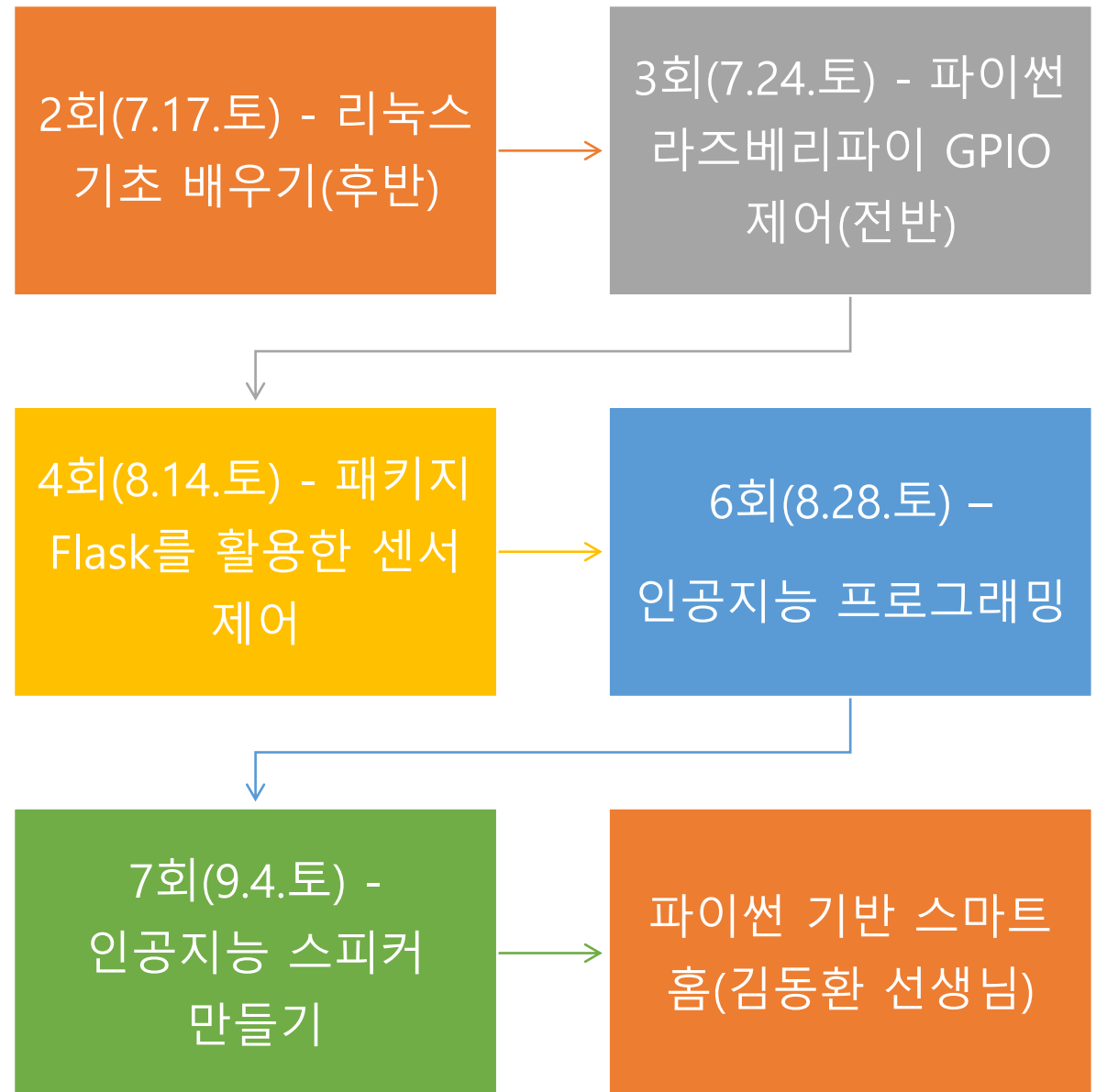


사물인터넷 과정 내용 일정 및 터넷

수행평가



IoT & 인공지능

- 0 - 음성 인식
- 1 - Tensorflow
- 2 - 딥러닝(DNN, CNN)

"헉...방금 통화한 상대가 AI?"...음성AI 두고
기술기업들 각축



라즈베리파이 + Google Assistant



Google Assistant 원격 음성 제어



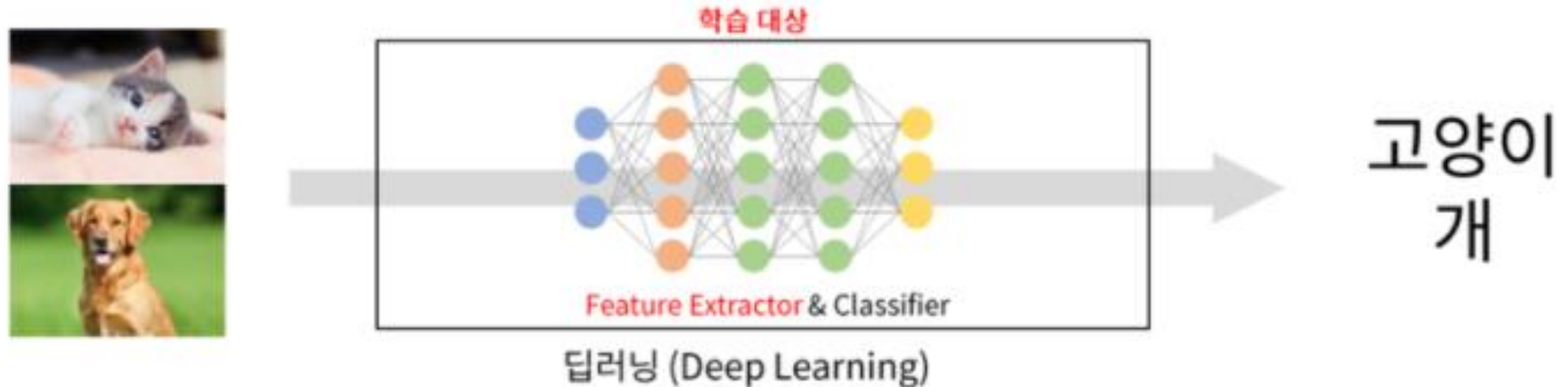
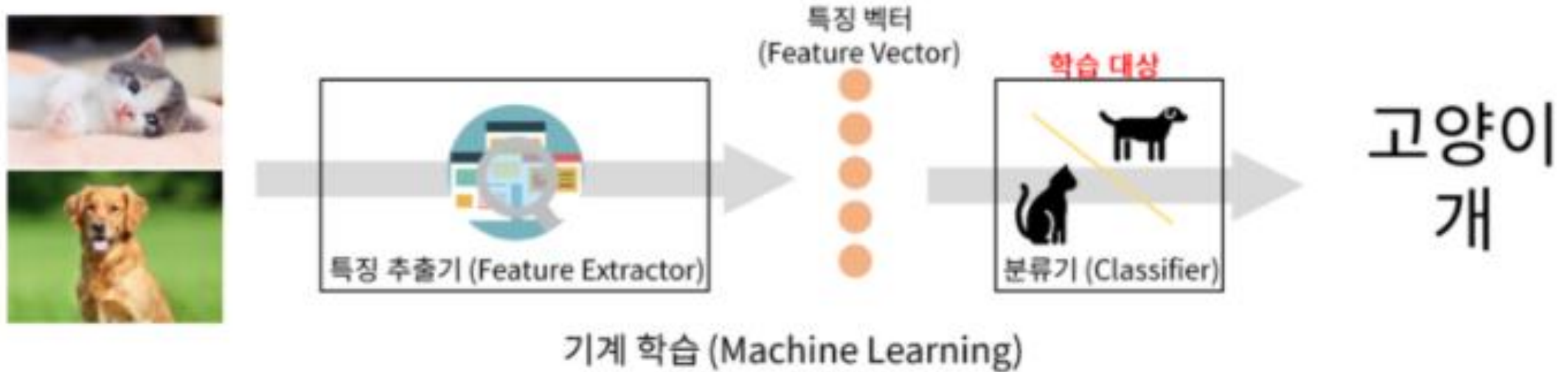
라즈베리파이를 이용한 원격 음성 제어 테스트

인공지능

- [Siri, 2011년](#) 10월 4일, [iPhone 4s](#)와 함께 발표
- 구글 웨이모 2012년 5월 8일에 구글은 처음으로 네바다주에서 시험 면허 획득
- 알파고 대 이세돌, 2016년 3월 9일부터 15일 알파고가 4승 1패로 이세돌에게 승리

전통적인 머신러닝 vs 딥러닝

스팸 메일 제목(광고, 할인)



딥러닝이 매우 잘하는 분야

- image classification
강아지 고양이
음성 인식
- object detection
자율주행
- sequence data
순서가 있는 데이터
번역기

수능 성적 예측



6월 (60점)

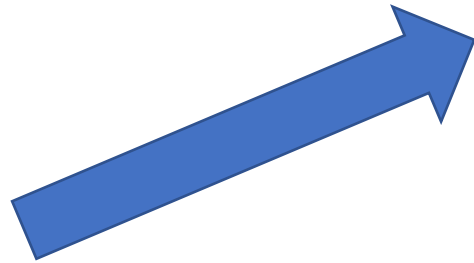
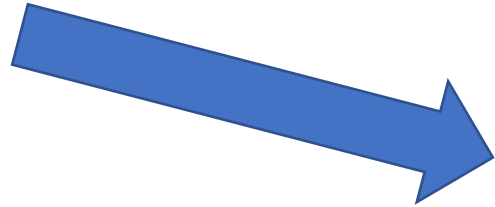
9월 (80점)

수능(?)

수능 성적 예측

6월 (60점)

9월 (80점)



인간러닝

(6월 성적, 9월 성적은 반반씩 영향있음)

$$= 6월 * 0.5 + 9월 * 0.5 = 70$$

$$= 6월 * w1 + 9월 * w2$$

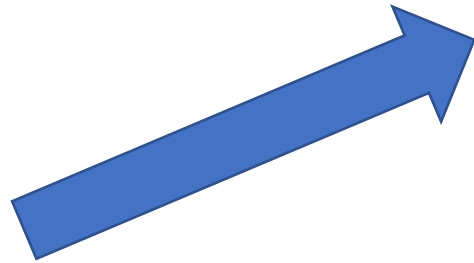
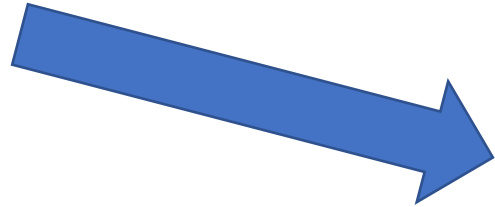
(w를 weight 가중치라 함)

수능(??)

수능 성적 예측

6월 (60점)

9월 (80점)



머신러닝

데이터를 주면서 w_1, w_2 찾아주라 함.

$$6월 * w_1 + 9월 * w_2 + b$$

보통 b 를 더하는데, 6, 9월 성적과 관련은 업는데 수능 성적에 관련 있는 요소.

bias 편향이라고도 함.

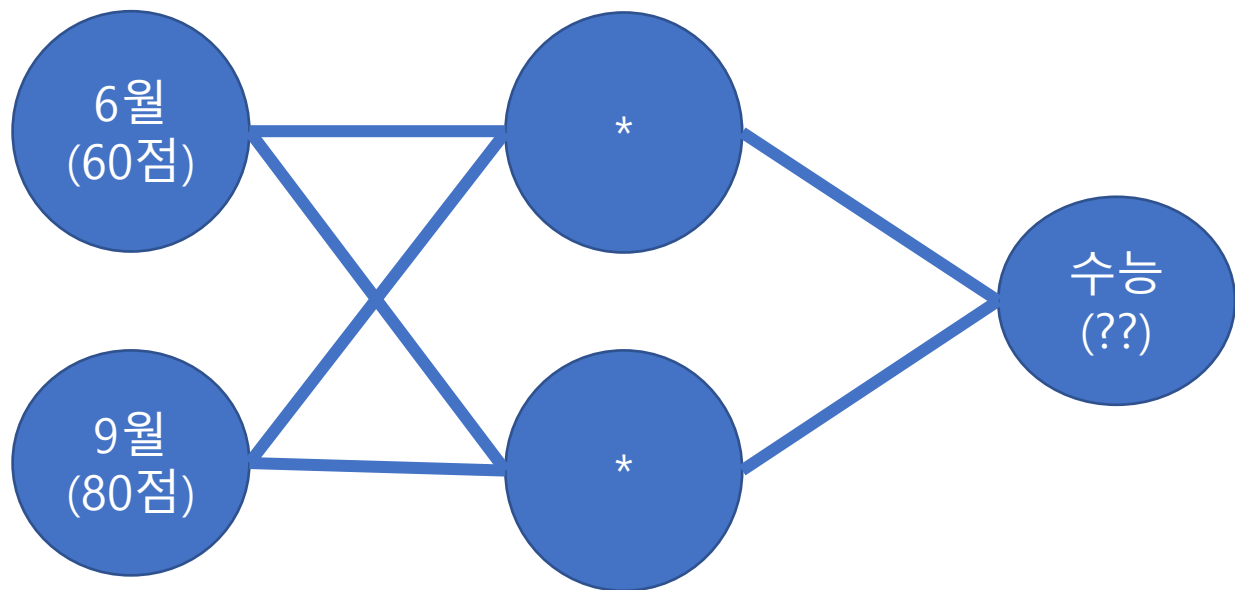
수능(??)

수능 성적 예측

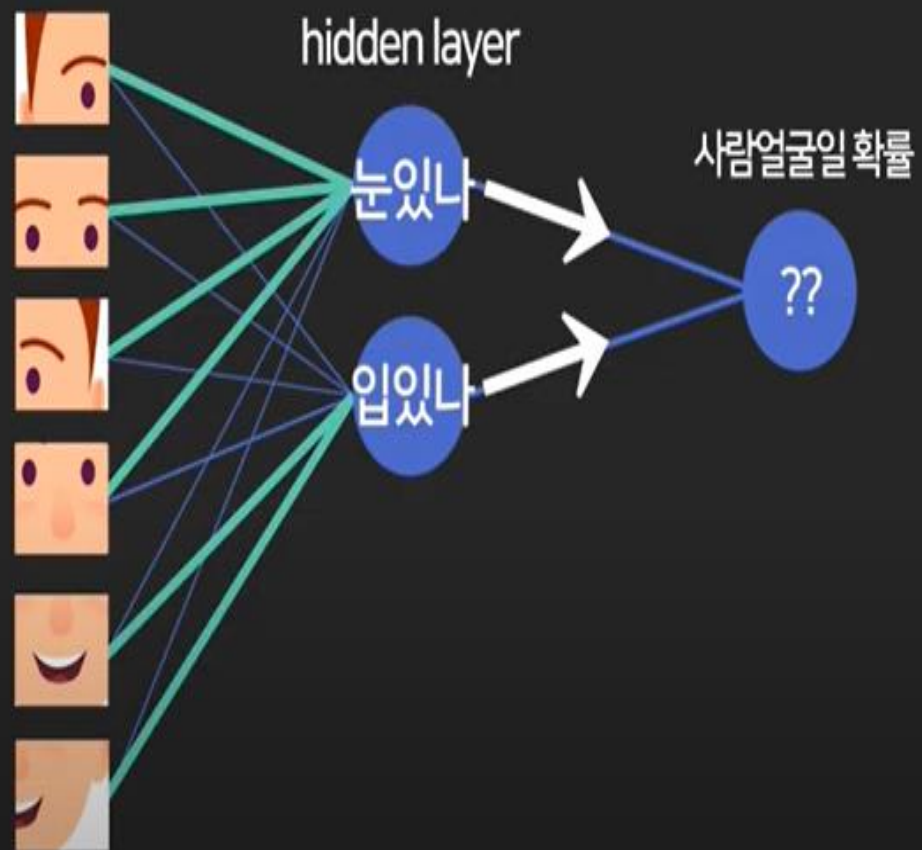
딥러닝

레이어를 여러개 만들면 더 정교하게 예측할 수 있음

중간 hidden layer



이 사진은 사람얼굴일까?



차 사진을 신경망에 집어넣었더니..



layer1



직선, 곡선

layer2



layer3

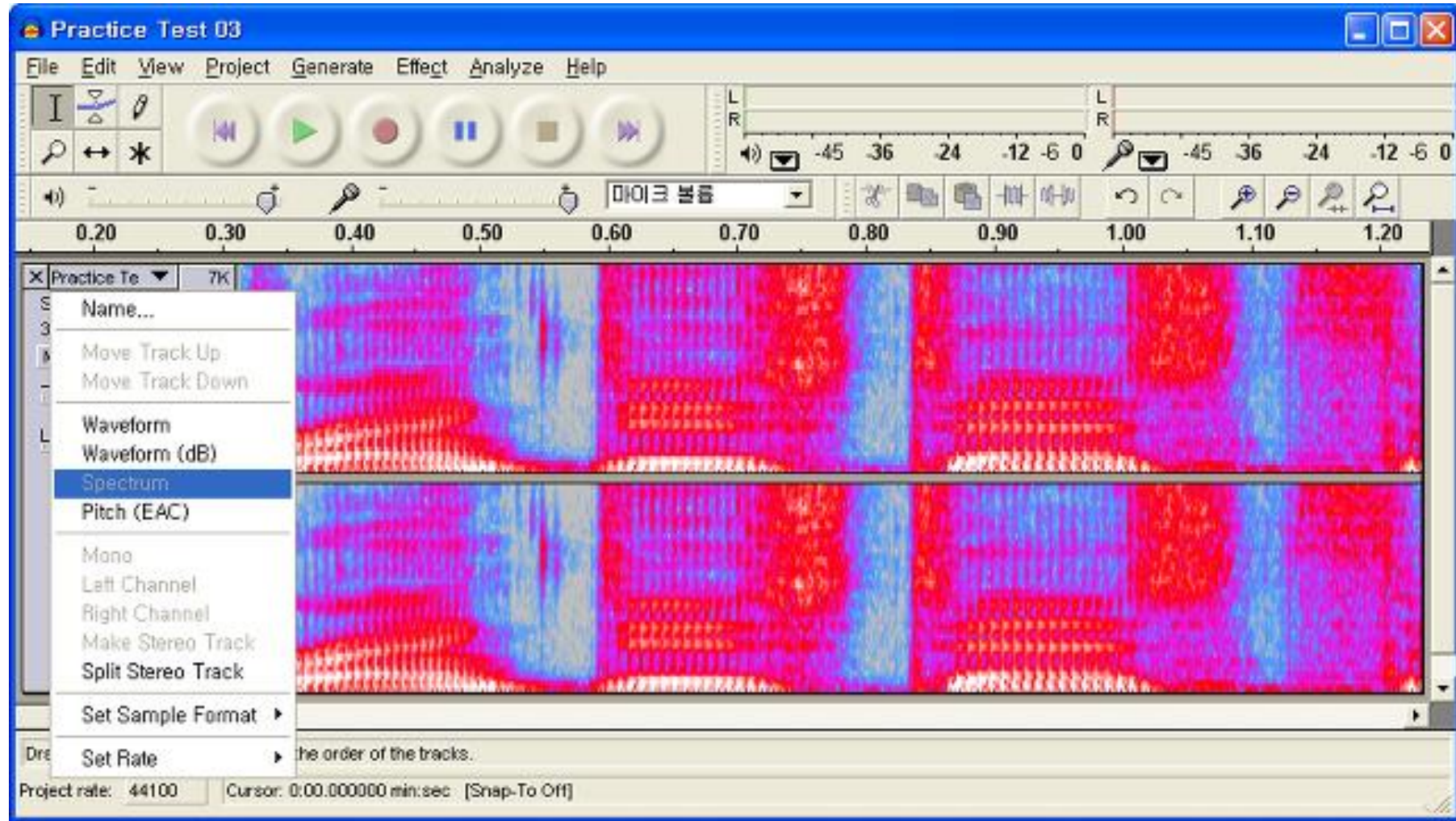


▲ 이거 바퀴

바퀴, 백미러, 손

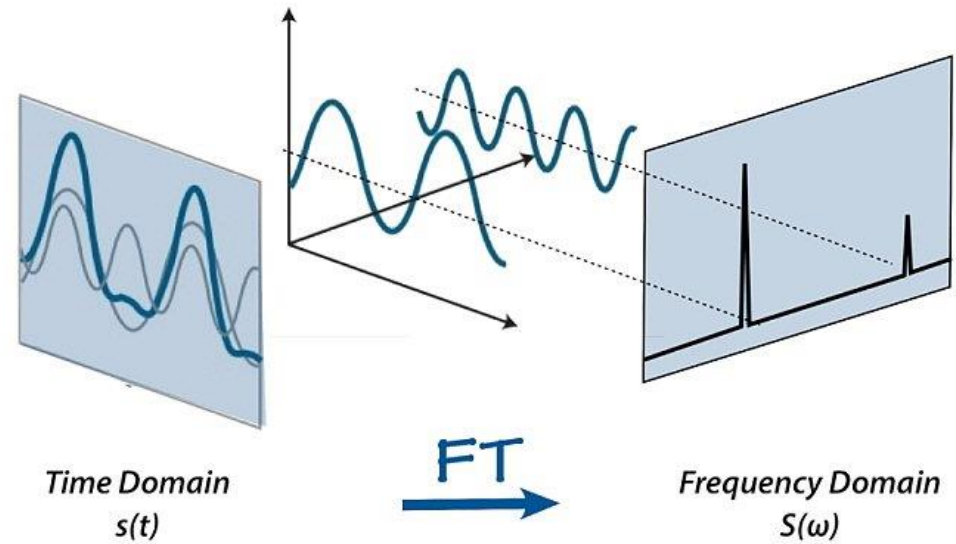
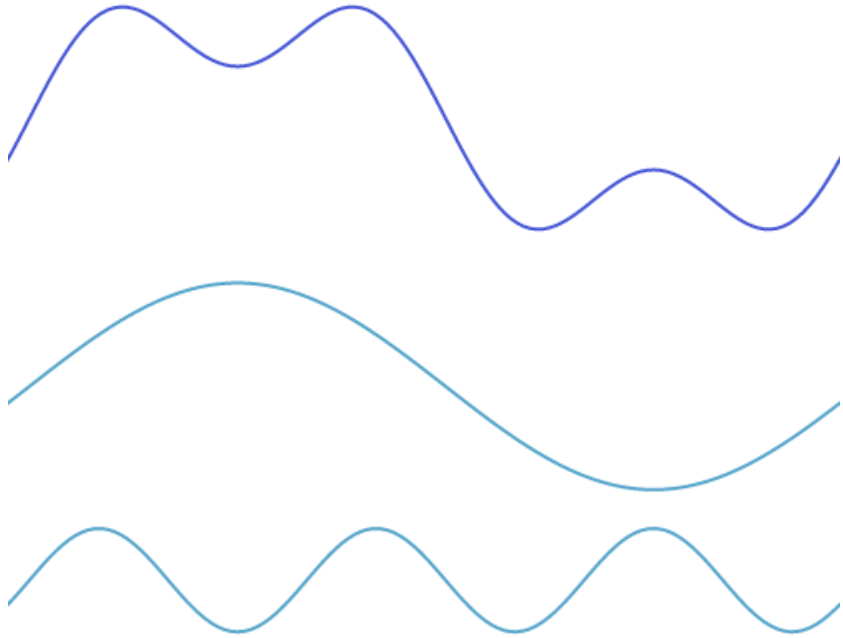
PCM(표본화-양자화-부호화)

음성 스펙트럼, 피치(with audacity)



푸리에 변환

- <https://www.jezzamon.com/fourier/ko.html>

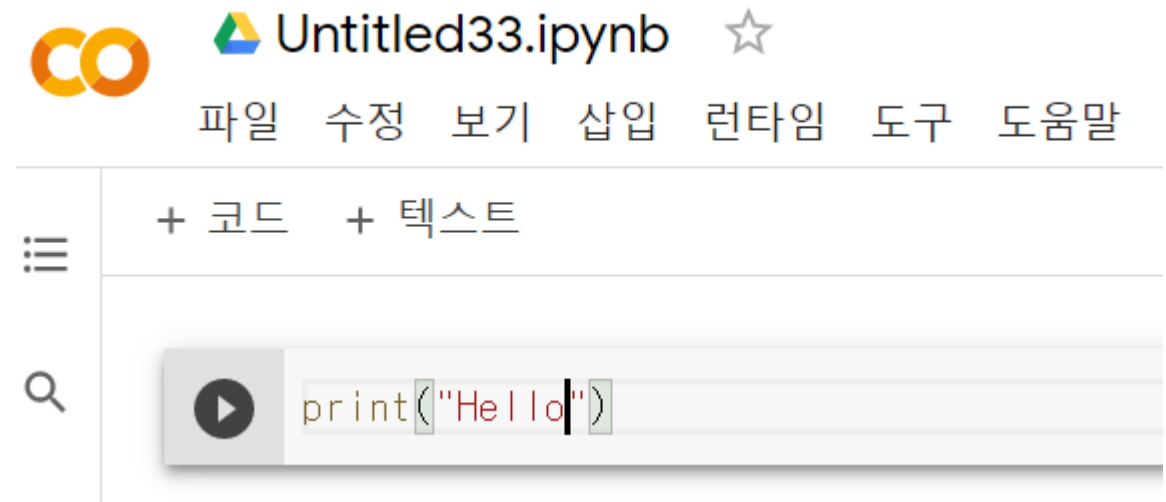
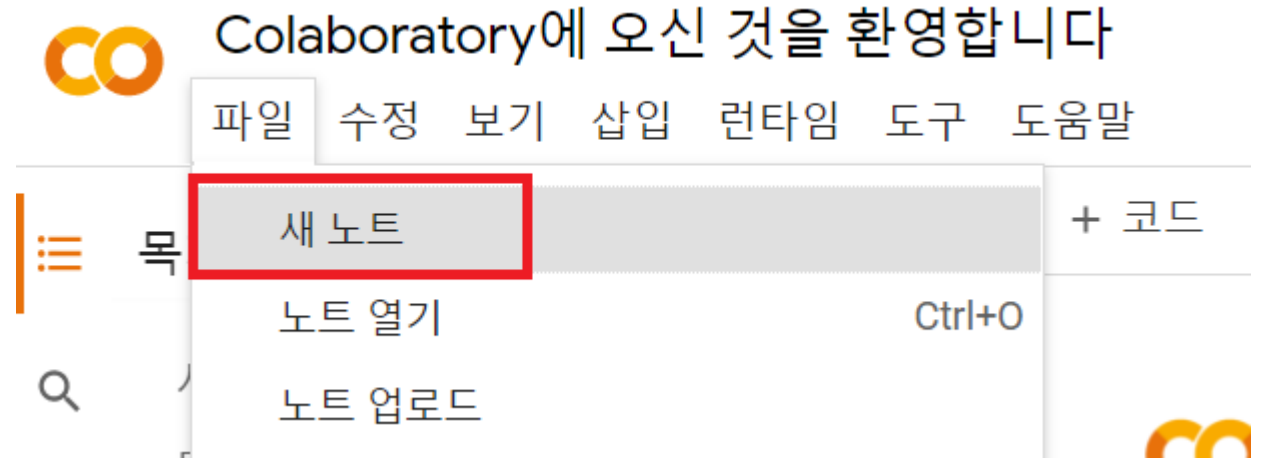


Google Colab

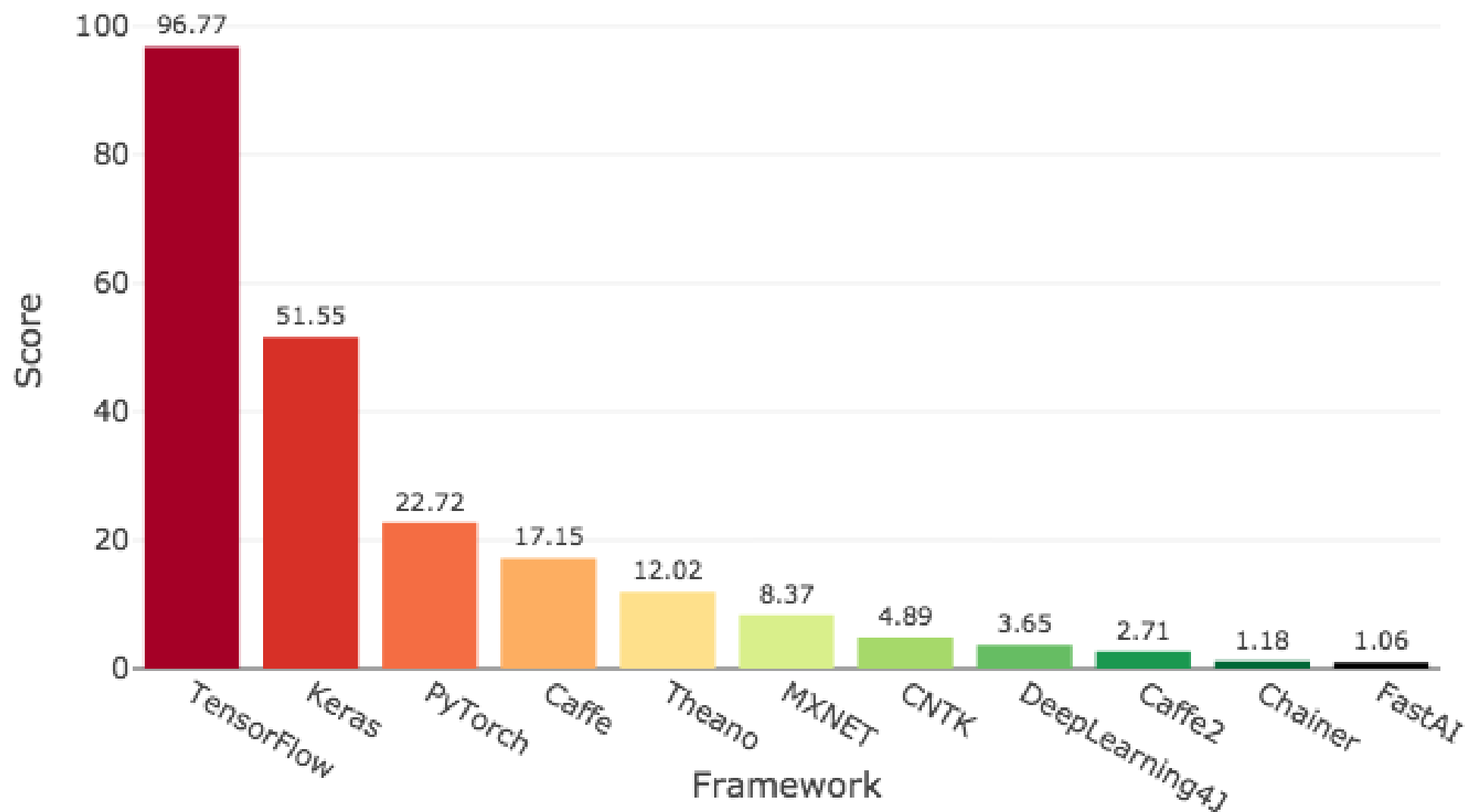
- 별도의 파이썬 설치가 필요 없습니다.
- 데이터에 분석 사용되는 Tensor Flow, Keras, matplotlib, scikit-learn, pandas와 같은 패키지가 기본적으로 설치되어 있습니다.
- GPU를 무료로 사용 가능합니다!
- Jupyter 노트북과 비슷하지만 더 좋은 기능을 제공합니다.
- 깃과 연동이 가능하여 사람들과 협업하여 코딩이 가능합니다.

코랩 사용하기

- 구글에서 colab 검색하여 접속
- 필요한 경우 로그인 합니다.
- 왼쪽 상단에 "새 노트 " 를 클릭합니다.
- `print("Hello")` 를 입력하고 실행해봅니다.

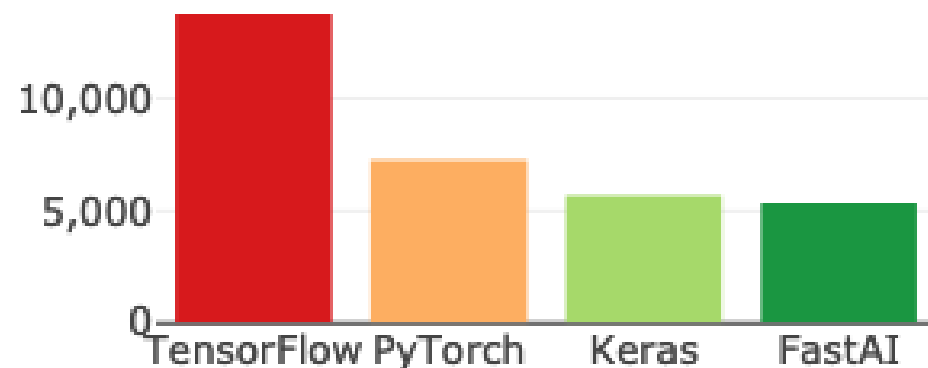


Deep Learning Framework Power Scores 2018

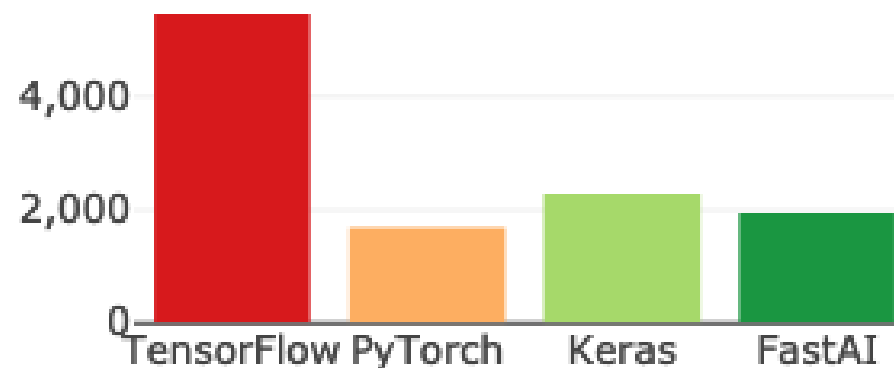


New GitHub Activity

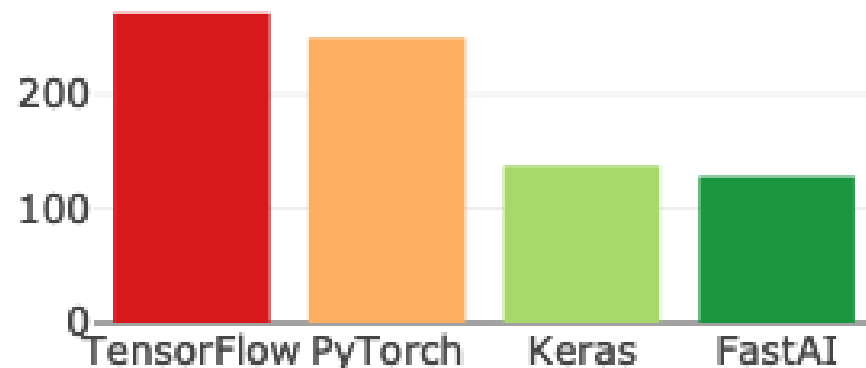
Stars



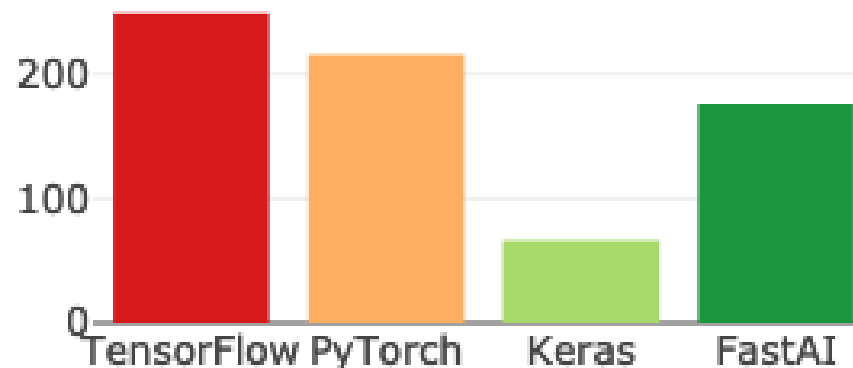
Forks



Watchers



Contributors



딥러닝 모델 설계하기



모델 설정



모델 컴파일



학습



예측

딥러닝 (목차)

회귀 모델

- 단순 선형회귀
- 다중 선형회귀

분류 모델

- 이진 분류
- 다중 분류

과적합 방지하기

학습 자동 중단하기

이미지 분류 하기

중간고사 성적 예측하기(with 공부시간)

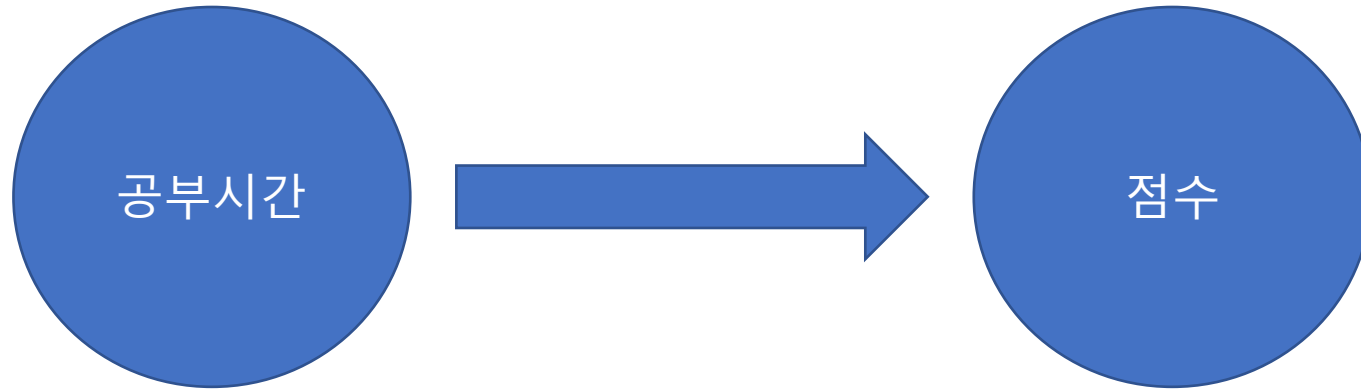
```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense

time = [2, 4, 6, 8]
score = [81, 88, 93, 97]

model = Sequential(
    Dense(1, input_dim = 1)
)
model.compile(optimizer='sgd', loss='mse', metrics=['mse', 'mae'])
model.fit(time, score, epochs = 5)

model.predict([5])
```


중간고사(선형 회귀)

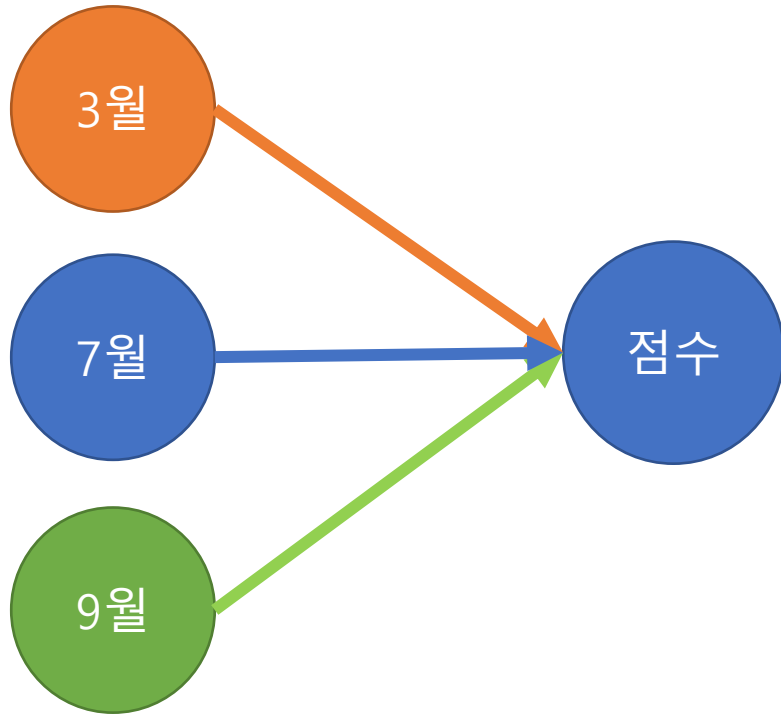


```
model = Sequential(  
    Dense(1, input_dim = 1)  
)
```

수능 점수 예측하기(다중 선형회귀)

3월 모의고사	7월 모의 고사	9월 모의고사	수능
73	80	75	76
93	88	93	93
89	91	90	90
96	98	100	98
73	66	70	71

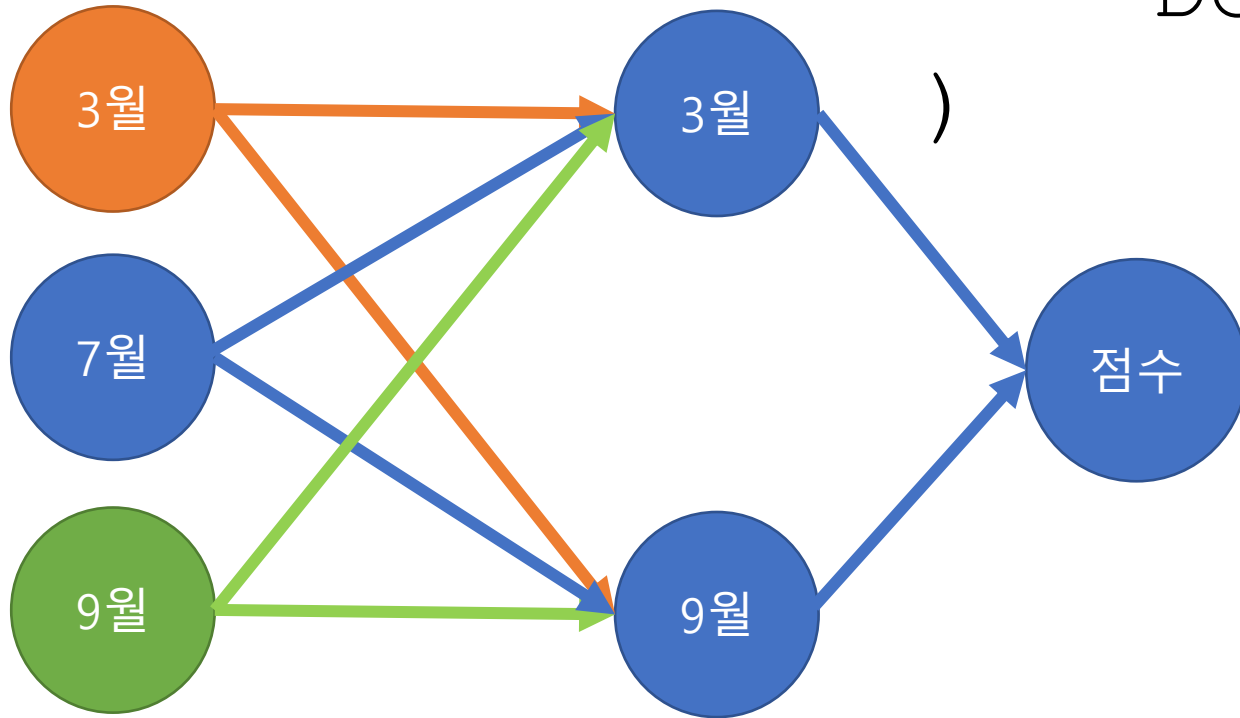
중간고사



```
model = Sequential(  
    Dense(1, input_dim = 3)  
)
```

중간고사

```
model = Sequential(  
    Dense(2, input_dim = 3)  
    Dense(1)  
)
```



```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
```

```
scores = [[73, 80, 75],
           [93, 88, 93],
           [89, 91, 90],
           [96, 98, 100],
           [73, 66, 70]]
```

```
final = [[76], [93], [90], [98], [71]]
```

```
model = Sequential()
```

```
model.add(Dense(1, input_dim=3))
```

```
model.compile(optimizer='adam', loss='mse', metrics=['mse', 'mae'])
```

```
model.fit(scores, final, epochs = 5)
```

```
model.predict([[81, 85, 83]])
```

보스턴 집값 예측

1970년대 보스턴 지역의 범죄율, 토지 지역의 비율, 방의 개수 등 정답을 포함한 총 14개의 특성으로 이루어짐

	0	1	2	3	...	12	13
0	0.0063 2	18.0	2.31	0	...	4.98	24.0
1	0.0273 1	0	7.07	0	...	9.14	21.6
2	0.0272 9	0	7.07	0	...	4.03	34.7
3	0.0323 7	0	2.18	0	...	2.94	33.4
4	0.0690 5	0	2.18	0	...	5.33	36.2

0	CRIM: 인구 1인당 범죄 발생 수	7	DIS: 5가지 보스턴 시 고용 시설까지의 거리
1	ZN: 25,000 평방 피트 이상의 주거 구역 비중	8	RAD: 순환고속도로의 접근 용이성
2	INDUS: 소매업 외 상업이 차지하는 면적 비율	9	TAX: \$10,000당 부동산 세율 총계
3	CHAS: 찰스강 위치 변수(1: 강 주변, 0: 이외)	10	PTRATIO: 지역별 학생과 교사 비율
4	NOX: 일산화질소 농도	11	B: 지역별 흑인 비율
5	RM: 집의 평균 방 수	12	LSTAT: 급여가 낮은 직업에 종사하는 인구 비율(%)
6	AGE: 1940년 이전에 지어진 비율	13	가격(단위: \$1,000)

폐암 수술 환자의 생존율 예측하기

속성																		클래스
줄 항목	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	293	1	3.8	2.8	0	0	0	0	0	0	12	0	0	0	1	0	62	0
2	1	2	2.88	2.16	1	0	0	0	1	1	14	0	0	0	1	0	60	0
3	8	2	3.19	2.5	1	0	0	0	1	0	11	0	0	1	1	0	66	1
...
470	447	8	5.2	4.1	0	0	0	0	0	0	12	0	0	0	0	0	49	0

종양의 유형, 폐활량, 호흡 곤란 여부, 고통 정도, 기침, 흡연, 친식 여부 등 17개의 환자 상태
18번째 항목: 1 생존, 0 사망

데이터 읽어오기

필요한 라이브러리를 불러옵니다.

```
import numpy as np
```

```
import tensorflow as tf
```

실행할 때마다 같은 결과를 출력하기 위해 설정하는 부분입니다.

```
np.random.seed(3)
```

```
tf.random.set_seed(3)
```

준비된 수술 환자 데이터를 불러들입니다.

```
Data_set = np.loadtxt("ThoracicSurgery.csv", delimiter=",")
```

환자의 기록과 수술 결과를 X와 Y로 구분하여 저장합니다.

```
X = Data_set[:,0:17]
```

```
Y = Data_set[:,17]
```


모델설정, 컴파일, 훈련

딥러닝 구조를 결정합니다 (모델을 설정하고 실행하는 부분입니다).

```
model = Sequential()  
model.add(Dense(30, input_dim=17, activation='relu'))  
model.add(Dense(1, activation='sigmoid'))
```

딥러닝을 실행합니다.

```
model.compile(loss='binary_crossentropy', optimizer=  
    'adam', metrics=['accuracy'])  
model.fit(X, Y, epochs=100, batch_size=10)
```

파마 인디언 데이터 분석

- 비만은 유전 vs 식습관 조절
- 1950년 비만 없음
- 현재 60% 당뇨, 80% 비만

아이리스 품종(다중 분류 모델)



버시컬러

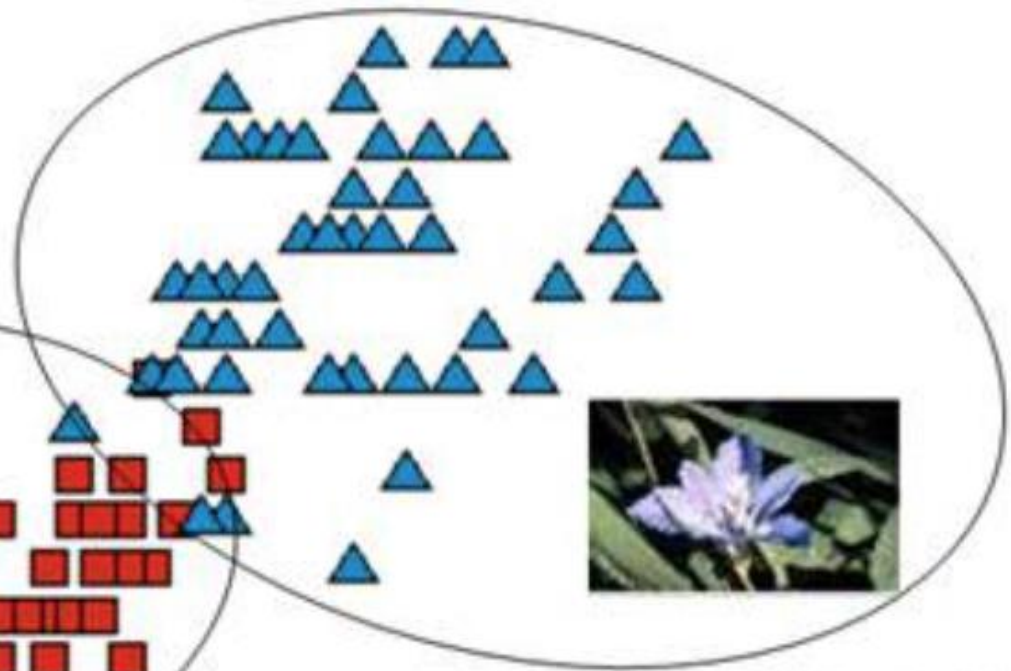
버지니카

세토사

꽃받침 길이, 꽃받침 폭, 꽃잎 길이, 꽃잎 폭

Petal width [cm]

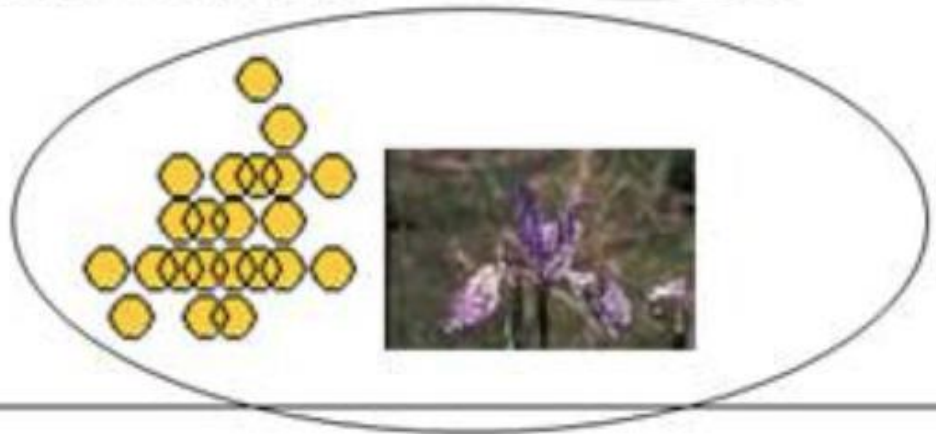
Iris Versicolour



Iris Virginica



Iris Setosa



Petal length [cm]

버시컬러
버지니카
세토사

꽃잎길이
꽃잎폭

베스트 모델 만들기

- 모델 체크포인트
- 자동 중단

컨볼루션 신경망(CNN)

1	0	1	0
0	1	1	0
0	0	1	1
0	0	1	0

Copyright © Gilbut, Inc. All rights reserved.

$\times 1$	$\times 0$
$\times 0$	$\times 1$

Copyright © Gilbut, Inc. All rights reserved.

$$(1 \times 1) + (0 \times 0) + (0 \times 0) + (1 \times 1) = 2$$

1×1	0×0	1	0
0×0	1×1	1	0
0	0	1	1
0	0	1	0

1	0×1	1×0	0
0	1×0	1×1	0
0	0	1	1
0	0	1	0

1	0	1×1	0×0
0	1	1×0	0×1
0	0	1	1
0	0	1	0

1	0	1	0
0×1	1×0	1	0
0×0	0×1	1	1
0	0	1	0
1	0	1	0
0	1×1	1×0	0
0	0×0	1×1	1
0	0	1	0
1	0	1	0
0	1	1	0
0	0×1	1×0	1
0	0×0	1×1	0
1	0	1	0
0	1	1	0
0	0	1×1	1×0
0	0	1×0	0×1

2	1	1
0	2	2
0	1	1

1	0	1	0
0	1	1	0
0	0	1	1
0	0	1	0

+

x1	x0
x0	x1

마스크 1

=

2	1	1
0	2	2
0	1	1

컨볼루션 1

1	0	1	0
0	1	1	0
0	0	1	1
0	0	1	0

+

x1	x1
x0	x0

마스크 2

=

1	1	1
1	2	1
0	1	2

컨볼루션 2

풀링(pooling), 서브 샘플링(sub sampling)

- 컨볼루션 층을 통한 이미지 특징을 다시 한번 축소
- 맥스 풀링(max pooling): 최댓값을 뽑아 내는 것

1	0	1	0
0	4	2	0
0	1	6	1
0	0	1	0

Copyright © Gilbut, Inc. All rights reserved.

1	0	1	0
0	4	2	0
0	1	6	1
0	0	1	0

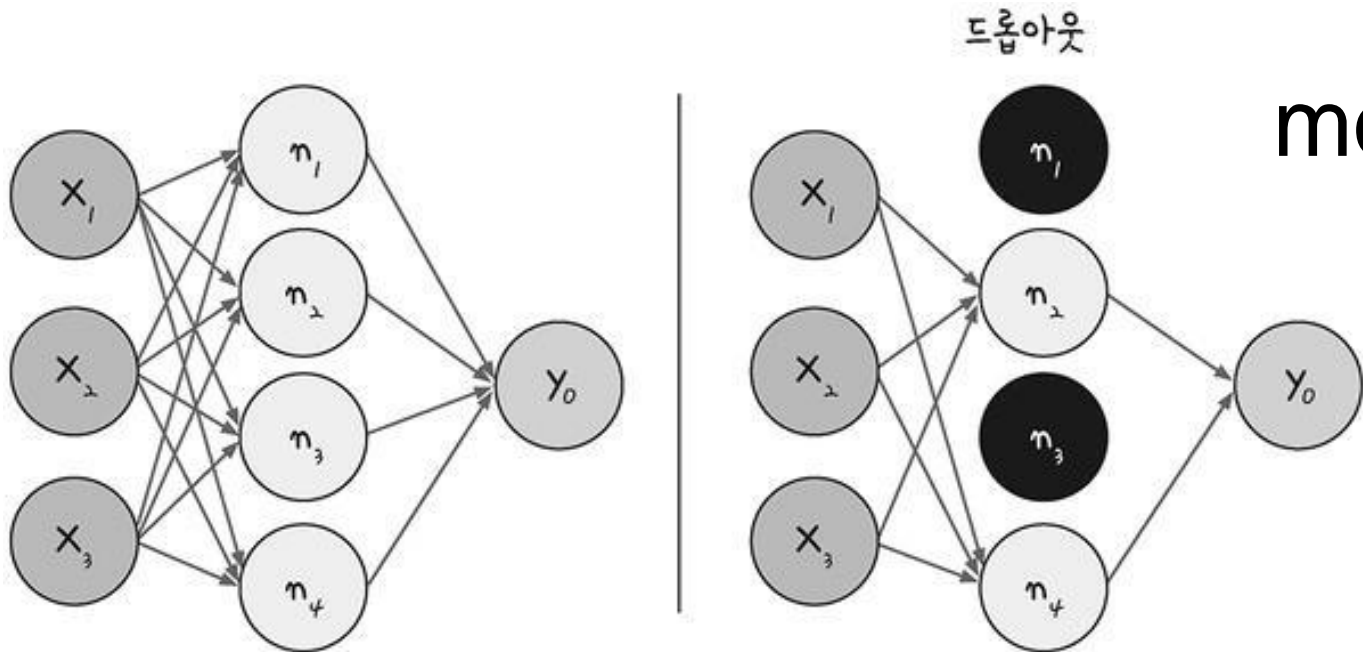
Copyright © Gilbut, Inc. All rights reserved.

4	2
1	6

Copyright © Gilbut, Inc. All rights reserved.

드롭아웃

- 드롭아웃은 은닉층에 배치된 노드 중 일부를 임의로 꺼주는 것이다.
- 과적합 문제를 간단하지만 효과가 큰 기법.



`model.add(Dropout(0.25))`

플래튼 Flatten()

- 2차원 배열을 1차원으로 바꿔줌.
- Flatten() 과정을 지나 Dense() 층으로 연결.

`model.add(Flatten())`

4	2
1	6

Copyright © Gibut, Inc. All rights reserved.

4	2	1	6
---	---	---	---

Convolutional Neural Network

