Github 版

一般的[*1]な機械学習入門

最終更新：2021 年 5 月 12 日

逢空れい@ranoiaru

[*1] ここで言う「一般的な」とは、「普通の人にもわかりやすい」という意味ではなく、数学界隈における「汎用性の高い」「広く対応可能な定義を導入している」という意味である。

目次

notation

- $\mathcal{X}$：特徴量空間。$\mathbb{R}^d$ の単連結な開部分集合
- $\mathcal{Y}$：ラベル空間。$\mathbb{R}^n$ もしくは $\mathbb{C}^n$ か、それらの上での確率測度の集合
- $(\Omega, \mathcal{F}, P)$：確率空間
- H：使う手法により異なる条件を満たす関数 $f : \mathcal{X} \to \mathcal{Y}$ の集合が為す可分ヒルベルト空間
- L：頻度論的手法における損失関数 $H \times \Omega \to \mathbb{R}$
- $\nabla_f L(f, \omega), f \in H$：頻度論的手法における勾配の定義。$\omega$ を固定した上でのフレシェ微分に f を代入したもの。
- $\mathcal{B}(A), A$ は距離空間：ボレル集合族
- $\mathcal{P}((\Omega, \mathcal{F})$：可測空間 $(\Omega, \mathcal{F})$ 上の確率測度全体の集合。$\mathcal{F}$ を略記し、Ω が距離空間である場合、$\mathcal{P}(\Omega)$ は $\mathcal{P}((\Omega, \mathcal{B}(\Omega))$ であるものとする。
- $\mathcal{R}$：リッジレット作用素
- $\mathcal{R}^*$:双対リッジレット作用素
- D_{ucp}：ucp 位相の入った cádlág な適合過程の集合。（すなわち発展的可測）
- L_{ucp}：同上。ただし上が確率積分の定義域であるのに対して、こちらは値域。確率過程として動いている空間が違ったりする。
- N：ポアソンランダム測度

1　はじめに

深夜テンションによる出来心で「数学徒向けの機械学習入門書を書いたら見てくれる人ふぁぼください。ふぁぼ多かったら実行します」と書いたら、なんか一晩で 500 以上ふぁぼられたうえ、なぜか 200 近くフォロワーが増えたのでやらざるを得ない。

1.1　本書執筆の経緯（よまなくてもいいよ）

本書の構想は三年前に遡る。当時私は機械学習に入門したばかりの修士課程学生だったが、その時に触れた書籍があまりに数学的厳密性にも一般性にも欠け、そのことに対する苛立ちをツイートにぶつけた。

その結果、燃えた。

通知が埋まり、私はしばらく鍵垢に籠った。

私の言い方も大いに悪かったのだが、あまりにも曲解されすぎではないかと当時の私は思ったものである。あの発言の意図としては、インターン時のパワハラ面接官や、twitter のひたすらマウントをとってくる某エンジニアに対して「数学ろくに知らないくせに『数学なんて機械学習に無意味』なんていうんじゃない」ということであり、断じて「純粋数学に明るくなきゃ本当に機械学習わかってるとは言えない」などということではない。誓ってもいい。

その時私は、数学徒向けの機械学習入門理論を整備すると宣言し、実際に修論にはそれを書いたり、数学徒のつどいで講演したりしたのだが、このように資料の形での公開はほとんどしていない。

機械学習界隈を騒がせてからちょうど三年。私も学生時代が終わり、数学寄りの機械学習研究者として仕事をしながら、修士時代の研究室へ社会人博士への入学をもくろむ日々。ツイートがバズったのもなにかの縁であろう。変な書き方のせいで不快な思いをさせてしまった方々へのお詫びも兼ねて、ここはひとつ、数学徒向け機械学習入門書の執筆を行うことにした。

(2020/10/12)

1.2　前提知識

本資料において、入門編の前提知識は「学部レベルの解析学」(主に関数解析とルベーグ積分, 確率論)、発展編はその都度必要な知識を補完されたし。

確率解析は基本的に [1] の流儀を使用する。[*2]

1.3　無料版と有料版

後に本書は booth 等で販売する予定です。有料版は論文公開直後の最先端研究の解説や、確率論・統計学の基礎からマリアヴァン解析まで数学の簡易的な解説も付属します。投げ銭感覚でもいいので買ってくださると非常にありがたいです。価格は 1 万円未満にはします。

また、有料版発売後、github で公開しているこの無料版は、よほどのミスや重大な数学的誤り以外修正しないので、ご了承ください。有料版は発見次第なるべく早く修正します。[*3]

1.4　5 月アップデートについて

本日、大幅なアップデートを施した。更新が遅れて申し訳ありません。

ここ半年と一か月、私の人生を激変させる出来事が立て続けに起き、二度と私が私として浮上することができない可能性まであったため、とてもじゃないですが本書の更新をしてくことができませんでした。

[*2] 数か月前まで無駄に難解なだけの読みにくい本だと思っていましたが、変な確率過程の研究をしだすとこんなにありがたい既存理論は他にないと分かります。この本がなかったら、私がここ 4 か月で出したの成果と同じ研究成果を出すのに 3 年くらいかかってそう。「ucp 位相の入った適合 cádlág 過程空間」の確率解析理論がそこそこ完成されてる事実があまりに便利すぎる。

[*3] booth は一度購入すれば制限なく新バージョン落とせます。

ですが当時に比べればかなり元気になりましたし、高卒認定試験のツイートでフォロワーが大幅に増えたため、今こそ Vtuber 計画を始動し、本の続きも書くべきだと考えて、一念発起してかなり文章を書きました。

先輩数学系 Vtuber さんはみんな声が男なので、その辺で差別化していけたらなと思います。(媚びるのを隠さないスタイル)

(2021.5.12)

2　入門：機械学習の一般的問題設定

この章では、機械学習の問題設定について、関数解析的に定義する。

定義 2.1　データの定義
多数の特徴量とラベルの組、$\mathcal{D}_t := \{(X_i, Y_i)\}_i \subset \mathcal{X} \times \mathcal{Y}$ を時刻 t のデータと呼ぶ。
また、データ観測前の σ 加法族を $\mathcal{F}_0$、データ $\mathcal{D}_t$ 観測後の σ 加法族を $\mathcal{F}_t$ と呼ぶ。

定義 2.2　推定量 $\mathcal{F}_t$ 可測な確率変数のことを推定量という。

データは $\mathcal{D}_1$ 一つだけのこともあれば、$\mathcal{D}_1, \mathcal{D}_2, ...,$ と時系列に渡って増え続けることもある。
推定量とは要するに「その時刻の情報で計算可能な数値」である。

2.1　頻度論的機械学習問題の定義

2.1.1　仮説空間

特徴量空間からラベル空間への写像のうち、手法によってことなる条件を満たすものが為す可分ヒルベルト空間 H を仮説空間と呼ぶ。この H の中から、なるべく良い f を選ぶのが、機械学習問題の目的である。

2.1.2　損失関数

損失関数 L と仮説空間 H の組は次を満たす必要がある。

定義 2.3　許容可能な組み合わせ
任意の $f \in H$ に対して、フレシェ微分に f を代入した値 $\nabla_f L(f, \omega)$ が確率 1 で H に含まれる。[*4]

$L(f, \omega)$ は $\mathcal{F}_1$ 可測な確率変数である。つまりデータがあってはじめて計算できる。
次の定理は関数解析的定義と、実際のアルゴリズムを結ぶ超重要定理である。

定理 2.1　フレシェ微分はユークリッド空間における勾配の拡張
H が有限次元ユークリッド空間 $\mathbb{R}^N$ と同型であるとき、フレシェ微分はユークリッド空間の勾配 $\nabla_\theta, \theta \in \mathbb{R}^N$ と一致する。

この関数解析学における基本的な定理により、本書の定義が実アルゴリズムの一般化であると言うことができる。

2.1.3　更新則

初期値関数 $f_0 \in H$ を乱数等によって決定したうえで、この関数は次のように更新されていく。[*5]

$$f_{n+1} := f_n - \alpha_n \nabla_f L(f, \omega) \tag{2.1}$$

$\{\alpha_n\}$ の列を学習率といい、$1e-4$ など小さめの数に設定される。後述の焼きなまし法やロビンスモンロー条件等により、数値が下がっていく場合も多い。
この更新を繰り返す行為を「学習」と呼び、$n \to \infty$ で $argmin_{f \in H} E[L(f, \omega)]$ となる f に収束することが示されているのが望ましい。[*6]

[*4] 実アルゴリズムやその無限次元化等はすべて問題ないので、この条件を意識せずとも別に構わないのですが、非線形変換かつ線形変換を自明に含まない無限次元の仮説空間という非常に病的な機械学習問題を設定する場合に必要になってきます。重箱の隅を突くなんてレベルじゃない話ですが、問題設定は可能な限り一般的に書きたいのでこんな条件が出てくる。

[*5] f_0 の決定方法は手法等により様々なテクニックが存在します。

[*6] 本当は解析的に $argmin_{f \in H} E[L(f, \omega)]$ を計算できるのが望ましいですが、そのような状況はまずありません。

2.2

機械学習とはなんなのか

機械学習とは、AI とは、この特徴量とラベルの関係を記述する関数 $f : \mathcal{X} \to \mathcal{Y}$ の関数が我々の観測不能な領域に存在すると考え、真の f をデータや様々な手法によって手元のコンピューター上に「近似」する手法である。

強化学習などの例外を踏まえるといささか乱暴な物言いであるが、初学者や機械学習を数理的に捉えなおしたい方はまずはこの認識を持ってほしい。

2.3 ベイズ論的問題設定

本書においてはあまりベイズ論には触れるつもりがないが、せっかくの問題設定なのでベイズ論にも応用しておく。

定義 2.4 頻度論とベイズ論

$\mathcal{Y}$ が $\mathbb{C}^n \mathrm{or} \mathbb{R}^n$ であるとき、この機械学習問題を頻度論的機械学習問題という。

$\mathcal{Y}$ が $\mathbb{C}^n \mathrm{or} \mathbb{R}^n$ を台に持つ確率測度の集合であるとき、これをベイズ論的機械楽手問題という。

$\mathcal{Y}$ が有限集合上の測度であるとき、頻度論かベイズ論かは事前分布の仮定を置くかによる。

$\mathcal{X}$ から確率測度の台 $\mathbb{C}^n \mathrm{or} \mathbb{R}^n$ への写像の集合で、適当なヒルベルト空間に制限したものを $\tilde{H}$ とおく

頻度論ではただ一つ真の f が存在すると考えたが、必ずしもそうではないと考えるのがベイズ論である。

$\tilde{H}$ 上に値を取る確率変数 $B(\omega)(f)$ を考える。当然この分布はデータとは独立ではない。

B の $\mathcal{F}_0$ 条件付き分布を事前分布[*7]、$\mathcal{F}_1$ 条件付き分布を事後分布という。[*8]

定理 2.2 ベイズの定理

B の $\mathcal{F}_t$ 条件付き分布を μ_t と置くと

$$d\mu_1(g) = \frac{\mathcal{L}(\mathcal{D}_1|g)d\mu(g)}{\int_H \mathcal{L}(\mathcal{D}_1|g')d\mu_0(g')} \tag{2.2}$$

これを用いて、ベイズ推論による f は次のように構成される。

定義 2.5 予測分布

ベイズ推定による f を構成する。$\mathcal{Y}$ の台上の測度 $f(x)$ を、x に対するラベルの予測分布という。

$$df(x)(y) := \int_H 1_{g(x)=y} d\mu_1(g) \tag{2.3}$$

要するに、ラベル空間の代わりにラベル空間上の測度をラベルと見做すのがベイズ論である。この考え方は、本書終盤の部分観測マルコフ決定過程と共通する。

これは謂わば最尤推定のベイズ化と言えるが、後の様々な頻度論的手法もベイズ化が可能である。具体的にどのような構成になるかは、あまりに記述が煩雑になり読者を置いてけぼりにしすぎるため、興味のある方のみ取り組んでほしい。

相当関数解析力が鍛えられるし、後の強化学習の章ではとんでもなく複雑な関数解析を乱用するため、良い練習にもなるだろう。

2.4 損失関数の構成例

ここでは、L の具体的な構成について、例を交えつつ解説する。

[*7] データを見る前に知っている情報等から構成される。

[*8] データを見たあとの分布です

損失関数は基本的に経験損失項と正則化項に分けられる。

2.4.1 経験損失

回帰問題においては、平均二乗誤差という概念を用いる。

二乗誤差とはその名の通り誤差の二乗で $|f(X)-Y|^2$ とすることで計算できる。この期待値を求めたい。X は i.i.d サンプリングであるため、X の出やすさの分布を $dp_X(x)$ という測度で表現するとすると、

$$E[|f(X)-Y|^2] = \int_{\mathcal{X}} |f(x)-y|^2 dp_X(x) \tag{2.4}$$

$$= \frac{1}{|\mathcal{D}|} \sum_{X_i,Y_i \in \mathcal{D}} |f(X_i)-Y_i|^2 \quad \text{（モンテカルロ近似）} \tag{2.5}$$

期待値を計算する際はモンテカルロ法であるため、データの数が多いほど近似制度が良くなる。

次に分類問題について。こちらは基本的に $\mathcal{Y}$ が有限集合でありを台に持つ確率測度の隔たり[*9]を測る。

台となる有限集合を $\tilde{\mathcal{Y}}$ と置き、データ Y はラベル i が正解であるとき、$Y=(0,.....,1,.....0)$(i 番目の成分のみ 1 で残りが 0) と表記することにする。有限集合上の確率測度のベクトル表記である。

近似する関数 f は x を受け取り $\tilde{\mathcal{Y}}$ 上の測度を返す。これを用いて、「正解ラベルに対して何パーセントの確率を返したか」で損失を測る。正解ラベルが 5 のとき、ラベル 5 である確率を 60 %と判定した場合と、40 %と判定した場合では、当然後者のほうが損失が多くなる。

この計算には対数 $-\log p$ を用いる。$p=1$ つまり 100 %そのラベルが正解であると判定しているなら、損失は 0 となる。

$$E[\sum_{j\in\tilde{\mathcal{Y}}} -Y_{\cdot_j} \log f(X)(j)] = \int_{\mathcal{X}} \sum_{j\in\tilde{\mathcal{Y}}} -Y_{\cdot_j} \log f(X)(j) dp_X(x) \tag{2.6}$$

$$= -\sum_{i} \in \mathcal{D} \sum_{j\in\tilde{\mathcal{Y}}} -Y_{i_j} \log f(X)(j) \tag{2.7}$$

この測度同士の隔たりの計算方法をクロスエントロピー誤差と言う。

問 2.1 これらの損失関数のベイズ版を考察せよ。

2.4.2 正則化

突然だが、ここで数列クイズである。

$$0, 3, 8, 15, 24, 35, \tag{2.8}$$

と来た時、次の数字はなんだろうか。

賢明な読者なら、$a_n = n^2-1$ とすぐに気づき、48 と答えるだろう。しかし、本当にそれだけだろうか。

例えば $n^1938576 - 34567n^35244 + ...$ といったトンデモ式で表現される可能性は考えられないだろうか。

しかし、誰もが n^2-1 が正しそうだと考える。これは「オッカムの剃刀」と呼ばれる考え方で、「どうせ事象が表現可能なら、単純な構造の方が正しそう」という考え方だ。

これを実現するために、f の複雑さそのものを「損失」と考え、経験損失と足し合わせる。これが正則化である。

代表的なものは Ridge 正則化と Losso 正則化である。

- Ridge 正則化:$||f||_H^2$

[*9] 回帰問題はただのユークリッド距離でしたが、こちらは距離の公理を満たさないので距離ではありません。そのため「距離」ではなく「へだたり」と書きます。

- Losso 正則化 H がユークリッド空間であるとき、マンハッタン距離 $|\theta_1| + |\theta_2| + |\theta_3| + ...$ と計算する

Losso はパラメータが疎になりやすい（0 になるパラメータが多い）というメリットがあり、Ridge 正則化は非常に数学的に扱いやすいメリットがある。本書では特に表記なき場合正則化と言えば Ridge 正則化である。

他にも、原点周りのみ線形で原点から離れたら 2 乗するいいとこどりの手法も存在する。

正則化項は上記項を 0.0001 倍など小さな数を掛けて（正則化率という）経験損失に足し合わせる。

H がユークリッド空間と同型であるとき、これは「大きなパラメータをなるべく使わない」ということである。特に Ridge 回帰では、「コンパクト集合外では原点に押し戻す作用がパラメータの大きさに線形に働く」[*10]という性質が、GLD 等のエルゴード性の考察において非常に重要になって来る。

2.5 機械学習の具体的な問題設定

損失関数の構成を踏まえて、有名な機械学習問題等を本書定義に当てはめていく。

2.5.1 具体例 1:線形回帰（フルバッチ）

上記の定義をもとに、決定論的なフルバッチの線形回帰は次のように書ける。

$$\mathcal{X} := \mathbb{R}^d \tag{2.9}$$
$$\mathcal{Y} := \mathbb{R} \tag{2.10}$$
$$\mathcal{H} := \{f : \mathcal{X} \to \mathcal{Y}, f(x) = \sum_{j=0}^{J} a_j \phi_j(x), a_j \in \mathbb{R}, \phi_j := x^j\} \tag{2.11}$$
$$L(f) := \frac{1}{n} \sum_{i=1}^{n} |y_i - f(x_i)|^2 \tag{2.12}$$

ここで、J, N, α は使用者自らが決定する変数で、「ハイパーパラメータ」と呼ばれる。

$\mathcal{H}$ は $\mathbb{R}^{J+1}$ と同型

2.5.2 具体例 2:線形回帰（確率的勾配法）

$$\mathcal{X} := \mathbb{R}^d \tag{2.13}$$
$$\mathcal{Y} := \mathbb{R} \tag{2.14}$$
$$\mathcal{H} := \{f : \mathcal{X} \to \mathcal{Y}, f(x) = \sum_{j=0}^{J} a_j \phi_j(x), a_j \in \mathbb{R}, \phi_j := x^j\} \tag{2.15}$$
$$L(f, \omega) := \frac{1}{|I(\omega)|} \sum_{i \in I(\omega)} |y_i - f(x_i)|^2 \tag{2.16}$$

$I(\omega) \subset \mathcal{D}$ はランダムに抽出してきたデータの一部で、確率空間 $(\Omega_1, \mathcal{F}_1, P_1)$ 上で定義されるものとする。

2.5.3 具体例 3:カーネル回帰（フルバッチ）

$\mathcal{X} := \mathbb{R}^d, \mathcal{Y} := \mathbb{R}$ と定める。あらかじめ正定値性を満たすカーネル関数 $k : \mathcal{X}^2 \to \mathcal{Y}$ を定めておく。

$$\mathcal{H} := \mathcal{H}_k \tag{2.17}$$
$$L(f) := \frac{1}{n} \sum_{i \in I(\omega)} |y_i - f(x_i)|^2 \tag{2.18}$$

[*10] 2 次なので勾配を計算すると線形オーダーになります。

2.5.4 具体例 4:浅いニューラルネット（回帰）

$$\mathcal{X} := \mathbb{R}^d \tag{2.19}$$
$$\mathcal{Y} := \mathbb{R} \tag{2.20}$$
$$\mathcal{H} := \{f : \mathcal{X} \to \mathcal{Y}, f(x) = W_2\eta(W_1x - b_1) - b_2\phi_j(x), W_1 \in \mathbb{R}^{L\times d}, W_2 \in \mathbb{R}^{1\times L}, b_1 \in \mathbb{R}^L, b_2 \in \mathbb{R}\} \tag{2.21}$$
$$L(f) := \frac{1}{n}\sum_{i=1}^{n} |y_i - f(x_i)|^2 \tag{2.22}$$

ただし $\eta : \mathbb{R}^L \to \mathbb{R}^L$ は活性化関数と呼ばれ、あらかじめ定めておいた非線形で Lipsitz 連続な関数 $\sigma : \mathbb{R} \to \mathbb{R}$ を用いて

$$\eta_i(z) := \sigma(z_i) \tag{2.23}$$

として定義される。

2.5.5 具体例 5:浅いニューラルネット（分類）

分類したいクラスの集合である有限集合 $C = \{c^{(1)}, c^{(2)}, ..., c^{(m)}\}$ に対して、データ $\tilde{\mathcal{D}} = \{x_i, c_i\}_{i=1}^n$ が存在している状況で、新たな x に対してどのクラスに属するかを予測する。

$\mathcal{X} := \mathbb{R}^d, \mathcal{Y} := \mathbb{R}^m$ とおき、成形されたデータ $\mathcal{D} := \{x_i, y_i\}_{i=1}^n$ を次のように定義する。

$$y_{ij} := \begin{cases} 1(c_i = c^{(j)}) \\ 0(else) \end{cases} \tag{2.24}$$

このうえで、$\mathcal{H}, L$ は次のように定義される。

$$\begin{aligned} \mathcal{H} := \{&g : \mathcal{X} \to \mathcal{Y}, f(x) = softmax(g(x)), g(x) = W_2\eta(W_1x - b_1) - b_2\phi_j(x), \\ &W_1 \in \mathbb{R}^{L\times d}, W_2 \in \mathbb{R}^{1\times L}, b_1 \in \mathbb{R}^L, b_2 \in \mathbb{R}\} \\ L(f) := &-\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m} y_{ij} \log f(x_i)_j \end{aligned}$$

この損失関数は交差エントロピーと呼ばれ、C 上の確率測度間の乖離度合いを表す距離のようなもの（距離の公理は満たさない）

新たな入力データ x に対して、$c_{\hat{i}}, \hat{i} = argmax_i f(x)$ を予測されるクラスとする。

2.5.6 具体例 6:深いニューラルネット（回帰）

ここでは中間層が N 層の場合を扱う。

基本的には具体例 4 と同じで、$\mathcal{H}$ のみが異なる。

$$\mathcal{H} := \{f : \mathcal{X} \to \mathcal{Y}, f(x) = V_{end}g_K \circ g_{K-1} \circ ..... \circ g_1(x) - b_{end}, g_i(x) = \eta(W_ix - b_i)\} \tag{2.25}$$

また、自然数列 $L := \{L_1, ..., L_K\}$ を用いて、$L_0 = d$ とおくと

$$W_i \in \mathbb{R}^{L_i\times L_{i-1}} \tag{2.26}$$
$$b_i \in \mathbb{R}^{L_i} \tag{2.27}$$
$$W_{end} \in \mathbb{R}^{1\times L_K} \tag{2.28}$$
$$b_{end} \in \mathbb{R} \tag{2.29}$$

$K = 1$ のとき、上記の浅いニューラルネットと等しくなる。

$K > 1$ のとき、このようなニューラルネットによる学習を「深層学習」という。

2.5.7 **具体例** 7:ResNet

近年主流になりつつある、深層学習の亜種である。ここでは本書で使用する定義を書く。

具体例 7,8 においては、$T \in (0, \infty)$ に対して、有限個の数列 $t_0 = 0 < t_1 < ..... < t_{K-1} < t_K = T$ を用いて

$$V_t = V_{t_i}(t \in [t_i, t_{i+1})) \tag{2.30}$$

とする。W, b^1, b^2 についても同様。具体例 9,10 においてはその限りではない。

この状況下で

$$\frac{dX_t^i}{dt} = g(t, X_t^i) \tag{2.31}$$

$$X_0^i = x_i \tag{2.32}$$

と置き、終端値 X_T をさらに処理し分類する関数 h(CNN においてはプーリング層と全結合層の合成となる) を用いて損失 $L(f), f := h \bigotimes g$ は、フルバッチの回帰問題の場合

$$L(f) := \frac{1}{n} \sum_{i=1}^{n} |y_i - h(X_T^i)|^2 \tag{2.33}$$

と定義される。

損失関数の値は各データに対する損失の平均と考えることができ $L^{(i)}(f) := \tilde{L}(x_i, f) := |y_i - h(X_T^i)|^2$ という関数を用いて

$$L(f) := \frac{1}{n} \sum_{i=1}^{n} L^{(i)}(f) \tag{2.34}$$

と書き直せる。ここで 4 章で重要になる終端値関数を定義する。

定義 2.6　終端値関数

終端値関数 $F : \mathbb{R}^d \to \mathbb{R}$ を次のように定義する。

$$F(X_T^i) := |y_i - h(X_T^i)|^2 \tag{2.35}$$

これは要するに「時系列 flow の終端から出力までの関数と、（世間一般で言うところの）損失関数の合成関数」と考えればよい。

実際のところ、明らかに次の等式が成り立つ。

$$L(f) = \frac{1}{n} \sum_{i=1}^{n} F(X_T^i) \tag{2.36}$$

ミニバッチ法、分類問題、そして具体例 8,9,10 の場合も同様に定義する。

損失関数をこう置き換えると、ResNet や SDEnet といった時系列 flow モデルに対して、非常に数学的解析がしやすくなる。そのため 4 章では L ではなく F を用いて様々な解析を行う。

2.5.8 **具体例** 8:StochasticDepth

確率空間 $(\Omega_2, \mathcal{F}_2, P_2)$ 上で定義された、ベルヌーイ分布に従う独立な確率変数列 $b_1, b_2, ..., b_N$ を考える。ただし各 b_i の確率分布はあらかじめ定められた写像 p を用いて $p(i) \in (0, 1)$ に従う。

$$x_{i+1} = x_i + b_i f(i, x_i) \tag{2.37}$$

という形で定義する。

ここで、もしミニバッチなら $(\Omega, \mathcal{F}, P) := (\Omega_1, \mathcal{F}_1, P_1) \bigotimes (\Omega_2, \mathcal{F}_2, P_2)$ とおく。

2.5.9 具体例 9:ODENet

$$\mathcal{H} = \{f : \mathcal{H}_1 \times \mathcal{H}_2 \to \mathbb{Y}, f(x) = h_{ab}(x_T)\} \tag{2.38}$$
$$\mathcal{H}_1 = \{h : \mathbb{T} \times \mathcal{X} \to \mathcal{X}, h(t, x) = V_t \eta(W_t x - b_t^1) - b_t^2 \tag{2.39}$$
$$\mathcal{H}_2 = \{h_{ab} : \mathcal{X} \to \mathcal{Y}, h_{ab}(x) = V_{ab} \eta(W_{ab} x - b_{ab}^1) - b_{ab}^2\} \tag{2.40}$$

ただし、$x_0 := x, x_t := \int_0^t h(s, x_s) ds$ とおく。

2.5.10 具体例 10:SDENet

本修士論文の主題である。上記の ODEnet を確率化する。

$$\mathcal{H} = \{f : \mathcal{H}_1 \times \mathcal{H}_2 \to \mathbb{Y}, f(x) = h_{ab}(X_T)\} \tag{2.41}$$
$$\mathcal{H}_1 = \{h : \mathbb{T} \times \mathcal{X} \to \mathcal{X}, h_1(t, x) = V_t \eta(W_t x - b_t^1) - b_t^2, h_2 = V_t^2 \eta(W_t^2 x - b_t^{12}) - b_t^{22}\} \tag{2.42}$$
$$\mathcal{H}_2 = \{h_{ab} : \mathcal{X} \to \mathcal{Y}, h_{ab}(x) = V_{ab} \eta(W_{ab} x - b_{ab}^1) - b_{ab}^2\} \tag{2.43}$$

ただし、$X_0 = x$ であり、X は $dX_t = h_1(t, X_t)dt + h_2(t, X_t)dB_t$ という確率微分方程式に従うものとする。

確率空間は $(\Omega, \mathcal{F}, P) := (\Omega_1, \mathcal{F}_1, P_1) \bigotimes (B, \mathcal{B}(B), \mu)$ と置く。ただし $(B, \mathcal{B}(B), \mu)$ は $B := C([0, T])$ とした場合の Wiener 空間である。h_1, h_2 が x に対して大域的 Lipschitz 連続で t に対して 1/2 次 Hölder 連続と置くことで、$X_T \in \mathbb{D}^\infty$ となる。つまり X_T が Wiener 汎関数と言え、上記の定義での損失関数が定義でき、また Malliavin 微分や部分積分の議論に持ち込める。

2.6 活性化関数の具体例

上述の $\sigma(x)$ について、一応大域的 Lipsitz 連続で非線形であればなんでもいいことになっているが、当然よく使われるものは存在する。

2.6.1 ReLu

$$\sigma(x) := max(0, x) \tag{2.44}$$

と定義される。「if 文一つで書ける」「勾配が消失しない」といった利点がある。

区分的に滑らかな関数を近似するにあたってこの形が都合がいいとする研究もある

本書では連続的に微分可能ではないこと、零点が Lebesgue 測度無限大に存在すること、区分的に微分しても 2 階微分が零関数になることなどから 6 章 1 項を除いて採用しない。

2.6.2 swish

$$\sigma(x) := x \cdot sigmoid(x) \tag{2.45}$$
$$sigmoid(x) := \frac{1}{1 + e^{-x}} \tag{2.46}$$

近年 Relu にとって代わって使われ始めている活性化関数。原点から離れるほど Relu に近づく。

C^∞ であること、任意の階数の導関数も含めて零点が Lebesgue 測度 0 であることなどから、準楕円性などを考察する状況おいては非常に都合がよく、原則こちらを用いることにする。

3　入門：ニューラルネットの積分表現理論

3.1　ニューラルネットの連続化

特徴量空間 $\mathcal{X}(:=\mathbb{R}^d)$ から、$\mathbb{C}$ への写像を構成する浅いニューラルネット考える。

$$f(x) = W_2\eta(W_1x - b_1) \tag{3.1}$$

ただし、W_2 は $1 \times J$ 複素行列、W_1 は $J \times d$ 実行列、b_1 は J 次元実ベクトルであるとする。

また $\eta : \mathbb{R}^J \to \mathbb{C}^J$ であり、ある非線形で大域的リプシッツ連続な写像 $\sigma : \mathbb{R} \to \mathbb{C}$ を用いて,$\eta_i(y) = \sigma(y_i)$ と書けるとする。

この計算をベクトル $a \in \mathbb{R}^d, b \in \mathbb{R}, \sigma, c \in \mathbb{C}$ を用いて書き直す

$$f(x) = \sum_{i=1}^{J} c_i\sigma(a_i \cdot x - b_i) \tag{3.2}$$

ここで、$J \to \infty$ とした形

$$f(x) = \int_{\mathbb{R}^{d+1}} \gamma(a, b)\sigma(a_i \cdot x - b_i)dadb \tag{3.3}$$

これを積分的ニューラルネットと呼ぶ。積分的ニューラルネットに対しては、正則化付き損失関数に対する大域的最適解が解析的に求められる場合がある。そのため、十分広いニューラルネットに対して、最適なパラメータの近似値が一回の数値計算で求められることになる。

この章では、特徴量空間上の測度 μ（データの分布）と、それに対して後の許容条件を満たすように自由に設定できる測度パラメータ空間上の測度 λ を扱えるようにするため、[2] で提唱された再生核 Hilbert 空間上の Ridgelet 解析を [3] による再生核 Hilbert 空間の理論により我流の再構成を行う。

3.2　再生核 Hilbert 空間上の積分表現理論

$\mathcal{X}$ 上の複素数値関数全体の集合を $\mathcal{F}(\mathcal{X})$ とおく。

パラメータ a, b の成す空間 $\mathbb{R}^{d+1}$ から $\mathbb{C}$ への写像のうち、$\mathbb{R}^{d+1}$ 上の測度 $\lambda(dadb)$ による L^2 空間を $\mathcal{G}(:= L^2(\mathbb{R}^{d+1} \to \mathbb{C}, \lambda(dadb)))$ とおく。

写像 $h : \mathcal{X} \to \mathcal{G}$ を固定し、次のような積分作用素 $S : \mathcal{G} \to \mathcal{F}(\mathcal{X})$ を、$F \in \mathcal{G}$ に対して、$f = SF$ となる $f \in \mathcal{F}(\mathcal{X})$ を、次の等式が成り立つ関数とすることによって定義する。

$$f(x) = \langle F, h(x)\rangle_{\mathcal{G}} \tag{3.4}$$

ここで、S の像空間 $\mathcal{E}(S) := S(\mathcal{G})$ に対して、次のようにノルムを入れる。

$$||f||_{\mathcal{E}(S)} := inf\{||F||_{\mathcal{G}} : SF = f\} \tag{3.5}$$

定理 3.1　再生核 Hilbert 空間 ([3])

$k : \mathcal{X}^2 \to \mathbb{C}$ を次のように定義する。

$$k(x, y) := \langle h(y), h(x)\rangle_{\mathcal{G}} \tag{3.6}$$

この時、$\mathcal{E}(S)$ は再生核 k を持つ再生核 Hilbert 空間

$\{h(x), x \in \mathcal{X}\}$ が $\mathcal{G}$ 上完全であることと、S が等距離写像であることは同値

今後、この $\mathcal{E}(S)$ を、再生核 Hilbert 空間であることを強調するために H_k と表記する。

定理 3.2　等距離元の存在 ([3])

任意の $f \in H_k$ に対し

$$||f||_{H_k} = ||F^*||_{\mathcal{G}} \tag{3.7}$$

を満たす $F^* \in \mathcal{G}$ が一意に存在する

この F^* を f の Ridgelet 変換と呼び、$\mathcal{R}f$ と表記する。

今後、定数 K を $\mathcal{X}$ 上の測度 μ を用いて,$K := \int_{\mathcal{X}} k(x,x)\mu(dx)$ と表記する。

$0 < K < \infty$ の時、この (μ, λ, h) の組は「許容条件を満たす」と呼ぶ。

定理 3.3　積分作用素の連続性

(μ, λ, h) が許容条件を満たすとき、$H_k \subset L^2(\mathcal{X}, \mu)$ であり、$S : \mathcal{G} \to L^2(\mathcal{X}, \mu)$ は連続作用素

proof.

H_k 上の関数列 $\{u_j\}_{j=1}^{\infty}$ を、$\mathcal{G}$ の正規直交基底 $\{v_j\}_{j=1}^{\infty}$ を用いて

$$u_j(x) := \langle v_j, h(x) \rangle_{\mathcal{G}} \tag{3.8}$$

と定義する。両辺の $L^2(\mathcal{X}, \mu)$ 上のノルムを取る。

$$\begin{aligned}
||u_j||^2_{L^2(\mathcal{X},\mu)} &= \int_{\mathcal{X}} u_j(x)\overline{u_j}(x)d\mu(x) && (3.9)\\
&= \int_{\mathcal{X}} (\int_{\mathbb{R}^{d+1}} v_j(z)\overline{h(x)(z)}d\lambda(z) \overline{\int_{\mathbb{R}^{d+1}} v_j(z)\overline{h(x)(z)}d\lambda(z)})d\mu(x) && (3.10)\\
&\leq \int_{\mathcal{X}} (\int_{\mathbb{R}^{d+1}} v_j(z)\overline{v_j(z)}d\lambda(z) \int_{\mathbb{R}^{d+1}} h(x)(z)\overline{h(x)(z)}d\lambda(z))d\mu(x) (Schwarz\ の不等式) && (3.11)\\
&= K||v_j||^2_{\mathcal{G}} && (3.12)
\end{aligned}$$

$u_j = Sv_j$ であることを踏まえると、任意の $F \in \mathcal{G}$ は $\{v_j\}_{j=1}^{\infty}$ の線形和で書けるため、同じ議論により

$$||SF||_{L^2(\mathcal{X},\mu)} \leq K||F||_{\mathcal{G}} \tag{3.13}$$

となる。$f \in H_k$ にはすべて $f = SF$ となる F が存在するため、定理の主張が言える。

許容条件に加え、$\{h(x) : x \in \mathcal{X}\}$ が $\mathcal{G}$ 上完全であるとき、「強い意味で許容条件を満たす」と呼ぶとする。

定理 3.4　H_k の正規直交基底

強い許容条件が満たされるとき、$\{u_j\}_{j=1}^{\infty}$ は H_k の正規直交基底

proof.

上記の定理より、$||f||_{H_k} = ||F^*||_{\mathcal{G}}$ となる F^* が存在し、また内積の線形性から、複素数列 $\{c_j\}_{j=1}^{\infty}, \sum |c_j|^2 < \infty$ を用いて

$$F^* = \sum_j c_j v_j \tag{3.14}$$

$$f = \sum_j c_j u_j \tag{3.15}$$

と書ける。強い許容条件が満たされる場合、S は等距離写像。そのため $f = u_j$ としたとき $F^* = v_j$ である。

一般の f, F^* に対して、Parseval の等式より $||F^*||_{\mathcal{G}} = \sum |c_j|^2$ で、これは $||f||_{H_k}$ と一致する。

分極公式により H_k の内積は $||\cdot||_{H_k}$ から陽に書け、$\langle u_j, u_i \rangle = \delta_{ij}$ が言える。

次はさらに強い条件を課す。この定理の条件を緩めていくことこそが、我々の再定義した積分表現理論における今後の課題となる。

定理 3.5　Ridgelet 変換の積分表示定理 ([3])

(μ, λ, h) は強い意味で許容条件を満たし、さらに任意の $f \in H_k$ に対して $||f||_{H_k} = ||f||_{L^2(\mathcal{X},\mu)}$ が成り立つとする。

また、単調増大な $\mathcal{X}$ の部分集合列 $\{E_N\}_{N=1}^{\infty}$ を、$\cup_{N=1}^{\infty} E_N = \mathcal{X}$ となるようにとり

$$(\mathcal{R}_N f)(z) := \int_{E_N} f(x)\overline{h(x)(z)}\mu(dx) \tag{3.16}$$

と置く、任意の N に対して $F_N \in \mathcal{G}$ なら

$$||\mathcal{R}_N f - \mathcal{R} f||_{\mathcal{G}} \to 0 (N \to \infty) \tag{3.17}$$

この定理によって F^* はまさに既存の Ridgelet 変換そのもので、既存研究は $\mu(dx) = dx$ と置いた場合であることがわかる。

今回の再定義の利点は、μ をデータの分布とすることで、よりデータに沿った形で Ridgelet 変換を定義できることにある。

実問題では、有限個のデータ $(x_i, y_i)_{i=1}^n$ から、なるべく”良い”写像 $y = f(x)$ を構成したい。既存の Ridgelet 解析では、この Ridgelet 変換を近似するにあたって

$$(\mathcal{R} f)(a, b) : = \int_{\mathbb{R}^d} f(x)\sigma(a \cdot x - b)dx \tag{3.18}$$

$$\approx \frac{1}{n}\sum_{i=1}^{n} f(x_i)\sigma(a \cdot x_i - b) \tag{3.19}$$

$$\approx \frac{1}{n}\sum_{i=1}^{n} y_i\sigma(a \cdot x_i - b) \tag{3.20}$$

とするしかない。しかしこの近似は積分の測度が Lebesgue 測度である場合は不自然な近似となっている。実際のデータは一様分布には程遠く、一様に近い分布になるようデータの一部を抽出するのは、d が大きい状況では非常に難しい。

ここで dx を特徴量データの分布 $\mu(x)$ に差し替えれば、上記の近似計算はモンテカルロ法として非常に自然なものとなる。

3.3　(先端研究) 定義域の制限

Relu は実装が楽で計算も早いが、簡単に爆発するため扱いにくい。[*11]

[4] では

[*11] 厳密には爆発を抑えてくれない。

4　残差学習の微分方程式解釈

4.1　微分方程式と深層学習

4.2　発展：マリアヴァン解析を用いた勾配誤差収束定理

5　最適化アルゴリズムとエルゴード性

5.1　様々な勾配法アルゴリズム

5.2　発展：SDE のエルゴード性と最適化アルゴリズムが非凸損失関数の大域的最適解に確率収束する条件

5.3　発展：（先端研究）現実的な計算時間で 1epoch 走らせられ、かつ非凸損失関数であっても大域的最適解に確率収束するアルゴリズム発見に向けた今後の課題

6　強化学習と確率制御

著者の今の飯の種である。

6.1　マルコフ決定過程

6.2　発展：部分観測マルコフ決定過程

6.3　発展：分布型強化学習

6.4　発展：統一理論・超一般化マルコフ決定過程

本当はこちらの定義を先に書いて、上記 3 定義はすべてこの特別な場合と言おうとしたが、強化学習初学者相手だといくら数学徒向けとはいえ鬼畜すぎるので最後に表記する。

6.5　（有料版限定）発展：（先端研究）レヴィ過程に基づく連続時間強化学習

著者の研究テーマである。

7　（有料版限定）：本書で用いられている数学の概説

参考文献

[1] P. Protter, Stochastic Integration and Differential Equations, Applications of Mathematics, Second edition, Vol. 21 (Springer-Verlag, Berlin, 2005).

[2] Sho Sonoda, Isao Ishikawa, Masahiro Ikeda, Kei Hagihara, Yoshihiro Sawano, Takuo Matsubara, Noboru Murata,Integral representation of shallow neural network that attains the global minimum.arXiv:1805.07517v2,2018

[3] S. Saitoh. Integral transforms, reproducing kernels and their applications. Addison Wesley Longman, 1997