



# AI総合演習

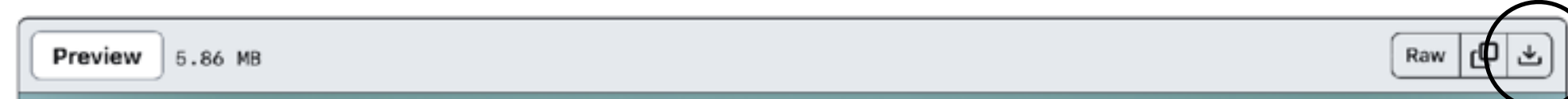
第3回: 機械学習の背景・数理

瓜生真也 (デザイン型AI教育研究センター・助教)

# 講義内容

講義に関する資料（スライド、補足資料等）を  GitHubに置いておきます

<https://github.com/uribo/exeai> ダウンロード可能



1. ガイダンス
2. プログラミング入門
3. 機械学習の背景・数理
4. 機械学習モデルの設計と評価
5. 機械学習の手法
6. 機械学習モデルの解釈・説明性
7. 演習1：プログラミング言語による機械学習モデルの実装
8. 深層学習の基礎
9. 実社会での応用：自然言語処理、推薦
10. 深層生成モデル
11. 演習2：プログラミング言語による深層学習の実装
12. 課題解決型演習1
13. 課題解決型演習2
14. 課題解決型演習3
15. 課題解決型演習の発表と振り返り

今日の目標

機械学習で用いる

数学の基礎をPythonで学ぶ

# 【課題】 講義中に扱う.ipynbを提出

提出期限: 来週の講義開始前まで

manabaのレポートとして提出してください

**注意: ファイル名は英数字のみにすること**

日本語（漢字、片仮名、平仮名）、全角英数字、スペース、記号等は使わない

**ファイルをダウンロードしても開けなくても問題ない（気にしない）**

# 機械学習における数学の必要性

# 人工知能プログラミングに数学は必要か

→この講義や「数理・データサイエンス・AI教育」においては、数学は必要

ライブラリ、フレームワークの利用を利用することで、  
数学の知識がなくても人工知能プログラミングは可能→数学は必要ない

☐ スマートフォン… 動作する仕組みを理解しなくても利用できる

多く的人是ブラックボックス（中身がわからない）状態

情報科学、プログラミングの知識があることでスマートフォンへの理解が深まる

## 機械学習・深層学習

背景・仕組みの理解を深める

応用、開発、研究を行う

→**数学の知識が必要**

# 機械学習で使われる主な数学

## 線形代数

ベクトルや行列、テンソルによってデータの構造を表現、演算を可能にする  
多次元のデータを処理するために多次元の変数により定義された関数を用いる

## 微分・積分

勾配降下法（損失関数の最小とするような最適なモデルのパラメータを見つけるアルゴリズム）において微分を用いる。具体的には損失関数の勾配（傾き）を計算し、勾配が最小（または最大）となる点に進むことで、最適なパラメータを求める

機械学習のさまざまなアルゴリズム（正則化、カーネル法、動的計画法など）で微分・積分が利用される

## 確率・統計

関数のパラメータ（重み）を最適化する際に確率論的な手法を用いる

データの分布、特性を理解するために基本的な統計量（平均、分散、相関など）を用いる

データの特徴に応じた前処理を適用することが可能となる

大学で学ぶ内容もたくさん。まずは中学・高校で学んだ内容を復習しよう




# おさらい: 機械学習における教師あり学習

例) 野菜の価格を予測するモデル

(参照) 第一回の講義

知りたい情報

野菜の価格 

利用可能な情報

野菜の産地

気温

降水量など 

利用可能な情報 (入力  $x$ ) をもとに野菜の価格 (出力  $y$ ) を予測する

$$y = f(x)$$

のようなモデル式を考える → 学習

## 学習データ

日付	品目	産地	気温(°C)	日降水量(mm)	価格 1kgあたり
4/30	トマト	熊本県	22.4	0	365
4/30	キュウリ	群馬県	19.5	20	320

⋮

10/8	カボチャ	北海道	8.6		192
------	------	-----	-----	--	-----

→ 予測の対象となる価格と  
その他のデータの関係から  
価格を予測するモデルを構築する

## モデルの予測

日付	品目	産地	気温	日降水量 (mm)	価格予測値 (モデルA)	価格予測値 (モデルB)	価格予測値 (実際の値)
4/20	トマト	鹿児島県	21.8	14	326	362	357
4/30	キュウリ	福島県	19.5	3	283	256	250

→ モデルAとモデルBどちらが優れたモデル?

$$price = temperature_{x1} + b$$

$$price = temperature_{x1} + precipitation_{x2} + \epsilon$$



# 回帰直線と回帰係数

2つの変数  $(x, y)$  の関係を説明する回帰直線  $y = ax + b$  (一次関数) を考える

## 回帰係数

$a$  切片 ( $x$  が0のときの $y$  の値)

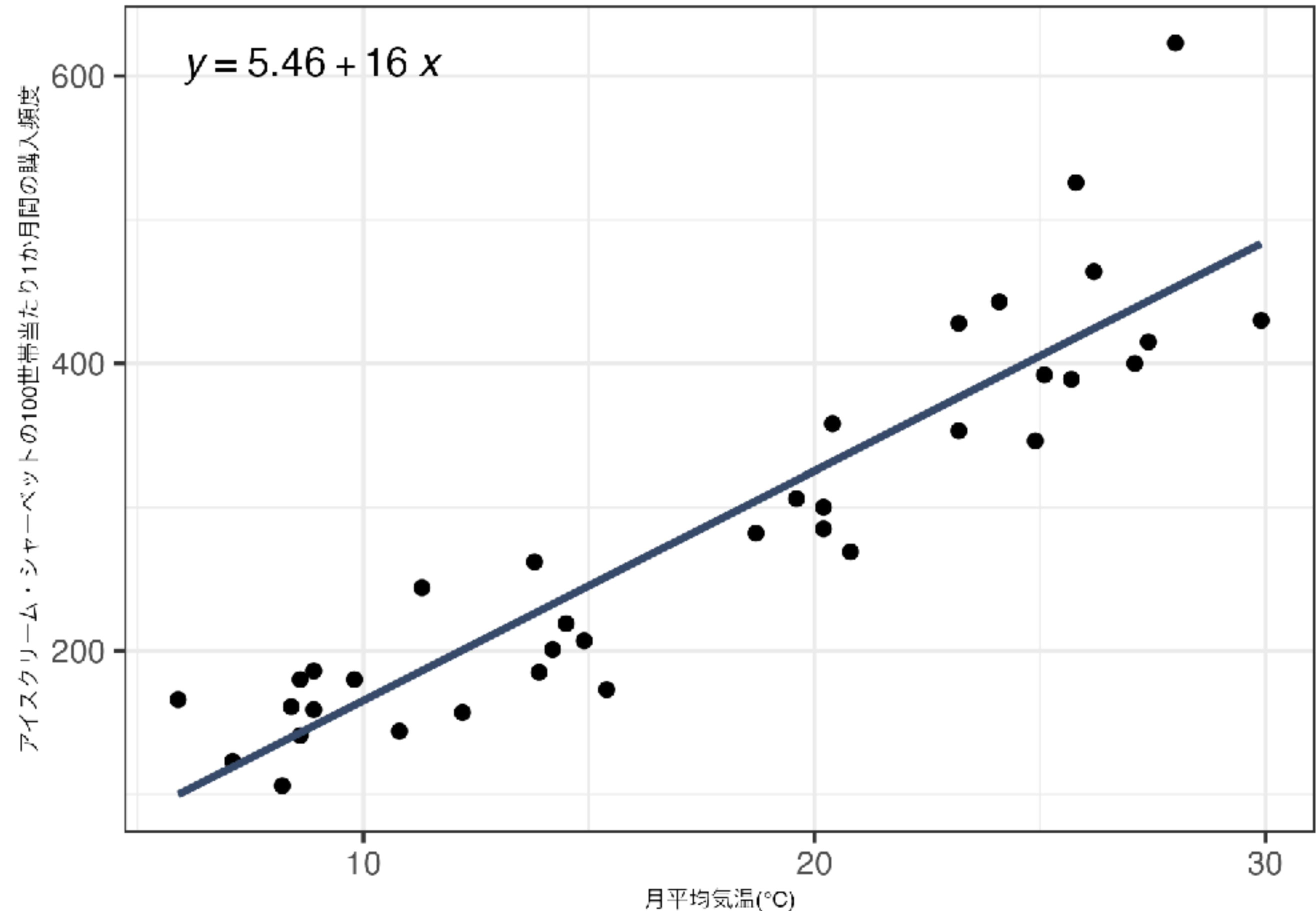
$b$  傾き

$a$   $b$  モデルの係数、重みまたはパラメータと呼ぶ

$$a = \beta_0$$

$$b = \beta_1$$

気温とアイスクリーム・シャーベットの売上の関係



データ: 気象庁 過去の気象データおよび  
家計調査 表番号4-1

1世帯当たり1か月間の支出金額、購入数量及び平均価格 都市階級・地方・都道府県庁所在地別

# 回帰直線

「切片aと傾きbからなる回帰直線上のxの値によってyの値が決まる」と考える

平均気温が高くなればアイスの売り上げが上がる？

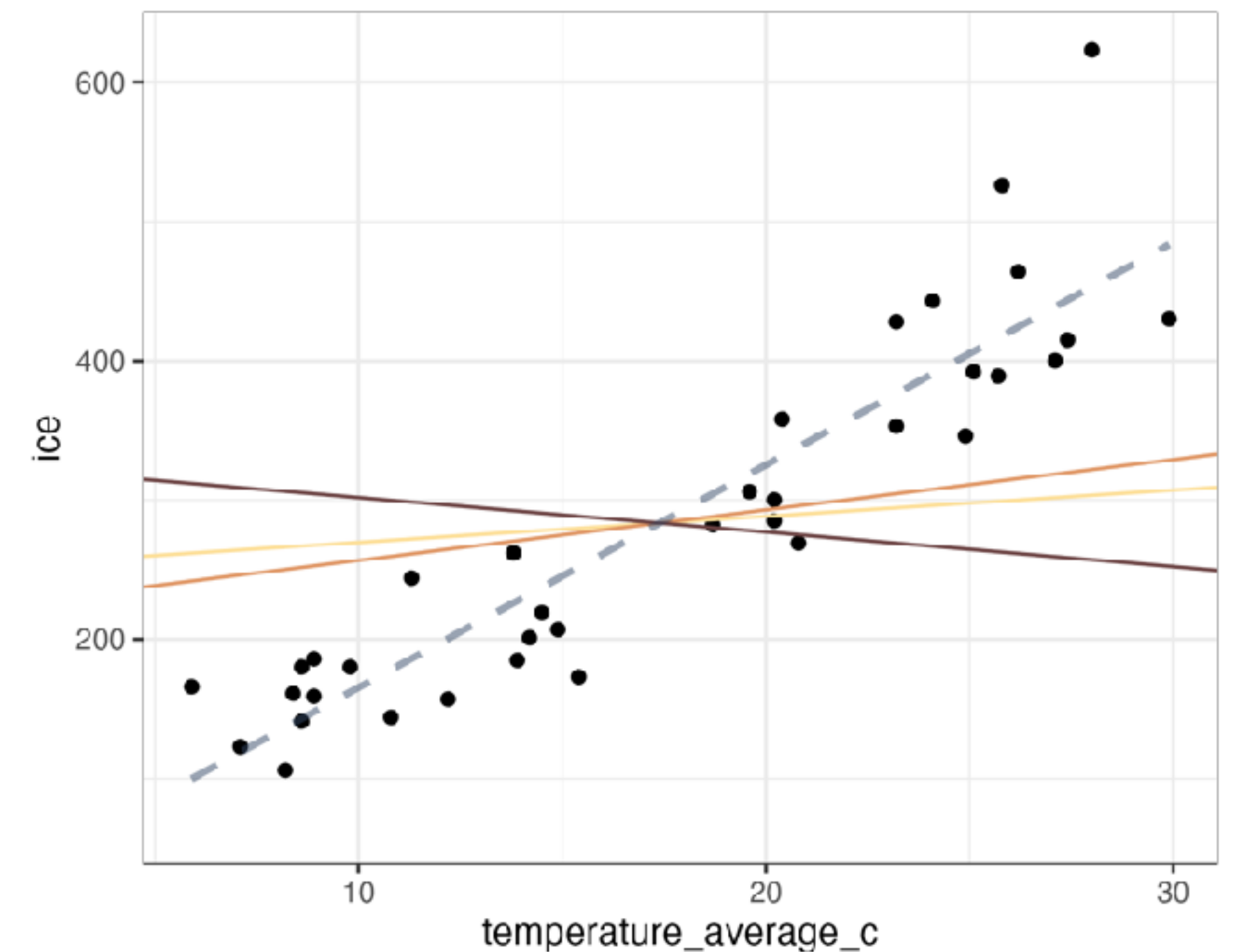
気温が1℃上がればアイスの購入頻度が16増える

回帰直線の係数と気温からアイスの売り上げを予測する

アイスの売り上げ =  $5.46 + 16.0 \times 30$  (気温)

切片と傾きが異なれば回帰直線の形も異なる

2つの変数の関係を最も説明できる回帰直線とは？



さまざまな回帰係数からなる回帰直線

# 最小二乗法

第四回で扱います

回帰直線から残差 (residual, 観測値 $y$  から予測値 $\hat{y}$  のズレ… 誤差) を求める

予測値  $\hat{y}$  は回帰直線の係数から推定する

$$\hat{y}_i = 5.46 + 16x_i$$

$$x_1 = 7.1$$

$$\hat{y}_1 = 5.46 + 16 \times 7.1$$

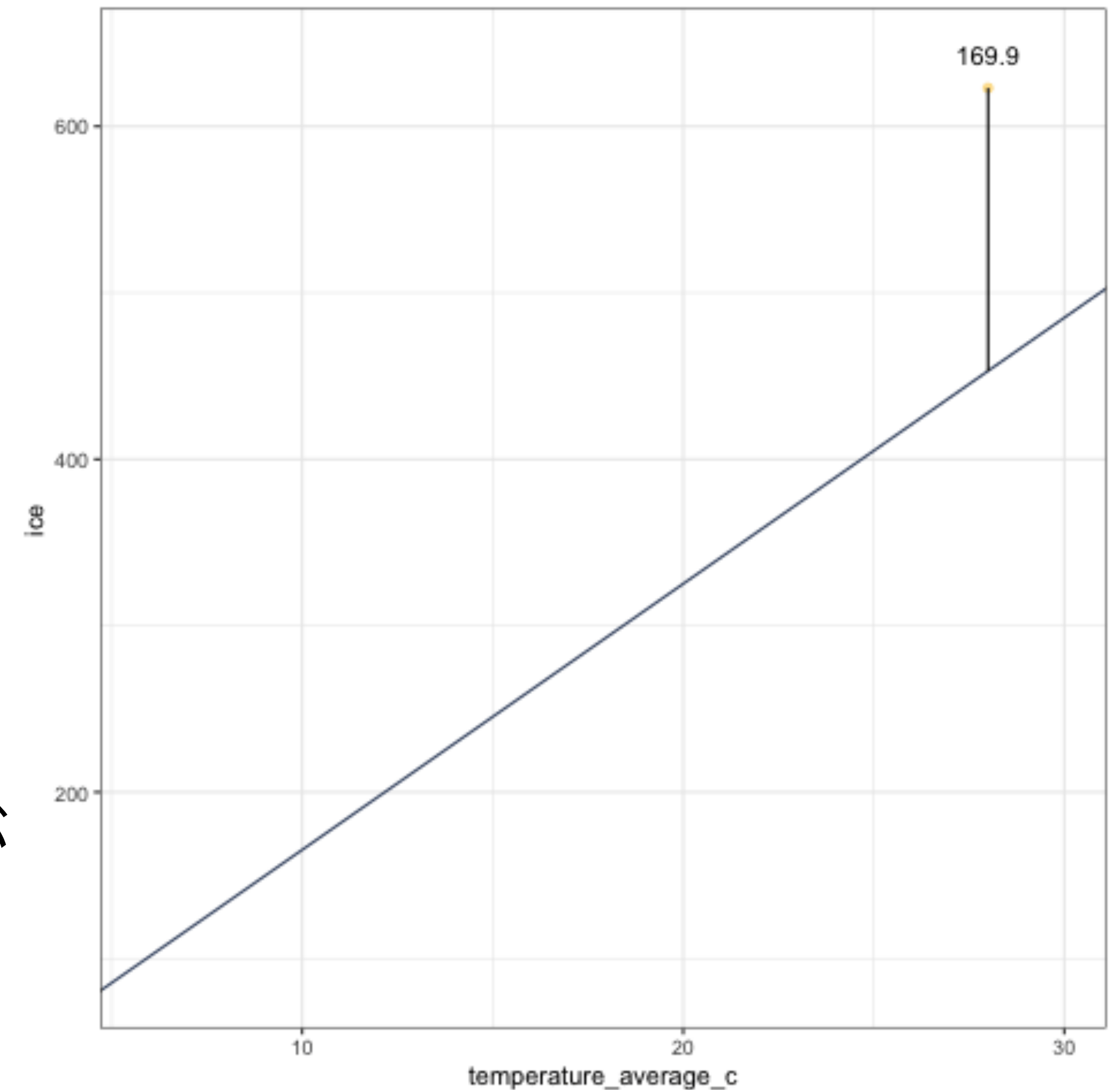
$$\hat{y}_1 = 118.9$$

$$residual_1 = y_1 - \hat{y}_i$$

$$y_1 = 123$$

$$123 - 118.9$$

残差平方和 (各残差を二乗した結果を合計する) が  
**最小となる定数項** (傾きと切片) を求める



残差平方和を最小にすることにより当てはめた回帰直線と気温。縦の棒が残差を示す。

# 中学・高校で学んだ数学の復習

# 関数（函数）：入力と出力の関係性を表す数式

$$x = f(x) = y$$

入力                      変換                      出力

入力  $x$  が決まると、  
通常は出力  $y$  も一つの値に決まる

🍅 トマト（大）1玉… 100円（税抜き）消費税率10%

3個買うと…

5個買うと…

$$y \text{ (支払う金額)} = (100 + (100/10)) \times 3$$

$$y = (100 + (100/10)) * 5$$

$$f(3) = 330$$

$$f(5) = 550$$

この関数では、トマト（大）が一玉100円

消費税率10% であることは変わらない → 定数

トマトを何個買うかは客によってさまざまな値を取る → 変数

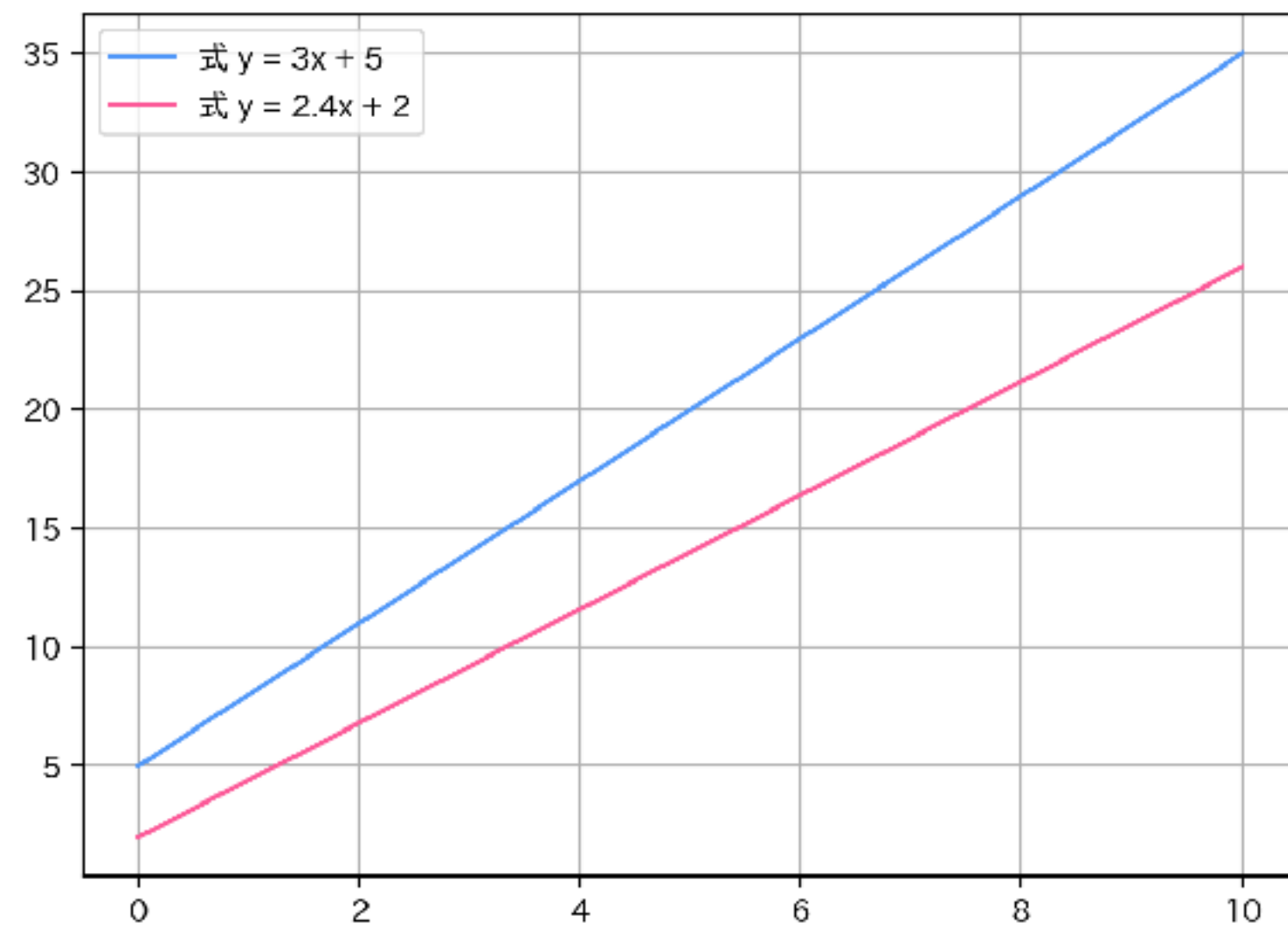
購入するトマトの数が増えると支払う金額も増加する



# 一次関数

$$y = ax + b$$

$a$  (傾き) と  $b$  (切片、 $x=0$ のときの $y$ の値) は定数  
 $x$  変数



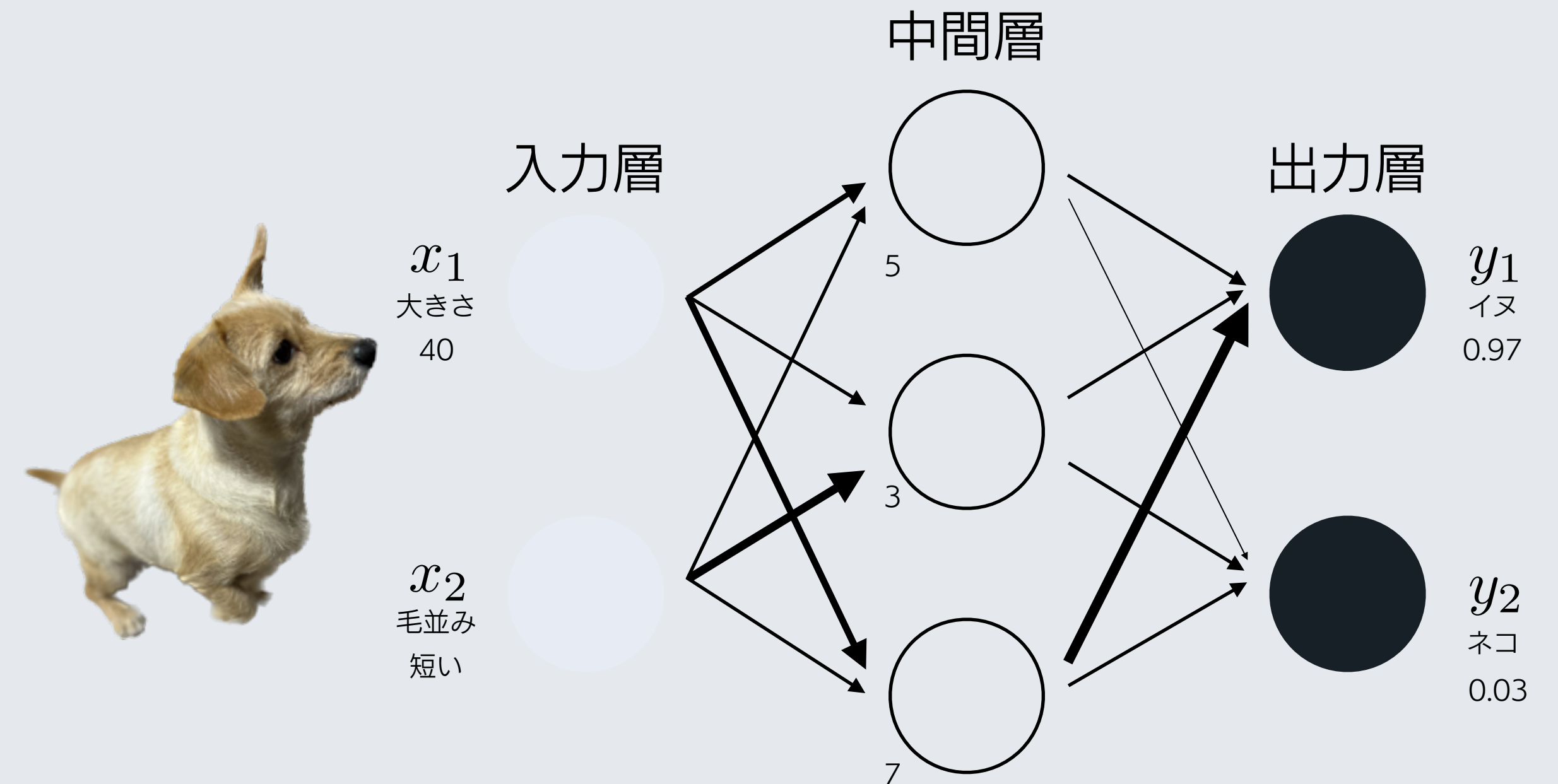
一次関数のグラフは直線

定数が異なると変数が  
同じ値であっても出力は変わる

機械学習・深層学習では

コンピュータが変数としての「重み」を学習する過程

学習モデルを利用するときは、定数として「重み」が用いられる



# Pythonで数学を扱うライブラリ

## math

標準ライブラリ

基本的な数学関数や定数、演算機能を提供

## NumPy

数値計算、多次元配列を扱うライブラリ

C言語での実装→高速な演算

Pythonでの機械学習や深層学習のためのライブラリで広く使われる

互換性やデータのやり取りが容易

線形代数や統計処理のための機能も備える (mathより広範囲)

<https://numpy.org/>

pipコマンドでのインストール

```
pip install numpy
```

Python内での呼び出し

```
import numpy as np
```

それぞれ異なる方法 (同じものもあり) で利用する

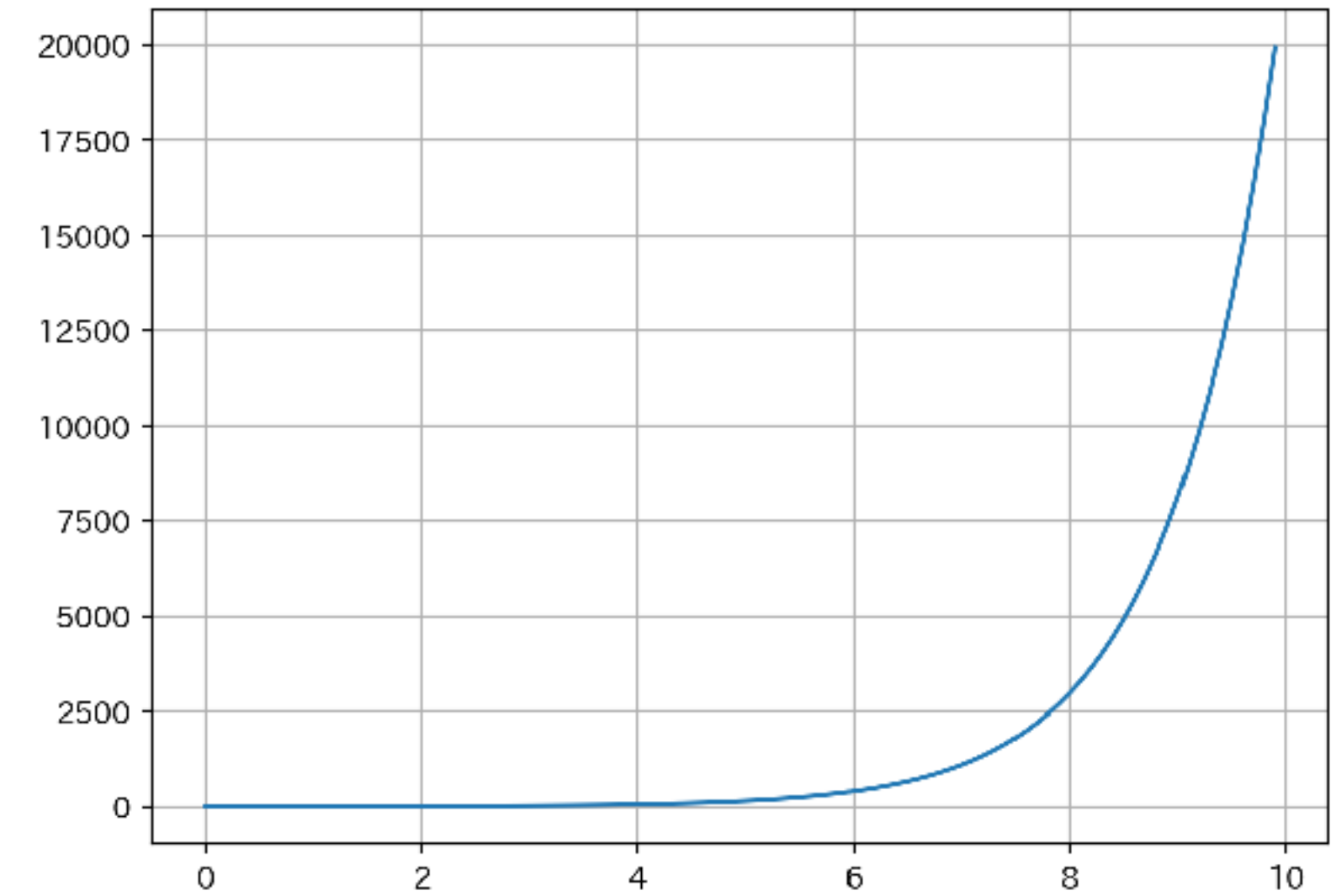


# 指数関数

指数を変数とした関数  $f(x) = a^x$   $a > 0, a \neq 1$

**特徴**  $x$ の値が大きいくほど、出力の値も大きくなる (単調増加)

出力は常に正の値



指数関数のグラフ

指数… ある数を何乗したかを表す数

## ネイピア数、自然対数の底( $e$ )

$$e = \lim_{n \rightarrow \infty} \left( 1 + \frac{1}{n} \right)^n \quad \lim_{n \rightarrow \infty} n \text{ を無限大に近づける}$$

# 対数関数

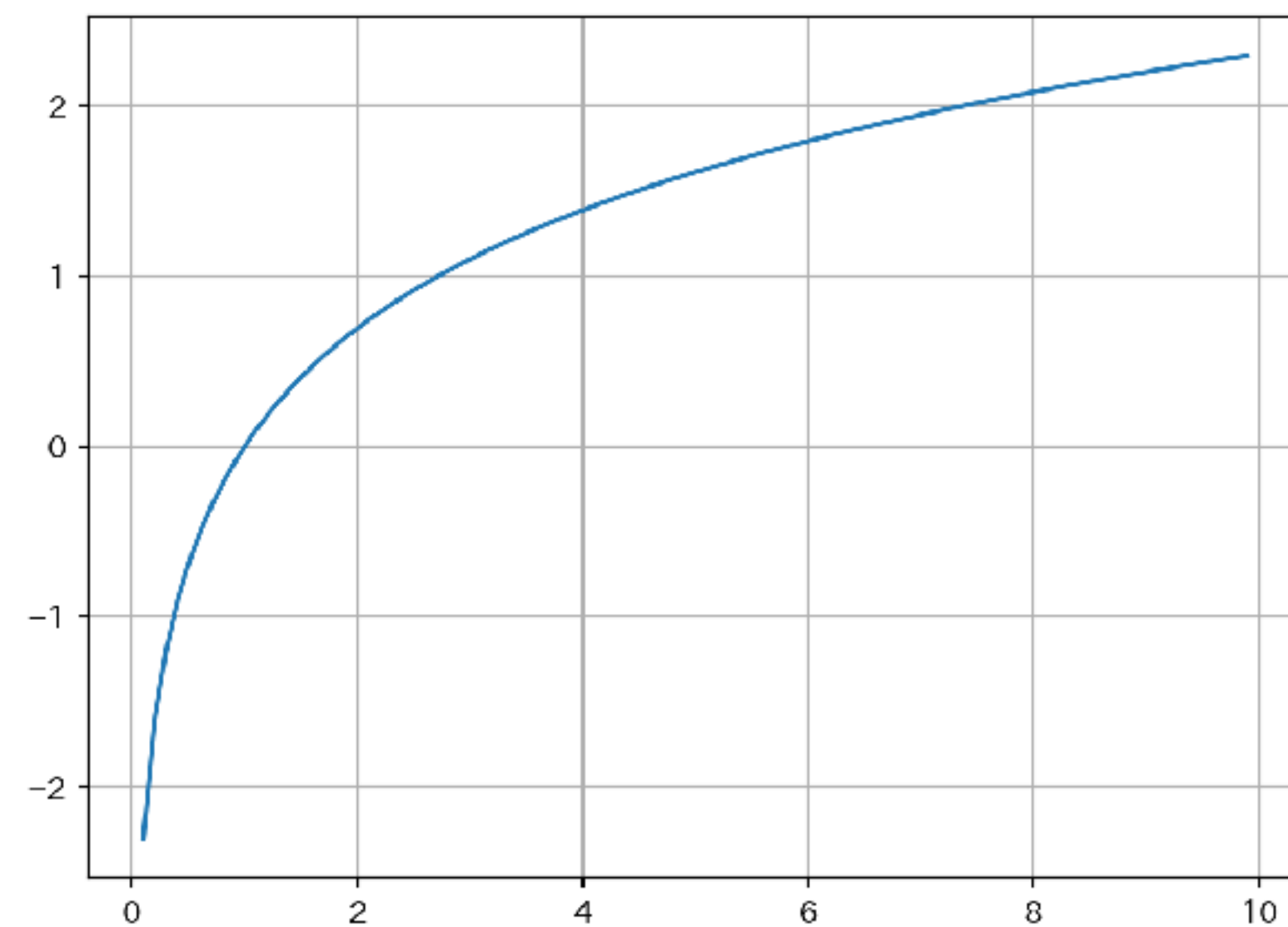
指数関数の逆関数（指数関数の入力に対数関数の出力）  $f(x) = \log_a x$

**特徴**  $x$ の値が大きいくほど、出力の値は大きくなる（単調増加）。  
 $x$ の値が大きくなるほど増加速度は減少する

指数関数の出力が正の値になるため、入力は正の値のみ

底（定数）には自然対数の底  $e = 2.718281828\dots$   
が使われる（ $e \doteq 2.7$ ）

底が 10 の対数関数は常用対数



対数関数のグラフ

ここまでのまとめ

# これまでのまとめ

数学の理解によって人工知能を扱うプログラミングの背景を明らかとなる

特にベクトル、行列を対象にする線形代数や、最適化アルゴリズムのための確率・統計および微積分は重要  
良いモデル→真の値に対して予測とのズレが少ない、最適なパラメータを見つける問題

## Pythonによる中学・高校で学んだ数学の復習

数学用途のライブラリとして、標準ライブラリのmathとサードパーティーライブラリのNumPyがある

機械学習で頻繁に利用する累乗、平方根、指数関数、対数関数についてPythonでの利用方法を学んだ

# 線形代数の理解に向けて

# ベクトル

2つ以上の数値からなる一次元の数値の組

1つのベクトルを構成する個々の数をベクトルの要素あるいは成分と呼ぶ

1つの成分しかもたないものは「スカラー」として区別する

要素の表記法による分類

**列（または縦）ベクトル**

$$\mathbf{a} = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_n \end{pmatrix}$$

**行（または横）ベクトル**

$$\mathbf{a} = (a_1, a_2, a_3, \dots, a_n)$$

# 行列

複雑な計算もシンプルに表現可能

ベクトルの概念を拡張

$m \times n$  行列

$$\mathbf{A} = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$$

転置 縦と横を入れ替える

$$\mathbf{A}^T = \begin{pmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{pmatrix}$$

対角成分 (1,5,9) は変わらず、  
非対角成分 (2と4、3と7、6と8) が入れ替わる

行列にTの添え字を付ける



# 参考資料・URL

- 石川聡彦 『人工知能プログラミングのための数学がわかる本』

(2018) KADOKAWA. ISBN: 978-4-04-602196-0

瓜生居室: あり (電子版)、徳大図書館: あり、市立図書館: なし、県立図書館: なし

- 椎名洋, 姫野哲人, 保科架風(著), 清水昌平(編) 『データサイエンスのための数学』

(2019) 講談社. ISBN: 978-4-06-516998-8

瓜生居室: あり (電子版)、徳大図書館: なし、市立図書館: なし、県立図書館: なし

- 八谷大岳 『ゼロからつくるPython機械学習プログラミング入門』

(2020) 講談社. ISBN: 978-4-06-520612-6

瓜生居室: あり (電子版)、徳大図書館: なし、市立図書館: なし、県立図書館: なし

- 吉田拓真, 尾原颯 『現場で使える!NumPyデータ処理入門: 機械学習・データサイエンスで役立つ

高速処理手法』 (2018) 翔泳社. ISBN: 978-4-7981-5591-3

瓜生居室: あり (電子版)、徳大図書館: なし、市立図書館: なし、県立図書館: なし

