

ニコニコAIスクール 第4回

scikit-learn

18/01/27

講師：電通国際情報サービス (ISID)

小川雄太郎

質問:Slack

前回お休みされた方はいますか？



はい



いいえ

質問：チームメンバーの ①研究室・研究内容、②科学的な興味の対象、③なぜAIスクールに参加したのか、AIスクールを通して何ができるようになりたいのか？④呼び名、を知っていますか？

1. (自己紹介)、前回のコメント・質問への回答
2. 本日の講義概要
3. 講義前半：13:15-14:30 (14:30-14:45休憩)
4. 講義後半：14:45-16:30

質問はチャネルlec_3_4_ogawaにバンバン貼ってください。
Slackで改行はALT+ENTERです。遠隔の人も同様をお願いします。
※Zoomは履歴が消えてしまうので・・



小川雄太郎（おがわさん、ゆたろさん）
明石高専→東大工学部精密工学科（B）
→東大新領域（M,D）→東大先端研（PD）
→電通国際情報サービス 開発技術部

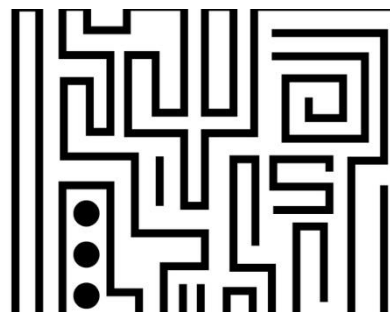
- 神保・小谷研究室で、NIRSと脳波の計測からワーキングメモリ能力の推定、神経細胞集団の数理モデルを縮約して同期現象の解析などを研究。2016年博士号（科学）を取得。
- 現在はディープラーニングをはじめとした機械学習の研究開発・案件支援および社内データ解析に従事
- 元脳科学若手の会代表(2012年)

興味があれば、いつか見てみてください。

- マイナビ出版の技術サイトManateeで強化学習の連載中

<https://book.mynavi.jp/manatee/series/detail/id=87626>

作りながら学ぶ!
強化学習
初歩からPyTorchによる
深層強化学習まで



- Qiita <https://qiita.com/sugulu/>



ユーザーランキング

週間 月間 全て



すぐる
@sugulu

1362

Contributions



株式会社電通国際情報サービス（ISID）

1975年に電通とGEのジョイントベンチャーとして設立されて以来、先進的な情報技術をベースに、アイデアとクリエイティビティを掛け合わせたユニークなIT専門家集団として成長。2000年に東証1部上場。



アラヤさんと一緒に、人工知能による画像解析で養殖マグロの個体数を自動カウント



<https://www.isid.co.jp/case/?all>

2. 前回のコメントに回答

8

講義の進め方

解析解やバイアス項について数式的にあまり理解していないままコードを書いてしまったので少し不安でした。

コードでの実践もそうですが、数式レベル/概念レベルでの解説をもう少し丁寧にしていただけるとありがたいです。

解析上解くとなると中身を理解していなくても実効的には解析出来てしまう(この関数を使う、以上になる)ので、補足資料として数式的なこともあると理解が深まったかと思われます。

⇒こちらのパワー不足で申し訳ないです。数式・概念は論文や検索、書籍でたくさん出てきますが、実装はあまりないので実装を重視。

2. 前回のコメントに回答

9

講義の難易度

- ・ 最後の実践は手がつかなかったが、それ以外は、初心者でもついていけるレベルと量の演習でした。ほかの人には簡単だったのではないかと思います..。

- ・ 内容は難しかったですが、1つずつ理解して進めていくことができました。

- ・ 実践的な内容で、とても興味が持てましたがあまりに難しかったです。

⇒え、俺、どないしたらいい？

宿題が欲しい

- ・家に持ち帰るような演習問題などがあればありがたいです.
- ・毎週の予習内容等あれば, 宿題の様な形で出していただけると学習の定着がよりはかれるかと思いました.

⇒こちらのパワー不足で申し訳ないです。そんなに宿題好きな人間がいるとは・・・

今後の勉強に向けた参考図書や情報を後ほど紹介します。

ALLENや自分の実験への応用について

- ・最後のAllenの解析については、もう少し時間や解説が欲しかったです（時間的に厳しいとは思いますが）
- ・Allenのデータを使うということでテンションが上がったのだが、前処理が複雑に見えてよくわからなかった。
- ・最後の結合強度の計算とか、ちょっと理解できなかった点もありました…

⇒ALLENについては、前半12回終了後に希望者のみで追加勉強会の実施を検討します。

その他

- ・グラフを書く練習はいつごろしたら、この講義についていきやすくなるでしょうか？

⇒とくに練習しなくて大丈夫です。必要なときに調べればOK。プログラミング全般について同じです。

- ・実際の実験データを扱う練習はもっとした方がいいように思っています。自分で応用しようとする、欠損値や外れ値などなどに泣かされるので、それらをどうのりこえるか、は、必要かなと。

⇒実データへの応用的な話を今週も追加します。欠損値や外れ値までは対応できませんが・・・

その他

ラフな感じよかったです！来週もよろしくお願いします

⇒こちらこそ、楽しかったです♪

みなさんが積極的に講義に参加していただけ、やりがいがあります。今週も楽しく、かつ学び多い時間になるように心がけます！

本日のゴール

機械学習ライブラリscikit-learnを使用して、回帰、分類、クラスタリング、次元圧縮を実装できるようになる

3. 本日の講義概要

15

4.1 オブジェクト指向

休憩

4.2 PDBによるデバッグ

4.6 scikit-learn
クラスタリング kMeans

4.3 scikit-learn回帰

4.7 scikit-learn
次元圧縮 PCA

4.4 scikit-learn全般

4. 演習
fMRIデータから分類

4.5 scikit-learn分類SVM

4.1 オブジェクト指向とは

16

Pythonのオブジェクト指向プログラミングを学びます

練習クイズ

線形回帰を実行するクラス、Original_LS_regressionを作成せよ。クラス変数（メンバ変数）は、theta_LSとする。

メソッドは、コンストラクタ、fit(X,t)、

y = predict(plt_X)の3つとせよ。

このクイズが解けるように、オブジェクト指向によるクラスとオブジェクトの作成を学習します

#オブジェクト指向とは

Thanks by wikipedia オブジェクト指向

<https://ja.wikipedia.org/wiki/%E3%82%AA%E3%83%96%E3%82%B8%E3%82%A7%E3%82%AF%E3%83%88%E6%8C%87%E5%90%91>

オブジェクト指向（オブジェクトしこう）とは、オブジェクト同士の相互作用として、システムの振る舞いをとらえる考え方である。英語の object-oriented (直訳は、「対象物志向の」「目的重視の」という意味の形容詞) の日本語訳である。

オブジェクト指向の枠組みが持つ道具立ては、一般的で強力な記述能力を持つ。複雑なシステム記述、巨大なライブラリ（特に部品間で緊密で複雑な相互関係を持つもの）の記述においては、オブジェクト指向の考え方は必須である。

はい、意味分かりません。

私なりのオブジェクト指向の説明（人それぞれ・・・）

1. 世界のはじまり、変数 xだけが存在。しかし

「まとめて変数が使えないの面倒だ・・・」

2. 変数をまとめた配列Xが誕生。しかし

「いろんな型をまとめて扱えないの面倒だ・・・」

3. 様々な型変数、配列をまとめた概念、構造体が誕生。

ここまでがC言語の世界。しかし

「変数と関数（メソッド）が一緒になっていないのは、面倒だ・・・」

⇒例を紹介：C言語で、Animalという構造体を用意し、構造体dogの、鳴き声を出力する関数を実現してみる

4.1 オブジェクト指向とは

19

```
# https://ideone.com/aifT7j
include <stdio.h>
typedef struct{
    char name[256]; //動物の名前
    char cry[256]; //鳴き声
}Animal;

void sound(Animal animal){
    printf("鳴き声:%s", animal.cry);
};

int main(void){
    Animal dog = {"犬", "ワン"};
    sound (dog);
}
```

→出力：鳴き声：ワン、だが、犬の構造体変数と、使いたい関数が、一緒になってなくて面倒だ・・・

#よし！変数と関数（メソッド）も、ひとつにまとめよう
⇒ クラスという概念、オブジェクト指向が誕生

クラス

動物のプロパティ（＝変数：例えば名前、鳴き声）と動作（関数：例えば鳴く）が、一体化した概念。

クラスを使えば現実世界の物体をひとつのもの（＝オブジェクト）として扱えて便利。

※その他オブジェクト指向では、継承、カプセル化、ポリモーフィズム、抽象クラス、インターフェース、セッター、ゲッターなどなど、保守性と再利用性を高める仕組みがありますが、今回はそこまでは説明しません。

→ Pythonノートブックへ

Pythonのデバツグライブラリpdbの使い方を学びます

#次のコードは前回線形回帰をするときに使ったコードですが、一部間違った箇所があり実行できません。

エラー文を読みつつpdb.set_traceを利用して、デバツグしてみましょう。

```
import numpy as np  
rnd = np.random.RandomState(1701)  
...
```

このクイズが解けるように、pdbの操作を学習します

→Pythonノートブックへ

PythonのIDE (Integrated Development Environment)

- PyCharm (無料版と有料版があるが、無料版で十分)
- Visual Studio
- Spyder
- JupyterLab など

本当にきちんとしたデバッグが必要なレベルでは、PyCharm無料版が個人的にはおすすめです。

私の場合、深層強化学習の実装時のようなきちんとしたデバッグをやりたいときはPyCharmを、簡単なWebアプリ作成の場合はAtomエディターを使用しています。

Pythonの機械学習ライブラリscikit-learnの使い方の流れを学び、回帰を実行します

練習クイズ

以下のデータに対して、scikit-learnの線形関数の最小二乗法を用いて、傾きと切片を求めよ。

```
x = np.linspace(-3, 3, N_TRAIN)
```

```
t = 5.5 * x + 1.2 + 2.0 * rnd.randn(N_TRAIN)
```

このクイズが解けるように、scikit-learnの操作を学習します

scikit-learn

Pythonの機械学習ライブラリです

<http://scikit-learn.org/stable/>

どんな機械学習があるかは、後ほど紹介するとして、
まずは回帰のみをやってみましょう。

自分たちで作った最小二乗法のクラスをscikit-learnのクラスで置き換えてみます。

→Pythonノートブックへ

機械学習の分類と例

教師あり学習

(ラベルデータを出力したい)

分類

- ・スパムメールをはじく
- ・手書き文字を認識

推薦

- ・おすすめ商品を提示
- ・自分の友人かもしれない人を表示

(数値データを出力したい)

回帰

- ・新商品の売上を予測
- ・株価を予測

教師なし学習

(データにラベルをつけたい)

グルーピング

- ・顧客をタイプ別にグループ分け
- ・様々な文章をカテゴリー分け

異常値検知

- ・不正なwebアクセスの検出
- ・不正なクレジットカード使用の検出

(データを解析しやすくしたい)

次元圧縮

- ・高次元データを2次元にして可視化
- ・データの次元を削減し、機械学習の解析性能を向上させる

強化学習

機器制御

- ・ロボットの歩行制御
- ・空調の最適化

対戦戦略構築

- ・囲碁
- ・ポーカー

合成

生成

- ・文章や画像の生成

変換

- ・画風の変換

これから、回帰、分類、クラスタリング、次元圧縮について説明します。

説明後、チームでできる限り、各手法が神経科学の分野でどのように使えそうか、アイデアをSlackにどんどん書き込んでください。間違っていてかまいません。数の勝負です。

例

回帰：実験結果から直線で近似して図を描く

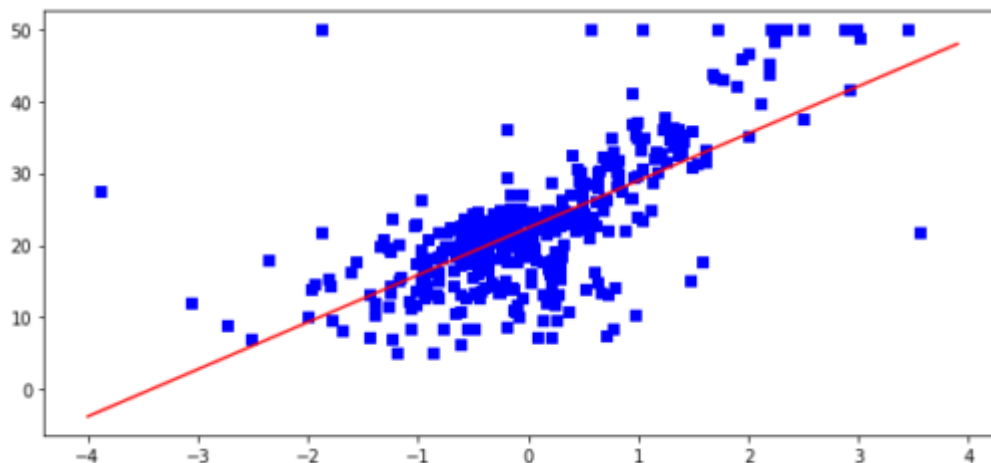
分類：脳波からP300とそうでないのを分ける

etc . . .

教師あり学習

回帰 (Regression)

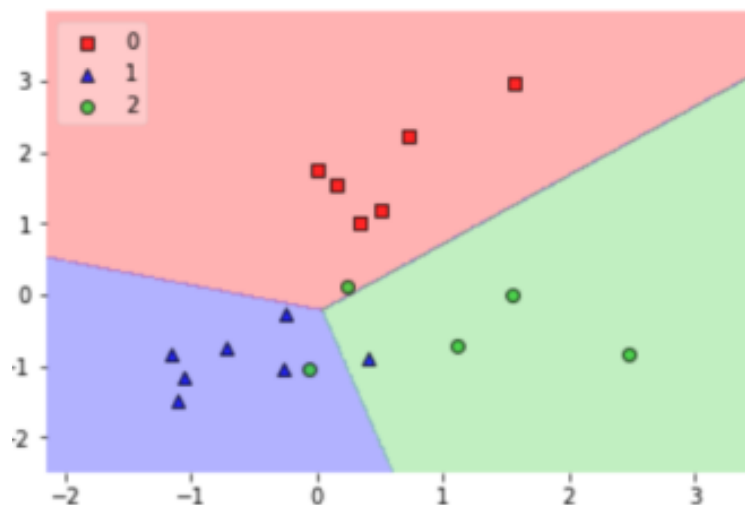
計測した入出力データから、データの関係性を再現する関数を推定し、未知の入力に対する出力を予測する



教師あり学習

分類 (Classification)

データとラベルの対からなる計測データに対して、識別面を作成し、未知のデータのラベルを推定する

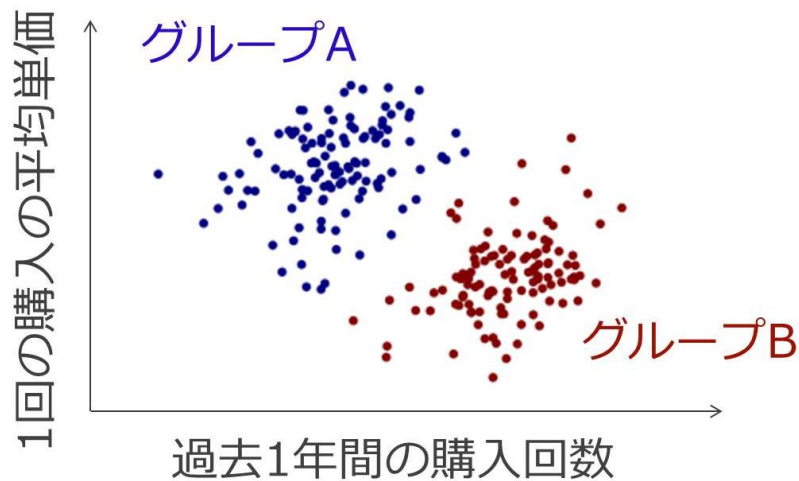
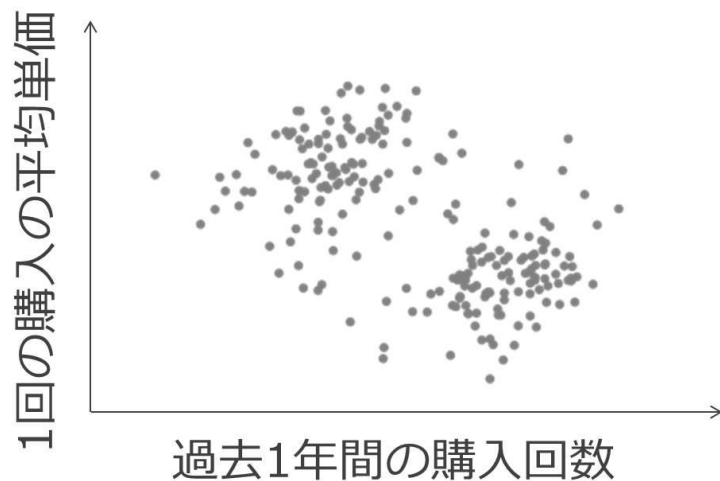


教師なし学習

クラスタリング (Clustering)

データを似ているグループごとにまとめる

(= データにラベル付けを行う)



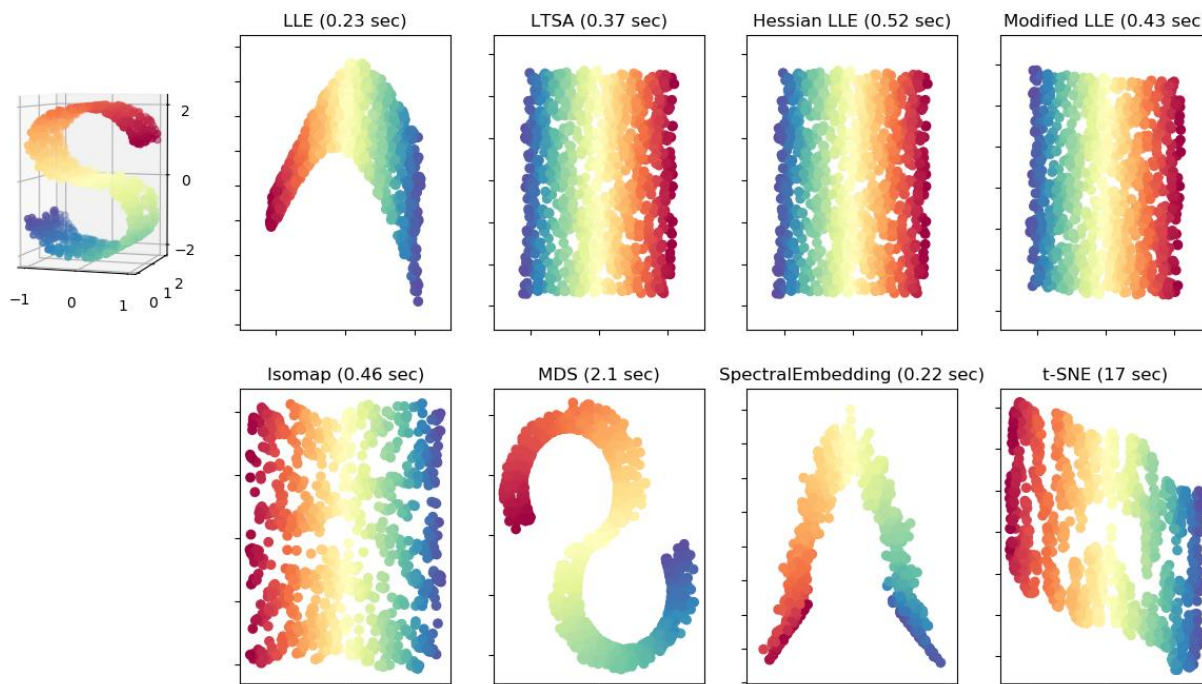
教師なし学習

次元圧縮 (Dimensionality Reduction)

多次元データの次元を縮約する

http://scikit-learn.org/stable/_images/sphx_glr_plot_compare_methods_001.png

Manifold Learning with 1000 points, 10 neighbors



#いまから10分間で、各機械学習手法が神経科学の分野でどのように使えそうか、チーム内で話し合い、アイデアをSlackにどんどん書き込んでください。間違っていてかまいません。数の勝負です。目標はクラスで30個です。

(入力例)

回帰：実験結果から直線で近似して図を描く

分類：Brain Computer Interfaceで脳波からP300を識別

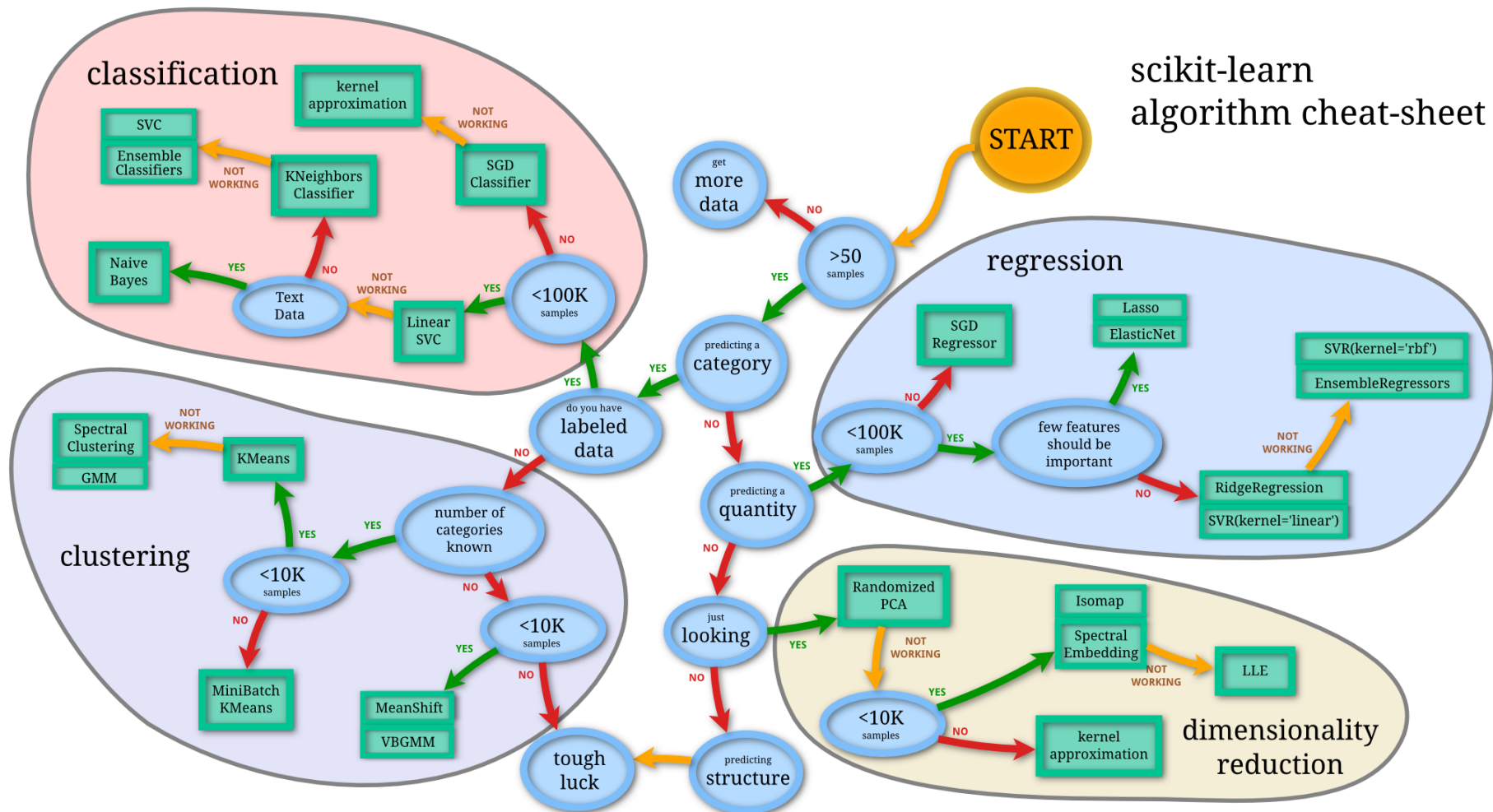
などなど・・・

#たぶんそろそろ、みんな疲れる（とくに私が・・・）

- connecting the dots
- Joseph Schumpeter
- Facebook

アルゴリズムマップ

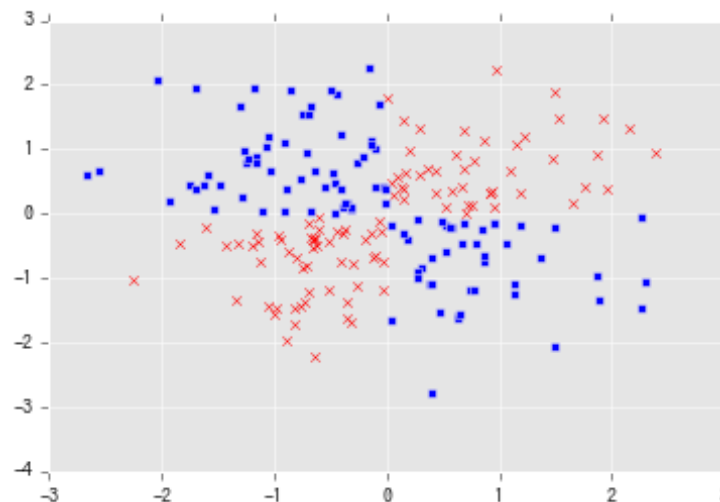
http://scikit-learn.org/stable/tutorial/machine_learning_map/



SVMを用いた教師あり学習による識別を学習します

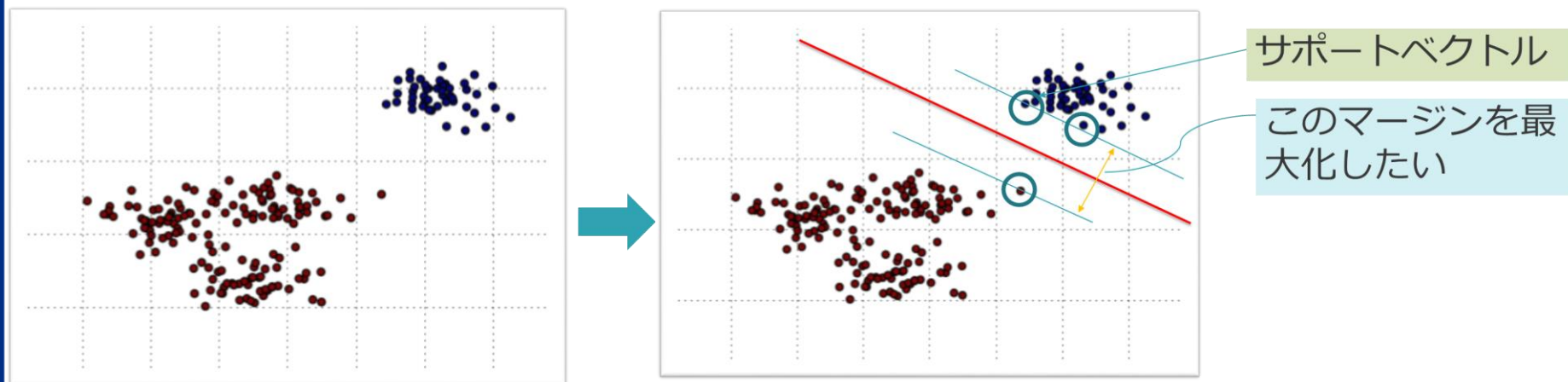
#練習クイズ

以下のデータに対して、scikit-learnのSVC（support vector machine classification）を用いて識別平面を作成し、任意のデータに対して識別せよ。



→Pythonノートブックへ

SVMを用いた教師あり学習による識別を学習します



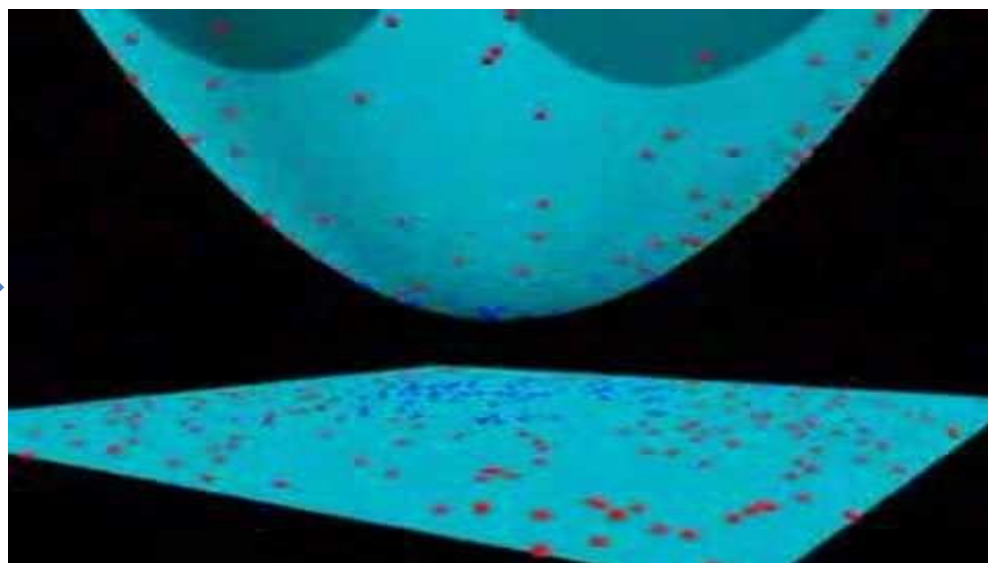
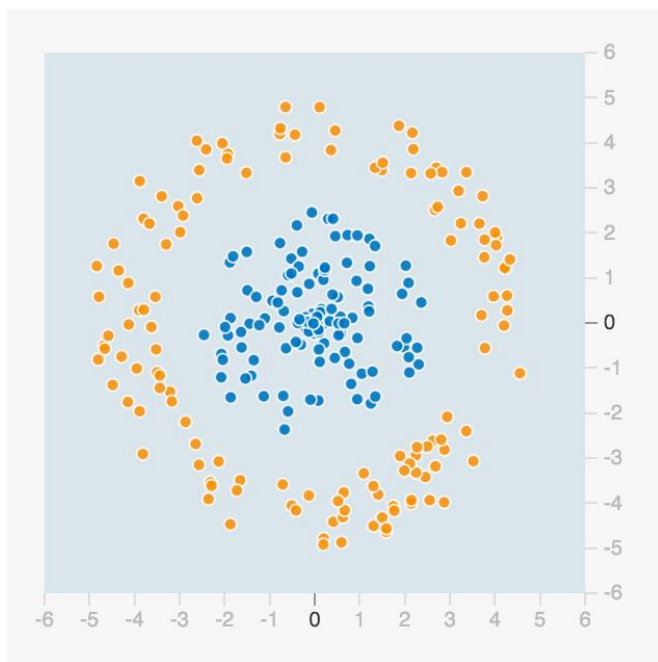
どうやって計算して赤線は決まるの？

→ラグランジェの未定乗数法を活用

or 損失関数をhinge関数にして最尤推定 詳細は省略・・・

非線形の場合 = 直線で分離できない場合。

カーネルSVMを使用する。



<https://www.youtube.com/watch?v=3liCbRZPrZA&feature=youtu.be>

→Python notebookへ 練習クイズ

多クラスの場合の識別

- One vs All 方式

それぞれのクラスに対し、「あるクラス対その他全てのクラス」という分類を行う。

→境界面が複雑になり、精度が悪い。計算量は小さい。

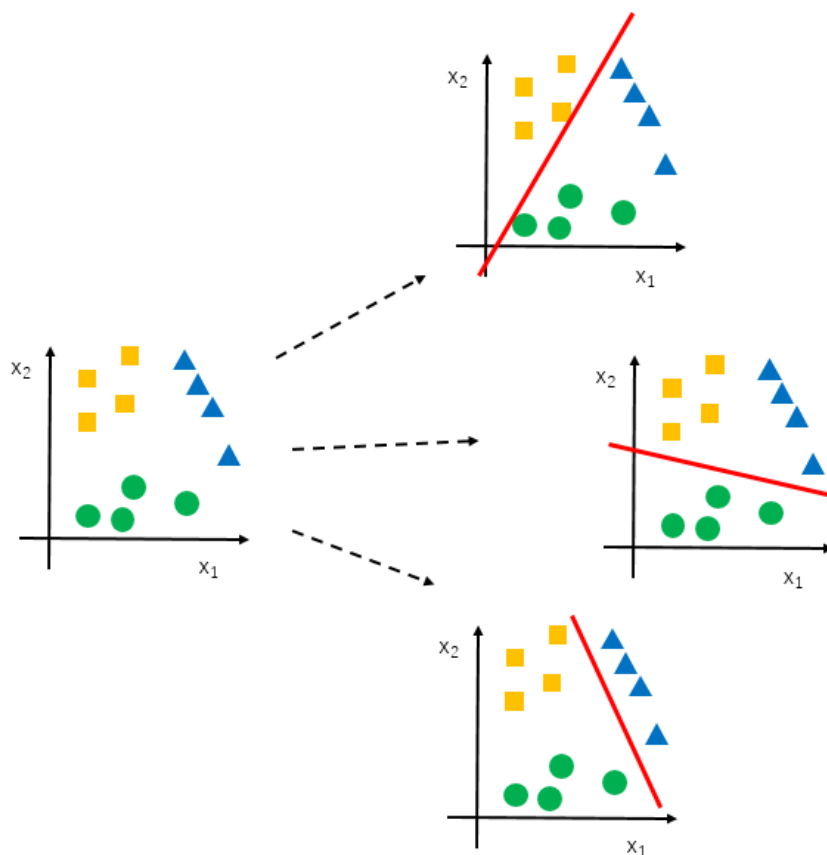
- One vs One 方式

$k(k-1)/2$ 通りのクラス分けの組み合わせに対して分類を行う。

→計算量が多い。その代わり線形な境界面が単純で複雑なデータに強い。

クイズ

下の絵はどっち？ 🐶 One vs All 🐱 One vs One
チームで相談OKです。

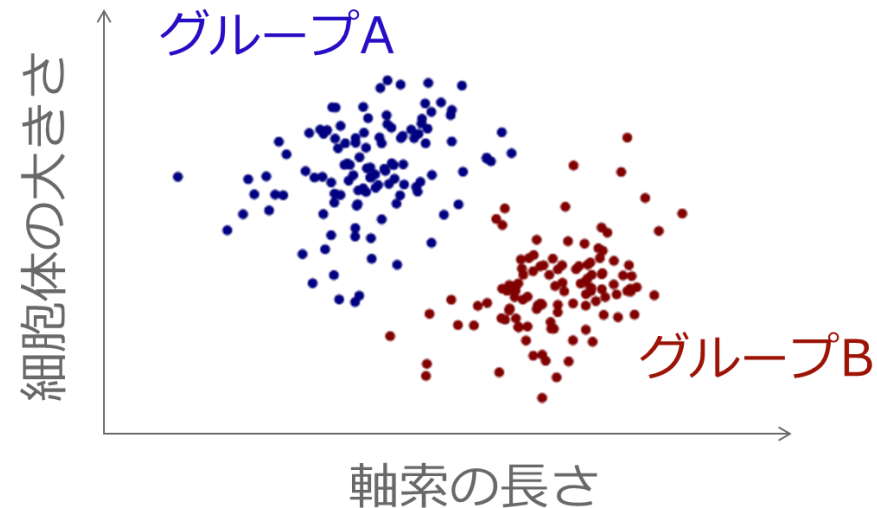
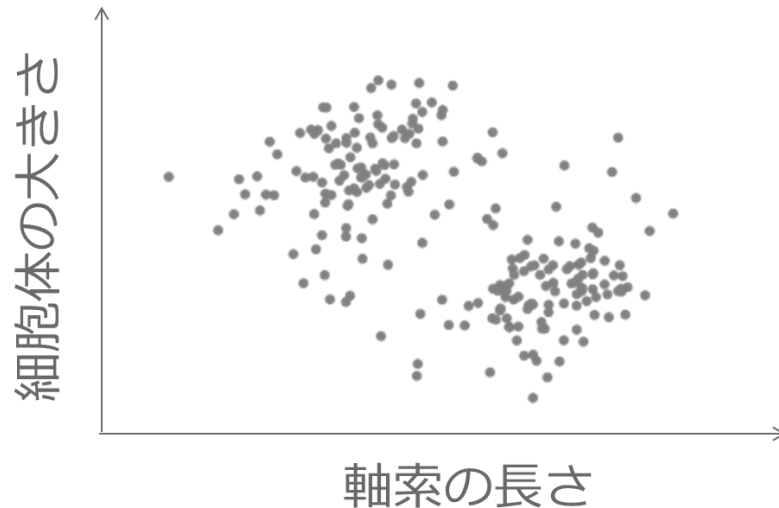


4.6 scikit-learn クラスタリング 39

kMeansを用いたクラスタリングを学習します

学習

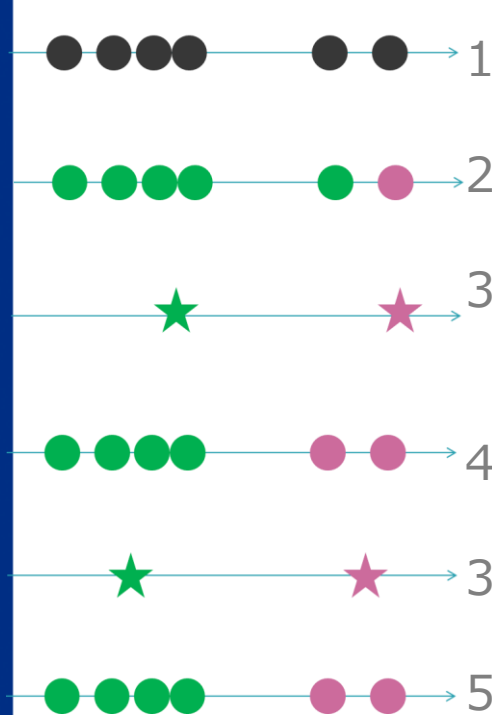
以下のデータに対して、scikit-learnのKMeansを用いて、2つのクラスターにデータを分類します



→次ページへ

4.6 scikit-learn クラスタリング 40

kMeansのアルゴリズム



1. 分離したいクラス数(k)を決める $k=2$
2. 初期化：データ集合をランダムに k 個のクラスタ分割し，初期クラスタを得る
3. 各クラスタについて重心 $\frac{1}{N_i} \sum_{x_j \in X_i} x_j$ を計算
4. 全てのデータ x を，各クラスタの重心との距離 $\|x - x_i\|$ が最小のクラスタへ割り当てる
5. 前の反復とクラスタに変化がないか反復数が上限を超えたら終了．そうでなければ，ステップ3に戻る．

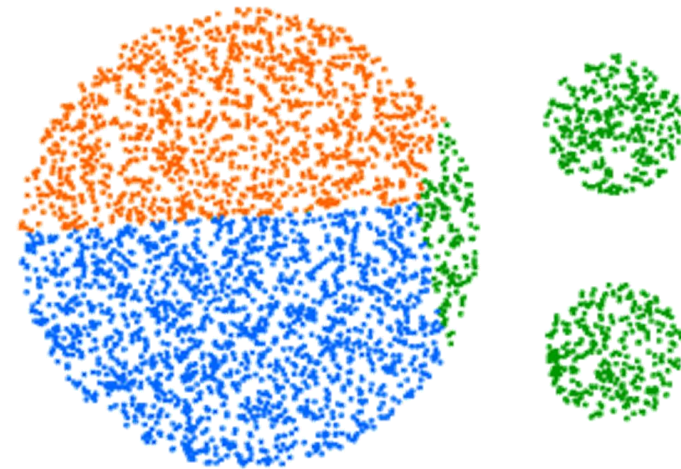
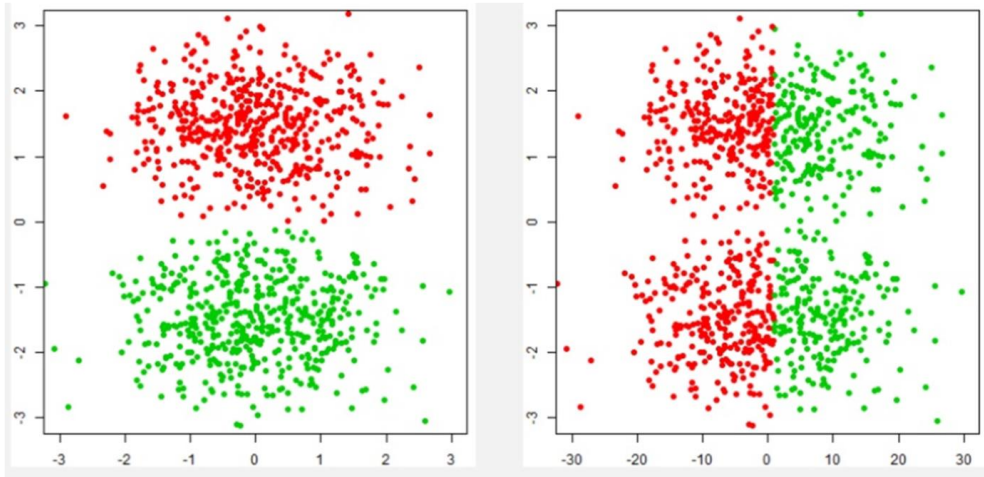
→Pythonノートブックへ

4.6 scikit-learn クラスタリング 41

kMeansの注意点（距離を測るので正規化必須）

Xだけ値が10倍

クラスタの大きさが異なる



クイズ：アルゴリズムから考えて、kMeansでうまくクラスタリングできなさそうなケースを、チームで考えSlackにどんどん書き込んでください（5分間）。

4.6 scikit-learn クラスタリング 42

その他のクラスタリング手法

- クラスタが楕円形だったら？

⇒ Gaussian Mixture Models (GMM)

- クラスタの形が非線形だったら（多様体の上）

⇒ Spectral Clustering

クラスタの数が適正だったかはどう判断するの？

⇒ エルボー法、シルエット分析など

クラスタ数が未知の場合は？

⇒ Variational Bayesian Gaussian Mixture Models (VBGMM)

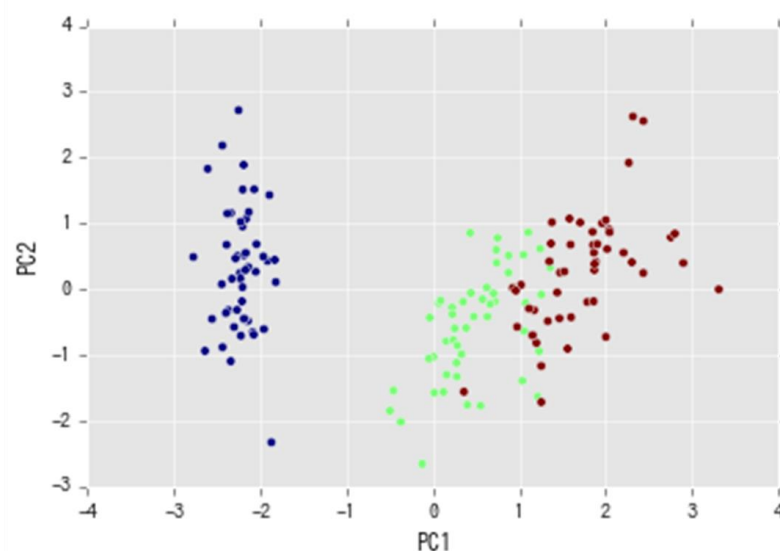
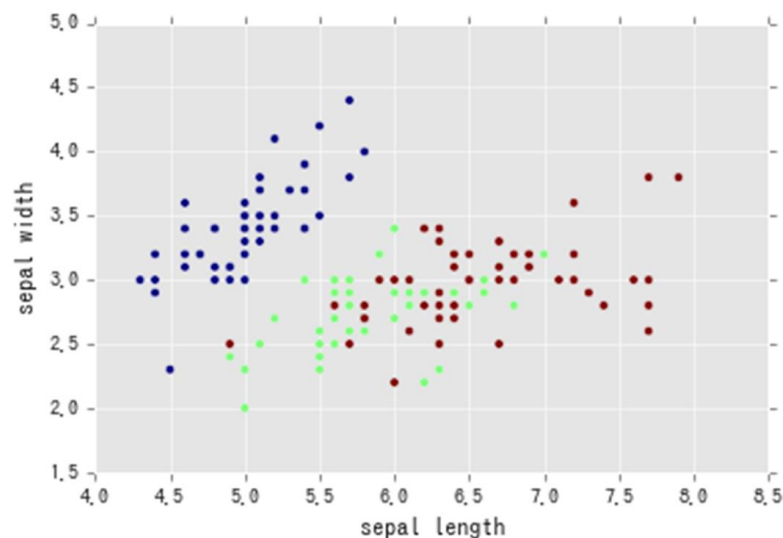
4.7 scikit-learn 次元圧縮

43

#主成分分析PCAを用いた次元圧縮（次元削減）を学習します

学習

Irisの4次元データに対して、scikit-learnのPCAを用いて、2次元にデータを圧縮せよ。



→次ページへ（なぜ次元圧縮が必要なのか）

次元圧縮が必要な場合

- ・ アンケート調査などで因子分析をしたい
- ・ 多次元データの特徴を可視化したい
- ・ 脳波などチャネルごとに重複した情報を持つデータを独立に分割（ICA）

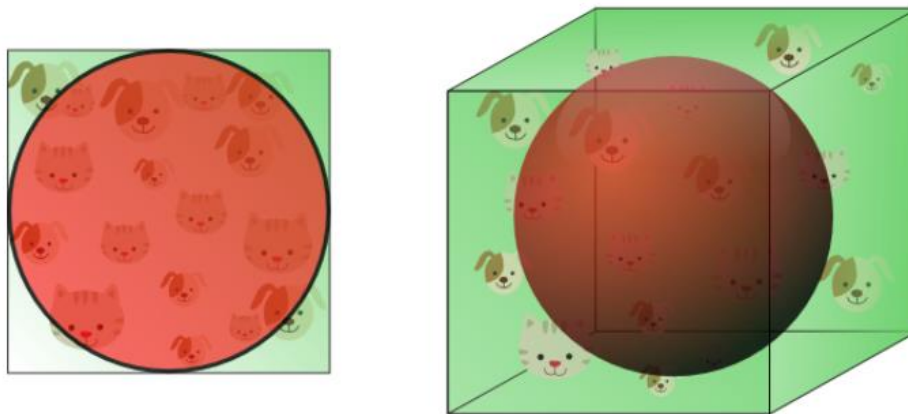
http://scikit-learn.org/dev/tutorial/statistical_inference/unsupervised_learning.html

- ・ 機械学習する際に、集めた多次元データから無駄に多い次元を減らす

次元の呪い

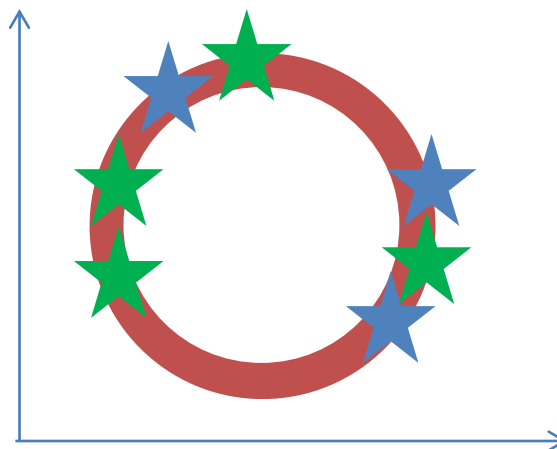
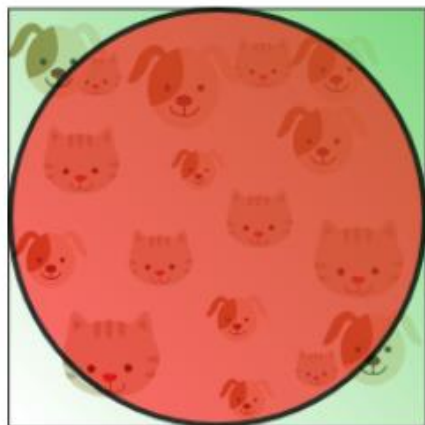
- ・ 学習に必要なデータ数が増える
- ・ 高次元空間では球面集中化現象によりデータ間の距離が均一になりやすくなる（赤色は半径1の領域、次元が増えると減る）。空間内のはじっこの領域（緑色）が次元が増えるに従い増えていく。

<http://www.visiondummv.com/2014/04/curse-dimensionality-affect-classification/>



次元の呪い

⇒ 2次元の場合、半径成分が無視され、角度の1次元でデータを示せる。



次元が無駄 = ランク落ちる = データが正則でなく計算破綻

⇒ 計算破綻を防ぐには、前回の正則化や次元圧縮をしよう

PCA (Principal Component Analysis、主成分分析)



PCAを実装した事がある



PCAを使った事がある

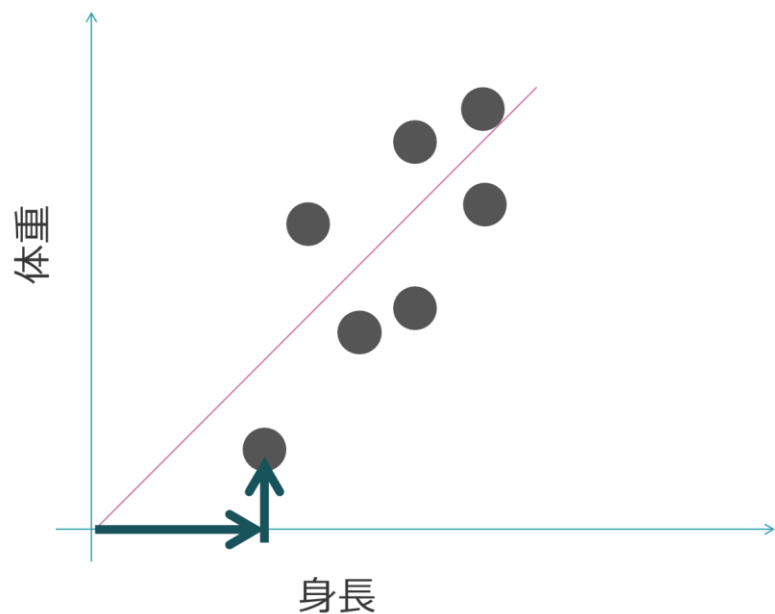


PCAを聞いたことがある

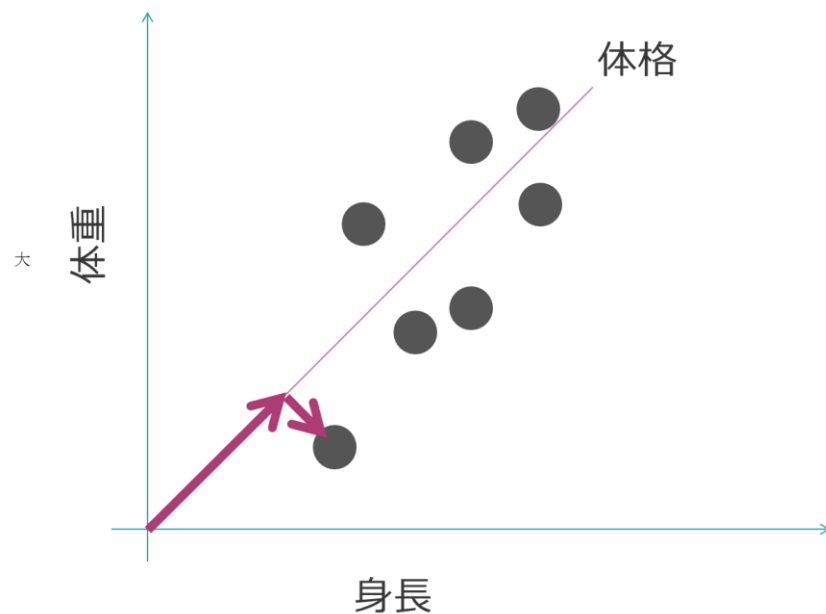


PCAは聞いたことがない

PCA



身長と体重は相関する



無相関にし、体格と太り具合
という新たな軸を作りたい

PCA

正規化されたデータ行列Dに対して、分散共分散行列は以下

$$\Sigma = D^T D = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{33} & \sigma_{33} \end{bmatrix}$$

対角成分=分散はデータのバラつき具合であり、共分散は相関に比例する。

データの次元を圧縮するには、

- ① 独立な軸を作りたい = 共分散を0にしたい
- ② データのバラつきの大きい順に並べたい = 固有値順

PCA

つまりDを変形したD2において、下を示す式になると嬉しい

$$\Sigma_2 = D_2^T D_2 = \begin{bmatrix} \sigma_{11} & 0 & 0 \\ 0 & \sigma_{22} & 0 \\ 0 & 0 & \Delta \end{bmatrix}$$

Σ_2 は結局($D^T \cdot D$)を固有値展開して求めた固有ベクトルWを使って、Dを変換した場合の固有値と同じ。

つまり、主成分分析 = 固有値分解

ただし、データの次元を圧縮するために新しいデータ軸一部の固有ベクトルのみを使用する → Pythonノートブックへ

本日の実践演習は

- ・ fMRIのBOLD信号から被験者が何を見ていたのかを機械学習で予測してもらいます。

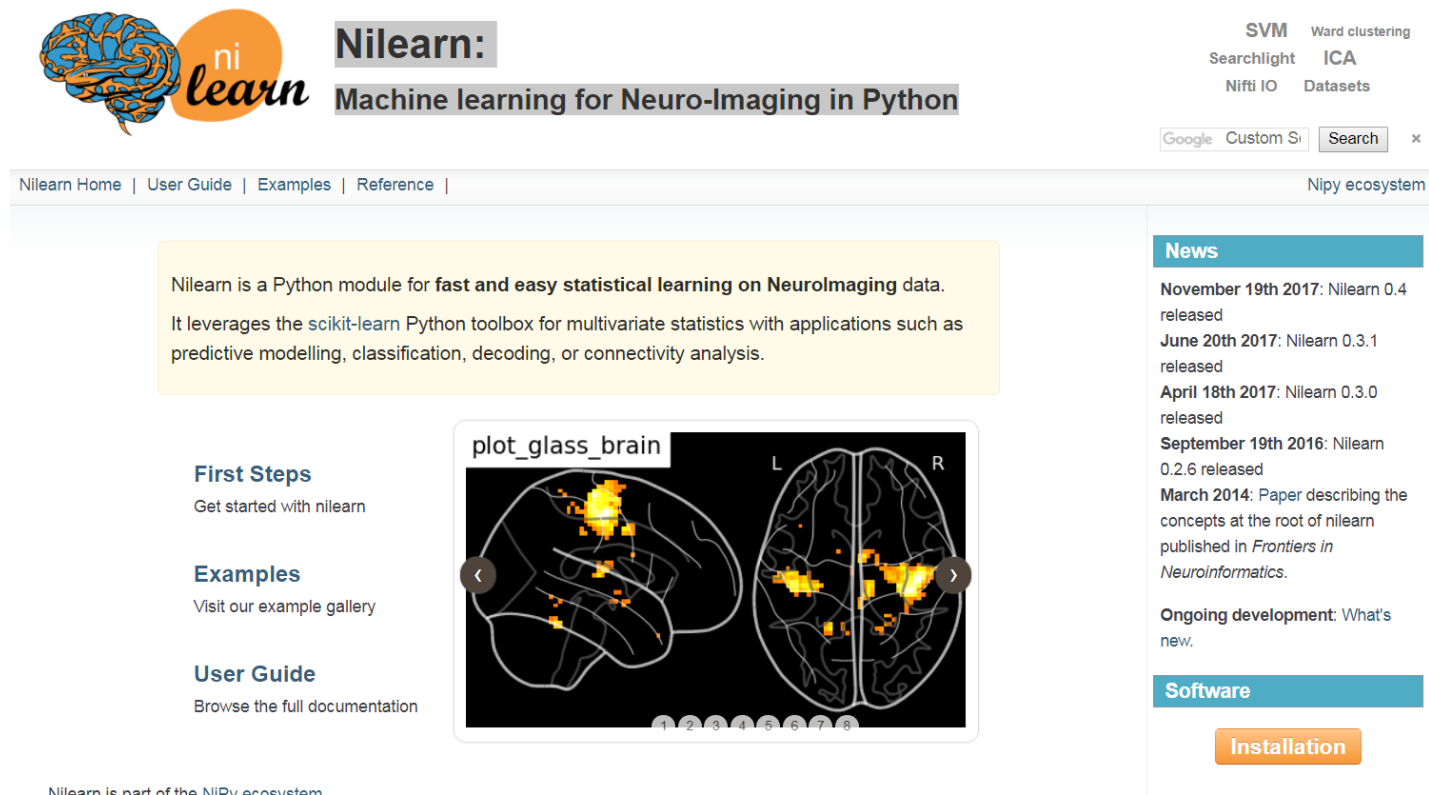
はじめに学習データとバリデーションデータ用のデータをお渡しします。

最後にテストデータを渡すので、テストデータでの結果が一番だったチームが優勝です。

Nilearn: Machine learning for Neuro-Imaging in Python

<http://nilearn.github.io/index.html>

fMRIの機械学習セット。神谷先生のデコーディングなどもあって楽しい



Nilearn:
Machine learning for Neuro-Imaging in Python

Nilearn is a Python module for **fast and easy statistical learning on NeuroImaging data**.
It leverages the **scikit-learn** Python toolbox for multivariate statistics with applications such as predictive modelling, classification, decoding, or connectivity analysis.

First Steps
Get started with nilearn

Examples
Visit our example gallery

User Guide
Browse the full documentation

plot_glass_brain

News

- November 19th 2017: Nilearn 0.4 released
- June 20th 2017: Nilearn 0.3.1 released
- April 18th 2017: Nilearn 0.3.0 released
- September 19th 2016: Nilearn 0.2.6 released
- March 2014: Paper describing the concepts at the root of nilearn published in *Frontiers in Neuroinformatics*.

Ongoing development: What's new.

Software

Installation

Nilearn is part of the NiPy ecosystem

Distributed and overlapping representations of faces and objects in ventral temporal cortex.

Haxby, James V., et al. Science 293.5539 (2001): 2425-2430.

http://www.cogsci.bme.hu/~ktkuser/PHD_iskola/docs/kgy/haxbycatscience01.pdf

顔や家や猫の絵見せたときのfMRIから絵のカテゴリー推定

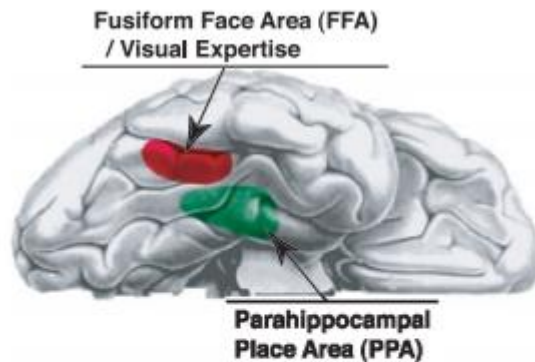


Fig. 1. Schematic diagram illustrating the locations of the fusiform face area (FFA), which also has been implicated in expert visual recognition, and the parahippocampal place area (PPA) on the ventral surface of the right temporal lobe. In most brains, these areas are bilateral.

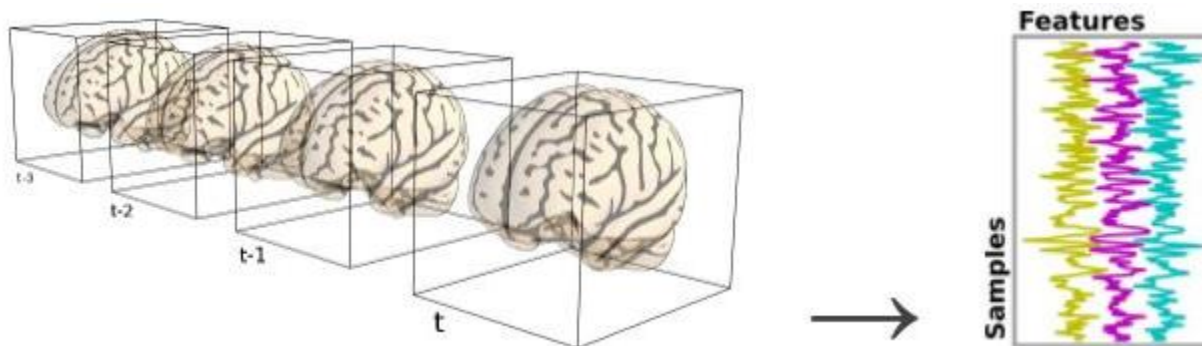


今回はHaxbyの家とネコの写真を見せたときのVentral Temporal Cortexの活動をfMRIで測定したBOLD信号から、家とネコのどちらを見ていたのかを分類してもらいます。正答率でコンテストを行います。優勝チームには後日、豪華？プレゼント用意します。

Colaboratoryにはnilearnは入っていないので、ライブラリをインストールして追加する必要があります。

http://nilearn.github.io/auto_examples/plot_decoding_tutorial.html#sphx-glr-auto-examples-plot-decoding-tutorial-py

私がデータまで落としたものを使用してもらいます。



http://nilearn.github.io/building_blocks/manual_pipeline.html#masking

1. 【機械学習初心者向け】scikit-learn「アルゴリズム・チートシート」の全手法を実装・解説してみた

<https://qiita.com/sugulu/items/e3fc39f2e552f2355209>

を読んで興味のあるアルゴリズムを2つ選び、リンク先に掲載されている例コードを実装してみてください。

2. 自分の研究で何か機械学習が活かせないかアイデアを10個挙げて下さい

3. 以下の本を借りるなり買うなりして、どんな本か把握しておいてください

【書籍】scikit-learn関係

- ・ Python機械学習プログラミング 達人データサイエンティストによる理論と実践
- ・ Pythonによるスクレイピング&機械学習

4. 余裕がある人は以下の情報をチェイスする習慣をつけてください

【Webサイト】

- ・ AINOW <http://ainow.ai/>
- ・ はてなブックマークのテクノロジー <http://b.hatena.ne.jp/ctop/it>
- ・ Qiita <https://qiita.com/> 登録しておけば、その週の人気記事が届く

(続き) 4. 余裕がある人は以下の情報をチェイスすると良いかも

【勉強会】

- techplay <https://techplay.jp/> 登録しておけば、興味のある勉強会情報が届く
- connpass <https://connpass.com/>

【Slack】

- モヒカン <https://qiita.com/kotakanbe@github/items/32cf4eb3de1741af26fb>
<https://mohikan-slackin.herokuapp.com/>

blog-ja, # deep_learning, # machine-learning, # slide_technology_ja あたりのチャンネルをたまに見る

【twitter】

- 人工知能,機械学習関係ニュース研究所@AI_m_lab
- 人工知能・機械学習ニュース@A_I_News

【メルマガ】

- Weekly Machine Learning <https://www.getrevue.co/profile/icoxfog417/> 日本語メルマガ
- Deep Learning Weekly <http://digest.deeplearningweekly.com/> 英語メルマガ

【雑誌】

- 日経ソフトウェア <https://eb.store.nikkei.com/asp/ShowSeriesDetail.do?seriesId=D2-00SW0000B>
- Software Design <http://gihyo.jp/magazine/SD>
- Interface <http://interface.cqpub.co.jp/>

あたりを毎月チェックして、興味のある号は買う。Software Designの4月号買って下さい。