CC BY

データサイエンス概論I＆II

パターン認識と分類

九州大学 数理・データサイエンス教育研究センター

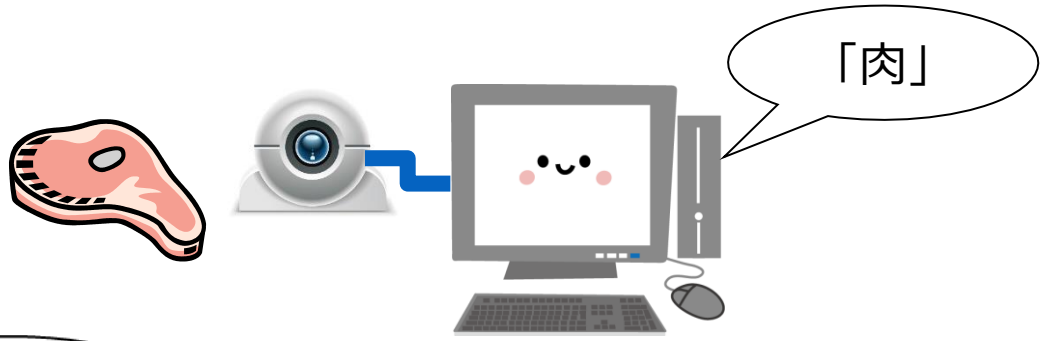
パターン認識とは？

皆さん，無意識にやってます

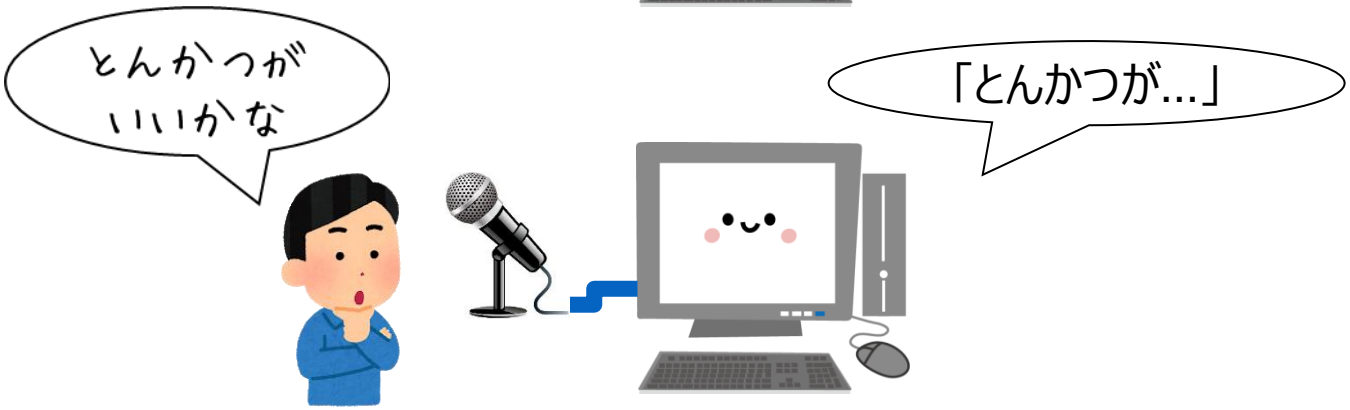
パターン認識とは何か？ ～ その目的

- 画像や音声，テキストなど，様々なデータを対象として，それが「何」であるかを当てる方法

- 画像認識



- 音声認識



色々なパターン認識

体温
咳の回数
…

病気診断

“風邪”

行動認識

“酔ってます”

感情認識

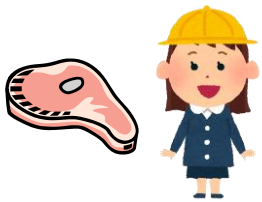
“ご機嫌
悪いです”

環境認識

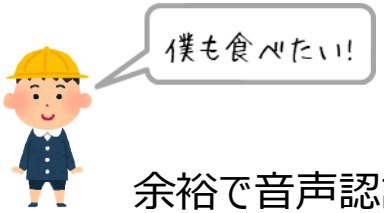
“前方安全”

人間には簡単で，コンピュータには難しいパターン認識：人間なら，無意識・高精度・高速に，認識実行可能

- データサイエンスなど全く知らない乳幼児ですら，パターン認識



余裕で画像認識



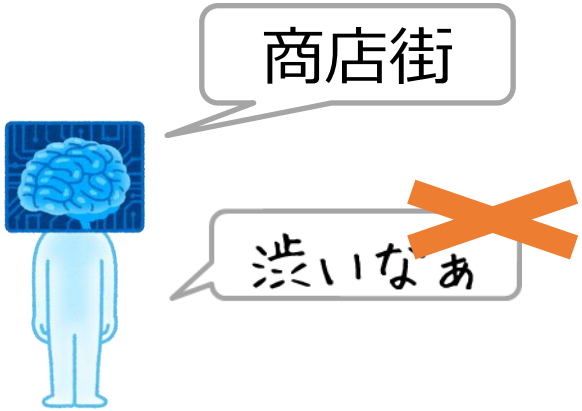
余裕で音声認識

- より高度な「雰囲気」や「おもしろさ」の把握なども

人間であれば，
この画像１枚から
雰囲気を含め
様々なことを
読み取れる

人間には簡単で，コンピュータには難しいパターン認識：コンピュータには，難しい「認識」がある

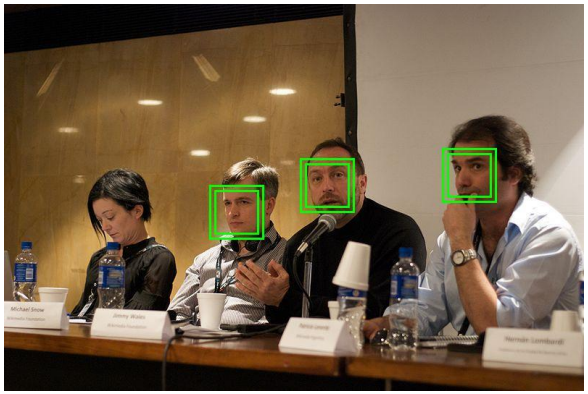
- しかしコンピュータにこの画像の「雰囲気」を認識させるのはまだ困難



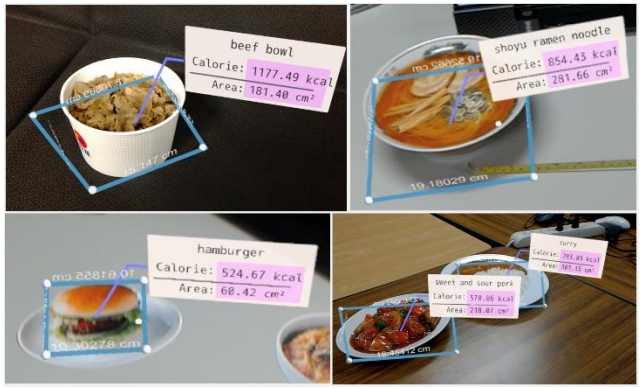
- 個々の物体等はある程度認識できても，それらの関係，全体的な雰囲気などは，まだ難しい

それでも身近になりつつあるパターン認識：画像認識の例

- 顔認識

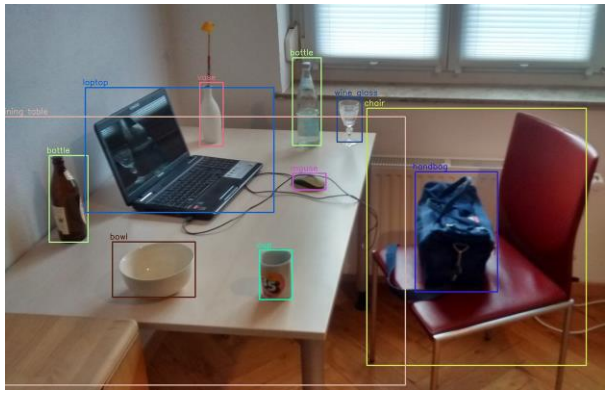


- 食事認識（さらにカロリー推定）



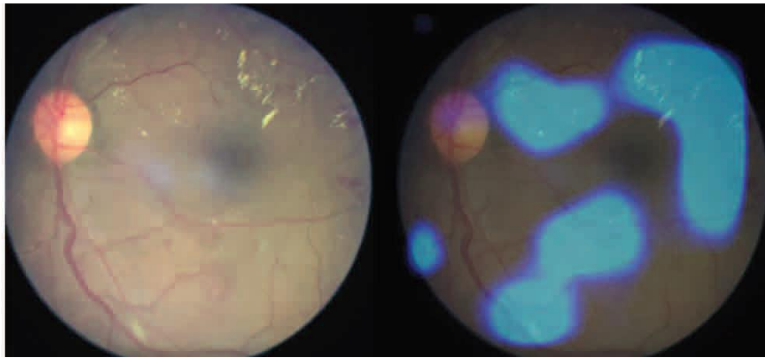
電気通信大学 柳井研究室
https://mm.cs.uec.ac.jp/research/?res=naritomi-s&utf8=1

- 物体認識



Mtheiler@Wikipedia 物体検出

- 医用画像認識

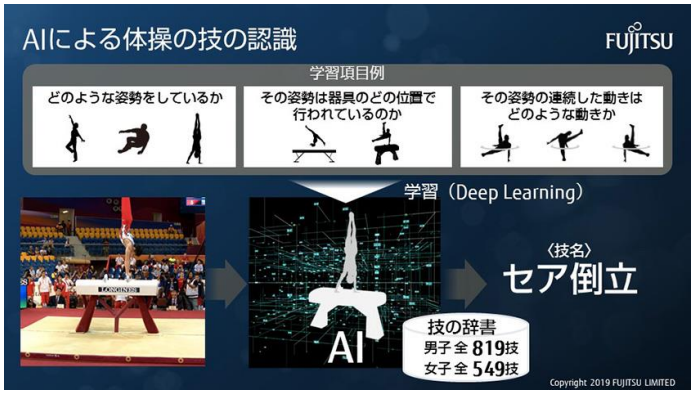


[Natarajan+, JAMA Ophthalmol., 2019]

それでも身近になりつつあるパターン認識：様々な認識

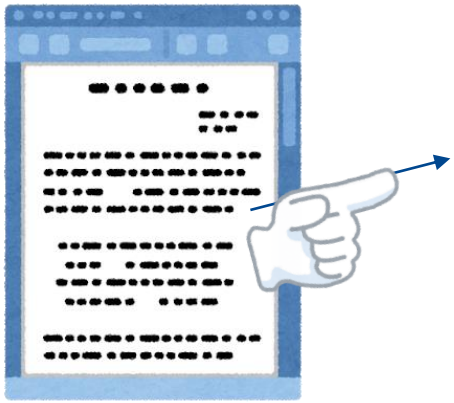
- 音声認識
  (スマートスピーカー)

- 行動認識



富士通「AIによる体操の技の認識」 https://blog.global.fujitsu.com

- テキストトピック認識



- 政治
- 経済
- スポーツ
- 国際
- 科学技術
- エッセイ

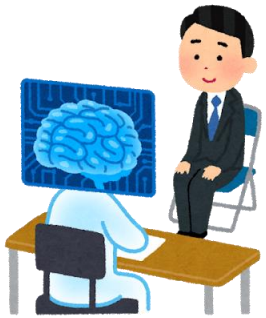
…

- 文字認識(郵便番号認識)

https://jpn.nec.com/kids/himitsu/04.html

パターン認識の応用～自動化技術

- 自動運転
  - 白線認識，先行車認識，歩行者認識，運転者の眠気認識
  - ロボットへのパターン認識搭載



- 自動診断
  - 病気の自動診断，機械の故障診断，適性診断
  - 食料品（果物・魚）の等級診断
- 無人店舗・無人工場・植物工場

- 自動採点…?!
  - 手書き答案の採点はまだまだ難しそう…

遊んでみたい人はどうぞ！ Trainable machine

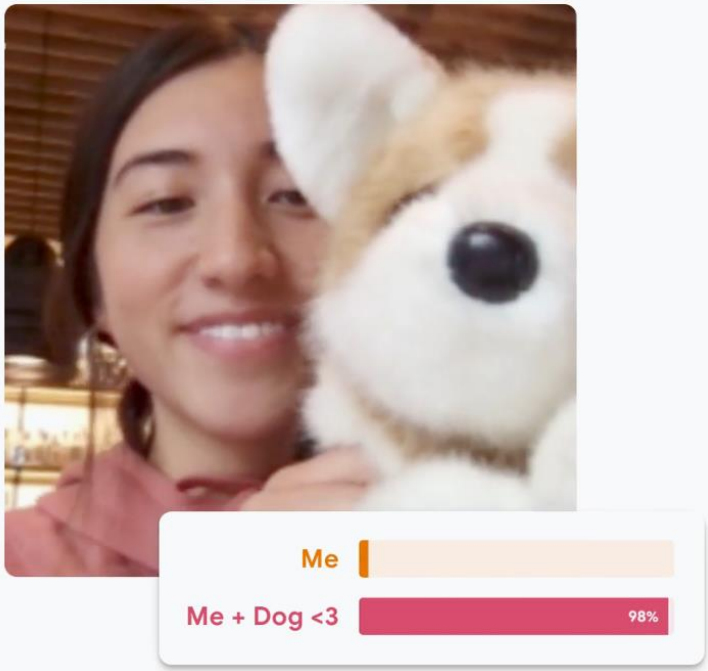
- https://teachablemachine.withgoogle.com/

Teachable Machine

独自の画像、音声、ポーズを認識するようコンピュータをトレーニングします。

サイト、アプリなどに使う機械学習モデルをすぐに、簡単に作成できる方法です。専門知識やコーディングは必要ありません。

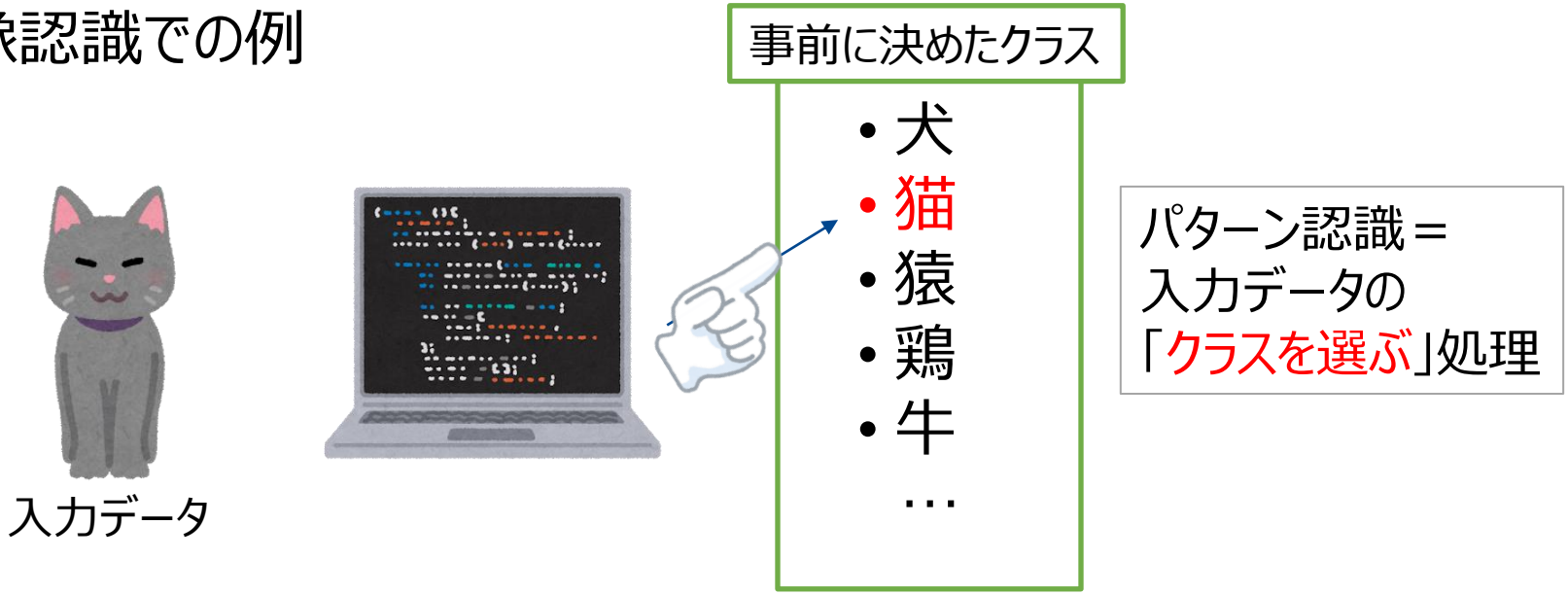
使ってみる



パターン認識＝「クラスへの分類」

パターン認識＝事前に決められているクラスへの分類

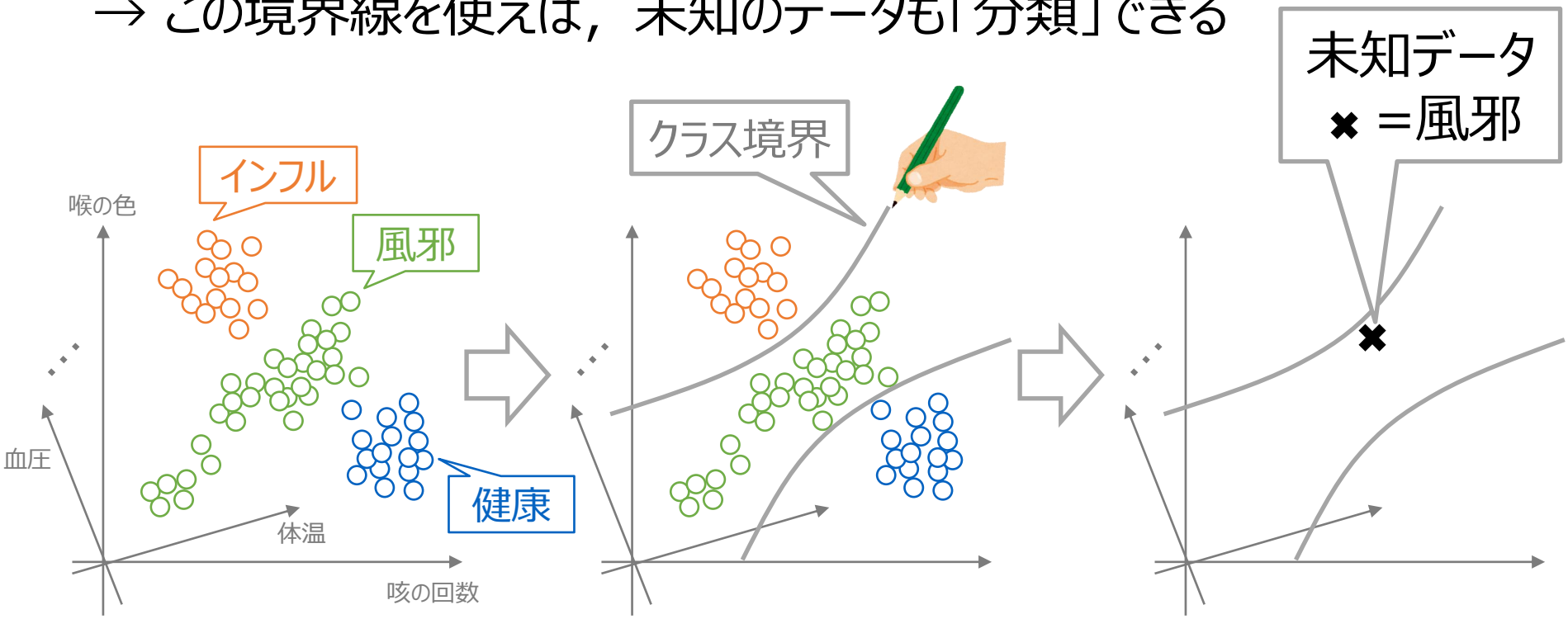
- 画像認識での例



- 従って，想定していないクラスには認識できない
  - {犬，猫，猿，…}などの動物クラスを対象としたパターン認識器では，自動車は絶対に認識できない(自動車クラスがないので)
  - 同様に{軽自動車，ワゴン車，トラック，…}などの自動車クラスを対象としたものでは，動物は絶対に認識できない

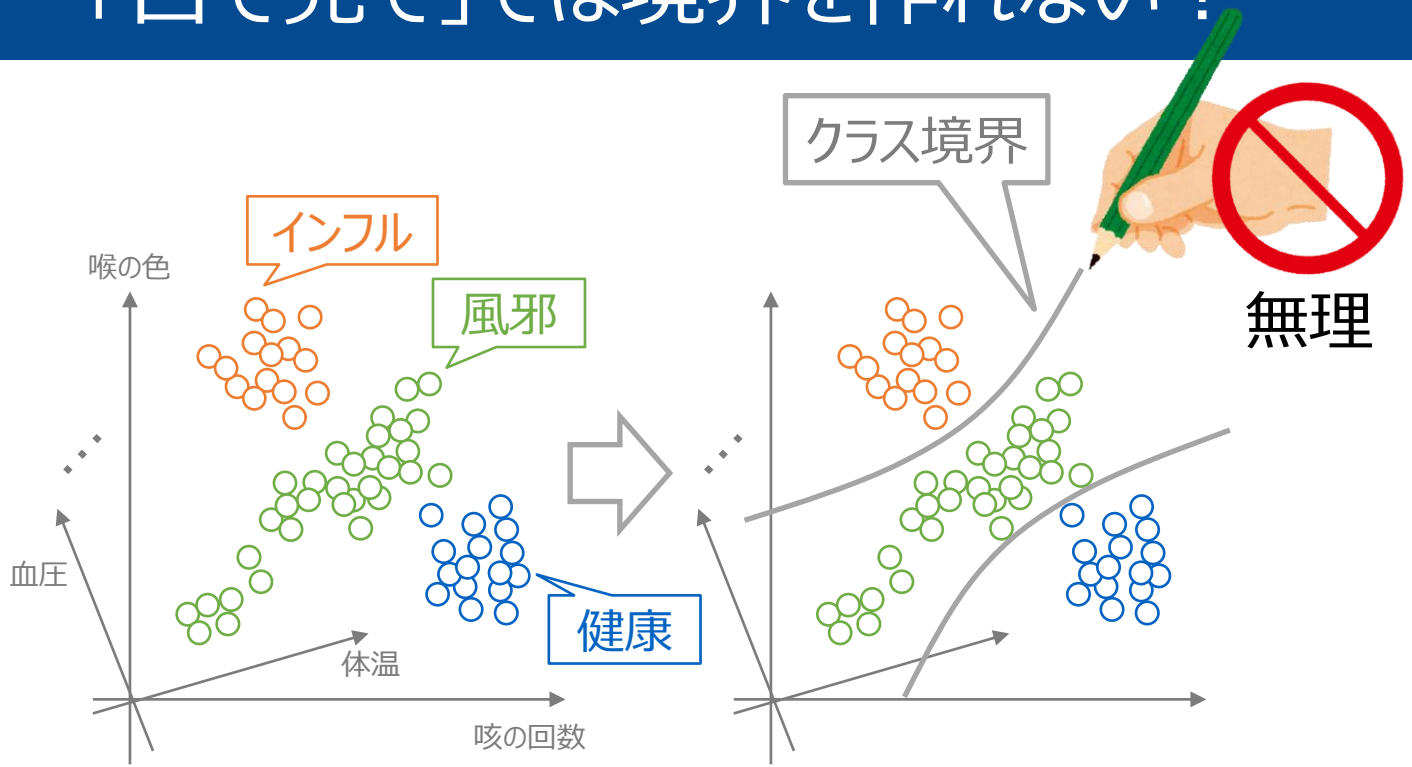
パターン認識を実現するための基本的な考え方

- 準備①：「クラスがわかっている」データを集める
- 準備②：それらに境界線(クラス境界)を引く
  → この境界線を使えば，未知のデータも「分類」できる



- しかし…

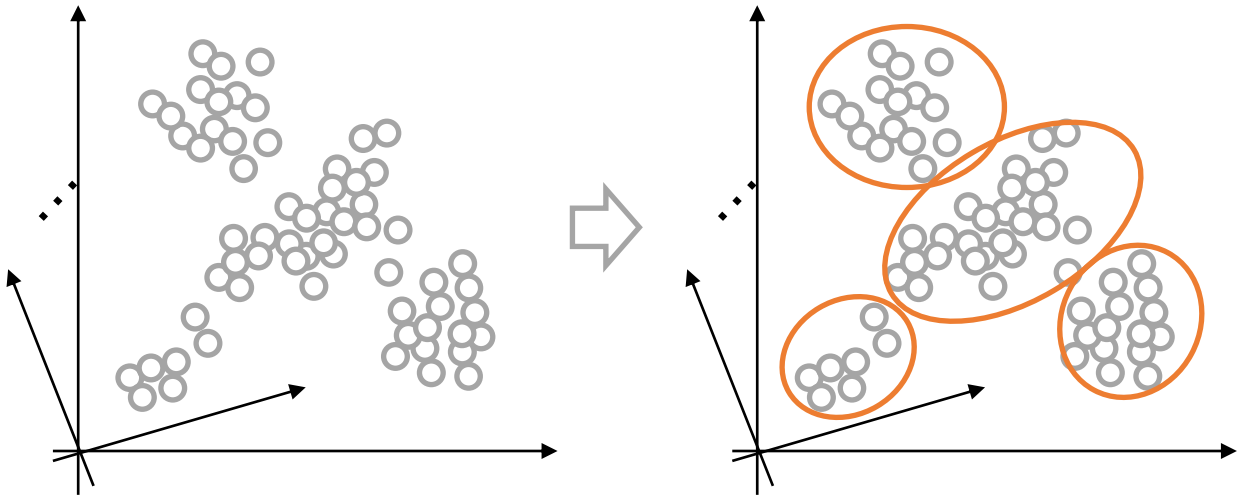
データ数が増える＆高次元になると，「目で見て」では境界を作れない！



- なので，与えられたデータから，自動でクラス境界を定めたい！
- そこで「機械学習」を利用
  - 最近傍法
  - 識別関数法

参考：以前学んだ「クラスタリング」は，パターン認識とはちょっと違う

- クラスタリング：似たデータを同じグループにわける手法
  - 各データが何(=どのクラス)のグループかは不明



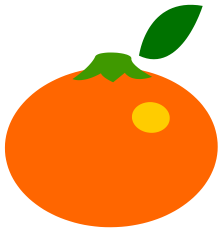
- パターン認識：どのクラスかわかっているデータをつかって境界線を作って，それを用いて未知データを分類

最近傍法：
最も基本的なパターン認識

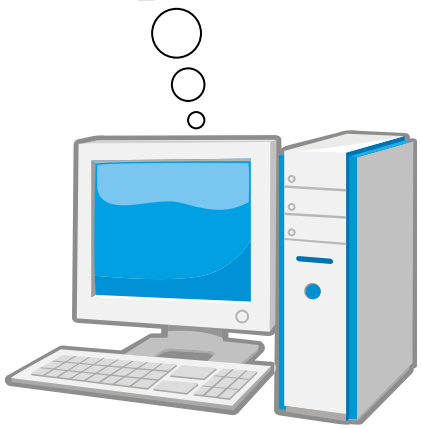
最も簡単な（？）機械学習！
単に「お手本」を計算機に覚えさせるだけ

コンピュータによる「パターン認識」の基本原理：食べ物画像認識を例に

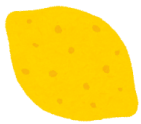
自分が知っている物体のどれと
最も似ている？
→(たぶん)みかん！



入力画像



コンピュータにあらかじめ
登録されている
“お手本画像”とその物体名



「レモン」

「鯛」

「牛肉」

「豚肉」

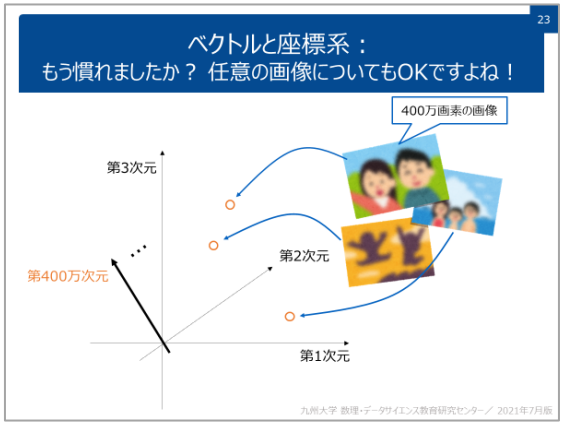
「みかん」

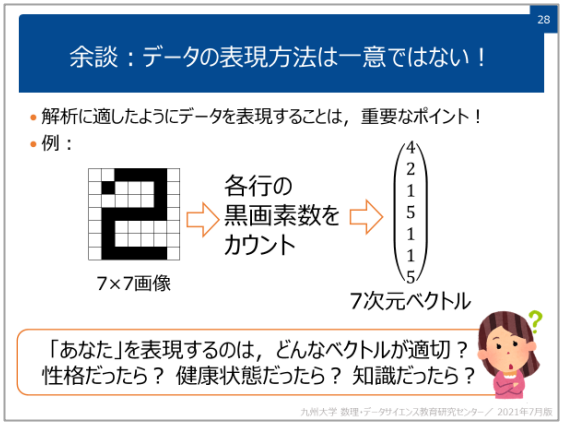
「すいか」

この「みかん」とか
「鯛」が「クラス」に相当.
この例では6クラスが存在

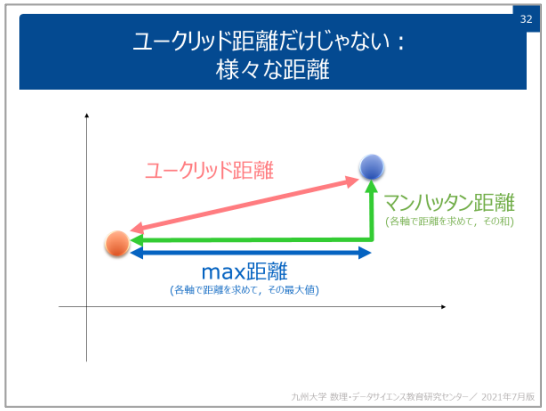
この基本原理について，もう少し踏み込んで考えると，以前習ったことと関係してきます

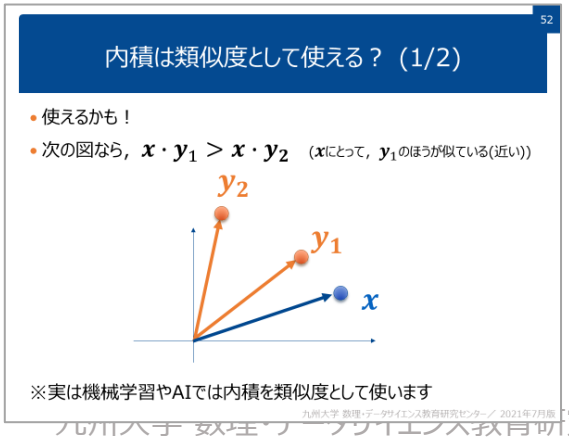
- 画像のようなデータをどうやってコンピュータに扱わせる？





- 似てるとは何だ？

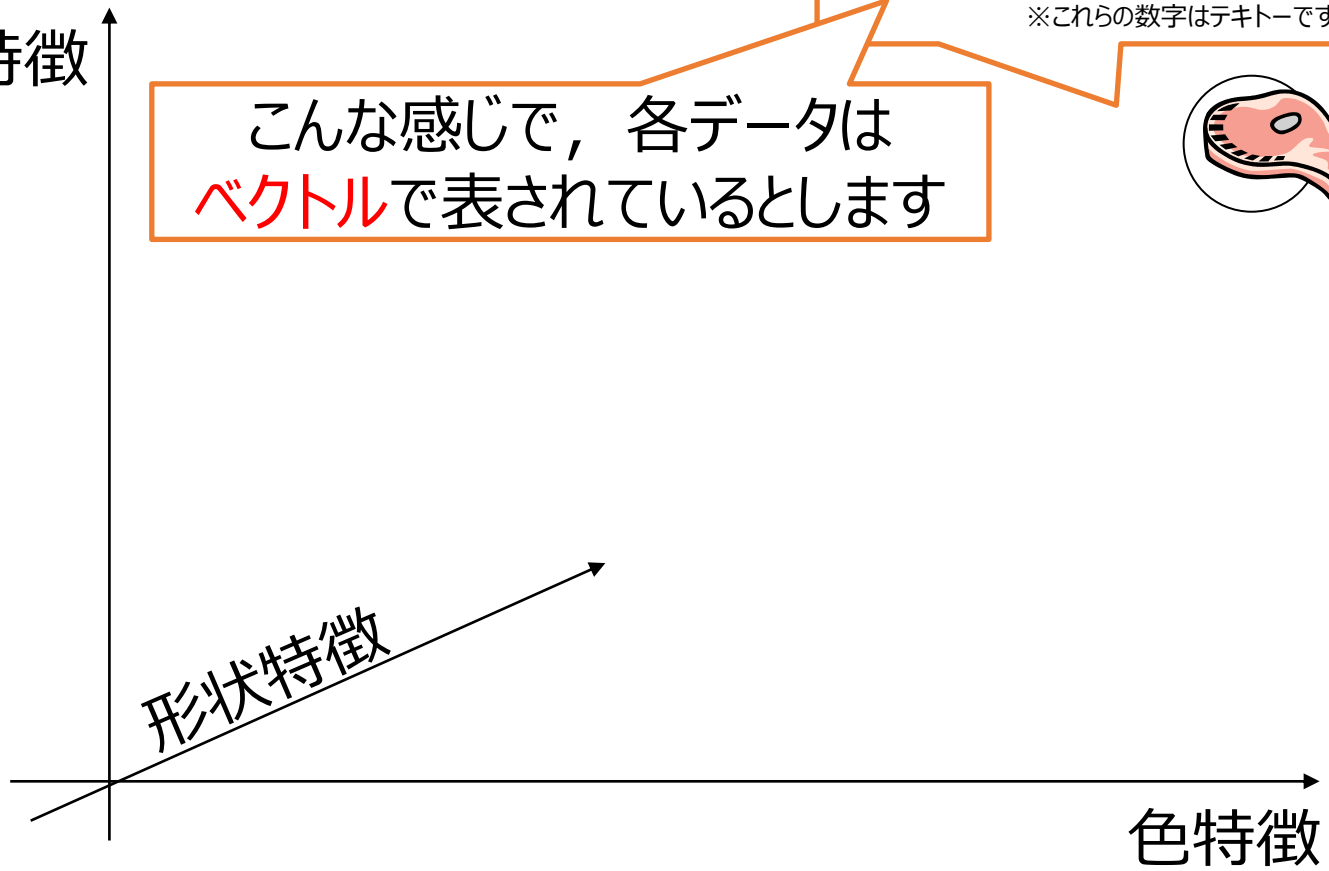




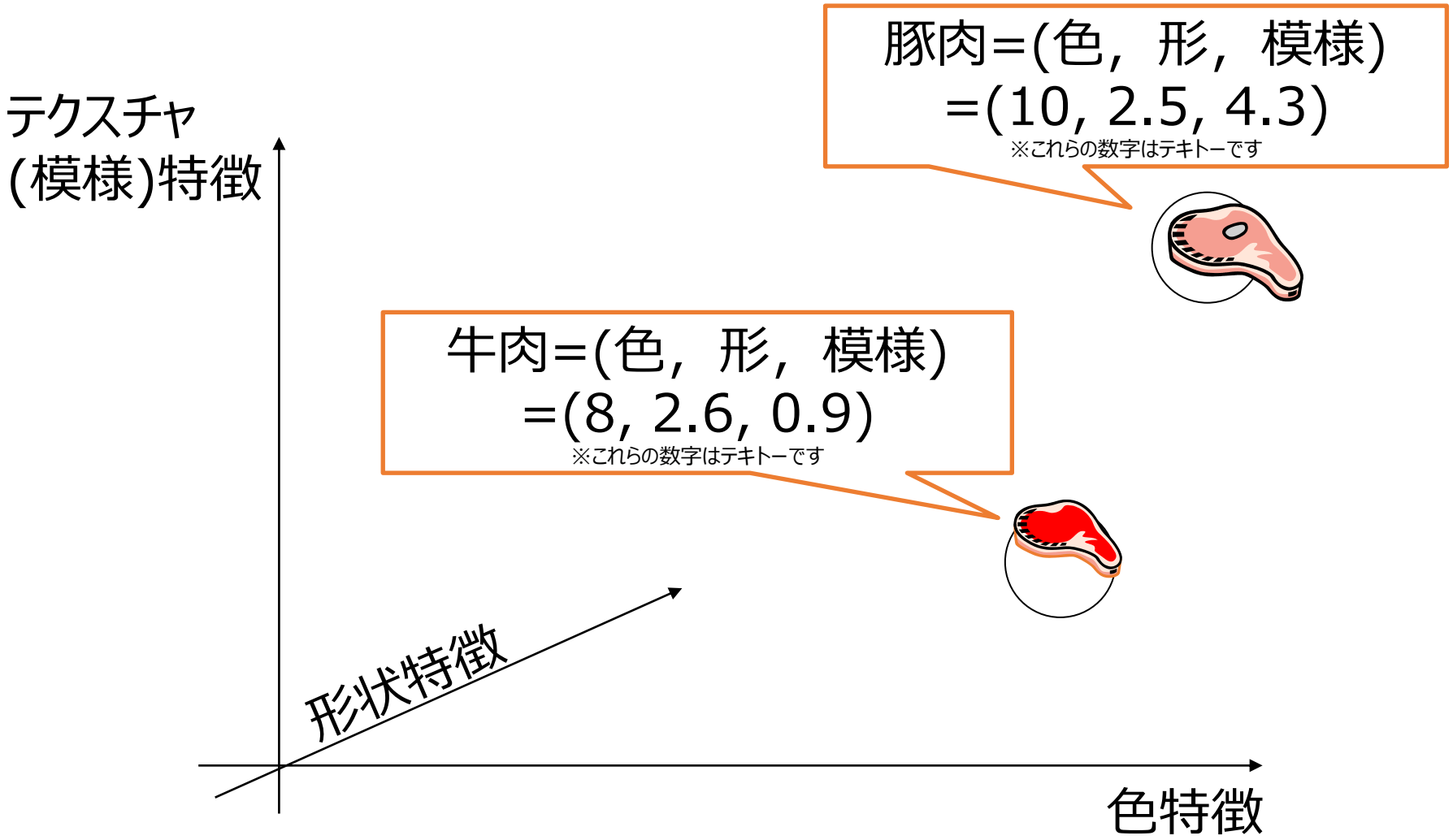
準備：データのベクトル表現(1/2)

豚肉=(色，形，模様)
=(10, 2.5, 4.3)
※これらの数字はテキトーです

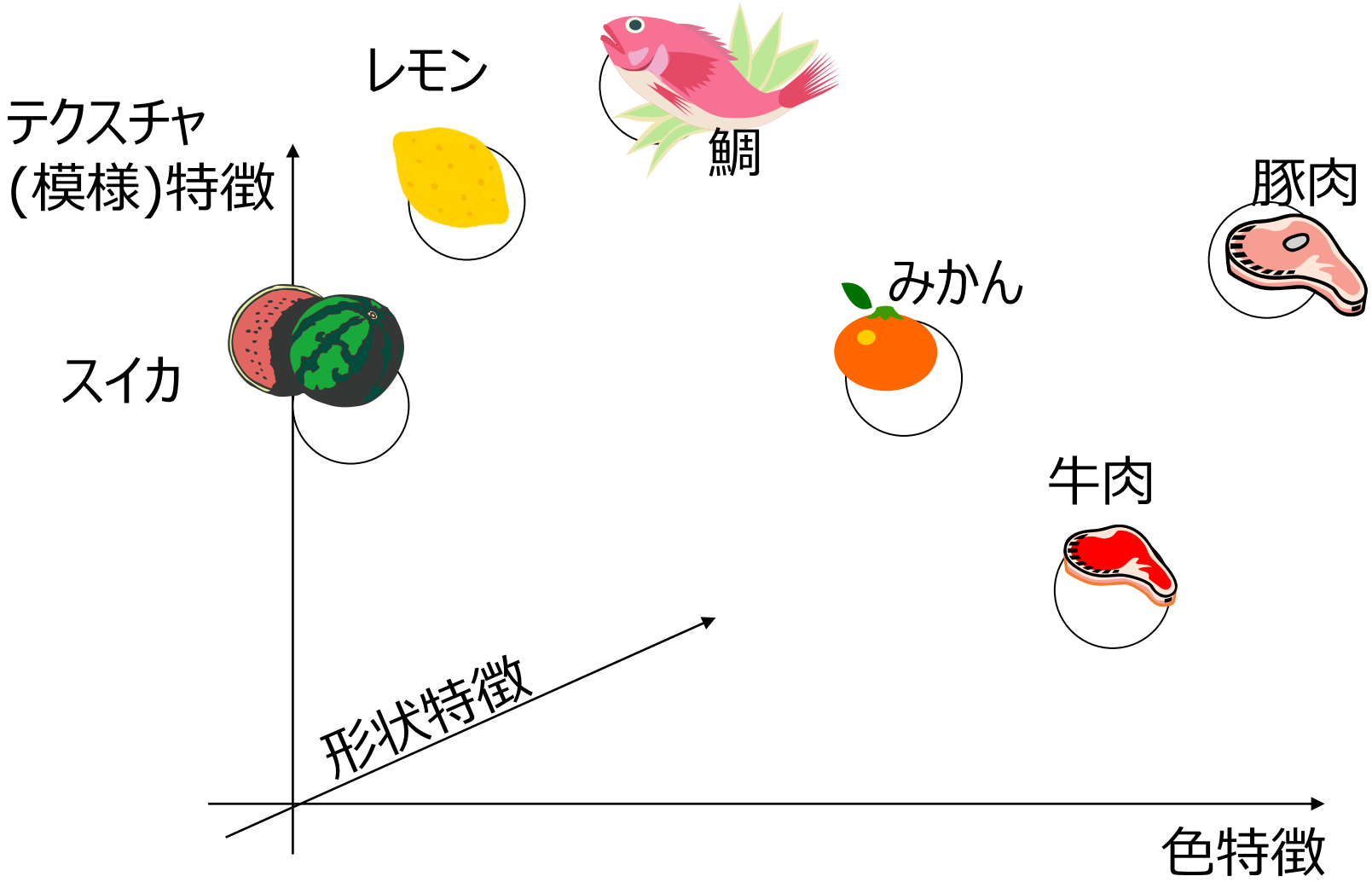
こんな感じで，各データは
ベクトルで表されているとします



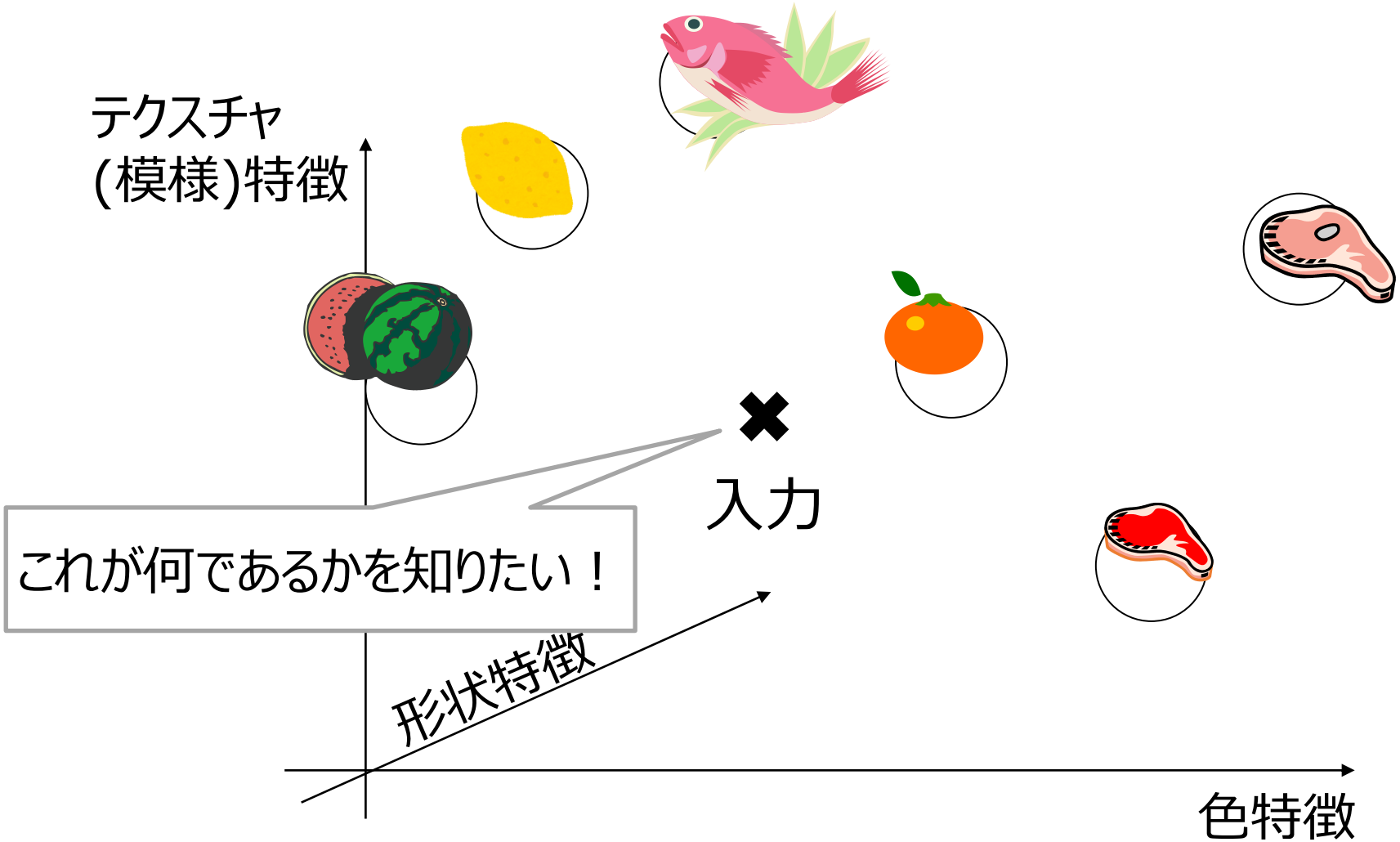
準備：データのベクトル表現(2/2)



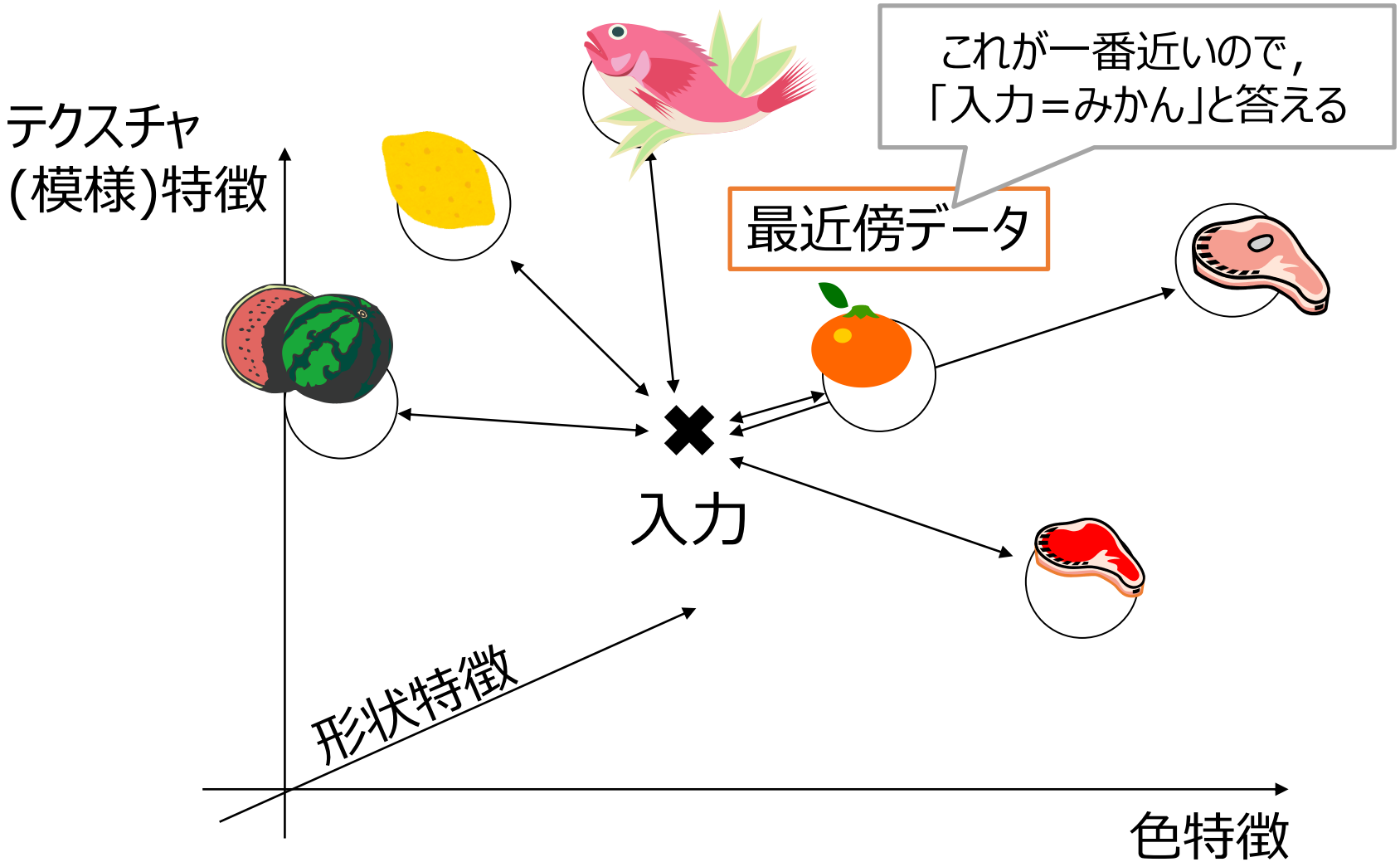
計算機に(クラス付き)データを覚えさせる！



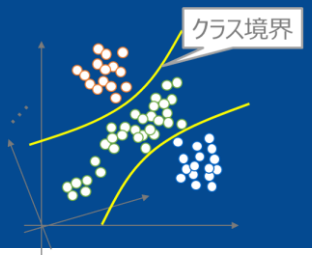
さて，クラス未知の入力データ

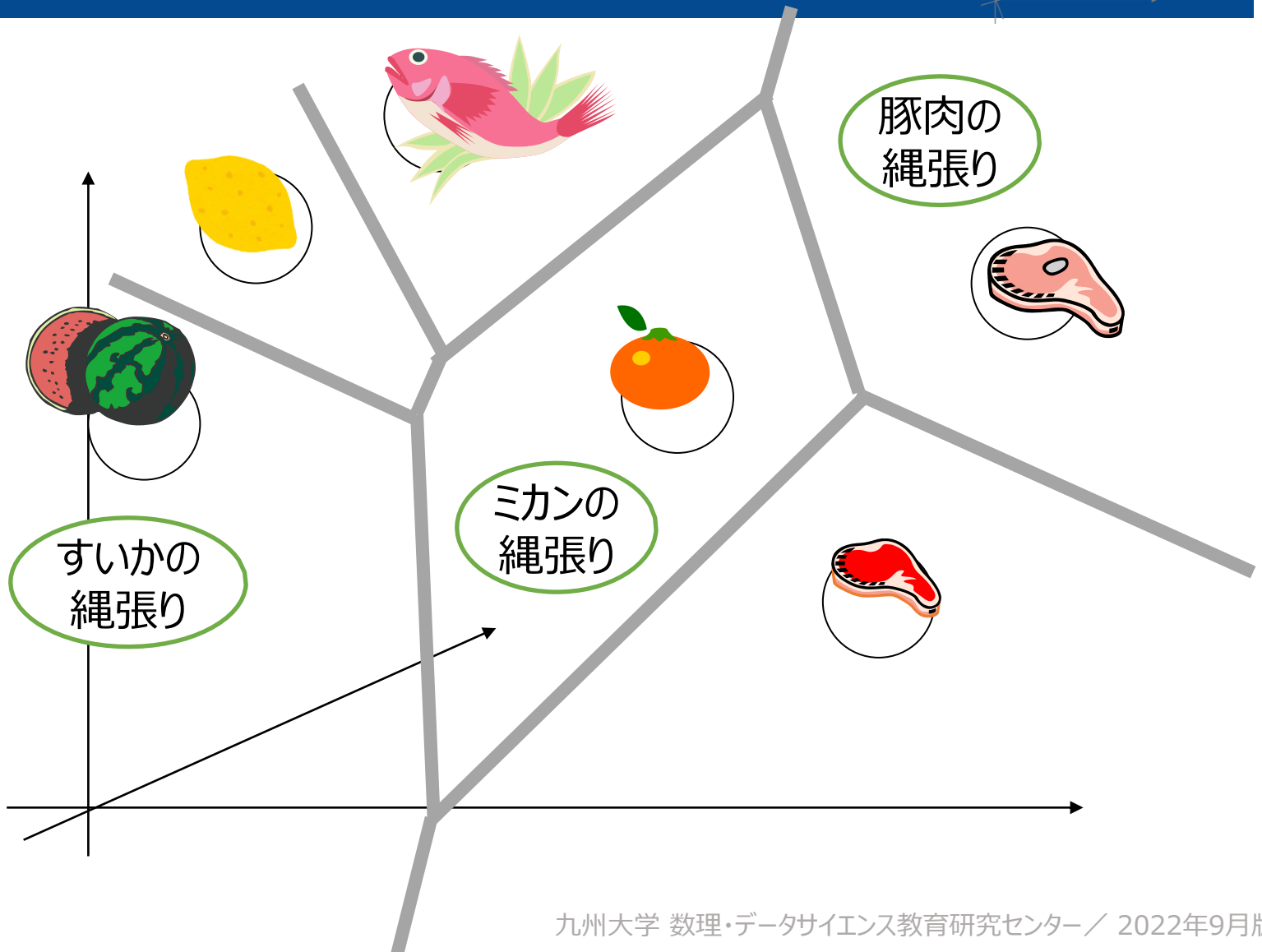


最近傍法：
距離最小(＝類似度最大)のクラスに分類

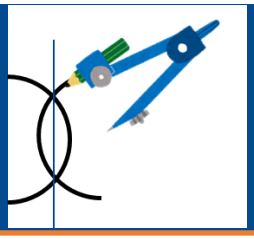


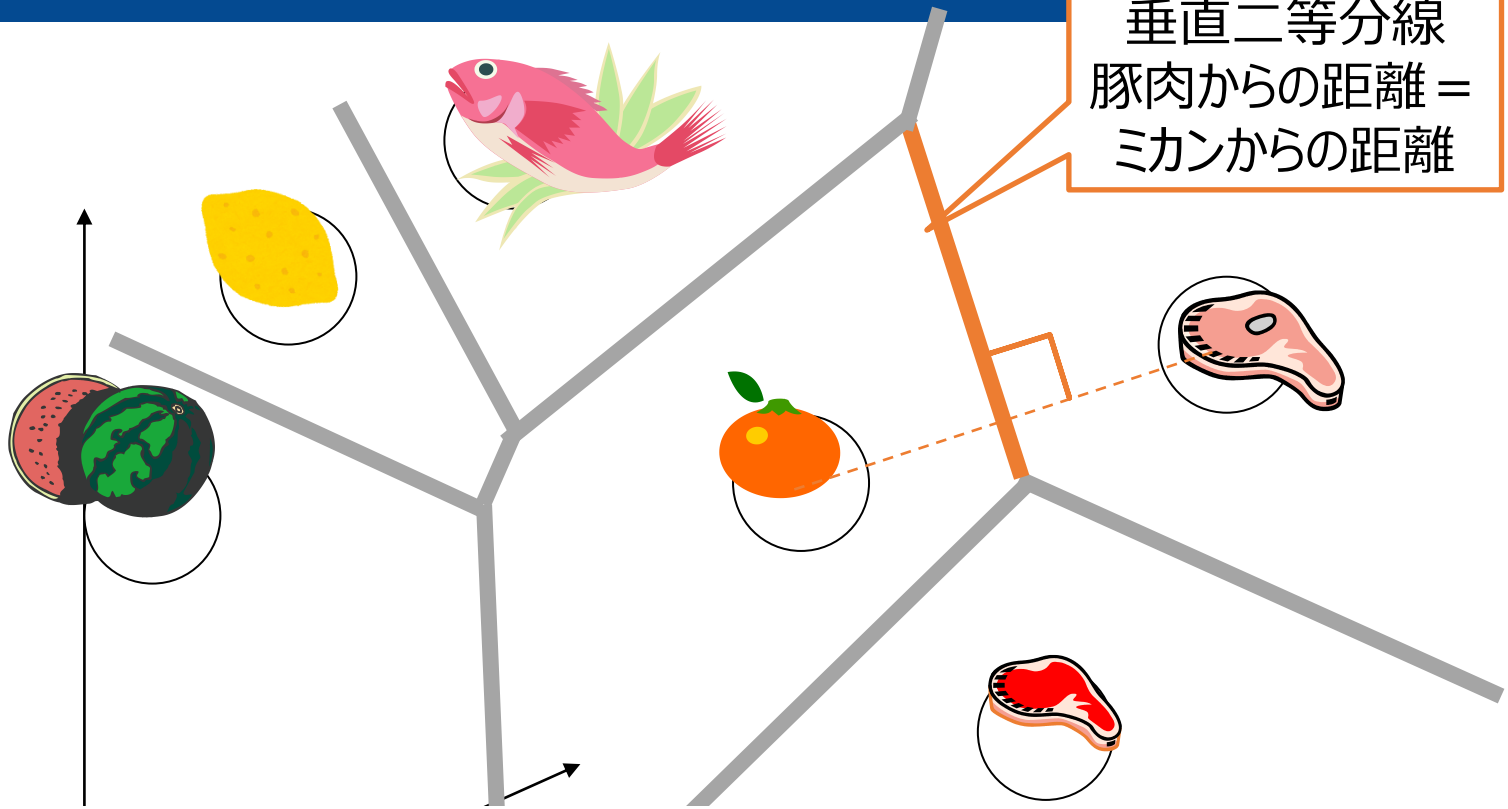
あれ？ 境界線は？
→ 実は暗に「なわばり」がある





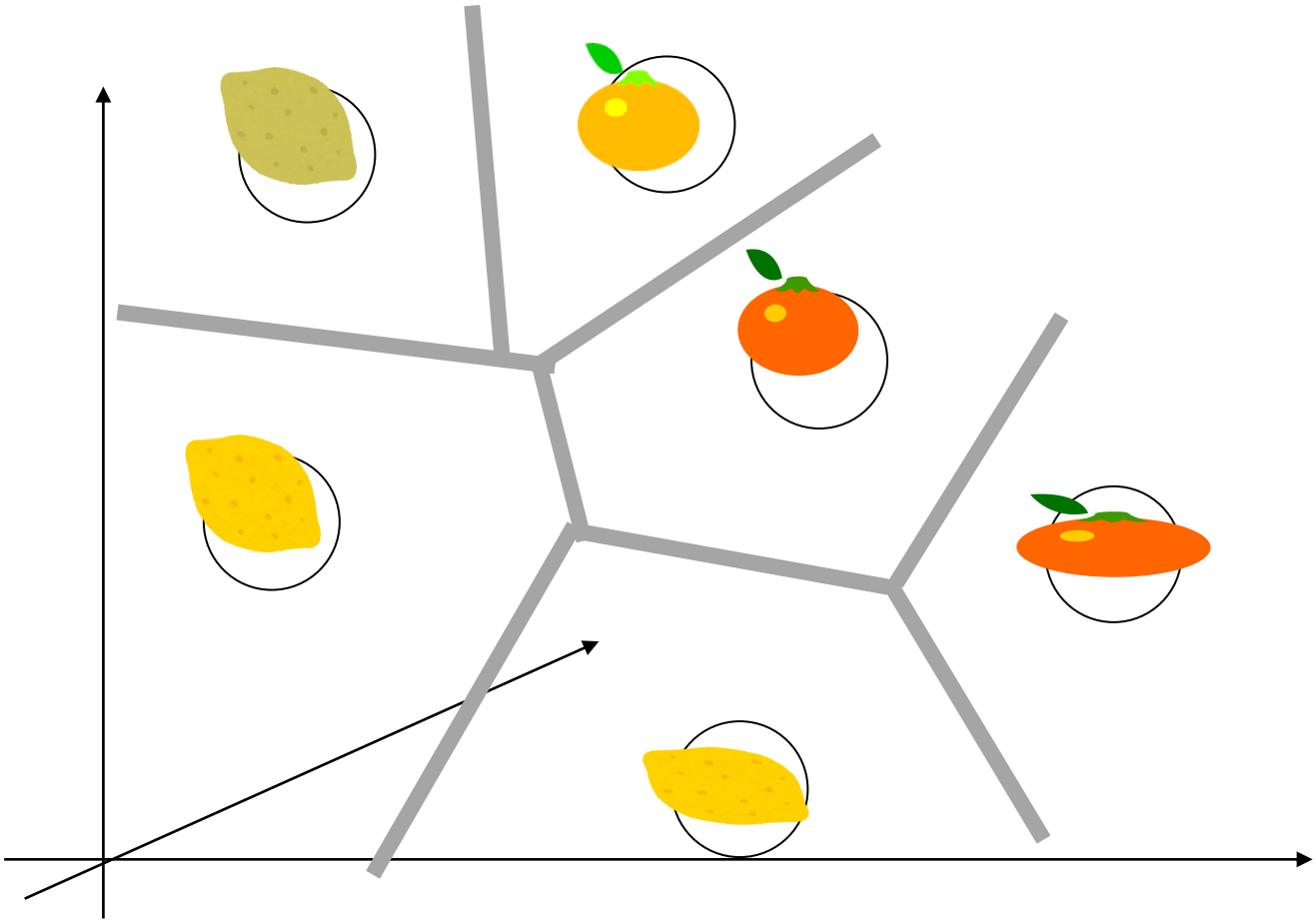
なわばりの作り方



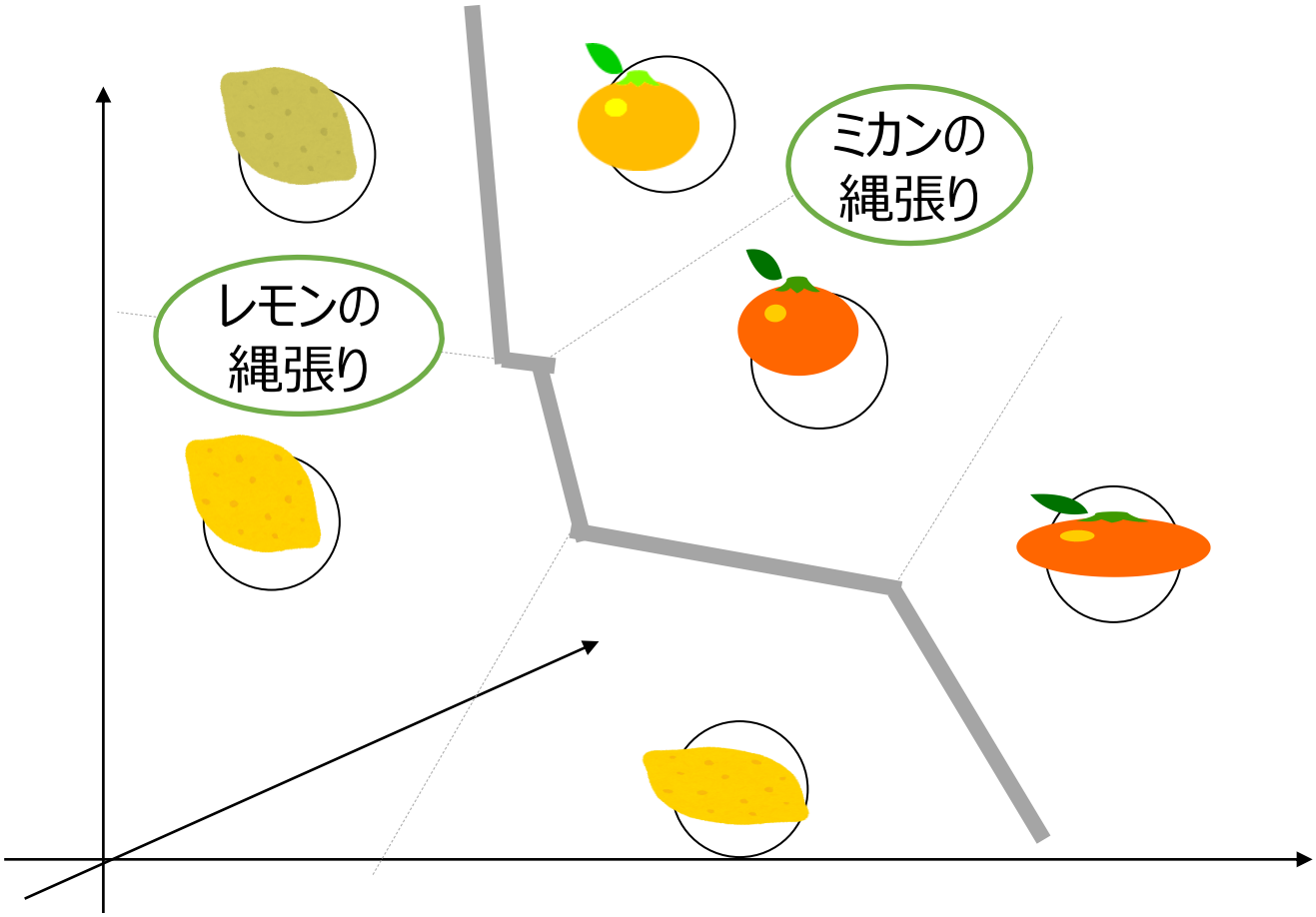


①この絵ではユークリッド距離で縄張りを作ってます
②この縄張りの様子は「ボロノイ図」と呼ばれます
　（校区の設計とか，分類以外でも色々使われます）
③本当は2次元ではないので，境界「面」になります(正確には超平面)

1 クラスあたり，複数個を覚えさせてもOK!
「ミカン vs レモン」分類の場合

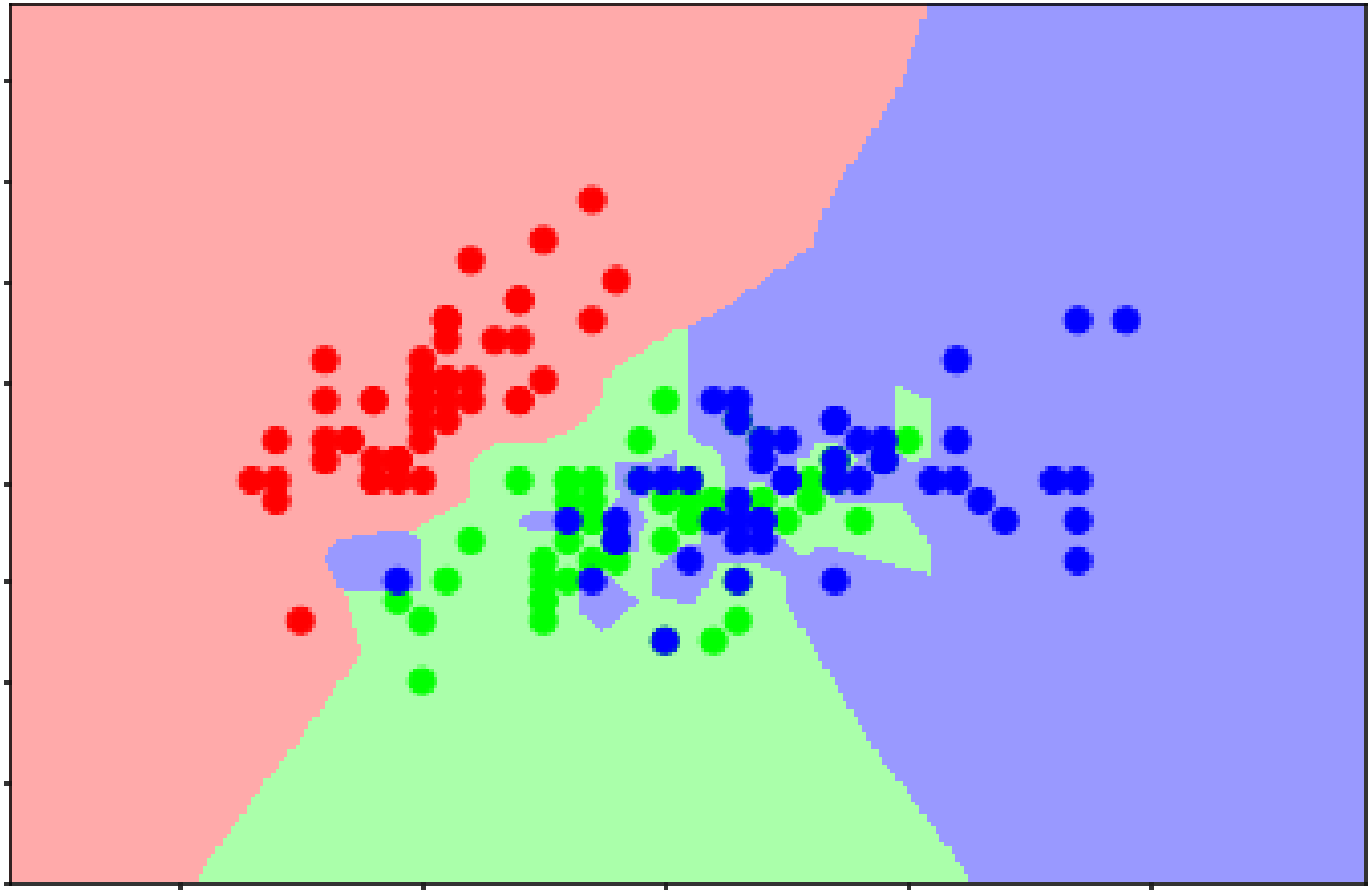


1 クラスあたり，複数個を覚えさせてもOK!「ミカン vs レモン」分類の場合



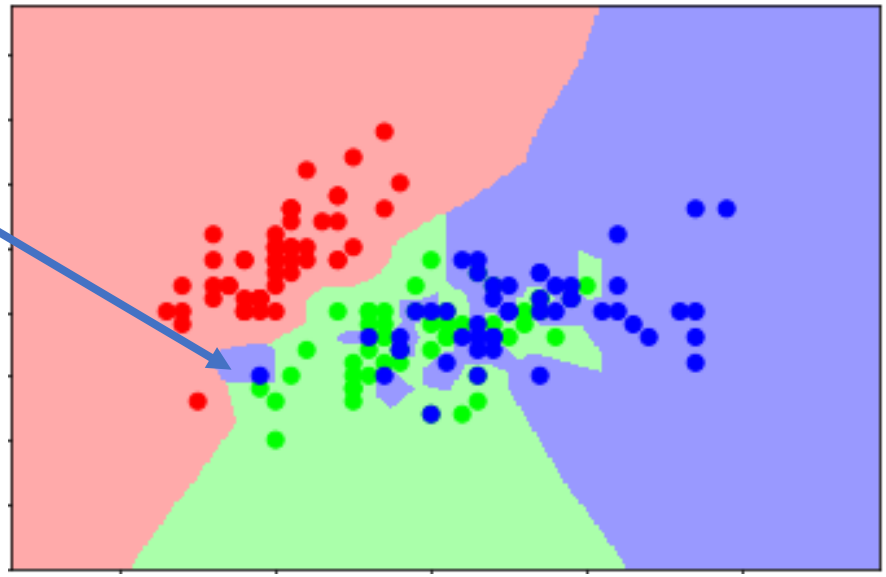
より複雑な分類境界が作られる！

３クラスの場合の縄張りの様子
(2次元．各クラスに多数のデータを登録)

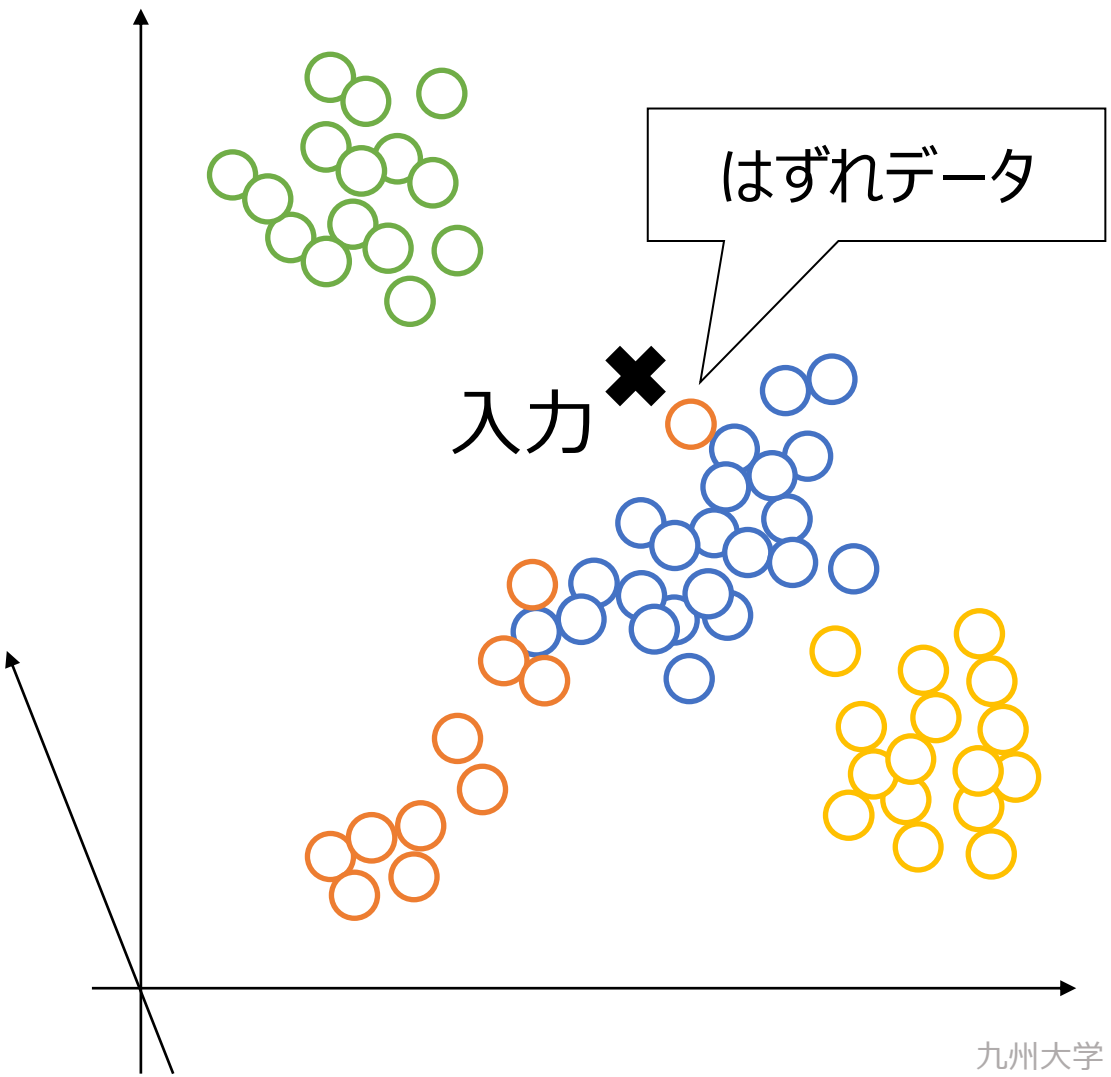


最近傍法の性質

- 非常に簡単！
  - 登録されているデータとの距離を測るだけで，分類可能
- 各クラスにつき複数のデータを登録することで，複雑な分類も可能
- 「はずれデータ」の影響大！



k近傍法 (k-nearest neighbor, k-NN)



入力 ✖

1番近い ○

2番近い ○

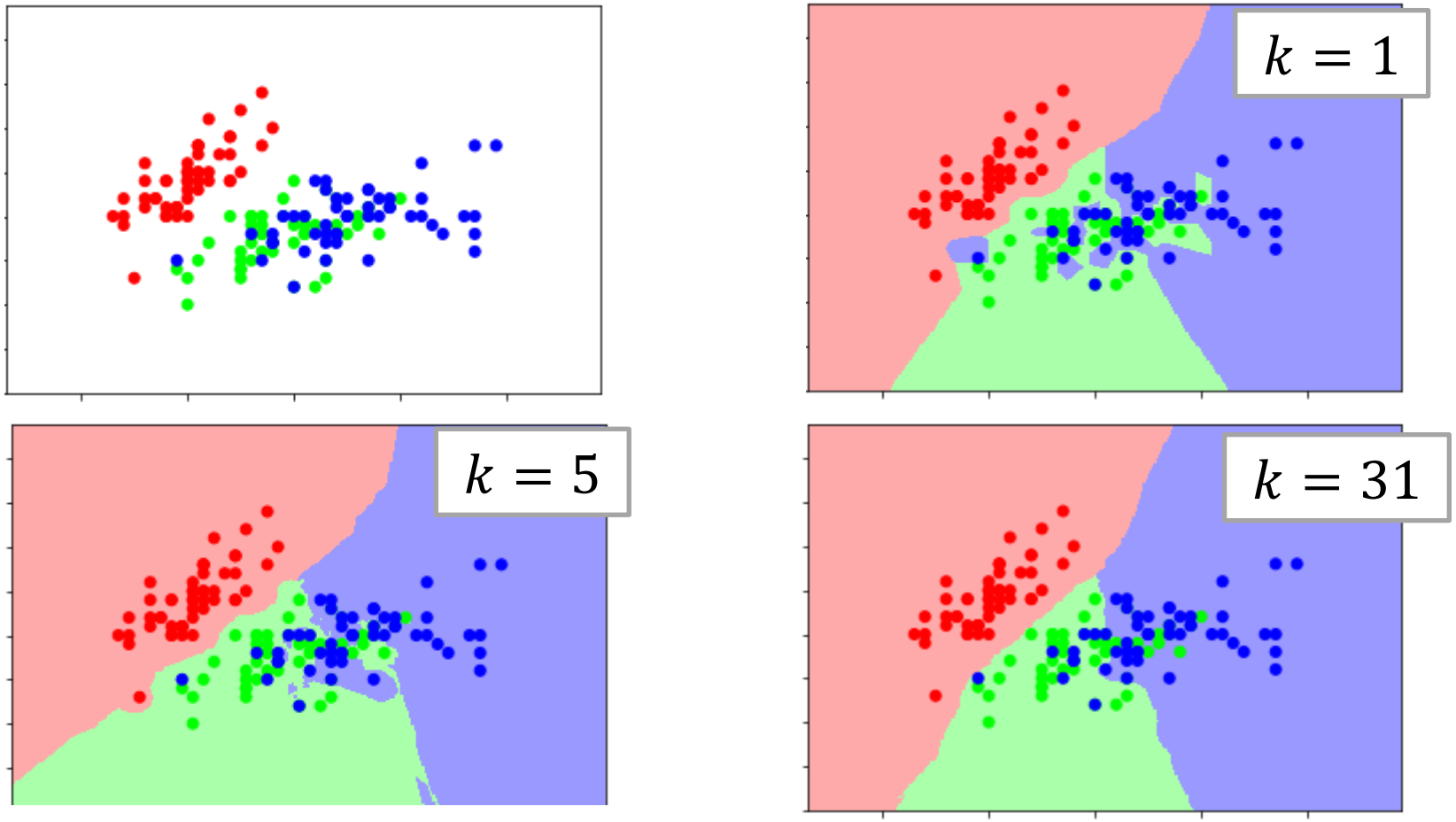
3番近い ○

多数決！

1NNだと ✖ → ○

3NNだと ✖ → ○

k近傍法：kによる境界の変化



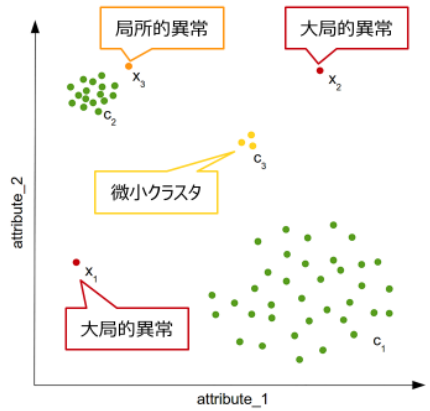
①kを大きくすると，それだけ平滑な境界が求まる
②それが「分類結果に好影響を与える」とは限らない

参考：以前やった異常検出にも出ていたk近傍法

異常の種類(1/3)

28

- 大局的異常
  - まわりに「近い」データがない
    「似た」データがない
    ＝どう考えても異常
  - x_1, x_2



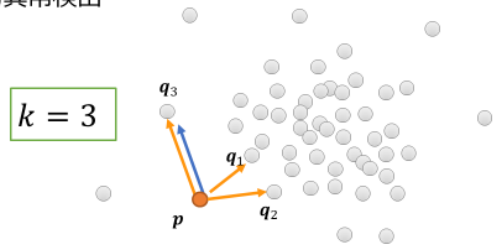
[Goldstein, Uchida, PLoSONE, 2016]

k近傍法による異常検出(1/3) 原理

31

k個の近いデータ

- 単一タイプ：k番目に近いデータへの距離 $\|\boldsymbol{p}-\boldsymbol{q}_k\|$ or
- 合計タイプ：$\|\boldsymbol{p}-\boldsymbol{q}_1\|+\|\boldsymbol{p}-\boldsymbol{q}_2\|+\cdots+\|\boldsymbol{p}-\boldsymbol{q}_k\|$
- 大局的異常検出

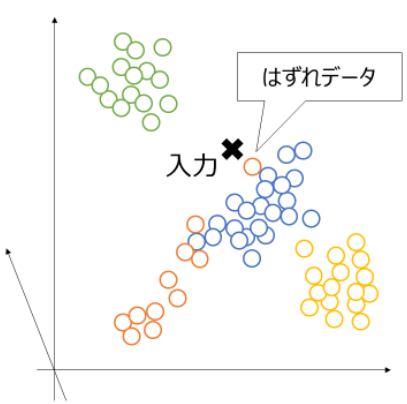


- $k-1$個のデータからなる微小クラスタも異常として検出可能

似ている※

k近傍法 (k-nearest neighbor, k-NN)

30



入力 ✖

1番近い ○

2番近い ○

3番近い ○

多数決！

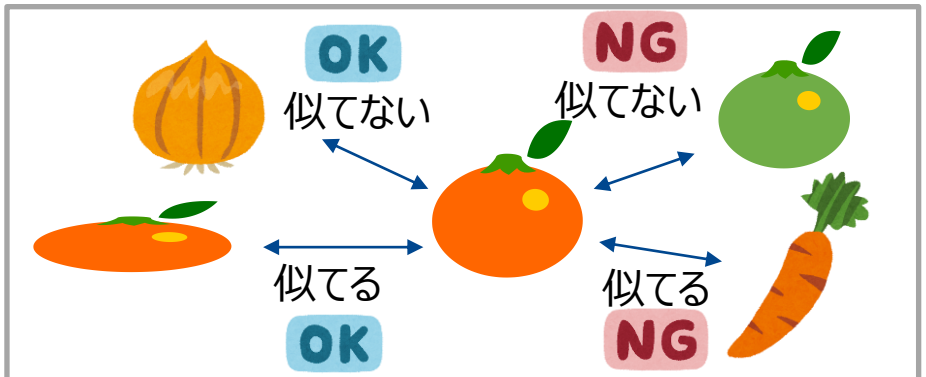
1NNだと ✖ → ○

3NNだと ✖ → ○

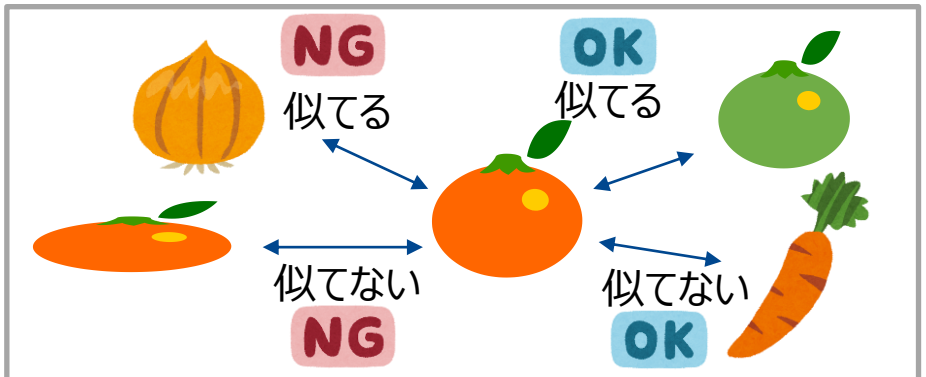
※実際にやっていることは少し違うが，上位k個を使う点では同じ

パターン認識における「類似度・距離」の考えどころ

- 「どこが」似ている？
  - 要するに「どのような特徴を使って類似度を評価するか」？

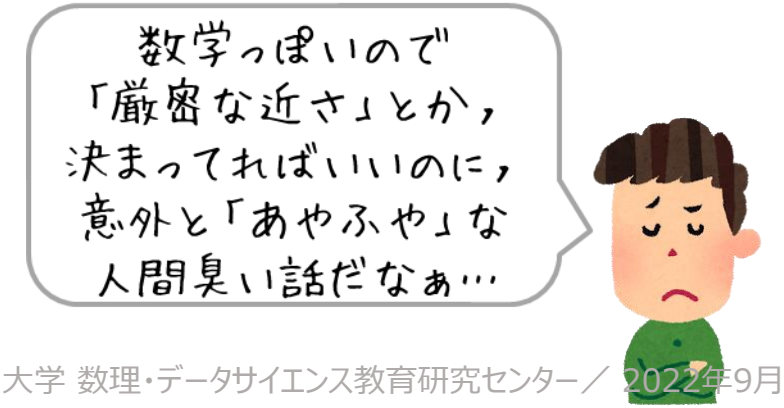


「色」に注目した類似度だと…



「形」に注目した類似度だと…

- 「どれぐらい」似ている？
  - 色の似てる具合をどう測る？
  - 形の似ている具合をどう測る？



この辺，深入りすると~~はまります~~面白い

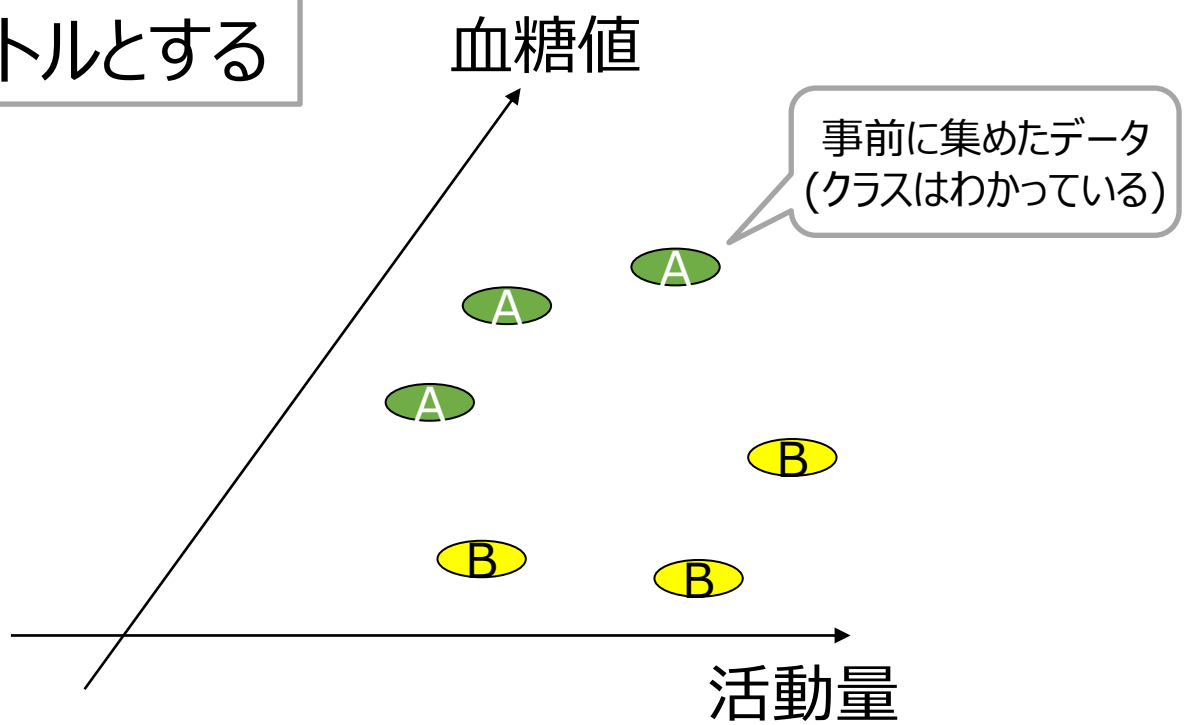
- そもそも「ミカン」って何だ？
- 食べられない（腐った）ミカンはミカンなのか？
- 食べ終わったミカンの皮はミカンなのか？
- ジュースにしたものはミカンなのか？
- 温州ミカンとオレンジは同じなのか？
- レモンとミカンの境界はあるのか？
- 「平均的なミカン」というのはどんなミカン？
- どうして な絵で「ミカン」とわかるのか？
- どうして「ミ」＋「カ」＋「ン」の３文字が特定の果物と結びつくのか？

- …
- などなど，認識論はギリシャ哲学時代からの難問です..
- 面白いのは「なぜか人間はそれでも生きていける」ところ…

識別関数法

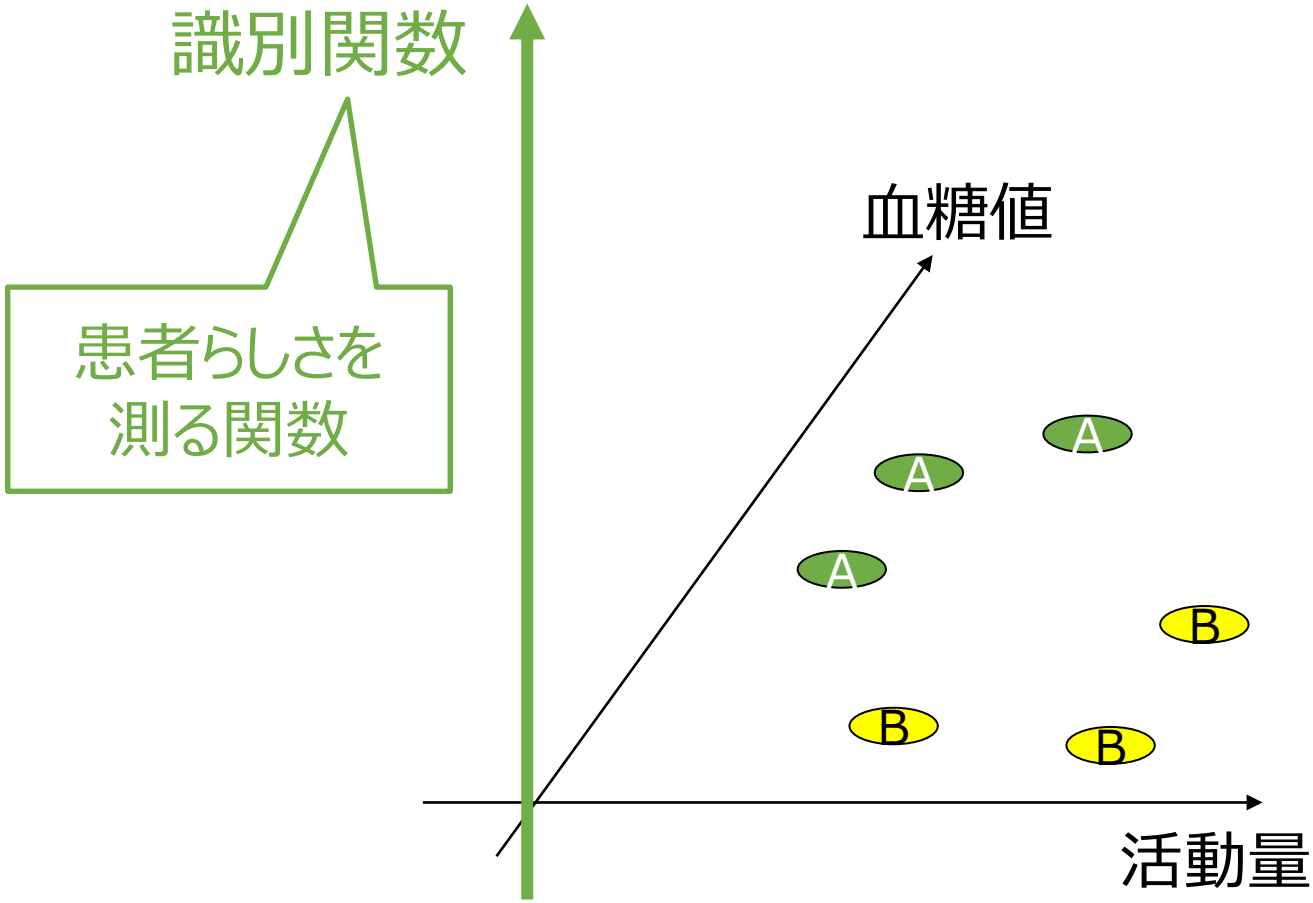
識別関数とは？
例：診断マシーンを作りたい！

各データ＝
2次元ベクトルとする



A: 患者(糖尿病)
B: 健康

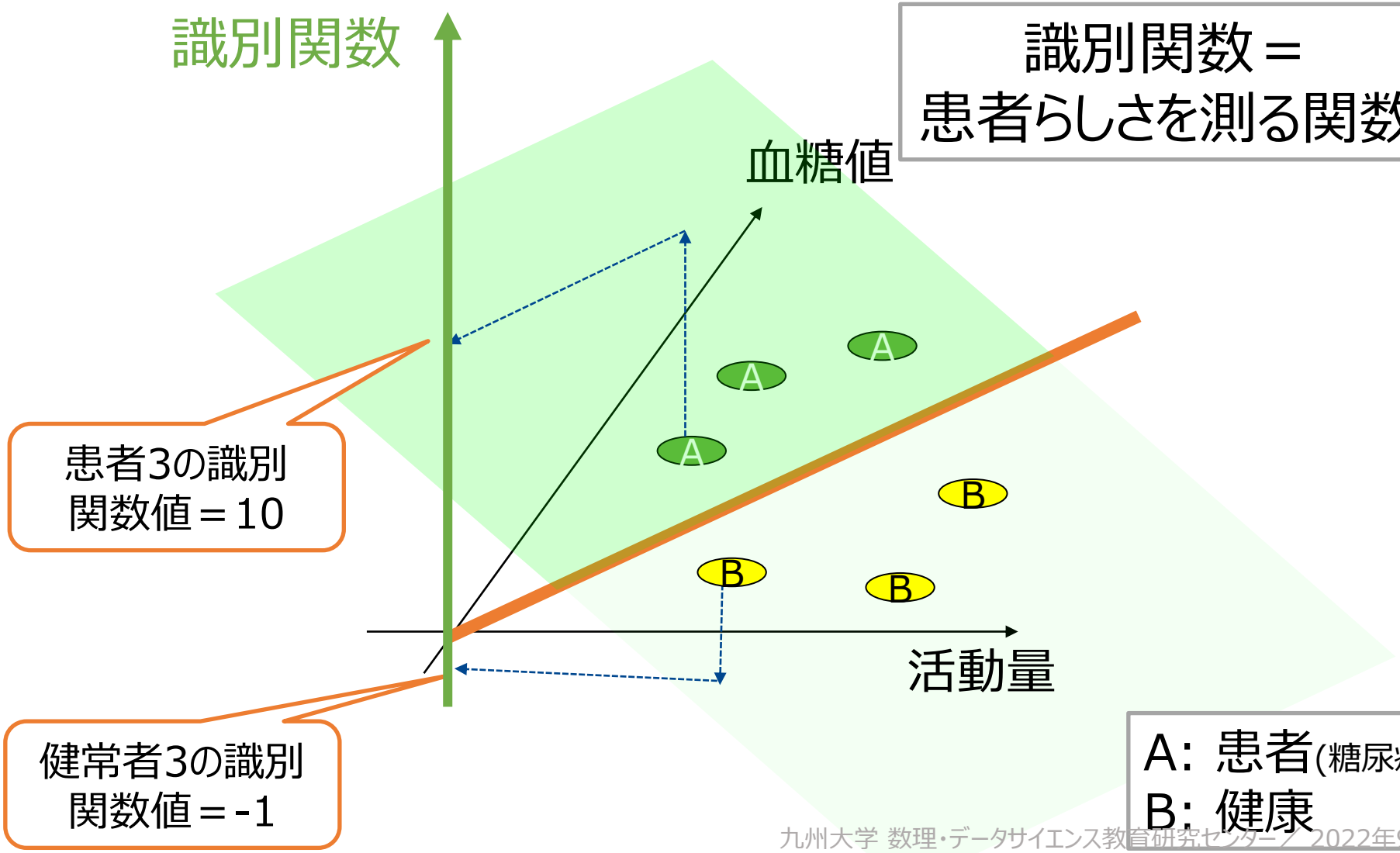
識別関数とは？
例：診断マシーンを作りたい！



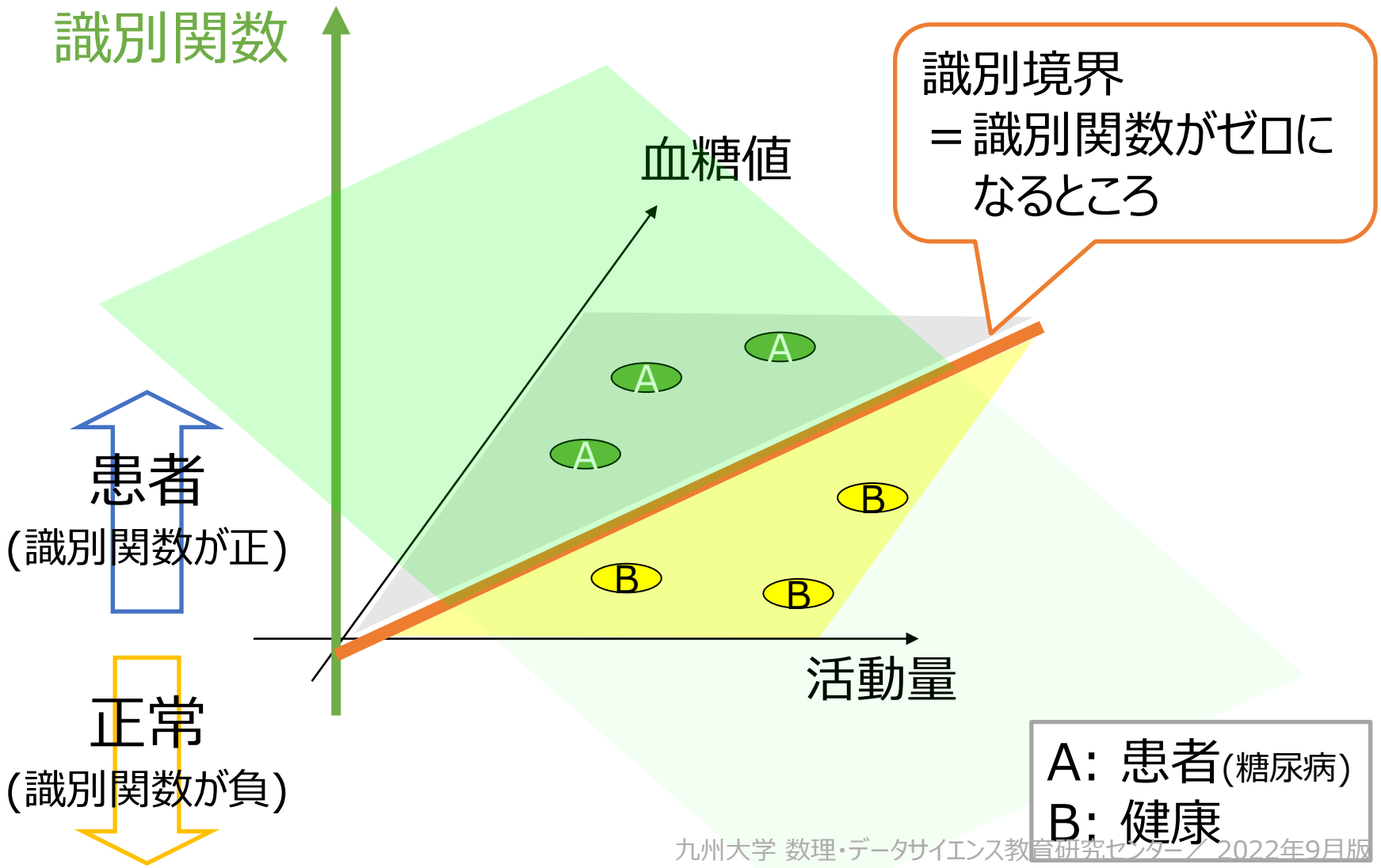
A: 患者(糖尿病)
B: 健康

識別関数とは？
例：診断マシーンを作りたい！

識別関数＝
患者らしさを測る関数

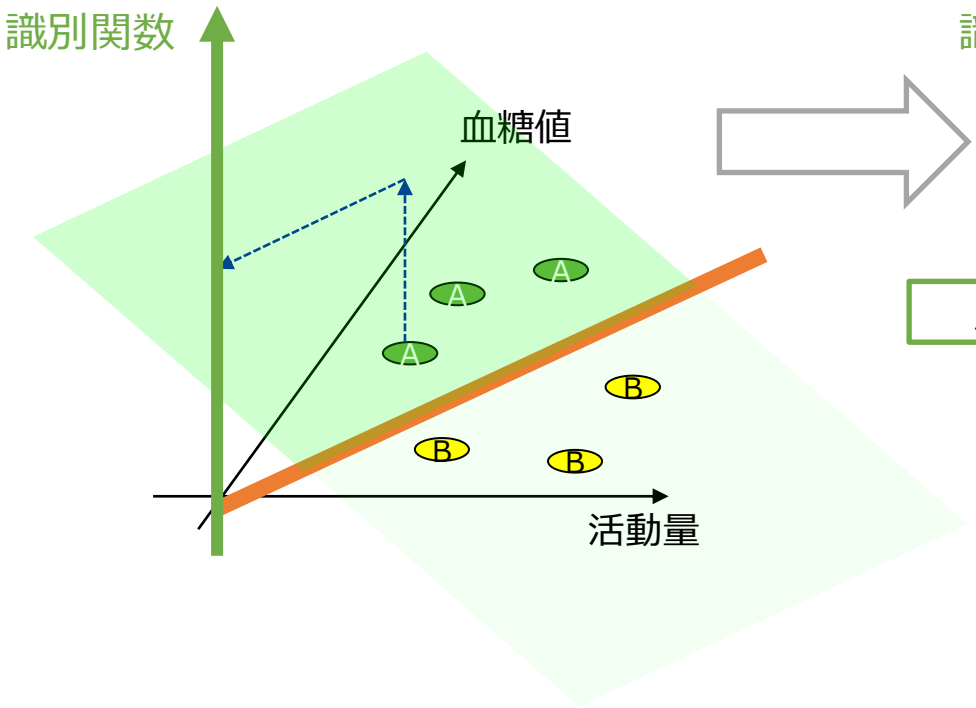


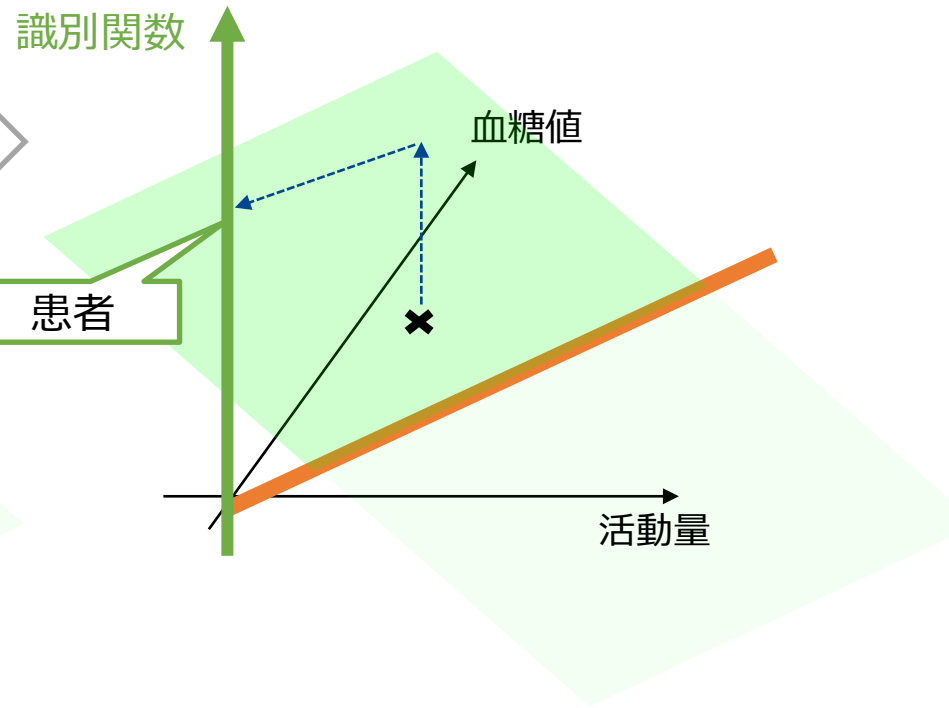
識別関数とは？
例：診断マシーンを作りたい！



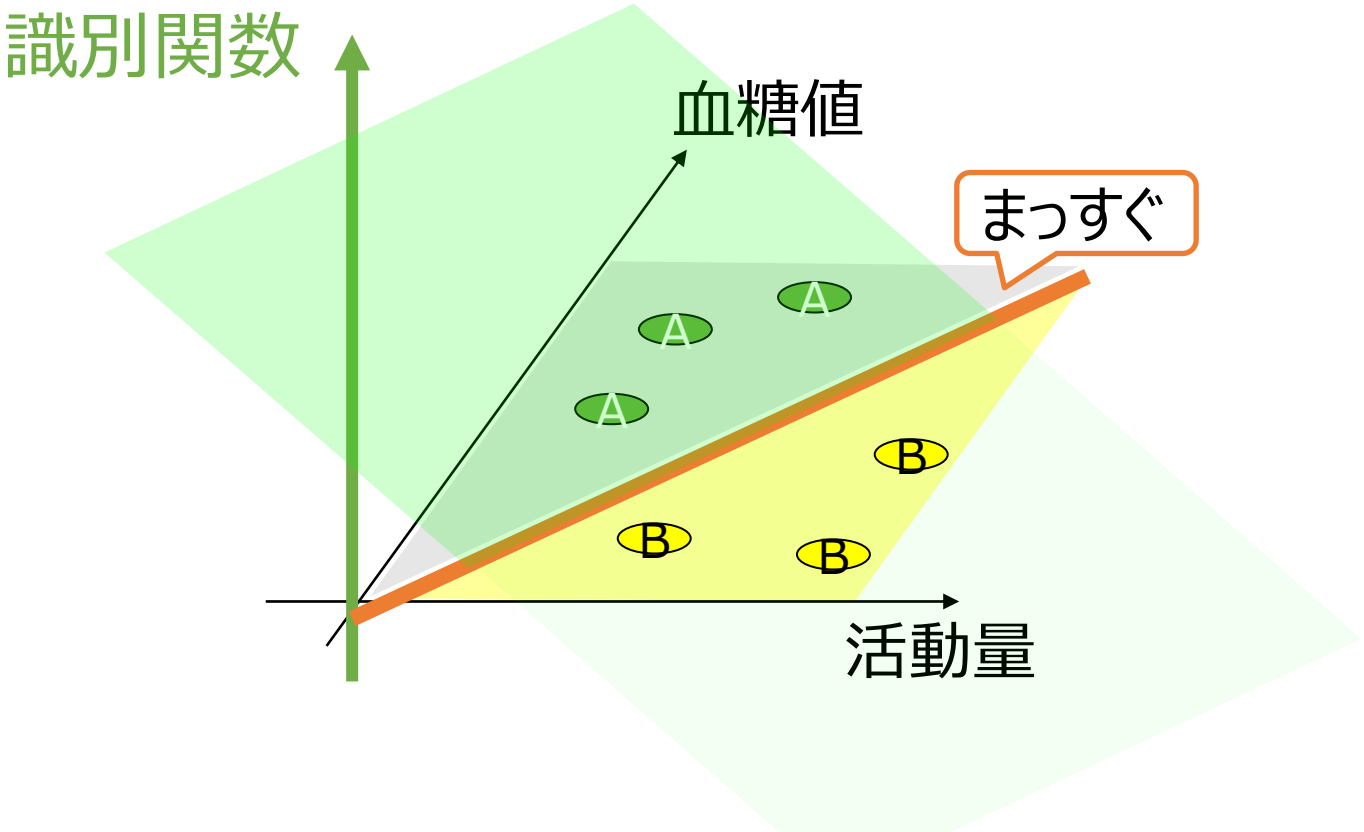
識別関数によるデータ分類の手順

- 【学習】クラスがわかったデータを用いて識別関数を決定する
- 【利用】未知データを識別関数に入力し，関数値を求めることで，分類を行う（以下の例では，関数値が正ならば患者）





こんなまっすぐな境界では不十分では？



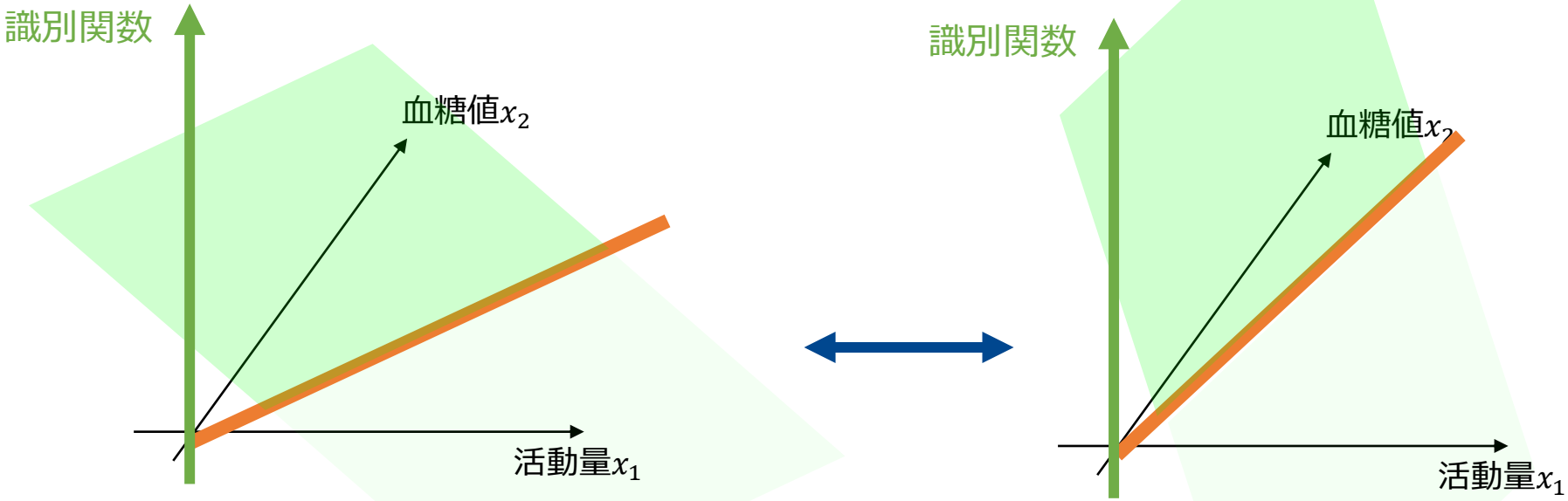
- 上のような平面的な関数ではなく，よりぐにゃぐにゃした関数を利用する場合も多い！ →【付録】深層ニューラルネットワークによるパターン認識：その基本的考え方

なぜ識別関数？ 最近傍法でいいのでは？

- 境界の形を，好きなようにコントロールできる
  - まっすぐ（線形）から，ぐにゃぐにゃまで
- はずれデータの影響を小さくできる
- 最適性など，理論的な性能保証がある
- パターン認識の問題を「識別関数のパラメータを決める問題」と捉えることができる！
  - パラメータ＝「調節つまみ」

Emoji one@wikimedia

パラメータを決める問題?

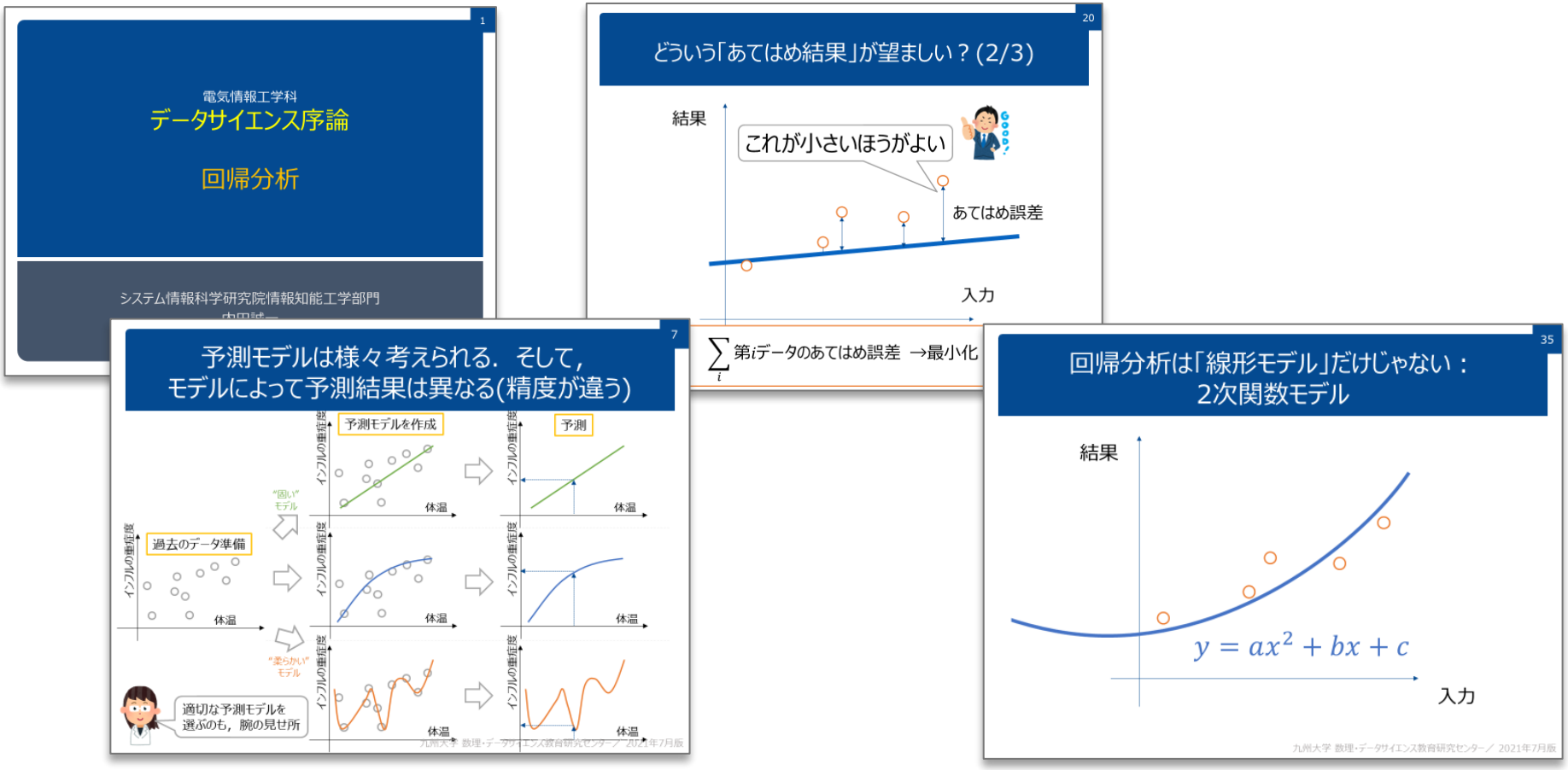


「識別関数の傾き方」つまみ

Emoji one@wikimedia

- 集めたデータがなるべく正しく分類できるように，つまみを調節する！
  - 上の絵では，識別関数は $y = ax_1 + bx_2 + c$ なので，つまみはa, b, cの3つ

回帰分析の話を思い出していただけると嬉しい
（パターン認識は回帰分析とも関係しているのです）

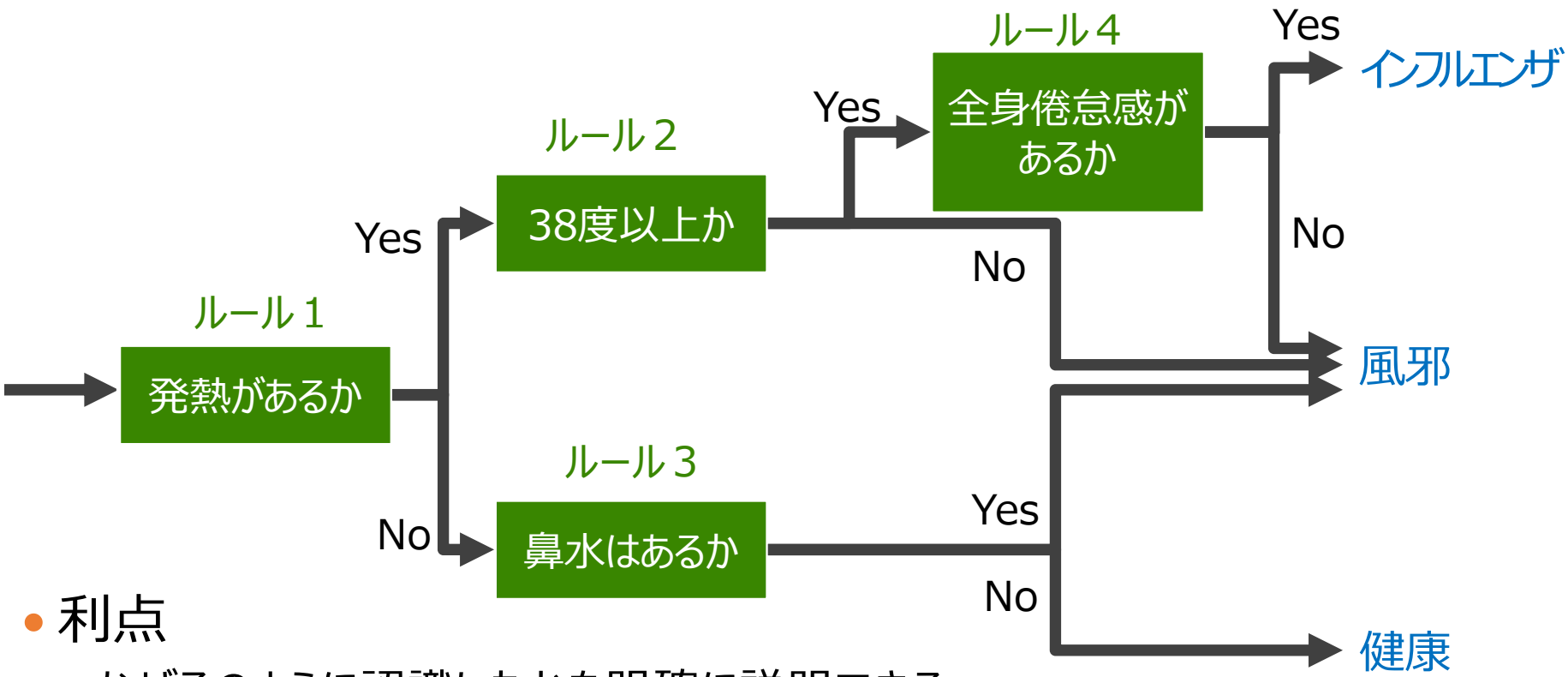


- この話の続きは「15-人工知能入門」の「機械学習」でも紹介

ルールベースのパターン認識

分類木とも呼ばれます

ルールベースのパターン認識

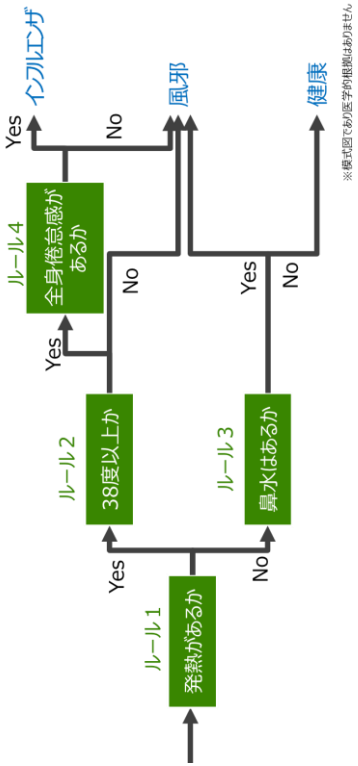


※模式図であり医学的根拠はありません

- 利点
  - なぜそのように認識したかを明確に説明できる
- 欠点
  - どのようなルールを設定すればよいかが明らかではない
  - 手動で設定したルールでは性能が出ない場合もある

先ほどのルールは「分類木」とも呼ばれます

- 90度回転させると，「木」に見える？



木の葉

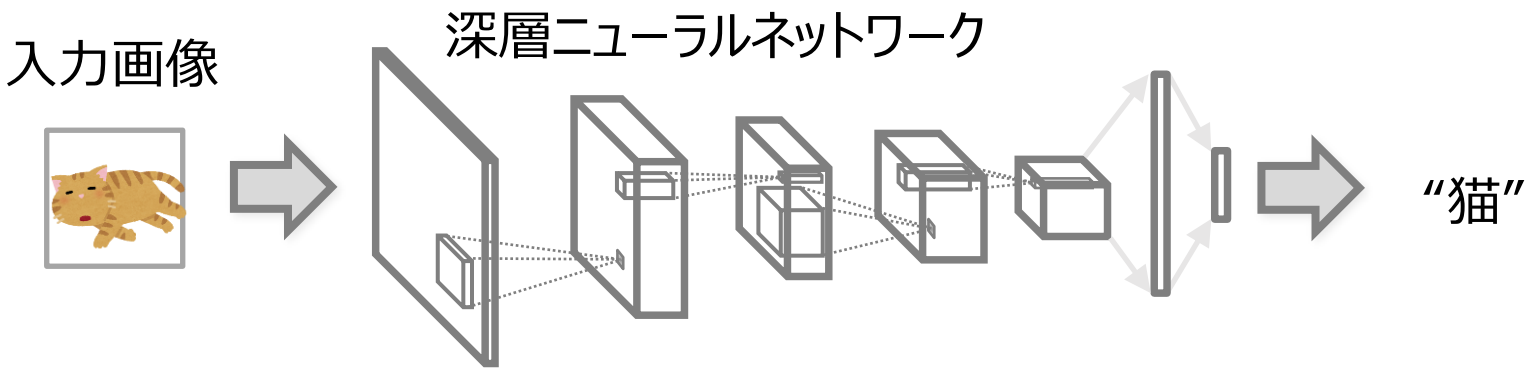
木の根

- 「木」の構造は，情報学等，様々なところで活用されています！
  - グラフ理論，ネットワーク理論でも活用される，データ構造の基本なのです

【付録1】
深層ニューラルネットワークによる
パターン認識：
その基本的考え方

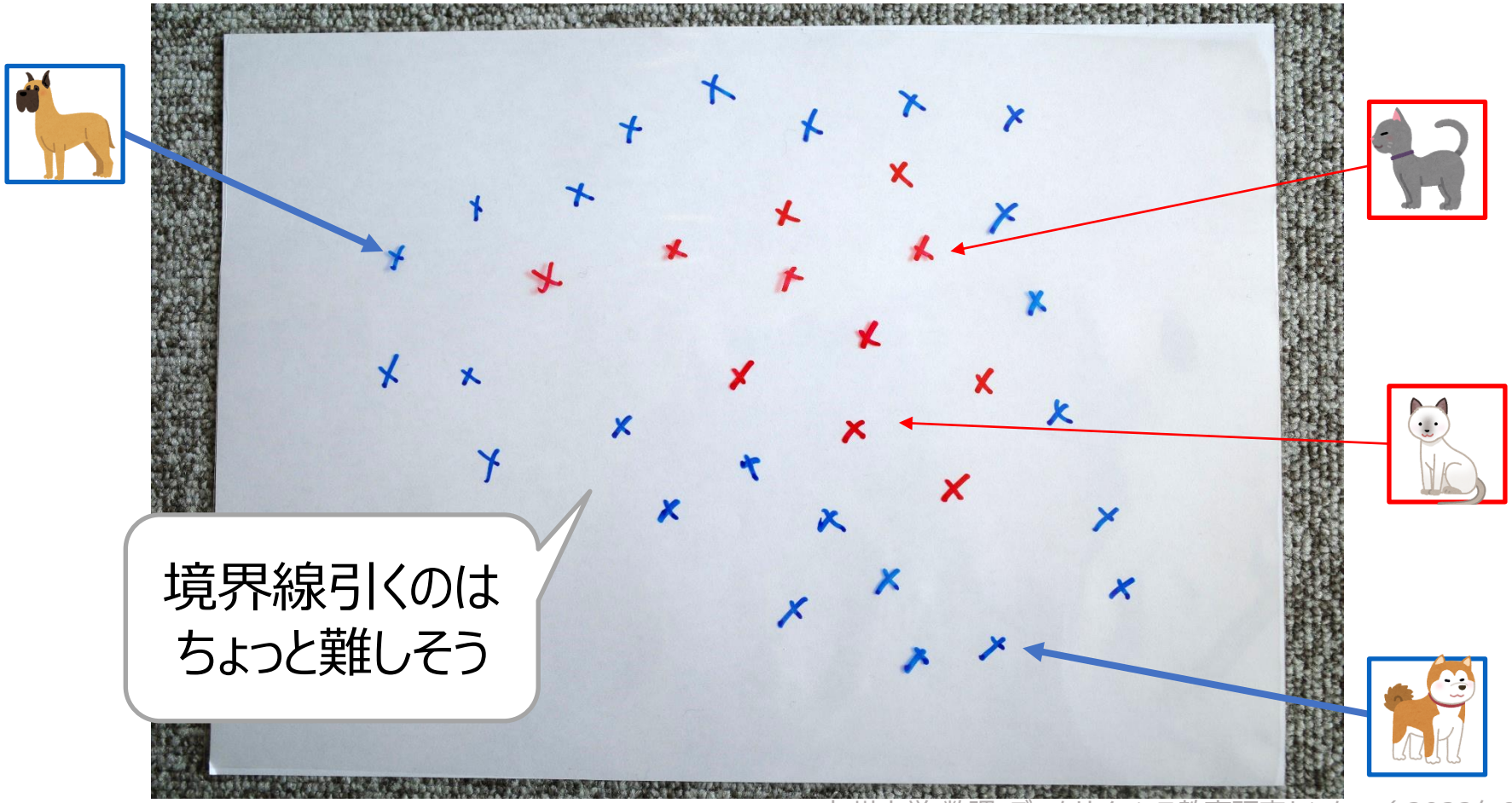
最近のAIブームの立役者

最近のパターン認識には「深層ニューラルネットワーク」が利用される

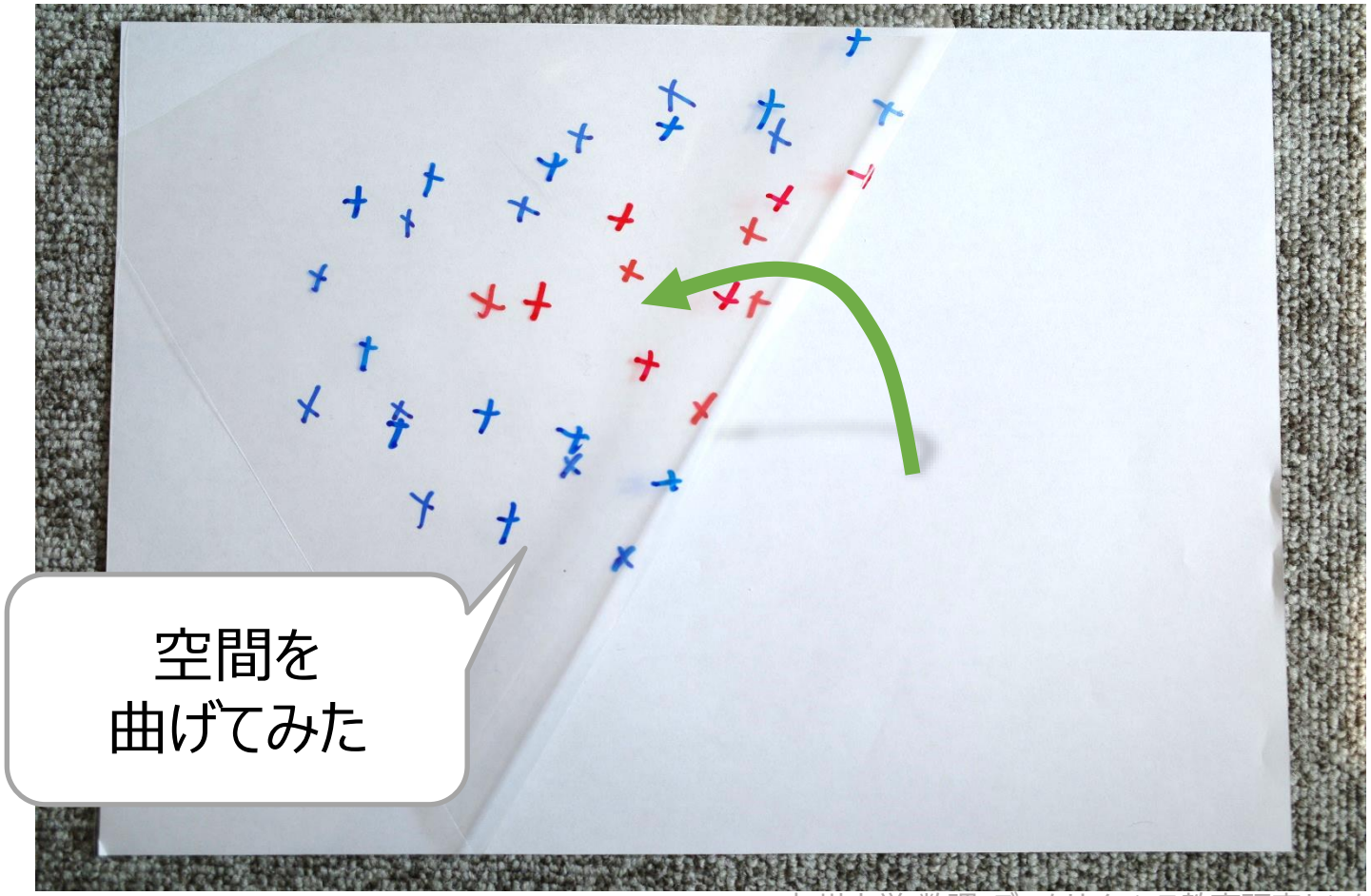


- この深層ニューラルネットワークがどのように「境界線」を引いているか，イメージ図で見てみよう

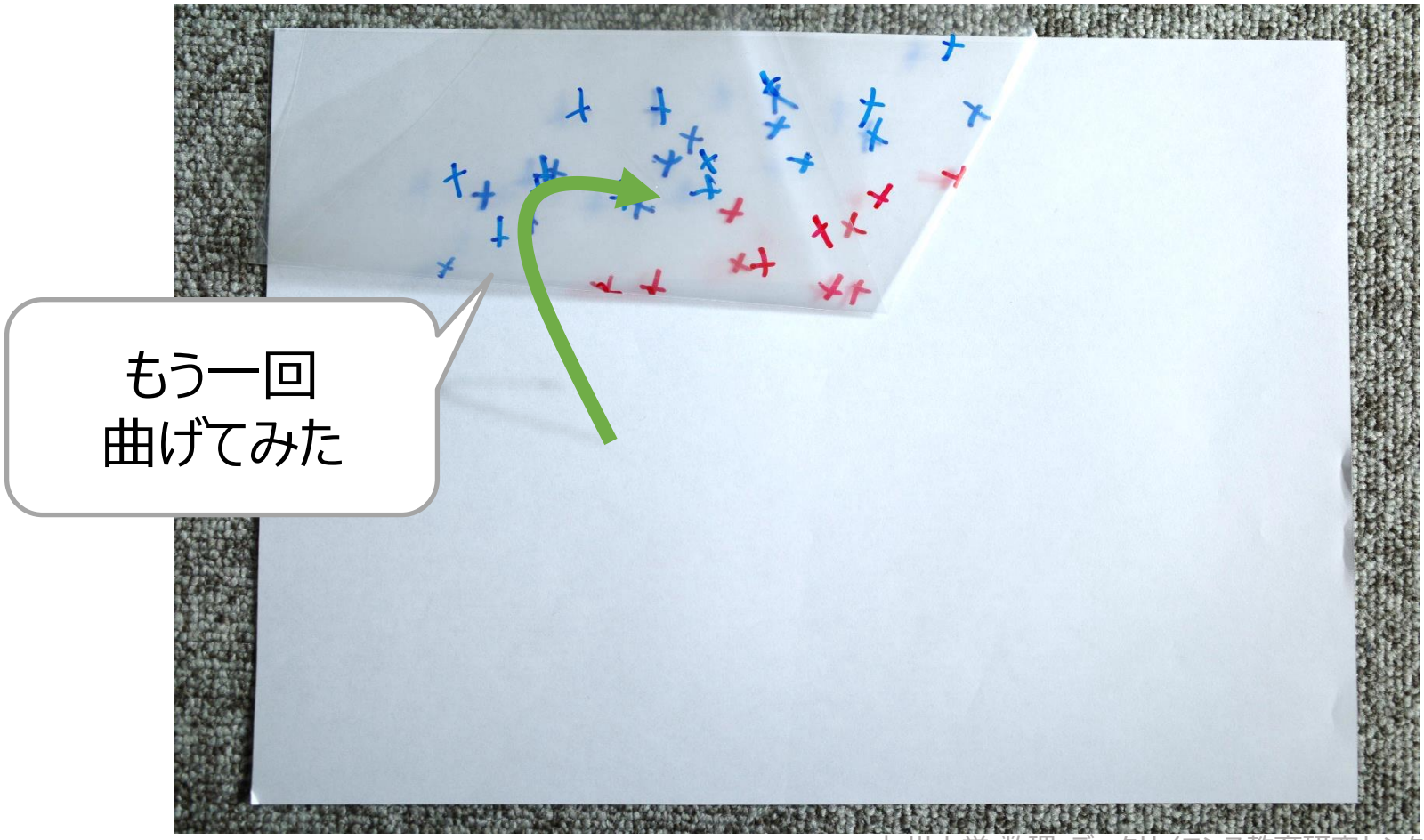
深層ニューラルネットワークによる パターン認識のイメージ図 (1/6)



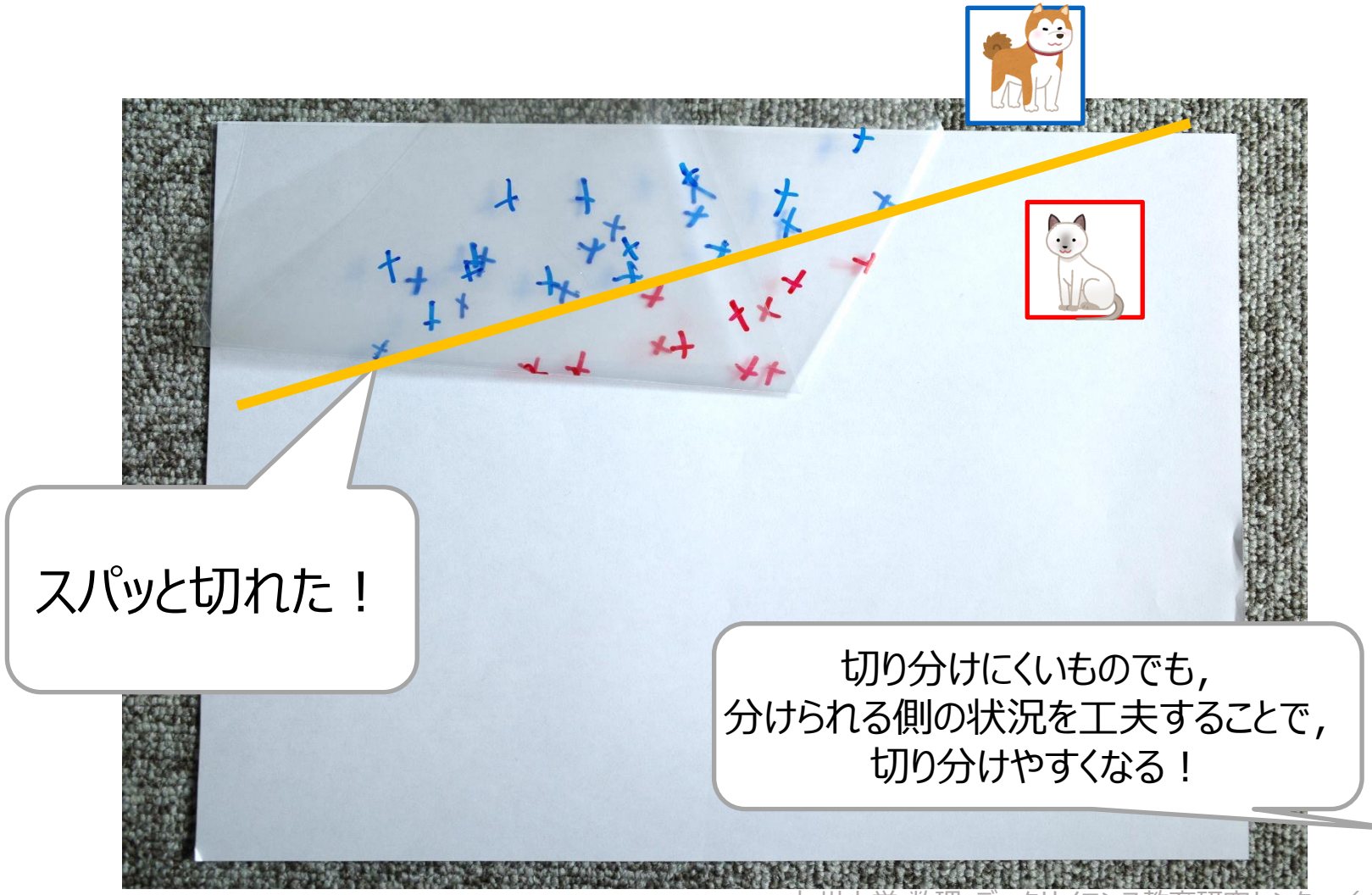
深層ニューラルネットワークによる パターン認識のイメージ図 (2/6)



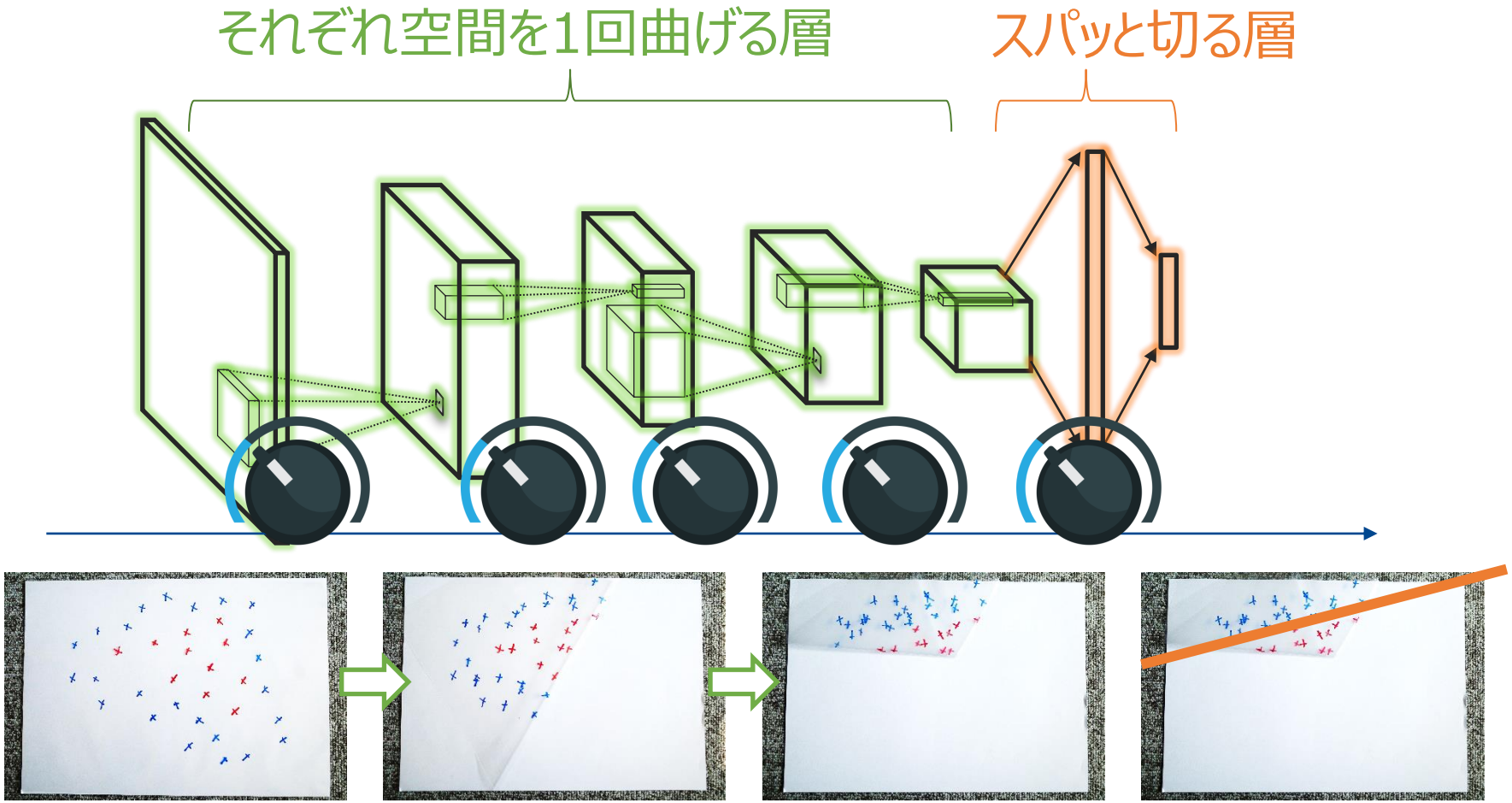
深層ニューラルネットワークによる パターン認識のイメージ図 (3/6)



深層ニューラルネットワークによる パターン認識のイメージ図 (4/6)

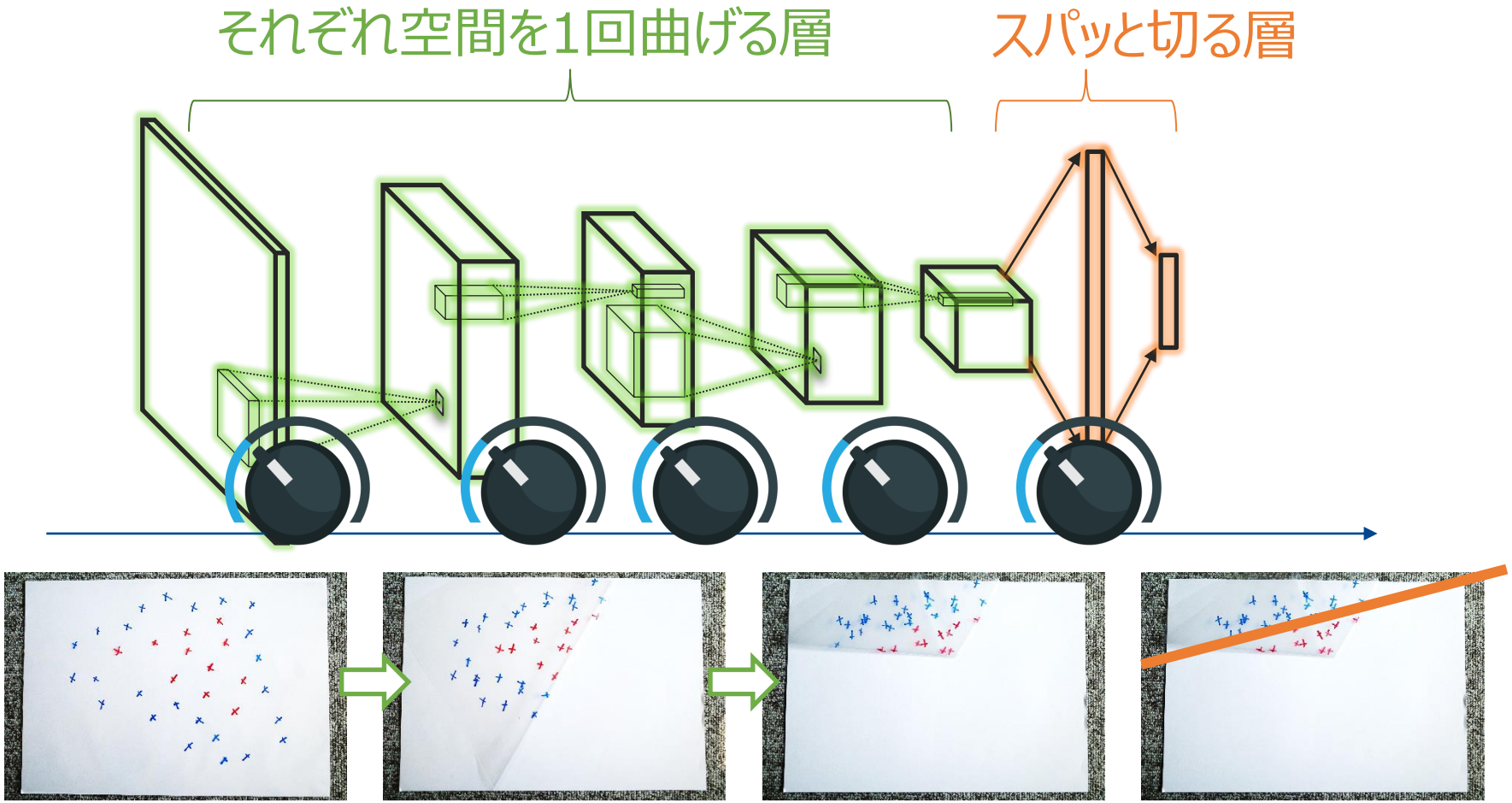


深層ニューラルネットワークによる パターン認識のイメージ図 (5/6)



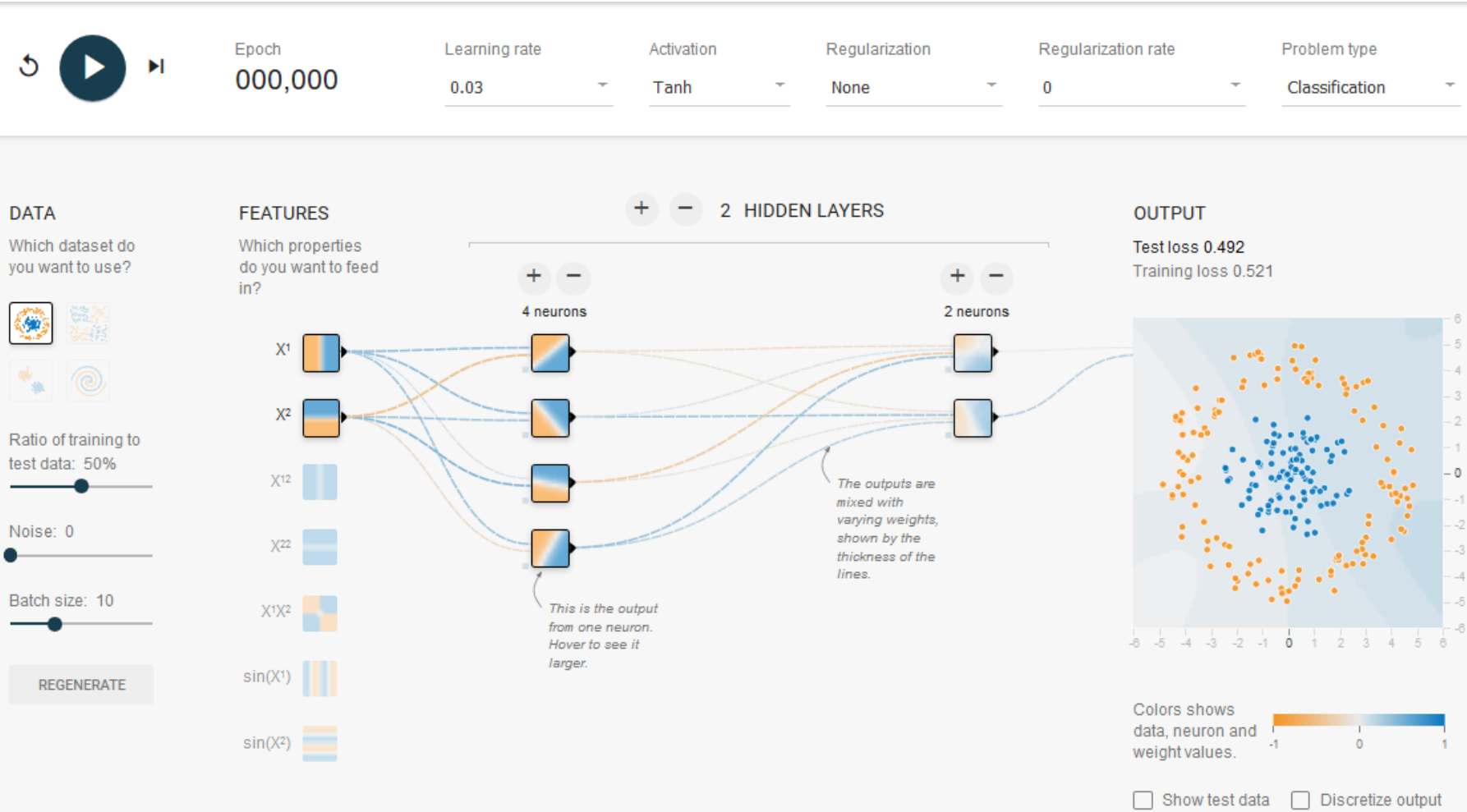
各層には膨大な数のパラメータ（つまみ）がある.
それらを調整して，空間の曲げ方や，識別関数（オレンジの線）を適切に設定する

深層ニューラルネットワークによる パターン認識のイメージ図 (6/6)



パラメータは微調整を何度も繰り返しながら決めていく（＝学習していく）！
具体的には準備されたデータの誤認識が少しずつでも減るように修正

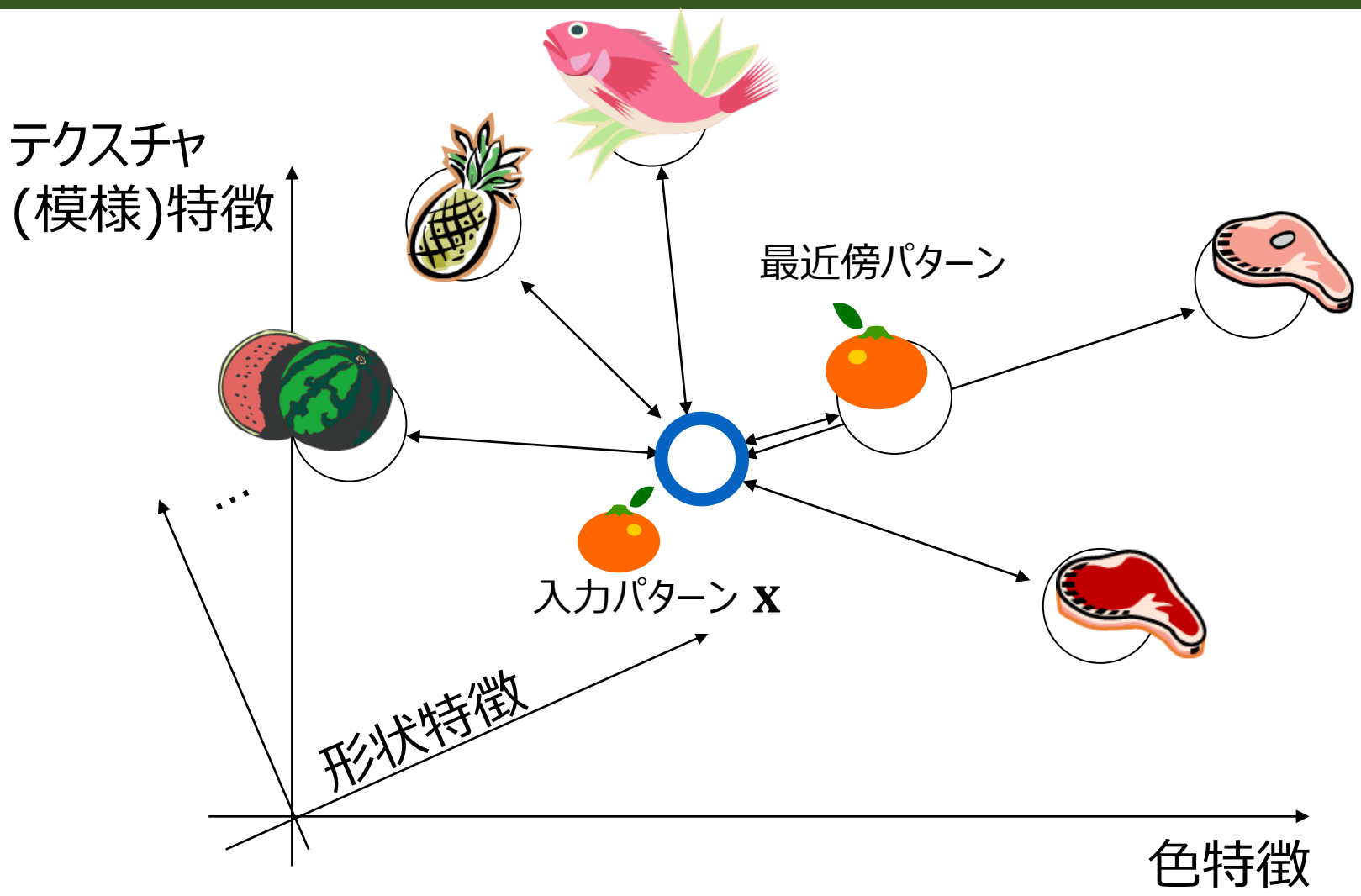
深層ニューラルネットワークの挙動を理解したい人におすすめ google playground https://playground.tensorflow.org/



【付録2】
深層ニューラルネットワークによる
パターン認識：
もう少ししっかり理解する！

これまでの知識で理解できるはずですが,
電気情報工学科3-4年生向け講義の資料の一部なので,
少々難しく感じるかもしれません

特徴空間と最近傍法(再掲)



ところで，特徴もイロイロある

パターン自身の性質

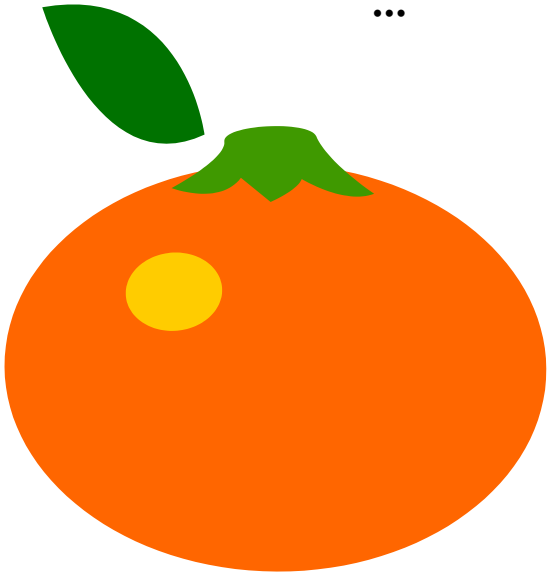
- オレンジの画素数→多い
- 青の画素数→小
- 円形度→高い
- 線対称度→高い
- 模様→細かい

....

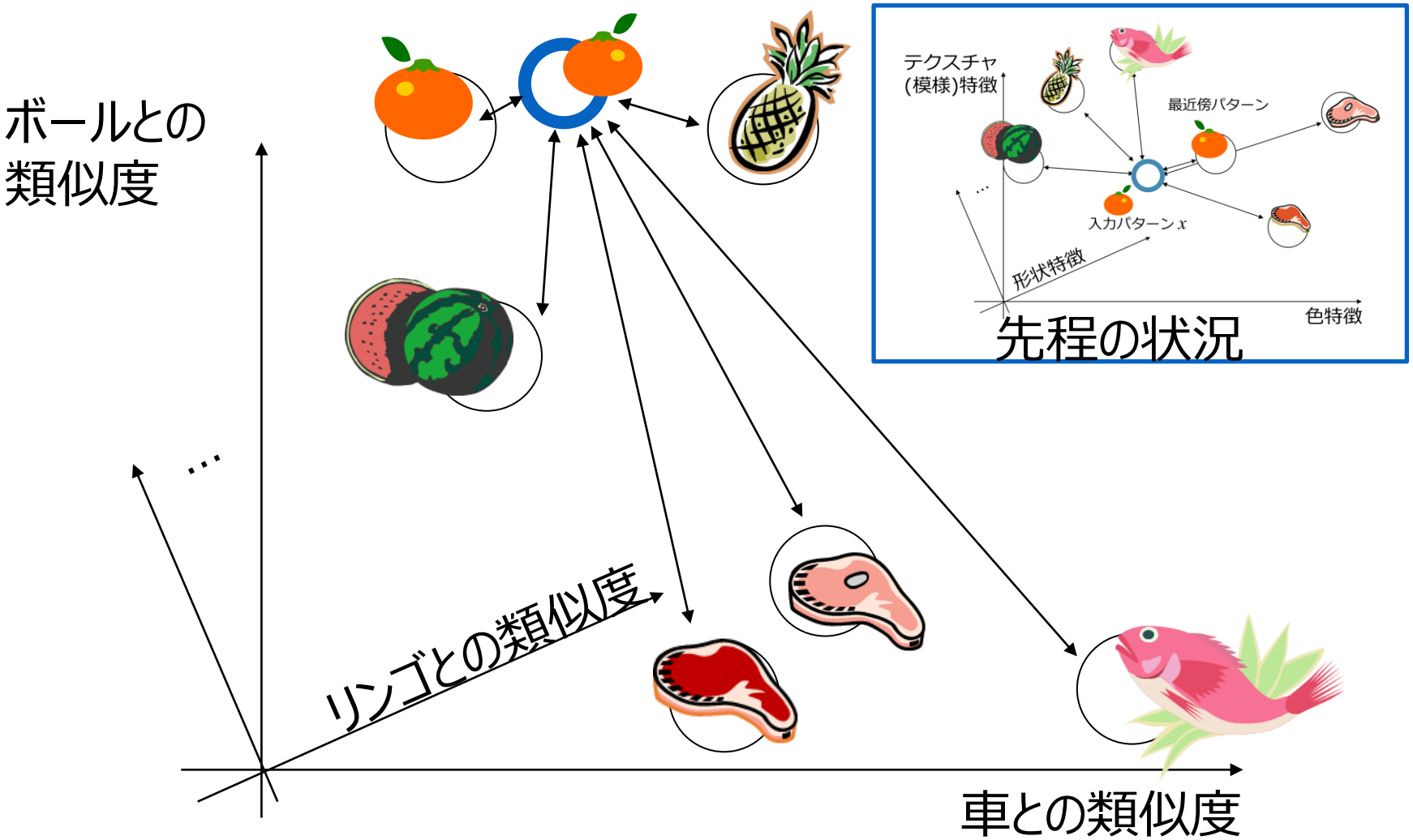
他のパターンとの関係

- 「車」との類似度→低
- 「リンゴ」との類似度→高
- 「猿」との類似度→低
- 「ボール」との類似度→高

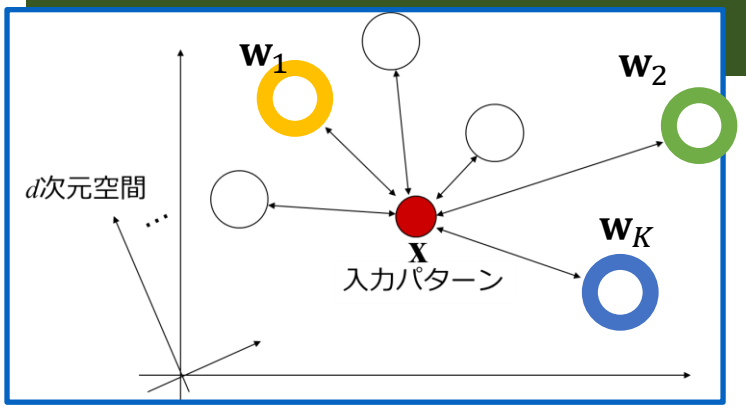
...



他のパターンとの関係（類似度）を特徴に



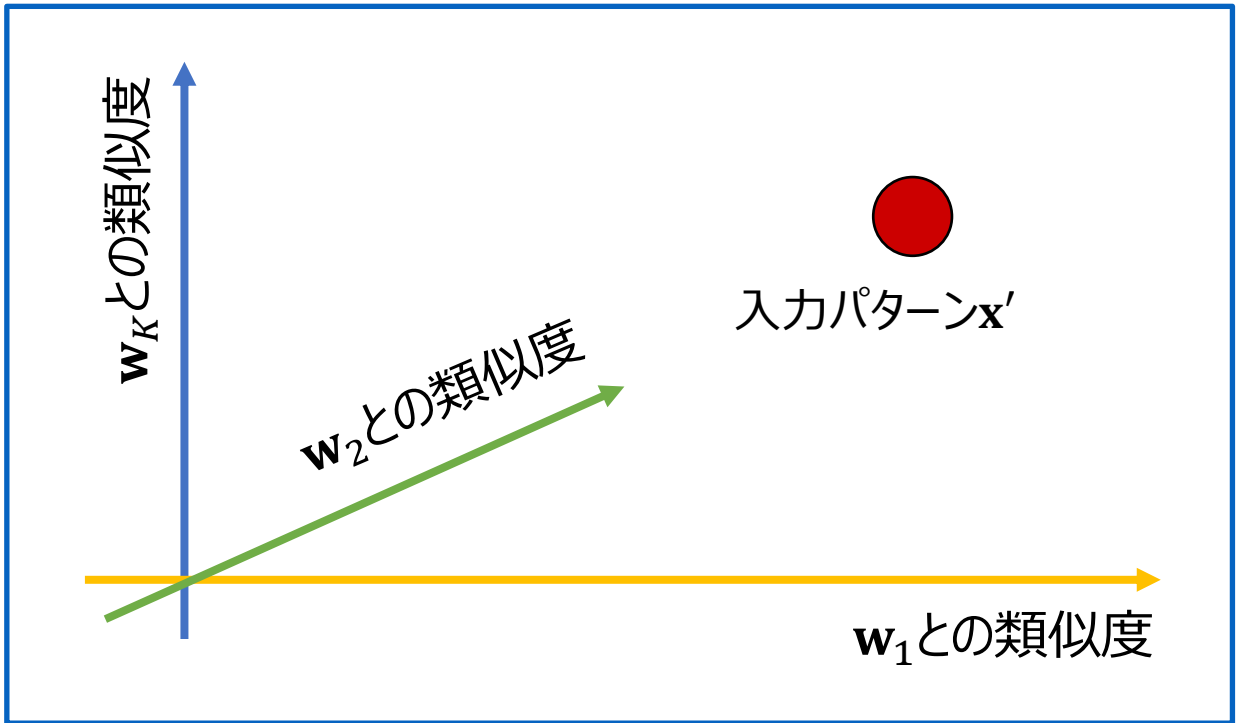
一種の「空間変換」となっている



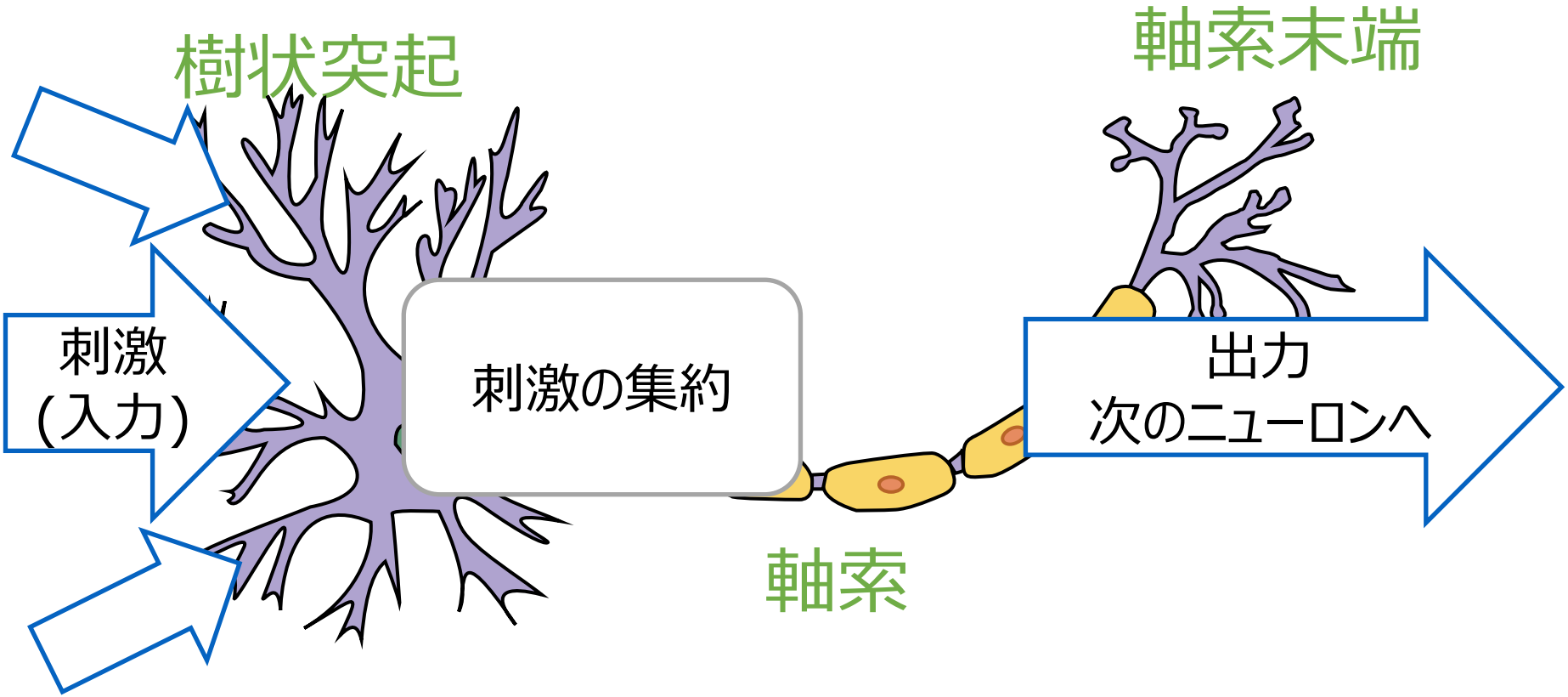
元のd次元空間

(Dis)similarity space embedding とも呼ばれる

新たなK次元空間



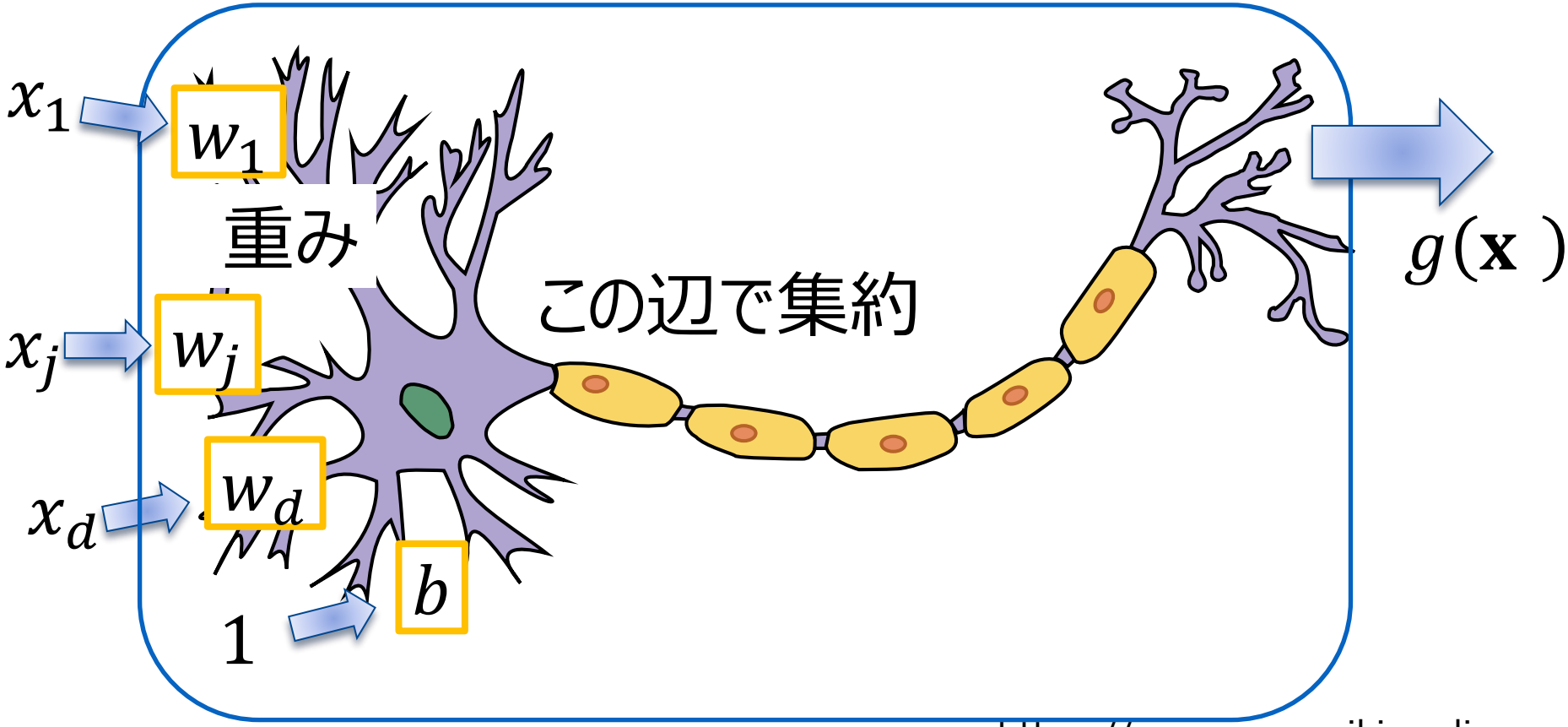
さて，神経細胞（ニューロン）



https://commons.wikimedia.org/

ニューロンの計算モデル

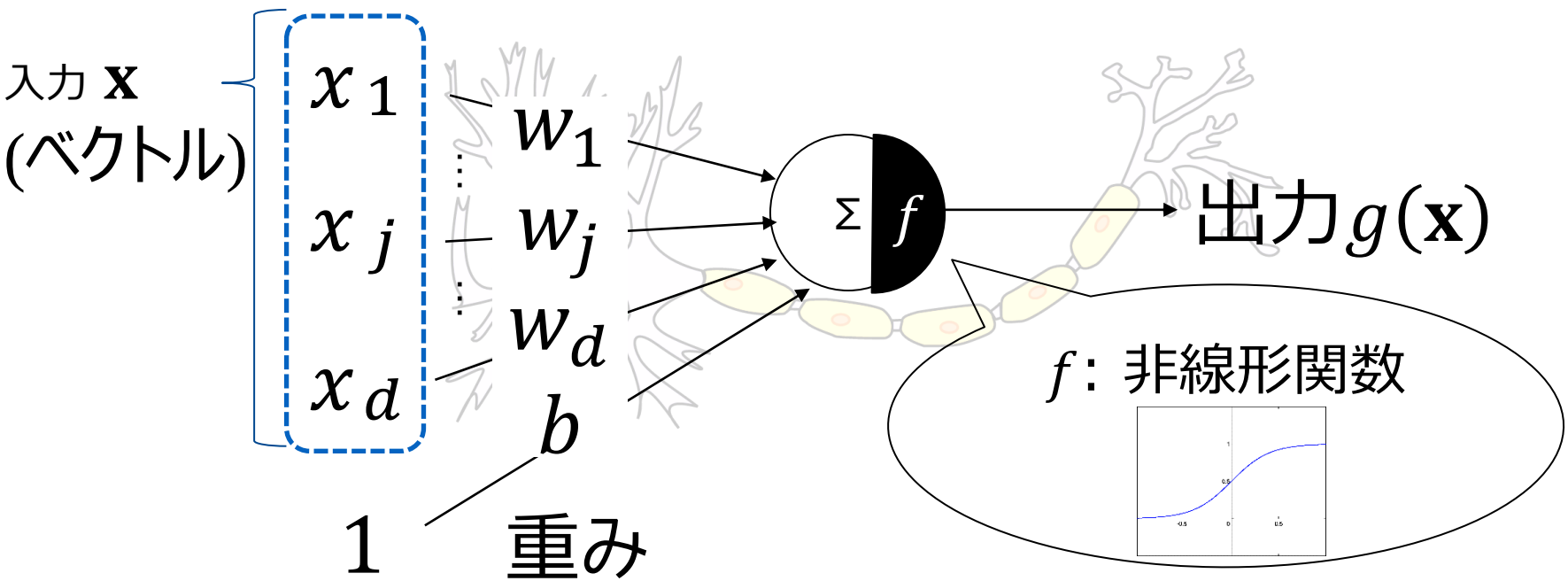
入力$\mathbf{x}$ → g → 出力$g(\mathbf{x})$



https://commons.wikimedia.org/

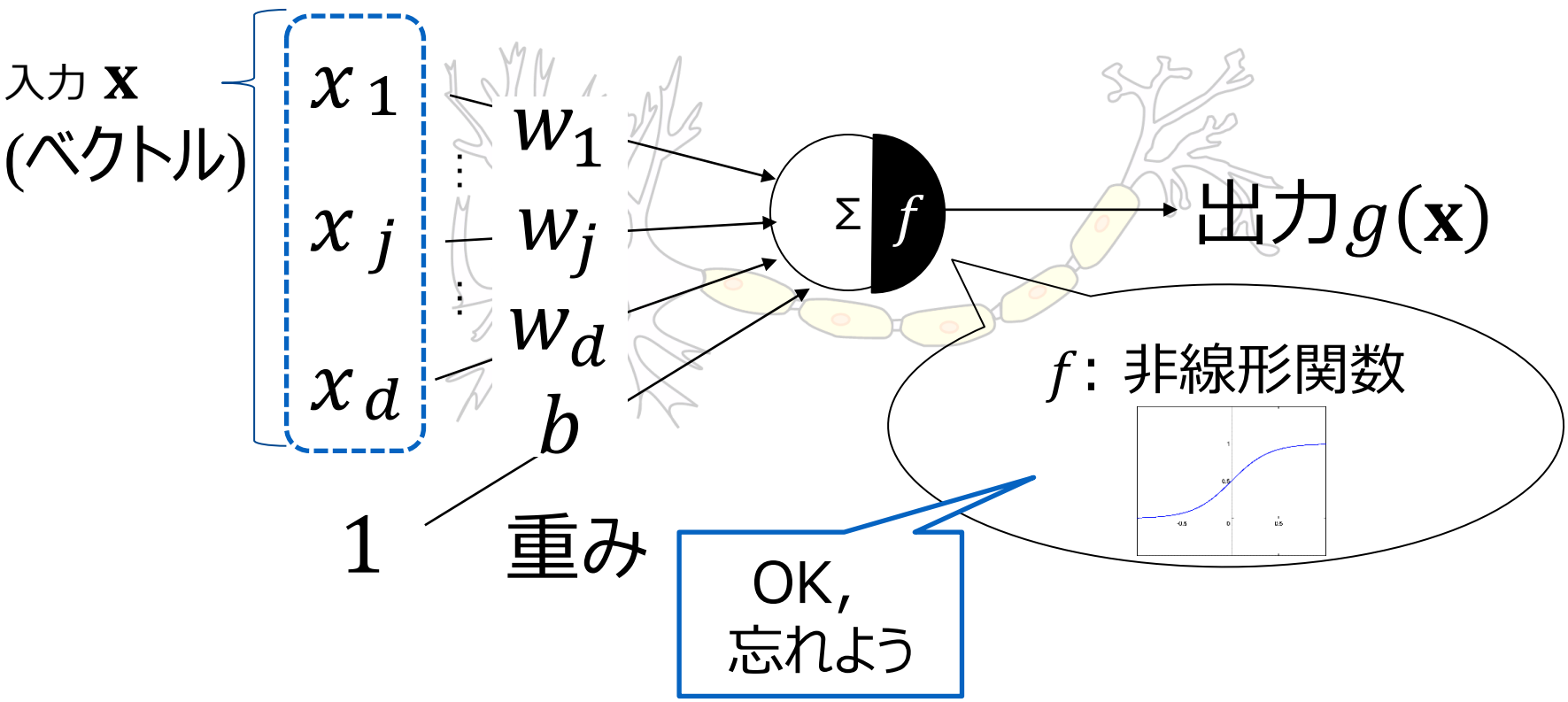
ニューロンの計算モデル

$$g(\mathbf{x}) = f\left(\sum_j w_j x_j + b\right)$$
$$= f(\mathbf{w}^T\mathbf{x} + b\ )$$



これが何なのか，単純化してみる

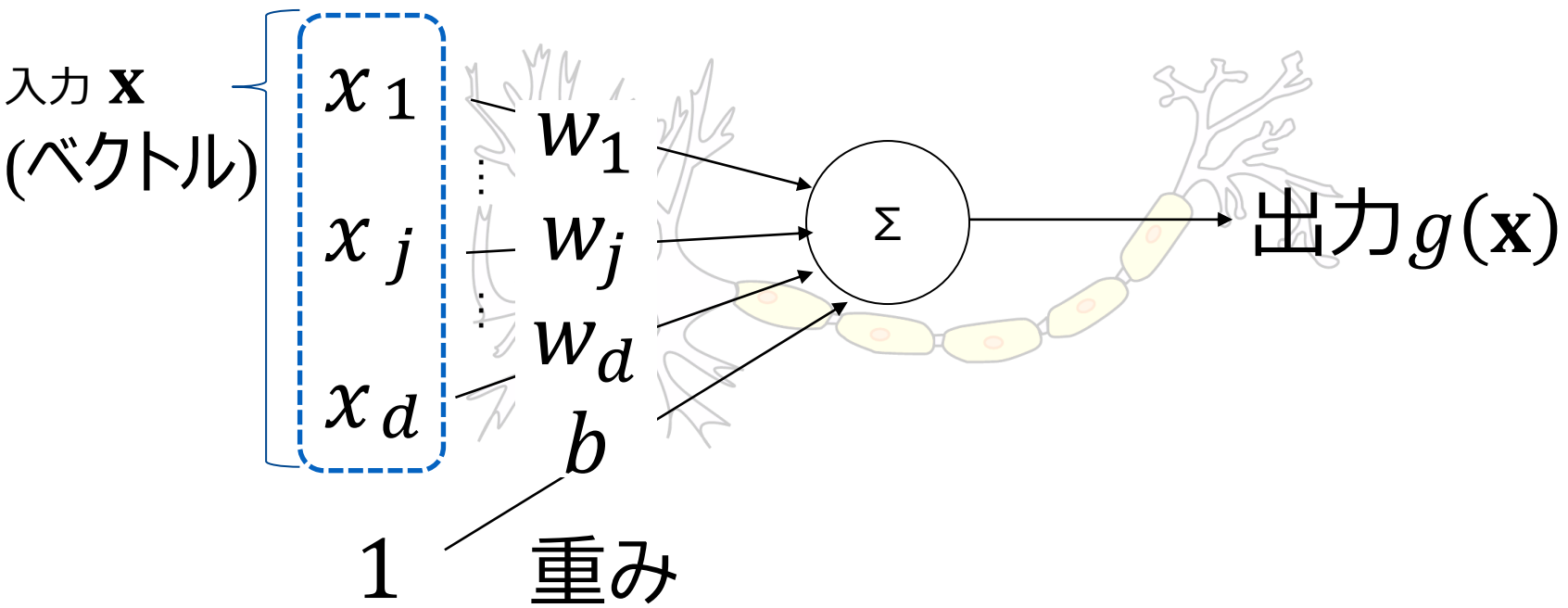
$$g(\mathbf{x}) = f\left(\sum_j w_j x_j + b\right)$$
$$= f(\mathbf{w}^T\mathbf{x} + b\ )$$



ニューロン≒線形識別関数だ！

線形
識別関数

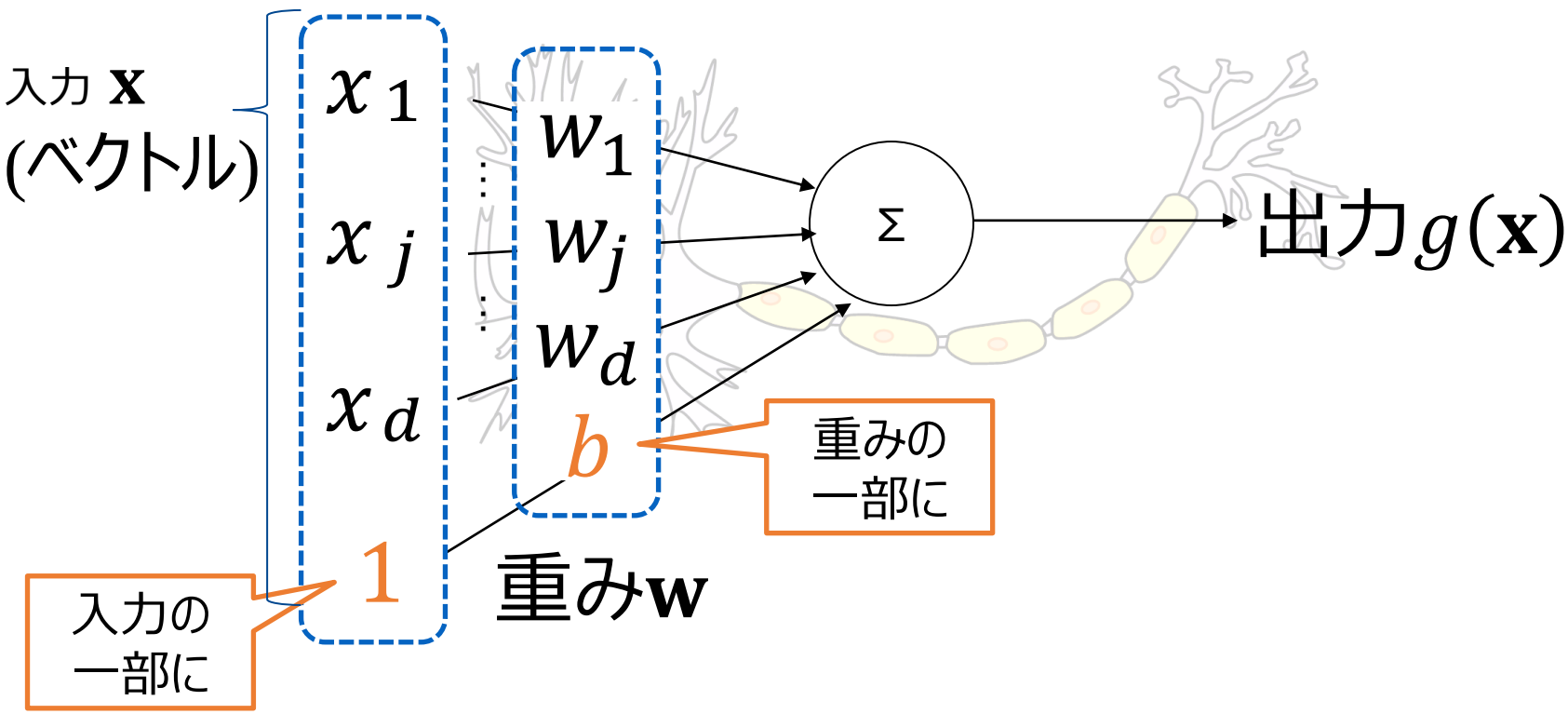
$$g(\mathbf{x}) = \sum_j w_j x_j + b$$
$$= \mathbf{w}^T \mathbf{x} + b$$



さらに単純化してみる

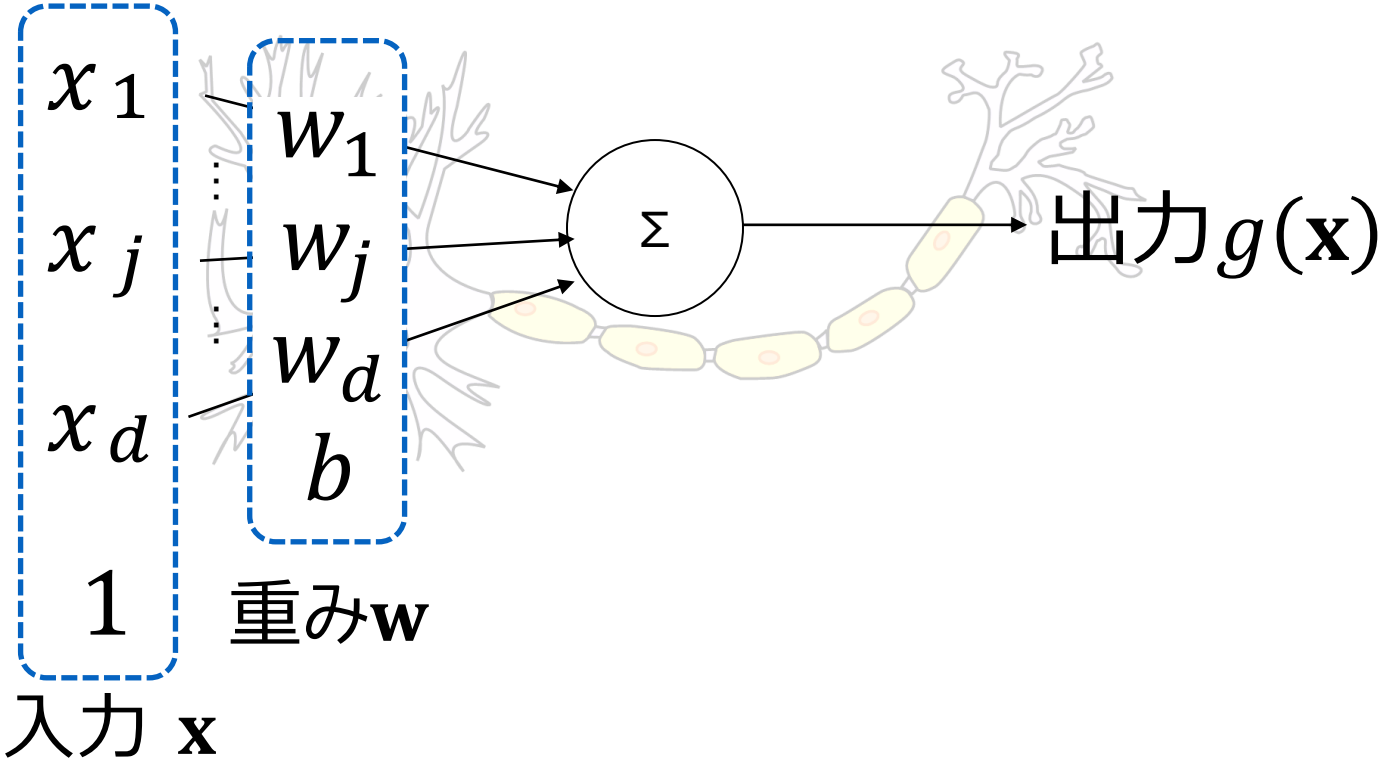
$$g(\mathbf{x}) = \sum_{j=1}^{d} w_j x_j + b$$
$$= \mathbf{w}^T\mathbf{x} + b$$

$$g(\mathbf{x}) = \sum_{j=1}^{d+1} w_j x_j$$
$$= \mathbf{w}^T\mathbf{x}$$



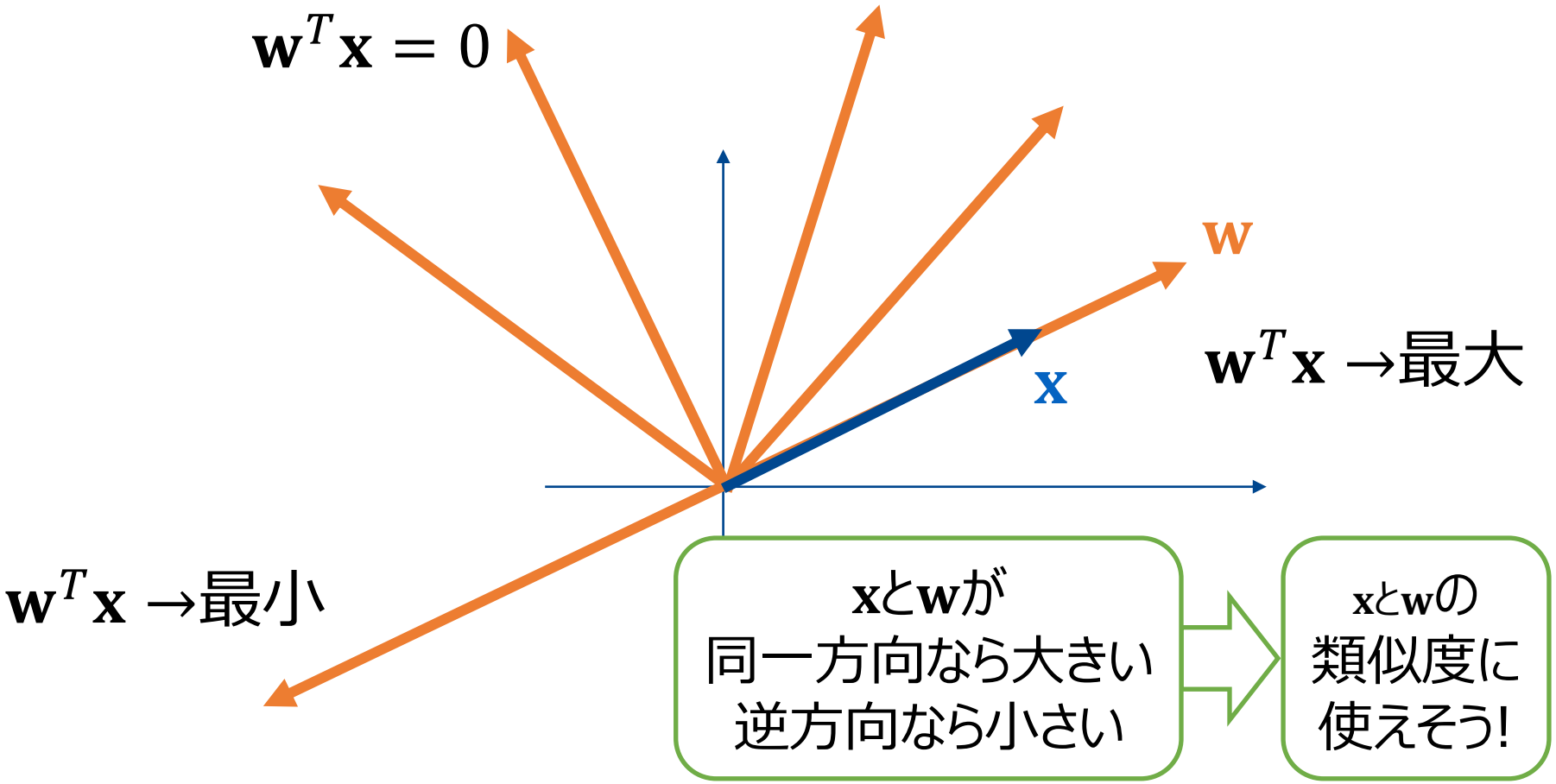
ニューロン≒ベクトルの内積だった

$$g(\mathbf{x}) = \sum_j w_j x_j$$
$$= \mathbf{w}^T \mathbf{x}$$



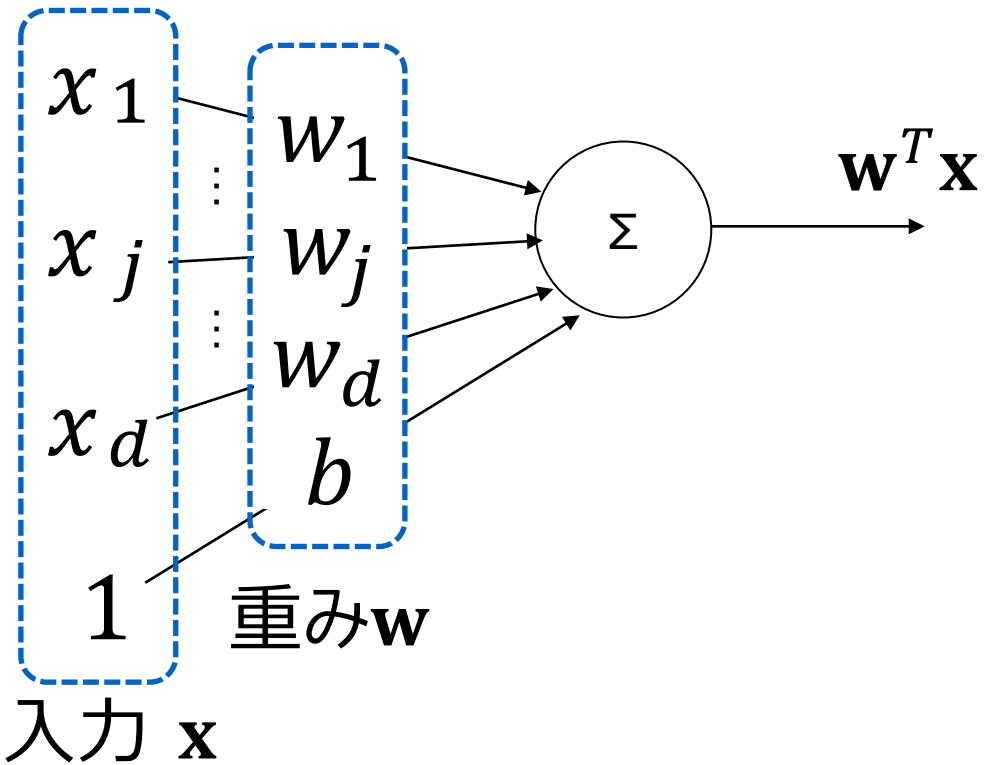
内積ってなんだ？～再び

- 一定の大きさのベクトル$\mathbf{w}$を回転させながら$\mathbf{x}$と内積を取ると…



ニューロンは類似度を測っている！

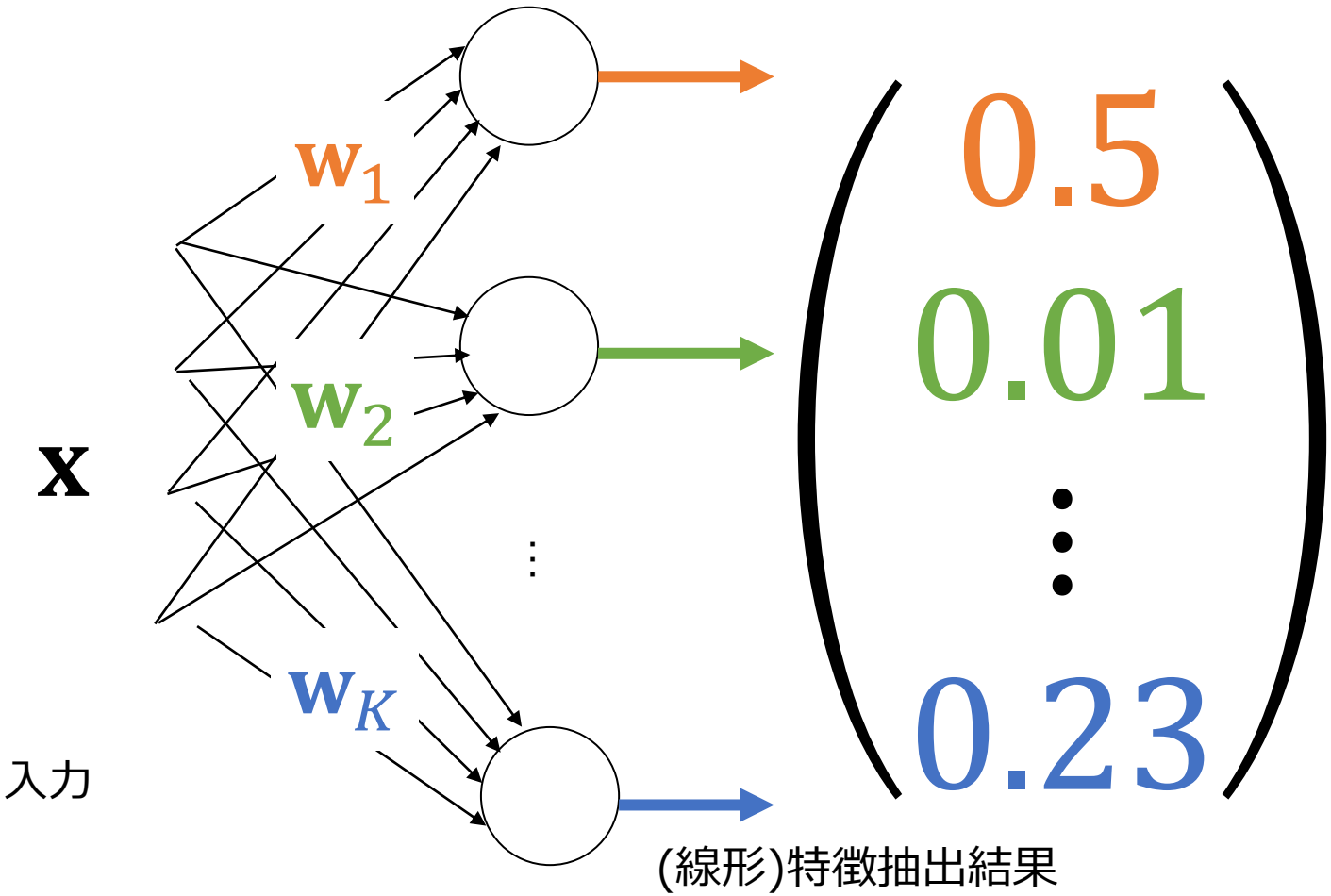
- ニューロン自身が持っている重み$\mathbf{w}$と似たベクトル$\mathbf{x}$が入力されると，大きく反応



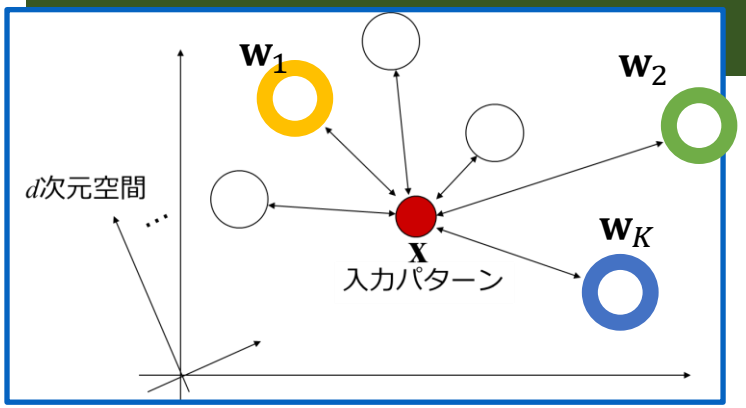
- $\mathbf{x}$と$\mathbf{w}$の類似度特徴を計算しているとも見える

ならば，K次元の「他パターンとの関係(類似度)」特徴を得るには

- ニューロンをK個並べればよい



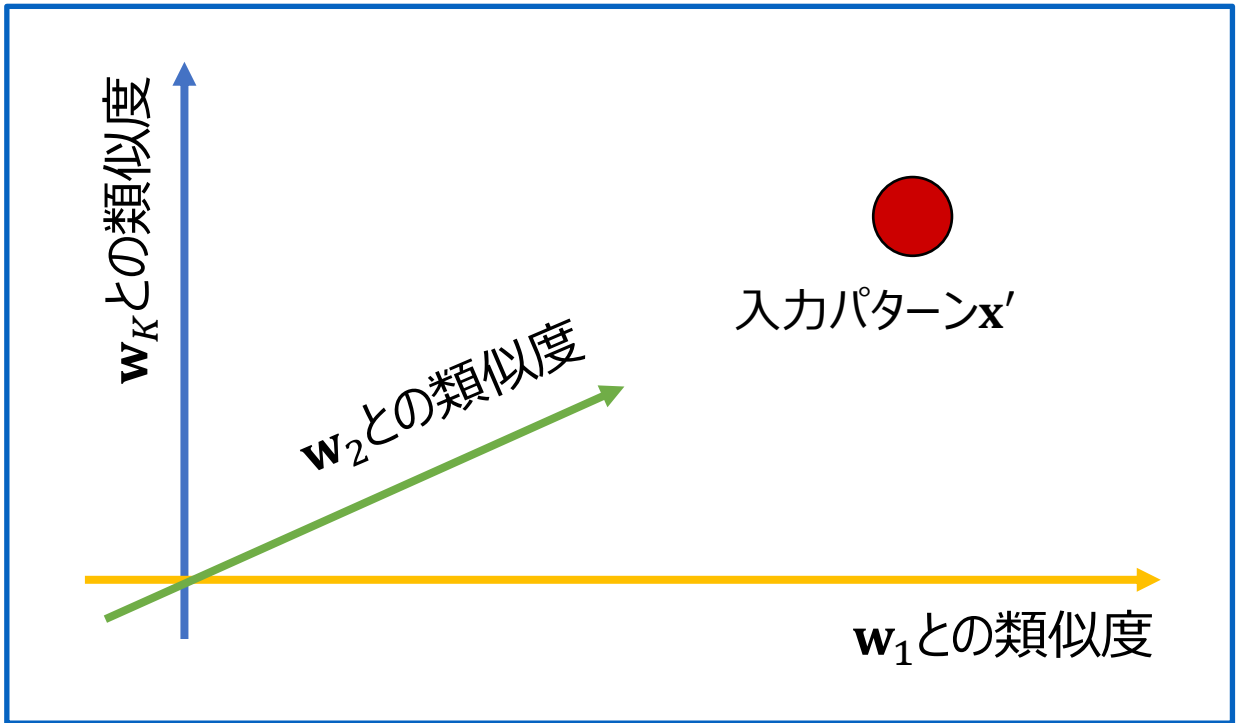
しつこいですが，「空間変換」となっている



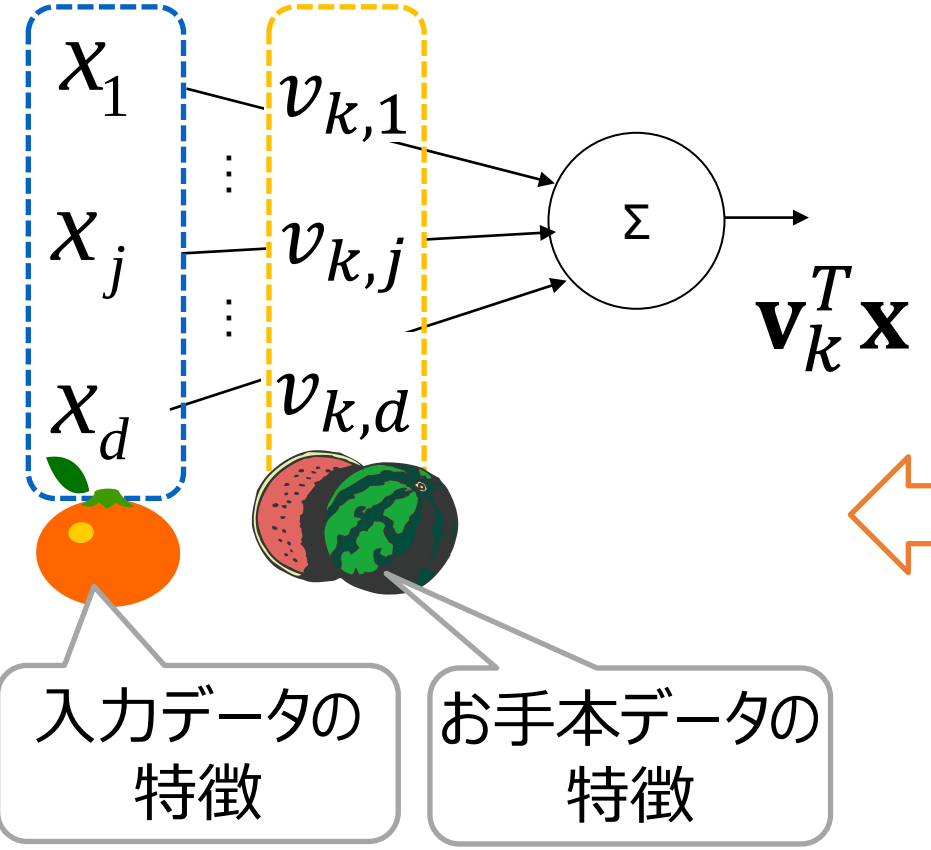
元のd次元空間

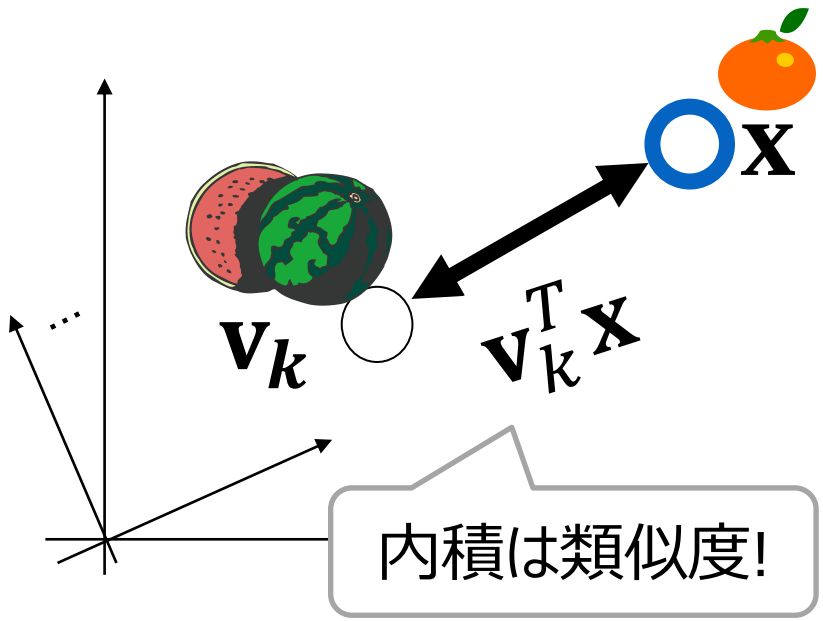
(Dis)similarity space embedding とも呼ばれる

新たなK次元空間

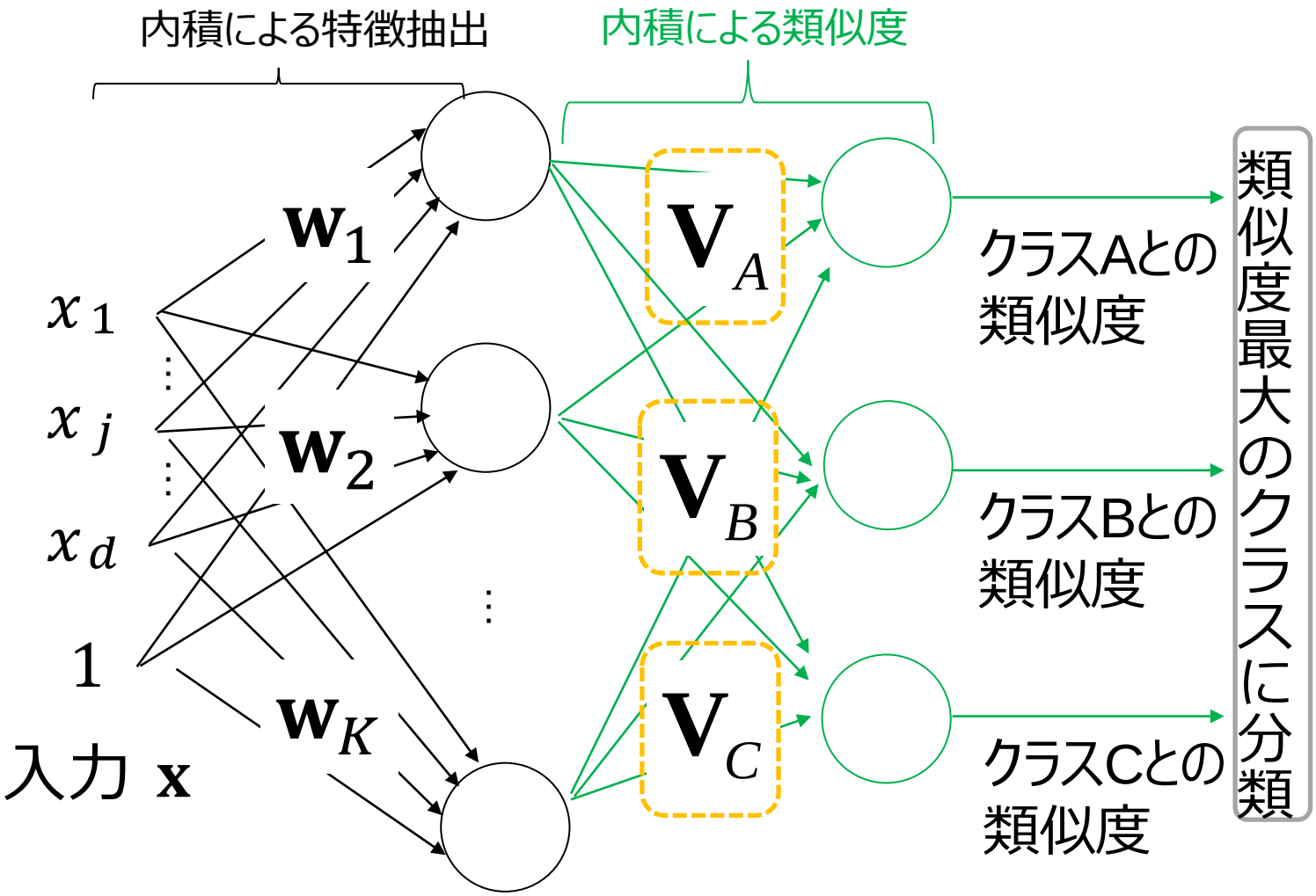


類似度ができる計算できるなら，最近傍法もできる





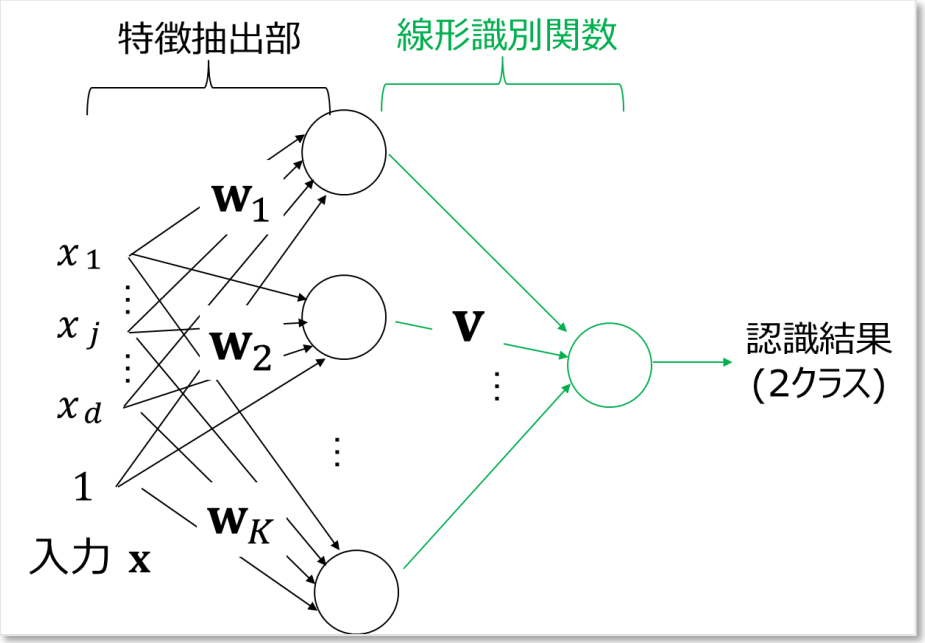
ニューロンを重ねるとニューラルネット完成



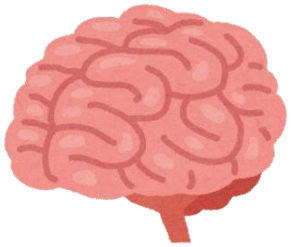
これが元祖ニューラルネット「パーセプトロン」

オモシロイところ

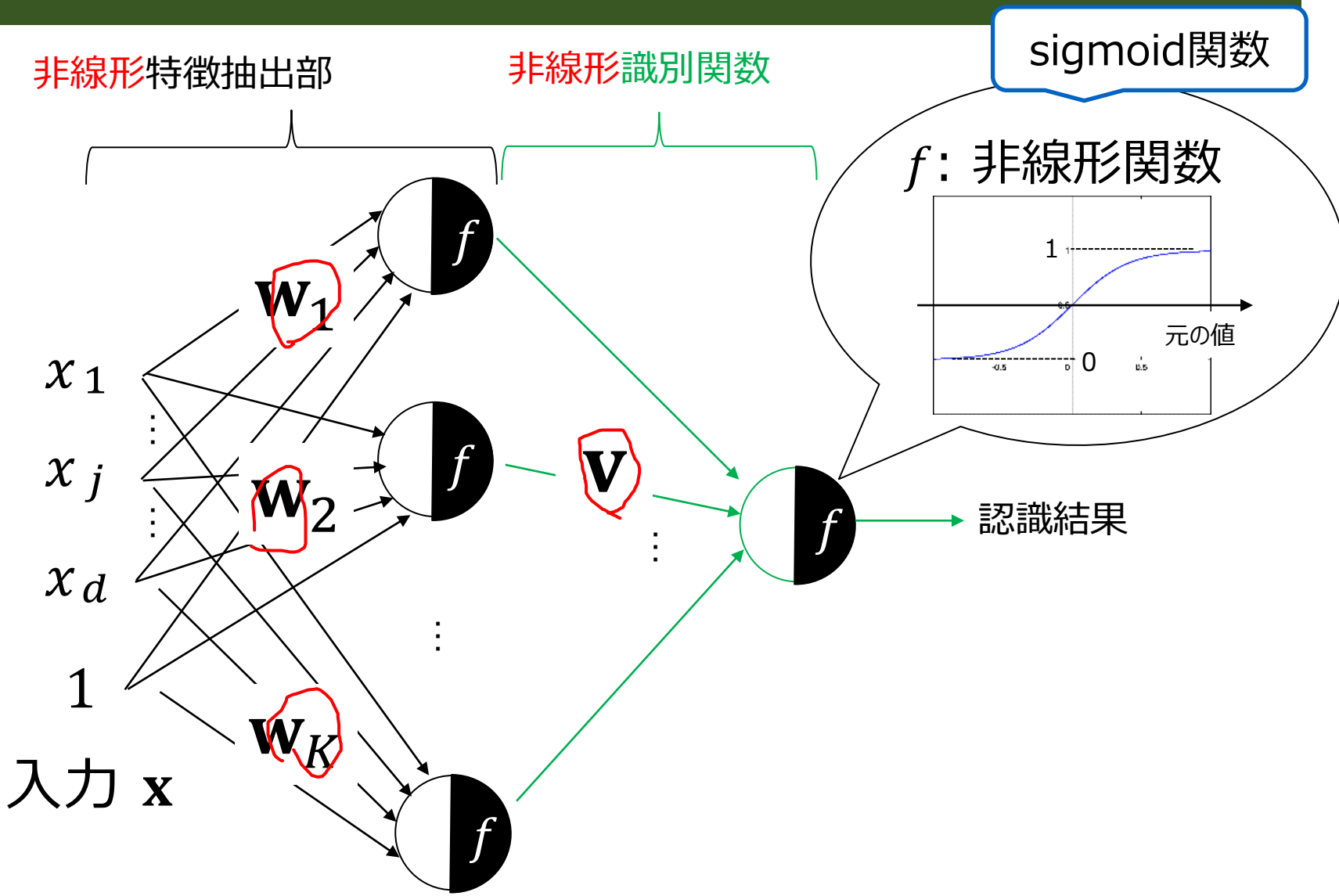
- 同じ機能のニューロンで，多機能(特徴抽出＆識別)を実現



- 脳も似たような感じ?
  - 同じような神経細胞で多機能を実現

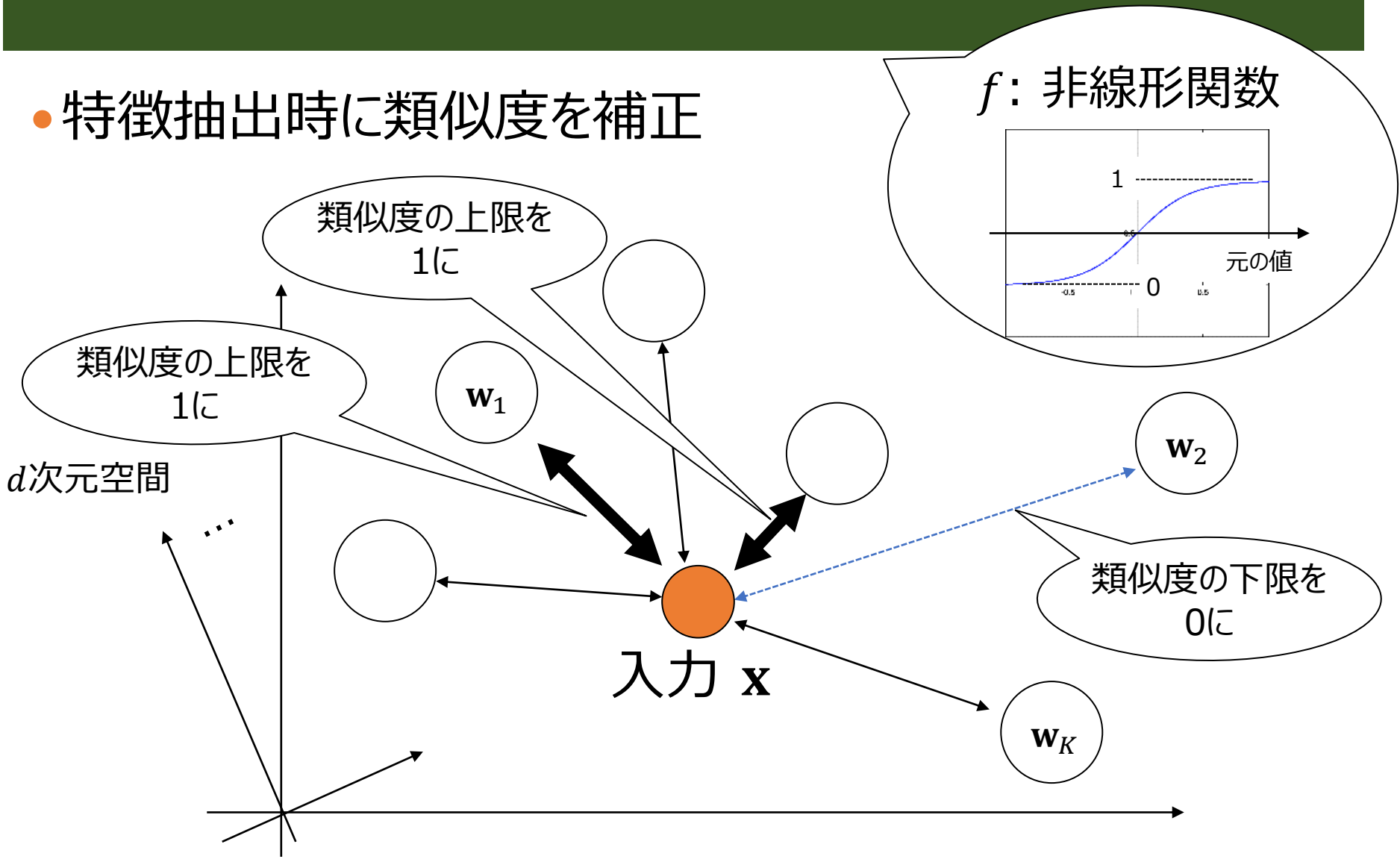


(とりあえず無視した)非線形関数は？

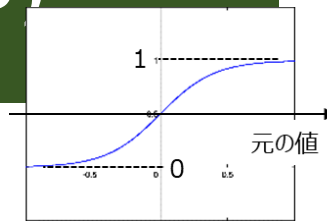


非線形関数の効果の直感的理解(1/3)

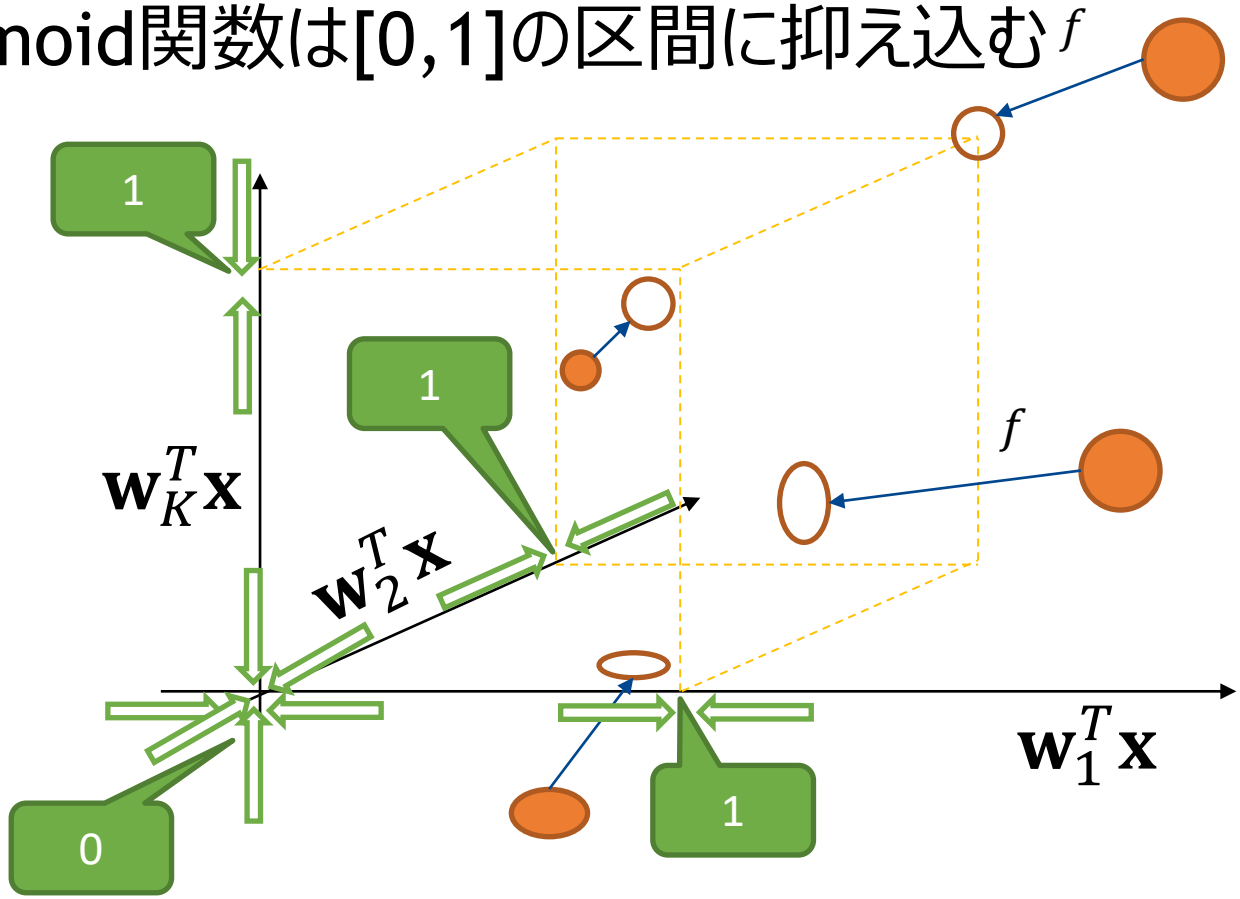
- 特徴抽出時に類似度を補正



非線形関数の効果の直感的理解(2/3)

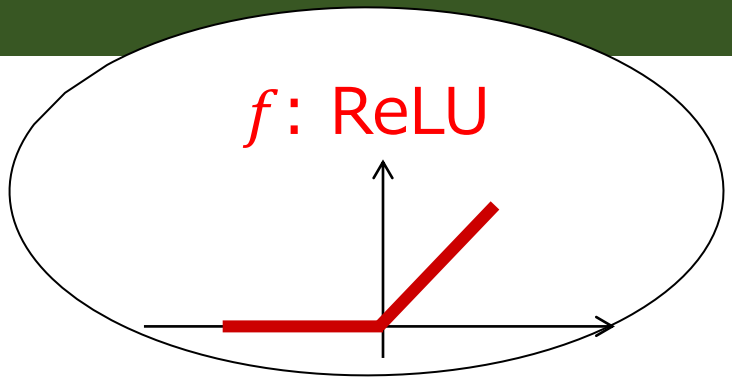


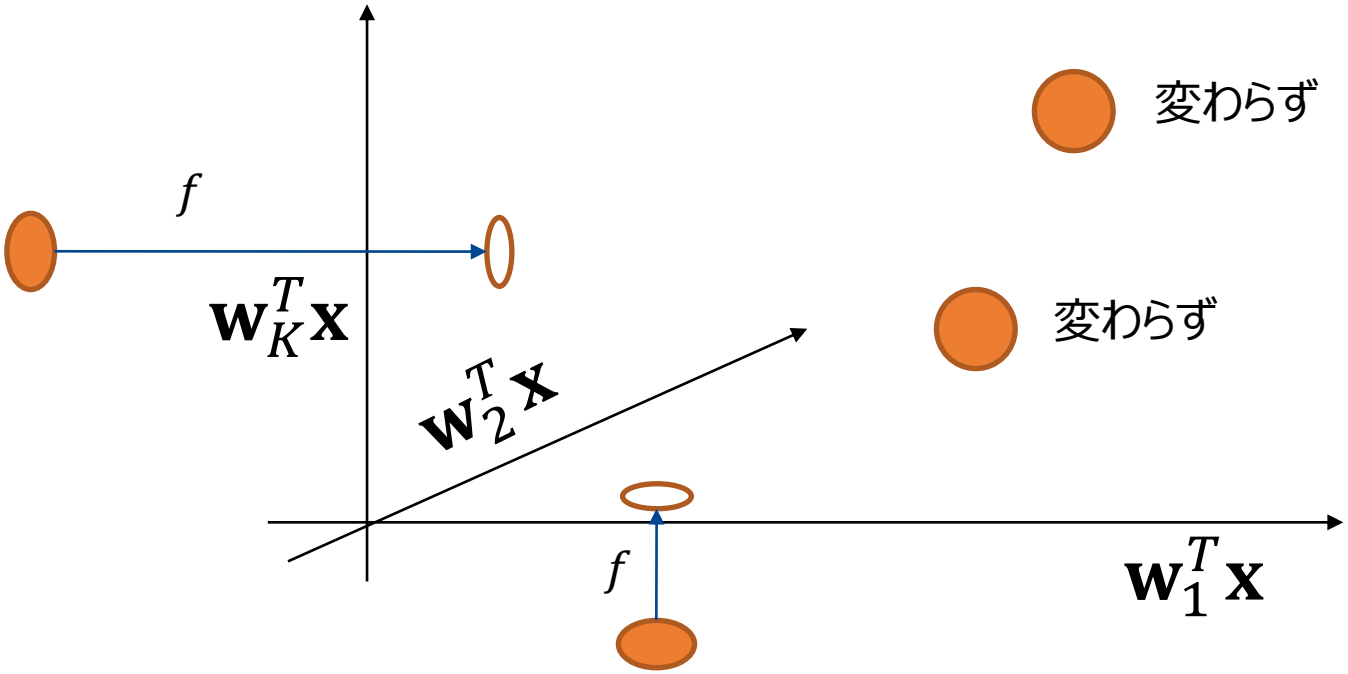
- 類似度特徴空間での見方
- sigmoid関数は[0,1]の区間に抑え込む



非線形関数の効果の直感的理解(3/3)

- 類似度特徴空間での見方
- ReLUは[0,∞)に抑え込む





誤差逆伝播法の概略

$\mathbf{w}_k$を変えて誤差を最小にしたい

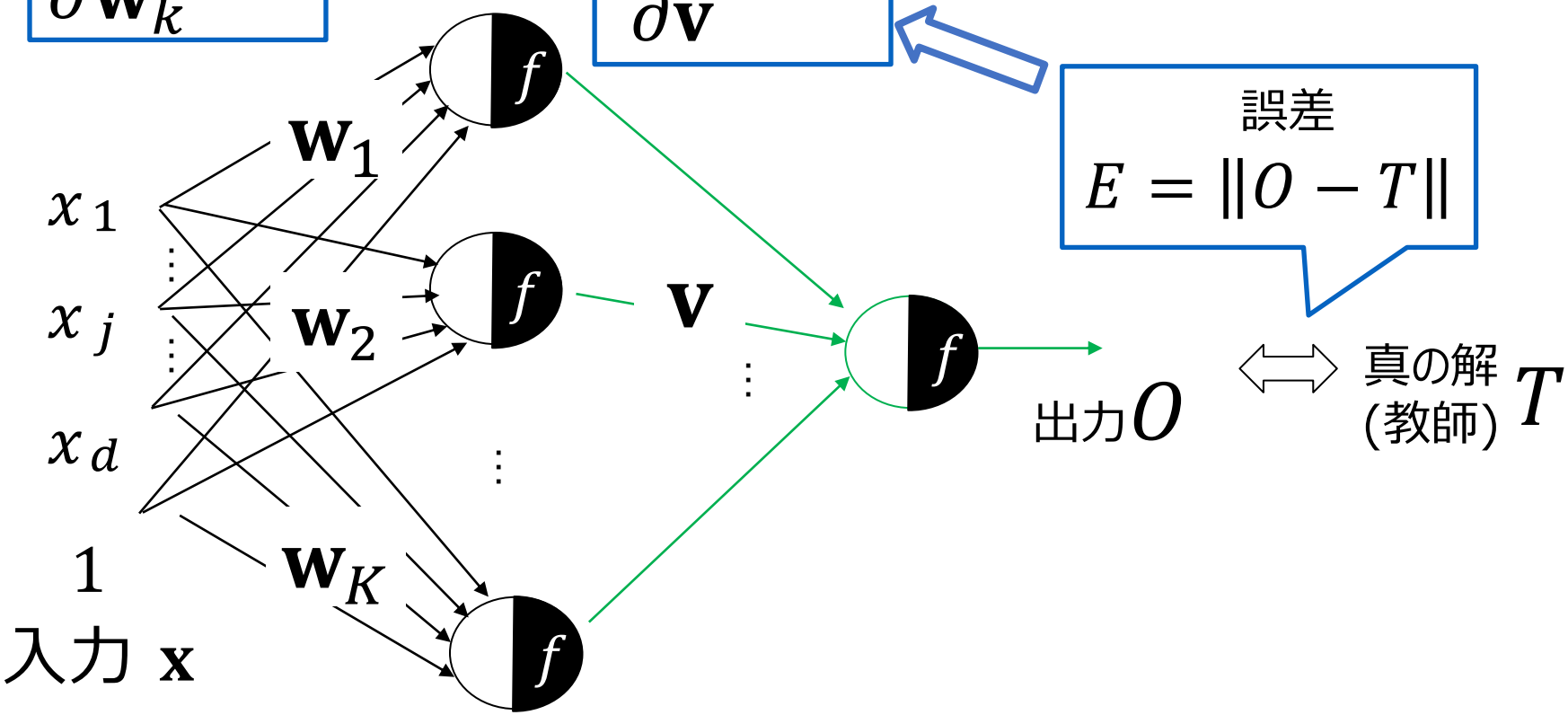
$$\frac{\partial E}{\partial \mathbf{w}_k} = 0$$

$\mathbf{v}$を変えて誤差を最小にしたい

$$\frac{\partial E}{\partial \mathbf{v}} = 0$$

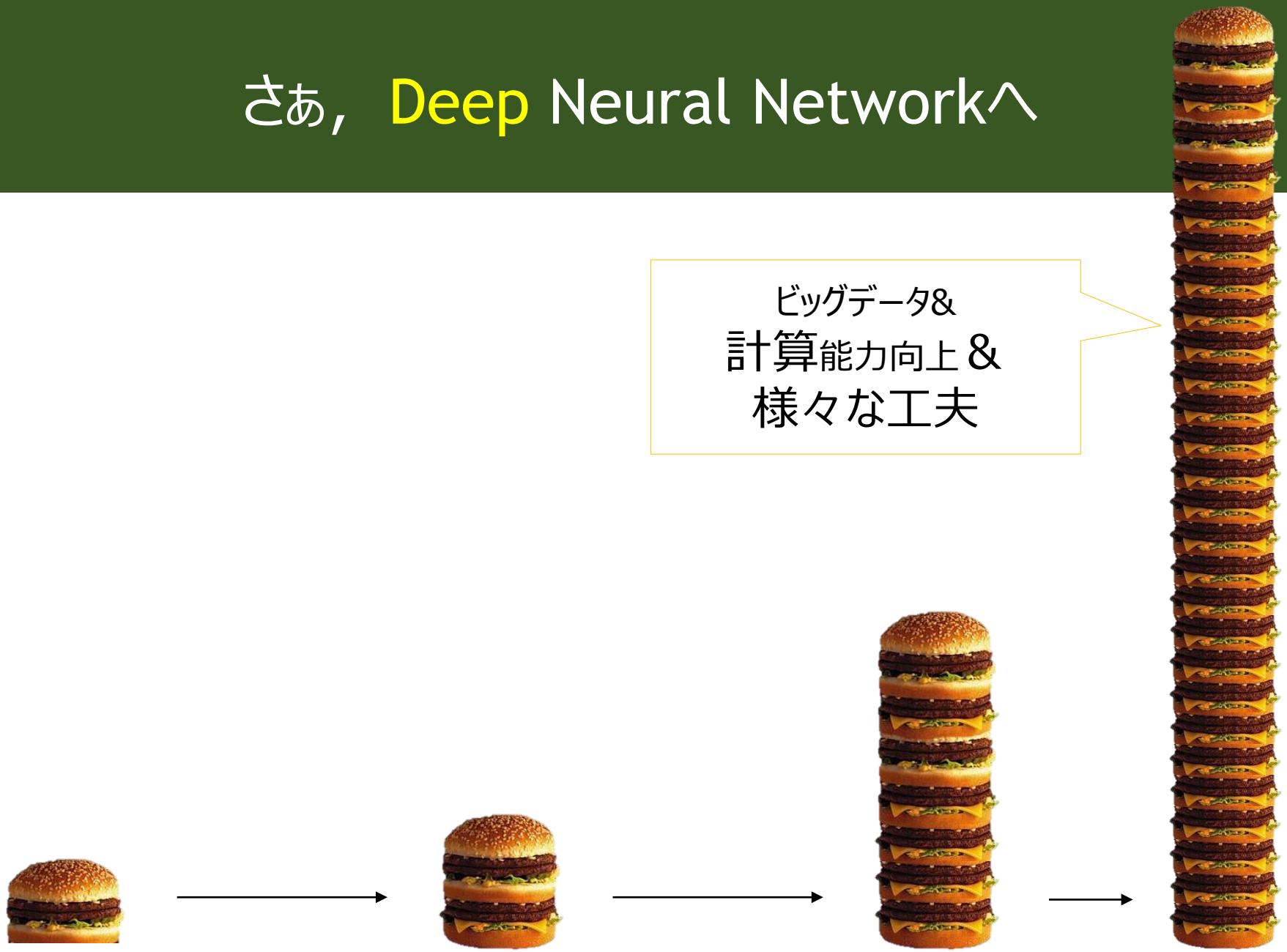
誤差

$$E = \|O - T\|$$



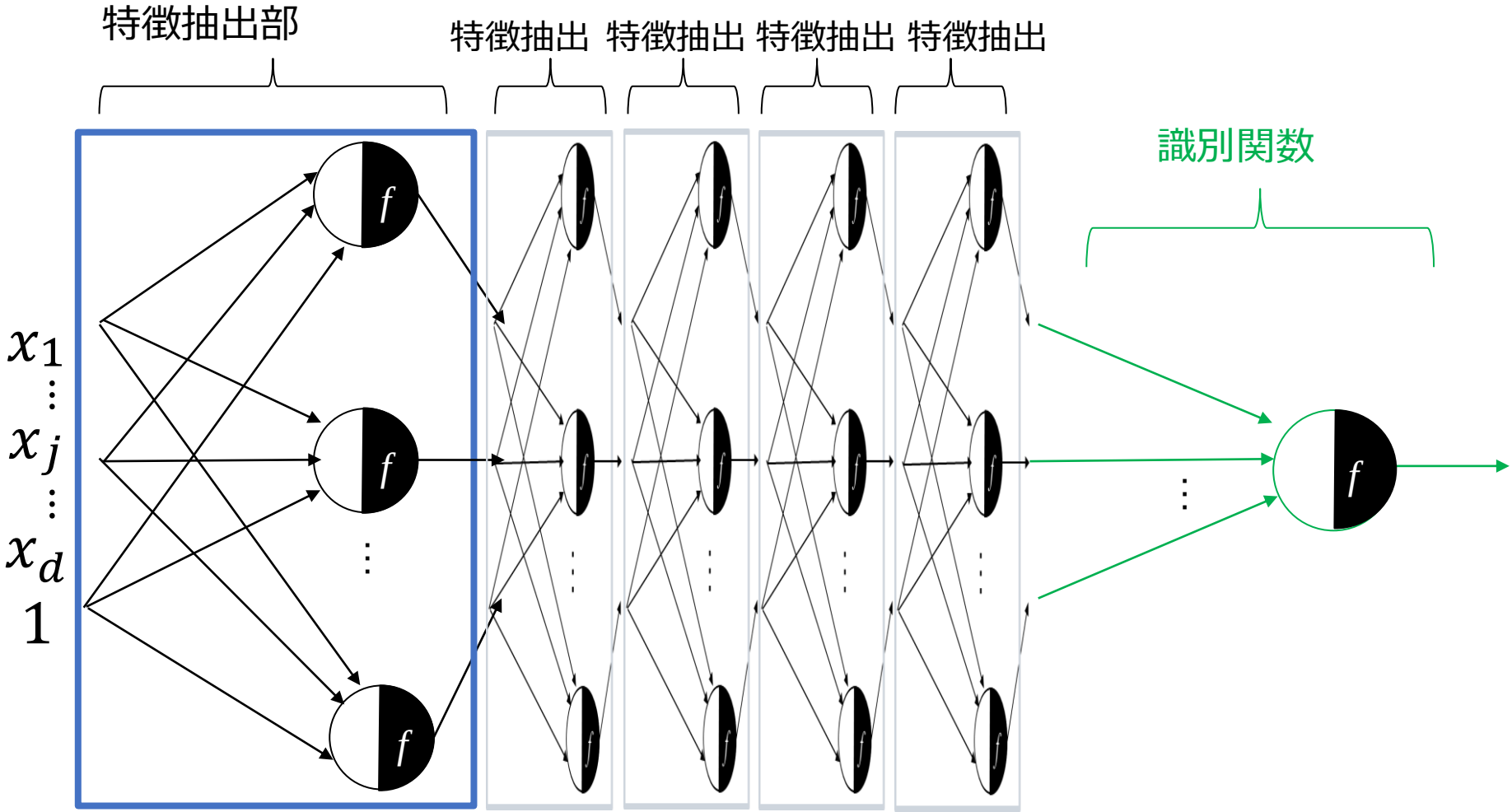
さぁ，Deep Neural Networkへ

ビッグデータ&
計算能力向上＆
様々な工夫

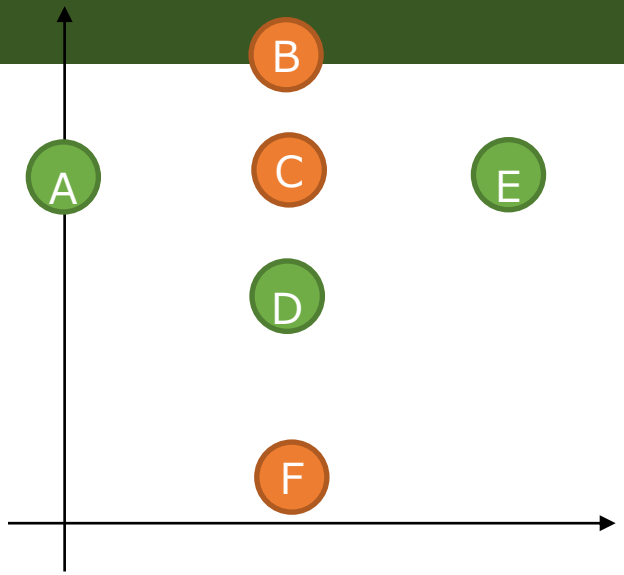


Deep Neural Networkの構造

- 特徴抽出を繰り返すところがポイント!

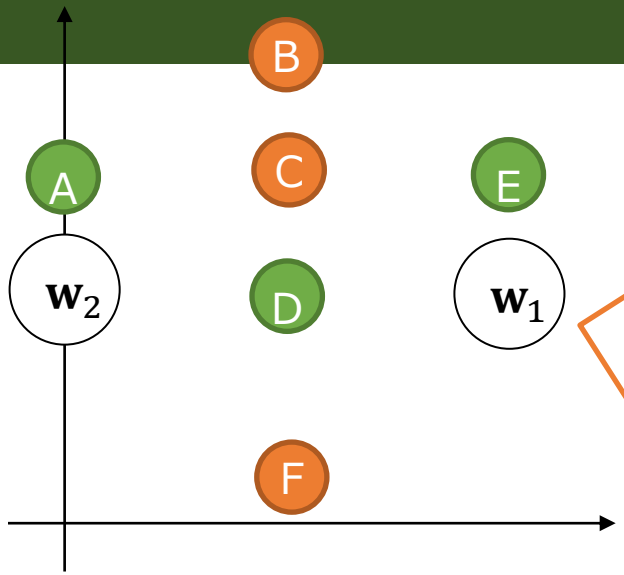


なぜ何度も特徴抽出？

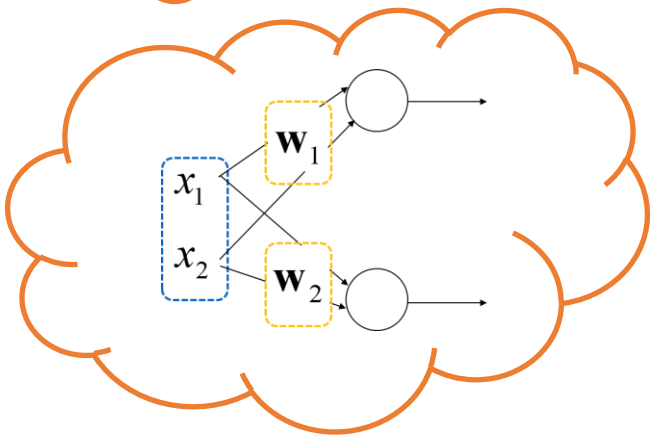


例えば,
みかんとレモンの認識問題：
ゴチャゴチャ混ざっていて,
誤認識が起きそう

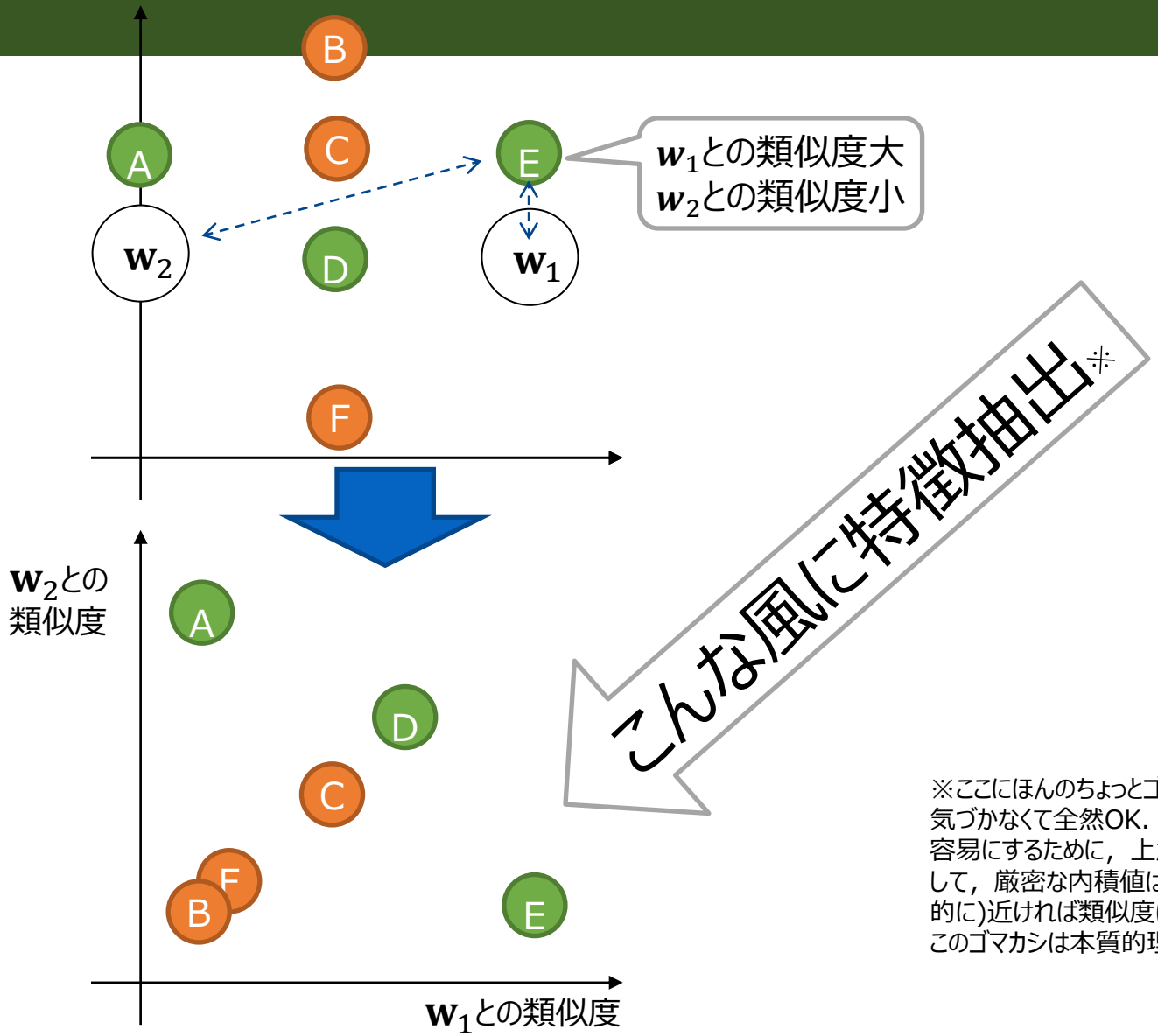
なぜ何度も特徴抽出？



第一層

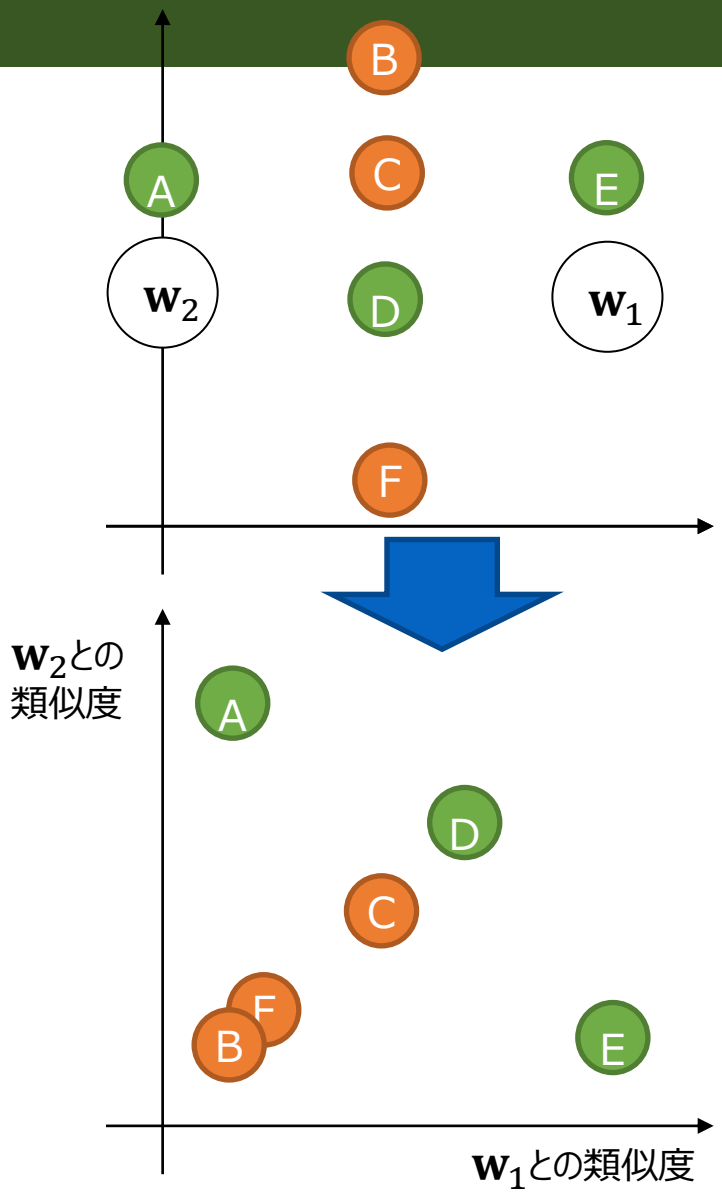


なぜ何度も特徴抽出？



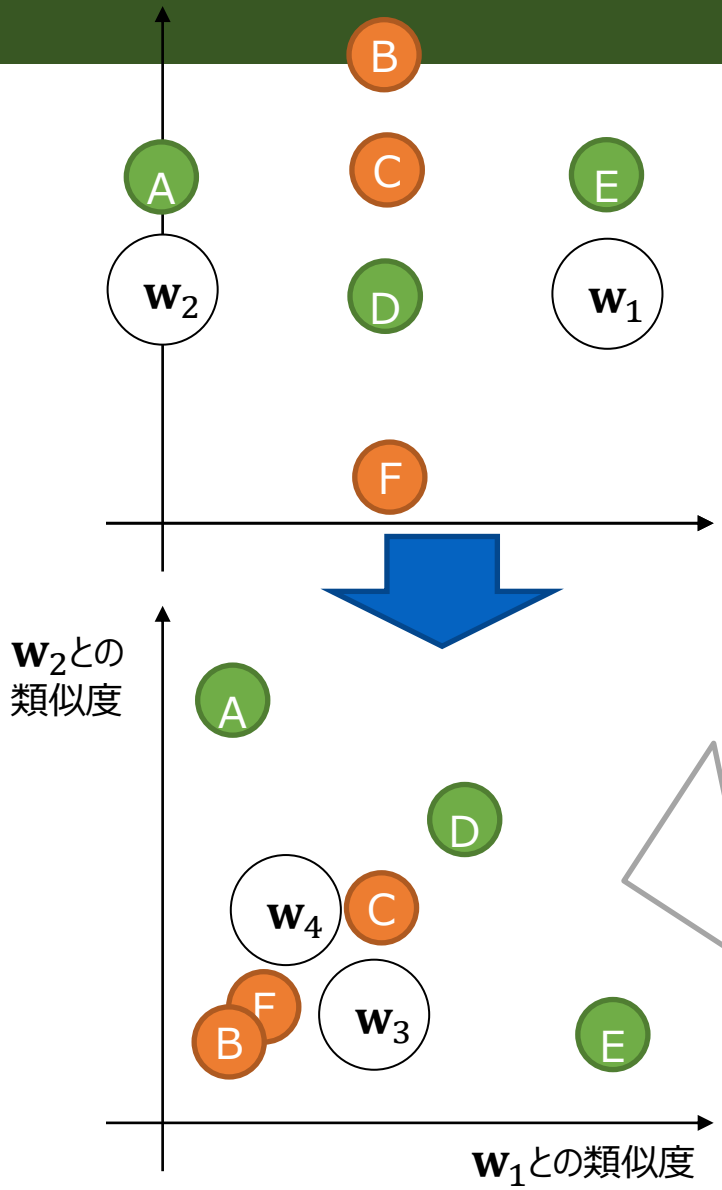
※ここにほんのちょっとゴマカシがあります．気づかない人は気づかなくて全然OK．一応白状しておくと，仕組みの理解を容易にするために，上から下に空間変換する際との類似度として，厳密な内積値は使っていません．図上では，(距離的に)近ければ類似度は大きい，という感じで空間変換してます．このゴマカシは本質的理解を妨げるものではないと信じてます．

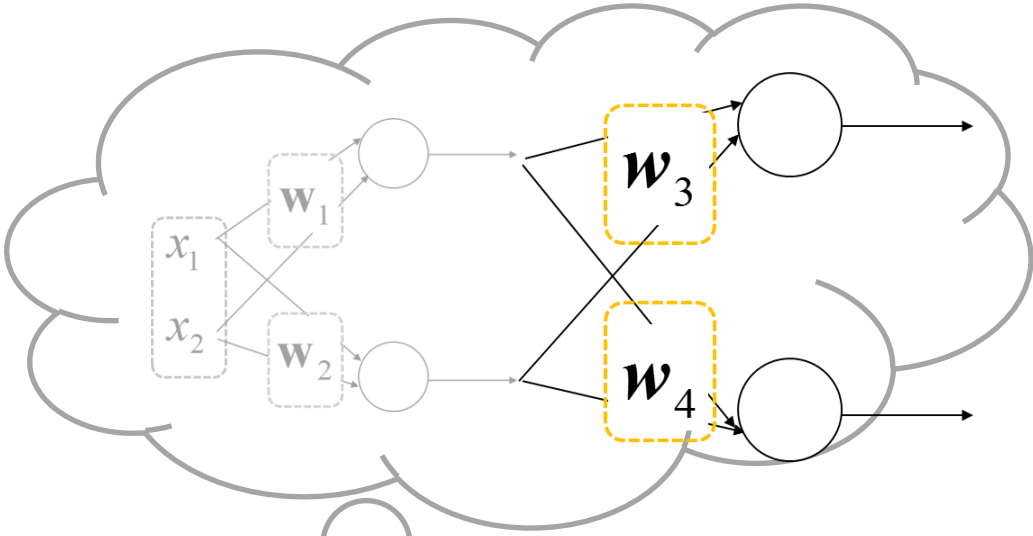
なぜ何度も特徴抽出？



上の状況より，
よりゴチャゴチャ度は減ったが
もうちょっとキレイに分けたい
(専門的には「まだ線形分離不能」)

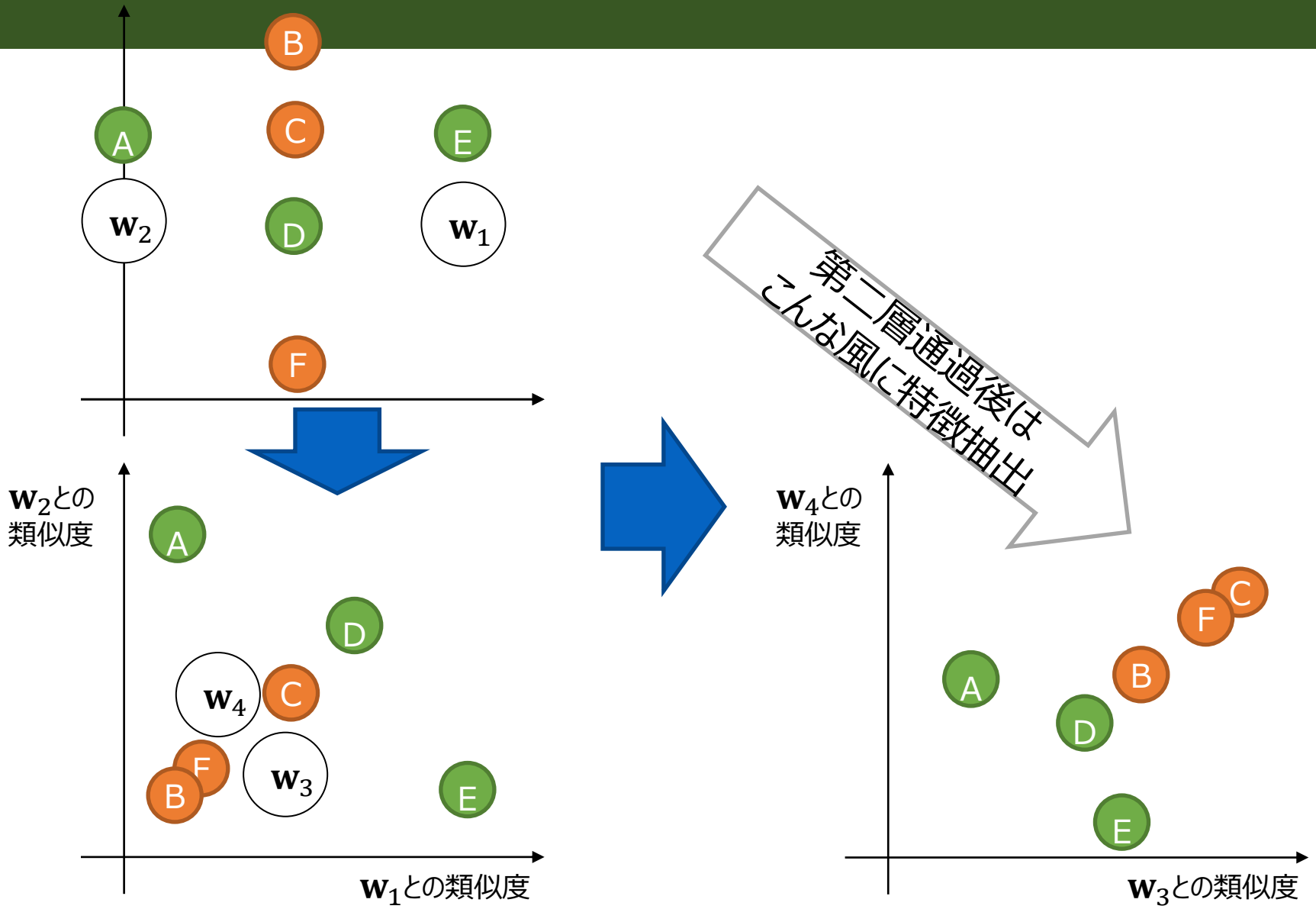
なぜ何度も特徴抽出？



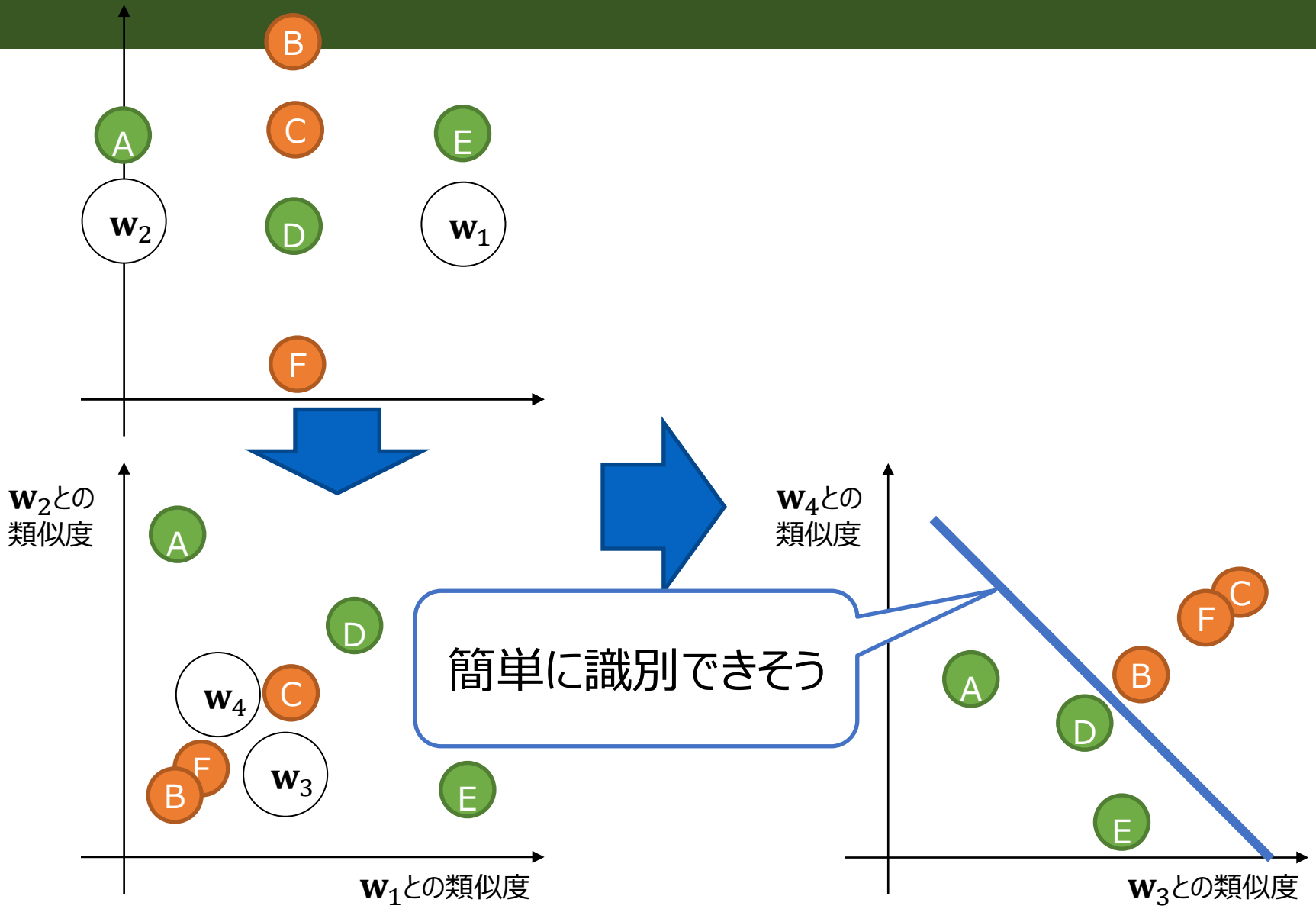


第二層へ渡す

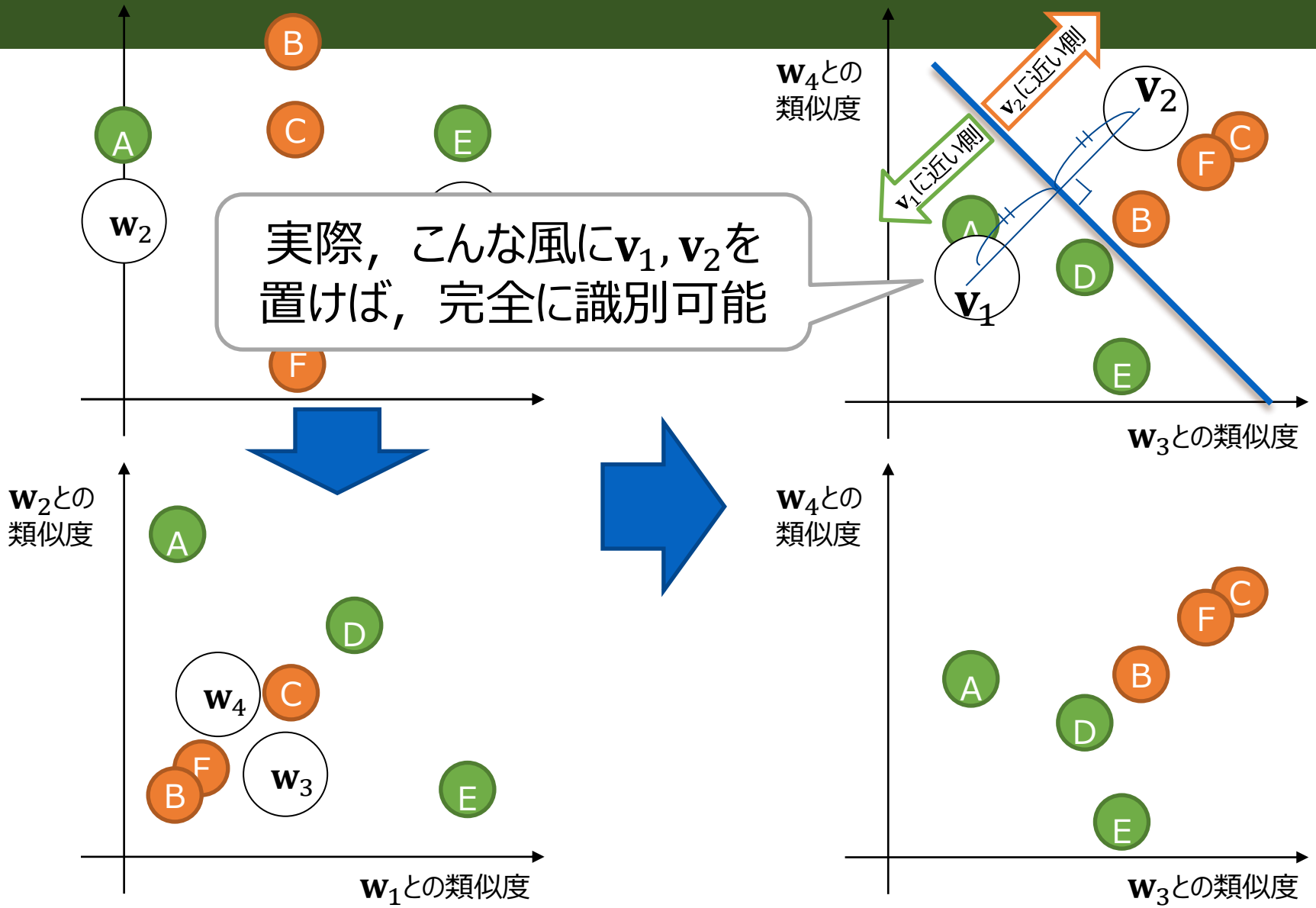
なぜ何度も特徴抽出？


A
B
C
D
E
F
$\mathbf{w}_2$
$\mathbf{w}_1$
$\mathbf{w}_2$との
類似度
$\mathbf{w}_4$
$\mathbf{w}_3$
$\mathbf{w}_1$との類似度
第二層通過後は
こんな風に特徴抽出
$\mathbf{w}_4$との
類似度
$\mathbf{w}_3$との類似度

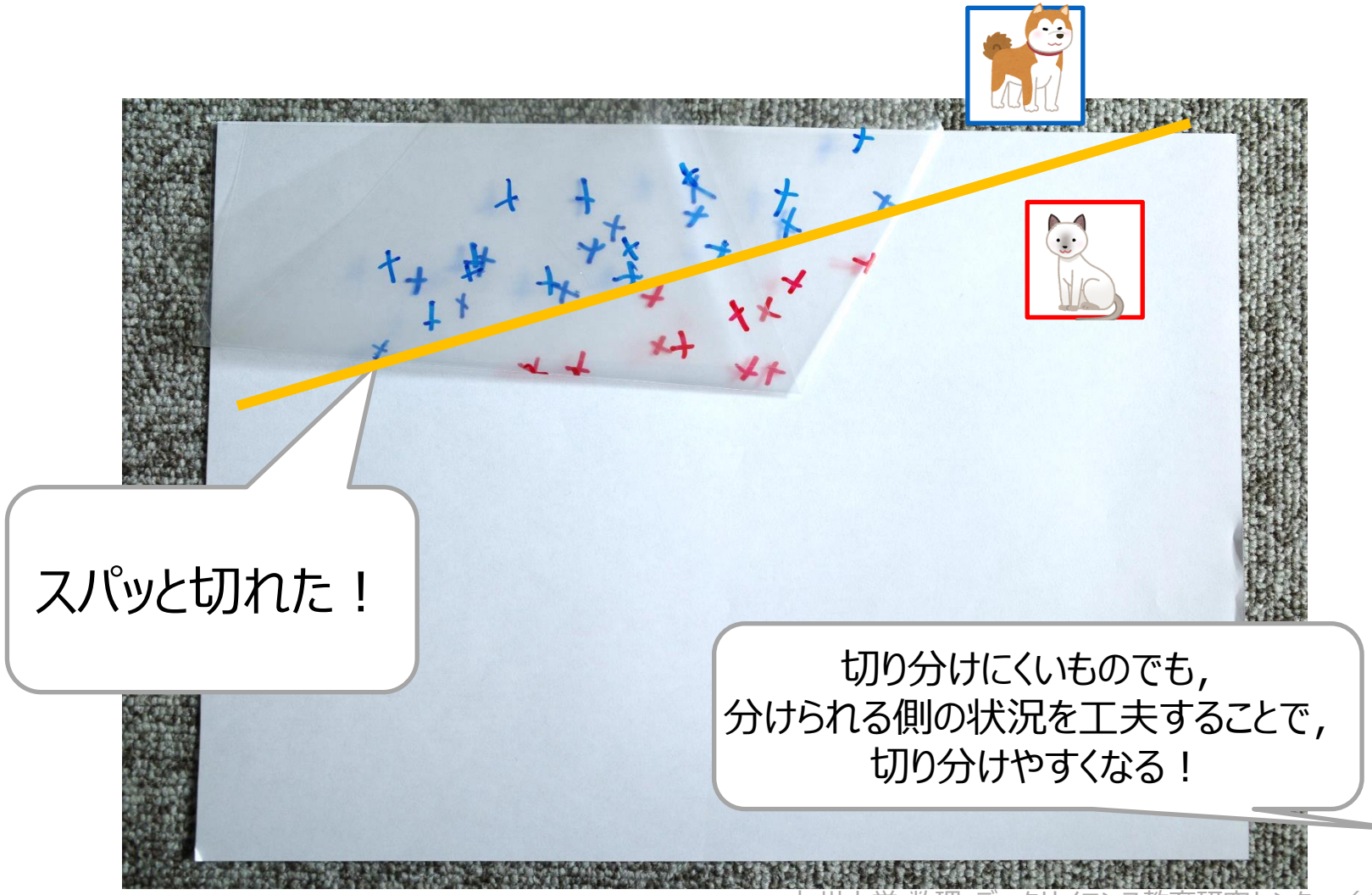
なぜ何度も特徴抽出？


A
B
C
D
E
F
$\mathbf{w}_2$
$\mathbf{w}_1$
$\mathbf{w}_2$との
類似度
$\mathbf{w}_4$
$\mathbf{w}_3$
$\mathbf{w}_1$との類似度
簡単に識別できそう
$\mathbf{w}_4$との
類似度
$\mathbf{w}_3$との類似度

なぜ何度も特徴抽出？


実際，こんな風に$\mathbf{v}_1, \mathbf{v}_2$を置けば，完全に識別可能
$\mathbf{w}_2$との類似度
$\mathbf{w}_1$との類似度
$\mathbf{w}_4$との類似度
$\mathbf{w}_3$との類似度
$\mathbf{v}_1$に近い側
$\mathbf{v}_2$に近い側
$\mathbf{v}_1$
$\mathbf{v}_2$
$\mathbf{w}_2$
$\mathbf{w}_4$
$\mathbf{w}_3$
A
B
C
D
E
F

そう，さっきの話

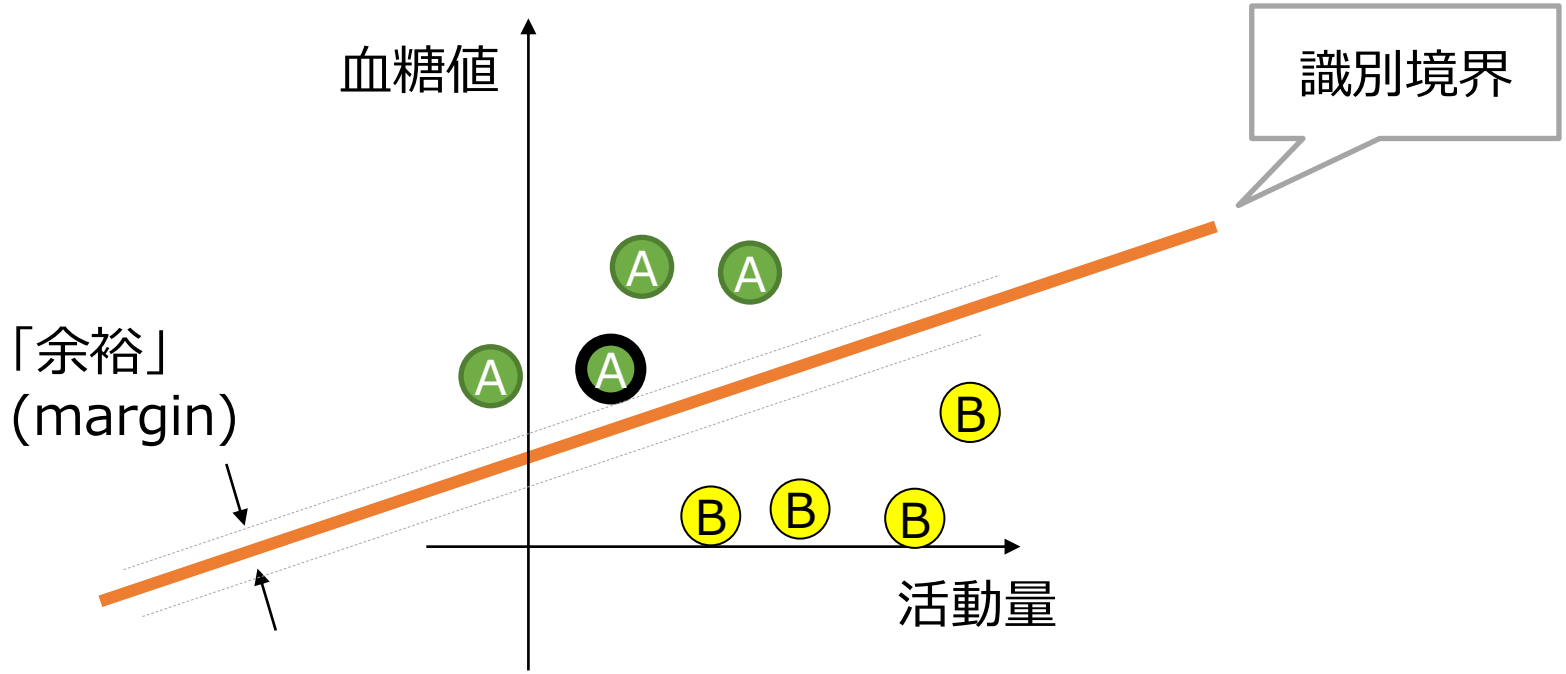


【付録3】
サポートベクトルマシン
Support Vector Machines (SVM)

識別関数を決定する一つの方法

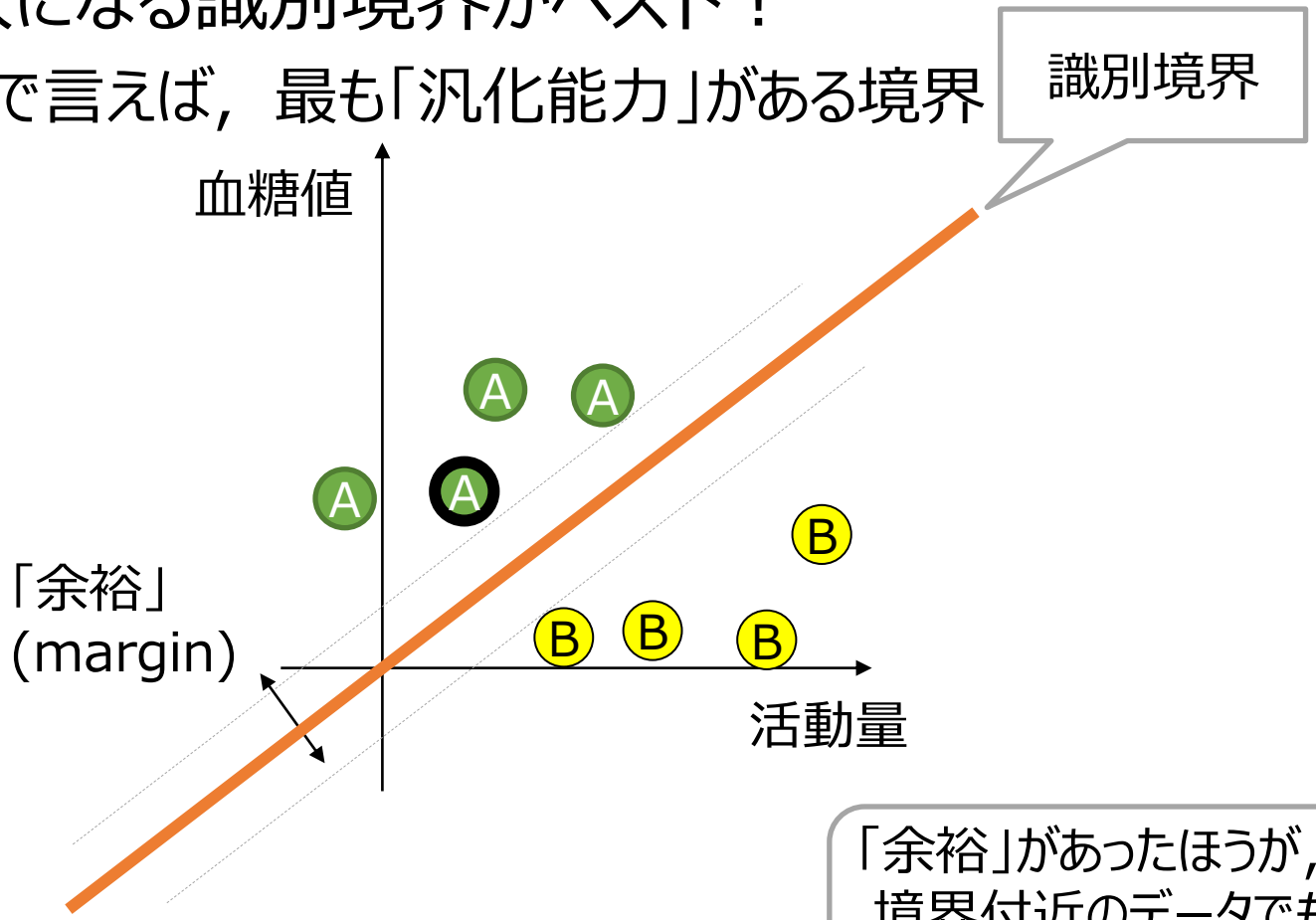
SVMの考え方 (1/2)

- 学習パターンは正しく識別できているが，ちょっとでもずれたら誤分類発生？



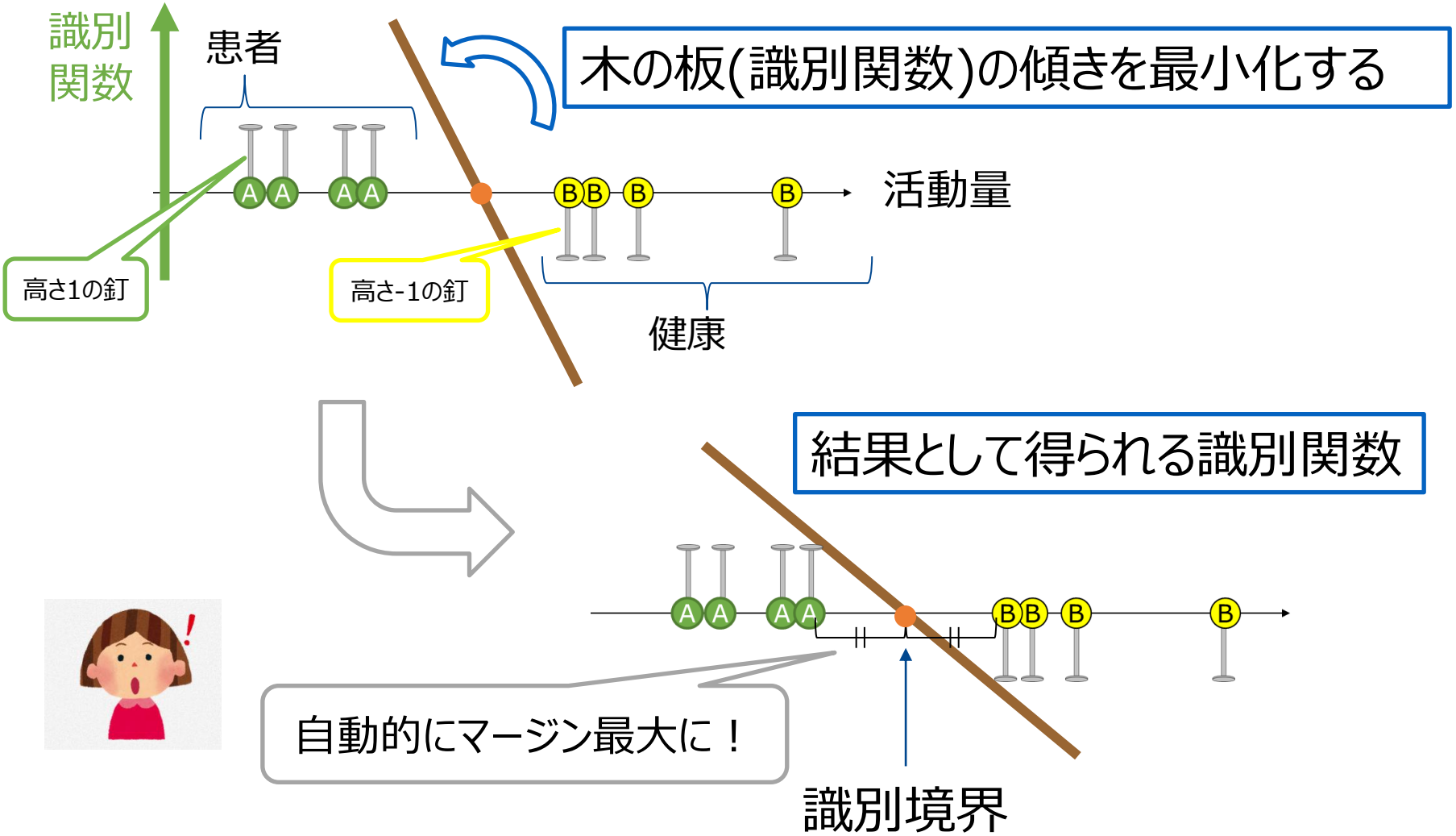
SVMの考え方 (2/2)

- 余裕が最大になる識別境界がベスト！
  - 難しい言葉で言えば，最も「汎化能力」がある境界

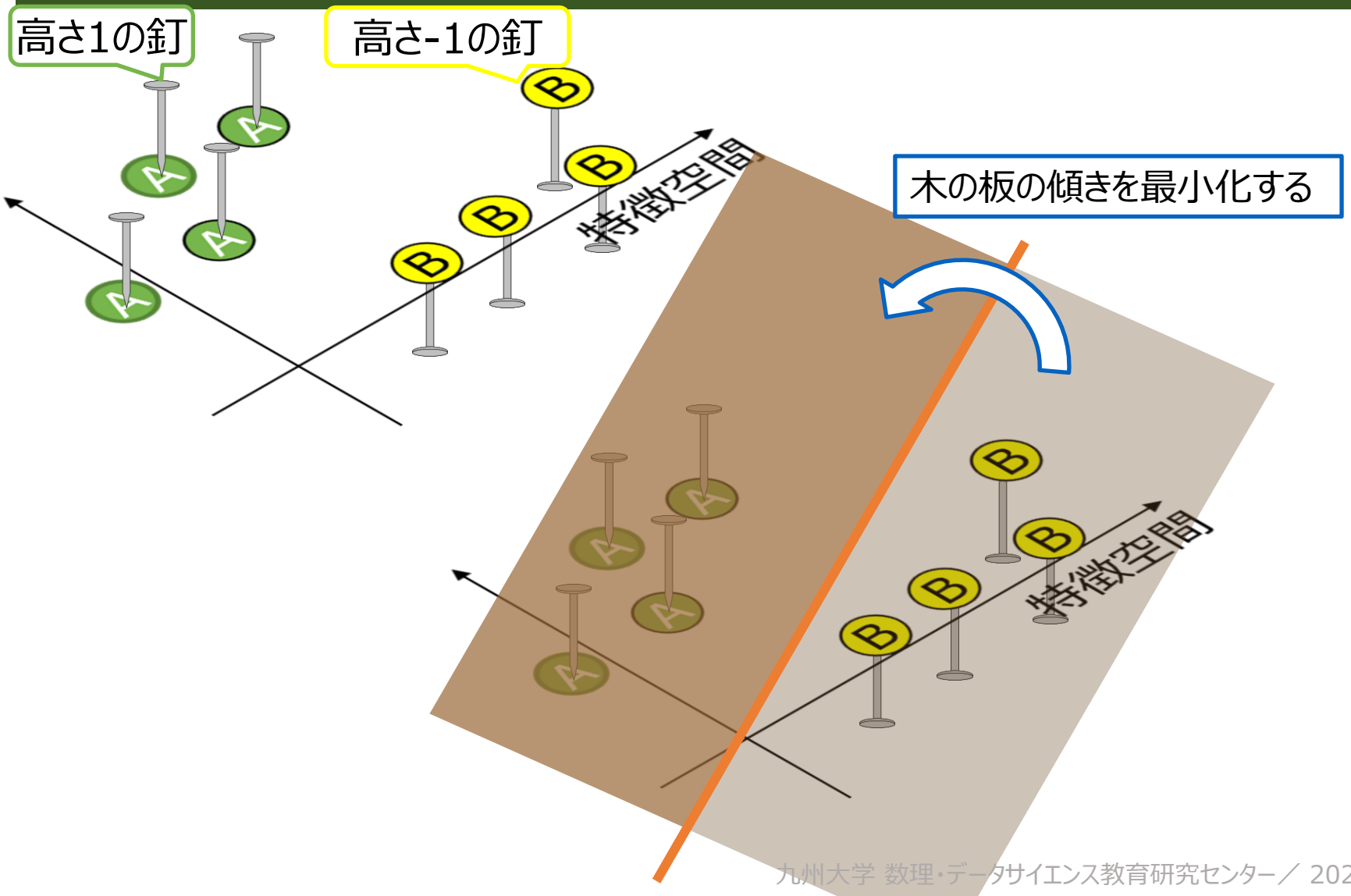


「余裕」があったほうが，
境界付近のデータでも
正しく認識できそう

マージン最大化はどうやって？
１次元の場合

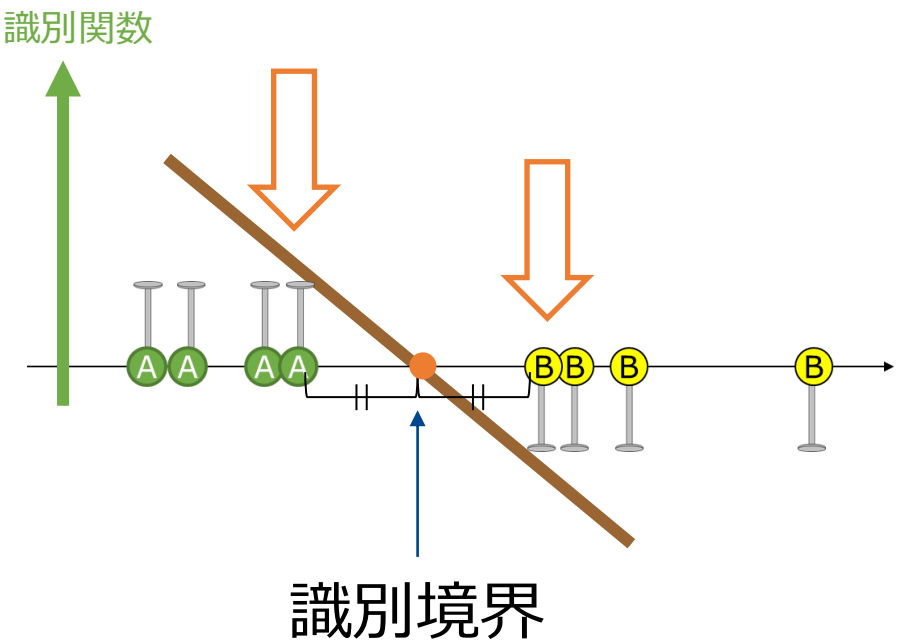


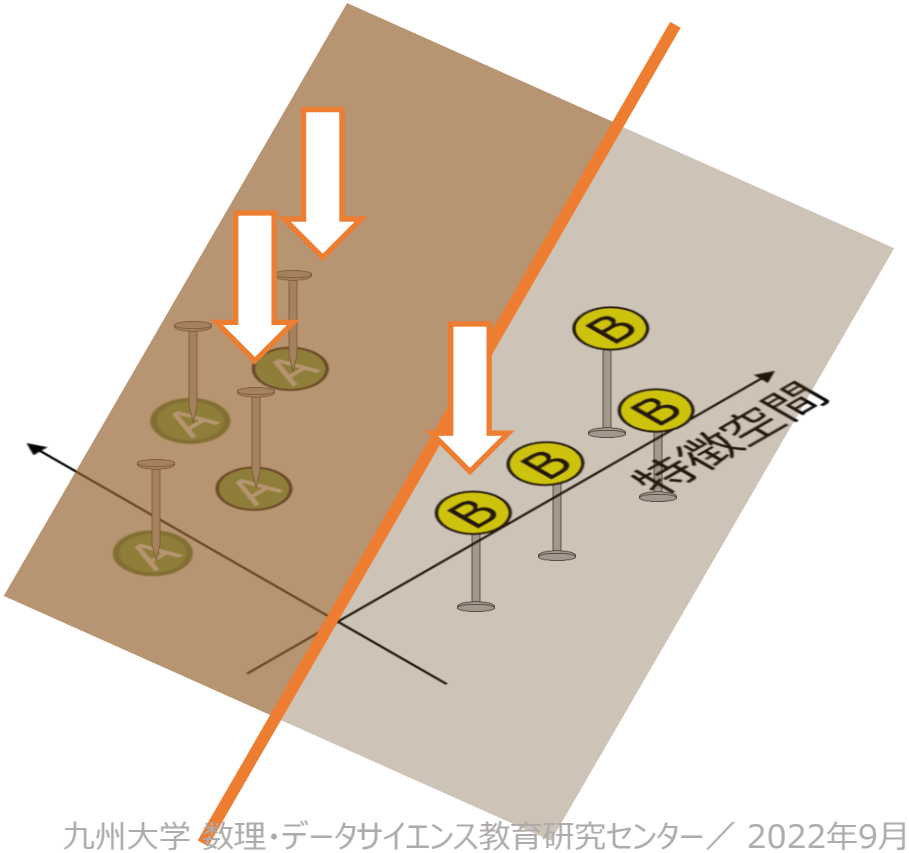
マージン最大化はどうやって？
2次元の場合



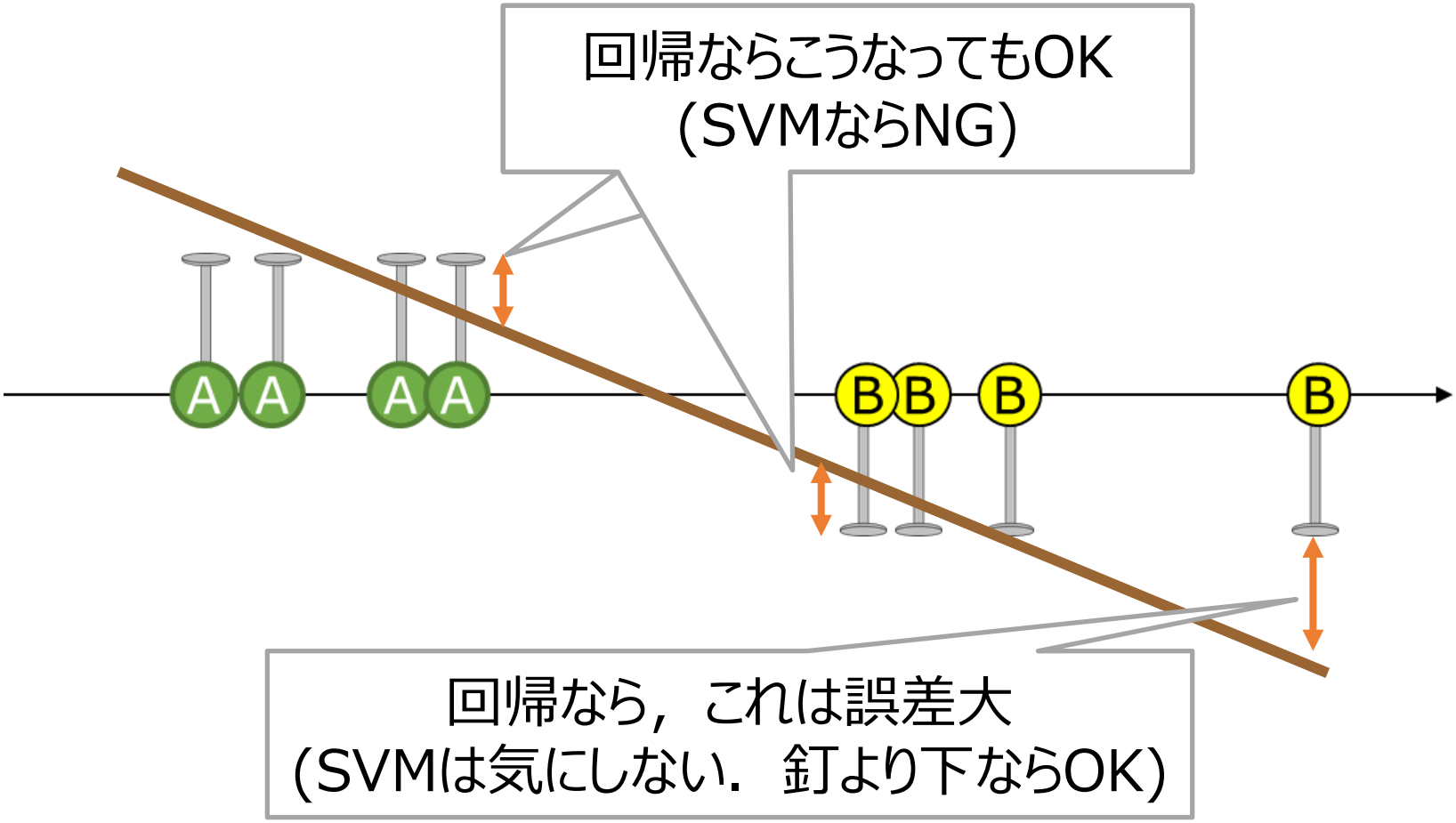
サポートベクトル？

- 識別関数を決めているデータのこと＝木の板が当たっているデータ

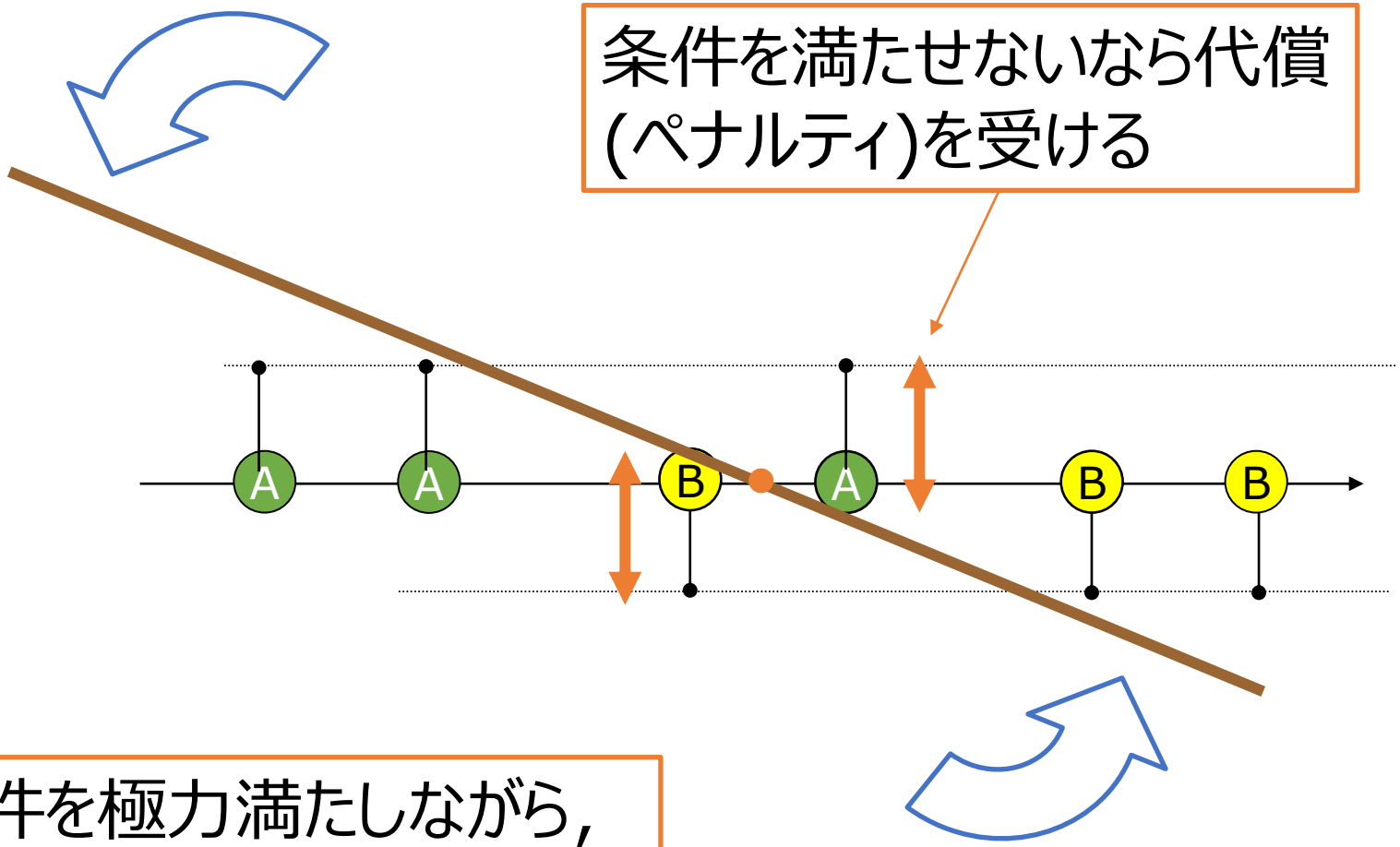




参考：回帰分析と似てるがちょっと違う



さらに参考：きれいに分けられない場合は？
→ソフトマージンSVM



条件を極力満たしながら,
ペナルティも最小化する