Table of Contents

Breast Cancer Wisconsin (Diagnostic) Data Set

https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+(diagnostic)

Attribute Information:

1. ID number
2. Diagnosis (M = malignant:-1, B = benign:1) M:悪性, B:良性
3. 3-32

Ten real-valued features are computed for each cell nucleus:

- 半径radius (mean of distances from center to points on the perimeter)
- テクスチャtexture (standard deviation of gray-scale values)
- 境界の長さperimeter
- 面積area
- なめらかさsmoothness (local variation in radius lengths)
- コンパクトさcompactness (perimeter^2 / area - 1.0)
- くぼみ度合いconcavity (severity of concave portions of the contour)
- くぼみの数concave points (number of concave portions of the contour)
- 対称性symmetry
- フラクタル次元fractal dimension ("coastline approximation" - 1)

のそれぞれのmean, stderr, worst数値を保持している.

http://people.idsia.ch/~juergen/deeplearningwinsMICCAIgrandchallenge.html

分類器

用意された訓練(training)データには, $\boldsymbol{A}$に上に記した特徴量が, $\boldsymbol{b}$に悪性(-1)か良性(1)かを 示す数値が入っている.

与えられた特徴ベクトル$\boldsymbol{y}$に対し, 細胞組織が悪性か良性かを分類する関数$C(\boldsymbol{y})$を選び出すプログラムを作成しよう.

仮説クラス

分類器は可能な分類器の集合(**仮説クラス**)から選ばれる. この場合, 仮説クラスとは特徴ベクトルの空間$\mathbb{R}^D$から$\mathbb{R}$への線形関数$h(\cdot)$である. すると分類器は次のような関数として定義される.

$$
C(\boldsymbol{y}) = \begin{cases} +1 & \text{when} \quad h(\boldsymbol{y}) \geq 0 \\ -1 & \text{when} \quad h(\boldsymbol{y}) < 0 \end{cases}
$$

各線形関数$h : \mathbb{R}^D \to \mathbb{R}$に対して, 次のような$D$ベクトル$\boldsymbol{w}$が存在する.

$$
h(\boldsymbol{y}) = \boldsymbol{w} \cdot \boldsymbol{y}
$$

したがって, そのような線形関数を選ぶことは, 結局Dベクトル$\boldsymbol{w}$を 選ぶことに等しい. 特に, $\boldsymbol{w}$を選ぶことは, 仮説クラスhを 選ぶことと等価なので, **$\boldsymbol{w}$を仮説ベクトル**と呼ぶ.

問題を単純化すると, 分類器を単なるベクトルとみなして, データとの掛け算で予測がつきます. 本来は予測を-1,1とかに投影しないといけないんですが, 単純化のためにそのままの値を用います. 問題はどうやってこの仮説ベクトル$\boldsymbol{w}$の各要素の値を決定するか？ですよね.

最急降下法

損失関数に

最小二乗法

$$
L(w) = \sum_{i=1}^{n} (A_i \cdot \boldsymbol{w} - b_i)^2
$$

を選ぶと, ベクトル$\boldsymbol{w}$の j偏微分は,

j = 30

$$
\begin{aligned}
\frac{\partial L}{\partial w_j} &= \sum_{i=1}^{n} \frac{\partial}{\partial w_j}(A_i \cdot w - b_i)^2 \\
&= \sum_{i=1}^{n} 2(A_i \cdot w - b_i) A_{ij}
\end{aligned}
$$

連立方程式

500

となる. ここで, A_{ij}はA_iのj番目の要素を意味する. この偏微分$\frac{\partial L}{\partial w_j}$を $\boldsymbol{w}_j$の勾配(slope)として, $L(w)$の極小値(local minimum)を求める.

このような探索方法を最急降下法(steepest descent method)と呼ぶ.

print_w

出てきた$\boldsymbol{w}$のj要素をきれいに表示する関数を用意しておきます.

```
In [ ]: def print_w(w):
```