Machine Learning: Theory and Implementation using Python

Keishi Okudera

2018 年 5 月 30 日

目次

第 1 章

Python

1.1 Matplotlib

1.1.1 figure と axes

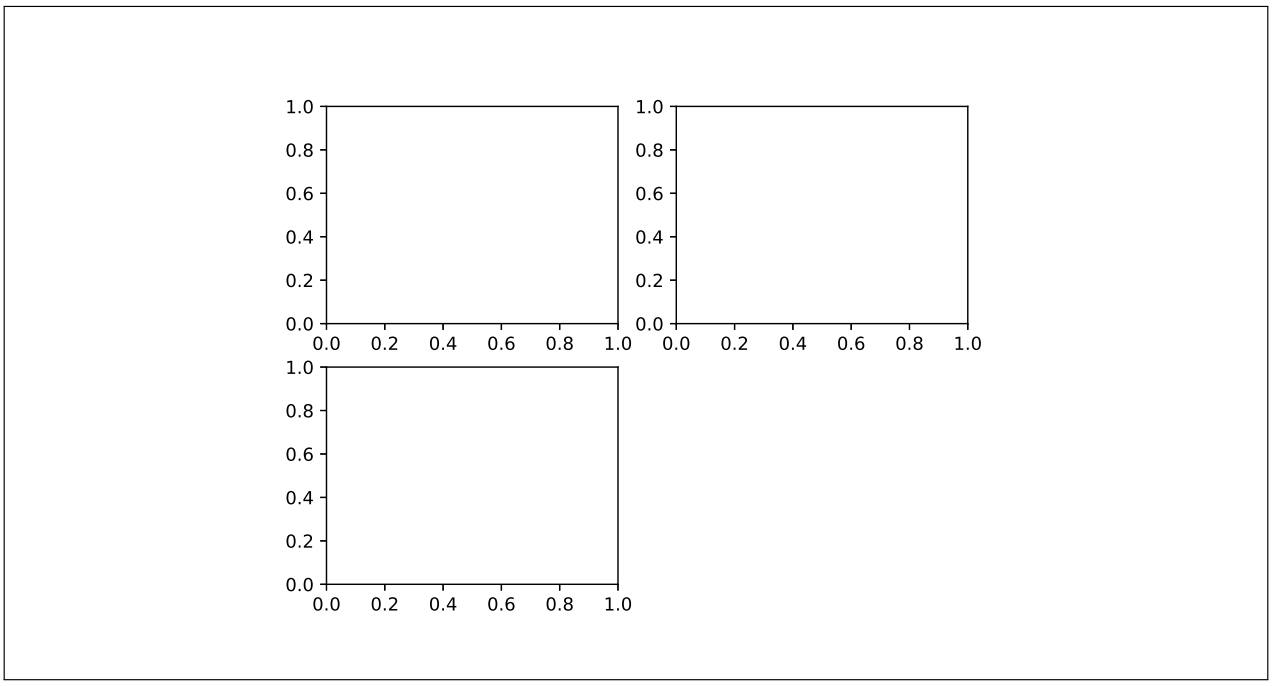
一つ一つのグラフの本体は，Matplotlib では axes オブジェクトとして表され，axes オブジェクトは，figure オブジェクトの中で管理される．イメージとしては、figure オブジェクトは白のキャンバスであり，そこにグラフの素である axes オブジェクトを置いていく．

Code 1 (fig1.py).

```
import matplotlib.pyplot as plt

fig = plt.figure()
ax1 = fig.add_subplot(2,2,1)
ax2 = fig.add_subplot(2,2,2)
ax3 = fig.add_subplot(2,2,3)

fig.savefig('fig1.eps')
```



- `plt.figure()`: 空の `figure` オブジェクトを作成する.
- `[figure].add_subplot(a,b,c)`: `figure` オブジェクトを a 行 b 列に分割した上で，c 番目の部分に空の `axes` オブジェクトを作成する.
- `[figure].savefig('XXXX.XXX')`: `figure` オブジェクトを `XXXX.XXX` として保存する.

1.1.2 散布図

まずはじめに散布図を描画してみよう.

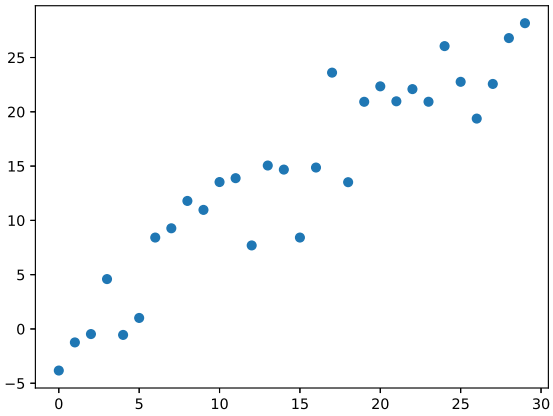
Code 2 (`fig2.py`).

```
import matplotlib.pyplot as plt
import numpy as np

fig = plt.figure()
ax = fig.add_subplot(1,1,1)

ax.scatter(np.arange(30), np.arange(30) + 3 * np.random.randn(30))

fig.savefig('fig2.eps')
```



第 2 章

機械学習の概念

2.1　機械学習の定義

Arthur Samuel は，機械学習を以下のとおり定義している．

定義 1 (機械学習 (Arthur Samuel(1959)))**.**
機械学習 (machine learning) とは，明示的にプログラミングしなくてもコンピュータに学習能力を与える研究分野のことである．

では，学習とは何か．Tom Mitchell は，学習を，タスク T，経験 E，性能指標 P を用いて，以下の通り定義している．

定義 2 (学習，タスク T，経験 E，性能指標 P(Tom Mitchell(1998)))**.**
性能指標 (performance measure)P で測定される，**タスク (task)**T における性能が**経験 (experience)**E により改善されることを，そのタスク T のクラスおよび性能指標 P に関して経験 E から**学習 (learn)** するという．

問題 1. 以下の機械学習の各事例においてタスク T，経験 E，性能指標 P を答えなさい．

1. 将棋プログラムが，自身を相手に数万回もの対局を行い，どのような棋譜が勝つまたは負ける傾向になるかを学習していき，人間よりも将棋が強くなった．
2. 電子メールクライアントが，どの電子メールをスパムとしてフラグを立てるかどうかを判断しようとしている．人間がどの電子メールがスパムかを電子メールクライアントに逐一報告していくことにより，より正確にスパムであるかどうかの判断を行えるようになっていった．
3. 過去の天気データから，将来の天気を予測する．

解答.

1. T は「将棋をさすこと」，E は「自身を相手に数万回もの対局を行なった経験」，P は「次の対戦で勝利する確率」.
2. T は「電子メールをスパムかどうか判断すること」，E は「各電子メールがスパムかどうかの報告内容」，P は「正しくスパムと判断できる確率」.
3. T は「将来の天気を予測すること」，E は「過去の天気データ」，P は「正しく天気を当てられる確率」.

□

2.2　教師あり学習と教師なし学習

機械学習には，教師あり学習と教師なし学習がある．それぞれ以下の通り定義される．

定義 3 (教師あり学習)**.**
教師あり学習 (supervised learning) とは，入力に対して正しい出力がわかるデータを用いた機械学習のことである．

定義 4 (教師なし学習)**.**
教師なし学習 (unsupervised learning) とは，単にデータが与えられ，そのデータから何らかの構造関係を導出する機械学習のことである．

教師あり学習は，回帰問題と分類問題に分けられる．

定義 5 (回帰問題)**.**
回帰問題 (regression problem) とは，出力が連続値である教師あり学習のことである．

定義 6 (分類問題)**.**
分類問題 (classification problem) とは，出力が離散値である教師あり学習のことである．

問題 2. 以下の機械学習の各事例において教師あり学習か教師なし学習か答えなさい．また，教師あり学習の場合，それが回帰問題であるか分類問題であるかを答えなさい．

1. 大量に在庫がある商品を抱えている．それが 3 ヶ月以内に何個売れるか予測する．
2. あるソフトウェアに使われている各ライセンスについて，それが正規ライセンスか不正ライセンスかを予測する．
3. 毎日何万ものニュースを集めてきて，関連する記事にグループ分けする．
4. ある人物の絵を見て，その人物の年齢を予測する．
5. 電子メールのやりとり履歴から，自動的にどれが密接な友人のグループかを特定する．
6. 2 つのマイクで拾った 2 人の声を解析して分離する (Cocktail Party Problem)．

解答.

1. 教師あり学習の回帰問題.
2. 教師あり学習の分類問題.
3. 教師なし学習.
4. 教師あり学習の回帰問題.
5. 教師なし学習.
6. 教師なし学習.

□

第 3 章

教師あり学習

3.1　トレーニングセットと仮説関数

教師あり学習は，特徴量と目的変数の組のデータを用いて学習する．学習に使用するデータをトレーニングセットという．

定義 7 (トレーニングセット)**.**
$\{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^{m} \subset ((\mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_n) \times \mathcal{Y})^m$ を**トレーニングセット (training set)** という．ここで，m は**トレーニングサンプル数 (number of training examples)**，$\boldsymbol{x}^{(i)} \in \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_n$ は i 番目の n 個の**入力変数 (input variables)** または**特徴量 (features)** で，$\boldsymbol{x}^{(i)} = (x_1^{(i)}, x_2^{(i)}, \ldots, x_n^{(i)})^T$ と表す．また，$y^{(i)} \in \mathcal{Y}$ は i 番目の**出力変数 (output variable)** または**目的変数 (target variable)** である．また，トレーニングセットの i 番目の要素 $(\boldsymbol{x}^{(i)}, y^{(i)}) \in (\mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_n) \times \mathcal{Y}$ を**トレーニングサンプル (training example)** という．また，入力変数のとる空間 $\mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_n$ を**入力変数空間 (space of input values)**，出力変数のとる空間 $\mathcal{Y}$ を**出力変数空間 (space of output values)** という．

問題 3. 次のトレーニングセットにおいて，$x_3^{(4)}, y^{(2)}$ を答えよ．

i	$x_1^{(i)}$	$x_2^{(i)}$	$x_3^{(i)}$	$x_4^{(i)}$	$y^{(i)}$
1	2104	5	1	45	460
2	1416	3	2	40	232
3	1534	3	2	30	315
4	852	2	1	36	178

解答. $x_3^{(4)} = 1,\ y^{(2)} = 232.$ □

教師あり学習を使って解きたいタスク T は，入力変数から出力変数を予測することであるが，それは言い換えると入力変数を引数として出力変数を出力する写像を設定することである．この写像を仮説関数という．仮説関数を以下で定義する．

定義 8 (仮説関数)**.**
入力変数 $\boldsymbol{x}$ から出力変数 y への写像 $h : \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_n \to \mathcal{Y}$，すなわち $h_{\boldsymbol{\theta}}(\boldsymbol{x})$ を**仮説関数 (hypothesis function)** という．ここで，$\boldsymbol{\theta}$ は仮説関数の**パラメータ**である (パラメータは複数あることがほとんどなのでベクトルとしている)．

すなわち，教師あり学習とは，仮説関数を設定し，トレーニングセットを用いて仮説関数の最適なパラメータを決定することといえる．

3.2　線形回帰

仮説関数は自らで与える必要があるが，仮説関数の形によって，教師あり学習に特別な名前がつくものがある．例えば，仮説関数を線形関数とし，出力変数空間を $\mathcal{Y} = \mathbb{R}$ としたときは線形回帰という．

定義 9 (線形回帰)**.**
トレーニングセット $\{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^{m} \subset ((\mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_n) \times \mathcal{Y})^m$ について，$\mathcal{Y} = \mathbb{R}$ であり，かつ仮説関数が式 (3.1) である教師あり学習を，特に**線形回帰 (linear regression)** という．ここで，特徴量 $\boldsymbol{x}^{(i)}$ は，常に 1 の値をとるような特徴量 $x_0^{(i)} = 1$ を付して $\boldsymbol{x}^{(i)} = (x_0^{(i)}, x_1^{(i)}, x_2^{(i)}, \ldots, x_n^{(i)})^T \in 1 \times \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_n$ と置き直すこととする．また，$\boldsymbol{\theta} = (\theta_0, \theta_1, \ldots, \theta_n)^T \in \mathbb{R}^{n+1}$ とする．

$$\begin{aligned} h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) &= \theta_0 x_0^{(i)} + \theta_1 x_1^{(i)} + \theta_2 x_2^{(i)} + \cdots + \theta_n x_n^{(i)} \\ &= \sum_{j=0}^{n} \theta_j x_j^{(i)} \\ &= \boldsymbol{\theta}^T \boldsymbol{x}^{(i)} \end{aligned} \tag{3.1}$$

問題 4. ある大学生について，1 年次の成績で優をとった個数から 2 年次にいくつ優をとるのか予測したい．そこで，何人かの大学生の 1 年次の成績の優の個数 x と 2 年次の成績の優の個数 y を集めた．その結果が次表である．このとき，次の問いに答えよ．

x	y
3	4
2	1
4	3
0	1

1. m はいくつか.
2. 仮説関数として $h_{\boldsymbol{\theta}}(x) = \theta_0 + \theta_1 x$ を設定し，本トレーニングセットを用いて線形回帰を行なった結果，パラメータは $\theta_0 = -1$, $\theta_1 = 2$ となった．このとき，1 年次の優の個数が 6 だった大学生の 2 年次の優の個数を予測せよ．

解答.

1. $m = 4$.
2. $h_{\boldsymbol{\theta}}(x) = -1 + 2x$ なので，$h_{\boldsymbol{\theta}}(6) = -1 + 2 \cdot 6 = 11$.

□

さて，仮説関数のパラメータをどう決めるかという問題がある．パラメータをでたらめに与えても良い予測値を返さないので，経験 E のトレーニングセットを使って，性能指標 P を高めるように仮説関数のパラメータをアップデートしていくことが必要になる．この手順を学習アルゴリズムといい，性能指標を測る関数を目的関数という．

定義 10 (学習アルゴリズム，目的関数)**.**
性能指標を測る関数 $J(\boldsymbol{\theta})$ を**目的関数**または**コスト関数 (cost function)** といい，この目的関数で測った性能指標が良くなるように仮説関数 (のパラメータ $\boldsymbol{\theta}$) をアップデートしていく手順を**学習アルゴリズム (learning algorithm)** という．

回帰問題における性能指標としては，各トレーニングサンプル $(\boldsymbol{x}^{(i)}, y^{(i)})$ での仮説関数 $h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})$ と $y^{(i)}$ の二乗誤差平均が考えられる．この二乗誤差平均を計算する目的関数を最小二乗誤差関数という．

定義 11 (最小二乗誤差関数)**.**
トレーニングセット $\{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^{m} \subset ((1 \times \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_n) \times \mathcal{Y})^m$ について，式 (3.2) で表される目的関数 $J(\boldsymbol{\theta})$ を**最小二乗誤差関数 (squared error function)** という．

$$J(\boldsymbol{\theta}) = \frac{1}{2m} \sum_{i=1}^{m} (h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) - y^{(i)})^2 \tag{3.2}$$

注意 1. m ではなく $2m$ で割っているのは，微分したときに出てくる 2 が消えるようにしているからである．m でも特段の問題はない (同じ結果が得られる)．

問題 5. あるトレーニングセット $\{(x^{(i)}, y^{(i)})\}_{i=1}^{3}$ をプロットしたところ以下の散布図となった (簡単のため，特徴量 x に $x_0 = 1$ である特徴量は追加していない)．仮説関数を $h_\theta(x) = \theta x$，目的関数 $J(\theta)$ を最小二乗誤差関数としたとき，$J(0)$ を求めよ．

Proof.
目的関数 $J(\theta)$ は次式となる．

$$\begin{aligned} J(\theta) &= \frac{1}{6} \sum_{i=1}^{3} (h_\theta(x^{(i)}) - y^{(i)})^2 \\ &= \frac{1}{6} \sum_{i=1}^{3} (\theta x^{(i)} - y^{(i)})^2 \end{aligned}$$

$\theta = 0$ を代入し，散布図から $y^{(i)}$ を読み取ると，

$$\begin{aligned} J(0) &= \frac{1}{6} \sum_{i=1}^{3} (-y^{(i)})^2 \\ &= \frac{1}{6}((-1)^2 + (-2)^2 + (-3)^2) \\ &= \frac{14}{6} \end{aligned}$$

□

「性能指標が良くなるように仮説関数のパラメータ $\boldsymbol{\theta}$ をアップデートしていく」とはどういうことか．目的関数が最小二乗誤差関数の場合，その最小二乗誤差がどんどん小さくなっていくことが，性能指標が良くなっていくといえる．すなわち，目的関数を最小にするパラメータ $\boldsymbol{\theta}$ を見つければよい．それを見つけるための手法が学習アルゴリズムである．

学習アルゴリズムの中で一般的なものとして最急降下法がある．最急降下法とは，目的関数 $J(\boldsymbol{\theta})$ のグラフ上に適当に点を打ち (すなわちパラメータ $\boldsymbol{\theta}$ として適当に初期値を決め)，その点からあたりを見渡してもっとも勾配が急な方向に一定程度進み，進んだ後の点からまたあたりを見渡してもっとも勾配が急な方向に一定程度進み... を繰り返して，どこを見渡しても勾配がないような点を探す方法である．

最急降下法を行うためには，関数上のある点について勾配が急な方向はどの方向であるかを計算しなければならない．勾配が最も急な方向を向くベクトルを勾配ベクトルといい，以下で定義される．

定義 12 (勾配ベクトル)**.**
k 次元ベクトル $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_k)^T$ からスカラー値に写る関数 $f(\boldsymbol{\theta})$，すなわち $f : \boldsymbol{\theta} \in \mathbb{R}^k \to \mathbb{R}$ である関数 $f(\boldsymbol{\theta})$ において，**勾配ベクトル** $\nabla_{\boldsymbol{\theta}} f = \frac{\partial f}{\partial \boldsymbol{\theta}}$ は次式で定義される．

$$\nabla_{\boldsymbol{\theta}} f = \frac{\partial f}{\partial \boldsymbol{\theta}} = \begin{pmatrix} \frac{\partial f}{\partial \theta_1} \\ \frac{\partial f}{\partial \theta_2} \\ \vdots \\ \frac{\partial f}{\partial \theta_k} \end{pmatrix} \tag{3.3}$$

問題 6. $f : \mathbb{R}^2 \to \mathbb{R}$ の関数 $f(x, y)$ のある点 $\mathrm{P}(x, y)$ における最大勾配方向が勾配ベクトルの方向と同方向であることを示せ．

Proof.
勾配とは，関数 f の変化度合いであり，勾配が最大ということは，関数の変化度合いが最も大きいということである．点 $\mathrm{P}(x, y)$ と，そこから微小量 Δx，Δy だけ動かした点 $\mathrm{Q}(x + \Delta x, y + \Delta y)$ においてそれぞれ関数値を求めて差をとったものを変化度合い Δf とすると，$\overrightarrow{PQ} = \overrightarrow{OQ} - \overrightarrow{OP} = (\Delta x, \Delta y)$ に注意して，以下の通り変形できる．

$$\begin{aligned}
\Delta f &= f(x + \Delta x, y + \Delta y) - f(x, y) \\
&= f(x + \Delta x, y + \Delta y) - f(x, y + \Delta y) + f(x, y + \Delta y) - f(x, y) \\
&= \frac{f(x + \Delta x, y + \Delta y) - f(x, y + \Delta y)}{\Delta x} \Delta x + \frac{f(x, y + \Delta y) - f(x, y)}{\Delta y} \Delta y \\
&\fallingdotseq \frac{\partial f}{\partial x} \Delta x + \frac{\partial f}{\partial y} \Delta y \\
&= \nabla f \cdot (\Delta x, \Delta y) \\
&= \nabla f \cdot \overrightarrow{PQ}
\end{aligned}$$

すなわちこれは，関数の変化度合いは勾配ベクトルと点 P から点 Q への方向ベクトル，すなわち微小量を動かした方向のベクトルの内積となっている．ここで，内積の定義より，

$$\Delta f = |\nabla f| |\overrightarrow{PQ}| \cos\theta$$

となる．ここで，θ は，∇f と $\overrightarrow{PQ}$ のなす角である．関数の変化度合い Δf が最大となるのは，$\cos\theta = 1$，すなわち $\theta = 0$ となる場合である．これはつまり ∇f と $\overrightarrow{PQ}$ が同じ方向を向いているときに関数の変化度合い Δf が最大となるということである．以上より，点 P から関数の変化度合い Δf が最大となるように進むためには (点 Q をとるためには)，勾配ベクトルの方向に進めばよいということである．　□

これで勾配が最も急な方向が勾配ベクトル方向であることがわかったので，それを用いて最急降下法を以下の通り定義する．

定義 13 (最急降下法)**.**
関数 $J(\boldsymbol{\theta})$ を最小とする $\boldsymbol{\theta}$ を次の手順で見つけるアルゴリズムを，**最急降下法 (gradient descent algorithm)** という．ここで，$\alpha(> 0)$ を**学習率**といい，勾配が最大の方向にどの程度移動させるかの強さを表す．

Algorithm 1 最急降下法

1: トレーニングセット $\{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^m$，仮説関数 $h_{\boldsymbol{\theta}}(\boldsymbol{x})$，目的関数 $J(\boldsymbol{\theta})$ を用意
2: $\alpha \leftarrow$ 初期値
3: $\boldsymbol{\theta} \leftarrow$ 初期値
4: **while** $\boldsymbol{\theta}$ が収束または有限回繰り返し **do**
5: $\quad \{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^m$ を代入して $J(\boldsymbol{\theta})$ を計算
6: $\quad \nabla_{\boldsymbol{\theta}} J$ を計算
7: $\quad \boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \alpha \nabla_{\boldsymbol{\theta}} J$ ▷ パラメータ $\theta_0, \theta_1 \ldots$ は同タイミングで更新
8: **end while**

問題 7. 特徴量が n 個のトレーニングセット $\{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^m$ の線形回帰において，目的関数 $J(\boldsymbol{\theta})$ を最小二乗誤差関数としたとき，$\nabla_{\boldsymbol{\theta}} J$ を計算せよ．

Proof.
$\nabla_{\boldsymbol{\theta}} J$ の j 番目の要素 $\frac{\partial J}{\partial \theta_j}$ を計算すると以下となる．

$$\begin{aligned}\frac{\partial J}{\partial \theta_j} &= \frac{1}{2m}\sum_{i=1}^m \frac{\partial}{\partial \theta_j}(h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) - y^{(i)})^2 \\ &= \frac{1}{2m}\sum_{i=1}^m 2(h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) - y^{(i)})\frac{\partial}{\partial \theta_j} h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) \\ &= \frac{1}{m}\sum_{i=1}^m (h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) - y^{(i)}) x_j^{(i)}\end{aligned}$$

ここで，$j = 0$ のときは $\frac{\partial}{\partial \theta_0} h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) = 1$ に注意してまとめると，

$$\nabla_{\boldsymbol{\theta}} J = \begin{pmatrix} \frac{1}{m}\sum_{i=1}^m (h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) - y^{(i)}) \\ \frac{1}{m}\sum_{i=1}^m (h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) - y^{(i)}) x_1^{(i)} \\ \frac{1}{m}\sum_{i=1}^m (h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) - y^{(i)}) x_2^{(i)} \\ \vdots \\ \frac{1}{m}\sum_{i=1}^m (h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) - y^{(i)}) x_j^{(i)} \\ \vdots \\ \frac{1}{m}\sum_{i=1}^m (h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) - y^{(i)}) x_n^{(i)} \end{pmatrix}$$

となる（$\nabla_{\boldsymbol{\theta}} J$ は $n+1$ 次元ベクトルであることに注意）． □

最急降下法がうまく収束しているかを確認する手段としては，ループ 1 回ごとに $J(\boldsymbol{\theta})$ の値をプロットしていき，値が順調に減少していっているかどうかをみるという方法がある．これをデバッグという．

定義 14 (デバッグ)**.**
最急降下法がうまく収束していることを確認するため，横軸に繰り返し回数 (number of iterations)，縦軸に目的関数値をとり図示することを**デバッグ (debugging)** という．

例えば，以下のような図となれば，最急降下法はうまく収束している．
学習率 α をどう選ぶかは難しい．小さすぎると収束までにかなりの時間がかかり，大きすぎると収束せずに発散することもある．ほどほどの数値が最も速く収束する．

問題 8. ある回帰問題の目的関数を最急降下法で最小化する．試しに，学習率 $\alpha = 0.01, 0.1, 1$ の 3 パターンで計算してデバッグしたところ，図 A,B,C の通りとなった．それぞれの図について，どの学習率で計算を行なったものと想定されるか答えよ．

Proof. A が $\alpha = 0.1$，B が $\alpha = 0.01$，C が $\alpha = 1$. □

特徴量は，与えられたものをそのまま使う必要はなく，それらを組み合わせるなどして自分で新しく作っても良い．うまく特徴量を作り出して，より単純な仮説関数にするという選択肢もある．

問題 9. 今，手元に特徴量 $x_1^{(i)}$:間口，$x_2^{(i)}$:奥行き，出力変数 $y^{(i)}$:土地の価格がある．このとき，土地の価格を予測する問題を線形回帰で解くことを考えると，仮説関数としてまず $h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) = \theta_0 + \theta_1 x_1^{(i)} + \theta_2 x_2^{(i)}$ が考えつくが，もっと簡単に仮説関数を定めるにはどうすれば良いか．

Proof. 新しい特徴量 $x_3^{(i)} = x_1^{(i)} x_2^{(i)}$:面積を設定し，仮説関数として $h_{\boldsymbol{\theta}}(x^{(i)}) = \theta_0 + \theta_1 x_3^{(i)}$ とする． □

3.3　多項式回帰

仮説関数として，線形関数を選ぶ必要はない．仮説関数として多項式を設定した場合は多項式回帰と呼ばれる．

定義 15 (多項式回帰)**.**
回帰問題において，仮説関数を多項式とした場合，特に**多項式回帰 (polynomial regression)** という．

問題 10. トレーニングセット $\{(x_1^{(i)}, y^{(i)})\}_{i=1}^{1} 0$ をプロットしたところ，下図となった．これについての回帰問題を解きたい．仮説関数としてどのようなものが考えられるか．

Proof. 新しく特徴量として $x_2^{(i)} = (x_1^{(i)})^2$ を設定し，仮説関数として $h_{\boldsymbol{\theta}}(x^{(i)}) = \theta_0 + \theta_1 x_1^{(i)} + \theta_2 x_2^{(i)} = \theta_0 + \theta_1 x_1^{(i)} + \theta_2 (x_1^{(i)})^2$ とする． □

複数の特徴量がある場合，各特徴量のとりうる範囲がだいたい同じような範囲にあると，最急降下法の収束速度は速くなる．特徴量を変換してとりうる範囲の調整を行うことを，特徴量スケーリングという．

定義 16 (特徴量スケーリング)**.**
トレーニングセット $\{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^{m}$ について，$\boldsymbol{x}^{(i)}$ の各特徴量のとりうる範囲を同じような範囲に変換することを**特徴量スケーリング (feature scalling)** という．特に，特徴量 x_j の平均 μ_j を用いて $\frac{x_j - \mu_j}{s_j}$ のようにスケーリングすることを，**平均標準化 (mean normalization)** という．ここで，s_j は，特徴量 x_j の標準偏差または $\max - \min$ とする．

注意 2. 特徴量スケーリングの方法は何か定まった方法があるわけではない．例えば，最大値で割るとか，平均を引いて最大と最小の差で割るとか，なんとなく 1000 分の 1 するとか，やり方はなんでもよいし，特徴量ごとに異なる方法をとっても良い．大事なことは全ての特徴量をだいたい似たような範囲にもっていくことである．なお，範囲としては，Andrew Ng 曰く，各特徴量の取りうる範囲がだいたい -3 〜3 の中の範囲をとるのを目安にしているとのこと．

特に，多項式回帰の場合は特徴量のオーダーが他の特徴量と比べて極端に異なることが多いため，特徴量スケーリングがほとんど必須となってくる．

問題 11. 住宅の平米数から住宅価格を予測する回帰問題を解くことを考える．集めたデータの平米数はおおむね 1 から 1000 フィートの範囲となっている．平米数と価格をプロットしたところ，以下の仮説関数があてはまりがよさそうと考えた．

$$h_{\boldsymbol{\theta}}(x^{(i)}) = \theta_0 + \theta_1 \text{平米数}^{(i)} + \theta_2 \sqrt{\text{平米数}^{(i)}}$$

ここで，特徴量スケーリングをしてより適切に回帰問題を解くために，新しく仮説関数を

$$h_{\boldsymbol{\theta}}(x^{(i)}) = \theta_0 + \theta_1 x_1^{(i)} + \theta_2 x_2^{(i)}$$

としたとき，$x_1^{(i)}$ と $x_2^{(i)}$ はそれぞれどのようにおくのがよいか．ここで，$\sqrt{1000} \fallingdotseq 32$ とする．

Proof. $x_1^{(i)} = \frac{\text{平米数}^{(i)}}{1000}$，$x_2^{(i)} = \frac{\sqrt{\text{平米数}^{(i)}}}{32}$. □

3.4 正規方程式

さて，最急降下法で収束して得られた $\boldsymbol{\theta}$ は，どこを見渡しても勾配がない状態となっている．これは，言い換えると目的関数 $J(\boldsymbol{\theta})$ の勾配ベクトルがゼロベクトルとなる点 $\boldsymbol{\theta}$ は，方程式 $\nabla_{\boldsymbol{\theta}} J = \mathbf{0}$ の解である．この方程式は正規方程式という．

定義 17 (正規方程式)**.**
目的関数 $J(\boldsymbol{\theta})$ の回帰問題について，次式を**正規方程式 (normal equation formula)** という．

$$\nabla_{\boldsymbol{\theta}} J = \mathbf{0} \tag{3.4}$$

ここから，線形回帰問題において目的関数を最小二乗誤差関数とした場合の正規方程式を導出する．その準備のため，デザイン行列を定義する．

定義 18 (デザイン行列)**.**
特徴量が n のトレーニングセット $\{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^m$ において，次式で定義する行列 $X \in \mathbb{R}^{m\times(n+1)}$ を**デザイン行列 (design matrix)** という．デザイン行列は，特徴量 $\boldsymbol{x}^{(i)}$ を転置してサンプル数の分縦に並べたもので表される．ここで，特徴量として $x_0 = 1$ が加わっていることに注意する．

$$X = \begin{pmatrix} {\boldsymbol{x}^{(1)}}^T \\ {\boldsymbol{x}^{(2)}}^T \\ \vdots \\ {\boldsymbol{x}^{(m)}}^T \end{pmatrix} = \begin{pmatrix} 1 & x_1^{(1)} & x_2^{(1)} & \cdots & x_n^{(1)} \\ 1 & x_1^{(2)} & x_2^{(2)} & \cdots & x_n^{(2)} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_1^{(m)} & x_2^{(m)} & \cdots & x_n^{(m)} \end{pmatrix} \tag{3.5}$$

問題 12. 次のトレーニングセットにおいて，デザイン行列 X を答えよ．

i	$x_1^{(i)}$	$x_2^{(i)}$	$x_3^{(i)}$	$x_4^{(i)}$	$y^{(i)}$
1	2104	5	1	45	460
2	1416	3	2	40	232
3	1534	3	2	30	315
4	852	2	1	36	178

Proof. 特徴量が 4 つで，サンプル数が 4 つなので，X は 4×5 次元の行列となり，

$$X = \begin{pmatrix} 1 & 2104 & 5 & 1 & 45 \\ 1 & 1416 & 3 & 2 & 40 \\ 1 & 1534 & 3 & 2 & 30 \\ 1 & 852 & 2 & 1 & 36 \end{pmatrix}$$

□

デザイン行列を使うと，線形回帰の仮説関数のベクトルを簡潔に表すことができる．

問題 13. 特徴量が n であるトレーニングセット $\{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^{m}$ における線形回帰問題を考える．このとき，$(h_{\boldsymbol{\theta}}(x^{(1)}), h_{\boldsymbol{\theta}}(x^{(2)}), \ldots, h_{\boldsymbol{\theta}}(x^{(m)}))^T$ をデザイン行列 X を用いて表せ．

Proof.

$$\begin{aligned} \begin{pmatrix} h_{\boldsymbol{\theta}}(x^{(1)}) \\ h_{\boldsymbol{\theta}}(x^{(2)}) \\ \vdots \\ h_{\boldsymbol{\theta}}(x^{(m)}) \end{pmatrix} &= \begin{pmatrix} \boldsymbol{\theta}^T \begin{pmatrix} 1 \\ \boldsymbol{x}^{(1)} \end{pmatrix} \\ \boldsymbol{\theta}^T \begin{pmatrix} 1 \\ \boldsymbol{x}^{(2)} \end{pmatrix} \\ \vdots \\ \boldsymbol{\theta}^T \begin{pmatrix} 1 \\ \boldsymbol{x}^{(m)} \end{pmatrix} \end{pmatrix} = \begin{pmatrix} \theta_0 + \theta_1 x_1^{(1)} + \theta_2 x_2^{(1)} + \cdots \theta_n x_n^{(1)} \\ \theta_0 + \theta_1 x_1^{(2)} + \theta_2 x_2^{(2)} + \cdots \theta_n x_n^{(2)} \\ \vdots \\ \theta_0 + \theta_1 x_1^{(m)} + \theta_2 x_2^{(m)} + \cdots \theta_n x_n^{(m)} \end{pmatrix} \\ &= \begin{pmatrix} 1 & x_1^{(1)} & x_2^{(1)} & \cdots & x_n^{(1)} \\ 1 & x_1^{(2)} & x_2^{(2)} & \cdots & x_n^{(2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_1^{(m)} & x_2^{(m)} & \cdots & x_n^{(m)} \end{pmatrix} \begin{pmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_n \end{pmatrix} = X\boldsymbol{\theta} \end{aligned} \tag{3.6}$$

□

また，この結果を用いて，目的関数を最小二乗誤差関数とした線形回帰問題において，目的関数をより簡潔に表すことができる．

問題 14. 特徴量が n であるトレーニングセット $\{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^{m}$ における線形回帰問題において，目的関数 $J(\boldsymbol{\theta})$ を最小二乗誤差関数とする．このとき，$J(\boldsymbol{\theta})$ をデザイン行列 X を用いて表せ．

Proof.

$$\begin{aligned} J(\boldsymbol{\theta}) &= \frac{1}{2m} \sum_{i=1}^{m} (h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) - y^{(i)})^2 \\ &= \frac{1}{2m} \left(h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(1)}) - y^{(1)}, h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(2)}) - y^{(2)}, \ldots, h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(m)}) - y^{(m)} \right) \begin{pmatrix} h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(1)}) - y^{(1)} \\ h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(2)}) - y^{(2)} \\ \vdots \\ h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(m)}) - y^{(m)} \end{pmatrix} \end{aligned}$$

となるが，ここで，

$$\begin{pmatrix} h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(1)}) - y^{(1)} \\ h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(2)}) - y^{(2)} \\ \vdots \\ h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(m)}) - y^{(m)} \end{pmatrix} = \begin{pmatrix} h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(1)}) \\ h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(2)}) \\ \vdots \\ h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(m)}) \end{pmatrix} - \begin{pmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(m)} \end{pmatrix} = X\boldsymbol{\theta} - \boldsymbol{y}$$

より，

$$J(\boldsymbol{\theta}) = \frac{1}{2m}(X\boldsymbol{\theta} - \boldsymbol{y})^T(X\boldsymbol{\theta} - \boldsymbol{y}) \tag{3.7}$$

□

目的関数を簡潔に書けたので，行列の微分の性質を使い，正規方程式を導出する．

問題 15. 特徴量が n であるトレーニングセット $\{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^m$ における線形回帰問題において，目的関数 $J(\boldsymbol{\theta})$ を最小二乗誤差関数とする．このとき，$\nabla_{\boldsymbol{\theta}} J = \mathbf{0}$ を簡単に表せ．

Proof. X^TX は対称行列であることに注意すると，

$$\begin{aligned}
\nabla_{\boldsymbol{\theta}} J &= \nabla_{\boldsymbol{\theta}} \left(\frac{1}{2m}(X\boldsymbol{\theta} - \boldsymbol{y})^T(X\boldsymbol{\theta} - \boldsymbol{y}) \right) \\
&= \nabla_{\boldsymbol{\theta}} \left(\frac{1}{2m} \left((X\boldsymbol{\theta})^T(X\boldsymbol{\theta}) - (X\boldsymbol{\theta})^T\boldsymbol{y} - \boldsymbol{y}^T(X\boldsymbol{\theta}) + \boldsymbol{y}^T\boldsymbol{y} \right) \right) \\
&= \nabla_{\boldsymbol{\theta}} \left(\frac{1}{2m} \left((X\boldsymbol{\theta})^T(X\boldsymbol{\theta}) - 2(X\boldsymbol{\theta})^T\boldsymbol{y} + \boldsymbol{y}^T\boldsymbol{y} \right) \right) \\
&= \nabla_{\boldsymbol{\theta}} \left(\frac{1}{2m} \left(\boldsymbol{\theta}^T X^T X\boldsymbol{\theta} - 2\boldsymbol{\theta}^T X^T\boldsymbol{y} + \boldsymbol{y}^T\boldsymbol{y} \right) \right) \\
&= \frac{1}{2m} \left(\nabla_{\boldsymbol{\theta}}(\boldsymbol{\theta}^T X^T X\boldsymbol{\theta}) - 2\nabla_{\boldsymbol{\theta}}(\boldsymbol{\theta}^T X^T\boldsymbol{y}) + \nabla_{\boldsymbol{\theta}}(\boldsymbol{y}^T\boldsymbol{y}) \right) \\
&= \frac{1}{2m} \left(2X^T X\boldsymbol{\theta} - 2X^T\boldsymbol{y} \right)
\end{aligned}$$

より，$\nabla_{\boldsymbol{\theta}} J = \mathbf{0}$ とすると，

$$X^T X\boldsymbol{\theta} = X^T\boldsymbol{y} \tag{3.8}$$

□

正規方程式の導出まで終わった．最後は，これを $\boldsymbol{\theta}$ について解けば終わりである．ここで，X^TX が正則であれば，逆行列 $(X^TX)^{-1}$ を左から掛けることによって解けるが，正則でない場合（非正則，非可逆，特異の場合ともいう），解を持たない（不能）もしくは複数または無数の解（不定）となる．ここで，不能や不定だからそれでお手上げ，というわけにはいかないので，「いい感じの」の解を設定したいとする．この「いい感じ」の解は，X^TX の**ムーア・ペンローズ一般逆行列** $(X^TX)^+$ を用いて，次式で書ける．

$$\boldsymbol{\theta} = (X^TX)^+X^T\boldsymbol{y} \tag{3.9}$$

この議論の詳細は，まだ著者が理解できていないため，今の所は割愛する．

では，どのような場合に X^TX は非正則なのか．厳密には（著者がまだ理解できていないので）言わないが，Andrew Ng 曰く，以下 2 つの場合を念頭に置いておけばとりあえず良いとのこと．

1. **特徴量が冗長**：例えば，住宅価格の予測についての特徴量で，縦の長さ，横の長さ，面積の 3 つを考えた場合，面積は縦の長さと横の長さですでに捉えられているので，面積という特徴量が冗長である．
2. **データ数より特徴量が多い** $(m \leq n)$：$n = 100$ 個の特徴量を，$m = 10$ サンプルでフィッティングするのは，うまくいくこともあるかもしれないが良いアイデアではない．

教師あり学習の回帰問題について，パラメータを探す方法として最急降下法と正規方程式を解く方法の 2 種類を取り上げた．それぞれの手法のメリットやデメリットについて，Andrew Ng は以下の通り言及している．

- 最急降下法は，学習率 α を適切に選択する必要があるため，良さげな数値を何度か試行することが必要となってくる場合がある．一方で，正規方程式はその手間がない．
- 最急降下法の場合は，アルゴリズムがちゃんと機能しているか，ちゃんと収束しているかなどを確認しなければならないが，正規方程式はその手間がない．
- 正規方程式の場合，特徴量スケーリングを行う必要はない．
- 正規方程式は，逆行列を求める必要があるが，逆行列を計算するコストが非常に大きい．具体的には，逆行列の計算コストは特徴量 n の 3 乗のオーダーとなる．一方，最急降下法は特徴量 n 本に対する計算を繰り返すだけなので，計算コストは n のオーダーと非常に少ない．つまり，最急降下法は特徴量が数百万個あるような場合でも正しく機能する．だいたい $n = 10000$ が正規方程式ではなく最急降下法を選ぶ目安．

問題 16. プログラミング課題を後で追記

参考文献

[1] Andrew Ng, “Machine Learning”, Stanford University, https://www.coursera.org/learn/machine-learning.

[2] Christopher Brooks, “Applied Plotting, Charting & Data Representation in Python”, University of Michigan, https://www.coursera.org/learn/python-plotting.

[3] McKinney W., “Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython”, 2nd Edition, O’Reilly Media, 2017.