

Machine learning & TensorFlow



9기 김남호

INDEX

- Machine Learning
- TensorFlow
- 성적 예측 프로그램

Machine Learning

기계학습

01

>> 관심

02

03

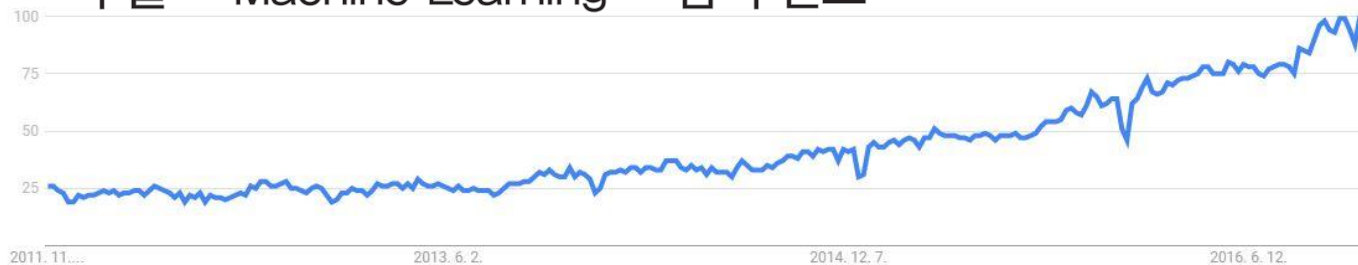
04



2017년 Gartner hypercycle 보고서

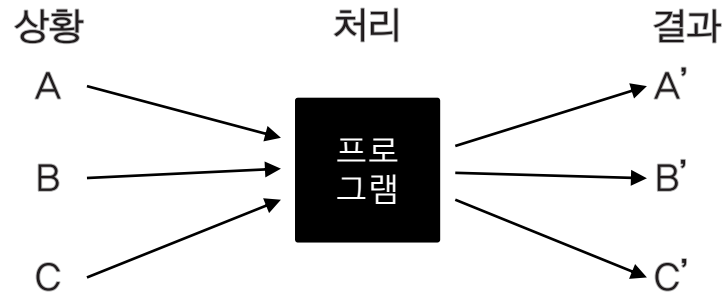
시간 흐름에 따른 관심도 변화 ?

구글 'Machine Learning' 검색 빈도



>> What is Machine Learning?

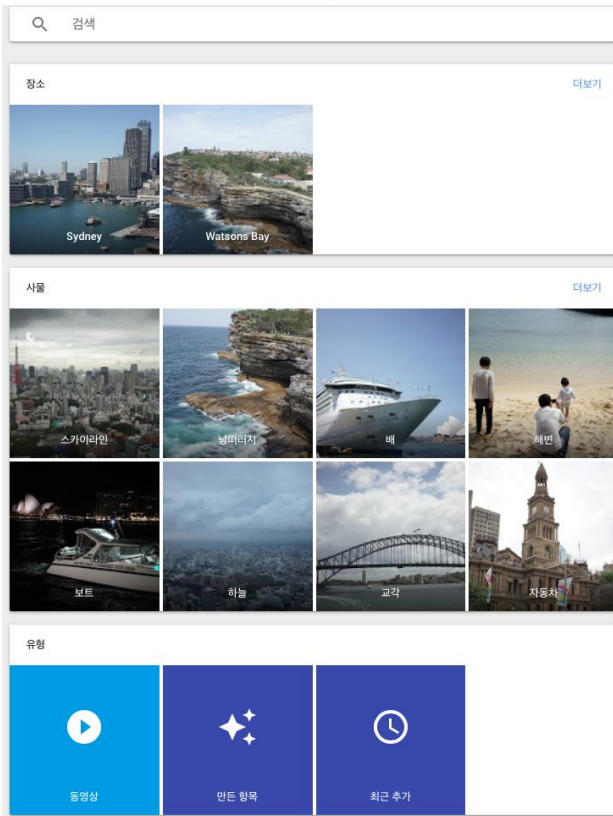
기존 Explicit Program



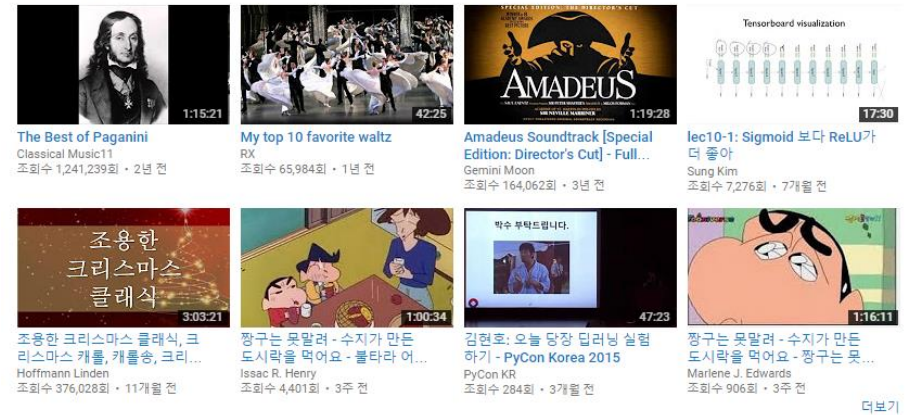
Machine Learning



Machine Learning in use



맞춤 동영상

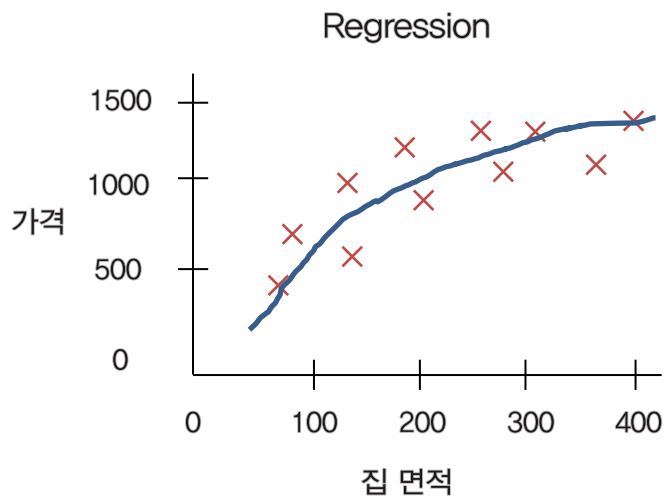


01
02
03
04

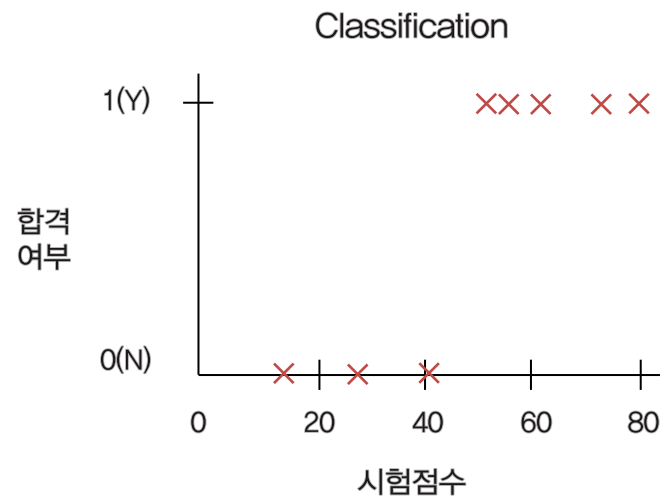
» Supervised Learning
Unsupervised learning

Supervised Learning

학습할 데이터의 결과가 무엇인지 정해져 있다.
mapping



결과가 연속적
집 평수 - 가격

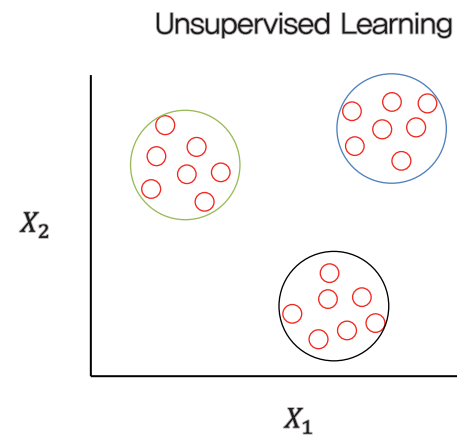
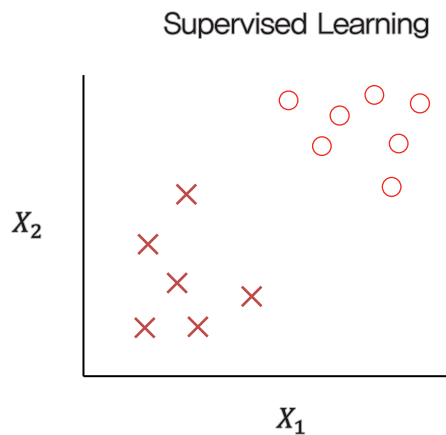


결과가 비연속적
시험점수 - 합/불
종양크기 - 음성/양성

01
02
03
04
»» Supervised Learning
Unsupervised learning

Unsupervised Learning

학습할 데이터의 결과가 무엇인지 정해져 있지 않다.
no label



군집화
SNS의 가까운 사람

TensorFlow

텐서플로

01

>>> What is TensorFlow?

02

03

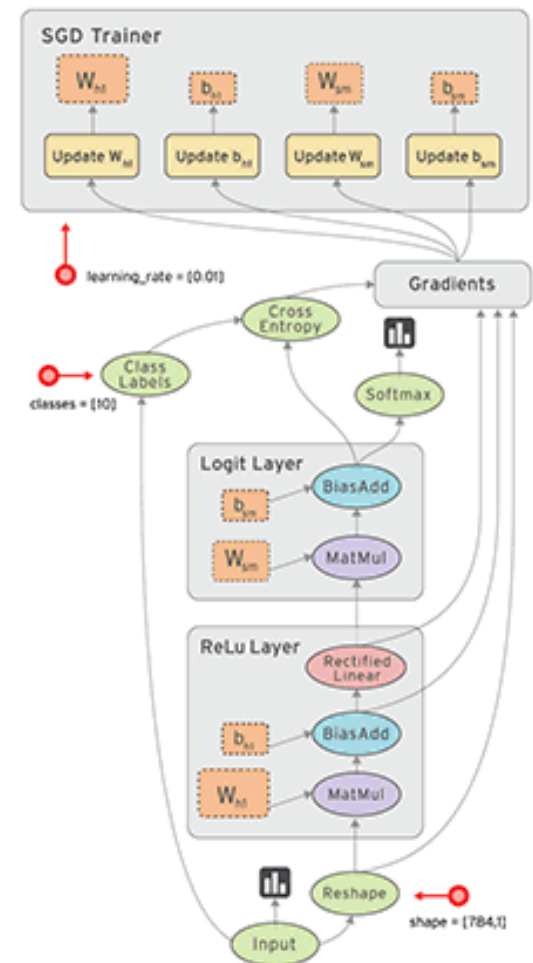
Open Source Library



Google Brain Team

Make machines intelligent. Improve people's lives.

Data Flow Graph



01

>> What is TensorFlow?

02

03

Creator



Language

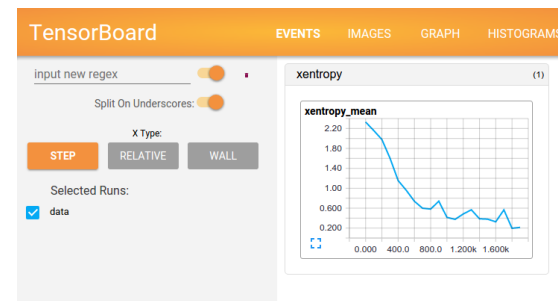


OS

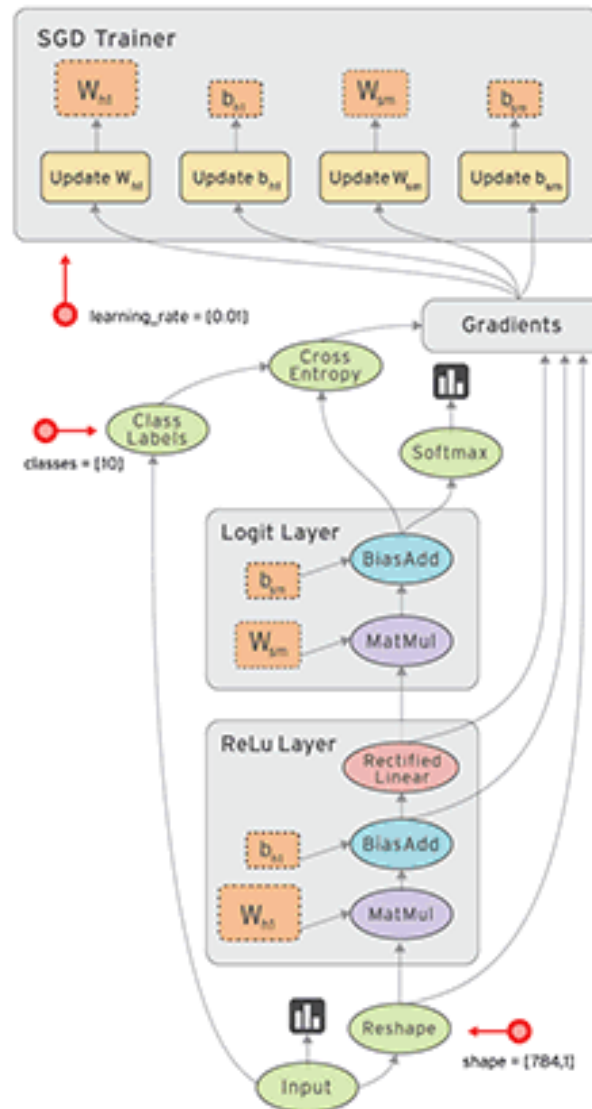
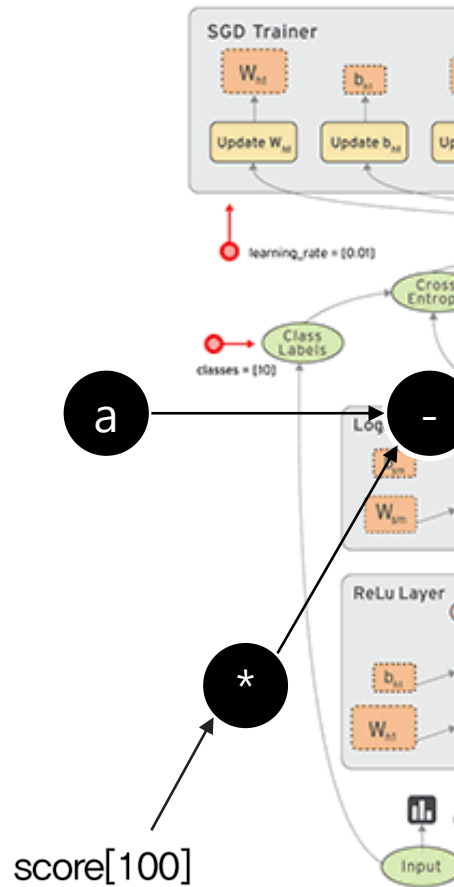


TensorBoard

Visualization



Data Flow Graph



operation

전달되는 tensor

Tensor

Array RGB

Page 3	0.689	0.706	0.118	0.884	...
blue	0.535	0.532	0.653	0.925	...
intensity	0.314	0.265	0.159	0.101	...
values	0.553	0.633	0.528	0.493	...
	0.441	0.465	0.512	0.512	...
	0.288	0.401	0.421	0.398	...
	1.647	0.515	0.816	0.912	0.713
	1.300	0.205	0.526	0.219	0.328
	1.428	0.712	0.929	0.128	0.133
	1.604	0.918	0.344
	1.121	0.113	0.126
	0.204	0.175
	0.760	0.531
	0.997	0.910
	0.995	0.726

01

02

03

Why TensorFlow?



성적 예측 프로그램

Linear Regression

01

>> Training Data

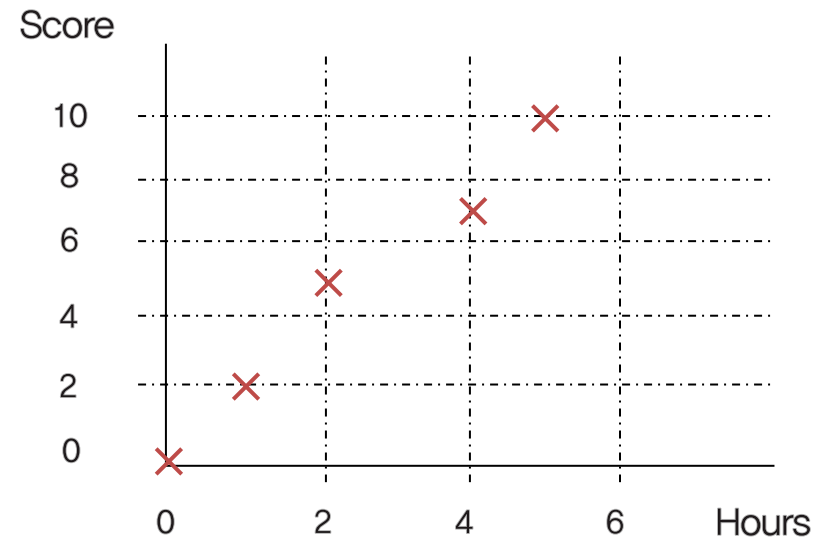
02

03

04

학습시간-성적 관계

X(Hours)	Y(Score)
0	0
1	2
2	5
4	7
5	10



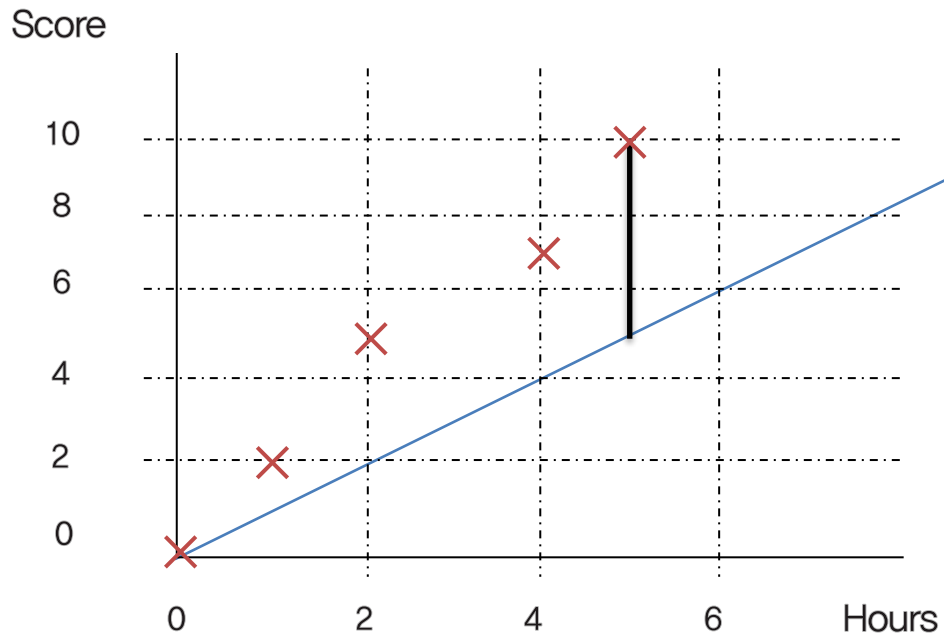
01

>> Hypothesis & Cost

02

03

04



Hypothesis(가설)

$$H(x) = Wx + b$$

Cost

$$Cost = (H(x) - y)^2$$

$$Cost(W, b) = \frac{1}{m} \sum_{i=1}^m (H(x^{(i)}) - y^{(i)})^2$$

소스코드

```

1 import tensorflow as tf
2
3 # train data
4 x_data = [0, 1, 2, 4, 5] # 공부시간
5 y_data = [0, 2, 5, 7, 10] # 점수
6
7 # 임의의 W, b 를 생성한다.
8 W = tf.Variable(tf.random_uniform([1], -1.0, 1.0))
9 b = tf.Variable(tf.random_uniform([1], -1.0, 1.0))
10
11 # placeholder
12 X = tf.placeholder(tf.float32)
13 Y = tf.placeholder(tf.float32)
14
15 # hypothesis
16 hypothesis = W * X + b
17
18 # cost function
19 cost = tf.reduce_mean(tf.square(hypothesis - Y))
20
21 # minimize
22 a = tf.Variable(0.1) # Learning rate, alpha
23 optimizer = tf.train.GradientDescentOptimizer(a)
24 train = optimizer.minimize(cost)
25
26 # initialize the variables
27 init = tf.initialize_all_variables()
28 sess = tf.Session()
29 sess.run(init)
30
31 # 2001 번의 학습
32 for step in xrange(2001):
33     sess.run(train, feed_dict={X: x_data, Y: y_data})
34
35 # 공부시간이 3시간?
36 print "3 hour Score : ", sess.run(hypothesis, feed_dict={X: 3})
37 # 공부시간이 6시간?
38 print "6 hour Score : ", sess.run(hypothesis, feed_dict={X: 6})

```

$$H(x) = Wx + b$$

$$Cost(W, b) = \frac{1}{m} \sum_{i=1}^m (H(x^{(i)}) - y^{(i)})^2$$

Cost를 최소화 하는 W,b를 찾기

실제 데이터 처리

01



결과

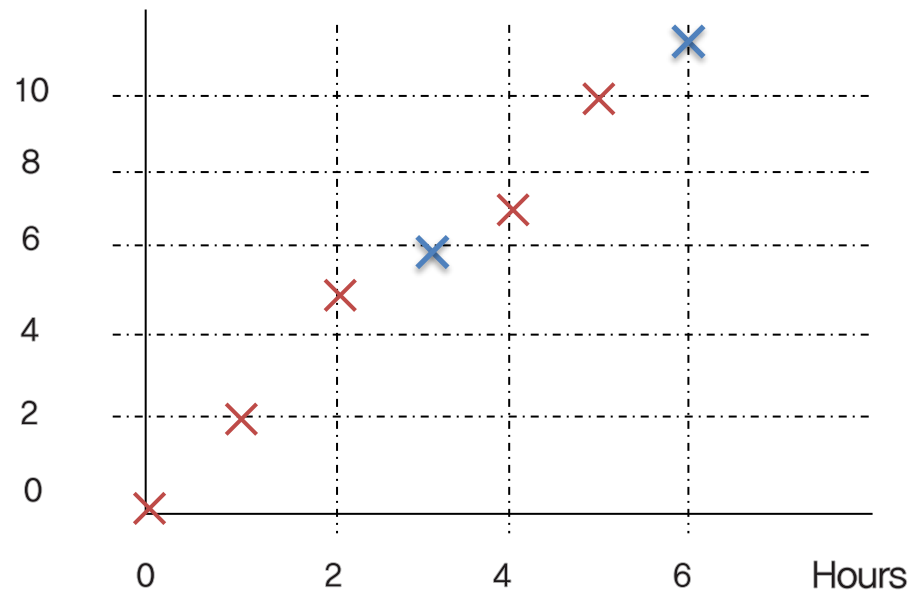
02

03

04

3 hour Score : [5.93024111]
6 hour Score : [11.58141232]

Score



Thank you

9기 김남호