

2016.7.30.SAT

GUIでらくらくRデータ解析

株式会社ef-prime

鈴木 了太

自己紹介

■ 鈴木 了太

- 株式会社ef-prime代表
- Rユーザー歴15年くらい
 - ・ 開発CRANパッケージ: pvclust
 - ・ 統数研のRユーザー会やUseR!にも出没しています

■ 株式会社ef-prime(エフプライム)

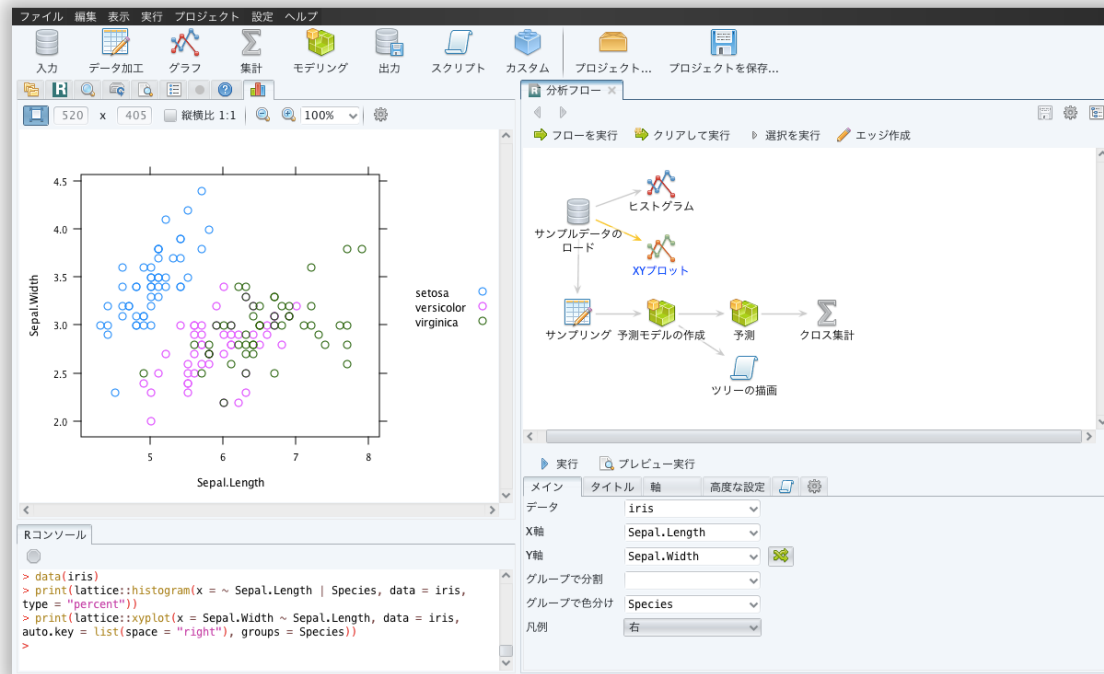
- 2006年3月設立
- データ分析コンサルティング(主に企業向け)
- 多くのプロジェクトでRを活用

GUIでらくらくRデータ解析

R AnalyticFlow

■ データ解析のためのR GUI

- Windows / Mac / Linux対応
- ef-prime社が開発、オープンソースで公開



R AnalyticFlowの特徴

R AnalyticFlowの特徴

■ マウスでかんたん操作

- プレビューをみながらRスクリプトを自動生成



The screenshot shows the R AnalyticFlow web interface. At the top, there are three tabs: 'メイン' (Main), '詳細設定' (Detailed Settings), and '高度な設定' (Advanced Settings). The 'メイン' tab is active. Below the tabs, there are input fields for '入力ファイル (file)' (Input File) with the value 'iris.txt', '出力オブジェクト' (Output Object) with the value 'iris', and 'ファイル形式' (File Format) set to '自動判定' (Automatic Detection). There is a checkbox for 'ファイル変更時に自動更新' (Automatic update when file changes) which is checked. Below these are dropdown menus for '区切り文字 (sep)' (Delimiter) set to ' " " (スペース)' (Space), 'ヘッダー (header)' (Header) set to 'TRUE', and '文字コード (fileEncoding)' (File Encoding) set to 'UTF-8'. At the bottom of the settings section, there are icons for 'プレビュー' (Preview), a document icon, and a folder icon. Below the settings, there are four radio buttons for 'テーブル' (Table), 'summary', 'print', and 'str', with 'テーブル' selected. A table with 5 rows and 6 columns is displayed, showing the first 5 rows of the iris dataset. The columns are 'Sepal.Length', 'Sepal.Width', 'Petal.Length', 'Petal.Width', and 'Species'. The 'Species' column has a dropdown menu showing 'setosa'. At the bottom of the interface, there are three buttons: '実行' (Execute), '+ ドラッグして追加' (Drag to add), and '× 閉じる' (Close).

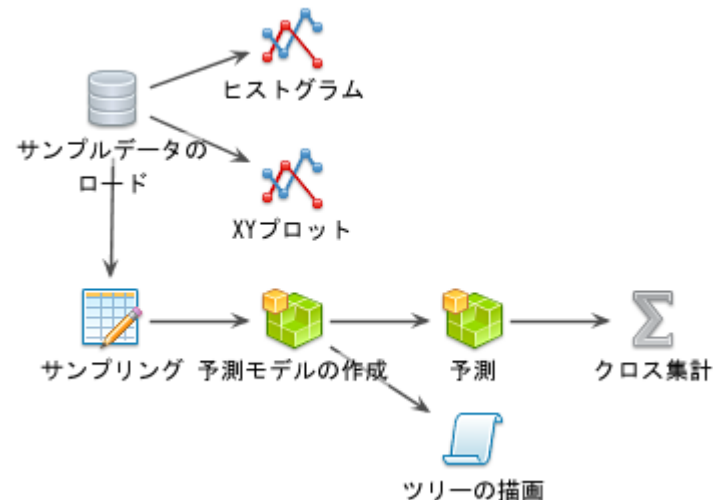
	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.8	1.4	0.2	setosa



```
> iris <- read.table(file = "iris.txt", header = TRUE, sep = " ", fileEncoding = "UTF-8",  
quote = "\"", stringsAsFactors = FALSE, comment.char = "#", na.strings = "")
```

R AnalyticFlowの特徴

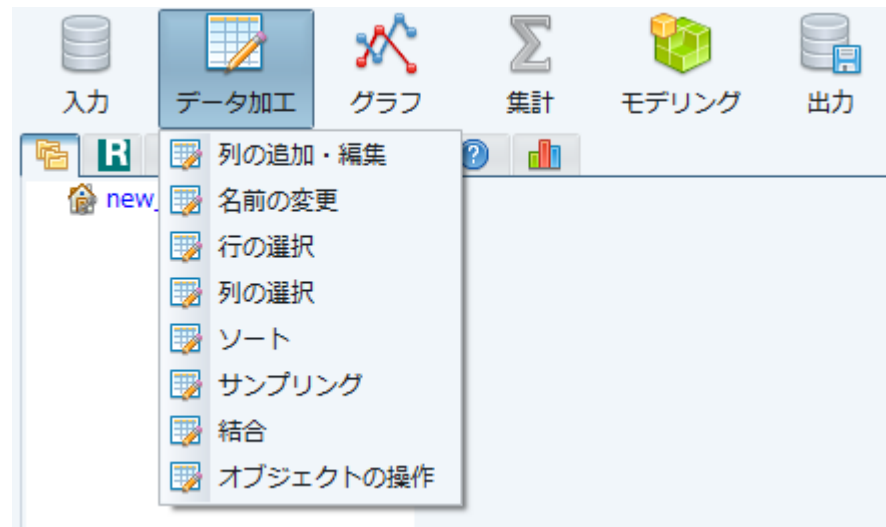
- 分析プロセスをフローチャート形式で整理
 - 作成した処理を順番に実行
 - 見目で処理の流れがわかり、簡単に再現できる



R AnalyticFlowの特徴

■ R初心者最適

- 文法や関数の名前を知らなくても利用できる
- 生成されたスクリプトを参考に学習することも可能



R AnalyticFlowの特徴

■ 分析作業をシンプルに

- 定型化された処理はオプションを選ぶだけで完結
- 単純ミスが起こりにくい
 - ・ 引数名のスペルミスが起こりにくい
 - ・ 同じ文字列を何度も記載しないといけないケースでも、GUI上の入力は一箇所だけ

A screenshot of the R AnalyticFlow web interface. The interface is divided into two main sections: '入力データ' (Input Data) and 'モデル設定' (Model Settings). In the '入力データ' section, 'iris' is selected for '入力データ' and 'iris.lm' for '出力オブジェクト'. The 'モデル式' (Model Formula) section has three radio buttons: 'GUI' (selected), '式オブジェクト', and '直接入力'. Below this, '目的変数 (Y)' is set to 'Sepal.Length'. For '説明変数 (X)', the 'すべて' (all) radio button is selected. The '定数項を含める' (include constant term) checkbox is checked. In the 'モデル設定' section, the '手法' (method) is '線形回帰' (linear regression). The 'ステップワイズ法' (stepwise method) checkbox is checked. The '探索方向' (search direction) is set to '増加ステップワイズ' (increasing stepwise). There is also an unchecked checkbox for '以下を常に含める:' (always include the following:).

入力データ: iris

出力オブジェクト: iris.lm

モデル式

- ☒ GUI
- ☐ 式オブジェクト
- ☐ 直接入力

目的変数 (Y): Sepal.Length

説明変数 (X): ☒ すべて ☐ 含める ☐ 除外する

定数項を含める: ☒

モデル設定

手法: 線形回帰

ステップワイズ法: ☒

探索方向: 増加ステップワイズ

☐ 以下を常に含める:



```
> iris.lm <- step(lm(formula = Sepal.Length ~ 1, data = iris), direction = "both",  
scope = terms(Sepal.Length ~ ., data = iris))
```

R AnalyticFlowの特徴

■ 複数ユーザーでの共有

- フローチャート形式で流れをイメージしやすい
- 「処理の記述」と「設定」を分離できる
- R上級者向けの機能も豊富
 - ・ 複雑な処理をスクリプトで記述
 - ・ 自作カスタムGUIを作成



```
*新規フロー ▶ # 予測モデルの作成...
▶ 実行 ▶ 選択を実行

1 ## 予測モデルの作成と確認
2
3 # データの読み込み
4 data(iris)
5
6 # サンプリング
7 local({
8   x <- iri
9   smpl <- iris iris.mean iris ... 00, replace = FALSE)
10  assign(x, Ctrl+Spaceで一覧を表示), , drop = FALSE], pos = parent.frame(n = 2))
11  assign(x = "test", value = x[-smpl, , drop = FALSE], pos = parent.frame(n = 2))
12 })
13
14 model <- rpart::rpart(formula = Species ~ ., data = train)
```

はじめてのR AnalyticFlow

ソフトウェアの入手

■ ダウンロード

- ウェブサイトにアクセス

<http://r.analyticflow.com>

- ダウンロードの方法
 - ・ ウェブサイトメニューの「ダウンロード」をクリック
 - ・ お使いのシステムに適したファイルをダウンロード



Windows 64bit
R AnalyticFlow 3.0.5

(Windows 64bitの場合)

インストールと起動

■ 準備

- OS共通: Rをインストール
- Macの場合: rJavaパッケージをインストール
 - ・ `install.packages("rJava")`
- Linuxの場合
 - ・ Oracle JDKのインストールなどの準備(ドキュメント参照)
 - ・ 文字化けする場合、デスクトップ環境のフォント設定を(明示的に)VLゴシックなど日本語フォントに設定するのがラク
 - ＞ Gnome Tweak Toolなど



※詳細はダウンロードページ記載の「スタートガイド ドキュメント」を参照

インストールと起動

■ Windows

- インストーラ(.exe)を実行
- R AnalyticFlowアイコンをダブルクリック

■ Mac

- 圧縮ファイル(.zip)を展開
- アプリケーション(.app)を実行

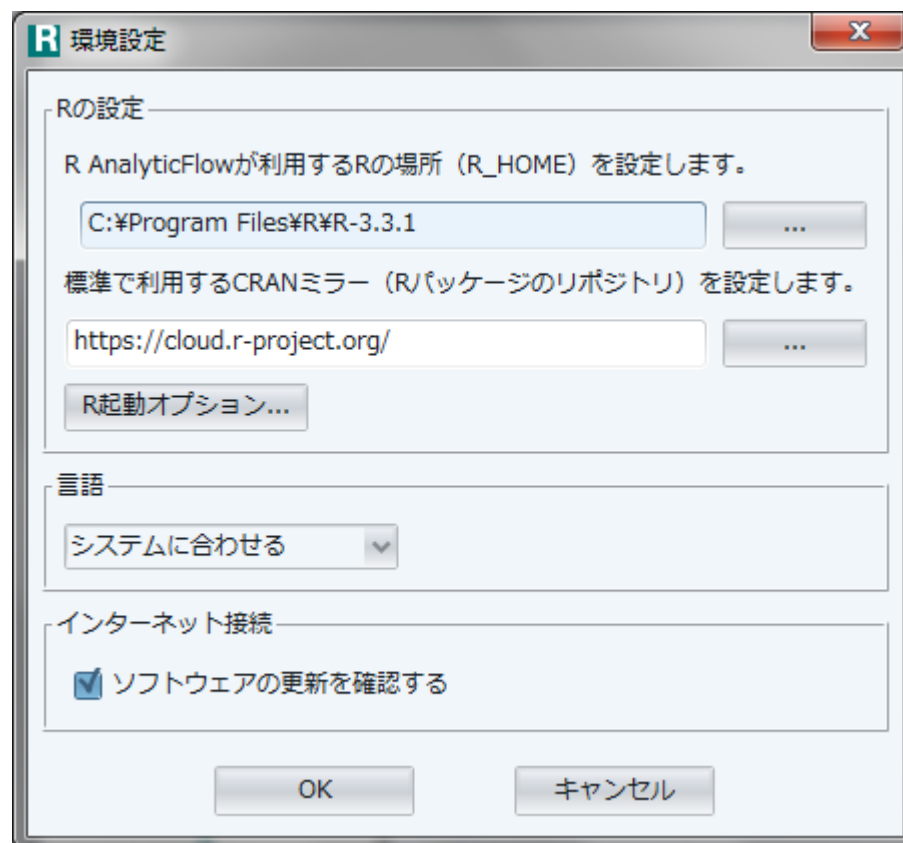
■ Linux

- 圧縮ファイル(tar.gz)を展開
- 起動スクリプト rflow を実行
 - ・ ./rflow

※詳細はダウンロードページ記載の「スタートガイド ドキュメント」を参照

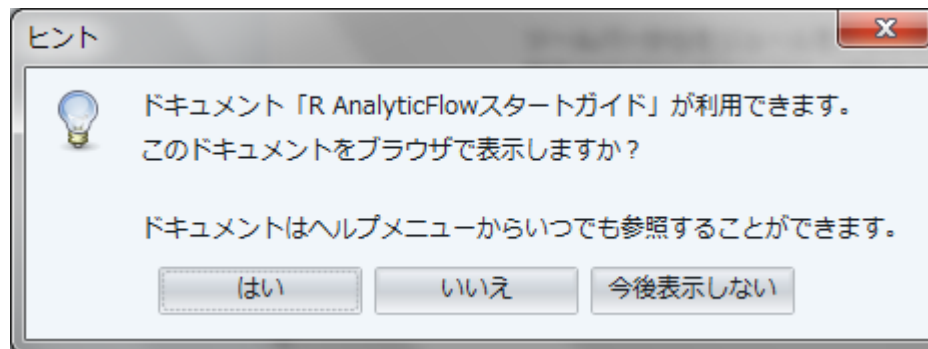
初期設定

「Rの設定」はWindowsの場合のみ。Mac/Linuxではシステム標準のR（Rコマンドで呼び出されるもの）が利用される



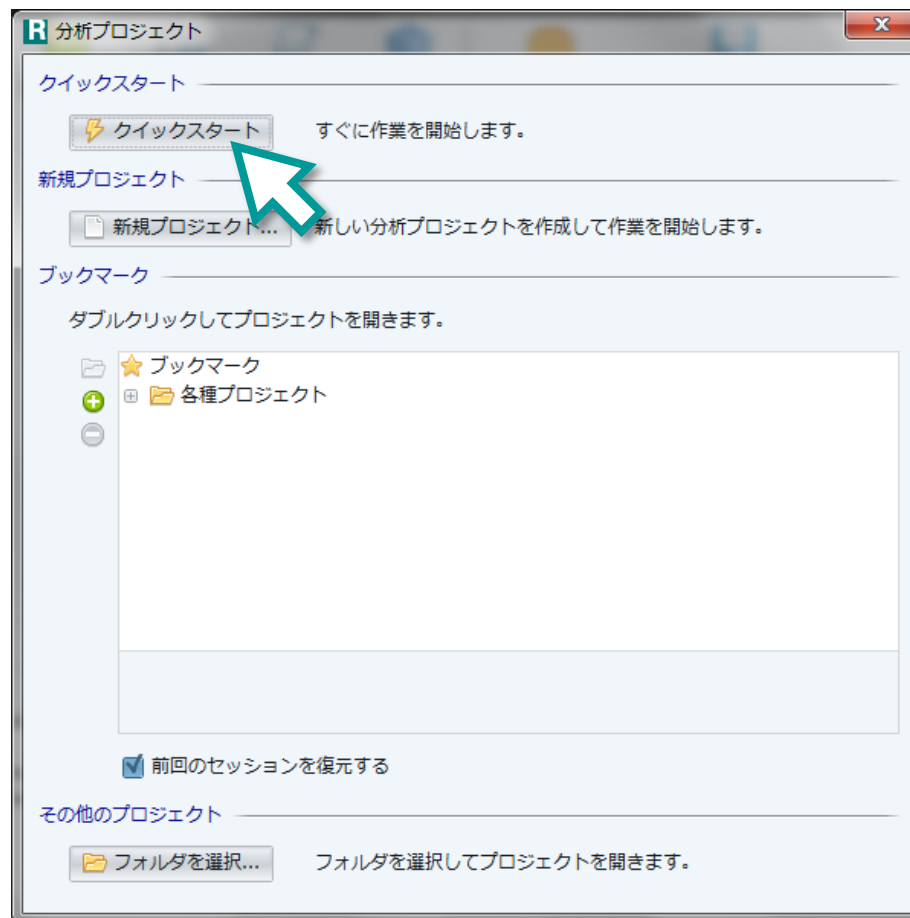
ドキュメントの表示

入門ドキュメントをブラウザに表示することが可能
(「今後表示しない」をクリックするとこのダイアログは表示されなくなります)



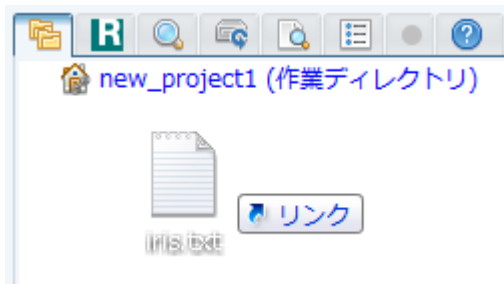
プロジェクトダイアログ

既存の分析プロジェクトがある場合はここで選択。
今回は「クイックスタート」をクリックして一時フォルダで作業



データの表示

作業ディレクトリの外にファイルをドロップするとプロジェクトで利用できる。
(ディレクトリにドロップするとファイルの移動またはコピー)
ファイルを選択すると、区切り文字などを自動判定してテーブル表示



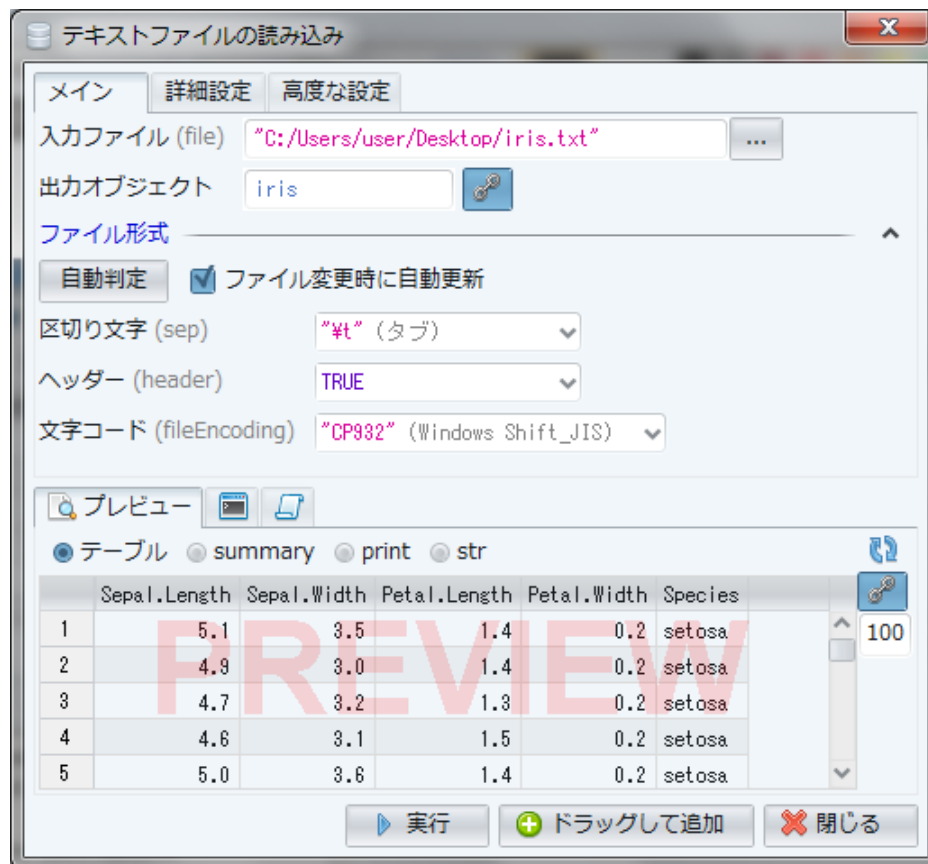
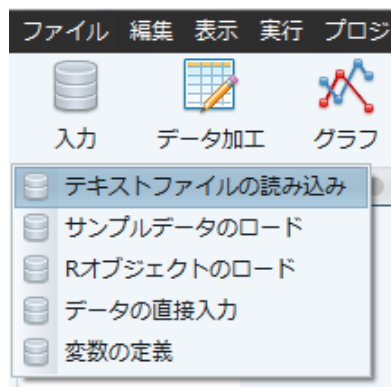
	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5	3.6	1.4	0.2	setosa

※サンプルデータとして、irisデータをタブ区切りテキストに出力したものを使用

```
> write.table(x = iris, file = "iris.txt", sep = "¥t", row.names = FALSE, col.names = TRUE)
```

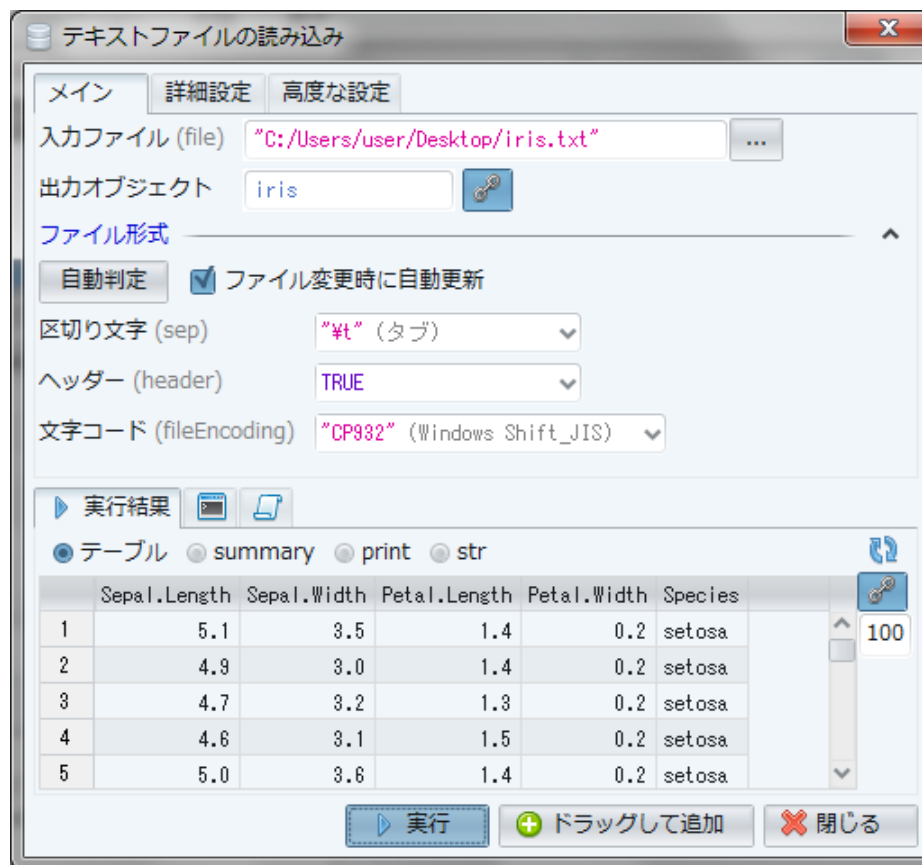
テキストファイルの読み込み

ファイルを選択し、ツールバーの「入力」から「テキストファイルの読み込み」をクリック。
読み込み設定は自動判定され、必要に応じて変更する。
データの一部(ここでは最初の100行)を使った実行結果のプレビューが表示される。



実行

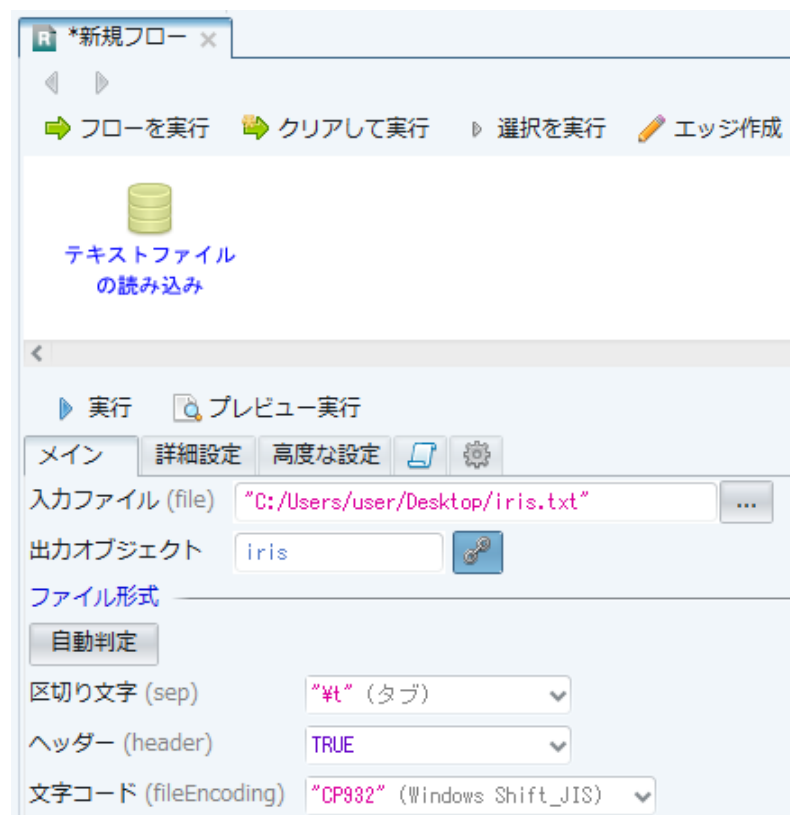
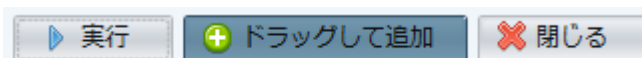
実行ボタンを押すと、対応するRスクリプトが実行されて結果が表示される



```
> iris <- read.table(file = "C:/Users/user/Desktop/iris.txt", header = TRUE, sep = "¥",  
fileEncoding = "CP932", quote = "¥", stringsAsFactors = FALSE, comment.char = "", na.strings = "")
```

分析フローに追加

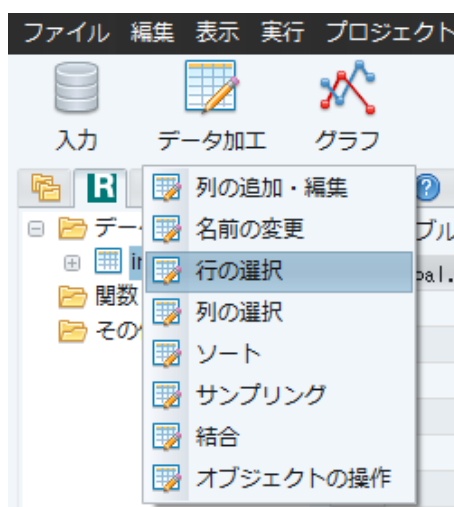
「ドラッグして追加」ボタンをクリックして分析フローにドラッグ & ドロップする。
作成した処理がフロー上のノード(アイコン)として設置される



行の選択

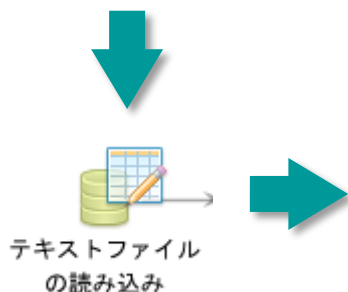
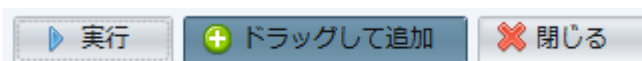
irisデータオブジェクトを選択、ツールバーの「データ加工」から「行の選択」をクリック。

 ボタンで「フォーム」を選び、条件を入力。実行して結果を確認



分析フローに追加

「ドラッグして追加」を分析フローにドラッグ。
先ほど作成したテキストファイルの読み込みノードの上にドロップすると
ノードがエッジ(矢印)で接続され、フローが作成される



→ フローを実行 → クリアして実行 ▶ 選択を実行 ✎ エッジ作成



実行 プレビュー実行

メイン 高度な設定 設定

入力 iris

出力 iris

タイプ ☒ 含める ☐ 除外する

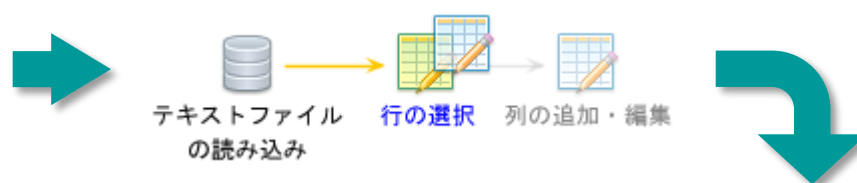
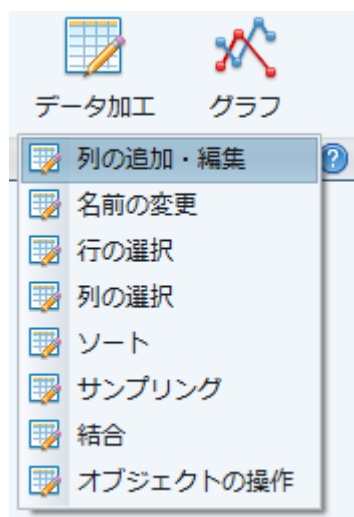
組み合わせ ☒ すべてを満たす ☐ いずれかを満たす

条件

+ Sepal.Length >= (以上) 5

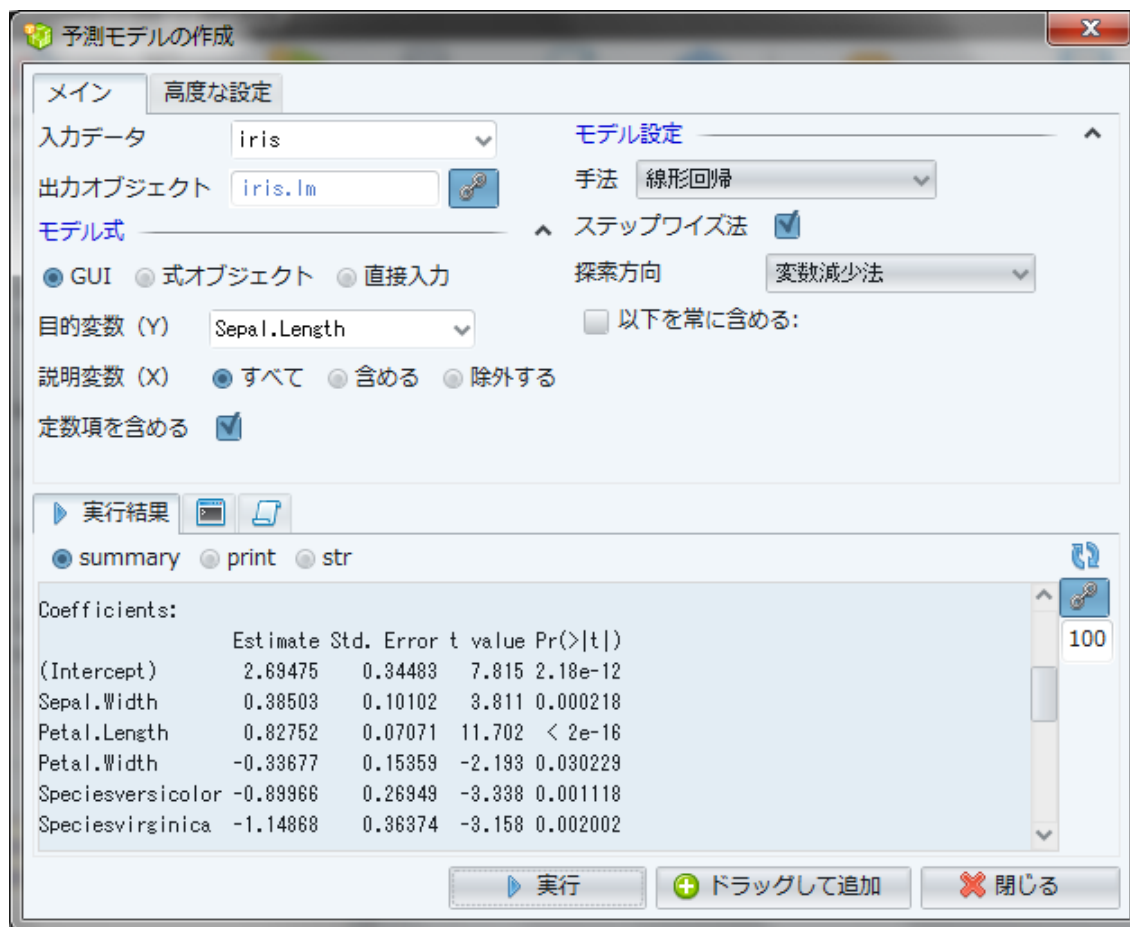
直接フローに追加する

ツールバーから処理を選んで、フローに直接ドラッグ＆ドロップすることでノードを追加することも可能。対話実行が必要ない場合はこちらでも



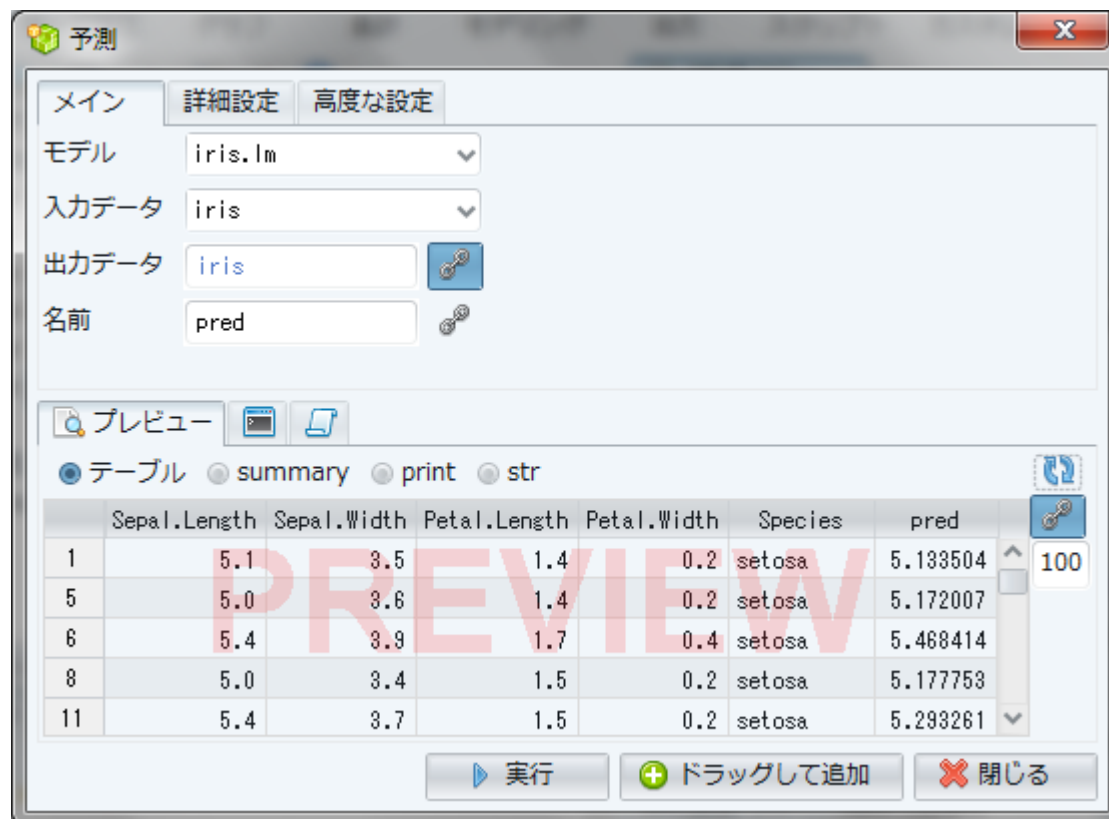
モデリング

そのほか様々な機能を利用することができる。「モデリング」から「予測モデルの作成」を用いて回帰分析モデルなどを作成可能



予測

「モデリング」の「予測」を用いて、作成したモデルを用いた予測が可能。
以下では元の iris データオブジェクトに pred という名前で予測値を格納



予測

メイン 詳細設定 高度な設定

モデル iris.lm

入力データ iris

出力データ iris

名前 pred

プレビュー

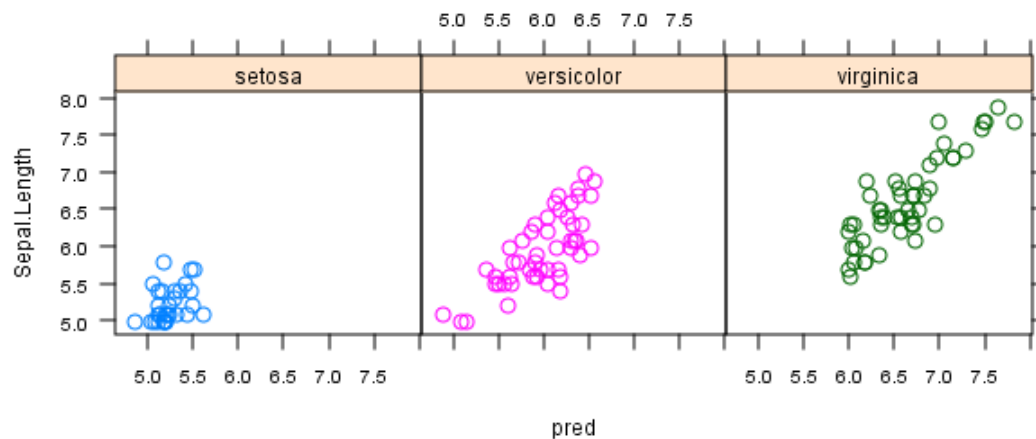
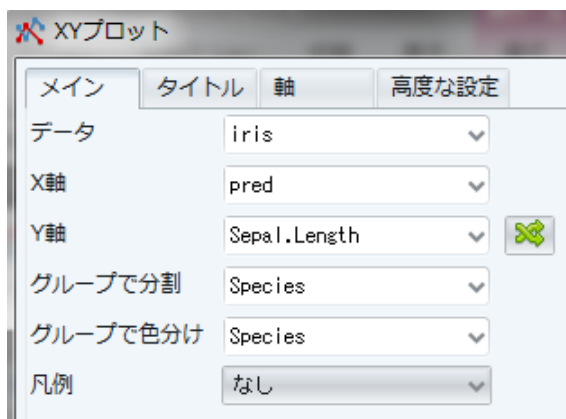
テーブル summary print str

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species	pred
1	5.1	3.5	1.4	0.2	setosa	5.133504
5	5.0	3.6	1.4	0.2	setosa	5.172007
6	5.4	3.9	1.7	0.4	setosa	5.468414
8	5.0	3.4	1.5	0.2	setosa	5.177753
11	5.4	3.7	1.5	0.2	setosa	5.293261

実行 ドラッグして追加 閉じる

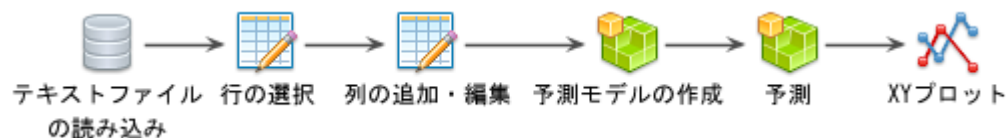
プロット

「プロット」の「XYプロット」で2軸のプロットが可能。
ここでは予測値をX軸、実際の値をY軸としてグループごとに散布図を作成



分析フローの完成

処理を追加していくことで分析フローが完成



※参考: 対応するRスクリプト(一部省略)

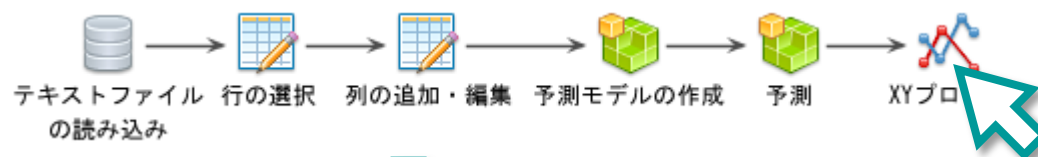
```

1 iris <- read.table(file = "C:/Users/user/Desktop/iris.txt", header = TRUE, sep = "#t", fileEncoding = "CP932",
2   quote = "\"", stringsAsFactors = FALSE, comment.char = "", na.strings = "")
3
4 iris <- subset(x = iris, subset = Sepal.Length >= 5)
5
6 iris <- local({
7   x <- iris
8   y <- iris
9   y[["dummy"]] <- with(x, Species == "setosa")
10  y
11 })
12
13 iris.lm <- step(lm(formula = Sepal.Length ~ ., data = iris), direction = "backward")
14
15 iris$pred <- (function(model, data, predType, type, ...){
16   }
17 )(model = iris.lm, data = iris, predType = "prob")
18
19 print(lattice::xyplot(x = Sepal.Length ~ pred | Species, data = iris, auto.key = FALSE, groups = Species))

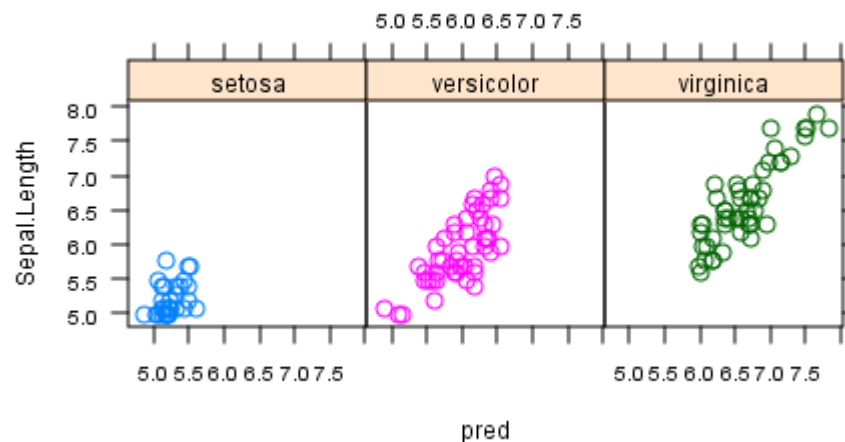
```

分析フローの実行

作成したフローのうち、実行したい最後のノードをクリック。
 「フローを実行」を選択すると、処理が順番に実行される。
 「クリアして実行」の場合はオブジェクトを一度すべて消去して実行する

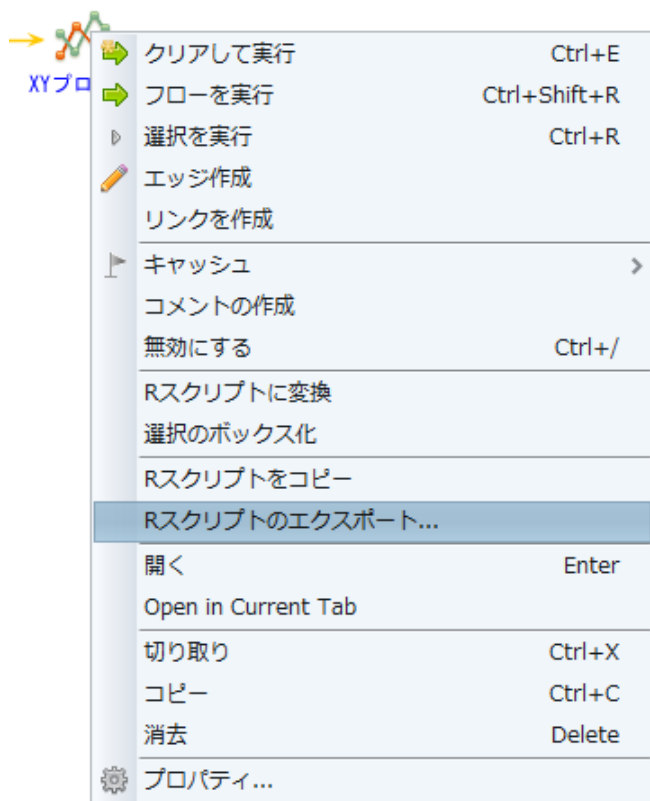


→ フローを実行 → クリアして実行 ▶ 選択を実行



Rスクリプトのエクスポート

ノードを実行する代わりに、右クリックメニュー(コンテキストメニュー)から「Rスクリプトのエクスポート...」を選択すると、実行されるスクリプトをRスクリプトファイルに出力することができる



```

1 iris <- read.table(file = "C:/Users/user/Desktop/iris.txt",
2   header = TRUE, sep = "¥t", fileEncoding = "CP932", quote = "¥"
3   , stringsAsFactors = FALSE, comment.char = "", na.strings = "")
4
5 iris <- subset(x = iris, subset = Sepal.Length >= 5)
6
7 iris <- local({
8   x <- iris
9   y <- iris
10  y[["dummy"]] <- with(x, Species == "setosa")
11  y
12 })
13
14 iris.lm <- step(lm(formula = Sepal.Length ~ ., data = iris),
15   direction = "backward")
16
17 iris$pred <- (function(model, data, predType, type, ...){
18   }(model = iris.lm, data = iris, predType = "prob")
19 )
20
21 print(lattice::xyplot(x = Sepal.Length ~ pred | Species, data =
22   iris, auto.key = FALSE, groups = Species))

```

ご清聴ありがとうございました



<http://r.analyticflow.com>



R AnalyticFlow



@efprime_jp

最新情報: <http://r.analyticflow.com/news-tips/>