第2章　ガイダンスの作成技術

2.1　はじめに [1]

第 1.2 節でも述べたように、ガイダンスでは目的変数と説明変数の関係式を何らかの方法で事前に作成しておき、それを最新初期時刻の数値予報モデルから算出した説明変数に適用することで予測値を作成する。この関係式を導く方法には様々なものがあるが、代表的な方法は本章で述べる線形重回帰やニューラルネットワーク、カルマンフィルタなどの統計手法である。

統計手法を用いてガイダンスを開発する場合、その手法に関する理論を理解し、実装し、検証する必要があるが、これには一般に多くの時間と労力を要する。また、このような処理を独自に実装した場合には、バグや非効率的なプログラムが含まれる可能性もあり、開発効率の低下や、将来的にその手法を維持・管理していくためのコストの増加につながる。そこで近年のガイダンスの開発においては、統計処理を行うためのソフトウェアやパッケージといった統計ツールが用いられている。統計ツールを利用すれば、新たな手法によるガイダンスを比較的容易に導入できるため、開発に掛かる時間を大幅に削減できるとともに、プログラムの可読性が向上し、バグが混入する可能性も低くなる。近年では、統計処理を行うための様々なツールが無償で利用できるようになっており、これらのツールはガイダンスの開発を効率的に進める上で不可欠な存在となっている。

一方で、統計ツールを利用すれば、その手法に関する知識がなくても予測値を作成できるため、不適切な手法を用いたり誤った入力データを与えたりすることや、開発の方向性を限定してしまうことにつながる。竹内ほか (1992) は、手法の意味を理解せずに統計ツールを利用したとしても、その結果を正しく利用することはできない上に、誤った判断をくだす危険があることを述べている。久保 (2012) は中身を理解せずに統計ツールを利用する手法をブラックボックス統計学と呼び、誤った手法を無自覚に用いたり統計や検証の結果を都合良く解釈したりするのではなく、データをよく見て目的に沿った統計モデルを構築することの重要性を述べている。開発においては、十分に注意していたとしても、手法の妥当性よりも目先の予測精度を重視してしまいがちである。しかしそれでは、数値予報モデルの予測特性の変化などにより、予測精度が大きく低下したり、不自然な予測値が出力されたりする可能性が高まり、将来的な維持・管理コストの増加につながる。ガイダンスの予測精度を長期的に維持・向上させるためには、統計ツールを有効に活用するとともに、用いられている統計手法について正しく理解し、正しく使用することが重要である。

本章では、ガイダンスの開発・管理に携わっている者、またはこれから携わる者を対象として、本稿執筆時（2018 年現在）の気象庁のガイダンスに用いられている作成技術について、統計手法の理論を中心に述べる。統計手法を理解する上では統計や数学の高度な知識が必要となる場合もあるが、ここではガイダンスに利用する上で必要となる知識に絞って簡潔に述べることにしたい。より詳しく知りたい方は統計学や機械学習、時系列解析等に関する解説書を適宜参照していただきたい。

本章の構成は以下のとおりである。まず第 2.2 節では、ガイダンスに用いられている様々な手法を分類しながら、ガイダンスに用いられている手法を概観する。続いて第 2.3 節では、本章で述べる統計手法を理解する上で必要となる統計の基礎について述べる。第 2.4 節から第 2.9 節では、気象庁のガイダンスに用いられている手法と、各手法をガイダンスに利用する上での留意点を述べる。最後に第 2.10 節では、気象庁のガイダンスにはまだ利用されていないが、海外の気象機関では利用されているなど、今後新たな手法の導入を検討する上で候補となりうる手法について、その一部を紹介する。

参考文献

久保拓弥, 2012: データ解析のための統計モデリング入門 — 一般化線形モデル・階層ベイズモデル・MCMC. 岩波書店, 267 pp.

竹内啓, 矢島美寛, 廣津千尋, 藤野和建, 竹村彰通, 縄田和満, 松原望, 伏見正則, 1992: 自然科学の統計学. 東京大学出版会, 366 pp.

[1] 工藤 淳

2.2 手法の分類 [1]

ガイダンスの関係式を作成する手法には様々なものがあり、目的変数や説明変数の特性やガイダンスの利用形態等を考慮して選択される。1 つのガイダンスに使用される手法は 1 つとは限らず、多くのガイダンスでは複数の手法を組み合わせて予測を行っている。本節ではガイダンスの作成に利用されている手法を分類しながら各手法の特徴を述べる。

2.2.1 統計手法と診断手法

過去データを統計的に処理することによって目的変数と説明変数の関係式を作成する手法を統計手法と呼ぶ。代表的な統計手法として線形重回帰やニューラルネットワークなどが挙げられる。統計手法では、統計モデル（目的変数と説明変数の関係は線形である等）を事前に設定しておき、過去データを用いて統計モデルの最適な係数を決定する（係数を学習するともいう）。これに対して診断手法では、目的変数と説明変数の関係式は過去の研究や目的変数の定義などに基づいて決定される。例えば第一圏界面の定義に従って圏界面気圧の予測を行う手法や、消散係数の調査結果と視程の定義に基づいて視程の予測を行う手法、パーセル法に基づいて積乱雲の雲頂高度を予測する手法などが診断手法に分類される。圏界面気圧の予測のような純粋な診断手法では関係式の作成に過去データを必要としないというメリットがあるが、実際には視程分布予想や積乱雲頂高度など多くの診断手法型ガイダンスでは、予測精度を向上させるために過去データも用いて係数やパラメータの調整を行っている。

統計手法では平均的な精度が良くなるように係数が学習されるため、稀な現象は予測されにくいという特徴がある。これに対して診断手法では数値予報モデルの結果が直接的に予測に反映されるため、シャープな予測を行うことになり、数値予報モデルが極端な値を予測すれば、ガイダンスも極端な値を予測しうる、という特徴がある。ただし、予測精度としては統計手法を用いた方が良くなりやすい。

2.2.2 MOS と PPM

統計手法を用いて関係式を作成するとき、係数の学習に用いる説明変数が数値予報モデルから算出される場合を MOS (Model Output Statistics, Glahn and Lowry (1972)) といい、実況値や解析値から算出される場合を PPM[2](Perfect Prognosis Method[3]) という。例えば発雷確率を予測する場合、MOS では過去の数値予報モデルによる降水量や安定度などの予測と過去の発雷の有無の実況を用いて係数を学習するのに対し、PPM では過去の実況の降水量や、高層気象観測や解析値から算出した安定度などと過去の発雷の有無の実況を用いて係数を学習する。PPM は数値予報モデルの予測が完璧で誤差がないと考えた場合の予測であるのに対し、MOS では数値予報モデルに含まれる誤差（系統誤差、ランダム誤差）も考慮される [4] ため、予測精度としては一般に MOS の方が高くなる。これは例えば、地上気温と湿度の予測値を入力として降水の雨雪を判別するような場合を考えればわかりやすい。仮に数値予報モデルの地上気温に正バイアス（気温を高めに予測する系統誤差）があり、湿度はバイアスが 0 であったとすると、PPM を用いた場合には実況と比べて雨と予測する頻度が多くなる。これに対し MOS を用いた場合には、地上気温の正バイアスを考慮して雨雪が判別されるため、予測精度は高くなる。また、PPM では数値予報モデルが直接予測する要素（地上の風や気温、降水量など）を予測することには意味をなさない。なぜならば、もし数値予報モデルの予測が完璧ならば統計手法を用いる必要はなく、数値予報の出力そのものを用いれば良いからである。このため、これらの要素を予測するガイダンスは必然的に MOS を用いることになる。

2018 年現在の気象庁のガイダンスの多くは予測精度や予測対象の理由から MOS 方式を採用しているが、PPM 方式には以下のようなメリットがあり、モデルが精緻化されて誤差が小さくなってきた現在の数値予報モデルにおいては PPM も考慮する価値のある手法である。まず、PPM で学習した係数は数値予報モデルの誤差特性に影響されないため、数値予報モデルが改良されて予測誤差が減少していけば自然にガイダンスの精度も向上していくことが期待できる。これに対して、MOS で学習した係数は数値予報モデルの誤差特性に強く影響されるため、数値予報モデルの更新の影響を受けやすく、数値予報モデルの誤差特性が大きく変わる場合にはガイダンスの予測精度が大きく低下することもある。また、PPM は過去の実況や解析値を元に係数を学習するため、データが保存されている限り、10 年や 20 年といった長期間のデータに基づいてガイダンスを作成することもできる。このため PPM では、比較的稀な現象に対してもある程度多くのサンプル数を用いて係数を学習することができる。これに対して MOS で学習した係数は、誤差特性が異なる数値予報モデルに適用することは不適切であるため、数値予報モデルの特性が大きくは変化していないと考えられる期間（通常はせいぜい 3, 4 年程度）しか過去に遡って学習することができない。

一般に予報時間が進むほど数値予報モデルの予測誤

---

[1] 工藤 淳

[2] perfect prog, perfect prognostic または PP と呼ばれることも多い（Wilks (1995) など）。

[3] `http://glossary.ametsoc.org/wiki/Main_Page`

[4] 第 1.2 節でも述べたように、ガイダンスは系統誤差は補正できるがランダム誤差は補正できない。ただし、ランダム誤差がどれくらいあるか（予測と実況の相関がどれくらい弱いか）、という情報は考慮される。

差は大きくなるため、数値予報モデルのランダム誤差を正しく反映した予測では、初期時刻に近い時刻はシャープな予測を行い、予報時間が進むにつれてメリハリのない（気候値に近い）予測をすることになる。実際、予報時間で層別化（第 2.2.5 項を参照）された MOS はこのような予測を行う。このため、予報時間で層別化された MOS の予測を初期時刻順に同じ対象時刻で並べた場合、気象場の予測が変わらなかったとしても、初期時刻が新しくなるにつれて予測がシャープになる。これは予報後半ほどランダム誤差が大きいという数値予報モデルの特徴を正しく反映させた確率予測を行うような場合には望ましい特性であるが、そうでない場合には、初期時刻が新しくなるにつれて対象とする現象が強まったかのような印象を与えてしまう。これに対して PPM では、数値予報モデルのランダム誤差を考慮しないため、常に数値予報モデルと同程度にシャープな予測をすることになる。初期時刻順に同じ対象時刻の PPM の予測を並べた場合、気象場の予測が変わらなければ、初期時刻に関わらず常に同じような強度の予測をすることになるため、PPM は現象の推移の把握に適している。

数値予報モデルに系統誤差がある場合、PPM を用いると予測誤差が大きくなってしまう。そこで予測時の入力データに数値予報モデルの出力値を用いる代わりに系統誤差が補正されたガイダンス値を用いることがある。例えば最大降水量ガイダンスでは、学習時には解析雨量から算出した平均降水量と最大降水量を用いてその比を表す関係を学習し、予測では数値予報モデルの降水量の代わりに平均降水量ガイダンスを入力として最大降水量を求めている。また降水種別ガイダンスや雪水比ガイダンスでは、地上気温の実況を用いて関係式を学習し、予測では格子気温ガイダンスを入力としている。ガイダンス値を予測時の入力データに用いることで、PPM の利点を活かしつつ系統誤差を軽減することができる。

MOS と PPM の中間的な方法として、数値予報の初期時刻に近い時刻で係数を学習し、その係数をそれ以外の時刻の予測にも適用するという手法もある [5]。空域予報で利用されている乱気流指数と着氷指数などはこの手法を採用している。空域予報では短時間の予測が特に重要であるため、初期時刻に比較的近い時刻のデータを利用して係数を学習し、その係数をほかの時刻の予測にも適用している。これにより、注目したい時刻では最適な予測がされつつ、それよりも先の予測時刻でも着目している時刻と同程度のメリハリを持って予測されるため、乱気流や着氷の予測域の推移が把握しやすくなっている。ただし数値予報モデルの系統誤差が予報時間とともに変化する場合には、予報時間によって予測特性が変化することに注意が必要である。

[5] この手法は pseudo-PP (Météo-France 2015) とも呼ばれている。PP は perfect prognosis の略。

2.2.3 一括学習と逐次学習

統計手法を用いて予測式の係数を学習する場合、過去の一定期間のデータを用意し、過去データによる予測値とそれに対応する実況値から求められる誤差関数を最小にするような係数を求めることになる。この時、用意した全ての過去データを用いて算出した誤差関数を最小にするように係数を決定する手法を一括学習（またはバッチ学習）という。これに対して、過去データを時系列に並べて 1 組ずつ与え、データが与えられる度に係数を更新する手法を逐次学習（またはオンライン学習）という。また、一括学習に近い手法としてミニバッチ学習と呼ばれる学習方法もある。ミニバッチ学習は過去データの中から一定数のデータをランダムに抽出して学習する方法で、主にニューラルネットワークの学習に用いられる。これについては第 2.6.6 項などで述べる。

本章で述べる統計手法の中では、線形重回帰とロジスティック回帰では一括学習が、ニューラルネットワークとカルマンフィルタでは逐次学習が用いられる場合が多いが、各統計手法の解説で述べるように、線形重回帰とロジスティック回帰でも逐次学習は可能で、ニューラルネットワークでも一括学習は用いられる。カルマンフィルタは基本的には係数を逐次更新することを前提とした手法であるが、一定期間で逐次学習してその後は係数を学習しないという手法を取るならば、一括学習ともいえる。ただしその場合には学習の最終日に最適化された係数で予測を行うことになるため、予測精度は線形重回帰や通常のカルマンフィルタと比べて低くなる。

一括学習を用いた場合、再学習するまで係数は変化せず、ガイダンスの予測特性は変わらない。このため、利用者にとってはガイダンスの特性が把握しやすいというメリットがある。しかし、季節変化や数値予報モデルの特性変化に追随できないため、季節による層別化（第 2.2.5 項を参照）が必要であったり、数値予報モデルの更新によりモデルの特性が大きく変化する場合には、ある程度長期間のデータを用いて係数を再学習したりする必要がある。これに対し逐次学習を用いた場合は、新しい実況データが得られる度に係数が変化しガイダンスの予測特性も変わる。このため、一括学習と比べるとガイダンスの特性は把握しにくくなるが、季節変化や数値予報モデルの変化に追随することができ、数値予報モデルの更新時には比較的短い期間で係数を最適化できる場合がある。ただし必ずしも速やかに追随するわけではなく、例えば大雨や強風など頻度の少ない現象を予測対象としている場合には、短い期間では学習に必要なサンプル数が得られないため、数値予報モデルの変化に適応するまでに数か月以上掛かる。学習速度を速めるように調整することもできるが、単に速めただけでは実況を後追いするだけになってしまい、予測精度は低下する。

逐次学習型のガイダンスでは、稀な現象を予測できなかった場合、次回はその現象を予測しようとして係数を大きく変化させる。しかしそのような稀な現象は続いては起きないことが多いため、係数が元の状態に戻るまで過剰な予測を続けてしまう、というようなことが生じうる。このため、稀な現象に対して逐次学習を用いると予測精度が低くなる場合がある。一括学習ではこのようなことは起こらないため、稀な現象に対してもある程度の予測精度を持つことになる。このように、比較的頻度が高い現象を予測する場合は逐次学習が適しており、稀な現象を予測する場合は一括学習が適している。

逐次学習型のガイダンスは、数値予報ルーチンを運用する場合や、過去のデータを用いて予報実験を行う場合には、数値予報モデルのデータと実況データ、前回の係数を適切な順番で与える必要があり、データの取り回しが複雑になる。また逐次学習を行う中で、係数が異常な値になってしまうこともあるため、係数や予測値をモニターし、異常な場合には係数を差し替える等の対応が必要になる。一括学習型ガイダンスの場合は数値予報モデルの更新時を除いては係数を更新する必要がないため、逐次学習型ガイダンスに比べて維持・管理する手間は少ない。

2.2.4 補正手法

降水量の実況の頻度分布は、弱い降水の頻度が非常に多く、強い降水は少ないという偏った形をしている。このような分布を持つデータに対して統計手法を用いると、平均的なスコアを良くするために、頻度の低い現象が予測されにくくなるという傾向がある[6]。しかしながら気象予測においては、頻度が低い現象（大雨、強風、低視程など）の予測が重要である場合が多く、このような現象に対しても実況と同程度の頻度で予測されることが望ましい。そこで、頻度バイアス補正と呼ばれる補正手法が用いられている。頻度バイアス補正では、統計手法により予測値を求めた後に、実況の頻度と予測の頻度が等しくなるように予測値の補正を行う。これにより、統計手法による予測では全く予測されない（頻度がゼロの）現象も学習期間内の実況と同程度に予測されるようになるため、空振りは増えるが的中も増え、稀な現象に対する精度は向上する。頻度バイアス補正は、平均降水量ガイダンス、風ガイダンス、視程ガイダンスなど、逐次学習型のガイダンスでかつ、実況の分布に偏りがあり、頻度が低い現象の予測が重要なガイダンスに利用されている。頻度バイアス補正について詳しくは第 2.9 節で述べる。

頻度バイアス補正以外にも、降水確率ガイダンスを用いた平均降水量ガイダンスの補正、上空の気温予測等を用いた降水種別の補正、予測に最適な閾値を一定に近づけるための乱気流指数の補正など、各ガイダンスの特性や使用目的に応じて独自の補正を行っているガイダンスもある。これらについて詳しくは、第 4 章の各ガイダンスの解説で述べる。

[6] 予測に位置ずれがあった場合、実況は頻度が多い側（降水の場合は降水量が少ない方）にずれる事例が多くなる。統計手法では平均的なスコアを良くするため頻度が多い現象を予測しやすくなるため、頻度が低い現象の予測は少なくなってしまう。

2.2.5 層別化

層別化は、学習データを地点や時刻、季節などに分割し、それぞれに対して係数を学習し予測に利用する手法である。地点で層別化した場合には、地点毎に異なる係数を持つことになる。例えば気温予測の場合、東風が吹くと A 地点では気温が下がるが、B 地点では逆に気温が上がるなど、対象とする地点によって気温に影響を及ぼす現象や及ぼし方が変わる。このような場合、A 地点と B 地点で同じ係数を用いるとどちらの地点に対しても予測が不十分になってしまうため、地点で層別化して別々の係数を用いることが望ましい。同じ地点でも時刻によって数値予報モデルの特性が異なる場合には時刻（予報時間または予報対象時刻）で層別化する。例えば図 1.2.6 で示した新潟県粟島の気温予測のように、数値予報モデルの気温予測が日中は負バイアス、夜間は正バイアスを持つような場合、時刻で層別化することで気温を適切に予測できる。

特殊な層別化を用いている例として視程ガイダンスがある。視程ガイダンスでは地点や時刻のほかに天気（雨、雪、無降水）でも層別化している。これは視程の予測式が天気によって大きく変わることを考慮するためであるが、天気の予測が外れると視程の予測も大外れする可能性があることに注意が必要である。

一般に、統計手法を用いる場合、係数の学習に用いるデータは同一の特性を持つことが望ましく、適切な層別化は予測精度の向上をもたらす。ガイダンスが予測対象とする現象は、地点や時刻、季節などによって特性が変わるため、層別化は多くのガイダンスで利用されている。一方、層別化の数を増やせば増やすほど学習に利用できるデータ数は少なくなるため、係数の推定精度が低くなったり、稀な現象に対する学習が不十分になったりする。十分なサンプル数を確保するためには、単純には学習期間を長くすればよいが、MOS を用いる場合には学習期間をいくらでも長くすることはできない。この対策として発雷確率ガイダンスではクラスター分析が用いられている。クラスター分析を用いることで似た特性を持つ地点を一つにまとめることができるため、適切な層別化を行いつつ学習のためのサンプル数を増やすことができる。

2.2.6 アンサンブル手法

アンサンブル予報 (気象庁予報部 2016) や複数の数値予報モデルが利用できる場合、各予測に対するガイダンス値のアンサンブル平均等を利用することで、決

定論予測に基づくガイダンスよりも予測精度が高くなる場合が多い（第 5.1.2 項）。ただし最大降水量ガイダンスや視程ガイダンスのように、現象の最大値や最小値を予測するガイダンスの場合には、各予測に対するガイダンス値の単純な平均を用いた場合には予測精度が低下する。アンサンブル平均を用いる代わりに各メンバーの最大または最小値を用いることもできるが、一般に予測頻度が過大になり予測精度も低下する。このような場合には各メンバーの予測の分布に基づいた補正や予測を行う必要がある。

アンサンブル予報のスプレッドが適切で十分なメンバー数が得られる場合には、気温や風など量的な予測の誤差幅を示すことができる。また、ガイダンス値がある閾値を上回る割合などを利用することで、ロジスティック回帰やニューラルネットワーク等を用いなくても、予測の不確実性に応じた確率予測を行うこともできる。スプレッドが不十分な場合には誤差幅や確率予測も不十分になるが、第 1.4 節で示した米国の EKDMOS のように、スプレッドスキルの関係を利用した統計式で予測誤差を補正する手法も開発されている。

アンサンブル予報が利用できない場合でも、初期値の異なる複数の予測値をアンサンブルメンバーのように利用することで、アンサンブル平均が得られる。これを LAF (Lagged Average Forecast) 法 (Hoffman and Kalney 1983) と呼ぶ。LAF 法はアンサンブル予報を実行する必要がないため低コストで実施でき、予測精度の向上も期待できる。また、初期値方向に平滑化するため、初期値変わりの影響を軽減する効果がある。LAF 法は発雷確率ガイダンスで用いられており、決定論予測に基づくガイダンスと比べて予測精度や信頼度が向上することが確かめられている (高田 2008)。

参考文献

Glahn, H. R. and D. A. Lowry, 1972: The use of Model Output Statistics (MOS) in objective weather forecasting. *J. Appl. Meteor.*, **11**, 1203–1211.

Hoffman, R. N. and E. Kalney, 1983: Lagged average forecasting, an alternative to Monte Carlo forecasting. *Tellus*, **35A**, 100–118.

気象庁予報部, 2016: 確率的な気象予測のためのアンサンブル予報の課題と展望. 124 pp.

Météo-France, 2015: WMO Technical Progress Report on the Global Data-Processing and Forecasting System and Numerical Weather Predition Research Activities for 2015. *WMO/GDPFS*.

高田伸一, 2008: 発雷確率ガイダンス. 平成 20 年度数値予報研修テキスト, 気象庁予報部, 88–89.

Wilks, D. S., 1995: *Statistical methods in the atmospheric sciences*. Academic press, 467 pp.

2.3 ガイダンスに用いる統計の基礎[1]

ここでは次節以降で述べる手法を記述する上で必要となる統計の基礎について書く。本章で利用する主な変数と添字の定義を付録 2.3.A にまとめる。

2.3.1 確率変数と確率分布

サイコロを振るというような試行を N 回行ったとき、ある事象（この場合はサイコロの目）y が m 回起きたとする。N を大きくした場合に、y が起きた相対頻度 m/N がある一定値 $p(y)$ に近づくと考えられるとき、$p(y)$ を事象 y の確率という。

$$p(y) = \frac{m}{N} \tag{2.3.1}$$

サイコロを振る場合、それぞれの目が出る確率は例えば表 2.3.1 のようになる。確率は全て正の値を取り、起りうる全ての事象について和を取ると 1 になる。

サイコロを振る場合、どの目が出るかは確率的にしか知ることはできない。このように得られる結果が確率的に決まる変数を確率変数という。ある変数 y が確率変数である場合、y の具体的な値はまだ決まっておらず（サイコロを振る前の状態）、y の値は何らかの分布に従って確率的に与えられることになる。これを y の確率分布という。一方、変数 y が非確率変数である場合、y の値はサイコロを振って出た目のように既に決まった値となる。ある確率変数 Y が具体的な値 y を取る確率を $P_r(Y = y)$ または単に $p(y)$ と書く。

y が確率変数でその分布が ϕ であるとき、「y の分布が ϕ に従う」という意味で $y \sim \phi$ と書く。例えば、y が平均 μ、分散 σ^2 の正規分布 $N(\mu, \sigma^2)$ に従うならば、$y \sim N(\mu, \sigma^2)$ となる。確率変数 Y の分布がある連続的な関数で与えられる場合、その関数を確率密度関数といい、以下の式で定義される。

$$f(y) = \lim_{h \to 0} \frac{P_r(y - h \leq Y \leq y + h)}{2h} \tag{2.3.2}$$

例えば Y が平均 μ、分散 σ^2 の正規分布に従うならば、その確率密度関数は、

$$f(y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(y-\mu)^2}{2\sigma^2}\right] \tag{2.3.3}$$

となる。本章では記述を簡単にするため、以下では確率値と確率密度関数を同じ $p(y)$ を使って書くこととする。

表 2.3.1 サイコロの目と確率

目の値 y	1	2	3	4	5	6	合計
確率 $p(y)$	1/6	1/6	1/6	1/6	1/6	1/6	1

[1] 工藤 淳

2.3.2 標本平均と期待値

例えば成人男性の平均身長を知りたい場合、成人男性全員の身長を測ることは困難であるため、一部の成人男性の身長を測ることで全体の平均を推定することになる。このような場合、調査対象全体の集合を母集団といい、調査のために母集団から抽出した一部を標本（またはサンプル）という。このように、標本を抽出する目的は、母集団がもつ何らかのパラメータ（平均など）を推定することである。

ある試行を N 回行い $y_1,\ y_2,\ \cdots,\ y_N$ が得られたとき、得られた値の平均を標本平均という。標本平均 $\bar{y}$ を式で書くと、

$$\bar{y} = \frac{1}{N} \sum_{n=1}^{N} y_n \tag{2.3.4}$$

となる。表 2.3.1 のサイコロを N 回振った結果、1 が m_1 回、2 が m_2 回、$\cdots$、6 が m_6 回出たとする。この時、サイコロの目の標本平均 $\bar{y}$ は、

$$\bar{y} = \frac{1}{N} \sum_{y=1}^{6} y\, m_y = \sum_{y=1}^{6} y \frac{m_y}{N} \tag{2.3.5}$$

となる。

確率変数 Y が離散的な値 $y_1, \cdots, y_N$ を取り、その確率が $p(y_1), \cdots, p(y_N)$ であったとすると、Y の期待値 $E(Y)$ は次の式で定義される。

$$E(Y) = \sum_{n=1}^{N} y_n\, p(y_n) \tag{2.3.6}$$

また、確率密度が連続分布である場合の期待値は、

$$E(Y) = \int y\, p(y) dy \tag{2.3.7}$$

と定義される。X と Y が確率変数で、a と b が非確率変数である場合、

$$E(X + Y) = E(X) + E(Y) \tag{2.3.8}$$

$$E(aX + b) = aE(X) + b \tag{2.3.9}$$

という関係が成り立つ。

サイコロの場合、y_n をそれぞれの目の値、$p(y_n)$ をそれぞれの目が出る確率とすれば、サイコロの目の期待値は、

$$E(Y) = \sum_{n=1}^{6} y_n\, p(y_n) \tag{2.3.10}$$

となる。サイコロを振る回数を多くした場合、(2.3.1) 式より $m_y/N \to p(y)$ となるので、この場合 $\bar{y} \to E(Y)$ となる。この例からも分かる通り、期待値は平均（無限回試行した場合の標本平均）を表す値である。期待値と平均値は同じような意味を持つが、平均といった

場合には標本平均を指す場合があることに注意する。期待値は「サイコロを 1 回振ったときに出る目の期待値」など、1 回の試行に対して定義できるのに対し、標本平均は複数回試行したときに定義できるという違いがある。

サイコロを N 回振り、その標本平均を求めるとする。サイコロを振る前の時点ではどの目が出るか決まっていないため、n 回目にサイコロを振ったときに出る目 y_n や N 回振った結果の標本平均 $\bar{y}$ も確率変数である。このとき、同じサイコロを N 回振るならば、各試行の期待値は等しい値 μ になる。すなわち、n によらず $E(y_n) = \mu$ であるから、標本平均の期待値は

$$E(\bar{y}) = \frac{1}{N}\sum_{n=1}^{N} E(y_n) = \mu \tag{2.3.11}$$

となり、各試行の期待値と一致する。この例のように、平均が既知の値 μ である母集団から N 個の標本を抽出してその標本平均をとる場合、標本平均の期待値は母集団の平均（母平均）と一致する。母集団のパラメータ θ を標本から推定したとき、その推定値を $\hat{\theta}$ とすると、$E(\hat{\theta}) = \theta$ であるならば、$\hat{\theta}$ は θ の不偏推定量であるという。(2.3.11) 式からわかるように、標本平均は母平均の不偏推定量である。

2.3.3 標本分散と分散

ある試行を N 回行い $y_1, y_2, \cdots, y_N$ が得られたとき、得られた値に対する分散を標本分散という。標本分散 s^2 を式で書くと、

$$s^2 = \frac{1}{N}\sum_{n=1}^{N} (y_n - \bar{y})^2 \tag{2.3.12}$$

となる。また、

$$s = \sqrt{\frac{1}{N}\sum_{n=1}^{N} (y_n - \bar{y})^2} \tag{2.3.13}$$

を標本標準偏差という。分散と標準偏差は分布の広がりを表す値である。確率変数 y の分散 $V(y)$ は次のように定義される。

$$\begin{aligned} V(y) &= E\left[(y - E(y))^2\right] \\ &= E\left(y^2\right) - E(y)^2 \end{aligned} \tag{2.3.14}$$

期待値と同様に、分散は 1 回の試行に対して定義できる値であるのに対し、標本分散は複数回試行したときに定義できる値である。X を確率変数、a と b を非確率変数とすると、

$$V(aX + b) = a^2 V(X) \tag{2.3.15}$$

が成り立つ。また、2 つの確率変数 X と Y が独立、すなわち、X の結果に Y が依存しない場合、

$$V(X + Y) = V(X) + V(Y) \tag{2.3.16}$$

が成り立つ。

(2.3.11) 式と同様に、平均と分散が既知の値 μ, σ^2 である母集団から N 個の標本 $y_1, \cdots, y_N$ を抽出して標本分散 s^2 を求めたとすると、その期待値は、

$$E(s^2) = \frac{N-1}{N}\sigma^2 \tag{2.3.17}$$

となる（詳細は付録 2.3.B）。標本分散の期待値は母集団の分散（母分散）と一致しないため、標本分散は母分散の不偏推定量ではない。不偏推定量にするためには標本分散に $N/(N-1)$ を掛ければよい。

$$\hat{\sigma}^2 = \frac{1}{N-1}\sum_{n=1}^{N} (y_n - \bar{y})^2 \tag{2.3.18}$$

これを不偏分散という。

2.3.4 共分散と相関係数

共分散と相関係数（相関関数ともいう）は対応のある 2 つのデータの関係性を表す値である。2 つの変数 x と y について N 組のデータ $(x_1, y_1), \cdots, (x_N, y_N)$ が得られたとき、標本共分散 γ_{xy} は次のように定義される。

$$\gamma_{xy} = \frac{1}{N}\sum_{n=1}^{N} (x_n - \bar{x})(y_n - \bar{y}) \tag{2.3.19}$$

x と y の標本標準偏差を s_x, s_y としたとき、x と y の標本相関係数 ρ_{xy} は次のように定義される。

$$\rho_{xy} = \frac{\gamma_{xy}}{s_x s_y} \tag{2.3.20}$$

相関係数は $[-1, 1]$ に規格化された共分散であり、x と y が完全な相関を持つ場合に $+1$、相関が全くない場合に 0、完全な逆相関を持つ場合に -1 となる。

x と y が確率変数の場合、共分散 $Cov(x, y)$ と相関係数 $R(x, y)$ は次のように定義される。

$$Cov(x, y) = E\left[(x - E(x))(y - E(y))\right] \tag{2.3.21}$$

$$R(x, y) = \frac{Cov(x, y)}{\sqrt{V(x)V(y)}} \tag{2.3.22}$$

2.3.5 自己共分散と自己相関係数

y_n が確率変数で時系列データである場合、n 番目のデータから見て時間方向に k だけ離れたデータとの共分散と相関係数をそれぞれラグ k の自己共分散 $C_{n,n+k}$ と自己相関係数 $R_{n,n+k}$ といい、以下のように定義される。

$$C_{n,n+k} = Cov(y_n, y_{n+k}) \tag{2.3.23}$$

$$R_{n,n+k} = \frac{C_{n,n+k}}{\sqrt{V(y_n)V(y_{n+k})}} \tag{2.3.24}$$

y_n の平均、分散、共分散が時間変化しない場合、すなわち、

$$E(y_n) = E(y_{n+k}) \tag{2.3.25}$$

$$V(y_n) = V(y_{n+k}) \tag{2.3.26}$$
$$Cov(y_n, y_m) = Cov(y_{n+k}, y_{m+k}) \tag{2.3.27}$$

である場合、y_n は定常であるという。この時、

$$C_{n,n+k} = C_{0,k} \tag{2.3.28}$$
$$R_{n,n+k} = R_{0,k} \tag{2.3.29}$$

であり、$C_k \equiv C_{0,k}$, $R_k \equiv R_{0,k}$ と書くことにすると、

$$R_k = \frac{C_k}{\sqrt{V(y_0)V(y_k)}} = \frac{C_k}{C_0} \tag{2.3.30}$$

となる。

y_n が非確率変数で定常な時系列データである場合、

$$\gamma_k = \frac{1}{N}\sum_{n=1}^{N}(y_n - \bar{y})(y_{n+k} - \bar{y}) \tag{2.3.31}$$

をラグ k の標本自己共分散という。また、

$$\rho_k = \frac{\gamma_k}{\gamma_0} \tag{2.3.32}$$

をラグ k の標本自己相関係数という。

定常な時系列データが平均 0 で時間方向に相関を持たない（$k \geq 1$ で$\rho_k = 0$）場合、その時系列はホワイトノイズであるという。ホワイトノイズの自己相関係数は、

$$R_k = \begin{cases} 1 & (k = 0) \\ 0 & (k \geq 1) \end{cases} \tag{2.3.33}$$

となる。時系列がホワイトノイズでサンプル数（時系列の長さ）N が大きい場合、標本自己相関係数は平均 0、分散 $1/N$ の正規分布に従う（例えば Shumway and Stoffer 2000 の第 1 章）。

2.3.6 同時確率、条件付確率、ベイズの定理

ある事象 x が起きたという条件の下で事象 y が起きる確率を条件付確率といい、$p(y|x)$ と書く。このとき、y は確率変数であるのに対し、x は既に値が決まっているため非確率変数である。例えば図 2.3.1 の左図のように、x が起きる確率が 60%、起きない確率が 40%などと与えられたとき、$p(y|x) = 12/(12+48) = 1/5$ となる。

2 つの事象 x と y があって、x が起きた場合と起きなかった場合で y の起きる確率が等しいとき、事象 y は事象 x と独立であるという。例えば、サイコロを 2 回振ったとき、2 回目に出た目 y は 1 回目に出た目 x の影響を受けないだろうから、この場合 x と y は独立である。x と y が独立であるならば、$p(y|x) = p(y)$ であり、逆に、$p(y|x) \neq p(y)$ ならば x と y は従属であるという。図 2.3.1 の例では、左の図では x の発生の有無に関わらず y が起きる確率は $p(y) = 1/5$ であり x と y は

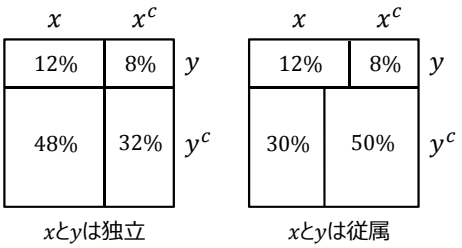

図 2.3.1 独立と従属の例。左は事象 x と y が独立、右は従属の場合。x^c, y^c はそれぞれ x と y の余事象を、数字は各事象が起きる確率を表す。

独立である。一方、右の図では x の発生の有無によって y が起きる確率が変わるため、x と y は従属である。

事象 x と y が同時に起きる確率を同時確率といい $p(x,y)$ と書く。$p(x,y)$ は、事象 x が起きた場合に事象 y が起きる確率であると同時に、事象 y が起きた場合に事象 x が起きる確率でもあるため、

$$p(x,y) = p(y|x)p(x) = p(x|y)p(y) \tag{2.3.34}$$

と書ける。これを確率の乗法定理という。図 2.3.1 の左図の例では $p(x,y) = 12/100 = 3/25$ で、これは $p(y|x)p(x) = 12/(12+48) \times (12+48)/100 = 3/25$ と等しい。このことは図 2.3.1 の右図に対しても成り立つことが確かめられる。

x と y が独立であるならば $p(y|x) = p(y)$ なので、

$$p(x,y) = p(x)p(y) \tag{2.3.35}$$

と書ける。つまり、独立な事象の同時確率は各事象の確率の積になる。これはシンプルな関係式ではあるが、多くの統計手法では観測データが独立であることを仮定することで N 個の観測データ $y_1, \cdots, y_N$ が得られた同時確率 $p(y_1, \cdots, y_N)$ を各確率の積 $p(y_1)\cdots p(y_N)$ として記述している。解説書の中には各観測が独立であると仮定していることを明示していない場合もあるので注意が必要である。当然だが、観測データが独立と見なせない場合にはこの関係を用いることはできない。

条件付確率の表記において、$p(x|y,z)$ と書いた場合には、y と z が同時に起きたという条件の下で x が起きる確率を意味し、$p(x,y|z)$ と書いた場合には z が起きたという条件の下で x と y が同時に起きる確率を意味する。この例のように、“|” の左右に書かれる変数は複数あっても良い。(2.3.34) 式に z が起きたという条件を追加すると、

$$p(x,y|z) = p(y|x,z)p(x|z) = p(x|y,z)p(y|z) \tag{2.3.36}$$

と書ける。(2.3.34) 式および (2.3.36) 式からすぐに、ベイズの定理と z で条件付けられたベイズの定理

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)} \tag{2.3.37}$$

$$p(x|y,z) = \frac{p(y|x,z)p(x|z)}{p(y|z)} \tag{2.3.38}$$

が得られる。

同時確率 $p(x,y)$ を x の全域で積分すると $p(y)$ になる。これを周辺化という。例えば図 2.3.1 の左の例では、$p(x,y) = 12\%$ と $p(x^c,y) = 8\%$ を足すと 20% となり、y が起きる確率になることが確かめられる。p を確率密度関数とし、(2.3.34) 式および (2.3.36) 式を x で積分すると、確率密度関数の周辺化の式

$$\begin{aligned} p(y) &= \int p(x,y)dx \\ &= \int p(y|x)p(x)dx = \int p(x|y)p(y)dx \end{aligned} \tag{2.3.39}$$

および

$$\begin{aligned} p(y|z) &= \int p(x,y|z)dx \\ &= \int p(y|x,z)p(x|z)dx = \int p(x|y,z)p(y|z)dx \end{aligned} \tag{2.3.40}$$

が得られる。

2.3.7 正規分布の確率密度関数

確率変数 w が平均 μ、分散 σ^2 の正規分布 $N(\mu,\sigma^2)$ に従うとき、その確率密度関数は、

$$p(w|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2\sigma^2}(w-\mu)^2\right] \tag{2.3.41}$$

と書ける。また、確率変数 $\boldsymbol{w}$ が K 次元ベクトルで、平均 $\boldsymbol{\mu}$、分散共分散行列 Σ の正規分布に従う場合の確率密度関数は、

$$\begin{aligned} &p(\boldsymbol{w}|\boldsymbol{\mu},\Sigma) \\ &= (2\pi)^{-\frac{K}{2}} |\Sigma|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\boldsymbol{w}-\boldsymbol{\mu})^T \Sigma^{-1}(\boldsymbol{w}-\boldsymbol{\mu})\right] \end{aligned} \tag{2.3.42}$$

と書ける。ここで $|\ |$ は行列式を、T は転置を表す。(2.3.42) 式を全区間で積分すると 1 になることから、

$$\int \exp\left[-\frac{1}{2}(\boldsymbol{w}-\boldsymbol{\mu})^T \Sigma^{-1}(\boldsymbol{w}-\boldsymbol{\mu})\right] d\boldsymbol{w} = (2\pi)^{\frac{K}{2}} |\Sigma|^{\frac{1}{2}} \tag{2.3.43}$$

となる。

2.3.8 回帰分析

目的変数 y と説明変数 $\boldsymbol{x}$ の組が与えられたとき、

$$\hat{y} = f(\boldsymbol{x},\boldsymbol{w}) \tag{2.3.44}$$

という y を近似する関係（以下ではこれを統計モデルまたは単にモデルと呼ぶ）を仮定し、係数 $\boldsymbol{w}$ を統計的に推定することを回帰分析または回帰という。ガイダンスの場合、例えば y は地上気温の観測値、$\boldsymbol{x}$ は数値予報で予測された地上気温、風、雲量などになる。どのような統計モデルを仮定するかによって、線形重回帰、ロジスティック回帰、ポアッソン回帰など様々な回帰手法がある（図 2.3.2）。どの統計モデルを採用するかは、データの特性から判断する。

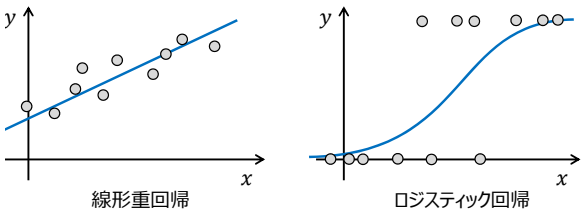

図 2.3.2 回帰の例

2.3.9 尤度と最尤法

統計モデルが (2.3.44) 式で与えられており、1 組の目的変数と説明変数のデータ $(y,\boldsymbol{x})$ が与えられたときに、係数 $\boldsymbol{w}$ を推定することを考える。このとき、$\boldsymbol{w}$ の推定値がどのような値かは分かっていないのだが、何か適当な値を与えれば、y が取る確率密度 $p(y|\boldsymbol{x},\boldsymbol{w})$ を求めることができる。今、データ $(y,\boldsymbol{x})$ が与えられているので、$p(y|\boldsymbol{x},\boldsymbol{w})$ は $\boldsymbol{w}$ のみの関数であり、これを $L(\boldsymbol{w})$ と書く。

$$L(\boldsymbol{w}) = p(y|\boldsymbol{x},\boldsymbol{w}) \tag{2.3.45}$$

このとき、$L(\boldsymbol{w})$ を尤度（または尤度関数）という。(2.3.45) 式は、左辺では $\boldsymbol{w}$ は変数として扱われているのに対し、右辺では確定した値として与えられているため、やや奇妙に見えるかもしれない。しかし、例えば $p(y|\boldsymbol{x},\boldsymbol{w})$ が正規分布で書けると仮定した場合には、

$$p(y|\boldsymbol{x},\boldsymbol{w}) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(y-f(\boldsymbol{x},\boldsymbol{w}))^2}{2\sigma^2}\right] \tag{2.3.46}$$

のように書けることから、数学的には $p(y|\boldsymbol{x},\boldsymbol{w})$ は $y,\boldsymbol{x},\boldsymbol{w}$ の関数であり、どれを変数と見なすかは自由に決めてよい。

例えば $\boldsymbol{w}$ を $\boldsymbol{w}_1$ と $\boldsymbol{w}_2$ に設定し、それぞれに対して尤度を求めたとき、$L(\boldsymbol{w}_1) > L(\boldsymbol{w}_2)$ であったとする。すなわち、$p(y|\boldsymbol{x},\boldsymbol{w}_1) > p(y|\boldsymbol{x},\boldsymbol{w}_2)$ であったとする。これは、$\boldsymbol{w}$ として $\boldsymbol{w}_2$ よりも $\boldsymbol{w}_1$ を採用した方が目的変数の値 y が得られる確率が高いことを意味している。例えばある 1 回の試行によって y という結果が得られた場合、y が得られる確率は低かったがたまたま y が得られたと考えるよりも、y が得られる確率が高かったので y が得られた、と考える方が自然だろう。今は $p(y|\boldsymbol{x},\boldsymbol{w}_1) > p(y|\boldsymbol{x},\boldsymbol{w}_2)$ としているので、$\boldsymbol{w}$ としては $\boldsymbol{w}_1$ の方、すなわち尤度が大きい方が尤もらしいという

ことができる。このような考えの下で尤度が最大になる時の $\boldsymbol{w}$ を係数の推定値とする手法を最尤法という。

次に、与えられたデータセットが 1 組ではなく、N 組 $(y_1, \boldsymbol{x}_1), \cdots, (y_N, \boldsymbol{x}_N)$ ある場合を考える。この場合の尤度は、

$$L(\boldsymbol{w}) = p(y_1, \cdots, y_N | \boldsymbol{x}_1, \cdots, \boldsymbol{x}_N, \boldsymbol{w}) \tag{2.3.47}$$

という同時確率で表されるが、それぞれの観測が独立であるならば、同時確率は各確率の積で表されることから、

$$L(\boldsymbol{w}) = \prod_{n=1}^{N} p(y_n | \boldsymbol{x}_n, \boldsymbol{w}) \tag{2.3.48}$$

と書ける。最尤法ではこれを最大にする $\boldsymbol{w}$ を求める。確率密度は常に正の値を取ることから、尤度を最大にする $\boldsymbol{w}$ を求めることは尤度の対数（対数尤度）を最大にする $\boldsymbol{w}$ を求めることと等しい。(2.3.48) 式の対数を取ると、

$$\ln L(\boldsymbol{w}) = \sum_{n=1}^{N} \ln p(y_n | \boldsymbol{x}_n, \boldsymbol{w}) \tag{2.3.49}$$

となる。対数を取ることで積が和になるため、次項で述べる最急降下法や確率的勾配降下法において微分の扱いが容易になる。

2.3.10 最急降下法と確率的勾配降下法

統計手法において、一定期間の過去データを用いて統計モデルの係数 $\boldsymbol{w}$ を決定することを学習という。係数を決定する場合、予測値 $f(\boldsymbol{x}, \boldsymbol{w})$ と実況値 y から計算される誤差関数 E を設定し、E を最小にするような係数を最適な係数とする、という手法がしばしば用いられる。誤差関数としては、平均二乗誤差 (MSE) や負の対数尤度（(2.3.49) 式に負号を付けた値）が用いられることから、一般に以下のような形をしている。

$$E = \sum_{n=1}^{N} E_n \tag{2.3.50}$$

$$E_n = h(y_n, \boldsymbol{x}_n, \boldsymbol{w}) \tag{2.3.51}$$

ここで h は MSE や負の対数尤度を表す関数である。求めたい係数は、上記の誤差関数を最小にする $\boldsymbol{w}$ であるから、E を $\boldsymbol{w}$ の各成分 w_k で微分して 0 になるときの $\boldsymbol{w}$ を求めれば良い。

$$\frac{\partial E}{\partial w_k} = \sum_{n=1}^{N} \frac{\partial E_n}{\partial w_k} = 0 \tag{2.3.52}$$

通常はこれを解析的に解くことはできないため、ニュートン・ラフソン法などを使って数値的に解くことになる。またはもっと単純に、E の w_k 方向の傾きに従ってある一定の正の割合 η で w_k を繰り返し変化させてもよい。このとき、繰り返しのステップを s 回行った時の係数を $\boldsymbol{w}^{(s)}$ と書くと係数更新のステップは以下のように書ける。

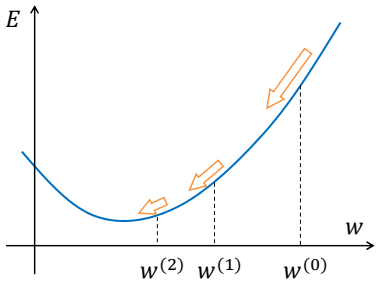

図 2.3.3 最急降下法による係数更新

$$w_k^{(s+1)} = w_k^{(s)} - \eta \sum_{n=1}^{N} \left. \frac{\partial E_n}{\partial w_k} \right|_{\boldsymbol{w}=\boldsymbol{w}^{(s)}} \tag{2.3.53}$$

これを適当な初期値 $\boldsymbol{w}^{(0)}$ から始めて繰り返し実行し、$w_k^{(s+1)}$ を用いて計算される誤差関数 E の値が s に依らず一定に近い値に収束したときの係数を $\boldsymbol{w}$ の推定値とする。これを最急降下法と呼ぶ 。(2.3.53) 式のように、全学習データを用いて係数を学習する方法を一括学習という。最急降下法による係数更新のイメージを図 2.3.3 に示す。

最急降下法の場合、1 回のステップ毎に学習データの数 N だけ微分計算を行う必要があるため計算量が多くなる。そこで、全学習データの中から各ステップ毎にサンプル $\mathcal{D}$（サンプル数 $N_{\mathcal{D}}$ 個）をランダムに選択し係数を更新することを考える。すなわち、

$$w_k^{(s+1)} = w_k^{(s)} - \eta \sum_{n \in \mathcal{D}} \left. \frac{\partial E_n}{\partial w_k} \right|_{\boldsymbol{w}=\boldsymbol{w}^{(s)}} \tag{2.3.54}$$

とする。この方法を確率的勾配降下法と呼ぶ。(2.3.54) 式のように、ランダムに選ばれたサンプルを用いて係数を学習する方法をミニバッチ学習という。

確率的勾配降下法では $N_{\mathcal{D}}$ 個のサンプルを抽出するが、抽出するサンプル数は 1 個でも良い。すなわち、

$$w_k^{(s+1)} = w_k^{(s)} - \eta \left. \frac{\partial E_n}{\partial w_k} \right|_{\boldsymbol{w}=\boldsymbol{w}^{(s)}} \tag{2.3.55}$$

とする。このとき学習データの番号を表す添字 n はランダムに選ばれたサンプルの番号を表す。ここで、学習データをランダムに選ぶのではなく、時系列順に与えることを考える。すなわち、(2.3.55) 式において、n を時刻 t で置き換える。さらに、係数の更新ステップは各時刻で 1 回だけ行うこととし、s も時刻 t で置き換えると、

$$w_k^{(t+1)} = w_k^{(t)} - \eta \left. \frac{\partial E_t}{\partial w_k} \right|_{\boldsymbol{w}=\boldsymbol{w}^{(t)}} \tag{2.3.56}$$

となり、時刻 t の係数と誤差関数を用いて次の時刻 $t+1$ の係数を求める式、すなわち係数を逐次更新する式と

なる。(2.3.56) 式のように、新たな学習データが得られる度に係数を学習する手法を逐次学習という。

最急降下法、確率的勾配降下法のいずれを用いる場合でも、η を適切に与えるとともに、説明変数を規格化しておく必要がある。η は、一括学習やミニバッチ学習においては学習の効率を決定するパラメータで、小さすぎると収束するまでに時間が掛かり、大きすぎると極小値付近で振動してしまいいつまでも収束しなくなる。一方、逐次学習においては 1 回の学習での係数の変動量を決めるパラメータで、小さすぎると季節変化や数値予報モデルの変更への追随が遅くなり、大きすぎると一つの事例で係数が大きく変わってしまい係数が安定しなくなる。このように、最急降下法や確率的勾配降下法において η は重要なパラメータであり、適切な値となるように調整する必要がある。

η が適切であったとしても、説明変数のオーダーに差があると学習がうまくいかない。例えば目的変数 y と 2 つの説明変数 x_1, x_2 に線形モデルを仮定し、y, x_1, x_2 のオーダーがそれぞれ、1, 1, 10^{-2} であったとする。各説明変数の寄与量 (x_1w_1 と x_2w_2) が同程度であるとすれば、w_1 と w_2 のオーダーはそれぞれ 1 と 10^2 になる。誤差関数に平均二乗誤差を用いた場合、誤差関数のオーダーは 1 で、$\partial E/\partial w_1$ と $\partial E/\partial w_2$ のオーダーはそれぞれ 1 と 10^{-2} になる。このとき $\eta = 10^{-2}$ を用いたとして (2.3.53) 式などに当てはめると、1 ステップの更新での $\boldsymbol{w}$ の変化量のオーダーは、w_1, w_2 それぞれに対して、10^{-2}, 10^{-4} となる。w_1, w_2 のオーダーは 1 と 10^2 であったから、w_1 に対して 10^{-2} 倍程度の幅で係数が更新されるのに対して、w_2 に対しては 10^{-6} 程度の幅でしか係数が更新されなくなってしまう。このため、w_2 の更新速度が非常に遅くなり、計算の効率が悪くなる。逆に、w_2 の更新速度を速めようとして $\eta = 10^2$ とすると、w_1 に対しては更新速度が速すぎていつまでも収束しなくなってしまう。このようなことを避けるために、説明変数を何らかの方法で規格化し、説明変数のオーダーを揃えておく必要がある。説明変数のオーダーを揃えるのではなく、係数毎に η を調整してもよいのだが、通常、η の最適値を探すために η を変えながら何度も調整する必要があるため、その度に全ての説明変数に対する η を調整すると煩雑になってしまう。説明変数を規格化しておけば、調整するパラメータは 1 つだけで良いため取扱いが容易になる。

説明変数を規格化する方法としては、次の 2 つがよく用いられる。

$$x_k \to \frac{x_k - \bar{x}_k}{s_k} \tag{2.3.57}$$

$$x_k \to \frac{x_k - x_{\min}}{x_{\max} - x_{\min}} \tag{2.3.58}$$

ここで、$\bar{x}_k$ と s_k は説明変数の標本平均と標本標準偏差、$x_{\min}$ と $x_{\max}$ は説明変数の最大値と最小値で、それぞれの値は学習データから求められるか、気候値や理論値、または実用的な値を設定している。(2.3.57) 式を用いた場合には平均 0、標準偏差 1 に、(2.3.58) 式を用いた場合には 0 ～ 1 の値に規格化される。

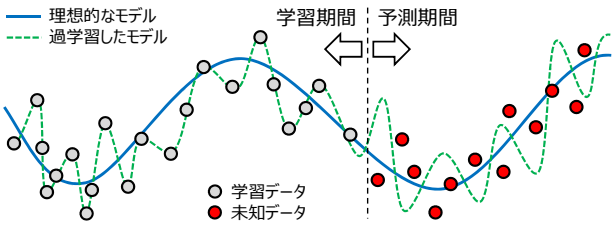

図 2.3.4　過学習のイメージ図

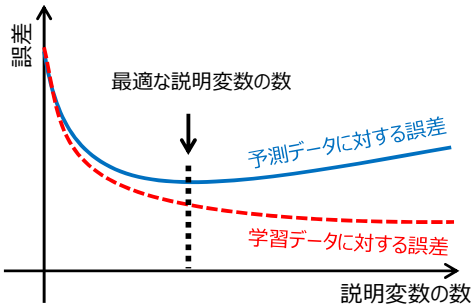

図 2.3.5　説明変数の数を変化させた場合の、学習データと予測データそれぞれに対する誤差のイメージ。

2.3.11　説明変数の選択

回帰分析において、目的変数 y が x の多項式 $y = a + bx + cx^2 + \cdots$ で表されると仮定した場合、多項式の次元を増やすほど学習データに最適化されたモデルが得られる（図 2.3.4 の破線）。しかしこのような回帰式を未知のデータに適用すると、図に示したように未知データに対する誤差（予測誤差）は大きくなってしまう。このように、学習データに対してモデルを最適化させ過ぎたために予測誤差が大きくなることを過学習（オーバーフィッティング）という。ガイダンスの目的は予測誤差を小さくすることであるから、図 2.3.4 の実線のようなある程度滑らかなモデルが理想的だと考えられる。

回帰分析では、過学習は説明変数の数が多すぎる場合に起きる。図 2.3.5 に、説明変数の数を変化させたときの学習データと予測データ（未知データ）それぞれに対する誤差のイメージを示す。一般に説明変数の数が少なすぎると、モデルの表現力が低いために学習データ・予測データのいずれに対しても誤差が大きくなる。そこで説明変数を増やしていくと、ある所までは学習データ・予測データとも誤差が小さくなっていくが、説明変数が多くなりすぎると、学習データに対する誤差は小さくなっていく一方で、予測データに対する誤差は増えていく。

予測誤差が小さいモデルを作成するためには、図 2.3.5 の矢印で示したように説明変数を適切に選択する必要がある。ただし予測誤差が最小になる説明変数の組み合わせを事前に知ることはできないため、学習データだけを用いて最適と思われる説明変数の組み合わせを選択することを考える。説明変数を選択する方法としては主に以下の 3 つが挙げられる。

- 赤池情報量基準 (Akaike 1973) 等の情報量基準を用いる方法
- 交差検証を用いる方法
- 主成分分析を用いる方法

以下ではそれぞれについて解説する。

(1) 赤池情報量基準を用いる方法

赤池情報量基準 (AIC) は以下の式で定義される。

$$\mathrm{AIC} = -2\ln L + 2K \tag{2.3.59}$$

ここで L は尤度、K はモデルの自由度である。回帰分析の場合、モデルの自由度は係数の数（説明変数の数にバイアス項の分の 1 を加えた数）である。右辺第 1 項は学習データに対する当てはまりの良さを表し、学習データに対する誤差が小さいほど値が小さくなる。右辺第 2 項はモデルの自由度を表し、回帰分析の場合は係数の数が少ないほど値が小さくなる。AIC は、学習データに対する誤差が小さく、係数の数が少ないほど値が小さくなる指標で、AIC が相対的に小さいモデルほど未知のデータに対する予測能力（汎化能力）が高いと考える。対数尤度 (2.3.49) 式は確率の対数（正の値）を学習データについて和を取った値であることから、学習データの数が多いほど負の対数尤度は小さくなり AIC も小さくなるが、これは学習データが多いほど予測能力が高いという意味ではない。AIC では同じ学習データに対して大小を比較することに意味がある。

学習データに対して、AIC が最も小さくなる説明変数の組み合わせを選択する方法としては、総当たり法や変数減少法、変数増加法、変数増減法が用いられる。総当たり法は、説明変数の全ての組み合わせに対して AIC を算出し、最も AIC が小さい組み合わせを選択する方法である。変数減少法では、初めに全ての説明変数を利用して AIC を算出し、次に説明変数を 1 つずつ除いてそれぞれの AIC を算出し、最も AIC が小さくなる説明変数を 1 つ削除する。これを繰り返し、どの説明変数を除いてもそれ以上 AIC が小さくならなくなった場合、それを最適な説明変数の組み合わせとする。変数増加法は変数減少法とは逆に説明変数を 1 つずつ増やしながら説明変数の組み合わせを決定する方法で、変数増減法は変数の増加・減少を行いながら最も AIC が小さくなる組み合わせを決定する方法である。説明変数の候補の数が計算機資源と比べてそれほど多くない場合には総当たり法を用いればよい。説明変数の候補が多すぎて総当たりでは計算に時間が掛かる場合は変数減少法等を用いる。AIC と同様の基準として、サンプル数が少ない場合に AIC を修正した AICc (Sugiura 1978) やブートストラップ法（第 2.3.12 項を参照）を用いて AIC を拡張した EIC (Ishiguro et al. 1997)、ベイズ統計学に基づく BIC (Schwarz 1978) など様々な情報量基準が提案されているが、ガイダンスに用いる場合は、経験的には AIC か次に述べる交差検証を用いれば十分である。

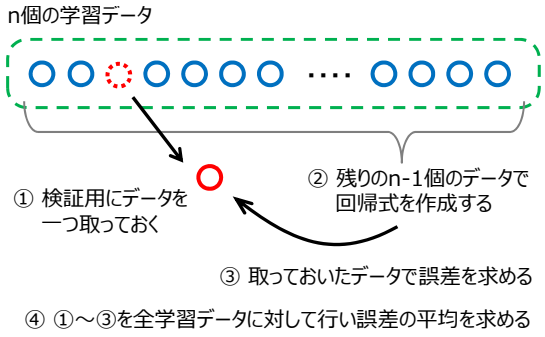

図 2.3.6 LOOCV の手順の模式図

(2) 交差検証を用いる方法

交差検証（クロスバリデーション (CV)）は学習データを用いて未知データに対する誤差を推定する手法の一つである。ガイダンスの開発においては、十分な数の検証用データが得られない場合に学習データから予測誤差を評価する手法として用いられることが多いが、説明変数の選択に利用することもできる。最も単純な CV として、LOOCV (Leave-One-Out CV) がある。図 2.3.6 に示すように、LOOCV では N 個の学習データの中から n 番目のデータ $(y_n, \boldsymbol{x}_n)$ を取っておいて、残りの $N-1$ 個のデータで予測式 $\hat{f}^{-n}$ の作成を行い、取っておいた n 番目のデータで検証を行う（二乗誤差等を求める）。これを N 個のデータ全てに対して行って誤差の平均を求めることで全体の誤差を評価する。式で書くと、LOOCV で平均二乗誤差 MSE を求める場合、

$$\mathrm{MSE}_{\mathrm{LOOCV}} = \frac{1}{N}\sum_{n=1}^{N}\left(y_n - \hat{f}^{-n}(\boldsymbol{x}_n)\right)^2 \tag{2.3.60}$$

となる。この計算を総当たり法や変数減少法等を用いて説明変数の組み合わせを変えながら実行し、CV で求められた誤差が最も小さくなる説明変数の組み合わせを選ぶ。

LOOCV のほかにも、k-fold CV とホールドアウト法と呼ばれる手法もある。k-fold CV は LOOCV と同様だが、学習データをデータ数の等しい k 個のグループに分けておき、1 つのグループを除いて予測式を作成したのち、取っておいた 1 つのグループで検証を行って誤差を求める。これを k 個全てのグループについて行い、誤差を平均することで全体の誤差を評価する。

k-fold CV と呼ぶ代わりに「1 か月抜き CV」のように、取っておくデータの期間を用いて呼ぶこともある。ホールドアウト法では N 個の学習データの中から k 個のデータをランダムに抽出して検証用データとし、残りの $N-k$ 個のデータで予測式を作成する。ホールドアウト法では予測式の作成と検証は 1 回のみ行い、LOOCV や k-fold CV のように全データをカバーするように検証を繰り返し行うことはしないため、厳密には交差検証とは呼ばれないが、シンプルで計算コストが少ないという利点がある。ただし繰り返し計算を行わないため、ランダム抽出に伴う検証結果の誤差（ばらつき）が大きくなってしまうので、計算時間に問題がない限りは LOOCV か k-fold CV を用いる。

説明変数の選択では、AIC を用いた方法と CV (LOOCV, k-fold CV) を用いた方法はどちらも同じような結果を与えることが多い。AIC を用いて説明変数を選択する場合はモデルの良さを評価する基準は 1 つしかないが、CV の場合は何でも良く、スレットスコアや RMSE など開発者が任意に設定できるという利点がある。ただし CV では予測式を繰り返し作成する必要があるため計算に時間が掛かる。

CV は学習データだけを用いて予測誤差を推定する手法であるが、検証用に取っておいたデータと予測式作成用のデータに相関がある場合には、実際に未知のデータを用いて検証を行った場合と比べて誤差がやや小さく見積もられてしまうことには注意しなければならない。日々の気象データの場合には一般に時間方向に強い相関があるため、LOOCV を用いた場合には上記の傾向は特に強くなる。このような場合には LOOCV ではなく k-fold CV を用いる。

CV を用いた場合には、CV による誤差を最小にする説明変数の組み合わせが 1 つだけ選ばれることになる [2]。この説明変数の組み合わせは全ての学習データを用いて選ばれており、その意味で、学習データに特化した説明変数の組み合わせが選ばれたということができる。しかし、もし異なる期間の学習データを用いることができたならば、異なる組み合わせの説明変数が選ばれる可能性もあるだろう。すなわち、説明変数の組み合わせを 1 つに限定してしまうことは、選択した説明変数の組み合わせが予測データに対しては最適ではないかもしれないという可能性を排除することになる。このため、CV によって学習データに特化されたモデルで誤差を推定した場合、誤差を過小評価してしまうことにつながる。このような場合、選択された説明変数による予測誤差を実際の値に近い値で推定する方法として、DCV（ダブル・クロスバリデーション）がある。LOOCV と同様に検証用にデータを 1 つ取っておく場合の DCV は以下の手順で行う。

1. N 個の学習データの中から検証用に n 番目のデータを一つ取り除く。
2. 残りの $N-1$ 個のデータに対して、AIC や CV に基づいて説明変数の選択を行い予測式を作成する。この予測式を $\hat{d}^{-n}$ と書く。
3. 全ての n に対して上記 1, 2 の手順を繰り返す。
4. 得られたそれぞれの予測式 $\hat{d}^{-n}$ と検証用に取っておいた n 番目のデータを用いて平均二乗誤差などを求め、それを予測誤差の推定値とする。

この手続きを式で書くと以下のようになる。

$$\mathrm{MSE}_{\mathrm{DCV}} = \frac{1}{N}\sum_{n=1}^{N}\left(y_n - \hat{d}^{-n}(\boldsymbol{x}_n)\right)^2 \qquad (2.3.61)$$

上記の手順では検証用データを 1 つだけ取っておいたが、k-fold CV と同様に学習データを k 個のグループに分けて DCV することもできる。DCV は CV と似た手法であるが、取り除いたデータとは独立に最適な説明変数が毎回選択されるため、(2.3.61) 式で用いられる予測式は n によって異なる。DCV ではモデル選択の誤差も含めて予測誤差を推定することができるため、CV や AIC 等で選択されたモデルを未知データに適用した場合の誤差を CV と比べて精度よく見積もることができる。

(3) 主成分分析を用いる方法

主成分分析では K 個の説明変数を M 個 ($M \leq K$) の合成変数に変換することで、元の説明変数が持つ情報をより少ない数の説明変数に縮約する。主成分分析は原理的には次のような手法である。ここでは簡単のために説明変数は x_1 と x_2 の 2 つだけだとする。2 つの説明変数を学習データについてプロットすると例えば図 2.3.7 のようになる。このデータを x_1–x_2 空間内の新たな軸に射影した時、射影したデータの不偏分散が最も大きくなる軸を z_1 とし、z_1 に直行する軸を z_2 とする。この例では x_1 と x_2 の相関が強く、x_1 と x_2 が持っていた情報はほぼ z_1 で説明することができるため、説明変数として z_1 だけ用いることで、説明変数の数を減らすことができる。

説明変数の数が K 個ある場合は、射影したデータの不偏分散が最も大きくなる軸を z_1 とし、z_1 に直行する軸の中で射影したデータの不偏分散が最も大きくなる軸を z_2 とする、ということを繰り返して z_K までの軸を決定する。m 番目の合成変数は次のように書ける。

$$z_m = \sum_{k=1}^{K} a_{mk}x_k\ , \quad \sum_{k=1}^{K} a_{mk}^2 = 1 \qquad (2.3.62)$$

ここで $\boldsymbol{a}_m$ は $\boldsymbol{x}$ を z_m に変換するための係数である。

実際の主成分分析では、N 個の学習データについて、規格化した説明変数の分散共分散行列の固有ベクトルと固有値を求めることになる。このとき、固有ベクトルが係数ベクトル $\boldsymbol{a}_m$ に、固有値 λ_m がその軸に射影し

[2] AIC を用いた場合でも、説明変数の組み合わせが 1 つだけ選ばれるという事情は同じである。

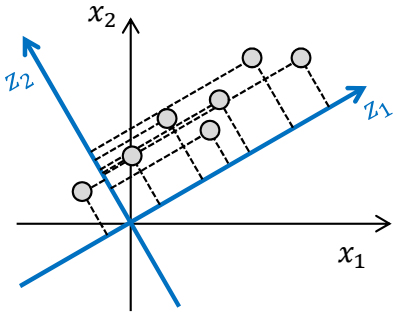

図 2.3.7　主成分分析の例

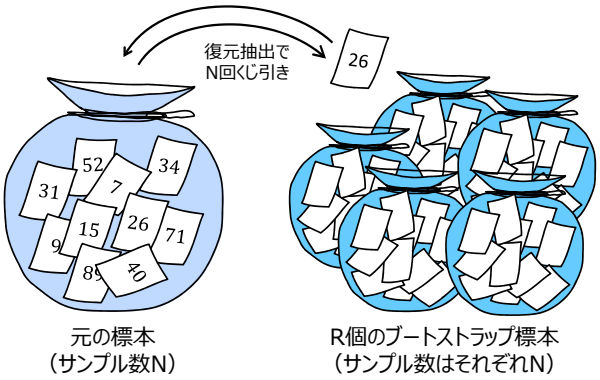

図 2.3.8　ブートストラップ法によるデータセット生成のイメージ

たデータの標本分散となるので、固有値が大きい順番に第一主成分、第二主成分、$\cdots$、第 K 主成分とする。固有値が大きい順番に $m = 1, 2, \cdots, K$ としたとき、

$$C = \frac{\sum_{m=1}^{M} \lambda_m}{\sum_{k=1}^{K} \lambda_k} \tag{2.3.63}$$

を第 M 主成分までの累積寄与率という。主成分分析では、累積寄与率がある値（例えば 0.7 など）になるまでの主成分を用いたり、あらかじめ設定した数（例えば元の説明変数の数の 7 割など）の主成分を用いたりすることで、元の説明変数が持つ情報をできるだけ保持しつつ説明変数の数を減らすことができる。

固有ベクトルと固有値を求める際に、分散共分散行列を用いる代わりに、説明変数の相関行列を用いる場合もある。相関行列を用いる場合は説明変数を事前に規格化する必要がないため扱いがシンプルになる。この場合の累積寄与率は以下のとおりである。

$$C = \frac{1}{K} \sum_{m=1}^{M} \lambda_m \tag{2.3.64}$$

2.3.12　ブートストラップ法

ブートストラップ法 (Efron 1979; Efron and Tibshirani 1986) はリサンプリングによって推定量の分布を見積もる手法である。例えば 100 人の成人男性の身長が得られたとして、これを元に母集団（成人男性全体）の身長の平均や分散（母平均、母分散）を推定することを考える。もし母集団の分布が分かっているならば、その分布に応じて母平均や母分散などを推定したり検定したりすることができる。これをパラメトリック法という。しかしながら母集団の分布が分かっているとは限らない。そのような場合に分布を推定する手法をノンパラメトリック法という。ブートストラップ法はノンパラメトリック法の一種であり、シンプルな手法で応用範囲が広く、様々な手法で利用されている。

ブートストラップ法によるパラメータ推定は以下のように行う。まず観測された 1 組（データ数 N）のデータセットから復元抽出によりランダムに N 個のデータを抽出（リサンプリング）する。復元抽出とは、例えばくじ引きをする場合であれば、一度引いたくじを戻して再びくじを引くことを意味する。このようにして抽出された N 個のデータからなるデータセットをブートストラップ標本という。これを R 回繰り返すことで R 組のブートストラップ標本を生成する（図 2.3.8）。得られた R 組のブートストラップ標本と元の 1 組の標本それぞれについて求めたい推定量を算出する。この $R+1$ 個の推定量により推定量の分布が得られる。また、$R+1$ 個の推定量の平均から、求めたい平均や分散などの推定量が得られる。R の大きさは計算に掛かる時間と推定の精度を考慮して設定することになるが、経験的には 500~5000 程度で十分である。

ブートストラップ法により、推定値の信頼区間（真値がある確率で含まれる区間）を求めることや、仮説検定を行うことができる。例えば平均値の 95%信頼区間を求める場合、$R+1$ 個のブートストラップ標本による平均値を昇順に並べたとき、下限から 2.5%に位置する値と、上限から 2.5%に位置する値を信頼区間の下限・上限とすることができる。同様の考え方で仮説検定を行うこともできる。

スレットスコアやバイアススコアなど、分割表から算出される検証スコアの信頼区間もブートストラップ法で求めることができる (Kane and Brown 2000)。サンプル数が N の検証スコアの信頼区間をブートストラップ法で求める場合には、FO, FX, XO, XX（定義は巻末の付録 A を参照）をくじと見なし、それぞれのくじが FO/N, FX/N, XO/N, XX/N の割合で含まれると考えて復元抽出により合計 N 回くじを引く。これを R 回繰り返し、それぞれのブートストラップ標本に対してスレットスコアなどを算出し、信頼区間を求める。このとき、1 つのブートストラップ標本を生成するために実際に乱数を N 回発生させてくじ引きすると、N に比例して計算量が多くなってしまう。そこで、くじを N 回引く代わりに FO, FX, XO, XX の割合に応じたサイズが N の 4 項分布に従う乱数を 1 回発生させる。これにより、サンプル数に関わらず 1 つのブートストラップ標本を一度に生成することができるため処理を高速化できる。

表 2.3.2 本章で利用する主な変数と添字の定義。表中の NN はニューラルネットワークを意味する。

文字	意味
x, X	説明変数、NN ではユニットへの入力データ
y, Y	目的変数、実況値、NN では教師データ
$\hat{y}, p$	ガイダンスの予測値（出力値）、確率値
w, W	回帰係数、係数、重み係数 （バイアス項を含める）
E	誤差関数（損失関数と呼ばれることも多い） $E()$ は期待値の意味で用いる
$V()$	分散
L	尤度
s	学習のステップ番号、標本標準偏差
t	時刻
f, g	関数、NN では活性化関数
ϕ	加重和
η	学習率
λ, β	正則化パラメータ
μ	平均
σ^2, Σ	分散、分散共分散行列

文字	最大値	意味
k	K	説明変数の番号、 NN では入力層のユニット番号
l	L	NN では中間層のユニット番号
m	M	NN では出力層のユニット番号
n	N	学習データの番号

付録 2.3.A 主な変数と添字の定義

本章で利用する主な変数と添字の定義を表 2.3.2 にまとめる。複数の統計手法を可能な限り同じ変数で表記するため、一般的なテキストやこれまでのガイダンスの解説に用いられてきた変数とは異なる場合があるので注意していただきたい。ここに書いていない文字やここでの定義と異なる意味で利用する際は本文中で適宜解説する。式の記述において、添字が同じ場合は和を取るというアインシュタインの縮約記法が用いられる場合があるが、統計手法においては添字が同じでも必ずしも和を取るとは限らないため、和を取る必要がある場合には和の記号 Σ を省略せずに表記する。説明変数 x と係数 w は、大文字で書いた場合は行列を、小文字の太字で書いた場合はベクトルを、小文字で書いた場合は行列またはベクトルの成分を表すものとする。

付録 2.3.B 標本分散の期待値

ここでは平均と分散が既知の値 μ, σ^2 である母集団から N 個の標本 $y_1, \cdots, y_N$ を独立に抽出して標本分散 s^2 を求める場合、その期待値が (2.3.17) 式

$$E(s^2) = \frac{N-1}{N}\sigma^2$$

となることを示す。

初めに標本平均 $\bar{y}$ の分散を求める。

$$V(\bar{y}) = \sum_{n=1}^{N} V\left(\frac{y_n}{N}\right) = \sum_{n=1}^{N} \frac{\sigma^2}{N^2} = \frac{\sigma^2}{N} \qquad (2.3.65)$$

ここで、独立な確率変数に対する分散の性質 (2.3.15) 式および (2.3.16) 式を用いている。標本平均 $\bar{y}$ の分散はまた、分散の定義 (2.3.14) 式と標本平均の期待値 (2.3.11) 式より、

$$V(\bar{y}) = E\left[(\bar{y} - E(\bar{y}))^2\right] = E\left[(\bar{y} - \mu)^2\right] \quad (2.3.66)$$

とも書ける。これらの式と標本平均の定義 (2.3.4) 式、分散の定義 (2.3.14) 式などを用いると、標本分散の期待値は

$$\begin{aligned} E(s^2) &= \frac{1}{N}\sum_{n=1}^{N} E\left[(y_n - \bar{y})^2\right] \\ &= \frac{1}{N}\sum_{n=1}^{N} E\left[\{(y_n - \mu) - (\bar{y} - \mu)\}^2\right] \\ &= \frac{1}{N}\sum_{n=1}^{N} E\left[(y_n - \mu)^2\right] + \frac{1}{N}\sum_{n=1}^{N} E\left[(\bar{y} - \mu)^2\right] \\ &\qquad - \frac{2}{N} E\left[(\bar{y} - \mu)\sum_{n=1}^{N}(y_n - \mu)\right] \\ &= \sigma^2 - E\left[(\bar{y} - \mu)^2\right] \\ &= \sigma^2 - \frac{\sigma^2}{N} = \frac{N-1}{N}\sigma^2 \end{aligned} \qquad (2.3.67)$$

となり、(2.3.17) 式が導かれる。

参考文献

Akaike, H., 1973: Information theory and an extension of the maximum likelihood principle. *B.N. Petrov, F. Csaki (Eds.), Second International Symposium on Information Theory, Akademiai Kiado, Budapest*, 267–281.

Efron, B., 1979: Bootstrap Methods: Another Look at the Jackknife. *The Annals of Statistics*, **7(1)**, 1–26.

Efron, B. and R. Tibshirani, 1986: Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. *Statistic Science*, **1**, 54–75.

Ishiguro, M., Y. Sakamoto, and G. Kitagawa, 1997: Bootstrapping log likelihood and EIC, an extension of AIC. *Annals of the Institute of Statistical Mathematics*, **49**, 411–434.

Kane, T. L. and B. G. Brown, 2000: Confidence intervals for some verification measures - a survey of several methods. *Preprints, 15th Conference on Probability and Statistics in the Atmospheric Sciences*, 46–49.

Schwarz, G., 1978: Estimating the dimension of a model. *The Annals of Statistics*, **6(2)**, 461–464.

Shumway, R. H. and D. S. Stoffer, 2000: *Time series analysis and its applications*. Springer, 549 pp.

Sugiura, N., 1978: Further analysis of the data by Akaike's information criterion and the finite corrections. *Communication in Statistics - Theory and Methods*, **7(1)**, 13–26.

2.4 線形重回帰 [1]

2.4.1 はじめに

線形重回帰は、2018 年現在の気象庁のガイダンスではあまり利用されていないものの、シンプルで最も基礎的な統計手法であり、正則化や説明変数の選択、利用上の注意点など線形重回帰に関連した周辺知識は、次節以降で述べるほかの統計手法を理解する上で有効である。本節では線形重回帰の基礎的な理論と周辺知識を述べる。

2.4.2 線形重回帰の基礎

線形重回帰は目的変数 y が説明変数 $\boldsymbol{x}$ と係数 $\boldsymbol{w}$ の線形結合で表されるとする回帰手法である。

$$\begin{aligned} y_n &= w_0 + x_{n1}w_1 + \cdots + x_{nK}w_K + \epsilon_n \\ &= \sum_{k=0}^{K} x_{nk}w_k + \epsilon_n \end{aligned} \tag{2.4.1}$$

ここで、n は学習データの番号、k は説明変数の番号、K は説明変数の数、ϵ_n は $x_{nk}w_k$ と独立な n 番目の観測 y_n のノイズである。w_0 はバイアス項で、$x_{n0} = 1$ とする。線形重回帰に関する多くのテキストでは $w_k x_{kn}$ というように、係数、説明変数の順番で表記することが多いが、本章では (2.4.1) 式のように説明変数、係数の順番で記述する。線形重回帰では、N 個の学習データが与えられたときに、次の仮定の下で係数を推定する。

仮定 1. モデルの線形性
　モデルが $y_n = \sum_{k=0}^{K} x_{nk}w_k + \epsilon_n$ と表されること
仮定 2. 説明変数は非確率変数
仮定 3. 観測ノイズの不偏性
　$E(\epsilon_n) = 0$ であること
仮定 4. 観測ノイズの等分散性
　各事例 n に依らず $V(\epsilon_n) = \sigma^2$ であること
仮定 5. 観測ノイズの独立性
　事例 n, m $(n \neq m)$ に対して、
　$E(\epsilon_n \epsilon_m) = E(\epsilon_n)E(\epsilon_m)$ であること
仮定 6. 説明変数に完全な多重共線性はない
　説明変数 x_{nk}, x_{nl} $(k \neq l)$ に対して、$x_{nk} \neq ax_{nl}+b$
　$(a, b$ は n に依らない定数) であること

仮定 2~5 は、観測ノイズの二乗和を最小にする $\boldsymbol{w}$ が係数の分散を最小にする不偏推定量（最良線形不偏推定量）になる [2] という、最小二乗原理（ガウス・マルコフの定理、廣津 (1992) など）を満たすために必要な仮定である。最小二乗原理に基づいて係数を決定する方法を最小二乗法という。

後で利用するために、上記の ϵ_n に対する仮定を y_n に対する仮定に書き換えておく。仮定 2, 3 と、係数 $\boldsymbol{w}$ は何らかの方法で確定させた値（非確率変数）であることより、

$$E(y_n) = \sum_{k=0}^{K} x_{nk}w_k \tag{2.4.2}$$

が成り立つ。また、ϵ_n は $x_{nk}w_k$ と独立であることから、$V(y_n) = V\left(\sum_{k=0}^{K} x_{nk}w_k\right) + V(\epsilon_n)$ となる。ここで $\boldsymbol{x}$ と $\boldsymbol{w}$ が非確率変数であることより $V(x_{nk}w_k) = 0$ なので、仮定 4 を用いると、

$$V(y_n) = \sigma^2 \tag{2.4.3}$$

が成り立つ。また、$n \neq m$ のとき、仮定 5 などを用いることで、

$$Cov(y_n, y_m) = 0 \tag{2.4.4}$$

が成り立つ。

以下では最小二乗法を用いて係数 $\boldsymbol{w}$ を決定する方法を示す。目的変数 y_n の推定値 $\hat{y}_n$ は

$$\hat{y}_n = E(y_n) = \sum_{k=0}^{K} x_{nk}w_k \tag{2.4.5}$$

であり、観測値 y_n と推定値 $\hat{y}_n$ の差（残差）の二乗和に 1/2 を掛けた値を誤差関数 E とする。

$$\begin{aligned} E &= \frac{1}{2}\sum_{n=1}^{N} \epsilon_n^2 \\ &= \frac{1}{2}\sum_{n=1}^{N} \left(y_n - \sum_{k=0}^{K} x_{nk}w_k \right)^2 \end{aligned} \tag{2.4.6}$$

最小二乗法では観測ノイズの二乗和を最小にする係数、すなわち上記の E が最小になる時の係数を求めたいので、E を係数の各成分 w_k で微分する。

$$\frac{\partial E}{\partial w_k} = -\sum_{n=1}^{N} \left(y_n - \sum_{i=1}^{K} x_{ni}w_i \right) x_{nk} \tag{2.4.7}$$

これが 0 になるときの $\boldsymbol{w}$ を係数の推定値 $\hat{\boldsymbol{w}}$ とすると、

$$\sum_{n=1}^{N} \left(x_{nk}y_n - x_{nk}\sum_{i=1}^{K} x_{ni}\hat{w}_i \right) = 0 \tag{2.4.8}$$

となる。これを行列形式で書くと次のようになる。

$$X^T\boldsymbol{y} - X^T X\hat{\boldsymbol{w}} = 0 \tag{2.4.9}$$

ここで X は x_{nk} を n, k 成分とする N 行 K 列の行列で、X^T は X の転置行列である。X は正方行列ではな

---

[1] 工藤 淳

[2] 観測ノイズの二乗和を最小にする $\boldsymbol{w}$ が最適な推定値であることは自明ではない。例えば絶対値の和や四乗の和を最小にする $\boldsymbol{w}$ でも良いように思える。しかしそうではなく、二乗和を最小にする $\boldsymbol{w}$ が最適な推定値であるという定理がガウス・マルコフの定理である。

いため逆行列を持たないが、X^TX は K 行 K 列の正方行列であるため逆行列を持ちうる。(2.4.9) 式を $\hat{\boldsymbol{w}}$ について解くと、

$$\hat{\boldsymbol{w}} = \left(X^TX\right)^{-1}X^T\boldsymbol{y} \tag{2.4.10}$$

となる。このとき、説明変数間に完全な多重共線性があると X がランク落ちしして X^TX が逆行列を持てなくなる（(2.4.10) 式の解が一意に決まらなくなる）ため、前述の仮定 6 が必要になる。(2.4.10) 式を計算することで線形重回帰の係数の推定値 $\hat{\boldsymbol{w}}$ が得られる。この $\hat{\boldsymbol{w}}$ を用いて、

$$\begin{aligned}\hat{\boldsymbol{y}} &= X\hat{\boldsymbol{w}} \\ &= X\left(X^TX\right)^{-1}X^T\boldsymbol{y} \\ &\equiv H\boldsymbol{y}\end{aligned} \tag{2.4.11}$$

と書いたとき、H をハット行列と呼ぶ。ハット行列の転置は、

$$\begin{aligned}H^T &= \left[X\left(X^TX\right)^{-1}X^T\right]^T \\ &= X\left(X^TX\right)^{-1}X^T = H\end{aligned} \tag{2.4.12}$$

であることから、ハット行列は対称行列である。

2.4.3 最小二乗法と最尤法の関係

第 2.3.9 項では、最尤法で係数が推定できることを述べた。線形重回帰は最小二乗法で係数を推定するが、最尤法を用いても同じ結果が得られることを以下で示す。

線形重回帰では観測ノイズの分布に対して、

- 不偏性: $E(\epsilon_n) = 0$
- 等分散性: n に依らず $V(\epsilon_n) = \sigma^2$
- 独立性: $n \neq m$ のとき $E(\epsilon_n\epsilon_m) = E(\epsilon_n)E(\epsilon_m)$

を仮定している。このような ϵ_n の分布として、正規分布 $N(0, \sigma^2)$ を考える。今、$y_n = \sum_{k=0}^{K} x_{nk}w_k + \epsilon_n$ であり、$\boldsymbol{w}$ と $\boldsymbol{x}_n$ が非確率変数として与えられたならば、ϵ_n は $\boldsymbol{w}$, $\boldsymbol{x}_n$ と独立であるから、$\epsilon_n|\boldsymbol{w}, \boldsymbol{x}_n \sim N(0, \sigma^2)$ である。ϵ_n を $y_n - \sum_{k=0}^{K} x_{nk}w_k$ で置き換え、非確率変数 $\sum_{k=0}^{K} x_{nk}w_k$ を右辺に移項すれば、

$$y_n|\boldsymbol{w}, \boldsymbol{x}_n \sim N\left(\sum_{k=0}^{K} x_{nk}w_k, \sigma^2\right) \tag{2.4.13}$$

となる。よって y_n の確率密度関数は、

$$p\left(y_n|\boldsymbol{w}, \boldsymbol{x}_n\right) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left[-\frac{1}{2\sigma^2}\left(y_n - \sum_{k=0}^{K} x_{nk}w_k\right)^2\right] \tag{2.4.14}$$

であり、観測ノイズが独立であることから、対数尤度は

$$\ln L(\boldsymbol{w}) = \sum_{n=1}^{N} \ln p\left(y_n|\boldsymbol{w}, \boldsymbol{x}_n\right)$$

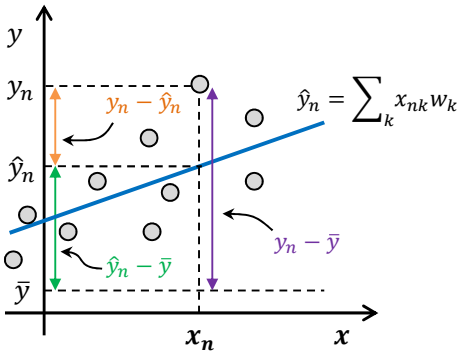

図 2.4.1 実況値 y_n、予測値 $\hat{y}_n$、実況の標本平均値 $\bar{y}$ の関係

$$= -\frac{N}{2}\ln\left(2\pi\sigma^2\right) - \frac{1}{2\sigma^2}\sum_{n=1}^{N}\left(y_n - \sum_{k=0}^{K} x_{nk}w_k\right)^2 \tag{2.4.15}$$

となる。N と σ^2 は定数なので、(2.4.15) 式を最大にする $\boldsymbol{w}$ は $\sum_{n=1}^{N}\left(y_n - \sum_{k=0}^{K} x_{nk}w_k\right)^2$ を最小にする $\boldsymbol{w}$ であることから、最尤法を用いた場合も最小二乗法を用いた場合と同じ結果が得られることになる。

2.4.4 決定係数

線形重回帰において、モデルの適合性を表す指標の一つに決定係数 R^2 がある。一般に用いられている決定係数の定義は下記の通りである。

$$R^2 \equiv \frac{\sum_{n=1}^{N}\left(\hat{y}_n - \bar{y}\right)^2}{\sum_{n=1}^{N}\left(y_n - \bar{y}\right)^2} \tag{2.4.16}$$

ここで $\hat{y}_n$ と $\bar{y}$ はそれぞれ、y_n の推定値（予測値）と標本平均である。決定係数の平方根 R は重相関係数と呼ばれる。y_n, $\hat{y}_n$, $\bar{y}$ の関係を図 2.4.1 に示す。もし回帰式によって実況値が完全に再現できた場合には、$y_n = \hat{y}_n$ であるから、$R^2 = 1$ となる。逆に、説明変数を全く用いなかった場合には回帰式は $\hat{y}_n = \bar{y}$ となるため、$R^2 = 0$ となる。すなわち決定係数は、いくつかの説明変数を用いて線形重回帰を行った場合、回帰式がどれくらい実況値を再現するかという指標である。決定係数の最大値は 1 で、1 に近いほどモデルが実況に適合しているといえる。(2.4.16) 式は以下のように変形できる（付録 2.4.A を参照）。

$$R^2 = 1 - \frac{\sum_{n=1}^{N}\left(y_n - \hat{y}_n\right)^2}{\sum_{n=1}^{N}\left(y_n - \bar{y}\right)^2} \tag{2.4.17}$$

この式の右辺第 2 項の分子は残差の二乗和である。残差の二乗和を最小にすることは R^2 を最大にすることと等しいので、最小二乗法では R^2 を最大にするように係数が決められるといえる。

(2.4.17) 式の右辺第 2 項の分母分子を N で割ると、

$$R^2 = 1 - \frac{\frac{1}{N}\sum_{n=1}^{N}\left(y_n - \hat{y}_n\right)^2}{\frac{1}{N}\sum_{n=1}^{N}\left(y_n - \bar{y}\right)^2} \tag{2.4.18}$$

である。右辺第 2 項の分母は定義より y_n の標本分散である。また、(2.4.5) 式より $\hat{y}_n = E(y_n)$ であるから、$y_n - \hat{y}_n$ の標本平均は 0 であることを考慮すれば、右辺第 2 項の分子は $y_n - \hat{y}_n$ の標本分散を表していることになる。サンプル数 N が限られている場合には、標本分散ではなく不偏分散を用いるように、決定係数の計算時にもサンプル数で割るのではなく、サンプル数から自由度（パラメータ数）を引いた数で割る。すなわち、

$$R^2_{adj} = 1 - \frac{\frac{1}{N-K-1}\sum_{n=1}^{N}(y_n - \hat{y}_n)^2}{\frac{1}{N-1}\sum_{n=1}^{N}(y_n - \bar{y})^2} \tag{2.4.19}$$

とし、R^2_{adj} を自由度調整済み決定係数という。

2.4.5 係数の推定精度

線形重回帰に限ったことではないが、少ない学習データを用いて学習した場合と多数の学習データを用いた場合とでは、係数の推定精度に差が出るであろうことは容易に想像できる。ここでは係数の推定精度に何が影響するかを示す。

係数の推定精度への影響を調べるため、最小二乗法で推定した係数 $\hat{\boldsymbol{w}}$ の分散共分散行列を求める。A が非確率変数からなる行列、$\boldsymbol{x}$ を確率変数からなるベクトルとしたとき、$V(A\boldsymbol{x}) = AV(\boldsymbol{x})A^T$ であることを利用すると、(2.4.10) 式より、

$$V(\hat{\boldsymbol{w}}) = \left(X^TX\right)^{-1}X^TV(\boldsymbol{y})X\left(X^TX\right)^{-1} \tag{2.4.20}$$

となる。$V(\hat{w}_k)$ は $V(\hat{\boldsymbol{w}})$ の k 行 k 列成分 $V(\hat{\boldsymbol{w}})_{kk}$ であることから、

$$V(\hat{w}_k) = \sum_{i,j}\left[\left(X^TX\right)^{-1}X^T\right]_{ki}V(\boldsymbol{y})_{ij}\left[X\left(X^TX\right)^{-1}\right]_{jk} \tag{2.4.21}$$

と書ける。ここで、(2.4.3) 式と (2.4.4) 式より、$V(\boldsymbol{y})_{ij} = \sigma^2\delta_{ij}$ なので、

$$\begin{aligned} V(\hat{w}_k) &= \sigma^2\left[\left(X^TX\right)^{-1}X^TX\left(X^TX\right)^{-1}\right]_{kk} \\ &= \sigma^2\left[\left(X^TX\right)^{-1}\right]_{kk} \end{aligned} \tag{2.4.22}$$

となる。続いて $\left(X^TX\right)^{-1}$ の具体的な値を求めてみるのだが、ここでは簡単のために説明変数が 2 個だけでバイアス項のないモデル

$$\begin{aligned} &y_n = x_{n1}w_1 + x_{n2}w_2 + \epsilon_n \\ &\sum_{n=1}^{N}x_{n1} = \sum_{n=1}^{N}x_{n2} = 0 \end{aligned} \tag{2.4.23}$$

を考えると、

$$\begin{aligned} X^TX &= \begin{pmatrix} x_{11} & \cdots & x_{N1} \\ x_{12} & \cdots & x_{N2} \end{pmatrix}\begin{pmatrix} x_{11} & x_{12} \\ \vdots & \vdots \\ x_{N1} & x_{N2} \end{pmatrix} \\ &= \begin{pmatrix} \sum_n x_{n1}^2 & \sum_n x_{n1}x_{n2} \\ \sum_n x_{n2}x_{n1} & \sum_n x_{n2}^2 \end{pmatrix} \end{aligned} \tag{2.4.24}$$

であるので、

$$\begin{aligned} \left(X^TX\right)^{-1} &= \frac{1}{D}\begin{pmatrix} \sum_n x_{n2}^2 & -\sum_n x_{n1}x_{n2} \\ -\sum_n x_{n2}x_{n1} & \sum_n x_{n1}^2 \end{pmatrix} \\ D &= \sum_{m,n}x_{m1}^2x_{n2}^2 - \sum_{m,n}x_{m1}x_{m2}x_{n1}x_{n2} \end{aligned} \tag{2.4.25}$$

となることから、例えば w_1 の推定値の分散は、

$$\begin{aligned} V(\hat{w}_1) &= \frac{\sigma^2}{D}\sum_{n=1}^{N}x_{n2}^2 \\ &= \frac{\sigma^2}{N}\frac{1}{\frac{1}{N}\sum_{n=1}^{N}x_{n1}^2}\frac{1}{1-\rho_{12}^2} \\ \rho_{12}^2 &\equiv \frac{\sum_{m,n}x_{m1}x_{m2}x_{n1}x_{n2}}{\sum_{m,n}x_{m1}^2x_{n2}^2} \end{aligned} \tag{2.4.26}$$

となる。ここで ρ_{12} は説明変数 $\boldsymbol{x}_1, \boldsymbol{x}_2$ 間の標本相関係数である。また、$\sum_{n=1}^{N}x_{n1} = 0$ であることから、$\frac{1}{N}\sum_{n=1}^{N}x_{n1}^2$ は $\boldsymbol{x}_1$ の標本分散である。$V(\hat{\boldsymbol{w}})$ が大きい場合、学習データのサンプリング方法が少し変わったり、学習データの一部に誤差の大きなデータが含まれるだけで推定値が大きく変わる可能性がある。これは係数の推定精度が低いことを意味する。(2.4.26) 式より、係数の推定値の分散を大きくする要因は以下の 4 点である。

- N が小さい（学習データが少ない）
- σ^2 が大きい（観測ノイズの分散が大きい）
- $\frac{1}{N}\sum_{n=1}^{N}x_{n1}^2$ が小さい（説明変数の分散が小さい）
- ρ_{12}^2 が 1 に近い（説明変数間の相関係数が 1 に近い）

これらのうち始めの 3 点が要因で係数推定の精度が悪くなる例を図 2.4.2 に示す。このような場合には、学習データを増やす、実況と相関が強い説明変数を導入する、学習データのサンプリングの妥当性を検討するなどの対応が必要となる。次に 4 点目の要因について考える。今、2 つの説明変数 x_1 と x_2 に強い多重共線性があり、$x_{n1} \approx ax_{n2}$（a は定数）と書けるとすると、$\rho_{12}^2 \approx 1$ となることから、説明変数に強い多重共線性がある場合も係数の推定精度が悪くなるといえる。ガイダンスの開発を行う場合、一般に説明変数間にはある程度の多重共線性がある。このこと自体は大きな問題ではないが、多重共線性が強い場合には係数の推定精度が悪くなることに注意しなければならない。

2.4.6 正則化法

説明変数間に強い多重共線性がある場合でも係数をもっともらしく推定する方法の一つがここで述べる正則化法である。正則化法は線形重回帰に限らず様々な統計手法で利用される。

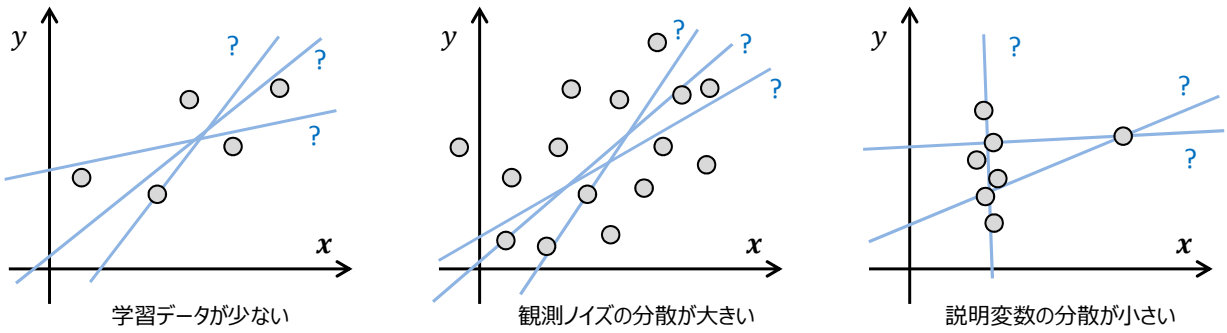

図 2.4.2　係数の推定精度が悪くなる場合の例

ここでは簡単のために、説明変数が 2 つでバイアス項が 0 のモデル (2.4.23) 式を考える。説明変数に強い多重共線性があり、$x_{n1} \approx x_{n2}$ とすると、

$$\begin{aligned} \hat{y}_n &= x_{n1} \\ \hat{y}_n &= 2x_{n1} - x_{n2} \\ \hat{y}_n &= 100x_{n1} - 99x_{n2} \end{aligned} \tag{2.4.27}$$

はいずれもほぼ同じ結果を与える。十分な学習データが与えられれば 1 番目の回帰式が得られることが期待されるが、学習データが不十分な場合には係数の推定精度が低くなり、2 番目や 3 番目のような回帰式が得られる可能性がある。そして 3 番目の式のように係数の絶対値が大きくなればなるほど、2 つの説明変数のバランスがわずかに崩れただけでも予測値が大きく変わってしまい、予測が不安定になる [3]。

線形重回帰では (2.4.6) 式で表される誤差関数を最小にする係数を求めるのだが、誤差関数に $\boldsymbol{w}$ の大きさに応じたペナルティー項を加えることを考える。

$$E = \frac{1}{2}\sum_{n=1}^{N}\left(y_n - \sum_{k=0}^{K} x_{nk}w_k\right)^2 + \frac{\lambda}{\beta}\sum_{k=0}^{K}|w_k|^\beta \tag{2.4.28}$$

ここで λ は正の定数で、β は 1, 2 などがよく用いられる。このように誤差関数にペナルティー項を加えることを正則化といい、$\beta = 1$ の場合を L1 正則化、$\beta = 2$ の場合を L2 正則化という [4]。線形重回帰の場合には、L1 正則化を Lasso (Tibshirani 1996)、L2 正則化をリッジ回帰 (Horel and Kennard 1970) という。いずれの場合も、平均二乗誤差が同程度である場合には係数の大きさの和が小さいほど良いモデルとして採用されるため、(2.4.27) 式の 2 番目や 3 番目のような回帰式は得られにくくなる。

正則化は説明変数の数が多い場合に特に有効である。第 2.3.11 項で述べたように、説明変数が多いとモデルの表現力（複雑な関係性を表現できる能力）が高くなるため、学習データに対して過剰に最適化した係数が得られてしまう。この場合、未知のデータに対する誤差が大きくなり、ガイダンスの精度は低下する。多すぎる説明変数を減らす方法としては、説明変数間の相関係数に基づいて開発者が減らす方法や、次項で述べる手法がよく用いられるが、正則化法を用いることで強い多重共線性を持つ説明変数が選ばれにくくなるため、説明変数の選択を効率的に行うことができるようになる。

2.4.7　説明変数の選択

線形重回帰において、説明変数を選択する方法としては主に以下の 3 つが挙げられる。

- 赤池情報量基準 (AIC) を用いる方法
- 交差検証 (CV) を用いる方法
- 主成分分析を用いる方法

ここでは線形重回帰における AIC と CV を用いた説明変数の選択について述べる。主成分分析については統計手法によらない手法であるため、第 2.3.11 項 (3) を参照していただきたい。

(1)　赤池情報量基準を用いる方法

線形重回帰での AIC を求める。第 2.4.3 項と同様に、ここでも観測ノイズの分布として正規分布 $\epsilon_n \sim N(0, \sigma^2)$ を考える。このとき、確率密度関数と対数尤度は、

$$\begin{aligned} p(\epsilon_n) &= \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{\epsilon_n^2}{2\sigma^2}\right) \\ \ln L &= \sum_{n=1}^{N}\ln\left[\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{\epsilon_n^2}{2\sigma^2}\right)\right] \\ &= -\frac{N}{2}\ln 2\pi - \frac{N}{2}\ln\sigma^2 - \sum_{n=1}^{N}\frac{\epsilon_n^2}{2\sigma^2} \end{aligned} \tag{2.4.29}$$

[3] このため、例えば $\hat{y}_n = 0.5x_{n1} + 0.5x_{n2}$ というような関係が得られた場合には、予測結果の解釈が難しくなるという問題はあるが、予測の安定性に関しては大きな問題は生じない。

[4] L1, L2 のほかにも、0 ではない係数の数に比例したペナルティー項を与える L0 正則化や、絶対値が最大となる係数に比例したペナルティー項を与える L∞ 正則化、L1 と L2 を組み合わせた Elastic net など、様々な正則化法が提案されている。

となる。ここで観測ノイズの分散 σ^2 は通常は未知のパラメータであるため、学習データから σ^2 の推定値 $\hat{\sigma}^2$ を最尤法で求めることにする。すなわち、上記の対数尤度を σ^2 の関数と見なし、$\ln L\left(\sigma^2\right)$ を σ^2 で微分して 0 になるときの σ^2 を分散の推定値 $\hat{\sigma}^2$ とする。

$$\left.\frac{\partial \ln L}{\partial \sigma^2}\right|_{\sigma^2=\hat{\sigma}^2} = -\frac{N}{2\hat{\sigma}^2} + \sum_{n=1}^{N} \frac{\epsilon_n^2}{2(\hat{\sigma}^2)^2} = 0 \quad (2.4.30)$$

よって、

$$\hat{\sigma}^2 = \frac{1}{N}\sum_{n=1}^{N} \epsilon_n^2 = \frac{\mathrm{RSS}}{N} \quad (2.4.31)$$

となる。ここで RSS は残差の二乗和 (Residual Sum of Square) である。この $\hat{\sigma}^2$ を用いて対数尤度を書くと、

$$\ln L\left(\hat{\sigma}^2\right) = -\frac{N}{2}\ln 2\pi - \frac{N}{2}\ln\frac{\mathrm{RSS}}{N} - \frac{N}{2} \quad (2.4.32)$$

であるので、AIC は、

$$\mathrm{AIC} = N\ln 2\pi + N\ln\frac{\mathrm{RSS}}{N} + N + 2K \quad (2.4.33)$$

となる。AIC を用いて説明変数を選択する場合、予測式を変える前と後での AIC の差に着目することになるため、定数部分は無視してよい。学習データのサンプル数 N はここでは定数であるから、右辺の第 1 項と第 3 項は AIC の計算には不要となる。よって線形重回帰の場合の AIC は、

$$\mathrm{AIC} = N\ln\frac{\mathrm{RSS}}{N} + 2K \quad (2.4.34)$$

と書くことができる。学習データに対して、AIC が最も小さくなる説明変数の組み合わせを総当たり法や変数増減法などで選択する。

(2) 交差検証を用いる方法

CV は繰り返し計算を行うため計算に時間が掛かるという問題があるが、線形重回帰に用いる場合には計算を効率的に行う方法がある。LOOCV で平均二乗誤差 MSE を求めると、

$$\mathrm{MSE}_{\mathrm{LOOCV}} = \frac{1}{N}\sum_{n=1}^{N}\left(y_n - \hat{f}^{-n}(\boldsymbol{x}_n)\right)^2 \quad (2.4.35)$$

と書ける（(2.3.60) 式を再掲）。ここで、$\hat{f}^{-n}$ は n 番目の学習データを除いて作成した回帰式である。煩雑なため証明は省略するが、(2.4.35) 式を変形すると、

$$\mathrm{MSE}_{\mathrm{LOOCV}} = \frac{1}{N}\sum_{n=1}^{N}\frac{\left(y_n - \hat{f}(\boldsymbol{x}_n)\right)^2}{\left(1-[H]_{nn}\right)^2} \quad (2.4.36)$$

と近似ではなく厳密に書ける。ここで $[H]_{nn}$ は (2.4.11) 式で示したハット行列の n 行 n 列成分、$\hat{f}$ は N 個の学習データを全て用いて作成した回帰式である。(2.4.36) 式では、$\hat{f}^{-n}$ ではなく $\hat{f}$ となっている所がポイントで、(2.4.35) 式ではデータを一つ抜いて回帰式を作成するということを N 回繰り返す必要があったのに対し、(2.4.36) 式では N 個のデータを全て用いた通常の回帰式の作成を 1 回だけ実行すれば良いため高速に計算できる。この方法は線形重回帰の場合に限られるが、学習データの数が多い場合には特に有効である。この計算を総当たり法や変数減少法等を用いて説明変数の組み合わせを変えながら実行し、$\mathrm{MSE}_{\mathrm{LOOCV}}$ が最も小さくなる説明変数の組み合わせを選ぶ。

2.4.8 逐次学習

線形重回帰では (2.4.10) 式を用いた一括学習により係数を決定する場合が多いが、第 2.3.10 項で述べた確率的勾配降下法を用いることで逐次学習することも可能である。線形重回帰の誤差関数 E は残差の二乗和であり、これを係数 $\boldsymbol{w}$ の各成分で微分すると (2.4.7) 式になることから、時刻 t における係数 $\boldsymbol{w}^{(t)}$ を逐次更新する式は (2.3.56) 式より、

$$w_k^{(t+1)} = w_k^{(t)} + \eta\, x_{tk}\left(y_t - \sum_{l=0}^{K} x_{tl}\, w_l^{(t)}\right) \quad (2.4.37)$$

となる。ここで y_t と $\boldsymbol{x}_t$ は時刻 t における観測値と説明変数である。第 2.3.10 項でも述べたように、逐次学習する場合には、学習率 η の値を適切に調整し、説明変数を規格化して説明変数のオーダーを揃えておく必要がある。

2.4.9 線形重回帰の利用上の注意点

線形重回帰は最もシンプルな統計手法であり広く利用されているが、利用に当たっては注意すべき点が多い。ここでは例を示しながら線形重回帰を利用する上での注意点を述べる。

(1) 線形重回帰の仮定に反していないか

第 2.4.2 項で示したように、線形重回帰では以下の 6 つの仮定の下で回帰式が決定される。

仮定 1. モデルの線形性
仮定 2. 説明変数は非確率変数
仮定 3. 観測ノイズの不偏性
仮定 4. 観測ノイズの等分散性
仮定 5. 観測ノイズの独立性
仮定 6. 説明変数に完全な多重共線性はない

ガイダンスでは、数値予報モデルの予測値などの確定した値を用いて説明変数を作成するため、通常は仮定 2 は満たされていると考えて問題ないだろう。また、説明変数間にはある程度の相関は必ずあるものの、例えばある説明変数を 2 倍にした変数を説明変数に加える等、開発者が意図的に操作しない限りは説明変数が完全な多重共線性を持つことはないので、通常は仮定 6 も満たされていると考えて問題はない。残りの 4 つの

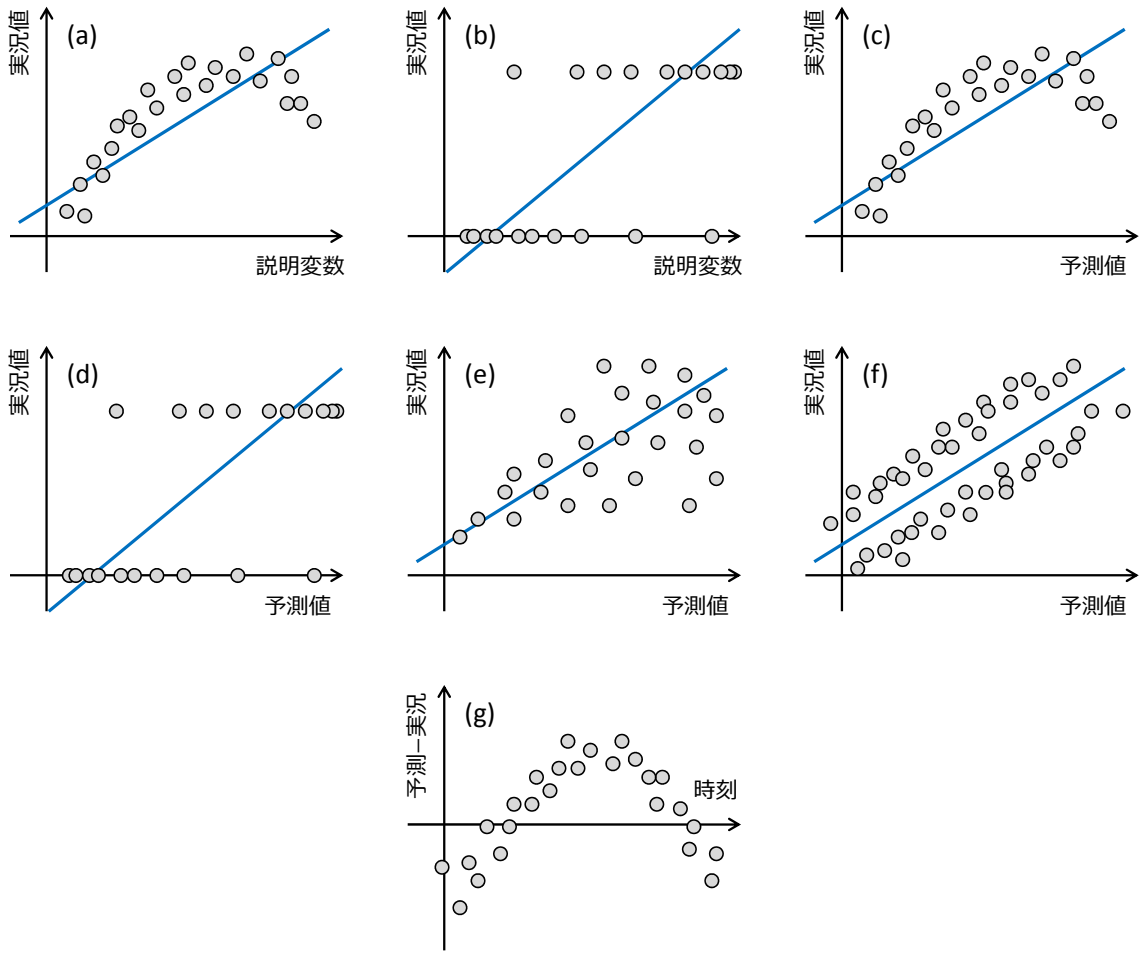

図 2.4.3　線形重回帰で用いられている仮定が成り立たない例

仮定に関しては、用いているデータが仮定を満たしているか否かは自明ではないため、開発を行うに当たってデータの性質を確かめておく必要がある。

データの性質を簡単に確かめる方法としてはデータを実際にプロットしてみると良い。まず、仮定 1 のモデルの線形性については、実際にデータをプロットすることで、目的変数（実況値）と説明変数に線形関係があるか否かを確かめる。このとき、個別の説明変数が実況値と線形関係を持たなければならないことに注意が必要である。各説明変数と実況値をプロットしたとき、図 2.4.3(a) のような関係にある場合には線形関係を満たさないため、このままでは線形重回帰の説明変数に用いることはできない。このような場合には、説明変数を何らかの関数で変換することで線形化できる場合がある。図 2.4.3(b) のような場合には、ロジスティック回帰等のほかの統計手法を用いることを検討する。また、例えばある変数 u を説明変数として用いる場合、u の 2 乗や対数など u の何らかの関数 $f(u)$ を説明変数に用いてはならない。なぜならば、もし u が実況値と線形関係にあるならば、u の線形変換 $f(u) = au + b$（a, b は定数）を除いては $f(u)$ は実況と非線形関係となるため仮定 1 に反することになり、$f(u) = au + b$ とした場合には u と完全な多重共線性を持つため仮定 6 に反するからである。逆に、もし $f(u) \neq u$ が実況値と線形関係にあるならば、u 自身は実況値と非線形関係を持つか $f(u)$ と完全な多重共線性を持つため、u を説明変数に用いることはできない。よって線形重回帰では、もし $f(u)$ を説明変数に用いた場合、u の別の関数 $g(u)$ を説明変数に用いることはできない。

仮定 3 の観測ノイズの不偏性の仮定が満たされているか否かは、ガイダンスの予測値と実況値の関係をプロットした時に、予測値に依らず予測と実況の差が一定と見なせるかどうかを確かめることになる。図 2.4.3(c) の場合は、全体としてはバイアスが 0 だったとしても、予測値の大きさによってバイアスが変化しているため、観測ノイズの不偏性は満たされていない。このような場合には、予測と実況が線形になるように説明変数の見直しを行うか、季節や予測値などで層別化するか、別の統計手法を用いるなどの検討が必要になる。図 2.4.3(d) は 2 値データを線形モデルで予測しようとした例であ

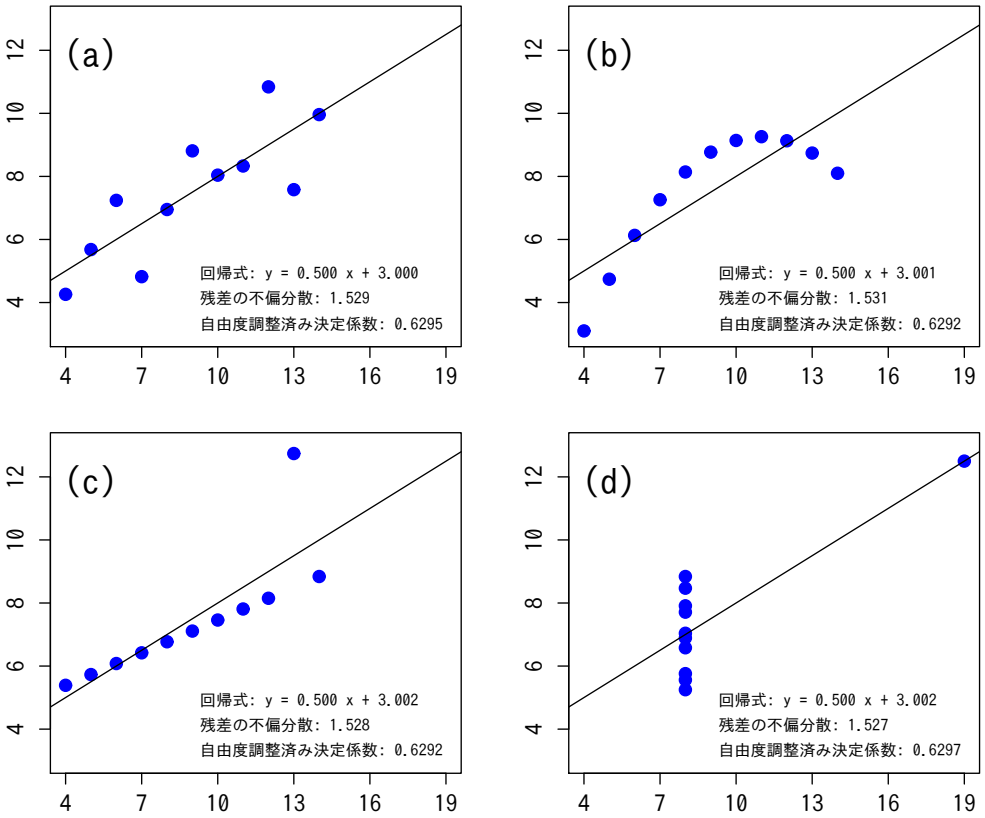

図 2.4.4　Anscombe (1973) の例

り、観測ノイズの不偏性は満たされていない。このような場合には、線形重回帰ではなくロジスティック回帰を用いるなどを検討する必要がある。

仮定 4 の観測ノイズの等分散性は、各データについて予測と実況の差の分散が一定であることを意味している。図 2.4.3(c)~(e) はいずれもこの等分散性の仮定を満たしていないため、説明変数や統計手法の見直しが必要となる。

仮定 5 の観測ノイズの独立性が成り立たない場合、すなわち観測ノイズに相関があるのは、空間方向や時間方向に相関がある場合である。例えば、バイアス特性が異なる 2 つの地点の気温をまとめて予測しようとした場合には図 2.4.3(f) のようになるため、地点で層別化するか、地点特性を考慮した説明変数を導入するなどの検討が必要である。また、時間方向に誤差の相関がある場合には図 2.4.3(g) のようになるため、時刻や季節などで層別化したり、時刻や季節の特性を考慮した説明変数を導入するなどの検討が必要になる。

(2)　回帰式の結果だけを見ていないか

回帰式を決定した後に、決定係数の値や回帰直線、学習データや予測データに対する誤差といった回帰の結果だけに注目するのではなく、予測値と実況値の関係をプロットしてみることが重要である。

図 2.4.4 は Anscombe (1973) の例で、図中の点は回帰式の作成に用いられたデータ、直線は回帰直線を表している。(a)~(d) のデータのうち、(a) 以外のデータは線形重回帰を適用できないデータであることは図を見ればすぐに分かるが、図中に示した回帰式、残差の不偏分散、自由度調整済み決定係数は (a)~(d) のいずれもほぼ同じ値になっており、これらの値を見ただけでは統計関係の正しさを知ることはできない。データと回帰直線の関係をプロットするほかにも、残差を予測値に対してプロットしたり、正規 Q-Q プロット（Wilks 1995 など）や Cook の距離 (Cook and Weisberg 1982) を利用したりすることで、線形重回帰に用いるデータの正しさを視覚的に確かめることができる。

(3)　学習時、検証時の注意点

線形重回帰に限らず、多くの統計手法では学習に用いるデータは独立であることを仮定しているが、一定期間の連続した気象データを学習に用いる場合、学習データ間にはある程度の相関が含まれていることを考慮しなければならない。例えば時刻方向に回帰式を層別化していない場合には、同じ観測データに対して複数の時刻の予測値が得られるため、学習に使用する初期時刻や予報時間を限定するなどの工夫が必要となる。

検証では係数の学習に使用していないデータ（学習データとは独立なデータ）を用いなければならない。これは当然のことに思われるが、係数を決定すること

に加えて、説明変数の算出に利用するパラメータを調整するような場合には注意が必要である。例えば 3 年分のデータを用意して、初めの 2 年分のデータで係数を学習し、残りの 1 年分のデータで検証することを考える。これ自体はほぼ独立なデータを用いた検証といえるのだが、検証した結果、期待していたような精度が得られなかった場合、説明変数のパラメータを調整して再び係数を学習して検証する、ということを繰り返すことがある。この場合、回帰式と説明変数は用意した 3 年分のデータにフィットするように調整されてしまうため、結果的に 3 年分のデータ全てを学習データとして扱ったことになってしまう。このような場合にはさらに別途検証用のデータを用意する必要がある。検証用のデータを別途用意できない場合には CV を用いて精度を評価することになる。ただし第 2.3.11 項 (2) でも述べたとおり、時間方向に相関を持つデータに CV を用いた場合には、実際の検証結果よりも誤差が小さく見積もられることに注意が必要である。

2.4.10 まとめ

本節では、線形重回帰の手法とその周辺の技術及び利用上の注意点を述べた。線形重回帰は最もシンプルな手法であり様々に利用されているが、係数を決定する上で置かれている仮定が多く、これらが満たされていないデータに対しては、説明変数を見直したり、別の統計手法を利用するなどの検討が必要である。また、手法を正しく用いるためには、回帰の結果や検証結果だけに注目するのではなく、説明変数と実況値、予測値と実況値の関係などを実際にプロットしてみることが重要である。これは線形重回帰に限らず、全ての統計手法に共通していえることである。

付録 2.4.A 決定係数が (2.4.17) 式と書けることの証明

ここでは、

$$\frac{\sum_{n=1}^{N}\left(\hat{y}_n-\bar{y}\right)^2}{\sum_{n=1}^{N}\left(y_n-\bar{y}\right)^2}=1-\frac{\sum_{n=1}^{N}\left(y_n-\hat{y}_n\right)^2}{\sum_{n=1}^{N}\left(y_n-\bar{y}\right)^2}$$

と書けることを示す。左辺の分母は、

$$\begin{aligned}\sum_{n=1}^{N}\left(y_n-\bar{y}\right)^2&=\sum_{n=1}^{N}\left(y_n-\hat{y}_n+\hat{y}_n-\bar{y}\right)^2\\&=\sum_{n=1}^{N}\left(y_n-\hat{y}_n\right)^2+\sum_{n=1}^{N}\left(\hat{y}_n-\bar{y}\right)^2\\&+2\sum_{n=1}^{N}\left(y_n\hat{y}_n-\hat{y}_n\hat{y}_n\right)+2\,\bar{y}\sum_{n=1}^{N}\left(\hat{y}_n-y_n\right)\end{aligned}\tag{2.4.38}$$

と書ける。ここで (2.4.38) 式の右辺第 3 項を行列形式で書くと $2\left(\boldsymbol{y}^T\hat{\boldsymbol{y}}-\hat{\boldsymbol{y}}^T\hat{\boldsymbol{y}}\right)$ であり、また、(2.4.11) 式より、$\hat{\boldsymbol{y}}=H\boldsymbol{y}$ であることから、

$$\begin{aligned}\hat{\boldsymbol{y}}^T\hat{\boldsymbol{y}}&=(H\boldsymbol{y})^TH\boldsymbol{y}\\&=\boldsymbol{y}^THH\boldsymbol{y}\\&=\boldsymbol{y}^TX\left(X^TX\right)^{-1}X^TX\left(X^TX\right)^{-1}X^T\boldsymbol{y}\\&=\boldsymbol{y}^TH\boldsymbol{y}\\&=\boldsymbol{y}^T\hat{\boldsymbol{y}}\end{aligned}\tag{2.4.39}$$

と書ける。ここで、ハット行列は対称行列であること (2.4.12) 式を用いた。これより (2.4.38) 式の右辺第 3 項は 0 になる。また (2.4.5) 式より $\hat{y}_n=E(y_n)$ であるから、右辺第 4 項も 0 となる。以上より、

$$\sum_{n=1}^{N}\left(\hat{y}_n-\bar{y}\right)^2=\sum_{n=1}^{N}\left(y_n-\bar{y}\right)^2-\sum_{n=1}^{N}\left(y_n-\hat{y}_n\right)^2\tag{2.4.40}$$

であるから、これを (2.4.16) 式に代入すると (2.4.17) 式が得られる。

参考文献

Anscombe, F. J., 1973: Graphs in statistical analysis. *The American Statistician*, **27(1)**, 17–21.

Cook, R. D. and S. Weisberg, 1982: *Residuals and influence in regression*. New York: Chapman and Hall, 229 pp.

廣津千尋, 1992: 線形モデルと最小二乗法. 自然科学の統計学 第 2 章, 東京大学出版会, 26–78.

Horel, A. E. and R. W. Kennard, 1970: Ridge regression: biased estimation for nonorthogonal problems. *Technometrics*, **12**, 55–67.

Tibshirani, R., 1996: Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B*, **58(1)**, 267–288.

Wilks, D. S., 1995: *Statistical methods in the atmospheric sciences*. Academic press, 467 pp.

2.5 ロジスティック回帰 [1]

2.5.1 はじめに

前節で述べた線形重回帰は、気温のような連続値の予測に用いられる統計手法の一つである。これに対して本節で述べるロジスティック回帰は、例えば雷の有無などのように、現象を 2 つのクラスに分類する問題に用いられる統計手法の一つである。後述するように、ロジスティック回帰により得られる予測値は現象の発生確率と考えることができるため、ロジスティック回帰は確率型のガイダンスによく用いられている。気象庁でのロジスティック回帰の利用は発雷確率ガイダンスに始まり (高田 2007)、その後、乱気流指数や雲底確率ガイダンス、ガスト発生確率ガイダンスなどにも利用されるようになった。

本節では初めにロジスティック回帰の基礎的な理論について述べ、続いてロジスティック回帰を多クラス分類に適用した順序ロジスティック回帰と多項ロジスティック回帰について述べる。最後に利用上の注意点等を述べる。

2.5.2 ロジスティック回帰の基礎

ロジスティック回帰 (Cox 1958) は、目的変数が 2 値データでベルヌーイ分布に従う場合に用いられる回帰分析の一種 [2] である。2 値データとは、雷があった場合を 1、なかった場合を 0 としたような 2 つの値だけを取るデータのことである。ここでは初めに、確率がベルヌーイ分布で表される一般的な場合について考える。

ベルヌーイ分布とは、確率変数 Y が確率 p で現象あり ($Y=1$)、確率 $1-p$ で現象なし ($Y=0$) を取る離散確率分布である。今、説明変数 $\boldsymbol{x}$ と係数 $\boldsymbol{w}$ が与えられているとする。このとき、現象ありとなる確率を $P_r(Y=1|\boldsymbol{x},\boldsymbol{w})$ とし、

$$p \equiv P_r(Y=1|\boldsymbol{x},\boldsymbol{w}) \tag{2.5.1}$$

とすると、現象なしとなる確率は、

$$P_r(Y=0|\boldsymbol{x},\boldsymbol{w}) = 1-p \tag{2.5.2}$$

となる。ここで p は $\boldsymbol{x}$ と $\boldsymbol{w}$ の関数である。この 2 つをまとめると、ベルヌーイ分布の確率関数は以下のように書ける。

$$P_r(Y=y|\boldsymbol{x},\boldsymbol{w}) = p^y\,(1-p)^{1-y} \tag{2.5.3}$$

y を 2 値データの観測値とし、N 組の観測値と説明変数のデータセット $(y_1,\boldsymbol{x}_1),\ \cdots,\ (y_N,\boldsymbol{x}_N)$ が与えられたとき、最適な係数 $\boldsymbol{w}$ を最尤法で求めることを考える。各観測が独立であると仮定すれば、尤度 L は係数

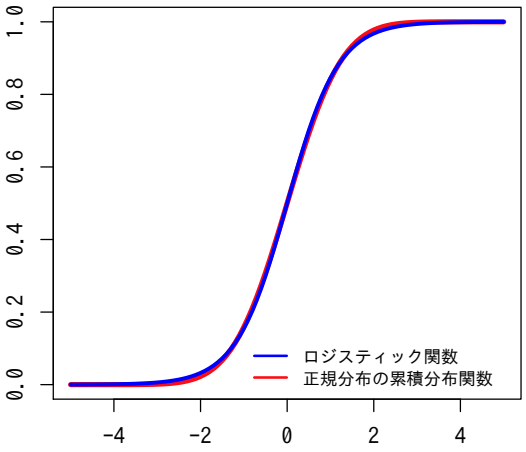

図 2.5.1 ロジスティック関数 $y=1/(1+\exp(-1.7x))$ と正規分布 $N(0,1)$ の累積分布関数

ベクトル $\boldsymbol{w}$ の関数として、

$$\begin{aligned} L(\boldsymbol{w}) &= \prod_{n=1}^{N} P_r(Y_n=y_n|\boldsymbol{x}_n,\boldsymbol{w}) \\ &= \prod_{n=1}^{N} {p_n}^{y_n}\,(1-p_n)^{1-y_n} \end{aligned} \tag{2.5.4}$$

と書ける。よって対数尤度は次のように書ける。

$$\ln L(\boldsymbol{w}) = \sum_{n=1}^{N} \left[y_n \ln p_n + (1-y_n)\ln(1-p_n)\right] \tag{2.5.5}$$

対数尤度が最大となる係数を求めたいので、対数尤度を $\boldsymbol{w}$ の各成分で微分する。

$$\begin{aligned} \frac{\partial \ln L}{\partial w_k} &= \sum_{n=1}^{N} \left[\frac{y_n}{p_n}\frac{\partial p_n}{\partial w_k} - \frac{1-y_n}{1-p_n}\frac{\partial p_n}{\partial w_k}\right] \\ &= \sum_{n=1}^{N} \frac{\partial p_n}{\partial w_k}\frac{y_n-p_n}{p_n(1-p_n)} \end{aligned} \tag{2.5.6}$$

ここまではベルヌーイ分布に従う統計モデルに対して一般的に成り立つ議論であり、確率 p_n として具体的にどのような関数を用いるかによって統計手法が異なる。ロジスティック回帰では、p_n が次のロジスティック関数で表されるものと仮定する。

$$p_n = \frac{1}{1+\exp\left(-\sum_{k=0}^{K} x_{nk}w_k\right)} \tag{2.5.7}$$

ここで w_0 はバイアス項で、$x_{n0}=1$ とする。ロジスティック関数は図 2.5.1 の青線のような関数である。(2.5.7) 式を用いると、

$$\frac{\partial p_n}{\partial w_k} = x_{nk}p_n\,(1-p_n) \tag{2.5.8}$$

[1] 工藤 淳

[2] ほかにも、プロビット回帰や complementary log-log 回帰などがある。

となることから、(2.5.6) 式は

$$\frac{\partial \ln L}{\partial w_k} = \sum_{n=1}^{N} x_{nk} \left(y_n - p_n \right) \tag{2.5.9}$$

というシンプルな形で書くことができる。これが 0 になるときの $\boldsymbol{w}$ をニュートン・ラフソン法などを用いて数値的に求めれば、ロジスティック回帰における係数が求まる。求められた係数と予測データを (2.5.7) 式に適用することで、ロジスティック回帰による予測値が得られる。ロジスティック回帰を 2 クラス分類問題に適用する場合には、p がある値（例えば 0.5）以上であれば現象あり、それよりも小さければ現象なし、というように分類すれば良い。

ロジスティック関数は 0 ～ 1 の値を取る単調増加関数で、全域で微分可能であるという特性を持っていることから、確率を表現するのに適した関数であるといえる。このため、(2.5.7) 式では確率がロジスティック関数で書けると仮定したのだが、上記のような特性を持つ関数はロジスティック関数だけではない。例えば確率という意味でいえば、正規分布の累積分布関数（図 2.5.1 の赤線）を用いても良さそうである。確率をこのように仮定した回帰をプロビット回帰といい、その確率密度関数は

$$p_n = \int_{-\infty}^{\sum_{k=0}^{K} x_{nk} w_k} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z^2}{2} \right) dz \tag{2.5.10}$$

である。しかし、(2.5.6) 式を求めるためには (2.5.10) 式を w_k で微分する必要があるため扱いが困難である。これに対してロジスティック関数は、(2.5.8) 式のように微分が容易にでき、対数尤度の微分もシンプルな形で記述できる。またパラメータを適当に調整すれば、図 2.5.1 のように正規分布の累積分布関数とほとんど同じ曲線になる。このような理由から、ロジスティック関数は 2 値データの回帰にしばしば用いられている。

2.5.3 正則化法、説明変数の選択、逐次学習

ロジスティック回帰の場合も線形重回帰と同様に、正則化、説明変数の選択、逐次学習といった手法を用いることができる。これらについて以下で述べる。

(1) 正則化法

ロジスティック回帰における誤差関数 E は負の対数尤度 $-\ln L$ である。これに係数の大きさに依存したペナルティー項を加えることで多重共線性の影響を減らすことができる。

$$E = -\ln L + \frac{\lambda}{\beta} \sum_{k=0}^{K} |w_k|^{\beta} \tag{2.5.11}$$

λ は正の正則化定数で、線形重回帰と同様に、β が 1 の場合は L1 正則化、2 の場合は L2 正則化となる。

(2) 説明変数の選択

ロジスティック回帰の場合も AIC や CV、主成分分析により説明変数を選択することができる。AIC を用いる場合には対数尤度の式 (2.5.5) をそのまま用いれば良い。

$$\begin{aligned} \mathrm{AIC} &= -2 \ln L + 2K \\ &= -2 \sum_{n=1}^{N} \left[y_n \ln p_n + (1 - y_n) \ln(1 - p_n) \right] + 2K \end{aligned} \tag{2.5.12}$$

この式を用いて、AIC が最小になる説明変数の組み合わせを選択する。

CV を用いる場合は、例えばブライアスコアを基準にすることになる。LOOCV を利用するならば、

$$\mathrm{BS}_{\mathrm{LOOCV}} = \frac{1}{N} \sum_{n=1}^{N} \left(y_n - \hat{p}^{-n}(\boldsymbol{x}_n) \right)^2 \tag{2.5.13}$$

となる。ここで $\hat{p}^{-n}(\boldsymbol{x}_n)$ は、n 番目のデータを除いて作成したロジスティック回帰に $\boldsymbol{x}_n$ を適用した場合の予測値である。

主成分分析は説明変数間の分散共分散または相関係数のみで決まり、目的変数には依らないため、統計手法に関わらず第 2.3.11 項 (3) で述べた手法で説明変数を選択できる。

(3) 逐次学習

ロジスティック回帰における誤差関数 E は負の対数尤度 $-\ln L$ であるため、誤差関数を $\boldsymbol{w}$ で微分した値は (2.5.9) 式に負号を付けたものになる。

$$\frac{\partial E}{\partial w_k} = -\sum_{n=1}^{N} x_{nk} \left(y_n - p_n \right) \tag{2.5.14}$$

これを (2.3.56) 式に代入すれば、ロジスティック回帰における逐次学習の式が得られる。

$$w_k^{(t+1)} = w_k^{(t)} + \eta \, x_{tk} \left(y_t - p(\boldsymbol{x}_t, \boldsymbol{w}^{(t)}) \right) \tag{2.5.15}$$

第 2.3.10 項で述べたように、係数を逐次学習する場合には、学習率 η を適切に調整するとともに、説明変数を規格化して説明変数のオーダーを揃えておく必要がある。

2.5.4 順序ロジスティック回帰

順序ロジスティック回帰 (McCullagh 1980) は、目的変数 y が 0, 1, 2, $\cdots$ のように順序のある階級に分かれている場合のロジスティック回帰である。例えば降水強度を弱、並、強に階級分けするような場合に用いることができる。

ここでは階級 m が 0, 1, $\cdots$, $M-1$ の M 個のクラスに分かれているとする。順序ロジスティック回帰

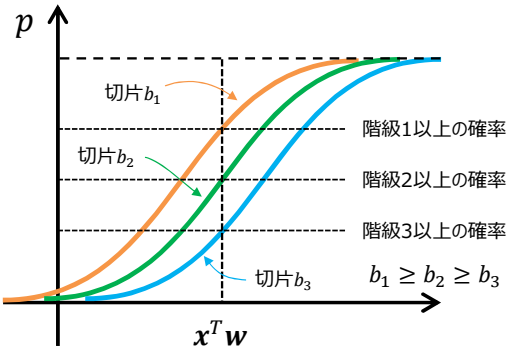

図 2.5.2　順序ロジスティック回帰の概念図

では、n 番目の実況に対応する確率変数 Y_n が m 番目以上の階級である確率は

$$P_r(Y_n \geq m|\boldsymbol{x}_n, \boldsymbol{w}) = \frac{1}{1+\exp\left(-b_m - \sum_{k=1}^{K} x_{nk} w_k\right)} \tag{2.5.16}$$

であると仮定する。ここで b_m は m 番目以上の階級である確率に対するロジスティック関数の切片である。(2.5.7) 式では切片を和に含めているが、上式では和の外に出して記述している。(2.5.16) 式では、階級の違いは切片のみであり、説明変数に掛かる係数は全ての階級で等しいと仮定している。例えば階級が 0 ～ 3 の 4 階級で分けられる場合を図で表すと図 2.5.2 のようになる。1 番目の曲線（左の曲線）は階級が 1 以上か 1 未満か（すなわち階級 0 か）を分ける曲線で、切片は b_1 である。2 番目の曲線（中央の曲線）、3 番目の曲線（右の曲線）も同様であり、$b_1 \geq b_2 \geq b_3$ である。説明変数 $\boldsymbol{x}$ と係数 $\boldsymbol{w}$ が与えられたとき、階級が 1, 2, 3 以上である確率は図に示したようになる。4 つの階級は 3 本のロジスティック関数で分けられ、各曲線の傾きは全て等しく、切片の大きさによって線が横軸方向に移動している。傾きを等しくすることで、確率の逆転が生じることがなくなる。

それぞれの階級になる確率は (2.5.16) 式の差で表すことができる。

$$\begin{aligned} &P_r(Y_n = m|\boldsymbol{x}_n, \boldsymbol{b}, \boldsymbol{w}) \\ &= P_r(Y_n \geq m|\boldsymbol{x}_n, \boldsymbol{b}, \boldsymbol{w}) - P_r(Y_n \geq m+1|\boldsymbol{x}_n, \boldsymbol{b}, \boldsymbol{w}) \\ &= \frac{1}{1+\exp\left(-b_m - \sum_{k=1}^{K} x_{nk} w_k\right)} \\ &\quad - \frac{1}{1+\exp\left(-b_{m+1} - \sum_{k=1}^{K} x_{nk} w_k\right)} \end{aligned} \tag{2.5.17}$$

ここで、$b_0 = +\infty, b_M = -\infty$ とする。N 組の学習データ $(\boldsymbol{y}_1, \boldsymbol{x}_1)$, $\cdots$, $(\boldsymbol{y}_N, \boldsymbol{x}_N)$ が与えられたとき、各観測が独立であると仮定すれば、対数尤度は切片 $\boldsymbol{b}$ と係数 $\boldsymbol{w}$ の関数として、

$$\ln L(\boldsymbol{b}, \boldsymbol{w}) = \sum_{n=1}^{N} \ln P_r(Y_n = m|\boldsymbol{x}_n, \boldsymbol{b}, \boldsymbol{w}) \tag{2.5.18}$$

となる。(2.5.18) 式を $\boldsymbol{b}$ と $\boldsymbol{w}$ でそれぞれ微分して 0 になる解を数値的に求めることで、順序ロジスティック回帰の係数が求まる。このようにして得られた係数と説明変数を (2.5.17) 式に代入することで、各階級になる確率の予測値を求めることができる。

2.5.5　多項ロジスティック回帰

ロジスティック回帰を多クラス分類に拡張したものを多項ロジスティック回帰[3](Engel 1988) という。多項ロジスティック回帰は、例えば天気を晴れ、曇り、雨、雪に分類するような問題に用いることができる。

目的変数 $\boldsymbol{y}$ は M クラスのカテゴリーに対応した M 次元ベクトルで、実況は m 番目のクラスが起き ($y_m = 1$)、残りは起きない ($y_m = 0$) ものとする。このような、ベクトルの 1 つの成分だけが 1 で残りは全て 0 であるベクトルを one-hot ベクトルという。$\boldsymbol{y}$ は one-hot ベクトルであるから、$\sum_{m=1}^{M} y_m = 1$ となる。説明変数 $\boldsymbol{x}$ と係数 W が与えられたとき、m 番目のクラスの確率変数 Y_m が 1 を取る確率を

$$p_m \equiv P_r\left(Y_m = 1|\boldsymbol{x}, W\right) \tag{2.5.19}$$

と書くことにする。説明変数が K 個ある場合、$\boldsymbol{x}$ は切片も含めた $K+1$ 次元ベクトルで、W は $K+1$ 行 M 列の行列である。1 番目のクラスの実況が y_1、2 番目のクラスの実況が y_2、$\cdots$ である確率は、

$$P_r\left(Y_1 = y_1, \cdots, Y_M = y_M|\boldsymbol{x}, W\right) = \prod_{m=1}^{M} {p_m}^{y_m} \tag{2.5.20}$$

となる。ここで $\sum_{m=1}^{M} p_m = 1$、$\sum_{m=1}^{M} y_m = 1$ である。2 クラス分類の場合、つまり $M = 2$ の場合は、$p_1 + p_2 = 1$、$y_1 + y_2 = 1$ であり、$p_1 \equiv p$、$y_1 \equiv y$ とすると、$p_2 = 1 - p$、$y_2 = 1 - y$ であることから、

$$\begin{aligned} &P_r\left(Y_1 = y_1, Y_2 = y_2|\boldsymbol{x}, W\right) \\ &\quad = {p_1}^{y_1} {p_2}^{y_2} = p^y (1-p)^{1-y} \end{aligned} \tag{2.5.21}$$

となり、ベルヌーイ分布の確率関数 (2.5.3) 式と一致することが確かめられる。

N 組の学習データ $(\boldsymbol{y}_1, \boldsymbol{x}_1)$, $\cdots$, $(\boldsymbol{y}_N, \boldsymbol{x}_N)$ が与えられたとする。各観測が独立であると仮定すれば、尤度 L は係数行列 W の関数として、

$$L(W) = \prod_{n=1}^{N} \prod_{m=1}^{M} {p_{nm}}^{y_{nm}} \tag{2.5.22}$$

[3] ソフトマックス回帰、多クラス回帰、多分岐ロジスティック回帰などと呼ばれることもある。

と書けるので、対数尤度は

$$\ln L(W) = \sum_{n=1}^{N} \sum_{m=1}^{M} y_{nm} \ln p_{nm} \tag{2.5.23}$$

となる。ここで、確率 p_{nm} が次のソフトマックス関数（第 2.6.5 項 (2) も参照）で表されるものと仮定する。

$$p_{nm} = \frac{e^{\phi_{nm}}}{\sum_{j=1}^{M} e^{\phi_{nj}}} \tag{2.5.24}$$

$$\phi_{nm} = \sum_{k=0}^{K} x_{nk} w_{km} \tag{2.5.25}$$

これを用いると、対数尤度は

$$\ln L(W) = \sum_{n=1}^{N} \sum_{m=1}^{M} y_{nm} \left(\phi_{nm} - \ln \sum_{j=1}^{M} e^{\phi_{nj}} \right) \tag{2.5.26}$$

となる。この対数尤度を係数行列 W の各成分で微分すると、$\sum_{m=1}^{M} y_m = 1$ であることを用いて、

$$\begin{aligned} \frac{\partial \ln L}{\partial w_{km}} &= \sum_{n=1}^{N} \sum_{l=1}^{M} y_{nl} \left[x_{nk} \delta_{lm} - x_{nk} \frac{e^{\phi_{nm}}}{\sum_{j=1}^{M} e^{\phi_{nj}}} \right] \\ &= \sum_{n=1}^{N} x_{nk} \left(y_{nm} - p_{nm} \right) \end{aligned} \tag{2.5.27}$$

となり、(2.5.9) 式と同様なシンプルな形で書ける。ここで δ_{lm} はクロネッカーのデルタである。(2.5.27) 式が 0 になる W を数値的に求めることで、多項ロジスティック回帰における係数が得られる。求められた係数を (2.5.24) 式に適用することで、m 番目のクラスに分類される確率が得られる。複数のクラスの中から 1 つのクラスを予測する場合には、確率が最も高いクラスを選択すれば良い。

2.5.6 利用上の注意点

ロジスティック回帰では以下の仮定を用いて回帰係数を導出しているため、用いるデータはこれらの仮定と矛盾してはいけない。

- 観測値がベルヌーイ分布に従うこと
- 確率がロジスティック関数で表されること
- 観測が独立であること

以下ではそれぞれについて注意点を述べ、最後に上記以外の一般的な注意点を述べる。

(1) 観測値がベルヌーイ分布に従う

ロジスティック回帰に用いる観測値は現象があり・なしの 2 値データ（0 か 1）でなければならない。もし観測値として何らかの確率的なもの（割合等）が得られるとしても、その割合を用いて (2.5.9) 式の解を求めてはならない。ベルヌーイ分布は確率変数 Y が確率 p で現象あり ($Y = 1$)、確率 $1 - p$ で現象なし ($Y = 0$) を取る離散確率分布であるから、期待値と分散はそれぞれ次のように書ける。

$$E(Y) = 1 \times p + 0 \times (1 - p) = p \tag{2.5.28}$$

$$V(Y) = E(Y^2) - E(Y)^2 = p\,(1 - p) \tag{2.5.29}$$

ここで p はロジスティック回帰により求められた確率である。横軸に p、縦軸に $V(Y)$ を取ると図 2.5.3 のようになる。ベルヌーイ分布の分散は、確率が 0 または 1 付近で 0 に近く（ばらつきが小さく）、0.5 で最大値 0.25 をとる。このため、例えば線形重回帰で仮定したような、等分散性を持つデータはロジスティック回帰には適用することはできない。

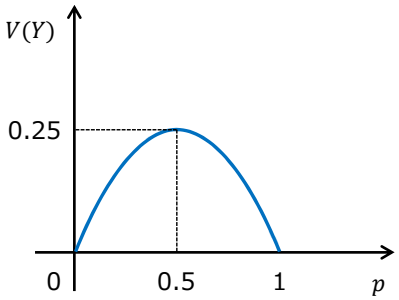

図 2.5.3 ベルヌーイ分布の分散

(2) 確率がロジスティック関数で表される

ロジスティック回帰では、現象ありの確率が (2.5.7) 式のようなロジスティック関数で表されると仮定している。(2.5.7) 式を変形（ロジット変換）すると、

$$\mathrm{logit}(p) \equiv \ln \frac{p}{1 - p} = \sum_{k=0}^{K} x_k w_k \tag{2.5.30}$$

となる。$\ln(p/(1-p))$ を p のロジットという。今、ある説明変数 x_k の値が $x_k \to x_k + \Delta x_k$ と変化したとき、

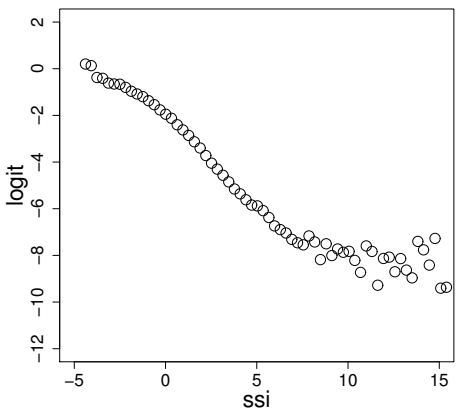

図 2.5.4 MSM の SSI と発雷の有無のロジットの関係。2012 年 3 月から 2013 年 2 月の 1 年間のデータを利用。

現象ありのロジットが $\mathrm{logit}(p) \to \mathrm{logit}(p) + \Delta\mathrm{logit}(p)$ と変化したとすると、

$$\Delta\mathrm{logit}(p) = \Delta x_k w_k \tag{2.5.31}$$

となる。これは、ある期間のデータに対して、横軸にある説明変数をとり、縦軸にその説明変数のある区間内に含まれる実況の有無（1 か 0 か）から算出したロジットをプロットした場合、説明変数がロジットと線形関係にあることを意味している。このことはまた、線形重回帰（第 2.4.9 項 (1)）でも述べた通り、ある変数 u の関数 $f(u)$（$f(u) = u$ を含む）が既に説明変数に用いられている場合、別の関数 $g(u)$ を説明変数に用いることはできないことも意味している。例として、MSM 発雷確率ガイダンスの説明変数に用いられている SSI と発雷の有無のロジットの関係を図 2.5.4 に示す。これを見ると、多少の非線形性はあるものの、発雷の予測に重要な 0~5 付近では SSI とロジットは概ね線形関係にあることが分かる。

(3) 観測の独立性

ロジスティック回帰では各観測データが独立であることを仮定している。すなわち観測データについて、空間方向や時間方向に相関がないことを仮定している。例えば図 2.5.5 のようなデータの場合、時間方向に相関があるため、時刻や季節で層別化したり、時間変化や季節変化を考慮した説明変数を加えたり、相関を考慮してデータを間引いたりする必要がある。また、同じ観測データを初期時刻の異なる説明変数に対応させた場合、同じような説明変数に対して同じ観測値を用いることになるため、相関の強い観測を用いたことと同様の影響を与えてしまう。同じ観測データを複数回使用しないように、回帰式を初期時刻や予報時間で層別化する必要がある。

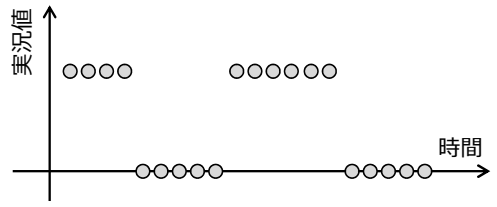

図 2.5.5 時間方向に相関のある観測データの例

(4) その他の注意事項

ここまで述べた (1)~(3) の事項のほかに、線形重回帰と同様の注意点も挙げられる。説明変数間に多重共線性がある場合には、ロジスティック回帰の場合でも係数の推定精度が低下するため、説明変数を選択して使用する説明変数の数を減らすか、学習データの数を増やして係数推定の誤差を減らす必要がある。また、第 2.4.9 項 (3) で述べたように、学習データやパラメータ調整用のデータとは独立なデータを用いて性能を評価する必要があることにも注意が必要である。

2.5.7 まとめ

本節ではロジスティック回帰の基礎的な理論及び利用上の注意点を述べた。ロジスティック回帰は目的変数が 2 値データである場合に発生確率を予測することができ、シンプルな手法でありながら従来用いられていた手法と比べて高い予測精度を持つことから、気象庁のガイダンスにおいても確率予測に広く利用されている。開発においては、ロジスティック回帰を用いて予測した場合の検証結果（ブライアスコアや信頼度曲線など）だけに着目するのではなく、第 2.5.6 項で述べたように、実況データや説明変数をプロットしてみて、データの特性がロジスティック回帰に適しているか確認することも重要である。

参考文献

Cox, D. R., 1958: The regression analysis of binary sequences. *Journal of the Royal Statistical Society. Series B*, **20(2)**, 215–242.

Engel, J., 1988: Polytomous logistic regression. *Statistica neerlandica*, **42(4)**, 233–252.

McCullagh, P., 1980: Regression models for ordinal data. *Journal of the Royal Statistical Society. Series B*, **42(2)**, 109–142.

高田伸一, 2007: 航空気象予報ガイダンス. 平成 19 年度数値予報研修テキスト, 気象庁予報部, 87–93.

2.6 ニューラルネットワーク [1]

2.6.1 はじめに

ニューラルネットワークは、神経細胞（ニューロン）の機能の一部をモデル化した機械学習アルゴリズムである。その研究は 1940 年代に始まり、Rosenblatt (1958, 1962) によるパーセプトロンと誤り訂正学習法を機に第 1 次ブームが、Rumelhart et al. (1986) による誤差逆伝播法の再発見を機に第 2 次ブームが起き、盛んに研究が行われた。その後 2012 年に ILSVRC [2] において Hinton らのグループ (Krizhevsky et al. 2012) がニューラルネットワークの階層を深くしたディープニューラルネットワーク (Hinton et al. 2006; LeCun et al. 2015) を用いて圧勝したことで第 3 次ブームが始まり、2018 年現在も大きな注目を集めている。

ニューラルネットワークの特徴の一つは、第 2.6.3 項でも示すように、線形分離不可能な（任意の）関係も扱うことができることである。これに対して線形重回帰やロジスティック回帰では線形分離可能な関係しか扱うことができない。またニューラルネットワークでは、線形重回帰やロジスティック回帰で課されていた仮定がなく、目的変数や説明変数の関係を詳しく吟味する必要がないという特徴もある。このためネットワークの構築に関する自由度が非常に高い一方で、調整すべきパラメータが多くなり、計算を効率的に行うことが重要になっている。

気象庁のガイダンスにニューラルネットワークが利用されるようになったのは 1996 年から (柳野 1995) であり、基本的には 1980 年代から 1990 年代に開発された、いわゆる第 2 世代ニューラルネットワークの技術に基づいている。2018 年現在では最大降水量ガイダンス、降雪量地点ガイダンス、日照率ガイダンス、最小湿度ガイダンス、雲ガイダンスにニューラルネットワークが利用されている。

本節では、第 2 世代ニューラルネットワークの技術を中心に解説するが、近年開発され、ディープニューラルネットワークに用いられている技術についてもいくつか述べる。これらの新しい技術は、第 2 世代のニューラルネットワークにもそのまま適用可能であり、ガイダンスの予測精度の向上に資すると考えられる。より詳しい解説は、例えば Kermanshahi (1999)、岡谷 (2015)、斎藤 (2016) などを参照していただきたい。以下では、単にニューラルネットワークと書いた場合には、第 2 世代のニューラルネットワークまたは、ディープニューラルネットワークまで含めた一般的な意味でのニューラルネットワークを指すものとする。

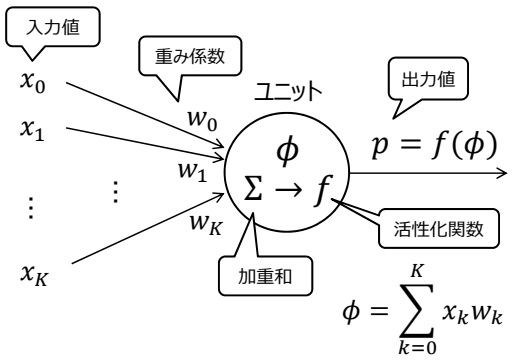

図 2.6.1 1 つのユニットへの入出力の模式図

本節では、初めに第 2.6.2 項から第 2.6.4 項でニューラルネットワークの構成と簡単な例を示し、第 2.6.5 項と第 2.6.6 項でニューラルネットワークによる予測値の算出と係数の学習について述べる。続いて第 2.6.7 項と第 2.6.8 項でニューラルネットワークの学習に必要な様々な技術を述べ、最後に利用上の注意点を述べる。ディープニューラルネットワーク（またはディープラーニング）の概要は第 5.2 節で述べる。ニューラルネットワークでは、その性質上ほかの統計手法と比べて変数や添字の数が多くなる。付録 2.3.A に、本章でも用いられる主な変数と添字をまとめてあるので、適宜参照していただきたい。また、ネットワークの構成によって重み係数 w などの添字の付き方が変わるため、一般的な議論をする場合にはいくつかの添字を省略する場合がある。具体的な表現を知りたい場合には適宜添字を補っていただきたい。

2.6.2 ニューラルネットワークを構成するパーツ

ニューラルネットワークは、図 2.6.1 で示したユニット [3] を組み合わせることで構成される。各ユニットに対して K 個の入力データ $\boldsymbol{x}$ があり、それぞれに対応する K 個の重み係数 $\boldsymbol{w}$ を掛けて加重和 ϕ を求める。ここで 0 番目の係数 w_0 はバイアス項であり、対応する入力値 x_0 は常に 1 であるとする。その後 ϕ を活性化関数 f で変換した結果を出力値 p として出力する。活性化関数としては、下記の関数など [4] が用いられている（図 2.6.2）。

$$\text{ステップ関数}: f(\phi) = \begin{cases} 1 & (\phi > 0) \\ 0 & (\phi \leq 0) \end{cases} \tag{2.6.1}$$

$$\text{ロジスティック関数}: f(\phi) = \frac{1}{1 + e^{-\phi}} \tag{2.6.2}$$

$$\text{双曲線正接関数}: f(\phi) = \tanh \phi \tag{2.6.3}$$

[1] 工藤 淳

[2] ImageNet Large Scale Visual Recognition Challenge (`http://www.image-net.org/challenges/LSVRC/`)。米国の複数の大学が共同で開催している大規模画像識別コンペティション。1000 クラスの物体判別などを競っている。

[3] ノードやニューロン、パーセプトロンと呼ばれることもある。

[4] ほかにも、ソフトサイン関数: $\phi/(1 + |\phi|)$、ソフトプラス関数: $\ln(1 + e^{\phi})$、動径基底関数: $e^{-\beta\phi}$、多項式: ϕ^n、絶対値: $|\phi|$ などもある。

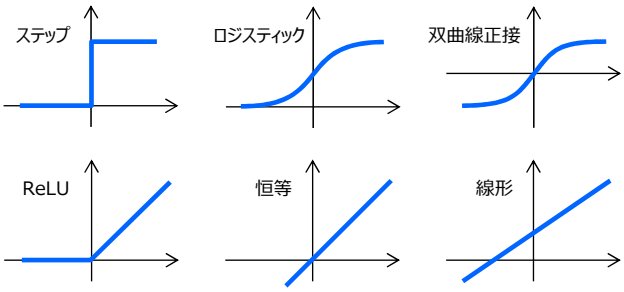

図 2.6.2　活性化関数の例

$$\text{ReLU 関数} : f(\phi) = \begin{cases} \phi & (\phi > 0) \\ 0 & (\phi \leq 0) \end{cases} \tag{2.6.4}$$

$$\text{恒等関数} : f(\phi) = \phi \tag{2.6.5}$$

$$\text{線形関数} : f(\phi) = a\phi + b \tag{2.6.6}$$

気象庁のガイダンスでは、ロジスティック関数、恒等関数、線形関数が用いられている。双曲線正接関数はロジスティック関数と同様な形をしているが、原点で回転対称性があるという特徴がある。ReLU 関数（Rectified Linear Unit（正規化線形関数）またはランプ関数）はシンプルでありながら非線形な関数で、無限大でも勾配が 0 にならないという特徴を持つ。従来用いられていたロジスティック関数や双曲線正接関数と比べて精度が向上する (Glorot et al. 2011; LeCun et al. 2015) ことから、ディープニューラルネットワークの多くでは ReLU 関数が用いられている。ここで示した例のうち、ステップ関数、ロジスティック関数、双曲線正接関数、ReLU 関数は非線形関数で、残りの 2 つは線形関数である。次項で示すように、ニューラルネットワークでは活性化関数が非線形関数であるか否かが重要な意味を持つ。なお、ニューラルネットワークに関する多くのテキストやこれまでのガイダンスの解説ではロジスティック関数のことをシグモイド関数と呼んでいるが、本稿ではロジスティック関数と呼ぶことにする。シグモイド関数は狭義にはロジスティック関数を指すが、一般には双曲線正接関数や累積正規分布関数などの S 字をした関数全般を指す。ここでは関数型を明確にするためと、第 2.5 節のロジスティック回帰の解説と用語を統一するために、ロジスティック関数と呼ぶことにする。

2.6.3　簡単な例

ここでは簡単な例を用いて、ニューラルネットワークが線形分離不可能な関係も表現可能であることと、活性化関数にロジスティック関数や ReLU 関数などのステップ関数以外の非線形関数が用いられている理由を述べる。

図 2.6.3 のようにユニットに 2 つのデータ x_1 と x_2 が入力され、p が出力されるものとする。それぞれの値は

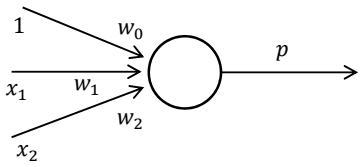

図 2.6.3　論理積を表現するユニット

表 2.6.1　論理積

	x_1	x_2	出力
1	1	1	1
2	1	0	0
3	0	1	0
4	0	0	0

表 2.6.2　排他的論理和

	x_1	x_2	出力
1	1	1	0
2	1	0	1
3	0	1	1
4	0	0	0

0 か 1 である。このとき表 2.6.1 で示した論理積 (AND) をユニットを用いて表現することを考える。活性化関数にはステップ関数を用いることにする。すなわち、

$$\phi = w_0 + x_1 w_1 + x_2 w_2$$

$$p = \begin{cases} 1 & (\phi > 0) \\ 0 & (\phi \leq 0) \end{cases} \tag{2.6.7}$$

とする。ここで例えば、$(w_0, w_1, w_2) = (-5, 3, 3)$ とすると、$x_1 = 1, x_2 = 1$ の場合には $\phi = 1 > 0$ で $p = 1$ などとなることから、論理積が表現できていることを確かめられる。この $\boldsymbol{w}$ を用いて (2.6.7) 式を書くと、

$$p = \begin{cases} 1 & (3x_1 + 3x_2 - 5 > 0) \\ 0 & (3x_1 + 3x_2 - 5 \leq 0) \end{cases} \tag{2.6.8}$$

となる。これは、$3x_1 + 3x_2 - 5 = 0$ を境に点 (x_1, x_2) がこの線よりも上側にあれば 1 を下側にあれば 0 を返す、と解釈できる（図 2.6.4 左）。このように、一本の直線でデータを分類できることを線形分離可能という。逆の見方をすれば、$\boldsymbol{w}$ としてどのような値を用いたとしても、1 本の直線でしかクラスを分類できないとい

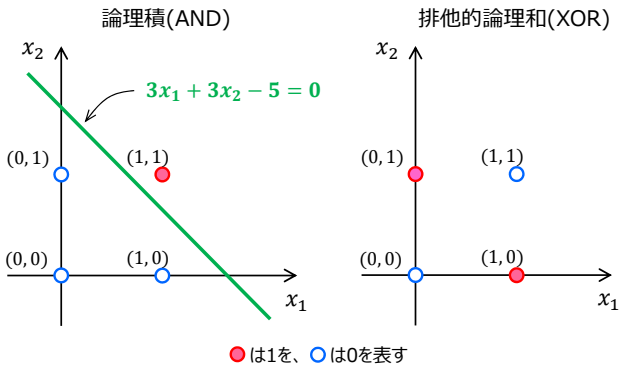

図 2.6.4　論理積（左）と排他的論理和（右）の図。赤丸は 1 を、青丸は 0 を表す。

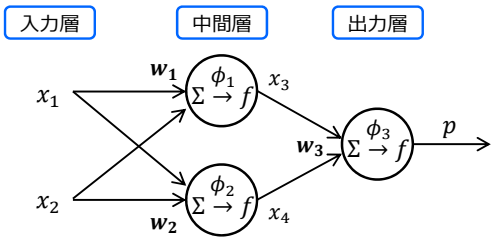

図 2.6.5　ユニットを 2 層に組み合わせたネットワーク。図を簡略化するために、ここではそれぞれのユニットへのバイアス項を省略している。

える。注意したいのは、活性化関数に用いたステップ関数は非線形関数であるにも関わらず、線形分離可能な問題しか解くことができないということである。これは活性化関数にロジスティック関数など他の非線形関数を用いても同じである。ロジスティック関数でクラス分類をする場合、例えば $p = 0.5$ を閾値として 1 か 0 かを分類することになる。(2.6.2) 式より、$p = 0.5$ となるのは $\phi = 0$ のときであり、やはり $\phi = 0$ という直線を境に 2 つのクラスに分類されることになる。

線形分離不可能な問題の例として、排他的論理和 (XOR) がある（表 2.6.2）。排他的論理和の入出力の関係を図に示すと図 2.6.4 右のようになる。この図を見れば分かるように、排他的論理和の場合には 1 本の直線では 1 と 0（赤丸と青丸）を分離することはできない。すなわち線形分離不可能である。そこで、ユニットを複数組み合わせて図 2.6.5 のようなネットワークを組むことにする。重み係数 $\boldsymbol{w}$ は、

$$\begin{aligned} \boldsymbol{w}_1 &= (w_{10}, w_{11}, w_{12}) = (5, -3, -3) \\ \boldsymbol{w}_2 &= (w_{20}, w_{21}, w_{22}) = (-3, 5, 5) \\ \boldsymbol{w}_3 &= (w_{30}, w_{31}, w_{32}) = (-5, 3, 3) \end{aligned} \tag{2.6.9}$$

であるとする。ここで、w_{10}, w_{11}, w_{12} などはそれぞれ、1 番目のユニットでのバイアス項と x_1, x_2 の重みを表す。活性化関数は全てステップ関数を用いる。この係数を用いて x_1, x_2 に 0 または 1 を与えて具体的に計算すると表 2.6.3 のようになり、排他的論理和を表現できていることがわかる。図 2.6.5 のように、複数のユニットを層に並べたとき、入力データが与えられる層を入力層、計算結果が出力される層を出力層と呼び、その中間の層を中間層（または隠れ層）と呼ぶ。中間層は 1 層だけでも良いし 2 層以上あっても良い。

論理積と排他的論理和を解くにあたり、上記では重み係数はパラメータとして与えられたが、実際の分類問題に適用しようとした場合には何らかの方法で自動的に最適な係数を求めなくてはならない。論理積については次の誤り訂正学習法と呼ばれる手法で係数を自

表 2.6.3　図 2.6.5 のネットワークに (2.6.9) 式の係数を与えて計算した結果

	x_1	x_2	x_3	x_4	出力
1	1	1	0	1	0
2	1	0	1	1	1
3	0	1	1	1	1
4	0	0	1	0	0

動的に求めることができる。

$$w_k^{(s+1)} = w_k^{(s)} + \eta \sum_n (y_n - p_n^{(s)})\, x_{nk} \tag{2.6.10}$$

ここで s は学習のステップ、n は学習データの番号（表 2.6.1 の左列の番号 1~4 に相当）、y_n は n 番目の教師データ（表 2.6.1 では右列の値に相当）、$p_n^{(s)}$ は s ステップ目の係数を用いて計算した n 番目の出力値、x_{nk} は n 番目のデータにおける x_k の値、η は正の学習率である。$\eta = 0.1$ などとし、ある適当な初期値 $\boldsymbol{w}_k^{(0)}$ から始めて上記ステップを複数回繰り返すことで正しい係数が得られる。

排他的論理和の場合は誤り訂正学習法では正しい係数を求めることはできないのだが、活性化関数に非線形関数を採用し、第 2.6.6 項で述べる誤差逆伝播法を用いることで最適な係数を学習することができる。ただし、活性化関数はステップ関数以外の非線形関数である必要がある。なぜならば、誤差逆伝播法では活性化関数の微分を用いて係数を更新するため、微分がデルタ関数になるステップ関数では係数を適切に更新できないからである。このため、第二世代以降のニューラルネットワークでは、活性化関数にロジスティック関数や ReLU 関数など、ステップ関数以外の非線形関数が用いられている。

1 つのユニットでは表現できなかった排他的論理和を表現できたのは、ユニットを 2 層に重ねたためである。中間層の活性化関数に非線形関数を用いて、ユニットを 2 層以上重ねることで、理論上は任意の非線形クラス分類や回帰を近似できる (Cybenko 1989)。では中間層の活性化関数が線形関数 (2.6.6) 式だったらどうか。これは実際に計算してみれば容易に確認できるが、中間層の活性化関数を線形関数にすると、出力層の加重和（図 2.6.5 の場合は ϕ_3）は入力値 $\boldsymbol{x}$ の線形結合で表される。これはすなわち中間層がないネットワークと等価であることを意味しており、非線形分類問題を解くことはできない。よって、ニューラルネットワークの特性を活かすためには中間層に非線形関数を用いる必要がある。中間層の活性化関数が非線形関数であれば、出力層の活性化関数は線形関数や恒等関数を用いても良い（非線形関数でも良い）。

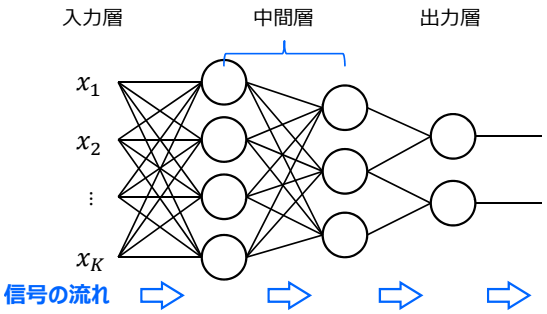

図 2.6.6　順伝播型ニューラルネットワークの構成

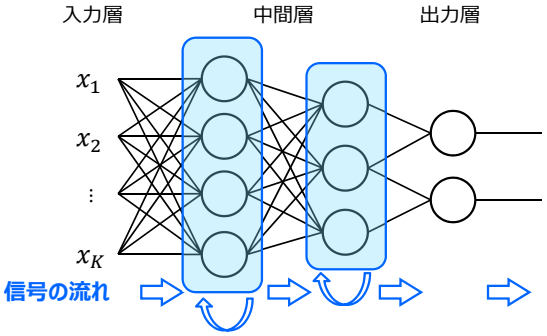

図 2.6.7　再帰型ニューラルネットワークの構成

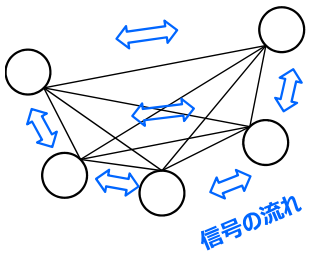

図 2.6.8　相互結合型ニューラルネットワークの構成

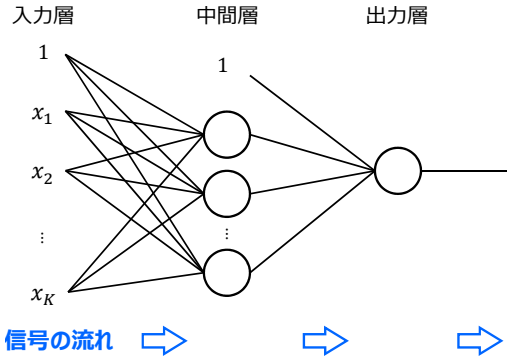

図 2.6.9　ガイダンスに用いられているニューラルネットワークの構成。入力層と中間層の 1 はバイアス項に対応。

2.6.4　ニューラルネットワークの構成

ニューラルネットワークは、ネットワークの構成によって様々な種類がある。主なネットワークには以下のようなものがある。

- 順伝播型ニューラルネットワーク
- 再帰型ニューラルネットワーク
- 相互結合型ニューラルネットワーク

順伝播型（フィードフォワード）ニューラルネットワークは、図 2.6.6 のようにユニットを複数の階層に並べたニューラルネットワークで、信号が入力層から出力層に向かって一方向のみに伝播していく。中間層は 1 層以上あり、それぞれの層のユニットは複数ある。画像認識などの分類・識別問題に用いられる最も基本的なニューラルネットワークである。

再帰型（リカレント）ニューラルネットワーク (Elman 1990) は、図 2.6.7 のように順伝播型ニューラルネットワークの中間層に有向閉路 [5] を持たせたネットワークである。中間層では前回の出力値が入力値として扱われるため、短期の記憶として作用する。これによって前後のデータとの関連性を扱うことができ、音声認識や言語処理などの時系列データの処理に用いられる。短期の記憶だけでなく、長期の記憶も持たせた LSTM (Long Short-Term Memory, Hochreiter and Schmidhuber 1997) と呼ばれるネットワークもある。

[5] 方向を持つ（双方向ではない）閉じた経路のこと。

相互結合型ニューラルネットワークは、図 2.6.8 のように自分自身を除く全てのユニットと結合したネットワークである。連想記憶や組み合わせ最適化問題に利用されるホップフィールドネットワーク (Hopfield 1982) や、ホップフィールドネットワークでのユニットの状態変化を確率的に行うボルツマンマシン (Ackley et al. 1985) などがある。

気象庁のガイダンスで用いられているニューラルネットワークは、図 2.6.9 のように、中間層が 1 層で出力層のユニット数が 1 の順伝播型ニューラルネットワークの構造を持っている。説明変数を入力層に与えることで、出力層からガイダンスの予測値が得られる。図 2.6.9 のように中間層が 1 層のニューラルネットワークを、本稿では（入力層 1 層と出力層 1 層を合わせて）3 層ニューラルネットワーク [6] と呼ぶ。

ここまで見てきたように、ニューラルネットワークには主に 3 つの構成があり、中間層の層数やユニットの数、出力ユニットの数などに自由度があるが、本節では気象庁のガイダンスに用いられている、中間層が 1 層の順伝播型ニューラルネットワークを中心に解説する。

2.6.5　順方向の計算

前節で見たように、順伝播型ニューラルネットワークでは、入力層に与えられた信号は出力層に向かって

[6] ユニットとしての重なりは 2 層であるため、2 層ニューラルネットワークと呼ぶ文献もある。

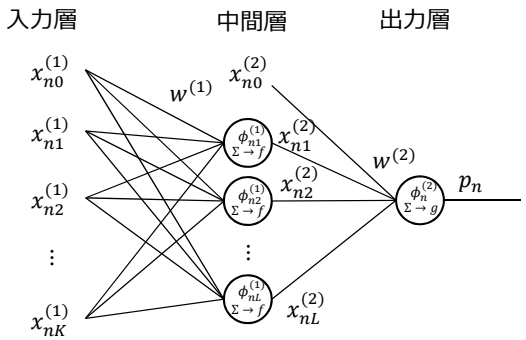

図 2.6.10 出力が 1 ユニットの 3 層順伝播型ニューラルネットワーク

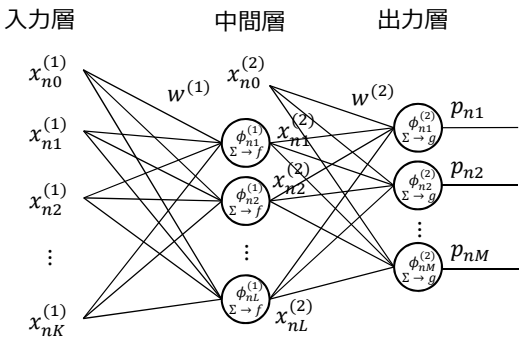

図 2.6.11 出力が M ユニットの 3 層順伝播型ニューラルネットワーク

一方向（順方向）に伝播して何らかの値を出力する。以下では 3 層ニューラルネットワークについて、出力層のユニット数が 1 個の場合と M 個の場合の順方向の計算を行う。本節と次節では、3 層の順伝播型ニューラルネットワークについて述べるため、記述を簡略化するためにこれを単にニューラルネットワークと呼ぶ。

(1) 出力が 1 ユニットの場合

図 2.6.10 に、気象庁のガイダンスで用いられているものと同じ、出力が 1 ユニットのニューラルネットワークの構成を示す。今、N 個の学習データが与えられており、その中の n 番目の入力データに対する出力値を求めることを考える。ここでは係数 $\boldsymbol{w}$ は何らかの方法で既に与えられているものとする。入力層から中間層への計算は以下の通りである。

$$\phi_{nl}^{(1)} = \sum_{k=0}^{K} x_{nk}^{(1)} w_{kl}^{(1)} \tag{2.6.11}$$

$$x_{nl}^{(2)} = f\left(\phi_{nl}^{(1)}\right) \tag{2.6.12}$$

ここで、$x_{nk}^{(1)}$ は図に示したように 1 番目の層、すなわち入力層から中間層への入力値で、$x_{n0}^{(1)} = 1$ である。$w_{kl}^{(1)}$ は入力層からの入力データに対応する中間層の l 番目のユニットの係数、$x_{nl}^{(2)}$ は中間層の l 番目のユニットからの出力値で、f は中間層の活性化関数を表す。中間層の活性化関数は、気象庁のガイダンスではロジスティック関数が用いられているが、ディープニューラルネットワークでは ReLU 関数がよく用いられている。

中間層から出力層への計算は以下の通りである。

$$\phi_{n}^{(2)} = \sum_{l=0}^{L} x_{nl}^{(2)} w_{l}^{(2)} \tag{2.6.13}$$

$$p_n = g\left(\phi_{n}^{(2)}\right) \tag{2.6.14}$$

出力層の活性化関数 g は、目的変数に応じて決めることになる。基本的には、目的変数が連続値（回帰）の場合には恒等関数を、2 値分類（確率予測）の場合にはロジスティック関数が用いられる。

(2) 出力が複数ユニットの場合

図 2.6.11 に、出力層のユニットが M 個ある場合のニューラルネットワークの構成を示す。出力が複数ユニットのネットワークは 2018 年現在の気象庁のガイダンスでは用いられていないが、手書き文字判別などの多クラス分類問題に適用できるため、ニューラルネットワークでは頻繁に用いられている。多クラス分類は、天気カテゴリーの判別や降水量の階級判別などガイダンスにも応用できるため、ここで出力層のユニットが多数ある場合の順方向の計算についても書いておく。

入力層から中間層への計算は (2.6.11) 式および (2.6.12) 式と全く同じである。中間層から出力層への計算は以下の通りである。

$$\phi_{nm}^{(2)} = \sum_{l=0}^{L} x_{nl}^{(2)} w_{lm}^{(2)} \tag{2.6.15}$$

$$p_{nm} = g\left(\phi_{nm}^{(2)}\right) \tag{2.6.16}$$

ここで m は出力層のユニット番号である。出力層の活性化関数には次のソフトマックス関数が用いられる。

$$g\left(\phi_{nm}^{(2)}\right) = \frac{e^{\phi_{nm}^{(2)}}}{\sum_{i=1}^{M} e^{\phi_{ni}^{(2)}}} \tag{2.6.17}$$

ソフトマックス関数は 0 ～ 1 の値をとり、全てのクラスについて和をとると 1 になるため、それぞれのクラスに分類される確率を表すと解釈できる。複数のクラスから一つのクラスを選ぶ場合には最も確率の高いクラスを選択すれば良い。ただし単に確率の高いクラスを選ぶだけならば (2.6.17) 式を計算する必要はなく、最も大きい $\phi_{nm}^{(2)}$ を選択するだけで良い。ソフトマックス関数によって求められた確率値は、次節の係数学習時に誤差関数を求める際に利用される。

実用上の問題として、$\phi_{nm}^{(2)}$ が大きくなると $e^{\phi_{nm}^{(2)}}$ の計算がオーバーフローする場合がある。このような場合の対策として、$\phi_{ni}^{(2)}$ の中の最大値を α_n としたとき、(2.6.17) 式の分母分子に予め $e^{-\alpha_n}$ を掛けておけばよい。

$$g\left(\phi_{nm}^{(2)}\right) = \frac{e^{\phi_{nm}^{(2)} - \alpha_n}}{\sum_{i=1}^{M} e^{\phi_{ni}^{(2)} - \alpha_n}} \tag{2.6.18}$$

中間層の層数が 2 層以上ある場合も、層の数に応じてここまでに述べた計算を繰り返していくだけで順方向の計算を行うことができる。

2.6.6 係数の学習

第 2.6.3 項で述べた誤り訂正学習法では、線形分離可能な場合しか適切に学習することはできなかったが、第 2.3.10 項で述べた最急降下法や確率的勾配降下法を用いることで、線形分離不可能な場合でも学習によって適切な係数を求めることが可能になる。

ニューラルネットワークでは (2.3.53) 式 ∼(2.3.55) 式を用いることで一括学習、ミニバッチ学習、逐次学習のいずれも可能である。一括学習の場合は全ての学習データを用いて学習し、ミニバッチ学習の場合はランダムに選んだ 1 つまたは複数のサンプルに対して学習する。逐次学習では新しいデータを入手する度にそのデータを用いて学習する。いずれの学習方法でも、係数の初期値にはランダムな値を与えることが多い（第 2.6.8 項 (3) も参照)。一括学習とミニバッチ学習の場合は、学習データに対して多数回の係数更新ステップ [7] を繰り返して係数を決定する。ガイダンスに用いられているニューラルネットワークの逐次学習の場合は、運用時には前回の係数を 1 ステップだけ更新するが、運用開始前にミニバッチ学習などを用いて多数回の繰り返し学習を行い、学習期間のデータに対して事前に係数を最適化している。

一括学習を用いた場合、1 ステップの係数更新を行うためには全ての学習データで算出した誤差関数 E_n に対して w_k での微分を求める必要があるため計算に時間が掛かるが、ミニバッチ学習では、ランダムに選択された一部の学習データに対して微分を計算すれば良いため、計算量が少なくなるというメリットがある。またミニバッチ学習を用いた場合には、選ばれたサンプルに応じて誤差関数の勾配の方向が変わるため、大域的な極小解が得られやすい (Keskar et al. 2017) という特徴がある。これにより一括学習を用いた場合と比べて過学習が抑制され予測精度（汎化能力）が高くなる傾向がある。このような理由により、近年のニューラルネットワークでは一括学習に代わってミニバッチ学習が用いられることが多く、降雪量地点ガイダンスでもミニバッチ学習が用いられている。

(2.3.54) 式で示したように、ミニバッチ学習では係数を以下のようにして決定する。

$$w_k^{(s+1)} = w_k^{(s)} - \eta \sum_{n\in\mathcal{D}} \left.\frac{\partial E_n}{\partial w_k}\right|_{\boldsymbol{w}=\boldsymbol{w}^{(s)}} \tag{2.6.19}$$

ここで、η は学習率、E_n は n 番目の学習データに対する誤差関数、$\mathcal{D}$ は N 個の学習データの中からランダムに選ばれたサンプルを表す。選ばれたサンプルの数をバッチサイズと呼ぶ。ミニバッチ学習ではサンプルの選び方に自由度があるが、通常はバッチサイズを固定して、ステップごとに選択するサンプルを変える。適切なバッチサイズを事前に知ることはできないが、多すぎるとサンプルによる勾配のばらつきが小さくなってしまうため、ミニバッチ学習の良さが失われてしまう。バッチサイズとしては 10∼100 程度にすることが多い (岡谷 2015)。降雪量地点ガイダンスの場合は数 100 程度の学習データに対してバッチサイズを 10 としている。

ミニバッチ学習を用いる場合、バッチサイズはパラメータであり、適切な値に設定するために、バッチサイズを変えた実験を何度か繰り返すことになる。このとき、トータルの誤差関数 $E = \sum_{n\in\mathcal{D}} E_n$ はバッチサイズに比例して大きくなる。なぜならば E_n は常に正なので、サンプルごとの誤差が同程度だとすれば、バッチサイズに比例してトータルの誤差関数は大きくなるからである。このため η を固定した場合、(2.6.19) 式での係数の 1 回の更新幅が設定したサンプル数に比例して変化してしまう。そこで、各誤差関数をバッチサイズ $N_{\mathcal{D}}$ で割った値として定義しておけば、バッチサイズを変える度に η を調整する必要がなくなる。これに加えて、開発の過程の中で、一括学習、ミニバッチ学習、逐次学習を使い分けながら最もよい手法を選択する、ということも考慮すれば、学習方法によらず 1 回の学習に利用するデータ数（以下、本項では学習方法に関わらずこれを N と表記する）で割った値を E_n と定義しておけば、その都度誤差関数の定義を変える必要がないため便利である。

E_n の具体的な表現は出力層の活性化関数に応じて設定する。活性化関数が線形関数や恒等関数の場合（回帰の場合）は二乗誤差が用いられる。

$$E_n = \frac{1}{2N}(p_n - y_n)^2 \tag{2.6.20}$$

活性化関数がロジスティック関数の場合（確率予測の場合）は上記の二乗誤差または負の対数尤度が用いられる。

$$E_n = -\frac{1}{N}\left[y_n \ln p_n + (1-y_n)\ln(1-p_n)\right] \tag{2.6.21}$$

2018 年現在の気象庁のガイダンスでは二乗誤差を用いているが、近年のニューラルネットワークでは負の対数尤度がよく用いられている。活性化関数がソフトマックス関数の場合（多クラス分類の場合）は交差エントロピーが用いられる。

$$E_n = -\frac{1}{N}\sum_{m=1}^{M} y_{nm} \ln p_{nm} \tag{2.6.22}$$

ここで y_{nm} の m 列成分は M 個の中の 1 つだけが 1 で残りは全て 0 のベクトル（one-hot ベクトル）で、$\sum_{m=1}^{M} y_{nm} = 1$ が成り立つ。また、各クラスに分類される確率の和は 1 であることから、$\sum_{m=1}^{M} p_{nm} = 1$ が

[7] 繰り返し回数のことをエポック数とも呼ぶ

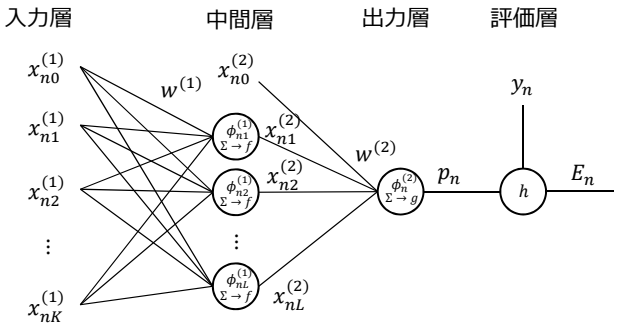

図 2.6.12 評価層を加えた出力が 1 ユニットの 3 層順伝播型ニューラルネットワーク

成り立つ。$M = 2$、すなわち 2 クラス分類の場合には、交差エントロピーは (2.6.21) 式と一致する。

一括学習、ミニバッチ学習、逐次学習のいずれを利用する場合でも、$\partial E_n/\partial w_k$ を求める必要がある。以下では数値微分と誤差逆伝播法という $\partial E_n/\partial w_k$ を求める 2 つの手法について述べる。

(1) 数値微分による勾配計算

誤差関数の w_k 方向の勾配を求める方法として、最も単純には、w_k を Δw_k だけ変化させて順方向の計算を行い、E_n の変化量 ΔE_n を求めれば良いだろう。ただしニューラルネットワークの出力は w_k に対して非線形なので中心差分を取ることにする。$w_k \to w_k + \Delta w_k$ としたとき、$E_n \to E_n + \Delta E_n^+$、$w_k \to w_k - \Delta w_k$ としたとき、$E_n \to E_n + \Delta E_n^-$ であるとすれば、

$$\frac{\partial E_n}{\partial w_k} \simeq \frac{\Delta E_n^+ - \Delta E_n^-}{2\Delta w_k} \tag{2.6.23}$$

となる。これを全ての w_k に対して行うことで係数を更新できる。この方法を数値微分と呼ぶ。

数値微分による学習は単純だが、1 つの係数を 1 ステップ更新する度に順方向の計算を 2 回行う必要があるため、全体の学習に掛かる時間は係数の数に比例して長くなってしまう。ニューラルネットワークは他の統計手法と比べて学習に掛かる時間が長い上に、第 2.6.1 項でも述べたように調整すべきパラメータが多く、最適なパラメータを得るために何度も繰り返して学習する必要がある。このため 1 ステップの係数更新に掛かる時間を短くすることは開発を効率的に行う上で非常に重要である。勾配計算を高速に行う方法が次に述べる誤差逆伝播法 (Rumelhart et al. 1986) である。

(2) 誤差逆伝播法による勾配計算

ここでは気象庁のガイダンスに用いられている、中間 1 層、出力 1 ユニットの場合の誤差逆伝播法について述べる。出力が多ユニットの場合や中間層が 2 層以上ある場合もここで述べる方法を拡張することで容易に計算できる。誤差逆伝播法の流れを把握するために、図 2.6.12 のように図 2.6.10 のネットワークに評価層を追加し、評価層ではニューラルネットワークの出力と教師データから関数 h により誤差関数 E_n を出力すると考える。

表 2.6.4 出力層の活性化関数と用いられる誤差関数

	誤差関数	出力層の活性化関数
①	二乗誤差	線形関数（恒等関数）
②	二乗誤差	ロジスティック関数
③	負の対数尤度	ロジスティック関数

係数更新のために最終的に求めたい値は E_n の w 方向の勾配 $\partial E_n/\partial w_{kl}^{(1)}$ と $\partial E_n/\partial w_l^{(2)}$ だが、まずは p_n 方向の勾配 $\partial E_n/\partial p_n$ を計算する。出力層が 1 ユニットの場合、活性化関数は線形関数（恒等関数を含む）かロジスティック関数である。この場合の誤差関数は (2.6.20) 式か (2.6.21) 式である。よって $\partial E_n/\partial p_n$ は、誤差関数が二乗誤差の場合は

$$\frac{\partial E_n}{\partial p_n} = \frac{1}{N}(p_n - y_n) \tag{2.6.24}$$

誤差関数が負の対数尤度の場合は

$$\frac{\partial E_n}{\partial p_n} = \frac{1}{N}\frac{p_n - y_n}{p_n(1 - p_n)} \tag{2.6.25}$$

となる。このことは、教師データ y_n が与えられれば、順方向の計算時に得られた出力値（すなわち既知の値）p_n を用いることで容易に $\partial E_n/\partial p_n$ が算出できることを表している。

続いて $w_l^{(2)}$ 方向の E_n の勾配を計算する。p_n は $w_l^{(2)}$ の関数であるから、合成関数の微分の連鎖公式により、

$$\frac{\partial E_n}{\partial w_l^{(2)}} = \frac{\partial E_n}{\partial p_n}\frac{\partial p_n}{\partial \phi_n^{(2)}}\frac{\partial \phi_n^{(2)}}{\partial w_l^{(2)}} \tag{2.6.26}$$

と書ける。$\partial E_n/\partial p_n$ は既に求めてあるので残りの部分を求める。$\partial p_n/\partial \phi_n^{(2)}$ は、出力層の活性化関数が線形関数 $p_n = a\phi_n^{(2)} + b$ の場合は

$$\frac{\partial p_n}{\partial \phi_n^{(2)}} = a \tag{2.6.27}$$

ロジスティック関数の場合は

$$\frac{\partial p_n}{\partial \phi_n^{(2)}} = p_n(1 - p_n) \tag{2.6.28}$$

となる。また $\partial \phi_n^{(2)}/\partial w_l^{(2)}$ は、$\phi_n^{(2)} = \sum_{l=0}^{L} x_{nl}^{(2)} w_l^{(2)}$ より、

$$\frac{\partial \phi_n^{(2)}}{\partial w_l^{(2)}} = x_{nl}^{(2)} \tag{2.6.29}$$

となる。以上を組み合わせることで $\partial E_n/\partial w_l^{(2)}$ が計算できる。以後の計算でも出力層の活性化関数と誤差関数の組み合わせにより計算式が異なるので、組み合わ

せを表 2.6.4 にまとめておく。それぞれの組み合わせについて誤差関数の勾配を計算すると、①の場合は、

$$\frac{\partial E_n}{\partial w_l^{(2)}} = \frac{a}{N}(p_n - y_n)x_{nl}^{(2)} \tag{2.6.30}$$

②の場合は、

$$\frac{\partial E_n}{\partial w_l^{(2)}} = \frac{1}{N}(p_n - y_n)p_n(1-p_n)x_{nl}^{(2)} \tag{2.6.31}$$

③の場合は、

$$\frac{\partial E_n}{\partial w_l^{(2)}} = \frac{1}{N}(p_n - y_n)x_{nl}^{(2)} \tag{2.6.32}$$

となる。出力層の活性化関数がロジスティック関数の場合、誤差関数に負の対数尤度を用いると、(2.6.32) 式のように (2.6.30) 式と同型になってシンプルに書けるため、近年のニューラルネットワークでは③の組み合わせが用いられている。w 方向の誤差関数の微分を求める際に、$\partial E_n/\partial p_n$ を求め、続いて $\partial p_n/\partial \phi_n^{(2)}$ を求め、$\cdots$、というように、図 2.6.12 で見た場合に順方向の計算方向とは逆向きに誤差の微分を計算することから、この方法を誤差逆伝播法という。

(2.6.30) 式 ∼(2.6.32) 式の計算に必要な変数は p_n, y_n, $x_{nl}^{(2)}$ であるが、y_n は教師データとして、$x_{nl}^{(2)}$ は説明変数として与えられている値であり、p_n は順方向の計算を 1 回行えば求められる値である。数値微分で勾配を求める場合には係数の数に比例して順方向の計算を行う必要があったが、誤差逆伝播法の場合には順方向の計算を 1 回行うだけで全ての係数について勾配が求まる。さらにいえば、(2.6.26) 式のうち $\partial E_n/\partial p_n \cdot \partial p_n/\partial \phi_n^{(2)}$ は全ての $w_l^{(2)}$ について共通であるから、1 つの $w_l^{(2)}$ について計算した結果を保持しておけばこの部分を毎回計算する必要もなくなる。このため誤差逆伝播法を用いれば勾配計算を高速に行うことができる。

次に $\partial E_n/\partial w_{kl}^{(1)}$ を求める。合成関数の微分より、

$$\frac{\partial E_n}{\partial w_{kl}^{(1)}} = \frac{\partial E_n}{\partial p_n}\frac{\partial p_n}{\partial \phi_n^{(2)}}\frac{\partial \phi_n^{(2)}}{\partial x_{nl}^{(2)}}\frac{\partial x_{nl}^{(2)}}{\partial \phi_{nl}^{(1)}}\frac{\partial \phi_{nl}^{(1)}}{\partial w_{kl}^{(1)}} \tag{2.6.33}$$

と書ける。ここでは出力 1 ユニットの場合のみ扱うが、出力層のユニットが複数ある場合には、合成関数の微分は

$$\frac{\partial E_n}{\partial w_{kl}^{(1)}} = \sum_{m=1}^{M}\frac{\partial E_n}{\partial p_{nm}}\frac{\partial p_{nm}}{\partial \phi_{nm}^{(2)}}\frac{\partial \phi_{nm}^{(2)}}{\partial x_{nl}^{(2)}}\frac{\partial x_{nl}^{(2)}}{\partial \phi_{nl}^{(1)}}\frac{\partial \phi_{nl}^{(1)}}{\partial w_{kl}^{(1)}} \tag{2.6.34}$$

のように和をとる必要があることに注意する。(2.6.33) 式の右辺の初めの 2 つは (2.6.24) 式 ∼(2.6.28) 式で既に求めてあるので残りの 3 つを計算する。$\phi_n^{(2)} = \sum_{l=0}^{L} x_{nl}^{(2)} w_l^{(2)}$ および $\phi_{nl}^{(1)} = \sum_{k=0}^{K} x_{nk}^{(1)} w_{kl}^{(1)}$ であることと、中間層の活性化関数がロジスティック関数 $x_{nl}^{(2)} = (1+e^{-\phi_{nl}^{(1)}})^{-1}$ であるならば、

$$\frac{\partial \phi_n^{(2)}}{\partial x_{nl}^{(2)}} = w_l^{(2)} \tag{2.6.35}$$

$$\frac{\partial x_{nl}^{(2)}}{\partial \phi_{nl}^{(1)}} = x_{nl}^{(2)}\left(1 - x_{nl}^{(2)}\right) \tag{2.6.36}$$

$$\frac{\partial \phi_{nl}^{(1)}}{\partial w_{kl}^{(1)}} = x_{nk}^{(1)} \tag{2.6.37}$$

となる。これらを合わせると、①と③の場合（③の場合は $a=1$ と見る）は、

$$\frac{\partial E_n}{\partial w_{kl}^{(1)}} = \frac{a}{N}(p_n - y_n)w_l^{(2)}x_{nl}^{(2)}\left(1 - x_{nl}^{(2)}\right)x_{nk}^{(1)} \tag{2.6.38}$$

②の場合は、

$$\frac{\partial E_n}{\partial w_{kl}^{(1)}} = \frac{1}{N}(p_n - y_n)p_n(1-p_n)w_l^{(2)}x_{nl}^{(2)}\left(1 - x_{nl}^{(2)}\right)x_{nk}^{(1)} \tag{2.6.39}$$

となる。以上で中間 1 層、出力 1 ユニットの場合の勾配を誤差逆伝播法で求めることができた。

誤差逆伝播法を利用することの最大の利点は、前述のとおり勾配計算を高速で行うことができることである。一方で、誤差逆伝播法では各ユニットでの入出力に対する局所的な勾配を掛け合わせることで誤差関数の勾配を求めているため、数値微分が順方向の計算に基づく非線形演算であるのに対して、誤差逆伝播法は線形演算 [8] となっている。このことは、誤差関数の情報が逆向きに伝播していく過程で、どこかのユニットで勾配が 0 または ∞ に近づいたとすると、そのユニットにつながる全ての係数の勾配が 0 または ∞ になってしまうことを意味する。活性化関数にロジスティック関数などのシグモイド型の関数を用いている場合、入力値が 0 から遠ざかると勾配が急速に 0 に近くなることから、このようなことは容易に起こりうる。勾配が 0 に近づくと係数の更新速度が非常に遅くなってしまい計算効率が悪くなり、∞ に近づくと係数の更新が非常に不安定になってしまう。これを勾配消失問題という。ニューラルネットワークの階層を深くすると、その分だけ微分を掛け合わせる数が多くなるため、勾配消失問題が起きやすくなる。このことが第 2 次ニューラルネットワークブームを終焉させた理由の一つ [9] で

[8] 数値微分では w の変化量を 2 倍にしても誤差関数は 2 倍にはならないが、誤差逆伝播法では誤差関数が 2 倍になれば w 方向の勾配も 2 倍になる。

[9] ニューラルネットワークでは係数以外のパラメータ（学習率やユニットの数、層の数など）が多く、これらを適切に決定する方法が見つかっていなかったことや、これらのパラメータを設定するコストの割に第 2.10 節で述べるサポートベクターマシンやランダムフォレストなどの手法と比べて精度が高くなかったことも理由の一つ。

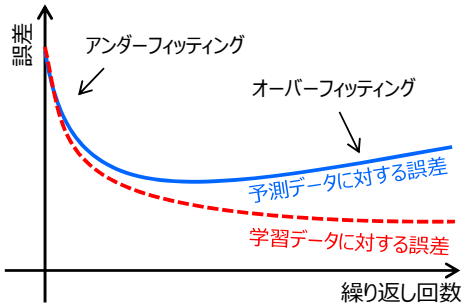

図 2.6.13 学習回数を変化させたときの学習データと予測データそれぞれに対する誤差の変化

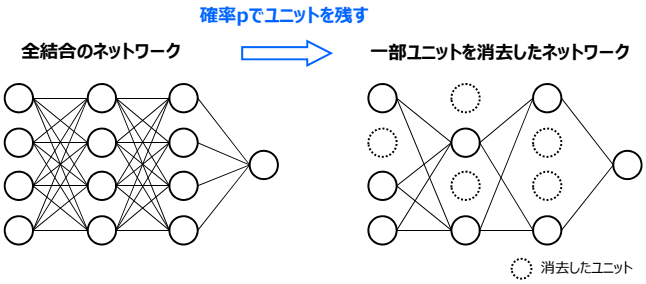

図 2.6.14 ドロップアウトの概念図

あり、勾配消失問題を回避する様々な手法が開発されてネットワークの階層を深くすることが可能になったことが第 3 次ニューラルネットワークブームを生み出した一つの要因となっている（岡谷 (2015) など）。

2.6.7 過学習とその対策

ニューラルネットワークでの係数の学習は最急降下法や確率的勾配降下法を用いて行うため、誤差関数の最小値を探索するために係数更新ステップを多数回繰り返して行うことになる。このとき、繰り返し回数が少ないと学習が十分に進まず誤差が大きくなる（アンダーフィッティング）ためある程度以上の繰り返し計算が必要になるが、繰り返し回数が多すぎると学習データに対する誤差は小さくなる一方で、未知データに対する誤差は大きくなってしまう。これを過学習（オーバーフィッティング）という（図 2.6.13）。

学習データの数に対してネットワークの自由度（ユニット数など）が多すぎる場合、ネットワークの表現能力が高過ぎるために繰り返し回数を多くすると学習データに適合しすぎて過学習が生じる。このため過学習対策としては、過学習の状態になる前に繰り返し学習を止める方法と、ネットワークの自由度を減らす方法の 2 つが考えられる。前者の方法として早期終了 (Early Stopping) が、後者の方法として正則化とドロップアウトがある。これらは組み合わせて用いることもできる。

早期終了は、学習データを係数学習用と検証用に分け、検証用データに対して誤差が増加し始めたところで学習を終了する手法である。早期終了は簡単な方法で予測データに対する誤差を評価できる一方、学習データの中から検証用のデータを用意する必要があるため、学習用のサンプルが減るという問題がある。学習データが少ない場合には交差検証を用いる方法もある。

正則化では誤差関数に係数の大きさに応じたペナルティー項を加えることで過学習を抑制する。すなわち、

$$E_n \to E_n + \frac{\lambda}{N\beta} \sum_{j,k,l} \left| w_{kl}^{(j)} \right|^{\beta} \tag{2.6.40}$$

とする。線形重回帰（第 2.4.6 項）などと同様に、$\beta = 1$ ならば L1 正則化、$\beta = 2$ ならば L2 正則化などとなる。

ネットワークの自由度が高すぎて過学習を起こすというのであれば、自由度（ユニット数）を減らせば良いだろうというのがドロップアウト (Hinton et al. 2012; Srivastava et al. 2014) の考え方である。ドロップアウトでは、学習時の係数更新ステップ毎にユニットの一部をランダムに消去する [10]（図 2.6.14）。消去されたユニットからは、順方向・逆方向とも情報は伝達されない。ユニットを残す割合 p は入力層と中間層で異なる値を用いても良い。ドロップアウトを用いて学習した場合でも、予測時には全てのユニットを用いる。ただし、学習時に確率 p でユニットを残したことにより各ユニットの寄与量（または係数の大きさ）が平均的に $1/p$ 倍になっていることから、これを戻すために各係数には p を掛けた値を用いる。

ドロップアウトにおいて、ユニットをランダムに消去することは、ネットワークの自由度を強制的に減らしているという意味のほかに、様々なネットワークで学習した結果をアンサンブル平均しているという意味もある。複数の手法で予測した結果を平均すると予測精度が高くなる場合が多いが、複数の手法やネットワークで学習・予測することは計算コストや維持コストが掛かってしまう。ドロップアウトはこれと比べると十分に低コストで実装できる。

2.6.8 学習のテクニック

線形重回帰やロジスティック回帰、カルマンフィルタなどの統計手法と比べ、ニューラルネットワークにはユニット数・学習率・正則化係数などの開発者が設定しなければならないパラメータが多く、適切な値を見つけるためには学習と検証を何度も繰り返す必要がある。また、係数の数が多く、係数空間上の誤差関数が複雑であるため、パラメータの設定によっては学習に時間が掛かったり最小値探索が失敗したりする場合もある。よってニューラルネットワークを用いた開発を効率的に進めるためには学習を効率的に行う必要があり、そのための様々なテクニックが開発されている。ここではその中からよく用いられている手法を紹介する。

[10] どのユニットを消去するかは確率的に決まるため、場合によってはある層の全てのユニットが消去されることもある。

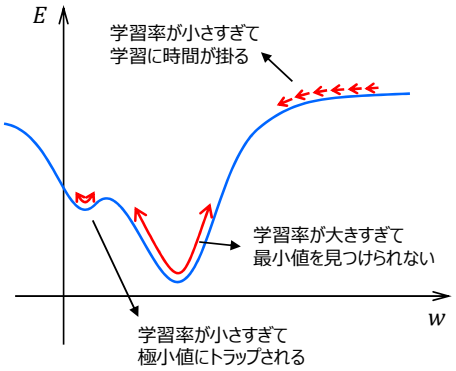

図 2.6.15　学習率の大きさと最小値探索の概念図

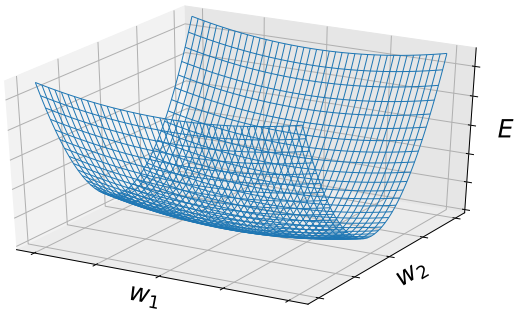

図 2.6.16　誤差関数が一つの方向に緩やかに変化する例

(1)　学習率に関する手法

ニューラルネットワークの係数の学習に最急降下法や確率的勾配降下法を用いる場合、学習率 η が小さすぎると学習に時間が掛かりすぎたり、極小値にトラップされたり、鞍部から抜け出すのに時間が掛かったりし、逆に大きすぎると最小値付近で振動して最小値を見つけられなかったりする（図 2.6.15）。このため学習率を適切に設定することは重要であるが、ニューラルネットワークの誤差関数は一般に複雑な形をしているため、固定の学習率を用いることには限界がある。特に図 2.6.16 のように、w のある成分に対して誤差関数が緩やかに変化する場合、単純な最急降下法ではそれに直行する方向に振動してしまい学習効率が悪くなる。ただし最小値付近で振動しているのであれば、最小値から多少離れた場所で計算を終了したとしても最小値と比べて誤差は同程度であるだろうから、予測精度には大きな違いはないかもしれない。しかし誤差関数が緩やかな鞍部になっている場合には、誤差の変化が小さいことから最小値に達したと判断し学習を止めてしまうと、本来の最小値から離れた係数を採用することになり、予測精度を低下させる原因となる。ネットワークのサイズが大きくなるほどこのような鞍部は多くなり、鞍部からいかにして抜け出すかということの方が重要になってくる (Dauphin et al. 2014)。

固定値ではない学習率を用いる方法として、初めは大きな学習率を使用し、学習が進むとともに小さくするという方法がある。例えば次のようにする。

$$\eta = \eta_0 - \alpha\, s \tag{2.6.41}$$

ここで η_0 は η の初期値、s は学習のステップで、α は正のパラメータである。このほかにも、初めは $\eta = \eta_0$ としておいて、ある程度学習が進んだと判断した時点で $\eta = 0.1\eta_0$ などとする方法もある。

より高度な方法として、モメンタム (Rumelhart et al. 1986) や AdaGrad (Duchi et al. 2011) などの方法もよく用いられる。モメンタムでは、前ステップとの係数の差を $\Delta w^{(s)} \equiv w^{(s)} - w^{(s-1)}$ として、各ステップの更新に慣性項 $\alpha\Delta w^{(s)}$ を加える。

$$w^{(s+1)} = w^{(s)} - \eta \left.\frac{\partial E}{\partial w}\right|_{w=w^{(s)}} + \alpha\Delta w^{(s)} \tag{2.6.42}$$

ここで α は正のパラメータで、0.5 ～ 0.9 程度の値を設定する。慣性項は、w がある方向に振動している場合には振動を抑制し、一定方向に進んでいる場合には速度を増加させるように働くことで収束を早める。

AdaGrad では係数更新を次のように行う。

$$w_k^{(s+1)} = w_k^{(s)} - \frac{\eta}{\sqrt{h_k^{(s)}}} \left.\frac{\partial E}{\partial w_k}\right|_{w=w^{(s)}} \tag{2.6.43}$$

$$h_k^{(s)} = \sum_{t=0}^{s} \left.\left(\frac{\partial E}{\partial w_k}\right)^2\right|_{w=w^{(t)}} \tag{2.6.44}$$

ここで $h_k^{(s)}$ はステップ 0 から s までの w_k 方向の勾配の二乗和であり、$h_k^{(s)}$ が大きくなるほど w_k の更新速度が遅くなっていく。例えば勾配が w_k 方向には急であった場合、$h_k^{(s)}$ は急速に大きくなるため、w_k の更新速度はすぐに遅くなる。逆に勾配が w_k 方向には緩やかであったとすると、$h_k^{(s)}$ は小さな値のままになり、更新速度が大きい状態が維持される。AdaGrad は w のそれぞれの方向について個別に更新速度が調整される。

ほかにも AdaGrad を改良した AdaDelta (Zeiler 2012) や、AdaGrad とモメンタムを組み合わせた Adam (Kingma and Ba 2015) など様々な手法が提案されている。AdaGrad, AdaDelta, Adam は最急降下法やモメンタムと比べて鞍部から効率的に抜け出すことができるため、特に近年のディープニューラルネットワークで良く利用されている。基本的には単純な最急降下法を用いるよりもモメンタムや AdaGrad などを用いた方が収束が早くなるが、常に最適な手法は存在せず、状況に応じて使い分けることになる。2018 年現在の気象庁のガイダンスではモメンタムを用いているものが多い。

(2)　説明変数の標準化

第 2.3.10 項でも述べたが、ニューラルネットワークにおいても説明変数のオーダーに大きな差があると係

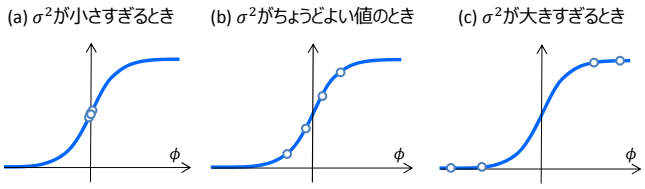

図 2.6.17　係数の初期値を用いて計算された中間層の各ユニットへの加重和（図中の丸）とそのユニットからの出力値の模式図（活性化関数がロジスティック関数の場合）。

数の学習効率が悪くなる。ニューラルネットワークでは係数の初期値は乱数で与えることが多いのだが、同じオーダーの乱数を初期値に与えた場合、説明変数のオーダーに大きな差があると加重和 $\phi = \sum_{k=0}^{K} x_k w_k$ を取ったときにオーダーの大きな説明変数だけが強調されてしまうため、大きなオーダーの説明変数を打ち消すような適切な係数に収束するまでに時間が掛かってしまう。このような理由から説明変数のオーダーに差がある場合には、学習期間の平均と標準偏差を用いて説明変数を標準化するか、最大・最小値を用いて 0 ～ 1 の値に変換しておく。

(3)　係数の初期値

ニューラルネットワークでは、係数の初期値は $N(0, \sigma^2)$ に従う乱数などで与える。これは初期値に揺らぎを与えずに完全に同じ値から学習を開始すると、中間層の各ユニットから次の層への出力は常に全て同じ値となり、中間層に複数のユニットを配置する意味がなくなるからである。よって係数の初期値にはある程度の揺らぎ（分散 σ^2）を与える必要があるのだが、σ^2 の与え方によって学習効率が変わってしまう。σ^2 が小さすぎると全ての係数の初期値が $w_0 \simeq 0$ となり、どのような組み合わせで説明変数が与えられたとしても、加重和はほとんど 0 になってしまう（図 2.6.17(a)）。これは係数の初期値に揺らぎを全く与えない場合に近く、学習がほとんど進まなくなってしまう。一方で σ^2 が大きすぎると、中間層への入力の加重和が 0 から大きく離れた値を取る可能性が高くなる（図 2.6.17(c)）。この時、図のように活性化関数にロジスティック関数などのシグモイド型の関数を用いていると、中間層からの出力が 0 または 1 に近い値に固定されてしまい、活性化関数の傾きが小さくなるため、誤差逆伝播法での学習が進まなくなってしまう（勾配消失問題が起きる）。以上のことから、図 2.6.17(b) のように、中間層への入力の加重和がちょうど良い分散を持つように w の初期値の分散 σ^2 を決めれば良いといえる。

Glorot and Bengio (2010) は、活性化関数 f が対称でかつ 0 での微分係数が 1、すなわち $f'(0) = 1$ の場合に、係数の初期値の分散を $\sigma^2 = 1/M$ にする手法を示した。ここで M は前の層のユニット数である。この初期化方法は Xavier の初期値と呼ばれ、ディープニューラルネットワークで広く利用されている。

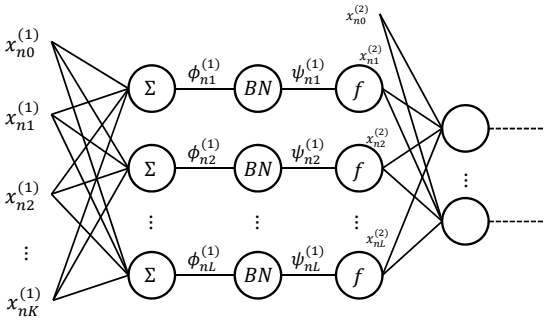

図 2.6.18　バッチ標準化を活性化関数の前に行う場合の模式図。バッチ標準化の演算を BN で表す。

He et al. (2015) は、活性化関数が非対称な ReLU 関数の場合には係数の初期値の分散を $\sigma^2 = 2/M$ とすることを提案し、ReLU 関数の場合に Xavier の初期値を用いた場合と比べて学習が速く進むことと、ネットワークを深くした場合に Xavier の初期値と比べて係数の収束性が良いことを示した [11]。He et al. (2015) による初期化の手法は He の初期値と呼ばれている。

Kumar (2017) は Xavier の初期値を 0 で微分可能な活性化関数に拡張し、中間層のユニット数が多い場合、活性化関数を 1 次の微分までで近似することで、

$$\sigma^2 = \frac{1}{M f'(0)^2 \left(1 + f(0)^2\right)} \tag{2.6.45}$$

とする手法を示した（この式の導出は付録 2.6.A に示す）。これを用いると、例えば 0 での微分係数が 1 になる双曲線正接関数では $\sigma^2 = 1/M$ となり Xavier の初期値と一致することが確かめられる。また、ロジスティック関数の場合には $f(0) = 0.5$, $f'(0) = 0.25$ より、$\sigma^2 = 64/(5M)$ となる。Kumar (2017) は活性化関数にロジスティック関数を用いた 10 層のニューラルネットワークで画像識別を行い、(2.6.45) 式を用いることで Xavier の初期値を利用した場合と比べて非常に速く学習が進むことを示している。

(4)　バッチ標準化

中間層からの出力値の分散が大きすぎたり小さすぎたりすることが問題、というのであれば、いつでも適切な分散になるように強制的に変換してやればよいだろう、というアイディアがバッチ標準化 (Batch Normalization, Ioffe and Szegedy 2015) である。バッチ標準化の演算は中間層の活性化関数の前か後に行う。中間層のユニットを演算の役割ごとに分割すると図 2.6.18 のようになる。図で BN はバッチ標準化の演算を表す。

バッチ標準化ではミニバッチ学習を行うことを前提としている。ミニバッチ学習による 1 ステップの係数

[11] 22 層と 30 層のネットワークで比較し、Xavier の初期値では 22 層では収束するが、30 層では収束しない一方、彼らの初期化を用いた場合には 30 層のネットワークでも収束することを示した。

更新を考えた場合、ミニバッチの中には $N_{\mathcal{D}}$ 個のサンプルが存在する。それぞれのサンプルに対して順方向の計算を行うことで、$N_{\mathcal{D}}$ 個の中間層への入力の加重和 $\phi^{(1)}$ が得られることになる。これらの加重和の標本平均 $\bar{\phi}_l^{(1)}$ と標本分散 ${s_l^{(1)}}^2$ はそれぞれ

$$\bar{\phi}_l^{(1)} = \frac{1}{N_{\mathcal{D}}} \sum_{n=1}^{N_{\mathcal{D}}} \phi_{nl}^{(1)} \tag{2.6.46}$$

$${s_l^{(1)}}^2 = \frac{1}{N_{\mathcal{D}}} \sum_{n=1}^{N_{\mathcal{D}}} \left(\phi_{nl}^{(1)} - \bar{\phi}_l^{(1)} \right)^2 \tag{2.6.47}$$

である。バッチ標準化では各加重和を

$$\hat{\phi}_{nl}^{(1)} = \frac{\phi_{nl}^{(1)} - \bar{\phi}_l^{(1)}}{\sqrt{{s_l^{(1)}}^2 + \epsilon}} \tag{2.6.48}$$

として標準化した後に、

$$\psi_{nl}^{(1)} = \gamma^{(1)} \hat{\phi}_{nl}^{(1)} + \beta^{(1)} \tag{2.6.49}$$

と変換する。ここで ϵ はゼロ割を防ぐための小さな数で、$\beta^{(1)}$と$\gamma^{(1)}$ は $\beta^{(1)} = 0,\ \gamma^{(1)} = 1$ を初期値としておいて、誤差逆伝播法で学習させる係数である。同様の演算を全ての中間層に対して行う。バッチ標準化を用いることで、学習率を大きくできて学習に掛かる時間を短縮できる、係数の初期値のことを細かく考えなくてもよい、過学習が抑制できる、という優れた効果があり、精度向上に大きく寄与することから 2018 年現在では広く利用されている。

2.6.9 まとめと利用上の注意点

本節では気象庁のガイダンスに利用されている出力 1 ユニットの 3 層順伝播型ニューラルネットワークを中心に、ニューラルネットワークでの予測値の計算、係数の学習方法、過学習への対策や学習のテクニックについて述べた。

ニューラルネットワークと線形重回帰やロジスティック回帰の大きな違いは、ニューラルネットワークは目的変数と説明変数が非線形関係を持つ場合にも適用できることと、係数を決定する上での仮定（制約）がないことが挙げられる [12]。ニューラルネットワークでは、線形重回帰やロジスティック回帰では注意しなければならなかった目的変数や説明変数の多重共線性や線形性、等分散性などを考慮する必要はない。ただし学習効率の観点から、第 2.6.8 項の (2) で述べたように説明変数を標準化する必要はある。また、予測対象が連続値（回帰）か 2 クラス分類（確率予測）か多クラス分類かなど、問題に応じてネットワークや活性化関数を設定する必要がある。

[12] ディープニューラルネットワークでは特徴量（顔の輪郭や目の形など）を自動的に学習することも特徴の一つに挙げられる。

ニューラルネットワークは適用範囲が広い反面、他の統計手法と比べて開発者が設定しなければならないパラメータが多く、かつ適切に設定しないと学習がうまく進まないことや、学習に時間が掛かるという問題がある。このため、学習を効率的に行うことがガイダンスの精度を向上させる上で重要なポイントとなる。また、線形重回帰やロジスティック回帰、および次節で述べるカルマンフィルタと比べて、予測の根拠（どの説明変数がどれくらい寄与しているか、異常な値が予測された場合に何が原因だったのかなど）を理解することが困難であるという問題がある。このため、ガイダンスを開発・運用するに当たっては、中間層からの出力値およびその分布、各ユニットの重み係数、説明変数を変化させた場合の予測値などをモニターすることも重要である。

中間層のユニット数や学習率、正則化の係数など、パラメータを調整する際には、学習データを係数学習用のデータとパラメータ調整の評価用データに分けて、評価用データに対して精度が高くなるようにパラメータを調整する（パラメータ調整の評価に検証用データを用いてはならない）。よって、パラメータの調整も含めてニューラルネットワークを用いたガイダンスを開発する場合には、用意したデータを係数学習用データ、パラメータ調整の評価用データ、予測精度検証用のデータに分割して使用することになる。分割することでサンプル数が少なくなる場合には交差検証の利用を検討する。

近年のニューラルネットワークブームにより、有効な手法が次々と生み出されている。これらの手法はディープニューラルネットワークでの利用を目的としたものではあるが、気象庁のガイダンスに利用されている 3 層のニューラルネットワークにも適用可能であるため、最新の研究の動向を踏まえつつ、有効な手法を適宜導入するよう検討していきたい。

付録 2.6.A Kumar による初期値

ここでは Kumar (2017) に基づいて (2.6.45) 式を導出する。全部で L 層（入力層を含めると $L+1$ 層）の深いネットワークを考える。第 l 層のユニット数を M、第 $l+1$ 層のユニット数を K とすると、第 $l+1$ 層からの出力値 $x_k^{(l+1)}$ は、

$$x_k^{(l+1)} = f\left(\phi_k^{(l)}\right) \tag{2.6.50}$$

$$\phi_k^{(l)} = \sum_{m=1}^{M} \psi_{mk}^{(l)} \tag{2.6.51}$$

$$\psi_{mk}^{(l)} = x_m^{(l)} w_{mk}^{(l)} \tag{2.6.52}$$

と書ける。ここで、係数 w の初期値の分布は $N(0, v^2)$ であるとする。また、入力データは $N(0,1)$ に標準化されているものとする。記述の定義として、$\phi_m^{(l)}$ の標本平均を $\bar{\phi}^{(l)}$、標本分散を ${s^{(l)}}^2$ と書く。

1 回目の更新ステップを考えた場合、$w_{mk}^{(l)}$ は $x_m^{(l)}$ と独立である。また、1 ステップ目の順方向の計算では、第 l 層への入力データ $x_m^{(l)}$ の期待値と分散はユニットによらず等しいとする。すなわち、

$$E\left(x_m^{(l)}\right) = \mu^{(l)} \tag{2.6.53}$$

$$V\left(x_m^{(l)}\right) = {\sigma^{(l)}}^2 \tag{2.6.54}$$

と書けるものとする。ここで知りたいのは 1 回目の更新ステップで各層への入力データの分散が全て等しい値 1、つまり、

$${\sigma^{(1)}}^2 \simeq {\sigma^{(2)}}^2 \simeq \cdots {\sigma^{(L)}}^2 = 1 \tag{2.6.55}$$

を満たすような w の初期値の分散 v^2 である。$w_{mk}^{(l)}$ の期待値が 0 であることから、$\bar{\phi}^{(l)}$ の期待値は、

$$\begin{aligned} E\left(\bar{\phi}^{(l)}\right) &= E\left(\phi_k^{(l)}\right) = E\left(\sum_{m=1}^{M} x_m^{(l)} w_{mk}^{(l)}\right) \\ &= \sum_{m=1}^{M} E\left(x_m^{(l)}\right) E\left(w_{mk}^{(l)}\right) = 0 \end{aligned} \tag{2.6.56}$$

となる。また $\phi_k^{(l)}$ の分散は、

$$\begin{aligned} V\left(\phi_k^{(l)}\right) &= E\left({\phi_k^{(l)}}^2\right) - E\left(\phi_k^{(l)}\right)^2 \\ &= \sum_{m,j} E\left(x_m^{(l)} w_{mk}^{(l)} x_j^{(l)} w_{jk}^{(l)}\right) \\ &= \sum_{m=j} E\left(x_m^{(l)} w_{mk}^{(l)} x_j^{(l)} w_{jk}^{(l)}\right) \\ &= \sum_{m=1}^{M} E\left({x_m^{(l)}}^2 {w_{mk}^{(l)}}^2\right) \\ &= \sum_{m=1}^{M} E\left({x_m^{(l)}}^2\right) E\left({w_{mk}^{(l)}}^2\right) \end{aligned} \tag{2.6.57}$$

である。ここで、(2.6.53) 式と (2.6.54) 式より、

$$\begin{aligned} {\sigma^{(l)}}^2 &= V\left(x_m^{(l)}\right) = E\left({x_m^{(l)}}^2\right) - E\left(x_m^{(l)}\right)^2 \\ &= E\left({x_m^{(l)}}^2\right) - {\mu^{(l)}}^2 \end{aligned} \tag{2.6.58}$$

である。これと、w の初期値の分散が v^2、すなわち $V\left(w_{mk}^{(l)}\right) = E\left({w_{mk}^{(l)}}^2\right) = v^2$ を (2.6.57) 式に代入すると、

$$V\left(\phi_k^{(l)}\right) = Mv^2\left({\sigma^{(l)}}^2 + {\mu^{(l)}}^2\right) \tag{2.6.59}$$

と書ける。ここで、1 ステップ目では、$w_{mk}^{(l)}$ は $x_m^{(l)}$ と独立であることから、

$$E\left(\psi_{mk}^{(l)}\right) = E\left(x_m^{(l)}\right) E\left(w_{mk}^{(l)}\right) = 0 \tag{2.6.60}$$

かつ、$i \neq j$ について、

$$\begin{aligned} &Cov\left(\psi_{mi}^{(l)}, \psi_{mj}^{(l)}\right) \\ &= E\left(\psi_{mi}^{(l)} \psi_{mj}^{(l)}\right) - E\left(\psi_{mi}^{(l)}\right) E\left(\psi_{mj}^{(l)}\right) \\ &= E\left(x_m^{(l)} w_{mi}^{(l)} x_m^{(l)} w_{mj}^{(l)}\right) \\ &= E\left(x_m^{(l)}\right) E\left(w_{mi}^{(l)}\right) E\left(x_m^{(l)}\right) E\left(w_{mj}^{(l)}\right) \\ &= 0 \end{aligned} \tag{2.6.61}$$

であることから、$\psi_{mi}^{(l)}$ と $\psi_{mj}^{(l)}$ $(i \neq j)$ は互いに独立かつ同じ確率分布を持つ。よって M が大きい場合には中心極限定理より、$\psi_{mk}^{(l)}$ の和である $\phi_k^{(l)} = \sum_{m=1}^{M} \psi_{mk}^{(l)}$ は正規分布で近似できる。このことは 1 ステップ目以外でも概ね成り立つ。

ここで、活性化関数 $f(\phi)$ が $\phi = 0$ で微分可能であると仮定する。(2.6.50) 式を 0 のまわりでテーラー展開して ϕ の 1 次までで近似すると、

$$x_k^{(l+1)} \simeq f(0) + f'(0)\,\phi_k^{(l)} \tag{2.6.62}$$

であるから、(2.6.56) 式より、

$$E\left(x_k^{(l+1)}\right) \simeq f(0) + f'(0) E\left(\phi_k^{(l)}\right) = f(0) \tag{2.6.63}$$

となる。よって、$\mu^{(l+1)} \simeq f(0)$ であり、$\mu^{(l+1)}$ は l に依らないことが分かる。すなわち、$l \geq 1$ について、

$$\mu^{(l)} \simeq f(0) \tag{2.6.64}$$

といえる。これと (2.6.54) 式、(2.6.59) 式および (2.6.62) 式より、

$$\begin{aligned} {\sigma^{(l+1)}}^2 &= V\left(x_k^{(l+1)}\right) \simeq f'(0)^2 V\left(\phi_k^{(l)}\right) \\ &= Mv^2 f'(0)^2\left({\sigma^{(l)}}^2 + f(0)^2\right) \end{aligned} \tag{2.6.65}$$

となる。条件 (2.6.55) 式より、${\sigma^{(l)}}^2$ が l に依らず 1 であるから、

$$v^2 = \frac{1}{Mf'(0)^2\left(1 + f(0)^2\right)} \tag{2.6.66}$$

が成り立つ。

参考文献

Ackley, D. H., G. E. Hinton, and T. J. Sejnowski, 1985: A learning algorithm for boltzmann machines. *Cognitive Science*, **9(1)**, 147–169.

Cybenko, G., 1989: Approximation by superpositions of a sigmoidal function. *Mathmatics of Control, Signals and Systems*, **2(4)**, 303–314.

Dauphin, Y. N., R. Pascanu, C. Gulcehre, K. Cho, S. Ganguli, and Y. Bengio, 2014: Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. *Proceedings of the 27th International Conference on Neural Information Processing Systems*, 2933–2941.

Duchi, J., E. Hazan, and Y. Singer, 2011: Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, **12**, 2121–2159.

Elman, J. L., 1990: Finding structure in time. *Cognitive Science*, **14**, 179–211.

Glorot, X. and Y. Bengio, 2010: Understanding the difficulty of training deep feedforward neural networks. *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics*, 249–256.

Glorot, X., A. Bordes, and Y. Bengio, 2011: Deep Sparse Rectifier Neural Networks. *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics*, **15**, 315–323.

He, K., X. Zhang, S. Ren, and J. Sun, 2015: Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. *Proceedings of the 2015 IEEE International Conference on Computer Vision*, 1026–1034.

Hinton, G. E., S. Osindero, and Y. Teh, 2006: A fast learning algorithm for deep belief nets. *Neural Computation*, **18**, 1527–1554.

Hinton, G. E., N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, 2012: Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint, arXiv:1207.0580*.

Hochreiter, S. and J. Schmidhuber, 1997: Long short-term memory. *Neural Computation*, **9(8)**, 1735–1780.

Hopfield, J. J., 1982: Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences of the United States of America*, **79(8)**, 2554–2558.

Ioffe, S. and C. Szegedy, 2015: Batch Normalization: Accelerating deep network training by reducing internal covariate shift. *Proceedings of the 32nd International Conference on Machine Learning*, 448–456.

Kermanshahi, B., 1999: ニューラルネットワークの設計と応用. 昭晃堂, 146 pp.

Keskar, N. S., D. Mudigere, J. Nocedal, M. Smelyanskiy, and P. T. P. Tang, 2017: On large-batch training for deep learning: Generalization gap and sharp minima. *Conference paper at ICLR 2017*.

Kingma, D. P. and J. L. Ba, 2015: Adam: A method for stochastic optimization. *Conference paper at ICLR 2015*.

Krizhevsky, A., I. Sutskever, and G. E. Hinton, 2012: ImageNet Classification with Deep Convolutional Neural Networks. *Proceedings of the 25th International Conference on Neural Information Processing Systems*, 1097–1105.

Kumar, S. K., 2017: On weight initialization in deep neural networks. *arXiv preprint, arXiv:1704.08863*.

LeCun, Y., Y. Bengio, and G. E. Hinton, 2015: Deep learning. *Nature*, **521**, 436–444.

岡谷貴之, 2015: 機械学習プロフェッショナルシリーズ 深層学習. 講談社, 165 pp.

Rosenblatt, F., 1958: The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain. *Psychological Review*, **65(6)**, 386–408.

Rosenblatt, F., 1962: *Principles of neurodynamics; perceptrons and the theory of brain mechanisms*. Spartan Books, 616 pp.

Rumelhart, D. E., G. E. Hinton, and R. J. Williams, 1986: Learning representations by back-propagating errors. *Nature*, **323**, 533–536.

斎藤康毅, 2016: ゼロから作る Deep Learning. オライリー・ジャパン, 298 pp.

Srivastava, N., G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, 2014: Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, **15**, 1929–1958.

柳野健, 1995: ニューラルネットによるガイダンス. 平成 7 年度量的予報研修テキスト, 気象庁予報部, 54–69.

Zeiler, M. D., 2012: ADADELTA: An adaptive learning rate method. *arXiv preprint, arXiv:1212.5701*.

2.7 カルマンフィルタ [1]

2.7.1 はじめに

カルマンフィルタは、ノイズを持つ観測の時系列データを元に、時々刻々と変化するシステムの現在の状態を推定する時系列解析の手法の一つである。カルマンフィルタの応用範囲は広く、人工衛星やロボットの制御、カーナビ、物体追跡、経済学、統計学、気象予測など様々な分野で利用されている。気象予測においては、ガイダンスのほかにもデータ同化にカルマンフィルタが利用されている。データ同化で求めたいものは現在の大気の状態であり、これを各種観測データと第一推定値（数値予報モデルの予測値）を用いて推定する。ガイダンスで求めたいものは目的変数と説明変数を結びつける係数であり、これを観測値と第一推定値（前回の係数）を用いて推定することになる。データ同化の場合には大気の状態を表す変数の次元が多すぎるため、カルマンフィルタではなく、その近似である 4 次元変分法やアンサンブルカルマンフィルタなどが用いられるが (堀田・太田 2011)、ガイダンスの場合には変数の次元（係数の数）が少ないため、カルマンフィルタを直接用いて係数を求めることが可能である。

ここまで見てきた線形重回帰やロジスティック回帰、ニューラルネットワークは、一定期間のデータに対して最適な係数を決定する、という統計手法であった。特に一括学習を用いた場合、これらの手法ではデータの並び順に意味はなく、どの順番でデータが並んでいたとしても同じ結果が得られる。これに対してカルマンフィルタは時系列データを扱う手法であるという点で上記の手法と大きく異なる。カルマンフィルタではデータの並び順に意味があり、順番を替えると異なる係数が得られる。

ガイダンスでカルマンフィルタを利用することの最大の利点は、係数が最適な値になるように逐次更新されることにある。確率的勾配降下法を用いることでも係数を逐次更新することは可能だが、ノイズを考慮した時系列データを扱うという意味で、カルマンフィルタの方がより洗練された手法である。ガイダンスへのカルマンフィルタの利用に関する研究は 1980 年代に始まっている (Persson 1989, 1991; Simonsen 1991)。当時のガイダンスは主に線形重回帰による一括学習が用いられており、数値予報モデルの更新時に係数を再学習しなければガイダンスの予測精度が低下することが問題となっていた (Simonsen 1991)。気象庁のガイダンスにも同様の問題があり（第 1.3 節を参照）、1996 年からカルマンフィルタが利用されるようになった (瀬上ほか 1995)。2018 年現在では気温、風、平均降水量、降水確率、時系列湿度、視程の各ガイダンスにカルマンフィルタが利用されている。

[1] 工藤 淳

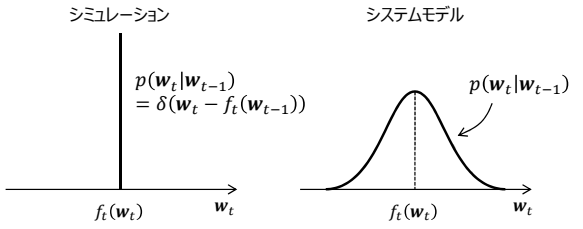

図 2.7.1 シミュレーションとシステムモデルの確率分布

本節では、初めにカルマンフィルタによる係数更新の関係式を導出し、続いてカルマンフィルタでのパラメータ設定や整合性の確認、利用上の注意点を述べる。カルマンフィルタの関係式の導出にはいくつかの方法が考案されており、最小分散推定値や線形最小分散推定値に基づいて解説される場合も多いが、本節では北川 (1993)、樋口 (2011) に基づき、係数の条件付き期待値から関係式を導出する。条件付き期待値は最小分散推定値と一致することが示されている (片山 2000)。

2.7.2 シミュレーションとシステムモデル

時刻 t の状態を表す変数 $\boldsymbol{w}$（これを状態変数と呼ぶ）の時間発展を表す方程式を $\boldsymbol{w}_t = f_t(\boldsymbol{w}_{t-1})$ と書くと、状態変数の初期値 $\boldsymbol{w}_0$ が与えられれば、任意の時刻の状態が厳密に求まる。これをシミュレーションという。シミュレーションでは結果は一意に決まるが、あえて確率密度関数で表すと、デルタ関数を用いて以下のように書ける。

$$p(\boldsymbol{w}_t|\boldsymbol{w}_{t-1}) = \delta(\boldsymbol{w}_t - f_t(\boldsymbol{w}_{t-1})) \tag{2.7.1}$$

ここでシミュレーションに“遊び”を許し、観測データを取り込める自由度を持った式 $\boldsymbol{w}_t \simeq f_t(\boldsymbol{w}_{t-1})$ へと拡張する。近似記号を用いる代わりに新たな項 $\boldsymbol{u}_t$ を導入し、以下のように書く。

$$\boldsymbol{w}_t = f_t(\boldsymbol{w}_{t-1}) + \boldsymbol{u}_t \tag{2.7.2}$$

これをシステムモデルという。$\boldsymbol{u}_t$ は $\boldsymbol{w}_t$ と独立なノイズで、システムノイズと呼ぶ。ガイダンスの場合には $\boldsymbol{w}_t$ は係数で、上式は時刻 $t-1$ の係数を用いて時刻 t の係数を求める式となる。シミュレーションでは結果は一つであるのに対し、システムモデルでは、$\boldsymbol{w}_{t-1}$ が与えられたときの $\boldsymbol{w}_t$ の確率密度関数は何らかの確率分布になる（図 2.7.1）。今、$\boldsymbol{u}_t$ が与えられたとして、(2.7.1) 式と同様に確率密度関数をデルタ関数を用いて書くと、

$$p(\boldsymbol{w}_t|\boldsymbol{w}_{t-1}) = \delta(\boldsymbol{w}_t - f_t(\boldsymbol{w}_{t-1}) - \boldsymbol{u}_t) \tag{2.7.3}$$

となる。

2.7.3 観測モデル

例えば日々の最高気温の時系列データなど、観測の時系列データを y_t とする。ガイダンスの場合、y_t はス

カラー量であるが、カルマンフィルタではベクトルとして扱われる。本節ではガイダンスに限らない一般的な場合として、初めは $\boldsymbol{y}_t$ をベクトルとして扱うことにする。

観測値 $\boldsymbol{y}_t$ と状態変数 $\boldsymbol{w}_t$ の関係を表す演算を h_t（これを観測演算子と呼ぶ）としたとき、シミュレーション的な考え方でいえば $\boldsymbol{y}_t = h_t(\boldsymbol{w}_t)$ であり、h_t と $\boldsymbol{w}_t$ が与えられれば $\boldsymbol{y}_t$ は一意に決まるが、通常は両者は一致せず、$\boldsymbol{y}_t \simeq h_t(\boldsymbol{w}_t)$ である。近似記号を用いる代わりにシステムモデルと同様に

$$\boldsymbol{y}_t = h_t(\boldsymbol{w}_t) + \boldsymbol{v}_t \tag{2.7.4}$$

と書き、この式を観測モデルという。$\boldsymbol{v}_t$ は $\boldsymbol{w}_t$ と独立なノイズで観測ノイズという。ガイダンスの場合には、$\boldsymbol{y}_t$ は目的変数、$h_t(\boldsymbol{w}_t)$ は時刻 t の予測値となる。(2.7.4) 式からもわかるように、観測ノイズは観測と予測の差を表す量である。観測ノイズという言葉から測定誤差のことを想像するかもしれないが、観測ノイズには測定誤差のほかに観測モデルが不完全であることによる誤差の両方が寄与していることに注意する。

2.7.4 一般状態空間モデル

システムモデルと観測モデルの連立モデルを状態空間モデルという。(2.7.2) 式と (2.7.4) 式をもっと一般的に、

$$\boldsymbol{w}_t \sim p(\boldsymbol{w}_t|\boldsymbol{w}_{t-1}) \tag{2.7.5}$$

$$\boldsymbol{y}_t \sim p(\boldsymbol{y}_t|\boldsymbol{w}_t) \tag{2.7.6}$$

と書いたとき、これを一般状態空間モデルという。後で述べるように、カルマンフィルタは最もシンプルな一般状態空間モデルである線形・ガウス状態空間モデルから導出される。

$\boldsymbol{w}_t$ や $\boldsymbol{y}_t$ の分布は、(2.7.5) 式や (2.7.6) 式のように $\boldsymbol{w}_{t-1}$ や $\boldsymbol{w}_t$ だけで決まるものではなく、それ以前の状態である $\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_{t-1}$ や $\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_t$ にも依存していると思うかもしれない。しかしここでは、時刻 t の条件付確率が時刻 $t-1$ の状態のみに依存し、それ以前の状態には依存しない、というマルコフ性の仮定を用いている。すなわち、

$$p(\boldsymbol{w}_t|\boldsymbol{w}_{1:t-1}, \boldsymbol{y}_{1:t}) = p(\boldsymbol{w}_t|\boldsymbol{w}_{t-1}) \tag{2.7.7}$$

$$p(\boldsymbol{y}_t|\boldsymbol{w}_{1:t}, \boldsymbol{y}_{1:t-1}) = p(\boldsymbol{y}_t|\boldsymbol{w}_t) \tag{2.7.8}$$

であることを仮定している。ここで、$x_{1:t}$ という表記は $x_1, x_2, \cdots, x_t$ を意味する。

ここではまず、一般状態空間モデルに対して $\boldsymbol{w}_t$ の確率分布を求め、その結果を線形・ガウス状態空間モデルに適用することで、カルマンフィルタでの係数更新式を導出する。

2.7.5 逐次ベイズフィルタ

状態空間モデルが与えられた時に、時刻 s $(s < t)$ までの観測データから時刻 t における状態変数の推定

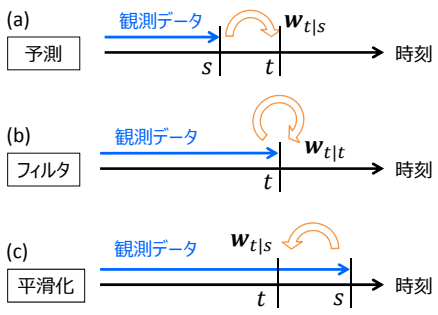

図 2.7.2 予測、フィルタ、平滑化

値を求めることを予測という（図 2.7.2(a)）。具体的には、$\boldsymbol{w}_t|\boldsymbol{y}_{1:s}$ の分布の期待値を求めることに相当する。特に、$s = t-1$ の場合を一期先予測という。同様に、時刻 t までの観測データから時刻 t の状態変数を推定することをフィルタ（図 2.7.2(b)）、時刻 s $(s > t)$ までの観測データから時刻 t の状態変数を推定することを平滑化という（図 2.7.2(c)）。ここで図にも示したように、時刻 s までの観測データから求められた時刻 t における状態変数の推定値を $\boldsymbol{w}_{t|s}$ と書く。このように書いた場合、$\boldsymbol{w}_{t|s}$ は確率変数ではなく何らかの確定した値であることに注意する。

ある時刻の状態変数の分布が分かっている場合、予測、フィルタ、平滑化の手続きを繰り返すことで、あらゆる時刻の確率分布を得ることができる。これを逐次ベイズフィルタという。カルマンフィルタは、一期先予測とフィルタを繰り返すことで現在の時刻の状態変数を推定する。

以下では一般状態空間モデルに対して、一期先予測の確率密度関数 $p(\boldsymbol{w}_t|\boldsymbol{y}_{1:t-1})$ とフィルタの確率密度関数 $p(\boldsymbol{w}_t|\boldsymbol{y}_{1:t})$ を書き下す。一期先予測の確率密度関数は、周辺化 (2.3.40) 式とマルコフ性 (2.7.7) 式を用いることで、次のように書ける。

$$\begin{aligned} &p(\boldsymbol{w}_t|\boldsymbol{y}_{1:t-1}) \\ &= \int p(\boldsymbol{w}_t|\boldsymbol{w}_{t-1}, \boldsymbol{y}_{1:t-1}) p(\boldsymbol{w}_{t-1}|\boldsymbol{y}_{1:t-1}) d\boldsymbol{w}_{t-1} \\ &= \int p(\boldsymbol{w}_t|\boldsymbol{w}_{t-1}) p(\boldsymbol{w}_{t-1}|\boldsymbol{y}_{1:t-1}) d\boldsymbol{w}_{t-1} \end{aligned} \tag{2.7.9}$$

またフィルタの確率密度関数は、ベイズの定理 (2.3.38) 式、周辺化 (2.3.40) 式とマルコフ性 (2.7.8) 式を用いることで、次のように書ける。

$$\begin{aligned} &p(\boldsymbol{w}_t|\boldsymbol{y}_{1:t}) \\ &= \frac{p(\boldsymbol{y}_t|\boldsymbol{w}_t, \boldsymbol{y}_{1:t-1}) p(\boldsymbol{w}_t|\boldsymbol{y}_{1:t-1})}{\int p(\boldsymbol{y}_t|\boldsymbol{w}_t, \boldsymbol{y}_{1:t-1}) p(\boldsymbol{w}_t|\boldsymbol{y}_{1:t-1}) d\boldsymbol{w}_t} \\ &= \frac{p(\boldsymbol{y}_t|\boldsymbol{w}_t) p(\boldsymbol{w}_t|\boldsymbol{y}_{1:t-1})}{\int p(\boldsymbol{y}_t|\boldsymbol{w}_t) p(\boldsymbol{w}_t|\boldsymbol{y}_{1:t-1}) d\boldsymbol{w}_t} \end{aligned} \tag{2.7.10}$$

一期先予測の式に含まれる $p(\boldsymbol{w}_t|\boldsymbol{w}_{t-1})$ はシステムモデルから与えられる確率密度関数を表し、$p(\boldsymbol{w}_{t-1}|\boldsymbol{y}_{1:t-1})$ は時刻 $t-1$ のフィルタ分布の確率密度関数を表している。このことは、時刻 $t-1$ のフィルタ分布が与えられれば、システムモデルを用いて時刻 t の一期先予測の確率密度関数 $p(\boldsymbol{w}_t|\boldsymbol{y}_{1:t-1})$ が得られることを表している。また、フィルタの式に含まれる $p(\boldsymbol{y}_t|\boldsymbol{w}_t)$ は観測モデルから与えられる確率密度関数を表し、$p(\boldsymbol{w}_t|\boldsymbol{y}_{1:t-1})$ は時刻 $t-1$ の一期先予測の確率密度関数を表している。このことは、時刻 $t-1$ の一期先予測の分布が与えられれば、観測モデルを用いて時刻 t のフィルタの確率密度関数 $p(\boldsymbol{w}_t|\boldsymbol{y}_{1:t})$ が得られることを表している。すなわち、ある時刻の一期先予測またはフィルタの分布が与えられれば、一期先予測とフィルタを繰り返すことで、それ以降の任意の時刻の一期先予測とフィルタの分布が得られることを意味している。

2.7.6 カルマンフィルタ

ここではカルマンフィルタに用いられる線形・ガウス状態空間モデルに対して一期先予測とフィルタを具体的に求めることで、ガイダンスにおけるカルマンフィルタの係数更新の関係式を導く。

(1) 線形・ガウス状態空間モデル

線形・ガウス状態空間モデルは以下のように表される状態空間モデルである。

$$\boldsymbol{w}_t = F_t\boldsymbol{w}_{t-1} + G_t\boldsymbol{u}_t \tag{2.7.11}$$

$$\boldsymbol{y}_t = H_t\boldsymbol{w}_t + \boldsymbol{v}_t \tag{2.7.12}$$

ここで、$\boldsymbol{w}_t$ は K 次元ベクトル、F_t は $K\times K$ 行列、$\boldsymbol{u}_t$ は L 次元ベクトル、G_t は $K\times L$ 行列、$\boldsymbol{y}_t$ と $\boldsymbol{v}_t$ は M 次元ベクトル、H_t は $M\times K$ 行列である。また、$\boldsymbol{u}_t$ と $\boldsymbol{v}_t$ は平均 0 の正規分布に従うノイズである。線形・ガウス状態空間モデルは、$\boldsymbol{w}_t$と $\boldsymbol{w}_{t-1}$、$\boldsymbol{u}_t$ の関係、および、$\boldsymbol{y}_t$と $\boldsymbol{w}_t$ の関係が線形であり、システムノイズと観測ノイズが正規分布で表される状態空間モデルである。

F_t, G_t, H_t は現象を支配する物理法則や、統計や経験、実験などに基づく関係式や仮定により決められる何らかの既知の行列である。ガイダンスの場合、係数やシステムノイズを時間変化させる法則はない、すなわち、$F_t \to I$（単位行列）、$G_t\boldsymbol{u}_t \to \boldsymbol{u}_t$ と考えて差し支えないだろう。また、目的変数はスカラーであるから、$\boldsymbol{y}_t$と $\boldsymbol{v}_t$ はスカラーになり、H_t は K 次元ベクトルになる。よってガイダンスに適用する場合には、線形・ガウス状態空間モデルは以下のようにシンプルな形で書ける。

$$\boldsymbol{w}_t = \boldsymbol{w}_{t-1} + \boldsymbol{u}_t \tag{2.7.13}$$

$$y_t = \boldsymbol{x}_t^T\boldsymbol{w}_t + v_t \tag{2.7.14}$$

ここで、$\boldsymbol{w}_t$, $\boldsymbol{x}_t$, $\boldsymbol{u}_t$ はいずれも K 次元ベクトル、$\boldsymbol{u}_t \sim N(0, U_t)$、$v_t \sim N(0, D_t)$ で、U_t は $K\times K$ の対称行列、D_t はスカラーである。

カルマンフィルタの係数更新式を導く前に、カルマンフィルタにおける仮定をまとめておく。最後の仮定については、次の (2) と (3) で述べる。

- 線形・ガウス状態空間モデル：(2.7.13), (2.7.14) 式
- $\boldsymbol{u}_t \sim N(0, U_t)$
- $v_t \sim N(0, D_t)$
- $\boldsymbol{u}_t$と $\boldsymbol{w}_t$ は独立： $p(\boldsymbol{u}_t|\boldsymbol{w}_t) = p(\boldsymbol{u}_t)$
- v_tと $\boldsymbol{w}_t$ は独立： $p(v_t|\boldsymbol{w}_t) = p(v_t)$
- マルコフ性： (2.7.7), (2.7.8) 式
- 時刻 $t-1$ の $\boldsymbol{w}$ のフィルタの分布は正規分布に従う

(2) 一期先予測

一期先予測 (2.7.9) 式をガイダンスのカルマンフィルタについて具体的に求める。まず $p(\boldsymbol{w}_t|\boldsymbol{w}_{t-1})$ については、システムノイズ $\boldsymbol{u}_t$ での周辺化 (2.3.40) 式と $\boldsymbol{u}_t$ と $\boldsymbol{w}_t$ が独立である仮定より、

$$p(\boldsymbol{w}_t|\boldsymbol{w}_{t-1}) = \int p(\boldsymbol{w}_t|\boldsymbol{w}_{t-1}, \boldsymbol{u}_t)p(\boldsymbol{u}_t)d\boldsymbol{u}_t \tag{2.7.15}$$

となる。$p(\boldsymbol{w}_t|\boldsymbol{w}_{t-1}, \boldsymbol{u}_t)$ はシステムモデル (2.7.13) 式から一意に決まるため、(2.7.3) 式と同様に、

$$p(\boldsymbol{w}_t|\boldsymbol{w}_{t-1}, \boldsymbol{u}_t) = \delta(\boldsymbol{w}_t - \boldsymbol{w}_{t-1} - \boldsymbol{u}_t) \tag{2.7.16}$$

と書ける。また、$\boldsymbol{u}_t \sim N(0, U_t)$ との仮定から、確率密度関数を具体的に書くと

$$p(\boldsymbol{u}_t) = (2\pi)^{-\frac{K}{2}}|U_t|^{-\frac{1}{2}}\exp\left[-\frac{1}{2}\boldsymbol{u}_t^T U_t^{-1}\boldsymbol{u}_t\right] \tag{2.7.17}$$

であり、これらを (2.7.15) 式に代入すると $\boldsymbol{w}_t|\boldsymbol{w}_{t-1}$ の確率密度関数は以下のようになる。

$$p(\boldsymbol{w}_t|\boldsymbol{w}_{t-1}) = (2\pi)^{-\frac{K}{2}}|U_t|^{-\frac{1}{2}}\exp\left[-\frac{1}{2}(\boldsymbol{w}_t - \boldsymbol{w}_{t-1})^T U_t^{-1}(\boldsymbol{w}_t - \boldsymbol{w}_{t-1})\right] \tag{2.7.18}$$

すなわち、

$$\boldsymbol{w}_t|\boldsymbol{w}_{t-1} \sim N(\boldsymbol{w}_{t-1}, U_t) \tag{2.7.19}$$

であり、$\boldsymbol{w}_t|\boldsymbol{w}_{t-1}$ は正規分布になる。

次に $p(\boldsymbol{w}_{t-1}|y_{1:t-1})$ を求めるのだが、これは時刻 $t-1$ におけるフィルタの分布であり、時刻 $t-1$ の時点で何らかの方法によって導かれているはずの確率密度関数である。そこで、これについては平均が $\boldsymbol{w}_{t-1|t-1}$ の K 次元ベクトル、分散共分散行列が $Q_{t-1|t-1}$ の $K\times K$ 行列の正規分布に従うと仮定する（第 2.7.6 項 (1) の最後の仮定）。つまり、

$$\boldsymbol{w}_{t-1}|\, y_{1:t-1} \sim N(\boldsymbol{w}_{t-1|t-1}, Q_{t-1|t-1}) \tag{2.7.20}$$

であると仮定する。この仮定を用いる理由は第 2.7.6 項 (3) で述べることにする。

(2.7.19) 式と (2.7.20) 式が成り立つ場合、$\boldsymbol{w}_t$ の一期先予測の分布は、

$$\boldsymbol{w}_t|\, y_{1:t-1} \sim N(\boldsymbol{w}_{t|t-1}, Q_{t|t-1}) \tag{2.7.21}$$
$$\boldsymbol{w}_{t|t-1} = \boldsymbol{w}_{t-1|t-1} \tag{2.7.22}$$
$$Q_{t|t-1} = Q_{t-1|t-1} + U_t \tag{2.7.23}$$

となり（証明は付録 2.7.A）、$\boldsymbol{w}_t$ の一期先予測の分布は平均が $\boldsymbol{w}_{t|t-1}$ で分散共分散行列が $Q_{t|t-1}$ の正規分布になることがわかる。このことを式で書くと、

$$E\left(\boldsymbol{w}_t|\, y_{1:t-1}\right) = \boldsymbol{w}_{t|t-1} \tag{2.7.24}$$
$$V\left(\boldsymbol{w}_t|\, y_{1:t-1}\right) = Q_{t|t-1} = E\left[\left(\boldsymbol{w}_t - \boldsymbol{w}_{t|t-1}\right)\left(\boldsymbol{w}_t - \boldsymbol{w}_{t|t-1}\right)^T \middle| y_{1:t-1}\right] \tag{2.7.25}$$

となる。(2.7.21)~(2.7.23) 式の結果は、時刻 $t-1$ におけるフィルタの分布が正規分布に従うという仮定を用いて導かれている。すなわち、$t-1$ のフィルタの分布が正規分布ならば、$t-1$ の一期先予測の分布もまた正規分布になる、といえる。

(3) フィルタ

フィルタの確率分布をガイダンスのカルマンフィルタについて具体的に求める。フィルタの確率分布を求めるためには、フィルタの式 (2.7.10) に含まれる $y_t|\boldsymbol{w}_t$ と $\boldsymbol{w}_t|y_{1:t-1}$ の分布が分かればよいのだが、$\boldsymbol{w}_t|y_{1:t-1}$ は (2.7.21) 式で既に求めているので、$y_t|\boldsymbol{w}_t$ の分布を求める。観測モデル (2.7.14) 式より

$$p(y_t - \boldsymbol{x}_t^T\boldsymbol{w}_t) = p(v_t) \tag{2.7.26}$$

となるので、この両辺に $\boldsymbol{w}_t$ の条件を付け、v_t と $\boldsymbol{w}_t$ は独立であるという仮定を用いると、

$$p(y_t - \boldsymbol{x}_t^T\boldsymbol{w}_t|\boldsymbol{w}_t) = p(v_t) \tag{2.7.27}$$

となる。観測ノイズ v_t の確率分布は $N(0, D_t)$ に従うと仮定しているので、

$$y_t|\boldsymbol{w}_t \sim N(\boldsymbol{x}_t^T\boldsymbol{w}_t, D_t) \tag{2.7.28}$$

となる。これと (2.7.21) 式が成り立つ場合、時刻 t におけるフィルタ分布 (2.7.10) 式は、

$$\boldsymbol{w}_t|\, y_{1:t} \sim N(\boldsymbol{w}_{t|t}, Q_{t|t}) \tag{2.7.29}$$
$$\boldsymbol{w}_{t|t} = \boldsymbol{w}_{t|t-1} + \boldsymbol{K}_t\nu_t \tag{2.7.30}$$
$$\nu_t = y_t - \boldsymbol{x}_t^T\boldsymbol{w}_{t|t-1} \tag{2.7.31}$$
$$Q_{t|t} = Q_{t|t-1} - \boldsymbol{K}_t\boldsymbol{x}_t^T Q_{t|t-1} \tag{2.7.32}$$
$$\boldsymbol{K}_t = Q_{t|t-1}\boldsymbol{x}_t\left(\boldsymbol{x}_t^T Q_{t|t-1}\boldsymbol{x}_t + D_t\right)^{-1} \tag{2.7.33}$$

となり（証明は付録 2.7.B）、$\boldsymbol{w}_t$ のフィルタ分布は平均が $\boldsymbol{w}_{t|t}$ で分散共分散行列が $Q_{t|t}$ の正規分布になることがわかる。これは時刻 t の観測値 y_t が得られたときの $\boldsymbol{w}_t$ の最適な推定値が $\boldsymbol{w}_{t|t}$ であることを表している。ν_t はイノベーションと呼ばれ、予測値と観測値の差を表す。$\boldsymbol{K}_t$ はカルマンゲインと呼ばれる K 次元ベクトルで、(2.7.30) 式から分かるように、カルマンフィルタでの状態変数更新の手続きにおいて $\boldsymbol{w}_t$ の変化率を表す量である。(2.7.24) 式, (2.7.25) 式と同様にフィルタ分布の期待値と分散共分散行列を示しておく。

$$E\left(\boldsymbol{w}_t|\, y_{1:t}\right) = \boldsymbol{w}_{t|t} \tag{2.7.34}$$
$$V\left(\boldsymbol{w}_t|\, y_{1:t}\right) = Q_{t|t} = E\left[\left(\boldsymbol{w}_t - \boldsymbol{w}_{t|t}\right)\left(\boldsymbol{w}_t - \boldsymbol{w}_{t|t}\right)^T \middle| y_{1:t}\right] \tag{2.7.35}$$

時刻 t のフィルタ分布は正規分布で表されることを示したが、これは時刻 $t-1$ の一期先予測の分布が正規分布であることから導かれた結果である。そして時刻 $t-1$ の一期先予測の分布が正規分布であることは、時刻 $t-1$ のフィルタ分布が正規分布で表されることから導かれている。すなわち、ある時刻のフィルタ分布が正規分布をしているのであれば、それ以降の全ての時刻において、一期先予測とフィルタの分布は正規分布になる。特定の時刻にこだわらないのであればフィルタ分布の初期値が正規分布に従うと仮定すればよく、これは一般的な（最小分散推定値に基づく）カルマンフィルタの導出に用いられている仮定と一致する。

一般に、分散共分散行列は半正定値である。すなわち、分散共分散行列 Q を $K \times K$ 行列、任意の K 次元ベクトルを $\boldsymbol{x}$ としたとき、$\boldsymbol{x}^T Q\boldsymbol{x} \geq 0$ であり、Q の全ての固有値は 0 以上となる。よって、(2.7.32) 式の両辺に左から $\boldsymbol{x}_t^T$、右から $\boldsymbol{x}_t$ を掛けると、

$$\begin{aligned}\boldsymbol{x}_t^T Q_{t|t}\boldsymbol{x}_t &= \left(\boldsymbol{x}_t^T - \boldsymbol{x}_t^T\boldsymbol{K}_t\boldsymbol{x}_t^T\right) Q_{t|t-1}\boldsymbol{x}_t \\ &= \left(1 - \frac{\boldsymbol{x}_t^T Q_{t|t-1}\boldsymbol{x}_t}{\boldsymbol{x}_t^T Q_{t|t-1}\boldsymbol{x}_t + D_t}\right)\boldsymbol{x}_t^T Q_{t|t-1}\boldsymbol{x}_t \\ &\leq \boldsymbol{x}_t^T Q_{t|t-1}\boldsymbol{x}_t\end{aligned} \tag{2.7.36}$$

が成り立つことから、$Q_{t|t} \preceq Q_{t|t-1}$ である。

一期先予測時の係数の誤差分散 $Q_{t|t-1}$ は前の時刻のフィルタ値 $Q_{t-1|t-1}$ にシステムノイズの分散 U_t を加えた値となっている（(2.7.23) 式）。つまり、予測によって係数の誤差分散が拡大することを示している。そこに観測値 y_t が得られることにより、フィルタ後の誤差分散 $Q_{t|t}$ は予測時 $Q_{t|t-1}$ よりも小さくなる（(2.7.36) 式）。このように、カルマンフィルタでは観測値を取り込むことで誤差分散を小さい状態に保ちながら係数を更新していることが分かる。

(4) ガイダンスにおけるカルマンフィルタのまとめ

ガイダンスにおけるカルマンフィルタの係数更新の関係式は、一期先予測の式 (2.7.22), (2.7.23) と、フィ

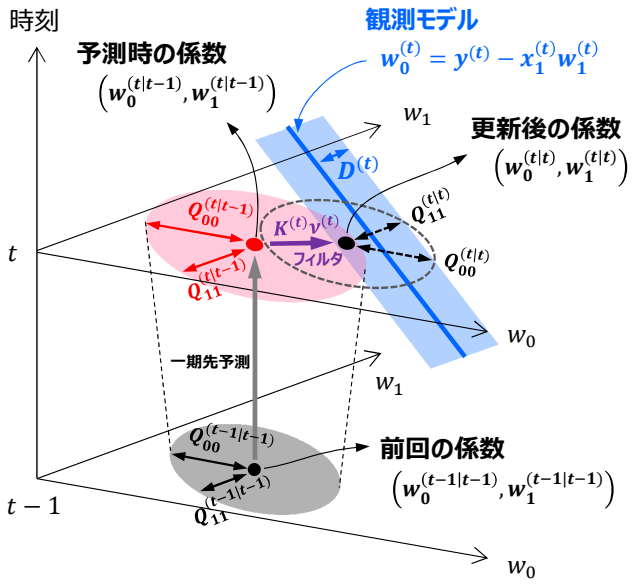

図 2.7.3　カルマンフィルタでの係数更新のイメージ。係数が 2 個の場合。時刻の添字は ( ) を付けて上付きで示している。

ルタの式 (2.7.30)〜(2.7.33) になる。また、一期先予測とフィルタの分布はともに正規分布（(2.7.21), (2.7.29) 式）になる。

実装時の参考とするために、ガイダンスにおけるカルマンフィルタの関係式を行列の成分で書いておく。添字が多くなるため、時刻は ( ) を付けて上付で示す。

$$w_i^{(t|t-1)} = w_i^{(t-1|t-1)} \tag{2.7.37}$$

$$Q_{ij}^{(t|t-1)} = Q_{ij}^{(t-1|t-1)} + U_{ij}^{(t)} \tag{2.7.38}$$

$$w_i^{(t|t)} = w_i^{(t|t-1)} + K_i^{(t)} \nu^{(t)} \tag{2.7.39}$$

$$\nu^{(t)} = y^{(t)} - \sum_{k=0}^{K} x_k^{(t)} w_k^{(t|t-1)} \tag{2.7.40}$$

$$Q_{ij}^{(t|t)} = Q_{ij}^{(t|t-1)} - K_i^{(t)} \sum_{k=0}^{K} x_k^{(t)} Q_{kj}^{(t|t-1)} \tag{2.7.41}$$

$$K_i^{(t)} = \frac{\sum_{k=0}^{K} Q_{ik}^{(t|t-1)} x_k^{(t)}}{\sum_{m,n=0}^{K} Q_{mn}^{(t|t-1)} x_m^{(t)} x_n^{(t)} + D^{(t)}} \tag{2.7.42}$$

$$x_0^{(t)} = 1, \quad i, j = 0, \cdots, K$$

係数が 2 個 (w_0, w_1) の場合のカルマンフィルタでの係数更新のイメージを図 2.7.3 に示す。ここでも添字が多くなるため、時刻は ( ) を付けて上付で示している。時刻 $t-1$ でのフィルタ分布から、係数の期待値は $\left(w_0^{(t-1|t-1)}, w_1^{(t-1|t-1)}\right)$、分散共分散行列は $Q^{(t-1|t-1)}$ であり、w_0, w_1 方向の分散は $Q_{00}^{(t-1|t-1)}, Q_{11}^{(t-1|t-1)}$ となる。(2.7.22) 式より、時刻 t での予測時には時刻 $t-1$ のフィルタの係数をそのまま利用することになるため、係数の値はフィルタの係数と同じだが、(2.7.23) 式より、その分散は時刻 $t-1$ のフィルタの値と比べてシステムノイズの分だけ大きくなる。このことは図では赤の楕円に対応しており、フィルタ分布（灰色の楕円）と比べて中心位置は変わらないが広がりは大きくなっている。ここで時刻 t の観測 $y^{(t)}$ が得られると、既に与えられている説明変数 $x_1^{(t)}$ と観測モデル (2.7.14) 式より、未知の変数である時刻 t の係数の真値 $w_0^{(t)}, w_1^{(t)}$ が $w_0^{(t)} = y^{(t)} - x_1^{(t)} w_1^{(t)}$ を中心として観測ノイズの分散 $D^{(t)}$ を持った領域（図中の青の直線および長方形）に分布していることがわかる。フィルタの式 (2.7.10) の右辺の分子は観測モデルの確率密度関数と時刻 $t-1$ の一期先予測の確率密度関数の積になっており、この 2 つの確率密度の積が最も大きくなる係数が時刻 t のフィルタ分布の期待値 $\left(w_0^{(t|t)}, w_1^{(t|t)}\right)$ であり、その分散共分散行列は $Q^{(t|t)}$ となる。このとき $D^{(t)}$ を大きな値に設定すると図中の青の長方形で示した領域が広がり、更新後の係数は予測時の係数に近い値になる。逆にシステムノイズ $U^{(t)}$ を大きな値に設定すると図中の赤の楕円で示した領域が広がり、更新後の係数は観測値に近い値になる。相対的に $D^{(t)}$ が小さい場合には、新たな観測が得られる度に観測に寄せるように係数が決まるため、係数の変動幅が大きくなる。逆に $D^{(t)}$ が大きい場合には係数の変動幅は小さくなり、予測と実況の差が大きかったとしても係数はあまり変化しなくなる。予測時の係数を更新後の係数に変化させるベクトル（図中の紫矢印）は、イノベーション $\nu^{(t)}$ にカルマンゲイン $\boldsymbol{K}^{(t)}$ を掛けたものになる。図で破線の楕円や第 2.7.6 項 (3) で示したように、観測値の情報を得ることで、フィルタ分布の分散は一期先予測の分散よりも小さくなる。以上のことを繰り返すことで、カルマンフィルタでは最新の観測値の情報を取り込みながら係数を逐次更新する。

2.7.7　整合性の確認

我々は通常、カルマンフィルタで推定される状態変数の真値を知ることはできない。よって推定値の正しさを直接的に見積もることはできないが、イノベーションを用いることでカルマンフィルタによる状態推定が正しく行われているか確認することができる。

カルマンフィルタにおいて、イノベーションは平均が 0 の正規分布に従うホワイトノイズであるという性質を持つ（例えば Brown and Rutan 1985; Jwo and Cho 2007; Bulut 2011）。すなわち、

$$E\left(\nu_t \middle| y_{1:t-1}\right) = 0 \tag{2.7.43}$$

$$E\left(\nu_t \nu_s^T \middle| y_{1:t-1}\right) = 0 \quad (t > s) \tag{2.7.44}$$

であり、その分散は

$$\begin{aligned} S_t &\equiv V\left(\nu_t \middle| y_{1:t-1}\right) = E\left(\nu_t \nu_t^T \middle| y_{1:t-1}\right) \\ &= E\left[\left(y_t - \boldsymbol{x}_t^T \boldsymbol{w}_{t|t-1}\right)\left(y_t - \boldsymbol{x}_t^T \boldsymbol{w}_{t|t-1}\right)^T \middle| y_{1:t-1}\right] \\ &= \boldsymbol{x}_t^T Q_{t|t-1} \boldsymbol{x}_t + D_t \end{aligned} \tag{2.7.45}$$

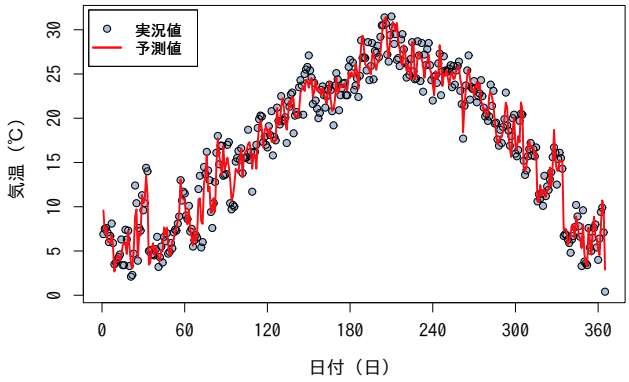

図 2.7.4　カルマンフィルタによる福岡の午前 9 時の気温予測。横軸は左端が 2014 年 1 月 1 日 00UTC 初期値、右端が 2014 年 12 月 31 日 00UTC 初期値。

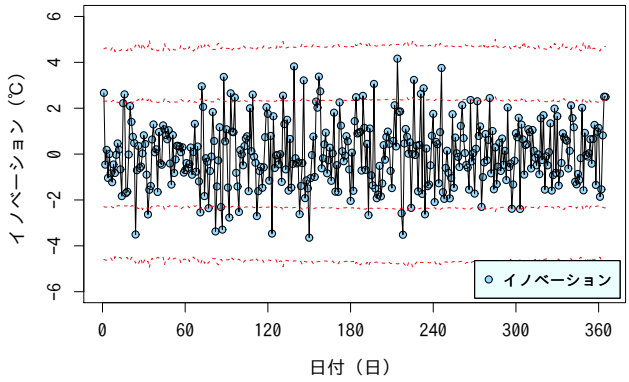

図 2.7.5　図 2.7.4 の気温予測におけるイノベーションの時系列。横軸は図 2.7.4 と同じ。赤破線は $\pm\sqrt{S_t}$ および $\pm 2\sqrt{S_t}$ を表す。

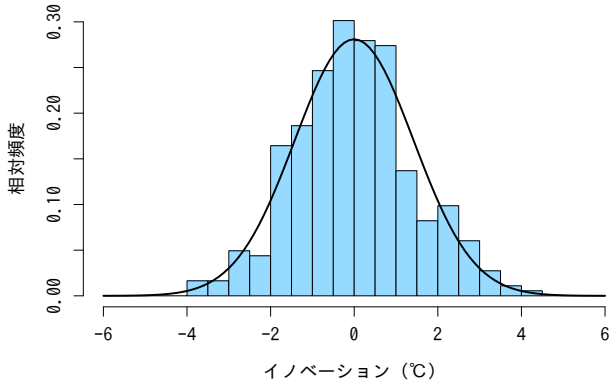

図 2.7.6　図 2.7.4 の気温予測におけるイノベーションの相対頻度のヒストグラム。実線は期間内のイノベーションから求めた平均と標準偏差をもつ正規分布。

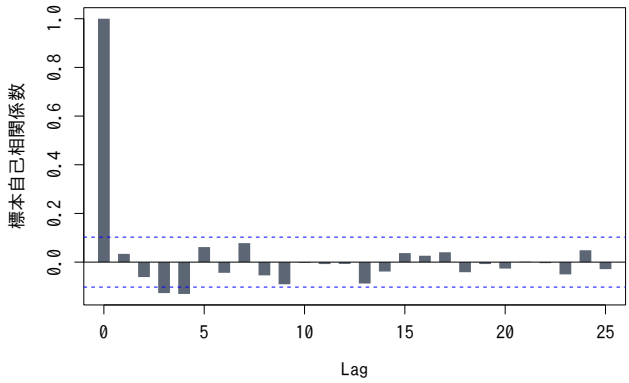

図 2.7.7　図 2.7.4 の気温予測におけるイノベーションの標本自己相関係数。横軸はラグで、青破線はホワイトノイズを仮定した場合の標本自己相関係数の 95%信頼区間（例えば Shumway and Stoffer 2000）。

と書ける。ここで (2.7.14) 式、(2.7.25) 式などを用いた。S_t はイノベーションの分散の理論値といえる。もし第 2.7.6 項 (1) で述べた仮定が満たされているならば、イノベーションの分布は (2.7.43) 式と (2.7.45) 式を満たす正規分布であり、$\pm\sqrt{S_t}$ の範囲に約 68%、$\pm 2\sqrt{S_t}$ の範囲に約 95%のイノベーションが含まれることになる。イノベーションを時系列などでプロットすることでこれらが満たされているか確認できる [2]。

図 2.7.4 は、カルマンフィルタによる福岡の午前 9 時の気温予測の例である。この例ではルーチンの気温ガイダンスと同様の説明変数と実況を用いているが、初期値や D_t, U_t の設定などは簡易的な値を用いている。学習期間は 2013 年の 1 年間とし、00UTC 初期値の 24 時間予測について、2014 年 1 月 1 日から 12 月 31 日までの 1 年間（365 日分）の予測結果を表示している。この期間の数値予報モデルの地上気温の予測は ME が −2.44 °C、RMSE が 2.93 °C であるのに対し、カルマンフィルタの予測は ME が −0.00 °C、RMSE が 1.42 °C であり、数値予報モデルを大幅に改善している。図を見る限りでも実況との対応は概ね良さそうであり、カルマンフィルタによる予測は適切に行われているように思われる。

図 2.7.4 と同じ気温予測の時系列について、イノベーションの時系列、相対頻度のヒストグラム、標本自己相関係数を図 2.7.5〜 図 2.7.7 に示す。イノベーションの時系列を見ると、日付によらず 0 を中心として同程度のばらつきを持っていることがわかる。またヒストグラムより、平均が 0 の正規分布に近い分布をしており、標本自己相関係数からは時間方向の相関が弱いことがわかる。これらのことから、イノベーションの性質のうち、平均 0 の正規分布に従うホワイトノイズであるということについては満たされていそうである。一方イノベーションのばらつきを S_t と比較すると、期待されるよりも狭い範囲にイノベーションが集中しており、イノベーションの分散が過小であるといえる。

イノベーションの分散は直接的には D_t と U_t の大きさによって変化するため、分散が過小または過大であ

[2] より厳密には、標準化 2 乗イノベーション $\nu_t^T S_t^{-1} \nu_t$ の移動平均が自由度 m（ガイダンスの場合は目的変数がスカラーであるため $m = 1$）の χ^2 分布に従うことを確かめればよい。

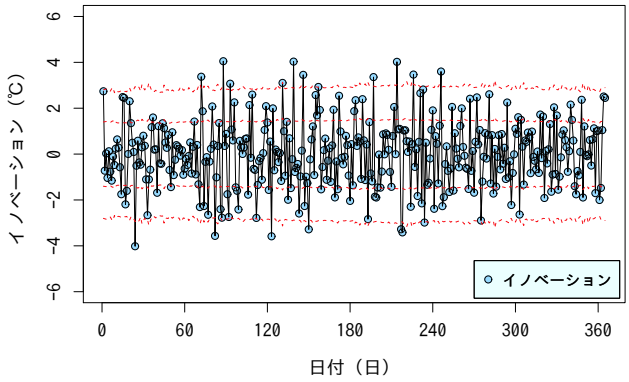

図 2.7.8　図 2.7.5 と同じ。ただし、D_t を調整してイノベーションのばらつきを調整した結果。

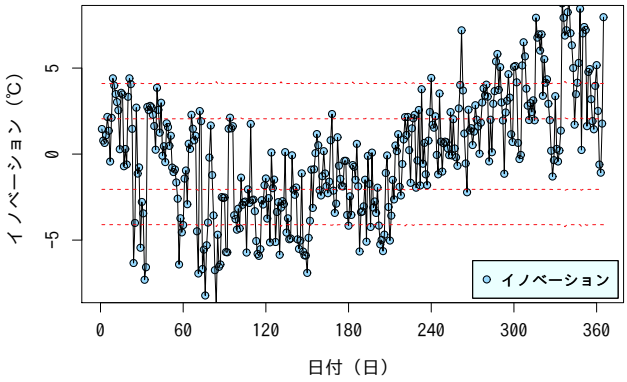

図 2.7.9　不適切なイノベーションの例。図の見方は図 2.7.5 や図 2.7.8 と同じ。

る場合には D_t や U_t を調整すれば適切なばらつきを持つように修正することはできる。しかし、目的変数と説明変数の関係が線形になっていない場合やノイズが正規分布ではない場合など、カルマンフィルタの仮定がそもそも満たされていないならば、D_t や U_t だけを調整することは予測精度の低下につながる場合がある。図 2.7.8 は上記の気温予測について、D_t を調整してイノベーションのばらつきが適切に近くなるようにした場合の実験結果である。この図を見ると、$\pm\sqrt{S_t}$ および $\pm 2\sqrt{S_t}$ の範囲に適切に近い数のイノベーションが含まれていることがわかる。また図には示さないが、予測値の時系列、イノベーションの相対頻度と標本自己相関には不適切な点は見られない。しかし予測精度は、ME が 0.001 °C、RMSE が 1.46 °C となっており、調整前と比べてやや低下している。

最後に不適切なイノベーションの分布の例を図 2.7.9 に示す。この図は図 2.7.4 と同じ気温ガイダンスに用いられている説明変数のうち、ガイダンスへの寄与量が小さい（すなわちあまり重要ではない）説明変数を 1 つだけ用いて気温を予測した場合のイノベーションの時系列である。このイノベーションは 0 を中心とした分布をしておらず、ばらつきの大きさは理論値よりも大きくなっている。また、時間方向に相関を持つことからホワイトノイズではなく、ばらつきの大きさとしても時間によって差が見られる。よってこの例では、イノベーションの性質をいずれも満たしていない。しかしながら図 2.7.4 で示したように、このような説明変数を用いているにもかかわらず、カルマンフィルタとしては一見すると正常な動作をしているように見え、また、この説明変数を除いた場合には予測精度としては低下してしまう。このことは、この説明変数の見直しや、他の説明変数も含めた全体的な見直しが必要であることを示唆している。

2.7.8　パラメータの調整

ガイダンスのカルマンフィルタにおいて、D_t, U_t および係数と $Q_{t|t}$ の初期値は既知のものとして扱われており、開発者が事前に設定しなければならないパラメータである。これらのうち係数の初期値は、予測精度に大きな影響は与えないため [3] 乱数や 0 を与えても問題ない。また $Q_{t|t}$ の初期値についても、乱数を与えても経験的には適切な値に収束するため大きな問題はない。一方、D_t と U_t は係数の変動幅を決める重要なパラメータであるため、乱数などを与えることは不適切であり、何らかの方法で適切な値を与える必要がある。D_t は観測ノイズの分散であるから、過去の一定期間のデータから算出した予測の平均二乗誤差を元に大まかな値を見積もることができる。また U_t はシステムノイズの分散共分散行列であるから、(2.7.13) 式より、今回と前回の係数の差の二乗和から大まかな値を見積もることができる。気象庁のガイダンスでは、D_t と U_t の非対角成分は 0 と仮定し、上記の見積りを元に、学習データに対して繰り返し計算しながら D_t と U_t（時間変化させる場合にはその初期値）を決定している。

より効率的に D_t と U_t を見積もる方法として、最尤法に基づく手法 (Shumway and Stoffer 2000) と EM アルゴリズム (Dempster et al. 1977; Shumway and Stoffer 1982; Ghahramani and Hinton 1996; Shumway and Stoffer 2000) が提案されている。これらの手法はいずれも、繰り返し計算によってパラメータを見積もっている。ここでは D_t, U_t などのパラメータをまとめて θ と置く。以下では θ は時間変化しないものとして、最尤法に基づいたパラメータの推定方法を示す。

イノベーション ν_t は平均 0、分散 S_t の正規分布に従うことから、イノベーションの確率密度関数は

$$p(\nu_t) = \frac{1}{\sqrt{2\pi S_t}} \exp\left(-\frac{\nu_t^2}{2S_t}\right) \tag{2.7.46}$$

[3] 例えば (瀬上ほか 1995) の第 5.1.3 項での理想実験を参照。係数を推定する手法がカルマンフィルタなのであるから当然のこととともいえる。

である。今、時刻 $t = 1 \sim N$ のデータが与えられているとする。イノベーションの負の対数尤度を誤差関数 E とすると、E は定数項を除いて

$$E \equiv -\ln L = \frac{1}{2}\sum_{t=1}^{N}\left(\ln S_t + \frac{\nu_t^2}{S_t}\right) \tag{2.7.47}$$

となる。ここで S_tやν_tはθ の関数である。$\theta \to \pm\Delta\theta$ とした場合の誤差関数の値を $E_\pm$ とすると、θ 方向の E の微分は

$$\frac{\partial E}{\partial \theta} \simeq \frac{E_+ - E_-}{2\Delta\theta} \tag{2.7.48}$$

となる。同様にして θ の 2 階微分も得られる。θ の更新は以下の手順で行う。

1. θ の初期値を与える。
2. 学習期間のデータ（$t = 1 \sim N$）に対してカルマンフィルタによる係数更新を行い、ν_tと S_t の時系列を求める。
3. ν_tと S_t および E の微分を用いてニュートン・ラフソン法などの繰り返し計算を 1 回分だけ行い、θ を更新する。
4. 上記の 2 ～ 3 の手順を繰り返し、誤差関数が変化しなくなったとみなせるところで計算を終了する。

見積もるパラメータが多いことから、実際に計算する場合には予測への影響が小さいパラメータ（係数や $Q_{t|t}$ の初期値など）には乱数や 0 を与えることとし、その他のパラメータについて上記の方法で見積もることになるだろう。

上記の手法では U_tと D_t を固定値として扱っているが、D_t は観測モデルが不完全であることも含めた誤差を表しており、予測が難しい場合には D_t を大きく、簡単な場合には小さくすることが望ましい。しかし予測の難しさを事前に知ることは困難である [4] ため、本稿を執筆している 2018 年現在のガイダンスでは D_t は固定値か、時間変化が緩やかな固定値に近い値として扱われている。例えば平均降水量ガイダンスでは、前回までの D_t と今回の予測の二乗誤差を重み付け平均することで D_t を更新しているが、前回までの D_t の重みを非常に大きくしており、場の状況に応じて変化させているというよりは長期的な予測誤差の変化に対応させることを目的としている。このような事情により、ガイダンスの予測が大きく外れた場合には、実況に合わせるために係数を大きく変化させてしまうことがある。しかし係数の大きな変化は、その後の予測で極端な値を予測してしまうなど、一般に予測精度を低下させる要因となる。そこで気温ガイダンスでは、D_t は固定値としているが、予測が大きく外れた場合には、係数更新時に一時的に D_t を大きくして係数を実況に寄せすぎない（大きく変化させない）ようにしている。

[4] アンサンブル予報のスプレッドが適切である場合、スプレッドを用いれば D_t を事前に見積もることができるかもしれない。

2.7.9 利用上の注意点

カルマンフィルタは線形・ガウス状態空間モデルに基づいて係数を更新しており、第 2.7.6 項 (1) で述べた仮定が満たされている必要がある。カルマンフィルタでは係数が変動するため、特にシステムモデルについては線形性やノイズの正規性を直接的に確認することは難しいが、観測モデルについては各目的変数と説明変数の関係をプロットすることで線形性やノイズの正規性をある程度確認することができる。新規に説明変数を追加する場合やガイダンスの改良を行う場合には、目的変数と説明変数の関係をプロットして線形性や正規性を確認するとよい。

第 2.7.7 項で述べたように、イノベーションの分布を時系列で見ることによって、カルマンフィルタによる係数更新が適切に行われているか否かを調べることができる。イノベーションの性質が満たされていない場合には、D_t, U_t および係数と $Q_{t|t}$ の初期値の設定や、目的変数と説明変数の関係が適切であるかを確認する。第 2.7.7 項で示したように、上記のどれか一つを調整しただけでは全体としてはバランスが崩れて予測精度が低下することもある。このような場合には全体的な調整をすることが精度向上につながるだろう。

カルマンフィルタは係数の最適な推定値を逐次更新するため、数値予報モデルが更新されて予測特性が大きく変化した場合でも一定期間学習すれば適切な係数が自動的に得られると期待される。しかし、一定期間がどの程度かは対象としている現象の頻度に依存する。気温ガイダンスの場合には 2,3 週間程度の学習で適切に近い値まで学習が進むが、強風や大雨など頻度が少ない現象を対象としたガイダンスでは数か月以上掛かる。このためカルマンフィルタを用いたガイダンスであっても、数値予報モデルの特性が大きく変化する場合には、長期間の過去データを用いた事前学習が必要になる場合がある。

カルマンフィルタの仮定が満たされていないような目的変数と説明変数に対しても、カルマンフィルタはある程度適切な予測結果を与える。これは線形重回帰やロジスティック回帰でも同様で、各手法で仮定している条件が成り立たない場合であっても、これらの統計手法はある程度適切な（精度を持つ）予測結果を与える。一括学習型の統計手法を用いた場合には、悪い部分も含めて予測特性は変化しないため、値が異常であることや不適切であることは判別しやすい。これに対してカルマンフィルタのような逐次学習型の統計手法の場合、手法の仮定が満たされていない状況では、ある時に突然異常な係数を学習してしまう可能性がある。そして予測特性が常に変化するという特性は、その変化が正常の範囲内であるのか異常であるのかを判断することを困難にしてしまう。カルマンフィルタを用いたガイダンスを開発・運用するに当たっては、カルマンフィルタの仮定との整合性や係数の日々の変化

に注意する必要がある。

2.7.10 まとめ

本節では、カルマンフィルタを用いてガイダンスを算出する場合の関係式をベイズ推定に基づく係数の条件付き期待値から導出した。また、イノベーションの時系列を用いた整合性の確認手法と、ガイダンスを開発する上で必要となるパラメータの調整について述べた。カルマンフィルタを用いることで、係数の特性が時間変化する場合にも係数が逐次調整されていくことから、2018 年現在では様々なガイダンスにカルマンフィルタが利用されている。一方で、ガイダンスの開発・運用においては注意すべき点も多い。目的変数と説明変数の特性や関係性を把握しながら開発を進めることが重要である。

参考文献

Brown, S. D. and S. C. Rutan, 1985: Adaptive Kalman filtering. *Journal of Research of the National Bureau of Standards*, **90**, 403–407.

Bulut, Y., 2011: Applied kalman filter theory. *Civil Engineering Dissertations*, **13**.

Dempster, A. P., N. M. Laird, and D. B. Rubin, 1977: Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B*, **39(1)**, 1–38.

Ghahramani, Z. and G. E. Hinton, 1996: Parameter estimation for linear dynamical systems. *Technical Report CRG-TR-96-2*, 1–6.

樋口知之, 2011: データ同化入門. 朝倉書店, 240 pp.

堀田大介, 太田洋一郎, 2011: ローレンツモデルによる学習. 数値予報課報告・別冊第 57 号, 気象庁予報部, 144–158.

Jwo, D. and T. Cho, 2007: A practical note on evaluating Kalman filter performance optimality and degradation. *Applied Mathematics and Computation*, **193**, 482–505.

片山徹, 2000: 新版 応用カルマンフィルタ. 朝倉書店, 255 pp.

北川源四郎, 1993: FORTRAN77 時系列解析プログラミング. 岩波書店, 390 pp.

Persson, A., 1989: Kalman filtering - a new approach to adaptive statistical interpretation of numerical meteorological forecasts. *ECMWF Newsletter*, **46**, 16–20.

Persson, A., 1991: Kalman filtering - a new approach to adaptive statistical interpretation of numerical meteorological forecasts. *WMO/TD*, **421**, XX27–XX32.

瀬上哲秀, 大林正典, 國次雅司, 藤田司, 1995: カルマンフィルター. 平成 7 年度数値予報研修テキスト, 気象庁予報部, 66–78.

Shumway, R. H. and D. S. Stoffer, 1982: An approach to time series smoothing and forecasting using the EM algorithm. *Journal of Time Series Analysis*, **3**, 253–264.

Shumway, R. H. and D. S. Stoffer, 2000: *Time series analysis and its applications*. Springer, 549 pp.

Simonsen, C., 1991: Self adaptive model output statistics based on kalman filtering. *WMO/TD*, **421**, XX33–XX37.

付録 2.7.A　一期先予測の関係式の導出

ここでは、$\boldsymbol{w}_{t-1}|y_{1:t-1} \sim N(\boldsymbol{w}_{t-1|t-1}, Q_{t-1|t-1})$ および $\boldsymbol{w}_t|\boldsymbol{w}_{t-1} \sim N(\boldsymbol{w}_{t-1}, U_t)$ のとき、ガイダンスのカルマンフィルタにおいて、下記の一期先予測の関係式が成り立つことを示す。

$$\boldsymbol{w}_t|y_{1:t-1} \sim N(\boldsymbol{w}_{t|t-1}, Q_{t|t-1}) \tag{2.7.49}$$

$$\boldsymbol{w}_{t|t-1} = \boldsymbol{w}_{t-1|t-1} \tag{2.7.50}$$

$$Q_{t|t-1} = Q_{t-1|t-1} + U_t \tag{2.7.51}$$

以下では表記を簡単にするため、一時的に $\boldsymbol{w}_{t-1} \to \boldsymbol{v}$, $\boldsymbol{w}_t \to \boldsymbol{w}$, $\boldsymbol{w}_{t-1|t-1} \to \hat{\boldsymbol{v}}$, $y_{1:t-1} \to y$, $Q_{t-1|t-1} \to Q$, $U_t \to U$ と記述する。y はスカラー、$\boldsymbol{v}$ と $\boldsymbol{w}$ は K 次元ベクトル、Q と U は $K \times K$ 対称行列である。

与えられた条件より、

$$p(\boldsymbol{v}|y) = (2\pi)^{-\frac{K}{2}}|Q|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\boldsymbol{v}-\hat{\boldsymbol{v}})^T Q^{-1}(\boldsymbol{v}-\hat{\boldsymbol{v}})\right] \tag{2.7.52}$$

$$p(\boldsymbol{w}|\boldsymbol{v}) = (2\pi)^{-\frac{K}{2}}|U|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\boldsymbol{w}-\boldsymbol{v})^T U^{-1}(\boldsymbol{w}-\boldsymbol{v})\right] \tag{2.7.53}$$

である。$\boldsymbol{w}|y$ の確率密度関数は、周辺化 (2.3.40) 式とマルコフ性 (2.7.7) 式より、

$$p(\boldsymbol{w}|y) = \int p(\boldsymbol{w}|\boldsymbol{v}, y)p(\boldsymbol{v}|y)d\boldsymbol{v} = \int p(\boldsymbol{w}|\boldsymbol{v})p(\boldsymbol{v}|y)d\boldsymbol{v} \tag{2.7.54}$$

と書けるので、右辺の被積分関数は以下のようになる。

$$p(\boldsymbol{w}|\boldsymbol{v})p(\boldsymbol{v}|y) = (2\pi)^{-K}|Q|^{-\frac{1}{2}}|U|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}\left\{(\boldsymbol{v}-\hat{\boldsymbol{v}})^T Q^{-1}(\boldsymbol{v}-\hat{\boldsymbol{v}}) + (\boldsymbol{w}-\boldsymbol{v})^T U^{-1}(\boldsymbol{w}-\boldsymbol{v})\right\}\right] \tag{2.7.55}$$

これを $\boldsymbol{v}$ で積分することから、上式の右辺を $\boldsymbol{v}$ の項とそれ以外の項に分けることを考える。(2.7.55) 式の右辺の { } 内を展開して整理すると、

$$\begin{aligned}\{\ \} &= \boldsymbol{v}^T Q^{-1}\boldsymbol{v} - \boldsymbol{v}^T Q^{-1}\hat{\boldsymbol{v}} - \hat{\boldsymbol{v}}^T Q^{-1}\boldsymbol{v} + \hat{\boldsymbol{v}}^T Q^{-1}\hat{\boldsymbol{v}} + \boldsymbol{w}^T U^{-1}\boldsymbol{w} - \boldsymbol{w}^T U^{-1}\boldsymbol{v} - \boldsymbol{v}^T U^{-1}\boldsymbol{w} + \boldsymbol{v}^T U^{-1}\boldsymbol{v} \\ &= \boldsymbol{v}^T\left(Q^{-1}+U^{-1}\right)\boldsymbol{v} - \boldsymbol{v}^T\left(Q^{-1}\hat{\boldsymbol{v}} + U^{-1}\boldsymbol{w}\right) - \left(\hat{\boldsymbol{v}}^T Q^{-1} + \boldsymbol{w}^T U^{-1}\right)\boldsymbol{v} + \hat{\boldsymbol{v}}^T Q^{-1}\hat{\boldsymbol{v}} + \boldsymbol{w}^T U^{-1}\boldsymbol{w}\end{aligned} \tag{2.7.56}$$

ここで、ある K 次元ベクトル $\boldsymbol{\xi}$ を用いて、

$$(\boldsymbol{v}-\boldsymbol{\xi})^T\left(Q^{-1}+U^{-1}\right)(\boldsymbol{v}-\boldsymbol{\xi}) = \boldsymbol{v}^T\left(Q^{-1}+U^{-1}\right)\boldsymbol{v} - \boldsymbol{v}^T\left(Q^{-1}+U^{-1}\right)\boldsymbol{\xi} - \boldsymbol{\xi}^T\left(Q^{-1}+U^{-1}\right)\boldsymbol{v} + \boldsymbol{\xi}^T\left(Q^{-1}+U^{-1}\right)\boldsymbol{\xi} \tag{2.7.57}$$

と書けることから、

$$\boldsymbol{\xi} \equiv \left(Q^{-1}+U^{-1}\right)^{-1}\left(Q^{-1}\hat{\boldsymbol{v}} + U^{-1}\boldsymbol{w}\right) \tag{2.7.58}$$

とすると、Q と U が対称行列であることから、$\boldsymbol{\xi}^T$ は以下のようになる。

$$\boldsymbol{\xi}^T = \left(Q^{-1}\hat{\boldsymbol{v}} + U^{-1}\boldsymbol{w}\right)^T\left\{\left(Q^{-1}+U^{-1}\right)^T\right\}^{-1} = \left(\hat{\boldsymbol{v}}^T Q^{-1} + \boldsymbol{w}^T U^{-1}\right)\left(Q^{-1}+U^{-1}\right)^{-1} \tag{2.7.59}$$

この $\boldsymbol{\xi}$ を用いると、(2.7.56) 式は、

$$\{\ \} = (\boldsymbol{v}-\boldsymbol{\xi})^T\left(Q^{-1}+U^{-1}\right)(\boldsymbol{v}-\boldsymbol{\xi}) - \boldsymbol{\xi}^T\left(Q^{-1}+U^{-1}\right)\boldsymbol{\xi} + \hat{\boldsymbol{v}}^T Q^{-1}\hat{\boldsymbol{v}} + \boldsymbol{w}^T U^{-1}\boldsymbol{w} \tag{2.7.60}$$

となり、被積分関数を $\boldsymbol{v}$ に関する項（第 1 項）とそれ以外の項に分けることができた。ここで (2.7.60) 式の右辺第 2 項を展開すると、

$$\begin{aligned}-\boldsymbol{\xi}^T\left(Q^{-1}+U^{-1}\right)\boldsymbol{\xi} &= -\left(\hat{\boldsymbol{v}}^T Q^{-1} + \boldsymbol{w}^T U^{-1}\right)\left(Q^{-1}+U^{-1}\right)^{-1}\left(Q^{-1}\hat{\boldsymbol{v}} + U^{-1}\boldsymbol{w}\right) \\ &= -\hat{\boldsymbol{v}}^T Q^{-1}\left(Q^{-1}+U^{-1}\right)^{-1}Q^{-1}\hat{\boldsymbol{v}} - \hat{\boldsymbol{v}}^T Q^{-1}\left(Q^{-1}+U^{-1}\right)^{-1}U^{-1}\boldsymbol{w} \\ &\quad - \boldsymbol{w}^T U^{-1}\left(Q^{-1}+U^{-1}\right)^{-1}Q^{-1}\hat{\boldsymbol{v}} - \boldsymbol{w}^T U^{-1}\left(Q^{-1}+U^{-1}\right)^{-1}U^{-1}\boldsymbol{w}\end{aligned} \tag{2.7.61}$$

であり、

$$U^{-1}\left(Q^{-1}+U^{-1}\right)^{-1}Q^{-1}=\left[Q\left(Q^{-1}+U^{-1}\right)U\right]^{-1}=(Q+U)^{-1} \tag{2.7.62}$$

$$Q^{-1}\left(Q^{-1}+U^{-1}\right)^{-1}U^{-1}=\left[U\left(Q^{-1}+U^{-1}\right)Q\right]^{-1}=(Q+U)^{-1} \tag{2.7.63}$$

$$\begin{aligned}Q^{-1}\left(Q^{-1}+U^{-1}\right)^{-1}Q^{-1}&=\left(Q^{-1}+U^{-1}\right)\left(Q^{-1}+U^{-1}\right)^{-1}Q^{-1}-U^{-1}\left(Q^{-1}+U^{-1}\right)^{-1}Q^{-1}\\&=Q^{-1}-(Q+U)^{-1}\end{aligned} \tag{2.7.64}$$

$$U^{-1}\left(Q^{-1}+U^{-1}\right)^{-1}U^{-1}=U^{-1}-(Q+U)^{-1} \tag{2.7.65}$$

であることを用いると、(2.7.60) 式の右辺第 2 項は、

$$\begin{aligned}&-\boldsymbol{\xi}^T\left(Q^{-1}+U^{-1}\right)\boldsymbol{\xi}\\&=-\hat{\boldsymbol{v}}^TQ^{-1}\hat{\boldsymbol{v}}+\hat{\boldsymbol{v}}^T(Q+U)^{-1}\hat{\boldsymbol{v}}-\hat{\boldsymbol{v}}^T(Q+U)^{-1}\boldsymbol{w}-\boldsymbol{w}^T(Q+U)^{-1}\hat{\boldsymbol{v}}-\boldsymbol{w}^TU^{-1}\boldsymbol{w}+\boldsymbol{w}^T(Q+U)^{-1}\boldsymbol{w}\end{aligned} \tag{2.7.66}$$

となる。よって (2.7.60) 式の右辺第 2, 3, 4 項を Δ と置くと、(2.7.60) 式の右辺第 3, 4 項と (2.7.66) 式の右辺第 1, 5 項が打ち消し合うため、

$$\begin{aligned}\Delta&\equiv\hat{\boldsymbol{v}}^T(Q+U)^{-1}\hat{\boldsymbol{v}}-\hat{\boldsymbol{v}}^T(Q+U)^{-1}\boldsymbol{w}-\boldsymbol{w}^T(Q+U)^{-1}\hat{\boldsymbol{v}}+\boldsymbol{w}^T(Q+U)^{-1}\boldsymbol{w}\\&=(\boldsymbol{w}-\hat{\boldsymbol{v}})^T(Q+U)^{-1}(\boldsymbol{w}-\hat{\boldsymbol{v}})\end{aligned} \tag{2.7.67}$$

となる。これで準備ができたので (2.7.54) 式の計算に戻る。

$$\begin{aligned}p(\boldsymbol{w}|y)&=(2\pi)^{-K}|Q|^{-\frac{1}{2}}|U|^{-\frac{1}{2}}\int\exp\left[-\frac{1}{2}\left\{(\boldsymbol{v}-\hat{\boldsymbol{v}})^TQ^{-1}(\boldsymbol{v}-\hat{\boldsymbol{v}})+(\boldsymbol{w}-\boldsymbol{v})^TU^{-1}(\boldsymbol{w}-\boldsymbol{v})\right\}\right]d\boldsymbol{v}\\&=(2\pi)^{-K}|Q|^{-\frac{1}{2}}|U|^{-\frac{1}{2}}\exp\left[-\frac{1}{2}(\boldsymbol{w}-\hat{\boldsymbol{v}})^T(Q+U)^{-1}(\boldsymbol{w}-\hat{\boldsymbol{v}})\right]\int\exp\left[-\frac{1}{2}(\boldsymbol{v}-\boldsymbol{\xi})^T\left(Q^{-1}+U^{-1}\right)(\boldsymbol{v}-\boldsymbol{\xi})\right]d\boldsymbol{v}\\&=(2\pi)^{-\frac{K}{2}}|Q|^{-\frac{1}{2}}\left|Q^{-1}+U^{-1}\right|^{-\frac{1}{2}}|U|^{-\frac{1}{2}}\exp\left[-\frac{1}{2}(\boldsymbol{w}-\hat{\boldsymbol{v}})^T(Q+U)^{-1}(\boldsymbol{w}-\hat{\boldsymbol{v}})\right]\end{aligned} \tag{2.7.68}$$

ここで正規分布の積分 (2.3.43) 式を用いた。Q と U はともに $K\times K$ 対称行列であり、$|QU|=|Q||U|$ となることを用いると、

$$|Q|^{-\frac{1}{2}}\left|Q^{-1}+U^{-1}\right|^{-\frac{1}{2}}|U|^{-\frac{1}{2}}=|Q+U|^{-\frac{1}{2}} \tag{2.7.69}$$

であるから、

$$p(\boldsymbol{w}|y)=(2\pi)^{-\frac{K}{2}}|Q+U|^{-\frac{1}{2}}\exp\left[-\frac{1}{2}(\boldsymbol{w}-\hat{\boldsymbol{v}})^T(Q+U)^{-1}(\boldsymbol{w}-\hat{\boldsymbol{v}})\right] \tag{2.7.70}$$

となり、$\boldsymbol{w}|y\sim N(\hat{\boldsymbol{v}},Q+U)$ となる。元の表記で書けば、$\boldsymbol{w}_t|y_{1:t-1}\sim N(\boldsymbol{w}_{t-1|t-1},Q_{t-1|t-1}+U_t)$ である。ここで $\boldsymbol{w}_t|y_{1:t-1}$ は時刻 $t-1$ の一期先予測の分布であるから、その期待値と分散共分散行列は $\boldsymbol{w}_{t|t-1}$と $Q_{t|t-1}$ であるので、

$$\boldsymbol{w}_{t|t-1}\equiv\boldsymbol{w}_{t-1|t-1} \tag{2.7.71}$$

$$Q_{t|t-1}\equiv Q_{t-1|t-1}+U_t \tag{2.7.72}$$

とすると、

$$\boldsymbol{w}_t|y_{1:t-1}\sim N(\boldsymbol{w}_{t|t-1},Q_{t|t-1}) \tag{2.7.73}$$

が得られる。

付録 2.7.B　フィルタの関係式の導出

ここでは、$y_t|\boldsymbol{w}_t \sim N(\boldsymbol{x}_t^T\boldsymbol{w}_t, D_t)$ および $\boldsymbol{w}_t|y_{1:t-1} \sim N(\boldsymbol{w}_{t|t-1}, Q_{t|t-1})$ のとき、フィルタ分布の式 (2.7.10) を解くことで、ガイダンスのカルマンフィルタにおいて、下記のフィルタの関係式が成り立つことを示す。

$$\boldsymbol{w}_t|y_{1:t} \sim N(\boldsymbol{w}_{t|t}, Q_{t|t}) \tag{2.7.74}$$
$$\boldsymbol{w}_{t|t} = \boldsymbol{w}_{t|t-1} + \boldsymbol{K}_t\left(y_t - \boldsymbol{x}_t^T\boldsymbol{w}_{t|t-1}\right) \tag{2.7.75}$$
$$Q_{t|t} = Q_{t|t-1} - \boldsymbol{K}_t\boldsymbol{x}_t^T Q_{t|t-1} \tag{2.7.76}$$
$$\boldsymbol{K}_t = Q_{t|t-1}\boldsymbol{x}_t\left(\boldsymbol{x}_t^T Q_{t|t-1}\boldsymbol{x}_t + D_t\right)^{-1} \tag{2.7.77}$$

以下では表記を簡単にするため、一時的に $\boldsymbol{w}_t \to \boldsymbol{w}$, $\boldsymbol{w}_{t|t-1} \to \hat{\boldsymbol{w}}$, $Q_{t|t-1} \to Q$, $y_t \to y$, $y_{1:t-1} \to z$, $\boldsymbol{x}_t \to \boldsymbol{x}$, $D_t \to D$ と記述する。y, z, D はスカラー、$\boldsymbol{w}$, $\hat{\boldsymbol{w}}$, $\boldsymbol{x}$ は K 次元ベクトル、Q は $K \times K$ 対称行列である。

与えられた条件より、

$$p(y|\boldsymbol{w}) = (2\pi)^{-\frac{1}{2}}|D|^{-\frac{1}{2}}\exp\left[-\frac{1}{2}(y-\boldsymbol{x}^T\boldsymbol{w})^T D^{-1}(y-\boldsymbol{x}^T\boldsymbol{w})\right] \tag{2.7.78}$$
$$p(\boldsymbol{w}|z) = (2\pi)^{-\frac{K}{2}}|Q|^{-\frac{1}{2}}\exp\left[-\frac{1}{2}(\boldsymbol{w}-\hat{\boldsymbol{w}})^T Q^{-1}(\boldsymbol{w}-\hat{\boldsymbol{w}})\right] \tag{2.7.79}$$

と書ける。(2.7.78) 式は 1 変数の正規分布の確率密度関数であるためもっとシンプルに書けるが、ここでは (2.7.79) 式との対称性を保持するため、1×1 行列として扱うことにする。フィルタ分布の式 (2.7.10) の分母の被積分関数と分子はこの 2 つの確率密度関数を掛けた関数である。

$$p(y|\boldsymbol{w})p(\boldsymbol{w}|z) = (2\pi)^{-\frac{K+1}{2}}|D|^{-\frac{1}{2}}|Q|^{-\frac{1}{2}}\exp\left[-\frac{1}{2}\left\{(y-\boldsymbol{x}^T\boldsymbol{w})^T D^{-1}(y-\boldsymbol{x}^T\boldsymbol{w}) + (\boldsymbol{w}-\hat{\boldsymbol{w}})^T Q^{-1}(\boldsymbol{w}-\hat{\boldsymbol{w}})\right\}\right] \tag{2.7.80}$$

(2.7.10) 式では $\boldsymbol{w}$ で積分することから、右辺の { } 内を $\boldsymbol{w}$ の項とそれ以外の項に分けることを考える。{ } 内を展開して整理すると、

$$\begin{aligned}
\{\,\} &= \boldsymbol{w}^T Q^{-1}\boldsymbol{w} - \boldsymbol{w}^T Q^{-1}\hat{\boldsymbol{w}} - \hat{\boldsymbol{w}}^T Q^{-1}\boldsymbol{w} + \hat{\boldsymbol{w}}^T Q^{-1}\hat{\boldsymbol{w}} + y^2 D^{-1} - yD^{-1}\boldsymbol{x}^T\boldsymbol{w} - \boldsymbol{w}^T\boldsymbol{x}D^{-1}y + \boldsymbol{w}^T\boldsymbol{x}D^{-1}\boldsymbol{x}^T\boldsymbol{w} \\
&= \boldsymbol{w}^T\left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)\boldsymbol{w} - \boldsymbol{w}^T\left(Q^{-1}\hat{\boldsymbol{w}} + \boldsymbol{x}D^{-1}y\right) - \left(\hat{\boldsymbol{w}}^T Q^{-1} + yD^{-1}\boldsymbol{x}^T\right)\boldsymbol{w} + \hat{\boldsymbol{w}}^T Q^{-1}\hat{\boldsymbol{w}} + y^2 D^{-1} \\
&= (\boldsymbol{w}-\boldsymbol{\xi})^T\left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)(\boldsymbol{w}-\boldsymbol{\xi}) + \hat{\boldsymbol{w}}^T Q^{-1}\hat{\boldsymbol{w}} + y^2 D^{-1} - \boldsymbol{\xi}^T\left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)\boldsymbol{\xi} \\
&= (\boldsymbol{w}-\boldsymbol{\xi})^T R^{-1}(\boldsymbol{w}-\boldsymbol{\xi}) + \Delta
\end{aligned} \tag{2.7.81}$$

となり、$\boldsymbol{w}$ の項とそれ以外の項に分けることができた。ここで、

$$R \equiv \left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)^{-1} \tag{2.7.82}$$
$$\Delta \equiv \hat{\boldsymbol{w}}^T Q^{-1}\hat{\boldsymbol{w}} + y^2 D^{-1} - \boldsymbol{\xi}^T\left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)\boldsymbol{\xi} \tag{2.7.83}$$
$$\boldsymbol{\xi} \equiv \left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)^{-1}\left(Q^{-1}\hat{\boldsymbol{w}} + \boldsymbol{x}D^{-1}y\right) \tag{2.7.84}$$

であり、$\boldsymbol{\xi}$ の転置は

$$\boldsymbol{\xi}^T = \left(Q^{-1}\hat{\boldsymbol{w}} + \boldsymbol{x}D^{-1}y\right)^T\left\{\left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)^{-1}\right\}^T = \left(\hat{\boldsymbol{w}}^T Q^{-1} + yD^{-1}\boldsymbol{x}^T\right)\left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)^{-1} \tag{2.7.85}$$

であることと、

$$\begin{aligned}
(\boldsymbol{w}-\boldsymbol{\xi})^T\left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)(\boldsymbol{w}-\boldsymbol{\xi}) = \boldsymbol{w}^T\left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)\boldsymbol{w} - \boldsymbol{w}^T\left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)\boldsymbol{\xi} \\
- \boldsymbol{\xi}^T\left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)\boldsymbol{w} + \boldsymbol{\xi}^T\left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)\boldsymbol{\xi}
\end{aligned} \tag{2.7.86}$$

を用いた。この結果を (2.7.10) 式に用いると、

$$p(\boldsymbol{w}|y,z) = \frac{(2\pi)^{-\frac{K+1}{2}}|D|^{-\frac{1}{2}}|Q|^{-\frac{1}{2}}e^{-\frac{\Delta}{2}}\exp\left[-\frac{1}{2}(\boldsymbol{w}-\boldsymbol{\xi})^T R^{-1}(\boldsymbol{w}-\boldsymbol{\xi})\right]}{(2\pi)^{-\frac{K+1}{2}}|D|^{-\frac{1}{2}}|Q|^{-\frac{1}{2}}e^{-\frac{\Delta}{2}}\displaystyle\int\exp\left[-\frac{1}{2}(\boldsymbol{w}-\boldsymbol{\xi})^T R^{-1}(\boldsymbol{w}-\boldsymbol{\xi})\right]d\boldsymbol{w}}$$

$$= (2\pi)^{-\frac{K}{2}} |R|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\boldsymbol{w}-\boldsymbol{\xi})^T R^{-1}(\boldsymbol{w}-\boldsymbol{\xi})\right] \tag{2.7.87}$$

となることから、

$$\boldsymbol{w}|y, z \sim N(\boldsymbol{\xi}, R) \tag{2.7.88}$$

であることがいえる。

続いて、$\boldsymbol{\xi}$と R を変形することで (2.7.30) 式 ∼(2.7.33) 式を導く。まず $\boldsymbol{\xi}$ については、

$$\begin{aligned}
\boldsymbol{\xi} &= \left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)^{-1}\left(Q^{-1}\hat{\boldsymbol{w}} + \boldsymbol{x}D^{-1}y\right) \\
&= \left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)^{-1}\left[\left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)\hat{\boldsymbol{w}} + \boldsymbol{x}D^{-1}\left(y - \boldsymbol{x}^T\hat{\boldsymbol{w}}\right)\right] \\
&= \hat{\boldsymbol{w}} + \left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)^{-1}\boldsymbol{x}D^{-1}\left(y - \boldsymbol{x}^T\hat{\boldsymbol{w}}\right) \\
&= \hat{\boldsymbol{w}} + Q\boldsymbol{x}\left(\boldsymbol{x}^TQ\boldsymbol{x} + D\right)^{-1}\left(y - \boldsymbol{x}^T\hat{\boldsymbol{w}}\right) \\
&= \hat{\boldsymbol{w}} + \boldsymbol{K}\left(y - \boldsymbol{x}^T\hat{\boldsymbol{w}}\right)
\end{aligned} \tag{2.7.89}$$

となる。ここで $\boldsymbol{K}$ は

$$\boldsymbol{K} \equiv Q\boldsymbol{x}\left(\boldsymbol{x}^TQ\boldsymbol{x} + D\right)^{-1} \tag{2.7.90}$$

である。上記の式変形では、恒等式 $\boldsymbol{x}D^{-1}\left(\boldsymbol{x}^TQ\boldsymbol{x} + D\right) = \left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)Q\boldsymbol{x}$ に左から $\left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)^{-1}$ を、右から $\left(\boldsymbol{x}^TQ\boldsymbol{x} + D\right)^{-1}$ を掛けると、

$$\left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)^{-1}\boldsymbol{x}D^{-1} = Q\boldsymbol{x}\left(\boldsymbol{x}^TQ\boldsymbol{x} + D\right)^{-1} \tag{2.7.91}$$

が得られることを用いた。最後に R については、

$$\begin{aligned}
R &= \left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)^{-1} \\
&= \left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)^{-1}\left[\left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)Q - \boldsymbol{x}D^{-1}\boldsymbol{x}^TQ\right] \\
&= Q - \left(\boldsymbol{x}D^{-1}\boldsymbol{x}^T + Q^{-1}\right)^{-1}\boldsymbol{x}D^{-1}\boldsymbol{x}^TQ \\
&= Q - Q\boldsymbol{x}\left(\boldsymbol{x}^TQ\boldsymbol{x} + D\right)^{-1}\boldsymbol{x}^TQ \\
&= Q - \boldsymbol{K}\boldsymbol{x}^TQ
\end{aligned} \tag{2.7.92}$$

ここで再び (2.7.91) 式を用いた。(2.7.88) 式を元の表記で書くと

$$\boldsymbol{w}_t|y_{1:t} \sim N(\boldsymbol{\xi}, R) \tag{2.7.93}$$

となる。これは $\boldsymbol{w}_t$ の時刻 t におけるフィルタ分布の期待値が $\boldsymbol{\xi}$、分散共分散行列が R であることを意味するので、$\boldsymbol{w}_{t|t} \equiv \boldsymbol{\xi}$ および $Q_{t|t} \equiv R$ とおく。以上の結果をまとめて元の表記で書くと、

$$\boldsymbol{w}_t|y_{1:t} \sim N(\boldsymbol{w}_{t|t}, Q_{t|t}) \tag{2.7.94}$$

$$\boldsymbol{w}_{t|t} = \boldsymbol{w}_{t|t-1} + \boldsymbol{K}_t\left(y_t - \boldsymbol{x}_t^T\boldsymbol{w}_{t|t-1}\right) \tag{2.7.95}$$

$$Q_{t|t} = Q_{t|t-1} - \boldsymbol{K}_t\boldsymbol{x}_t^TQ_{t|t-1} \tag{2.7.96}$$

$$\boldsymbol{K}_t = Q_{t|t-1}\boldsymbol{x}_t\left(\boldsymbol{x}_t^TQ_{t|t-1}\boldsymbol{x}_t + D_t\right)^{-1} \tag{2.7.97}$$

が得られる。ここで $\boldsymbol{K}_t \equiv \boldsymbol{K}$ とした。

2.8 診断手法 [1]

2.8.1 概要

診断手法は、前節までに述べた統計手法と異なり、過去の研究や目的変数の定義から予測式を決定し、ガイダンスの予測値を算出する手法である。診断手法が利用されるようになった背景には、数値予報モデルの精緻化や予測精度の向上がある。近年の数値予報は、ガイダンスの運用が開始された頃に比べ、統計的な補正を行わない簡易的な手法でも、実用的な精度を持つガイダンスを作成することが可能になってきている。

統計手法と比べて、診断手法は次のような特長を持つ。

① 観測や数値予報モデルの長期間のデータが不要。
② 観測密度に起因する予測精度の不均一性がない。
③ メリハリの効いた予測が可能。

統計手法では係数を学習するために長期間の観測と数値予報モデルのデータが必要となるが、診断手法では係数を学習する必要がないため、長期間のデータの蓄積は不要[2]である（①）。また、統計手法では、係数を学習するために予測対象とする地点や領域で十分な数の観測データが必要となるが、診断手法では観測データが少ない海上や上空でも他の地点と同程度の精度を持つ予測値が得られる（②）。さらに、統計手法では過去データ全体に対して平均的な誤差が小さくなるように予測式が求まる。その結果、稀な現象の予測頻度は低くなり、また、MOS の場合には予報時間が長くなると予測が気候値に近づいていく (第 2.9 節)。これに対し、診断手法では数値予報モデルの誤差を考慮しないため、メリハリの付いた予測が可能となる（③）。これは現象の発生頻度や予報時間に応じた予測の不確実性によらず同じ予測式を適用することで得られる特徴である。

診断手法には統一的な作成手法はなく、予測要素に応じて手法を開発する必要がある。本節では、診断手法を利用したガイダンスを例示し、診断手法を利用する上での留意事項を述べる。

2.8.2 診断手法を利用したガイダンスの運用

前項で診断手法は過去の研究や目的変数の定義から予測式を決定することを述べたが、気象庁で診断手法を利用するガイダンスの多くは、実際にはモデルの予測特性や観測値に合わせて予測式を調整している。これは診断手法の利点を生かしつつ、数値予報モデルの系統誤差などを出来るだけ軽減し、ガイダンスの予測精度の向上を目指すためである。利用するモデルに合わせた調整の実際を、MSM 視程分布予想を例として紹介する。

MSM 視程分布予想は MSM で予測された降水量、降雪量、相対湿度、風速から、光の消散率（消散係数 σ）を求め、これを視程 VIS($=3/\sigma$) に変換する。詳細は第 4.9 節で述べるが、この消散係数の一つである雲粒による消散係数 σ_c の予測式は、Gultepe et al. (2006) を参考に METAR[3] 及び SPECI[4] の視程と MSM の雲水量の関係を元に調整した (井藤 2011)。Gultepe et al. (2006) は測器観測による雲水量 LWC [g m^{-3}] から、雲粒の消散係数 σ_c による視程 VIS を以下のように見積もった。

$$\mathrm{VIS} \quad = \quad 3/\sigma_c = 0.0219 \times \mathrm{LWC}^{-0.9603} \qquad (2.8.1)$$

MSM 視程分布予想（運用開始時）は、この式を参考に、MSM の雲水量 QC [g kg^{-1}] と観測値の関係から、以下を予測式とした。

$$\mathrm{VIS} \quad = \quad 3/\sigma_c = 0.333 \times \mathrm{QC}^{-0.9} \qquad (2.8.2)$$

この例のように、診断手法を用いたガイダンスの多くは、利用するモデルや観測値に合わせて予測式の調整を行っている。ただし、診断手法で得られる予測式を用いて予測値の分布や特性を確認するといった簡易的な調整がほとんどであるため、調整に必要なデータは統計手法に比べて少なくて済む。

2.8.3 診断手法の適用例

ここでは診断手法を利用しているガイダンスを紹介する。なお、各ガイダンスの詳細は第 4 章で解説するため、ここでは概略を述べるに留める。

(1) 視程分布予想

詳細は第 4.9 節を参照。視程分布予想は地上及び海上の水平視程を面的に予測するガイダンスで、濃霧等の視程障害の予報に利用されている。視程の観測点は、気温や雨の観測点に比べて数が少ないため、視程分布予想の運用を開始するまでは視程障害に関する面的な予測資料がなかった。しかし、数値予報モデルの改良によって地表付近の雲水量が精度よく予測できるようになり、診断手法による予測が視程障害を予測する際の参考資料として実用的な精度を持つことが確かめられたため、2011 年に MSM 視程分布予想の運用を開始した。視程分布予想は GSM, MSM, LFM[5] で運用されている。各数値予報モデルの特性を考慮するために、光消散係数の各要素（浮遊塵 σ_p、雲粒 σ_c、雨粒 σ_r、雪 σ_s）の予測式をモデルに合わせて調整している (井藤 2011, 2013; 金井ほか 2015; 後藤 2017)。ただし、領域によって予測式を変えていない[6]ため、例えば、北日

[1] 後藤 尚親
[2] 第 2.8.2 項で示すように、予測式の調整に数値予報モデルのデータを利用するガイダンスの場合は、データの蓄積とモデル更新時に作業が必要となる。ただしその場合でも統計手法に比べると、必要なデータの量や作業は少なくて済む。
[3] 航空気象定時観測気象報
[4] 航空気象特別観測気象報
[5] LFM 航空悪天 GPV の一要素として作成している。
[6] 例外として、GSM 視程分布予想ではオホーツク海とそれ以外の領域で予測式を変えている。

本・東日本太平洋側では霧が多く予測されるが、南シナ海では霧の予測が少ないといった問題がある。

(2) 降水種別ガイダンス

詳細は第 4.3 節を参照。降水種別ガイダンスは雨、雨か雪、雪か雨、雪の 4 つの降水の種別を予測するガイダンスである。降水の種別を観測する地点は少ないが、面的な天気ガイダンスや最大降雪量ガイダンスには面的な降水種別の予測が必要であることから診断手法が用いられている。降水種別の判別には柳野 (1995) の地上気温と相対湿度で降水種別を判定する雨雪境界線を基に、2004 年から 2008 年の 5 年間の冬季の 51 の気象官署の観測データを対象に調整を行った結果を利用している。

降水種別ガイダンスは、実況の気温と相対湿度から作成した雨雪境界線に数値予報モデルの系統誤差を補正した格子形式気温ガイダンスと数値予報モデルの相対湿度を入力し、上空の気温等で補正することで降水種別を予測している。相対湿度については数値予報モデルの予測値をそのまま利用しているため、系統誤差が補正されていない課題がある。

(3) 着氷指数

詳細は第 4.12 節を参照。着氷指数は航空悪天 GPV で算出している要素の一つで、航空機への着氷を予測する指数である。飛行中に発生する着氷の予測には過冷却水滴の有無やその量が重要になるが、数値予報モデルで過冷却水滴を正確に予測することは現状ではまだ難しい。また、航空機による着氷の事例数が極端に少ないことから、一般的な統計手法を利用することは難しい。そこで過去の気温と着氷頻度及び湿数と着氷頻度の調査に基づき着氷指数を開発し運用している (工藤 2008)。この指数は数値予報モデルの気温・湿数と観測された着氷頻度の関係を関数で当てはめ、2 つの関数を掛けあわせて予測式を作成している。予測式の作成にモデルの気温と湿数を用いることで、利用するモデルに適した予測式となるように調整している。

(4) 積乱雲量・積乱雲頂高度

詳細は第 4.13 節を参照。積乱雲量・積乱雲頂高度は航空悪天 GPV で算出している要素の一つで、パーセル法に基づいて、数値予報モデルの気温や相対湿度などから積乱雲の雲量と雲頂高度を算出している。空域予報においては陸上だけでなく海上も含めた領域での予測が必要であることや、積乱雲の雲量や雲頂高度の実況を面的に入手することが出来ないことから、診断手法を用いて積乱雲の予測を行っている。

(5) 圏界面

圏界面は航空悪天 GPV で算出している要素の一つで、高層気象観測指針にある気圧と気温を用いた第一圏界面の定義を元に、モデルの気圧と気温から算出している。圏界面の実況は高層観測地点でしか得られないが、航空機の運航を支援するためには面的な予測が必要となるため、診断手法により圏界面の気圧などを面的に算出している。

2.8.4 利用上の留意点

診断手法は過去の研究や目的変数の定義に基づく予測手法であり、統計手法を利用する場合と比べて、長期間のデータ蓄積が不要、観測データが少ない地点や領域での予測が可能、メリハリのある予測が可能、という利点があることを述べた。一方で、診断手法には以下のような問題点もある。

① 数値予報モデルの系統誤差を補正できない。
② 予測対象地点や時刻ごとの誤差特性を反映した予測式を作成することが難しい。
③ 数値予報モデルの特性が大きく変わった場合に予測精度の低下や予測特性の変化が起きる可能性がある。

①については、本節で述べたように、モデルごとに予測式を調整することである程度の系統誤差を取り除くことができる。ただし、予測式を調整しすぎると③の問題も生じる。降水種別ガイダンスで行っているように、数値予報モデルを入力とする代わりに、系統誤差が補正されたガイダンスを利用することで解決できる場合もあるが、視程分布予想における雲水量や積乱雲予測における上空の気温や相対湿度など、多くの場合はガイダンス値が得られない。このため診断手法では、同じモデル予測値を利用するとしても、可能な限り系統誤差の少ないデータを入力とすることが望ましい。②は診断手法にとってはデメリットであり、これを解決するには、多くの観測データを入手して統計手法を利用する必要がある。③について、同様のことは統計手法を用いたガイダンスでも起こりうるが、例えば第 3.2 節で述べるように、統計手法の場合にはモデル更新時に長期間の再予報を実施して係数を再学習することで、更新前と同程度の予測精度を得られる場合が多い。診断手法でも、係数を調整することで予測精度の低下を回避・軽減することは可能であるが、係数の調整のみでは十分な予測精度が得られない場合がある。このような場合には予測手法自体の大幅な見直しも必要になる。診断手法によるガイダンスを開発・運用するに当たっては、これらの点に留意が必要である。

参考文献

後藤尚親, 2017: MSM ガイダンスの特性の変化. 平成 29 年度数値予報研修テキスト, 気象庁予報部, 56–60.

Gultepe, I, M.D.Müller, and Z.Boybeyi, 2006: A New Visibility Parameterization for Warm-Fog Applications in Numerical Weather Prediction Models. *J. Appl. Meteor. Climat.*, **45**, 1469–1480.

井藤智史, 2011: 視程分布予想の解説. 平成 23 年度数値予報研修テキスト, 気象庁予報部, 25–29.

井藤智史, 2013: GSM 視程分布予想の開発. 平成 25 年度数値予報研修テキスト, 気象庁予報部, 58–62.
金井義文, 満満男, 工藤淳, 2015: 下層悪天予想図及び新しい狭域悪天予想図. 航空気象ノート第 77 号, 気象庁航空気象管理官.
工藤淳, 2008: 国内航空悪天 GPV. 平成 20 年度数値予報研修テキスト, 気象庁予報部, 92–97.
柳野健, 1995: ニューラルネットによるガイダンス. 平成 7 年度量的予報研修テキスト, 気象庁予報部, 54–69.

2.9 頻度バイアス補正 [1]

2.9.1 はじめに

頻度バイアス補正は、予測頻度を実況頻度に合わせるように予測値を補正する CDF (Cumulative Distribution Function; 累積分布関数) マッチング [2] の一手法である。気象庁では予測頻度の実況頻度に対するバイアスを補正する手法という意味で「頻度バイアス補正」という名称で呼んでいる。海外の気象機関でもバイアス補正または quantile mapping (Maraun 2013) と呼ばれている。

気象庁のガイダンスには、1996 年 3 月、降水量ガイダンスにカルマンフィルタ (KF) を導入した際に合わせて導入された (國次・藤田 1996; 藤田 1996)。KF の導入と同時に導入されたが、KF の手法に問題があって導入されたわけではなく、線形重回帰 (MLR) を手法として使っていた際にも課題であった大雨の予測頻度が少ない点を修正するために導入されている。その後、頻度バイアス補正は降水量ガイダンス以外にも適用され、2018 年現在、風ガイダンス、視程ガイダンス、降雪量地点ガイダンス、雲ガイダンスにも使われている。これらのガイダンスではいずれも KF またはニューラルネットワーク (NN) の後処理として頻度バイアス補正が行われている。

ここでは、頻度バイアス補正がガイダンスの後処理として何故必要かを解説し、その後、気象庁での頻度バイアス補正の具体的な手法及びその逐次更新の手法を解説する。最後にまとめとして気象庁以外での利用について簡単に述べる。

2.9.2 頻度バイアス補正の必要性

例えば降水量の実況頻度は弱い降水に偏っており、強い降水の頻度は極端に少ない。このような場合に、予測の誤差を減らそうとすると、頻度の多い弱雨のデータを重視した予測となる。図 2.9.1 に予測式と、各説明変数における誤差の頻度分布のイメージを示した。弱雨の頻度が多いことにより、①弱雨を予測する所で誤差の頻度分布のピークが大きくなること、②誤差分布は正規分布ではなく、弱雨側にピークがあり、この誤差分布に引きずられて予測式は弱雨側に設定される [3]。また、気象予測の場合は低気圧等の擾乱の位置ずれなどで誤差が生じ、同じ説明変数でも実況は大雨と弱雨となる場合があるため、予測は中間的な値となって、大雨が予測されにくくなることも原因の一つと考えられる（図 2.9.1 の A）。

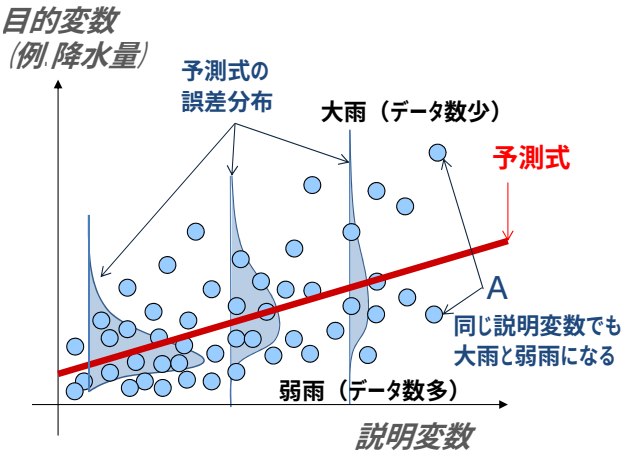

図 2.9.1 頻度が偏った目的変数を予測した場合の予測式と、その誤差の頻度分布のイメージ。説明変数を 1 個と仮定した場合の例。青の丸は個々の説明変数と目的変数のデータ、青のハッチの分布は予測式の各説明変数値での誤差の頻度分布（横軸側に頻度分布）イメージを示す。

図 2.9.2 に、風ガイダンスの場合における、頻度バイアス補正前と後の予測頻度と実況頻度を示す。補正前は、頻度の多い弱風の予測頻度が実況頻度より多く、頻度の少ない強風の予測頻度は実況頻度より少なくなっていることがわかる。強風の予測が実況より頻度が少ないということは、見逃しが多いことを意味するため、防災気象情報の作成を支援するガイダンスとしては問題となる。これに対し頻度バイアス補正後は、実況と予測の頻度が概ね一致し、強風の予測事例が増えている。図 2.9.3 には同じ風ガイダンスの予測と実況の散布図を示した。頻度バイアス補正前には弱めに予測されていること、頻度バイアス補正後は風速を強くするように補正が行われていることがわかる。風速を強くすることによって、強風の捕捉率が上昇しており（赤のハッチ部分）、防災気象情報にとってはより有効な予測となっている。一方、頻度バイアス補正は実況の頻度に合わせるように風を強めに補正するため、空振りも増えている（青のハッチ部分）。

2.9.3 頻度バイアス補正の手法

頻度バイアス補正の手法は単純で、予測の頻度が実況の頻度と合うように予測値を補正するだけである。図 2.9.4 は、風ガイダンスを例とした、CDF マッチングとしての頻度バイアス補正のイメージ図である。補正前のガイダンスは弱風の頻度が実況より多く、強風の頻度が少ないため、累積した頻度分布である CDF は実況の CDF より左側にある。その予測の CDF を実況の CDF に補正する手法が頻度バイアス補正である。つまり、図中に示した太矢印のようにガイダンス（補正前）の出力値を上方補正し、実況の CDF に合わせる。図 2.9.5 に風ガイダンス（定時風）等で実際に使われている頻度バイアス補正の方法を示す。頻度バイアス補正では、観測値と予測値に数個の閾値を設定

---

[1] 高田 伸一

[2] 2 つの母集団があり、片方の頻度分布をもう片方の頻度分布に合わせるように値を補正する手法。以下で説明する手法の他に、変換係数で合わせる手法 (幾田 2017) がある。

[3] KF は観測ノイズが正規分布に従うことを仮定しており、この仮定が成り立っていない影響が出ているとも言える。今後は目的変数を変換したり、正規分布を仮定しない他の手法の導入を検討する必要がある。

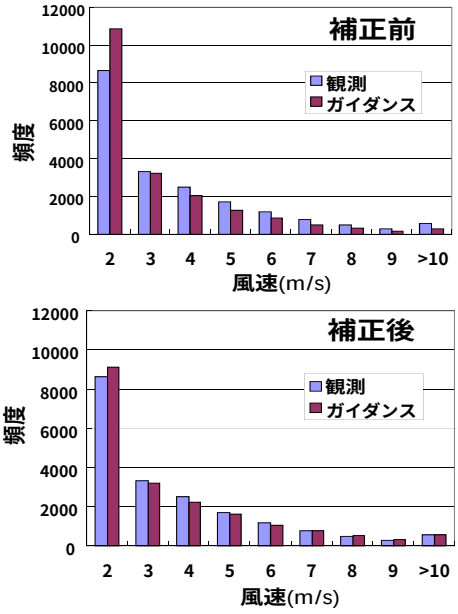

図 2.9.2　風ガイダンス（定時風）の予測頻度と実況頻度。上が頻度バイアス補正前、下が頻度バイアス補正後。1997 年 12 月から 1998 年 2 月の全アメダス地点の合計。

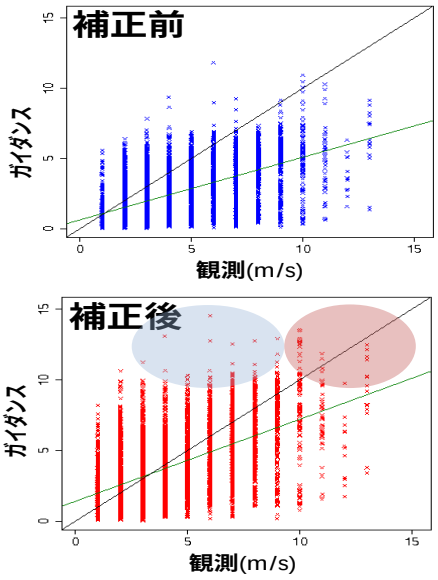

図 2.9.3　風ガイダンス（定時風）の実況と予測の散布図。上が頻度バイアス補正前、下が頻度バイアス補正後。図 2.9.2 と同じ 1997 年 12 月から 1998 年 2 月の全アメダス地点のデータを元に作成。下の赤のハッチ部分は、頻度バイアス補正によって 10 m/s 以上の風の捕捉率が増えたことを示す。青のハッチ部分は逆に 10 m/s 以上の予測の空振りが増えたことを示す。緑線は観測とガイダンスの回帰直線を示す。

し、各カテゴリでの頻度が等しくなるように補正を行う。風ガイダンスの場合は、観測側に 4 つの閾値（2.5, 5.5, 9.5, 13.0 m/s）を設定し、カテゴリを 4 つに分ける。その 4 つのカテゴリ毎に観測の頻度を数え、それと同じ頻度となる予測側の閾値を計算する。この例の場合は 4 つの観測側の閾値に対応し、4 つの予測側の閾値（1.9, 3.8, 7.1, 9.8 m/s）を計算で求めている。こ

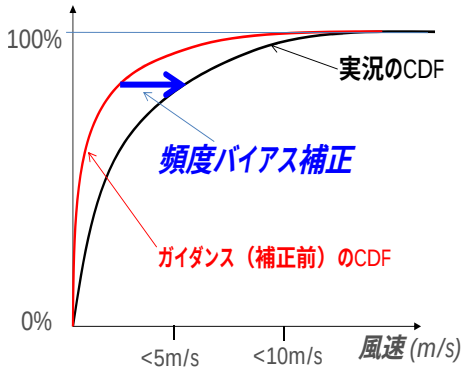

図 2.9.4　風ガイダンスを例とした、実況の CDF（累積分布関数）、補正前のガイダンスの CDF 及び頻度バイアス補正のイメージ。

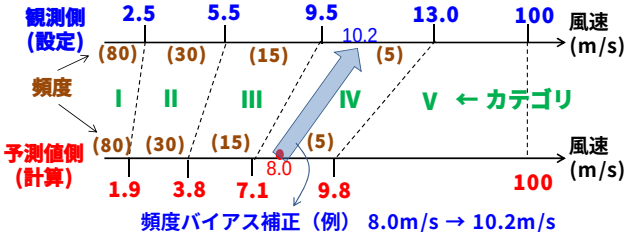

図 2.9.5　風ガイダンス等で実際に使われている頻度バイアス補正の手法。観測側の閾値（青色）を設定して各カテゴリ（緑色）に分け、その各カテゴリの頻度と同じになるように予測値側の閾値（赤色）を計算する。頻度バイアス補正は太矢印（薄青色）のように行う。

の両方の閾値に沿って例えば予測値が 7.1 m/s であれば 9.5 m/s に補正する。途中の 8.0 m/s であれば線形補間で 10.2 m/s と補正値を決定する。なお、この例では一番右側に補正を行う上限値として 100 m/s が設定してある。この上限値以上では補正を行わず、ガイダンス（補正後） ＝ ガイダンス（補正前）とする。

頻度バイアス補正を行う場合に留意すべき点として、各カテゴリに含まれるサンプル数が少なくなりすぎないように閾値を設定することが挙げられる。サンプル数が少ない状態で CDF マッチングすると、過大な予測がされる可能性がある。図 2.9.4 の 100%近いところ（図の上部）では補正量が大きく、またわずかな頻度の違いで CDF の線が変わる所でもある。強風、大雨等の予測に大きな影響を与えることから、無理な補正とならないよう、サンプル数に十分気をつけて観測側の閾値を設定する必要がある。

2.9.4　頻度バイアス補正の逐次更新

逐次学習型ガイダンスの場合には、前項で説明した予測側の閾値を逐次更新する。KF や NN で予測式の係数が更新する際に、頻度バイアス補正の予測側の閾値も同時に更新する。閾値更新のイメージ図を図 2.9.6 に示す。頻度バイアス補正では、予測値と観測値のカテゴリが異なった場合に予測側の閾値を更新する。観測値の方が低いカテゴリに入った場合は予測側の閾値を上げるように、観測値の方が高いカテゴリに入った

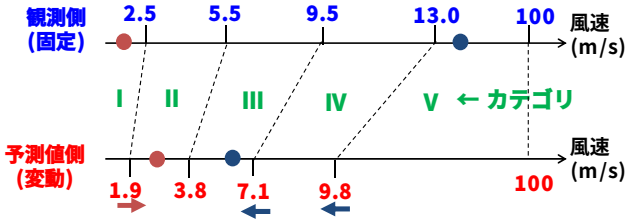

図 2.9.6 風ガイダンスを例とした、頻度バイアス補正の逐次的な更新方法。丸（赤、紺）は新しい予測と観測の事例を示し、矢印（赤、紺）はその時の予測値側閾値の修正の方向を示している。その他は図 2.9.5 に同じ。

場合は予測側の閾値を下げる。この閾値をどの程度変化させるかは、ガイダンスの予測精度を調べながら調整して決めている。図の紺色の例のように、カテゴリが 2 つ以上ずれた場合には、その途中の閾値を全て更新する。また、あるカテゴリの予測値側の閾値を変化させる場合に、その上下の閾値を越えないように調整を行う。閾値を下げる場合と上げる場合で、閾値を変化させる割合は同じ値が用いられることが多いが、降水量ガイダンスでは同じ割合とすると大雨の予測頻度が過剰になるため、予測値の閾値を上げる割合の方が多い設定としている (白山 2017)。

2.9.5 アンサンブル平均での頻度バイアス補正の利用

上記では、実況の頻度に偏りがある場合の対策として、頻度バイアス補正を利用することを説明した。一方、アンサンブル予報のメンバーの予測を平均すると、同じように強雨や強風の頻度が減る。なぜなら、例えば頻度の少ない強雨域は通常狭く予測されるため、強雨域が異なった予測を平均すると、強雨と弱雨の平均値となり強雨のピークが減るからである。また、メンバー間の強雨の時間的なずれによっても強雨のピークが減る。この平均によって降水量の誤差は減るかもしれないが、全体的に鈍った予測となり、強雨の予測が出にくくなる。このようなアンサンブル平均の問題点を補正するためにも、頻度バイアス補正が使われている。例えば、第 1.4 節で説明した米国気象局の NBM（National Blend of Models）の降水量予測においては、多くの予測が結合された後に、頻度バイアス補正（論文中では quantile mapping）によって大雨が予測されるように補正されている (Hamill et al. 2017)。また、第 5.1 節で説明する、開発中の LFM 降水量ガイダンスでも、初期値アンサンブルを行った後に頻度バイアス補正を行う予定である。

2.9.6 海外での状況

米国気象局では、風ガイダンスにおいて強風の予測が出にくいことから、風速のバイアス補正として inflation という方法を 1975 年から利用していた (Schwartz and Carter 1982)。ただし、この手法は頻度バイアス補正ではなく、MLR で予測式を作成した際の重相関係数 R（第 2.4 節）を使って以下のように風速を強めに補正していた。

$$S_{\mathrm{after}} = S_{\mathrm{mean}} + \frac{S_{\mathrm{before}} - S_{\mathrm{mean}}}{R} \qquad (2.9.1)$$

ここで S_{after} は補正後の風速、S_{before} は補正前の風速、S_{mean} は重回帰分析を行った期間の平均風速であり、S_{before} が S_{mean} より小さい場合は補正を行わない。この手法は重相関係数 R が大きい（予測精度が高い）場合は補正を小さく、R が小さい（予測精度が低い）場合は補正を大きくする手法である。図 2.9.1 で予測式の誤差分布のイメージを示したが、予測精度が低いと（誤差が大きいと）、中間的な予測となりやすく強風がより予測されにくくなるため、このような予測精度を考慮した補正方法が有効だったと考えられる。なお、2018 年現在はこの手法でなく、予測と実況を比較し、その差を逐次的に計算した量を inflation として風速を増大する手法が使われている (Glahn et al. 2014)。

フランス気象局では、WMO の Technical Progress Report によると風速、降水量ガイダンスにおいてモデル予測値を校正 (Calibration) していると記述されているなど頻度バイアス補正を利用しているようであるが、具体的な手法は不明である。

参考文献

藤田司, 1996: ガイダンスの検証. 平成 8 年度数値予報研修テキスト, 気象庁予報部, 34–40.

Glahn, B., D. Rudack, and B. Veenhuis, 2014: On bias correcting MOS wind speed forecasts. *MDL Office Note*, **14-1**, 1–19.

Hamill, T. M., E. Engle, D. Myrick, M. Peroutka, C. Finan, and M. Scheuerer, 2017: The U.S. National Blend of Models for Statistical Postprocessing of Probability of Precipitation and Deterministic Precipitation Amount. *Mon. Wea. Rev.*, **145**, 3441–3463.

幾田泰酵, 2017: 局地数値予報システムにおける新規観測データの利用開始及び同化手法の高度化. 平成 29 年度数値予報研修テキスト, 気象庁予報部, 82–85.

國次雅司, 藤田司, 1996: カルマンフィルターによるガイダンス. 平成 7 年度量的予報研修テキスト, 予報部予報課, 44–53.

Maraun, D., 2013: Bias correction, quantile mapping, and downscaling:Revisiting the inflation issue. *J. Climate*, **26**, 2137–2143.

Schwartz, B. E. and G. M. Carter, 1982: An evaluation of a modified speed enhancement technique for objective surface wind forecasting. *TDL Office Note*, **82-1**, 10pp.

白山洋平, 2017: 降水ガイダンスの改良. 平成 29 年度数値予報研修テキスト, 気象庁予報部, 86–93.

2.10　その他の統計手法 [1]

本節では第 2.4 節 ~ 第 2.9 節で述べた手法以外の統計手法について、その一部を簡単に紹介する。本節で述べる統計手法はいずれも 2018 年現在の気象庁のガイダンスには利用されていないが、海外気象機関のガイダンスでは利用されている手法もあり、今後新たな統計手法を導入するに当たって候補となりうる手法である。

2.10.1　決定木

決定木はフローチャートのように設定した説明変数の閾値によって、事象（晴、曇、雨など）をいくつかのクラスに分類する手法である。例えば図 2.10.1 の左図のように、2 つの説明変数 x_1 と x_2 を用いて現象あり（○）となし（×）を分けることを考える。このとき閾値 $\theta_1 \sim \theta_3$ を用いると、右図のような判別により現象のあり・なしを分類できる。図からわかるように、決定木では分類するクラスが線形分離不可能な場合でも判別可能である。予報作業で用いられるフローチャートは過去の事例を元に開発者が分類方法や閾値を決定する場合が多い [2] が、決定木ではそれを統計手法によって行う。決定木は主に分類問題に利用されるが、連続値に対する回帰を行うこともできる。回帰の場合には、分類された結果に属する学習データの平均値を出力値とする。このため出力される値は離散的な値となる。

決定木で分類方法を決定する代表的なアルゴリズムに CART(Breiman et al. 1984) や C4.5(Quinlan 1996) がある。これらは以下の手順で分類を行う。

1. 全ての学習データに対して設定した誤差関数が小さくなるようにデータを 2 分割する
2. 分割されたデータに対して誤差関数が小さくなるようにデータをさらに 2 分割する
3. 設定した停止条件（分岐の数や階層の数など）が満たされるまで手順 2 を繰り返す
4. 過学習を防ぐため、設定したパラメータに基づいて分岐を剪定する

決定木では最上位の分岐をルート、各分岐をノード、最下位の判別結果をターミナルノードと呼び、各ノードに含まれる学習データの数を例題と呼ぶ。誤差関数としては負のエントロピーやジニ係数 [3] などが用いられ、2 つに分けたデータの不純度が低くなるように（片方のクラスのデータが多く含まれるように）データが分割される。手順 4 の分岐の剪定では、ターミナルノードの数や階層の数などに比例したペナルティー項を誤差関数に加えることで、分類精度への影響の小さい分岐を剪定（削除）する。

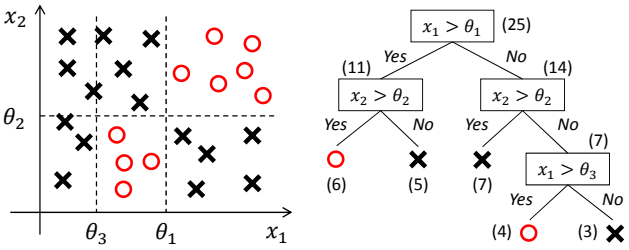

図 2.10.1　決定木での分類の例。○は現象あり、×は現象なしを表す。括弧内の数字は各ノードに含まれる例題（学習データ）の数。

決定木のメリットとして、判別の過程が理解しやすいことと、線形分離不可能な場合でも判別可能であることが挙げられる。デメリットとしては、判別結果の分散が大きいことが挙げられる。このことは、学習データが少し変わっただけで分割方法が変わってしまうことを意味する。特に上位の階層で分割が変わるとそれ以下の階層にも影響が伝播するため、決定木の構造が大きく変化してしまう。これは予測精度の低下を招く要因となる。

2.10.2　集団学習

線形重回帰やロジスティック回帰、ニューラルネットワークなどの統計手法は、目的変数と説明変数の関係を学習する機械（学習器）といえる。これらの学習器に説明変数を与えると予測値が得られる。ガイダンスの予測精度を向上させようとした場合、表現能力の高い学習器を用いることもできるが、そのような学習器は一般に計算コストが高く、またチューニングのためのパラメータも多くなる。そこで、表現能力は低いが計算コストやパラメータの少ない学習器を多く作成し、それらを組み合わせることで予測精度を向上させようという手法が集団学習である。集団学習はそれ自体は学習器ではないが、回帰手法や分類手法の精度を大きく向上させる可能性のある手法である。代表的な集団学習にバギング、ブースティング、ランダムフォレストがある。

(1)　バギング

バギング [4](Breiman 1994) では以下の手順で予測を行う（図 2.10.2）。

1. N 個の学習データからブートストラップ法（第 2.3.12 項）でサンプル数 N の学習データを R 組生成する
2. R 組の学習データに元の学習データを加えた $R+1$ 組の学習データに対して、ある一種類の統計手法（線形重回帰や決定木など）を用いて $R+1$ 個の

[1] 工藤 淳
[2] このためフローチャートは “subjective decision tree”（主観的決定木）とも呼ばれる。
[3] あるノードに含まれる例題を 2 つに分割する割合を p としたとき、負のエントロピーは $p \ln p + (1-p)\ln(1-p)$、ジニ係数は $2p(1-p)$ となる。
[4] Bagging, Bootstrap AGGregatING の略。

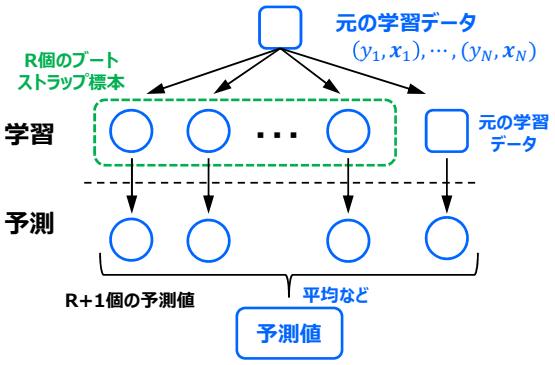

図 2.10.2　バギングによる予測の模式図

予測式を作成する

3. $R+1$ 個の予測式を用いて予測を行い、予測対象が連続値（回帰）の場合には各予測の平均を、クラス分類の場合には最も多く予測されたクラスを予測値とする

バギングは手法がシンプルであるため導入が容易であり、また各学習データに対する予測式作成を並列に実行できるため比較的短時間で学習することができる。学習器として回帰を利用した場合には、複数の予測式で予測する代わりに係数の平均を用いて予測を行えばよいため運用も容易である。

バギングでは、学習に使用するデータはランダムに選ぶことになるため、各ブートストラップ標本ごとに約 1/3 のデータは学習に使用されないことになる [5]。このことを利用すれば、交差検証と同様の効果が得られる。具体的には、学習に使用されなかったデータを検証データとして各ブートストラップ標本について予測精度を求め、それを全標本で平均する。これを Out-Of-Bag (OOB) 推定といい、OOB 推定に用いられる標本を OOB 標本と呼ぶ。

バギングは学習データが十分には得られない場合ほど予測精度改善の効果が大きいと考えられる。第 2.4.5 項で述べたように、学習データが少ない場合には係数の分散が大きく（推定の精度が低く）なり、そのことが予測精度を低下させる要因となっている。バギングでは精度の低い予測を平均することで係数の推定精度を高めることができる。

(2)　ブースティング

ブースティング (Kearns and Valiant 1994) は精度の低い学習器（弱学習器）を逐次的に組み合わせることで、精度の高い予測を行う手法である。ここでは代表的な 2 クラス分類に対するブースティングの手法である AdaBoost(Freund and Schapire 1997) について述べるが、ほかにも多クラス分類に対する AdaBoost の拡張である AdaBoost.M1, AdaBoost.M2 (Freund and Schapire 1997), AdaBoost.MH (Schapire and Singer 1999) や、ロジスティック回帰に適用した Logit Boost (Friedman et al. 2000) など様々な拡張手法が提案されている。

バギングでは学習データから均一な割合でサンプリングすることで新たな学習データを生成したが、AdaBoost では判別できなかったデータほど抽出されやすいように重みを付けることで、誤差が大きいデータを重点的に学習するようにする。ただし実際には重みに応じてサンプリングするのではなく、弱学習器を重みに応じて選択することでサンプリングに代えている。

今、N 個の学習データ $(y_1, \boldsymbol{x}_1), \cdots, (y_N, \boldsymbol{x}_N)$ と M 個の弱学習器 f_m $(m = 1, \cdots, M)$ があるとする。各 $\boldsymbol{x}$ は K 次元の説明変数ベクトル、y_n は目的変数で -1 または $+1$ である。f_m としては、例えば k 番目の説明変数を 1 つだけ用いた、1 階層しかない決定木（閾値 θ_m）などが用いられる。

$$f_m(x_{nk}) = \mathrm{sgn}(x_{nk} - \theta_m) \tag{2.10.1}$$

このような弱学習器の集合 $\{f_1, \cdots, f_M\}$ を $\mathcal{F}$ とする。最終的に求めたいものは、弱学習器を重み $\boldsymbol{\beta}$ で線形結合した $F(\boldsymbol{x}) = \sum_{m=1}^{M} \beta_m f_m(\boldsymbol{x})$ である。ここでは F を判別関数と呼ぶことにする。AdaBoost による繰り返し学習のステップ数を $r = 1, \cdots, R$ とし、r 回目の学習ステップでの判別関数を F_r、重みを $w_{n,r}$ と書く。

AdaBoost では以下の手順で判別関数を求める。

1. 重みの初期値 $w_{n,0}$ は全て $1/N$、判別関数の初期値は $F_0(\boldsymbol{x}) = 0$ とする。
2. 学習ステップ $r = 1, \cdots, R$ について以下の手順 3 ～ 4 を繰り返す。
3. $\mathcal{F}$ の中から以下の誤り率 ε_r が最も小さい弱学習器を選択し、それを g_r とする。

$$\varepsilon_r(f) = \sum_{n=1}^{N} w_{n,r} I\left[f(\boldsymbol{x}_n) \neq y_n\right] \tag{2.10.2}$$

$$g_r(\boldsymbol{x}_n) = \mathrm{argmin}_{f \in \mathcal{F}}\, \varepsilon_r(f) \tag{2.10.3}$$

ここで関数 $I[\,]$ は $[\,]$ 内の条件が真ならば 1、偽ならば 0 を返す関数である。

4. 次式により重みと判別関数を更新する。

$$w_{n,r+1} = \frac{\exp\left[-F_r(\boldsymbol{x}_n) y_n\right]}{Z_{r+1}} \tag{2.10.4}$$

$$F_r(\boldsymbol{x}_n) = F_{r-1}(\boldsymbol{x}_n) + \alpha_r g_r(\boldsymbol{x}_n) \tag{2.10.5}$$

$$Z_{r+1} = \sum_{n=1}^{N} \exp\left[-F_r(\boldsymbol{x}_n) y_n\right] \tag{2.10.6}$$

$$\alpha_r = \frac{1}{2} \ln \frac{1 - \varepsilon_r}{\varepsilon_r} \tag{2.10.7}$$

5. 判別関数を $F = F_R$ とし、$F(\boldsymbol{x})$ または $\mathrm{sgn}(F(\boldsymbol{x}))$ を出力する。

[5] N 回のサンプリングで、あるデータが一度も選ばれない確率は $(1 - 1/N)^N$ であり、N が大きい場合は $\lim_{N \to \infty} (1 - 1/N)^N = e^{-1} \simeq 0.368$

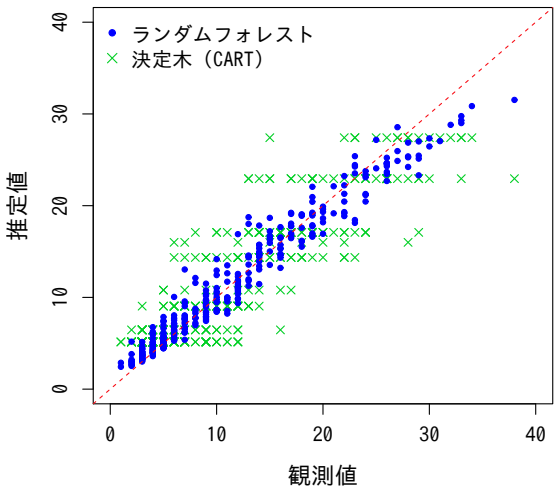

図 2.10.3　ランダムフォレストと CART を用いた決定木によるオゾン濃度の推定。横軸は統計ツール R のパッケージ“earth”に含まれている、ロサンゼルスで観測されたオゾン濃度の日最大値、縦軸は同データセットに含まれているロサンゼルスの地上の風や湿度などの観測値を説明変数として回帰を行った場合の推定値。

ブースティングは重みを逐次的に学習するため、並列実行可能なバギングと比べて計算に時間が掛かるというデメリットはあるが、バギングよりも予測精度が高くなる傾向がある。

(3)　ランダムフォレスト

ランダムフォレスト (Breiman 2001) は、決定木の集団学習法の一種で、集団学習の方法としてはバギングを採用している。ランダムフォレストではブートストラップ法でサンプリングした複数個の学習データに対して決定木で学習を行うが、決定木の学習とは以下の点が異なる。

- 学習に利用する説明変数は、全ての説明変数（K 個）を使うのではなく、ランダムに抽出された $\widetilde{K}$ 個だけ利用する。分類の場合には $\widetilde{K} = K/3$、回帰の場合には $\widetilde{K} = \sqrt{K}$ が推奨されている。
- ターミナルノードの例題の数は、分類の場合は 1 個、回帰の場合は 5 個が推奨されている。
- 決定木では過学習を防ぐために剪定を行うが、ランダムフォレストでは行わない。

ランダムフォレストの予測はバギングと同様に、分類の場合には各クラスに分類された例題の割合を、回帰の場合には平均値を用いる。

図 2.10.3 は、統計ツール R のパッケージ“earth”に含まれている、ロサンゼルスで観測されたオゾン濃度の日最大値を、地上の風や湿度などの観測値を説明変数としてランダムフォレストと CART による決定木で回帰を行った結果である。回帰の場合、ランダムフォレストと決定木はいずれも離散的な値をとるが、ランダムフォレストでは複数の決定木を用いて回帰を行うため、連続値に近い推定値が得られている。またランダムフォレストの方がばらつきが小さく推定の精度が高い。

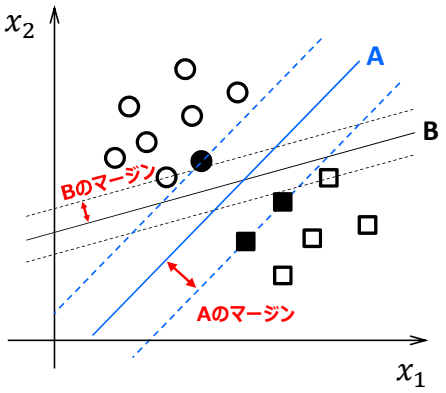

図 2.10.4　サポートベクターマシンでのマージン最大化の模式図。x_1, x_2 は説明変数、○と□は分類したい 2 つのクラス、● と■ はサポートベクターを表す。

ランダムフォレストは決定木のシンプルさを保持しつつ、予測精度が低い問題を解決している。また OOB 推定を用いることで、学習データだけを用いて予測データに対する精度を推定できるほか、説明変数の重要度 (Gregorutti et al. 2017) を評価することもできる。その方法は以下のとおりである。まず OOB 標本を用いて予測誤差を評価する。各 OOB 標本の誤差の平均値を E_{OOB} とする。続いて、OOB 標本の中の k 番目の説明変数をランダムに入れ替えて同様に予測誤差を評価する。このときの各 OOB 標本での予測誤差の平均値を $E^{(k)}_{OOB}$ とする。k 番目の説明変数の重要度 J_k は、

$$J_k = E^{(k)}_{OOB} - E_{OOB} \tag{2.10.8}$$

と定義される。入れ替えた説明変数の重要度が高いほど説明変数を入れ替えたことによる誤差が大きくなるため J_k が大きくなる。

2.10.3　サポートベクターマシン

サポートベクターマシン (Vapnik and Lerner 1963) は、クラス分類や回帰、パターン認識に用いられる統計手法の一つであり、優れた判別能力を持つことが知られている。サポートベクターマシンの特徴として、マージンを最大化するようにクラス分類を行うことと、カーネル法 (Boser et al. 1992) を用いることで線形分離不可能な場合でも適用できることが挙げられる。

サポートベクターマシンでクラス分類を行う例を図 2.10.4 に示す。図の○と□が分類したい 2 つのクラスで、x_1, x_2 は説明変数である。この場合 2 つのクラスを直線で分離する方法は、例えば図中に A と B で示した直線など無数に存在するのだが、サポートベクターマシンでは直線に最も近い各クラスの点との距離（これをマージンと呼ぶ）が最大になる直線を選択する。

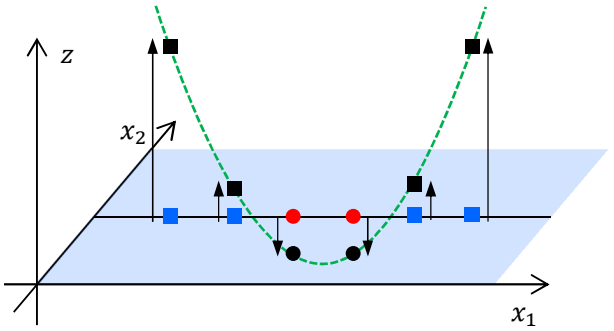

図 2.10.5　線形分離不可能なデータを高次元に射影することで線形分離可能になる例

図の例では青線で示した A の直線が選ばれることになる。このようにすることで 2 つのクラスが最も明確に分離されるとともに、直線 A を決定しているのは図中に● と■ で示した 3 点のみであり、元のデータと比べて非常に少ない数のデータでパラメータが決定されることになる。● と■ のように直線を決定するデータをサポートベクターと呼ぶ。

この例では一本の直線で誤りなく分離できているが、実際の問題では線形分離できるとみなせるような場合であったとしても、○の一部は□側に、□の一部は○側に含まれる場合が多いだろう。このような場合には、多少の誤りを許してマージンを決定することになる。これをソフトマージンという。

ソフトマージンを用いたとしても、そもそも線形分離不可能な場合にはクラス分けすることはできない。そこでサポートベクターマシンではカーネル法が利用される。カーネル法では元のデータを高次元空間に射影することで線形分離不可能なデータでも線形分離可能にすることができる。図 2.10.5 に簡単な例を示す。この図で x_1–x_2 平面上にある青四角と赤丸は一本の直線で分けることができない。すなわち線形分離不可能である。カーネル法ではこれを z 方向を含むような元の空間よりも高次元の空間に射影する。射影した空間上では元の青四角と赤丸は黒四角と黒丸のようになり、線形分離可能になる。ただし高次元に射影しただけでは計算量が増えてしまうため、射影したデータをそのまま利用するのではなく、射影したデータの内積をカーネル関数に置き換えるという処理を行う。

2.10.4　拡張カルマンフィルタ

第 2.7 節で述べたように、カルマンフィルタは線形・ガウス状態空間モデルに基づく手法であり、システムモデルや観測モデルが非線形である場合には適用することができない。そこで、非線形項を 1 次の項までででテーラー展開して線形化することで非線形関係も扱うことができるようにする。ここではガイダンスのカルマンフィルタにおいて、観測モデルが非線形である以下の状態空間モデルを考える。

$$\boldsymbol{w}_t = \boldsymbol{w}_{t-1} + \boldsymbol{u}_t \tag{2.10.9}$$

$$y_t = f_t(\boldsymbol{w}_t) + v_t \tag{2.10.10}$$

変数や添字の定義は第 2.7 節を参照していただきたい。ガイダンスで予測を行う時点での $\boldsymbol{w}_t$ の最適な推定量は一期先予測の係数 $\boldsymbol{w}_{t|t-1}$ であるから、$f_t(\boldsymbol{w}_t)$ を $\boldsymbol{w}_{t|t-1}$ のまわりでテーラー展開する。

$$f_t(\boldsymbol{w}_t) \approx c_{t|t-1} + \boldsymbol{f}_t'^{T}(\boldsymbol{w}_{t|t-1})\ \boldsymbol{w}_t \tag{2.10.11}$$

$$c_{t|t-1} \equiv f_t(\boldsymbol{w}_{t|t-1}) - \boldsymbol{f}_t'^{T}(\boldsymbol{w}_{t|t-1})\ \boldsymbol{w}_{t|t-1} \tag{2.10.12}$$

ここで $\boldsymbol{f}_t'$ は f_t の 1 階微分で $\boldsymbol{w}_t$ と同じ K 次元ベクトル、$c_{t|t-1}$ は非確率変数である。上式を用いると、(2.10.10) 式は、

$$y_t - c_{t|t-1} = \boldsymbol{f}_t'^{T}(\boldsymbol{w}_{t|t-1})\ \boldsymbol{w}_t + v_t \tag{2.10.13}$$

となり、ガイダンスのカルマンフィルタの観測方程式 (2.7.4) において、$y_t \to y_t - c_{t|t-1}$、$\boldsymbol{x}_t \to \boldsymbol{f}_t'(\boldsymbol{w}_{t|t-1})$ とした線形モデルとみなすことができる。この場合、ガイダンスのカルマンフィルタとの違いはフィルタの式のみである。

$$\nu_t = y_t - f_t(\boldsymbol{w}_{t|t-1}) \tag{2.10.14}$$

$$Q_{t|t} = Q_{t|t-1} - \boldsymbol{K}_t \boldsymbol{f}_t'^{T}(\boldsymbol{w}_{t|t-1}) Q_{t|t-1} \tag{2.10.15}$$

$$\boldsymbol{K}_t = \frac{Q_{t|t-1}\boldsymbol{f}_t'(\boldsymbol{w}_{t|t-1})}{\boldsymbol{f}_t'^{T}(\boldsymbol{w}_{t|t-1}) Q_{t|t-1} \boldsymbol{f}_t'(\boldsymbol{w}_{t|t-1}) + D_t} \tag{2.10.16}$$

これらの式を用いることで、観測モデルが非線形関数で表される場合でも、カルマンフィルタと同様に係数を更新することができる。カルマンフィルタの場合には係数の初期値は乱数や 0 などの適当な値で問題なかったが、ここでは $\boldsymbol{w}_t$ が $\boldsymbol{w}_{t|t-1}$ に近いと仮定して近似しているため、係数の初期値は何らかの方法（$y = f(\boldsymbol{w})$ を用いた非線形回帰など）で見積もった真値に近い値に設定する必要がある。

参考文献

Boser, B. E., I. M. Guyon, and V. N. Vapnik, 1992: A training algorithm for optimal margin classifiers. *Proceedings of the 5th Annual Workshop on Computational Learning Theory*, 144–152.

Breiman, L., 1994: Bagging predictors. *Technical Report No.421, Department of Statistics, University of California*, 1–19.

Breiman, L., 2001: Random forests. *Machine Learning*, **45(1)**, 5–32.

Breiman, L., J. H. Friedman, R. A. Olshen, and C. J. Stone, 1984: *Classification and Regression Trees*. CRC Press, 368 pp.

Freund, Y. and R. E. Schapire, 1997: A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, **55(1)**, 119–139.

Friedman, J., T. Hastie, and R. Tibshirani, 2000: Additive logistic regression: A statistical view of boosting. *The Annals of Statistics*, **28(2)**, 337–407.

Gregorutti, B., B. Michel, and P. Saint-Pierre, 2017: Correlation and variable importance in random forests. *Statistics and Computing*, **27(3)**, 659–678.

Kearns, M. and L. G. Valiant, 1994: Cryptographic limitations on learning Boolean formulae and finite automata. *Journal of the Association for Computing Machinery*, **41(1)**, 67–95.

Quinlan, J. R., 1996: Bagging, boosting, and C4.5. *Proceedings of the 13th National Conference on Artificial Intelligence*, 725–730.

Schapire, R. E. and Y. Singer, 1999: Improved boosting algorithms using confidence-rated predictions. *Machine Learning*, **37(3)**, 297–336.

Vapnik, V. and A. Lerner, 1963: Pattern recognition using generalized portrait method. *Automation and Remote Control*, **24(6)**, 774–780.