不動産市場とテック

Author1, Author2, Author3, Author4

xxx, 2019

目次

第 1 章

不動産市場とテクノロジー

1.1 AI と不動産業

AI(Artificial Intelligence)・ビッグデータ・IOT(Internet of Things) といった新しい技術は，新しい産業を創出するとともに，従来の産業の在り方を大きく変容させようとしている。国民経済計算 (GDP) の 10% 以上を占める不動産業においても同様であり，不動産業を取り巻き大きな変化が起ころうとしている。

不動産業において，AI 等のテクノロジーが注目されるきっかけとなったのが，米国の Zillow 社が，Zesimate ((https://www.zillow.com/zestimate/)) と呼ばれるサービスを世に出したことに始まったと言ってもいいであろう。同サービスでは，全米のすべての不動産に対してリアルタイムに価格の相場が表示されるとともに，その相場の変化をも示している。不動産市場は，とりわけ売りたいと思ったときにいくらで売れるのか，買いたいと思ったときに，提示された価格は適切であるのかといったことがわかりづらいことから，しばしば不透明な市場であると言われてきたことから，市場において，大きなインパクトがあったと言ってもいいであろう。

わが国の不動産業の法人数は，2010 年には 30 万社を超え，通年を通じて GDP に占める割合とほぼ同じ全産業の法人数の 10% を超える水準にあり，120 万人程度の就業者数を抱える。また，不動産業といっても幅が広く，①建物売業，土地売業，②不動産代理業 ・仲介業，③不動産賃業，④家業， 間業，⑤不動産管理業と，多くの業態を持つ。

一般に不動産業と聞くと，不動産代理業・仲介業を想像することが多いのではないか。全国どこに行っても，駅を降りると不動産屋さんの看板を見る。それらの多くが，不動産代理業，仲介業と呼ばれるものであり，不動産，とりわけ不動産を売りたい方，貸したい方と，買いたい方，借りたい方のマッチングをしている。そのような行為の中で，最も重要な仕事の一つで，売れる価格，または買う際の適正な価格を決定するということである。

しかし，前述のように，不動産市場は情報が不完全であり, そのために不透明な市場であるといった指摘を受けることが多い。一般に市場がその資源配分機能を十分に発揮するためには, 取引対象となる財の質と価格についての情報が市場における取引参加者に十分にいきわたっていること，そして適切な取引対象（相手）を見出し，取引を実現するための特別な費用が存在していないという条件が求められる。しかし，多くの市場においては，情報は完全ではなく，取引を実現するための機会費用も含めてさまざまな費

用が発生していると考えてもいいであろう。そのために，不動産業が十分に機能しないために，または，そのような情報の非対称性が不動産仲介業の介在価値の一つとして考えられていることから，不動産業に対する誤解が生まれ，不動産業者が情報を囲い込んでいて，その社会的費用を増幅させているようなゴシップ記事などが週刊誌を賑わすことも少なくない。

しかし，不動産業者においても，実は正しい価格はわからないと思ったほうがよい。不動産市場は，「同質の財が存在しない」といった特殊性を有しているために，他の市場財と比較して，正しい市場価格を探し出すことは，極めて困難なのである。さらに，不動産市場においては，わが国だけでなく，ほとんどの国で市場情報が不足している。すべての不動産が常に取引がされているわけではなく，不動産の価格を決定している，不動産や土地の品質を含むすべての属性情報やその不動産を取り巻くエリアの環境情報が揃っているわけではない。

つまり，多くの市場財では，価格を決定している品質に関わる情報の種類が少なく，また，市場で取引されている取引価格を容易に知ることができるものの，不動産市場においては，取引価格に関する情報を知ることができないことの方が多い。このような社会的な問題を解決するために，多くの国で不動産鑑定業というものも存在しており，不動産価格を「不動産鑑定士」と呼ばれる専門家が，市場に代わって決定している。

また，品質情報についても，構造偽装問題や欠陥不動産に象徴されるように，開示されている情報そのものに信頼が置けないといった問題も指摘されてきた。つまり，不動産市場において流通している情報の中には，情報の量的問題のほかに，情報の正確度 (accuracy) といった質的問題が存在しており，なかでも財の品質に関する不確実性が高い。

このような情報が不足しているという問題は，特に不動産市場で顕著であるが，多くの市場財において，問題の程度の差こそあれ，共通に抱える問題である。市場に出回っている多くの市場財では，使用目的が同じであったとしても，性能や機能面で多くの差別化が図られている。仮に規格や設備が同じであっても使用後の時間が異なれば，質の劣化の程度が異なり同質のものではなくなる。このような性能や機能面での質の違いはその商品の市場価格に反映される。同時に，その市場財，つまり商品独自の性能や機能に対する消費者の評価もまた市場で決まる価格に反映されているといえる。

そのような中で，米国の Zillow 社が，Zesimate と呼ばれるサービスを通じて，全米の不動産価格をビッグデータと AI を用いて査定し，それを公開されたことは，大きな挑戦であったともいえる。

このサービスを実現するための技術とは，不動産価格を形成する特性に応じた価格付けを統計的な手法によって分解し，その結果を用いて測定したい対象の不動産の特性と掛け合わせることで価格を予測しようとするものである。

データマイニング・ビッグデータ・機械学習・AI などといった言葉の変遷はあるが，不動産価格を予測するということにおいては，その本質的な技術は 1970 年代においてヘドニック理論 (第 2 章参照) が登場してから理論的にも，実証的にも大きな進化はあったが，そのような理論が登場する前から，研究分野では実施されてきた。

そのような不動産の価格形成要因を，統計的な技術を用いて分解し，予測することが実用化されるようになってきたのは，コンピューター技術の進化によって可能になってきた。計算能力が飛躍的に向上し始めると，従来は小サンプルで実験されてきた研究が，実用化される道筋が立ち始めてくるのである。とりわけ 1990 年代に入ってから 2000 年代初頭において，回帰分析と呼ばれる伝統的な統計分析を含み，

データマイニングと呼ばれる様々な技術が注目を浴びた。その統計的手法の中核は，回帰分析と共に，クラスター分析，ニューラルネットワークや回帰木 (Regression Tree) などであり，さらにはマーケットバスケット分析，記憶ベース推論 (MBR)，リンク分析や遺伝的アルゴリズムなど，現在の機械学習と呼ばれる分野で利用される多くの手法が既に含まれていた。

当時は，コンピューターの計算能力が整ってくる一方で，「データ資源」の脆弱性から，データマイニングのブームが大きな広がりを見せることはなかった。また，金融ブームが訪れることで，金融工学などが注目されるようになる。そのような中では，データマイニング分野で活躍していた技術者や研究者も，金融の世界へと参入していった。しかし，リーマンショックに端を発した経済危機に陥る中で，金融工学は衰退し，最近では金融と AI とを掛け合わせた FinTec(Financial Technology) と呼ばれる一つの産業へと変容してきた。

不動産業においても，不動産テック，RealTec(Real Estate Technology) と呼ばれる産業が成長しようとしている。その背景には，米国の成功もあるが，不動産に関する情報の整備が進み，その利用可能性が合法・違法関わらず容易になってきたためである。統計的な技術の進化としては，ニューラルネットワークの進化形である深層学習 (Deep Learning) や回帰木が容易に利用できるようになったことも，それを後押ししていると言ってもいいであろう。

不動産価格を予測する技術の応用は，もともとは 1990 年代に米国における不動産ローンの二次市場の発達の技術的なけん引力となった自動不動産価格システムの開発時に進化した。具体的には，かつての Federal Home Loan Mortgage Corporation の Freddie Mac は，アメリカ全土の不動産に対する価格査定を自動的に行うことができる Loan Prospector と呼ばれる製品開発を既に開発していた。そして，そのエンジンには，HNC,Inc のニューラルネットワークに基づいていた。さらに，IBM においても同様の技術で不動産評価の予測に関する研究開発が行われていた。わが国では，1997 年に筆者らの研究チームで同様のシステムを開発し，2000 年代初頭から，大手メガバンクの不動産ローン，アパートローンの自動審査システムとして利用されており，現在もそのシステムは 20 年余り活用され続けている。不動産価格の予測という分野は，機械学習または AI とよばれる技術が古くから適用され，実用化されてきているのである。

不動産テックと呼ばれる産業において，このような不動産価格の予測にとどまらない，多くの技術の応用が登場してきている。本章では，不動産仲介業に焦点を当てて，現在において登場してきているサービスの裏側を支えて技術について整理することを目的とする。

1.2 不動産のマッチング

1.2.1 不動産仲介業の役割

ここで，不動産業の中でも，就業者数といった意味で，最も大きなシェアを持つ産業の一つである不動産仲介業に注目してみよう。消費者が不動産を購入しようとした際には，その情報収集から開始する。不動産情報のポータルサイトなどを通じて，不動産の販売広告を見る。しかし，その広告には，その不動産周辺の住環境，維持管理の経歴，外見からでは判断できない構造強度など，必要な情報が全て網羅されているわけではない。これらの欠けている情報のなかには，実際に物件およびその周辺を注意深く見ること

で初めてわかるものも多い。このように消費者は最も良い物件を求めて自分で探索作業を行わざるを得ないが，それには多くの時間と手間が必要となる。

このような費用を節約するために，不動産仲介業が介在し，またはテクノロジーがそれを軽減するように利用されている。ここでは，買い手がどのような費用を支払っていると考えたらいいのか，といったことから整理しよう。

まず，消費者が売られている物件を全て見て回るなどということは不可能である。そのことは逆に売り手には物件の「本来の価値」（情報が完全で誰も市場支配力をもたない場合の価格）よりも高い価格で不動産を売却することができる可能性がある。強気で臨む売り手は，買い急いでいる消費者が現れることを期待して，不動産に高めの値段をつけて売ろうとするかもしれない。逆に，売り急いでいる売り手は本来の価値よりも低い価格をつけて売ろうとするかもしれない。それでも売却に至るまでにある程度の時間を待たねばならないかもしれない。このように情報の不完全性のもとでは，不動産価格は本来の価値から乖離する可能性があり，消費者は時間と手間がかかっても探索活動をやめるわけには行かないのである。

続いて，売り手である。売り手においても，売却を希望してから契約にいたるまでの時間が必要以上に多くかかったり，契約にいたることができなかったりするといったことが発生する。当然，売り手にとっては，すこしでも高い価格で，かつできるだけ早く売却したい。しかし，売却の目的にもよるが，しばしば期待が高すぎ，またローンの借り換えの都合等なんらかの理由で，当初売り手側が高すぎる下限価格が設定している場合も多い。その場合は，買い希望者が登場したとしても，契約が成立しないケースが発生する。

以上のように，不動産市場では，売り手・買い手ともに高い費用が発生しているため，その社会的な費用を解消するために，不動産仲介業が存在していると言ってもいいであろう。

1.2.2　買い手のコストと売り手のコスト

それでは，「買い手」の探索コストを考えてみよう。情報が完全な市場ならば，不動産の「本来の価値」と売値が一致する。しかし，買い手は真の価格を知ることができないために，探索 (サーチ) をしないといけない。このような費用を測定するためには，サーチ理論が有用である。しかし，伝統的なサーチ理論では，同質的な（homogeneous）財を対象とし（つまり財の品質については情報の不完全性はない），誰がどんな価格をつけているかを知らないという形でモデルを組み立てる (例えば，Turnbull and Sirmans(1993)[1])。しかし現実の不動産市場では価格情報はインターネットで得ることができるが，不動産の品質は均一ではなく，その情報は実際に訪問精査しない限りわからない。

Shimizu. Nishimura and Asami(2004)[2] では，サーチ理論の「価格」を「品質調整済価格」と置き換えて，「価格」は確かに不動産情報誌などを見ればわかるが，「品質調整済価格」は実際に物件を訪問精査しなければわからないと考え，「品質調整済価格」についてサーチ理論を援用した。不動産の広義の品質となる立地条件や建物構造については，買い手はすべて共通の認識を持つとした。従って同一の品質の物件に対しては，もし情報が完全ならすべての買い手が同一の「品質調整済の価格」の値付けをするはずである。

また，現実には不動産品質と同様に，それを探す買い手も異質的（heterogeneous）であり，サーチのコストも本来ならばそれぞれの買い手のサーチコスト合計を考えなければならない。しかし買い手のサーチ

コストの異質性を正面から取り上げるには，膨大な情報と費用がかかる。このような視点を踏まえてマーケティングを考えていかなければならない。また，品質調整済みの不動産価格は，第 2 章において整理する。そうすると，品質調整済み価格がわかっているとすれば，実際の売値と理論価格との乖離，つまり残差項の分布（ヘドニック式の残差）が，超過価格を表すことを意味する。つまり，買い手は，超過価格が存在する限り，実際の市場価格よりも，高い価格で不動産を買わないといけなくなってしまうのである。そのため，買い手は，品質に応じた適切な価格で買うことができる物件を探索し続けることになる。

この「超過価格」について，標準的なサーチ理論の仮定に従って，標準的な買い手は，個別の物件単位の超過価格は知らないものの，その確率分布は知っているものと仮定し，その分布を分布関数 F，確率密度関数 f とすると，買い手のサーチコストを推計することが可能となる。

買い手は，すでにいくつかの物件を見て，その時の最低超過価格が y であったとする。その時に，次の物件をサーチすると，サーチ費用として s がかかるものとする。次のサーチを実行すると，y 以下の超過価格の物件が見つからない（最低超過価格が変わらない）確率は，$1 - F(y)$ である。それ以外の場合では，$x(< y)$ の超過価格が見つかる確率密度が $f(x)$ である。従って，次のサーチを行った場合の超過価格の期待値 $X(y)$ は，

$$X(y) = \int_{-\infty}^{y} xf(x)dx + y[1 - F(y)] \tag{1.1}$$

となる．よって，次のサーチを行う純便益 $B(y, s)$ は，

$$B(y, s) = \{y - X(y)\} - s = \int_{-\infty}^{y} (y - x)f(x)dx - s \tag{1.2}$$

となる。これは，x について単調増加関数であり，f についてのデータがあれば，$B(y, s) = 0$ となる y の値を求められる。この y を以下，y^* で表す。そうすると，最適なサーチ戦略は，サーチをして得られた超過価格が y^* 以下になるまでサーチを続けるというものになる。

この戦略に基づいて，初回のサーチでサーチをやめる確率は $F(y^*)$ となる。また，2 回目でやめる確率は，初回でサーチをやめない確率 $[1 - F(y^*)]$ に，その回にサーチをやめる確率 $F(y^*)$ をかけたものに等しいから $F(y^*)[1 - F(y^*)]$ となる。一般に，ちょうど n 回だけサーチを行う確率 $Q(n)$ は，

$$Q(n) = F(y^*)[1 - F(y^*)]^{n-1} \tag{1.3}$$

であるから，サーチ回数 n の期待値を求めると，

$$\sum_n nQ(n) = \sum_n nF(y^*)[1 - F(y^*)]^{n-1} = \frac{F(y^*)}{1 - F(y^*)} \sum_n n[1 - F(y^*)]^n \tag{1.4}$$

となるが，無限等比級数の公式から

$$\sum_n n[1 - F(y^*)]^n = \frac{1 - F(y^*)}{F(y^*)^2} \tag{1.5}$$

が知られているので，それを代入して整理するとサーチ回数の期待値は $1/F(y^*)$ となる。そこで，買い手のサーチによるコストは，$s/F(y^*)$ となる。問題は，いかに s を求めるかであるが，これは，平均して 1 つの不動産に訪れる時に要する時間コスト（訪問に要する時間に賃金率を乗じたもので代用）に加えて，1 回訪れるに要するその他の費用（交通費用，情報交換費用など）を加えれば良い。

売り手のコストは，もう少し簡単に考えることができる。「売り手」は売買成立までにある程度の時間がかかるため，その間は住戸資産を生産的用途に用いることができず，いわば無駄に所有していることになる。仮に情報が完全な市場では，売り手は自らの物件に関する情報は十分に認識していることから，その品質情報に応じた「本来の価値」で，すぐに売却することが可能となる。その意味で，物件売却までにかかる機会費用は情報の不完全性に伴う売り手のコストとなる。

そこで，売り手の損失としては，その間の機会費用を計上することができる。その計上方法としては，市場滞留時間を T，レンタル価格，つまり賃料を $Rent$ とすれば，

$$Rent \times T \tag{1.6}$$

となる。

また，新古典派の資本理論に基づけば，貸し借りが完全に自由にできる資本ストック市場における均衡では，レンタルコストが資本コストに等しくなる。そこでレンタルコストの代わりに資本コストで機会費用を表すことができる。最終売値価格を P，利子率を r とすると，その物件の市場価値は最終売値価格に近いと考えられることから，機会費用を

$$P \times r \times T \tag{1.7}$$

で近似できる。

1.3　不動産テックによる社会課題解決

以上のような買い手・売り手の費用が発生しているということは，その費用を節約するために，不動産仲介サービスを利用しようとする。その費用を不動産仲介業者に支払ってきたわけであるが，テクノロジーの進化は，その業務を一層高度化・効率化することを可能とする。

まず，買い手にとっても売り手にとっても，市場で成立している価格や家賃は，サーチ行動または売却行動にとって最も重要な情報となる。適正な市場価格がわかれば，売り手はすぐに売却が可能となり，買い手は住宅購入の意思決定を容易にする。

このような品質調整済みの予測価格 (P) を生成することを実現するためには，ヘドニック・アプローチと呼ばれる経済理論的な背景を伴いながら発達してきた理論と分析手法を用いて，不動産の価格・家賃に関するビッグデータが整備され，コンピューターの計算技術と機械学習と呼ばれる計算手法が進化する中では，多くのサービスが実用化されてきている。このようなサービスが一層進化していくためには，データの発生プロセスをも踏まえた経済理論的な背景の深い理解と，不動産のエリア情報をも含む属性データが，一層整備されていくことが求められている。本書では，経済理論的な背景を第 2 章において，そして，機械学習の数学的な基礎を第 3 章において，それぞれ整理した。また，第 4 章では，伝統的な回帰分析から出発し，不動産の価格予測をするための機械学習を含む推計方法の特徴を示した。

このような価格予測をしていく手続きにおいては，立地情報やエリア情報の入手は極めて重要になる。とりわけデータ資源を生成していく技術は，不動産テックのみならず全ての統計分析において欠かせない技術である。不動産分析においては，不動産情報は，住所と建物属性だけが入手できる場合が多く，その場合には，不動産の特徴量の中でも重要な要因となる，「最寄り駅」の特定やそこまでの「距離」，または周辺の商店の集積や学校への近接性などといった情報が必要となる。そのような情報を生成する最も有力

なデータ生成技術として，地理情報システム，または GIS(Geographic Information System) の活用が挙げられる。第 5 章では，不動産テックを推進するためのデータ生成技術としての GIS 技術とそれに付随する統計分析の基礎的な知識を提供する。また，このような GIS の技術を使い，エリア特性を指標化したエリア指標については，第 10 章に紹介する。

第**??**章では，観察可能となった情報を用いた実際の推計方法について具体的なデータと推計手法の適用方法を紹介するとともに，それぞれの手法の特徴を紹介する。第**??**章では，不動産価格の分析として，近年において発達してきた二つの手法を追加する。第一は，ビルダーズモデルと呼ばれる推計技術である。不動産テックと呼ばれる産業で開発されているサービスでは，価格モデルが欠如していることが多い。ビルダーズモデルは，生産関数から出発したモデルであり，そのモデルを適用することにおいて，不動産価格を土地と建物に分離することが可能となる。また，GIS を用いたアドレスマッチングサービスが進化する中では，不動産の特性としての緯度・経度といった位置座標が利用できるようになってきた。そのような空間的な位置によっても，不動産価格は差別化される。また，その価格差は，建物価格には発生することがなく，土地価格においてのみ発生する。そこで，ビルダーズモデルをさらに発展させ，座標位置を用いて，空間的な特性を入れたモデルへの拡張方法を紹介する。

また，不動産市場分析においては，長期優良住宅であればどの程度の価格プレミアムが付くのか，新耐震と旧耐震でどの程度の価格差が産まれているのか，オートロックとそうでない建物で家賃は違うのか，断熱性能を高めたら高い家賃が取れるのか，といった問いに対して，解を求められることがある。このような分析は，計量経済では介入効果の測定といった分野に位置づけられる。近年では，このような問題に対応するために，傾向スコア分析と呼ばれる手法が利用されるようになってきている。この手法は，不動産価格指数の推計にも応用されてきている。第**??**章では，傾向スコアによる不動産市場分析について整理した。

これらの一連のテクノロジーを活用することで，不動産価格を予測することができ，売り手と買い手，または仲介業者もビッグデータとテクノロジーを融合させて，市場価格 (P) を得ることができるようになる。それを情報提供サービスとして提供することができるようになってきたが，これらのサービスにより，「売り手」サイドの空室の機会費用削減とともに，「買い手」サイドでは物件を探索するための機会費用の低減させることができるようになったと言ってもいい。また，家賃を対象として価格を予測したサービスも生まれてきたことから，数式 (1.6) の $Rent$ もまた，市場で容易に観察が可能できるようになった。

このような情報とともに，消費者に対しては，住宅選択が可能な具体的な情報を提供していかなければならない。例えば，部屋の「間取り」情報であったり，「エリア情報」であったりと，不動産を取り巻く包括的な情報を正しく生成し，消費者へと届けなければならないのである。第**??**章では，機械学習の手法を用いて「間取り」を認識する技術を，第 10 章では，エリア情報の生成手法について整理した。米国の不動産テック企業が配信を始めた「Walk Score」と呼ばれるものが注目されているが，第 10 章では，日本で入手可能なエリアに関わるデータ資源を用いて，日本版の Walk Score を推計していくうえでの技術的な背景を紹介した。

さらに，消費者に対しては，市場に流通している不動産物件に関する網羅性が高く正確な不動産物件データベースを提供していかなければ，買い手・売り手のコストも低減させることができない。第**??**章では，データベース・情報アクセスの基礎技術について解説するとともに，深層学習を不動産物件画像に適用する取り組みの事例についても紹介する。

このような情報提供だけでは，まだまだ残された課題も多い。

第 2 章で整理するように，買い手の特性は一様ではないためである。単身者であったり，子育て世帯であったり，または高齢者の夫婦だったりすることもある。そのような場合には，それぞれの特性に応じた情報にだけ意味を持つ。

例えば，子育てのしやすさといった「保育環境」「教育環境」，日常の買い物のしやすさといった「商業集積」なども，すべてが価格に反映されているわけではない。また，水害が発生する頻度や災害に対する地盤強度，さらには，大気等の環境汚染が健康に害を与えることも多く，居住地選択とは無関係ではない。このような情報は，探索して初めてわかる情報であり，後者などについては住んでみてはじめてわかることが多い。さらには，維持管理の経歴や外見からでは判断できない構造強度などは，高度に専門的な知識が必要となるため，探索しても十分に判断できない場合が多い。

様々なテクノロジーの進化が不動産のマッチング効果を高め，市場の潜む費用を低下させるように機能し始めているところであるが，今後のデータ資源の整備とテクノロジーの発達が待たれる分野も多く残されており，今後の発展に期待するとともに，研究開発を進めていかなければならないと考えている。

不動産テックは，上記のような不動産流通分野に限らず，より広範囲で利用されるようになってきた。第**??**章では，官民が持つビッグデータを用いた空き家の予測モデルを紹介した。高齢化の進展に加え，人口減少が進む地方都市では，都市全体が縮退していく中で，所有者がわからない土地が発生するとともに，空き家も増加してきている。しかし，その数を正確に把握することもまた困難であり，さらに将来に発生する空き家を予測するといったことは極めて困難である。しかし，このような問題に対しても，新しいデータ資源の活用が可能となり，テクノロジーが進化する中では，可能としてくる可能性は高まってきている。

さらには，不動産市場は，金融市場と融合する中で，不動産投資信託をはじめとする不動産金融市場もまた，新しい産業として，21 世紀に入ってから大きく成長してきている。第**??**章では，不動産投資信託市場で入手可能なデータ資源を紹介するとともに，応用例を示した。

1.4　良質なデータ資源の重要性:Garbage in garbage out

近年において，不動産分野においても，AI の活用可能性が注目され，不動産 Tec または Real-Tec などともてはやされるようになってきた。しかし，実用化されているモデルを見ると，不動産市場の専門家として看過できないようなモデルも少なくない。

例えば，個別物件単位における売りの仲介として不動産市場に出現してくると考えられる対象物件をあぶりだすためのモデルに，様々な内外のマクロ経済指標を学習させているようなケースもある。また，同様のマクロ経済の集計量で，企業単位での土地の放出を予測するようなモデルなども提案されていたりする。不均一性を考慮する必要があるものの，データ収集の容易さだけで，個別性の強い不動産市場のモデルを構築しているのである。

これは，機械学習に対する過度の期待による弊害であり，これらモデルは汎用性や一般性はないことは専門家であれば判断できる。このような事態をもたらしている原因としては，AI の活用を担当する企業の担当者と開発を担当するエンジニアの双方に，不動産市場分析技術・統計技術の知識のいずれか，または両方が欠如しているためと考える。

統計学では,「いくらゴミを学習させてもゴミしか出てこない (Garbage in garbage out)」という言葉がある。AI などの科学技術の進歩は，必ず市場を進化させる。しかし，誤った技術の使い方は，市場の進化を阻害するだけでなく，その技術の評価を低下させてしまうことにもつながる。不動産市場を一層進化させていくためには，不動産市場と AI 等の技術にも精通した高度不動産人材を育成していくことが急務である。

また，市場分析に欠かせないデータ資源の権利保護と収集手続きも重要な課題である。このような問題は，本著の範囲を超える分野となるものの，そのようなデータ資源への精通もまた，不動産テックを習得していくうえで，極めて重要な知識となってくることも，留意していただきたい。

参考文献

[1] Turnbull, G. K. and C. F.Sirmans, (1993), "Information, Search, and House Prices", *Regional Science and Urban Economics*, Vol.23, pp. 545-557.

[2] Shimizu, C., K. G. Nishimura and Y. Asami (2004), "Search and Vacancy Costs in the Tokyo housing market: Attempt to measure social costs of imperfect information," *Regional and Urban Development Studies*, 16(3), 210-230.

第 2 章

不動産市場分析の理論[*1]

2.1 ヘドニック・アプローチによる不動産価格分析

不動産テックに関連する，わが国の論文や著書を見ると，「ヘドニック・アプローチ」によって価格予測を行ったという記述を見ることが多い。しかし，多くの場合において，単なる回帰分析や機械学習を用いて不動産価格の価格形成要因を分解し，それを束ねる形で価格予測をしているだけであり，厳密な意味でのヘドニック法を適用しているものは極めて少ない。ヘドニック・アプローチとは，ある商品の価格をさまざまな性能や機能の価値の集合体（属性の束）とみなし，統計学における回帰分析や機械学習のテクニックを利用して商品価格を推定する方法である。経済的な理論の裏付けをもって関数形を設定しながら商品価格は分析していく手続きであるために，属性の束からなる方程式で表現され，このような式をヘドニック価格関数とよぶ。ヘドニック・モデルでは，その方程式を解いていくことから，厳密な意味での関数形を設定しない機械学習によって推計されたモデルが，ヘドニック・アプローチによって推計されたという表現は，正しいのかどうかは疑わしいところである。また，ヘドニック価格関数を具体化することは，消費者が個々の機能や性能に対してどの程度の価値を見出しているかを明らかにすることと同じであるために，その推計値によって，各商品が持つ属性の効果を識別していくことができる。

伝統的な価格理論との大きな相違は，一般的な市場財では一物一価の法則が市場分析を行う上での有効な仮定となるが，Lancaster(1966)[3] が分析しているように，この仮定は差別化された商品を扱う上で理論的にも，または実証分析を行う上でも不都合となる。そのような中で，Rosen(1974)[4] はこのような属性の束としての商品価格データが，どのような市場メカニズムで発生するのかを理論的に解明するとともに，実際の価格関数の推計手順も提案した最初の研究であった。不動産市場分析の文脈で考えたときに，Rosen(1974)[4] 以前においても，不動産価格を回帰分析の技術を使って分解する研究が存在していたが，データ発生プロセスをどのように記述するか，識別問題や一致性をどのように考えたらいいのかといった観点から見て，ヘドニック価格関数は正しく理解されていなかった。つまり，現在において，不動産テックと呼ばれる産業界において活用されている，不動産に関わる価格や家賃のデータを単に収集し，機械学習の手法を用いて開発されている多くのシステムにおいては，Rosen(1974)[4] 以前の技術を使って推計しているだけであると言っても言い過ぎではないであろう。そのために，実際の活用において，多くの問

[*1] 本章は，清水千弘・唐渡広志 (2017)[7]『不動産市場の経済経済分析』朝倉書店および清水千弘 (2017)[6],「ビッグデータで見る不動産価格の決まり方」不動産学会誌，120 号,45-51 をもとに，整理したものである。

題を引き起こしてしまっている。

Rosen の研究は，Tinbergen(1959)[11] の提起による差別化された生産物の市場均衡理論を発展させたものである。商品供給者のオファー関数（offer function），商品需要者の付け値関数 (bid function) およびヘドニック価格関数の構造との間の関係を厳密に検討し，商品の市場価格を消費者および生産者の行動から特徴づけている。実際に実証分析を行ってはいないものの，計量経済学的な推定手順についての概略も示している。Witte, Sumka and Erekson (1979)[12] は Rosen 理論を元に具体的に実証分析した研究である。

Rosen 理論では，単純化されたケース（生産者を同質に扱うケース）においてすら，ヘドニック価格関数から選好や技術の構造を識別するためには非常に複雑な解析を必要とする。Epple(1987)[1] は多数の消費者と生産者を想定した上で，Rosen 理論を発展させた計量経済モデルを定式化している。Rosen 理論の問題点は，需要と供給からなる構造方程式において，同時性バイアスが生じるケースを排除できない点である。もし，重要な属性が観察されておらず，それらが観察された属性と相関している場合には，均衡におけるヘドニック価格関数の観察された属性の係数推定量には不偏性もなければ一致性もない。ヘドニック・アプローチを不動産市場に適用しようとした場合には，不動産価格の複雑さから，この問題は深刻な問題になってしまう。具体的には，不動産価格は，立地や建物といった特性だけでなく，周辺環境といったエリア特性までもが価格に影響をもたらすことから，それら変数も含めて考慮しなければならないためである。そのため，分析者は常に必要な属性を観察できるわけではなく，利用できる変数が限定的になってしまうという問題は，ヘドニック・アプローチの利用上最も注意すべき問題点の一つである。

この点に関して，Epple のモデルは観測誤差を正確に処理できるヘドニック価格関数を提起するアプローチとなっている。ただし，このアプローチは効用関数に次の先見的な仮定をおいた上で，閉じた市場均衡におけるヘドニック価格関数を導き出し，推定を行うことになる。

- 効用関数の関数型はすべての消費者について同質である。ただし，選好パラメータが正規分布に従う（共分散は非対角要素が 0 の対角行列）。
- 消費者の効用関数は属性変数が加法分離的で 2 次形式である。
- 差別化された商品の供給が外生的に与えられている。

上記は経済主体間の相互作用がないこと，および市場均衡におけるヘドニック価格関数が描写できるように実現可能な関数型を想定しており，決定的な強い仮定である。

2.2　ヘドニック価格関数の推定

2.2.1　付け値関数

ヘドニック・アプローチの理論的枠組みを Rosen(1974)[4] および Epple(1987)[1] にしたがって整理する。ここでは，$K \times 1$ の属性ベクトル $\boldsymbol{X}$ （属性の束）からなる不動産の需要を考える。ここでいう属性とは，不動産価格を差別化している，「最寄り駅までの距離」，「都心までの距離」，「大きさ」，「建築後年数」などである。

属性の束で示される不動産の市場価格関数を $P(\boldsymbol{X})$ としよう。消費者の効用関数を $u(c, \boldsymbol{X}; \boldsymbol{A})$ と書

く．ここで， c は価格が 1 に基準化された価値尺度財（スカラー）， $\boldsymbol{A}$ は消費者個人を特徴付ける選好パラメータのベクトルである。消費者の所得を I とするとき，予算制約式は $I = P(\boldsymbol{X}) + c$ となる。消費者の所得と選好の分布を確率密度関数で考え，これを結合確率密度関数 $f(I, \boldsymbol{A})$ で表わす。

与えられた予算制約のもとで， $(c, \boldsymbol{X})$ について効用を最大化するとき，次の最適化条件が得られる。

$$\frac{\frac{\partial}{\partial \boldsymbol{X}} u(I - P(\boldsymbol{X}), \boldsymbol{X}; \boldsymbol{A})}{\frac{\partial}{\partial c} u(I - P(\boldsymbol{X}), \boldsymbol{X}; \boldsymbol{A})} = P_{\boldsymbol{X}}(\boldsymbol{X}) \tag{2.1}$$

ここで，$P_{\boldsymbol{X}}$ は属性の 1 階微分を示している。すなわち，最適な属性の選択は合成財に対する個々の属性の限界代替率が不動産市場価格の限界的価値に等しいところで決定される。不動産市場価格の限界的価値は需要者がその属性に対して支払ってもよい (willingness to pay) と考える属性の価値に等しくなっている。したがって，個々の属性価値を調べるためには，市場価格関数 $P(\boldsymbol{X})$ における各属性の微係数を知る必要がある。

需要者が不動産に対して支払ってもよいと考える最大の価格のことを付け値（bid price）とよぶ。これを θ という記号で定義する。いま，ある一定の効用水準 u^* のもとで選択された属性の束が $\boldsymbol{X}^*$ であるとき

$$u(I - P(\boldsymbol{X}^*), \boldsymbol{X}^*; \boldsymbol{A}) = u^* = u(I - \theta, \boldsymbol{X}; \boldsymbol{A}) \tag{2.2}$$

である。したがって，付け値と属性の関係を示す付け値関数は，この効用関数のもとで $\theta = \theta(\boldsymbol{X}, I, u; \boldsymbol{A})$ と陽表的に示すことができる．すると，効用が最大化されるとき，任意の $f(I, \boldsymbol{A})$ のもとで

$$P_{\boldsymbol{X}}(\boldsymbol{X}^*) = \frac{\partial}{\partial \boldsymbol{X}} \theta(\boldsymbol{X}^*; u^*, I, \boldsymbol{A}) \tag{2.3}$$

でなければならない。このことは，市場価格関数の勾配が所得の限界効用に対する属性の限界効用に等しいだけでなく，付け値関数の勾配にも等しくなっていなければならないことを示している。ヘドニック・アプローチとは，不動産価格を不動産のさまざまな属性に回帰させたモデルを推定することによって，各属性の価値を予測する手法である。ヘドニック価格関数を 1 次近似すると

$$P(\boldsymbol{X}) \cong \tilde{P} + \sum_k \frac{\partial P(\tilde{\boldsymbol{X}})}{\partial X_k} X_k \tag{2.4}$$

であるから，ヘドニック価格関数はさまざまな属性の限界的価値の線型結合式とみなせる。例えば，第 i 属性ベクトル $\boldsymbol{X}_i = (X_{i1}, X_{i2}, \cdots, X_{iK})$ に不動産市場価格 P_i を回帰させた古典的な線型回帰モデルは

$$P_i = \beta_0 + \sum_{k=1}^{K} \beta_k X_{ik} + u_i \, (i = 1, 2, \cdots, n) \tag{2.5}$$

と表現される。ここで， $\beta_1, \beta_2, \cdots, \beta_K$ は不動産属性の限界的価値を示す未知パラメータであり，u_i は撹乱項である．しかしながら，この線型近似式だけでは多数の消費者の選好を反映したヘドニック価格関数かどうかを識別する手がかりはない．生産者の行動も考慮に入れてモデルを閉じて，均衡状態のヘドニック価格関数を描写する必要がある。

2.2.2 市場均衡とヘドニック価格関数

(2.5) の左辺は不動産市場の需給均衡で決まる市場価格であるから，生産者の行動も描写しなければモデルを閉じることができない。不動産のように差別化された商品の費用関数を $C(\boldsymbol{X}, M; \boldsymbol{B})$ とする。ここで，M は建設される不動産の数を示しており，$\boldsymbol{B}$ は各生産者を特徴づけるパラメータ・ベクトルである。$\boldsymbol{B}$ の分布は確率密度関数 $g(\boldsymbol{B})$ で与えられているものとする．生産者は不動産市場価格を所与として，次の利潤を最大化する属性の束を決定する．

$$\pi = P(\boldsymbol{X})\, M - C(\boldsymbol{X}, M; \boldsymbol{B}) \tag{2.6}$$

生産者の行動は，短期か長期かによっても異なり，Rosen が示したように短期には 2 パターンの状況を想定できる．

- 生産者にとって M だけが可変的な短期経済
- M および $\boldsymbol{X}$ のどちらも可変的な短期経済

長期の経済では固定資本（費用関数に明示されていない）も可変的になり，参入・退出の自由が認められる。ここでは，二つめの短期経済を想定して，次の最適化条件を得る。

$$P_{\boldsymbol{X}}(\boldsymbol{X}) = \frac{1}{M} \cdot \frac{\partial}{\partial \boldsymbol{X}} C(\boldsymbol{X}, M; \boldsymbol{B}) \tag{2.7}$$

$$P(\boldsymbol{X}) = \frac{\partial}{\partial M} C(\boldsymbol{X}, M; \boldsymbol{B}) \tag{2.8}$$

(2.7) より各生産者は属性の限界的価値が不動産 1 単位あたりの属性の限界費用に等しく，そして，(2.8) より与えられた属性の束のもとで，不動産の市場価格は任意の生産技術をもつ生産者の不動産生産限界費用に等しくなければならな。このとき達成される最大利潤はパラメータ $\boldsymbol{B}$ によって異なる。

ある一定の利潤 π^* のもとでの最適な属性の束 $\boldsymbol{X}^*$ と生産個数 M^* を選択しているものとしよう。このとき，生産者が提示できる最低の価格（オファー価格）を φ という記号で表わす．すなわち，

$$\varphi M - C(\boldsymbol{X}, M; \boldsymbol{B}) = \pi^* = P(\boldsymbol{X}^*)\, M^* - C(\boldsymbol{X}^*, M^*; \boldsymbol{B}) \tag{2.9}$$

である。この式は，一定の π^* のもとで φ が $(\boldsymbol{X}, M)$ とどのような関係を持つのかを示している。(2.9) より，$\varphi = \partial C(\boldsymbol{X}, M; \boldsymbol{B}) / \partial M$ であるから，これを M について解き，利潤定義式に代入すると，$\pi^* = \varphi \tilde{M}(\boldsymbol{X}, \varphi; \boldsymbol{B}) - C(\boldsymbol{X}, \tilde{M}(\boldsymbol{X}, \varphi; \boldsymbol{B}); \boldsymbol{B})$ が得られる．すなわち，この関係より，オファー関数は $\varphi = \varphi(\boldsymbol{X}; \pi^*, \boldsymbol{B})$ と書くことができる。(2.7) より，利潤が最大化されているとき

$$P_{\boldsymbol{X}}(\boldsymbol{X}^*) = \frac{\partial}{\partial \boldsymbol{X}} \varphi(\boldsymbol{X}^*; \pi^*, \boldsymbol{B}) \tag{2.10}$$

でなければならない。

$\boldsymbol{X}$ に対応したあらゆるタイプの不動産の需要と供給とが等しくなるところで市場均衡が成立し，市場価格 $P(\boldsymbol{X})$ が得られる。(2.3) と (2.7) より，属性の付け値関数とオファー関数との接線の軌跡として均衡における市場価格 $P(\boldsymbol{X})$ を表わすことができる。すなわち，市場をクリアする価格関数は消費者の付

け値関数と生産者のオファー関数との包絡線でなければならない。図 2.1 は第 1 番目の属性 X_1 に関する付け値関数とオファー関数の接線上に市場価格が成立していることを示している。曲線 $P(X_1, \boldsymbol{X}^*_{-1})$ は，X_1 以外の属性ベクトル $\boldsymbol{X}_{-1}$ が $\boldsymbol{X}^*_{-1}$ において最適化されているとき，さまざまな消費者と生産者との間で成立する市場価格の軌跡を示している。

Epple(1987)[1] が指摘したように，市場をクリアするヘドニック価格関数は消費者の所得と選好の確率分布 $f(I, \boldsymbol{A})$ と生産者のパラメータ分布 $g(\boldsymbol{B})$ に依存して決まる。もし，生産者が 1 タイプしか存在しなければ，限界費用関数そのものが市場価格関数になる。限界費用と付け値関数の傾きとが等しくなるところで市場がクリアするので，その包絡線は 1 生産者の限界費用関数に一致するからである。

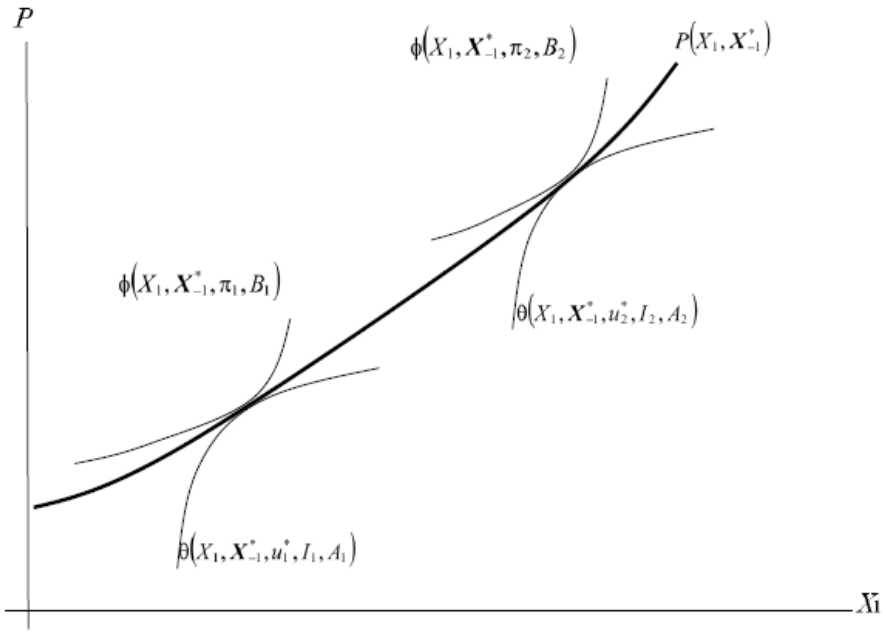


図 2.1　属性 X_1 に関する付け値関数，オファー関数，市場価格関数

2.3　不動産価格の分解と予測

不動産テックの分野では，不動産価格を予測するサービスが提供されるようになってきている。そのようなサービスの裏側では，広い意味でのヘドニック・アプローチを用いて，回帰分析や機械学習のテクニックを用いて，市場で観察された不動産の価格・家賃と不動産の価格を決定する属性データを用いて，価格を分解し，そして予測したい不動産の属性データを取得したうえで予測していく。それでは，市場で観察されている不動産情報とはどのように観察することができるのであろうか。

不動産価格については，しばしば新聞などでも，その変化が報道されることがある。前年と比較して相場が上がった，または下がったとか，ある地域と比較して，分析対象の地域が，どの程度高いまたは低いといったことがしばしばみられる。そのような相場の水準や変化は，本来はどのように比較しないといけないのであろうか。

不動産は，同一の財が存在しないという特性を持つことから，また，すべての不動産がいつも取引されているわけではないことから，異なる場所で異なる属性を持つ不動産が取引されている。不動産市況が好調にもかかわらず，市場で観察される不動産価格が下落しているということもあるが，その背後には，良質な不動産が市場で取引されてしまい，質の低い不動産が，そのあとに取引されれば，市況が好調であったとしても，実際に取引される不動産の価格の平均は低くなってしまうということもある。

そのようななかで，不動産の専門家は，相場の水準・変化を見るために，異なる物件の相違や異なる二時点の二つの不動産価格に関しての分布を頭の中で想定している。他の経済市場の財・サービスとの対比において，不動産の専門家が考慮しないといけないのが，前述のように，不動産価格の分布は不動産の性能や属性によって変化するということである。不動産価格は，最寄り駅からの距離などの交通利便性や，同じ場所に立地する不動産であったとしても大きさや建築後年数，または構造によって価格が変化するため，そのような相違を定量的に制御しなければならない。二つの異なる空間または時間の価格の相違を見ようとした場合には，品質を調整しなければならないのである。それでは，時間に関しての統計的な品質の調整方法を説明しよう。ここで時間を空間と読み替えれば，異なる空間間で比較することができることを意味する。

まず， $F_1(p)$ を第 1 期の価格 $(P1)$ の累積分布関数 (CDF) とすると，不動産属性 (z) といった条件付きの価格の分布は $F_1(p \mid z)$ と表すことができる。この時，価格 $F_1(p)$ と属性 $F_1(p \mid z)$ の関係は (2.11) 式のようになる。

$$F_1(p) = \int_{-\infty}^{\infty} F_1(p \mid z)\, u_1(z)\, dz \tag{2.11}$$

$u_1(z)$ は不動産価格を構成する属性 z の分布である。同様に，$F_2(p)$ および $F_2(p \mid z)$ を第 2 期の不動産価格の属性 $u_2(z)$ に対応した不動産価格の累積分布関数とする。そうすると， $F_1(p)$ から $F_2(p)$ の価格分布の変化は，(2.12) 式のようになる。

$$F_1(p) - F_2(p) = \int_{-\infty}^{\infty} [F_1(p \mid z) - F_2(p \mid z)] u_1(z) dz + \int_{-\infty}^{\infty} F_2(p \mid z)[u_1(z) - u_2(z)] dz \tag{2.12}$$

(2.12) 式の右側をみれば，第一項が不動産属性 z のもとでの品質調整済み不動産価格の差を表し，第二項がそれぞれの時点の不動産属性の相違を意味する。つまり，一般的に市場で観察される二次点の価格の分布の相違は，「価格の変化」+「属性の変化」といった二つの要素から観察されることになる。つまり，実際には第 1 期から第 2 期に対して価格が下落していたとしても，その変化は価格が変化したわけではなく，最寄り駅から遠い物件や築年が古い物件などが中心に取引され，属性の変化によって価格分布が下落しているように見えることもある。そうすると，二つの不動産価格の分布，つまり「相場」を比較しようとした場合には，この第二項である不動産属性の相違を取り除いたうえで価格を比較していかなければならないことがわかる。つまり，(2.13) 式のように同じ属性 z のもとでの価格の相違を見なければならない。

$$\int_{-\infty}^{\infty} [F_1(p \mid z) - F_2(p \mid z)] u_1(z) dz \tag{2.13}$$

近年，内外において実用化されている価格予測システムは，実際には，その価格特性 z に対応した価格ベクトルを推計しているにすぎない。また，価格の変化については，品質調整済みの価格分布の中央値または平均値を示している。具体的には，次の手続きで計算することができる。

$Q_i^\theta(p \mid z)$ を価格の累積分布 $(F_i(p \mid z))$ の第 θ -番目の分位点とする $(\theta \in (0,\, 1))$。これを次のように条件付き分位 (conditional quantiles) として定義する。

$$Q_i^\theta(p \mid z) = z\beta_i(\theta) \tag{2.14}$$

条件付き分位は，様々な不動産属性の加重平均として考えられる。ここでは属性価格 $\beta_i(\theta)$ は，θ 点の価格水準に依存するものと考えればよいため、各分位点の回帰係数と考える。まず，第一期の価格 $(P1)$ の $\beta_1(\theta)$ を推計するために， P_1 を用いた分位点ごとの回帰を行うことで，推定統計量の $\beta_1(\theta)$ を得る。そうすると，不動産属性 z を所与とすれば， $p = z\hat{\beta}_1(\theta)$ によって $F_1(p \mid z)$ が計算される。 $F_1(p \mid z)$ の推計値を $\hat{F}_1(p \mid z)$ とする。同様の方法で第二期の価格 P_2 の条件付き価格の累積分布 $F_2(p \mid z)$ の推計値は， $\hat{F}_2(p \mid z)$ ，となる。そうすると z に関して積分することによって，次のように表現できる。(Shimizu，Nishimura and Watanabe(2016)[10])

$$\begin{aligned} \hat{F}_1(p) &\equiv \int_{-\infty}^{\infty} \hat{F}_1(p \mid z)\, u_1(z)\, dz; \\ \hat{F}_2(p) &\equiv \int_{-\infty}^{\infty} \hat{F}_2(p \mid z)\, u_2(z)\, dz \end{aligned} \tag{2.15}$$

そうすると，実際の計算においては，数式 (2.12) は，次のように書き換えることができる。

$$\hat{F}_1(p) - \hat{F}_2(p) = \int_{-\infty}^{\infty} [\hat{F}_1(p \mid z) - \hat{F}_2(p \mid z)]u_1(z)dz + \int_{-\infty}^{\infty} \hat{F}_2(p \mid z)[u_1(z) - u_2(z)]dz \tag{2.16}$$

相場の水準や変化を予測するサービスは，それぞれの不動産に関する属性に対する係数または重みを推計し，時間の変化に関する係数などを用いて，それをわかりやすい形で消費者に提供されているだけである。

2.4 不動産価格の実際の推計

不動産価格に対して，交通利便性や規模・建築後年数等の相違などの特性に応じた係数または重みを計算する技術は，回帰分析に代表されるように古くから活用されてきた。近年において，再度注目されている機械学習と呼ばれる手法の中でも，中心的な役割を担い，多くの分野で実用化されているのがニューラルネットワークとよばれる技術が進化した深層学習 (Deep Learning) や回帰木と呼ばれる手法である。そのような手法は，どの程度の予測力を持つのであろうか。

ニューラルネットワークや回帰木と呼ばれる推計手法は，そもそも伝統的な回帰分析などと発達するモチベーションが異なっていた。それらの技術は，「機械は人間を超えることができるのか？」という問いを意識して進化してきた。この研究は，1940 年代に本格化し，フォン・ノイマンによる直列処理による計算機とともに，並列処理のそれに対する研究という大きな二つの流れのなかで，計算技術の問題として行われてきたのである。

ニューラルネットワークとは，人間の脳神経細胞の並列処理システムを模して開発がすすめられた。人間の脳神経細胞 (neuron:ニューロン) は，細胞体・樹状突起・軸策の 3 つの部分から構成されており，さらに軸策の末端にはシナプス (synapse) という部位があり，ここを通じて各細胞の情報伝達が行われる。

そして，脳神経細胞の情報伝達は，軸策・樹状突起の結合部で行われてあり，それはシナプス結合と呼ばれている。このような脳神経細胞の情報伝達を模して開発されたのがニューラルネットワークなのである。その詳細は，清水 (2016)[5] または，本書の第 3 章，4 章，**??**章をご覧いただきたい。

また，このようなクロスセクショナルな価格を予測するという行為とともに，価格の時間的な変化をリアルタイムに測定したいということも多い。一般的に，不動産専門家が相場の水準や変化を大きく見誤るのは，市場の転換点である。そして，従来は，市場の変化を公示地価に代表されるような不動産鑑定士によって決定された価格によって観察されることが多かったために，しばしば見間違えることがあった。Shimizu and Nishimura(2006)[8] における統計実験では，実際の取引価格と公示地価といった不動産鑑定価格とのかい離を時間的に調べている。その結果を見ると，1990 年代のバブル期には，不動産鑑定価格は実際の市場価格の半分から 6 割程度であり，バブル崩壊後には，2 割程度高い価格がつけられていたことを示している。その最も大きな原因が，相場を決定する技術よりも，情報の入手速度と選択技術に起因していると結論付けている。

具体的には，従来，不動産鑑定士がその価格決定において利用可能な取引事例と呼ばれるデータ基盤は，アンケート調査に基づくことから 3 か月から半年程度遅れて入手される。さらには，その網羅性も 3 割程度と低い。そのため，高い価格決定技術を有していても，データ基盤の脆弱性からその専門性を生かすことができていない。そのような情報入手の時間的なラグや網羅性の低さから，系統的な誤差が生まれてしまうのである。

一方で，近年における IOT 技術の発達によって，一週間，または数日，場合によっては数時間の時間的な粒度の中で不動産価格に関する大規模データ基盤が更新することができるようになってきた。また，地理情報基盤の整備も進化していることから，実際に人間によって調査を行わなくても，不動産に関する情報だけでなく，地域情報も入手が可能である。また，サイトに対するログも利用することができるようになってきたことで，消費者が求めている各属性別の選好を，その解析を通じて得ることも可能となった。

そうすると，そのように時間的に粒度の細かいデータ基盤をどのように学習させ，リアルタイムに価格を決定することができる技術があるかどうかということが課題になる。その問題に対応するために，Hill, Scholz, Shimizu and Steurer (2018)[2] では，東京とシドニーのデータを用いて，一定の精度を担保しつつ週次単位で予測していくことが可能であることが示された。具体的には，その週単位で得られたデータを用いた価格推計は極めてボラティリティが高くなり，実用化が困難であるものの，一定の期間を重複されることで，時間的にも安定した価格査定が実現できることを示している。

実際の推計手順を示せば，ある週単位での期間 $1, 2, \cdots, T$ 期のうちの r 期からはじまる τ 期間を $[r, r+\tau-1]$ のように表すと，$[1, \tau]$ のようにそれぞれの期のデータを用いて推計するのではなく，$[2, \tau+1] \sim [r, r+\tau-1] \sim [T-\tau+1, T]$ といったように，一定の期間を重複させつつ逐次的に適用することで時間的な安定性を持つことが示されている。つまり，移動平均のように，一定の期間の情報を共有しつつ，新しく出現する情報を追加しながらモデルを推計していくのである。これにより市場構造の逐次的な変化をパラメータに反映させることが可能となる (Shimizu, Nishimura and Watanabe(2010)[9])。

これは，一つの手法に過ぎないが，タイムリーかつ大規模なデータ基盤の整備によって，従来において不動産の専門家が直面していた課題が克服され，より正確な相場の決定やその粒度の細かい相場の時間的な変化を，機械学習などによって推計することができるようになってきたのである。時系列的な不動産価格の測定方法の詳細は，第**??**章を参照されたい。

参考文献

[1] Epple, D., (1987), "Hedonic Prices and Implicit Markets: Estimating Demand and Supply Functions for Differentiated Products", *Journal of Political Economy*, Vol.95, pp.58-80.

[2] Hill, R, M. Scholz, C. Shimizu and M.Steurer (2018) "An Evaluation of the Methods Used by European Countries to Compute their Official House Price Indices," *Economie et Statistique* n 500-501-502, 221–238.

[3] Lancaster, K., (1966), "A new approach to consumer theory", *Journal of Political Economy*, Vol.74, pp.132-157.

[4] Rosen, S., (1974), "Hedonic Prices and Implicit Markets, Product Differentiation in Pure Competition", *Journal of Political Economy*, Vol.82, pp.34-55.

[5] 清水千弘 (2016),「市場分析のための統計学入門」朝倉書店.

[6] 清水千弘 (2017),「ビッグデータで見る不動産価格の決まり方」不動産学会誌, 120 号,45-51.

[7] 清水千弘・唐渡広志 (2017)『不動産市場の経済経済分析』朝倉書店.

[8] Shimizu, C. and K.G.Nishimura, (2006), "Biases in Appraisal Land Price Information: The Case of Japan", *Journal of Property Investment and Finance*, Vol.26, No.2, pp.150-175.

[9] Shimizu, C., K. G. Nishimura and T. Watanabe (2010), "House Prices in Tokyo - A Comparison of Repeat-sales and Hedonic Measures-," *Journal of Economics and Statistics*, 230 (6), 792-813.

[10] Shimizu,C, K.G.Nishimura and T.Watanabe(2016), "House Prices at Different Stages of Buying/Selling Process ," *Regional Science and Urban Economics*, 59, 37-53.

[11] Tinbergen, J., (1959), "On the theory of income distribution", in: L.M.K.L.H. Klaasen and H.J. Witteveen, eds, *Selected Paper of Jan Tinbergen* (North-Holland, Amsterdam).

[12] Witte, A. D., H. Sumka and J. Erekson, (1979), "An Estimate of a Structural Hedonic Price Model of the Housing Market: An Application of Rosen's Theory of Implicit Markets", *Econometrica*, Vol.47, pp.1151-72.

第 3 章

機械学習の数理

3.1 機械学習とは

今日の急速な AI 技術進展の背後には機械学習の進展がある。本章は機械学習の数学的な基礎を概観することを目的とする。直感的な理解を目指しつつも数理的な背景が分かるように記述をした。 機械学習の数理についてはすでに Bishop (2011)[1]，Murphy (2012)[4]，Goodfellow et al。(2016)[2] のような定評のある成書やその日本語訳も出版されている。本章はそういった本格的な本を読む前に，機械学習の「あらすじ」を把握する目的で読んでいただければ幸いである。

データに潜んでいる法則性を見つけ出し，それに基づいた予測や判断を行うためのアルゴリズムの総称を機械学習という。Mitchell (1997)[3] は機械（コンピュータプログラム）が学習をするとは，経験 E を通じてタスク T のパフォーマンス P が向上すること，と定義している。 この定義によれば，機械学習の基本的な構成要素はタスク T， パフォーマンス尺度 P，経験 E ということになる。タスク T とは例えば，不動産の価格予測，画像認識であり， 経験 E とは過去に取得された不動産価格やその不動産の属性データ，画像データである。パフォーマンス尺度 P は例えば，予測された不動産価格と真の値との差や画像認識の正答率のような，誤差を計測する尺度である。機械学習にはいくつかの分類があるが，ここでは教師あり学習，教師なし学習の 2 つを紹介する。

教師あり学習では，まず N 個のデータのペア $\{(\mathbf{x}_i, y_i)\}_{i=1,2,\ldots,N}$ が与えられる[*1]。 これらを訓練データと呼ぶ。ここで $\mathbf{x}_i$ は特徴量と呼ばれる入力データ，y_i は $\mathbf{x}_i$ に対応する正解（またはラベル）と呼ばれる出力データを表す[*2] 。教師あり学習では訓練データから特徴量 $\mathbf{x}$ と正解 y の対応関係を表す関数 f （$y = f(\mathbf{x})$）を推定することが目標となる。 y が連続的なデータのとき，このタスクを回帰という。 例えば，y が不動産価格で，$\mathbf{x}$ が最寄り駅からの距離や建築年数などの属性データであるとき，$\mathbf{x}$ から y を推定する問題が回帰である。 一方，y が離散的なデータ，例えば y が 0 または 1 の値をとるとき，このタスクを分類という。 例えば，$\mathbf{x}$ が動物の画像データであり，y が犬か（$y = 1$），そうでないか（$y = 0$）を表すような場合である。 機械学習の多くのアルゴリズムでは関数 f を直接推定するのではなく， パラメータ化された関数 $f(\mathbf{x}; \mathbf{w})$ を考え， パラメータ $\mathbf{w}$ を推定する問題に帰着させる。 本章でもパラメータ化された関数の推定のみを扱う。

[*1] 数学的表記については章末を参照のこと.

[*2] 正解 y_i はベクトル $\mathbf{y}_i$ であってもよい.

パフォーマンス尺度 P は何らかの誤差によって定義される。この誤差を表す関数を損失関数と呼ぶことにしよう[*3]。 パラメータ化された関数 $f(\mathbf{x};\mathbf{w})$ を考えるときには， 損失関数はパラメータ $\mathbf{w}$ の関数 $L(\mathbf{w})$ となる。 モデルを $y = f(\mathbf{x};\mathbf{w})$ としたときの，個々の訓練データ $(\mathbf{x}_i, y_i)$ の誤差を関数 l を用いて $l(\mathbf{x}_i, y_i; \mathbf{w})$ と表現すれば，損失関数は

$$L(\mathbf{w}) = \frac{1}{N}\sum_{i=1}^{N} l(\mathbf{x}_i, y_i; \mathbf{w}) \tag{3.1}$$

と書くことができる。例えば，誤差を二乗誤差により定義すると，

$$L(\mathbf{w}) = \frac{1}{N}\sum_{i=1}^{N} \{y_i - f(\mathbf{x}_i; \mathbf{w})\}^2 \tag{3.2}$$

が損失関数となる。そして，最適化問題

$$\min_{\mathbf{w}} \ L(\mathbf{w})$$

の解 $\mathbf{w}^*$ によって得られた関数 $f(\mathbf{x};\mathbf{w}^*)$ が推定された関数となる。 新たな特徴量 $\mathbf{x}'$ が得られたとき，$\hat{y} = f(\mathbf{x}';\mathbf{w}^*)$ を正解の推定値として予測や判断を行う。 ここで，損失関数 (3.1) はモデル $y = f(\mathbf{x};\mathbf{w})$ の訓練データへの当てはまりのよさを計測していることに注意する。 機械学習の本来の目的は未知のデータを含むあらゆるデータに対してよいパフォーマンスをもつモデルを推定することにある。すなわち，

$$L_{\text{true}}(\mathbf{w}) = E[l(\mathbf{x}, y; \mathbf{w})] \tag{3.3}$$

を最小にする $\mathbf{w}$ を求めることにある。ここで，(3.3) における $E[\ ]$ は $(\mathbf{x}, y)$ を生成している確率分布のもとでの期待値である。(3.3) によって計測する誤差を汎化誤差という。 一方，(3.1) によって計測する誤差は訓練誤差と呼ばれる。機械学習の本来の目的は汎化誤差の最小化である。しかし $(\mathbf{x}, y)$ を生成している確率分布を厳密に知ることは実際には不可能であり，したがって汎化誤差を計測することも実際には不可能である。よって，訓練誤差を汎化誤差の近似として最小化を行わざるを得ない。しかし，汎化誤差を意識することは過学習を防ぐためにも重要である。過学習については，後の節で触れる。

本章は教師あり学習に焦点を当てるが，教師なし学習についても簡単に触れておく。教師なし学習では，はじめに N 個のデータ $\{\mathbf{x}_i\}_{i=1,2,\ldots,N}$ が与えられる。 教師あり学習と違い，教師なし学習では正解データが与えられない。ここでのタスクはデータ $\mathbf{x}_i$ の背後に潜んでいる構造を探し出すことである。例えば，教師なし学習の問題には，データを類似したいくつかのグループに分類するクラスタリング，高次元データを可視化するための次元削減などがある。

3.2 勾配降下法

機械学習においては損失関数 $L(\mathbf{w})$ の最小化問題

$$\min_{\mathbf{w}} L(\mathbf{w}) \tag{3.4}$$

[*3] 誤差関数やコスト関数と呼ぶこともある．

を解くことが鍵となる。この節では最小化問題の解法について述べる。

もし $\mathbf{w}^*$ が (3.4) の解であるならば

$$\nabla L(\mathbf{w}^*) = 0 \tag{3.5}$$

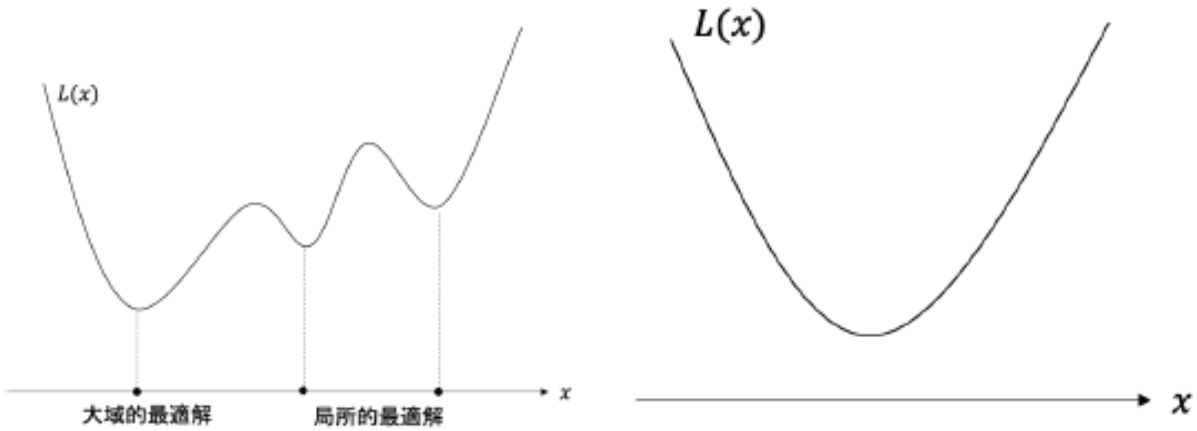
を満たす。ここで $\nabla L(\mathbf{w})$ は勾配ベクトルで

$$\nabla L(\mathbf{w}) = \left(\frac{\partial L}{\partial w_1}(\mathbf{w}), \frac{\partial L}{\partial w_2}(\mathbf{w}), \ldots, \frac{\partial L}{\partial w_D}(\mathbf{w}) \right)^T$$

により定義される。ここで，D はベクトル $\mathbf{w}$ の要素数を表す。 $\mathbf{w}$ が 1 次元の場合は通常の微分 $L'(w)$ である。しかし，ある $\mathbf{w}$ が (3.5) を満たしたとしても，それが最小化問題 (3.4) の解になるとは限らず，局所的最適解になっている可能性がある（図 3.1(a)）。 局所的最適解に対して，(3.4) の解を大域的最適解と呼ぶ。関数 L が凸関数であるときには，条件 (3.5) を満たす $\mathbf{w}$ は大域的最適解になる。 関数 L が条件

$$L(t\mathbf{w}_1 + (1-t)\mathbf{w}_2) \leq tL(\mathbf{w}_1) + (1-t)L(\mathbf{w}_2),\ \forall t \in [0,1],\ \ \forall \mathbf{w}_1, \mathbf{w}_2$$

を満たすとき，凸関数であるという（図 3.1(b)）。いくつかのモデルでは損失関数は凸関数となるが，ニューラルネットワークなどでは損失関数が必ずしも凸にならないので，注意が必要となる。非凸関数に対して (3.4) を解くことは一般には難しく，通常は局所的最適解を求めることで満足することが多い。以下でも，局所最適解を求める手法のみを考える。



(a) 大域最適解と局所最適解　　(b) 凸関数

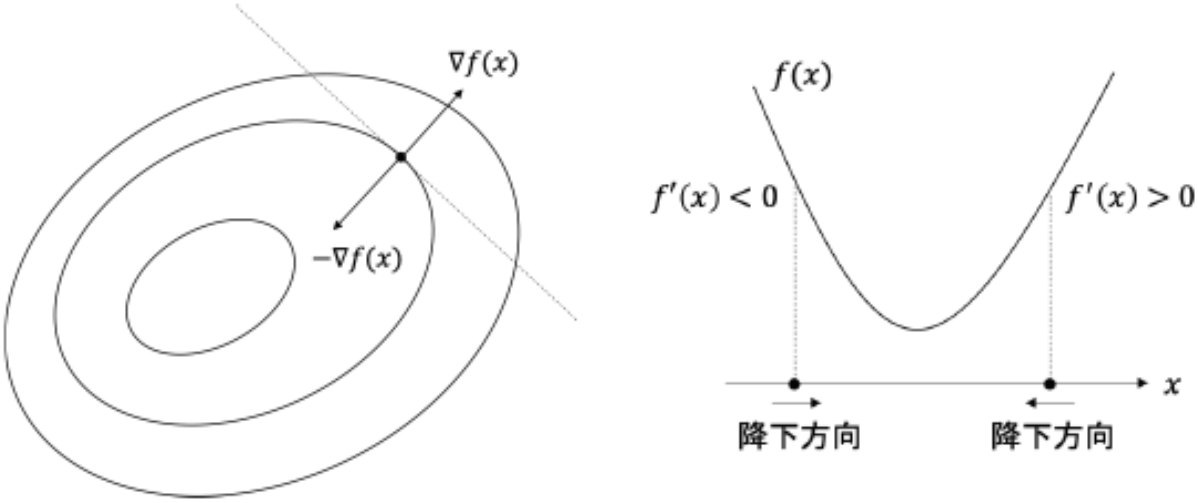
図 3.1　最小化問題の解

線形回帰の場合など $\nabla L(\mathbf{w}) = 0$ が解析的に解ける， すなわち $\mathbf{w} = \cdots$ という表現を得られることがある。 しかし，機械学習ではデータの次元が非常に大きく，解析的表現による解法は実用的ではないことが多い。 そこで，反復法を用いて $\nabla L(\mathbf{w}) = 0$ を満たす $\mathbf{w}$ を近似することを考える。勾配ベクトル ∇L の情報をもとに (3.5) の解を近似する手法を勾配降下法という。 一般に実数値関数 $f(\mathbf{w})$ について，ベクトル $\mathbf{d}$ に対して， ある実数 $\eta > 0$ が存在して $f(\mathbf{w} + \eta\mathbf{d}) < f(\mathbf{w})$ となるとき，ベクトル $\mathbf{d}$ を $\mathbf{w}$ における降下方向とよぶ。 $\nabla f(\mathbf{w})^T\mathbf{d} < 0$ を満たす $\mathbf{d}$ は降下方向である。 なぜなら，十分小さく $\eta > 0$

を取れば，テイラーの定理より

$$f(\mathbf{w}+\eta\mathbf{d}) \approx f(\mathbf{w}) + \eta\nabla f(\mathbf{w})^T d \tag{3.6}$$

であるから，$\nabla f(\mathbf{w})^T\mathbf{d} < 0$ より $f(\mathbf{w}+\eta\mathbf{d}) < f(\mathbf{w})$ とすることができる。 $\mathbf{d} = -\nabla f(\mathbf{w})$ が $\mathbf{w}$ における降下方向になることはすぐに分かる。 $\mathbf{w}$ が 1 次元の場合，$-f'(\mathbf{w})$ が降下方向であることは明らかであろう（図 3.2(b)）。



(a) 勾配ベクトルと降下方向．楕円状の線は凸関数の等高線を表す．

(b) 降下方向

図 3.2　勾配降下法

勾配降下法[*4]はつぎのような反復法である。

勾配降下法

ステップ 1.　適当な初期点 $\mathbf{w}(0)$ を選び，$k = 0$ とする。
ステップ 2.　$L(\mathbf{w}(k)) = 0$ ならば解を $\mathbf{w}^* = \mathbf{w}(k)$ として，反復法を終了する[*5]
ステップ 3.　適当なステップ幅 $\eta(k) > 0$ に対して

$$w(k+1) = w(k) - \eta(k)\nabla L(\mathbf{w}(k))$$

とする。
ステップ 4.　$k := k+1$ （k の値を 1 増やす）としてステップ 2 に戻る。

すなわち，勾配降下法とは点 $\mathbf{w}(0)$ から関数 L が減少する方向 $-\nabla L(\mathbf{w}(0))$ に沿って少し進み点 $\mathbf{w}(1)$ に到達，そこで $-\nabla L(\mathbf{w}(1))$ に方向を変えてさらに少し進んで点 $\mathbf{w}(2)$ に到達，そこで $-\nabla L(\mathbf{w}(2))$ に方向を変えて，という手続きを $\nabla L(\mathbf{w}(k)) = 0$ となるまで続ける，というアルゴリズムである。

[*4] 勾配降下法 はつぎのような反復法である．

[*5] 実際にはあらかじめ決められた十分小さな ϵ に対して，$||\nabla L(w(k))|| < \epsilon$ となったところで終了する．

各ステップで設定するステップ幅 $\eta(k)$ を機械学習では学習率と呼ぶ。 機械学習において分析者があらかじめ設定すべきパラメータをハイパーパラメータと呼び，学習率はハイパーパラメータの一つである。学習率をステップ k によらず一定値 η とすることもある。 学習率の設定法については問題に応じていろいろな手法が開発されているが，決定的な方法は存在せず，試行錯誤により設定せざるを得ない。 η を小さく取りすぎると解への収束が遅くなる一方，η を大きく取りすぎると (3.6) の議論から分かるように，関数 L の値が減少しない点に $\mathbf{w}(k)$ が到達する可能性がある。 この学習率の設定が解を求めるための鍵となる。

前節で述べたように機械学習では最小化する関数が

$$L(\mathbf{w}) = \frac{1}{N}\sum_{i=1}^{N} l(\mathbf{x}_i, y_i; \mathbf{w})$$

という形をしている。 機械学習においては訓練データ数 N は非常に大きい。したがって，損失関数の勾配の式

$$\nabla L(\mathbf{w}) = \frac{1}{N}\sum_{i=1}^{N} \nabla l(\mathbf{x}_i, y_i; \mathbf{w})$$

から分かるように, 勾配降下法において損失関数の勾配を求めるたびに N 個の勾配 $\{\nabla l(\mathbf{x}_i, y_i; \mathbf{w})\}_{i=1,\ldots,N}$ の計算が必要になり計算負荷が非常に大きくなってしまう。そこで N 個のデータをすべて使わずに，1 つまたは少数のデータのみを使って勾配ベクトルを計算する，確率的勾配降下法という反復法を用いる。確率的勾配降下法では勾配降下法のステップ 3 がつぎのようになる:

確率的勾配降下法

ステップ 3. （ランダムに）選んだ訓練データ $(\mathbf{x}_i, y_i)$ を用いて $\nabla l(\mathbf{x}_i, y_i; \mathbf{w}(k))$ を計算し，適当なステップ幅 $\eta(k) > 0$ に対して

$$w(k+1) = w(k) - \eta(k)\nabla l(\mathbf{x}_i, y_i; \mathbf{w}(k))$$

とする。（他のステップは勾配降下法と同じ）

勾配の計算に用いる訓練データ $(\mathbf{x}_i, y_i)$ はランダムに選んでもよいし，何らかの順序に従って選んでもよい。 また，訓練データがつぎつぎと到着する状況において，訓練データを 1 つ観測する度にパラメータ $\mathbf{w}$ を更新する場合にも確率的勾配降下法が利用できる。

3.3 線形回帰

ここからは機械学習のアルゴリズムを具体的に見ていくことにする。 まずは線形回帰を扱う。線形回帰では特徴量 $\mathbf{x}$ と正解 y の間の関係をパラメータ $\mathbf{w} = (w_0, w_1, \ldots, w_D)^T$ を用いて

$$y = w_0 + \sum_{i=1}^{D} w_i x_i$$

とモデル化する。機械学習では w_0 をバイアスと呼ぶ。線形回帰モデルは統計学や計量経済学においてもよく用いられるモデルである。 しかし，統計学や計量経済学においては回帰係数 w_i の統計的有意性や特

徴量（説明変数）と正解（独立変数）の間の相関関係または因果関係に主な関心がある一方，機械学習においては汎化誤差の最小化，すなわち精度のよい予測をすることに主な関心がある。また，線形回帰モデルを用いると機械学習のさまざまな概念を比較的分かりやすく説明できるため，線形回帰モデルは機械学習の教科書で取り上げられることが多い。

より一般的に

$$y = w_0 + \sum_{j=1}^{M} w_j \phi_j(\mathbf{x}) \tag{3.7}$$

というモデルを考えることもできる。ここで $\{\phi_j\}_{j=1,2,\ldots,M}$ は基底関数と呼ばれる非線形関数である。非線形関数を用いることでモデルの表現能力が上昇する，すなわち訓練誤差の小さいパラメータを求めやすくなる。 特徴量 $\mathbf{x}$ が 2 次元 $\left(\mathbf{x} = (x_1, x_2)^T\right)$ のときは

$$\phi_j(\mathbf{x}) = x_1^{3-j} x_2^{j-1},\ j = 1, 2, 3$$

が基底関数の一例となる。 非線形関数を使うものの，モデル (3.7) はパラメータ $\mathbf{w}$ に関して線形であるため線形回帰モデルと呼ばれる。 また，$\phi_0(\mathbf{x}) = 1$ と定義することで，(3.7) を $y = \mathbf{w}^T \phi(\mathbf{x})$ と簡潔に表現することができる。ここで，$\mathbf{w} = (w_0, w_1, \ldots, w_M)^T$，$\phi(\mathbf{x}) = (\phi_0(\mathbf{x}), \phi_1(\mathbf{x}), \ldots, \phi_M(\mathbf{x}))^T$ である。

線形回帰では平均二乗誤差をパフォーマンス尺度とし，損失関数は

$$L(\mathbf{w}) = \frac{1}{N} \sum_{i=1}^{N} \left\{ y_i - \mathbf{w}^T \phi(\mathbf{x}_i) \right\}^2 \tag{3.8}$$

となる。損失関数 $L(\mathbf{w})$ は凸関数であるため， 最小化問題 $\min_{\mathbf{w}} L(\mathbf{w})$ の解は $\nabla L(\mathbf{w}) = 0$ を解くことによって求めることができる。 そこで

$$\nabla L(\mathbf{w}) = -\frac{1}{2N} \sum_{i=1}^{N} (y_i - \mathbf{w}^T \phi(\mathbf{x}_i)) \phi(\mathbf{x}_i)$$

であることを用いると，最適解は

$$\mathbf{w}^* = \left(\mathbf{\Phi}^T \mathbf{\Phi} \right)^{-1} \mathbf{\Phi}^T \mathbf{y} \tag{3.9}$$

と求まる。ここで $\mathbf{y} = (y_1, y_2, \ldots, y_N)^T$,

$$\mathbf{\Phi} = \begin{pmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \ldots & \phi_M(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \phi_1(\mathbf{x}_2) & \ldots & \phi_M(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(\mathbf{x}_N) & \phi_1(\mathbf{x}_N) & \ldots & \phi_M(\mathbf{x}_N) \end{pmatrix}$$

である。数学的には簡潔な解を求めることができるが, 機械学習ではデータ数 N が非常に大きいことが通常であり， その場合 (3.9) の逆行列 $(\mathbf{\Phi}^T \mathbf{\Phi})^{-1}$ を計算することは現実的でなく[*6]， 実際には $\nabla L(\mathbf{w}) = 0$ の解を（確率的）勾配降下法を使って近似する。

[*6] 逆行列が存在しないこともある.

さて，ここで単純な線形回帰モデルを用いて過学習について見ておく。（機械学習では実際にはありえないが）特徴量が 1 つ（$D = 1$）の場合で， 基底関数を $\phi_j(x) = x^j$ とした線形回帰モデル

$$y = \sum_{j=0}^{M} w_j x^j \tag{3.10}$$

を考える。 ここで M はハイパーパラメータとなる。M を大きく取ると訓練誤差を小さくできるが，モデルが訓練データにフィットしすぎてしまい，汎化誤差が大きくなる，すなわち未知のデータに対する予測能力が落ちてしまう現象が起きる。この現象を過学習と呼ぶ。訓練データの数が少ないときにパラメータが多い複雑なモデルを用いると過学習が起きやすくなる。極端な例ではあるが，異なる訓練データの数が $(N + 1)$ 個のときに 最大次数 $M = N$ の多項式でモデル化をすると，訓練誤差をゼロにするパラメータ $\{w_j\}_{j=0,1,\ldots,M}$ を求めることができる（図 3.3)。 しかし，そういったモデルの未知のデータに対する予測力は怪しい。一方で M を小さく取りすぎると訓練誤差と汎化誤差が大きくなる過小学習という現象が起きる可能性がある。

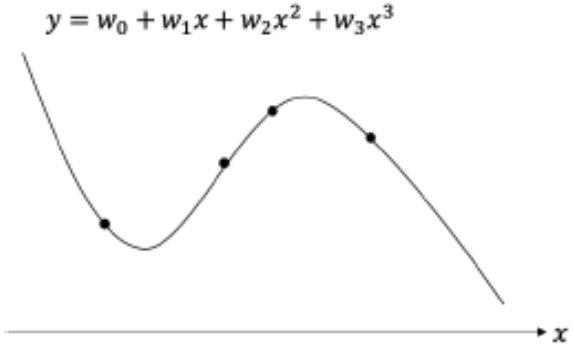


図 3.3　過学習．データ数が 4 のときに 3 次関数でモデル化.

過学習を防ぐ方法の一つとして正則化がある。正則化とは損失関数に正則化項と呼ばれるパラメータ $\mathbf{w}$ が大きくなることに対するペナルティ項を加えて，最小化問題を解く手法である。正則化学習では例えば最適化問題

$$\min_{\mathbf{w}} \frac{1}{N} \sum_{i=1}^{N} l(\mathbf{x}_i, y_i; \mathbf{w}) + \lambda \sum_{j=1}^{M} w_j^2$$

を解く。ここで $\sum_{j=1}^{M} w_j^2$ が L^2 正則化項と呼ばれる正則化項，$\lambda > 0$ は正則化の強さを表すハイパーパラメータである。正則化項 $\sum_{j=1}^{M} w_j^2$ を最小化問題に組み込むことで，なるべくゼロに近いパラメータ $\{w_j\}_{j=1,2,\ldots,N}$ が求められるようになり， 過学習を防ぐことができる。ほかにもさまざまな正則化項が考えられており，L^1 正則化項 $\sum_{j=1}^{M} |w_j|$ もよく用いられる。L^1 正則化項を用いると，ゼロとなるパラメータ $\{w_j\}_{j=1,2,\ldots,N}$ の数が多くなるように最小化問題が解かれる。

ここまでいくつかのハイパーパラメータが出てきた。ハイパーパラメータはどのように選択すればよいのだろうか。当然汎化誤差を小さくするように選ぶべきであるが，実際に汎化誤差を計測することはできない。そこで，交差検証と呼ばれる技術を使う。交差検証とは，はじめに与えられたデータを訓練データ

と検証データに分割し，訓練データを用いてパラメータを推定し，検証データによって誤差を計測する手法である。訓練データと検証データへの分割の仕方を何回か変えて，それぞれの回での誤差を求め，それらを平均することでモデルのよさを検証する。ハイパーパラメータをいろいろに変化させ交差検証を行うことで，最も適当なハイパーパラメータを選ぶことができるようになる。

3.4 分類（ロジスティック回帰）

正解 y が離散値を取る分類問題をロジスティック回帰と呼ばれるアルゴリズムを使って解くことを考える。 2 クラスに分類する問題では，正解 y は 0 または 1 の値をとり，例えば $y = 1$ のときクラス 1，$y = 0$ のときクラス 2 と定義する。 $K(> 2)$ クラスに分類する多クラス分類問題では，クラス k に属することを，k 番目の要素が 1，それ以外の要素はすべて 0 である K 次元ベクトルで表現する。 例えば，クラス 3 に属することを $y = (0, 0, 1, 0, \ldots, 0)^T$ という K 次元ベクトルで表現する。

まずは 2 クラス分類問題，すなわち正解 y が $y \in \{0, 1\}$ となる場合を扱う。 2 クラス分類問題では識別関数 $c(\mathbf{x}; \mathbf{w})$ を用意し $c(\mathbf{x}; \mathbf{w}) > 0$ ならばクラス 1（$y = 1$），$c(\mathbf{x}; \mathbf{w}) < 0$ ならばクラス 2（$y = 0$）となるように，訓練データからパラメータ $\mathbf{w}$ を推定することが目標となる。 ここでは線形識別関数

$$c(\mathbf{x}; \mathbf{w}) = w_0 + w_1 x_1 + \cdots + w_D x_D = \mathbf{w}^T \mathbf{x} \tag{3.11}$$

を考える。 ここで $\mathbf{w} = (w_0, w_1, \ldots, w_D)^T$，$\mathbf{x} = (1, x_1, \ldots, x_D)^T$ とした。 線形回帰の場合と同様に，基底関数を使って識別関数を $c(\mathbf{x}; \mathbf{w}) = \mathbf{w}^T \phi(\mathbf{x})$ としてもよい。 線形識別関数によって 2 クラスを分類するということは，2 つのクラスを超平面（直線）で区分することになる（図 3.4(a)）。 ロジスティック回帰では特徴量 $\mathbf{x}$ と正解 y の関係を，シグモイド関数 σ を作用させた $\sigma(\mathbf{w}^T \mathbf{x})$ を用いて

$$y = \begin{cases} 1, & \sigma(\mathbf{w}^T \mathbf{x}) \geq 0.5 \\ 0, & \sigma(\mathbf{w}^T \mathbf{x}) < 0.5 \end{cases} \tag{3.12}$$

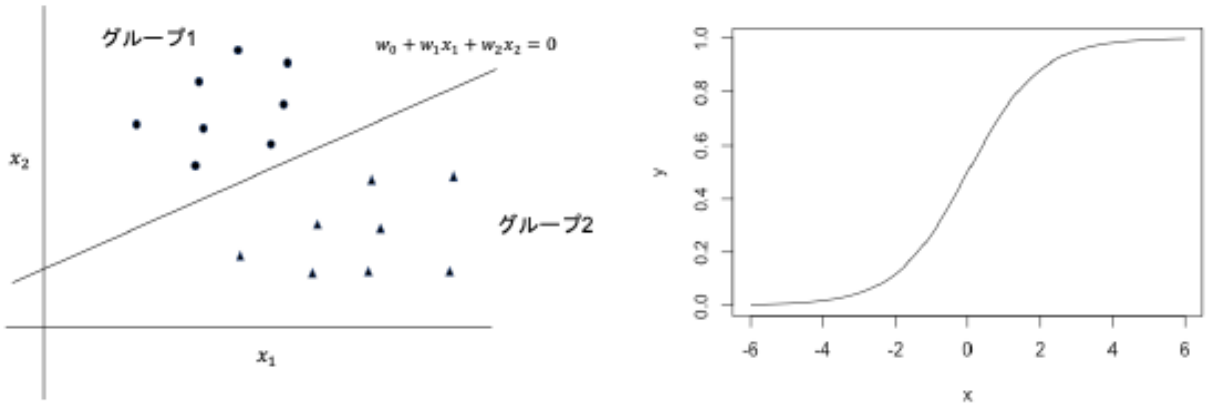
とモデル化する。シグモイド関数 σ （ロジスティック関数とも言う）とは

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

によって定義される単調増加関数である（図 3.4(b)）。シグモイド関数は $[0, 1]$ に値を取る関数であり，$\mathbf{w}^T \mathbf{x}$ にシグモイド関数を作用させることで $\sigma(\mathbf{w}^T \mathbf{x})$ を確率とみなすことができるようになる。 そこで，特徴量 $\mathbf{x}$ を観測したときにそれがクラス C_k（$k = 1, 2$） に含まれる確率 $P(C_k|\mathbf{x})$ を

$$P(C_1|\mathbf{x}) = \sigma(\mathbf{w}^T \mathbf{x})$$

と定義する。 すると，(3.12) は 2 つの確率 $P(C_1|\mathbf{x})$ と $P(C_2|\mathbf{x}) = 1 - P(C_1|\mathbf{x})$ を比較して値の大きい方のクラスに特徴量 $\mathbf{x}$ を分類するというルールになっている。なお，(3.12) は非線形関数によって識別をしているように見えるが， $\sigma(\mathbf{w}^T \mathbf{x}) = 0.5$ を解くと $\mathbf{w}^T \mathbf{x} = 0$ であり， 線形識別関数による識別であることが分かる。



(a) 線形識別関数による分類　　(b) シグモイド関数 $y = 1/(1 + e^{-x})$

図 3.4　ロジスティック回帰

パラメータ $\mathbf{w}$ は最尤法を用いて推定する。 $\pi(\mathbf{x}_i; \mathbf{w}) = P(C_1|\mathbf{x}_i)$ とおく。 訓練データ $\{(\mathbf{x}_i, y_i)\}_{i=1,2,\ldots,N}$ が観測される確率は

$$P(\mathbf{w}) = \prod_{i=1}^{N} \pi(\mathbf{x}_i; \mathbf{w})^{y_i} (1 - \pi(\mathbf{x}_i; \mathbf{w}))^{1-y_i}$$

となる。この $P(\mathbf{w})$ は尤度関数と呼ばれる。 最尤法とは尤度関数 $P(\mathbf{w})$ を最大にするパラメータ $\mathbf{w}^*$ を推定値とする手法である。 簡単のため $P(\mathbf{w})$ を最大化するのではなく，$-\log P(\mathbf{w})$ すなわち

$$L(\mathbf{w}) = -\sum_{i=1}^{N} \{y_i \log(\pi(\mathbf{x}_i; \mathbf{w})) + (1 - y_i) \log(1 - \pi(\mathbf{x}_i; \mathbf{w}))\}$$

の最小化問題を解く。 この $L(\mathbf{w})$ がロジスティック回帰における損失関数となる。 あとは

$$\nabla L(\mathbf{w}) = \sum_{i=1}^{N} (\pi(\mathbf{x}_i; \mathbf{w}) - y_i)\mathbf{x}_i$$

であることを使って，確率的勾配降下法などを用いて最適解 $\mathbf{w}^*$ を求めればよい。

K クラス分類の場合には K 個の線形識別関数 $c(\mathbf{x}; \mathbf{w}_k) = \mathbf{w}_k^T \mathbf{x}$ を用いて識別をする。 ここで，$\mathbf{w}_k = (w_{k0}, w_{k1}, \ldots, w_{kD})^T$ である。この場合には，シグモイド関数の代わりにソフトマックス関数と呼ばれる関数を使って

$$P(C_k|\mathbf{x}) = \frac{\exp(\mathbf{w}_k^T \mathbf{x})}{\sum_{j=1}^{K} \exp(\mathbf{w}_j^T \mathbf{x})}$$

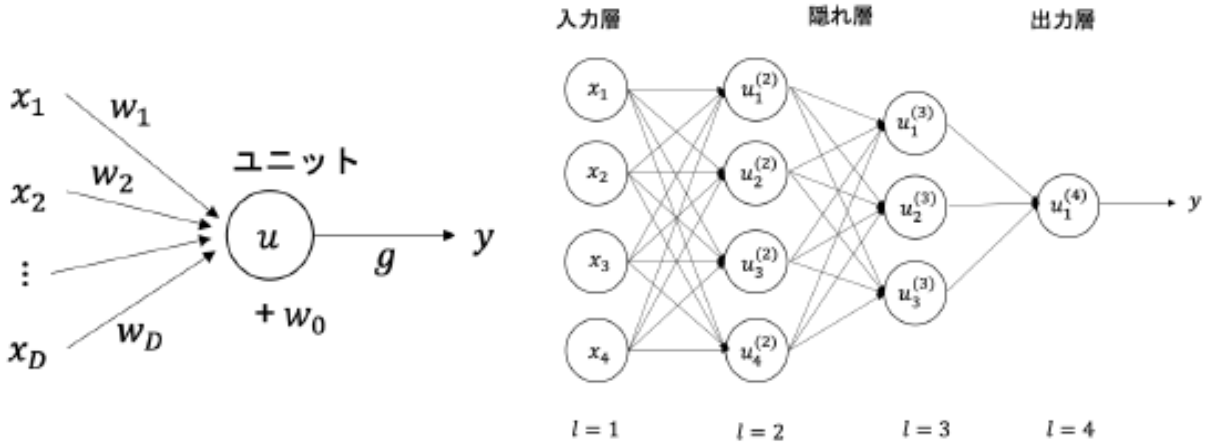
として，最尤法を用いてパラメータ $\{\mathbf{w}_k^*\}_{k=1,\ldots,K}$ を推定すればよい。 そして，特徴量 $\mathbf{x}$ を $\mathbf{w}_k^*$ を用いて計算した $P(C_k|\mathbf{x})$ が最も大きくなるクラスに分類すればよい。

3.5　ニューラルネットワーク

機械学習（教師あり学習）とは訓練データ $\{(\mathbf{x}_i, y_i)\}_{i=1,2,\ldots,N}$ から特徴量 $\mathbf{x}$ と正解 y の関係 $y = f(\mathbf{x}; \mathbf{w})$ を推定することであった。 前節までは

$$y = g\left(w_0 + \sum_{i=1}^{D} w_i x_i\right) \tag{3.13}$$

とモデル化し，パラメータ $\mathbf{w}$ を求める問題を主に扱った。 関数 g は線形回帰では恒等関数 $g(u) = u$ であり， ロジスティック回帰では (3.12) によって定義した。 この節ではより柔軟で表現力の高いモデルを構築できるニューラルネットワークを紹介する。



(a) ユニット　　(b) ニューラルネットワーク

図 3.5　ニューラルネットワーク

(3.13) を図 3.5(a) のように表現してみる。この図は，入力 $x_1, x_2, \ldots, x_D$ に重み $w_1, w_2, \ldots, w_D$ を掛けて足し合わせた $u = \sum_{i=1}^{D} w_i x_i$ をユニットが受け取り，バイアス w_0 を加えた上で関数 g による変換を施して $y = g(w_0 + u)$ を出力する，と読む[*7]。 このユニットを図 3.5(b) のように層状に並べたものをニューラルネットワークとよぶ。とくに，入力から出力に向けて一方向に情報が流れるニューラルネットワークを順伝播型ニューラルネットワークと呼ぶ。また，ニューラルネットワークやそのバリエーション，それらを解くための方法などを総称して深層学習（ディープラーニング）と呼ぶ。本節では順伝播型ニューラルネットワークの基本を概観する。

l を層のインデックス，L を層数として，最初の層（$l = 1$）を入力層，最後の層（$l = L$）を出力層，それ以外を隠れ層と呼ぶ。 層数と各層のユニット数は任意に決めることができる。第 l 層 i 番目のユニットへの入力を $u_i^{(l)}$，第 l 層 i 番目のユニットからの出力を $z_i^{(l)}$ と表す。 入力層 $l = 1$ においては $u_i^{(1)} = x_i$ であり， 出力層 $l = L$ においては $z_i^{(L)} = y$ である。 さらに，第 l 層 i 番目のユニットから第 $(l+1)$ 層

[*7]　g がヘビサイドの階段関数のとき，このモデルを（単純）パーセプトロンと呼ぶ.

j 番目のユニットへの出力の重みを $w_{ji}^{(l+1)}$，バイアスを $w_{j0}^{(l+1)}$ と書くことにすれば，

$$u_j^{(l+1)} = w_{j0}^{(l+1)} + \sum_i w_{ji}^{(l+1)} z_i^{(l)} \tag{3.14}$$

となる。ここで，$z_j^{(l+1)} = u_j^{(l+1)}$ としてしまっては，入力 $\mathbf{x}$ と出力 y の関係は線形となってしまい，表現力がまったく改善しない。ニューラルネットワークでは活性化関数と呼ばれる関数 $h^{(l)}$ を用いて

$$z_j^{(l+1)} = h^{(l+1)}\left(u_j^{(l+1)}\right) \tag{3.15}$$

とする。活性化関数としてはシグモイド関数や正規化線形関数 $h(u) = \max\{0, u\}$ などが用いられる。また，出力層における活性化関数として，回帰では恒等関数 $h(u) = u$ を使えばよく，2 クラス分類ではシグモイド関数を用いた (3.12) を使えばよい。

(3.14)(3.15) はまとめて

$$\mathbf{z}^{(l+1)} = h^{(l+1)}(\mathbf{u}^{(l+1)}),\ \mathbf{u}^{(l+1)} = \mathbf{W}^{(l+1)}\mathbf{z}^{(l)},\quad l = 1, 2, \ldots, L-1$$

と書くことができる。ここで

$$\mathbf{W}^{(l+1)} = \begin{pmatrix} w_{10}^{(l+1)} & w_{11}^{(l+1)} & w_{12}^{(l+1)} & \cdots \\ w_{20}^{(l+1)} & w_{21}^{(l+1)} & w_{22}^{(l+1)} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix},\ \mathbf{z}^{(l)} = \begin{pmatrix} 1 \\ z_1^{(l)} \\ z_2^{(l)} \\ \vdots \end{pmatrix}$$

である。$\mathbf{W}^{(l+1)}$ は (第 $(l+1)$ 層のユニット数) × (第 l 層のユニット数 $+1$) 行列，$\mathbf{z}^{(l)}$ は (第 l 層のユニット数 $+1$) 次元ベクトルである。また，活性化関数 $h^{(l)}$ の引数がベクトル $\mathbf{u}$ のときは $h^{(l)}(\mathbf{u}) = (h^{(l)}(u_1), h^{(l)}(u_2), \ldots)^T$ と定義する。この表記を使うと，入力 $\mathbf{x}$ と出力 y の関係は

$$y = h^{(L)}\left(\mathbf{W}^{(L)} h^{(L-1)}\left(\cdots h^{(3)}\left(\mathbf{W}^{(3)} h^{(2)}\left(\mathbf{W}^{(2)}\mathbf{x}\right)\right)\right)\right) \tag{3.16}$$

と書ける。すなわち，入力 $\mathbf{x}$ からスタートして線形変換 $\mathbf{W}^{(l)}$ と活性化関数 $h^{(l)}$ による変換をつぎつぎと作用させて，出力 y に至る。よって，かなり複雑ではあるが，入力 $\mathbf{x}$ と出力 y の間の関係をパラメータ $\mathbf{w}$ をもつ関数 f を用いて $y = f(\mathbf{x}; \mathbf{w})$ と表現するという機械学習の基本的な考え方は保たれている。表現力の高い非線形関数 (3.16) よってモデル化をすることにより，例えば線形識別関数では分類できない問題も解ける可能性が出てくる。あとは，解きたい問題に応じて損失関数

$$L(\mathbf{w}) = \frac{1}{N}\sum_{i=1}^{N} l(\mathbf{x}_i, y_i; \mathbf{w})$$

を定義して，最小化問題 $\min_{\mathbf{w}} L(\mathbf{w})$ を確率的勾配降下法などによって解けばよい。しかし，ニューラルネットワークにはいくつかの困難がある。

まず，(確率的) 勾配降下法には勾配 $\nabla L(\mathbf{w})$ の計算が必要になるが，$\nabla L(\mathbf{w})$ の計算には (3.16) によって定義される y のパラメータ w_{ji} に関する偏微分 $\partial y/\partial w_{ji}$ の計算が必要になる。単純に数値微分を適用すると，合成関数の微分を何度も繰り返すことになるため，計算量が莫大になり実用的ではない。し

かし，誤差逆伝播法と呼ばれる微分計算がこの問題を解決する。誤差逆伝播法については参考文献を参照してほしいが，ニューラルネットワークを含む深層学習が実用化されるようになった理論的背景の一つは誤差逆伝播法の発明とその改良にある。 つぎに，ニューラルネットワークにおける損失関数 $L(\mathbf{w})$ は一般には凸関数にはならないという問題がある。 したがって，確率的勾配降下法を使って $\nabla L(\mathbf{w}) = 0$ の解を求めたとしても，それが大域的最適解になっている保障はない。 また，ニューラルネットワークでは多くのパラメータを持つ非線形関数により入力 $\mathbf{x}$ と出力 y の関係をモデル化することから，過学習が起こりやすことが容易に想像できる。したがって，深層学習においては正則化の技術が一層重要となる。また，層数や各層のユニット数をいくつにすればよいのかについても一般論は知られていない。

このようにさまざまな問題があるにも関わらず，ニューラルネットワークを含め深層学習が画像認識をはじめさまざまな問題に対して高いパフォーマンスを発揮している。しかし，なぜ深層学習がそこまでうまく機能するのかについては，理論的にはまだ分かっていないことが多い。

3.6 ノーフリーランチ定理

ここまでいくつかの機械学習のアルゴリズムを紹介してきた。ここで紹介したアルゴリズム以外にもさまざまなものが知られており，同じ問題に対して複数のアルゴリズムが適用できる場合はたくさんある。では，結局どのアルゴリズムを用いるのが一番よいのだろうか。最後に Wolpert (1996)[5] や Wolpert and Macready (1997)[6] によるノーフリーランチ定理を紹介する。これらの論文の記述は数学的であるが，簡単に述べれば「データに関して何の前提条件も設けなければ，あらゆる問題について最高のパフォーマンスをもつアルゴリズムは存在しない」もしくは「アルゴリズム A がある問題について最もよいパフォーマンスをもっていたとしても，それとは別の問題が存在して，そこでは別のアルゴリズム B がアルゴリズム A よりよいパフォーマンスをもつ」という定理である。すなわち，あらゆる問題を効率的に解く万能なアルゴリズムは存在しないということである。したがって，機械学習においてはデータや問題の特性にあわせたアルゴリズムを選択することが重要ということになる。

数学的表記の注意

本章ではイタリック体の小文字 x はスカラー（実数）を表し， 太字のローマン体小文字 $\mathbf{x}$ はベクトルを表す。すべてのベクトルは列ベクトルとする。T は転置を表す記号であり， 文中で列ベクトルを表記する際には，$\mathbf{x}^T$ という表記を使う。例えば，

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_D \end{pmatrix}$$

のとき，$\mathbf{x} = (x_1, x_2, \ldots, x_D)^T$ と表記する。 $\mathbf{x}^T\mathbf{y}$ はベクトルの内積を表す。 すなわち，$\mathbf{y} = (y_1, y_2, \ldots, y_D)^T$ として，

$$\mathbf{x}^T\mathbf{y} = \sum_{i=1}^{D} x_i y_i$$

である。行列は太字のローマン体大文字 $\mathbf{A}$ で記す。 また添字の解釈に注意する。本章では N で訓練データの数，D で特徴量の数を表している。$\mathbf{x}$ が特徴量であるとき $\mathbf{x} = (x_1, x_2, \ldots, x_D)^T$ を意味する。また，$\mathbf{x}_i$ は i 番目の訓練データの特徴量を表し，成分表示すれば $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iD})^T$ となる。また，ベクトル $\mathbf{x}$ のノルム $\|\mathbf{x}\|$ を $\|\mathbf{x}\| = \sqrt{\sum_{i=1}^{D} x_i^2}$ と定義する。

参考文献

[1] Bishop, C. M. (2011). *Pattern Recognition and Machine Learning*. Springer. （元田浩ら訳,『パターン認識と機械学習（上）（下)』，丸善出版，2007 年）

[2] Goodfellow, I., Y. Bengio, and A. Courville (2016). *Deep Learning*. MIT Press.（岩澤有祐ら監訳,『深層学習』，アスキードワンゴ，2018 年）

[3] Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill.

[4] Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. MIT Press.

[5] Wolpert, D. H. (1996). "The lack of a priori distinctions between learning algorithms," Neural Computation, 8(7), 1341–1390.

[6] Wolpert, D. H. and W. G. Macready (1997) "No free lunch theorems for optimization," IEEE Transactions on Evolutionary Computation, 1(1), 67-82

第 4 章

不動産分野における統計・機械学習の利用

4.1 統計・機械学習の手法

本章では，不動産分野で利用される統計・機械学習の手法とその特性について概説する。不動産分野における統計・機械学習の利用可能性は多岐にわたるが， 本章では特に，不動産価格関数の推定と予測に用いられる回帰モデルに絞って説明を行う。4.1 節では，回帰モデルの仕組みとその推定法に関して説明する。 基本的な線形回帰モデルから解説を始め，発展的な手法にまで言及するが，特に線形回帰モデルに関しては基底関数を導入することで， 幅広い関数の近似に応用できるように解説した。4.8 節では，4.1 節で述べた手法を 実際の分析に適用する際の注意点・問題点について概説をする。理論的な背景はより専門的な統計の書籍に譲り，本章ではコンピュータによる 数値シミュレーションを用いて問題点を具体的に示すことにする。

統計的な検定や推定値の理論的な性質，空間モデルについては取り扱わないので， これらに関する詳細は清水千弘& 唐渡広志 (2007)[5] の 3・4 章を参照されたい。 また各モデルとその学習方法に関しては，Bishop (2012)[1] でより詳細に検討・解説されている。

本章で使用したプログラムのソースコードは，`https://github.com/hayato-n` で公開予定である。

4.2 線形回帰モデル

4.2.1 線形回帰

さて，不動産物件のデータが N 件集まっていたとする。また，各物件 $n = 1, \ldots, N$ に対して，その価格 y_n と 不動産の特性 $\boldsymbol{x}_n$ が分かっているとしよう。 このとき，新しい物件 $*$ の不動産の特性 $\boldsymbol{x}_*$ から，未知の価格 y_* を予測するモデルを構成しよう。 このようなモデルを回帰モデルという。

線形回帰モデルの枠組みでは，不動産価格は不動産の特性とその効果量の線形結合で与えられる。 すなわち，効果量を $\boldsymbol{\beta}$ とすると，

$$y_n \approx \boldsymbol{\beta}^T \boldsymbol{\phi}(\boldsymbol{x}_n) \tag{4.1}$$

によって，不動産価格を近似する。ここで $\boldsymbol{\phi}$ は不動産特性ベクトルを入力としてそれに何らかの変換を施したベクトルを返す関数であり，基底関数と呼ばれる。最も単純な基底関数の一例として，定数項を加

えるものが考えられる。また，線形回帰モデルにおいて $\boldsymbol{\beta}$ は回帰係数とよばれる。

基底関数の導入は，線形回帰モデルの枠組みの中で非線形関数の近似を実現する。 例えば，基底関数として多項式変換を考えてみよう。 入力としては，1 次元の x を仮定する。 このとき D 次の多項式基底関数を以下のように定義する。

$$\boldsymbol{\phi}(x) = (x^0, x^1, \ldots, x^{D-1})^T \tag{4.2}$$

ここで $x^0 = 1$ であるから， 2 次元の多項式変換は単純に定数項を加える処理に相当する。

図 4.1 は多項式基底関数を用いた非線形関数の近似例である。 線形回帰モデルの枠組みの中で，確かに非線形関数の関数を学習できていることがわかる。 ただしモデルの推定法によって，学習した関数の形状に多少の差異が見られる。

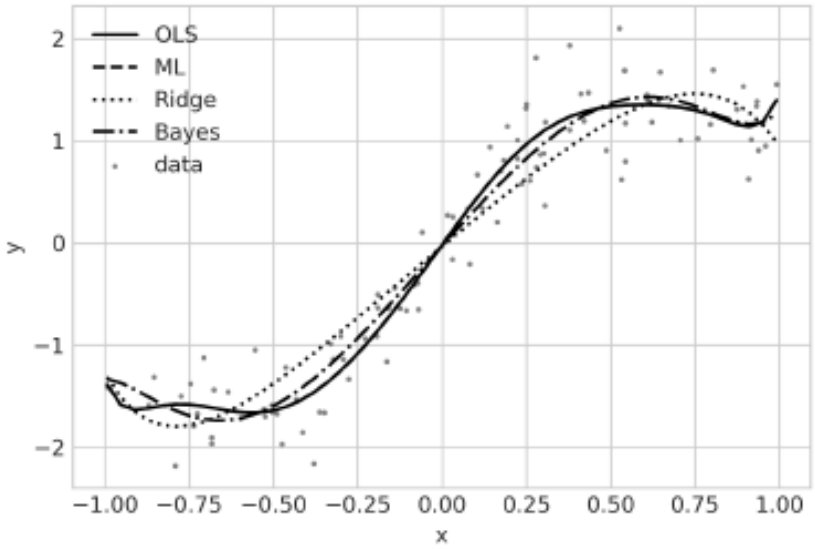


図 4.1 線形回帰モデルによる非線形関数の近似

以降では，線形回帰モデルの推定法について述べる。

4.2.2 最小二乗法

式 (4.1) が精度の良い近似になるように回帰係数 $\boldsymbol{\beta}$ を 学習することを考える。 このような，学習される値のことをパラメータとよぶ。パラメータを学習するための 1 つの方法は，「精度の悪さ」を何らかの方法で数値化した損失関数を定義し，それを最小化するようにパラメータを最適化する，というものである。このとき，損失関数はパラメータを入力とする関数である。 最もよく使われる損失関数として，実測値と予測値の差の二乗を採用するものがある。 具体的には，損失関数として以下のものを採用する。

$$E_{OLS} = \sum_n \varepsilon_n^2 = \sum_n (y_n - \boldsymbol{\beta}^T \boldsymbol{\phi}(\boldsymbol{x}_n))^2 \tag{4.3}$$

ここで ε_n は物件 n の予測誤差である。二乗損失を最小化するようなパラメータの推定法を，最小二乗法 (OLS: Ordinary Least Squares method) と呼ぶ。

幸運にも，二乗損失 E_{OLS} を最小化するパラメータ $\boldsymbol{\beta}$ は解析的に求めることができる。E_{OLS} を行列を用いて書き直すと，以下のようになる。

$$E_{OLS} = (\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{\beta})^T(\boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{\beta}) \tag{4.4}$$

ただし，$\boldsymbol{y} = (y_1, \ldots, y_N)^T, \boldsymbol{\Phi} = (\boldsymbol{\phi}(\boldsymbol{x}_1), \ldots, \boldsymbol{\phi}(\boldsymbol{x}_N))^T$ とおいた。$\boldsymbol{\Phi}$ は計画行列とよばれることがある。

このような定式化の下で，E_{OLS} はベクトル $\boldsymbol{\beta}$ で解析的に微分可能である。ここから最適なパラメータ $\boldsymbol{\beta}$ が満たすべき条件は，

$$\frac{dE_{OLS}}{d\boldsymbol{\beta}} = -2\boldsymbol{\Phi}^T\boldsymbol{y} + 2\boldsymbol{\Phi}^T\boldsymbol{\Phi}\boldsymbol{\beta} = 0 \tag{4.5}$$

となる。ここで，$\boldsymbol{\Phi}^T\boldsymbol{\Phi}$ が 正則行列（逆行列を持つ行列）であると仮定すれば，

$$\boldsymbol{\beta} = (\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T\boldsymbol{y} \tag{4.6}$$

が得られる。

4.2.3 最尤推定法

最小二乗法は，二乗損失が最小化されるようにパラメータを調整する方法であった。別のアプローチとして，線形回帰モデルを確率的なモデルと捉える方法がある。 具体的には，誤差 ε_n が独立同分布な平均ゼロのガウス分布に従うと仮定する。このとき，$\boldsymbol{y}$ の $\mathbf{X} = (\boldsymbol{x}_1^T, \ldots, \boldsymbol{x}_N^T)^T$ による条件付き分布は

$$\begin{aligned} p(\boldsymbol{y} \mid \mathbf{X}) &= \prod_n p(y_n \mid \boldsymbol{x}_n) \\ &= \prod_n \mathcal{N}(y_n \mid \boldsymbol{\beta}^T\boldsymbol{\phi}(\boldsymbol{x}_n), \sigma^2) \end{aligned} \tag{4.7}$$

となる。ただし，$\mathcal{N}(\cdot \mid \mu, \sigma^2)$ は平均 μ・分散 σ^2 のガウス分布の確率密度関数を表す。

この条件付き分布をパラメータ $\{\boldsymbol{\beta}, \sigma^2\}$ の関数と見なしたものを， 尤度関数 $L(\boldsymbol{\beta}, \sigma^2)$ とよぶ。尤度関数が大きくなるようなパラメータを設定すると，データの条件付き確率が大きくなるので，これを最大化するようにパラメータを調整するという方法が考えられる。 これを最尤推定 (MLE: Maximum Likelihood Estimation) とよぶ。

最尤推定を行うとき，通常は尤度関数そのものではなく， その対数 $\ell(\boldsymbol{\beta}, \sigma^2) = \log L(\boldsymbol{\beta}, \sigma^2)$ を使うことが多い。 この理由としては，対数尤度の方が尤度そのものよりも数学的に取り扱いやすい場合があること， 尤度は個々の確率密度の積であることから，値として極めてゼロに近くなり，コンピュータ上での数値的な誤差が大きくなる場合があること，などが挙げられる。 なお，対数は単調増加関数なので，対数尤度の最大化は尤度の最大化と等価である。

さて，線形回帰モデルにおける対数尤度関数の具体的な値を見ていこう。

$$\begin{aligned} \ell(\boldsymbol{\beta}, \sigma^2) &= \sum_n \log p(y_n \mid \boldsymbol{x}_n) \\ &= \sum_n \left\{ -\frac{1}{2\sigma^2}(y_n - \boldsymbol{\beta}^T\boldsymbol{\phi}(\boldsymbol{x}_n))^2 - \frac{1}{2}\log(2\pi\sigma^2) \right\} \\ &= -\frac{\sigma^{-2}}{2}E_{OLS} - \frac{N}{2}\log 2\pi + \frac{N}{2}\log\sigma^{-2} \end{aligned} \tag{4.8}$$

このように，誤差の分布としてガウス分布を採用すれば， 対数尤度関数の内部に二乗損失が現れることがわかる。 二乗損失 E_{OLS} はパラメータ $\boldsymbol{\beta}$ のみに依存するので， 対数尤度関数が最大化となるときにパラメータが満たすべき条件は以下のようになる。

$$\begin{aligned}\frac{\partial \ell}{\partial \sigma^{-2}} &= -\frac{1}{2}E_{OLS} + \frac{N}{2}\sigma^2 = 0 \\ \frac{\partial \ell}{\partial \boldsymbol{\beta}} &= -\frac{\sigma^{-2}}{2}\frac{dE_{OLS}}{d\boldsymbol{\beta}} = 0\end{aligned} \tag{4.9}$$

よって，$\boldsymbol{\beta}$ の最尤推定値は最小二乗推定値と一致し，かつ分散の推定値は $\sigma^2 = \frac{1}{N}E_{OLS}$ となる。

4.3 正則化

ここまで最小二乗法と最尤推定法を紹介してきたが， これらはあくまで得られたデータ $\{y_n, \boldsymbol{x}_n\}_{n=1,\ldots,N}$ に対する当てはまりを良くするものであり， 新しく得られたデータ $\{y_*, \boldsymbol{x}_*\}$ に対しても良く当てはまるとは限らない。 得られたデータに対してモデルが過剰に適合してしまう状況のことを過学習というが， これについては次節で詳しく確認していく。 ここでは過学習を防ぐための手法の 1 つとして，正則化を紹介する。

正則化の枠組みでは，回帰係数 $\boldsymbol{\beta}$ が過剰に大きな値をとることに対してペナルティを与える。 このペナルティの関数を $E_\beta(\boldsymbol{\beta})$ とおくと， 正則化を行った損失関数 E_r は以下のようになる。

$$E_r = E_{OLS} + \alpha E_\beta(\boldsymbol{\beta}) \tag{4.10}$$

ここで α はペナルティの大きさを決める係数である。

よく使われる正則化の方法として，Ridge 回帰は

$$E_\beta(\boldsymbol{\beta}) = \sum_d \beta_d^2 = \boldsymbol{\beta}^T\boldsymbol{\beta} \tag{4.11}$$

LASSO 回帰は

$$E_\beta(\boldsymbol{\beta}) = \sum_d |\beta_d| \tag{4.12}$$

と正則化項をおく。ここで，$\boldsymbol{\beta} = (\beta_1, \ldots, \beta_D)^T$ とした。

Ridge 回帰の場合，最小二乗法と同様に解析的に $\boldsymbol{\beta}$ の値を求められる。 このときの $\boldsymbol{\beta}$ は，式 (4.6) 内の $\boldsymbol{\Phi}^T\boldsymbol{\Phi}$ を $\boldsymbol{\Phi}^T\boldsymbol{\Phi} + \alpha\mathbf{I}$ におきかえて，

$$\boldsymbol{\beta} = (\boldsymbol{\Phi}^T\boldsymbol{\Phi} + \alpha\mathbf{I})^{-1}\boldsymbol{\Phi}^T\boldsymbol{y} \tag{4.13}$$

となる。式 (4.6) の導出では $\boldsymbol{\Phi}^T\boldsymbol{\Phi}$ が正則であることを仮定したが，これが正則でない場合も， Ridge 正則化の下ではこれを正則にできることがわかる。

一方で LASSO は Ridge に比べて求解が難しいものの， 回帰係数のうちいくつかのものがゼロとなるように推定されるというメリットがある。

4.4　ベイズモデル

最尤推定法の限界として，最尤推定値以外での尤度関数の情報をすべて捨てているという点が挙げられる。 例えば，尤度関数が大きくなる推定値が複数あったとする。 もし尤度関数がパラメータの「尤もらしさ」を示すものであると解釈するならば， 最尤推定法は無数に存在するパラメータの候補の中から「最も尤もらしい」ものだけを採用して， それ以外はすべて捨ててしまう。 しかし，尤度関数はパラメータに関する多くの情報を含むので，最尤推定値以外の尤度も全て参照して，モデルを構成する方法があってもよい。

ベイズモデルでは，パラメータに事前分布をおくことで，このようなモデル構成を実現する。 例えば，回帰係数 $\boldsymbol{\beta}$ に事前分布 $p(\boldsymbol{\beta})$ をおき， 分散 σ^2 には事前分布をおかずにこれをハイパーパラメータとしよう。 このとき，条件付き確率分布 $p(\boldsymbol{y} \mid \mathbf{X}, \sigma^2)$ は，

$$p(\boldsymbol{y} \mid \mathbf{X}, \sigma^2) = \int p(\boldsymbol{y} \mid \mathbf{X}, \boldsymbol{\beta}, \sigma^2) p(\boldsymbol{\beta}) d\boldsymbol{\beta} \tag{4.14}$$

となる。$\boldsymbol{\beta}$ の関数としての尤度関数 $L(\boldsymbol{\beta}) = p(\boldsymbol{y} \mid \mathbf{X}, \boldsymbol{\beta}, \sigma^2)$ を考えたとき，この条件付き分布は（事前分布のサポートがある範囲における）あらゆる $\boldsymbol{\beta}$ に対する尤度関数の情報を参照している。このような方法をベイズ推論という。

条件付き確率分布 $p(\boldsymbol{y} \mid \mathbf{X}, \sigma^2)$ のことを周辺尤度という。周辺尤度はハイパーパラメータ σ^2 や事前分布 $p(\boldsymbol{\beta})$ を決定するハイパーパラメータ[*1]の関数としても見なせるので，周辺尤度を最大化するようにハイパーパラメータを選択することもできる。 このような方法を第二種の最尤推定法や経験ベイズ法とよぶことがある。 しかし完全にベイズ的な枠組みでは，ハイパーパラメータに対してさらに事前分布をおいて推論を行う。

回帰係数 $\boldsymbol{\beta}$ の値を推論したい場合は， ベイズの定理を用いることによって， 形式的には以下のようにパラメータの事後分布が得られる。

$$\begin{aligned} p(\boldsymbol{\beta} \mid \boldsymbol{y}, \mathbf{X}, \sigma^2) &= \frac{p(\boldsymbol{y} \mid \mathbf{X}, \boldsymbol{\beta}, \sigma^2) p(\boldsymbol{\beta})}{p(\boldsymbol{y} \mid \mathbf{X}, \sigma^2)} \\ &= \frac{p(\boldsymbol{y} \mid \mathbf{X}, \boldsymbol{\beta}, \sigma^2) p(\boldsymbol{\beta})}{\int p(\boldsymbol{y} \mid \mathbf{X}, \boldsymbol{\beta}, \sigma^2) p(\boldsymbol{\beta}) d\boldsymbol{\beta}} \end{aligned} \tag{4.15}$$

これを用いれば，新規データに関する不動産特性 $\boldsymbol{x}_*$ が得られたときの価格 y_* の予測分布は以下のように構成できる。

$$p(y_* \mid \boldsymbol{x}_*, \boldsymbol{y}, \mathbf{X}, \sigma^2) = \int p(y_* \mid \boldsymbol{\beta}, \boldsymbol{x}_*, \sigma^2) p(\boldsymbol{\beta} \mid \boldsymbol{y}, \mathbf{X}, \sigma^2) d\boldsymbol{\beta} \tag{4.16}$$

ベイズ法ではこれまで見てきた手法とは異なり，$\boldsymbol{\beta}$ の推定値はある 1 つの値としてではなく，確率分布として与えられる。 もし $\boldsymbol{\beta}$ を点推定したい場合，いくつかの手法が考えられるものの， よく用いられる

[*1] ここでは明示していないが，例えば事前分布がガウス分布で与えられているとするならば，その平均や分散がハイパーパラメータとなる

のは MAP(Maximum a posterior) 推定法である。MAP 推定ではその名の通り，事後確率を最大化する値をパラメータの推定値として採用する。すなわち，回帰係数の MAP 推定値 $\boldsymbol{\beta}_{MAP}$ は，

$$\begin{aligned}\boldsymbol{\beta}_{MAP} &= \arg\max_{\beta}\ p(\boldsymbol{\beta} \mid \boldsymbol{y}, \mathbf{X}, \sigma^2) \\ &= \arg\max_{\beta}\ \log p(\boldsymbol{\beta} \mid \boldsymbol{y}, \mathbf{X}, \sigma^2) \\ &= \arg\max_{\beta}\ \left\{\log p(\boldsymbol{y} \mid \mathbf{X}, \boldsymbol{\beta}, \sigma^2) + \log p(\boldsymbol{\beta})\right\} \end{aligned} \tag{4.17}$$

で与えられる。 なお，式 (4.17) において，事前分布 $p(\boldsymbol{\beta})$ を平均ゼロのガウス分布とするならば，MAP 推定が Ridge 回帰と一致することが確認できる [*2]。

4.5 分位点回帰

4.5.1 分位点

線形回帰モデルにおける最尤推定法の議論から分かるように，前述した線形回帰モデルは不動産価格の期待値（平均値）に着目したモデルである。 しかし，関心がある統計量は必ずしも期待値であるとは限らず，価格分布上の他の値を知りたい場合もあるだろう。 このような要求に応える手法として，分位点回帰法 (Koenker & Bassett 1978[3]) を紹介する。

図 4.2 は分位点回帰法を用いて推定した条件付き分位点の例である。確かに 2.5% 点と 97.5% 点の間にほとんどのデータが収まっていることが確認でき，分位点回帰による分布の予測が実現されていることがわかる。

分位点回帰を理解するために，まずは分位点 (Quantile) について概説しよう。A なる事象が起こる確率を $P(A)$ と表すとする。 q 分位点 $(0 < q < 1)$ とは，単変量の分布 $p(x)$ について， $P(x \leq \theta_q)$ と $P(\theta_q < x)$ が $q : 1-q$ となるように分布 $p(x)$ を分割する点 θ_q のことである。特に $q = 0.5 = 50\%$ のとき，この点を中央値 (median) とよぶ。

もしデータ $\boldsymbol{x} = (x_1, \ldots, x_n)^T$ が与えられているならば， このデータに対する θ_q の推定値は，損失関数

$$\sum_{n \in \{n:\theta_q \leq x_n\}} q|x_n - \theta_q| + \sum_{n \in \{n:x_n < \theta_q\}} (1-q)|x_n - \theta_q| \tag{4.18}$$

を最小化する θ_q として与えられる。もしあらゆる q に対応する θ_q が求められているなら， これらの分位点から分布全体の形状がわかることになる。

4.5.2 分位点回帰

[*2] このことは Ridge 回帰における係数 α を経験ベイズ法で選択できることを示唆するが， $\boldsymbol{\beta}$ を周辺化することが前提となっている

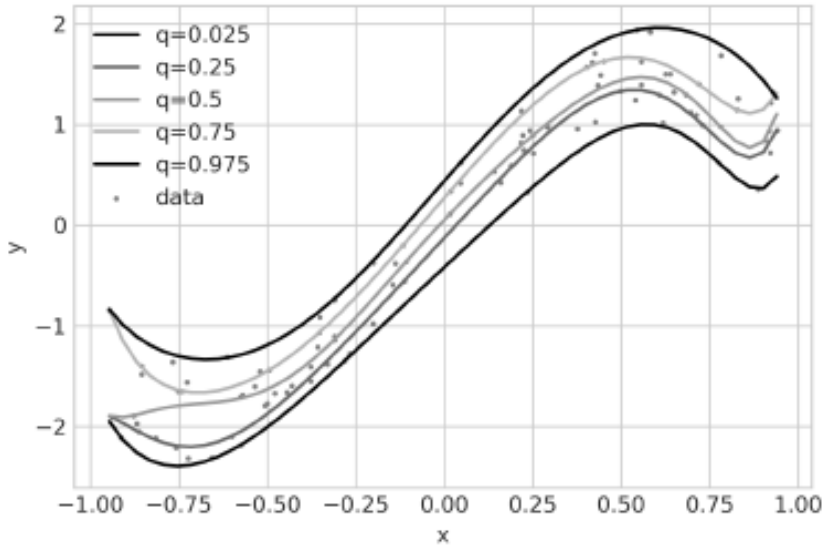


図 4.2　分位点回帰

分位点回帰 (QR: Quantile Regression) では，不動産価格の条件付き分布 $p(y_n \mid \boldsymbol{x}_n)$ を考える。このとき，条件付き q 分布点 $\theta_q \mid \boldsymbol{x}_n$ が線形モデル $\theta_q \mid \boldsymbol{x}_n = \boldsymbol{\beta}_q^T \boldsymbol{\phi}(\boldsymbol{x}_n)$ で与えられるとすると，$\boldsymbol{\beta}_q$ は損失関数

$$E_{QR} = \sum_{n \in \{n:\boldsymbol{\beta}_q^T \boldsymbol{\phi}(\boldsymbol{x}_n) \leq y_n\}} q \left| y_n - \boldsymbol{\beta}_q^T \boldsymbol{\phi}(\boldsymbol{x}_n) \right| + \sum_{n \in \left\{n:y_n < \boldsymbol{\beta}_q^T \boldsymbol{\phi}(\boldsymbol{x}_n)\right\}} (1-q) \left| y_n - \boldsymbol{\beta}_q^T \boldsymbol{\phi}(\boldsymbol{x}_n) \right| \tag{4.19}$$

を最小化する $\boldsymbol{\beta}_q$ である。 分位点のときと同様に，もしあらゆる q に対応する $\boldsymbol{\beta}_q$ が 求められているなら，条件付き分布の形状全体がわかることになる。 つまり，ある不動産の特性 $\boldsymbol{x}_*$ を持つ不動産の集合の価格が，どのような分布に従っているかが予測できる。

$q = 0.5$ としたとき，分位点回帰モデルは中央値を予測する。このモデルは通常の線形回帰モデルと類似しているように思われるが，重要な特長として，外れ値に対してより頑健であることが挙げられる。

図 4.3 は最小二乗法による線形回帰モデルと 中央値回帰モデルを比較したものである。この図は 100 個のサンプルからランダムに 30 個のサンプルをランダムに選んで，それらを用いて回帰モデルを学習している。 このとき外れ値として 1 つのサンプルにノイズを加えた。 このような試験を繰り返し行うことで，モデルが学習する回帰曲線が安定しているかを確認できる。 図を見ると，最小二乗法に比べて中央値回帰の学習した曲線は蛇行が少なく， 安定していることが読み取れる。 このようなシミュレーションからも，中央値回帰の安定性が確認できる。

4.6　ニューラルネットワーク

4.6.1　基底関数の学習

線形回帰モデルでは，固定された基底関数を用いて回帰を行った。 この自然な拡張として，基底関数の形状自体をデータから学習したいという要求が考えられる。ニューラルネットワークはこうした基底関

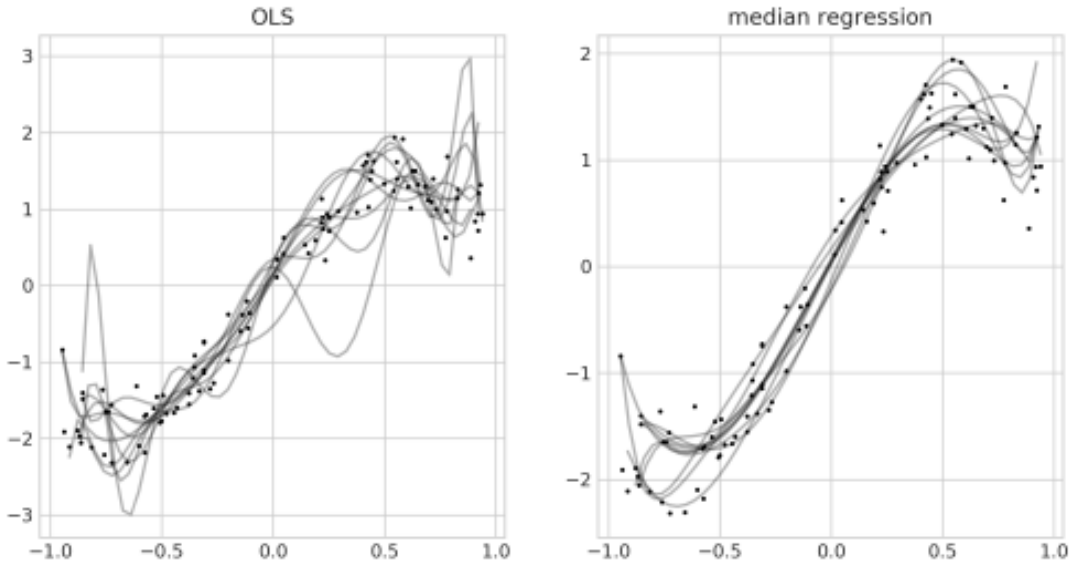


図 4.3 最小二乗法と中央値回帰

数の学習を実現する手法の 1 つである。 ニューラルネットワークのパラメータ推定は解析的にはできないが， その再帰的な構造を利用すれば，パラメータによる損失関数の勾配が効率的かつ厳密に計算できる。そのため，勾配法を用いた最適化がパラメータ推定に広く用いられている。 また，確率的勾配降下法を導入すれば，大規模データに適した推定アルゴリズムが得られる。

図 4.4 はニューラルネットワークを用いて大規模データから学習した 近似関数の例である。ここでは 10,000 個のサンプルからモデルを学習している。 例では関数の形状があまり複雑ではないのでニューラルネットワークの恩恵は あまり感じられないかもしれないが， 複雑な関数であっても大規模データの力を使えば近似できてしまうのが， ニューラルネットワークの強みである。

ニューラルネットワークは高い表現力を持つ反面，その学習は難しい。そのため，数多くの学習法やテクニックが日々開発されている。 こうした研究をすべて追うことは困難であるが，基本的な部分については Bishop (2012)[1] や斎藤康毅 (2016)[4] を参照するとよい。

線形回帰モデルによる予測 $\boldsymbol{\beta}^T\boldsymbol{\phi}(\boldsymbol{x}_n) = t_{D,1}$ を下のように書き直す。

$$t_{D,1} = \sum_j w_{D-1,1,j} h_{D-1,j} + b_{D-1,1} \tag{4.20}$$

ここで $h_{D-1,j}$ は $\boldsymbol{x}_n$ の関数（パラメータを持つ基底関数による変換）である。この予測も $h_{D-1,j}$ に関する線形システムであるから， ニューラルネットワークは線形回帰モデルの拡張と見なせる。

ここで，$h_{d,i}(d = 0, \ldots, D-1)$ は以下のように再帰的に定義される。

$$\begin{aligned} h_{d,i} &= \sigma(t_{d,i}) \\ t_{d,i} &= \sum_j w_{d-1,i,j} h_{d-1,j} + b_{d-1,i} \end{aligned} \tag{4.21}$$

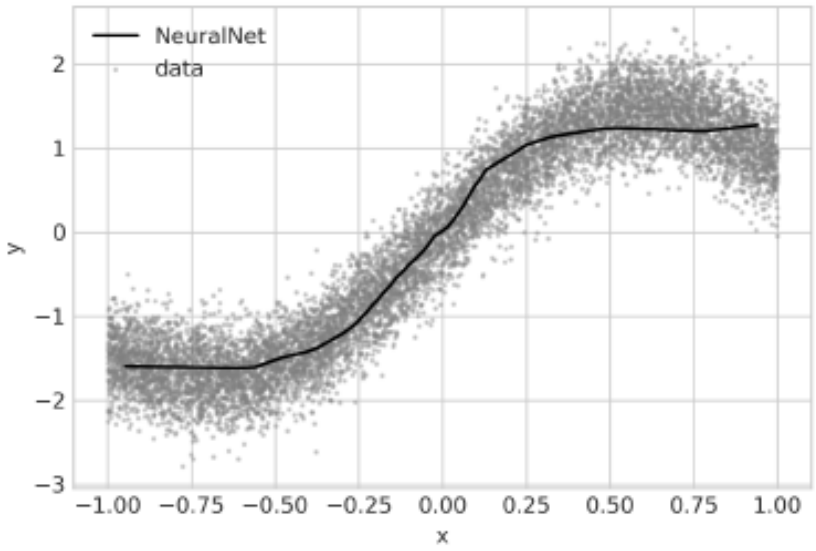


図 4.4　ニューラルネットワーク

ただし $\boldsymbol{h}_0 = (h_{0,1}, h_{0,2}, \ldots)^T = \boldsymbol{x}_n$ であり，σ は $t_{d,i}$ に非線形変換を施す活性化関数である [*3]。再帰的な定義から，D はニューラルネットワークの“層”の深さであると解釈できる。

層の深さと各層の幅は任意であり，特に深いネットワークを深層学習（Deep Learning）と呼ぶことがある。 ここでは全結合のネットワーク構造のみを紹介したが，タスクに応じてネットワーク構造を工夫することも可能である。例えば，近年画像認識で目覚ましい成果を上げているネットワーク群は，その構造から畳み込みニューラルネットワークとよばれている。

4.6.2　誤差逆伝播

ニューラルネットワークのパラメータ $\{w_{d,i,j}, b_{d,i}\}$ は非常に数が多く， 解析的に最適化することはできない。このため，勾配法（勾配降下法）を用いた数値的な最適化を行う。 勾配法を簡単に説明しよう。損失関数を最小化するようなプロセスを考えたとき， もし損失関数をパラメータで微分した勾配が与えられていれば，どの方向にパラメータを動かせば損失が減少するかがわかる。 よって，この勾配情報を使ってパラメータを少しずつ動かし，損失関数を最小化するパラメータを探索するのが勾配法である。

勾配法の適用には損失関数の勾配が要求されるが， ニューラルネットワークの場合，その再帰的な構造を用いることで， 勾配を効率的に（しかも厳密に）求めることができる。 以下で説明するこのような勾配計算のプロセスは， 誤差逆伝播 (Error Backpropagation) とよばれる。

回帰モデルの場合，ニューラルネットワークの損失関数を二乗損失 E_{OLS} とすれば，

$$E_{OLS} = \sum_n (y_n - t_{D,1}(\boldsymbol{x}_n))^2 = \sum_n E_n \tag{4.22}$$

[*3] 活性化関数は層ごとに異なっていても良い

と書ける。ここで $t_{D,1}$ は $\boldsymbol{x}_n$ の関数であり，ニューラルネットワークの任意のパラメータを θ とすれば，

$$\frac{\partial E_{OLS}}{\partial \theta} = \sum_n \frac{\partial E_n}{\partial \theta} \tag{4.23}$$

である。よって，各データに対して $\frac{\partial E_n}{\partial \theta}$ を計算して，それをすべて足せば損失関数の勾配が求められる。

パラメータ $\{w_{d,i,j}, b_{d,i}\}$ による偏微分は，以下のような再帰的な手続きによって求められる。

[1] まず，最終層のパラメータ $\{w_{D-1,1,j}, b_{D-1,1}\}$ による偏微分を求める。予測値 $t_{D,1}$ による偏微分は

$$\frac{\partial E_n}{\partial t_{D,1}} = -2(y_n - t_{D,1}) \tag{4.24}$$

であるが，これと微分の連鎖律 (Chain Rule) を用いれば，パラメータによる偏微分が以下のように求められる。

$$\frac{\partial E_n}{\partial w_{D-1,1,j}} = \frac{\partial E_n}{\partial t_{D,1}} \frac{\partial t_{D,1}}{\partial w_{D-1,1,j}}, \quad \frac{\partial t_{D,1}}{\partial w_{D-1,1,j}} = h_{D-1,j} \tag{4.25}$$

$$\frac{\partial E_n}{\partial b_{D-1,1}} = \frac{\partial E_n}{\partial t_{D,1}} \frac{\partial t_{D,1}}{\partial b_{D-1,1}}, \quad \frac{\partial t_{D,1}}{\partial b_{D-1,1}} = 1 \tag{4.26}$$

また，次の偏微分を [2] で使用する。

$$\frac{\partial E_n}{\partial h_{D-1,j}} = \frac{\partial E_n}{\partial t_{D,1}} \frac{\partial t_{D,1}}{\partial h_{D-1,j}}, \quad \frac{\partial t_{D,1}}{\partial h_{D-1,j}} = w_{D-1,1,j} \tag{4.27}$$

[2] $\frac{\partial E_n}{\partial h_{d,i}}$ が分かっているとき，以下の偏微分が求められる。これにより，再帰的にすべてのパラメータで偏微分が実行できる。

$$\frac{\partial E_n}{\partial t_{d,i}} = \frac{\partial E_n}{\partial h_{d,i}} \frac{\partial h_{d,i}}{\partial t_{d,i}}, \quad \frac{\partial h_{d,i}}{\partial t_{d,i}} = \frac{\partial}{\partial t_{d,i}} \sigma(t_{d,i}) \tag{4.28}$$

$$\frac{\partial E_n}{\partial w_{d-1,i,j}} = \frac{\partial E_n}{\partial t_{d,i}} \frac{\partial t_{d,i}}{\partial w_{d-1,i,j}}, \quad \frac{\partial t_{d,i}}{\partial w_{d,i,j}} = h_{d-1,j} \tag{4.29}$$

$$\frac{\partial E_n}{\partial b_{d-1,i}} = \frac{\partial E_n}{\partial t_{d,i}} \frac{\partial t_{d,i}}{\partial b_{d-1,i}}, \quad \frac{\partial t_{d,i}}{\partial b_{d-1,i}} = 1 \tag{4.30}$$

$$\frac{\partial E_n}{\partial h_{d-1,j}} = \sum_i \frac{\partial E_n}{\partial t_{d,i}} \frac{\partial t_{d,i}}{\partial h_{d-1,j}}, \quad \frac{\partial t_{d,i}}{\partial h_{d-1,j}} = w_{d-1,i,j} \tag{4.31}$$

以上から，解析的に微分可能な活性化関数を採用すれば，ニューラルネットワークの 偏微分は誤差逆伝播の手続きによって解析的に求められることがわかる。

4.7 その他手法

大規模データを用いて学習したいとき，勾配 $\frac{\partial E_n}{\partial \theta}$ をすべてのデータについて計算するのは計算コストが大きい。そこで，式 (4.23) を確率的に近似推定する方法を考える。

もし N 個のデータの中から偏りなく 1 つのデータをランダムサンプリングしたとすると，$N\frac{\partial E_n}{\partial \theta}$ を用いて式 (4.23) の勾配を推定できる。これを用いた勾配降下法を，確率的勾配降下法とよぶ。複数のサンプルを抽出すれば，勾配の推定はより安定する。

確率的勾配降下法の他の特色として，勾配の推定量に常にノイズが入るために， 通常の勾配法とは異なり損失関数が常には減少しない点が挙げられる。 これにより，浅い局所最適解を回避できるとされている。 推定された勾配を使用した具体的なパラメータの更新方法は，様々なものが提案されている。

他の重要な手法として，アンサンブル学習が挙げられる。 アンサンブル学習では，多数の回帰モデルを組み合わせることによって， 予測の安定化と精度向上を図る。 代表的なモデルとして，ランダムフォレスト (Breiman 2001)[2] などが挙げられる。アンサンブル学習では決定木ベースのモデルがよく用いられる。それらの手法については，第 6 章を参照されたい。

4.8 手法の適用

4.8.1 過少定式化バイアス

回帰モデルの典型的な応用例として，各不動産の特性が不動産価格に与える影響を検討するというものが挙げられる [*4]。例えば，線形回帰モデルで不動産価格関数が充分に近似できたとすると，d 番目の特性 $\phi_d(\boldsymbol{x})$ の効果量は β_d なので， 特性 $\phi_d(\boldsymbol{x})$ が 1 単位増加すると β_d だけ価格が上昇すると予想される。すなわち，推定された回帰係数 $\boldsymbol{\beta}$ から各特性が不動産価格に与える影響を検討できる。

しかしながら，不動産価格に影響する特性をすべてデータとして取得するのは， 現実には困難である。この場合，仮に不動産価格が本当に線形回帰モデルから生成されている場合であっても， 入手できなかった特性の存在によって，線形回帰モデルの回帰係数の推定量にバイアスが生じる可能性がある。これを過少定式化バイアスとよぶ。入手できなかった特性が価格に強く影響している場合や， また入手できなかった特性と入手できた特性の間に強い相関がある場合， 過少定式化バイアスが発生する懸念は強くなる。

過少定式化バイアスの発生を，数値的なシミュレーションを行って確認してみよう。 不動産特性と価格を以下のように生成する。 まず 2 つの特性を，これらが相関を持つように疑似乱数で生成する。その後，適当に生成した回帰係数を持つ線形回帰モデルを用いて， これらの特性から不動産価格を生成する。このようにすれば，真の回帰係数が分かっているから，線形回帰モデルを推定したときの回帰係数の推定誤差を検証できる。

図 4.5 は 500 個のサンプルを用いて 10,000 回推定を行い，観測された推定バイアスをヒストグラムに整理したものである。 ここでは，価格に影響する不動産特性は両方分かっていると仮定している。 図から分かるように，推定誤差は 0 を中心として分布することがわかる。 このように完全なデータを用いれば，平均的には正しく回帰係数が推定できることがわかる [*5]。

一方で図 4.6 は，一方の特性の回帰係数を推定する際には，もう一方の特性が分からないという仮定のもので推定した結果である。 つまり，過少定式化バイアスが発生するような状況をシミュレーションしている。この場合は，完全なデータを用いたときと比べて推定誤差のばらつきが大きいだけでなく，分布の中心も 0 になっていないことがわかる。 つまり，線形回帰モデルを推定したとき，平均的に見ても回

[*4] 通常の回帰モデルの枠組みの中では，変数間の因果関係に関する知見は本来得られない． ここで「影響」といっているのは，あくまで不動産価格がその特性によって決定されるという仮定のもとで導かれるものである．

[*5] ただし，ここでは価格を生成する真の構造が線形回帰モデルであると分かっていることが仮定されていることに注意されたい．実際の不動産価格は線形回帰モデルから生成されているわけではないので， 実際に推定される価格関数はあくまで近似に過ぎない．

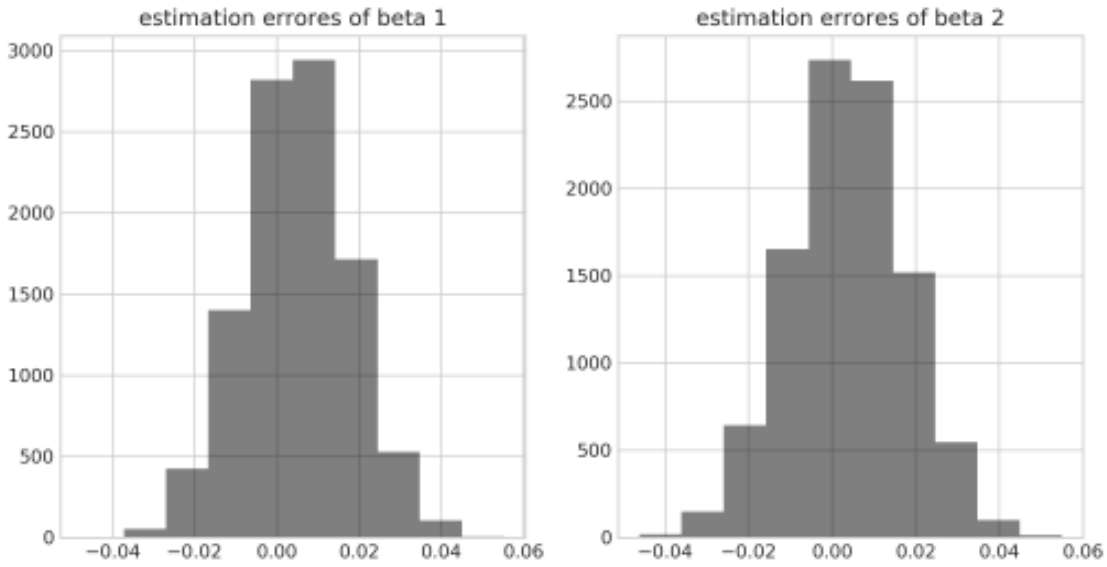


図 4.5　欠測のないデータを使用したときの推定誤差

帰係数の推定に誤差が生じることが期待される。

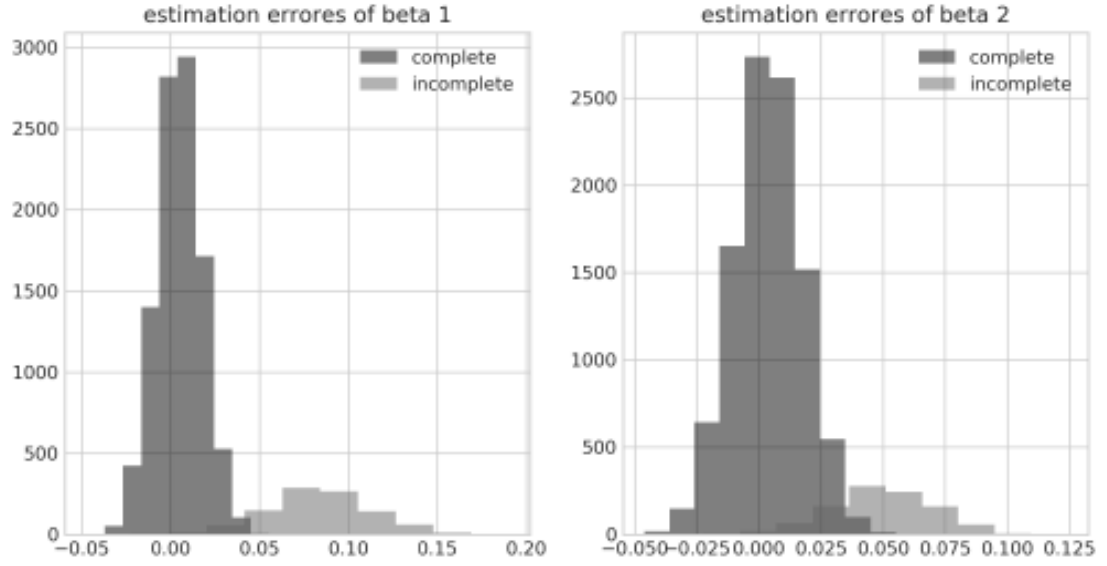


図 4.6　欠測があるデータを使用したときの推定誤差

実際の不動産分析においては，以下のような状況が過少定式化バイアスの例として想定しうる。 まず自然公園などの公共空間に近いと，不動産価格が少し上昇するとしよう。 このとき，自然公園までの距離は不動産価格に負の効果を持つ（近いほど価格が上がる）。 一方で，駅までの距離は不動産価格により大きな影響があると考えられる。しかし分析者の手違いで，回帰モデルに駅までの距離を含めるのを忘れてしまったとする。自然公園が駅の近くに立地していることは稀なので，自然公園までの距離と駅までの

距離は負の相関を持つ。よって，このとき自然公園までの距離の小ささは駅までの距離の大きさとして解釈され，結果として自然公園までの距離が不動産価格に正の効果があるかのように推定されてしまう（つまり，遠いほど価格が上がると推定される)。 これが過少定式化バイアスである。

このような過少定式化バイアスを防ぐためには，価格に強く影響していると思われる不動産特性をあらかじめ見極め，それらを適切にモデルに組み込むことが求められる。

4.8.2 過学習

近年は機械学習とよばれる非常に柔軟かつ複雑なモデル群が， その予測精度の高さを喧伝されている。本章で紹介した中では，ニューラルネットワークがそれに該当する。 これらの手法は様々な関数を近似することができるが，その近似能力の高さは過学習とよばれる問題も引き起こす。

回帰モデル $f(\boldsymbol{x})$ の学習は，データ $\{y_n, \boldsymbol{x}_n\}_{n=1,\ldots,N}$ を用いて損失 E を計算し，それを最小化することで達成される。二乗損失などの通常の損失関数であれば, 全体の損失は不動産ごとの予測損失 E_n の和で表され, かつそれは実際の価格 y_n と予測価格 $f(\boldsymbol{x}_n)$ によって決定される。つまり, $E = \sum_n E_n(y_n, f(\boldsymbol{x}_n))$ である。 ここから，学習済みのモデルは学習に使用したデータに関して，平均的な予測損失を最小化していると考えられる。

ここで，もしサンプルサイズに比べて圧倒的に複雑なモデルを採用したとする。 このようにモデルが過剰に柔軟な場合，得られたデータをすべて“丸暗記”できてしまう可能性がある。 もしデータを丸暗記したならば，データに対する予測損失は 0 となる。

しかしながら，モデルの学習における目的の 1 つは，いまだ得られていないデータ $\{y_*, \boldsymbol{x}_*\}$ に関して，良い予測をすることである。 すなわち，新規データに関する予測損失 $E_*(y_*, \boldsymbol{x}_*)$ を（平均的に）最小化するようにモデルを学習したい。上記で示したような“丸暗記”したモデルは，この目的に適しているとは限らない。というのも，現実の不動産価格予測は暗記テストではないので， 丸暗記したモデルの予測はむしろ大きく外れるということがありうるからである。

このような状態を，過学習とよぶ。

図 4.7 は，線形回帰モデルの学習において， 過剰に高次の多項式基底関数を採用した場合の学習例である。この例では回帰曲線がデータの近傍を通るために，過剰に蛇行している。その結果，新しくデータが得られた際におよそあり得なさそうな価格をモデルが予測してしまっていることがわかる。

これは過学習の典型的な例である。

過学習を避ける方法としては，データを学習用と検証用に分割する方法が考えられる。 つまり，学習用データで学習したモデルの精度を検証用データを用いて確認するのである。また，Ridge 回帰のような正則化も有効である。 線形回帰モデルのような情報量規準が容易に計算できるモデルならば， それを用いて変数選択を行うこともできる。 例えば今回の場合であれば，変数選択によって多項式基底関数の次数を適切に削減することで， 過学習を回避できる。

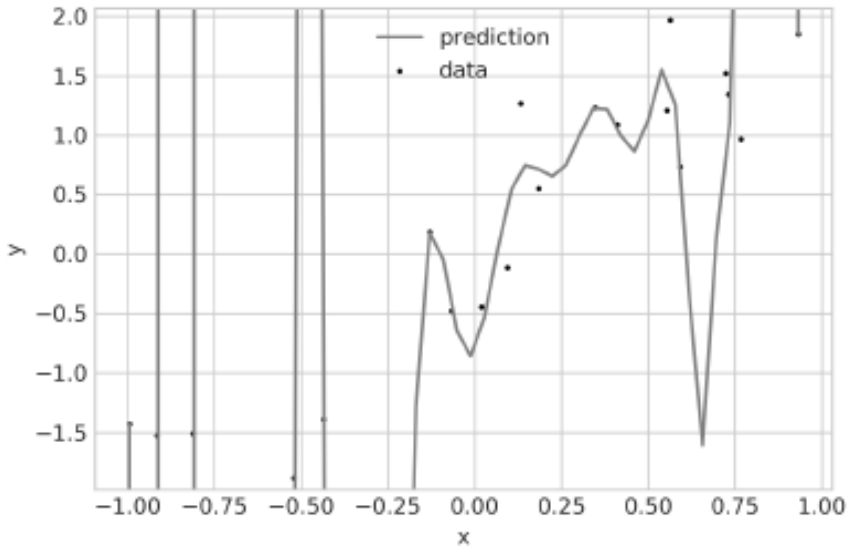


図 4.7 過学習の例

4.9 バイアスとバリアンス

ここでは過学習が起こる原因について，予測のバイアスとバリアンスの観点から， シミュレーションを用いた直感的な説明を通して整理する。この議論に関する数学的な詳細は Bishop (2012)[1] を参照されたい。

もしこの世に存在するすべての不動産のデータを持っていれば， （このような条件で価格予測が必要なのかという疑問はあるものの） 予測誤差が最も小さくなるような理想的な関数を作ることができそうだ。 しかし実際に入手可能なデータは，世の中に存在するうちの一部の不動産に関するものだけなので，このような一部のデータから（データとして取得できなかった不動産の価格についても）予測誤差の小さい関数を学習したい。

注意するべきなのは，この世に存在するすべての不動産から一部の不動産のデータを得る際に，どの不動産のデータが得られるか，という点については偶発的な要因が絡むという点である。

あり得ない仮定であるが，データの取得とモデルの学習を何度も繰り返し行うことができるとしよう。データの取得には偶発的な要因が絡むので，そのとき「たまたま」得られたデータに依存して，そのとき学習されたモデルの関数形も異なってくると予想される。

図 4.8 はこのような仮定をシミュレーションを用いて実践した結果である。ここでは 100 個のデータから 30 個のサンプルをランダムにとって， それを用いてモデルを学習する，という手続きを 10 回繰り返している。 ここでは多項式基底関数による線形回帰モデルを最小二乗法で学習した。 多項式の次数は，1, 3, 10, 100 の 4 通りで試している。

この結果を見ると，以下の興味深い知見が得られる。まず明確にわかることは，推定される曲線の形状が次数が高くなるほどにばらついているということである。このようなばらつきの大きさは，データのと

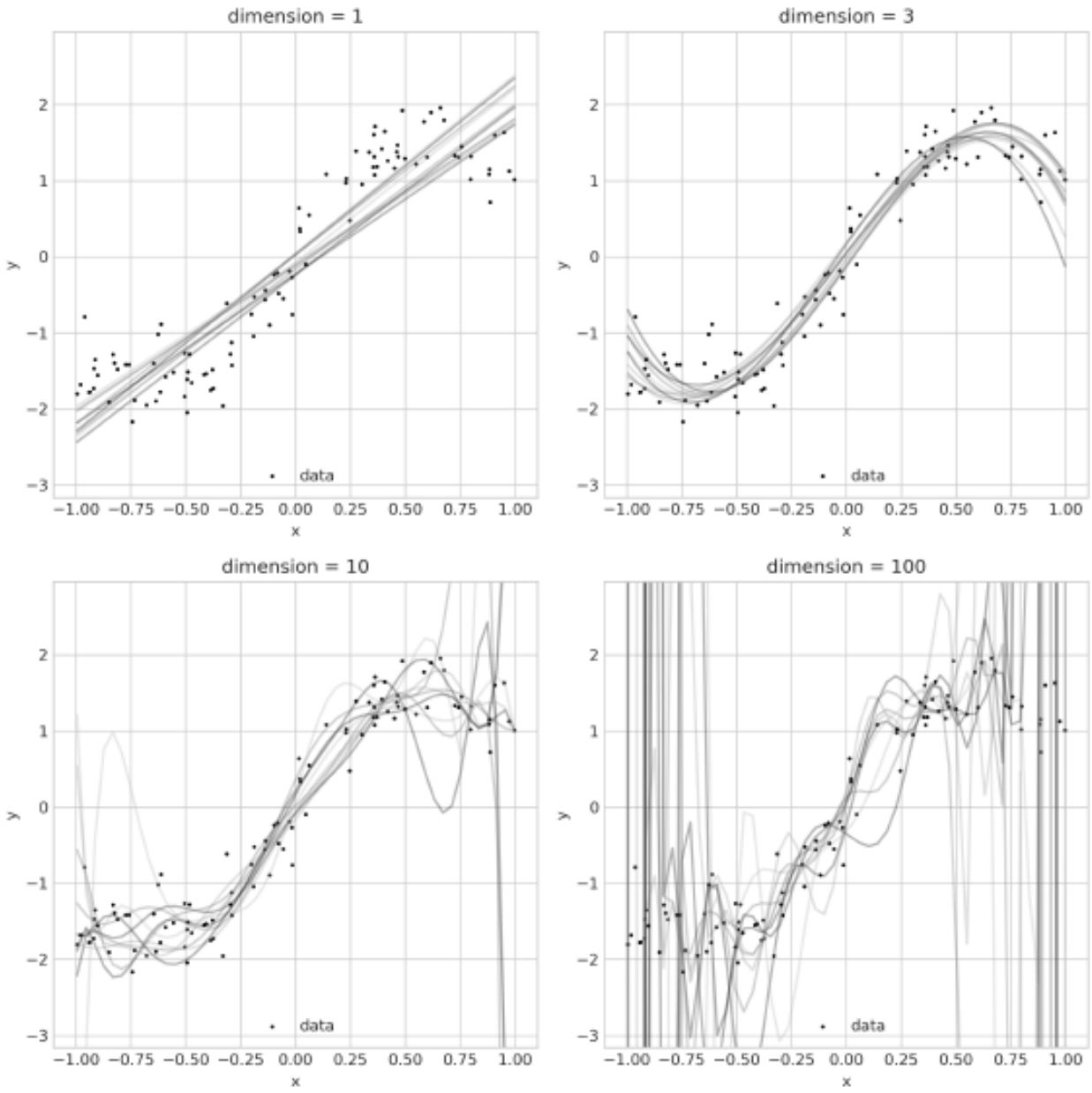


図 4.8　学習されたモデルの偶発的なばらつき

り方が学習される回帰モデルの形状に大きく影響するということを示唆しており，望ましくない。このようなばらつきのことを，予測のバリアンスという。 これは言い換えれば，偶然得られたデータに回帰モデルが必要以上に適合してしまっているということであるので，バリアンスの大きさは過学習のしやすさと強い関係があるといえる。

一方でよく見ると，高次のモデルであっても多数ある曲線の平均をとれば， データ全体をうまく予測できていそうに見える。このように，多数のモデルを学習できたときに，それらの予測を平均値を全体としての予測として取り扱うことを考えよう。このような平均値による予測の悪さを，予測のバイアスとい

う。 高次のモデルはバリアンスが大きいので予測が悪そうに見えるが， バイアスの観点で見ると，実は予測が良い。 一方で次数が低すぎるモデル（ここでは次数 1 のモデル）は実際の価格形成の構造に対してモデルの柔軟性が足りていないので，どの回帰モデルの予測もさほど高くなっていない。 この結果，これらの平均をとっても予測は悪く，バイアスが大きい。

この例から分かるように，バイアスとバリアンスはトレードオフの関係にある。 「データの取得」という偶発的な要因に影響されずに良い予測モデルを作るためには， バイアスもバリアンスも適度に小さくなるような，ちょうどいい柔軟性を持つモデルを選ぶことが重要であることがわかる。

高いバリアンスは過学習の原因であるので，これを適度に抑制する方法が望まれる。 すでに紹介したモデルの正則化は，バリアンスを低減するのに有効な手法の 1 つである。 図 4.9 は推定法を最小二乗法から Ridge 回帰に変えて， 全く同じシミュレーションを行った結果である。ここでは正則化の強さは緩めに設定しているが，バリアンスが大幅に低減され， 100 次元のモデルでも関数が安定していることがわかる。

しかしながら，バリアンスの過剰な抑制はバイアスの増大を招くので，正則化を行う場合でもその強さは適切に調整する必要がある。

今回のシミュレーションでは「データの取得」というプロセスを繰り返し実行できたので，図からどの程度バリアンスを抑制すればよいのか判断することができた。しかし実際には，データの取得は一度しかできないことが多いので， 与えられた 1 セットのデータから適度な複雑さを持ったモデルを選んだり，正則化の強さを適切に調整したりしなければならない。すでに述べたように，データを学習用と検証用に分割する方法は， この課題を解決する 1 つの方法である。 しかしデータが限られている場合は，手持ちのデータすべてを使ってモデルを学習させたい場合もあるだろう。

ここまでの議論を踏まえると，より良い予測モデルを作るためのいくつかの方法を新たに考えることができる。

まず，バイアスのみを見れば，複雑なモデルの予測が良さそうであることに着目しよう。複雑なモデルの問題点はバリアンスが大きいことであるが，バイアスは小さいので， 多数のモデルの平均をとれば，良い予測ができそうである。アンサンブル学習はこのようなアイデアを実現する方法である。 具体的な手順の一例としては，得られたデータからランダムにサンプルを抽出し， モデルに学習させる。これを何度も繰り返すと多数のモデルが得られる。 これらのモデル群すべてに予測を行わせ，その平均を全体の予測とするのである。

もう 1 つの方法として，予測のバリアンスが大きいことは，推定されたパラメータがばらついていることに由来している，という点に着目することが考えられる。多数のモデルを平均化するということは，様々なパラメータについてモデルの予測を構築してその平均をとっているのであるから， 究極的にはありとあらゆるパラメータを使って予測を行い，それを平均化すればよい。 ここでベイズ推論の式を見ると，ベイズ予測では不動産価格の条件付き確率に事前分布や事後分布をかけて積分しているので， これはあらゆるパラメータを用いた予測を事前分布や事後分布で重みづけして平均化していることに相当することがわかる。

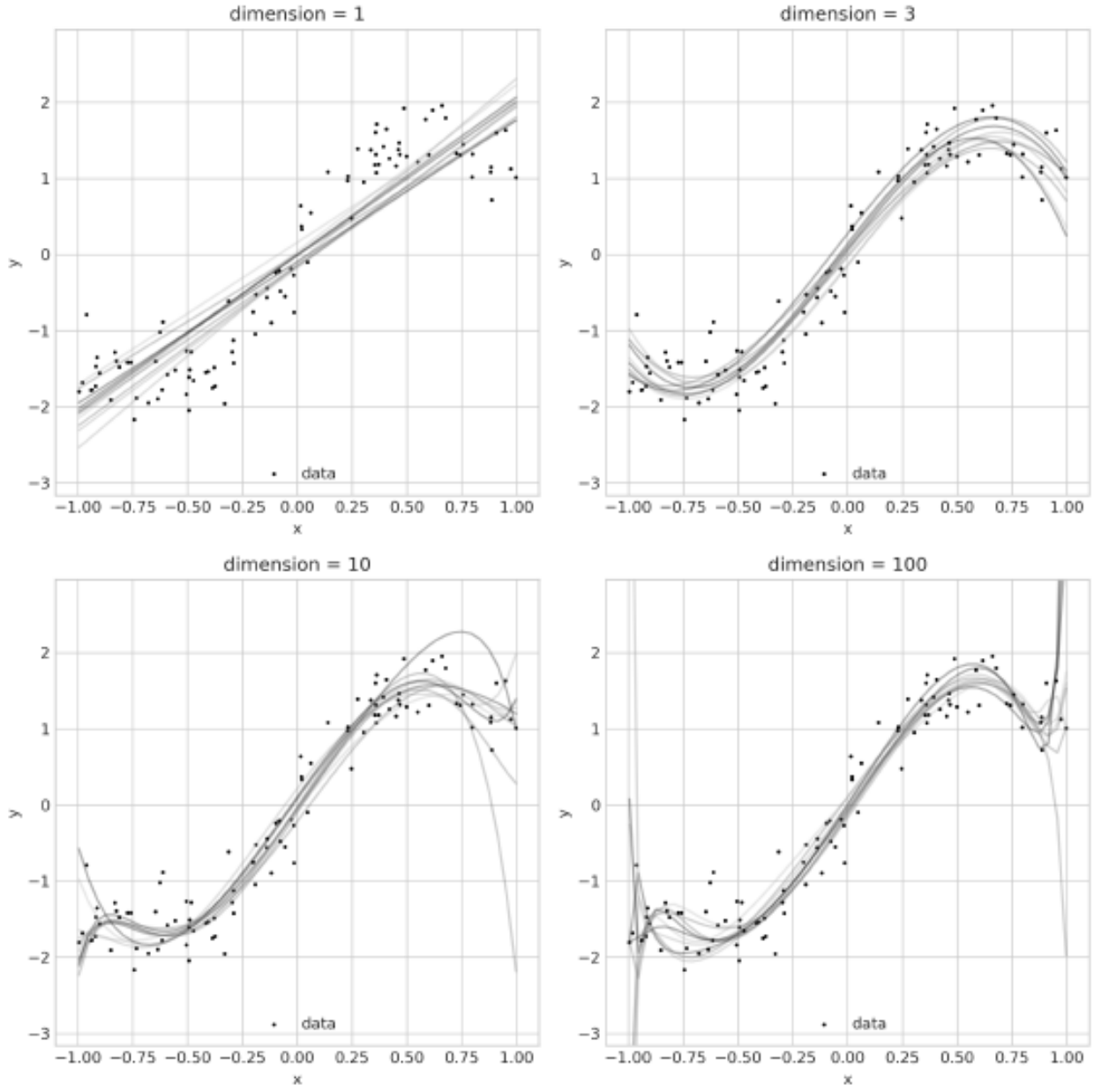


図 4.9　正則化によるバリアンスの低減

参考文献

[1] Bishop, C. (2012), パターン認識と機械学習上：ベイズ理論による統計的予測, 丸善.

[2] Breiman, L. (2001), 'Random forests', Machine Learning 45(1), 5-32.

[3] Koenker, R. & Bassett, G. (1978), 'Regression Quantiles', Econometrica 46(1), 33.

[4] 斎藤康毅 (2016), ゼロから作る Deep Learning——Python で学ぶディープラーニングの理論と実装, オライリー・ジャパン.

[5] 清水千弘& 唐渡広志 (2007), シリーズ：応用ファイナンス講座 4 不動産市場の計量経済分析, 朝倉書店.

第 5 章

不動産テックにおける GIS の理論と実際

5.1 GIS の概念

5.1.1 データ利用

近年，様々なデータが入手可能になっており，特に空間座標が付与されたデータは不動産市場を分析する際に有用である。例えば，最寄り駅までの距離を測定したい場合，駅と物件までの緯度経度座標があれば計算可能であり，徒歩 10 分圏内のスーパーの店舗数なども瞬時に把握できる。このようなデータベースは分析者に直観的な理解を促すとともに，資本化仮説に基づくヘドニック価格推定法の説明変数として有用である。本章では，GIS の理論について概観し，不動産分析に関連する空間集計と解析的手法について説明する。

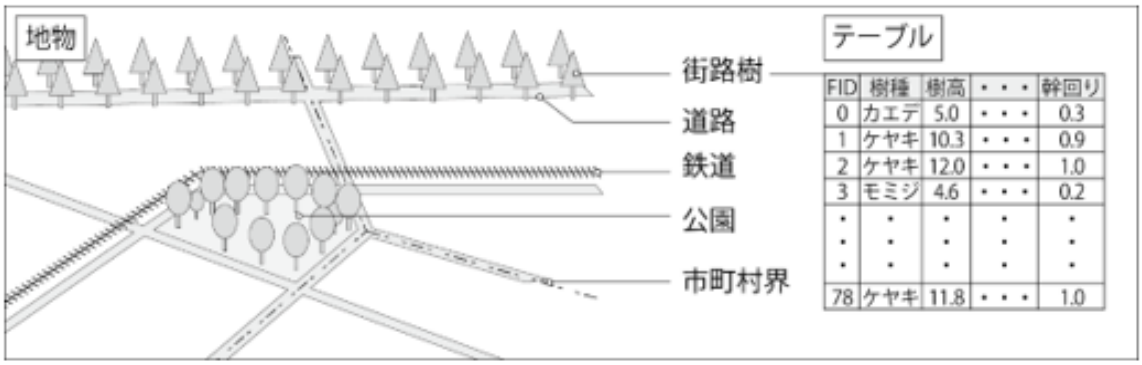


図 5.1　地物とテーブルのイメージ

GIS（Geographic Information System）は空間に関わるストック，フローなどを地物として表示し，分析を行うシステム全般を指す。地物とは空間上に存在する街路樹，住宅，鉄道などの点，線，面的オブジェクトであり，さらに GIS では実際には見えない境界線，用途地域なども地物として捉えられる。このような情報は表形式で格納されたテーブルに格納され，行はレコード，列はフィールドと呼ばれる。各フィールドに対して数値，文字列などのデータを格納することができ，街路樹の場合，樹種や幹回りなどが該当する（図 5.1）。このように，GIS ではこのような地物の重ね合わせによって，空間の見通しを良く

するシステムといえる。GIS で利用するデータは一般的な統計で利用されているデータとは異なり，座標が付与されている。GIS は世界共通で利用できる反面，そのデータ構造や投影法などを正確に理解し，定義する必要がある。

5.1.2 データ形式とそのモデル化

データ形式には大きく分けてラスター（raster）形式とベクター（vector）形式が存在する（図 5.2）。前者はデジタル写真や気温図など，データがセル内に格納されており，行と列に整理されている。ラスターデータはその構造がシンプルであるため，降雨量など連続的な変化量の可視化やメッシュベースの空間解析に優れているが，地点間の距離測定などは難しい。後者は幾何学的解析に優れている一方で，空間的な連続量を可視化するためには空間補間（spatial interpolation）を行い，連続量の推計を行う必要がある。なお，不動産市場の分析では衛星画像などのラスターデータを用いるよりもベクターデータの方が主であるため，以降ベクターデータを対象として説明する。

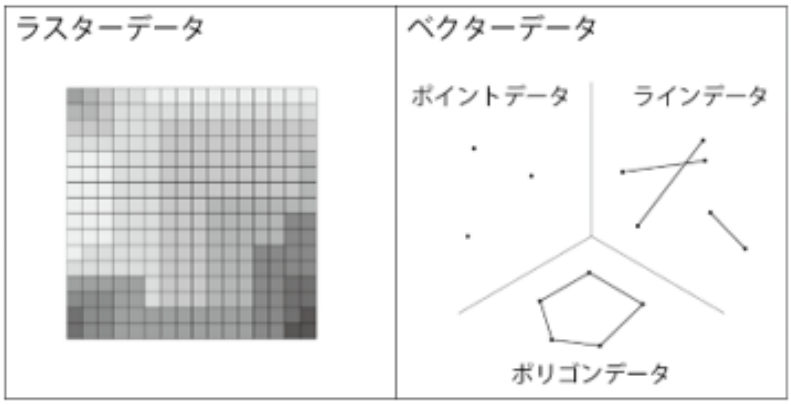


図 5.2 ラスター形式とベクター形式の差異

5.1.3 ジオコーディングと座標系

得られた情報がテキストベースの住所である場合，GIS 上にマッピングするには緯度経度座標を付加する必要がある。このような操作をジオコーディング（geocoding）という（図 5.3）。これは不動産取引データをポイントベクターとして扱う際に必要となるため，重要な前処理である。2019 年 10 月現在，東京大学空間情報科学研究センターが提供する CSV アドレスマッチングサービス（http://newspat.csis.u-tokyo.ac.jp/geocode/）などを用いることにより，住所の表データから緯度経度座標を自動的に生成することができる。

ジオコーディングしたデータを GIS 上に投影する際にもいくつかの注意がある。まず，測地系とよばれる特定の空間座標を示すために必要な測量方法及び基準の体系である。我が国では日本測地系と世界測地系が用いられている。日本測地系はベッセル楕円体に基づく測地系であり，旧国立天文台跡地を経緯度原点として定めている。一方で，世界測地系は人工衛星の計測に基づく，地球形状を高い精度で測量した

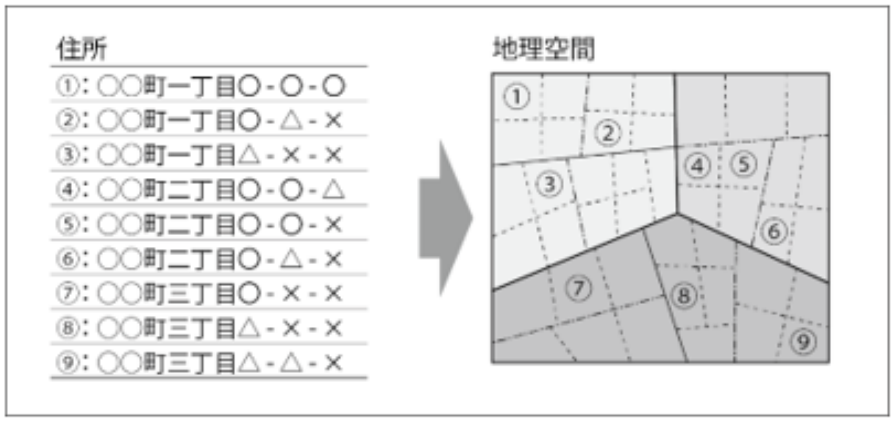


図 5.3　ジオコーディングのイメージ

測地系である。日本測地系は独自の測地系であるため，航空機の GPS による把握などの際，世界測地系と相対誤差が生じる。そのため，2002 年の測量法改正以降，国土地理院は我が国の測量基準を世界測地系としている。世界測地系の例としては，WGS84 や JGD2000 などがある。

座標の表現についても，いくつかの方法がある。代表的なものは緯度経度を度数で表す地理座標系である。これは，設定した測地系の楕円形上にある座標を表現しており，正確な位置が捕捉される。しかしながら，実務では位置・方向・距離等を平面上に投影して測量計算するほうが簡便に処理できる。これは平面直角座標とよばれ，現在でも公共測量で用いられている。投影法はガウス・クリューゲル図法を採用し，我が国では 19 の座標系が存在している。例えば，東京都は第 9 系に位置し，座標原点から東西約 130km が適用範囲である。

5.1.4　利用可能な不動産関連データの例

現在，我が国の基盤となる地図情報の多くは GIS データとして利用可能であり，分析者の用途に応じてデータを取得することができる。

国土地理院は主に我が国での基本的な地図情報を公開しており，道路，建物，行政界等を提供する基盤地図情報や，土地利用，ハザードマップ，都市施設などを幅広くカバーする国土数値情報がある。前者は全国で整備されており，都市計画区域内では縮尺 1/2,500 で，都市計画区域外では縮尺 1/25,000 でそれぞれ提供されている。有償の住宅地図が利用出来ない際，基盤地図情報をベースマップとして利用するのに優れている。後者は幅広いデータを提供しているため，分析者の関心に合わせてダウンロードするのが良い。不動産分析に関係するものとして，例えば地価公示ポイント，学校などの公共施設ポイント，用途地域ポリゴン，浸水想定区域ポリゴンなどがある。加えて，将来人口推計の 500m 及び 1,000m メッシュも提供されているため，将来時点でのシミュレーションを行う際に有用である。

総務省は小地域，メッシュ単位で境界データを提供しており，国勢調査や経済センサスなどと対応させることができる。国勢調査は人口構成や建物の建て方別世帯数などを公開しており，信頼性の高い結果を空間上で得ることができる。経済センサスでは産業別・従業者規模別全事業所数などがわかるため，例え

ば不動産価格と商業集積との関係を分析する際に利用を検討される。

近年，民間企業が構築しているデータにも注目が集まっている。2019 年 10 月現在，東京大学空間情報科学研究センターは JoRAS (Joint Research Application System) という共同研究利用システムを運営しており，その枠組みのなかで必要な民間企業提供データの利用が可能となっている。さらに，国立情報学研究所情報学研究データリポジトリでも共同研究を前提として民間企業データの提供が行われている。以下，不動産分析に関連するデータをいくつか挙げる。

ゼンリン住宅地図は道路，建物ポリゴンなどを現地調査から作成しており，基盤地図情報よりも精度の高いデータを得られる。さらに，建物ポリゴン内に用途，階数などの情報が格納されており，それらはヘドニック価格推定において重要な変数となる。

アットホーム株式会社は不動産取引データを JoRAS 経由で提供しており，住宅種別及び分譲・賃貸別にデータが整備されている。

国立情報学研究所経由で提供される LIFULL HOME'S データセットは，賃料，面積，築年数などの住宅特性だけでなく，高解像度の間取り図画像データも提供している。従って，画像データを利用して住宅特性を補間することが可能であり，画像認識分野との共同分析など今後の発展が見込まれる。

住宅・土地統計調査などの公的統計は GIS データとして直接利用可能ではないが，市区町村別で集計されている場合市区町村コードを GIS のポリゴンと対応させることで分析可能である。例えば，市町村単位での空き家率を地理空間上に投影し，分布の傾向や空間集積などを考察することができる。

5.2 空間集計における基本操作

5.2.1 基本量の測定

不動産分析を行う際，例えば対象物件から最寄り駅までの直線距離や建築面積を知りたい場合がある。この場合，GIS を用いることで瞬時に計算可能である。また，ある用途地域にかかる敷地の面積や，対象とする地域だけ取り出して分析したい場合など，空間的な重ね合わせが役に立つ。加えて，GIS ではある施設までの距離が最近隣となる領域などを求めることも可能である。本節では空間集計の操作について基本的な事項を述べる。

ベクターデータにはポイントデータに緯度経度座標が格納されており，2 点のポイントデータ (x_i, y_i), (x_{i+1}, y_{i+1}) が存在する場合，その間のユークリッド距離（Euclidean distance）は

$$d_i = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \tag{5.1}$$

で定まる。複数の点が存在する場合，各二点間で距離測定を行えば良い。続いて，線分に囲まれた多角形の面積について求める。左記の二点に加えて (x_{i+2}, y_{i+2}), (x_{i+3}, y_{i+3}) も存在し，図 5.4 の線分の内側の面積 S を求めたいとする。この時， S_1 の台形の面積は

$$S_i = \frac{1}{2}(x_{i+1} - x_i)(y_{i+1} + y_i) \tag{5.2}$$

と表せる。よって，台形 S の面積は

$$S = \frac{1}{2}\{(x_{i+1} - x_i)(y_{i+1} + y_i) + (x_{i+2} - x_{i+1})(y_{i+2} + y_{i+1}) - (x_{i+2} - x_{i+3})(y_{i+2} + y_{i+3}) - (x_{i+3} - x_i)(y_{i+3} + y_i)\} \tag{5.3}$$

によって求積可能である。

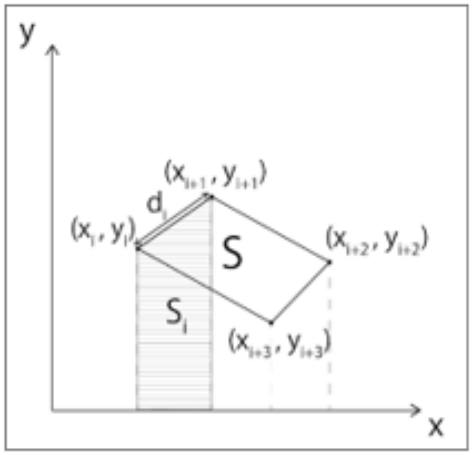


図 5.4　線分と面積の測定方法

このような方法により，対象物件から最寄り商店までの距離や建築面積などを計算できる。

5.2.2　ジオプロセシング

ジオプロセシング（geoprocessing）とは，GIS に関連するデータ処理を行い，新しいデータを出力する操作全般のことをいう。GIS では空間の領域生成や重ね合わせにより，分析者のニーズに応じた操作を行うことができる。領域生成には主にバッファー（buffer）生成によるものやボロノイ分割（Voronoi division）によるものなどがある。空間的重ね合わせには複数オブジェクトの足し引きで新たな領域を生成し，インターセクト（intersect），クリップ（clip），ユニオン（union）が主に利用される操作である。さらに，各ポリゴンのフィールドに基づく空間生成にもいくつか種類が存在し，マージ（merge），ディゾルブ（dissolve）が代表的な操作である。

任意のオブジェクトに対して，その近傍の領域を生成する操作をバッファーといい，ポイント，ライン，ポリゴンのそれぞれで生成可能である。例えば対象物件から 10 分圏内の領域を生成することができ，その領域内のコンビニエンスストア数の集計などに利用される。

ある点において近傍の領域を求める際に用いられるのが，ボロノイ図（Voronoi diagram）である。ある母点 $i \in (1, 2, \ldots, I)$ について，距離空間内の有限部分集合 $P = (p_1, p_2, \cdots, p_I)$ は，

$$V(p_i) = \{p \,|\, d(p, p_i) \leq d(p, p_j), j \neq i\} \tag{5.4}$$

と表現できる。ただし，$d(\cdot)$ は距離関数である。各ボロノイ領域 $V(p_i)$ の集合 $\{V(p_1), V(p_2), \ldots, V(p_I)\}$ をボロノイ図と呼び，各母点からの近接する領域を知ることができる。さらに，各母点同士を結ぶことで

描かれる三角図形をドローネ図（Delaunay diagram）と呼び，ボロノイ図と双対の関係にあることが知られている。このような図を作成することで，スーパーマーケットの商圏や各駅の駅勢圏などを明らかにすることが出来る（図 5.5)。

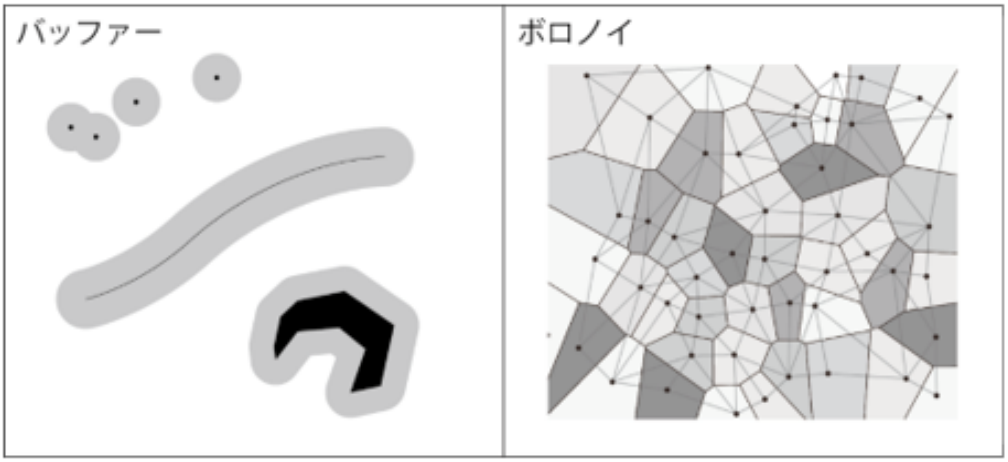


図 5.5　バッファーとボロノイによる空間的重ね合わせ

次に，空間的重ね合わせについて述べる。インターセクトはあるポリゴン A と B の積集合（ $A \cap B$ ）を求める操作である。生成されたポリゴンは A と B のどちらのフィールドも受け継がれる。一方，クリップはあるポリゴン A と B の積集合（ $A \cap B$ ）を求める操作であるが，インターセクトと異なり，ある空間領域 B に入るポリゴン A を取り出す操作である。そのため，ポリゴン B の属性は生成されたポリゴンに含まれない。ユニオンはあるポリゴン A と B の和集合（ $A \cup B$ ）を求める操作である。生成されたポリゴンは空間領域で分割され， A と B のどちらのフィールドも受け継がれる。例えば，複数市町村についてそれぞれ駅勢圏を分割した時にはユニオンを用いることで全ての情報が保たれて分割される。あるいは，A 市のみに対して駅勢圏を抜き出したい場合には駅勢圏を A 市境界のポリゴンでクリップすると良い。

続いて，フィールドに基づく空間生成について述べる。マージはユニオンと同様， A と B の和集合（ $A \cup B$ ）が出力されるが，同じフィールドを持つ空間データにより統合される。従って，生成されたポリゴンはマージに用いたフィールドに対してユニークに分割される。ディゾルブはフィールドに基づく空間データの集約であり，単体の空間データでも操作可能である。上記の演算イメージとテーブルの変化は図 5.6 のようにまとめられる。

5.3　空間データの相関と補間

5.3.1　空間的自己相関

本節では，不動産分析に関連する空間解析手法について，空間解析特有の概念である空間的自己相関と空間補間について説明する。下記のような分析により，不動産市場における空間的構造の理解を深めるとともに，出力結果を用いてデータベースの充実を図ることも可能である。

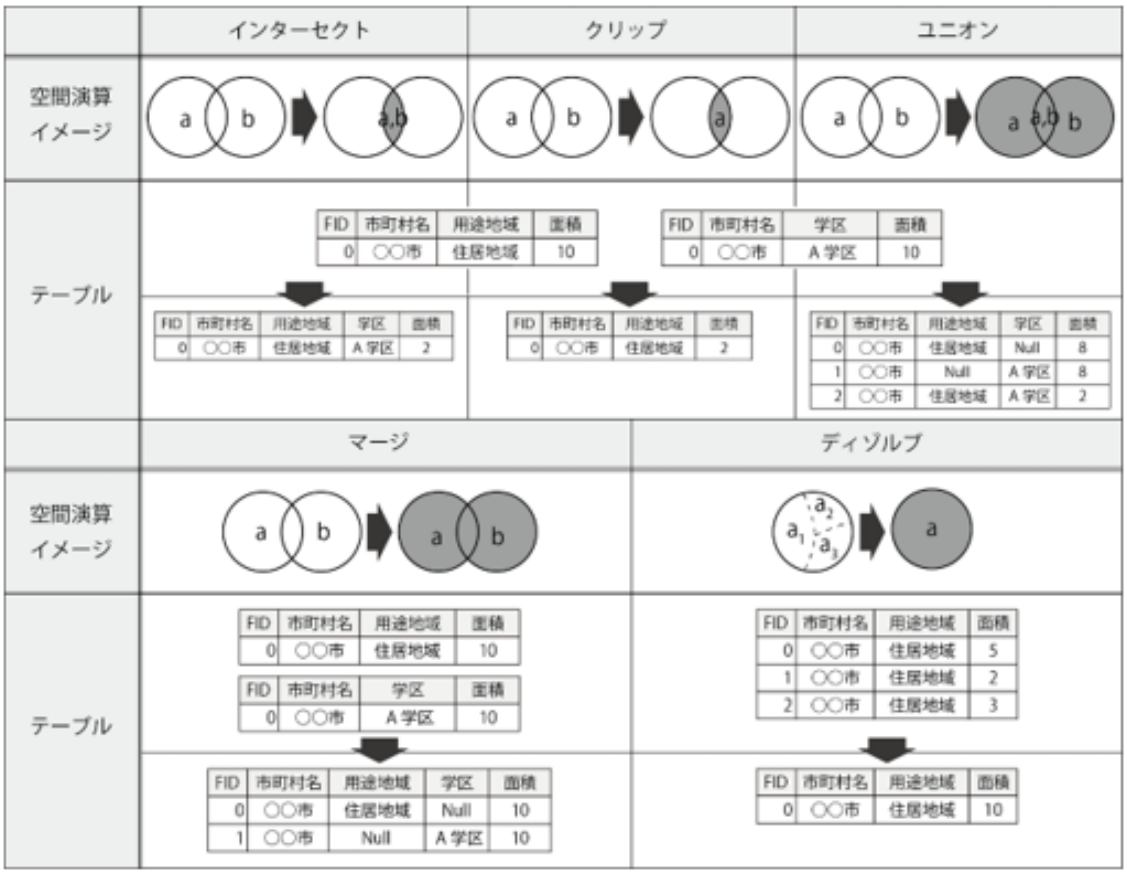


図 5.6　ジオプロセシングの空間演算と出力テーブル

ある空間領域について，興味のあるフィールドと相関があるかは分析を行ううえで重要である。例えば，市場滞留期間の長い不動産物件に集積があるかについて，客観的な判断が出来れば将来の投資判断につながると考えられる。以下，多くの既往論文で用いられている Moran's I を用いて概念について述べる。

一般に相関関係を求める際，二変数でよく用いられるのはピアソンの相関関数であり，以下の式で定義される。

$$r_{\mathrm{xy}} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{5.5}$$

ただし，$\bar{x}, \bar{y}$ はそれぞれ x と y の平均値を表す。

では，このような一般的な相関と空間相関との差異とは何であろうか。空間データは特定の座標上に観測値を持つため，空間上のどの位置を参照するかによって対応関係が異なる。特に，同一の観測値に関して空間的位置に起因する相関を空間的自己相関（spatial autocorrelation）とよぶ。

空間的自己相関について，代表的に用いられている指標に Moran's I が存在し，以下の式で表される。

$$I = \frac{n}{S_0}\frac{\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}(y_i - \bar{y})(y_j - \bar{y})}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{5.6}$$

ただし，$S_0 \equiv \sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}$，$w_{ij}$ は地域 i と j の距離に関する重み付け値であり，$d(i,j)$ を i と j

間の距離とすると，$d(i,j) > d(i,k)$ のとき，$w_{ij} < w_{ik}$ と定義する。これは，観測値同士が空間的に近いものほど似通っているというトブラーの地理学の第一法則（Tobler, 1970[6]）を反映したものといえる。なお，$w_{ii} = 0$ とする。w_{ij} の決め方としては，隣接している観測値を 1，そうでないものを 0 とおく場合や，$d^{-\alpha}$ や $\exp(-\alpha d)$ などの距離減衰関数を設定して数値化する場合がある。上記の統計量は，大域的な空間的自己相関を判定するため，グローバルモラン統計量（global Moran's I）とよばれている。グローバルモラン統計量が大きいときに正の自己相関であり，一方で小さいときには負の自己相関を示す（図 5.7）。

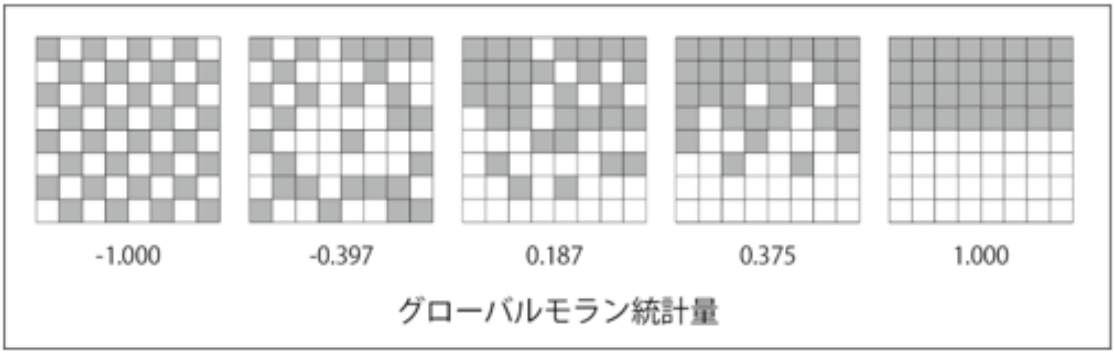


図 5.7　グローバルモラン統計量による空間的自己相関の図化

地域 i に着目してモラン統計量を算出する方法も存在する。さきほどの I について地域 i を固定すると，

$$I_i = \frac{n(y_i - \bar{y}) \sum_{j=1}^{n} w_{ij}(y_j - \bar{y})}{\sum_{j=1}^{n}(y_j - \bar{y})^2} \tag{5.7}$$

と書き下せる。これをローカルモラン統計量（local Moran's I）とよび，近隣地域との類似性について表す（Anselin, 1995[1]）。I_i が正であるとき，地域 i は近隣と類似している一方で，負であるときには類似していない。なお，$I = \sum_{i=1}^{n} I_i$ である。

このような統計量を用いることで，空間的自己相関に関する分析を行うことができる。具体的には，各地域のローカルモラン統計量を計算し，対象地域と周辺地域との関係性について類型化し，それぞれの関係性について相関の程度を High と Low で表す。これにより作成される図をモラン散布図（Moran scatterplot）といい，図 5.9 のように 4 分類して分析することが一般的である（Anselin, 1996[2]）。例えば，図 5.8 の第一象限にある地域は High-High を示しており，自地域と周辺地域が共に高いローカルモラン統計量をもつような状態である。これはホットスポットと呼ばれ，例えば産業集積の立地や度合いを分析する際に利用される。一方で第三象限にある Low-Low に当てはまる地域はクールスポットとよばれる。不動産市場において，高経年マンションの立地傾向の分析や，高層マンションの集積度合いを把握する際に役立つ。空間的自己相関の扱いについて，その影響を考慮した空間統計学（spatial statistics）という分野が発展しているが，紙面の都合上割愛する。詳細は瀬谷・堤（2014）[9] を参照されたい。

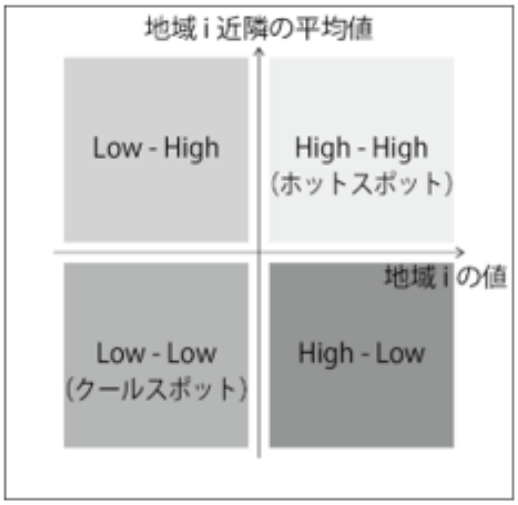


図 5.8　モラン散布図

5.3.2　空間補間

ここまで空間的な相関関係についてみてきたが，そもそも近隣の観測値が不明である場合，それを補間する方法がある。例えば，ある地点の地価を知りたい時に近隣の地価からある程度の推定が可能である。空間補間には大きく補間点近傍の観測値を用いる場合と，大域的な観測値を求める場合の 2 通り存在する。本節では，比較的容易に補間可能な前者について述べる。なお，空間補間に関する全般的な説明は Lam (1983)[3] において詳述されている。

空間補間のなかで最も簡便なものが最近隣法（nearest neighbor method）である。この方法では，任意のデータに対して，最近隣の観測点を当てはめる手法である。従って，観測点を母点としたボロノイ分割によって，任意の点に対する最近隣のデータ点を求めることができる。最近隣法は推定が比較的簡単である一方で，各ボロノイ領域は離散的な値をとるため，ボロノイ境界付近での値に誤差が生じる場合がある。すなわち，最近隣法による空間補間は観測点が十分密である場合には有効であるが，観測点間に大きなギャップが存在する場合や観測値の変動が大きい場合，良好な精度を保証できるとは限らない。

続いて，不整形三角網（TIN: Triangulated Irregular Network）から近傍を求める手法について述べる。これは，何らかの方法で不整形三角網を構築し，各三角形の頂点の観測値から内部にある任意のデータを補間するものである。この時，一般的に利用されるのが，ボロノイ図を描画する際に登場したドローネ三角網である。ドローネ三角網は各三角形の最小角を最大化するように作成するため，より正三角形に近い三角網を構築可能である。補間推定値を求める際には，xy 座標と観測値 z について，$A\,(x_1, y_1, z_1)$，$B\,(x_2, y_2, z_2)$，$C\,(x_3, y_3, z_3)$ の三つの頂点からなる平面を考える。平面上の任意の点を $P\,(x, y, z^*)$ とすると，ベクトル $\overrightarrow{AP}$ が $\overrightarrow{AB}$ と $\overrightarrow{AC}$ で書き表せる。すなわち，$\overrightarrow{AP}, \overrightarrow{AB}, \overrightarrow{AC}$ が一次従属

であることと同値である。従って，行列式を用いて，

$$\begin{vmatrix} x - x_1 & x_2 - x_1 & x_3 - x_1 \\ y - y_1 & y_2 - y_1 & y_3 - y_1 \\ z^* - z_1 & z_2 - z_1 & z_3 - z_1 \end{vmatrix} = 0 \tag{5.8}$$

を解くことで補間推定値 z^* を求めることが可能である。これは地形表現などでよく利用され，可能な限りコンパクトな観測値の組から補間値を得られる一方で，最近隣法と同様に表面が平滑化されない。

最後に，一定の仮定をおいて連続的な表面を作成する方法について述べる。代表的なものが逆距離加重法（IDW: Inverse Distance Weighted）である。これは，空間的自己相関と同様に地理学の第一法則を踏まえたものである。いま，観測点数 n ，補間点 z^* と観測点 z_i との距離を d_i ，距離に応じた重み付け関数を $w(\cdot)$ とする。このとき，補間推定値は

$$z^* = \frac{\sum_{i=1}^{n} z_i w(d_i)}{\sum_{i=1}^{n} w(d_i)} \tag{5.9}$$

で表される。なお，$w(\cdot)$ は通常距離で減衰し，空間的自己相関のように距離に関して $d^{-\alpha}$ などの仮定をおき，数値化する。この式の分母は距離による重みの和を基準化するための項であり，分子で距離的な重みに基づき観測値を足し合わせている。逆距離加重法の利点は直観的に自然な連続表面を作成することであり，補間推定値は連続値として求められる。一方で，真の空間分布が連続的に変化しない場合や，設定する距離関数が適切でない場合，真値とうまくフィットしない。この方法では距離関数やパラメータの精度を確認する必要があり，主に交差検証が用いられる。これは，観測値の一部を評価用にしてパラメータ推定に利用せず，推定補間値が評価用観測値とどの程度乖離しているか検証するものである。評価には，観測値と補間値の差分二乗値を平均化する二乗平均誤差（MSE: Mean Square Error）や，その平方根をとる二乗平均平方根誤差（RMSE: Root Mean Square Error）などが用いられる。以下，図 5.9 に空間補間のイメージをまとめた。

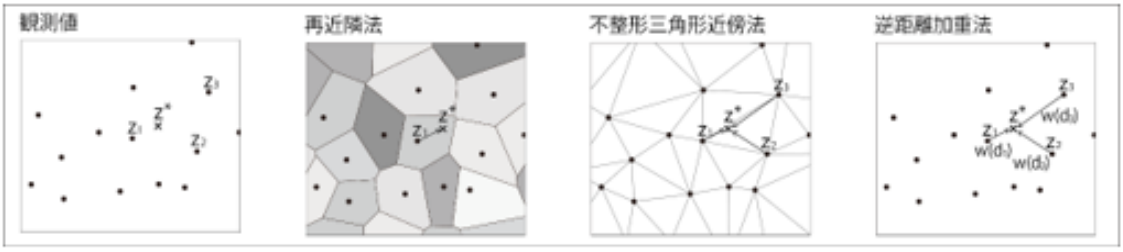


図 5.9 空間補間のイメージ

5.4 おわりに

本章は不動産分析に資する GIS 分析手法について早足で概観した。不動産市場分析に有用な GIS 的処理は，データベース構築のための空間演算から高度な解析まで多岐にわたる。本章では GIS の基本的な処理及び分析について多くの時間を割いたが，それぞれの概念は相互に関連していることがわかる。例え

ば空間補間を行う際，再近隣法ではボロノイ領域の概念を利用し，逆距離加重法の重み付け関数は空間的自己相関のものと類似した考え方である。従って，一度空間解析の基礎を身に付ければ，様々な応用に派生可能といえる。

これまでに紹介した分析は，実際には GIS に関連するソフトウェアにより計算されるため，その習熟も重要になる。一般的に利用されるのは ESRI 社の ArcGIS やフリーソフトウェアである QGIS である。さらに，統計プログラミング言語である R でもいくらかの処理を行うことができ，例えば spdep パッケージを利用すればドローネ三角網の図化や空間的自己相関の確認などが可能である。

近年になり，不動産分析における地理的特性や対象物件周辺のアメニティが重要視されており（例えば Shimizu, 2014[5]），今後さらに GIS を用いた情報の取得が重要になってくると考えられる。このような潮流に乗るため，浅見ら（2015）[7] や貞広・山田・石井（2018）[8] なども参考にすると理解が深まる。なお，頁の都合上割愛したが，空間回帰分析や空間相互作用モデルなど，やや発展的な解析も不動産分析を行ううえで重要である。深く学びたい場合は Longley et al. (2005)[4] などを参照すると良い。

参考文献

[1] Anselin, L. (1995). Local indicators of spatial association-LISA. *Geographical analysis*, 27(2), 93-115.

[2] Anselin, L. (1996). The Moran scatterplot as an ESDA tool to assess lo-cal instability in spatial association. *Spatial Analytical Perspectives on GIS*, 111-125.

[3] Lam, N. S. N. (1983). Spatial interpolation methods: a review. *The American Cartographer*, 10(2), 129-150.

[4] Longley, P. A., Goodchild, M. F., Maguire, D. J., & Rhind, D. W. (2005).*Geographic information systems and science*. John Wiley & Sons.

[5] Shimizu, C. (2014). Estimation of Hedonic single-family house price function considering neighborhood effect variables. *Sustainability*, 6(5), 2946-2960.

[6] Tobler, W. R. (1970). A computer movie simulating urban growth in the Detroit region. *Economic geography*, 46(sup1), 234-240.

[7] 浅見泰司・矢野桂司・貞広幸雄・湯田ミノリ (2015) 地理情報科学—GIS スタンダード. 古今書院.

[8] 貞広幸雄・山田育穂・石井儀光 (2018) 空間解析入門 —都市を測る・都市がわかる. 朝倉書店.

[9] 瀬谷創・堤盛人 (2014) 空間統計学: 自然科学から人文・社会科学まで. 朝倉書店.

第 6 章

XX

第 7 章

XX

第 8 章

XX

第 9 章

XX

第 10 章

GIS を用いたエリア指標の開発

10.1　エリア指標と不動産テック

消費者の住宅選択において，土地及び建物だけでなく，その住宅を取り巻く周辺の環境水準は，極めて重要になってきている。人々は，広義のエンターテイメントに基づく「アメニティ」によってもたらされる文化的・自然的消費の機会を重視するようになることで，住宅選び=エリア選びといった方向へと，多くの国において，エリアの価値が重視されるようになってきている。米国では，2007 年から「Walk Score」と呼ばれる指標が開発され，そして，不動産情報サイトによって公開されるようになってきたことから，またそのようなスコアが不動産価格の予測モデルの中の特徴量として利用されていることから，不動産テック分野においても高く注目される研究領域になってきている。

このようにエリア指標が注目されるようになった理由には，かつての労働集約型の企業が大部分を占めていた経済構造から，情報と知識集約型産業が主となる形へとシフトし，人々の生活において余暇を楽しむ機会が増えたことが挙げられる (Fogel (2000)[2]， Glaeser, Kolko and Saiz (2004)[3])。このような傾向は，「働き方改革」が推し進められる，今後の日本において一層強くなっていくものと考える。

都市の集積のメカニズムの変化は，都市の役割を「生産のための場」から「消費のための場」へとシフトさせた (Glaeser, Kolko, and Saiz (2004)[3])。いわゆる，経済学でいう生産関数から消費関数へ，企業から家計へと主役が変化し，都市の民主化が進み，「Consumer City Theory」が発展してきたのである。

そのような中で，GIS 情報を用いて，暮らしやすさの観点から不動産の立地環境を表す指標「日本版 WalkScore（仮称）」を開発した。具体的には，株式会社ゼンリン保有の各種施設ポイントデータ，ネットワークデータからデータベースを構築し，個々の場所とその場所から徒歩でアクセス可能な周辺アメニティとを紐づけ，周辺アメニティの充実度からその場所の暮らしやすさを評価したスコアを算出した。

住宅向け，オフィス向けといった「用途別スコア」に加え，家族世帯向け，単身世帯向け，高齢者向けなどの「タイプ別スコア」など，多様なニーズに応じたスコア提供へと発展させることが可能である。また，スコアは現状で 50m メッシュごとに算出が可能であり，街区レベルに匹敵する詳細な立地環境の把握が可能である。

これにより，同一駅周辺の複数の物件についても周辺環境を一目で比較できるようになる。さらには，ヒートマップ上から立地環境の優れたエリアを見定めたり，レーダーチャートの形状から類似した立地環境の物件を提案したり，といった幅広いサービス展開を行うことができる。

本章では，エリア指標の開発におけるデータ資源と技術について解説したい。

10.2　不動産の価値評価

10.2.1　不動産価値評価の現状

不動産価値の評価手法は種々あるが，近年，Real Quality Rating（RQR）という評価指標が注目されている。現状，不動産の価格（market price）については多くの情報にアクセスが可能であるが，それらの不動産の質（market quality）についての情報を入手することは容易ではない。RQR は，不動産の立地環境情報や建物自体の情報，建物内についての情報から総合的に不動産を評価することでその品質を明らかにし，投資を行う上での判断基準となる指標として 2017 年に開発されたものである。

近年開発された不動産価値評価の指標としては，RQR 以外にも Well Building Standard (2012-)[9] や Fitwel (2017-）[1] などがあるが，いずれも建物自体の評価にとどまらず，建物を利用する人の快適性や健康，ウェルビーイングといった観点から評価項目が選定されているという特徴がある。我が国においても，日本政策投資銀行が創設した DBJ Green Building 認証 (2014-)[12] ，CASBEE-ウェルネスオフィス（2019-）[11] で同様の視点が取り入れられており，利用者目線を考慮した不動産価値評価の考え方が世界的な潮流となりつつある。

表 10.1　近年開発された不動産価値評価の指標（例）

10.2.2　立地評価

不動産価値評価指標について，特に立地の評価の考え方にも近年変化が見られる。米国では，2007 年から「Walk Score」という指標が開発されサービス提供されている。Walk Score は徒歩での生活のしやすさを表す指標であり，任意の住所に対してその周辺に「Dining&Drinking（飲食店）」，「Shopping（買い物）」，「Parks（公園）」，「Schools（学校）」などの都市アメニティがどれだけ充実しているかを算出し 100 点満点のスコアを提供している。徒歩でアクセスできるアメニティが多いほど生活しやすい場所という評価がなされ，このスコアを見るだけでその物件の周辺環境の良し悪しを把握することができる。WalkScore は，米国の大手不動産ポータルサイトに掲載されており，物件探しをしている人が気になる物件の詳細ページを開いた際に，家賃や間取りなどの情報に併記される形でスコアが表示されており，物件検索条件の一要素として定着しつつある。また，米国を中心に普及している不動産価値評価指標「Fitwel」においては，このスコアが立地評価における採点基準としても取り入れられている。

一方，我が国においてはこのような一般向けの不動産立地環境評価の指標は確立されていない。しかし，私たちが住む家を探す際には，コンビニやスーパーが近くに充実しているかどうかや，小さな子供がいる世帯であれば子供が遊べるような公園が近くにあるかどうかなどは当然のように関心を持っている。また，オフィスの立地についても，ランチで行くレストラン・弁当屋や夜の居酒屋などが充実しているか

図 10.1　米国 WalkScore ウェブサイトのトップページ（https://www.walkscore.com）

どうかはオフィスワーカーにとって有益な情報となりうる。オフィスの中でも，建物内に飲食店が併設されていることも多い大規模オフィスとは異なり，中小規模のオフィスの場合には，建物の周辺にどれだけ利便施設が充実しているかという情報は相対的にニーズが大きいと考えられる。投資家目線では，同じ中小規模のオフィスであっても，周辺施設がより充実しているものほど市場価値が高いため，そちらにより多くの投資を行うという判断を行うことができる。国全体で不動産ストックの老朽化が大きな課題となっている我が国において，優れた立地ポテンシャルを持ち市場価値の高いストックを峻別し再投資を行う上で，このような客観的な判断基準が整備されることには大きな意義がある。

10.3 「日本版 WalkScore（仮称）」の開発

10.3.1 データ資源とデータベース構築

従来，不動産価値評価では，主に家賃，最寄り駅までの距離，建物・設備スペックについての情報が用いられてきた一方で，不動産周辺環境については分かりやすい客観的な評価指標が整備されてこなかった。

本取組みは公共及び民間が保有する GIS 情報を用いて，上述したような我が国の社会課題解決に向けて，不動産立地環境に関する新たな評価指標「日本版 WalkScore（仮称）」の開発を行うものである。

「日本版 WalkScore（仮称）」は暮らしやすさの観点から，不動産の立地環境（周辺の都市アメニティ充実度）を表す指標である。全国の市街化区域を対象として，不動産とそこから徒歩でアクセス可能なアメニティ群（スーパー，コンビニ，公園，飲食店，カフェなど）のデータを紐づけ，アメニティ分類ごとの周辺立地数をもとにその充実度を 100 点満点でスコア化するものである。

図 10.2 従来の不動産価値評価における主な指標と日本版 WalkScore（仮称）が提供する指標

(データ資源)

主に使用したデータは，様々な都市アメニティの位置情報を有したポイントデータと，徒歩での所要時間を算出するための経路情報であるネットワークデータとに分けられる。いずれも株式会社ゼンリンの

データを使用しており，前者はテレポイント Pack!データ，建物ポイントデータ及び POI データ，後者は主に歩行者ネットワークデータを用いている。

テレポイント Pack!データではアメニティ業種が 2000 超まで細分化されており，その詳細な業種ごとに個々のアメニティの位置情報が入手できる。コンビニ等の業種に関しては個別の企業ブランド名（例：セブンイレブン，ローソン，ファミリーマートなど）まで補足できるなど，非常に詳細なアメニティ分類となっている。

また，テレポイント Pack!データ，建物ポイントデータ，POI データそれぞれに，データの収集・作成の過程で，アメニティ業種ごとの捕捉率に差異が生じていることがある。しかし，業種ごとに各データを突き合わせて比較することで，より実態に即した適切なデータを採用している。

スコアを集計する際には，各種分析を通じて，それら膨大なアメニティ業種の中から，不動産価値評価においてより重要度が高いと思われるアメニティ項目を選定して使用している。当然ながら，様々なニーズに合わせ，スコア集計時に採用するアメニティ業種をカスタマイズしていくことも可能である。実際に，目的別スコア，タイプ別スコアはそれぞれの用途・タイプごとに選好されるアメニティ業種を考慮し，スコアごとに集計対象のアメニティ業種を変化させている。

(データベース構築)

日本全国の市街化区域を 50m メッシュで分割し，その 1 つ 1 つのメッシュに対してスコアを算出している。スコア算出に際し，まず各 50m メッシュを起点として徒歩で到達可能な範囲として「徒歩圏」を設定し，その「徒歩圏」内に立地するアメニティを特定する。その上で，その 50m メッシュと「徒歩圏」内アメニティとを紐づける。この操作を全ての 50m メッシュに対して行うことで，日本全国の任意の場所とそこから徒歩でアクセス可能なアメニティに関するデータベースを構築することができる。

50m という距離は、徒歩 1 分で 80m の距離を進むとした場合，徒歩で約 40 秒程度の距離である。50m メッシュという地理的に非常に細かい単位でスコア集計をしていることにより，より正確な周辺環境の評価が可能となる。例えば，同じ駅であっても，駅の西側は昔ながらの商店街が続く商業エリア，駅の東側は大学キャンパスが広がる文教エリア，などというように，駅の出口ごとに街の特徴が大きく異なる場合は少なくない。このような場合，立地環境の評価を駅という単位で行ってしまうと，当然ながら実態を上手く反映した評価ができないということは想像に難くない。駅の東西，南北でエリアの特徴が大きく異なるような場合でも，50m メッシュ単位で評価を行うことでスコアにその差異を十分に反映させることができる。

これにより，同じ東京駅周辺であっても，皇居側の丸の内エリアと，日本橋側の八重洲エリアの差異を正しく捕捉することができ，さらに言えば，同じ丸の内エリアの中でもより周辺アメニティが充実している場所とそうでない場所とを把握することができる。丸の内エリアという業務中心地にオフィスを構えたいと考える企業は多く存在するが，丸の内エリアの中でも周辺環境が異なり，従業員の満足度を左右し得るという点から見れば，50m メッシュでのスコア提供はオフィス立地選択における新たな価値提供にもつながると考えられる。

また，不動産から徒歩でアクセス可能なアメニティを集計する際に，単純なポイント間の直線距離ではなく，歩行者ネットワークデータを使用した徒歩経路距離を用いている点も重要である。例えば，横断で

きる地点が限られている幹線道路や大規模な施設が存在するエリアでは，直線距離と比べて徒歩経路距離がより長くなっていることが考えられ，直線距離を用いて周辺のアメニティを定義した場合には実態とのかい離が大きくなってしまう。しかし，今回は徒歩経路を用いることで，そのような実態との乖離をなくしている。また，大規模な公園内の歩道なども反映されており，実状を正確に反映した周辺アメニティの捕捉が可能になっている。

なお，各 50m メッシュを起点とした「徒歩圏」は移動距離に応じて複数設定しており，のちのスコア算出の際に距離減衰を考慮した重み付けを行っている。各アメニティ数は、「徒歩圏」ごとの距離に応じた重み付けを行う場合には、下記のように集計できる。

$$AM_i = \sum_j a_j \cdot AM_{ij} \tag{10.1}$$

AM_i : アメニティ数
- SM : スーパーマーケット
- CS : コンビニエンスストア
- DS : 薬局
- PF : 公共施設
- $\vdots$

a_j : 各「徒歩圏」の距離に応じた低減係数 $(j = 1, \ldots, J)$

図 10.3 経路データを用いた「徒歩圏」の算出と「徒歩圏」内のアメニティの特定のイメージ

10.3.2 日本版 WalkScore の計算

以上のように構築されたデータベースを用いて，エリア別指標を計算していく。

(キーアメニティの抽出)

アメニティの業種数については，テレポイントデータの特性上，最多で 2000 超を区別することが可能である。しかし，この中には，不動産の立地環境の評価において必ずしも大きな影響を与えない業種も含まれているため，それらの業種を省き，より重要度の高い業種（キーアメニティ）のみを抽出してスコア算出に用いている。

キーアメニティの抽出には，ヘドニック理論と呼ばれる手法を活用する。

Rosen(1974)[6] によって提案されたヘドニックモデルは、差別化された生産物の市場均衡理論を発展させ、住宅のような財をどのように分析することができるのかを、経済理論と計量経済モデルの両面から示した。具体的には、商品供給者のオファー関数 (offer function)、商品需要者の付け値関数 (bid function) およびヘドニック価格関数の構造との間の関係を厳密に検討し、市場価格を消費者および生産者の行動から特徴づけている。そうすると，ヘドニック理論を援用することで，住宅の選択者がどのような要因に基づき，そして，どの程度のウェイトをもってその要因を重視して住宅選択をしているのかを定量的に把握することができる。

そこで，下記の 3 つの手続きによって，アメニティ業種の抽出作業を行っている。第 1 段階として，地価分析を行っている。商業・業務地と住宅地とを区別した上で，各アメニティの立地数と地価との関係性を分析し，地価に対して有意な影響を与えているアメニティを採用している。第 2 段階としてデモグラフィック分析を行っている。各アメニティ立地数と人口動態との関係性を分析し，様々な社会的属性を持つ人々が選好していると考えられるアメニティを採用している。第 3 段階として，既往の類似指標調査等を行っている。上記のような分析に加え，既存の類似指標や不動産ポータルサイト等で用いられているアメニティを参照し，重要度が高いと考えられるアメニティを採用している。

地価分析に際しては，ヘドニックアプローチを用いている。被説明変数として地価，説明変数として，用途地域，実行容積率等の都市計画条件およびスーパー，コンビニ，公園，飲食店等のアメニティ充実度を用いている。

一つの例を挙げれば，一般的なヘドニック理論に基づく地価関数は，下記のように推計されることが多い。

$$\log P = a_0 + \sum_h a_{1h} X_h + \sum_j a_{2j} Z_j + \sum_k a_{3k} \cdot LD_k + \sum_l a_{4l} \cdot RD_l + \sum_m a_{5m} \cdot TD_m + \varepsilon \quad (10.2)$$

DP/GA : 戸建て住宅価格 (円/m^2)
X_h : Main variables
GA : 土地面積
FS : 建物面積
RW : 前面道路幅員
Age : 建築後年数
TS : 最寄り駅までの時間
TT : 都心までの時間
Z_j : Other variables
ZD : 土地利用規制:容積率・建ぺい率・用途規制
BC : その他の要因:南向きダミー等
LD_k : Location(Ward) Dummy ($k = 0, \ldots, K$)
RD_l : Railway Dummy ($l = 0, \ldots, L$)
TD_m : Time Dummy ($m = 0, \ldots, M$)

そのような関数に，アメニティや地域環境を追加していく。

$$\begin{aligned}\log P = a_0 &+ \sum_h a_{1h} X_h + \sum_j a_{2j} Z_j + \sum_k a_{3k} \cdot LD_k + \sum_l a_{4l} \cdot RD_l + \sum_m a_{5m} \cdot TD_m \\ &+ a_6 Dm_{(l \le LA < m)} + a_7 Dm_{(m \le LA)} + a_8(LA)(Dm_{(l \le LA < m)}) + a_9(LA)(Dm_{(m \le LA)}) \\ &+ \sum_i a_{10i} \log V_i + \sum_{h,i} a_{11h,i} X_h \cdot V_i + a_{12} u + a_{13} v + \varepsilon \end{aligned} \tag{10.3}$$

V_i : Neighborhood Effects
AM : アメニティ
CS : 世帯特性:国勢調査
u, v : longitude, latitude

(スコアの算出と可視化)

データベースからキーアメニティの立地数を集計し，最終的に 50m メッシュごとに 100 点満点のスコアを算出している。なお，スコア集計に際し，同一アメニティについての効用逓減，歩行経路距離による効用逓減を考慮している。

現時点で，スコアは「用途別スコア」と「タイプ別スコア」の 2 種類を開発している。「用途別スコア」は，住宅向け（for Residence），オフィス向け（for Office）のスコアを想定し，用途に応じたスコア算出手法を用い，住宅用途，オフィス用途それぞれに対応している。住宅向けについては，さらに「タイプ別スコア」として，家族世帯向け（for Family），単身世帯向け（for Family），高齢者向け（for Elderly）を想定し，住む人に応じたスコア算出方法を用い，各タイプに対応している。

本スコアは，不動産を探す希望エリアの 1 次スクリーニングツールとして活用することが想定される。スコアをヒートマップとして表示することで，例えば東京 23 区全体を俯瞰して，どのエリアが特にスコ

アが高いかを直感的に把握することができる。鉄道駅周辺や主要路線沿線一体が赤く表示され，特にアメニティ充実度が高いことが見て取れる。逆に，大きな河川沿いや大規模な公園の付近では，川や公園によって「徒歩圏」内のアメニティ数が比較的少なくなってしまい，アメニティ充実度の観点からは比較的低いスコアとなっている。また，駅から離れたエリアでも，大規模な商業施設や商店街が近いエリアでは，アメニティが充実し，スコアが高い傾向になっている。

通常，不動産を探す場合には駅からの近さを重視する傾向があるが，駅から多少離れていても周辺環境が優れた不動産をきちんと評価，可視化することで，不動産の価値をより正確に伝えることが可能になる。

また，レーダーチャート表示で，アメニティ分野別のスコアを表現することで，その不動産の立地環境の特徴を一目で把握できるようになる。また，レーダーチャートの形状により，まちの類型化が可能になり，第一希望のエリアは家賃面で断念せざるを得ないという顧客に対し，それと類似した形状のレーダーチャートを持つエリアを代替候補として提案するなど，顧客ニーズに合ったエリアの提案を客観的なデータをもとに効果的に行えるようになる。

これにより，不動産の需要者と供給者の間の最適なマッチングを促進し，わが国の不動産市場の活性化を促進することも可能となる。

図 10.4　日本版 WalkScore（仮称）のスコア可視化イメージ（ヒートマップ表示の例）

図 10.5　日本版 WalkScore（仮称）のスコア可視化イメージ（レーダーチャート表示の例）

図 10.6　日本版 WalkScore（仮称）のスコアを用いた街比較の例

10.4　日本版 WalkScore 研究の発展可能性

「日本版 WalkScore（仮称）」に代表される地域指標の開発は，今後，大きな研究分野として成長していくことが予想される。

今後も急速に進む高齢化もまた住宅を取り巻くアメニティとの関係に大きな変化をもたらす。働く時間、通勤する時間の縮小と働き方が大きく変化すれば、または労働から解放された人々が増加していく中では、住宅およびそれを取り巻く地域での過ごす時間の質と密度が大きく変化していく。休息をとるためだけの家であれば、職場と近接した場所に住まうことで通勤時間を節約し、住宅から受けるサービスの水準を上昇させることができた。そのため、従来の住宅選択では、「最寄り駅や都心までの距離」や「大きさ」、各種性能といった「住宅」の物理的機能だけにしか関心がなく、そのような機能だけによって価格差が生まれてきた。

しかし、住宅を中心とした歩行可能空間での消費活動が活発化することで、つまり家で過ごす時間が長くなることで、多様な消費ができる地域に人々が集まる傾向は強くなっていくであろう。

しかし、ここで重要になってくるのが、「Scene:シーン」という概念である。Shimizu et al (2014)[7] では、アメニティの種類をシーンと呼んだ。これは、Clark 教授が主導した国際比較プロジェクトの中で出てきた概念である。そして、そのシーンは、それぞれの街を見る個人によって異なる評価が存在することを意味している。例えば、バーやカラオケが集積している街のシーンを見た時に、ある主体はワクワクとするようなエキサイティングをする場合もあれば、違う主体では嫌悪感を覚える場合もある。緑や公園をみて、心が穏やかになる人もいれば、寂しくなる人もいる。また、そのシーンも一日の中での時間や一年の中での季節によっても変化していく。

Walk Score などの定量的な指標は、そのような意味では平均的な街の顔を写像しているだけであり、街の特性を十分に踏まえた万能な指標ではない。歩行可能な空間が魅力的な街になりうる場合もあれば、それが人によっては違う街の顔として映る場合もある。

街づくりや都市計画を進めるにあたり、誰の、どの視点から街を眺め、評価していけばいいのであろう

か。都市が縮退し、高齢化が進展する中で、多数決原理では最適な解は見つけることは出来ず、多様な主体からの見え方を重視していかなければならない。将来を見据えた時には、子供や若者などの目から見えるシーンを大切にしなければならないはずである。色のついていないできる限り純粋な目をもって、街を眺めていくことが重要になってきていると考える。

それを実現していくためには，道路傾斜等も考慮した徒歩経路の設定や，時間帯を考慮したスコア集計（飲食店の営業時間等を考慮したナイトライフスコアなど）などを組み込んでいくことも考えられる。このように，日本版 WalkScore（仮称）には様々な発展可能性がある。現在，日本版 WalkScore（仮称）と同様，前述の RQR についても日本に適した形で開発を進めている。これらの指標開発及び実用化を通じて，不動産立地環境情報の見える化を実現し，不動産に関する情報の非対称性を解消し，不動産とユーザーとのマッチングを推進することが，我が国における不動産市場のさらなる活性化につながるものと考える。

今まで可視化できていない不動産にかかわる情報を，テクノロジーの進化によって実現できるようになってきているのである。

参考文献

[1] Fitwel ウェブサイト (最終閲覧日：2019 年 12 月 5 日)
https://www.fitwel.org

[2] Fogel,R.W. (2000), “The Fourth Great Awakening the Future of Egalitarianism”, University of Chicago Press.

[3] Glaeser, E., Kolko, J.K., Saiz, A. (2004), “Consumers and Cities, The City as an Entertainment Machine”, Research in Urban Policy 9, Elsevier, 177-184.

[4] Navarro, C. J., Mateos, C. and Rodriguez, M.J. (2012) Cultural scenes, the creative class and development in Spanish municipalities, European Urban and Regional Studies, 21: 301-317

[5] Real Quality Rating ウェブサイト (最終閲覧日：2019 年 12 月 5 日)
http://rqr-global.com/

[6] Rosen, S.（1974）, “Hedonic Prices and Implicit Markets, Product Differentiation in Pure Competition,” *Journal of Political Economy*, Vol.82, pp34-55.

[7] Shimizu, C., S. Yasumoto, Y. Asami and T. N. Clark (2014), “Do Urban Amenities drive Housing Rent? ” , CSIS Discussion Paper: (The University of Tokyo), No.131.

[8] Silver, D., T. N. Clark and C. J. Navarro. (2010) Scenes: Social Context in an Age of Contingency, Social Forces 88 (5): 2293-2324

[9] The International WELL Building Institute ウェブサイト (最終閲覧日：2019 年 12 月 5 日)
https://www.wellcertified.com

[10] Walk Score ウェブサイト (最終閲覧日：2019 年 12 月 5 日)
https://www.walkscore.com

[11] 建築環境・省エネルギー機構ウェブサイト, “CASBEE ウェルネスオフィス評価認証” (最終閲覧日：2019 年 12 月 5 日)
http://www.ibec.or.jp/CASBEE/certification/WO_certification.html

[12] 日本政策投資銀行ウェブサイト, ”DBJ Green Building 認証” (最終閲覧日：2019 年 12 月 5 日)
https://www.dbj.jp/service/finance/g_building/