情報管理

第６回：クラスタリング
教師ありクラスタリング

今回の講義内容

今回からはデータを複数のクラスに分類する「クラスタリング」について解説していきます。

今回は教師ありクラスタリングの方法について解説します。

また，学習のためのパラメータ最適化法として「勾配降下法」についても解説します。

- 線形判別分析
- 勾配降下法
- ロジスティック回帰

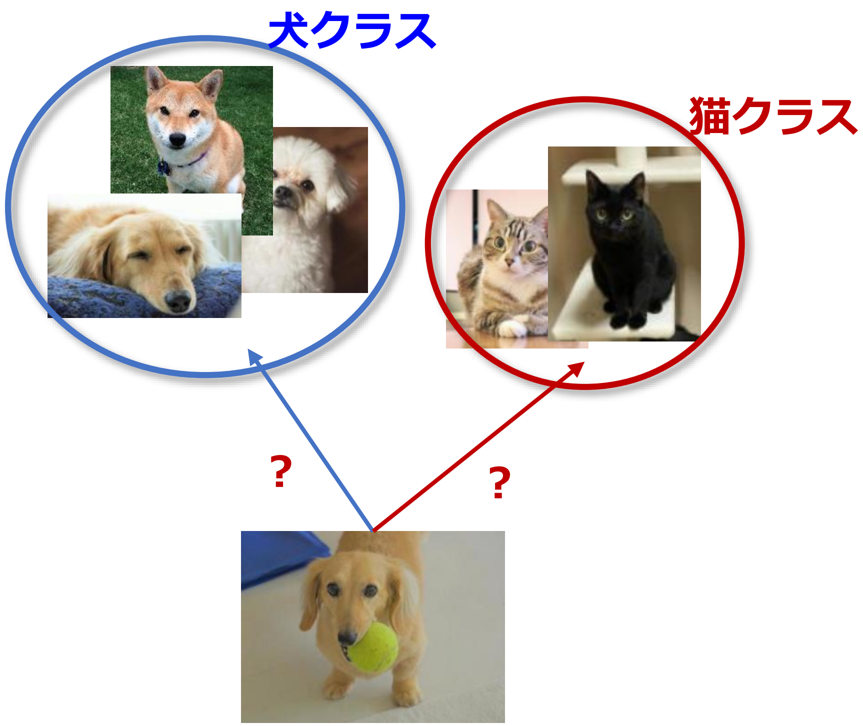
ロジスティック回帰を理解しておくと，以降で説明するニューラルネットワークも理解しやすくなります。

クラスタリングとは？

一言でいうと，データを分類することです。

右の例では，入力された画像が
犬のクラスなのか，猫のクラスなのかを
分類（クラスタリング）しています。

これは言い換えれば
犬の画像と猫の画像の境界線はどこか？
を求める問題と言えます。

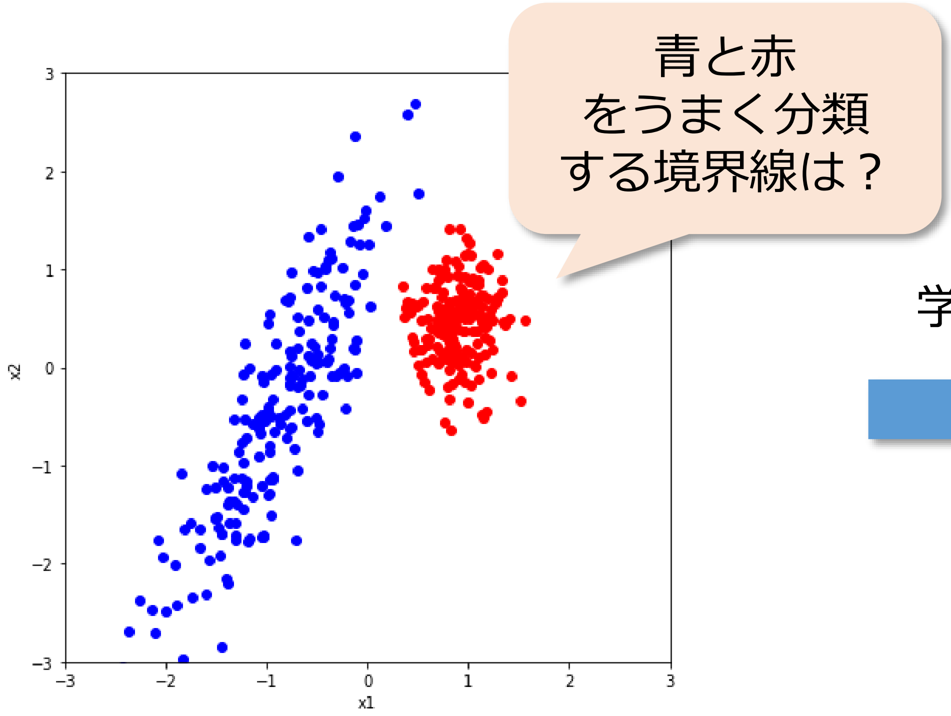


教師ありクラスタリング

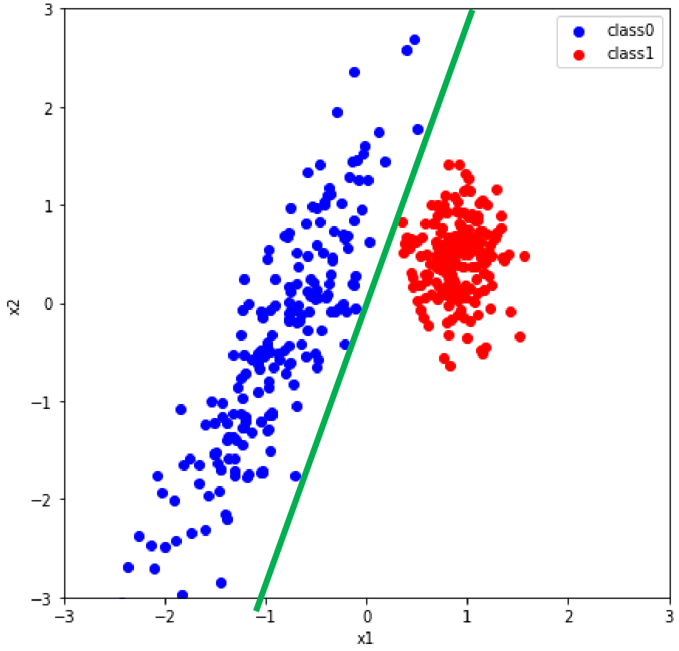
教師ありクラスタリングの目的

教師とは？　→　学習データのクラスラベルのことです。

教師あり学習では，学習データをラベル通りにクラス分類する境界線を学習します。



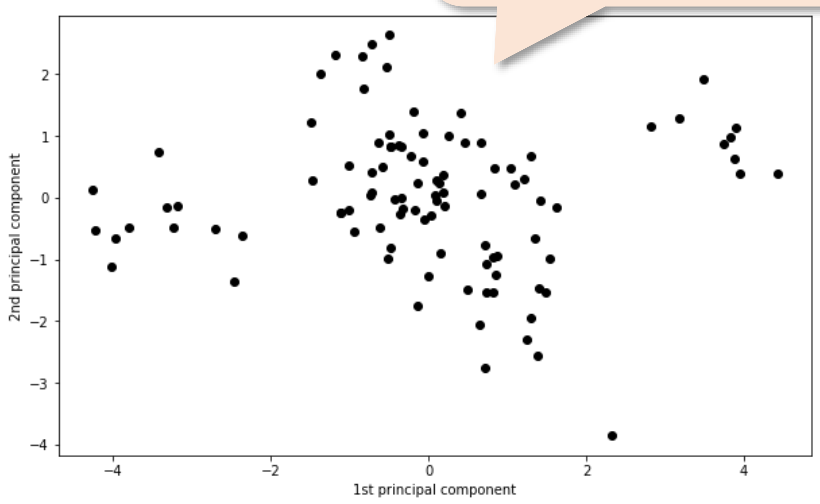
学習



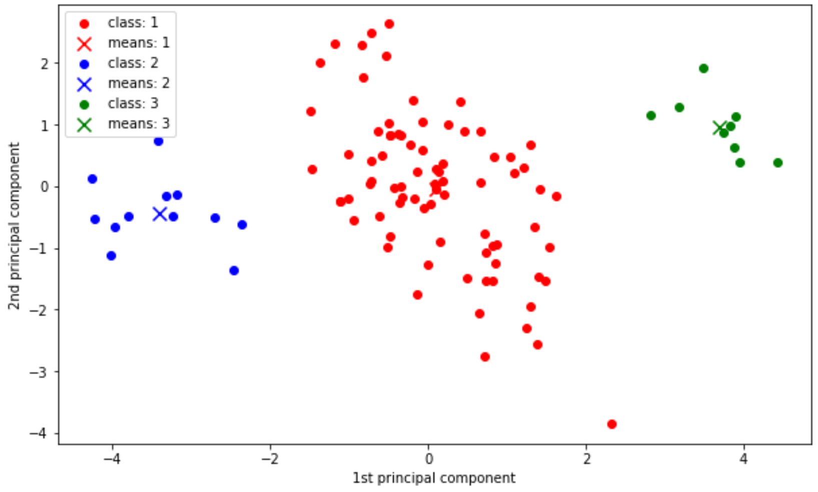
補足：教師なしクラスタリング

教師なしクラスタリングでは，クラスラベルを使用せずにデータを自動的にクラス分類します。

データを3クラスに分類すると？



分類

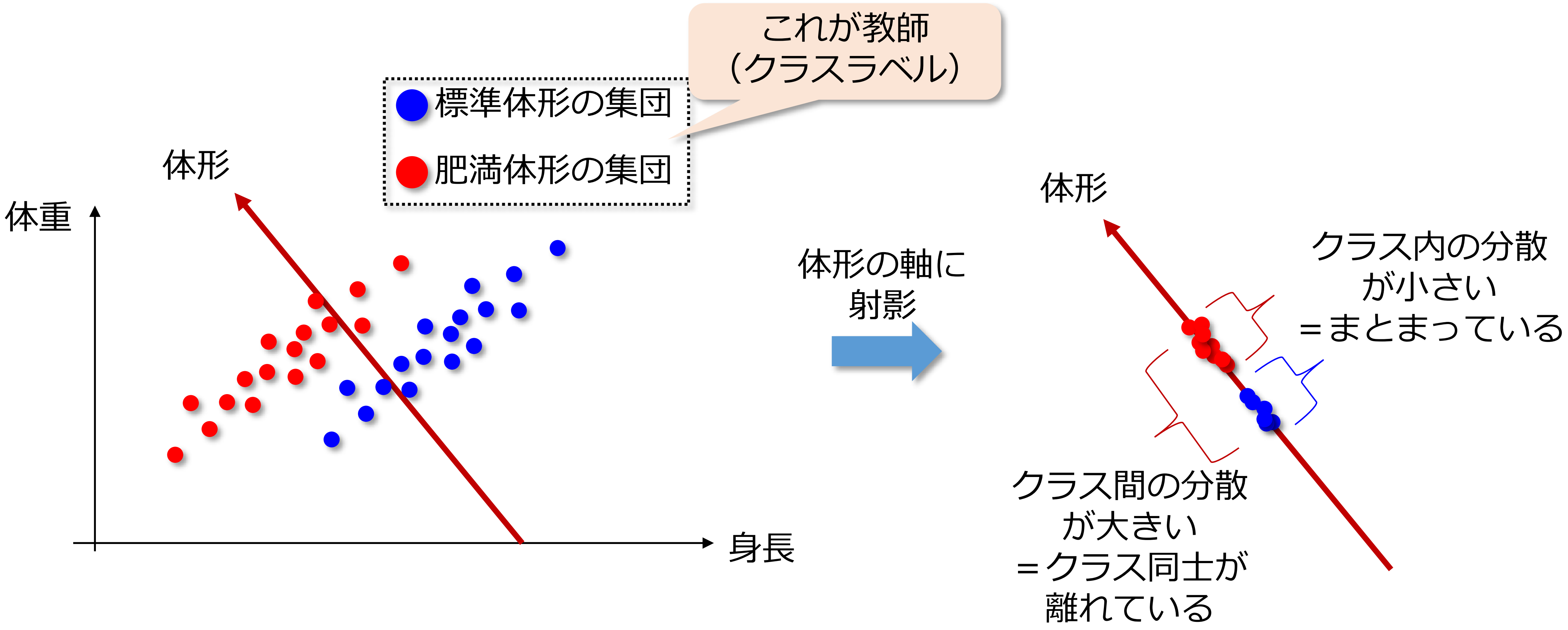


教師なしクラスタリングは次回に解説します。

線形判別分析による
クラスタリング

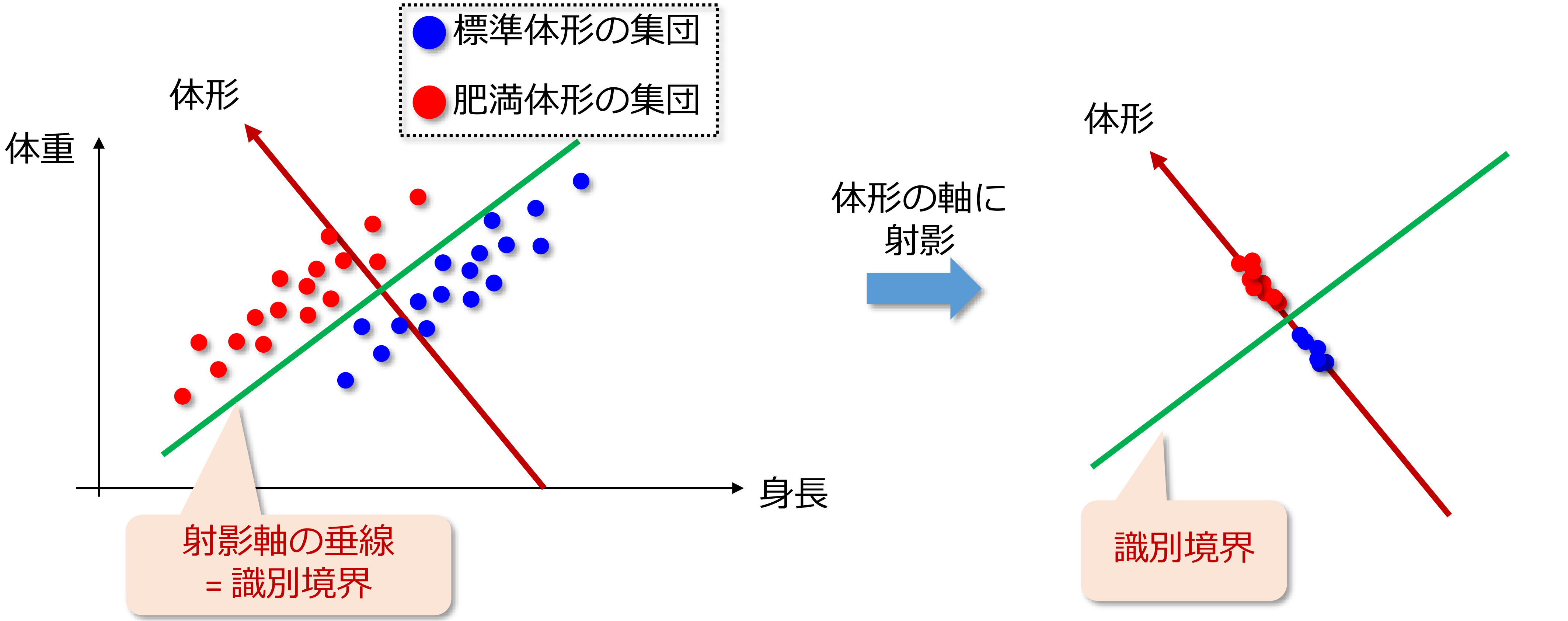
線形判別分析：おさらい

線形判別分析は，クラス内の分散を小さく，クラス間の分散を大きくするように次元を削減する手法でした。



線形判別分析による識別境界

線形判別分析によって1次元に圧縮するとき，射影軸の垂直線が識別境界として見なすことができます。



おさらい：クラス内分散共分散とクラス間分散共分散

クラス内分散共分散：小さいほど良い

$$S_{inner} = \underbrace{\sum_{c=1}^{C} \sum_{n=1}^{N^c} (\boldsymbol{x}_n^c - \overline{\boldsymbol{x}}^c)(\boldsymbol{x}_n^c - \overline{\boldsymbol{x}}^c)^T}_{}$$

クラスcに属するデータの
分散共分散行列

$\boldsymbol{x}_n^c$:クラスcに属するデータのnサンプル目
$\overline{\boldsymbol{x}}^c$: クラスcに属するデータの平均値ベクトル
C: クラスの数
N^c: クラスcに属するデータのサンプル数

クラス間分散共分散：大きいほど良い

$$S_{intra} = \sum_{c=1}^{C} \underbrace{N^c (\overline{\boldsymbol{x}}^c - \overline{\boldsymbol{x}})(\overline{\boldsymbol{x}}^c - \overline{\boldsymbol{x}})^T}_{}$$

$\overline{\boldsymbol{x}}$: データ全体の平均値ベクトル

各クラスの平均値ベクトル$\overline{\boldsymbol{x}}^c$を
各データ$\boldsymbol{x}_n^c$の代わりに
用いた分散共分散

最終的に，クラス間分散/クラス内分散が大きくなれば良い。

$$J = (S_{inner})^{-1} S_{intra}$$

線形判別分析による識別境界の求め方

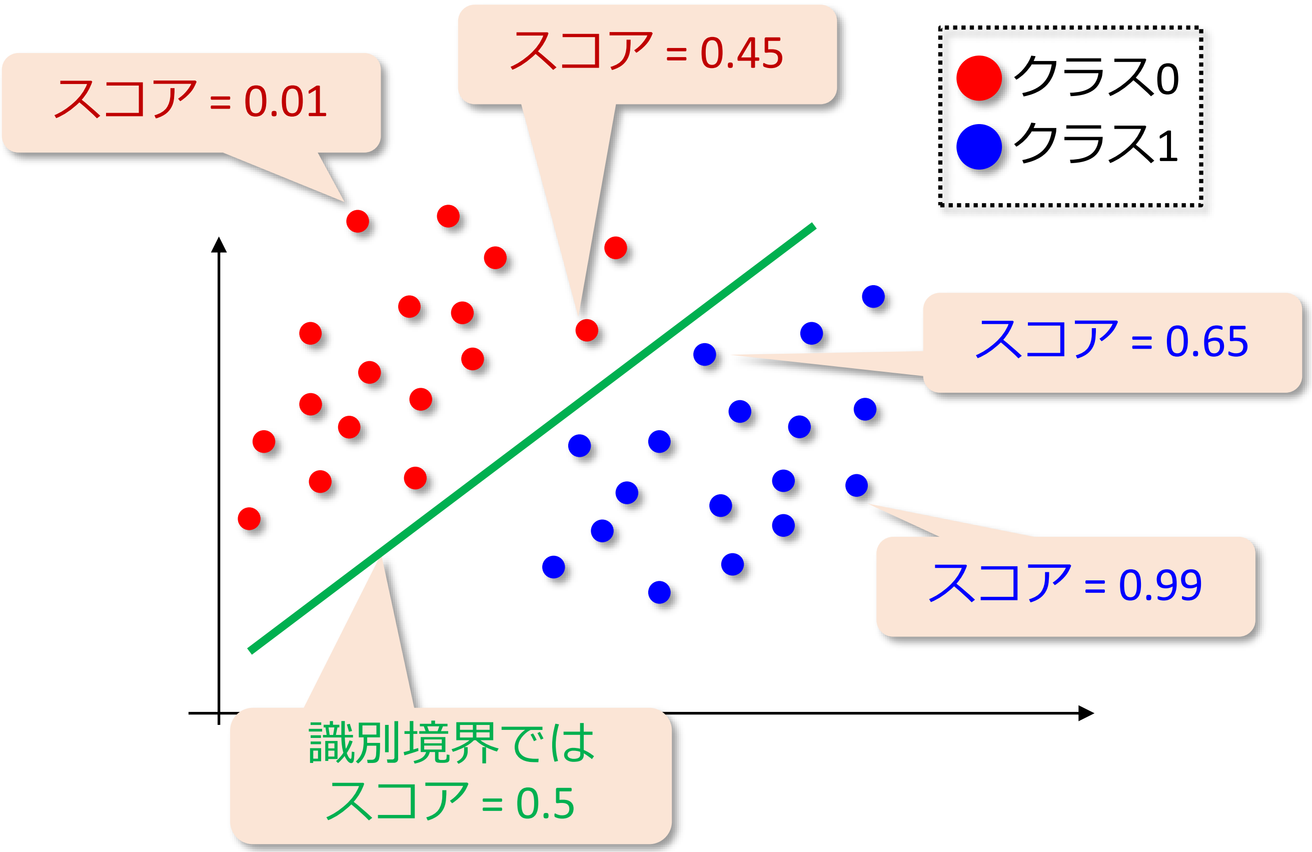
1. 元のデータ X のクラス間分散/クラス内分散比 $J = (S_{inner})^{-1} S_{intra}$ を求める。
2. J を固有値と固有ベクトルに分解する。
3. 固有値が最大となる固有ベクトル（第一固有ベクトル）を取り出す。
4. 抽出した第一固有ベクトルがクラスを識別する境界となっている。

06_01_linear_discriminant_analysis.ipynb を動かして，実際に見てみましょう。

ロジスティック回帰による クラスタリング

ロジスティック回帰クラスタリングの目的

2種類のクラスに対して，それぞれのクラスラベルを0，1，としたとき，
あるサンプルの識別結果を 0 から 1 までの連続値スコアとして出力したい。

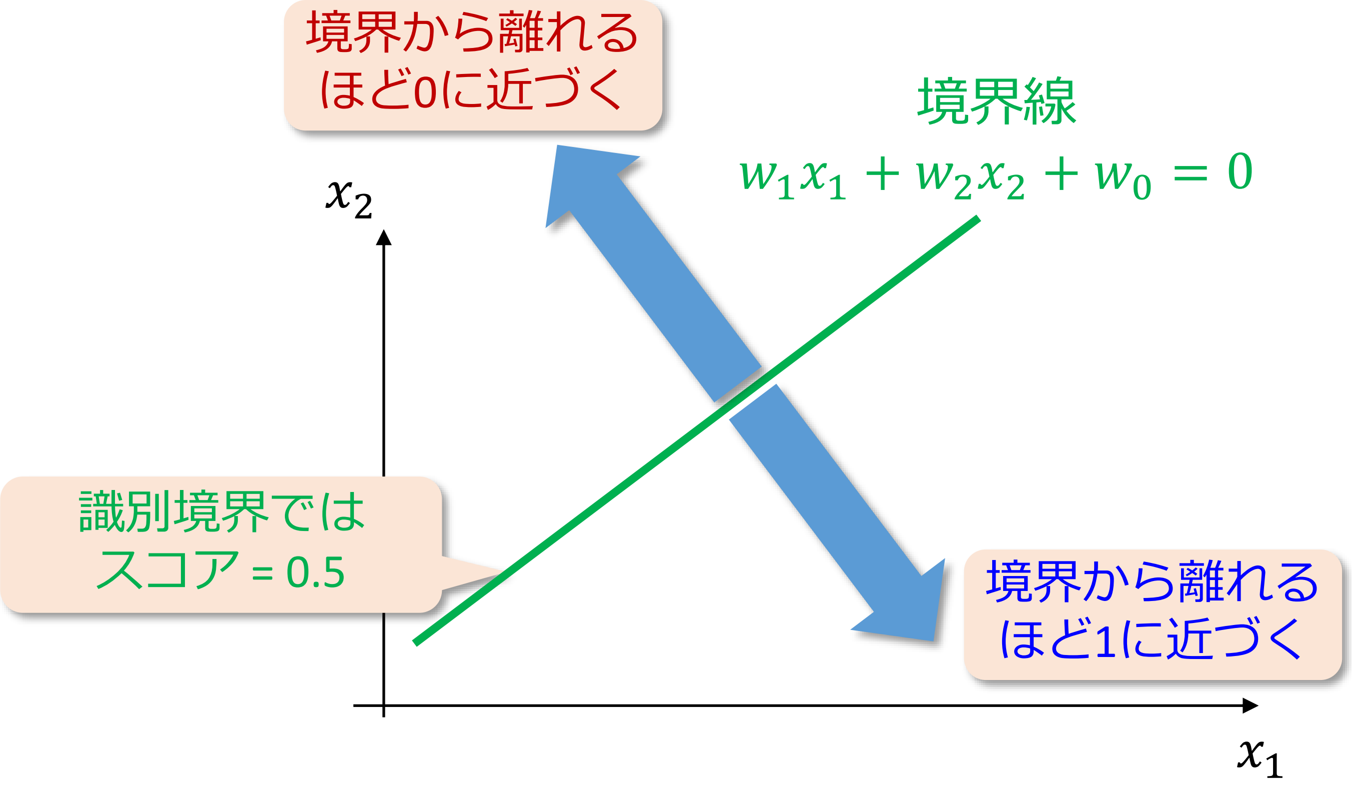


連続値のスコアとして出力できると，
単に識別クラスだけでなく，
どれくらい近いかの尺度としても使えます。

ロジスティック回帰の定式化

2次元データ(x_1, x_2)に対して，
- 境界線 $w_1x_1 + w_2x_2 + w_0 = 0$ の上ではスコアが0.5になる。
- $w_1x_1 + w_2x_2 + w_0$が負になると0に近づく。
- $w_1x_1 + w_2x_2 + w_0$が正になると1に近づく。

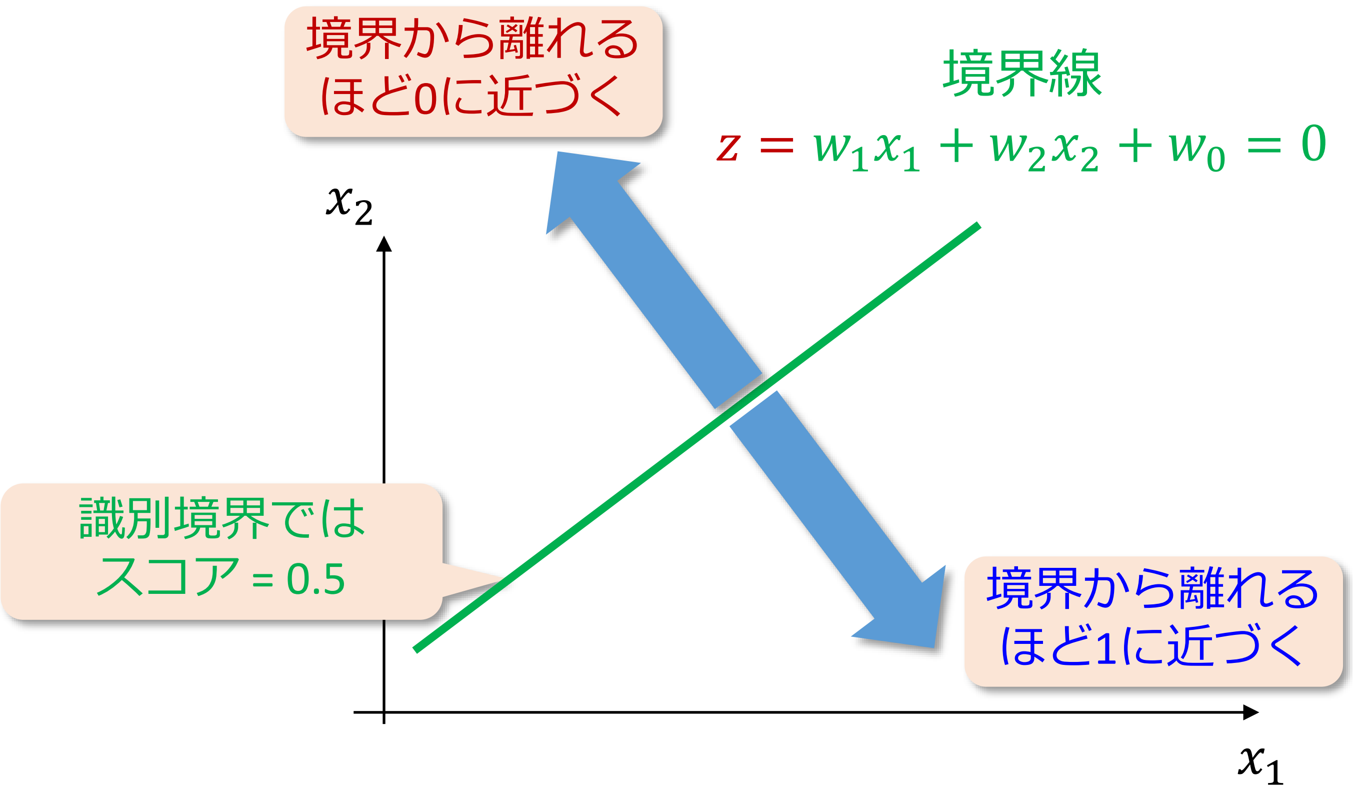
関数を考えます。

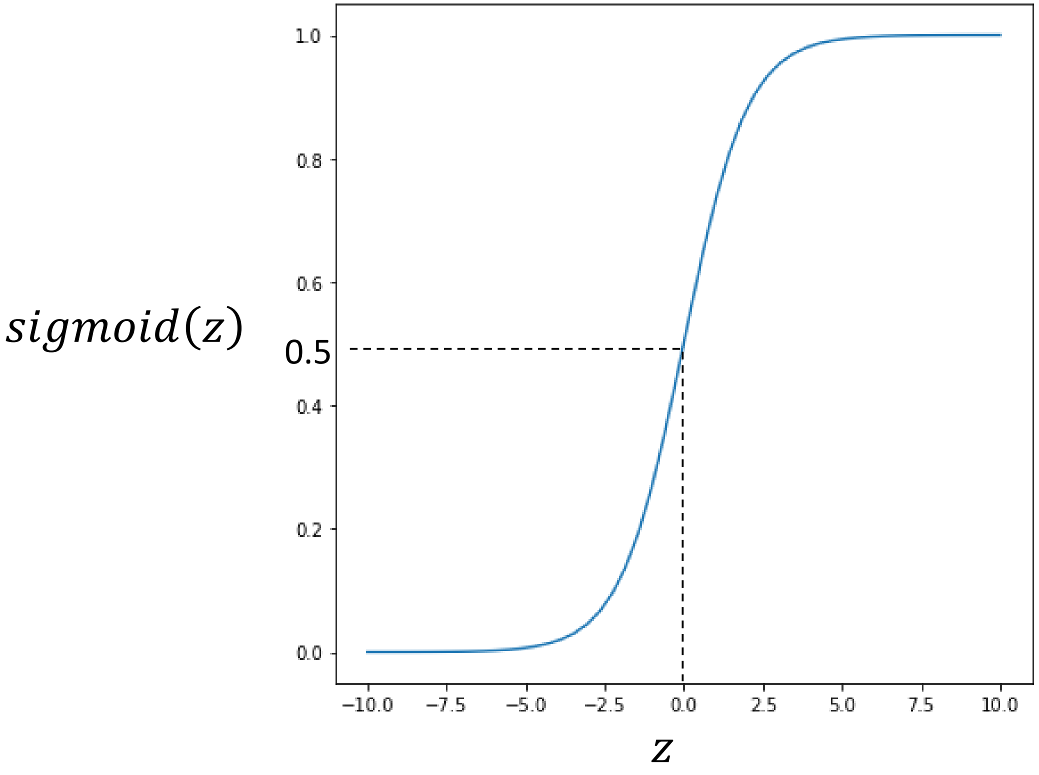


ロジスティック回帰の定式化

前ページで述べたような関数は「**シグモイド関数**」によって実現できます。

$$sigmoid(z) = \frac{1}{1 + e^{-z}}$$





ロジスティック回帰の定式化

以上をまとめると，ロジスティック回帰は以下のように定式化されます。

ある2次元のサンプル(x_1, x_2)のロジスティック回帰結果を$\hat{y}$とすると

各次元に対する重み係数

$$z = w_1 x_1 + w_2 x_2 + w_0$$

識別境界の切片

$$\hat{y} = sigmoid(z) = \frac{1}{1 + e^{-z}}$$

求めるべきパラメータはw_0, w_1, w_2です。

パラメータはどうやって求める？

今，各データには0 or 1のクラスラベルが付与されています。
つまり，ロジスティック回帰結果 $\hat{y}$ に対して正解の値 y があります。
そこで，正解の値とロジスティック回帰結果との誤差を損失関数 L として，
L を最小化するようにパラメータ w_0, w_1, w_2 を求めます。
つまり，第3回，第4回で行った**最小二乗誤差基準**を使って最適化します。

$$L = (y - \hat{y})^2 \rightarrow minimize$$

第3回，第4回では L の偏微分=0を解いていましたが，ロジスティック回帰は式がやや複雑なので，= 0を解くのが難しいです。
そこで，ここでは「**勾配降下法**」を使って最適化します。

勾配降下法とは？

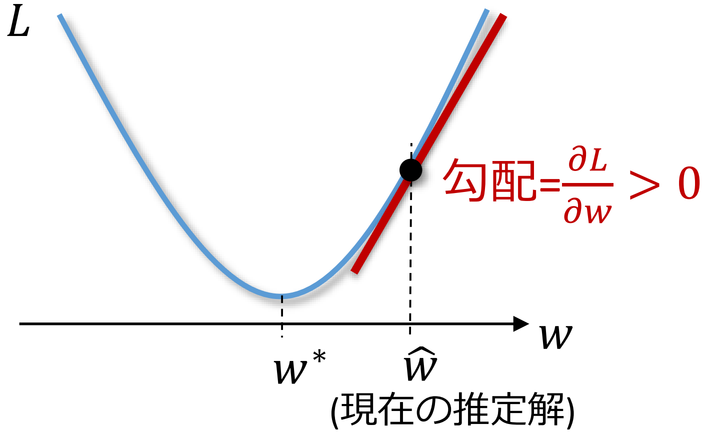
勾配に基づいてパラメータを逐次更新しながら，少しずつ最適解に近づけて行く方法です。

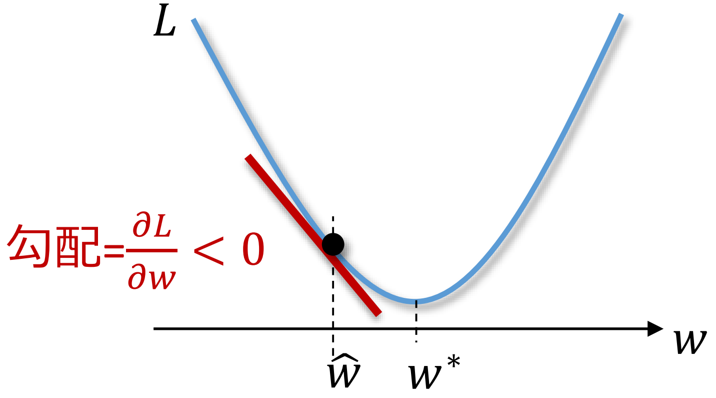
損失関数 L が下に凸の関数で，パラメータ w^* のときに最小になるとします。

もし現在のパラメータ $\hat{w}$ がw^*より大きければ，その点における関数 L の接線の傾き（=**勾配**）は正になります。

逆にw^*より小さければ，勾配は負になります。

つまり，勾配を見れば，現在のパラメータが最適解より大きいか小さいかが分かることになります。



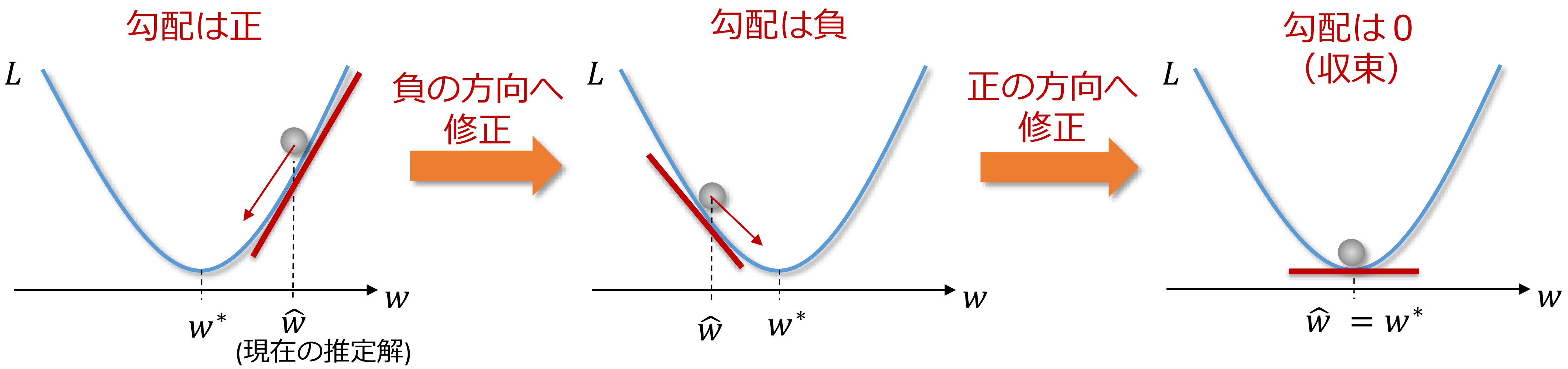


勾配降下法とは？

そこで，計算した勾配を使って，以下の式によってパラメータを更新していきます。
お椀にボールを転がすような形でパラメータが最適解へ導かれます。

$$\hat{w}_{next} \leftarrow \hat{w}_{old} - \mu \frac{\partial L}{\partial w}$$

μ は学習率と呼ばれ，更新の度合いを調節するパラメータです。



勾配降下法の動作を確認しよう

06_02_gradient_descent.ipynb を動かして，勾配降下法の挙動を確認しましょう。

また，学習率による挙動の差や局所最適解について学びましょう。

ロジスティック回帰の学習

ロジスティック回帰の定式化および勾配降下法の式をまとめます。

ロジスティック回帰

$$z = w_1 x_1 + w_2 x_2 + w_0$$

$$\hat{y} = sigmoid(z) = \frac{1}{1 + e^{-z}}$$

平均二乗誤差損失関数

$$L = (y - \hat{y})^2$$

勾配降下法

$$\widehat{w}_{next} \leftarrow \widehat{w}_{old} - \mu \frac{\partial L}{\partial w}$$

実際に $\frac{\partial L}{\partial w}$ を計算し，ロジスティック回帰の更新式を導出しましょう。

ロジスティック回帰の更新式

合成関数の偏微分の公式を使って，$\frac{\partial L}{\partial w}$ を計算します。

$$\frac{\partial L}{\partial w} = \frac{\partial L}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial z} \frac{\partial z}{\partial w}$$

$$\frac{\partial L}{\partial \hat{y}} = \frac{\partial}{\partial \hat{y}} (y - \hat{y})^2 = 2(\hat{y} - y)$$ (ただし2は定数なので省略する)

$$\frac{\partial \hat{y}}{\partial z} = \frac{\partial}{\partial z} \frac{1}{1 + e^{-z}} = \frac{e^{-z}}{(1 + e^{-z})^2} = \frac{e^{-z}}{1 + e^{-z}} \frac{1}{1 + e^{-z}} = \left(\frac{1 + e^{-z}}{1 + e^{-z}} - \frac{1}{1 + e^{-z}} \right) \frac{1}{1 + e^{-z}} = (1 - \hat{y})\hat{y}$$

$$\frac{\partial z}{\partial w} = \frac{\partial}{\partial w} w_0 + w_1 x_1 + w_2 x_2 = [1, x_1, x_2]$$ (← それぞれw_0, w_1, w_2に関する偏微分)

よって，w_0, w_1, w_2に関する勾配は

$$\frac{\partial L}{\partial w} = (\hat{y} - y)(1 - \hat{y})\hat{y} * [1, x_1, x_2]$$

ロジスティック回帰を動かしてみよう

06_03_logistic_regression.ipynb を動かして，ロジスティック回帰の動作を確認しましょう。

おわりに

今回は，教師ありクラスタリングの基本的な方法を解説しました。

特にロジスティック回帰と勾配降下法は，以降で説明するニューラルネットワークを理解する上で重要な理論となります。

しっかり理解するようにしましょう。

レポート課題 その１

第６回ファイル一式に含まれる， “2class_data_report.csv” に対して線形判別分析とロジスティック回帰をそれぞれ適用し，結果にどのような違いが出るかを確認せよ。

また，なぜ違いが出るのかについて理由や，手法の良し悪しについて考察せよ。

なお，ロジスティック回帰の学習率やエポック数は自由に設定してよい。

レポート課題 その２（オプション）

損失関数として，p.17の平均二乗誤差のほかに，**クロスエントロピー**も存在する。

$$L_{ce} = -y \log \hat{y} - (1 - y) \log(1 - \hat{y}) \rightarrow minimize$$

06_03_logistic_regression.ipynb で使用した “2class_data.py” に対して
平均二乗誤差基準の損失関数を使った場合，クロスエントロピー基準の
損失関数を使った場合のロジスティック回帰を適用せよ。
その結果，学習過程にどのような違いが出るかを確認し，なぜそのような
違いが出るのかについて考察せよ。

レポート課題 その３（オプション）

06_02_gradient_descent.ipynb において，局所最適解の説明で用いた損失関数の例

$$L = \sin(x) + 0.05(5 - x)^2$$

に対して，大域最適解が得られるように改良を行え。

レポート提出期限：6/14(火) AM10:30，ipynbファイルをhtmlファイルに変換して提出

※6/7 は第一クォータ定期試験期間のため授業は無し