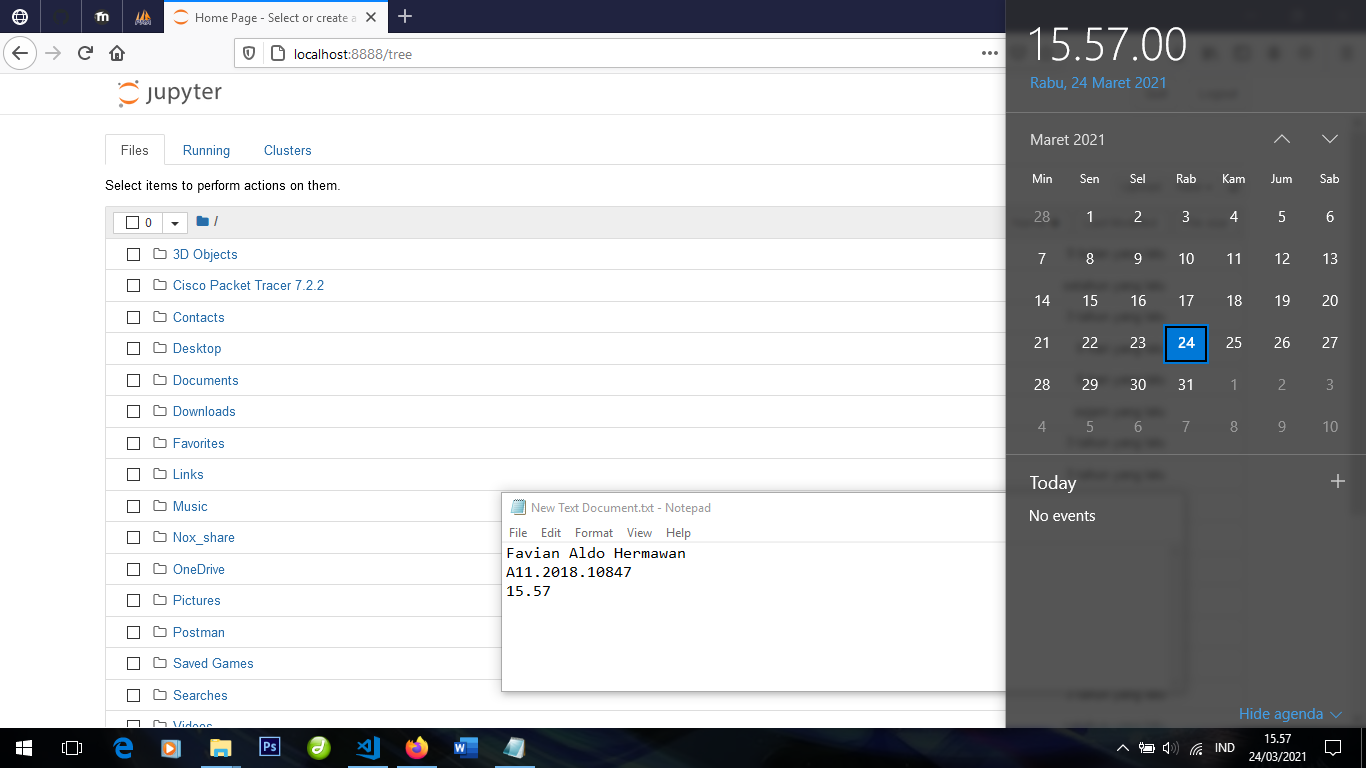
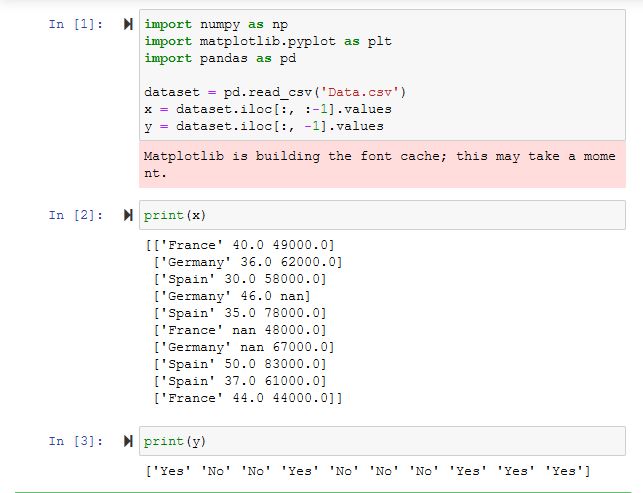
* Kerjakan Latihan tahapan preprocessing data pada slide sebelumnya, dapat menggunakan dataset yang lain / dimodifikasi, simpan dalam ***preprocessing.py*** atau ***preprocessing.ipynb***, repositorikan file pada **github.com** dan kirimkan URL github melalui Assignment pada kulino (Pada blok Minggu ke-3).

1. Instal Jupyter Notebook di komputer/laptop Anda. Kirim screenshot halaman jupyter notebook beserta bukti (nama, nim, hari dan jam saat itu) di forum ini maksimal tanggal 26 maret jam 23:55)
2. Kerjakan tugas yang ada di slide pertemuan 3 (slide ke 29). Dikumpulkan maximal tanggal 6 april 2021, URLnya dikirim ke komting.

Favian Aldo Hermawan  
A11.2018.10847  
15.57



Nama file : preprocessing.ipynb



Lakukan import numpy. Numpy sendiri meimliki kegunaan untuk operasi vektor dan matriks. Fiturnya hampir sama dengan MATLAB dalam mengelola array dan array multi dimensi.

Import matplotlib.pyplot. matplotlib untuk plotting 2D Python yang menghasilkan gambar publikasi bermutu di dalam berbagai format hardcopy dan lingkungan interaktif sepanjang platform.

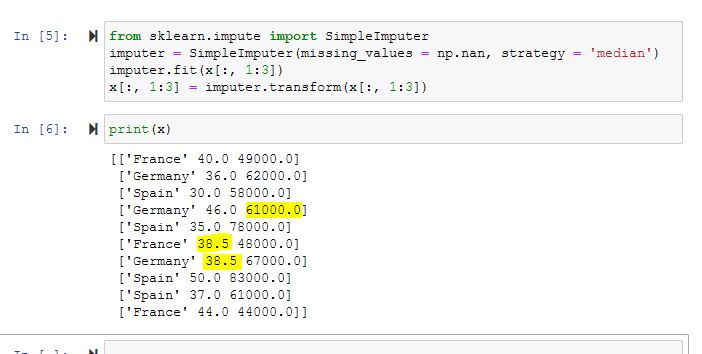
Import pandas. Pandas adalah librari analisis data yang memiliki struktur data yang diperlukan untuk membersihkan data mentah ke dalam sebuah bentuk yang cocok untuk analisis (yaitu tabel). Pandas melakukan tugas penting seperti menyelaraskan data untuk perbandingan dan penggabungan set data, penanganan data yang hilang, dll, itu telah menjadi sebuah librari de facto untuk pemrosesan data tingkat tinggi dalam Python (yaitu statistik).

Dataset disini untuk untuk membaca data csv ke dalam bentuk array.

dataset.iloc untuk memilih data berdasarkan index.

“:” itu menandakan memilih semua semua urutan pada index. “,”untuk pembatas baris dan kolom.

Jika data ada yang noisy (data yang salah/ tidak diinginkan) maka akan menampilkan nan.



//menghilangkan missing value

from sklearn.impute import SimpleImputer untuk menangani nilai kosong.

//mengganti nilai nan menjadi nilai

imputer = SimpleImputer(missing\_values=np.nan, strategy='median')

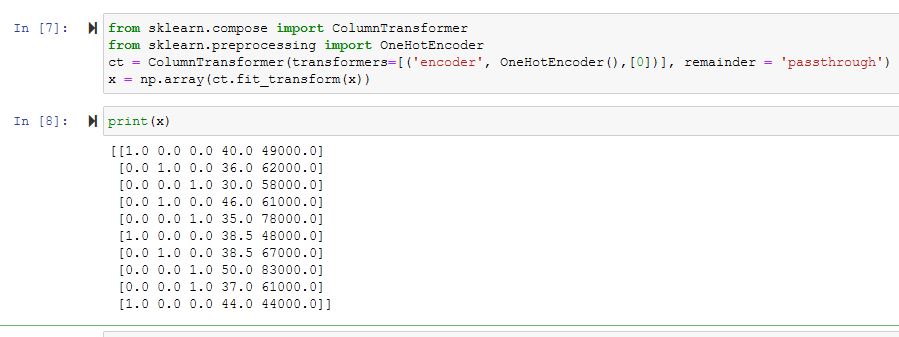
imputer = imputer.fit(X[:, 1:3])

X[:, 1:3] = imputer.transform(X[:, 1:3])

**missing\_values** : nilai data yang digunakan sebagai penanda bahwa nilai aslinya tidak ada (*missing*); dalam hal ini NaN (np.nan)

**strategy** : dalam hal ini rata-rata kolom (*'mean*'), bisa juga menggunakan *'median*', *'most\_frequent*' (modus), atau *'constant*’

Menjadi nan menjadi nilai median.



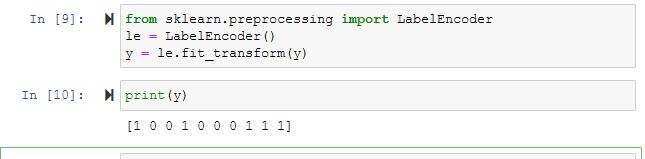
//mengkodekan data berdasarkan kategori

from sklearn.compose import ColumnTransformer

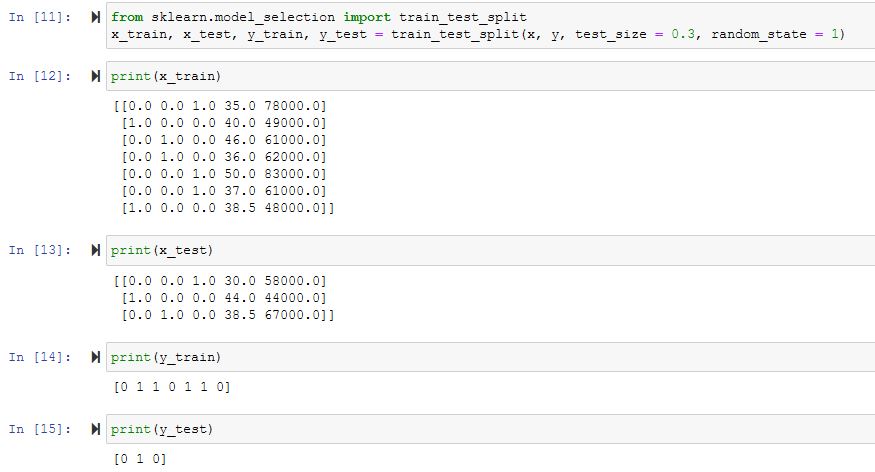
from sklearn.preprocessing import OneHotEncoder

membuat obyek ct dari kelas ColumnTransformer, dan parameternya :

* List Python berisi tuple berisi: string yang digunakan untuk mengenali transformer (misal ‘encoder’), nama transformer (OneHotEncoder()), dan list kolom (country memiliki index 0)
* Remainder : apakah kolom sisanya akan disertakan pada output ('passthrough') atau tidak ikut (‘drop’).



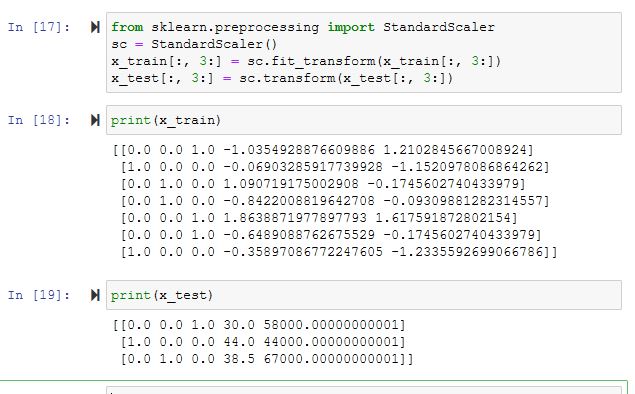
Menggunakan fit\_transform(), jalan pintas dari fit() dan transform(). Sekarang y akan menjadi seperti terlihat pada gambar di bawah. Terlihat bahwa 1 dan 0 telah menggantikan Yes dan No.



//membagi dataset ke dalam training set dan test set

* test\_size : proporsi test set, dalam hal ini 0.3
* train\_size: proporsi train size. Jika tidak di set, maka akan menyesuaikan dengan test size (dalam kasus ini 0.7). Berlaku kebalikannya.
* random\_state : konstan ini akan membuat hasil splitting tetap sama antar runtime atau antar mesin. Nilai bebas.

Karena kita hanya memiliki 10 observasi, rasionya menjadi 7:3.



**StandardScaler** menghilangkan mean (terpusat pada 0) dan menskalakan ke variansi (deviasi standar = 1), dengan asumsi data terdistribusi normal (gauss) untuk semua fitur.