

## 머신러닝 vs 딥러닝

- 머신러닝
  1. 특징 추출을 인간이 해야함! => 머신러닝 모델 => 결과
  2. Scikit-learn(sklearn)
- 딥러닝
  1. 데이터 넣음 => 딥러닝(특징 추출) => 결과
  2. Tensorflow(KERAS) => 고수준 API
  3. PyTorch
    - 빅데이터분석기사, ADsP, tensorflow => kaggle Master

In [2]:

```
from tensorflow import keras
```

In [4]:

```
(train_input, train_target), (test_input, test_target) = keras.datasets.fashion_mnist
```

```

Downloading data from https://storage.googleapis.com/tensorflow/tf-ker
as-datasets/train-labels-idx1-ubyte.gz (https://storage.googleapis.co
m/tensorflow/tf-keras-datasets/train-labels-idx1-ubyte.gz)
32768/29515 [=====] - 0s 0us/step
40960/29515 [=====] - 0s 0us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-ker
as-datasets/train-images-idx3-ubyte.gz (https://storage.googleapis.co
m/tensorflow/tf-keras-datasets/train-images-idx3-ubyte.gz)
26427392/26421880 [=====] - 0s 0us/step
26435584/26421880 [=====] - 0s 0us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-ker
as-datasets/t10k-labels-idx1-ubyte.gz (https://storage.googleapis.com/
tensorflow/tf-keras-datasets/t10k-labels-idx1-ubyte.gz)
16384/5148 [=====] - 0s 0us/step
=====
Downloading data from https://storage.googleapis.com/tensorflow/tf-ker
as-datasets/t10k-images-idx3-ubyte.gz (https://storage.googleapis.com/
tensorflow/tf-keras-datasets/t10k-images-idx3-ubyte.gz)
4423680/4422102 [=====] - 0s 0us/step
4431872/4422102 [=====] - 0s 0us/step

```

## 2. 데이터 탐색

In [6]:

```
print(train_input.shape, train_target.shape)
```

```
(60000, 28, 28) (60000,)
```

In [7]:

```
print(test_input.shape, test_target.shape)
```

```
(10000, 28, 28) (10000,)
```

In [9]:

```
import matplotlib.pyplot as plt

fig, axs = plt.subplots(1, 10, figsize=(10,10))
for i in range(10):
    axs[i].imshow(train_input[i], cmap='gray_r')
    axs[i].axis('off')
plt.show()
```



In [11]:

```
import numpy as np
# 패션 mnist 는 0-9(10개의 패션 아이템 카테고리)
# 티셔츠, 바지, 스웨터, 드레스, 코트, 샌달, 셔츠, 스니커즈, 가방, 앵클부츠

print(np.unique(train_target, return_counts=True))
```

```
(array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9], dtype=uint8), array([6000, 600
0, 6000, 6000, 6000, 6000, 6000, 6000, 6000, 6000]))
```

### 3. 데이터 전처리

In [12]:

```
train_scaled = train_input / 255.0
train_scaled = train_scaled.reshape(-1, 28*28)
```

In [13]:

```
print(train_scaled.shape)
```

```
(60000, 784)
```

### 4. 인공신경망

In [14]:

```
import tensorflow as tf
from tensorflow import keras
```

In [15]:

```
from sklearn.model_selection import train_test_split

train_scaled, val_scaled, train_target, val_target = train_test_split(train_scaled,
```

In [18]:

```
print(train_scaled.shape, train_target.shape) # 학습데이터를 train과 val로 나눔
```

```
(48000, 784) (48000,)
```

In [17]:

```
print(val_scaled.shape, val_target.shape)
```

```
(12000, 784) (12000,)
```

In [19]:

```
desne = keras.layers.Dense(10, activation='softmax', input_shape=(784, )) #딥러닝모델
```