社会人向け講座「データ分析者養成コース」

機械学習の概要

鈴木大慈
東京大学大学院情報理工学系研究科数理情報学専攻
理研AIP

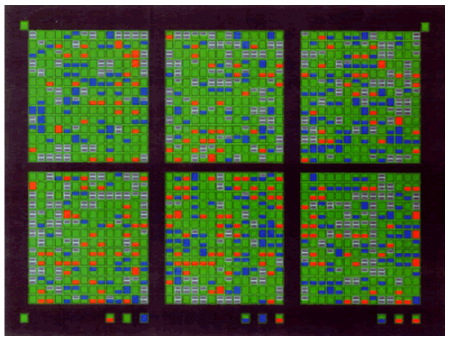
2018年11月1日

身近にあふれる機械学習

検索エンジン

推薦システム

遺伝子データ解析



音声認識

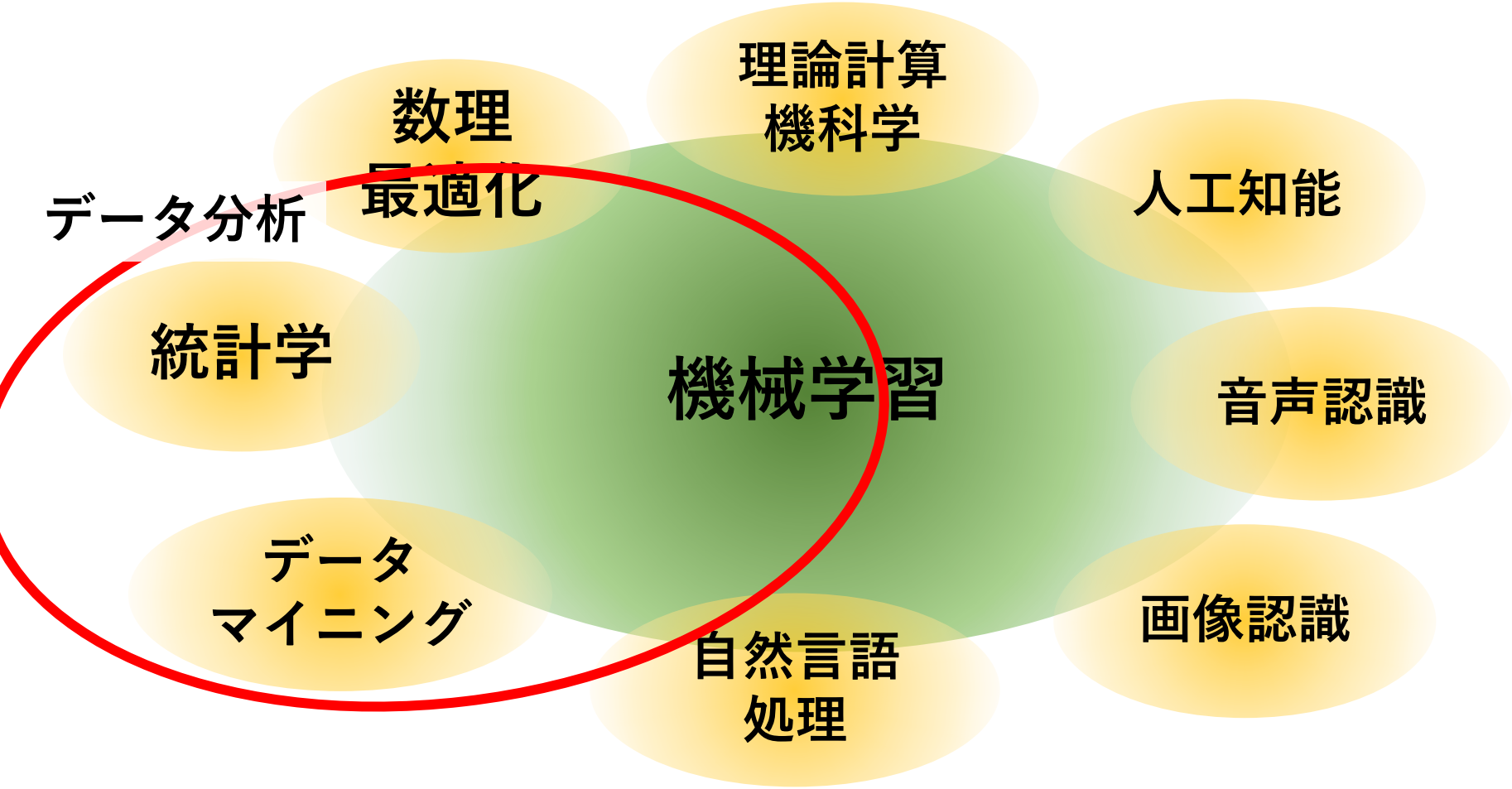
機械学習プラットフォーム

Google Cloud Platform

Microsoft Azure

amazon web services™

機械学習の立ち位置



さまざまな分野の複合領域

機械学習コミュニティの実体

機械学習主要国際会議

NIPS (Neural Information Processing Systems)
ICML (International Conference of Machine Learning)
COLT (Conference of Learning Theory)
ICLR (International Conference on Learning Representations)
AISTATS, UAI, ECML, …

関連国際会議

- データマイニング
  KDD, ICDM, WWW, WISDM, SIGIR, SDM
- 人工知能
  IJCAI, AAAI
- コンピュータビジョン
  CVPR, ICCV, ECCV
- 自然言語処理
  ACL, NAACL, EMNLP, COLING

NIPS2015@Montreal

ICML2016@NYC

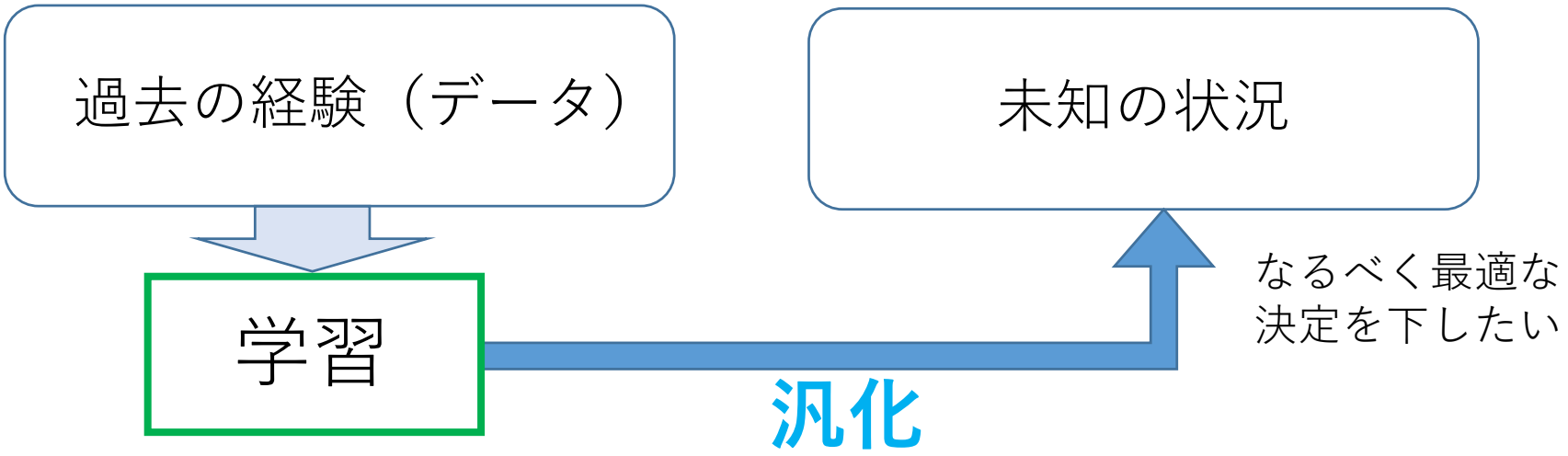
本日の内容

- データからその裏にある法則を見つける
- 教師あり学習
  - 線形回帰
  - 線形判別
  - 決定木，勾配ブースティング
- 教師なし学習
  - クラスタリング：トピックモデル
  - word2vec
  - 関係データ解析
  - 異常検知
- 深層学習
  - 物体認識・物体検出
  - 生成モデル

機械学習の目的

- 人間と同様の知的情報処理を計算機で実現するための技術・手法

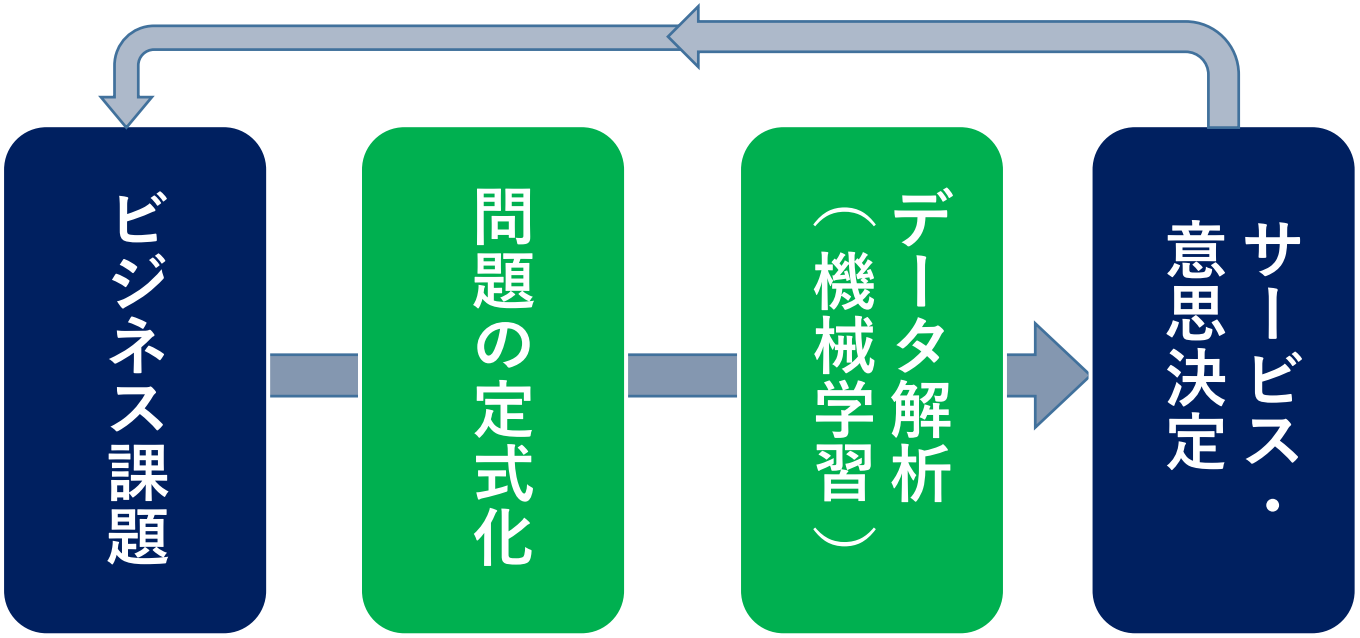
Arthur Samuel 「Field of study that gives computers the ability to learn without being explicitly programmed 」（1959）



- 本来は人工知能の一分野
- 人工知能に統計的アプローチが有用であることから，
  データ解析手法としても発展

機械学習のビジネス利活用

- 目的に応じて手法を選択する必要



- まずは問題を明確化
- 「本当に機械学習が必要か？」
- 簡単な方法での解決方法を第一に探索

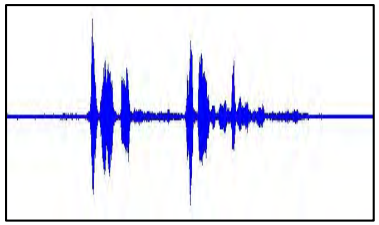
各種機械学習手法で何ができるのか？
→仕組みを把握する重要性

統計と機械学習

予測

（より機械学習的）

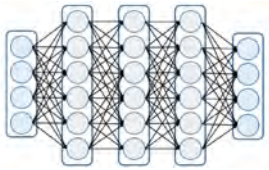
➡ 船



➡ “Hello”

- Outcomeを正しく当てる.
- 解釈よりも予測精度を重視.

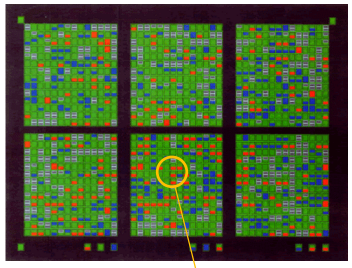
例：深層学習



構造が複雑
解釈可能性に難

推測

（より数理統計的）



⟷ 肺癌

第〇〇遺伝子が肺癌に寄与
有意水準5%

- 原因の究明.
- 仮説検定は典型例.

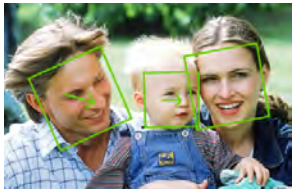
例：線形回帰分析

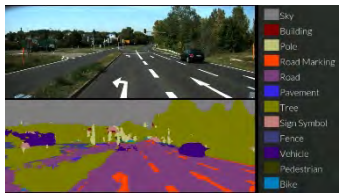
構造が単純

統計的学習の考え方

- データから裏にある法則（パターン）を自動的に見つけてもらう．見つけ方の方法が「機械学習」．

- 強い将棋ソフトを作りたい → 大量の棋譜データで学習
- 顔認識ソフトを作りたい → 大量の画像データで学習
- 車道を認識したい → 大量の車載カメラ画像で学習





機械学習の良い点

- 大規模データ
  - 人が処理できない量のデータを扱える（次元,量）
- 計算速度
  - 人よりも速く情報処理できる
- ２４時間稼働
  - 休まず動き続けられる

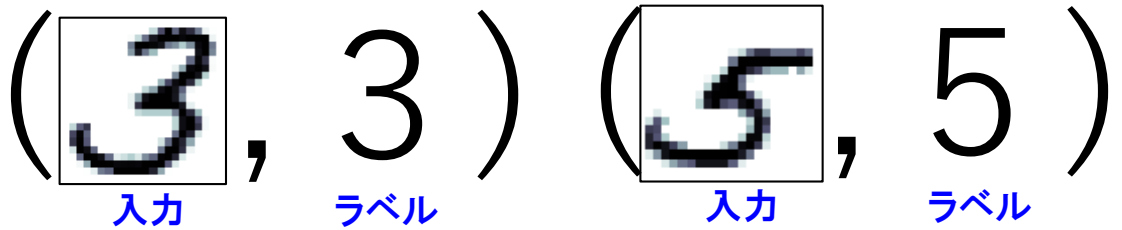
一方，データが十分でない問題や人間が手を動かせばよい問題には合わない.

機械学習の問題設定

- **教師あり学習**：

  データ：(x,y)←ある入力xとそれに対するラベルyの組

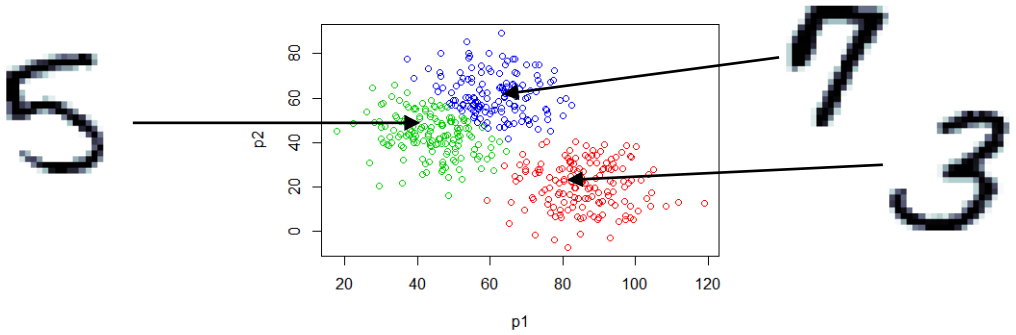
  問題の例：回帰，判別



- **教師なし学習**：

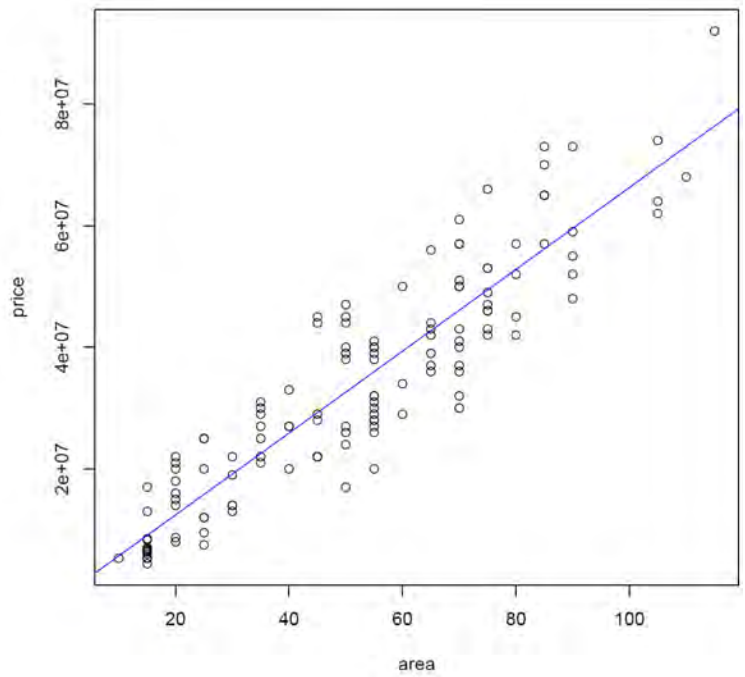
  データ：(x) ←ラベルがない

  問題の例：クラスタリング，音源分離，異常検知

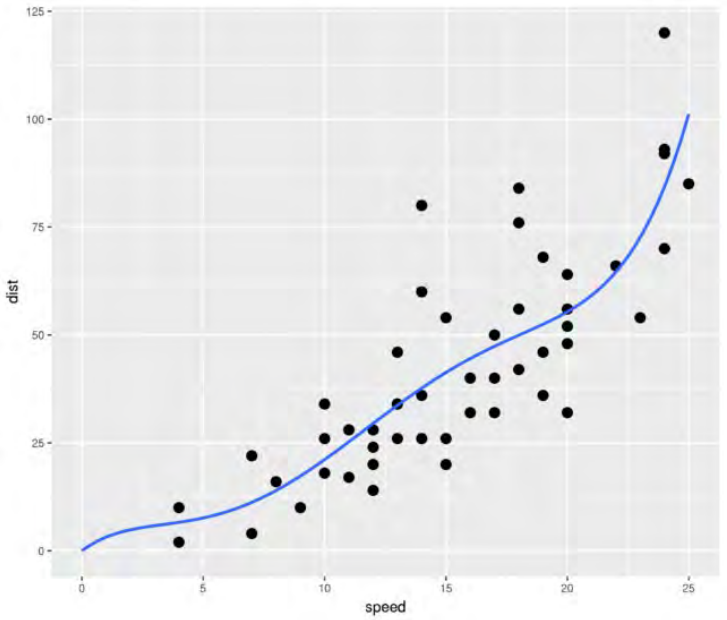


- **半教師有り学習**：ラベルの付いているデータと付いてないデータが混在

回帰（教師あり学習）



線形



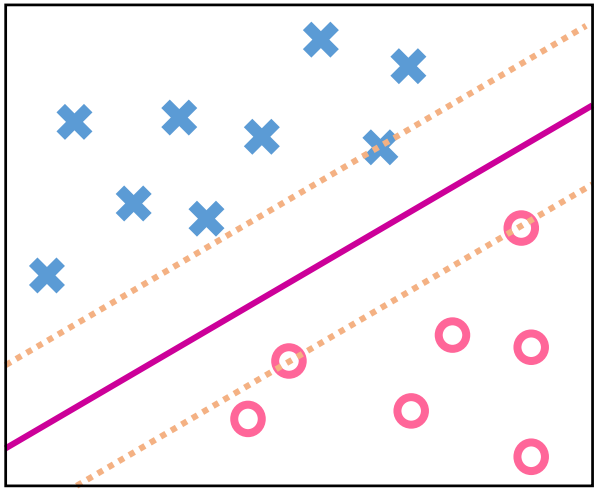
非線形

入力xから実数の出力yを予測

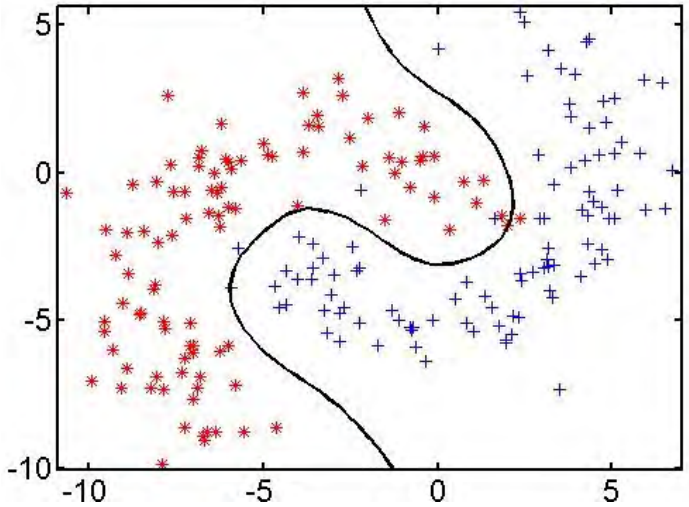
- 線形
- 非線形

例：マンションの床面積(x)→価格(y)，気温(x)→飲料の売り上げ(y)

判別（教師あり学習）



線形



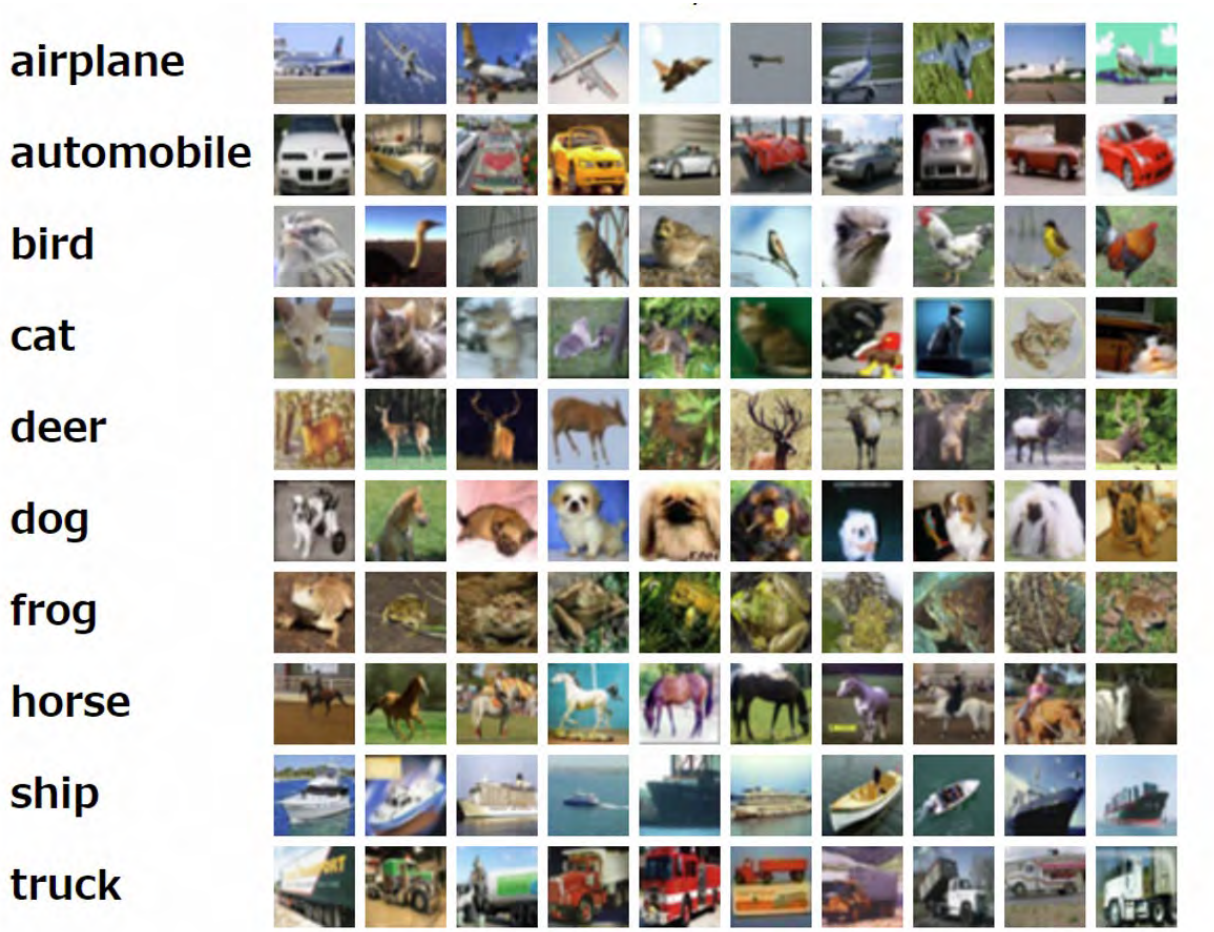
非線形

入力xからカテゴリーの出力yを予測

- 線形
- 非線形

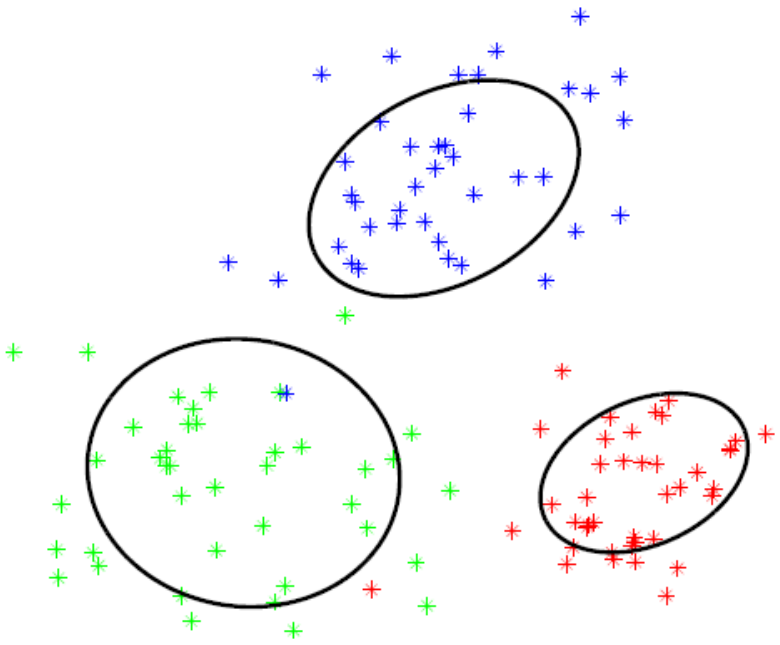
判別の例

- 画像認識

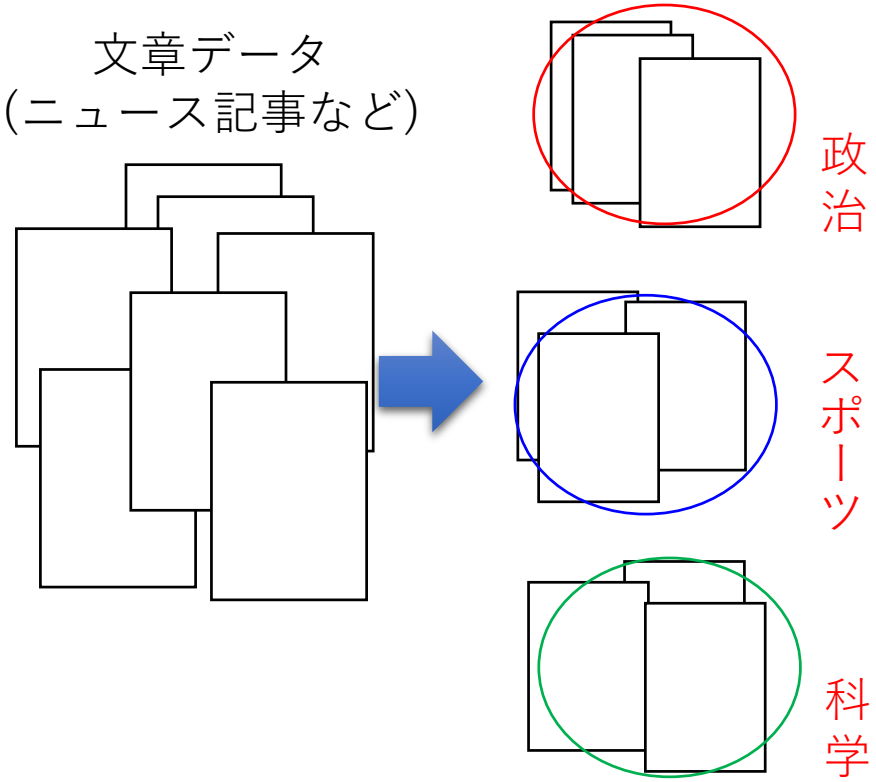


- 音声認識，疾病の予測

クラスタリング（教師なし学習）



混合ガウス分布によるクラスタリング



トピックモデル

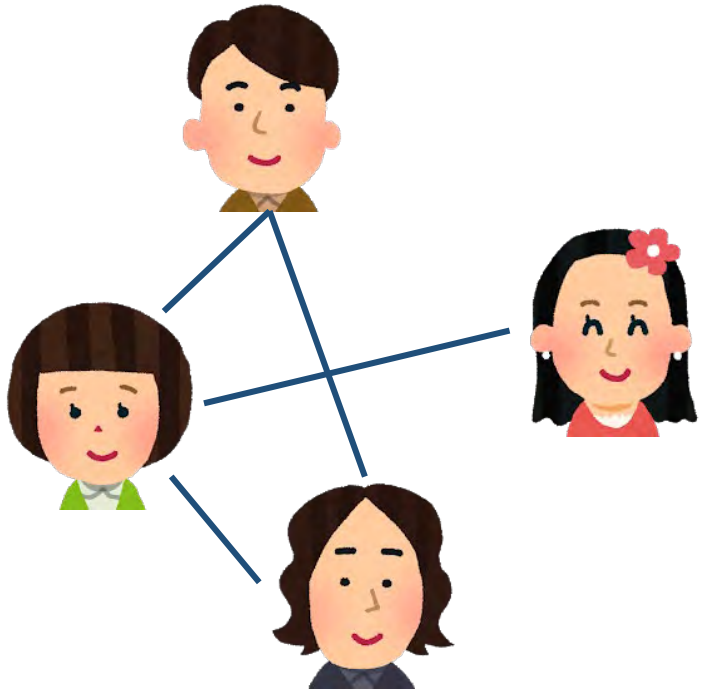
- 文章分類

関係データ解析（教師なし学習）

	映画 A	映画 B	映画 C	・・・	映画 X
ユーザ 1	4	8	4	・・・	2
ユーザ 2	2	4	2	・・・	1
ユーザ 3	2	4	2	・・・	1
⋮					

推薦システム

- 映画・商品の推薦
- 表示広告の最適化

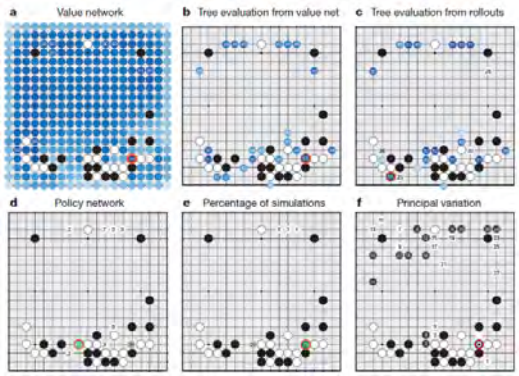


SNS解析

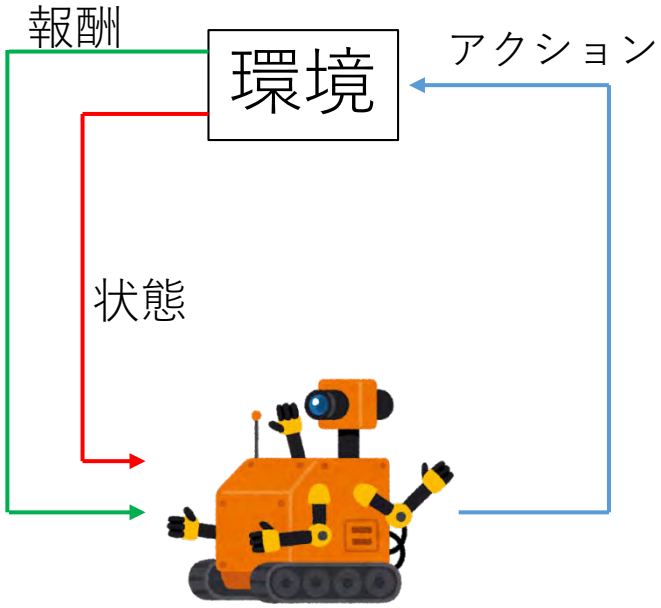
- 友人コミュニティの検出
- つぶやきからの趣味趣向の推定

強化学習

Google research blog, 8/March/2016.
“Deep Learning for Robots: Learning from Large-Scale Interaction.”



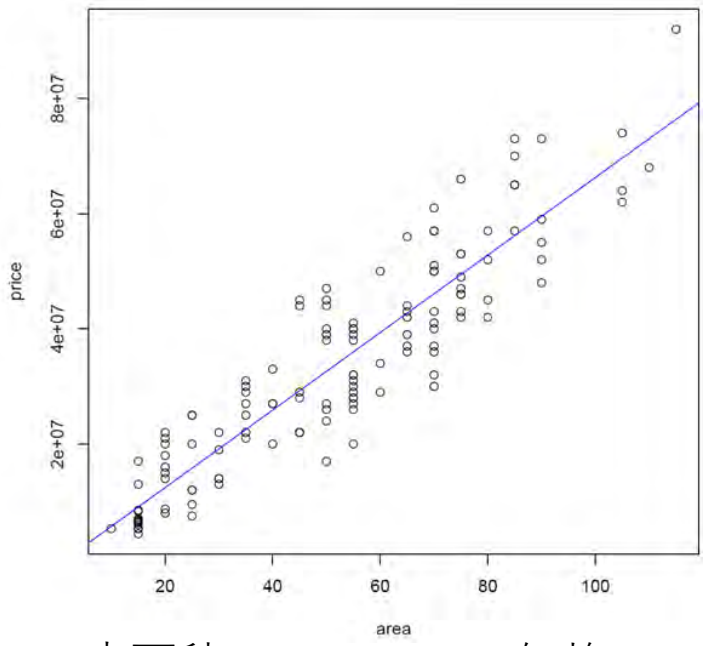
[Silver et al. (Google Deep Mind): Mastering the game of Go with deep neural networks and tree search,
Nature, 529, 484—489, 2016]



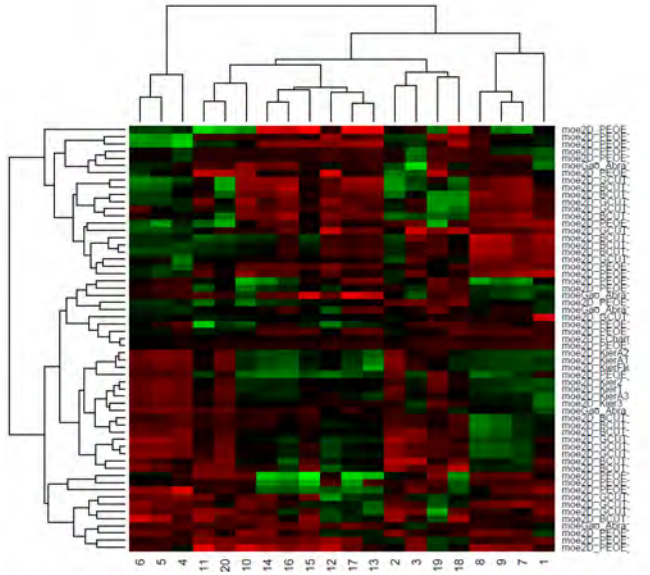
機械学習手法の解説

機械学習の考え方

- データから複数の変数間の関係・傾向を抽出
- まず「データ」を用意
- 「データ」は数値で表現されている必要あり
  - “画像”はピクセルで表現
  - “男女”は±1で表現



床面積×マンション価格



ジヒドロ葉酸還元酵素阻害剤データ

機械学習の流れ

問題ごと　　共有化可能

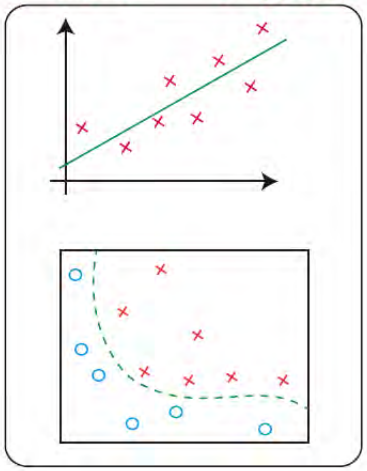
特徴抽出

特徴ベクトル

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{bmatrix}$$

分析

パラメータ推定



判別・回帰など

学習規準：汎化誤差最小化

分野ごとに様々なノウハウ
深層学習（Deep learning）は自動的に特徴量を抽出

予測モデルの構築

$$y = f(x;\theta)$$

モデルの
パラメータ

一度特徴ベクトルに変換してしまえばあとは統計の問題.
→ 汎用的な手法（機械学習）を適用できる.

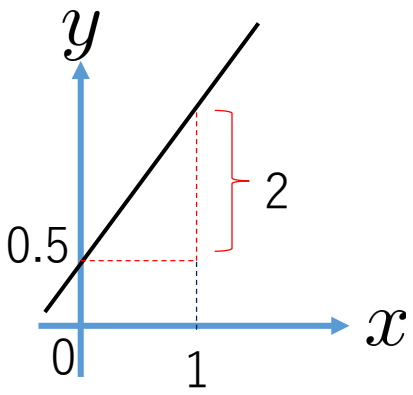
「モデル」とは？

- 「関数」とは？

入力xに対して出力y=f(x)を出す対応関係を記述

$$y = f(x)$$

$$y = 2x + 0.5$$



例：1g単位の材料費xが1円増えれば製造費単価yが2円増える.

係数が不明な場合：　$y = \beta_1 x + \beta_0$

- このように変数間の関係を数式で表したものを**「モデル」**と言う.
- 未知の係数β_0, β_1を**「パラメータ」**と呼ぶ.

データ解析の方針：このパラメータβ_0, β_1をデータから推定（”学習”）

その他のモデル

$$y = \beta_0 + \exp(\beta_1 x)$$

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2$$

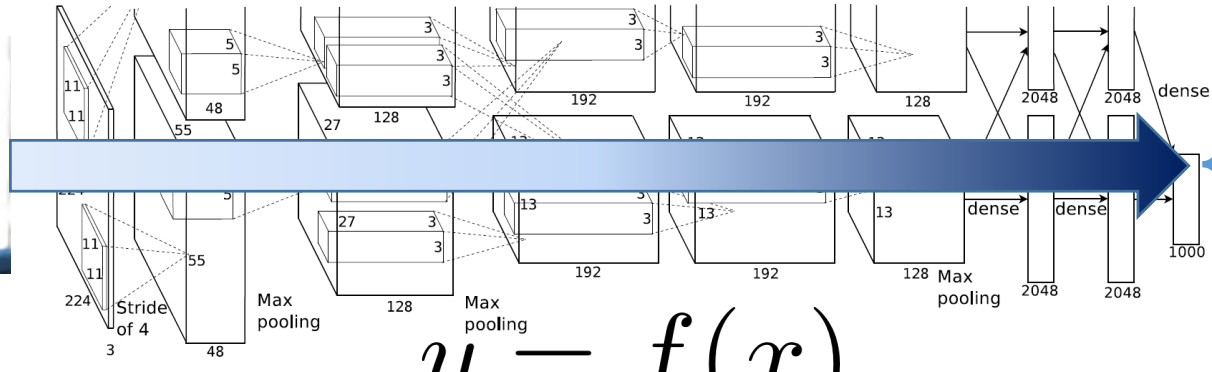
$$y = \beta_0 + \beta_1 \cos(2\pi x) + \beta_2 \cos(4\pi x) + \beta_3 \cos(8\pi x)$$

$$y = \beta_0 + \sum_{i=1}^{M} \beta_i \exp(-(x - \mu_i)^2)$$

教師あり学習

教師あり学習

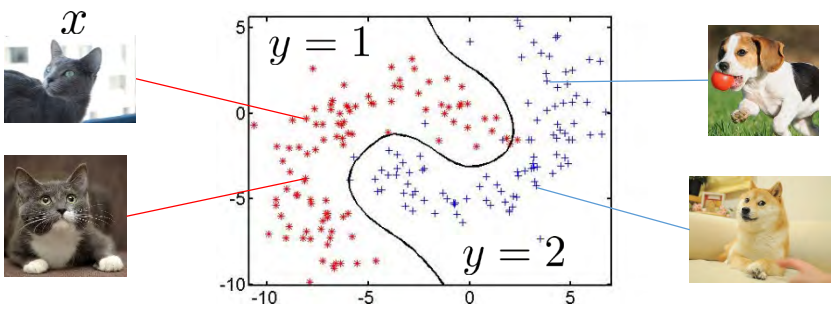
画像



x

$y = f(x)$

y



学習：「関数」をデータに当てはめる

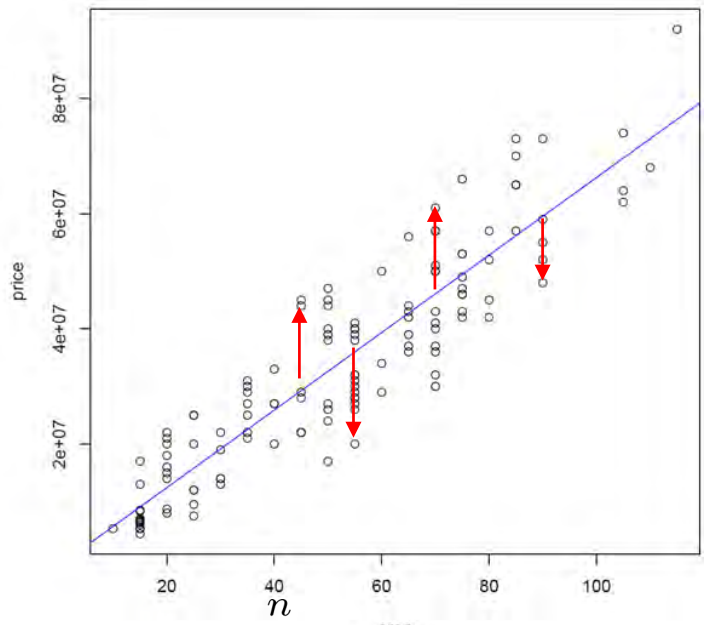
モデル：関数の集合（例：深層NNの表せる関数の集合）

線形モデル

$$y = \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p + \beta_0 + \epsilon$$

y:従属変数, x:特徴ベクトル

マンション価格=$\beta_1 \times$ 床面積 + $\beta_2 \times$ 築年数 + β_3+(揺らぎ)



n

最小二乗法 $\min_{\beta_0,\beta_1,\beta_2,\beta_3} \sum_{i=1} (y_i - \beta_1 x_{i,1} - \beta_2 x_{i,2} - \beta_3 x_{i,3} - \beta_0)^2$

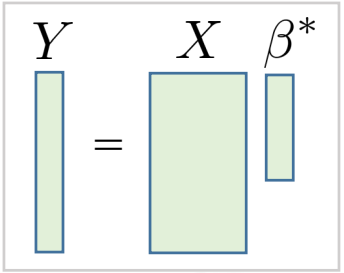
最小二乗法

n個の観測値（サンプル）：　$(y_i, \boldsymbol{x}_i) \in \mathbb{R} \times \mathbb{R}^d \ (i = 1, \ldots, n)$

$$Y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \in \mathbb{R}^n, \ X = \begin{bmatrix} \boldsymbol{x}_1^\top & 1 \\ \vdots & \vdots \\ \boldsymbol{x}_n^\top & 1 \end{bmatrix} \in \mathbb{R}^{n \times (d+1)}, \ \boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{bmatrix} \in \mathbb{R}^n$$

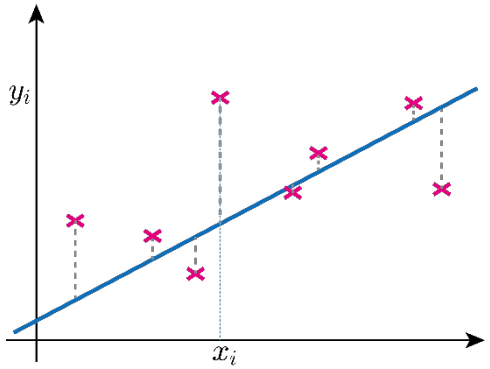
β^*を真の回帰係数（これを推定したい）とすると，

$$Y = X\beta^* + \boldsymbol{\epsilon}$$



最小二乗推定量 (最尤推定量)：

$$\begin{aligned} \hat{\boldsymbol{\beta}} &= \arg\min_{\boldsymbol{\beta} \in \mathbb{R}^{d+1}} \sum_{i=1}^{n} (y_i - [\boldsymbol{x}_i^\top 1]\boldsymbol{\beta})^2 \\ &= \arg\min_{\boldsymbol{\beta} \in \mathbb{R}^{d+1}} \|Y - X\boldsymbol{\beta}\|^2 \\ &= (X^\top X)^{-1} X^\top Y \end{aligned}$$



中古マンション価格データの分析

国土交通省が公開している不動産取引価格情報から世田谷区の中古マンション取引価格データ (平成２５年度第３四半期分) を取得．ここから一部を抜粋したデータで回帰分析をやってみる．
http://www.land.mlit.go.jp/webland/download.html

従属変数(y)：価格
説明変数(x)：
1. 最寄駅からの距離（徒歩）
2. 延床面積
3. 建物の構造
4. 建ぺい率
5. 容積率
6. 建築年
7. 最寄り駅に急行が止まるか (0-1変数で表現)

Rの関数 lm を使って分析.

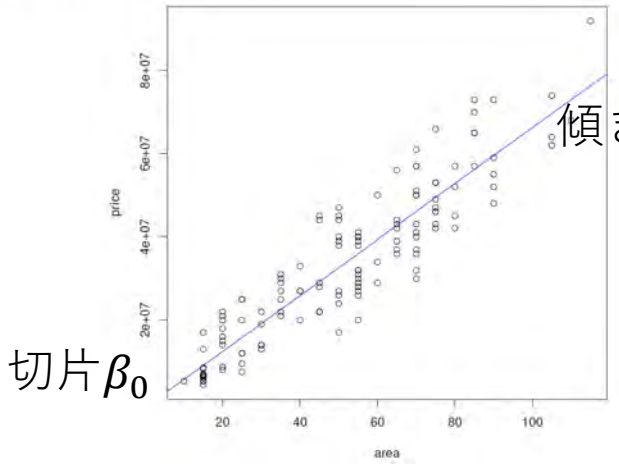
回帰分析関数（lm）

最小二乗法の計算：床面積を説明変数として価格を予測

```
sman.lm <- lm(price ~ area,data=sman)  #回帰分析はこの一行でOK
plot(sman$area,sman$price, xlab="area",ylab="price")  #結果をプロット
abline(sman.lm , lwd=1 , col="blue")
```

$$y = \beta_1 x + \beta_0 + \epsilon$$

価格　　床面積



分析結果

β_0　β_1

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -1029167    1519818  -0.677      0.5
area          673025      26408  25.485   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

切片項
床面積
推定された係数　標準偏差　t-統計量　t検定のp-値

P-値：その変数が予測に寄与している度合い.
「統計的仮説検定」：通例0.05以下なら寄与していると判定

変数選択

- 価格を予測するのに意味のある変数を選択

AICによる特徴選択

```
sman.lmall <- lm(price ~., data=sman)
sman.lmAIC <- step(sman.lmall)
summary(sman.lmAIC)
```

最寄駅からの距離 + 床面積 + 築年数

の三変数モデルが採用された.

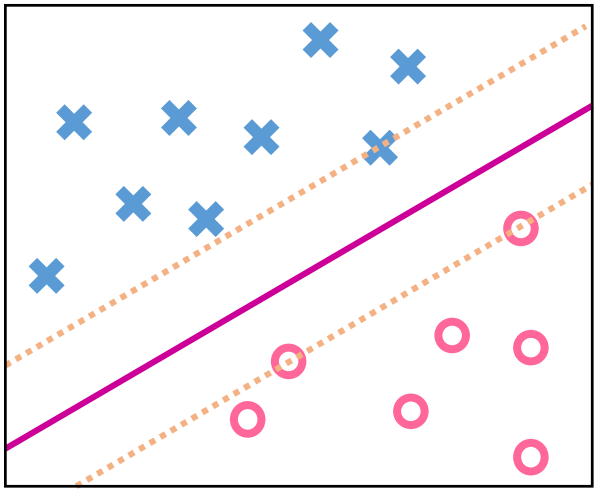
※ AIC：予測精度の良さを測る指標

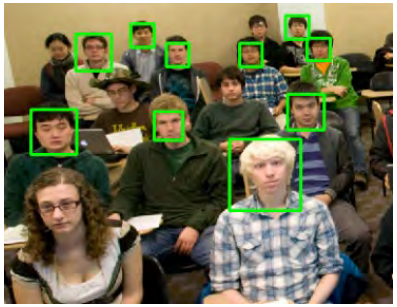
1. 最寄駅からの距離（徒歩）
2. 延床面積
3. 建物の構造
4. 建ぺい率
5. 容積率
6. 築年数
7. 最寄り駅に急行が止まるか

解釈： それ以外の変数を入れるとむしろノイズになって予測精度が落ちる.

教訓： なんでもかんでも説明変数として使うべきではない
→「**過学習**」

判別





メール：スパムか？スパムでないか？

顔認識：顔か？顔でないか？

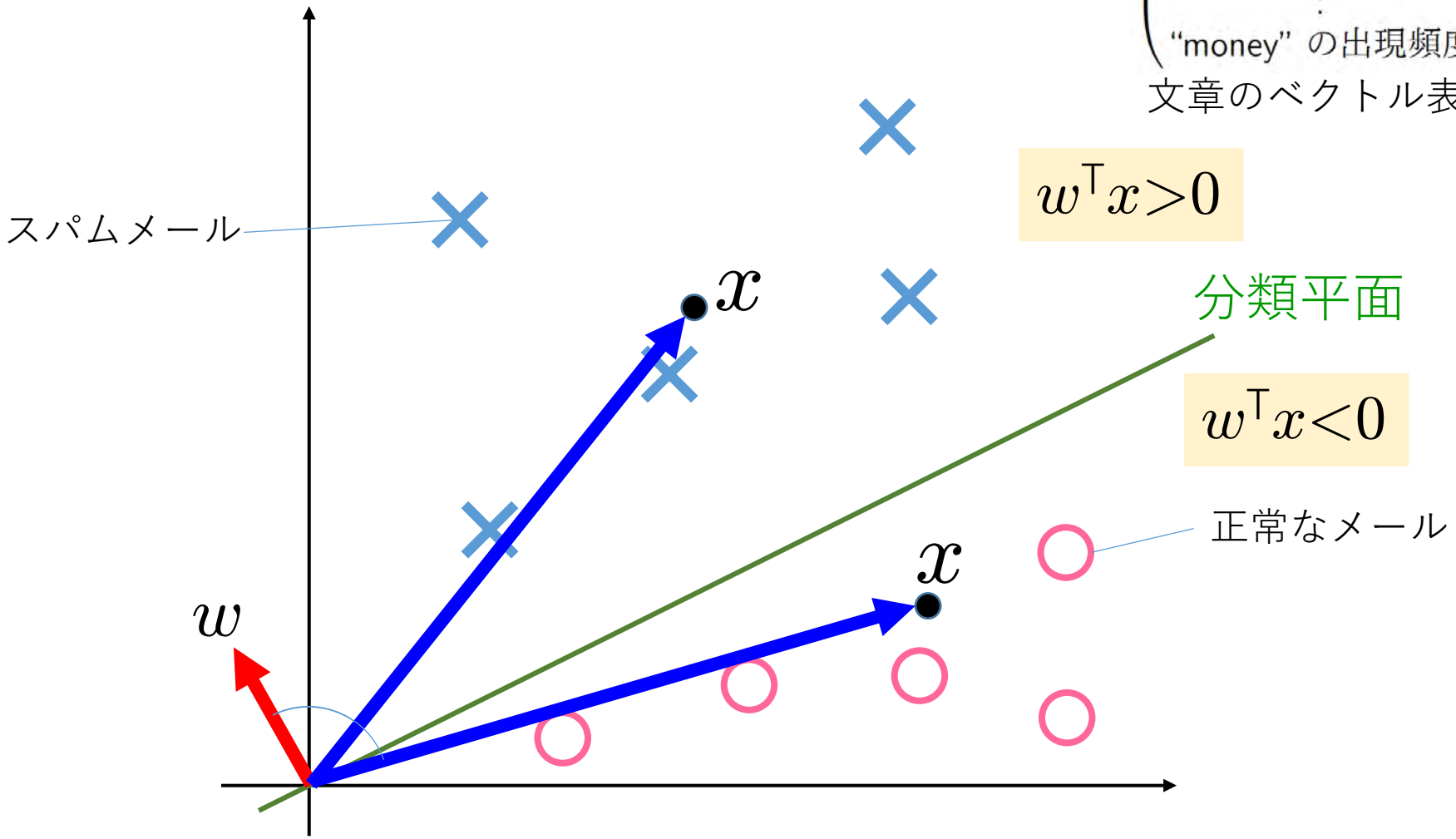
線形分類機

$$w^\top x = w_1 x_1 + w_2 x_2 + \cdots + w_d x_d$$

Bag-of-words

$$x = \begin{pmatrix} \text{“please” の出現頻度} \\ \text{“credit” の出現頻度} \\ \vdots \\ \text{“money” の出現頻度} \end{pmatrix}$$

文章のベクトル表現例



サポートベクトルマシン（SVM）

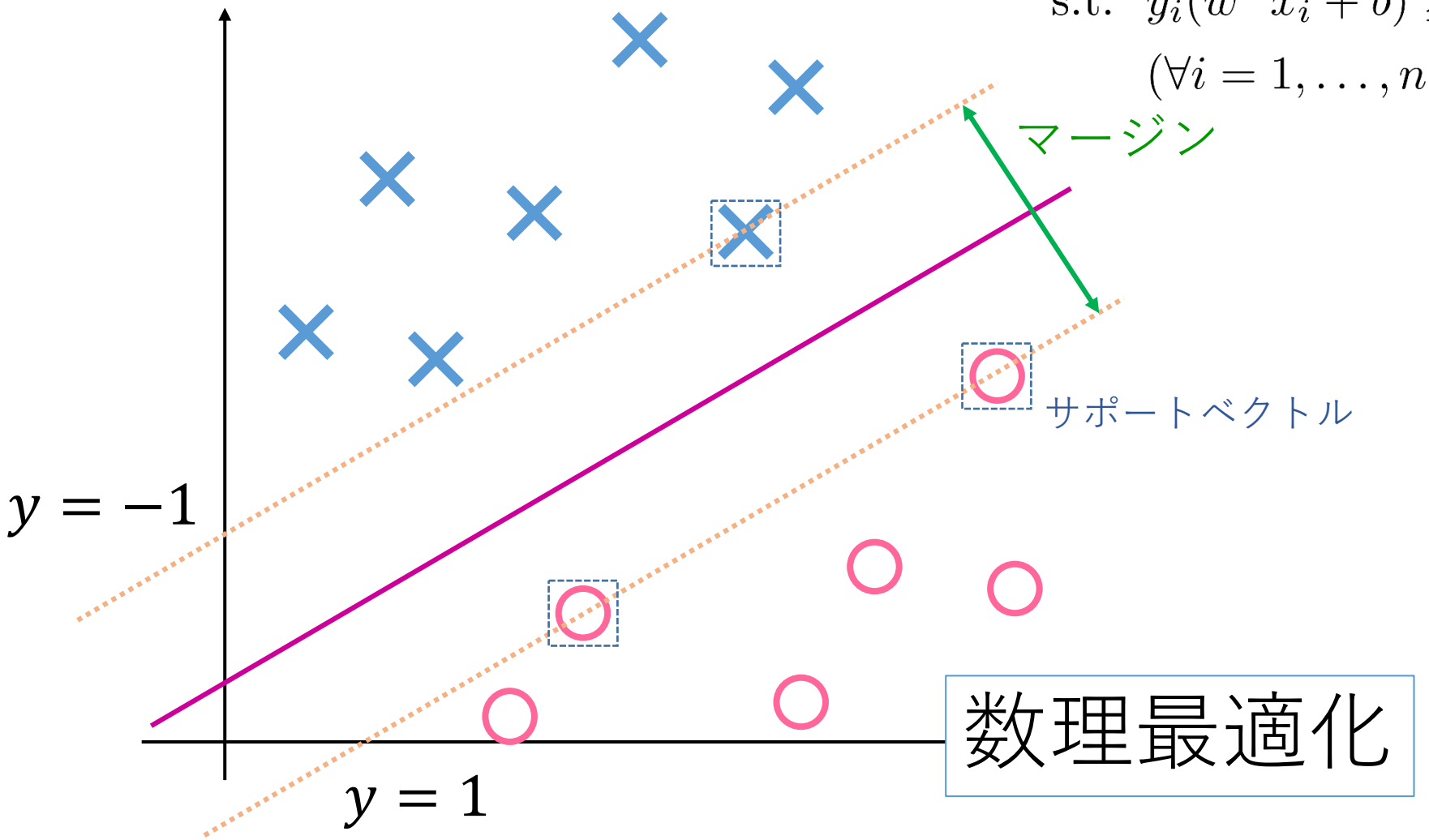
[Vapnik,63]

マージンを最大化

VC (Vapnik-Chervonenkis) 理論による正当化

$$\min_{w,b} \quad \frac{\|w\|^2}{2}$$

$$\text{s.t.} \quad y_i(w^\top x_i + b) \geq 1 \quad (\forall i = 1, \ldots, n)$$



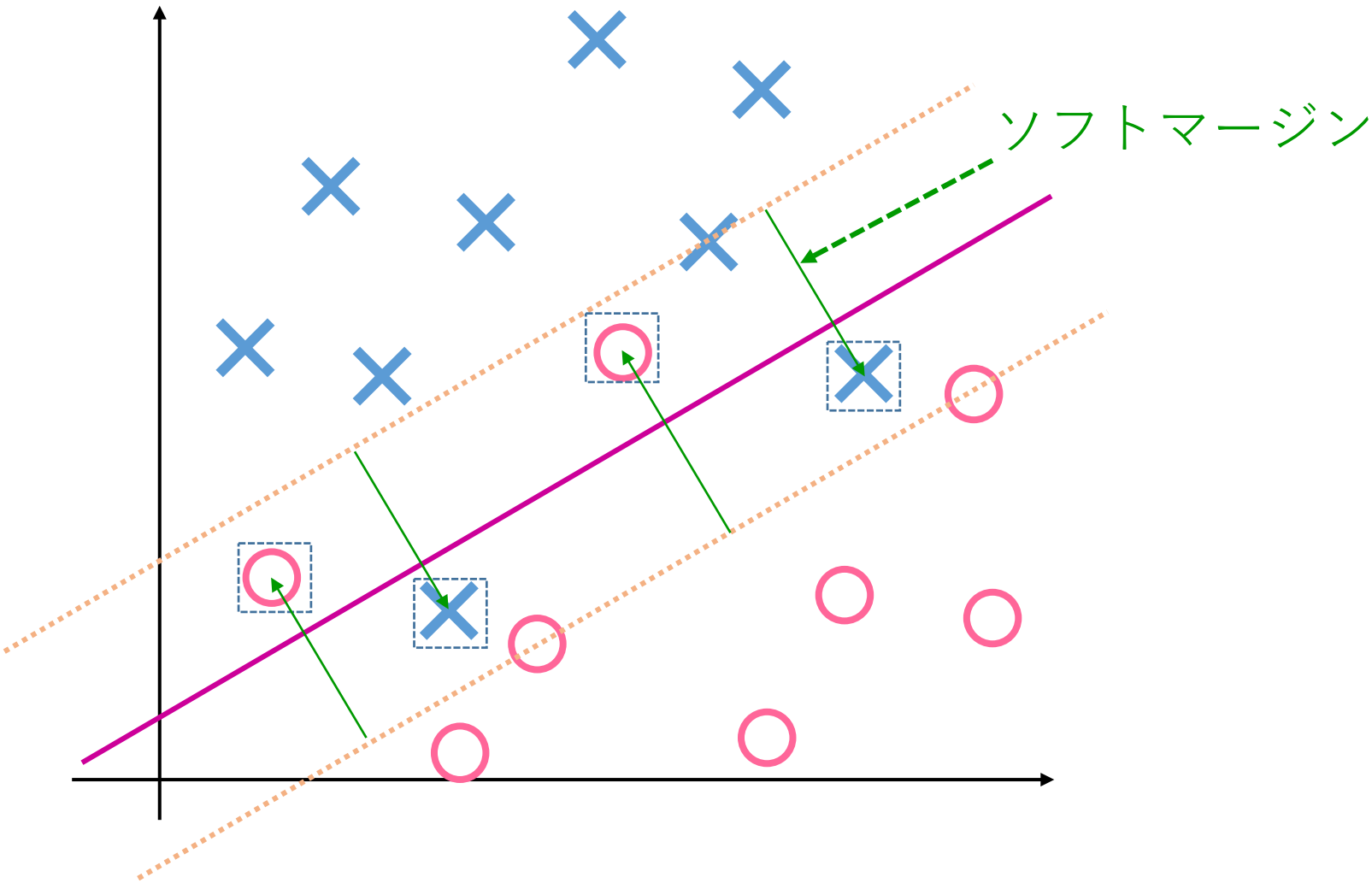
数理最適化

ソフトマージンSVM

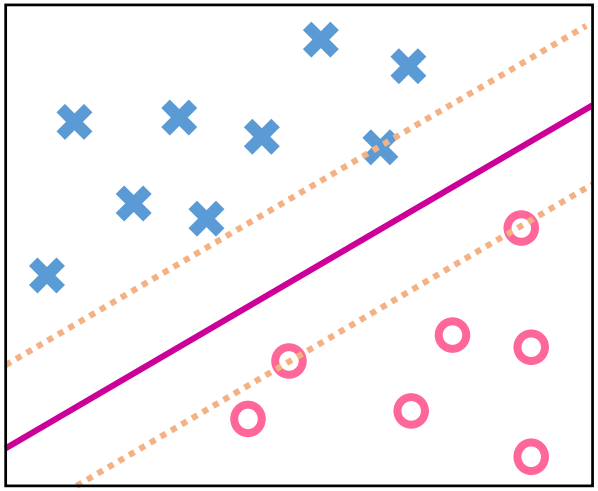
[Cortes+Vapnik,95]

マージンを最大化
誤分類も許す

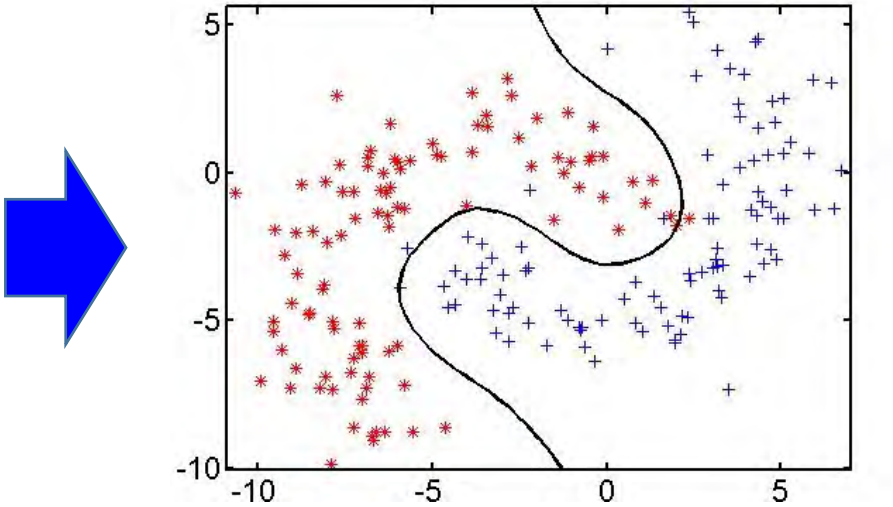
$$\min_{w,b} \sum_{i=1}^{n} \max\{1 - y_i(w^\top x_i + b), 0\} + C\frac{\|w\|^2}{2}$$



線形から非線形へ



線形

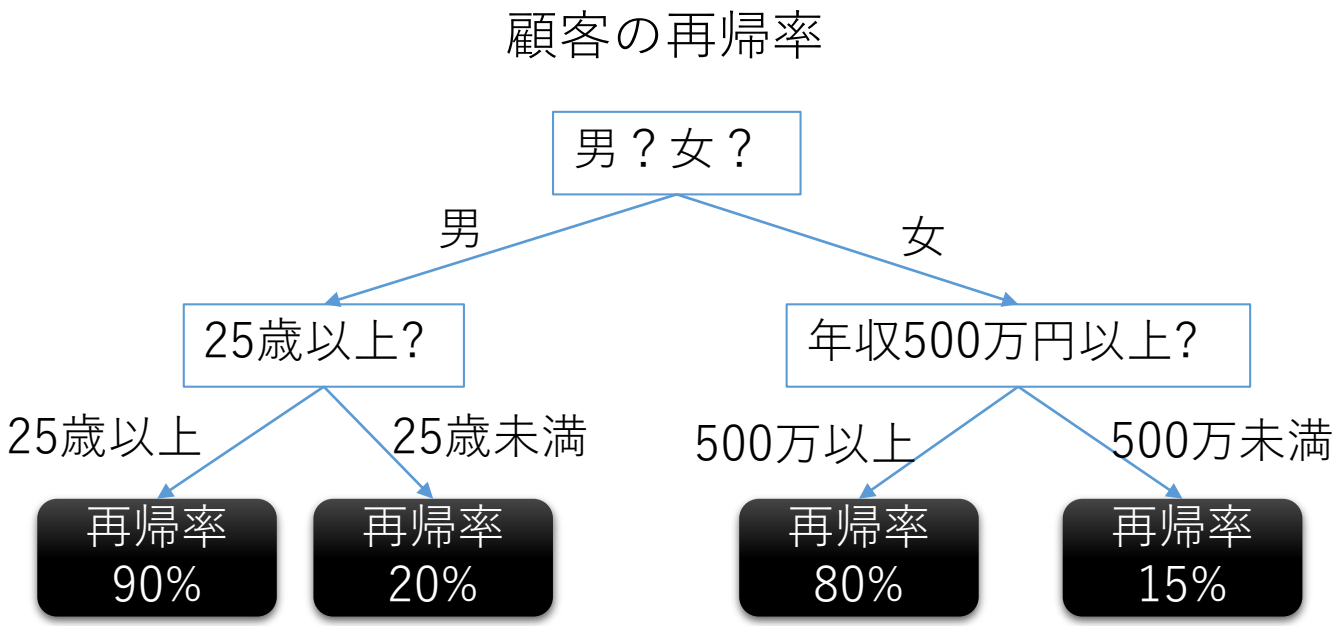


非線形

もっと複雑なルールで判別をしたい

決定木

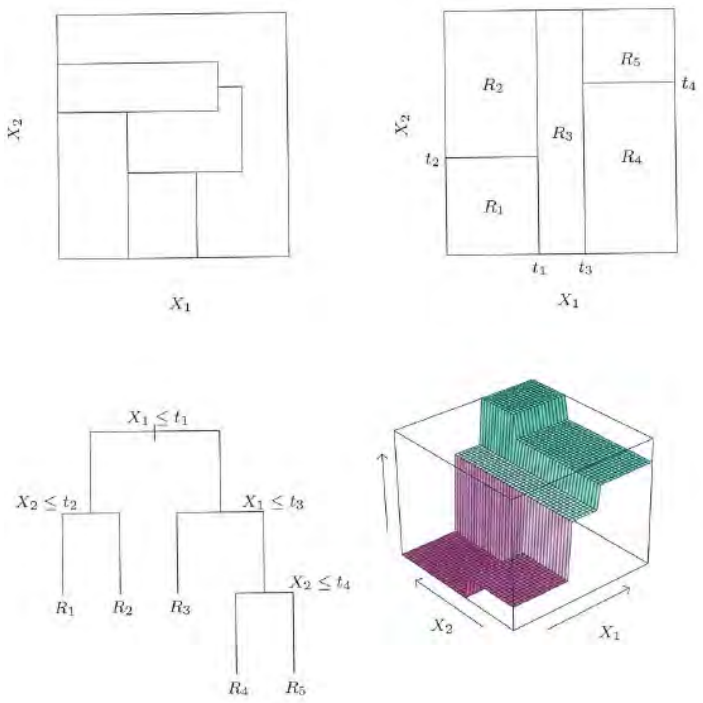
- 解釈可能性が高い
- 決定木を組み合わせた勾配ブースティングはデータマイニング系コンペティションで常連
  (判別だけでなく回帰にも利用可能)



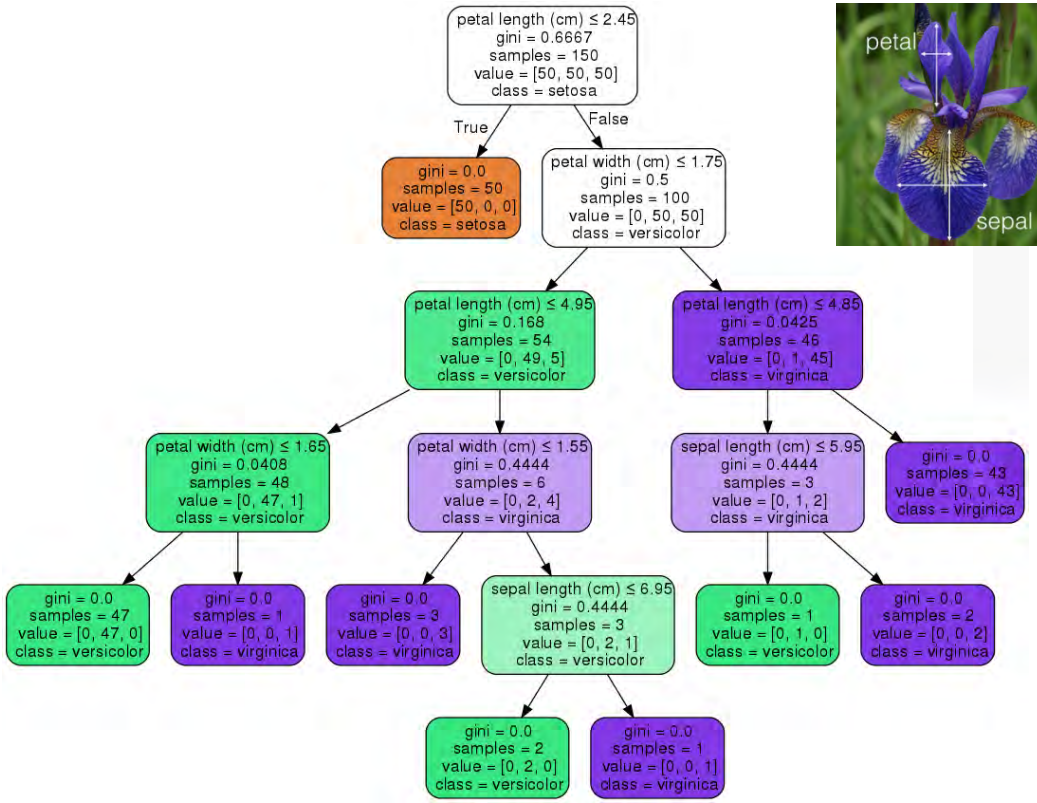
- 学習された決定木から要因を把握しやすい.
- 分析結果から対策を立てるのに有用.

決定木

決定木の様子
2次元の説明変数



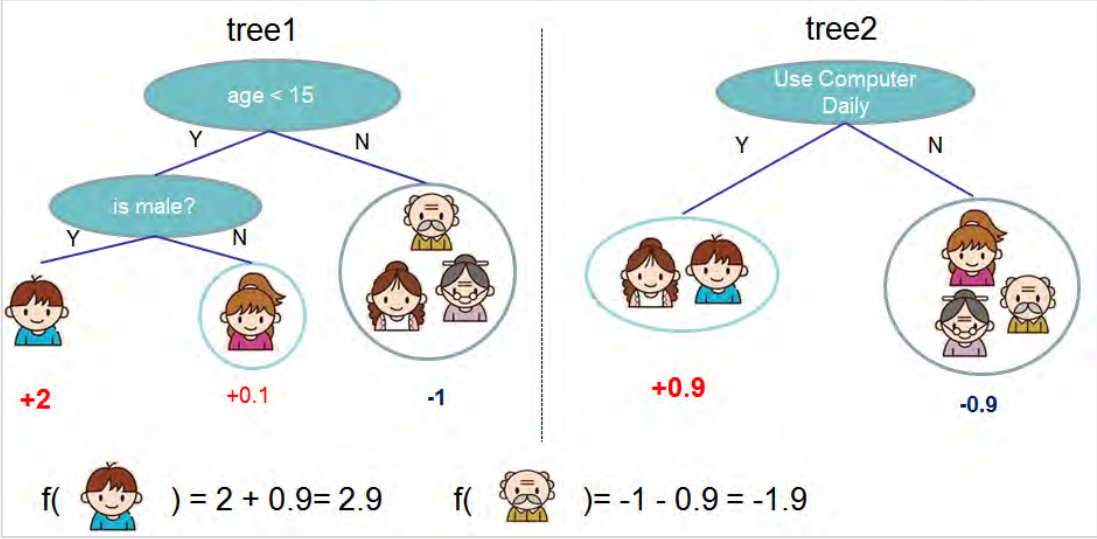
図はHastie, Tibshirani, Friedman: The Elements of Statistical Learning, Springer, 2001.より



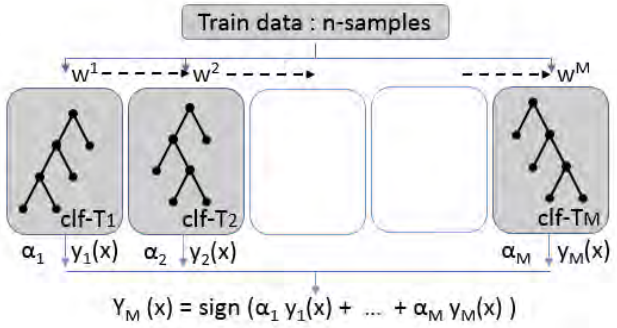
Python scikit-learnによるiris(アヤメ)データ分類

勾配ブースティング

- XGBoostやLightGBMが有名
- 「決定木の和」で強力な判別を実現
  - 決定木一つでは複雑な判別が難しい→ 複数用意してその和(多数決)を取る
  - 和の取り方に勾配ブースティングと呼ばれる技法を使用



「コンピュータゲームが好きか？」を判別



沢山の決定木の**多数決**を出力

[Chen, Guestrin: XGBoost: A Scalable Tree Boosting System. KDD2016.]

[Ke, Meng, Finley, Wang, Chen, Ma, Ye, Liu: LightGBM: A Highly Efficient Gradient Boosting Decision Tree. NIPS2017.]

勾配ブースティング

XGBoostは各種データ解析コンペティションで好成績

Machine Learning Challenge Winning Solutions

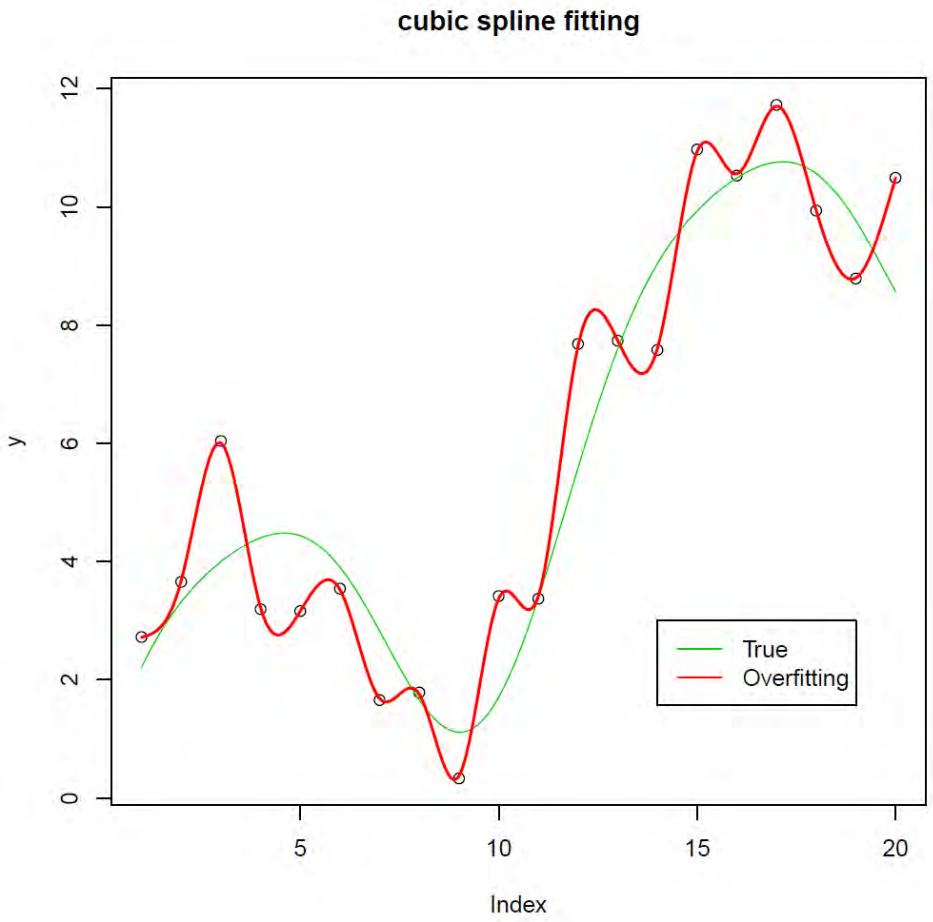
XGBoost is extensively used by machine learning practitioners to create state of art data science solutions, this is a list of machine learning winning solutions with XGBoost. Please send pull requests if you find ones that are missing here.

- Maksims Volkovs, Guangwei Yu and Tomi Poutanen, 1st place of the 2017 ACM RecSys challenge. Link to paper.
- Vlad Sandulescu, Mihai Chiru, 1st place of the KDD Cup 2016 competition. Link to the arxiv paper.
- Marios Michailidis, Mathias Müller and HJ van Veen, 1st place of the Dato Truely Native? competition. Link to the Kaggle interview.
- Vlad Mironov, Alexander Guschin, 1st place of the CERN LHCb experiment Flavour of Physics competition. Link to the Kaggle interview.
- Josef Slavicek, 3rd place of the CERN LHCb experiment Flavour of Physics competition. Link to the Kaggle interview.
- Mario Filho, Josef Feigl, Lucas, Gilberto, 1st place of the Caterpillar Tube Pricing competition. Link to the Kaggle interview.
- Qingchen Wang, 1st place of the Liberty Mutual Property Inspection. Link to the Kaggle interview.
- Chenglong Chen, 1st place of the Crowdflower Search Results Relevance. Link to the winning solution.
- Alexandre Barachant (“Cat”) and Rafał Cycoń (“Dog”), 1st place of the Grasp-and-Lift EEG Detection. Link to the Kaggle interview.
- Halla Yang, 2nd place of the Recruit Coupon Purchase Prediction Challenge. Link to the Kaggle interview.
- Owen Zhang, 1st place of the Avito Context Ad Clicks competition. Link to the Kaggle interview.
- Keiichi Kuroyanagi, 2nd place of the Airbnb New User Bookings. Link to the Kaggle interview.
- Marios Michailidis, Mathias Müller and Ning Situ, 1st place Homesite Quote Conversion. Link to the Kaggle interview.

過学習

複雑なモデル（例えば深層ニューラルネット）を用いるのが常に良い選択か？

→ そうとは限らない．「**過学習**」に注意する必要あり．



評価用データの準備

手元にあるデータ：10,000データ点

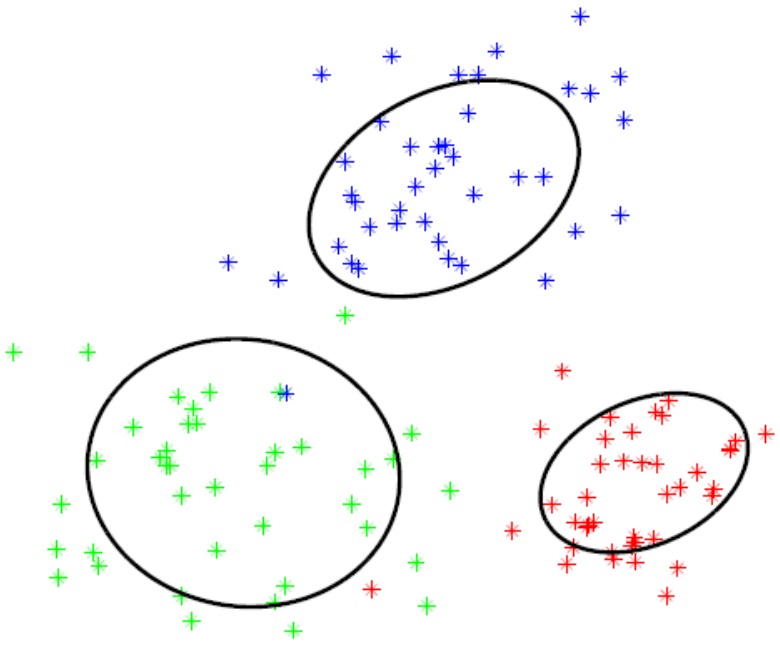
- トレーニング：　　6,000データ点
- バリデーション：　2,000データ点
- テスト：　　2,000データ点

トレーニング：モデルを学習
バリデーション：学習のパラメータおよび閾値を決定
　　Detection rate, Recall, F値
テスト：最終的な評価．よりメタな特徴量の選択などに用いる．

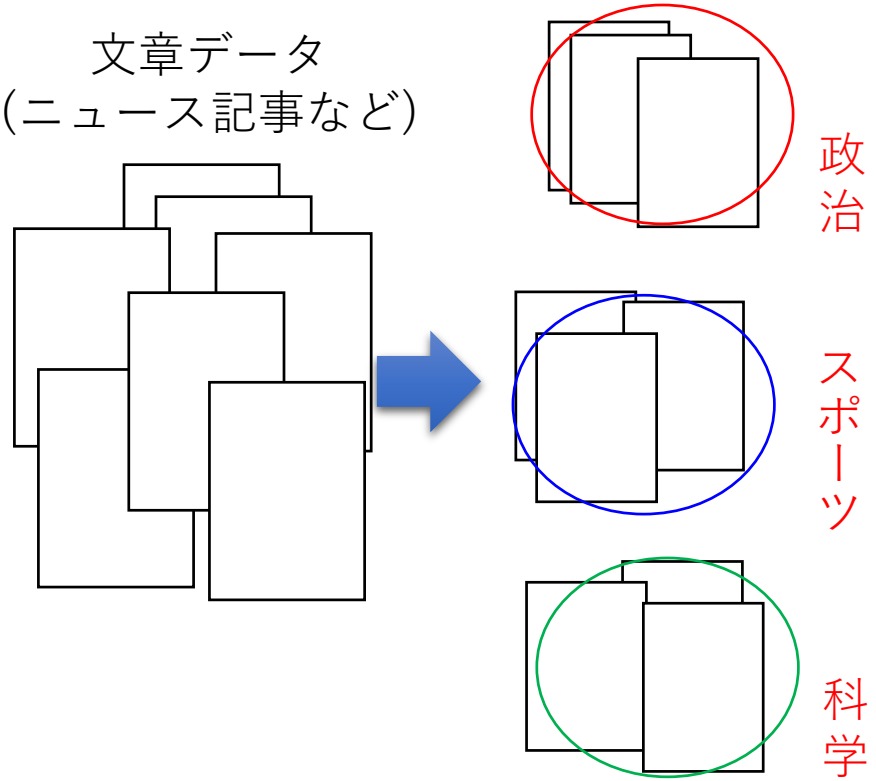
参考： Machine learning lecture by Andrew Ng in Coursera.

教師なし学習

クラスタリング



混合ガウス分布によるクラスタリング



トピックモデル

- 文章分類

トピックモデリング

やりたいこと：「文章分類」

各文章に「トピック」を自動的に割り当てたい
（例：ブログ記事の分類）

- ゲッツェの１点でドイツが世界制覇
- フィオレンティーナはDF ゴンサロ・ロドリゲスとの契約を延長

どちらもサッカーにまつわる話だとわかる.
しかし，「サッカー」という単語は出ていない.
→ 文章に表れる単語がサッカーに関係する記事で目にするものばかり.
→ トピック：単語頻度の傾向．同じ記事に現れやすい単語は同じトピックに属するだろう.

単語の共起関係が単語の意味を定める.

Bag of Words

	単語１	単語２	単語３	…	単語M
文章１	4	8	0	…	2
文章２	2	0	1	…	6
文章３	7	0	8	…	4
文章４	3	4	3	…	1
⋮	⋮	⋮	⋮	…	⋮
文章N	0	2	5	…	6

Bag of words

Syria	13
people	5
bomb	7
economy	1
immigrants	2
soccer	0
walk	1

文章数×単語数の単語頻度行列.
基本的に単語頻度行列は超スパース (ほとんどの要素が0).

この表だけからトピックを抽出し文章をトピックに分類(クラスタリング)

ニュース記事の分類などに応用可能

Latend Dirichlet Allocation（LDA）

LDAでは各文章に出現する単語の頻度を確率モデルでモデル化
→ 単語の出現傾向からどのトピックに分類されるか推論可能

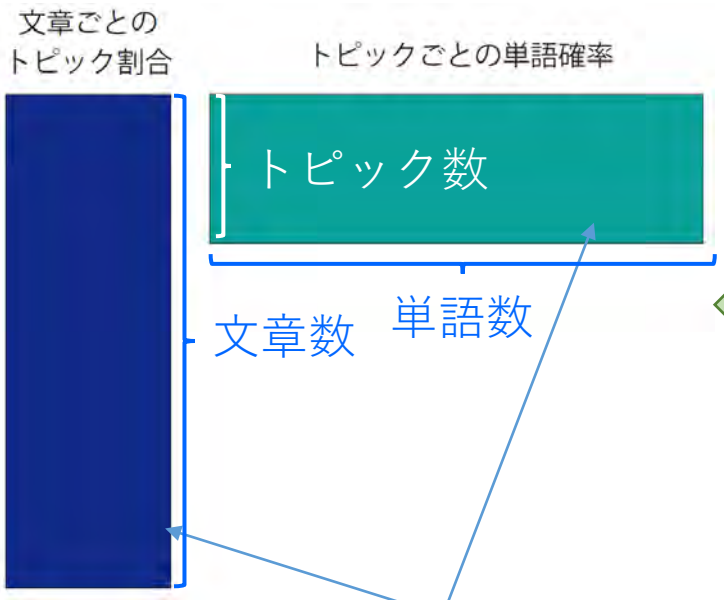
$$= \sum_{k=1}^{K} \underline{\text{文章 } d \text{ 内のトピック } k \text{ の割合 } (P(k|d))} \times \text{トピック } k \text{ で単語 } w \text{ が出現する確率 } (P(w|k))$$

文章 d 内で単語 w が出現する確率 $(P(w|d))$

各文章の各トピックの割合を推定
→文章分類

$$(P(w|d))_{w,d} =$$

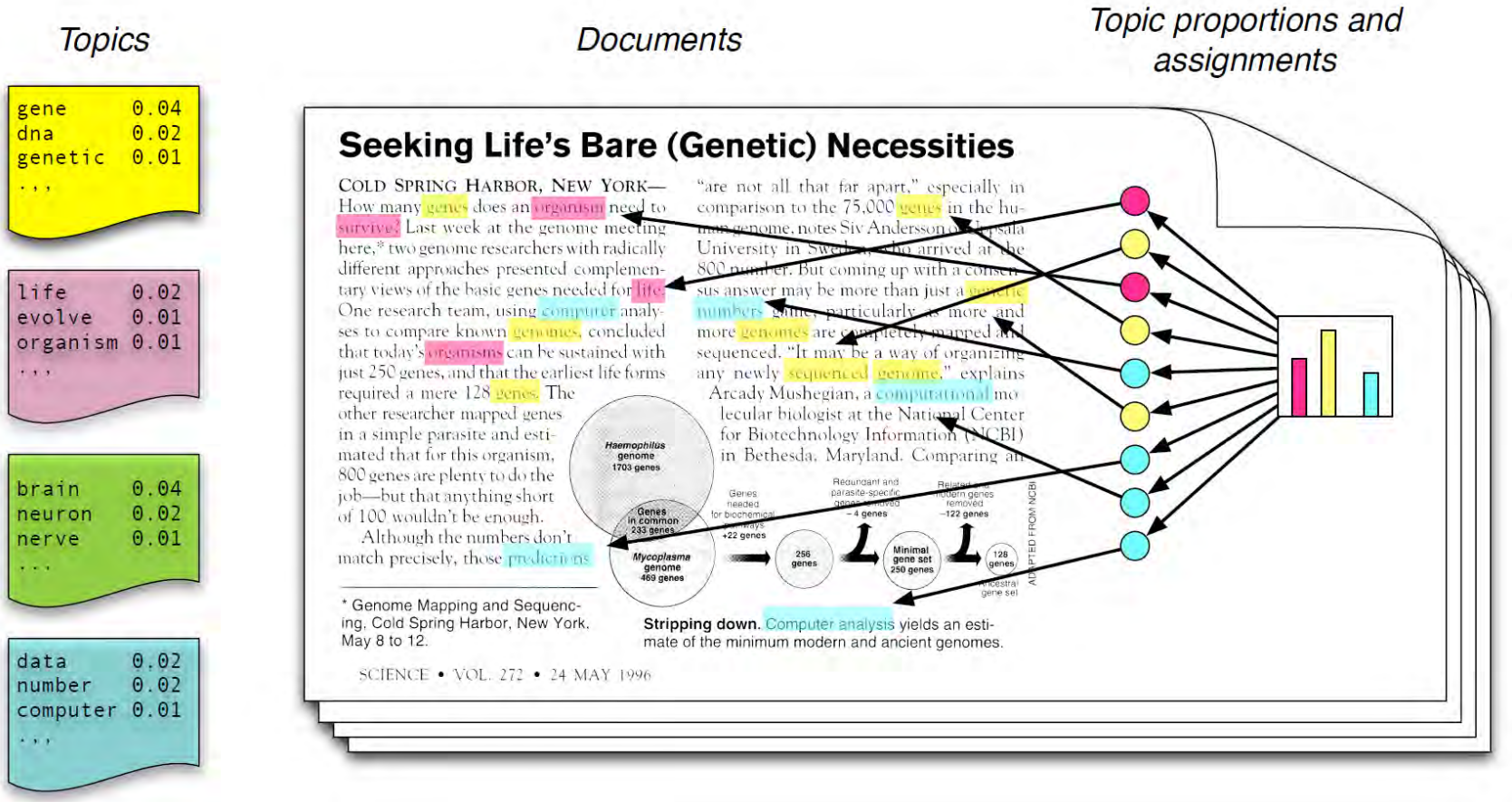
各文章d内で単語wが出現する確率



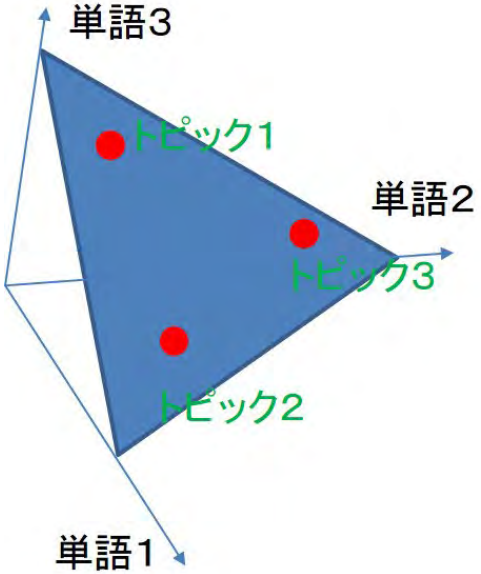
これらをデータに合うよう学習
（ベイズ推定）

データ

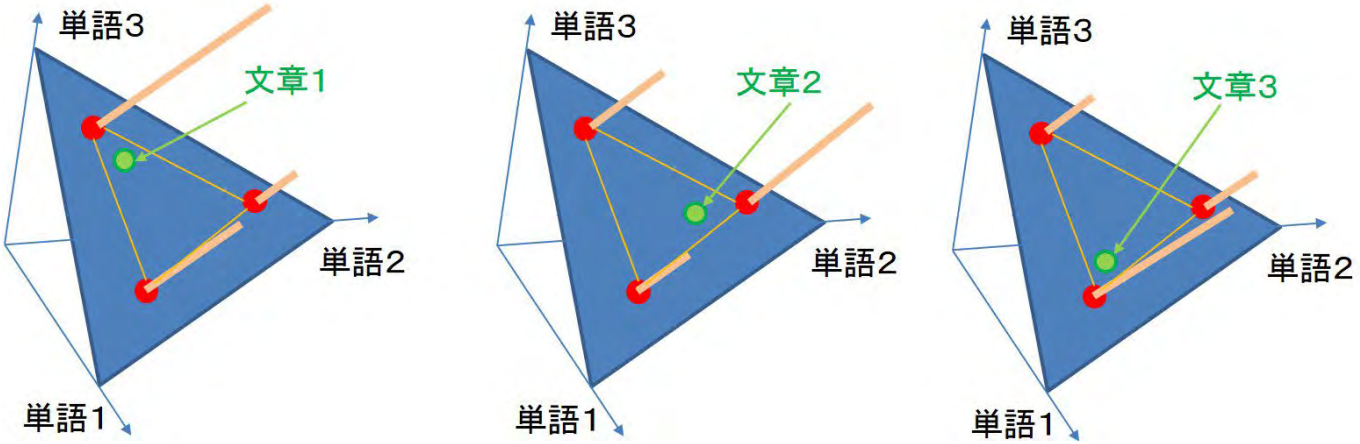
	単語1	単語2	単語3
文章1	4	8	0
文章2	2	0	1
文章3	7	0	8
文章4	3	4	3



David Blei: KDD 2011 tutorial.



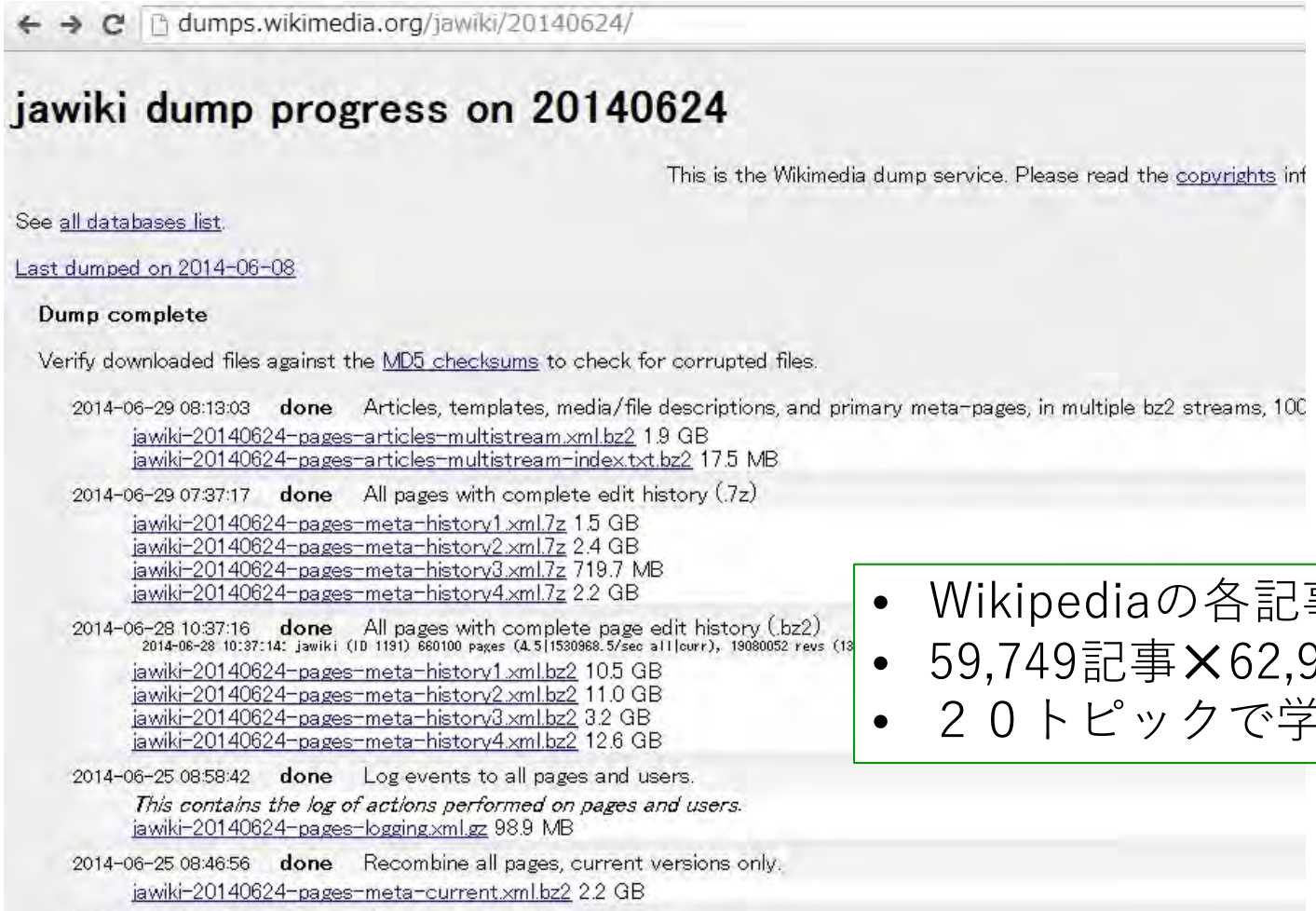
各トピックは単語の出現頻度で特徴付けられる
（サッカーに関するトピックならサッカー関係の単語が出やすい）



各文章における単語の出現頻度はトピックの混合で決まる

日本語WikipediaでLDA

2014 年 6 月の日本語Wikipedia の記事データ.
http://dumps.wikimedia.org/jawiki/20140624/ からjawiki-20140624-pages-articles1.xml.bz2 をダウンロード.
pythonライブラリのgensimでLDAを実行.



- Wikipediaの各記事が各文章
- 59,749記事×62,999単語
- ２０トピックで学習（低ランク

トピックごとの主要単語

Topic-1に関する話題で出てきやすい単語

	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
[1,]	"丁目"	"de"	"オブ"	"windows"	"モハ"
[2,]	"人口"	"la"	"シリーズ"	"gt"	"mm"
[3,]	"交通"	"サン"	"ゲーム"	"pc"	"クハ"
[4,]	"教育"	"cc"	"ドラマ cd"	"lt"	"km"
[5,]	"地理"	"file"	"vol"	"例えば"	"番台"
[6,]	"道路"	"フォン"	"アニメ"	"os"	"キハ"
[7,]	"中学校"	"年頃"	"名探偵コナン"	"ms"	"両編成"
[8,]	"北海道道"	"ルイ"	"機動戦士ガンダム"	"ii"	"編成"
[9,]	"高等学校"	"フランス"	"one"	"mhz"	"系電車"
[10,]	"小学校"	"ドイツ"	"ナレーション"	"mb"	"サハ"
[11,]	"年生"	"le"	"劇場版"	"vs"	"国鉄"
[12,]	"鉄道"	"マリア"	"テレビ朝日版"	"minus"	"cm"
[13,]	"行政"	"ジャン"	"ドラえもん"	"for"	"クモハ"
[14,]	"自由民主党"	"image"	"テレビアニメ"	"mac"	"形電車"
[15,]	"番地"	"パリ"	"それいけ"	"system"	"dd"

トピックごとの主要単語

	Topic 15	Topic 16	Topic 17	Topic 18
[1,]	"and"	"km"	"ch"	"紀元前"
[2,]	"in"	"text"	"土曜"	"在位"
[3,]	"file"	"style"	"金曜"	"年頃"
[4,]	"to"	"東京都"	"日曜"	"天正"
[5,]	"university"	"億円"	"月曜"	"年代"
[6,]	"new"	"北海道"	"月から"	"には"
[7,]	"on"	"center"	"日から"	"慶長"
[8,]	"by"	"align"	"木曜"	"代藩主"
[9,]	"with"	"bar"	"月まで"	"在位紀元前"
[10,]	"press"	"県道"	"kw"	"万石"
[11,]	"for"	"時間"	"日本テレビ系列"	"ユリウス暦"
[12,]	"at"	"部リーグ"	"出力"	"天文"
[13,]	"en"	"大阪府"	"備考"	"寛永"
[14,]	"white"	"新潟県"	"火曜"	"世紀"
[15,]	"black"	"bull"	"fm"	"任官"

出現しやすい単語からトピックの意味がよくわかる.

各トピックの成分が大きい記事タイトル

"Topic 3 :" アニメ関係
"架空の国一覧 | 岡村明美 | 佐久間レイ | 三木眞一郎 | 石田彰 | うえだゆうじ | 山口勝平 | 根谷美智子 | 広瀬正志 | 小西克幸 | 八奈見乗児 | 山口由里子 | 進藤尚美 | くまいもとこ | 関俊彦 | 千葉一伸 | 草尾毅 | 坂本千夏 | 飛田展男 | 三宅健太"

"Topic 4 :" PC関係
"Xeon | PC-9821シリーズ | 順序数 | ThinkCentre | Safari | Microsoftオン化傾向 | X68000 | Unicode一覧 E0000-E0FFF | MC68000 | .NET Framev

"Topic 18 :" 歴史物関係
"中国帝王一覧 | 元号一覧（日本） | 天文（元号） | 従一位 | 後白河天皇 | 延暦 | 享保 | 紀元前1千年紀 | 文化（元号） | 伺候席 | 征夷大将軍 | 夏商周年表 | 宝暦 | 備前国 | 守護代 | 醍醐天皇 | 伊勢国 | 摂津国 | 相模国 | 紀元前4世紀"

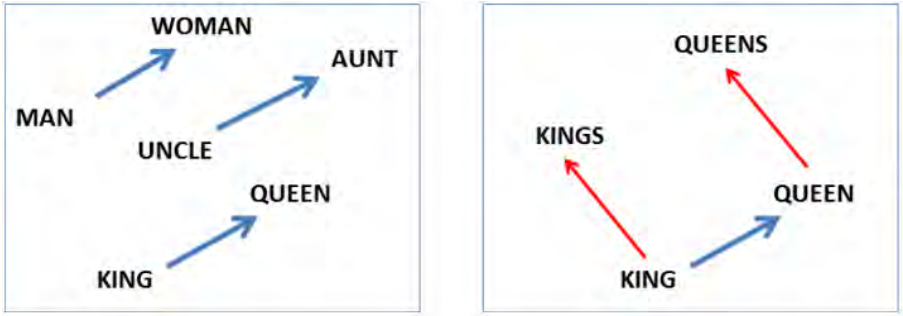
記事タイトルから関連した話題が集まっていることがわかる
※単語の出現頻度のみから学習されていることに注意

Word2vec [Mikolov et al., 2013]

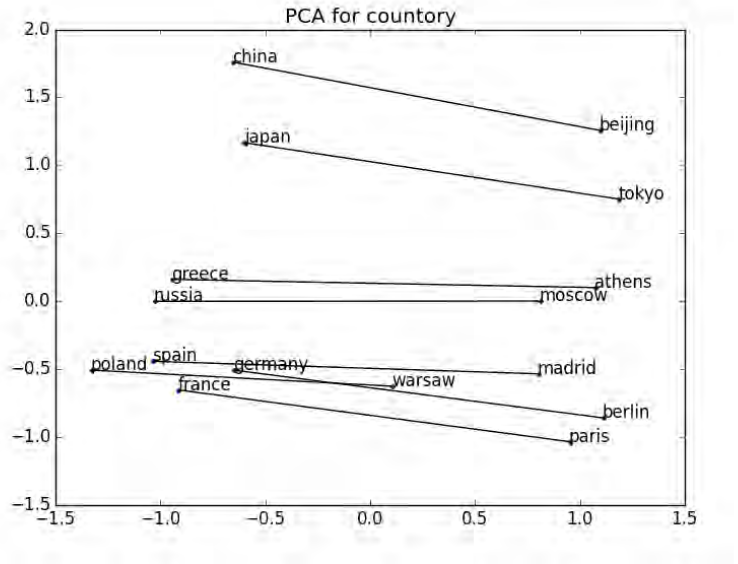
- 単語のベクトル表現を得る方法

“King” – “Man” + “Woman” = “Queen”
“Tokyo” – “Japan” + “China” = “Beijing”

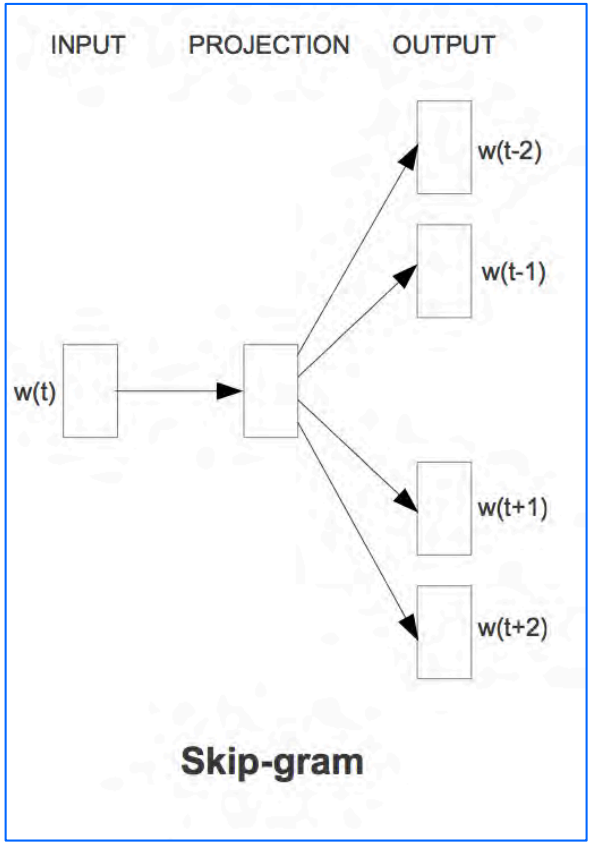
意味を足し引きできるような表現が得られる.



(Mikolov et al., NAACL HLT, 2013)

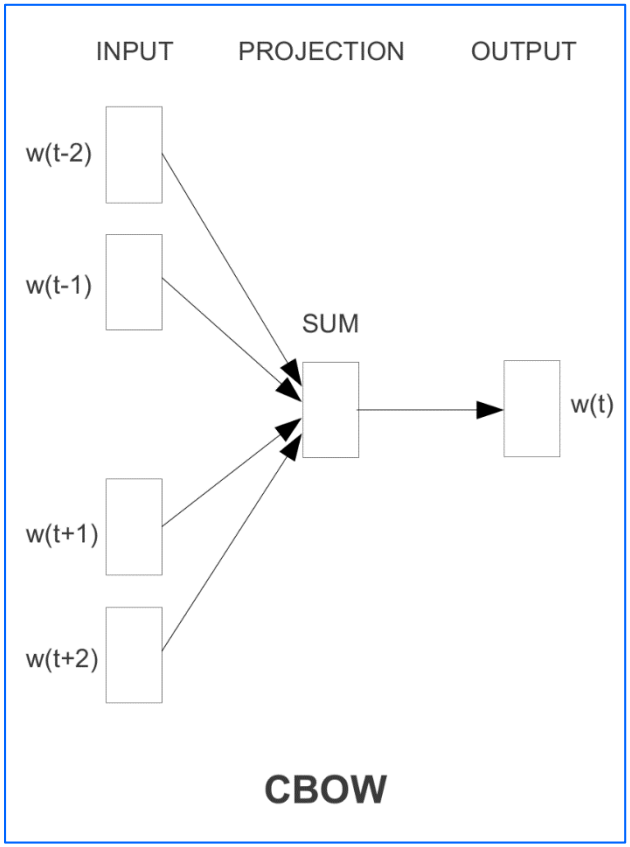


skip-gramとContinuous Bag-of-Words (CBOW)



skip-gramモデル

ある単語のまわりに出現する
単語の確率分布をモデル化



CBOWモデル

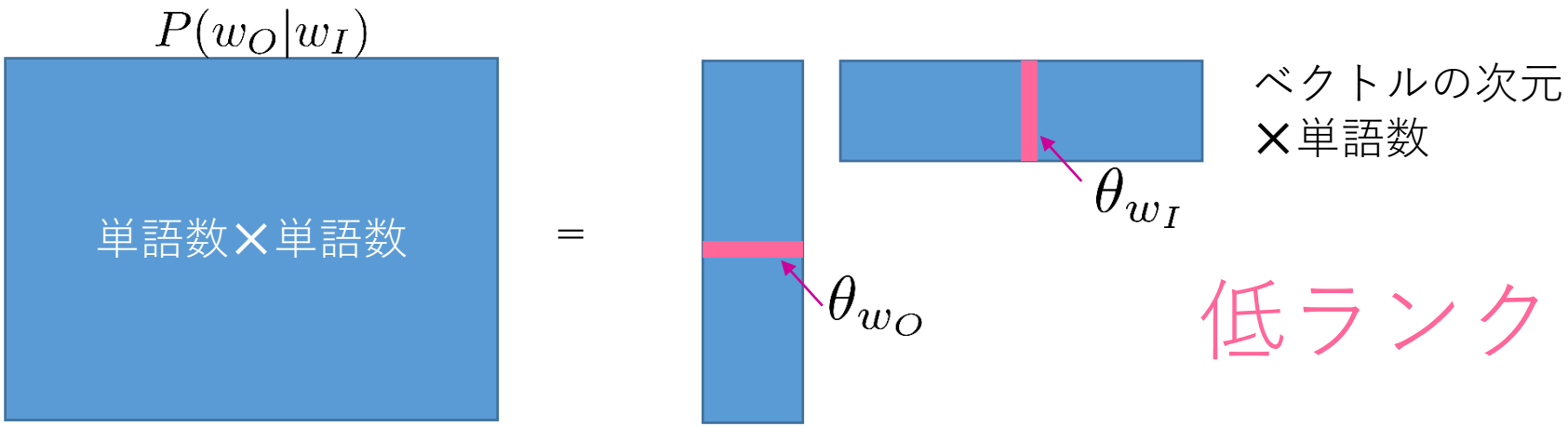
まわりの単語からその場所に
ある単語が出現する確率をモデル化

Skip-gramモデル

単語w_Oがw_Iの周辺（前後１０単語ほど）に現れる確率

$$P(w_O|w_I) = \frac{\exp(\langle \theta_{w_O}, \theta_{w_I} \rangle)}{\sum_{w'} \exp(\langle \theta_{w'}, \theta_{w_I} \rangle)}$$
$$\propto \exp(\langle \theta_{w_O}, \theta_{w_I} \rangle)$$

- 単語のベクトル表現　θ_{w_O} と θ_{w_I} の内積で表現.
- ベクトルの次元はせいぜい500ほど



実際の挙動

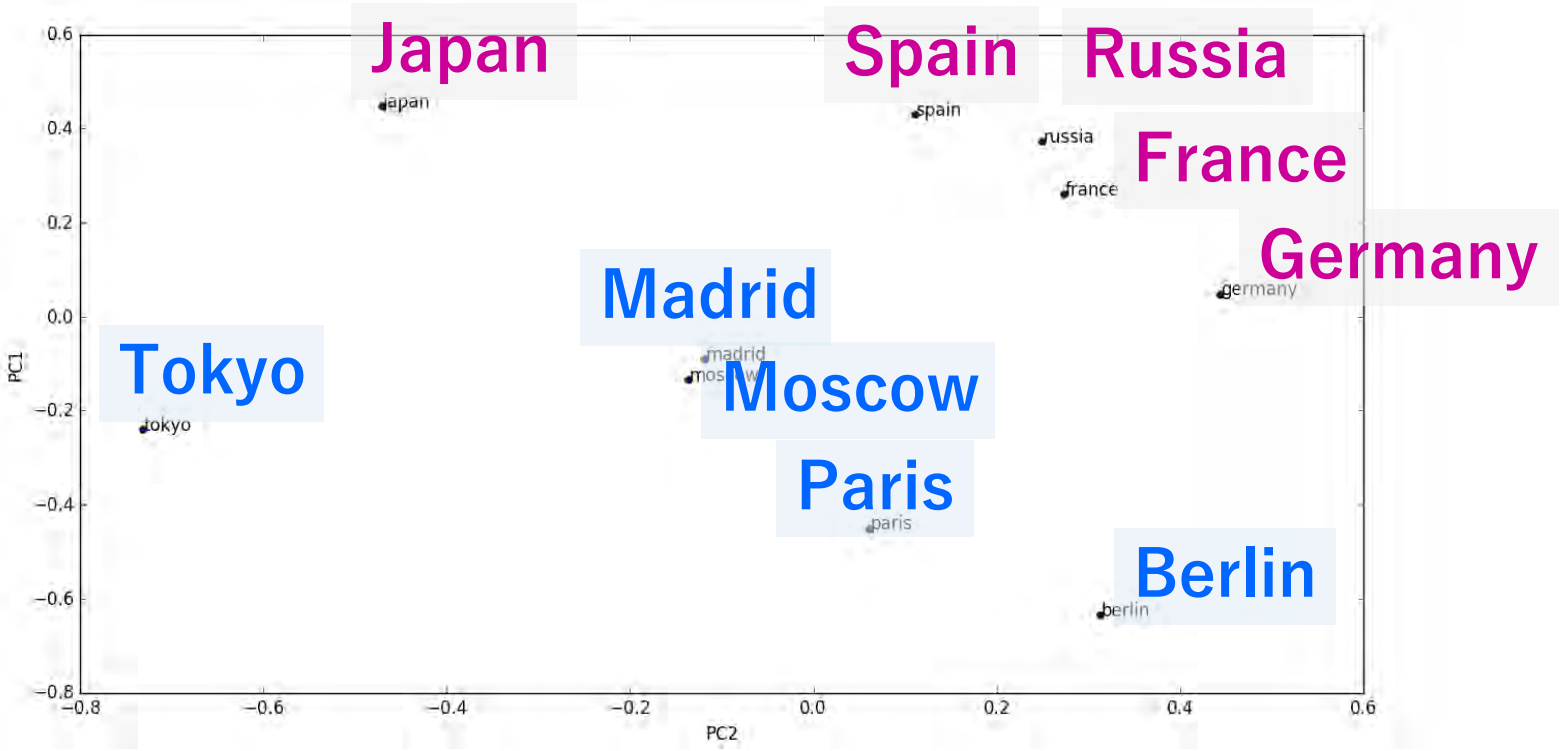

```
from gensim.models import word2vec

train_file = "./mldata/text8"

data = word2vec.Text8Corpus(train_file)
model = word2vec.Word2Vec(size=100, window=5, min_count=5, workers=7)

model.build_vocab(data)
model.train(data)
```

次元１００，前後５単語の出現頻度をモデル化，５回以下の出現単語は無視

“Queen” + “Man” - “Woman” = “King” ?

```
>>> model.most_similar(positive=['queen','man'],negative=['woman'])
[('king', 0.6050819158554077), ('scotland', 0.587989091873169), ('prince', 0.573
6681222915649), ('elizabeth', 0.571208119392395), ('lord', 0.5638244152069092),
('duchess', 0.5520190000534058), ('duke', 0.5498123168945312), ('crown', 0.54618
62087249756), ('sir', 0.544183969497680 7), ('lorraine', 0.5441141128540039)]
```

国と首都



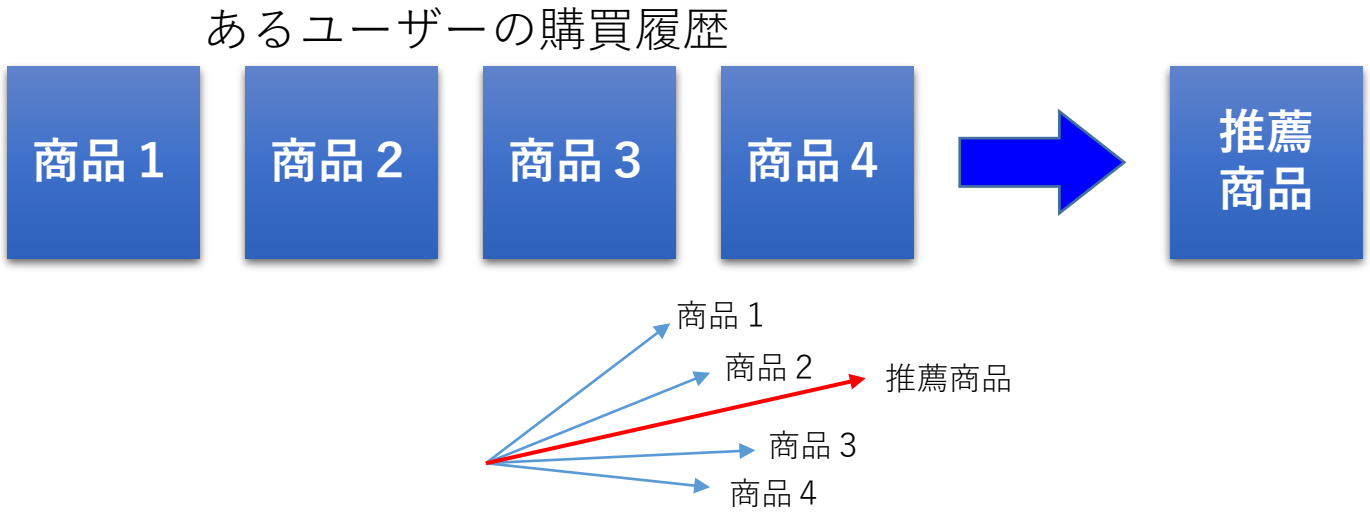
word2vecの貢献：

データの「意味」を低次元ベクトルとして表現できることを実験的に示した.

→ 深層学習にもつながる考え方.

Word2vecの応用

- 商品タグから関連した商品を推薦



- 口コミの要約

<u>文章</u>をベクトル化する手法：skip-thought, text2vec

- 記事から関連広告を推薦
- 感情分析

[Kiros, Zhu, Salakhutdinov, Zemel, Torralba, Urtasun, and Fidler: Skip-Thought Vectors. *arXiv preprint arXiv:1506.06726 (2015).*]

関係データ解析

- 推薦システム

	映画 A	映画 B	映画 C	・・・	映画 X
ユーザ 1	4	8	4	・・・	2
ユーザ 2	2	4	2	・・・	1
ユーザ 3	2	4	2	・・・	1
⋮					

ランク 1 と仮定

各ユーザーが各映画をどれだけ好むかという部分的情報がある.
→　残りの部分（*の部分）を埋めれば推薦できる.

c.f., Netflix prize (100万ドルの賞金,　48万ユーザ✕ 1 万8千映画)

特徴ベクトルを用いた推薦

監督：アンディ・ウォシャウスキー
ラリー・ウォシャウスキー.
粗筋：ニューヨークの会社でしがないコンピュータプログラマーとして働くトマス・アンダーソンには、裏世界の凄腕ハッカー“ネオ”…

性別：女性
年齢：２０代
職業：会社員
住所：都内

	映画 A	映画 B	映画 C	…	映画 X
ユーザ 1	4	8	*	…	2
ユーザ 2	2	*	2	…	*
ユーザ 3	2	4	*	…	*

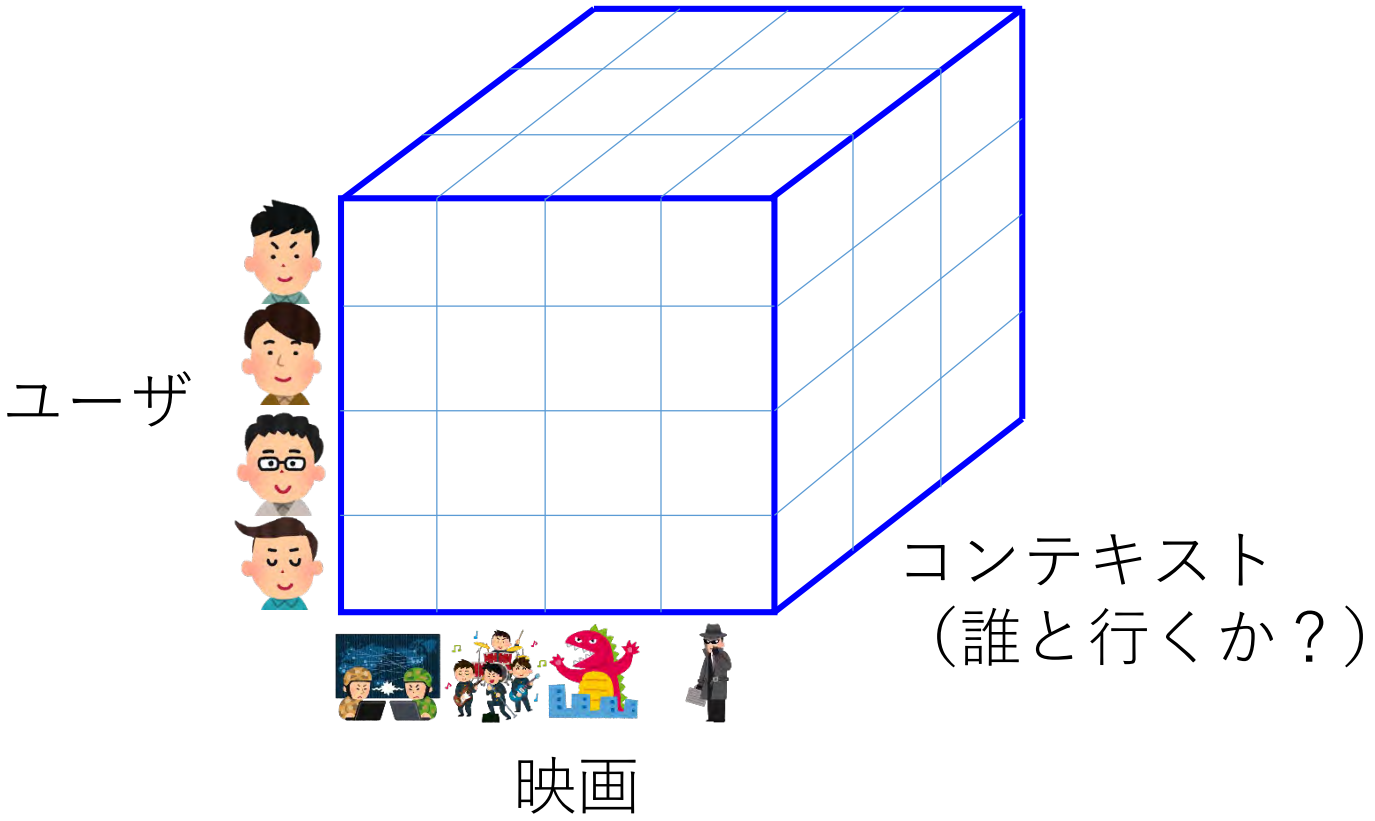
単なる購買履歴だけでなく，各ユーザ・映画の特徴ベクトルも用いたい.

$$f(x^{(1)}, x^{(2)}) = \langle u_1^{(1)}, x^{(1)} \rangle \langle u_1^{(2)}, x^{(2)} \rangle + \cdots + \langle u_d^{(1)}, x^{(1)} \rangle \langle u_d^{(2)}, x^{(2)} \rangle$$

$$= \sum_{r=1}^{d} \langle u_r^{(1)}, x^{(1)} \rangle \langle u_r^{(2)}, x^{(2)} \rangle$$

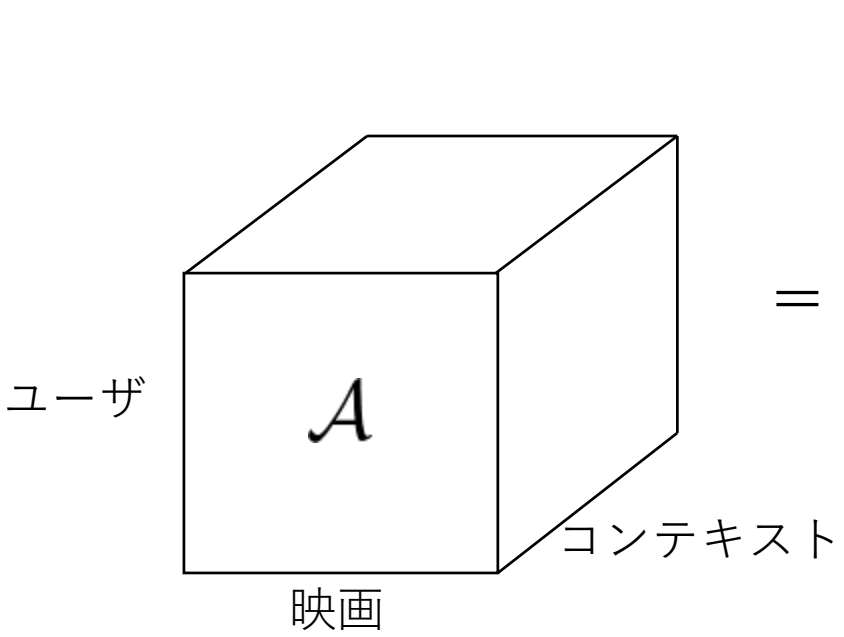
ユーザ 2 の特徴　映画 B の特徴　：word2vecなどを援用して特徴量を作成

補助情報も入れた推薦

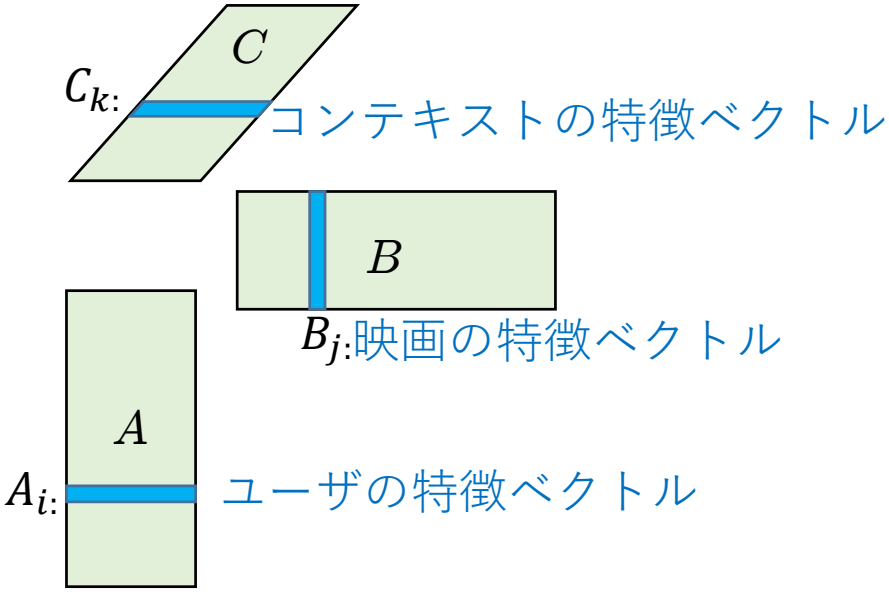


推薦システム：ユーザー×商品×季節
広告モデル：ユーザー×サイト×広告
バイクシェアリング：会員種類×時間×位置

低ランクテンソルモデル



$=$



$$\mathcal{A}_{ijk} = A_{i1}B_{j1}C_{k1} + A_{i2}B_{j2}C_{k2} + \cdots + A_{id}B_{jd}C_{kd}$$

ユーザiが持つ因子1の重み

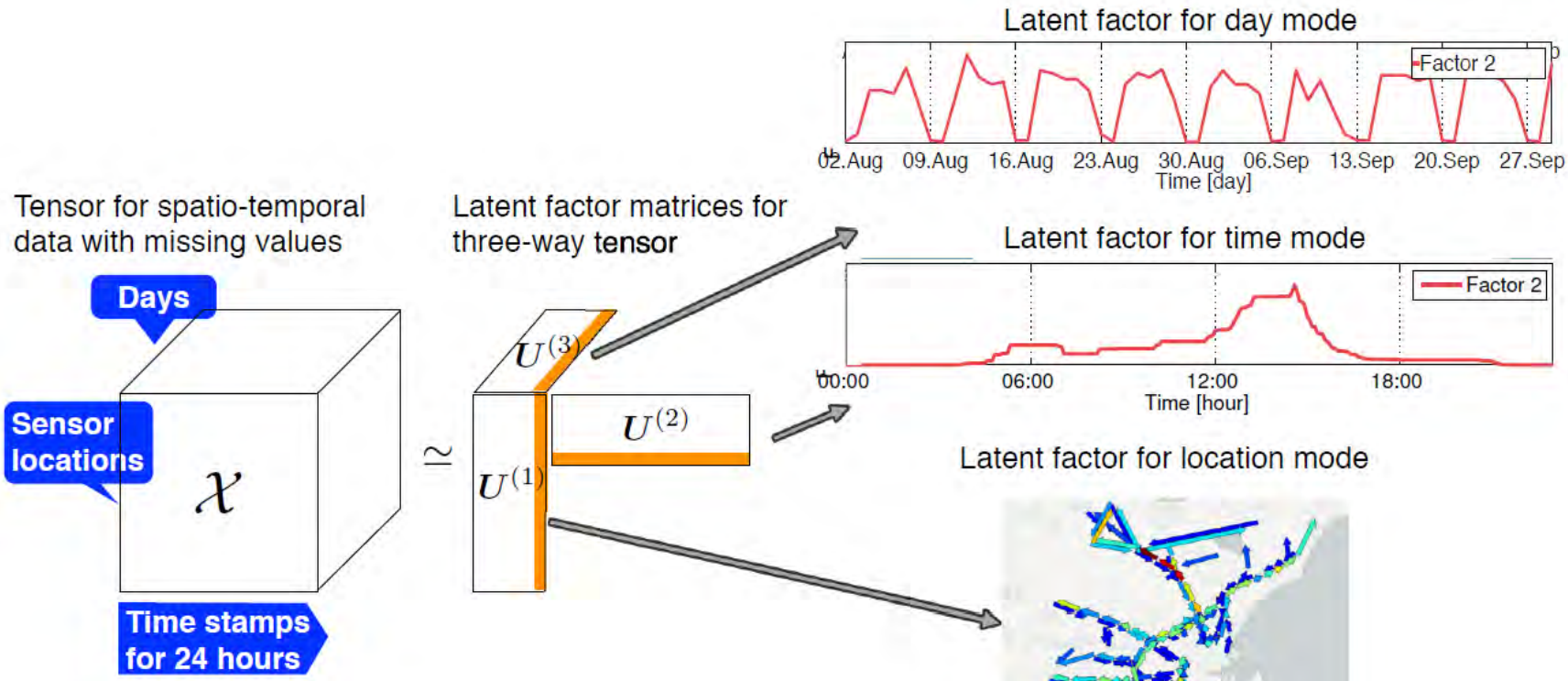
映画jが持つ因子1の重み

コンテキストkが持つ因子1の重み

A,B,Cを観察することで学習結果の解釈も可能

時空間解析への応用

- 交通量の解析



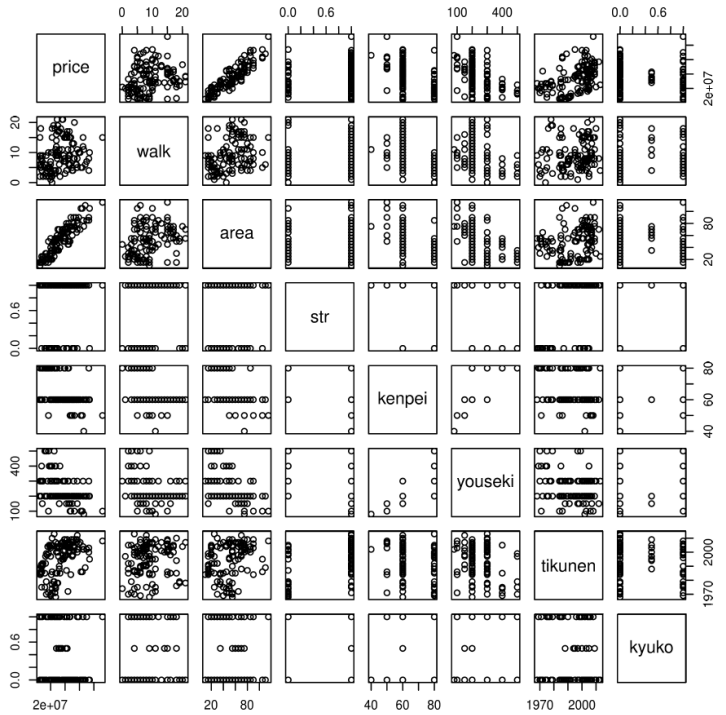
交通量データ

[Takeuchi, Kawahara, Iwata: Structurally Regularized Non-negative Tensor Factorization for Spatio-temporal Pattern Discoveries. ECML-PKDD2017.]

可視化

可能ならまずはデータを可視化してみる.

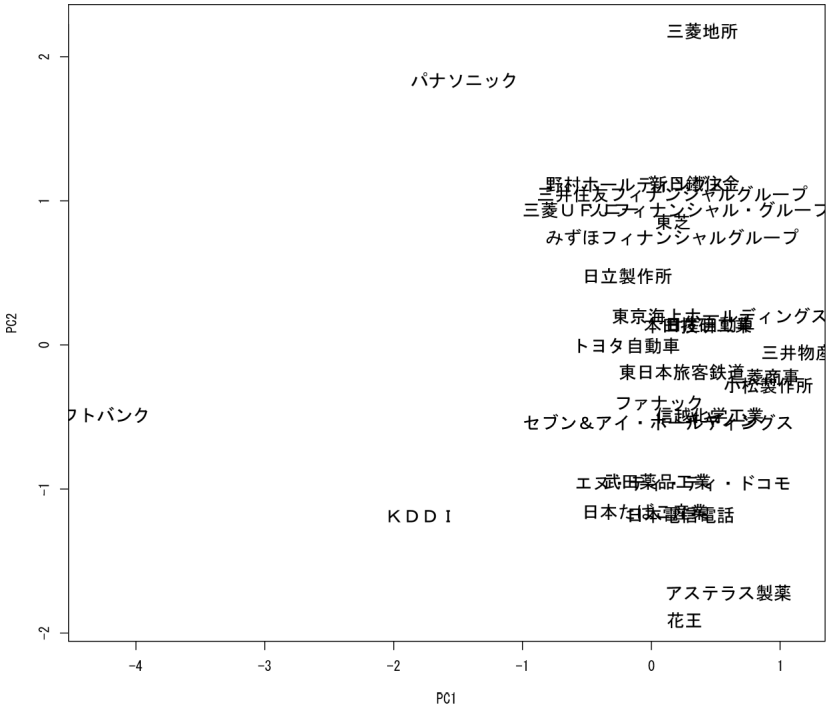
- 散布図



マンションデータ

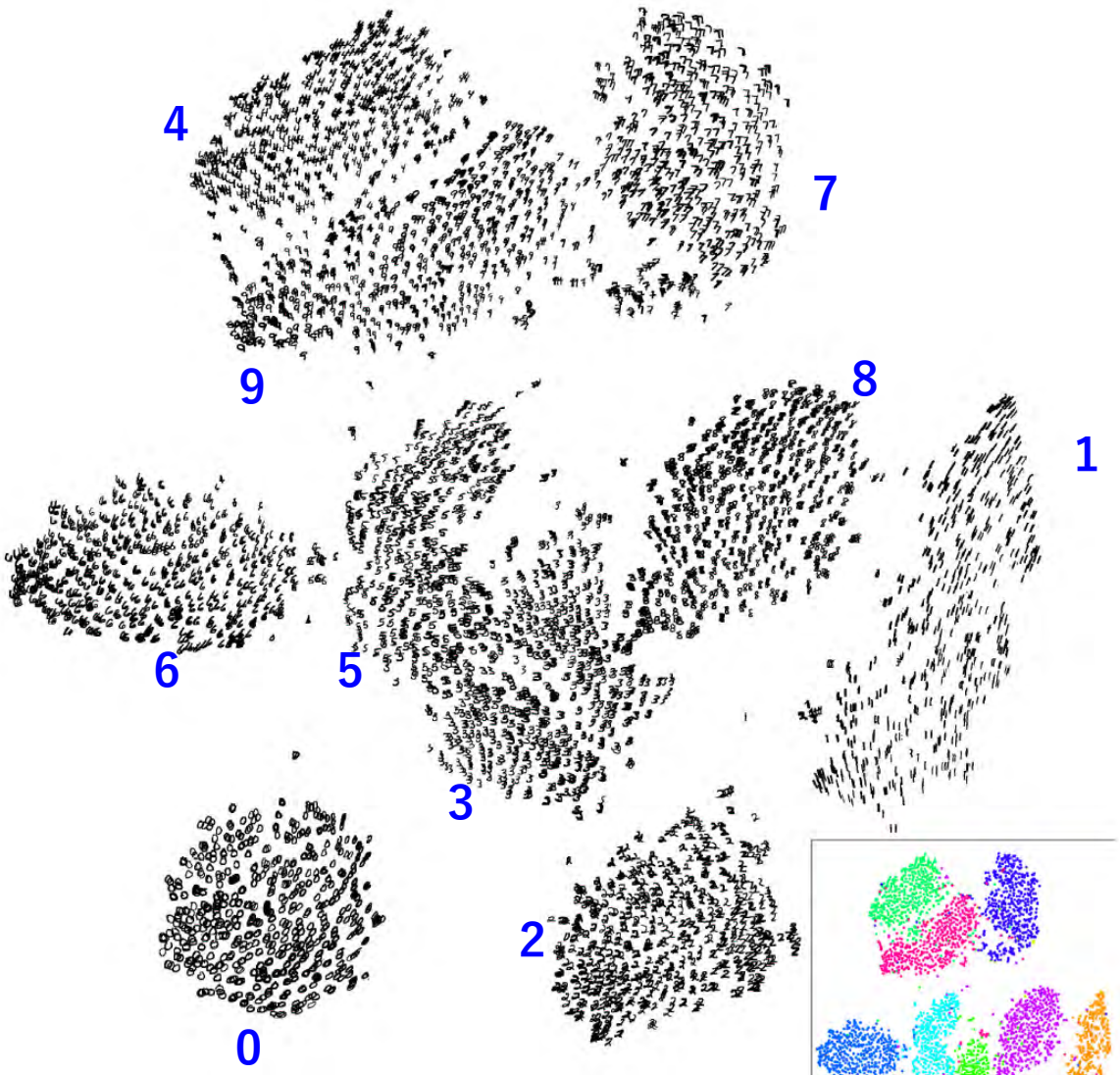
- PCA（主成分分析）

高次元データを低次元に可視化



TOPIX CORE 30
250日分の株価変動を用いて主成分分析
(250次元→2次元)

t-SNE



[van der Maaten, Hinton:
Visualizing Data using t-SNE. JMLR2008.]

MNISTデータセット

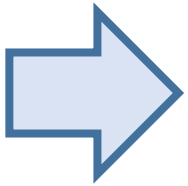
手書き文字の分布を可視化
28x28=784ピクセル→ 2次元

異常検知

「異常」とは

データ・状況

- センサーデータ
  - 機械稼働状況のセンサー情報
  - 工場・プラントのセンサー情報
- イベントデータ
  - クレジットカード利用履歴
- ネットワーク利用データ
  - トラフィック流量
  - アクセスログ
  - パケット経路
- 動画像データ
  - 監視カメラ動画
- 生体情報
  - 検診結果
  - 遺伝子データ
- ソーシャルネット
  - 新聞・ニュース
  - twitterタイムライン



「異常」

- センサーデータ
  - 機械の故障
  - 工場の故障
- イベントデータ
  - クレジットカード不正利用
- ネットワーク利用データ
  - DDoS攻撃
  - 不正侵入
  - マルウェア感染
- 動画像データ
  - 侵入者
- 生体情報
  - 後天的疾患
  - 遺伝子異常
- ソーシャルネット
  - 流行
  - 人間関係の変化

異常検知への機械学習的アプローチ

異常検知の難しさ：

異常データの数が少ない．手元にあるデータは大体正常．

アンバランスデータ

- 教師あり学習：

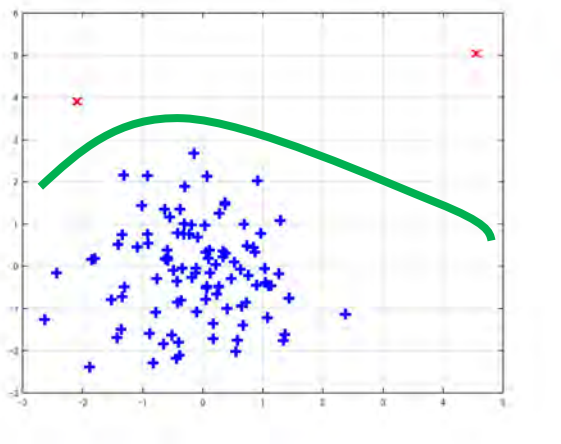
  異常/正常データがともにたくさん取れる時

  異常データと正常データを分けて通常の教師あり学習

- 教師なし学習：

  異常データがほとんど取れない時

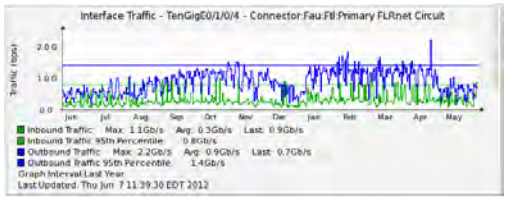
  正常データのみを用いて「異常度」のスケールを学習



教師なし学習による異常検知の基本手順

1．データを適切な特徴量で記述：
　　専門家の知識が必要．問題異存．完全には自動化できない部分．

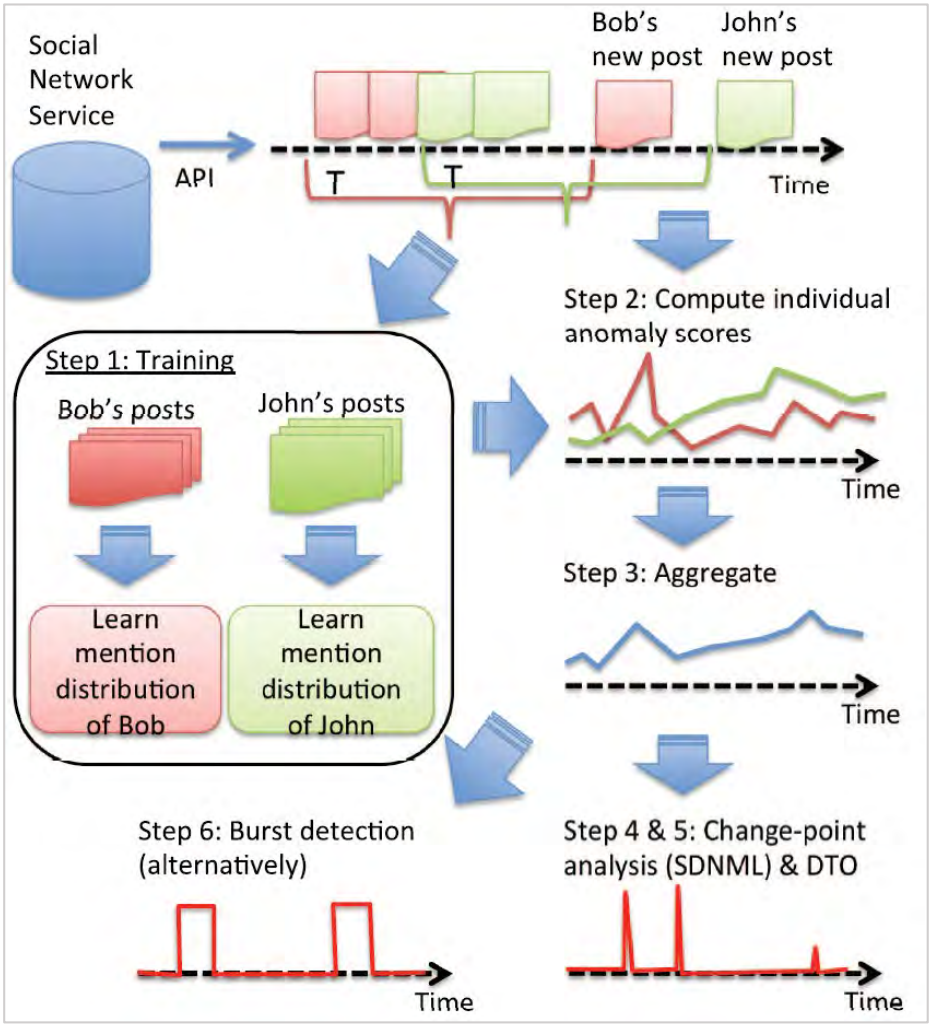
例：エンジン出力　　トラフィックの量　　通行人の数

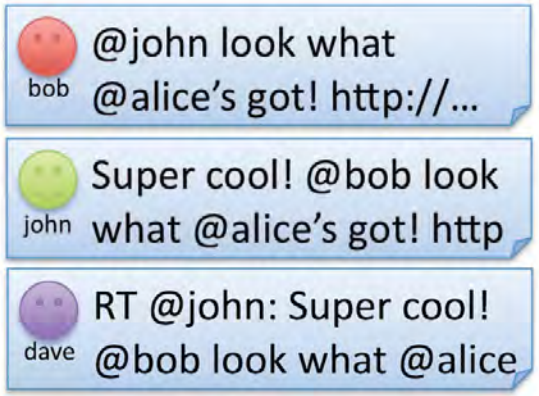


2．何らかの尺度で，正常データの分布からの離れ具合を計算
　　→　「異常度」
3．異常度がある閾値を超えたらアラートを出す．

機械学習の研究：異常度尺度の設計とその学習方法
　　→　各種手法は「何らかの尺度」の違い
実応用の研究：特徴量の設計
教師ありデータがあるなら，それを最大限活用するべき

応用例：SNSからの新しい話題検知



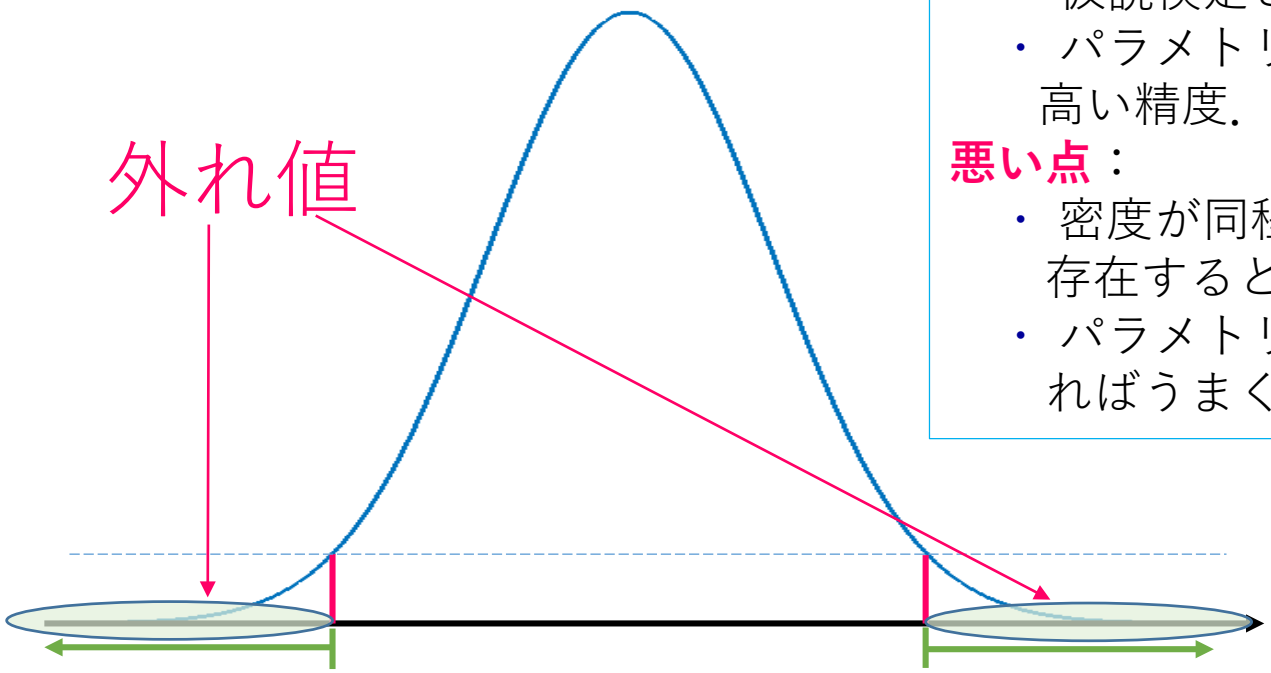


[Takahashi, Tomioka, Yamanishi: Discovering Emerging Topics in Social Streams via Link-Anomaly Detection. IEEE-TKDE2014]

- ユーザーのmentionリンクの変化から話題の変化を検知
- 使われている単語の変化から検知

教師なし異常検知の方法（１）
密度関数を用いた方法

正常データから確率密度を推定：$\hat{p}(x)$
新しいデータ点xで密度が小さければアラートを出す：$\hat{p}(x) \leq \delta$.



良い点：
- 仮説検定と深い関係.
- パラメトリックモデルで推定すれば高い精度.　（代表例：ガウシアン）

悪い点：
- 密度が同程度に低い領域が広く存在すると失敗.
- パラメトリックモデルが正しくなければうまくいかない.

例：インサイダー脅威の検出

教師なし異常検知の方法（2）
MT方法

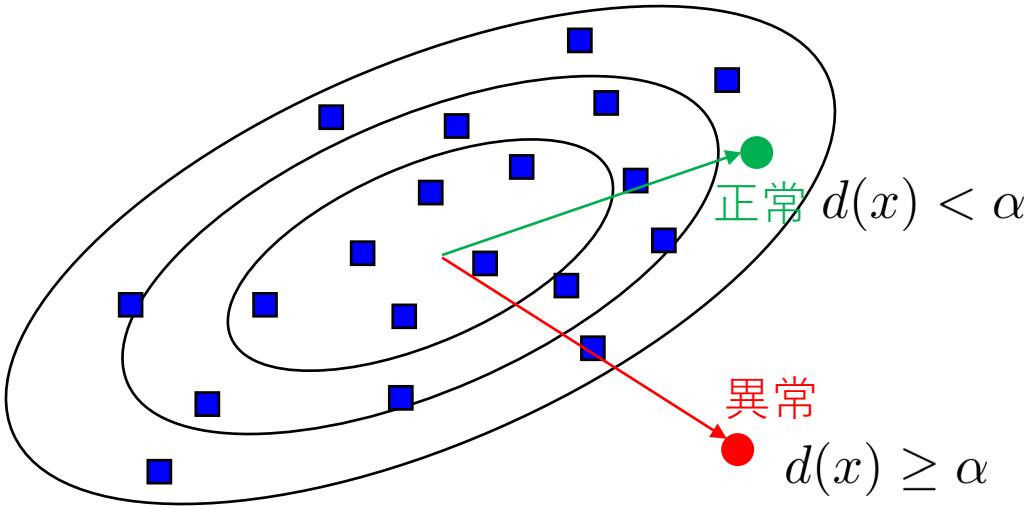
マハラノビス-タグチ法

- 異常度合をマハラノビス距離で測る.

$$\Sigma = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)(x_i - \mu)^\top$$

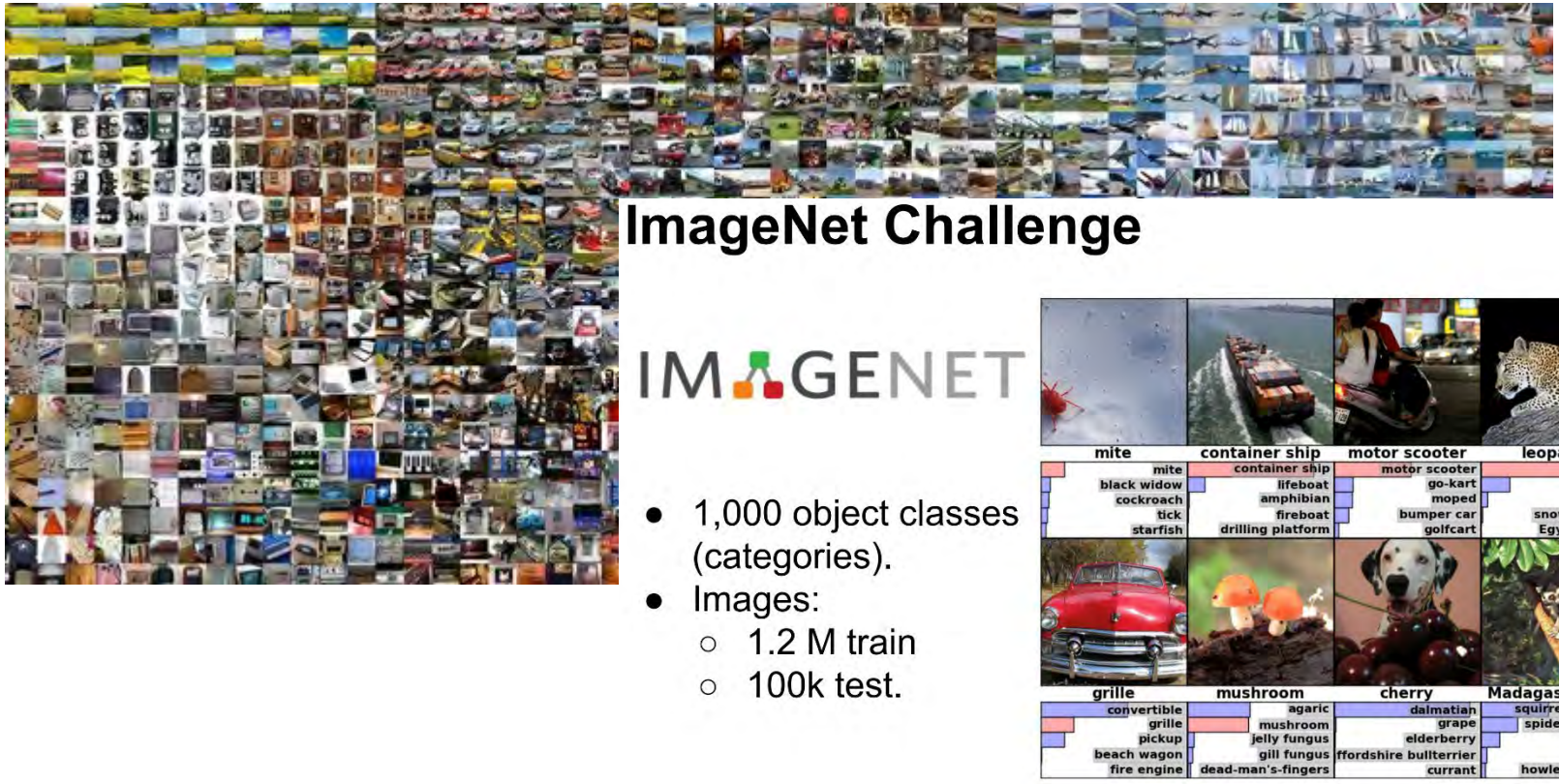
$$d(x) = \|x - \mu\|_\Sigma = \sqrt{(x - \mu)^\top \Sigma^{-1}(x - \mu)}$$

分布として多変量正規分布を仮定した密度関数を利用した方法に対応



Deep Learning
深層学習

ImageNet

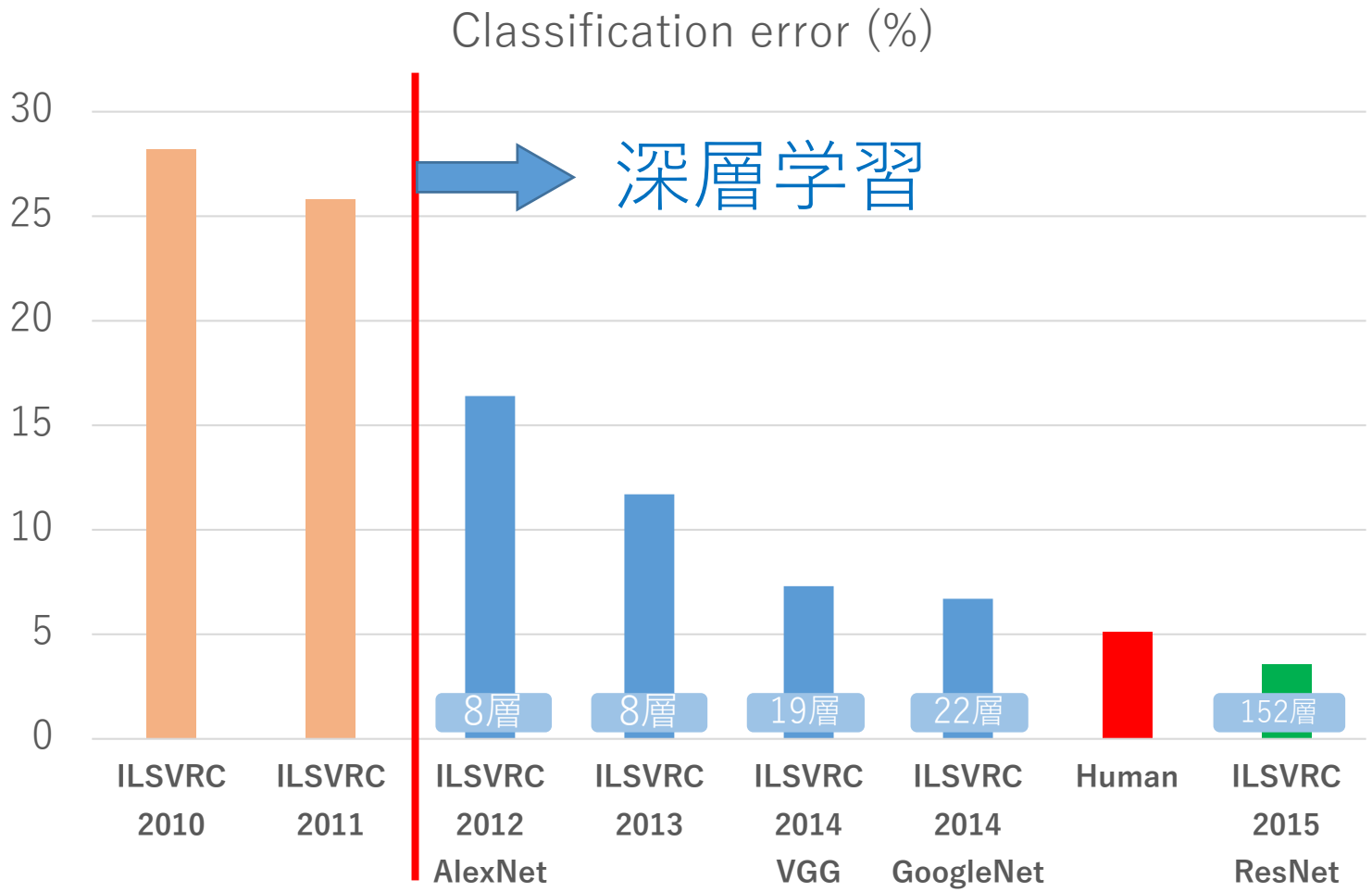


- 1,000 object classes (categories).
- Images:
  - 1.2 M train
  - 100k test.

ImageNet: 1,000カテゴリ，約120万枚の訓練画像データ

[J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei.
ImageNet: A Large-Scale Hierarchical Image Database. In CVPR09, 2009.]

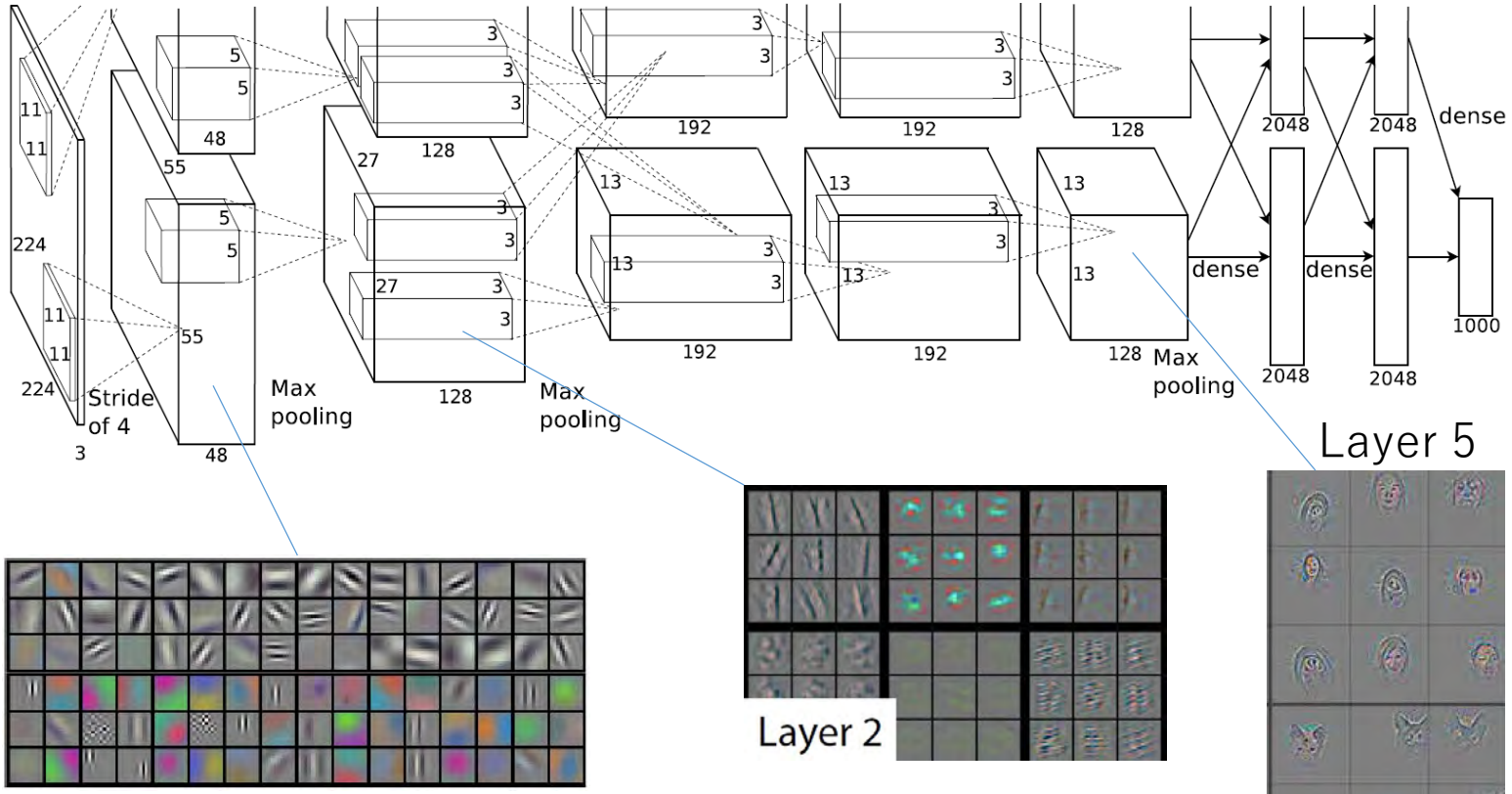
ImageNetデータにおける識別精度の変遷



ImageNet: 21841クラス，14,197,122枚の訓練画像データ
そのうち1000クラスでコンペティション

Alex-net [Krizhevsky, Sutskever + Hinton, 2012]

畳み込みニューラルネットを 5 層積み重ね（+pooling+3層の全結合層）

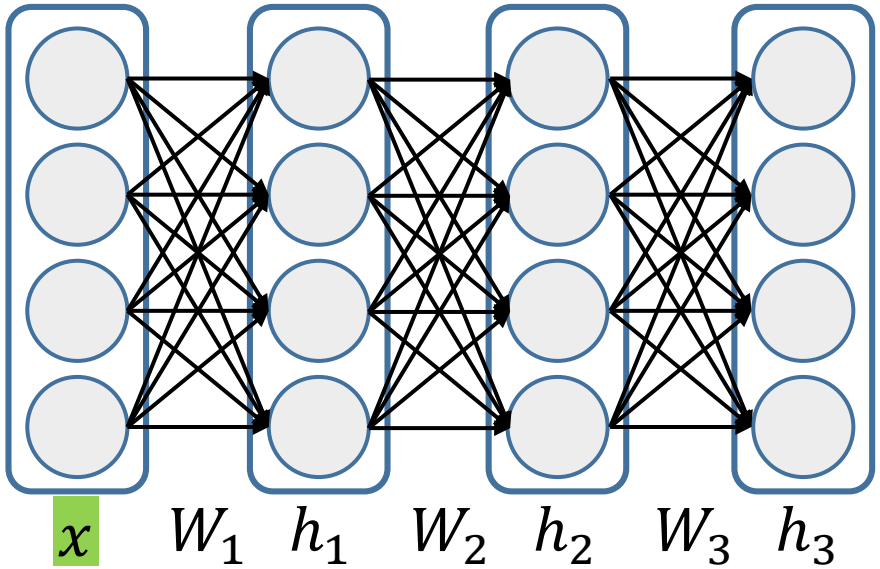


イメージパッチのようなものが学習されている

⇒　特徴量の自動学習

中間層ではより抽象的な情報がコードされる

深層学習の構造



基本的に「線形変換」と「非線形活性化関数」の繰り返し．

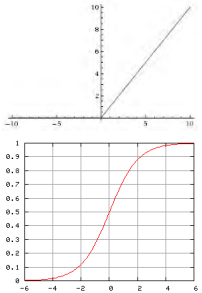
x → W_1x → $h_1(W_1x)$ → $W_2h_1(W_1x)$ → $h_2(W_2h_1(W_1x))$

入力　線形変換　非線形変換　線形変換　非線形変換

$$h_1(u) = [h_{11}(u_1), h_{12}(u_2), \ldots, h_{1d}(u_d)]^T$$

活性化関数は通常要素ごとにかかる．Poolingのように要素ごとでない非線形変換もある．

- ☆ReLU (Rectified Linear Unit)：　$h(u) = \max\{u, 0\}$
- シグモイド関数：　$h(u) = \dfrac{1}{1+e^{-u}}$



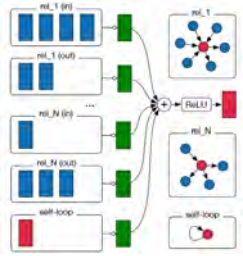
諸分野への波及

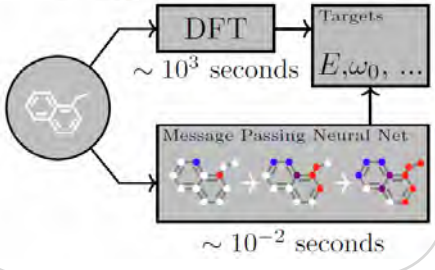
ロボット

[タオル畳み、サラダ盛り付け　「指動く」ロボット初公開, ITMedia:http://www.itmedia.co.jp/news/articles/1711/30/news089.html]

量子化学計算，分子の物性予測

$$h_i^{(l+1)} = \sigma\left(\sum_{r\in\mathcal{R}}\sum_{j\in\mathcal{N}_i^r}\frac{1}{c_{i,r}}W_r^{(l)}h_j^{(l)} + W_0^{(l)}h_i^{(l)}\right)$$

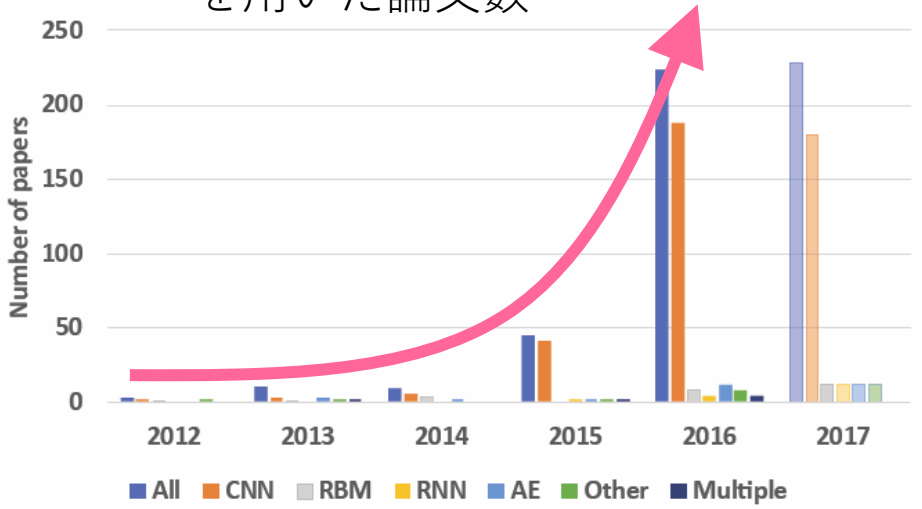




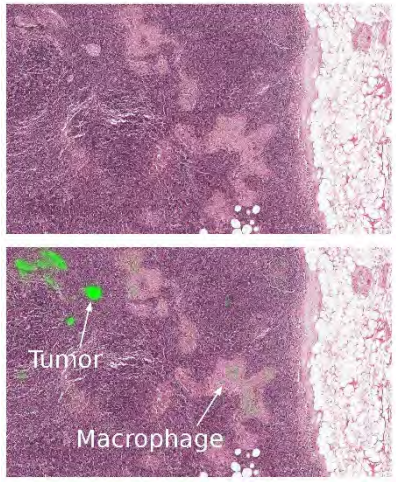
[Niepert, Ahmed&Kutzkov: Learning Convolutional Neural Networks for Graphs, 2016]
[Gilmer et al.: Neural Message Passing for Quantum Chemistry, 2017]
[Faber et al.:Machine learning prediction errors better than DFT accuracy, 2017.]

医療

医療分野における「深層学習」を用いた論文数



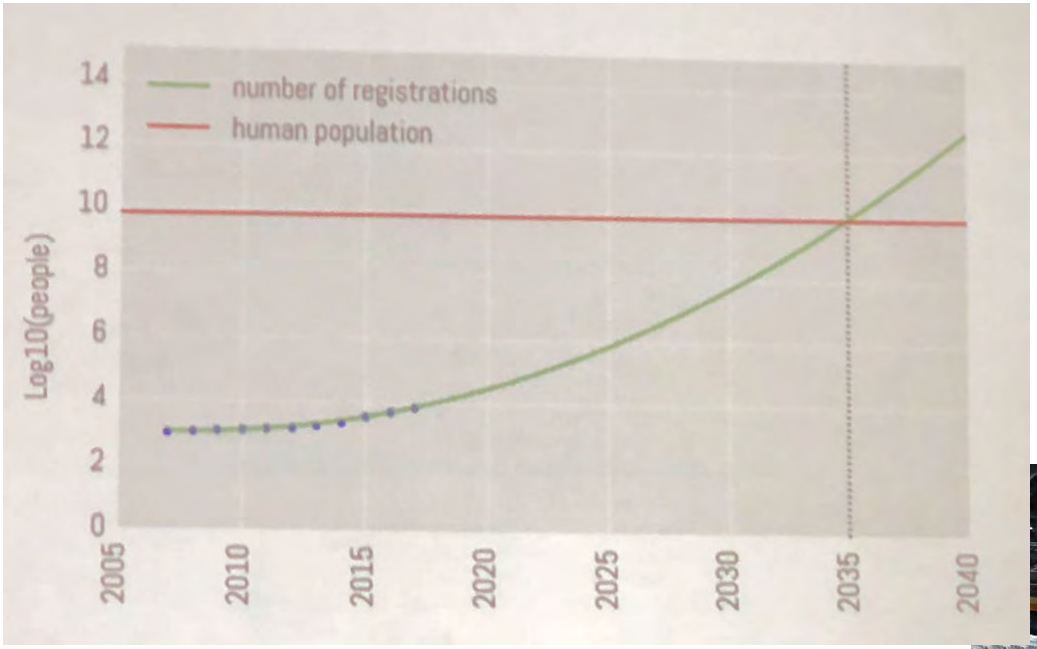
[Litjens, et al. (2017)]



- 人を超える精度
  （FROC73.3% -> 87.3%）
- 悪性腫瘍の場所も特定

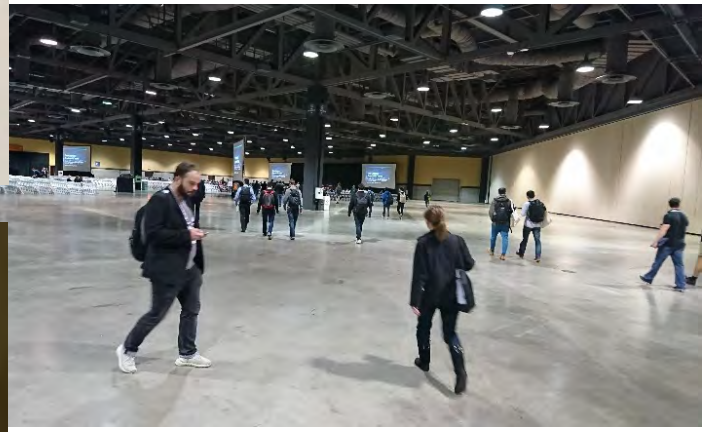
[Detecting Cancer Metastases on Gigapixel Pathology Images: Liu et al., arXiv:1703.02442, 2017]

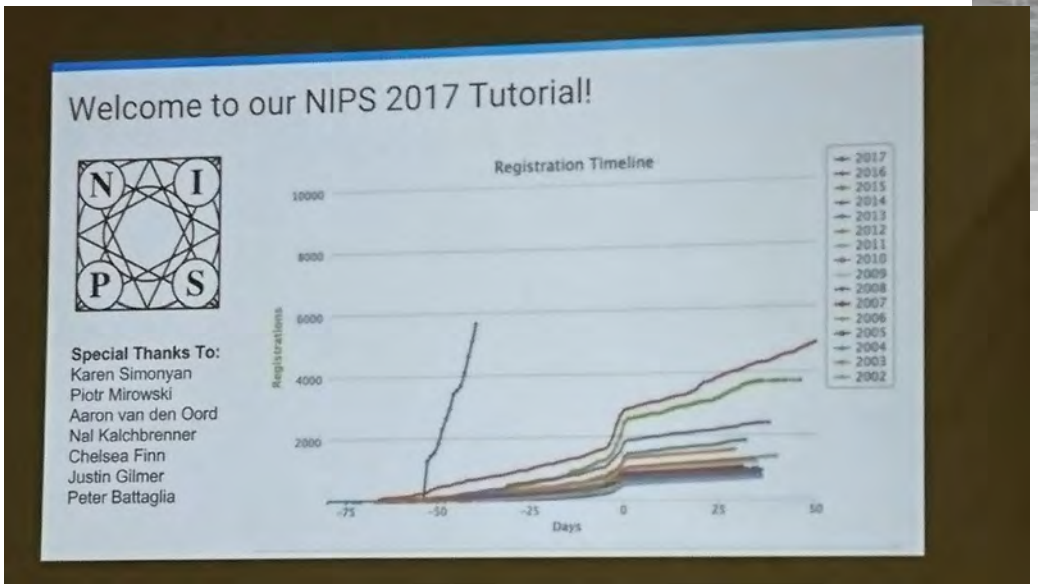
DLの学会への影響：NIPS2017の状況



参加者の増加

会場の様子





参加者登録数の遷移
２週間で売り切れ

NIPS2018の状況

- 投稿数：4856件
- 採択数：1011件
- 査読プロセス：double blind

(Senior) Area Chair: 約280人，査読者：約2500人

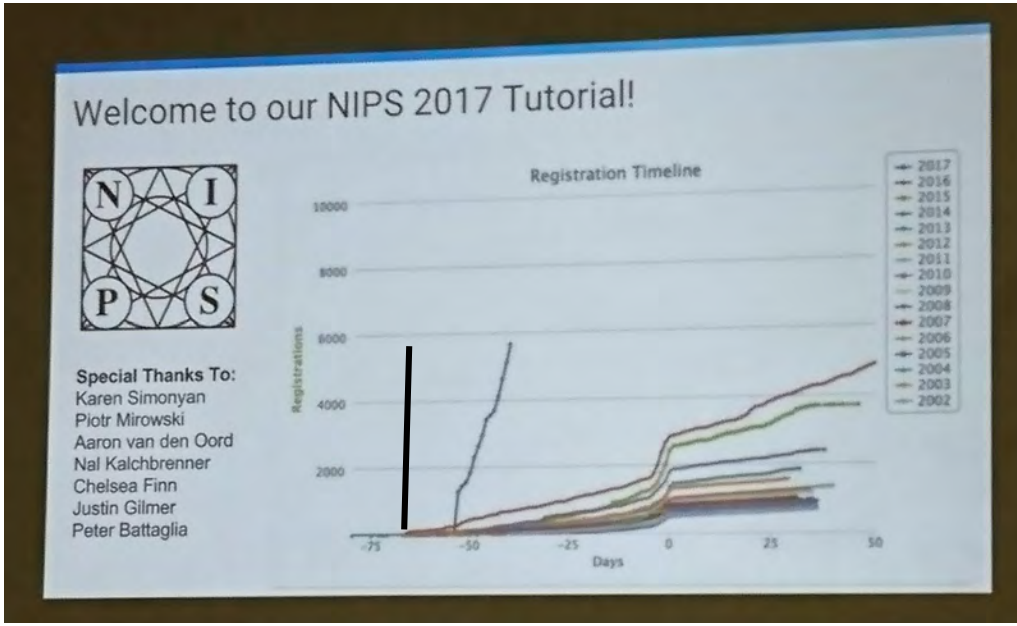
✓1査読者あたり6本を査読，そのうち1-2本がaccept

通常レジストレーション（2000席）

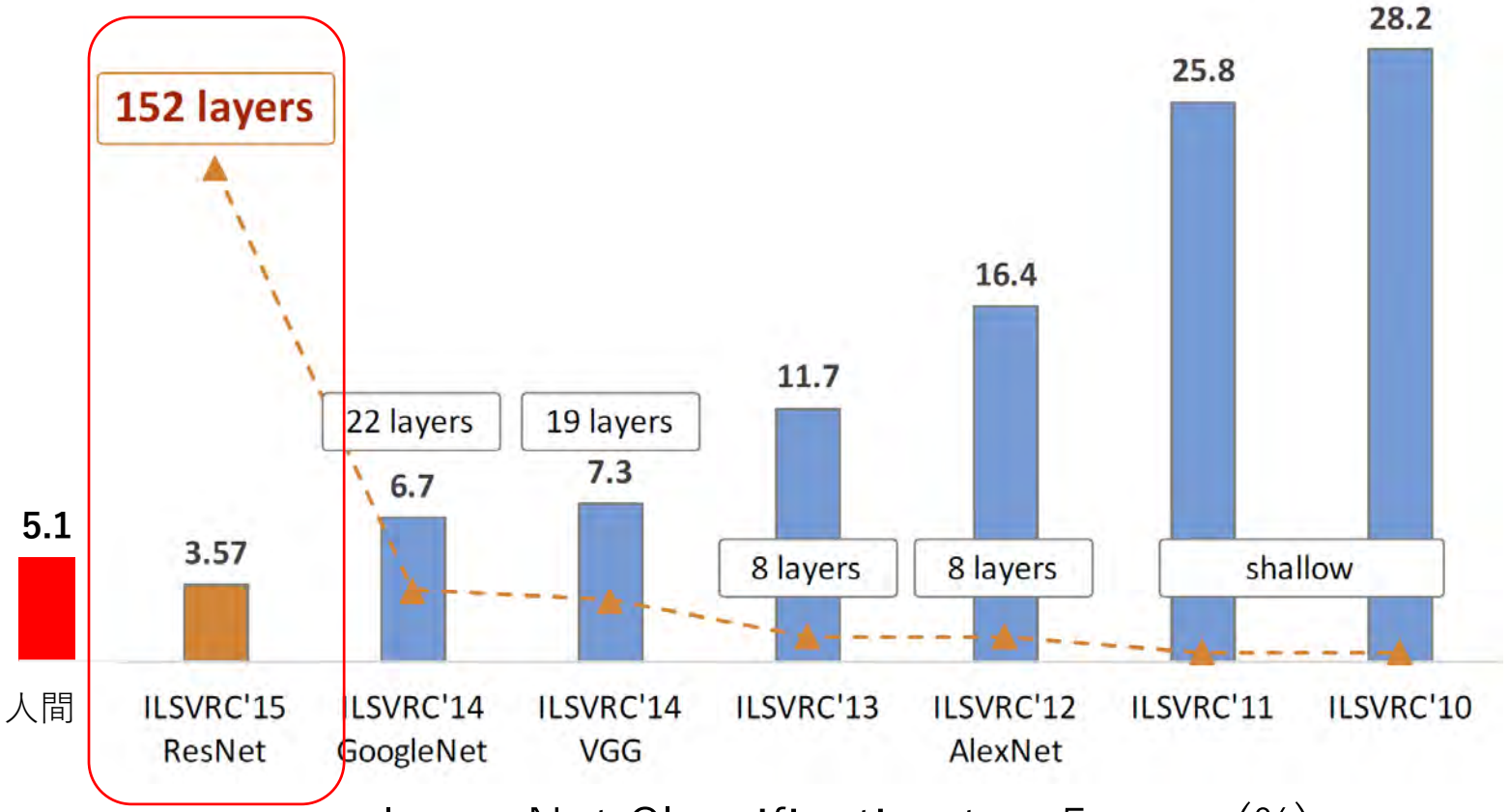
<u>11分38秒</u>で完売

（論文採録者，トップ査読者は別）

[去年は8000席が2週間で完売]



ResNet (Deep Residual Net)



ImageNet Classification top-5 error (%)

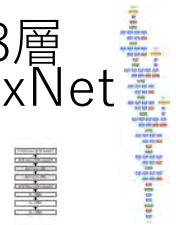
He, Zhang, Ren, & Sun. "Deep Residual Learning for Image Recognition". CVPR 2016. (CVPR2016 best paper award)

He. "Deep Residual Network". ICML2016 tutorial.

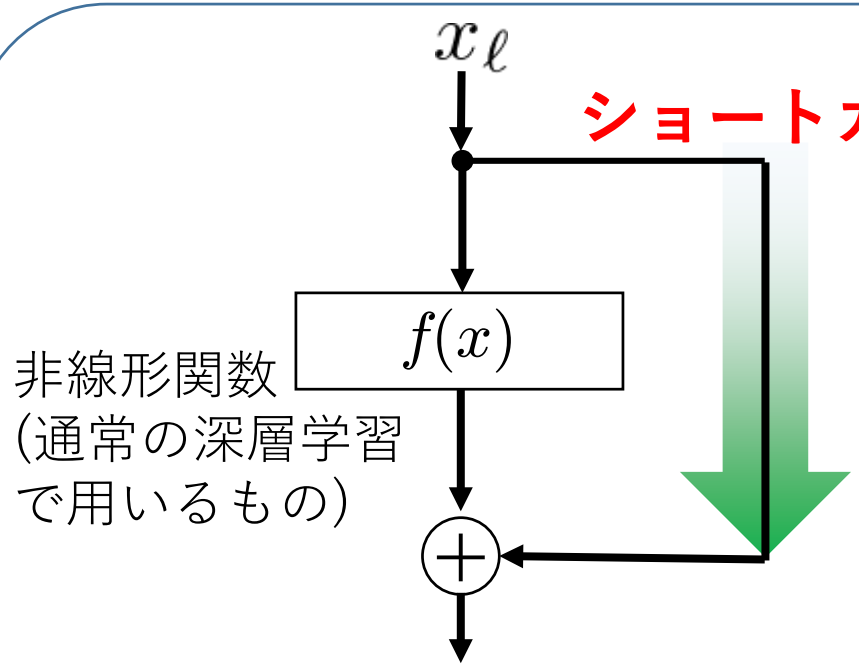
ResNet
152層

22層
GoogleNet

8層
AlexNet



ResNetの構造



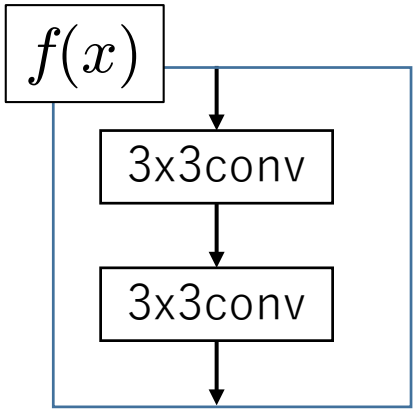
$$x_{\ell+1} = x_\ell + f(x_\ell)$$

$$x_{\ell+2} = x_\ell + f(x_\ell) + f(x_{\ell+1})$$

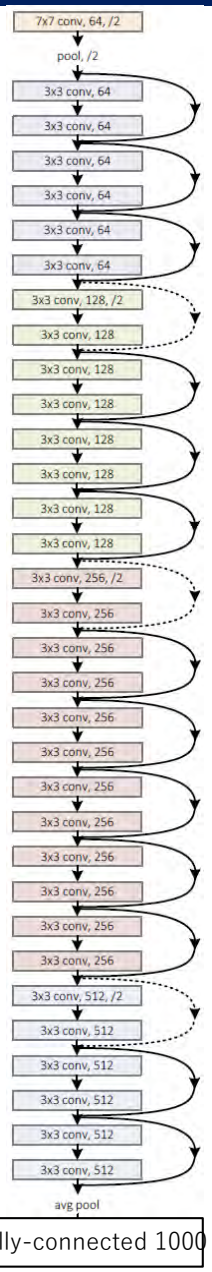
$$\vdots$$

$$x_L = x_\ell + \sum_{k=\ell}^{L-1} f(x_k)$$

情報が減衰せずに伝わる



CIFAR-10などの画像認識タスクでは$f(x)$として2層の畳み込み層を用いたものが良かった.



1000層を超えるものもある

ResNetの変種

- Stochastic Depth

  [Huang,Sun,Liu,Sedra,Weinberger: Deep Networks with Stochastic Depth, 2016]

  学習中に接続を確率的に切る.

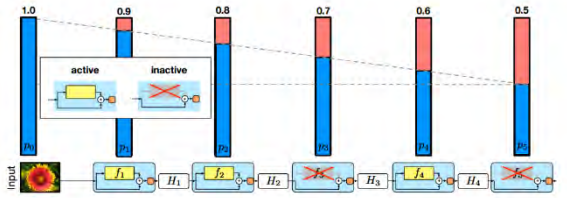
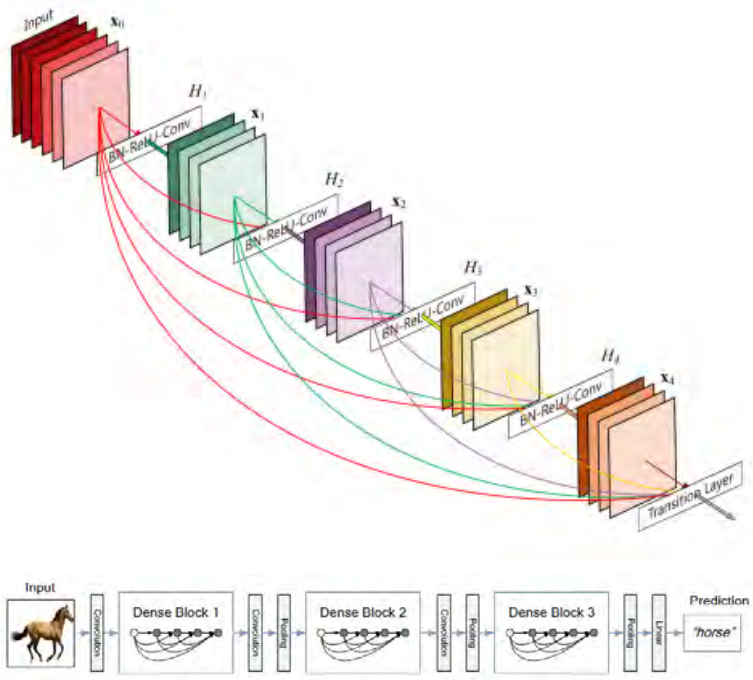


Fig. 2. The linear decay of p_ℓ illustrated on a ResNet with stochastic depth for $p_0 = 1$ and $p_L = 0.5$. Conceptually, we treat the input to the first ResBlock as H_0, which is always active.

- DenseNet [Huang, Liu, Weinberger, van der Maaten: Densely Connected Convolutional Networks, 2016]

  (CVPR2017 best paper award)

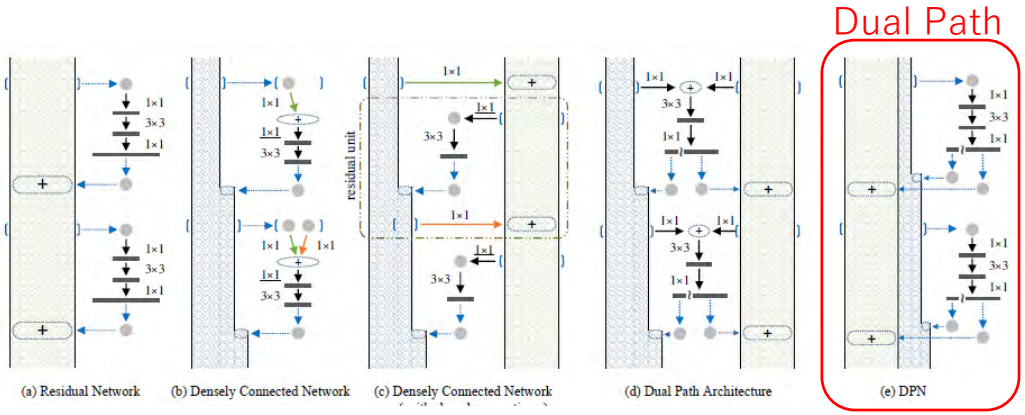
長いスキップを用いて密な結合を用いる



DenseNetの様子

- Dual Path Networks

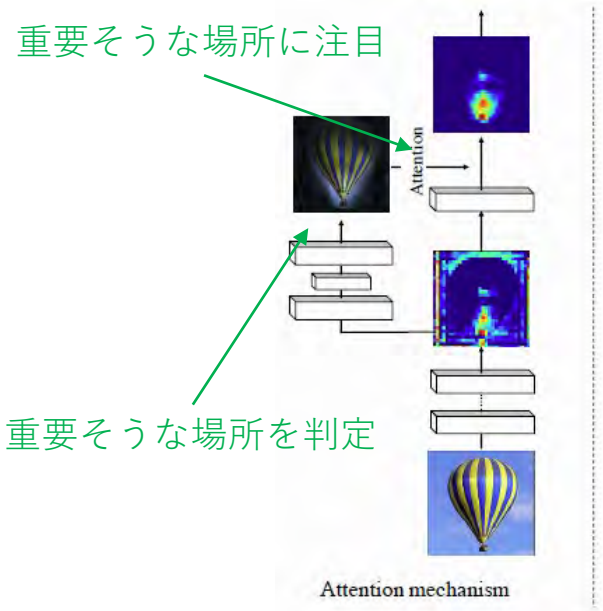
[Chen, Li, Xiao, Jin, Yan, Feng: Dual Path Networks, 2017]

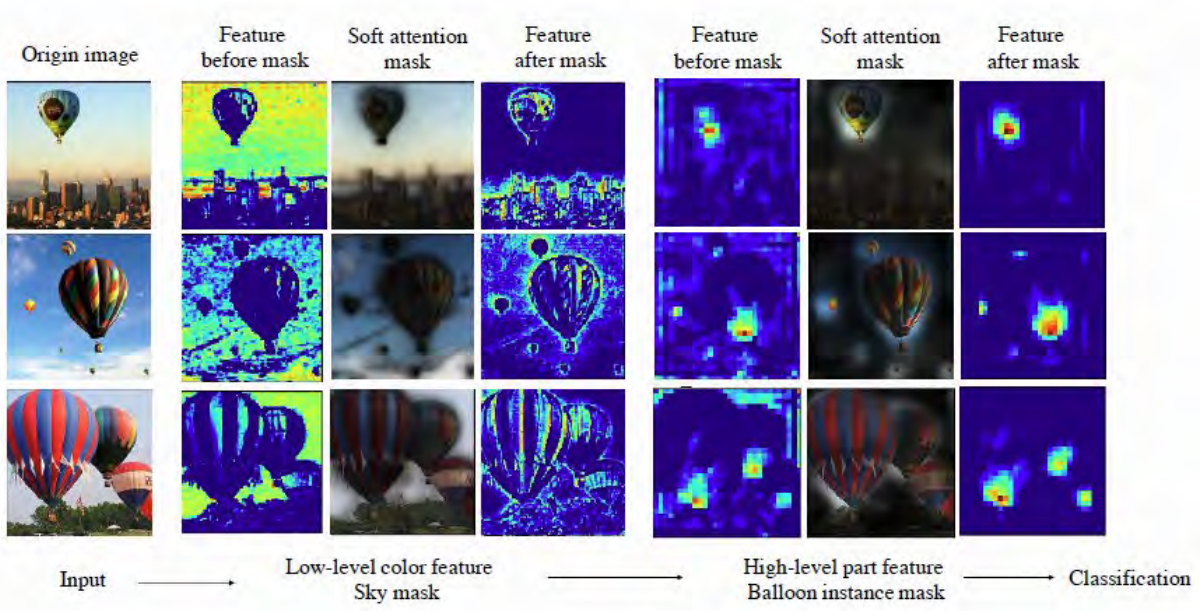


ResNetとDenseNetの良い部分を組み合わせ.
ILSVRC2017のObject localization部門で１位.

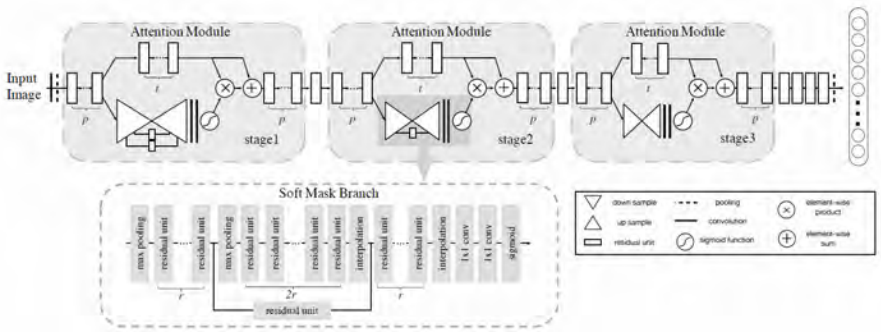
Residual Attention Network

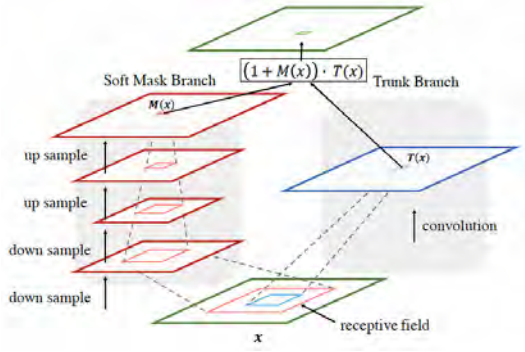
ILSVRC2017のObject detection部門１位





ResNetに選択的注意の機構を付与

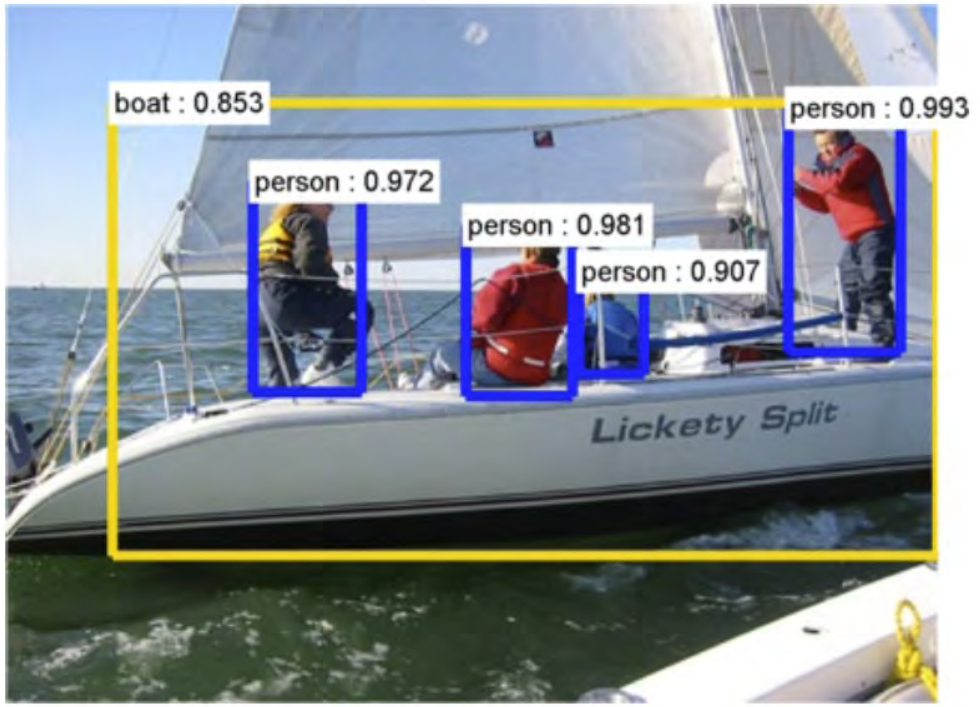




[Wang F, Jiang M, Qian C, et al. Residual Attention Network for Image Classification. arXiv:1704.06904, 2017]

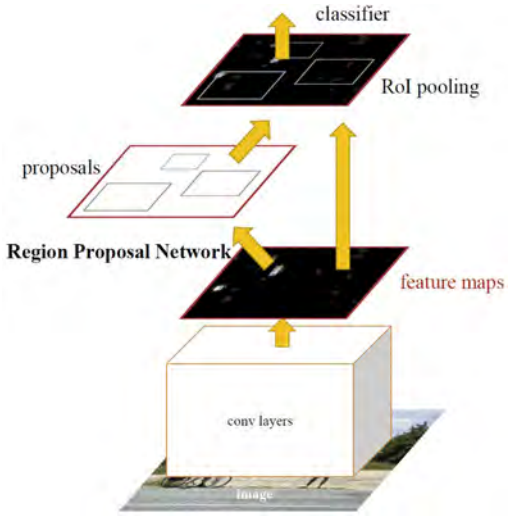
物体認識→物体検出

Faster R-CNN (2015)



「どこに」＋「なにが」
写っているか

物体認識より難しい.
物体認識：「なにが」のみ



	mAP(%)
ResNet	**48.4**
VGG	41.5

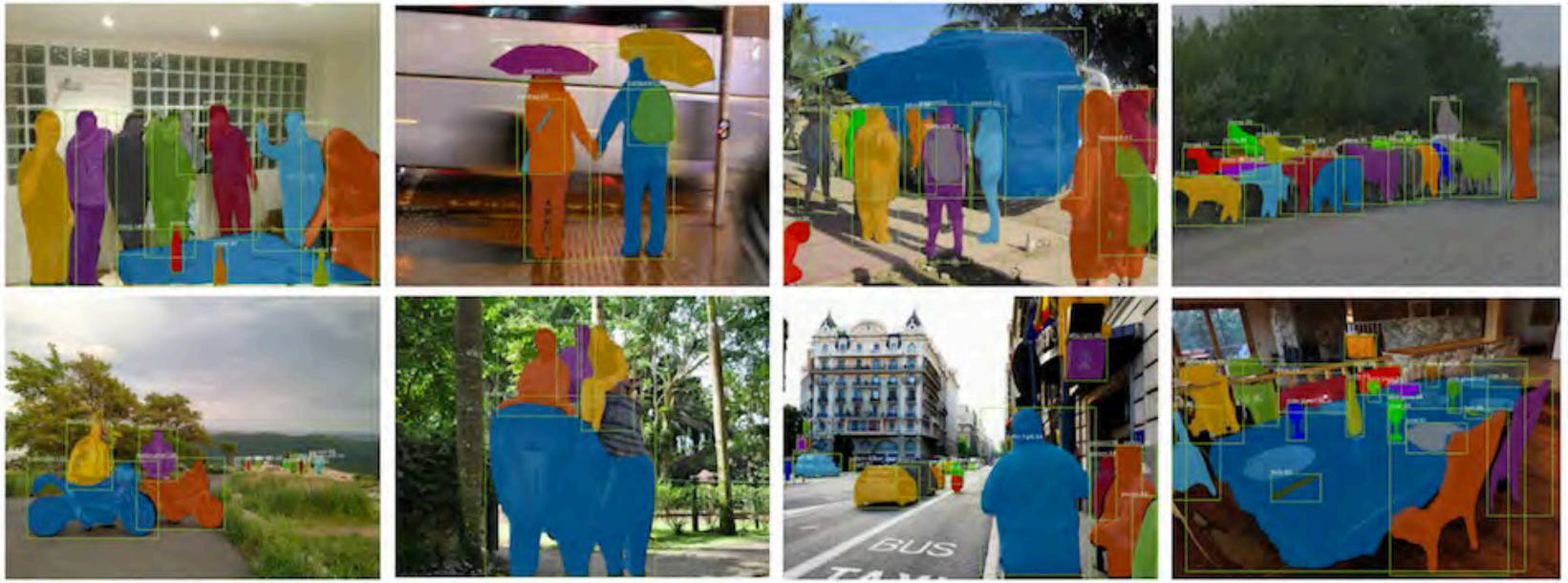
(従来)

COCO 2015 dataset

Ren, He, Girshick, & Sun. “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks”. NIPS 2015.

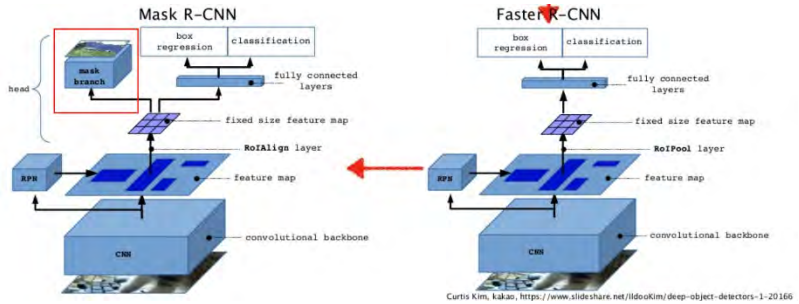
物体検出→マスキング

Mask R-CNN [He, Gkioxari, Dollár, Girshick, ICCV2017]
https://arxiv.org/abs/1703.06870



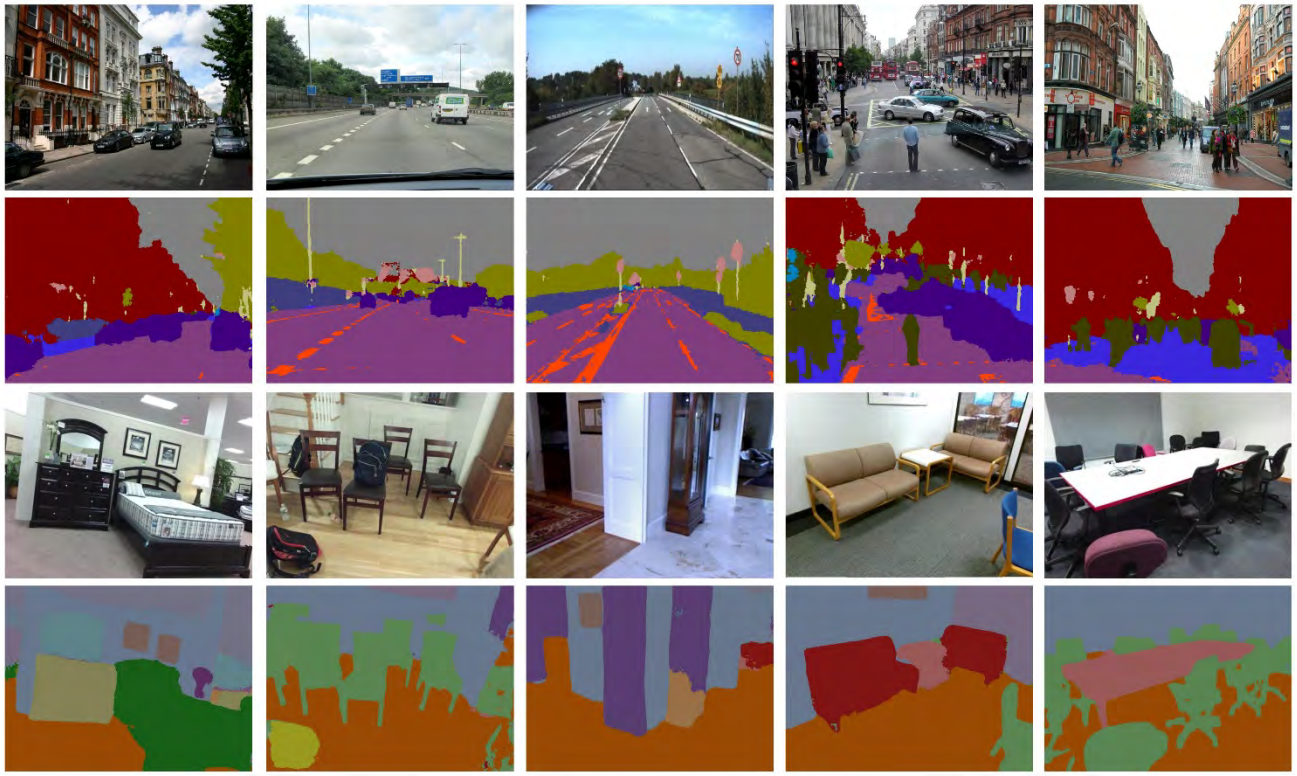
四角で囲むだけでなくマスキングも実行

人物姿勢推定も可能



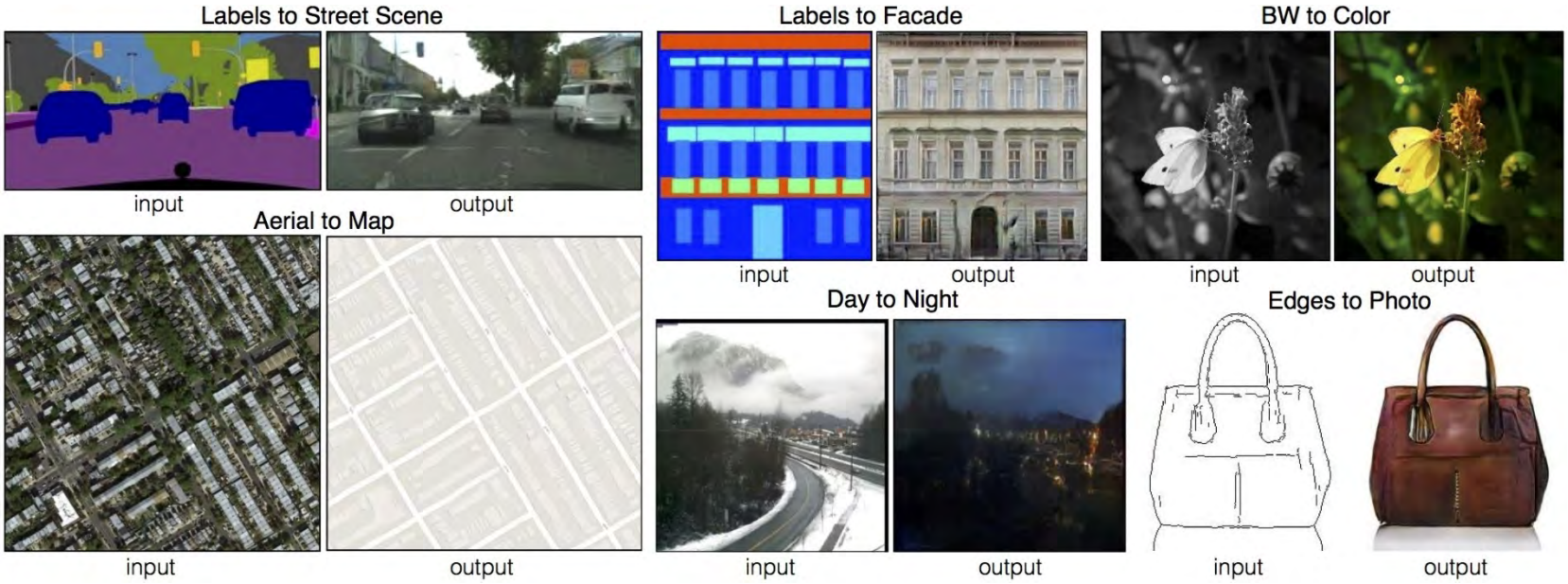
(動画も紹介)

SegNet



Badrinarayanan, Kendall, Cipolla: SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. 2015.

Pix2Pix



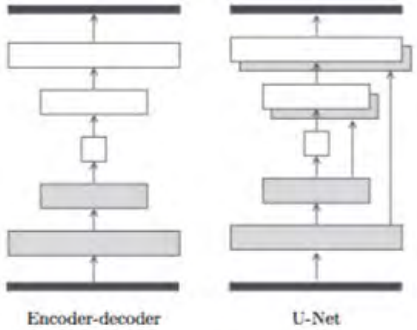
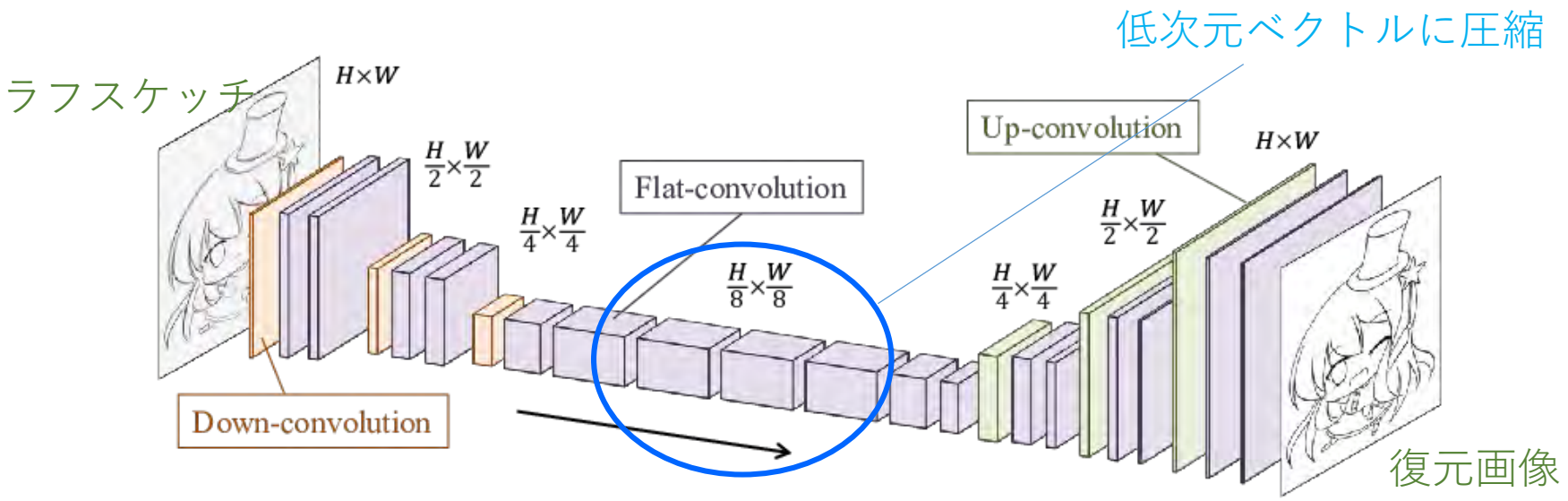


Figure 3: Two choices for the architecture of the generator. The "U-Net" [34] is an encoder-decoder with skip connections between mirrored layers in the encoder and decoder stacks.

Chainerによる自動彩色

Isola, Zhu, Zhou, and Efros :Image-to-Image Translation with Conditional Adversarial Networks. 2016

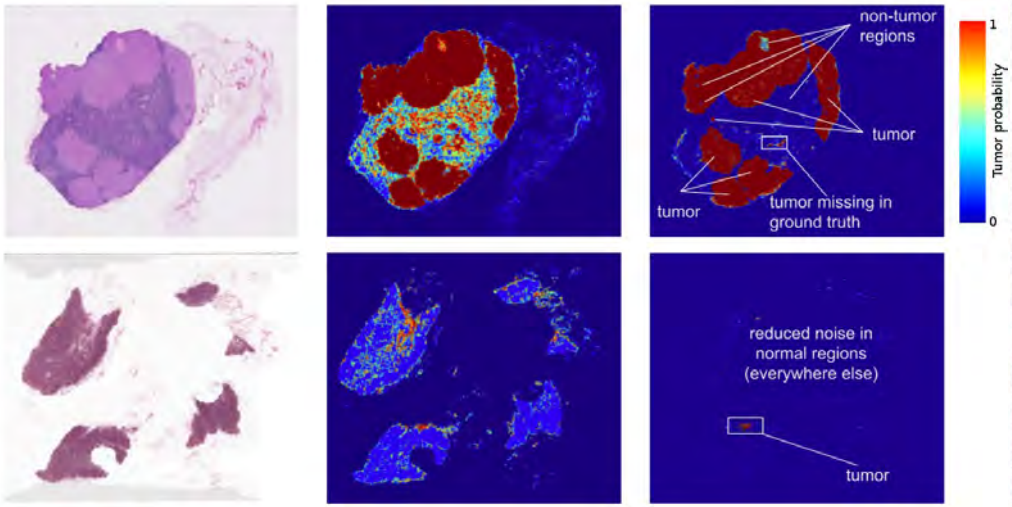
ラフスケッチの自動線画化

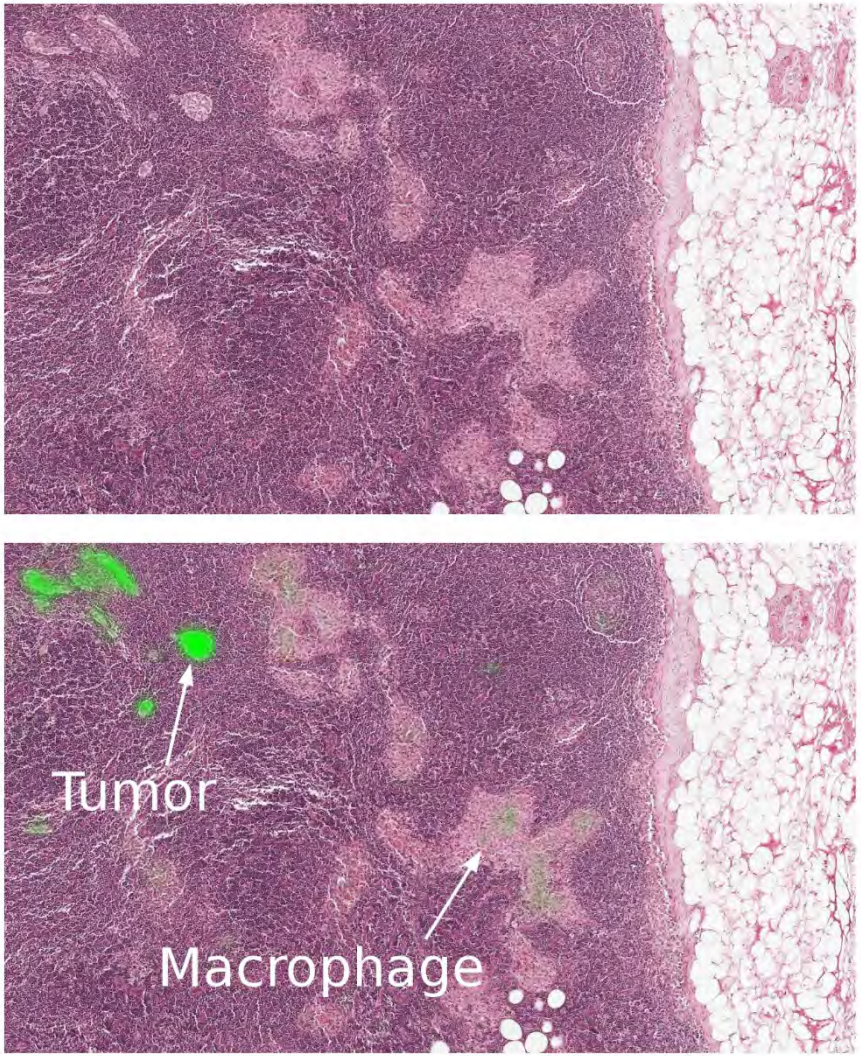


[Simo-Serra et al., SIGGRAPH2016]

高精細医用画像の分析

ギガピクセル画像の認識もできつつある.





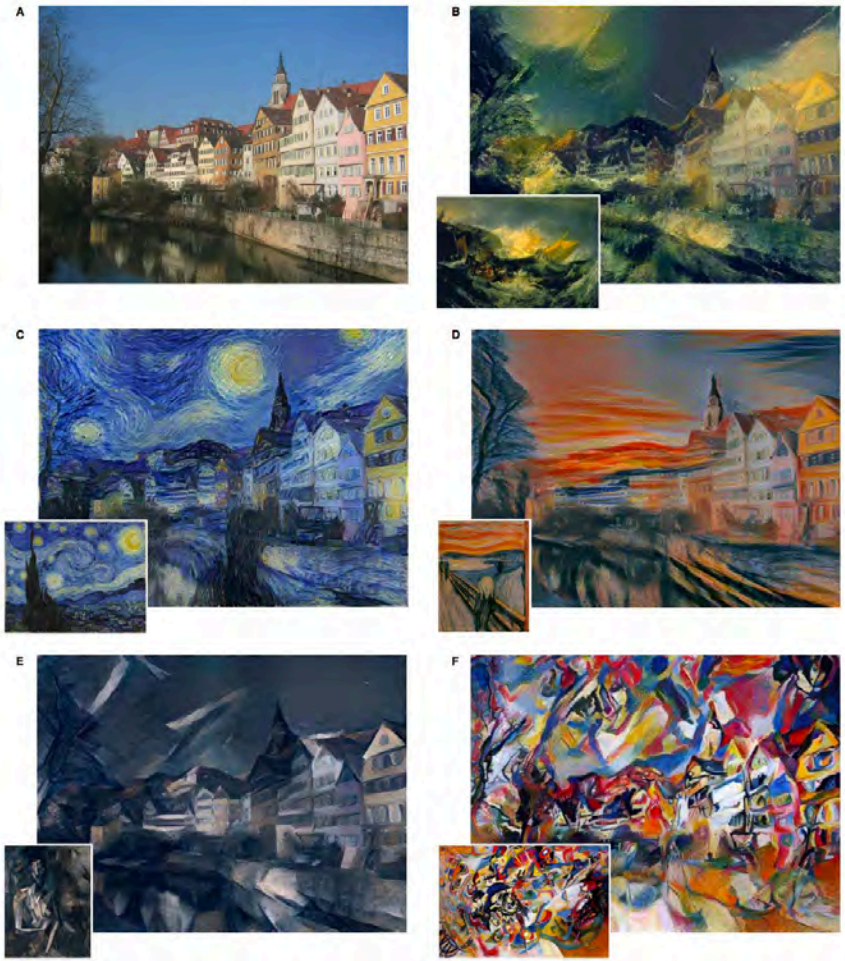
- 人を超える精度
  （FROC73.3% -> 87.3%）
- 悪性腫瘍の場所も特定

[Detecting Cancer Metastases on Gigapixel Pathology Images: Liu et al., arXiv:1703.02442, 2017]

スタイル変換

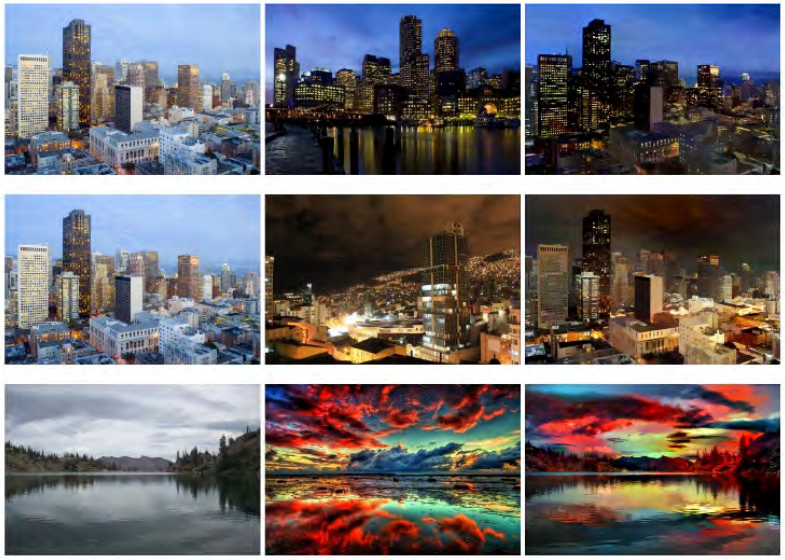
[Gatys, Ecker, Bethge : Image Style Transfer Using Convolutional Neural Networks, CVPR2016]

元画像



[Gene Kogan: Experiments with style transfer, 2015]

http://genekogan.com/works/style-transfer/



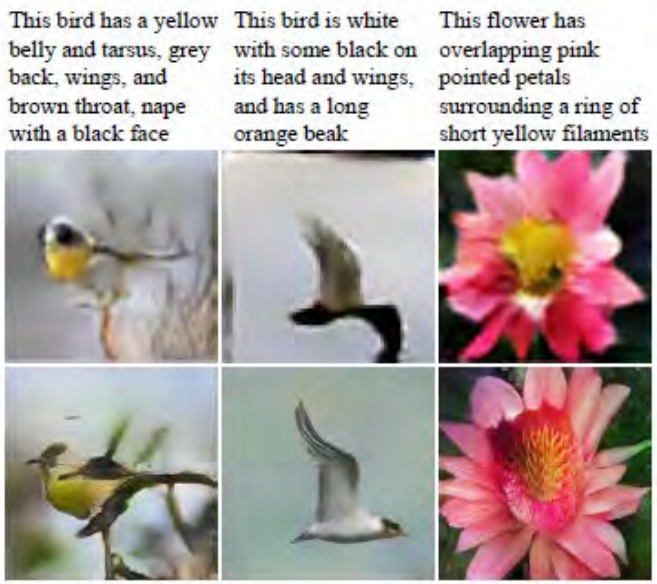
[Luan, Paris, Shechtman, Bala: Deep Photo Style Transfer, 2017] https://arxiv.org/abs/1703.07511

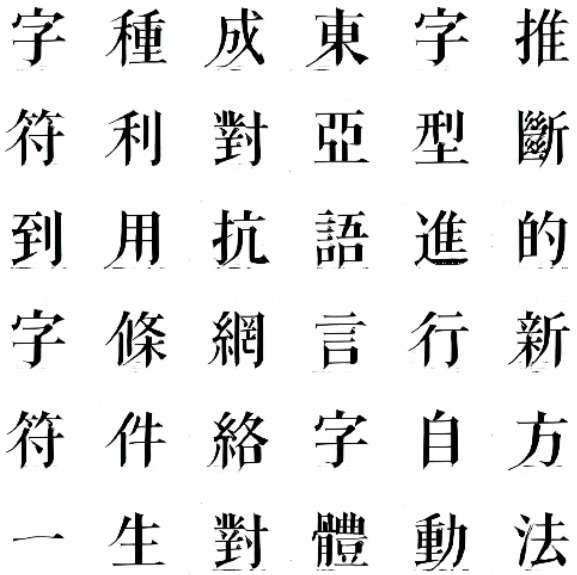
生成モデル

本物らしいデータを生成したい

深層学習が生成した画像

生成データ　　訓練データ





[Tian: zi2zi, Master Chinese Calligraphy with Conditional Adversarial Networks,2017]

CycleGAN

[Zhu et al.: Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. 2017]

生成モデル

目標：本物らしい画像を生成したい.

- GAN (Generative Adversarial Network) [Goodfellow+et al., 2014]

2つの構成要素

Generator: $x = G(z)$

Discriminator: $D(x) = P(x\text{が本物})$

G: 画像の素z (乱数) から偽画像xを生成. Dを騙そうとする.
D: 画像xが本物か偽物か判別. Gに騙されないようにする.

最適化問題

$$\min_G \max_D \mathrm{E}_{x\sim p_{\mathrm{data}}}[\log D(x)] + \mathrm{E}_{z\sim p_x}[\log(1 - D(G(z)))]$$

本当の画像を
本物と判別する確率

偽物の画像を
偽物と判別する確率

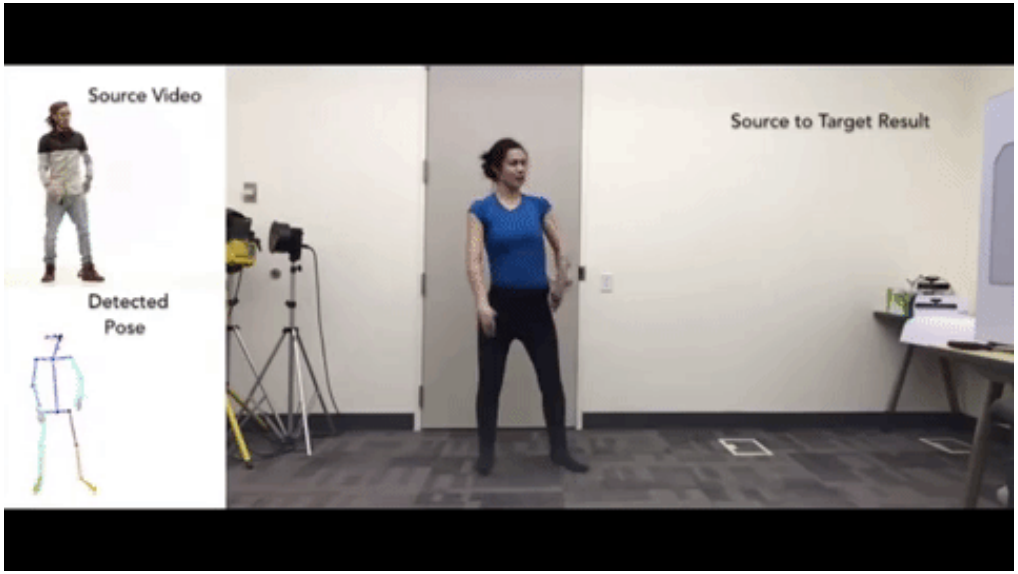
2016-2017にかなり流行
GANの変種まとめ：https://github.com/hindupuravinash/the-gan-zoo
※GANの他にもVAE (Variational Auto-Encoder)と呼ばれる方法もよく用いられている.

More applications

[Glow: Generative Flow with Invertible 1x1 Convolutions. Kingma and Dhariwal, 2018]

Crypko, 2018

条件づけた画像生成

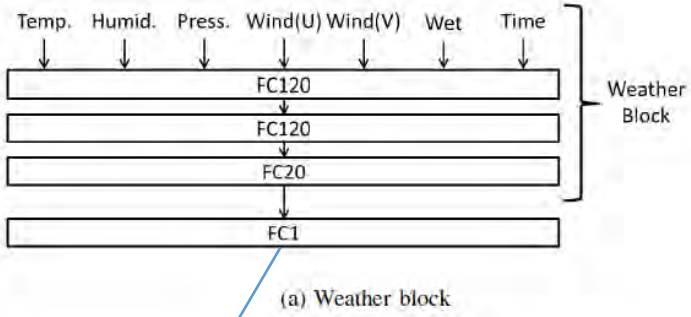


[Chan, Ginosar, Zhou and Efros: Everybody Dance Now. https://arxiv.org/abs/1808.07371, 2018]

深層学習による天気予報

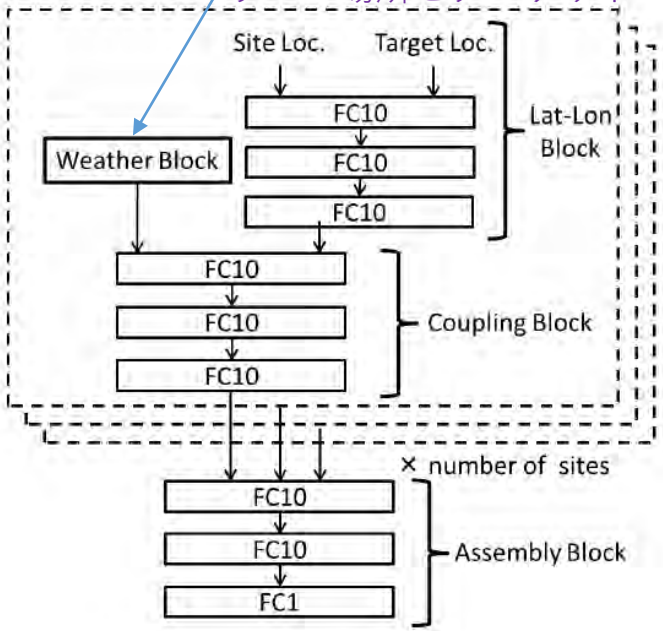
短期的天気予報の深層ニューラルネットワークモデル

過去の気温，湿度，降雨などを入力

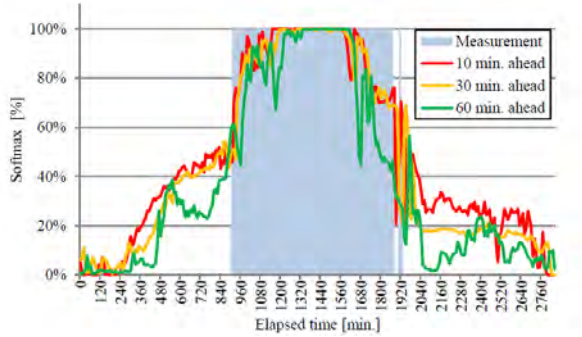


(a) Weather block

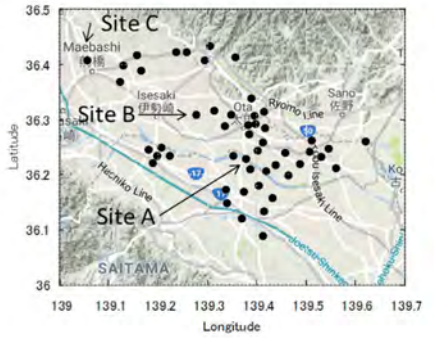
センサーの場所とターゲットの場所を入力



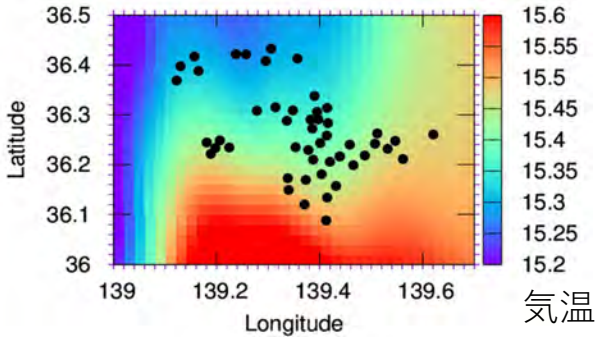
(b) Tensor lerning block



降雨の短期予測



センサーの場所



気温の予測・保管

[Yonekura, Hattori, and Suzuki: Short-term local weather forecast using dense weather station by deep neural network. IEEE BigData 2018]

さらに知りたい場合

30 AMAZING APPLICATIONS OF DEEP LEARNING

http://www.yaronhadad.com/deep-learning-most-amazing-applications/

まとめ

- 機械学習の各種手法を紹介
  - 教師あり学習
    - 回帰，判別
  - 教師なし学習
    - クラスタリング，word2vec，関係データ解析，異常検知
  - 深層学習

- データから背後の構造を学習
- 構造を数式で表した「モデル」をデータにフィット
  - 「機械学習手法」はモデルと学習手法の組
- 正しく使うための正しい理解
  - 「過学習」「予測と推測」「データ形式」...