

ペイ
全(金)

C.N
元田松本

ペ
アル
近年
論に
ター書で
研基
なふ
ベク相
機相
リッ
さら
MC
触れ
こと

セ
注
広
次
ツ
う
論
に
たす



Translation from the English language edition:
Pattern Recognition and Machine Learning
by Christopher M. Bishop
Copyright © 2006 Springer-Verlag New York, LLC
Springer is a part of Springer Science+Business Media
All Rights Reserved

This book is dedicated to my family:

Jenna, Mark, and Hugh



Total eclipse of the sun, Antalya, Turkey, 29 March 2006.

本書について

本書には、講義資料や本書で使われたすべての図表など、多数の追加資料が用意されている。これらについての最新情報を得るには次の Web ページを参照されたい。

<http://research.microsoft.com/~cmbishop/PRML>

本書では、各章末の演習問題も重視している。問題には、本文で説明した概念を発展させたり、新たな手法を開発したり、手法を一般化するのに役立つようなものを注意深く選んだ。問題には難易度も示し、(基本)は数分で解けるような簡単なもの、(難問)は非常に複雑な演習を示している。

演習問題の解答を、どれくらい入手しやすくすべきかを決めるのは難しい。本書で独学する読者には解答はとても役立つだろう。だが、本書を教科書とする講師にとっては、演習問題を講義で利用できるように、解答は出版社から取り寄せられるようにしておく方が良いであろう。こうした相反する要求に応じるようにするために、本文の重要な点を拡充するのに役立つ演習問題や、重要な細部を補足するような問題についてのみ、本書の Web サイトから解答を PDF ファイルで入手できるようにした。こうした演習問題は [www](#) で示した。他の演習問題の解答は、出版社に連絡すれば(詳細は Web ページを参照)、講師には入手できるようになる。だが、読者には、すべて独立でこれらの演習問題を解き、必要なときにのみ解答を見るようにすることを強く薦める。

本書は概念的・原理的な事柄を中心に執筆した。だが、できれば学生は適当なデータ集合を用いて、主なアルゴリズムのいくつかを実験してみるとよい。本書で示したほとんどのアルゴリズムを Matlab で実装したソフトウェアと、例題用データ集合は Web サイトから入手できるようにする。また、これらは、機械学習に現れる最適化問題を解く実用的アルゴリズムについての姉妹書 (Bishop and Nabney, 2008) にも収録する予定である。

数式の表記

本書では、数学的な内容は、この分野を正しく理解するのに必要最小限に留めた。だが、この最小限は 0 ではなく、最近のパターン認識や機械学習を明確に理解するには、微積分、線形代数、確率論を、十分に把握しておくことは必須である。だが、本書では、数学的な厳密さより、背後の概念を説明することを重視している。

本書を通して一貫した表記を用いるように努めた。そのため、該当する研究分野で使われる一般的な表記とは、しばしば違ったものとなることもある。ベクトルは、 \mathbf{x} などの太字のローマン体小文字で記し、すべてのベクトルは列ベクトルと仮定する。上付きの T は、行列やベクトルの転置を表す。よって、 \mathbf{x}^T は行ベクトルになる。 \mathbf{M} などの太字のローマン体大文字は行列を表す。 (w_1, \dots, w_M) は M 要素の行ベクトルであり、これに対応する列ベクトルは $\mathbf{w} = (w_1, \dots, w_M)^T$ と書く。

$[a, b]$ の表記は、 a から b への閉区間、すなわち、 a と b の値を含む区間を記すのに用いる。一方、 (a, b) は開区間、すなわち、 a や b は含まない区間を記す。同様に、 $[a, b)$ は、 a は含むが b は含まない区間を表す。しかし、ほとんどの場合、区間の端点を含むかどうかといった細部について、あまり考える必要はない。

$M \times M$ の単位行列は \mathbf{I}_M と記し、次元数が曖昧でなければ \mathbf{I} と略記する。この行列の要素 I_{ij} は、 $i = j$ なら 1 で、 $i \neq j$ なら 0 である。

$y(x)$ を関数として、汎関数を $f[y]$ と記す。汎関数の概念については付録 D で述べる。

$g(x) = O(f(x))$ は、 $|f(x)/g(x)|$ が、 $x \rightarrow \infty$ で有界であることを示す。例えば、 $g(x) = 3x^2 + 2$ なら、 $g(x) = O(x^2)$ である。

関数 $f(x, y)$ の、確率変数 x についての期待値を、 $\mathbb{E}_x[f(x, y)]$ と記す。どの変数で期待値をとるのかが曖昧でないときは、添え字を省略して、 $\mathbb{E}[x]$ などと簡略化する。 x の分布が、別の変数 z で条件付けされているなら、このときの条件付き期待値は $\mathbb{E}_x[f(x)|z]$ と書く。同様に、分散は $\text{var}[f(x)]$ と記し、ベクトル変数に対する共分散は $\text{cov}[\mathbf{x}, \mathbf{y}]$ と記す。また、 $\text{cov}[\mathbf{x}, \mathbf{x}]$ を短くした表記として $\text{cov}[\mathbf{x}]$ も用いる。期待値や共分散については、1.2.2 節で紹介する。

D 次元ベクトル $\mathbf{x} = (x_1, \dots, x_D)^T$ が、 $\mathbf{x}_1, \dots, \mathbf{x}_N$ のように N 個あるとき、これらの観測値を、第 n 行が、行ベクトル \mathbf{x}_n^T となるようなデータ行列 \mathbf{X} にまとめることができる。よって、 \mathbf{X} の n, i 要素は、第 n 観測値 \mathbf{x}_n の、第 i 要素に該当する。1 次元変数の場合では、このような行列は \mathbf{x} と記す。これは、第 n 要素が x_n であるような列ベクトルである。なお、(次元数が D の) \mathbf{x} と区別するため、(次元数が N の) \mathbf{x} には違った書体を用いる。

謝辞

まず最初に、本書の図表や L^AT_EX での組版の準備に多大な貢献をしてくれた Markus Svensén に心からの感謝を示したい。彼の手助けは計りしえないのであった。

また、非常に刺激的な研究環境を提供し、本書を執筆できるよう取りはからってくれた Microsoft Research 社に謝意を表す（しかし、本書の立場や意見は私自身のもので、したがってそれらは必ずしも Microsoft 社やその関係団体のそれと同じではない）。

Springer 社は、本書の執筆の最終段階を通して、すばらしい援助をしてくれた。担当編集者 John Kimmel には、彼の支援とプロ精神に対して、Joseph Piliero には、本書の表紙と体裁への手助けに対して、MaryAnn Brickner には、制作段階での多大な貢献に対して感謝したい。表紙のデザインは、Antonio Criminisi との議論に触発されたものである。

以前の教科書 *Neural Networks for Pattern Recognition* (Bishop, 1995) からの抜粋を許可してくれた Oxford University Press 社にも感謝したい。Mark 1 パーセプトロンと Frank Rosenblatt の画像は、Arvin Calspan Advanced Technology Center の許可を得て掲載した。図 13.1 のスペクトル図を描いてくれた Asela Gunawardana と、図 12.7 を描くためにカーネル PCA のコードを利用させてくれた Bernhard Schölkopf にも感謝したい。

また、本書の予稿を閲読し、助言や提言をしてくれた次の方々のお名前を挙げておきたい。Shivani Agarwal, Cédric Archambeau, Arik Azran, Andrew Blake, Hakan Cevikalp, Michael Fourman, Brendan Frey, Zoubin Ghahramani, Thore Graepel, Katherine Heller, Ralf Herbrich, Geoffrey Hinton, Adam Johansen, Matthew Johnson, Michael Jordan, Eva Kalyvianaki, Anitha Kannan, Julia Lasserre, David Liu, Tom Minka, Ian Nabney, Tonatiuh Pena, Yuan Qi, Sam Roweis, Balaji Sanjiya, Toby Sharp, Ana Costa e Silva, David Spiegelhalter, Jay Stokes, Tara Symeonides, Martin Szummer, Marshall Tappen, Ilkay Ulusoy, Chris Williams, John Winn, Andrew Zisserman。

最後に、本書の執筆に費やした数年間を通じて大きな支えとなってくれた妻 Jenna にお礼を述べたい。

2006 年 2 月ケンブリッジにて C.M. ビショップ

◆ 日本語版についての補足 ◆

原著の Web ページとは別に、日本語版にも Web ページを用意した。

<http://ibisforest.org/index.php?PRML>

ここには、主に、日本語版の書誌情報や正誤表を掲載する。講義資料や演習問題などの解答は、原著の Web ページを参照されたい。

校正には十分努めたが、もし誤りを見つけた場合は

prml@ibisforest.org

まで、お知らせいただきたい。なお、本アドレスは、演習問題の解答や、質問等については扱わないのでご留意いただきたい。また、原著は 1 卷本だが、日本語版は 2 分冊とし、原著の 1~5 章と付録は上巻に、6~14 章は下巻に収録した。

目 次

第 6 章 カーネル法	1
6.1 双対表現	2
6.2 カーネル関数の構成	4
6.3 RBF ネットワーク	10
6.3.1 Nadaraya-Watson モデル	12
6.4 ガウス過程	14
6.4.1 線形回帰再訪	15
6.4.2 ガウス過程による回帰	17
6.4.3 超パラメータの学習	22
6.4.4 関連度自動決定	23
6.4.5 ガウス過程による分類	25
6.4.6 ラプラス近似	27
6.4.7 ニューラルネットワークとの関係	31
演習問題	31
第 7 章 疎な解を持つカーネルマシン	35
7.1 最大マージン分類器	35
7.1.1 重なりのあるクラス分布	41
7.1.2 ロジスティック回帰との関係	47
7.1.3 多クラス SVM	48
7.1.4 回帰のための SVM	50
7.1.5 計算論的学習理論	54
7.2 関連ベクトルマシン	56
7.2.1 回帰問題に対する RVM	56
7.2.2 疎性の解析	60
7.2.3 分類問題に対する RVM	64

演習問題	68
第8章 グラフィカルモデル	71
8.1 ベイジアンネットワーク	72
8.1.1 例：多項式曲線フィッティング	74
8.1.2 生成モデル	77
8.1.3 離散変数	78
8.1.4 線形ガウスモデル	82
8.2 条件付き独立性	84
8.2.1 3つのグラフの例	85
8.2.2 有向分離 (D 分離)	90
8.3 マルコフ確率場	96
8.3.1 条件付き独立性	96
8.3.2 分解特性	98
8.3.3 例：画像のノイズ除去	100
8.3.4 有向グラフとの関係	104
8.4 グラフィカルモデルにおける推論	107
8.4.1 連鎖における推論	108
8.4.2 木	112
8.4.3 因子グラフ	113
8.4.4 積和アルゴリズム	116
8.4.5 max-sum アルゴリズム	126
8.4.6 一般のグラフにおける厳密推論	131
8.4.7 ループあり確率伝播	132
8.4.8 グラフ構造の学習	134
演習問題	134
第9章 混合モデルと EM	139
9.1 K-means クラスタリング	140
9.1.1 画像分割と画像圧縮	144
9.2 混合ガウス分布 (Mixtures of Gaussians)	146
9.2.1 最尤推定	149
9.2.2 混合ガウス分布の EM アルゴリズム	151
9.3 EM アルゴリズムのもう一つの解釈	155
9.3.1 混合ガウス分布再訪	157
9.3.2 K-means との関連	159
演習問題	160
9.3.3 混合ベルヌーイ分布	164
9.3.4 ベイズ線形回帰に関する EM アルゴリズム	165
9.4 一般の EM アルゴリズム	171
演習問題	171
第10章 近似推論法	175
10.1 変分推論	176
10.1.1 分布の分解	177
10.1.2 分解による近似のもつ性質	180
10.1.3 例：一変数ガウス分布	184
10.1.4 モデル比較	187
10.2 例：変分混合ガウス分布	187
10.2.1 変分事後分布	189
10.2.2 変分下限	195
10.2.3 予測分布	196
10.2.4 混合要素数の決定	197
10.2.5 導出された分解	199
10.3 変分線形回帰	200
10.3.1 変分分布	201
10.3.2 予測分布	203
10.3.3 変分下限	203
10.4 指数型分布族	204
10.4.1 変分メッセージパッシング	206
10.5 局所的変分推論法	207
10.6 変分ロジスティック回帰	212
10.6.1 変分事後分布	213
10.6.2 変分パラメータの最適化	215
10.6.3 超パラメータの推論	216
10.7 EP 法	219
10.7.1 例：雑音データ問題	225
10.7.2 グラフィカルモデルと EP 法	227
演習問題	231
第11章 サンプリング法	237
11.1 基本的なサンプリングアルゴリズム	239
11.1.1 標準的な分布	240

11.1.2 棄却サンプリング	242
11.1.3 適応的棄却サンプリング	244
11.1.4 重点サンプリング	246
11.1.5 SIR	249
11.1.6 サンプリングと EM アルゴリズム	250
11.2 マルコフ連鎖モンテカルロ	252
11.2.1 マルコフ連鎖	253
11.2.2 Metropolis-Hastings アルゴリズム	255
11.3 ギブスサンプリング	257
11.4 スライスサンプリング	261
11.5 ハイブリッドモンテカルロアルゴリズム	263
11.5.1 力学系	263
11.5.2 ハイブリッドモンテカルロアルゴリズム	267
11.6 分配関数の推定	269
演習問題	271
第 12 章 連続潜在変数	275
12.1 主成分分析	277
12.1.1 分散最大化による定式化	277
12.1.2 誤差最小化による定式化	279
12.1.3 主成分分析の応用	282
12.1.4 高次元データに対する主成分分析	285
12.2 確率的主成分分析	286
12.2.1 最尤法による主成分分析	290
12.2.2 EM アルゴリズムによる主成分分析	294
12.2.3 ベイズ的主成分分析	297
12.2.4 因子分析	302
12.3 カーネル主成分分析	304
12.4 非線形潜在変数モデル	308
12.4.1 独立成分分析	309
12.4.2 自己連想ニューラルネットワーク	310
12.4.3 非線形多様体のモデル化	313
演習問題	318
第 13 章 系列データ	323
13.1 マルコフモデル	324

13.2 隠れマルコフモデル	328
13.2.1 HMM の最尤推定	333
13.2.2 フォワード-バックワードアルゴリズム	336
13.2.3 HMM の積和アルゴリズム	343
13.2.4 スケーリング係数	345
13.2.5 Viterbi アルゴリズム	347
13.2.6 隠れマルコフモデルの拡張	349
13.3 線形動的システム	353
13.3.1 LDS における推論	356
13.3.2 LDS の学習	360
13.3.3 LDS の拡張	362
13.3.4 粒子フィルタ	364
演習問題	365
第 14 章 モデルの結合	371
14.1 ベイズモデル平均化	372
14.2 コミッティ	373
14.3 ブースティング	374
14.3.1 指数誤差の最小化	377
14.3.2 ブースティングのための誤差関数	379
14.4 木構造モデル	380
14.5 条件付き混合モデル	384
14.5.1 線形回帰モデルの混合	384
14.5.2 ロジスティックモデルの混合	387
14.5.3 混合エキスペートモデル	390
演習問題	392
下巻のための参考文献	395
訳者あとがき	405
和文索引	408
英文索引	421

上巻の目次

第1章 序論

§1.1 例：多項式曲線フィッティング／§1.2 確率論／§1.3 モデル選択／§1.4 次元の呪い／§1.5 決定理論／§1.6 情報理論／演習問題

第2章 確率分布

§2.1 二値変数／§2.2 多値変数／§2.3 ガウス分布／§2.4 指数型分布族／§2.5 ノンパラメトリック法／演習問題

第3章 線形回帰モデル

§3.1 線形基底関数モデル／§3.2 バイアス-バリアンス分解／§3.3 ベイズ線形回帰／§3.4 ベイズモデル比較／§3.5 エビデンス近似／§3.6 固定された基底関数の限界／演習問題

第4章 線形識別モデル

§4.1 識別関数（判別関数）／§4.2 確率的生成モデル／§4.3 確率的識別モデル／§4.4 ラプラス近似／§4.5 ベイズロジスティック回帰／演習問題

第5章 ニューラルネットワーク

§5.1 フィードフォワードネットワーク関数／§5.2 ネットワーク訓練／§5.3 誤差逆伝播／§5.4 ヘッセ行列／§5.5 ニューラルネットワークの正則化／§5.6 混合密度ネットワーク／§5.7 ベイズニューラルネットワーク／演習問題

付録 A データ集合

付録 B 確率分布の一覧

付録 C 行列の性質

付録 D 変分法

付録 E ラグランジュ乗数

上巻のための参考文献／和文索引／英文索引

第6章 カーネル法

3章と4章では、回帰と分類のための線形なパラメトリックモデルを考えた。これは、入力 \mathbf{x} から出力 y への写像 $y(\mathbf{x}, \mathbf{w})$ をパラメータベクトル \mathbf{w} によって関連付けるような形式をしたものであった。訓練時には、訓練データはパラメータベクトルの点推定値、あるいは、パラメータベクトルの事後分布を得るために利用される。訓練後には、訓練データは捨てられ、新しい入力に対する予測は、学習済みのパラメータベクトル \mathbf{w} のみを用いて行われる。このアプローチはニューラルネットワーク（図5章）のような非線形のパラメトリックモデルにおいても同様である。

しかしながら、訓練データ点の全部あるいは一部を、予測時にも利用するようなパターン認識法のクラスが存在する。例えば Parzen 推定法（図2.5.1節）では、モデルが各訓練データ点を中心とするような「カーネル関数」の線形和として表現される。2.5.2節で紹介した最近傍法と呼ばれる単純な分類手法では、新しいテスト点は、訓練データの中で最も近いサンプルと同じラベルが割り当てられる。これらは、メモリベース法 (memory-based method) と呼ばれるものの例であり、すべての訓練データを予測時まで保存しておく必要がある。通常、これらの方では、入力空間における任意の2つのベクトルの類似度を測る指標が必要になる。また、一般に、「訓練」は高速に行うことができる反面、テスト点に対する予測には時間がかかる傾向がある。

多くのパラメトリックな線形モデルは、同値な「双対表現」の形に書き直すことができ、予測もまた、訓練データ点を中心として定義されるカーネル関数（核関数; kernel function）の線形結合を用いて行われる。後に見るように、あらかじめ定義された非線形の特徴空間 (feature space) への写像 $\phi(\mathbf{x})$ に基づくモデルにおいて、カーネル関数は、以下の関係によって与えられる。

$$k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^T \phi(\mathbf{x}'). \quad (6.1)$$

この定義から、カーネル関数は、その引数について対称、すなわち $k(\mathbf{x}, \mathbf{x}') = k(\mathbf{x}', \mathbf{x})$ であることがわかる。カーネル法の考え方には、Aizerman *et al.* (1964) において、静電気学におけるポテンシャル関数とのアナロジーから、パターン認識の分野に導入され

た。その後長年にわたり忘れていたが、Boser *et al.* (1992)において最大マージン分類器の文脈で機械学習の分野に再び紹介され、その考え方はサポートベクトルマシン (support vector machine) (☞7章) に引き継がれることとなった。以来、このトピックは、理論と応用の両面から非常に注目され続けており、特に顕著な功績の1つとしては、カーネル法を文字列などの記号的オブジェクトを扱えるように拡張したことが挙げられる。これによって、カーネル法によって扱うことのできる問題の範囲は格段に広がったと言える。

最も簡単なカーネル関数の定義の例は、特徴空間を恒等写像に取ること、つまり、 $\phi(\mathbf{x}) = \mathbf{x}$ のように取ることである。このとき、カーネル関数は $k(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T \mathbf{x}'$ となる。これを線形カーネルと呼ぶことにする。

カーネル関数を、特徴空間における内積として捉えることで、カーネルトリック (kernel trick) あるいはカーネル置換 (kernel substitution) と呼ばれるテクニックを用いて、多くのよく知られたアルゴリズムを拡張することができるようになる。その一般的な考え方は、もし学習アルゴリズムにおいて、入力ベクトル \mathbf{x} が、スカラー積の中にのみ現れるならば、スカラー積を何らかのカーネル関数で置き換えることができるというものである。例えば、カーネル置換を主成分分析に適用することで、非線形版の主成分分析 (☞12.3節) (Schölkopf *et al.*, 1998) を導くことができる。他の例としては、最近傍分類器や、カーネルフィッシャー判別 (Mika *et al.*, 1999; Roth and Steinhage, 2000; Baudat and Anouar, 2000) のカーネル版などが挙げられる。

よく使われるカーネル関数には多くの種類があるが、そのうちのいくつかについては、この章で紹介することになるであろう。多くのカーネル関数は、引数の差にのみ依存したもの、つまり、 $k(\mathbf{x}, \mathbf{x}') = k(\mathbf{x} - \mathbf{x}')$ の形をもったもので、いくつかの入力空間の変換に関して不变であることから、不変カーネル (stationary kernel) と呼ばれる。一方、 $k(\mathbf{x}, \mathbf{x}') = k(\|\mathbf{x} - \mathbf{x}'\|)$ のように、2つの入力ベクトルの間の距離（主にユークリッド距離）にのみ依存するカーネルは、均一カーネル (homogeneous kernel) あるいは、RBF (動径基底関数; radial basis function) (☞6.3節) と呼ばれる。

最近のカーネル法についての教科書としては、Schölkopf and Smola (2002) や Herbrich (2002)，あるいは Shawe-Taylor and Cristianini (2004) などがある。

6.1 双対表現

回帰や分類に用いられる多くの線形モデルは、双対表現で表すことによって、カーネル関数が自然に現れてくる。この考え方は、次の章で扱うサポートベクトルマシンにおいて重要な役割を持つ。ここでは、線形回帰モデルで、パラメータが以下のような正則化された二乗和誤差を最小化することで求められるようなものを考える。

$$J(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{\mathbf{w}^T \phi(\mathbf{x}_n) - t_n\}^2 + \frac{\lambda}{2} \mathbf{w}^T \mathbf{w}. \quad (6.2)$$

なお、 $\lambda \geq 0$ であるとする。 $J(\mathbf{w})$ の \mathbf{w} についての勾配を零とおくと、 \mathbf{w} は、以下のように、係数が \mathbf{w} の関数であるような、ベクトル集合 $\phi(\mathbf{x}_n)$ の線形結合の形になることがわかる。

$$\mathbf{w} = -\frac{1}{\lambda} \sum_{n=1}^N \{\mathbf{w}^T \phi(\mathbf{x}_n) - t_n\} \phi(\mathbf{x}_n) = \sum_{n=1}^N a_n \phi(\mathbf{x}_n) = \Phi^T \mathbf{a}. \quad (6.3)$$

ここで、 Φ は、 n 番目の行が $\phi(\mathbf{x}_n)^T$ で与えられるような計画行列である。また、

$$a_n = -\frac{1}{\lambda} \{\mathbf{w}^T \phi(\mathbf{x}_n) - t_n\} \quad (6.4)$$

として、そのベクトル形式を $\mathbf{a} = (a_1, \dots, a_N)^T$ とする。さて、パラメータベクトル \mathbf{w} を直接扱う代わりに、最小二乗法のアルゴリズムをパラメータベクトル \mathbf{a} で表現し直すことにする。これは、双対表現 (dual representation) と呼ばれる。 $\mathbf{w} = \Phi^T \mathbf{a}$ を $J(\mathbf{w})$ に代入すると、

$$J(\mathbf{a}) = \frac{1}{2} \mathbf{a}^T \Phi \Phi^T \Phi \Phi^T \mathbf{a} - \mathbf{a}^T \Phi \Phi^T \mathbf{t} + \frac{1}{2} \mathbf{t}^T \mathbf{t} + \frac{\lambda}{2} \mathbf{a}^T \Phi \Phi^T \mathbf{a} \quad (6.5)$$

を得る。ここで、 $\mathbf{t} = (t_1, \dots, t_N)^T$ であるとする。次に、 $N \times N$ の対称行列であり、その要素が、

$$K_{nm} = \phi(\mathbf{x}_n)^T \phi(\mathbf{x}_m) = k(\mathbf{x}_n, \mathbf{x}_m) \quad (6.6)$$

で定められるグラム行列 (Gram matrix) $\mathbf{K} = \Phi \Phi^T$ を定義する。なおここで、(6.1) で定義されるカーネル関数 (kernel function) $k(\mathbf{x}, \mathbf{x}')$ を使った。グラム行列を用いると、二乗和誤差関数は、

$$J(\mathbf{a}) = \frac{1}{2} \mathbf{a}^T \mathbf{K} \mathbf{K} \mathbf{a} - \mathbf{a}^T \mathbf{K} \mathbf{t} + \frac{1}{2} \mathbf{t}^T \mathbf{t} + \frac{\lambda}{2} \mathbf{a}^T \mathbf{K} \mathbf{a} \quad (6.7)$$

と書くことができる。

(6.3) を用いて (6.4) から \mathbf{w} を消去して \mathbf{a} について解くと、次の解が得られる。

$$\mathbf{a} = (\mathbf{K} + \lambda \mathbf{I}_N)^{-1} \mathbf{t}. \quad (6.8)$$

これを線形回帰モデルに代入し直すことによって、新しい入力 \mathbf{x} に対する予測は次のように与えられることがわかる。

$$y(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) = \mathbf{a}^T \Phi \phi(\mathbf{x}) = \mathbf{k}(\mathbf{x})^T (\mathbf{K} + \lambda \mathbf{I}_N)^{-1} \mathbf{t}. \quad (6.9)$$

ここで、 $\mathbf{k}(\mathbf{x})$ は要素 $k_n(\mathbf{x}) = k(\mathbf{x}_n, \mathbf{x})$ を持つベクトルである。以上より、双対表現に変換することによって、最小二乗法の解はカーネル関数 $k(\mathbf{x}, \mathbf{x}')$ のみによって表現できることがわかる。これが双対表現と呼ばれる理由は、 $\phi(\mathbf{x})$ の要素の線形結合によっ

て \mathbf{a} が表現できることから、パラメータベクトル \mathbf{w} を使った、もともとの定式化を復元できるためである(☞演習 6.1)。なお、 \mathbf{x} に対する予測は訓練データ集合に対する目標値の線形結合で与えられることに注意する。実際のところ、この結果は、3.3.3 節において、(若干異なる表記によってではあるが) すでに得られている。

双対表現においては、パラメータベクトル \mathbf{a} は、 $N \times N$ 行列の逆行列を求ることによって得られるが、一方、もともとのパラメータ空間における定式化では、 $M \times M$ 行列の逆行列を求ることによって \mathbf{w} が得られる。通常、 N は M よりもずっと大きいため、双対表現はあまり有用ではないように思えるかもしれない。しかしながら、後に見るように、双対表現の意義は、すべてがカーネル関数 $k(\mathbf{x}, \mathbf{x}')$ で表現されることにある。常にカーネル関数を通じて問題を扱うことができるために、特徴ベクトル $\phi(\mathbf{x})$ を明示的に考えることを避け、高次元の、時には無限次元の特徴空間を間接的に扱うことができるようになるのである。

グラム行列に基づく双対表現は、パーセプトロンなど、多くの線形モデルに対して存在する(☞演習 6.2)。6.4 節では、回帰のための確率的な線形モデルと、ガウス過程との双対性を示す。また、双対性は 7 章で議論するサポートベクトルマシンにおいても重要な役割を果たすことになる。

6.2 カーネル関数の構成

実際にカーネル置換を行うためには、カーネル関数として有効なものを構成する必要がある。ひとつ的方法は、図 6.1 に示したように、特徴空間への写像 $\phi(\mathbf{x})$ を考え、これをもとに、対応するカーネルを構成することである。この例では、カーネル関数は次のように一次元の入力空間において定義されている。

$$k(x, x') = \phi(x)^T \phi(x') = \sum_{i=1}^M \phi_i(x) \phi_i(x'). \quad (6.10)$$

ここで、 $\phi_i(x)$ は基底関数である。

別のアプローチとして、カーネル関数を直接定義することもできる。この場合、与えた関数がカーネル関数として有効であることを保証する必要がある。言い換えれば、(場合によっては無限の次元数をも持つ) ある特徴空間におけるスカラー積であることを保証する必要がある。簡単な例として次のカーネル関数を考えてみる。

$$k(\mathbf{x}, \mathbf{z}) = (\mathbf{x}^T \mathbf{z})^2. \quad (6.11)$$

仮に、2 次元の入力空間 $\mathbf{x} = (x_1, x_2)$ を考えると、上式を展開することで、対応する特徴空間への非線形写像を得ることができる。

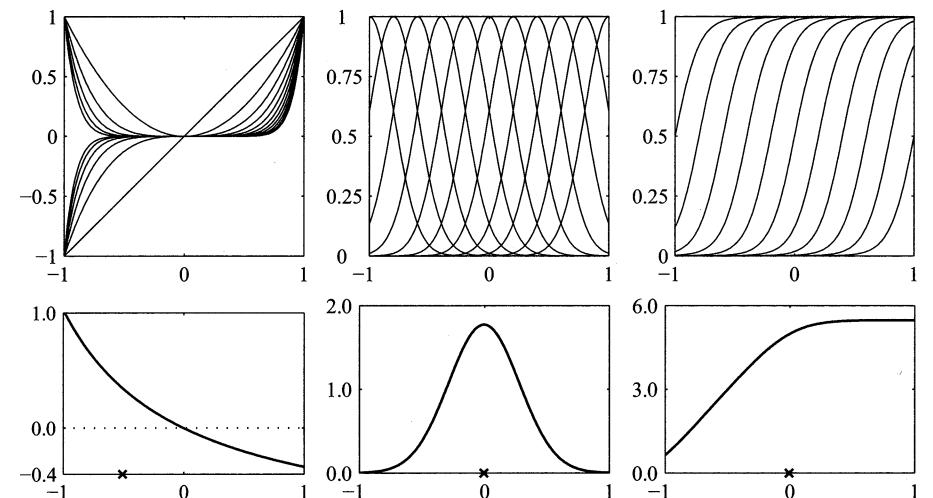


図 6.1 対応する基底関数の集合からカーネル関数を構成する様子を図示したもの。下段は、(6.10) で定義されるカーネル関数 $k(x, x')$ を、 x の関数としてプロットしたもの。ここで、 x' は \times で示されている。上段は、それぞれに対応する基底関数（左から、多項式関数、ガウス分布、ロジスティックモード関数）をプロットしたもの。

$$\begin{aligned} k(\mathbf{x}, \mathbf{z}) &= (\mathbf{x}^T \mathbf{z})^2 = (x_1 z_1 + x_2 z_2)^2 \\ &= x_1^2 z_1^2 + 2x_1 z_1 x_2 z_2 + x_2^2 z_2^2 \\ &= (x_1^2, \sqrt{2}x_1 x_2, x_2^2)(z_1^2, \sqrt{2}z_1 z_2, z_2^2)^T \\ &= \phi(\mathbf{x})^T \phi(\mathbf{z}). \end{aligned} \quad (6.12)$$

特徴空間への写像は $\phi(\mathbf{x}) = (x_1^2, \sqrt{2}x_1 x_2, x_2^2)^T$ の形を持ち、したがって、すべての 2 次の項を（適当な重み付きで）含むことがわかる。

しかしながら、より一般的には、 $\phi(\mathbf{x})$ を明示的に構成することなく、関数が有効なカーネルであるかどうかを簡単に調べる方法が望まれる。関数 $k(\mathbf{x}, \mathbf{x}')$ が有効なカーネルであるための必要十分条件は、任意の $\{\mathbf{x}_n\}$ に対して、要素が $k(\mathbf{x}_n, \mathbf{x}_m)$ で与えられるグラム行列 \mathbf{K} が半正定値であることである (Shawe-Taylor and Cristianini, 2004)。なお、行列が半正定値であることと、行列のすべての要素が非負であることとは異なることに注意する(☞付録C)。

新たなカーネルを構築するための便利な方法は、より単純なカーネルを構成要素として用いることである。これには、次の性質を利用することができます。

◆ 新たなカーネルを構築するための方法 ◆

有効なカーネルとして $k_1(\mathbf{x}, \mathbf{x}')$ と $k_2(\mathbf{x}, \mathbf{x}')$ が与えられたとき、次の関数もやはりカーネル関数として有効である。

$$k(\mathbf{x}, \mathbf{x}') = ck_1(\mathbf{x}, \mathbf{x}')$$
(6.13)

$$k(\mathbf{x}, \mathbf{x}') = f(\mathbf{x})k_1(\mathbf{x}, \mathbf{x}')f(\mathbf{x}')$$
(6.14)

$$k(\mathbf{x}, \mathbf{x}') = q(k_1(\mathbf{x}, \mathbf{x}'))$$
(6.15)

$$k(\mathbf{x}, \mathbf{x}') = \exp(k_1(\mathbf{x}, \mathbf{x}'))$$
(6.16)

$$k(\mathbf{x}, \mathbf{x}') = k_1(\mathbf{x}, \mathbf{x}') + k_2(\mathbf{x}, \mathbf{x}')$$
(6.17)

$$k(\mathbf{x}, \mathbf{x}') = k_1(\mathbf{x}, \mathbf{x}')k_2(\mathbf{x}, \mathbf{x}')$$
(6.18)

$$k(\mathbf{x}, \mathbf{x}') = k_3(\phi(\mathbf{x}), \phi(\mathbf{x}'))$$
(6.19)

$$k(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T \mathbf{A} \mathbf{x}'$$
(6.20)

$$k(\mathbf{x}, \mathbf{x}') = k_a(\mathbf{x}_a, \mathbf{x}'_a) + k_b(\mathbf{x}_b, \mathbf{x}'_b)$$
(6.21)

$$k(\mathbf{x}, \mathbf{x}') = k_a(\mathbf{x}_a, \mathbf{x}'_a)k_b(\mathbf{x}_b, \mathbf{x}'_b).$$
(6.22)

ここで、 $c > 0$ は定数であり、 $f(\cdot)$ は任意の関数、 $q(\cdot)$ は非負の係数をもつ多項式、 $\phi(\mathbf{x})$ は \mathbf{x} から \mathbb{R}^M への関数、 $k_3(\cdot, \cdot)$ は \mathbb{R}^M で定義された有効なカーネル、 \mathbf{A} は対称な半正定値行列、 \mathbf{x}_a と \mathbf{x}_b は $\mathbf{x} = (\mathbf{x}_a, \mathbf{x}_b)$ であるような変数（必ずしも互いに素である必要はない）、また、 k_a と k_b はそれぞれの特徴空間において有効なカーネル関数であるとする。

これらの性質を用いると、特定の応用先に適した、より複雑なカーネルを構成することが可能になる。なお、カーネル $k(\mathbf{x}, \mathbf{x}')$ は、対称で、半正定値であり、また、適用先の問題領域における \mathbf{x} と \mathbf{x}' の適切な類似度となっていることが必要である。ここでは、いくつかのよく使われるカーネル関数を紹介する。より詳細な「カーネル設計」の方法については、Shawe-Taylor and Cristianini (2004) を参照するとよい。

単純な多項式カーネル $k(\mathbf{x}, \mathbf{x}') = (\mathbf{x}^T \mathbf{x}')^2$ は 2 次の項のみを含んでいたが、少し一般化して、定数 $c > 0$ を用いて $k(\mathbf{x}, \mathbf{x}') = (\mathbf{x}^T \mathbf{x}' + c)^2$ のようなカーネルを考えると、対応する特徴空間への写像 $\phi(\mathbf{x})$ が、定数の項と、1 次の項も持つようになります。また、 $k(\mathbf{x}, \mathbf{x}') = (\mathbf{x}^T \mathbf{x}')^M$ は M 次の項すべてを持つ。例えば、 \mathbf{x} と \mathbf{x}' を 2 つの画像とすると、このカーネル関数は、片方の画像中のあらゆる M 個の組み合わせのピクセルと、もう片方の画像中の M 個のピクセルとの積の、ある重み付き和となる。これも同様に、 $c > 0$ を用いて $k(\mathbf{x}, \mathbf{x}') = (\mathbf{x}^T \mathbf{x}' + c)^M$ とすることで、 M 次までのすべての次数の項を含むように一般化することができる。(6.17) と (6.18) の結果を利用すれば、これらはすべて有効なカーネル関数であることがわかる。

もうひとつのよく使われるカーネル関数としては、以下の、ガウスカーネルと呼ばれるものがある。

$$k(\mathbf{x}, \mathbf{x}') = \exp(-\|\mathbf{x} - \mathbf{x}'\|^2/2\sigma^2).$$
(6.23)

ただし、カーネル法の文脈では、これは確率密度関数としては解釈されず、したがって、正規化のための定数は省かれていることに注意する。これが有効なカーネルであることは、括弧内の平方を、

$$\|\mathbf{x} - \mathbf{x}'\|^2 = \mathbf{x}^T \mathbf{x} + (\mathbf{x}')^T \mathbf{x}' - 2\mathbf{x}^T \mathbf{x}'$$
(6.24)

のように展開したものを利用して、次の変形

$$k(\mathbf{x}, \mathbf{x}') = \exp(-\mathbf{x}^T \mathbf{x}/2\sigma^2) \exp(\mathbf{x}^T \mathbf{x}'/\sigma^2) \exp(-(\mathbf{x}')^T \mathbf{x}'/2\sigma^2)$$
(6.25)

を行い、さらに、(6.14) と (6.16)、および線形カーネル $k(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T \mathbf{x}'$ が有効であることを用いると示すことができる。なお、ガウスカーネルに対応する特徴ベクトルは無限次元である（☞演習 6.11）。

ガウスカーネルは、必ずしもユークリッド距離に限定されたものではなく、(6.24) のカーネル置換を用いて、 $\mathbf{x}^T \mathbf{x}'$ を非線形カーネル $\kappa(\mathbf{x}, \mathbf{x}')$ で置き換えれば、次のようなカーネルが得られる。

$$k(\mathbf{x}, \mathbf{x}') = \exp\left\{-\frac{1}{2\sigma^2}(\kappa(\mathbf{x}, \mathbf{x}) + \kappa(\mathbf{x}', \mathbf{x}') - 2\kappa(\mathbf{x}, \mathbf{x}'))\right\}.$$
(6.26)

カーネル法の考え方によって得られる重要な利点は、入力が実数値ベクトルだけではなく、記号であるような場合にも適用できることである。実際に、グラフ、集合、文字列、テキスト文書などのさまざまな対象に対して、カーネル関数が定義されている。例えば、対象として、ある集合を考え、この集合のすべての部分集合で構成される、ベクトル形式を持たない入力空間を定義する。 A_1 と A_2 をこのような部分集合とすると、カーネル関数のひとつの定義としては、以下のようなものが考えられる。

$$k(A_1, A_2) = 2^{|A_1 \cap A_2|}.$$
(6.27)

ここで、 $A_1 \cap A_2$ は A_1 と A_2 の共通集合とし、 $|A|$ を A に含まれる要素の数とする。これは、ある特徴空間における内積になっていることが示されるため、有効なカーネル関数である（☞演習 6.12）。

また、別の強力なアプローチとして、確率的生成モデルからカーネル関数を構成する方法がある (Haussler, 1999)。これによって、生成モデルを分類に用いることができるようになる。生成モデルは、欠損データを自然に扱うことができ、また、隠れマルコフモデル (hidden Markov model) などを用いれば可変長の配列を扱うことができる。一方、識別モデルは、分類問題においては生成モデルよりも一般に性能が良いことが知られており、したがって、これら 2 つのアプローチを組み合わせることは、しばしば興味の対象となる (Lasserre et al., 2006)。これを実現する方法のひとつとして考えられるのは、生成モデルを用いてカーネルを定義し、このカーネルを用いて識別アプローチをとるという方法である。

生成モデル $p(\mathbf{x})$ が与えられたとき、これを用いて、カーネルを以下のように定義することができる。

$$k(\mathbf{x}, \mathbf{x}') = p(\mathbf{x})p(\mathbf{x}'). \quad (6.28)$$

明らかに、これは写像 $p(\mathbf{x})$ で定義された 1 次元の特徴空間における内積として解釈できるため、有効なカーネル関数であることがわかる。このカーネルでは、2つの入力 \mathbf{x} と \mathbf{x}' の確率が共に大きいときに、2つの入力が似ているとみなされる。さらに、(6.13) と (6.17) を用いれば、複数の確率分布があるときに、確率分布の積の和を考えることによって、正の値を取る重み係数 $p(i)$ を用いて、以下のように拡張することができる。

$$k(\mathbf{x}, \mathbf{x}') = \sum_i p(\mathbf{x}|i)p(\mathbf{x}'|i)p(i). \quad (6.29)$$

これは全体を定数倍すれば、混合分布に等しい。混合要素を指定する i は「潜在」変数（☞9.2 節）であると解釈できる。2つの入力 \mathbf{x} と \mathbf{x}' が、 i が異なる複数の確率分布においても共に大きな確率をもつならば、カーネル関数の値が大きくなるため、2つの入力がよく似ていることになる。さらに、無限個の構成要素の和を考えることで、次のようなカーネルを考えることができる。

$$k(\mathbf{x}, \mathbf{x}') = \int p(\mathbf{x}|\mathbf{z})p(\mathbf{x}'|\mathbf{z})p(\mathbf{z}) d\mathbf{z}. \quad (6.30)$$

ここで、 \mathbf{z} は連続値をとる潜在変数である。

次に、データが長さ L の配列、つまり、入力変数が $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_L\}$ の形で与えられるような場合を考える。配列の生成モデルとしてよく使われるものとしては、隠れマルコフモデル（☞13.2 節）がある。隠れマルコフモデルは、 $p(\mathbf{X})$ の分布を、対応する隠れ変数 $\mathbf{Z} = \{\mathbf{z}_1, \dots, \mathbf{z}_L\}$ について周辺化したものとして表す。このアプローチを用いることで、混合表現 (6.29) を拡張して、2つの配列 \mathbf{X} と \mathbf{X}' の類似度を測るカーネル関数を定義することができる。

$$k(\mathbf{X}, \mathbf{X}') = \sum_{\mathbf{Z}} p(\mathbf{X}|\mathbf{Z})p(\mathbf{X}'|\mathbf{Z})p(\mathbf{Z}). \quad (6.31)$$

観測された2つの配列は、共通の隠れ変数の列 \mathbf{Z} から生成されたものと考える。このモデルは2つの配列の長さが異なる場合にも容易に拡張が可能である。

生成モデルを用いてカーネル関数を定義する別の方法としては、フィッシャーカーネル (Fisher kernel) (Jaakkola and Haussler, 1999) がよく知られている。パラメータベクトル θ を持つ、パラメトリックな生成モデル $p(\mathbf{x}|\theta)$ を考える。目的は、生成モデルから、2つの入力ベクトル \mathbf{x} と \mathbf{x}' の間の類似度を測るカーネルを見つけることである。Jaakkola and Haussler (1999) では、フィッシャースコア (Fisher score) と呼ばれる、 θ についての勾配

$$\mathbf{g}(\theta, \mathbf{x}) = \nabla_{\theta} \ln p(\mathbf{x}|\theta) \quad (6.32)$$

によって、 θ と同じ次元の特徴空間におけるベクトルを考え、フィッシャーカーネルを以下のように定義した。

$$k(\mathbf{x}, \mathbf{x}') = \mathbf{g}(\theta, \mathbf{x})^T \mathbf{F}^{-1} \mathbf{g}(\theta, \mathbf{x}'). \quad (6.33)$$

ここで、 \mathbf{F} はフィッシャー情報量行列 (Fisher information matrix) と呼ばれ、以下によって定義される行列である。

$$\mathbf{F} = \mathbb{E}_{\mathbf{x}} [\mathbf{g}(\theta, \mathbf{x}) \mathbf{g}(\theta, \mathbf{x})^T]. \quad (6.34)$$

期待値は、確率分布 $p(\mathbf{x}|\theta)$ の下で、 \mathbf{x} について取ったものである。これはモデルのパラメータ空間における微分幾何を考える情報幾何 (information geometry) (Amari, 1998) の考え方に基づいたものであるが、ここでは、フィッシャー情報量行列によって、カーネルが、確率密度モデルのパラメータの非線形な変換 $\theta \rightarrow \psi(\theta)$ について不变となるということにのみ言及しておく（☞演習 6.13）。

実際の利用においては、フィッシャー情報量行列を計算するのは不可能であるため、代わりに次のように期待値を単純平均で置き換えることが行われる。

$$\mathbf{F} \simeq \frac{1}{N} \sum_{n=1}^N \mathbf{g}(\theta, \mathbf{x}_n) \mathbf{g}(\theta, \mathbf{x}_n)^T. \quad (6.35)$$

これはフィッシャースコアの共分散行列であり、フィッシャーカーネルは、スコアの白色化に対応している。不变ではなくなるものの、より簡単に、フィッシャー情報量行列を省いて、次のように定義することもある。

$$k(\mathbf{x}, \mathbf{x}') = \mathbf{g}(\theta, \mathbf{x})^T \mathbf{g}(\theta, \mathbf{x}'). \quad (6.36)$$

フィッシャーカーネルの応用例としては、Hofmann (2000) による文書検索の例などがある。

カーネル関数の最後の例として、以下で定義されるシグモイドカーネルを挙げる。

$$k(\mathbf{x}, \mathbf{x}') = \tanh(a \mathbf{x}^T \mathbf{x}' + b). \quad (6.37)$$

シグモイドカーネルのグラム行列は必ずしも半正定値にはならないものの、実用的には、以前からよく使用されている (Vapnik, 1995)。その理由はおそらく、シグモイドカーネルを使うことで、サポートベクトルマシンとニューラルネットワークが表層的に類似したものになるためであろう。しかし、後に見るように、基底関数が無限にある場合には、適当な事前分布をもつベイズニューラルネットワークは、ガウス過程に一致し、ニューラルネットワークとカーネル法の、より深いつながりが明らかになる（☞6.4.7 節）。

6.3 RBF ネットワーク

3章では、あらかじめ固定された基底関数の線形結合として表されるような回帰モデルを考えたが、基底関数としてどのような形のものを取ればよいかという点については考えていなかった。一般的には、RBF（動径基底関数; radial basis function）と呼ばれる、基底関数が、その中心 μ_j からの動径（通常、ユークリッド距離が使われる）のみに依存して、 $\phi_j(\mathbf{x}) = h(\|\mathbf{x} - \mu_j\|)$ のような形式を持ったものがよく利用されている。

歴史的には、RBFが初めて導入されたのは関数補間(function interpolation)を正確に行なうためであった(Powell, 1987)。関数補間では、入力ベクトルの集合 $\{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ と、対応する目的変数の値の集合 $\{t_1, \dots, t_N\}$ が与えられたときに、 $n = 1, \dots, N$ について $f(\mathbf{x}_n) = t_n$ となるように、すなわち、目的変数の値を正確に再現することのできる滑らかな関数 $f(\mathbf{x})$ を求めることが目的である。これは、次のように、 $f(\mathbf{x})$ を各データ点を中心とした RBF の線形結合で表すことによって実現できる。

$$f(\mathbf{x}) = \sum_{n=1}^N w_n h(\|\mathbf{x} - \mathbf{x}_n\|). \quad (6.38)$$

係数 $\{w_n\}$ の値は、最小二乗法によって求めることができる。係数の数と制約の数が等しいため、結果として得られる関数はすべての目的変数の値を正確に再現する。しかしながら、パターン認識の応用においては、通常、目的変数の値にはノイズが含まれており、これを正確に再現しようとすることは、過学習の恐れがあるため必ずしも良いこととは限らない。

RBFを用いた展開は、正則化理論(Poggio and Girosi, 1990; Bishop, 1995a)においても現れる。微分作用素で定義された正則化項を持つ二乗和誤差関数に対して、最適解は作用素のグリーン関数(Green's function)によって展開される(グリーン関数は離散値行列の固有値に対応するものである)。ここでも各基底関数はそれぞれのデータ点を中心を持つ。微分作用素が等方的であるならば、グリーン関数の値は、対応するデータ点からの動径距離にのみ依存する。正則化項を導入することによって、解は、必ずしも訓練データの値を正確に再現するとは限らなくなる。

RBFを導入するもう1つの動機としては、(目的変数ではなく)入力変数にノイズが含まれる場合の補間の問題が挙げられる(Webb, 1994; Bishop, 1995a)。入力変数 \mathbf{x} に含まれるノイズが、確率分布 $\nu(\xi)$ に従う確率変数 ξ によって表されるとき、二乗和誤差関数は以下のようになる。

$$E = \frac{1}{2} \sum_{n=1}^N \int \{y(\mathbf{x}_n + \xi) - t_n\}^2 \nu(\xi) d\xi. \quad (6.39)$$

変分法(☞付録D)を用いることによって関数 $y(\mathbf{x})$ についての最適化を行うことができ、

次を得ることができる(☞演習 6.17)。

$$y(\mathbf{x}) = \sum_{n=1}^N t_n h(\mathbf{x} - \mathbf{x}_n). \quad (6.40)$$

なお、RBFは以下によって与えられる。

$$h(\mathbf{x} - \mathbf{x}_n) = \frac{\nu(\mathbf{x} - \mathbf{x}_n)}{\sum_{n=1}^N \nu(\mathbf{x} - \mathbf{x}_n)}. \quad (6.41)$$

このモデルは、すべてのデータ点において基底関数を持っていることがわかる。これは、Nadaraya-Watson モデルとして知られており、6.3.1節では別の導出によっても再び導かれる。ノイズ分布 $\nu(\xi)$ が等方的であるならば、 $\|\xi\|$ のみに依存する関数となるため、基底関数もまた放射状の関数となる。

基底関数(6.41)が正規化されているため、任意の \mathbf{x} に対して $\sum_n h(\mathbf{x} - \mathbf{x}_n) = 1$ となることに注意する。正規化を行うことによる効果を図 6.2 に示す。実用において、正規化はすべての基底関数が小さな値を持ってしまうような入力空間の領域が存在するのを避けるために使われることがある。そのような領域が存在することによって、領域内の予測値が小さくなるか、あるいは、予測値がほとんどバイアスパラメータによって決定されてしまうためである。

正規化された RBF が現れてくるような別の状況としては、6.3.1節で扱う、カーネル密度推定を用いた回帰がある。

各データ点に基底関数が関連付けられているために、新しいデータ点に対する予測は、計算コストがかかる。それゆえ、基底関数の数 M が、データ数 N よりも少ないモデルが提案されている(Broomhead and Lowe, 1988; Moody and Darken, 1989; Poggio and Girosi, 1990)。典型的には、基底関数の数と、その中心の位置 μ_i は、入力データ $\{\mathbf{x}_n\}$ のみから決定される。基底関数を固定しておいて、係数 $\{w_i\}$ が、最小二乗法に

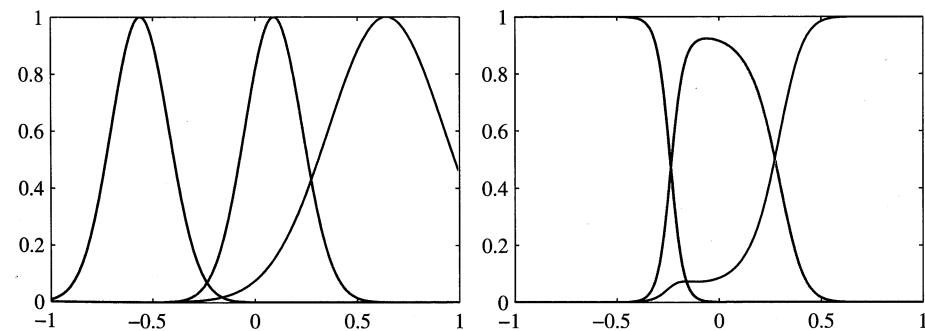


図 6.2 左図は3つのガウス基底関数。右図はそれぞれの基底関数を正規化したもの。

よって決められる。これは、3.1.1節で見たように、線形の連立方程式を解くことによって行うことができる。

基底関数の中心の最も単純な選び方は、データ点の部分集合をランダムに選ぶことである。より体系的な選択方法としては、直交最小二乗法 (orthogonal least squares) (Chen et al., 1991) がある。これは、逐次的に次のデータ点を基底関数の中心として選択していく方法であり、二乗和誤差を最も減少させるようなデータ点を追加していく。係数の決定は、アルゴリズムの一部として行われる。K-means クラスタリングアルゴリズム (☞9.1節)などのクラスタリングアルゴリズムが用いられることがある。この場合、基底関数の中心は、訓練データ点とは一致しない。

6.3.1 Nadaraya–Watson モデル

3.3.3節では、新しい入力 \mathbf{x} に対する、線形回帰モデルによる予測は、訓練集合の目標値の線形結合によって表されることを見た。線形結合の係数は、和の制約 (3.64) を満たす等価カーネル (equivalent kernel) (3.62) によって与えられた。

カーネル回帰モデル (3.61) を、カーネル密度推定の観点から動機付けることもできる。訓練集合を $\{\mathbf{x}_n, t_n\}$ として、同時分布 $p(\mathbf{x}, t)$ を推定するために、以下のように Parzen 推定法 (☞2.5.1節) を用いるとする。

$$p(\mathbf{x}, t) = \frac{1}{N} \sum_{n=1}^N f(\mathbf{x} - \mathbf{x}_n, t - t_n). \quad (6.42)$$

ここで、 $f(\mathbf{x}, t)$ は密度関数の要素であり、それぞれが、1つのデータ点を中心に持っているものとする。回帰関数 $y(\mathbf{x})$ を求めるためには、入力変数で条件付けられた目標変数の条件付き期待値を考えればよく、これは以下の式で与えられる。

$$\begin{aligned} y(\mathbf{x}) &= \mathbb{E}[t|\mathbf{x}] = \int_{-\infty}^{\infty} tp(t|\mathbf{x}) dt \\ &= \frac{\int tp(\mathbf{x}, t) dt}{\int p(\mathbf{x}, t) dt} \\ &= \frac{\sum_n \int tf(\mathbf{x} - \mathbf{x}_n, t - t_n) dt}{\sum_m \int f(\mathbf{x} - \mathbf{x}_m, t - t_m) dt}. \end{aligned} \quad (6.43)$$

簡単のため、密度関数の各要素は平均が零であるとする、つまり、以下の式がすべての \mathbf{x} に対して成り立つとする。

$$\int_{-\infty}^{\infty} f(\mathbf{x}, t) t dt = 0. \quad (6.44)$$

変数を置き換えると、次の式が得られる。

$$\begin{aligned} y(\mathbf{x}) &= \frac{\sum_n g(\mathbf{x} - \mathbf{x}_n) t_n}{\sum_m g(\mathbf{x} - \mathbf{x}_m)} \\ &= \sum_n k(\mathbf{x}, \mathbf{x}_n) t_n. \end{aligned} \quad (6.45)$$

ここで、 $n, m = 1, \dots, N$ であり、カーネル関数 $k(\mathbf{x}, \mathbf{x}_n)$ は以下で与えられるものとする。

$$k(\mathbf{x}, \mathbf{x}_n) = \frac{g(\mathbf{x} - \mathbf{x}_n)}{\sum_m g(\mathbf{x} - \mathbf{x}_m)}. \quad (6.46)$$

また、

$$g(\mathbf{x}) = \int_{-\infty}^{\infty} f(\mathbf{x}, t) dt \quad (6.47)$$

であるとする。(6.45) の結果は、Nadaraya–Watson モデル、あるいは、カーネル回帰 (kernel regression) (Nadaraya, 1964; Watson, 1964) と呼ばれる。局所的なカーネル関数を用いる場合、データ点 \mathbf{x} に近いデータ点 \mathbf{x}_n ほど大きな重みを与えられる。なお、カーネル (6.46) は、和の制約

$$\sum_{n=1}^N k(\mathbf{x}, \mathbf{x}_n) = 1$$

を満たしていることに注意する。

実際のところ、このモデルは条件付き期待値だけではなく、次で与えられる完全な条件付き確率分布も定義する。

$$p(t|\mathbf{x}) = \frac{p(t, \mathbf{x})}{\int p(t, \mathbf{x}) dt} = \frac{\sum_n f(\mathbf{x} - \mathbf{x}_n, t - t_n)}{\sum_m \int f(\mathbf{x} - \mathbf{x}_m, t - t_m) dt}. \quad (6.48)$$

これをもとにその他の期待値も計算することができる。

例として、入力 x が 1 次元で、 $f(x, t)$ が平均 0、分散 σ^2 の等方的なガウス分布 $\mathbf{z} = (x, t)$ で与えられるような場合を考える。対応する条件付き分布 (6.48) は、混合ガウス分布で与えられ(☞演習 6.18)、これを三角関数データに対して計算したものの条件付き期待値などを図 6.3 に示す。

このモデルの自明な拡張としては、ガウスカーネルをより柔軟に、例えば、入力変数と目標変数で異なる分散パラメータを持つようにすることなどが考えられる。より一般的には、同時分布 $p(t, \mathbf{x})$ を混合ガウス分布で表すこと也可能であり、9章で紹介するテクニック (Ghahramani and Jordan, 1994) を用いて学習を行い、対応する条件

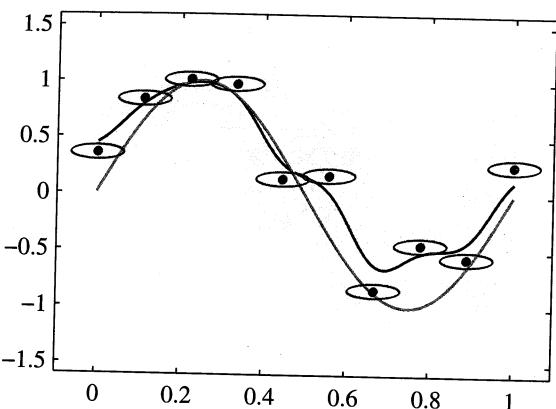


図 6.3 三角関数データに対して、等方的なガウスカーネルを用いたときの Nadaraya-Watson カーネル回帰モデルを図示したもの。もともとの正弦関数は緑の曲線で示してある。また、データ点は青で示してあり、それぞれが等方的なガウスカーネルの中心となっている。結果として得られる回帰関数は、赤い曲線で示した条件付き期待値とともに、そこから条件付き分布 $p(t|x)$ の標準偏差の 2 倍までの領域が赤い色で示されている。各データ点の周りの青い楕円は、対応するカーネルの標準偏差の等高線を示している。円になっていないのは、縦軸と横軸の縮尺が異なるためである。

付き分布 $p(t|x)$ を求めることができる。後者の場合には、もはやモデルは訓練集合のデータ点を中心にしたカーネル関数では表現されないが、混合分布において用いられる要素の数は、訓練集合の大きさよりも小さくできるため、テスト集合に対する予測をより高速に行なうことが可能になる。つまり、予測時に高速なモデルを得るために、訓練時の計算コストの増加を許容することになる。

6.4 ガウス過程

6.1 節では、回帰のための非確率的モデルに対し、双対性の概念を用いることによって、カーネル法を導いた。ここでは、カーネルを確率的識別モデルに対しても適用することで、ガウス過程を導き、ベイズ的な設定においても、自然にカーネルが現れることを見る。

3 章で、 $y(\mathbf{x}, \mathbf{w}) = \mathbf{w}^T \phi(\mathbf{x})$ の形を持つ線形回帰モデルを考えた。ここで、 \mathbf{w} はパラメータベクトルであり、 $\phi(\mathbf{x})$ は \mathbf{x} に依存する、あらかじめ固定された非線形の基底関数のベクトルを表すとする。 \mathbf{w} の事前分布を決めることで、関数 $y(\mathbf{x}, \mathbf{w})$ に対する事前分布が決まる。訓練データ集合が与えられると、今度は \mathbf{w} の事後分布が求まり、したがって、これに対応する回帰関数の事後分布が求まることになり、最終的に（これにノイズが加えられて）新しい入力ベクトル \mathbf{x} に対する予測分布 $p(t|\mathbf{x})$ が導かれる。

ガウス過程の視点から見ると、パラメトリックモデルを経由することなしに、関数に対する事前分布を直接定義しているようにも見ることができる。一見、非加算無限個の関数の上での分布を実際に扱うことできるとは思えないが、後で見るように、有限の訓練集合に対しては、訓練集合とテスト集合の入力 \mathbf{x}_n に対する有限個の関数の値のみを考えればよいことがわかり、したがって、実際には有限の空間でのみ考えればよいことになる。

ガウス過程と等価なモデルはさまざまな分野で研究されており、例えば、地球統計学においては、ガウス過程による回帰は、クリギング(kriging) (Cressie, 1993) として知られている。また、自己回帰移動平均モデル (autoregressive moving average model), カルマンフィルタ (Kalman filter), RBF ネットワークなども、ガウス過程の一種として見ることができる。機械学習から見たガウス過程の解説としては、MacKay (1998) や Williams (1999), MacKay (2003)などを参照するとよい。Rasmussen (1996) では、ガウス過程と、その他のアプローチの比較が行われている。また、最近の教科書としては、Rasmussen and Williams (2006) がある。

6.4.1 線形回帰再訪

ガウス過程の考え方を動機付けるために、まずは線形回帰の例に立ち返り、関数 $y(\mathbf{x}, \mathbf{w})$ の上での分布を考えることで、予測分布を再導出することにする。これは、ガウス過程の 1 つの特殊な例になっている。

M 個の固定された基底関数の線形結合で表されたモデルを考える。基底関数を、ベクトル $\phi(\mathbf{x})$ の要素として、以下のように書くこととする。

$$y(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}). \quad (6.49)$$

ここで、 \mathbf{x} は入力ベクトルであり、 \mathbf{w} は M 次元の重みベクトルである。次に、 \mathbf{w} の事前分布として、等方的なガウス分布、すなわち次の分布を考える。

$$p(\mathbf{w}) = \mathcal{N}(\mathbf{w} | \mathbf{0}, \alpha^{-1} \mathbf{I}). \quad (6.50)$$

ここで、 α は超パラメータであり、分布の精度（分散の逆数）を表す。任意の与えられた \mathbf{w} に対して、定義(6.49)より、 \mathbf{x} についてのある特定の関数が決まる。したがって、(6.50)で定義される、 \mathbf{w} の上での確率分布は、関数 $y(\mathbf{x})$ の上での確率分布を導くことになる。実用的には、ある入力 \mathbf{x} 、例えば、訓練データ点の集合 $\mathbf{x}_1, \dots, \mathbf{x}_N$ における関数の値を評価したいということが起こるため、関数の値の集合 $y(\mathbf{x}_1), \dots, y(\mathbf{x}_N)$ の同時分布が必要になる。関数の値の集合を、要素 $y_n = y(\mathbf{x}_n) (n = 1, \dots, N)$ を持つベクトル \mathbf{y} として表すことになると、(6.49)から、このベクトルは次の式によって与えられる。

$$\mathbf{y} = \Phi \mathbf{w}. \quad (6.51)$$

ここで、 Φ は要素 $\Phi_{nk} = \phi_k(\mathbf{x}_n)$ を持つ計画行列であるとする。 \mathbf{y} の確率分布は以下のようにして求めることができる。まず、 \mathbf{y} はガウス分布に従う変数集合である \mathbf{w} の線形結合であるから、 \mathbf{y} 自身がガウス分布に従うことがわかる(☞演習 2.31)。したがって、その平均と共分散を求めれば十分であり、それは(6.50)から、以下の式によって与えられる。

$$\mathbb{E}[\mathbf{y}] = \Phi \mathbb{E}[\mathbf{w}] = \mathbf{0} \quad (6.52)$$

$$\text{cov}[\mathbf{y}] = \mathbb{E}[\mathbf{y}\mathbf{y}^T] = \Phi \mathbb{E}[\mathbf{w}\mathbf{w}^T] \Phi^T = \frac{1}{\alpha} \Phi \Phi^T = \mathbf{K}. \quad (6.53)$$

ここで、 \mathbf{K} は、

$$K_{nm} = k(\mathbf{x}_n, \mathbf{x}_m) = \frac{1}{\alpha} \phi(\mathbf{x}_n)^T \phi(\mathbf{x}_m) \quad (6.54)$$

を要素にもつグラム行列であり、 $k(\mathbf{x}, \mathbf{x}')$ はカーネル関数である。

このモデルは、ガウス過程の 1 つの例になっており、一般的には、ガウス過程は関数 $y(\mathbf{x})$ の上の確率分布として定義され、任意の点集合 $\mathbf{x}_1, \dots, \mathbf{x}_N$ に対する $y(\mathbf{x})$ の値の同時分布が、ガウス分布に従うとしたものである。特に、入力ベクトル \mathbf{x} の次元が 2 次元のときには、ガウス確率場 (Gaussian random field) とも呼ばれる。より一般的には、確率過程 (stochastic process) $y(\mathbf{x})$ とは、任意の有限な値集合 $y(\mathbf{x}_1), \dots, y(\mathbf{x}_N)$ に対して、矛盾のない同時分布を与えるものである。

ガウス過程の重要な点は、 N 個の変数 y_1, \dots, y_N の同時分布が、平均と共分散といった、2 次までの統計量で完全に記述される点である。ほとんどの応用において、 $y(\mathbf{x})$ の平均についての事前知識はないため、対称性から、これを零とすることが多い。基底関数で見ると、これは重み事前分布 $p(\mathbf{w}|\alpha)$ の平均を零とおくことと等価である。このとき、ガウス過程は、カーネル関数

$$\mathbb{E}[y(\mathbf{x}_n)y(\mathbf{x}_m)] = k(\mathbf{x}_n, \mathbf{x}_m) \quad (6.55)$$

として与えられる、任意の 2 つの \mathbf{x} に対する $y(\mathbf{x})$ の共分散によって定まる。事前分布として重みベクトル (6.50) を持つ線形回帰モデル (6.49) によって定義されるガウス過程の場合、カーネル関数は (6.54) によって与えられる。

カーネル関数は、基底関数を選択することによって間接的に求めることもできるが、直接定義することも可能である。図 6.4 は 2 種類のカーネル関数それぞれを用いたときのガウス過程からサンプルされた関数を示している。1 つ目の方は、ガウスカーネル (6.23) を使ったものであり、2 つ目は、以下で定義される指数カーネルを使ったものである。

$$k(x, x') = \exp(-\theta|x - x'|). \quad (6.56)$$

これはもともとは、オルンシュタイン-ウーレンベック過程 (Ornstein-Uhlenbeck pro-

cess) と呼ばれる、ブラウン運動を記述するために Uhlenbeck and Ornstein (1930) によって導入されたモデルに対応している。

6.4.2 ガウス過程による回帰

ガウス過程を回帰問題に適用するには、まず、以下のように、観測される目標変数の値に含まれるノイズを考える必要がある。

$$t_n = y_n + \epsilon_n. \quad (6.57)$$

ここで、 $y_n = y(\mathbf{x}_n)$ であり、また、 ϵ_n は n 番目の観測値に加えられるノイズで、それぞれの観測値に対して独立に決定される。ここでは、ノイズもガウス分布に従う、つまり、

$$p(t_n|y_n) = \mathcal{N}(t_n|y_n, \beta^{-1}) \quad (6.58)$$

であるものとする。ここで、 β はノイズの精度を表す超パラメータである。ノイズは各データ点に対して独立に決まるため、 $\mathbf{y} = (y_1, \dots, y_N)^T$ が与えられた下での目標値 $\mathbf{t} = (t_1, \dots, t_N)^T$ の同時分布は以下の等方的なガウス分布に従う。

$$p(\mathbf{t}|\mathbf{y}) = \mathcal{N}(\mathbf{t}|\mathbf{y}, \beta^{-1} \mathbf{I}_N). \quad (6.59)$$

ここで、 \mathbf{I}_N は $N \times N$ の単位行列とする。ガウス過程の定義より、周辺分布 $p(\mathbf{y})$ は、平均が $\mathbf{0}$ で共分散がグラム行列 \mathbf{K} で与えられるガウス分布となる。

$$p(\mathbf{y}) = \mathcal{N}(\mathbf{y}|\mathbf{0}, \mathbf{K}). \quad (6.60)$$

通常は、 \mathbf{K} を決めるカーネル関数は、お互いに似ている 2 つの点 \mathbf{x}_n と \mathbf{x}_m に対して、対応する値 $y(\mathbf{x}_n)$ と $y(\mathbf{x}_m)$ が（似ていない点同士よりも）高い相関を持つという性質を持つように決められるが、「似ている」ということの定義は、それぞれの問題に依存する。

入力値 $\mathbf{x}_1, \dots, \mathbf{x}_N$ で条件付けられたときの周辺分布 $p(\mathbf{t})$ を求めるためには、 \mathbf{y} についての積分を行うことになるが、これは 2.3.3 節の線形ガウスモデルに対する結果を利用すれば可能である。(2.115) を用いると、 \mathbf{t} の周辺分布が以下のように求まる。

$$p(\mathbf{t}) = \int p(\mathbf{t}|\mathbf{y})p(\mathbf{y}) d\mathbf{y} = \mathcal{N}(\mathbf{t}|\mathbf{0}, \mathbf{C}). \quad (6.61)$$

ここで、共分散行列 \mathbf{C} は要素

$$C(\mathbf{x}_n, \mathbf{x}_m) = k(\mathbf{x}_n, \mathbf{x}_m) + \beta^{-1} \delta_{nm} \quad (6.62)$$

を持つ。この結果は、2 つのガウス分布に対応した確率変数である $y(\mathbf{x})$ と ϵ が、互いに独立であるため、その共分散も単純に足し合わせるだけでよいという事実に基づいています。

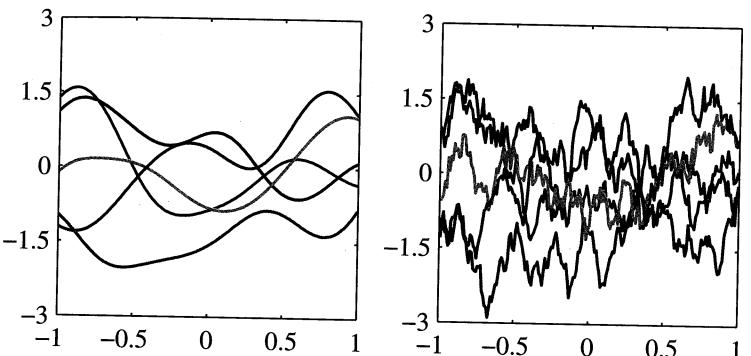


図 6.4 左図は、「ガウス」カーネルを用いたガウス過程からのサンプル。右図は指数カーネルを用いたもの。

ている。

ガウス過程回帰に用いるカーネル関数としては、以下のように、2次形式の指数をとったものに、定数と線形の項を加えたものが広く使われている。

$$k(\mathbf{x}_n, \mathbf{x}_m) = \theta_0 \exp \left\{ -\frac{\theta_1}{2} \|\mathbf{x}_n - \mathbf{x}_m\|^2 \right\} + \theta_2 + \theta_3 \mathbf{x}_n^T \mathbf{x}_m. \quad (6.63)$$

θ_3 を含む項は、入力変数の線形関数となるようなパラメトリックモデルに対応していることに注意する。この事前分布からのサンプルを、パラメータ $\theta_0, \dots, \theta_3$ の値をいろいろと変えてプロットしたものを図 6.5 に示す。また、図 6.6 は同時分布 (6.60) からのサンプルを、対応する値 (6.61) と共に示したものである。

これまでのところ、ガウス過程の視点から、データ点の集合の上の同時分布をモデル化することを考えてきたが、回帰においては、訓練データの集合が与えられたときに、新しい入力に対する目標変数の値を予測することが必要である。訓練集合として、入力 $\mathbf{x}_1, \dots, \mathbf{x}_N$ と、対応する $\mathbf{t}_N = (t_1, \dots, t_N)^T$ が与えられているときに、新しい入力ベクトル \mathbf{x}_{N+1} に対する目標変数 t_{N+1} を予測したいものとする。そのためには、予測分布 $p(t_{N+1} | \mathbf{t}_N)$ を求めることが必要となる。この分布は $\mathbf{x}_1, \dots, \mathbf{x}_N$ と \mathbf{x}_{N+1} にも依存するが、表記を単純にするために、これらの変数に依存していることを明示的には書かないことにする。

条件付き分布 $p(t_{N+1} | \mathbf{t})$ を求めるためには、まず同時分布 $p(\mathbf{t}_{N+1})$ を書き下す必要がある。ここで、 \mathbf{t}_{N+1} は、ベクトル $(t_1, \dots, t_N, t_{N+1})^T$ を表す。次に、2.3.1 節での結果を適用することで、図 6.7 に示したような所望の条件付き分布が得られる。

(6.61) から、 t_1, \dots, t_{N+1} の同時分布は次で与えられる。

$$p(\mathbf{t}_{N+1}) = \mathcal{N}(\mathbf{t}_{N+1} | \mathbf{0}, \mathbf{C}_{N+1}). \quad (6.64)$$

ここで、 \mathbf{C}_{N+1} は、 $(N+1) \times (N+1)$ の共分散行列であり、その要素は (6.62) で与え

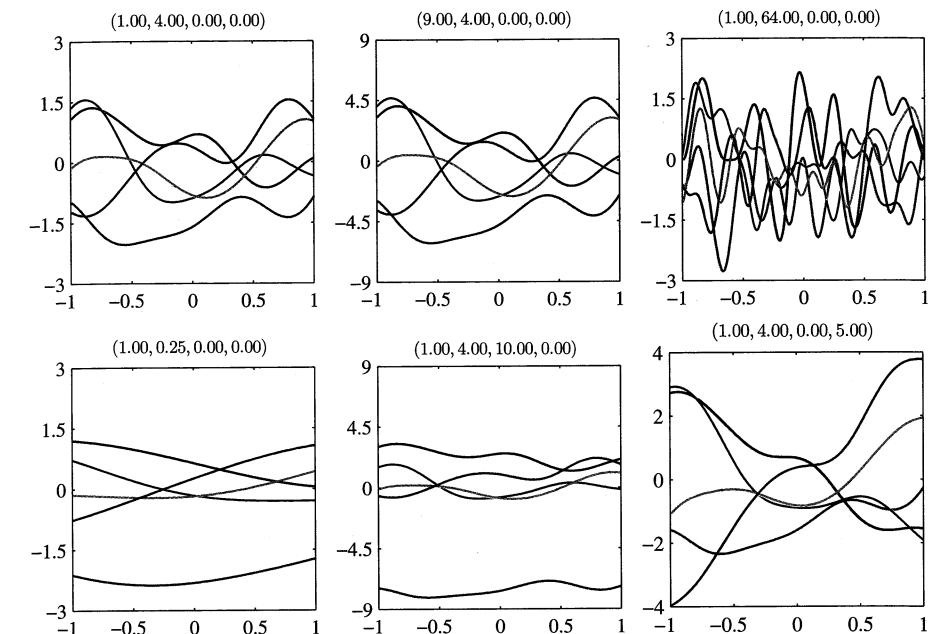


図 6.5 共分散関数 (6.63) によって定義されるガウス過程による事前分布からのサンプル。それぞれのプロットの上に $(\theta_0, \theta_1, \theta_2, \theta_3)$ が示されている。

られる。この同時分布はガウス分布であるため、2.3.1 節の結果を利用すれば、条件付きガウス分布が得られる。これを行うためには、次のように共分散行列の分割を行う。

$$\mathbf{C}_{N+1} = \begin{pmatrix} \mathbf{C}_N & \mathbf{k} \\ \mathbf{k}^T & c \end{pmatrix}. \quad (6.65)$$

ここで、 \mathbf{C}_N は、その ($n, m = 1, \dots, N$ に対する) 要素が (6.62) であるような、 $N \times N$ の共分散行列、 \mathbf{k} は、要素 $k(\mathbf{x}_n, \mathbf{x}_{N+1})$ ($n = 1, \dots, N$) を持つベクトルであるとする。また、スカラー $c = k(\mathbf{x}_{N+1}, \mathbf{x}_{N+1}) + \beta^{-1}$ とする。 (2.81) と (2.82) の結果を用いると、条件付き分布 $p(t_{N+1} | \mathbf{t})$ は、次に示す平均と共分散を持つようなガウス分布になることがわかる。

$$m(\mathbf{x}_{N+1}) = \mathbf{k}^T \mathbf{C}_N^{-1} \mathbf{t} \quad (6.66)$$

$$\sigma^2(\mathbf{x}_{N+1}) = c - \mathbf{k}^T \mathbf{C}_N^{-1} \mathbf{k}. \quad (6.67)$$

これらは、ガウス過程による回帰における重要な結果である。ベクトル \mathbf{k} は、テスト点の入力 \mathbf{x}_{N+1} の関数であるため、ガウス分布である予測分布の平均と分散もまた \mathbf{x}_{N+1} に依存する。ガウス過程による回帰の例を図 6.8 に示す。

カーネル関数についての唯一の制約は、(6.62) で与えられる共分散行列が正定値でなければならないことである。 λ_i を \mathbf{K} の固有値とすると、 \mathbf{C} の対応する固有値は

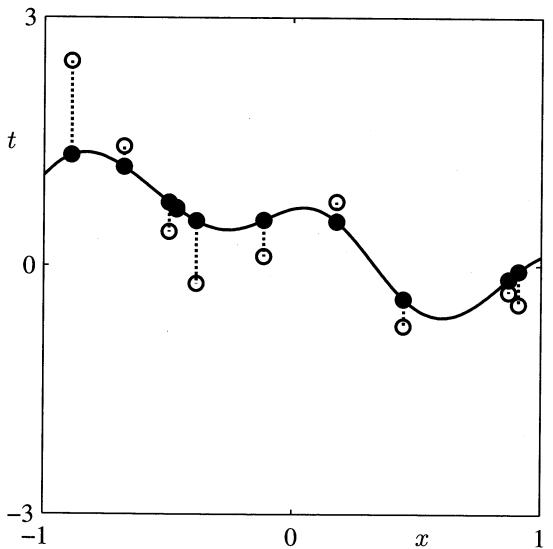


図 6.6 ガウス過程からのデータ点のサンプル $\{t_n\}$ を示したもの。実線は、関数上で定義されたガウス過程による事前分布からサンプリングされた関数を、黒い点は、入力集合 $\{x_n\}$ に対応する値 y_n を示している。また、 $\{y_n\}$ のそれぞれに独立にガウスノイズを加えた点 $\{t_n\}$ が白い丸で示されている。

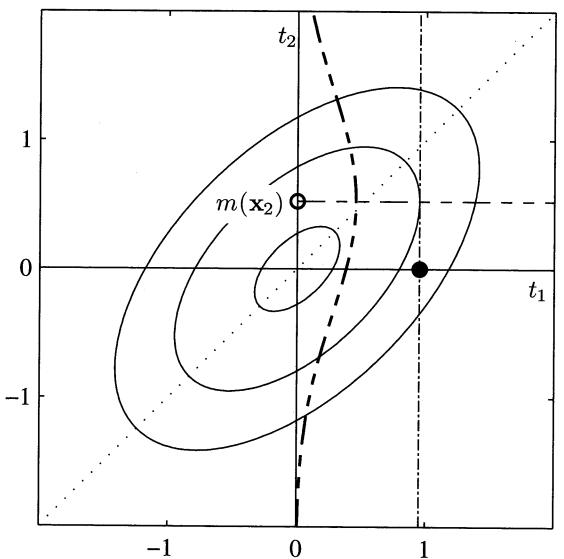


図 6.7 訓練データとテストデータが 1 つずつの場合のガウス過程による回帰の仕組みを表したもの。楕円が、同時分布 $p(t_1, t_2)$ の等高線を示している。 t_1 は訓練データ点であり、 t_1 の値に依存して、一点鎖線に対応して、 $p(t_2|t_1)$ を t_2 の関数として二点鎖線で示している。

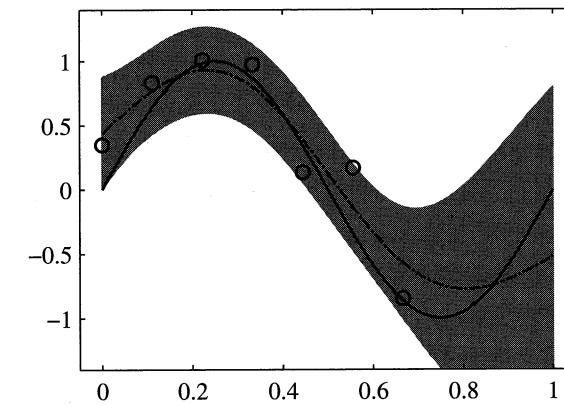


図 6.8 図 A.6 の正弦関数データに対してガウス過程を適用した結果を図示したもの。一番右から 3 つのデータ点は省かれている。実線は正弦関数を、そこから、ガウス分布に従うノイズを加えてサンプリングされたデータ点を白い丸で示している。一点鎖線は、ガウス過程による予測分布の平均を、そこから標準偏差の 2 倍までの領域が影の付いた領域として示されている。右の方に行くに従って、不確かさが大きくなっていくことがわかる。

$\lambda_i + \beta^{-1}$ になる。したがって、カーネル行列 $k(\mathbf{x}_n, \mathbf{x}_m)$ が、任意の \mathbf{x}_n と \mathbf{x}_m に対して半正定値、つまり、 $\lambda_i \geq 0$ であることを示せば十分である。これは、 \mathbf{K} の固有値の中で、零であるような λ_i があった場合でも、 $\beta > 0$ であることから、 \mathbf{C} の固有値はすべて正となるからである。この制約は、カーネル関数について以前議論したものと同じであり、6.2 節で用いたテクニックを再び用いて、適切なカーネルを設計することができる。

なお、予測分布の平均 (6.66) は、 \mathbf{x}_{N+1} の関数として、次のように書くこともできる。

$$m(\mathbf{x}_{N+1}) = \sum_{n=1}^N a_n k(\mathbf{x}_n, \mathbf{x}_{N+1}). \quad (6.68)$$

ここで、 a_n は $\mathbf{C}_N^{-1} \mathbf{t}$ の n 番目の要素であり、したがって、もしもカーネル関数 $k(\mathbf{x}_n, \mathbf{x}_m)$ が距離 $\|\mathbf{x}_n - \mathbf{x}_m\|$ にのみ依存するならば、RBF によって展開することができる。

(6.66) と (6.67) は、任意のカーネル関数 $k(\mathbf{x}_n, \mathbf{x}_m)$ を用いて、ガウス過程による回帰の予測分布を定義する。特に、 $k(\mathbf{x}, \mathbf{x}')$ が有限の基底関数で定義される場合には、3.3.2 節で得られた、ガウス過程の観点から導いた線形回帰の結果を再び導くことができる(☞演習 6.21)。

したがって、そのようなモデルに対して、予測分布は、線形回帰のパラメータ空間で考えることによって得ることができる一方、関数の空間でガウス過程を考えることによっても得ることができる。

ガウス過程を実際に計算する上で、最も大きな計算量を要する部分が、 $N \times N$ の行列の逆行列を計算する部分であり、通常の方法では $O(N^3)$ の計算量がかかる。一

方、基底関数を用いたモデルでは、 $M \times M$ の行列 \mathbf{S}_N の逆行列を計算することになり、 $O(M^3)$ の計算量になる。どちらにおいても、逆行列の計算は、与えられた訓練集合に対して1回行う必要がある。新しいテスト点が与えられると、どちらの方法もベクトルと行列の掛け算を行い、これは、ガウス過程では $O(N^2)$ 、線形の基底関数モデルでは $O(M^2)$ の計算量である。基底関数の数 M が、データ数 N よりも小さい場合には、基底関数モデルで考える方が計算量的な観点からは都合が良い。一方、ガウス過程で考えることには、無限個の基底関数でしか表せないような共分散関数を考えることができるという利点がある。

しかしながら、大きな訓練データ集合に対して、ガウス過程の直接的な適用は不可能になるため、さまざまな近似手法が提案されており、厳密な手法と比較して、より大きな訓練集合に対して適用可能になっている (Gibbs, 1997; Tresp, 2001; Smola and Bartlett, 2001; Williams and Seeger, 2001; Csató and Opper, 2002; Seeger *et al.*, 2003)。

ここでは、単一の目標変数に対するガウス過程による回帰を紹介したが、複数の目標変数に対する拡張は容易であり (☞演習 6.23)，同時クリギング (co-kriging) (Cressie, 1993) としても知られている。ガウス過程による回帰の拡張には、他にもさまざまなもののが考えられており、教師なし学習のための低次元の多様体上の分布 (Bishop *et al.*, 1998a) や、確率微分方程式の解 (Graepel, 2003) のモデル化などの目的にも用いられている。

6.4.3 超パラメータの学習

ガウス過程による予測は、ある程度は、共分散関数の選択に依存している。実際には、共分散関数をあらかじめ固定するよりも、パラメトリックな関数の族を考えて、そのパラメータをデータから推定する方が好まれる場合もある。これらのパラメータは、通常のパラメトリックモデルにおける超パラメータに対応しており、相関のスケールや、ノイズの精度などを調整する。

超パラメータを学習する方法は、尤度関数 $p(\mathbf{t}|\boldsymbol{\theta})$ の評価に基づいている。ここで、 $\boldsymbol{\theta}$ はガウス過程のモデルの超パラメータとする。最も単純なアプローチは、対数尤度関数を最大化するような $\boldsymbol{\theta}$ の点推定を行うことである。 $\boldsymbol{\theta}$ は回帰問題の超パラメータの集合を表すため、これは線形回帰モデルの第2種最尤推定として見ることもできる (☞3.5節)。対数尤度の最大化は、効率の良い、共役勾配法などの勾配を用いた最適化アルゴリズムが用いられる (Fletcher, 1987; Nocedal and Wright, 1999; Bishop and Nabney, 2008)。

ガウス過程による回帰モデルにおける対数尤度関数は、標準的な多次元のガウス分布を用いて、以下の式によって簡単に評価することができる。

$$\ln p(\mathbf{t}|\boldsymbol{\theta}) = -\frac{1}{2} \ln |\mathbf{C}_N| - \frac{1}{2} \mathbf{t}^T \mathbf{C}_N^{-1} \mathbf{t} - \frac{N}{2} \ln(2\pi). \quad (6.69)$$

非線形の最適化では、さらに、対数尤度関数のパラメータベクトル $\boldsymbol{\theta}$ についての勾配も必要になる。この章で紹介した共分散関数などのように、 \mathbf{C}_N の微分は簡単に評価できるものであると仮定する。 \mathbf{C}_N^{-1} の微分を求めるために (C.21) を利用し、また、 $\ln |\mathbf{C}_N|$ の微分を求めるために (C.22) を利用すると、次のように、対数尤度関数の微分が得られる。

$$\frac{\partial}{\partial \theta_i} \ln p(\mathbf{t}|\boldsymbol{\theta}) = -\frac{1}{2} \text{Tr} \left(\mathbf{C}_N^{-1} \frac{\partial \mathbf{C}_N}{\partial \theta_i} \right) + \frac{1}{2} \mathbf{t}^T \mathbf{C}_N^{-1} \frac{\partial \mathbf{C}_N}{\partial \theta_i} \mathbf{C}_N^{-1} \mathbf{t}. \quad (6.70)$$

一般的には、 $\ln p(\mathbf{t}|\boldsymbol{\theta})$ は非凸関数であるため、複数の極大点を持ち得る。

$\boldsymbol{\theta}$ の事前分布を考え、対数事後分布を勾配法を用いて最大化することも容易に考えられる。しかしながら、完全にベイズ的な扱いをするためには、事前分布 $p(\boldsymbol{\theta})$ と尤度関数 $p(\mathbf{t}|\boldsymbol{\theta})$ の積で重み付けされた $\boldsymbol{\theta}$ を周辺化したもの評価する必要がある。ところが一般に、厳密な周辺化は不可能であるため、近似を用いなければならない。

ガウス過程による回帰モデルは、平均と分散が、入力ベクトル \mathbf{x} の関数になっているような予測分布を与えるが、予測分布の分散への寄与は、定数パラメータ β によって決まる加法的なノイズによるものであると仮定してきた。異分散 (heteroscedastic) な場合として知られるような状況にある問題では、ノイズの分散自身も \mathbf{x} に従う。これを扱うためには、ガウス過程の枠組みに、 β が入力 \mathbf{x} に依存することを表す、第2のガウス過程を導入する (Goldberg *et al.*, 1998)。 β は分散であるから、これは非負でなければならず、ガウス過程によって $\ln \beta(\mathbf{x})$ をモデル化することになる。

6.4.4 関連度自動決定

前の節では、ガウス過程における相関のスケールパラメータの値を決定するために、最尤推定を用いる例を見た。この方法は、各入力変数に対して別々のパラメータを与えるように拡張することができるため有用である (Rasmussen and Williams, 2006)。後に見るよう、これらのパラメータを最尤推定によって最適化することは、入力間の相対的な重要度をデータから決定することになる。ここでは、ガウス過程の関連度自動決定 (ARD) の文脈で例を示すことにする。ARD はもともとはニューラルネットワークの文脈で提案されたものである (MacKay, 1994; Neal, 1996)。どのような仕組みによって、適切な入力が選択されるかについては、7.2.2節であらためて説明する。

2次元の入力空間 $\mathbf{x} = (x_1, x_2)$ をもつガウス過程を考える。カーネル関数は次の形のものを用いる。

$$k(\mathbf{x}, \mathbf{x}') = \theta_0 \exp \left\{ -\frac{1}{2} \sum_{i=1}^2 \eta_i (x_i - x'_i)^2 \right\}. \quad (6.71)$$

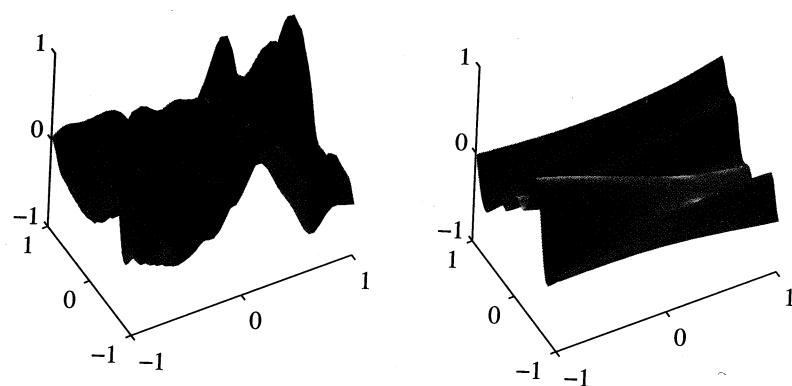


図 6.9 ガウス過程における ARD 事前分布からのサンプルを示したもの。カーネル関数は (6.71) を用いる。左図は $\eta_1 = \eta_2 = 1$ の場合、右図は $\eta_1 = 1, \eta_2 = 0.01$ の場合である。

図 6.9 に、関数 $y(\mathbf{x})$ の上の事前分布によって得られるサンプルを、精度パラメータ η_i を変えて 2 つの場合で示す。パラメータ η_i が小さくなると、関数の値が、対応する入力変数 x_i の変化に対して敏感でなくなることがわかる。最尤推定によって、これらのパラメータをデータに適応させると、対応する η_i の値が小さくなることから、予測分布にあまり寄与しない入力変数を検出することが可能になる。これは、不要な入力変数を取り除くことができるため、実用的には有用である。ARD を x_1, x_2, x_3 の 3 次元の入力変数を持つ単純な人工データ (Nabney, 2002) に対して適用した結果を図 6.10 に示す。目標変数 t は、まず x_1 の値をガウス分布を用いて 100 個生成し、これらに関数 $\sin(2\pi x_1)$ を適用して、さらにガウスノイズを加えることによって生成した。 x_2 の値は対応する x_1 の値にノイズを加えることによって、また、 x_3 は 2 つの変数とは独立にガウス分布によって生成した。したがって、 t の予測に際して x_1 は最も役に立つ変数であり、 x_2 はより大きなノイズの加わった比較的役に立たない変数であり、また、 x_3 は t と偶然以上の関連を持たない、全く役に立たない変数であると言えることができる。ARD のパラメータ η_1, η_2, η_3 をもったガウス過程の周辺化尤度の最適化は、スケーリングを伴う共役勾配法 (scaled conjugate gradients algorithm) によって行った。図 6.10 を見ると、 η_1 は比較的大きな値に収束し、 η_2 はもっと小さな値に収束していることがわかる。また、 η_3 は非常に小さい値に収束しているが、これは、 x_3 が t の予測には役に立たないことを示している。

ARD の枠組みは指数-2 次カーネル (6.63) に容易に組み込むことができ、次のような形のカーネル関数を与える。

$$k(\mathbf{x}_n, \mathbf{x}_m) = \theta_0 \exp \left\{ -\frac{1}{2} \sum_{i=1}^D \eta_i (x_{ni} - x_{mi})^2 \right\} + \theta_2 + \theta_3 \sum_{i=1}^D x_{ni} x_{mi}. \quad (6.72)$$

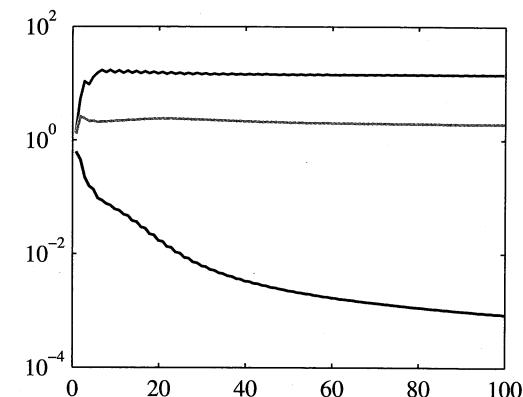


図 6.10 ガウス過程における ARD を x_1, x_2, x_3 の 3 次元の入力変数を持つ人工データに対して適用した結果を示したもの。曲線は、対応する超パラメータ η_1 (赤)、 η_2 (緑)、 η_3 (青) の値を、周辺尤度を最適化する際の繰り返しの回数の関数として示す。詳細は本文を参照のこと。縦軸は対数スケールであることに注意する。

ここで、 D は入力空間の次元を表す。このカーネルは、ガウス過程による回帰問題の種々の応用において有用であることがわかっている。

6.4.5 ガウス過程による分類

確率的なアプローチを用いた分類では、訓練データ集合が与えられたときに、新しい入力ベクトルに対する目標変数の事後確率をモデル化することが目的である。これらの確率は $(0, 1)$ の間に収まる必要があるが、ガウス過程のモデルの予測は実数値全体での値をとり得る。しかしながら、ガウス過程の出力を適当な非線形の活性化関数を用いて変換することによって、簡単にガウス過程を分類問題に適用できるよう変更できる。

まずは、目標変数が $t \in \{0, 1\}$ であるような、2 クラス分類問題を考える。関数 $a(\mathbf{x})$ の上でのガウス過程を定義し、これを (4.59) で与えられるロジスティックシグモイド関数 $y = \sigma(a)$ で変換することで、 $y \in (0, 1)$ であるような関数 $y(\mathbf{x})$ の上での非ガウス確率過程が得られる。1 次元の入力空間の場合の例を図 6.11 に示す。目標変数 t の確率分布は、次のベルヌーイ分布で与えられる。

$$p(t|a) = \sigma(a)^t (1 - \sigma(a))^{1-t}. \quad (6.73)$$

通常通り、入力の訓練集合を $\mathbf{x}_1, \dots, \mathbf{x}_N$ 、対応する目標変数の観測値を $\mathbf{t} = (t_1, \dots, t_N)^T$ のようにおくことにする。また、テスト点の入力を \mathbf{x}_{N+1} 、その目標変数値を

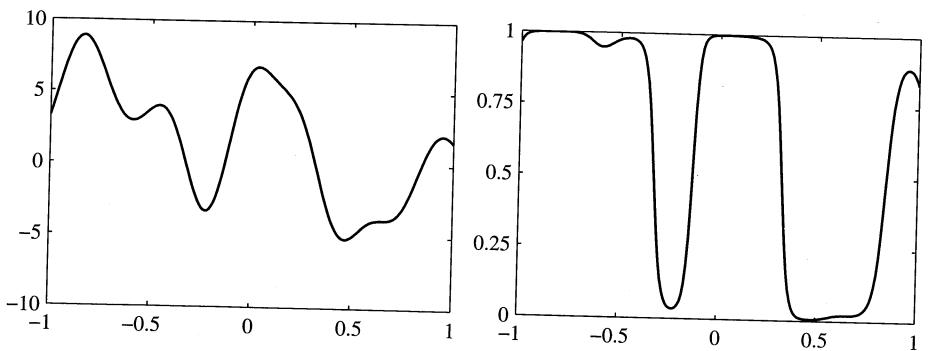


図 6.11 左図は、関数 $a(x)$ に対するガウス過程の事前分布からのサンプルを、右図は、これをロジスティックシグモイド関数で変換した結果を示す。

t_{N+1} とする。目的は、予測分布 $p(t_{N+1}|\mathbf{t})$ を決定することである。(なお、条件部の入力変数は省略して表記することにする。) そのために、要素 $a(\mathbf{x}_1), \dots, a(\mathbf{x}_{N+1})$ を持つベクトル \mathbf{a}_{N+1} に対するガウス過程による事前分布を考える。これが \mathbf{t}_{N+1} に対する非ガウス過程による事前分布を導き、訓練データ \mathbf{t}_N が与えられた条件の下で、所望の予測分布が与えられることになる。 \mathbf{a}_{N+1} に対するガウス過程による事前分布は以下の形式を取る。

$$p(\mathbf{a}_{N+1}) = \mathcal{N}(\mathbf{a}_{N+1} | \mathbf{0}, \mathbf{C}_{N+1}). \quad (6.74)$$

すべての訓練データ点は正しいクラスラベルが与えられているとすると、回帰の場合とは異なり、共分散行列はノイズ項を含まない。しかしながら、数値的な安定性の問題から、共分散行列の正定値性を保証するために、パラメータ ν をもつノイズのような項を入れておくと便利である。結局、共分散行列 \mathbf{C}_{N+1} の各要素は以下のように与えられる。

$$C(\mathbf{x}_n, \mathbf{x}_m) = k(\mathbf{x}_n, \mathbf{x}_m) + \nu \delta_{nm}. \quad (6.75)$$

ここで $k(\mathbf{x}_n, \mathbf{x}_m)$ は、6.2節で考えたような、任意の半正定値であるカーネル関数であり、 ν の値は通常、あらかじめ与えられた定数である。カーネル関数 $k(\mathbf{x}, \mathbf{x}')$ は、パラメータベクトル θ によって決定されるとする。後で議論するように、 θ も訓練データから学習することができる。

2クラス分類問題においては、 $p(t_{N+1} = 0|\mathbf{t}_N)$ は $1 - p(t_{N+1} = 1|\mathbf{t}_N)$ によって与えられるため、 $p(t_{N+1} = 1|\mathbf{t}_N)$ を予測するだけで十分である。求めるべき予測分布は、以下で与えられる。

$$p(t_{N+1} = 1|\mathbf{t}_N) = \int p(t_{N+1} = 1|a_{N+1}) p(a_{N+1}|\mathbf{t}_N) da_{N+1}. \quad (6.76)$$

ここで、 $p(t_{N+1} = 1|a_{N+1}) = \sigma(a_{N+1})$ であるとする。

この積分は、解析的に求めることは不可能であるため、サンプリング (Neal, 1997)

を用いて近似される。あるいは別の方法として、解析的な近似に基づくテクニックを用いることもできる。4.5.2節において、ガウス分布によるロジスティックシグモイド関数の重畠積分の近似公式 (4.153) を示したが、この結果を用いて (6.76) の積分を評価し、事後分布 $p(a_{N+1}|\mathbf{t}_N)$ のガウス分布による近似を求めることができる。ガウス分布による近似は、通常、中心極限定理 (2.3節) によって、真の事後分布がデータ点の数の増加とともにガウス分布に近づくことから正当化される。ガウス過程の場合には、変数の数はデータ点の数の増加とともに大きくなるため、この議論は直接には適用されない。しかしながら、入力 \mathbf{x} の空間のある決まった領域に含まれるデータ点の数が増加すると考えると、対応する関数 $a(\mathbf{x})$ の不確定性は減少し、やはり結果として、漸近的にガウス分布へと近づくことになる (Williams and Barber, 1998)。

ガウス分布による近似の方法としては、3つの異なるアプローチが提案されている。1つ目は、変分推論法 (variational inference) (10.1節) に基づく方法 (Gibbs and MacKay, 2000) で、ロジスティックシグモイド関数の局所的な変分近似 (10.144) を用いる。この方法では、シグモイド関数の積を、ガウス分布の積によって近似し、それによって、 \mathbf{a}_N の周辺化を解析的に行うことができる。この方法では、尤度関数 $p(\mathbf{t}_N|\theta)$ の下界を得ることもできる。ガウス過程による分類への変分アプローチの枠組みは多クラス ($K > 2$) の分類問題の場合にも拡張可能であり、ソフトマックス関数をガウス分布によって近似することによって達成される (Gibbs, 1997)。

2つ目のアプローチは、EP法 (expectation propagation method) (10.7節) を用いたものである (Opper and Winther, 2000b; Minka, 2001b; Seeger, 2003)。真の事後分布は单峰性を持つため、後で見るように、EP法は良い結果をもたらす。

6.4.6 ラプラス近似

ガウス過程による分類に対する3つ目のアプローチは、ラプラス近似 (4.4節) を用いた方法であり、ここではこれを詳細に述べることにする。予測分布 (6.76) を求めるために、 a_{N+1} の事後分布のガウス分布による近似を行う。ベイズの定理と、 $p(\mathbf{t}_N|a_{N+1}, \mathbf{a}_N) = p(\mathbf{t}_N|\mathbf{a}_N)$ を用いることによって、以下の式が得られる。

$$\begin{aligned} p(a_{N+1}|\mathbf{t}_N) &= \int p(a_{N+1}, \mathbf{a}_N|\mathbf{t}_N) d\mathbf{a}_N \\ &= \frac{1}{p(\mathbf{t}_N)} \int p(a_{N+1}, \mathbf{a}_N) p(\mathbf{t}_N|a_{N+1}, \mathbf{a}_N) d\mathbf{a}_N \\ &= \frac{1}{p(\mathbf{t}_N)} \int p(a_{N+1}|\mathbf{a}_N) p(\mathbf{a}_N) p(\mathbf{t}_N|\mathbf{a}_N) d\mathbf{a}_N \\ &= \int p(a_{N+1}|\mathbf{a}_N) p(\mathbf{a}_N|\mathbf{t}_N) d\mathbf{a}_N. \end{aligned} \quad (6.77)$$

条件付き分布 $p(a_{N+1}|\mathbf{a}_N)$ は、ガウス過程による回帰における結果である (6.66) と (6.67) を用いることで、以下のように得られる。

$$p(a_{N+1}|\mathbf{a}_N) = \mathcal{N}(a_{N+1}|\mathbf{k}^T \mathbf{C}_N^{-1} \mathbf{a}_N, c - \mathbf{k}^T \mathbf{C}_N^{-1} \mathbf{k}). \quad (6.78)$$

したがって、(6.77) の積分は、事後分布 $p(\mathbf{a}_N|\mathbf{t}_N)$ のラプラス近似を用いた後に、2つのガウス分布のたたみ込みについての結果を用いることで得られる。

事前分布 $p(\mathbf{a}_N)$ は、平均が零で、共分散行列が \mathbf{C}_N であるようなガウス過程によって与えられ、データについての項は（データ点が互いに独立であるとして）以下のように与えられる。

$$p(\mathbf{t}_N|\mathbf{a}_N) = \prod_{n=1}^N \sigma(a_n)^{t_n} (1 - \sigma(a_n))^{1-t_n} = \prod_{n=1}^N e^{a_n t_n} \sigma(-a_n). \quad (6.79)$$

次に、テイラー展開によって、 $p(\mathbf{a}_N|\mathbf{t}_N)$ の対数を展開することでラプラス近似を得る。これは、定数である正規化項を無視すると、次のように与えられる。

$$\begin{aligned} \Psi(\mathbf{a}_N) &= \ln p(\mathbf{a}_N) + \ln p(\mathbf{t}_N|\mathbf{a}_N) \\ &= -\frac{1}{2} \mathbf{a}_N^T \mathbf{C}_N^{-1} \mathbf{a}_N - \frac{N}{2} \ln(2\pi) - \frac{1}{2} \ln |\mathbf{C}_N| + \mathbf{t}_N^T \mathbf{a}_N \\ &\quad - \sum_{n=1}^N \ln(1 + e^{a_n}). \end{aligned} \quad (6.80)$$

まず、この事後分布のモードを求める必要がある。それには、以下のように与えられる $\Psi(\mathbf{a}_N)$ の勾配が必要である。

$$\nabla \Psi(\mathbf{a}_N) = \mathbf{t}_N - \boldsymbol{\sigma}_N - \mathbf{C}_N^{-1} \mathbf{a}_N. \quad (6.81)$$

ここで、 $\boldsymbol{\sigma}_N$ は、要素 $\sigma(a_n)$ を持つベクトルである。 $\boldsymbol{\sigma}_N$ は \mathbf{a}_N の非線形的関数であるため、これを単純に零とおくことによってモードを求めるることはできないので、ニュートン-ラフソン法に基づく、繰り返し法によって求める。これは結果的に、反復再重み付け最小二乗法 (iterative reweighted least squares method) となる (☞4.3.3 節)。計算には、ラプラス近似においても必要となる $\Psi(\mathbf{a}_N)$ の2階微分が必要であるが、これは次の式で与えられる。

$$\nabla \nabla \Psi(\mathbf{a}_N) = -\mathbf{W}_N - \mathbf{C}_N^{-1}. \quad (6.82)$$

ここで、 \mathbf{W}_N は $\sigma(a_n)(1 - \sigma(a_n))$ を要素にもつ対角行列であるとする。また、ロジスティックシグモイド関数の微分の結果 (4.88) を用いた。対角要素は $(0, 1/4)$ の区間の値を持つため、 \mathbf{W}_N は正定値行列であることに注意する。 \mathbf{C}_N (とその逆行列) は定義より正定値であり、また、2つの正定値行列の和は、やはり正定値行列であることから (☞演習 6.24)，ヘッセ行列 $\mathbf{A} = -\nabla \nabla \Psi(\mathbf{a}_N)$ は正定値であるため、事後分布

$p(\mathbf{a}_N|\mathbf{t}_N)$ の対数は凸関数であり、したがって、この関数は单峰で大域的な最適解を持つ。しかしながら、ヘッセ行列は \mathbf{a}_N の関数であるため、事後分布はガウス分布ではない。

ニュートン-ラフソン法 (Newton-Raphson method) の公式 (4.92) を用いることで、 \mathbf{a}_N の逐次更新式は以下で与えられる (☞演習 6.25)。

$$\mathbf{a}_N^{\text{new}} = \mathbf{C}_N(\mathbf{I} + \mathbf{W}_N \mathbf{C}_N)^{-1} \{ \mathbf{t}_N - \boldsymbol{\sigma}_N + \mathbf{W}_N \mathbf{a}_N \}. \quad (6.83)$$

繰り返しは、モード \mathbf{a}_N^* に収束するまで続けられる。モードにおいては、勾配 $\nabla \Psi(\mathbf{a}_N)$ は零になるため、 \mathbf{a}_N^* は次の式を満たす。

$$\mathbf{a}_N^* = \mathbf{C}_N(\mathbf{t}_N - \boldsymbol{\sigma}_N). \quad (6.84)$$

事後分布のモード \mathbf{a}_N^* に到達したら、ヘッセ行列

$$\mathbf{H} = -\nabla \nabla \Psi(\mathbf{a}_N) = \mathbf{W}_N + \mathbf{C}_N^{-1} \quad (6.85)$$

を求める。ここで、 \mathbf{W}_N の要素は \mathbf{a}_N^* を用いて評価する。これを用いて、事後分布 $p(\mathbf{a}_N|\mathbf{t}_N)$ のガウス分布による近似が以下のように求まる。

$$q(\mathbf{a}_N) = \mathcal{N}(\mathbf{a}_N|\mathbf{a}_N^*, \mathbf{H}^{-1}). \quad (6.86)$$

これと、(6.78) を組み合わせることで、(6.77) の積分を評価することができる。これは、線形ガウスモデルに対応しているため、(2.115) の一般的な結果を用いることで次を得る (☞演習 6.26)。

$$\mathbb{E}[a_{N+1}|\mathbf{t}_N] = \mathbf{k}^T (\mathbf{t}_N - \boldsymbol{\sigma}_N) \quad (6.87)$$

$$\text{var}[a_{N+1}|\mathbf{t}_N] = c - \mathbf{k}^T (\mathbf{W}_N^{-1} + \mathbf{C}_N)^{-1} \mathbf{k}. \quad (6.88)$$

$p(a_{N+1}|\mathbf{t}_N)$ のガウス分布による近似が得られたため、(4.153) を用いて、積分 (6.76) を近似することができる。4.5節のベイズロジスティック回帰モデルと同様、 $p(t_{N+1}|\mathbf{t}_N) = 0.5$ に対応する決定面のみに興味がある場合には、平均のみを考えれば十分であり、分散は無視してよい。

さらに、共分散関数のパラメータ $\boldsymbol{\theta}$ も決定する必要がある。1つのアプローチは、対数尤度関数とその勾配を考えることによって、尤度関数 $p(\mathbf{t}_N|\boldsymbol{\theta})$ を最大化するという方法である。必要であれば、適当な正則化項を加え、ペナルティー付きの最尤推定を行うこともできる。尤度関数は次のように定義される。

$$p(\mathbf{t}_N|\boldsymbol{\theta}) = \int p(\mathbf{t}_N|\mathbf{a}_N) p(\mathbf{a}_N|\boldsymbol{\theta}) d\mathbf{a}_N. \quad (6.89)$$

この積分は、解析的には求められないため、再びラプラス近似を行う。(4.135) を利用すると、次のように対数尤度関数の近似を求めることができる。

$$\ln p(\mathbf{t}_N|\boldsymbol{\theta}) = \Psi(\mathbf{a}_N^*) - \frac{1}{2} \ln |\mathbf{W}_N + \mathbf{C}_N^{-1}| + \frac{N}{2} \ln(2\pi). \quad (6.90)$$

ここで、 $\Psi(\mathbf{a}_N^*) = \ln p(\mathbf{a}_N^*|\boldsymbol{\theta}) + \ln p(\mathbf{t}_N|\mathbf{a}_N^*)$ とする。さらに、 $\ln p(\mathbf{t}_N|\boldsymbol{\theta})$ のパラメータベクトル $\boldsymbol{\theta}$ についての勾配が必要になる。 $\boldsymbol{\theta}$ を変更すると、 \mathbf{a}_N^* も変化するため、勾配に新たな項が加わる。したがって、(6.90) を $\boldsymbol{\theta}$ について微分するときには、共分散行列 \mathbf{C}_N が $\boldsymbol{\theta}$ に依存することによる部分と、 \mathbf{a}_N^* が $\boldsymbol{\theta}$ に依存することによる部分の2種類の項が現れることになる。

$\boldsymbol{\theta}$ に明示的に依存することに由来する項は、(6.80) および、(C.21) と (C.22) の結果を用いることによって得られる。

$$\begin{aligned} \frac{\partial \ln p(\mathbf{t}_N|\boldsymbol{\theta})}{\partial \theta_j} &= \frac{1}{2} \mathbf{a}_N^{*\mathrm{T}} \mathbf{C}_N^{-1} \frac{\partial \mathbf{C}_N}{\partial \theta_j} \mathbf{C}_N^{-1} \mathbf{a}_N^* \\ &\quad - \frac{1}{2} \text{Tr} \left[(\mathbf{I} + \mathbf{C}_N \mathbf{W}_N)^{-1} \mathbf{W}_N \frac{\partial \mathbf{C}_N}{\partial \theta_j} \right]. \end{aligned} \quad (6.91)$$

\mathbf{a}_N^* が $\boldsymbol{\theta}$ に依存することに由来する項を計算するために、ラプラス近似が、 $\mathbf{a}_N = \mathbf{a}_N^*$ において、 $\Psi(\mathbf{a}_N)$ の勾配が零になるように構成されているため、 $\Psi(\mathbf{a}_N^*)$ は、 \mathbf{a}_N^* に依存することによる勾配には寄与しないことに注意する。したがって、 $\boldsymbol{\theta}$ の要素 θ_j についての微分は、次のように求めることができる。

$$\begin{aligned} &- \frac{1}{2} \sum_{n=1}^N \frac{\partial \ln |\mathbf{W}_N + \mathbf{C}_N^{-1}|}{\partial a_n^*} \frac{\partial a_n^*}{\partial \theta_j} \\ &= - \frac{1}{2} \sum_{n=1}^N [(\mathbf{I} + \mathbf{C}_N \mathbf{W}_N)^{-1} \mathbf{C}_N]_{nn} \sigma_n^*(1 - \sigma_n^*)(1 - 2\sigma_n^*) \frac{\partial a_n^*}{\partial \theta_j}. \end{aligned} \quad (6.92)$$

ここで、 $\sigma_n^* = \sigma(a_n^*)$ である。また、ここでも (C.22) の結果と、 \mathbf{W}_N の定義を用いた。関係 (6.84) を θ_j について微分したものを使って、 a_N^* を θ_j について微分したものを求めると、

$$\frac{\partial a_n^*}{\partial \theta_j} = \frac{\partial \mathbf{C}_N}{\partial \theta_j} (\mathbf{t}_N - \boldsymbol{\sigma}_N) - \mathbf{C}_N \mathbf{W}_N \frac{\partial a_n^*}{\partial \theta_j} \quad (6.93)$$

となり、これを並べ替えることで以下が得られる。

$$\frac{\partial a_n^*}{\partial \theta_j} = (\mathbf{I} + \mathbf{W}_N \mathbf{C}_N)^{-1} \frac{\partial \mathbf{C}_N}{\partial \theta_j} (\mathbf{t}_N - \boldsymbol{\sigma}_N). \quad (6.94)$$

(6.91) と (6.92)、および (6.94) を組み合わせると、対数尤度関数の勾配を求めることができる。これを標準的な非線形最適化のアルゴリズムと共に用いることで、 $\boldsymbol{\theta}$ の値を決定することができる。

ラプラス近似によるガウス過程を人工的な2クラス分類問題のデータ（付録A）に対して適用した結果を図6.12に示す。ラプラス近似を使ったガウス過程による分類は、ソフトマックス活性化関数を用いることで、容易に多クラス ($K > 2$) 分類問題へ拡張することができる（Williams and Barber, 1998）。

6.4.7 ニューラルネットワークとの関係

すでに、ニューラルネットワークによって表現できる関数の種類は、隠れユニットの数 M に依存し、十分に大きい M を取ることによって、2層のニューラルネットワークは任意の関数を任意の精度で近似できることを見てきたが、最尤推定の枠組みでは、過学習を避けるために、隠れユニットの数は（訓練集合のサイズに合う程度まで）制限する必要がある。しかしながら、ペイズの観点からは、訓練集合のサイズに依存して、ネットワークのパラメータの数を制限することは、ほとんど意味を持たない。

ペイズニューラルネットワークでは、パラメータベクトル \mathbf{w} の事前分布とネットワーク関数 $f(\mathbf{x}, \mathbf{w})$ を組み合わせることによって、 $y(\mathbf{x})$ の上の関数についての事前分布が得られる。ここで、 \mathbf{y} はネットワークの出力ベクトルである。Neal (1996) では、 \mathbf{w} の事前分布として広いクラスの分布に対して、ニューラルネットワークによって生成される関数の分布が、 $M \rightarrow \infty$ の極限においてガウス過程に近づくことが示されている。しかしながら、この極限においては、ニューラルネットワークの出力変数は独立になることに注意せねばならない。ニューラルネットワークで最も有用な点の1つは、出力が隠れユニットを共有することであり、これによって、お互いに「統計的な強度を借りる」ことが可能になる。つまり、それぞれの隠れユニットに関連付けられた重みは、（1つではなく）すべての出力変数から影響を受けることになる。この性質は、極限でのガウス過程においては失われてしまう。

すでに、ガウス過程は、その共分散（カーネル）関数によって決定されることを見たが、Williams (1998) では、プロビット関数とガウス関数の2つの活性化関数を隠れユニットに用いた場合の共分散を明示的に導いている。零を中心としたガウス関数による重みの事前分布は、重み空間における平行移動不変性が成り立たないため、結果として、これらのカーネル関数 $k(\mathbf{x}, \mathbf{x}')$ は不变にはならない、つまり、差 $\mathbf{x} - \mathbf{x}'$ の関数として表現されない。

共分散関数を直接的に扱うことによって、事前分布の重みの分布を暗黙的に周辺化していることになる。事前分布の重みパラメータが、超パラメータによって決定されるならば、図5.11の無限個の隠れユニットの例からもわかるように、それらの値は、関数の分布の長さスケールを決定する。なお、超パラメータについては解析的に周辺化することはできないため、6.4節で見たようなテクニックを用いる必要がある。

演習問題

- 6.1 (標準) [www](#) 6.1節で紹介した最小二乗法線形回帰問題の双対表現を示せ。また、解のベクトル \mathbf{a} の要素 a_n がベクトル $\phi(\mathbf{x}_n)$ の要素の線形結合で表されることを示せ。それらの係数をベクトル \mathbf{w} として、双対表現の双対表現がもともとの表現に戻ること

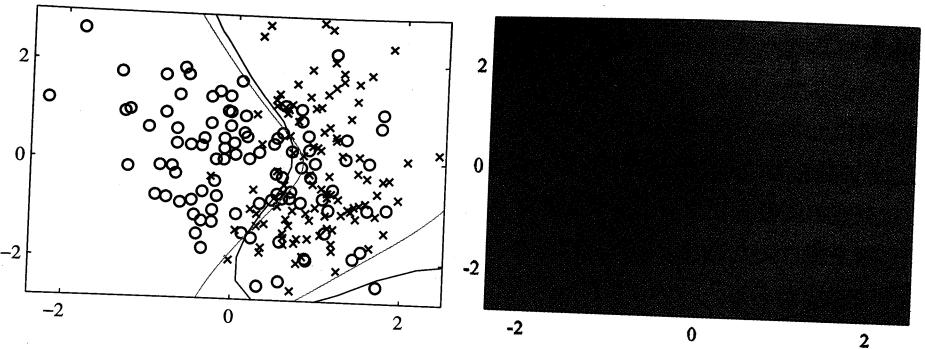


図 6.12 ガウス過程による分類を図示したもの。左図は、データと、真の分布から求まる最適な決定面を緑で、ガウス過程によって求まった決定面を黒で示している。右図は、青と赤のそれぞれのクラスに対して予測された事後分布をガウス過程による決定面とともに示している。
を、 \mathbf{w} をパラメータベクトルとして示せ。

6.2 (標準) この演習問題では、パーセプトロンの学習アルゴリズムの双対表現を導く。パーセプトロンでの更新則(4.55)を用いて、訓練後の重みベクトル \mathbf{w} が、ベクトル $t_n \phi(\mathbf{x}_n)$ (ただし $t_n \in \{-1, +1\}$) の線形結合で表されることを示せ。この線形結合の係数を α_n として、パーセプトロンの学習アルゴリズムを導き、また、 α_n を用いてパーセプトロンの予測関数を示せ。また、特徴ベクトル $\phi(\mathbf{x})$ は、カーネル関数 $k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^T \phi(\mathbf{x}')$ の形でのみ現れることを示せ。

6.3 (基本) 最近傍法(2.5.2節)は、新しい入力ベクトル \mathbf{x} を、訓練集合の中でこれに最も近い入力ベクトル \mathbf{x}_n を持つものと同じクラスに分類する。最も単純な場合では、距離はユークリッド距離 $\|\mathbf{x} - \mathbf{x}_n\|^2$ が用いられる。これをスカラー積で表すことで、カーネル置換を用いて、一般的な非線形カーネルを用いた最近傍法を導け。

6.4 (基本) 付録 C では、要素がすべて正であるが、負の固有値をもつために、正定値ではない行列の例を紹介している。逆に、 2×2 行列で、すべての固有値が正であるが、少なくとも 1 つの負の要素をもつような行列を挙げよ。

6.5 (基本) **WWW** 有効なカーネル関数を構成するために利用できる等式(6.13)と(6.14)を確かめよ。

6.6 (基本) 有効なカーネル関数を構成するために利用できる等式(6.15)と(6.16)を確かめよ。

6.7 (基本) **WWW** 有効なカーネル関数を構成するために利用できる等式(6.17)と(6.18)を確かめよ。

6.8 (基本) 有効なカーネル関数を構成するために利用できる等式(6.19)と(6.20)を確かめよ。

6.9 (基本) 有効なカーネル関数を構成するために利用できる等式(6.21)と(6.22)を確かめよ。

6.10 (基本) 関数 $f(\mathbf{x})$ を学習するためのカーネルとして $k(\mathbf{x}, \mathbf{x}') = f(\mathbf{x})f(\mathbf{x}')$ が理想的であることを、このカーネルに基づく線形の学習器は、常に $f(\mathbf{x})$ に比例する解を見つけることを示すことで示せ。

6.11 (基本) (6.25) の展開の中央の要素を、べき級数展開することによって、ガウスカーネル(6.23)は、無限次元の特徴ベクトルの内積で表されることを示せ。

6.12 (標準) **WWW** あらかじめ固定された集合 D のすべての部分集合 A の空間を考え、カーネル関数(6.27)は、写像 $\phi(A)$ によって定義される $2^{|D|}$ 次元の特徴空間における内積であることを示せ。なお、 A は D の部分集合であり、部分集合 U で指定される $\phi(A)$ の各要素 $\phi_U(A)$ は、以下で与えられるとする。

$$\phi_U(A) = \begin{cases} 1, & U \subseteq A \text{ のとき} \\ 0, & \text{それ以外。} \end{cases} \quad (6.95)$$

ここで、 $U \subseteq A$ は、 U は A の部分集合であるか、 A そのものであることを表すとする。

6.13 (基本) (6.33) で定義されるフィッシャーカーネルは、パラメータベクトル θ に非線形の変換 $\theta \rightarrow \psi(\theta)$ を行っても不变であることを示せ。なお、 $\psi(\cdot)$ は可逆で、かつ、微分可能であるとする。

6.14 (基本) **WWW** 平均 μ と共分散 \mathbf{S} をもつガウス分布 $p(\mathbf{x}|\mu, \mathbf{S}) = \mathcal{N}(\mathbf{x}|\mu, \mathbf{S})$ に対して、(6.33) で定義されるフィッシャーカーネルの具体的な形式を導け。

6.15 (基本) 2×2 のグラム行列の行列式を考えて、正定値であるカーネル関数 $k(x, x')$ はコーシー・シュワルツの不等式

$$k(x_1, x_2)^2 \leq k(x_1, x_1)k(x_2, x_2) \quad (6.96)$$

を満たすことを示せ。

6.16 (標準) パラメータベクトル \mathbf{w} と入力のデータ集合 $\mathbf{x}_1, \dots, \mathbf{x}_N$ 、および非線形の特徴空間への写像 $\phi(\mathbf{x})$ を持つパラメトリックモデルに対し、誤差関数が \mathbf{w} の関数として次のように与えられるとする。

$$J(\mathbf{w}) = f(\mathbf{w}^T \phi(\mathbf{x}_1), \dots, \mathbf{w}^T \phi(\mathbf{x}_N)) + g(\mathbf{w}^T \mathbf{w}). \quad (6.97)$$

ここで、 $g(\cdot)$ は単調増加関数であるとする。 \mathbf{w} を

$$\mathbf{w} = \sum_{n=1}^N \alpha_n \phi(\mathbf{x}_n) + \mathbf{w}_\perp \quad (6.98)$$

という形式で書くことによって、 $J(\mathbf{w})$ を最小化する \mathbf{w} の値は、 $n = 1, \dots, N$ についての基底関数 $\phi(\mathbf{x}_n)$ の線形結合で表されることを示せ。ただし、すべての n について、 $\mathbf{w}_\perp^T \phi(\mathbf{x}_n) = 0$ であるとする。

6.17 (標準) **WWW** 入力に、分布 $\nu(\xi)$ を持つノイズがある場合の二乗和誤差関数(6.39)を考える。変分法を用いて、この誤差関数を関数 $y(\mathbf{x})$ について最小化し、最適な解は、基底関数として(6.41)を用いた展開(6.40)の形で与えられることを示せ。

6.18 (基本) 等方共分散をもつ、つまり、共分散行列が $\sigma^2 \mathbf{I}$ (\mathbf{I} は単位行列) で与えられるようなガウス基底を持つような Nadaraya-Watson モデルを考える。入力変数 x と、目標変数 t はそれぞれ 1 次元であるとする。このとき、条件付き密度 $p(t|x)$ 、条件付き期待値 $\mathbb{E}[t|x]$ 、および条件付き分散 $\text{var}[t|x]$ をそれぞれカーネル関数 $k(x, x_n)$ を用いて書け。

第6章 カーネル法

カーネル回帰の問題を別の視点から見ると、入力変数と目標変数が加法的なノイズによって影響されていると考えることができる。通常通り、各目標変数 t_n を、点 \mathbf{z}_n において評価された関数 $y(\mathbf{z}_n)$ に、ガウスノイズが加わったものとする。 \mathbf{z}_n は直接観測されることはなく、ノイズが加わった $\mathbf{x}_n = \mathbf{z}_n + \boldsymbol{\xi}_n$ が観測される。ここで、確率変数 $\boldsymbol{\xi}$ は、ある分布 $g(\boldsymbol{\xi})$ に従うとする。観測された集合 $\{\mathbf{x}_n, t_n\} (n=1, \dots, N)$ に対して、入力変数に加えられたノイズの分布で期待値を取った二乗和誤差関数

$$E = \frac{1}{2} \sum_{n=1}^N \int \{y(\mathbf{x}_n - \boldsymbol{\xi}_n) - t_n\}^2 g(\boldsymbol{\xi}_n) d\boldsymbol{\xi}_n \quad (6.99)$$

を考える。変分法（付録 D）を用いて、 E を関数 $y(\mathbf{z})$ について最小化することで、最適な $y(\mathbf{x})$ は、カーネル（6.46）を持った、Nadaraya-Watson カーネル回帰（6.45）の形になることを示せ。

6.20 (標準) [www] (6.66) と (6.67) の結果を確認せよ。

6.21 (標準) [www] 固定された非線形の基底関数を使ってカーネル関数が定義されたガウス過程による回帰モデルを考え、その予測分布が 3.3.2 節で得られたベイズ線形回帰モデルに対する結果（3.58）と同じになることを示せ。両方のモデルがガウス予測分布を持つことに注意する、つまり、条件付き期待値と条件付き分散がそれぞれ等しくなることを示せばよい。条件付き期待値については、行列に関する等式（C.6）を、条件付き分散については（C.7）を用いよ。

6.22 (標準) N 個の入力ベクトル $\mathbf{x}_1, \dots, \mathbf{x}_N$ を持つ訓練集合と、 L 個の入力ベクトル $\mathbf{x}_{N+1}, \dots, \mathbf{x}_{N+L}$ を持つテスト集合があるような回帰問題を考える。また、関数 $t(\mathbf{x})$ 上の事前分布としてガウス過程を考える。 $t(\mathbf{x}_1), \dots, t(\mathbf{x}_N)$ が与えられたとき、 $t(\mathbf{x}_{N+1}), \dots, t(\mathbf{x}_{N+L})$ の同時予測分布を導け。この分布の、ある t_j ($N+1 \leq j \leq N+L$ とする) についての周辺分布を考えたとき、それは通常のガウス過程による回帰の結果（6.66）と（6.67）に一致することを示せ。

6.23 (標準) [www] ガウス過程による回帰モデルで、目標変数 \mathbf{t} の次元が D であるようなものを考える。入力ベクトル $\mathbf{x}_1, \dots, \mathbf{x}_N$ を持つ訓練集合と、対応する目標変数の値の集合 $\mathbf{t}_1, \dots, \mathbf{t}_N$ が与えられたとき、テストデータ \mathbf{x}_{N+1} に対する、 \mathbf{t}_{N+1} の条件付き分布を導け。

6.24 (基本) 対角行列 \mathbf{W} で、その要素が $0 < W_{ii} < 1$ を満たすものは、正定値であることを示せ。また、2 つの正定値行列の和は、やはり正定値になることを示せ。

6.25 (基本) [www] ニュートン-ラフソン法の公式（4.92）を用いて、ガウス過程による分類モデルに対する、事後分布のモード \mathbf{a}_N^* を求めるための逐次更新の公式（6.83）を導け。

6.26 (基本) (2.115) の結果を用いて、ガウス過程による分類モデルに対する事後分布 $p(a_{N+1} | \mathbf{t}_N)$ の平均（6.87）と分散（6.88）を導け。

6.27 (難問) ガウス過程による分類モデルのラプラス近似による対数尤度関数（6.90）を導け。また、（6.91）と（6.92）、および（6.94）を対数尤度の勾配を用いて表せ。

第7章 疎な解を持つカーネルマシン

前の章では、非線形カーネルを用いたさまざまな学習アルゴリズムに触れた。そこでの大きな制限の一つは、カーネル関数 $k(\mathbf{x}_n, \mathbf{x}_m)$ をすべての訓練データ対 $\mathbf{x}_n, \mathbf{x}_m$ について計算しなければならないため、学習および予測時に非常に計算時間がかかる可能性があることである。そこで、本章では疎な解（sparse solution）を持ち、訓練データ点の一部だけに対してカーネル関数を計算することで新しい入力の予測ができるアルゴリズムを見ていくことにする。

まず、クラス分類、回帰、新規性検出などの分野で近年よく使われるようになったサポートベクトルマシン（SVM; support vector machine）について詳しく説明する。SVM の重要な特徴は、モデルパラメータがある凸最適化問題の解として求まるため、局所解があればそれが大域解にもなる点である。SVM についての議論はラグランジュ乗数法についての知識が必要となるので、なじみの薄い読者はここで付録 E を確認することをお薦めする。また、SVM に関するより詳しい情報は Vapnik (1995), Burges (1998), Cristianini and Shawe-Taylor (2000), Müller et al. (2001), Schölkopf and Smola (2002), Herbrich (2002) を参照のこと。

SVM は識別関数の一種であり、出力の事後確率は得られない。しかし、1.5.4 節において説明したように、一般には事後確率がわかると便利な場合も多い。一方、関連ベクトルマシン（RVM; relevant vector machine）（☞7.2 節）のように、SVM と同様に疎なカーネルに基づいた手法でありながら、ベイズ理論に基づき事後確率の推定ができる、かつ、SVM よりもさらに疎なモデルが得られる手法も存在する。

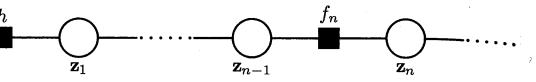
7.1 最大マージン分類器

まず次の線形モデルを用いて 2 値分類問題を解くことから SVM の話を始めよう。

$$y(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) + b. \quad (7.1)$$

ここで、 $\phi(\mathbf{x})$ はある固定された特徴空間変換関数であり、バイアスパラメータ b も陽

図 13.15 隠れマルコフモデルを表す単純化された因子グラフ.



の節で導いた α 再帰と同一である。なお、変数ノード z_n では、2つしか近隣のノードがないために、全く計算をしないことに注意しよう。

(13.47) を用いて、(13.48) から $\mu_{z_{n-1} \rightarrow f_n}(z_{n-1})$ を消去することができ、そうすると、 $f \rightarrow z$ メッセージについて以下の再帰式を得る。

$$\mu_{f_n \rightarrow z_n}(z_n) = \sum_{z_{n-1}} f_n(z_{n-1}, z_n) \mu_{f_{n-1} \rightarrow z_{n-1}}(z_{n-1}). \quad (13.49)$$

(13.46) の定義を思い出し、また、

$$\alpha(z_n) = \mu_{f_n \rightarrow z_n}(z_n) \quad (13.50)$$

を定義すると、(13.36) で与えられた α 再帰を得ることができる。さらに、 $\alpha(z_n)$ が以前定義されたものと同一であることを確かめる必要がある。これは、初期条件(8.71)を用い、また、 $\alpha(z_1)$ が、 $h(z_1) = p(z_1)p(x_1|z_1)$ で与えられること、そして、これが(13.37) で同一であることに注意すれば簡単に確認できる。最初の α が同一であり、同一の式により繰り返して計算されるので、すべての後に続く α も必ず同じ値を取る。

次に根ノードから葉ノードに逆に伝播されるメッセージについて考えよう。これらは次の形を取る。

$$\mu_{f_{n+1} \rightarrow f_n}(z_n) = \sum_{z_{n+1}} f_{n+1}(z_n, z_{n+1}) \mu_{f_{n+2} \rightarrow f_{n+1}}(z_{n+1}). \quad (13.51)$$

ここで、以前と同様に、 $z \rightarrow f$ のタイプのメッセージを消去した。変数ノードでは何の計算も行われないからである。定義(13.46)を使って、 $f_{n+1}(z_n, z_{n+1})$ を置き換える、

$$\beta(z_n) = \mu_{f_{n+1} \rightarrow z_n}(z_n) \quad (13.52)$$

を定義することにより、(13.38) で与えられる β 再帰を得ることができる。以前と同様に、 β 变数が同一であることを確認できる。これは、(8.70) は根ノードによって送られた最初のメッセージが $\mu_{z_N \rightarrow f_N}(z_N) = 1$ であることを示しており、それが 13.2.2 節で得られた $\beta(z_N)$ の初期化と同一であることに注意すればよい。

積和アルゴリズムは、さらに、一旦すべてのメッセージが求められた後に、周辺確率を求める方法を示している。特に(8.63) の結果は、ノード z_n における局所的な周辺確率が、入ってくるメッセージの積により与えられることを示している。すでに变数 $\mathbf{X} = \{x_1, \dots, x_N\}$ により条件付けられているので、同時確率は以下のように計算される。

$$p(z_n, \mathbf{X}) = \mu_{f_n \rightarrow z_n}(z_n) \mu_{f_{n+1} \rightarrow z_n}(z_n) = \alpha(z_n) \beta(z_n). \quad (13.53)$$

両辺を $p(\mathbf{X})$ で割ることにより、以下を得る。

$$\gamma(z_n) = \frac{p(z_n, \mathbf{X})}{p(\mathbf{X})} = \frac{\alpha(z_n) \beta(z_n)}{p(\mathbf{X})}. \quad (13.54)$$

これは、(13.33) と一致する。(13.43) の結果も同様に(8.72) から導くことができる(☞演習 13.11)。

13.2.4 スケーリング係数

実際にフォワード-バックワードアルゴリズムを利用する前に、議論しておかなければならない重要な問題がある。再帰式(13.36)を見ると、各々のステップの新しい値 $\alpha(z_n)$ は、直前の値 $\alpha(z_{n-1})$ に $p(z_n|z_{n-1})$ と $p(x_n|z_n)$ を掛けることにより得られることがわかる。これらの確率はしばしば 1 に比べても小さく、鎖に沿って前向きに進んでいくにつれて、 $\alpha(z_n)$ の値は指数的な速さで急速にゼロに近づいていく可能性が高い。たとえ倍精度浮動小数点型で計算したとしても、常識的な長さの鎖(例えば 100 くらい)で、 $\alpha(z_n)$ の計算はすぐに計算機のダイナミックレンジを超ってしまうだろう。

独立同分布に従うデータの場合は、対数を取って尤度関数を計算することにより、この問題が起きるのを暗に避けることができた。残念ながら、この方法はここでは使えない。なぜなら、小さい数同士の積の和を取っているからである(実際には、図 13.7 の格子図にあるすべての可能な経路についての和を暗に計算していることになる)。そこで、 $\alpha(z_n)$ と $\beta(z_n)$ にスケーリングを施し、それらの値が 1 のオーダーに留まるようにする。これから見るように、EM アルゴリズムでこれらスケーリングを施した量を使う場合、対応するスケーリング係数同士が打ち消し合い消えてしまう。

(13.34)において、 x_n までのすべての観測変数と潜在変数 z_n の同時分布を表す量として、 $\alpha(z_n) = p(x_1, \dots, x_n, z_n)$ を定義した。正規化された α の定義は以下で与えられる。

$$\hat{\alpha}(z_n) = p(z_n|x_1, \dots, x_n) = \frac{\alpha(z_n)}{p(x_1, \dots, x_n)}. \quad (13.55)$$

この値は数値計算において良い振る舞いをすることが期待できる。なぜなら、どの n の値に対しても K 個の変数上の確率分布であるからである。スケーリングを施した α 变数と、もともとの α 变数とを関係付けるために、観測変数上の条件付き分布によつて定義されるスケーリング係数を導入する。

$$c_n = p(x_n|x_1, \dots, x_{n-1}). \quad (13.56)$$

乗法定理より以下を得る。

$$p(x_1, \dots, x_n) = \prod_{m=1}^n c_m. \quad (13.57)$$

$$k(\mathbf{x}, \mathbf{x}') = ck_1(\mathbf{x}, \mathbf{x}')$$
(6.13)

$$k(\mathbf{x}, \mathbf{x}') = f(\mathbf{x})k_1(\mathbf{x}, \mathbf{x}')f(\mathbf{x}')$$
(6.14)

$$k(\mathbf{x}, \mathbf{x}') = q(k_1(\mathbf{x}, \mathbf{x}'))$$
(6.15)

$$k(\mathbf{x}, \mathbf{x}') = \exp(k_1(\mathbf{x}, \mathbf{x}'))$$
(6.16)

$$k(\mathbf{x}, \mathbf{x}') = k_1(\mathbf{x}, \mathbf{x}') + k_2(\mathbf{x}, \mathbf{x}')$$
(6.17)

$$k(\mathbf{x}, \mathbf{x}') = k_1(\mathbf{x}, \mathbf{x}')k_2(\mathbf{x}, \mathbf{x}')$$
(6.18)

$$k(\mathbf{x}, \mathbf{x}') = k_3(\phi(\mathbf{x}), \phi(\mathbf{x}'))$$
(6.19)

$$k(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T \mathbf{A} \mathbf{x}'$$
(6.20)

$$k(\mathbf{x}, \mathbf{x}') = k_a(\mathbf{x}_a, \mathbf{x}'_a) + k_b(\mathbf{x}_b, \mathbf{x}'_b)$$
(6.21)

$$k(\mathbf{x}, \mathbf{x}') = k_a(\mathbf{x}_a, \mathbf{x}'_a)k_b(\mathbf{x}_b, \mathbf{x}'_b).$$
(6.22)

ここで、 $c > 0$ は定数であり、 $f(\cdot)$ は任意の関数、 $q(\cdot)$ は非負の係数をもつ多項式、 $\phi(\mathbf{x})$ は \mathbf{x} から \mathbb{R}^M への関数、 $k_3(\cdot, \cdot)$ は \mathbb{R}^M で定義された有効なカーネル、 \mathbf{A} は対称な半正定値行列、 \mathbf{x}_a と \mathbf{x}_b は $\mathbf{x} = (\mathbf{x}_a, \mathbf{x}_b)$ であるような変数（必ずしも互いに素である必要はない）、また、 k_a と k_b はそれぞれの特徴空間において有効なカーネル関数であるとする。

これらの性質を用いると、特定の応用先に適した、より複雑なカーネルを構成することが可能になる。なお、カーネル $k(\mathbf{x}, \mathbf{x}')$ は、対称で、半正定値であり、また、適用先の問題領域における \mathbf{x} と \mathbf{x}' の適切な類似度となっていることが必要である。ここでは、いくつかのよく使われるカーネル関数を紹介する。より詳細な「カーネル設計」の方法については、Shawe-Taylor and Cristianini (2004) を参照するとよい。

単純な多項式カーネル $k(\mathbf{x}, \mathbf{x}') = (\mathbf{x}^T \mathbf{x}')^2$ は 2 次の項のみを含んでいたが、少し一般化して、定数 $c > 0$ を用いて $k(\mathbf{x}, \mathbf{x}') = (\mathbf{x}^T \mathbf{x}' + c)^2$ のようなカーネルを考えると、対応する特徴空間への写像 $\phi(\mathbf{x})$ が、定数の項と、1 次の項も持つようになります。また、 $k(\mathbf{x}, \mathbf{x}') = (\mathbf{x}^T \mathbf{x}')^M$ は M 次の項すべてを持つ。例えば、 \mathbf{x} と \mathbf{x}' を 2 つの画像とすると、このカーネル関数は、片方の画像中のあらゆる M 個の組み合わせのピクセルと、もう片方の画像中の M 個のピクセルとの積の、ある重み付き和となる。これも同様に、 $c > 0$ を用いて $k(\mathbf{x}, \mathbf{x}') = (\mathbf{x}^T \mathbf{x}' + c)^M$ とすることで、 M 次までのすべての次数の項を含むように一般化することができる。(6.17) と (6.18) の結果を利用すれば、これらはすべて有効なカーネル関数であることがわかる。

もうひとつのよく使われるカーネル関数としては、以下の、ガウスカーネルと呼ばれるものがある。

$$k(\mathbf{x}, \mathbf{x}') = \exp(-\|\mathbf{x} - \mathbf{x}'\|^2/2\sigma^2).$$
(6.23)

ただし、カーネル法の文脈では、これは確率密度関数としては解釈されず、したがって、正規化のための定数は省かれていることに注意する。これが有効なカーネルであることは、括弧内の平方を、

$$\|\mathbf{x} - \mathbf{x}'\|^2 = \mathbf{x}^T \mathbf{x} + (\mathbf{x}')^T \mathbf{x}' - 2\mathbf{x}^T \mathbf{x}'$$
(6.24)

のように展開したものを利用して、次の変形

$$k(\mathbf{x}, \mathbf{x}') = \exp(-\mathbf{x}^T \mathbf{x}/2\sigma^2) \exp(\mathbf{x}^T \mathbf{x}'/\sigma^2) \exp(-(\mathbf{x}')^T \mathbf{x}'/2\sigma^2)$$
(6.25)

を行い、さらに、(6.14) と (6.16)、および線形カーネル $k(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T \mathbf{x}'$ が有効であることを用いると示すことができる。なお、ガウスカーネルに対応する特徴ベクトルは無限次元である（☞演習 6.11）。

ガウスカーネルは、必ずしもユークリッド距離に限定されたものではなく、(6.24) のカーネル置換を用いて、 $\mathbf{x}^T \mathbf{x}'$ を非線形カーネル $\kappa(\mathbf{x}, \mathbf{x}')$ で置き換えれば、次のようなカーネルが得られる。

$$k(\mathbf{x}, \mathbf{x}') = \exp\left\{-\frac{1}{2\sigma^2}(\kappa(\mathbf{x}, \mathbf{x}) + \kappa(\mathbf{x}', \mathbf{x}') - 2\kappa(\mathbf{x}, \mathbf{x}'))\right\}.$$
(6.26)

カーネル法の考え方によって得られる重要な利点は、入力が実数値ベクトルだけではなく、記号であるような場合にも適用できることである。実際に、グラフ、集合、文字列、テキスト文書などのさまざまな対象に対して、カーネル関数が定義されている。例えば、対象として、ある集合を考え、この集合のすべての部分集合で構成される、ベクトル形式を持たない入力空間を定義する。 A_1 と A_2 をこのような部分集合とすると、カーネル関数のひとつの定義としては、以下のようなものが考えられる。

$$k(A_1, A_2) = 2^{|A_1 \cap A_2|}.$$
(6.27)

ここで、 $A_1 \cap A_2$ は A_1 と A_2 の共通集合とし、 $|A|$ を A に含まれる要素の数とする。これは、ある特徴空間における内積になっていることが示されるため、有効なカーネル関数である（☞演習 6.12）。

また、別の強力なアプローチとして、確率的生成モデルからカーネル関数を構成する方法がある (Haussler, 1999)。これによって、生成モデルを分類に用いることができるようになる。生成モデルは、欠損データを自然に扱うことができ、また、隠れマルコフモデル (hidden Markov model) などを用いれば可変長の配列を扱うことができる。一方、識別モデルは、分類問題においては生成モデルよりも一般に性能が良いことが知られており、したがって、これら 2 つのアプローチを組み合わせることは、しばしば興味の対象となる (Lasserre et al., 2006)。これを実現する方法のひとつとして考えられるのは、生成モデルを用いてカーネルを定義し、このカーネルを用いて識別アプローチをとるという方法である。