深層学習入門

回帰と分類

一関高専 未来創造工学科 情報・ソフトウェア系
小池　敦

教師あり学習（復習）

- 教師あり学習におけるアプローチ

$\boxed{f(x)}$ これを求めたい

x	y
1	1
2	8
3	27
4	64

$f(x)$の例：
$f(x) = x^3$

概要

- 教師あり学習の学習フェーズにおいて，
どのように学習するのかを説明する
- 基本的には以下の流れ
1. 適当に関数を作って試してみる
2. うまくフィットするまで，関数を微修正し続ける

簡単な例

$\boxed{f(x)\ ?}$

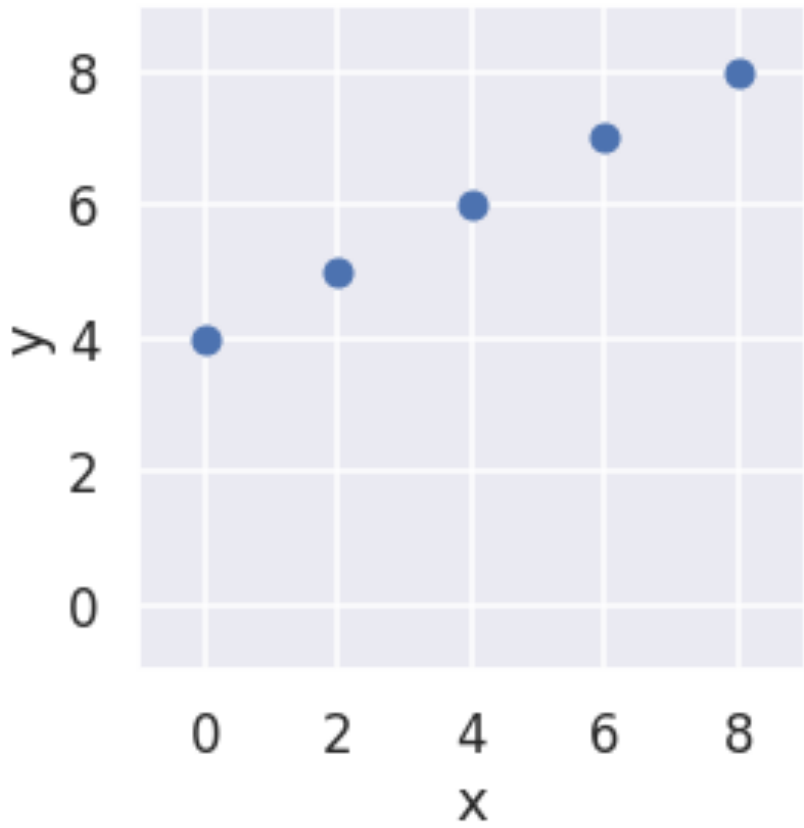
x	y
0	4
2	5
4	6
6	7
8	8

簡単な例

$\boxed{f(x)\text{？}}$

x	y
0	4
2	5
4	6
6	7
8	8

図示してみる



簡単な例：関数の求め方

- 直線っぽいので

  $y = ax + b$

  と置くことにし,

  aとbの値を求めることにする

- 多くの機械学習手法では,

  式の形（モデル）は人間が決める
  - そうでないやり方も多くある

簡単な例：関数の求め方

- 式の形が全然推定できない時は

$$y = a_{100}x^{100} + a_{99}x^{99} + \cdots + a_1 x + b$$

のように，とりあえず複雑にしておくこともできるが，問題点もある

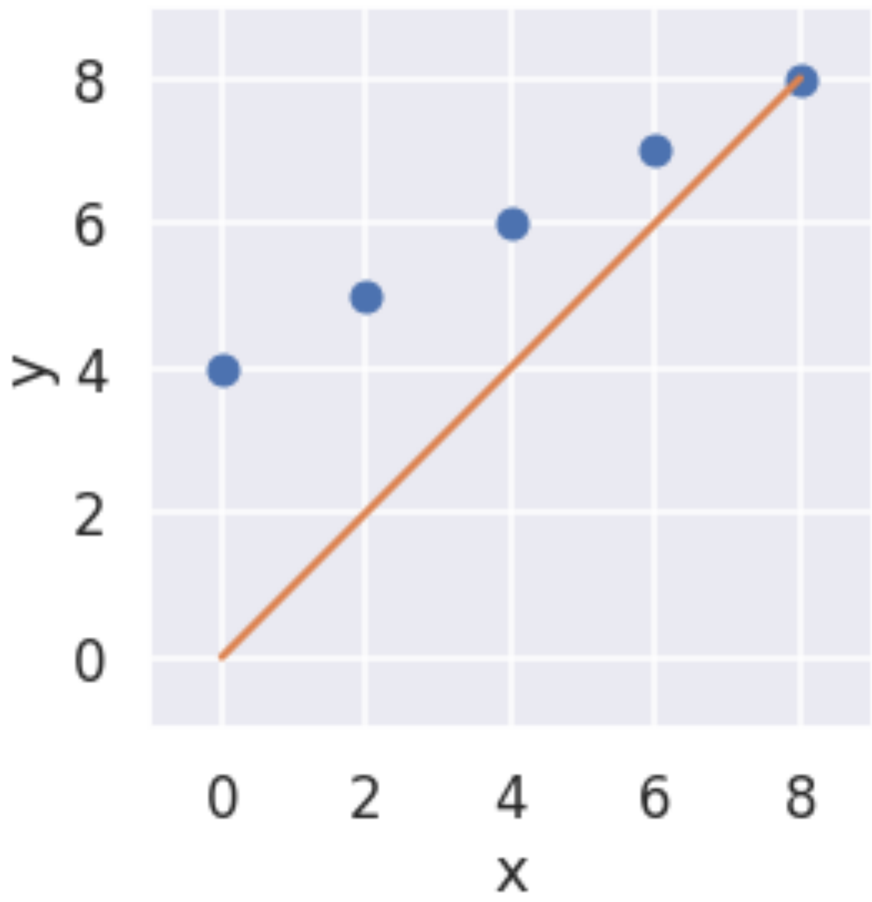
  - このようなことをすると，
    オーバーフィッティングと呼ばれる問題が起きやすくなる
    → モデルを不必要に複雑にしてはいけない
    （オッカムの剃刀）

簡単な例：関数の求め方

- まずは適当なaとbの値でどれくらいフィットするか試してみる
  - $a = 1$と$b = 0$としてみる（つまり$\mathrm{y} = \mathrm{x}$）

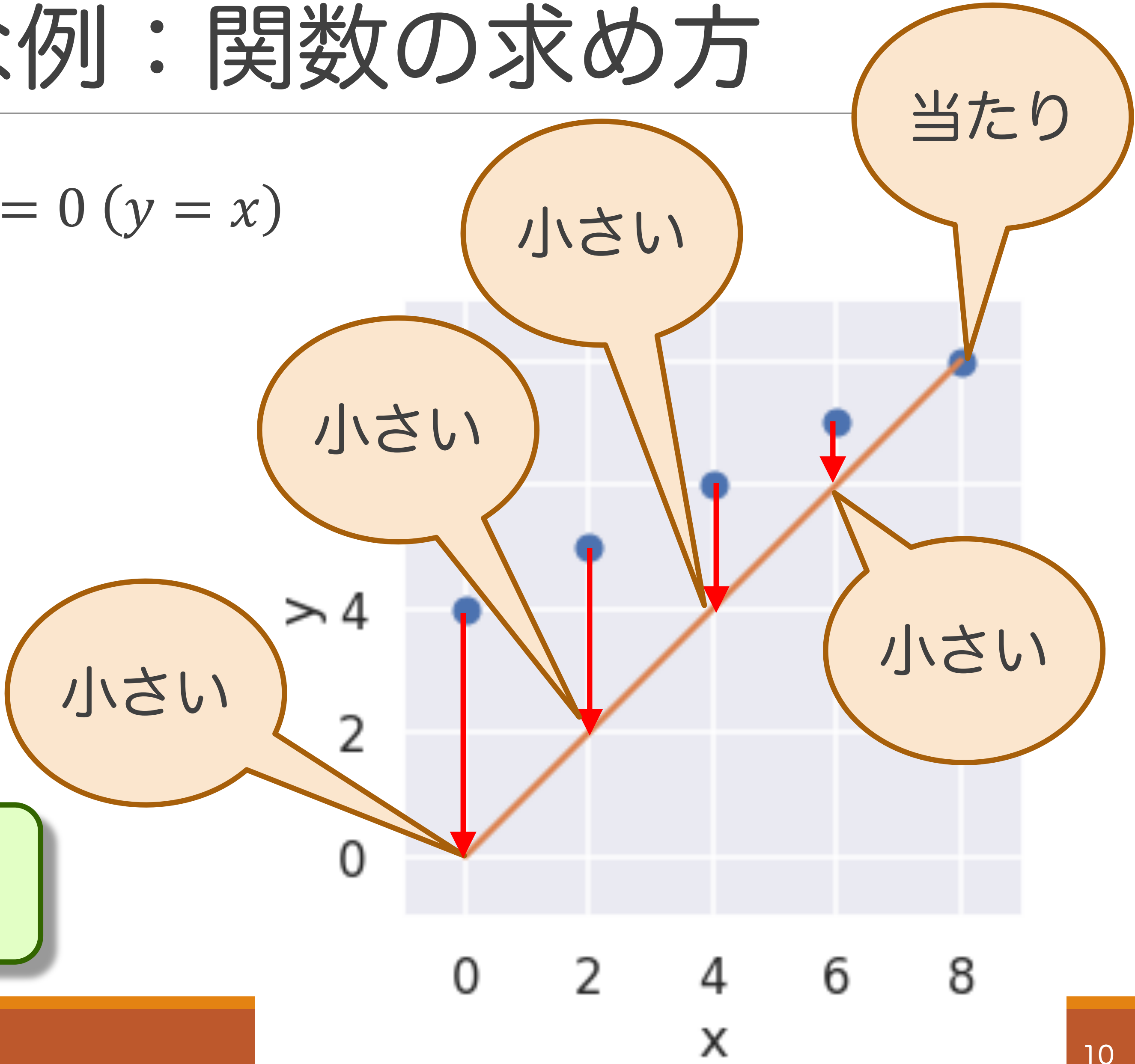
簡単な例：関数の求め方

- $a = 1, b = 0\ (y = x)$



簡単な例：関数の求め方

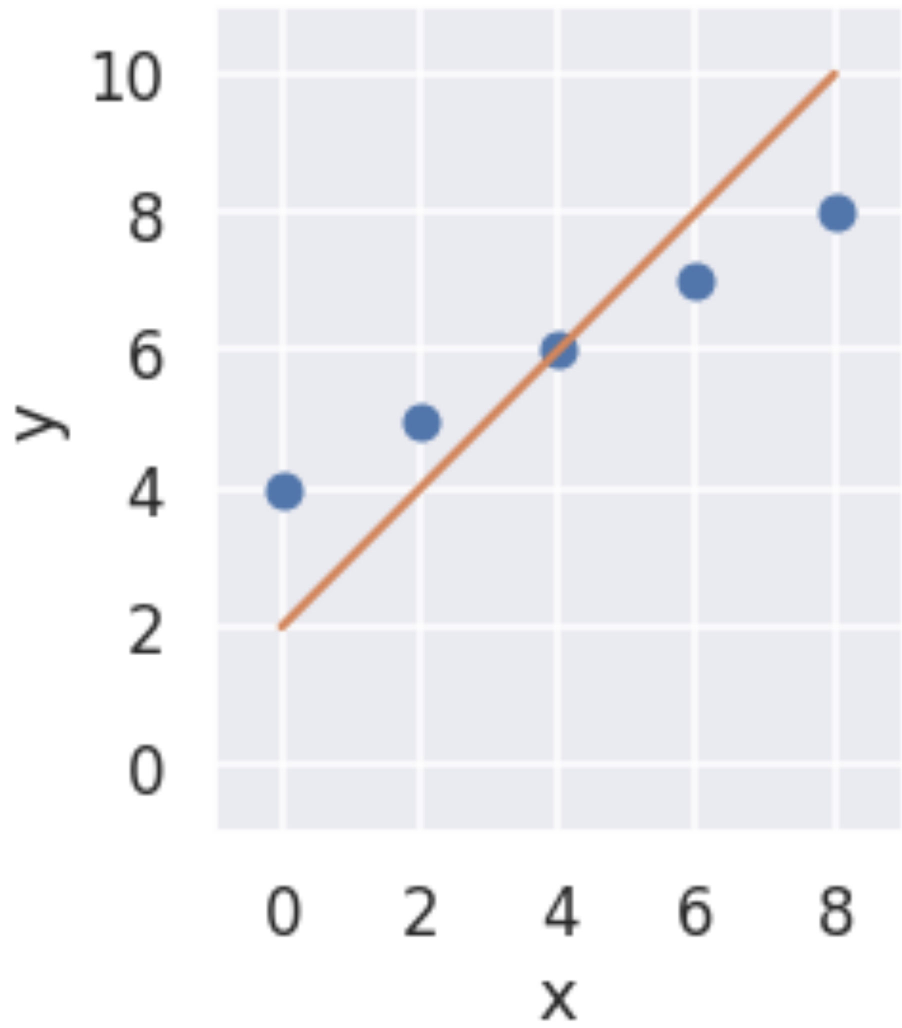
- $a = 1, b = 0 \ (y = x)$



もっと
bを大きくする

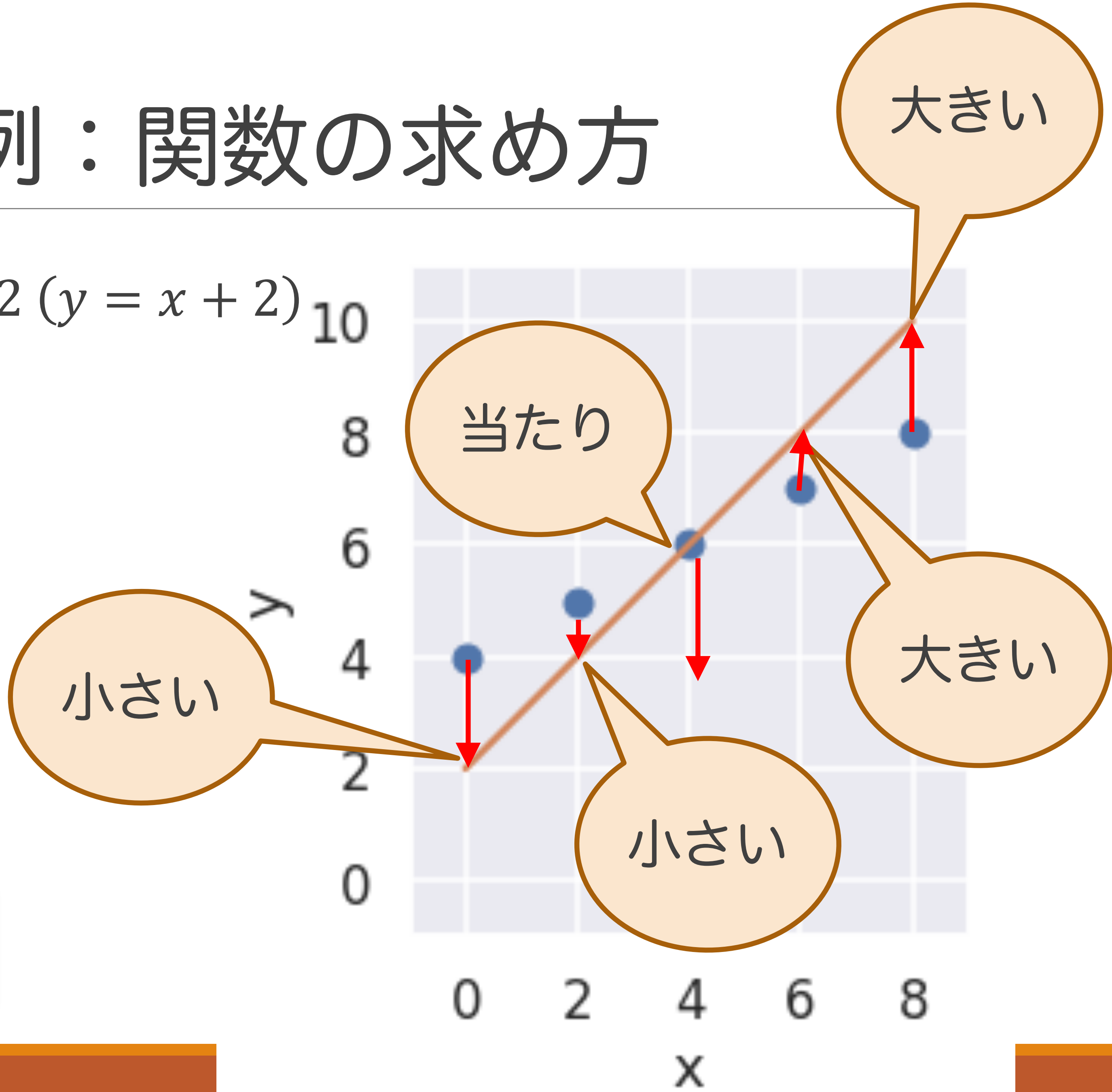
簡単な例：関数の求め方

- $a = 1, b = 2\ (y = x + 2)$



簡単な例：関数の求め方

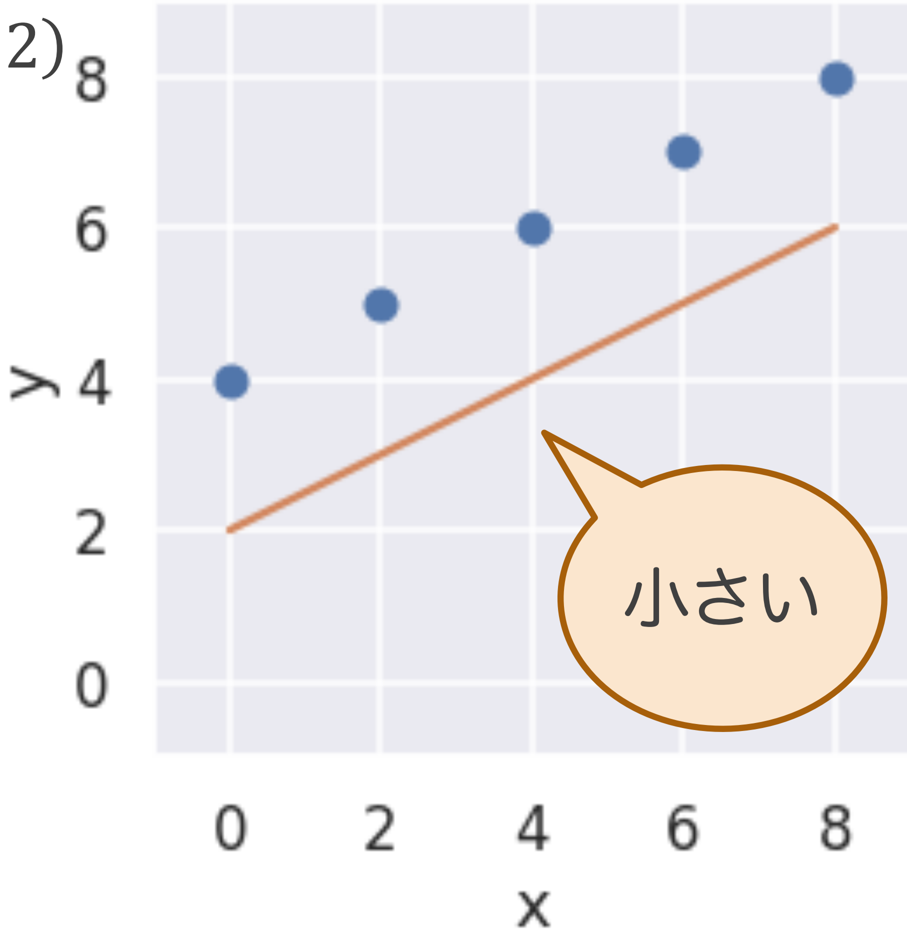
- $a = 1, b = 2\ (y = x + 2)$



もっと
aを小さくする

簡単な例：関数の求め方

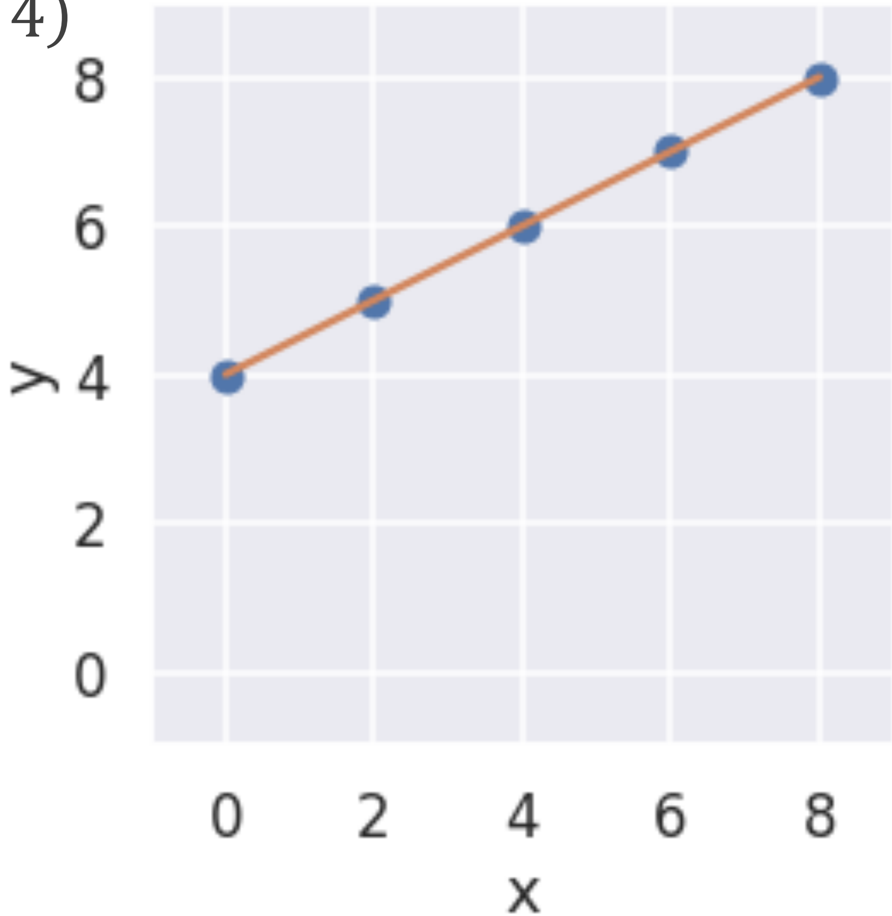
- $a = 0.5, b = 2\ (y = 0.5x + 2)$



もっと
bを大きくする

簡単な例：関数の求め方

- $a = 0.5, b = 4\ (y = 0.5x + 4)$



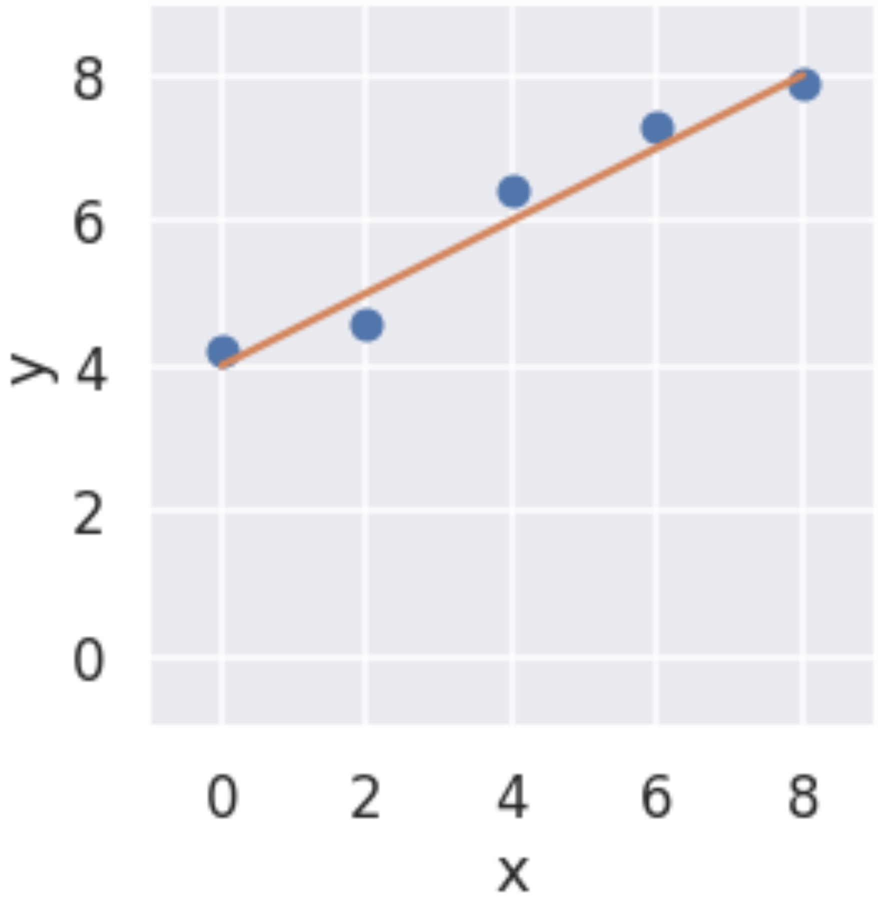
フィット完了！

学習フェーズの基本的な手順

1. 関数の式の形を決める
2. 適当なパラメータで描画してみて，データとの誤差をチェックする
3. 誤差が小さくなるよう繰り返しパラメータの値を修正する

このような手法で必ずしも良い関数が求まるとは限らないが様々な工夫をすることで結構うまくいく

誤差を0にできないケース

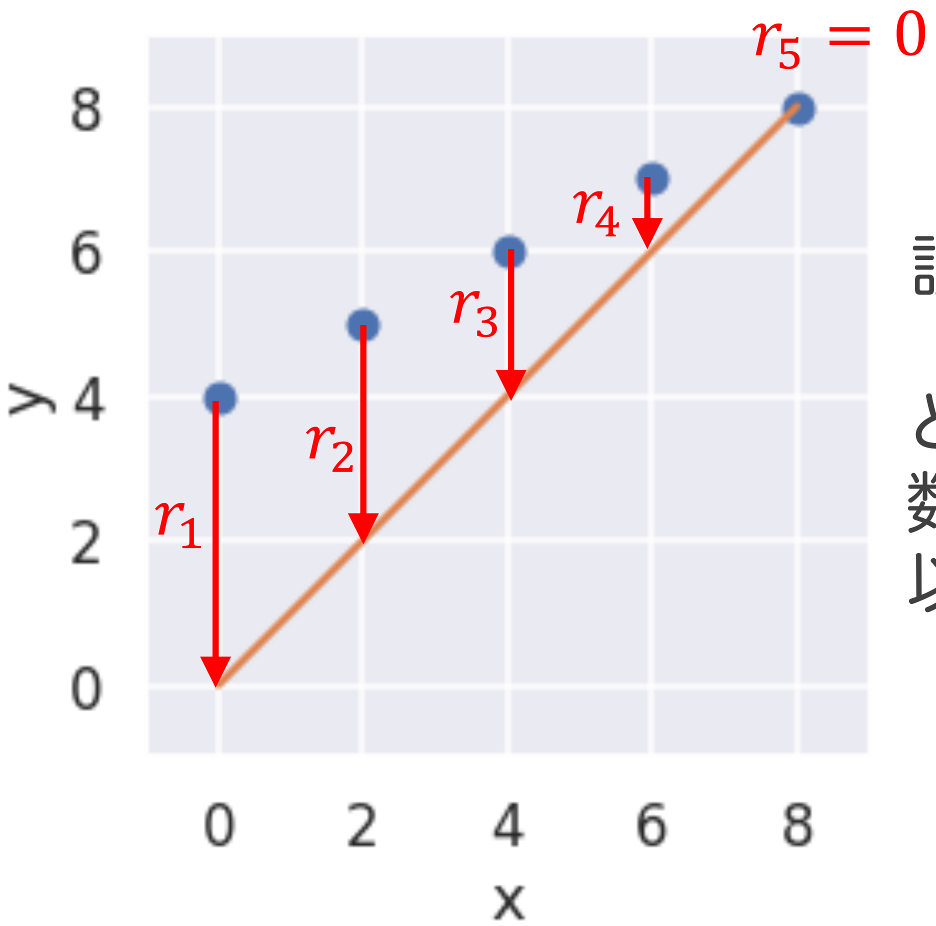


直線でフィットしよう
とする限り
絶対に誤差0にはできない

対策：

・より複雑な式を使う
or
・気にしない

誤差について



$$誤差 = |r_1| + |r_2| + |r_3| + |r_4| + |r_5|$$

としても良いが,
数学的な扱いやすさの点から
以下のようにすることが多い
（二乗誤差と呼ぶ）

$$誤差 = r_1^2 + r_2^2 + r_3^2 + r_4^2 + r_5^2$$

パラメータ値の更新の仕方

- 基本的には以下の方針で値を更新する
  - 値を少しだけ大きくした際，誤差が減るのであれば値を大きくする
  - そうでなければ値を小さくする
- 最初の直線フィッティングの例
  - 初期状態($a = 1$)でaを大きくすると誤差が増えるので，aの値は減少させる
  - 初期状態($b = 1$)でbを大きくすると誤差が減るので，bの値は増加させる

パラメータ値の更新の仕方

- もう少し数学的に述べると，これは
誤差関数をパラメータで偏微分することに相当する
  - そして偏微分の符号と逆の方向に値を変化させる

- 最初の直線フィッティングの更新例（勾配法）
  - aのt回目の更新での値をa_tとするとa_{t+1}は以下のように計算される

$$a_{t+1} \leftarrow a_t - \alpha \cdot \left. \frac{\partial E}{\partial a} \right|_{a=a_t}$$

  - Eは誤差関数，　αは学習率と呼ばれユーザが事前に決める値

教師あり学習まとめ

- 学習は誤差最小化問題として定式化できる
  - 誤差として二乗誤差がよく使われる
  - 誤差はできるだけ小さくしたいが0にならなくても良い
- 最小化問題は解析的に解けることもあるが
（つまり式変形していって最適解を導けることもある）
多くの手法では誤差が小さくなるように
パラメータ値を繰り返し修正する
  - 実は直線でのフィッティングは解析的に解ける

線形回帰

- あるモデルを用いてxとyの関係を求めることを回帰と呼ぶ（yは連続データ）
- 今回の例のように，xとyの関係を一次式でフィッティングすることを線形回帰と呼ぶ
  - x：説明変数
  - y：目的変数 と呼ぶ
- 説明変数は複数でも良い
  - 一般には$y = w_1x_1 + w_2x_2 + \cdots + w_px_p + w_0$の形になる
  - 説明変数が一つの時の回帰を単回帰と呼ぶ

分類

- サンプルごとにそのサンプルが属するクラスを予測するタスクを分類と呼ぶ
- xとyの関係を求める際,
  - yが連続値なら回帰
  - yが有限集合の元なら分類
- 例
  - 各種情報から試験の点数を実数値で予測 → 回帰
  - 各種情報から試験の合否を予測 → 分類

2クラス分類

- 分類されるクラスが二つの時の分類を2クラス分類と呼ぶ．本講義ではこれを扱う

$f(x)$？

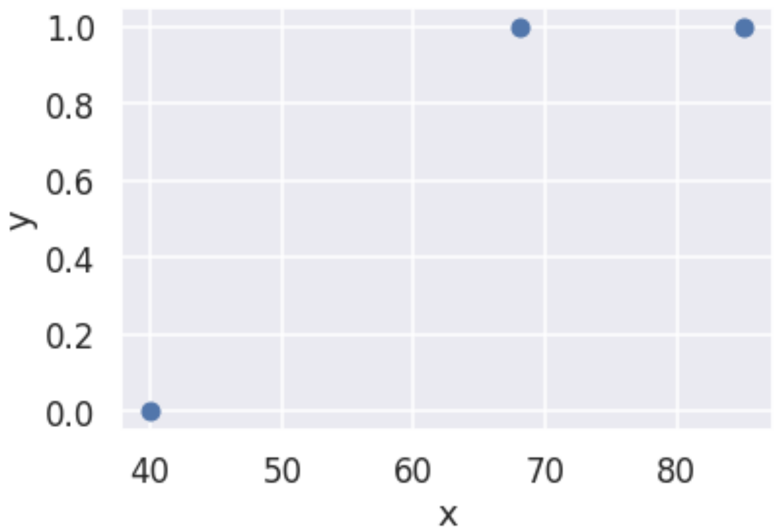
試験の点数 x	y 合否
40	不合格
61	合格
85	合格

ロジスティック回帰

- 2クラス分類に使用される手法の一つ
- 2クラスの片方を0，もう片方を1として
  シグモイド関数で回帰を行う

$\boxed{f(x)\text{？}}$

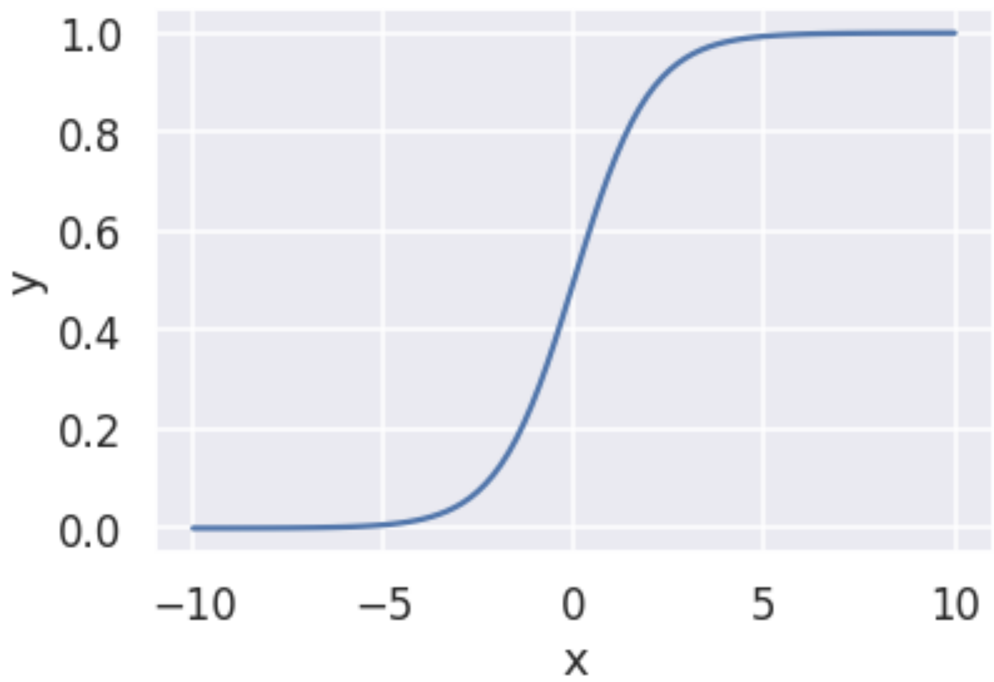
試験の点数	x	y	合否
	40		不合格(0)
	68		合格(1)
	85		合格(1)



シグモイド関数

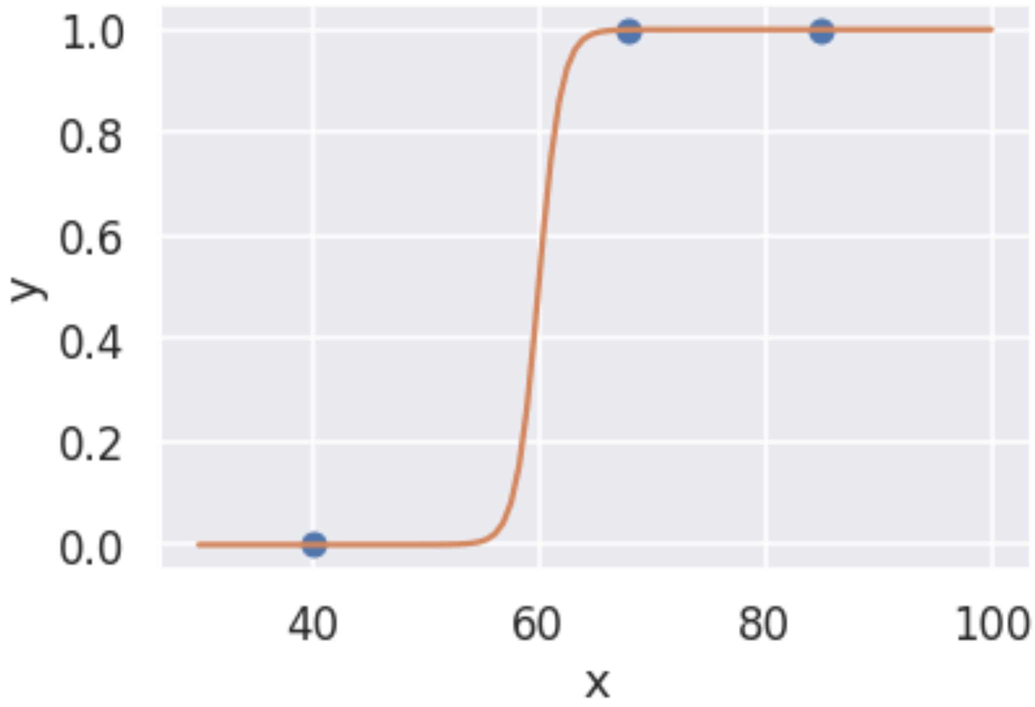
- こんな形の関数
  - $\sigma(x) = \frac{1}{1+e^{-a(x-b)}}$

aは中心部の傾きの急峻さ，
bは中心の位置



シグモイド関数によるフィッティング

- こんな感じでデータに当てはめる
  - 関数値が0.5以上の時，1と判定する
  - 関数値は「出力が１となる確率」と解釈できる



ロジスティック回帰まとめ

- 分類のための手法
- 閾値によりクラスが変わるような単純な場合の分類には有効だが，複雑な条件による分類はできない
- より複雑な条件を扱える分類手法として以下がある
  - サポートベクターマシン
  - ランダムフォレスト（決定木と呼ばれる手法の一つ）