

**Laporan Case Based-1**  
**(CII3C3) Pembelajaran Mesin**



oleh

Akmal Muhamad Firdaus – 1301204188

Kode dosen pengampu :

**BDP**

INFORMATIKA  
FAKULTAS INFORMATIKA  
UNIVERSITAS TELKOM

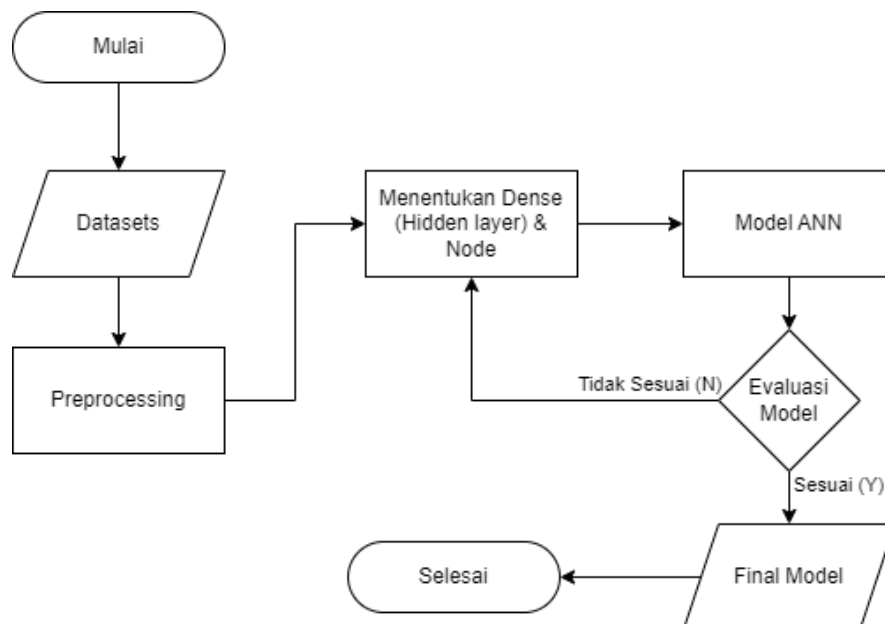
2022

Saya mengerjakan tugas ini dengan cara yang tidak melanggar aturan perkuliahan dan kode etik akademisi.

## 1. Ikhtisar

### 1.1 Alur Program

Pada tugas ini, saya menggunakan metode ANN untuk klasifikasi Aritmia pada *dataset* [data\\_arrhythmia.csv](#). Tahap pertama dilakukan pengumpulan data, tahap kedua dilakukan preprocessing data, kemudian mengimplementasikan model usulan, melakukan evaluasi model untuk melakukan perbaikan pada model. Diagram alur program yang akan dirancang adalah sebagai berikut.



**Gambar 1.1** Flowchart program

### 1.2 Datasets

Pada tugas ini, *dataset* yang digunakan adalah [data\\_arrhythmia.csv](#), berjumlah 452 records dan 280 atribut. Berikut adalah example data dari dataset tersebut.

	age	sex	height	weight	qrs_duration	p-r_interval	q-t_interval	t_interval	p_interval	qrs	...	KY	KZ	LA	LB	LC	LD	LE	LF	LG	diagnosis
0	75	0	190	80	91	193	371	174	121	-16	...	0.0	9.0	-0.9	0.0	0	0.9	2.9	23.3	49.4	8
1	56	1	165	64	81	174	401	149	39	25	...	0.0	8.5	0.0	0.0	0	0.2	2.1	20.4	38.8	6
2	54	0	172	95	138	163	386	185	102	96	...	0.0	9.5	-2.4	0.0	0	0.3	3.4	12.3	49.0	10
3	55	0	175	94	100	202	380	179	143	28	...	0.0	12.2	-2.2	0.0	0	0.4	2.6	34.6	61.6	1
4	75	0	190	80	88	181	360	177	103	-16	...	0.0	13.1	-3.6	0.0	0	-0.1	3.9	25.4	62.8	7

**Gambar 1.2** Sample dari datasets

### 1.3 Overview Datasets

Pada datasets yang diberikan terdapat 16 class yang masing masing sebaran datanya tidak merata, maka dari itu saya memutuskan untuk membuatnya menjadi binary classification (jantung normal / tidak) dengan tujuan untuk menghindari overfitting / undefittingnya model.

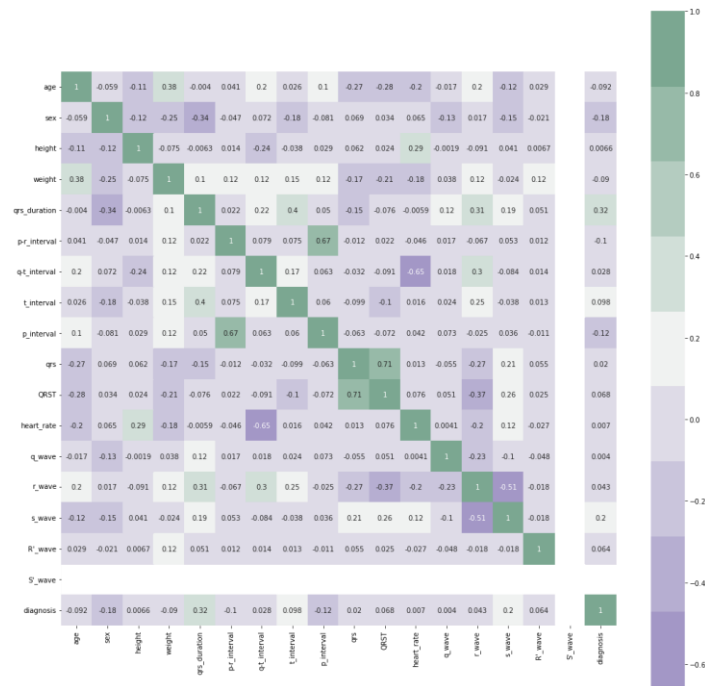
## 2. Pra-pemrosesan data

Pra-pemrosesan data dilakukan dengan menghilangkan atribut yang tidak dibutuhkan, membersihkan records data null/kosong. Hal yang pertama dilakukan adalah mencari dan menganalisis terlebih dahulu data mana saja yang banyak terdapat null data. Setelah dilakukan pengamatan secara manual, atribut bernama “J” dapat dihilangkan karena terlalu banyaknya null data. Lalu selanjutnya saya memutuskan untuk mengambil data yang dijadikan atribut adalah sebagai berikut.

No	Nama Atribut
1.	age
2.	sex
3.	height
4.	weight
5.	qrs_duration
6.	p-r_interval
7.	q-t_interval
8.	t_interval
9.	p_interval
10.	qrs
11.	QRST
12.	heart_rate
13.	q_wave
14.	r_wave
15.	s_wave
16.	R' wave
17.	S' wave
18.	diagnosis

**Tabel 2.1** List atribut

Alasan pengambilan data tersebut dikarenakan data tersebut merupakan data yang familiar digunakan dan terdapat korelasi yang cukup tinggi.



**Gambar 2.1** korelasi antar atribut

Pada gambar 2.1 menunjukan bahwa semakin hijau adalah atribut yang saling berkorelasi.

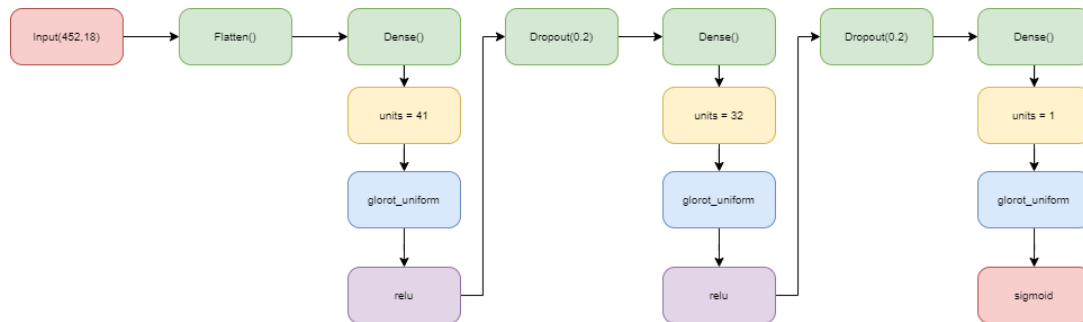
```
cmap = sns.diverging_palette(275,150, s=40, l=65, n=9)
corrmat = df.corr()
plt.subplots(figsize=(18,18))
sns.heatmap(corrmat,cmap= cmap,annot=True, square=True);
```

**Gambar 2.2** Kodingan menentukan cov antara atribut

### 3. Menerapkan Algoritma

#### 3.1 Rancangan Arsitektur Model

Arsitektur model Artificial Neural Networks (ANN) yang diusulkan pada tugas ini terdiri dari input layer, Flatten, Dense, Dropout, dan sigmoid.



**Gambar 3.1** Desain model ANN

```
# Initialising the NN
model = Sequential()
model.add(Flatten())
model.add(Dense(units = 41, kernel_initializer = 'glorot_uniform', activation = 'relu', input_dim = 17))
model.add(Dropout(0.2))
model.add(Dense(units = 32, kernel_initializer = 'glorot_uniform', activation = 'relu'))
model.add(Dropout(0.2))
model.add(Dense(units = 1, kernel_initializer = 'glorot_uniform', activation = 'sigmoid'))
```

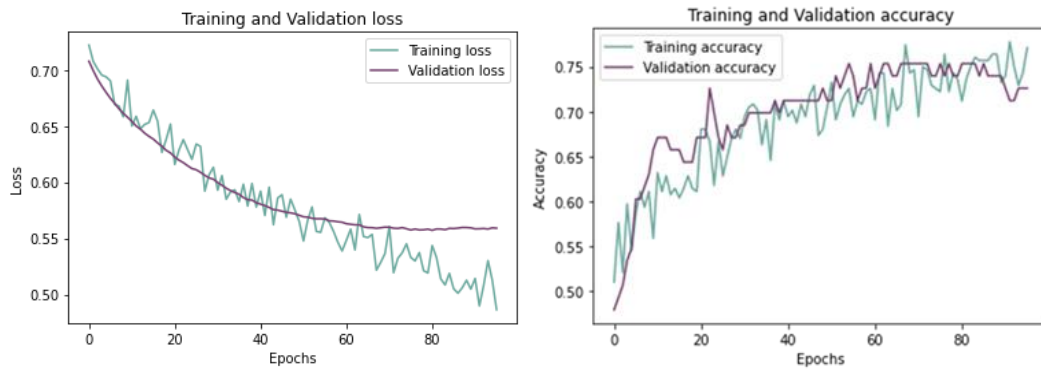
**Gambar 3.2** Kodingan model ANN

#### 3.2 Hyperparameter Tuning

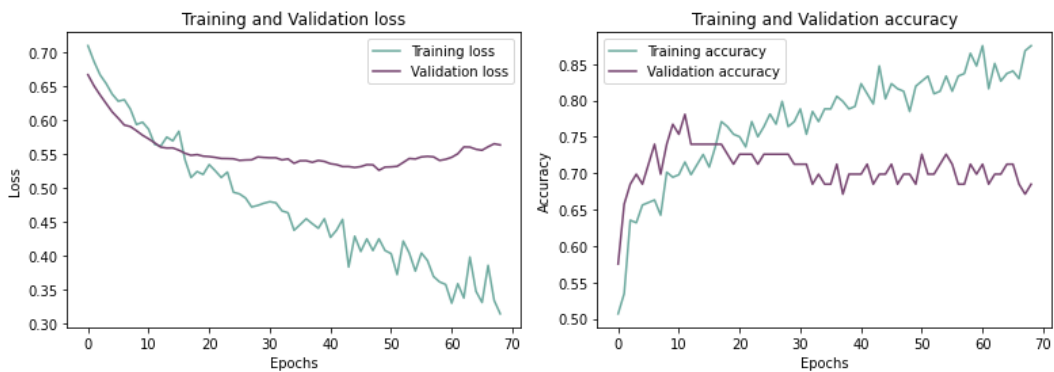
Pemilihan terhadap parameter yang tepat pada model yang diusulkan perlu dilakukan untuk meningkatkan hasil dari klasifikasi. Hyperparameter Tuning memiliki peran penting dalam pembelajaran mesin karena parameter yang dihasilkan mempengaruhi kinerja model ANN secara signifikan. Oleh karena itu Hyperparameter Tuning diusulkan untuk mengoptimalkan pemrosesan datasets medis pada Model ANN yang dibentuk. Pada tugas ini, dilakukan metode hyperparameter tuning untuk menentukan parameter optimizer dan dropout. Pelatihan model pada saat hyperparameter dilakukan menggunakan epoch berjumlah 100. Detail parameter yang akan dibandingkan saat hyperparameter tuning dapat dilihat pada tabel berikut.

Pembanding	Parameter
Optimizer	Adamax, SGD, RMSprop
Dropout	0.2, 0.5

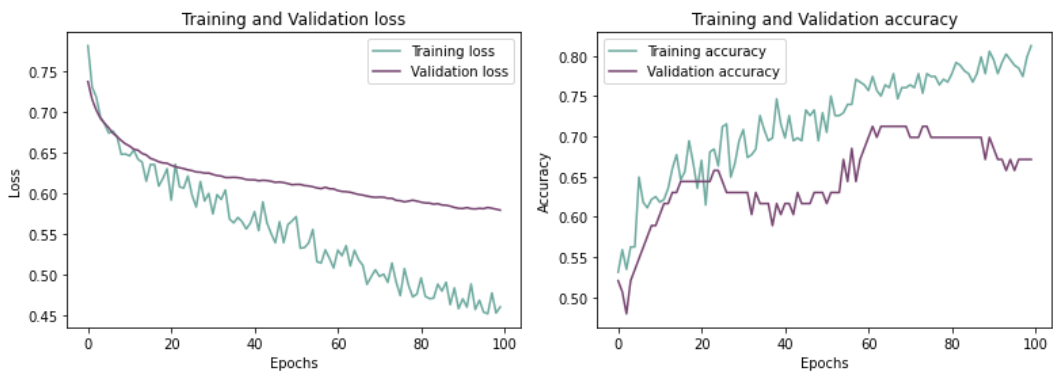
**Tabel 3.1** Pembanding Parameter Hyperparameter Tuning



**Gambar 3.3** (Training & Val loss/acc) with SGD



**Gambar 3.4** (Training & Val loss/acc) with RMSprop



**Gambar 3.5** (Training & Val loss/acc) with Adamax

```
from keras.optimizers import SGD
# Compiling the ANN
model.compile(optimizer = 'SGD', loss = 'binary_crossentropy', metrics = ['accuracy'])
```

**Gambar 3.6** kode untuk mengubah hyperparameter

Skenario	Optimize	Droupout	Akurasi
<b>1</b>	<b>SGD</b>	<b>0.2</b>	<b>75%</b>
2	Adamax	0.2	68%
3	RMSprop	0.2	70%

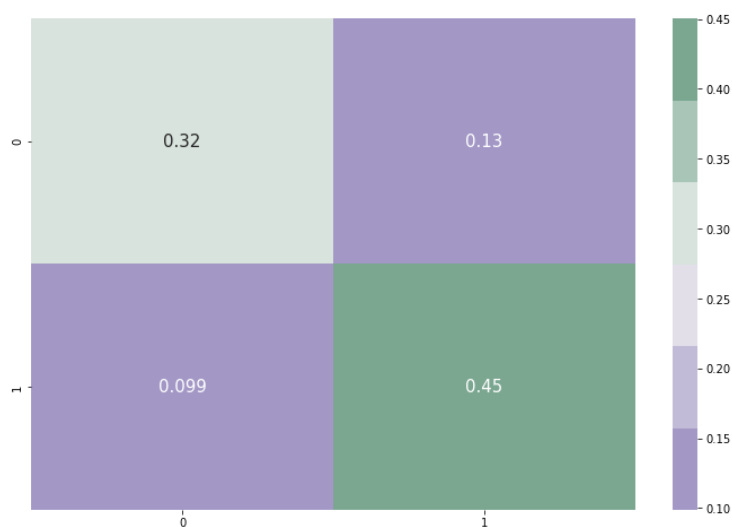
**Tabel 3.2** Hasil Parameter Terbaik dari Hyperparameter Tuning

### 3.3 Hasil Analisis

Berdasarkan hasil dari setiap skenario, terdapat beberapa perbedaan seperti hasil dari scenario pertama(Gambar 3.3) dengan skenario kedua(Gambar 3.4) dan ketiga(Gambar 3.5), sedangkan skenario kedua dan ketiga mendapatkan hasil yang hampir sama dalam loss validasi. Namun, dari ke-3 skenario tersebut, skenario pertama lebih baik dalam akurasi, loss, precision, recall, f1-score. Dapat kita simpulkan bahwa untuk penggunaan optimizer dalam kasus ini yang terbaik adalah menggunakan SGD, karena menghasilkan grafik yang selaras antara training dan validation (tidak overfitting / undefitting). Dan untuk penggunaan dropout menggunakan 0.2.

## 4. Evaluasi hasil

Setelah model dibuat, langkah selanjutnya adalah proses testing(prediction) dari model tersebut. Model akan memprediksi sebanyak 91 data. Berikut adalah hasil dari prediksi tersebut.



**Gambar 4.1** Evaluating the confusion matrix

Dari ke-91 data tersebut didapatkan lah akurasi memprediksi sebesar 77%. Setiap bentuk model dan bentuk hyperparameter sangat berpengaruh terhadap output yang diberikan machine learening. Dengan metode yang sudah disebutkan diatas (poin 3.2) metode tersebut sudahlah sangat optimal yang bisa saya temukan pada datasets lain. Hasil tersebut bisa dikatakan kurang memuaskan, faktor tersebut bias terjadi karena kurang bervariasinya datasets, pengambilan atribut yang salah, terlalu sedikitnya datasets, atau berbagai macam masalah lainnya. Saya menyarankan pembaca agar dapat mengembangkan program dengan algoritma yang berbeda untuk mengetahui bagaimana karakteristik datasets tersebut jika menggunakan algoritma yang berbeda.

Link Video Presentasi : <https://youtu.be/1UPJAFxxqII>

Link Goggle Colab : <https://colab.research.google.com/drive/1F4T9t1jSLLcBhCmp67LjDSwY2kIXRo4g?usp=sharing>