

# mi-8. ニューラルネットワーク の基礎

(人工知能シリーズ)

<https://www.kkaneko.jp/cc/mi/index.html>

金子邦彦



## アウトライン

- 8-1 はじめに
- 8-2 ニューラルネットワークでの学習
- 8-3 ニューラルネットワークの現状
- 8-4 ニューラルネットワークの構造
- 8-5 ユニットの活性化と伝搬
- 8-6 ニューラルネットワークによる判定, 予測
- 8-7 ニューラルネットワークを作成するプログラム
- 8-8 機械学習

# 8-1 はじめに

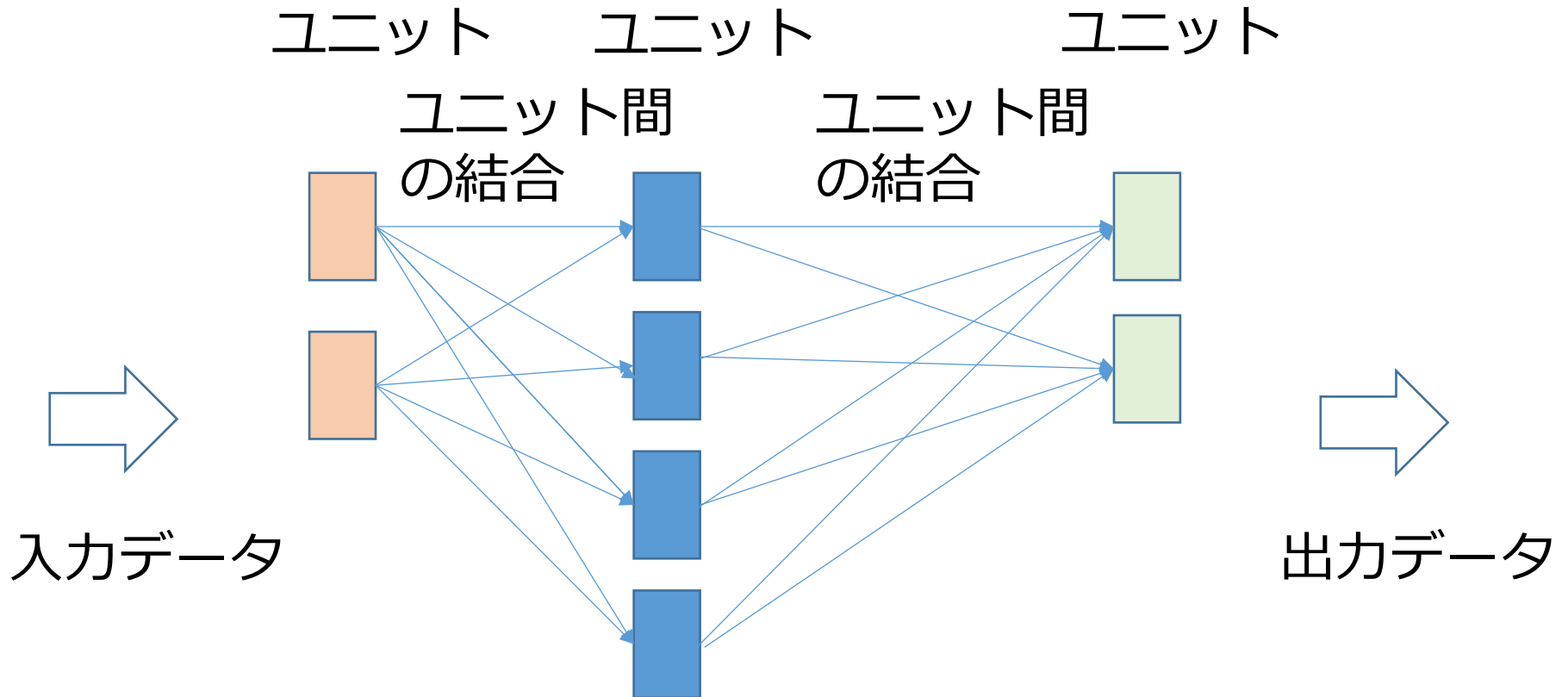
# 今日の内容



- ニューラルネットワークの学習能力について知る
- 機械学習について知る

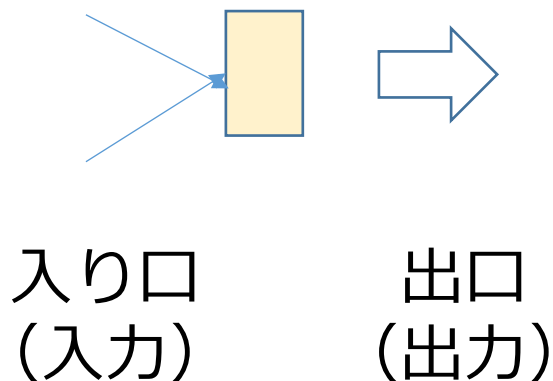
# 8-2 ニューラルネットワーク での学習

# ニューラルネットワークの仕組み



データは入力から出力の方向へ

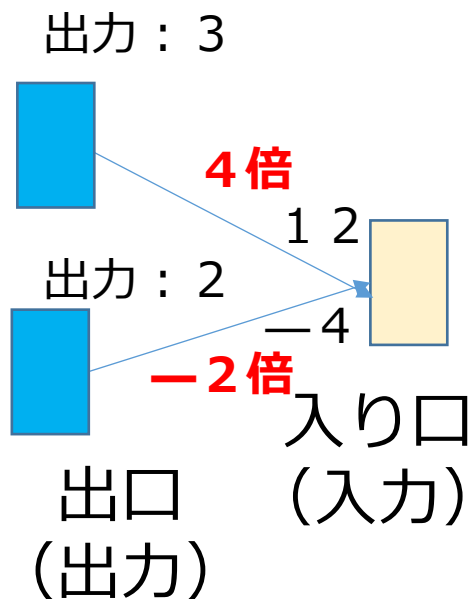
## ユニット



**ユニット**は、

- 入力の合計が大きいと  
**活性化**する  
(高い値を出力する)
- 入力の合計が低いと  
**非活性化**する  
(低い値を出力する)

# 確認問題



次のような**ユニット**がある

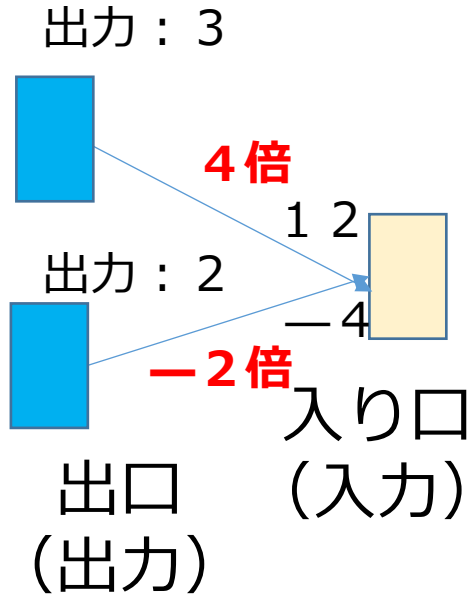
- 入力の合計が **10以上** のとき、**活性化** し、 **1** を出力する
- 入力の合計が **10未満** のとき、**非活性化** し、 **0** を出力する

入力が 3 と 5 のとき出力は?  
0

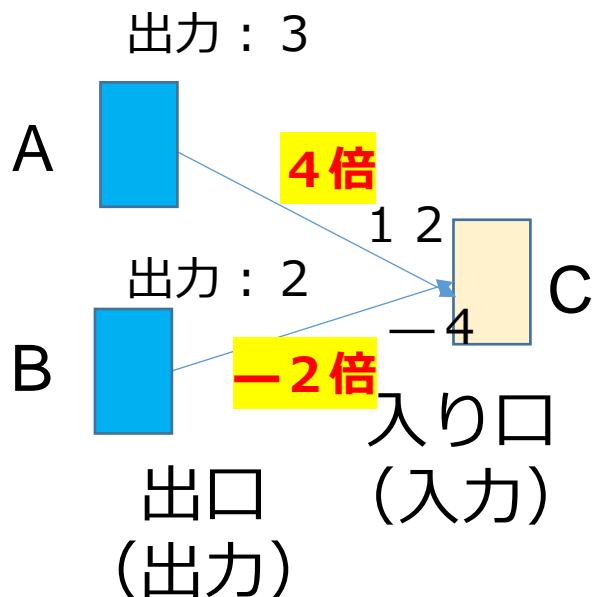
入力が 20 と 30 のとき出力は?  
1



# ユニット間の結合



**ユニット間の結合**では、  
**出力**の値が  
**〇〇倍**  
されて、次の**ユニット**の  
**入力**になる

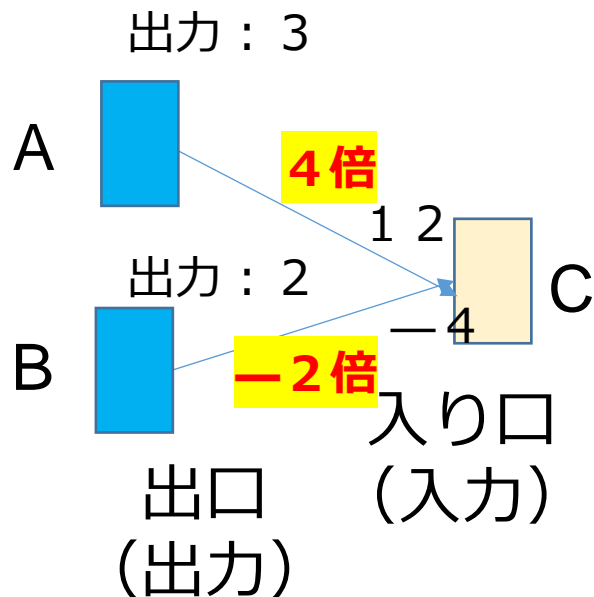


ユニット間の結合の  
「〇〇倍」

は、学習の途中で**変化**する

ニューラルネットワークの  
学習は、  
望み通りの出力が得られる  
ように、「〇〇倍」のところ  
（結合の重み）を**自動調整**  
すること

# 確認問題



次のような**ユニット**がある

- ・ 入力の合計が **10以上** のとき、**活性化** し、 **1** を出力する
- ・ 入力の合計が **10未満** のとき、**非活性化** し、 **0** を出力する

ユニットAの出力は3であるとする。

ユニットBの出力が**2以下**のとき

ユニットCは**活性化**

ユニットBの出力が少しでも**2を超えたとき**

ユニットCは**非活性化**

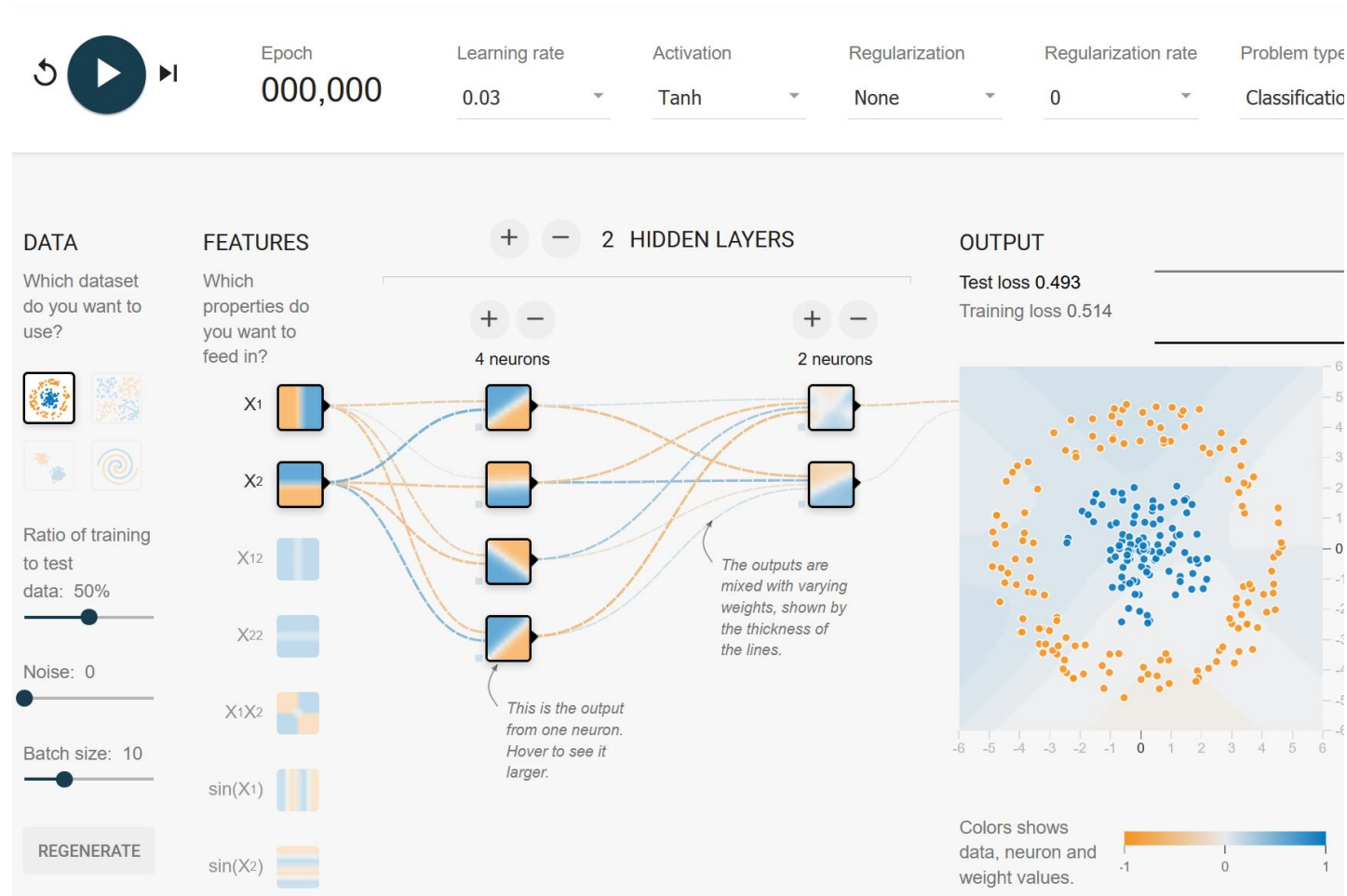
にしたい。結合の重みをどう調整するか？

(答え) 「-2倍」を「-1倍」へ

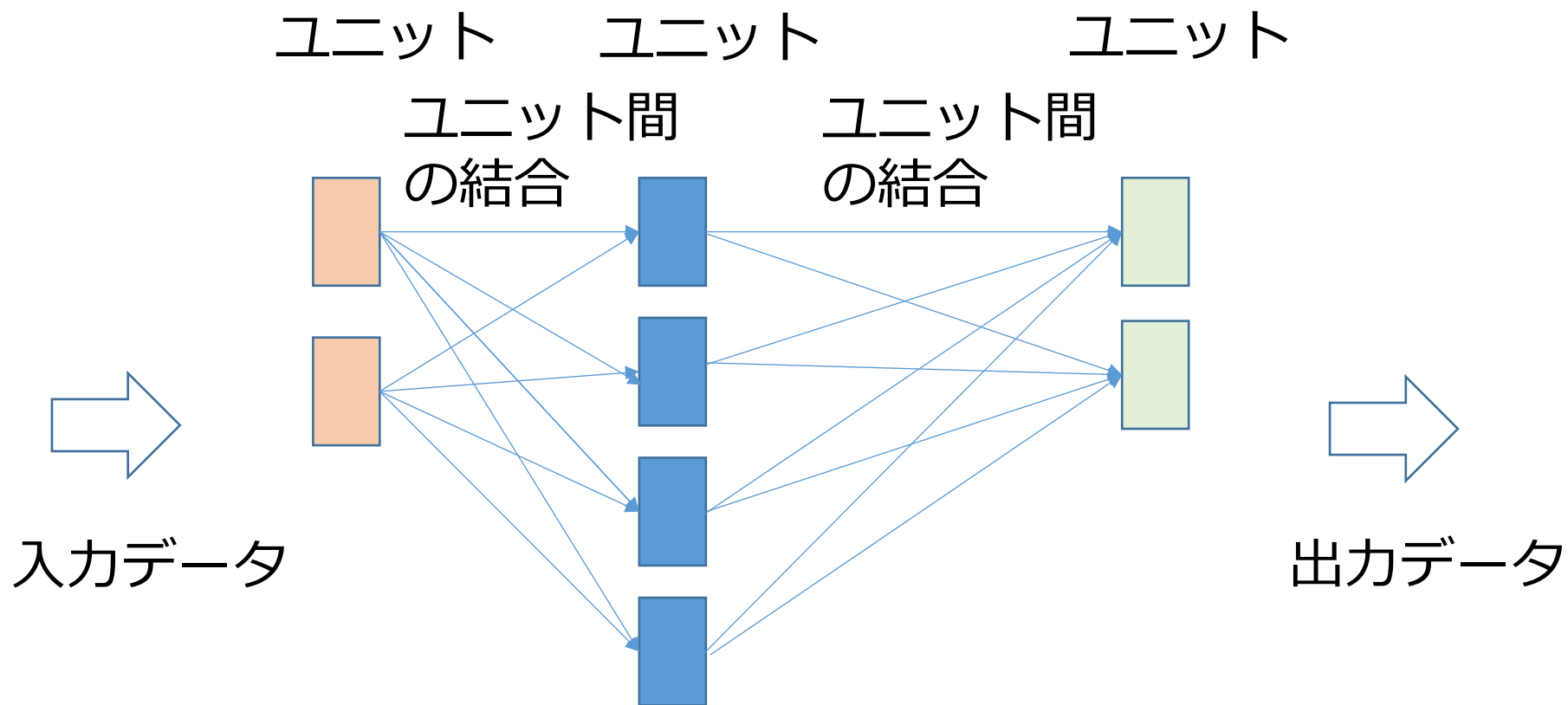
# 学習能力をコンピュータに組み込んでおき、 あとでデータを与えて学習させる



<http://playground.tensorflow.org>



# まとめ



データは入力から出力の方向へ

ニューラルネットワークの学習では、**ユニット間の結合の重み**が**変化**

# 8-3 ニューラルネットワーク の現状

# ニューラルネットワークの進展

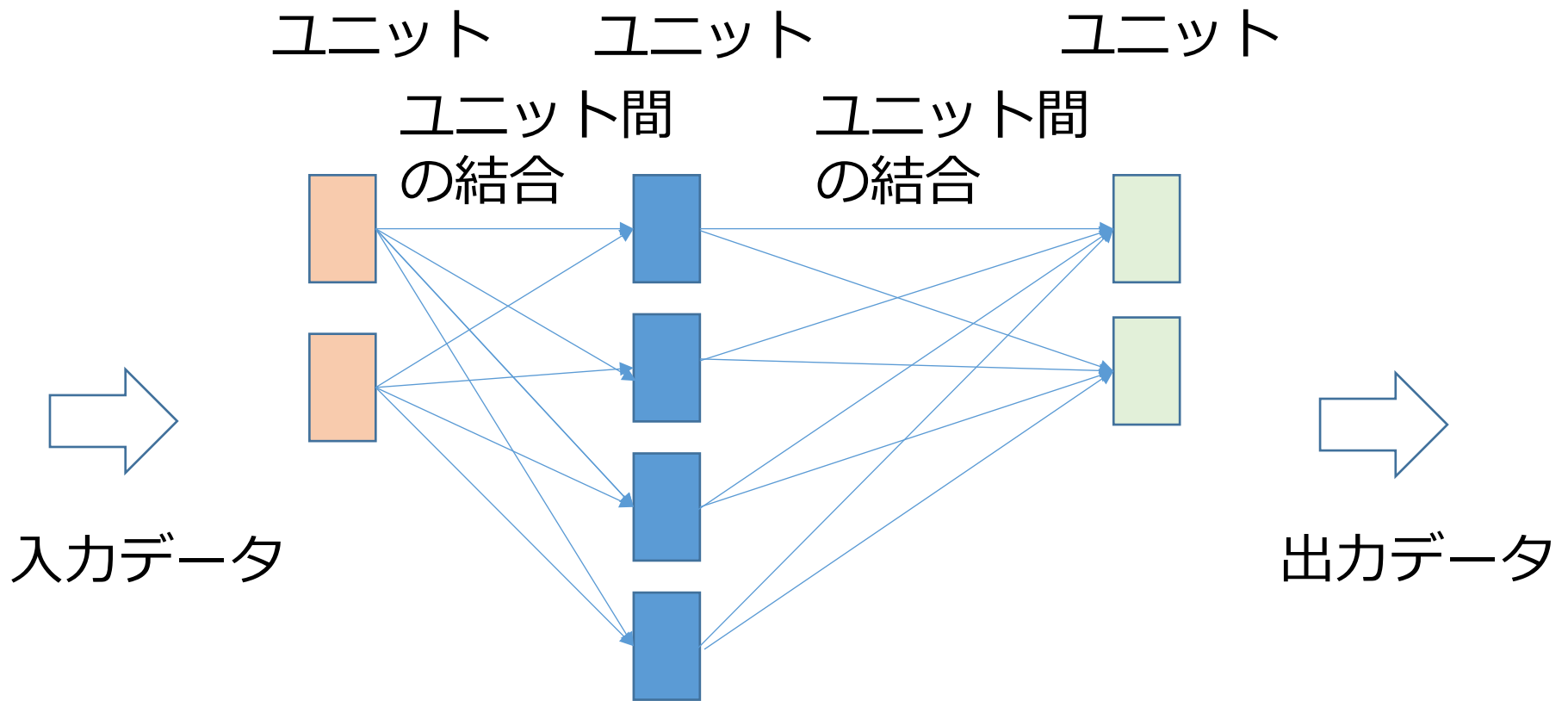


- ニューラルネットワークの技術革新, 学習能力の向上  
基盤技術: Heの初期化, Batch Normalization,  
Dropout
- 多層の構成, 多種のユニット等により広範囲の用途へ  
CNN, LTSM, GAN など
- ニューラルネットワークの能力の実証  
VGG16 など多数
- ニューラルネットワークを高速にシミュレーションできる  
高性能のコンピュータ  
高性能プロセッサ、GPU
- ニューラルネットワークの学習に役立つ大量のデータ  
データ計測、データ収集

# 8-4 ニューラルネットワーク の構造

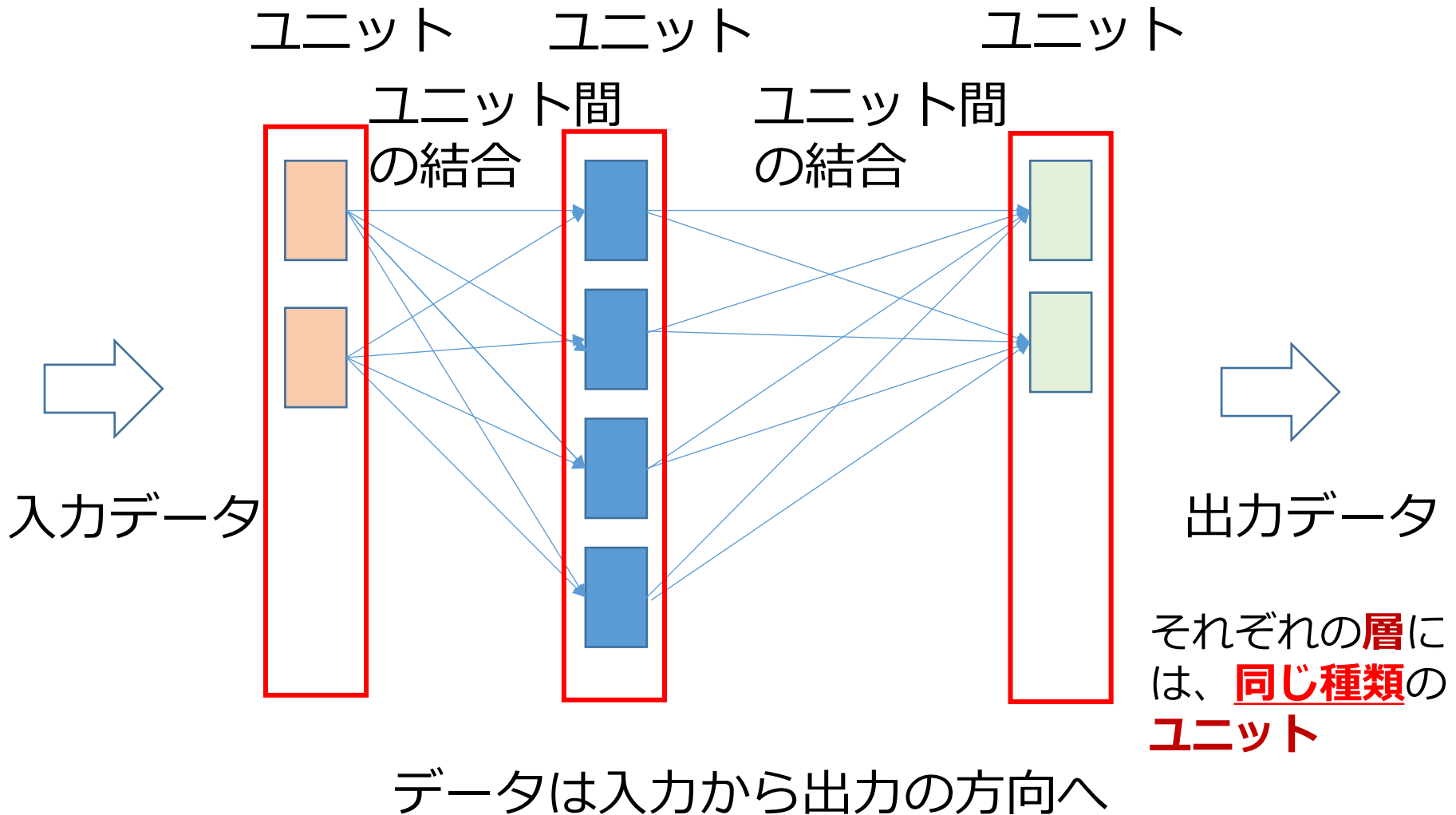


# ニューラルネットワークの構造

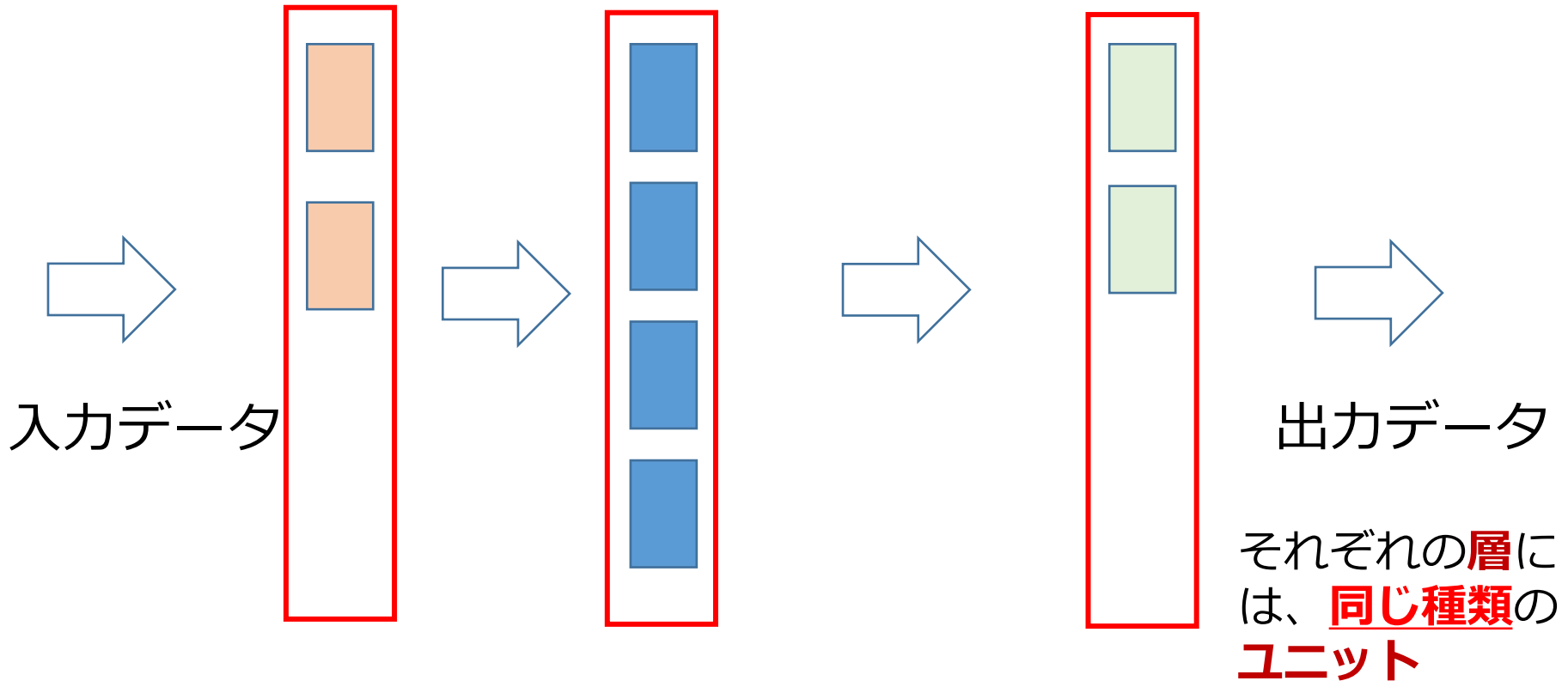


データは入力から出力の方向へ

# ニューラルネットワークの構造



# ニューラルネットワークの構造

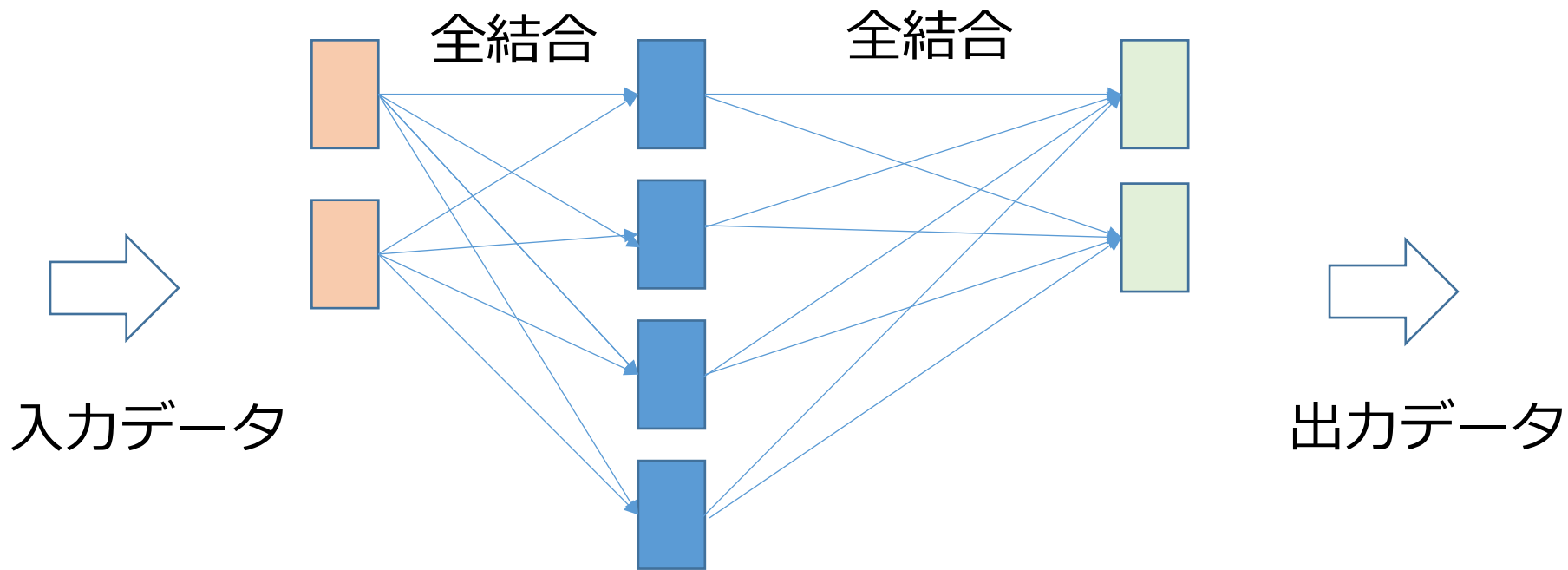


データは入力から出力の方向へ

# ユニット間の結合



前の層の**ユニット**と、次の層の**ユニット**の間をつなぐ

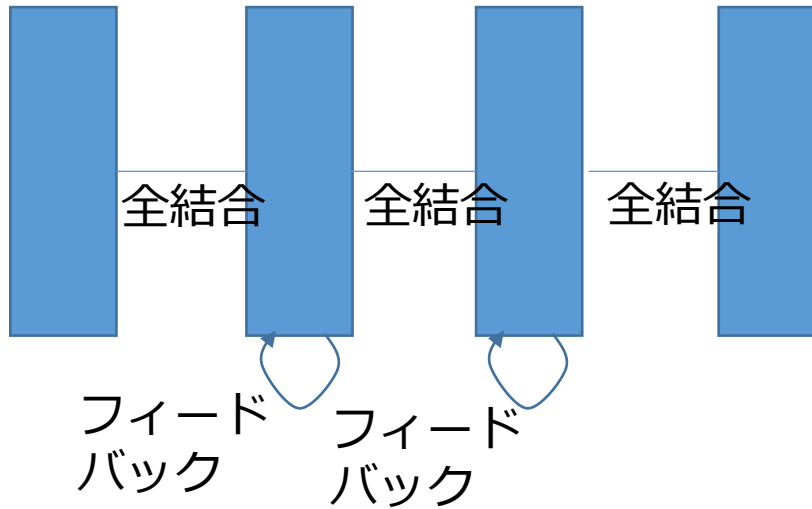


データは入力から出力の方向へ

# ニューラルネットワークのバリエーション

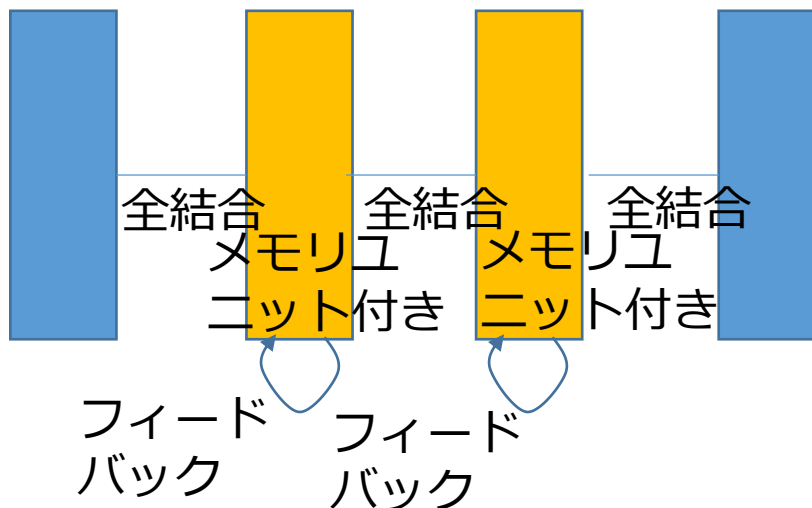


## • Recurrent Network



フィード  
バック付き

## • LSTM Recurrent Neural Network

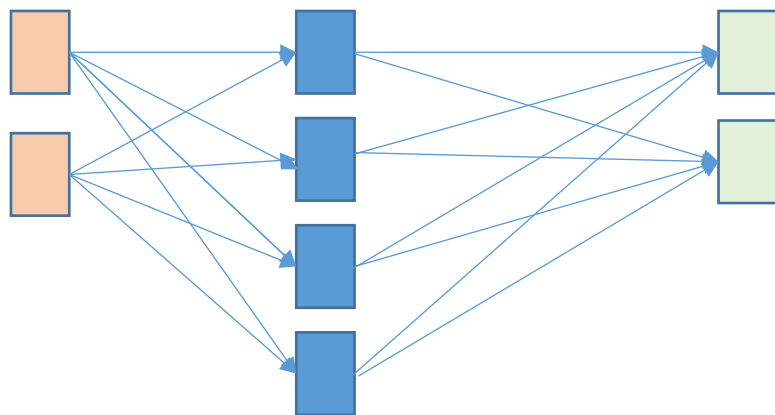


メモリユニット  
付き

# まとめ

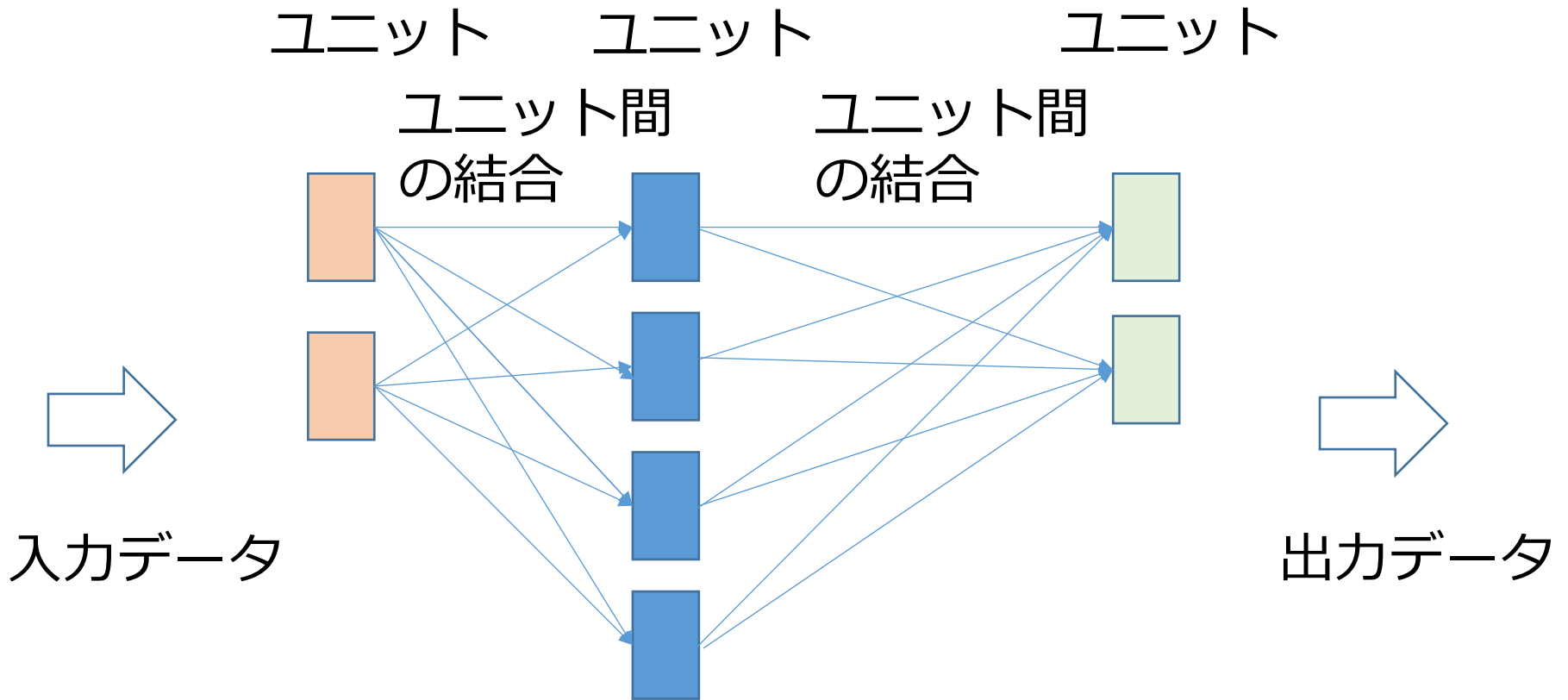


- **ニューラルネットワーク**は、層が積み重なっている
- 層の中には、**ユニット**が並ぶ。
- **ユニット**は、互いにつながる。
- より複雑な構造のバリエーションもある。



# 8-5 ユニットの活性化と伝搬

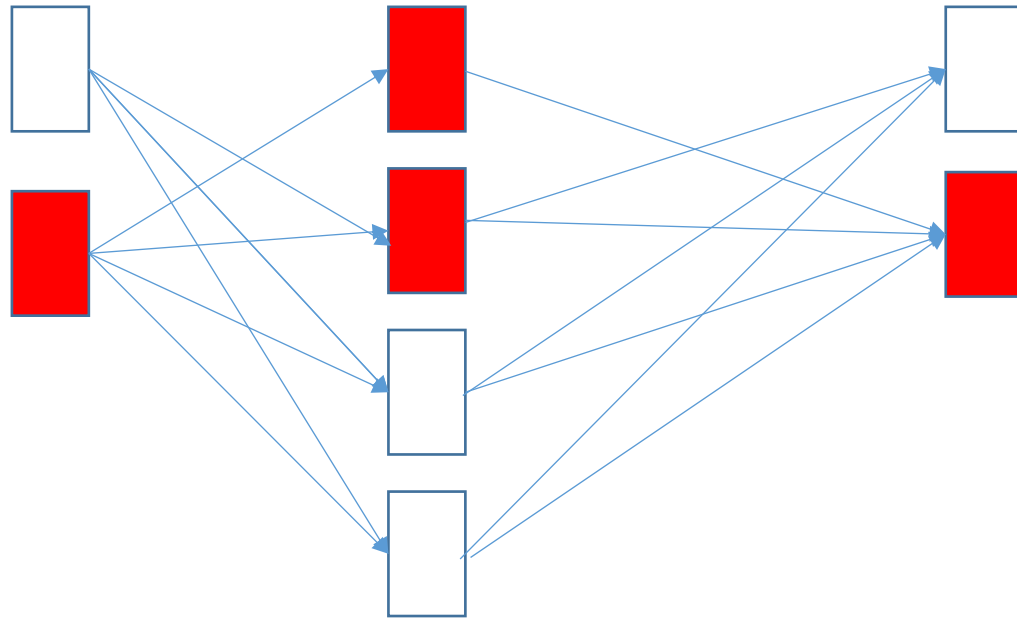
# ニューラルネットワークの構造



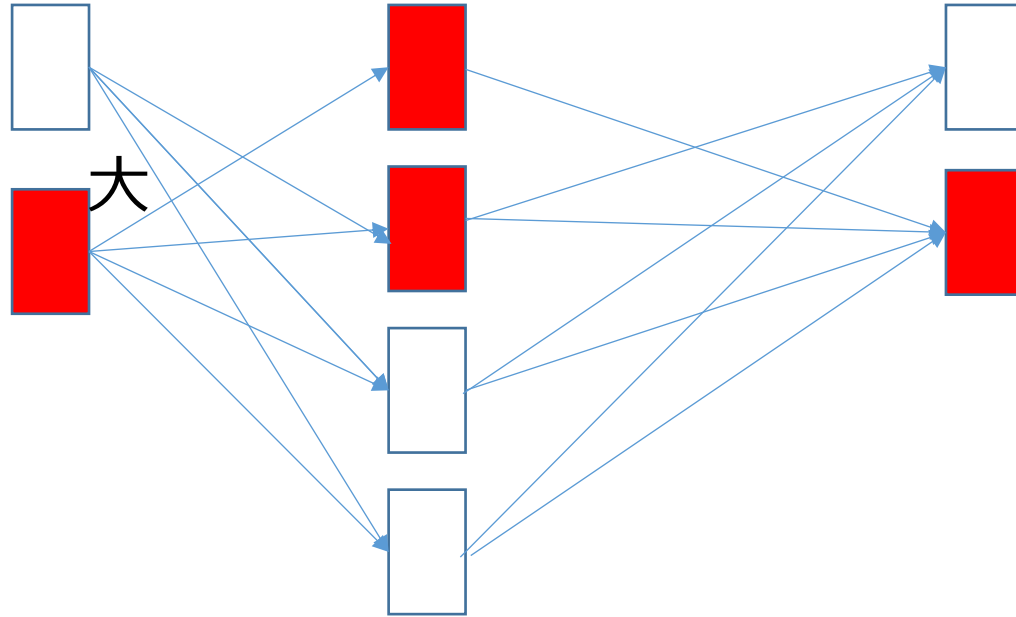
データは入力から出力の方向へ



# 活性化と非活性化

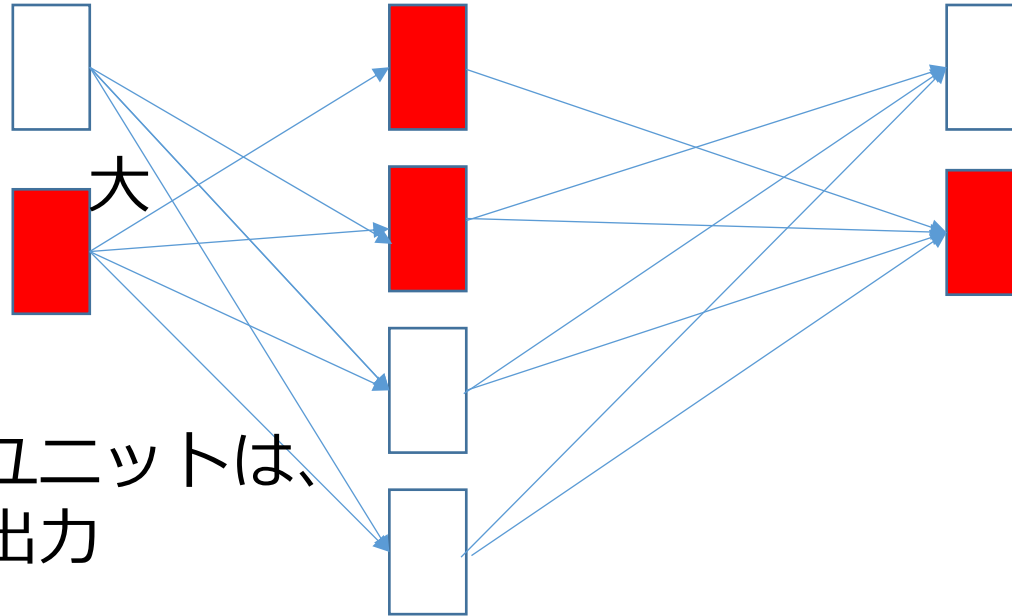


**ユニット**は**活性化**したり、**非活性化**したりする  
(入力に応じて**ダイナミック**に変化)



活性化したユニットは、  
大きな値を出力

正の値で大  
負の値で大

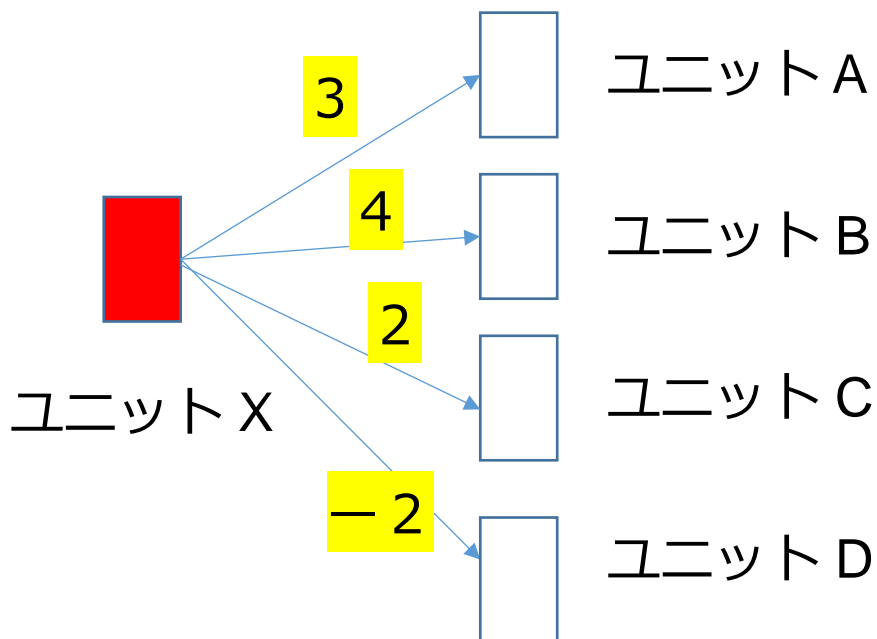


活性化したユニットは、  
大きな値を出力

次の層のユニットに、大きな値が  
伝搬される

- ・ 重みが正なら、正の大きな値
- ・ 重みが負なら、負の多いな値

# 確認問題①



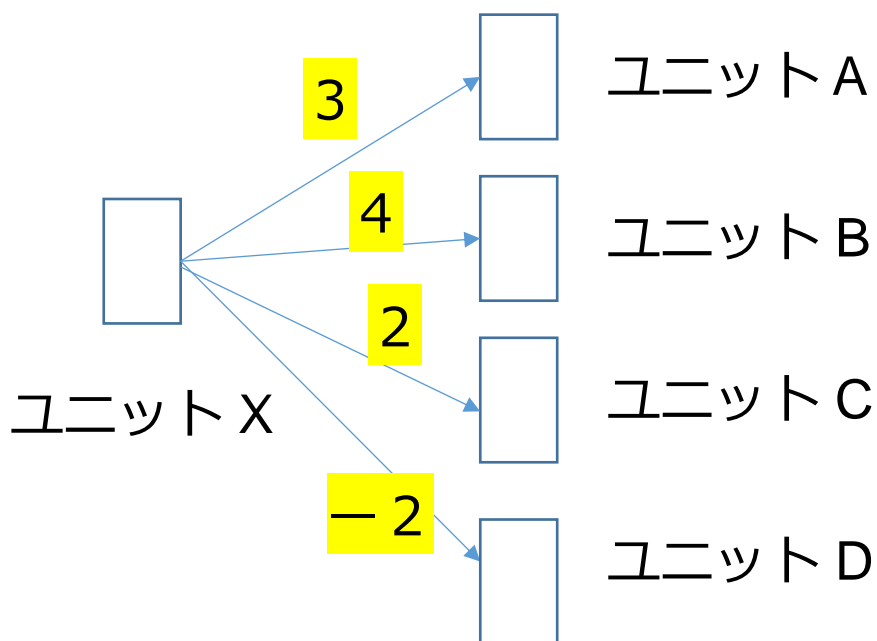
**ユニット**が、左の図のように  
**結合**している。  
**結合の重み**は 3, 4, 2, -2 である。

**ユニット** X は**活性化**し、  
1  
を出力したとき、それにより  
**ユニット** A, B, C, D が受け取る値は  
何か？

(答え)

- A 3
- B 4
- C 2
- D -2

## 確認問題②



**ユニット**が、左の図のように結合している。  
**結合の重み**は 3, 4, 2, -2 である。

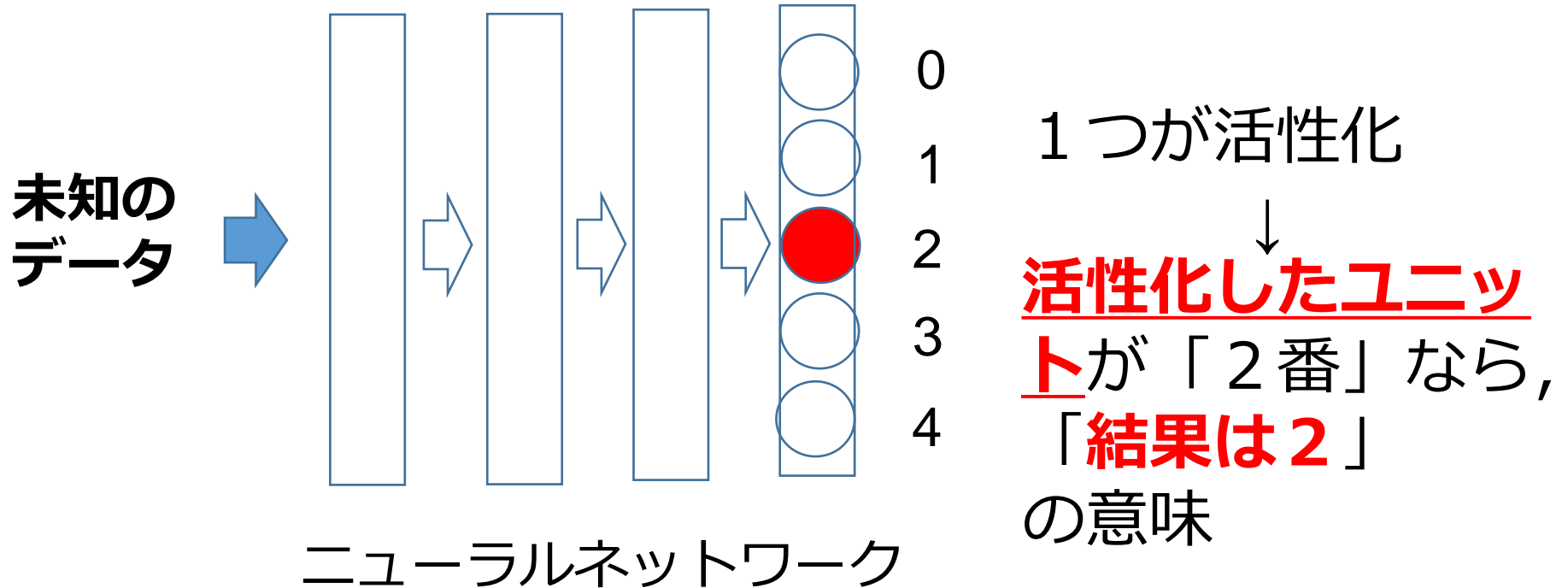
**ユニット** X が **非活性化**し、  
0  
を出力したとき、それにより  
**ユニット** A, B, C, D が受け取る値は何か？

(答え)

- A 0
- B 0
- C 0
- D 0

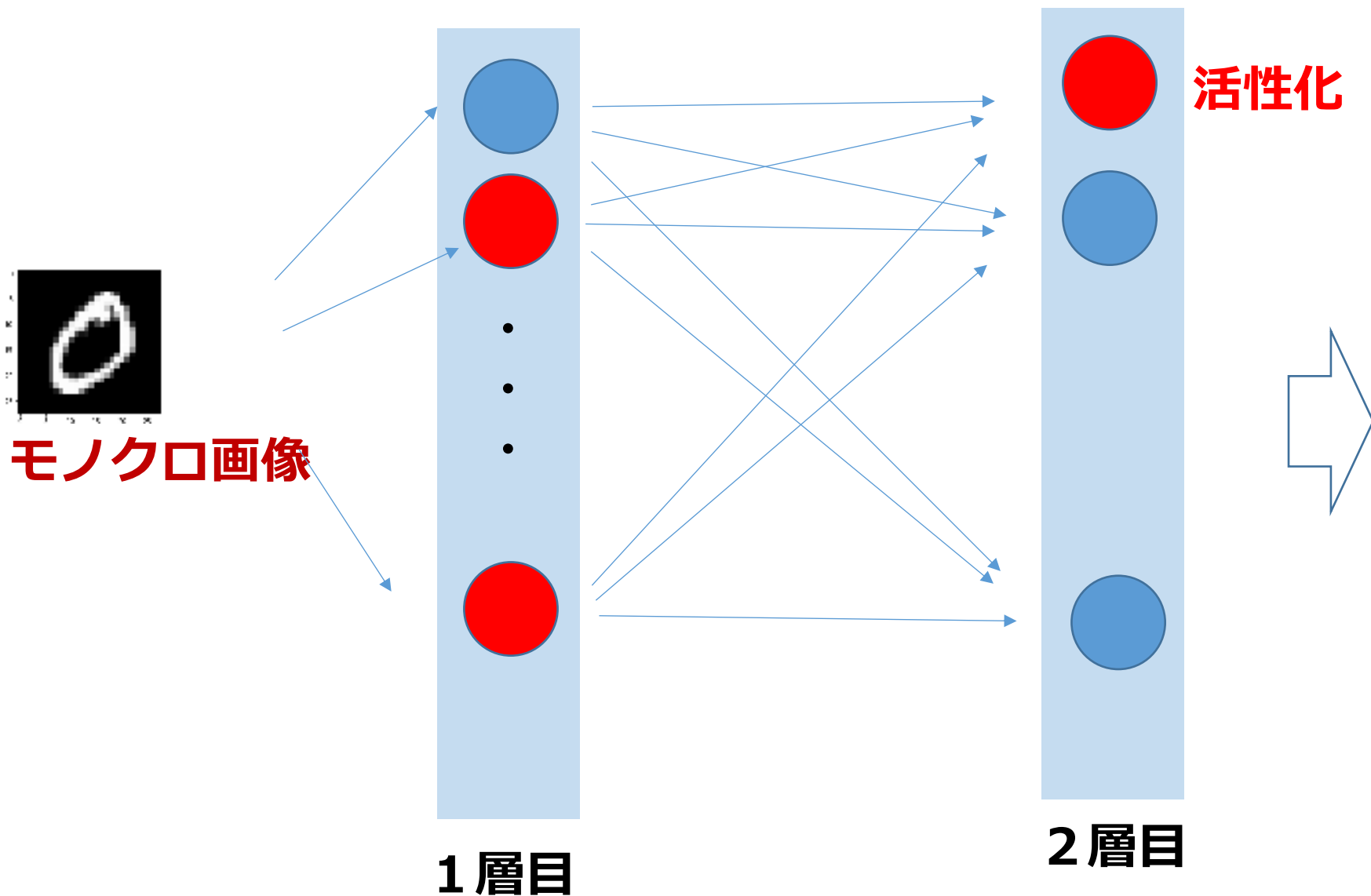
# 8-6 ニューラルネットワーク による判定, 予測

# ニューラルネットワークによる判定、予測



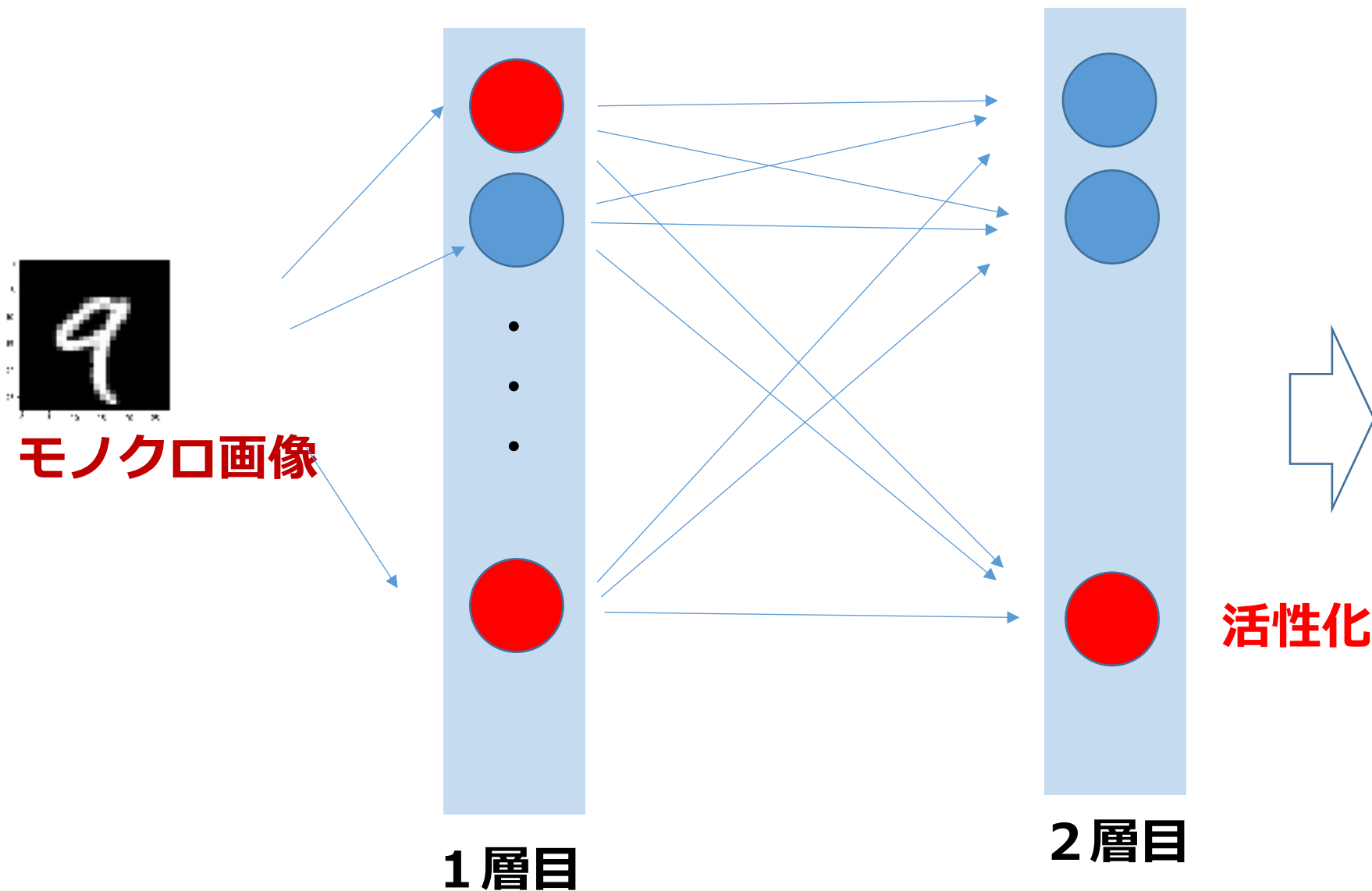
- ・ **ニューラルネットワーク**での判定、予測は、**未知のデータ**を与えて、最終層のユニットを活性化させることで行う

# ニューラルネットワークによる予測





# ニューラルネットワークによる予測

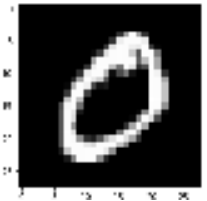


# ニューラルネットワークの仕組み

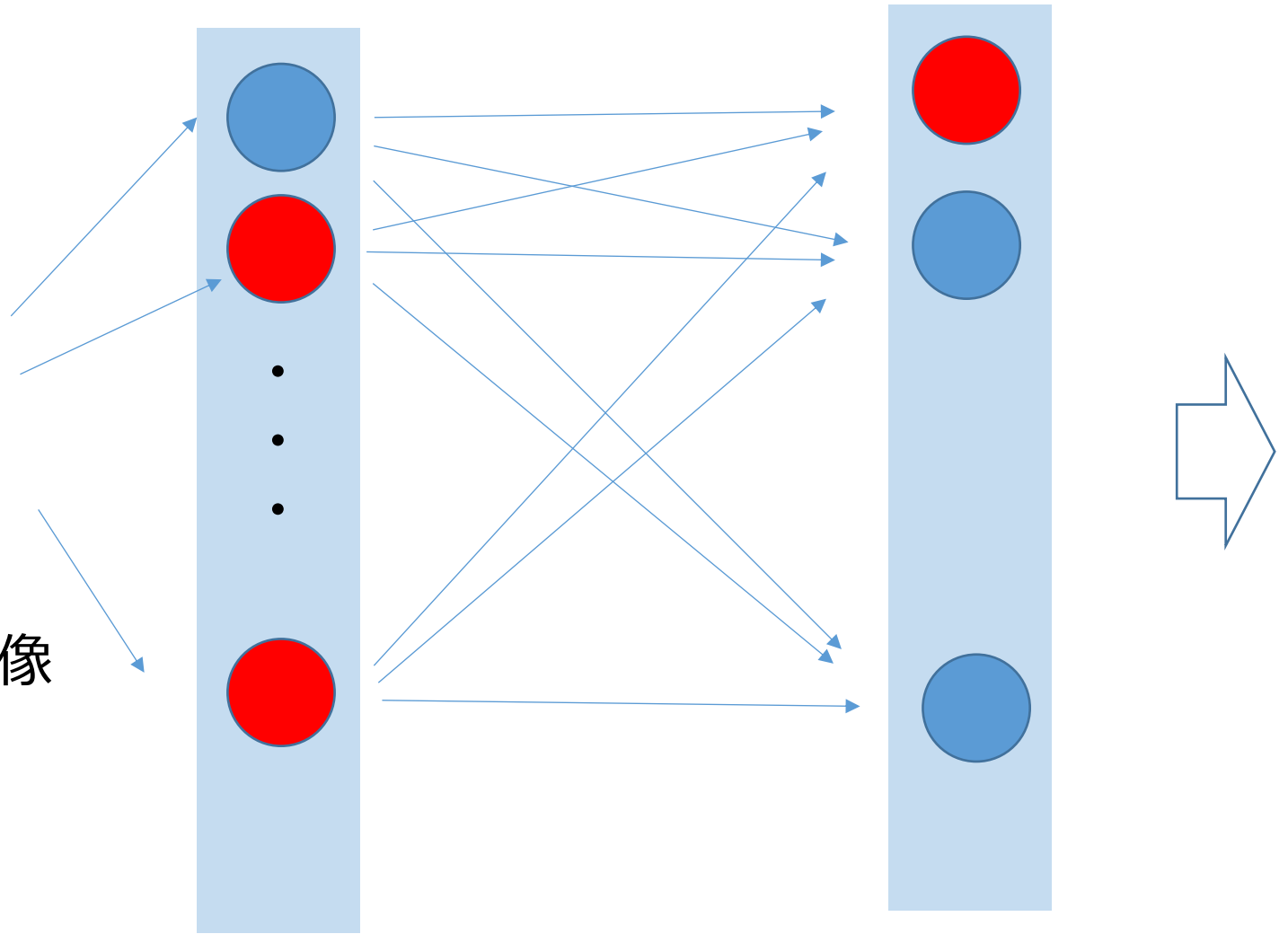
- **ニューラルネットワーク**での判定、予測は、**未知のデータ**を与えて、**最終層のユニット**を1つ**活性化**させることで行う

# 8-7 ニューラルネットワーク を作成するプログラム

# ニューラルネットワークによる予測



画素数  
**100** の  
モノクロ画像



ユニット数: **64**  
種類: **relu**

ユニット数: **10**  
種類: **softmax**

# ニューラルネットワークを作るプログラム



```
import tensorflow as tf
import keras
from keras.models import Sequential
m = Sequential()

from keras.layers import Dense, Activation
import keras.optimizers
m.add(Dense(units=64, input_dim=100))
m.add(Activation('relu'))
m.add(Dense(units=10))
m.add(Activation('softmax'))
m.compile(loss=keras.losses.categorical_crossentropy,
          optimizer=keras.optimizers.SGD(lr=0.01, momentum=0.9,
          nesterov=True))
```

入力は **100**個の数値  
1層目のユニット数は **64**

2層目のユニット数は **10**

# ニューラルネットワークの作成

- **ニューラルネットワーク**の作成は、
    - **ユニット**の数
    - **ユニット**の種類
- などを、**層ごと**に指定して行う

# 8-8 機械学習

## 8-8 機械学習



- **機械学習**とは、

与えられたデータ（教師データ）を使い、  
未知のデータに対しても当てはまる  
パターンや規則を、コンピュータが抽出  
すること

ノート  
ページ

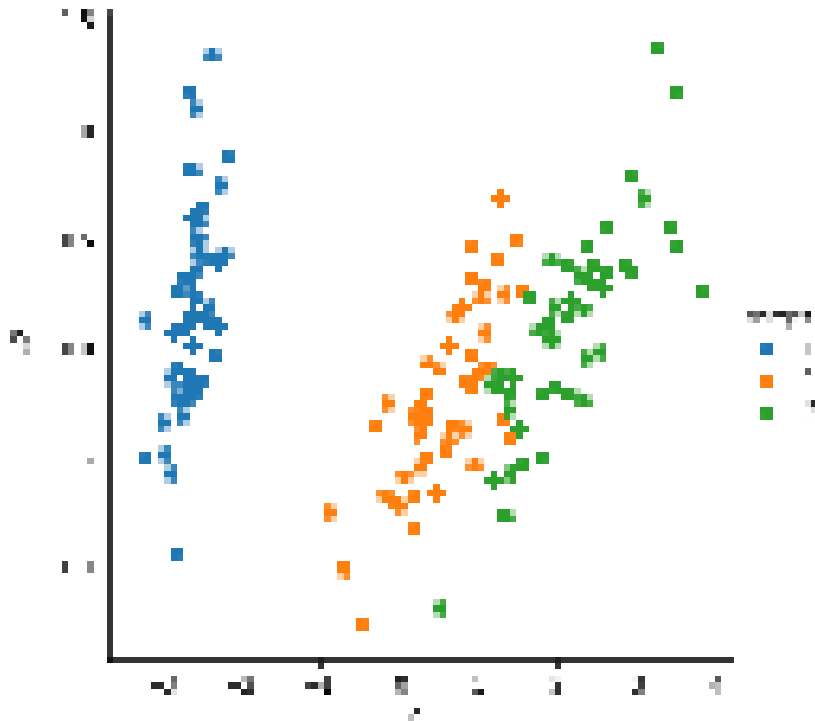


# 機械学習の用途



- 未知のデータの分類
- 予測
- 幅広い応用：画像認識，音声認識，自然言語処理，  
データ分析

# 教師データの例



教師データは、多数のデータの集まり

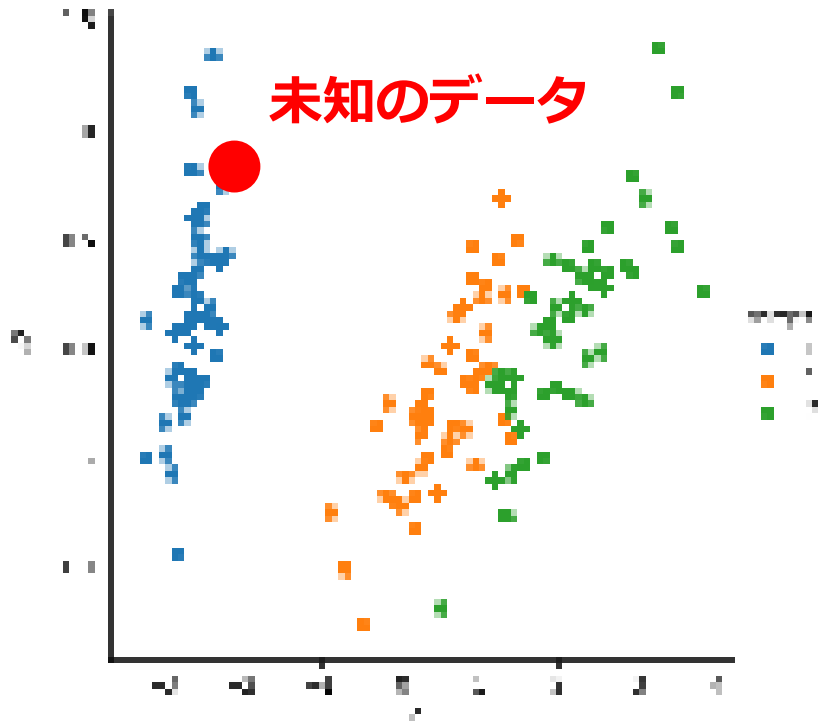
上の図では、点1つで、1つのデータ

Iris データセット

・ 3種, 150のアヤメの花びらのデータ

※ 右図は、主成分分析の結果のプロット

# 教師データによる予測



- 新しいデータ (未知のデータ) があるとき, 花の種類は何でありそうか
- 教師データの利用により, 未知のデータについても見通しを立てることが可能に

# おわりに



- **ニューラルネットワーク**など、**機械学習**を可能にする技術がある