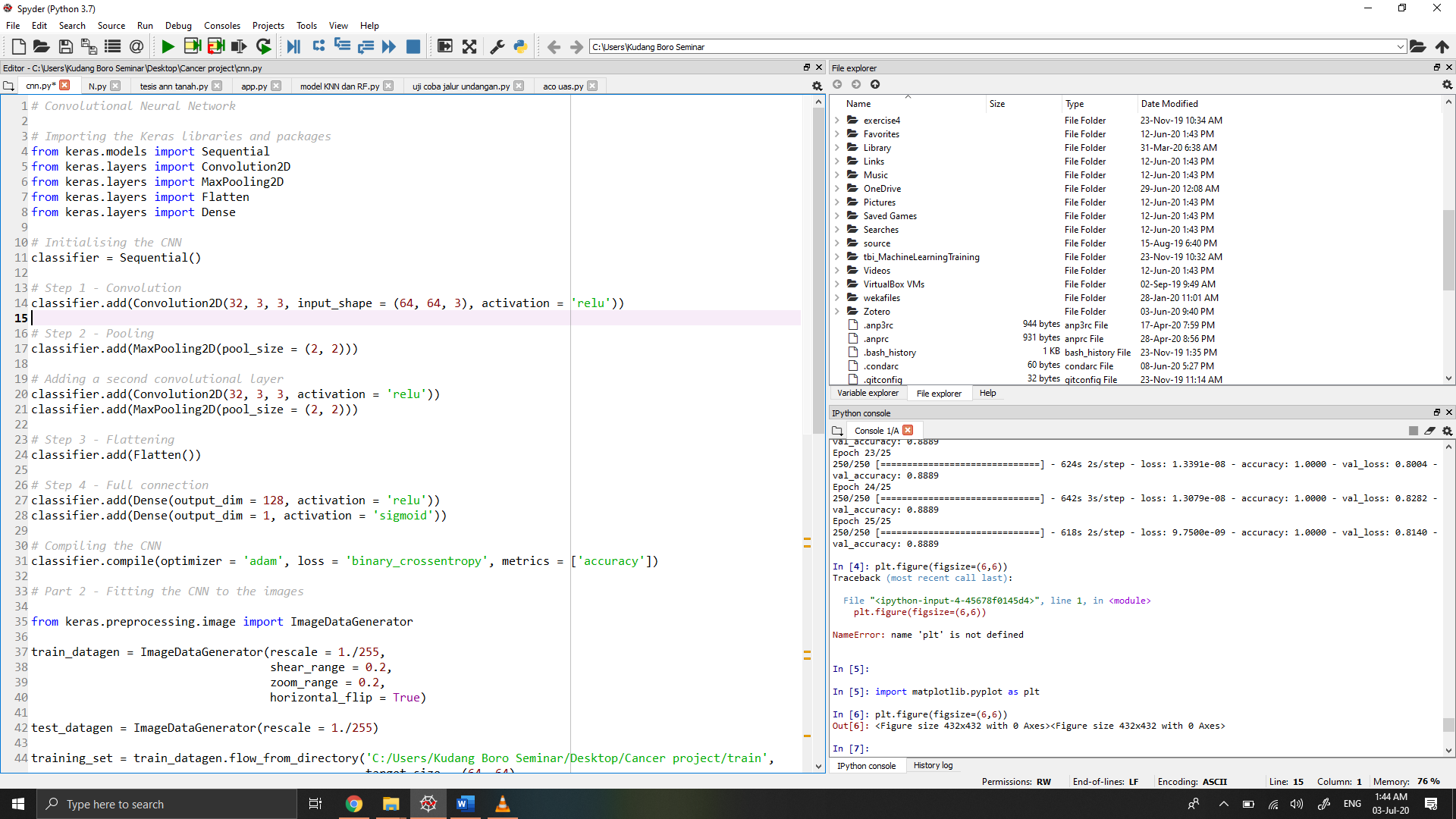
Step step CNN

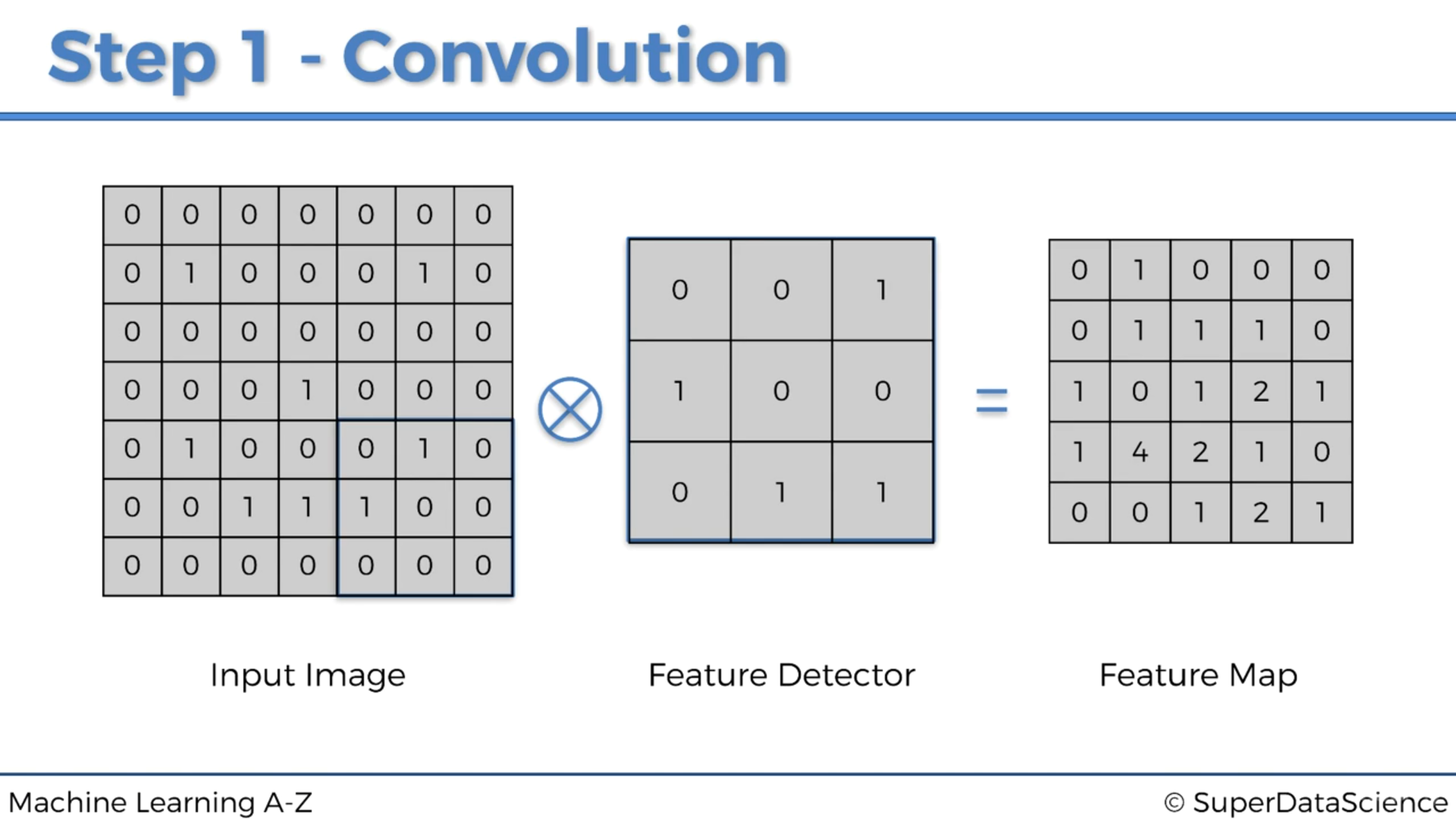


Untuk step pertama convolution



Itu coding pyhtonnya, pertama buat deklarasi pengen klasifikasi, saya kasih nama classifier yang nanti di panggil fungsi sequential dari keras, fungsinya jadi model classifier bisa ditambahin layer layer lain

Untuk step layer convolutionnya bisa di kodingan, yg digunakan covolution2d karena gambar yg dibahaskan 2d dan hasil dimensi imagenya juga 2d, bisa aja 3d tapi berlum pernah nyoba eheheh, bisa di liat dibawah

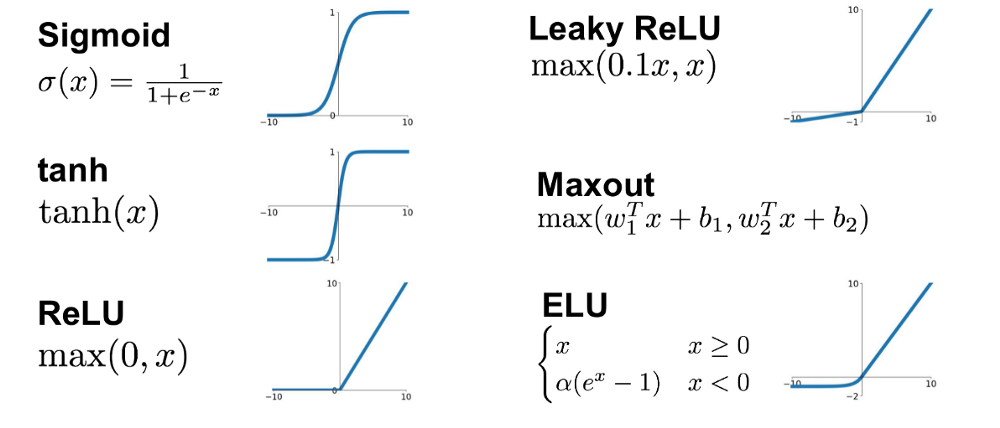


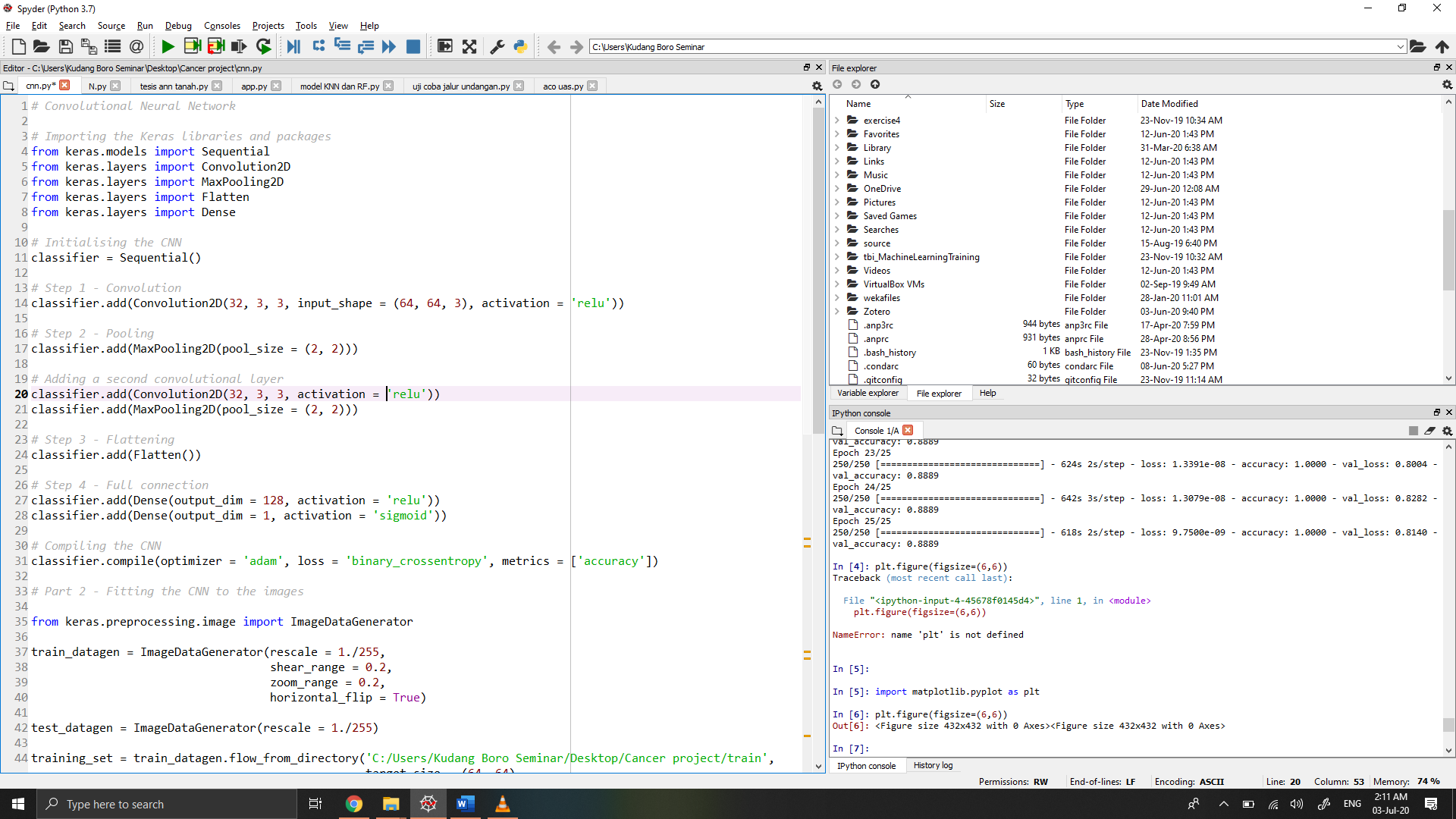
Abis di inisiasi convolution2dnya ada beberapa parameter yg harus ditentukan, yg saya pake 32, 3, 3, maksudnya untuk 32 itu number of filter atau jumlah filter yang dipakai, maksudnya filter disini adalah filter atau kernel (ada yang sebut juga neuron, fungsinya kalau saya cari cari hampir sama makin banyak jumlah kernel atau neuron yg digunakan makin bagus, “katanya” tapi untuk tau mana yg optimum kudu di hyperparameter tuning, tapi laptop saya yang ga kuad ehehe, defaultnya 64, tapi saya buat setengah saja, beberapa tutorial bilang ga terlalu masalah, untuk akurasi berpengaruh Cuma 1 sampai 2 persen dari yg say abaca beberapa) yang dipakai untuk convolution step ini, untuk 3,3 itu adalah kolom dan baris feature detector,

Setelah itu input shape 64,64,3 maksudnya 64,64 itu pixel yang bakal di input, biasanya kan pixel untuk gambar berwarna sampai 256, disini di reduksi gambar warnanya, sebenarnya bisa saja 256 Cuma berat sekali untuk prosesnya, kasihan laptop saya ahaha, dan untuk 3 itu untuk mendeklarasi kalua gambar input yang dipakai berwarna atau colored image

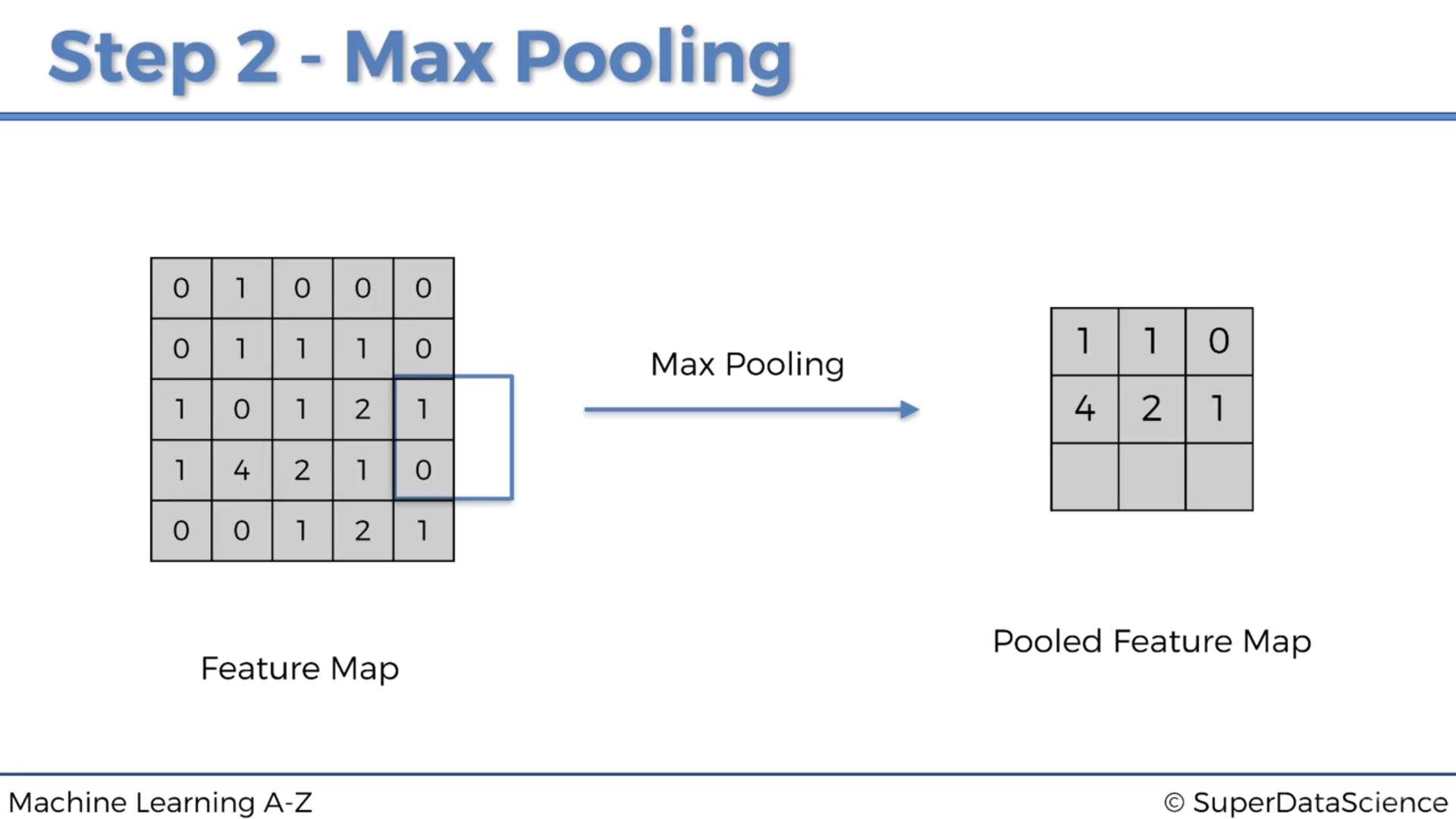


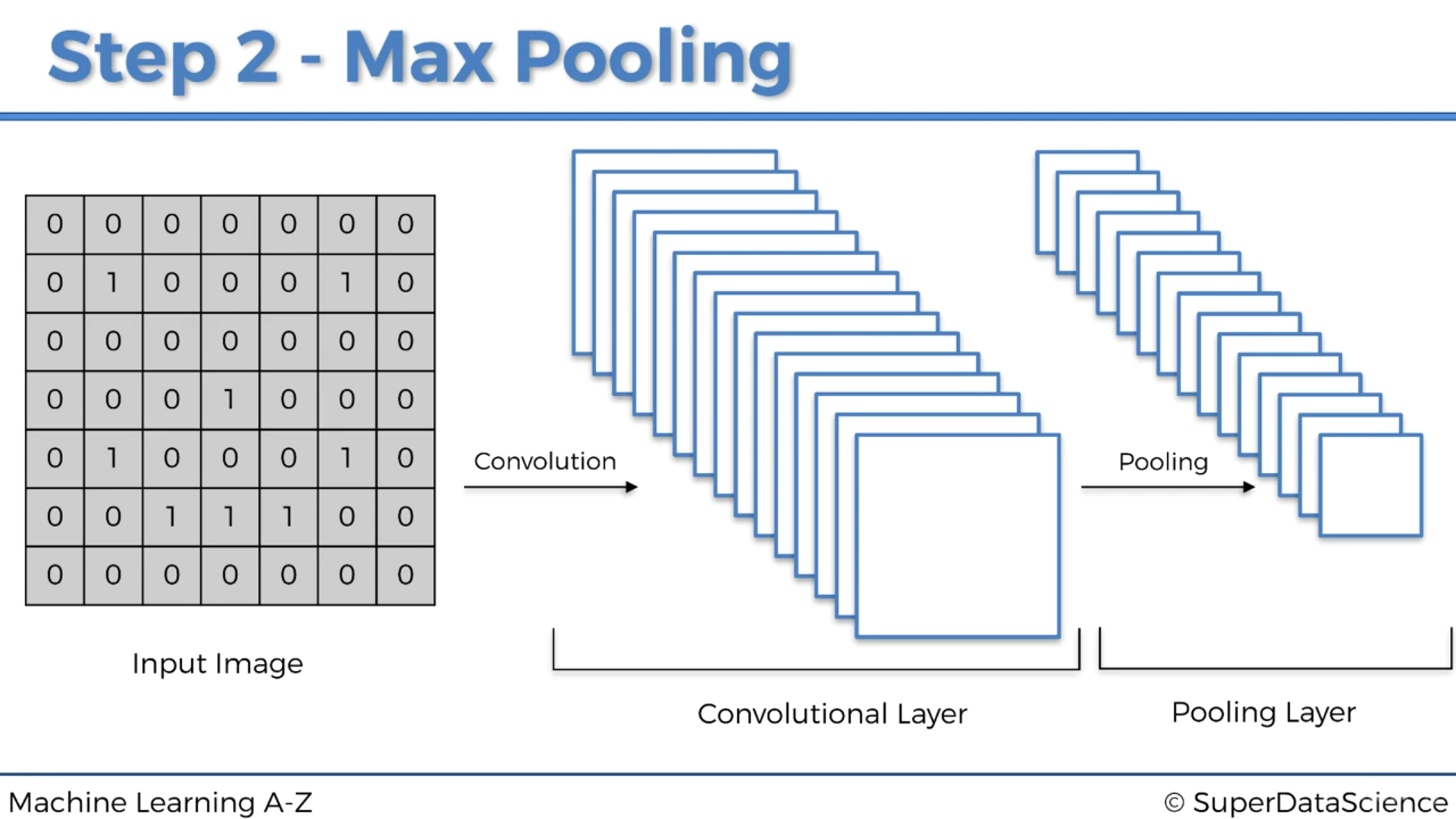
Nah untuk code activation relu, itu fungsi aktivasi yang dipakai, umumnya menggunakan relu, tapi fungsi aktivasi ada banyak, untuk ini saya males coba coba sih, soalnya bisa aja lebih tinggi akurasinya, tapi gitu, kondisi laptop yang tidak memungkinkan

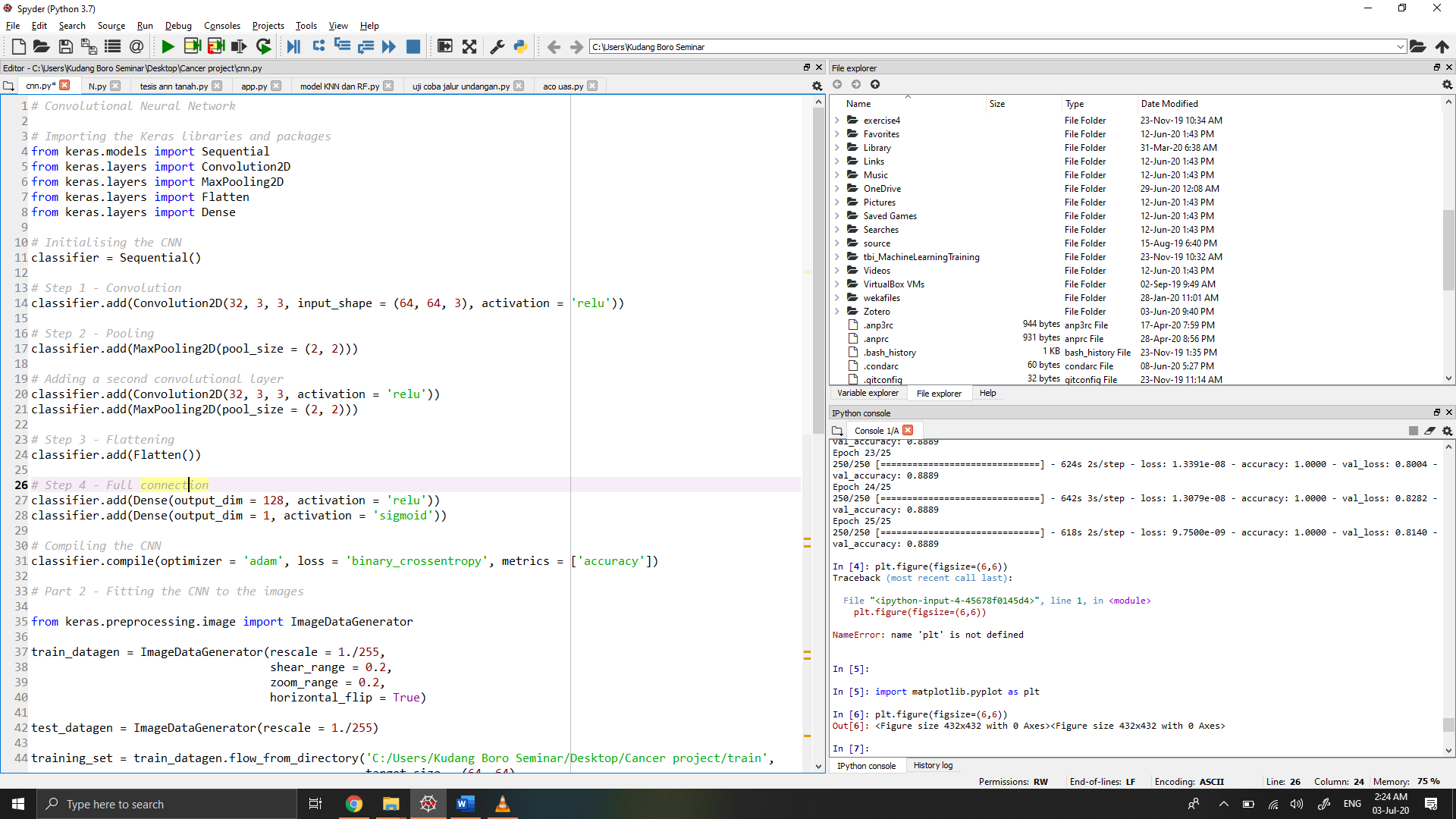




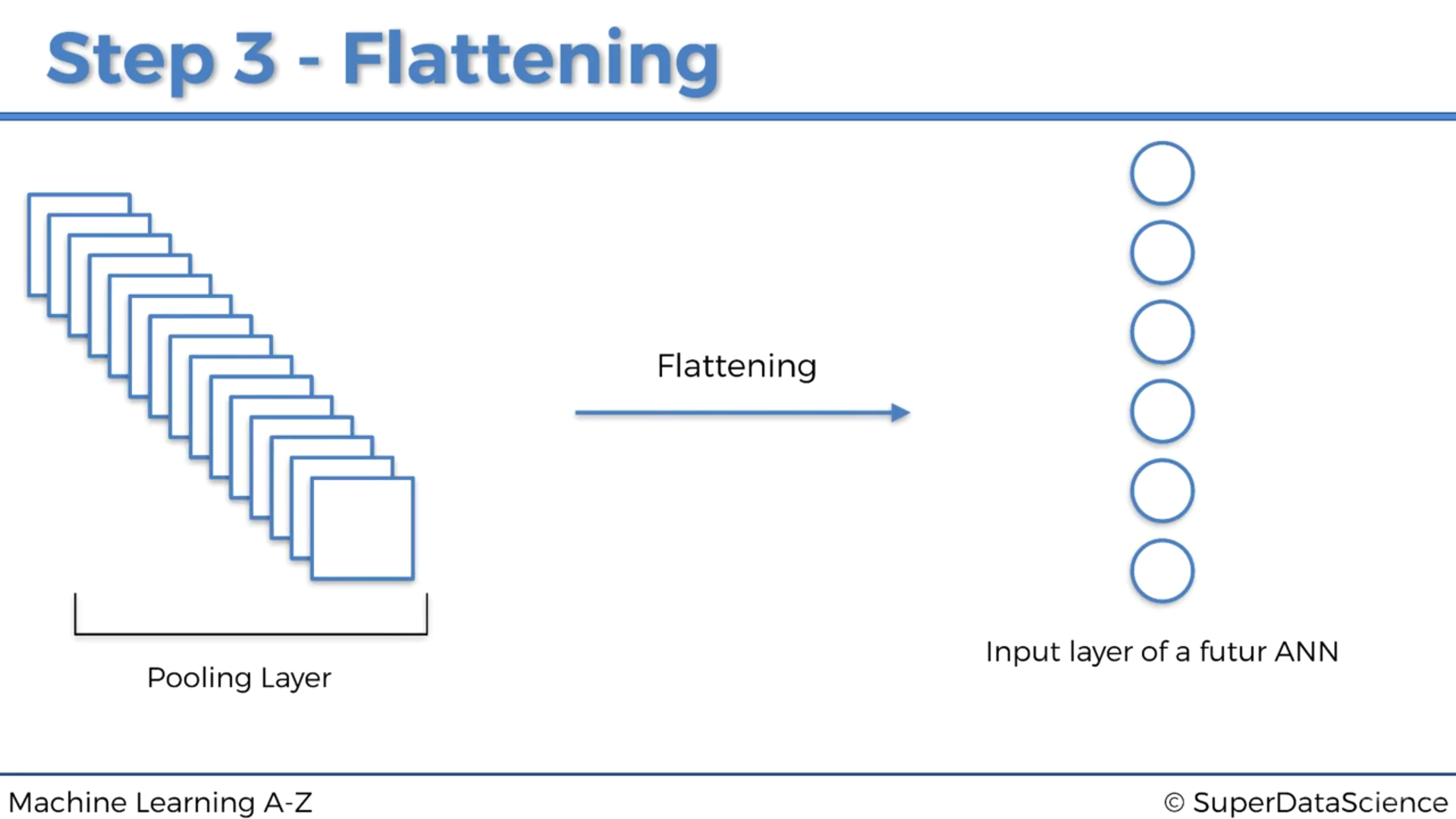
Nah untuk koding ketiga ini layer untuk max pooling, untuk proses ini sebenernya reduksi lagi hasil feature map covolution, untuk hasil pool sizenya rekomendasinya 2,2 tapi kalau mau custom bsia, Cuma saya enggan melakukannya, time consuming banget



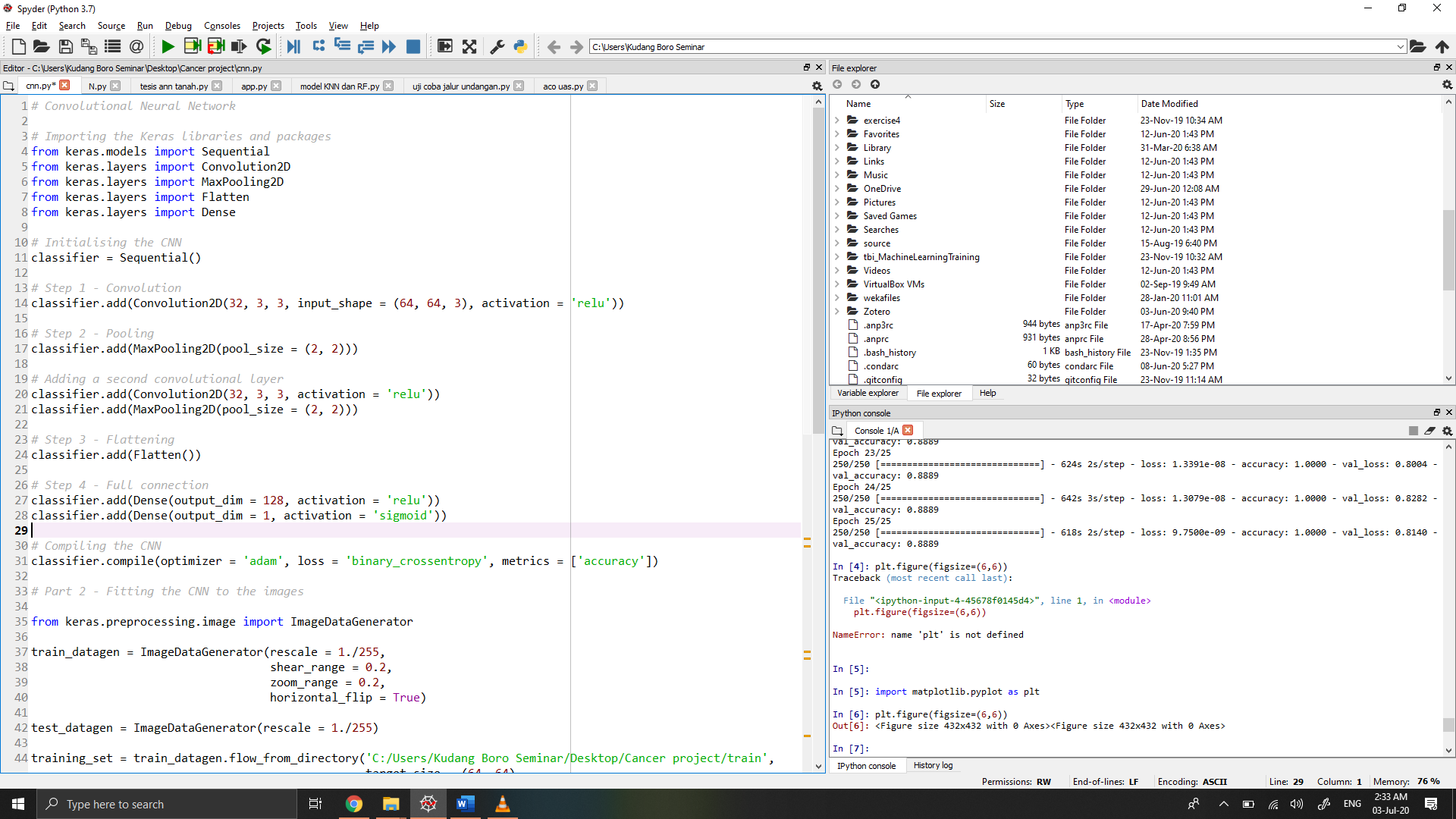




Nah step ketiga flattening, dari Namanya udh ketauan , proses ini yg menjadikan vector yg berbentuk 2 dimensi jadi satu dimensi

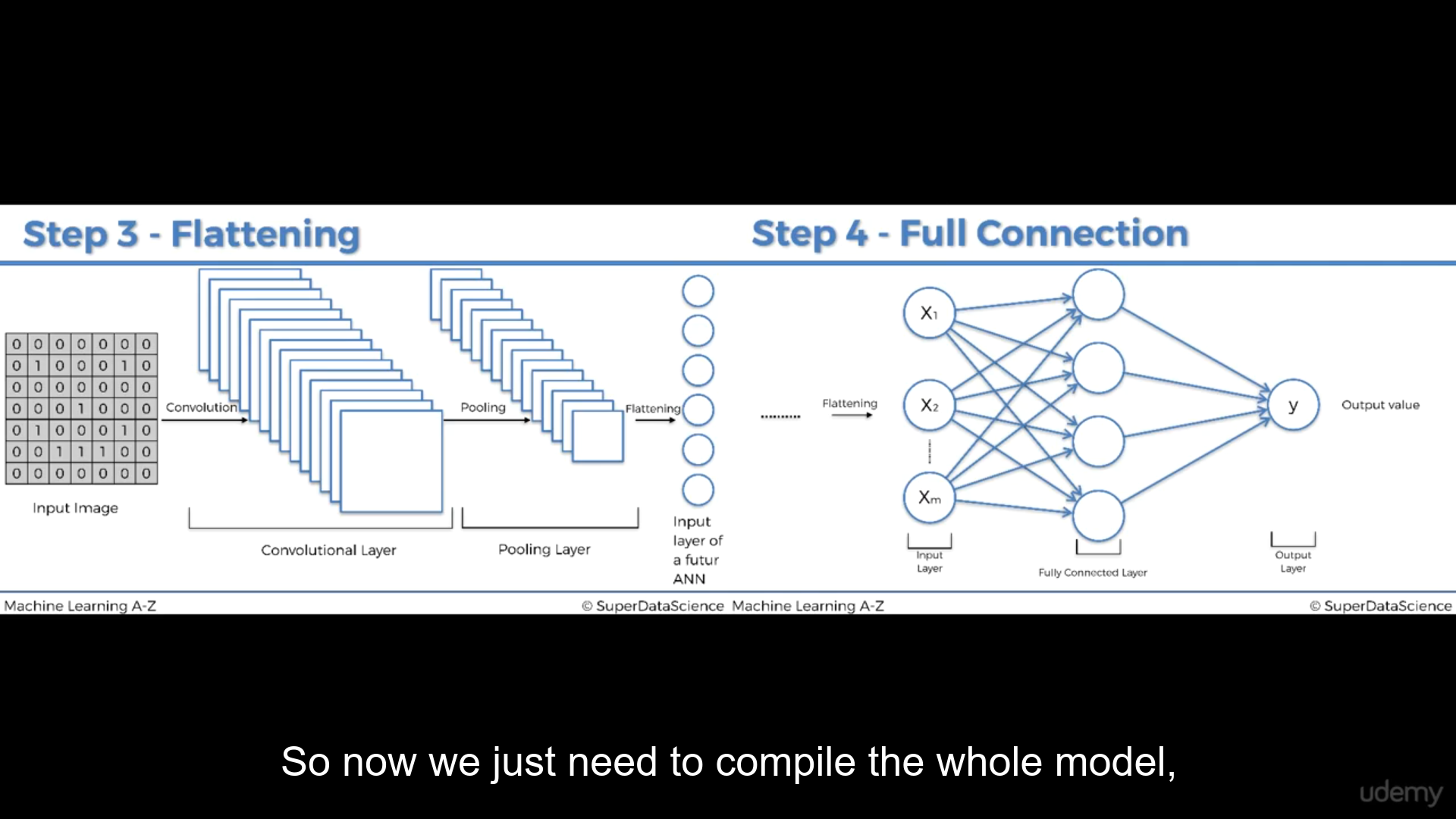


Nah step ketiga full connected



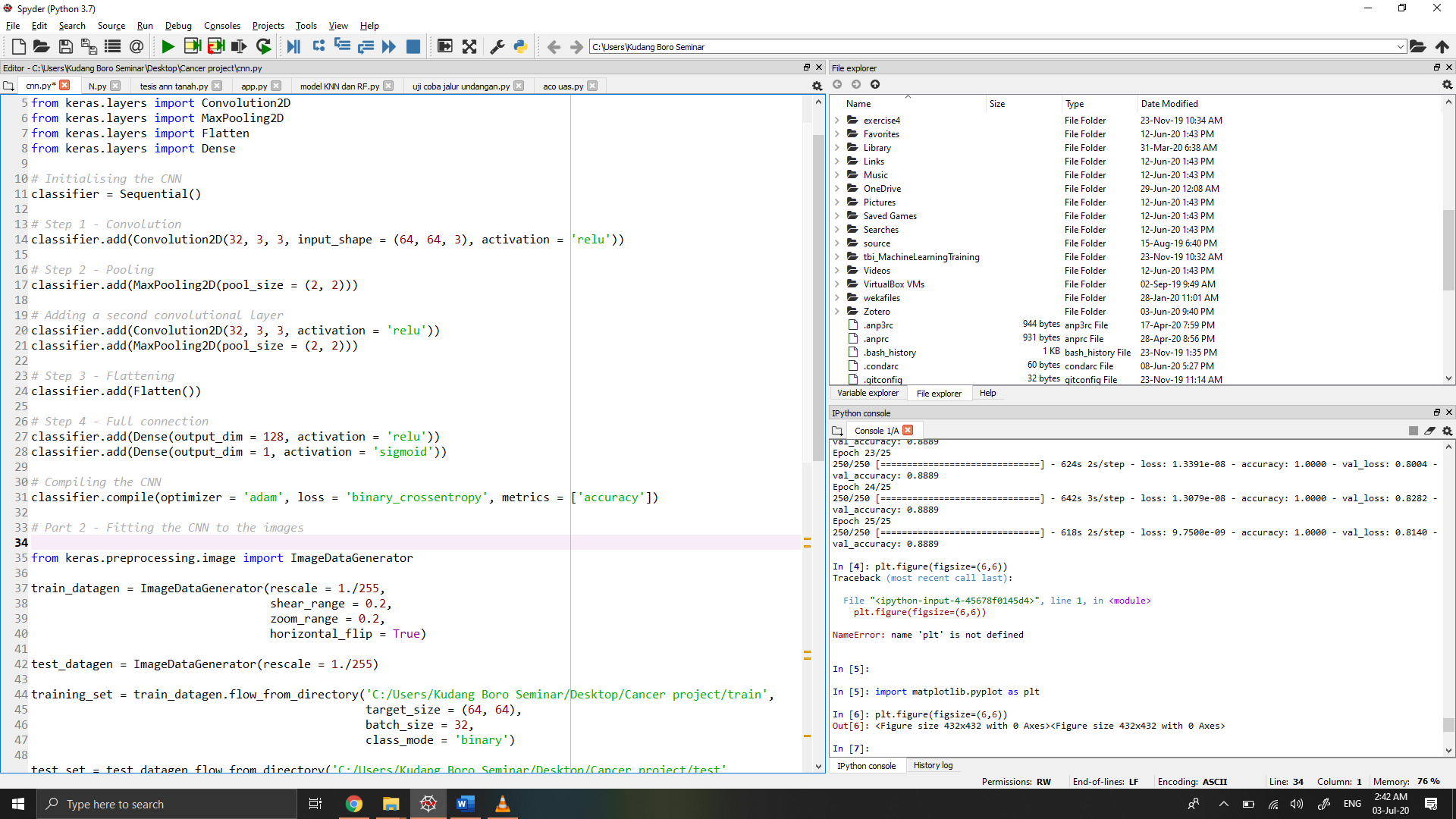
Untuk dense function ini neural neeywork rekomendasi yang dipakai, saya belum pernah coba yang lain tapi menarik si untuk hyperparameter tuning dengan didukung kekuatan laptop ahahaha

Untuk baris pertama output dim disini maksudnya hidden layer yang digunakan dan menggunakan fungsi aktivasi relu,” rekomendasi beberapa tutorial” nah untuk baris kedua untuk hasil yang diiinginkan, kan tujuan nya klasifikasi dengan 2 hasil yaitu normal atau kanker, nah disana output dimensi yang digunakan 1 dan fungsi aktivasi yang digunakan sigmoid untuk hasil klasifikasi dengan hasil 0,1



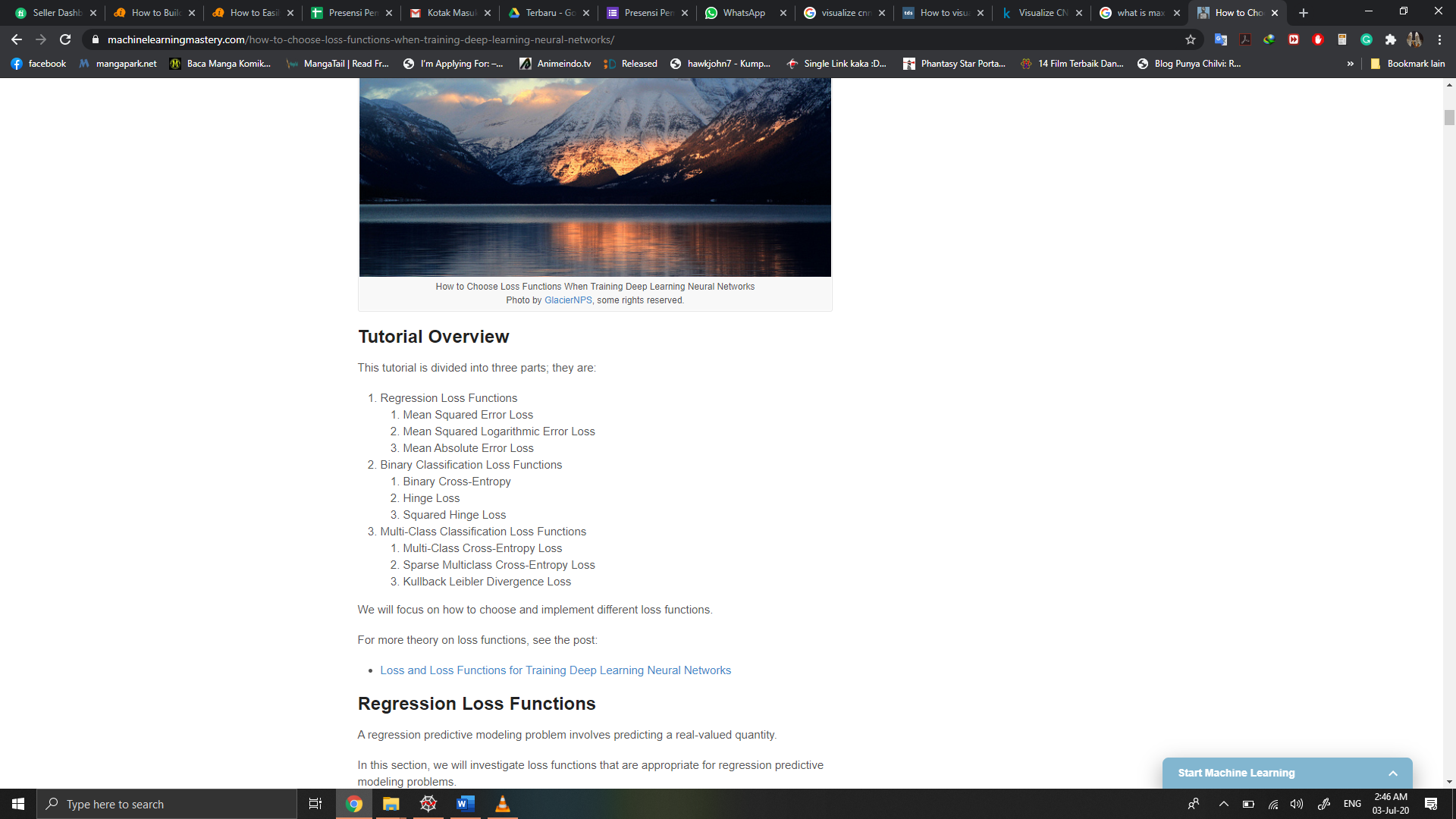
Ini secara keseluruhan proses yang terjadi

Ini step untuk compilenya

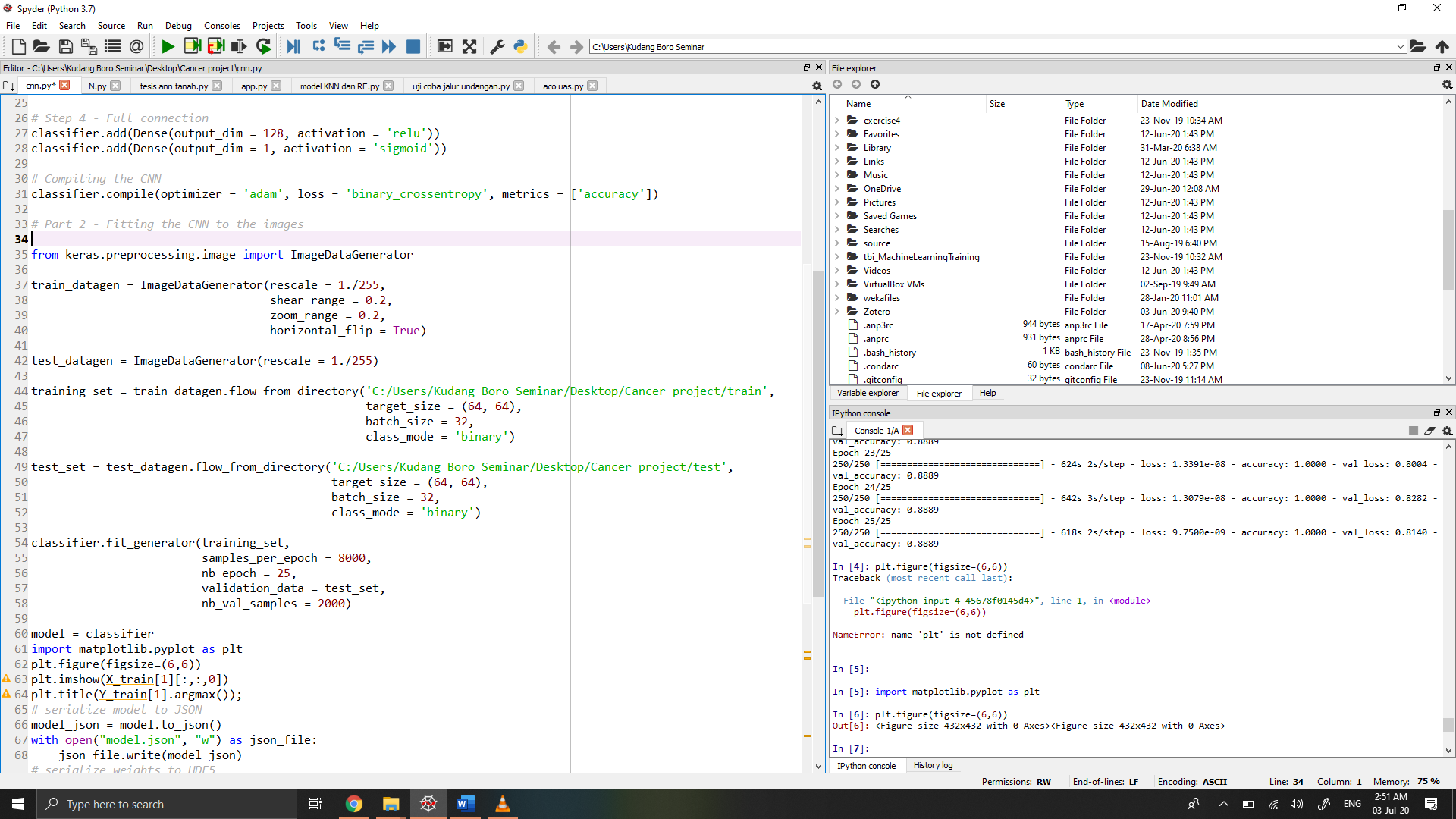


Optimizer yang digunakan adam, nah adam ini algoritma optimasi, untuk lebih jelasnya cari diinternet ehehehe

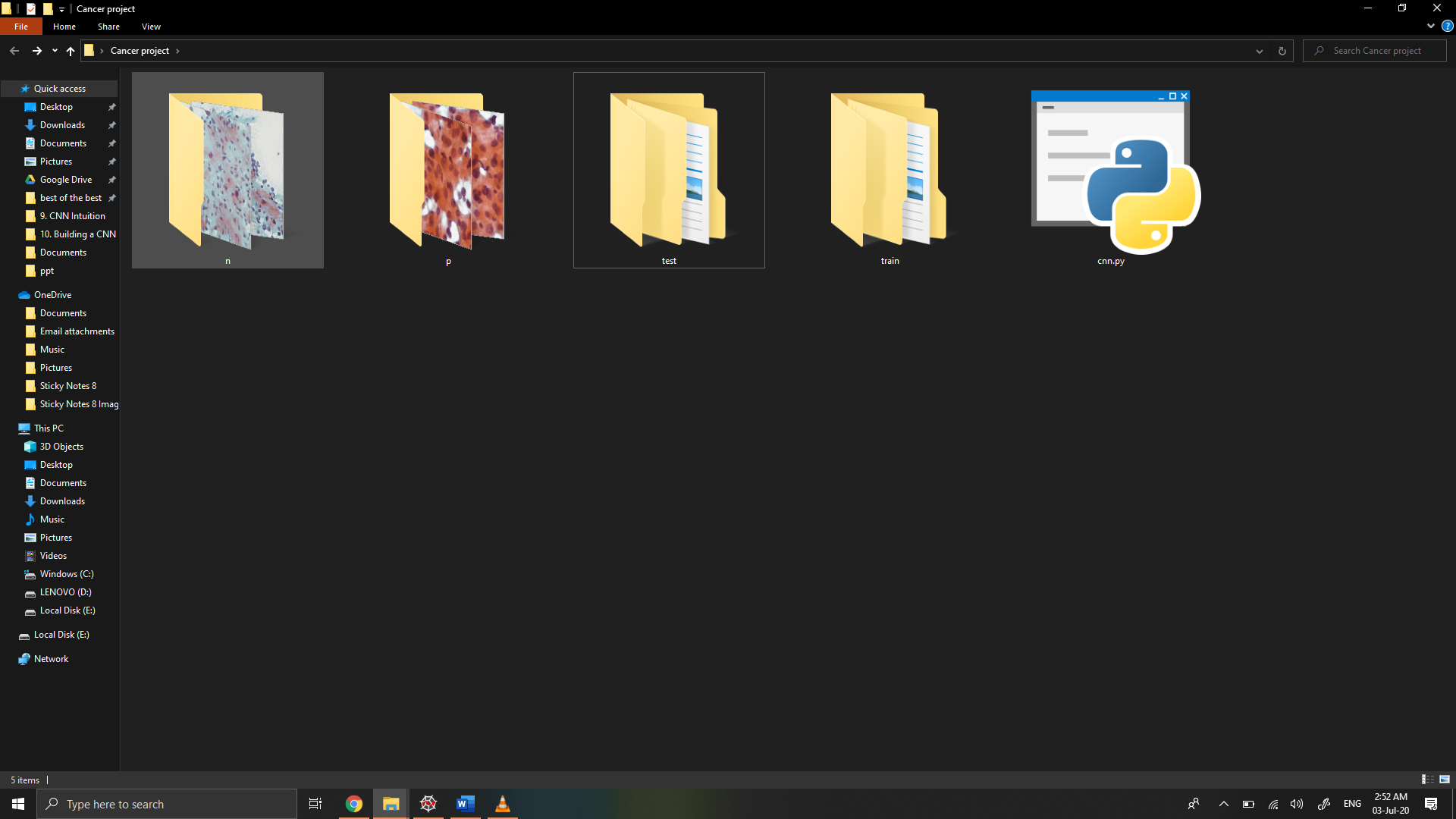
Untuk loss atau loss function disini menggunakan binary crossentropy, untuk loss function ini ada beberapa, tapi saya pilih itu mengikuti tutorial. Untuk loss function dapa dilihat di gambar di bawah



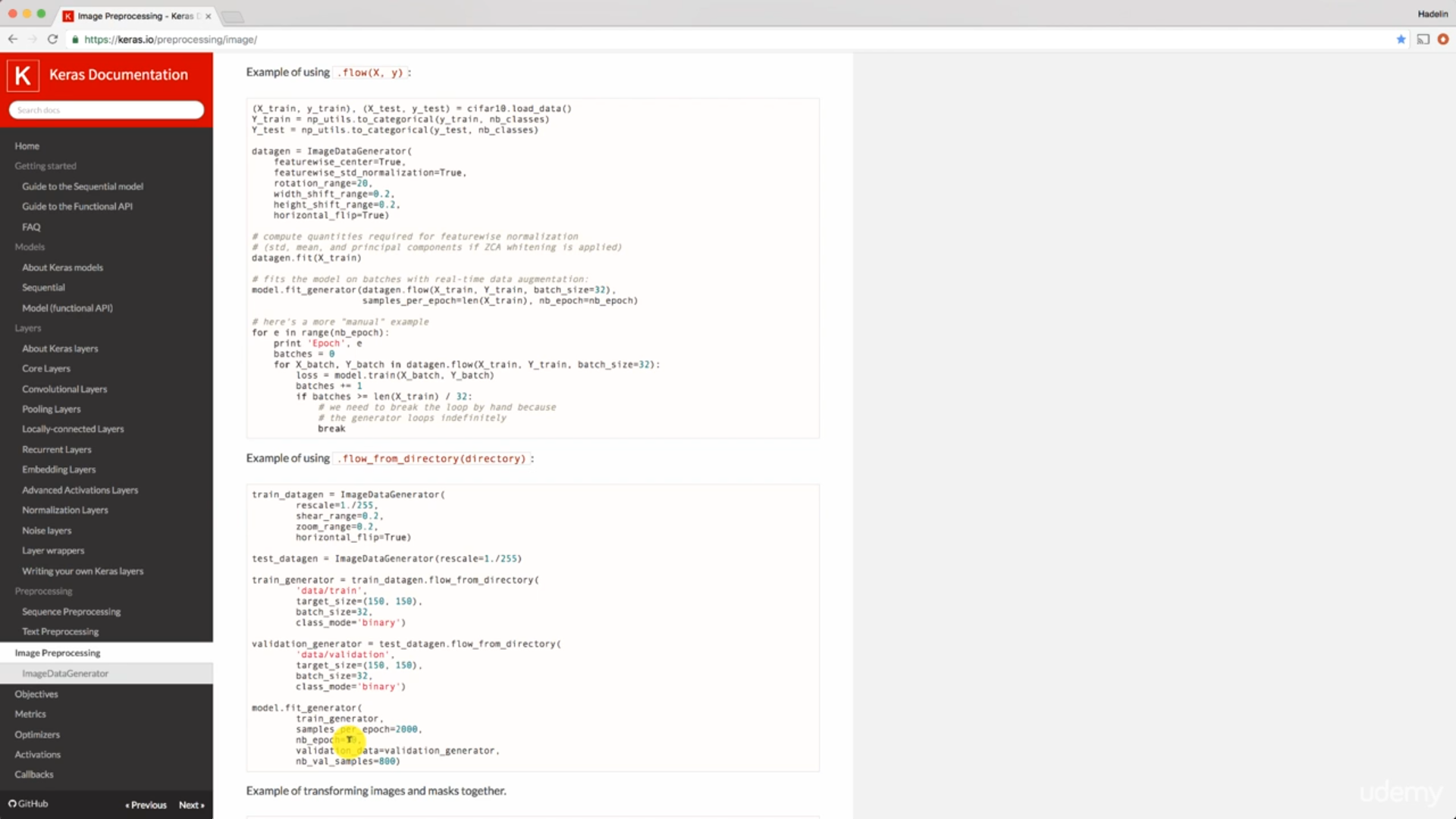
Nah step ini untuk menyiapkan input gamabr yang digunakan



Saya bagi dua data jadi data test dan data training kayak gambar di bawah tetapi saya kelompokkan mana dulu yang normal dan yg kanker setelah itu pilih acak dam masukkan ke folder datatest dan data training



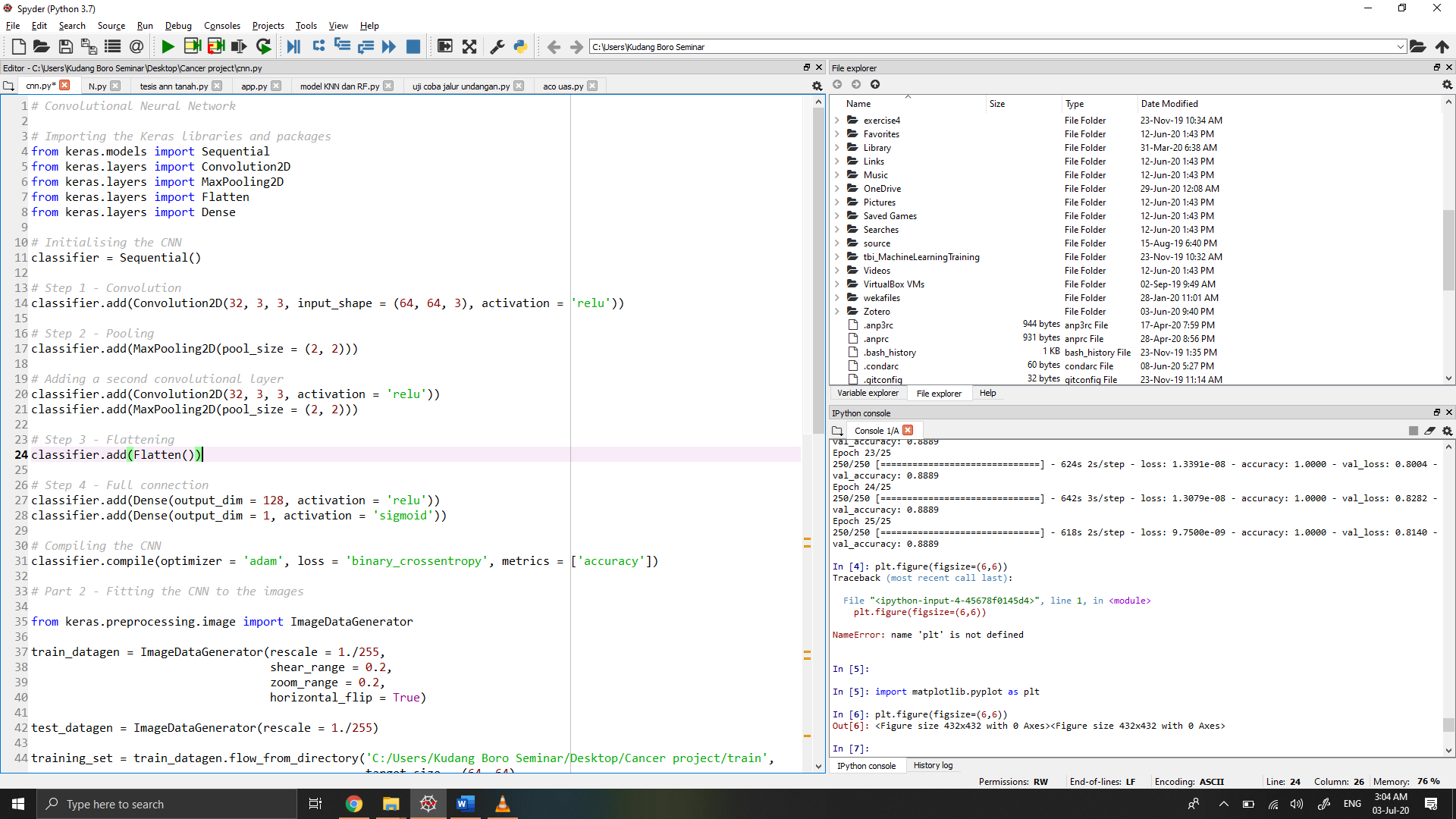
Untuk train data gen dan test datagen menggunakan rekomendais keras documentation dari rescale, shear range, zoom range dan horizontal flip, untuk training set yang diganti dari keras



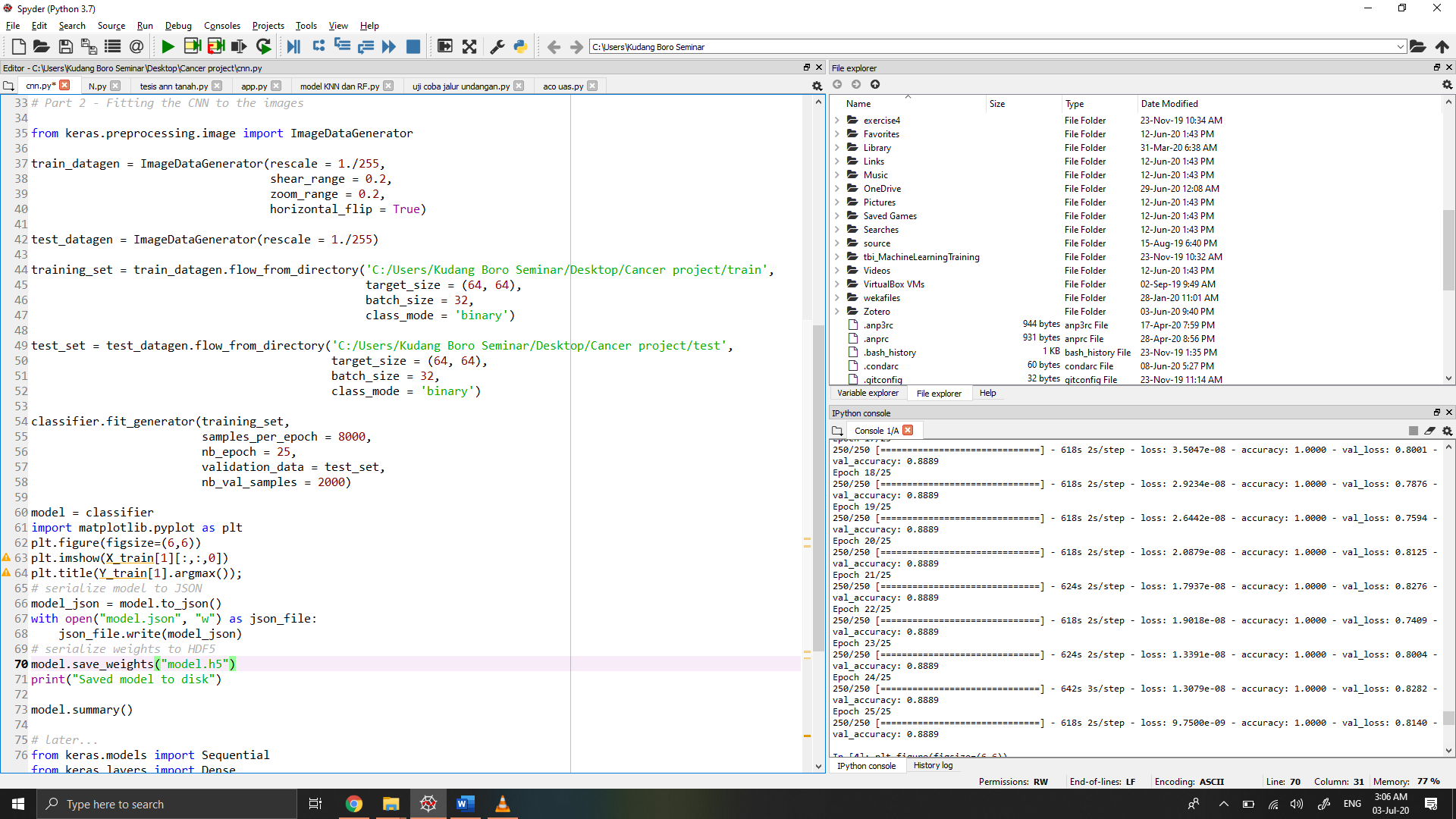
Nah yg diganti hanya target class dan class mode untuk training set dan test set, disessuaikan dengan parameter yang kita gunakan, disini target sizenya sesuai input dimension convolution yaitu 64,64 dan class mode binary

Untuk menggabungkan input dan model yang digunakan, paka fungsi fit generator, disini persampel dicoba di iterasi sekitar 8000 kali dengan filter iterasinya 25 untuk training set, dan untuk validations setnya 2000, saya mengikuti beberapa tutorial;

Dan dapat beberapa tambahan untuk menaikan akurasi, yaitu membuat layer convolution tambahan



Nah untuk hasil dari run



Akurasi yang didapat 0.889 persen, tapi kalau menurut saya masih bisa lebih bagus lagi kalau parameter yang digunakan di hyperparameter tuning (ini proses seluruh parameter di uji coba dan di cari mana parameter paling optimum untuk model) , tapi belum bisa di coba dengan spec laptop yang belum memadai, untuk proses ini saja memakan waktu 6 jam hanya dengan 30 gambar, mungkin kalau dimasa depan ada sekitar 500 gambar kanker dan non kanker bisa lebih meyakinkan lagi hasil yang didapat, belum lagi deep learning untuk image processing tidak hanya cnn tapi ada beberapa lain yang lebih optimum dan lebih cepat