

## Chapter.04

### 딥러닝 패키지 활용 시계열 예측

# 딥러닝과 Keras 패키지

FASTCAMPUS  
ONLINE

금융공학/퀀트 I

강사. 장순용

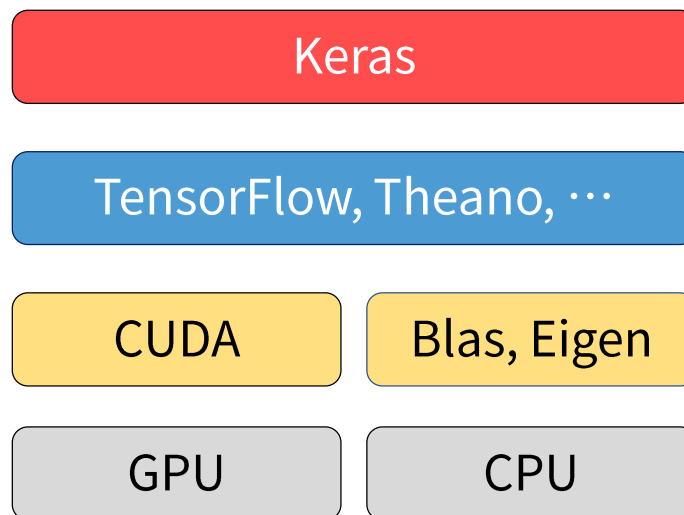
# I 키포인트

- Keras 와 TensorFlow.
- Keras 학습 순서.
- 비용함수/손실함수.
- 활성화 함수.
- 경사 하강법.
- 배치학습.



# I Keras에 대해서

- Keras는 모델 레벨의 하이엔드 딥러닝 라이브러리이다.
  - ⇒ 로우레벨 텐서 연산은 지원하지 않는다. 텐서  $\cong$  고차원 행렬.
  - ⇒ TensorFlow를 바탕으로 두고 실행된다.

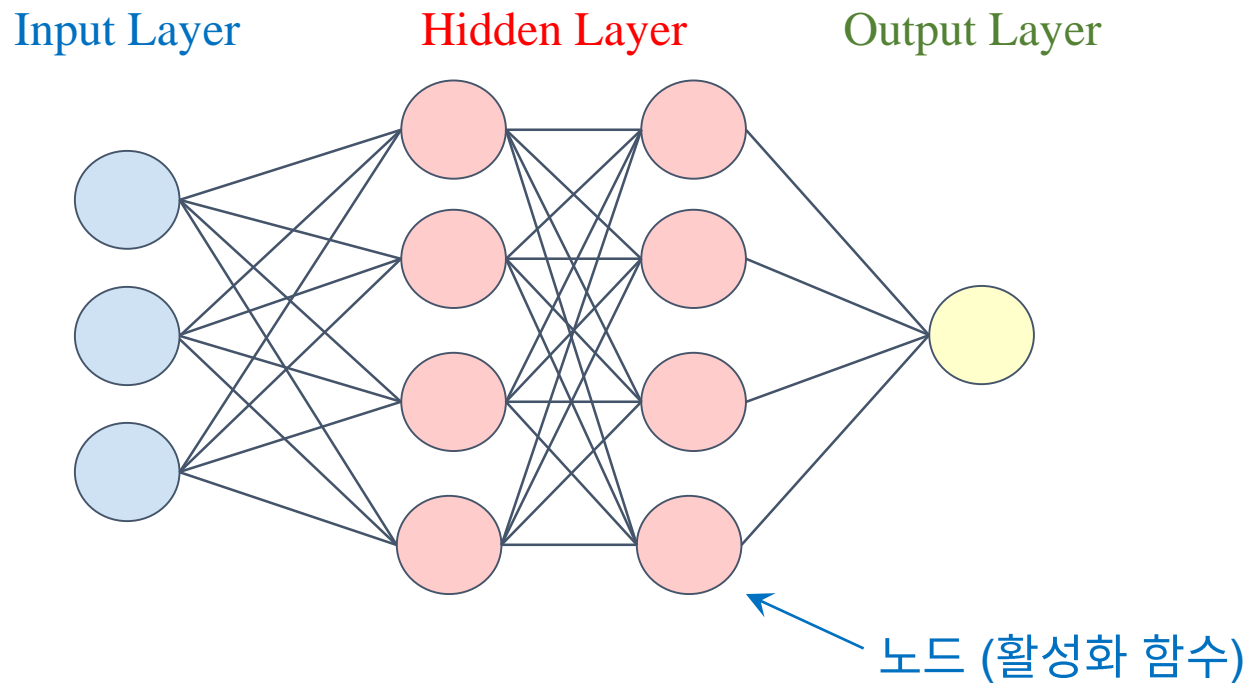


# I TensorFlow에 대해서

- TensorFlow는 머신러닝과 딥러닝 용도로 Google Brain 팀이 개발한 오픈소스 라이브러리이다.
  - ⇒ 수치 연산을 표현한 Graph 구조를 만들고 처리한다.
  - ⇒ CPU, GPU의 장점을 모두 사용할 수 있다.
  - ⇒ 다양한 운영체제 지원: Windows, macOS, Linux, iOS, Android, 등.
  - ⇒ 다양한 언어 지원: Python, Java, C++, R, 등.

# I 딥러닝에 대해서

- 딥러닝은 인공신경망에 기반한 머신러닝 방법이다.
- 인공신경망의 보편적인 구조는 다음과 같다.



# I Keras 설치

- Python에서의 Keras 패키지 설치.
  - 먼저 Python Anaconda를 설치한다.
  - 그리고 명령어 창에서 다음 순서로 설치를 마무리 한다.

```
pip install tensorflow
```

```
pip install keras
```

또는

```
python -m pip install tensorflow
```

```
python -m pip install keras
```

# I Keras 설치

- R에서의 Keras 패키지 설치.
  - 먼저 Python Anaconda를 설치한다.
  - 그리고 R에서 다음 순서로 설치를 마무리 한다.

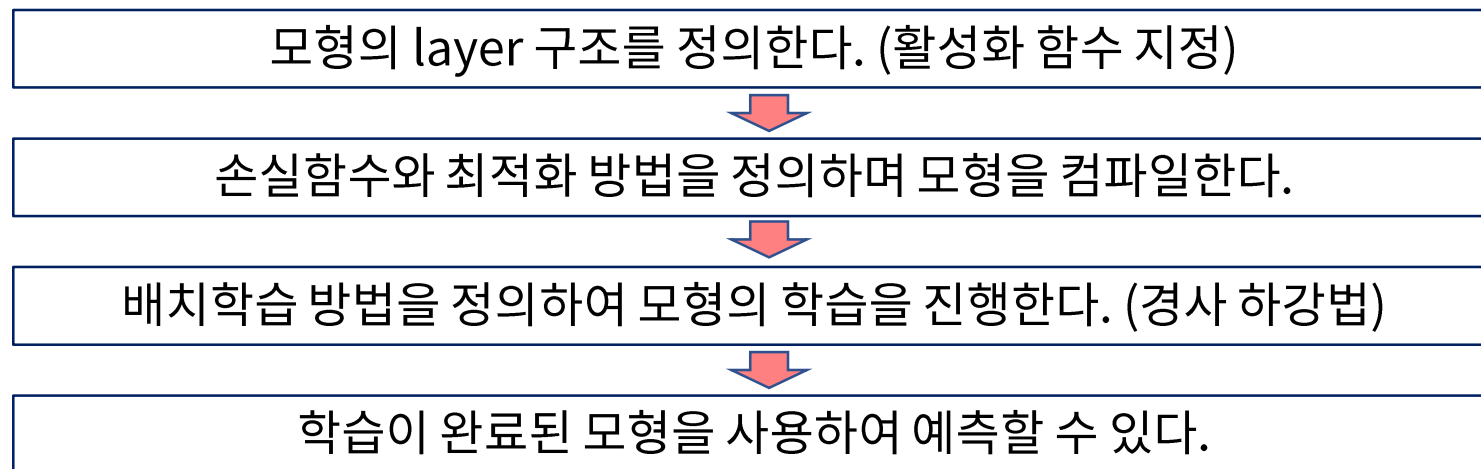
```
install.packages("keras")
```

```
library(keras)
```

```
install_keras()
```

# I Keras 학습 순서

- Keras에서는 일반적으로 다음과 같은 순서로 학습을 한다.





# I 활성화 함수

- 다음과 같은 유형의 활성화 함수 (Activation function)가 있다.

이름	수식	Keras
선형함수	$x$	linear
Sigmoid	$\sigma(x) = \frac{1}{1 + e^{-x}}$	sigmoid
ReLu	$ReLU(x) = \max(0, x)$	relu
Softmax	$\sigma(\mathbf{z})_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}}$	softmax

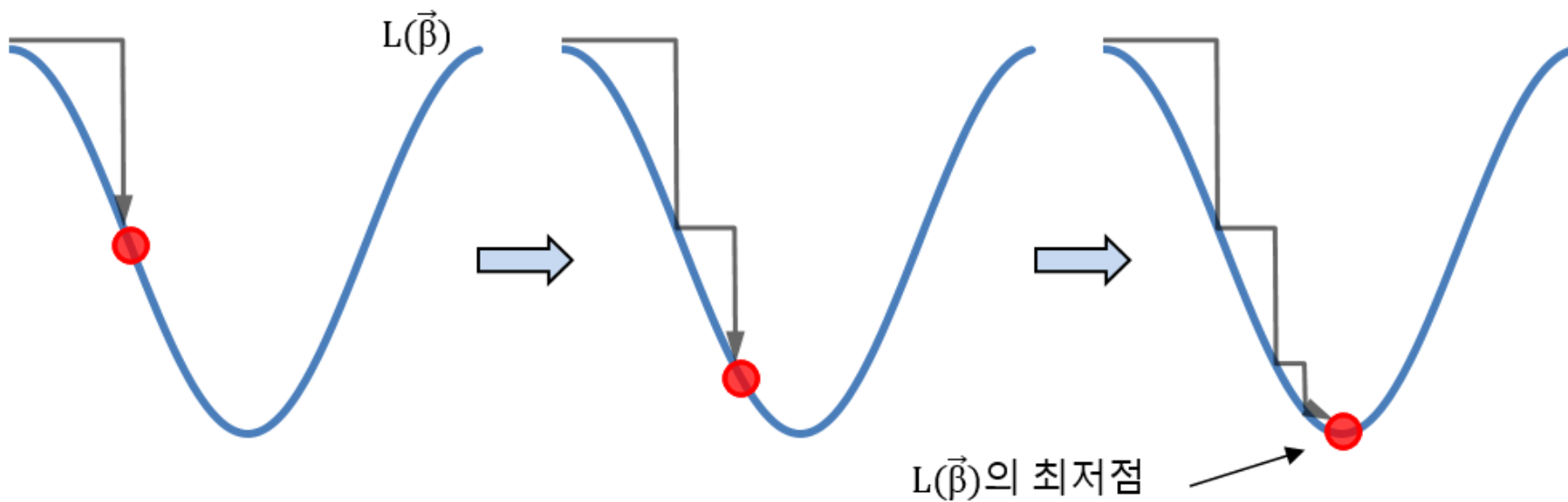
## I 비용함수/손실함수

- 다음과 같은 비용함수 (cost)/손실함수 (loss)가 있다.

이름	수식	Keras
Mean Squared Error	$L = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$	mse
Mean Absolute Error	$L = \frac{1}{n} \sum_{i=1}^n  y_i - \hat{y}_i $	mae
Binary Entropy	$L = -\frac{1}{n} \sum_{i=1}^n (y_i \ln(\hat{y}_i) + (1 - y_i) \ln(1 - \hat{y}_i))$	binary_crossentropy
Categorical Entropy	$L = -\frac{1}{n} \sum_{j=1}^k \sum_{i=1}^n (y_{j,i} \ln(\hat{y}_{j,i}) + (1 - y_{j,i}) \ln(1 - \hat{y}_{j,i}))$	categorical_crossentropy

# I 학습 알고리즘

- 학습은 일반적으로 경사 하강법에 기반한 방식으로 실행된다.



그래디언트  $-\nabla L(\vec{\beta})$ 는 벡터  $\vec{\beta}$ 를  $L(\vec{\beta})$ 의 최소화 지점으로 이끈다.

# I 학습 알고리즘

- Keras는 다음과 같은 개선된 경사 하강법을 제공한다.

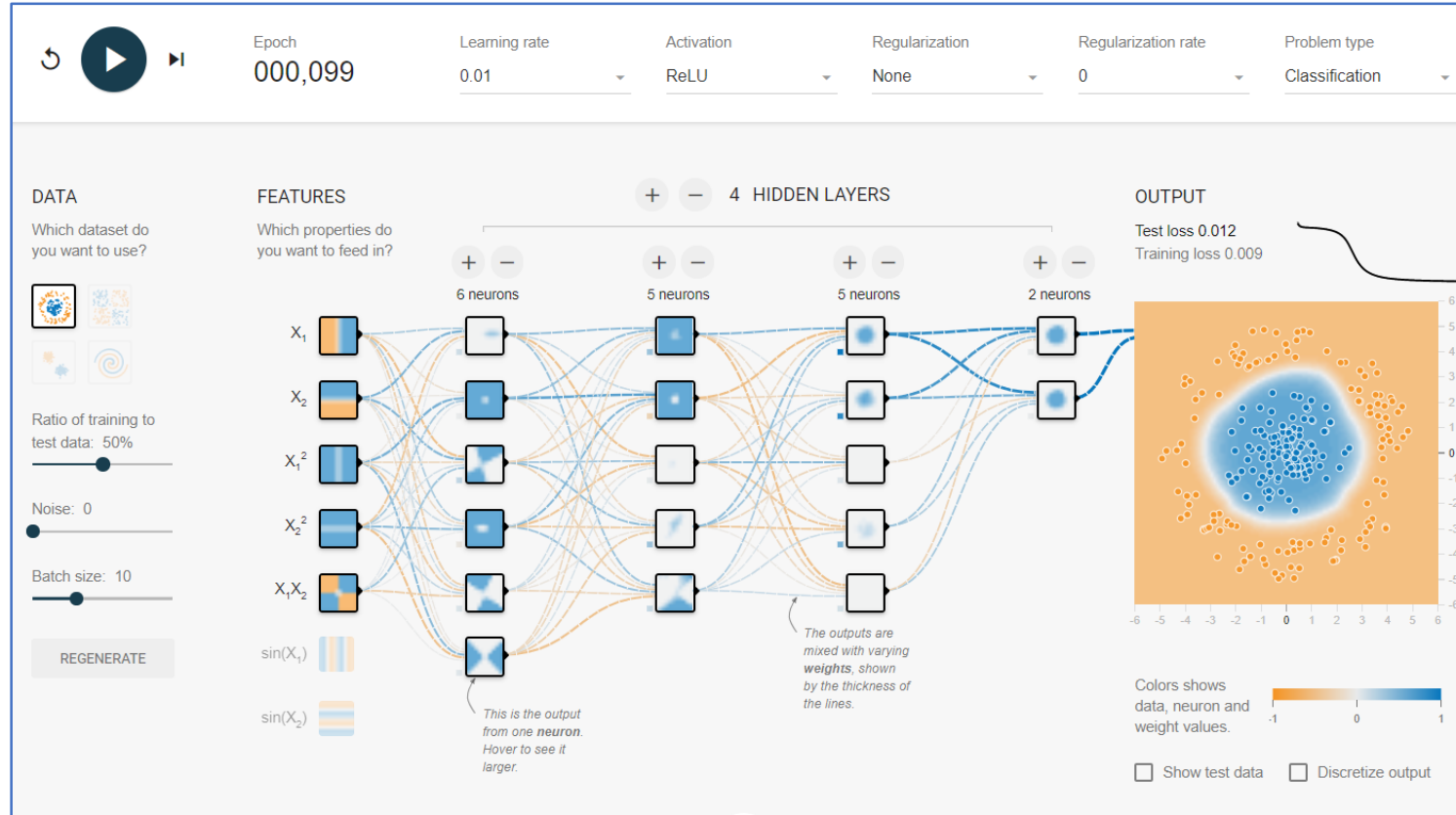
방법	설명	효과	Keras 함수
SGD 확률적 경사 하강법	배치로 경사 하강법 실행.	속도 개선.	SGD(lr=0.1)
Momentum 모멘텀	이전 스텝을 고려해서 관성을 적용함. 불필요한 오감을 줄임.	정확도 개선.	SGD(momentum=0.9)
NAG 네스테롭 모멘텀	불필요한 이동을 줄임.	정확도 개선.	SGD(lr=0.01, momentum=0.9, nesterov=True)
Adagrad	이동 보폭을 조절해 줌.	보폭 크기 개선.	Adagrad(lr=0.01)
RMSProp	Adagrad 보폭 민감도 개선.	보폭 크기 개선.	RMSProp(lr=0.01)
Adam	Momentum과 RMSProp을 합친 방법.	정확도와 보폭 크기 개선.	Adam(lr=0.01)

## I 배치 학습

- 데이터 전체를 한번에 사용하기 보다는 작은 크기의 배치 (batch)로 나누어서 학습시킨다.
- 비교적 작은 배치를 사용하면 학습이 효율적이다.
- 배치 학습은 일반화 (시험) 결과에도 유리하게 작용한다.
- 배치는 랜덤으로 뽑아오고 모든 데이터를 다 사용하지 않아도 좋다.

# I 딥러닝 추천 사이트

- 다음 주소의 웹사이트로 간다: [playground.tensorflow.org](https://playground.tensorflow.org)





# I 딥러닝 추천 사이트

- 다음 주소의 웹사이트로 간다: *playground.tensorflow.org*
  1. 왼쪽 사이드 패널에서 데이터를 선택한다.
  2. 속성 (feature)을 선택할 수 있다
  3. 노드 (node)의 개수, 은닉층의 개수 등을 조절할 수 있다
  4. 드롭다운 메뉴에서 활성화함수, 학습속도 등을 설정할 수 있다.
  5. 연결선의 굵기가 가중치의 크기를 나타낸다. 클릭하여 수동조절 가능하다.
  6. [Play] 버튼을 클릭하여 인공신경망의 학습 개시. Loss 가 감소하며 학습이 진행된다.

| 끝.

감사합니다.

