**Studi Algoritma Klasifikasi Untuk Mengenal Pola *Activity Daily Living* (ADL) Menggunakan K-Nearest Neighbor Pada Orang Dewasa Sehat**

**Manual Book**

****

**Disusun oleh :**

**Askawati (1301144215)**

**Program Studi Sarjana S1 Teknik Informatika**

**Fakultas Informatika**

**Universitas Telkom**

**Bandung**

**2018**

DAFTAR ISI

[**Daftar Gambar** 4](#_Toc523310719)

[**1.** **Data** 5](#_Toc523310720)

[**1.1.** **Data Training** 5](#_Toc523310721)

[**1.2.** **Data Pengujian** 6](#_Toc523310722)

[**2.** **Program** 7](#_Toc523310723)

[**2.1.** **Program Arduino** 7](#_Toc523310724)

[**2.2.** **Program Pengolahan Data** 10](#_Toc523310725)

# **Daftar Gambar**

[Gambar 1 Data sensor aktivitas duduk subjek 1 5](#_Toc523307972)

[Gambar 2 Data hasil ekstraksi ciri 6](#_Toc523307973)

[Gambar 3 Data pengujian aktvitas berjalan subjek 1 7](#_Toc523307974)

[Gambar 4 Tampilan awal arduino 7](#_Toc523307975)

[Gambar 5 Pemilihan board 8](#_Toc523307976)

[Gambar 6 Pemilihan flash frequency dan uploud speed 8](#_Toc523307977)

[Gambar 7 Pemilihan port 9](#_Toc523307978)

[Gambar 8 Konfigurasi server dan hotspot 9](#_Toc523307979)

[Gambar 9 Serial monitor 10](#_Toc523307980)

[Gambar 10 Tampilan awal pycharm edu 2018 10](#_Toc523307981)

[Gambar 11 Menu settings 11](#_Toc523307982)

[Gambar 12 Install packages 11](#_Toc523307983)

[Gambar 13 Tampilan awal program file ekstraksi ciri 12](#_Toc523307984)

[Gambar 14 Membuka file yang di ekstraksi 12](#_Toc523307985)

[Gambar 15 Ubah label aktivitas pada ekstraksi ciri 13](#_Toc523307986)

[Gambar 16 Simpan hasil ekstraksi 13](#_Toc523307987)

[Gambar 17 Tampilan program file knn.py 13](#_Toc523307988)

[Gambar 18 Membuka file dataset 14](#_Toc523307989)

[Gambar 19 Pembagian data trainin dan testing 14](#_Toc523307990)

[Gambar 20 pemilihan parameter k dan metrik 14](#_Toc523307991)

[Gambar 21 Menjalankan program file knn.py 15](#_Toc523307992)

[Gambar 22 Hasil menjalankan program file knn.py 15](#_Toc523307993)

[Gambar 23 Tampilan awal program file Testing.py 16](#_Toc523307994)

[Gambar 24 Membuka file prediksi 16](#_Toc523307995)

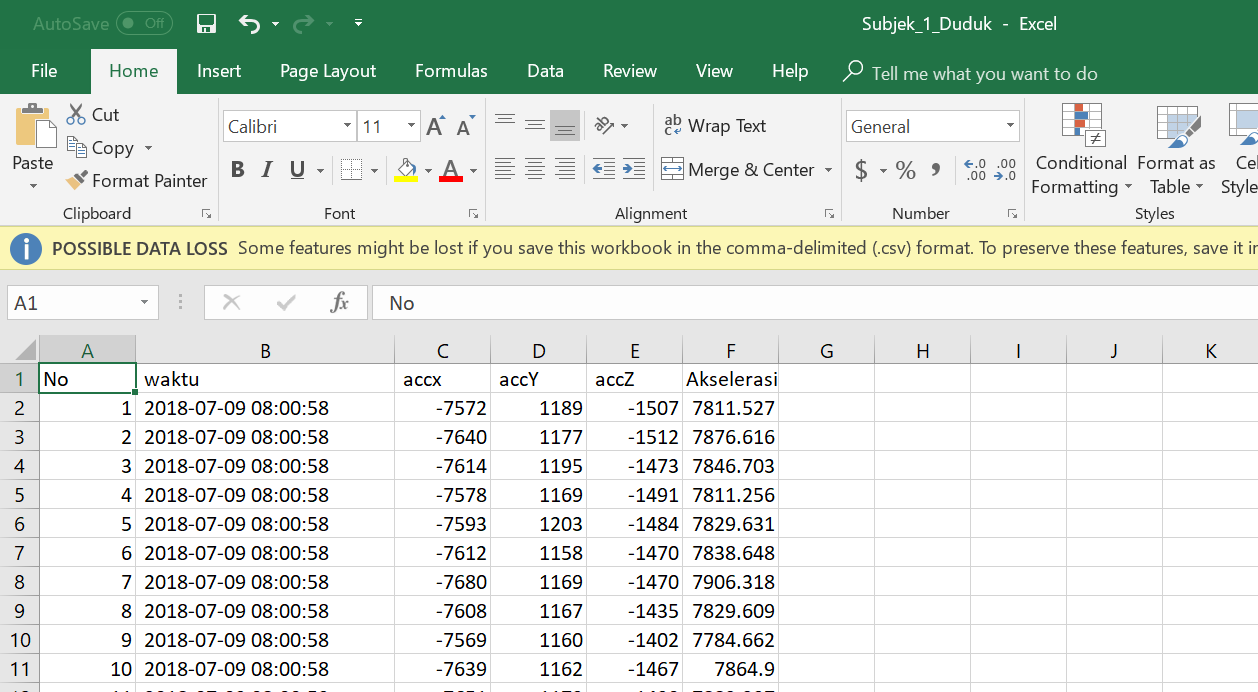
[Gambar 25 Hasil menjalankan program file Testing.py 16](#_Toc523307996)

# **Data**

Data diperoleh berdasarkan hasil percobaan menggunakan perangkat sensor MPU605O dan mikrokontroler ESP32. Perangkat sensor diletakkan pada pinggang pengguna saat percobaan, percobaan menggunakan 30 subjek dengan aktivitas seperti :

* Aktivitas berjalan
* Aktivitas berdiri
* Aktivitas duduk
* Aktivitas turun tangga
* Aktivitas naik tangga

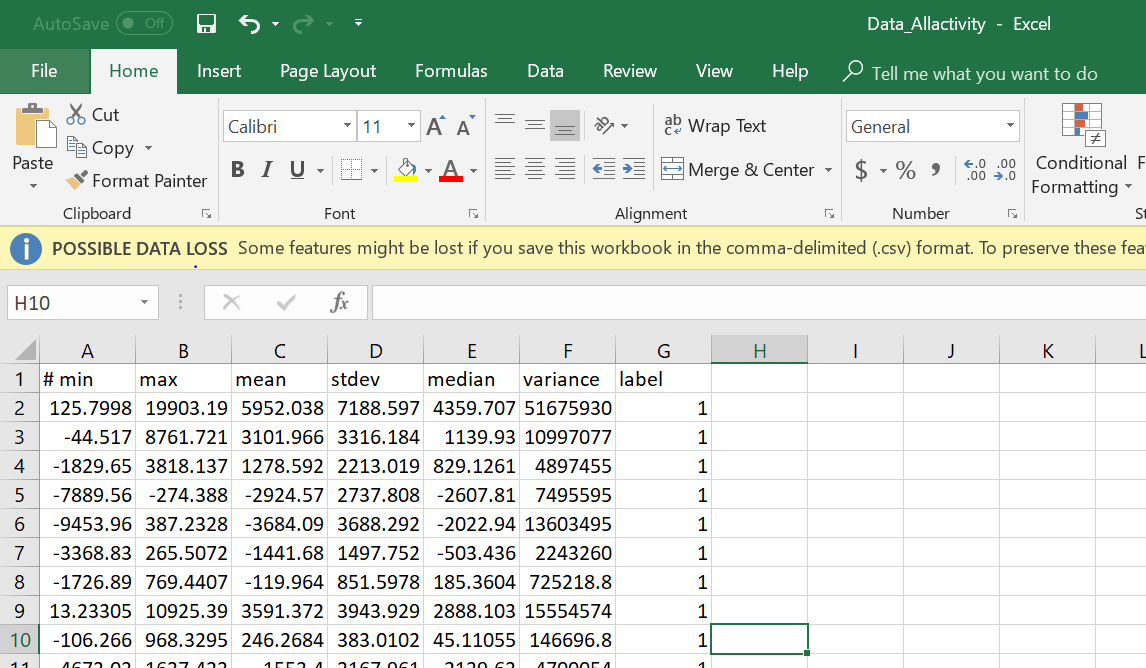
Data yang dihasilkan sensor berupa banyak data, waktu, akselerometer dengan sumbu x, sumbu y dan sumbu z. Data pada folder data sensor terdiri dari device/banyaknya data, waktu, accx, accy, accz, dan akselerasi. Berikut gambar data yang dihasilkan.



Gambar 1 Data sensor aktivitas duduk subjek 1

## **Data Training**

Data *training* diperoleh berdasarkan hasil ekstraksi ciri menggunakan file Ekstraksi ciri.py pada setiap aktivitas. Data pada folder data *training* terdiri dari data *minimum, maksimum,* *mean,* standar deviasi*, median, variance*, dan label.



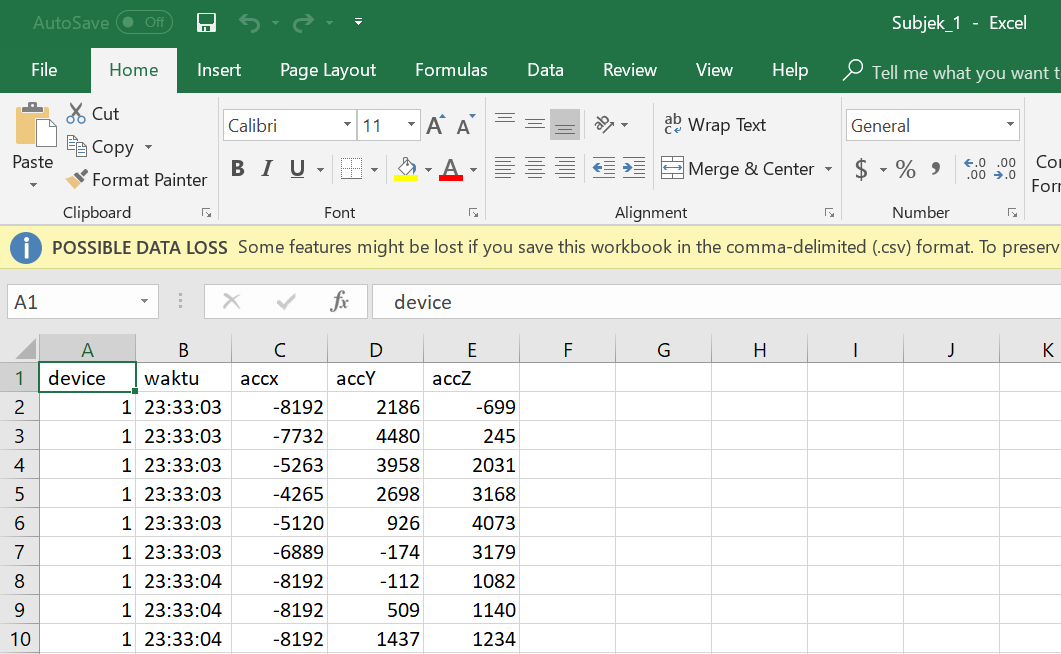
Gambar 2 Data hasil ekstraksi ciri

Berdasarkan gambar 2. Pada setiap aktivitas memiliki label yang berbeda-beda. Berikut penjelasan file pada folder data *training.*

* Data\_Berjalan\_1 (label 1)
* Data\_Berdiri\_2 (label 2)
* Data\_NaikTangga\_3 (label 3)
* Data\_TurunTangga\_4 (label 4)
* Data\_Duduk\_5 (label 5)
* Data\_Allactivity, Kumpulan seluruh data aktivitas beserta label.

## **Data Pengujian**

Data pengujian terdiri dari 10 subjek dengan 5 aktivitas yang sama. Data pada folder Data Pengujian terdiri dari device atau banyaknya data, waktu, accx,accy, dan accz. Berikut gambar data yang dihasilkan.



Gambar 3 Data pengujian aktvitas berjalan subjek 1

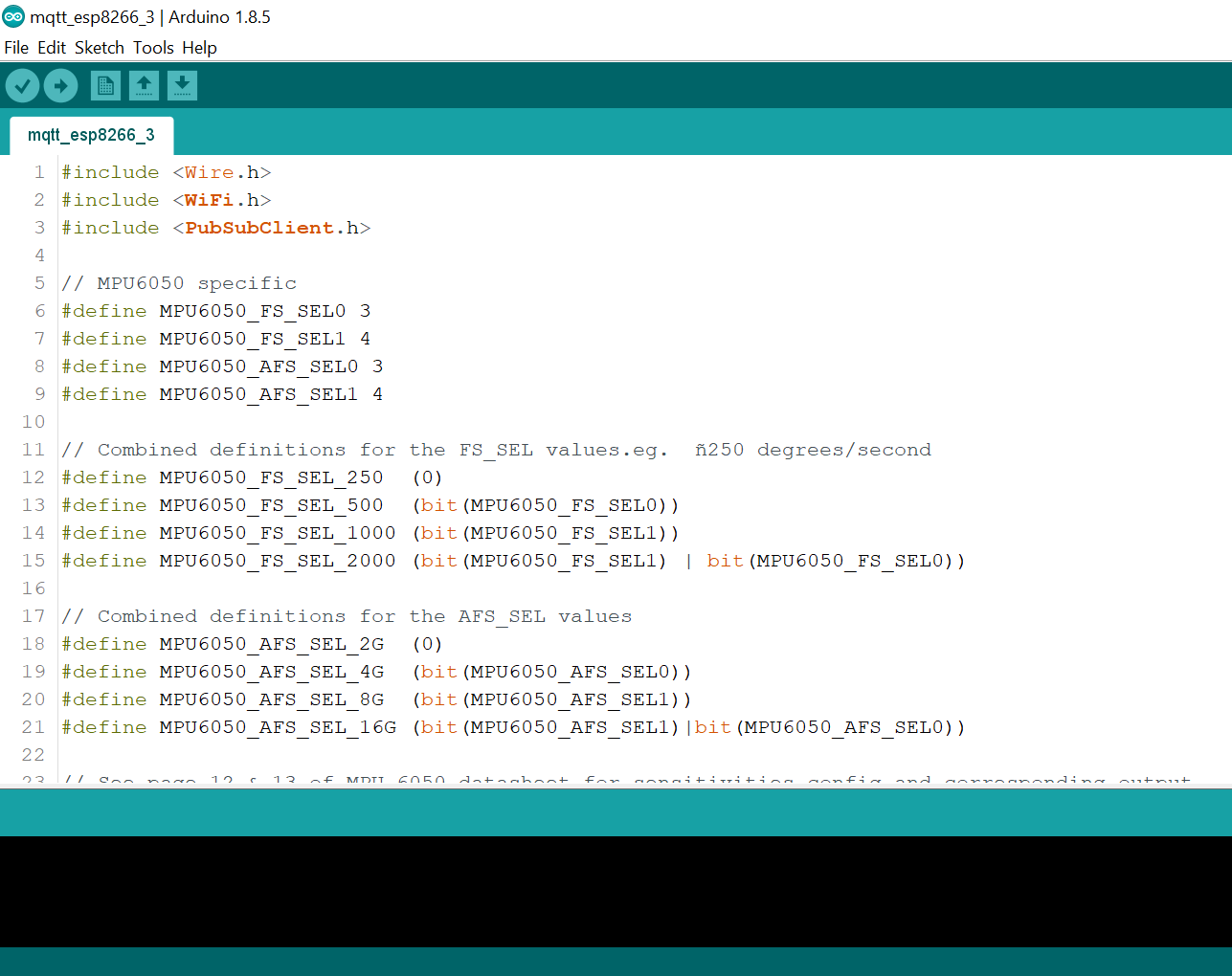
# **Program**

Program pada penelitian ini terdiri dari dua program yaitu program Arduino untuk perangkat yang digunakan dan program pengolahan data menggunakan Bahasa pemrograman python.

## **Program Arduino**

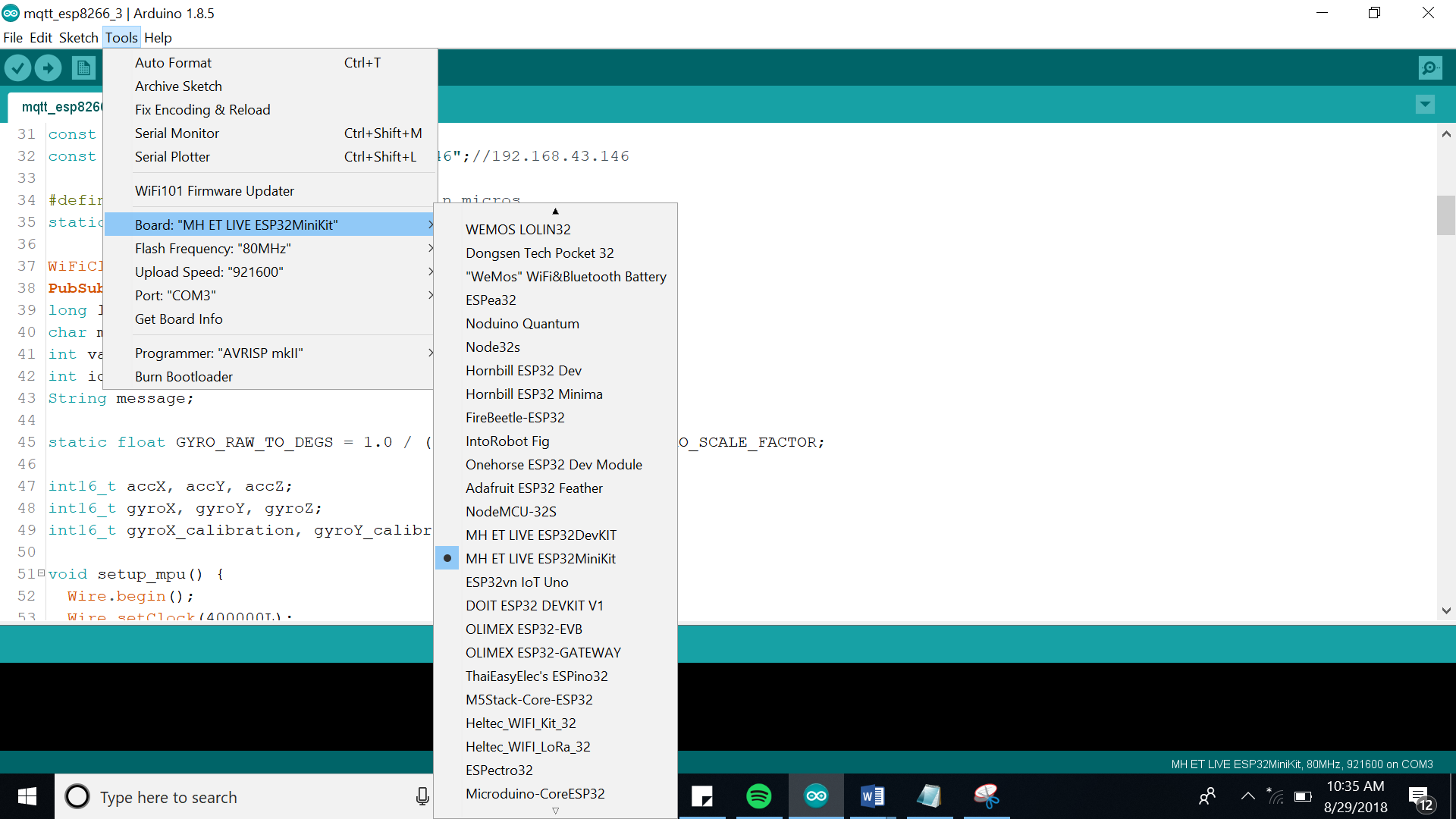
Berikut langkat-langkah menjalankan program arduino.

1. Buka arduino dan buka file mqtt\_esp8266\_3



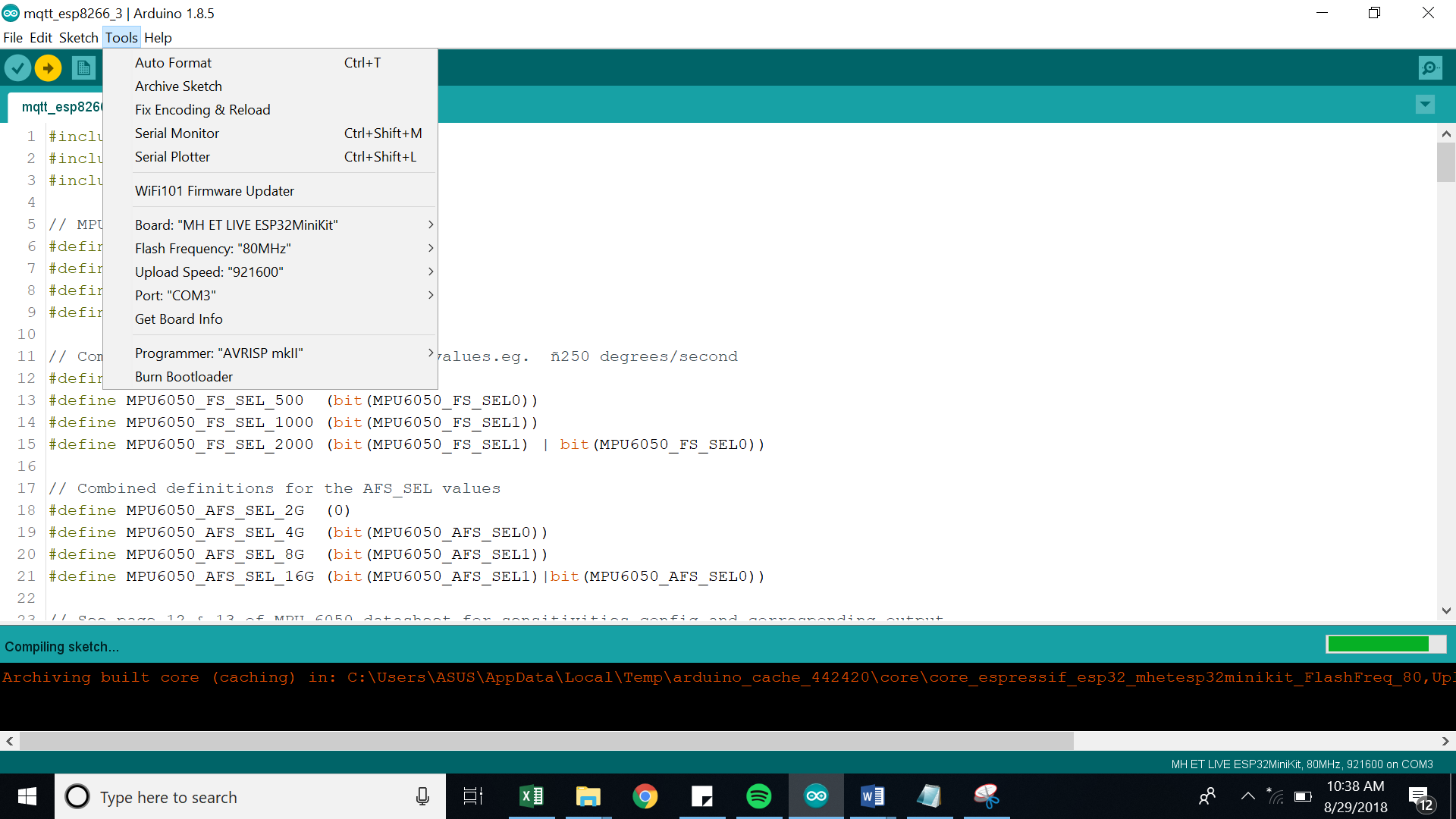
Gambar 4 Tampilan awal arduino

1. Pilih *board* yang digunakan. Pilih tools dan pilih board MH ET LIBE ESP32MiniKit.



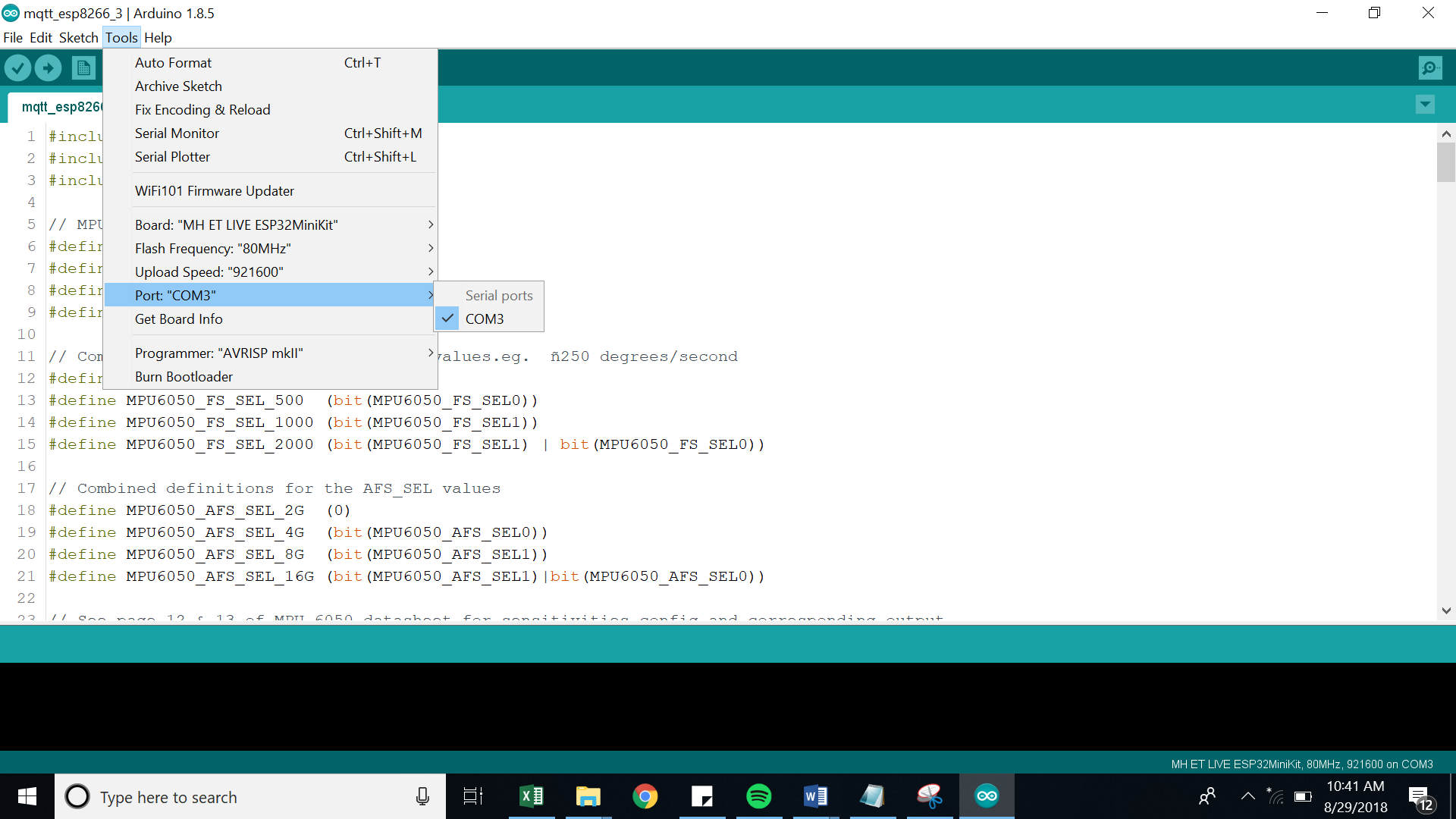
Gambar 5 Pemilihan board

1. Pada menu tools, pilih flash frequency 80MHz dan uploud speed 921600.



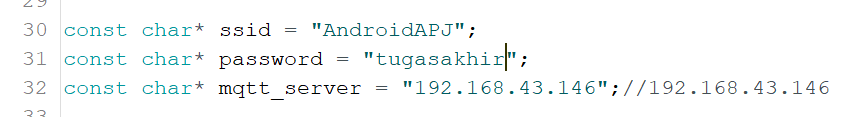
Gambar 6 Pemilihan flash frequency dan uploud speed

1. Setelah itu hubungkan perangkat dan pilih port yang digunakan. Setiap perangkat memiliki port yang berbeda-beda.



Gambar 7 Pemilihan port

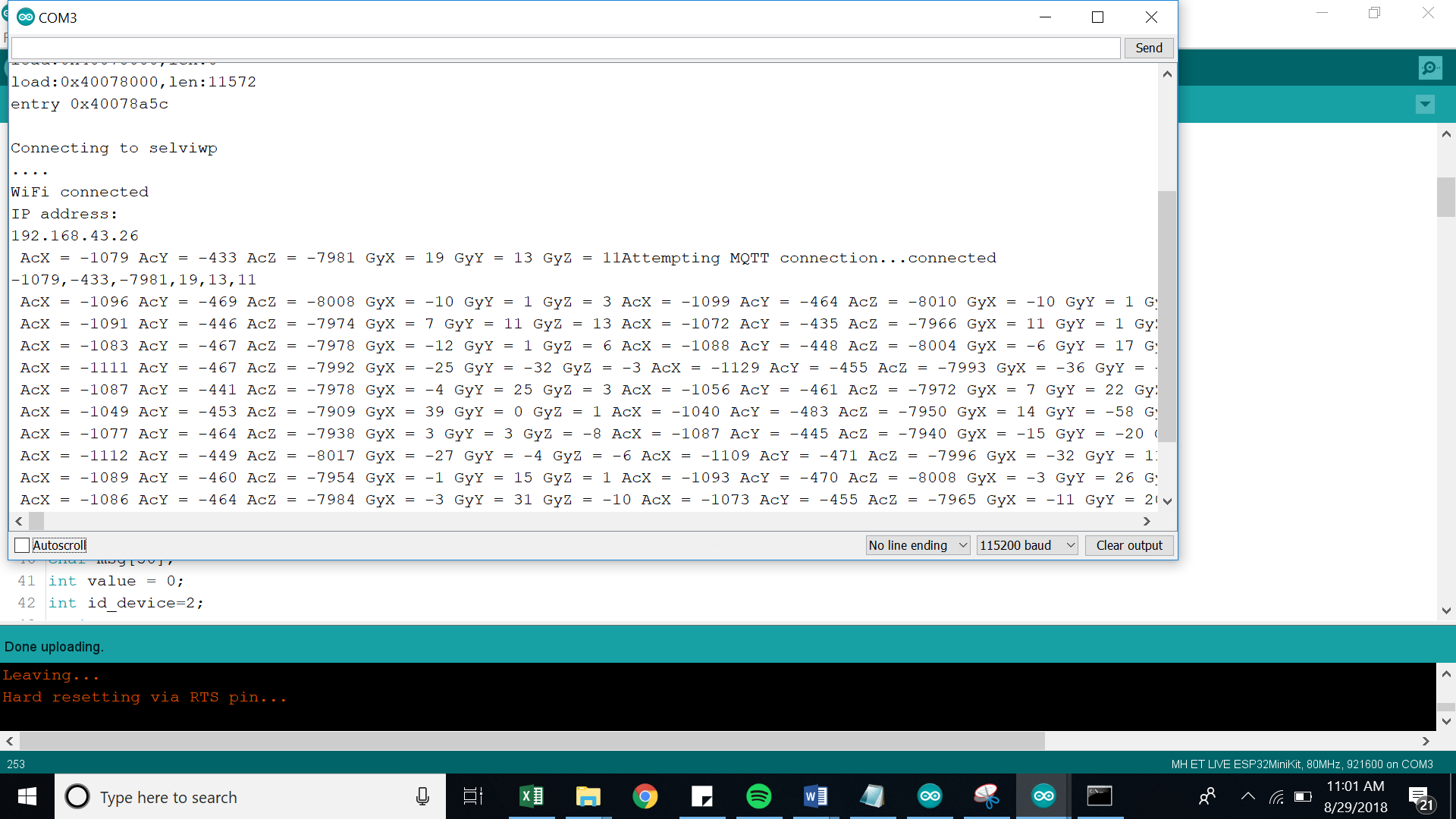
1. Ubah koneksi yang digunakan sehingga data dapat melakukan uploud.



Gambar 8 Konfigurasi server dan hotspot

Berdasarkan gambar 8. Ssid merupakan nama hotspot yang digunakan, password merupakan password dari hotspot yang digunakan dan mqtt\_server merupakan alamat IP dari komputer yang akan menjadi server.

1. Jika sudah di konfigurasi selanjutnya uploud file tersebut dan buka serial monitor.

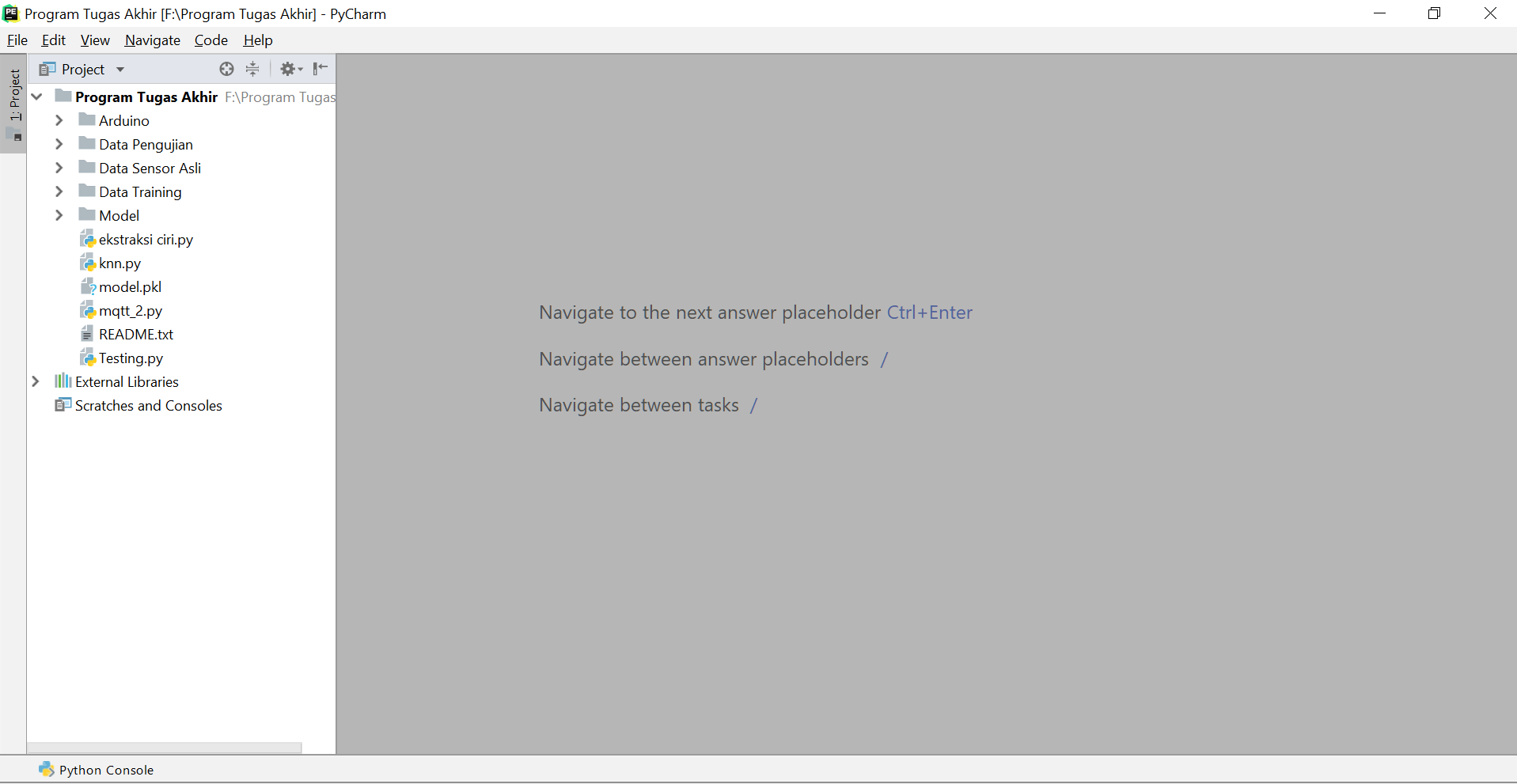


Gambar 9 Serial monitor

## **Program Pengolahan Data**

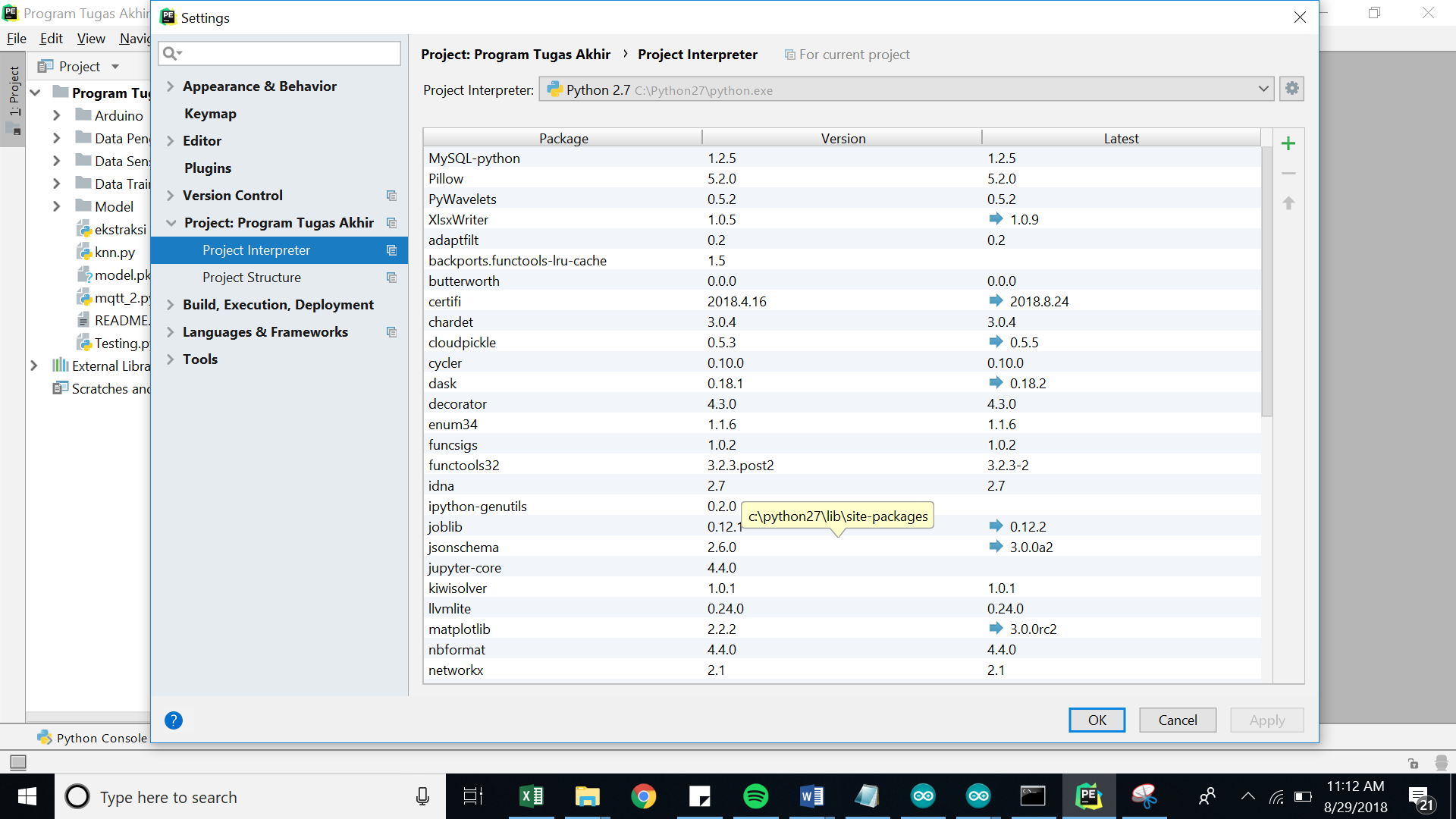
Program ini menggunakan python 2.7 dengan software pycharm edu 2108. Berikut langkah-langkah menjalankan program menggunakan pycharm edu 2018 :

1. Buka pycharm edu 2018, dan buka folder program tugas akhir.



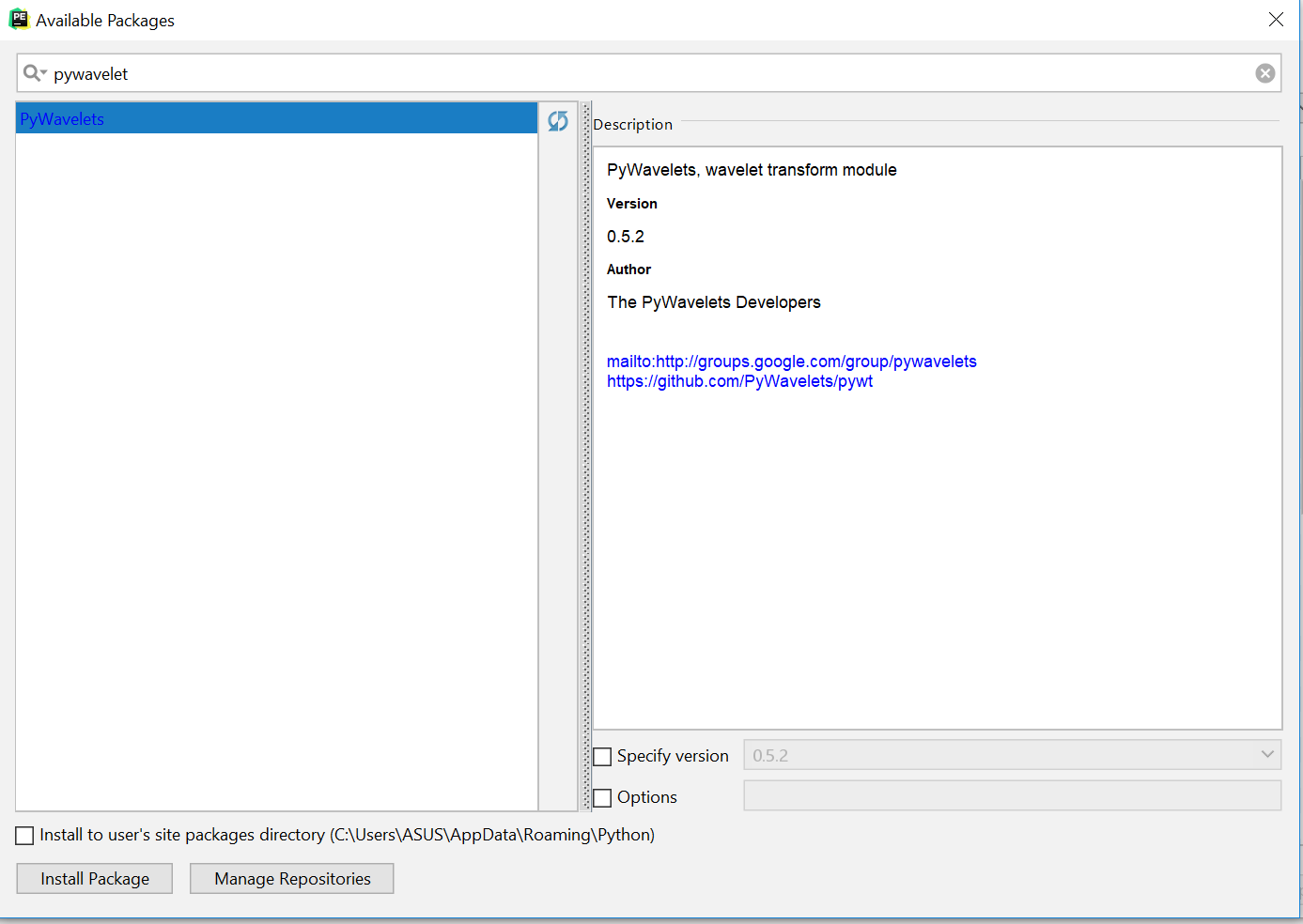
Gambar 10 Tampilan awal pycharm edu 2018

1. Selanjutnya sebelum membuka dan menjalankan program terdapat konfigurasi pada menu file pilih settings.



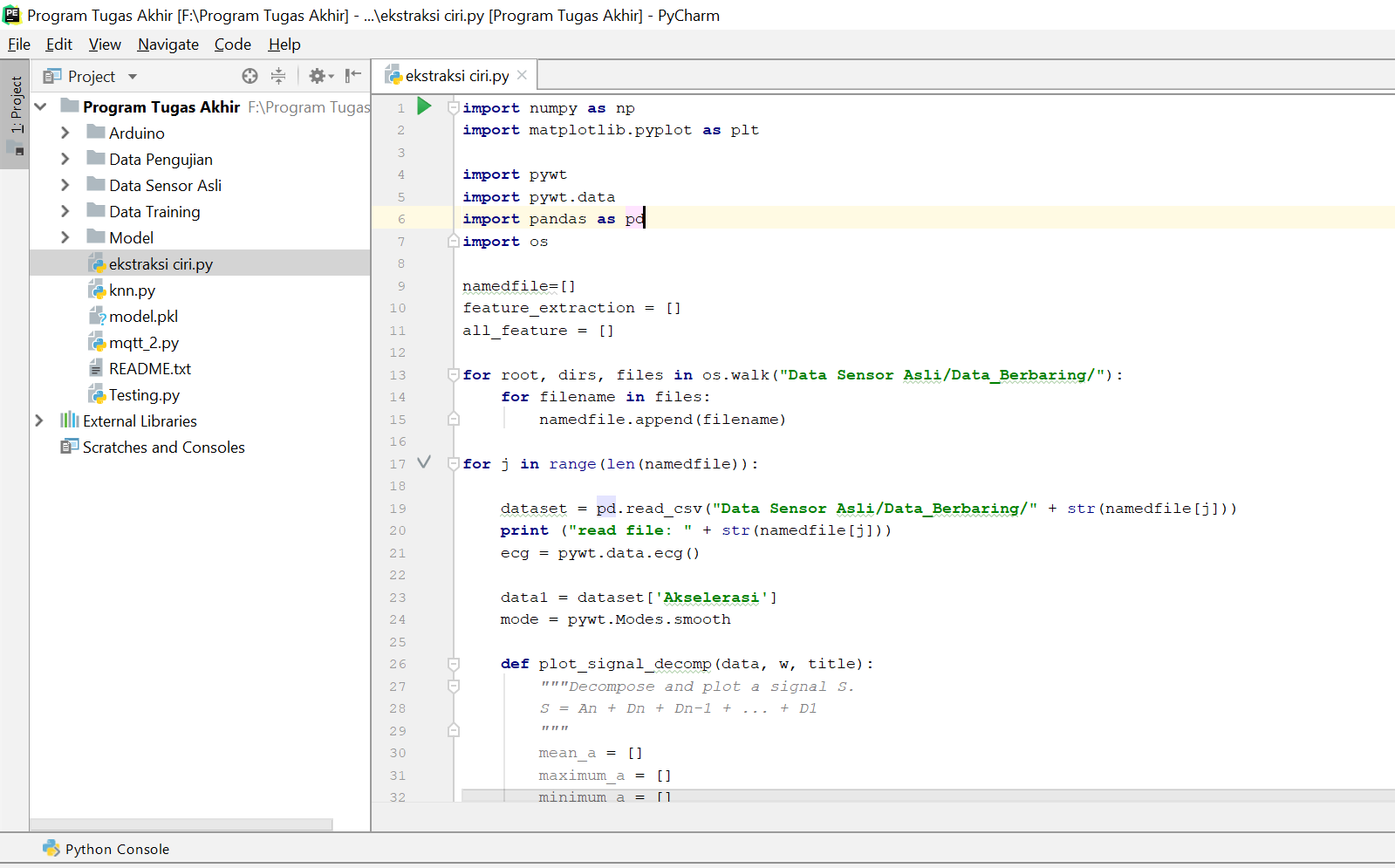
Gambar 11 Menu settings

Setelah ada tampilan seperti pada gambar 10. Pilih menu Project:Program Tugas Akhir. Lalu pilih Project Interpreter. Ubah project interpreter menjadi python 2.7 atau versi pyton lain yang dapat digunakan. Dalam tugas akhir ini digunakan python 2.7. Selanjutntya, install package-package yang diperlukan. Package yang digunakan pada program ini yaitu PyWavelet, joblib, matplotlib, seaborn, sklearn, pandas, numpy.



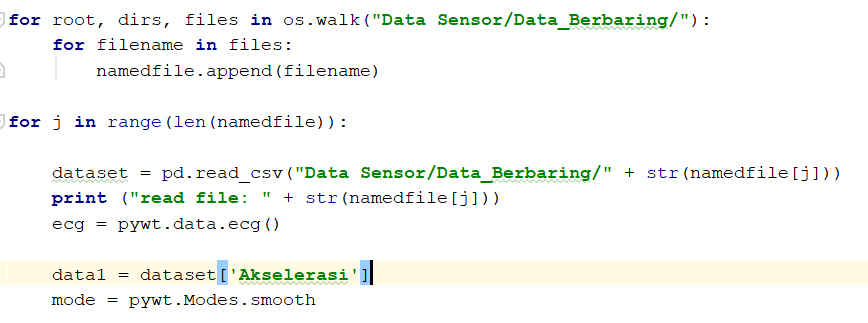
Gambar 12 Install packages

1. Setelah semua package telah terinstall. Buka program file Ekstraksi ciri.py.



Gambar 13 Tampilan awal program file ekstraksi ciri

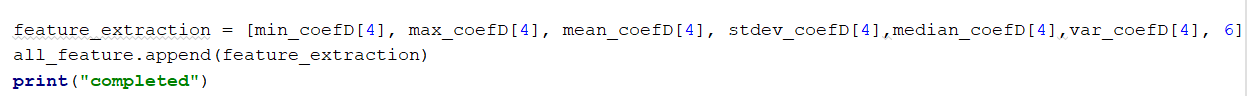
1. Ubah file yang akan di ekstraksi.



Gambar 14 Membuka file yang di ekstraksi

File yang digunakan untuk melakukan ekstraksi fitur yaitu pada folder data sensor. Masukkan semua aktivitas yang ada sehingga dapat diperoleh ciri dari setiap aktivitas yang digunakan. Program akan membaca data pada kolom “Akselerasi” dan menghasilkan ciri data dari aktivitas tersebut seperti data *minimum, maksimum,* *mean,* standar deviasi*, median, variance*.

1. Ubah label pada setiap aktivitas.

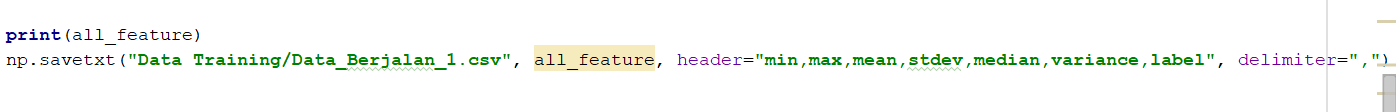


Gambar 15 Ubah label aktivitas pada ekstraksi ciri

Berikut adalah daftar label yang digunakan pada setiap aktivitas :

* Data\_Berjalan = label 1
* Data\_Berdiri = label 2
* Data\_NaikTangga = label 3
* Data\_TurunTangga = label 4
* Data\_Duduk = label 5

1. Jika telah mengubah label dari aktivitas yang akan di esktraksi. Selanjutnya simpan data hasil ekstraksi ciri dengan format .csv seperti pada gambar 16.

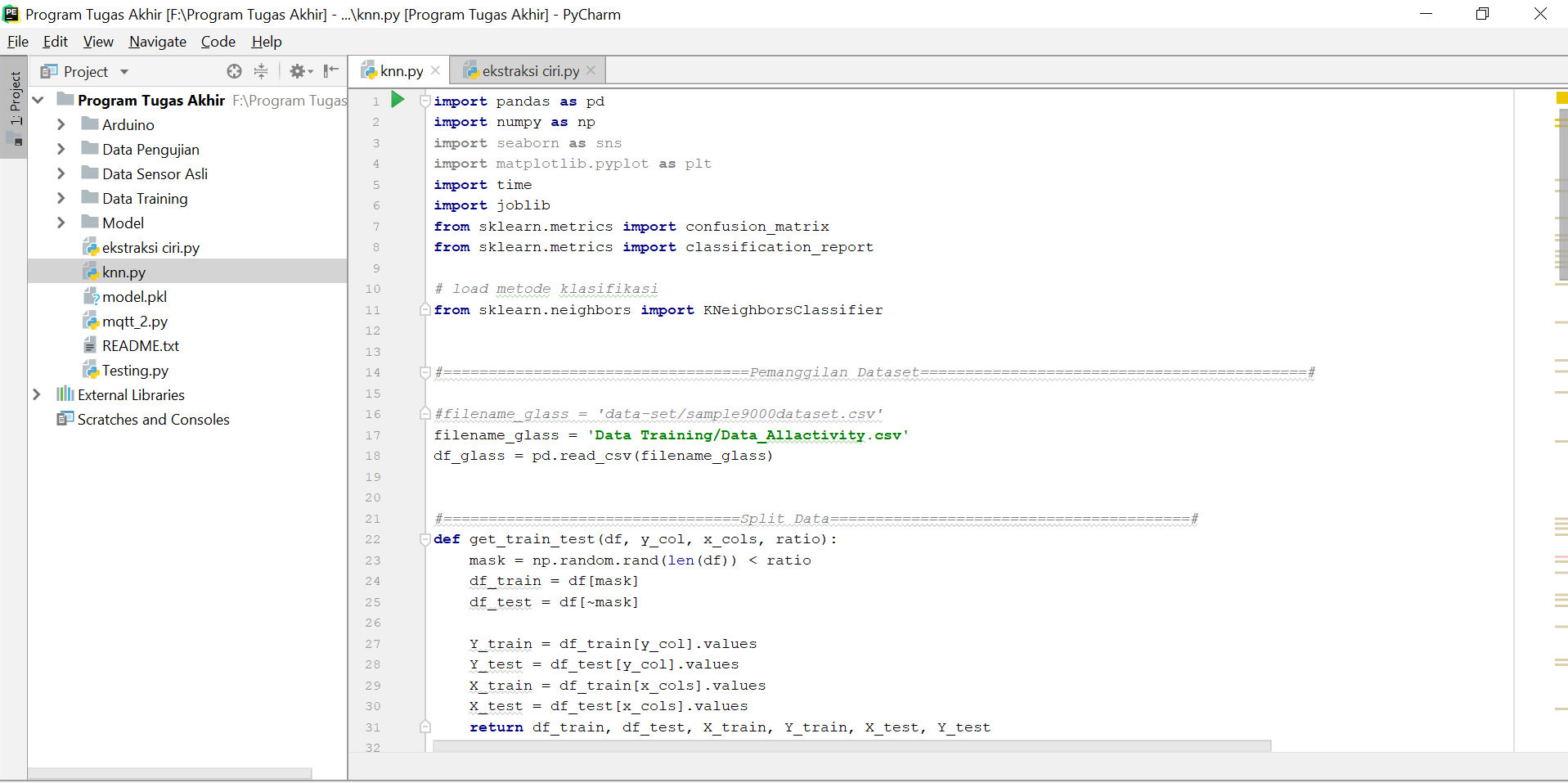


Gambar 16 Simpan hasil ekstraksi

Catatan :

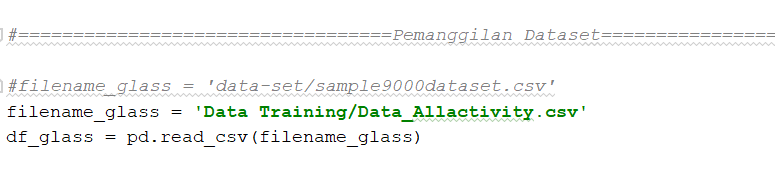
Run semua file pada folder data sensor sehingga mendapatkan ciri dan label data setiap aktiitas.

1. Buka program file knn.py.



Gambar 17 Tampilan program file knn.py

1. Buka file yang akan digunakan untuk mencari akurasi, sensitifitas, dan spesifisitas seperti pada gambar 18.

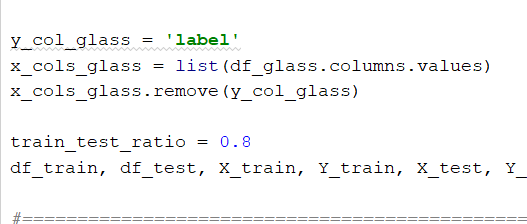


Gambar 18 Membuka file dataset

Catatan :

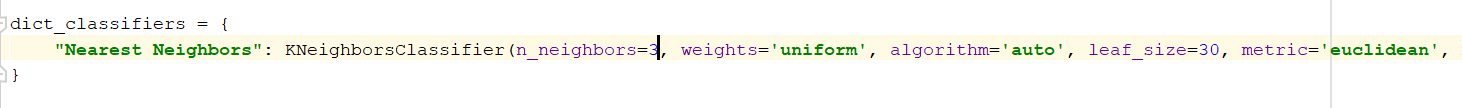
File Data\_Allactivity.csv merupakan data hasil pengabungan dari aktivitas-aktivitas yang digunakan beserta label pada setiap aktivitas.

1. Jika diinginkan ubah train\_test\_ratio untuk pembagian data *training* dan *testing* pada. Pada tugas akhir ini menggunakan pembagian data 0.6, 0.7 dan 0.8. Pada gambar 19. Train\_test\_ratio yang digunakan yaitu 0.8 yang artinya data training 0.8 dan sisanya adalah data *testing.*



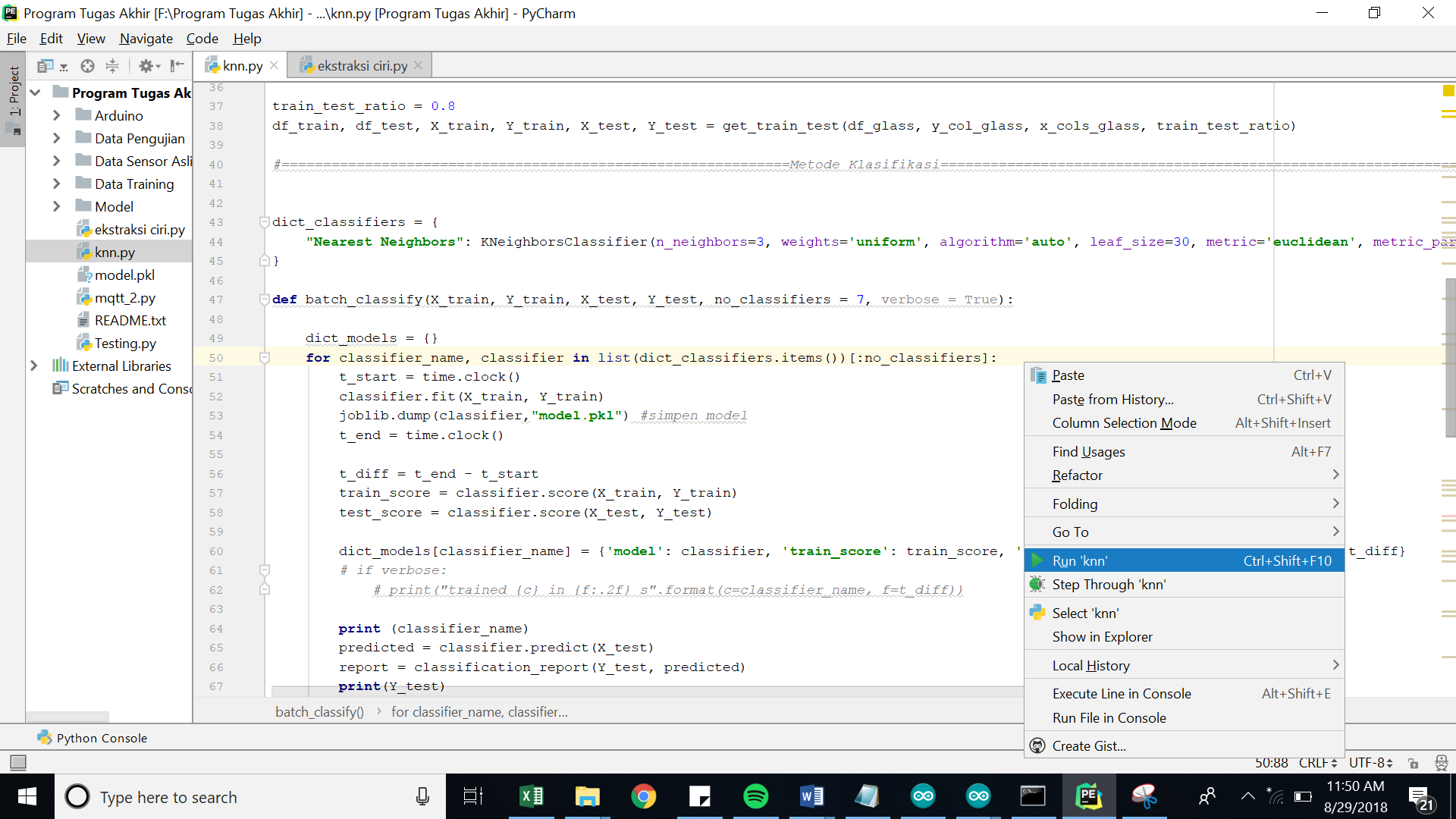
Gambar 19 Pembagian data trainin dan testing

1. pada dict\_classifier, ubah n\_neighbors yang merupakan parameter k . Contoh n\_neighbors=3. Ubah metric atau metode pengukuran jarak yang digunakan. Contoh: (Metric='euclidean') .

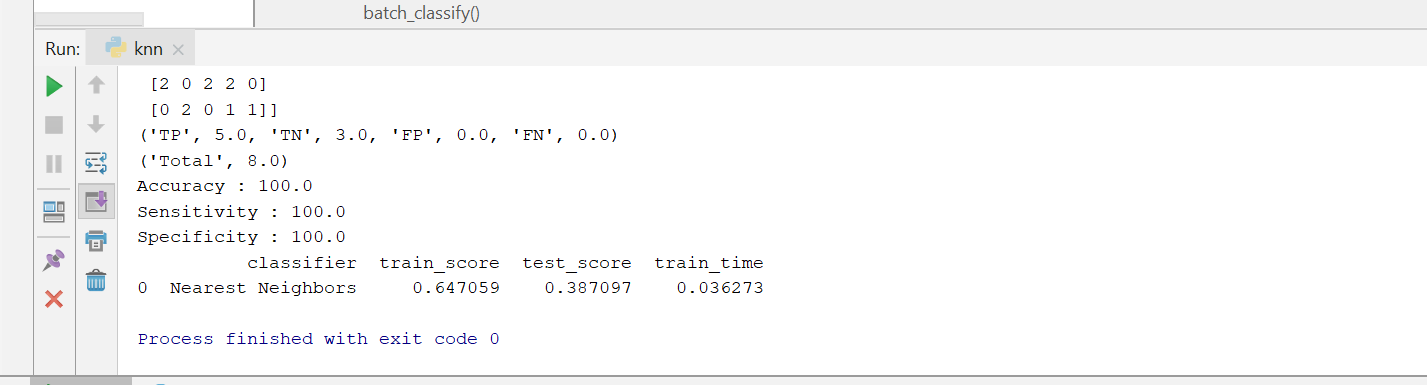


Gambar 20 pemilihan parameter k dan metrik

1. Jika semua telah diubah sesuai yang diinginkan. Jalankan program seperti pada gambar 21.



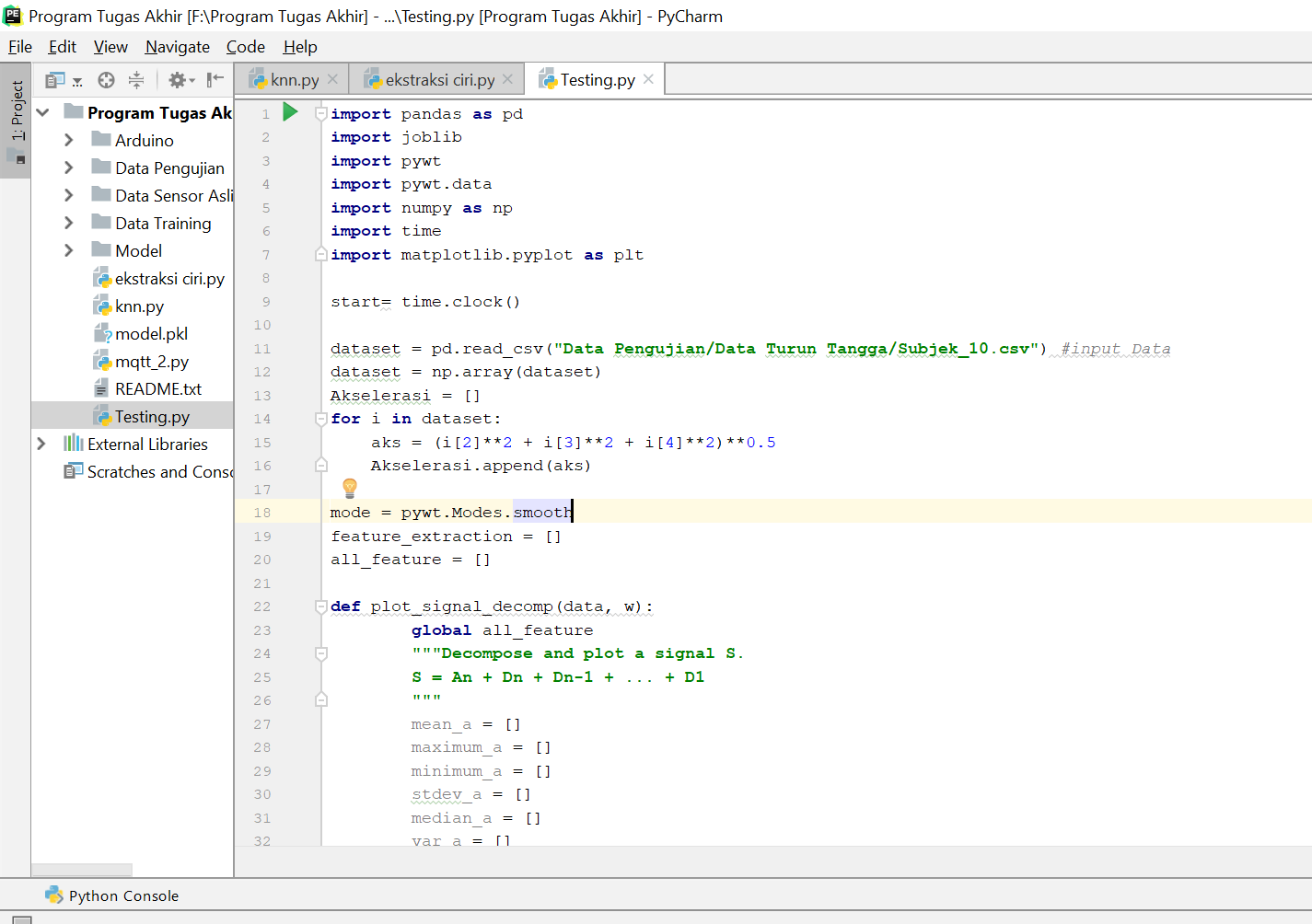
Gambar 21 Menjalankan program file knn.py



Gambar 22 Hasil menjalankan program file knn.py

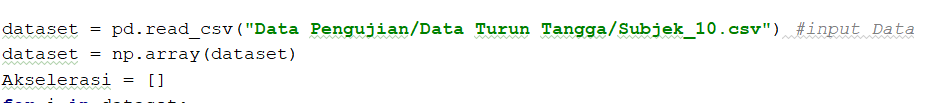
Gambar 21. Akan dilakukan sebanyak 10 kali untuk mendapatkan hasil rata-rata dari setiap aktivitas berdasarkan parameter k dan metode yang digunakan. Hasil rata-rata terbaik yang dihasilkan akan menjadi inputan dalam melakukan prediksi yang akan dilakukan pada program file testing.py.

1. Buka program file Testing.py



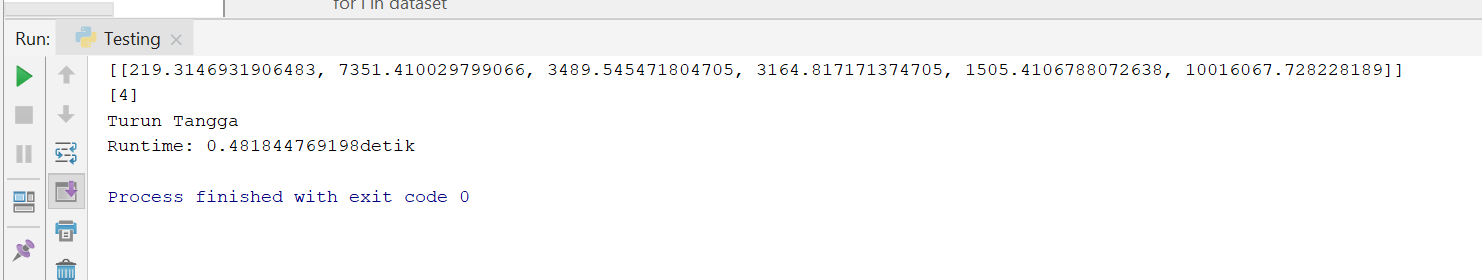
Gambar 23 Tampilan awal program file Testing.py

1. Pilih data untuk melakukan prediksi aktivitas pada folder data testing seperti pada gambar 24.



Gambar 24 Membuka file prediksi

1. Jalankan program file Testing.py untuk melihat hasil prediksi dari data testing yang diperoleh seperti pada gambar 25.



Gambar 25 Hasil menjalankan program file Testing.py