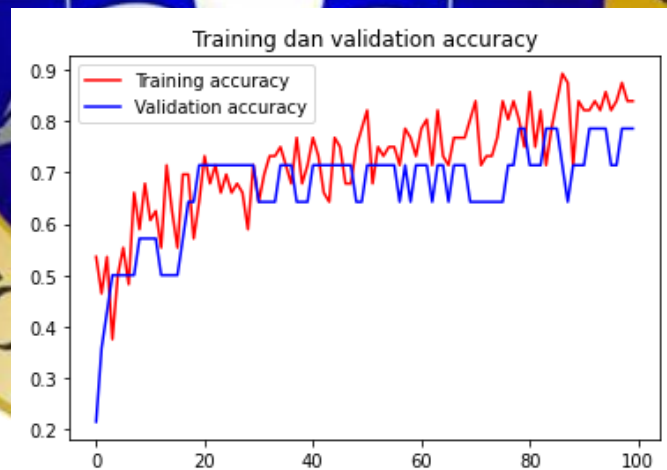
```

2/2 - 0s - loss: 0.4873 - accuracy: 0.7143 - val_loss: 0.6190 - val_accuracy: 0.7857
Epoch 85/100
2/2 - 0s - loss: 0.4946 - accuracy: 0.7857 - val_loss: 0.6174 - val_accuracy: 0.7857
Epoch 86/100
2/2 - 0s - loss: 0.4190 - accuracy: 0.8393 - val_loss: 0.6172 - val_accuracy: 0.7857
Epoch 87/100
2/2 - 0s - loss: 0.4094 - accuracy: 0.8929 - val_loss: 0.6236 - val_accuracy: 0.7143
Epoch 88/100
2/2 - 0s - loss: 0.4504 - accuracy: 0.8750 - val_loss: 0.6312 - val_accuracy: 0.6429
Epoch 89/100
2/2 - 0s - loss: 0.4768 - accuracy: 0.7143 - val_loss: 0.6311 - val_accuracy: 0.7143
Epoch 90/100
2/2 - 0s - loss: 0.4256 - accuracy: 0.8393 - val_loss: 0.6300 - val_accuracy: 0.7143
Epoch 91/100
2/2 - 0s - loss: 0.4649 - accuracy: 0.8214 - val_loss: 0.6280 - val_accuracy: 0.7143
Epoch 92/100
2/2 - 0s - loss: 0.4041 - accuracy: 0.8214 - val_loss: 0.6246 - val_accuracy: 0.7857
Epoch 93/100
2/2 - 0s - loss: 0.4021 - accuracy: 0.8393 - val_loss: 0.6227 - val_accuracy: 0.7857
Epoch 94/100
2/2 - 0s - loss: 0.4460 - accuracy: 0.8214 - val_loss: 0.6245 - val_accuracy: 0.7857
Epoch 95/100
2/2 - 0s - loss: 0.4431 - accuracy: 0.8571 - val_loss: 0.6232 - val_accuracy: 0.7857
Epoch 96/100
2/2 - 0s - loss: 0.4179 - accuracy: 0.8214 - val_loss: 0.6288 - val_accuracy: 0.7143
Epoch 97/100
2/2 - 0s - loss: 0.3821 - accuracy: 0.8393 - val_loss: 0.6217 - val_accuracy: 0.7143
Epoch 98/100
2/2 - 0s - loss: 0.3994 - accuracy: 0.8750 - val_loss: 0.6320 - val_accuracy: 0.7857
Epoch 99/100
2/2 - 0s - loss: 0.3933 - accuracy: 0.8393 - val_loss: 0.6323 - val_accuracy: 0.7857
Epoch 100/100
2/2 - 0s - loss: 0.4164 - accuracy: 0.8393 - val_loss: 0.6201 - val_accuracy: 0.7857

```

Gambar 4.10 Proses epoch

Untuk melihat grafik dari hasil *training validation accuracy* dan Grafik hasil *training validation loss* bisa dilihat pada **gambar 4.11** & **gambar 4.12** dibawah ini.



<Figure size 432x288 with 0 Axes>

Gambar 4.11 Grafik hasil *training validation accuracy*



<Figure size 432x288 with 0 Axes>

Gambar 4.12 Grafik hasil *training* *validaton* *loss*

4.4 Uji Coba

Setelah dilakukan *training* pada model, maka selanjutnya dilakukan pengujian pada model untuk mengetahui kinerja model. Kemudian untuk melihat hasil evaluasi terhadap data tes ditunjukkan pada **gambar 4.13** dibawah ini. Hasil dari data latih mendapat skor 37% dan akurasi mendapat 89%. Sedangkan pada data tes mendapat skor 62% dan akurasi sebesar 78%.

Segmen Program 4.10 Evaluasi model

```
# evaluasi model
score, acc = model.evaluate(X_train, Y_train, batch_size=10)
print('Train score:', score)
print('Train accuracy:', acc)

y_predis = model.predict(X_test)
y_preds = (y_predis > 0.5)

print('*'*20)
score, acc = model.evaluate(X_test, Y_test, batch_size=10)

print('Test score:', score)
print('Test accuracy:', acc)
```

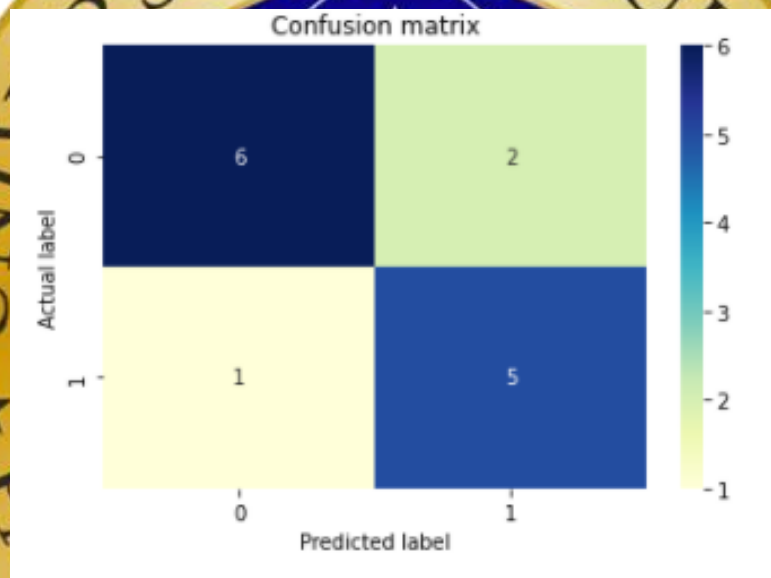
```

6/6 [=====] - 0s 2ms/step - loss: 0.3712 - accuracy: 0.8929
Train score: 0.37121957540512085
Train accuracy: 0.8928571343421936
*****
2/2 [=====] - 0s 4ms/step - loss: 0.6201 - accuracy: 0.7857
Test score: 0.6200622916221619
Test accuracy: 0.7857142686843872

```

Gambar 4.13 Akurasi Data Uji

Kemudian untuk melihat hasil prediksi terhadap data tes akan ditunjukkan melalui *confusion matrix* pada **gambar 4.14** dibawah ini. Dari 14 yang terdapat pada data tes 8 terdeteksi label 0 (komputasi bergerak) dan 6 terdeteksi label 1 (komputasi cerdas)



Gambar 4.14 Hasil *confusion matrix*

Kemudian untuk mengukur seberapa baik kinerja model prediksi terhadap data tes dari data latih akan ditunjukkan melalui nilai *precision*, *recall*, *f-1 score* dan *support* . Seperti pada **gambar 4.15** dibawah ini.

Segmen Program 4.11 Uji performa model

```

# uji performa model
from sklearn.metrics import classification_report
print(classification_report(Y_test, y_pred))

```

Adapun hasil dari uji performa model mendapatkan accuracy sebesar 0.79 atau 79%. Seperti yang ditunjukkan pada **gambar 4.15** dibawah ini.

	precision	recall	f1-score	support
0.0	0.86	0.75	0.80	8
1.0	0.71	0.83	0.77	6
accuracy			0.79	14
macro avg	0.79	0.79	0.78	14
weighted avg	0.80	0.79	0.79	14

Gambar 4.15 Hasil uji performa model

