

販売量予測モデル構築のための作業手順書

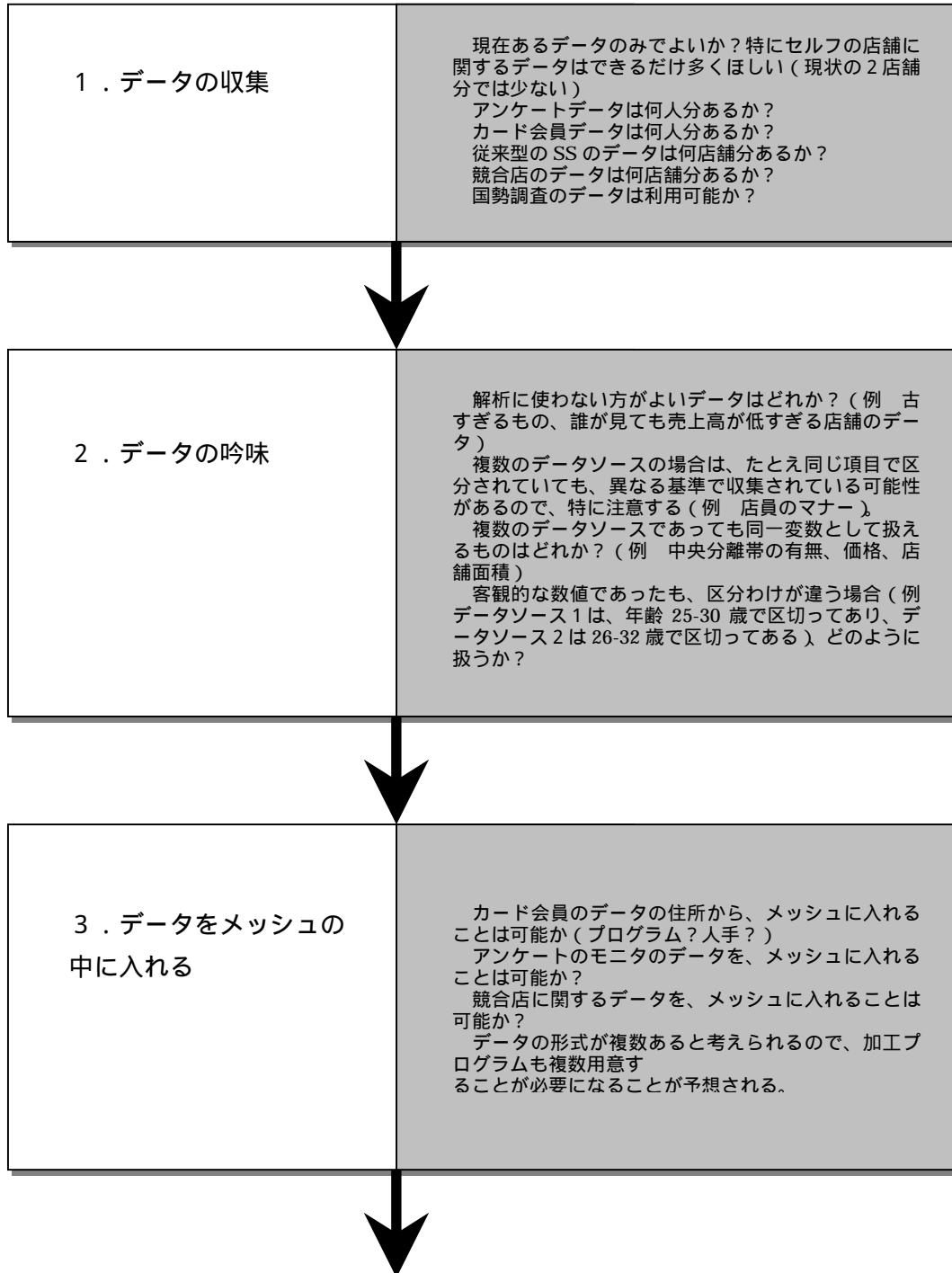
目次

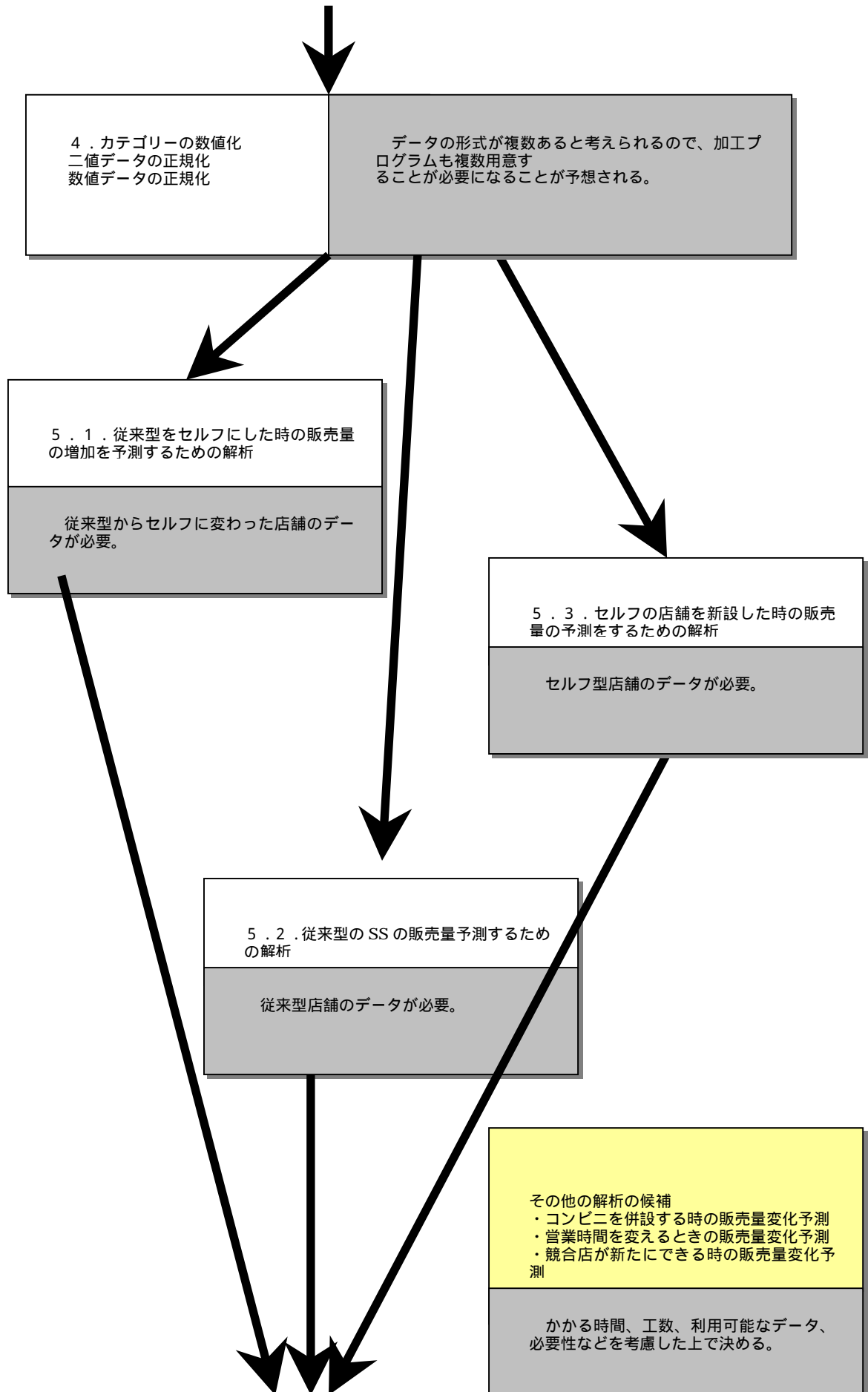
作業手順のフローチャート	4
構成的に商圈を設定していく方法の問題点	8
商圈を設定しない方法（構成的でないモデルの構築法、あるいはデータを学習させること によってモデルを構築する方法）	8
メッシュごとに分けられているデータ	9
潜在的な需要に関するデータ	9
メッシュ内の競合 SS	9
メッシュ内の他の商業施設	9
対象となる SS そのものの属性	10
顧客に関するデータ	10
カード顧客関連	10
アンケートに答えてもらったモニターに関するデータ	10
留意点	11
データの区別に関して	11
解析時に新たに付け加えるべき変数	11
データのプレシーズニング（前処理）	11
データの形式を揃える	11
数値データの正規化	11
二値データの正規化	11
カテゴリカルデータの正規化	11
データの数を増やす	12
データの直積によって大量のサンプルデータを作る	12
ブートストラップによって大量のサンプルデータを作る	13
直積とブートストラップを組み合わせる大量のサンプルデータを作る	14
解析法およびデータ	14
三つの解析法	14
データの形式	14
3つの解析法の内容と特徴	15
重回帰モデル	15
ニューラルネットワーク	16
樹状モデル（または樹形モデル）	17
なぜ樹状モデルやニューラルネットワークは重回帰よりも優れているか？	19
学習データとテストデータ	20

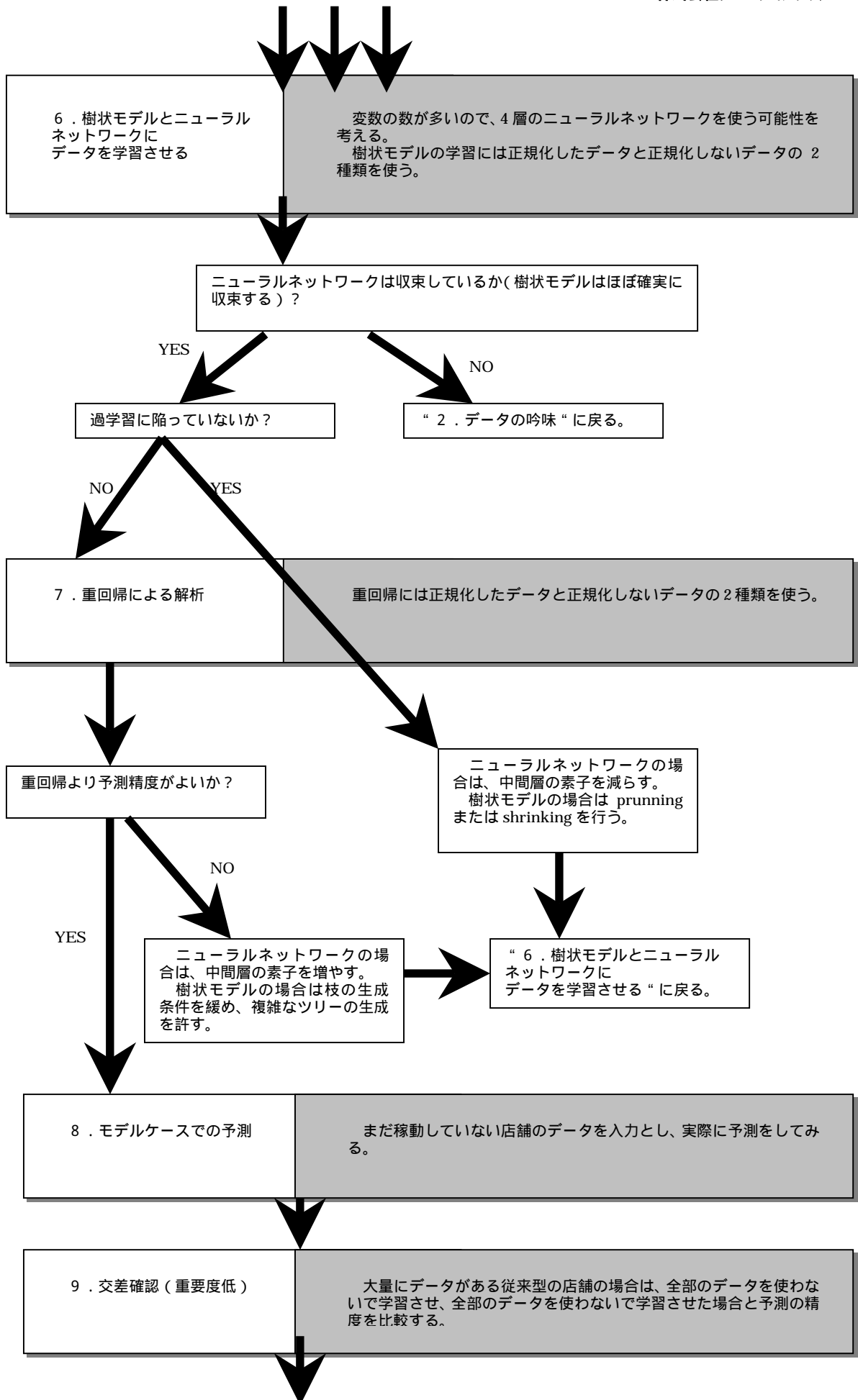
実際の解析にあたって.....	20
欠損値 (missing value) の扱い.....	20
外れ値 (outlier) の扱い.....	20
学習の回数.....	20
交差確認法 (cross-validation)	20
過学習 (over-learning) に対する対策.....	21
変数の間引き.....	21
データから決定される商圏の設定.....	21
変数間の関係の調査.....	21
方程式の構築.....	21
計算可能性.....	21

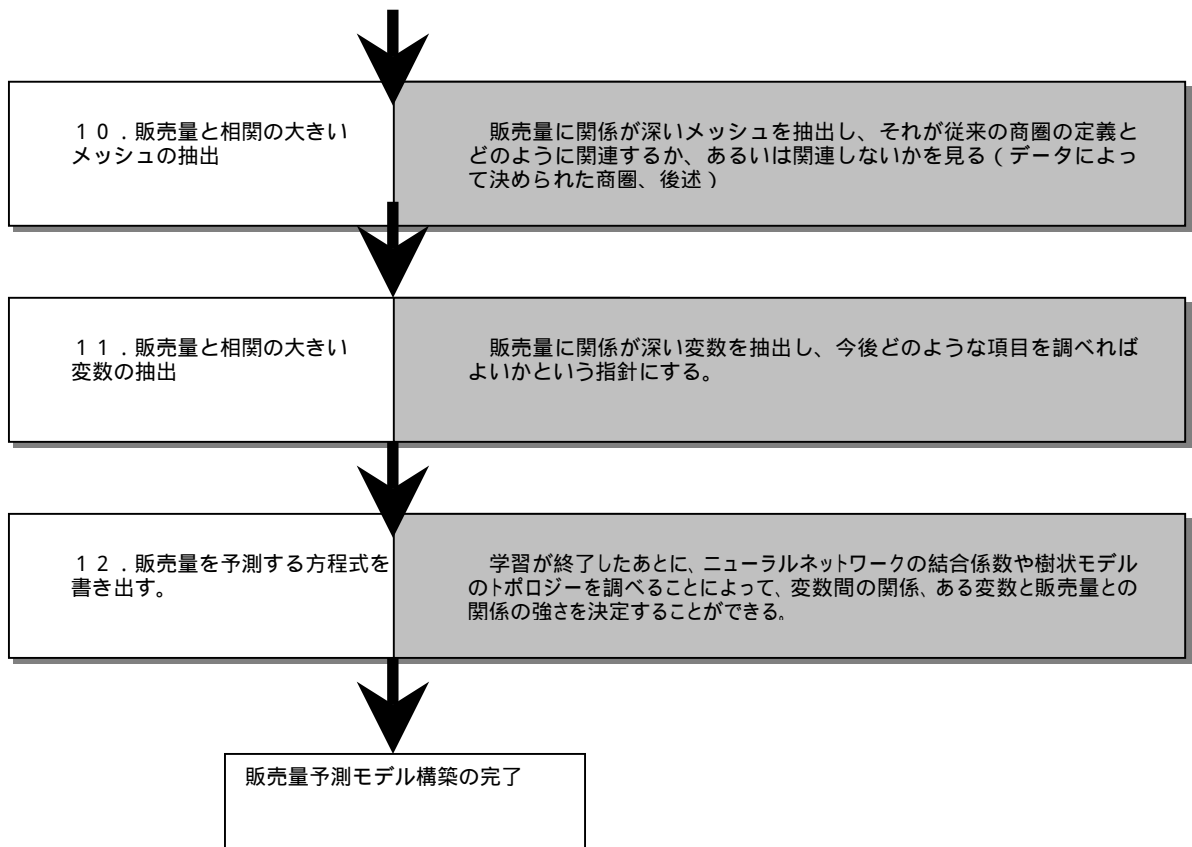
作業手順のフローチャート

解析の手順をフローチャートを使って説明する。









構成的に商圈を設定していく方法の問題点

世帯数、交通量、昼間人口、商業統計、競合 SS の影響度によって、商圈を構成的に設定していく方法は、

- ・比較的単純にモデルを構築できる。
- ・モデルの構成が直感的で、理解しやすい。

といった長所がある一方、

- ・カード顧客やアンケートに答えてもらったモニターなどの情報を設定した商圈の情報とどのように結びつけるか明らかではない。
- ・補足率や競合 SS などの影響度を正確に評価しにくく、人間が評価すると、恣意性が入りやすい。

といった短所がある。従来のライリーの法則やハフモデルも、市場ポテンシャル、買い物出向比率、距離抵抗係数といった客観的に決めるのが困難なパラメータを含んでおり、そこがこれらのモデルの限界である。ただし有効な考え方も多く含まれており、前向きに考慮する。

商圈を設定しない方法（構成的でないモデルの構築法、あるいはデータを学習させることによってモデルを構築する方法）

今回ご提案するのは、商圈を初めからは設定しない方法である。この方法の主眼は、解析者による恣意的な判断を可能な限り除いた形で解析を行うということである。

長所

- ・恣意性が入りにくい。

短所

- ・必ずしも直感的ではないので、理解が難しい。
- ・データ解析で扱わなければならないの変数が多い。

という特徴がある。各 SS について御社様が持っているデータ、MPSI から購入する競合 SS のデータとアンケートに答えてもらったモニターのデータ、カード会員などの特定の顧客のデータをすべて組み合わせて解析および予測を行う。変数としては次のようなものを考慮する。

メッシュごとに分けられているデータ

メッシュごとに利用可能なデータはメッシュ毎に1変数として扱う。下図のように解析をするすべてのSSに対して相対的な位置ができるだけ同じになるようにメッシュをきる 것이望ましい。

		御社 1		

		御社 2		

潜在的な需要に関するデータ

- ・メッシュの中心とSSとの距離(もし利用可能であれば来店時間、帰店時間、走行時間を使う。不明な場合は直線距離で代用)
 - ・メッシュ内の昼間人口
 - ・メッシュ内に登録されている車の台数
 - ・メッシュ内の累積交通量
 - ・メッシュ内の世帯数
- など

メッシュ内の競合SS

- ・競合SSと御社SSとの距離(もし利用可能であれば走行時間を使う。不明な場合は直線距離で代用)
 - ・競合SSの規模
 - ・競合SSの売上
- など

メッシュ内の他の商業施設

- ・他の商業施設と御社SSとの距離(もし利用可能であれば走行時間を使う。不明な場合は直線距離で代用)

- ・他の商業施設の規模
 - ・他の商業施設の売上
- など

対象となる SS そのものの属性

- ・前面中央分離帯の有無
 - ・側面中央分離帯の有無
 - ・前面交通量
 - ・ガソリンフル軽量機数
- など

顧客に関するデータ

もともと地図情報に入っていないが、プログラマ的に地図に埋め込めるものであれば、顧客に関する情報もメッシュ内の変数として扱うことも考慮する(ただし、顧客データの形式がどのようなものであるか現時点では不明なので工数の評価が難しい)。

カード顧客関連

カード顧客は特にロイヤリティが高いと考えられるので、解析に用いる。

- ・カード顧客の住所と SS との直線距離(理想的には、来店時間、帰店時間、走行時間を使うべきだが、不明な場合は直線距離で代用)
 - ・カード顧客の利用頻度
- など

アンケートに答えてもらったモニターに関するデータ

- ・モニタがきた位置と SS との直線距離(理想的には、来店時間、帰店時間、走行時間を使うべきだが、不明な場合は直線距離で代用)
 - ・モニタの利用頻度
- など

留意点

データの区別に関して

基本的に MPSI のデータと御社様の持っているデータは2種類のソースと考えられるので、たとえ同じ名前を持つ項目があったとしても、その内容は整合的であるとは限らない。その時は重複のない2つの項目として扱う。

解析時に新たに付け加えるべき変数

競合店か否かを区別するフラグ、商業施設か否かを区別するフラグ、カード顧客であるか否かを区別するフラグ、アンケートに答えてもらったモニタであるかを示すフラグなど。

データのプレシーズニング（前処理）

データの形式を揃える

すべての SS に関するデータを一律なフォーマットにする（この部分で PERL を用いたデータの整形作業が大量に発生する）。

数値データの正規化

すべての変数は 0.1 から 0.9 までの実数値（二値データの場合、0,1 を使うとニューラルネットワークが収束しない場合があることがわかっている）をとるようにデータを正規化する。これによってすべての変数の相対的な寄与率を同じにすることができる。数値データは最大値、最小値を検出したあと、それぞれが、0.9,0.1 にマッピングするような正規化を行う（この部分で PERL を用いたデータの整形作業が大量に発生する）。

二値データの正規化

中央分離帯有無のような二値データは、0.9,0.1 にマッピングするような正規化を行う。

カテゴリカルデータの正規化

カテゴリカルデータ（文字列によるデータ）はデータに現れるすべてのカテゴリを列挙して、正の整数値にアサインする。次に[0.1,0.9]の間に等間隔にマッピングするような正規化を行う。なおオープンアンサーなどの回答は解析に含めない（この部分で PERL を用いたデータの整形作業が大量に発生する）。

データの数を増やす

次のような方法によって、学習に用いるデータの数を増やすことが可能である。

データの直積によって大量のサンプルデータを作る

SS1に関してメッシュによって分けられている情報(データMと呼ぶ)と顧客データのようにメッシュの形で入っていないデータ(データNと呼ぶ)があるとする。MxNを次のように定義することによって、顧客の人数分のサンプルデータを生成することができる。

サンプルデータ1

= (

前面中央分離帯有無、側面中央分離帯有無、前面交通量、.....

メッシュ1登録車両台数、メッシュ2登録車両台数、メッシュ3登録車両台数、.....、メッシュ100登録車両台数、

メッシュ1昼間人口、メッシュ2昼間人口、メッシュ3昼間人口、.....、メッシュ100昼間人口、

メッシュ1SSまでの走行時間、メッシュ2SSまでの走行時間、メッシュ3SSまでの走行時間、.....、メッシュ100SSまでの走行時間、

.....

カード会員1自宅からSSまでの走行時間、カード会員1利用頻度、カード会員1週末外出頻度、.....、カード会員1自家用車保有台数

)

サンプルデータ2

= (

前面中央分離帯有無、側面中央分離帯有無、前面交通量、.....

メッシュ1登録車両台数、メッシュ2登録車両台数、メッシュ3登録車両台数、.....、メッシュ100登録車両台数、

メッシュ1昼間人口、メッシュ2昼間人口、メッシュ3昼間人口、.....、メッシュ100昼間人口、

メッシュ1SSまでの走行時間、メッシュ2SSまでの走行時間、メッシュ3SSまでの走行時間、.....、メッシュ100SSまでの走行時間、

.....

カード会員2自宅からSSまでの走行時間、カード会員2利用頻度、カード会員2週末外出頻度、.....、カード会員2自家用車保有台数

)

ただしカード会員1、カード会員2はSS1を利用する顧客である。同様にSS2についても直積を定

義し、SS2 を利用するカード会員分のデータを作ることができる。

ブートストラップによって大量のサンプルデータを作る

元のデータ

= (

前面中央分離帯有無、側面中央分離帯有無、前面交通量、.....

メッシュ1登録車両台数、メッシュ2登録車両台数、メッシュ3登録車両台数,.....、メッシュ100登録車両台数、

メッシュ1昼間人口、メッシュ2昼間人口、メッシュ3昼間人口,.....、メッシュ100昼間人口、

メッシュ1SSまでの走行時間、メッシュ2SSまでの走行時間、メッシュ3SSまでの走行時間,.....、メッシュ100SSまでの走行時間、

.....

メッシュ1カード会員人数、メッシュ2カード会員人数、メッシュ3カード会員人数,.....、メッシュ100カード会員人数

)

とすると、[0.1,0.9]上で一様乱数を発生させ、1 番目の変数を発生させた乱数で置き換える。すなわち

サンプルデータ1

= (

乱数、側面中央分離帯有無、前面交通量、.....

メッシュ1登録車両台数、メッシュ2登録車両台数、メッシュ3登録車両台数,.....、メッシュ100登録車両台数、

メッシュ1昼間人口、メッシュ2昼間人口、メッシュ3昼間人口,.....、メッシュ100昼間人口、

メッシュ1SSまでの走行時間、メッシュ2SSまでの走行時間、メッシュ3SSまでの走行時間,.....、メッシュ100SSまでの走行時間、

.....

メッシュ1カード会員人数、メッシュ2カード会員人数、メッシュ3カード会員人数,.....、メッシュ100カード会員人数

)

サンプルデータ2

= (

前面中央分離帯有無、乱数、前面交通量、.....

メッシュ1登録車両台数、メッシュ2登録車両台数、メッシュ3登録車両台数,....、メッシュ100登録車両台数、

メッシュ1昼間人口、メッシュ2昼間人口、メッシュ3昼間人口,....、メッシュ100昼間人口、

メッシュ1SSまでの走行時間、メッシュ2SSまでの走行時間、メッシュ3SSまでの走行時間,....、メッシュ100SSまでの走行時間、

.....

メッシュ1カード会員人数、メッシュ2カード会員人数、メッシュ3カード会員人数,....、メッシュ100カード会員人数

)

変数の個数分の学習データができる。

直積とブートストラップを組み合わせることで大量のサンプルデータを作る

上記2つの方法を組み合わせることも可能である。

解析法およびデータ

三つの解析法

今回用いるのは、重回帰モデル、樹状モデル、ニューラルネットワークの3つである。重回帰は線形データに適用可能、他の2つは非線形なデータに関しても適用可能である。

データの形式

1(0,0)	2(0,1)	3(0,2)	4(0,3)	5(0,4)
6(1,0)	7(1,1)	8(1,2)	9(1,3)	10(1,4)
11(2,0)	12(2,1)	御社 13(2,2)	14(2,3)	15(2,4)
16(3,0)	17(3,1)	18(3,2)	19(3,3)	20(3,4)
21(4,0)	22(4,1)	23(4,2)	24(4,3)	25(4,4)

メッシュごとのデータを次のようなCSVファイルに落とす。

メ シ ュ	メ シ ュ	メ シ ュ	...	昼間 人口	SSま での	登録 車両	登録 車両	正規 化さ	正規 化さ	正規 化さ	正規 化さ
-------------	-------------	-------------	-----	----------	-----------	----------	----------	-----	-----	----------	----------	----------	----------	-----	-----

通し 番号	x 座 標	y 座 標			走行 時間	台数	台数			れ た 昼 間 人 口	れ た ま で の 走 行 時 間	れ た 登 録 車 両 台 数	れ た 登 録 車 両 台 数		
1	0	0													
2	0	1													
3	0	2													
4	0	3													
5	0	4													

アンケートのモニタやカード会員のデータは次のような CSV ファイルに落とす。

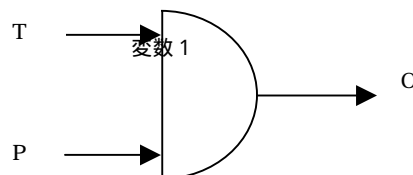
サン プル 番号	御社 SS に一意に つけられ た ID	当該 SS に対 する 位置	当該 SS に対 する x 座 標	当該 SS に対 する y 座 標			性別	年 齢							
1	ide45628	4	0	3											
2	ide45679	5	0	4											
3	ide31890	10	1	4											
4	ide40231	3													
5	ide45012	13													

3つの解析法の内容と特徴

この節では重回帰、ニューラルネットワーク、樹状モデルについてその共通点と相違点などを述べる。

重回帰モデル

いまあるサービスステーションの売上高 O 、交通量 T 、ガソリン価格 P の関係を知りたいとする(これをシステム S と記す)。



現実の中のシステムにはさまざまな因果関係や影響があり、 O, T, P の間の関係を表現する厳密な数式を書き下すことはもちろん不可能である。重回帰モデルを使うということは O, T, P の間にとりあえず、線形関係が成り立つと仮定することである(非常に乱暴な仮定ではある)。

$$O = aT + bP + c$$

実測値と予測値の間の誤差(最小二乗誤差)が小さくなるように a,b,c を決めてやるのが重回帰の本質である。

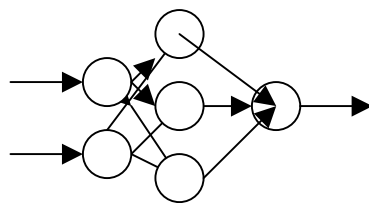
ところが重回帰は次のような単純なデータ(この例は XOR と呼ばれる)さえ、回帰させることができない(このような例は無数にある)。

	T	P	O
データ1	0	0	0
データ2	1	0	1
データ3	0	1	1
データ4	1	1	0

このようなデータは現実世界にはいくらでもあり、そこに重回帰の限界がある。

ニューラルネットワーク

ニューラルネットワークや樹状モデルはいずれも machine learning と呼ばれるカテゴリーに属する方法であり、重回帰が線形モデルの存在をアприオリに仮定しているのに対し、ニューラルネットワークや樹状モデルでは、アприオリに特定の形のモデルの存在を仮定せず、アルゴリズムによってシステム S と非常に近い振る舞いをする擬似システム S' を作り出すことを主眼としている。



ニューラルネットワークは上のような構成をする素子の集まりである。1 層目は入力層、2 層目は中間層、3 層目は出力層である。素子の数はそれぞれいくつにとってもよい(上の T,P,O の例ならば入力層の素子の数は 2 にとり、出力層の素子の数は 1 にとる)。

1 層目と 2 層目は

$$I_i^2 = F\left(\sum_j^N w_{ij} O_j^1\right)$$

の関係で結ばれている。ここで I_i^2 は 2 層目の i -番目の素子への入力、 O_j^1 は 1 層目の j -番目の素子からの出力、 w_{ij} は 2 層目の i -番目の素子と 1 層目の j -番目の素子との間の結合係数、 F はシグモイド関数である。2 層目と 3 層目の素子の間にも同様な関係が成立する。すなわち

$$I_i^3 = F' \left(\sum_j v_{ij} O_j^2 \right)$$

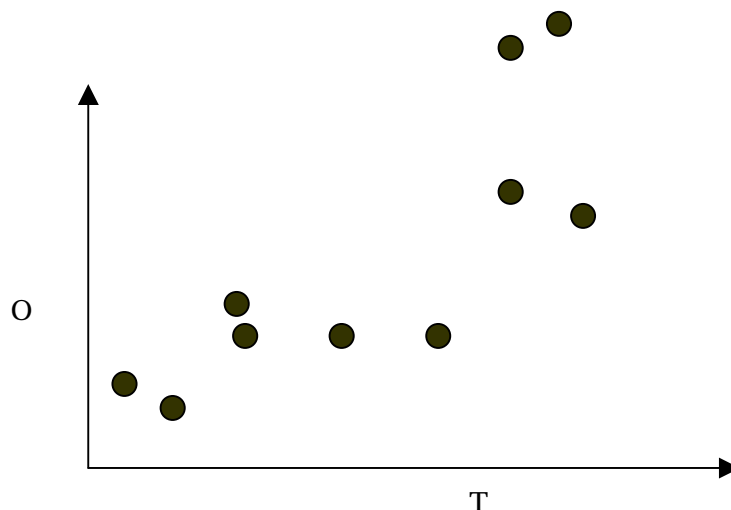
ここで I_i^3 は 3 層目の i -番目の素子への入力、 O_j^2 は 2 層目の j -番目の素子からの出力、 w_{ij} は 2 層目の i -番目の素子と 1 層目の j -番目の素子との間の結合係数、 F' はシグモイド関数である。

ニューラルネットワークの学習は、出力がデータ 1 からデータ n までを順次入力層に与えたとき、出力素子がそれぞれ O_1, O_2, \dots, O_n に近づくように w_{ij} の値を調節することと定義される。

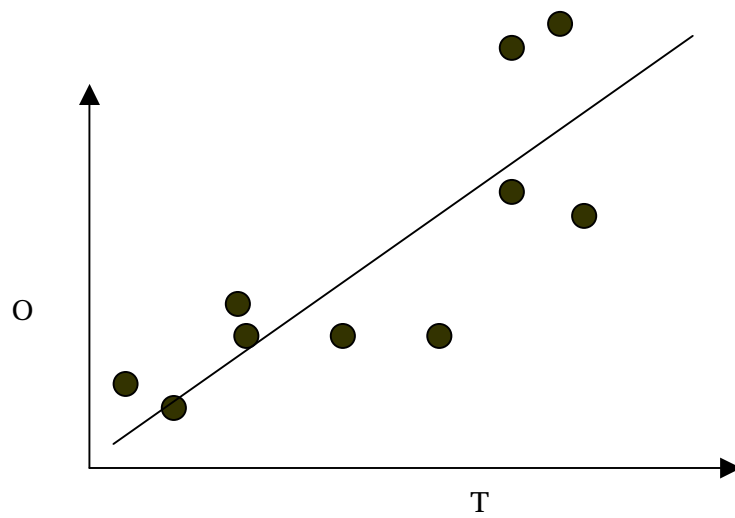
	入力1	入力2	出力
データ1	T1	P1	O1
データ2	T2	P2	O2
データ3	T3	P3	O3
データ4	T4	P4	O4
.....			
データ n	T_n	P_n	O_n

ニューラルネットワークは重回帰が失敗する XOR のようなデータであっても問題なく学習できる。どのように w_{ij} を変えていくかというアルゴリズムは多数あるが、逆誤差伝播法 (BACK-PROPAGATION) がよく使われる (これについては説明を省略させていただきます)

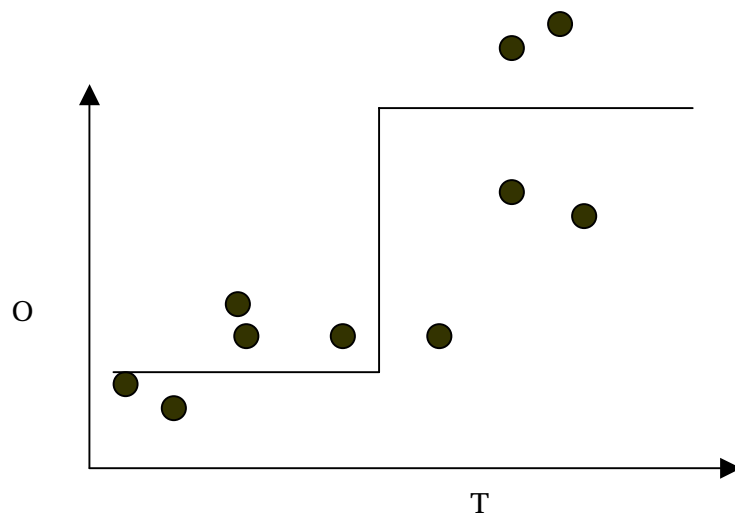
樹状モデル (または樹形モデル)



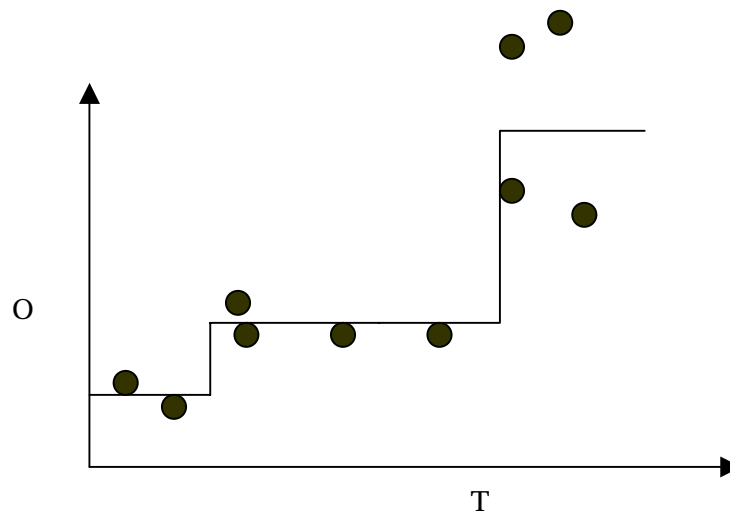
今売上と交通量の間に関係があるとする。1個の は1つのサービスステーションからのデータを表す。回帰モデルの場合は



のようなモデルを選択することになるが、明らかに中央部や右半分でデータをうまく反映していない。一方樹状モデルでは



といったモデルや



といったモデルが構成される(モデルを構成するためのアルゴリズムについての説明は省略させていただきます)。明らかに回帰より正確にデータを反映している。T、P、O の場合は3次元空間内に散らばった $(T_1, P_1, O_1), (T_2, P_2, O_2), (T_3, P_3, O_3) \dots (T_n, P_n, O_n)$ をうまく近似するようなカクカクした平面が生成される。

なぜ樹状モデルやニューラルネットワークは重回帰よりも優れているか？

現実のデータでは変数間にさまざまな相互作用が起こる。たとえば売上は一般に交通量が大きければ高くなるが、交通量が大きくなってもたとえば近隣に競合店があれば、競合店がない場合に比べて売上は伸びない。このような変数間の相互作用を回帰モデルは反映できない。変数間に相互作用がある場合、特にニューラルネットワークは優れている。

	重回帰	ニューラルネットワーク	樹状モデル
目的	回帰、予測、分類、判別	回帰、予測、分類、判別	回帰、予測、分類、判別
計算時間	短い	長い	やや長い
非線型データへの適用	不可	可能	可能
理解しやすさ	しやすい	やや難しい	やや難しい

学習データとテストデータ

データが100個ある場合、学習に99個を使い、1個をテストに使うことができる。このような方法を交差確認法(CROSS - VALIDATION)という。交差確認法によって実際の売上とニューラルネットワークや樹状モデルが予測した値との差が小さければ、ニューラルネットワークや樹状モデルは現実のシステム S を十分によく近似している擬似システム S' を構成したと考えられる。

今101番目の新規店舗については売上高のデータはないが、その他のデータはすべてそろっているとす。101番目の売上を学習の終了したニューラルネットワークや樹状モデルで予測することが、このプロジェクトの1つの主題である。

実際の解析にあたって

欠損値 (missing value) の扱い

データがすべての変数に対して存在するわけではないので、欠損値をうまく処理することが必要である。販売量や他の変数と相関を持たないように、一様乱数を[0.1,0.9]で発生させ、データとして用いる。

外れ値 (outlier) の扱い

初めは外れ値を除外しないで解析を行う。正規化を行っているので、外れ値の寄与は他の正常なデータに比較して、大きくなる。結果が著しく不合理な場合には、外れ値を除外する必要がある可能性がある。

学習の回数

変数が多いので、ニューラルネットワークの学習は回数を増やすことが必要であることが予測される(樹状モデルの場合収束は一般に早い)。学習するデータ数をSS20店舗分で10000個とすると、1回のループでその10000個のデータを学習させる。収束させるために多数回のループを回す。

交差確認法 (cross-validation)

樹状モデルについては n-重の交差確認を行うことを考慮する(S-plus ではビルトインで交差確認

法機能が使える)。ニューラルネットワークではプログラミングが複雑になりすぎるので、特に交差確認法は用いない。

過学習 (over-learning) に対する対策

学習によってできたモデルが過学習に陥ると、データに対するエラーは小さくなるが、予測能力が落ちる場合がある。その場合、いくつかの方法でモデルを単純化する。

- ・AIC(Akaike Information Criteria)を使う。
- ・樹状モデルの場合は pruning または shrinking によってモデルを単純化する。
- ・中間層の素子の数を減らす。

などを使うことを考慮する。

変数の間引き

解析が進むと販売量に寄与している変数とそうでない変数が区別できると考えられるので、寄与度の小さい変数は間引く。これによって計算時間の短縮が図れる。

データから決定される商圈の設定

商圈は従来、走行時間 3 分、5 分、10 分、15 分といったもので決められてきたが、この中で解析のなかで、販売量と相関が強いメッシュの位置を決定できる。従来の商圈とどのように関係するか、あるいはしないかを見ることができる。

変数間の関係の調査

方程式の構築

学習が終了した樹状モデルやニューラルネットワークは、変数に値を入力することによって販売量を出力する。この方程式を書き下すために、ニューラルネットワークの結合係数や樹状モデルのトポロジーを調べることによって、変数間の関係、ある変数と販売量との関係の強さを決定する。

計算可能性

上述の計算式は実際に計算可能な関数から構成される。計算式自体は複雑だが、段階を追えば必ず計算のできる関数である。

