Artificial Neural Network

- AI 스터디 1주차

2024. 03. 31.

01 AI 스터디 방향성

I. ANN & 진행 방식 소개

1. 진행방식 조율
2. Dataset 형식에 대한 이해
3. Python 으로 모델 구조 직접 구현
4. Dataset 직접 제작, 원하는 형식의 Output
5. MLP 응용을 통한 여러 Experiments

II. CNN

1. Image Dataset 직접 제작, 원하는 형식의 Ouput Matrix
2. Roboflow 를 활용한 Dataset 제작
3. Yolo 모델 응용을 통한 Experiments (Object detection, Segmentation, Tracking

III. RNN

1. Value Prediction & Text Analysis Experiments
2. Kaggle & Sota 응용
    1. 1 file 코드 실행법,
    2. many folder 코드 실행법

IV. GAN

1. GAN, CGAN, DCGAN, Pix2Pix, CycleGAN, StarGAN, Inpainting with GAN
2. GAN 모델을 활용한 서비스 개발 시작

V. Autonomous Learning of Learned Humans

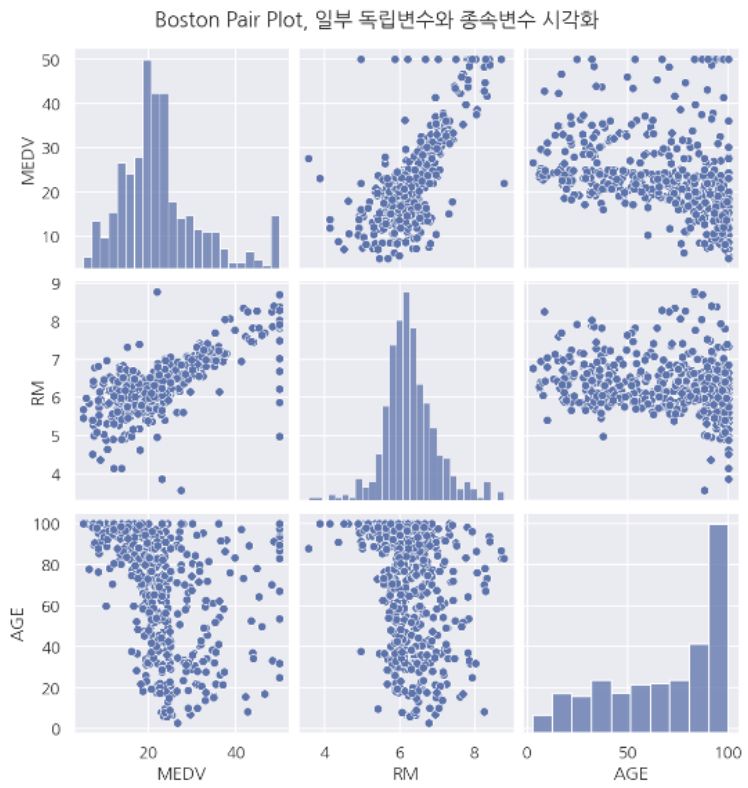
1. 각자의 Title Research 을 통한 스터디 방향성 전환
2. Model Structure Project

02 ANN – 데이터 형식에 대한 이해

- **데이터 입력 형태**를 알고, **모델 전반적인 구조**를 알고, **결과를 마음대로** 다룰 수 있어야 합니다.

Boston Housing Price

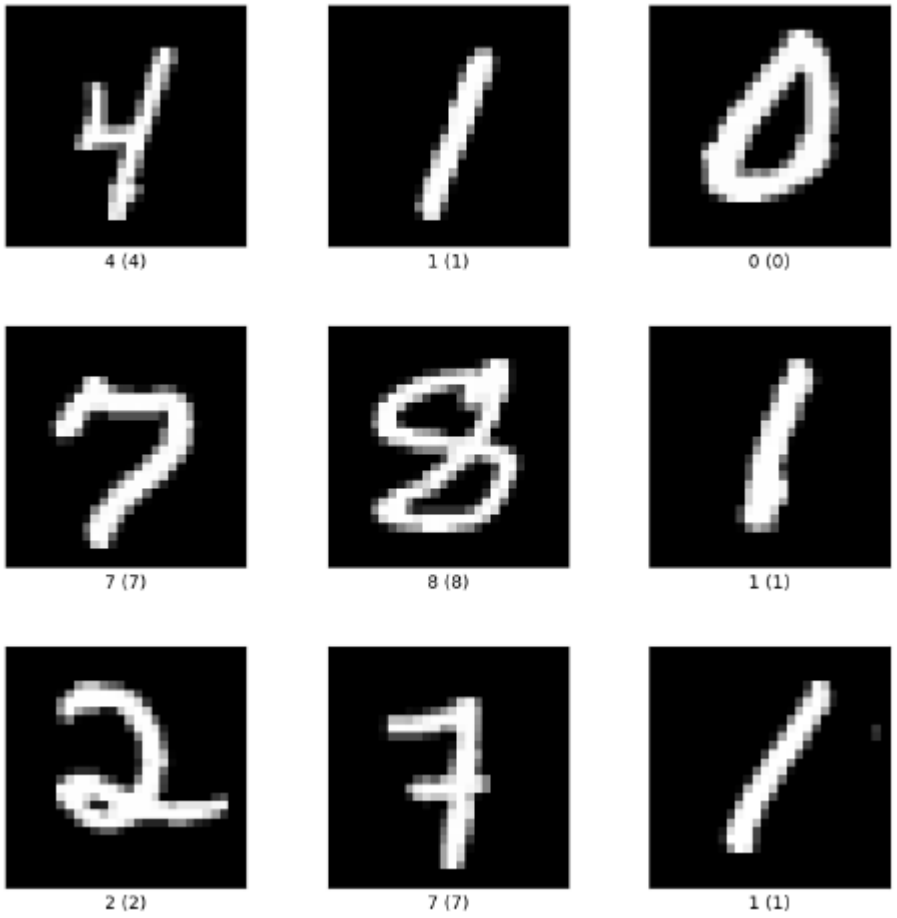
1	2	3	4	5	6	7	8	9	10	11	12	13	14
CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
범죄율			강변		평균 방 수	노후주택 비율			재산세 세율	학생/교사 비율		하위계층 비율	집값
0.00632	18	2.31	0	0.538	6.575	65.2	4.09	1	296	15.3	396.9	4.98	24
0.02731	0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17.8	396.9	9.14	21.6
0.02729	0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7
0.03237	0	2.18	0	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4
0.06905	0	2.18	0	0.458	7.147	54.2	6.0622	3	222	18.7	396.9	5.33	36.2
0.02985	0	2.18	0	0.458	6.43	58.7	6.0622	3	222	18.7	394.12	5.21	28.7
0.08829	12.5	7.87	0	0.524	6.012	66.6	5.5605	5	311	15.2	395.6	12.43	22.9
0.14455	12.5	7.87	0	0.524	6.172	96.1	5.9505	5	311	15.2	396.9	19.15	27.1
1.23247	0	8.14	0	0.538	6.142	91.7	3.9769	4	307	21	396.9	18.72	15.2
0.17004	12.5	7.87	0	0.524	6.004	85.9	6.5921	5	311	15.2	386.71	17.1	18.9
0.22489	12.5	7.87	0	0.524	6.377	94.3	6.3467	5	311	15.2	392.52	20.45	15
8.98296	0	18.1	1	0.77	6.212	97.4	2.1222	24	666	20.2	377.73	17.6	17.8

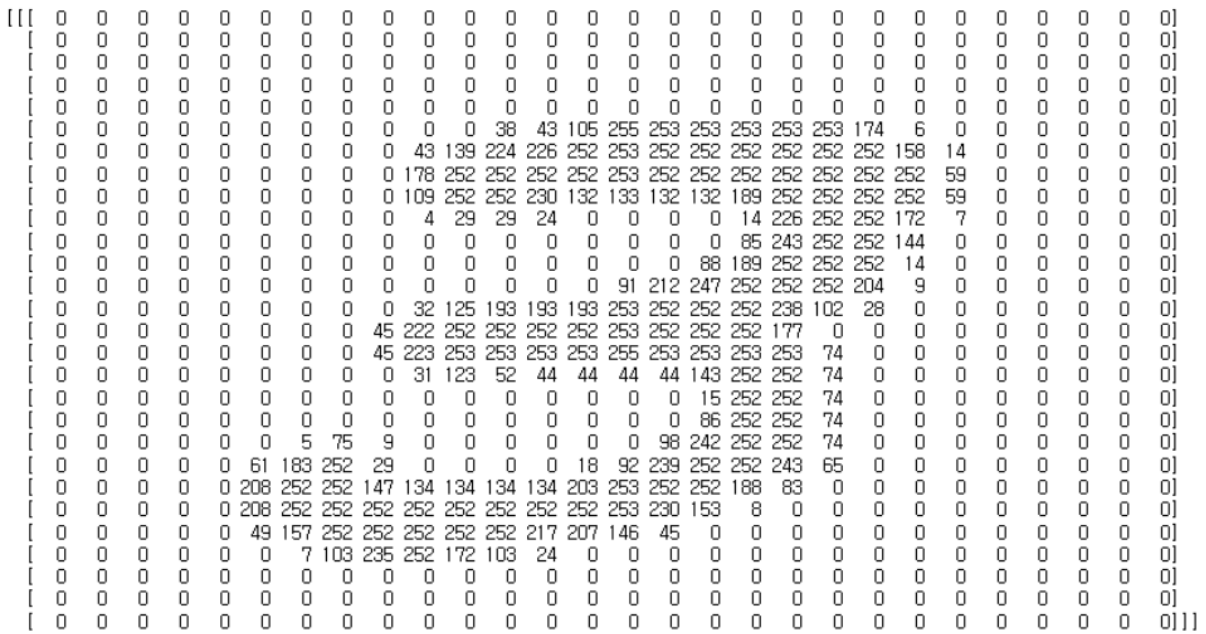


Price[i]

02 ANN – 데이터 형식에 대한 이해

- **데이터 입력 형태**를 알고, **모델 전반적인 구조**를 알고, **결과를 마음대로** 다룰 수 있어야 합니다.





Img[i][j]

02 ANN – 데이터 형식에 대한 이해

- 데이터 입력 형태를 알고, 모델 전반적인 구조를 알고, 결과를 마음대로 다룰 수 있어야 합니다.

```
class ANNModel(nn.Module):

    def __init__(self, input_dim, hidden_dim, output_dim):
        super(ANNModel, self).__init__()
        self.fc1 = nn.Linear(input_dim, hidden_dim)
        self.relu1 = nn.ReLU()
        self.fc2 = nn.Linear(hidden_dim, hidden_dim)
        self.tanh2 = nn.Tanh()
        self.fc3 = nn.Linear(hidden_dim, hidden_dim)
        self.elu3 = nn.ELU()
        self.fc4 = nn.Linear(hidden_dim, output_dim)

    def forward(self, x):
        out = self.fc1(x)
        out = self.relu1(out)
        out = self.fc2(out)
        out = self.tanh2(out)
        out = self.fc3(out)
        out = self.elu3(out)
        out = self.fc4(out)
        return out
```

입력 데이터의 차원 (intput dim)

출력 데이터의 차원 (output dim)

입력 차원을 임의로 바꾸어 학습도 가능!

➔ 2차원 배열의 mnist 데이터를 1차원 배열로 만들어 ANN 학습

➔ Multi Modal 의 핵심 (데이터의 형식의 자유)

03 ANN – 데이터 형식에 대한 이해

- **데이터셋의 형태**에 따른 선형 & 비선형 회귀

X = 실수 데이터 1종류
Y = 실수 데이터 1종류

	A	B	C	D
1	no	gender	weight_kg	height_cm
2	1	m	98	198
3	2	m	77	170
4	3	m	70	170
5	4	m	90	198
6	5	m	71	170
7	6	m	70	165
8	7	m	73	193
9	8	m	59	142
10	9	m	68	137
11	10	m	86	155
12	11	m	84	165
13	12	m	67	147
14	13	m	70	170
15	14	m	86	140
16	15	m	70	165
17	16	m	75	132
18	17	m	70	137
19	18	m	68	165
20	19	m	68	165
21	20	m	77	180
22	21	m	64	168
23	22	m	86	142

X = 실수 데이터 N종류
Y = 실수 데이터 1종류

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	AGE	B	RM	CRIM	DIS	INDUS	LSTAT	NOX	PTRATIO	RAD	ZN	TAX	CHAS	Target
2	65.2	396.9	6.575	0.00632	4.09	2.31	4.98	0.538	15.3	1	18	296	0	24
3	78.9	396.9	6.421	0.02731	4.9671	7.07	9.14	0.469	17.8	2	0	242	0	21.6
4	61.1	392.83	7.185	0.02729	4.9671	7.07	4.03	0.469	17.8	2	0	242	0	34.7
5	45.8	394.63	6.998	0.03237	6.0622	2.18	2.94	0.458	18.7	3	0	222	0	33.4
6	54.2	396.9	7.147	0.06905	6.0622	2.18	5.33	0.458	18.7	3	0	222	0	36.2
7	58.7	394.12	6.43	0.02985	6.0622	2.18	5.21	0.458	18.7	3	0	222	0	28.7
8	66.6	395.6	6.012	0.08829	5.5605	7.87	12.43	0.524	15.2	5	12.5	311	0	22.9
9	96.1	396.9	6.172	0.14455	5.9505	7.87	19.15	0.524	15.2	5	12.5	311	0	27.1
10	100	386.63	5.631	0.21124	6.0821	7.87	29.93	0.524	15.2	5	12.5	311	0	16.5
11	85.9	386.71	6.004	0.17004	6.5921	7.87	17.1	0.524	15.2	5	12.5	311	0	18.9
12	94.3	392.52	6.377	0.22489	6.3467	7.87	20.45	0.524	15.2	5	12.5	311	0	15
13	82.9	396.9	6.009	0.11747	6.2267	7.87	13.27	0.524	15.2	5	12.5	311	0	18.9
14	39	390.5	5.889	0.09378	5.4509	7.87	15.71	0.524	15.2	5	12.5	311	0	21.7
15	61.8	396.9	5.949	0.62976	4.7075	8.14	8.26	0.538	21	4	0	307	0	20.4
16	84.5	380.02	6.096	0.63796	4.4619	8.14	10.26	0.538	21	4	0	307	0	18.2
17	56.5	395.62	5.834	0.62739	4.4986	8.14	8.47	0.538	21	4	0	307	0	19.9
18	29.3	386.85	5.935	1.05393	4.4986	8.14	6.58	0.538	21	4	0	307	0	23.1
19	81.7	386.75	5.99	0.7842	4.2579	8.14	14.67	0.538	21	4	0	307	0	17.5
20	36.6	288.99	5.456	0.80271	3.7965	8.14	11.69	0.538	21	4	0	307	0	20.2
21	69.5	390.95	5.727	0.7258	3.7965	8.14	11.28	0.538	21	4	0	307	0	18.2
22	98.1	376.57	5.57	1.25179	3.7979	8.14	21.02	0.538	21	4	0	307	0	13.6
23	89.2	392.53	5.965	0.85204	4.0123	8.14	13.83	0.538	21	4	0	307	0	19.6
24	91.7	396.9	6.142	1.23247	3.9769	8.14	18.72	0.538	21	4	0	307	0	15.2
25	100	394.54	5.813	0.98843	4.0952	8.14	19.88	0.538	21	4	0	307	0	14.5
26	94.1	394.33	5.924	0.75026	4.3996	8.14	16.3	0.538	21	4	0	307	0	15.6
27	85.7	303.42	5.599	0.84054	4.4546	8.14	16.51	0.538	21	4	0	307	0	13.9
28	90.3	376.88	5.813	0.67191	4.682	8.14	14.81	0.538	21	4	0	307	0	16.6
29	88.8	306.38	6.047	0.95577	4.4534	8.14	17.28	0.538	21	4	0	307	0	14.8

X = 실수 데이터 N종류
Y = N

	A	B	C	D	E	F
1	caseno	SepalLeng	SepalWidt	PetalLeng	PetalWidt	Species
2	1	5.1	3.5	1.4	0.2	setosa
3	2	4.9	3	1.4	0.2	setosa
4	3	4.7	3.2	1.3	0.2	setosa
5	4	4.6	3.1	1.5	0.2	setosa
6	5	5	3.6	1.4	0.2	setosa
7	6	5.4	3.9	1.7	0.4	setosa
8	7	4.6	3.4	1.4	0.3	setosa
9	8	5	3.4	1.5	0.2	setosa
10	9	4.4	2.9	1.4	0.2	setosa
11	10	4.9	3.1	1.5	0.1	setosa
12	11	5.4	3.7	1.5	0.2	setosa
13	12	4.8	3.4	1.6	0.2	setosa
14	13	4.8	3	1.4	0.1	setosa
15	14	4.3	3	1.1	0.1	setosa
16	15	5.8	4	1.2	0.2	setosa
17	16	5.7	4.4	1.5	0.4	setosa
18	17	5.4	3.9	1.3	0.4	setosa
19	18	5.1	3.5	1.4	0.3	setosa
20	19	5.7	3.8	1.7	0.3	setosa
21	20	5.1	3.8	1.5	0.3	setosa
22	21	5.4	3.4	1.7	0.2	setosa
23	22	5.1	3.7	1.5	0.4	setosa
24	23	4.6	3.6	1	0.2	setosa

03 ANN – Python 으로 모델 구조 직접 구현 + Dataset 직접 제작, 원하는 형식의 Output

- Jupyter notebook 으로 실습

04 MLP 응용을 통한 여러 Experiments

- **HW1. 다음과 같은 조건으로 Tensorflow 모델을 만들어 학습 시키고 학습 로그 공유하기**

1. 테스트, 훈련 데이터 셋을 9:1 (학습:테스트) 로 하여 학습 진행
2. Layer 를 6개로 만들기
3. Node 를 16으로 시작해서 피라미드 구조로 구성 (16, 32 ... 32, 16)
4. Activation function 을 relu 보다 더 좋은 함수를 사용해보기
5. optimizer 를 adam 이 아닌 다른 함수를 이용해보기
6. loss 의 경우 mse 가 아닌 다른 수식으로 나타내보기 (이에 따른 metrics 수정 필요)
7. epoch 를 조정해가며 최적의 loss 찾아보기
8. 데이터셋 다른거 가져오기
9. 학습 로그 공유

```
Epoch 994/1000
13/13 [==============================] - 0s 759us/step - loss: 1.7453 - mean_squared_error: 1.7453
Epoch 995/1000
13/13 [==============================] - 0s 555us/step - loss: 2.2642 - mean_squared_error: 2.2642
Epoch 996/1000
13/13 [==============================] - 0s 546us/step - loss: 1.7498 - mean_squared_error: 1.7498
Epoch 997/1000
13/13 [==============================] - 0s 587us/step - loss: 2.7734 - mean_squared_error: 2.7734
Epoch 998/1000
13/13 [==============================] - 0s 555us/step - loss: 2.3631 - mean_squared_error: 2.3631
Epoch 999/1000
13/13 [==============================] - 0s 559us/step - loss: 2.6625 - mean_squared_error: 2.6625
Epoch 1000/1000
13/13 [==============================] - 0s 703us/step - loss: 2.2430 - mean_squared_error: 2.2430
4/4 [==============================] - 0s 749us/step
```

04 MLP 응용을 통한 여러 Experiments

- **실습1. Kaggle 실습**

1. Kaggle 실습 데이터 다운로드
2. Kaggle 코드 다운로드
3. Kaggle 전반적으로 실습해보기