

OpenCAE勉強会@岐阜  
2015/10/31

# PythonとCAEツール

OpenCAE勉強会

SH

# 本日の発表内容

- 背景
- Sfepy
- GetFEM++
- Pythonの最適化・統計分析ツール
- まとめ

# 背景

- 9月のOpenCAE勉強会@関東にてPython関連CAEソフトの紹介があったのでこちらの勉強会にも展開したいと思います。
- 最適化ツールや結果分析に必要な統計分析も最近ほとんどPythonのライブラリでできるのであわせて紹介する。

# *SfePy*

- SfePyはチェコ共和国のRobert Cimrmanという人が開発されている有限要素法モジュール、前回藤田さんが紹介。
- サイト <http://sfepy.org/doc-devel/index.html>
- 接触問題は簡単なものしか解けない
- IGA(アイソジオメトリック解析)があるらしい？
- Gmshのファイルが読み込みできる

# *SfePy* その2 詳しくは藤田 さんの資料を参照

- インストール方法 メモ(Ubuntu)
- `$ sudo apt-get install python-scipy python-matplotlib python-tables python-pyparsing libsuitesparse-dev python-setuptools mayavi2 python-dev ipython python-sympy cython python-sparse`
- `$ git clone git://github.com/sfepy/sfepy.git`  
( コマンド実行時Directoryに展開されるので実行場所は注意  
~ ホーム直下で実行するなどが無難 )
- `$ cd sfepy`
- `$ python setup.py build_ext --inplace`
- `$ apt-get install mercurial`
- `$ hg clone https://bitbucket.org/dalcinl/igakit`
- `$ cd igakit`
- `$ python setup.py install`

こちらはアイソジ  
オメトリック解析の  
モジュール

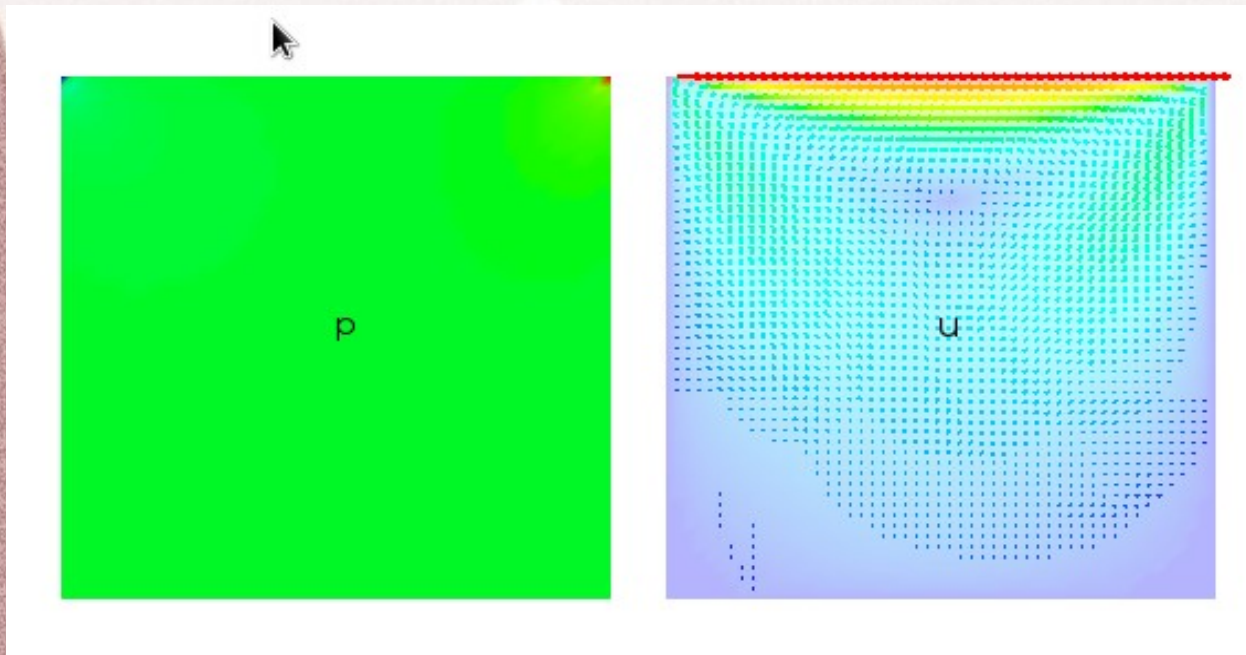
# *SfePy* その3

- コマンド実行

```
$ ~/sfepy/simple.py navier_stokes2d.py
```

- 結果処理 ( vtkができるのでparaViewで見れるが )

```
$ ~/sfepy/postproc.py user_block.vtk
```



# *GetFem++*

- 有限要素法のモジュールでPythonやScilab, Matlab から使うことができる
- <http://download.gna.org/getfem/html/homepage/index.html>
- 前回の関東勉強会で小山さんが紹介
- 接触や亀裂進展解析、連成解析など複雑な現象を解析できるらしい

# *GetFem++* その2

- インストール方法 メモ(Ubuntu)詳しくは下記に小山さんがWindowsへのインストールモジュールを準備してくれているので、それを使う
- [https://github.com/tkoyama010/getfem\\_presentation](https://github.com/tkoyama010/getfem_presentation)
- スーパーユーザになる (`$ sudo su`)
- `$ apt-get install python-getfem++ python-pip paraview`
- `$ pip install notebook`
- `$ easy_install jsonschema`
- `$ easy_install functools32`
- `$ apt-get install gms`
  
- 使い方 : `ipython notebook` を ノートブックのあるディレクトリで起動する。
- つづきは小山さんの準備したデモを見る



# Python統計分析ツール データ結果可視化・統計分析ツール

- Pythonのライブラリには統計分析や最近流行りの機械学習に有効なものが多く、CAEの最適化解析などの結果分析にも有効です。
- 市販のCAEむけ最適化ツールの機能を事前に紹介します

# CAE商用最適化ツール (PIDOツール)

- 商用の最適化ツールは最適設計支援ツール (以下PIDOツール : Process Integration & Design Optimization) と呼ばれて基本機能はCAEソフトの自動実行・ツール間統合・最適化アルゴリズム・近似モデル作成(応答曲面)、結果データ分析(可視化)などが行える。
- 代表的なソフトにmodeFrontier(IDAJ社), Isight(ダッソー社), Optimus(サイバーネット社)などがある。

ツール間の連携手順  
データ(変数,目的関数)定義、パラメータ変更ルール(実験計画法)定義(GUI)

CAEツール自動実行  
(パラメータ自動変更)  
結果自動収集

結果分析と  
可視化(グラフ化等)

実験計画法

応答曲面法

-最適解  
-相関分析  
-設計ルール抽出

最適化アルゴリズム

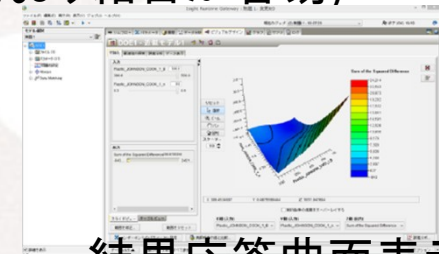
# 市販CAE最適化ツールの例① (ISIGHTの例)

特徴:

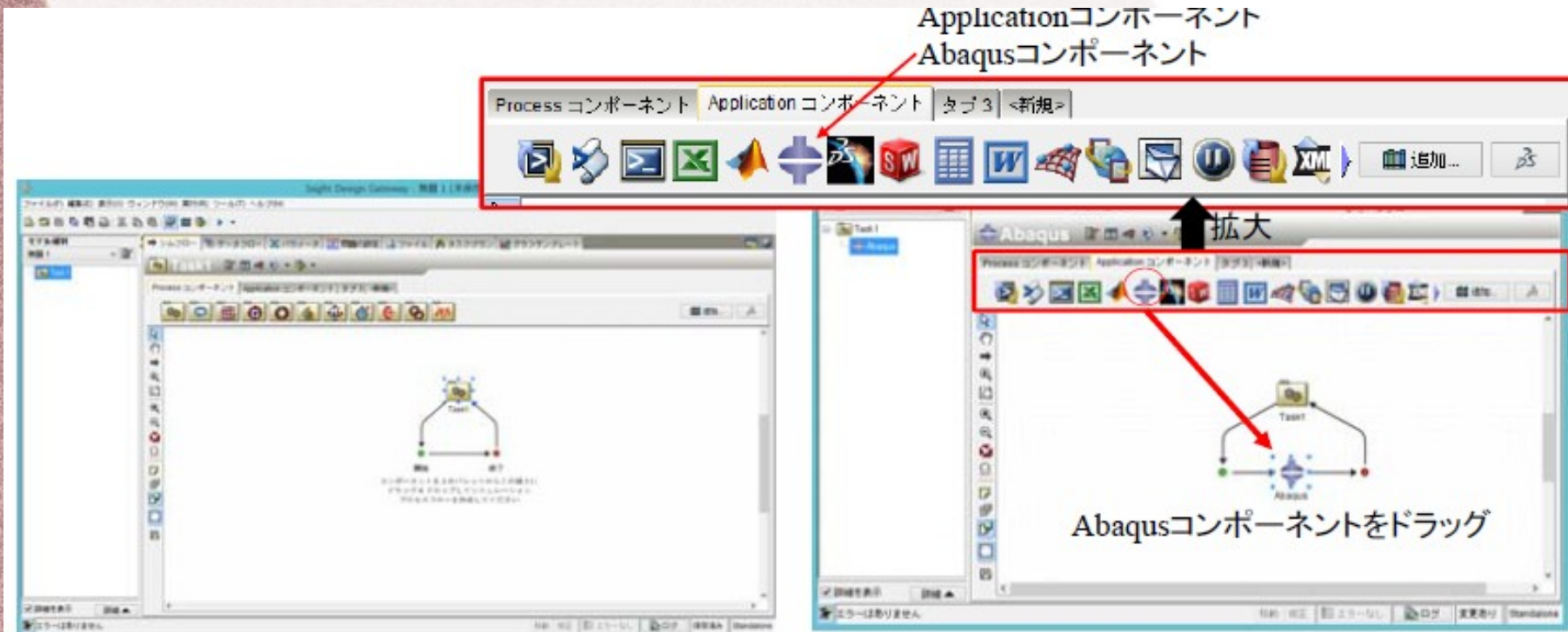
- ・設計で利用されるCAD, CAE, 独自ツール, Excel など表計算、統計ツール etc. と連携
- ・インターフェースが公開されており、ツール統合をGUI画面上で比較的容易に短時間でできる(Excel, ABAQUSなどは最初からコンポーネント化されており結合が容易)

→ GUIによるツール間連携設定

- ・最適化アルゴリズム(古典的、ヒューリスティックetc.)
- ・近似モデル(多項式, Kriging, RBF)
- ・解析結果、設計変数、目的関数関係の可視化・分析

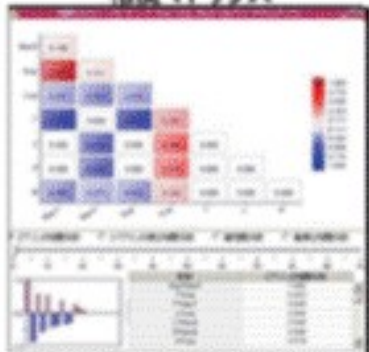


結果応答曲面表示



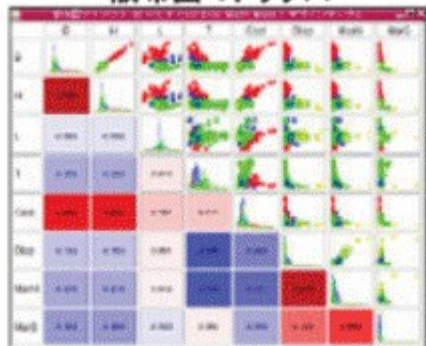
# 市販CAE最適化ツールの例② (modeFrontierの例)

相関マトリクス



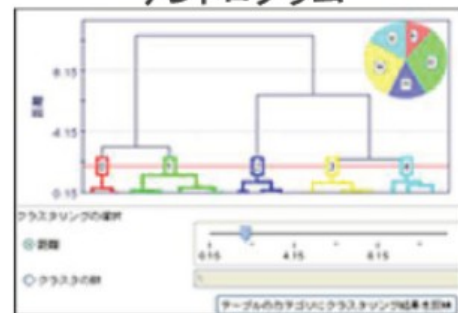
各2変数間の4種の相関係数をマトリクス表示  
相関係数値の大きい2変数をランキング表示

散布図マトリクス



散布図において色によるカテゴリ表示が可能  
拡大縮小表示により詳細を把握

デンドログラム



クラスタ散布図

直感的でわかりやすい日本語GUI

一般的な実験計画法から独自手法まで

豊富なダイレクトインターフェースを標準搭載

他社の追随を許さない最先端の探索アルゴリズム

分散実行環境構築が容易

多目的ロバスト設計による同時満足化

実験データを活用した仮想最適化

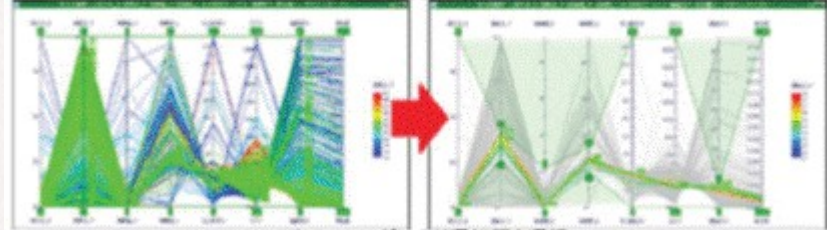
50以上の結果処理機能、多変量解析機能

豊富な応答曲面手法

設計の原理原則抽出支援機能(CAP)

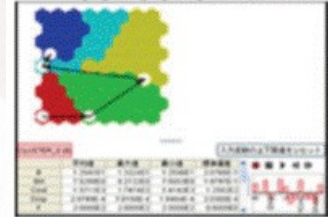
基本モジュール / Runtimeモジュール  
Advance / Standard / Basic

多次元解析チャート

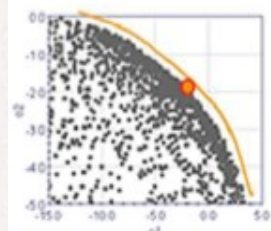


スクリーニングにより選好解を選択  
デザイン分布からデザインの特徴を把握

クラスタリングSOM



クラスタ多次元解析チャート



パレート最適解

Isight, Optimus などと同様機能の他  
- 多目的最適化機能(MOGA等, パレート最適解), 結果分析(マイニング機能)

# 市販ツールとOSSツール比較

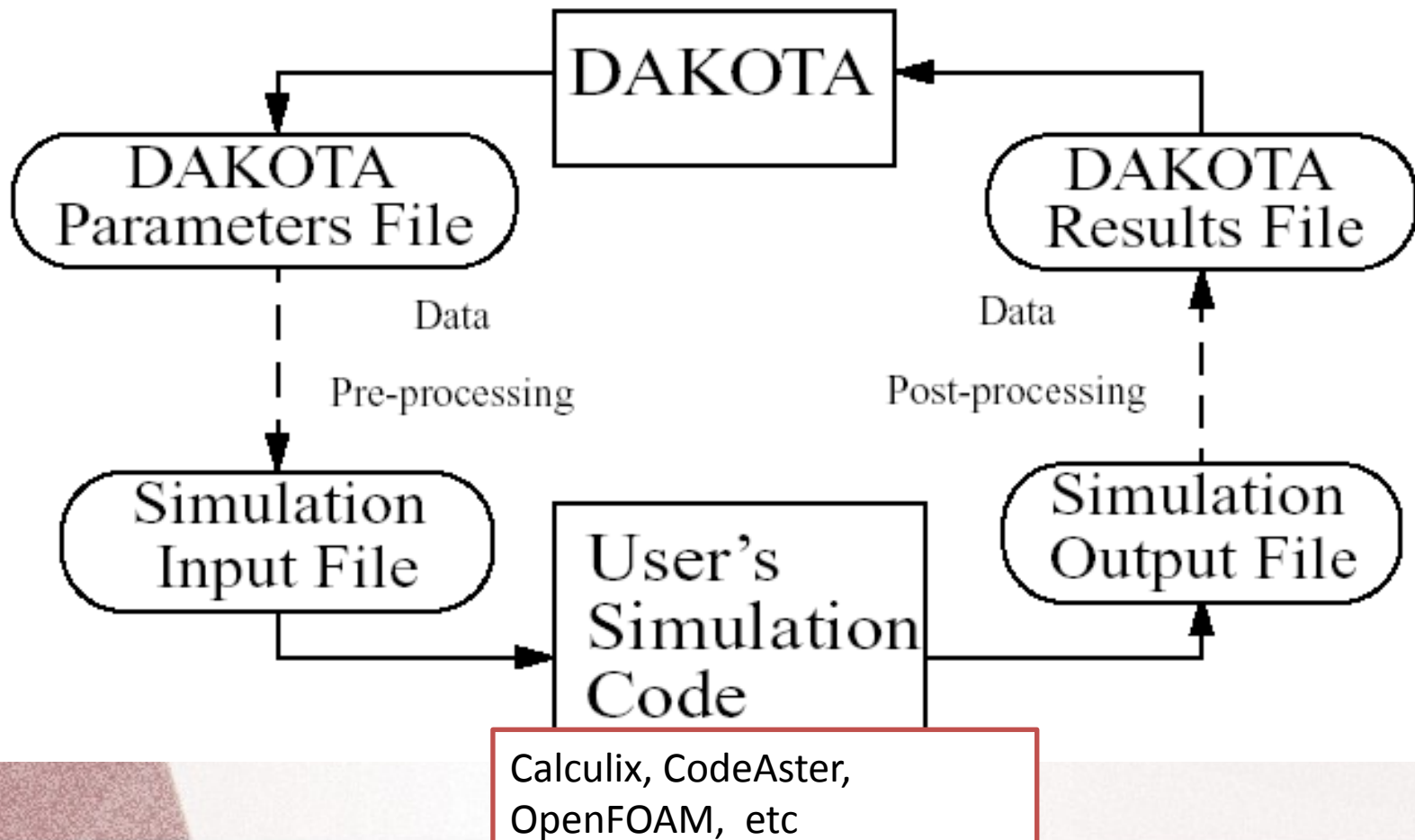
	商用ソフト (modeFrontier, Isight, OPTIMUS)	OSS Dakota	OSS OpenMDAO	備考
サンプリング 実験計画法	○	○	○	OSSでも結構いける 直交表(OA), ラテン超方格子(LH)、CCD etc.
最適化アルゴリズム	○	○	○	OSSでも結構いける 数理計画, GAなど
近似手法	○	○	○	OSSでも結構いける 多項式近似, Krigingなど
GUIによるツール連携	○	×	△(そこそこ使える が。。)	OpenMDAOは商用に近い DakotaはGUI設定画面なし
モデル形状の自動変 更	×	×	×	モデル形状の自動変更は商用ツールで も無理 → 別の形状作成編集ツール要
結果分析	○	×	×	OSSでは結果分析が苦手

# Dakotaとは①？

自動的に解析に入力するパラメータを変更したパラメータSTUDYや目的となる指標(目的関数)を与えると自動的にそのような目的の値になるようなパラメータを見つけてくれるツールです(最適化ツール)

- ParaViewと同じ Sandia National Laboで開発されている

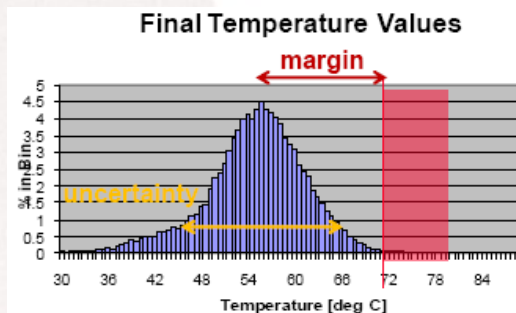
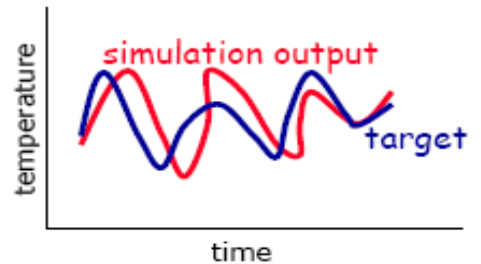
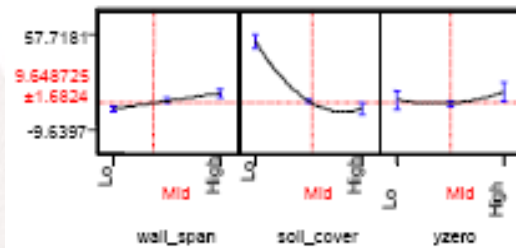
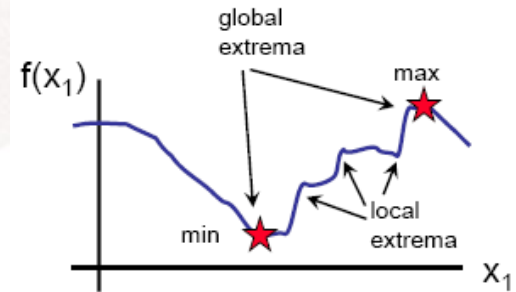
- <http://dakota.sandia.gov/>



# Dakotaとは②？

• DAKOTAは具体的には何に使えるのか？

- Optimization 設計最適化：  
最適寸法や最適パラメータ決定
- Sensitivity Analysis 感度解析：  
入力パラメータ変動に対する出力感度
- Calibration パラメータ同定：  
未知パラメータ決定 (物性値同定など)
- Uncertainty Quantification 信頼度：  
入力パラメータが変動した場合の応答  
変動の確率分布推定



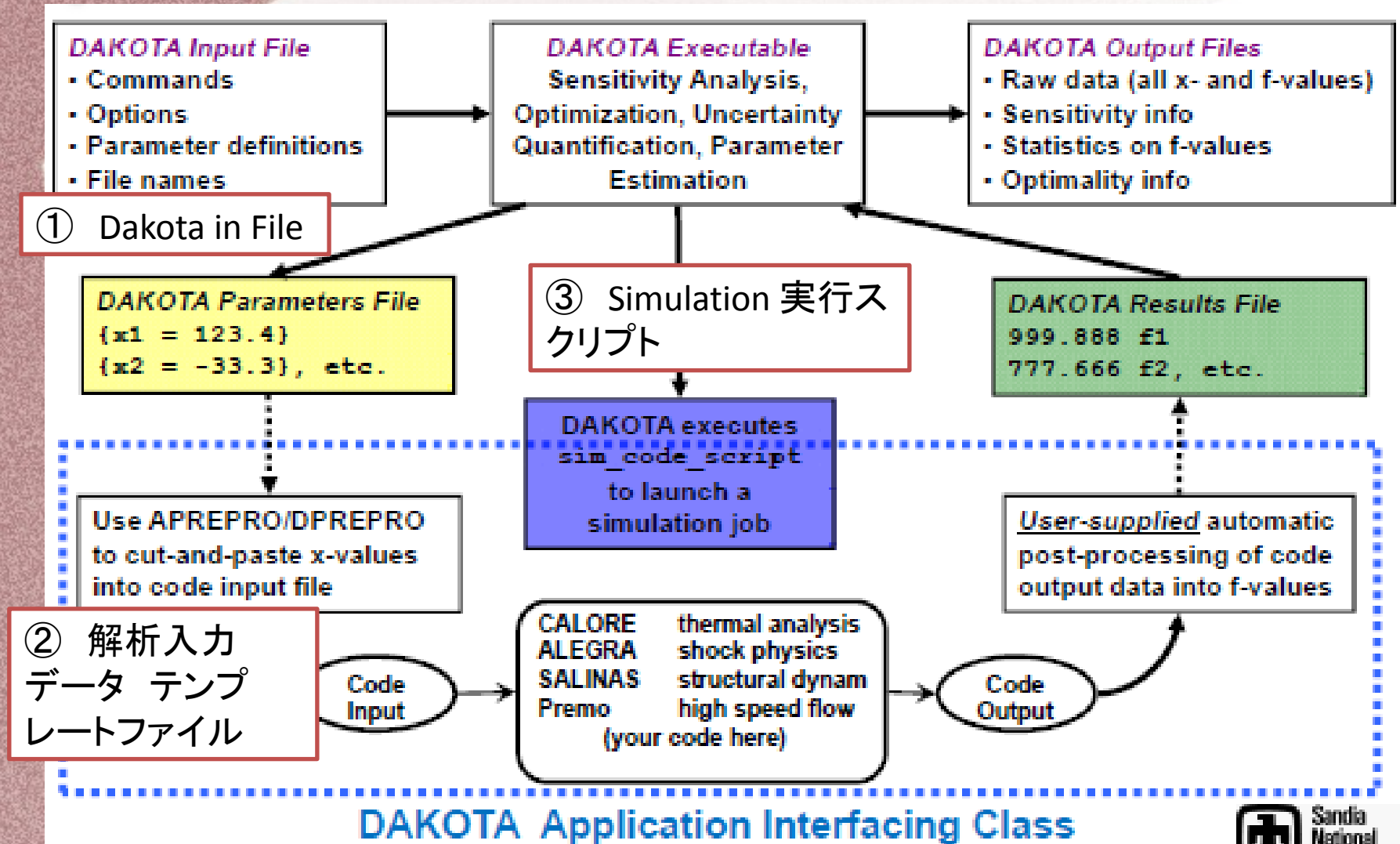
# Dakota と各ツールとの連携①

- Dakotaは任意のプログラムや解析ツールと連成する場合、スクリプトなど外部インターフェースを使って自動実行させる。
- 幾つかのスクリプトやbinary 連携ツールが examples Directory 以下に提供されているのでこれを使って連携実行させる compiled interface 以下にはmatlab 以外はつかえそうもなく script\_interfaces の下のサンプルを利用することにした ( [V5.3からMatlab以外にOSS scalibの直接interface が準備された](#))。
- ABAQUS, NASTRANなど解析ツールの他、Python script, Excel Visual Basic script と 連携する例のsample がある。



# Dakota と各ツールとの連携②

- Dakotaと解析ツールは下記のように連携ユーザは以下3つファイルを準備する必要あり。



# Dakota参考資料

① マニュアル類: User Manual , Reference Manual, Theory Manual, Developers Manual 4種類マニュアルがある。いずれも Sandia national laboratory のホームページからDownload 可能.

<http://dakota.sandia.gov>

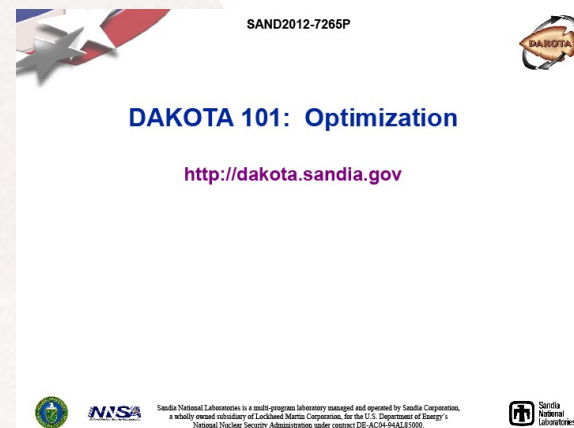
初心者は User's Manual, Chapter 2: Tutorialから読む

●Dakota の教育コースをSandia Labo. で実施している模様でこの Presentation 資料もホームページで公開されている。Optimization (最適化), Calibration(パラメータチューニング), Uncertain Qualification(信頼性) などがある。マニュアルよりは分かりやすい。

●Support(連絡先):

–dakota-users@software.sandia.gov

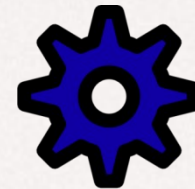
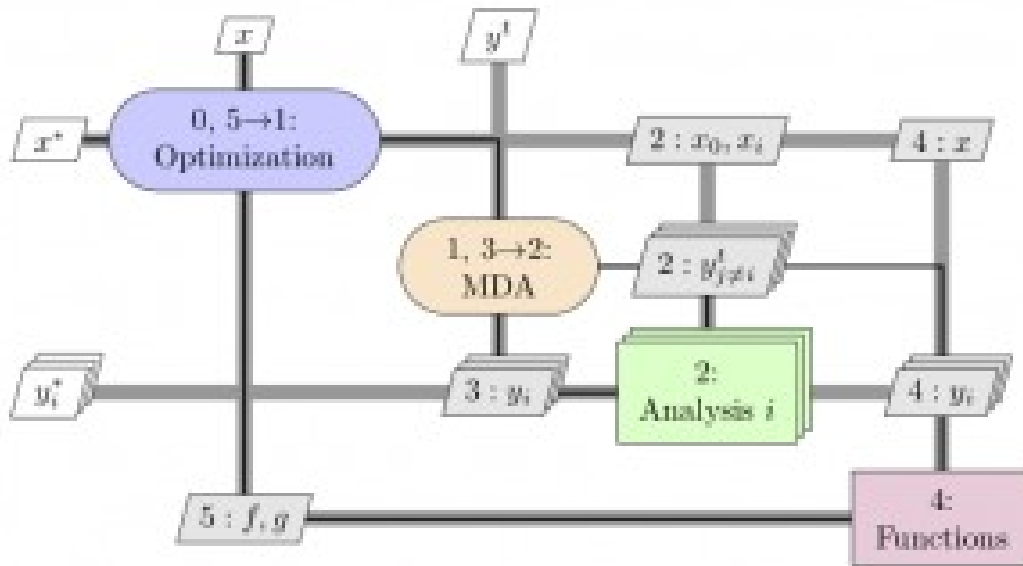
–dakota-help@sandia.gov



# OpenMDAOとは？

- ・いわゆる最適化ツール、自動的に解析に入力するパラメータを変更したパラメータSTUDYや目的となる指標(目的関数)を与えると自動的にそのような目的の値になるようなパラメータを見つけてくれるツール。Pythonで記述される。
  - ・2014/6/21のOpenCAE学会講習会にて片山さんが講習会を実施
  - ・片山さん資料に基づき自分のPC@Ubuntu12.4LTSにインストールしてみた
  - ・春日さんのホームページにもインストール方法が記載されている
- <http://openmdao.org/>

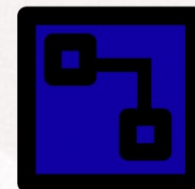
# open M D A O



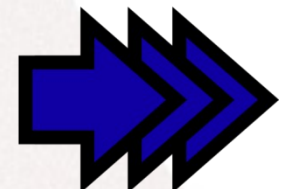
Component



Driver



Assembly



Workflow

# OpenMDAOインストールメモ①

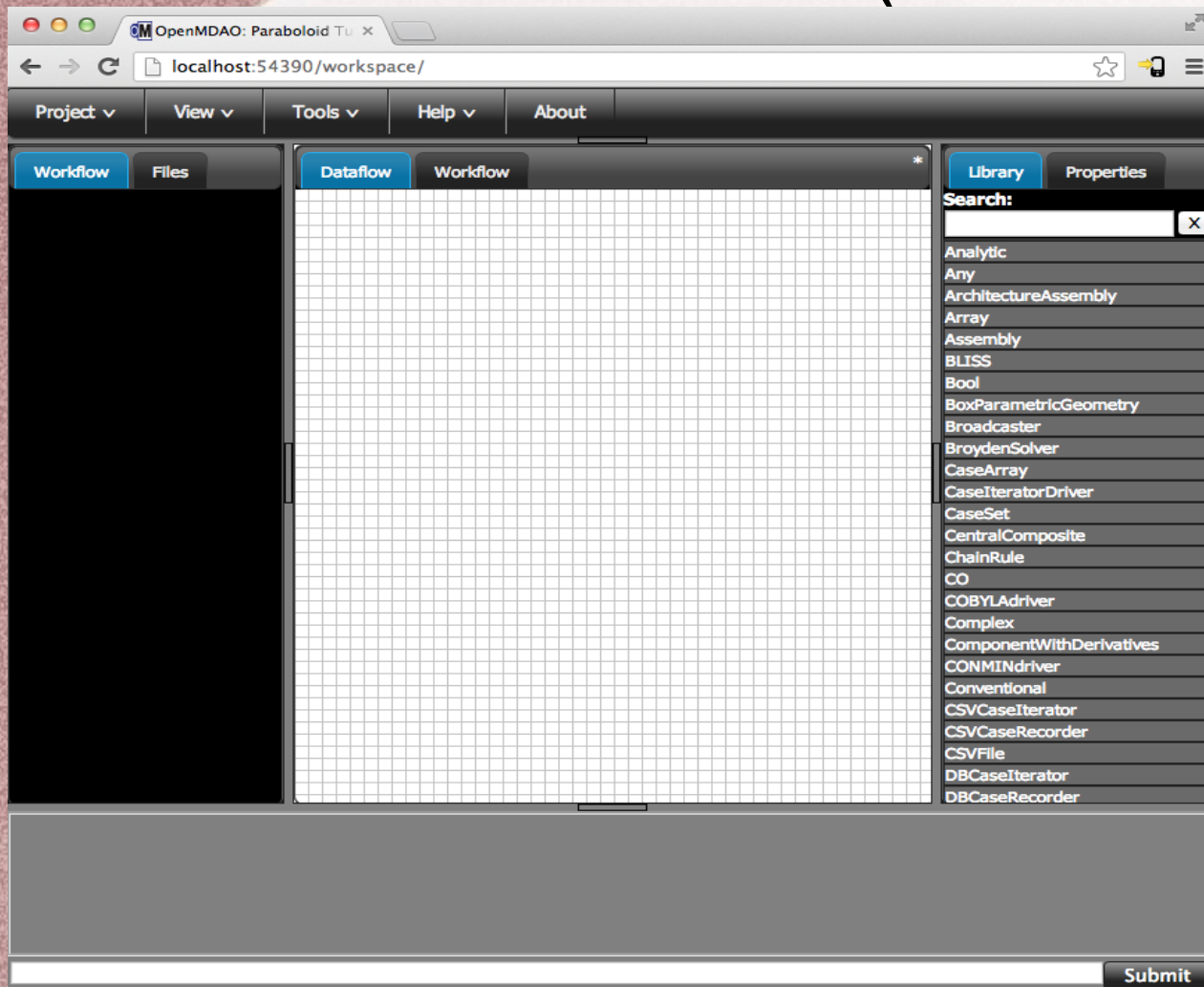
- インストールメモ
  - 基本的には片山さんが関西の勉強会  
2013/8/3 の資料にアップしているやりかたにて問題なし。
  - <http://openmdao.org/>
- go-openmdao.py を上記HOMEページからDownloadして、端末から実行するだけ  
`abc$ python go-openmdao.py`
- 上記 Python script は インストーラで本体は入っていない。ネットワークからダウンロードしつつインストールするので、ネット接続した状態で実行する必要がある。
- 私の環境では、python-scipy が無いよ！と怒られたので、これのみ追加インストール (人によって異なると思われる)  
`abc$ sudo apt-get install python-scipy`
- Python がInstall されていればWindowsにもそのままインストール可能な模様だが、未確認  
(Windowsインストールパッケージが公開されている模様)

# OpenMDAOテストメモ②

- `. bin/activate`  
にて仮想環境(virtualenv)に入る。  
この仮想環境はどういう意味があるのか不明であるが、とにかくそういうものらしい
- `(openmdao-0.9.5)dexcs@dexcs-laptop:~/openmdao-0.9.5$ openmdao test`  
プロンプトが変わって仮想環境に入ったことが確認できる。  
テストコマンドを実行  
(現在の最新版は0.9.7?)
- テストが無事終了すると  
“`Ran 813 tests in 124.134`”  
のメッセージが表示

# OpenMDAOテストメモ③

- openmdao gui  
にてGUIを起動する (ブラウザが起動)



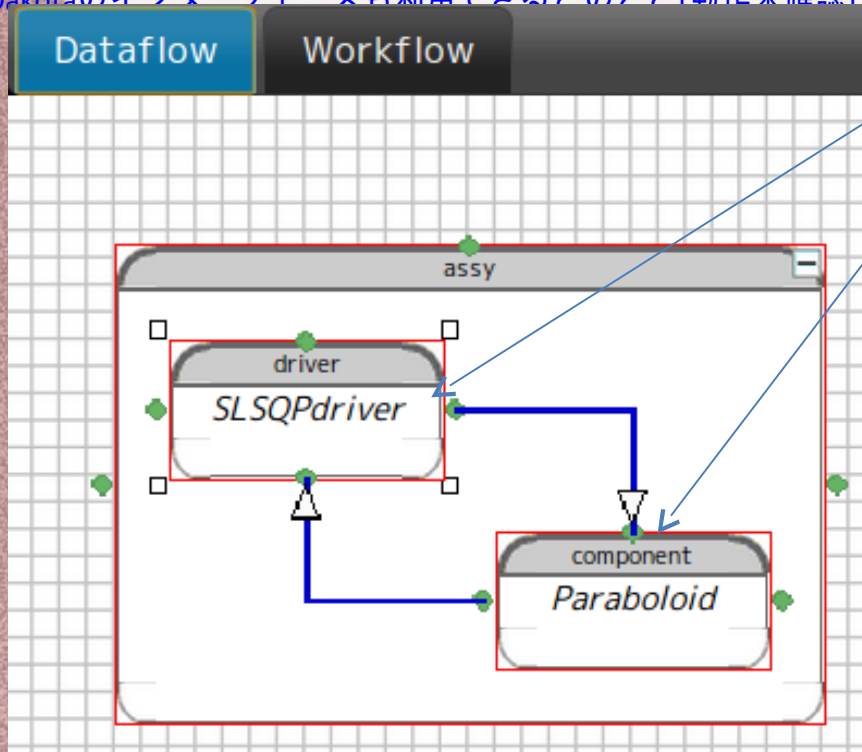
-Workflowにて動作手順  
が定義できる

Dakota(後述)よりは商用  
最適化ツールに近いイ  
メージになっている。

- Dakotaと比較してこの  
GUIが使いやすそうで  
売りなわけだが、**恐ろ  
しく不安定で良く落ち  
る (OpenCAEユーザに  
はおなじみの現象)**
- 上手く起動しないとき  
は一度killして再度プ  
ロセスを起動するとう  
まくいくことが多い

# OpenMDAOテストメモ④

- ワークフローをGUIで作成する
- Driver(実験計画法や最適化アルゴリズム(GA, 古典的数理計画法)はパラメータ変数をどのように変化させる指示を行う部分。ソフトに通常に準備されているものを使う
- パラメータ変数と目的関数はコンポーネントとして自分でPythonで作成するか、既存または他の人の作ったインターフェースプログラムを利用する
- OpenFOAMのインターフェースは片山さん自作?のWrapperプログラムがある  
Install 方法: ネット接続した状態でOpenMDAOの仮想環境に入り下記コマンドをコマンドラインから実行:  
lib/python2.7/site-packages/openfoam\_wrapper/openfoam\_wrapper.py がDownloadされる
- pip install git+https://github.com/TatsuyaKatayama/openfoam\_wrapper.gitDownloading/unpacking  
git+https://github.com/TatsuyaKatayama/openfoam\_wrapper.git
- Dakotaのインターフェースも利用できるとのこと(動作未確認)



**Driver部分(SLSQP:逐次最小2乗計画法  
(数理計画法の一種)最適化)**

**Component部分(Pythonプログラム↓)**

```
#!/bin/pythonfrom openmdao.main.api import  
Component  
from openmdao.lib.datatypes.api import  
Float (中略)  
x = self.x  
y = self.y  
self.f_xy = (x-3.0)**2 + x*y + (y+4.0)**2 - 3.0
```

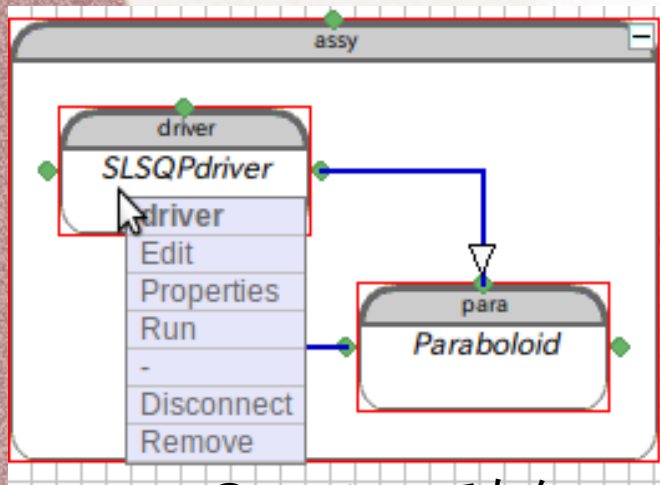
# OpenMDAOテストメモ⑤

## 簡易ベンチマーク結果



ToolのPlot Variableで  
“assy.para.f\_xy”と入力すると  
目的関数  $f(x,y)$  の値が  
グラフで出力される。

最適解の時の  $x, y, F(xy)$   
はParaboloidのWindow  
で左クリックEditで表示



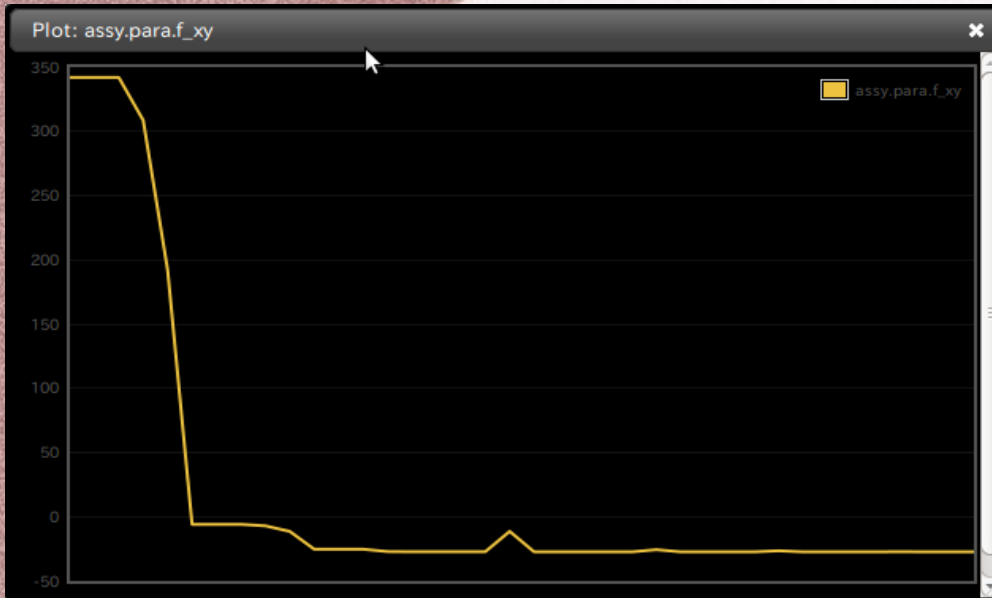
AssyのWindowsで左ク  
リック:Runを選択

	Name	Value	Units	Description
i	x	6.666666633...		The variable x
i	y	-7.333333366...		The variable y
i	directory			If non-blank, the directory to execute in.
i	force_exec...	False		If True, always execute even if all IO traits are ...
i	force_fd	False		If True, always finite difference this component.
i	missing_d...	error		Determines behavior when some analytical de...



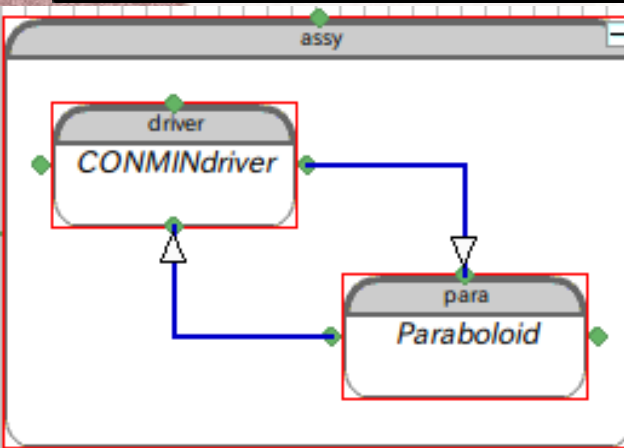
# OpenMDAOテストメモ⑥

- 数理計画法部分を別のアルゴリズムに入れ替えてみる。SLSQPをCONMINに置き換える(CONMINはCG勾配法のハズ?)



PLOT の収束履歴が変わる。  
CONMINの方が収束に時間が掛る。

最適解の時の $x, y, F(xy)$ はSLSQPの時と同じであることが確認できる



	Name	Value	Units	Description
i	x	6.66666643...		The variable x
i	y	-7.33333372...		The variable y
i	directory			If non-blank, the directory to execute in.
i	force_exec...	False		If True, always execute even if all IO traits are ...
i	force_fd	False		If True, always finite difference this component.
i	missing_d...	error		Determines behavior when some analytical de...

CONMINDriverをSLSQPのBOXに移動



CONMIN, SLSQP のように入れ替えても基本動作が変わらないDriverの場合、変数間の連結が維持される

# 無料のデータ結果可視化・統計分析ツール

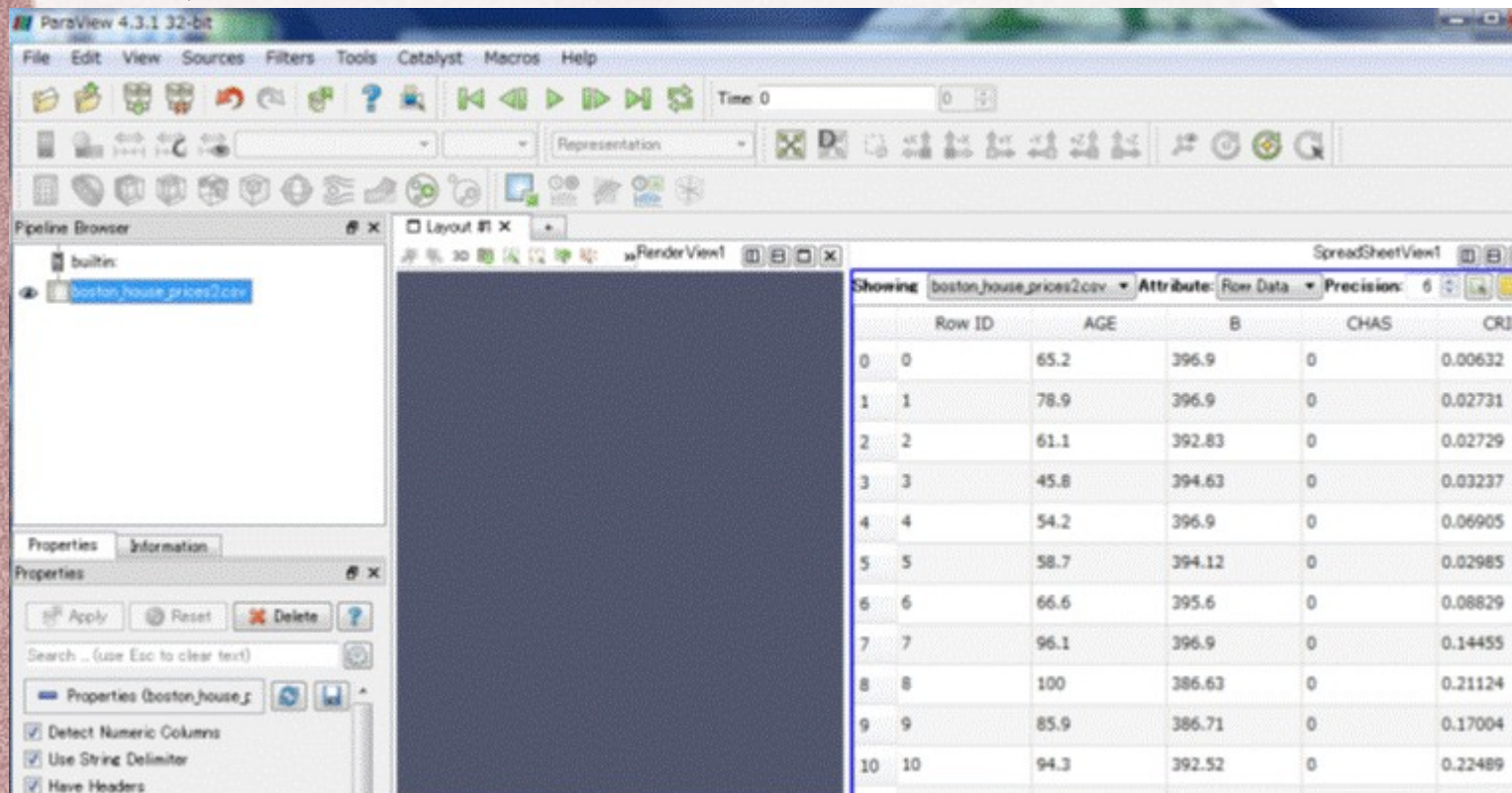
- 結果のグラフ化だけならParaViewでそこそこ商用最適化ツールのような表示が可能
- もう少し難しい統計分析（クラスタリングやデンドログラム、主成分分析 など）は統計分析用の専門ソフト（R または Pythonの統計分析関係ライブラリ群）を使うと商用ソフトとほぼ同等のことができる。
- 統計解析、データマイニング関連のフリーのライブライ（Python関連）の充実度は最近すさまじく、はっきり言って商用ソフト凌駕した？



商用の最適化ソフトの売りの一つは大量データの結果分析機能充実であったが、ここでの差はほとんどなくなったと考えられる

# 無料のデータ結果可視化・統計分析ツール ParaViewでの可視化例①

- ParaViewでCSV形式で出力したデータを読み込む。最初はテーブルがそのままこんな感じで表示される

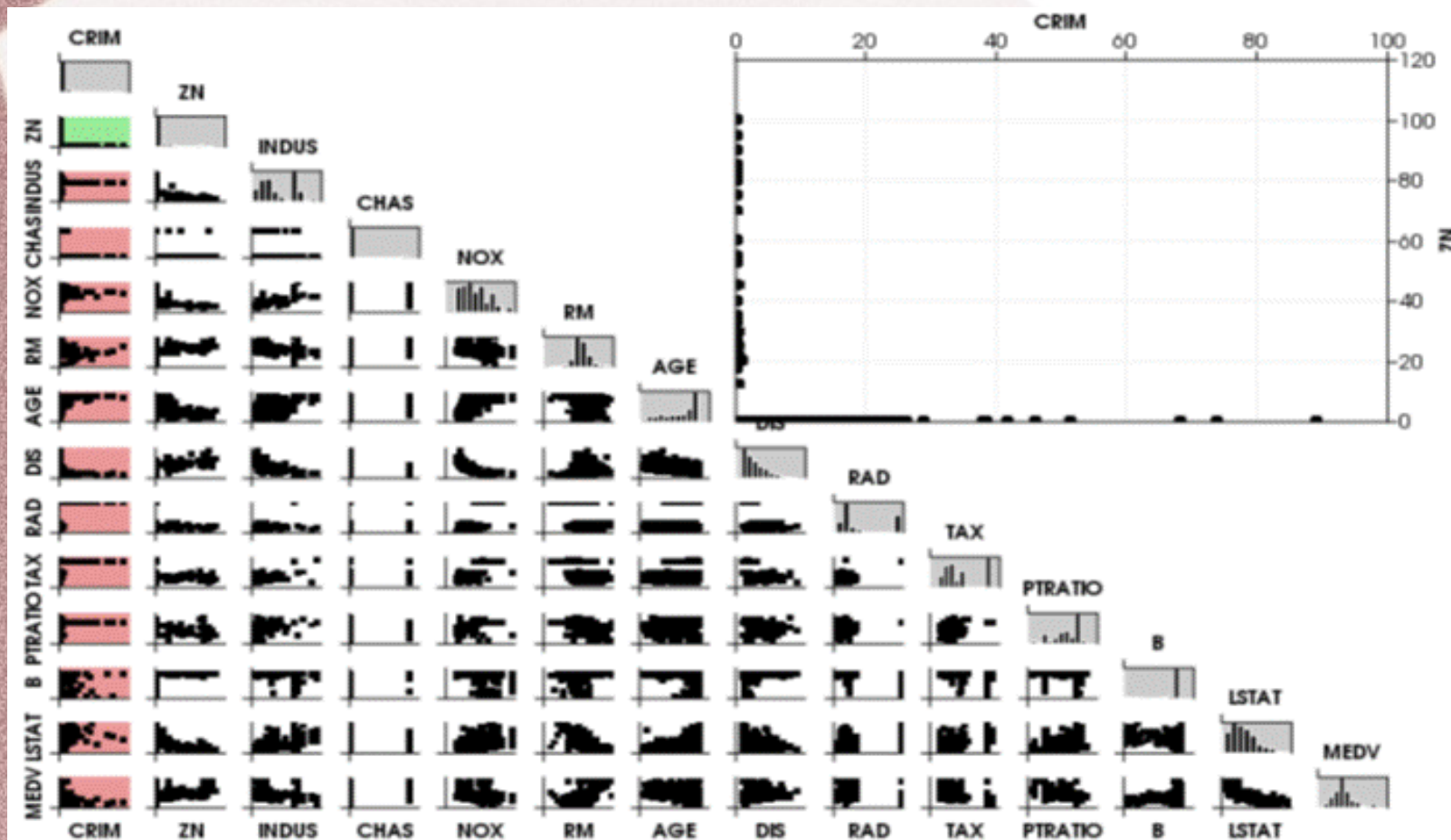


The screenshot shows the ParaView 4.3.1 32-bit interface. The main window displays a spreadsheet view of data from a CSV file named 'boston\_house\_prices2.csv'. The data is presented in a table with columns for Row ID, AGE, B, CHAS, and CRIM. The 'RenderView1' window is currently empty, showing a dark blue background.

Row ID	AGE	B	CHAS	CRIM
0	65.2	396.9	0	0.00632
1	78.9	396.9	0	0.02731
2	61.1	392.83	0	0.02729
3	45.8	394.63	0	0.03237
4	54.2	396.9	0	0.06905
5	58.7	394.12	0	0.02985
6	66.6	395.6	0	0.08829
7	96.1	396.9	0	0.14455
8	100	386.63	0	0.21124
9	85.9	386.71	0	0.17004
10	94.3	392.52	0	0.22489

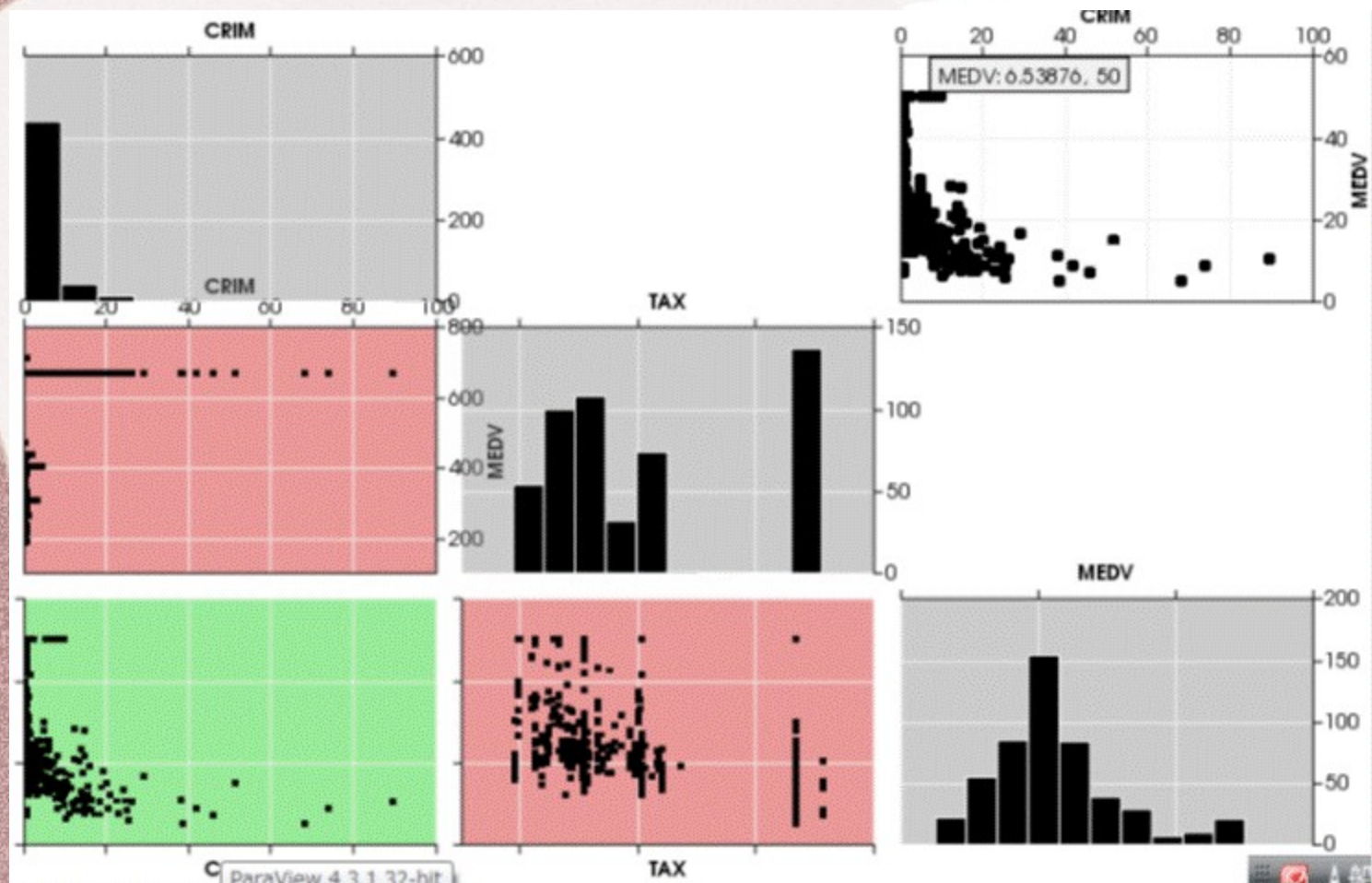
# 無料のデータ結果可視化・統計分析ツール ParaViewでの可視化例②

- 新規Layoutを選択し、Plot Matrix View を選択すると散布図マトリックスが表示される。



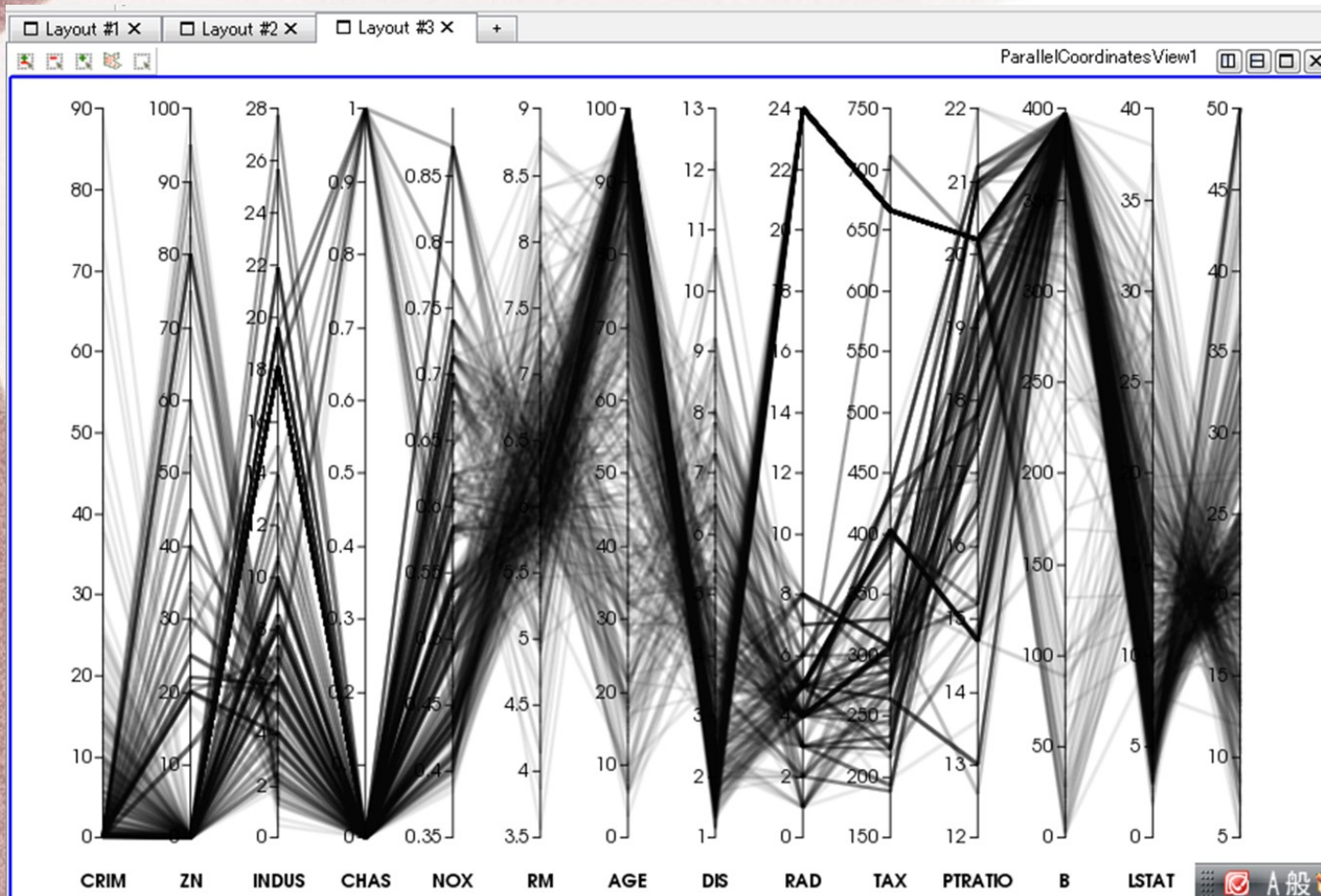
# 無料のデータ結果可視化・統計分析ツール ParaViewでの可視化例③

散布図マトリックスの変数をしぼりこんだ例



# 無料のデータ結果可視化・統計分析ツール ParaViewでの可視化例④

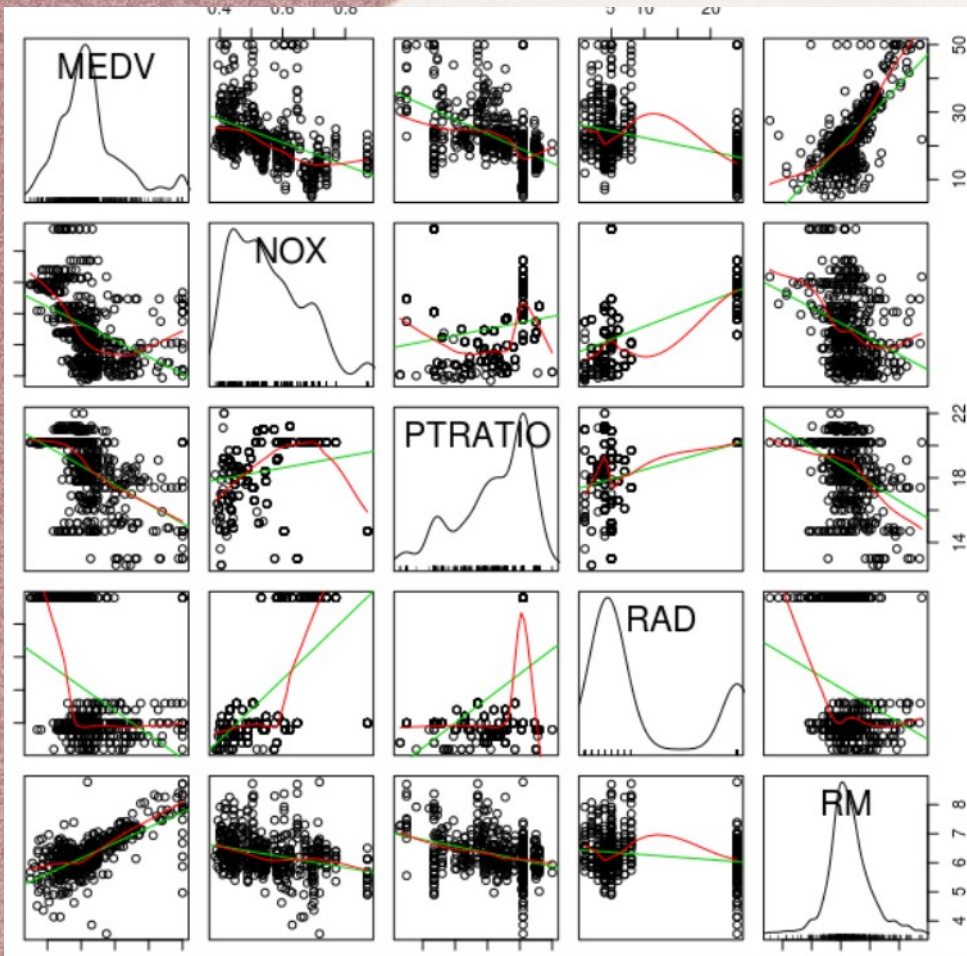
新規レイアウトにてParallel Coordinate Viewを選択する。modeFrontier の多次元レーダチャートのようなPlotを描くことが可能



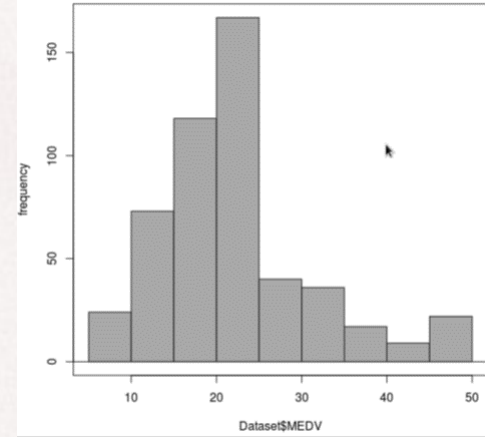
# 無料のデータ結果可視化・統計分析ツール

## 統計解析ソフトRによる分析①

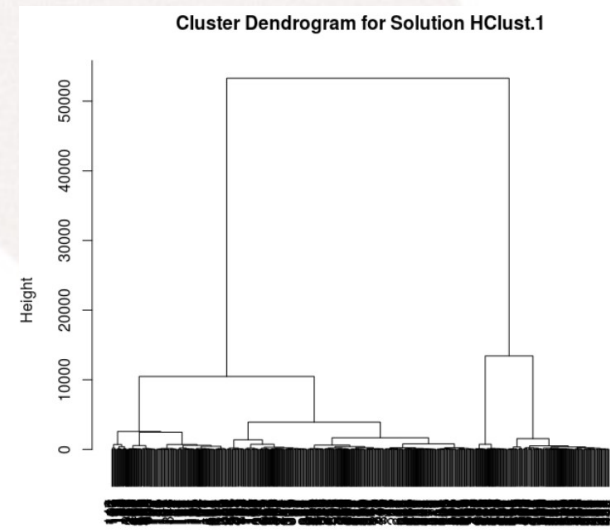
もう少し統計的な分析はRなどできる



散布図マトリックス



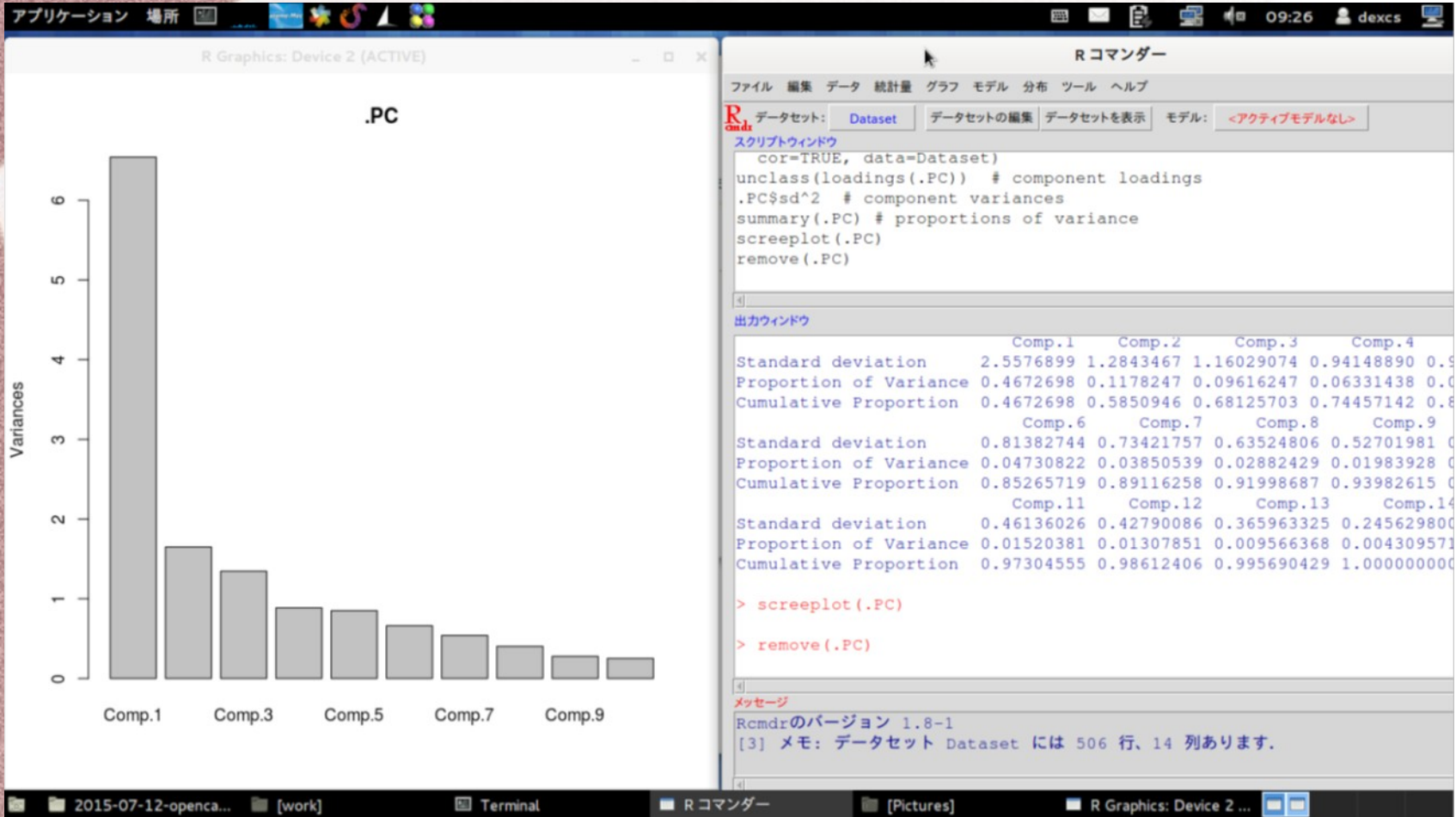
度数分布



デンドログラム

# 無料のデータ結果可視化・統計分析ツール 統計解析ソフトRによる分析②

次元削減の例



主成分分析

RのGUI“Rコマンダー”や“EZR”が 出ているのでコマンドを知らなくても問題ない！



# Pythonの統計分析に便利ツール

これだけあればほとんどの統計分析、グラフ可視化ができる！

1. Numpy

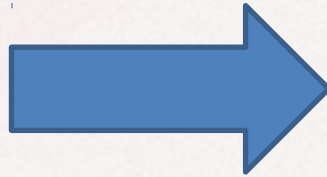
2. pandas

3. Matplotlib

4. Ipython/Ipython notebook

5. Scikit-Learn

6. SciPy



それぞれインストールしてもよいが面倒な場合はAnaconda Pythonというパッケージ(Linux, Windows etc.)が無料でおとせるので

これを使うとよい。

# Pythonの統計分析に便利ツール

## Numpy

- Numerical Python
- Pythonで科学計算をするためのパッケージ
- **強力な多次元配列オブジェクト**
- C, C++, FORTRANのコードとPythonを結合・統合するツール
- 線形代数、フーリエ変換、乱数発生
- データの格納・操作が効率的
- 配列の数学的演算を行う関数

# Pythonの統計分析に便利ツール

## pandas

- panel data + Python data analysis
- Pythonのデータ分析ライブラリ
- 様々なフォーマットのファイルからのデータ読み書き
- CSV, txt, Excel, SQLデータベース等
- **DataFrame**
- 2次元の表形式・列指向のデータ構造オブジェクト

# Pythonの統計分析に便利ツール

pandas

- DataFrame
- データの並び替え、欠損値の取り扱い、再形成、スライス、ダイシング、集約、部分集合の選択が容易
- Rのdata.frameと似ているがより多機能らしい

# Pythonの統計分析に便利ツール

## Matplotlib

- Mathematical + plot + library ?
- Python向け2D描画ライブラリ
- **グラフ等データの可視化**
- MATLABライクなインターフェイス

# Pythonの統計分析に便利ツール

## IPython/IPython notebook

- Pythonを対話的に実行するためのシェル
- Pythonのシェルの拡張したもの
- 型推定の強化
- コードハイライティング、タブによる補完
- Python用に開発されたが他の言語でも使える
- **IPython notebook**



わたしPythonは素人なのですが、これはバカでも使える仕様で大変使いやすい！！

# Pythonの統計分析に便利ツール

- IPython/IPython notebook

- IPython notebook

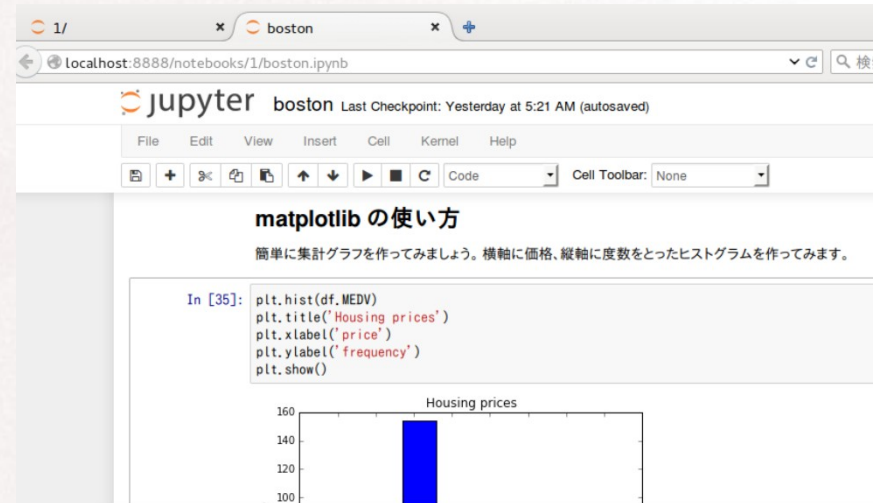
- ウェブブラウザ上で動作

- インラインで図の表示も可能

- コードと結果がブラウザ上で同時に見られるため  
試行錯誤しながら数値計算するのにお勧め

- セル単位で実行

- メモもできる。学校授業など向き？



# Pythonの統計分析に便利ツール

## SciPy

- 科学計算のためのオープンソースソフトウェアのコレクション

- SciPyライブラリ

- 数値計算ルーティーン

- SciPy Stack core packages

=Python + NumPy + SciPy library

+Matplotlib +pandas + Ipython

+SymPy + nose

- nose = テストフレームワーク

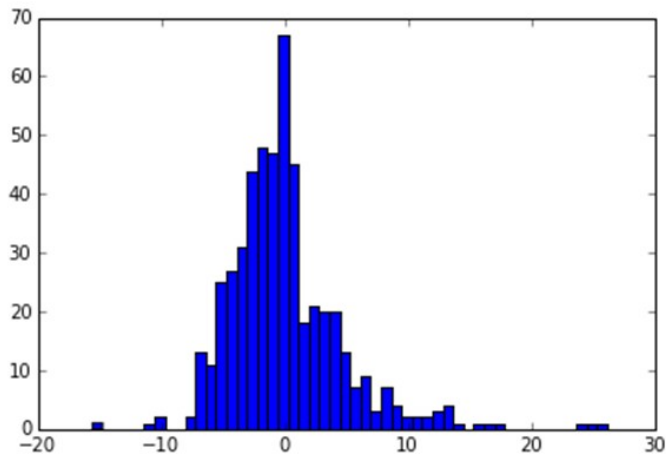


# Scikit-learn を使った統計分析例①

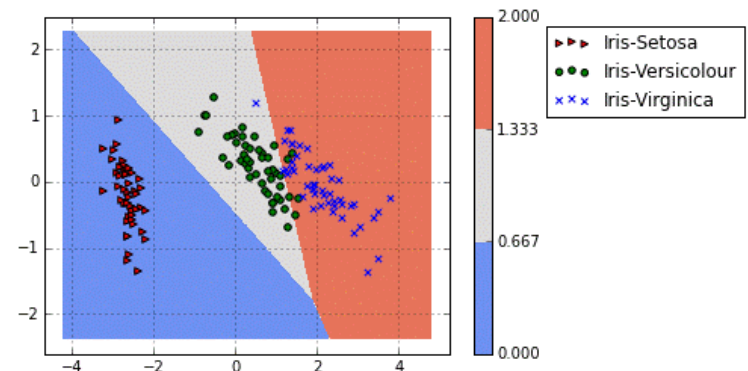
線形回帰など基本機能は当然あります

まず一番標準的な線形回帰を行い、実測値と予測の差をヒストグラムにしてみましょう。線形回帰は `line` ティングを行い、`predict` で予測値を計算します。

```
In [15]: from sklearn.linear_model import LinearRegression
lr = LinearRegression(normalize=True)
lr.fit(boston.data, boston.target)
predicted = lr.predict(boston.data)
plt.hist(boston.target - predicted, bins=50)
plt.show()
```

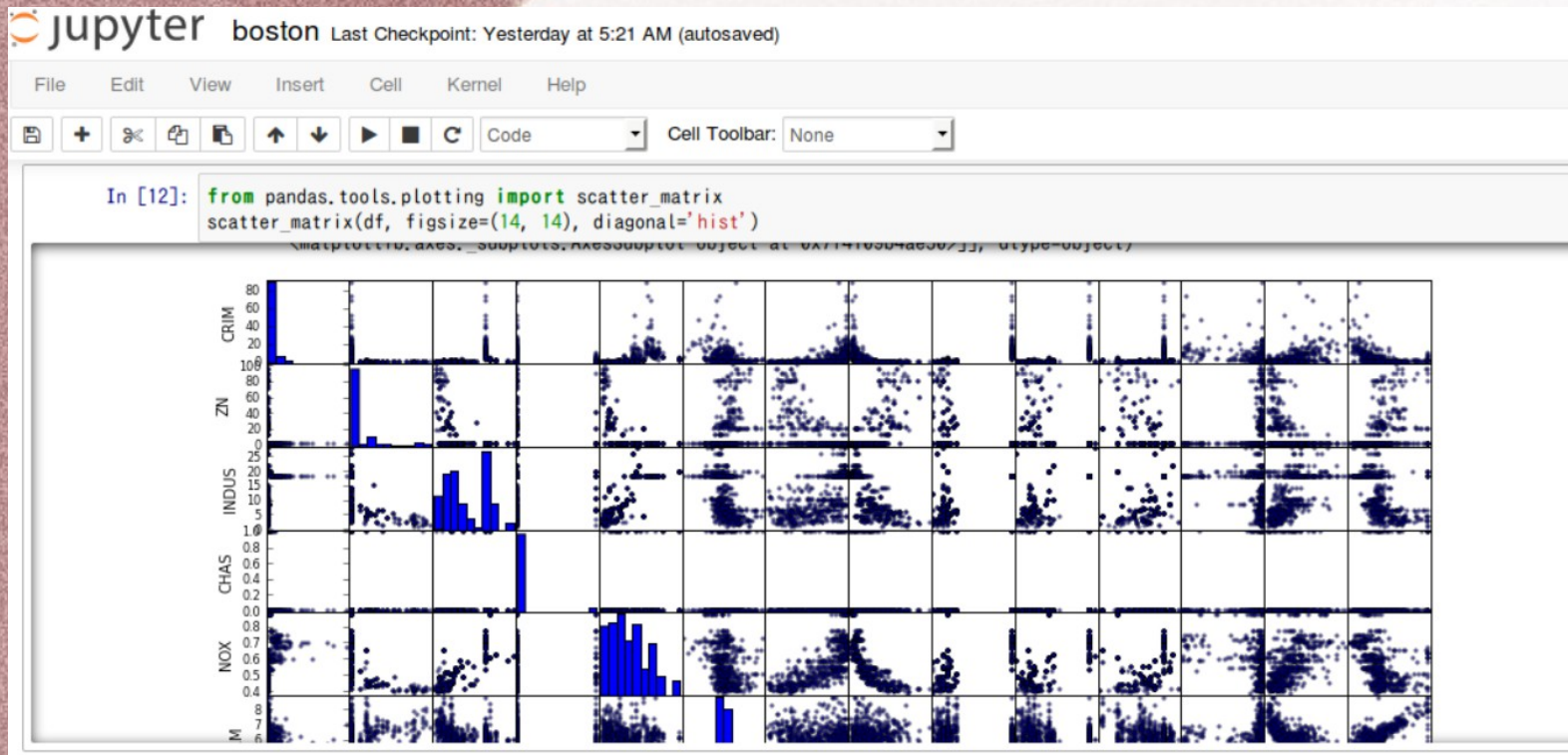


```
# LinearSVC識別器を作って等高線を書く
draw_contour_and_data(data_pca, LinearSVC(dual=False, penalty='l1'))
```



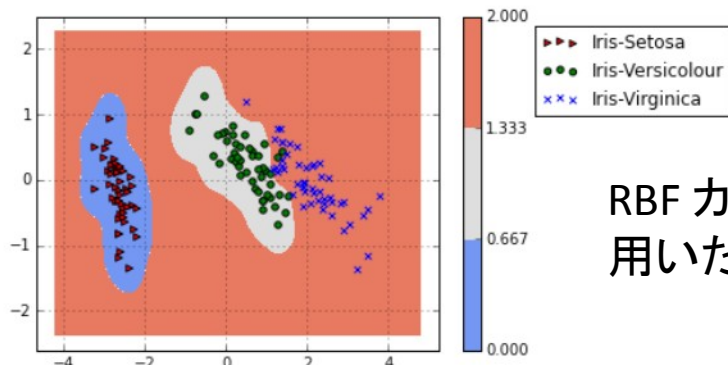
グラフはMatplotlib で書きます

# Scikit-learn を使った統計分析例②



散布図マトリックス

In [20]: `draw_contour_and_data(data_pca, SVC(kernel='rbf', gamma=10.0, ))`



RBF カーネルを用いた識別

Scikit-learn のR より優れた点はガウス関数を用いたRBF ネットワークモデルで近似関数、識別を行うことができる点(R は線形近似のみ)

# まとめ

- Python から無料で使えるCAEと最適化ツールと関連ソフトについて調査したこれまでの結果をまとめてみました。
- 最近のPythonオープンソースの充実度はすさまじく、ただで数年前は？？？百万円はしたソフトと同様のことができます。
- 関連資料を無料で公開してくれている有志のみなさんに感謝します。