# LAPORAN PRAKTIK PREPROSESSING DATASET HOUSING

Nama : Zulfikar Junirto

Npm : 22010022

## A. Source Lengkap

```
import os
import pandas as pd
from sklearn.preprocessing import LabelEncoder, MinMaxScaler
director = []
for i in os.listdir():
 direc_dir = os.path.join(os.getcwd(),i)
 director.append(direc dir)
print(director)
dataset_housing_train = pd.read_csv(director[1])
dataset housing train.head(10)
dataset housing train[['LotFrontage','MasVnrArea']].head(10) #menampilkan 10 kolom
LotFrontage dan MasVnrArea
dataset_housing_train[['MSZoning','Neighborhood']].head(10)
#Membuat Objek pada labelImage
dataset Neighborhood = LabelEncoder()
dataset_MSZoning = LabelEncoder()
# Transformasi kolom 'Neighborhood'
dataset housing train['Neighborhood'] =
dataset Neighborhood.fit transform(dataset housing train['Neighborhood'])
# Transformasi Kolom 'MSZoning'
dataset housing train['MSZoning'] =
dataset_MSZoning.fit_transform(dataset_housing_train['MSZoning'])
```

```
dataset_housing_train[['Neighborhood','MSZoning']].head(10)

#membuat Objel masing masing masing Kolom untuk normalisasi
normalisasi_SalePrice = MinMaxScaler()
normalisasi_GrLivArea = MinMaxScaler()

# Melakukan Normalisasi pada Kolom SalePrice
dataset_housing_train['SalePrice'] =
normalisasi_SalePrice.fit_transform(dataset_housing_train[['SalePrice']])
dataset_housing_train['GrLivArea'] =
normalisasi_GrLivArea.fit_transform(dataset_housing_train[['GrLivArea']])
dataset_housing_train[['SalePrice','GrLivArea']].head(10)
dataset_housing_train.to_csv('dataset_housing_train_fix.csv',index=False)
```

## **B.** Penjelasan Singkat Preprocessing

1. Import Library

```
[101] import os
import pandas as pd
from sklearn.preprocessing import LabelEncoder, MinMaxScaler
```

Melakukan import Beberapa Library yang digunakan untuk preprosessing Data, yaitu *Pandas,Sklearn*, dan *os*.

#### Mengatur Director

```
Mengabungkan Director dengan path File

[102] director = []
    for i in os.listdir():
        direc_dir = os.path.join(os.getcwd(),i)
        director.append(direc_dir)

Menampilkan file yang ada di director

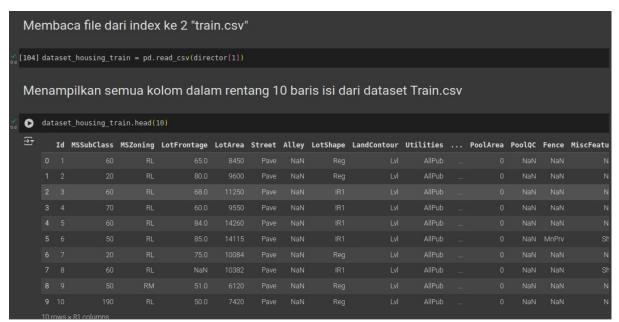
[103] print(director)

[103] print(director)

[104] [105] [107] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [108] [1
```

Melakukan looping pada Director dan menggabungkan Path dan update kedalam Array Agar mudah untuk Manage file file tertentu.

#### 3. Membaca semua Kolom yang ada di file



Lalu membuat sebuah variabe untuk menampung method  $pd\_read\_csv(FILE\ DIRECTOR)$ , lalu dipanggil lagi variabel tadi juga ditambahkan method head() tapi dengan rentang nilai index atau urutannya itu 10.

#### Note:

Kenapa Outputnya 9 bukan 10? Karna Kelompok lokasi memori berurutan yang semuanya memiliki tipe yang sama. Kumpulan data diindeks, atau diberi nomor, dan dimulai dari 0.

### 4. Missing Value

```
↑ ↓ ↑ ◎ 및 章 및 回 :
print(dataset_housing_train[['LotFrontage','MasVnrArea']].isnull().sum())

LotFrontage 259

MasVnrArea 8
dtype: int64
```

Menampilkan Jumla data yang kosong kolom *LotFrontage*, *dan MasVnrArea*, mengeluarkan Output dengan *LotFrontage 250 data* dan *MasVnArea 8* data.



Lalu menampilkan 10 data menggunakan *Method()* dengan rentang 10 untuk melihat mana data yang kosong, dan ternyara ada di bagian *LotFrontage*.

```
Mengisi Masing masing Kolom yang Sebelumnya Null menggunakan Method median() dari pandas dan mengatur 1 angka dibelakang koma dengan method round()

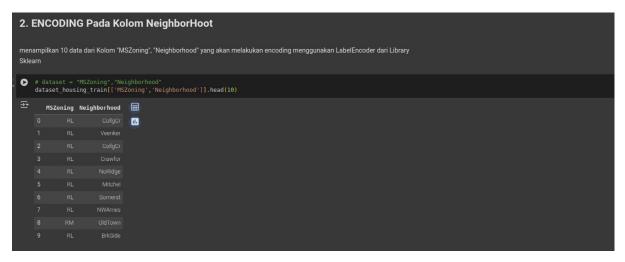
[115] dataset_housing_train['LotFrontage'] = dataset_housing_train['LotFrontage'].fillna(dataset_housing_train['MasVnrArea'].mean()).round(1)
dataset_housing_train['MasVnrArea'] = dataset_housing_train['MasVnrArea'].fillna(dataset_housing_train['MasVnrArea'].mean()).round(1)
```

Lalu Menggunakan *Method mean()* dan *fillnaa()*, untuk mengisi nilai nilai yang *Nan/atau kosong* lalu menggunakan *method round()* dengan input 1 yang artinnya membuat nilai bilangan binner dibelakang koma berjumlah 1 digit.



setelah menggunakan method *mean()* lalu menampilkan hasilnya, menggunakan method *isallnum().sum()* akan menampilkan jumlah data yang hilang, lalu dibawahnya juga diperlihatkan menggunakan Method *head(10)* yang artinnya menampilkan 10 baris. Dan terlihat pada kolom *LotFrontage()* pada index ke 7 yang tadinya *Nan* menjadi ada nilainya.

## 5. Encoding



pada proses Encoding ini adalah proses mengubah yang tadinya itu data kategorial menjadi numerik

pada source code tersebut yang pertama adalah membuat 2 variabel yang masing masing digunakan untuk per kolom, yang pertama ada *dataset\_Neighborhood* dan *dataset\_MSZoning* yang itu digunakan untuk membuat objek dari library *sklearn* dengan method *labelEncoder karna* nantinya akan otomatis merubah menjadi numerik.



Setelah melakukan proses encoding hasilny akan seperti itu karna dia merubah numerik pada label tertentu misal *RL* adalah angka *1* dan *RM* adalah angka *4*.

#### 6. Normalisasi



menampilkan data sebelum di normalisasi yaitu pada Kolom "SalePrice" dan "GrLivArea".

```
Melakukan Normalisasi Data pada Kolom SalePrice GrLivArea

** [136] #membuat Objel masing masing masing Kolom untuk normalisasi
normalisasi_SalePrice = MinMaxScaler()
normalisasi_GrLivArea = MinMaxScaler()

# Melakukan Normalisasi pada Kolom SalePrice
dataset_housing_train['SalePrice'] = normalisasi_SalePrice.fit_transform(dataset_housing_train['SalePrice']])
dataset_housing_train['GrLivArea'] = normalisasi_GrLivArea.fit_transform(dataset_housing_train['GrLivArea']])
```

Step ini sama seperti encoding cuma bedanya hanya pada *value* di variabel masing masing kolom yang akan di *Normalisasi*.



Tampilan setealah kolom *SalePrice* dan GrLivArea menjadi bilangan biner yang bertipe data float64. Hal ini akan memudahkan model untuk dilatih.