

## Code used to normalize data with MinMaxScaler

```
In [14]: import numpy as np
import pickle
import os
from sklearn.preprocessing import MinMaxScaler
```

```
In [9]: # Load data
path_to_file = "HDataset.pkl"
infile_c = open(path_to_file, 'rb')
HDataset = pickle.load(infile_c)
infile_c.close()
```

```
In [10]: # Clear data
# Remove vectors with len != 42 (in my case lenght of hand landmarks is 42)
# Note: some case is 84 because two hands are detected
print(len(HDataset["vectors"]), len(HDataset["labels"]))

new_v = []
new_l = []

for i in range(len(HDataset["vectors"])):
    if len(HDataset["vectors"][i]) != 42:
        print("remove:", len(HDataset["vectors"][i]), i, HDataset["labels"][i])
    else:
        new_v.append(HDataset["vectors"][i])
        new_l.append(HDataset["labels"][i])

HDataset["vectors"] = new_v
HDataset["labels"] = new_l

print(len(HDataset["vectors"]), len(HDataset["labels"]))

13000 13000
remove: 84 2428 E
12999 12999
```

```
In [11]: # Normalization MinMax scaler [0,1]

def norm_min_max(vec: np.array):
    scaler = MinMaxScaler()
    scaler.fit(vec)
    norm = scaler.transform(vec)
    norm = norm.reshape(1, len(norm))[0]
    return list(norm)

normalized_v = []

for vec in HDataset["vectors"]:
    normalized_v.append(norm_min_max(np.array(vec).reshape(-1, 1)))

HDataset["vectors"] = normalized_v
```

```
In [12]: # Replace string labels (str) [a to z] with numbers (int) [0 to 26]

from string import ascii_lowercase

# Dict of alphabet
letters = {letter: index for index, letter in enumerate(ascii_lowercase, start=0)}

for idx in range(len(HDataset["labels"])):
```

```
# Replace char with class (int)
HDataset["labels"][idx] = letters[HDataset["labels"][idx].lower()]
```

```
In [ ]: # Save

with open('HDataset_norm.pkl', 'wb') as f:
    pickle.dump(HDataset, f, protocol=pickle.HIGHEST_PROTOCOL)
```