

# AI、数学、教育

第124回数学実践研究会講演

2023/01/28

丸山 哲太郎

# 自己紹介

丸山 哲太郎

- 東京大学大学院工学系研究科 修了
  - 非線形数理 (カオス)
- 富士通研究所
  - 大規模ストレージの研究開発
- リクルートコミュニケーションズ
  - Data Scientist / Machine Learning Engineer / Data Engineer
- メルカリ
  - Machine Learning Engineer / Engineering Manager
- TechnoFACE
  - Data Scientist / General Manager / 取締役



# Agenda

- AI とは何か
  - AI の種類
  - 機械学習の種類
- AI と数学
  - AI だから分かる数学の必要性
  - AI だから分かる数学の美しさ
- AI と数学と教育
  - どんな人材が求められているのか
  - どんな教育が求められているのか

A decorative graphic on the left side of the slide, consisting of several overlapping, flowing blue lines that create a sense of movement and depth. The lines are semi-transparent and vary in intensity from light to dark blue.

「AI」とは何か

# 「AI」という言葉の意味

- AI: Artificial Intelligence (人工知能)
  - "Intelligence demonstrated by machines"
    - from Wikipedia(en) "Artificial Intelligence"
  - コンピューターによって実現される『知能』
- Intelligence (知能)
  - "The ability to perceive or infer information, and to retain it as knowledge to be applied towards adaptive behaviors within an environment or context"
    - from Wikipedia(en) "Intelligence"
  - ある環境・状況下において、適切な**行動**を取れるような知識として、情報を**認識**・**推論**・**保持**できる能力

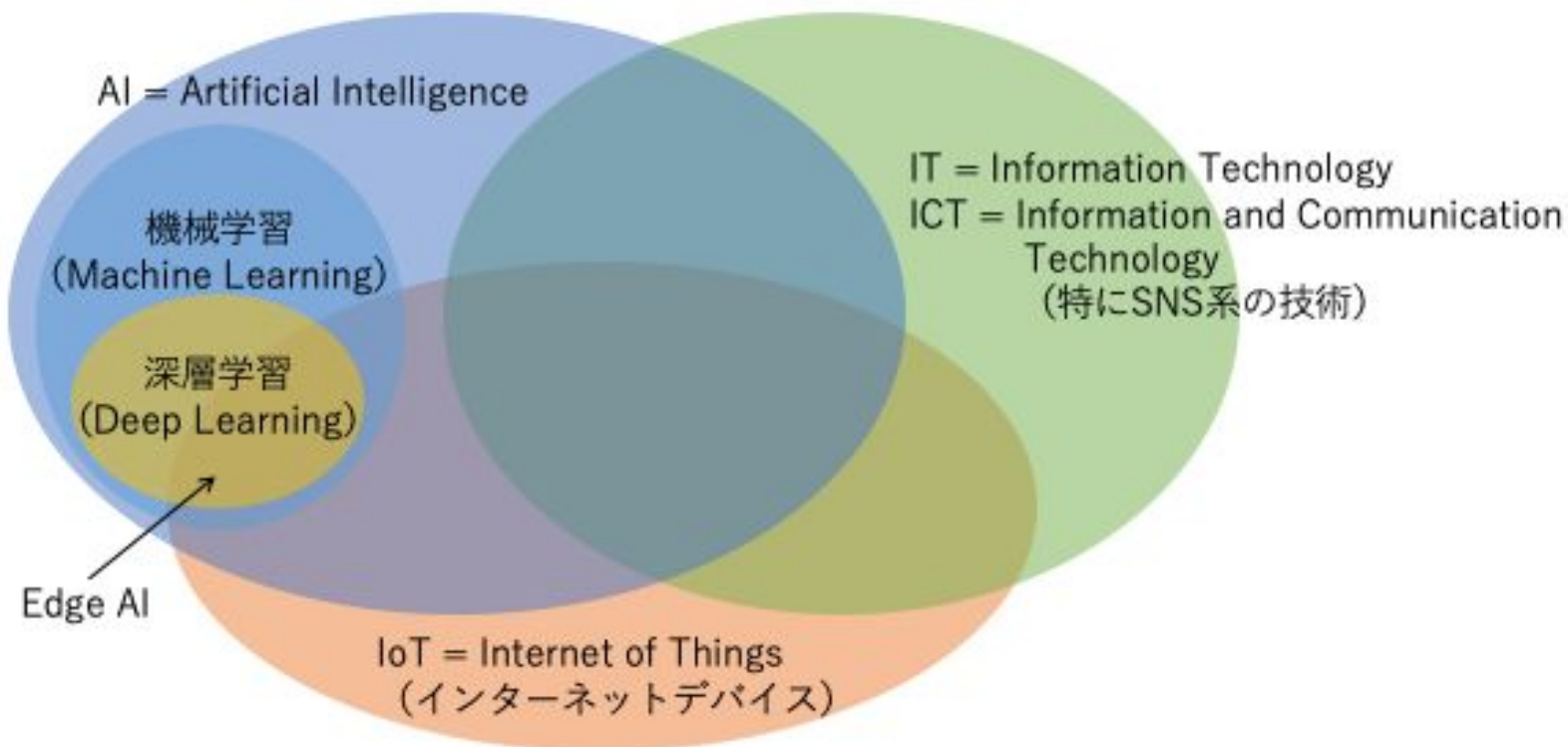
# 「AI」の一例

## ログインシステム

- ユーザー名とパスワードがデータベースに登録されている
  - 情報の**保持**
- ユーザーがユーザー名とパスワードを入力する
  - 情報の**認識**
- ユーザー名とそのパスワードが登録されているものと一致
  - 情報の**推論**
- そのユーザーがシステムの機能を使えるようになる
  - 適切な**行動**

ログインシステムは、立派な「AI」

# 「AI」という言葉のイメージ



「AI」という言葉は非常に広い意味を持つ

# 「AI」は何が出来るのか

「AI」は何が出来るのかを理解するためには、  
「AI」にはどんな種類があるのかを知る必要がある。

一般的な「AI」は、大きく分けて3種類

- なんちゃってAI (ルールベース)
- 最適化
- 機械学習 (Machine Learning:ML) ← 狭義のAI



## 例え話：人生ゲーム

各自が自分の作ったルーレットを持ち寄って遊ぶ

- 人生ゲームをゴールする=目的
- 独自のルーレットを作る=自分が使う「AI」



# 人生ゲームのゴールとルーレット

- 目的：とにかく最速でゴールする！
  - 手段：必ず10が出るルーレットを作る
  - → なんちゃってAIルーレット
    - 単純なルールを守り動作するシステム
    - 例) 特徴的な単語に反応して返信するチャットボット
- 目的：あまり派手にならず、それでいて最速にゴール
  - 手段：直近5回の合計が40になるルーレットを作る
  - → 最適化ルーレット
    - 限られた資源の中で最大の効果を得るよう計算する
    - 例) アルバイトのスケジュールを自動で組む
- どちらも、バグ無く動作すれば永遠に使い続けられる

# 機械学習ルーレット

人生ゲームは最速でゴールすることだけが目的じゃない

- 人生ゲームをどのようにゴールしたいか（要件）
  - 所持金最大、子供が最多、.....
- 要件を表すデータは何か（目的変数）
  - 所持金の額、子供の数、.....
- 要件の達成のために必要なデータは何か（特徴量）
  - 駒の位置、升目の内容、カードの内容、.....
- 要件が満たされているか判断するデータは何か（指標）
  - 所持金ランキング、.....

単純なシステムではなく、様々な要求・状況に対応する

## 3 種類の一般的な「AI」

- なんちゃって AI の要件の満たし方
  - 入力 → ルールに従い計算 → 出力 (要するにITとほぼ同義)
  - 学習：しない
  - 例) 質問に「電源」という単語 → 「コンセントを確認」
- 最適化の要件の満たし方
  - 入力
    - 限られた条件下で最大効果を得る特殊な計算 (線形計画法等)
    - 出力
  - 学習：しない
  - 例) 常に店に2人いるように、バイトのシフトを調整

### 3 種類の一般的な「AI」 (Cont.)

- 機械学習の要件の満たし方

- 学習により、データの中から最適な「ルール」を探す
  - モデル：最適な予測が出来る可能性がある方法（式）
    - $y$ （目的変数） =  $a * x$ （特徴量） +  $b$
  - 学習：データからモデルのパラメータを「統計的」に「推定」すること
    - 例) 最適な  $a, b$  の値を最尤法で推定
    - データには誤差があるので、決定的にはなり得ない
  - ルール = モデル + パラメータ
- 学習済みモデルが要件を満たすか、指標で確認
  - 要件 = 目的変数になるべく理想的な値を示すこと
  - 指標 = 理想的な値からの誤差を計算する式
- 入力 → 学習済みモデルで計算 → 「予測」（推定だから！）

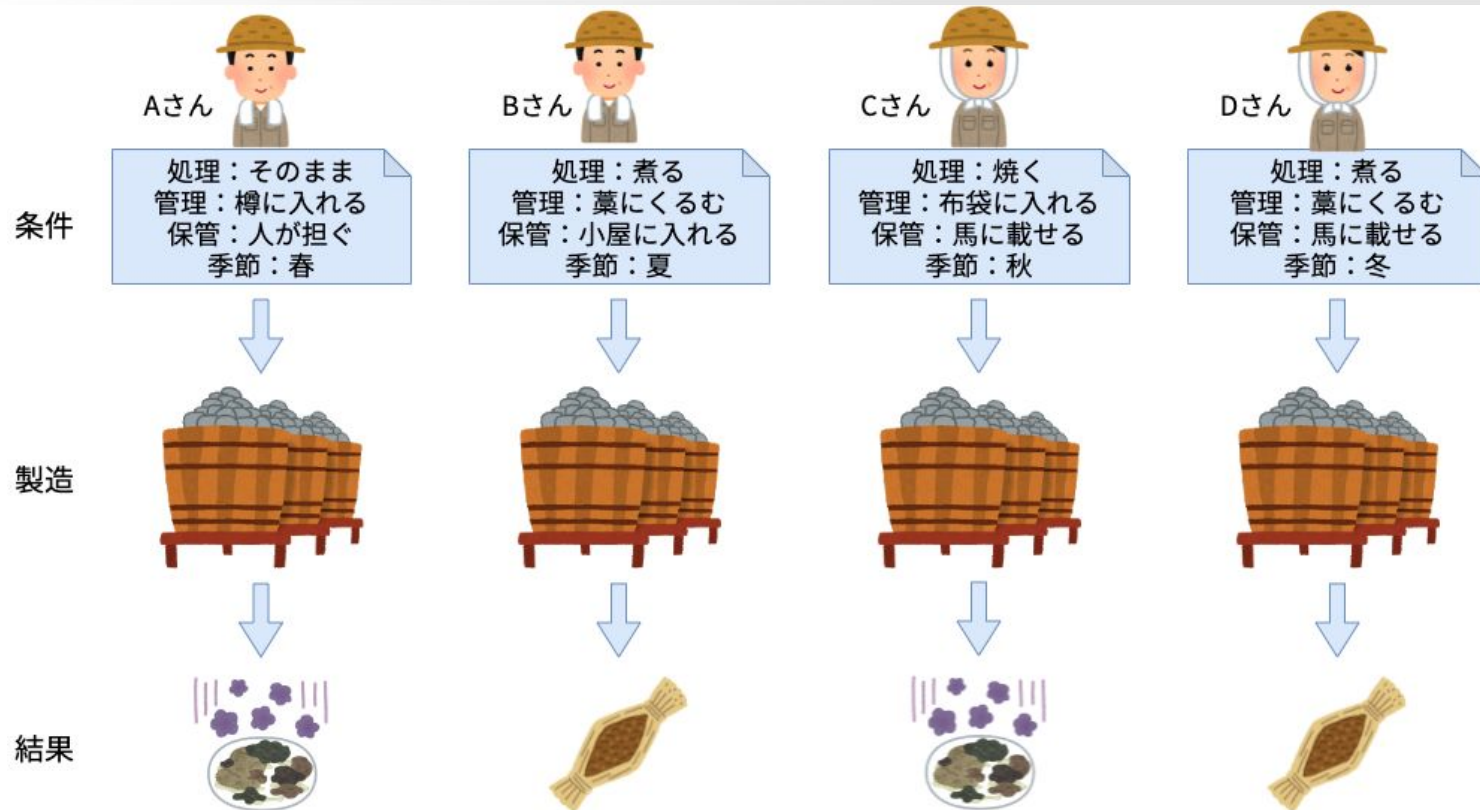
Linear  
Regression  
model

# 例え話：納豆を作る（機械学習の使い方）

## 納豆を初めて発見した人（源義家説）

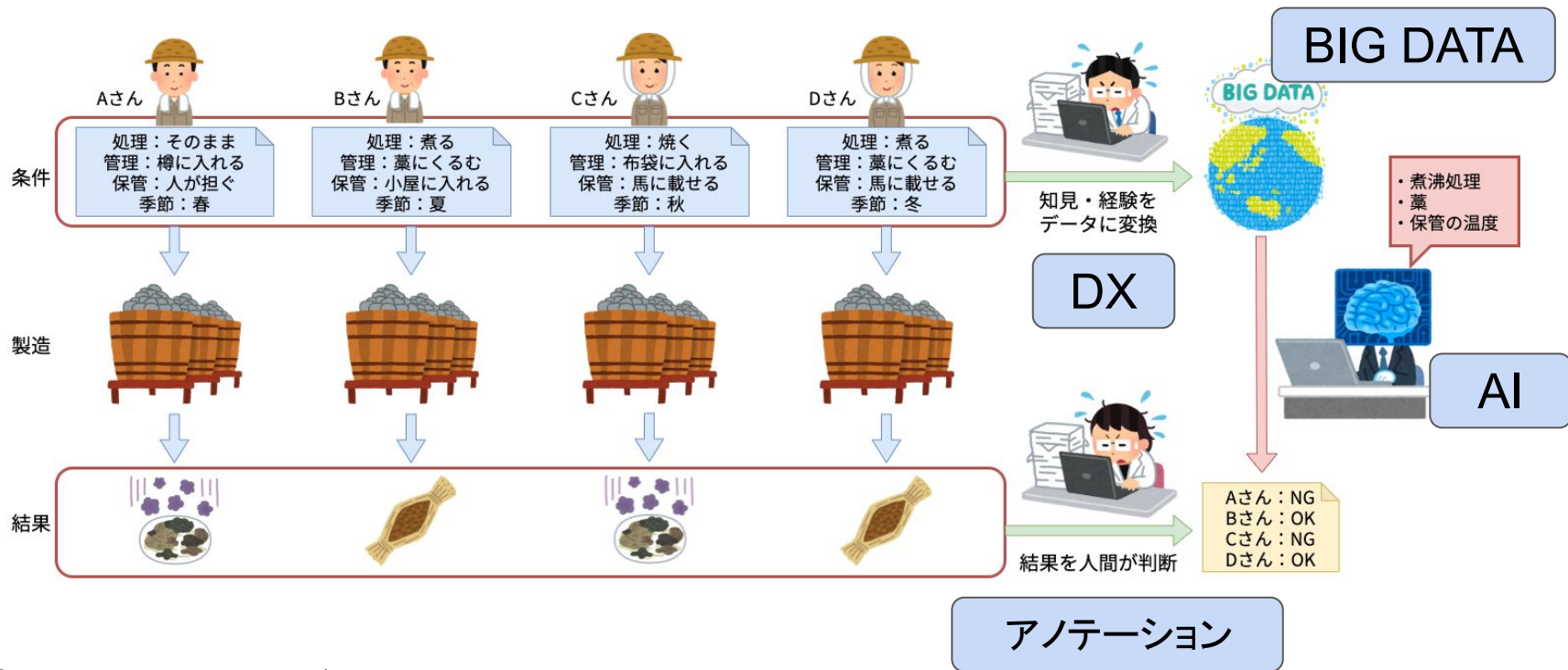
- 平安時代の中頃、東北地方で反乱があり、治めるべく平泉へ
- 農家から貰った大豆を煮ていた
- (K) 急に (T) 敵が (A) 現れた！
- 煮ていた豆を急いで藁にくるみ、馬に乗せて持ち歩く
- 落ち着いた後に藁を解いてみる
  - ネバナネバしているし、臭いも酷い
- 完全に腐ってると思うけど、もったいないから食ってみる
- うまい！
- この臭くてネバナネバする大豆をちゃんと作りたい！
  - **ビジネス課題**

# 納豆を大量生産したい



試行錯誤の繰り返し！

# 納豆作りを機械学習でやるならば



客観的な事実（データ）からルールを導き出す  
＝ベテランじゃなくても納豆が作れるように！



# 具体的に機械学習は何が出来るのか

機械学習は何が出来るのかを理解するためには、  
機械学習にはどんな種類があるのかを知る必要がある。

- データの種類
  - 数値・テキスト・画像.....
- モデルの種類
  - 線形 or 非線形、Deep or Shallow
- 目的変数の有る無し
  - 教師あり、教師なし、半教師あり
- アルゴリズムの種類
  - 分類・回帰・クラスタリング・生成

# データの種類

- 数値データ
  - データ分析（一般的な Machine Learning: ML）
- テキストデータ
  - 自然言語処理（Natural Language Processing: NLP）
- 画像データ
  - 画像処理（Image Processing）

コンピューターは数値しか扱えないので、  
テキスト・画像は何とかして数値（ベクトル）  
に変換している。

単語：Word2Vec

画素：RGB ベクトル

# モデルの種類

- 線形モデル or 非線形モデル
  - 線形：特徴量に関する1次式しか使わない
  - 非線形：それ以外を使う
- Deep Learning or Shallow Learning
  - Deep：ニューロンを模したパーセプトロンを複雑に繋げたニューラルネットワークを使う
  - Shallow：ニューラルネットワークを使わない

線形モデル	Shallow Learning
非線形モデル	
	Deep Learning

# 目的変数の有る無し

## 目的変数が無い Machine Learning もある

- 教師あり学習 (Supervised Learning)
  - 人間がアノテーションした値 (目的変数) になるよう学習する
- 教師なし学習 (Un-Supervised Learning)
  - データの分布に従って、良い感じになるよう分類する
- 半教師あり学習 (Semi-Supervised Learning)
  - データに従って学習はするけど、正解は無い
  - 風景の写真とゴッホの絵を学習して、写真をゴッホ風に変換
  - 「この写真をゴッホが描いたらこうなる」という正解は無い

# アルゴリズムの種類

機械学習は以下の4種類のことしか出来ない

- 分類 (Classification)
  - 人間がつけたラベルを学習し、未知の対象のラベルを予測する
- 回帰 (Regression)
  - 人間がつけた値を学習し、未知の対象の値を予測する
- クラスタリング
  - データの分布に従って、良い感じに分類する
- 生成 (Generation)
  - 既存のデータを学習して、新しいものを作る

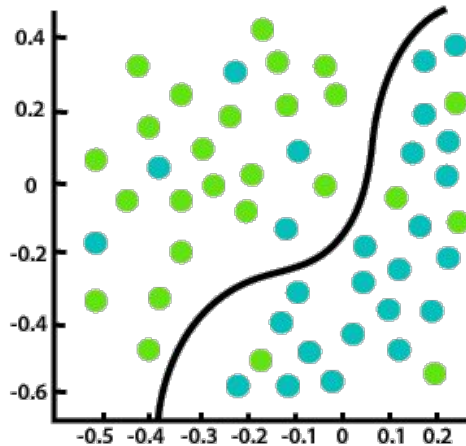
# 機械学習の種類と活用例

機械学習の種類		データ分析	自然言語処理	画像処理
教師あり	分類	レコメンデーション (例：おすすめ商品) 異常検知	テキスト分類 (例：メール仕分け)	画像分類 物体認識
	回帰	データ予測 (例：商品価格予測)	単語穴埋め問題	
教師なし	クラスタリング	セグメンテーション (例：顧客の分類)	トピック分類	
半教師あり	生成		自動翻訳	人工画像生成 (例：ゴッホ画像)

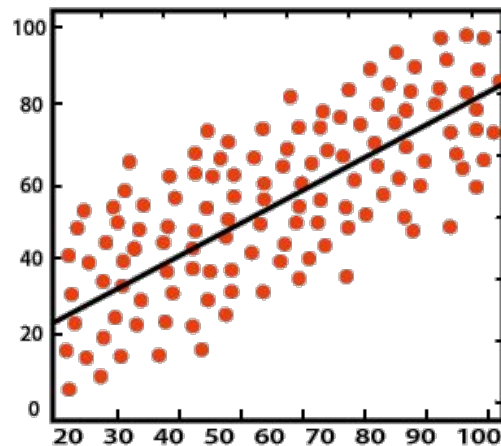
# AI は結局 2 種類

- 生成はちょっと例外なので、見なかったことに
- 分類とクラスタリングは、両方ともデータを分ける
- ということは、AI には 2 種類しか存在しない

データの無い  
隙間に  
線を引く  
(空間を切る)



Classification



データの有る  
部分に沿って  
線を引く

Regression

A decorative graphic on the left side of the page, consisting of several overlapping, flowing, translucent blue ribbons that curve upwards and then downwards, creating a sense of movement and depth.

# AI と数学



# 誰もが AI 出来る時代

- AI は今やコモデティ化し、誰もが無料で使える時代
- 10行ぐらいのプログラムを書けば、AI 出来てしまう

```
1 from sklearn.svm import LinearSVC
2 from sklearn.model_selection import StratifiedKFold
3 from sklearn.model_selection import RandomizedSearchCV
4 from scipy.stats import expon
5
6 hparams_classify_svc = {
7     'C': expon(scale=1),
8     'class_weight': [None, 'balanced'],
9 }
10 svc_classify = LinearSVC(penalty='l2', max_iter=1000, random_state=14)
11 skf_classify_svc = StratifiedKFold(n_splits=5, shuffle=True, random_state=14)
12 hpsearch_classify_svc = RandomizedSearchCV(
13     svc_classify, hparams_classify_svc, cv=skf_classify_svc, scoring='f1',
14     n_iter=10, n_jobs=-1, refit=True, random_state=14
15 ).fit([data_classify_train_X_normed, data_classify_train_y])
```

データを Stratify に分けて HyperParameterSearch をしながら  
線形 SupportVectorMachine の Classifier を学習しているプログラム

# AI に数学は必要か

- プログラムを書くだけなら、全く数学を意識しない
- しかし、モデルの中身を理解するには数学が必須
  - このモデルは何をしているのか
  - このモデルはどのような場合に使うべきなのか
  - このモデルを使う時にどのような点に注意するべきなのか
- そもそも AI はデータからルールを導き出すもの
  - 「言葉」は人間の意思を伝え理解するツール
  - 「数学」はデータの意味を伝え理解するツール
  - 数学というツール無しに Data Science は出来ない
  - 数学はデータに潜む「真理」を発見する「言葉」

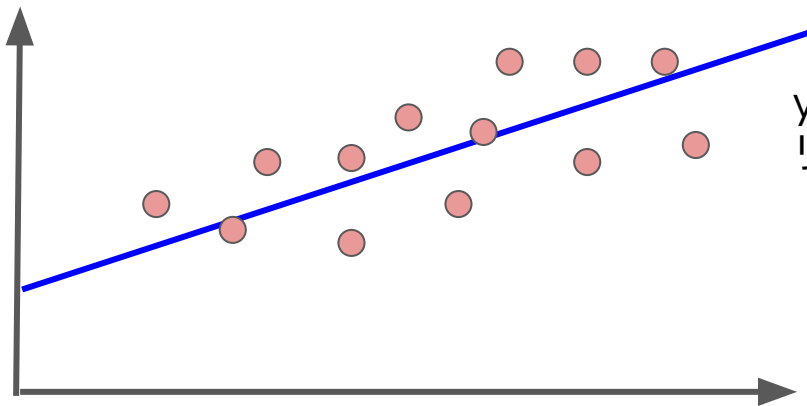
## AI から数学の必要性を訴える

- かのよう、AI（データサイエンス）には数学が必須
  - ちゃんと勉強しておけば良かったと何度も後悔してます
- 高校生、もしくは大学生で数学を勉強する際にも
  - 「この理論はどこで必要なのか」
  - 「この理論はどのようにして使うのか」ということを理解するに越したことはない
- なので、データサイエンスの基本の所からどのように数学を使っていくのか紹介します

# 1 次方程式

- 1 次方程式は AI モデル
  - 線形回帰モデル (Linear Regression model)

y: リンゴの値段 (目的変数)



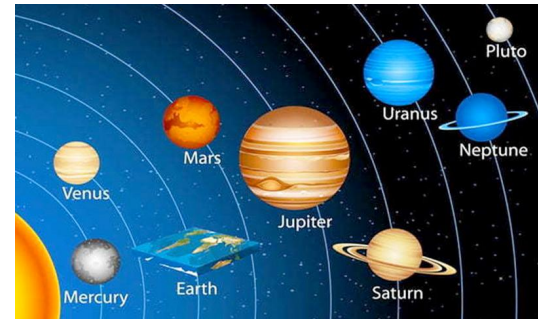
$y = a * x + b$   
リンゴの重さから値段を予測！

x: リンゴの重さ (特徴量)

2 次 (以降の) 方程式は、あまりに難しいので、それほど使いません。  
ただし、x の種類はいっぱい増えます。(線形重回帰)

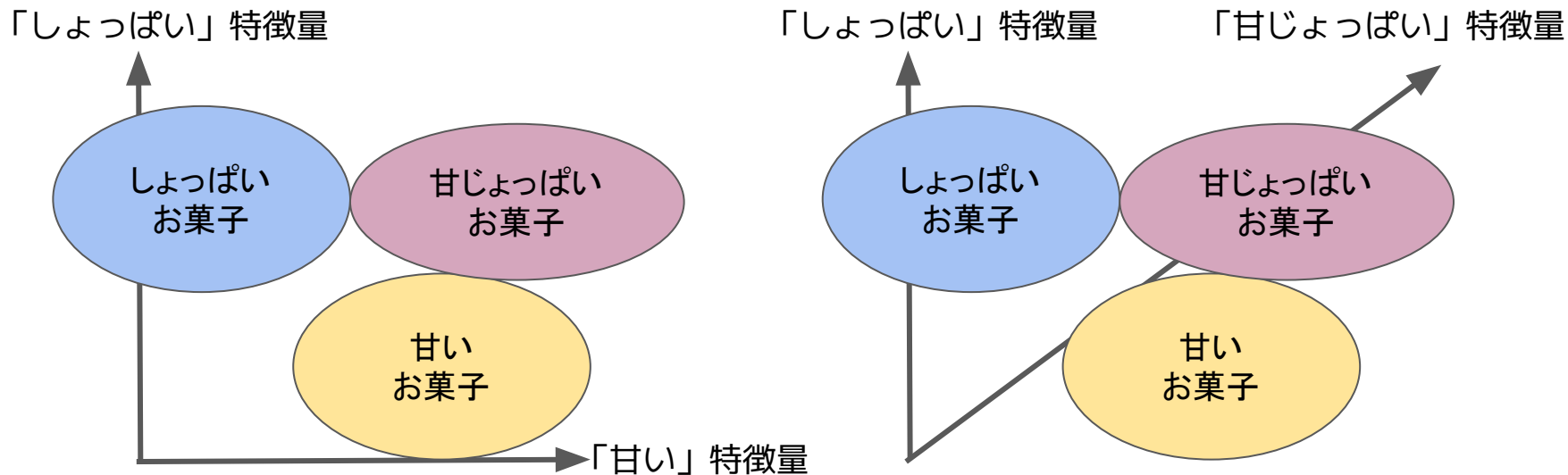
# 1 次方程式

- Deep Learning 全盛の世の中
- だが、Deep Learning は Black Box
  - なぜその予測をしたのかが簡単に説明・理解できない
- Explainability という側面が重要視されてきている
  - Deep Learning の予測の理由をいかに明瞭に説明するか
- 一説に、人間が理解できる限界は 1 次方程式まで
  - 人間は線形なものしか理解できない
- Deep Learning の結果をわざわざ 1 次方程式で説明
- 因果推論も 1 次方程式



地球平面説

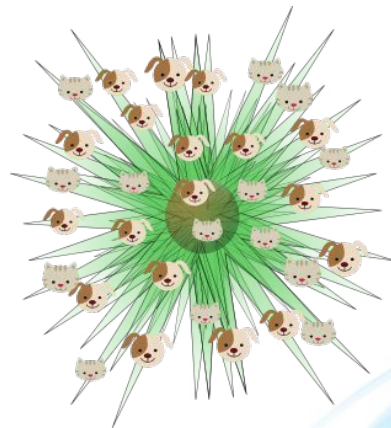
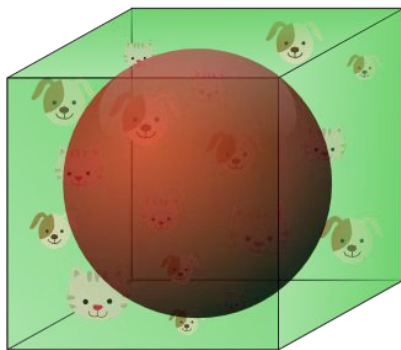
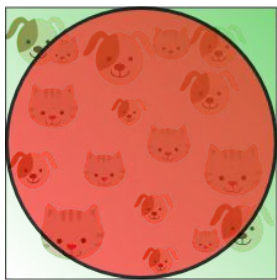
# 空間と軸・次元



軸 = 特徴量 = 分析対象を特徴づける要素  
軸が直交する = 意味が重ならない = 「良い」特徴  
特徴量の種類はいっぱい増えるので、高次元空間になる。

# 次元の呪い

- N次元の直方体とそれに内接する球があったとする
  - 2次元なら正方形と円、3次元なら立方体と球
- 立方体に満遍なく点があった時に、どれぐらいが球の中にあるか
  - 2次元の場合は、ほとんどが円の中に入るだろう
  - 3次元の場合は、2次元に比べて隙間が多いので、球からはみでる
- ではもっと次元数が大きくなったら？
  - 無理やり球で考えると、立方体はトゲトゲのウニみたいな形に



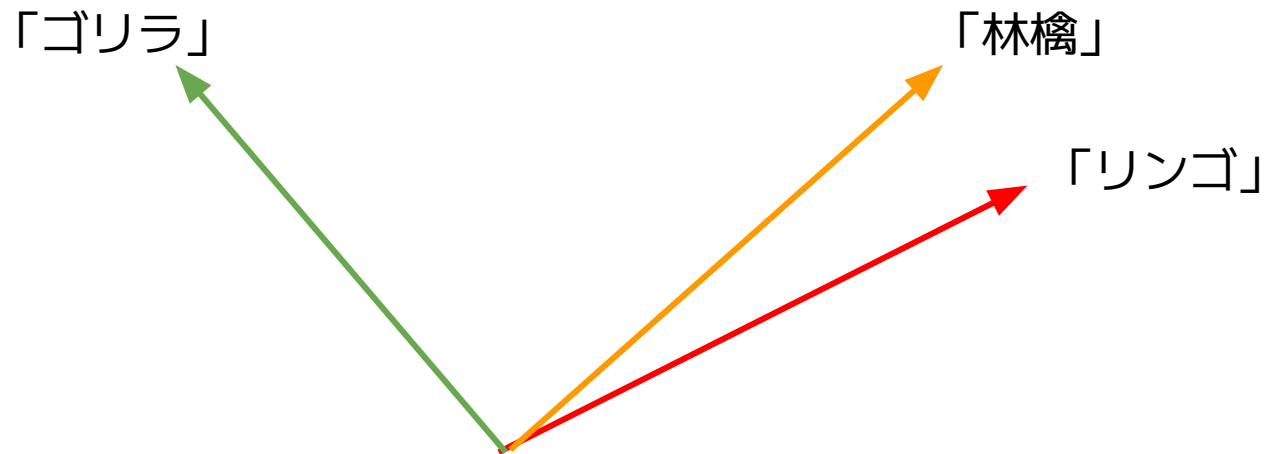
## 次元の呪い

- データの量があっても、あまりにも特徴量の数（次元）が多いと、そのデータはほとんど空間のすみっこに存在し、疎（sparse：スカスカ）な状態になる
- 空間をスパッと2つに分けるだけ（分類）でも、ちょっとでも切る位置や角度が違うだけで全然違う結果になってしまう
  - 微妙な誤差が結果に深刻な影響を与えてしまう
  - 空間をぐねぐねとした曲線で分ける非線形モデルはもっとヤバイ
- 特に NLP（自然言語処理）は高次元になりがち
  - NLP の Shallow Learning は、大概、線形モデル



## 角度と意味

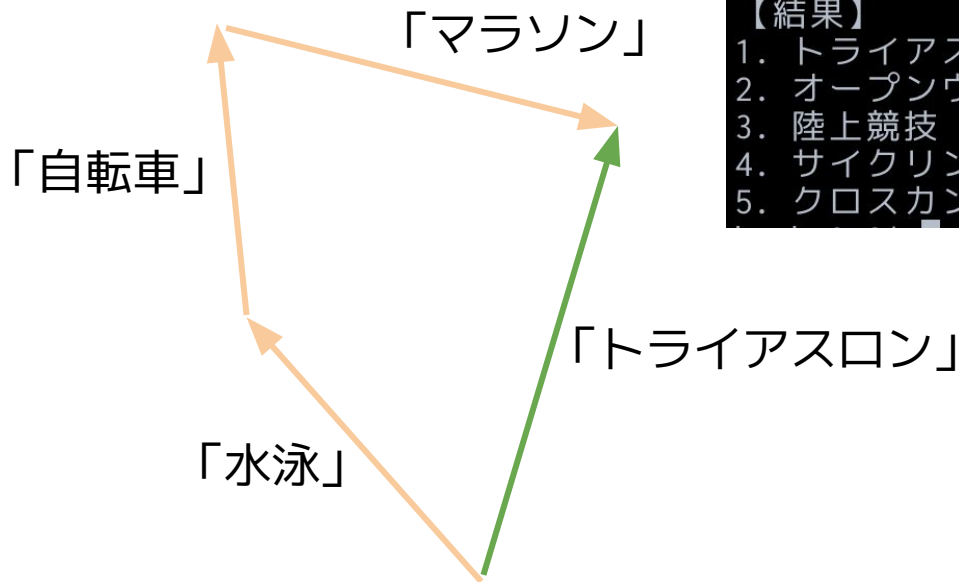
- Word2Vec は単語をベクトルに変換する
- 「単語の意味が近い」 = 「ベクトルのなす角が小さい」



単語が出現する前後関係から意味の近さを学習し、これを高次元空間内のベクトルとして埋め込む

# ベクトルの足し算

- 単語をベクトルに変換したので、足し算が可能に
- ベクトルを足し算すると、意味が足し算される



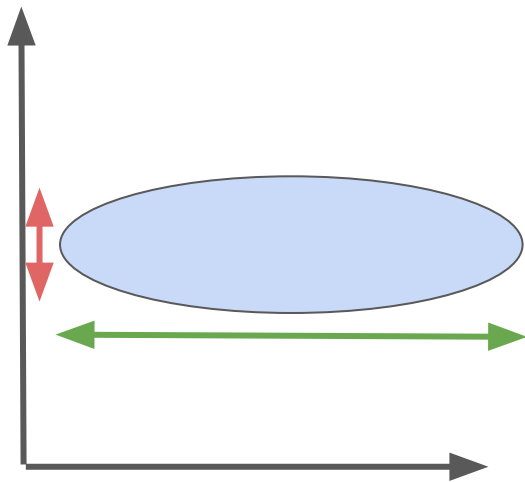
```
bash-3.2$ ./w2v.py 水泳 自転車 マラソン
kv_fasttext_jawiki_20221201.bin
【結果】
1. トライアスロン : 0.7410668134689331
2. オープンウォータースイミング : 0.7090177536010742
3. 陸上競技 : 0.7036643028259277
4. サイクリング : 0.698056161403656
5. クロスカントリー競走 : 0.6872199177742004
```

日本語Wikipedia を学習した  
Word2Vec で実際に計算した結果

# 分散の意味

- 分散が大きい = 色々な意見があるので、  
予測の参考になる、良い特徴
- 分散が小さい = みんな同じ意見（平均値）だけなので、  
予測の参考にならない、悪い特徴

「しょっぱい」特徴量

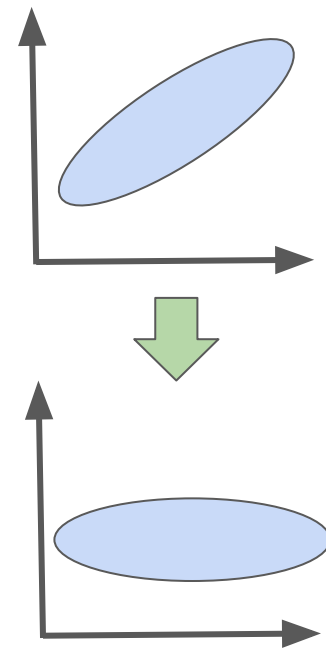


課題：お菓子の味を調査して、  
売れるお菓子を予測したい

- 色々な甘さのお菓子を調査したので、  
甘いことで売れるか売れないか  
分析が可能
- 同じ位のしょっぱさのお菓子しか  
調べてないので、参考にならない

# 主成分分析 (Principal Component Analysis: PCA)

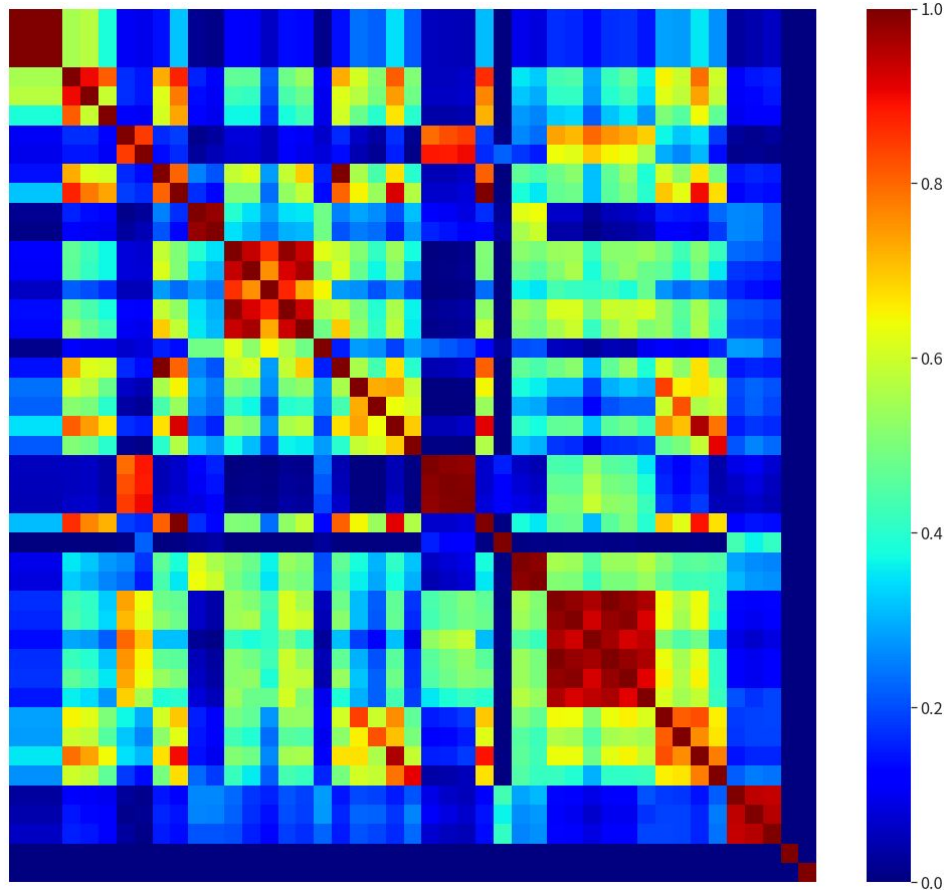
- 分散が大きいほうが良い特徴量
- データを回転させ、分散を大きくする
- 回転 = 行列演算
- 固有値  $\propto$  分散 = 意味の幅
- 固有ベクトル = 意味の幅が大きい特徴
- データを回転させるということは、  
軸を回転させているということ
- 要するに「物の見方を少し変える」方法



# 相関

- 目的変数と特徴量の相関が高い ← 良い事
  - その特徴量を使えば、目的変数を予測しやすい
- 特徴量間の相関が高い ← 悪い事
  - 同じ意味の事を2回言っている
  - マルチコ（多重共線性：multi-Collinearity）
  - モデルの中には逆行列を計算するものが沢山ある
  - 相関が高い特徴量が複数あると、  
逆行列を計算する際に行列のランク落ちを起こし、  
予測結果が不定値になる
  - 対処法：PCAなどで特徴量間の相関を無くす
    - 物の見方を少し変えて、意味のある特徴を使う

# 相関のヒートマップ

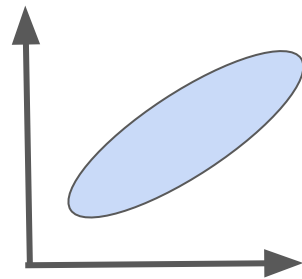
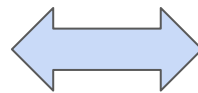


データ分析の基本は  
とにかく何でも可視化！

# データサイエンスは全数学が必要

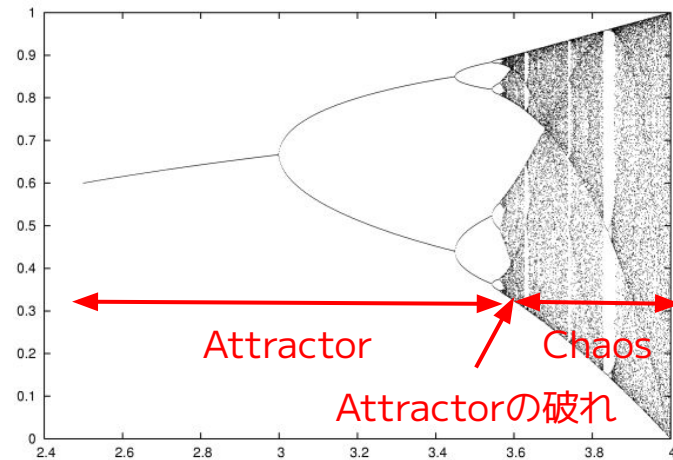
- 数学 1 : 代数的な考え方 (モデル)
- 数学 2 : 幾何的な考え方 (グラフ)
- 数学 3 : 確率的な考え方 (学習)
- このような縦割りの考え方はあまり良くない
- 数式と図形を紐付けて考える力が必要

$$r_{xy} = \frac{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2} \times \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \bar{y})^2}}$$



# 私が考える数学の美しさ

- 私の大学の専攻は複雑系数理（カオス）でした
- 代表的なもの：ロジスティック写像
  - $X_{n+1} = aX_n(1-X_n)$
- 単純な数式が複雑な挙動を表す
- 生物の多様性を表す
  - かもしれない
- 真理は簡単な式で表現できる
  - かもしれない
  - だとしたら、美しい！

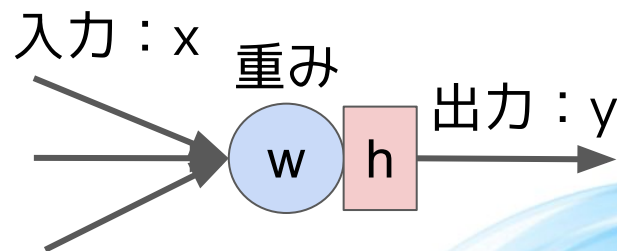
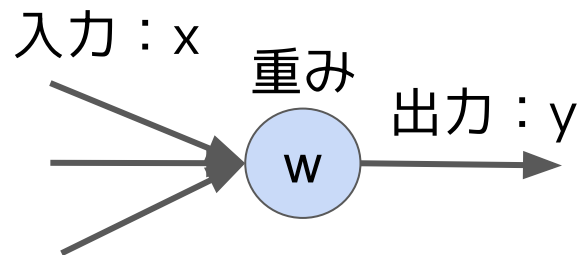
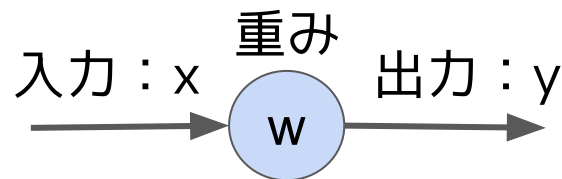


対称性の自発的破れと相転移に似ている



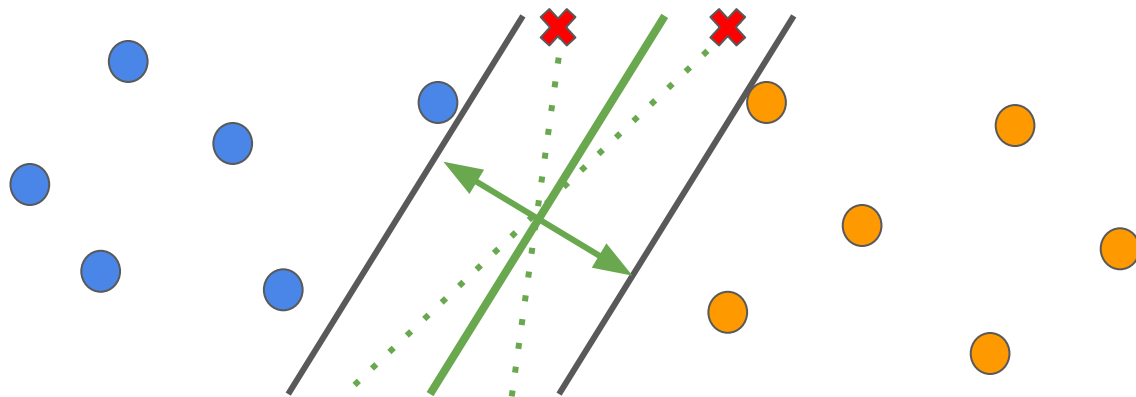
# AI モデルも美しい

- 基本は線形回帰
- 特徴量の種類が増えて、線形重回帰
- 活性化関数をつけると、パーセプトロン
  - ニューラルネットワークの基本素子
  - 重みの学習は Back Propagation = 微分



# AI モデルも美しい (Cont.)

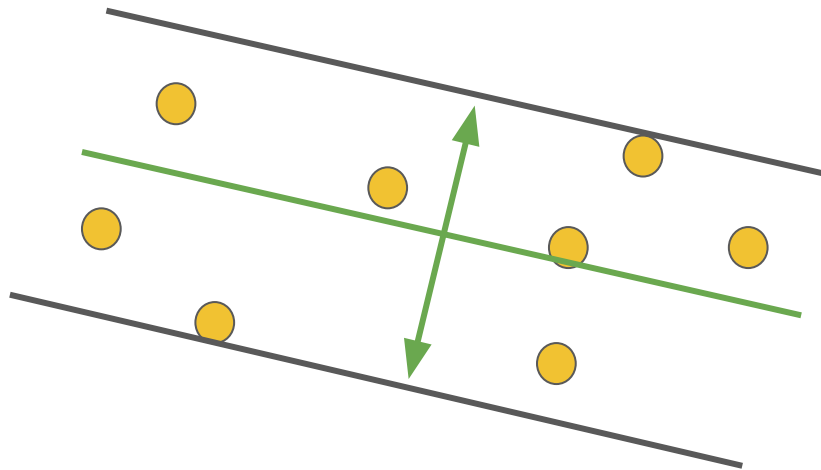
- パーセプトロンを分類に使ったものが、  
Support Vector Machine



1本の線で区切るのではなく、幅（マージン）を持った線で区切る  
マージンが最大となる線が最も良い区切り線（マージン最大化）  
この導出が美しい！（双対問題・ラグランジュ乗数）

## AI モデルも美しい (Cont.)

- Support Vector Machine は分類モデル
- しかし、回帰にも使える
  - スラック変数を導入し、再びラグランジュ乗数を用いて、Karush-Kuhn-Tucker条件の下、双対問題を解く
  - この導出も素晴らしく美しい  
(何回やっても覚えられないけど)

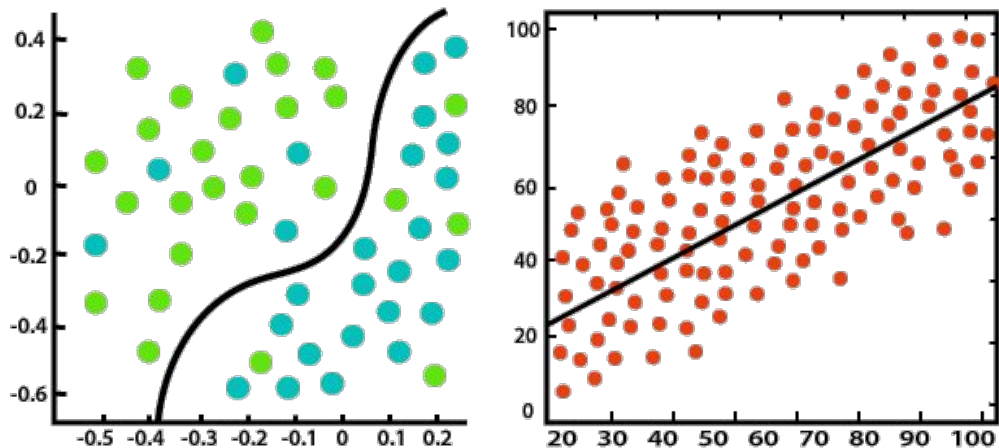


マージンが最大となる  
線を引く！

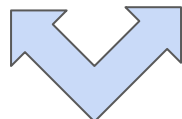
[参考：パターン認識と機械学習](#)

# AI は結局 1 種類

- AI は、線を引きます。
  - なんて単純！なんて美しい！



Classification



Regression

結局は同じものだった

A decorative graphic on the left side of the slide, consisting of several overlapping, flowing, light blue and white wavy lines that create a sense of movement and depth.

# AI と数学と教育

# AI の種類毎に難しさが異なる

機械学習の種類	
教師あり	分類
	回帰
教師なし	クラスタリング
半教師あり	生成

易  
↑  
↓  
難

慎重に設計・学習すれば  
ビジネスで利用できる

高い技術力を持つ専門家が  
数ヶ月努力すれば  
ビジネスで利用できる可能性がある

高い技術力を持つ専門家が  
数ヶ月努力しても  
ビジネスで使えないことが多い

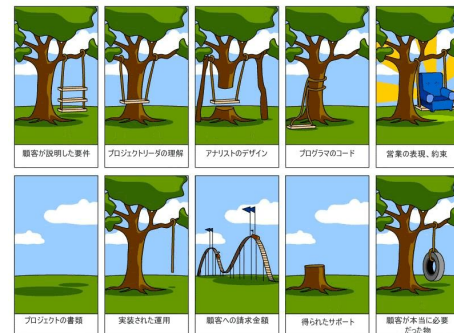
非常に高い技術力を持つ専門家が  
年単位で努力して億単位のお金をかけて  
使える兆しが見えることがある

# 例え話：中古車の販売価格を AI で決める

- AIを用いる意義
  - 価格査定の人的コストを削減する
  - 価格査定担当者のスキルに依存しない、市場と連動した価格設定
- 「回帰」を使おう
  - 価格を正確に予測することは非常に困難 → 頓挫
- 正確な販売価格は本当に必要だったのか
  - 価格帯を示すだけでも、人的コスト削減の効果は十分
  - 同様の中古車の、最新の市場価格を簡単に参照出来れば良い
- 方針の再策定
  - 価格の「回帰」ではなく、価格帯の「分類」を行う
  - システムで市場価格の参照を行うなど、実装と運用でカバー
- データとノウハウを蓄積し、将来の完全AI化を目指す

# AI が大事なのではなく、AI の使い方が大事

- 顧客が本当に必要なものを汲み取る
- それには何をすれば良いのかを考える
- もっと簡単に実現する方法を考える
  - 回帰よりも分類
  - AI より、なんちゃって AI (AI を使わない)
  - MVP([Minimum Viable Product](#)) を作る
    - スタートアップ企業の思考 ([Lean Startup](#))
- 「AI デザイナー」が世界的に圧倒的に足りていない
  - AI はコモデティ化して自動化も進んでいる ([AutoML](#))
  - 人間が使えるように設計する仕事は、無くならない





# AI をデザインするために必要なもの

- 数学・AI の知識
  - 論理的思考
- 顧客が本当に欲しいものを考える能力
  - 理解力・コミュニケーション能力
- さまざまな可能性を考えて、試行錯誤する能力
  - 地道な努力を厭わない心・柔軟な思考
- 考えたデザインを仲間・顧客に伝え、理解を得る能力
  - プレゼンテーション能力・コミュニケーション能力
- これらがあったら、別に AI じゃなくてもやっていける
- でも、AI や数学はこれらを養うに良い分野

# 提案：こんな授業はどうだろうか

- Public Data を持ってくる
  - この「データを探す」のがとても大変（特に日本では）
    - [UCI Machine Learning Repository](#)
    - [Kaggle Dataset](#)
- 例：[Insurance Premium Prediction](#)
  - age: 年齢
  - sex: male or female
  - bmi: BMI 値
  - children: 子供の数
  - smoker: yes or no
  - region: southwest, southeast, northwest, northeast
  - charges: 恐らく保険金額

## 提案：こんな授業はどうだろうか（Cont.）

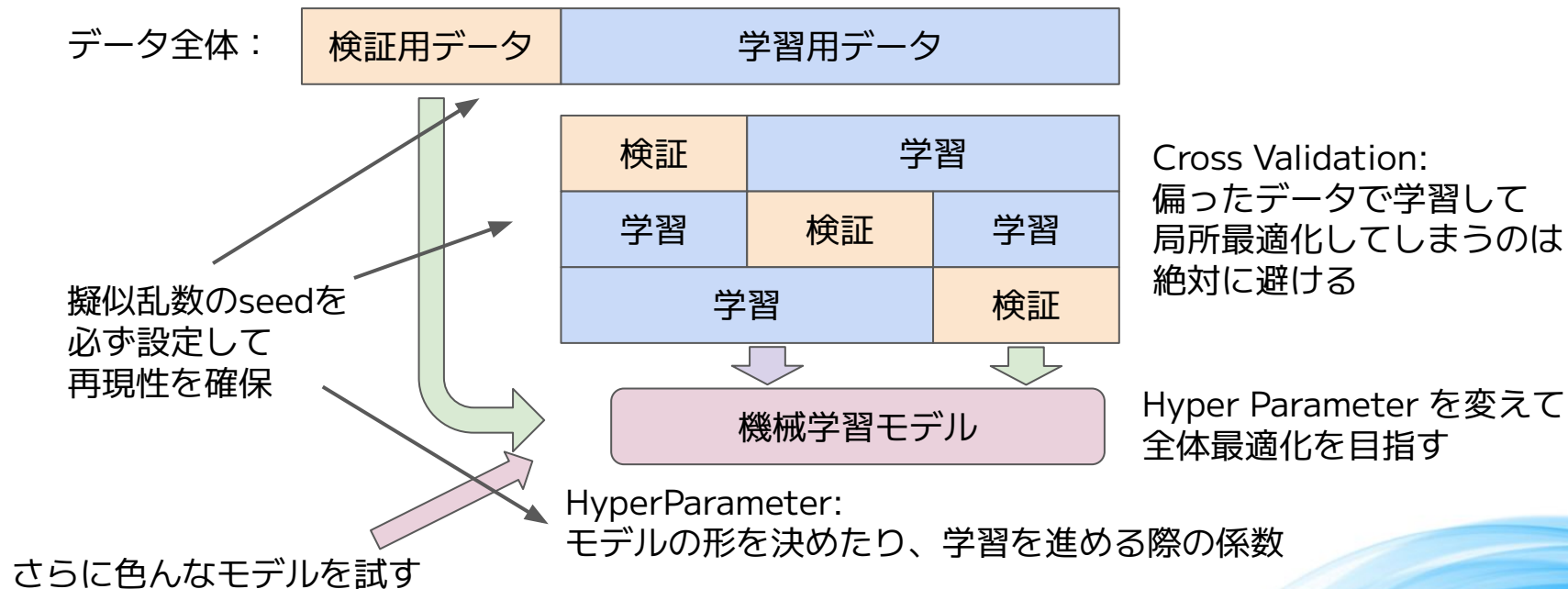
- この項目のうち、どれかひとつを目的変数に設定し、「その予測が必要な」カバーストーリーを考える
  - age が目的変数：予測した年齢との差分が不健康度
  - smoker が目的変数：喫煙者の傾向を知る
  - bmi：体型の予測をする
    - 顧客が本当に求めるものを考える訓練
- 何通りもの方法を考えて、予測する
  - 分類？回帰？クラスタリングして分類問題に？
    - 柔軟な思考能力の訓練
  - 正しい方法で予測してる？過学習は防いでる？
    - 純粋な機械学習の訓練

## 提案：こんな授業はどうだろうか？（Cont.）

- 結果を可視化し、プレゼンテーションする
  - どんなグラフを描く？それは正しく伝わる？
    - プレゼンテーション能力の訓練
  - 正しい評価指標を使っている？
    - 統計的思考の訓練
- 以上をグループで協力して行う
  - コミュニケーション能力の訓練
- これらを、本当のデータと依頼者で行えると最高
  - 私が業務としてやっていることに他ならない（笑）

# 余談：データサイエンスは「サイエンス」だ

- 機械学習する時に最も気を付けるのは「再現性」と「過学習」(例：マンモスの牙)



# 数学・AI 教育に求めるもの

- これら能力の訓練を通じて、社会に通用する個人の育成
  - 横断的な数学の能力、論理的思考能力
  - 柔軟な思考能力、地道な努力を続けられる能力
  - コミュニケーション能力、プレゼンテーション能力
- 最終的には、「リテラシー」を upbringing すること
  - 統計的に正しいものの考え方
    - 数字に騙されない思考
  - データサイエンス・AI を活用できる社会の基盤
    - AI 以前に重要なことがまだまだある
  - データサイエンス・AI というものへの理解
    - データサイエンス自体がコモデティ化する必要

# 数字に騙されないために

- ある意味 AI よりも重要な「指標」
  - 回帰の場合は平均2乗誤差かそれに類するものしかない
  - 分類の場合は、実はいっぱいある
- 一般的に、予測したラベルが正確かどうかを測るのは精度 (Accuracy) だと思われている
- しかし、精度は使ってはいけない
  - 例) あまり多くの人がかからない病気を予測する  
1000人に1人 = データが不均衡 (Imbalance)
    - 何もせずに「その病気ではない」と出力すれば  
精度は 99.9 %

# 混合行列 (Confusion Matrix)

混合行列		予測結果		指標
		目的のラベルを持つ	目的のラベルを持たない	
実際の状態	目的のラベルを持つ	True Positive TP	False Negative FN "予測漏れ"	$\frac{\text{Recall}}{\text{TP} + \text{FN}}$
	目的のラベルを持たない	False Positive FP "予測過多"	True Negative TN	
指標		$\frac{\text{Precision}}{\text{TP} + \text{FP}}$		$\frac{\text{Accuracy}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$

Precision: 予測過多がリスクになる場合に使う指標：違反取引者の予測

Recall: 予測漏れがリスクになる場合に使う指標：ガン患者の予測

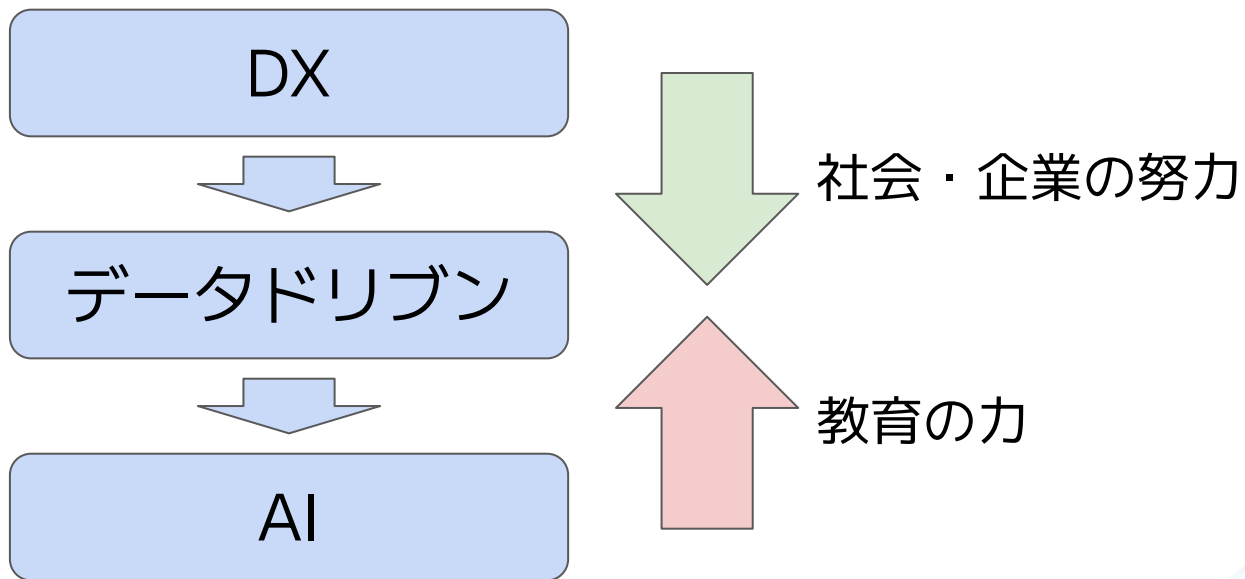


# AI の前に必要な「基盤」

- DX (Digital Transformation) (総務省)
  - すべてのデータがデジタル化されていなければならない
  - AI が学習するための Big Data (特徴量) が必要
  - アノテーション=人間にとって正解とは何か (目的変数)
    - AI はあくまで「人間にとって良い状態を予測する」ツール
    - ここにも人間の力がまだ全然必要
  - IT の力とお金が必要
    - 正直、全然進んでない (お金が儲からないから)
  - 生まれたデータを集約・整理・準備するデータパイプライン
    - これを作る人 (Data Engineer) も全然いない

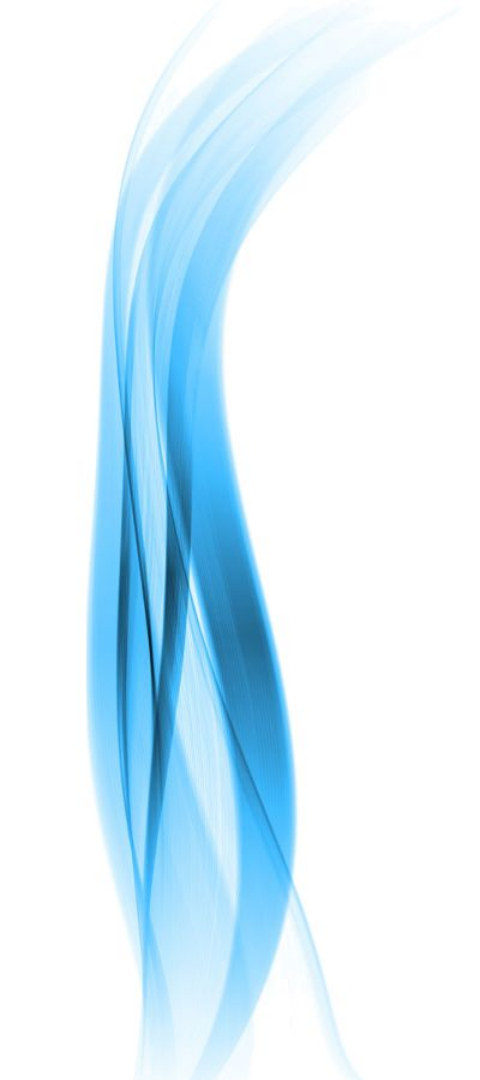
# AI の前に必要な「基盤」 (Cont.)

- データドリブン (Data Driven) な思考
  - データという客観的な情報を見て判断できること
    - ワークマンは商品を変えずに売り方を変えただけでなぜ2倍売れたのか
  - この思考を浸透させるために、教育の力が必要なのではないか



# データサイエンスがコモデティ化する未来

- AI を作るプログラムはコモデティ化した
- データサイエンスという思考はまだコモデティから遠い
  - まだ周囲の理解を得ることは難しい
- しかし、遠くない未来、誰もがデータサイエンスの  
エッセンスを持って考え、行動するのではないか
  - データサイエンティストなマネージャー
    - 業務判断に AI を使う
  - データサイエンティストな経営者
    - 経営判断に AI を使う
  - もちろん、データサイエンティストな教師



ありがとうございました