データ分析の基礎：
考え方とPythonによる実習

奥村晴彦（okumura@okumuralab.org）

情報I (4) 情報通信ネットワークとデータの活用

　情報通信ネットワークを介して流通するデータに着目し，情報通信ネットワークや情報システムにより提供されるサービスを活用し，問題を発見・解決する活動を通して，次の事項を身に付けることができるよう指導する。

ア　次のような知識及び技能を身に付けること。

(ｱ) 情報通信ネットワークの仕組みや構成要素，プロトコルの役割及び情報セキュリティを確保するための方法や技術について理解すること。

(ｲ) **データを蓄積，管理，提供する方法，**情報通信ネットワークを介して情報システムがサービスを提供する仕組みと特徴について理解すること。

(ｳ) **データを表現，蓄積するための表し方と，データを収集，整理，分析する方法について理解し技能を身に付けること。**

イ　次のような思考力，判断力，表現力等を身に付けること。

(ｱ) 目的や状況に応じて，情報通信ネットワークにおける必要な構成要素を選択するとともに，情報セキュリティを確保する方法について考えること。

(ｲ) 情報システムが提供するサービスの効果的な活用について考えること。

(ｳ) **データの収集，整理，分析及び結果の表現の方法を適切に選択し，実行し，評価し改善すること。**

（参考）情報II (3) 情報とデータサイエンス

　多様かつ大量のデータを活用することの有用性に着目し，データサイエンスの手法によりデータを分析し，その結果を読み取り解釈する活動を通して，次の事項を身に付けることができるよう指導する。

ア　次のような知識及び技能を身に付けること。

(ｱ) 多様かつ大量のデータの存在やデータ活用の有用性，データサイエンスが社会に果たす役割について理解し，目的に応じた適切なデータの収集や整理，整形について理解し技能を身に付けること。

(ｲ) データに基づく現象のモデル化やデータの処理を行い解釈・表現する方法について理解し技能を身に付けること。

(ｳ) データ処理の結果を基にモデルを評価することの意義とその方法について理解し技能を身に付けること。

イ　次のような思考力，判断力，表現力等を身に付けること。

(ｱ) 目的に応じて，適切なデータを収集し，整理し，整形すること。

(ｲ) 将来の現象を予測したり，複数の現象間の関連を明らかにしたりするために，適切なモデル化や処理，解釈・表現を行うこと。

(ｳ) モデルやデータ処理の結果を評価し，モデル化や処理，解釈・表現の方法を改善すること。

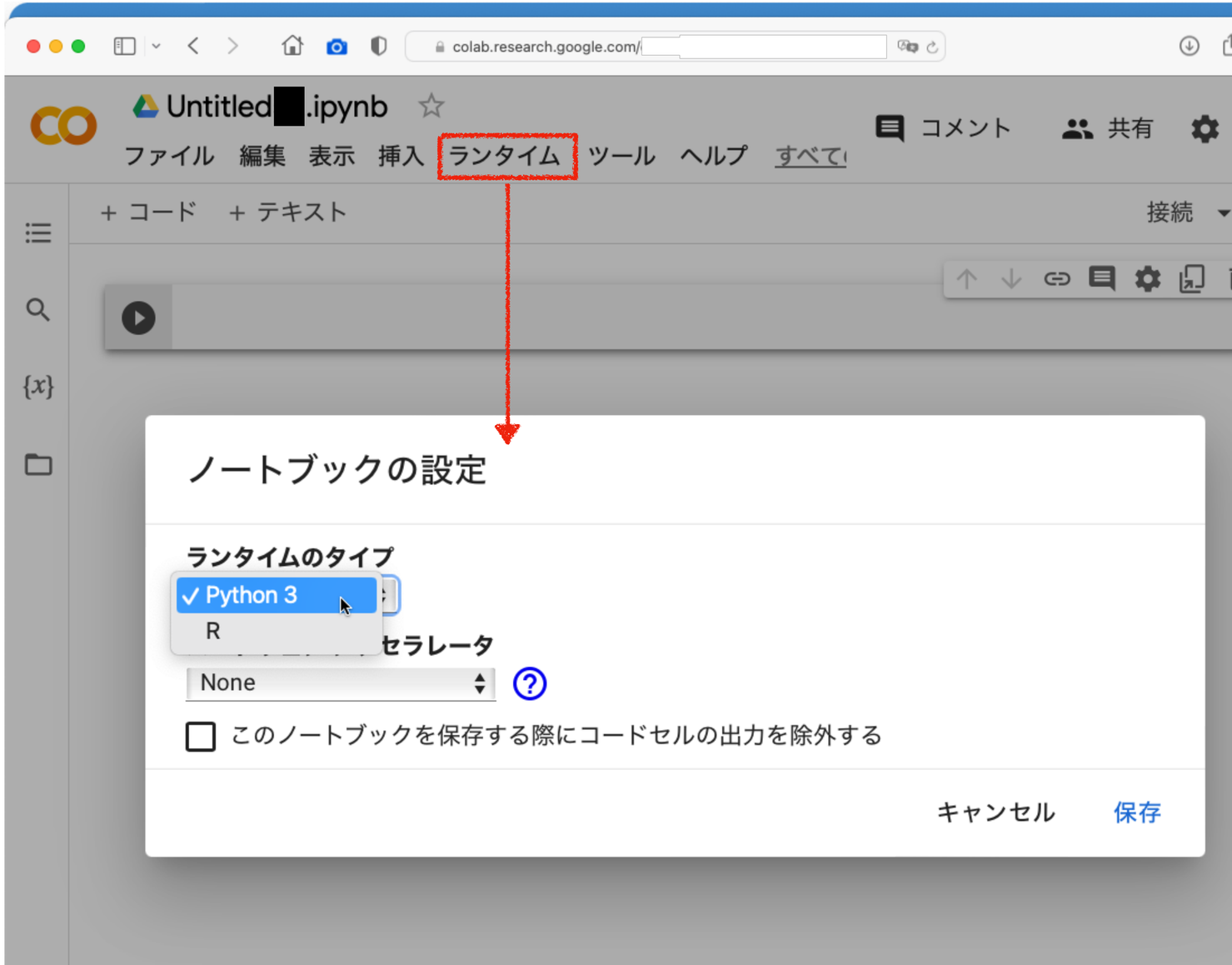
何を使う？

Excel　Python　R

Google Colaboratory

https://colab.research.google.com/

https://okumuralab.org/~okumura/python/colab.html



Pythonコードを綺麗に！（PEP 8)

https://pep8-ja.readthedocs.io/ja/latest/

原則：インデントはスペース4個
　　演算子の前後・コンマの後にスペースを入れる

```
s = 0
for x in [1, 1, 2, 3, 5, 8]:
    s = s + x
print(s)
```

ただし、関数の引数の = の前後はスペース不要

```
df = pd.read_csv(URL, encoding="cp932", comment="#")
```

オープンデータを活用しよう

自由に加工し公開できるデータ
オープンデータの典型的なライセンス：
　　政府標準利用規約（第2.0版）またはCreative Commonsの 
例：e-Stat　https://www.e-stat.go.jp
　　気象庁 過去の気象データ・ダウンロード　https://www.data.jma.go.jp/gmd/risk/obsdl/
　　自治体のデータ

オープンデータの形式：CSV・JSON推奨（Excel形式もまだ多い）
セル結合・罫線による視覚的レイアウト：データとして使いにくい

データの取得と整理

Pythonなら pandas を使う。
SQLと同様な処理ができる。

https://okumuralab.org/~okumura/python/query.html

Web APIについては以下参照

https://okumuralab.org/~okumura/python/webapi.html

Web APIとJSON

https://okumuralab.org/~okumura/python/webapi.html 参照

例　https://zipcloud.ibsnet.co.jp/api/search?zipcode=5140007


```
{
    "message": null,
    "results": [
        {
            "address1": "三重県",
            "address2": "津市",
            "address3": "大谷町",
            "kana1": "ﾐｴｹﾝ",
            "kana2": "ﾂｼ",
            "kana3": "ｵｵﾀﾆﾁｮｳ",
            "prefcode": "24",
            "zipcode": "5140007"
        }
    ],
    "status": 200
}
```

データ整理についてのよくある誤解

外れ値：例えば2σ以上を外れ値として削除するようなことは絶対ダメ
　東京都の人口は4σ以上
　本当に誤入力があれば欠測値とする（配布元に連絡する）
　外れ値に強い統計量（中央値など）を使うのも良い

欠測値（欠損値）：全体の平均値を代入してはダメ
　欠測値のある行全体を削除するのが一番簡単
　欠測値を推定する方法もあるが、適用が難しい

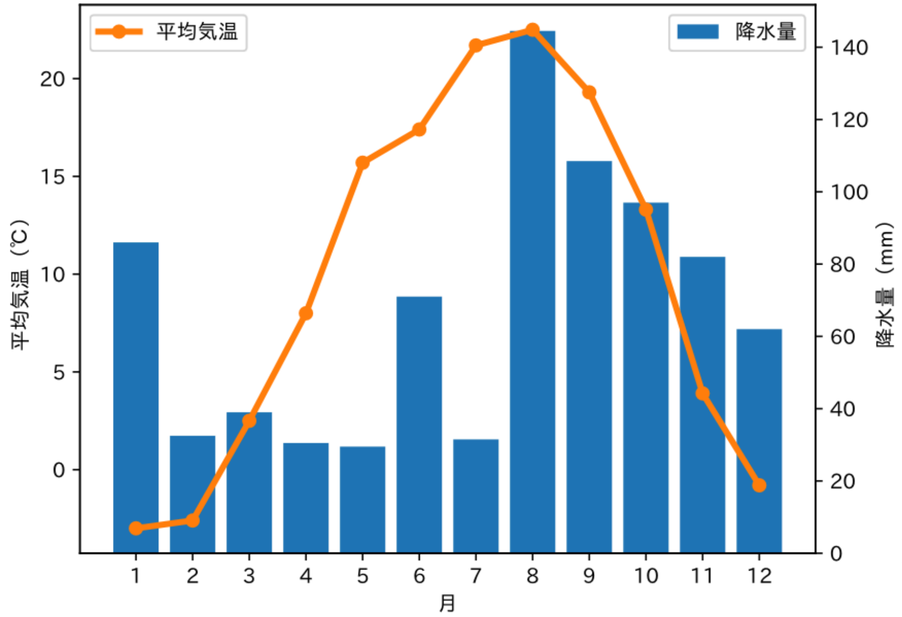
データの尺度とグラフ

質的　名義尺度　0:男 1:女

　　順序尺度　1:反対 2:やや反対 3:やや賛成 4:賛成

　　間隔尺度　セ氏温度　→折れ線グラフ

量的　比例尺度　雨量　　→棒グラフ（0から始める）［折れ線グラフも可］



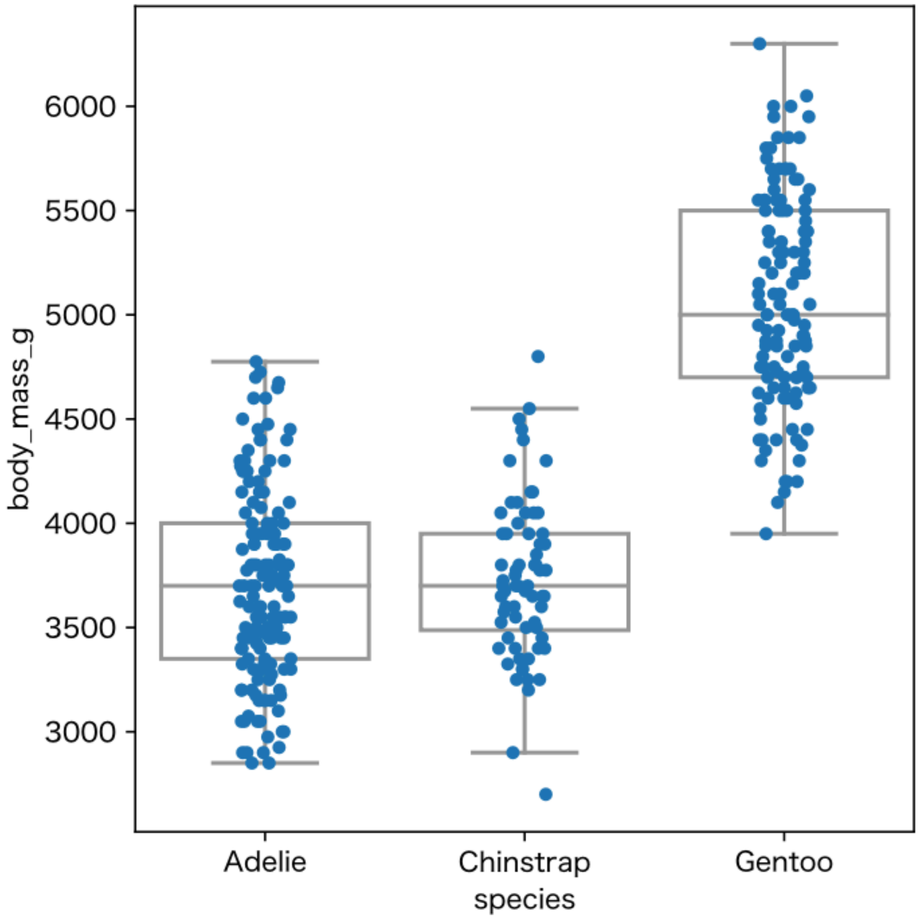
https://okumuralab.org/~okumura/python/2axis.html

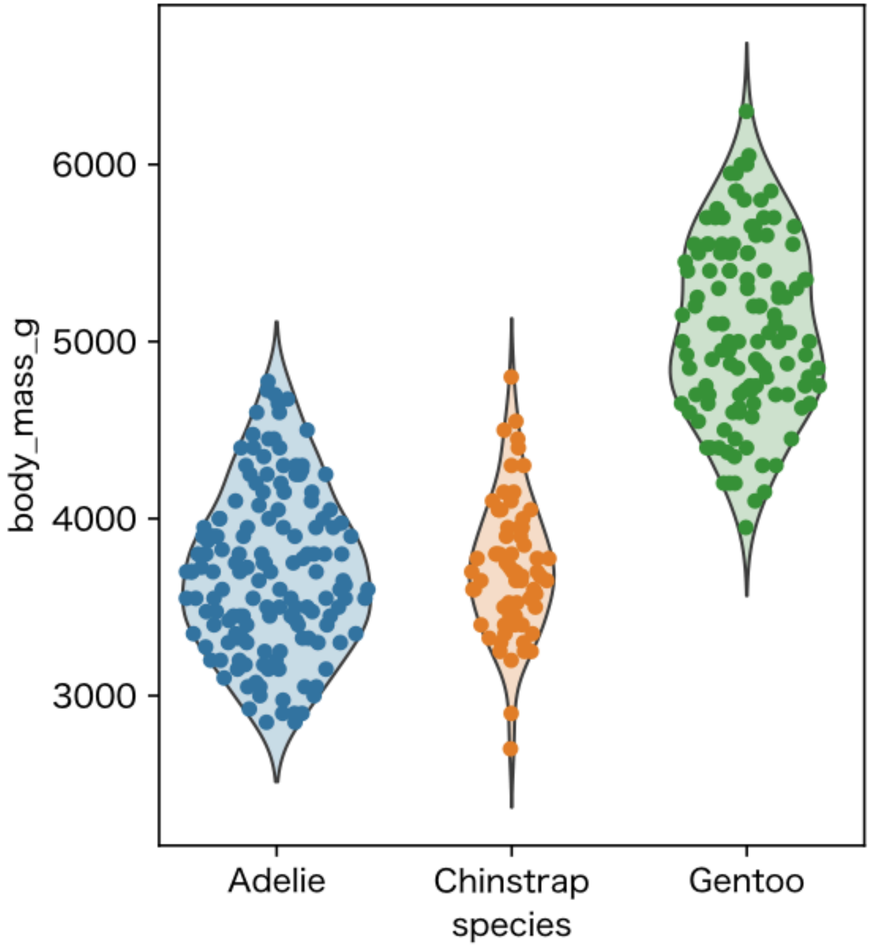
グラフについての注意

3次元グラフ：使わない
円グラフ：複数回答可（合計が100%にならない）は棒グラフに
棒グラフ：比例尺度（比率尺度）のデータに使う。0から始める
折れ線グラフ：
　Excelでは横軸が質的データの場合は「折れ線グラフ」、
　　横軸が量的データの場合は「散布図」（線分・マーカー）
箱ひげ図：必ずしもベストではない（次ページ）
ワードクラウド：素人ウケするけれど・・・

箱ひげ図より情報量のあるグラフ

https://okumuralab.org/~okumura/python/penguins.html



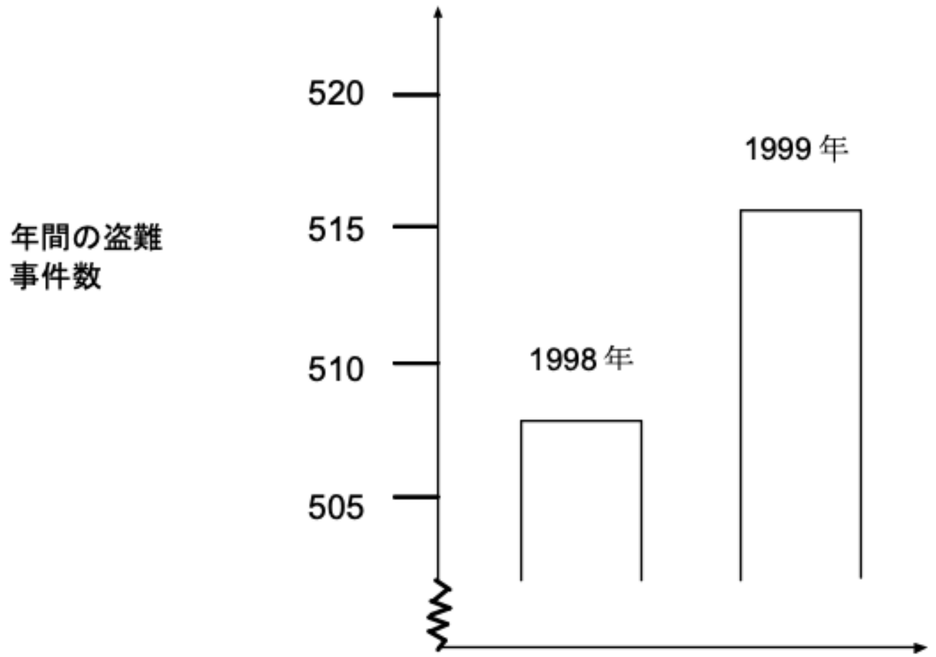


仮説検定、*p* 値

https://okumuralab.org/~okumura/python/pvalue.html

盗難事件

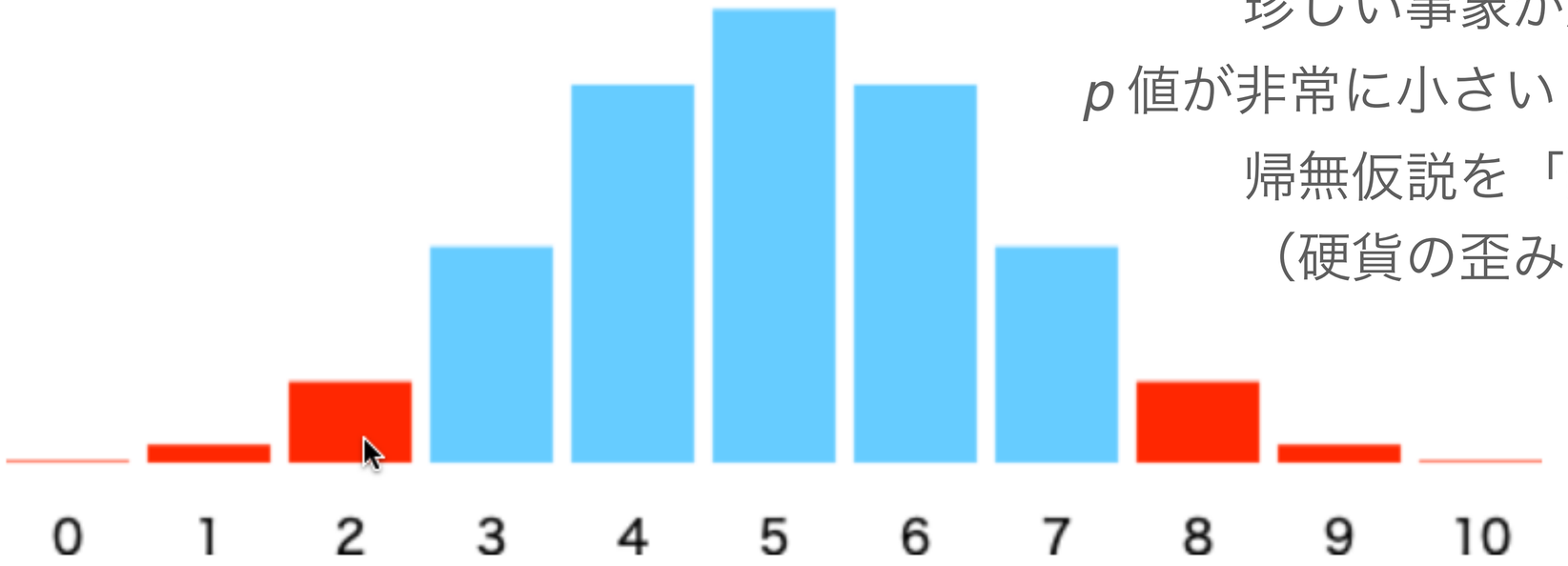
ある TV レポーターがこのグラフを示して、「1999 年は 1998 年に比べて、盗難事件が激増しています」と言いました。



このレポーターの発言は、このグラフの説明として適切ですか。適切である、または適切でない理由を説明してください。

仮説検定、p 値

例：2項分布　${}_nC_r\theta^r(1-\theta)^{n-r}$



表が出る確率 θ = 0.5 の硬貨を10回投げて表が 2 回出ました。$p = 0.109$

https://okumuralab.org/~okumura/stat/binomp.html

帰無仮説 = 基準とする仮説
（硬貨は歪んでいない： $\theta = 0.5$）

p 値 = 帰無仮説のもとで、
その事象またはそれより
珍しい事象が起こる確率

p 値が非常に小さい（例: $p \leqq 0.05$）ならば
帰無仮説を「棄却」する
（硬貨の歪みは統計的に有意である）

p 値の解釈のよくある間違い

p 値は「帰無仮説の起こる確率」？❌

帰無仮説は「表と裏の確率がぴったり
等しい」といった非現実的な仮説であり、
これが正しい確率は考えない
（ある仮説が正しい確率は伝統的統計学では考えない）。

$p > 0.05$ なら帰無仮説は正しいことが証明された？❌

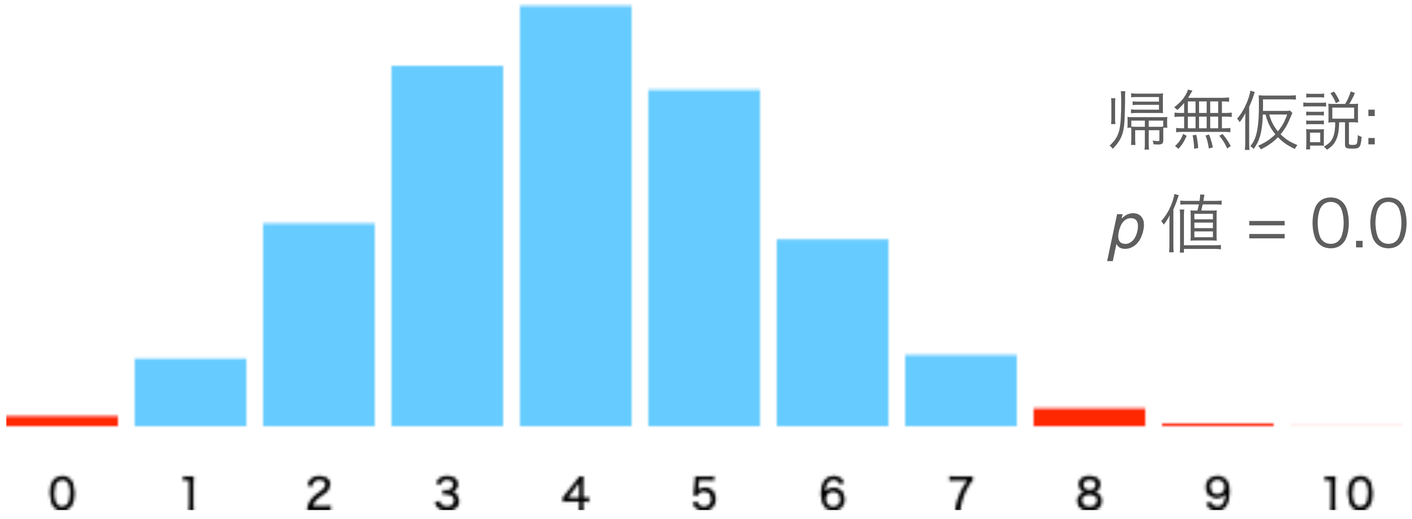
この場合は結論保留が正しい。例えば「この薬は効かない」
という帰無仮説に対して $p > 0.05$ が得られた場合、
「薬の効果はこの実験では確認できなかった」
（効くとも効かないとも言えない）が正しい。

2項検定の例

従来の薬は40%の人に効果がある。
新薬を10人に投与したところ8人に
効果があった。従来の薬より有意に
効果があるといえるか。

帰無仮説: $\theta = 0.4$の2項分布

p 値 $= 0.018$



表が出る確率 θ = 0.4 の硬貨を10回投げて表が 8 回出ました。$p = 0.018$

https://okumuralab.org/~okumura/stat/binomp.html

Pythonで2項検定

```
>>> from scipy import stats

>>> stats.binomtest(2, 10, 0.5)
BinomTestResult(k=2, n=10, alternative='two-sided', statistic=0.2, pvalue=0.109375)

>>> stats.binomtest(2, 10).proportion_ci()
ConfidenceInterval(low=0.025210726326833497, high=0.5560954623076415)
```

p 値

表が出る確率の95%信頼区間

Rで2項検定

```
> binom.test(2, 10, 0.5)

        Exact binomial test

data:  2 and 10
number of successes = 2, number of trials = 10, p-value = 0.1094
alternative hypothesis: true probability of success is not equal to 0.5
95 percent confidence interval:
 0.02521073 0.55609546
sample estimates:
probability of success
                   0.2
```

p 値

表が出る確率の95%信頼区間

Excelで2項検定

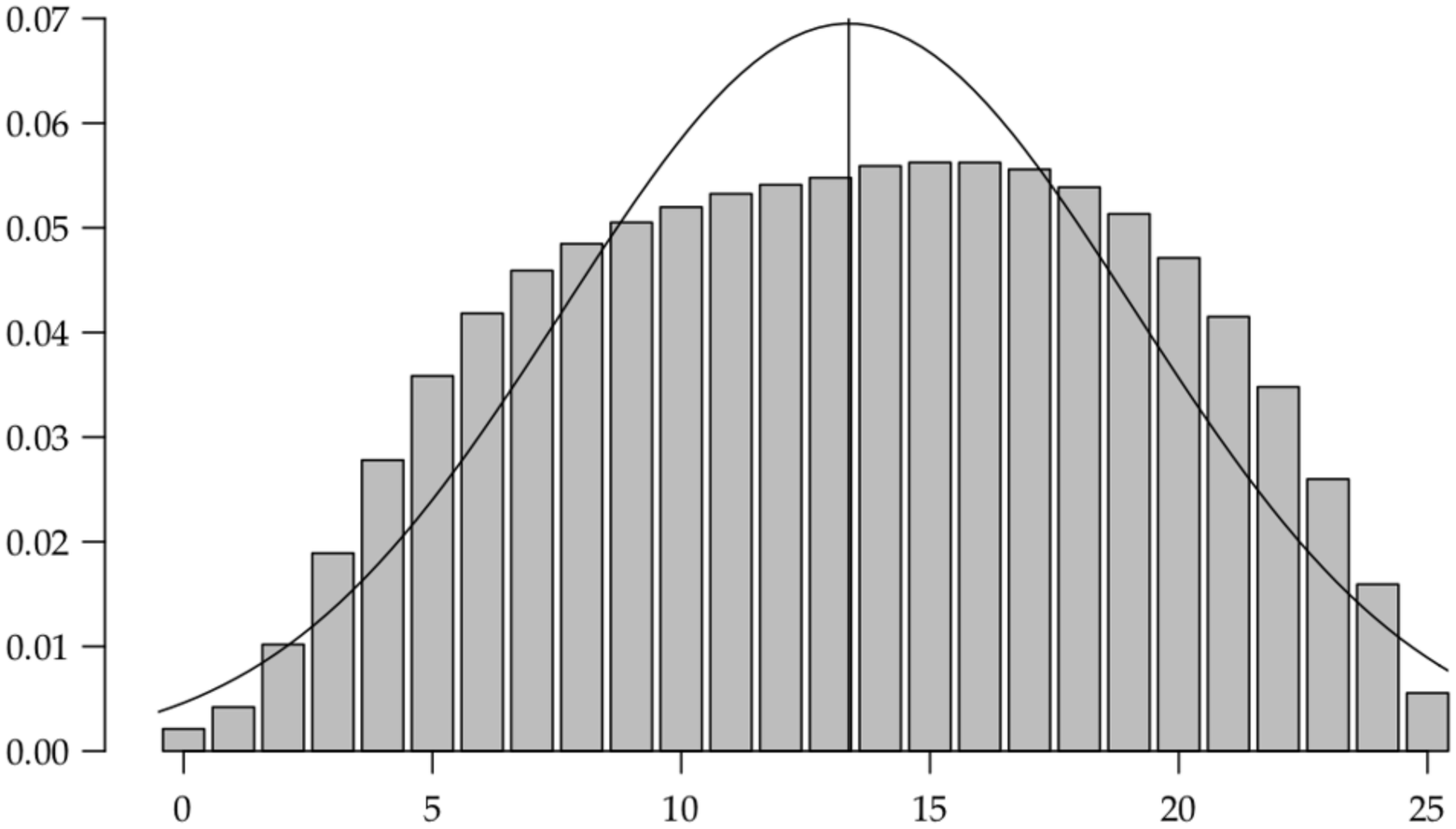
B13 =SUMIF(B2:B12, "<="&B4)

	A	B	C
1	表の枚数	確率	
2	0	0.00097656	
3	1	0.00976563	
4	2	0.04394531	
5	3	0.1171875	
6	4	0.20507813	
7	5	0.24609375	
8	6	0.20507813	
9	7	0.1171875	
10	8	0.04394531	
11	9	0.00976563	
12	10	0.00097656	
13	p値	0.109375	

```
=BINOM.DIST(A2, 10, 0.5, FALSE)
```

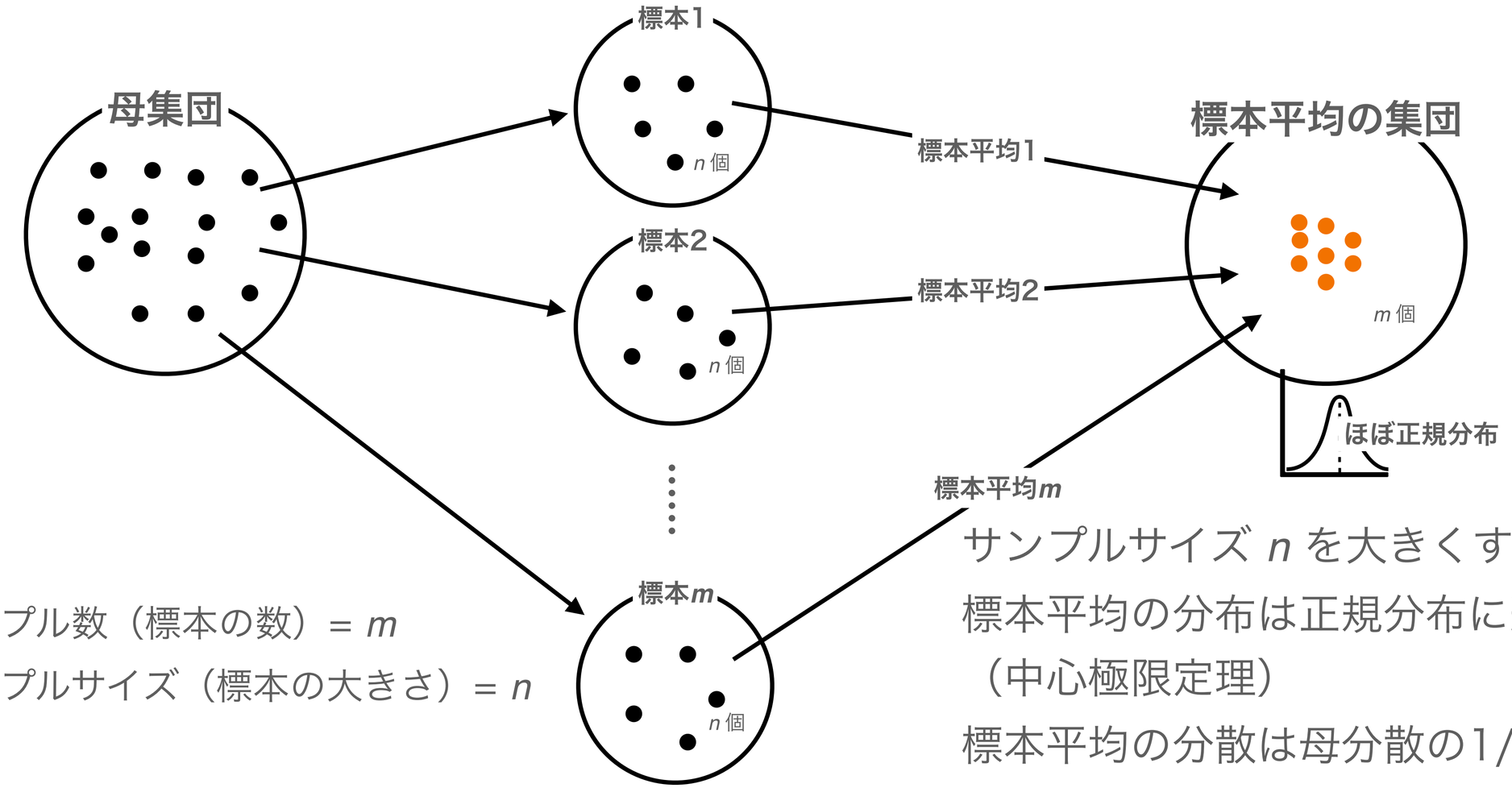
```
=SUMIF(B2:B12, "<="&B4)
```

データは（たいてい）正規分布に従わない



2015年全国学力・学習状況調査　中学理科　http://www.nier.go.jp/kaihatsu/zenkokugakuryoku.html

サンプリングと正規分布（中心極限定理）



サンプル数（標本の数）＝ m

サンプルサイズ（標本の大きさ）＝ n

サンプルサイズ n を大きくすると

標本平均の分布は正規分布に近づく

（中心極限定理）

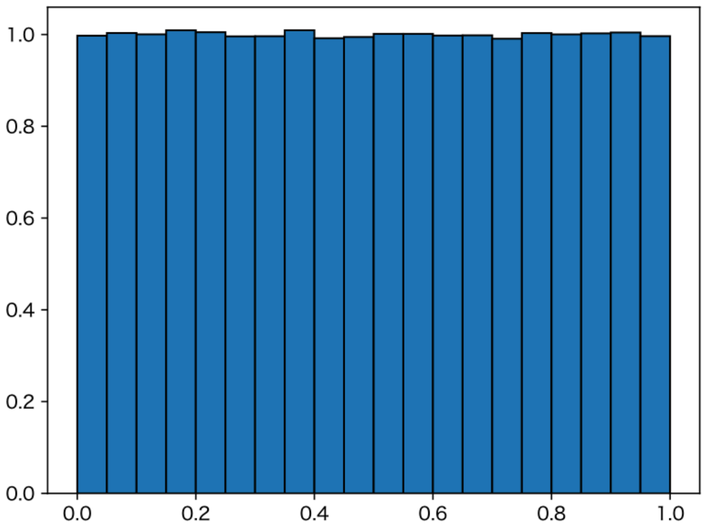
標本平均の分散は母分散の1/n

中心極限定理のシミュレーション

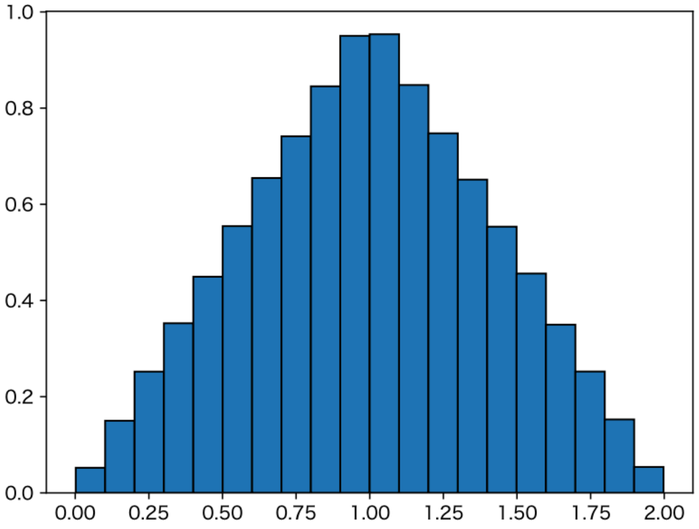
https://okumuralab.org/~okumura/python/normal.html

```
import numpy as np

rng = np.random.default_rng()

N = 1000000  # 百万
```

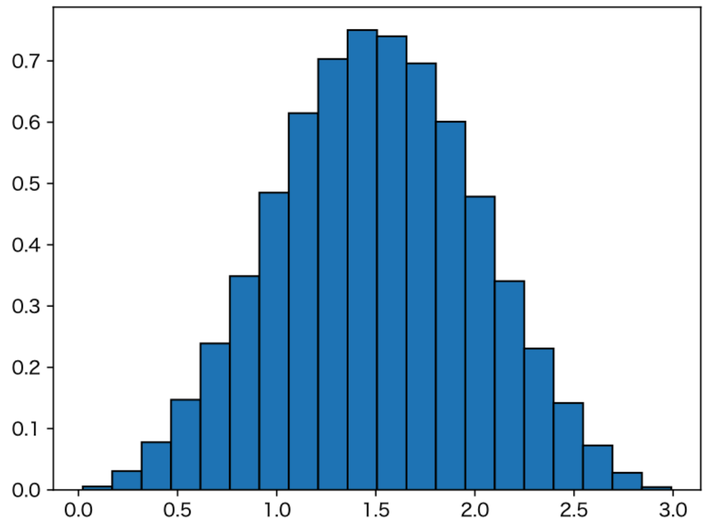
```
plt.hist(rng.random(N),
         edgecolor="black", bins=20, density=True)
```




```
plt.hist(rng.random(N) + rng.random(N),
         bins=20, edgecolor="black", density=True)
```




```
plt.hist(rng.random(N) + rng.random(N) + rng.random(N),
         bins=20, edgecolor="black", density=True)
```



さいころと正規分布（中心極限定理）

さいころ1個

1 ⚀
2 ⚁
3 ⚂
4 ⚃
5 ⚄
6 ⚅

さいころ2個の和

2 ⚀⚀
3 ⚁⚀ ⚀⚁
4 ⚂⚀ ⚁⚁ ⚀⚂
5 ⚃⚀ ⚂⚁ ⚁⚂ ⚀⚃
6 ⚄⚀ ⚃⚁ ⚂⚂ ⚁⚃ ⚀⚄
7 ⚅⚀ ⚄⚁ ⚃⚂ ⚂⚃ ⚁⚄ ⚀⚅
8 ⚅⚁ ⚄⚂ ⚃⚃ ⚂⚄ ⚁⚅
9 ⚅⚂ ⚄⚃ ⚃⚄ ⚂⚅
10 ⚅⚃ ⚄⚄ ⚃⚅
11 ⚅⚄ ⚄⚅
12 ⚅⚅

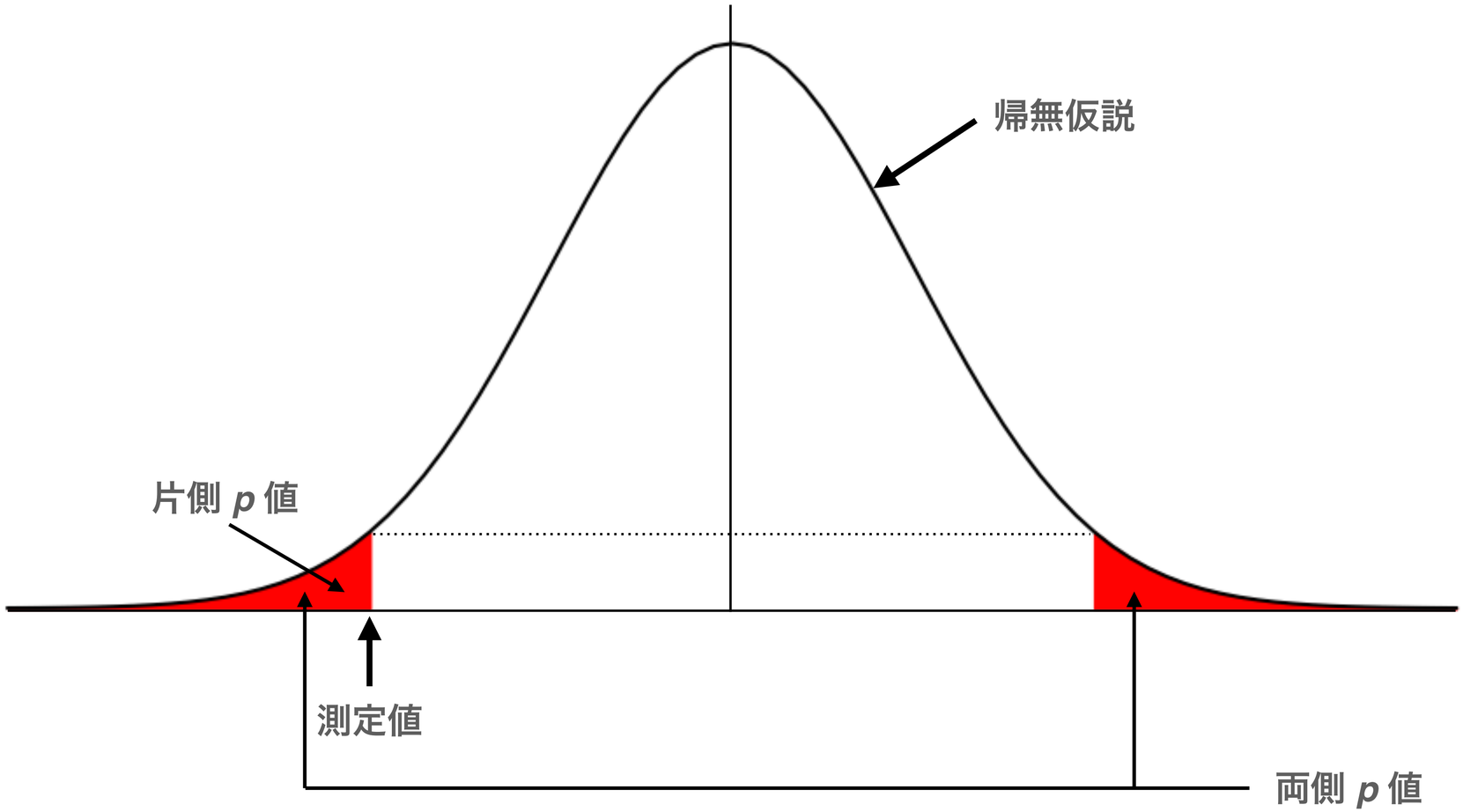
さいころ3個の和

```
def count(n):
    c = 0
    for i in range(1, 7):
        for j in range(1, 7):
            for k in range(1, 7):
                if i + j + k == n:
                    c += 1
    return c

for x in range(3, 19):
    print(f"{x:2d} " + "*" * count(x))
```

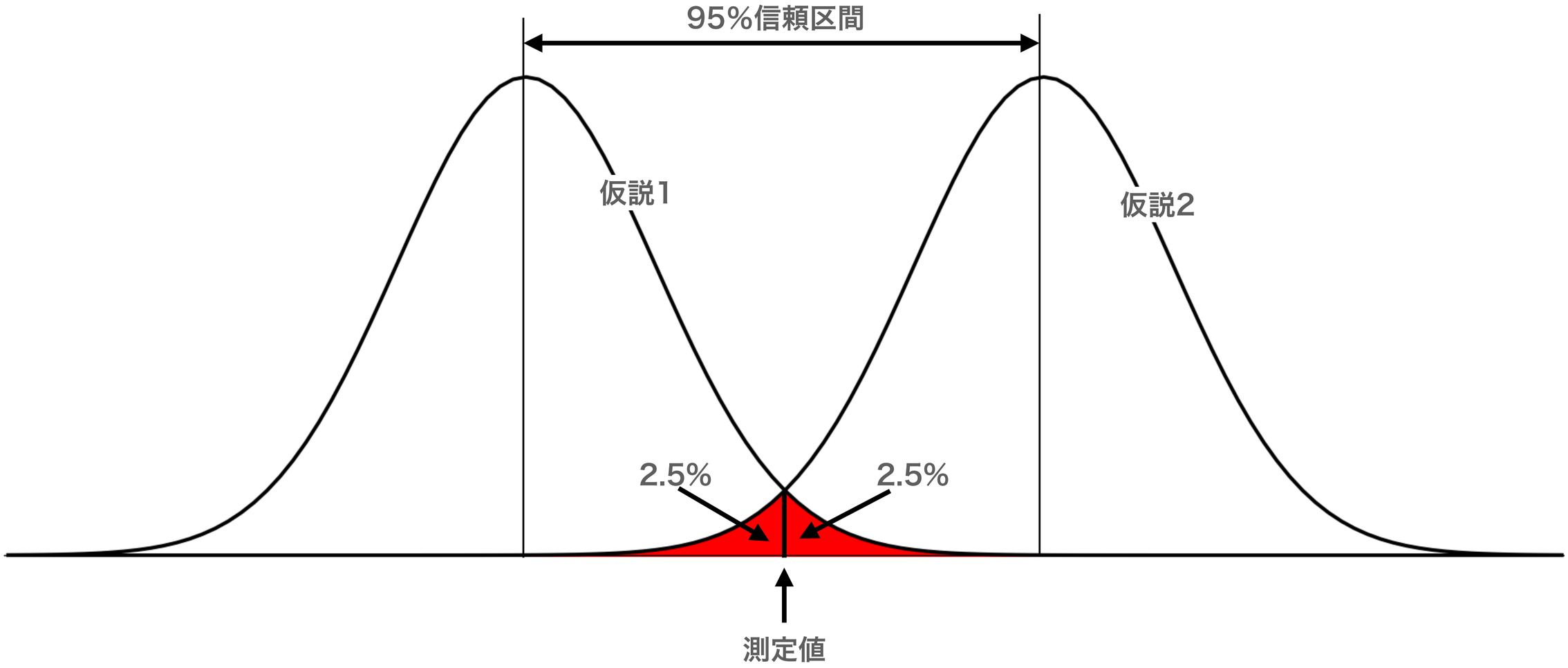
```
 3 *
 4 ***
 5 ******
 6 **********
 7 **************
 8 **********************
 9 **************************
10 ******************************
11 ******************************
12 **************************
13 **********************
14 **************
15 **********
16 ******
17 ***
18 *
```

https://okumuralab.org/~okumura/python/dice.html

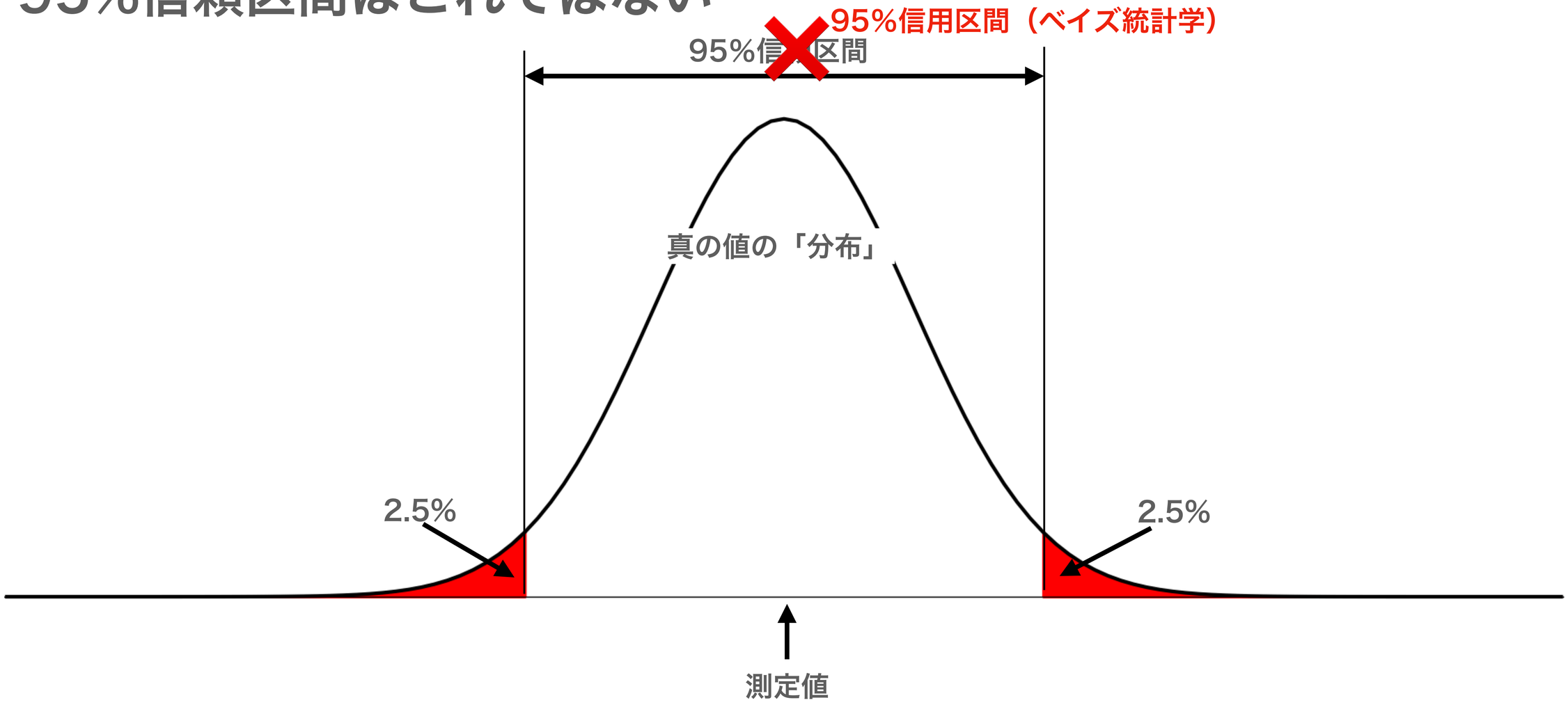
仮説検定、*p* 値（連続分布の場合）



95%信頼区間



95%信頼区間はこれではない

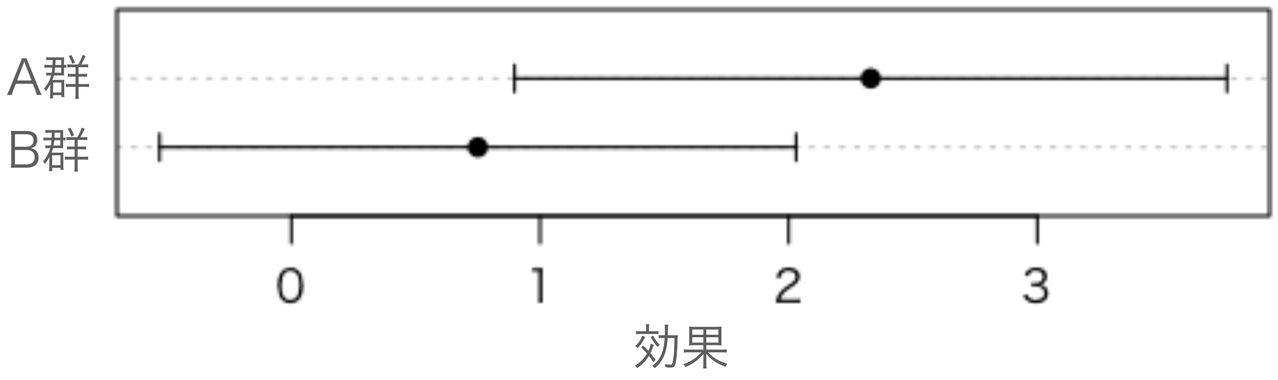


p 値より信頼区間を使おう

95%信頼区間（95% CI）

p > 0.05 になるような帰無仮説のパラメータの範囲

（つまり、どの範囲のパラメータなら実験結果と両立しうるか）



95%信頼区間の
エラーバー

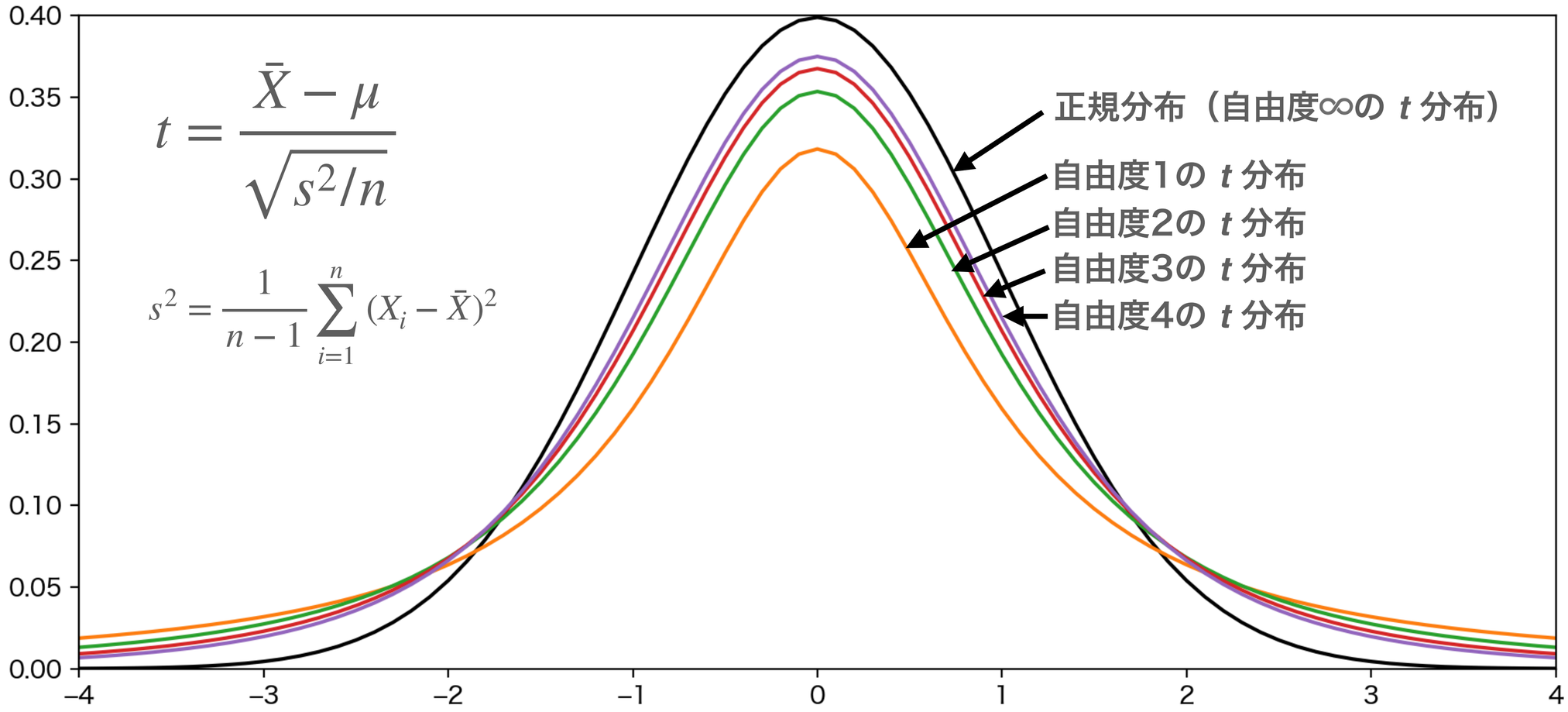
統計学の勘所：幅をもって考える

100人に聞きました！50人が賛成しました！

```
>>> from scipy import stats

>>> stats.binomtest(50, 100).proportion_ci()
ConfidenceInterval(low=0.3983211295033011, high=0.601678870496699)
```

賛成の割合の95%信頼区間は40%〜60%

大きさ *n* の標本平均は自由度 *n*−1 の *t* 分布で検定する

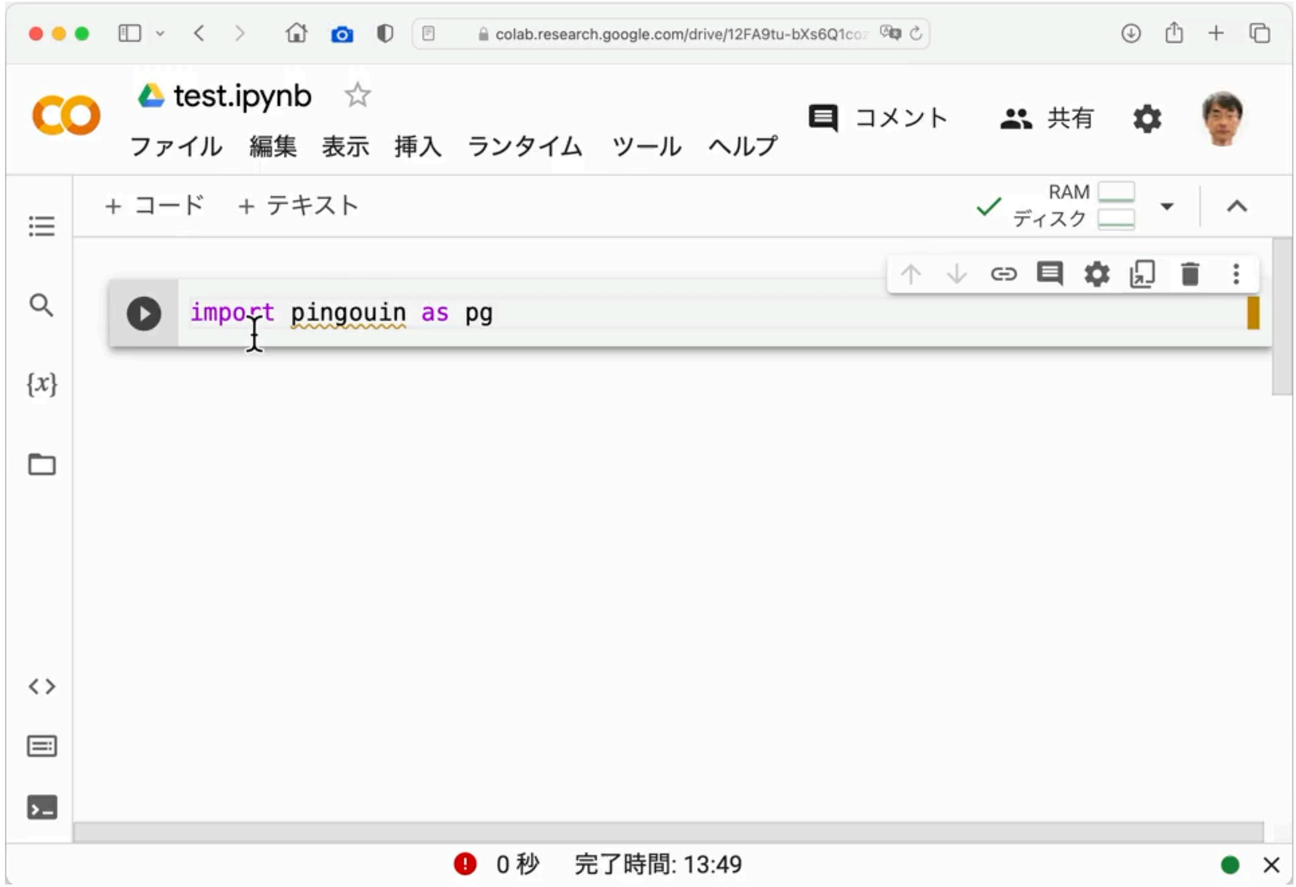


Pythonで *t* 検定（1標本）

Pingouin（ピングイン）というパッケージが便利。Google Colabなら

```
!pip install pingouin
```

しておく。




```
>>> import pingouin as pg

>>> x = [1, 2, 3, 4]
>>> pg.ttest(x, 0)

               T  dof alternative     p-val         CI95%  ...
T-test  3.872983    3   two-sided  0.030466  [0.45, 4.55]  ...
```

← 標本（データ）

← 母集団の平均0という帰無仮説を検定

p 値

標本平均の95%信頼区間

Pythonで *t* 検定（2標本）

```
>>> import pingouin as pg

>>> x = [1, 2, 3, 4]
>>> y = [3, 4, 5, 6, 7]
>>> pg.ttest(x, y, correction=True)

                T       dof alternative     p-val          CI95%  ...
T-test -2.611165  6.980769   two-sided  0.034939  [-4.77, -0.23]  ...
```

***p* 値**

標本平均の差の95%信頼区間

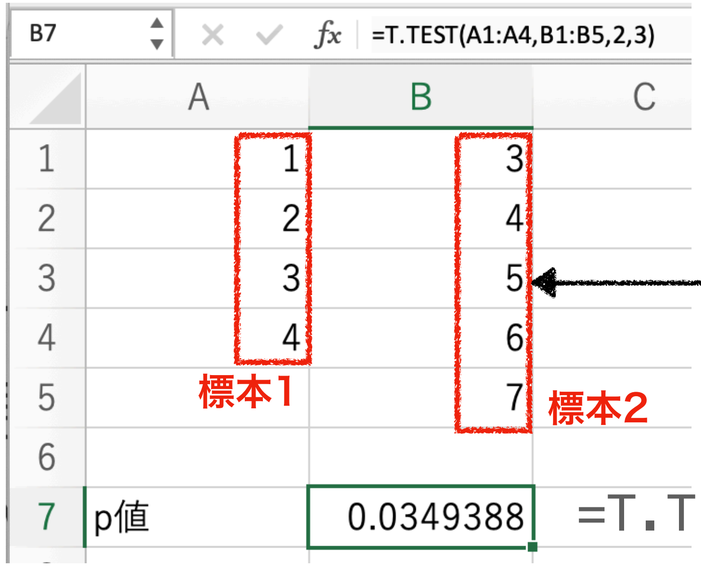
Pythonで *t* 検定（対応のある2標本）

```
>>> import pingouin as pg

>>> x = [1, 2, 3, 4, 5]
>>> y = [3, 4, 5, 7, 6]
>>> pg.ttest(x, y, paired=True)

                T  dof alternative     p-val          CI95%  ...
T-test -6.324555    4   two-sided  0.003198  [-2.88, -1.12]  ...
```

p 値

標本平均の差の95%信頼区間

Excelで *t* 検定



1標本の場合は標本1と同数の0を並べ、
対応のある検定を行う

```
=T.TEST(A1:A4,B1:B5,2,3)
```

1: 片側　1: 対応のある場合
2: 両側　2: 等分散を仮定する
　　3: 等分散を仮定しない

Rで *t* 検定

```
> x = c(1, 2, 3, 4)
> y = c(3, 4, 5, 6, 7)
> t.test(x, y)
```

1標本の場合は `t.test(x)`

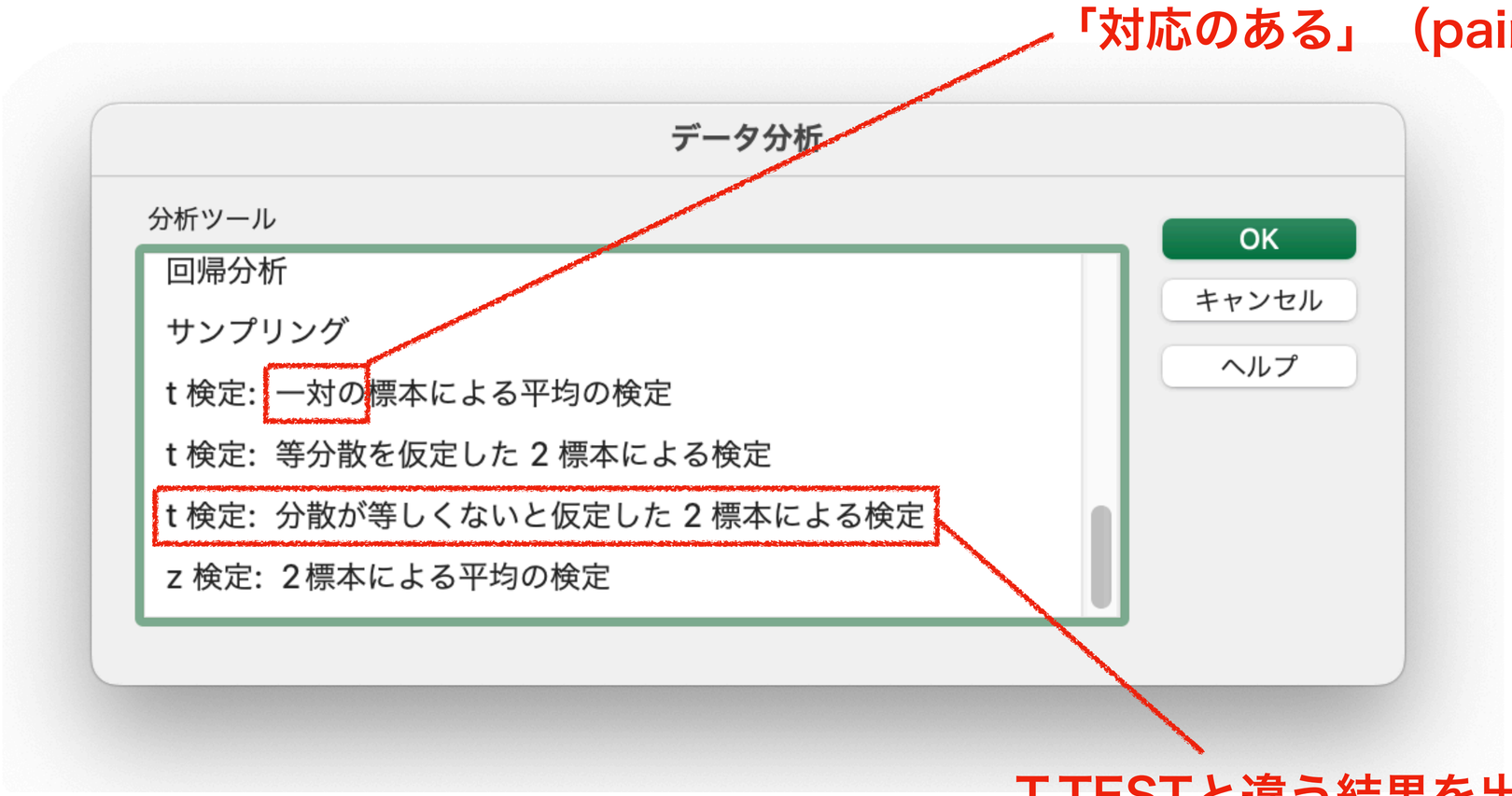
対応のある場合は `t.test(x, y, paired=TRUE)`

```
        Welch Two Sample t-test

data:  x and y
t = -2.6112, df = 6.9808, p-value = 0.03494
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -4.7652203 -0.2347797
sample estimates:
mean of x mean of y
      2.5       5.0
```

***p* 値**

標本平均の差の95%信頼区間

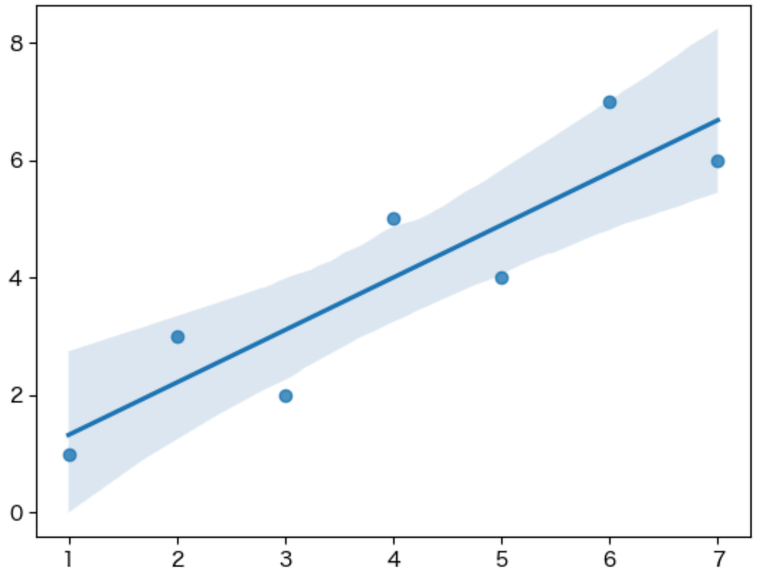
Excelの「データ分析」の *t* 検定は要注意



Pythonで回帰分析

```
>>> import seaborn as sns

>>> x = [1, 2, 3, 4, 5, 6, 7]
>>> y = [1, 3, 2, 5, 4, 7, 6]
>>> sns.regplot(x=x, y=y)

>>> import pingouin as pg
```

散布図



相関係数

```
>>> pg.corr(x, y)
             n         r         CI95%     p-val   BF10     power
pearson  7  0.892857  [0.43, 0.98]  0.006807  8.961  0.854032
```

相関係数　95%信頼区間　*p* 値

線形回帰

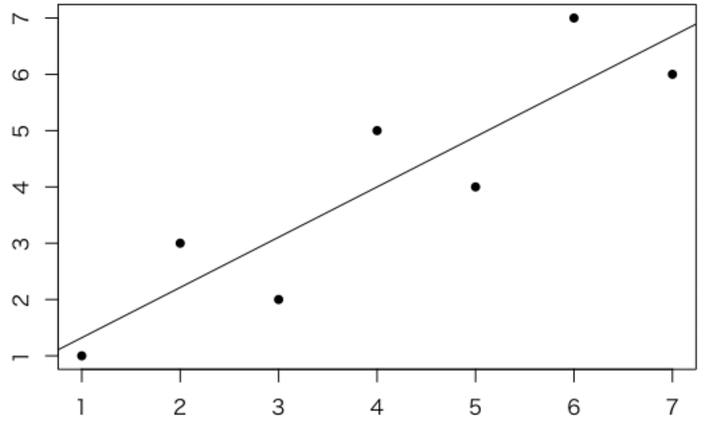
```
>>> pg.linear_regression(x, y)
        names      coef        se         T      pval        r2    adj_r2  CI[2.5%]  CI[97.5%]
0   Intercept  0.428571  0.900680  0.475831  0.654258  0.797194  0.756633 -1.886700   2.743843
1          x1  0.892857  0.201398  4.433293  0.006807  0.797194  0.756633  0.375147   1.410568
```

傾き　*p* 値　R^2（相関係数の2乗）　95%信頼区間

Rで回帰分析

```
> x = c(1, 2, 3, 4, 5, 6, 7)
> y = c(1, 3, 2, 5, 4, 7, 6)
> plot(x, y, pch=16)
> abline(lm(y ~ x))
```

散布図



相関係数

```
> cor.test(x, y)

        Pearson's product-moment correlation

data:  x and y
t = 4.4333, df = 5, p-value = 0.006807
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.4267022 0.9841793
sample estimates:
      cor
0.8928571
```

p 値

95%信頼区間

相関係数

線形回帰

```
> summary(lm(y ~ x))

Call:
lm(formula = y ~ x)

Residuals:
      1       2       3       4       5       6       7
-0.3214  0.7857 -1.1071  1.0000 -0.8929  1.2143 -0.6786

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   0.4286     0.9007   0.476  0.65426
x             0.8929     0.2014   4.433  0.00681 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.066 on 5 degrees of freedom
Multiple R-squared:  0.7972,    Adjusted R-squared:  0.7566
F-statistic: 19.65 on 1 and 5 DF,  p-value: 0.006807
```

傾き

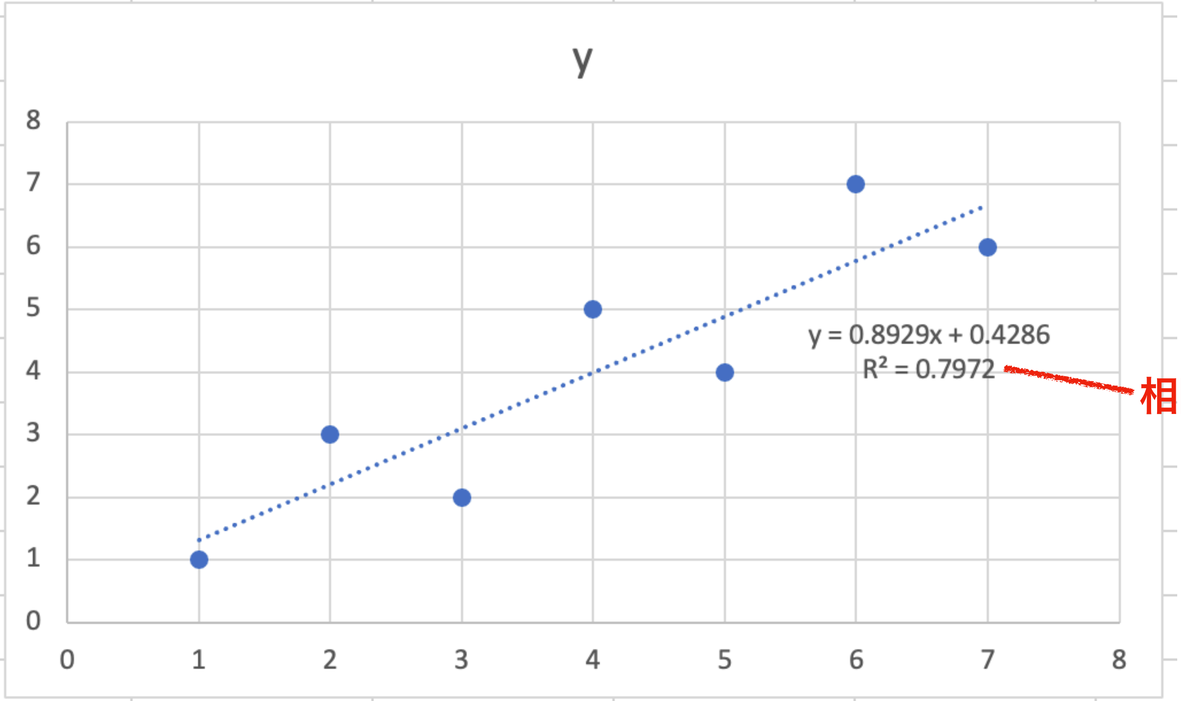
p 値

R^2（相関係数の2乗）

Excelで回帰分析

B10 =CORREL(A2:A8,B2:B8)

	A	B
1	x	y
2	1	1
3	2	3
4	3	2
5	4	5
6	5	4
7	6	7
8	7	6
9		
10	相関係数	0.89285714
11		
12		



相関係数の2乗

「近似曲線の追加…」で回帰直線を描く

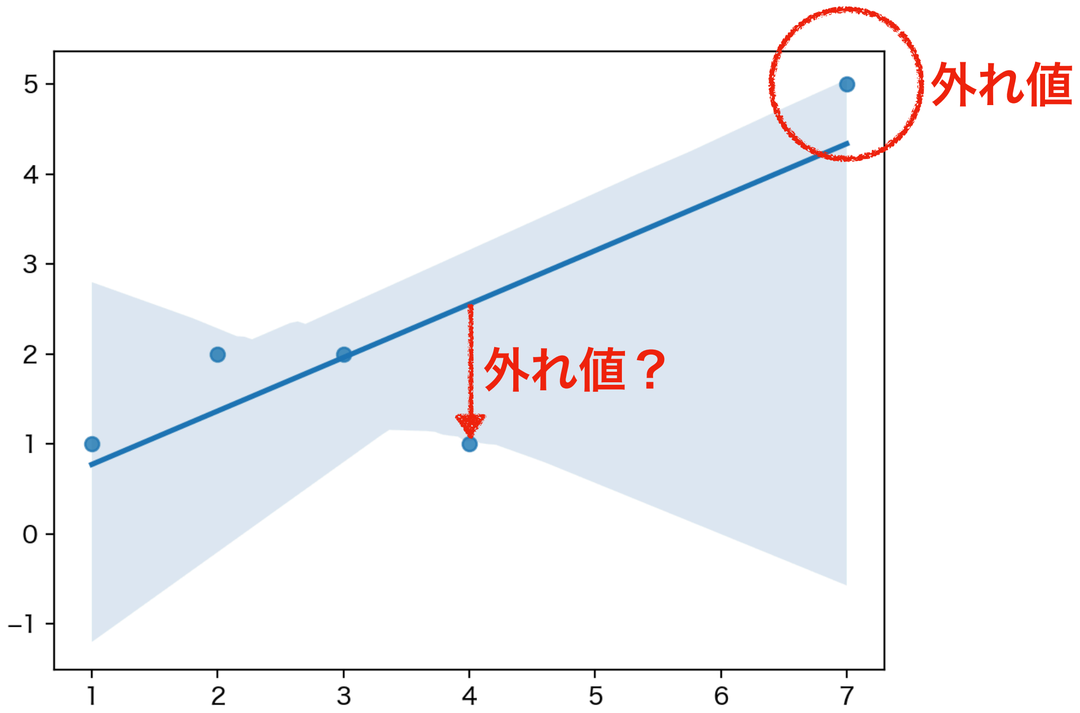
Excelの分析ツールで回帰分析

	A	B	C	D	E	F	G	H	I
1	概要								
2									
3	回帰統計								
4	重相関 R	0.89285714	相関係数						
5	重決定 R2	0.79719388	相関係数の2乗						
6	補正 R2	0.75663265							
7	標準誤差	1.06569897							
8	観測数	7							
9									
10	分散分析表								
11		自由度	変動	分散	測された分散	有意 F			
12	回帰	1	22.3214286	22.3214286	19.6540881	0.00680719			
13	残差	5	5.67857143	1.13571429					
14	合計	6	28						
15									
16		係数	標準誤差	t	P-値	下限 95%	上限 95%	下限 95.0%	上限 95.0%
17	切片	0.42857143	0.90068002	0.47583095	0.65425818	-1.8867003	2.74384312	-1.8867003	2.74384312
18	X 値 1	0.89285714	0.20139817	4.43329314	0.00680719	0.37514666	1.41056763	0.37514666	1.41056763

傾き　　*p* 値　　95%信頼区間

外れ値はどっち？

```
import seaborn as sns

x = [1, 2, 3, 4, 7]
y = [1, 2, 2, 1, 5]
sns.regplot(x=x, y=y)
```



x = 4 の点が回帰直線から一番外れているが、真の外れ値は x = 7 の点

共通テスト試作問題「情報I」

https://okumuralab.org/~okumura/python/221118.html

第４問　次の文章を読み，後の問い（**問１～５**）に答えよ。（配点　25）

次の表１は，国が実施した生活時間の実態に関する統計調査をもとに，15 歳以上 19 歳以下の若年層について，都道府県別に平日１日の中で各生活行動に費やした時間（分）の平均値を，スマートフォン・パソコンなどの使用時間をもとにグループに分けてまとめたものの一部である。ここでは，１日のスマートフォン・パソコンなどの使用時間が１時間未満の人を表１－A，３時間以上６時間未満の人を表１－Bとしている。

表１－A：スマートフォン・パソコンなどの使用時間が１時間未満の人の生活行動時間に関する都道府県別平均値

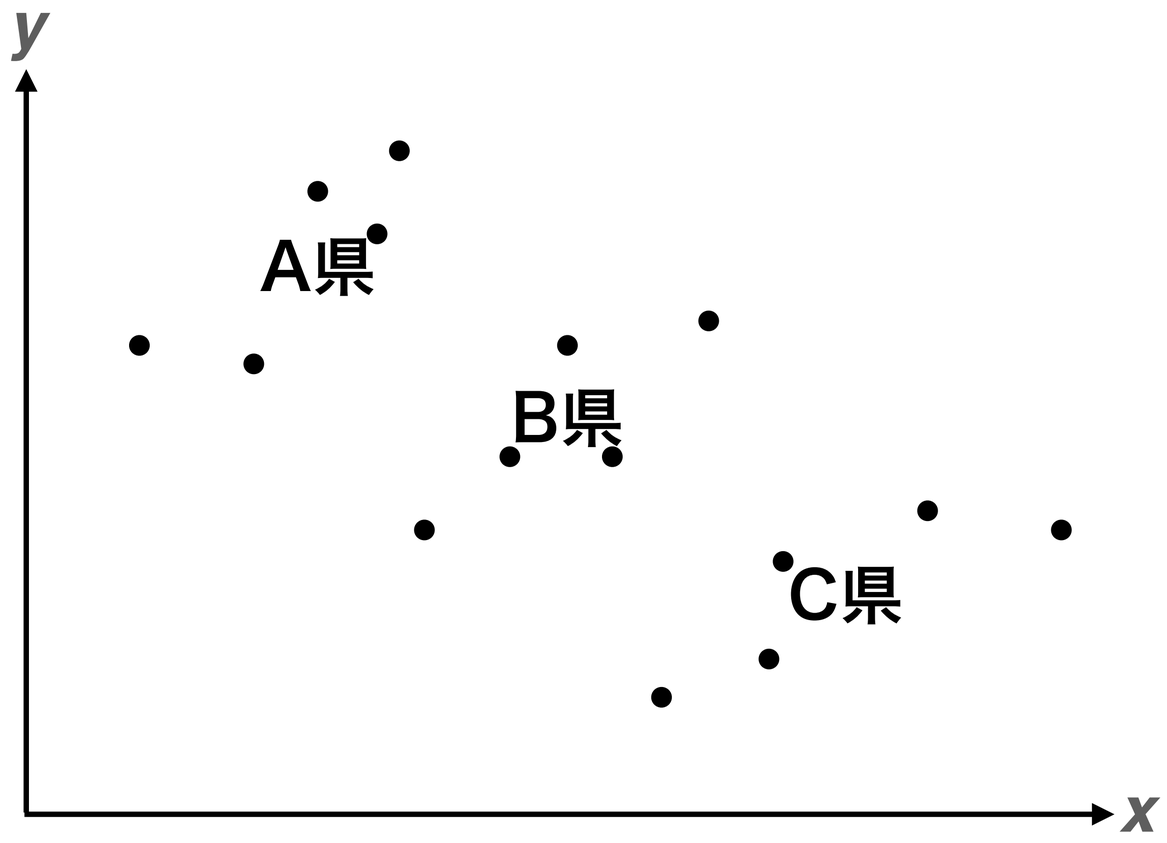
都道府県	睡眠（分）	身の回りの用事（分）	食事（分）	通学（分）	学業（分）	趣味・娯楽（分）
北海道	439	74	79	60	465	8
青森県	411	74	73	98	480	13
茨城県	407	61	80	79	552	11
栃木県	433	76	113	50	445	57

表１－B：スマートフォン・パソコンなどの使用時間が３時間以上６時間未満の人の生活行動時間に関する都道府県別平均値

都道府県	睡眠（分）	身の回りの用事（分）	食事（分）	通学（分）	学業（分）	趣味・娯楽（分）
北海道	436	74	88	63	411	64
青森県	461	57	83	55	269	44
茨城県	443	80	81	82	423	63
栃木県	386	120	79	77	504	33

（出典：総務省統計局の平成 28 年社会生活基本調査により作成）

生態学的誤謬・シンプソンのパラドックス

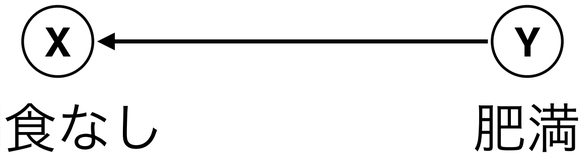


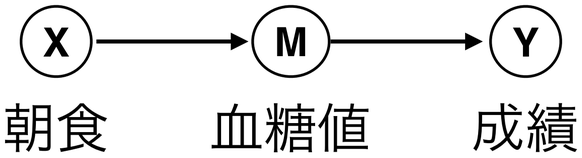
都道府県データは要注意！

生態学的誤謬：
都道府県平均の相関は負なのに
各都道府県内の個人の相関は正

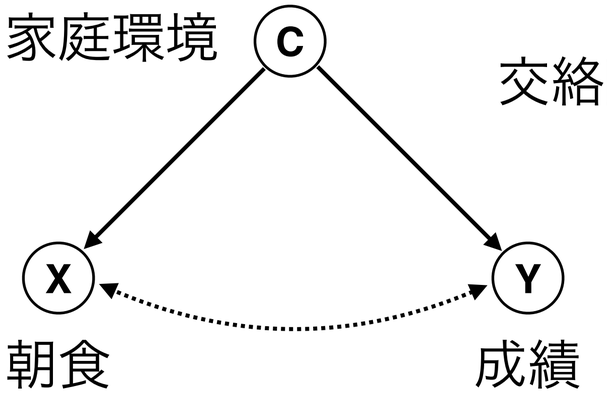
シンプソンのパラドックス：
どの都道府県でも相関は
正なのに全国の相関は負

相関・因果・擬似相関

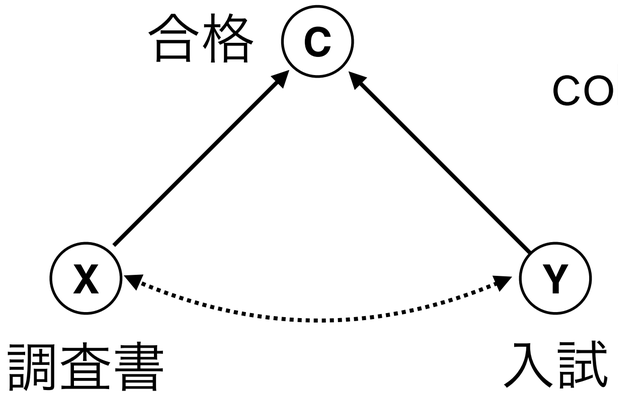




mediator

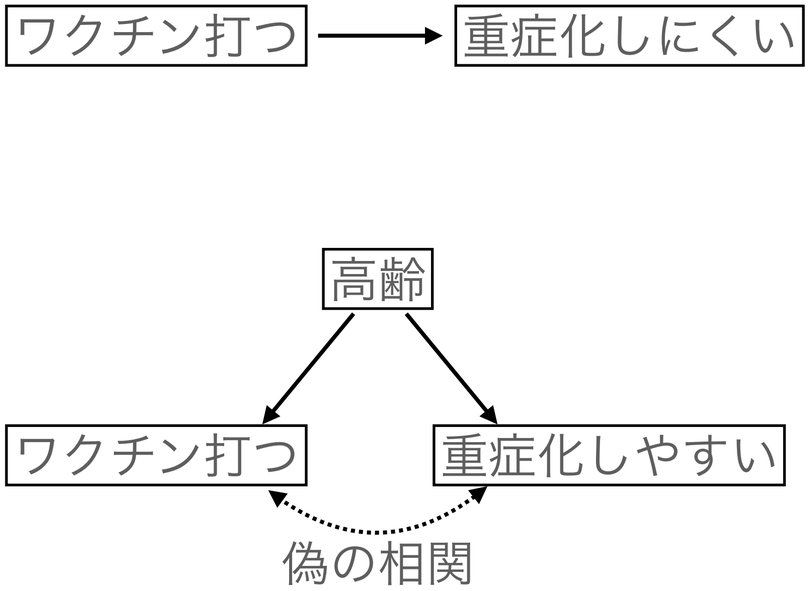


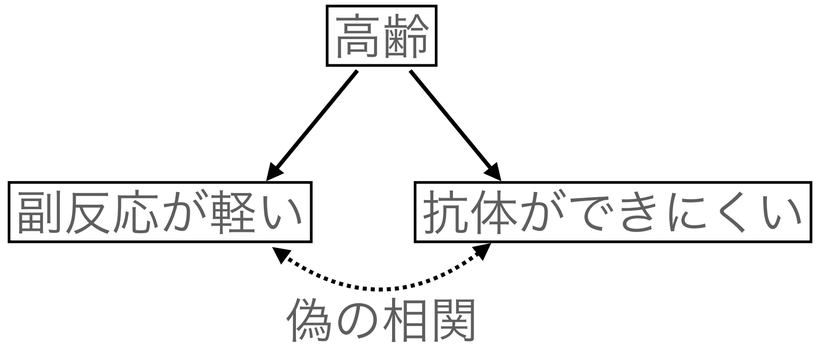
交絡因子（confounder）



collider・選択バイアス

擬似相関

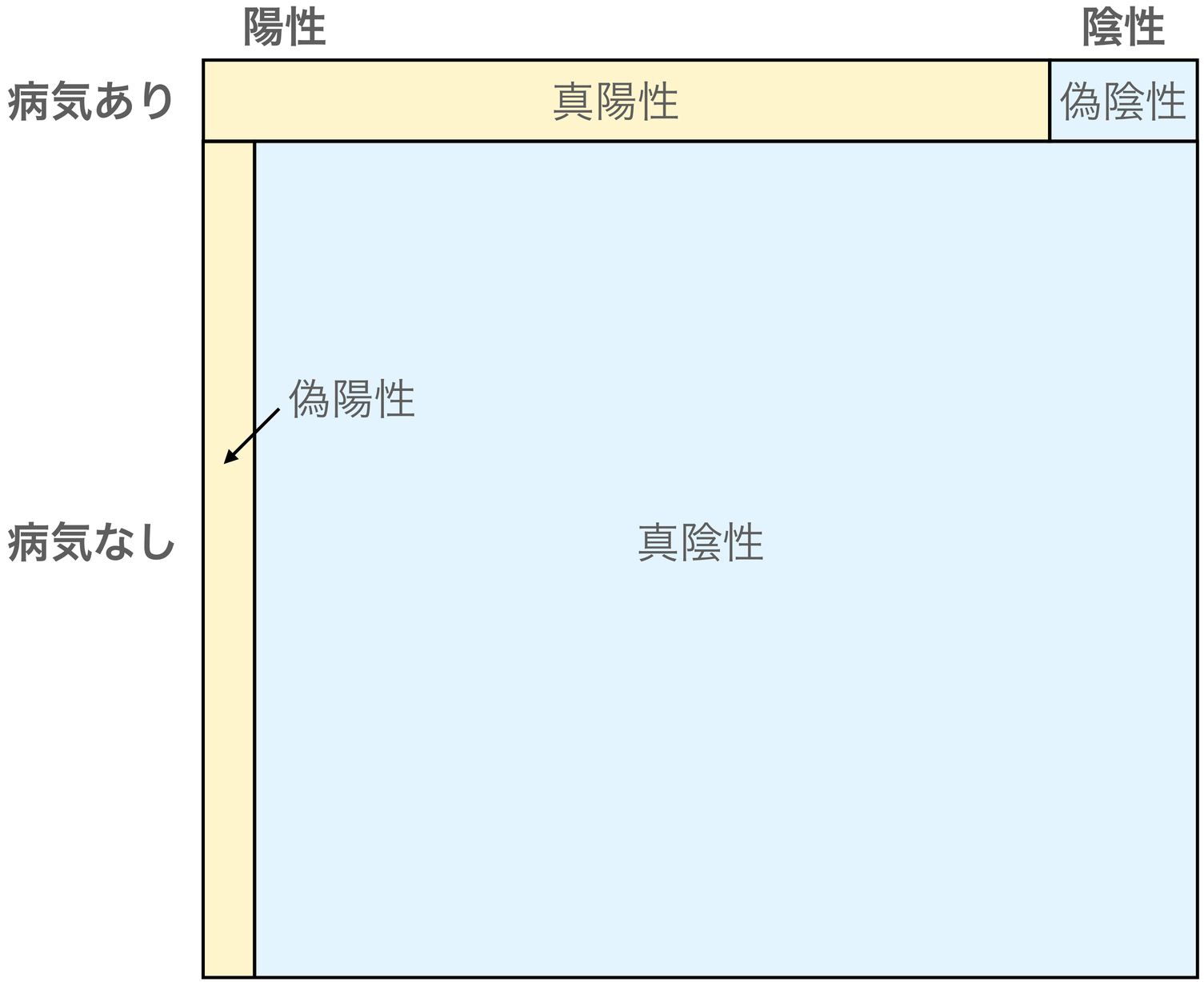

ワクチン打つ
重症化しにくい
高齢
ワクチン打つ
重症化しやすい
偽の相関


高齢
副反応が軽い
抗体ができにくい
偽の相関

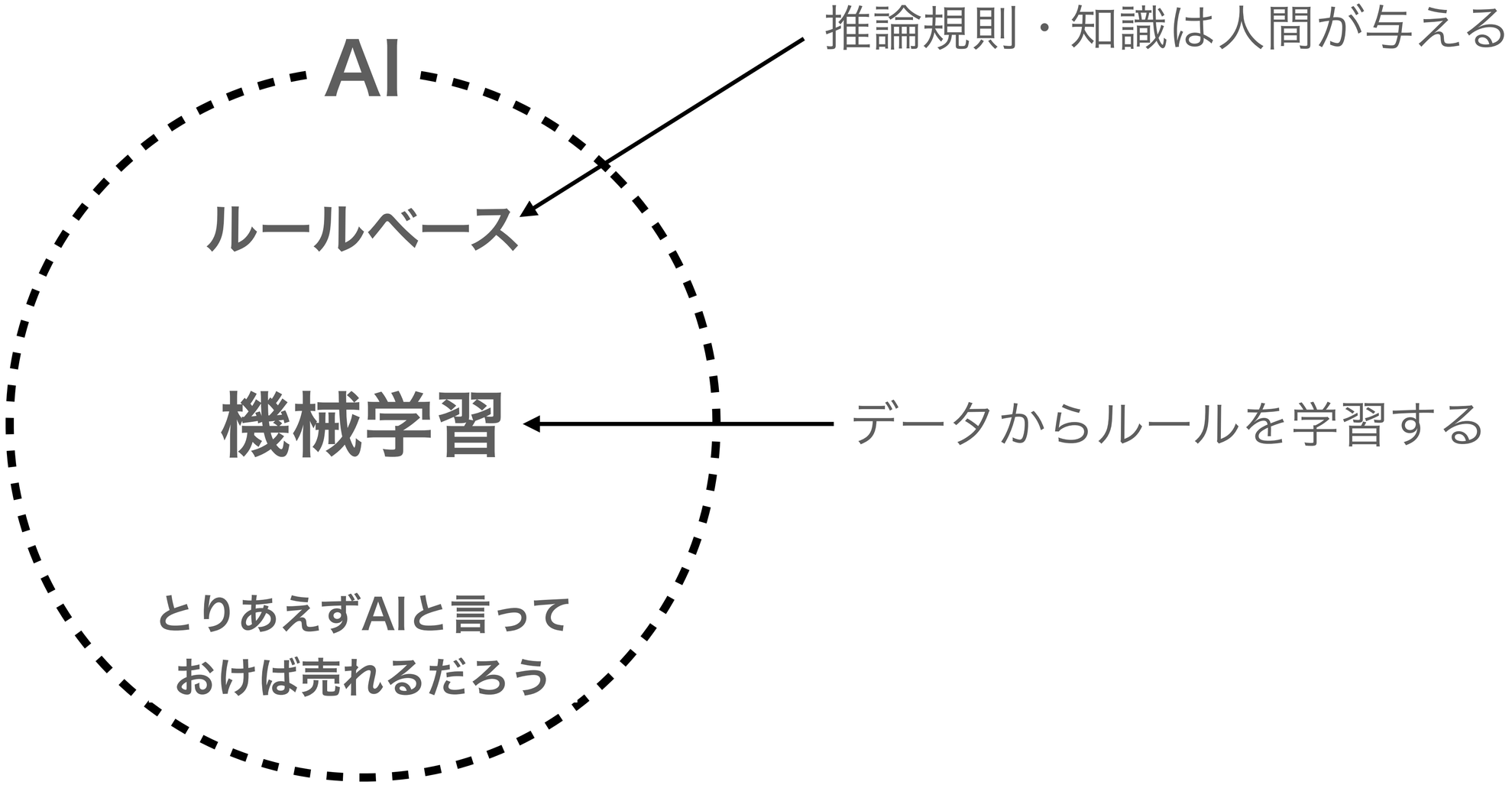
ベイズの定理？

$$感度 = \frac{真陽性}{真陽性 + 偽陰性}$$

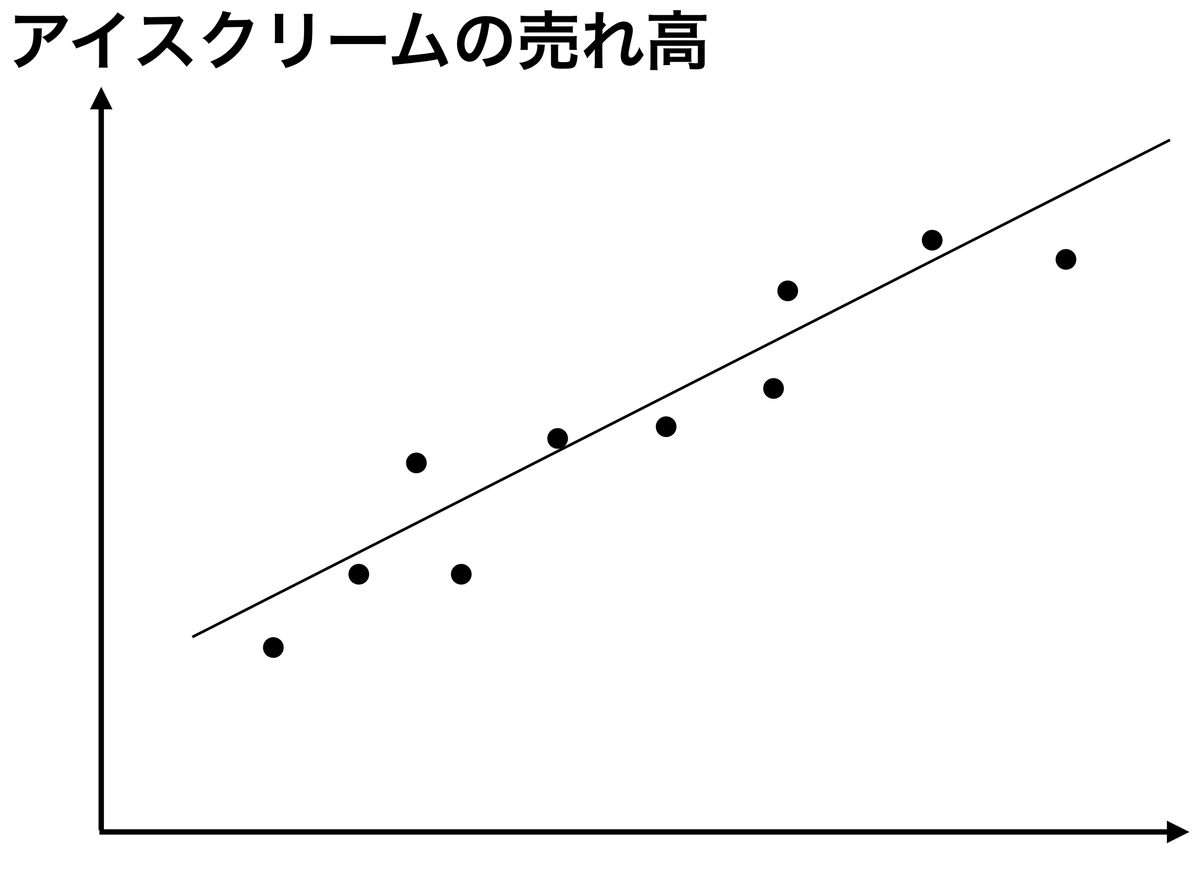
$$特異度 = \frac{真陰性}{偽陽性 + 真陰性}$$



AIをどう教える？

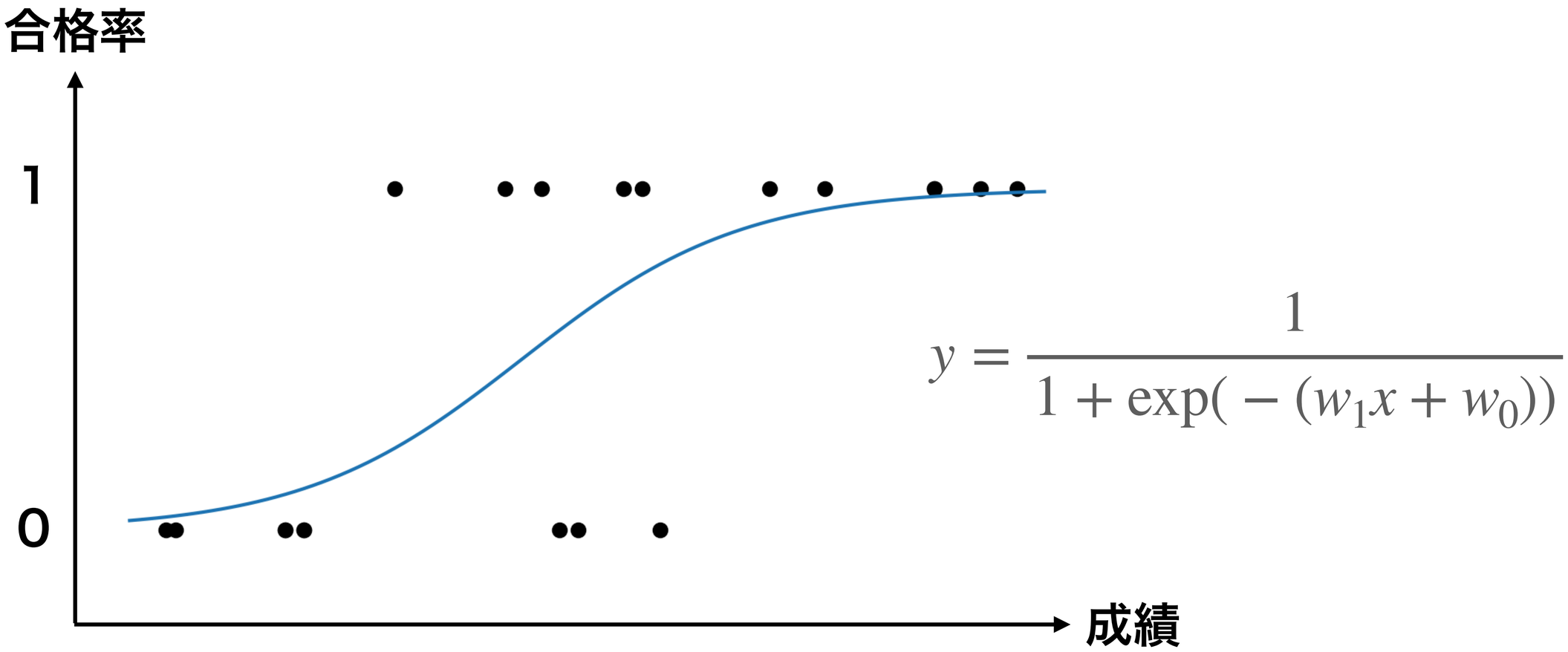

AI
ルールベース
機械学習
とりあえずAIと言って
おけば売れるだろう
推論規則・知識は人間が与える
データからルールを学習する

回帰も機械学習



データからルールを学習する
という意味では機械学習

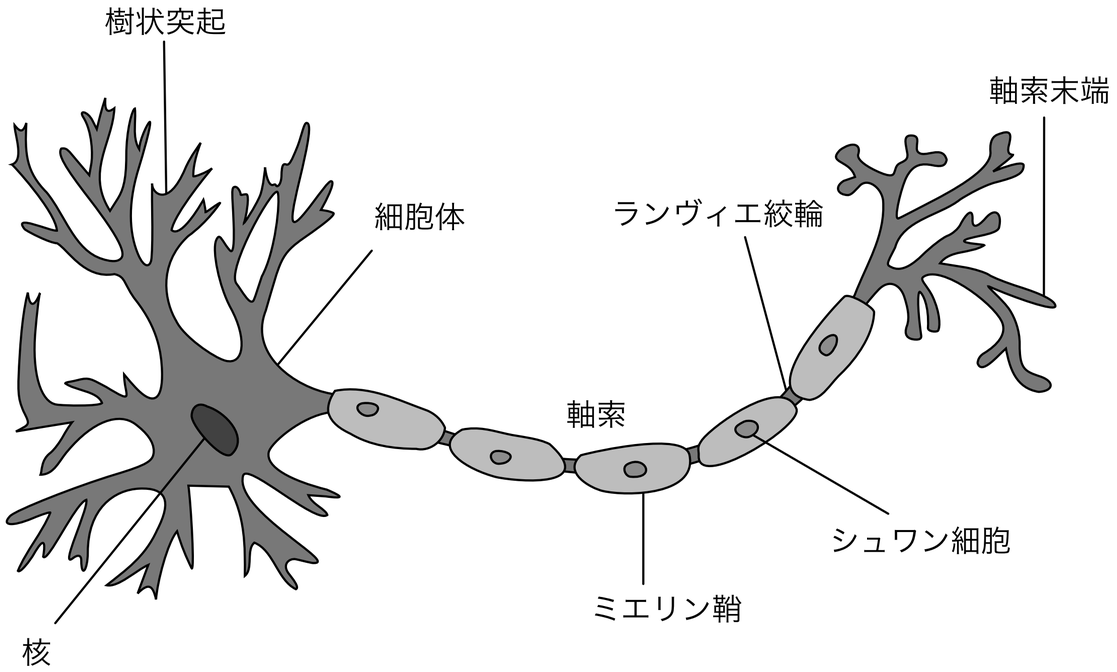
分類も機械学習（ロジスティック回帰）

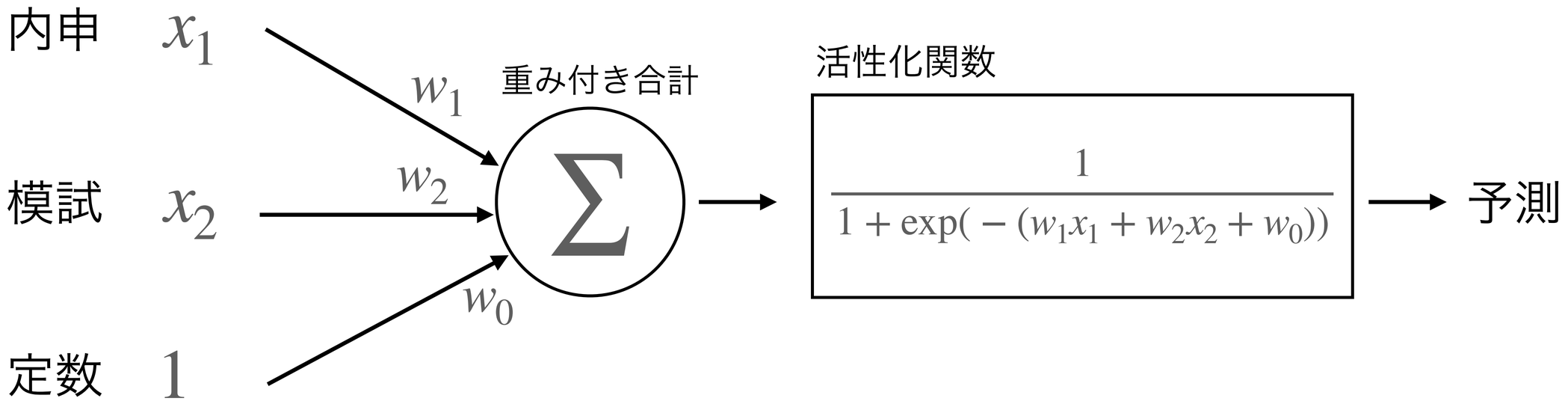


$$y = \frac{1}{1 + \exp(-(w_1 x + w_0))}$$

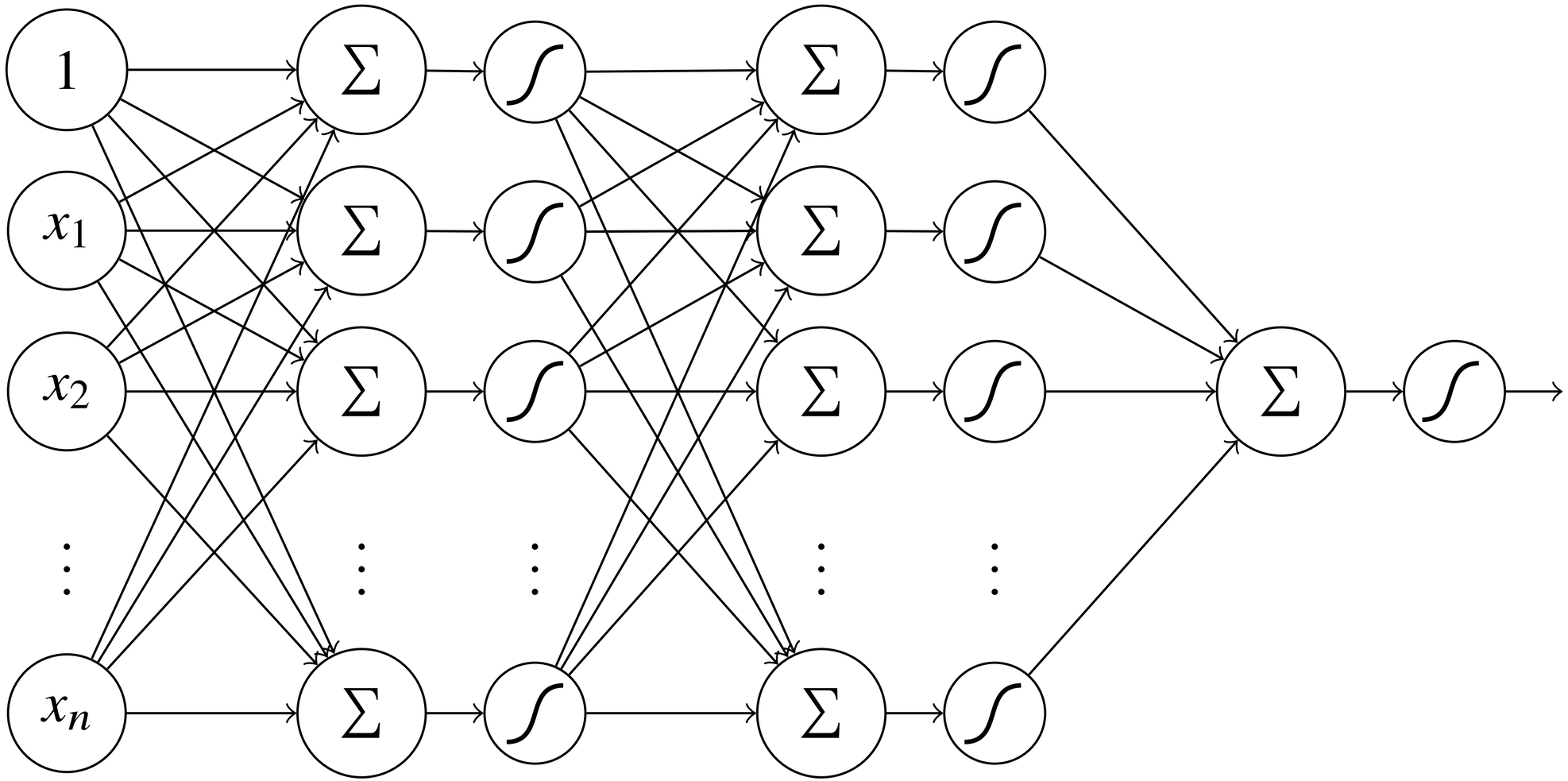
https://okumuralab.org/~okumura/python/logisticregression.html

多変数のロジスティック回帰





ニューラルネット→ディープラーニング



ChatGPTなどの仕組み

昔々あるところに [　　] ← これくらいなら過去の統計で出せる

…… ……（何千文字） …… …… [　　] ← 過去に出会っていない長文の補完

さらに人間が作成した模範問答を学習する

さらに実際に生成した答えを人間が評価する（強化学習）

最後に

データサイエンス・AIは日に日に進歩している。

新しいことを学ぶ姿勢、ないものは自分で作り出す姿勢こそ大切。