１．目的

「与えられたすべてのデータ点の近くを通るように直線(ただし)を引くというのが、最小二乗法を利用して解くことのできる問題である。ここでは、その問題を拡張し、が与えられた場合においても、すべてのデータ点の近くを通る性質を持つ超平面

におけるを決定する問題を扱う。なお、を説明変数という。このような問題を解くことが工学的に重要であることは、次のようにして理解できる。まず、与えられたデータ点がどのような現象から得られたデータであるかということが重要であり、とのように2次元内のデータであったとしても、その現象を記述する数理モデルとして直線(ただし)が最良のモデルであるとは限らない。例えば、多項式を利用して、

すべてのデータ点の近くを通る性質を持つ曲線を構築した方が、その現象をより良く説明する数理モデルになっている場合もある。すると、直線でも超平面でもないために、第３の問題が出現したように思うかもしれないがその認識は正しくない。のように2次元内のデータに対して、ユーザが新しいデータを計算によって生成することができる。すなわち、が与えられているから、を計算することが容易である。すると、次元のデータ点が与えられた場合、すべてのデータ点の近くを通る性質を持つ超平面

を決定する問題を扱うコンピュータログラムを利用すれば、多項式

の係数を決定できる。よっての２次元データに対して、多項式の数理モデルを計算することは第３の問題と呼ぶべきではない。さらに言えば、の２次元内のデータに対して、次の数理モデルあるいはlogただし、を考える。この問題も、第３の問題ではない。が与えられているから、を計算することは容易であるため、データ点が与えられたのではなくデータ点が与えられてと考えれば、「直線(ただし)を引く」問題と解釈できる。

同様のことは次の問題についても言える。は次のように式変形できるためである。　　　　　　　　　　　　　　　log

以上の例で見たように、直線または超平面を決定するコンピュータプログラムの応用範囲はとても広く、実験のデータ整理のために極めて有益な道具となるために、この実験においては、それを作成する。さらによりよい別の数理モデルが存在するかもしれないというアラートを発する機能までも備わっているため、その機能についても各自で確認を行う。

２．原理

説明変数が重要な働きをしないことを確認する方法

が０である確率を計算する方法について、詳細な説明は省略し結論のみ紹介する。まず、すべての変数が0でないと仮定しRSS(二乗和残差)を計算する。RSSは、

RSS＝と定義する。ここでのRSSをと書き、この時の検定統計量

for (1)

とF分布の数表と照らし合わせると、が０である確率を計算できる。が０である確率が高いならば、説明変数が重要な働きをしていないと考えられる。しかし、F分布の数表が必要となるため、この実験では、が０である確率を計算するために、C言語のプログラミングではなく、フリーソフトウェアLibreofficeのCalcを利用する。

※ただ１つのについてのもとで、が０である確率を計算する場合には、F分布を利用することは効率が悪い。実は、ｔ分布で代用できることが知られている。

QR分解による最小二乗法の解法

,

とすると、最小二乗法とはを求めることである。

がフルランクの行列の場合には、

であり、は正則な正方行列であるから逆行列が存在し、(2)

となる。数学上は、式(2)の計算法で正しいのであるが、数値計算においては、上記の計算によってを求めることは多大な計算誤差を含むことになるために、採用されない。はじめにをQR分解する。

ここで、は列直行行列であり、を満たす。式(3)を式(2)に代入する。

となる。よってQR分解を行うことができれば、最小二乗法の計算は完了したことになる。しかし、RSSを計算しなければならないことを考えると、を確定させた後に、

RSS=

を計算する必要がある。さらに式(5)を利用してRSSを計算する戦略は、計算精度の観点で優れた戦略とは言えない。そこで、次の行列を考える。

このをQR分解する。を分解するまでの計算が完了し、さらに、のQR分解が完了したとする。となる。これを成分で書く。QR分解が列交換を伴わないことに注目し、のQR分解はと書くことから、

について詳細に書くと、

が成り立つ。QR分解の性質より、が言えるため、

が成り立つ。加えて、

ここで、であるからであるため、RSSが成り立つ。最後に、は、として計算する。正確には、次の連立一次方程式　　　　　を解けばよい。

古典・修正グラムシュミット法によるQR分解

1. 古典グラムシュミット法

ｎ次元計量実ベクトル空間の任意の本のベクトルが与えられているとき、すなわち、行列,　が与えられているとき、次の手続きで正規直交基底をつくることができる。

(13)

行列をQR分解する。 ここで、は列直交行列であり、を満たす。行列と行列は、古典グラムシュミット法により計算でき、具体的には、

となる。上記のとによるという式と、式(10)-(14)が完全に一致することによる。

1. 修正グラムシュミット法

古典グラムシュミット法における、の部分の計算を以下のように変更する。

からの方向成分を抜き取りとするとき、数値誤差によりの方向に変化が起きる。その変化を考慮して、もとのではなくよりりの方向成分を抜き取るほうがより高精度な計算ができるというアイディアに基づいている。この工夫を、を計算するときにも用いる方法を、修正グラムシュミット法という。

式(16)に見たように、古典グラムシュミット法による生成される行列はを含んでいるが、修正グラムシュミット法の計算においてはは現れない・しかし、

であるから、の代わりに、を用いることができる。同様のことが、一般の場合にも成立する。以上より、修正グラムシュミット法による行列は次のようになる。

3.実験方法　a

学籍番号に応じた正解の数理モデルを作成する。すなわち、乱数の種に学籍番号を利用する。その際、０でないランダムに定めた３個の整数について、

とする。この実験では、説明変数のデータを乱数で生成した場合には、修正グラムシュミット法によりこの問題を扱うことができないことが知られているために、説明変数のデータは所定のURLよりダウンロードする。さらに、に値は次の手順で作成する。

で作成したの実数値をソースコードの

+0.1\*(TWO\*(rand()/(double)RAND\_MAX)-ONE)でノイズを混入させる。

次に、が与えられた場合において、すべてのデータ点の近くを通る性質を持つ超平面

におけるを決定する問題を解く。を決定するためには、QR分解を行えばいい。そこで、QR分解のコンピュータプログラムを作成する。その際には、修正グラムシュミット法を採用する・QR分解の実装においては、統計学において重要なRSSを計算する機能をその計算法の内部に含む。この実験の目的は、検定統計量の性能を評価することとする。すなわち、はじめに、すべての変数が０でないと仮定し、修正グラムシュミット法を利用して、RSSを計算する。次に、を除く特定のを固定し、のもとでRSSを計算する。はからまでの種類が存在するため、合計で個のを計算する必要がある。RSSとを利用して検定統計量を計算し、横軸は,縦軸はを採用したグラフを作成し、を示唆する結果を得られたかどうかの検討を行う。しかし、検定統計量からが０である確率を計算するためには数表と照らし合わせる必要があるため、時間を短縮するために同じ実験データからLibreofficeを利用し、が０である確率を計算する。すべてのについて、０である確率が高いならば、すべての説明変数が重要な働きをしていないことになるため、より良い別の数理モデルを探すべきという結論となる。

今回の実験では世界各国１６６ヶ国の１２か月の気象データを基に説明変数をとし、実験結果に示すコードから求めた幸福度をとし実験を行う。また、2月・５月・１０月の気温は幸福度に影響しないとして、

としている。

４．実験結果

実験に用いたソースコードを以下に示す。

%%file mgs\_solve\_2.c

#include<stdio.h>

#include<stdlib.h>

#include<math.h>

#define ZERO (0.0)

#define ONE (1.0)

#define TWO (2.0)

#define M 166

#define N 14

double innerp(int n,double \*x,double \*y,int stride){ //成分がｎ個のベクトルｘとベクトル//ｙの内積

    double temp;//総和を維持するための変数tempを用意

    int i; //ベクトルの特定のインデックスを指し示すためのカウンタiを用意

    temp=ZERO; //tempをZEROで初期化

    for(i=0;i<n;i++){  //iでループ

        temp+=x[i\*stride]\*y[i\*stride]; //xの要素とyの要素を掛け算しtempに足しこむ

    }

    return temp; //tempを呼び出し元へ戻す

}

int main(){

    double temp;//変数tempを用意

    int i,j,k;//変数i,j,kを用意

    double (\*A)[N],(\*R)[N],\*X;//行列A,行列R,ベクトルｘのためのポインタを用意

    A=(double (\*)[N])malloc(sizeof(double)\*M\*N);//行列Aの実態を実体を作成

    R=(double (\*)[N])malloc(sizeof(double)\*N\*N);//行列Rの実体を作成

    X=(double \*)malloc(sizeof(double)\*(N-1));//ベクトルXの実体を作成

    if(A==NULL || R==NULL || X==NULL){//行列A,行列R,ベクトルXのいずれかが //NULLポインタならば、

        printf("Out of memory!\n");//エラーを表示

        return 0;//終了

    }

    srand(22221280);//乱数の種を学籍番号の１に初期化

    for(i=0;i<N-1;i++){//変数iを0からN-2まで変化させる

        X[i]=TWO\*(rand()/(double)RAND\_MAX)-ONE;//各月の気温に正解a(i)を乗算して幸福度を出す。（正解の作成）

}

    X[2]=ZERO;//2月の気温は幸福度に影響しないという状況を作るためにa(2)=0にする

    X[5]=ZERO;//5月の気温は幸福度に影響しないという状況を作るためにa(5)=0にする

    X[10]=ZERO;//10月の気温は幸福度に影響しないという状況を作るためにa(10)=0にする

    printf("ANS\n");//正解を表示

    for(i=0;i<N-1;i++){//変数iを0からN-2まで変化させる

        printf("%g\n",X[i]);//a(i)を表示

    }

    for(i=0;i<M;i++){//変数iを0からM-1まで変化させ、166か国分のループ

        A[i][0]=ONE;//i番目の国の幸福度の計算における切片を用意

        for(j=1;j<N-1;j++){//変数jを1月から12月まで変化させる

            scanf("%lf",&A[i][j]);//i番目の国の気温のデータをファイルから読み込む

        }

    }

    printf("Happy\n");//幸福度を表示

    for(i=0;i<M;i++){//変数iを0からM-1まで変化させ、166か国分のループ

        temp=ZERO;//変数tempを0に初期化

        for(j=0;j<N-1;j++){//変数jを切片0番からはじめ、さらに1月から12月まで変化させる

            temp=temp+A[i][j]\*X[j];//tempに正解a(i)\*気温を足しこむ(ただし0番の切片の場合は正解a(0)\*1を足しこむ)

        }

        A[i][N-1]=temp+0.1\*(TWO\*(rand()/(double)RAND\_MAX)-ONE);//幸福度の配列にtempを代入。ただし、tempをそのまま代入するのではなく、観測誤差も混入させる。

        printf("%f\n",A[i][N-1]);//変数iの国の幸福度を表示

    }

    for(i=0;i<N;i++){//変数グラムシュミット法によるQR分解

        for(j=0;j<i;j++){//

            R[j][i]=innerp(M,&A[0][i],&A[0][j],N);//ベクトルa\_iとベクトルa\_jの内積の計算、結果をr\_{j,i}に格納

            for(k=0;k<M;k++){//

                A[k][i]=A[k][i]-R[j][i]\*A[k][j];//ベクトルa\_iからq\_k方向を抜き取る

            }

        }

        R[i][i]=sqrt(innerp(M,&A[0][i],&A[0][i],N));//ベクトルa\_iの長さを計算

        for(j=0;j<M;j++){

            A[j][i]=A[j][i]/R[i][i];//ベクトルa\_iの長さを1に規格化し、a\_iに代入

        }

    }

    for(i=N-2;i>=0;i--){//連立一次方程式 Rx=Q^Tyを解く

        for(j=i+1;j<N-1;j++){

            R[i][N-1]=R[i][N-1]-R[i][j]\*R[j][N-1];//これまでに確定したR\_{j,N-1}にR\_{i,j}を乗算した後、R\_{i,N-1}から減算する

        }

        R[i][N-1]=R[i][N-1]/R[i][i];//R\_{i,N-1}をR\_{i,i}で除算する

     }

    printf("GUESS\n");//統計的に推定したa(i)の値を表示する

    for(i=0;i<N-1;i++){//変数iを切片0番からはじめ、さらに1月から12月まで変化させる

        printf("%g\n",R[i][N-1]);//統計的に推定したa(i)の値を表示する

    }

    printf("residual:%f\n",pow(R[N-1][N-1],TWO));//RSSを表示

    return 0;//終了

    }

テキスト, チャットまたはテキスト メッセージ

自動的に生成された説明上記のコードをコンパイルし、ダウンロードした説明変数のデータを入力し結果を出力すると、結果は以下のようになった。（幸福度は省略）

テキスト, チャットまたはテキスト メッセージ

自動的に生成された説明

写真１．２　最小二乗法から計算した説明変数と説明変数から計算した幸福度から推定した説明変数と残差

写真より、最小二乗法から計算した説明変数の値は、定義した通りとなっており、また、計算した説明変数からの幸福度から修正グラムシュミット法を利用して推定した説明変数も、ほとんど最小二乗法から計算した説明変数に等しく、もそれぞれ０にほぼ等しい値となっていることが分かる。

次に、コードで得られた結果を用いて、回帰分析を行った。独立変数を世界166ヶ国の気温のデータ、従属変数を幸福度とし、信頼水準を0.90, 0.95, 0.98に変化させ、それぞれの時の変化を確認した。以下にその結果を示す。

表１．信頼水準が0.90のときの回帰分析

新聞の記事

低い精度で自動的に生成された説明

表２．信頼水準が0.95のときの回帰分析

テキスト, 新聞 が含まれている画像

自動的に生成された説明

表３．信頼水準が0.98のときの回帰分析



表より、信頼水準を上げることによって、ある区間にはいる確率が高くなければいけないので、範囲が広くなることが分かる。そして、今回はとして幸福度を計算したので、その幸福度と気温データを用いた回帰分析はP値()がそれぞれでそれ以外のと比べはるかに高いことが分かる。また、信頼区間について、は今回０として実験を行ったので下限から上限で０をまたいでしまいその月の気温が幸福度に対して重要かわからない。これは、P値＜１－信頼水準のとき、が０でないと言えるが、そうでない場合は０かわからないという表現のみが統計では許されている。

信頼水準を0.90から0.98に上げるにつれ下限と上限の幅が広くなっていることから、信頼水準を1.00に限りなく近づけていくと下限と上限が０をまたぐようになると考えられる。そうなると、すべてのが０かわからない状態になってしまうので、どの月の気温が重要であるかわからなくなる。逆に、信頼水準を下げていくと、下限と上限の幅が狭くなるので、P値＜１－信頼水準である可能性が高くなり、０でないと言えるが増え、最終的にはすべての月が重要と考えることとなり、幸福度に関係していない月を探すという目的の観点では役に立たない結果となる。

今回の実験では、数学としての統計において重要であるものを判断することしかできないが、コンビニの売り上げを例として考えると、売り上げ(今回の実験での幸福度)に影響がない商品は仕入れたくないという考えが当然であり、すべてが重要(重要かわからない)であるとなると、仕入れる必要のないものまで仕入れることになったり、どの商品を仕入れるべきかわからなくなったりしてしまう。逆に、全商品がP値＞１－信頼水準となる場合、仕入れるべき商品とそうでない商品の区別は一切できなくなる。

変数のデータに対する重要性の有無を信頼水準によってどう変わるか実験することで一般的に信頼水準が0.95付近で設定される理由を実験によって確かめることができた。