

jDllServer による J 言語用アプリケーション (2)

Visual Studio 2015 VB.NET

鳥邊 錬太郎

2017 年 03 月 11 日

JAPLA 2017/3 総会研究会

於：横浜関内・会場

今回は、昨年 12 月 10 日の (1) の継続で、表題の第 2 回目です。

昨年暮れごろから風邪を患い、また開発マシンの OS が Windows10 に、言語プロセッサも Microsoft Visual Studio 2015 となり、それらへの習熟もてんやわんやでした。そこへ年末に Windows10 への更新の影響で動作不能のアプリケーションが幾つか出てきたため、やむなくマシンの買い替えも余儀なくされました。このような状況で、jDllServer の進捗は遅々として進まず、体力だけは迅速に衰えて行く有様です。

本プログラムは 2007 年 8 月に日本 APL 協会の蓼科研修センターにて開催の JAPLA2007 サマーセミナーで発表したのが始めてでした。

その当時は、忙しく多くの時間を割くことは出来なかったもので、当時はそのまま中断したものです。

今回は、毎日の時間はたっぷり有りますが、生涯の残り時間はわずかなので、この開発が全うされるかどうかは計り知れません。一応、持ち時間を今年度末 (2018 年 3 月) として、頑張ります。

jDllServer という名前は、jsoftware.com の DLL 名そのものですから、この辺でアプリケーションの呼び名を変更してみました。

また、今回の開発プログラムの定義をここに明確にします。そして、何よりこのプログラムは、筆者の単なる独りよがりの行為です。

| | |
|----|--|
| 名称 | jPlanProject On jDllServer j804 By JAPLA (仮称) |
| 目的 | Windows プログラムをベースに jDllServer(j804 版)を組み込み、様々なデータ処理の数値解析部分を J 言語の、 1. J 対話型処理 2. J Script 処理 に担ってもらう汎用データ解析プログラムを作成する。 そして、本プログラムが我々が愛する「J 言語」が世の中でこよなく愛され、多くの愛好者、利用者によって認識されることを目的としています。 |

| | |
|------|---|
| 開発言語 | Microsoft Visual Studio 2015 Visual Basic.NET |
| 特徴 | <ol style="list-style-type: none"> 1. jPlanProject の DataBase に収録する jExpression には、JAPLA 編纂 jExpression とユーザ作成 jExpression の 2 種類に分類することができる。 2. ユーザが本プログラムで処理する単位を Project と呼び、ユーザは、Project 単位でジョブを保存することができる。Project には、ユーザが作成した Win 変数、j 変数、対話型 jExpression、jScript など実行中に削除しなかった全ての要件が保存され、必要なときに何時でも読み込むことができます。 3. ユーザが作成する Project には、DataBase の JAPLA 登録やユーザ登録の j Expression や jScript を自由に組み込むことができる。 |

jDllServer の魅力に取りつかれてから、ほぼ 10 数年になりますが、これまで間違って理解していたことなども多数ありました。かなり丁寧に設計された jDllServer であることに改めて気づいた次第です。

それでは、ここで jDllServer 関数を使って、jDllServer の関数の機能や使用方法を見てみましょう。

jDllServer をプログラムで使用するには、プログラムの頭で、使用するための宣言をする必要があります。

jDllServer は、jsoftware.com からリリースされる、j804 に同梱され、これを当該 PC にインストールすることによって、自動的に VS2015 の参照一覧に登録され、VS2015 に解放されます。

jDllServer の宣言

```
Imports JDLLServerLib
```

J Expression を入力して、[Enter]を押す。

i.3

上記のように、「i.3」や「i.3 3」の J Expression を入力して実行すると、本

プログラムは、以下のような j 関数を起動して処理します。

Text(String 型)の内容に i.3 が入っているとき、

```
ec = Js.Do("J_Get=. " & Text)
```

ec には、実行時のエラーコードが入り。エラーがなければ、結果は jSpace の J_Get という名前の変数に入ります。

この J_Get という変数名は WinSpace では識別出来ません。したがって、この J_Get 変数の内容を WinSpace で利用可能にするには、次の関数を実行する必要があります。

j Space の J_Get を WinSpace で識別するには、以下の関数を使用しますが、意味は、jSpace の J_Get 変数の内容を WinSpace の Object 型 x に取り込む、というものです。

```
ec = Js.GetB("J_Get", x)
```

ただし、x は Object 型

利用例：WinSpace の Excel データを jSpace で利用する。

以下の場合、i.3 3 という jExpression の場合、配列型の処理をご覧ください。

Input 対話型Expression

i.3 3

```
【jExpression-4】i.3 3  
0 1 2  
3 4 5  
6 7 8
```

上図の場合は、Object 型 x には、配列型で x(3,3)の形式で

{Length=9}

(0, 0): 0 (1, 0): 3 (2, 0): 6

(0, 1): 1 (1, 1): 4 (2, 1): 7

(0, 2): 2 (1, 2): 5 (2, 2): 8

のように、セットされているので、これを行列として処理しなければならない。

この他に、まだ幾つかの J 関数がありますが、それらは今後その都度解説したいと思います。

本プログラムの説明

現在まで出来ている部分を説明します。

全体の完成目標は今年度末を目指しています。

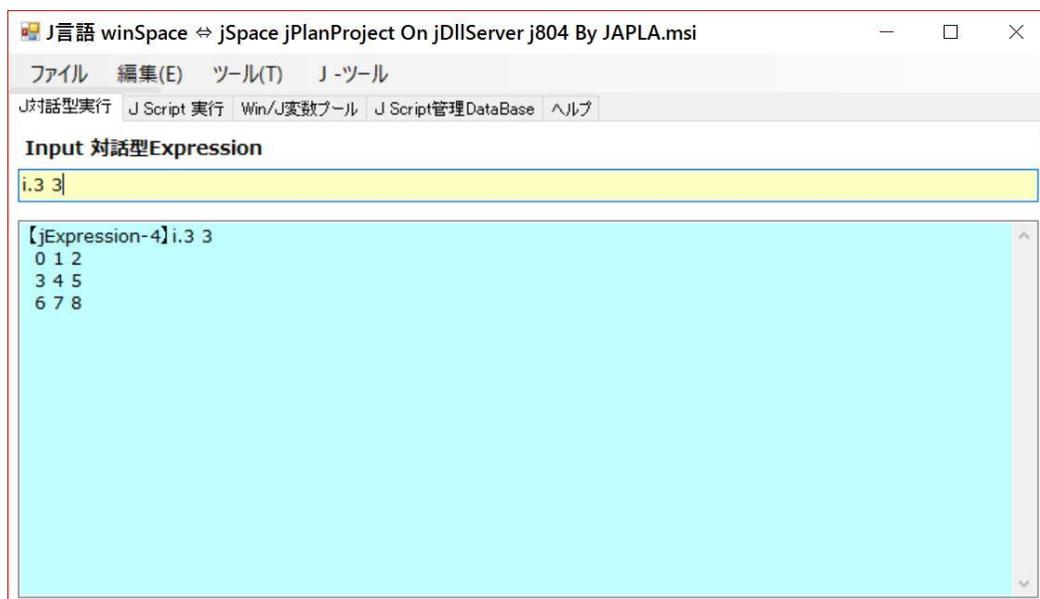
1. プログラムの説明



設計では、5つのタブが設定されています。

1.1. 「J対話型実行」タブ

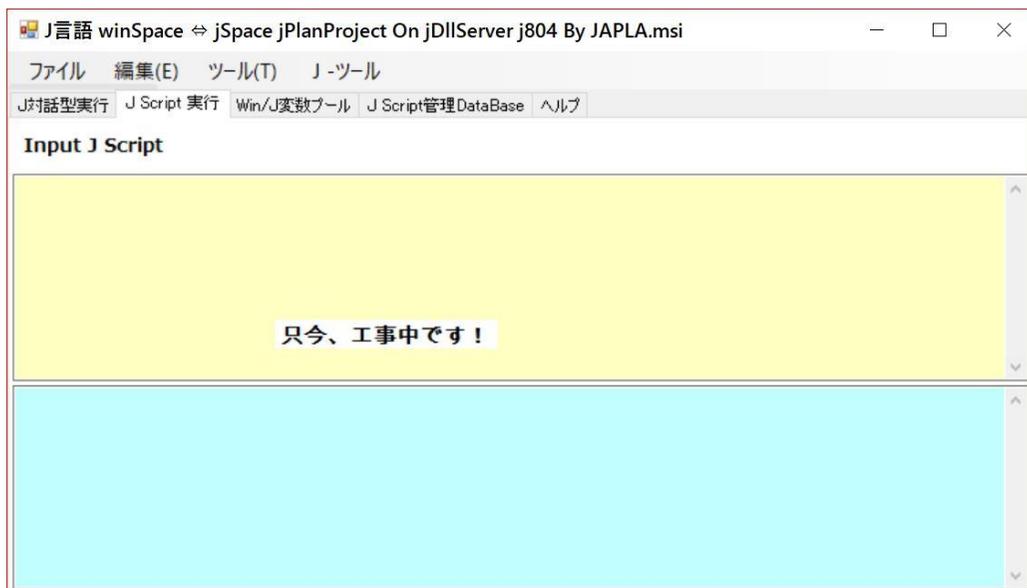
jExpression を対話型で実行するところです。



通常の jExpression を実行できます。結果はインタープリター形式で jExpression を入力して「Enter」キーを叩くとすぐに実行、結果を表示します。

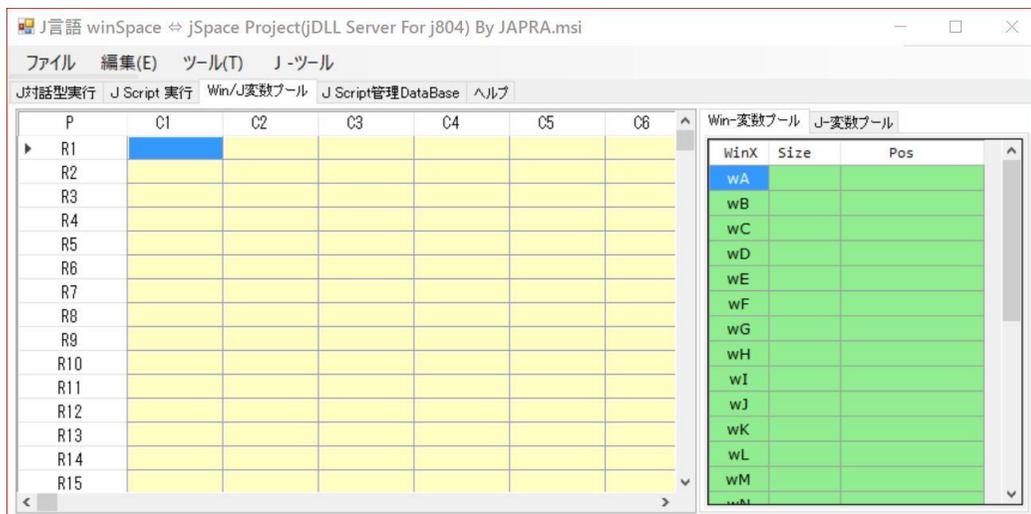
1.2.「jScript 実行」タブ（現在作成中）

jScript を対話型でプログラミング、デバッグしたり、完成した jScript を実行する画面。



1.3.「Win/ j 変数プール」タブ

jScript を対話型でプログラミング、デバッグしたり、完成した jScript を実行する画面。



基本的には、Windows 側から、Excel などのデータをコピーして、このデータシートに貼付け、データ部分を範囲確定することにより j 変数プールに登録します。

Excel データをコピーして、データシートを右クリックしてポップアップメニューから「貼付け」ます。

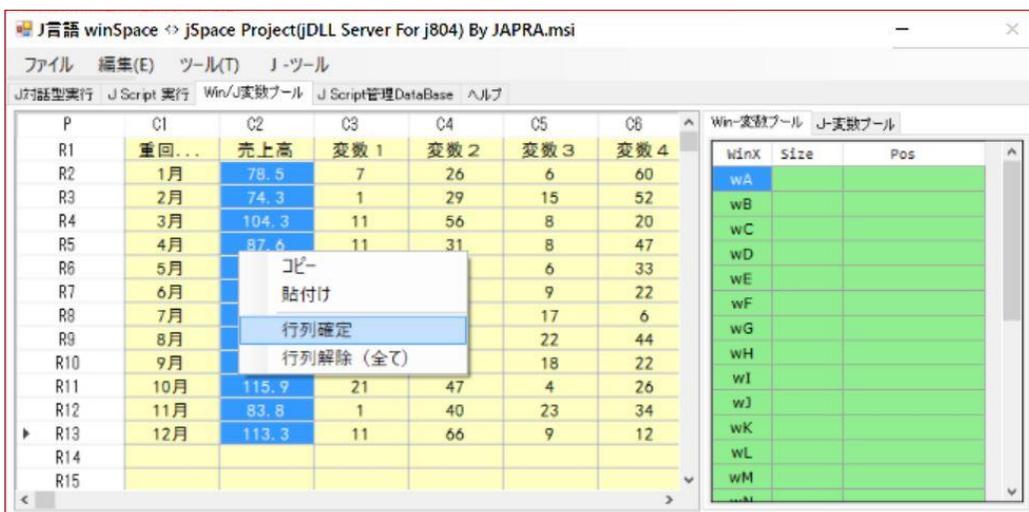
| | 売上高 | 変数1 | 変数2 | 変数3 | 変数4 |
|-----|-------|-----|-----|-----|-----|
| 1月 | 78.5 | 7 | 26 | 6 | 60 |
| 2月 | 74.3 | 1 | 29 | 15 | 52 |
| 3月 | 104.3 | 11 | 56 | 8 | 20 |
| 4月 | 87.6 | 11 | 31 | 8 | 47 |
| 5月 | 95.9 | 7 | 52 | 6 | 33 |
| 6月 | 109.2 | 11 | 55 | 9 | 22 |
| 7月 | 102.7 | 3 | 71 | 17 | 6 |
| 8月 | 72.5 | 1 | 31 | 22 | 44 |
| 9月 | 93.1 | 2 | 54 | 18 | 22 |
| 10月 | 115.9 | 21 | 47 | 4 | 26 |
| 11月 | 83.8 | 1 | 40 | 23 | 34 |
| 12月 | 113.3 | 11 | 66 | 9 | 12 |

データシートの貼付け位置をクリックして、右クリックし、ポップアップメニューの「貼付け」を選択します。



貼付けたデータの、当該データ部分を選択して、右クリックし、ポップアップメニューの「行列確定」を選択します。

ここでは、まず売上高を選択し行列確定をします。



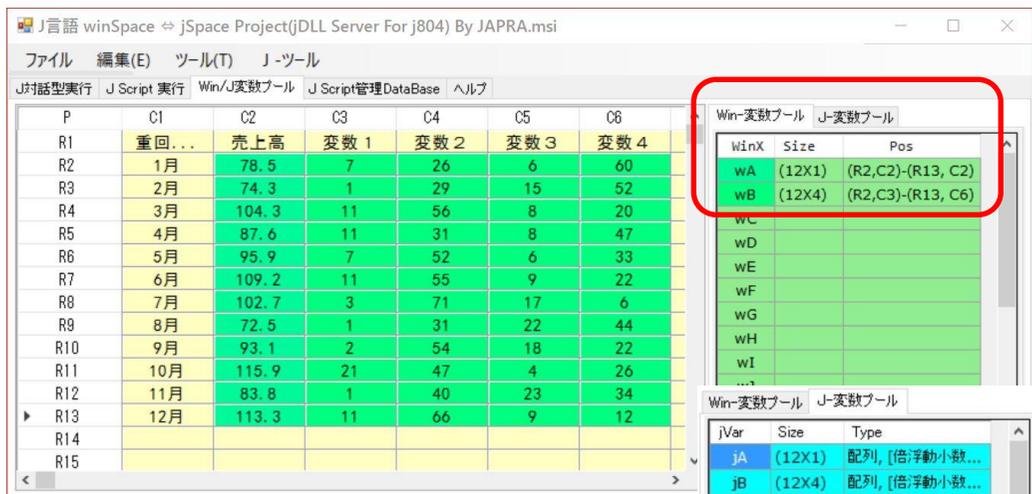
引き続き、変数部分を選択し行列確定をします。



その結果、画面右の変数プール（赤枠）には、2つの Win 変数が登録されていることが解ります。

変数プールの右側のタブ（J-変数プール）をクリックすると、

Excel データが J 変数に変換されて j 変数が割り当てられていることが解ります。



また、J-変数プールに登録された j 変数には、

売上高に jA が

要因変数に jB が

jSpace で使用できる変数名で登録覚ました。

この変数名を用いて、J 対話型 Expression や

J Script を作成して、J 言語プログラムを実行

できます。

右側変数
プールの内容

1.4.「J Script 管理 DataBase」タブ（現在設計中）

「J Script 実行」タブで作成した jScript を系統付けされた jScriptDataBase に登録することができます。

自作の jScript や、その他の jScript の登録、編集などができます。

ご期待下さい。

実際の計算問題を当てはめてみる。

J 言語で計算するために、重回帰分析を考えます。

a) 売上高と会社指標データ

| jDIIServer | | | | | |
|------------|-------|------|------|------|------|
| 重回帰分析 | 売上 | 変数 1 | 変数 2 | 変数 3 | 変数 4 |
| 1 月 | 78.5 | 7 | 26 | 6 | 60 |
| 2 月 | 74.3 | 1 | 29 | 15 | 52 |
| 3 月 | 104.3 | 11 | 56 | 8 | 20 |
| 4 月 | 87.6 | 11 | 31 | 8 | 47 |
| 5 月 | 95.9 | 7 | 52 | 6 | 33 |
| 6 月 | 109.2 | 11 | 55 | 9 | 22 |
| 7 月 | 102.7 | 3 | 71 | 17 | 6 |
| 8 月 | 72.5 | 1 | 31 | 22 | 44 |
| 9 月 | 93.1 | 2 | 54 | 18 | 22 |
| 10 月 | 115.9 | 21 | 47 | 4 | 26 |
| 11 月 | 83.8 | 1 | 40 | 23 | 34 |
| 12 月 | 113.3 | 11 | 66 | 9 | 12 |

表 1 目的変数（売上高）と独立変数（変数 1, 2, 3, 4）

表 1 は、ある会社の年間の月別売上高と売上高を左右するとおもわれる会社指標のデータです。

このデータをもとに、来年の 1 月の売上高を予測するモデルを考えてみます。ただし、来年 1 月の各変数の指標は、以下のとおりです。この指標を回帰モデルに当てはめて、売上高を予測します。

b) データと行列を関連付けるために、重回帰分析の一般形を示します。

$$Y = \begin{pmatrix} 78.5 \\ 74.3 \\ 104.3 \\ 87.6 \\ 95.9 \\ 109.2 \\ 102.7 \\ 72.5 \\ 93.1 \\ 115.9 \\ 83.8 \\ 113.3 \end{pmatrix} \quad x = \begin{pmatrix} 7 & 26 & 6 & 60 \\ 1 & 29 & 15 & 52 \\ 11 & 56 & 8 & 20 \\ 11 & 31 & 8 & 47 \\ 7 & 52 & 6 & 33 \\ 11 & 55 & 9 & 22 \\ 3 & 71 & 17 & 6 \\ 1 & 31 & 22 & 44 \\ 2 & 54 & 18 & 22 \\ 21 & 47 & 4 & 26 \\ 1 & 40 & 23 & 34 \\ 11 & 66 & 9 & 12 \end{pmatrix}$$

正規方程式を作成するため、 x を X に変形します。

$$X = \begin{pmatrix} 1 & 7 & 26 & 6 & 60 \\ 1 & 1 & 29 & 15 & 52 \\ 1 & 11 & 56 & 8 & 20 \\ 1 & 11 & 31 & 8 & 47 \\ 1 & 7 & 52 & 6 & 33 \\ 1 & 11 & 55 & 9 & 22 \\ 1 & 3 & 71 & 17 & 6 \\ 1 & 1 & 31 & 22 & 44 \\ 1 & 2 & 54 & 18 & 22 \\ 1 & 21 & 47 & 4 & 26 \\ 1 & 1 & 40 & 23 & 34 \\ 1 & 11 & 66 & 9 & 12 \end{pmatrix}$$

この変形は、 x 行列の先頭列の前にすべて 1 の列を付加したものです。

これらの行列定義のもとに、以下の行列演算式で、 \hat{B} を推定するのが、重回帰分析の目的です。その計算は、以下の通りです。

$$\hat{B} = (X'X)^{-1}X'Y$$

(注意) 行列演算は、行列の順序が大事です。積の順序を変えると計算が不能になったり、計算できても全く意味の違った結果になります。

また、この計算式が導かれる経過は、最小二乗法と呼ばれる方式をもちい、その方程式を求めるべき未知数 $\hat{\beta}$ でそれぞれ偏微分して方程式を作成します。

ちなみに、 \hat{B} は、

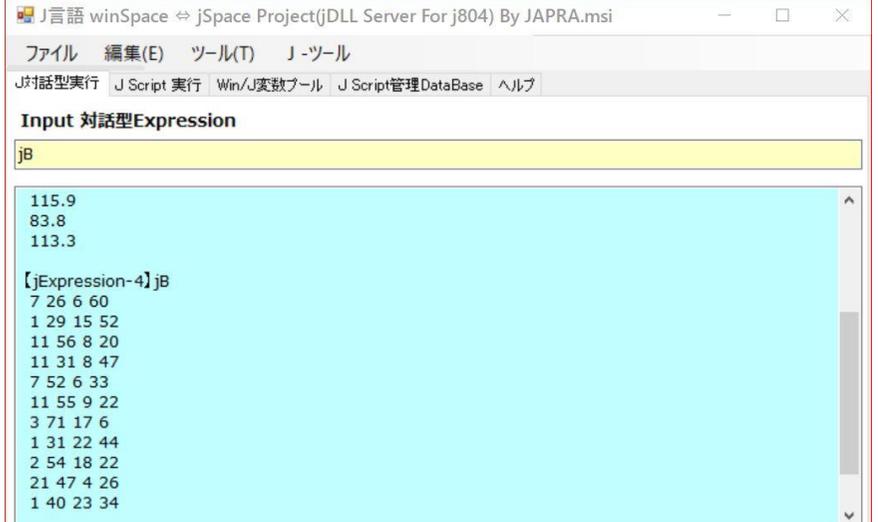
$$\hat{B} = \begin{Bmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \\ \hat{\beta}_2 \\ \hat{\beta}_3 \\ \hat{\beta}_4 \end{Bmatrix}$$

のように表現し、この重回帰分析モデルの推定回帰係数と呼ばれます。

7 ページのデータを EXCEL にコピーして、本プログラムで実行してください。

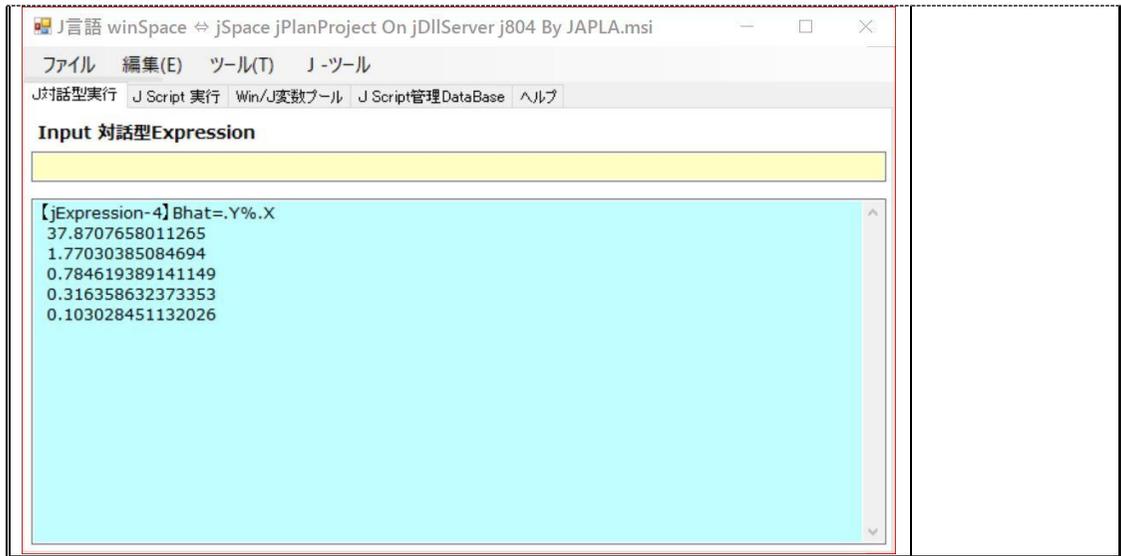
データを登録（5, 6 ページ参照）してから、下記表の右端列の J Expression を入力ライン（黄色）に貼付けて[Enter]で実行してください。

| データ表示 | J Expression 表記 |
|---|-----------------|
|  <p>The screenshot shows a window titled 'J言語 winSpace ⇔ jSpace Project(jDLL Server For j804) By JAPRA.msi'. The menu bar includes 'ファイル', '編集(E)', 'ツール(T)', and 'J-ツール'. Below the menu is a toolbar with buttons for '対話型実行', 'J Script 実行', 'Win/J変数プール', 'J Script管理DataBase', and 'ヘルプ'. The main area is titled 'Input 対話型Expression' and contains a yellow input field with 'jA'. Below this is a light blue output area labeled '[jExpression-4] jA' containing a list of numerical values: 78.5, 74.3, 104.3, 87.6, 95.9, 109.2, 102.7, 72.5, 93.1, 115.9, 83.8, and 113.3.</p> | <p>Y=.:i.3</p> |

| | |
|---|---------------------------------------|
|  <p>Input 対話型Expression</p> <pre>jB</pre> <pre>115.9 83.8 113.3 【jExpression-4】jB 7 26 6 60 1 29 15 52 11 56 8 20 11 31 8 47 7 52 6 33 11 55 9 22 3 71 17 6 1 31 22 44 2 54 18 22 21 47 4 26 1 40 23 34</pre> | <p>$x = .> :i.3\ 3$</p> |
|---|---------------------------------------|

| | |
|---|--|
| <p>変数項目行列の 1 列目にオール 1 を加える</p>  <p>Input 対話型Expression</p> <pre>【jExpression-4】x=. :1&, : x 1 7 26 6 60 1 1 29 15 52 1 11 56 8 20 1 11 31 8 47 1 7 52 6 33 1 11 55 9 22 1 3 71 17 6 1 1 31 22 44 1 2 54 18 22 1 21 47 4 26 1 1 40 23 34 1 11 66 9 12</pre> | <p>$X = . :1&, : x$</p> |
|---|--|

| | |
|---|-------------------|
| <p>重回帰分析の推定パラメータ $(\widehat{X'X})\beta = X'Y$ の計算は以下</p> | |
| <p>$(X'X)\hat{\beta} = X'Y$この式を整理すると、</p> $\hat{\beta} = \frac{Y}{X} = \begin{bmatrix} \beta_0 = 37.870 \\ \beta_1 = 1.770 \\ \beta_2 = 0.784 \\ \beta_3 = 0.316 \\ \beta_4 = 0.103 \end{bmatrix}$ | <p>Bhat=.Y%.X</p> |



2017/03/11

Mail: rtsi@live.jp

JAPLA : Toribe Rentaro