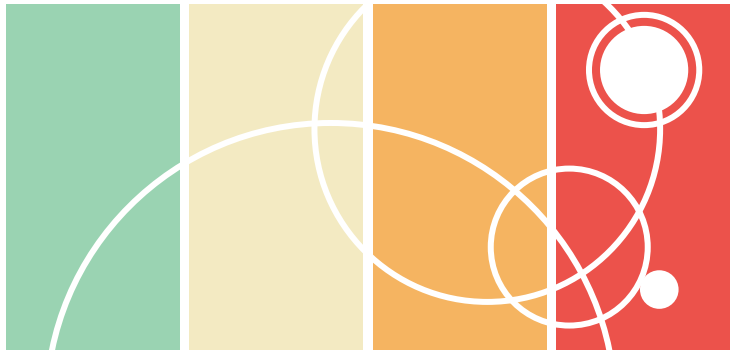


Introduction to Machine Learning

孫 建強
どんぐり研究所

免責事項

- ここで取り扱う内容は、正確である保証はありません。
- ここで取り扱う内容は、私個人の見解である。どんぐり研究所ならび現実世界における私の所属機関を代表する見解ではありません。
- ここで取り扱う内容を利用して、あなたに損害が生じても、一切責任を負いません。
- ここの内容は、予告なく変更や公開の取り消しする場合があります。

回帰分析

回帰分析

変数選択

スパース推定

様々な回帰分析

回帰分析

回帰分析

変数選択

スパース推定

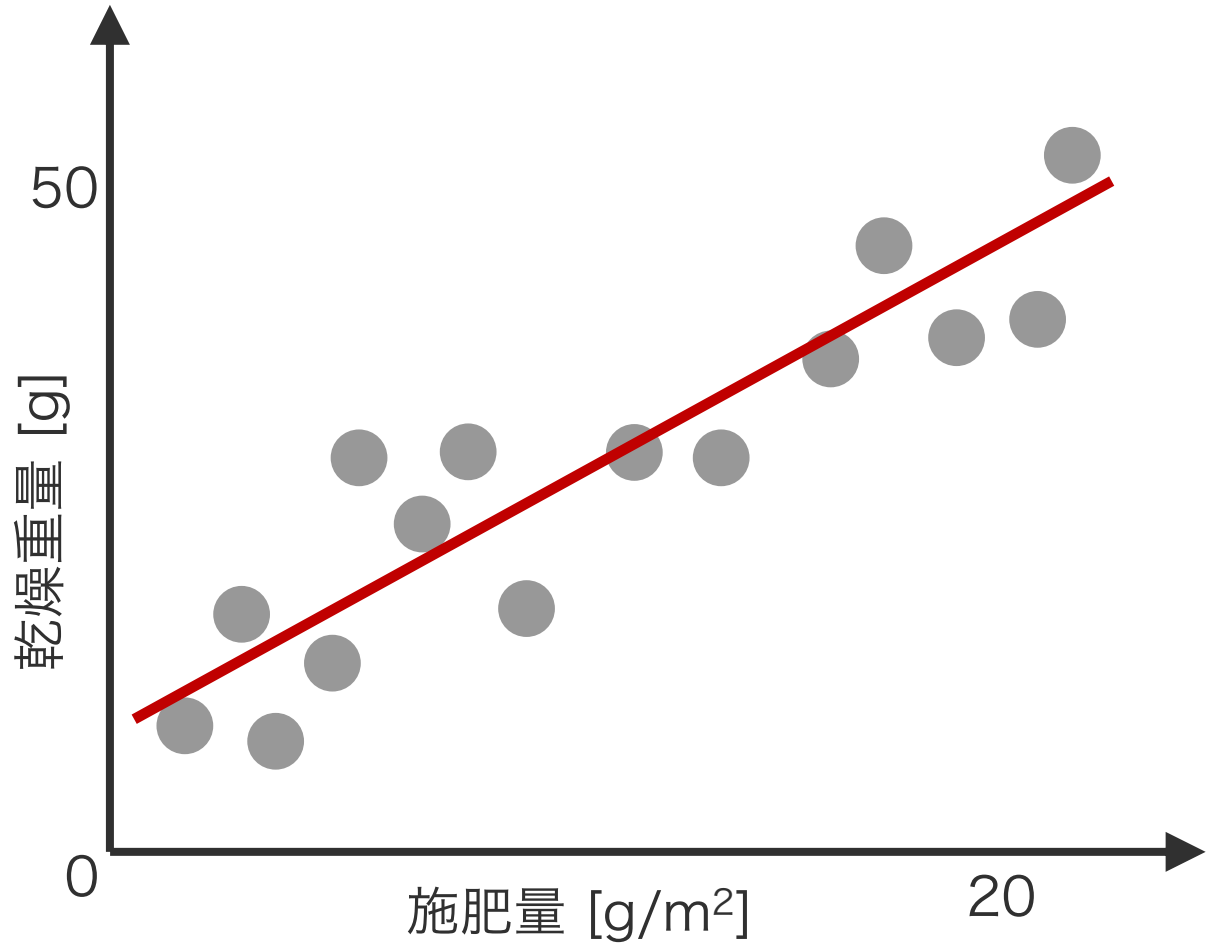
様々な回帰分析

回帰分析

回帰分析は、連続値をとる目的変数を予測するために使われる教師あり学習の一つである。具体的に、特徴量とその特徴量に応答する連続値の関係を学習するのが目的である。回帰分析では、特徴量のことを説明変数、予測したい値を目的変数という。

$$y = w_0 + w_1 x = \boldsymbol{xw}$$

▲ 乾燥重量　　▲ 施肥量

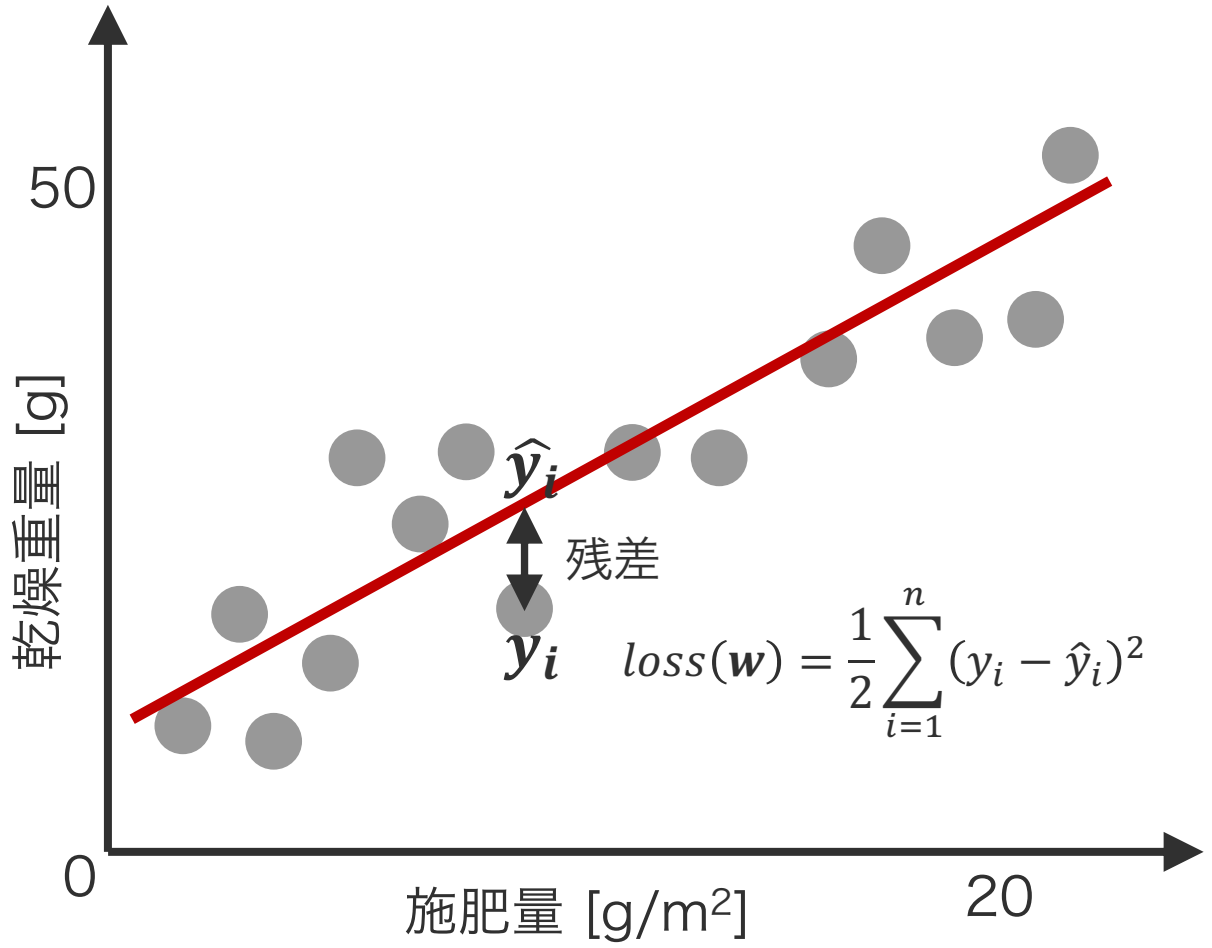


回帰分析

「特徴量とその特徴量に応答する連続値の関係」を表す最適な直線とは、観測値とその直線で予測した値の差が最も小さいということである。全サンプルの観測値と予測値の残差に着目することで、損失関数を次のように置くことができる。

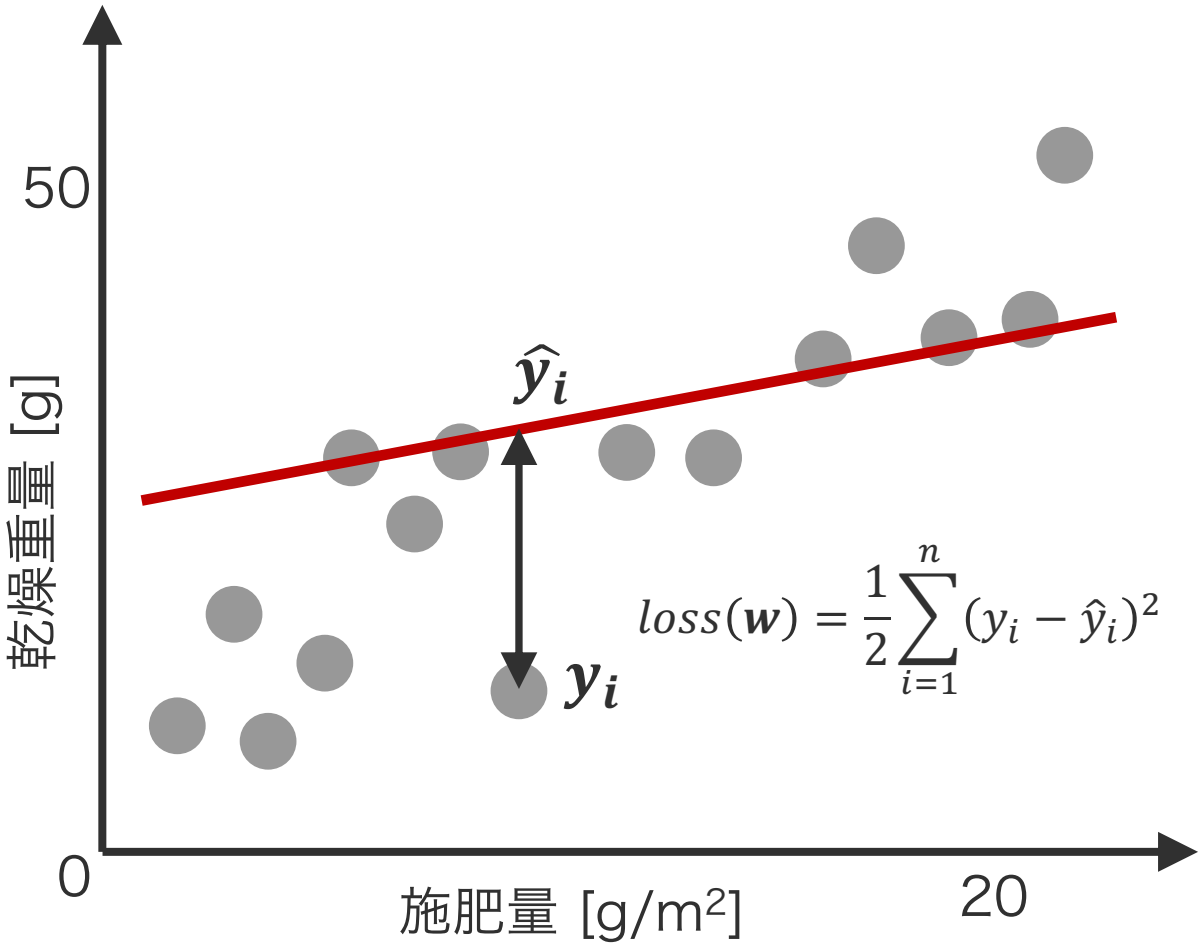
$$loss(\boldsymbol{w}) = \frac{1}{2}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

$$= \frac{1}{2}\sum_{i=1}^{n}(y_i - \boldsymbol{x}_i\boldsymbol{w})^2$$

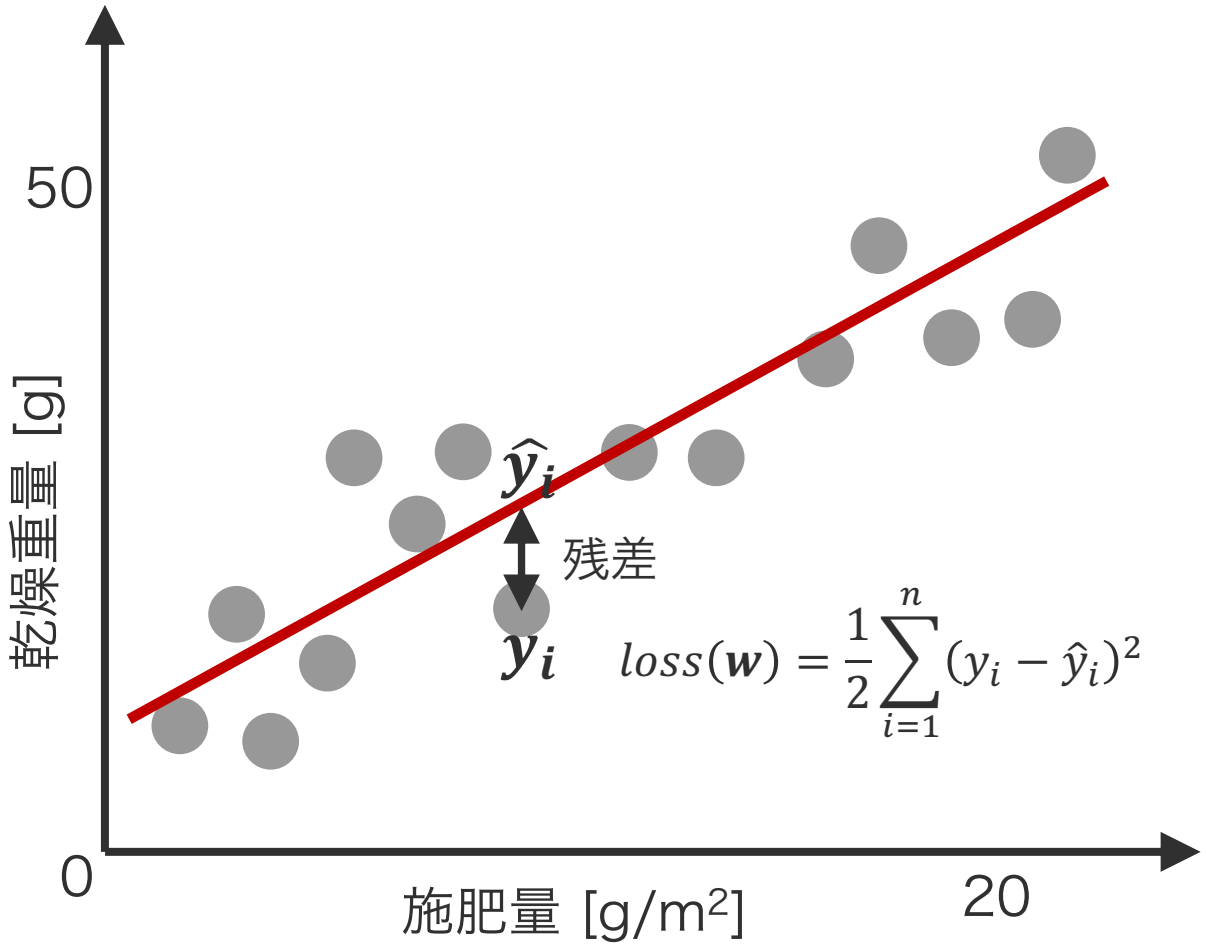


回帰分析

loss(**w**) が大きい



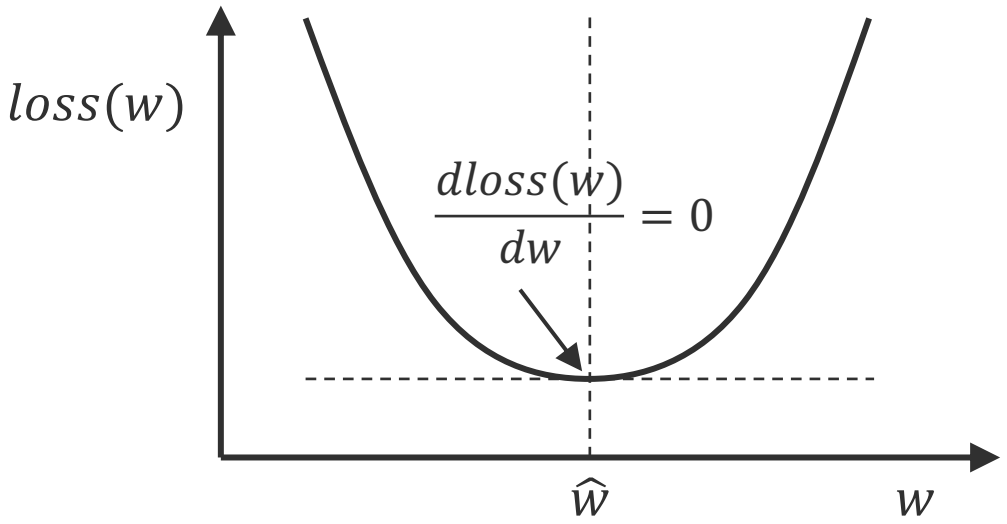
loss(**w**) が小さい



回帰分析

最適な回帰モデルを構築するには、最適な重み **w** を見つければよく、つまり、loss(**w**) を最小にする **w** を見つければよい。

$$loss(\boldsymbol{w}) = \frac{1}{2}\sum_{i=1}^{n}(y_i - \boldsymbol{x}_i\boldsymbol{w})^2$$



最小二乗法

$$\frac{\partial loss(\boldsymbol{w})}{\partial \boldsymbol{w}} = -\boldsymbol{X}^t\boldsymbol{y} + (\boldsymbol{X}^t\boldsymbol{X})^t\boldsymbol{w} = 0$$

$$\boldsymbol{w} = (\boldsymbol{X}^t\boldsymbol{X})^{-1}\boldsymbol{X}^t\boldsymbol{y}$$

- ある説明変数の傾向が他の説明変数の傾向と似ている場合、**X** の逆行列を求めることができない。

  ＞ Ridge 回帰、スパース回帰

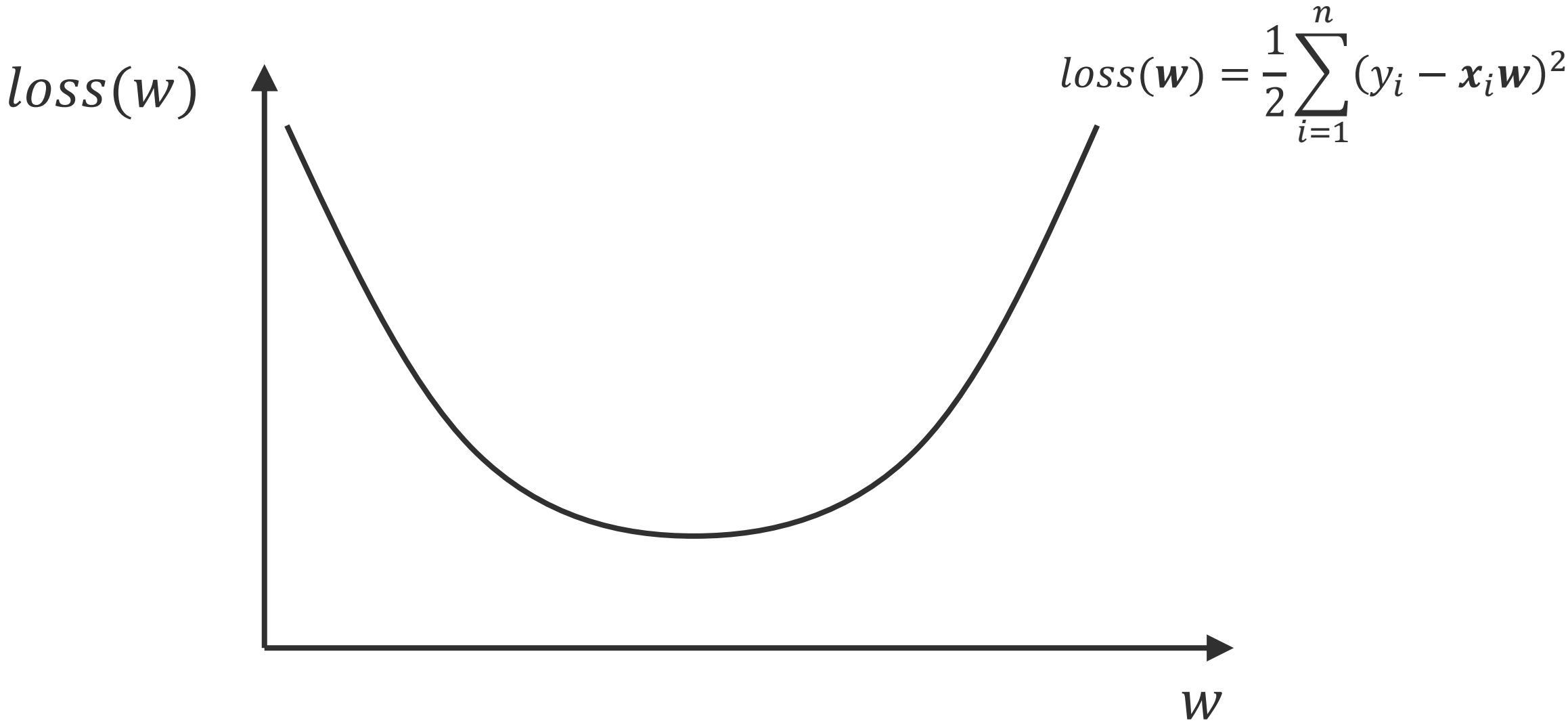
$$\boldsymbol{w} = (\boldsymbol{X}^t\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^t\boldsymbol{y}$$

- 標本数（**y**, **x**）が大量にあるとき、ハードウェアの制限で逆行列を求めることができない場合がある。

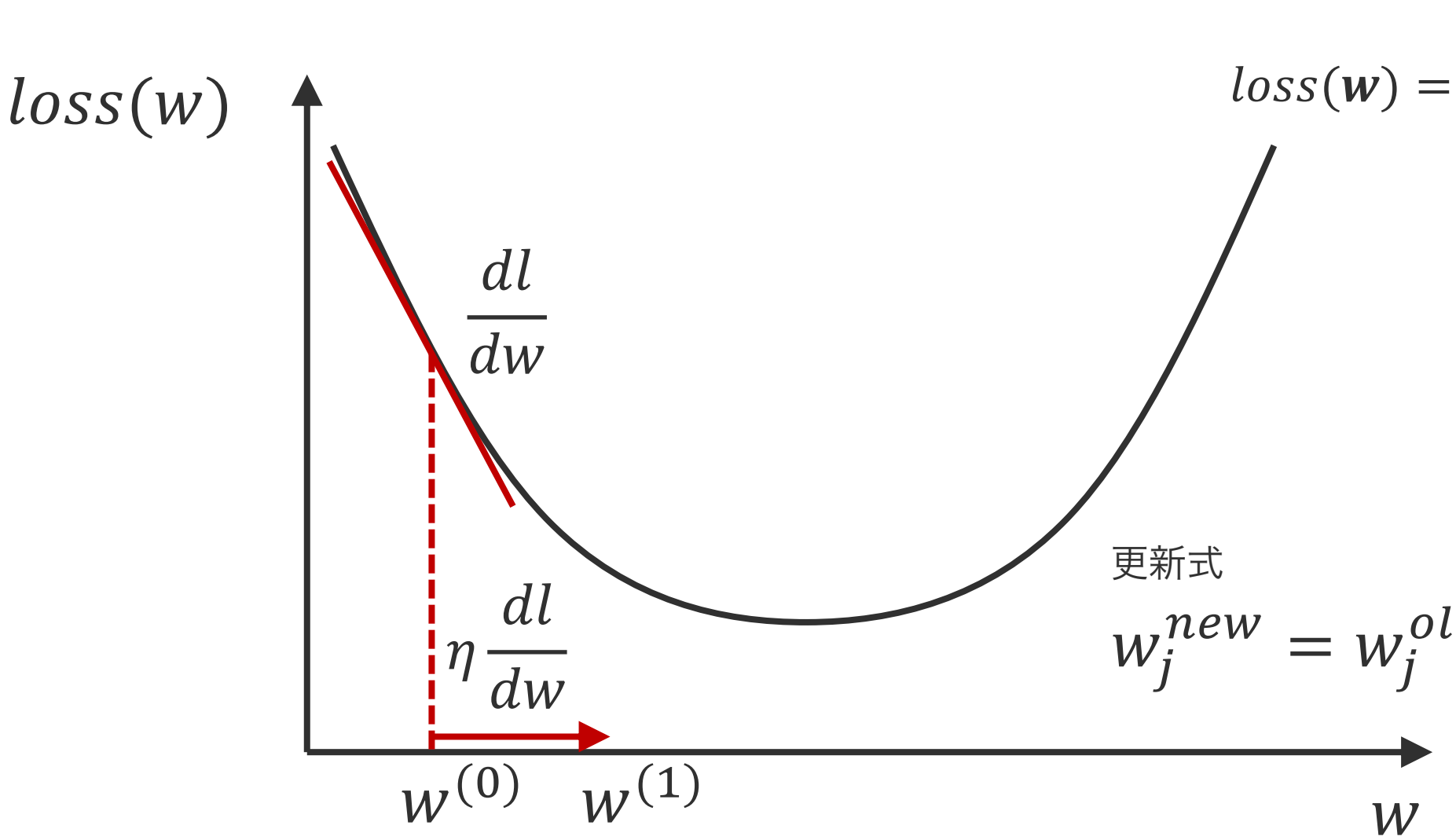
  ＞ 勾配降下法

勾配降下法

$$loss(\boldsymbol{w}) = \frac{1}{2}\sum_{i=1}^{n}(y_i - \boldsymbol{x}_i\boldsymbol{w})^2$$



勾配降下法

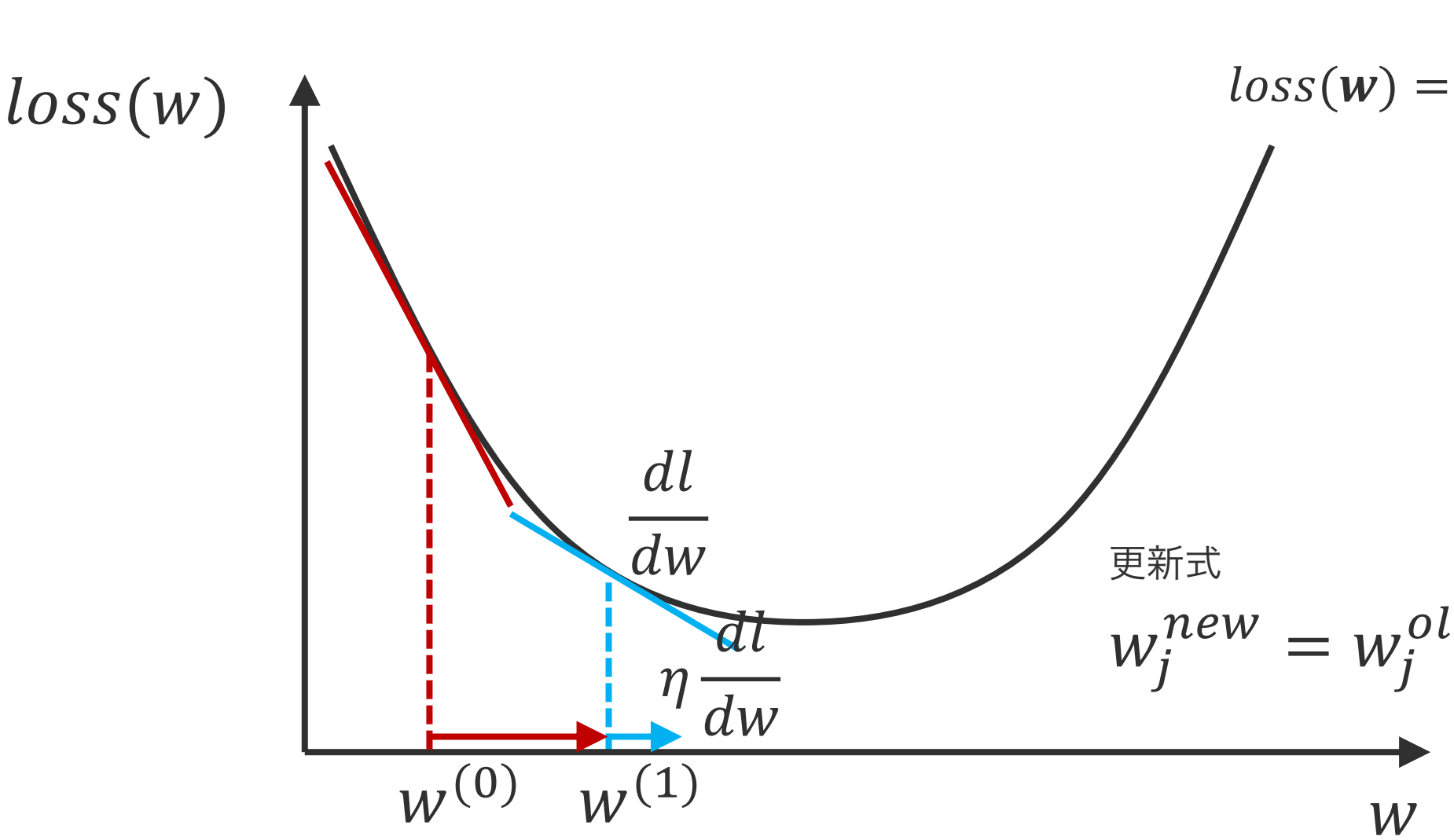


$$loss(\boldsymbol{w}) = \frac{1}{2}\sum_{i=1}^{n}(y_i - \boldsymbol{x}_i\boldsymbol{w})^2$$

更新式

$$w_j^{new} = w_j^{old} - \eta \left.\frac{\partial l}{\partial w}\right|_{w=w_j^{old}}$$

勾配降下法

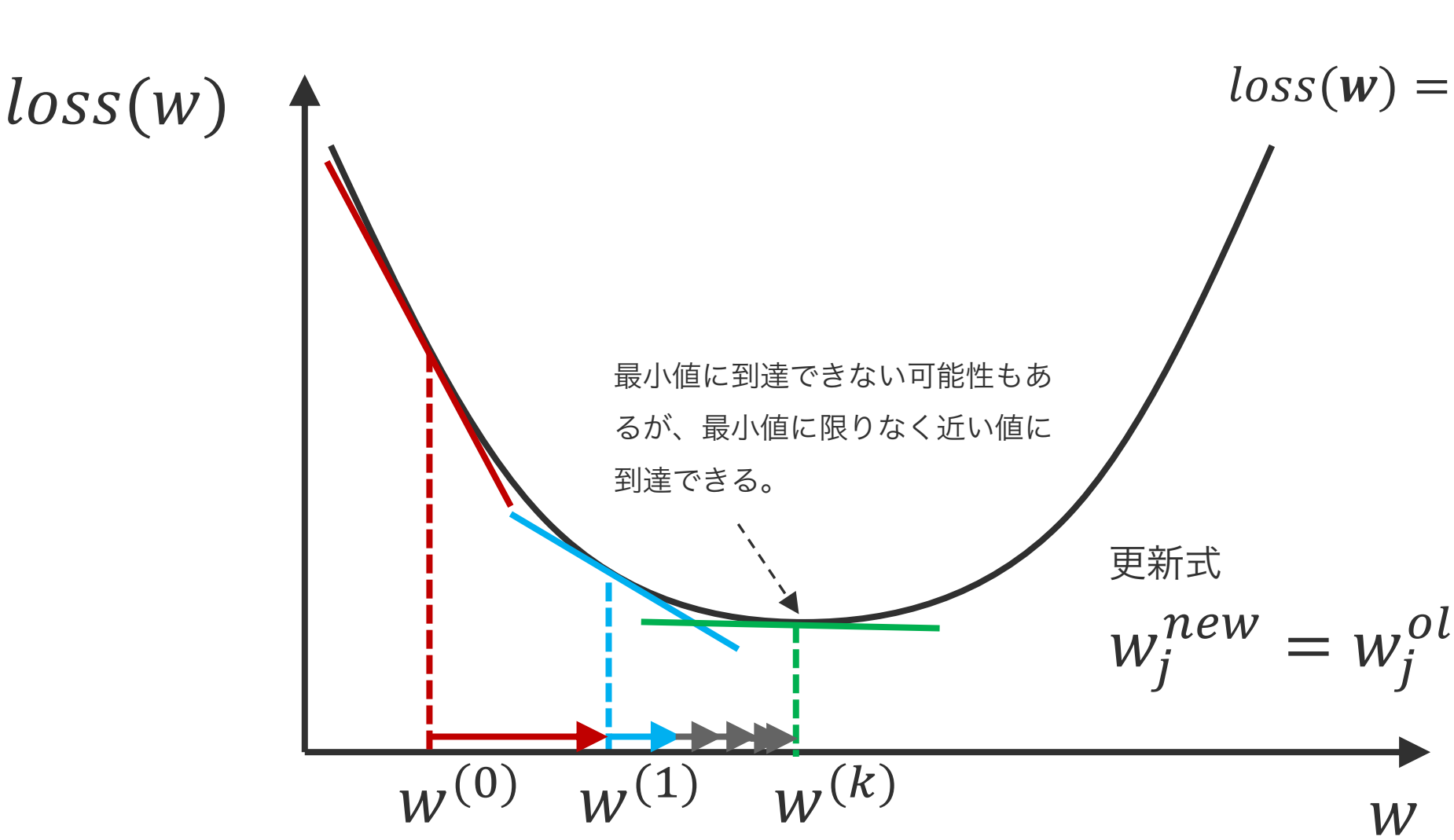


$$loss(\boldsymbol{w}) = \frac{1}{2}\sum_{i=1}^{n}(y_i - \boldsymbol{x}_i\boldsymbol{w})^2$$

更新式

$$w_j^{new} = w_j^{old} - \eta \left.\frac{\partial l}{\partial w}\right|_{w=w_j^{old}}$$

勾配降下法



$$loss(\boldsymbol{w}) = \frac{1}{2}\sum_{i=1}^{n}(y_i - \boldsymbol{x}_i\boldsymbol{w})^2$$

更新式

$$w_j^{new} = w_j^{old} - \eta \left.\frac{\partial l}{\partial w}\right|_{w=w_j^{old}}$$

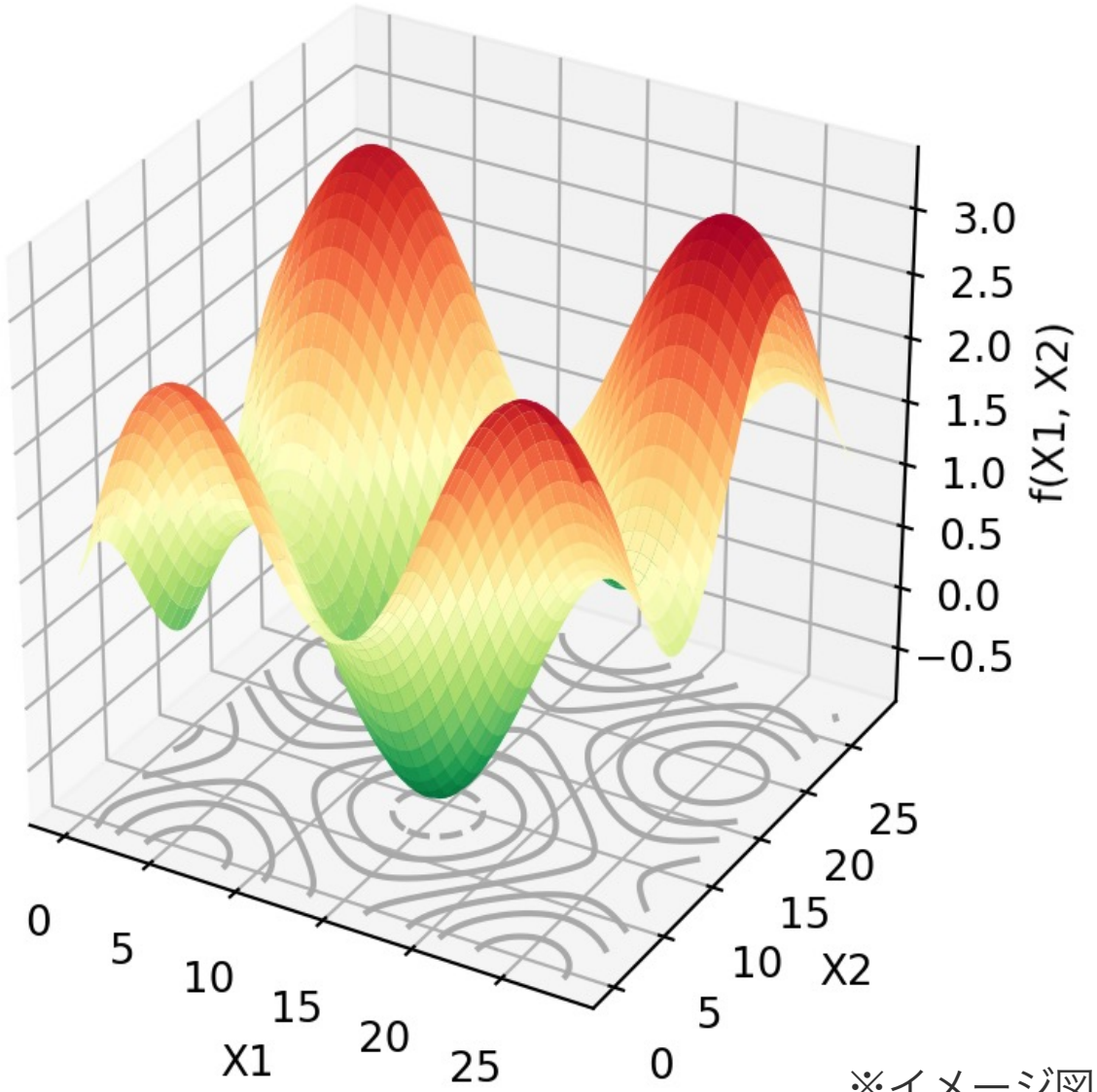
重回帰分析

説明変数が多くなると、求める **w** も多くなる。
そのため、loss(**w**) 関数の形は複雑になり、最小値を求めるのがますます困難になる。

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

▲ 乾燥重量　▲ 施肥量　▲ 気温

$$loss(\boldsymbol{w}) = \frac{1}{2}\sum_{i=1}^{n}(y_i - \boldsymbol{x}_i \boldsymbol{w})^2$$



※イメージ図

回帰分析

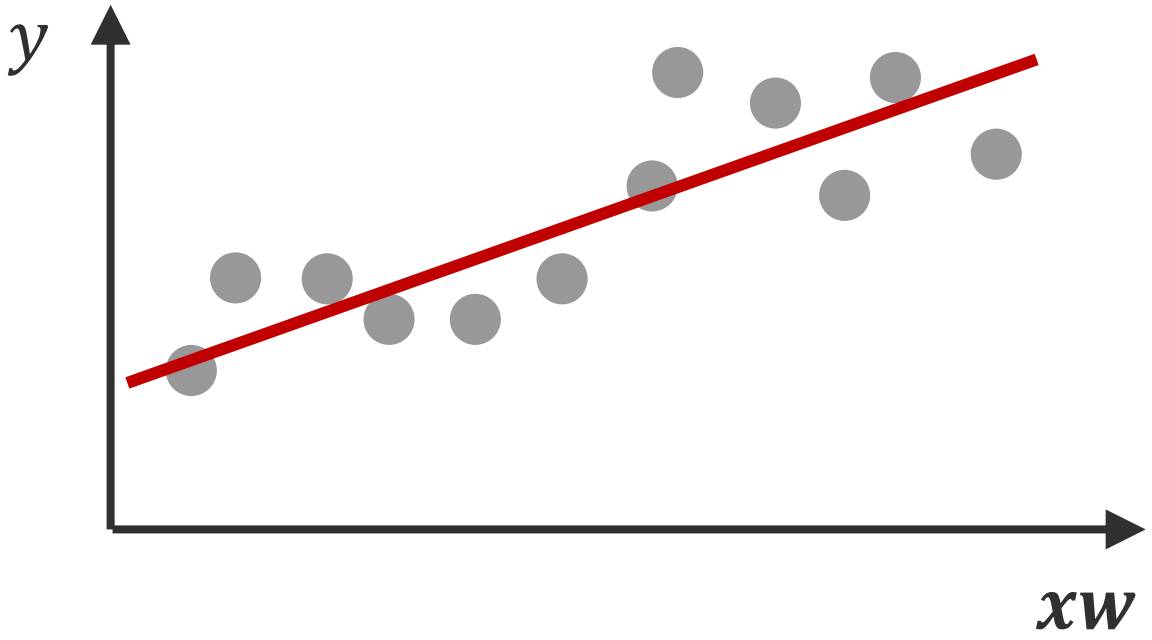
- 回帰分析
-  変数選択
- スパース推定
- 様々な回帰分析

回帰分析

単純なモデル

$$y = w_0 + w_1 x_1$$

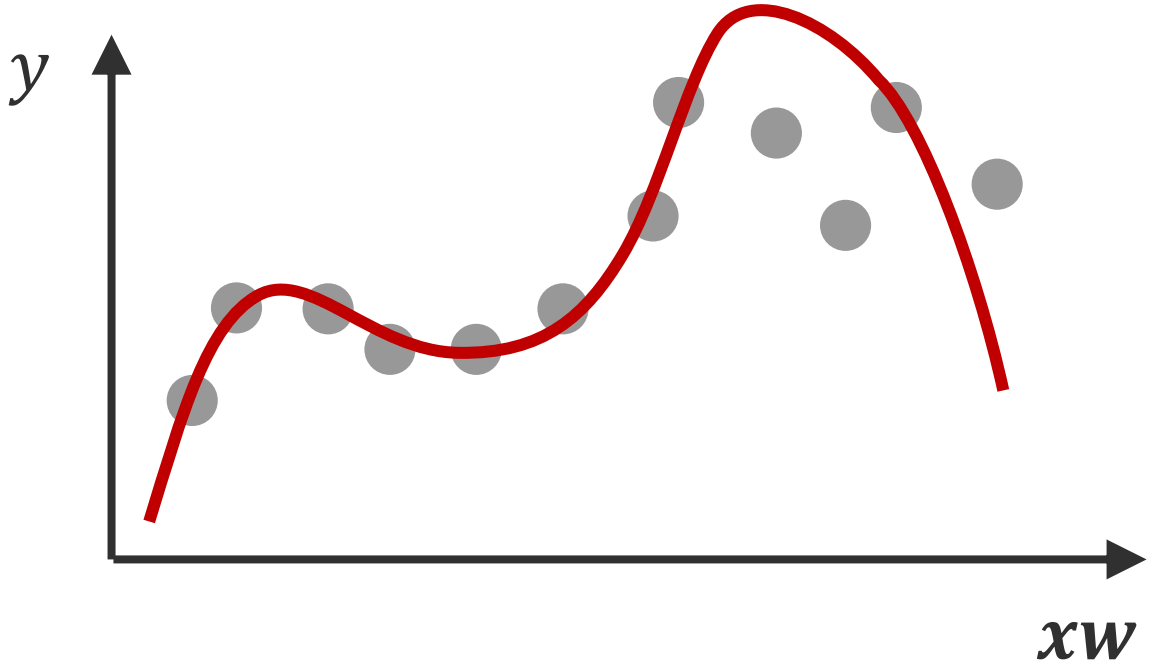
乾燥重量　施肥量



複雑なモデル

$$y = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_3$$

乾燥重量　施肥量　気温　日照

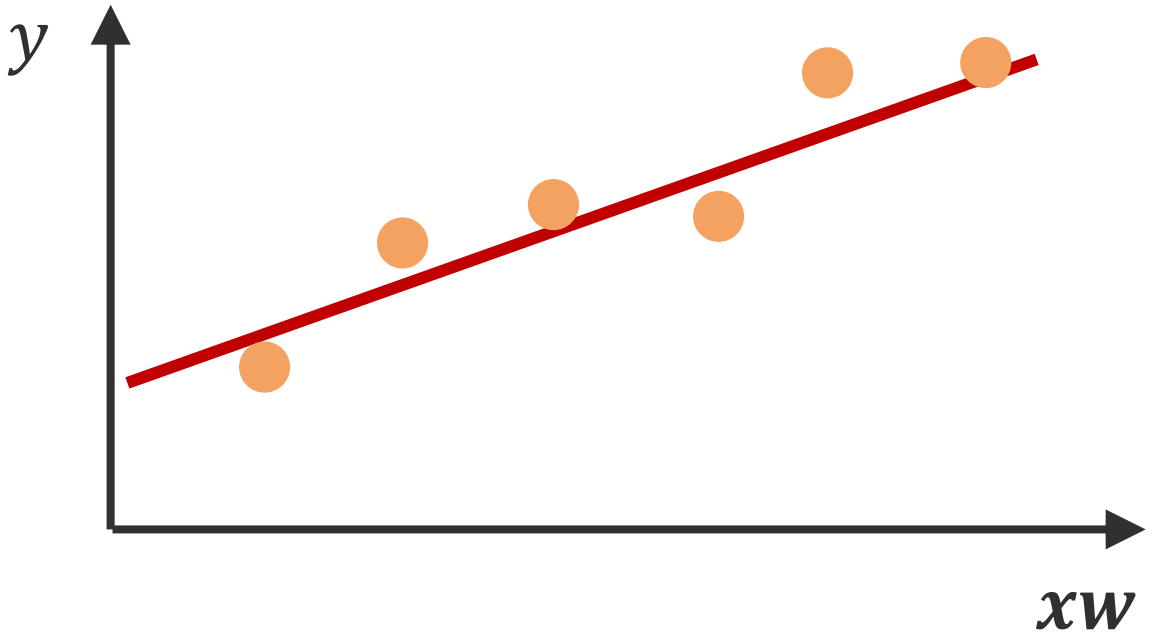


回帰分析

単純なモデル

$$y = w_0 + w_1 x_1$$

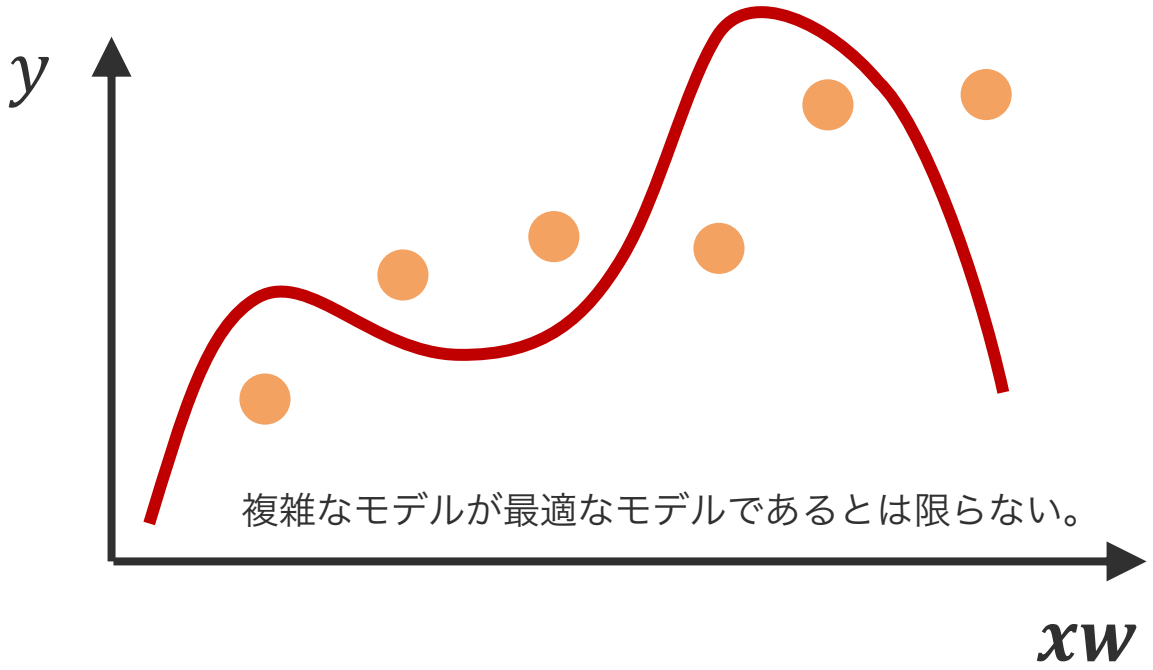
乾燥重量　施肥量



複雑なモデル

$$y = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_3$$

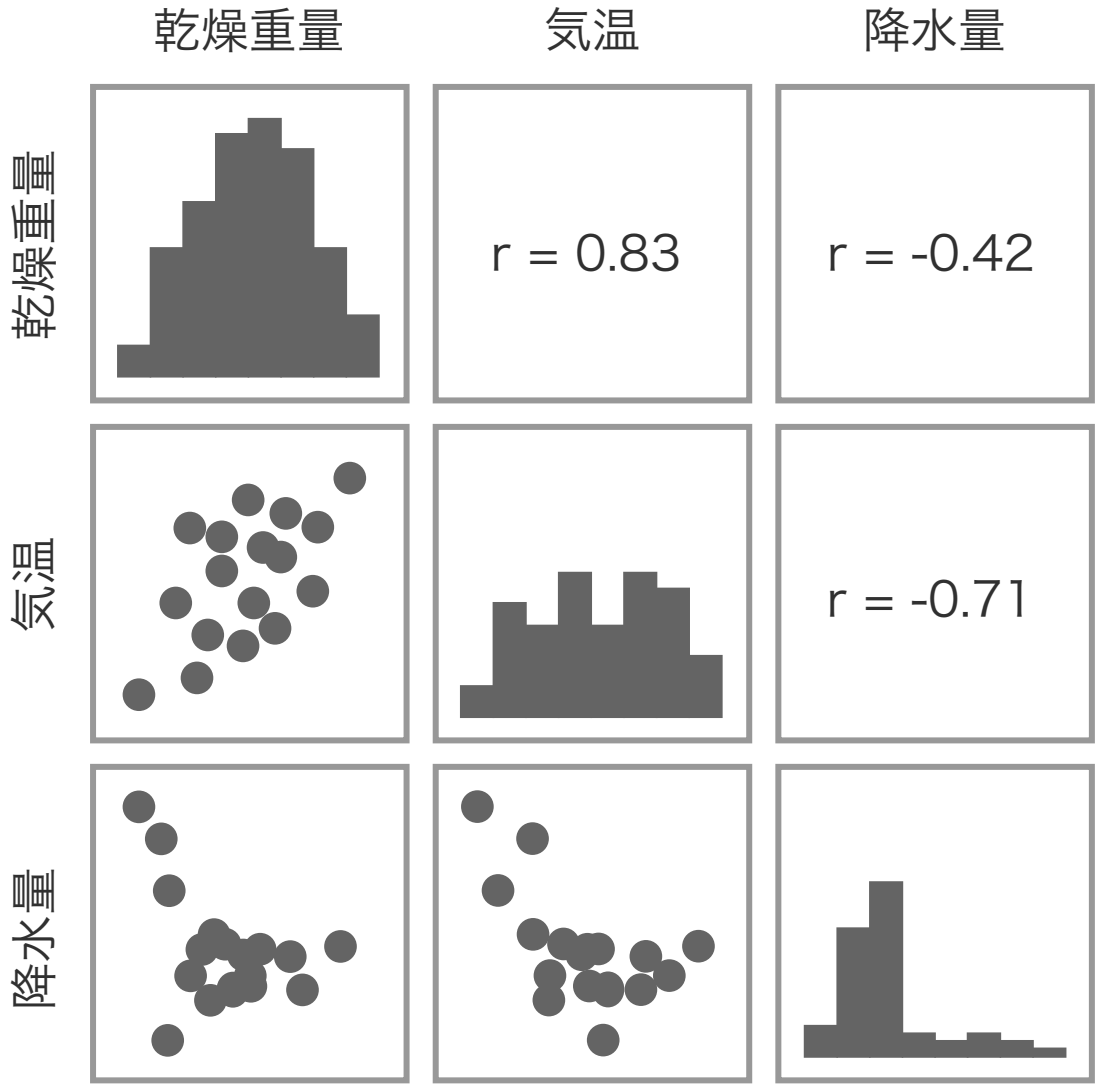
乾燥重量　施肥量　気温　日照



変数選択

重要な目的変数を選択するには、各変数の分布や変数間の相関等を調べる必要がある。そのため、モデルを構築する前に、例えば右図のようなペアプロットを描いて、変数の分布や特徴などを確認することが推奨される。

各変数の分布について、統計学における回帰分析では、変数が正規分布に従うことが必要とされる。これに対して、機械学習の分野では、説明変数および目的変数が正規分布に従う必要はないとされる。これは、機械学習において、第１種のエラーおよび第２種のエラーを評価する必要がないため、エラーを調べるための分布を仮定しなくてもよいからである。



変数選択

機械学習モデルが過度に複雑すぎる（パラメーター数が多すぎる）と、モデルが訓練データに過度に適合してしまい、汎用性を失うことがある。このことを過学習という。過学習を抑えるためには、データを増やしたり、パラメーターの数を減らしたりする必要がある。データが増やせない場合に、重要な特徴量だけを選択してモデルを構築することが、予測性能を高くする上で非常に役立つ。

- フィルタ法
- ラッパー法
- 組み込み法

変数選択

機械学習モデルが過度に複雑すぎる（パラメーター数が多すぎる）と、モデルが訓練データに過度に適合してしまい、汎用性を失うことがある。このことを過学習という。過学習を抑えるためには、データを増やしたり、パラメーターの数を減らしたりする必要がある。データが増やせない場合に、重要な特徴量だけを選択してモデルを構築することが、予測性能を高くする上で非常に役立つ。

- **フィルタ法**
- ラッパー法
- 組み込み法

特徴量と正解ラベルの相関や交互情報量等を計算して、閾値よりも小さければ削除する方法。

- 相関係数
- 交互情報量
- カイ二乗検定
- フィッシャースコア
- 回帰分析
- 分散
- 分散分析

変数選択

機械学習モデルが過度に複雑すぎる（パラメーター数が多すぎる）と、モデルが訓練データに過度に適合してしまい、汎用性を失うことがある。このことを過学習という。過学習を抑えるためには、データを増やしたり、パラメーターの数を減らしたりする必要がある。データが増やせない場合に、重要な特徴量だけを選択してモデルを構築することが、予測性能を高くする上で非常に役立つ。

- フィルタ法
- **ラッパー法**
- 組み込み法

特徴量の一部だけを使用してモデルを構築し評価する。これを繰り返すことで最適な特徴量を選択する方法。

- 逐次前進法

  特徴量が1つだけモデルの作成と評価を行い、最適な特徴量を決める。次に、選択された特徴量に加えて、もう1つの特徴量を加えたモデルの作成と評価を行う。この操作を、予測性能が改善されなくなるまで繰り返す。
- 逐次後退法

  特徴量をすべて含むモデルから特徴量を1つだけ除去したモデルに対して評価を行い、最も不要な特徴量を取り除く。次に、そのモデルからさらにもう1つの特徴量を除去したモデルに対して評価を行い、最も不要な特徴量を取り除く。この操作を、予測性能が改善されなくなるまで繰り返す。

変数選択

機械学習モデルが過度に複雑すぎる（パラメーター数が多すぎる）と、モデルが訓練データに過度に適合してしまい、汎用性を失うことがある。このことを過学習という。過学習を抑えるためには、データを増やしたり、パラメーターの数を減らしたりする必要がある。データが増やせない場合に、重要な特徴量だけを選択してモデルを構築することが、予測性能を高くする上で非常に役立つ。

- フィルタ法
- ラッパー法
- **組み込み法**

学習アルゴリムズに組み込まれている特徴量選択法。決定木やL1 正則化などが組み込み法である。

- 決定木
- ランダムフォレスト
- 正則化
  - LASSO
  - Elastic Net

モデル構築

乾燥重量　～　施肥量　気温　地温　日照　降水量　湿度　・・・

変数選択

施肥量　気温　日照

モデル構築

$$y = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_3$$

▲ 乾燥重量　▲ 施肥量　▲ 気温　▲ 日照

回帰分析

回帰分析

変数選択

スパース推定

様々な回帰分析

モデル構築

乾燥重量　～　施肥量　気温　地温　日照　降水量　湿度　・・・

変数選択

施肥量　気温　日照

モデル構築

正則化

$$y = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_3$$

▲ 乾燥重量　▲ 施肥量　▲ 気温　▲ 日照

L1 正則化

回帰分析

$$loss(\boldsymbol{w}) = \frac{1}{2}\sum_{i=1}^{n}(y_i - \boldsymbol{w}^t x_i)^2$$

特徴量がすべてゼロあるいは共相関の特徴量でない限り、ゼロにならない。

回帰分析（L1 正則化）

$$loss(\boldsymbol{w}) = \frac{1}{2}\sum_{i=1}^{n}(y_i - \boldsymbol{w}^t x_i)^2 + \lambda\sum_{j=1}^{m}|w_j|$$

すべての特徴量の係数 w がゼロでないときに、loss(**w**) が最小値をとる。

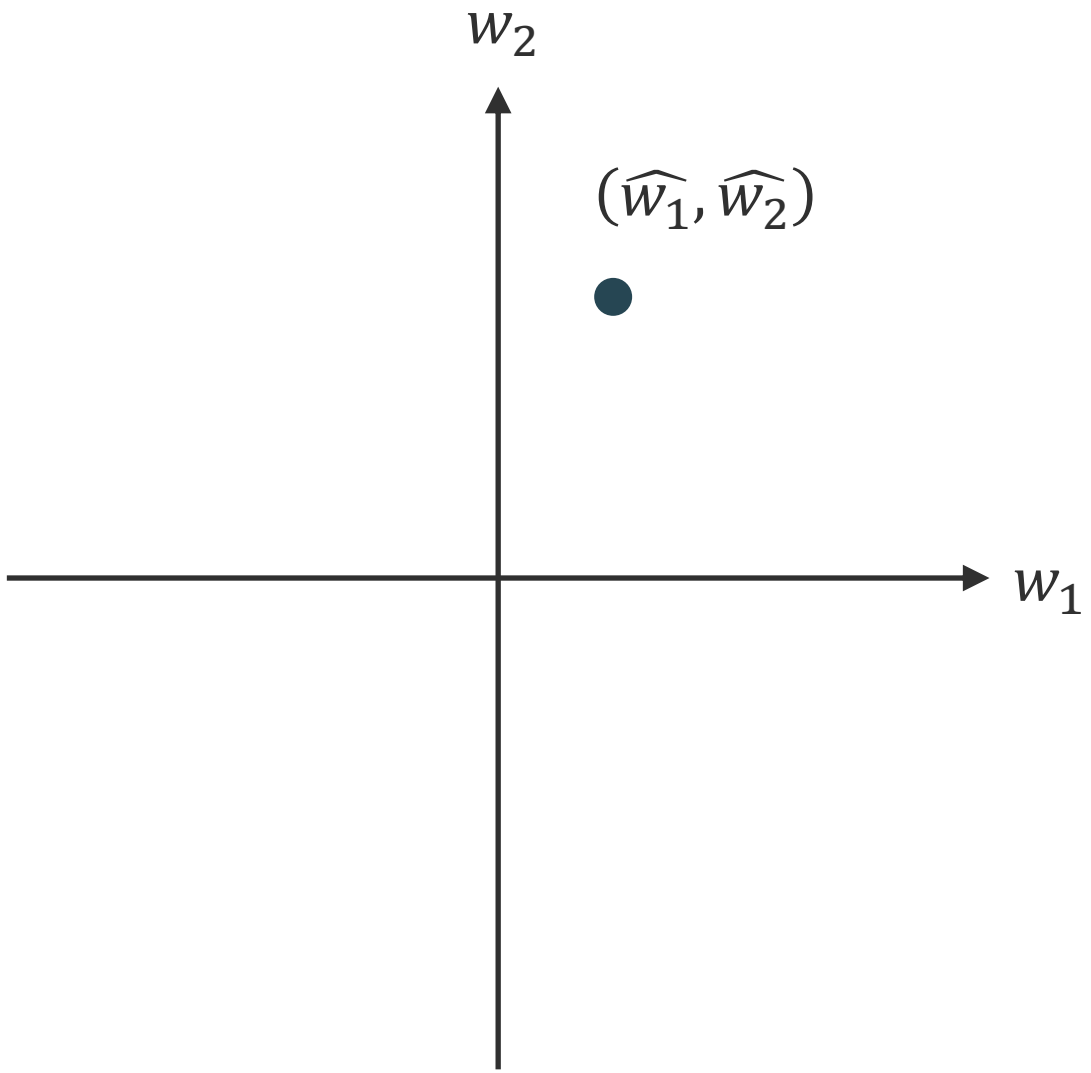
すべての w がゼロのときに、loss(**w**) 最小値をとる。

最小二乗解

L1 正則化（LASSO）仕組みを説明するために、説明変数が 2 つの回帰モデルを考える。このモデルにおいて、最小二乗法により推定された最適解を $\widehat{w_1}$ と $\widehat{w_2}$ とする。この最適解の組み合わせをパラメータ空間上にプロットすると右図のようになる。

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

$$loss(\boldsymbol{w}) = \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{w}^t \boldsymbol{x}\|_2^2$$



L1 正則化

損失関数に罰則項を加える。

$$loss(\boldsymbol{w}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{w}^t\boldsymbol{x}\|_2^2 + \lambda\|\boldsymbol{w}\|_1^1$$

制約条件の書き方を書き換えると次のようになる。

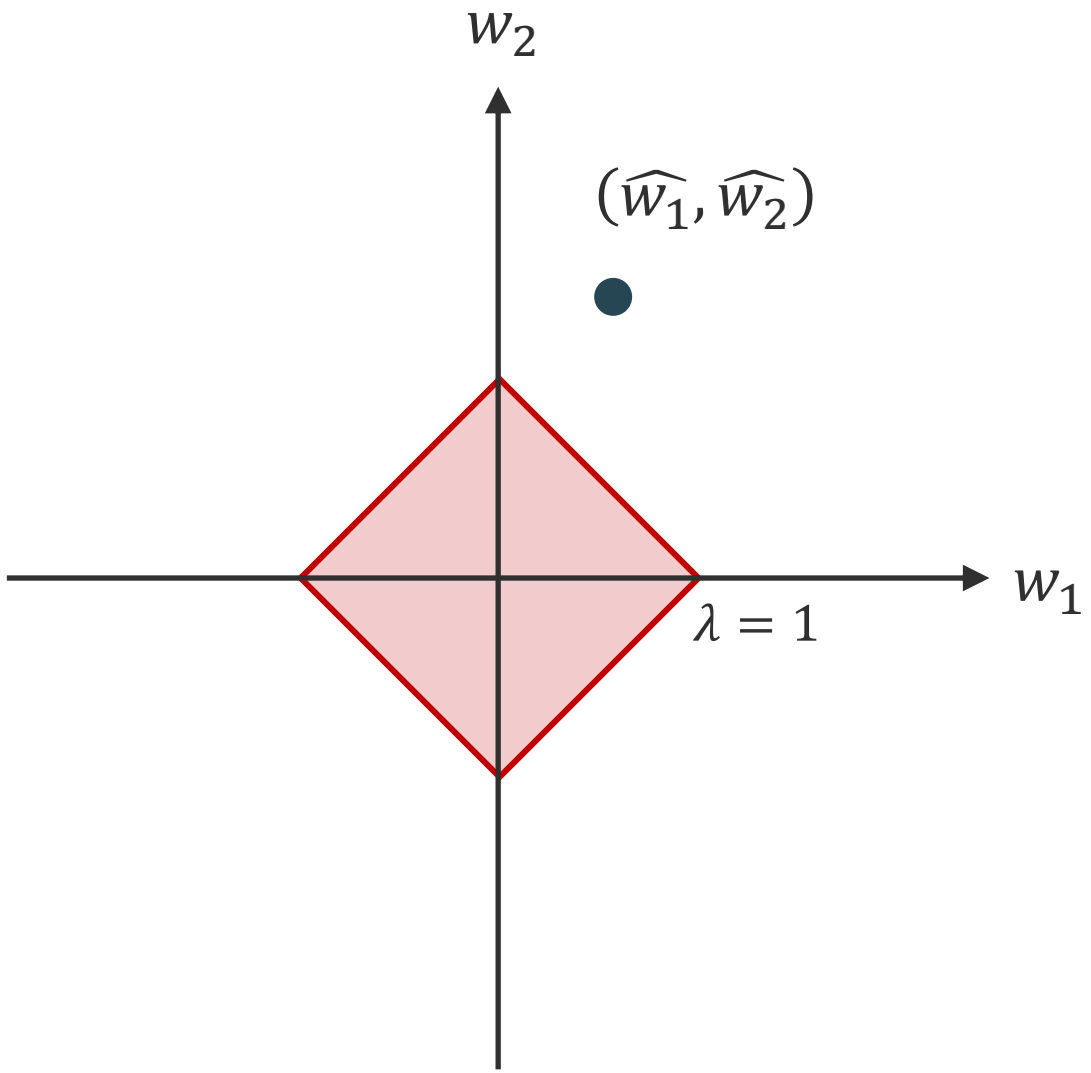
$$loss(\boldsymbol{w}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{w}^t\boldsymbol{x}\|_2^2$$

$$\text{subject to} \quad \|\boldsymbol{w}\|_1^1 \leq \lambda$$

制約条件（罰則項）を展開すると、

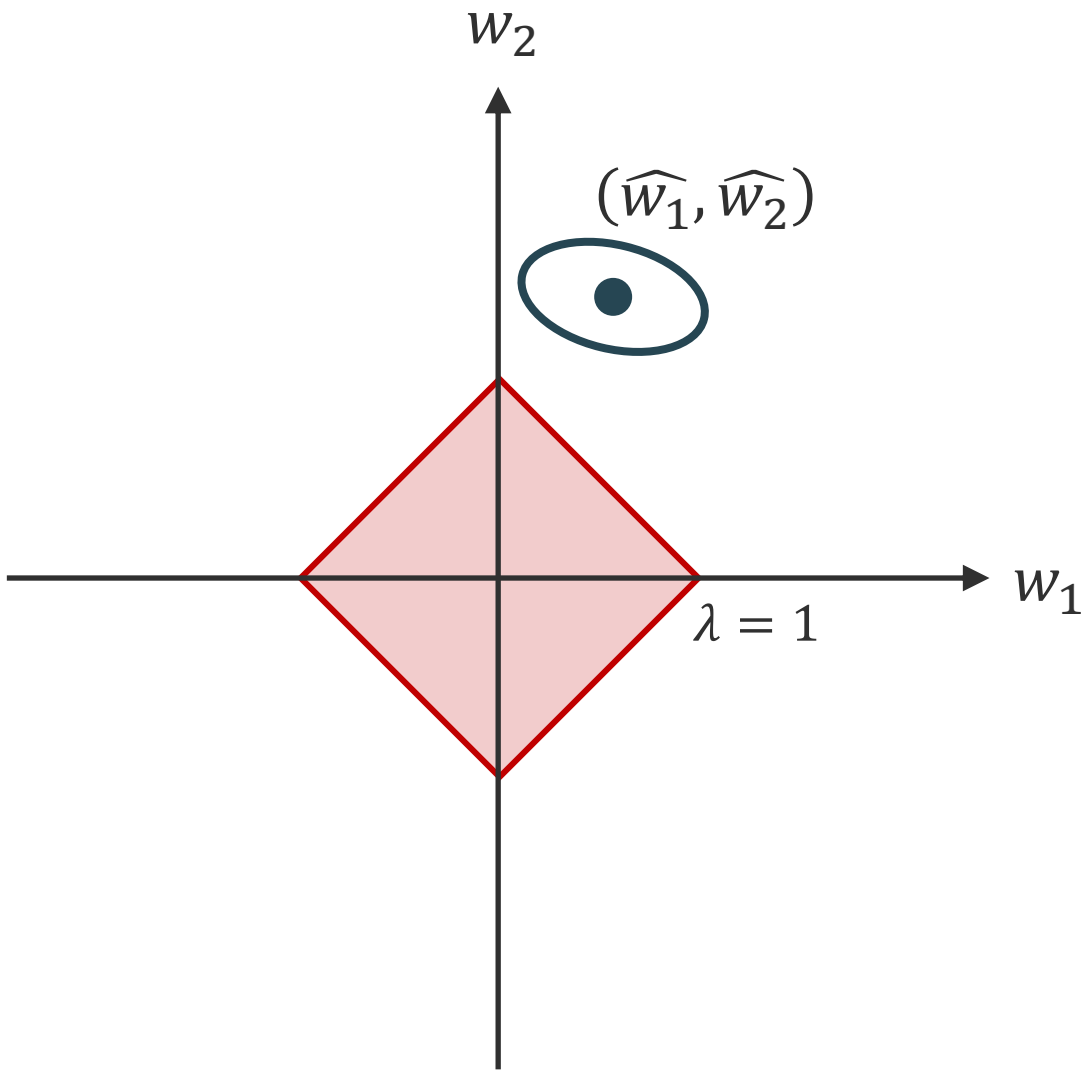
$$\|\boldsymbol{w}\|_1^1 = |w_1| + |w_2| \leq \lambda$$

つまり、解となる $(\widehat{w_1}, \widehat{w_2})$ の組み合わせは、右図の赤枠またはその内部に存在しなければならない。



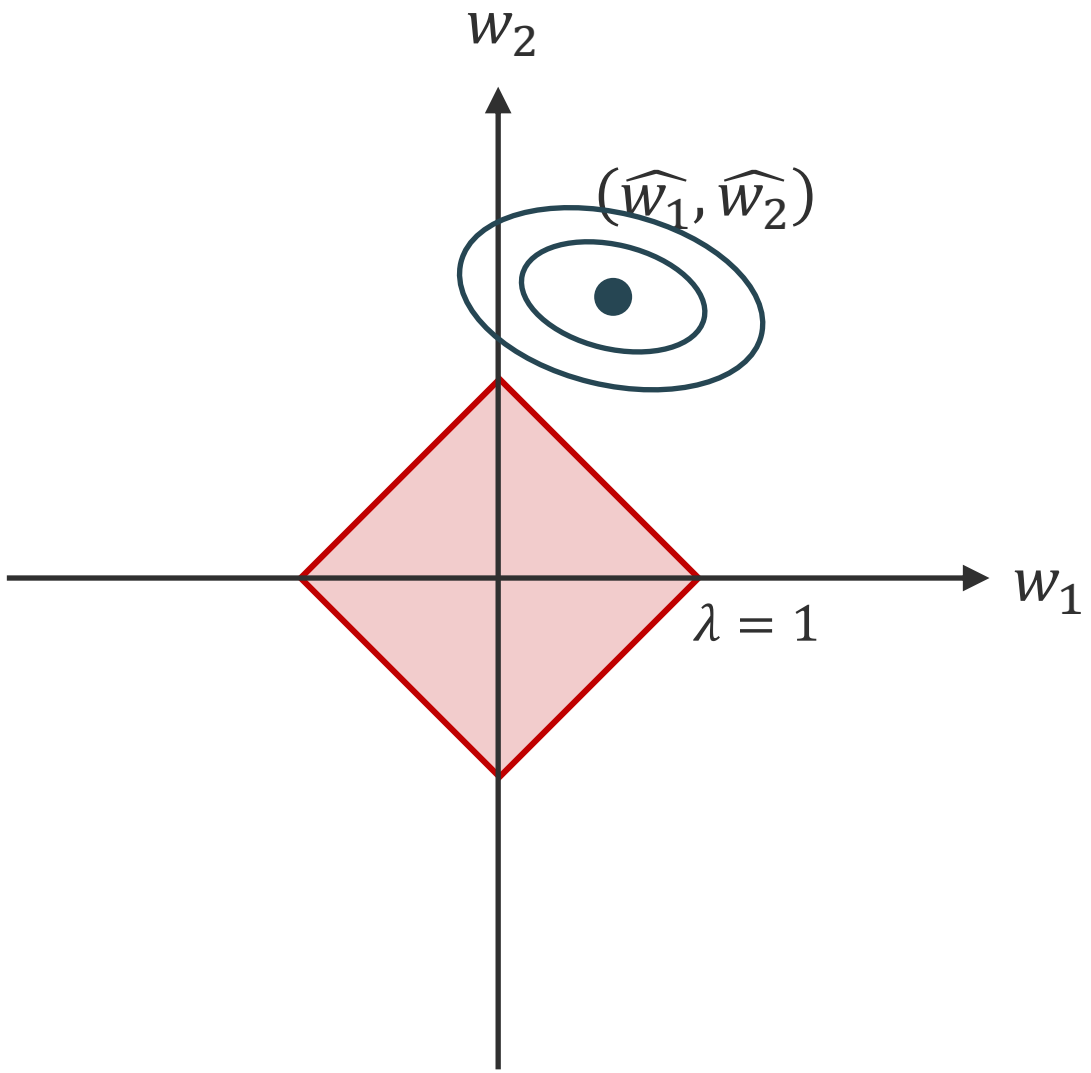
L1 正則化

パラメータの組み合わせが、赤枠以内に存在するようにするためには、残差二乗和を少し大きく許容する必要がある。そこで、残差二乗和を少しだけ大きく許容する。このとき、パラメータ $(\widehat{w_1}, \widehat{w_2})$ の組み合わせ複数存在し、例えば右図のように、最適解の周りに楕円状で分布している。



L1 正則化

パラメータの組み合わせが、赤枠以内に存在するようにするためには、残差二乗和を少し大きく許容する必要がある。そこで、残差二乗和を少しだけ大きく許容する。このとき、パラメータ $(\widehat{w_1}, \widehat{w_2})$ の組み合わせ複数存在し、例えば右図のように、最適解の周りに楕円状で分布している。



L1 正則化

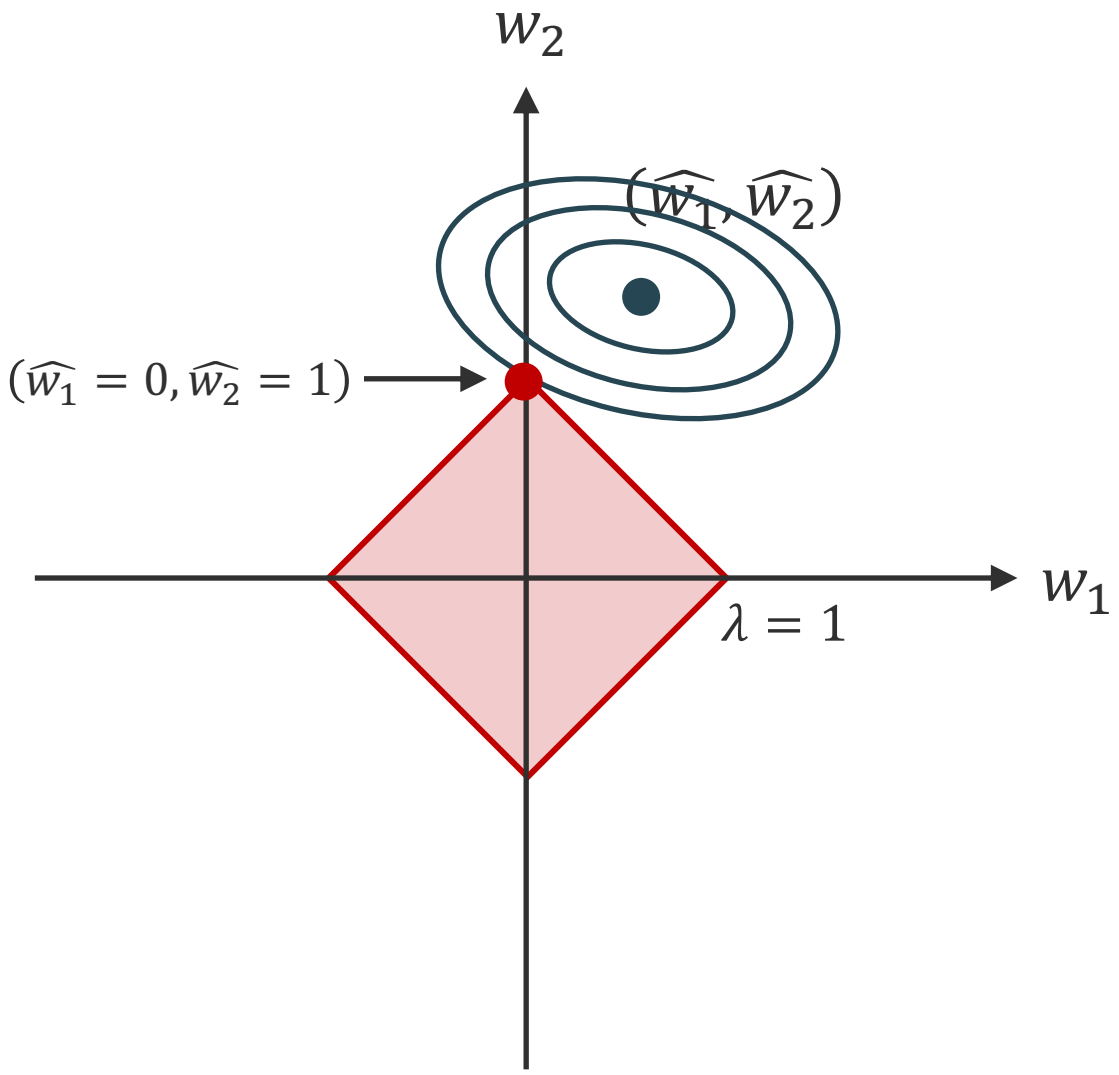
パラメータの組み合わせが、赤枠以内に存在するようにするためには、残差二乗和を少し大きく許容する必要がある。そこで、残差二乗和を少しだけ大きく許容する。このとき、パラメータ $(\widehat{w_1}, \widehat{w_2})$ の組み合わせ複数存在し、例えば右図のように、最適解の周りに楕円状で分布している。

残差二乗和の許容範囲を徐々に大きくすると、制約条件を満たす点 $(\widehat{w_1} = 0, \widehat{w_2} = 1)$ が出現する。このとき、次のようなモデルが推定される。

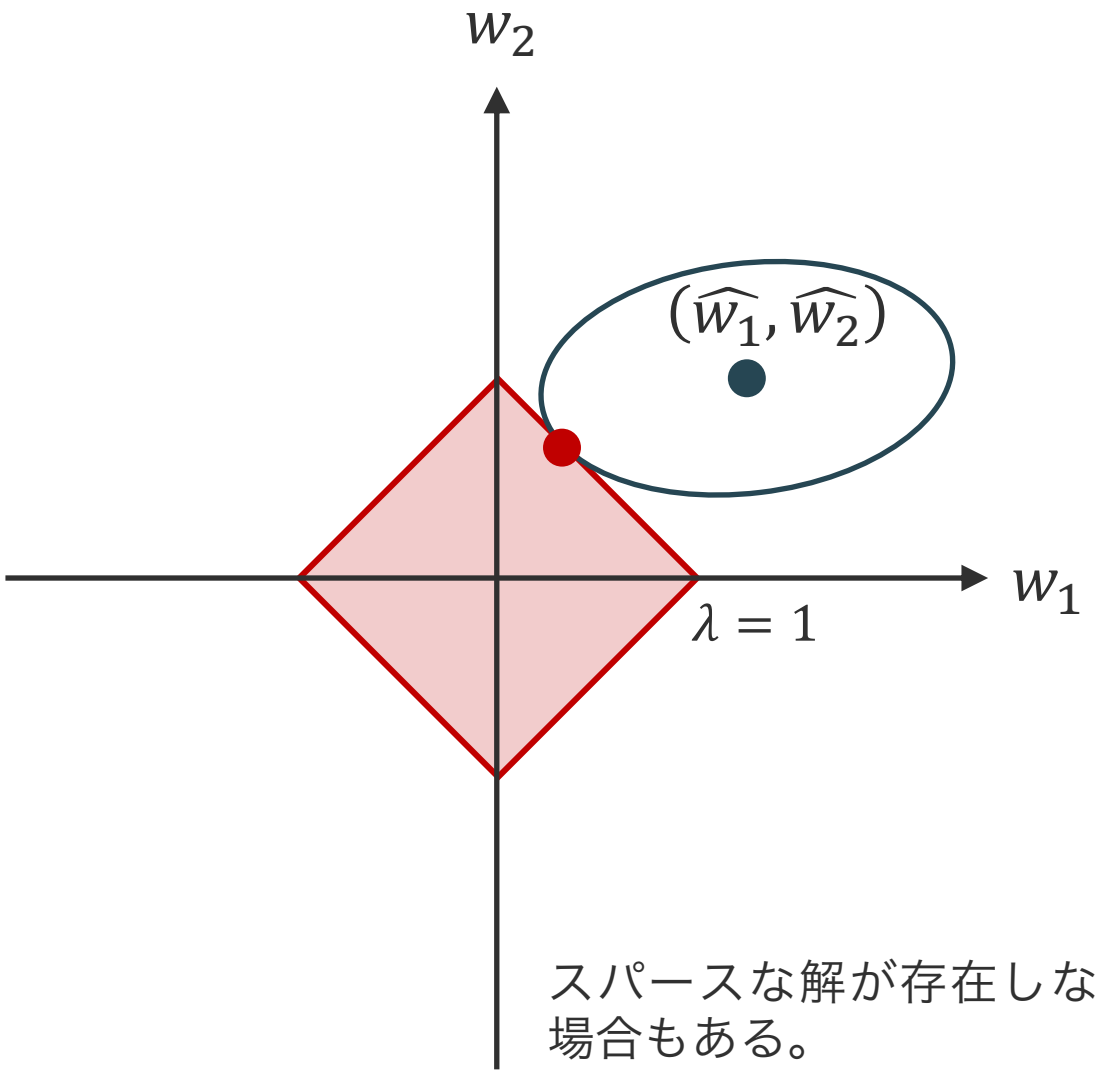
$$y = w_0 + w_1 x_1 + w_2 x_2$$

↓ L1 正則化

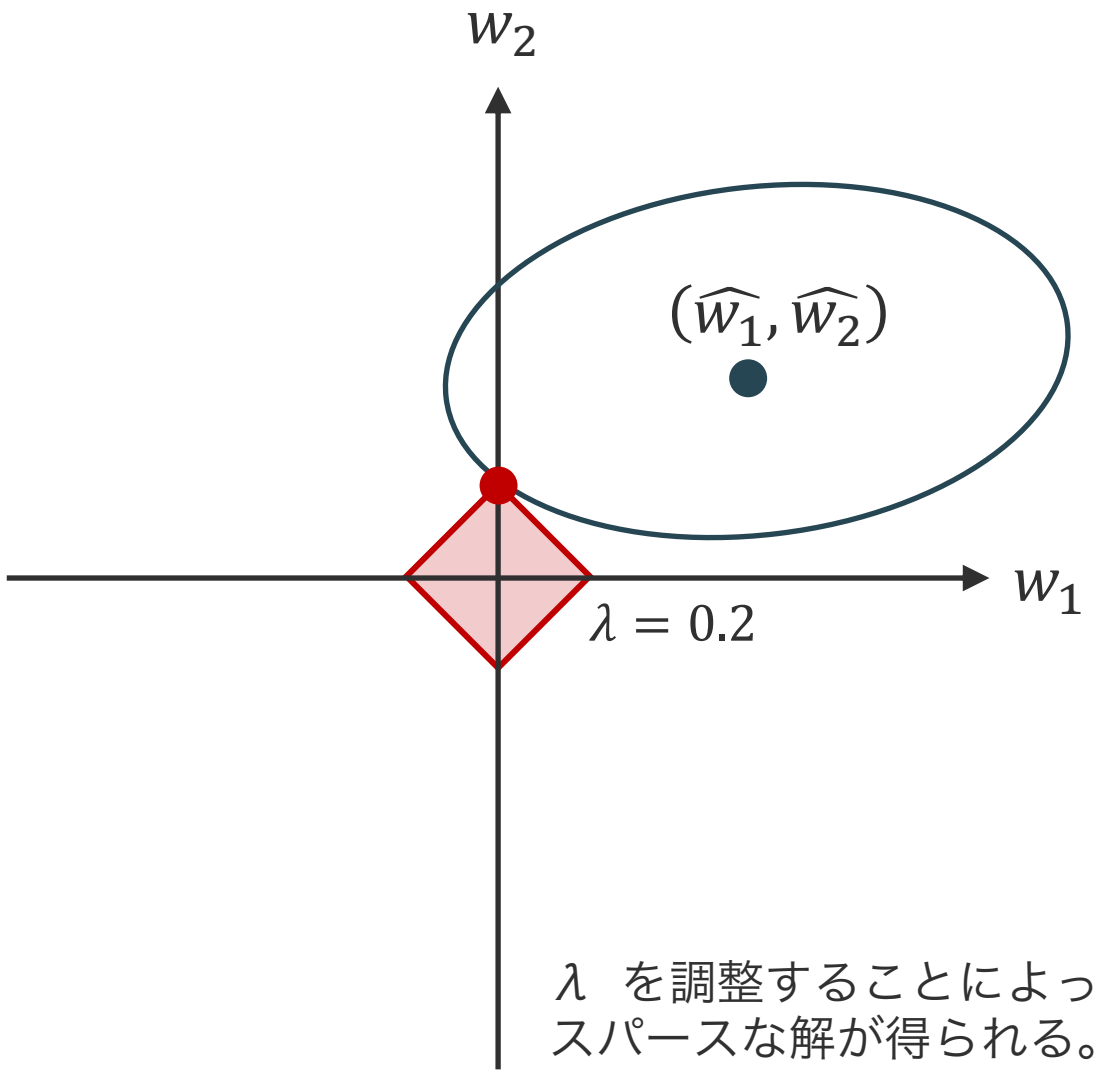
$$y = w_0 + x_2$$



L1 正則化



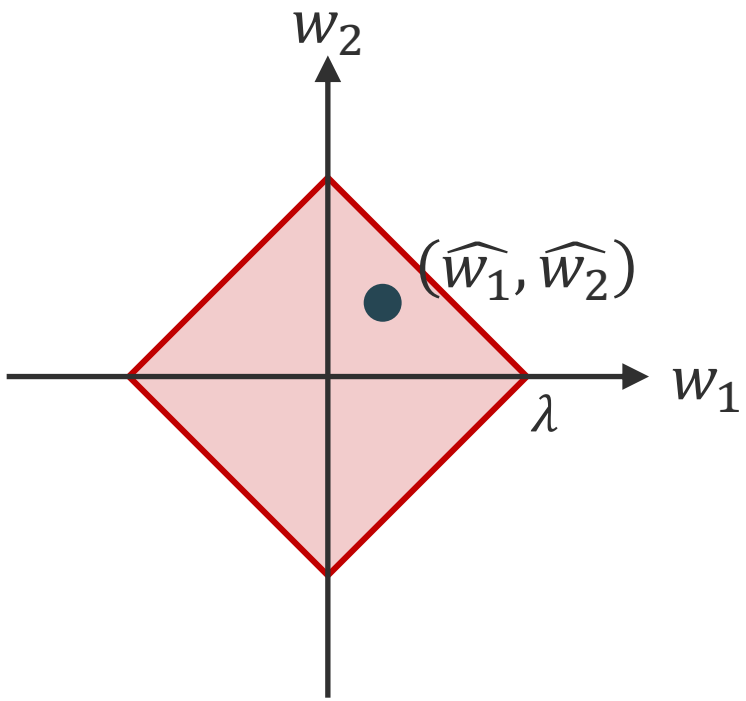
スパースな解が存在しない
場合もある。

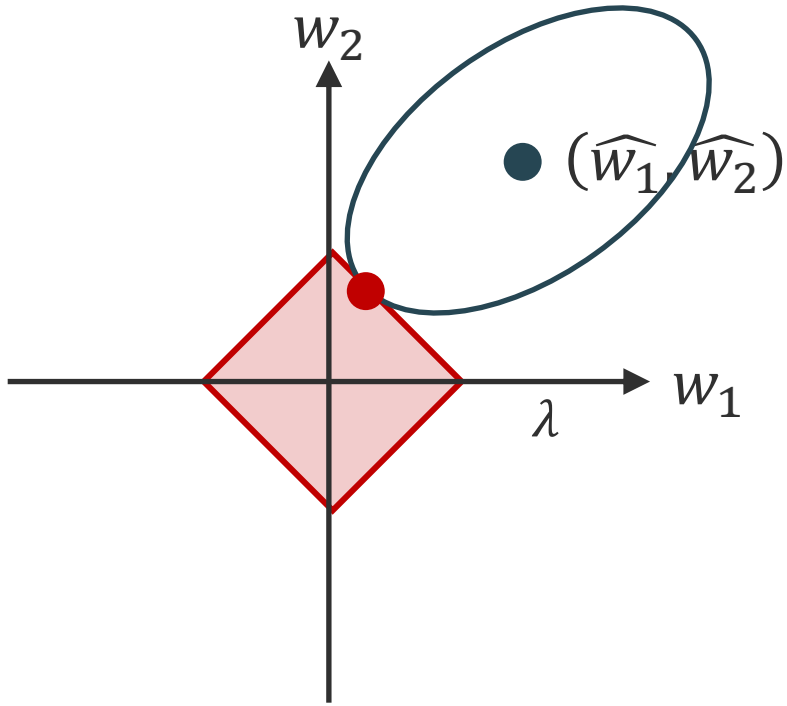


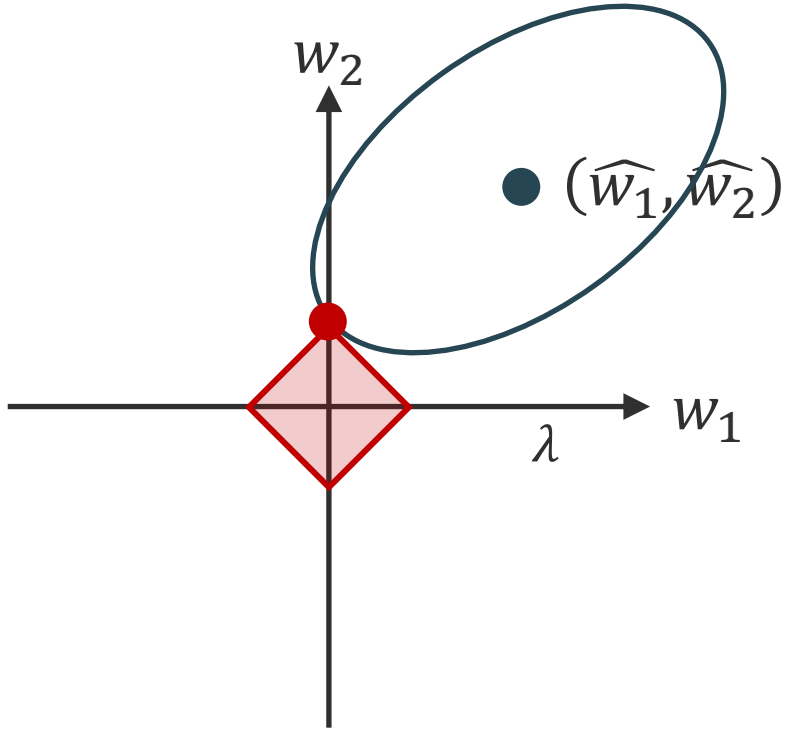
λ を調整することによって
スパースな解が得られる。

L1 正則化

スパースな解が存在しない場合もある。







正則化

L1 正則化（LASSO）

$$loss(\boldsymbol{w}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{w}^t\boldsymbol{x}\|_2^2 + \lambda\|\boldsymbol{w}\|_1^1$$

罰則項

罰則の強さ

L2 正則化（Ridge）

$$loss(\boldsymbol{w}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{w}^t\boldsymbol{x}\|_2^2 + \lambda\|\boldsymbol{w}\|_2^2$$

罰則項

罰則の強さ

Elastic Net

$$loss(\boldsymbol{w}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{w}^t\boldsymbol{x}\|_2^2 + \lambda(\alpha\|\boldsymbol{w}\|_1^1 + (1-\alpha)\|\boldsymbol{w}\|_2^2)$$

罰則項

L1 正則化と L2 正則化の割合

L2 正則化

損失関数に罰則項を加える。

$$loss(\boldsymbol{w}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{w}^t\boldsymbol{x}\|_2^2 + \lambda\|\boldsymbol{w}\|_2^2$$

制約条件の書き方を書き換えると次のようになる。

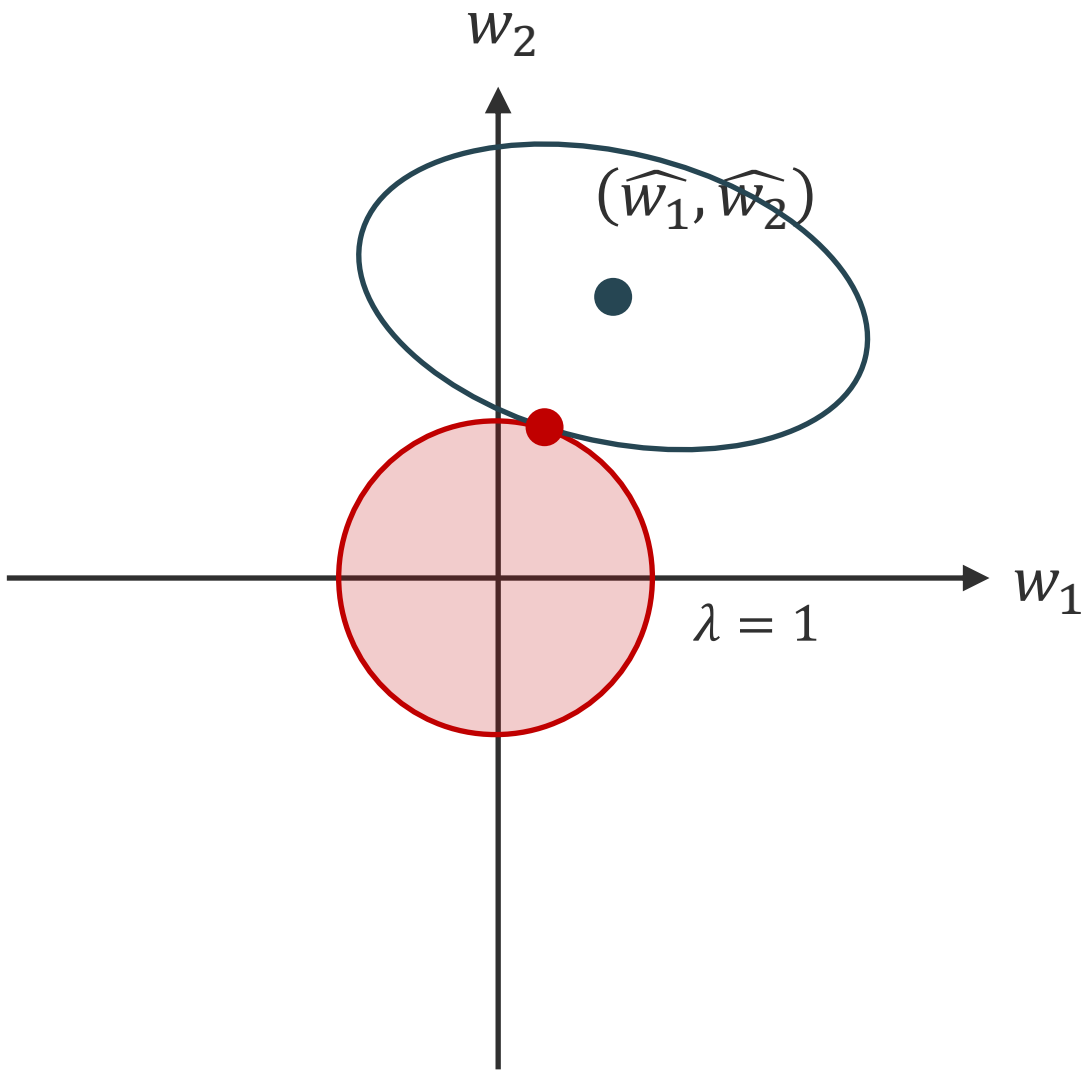
$$loss(\boldsymbol{w}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{w}^t\boldsymbol{x}\|_2^2$$

$$\text{subject to} \quad \|\boldsymbol{w}\|_2^2 \leq \lambda$$

制約条件（罰則項）を展開すると、

$$\|\boldsymbol{w}\|_2^2 = w_1^2 + w_2^2 \leq \lambda$$

右図のように L2 正則化は、L1 正則化と異なり、スパースとなる解を得るのが非常に難しい。

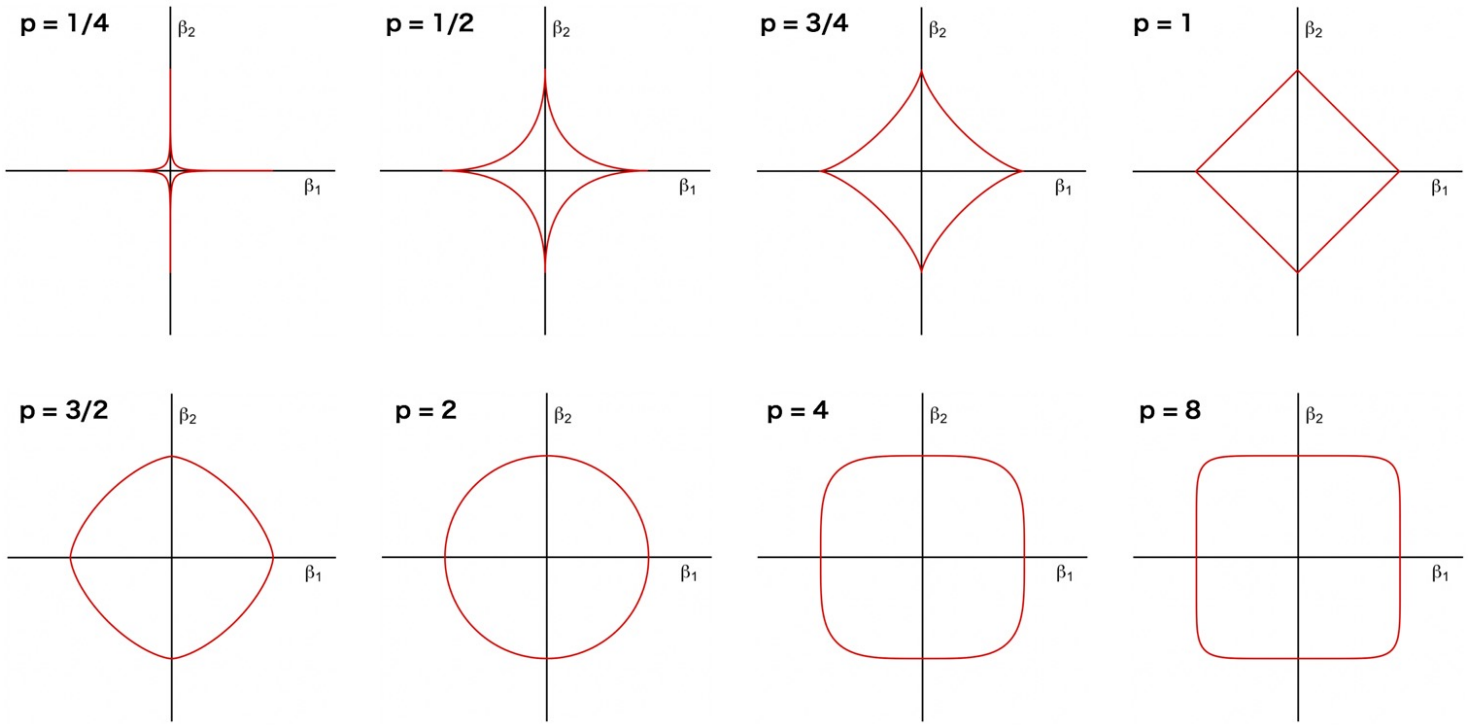


Bridge

回帰分析において、罰則項として p 次のノルムを加えてパラメータ推定を行う場合を Bridge とよぶ。L1 正則化（LASSO）および L2 正則化（Ridge）は、Bridge 回帰の特別な場合として捉えることができる。

$$loss(\boldsymbol{w}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{w}^t\boldsymbol{x}\|_2^2 + \lambda\|\boldsymbol{w}\|_p^p$$

- p > 1 のとき、値の小さいパラメータをさらに小さくするよりも、値の大きいパラメータを小さくした方が罰則が小さいので、個々のパラメータがゼロになりにくい。
- 0 < p < 1 のとき、どのパラメータを小さくしても、罰則の減少は同じなので、個々のパラメータがゼロになりやすくなる ただし、罰則項が非凸関数になるため計算が困難である。
- スパース推定として p = 1 が一般的に使われている

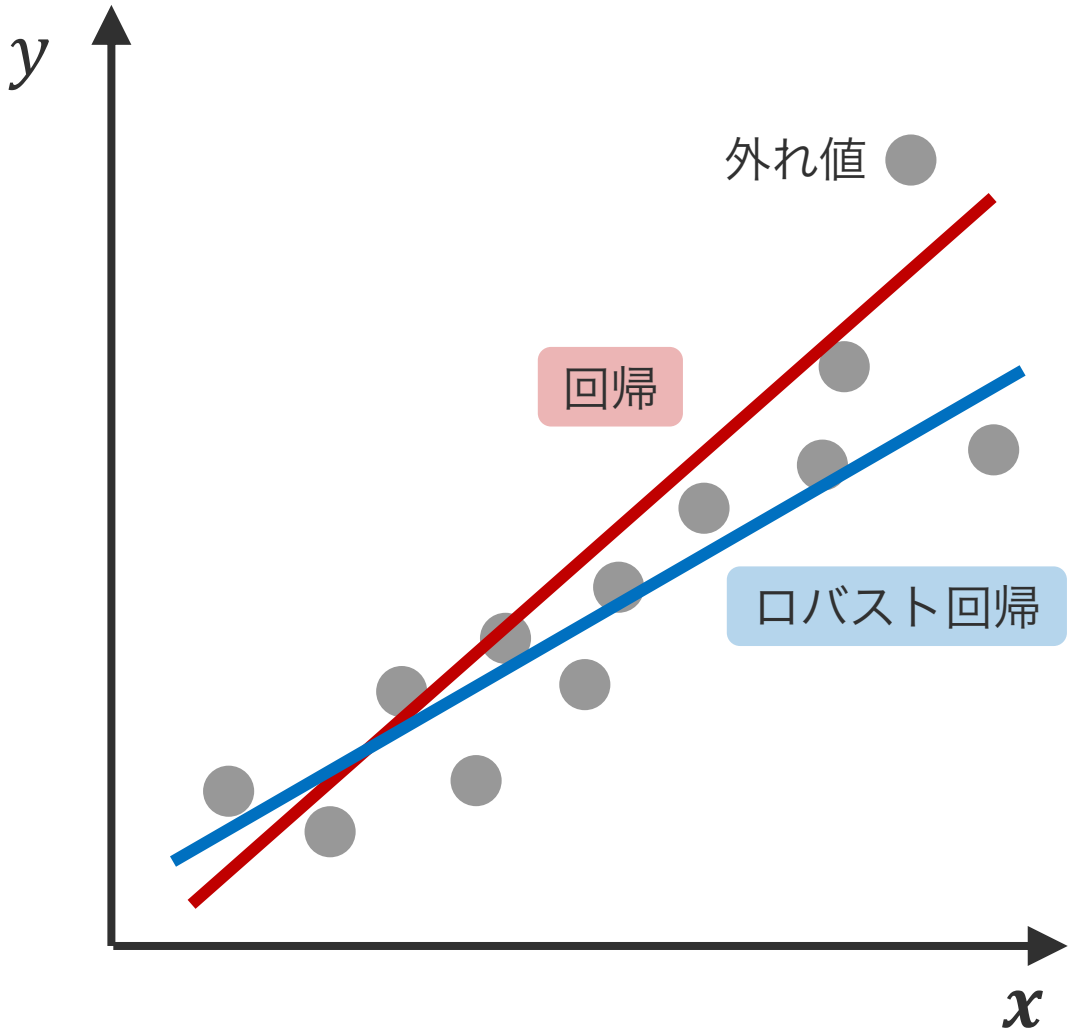


回帰分析

- 回帰分析
- 変数選択
- スパース推定
- 
  様々な回帰分析

ロバスト回帰

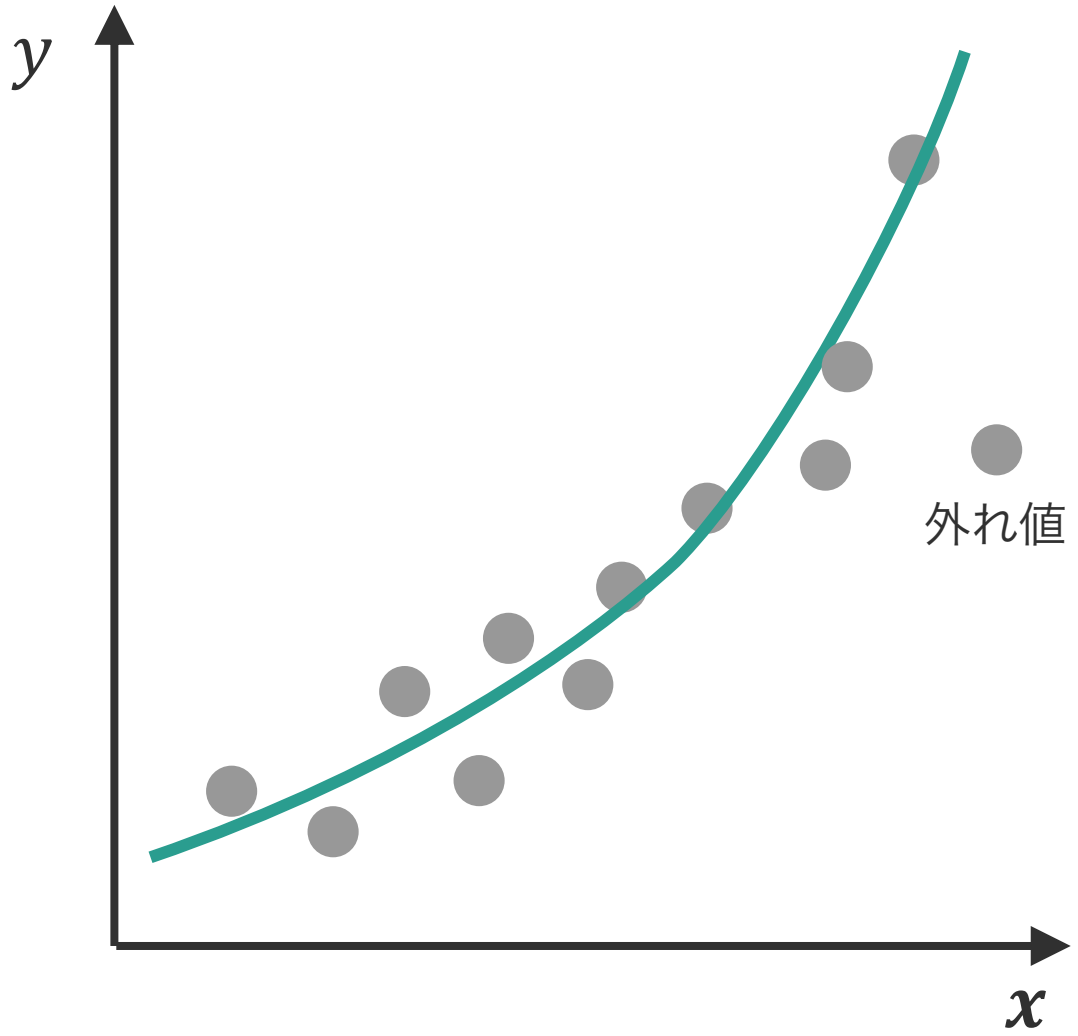
各サンプルに重みをかけて回帰直線を求める。回帰直線から遠く離れているサンプルを無視して回帰直線を求める。データに外れ値が含まれている場合に有効な方法である。



ロバスト回帰

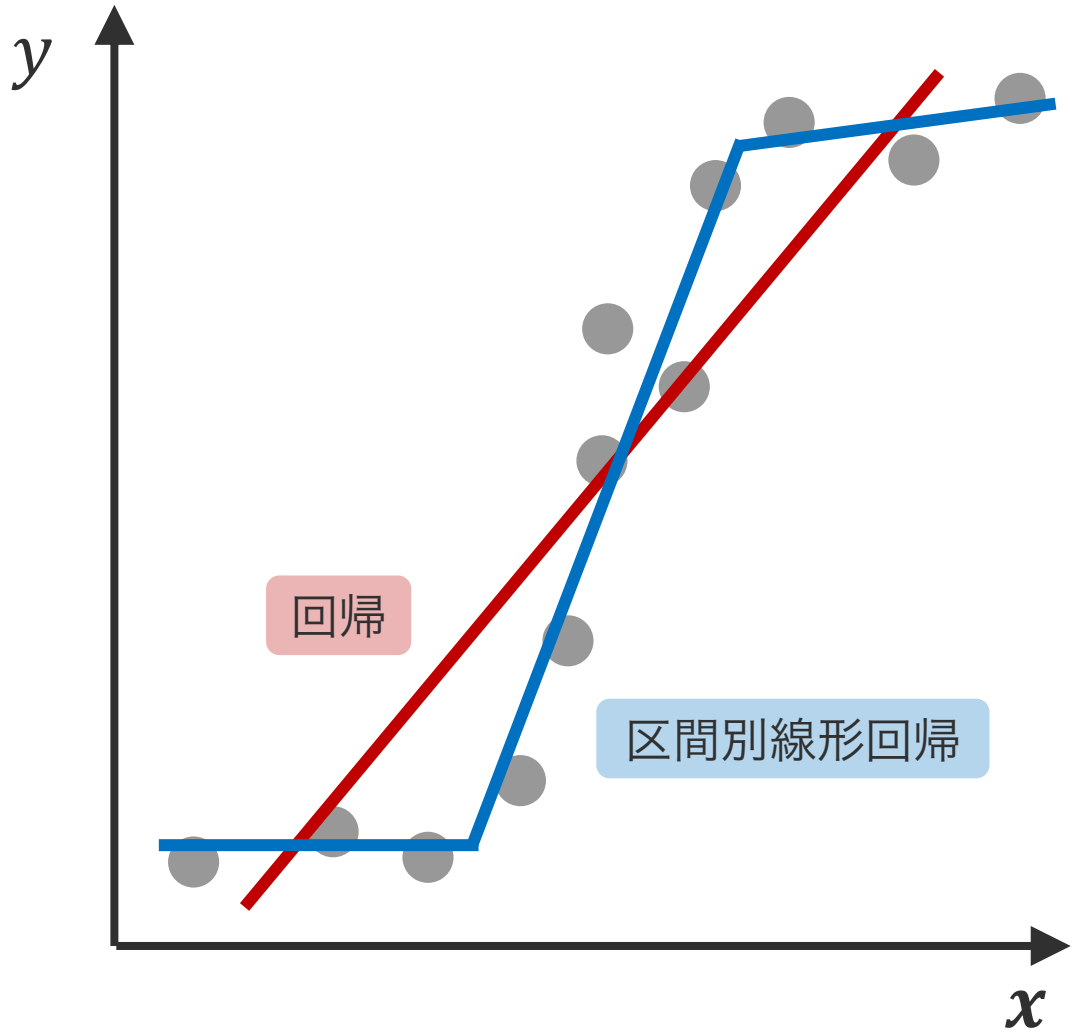
各サンプルに重みをかけて回帰直線を求める。回帰直線から遠く離れているサンプルを無視して回帰直線を求める。データに外れ値が含まれている場合に有効な方法である。

モデルを構築する上でデータをしっかり吟味する必要がある。外れ値を、単に他の点から外れているからと視覚的に決めつけるのではなく、しっかりデータの意味を考えて外れ値かどうかを決める。



区間別線形回帰

データをいくつかの区間に分けて、それぞれの区間において線形回帰を行う。



区間別線形回帰

データをいくつかの区間に分けて、それぞれの区間において線形回帰を行う。線形回帰で解けない問題を無理に線形的に解いている可能性があるので、目的と観測事象を確認し、適切な方法を使用すべきである。