Train Test Split

Import the relevant libraries

In [1]:

```
# In this lesson we will explore the train_test_split module
# Therefore we need no more than the module itself and NumPy
import numpy as np
from sklearn.model_selection import train_test_split
```

Generate some data we are going to split

In [2]:

```
# Let's generate a new data frame 'a' which will contain all integers from 1
# The method np.arange works like the built-in method 'range' with the diffe
a = np.arange(1,101)
```

In [3]:

```
# Let's check it out
a
```

Out[3]:

```
array([  1,   2,   3,   4,   5,   6,   7,   8,   9,  10,  11,  12,  13,
        14,  15,  16,  17,  18,  19,  20,  21,  22,  23,  24,  25,  26,
        27,  28,  29,  30,  31,  32,  33,  34,  35,  36,  37,  38,  39,
        40,  41,  42,  43,  44,  45,  46,  47,  48,  49,  50,  51,  52,
        53,  54,  55,  56,  57,  58,  59,  60,  61,  62,  63,  64,  65,
        66,  67,  68,  69,  70,  71,  72,  73,  74,  75,  76,  77,  78,
        79,  80,  81,  82,  83,  84,  85,  86,  87,  88,  89,  90,  91,
        92,  93,  94,  95,  96,  97,  98,  99, 100])
```

In [4]:

```
# Similarly, let's create another ndarray 'b', which will contain integers f
# We have intentionally picked these numbers so we can easily compare the tw
# Obviously, the difference between the elements of the two arrays is 500 fo
b = np.arange(501,601)
b
```

Out[4]:

```
array([501, 502, 503, 504, 505, 506, 507, 508, 509, 510, 511, 512, 513,
       514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526,
       527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539,
       540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552,
       553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 565,
       566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 578,
       579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589, 590, 591,
       592, 593, 594, 595, 596, 597, 598, 599, 600])
```

Split the data

Full documentation: [https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html (https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html)](https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html)

In [5]:

```
# Let's check out how this works
train_test_split(a)
```

Out[5]:

```
[array([87, 32, 90,  1,  2,  8, 51, 73, 22, 95,  4, 57, 27, 58, 48, 99, 96,
        74, 72, 29, 76, 64,  3, 12, 53,  6, 18, 16, 65, 66, 63, 46, 39, 17,
        91, 25, 15, 78, 83, 19, 45, 68, 33, 98, 97, 14, 44, 86, 80, 34, 70,
        47, 54, 93, 94, 85, 42, 60, 92, 41, 61, 71, 89, 23, 21, 11, 84, 13,
        82, 59, 49, 79, 36, 55,  5]),
 array([ 24,  56,  40,   9,  69,  75,  10,  28,  38,  30,  62,  67, 100,
         88,  37,  20,   7,  31,  77,  43,  35,  26,  81,  52,  50])]
```

In [54]:

```
# There are several different arguments we can set when we employ this metho
# Most often, we have inputs and targets, so we have to split 2 different ar
# we are simulating this situation by splitting 'a' and 'b'

# You can specify the 'test_size' or the 'train_size' (but the latter is dep
# essentially the two have the same meaning
# Common splits are 75-25, 80-20, 85-15, 90-10

# Finally, you should always employ a 'random_state'
# In this way you ensure that when you are splitting the data you will alway

# Note 2 arrays will be split into 4
# The order is train1, test1, train2, test2
# It is very useful to store them in 4 variables, so we can later use them
a_train, a_test, b_train, b_test = train_test_split(a, b, test_size=0.2, ran
```

Explore the result

In [55]:

```
# Let's check the shapes
# Basically, we are checking how does the 'test_size' work
a_train.shape, a_test.shape
```

Out[55]:

```
((80,), (20,))
```

In [56]:

```
# Explore manually
a_train
```

Out[56]:

```
array([ 25,  32,  99,  73,  91,  66,   3,  59,  94,   1,   8,  15,  90,
        54,  31,  20,  77,  82,  30,  35,  95,  42,  38,   7,  11,  50,
        21,  48,   2,  17,  10,  58,  68,  43,  41,  16,  88,  72,  79,
       100,  80,  39,  24,  86,  22,  23,  62,  76,  18,  47,  55,  26,
        60,  19,  71,  64,  51,  63,  65,  28,  12,  78,  13,  44,  75,
        87,  40,   4,  29,  49,  37,  57,  27,  74,   6,  45,  92,  34,
        53,  83])
```

In [57]:
```
# Explore manually
a_test
```

Out[57]:
```
array([ 9, 69, 81, 56, 33, 93, 84, 61, 46, 89, 85, 67, 97,  5, 70, 36, 98,
       96, 14, 52])
```

In [58]:
```
b_train.shape, b_test.shape
```

Out[58]:
```
((80,), (20,))
```

In [59]:
```
b_train
```

Out[59]:
```
array([525, 532, 599, 573, 591, 566, 503, 559, 594, 501, 508, 515, 590,
       554, 531, 520, 577, 582, 530, 535, 595, 542, 538, 507, 511, 550,
       521, 548, 502, 517, 510, 558, 568, 543, 541, 516, 588, 572, 579,
       600, 580, 539, 524, 586, 522, 523, 562, 576, 518, 547, 555, 526,
       560, 519, 571, 564, 551, 563, 565, 528, 512, 578, 513, 544, 575,
       587, 540, 504, 529, 549, 537, 557, 527, 574, 506, 545, 592, 534,
       553, 583])
```

In [60]:
```
b_test
```

Out[60]:
```
array([509, 569, 581, 556, 533, 593, 584, 561, 546, 589, 585, 567, 597,
       505, 570, 536, 598, 596, 514, 552])
```