サポートベクトルマシーン

$\{(\mathbf{x}_i, y_i)\}_{i=1}^N$ をラベル付けされたデータとし、N をデータの数、$\mathbf{x}_i$ を D 次元特徴ベクトル、y_i を $\mathbf{x}_i$ のラベルとする。ただし、y_i の値は $-1, 1$ のいずれかであるとする。$\mathbf{w}$ を D 次元ベクトル、b を実数とし、

$$f_{\mathbf{w},b}(\mathbf{x}) := \mathbf{w}\mathbf{x} + b$$

とおく。この式を用いて、未知の D 次元特徴ベクトル $\mathbf{x}$ に対して、ラベル $y = f_{\mathbf{w},b}(\mathbf{x})$ を予測する。すなわち、$f_{\mathbf{w},b}$ が以下の条件

(1) $y_i = 1$ ならば、$f_{\mathbf{w},b}(\mathbf{x}_i) \geq 0$
(2) $y_i = -1$ ならば、$f_{\mathbf{w},b}(\mathbf{x}_i) < 0$

を満たすような $\mathbf{w}, b$ の値を求める。上の条件は

$$y_i f_{\mathbf{w},b}(\mathbf{x}_i) \geq 0 \tag{1}$$

という条件にまとめることができる。$f_{\mathbf{w},b}(\mathbf{x}) = 0$ を**分類境界**という。$f_{\mathbf{w},b}(\mathbf{x}) = 0$ は自身を境界線として、X を二つのグループ

$$A = \{\mathbf{x} \in X \mid f_{\mathbf{w},b}(\mathbf{x}_i) \geq 0\}, B = \{\mathbf{x} \in X \mid f_{\mathbf{w},b}(\mathbf{x}_i) < 0\}$$

へと分類する。このように、X を $f_{\mathbf{w},b}(\mathbf{x}) = 0$ によって、二つのグループへと分類することを**ハードマージン**という。分類境界 $f_{\mathbf{w},b}(\mathbf{x}) = 0$ と分類境界から最も近くにあるデータ $\mathbf{x}_i \in X$ との距離は $|f_{\mathbf{w},b}(\mathbf{x}_i)|/\|\mathbf{w}\|$ である。(1) より、$|f_{\mathbf{w},b}(\mathbf{x}_i)|/\|\mathbf{w}\| = y_i f_{\mathbf{w},b}(\mathbf{x}_i)/\|\mathbf{w}\|$ が成り立つ。

$$M(\mathbf{w}, b) := \min\{y_i f_{\mathbf{w},b}(\mathbf{x}_i) \mid 1 \leq i \leq N\}$$

とおく。(1) を満たす $\mathbf{w}, b$ の中で、$M(\mathbf{w}, b)/\|\mathbf{w}\|$ を最大化するものを求める。すなわち、最適化問題

$$\max_{\mathbf{w},b} M(\mathbf{w}, b)/\|\mathbf{w}\|, \quad \text{ただし、全ての } 1 \leq i \leq N \text{ に対して、} M(\mathbf{w}, b) \leq y_i f_{\mathbf{w},b}(\mathbf{x}_i) \tag{2}$$

を解く。これは、分類境界 $f_{\mathbf{w},b}(\mathbf{x}) = 0$ と分類境界からデータへの距離を最大化することを意味する。(2) は以下の扱いやすい形

$$\min_{\mathbf{w},b} \|\mathbf{w}\|^2/2, \quad \text{ただし、全ての } 1 \leq i \leq N \text{ に対して、} y_i f_{\mathbf{w},b}(\mathbf{x}_i) \geq 1 \tag{3}$$

へと変形できる。ハードマージンでは、X を二つのグループへと分類する分類境界 $f_{\mathbf{w},b}(\mathbf{x}) = 0$ が存在することを仮定した。しかし、現実の多くの問題においては、そのような仮定は強すぎる。以下では、この仮定をなくし、必ずしも二つのグループへと分類できないようなデータへ適用できるように拡張する。これを行うために、$\xi = (\xi_1, \ldots, \xi_N)$（$\xi_i \geq 0$）とおき、$C$ を正の実数とし、(3) を一般化して、最適化問題

$$\min_{\mathbf{w},b,\xi} (\|\mathbf{w}\|^2/2 + C \sum_{i=1}^N \xi_i), \quad \text{ただし、全ての } 1 \leq i \leq N \text{ に対して、} y_i f_{\mathbf{w},b}(\mathbf{x}_i) \geq 1 - \xi_i \tag{4}$$

を解く。このように、X が必ずしも二つのグループへと分類できないとし、分類誤差を許容した上で二つのグループへと分類することを**ソフトマージン**という。$C \to \infty$ であるとき、(4) は (3) に一致する。逆に、$C \to 0$ であるときは、誤分類が許容されやすくなる。実装する際は適切な C を設定する必要がある。

例 0.1. (コードはサポートベクトルマシーン.ipynb) iris データセットを用いてサポートベクトルマシーンを実装する。

```
#データを取得
import numpy as np
from sklearn import datasets
iris = datasets.load_iris()
X = iris["data"][:, (2, 3)]  # petal length, petal width
y = (iris["target"] == 2).astype(np.float64)  # Iris virginica

from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.svm import LinearSVC
svm_clf = Pipeline([
        ("scaler", StandardScaler()),
        ("linear_svc", LinearSVC(C=1, loss="hinge", random_state=42)),
    ])

svm_clf.fit(X, y) #データを正規化し、C=1 としてサポートベクトルマシーンを適用

svm_clf.predict([[5.5, 1.7]]) #値を予測
>array([1.])
```

参考文献

[1] Andriy Burkov. (2019). The hundred-page machine learning book.

[2] Marc Peter Deisenroth., A. Aldo Faisal., Cheng Soon Ong. (2020). Mathematics for machine learning. Cambridge University Press.

[3] Aurëlien Gëron. (2019). Hands-on machine learning with Scikit-Learn, Keras & TensorFlow. 2nd Edition. Oreilly.

[4] 小縣信也., 斎藤翔汰., 溝口聡., 若杉一幸. (2021). ディープラーニング E 資格エンジニア問題集 インプレス.

[5] Sebastian Raschka., Vahid Mirjalili. (2019). Python machine learning. Third Edition. Packt.