**TUGAS CASE BASED 1  
MACHINE LEARNING**

Logo, company name

Description automatically generated

**Nama**:

Hilman Taris Muttaqin 1301204208

**Kode Dosen**:

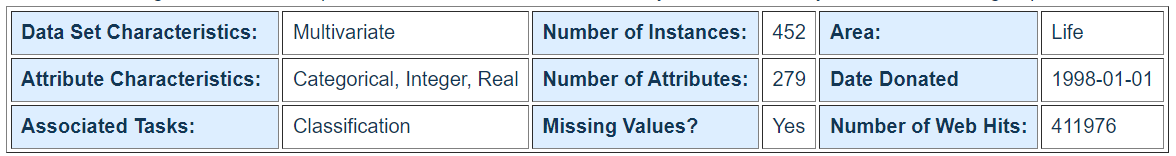
DDR

UNIVERSITAS TELKOM

TAHUN AKADEMIK 2022/2023

1. **Ikhtisar kumpulan data yang dipilih**

Mahasiswa dengan NIM genap mendapatkan data tentang aritmia, Membedakan antara ada dan tidak adanya aritmia jantung. Dibawah ini adalah karakteristik data yang digunakan.



Masalah pertama saat melihat dataset ini adalah memiliki dimensi yang sangat besar, yaitu 279 dimensi data dan memiliki nilai yang bervariasi dari data float minus nol koma, hingga data yang memiliki nilai ratusan.

Saat pertama kali data dibuat menjadi dataframe, hal yang pertama kali terlihat adalah data tersebut tidak memiliki header atau nama kolom.

A picture containing text, electronics, keyboard

Description automatically generated

Sehingga untuk mempermudah pembacaan dan pemrosesan data, dataset akan disatukan terlebih dahulu dengan data kolomnya. Selain itu, secara sekila apabila melihat dataset yang digunakan, data tersebut mengandung banyak data kosong yang direpresentasikan dengan tanda tanya.

A picture containing text

Description automatically generated

Sehingga hal ini menjadi salah satu masalah yang harus diatasi dari data ini.

1. **Ringkasan pra-pemrosesan data yang diusulkan**

Dari masalah-masalah yang sudah dijelaskan diatas, berikut adalah ringkasan Langkah-langkah pra-pemrosesan data yang akan dilakukan.

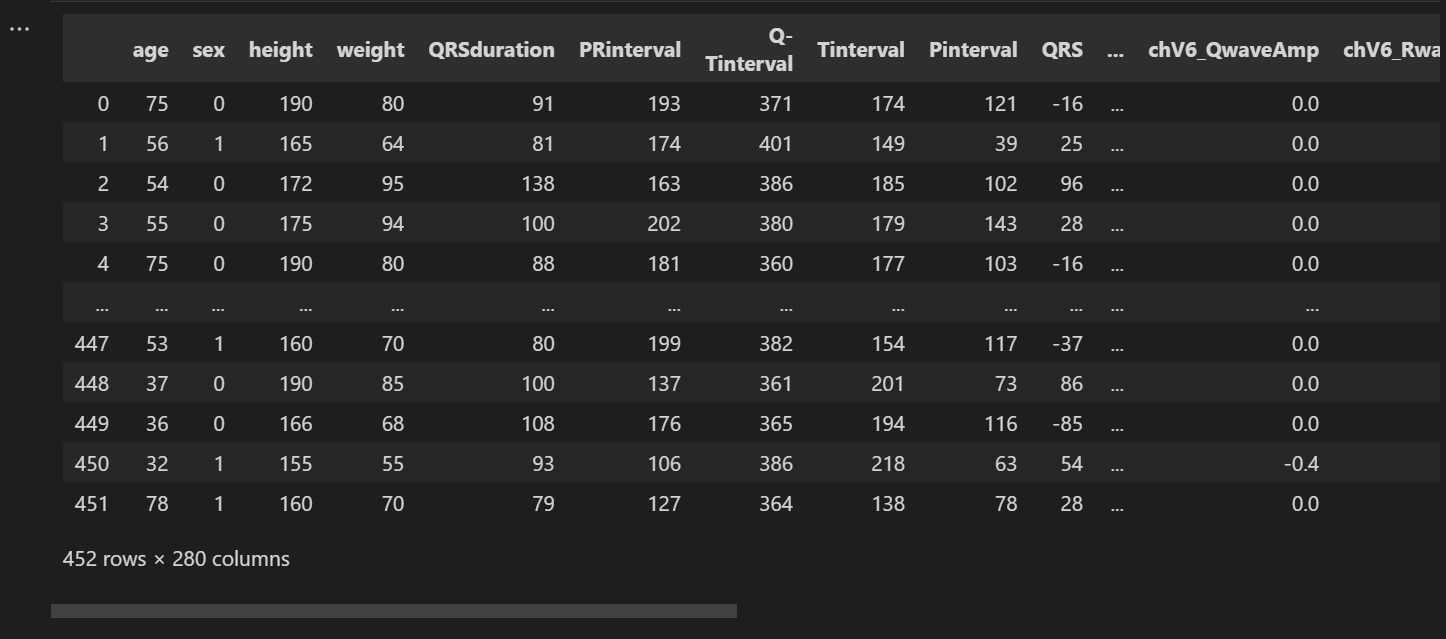
* **Menggabungkan dataset dengan judul kolomnya.**

Hal ini tentunya akan lebih mempermudah pra-pemrosesan data dan visualisasi data. Data yang semula seperti ini:

A picture containing text, electronics, keyboard

Description automatically generated

Menjadi seperti ini:



* **Mengubah data kosong (?) menjadi data NaN.**

Data kosong yang direpresentasikan menggunakan tanda tanya. Hal ini diketahui secara kasat mata ketika data divisualisasikan baik itu menggunakan grafik ataupun tidak menggunakan apa apa (raw data). Hal ini terbukti ketika pengecekan dilakukan.

**Text

Description automatically generated**

Data kosong terdapat pada kolom T, P, QRST, J, dan heartrate. Ubah representasi data kosong menjadi numpy nan.

Graphical user interface, text

Description automatically generated

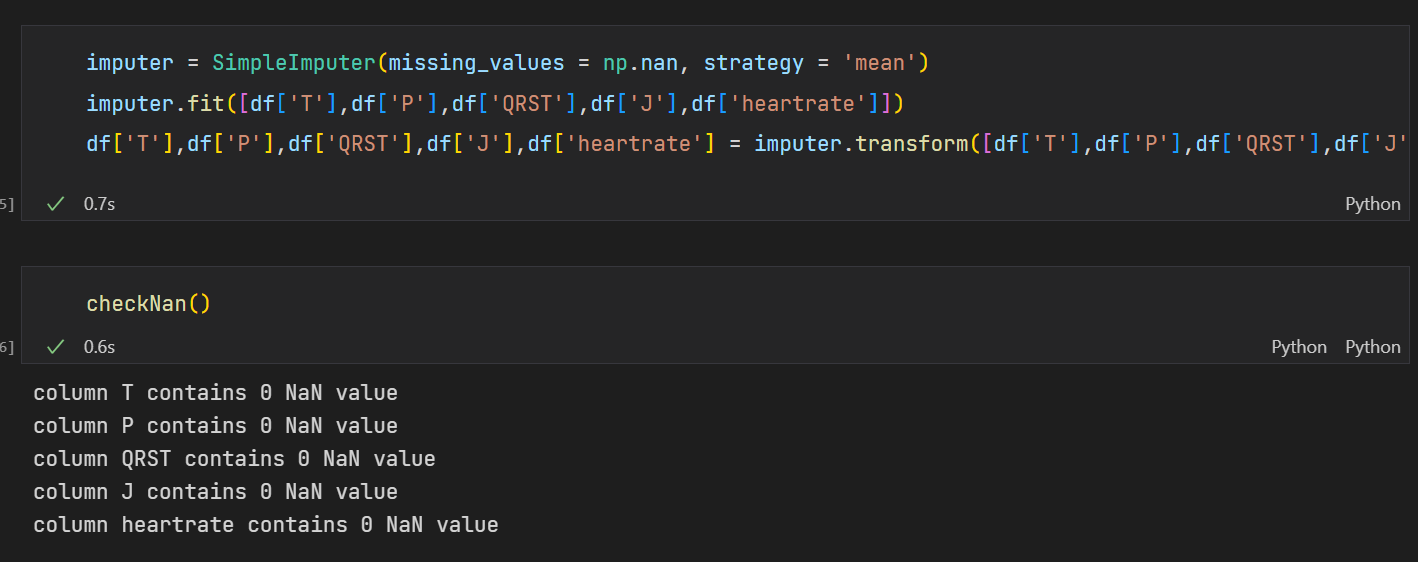
Proses ini bertujuan untunk mempermudah pra-pemrosesan data selanjutnya.

* **Handling data kosong menggunakan Mean Imputation.**

Setelah data kosong dirubah menjadi data NaN, maka selanjutnya adalah proses empty data handling. Terdapat tiga cara untuk handle data tersebut, yaitu:

1. Hapus records
2. Mean/Mode/Median Imputation
3. Model prediksi

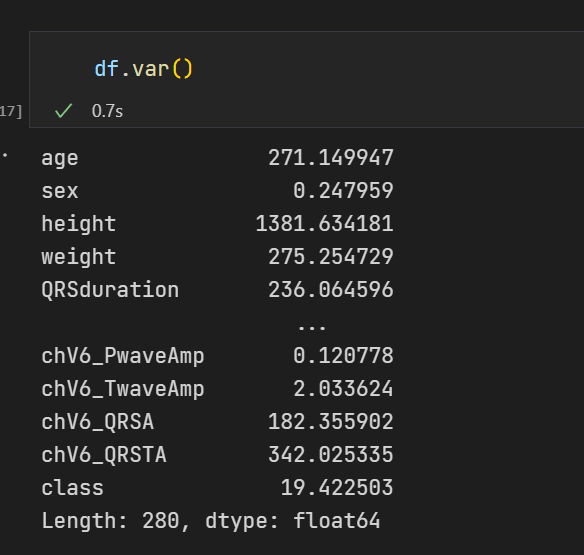
Pada studi kasus ini, cara yang digunakan adalah **Mean Imputation**. Cara tersebut dilakukan untuk menjaga jumlah data tetap sama seperti data awal yang tentunya akan berpengaruh pada performansi pembelajaran mesin.



Setelah dilakukannya proses mean imputation, cek kembali apakah masih terdapat data kosong pada kolom tersebut. Hasilnya data sudah tidak ada yang NaN atau kosong.

* **Melihat kualitas data.**

Jumlah data yang sangat banyak sangat rentan terhadap kesalahan pengukuran dan outliers seperti anomali objek. Untuk mengukur kualitas data, dapat menggunakan variansi dan covariansi.



Graphical user interface, text

Description automatically generated

* **Drop kolom yang memiliki satu nilai.**

Sebelum melakukan scaling, terdapat kolom yang hanya terdapat 1 nilai. Hal tersebut akan memberatkan proses perhitungan kedepannya dan solusinya adalah drop kolom tersebut. Kenapa? karena data dengan 1 nilai tidak memiliki korelasi apapun terhadap record yang ada. Contohnya adalah kolom 'chDI\_SPwave'. Proses ini disebut dimentionality reduction.

Text

Description automatically generated

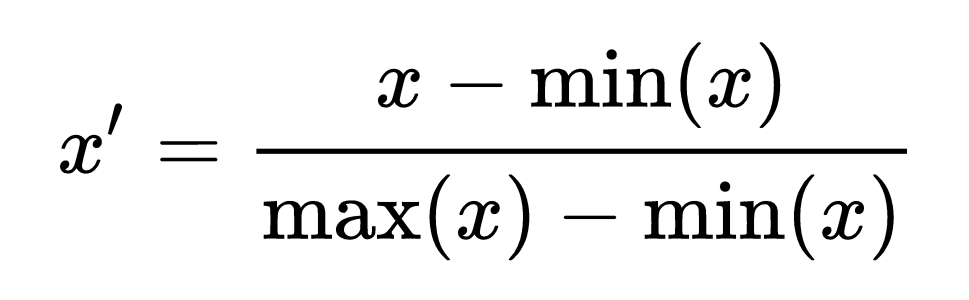
Cek beberapa kolom tersebut apakah benar memiliki 1 nilai saja.

Text

Description automatically generated

Setelah dipastikan, maka drop kolom tersebut. Karena proses ini, dimensi dataset berkurang 17 kolom.

* **Scaling data.**



Untuk scaling data, menggunakan min max scaling. Metode normalisasi min max mengubah kumpulan data menjadi skala mulai dari 0 hingga 1. Proses ini dilakukan untuk memperkecil angka dalam proses perhitungan, proses ini berpengaruh pada kecepatan pembelajaran mesin karena angka yang diproses bukan angka yang besar.

Graphical user interface, text, application

Description automatically generated

Gambar diatas adalah hasil dari min max scaling pada dataset.

1. **Menerapkan algoritma yang dipilih**

Algoritma yang digunakan adalah ANN (Artificial Neural Network) / JST (Jaringan Syaraf Tiruan). JST (Jaringan Syaraf Tiruan) merupakan kecerdasan buatan yang memiliki karakteristik mirip dengan jaringan syaraf biologi. JST dibangun dengan prinsip dasar perambatan sinyal sehingga dapat mengenali pola pelatihan. Perceptron merupakan analogi dari Neuron pada otak manusia (sel syaraf buatan).

Gambaran sederhana perceptron

Input function, summirizing function, and actication function.

Input function:

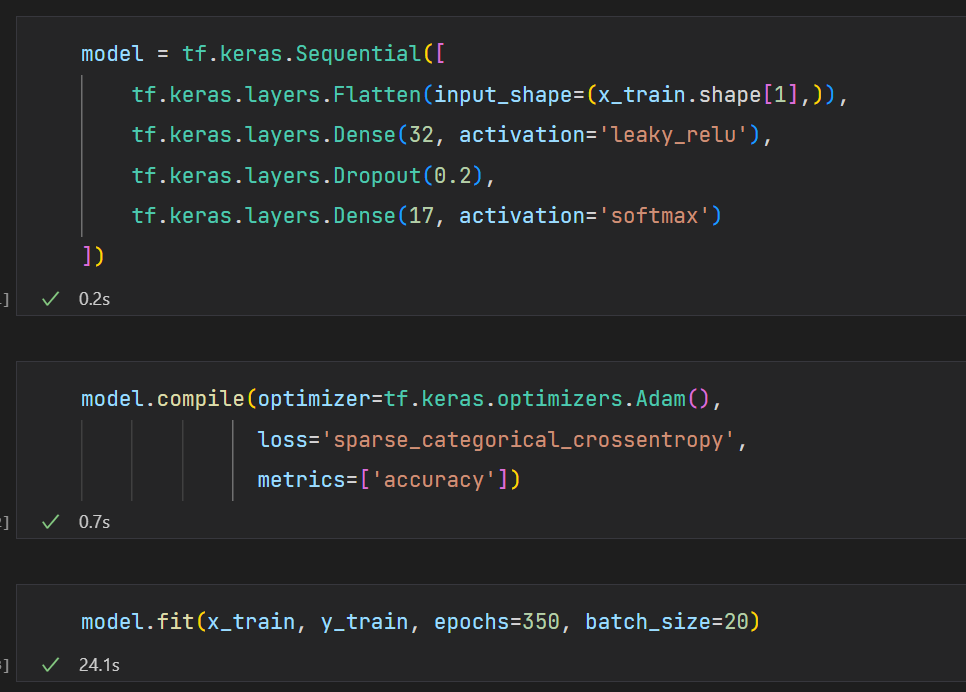
Menerima variable inputan dan mengalikannya dengan bobot (W).

Activation function:

Mengolah data inputan menggunakan model matematika yang tepat.

Satu perceptron dapat digunakan untuk klasifikasi, prediksi dan kategorisasi.

Pada studi kasus ini, menggunakan library tensorflow untuk melakukan train.



Model dilakukan sebanyak epochs 350 kali yang menghasilkan akurasi yang cukup tinggi pada data train.

Text

Description automatically generated

1. **Evaluasi hasil**

Kesimpulan diambil ketika data test digunakan pada model yang telah dilatih. Hasilnya adalah sebagai berikut.

Text

Description automatically generated with medium confidence

Memiliki akurasi sekitar 69% pada data test. Hasil akurasi pada data test ini memiliki nilai yang lumayan baik sehingga dimulai dari pra-pemrosesan data sudah dilakukan dengan baik, data dilatih dan menghasilkan akurasi data test sekitar 60-70%.

1. **Presentasi video**

[**https://youtu.be/UOLSp6DpuyY**](https://youtu.be/UOLSp6DpuyY)

[**https://github.com/Hilmantm/tugas-machine-learning-case-based-1**](https://github.com/Hilmantm/tugas-machine-learning-case-based-1)