3주. Machine Learning Concept			
학번	32183164	이름	이석현

Q1. 전통적인 SW 와 머신러닝을 적용한 SW 의 차이점을 설명하시오

전통적 SW: 규칙을 인간이 알아내어 알고리즘의 형태로 SW 안에 구현한다.
머신러닝을 적용한 SW: 규칙을 알아내는 방법을 인간이 제시하고 머신이 실제 규칙을 알아낸다.

Q2. 머신러닝에서 러닝(learning)의 실제적인 의미를 설명하시오.

머신이 규칙을 알아내는 과정(인간 입장에서는 머신을 훈련시키는 과정)

Q3. 머신러닝이 가능한 이유를 설명하시오.

머신러닝은 과거의 축적된 데이터를 학습하여 미래를 예측하는 기술로, 과거의 데이터로부터 규칙을 찾아 일반화 한 후, 이를 미래의 예측에 활용하기에 머신러닝이 가능하다.

Q4. 회귀(regression) 와 분류(classification)의 차이점을 설명하시오

회귀는 반응변수가 수치형일 때 사용하고 분류는 반응변수가 정확한 수치가 아니라 범위일 때 사용한다.

Q5. 기후변화에 따른 연평균 기온을 예측하는 머신러닝 모델을 만들려고 한다. (2점)
1) 모델을 만들기 위해 필요한 것은 무엇인가
2) 이 모델은 회귀, 분류, 군집화, 강화학습중 어느 기술을 적용해야 하는가? 그 이유는 무엇인가

1) 과거데이터를 이용해 training data, validation data, test data로 나눈 후 training을 시켜 model을 만들고 test data로 model accuracy를 평가해야한다.
2) 회귀. 과거의 데이터를 이용해 답을 줄 수 있으므로 지도학습을 택하고 그 중에서도 반응변수가 수치형이기 때문에 회귀를 적용해야한다.

Q6. 머신러닝 모델을 개발할 때 데이터셋을 training data 와 test data 로 나누는 이유는 무엇인가? 나누지 않는다면 어떤 문제가 발생하는가

training data는 과거 데이터의 역할로 이를 이용하여 학습을 시키고 test data는 미래데이터의 역할로 이를 이용해 미래 예측시 모델의 성능을 판단하는 자료로 사용된다.
나누지 않는다면 과거 데이터로 모두 사용하게 될 것이므로 미래 예측시 모델의 성능을 판단할 수 없게 된다.

Q7. scikit-learn 홈페이지(https://scikit-learn.org/stable/)를 방문하여 scikit-learn에서 제공하는 군집화(clustering) 일고리즘에는 어떤 것들이 있는지 찾아서 제시하시오

K-Means, Affinity propagation, Mean-shift, Spectral clustering, Ward hierarchical clustering, DBSCAN, OPTICS, Gaussian mixtures, Birch

Q8. Pandas 모듈을 이용하여 배포된 데이터셋중 cars 데이터셋을 읽어온 후 다음 문제를 해결하시오 (2점)
(1) 데이터셋의 위쪽 5행을 보이시오
(2) 데이터셋의 컬럼들 이름을 보이시오
(3) 데이터셋의 두 번째 컬럼의 값들만 보이시오.
(4) 데이터셋의 11~20행 자료중 speed 컬럼의 값들만 보이시오.
(5) speed 가 20 이상인 행들의 자료만 보이시오
(6) speed 가 10 보다 크고 dist 가 50보다 큰 행들의 자료만 보이시오.
(7) speed 가 15 보다 크고 dist 가 50보다 큰 행들은 몇 개인지 보이시오

Source code :

```
// source code 의 폰트는 Courier10 BT Bold으로 하시오
import pandas as pd
(1)
cars= pd.read_csv('d:/data/dataset_0914/cars.csv')
cars.head(5)
(2)
cars.columns
(3)
cars.iloc[:,1]
(4)
cars.iloc[10:20,0]
(5)
speed = cars['speed'] >= 20
cars[speed]
(6)
speed_more_than_10 = cars['speed']>10
dist_more_than_50 = cars['dist'] > 50
cars[speed_more_than_10 & dist_more_than_50]
(7)
speed_more_than_15 = cars['speed'] > 15
len(cars[speed_more_than_15 & dist_more_than_50])
```

실행화면 캡쳐:

```
In [22]: import pandas as pd

In [23]: cars= pd.read_csv('d:/data/dataset_0914/cars.csv')

In [24]: cars.head(5)
Out[24]:
   speed  dist
0      4     2
1      4    10
2      7     4
3      7    22
4      8    16

In [25]: cars.columns
Out[25]: Index(['speed', 'dist'], dtype='object')
```

```
In [27]: cars.iloc[:,1]
Out[27]:
0       2
1      10
2       4
3      22
4      16
5      10
6      18
7      26
8      34
9      17
10     28
11     14
12     20
13     24
14     28
15     26
16     34
17     34
18     46
19     26
20     36
21     60
22     80
23     20
24     26
25     54
26     32
27     40
28     32
29     40
30     50
31     42
32     56
33     76
34     84
35     36
36     46
37     68
38     32
39     48
40     52
41     56
42     64
43     66
44     54
45     70
46     92
```

```
47     93
48    120
49     85
Name: dist, dtype: int64
```

```
In [30]: cars.iloc[10:20,0]
Out[30]:
10    11
11    12
12    12
13    12
14    12
15    13
16    13
17    13
18    13
19    14
Name: speed, dtype: int64
```

```
In [31]: speed = cars['speed'] >= 20
    ...: cars[speed]
Out[31]:
    speed  dist
38     20    32
39     20    48
40     20    52
41     20    56
42     20    64
43     22    66
44     23    54
45     24    70
46     24    92
47     24    93
48     24   120
49     25    85
```

```
In [32]: speed_more_than_10 = cars['speed']>10
    ...: dist_more_than_50 = cars['dist'] > 50
    ...: cars[speed_more_than_10 & dist_more_than_50]
Out[32]:
    speed  dist
21     14    60
22     14    80
25     15    54
32     18    56
33     18    76
34     18    84
37     19    68
40     20    52
41     20    56
42     20    64
43     22    66
44     23    54
45     24    70
46     24    92
47     24    93
48     24   120
49     25    85

In [34]: speed_more_than_15 = cars['speed'] > 15
    ...: len(cars[speed_more_than_15 & dist_more_than_50])
Out[34]: 14
```