

# Train Test Spilt (Scikit-Learn)

## Machine Learning with Pyhton

In [11]:

```
from PIL import Image
import urllib.request
import io
import os
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
```

In [ ]:

```
# Use this to inform an image from local directory (converting to gray scale)
im = Image.open('images_teste/image05.jpg').convert("L")
```

In [ ]:

```
# Use this to inform an image from web, converting to Gray scale
web_image = "https://img1.thelist.com/img/gallery/what-happens-to-your-body-when-you-cry/intro-1525376112.jpg"
fd = urllib.request.urlopen(web_image)
image_file = io.BytesIO(fd.read())
im = Image.open(image_file).convert("L")
```

In [ ]:

```
# Run it if you want to open the image file
im.show()
```

In [2]:

```
# Indique aqui o diretorio de envio e o arquivo para enviar
path = 'images_teste/'

included_extensions = ['jpg', 'bmp', 'png', 'gif']
image_list = [fn for fn in os.listdir(path)
               if any(fn.endswith(ext) for ext in included_extensions)]
```

In [3]:

```
len(image_list)
```

Out[3]:

In [4]:

```
pixels = []
for image_file in image_list:
    im = Image.open(path+image_file).convert("L")
    pix_val = [im.getdata()]
    pix_val_flat = [x for sets in pix_val for x in sets]
    pixels.append(' '.join(str(x) for x in pix_val_flat))
```

In [5]:

```
len(pixels)
```

Out[5]:

6

In [6]:

```
# Create a DataFrame
d = {'Class':np.nan, 'Image':image_list, 'Pixels':pixels}
df = pd.DataFrame(d)
```

In [7]:

```
# Show DataFrame
df
```

Out[7]:

	Class	Image	Pixels
0	NaN	image01.jpg	52 52 51 52 51 52 50 51 53 53 52 51 53 50 50 5...
1	NaN	image02.jpg	4 4 3 2 2 2 2 3 3 3 3 4 2 2 3 2 3 2 2 3 3 ...
2	NaN	image03.jpg	84 79 79 90 100 107 120 134 148 155 170 185 20...
3	NaN	image04.jpg	2 2 2 2 2 2 2 2 3 3 3 3 3 4 4 5 7 8 10 12 13 1...
4	NaN	image05.jpg	94 94 94 94 95 95 96 96 96 96 97 97 97 98 98 9...
5	NaN	image06.jpg	39 39 39 39 39 39 39 39 39 39 40 40 40 40 40 4...

In [8]:

```
train, test = train_test_split(df, test_size=0.2, random_state=200)
```

In [9]:

```
train
```

Out[9]:

	Class	Image	Pixels
3	NaN	image04.jpg	2 2 2 2 2 2 2 3 3 3 3 3 4 4 5 7 8 10 12 13 1...
0	NaN	image01.jpg	52 52 51 52 51 52 50 51 53 53 52 51 53 50 50 5...
1	NaN	image02.jpg	4 4 3 2 2 2 2 3 3 3 3 4 2 2 3 2 3 2 2 3 3 ...
2	NaN	image03.jpg	84 79 79 90 100 107 120 134 148 155 170 185 20...

In [10]:

```
test
```

Out[10]:

	Class	Image	Pixels
4	NaN	image05.jpg	94 94 94 94 95 95 96 96 96 96 97 97 97 98 98 9...
5	NaN	image06.jpg	39 39 39 39 39 39 39 39 39 39 40 40 40 40 40 4...

In [ ]:

```
# Now you are able to export to csv files and to add classification to train dataset
```