書籍購入特典

用語集

用語集

数字

MEMO

4次元以上の行列式

4次元以上の行列式を求める際は1つ下の次元の行列式の線形結合に落とし込むことができるので、それを繰り返すことで行列式を求めていくことが可能です。

D

MEMO

DQN

deep Q-networkの略。強化学習の1つの方法であるQ学習をニューラルネットワークで行えるようにしたものです。Q学習については本書の4.8節で詳しく扱います。

M

MEMO

matplotlibとその機能

matplotlibではsubplotを使うと複数のグラフ画面を同時に表示させることができます。`subplot`（全体の行数, 全体の列数, 今プロットしたいグラフの位置）で、自分が現在プロットしているグラフの位置を指定してからグラフをプロットします。位置を指定する時は左上から0、1、2、3、4で次の行が5、6、7、8、9のように連番で指定します。
画像を表示させる機能も付いており、`plt.imshow`で表示させることが可能です。モノクロ画像だけでなくRGBの情報が入った画像も表示させることができます。

N

MEMO

NaNについて

NaNはその型では表せない数値やそもそも値が入っていない時などに使われる数となっており、Not a Numberの略称となっています。

O

MEMO

OpenAI

OpenAIは人工知能について研究を行っている非営利団体のことです。

P

MEMO

plot1()

plt.gca().set_aspect("equal", adjustable="box")は`plt.axis("equal")`で縦横比をうまく調整できなかった時に使うコマンドで、役割としては`plt.axis("equal")`と同じで、グラフの縦横比を一緒にします。
`plt.contourf()`は等高線を描き色分けするコマンドで`plt.contourf(X,Y,Z)`で指定することができますが、この等高線に白黒の模様などといった色を付けたい時はcmap引数のところで個別に指定します。ここで指定したplt.cm.boneでは白黒模様が作られるようにできますが、それ以外にも`autumn`や`copper`などいろいろな設定ができます。

MEMO

plot2()

plot2のコードの中身が本書のこれまで（P.365まで）と違う書き方になっています。これは、オブジェクト指向に則した書き方になっているからです。プロットするグラフをfigというオブジェクトに格納して、その中での3Dプロットするためのオブジェクトをaxに格納しています。なのでプロットする時は`plt.plot`ではなく、`ax.plot`という形になるわけです。
`Axes3D(plt.figure()).plot_wireframe(xx, yy, ret)`のようにひとまとめに書いても問題はありませんが、これをわかりやすくするためにこのようなオブジェクト指向の書き方になっています。また、Axes3Dはmatplotlibで3次元プロットをしたい時にインポートされるオブジェクトとなっています。このようなワイヤーでつながれたグラフ以外にも散布図などをプロットできます。

S

MEMO

SciPy.orgの解説

公式サイトの解説も参照してください。

- **numpy.convolve — NumPy v1.14 Manual - NumPy and SciPy Documentation**
  URL https://docs.scipy.org/doc/numpy-1.14.0/reference/generated/numpy.convolve.html

T

MEMO

TensorFlow Playground

- **A Neural Network Playground**
  URL https://playground.tensorflow.org

あ

MEMO

アダマール積

アダマール積とは、対応する要素同士の積を計算するもので内積や外積のようにそれらの和や差をとるものではありません。例えば、[1, 2] と [3, 4] の配列があるとすればこれらのアダマール積は [1x3, 2x4]=[3, 8] となります。

MEMO

一様分布

一様分布では外れ値が出てもその値が出る確率も平均値付近のものと同様とみなすことができてしまうので無視することができないものとなってしまいます。

MEMO

エルミート行列

エルミート行列とは、それぞれの値の複素共役な値を取った時に元の行列を転置したものと同一になるような行列のことを指します。すべての要素が実数の時は対角成分を境に要素が対象に並んでいる対象行列のことを指します。

か

MEMO

カイ二乗分布

カイ二乗分布というのは、標準正規分布に従った乱数の二乗をいくつか足したものを言います。足していく個数というのは、自由度kで決まります。

$$Z = \sum_{i=1}^{k} X_i^2$$

X_iは標準正規分布に従う独立の乱数

このZの分布がカイ二乗分布になります。

MEMO

確率密度関数のグラフ

グラフのヒストグラムを作成する時にbinsの値を指定していますが、ヒストグラムでいくつに分割して値をカウントしていくかを指定する引数（これを基数と言います）となります。この例では10万個サンプルがあったので基数を1,000個と多めにとってヒストグラムを作成しています。

MEMO

ガンマ分布

ガンマ分布とは、以下で表される$f(x)$に従った分布のことを言います。

$$f(x) = x^{a-1} \frac{e^{-\frac{x}{\beta}}}{\Gamma(\alpha)\beta^{\alpha}}$$

α、βは正のパラメータ

確率$\frac{1}{\beta}$で起こる事象がα回起こるまでの時間を表しています。ここで用いられる$\Gamma(\alpha)$はガンマ関数と呼ばれるものです。

MEMO

クォータニオン

クォータニオンは複素数を拡張した概念で、実部が1つと虚部が3つで構成されている数です。3次元空間での回転を表現することなどに使われています。

MEMO

グラフ化のコード

matplotlibのグラフ化のコードでplt.tight_layout()という関数を使っていますが、これを使うことでグラフ同士の重なり合いを防ぎ、グラフのタイトルが隠れたりすることを防いでくれます。

さ

MEMO

散布図のプロット

散布図をプロットする際にいろいろなオプションを付けることができます。このサンプルコードのようにプロットする色（`color`）やプロットの形（`marker`）を指定することができ、プロット自体にラベル（`label`）を付けることで凡例を出した時に名前がきちんと出るようになります。

MEMO

スパースグリッド

スパースグリッドとは格子点に対応する座標をすべて出力せずにNumPy上でブロードキャストが適用できるような形で格子点を出力したものです。メモリ使用量を抑えることができ、これを用いて計算処理を行ってもスパースグリッドを用いない場合と同じ結果が得られます。

MEMO

正規分布

正規分布は、以下の確率密度関数$f(x)$に従った分布のことを言います。

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

σ ＝標準偏差　　μ ＝平均

この中で、特に$\sigma = 1$、$\mu = 0$となっているものを**標準正規分布**と呼びます。

MEMO

正規分布

正規分布ということは平均値の付近ではその値がよく観測され、平均値より遠のけば遠のくほど、その値は観測されにくくなるということになります。そういった分布を想定した時は外れ値が出たとしても不思議ではないので、そういったものの影響を受けないような正規化をする必要があります。

MEMO

その他の逆行列を求める方法

他に求める手法としてLU分解や特異値分解、Gaus-Jordan法などがあります。

た

MEMO

畳み込み（convolution）の利用

音声処理や画像処理の分野ではデータの平滑化（ノイズの除去）やエッジの検出によく使われており、これにより音声や画像をよりクリアにすることが可能となります。

MEMO

畳み込みニューラルネットワーク（convolutional neural network）

畳み込みニューラルネットワークというのは主に画像に対して有効とされているニューラルネットワークで、$k \times k$でできたフィルターを１つ１つの画素について掛けていくことで、フィルターにより抽出された特徴を持った画像を生成していくニューラルネットワークです。

MEMO

中央値は外れ値に対してあまり影響を受けない

例えば、６人の生徒が受験した100点満点のテストでそれぞれの生徒が$(10, 10, 10, 10, 10, 100)$のような点数をとったとします。
点数を見ると、とても難しいテストでただ１人だけの生徒がうまく全部解き切ることができた状態です。これの平均値を求めてみると、

$$\frac{10+10+10+10+10+100}{6} = 25$$

となってしまい、大半の生徒がテストの４分の１以上解けているかのように錯覚する値になってしまいます。これの中央値をとると10なので、そのような紛らわしさを起こす心配がありません。

MEMO

データ分析

例えばですが、ある平均値$\bar{x} = 10$のデータ集合に対して標準偏差$s = 5$だとします。この時、$10 - 5 \leq x \leq 10 + 5$の範囲内に全体の68%が含まれていると、統計的には考えられています。
このように標準偏差を計算することにより、データのばらつき度合いからどの値の範囲にデータの大部分が存在しているかについて、おおよそ知ることができます。

MEMO

統計学の主成分分析

主成分分析では$n \times n$の多次元データから、固有値の大きいものから順に選んでいった固有ベクトルを掛け合わせることで、分散がなるべく大きくなる方向に要素を並べた行列を生成し、次元数を削除する手法を用います。古典的な手法ですが次元削除のために機械学習の分野では頻繁に使われています。

な

MEMO

二項係数のもとと一致

例えば $(x-1)^2$ は x^2+2x+1 と展開できます。本書のP.254～255での係数は1、2、1となります。これを $(x-1)^n$ の場合に一般化されたものを二項係数と呼びます。これのi番目の係数は ${}_n\mathrm{C}_i$ で求めることができます

MEMO

二項分布

二項分布は、以下で表される関数$P(n)$に従った分布のことを言います。

$$P(n) = {}_n\mathrm{C}_k p^k (1-p)^{n-k}$$

n、pがパラメータとなっており、統計学においてはそれぞれ、試行の回数、事象の起こる確率として設定されることが多いです。また、${}_n\mathrm{C}_k$は組み合わせの数を表しており、以下で計算できます。

$${}_n\mathrm{C}_k = \frac{n!}{k!(n-k)!}$$

MEMO

ネイピア数e

ネイピア数eというのはe^xを何回xについて微分しても同じe^xが出てくるように調整された値のことを指します。
定義上は$e = \lim_{t \to 0}(1+t)^{\frac{1}{t}}$となっています。この性質のおかげで数学においては様々な場面で利用されており、非常に重要な数となっています。

は

MEMO

罰則項を損失関数に付け加える

線形回帰ではパラメータの絶対値の和を損失関数に加えたLasso回帰と2乗和を加えたRidge回帰が有名な手法として存在します。

MEMO

ヒープ木について

ヒープ木はいわゆる木構造の一種で、親要素よりも子要素の値が等しいか大きい（小さい）制約を持ったもので、特に大きい（小さい）値が親に来るものを最大（最小）ヒープ構造と呼びます。

MEMO

標本分散

与えられたデータそのものから求めた分散を不偏分散を区別する意味で特に標本分散と呼ぶことがあります。

MEMO

浮動小数点数について

浮動小数点数というのはある値についてそれを符号、仮数部、指数部の3つの要素で表現されたもののことを言います。仮数部はその値の有効桁数分の値、指数部がその値の大きさ（オーダー）を表す部分となっておりコンピュータ上ではこの形に表現された値を使って演算することが多いです。

MEMO

不偏標準偏差

不偏標準偏差は、母集団からいくつか抽出されたデータがある場合、手元のデータから母集団の標準偏差の推測を行う時に使われます。
標準偏差との実際の違いとしては分数部分である$\frac{1}{N}$が$\frac{1}{N-1}$になるだけではありますが、統計上は重要な意味合いを持っています。
Nが充分に大きい時にはほとんど差がないので、分母をNに近似して不偏標準偏差とすることもあります。

MEMO

不偏分散

不偏分散は不偏標準偏差と同様に母集団から抽出されたいくつかのデータから母集団の分散を推定する際に使われる値となります。こちらも同様に分散（V）を求める式の$\frac{1}{N}$を$\frac{1}{N-1}$にすることで求められます。

MEMO

ベータ分布

ベータ分布は、以下で表される確率密度関数$f(x)$に従った分布で、xの範囲は$0 \leqq x \leqq 1$で与えらます。

$$f(x) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha,\beta)}$$

$$B(\alpha,\beta) = \int_1^0 x^{\alpha-1}(1-x)^{\beta-1}dx$$

α、βはパラメータ

MEMO

ポアソン分布

ポアソン分布は、以下に従った確率分布のことを言います。これは、ある一定期間の間に平均λ回起こるような事象がその期間の中でk回起こる確率を示しています。

$$P(k) = e^{-\lambda}\frac{\lambda^k}{k!}$$

ら

MEMO

離散的な高階微分の求め方

P.255 で解説しているような離散的な微分の階数を増やしたものを求める時は1つ下の階層の前後の微分の差をとることで求めることができます。ひたすら離散的な微分を繰り返していくことになります。