機械学習エンジニアコース
Sprint

ー 線形回帰 ー

DIVE INTO CODE

今回のモチベーション

目的はなにか

1. 統計モデルを知る
   スクラッチを通して線形回帰を理解する

2. クラスを作成する
   オブジェクト指向を意識した実装に慣れる

3. 数式から機械学習の計算過程を知る
   数式をコードに落とし込めるように

このスライドは?

ここでは、線形回帰の基本的な知識を学びましょう

線形回帰とはなにか

　機械学習においては、xからyへの写像を学習することを、回帰問題を解くこととみなす(1)。

　回帰とは、英語の「regression」の翻訳である。この語源は、19世紀においてフランシス・ゴルトンが生物データを観察したところ、背の高い祖先の子孫の身長が必ずしも遺伝せず、平均値に戻っていく、つまり「後退(=regression)」する傾向を発見したことに由来する。ゴルトンはこの事象を分析するために「線形回帰(linear regression)」を発明した(2)。

　線形(3)回帰とは、目的変数(y)が、説明変数(x)にどれほど依存しているかを表すモデル(近似関数)のことである。
説明変数が一つなら、線形単回帰モデルを用い、説明変数が二つ以上あれば、線形重回帰モデルを用いる。

(1) yはこの場合、連続変数である。
(2) ウィキペディア(Wikipedia)より https://ja.wikipedia.org/wiki/%E5%9B%9E%E5%B8%B0%E5%88%86%E6%9E%90
(3) 線形が満たす条件はこちら。https://mathtrain.jp/linear

線形回帰というテーマについて

線形回帰は解析的に扱いやすく(1)、
より洗練されたモデルの基礎をなすゆえに、大切。

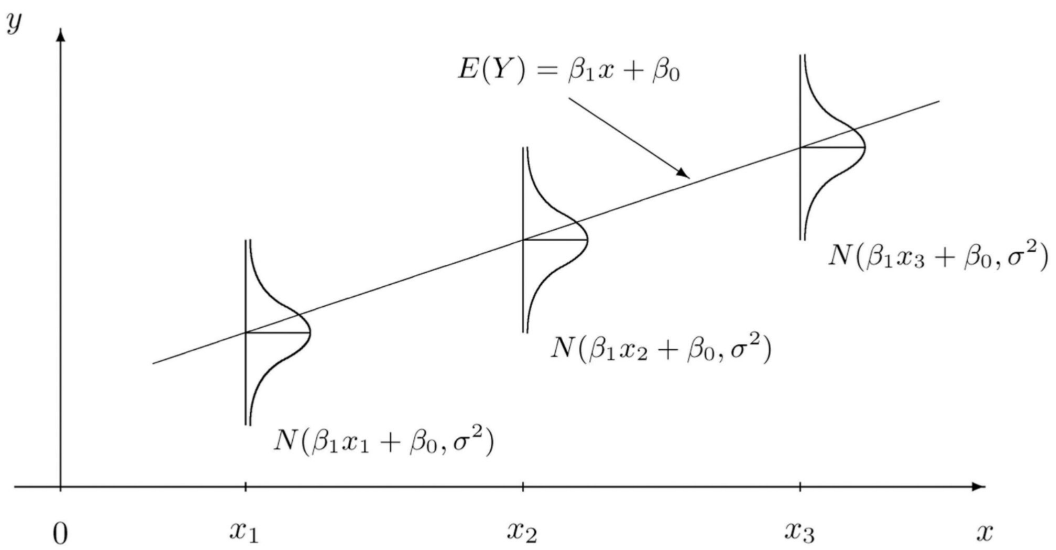
(1)ある変数の変化に対して、解がどういう変化を示すかがわかりやすい、という意味で。

与えられた条件は何か

線形回帰においては以下が仮定されている。

①説明変数(x)は連続値 or 離散値で、目的変数(y)は連続値である。
②予測値(y)は、正規分布(平均 μ 、標準偏差 σ)に従う。(1)

(1) i番目のデータ点 xi において、予測値はこの正規分布の平均値 μ i となる。



②のイメージ

この課題の対象者

① scikit-learnの線形回帰モデルを用いて、学習、推定するコードが書ける方

②勾配降下法のアルゴリズムを少し知っている方(1)

(1) week2授業課題2の富士下山問題

この後の流れ

線形回帰の問題設定を知る

①予測値を導く式(仮定関数)をたてる

②目的変数と予測値の誤差を求める

③この誤差を最小化する問題を設定し、式(目的関数)をたてる

④目的関数の最適解を探索的に求める(最急降下法)

⑤最適解に至るとき、仮定関数の最適なパラメータが求まる

この後の流れ

HousePrice data（week4_work2）

いまここにHousePriceデータセットがあるとしよう。
目的変数 y(SalePrice)に対し、ひとつの説明変数 x (GrLivArea)を選び、二変数間の関係 をプロットしてみよう。

線形関係が存在しそう

プロットされたデータの傾向をみると、なにやら直線が引けそうだ。

この後の流れ

変数 x と変数 y の間に線形関係が存在するとき、このような式で表すことができる。

$$\hat{y}_i = \beta_0 + \beta_1 x_i \qquad i = 1, 2, \ldots, n$$

この式を用いて直線を
パラメータβ1 は 直線の傾きを制御している。
つまり、変数 xi の1単位あたりの変化がどのくらい変数 yi の変化に対応しているかを示している。
パラメータβ0 は、xi = 0のときの y軸と交差する点を示す切片である。

この後の流れ

線形回帰の問題設定を知る

①予測値を導く式(仮定関数)をたてる

②目的変数と予測値の誤差を求める

③この誤差を最小化する問題を設定し、式(目的関数)をたてる

④目的関数の最適解を探索的に求める(最急降下法)

⑤最適解に至るとき、仮定関数の最適なパラメータが求まる

この後の流れ

手元にあるHousePriceデータセット(train_X)によく当てはまる直線があれば、未知のデータセット(test_X)を手に入れたとき、それに対応する y(つまり未知のSalePrice)を予測できそうだ。

直線はよく当てはまっているか?

説明変数のあるサンプル xi (3450)における、予測値 yi(100000)と、目的変数 yi (200000)の間に大きな誤差(error)がある。

この後の流れ

この誤差を評価し、最小化したい
全体としてどれほどの誤差があるかを評価するために、すべての誤差を足す。
この誤差の合計が小さいほど、直線はよく当てはまっていると言えるだろう。

この後の流れ

誤差を評価する式は以下のように書ける(この評価の方法を、最小二乗法という)。

※各点が直線の $\mathcal{L}(\beta_0, \beta_1) = \sum_{i=1}^{n} [y_i - (\beta_0 + \beta_1 x_i)]^2 \quad i = 1, 2, \ldots, n$ している。

DIVERは重回帰モデルで定式化しているがこれは単回帰モデルです。

この後の流れ

線形回帰の問題設定を知る

①予測値を導く式(仮定関数)をたてる

②目的変数と予測値の誤差を求める

③この誤差を最小化する問題を設定し、式(目的関数)をたてる

④目的関数の最適解を探索的に求める(最急降下法)

⑤最適解に至るとき、仮定関数の最適なパラメータが求まる

この後の流れ

評価指標は平均二乗誤差(MSE)とする

すべての誤差の平均をとると、先ほどの最小二乗法から、**平均二乗誤差(= 分散)**という指標に変わる。
線形回帰の評価指標として、この平均二乗誤差を用いる。

さらに、この評価指標を2で割ったものを線形回帰の**目的関数(Cost Function)**とする。
また、予測値を計算する直線の式を**仮定関数(Hypothesis)**とする。
※目的関数は後で偏微分するため、計算の便宜上、2で除算している。

線型結合

Hypothesis: $h_\theta(x) = \boxed{\theta_0 + \theta_1 x}$

DIVERでは
$\theta_0 x_0 + \theta_1 x_1$
$(x_0 = 1)$

特徴量方向の足し算

Parameters: θ_0, θ_1

サンプル(m個)方向の足し算

Cost Function: $J(\theta_0, \theta_1) = \frac{1}{2m} \boxed{\sum_{i=1}^{m}} \left(h_\theta(x^{(i)}) - y^{(i)}\right)^2$

Goal: $\underset{\theta_0, \theta_1}{\text{minimize}}\ J(\theta_0, \theta_1)$

この後の流れ

目的は $J(\theta_0, \theta_1)$ 値を最小化すること

目的関数の最小値を求めるような問題を一般に最適化問題(与えられた条件のもとで何らかの関数を最小化、もしくは最大化する問題)と呼ぶ。

Hypothesis: $h_\theta(x) = \theta_0 + \theta_1 x$

Parameters: θ_0, θ_1

Cost Function: $J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right)^2$

Goal: $\underset{\theta_0, \theta_1}{\text{minimize}}\ J(\theta_0, \theta_1)$

J(Θ₀,Θ₁)を最小化するようなパラメータΘ₀,Θ₁を求める

この後の流れ

線形回帰の問題設定を知る

①予測値を導く式(仮定関数)をたてる

②目的変数と予測値の誤差を求める

③この誤差を最小化する問題を設定し、式(目的関数)をたてる

④目的関数の最適解を探索的に求める(最急降下法)

⑤最適解に至るとき、仮定関数の最適なパラメータが求まる

この後の流れ

最適解は関数のどこにあるのか?

今回は最小値を求める問題なので、グラフでみると関数値が一番小さい点が最適解になる。

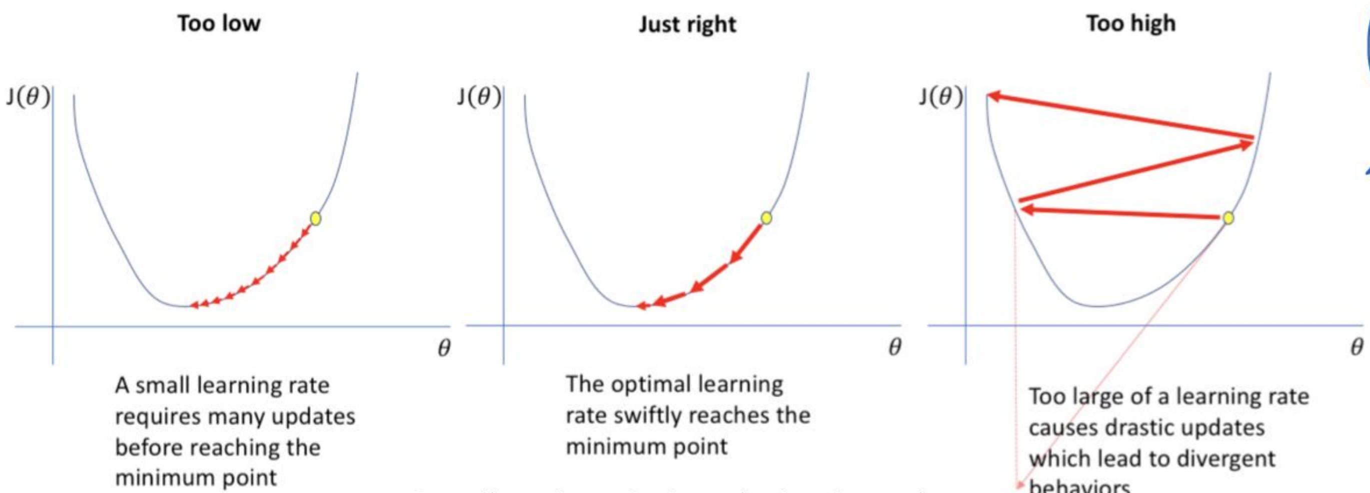
「最適化手法」としての最急降下法(1)の特徴は、学習データのすべての誤差を合計し、パラメーターを更新する。これは、いくつか種類のある勾配降下法の中でも、最もシンプルで古典的な手法である。

(1) 「分析手法」としての最急降下法とは区別する。
https://en.wikipedia.org/wiki/Method_of_steepest_descent#Extensions_and_generalizations

この後の流れ

最適解の探索

目的関数 J(θi) は、学習回数ごとに算出されるため、
どんな形のグラフになるかあらかじめ知ることができない。ここでの最適化問題では、
関数の最小値(J(θi) = 0):「**接線の傾き**が0になる」ような点
を探索的に求める。
(制約付き最小化問題の場合は、制約の中での最小値が最適解となる)

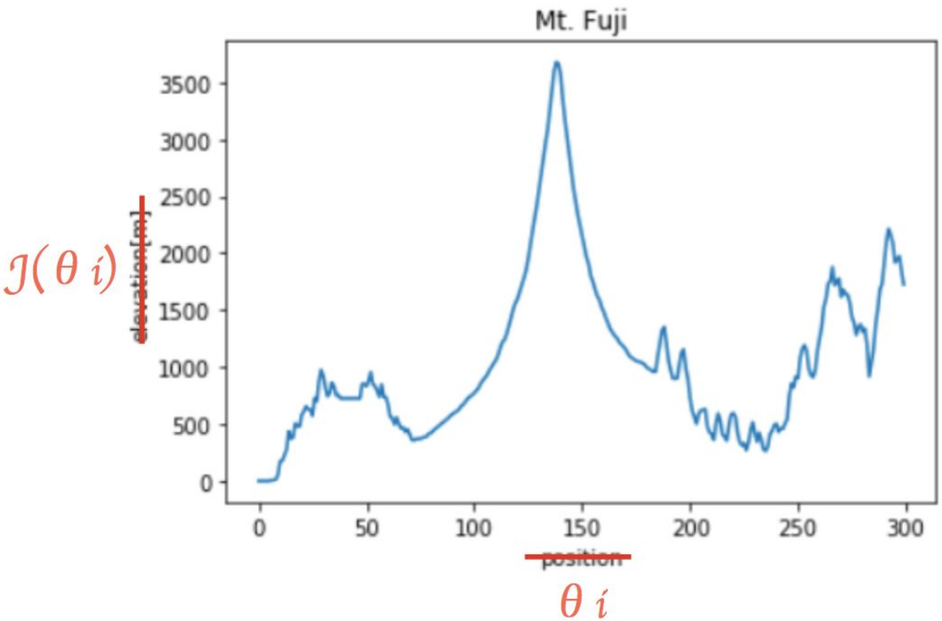


https://www.jeremyjordan.me/nn-learning-rate/

探索のためのステップ幅が大きいと
最適解にたどり着けない
orz

この後の流れ

最急降下法とは?



※ week2 授業課題2 富士下山問題 より

適当な初期点からスタートして、点を次のように更新する、反復的アルゴリズムの一種。

$$\theta_{i+1} := \theta_i - \alpha \frac{\partial J(\theta_i)}{\partial \theta_i}$$

$\frac{\partial J(\theta_i)}{\partial \theta_i}$ **:探索方向を表し、この方向に進む**

ことで i 回目の反復点より、i + 1 回目の反復点の方が解に近づくことを期待する。
α はスカラーで、探索方向にどれぐらい進むかを制御するステップ幅で、学習率(learning rate)とも呼ばれる。
最急降下法は、その名前にある通り、最小とする方向(勾配)を用いて、解を探索している。

この後の流れ

線形回帰の問題設定を知る

①予測値を導く式(仮定関数)をたてる

②目的変数と予測値の誤差を求める

③この誤差を最小化する問題を設定し、式(目的関数)をたてる

④目的関数の最適解を探索的に求める(最急降下法)

⑤最適解に至るとき、仮定関数の最適なパラメータが求まる

この後の流れ

仮定関数が最適なパラメータを得たら

変数 θ を最終更新した仮定関数を用いて、推定してみよう。
式の見た目は同じだけど、θ の値が未知でなくなった。

np.random.randで一様分布から
サンプリングした乱数（[0,1]の範囲）

学習前 $$h_\theta(x) = \boxed{\theta_0} + \boxed{\theta_1} x$$ ← 誤差の算出に使用する

（x：学習データ）

学習後 $$h_\theta(x) = \boxed{\theta_0} + \boxed{\theta_1} x$$ ← 推定に使用する

（x：テストデータ）

最適化された値

この後の流れ

線形回帰の問題設定を知った!

①予測値を導く式(仮定関数)をたてる

②目的変数と予測値の誤差を求める

③この誤差を最小化する問題を設定し、式(目的関数)をたてる

④目的関数の最適解を探索的に求める(最急降下法)

⑤最適解に至るとき、仮定関数の最適なパラメータが求まる

線形回帰
完