

線形回帰モデル

$\{(\mathbf{x}_i, y_i)\}_{i=1}^N$ 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。 \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。最適な \mathbf{w}, b は

$$\min_{\mathbf{w}, b} \frac{1}{N} \sum_{i=1}^N (f_{\mathbf{w},b}(\mathbf{x}_i) - y_i)^2$$

で求められる。

例 0.1. (コードは線形回帰モデル.ipynb) $D = 1$ である場合を考える。

#データを生成

```
import numpy as np
```

```
X = 2 * np.random.rand(100, 1)
```

```
y = 4 + 3 * X + np.random.randn(100, 1)
```

```
from sklearn.linear_model import LinearRegression
```

```
lin_reg = LinearRegression() #線形回帰モデルを選択
```

```
lin_reg.fit(X, y) #最適解を求める
```

```
lin_reg.coef_ #w の最適解
```

```
>array([[2.93623137]])
```

```
lin_reg.intercept_ #b の最適解
```

```
>array([4.14795994])
```

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] Aurélien Geron. (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.