

近似計算(ロバスト推定)

(Ver.1.5)

2016 年 5 月

株式会社 アイディール

目次

eyemFitMean1dim	1
機能 1次元点群の平均値(重心)	1
eyemFitMean2dim	2
機能 2次元点群の平均値(重心)	2
eyemFitMean3dim	3
機能 3次元点群の平均値(重心)	3
eyemFitLine	4
機能 平面上の直線近似	4
eyemFitLineIn3d	5
機能 3次元空間内の直線近似	5
eyemFitPlane	6
機能 平面近似	6
eyemFitCircle	7
機能 円近似	7
eyemFitEllipse	8
機能 楕円近似(楕円パラメータの算出)	8
eyemFitEllipseC	9
機能 楕円近似(2次曲線の係数の算出)	9
eyemFitConics	10
機能 2次曲線の当てはめ	10
eyemFitParabola	11
機能 2次関数(放物線)の当てはめ	11
eyemFitEllipsoid	12
機能 楕円体近似	12

機 能	1次元点群の平均値(重心)
-----	---------------

形 式 #include “eyemLib.h”

解 説 1次元点群に対して、平均値(重心)のロバスト推定を行います.

引 数 n データ(座標)点の数です. 2以上を入力してください.

daPoint□ 各点の1次元座標値です. 2点以上を入力してください.

iCalcMode 処理選択のフラグです。最小二乗法(通常平均)の場合は**0**, ロバスト推定の場合は**1**を入力してください。

dRobustCoef 点群のばらつきを表す係数です。iCalcMode=1 (ロバスト推定を選択) のときのみ有効となります。この係数の目安は、データのばらつきの標準偏差が σ であって $k\sigma$ 以上のデータ点を無視したい場合には、この係数 ' k ' を入力してください(推奨値:2.7)。

*dpMean 推定した平均値(重心)が格納されます.

戻り値 エラー報告です.

FUNC_OK	正常終了
FUNC_NOT_ENOUGH_MEM	メモリ不足
FUNC_ILLEGAL_ARGUMENT	パラメータが不正
FUNC_FAILED_ROBUST	ロバスト推定失敗

留意事項 特にありません.

eyemFitMean2dim

機 能 2次元点群の平均値(重心)

形 式 `#include "eyemLib.h"`
`int eyedFitMean2dim (int n, EyemOcsDXY taPoint[], int iCalcMode,`
`double dRobustCoef, EyemOcsDXY *tpMean);`

解 説 2次元点群に対して, 平均値(重心)のロバスト推定を行います.

引 数 `n` データ(座標)点の数です. 2以上を入力してください.
`taPoint[]` 各点の2次元座標値です. 2点以上を入力してください.
`iCalcMode` 処理選択のフラグです. 最小二乗法(通常平均)の場合は0, ロバスト推定の場合は1を入力してください.
`dRobustCoef` 点群のばらつきを表す係数です. `iCalcMode=1` (ロバスト推定を選択)のときのみ有効となります. この係数の目安は, データのばらつきの標準偏差が σ であって $k\sigma$ 以上のデータ点を無視したい場合には, この係数 ' k ' を入力してください(推奨値:2.7).
`*tpMean` 推定した平均値(重心)が格納されます.

戻り値 エラー報告です.

<code>FUNC_OK</code>	正常終了
<code>FUNC_NOT_ENOUGH_MEM</code>	メモリ不足
<code>FUNC_ILLEGAL_ARGUMENT</code>	パラメータが不正
<code>FUNC_FAILED_ROBUST</code>	ロバスト推定失敗

留意事項 特にありません.

eyemFitMean3dim

機 能 3次元点群の平均値(重心)

形 式

```
#include "eyemLib.h"

int      eyedFitMean3dim ( int n, EyemOcsDXYZ taPoint[], int iCalcMode,
                           double dRobustCoef, EyemOcsDXYZ *tpMean );
```

解 説 3次元点群に対して, 平均値(重心)のロバスト推定を行います.

引 数

n	データ(座標)点の数です. 2以上を入力してください.
taPoint[]	各点の3次元座標値です. 2点以上を入力してください.
iCalcMode	処理選択のフラグです. 最小二乗法(通常平均)の場合は0, ロバスト推定の場合は1を入力してください.
dRobustCoef	点群のばらつきを表す係数です. iCalcMode=1(ロバスト推定を選択)のときのみ有効となります. この係数の目安は, データのばらつきの標準偏差が σ であって $k\sigma$ 以上のデータ点を無視したい場合には, この係数'k'を入力してください(推奨値:2.7).
*tpMean	推定した平均値(重心)が格納されます.

戻り値 エラー報告です.

FUNC_OK	正常終了
FUNC_NOT_ENOUGH_MEM	メモリ不足
FUNC_ILLEGAL_ARGUMENT	パラメータが不正
FUNC_FAILED_ROBUST	ロバスト推定失敗

留意事項 特にありません.

eyemFitLine

機 能 平面上の直線近似

形 式 `#include "eyemLib.h"`
`int eyedFitLine (int n, EyemOcsDXY taPoint[], int iCalcMode,`
`double dRobustCoef, EyemOcsDABC *tpLine);`

解 説 2次元点群に対して直線 $ax + by + c = 0$ を当てはめ、その係数 a , b , c のロバスト推定を行います。

引 数 `n` データ(座標)点の数です。2以上を入力してください。
`taPoint[]` 各点の2次元座標値です。2点以上を入力してください。
`iCalcMode` 処理選択のフラグです。最小二乗法の場合は`0`、ロバスト推定の場合は`1`を入力してください。
`dRobustCoef` 点群のばらつきを表す係数です。 `iCalcMode=1` (ロバスト推定を選択) のときのみ有効となります。この係数の目安は、データのばらつきの標準偏差が σ であって $k\sigma$ 以上のデータ点を無視したい場合には、この係数 ' k ' を入力してください(推奨値:2.7)。
`*tpLine` 推定した近似直線の係数が格納されます。なお、法線ベクトル (a,b) は単位ベクトルとなっています。

戻り値 エラー報告です。

FUNC_OK	正常終了
FUNC_NOT_ENOUGH_MEM	メモリ不足
FUNC_ILLEGAL_ARGUMENT	パラメータが不正
FUNC_FAILED_ROBUST	ロバスト推定失敗
FUNC_FAILED_EIGEN	固有値計算失敗

留意事項 特にありません。

eyemFitLineIn3d

機 能 3次元空間内の直線近似

形 式 `#include "eyemLib.h"`
`int eyedFitLineIn3d (int n, EyemOcsDXYZ taPoint[], int iCalcMode,`
`double dRobustCoef, EyemOcsDPV *tpLine);`

解 説 3次元点群に対して直線 $\mathbf{x} = \mathbf{p} + t\mathbf{v}$ (直線上の1点 \mathbf{p} と方向ベクトル \mathbf{v} で表したベクトル方程式. t は実数) を当てはめ, その直線のパラメータ \mathbf{p} と \mathbf{v} のロバスト推定を行います.

引 数 `n` データ(座標)点の数です. 2以上を入力してください.
`taPoint[]` 各点の3次元座標値です. 2点以上を入力してください.
`iCalcMode` 処理選択のフラグです. 最小二乗法の場合は`0`, ロバスト推定の場合は`1`を入力してください.
`dRobustCoef` 点群のばらつきを表す係数です. `iCalcMode=1` (ロバスト推定を選択) のときのみ有効となります. この係数の目安は, データのばらつきの標準偏差が σ であって $k\sigma$ 以上のデータ点を無視したい場合には, この係数 ' k ' を入力してください(推奨値:2.7).
`*tpLine` 推定した近似直線のパラメータ \mathbf{p} と \mathbf{v} が格納されます. なお, 方向ベクトル \mathbf{v} は単位ベクトルとなっています.

戻り値 エラー報告です.

<code>FUNC_OK</code>	正常終了
<code>FUNC_NOT_ENOUGH_MEM</code>	メモリ不足
<code>FUNC_ILLEGAL_ARGUMENT</code>	パラメータが不正
<code>FUNC_FAILED_ROBUST</code>	ロバスト推定失敗
<code>FUNC_FAILED_EIGEN</code>	固有値計算失敗

留意事項 特にありません.

eyemFitPlane

機 能 平面近似

形 式 `#include "eyemLib.h"`
`int eyedFitPlane (int n, EyemOcsDXYZ taPoint[], int iCalcMode,`
`double dRobustCoef, EyemOcsDABCD *tpPlane);`

解 説 3次元点群に対して平面 $ax + by + cz + d = 0$ を当てはめ, その係数 a, b, c, d のロバスト推定を行います.

引 数 `n` データ(座標)点の数です. 3以上を入力してください.
`taPoint[]` 各点の3次元座標値です. 3以上を入力してください.
`iCalcMode` 処理選択のフラグです. 最小二乗法の場合は`0`, ロバスト推定の場合は`1`を入力してください.
`dRobustCoef` 点群のばらつきを表す係数です. `iCalcMode=1` (ロバスト推定を選択) のときのみ有効となります. この係数の目安は, データのばらつきの標準偏差が σ であって $k\sigma$ 以上のデータ点を無視したい場合には, この係数 ' k ' を入力してください(推奨値:2.7).
`*tpPlane` 推定した近似平面の係数が格納されます. なお, 法線ベクトル (a,b,c) は単位ベクトルとなっています.

戻り値 エラー報告です.

<code>FUNC_OK</code>	正常終了
<code>FUNC_NOT_ENOUGH_MEM</code>	メモリ不足
<code>FUNC_ILLEGAL_ARGUMENT</code>	パラメータが不正
<code>FUNC_FAILED_ROBUST</code>	ロバスト推定失敗
<code>FUNC_FAILED_EIGEN</code>	固有値計算失敗

留意事項 特にありません.

eyemFitCircle

機 能 円近似

形 式 `#include "eyemLib.h"`
`int eyedFitCircle (int n, EyemOcsDXY taPoint[], int iCalcMode,`
`double dRobustCoef, EyemOcsDXYR *tpCircle);`

解 説 2次元点群に対して円 $(x-a)^2 + (y-b)^2 = r^2$ を当てはめ, その中心座標 (a, b) および半径 r のロバスト推定を行います.

引 数 `n` データ(座標)点の数です. 3以上を入力してください.
`taPoint[]` 各点の2次元座標値です. 3以上を入力してください.
`iCalcMode` 処理選択のフラグです. 最小二乗法の場合は`0`, ロバスト推定の場合は`1`を入力してください.
`dRobustCoef` 点群のばらつきを表す係数です. `iCalcMode=1` (ロバスト推定を選択) のときのみ有効となります. この係数の目安は, データのばらつきの標準偏差が σ であって $k\sigma$ 以上のデータ点を無視したい場合には, この係数 ' k ' を入力してください(推奨値:2.7).
`*tpCircle` 推定した近似円の中心座標と半径が格納されます.

戻り値 エラー報告です.

<code>FUNC_OK</code>	正常終了
<code>FUNC_NOT_ENOUGH_MEM</code>	メモリ不足
<code>FUNC_ILLEGAL_ARGUMENT</code>	パラメータが不正
<code>FUNC_FAILED_ROBUST</code>	ロバスト推定失敗

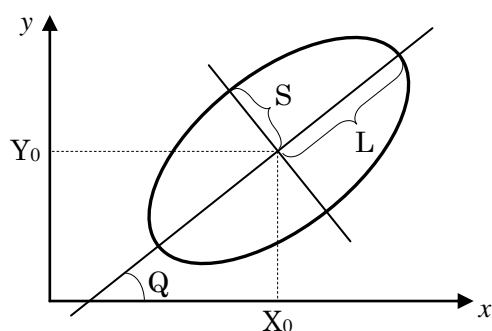
留意事項 特にありません.

eyemFitEllipse

機 能 楕円近似 (楕円パラメータの算出)

形 式 `#include "eyemLib.h"`
`int eyemFitEllipse (int n, EyemOcsDXY taPoint[], int iCalcMode,`
`double dRobustCoef, EyemOcsDXYLSQ *tpEllipse);`

解 説 2次元点群に対して楕円を当てはめ, その中心座標 (X_0, Y_0) , 長軸半径 L , 短軸半径 S および長軸の傾き Q (x 軸から測った角) のロバスト推定を行います.



引 数 `n` データ(座標)点の数です. 5以上を入力してください.
`taPoint[]` 各点の2次元座標値です. 5点以上を入力してください.
`iCalcMode` 処理選択のフラグです. 最小二乗法の場合は**0**, ロバスト推定の場合は**1**を入力してください.
`dRobustCoef` 点群のばらつきを表す係数です. `iCalcMode=1` (ロバスト推定を選択) のときのみ有効となります. この係数の目安は, データのばらつきの標準偏差が σ であって $k\sigma$ 以上のデータ点を無視したい場合には, この係数 ' k ' を入力してください(推奨値:2.7).
`*tpEllipse` 推定した近似楕円の中心座標 (X_0, Y_0) , 長軸半径 L , 短軸半径 S および長軸の傾き Q (x 軸から測った角:rad単位, $-\pi/2 \leq Q \leq \pi/2$) が格納されます.

戻り値 エラー報告です.

<code>FUNC_OK</code>	正常終了
<code>FUNC_NOT_ENOUGH_MEM</code>	メモリ不足
<code>FUNC_ILLEGAL_ARGUMENT</code>	パラメータが不正
<code>FUNC_FAILED_ROBUST</code>	ロバスト推定失敗
<code>FUNC_DET_EQ_ZERO</code>	行列式がゼロ
<code>FUNC_FAILED_EIGEN</code>	固有値計算失敗
<code>FUNC_FAILED_ELLIPSE</code>	楕円にならなかった

留意事項 特にありません.

eyemFitEllipseC

機 能 楕円近似 (2次曲線の係数の算出)

形 式 `#include "eyemLib.h"`
`int eyedFitEllipseC (int n, EyemOcsDXY taPoint[], int iCalcMode,`
`double dRobustCoef, double daCoef[]);`

解 説 2次元点群に対して楕円を当てはめ、相当する2次曲線 $ax^2 + bxy + cy^2 + dx + ey + f = 0$ の係数 a, b, c, d, e, f のロバスト推定を行います。

引 数 `n` データ(座標)点の数です。5以上を入力してください。
`taPoint[]` 各点の2次元座標値です。5点以上を入力してください。
`iCalcMode` 処理選択のフラグです。最小二乗法の場合は**0**、ロバスト推定の場合は**1**を入力してください。
`dRobustCoef` 点群のばらつきを表す係数です。iCalcMode=1 (ロバスト推定を選択) のときのみ有効となります。この係数の目安は、データのばらつきの標準偏差が σ であって $k\sigma$ 以上のデータ点を無視したい場合には、この係数 ' k ' を入力してください(推奨値:2.7)。
`daCoef[]` 推定した近似楕円を表す2次曲線 $ax^2 + bxy + cy^2 + dx + ey + f = 0$ の係数が格納されます。要素数6の配列を指定してください。係数 a, b, c, d, e, f の順に配列[0]~[5]へ格納されます。

戻り値 エラー報告です。

FUNC_OK	正常終了
FUNC_NOT_ENOUGH_MEM	メモリ不足
FUNC_ILLEGAL_ARGUMENT	パラメータが不正
FUNC_FAILED_ROBUST	ロバスト推定失敗
FUNC_DET_EQ_ZERO	行列式がゼロ
FUNC_FAILED_EIGEN	固有値計算失敗
FUNC_FAILED_ELLIPSE	楕円にならなかった

留意事項 特にありません。

eyemFitConics

機 能 2次曲線の当てはめ

形 式

```
#include "eyemLib.h"
int      eyemFitConics ( int n, EyemOcsDXY taPoint[], int iCalcMode,
                        double dRobustCoef, double daCoef[] );
```

解 説 2次元点群に対して2次曲線 $ax^2 + bxy + cy^2 + dx + ey + f = 0$ を当てはめ、その係数 a, b, c, d, e, f のロバスト推定を行います。

引 数

n	データ(座標)点の数です。5以上を入力してください。
taPoint[]	各点の2次元座標値です。5点以上を入力してください。
iCalcMode	処理選択のフラグです。最小二乗法の場合は 0 、ロバスト推定の場合は 1 を入力してください。
dRobustCoef	点群のばらつきを表す係数です。iCalcMode=1(ロバスト推定を選択)のときのみ有効となります。この係数の目安は、データのばらつきの標準偏差が σ であって $k\sigma$ 以上のデータ点を無視したい場合には、この係数 ' k ' を入力してください(推奨値:2.7)。
daCoef[]	推定した2次曲線 $ax^2 + bxy + cy^2 + dx + ey + f = 0$ の係数が格納されます。要素数6の配列を指定してください。係数 a, b, c, d, e, f の順に配列[0]～[5]へ格納されます。

戻り値 エラー報告です。

FUNC_OK	正常終了
FUNC_NOT_ENOUGH_MEM	メモリ不足
FUNC_ILLEGAL_ARGUMENT	パラメータが不正
FUNC_FAILED_ROBUST	ロバスト推定失敗
FUNC_FAILED_EIGEN	固有値計算失敗

留意事項 特にありません。

eyemFitParabola

機 能 2次関数(放物線)の当てはめ

形 式 `#include "eyemLib.h"`
`int eyedFitParabola(int n, EyemOcsDXY taPoint[], int iCalcMode, double dRobustCoef, EyemOcsDABC *tpParabola);`

解 説 2次元点群に対して2次関数(放物線) $y = ax^2 + bx + c$ を当てはめ, その係数 a, b, c のロバスト推定を行います.

引 数 `n` データ(座標)点の数です. 3以上を入力してください.
`taPoint[]` 各点の2次元座標値です. 3以上を入力してください.
`iCalcMode` 処理選択のフラグです. 最小二乗法の場合は`0`, ロバスト推定の場合は`1`を入力してください.
`dRobustCoef` 点群のばらつきを表す係数です. `iCalcMode=1` (ロバスト推定を選択) のときのみ有効となります. この係数の目安は, データのばらつきの標準偏差が σ であって $k\sigma$ 以上のデータ点を無視したい場合には, この係数 ' k ' を入力してください(推奨値:2.7).
`tpParabola` 推定した2次関数 $y = ax^2 + bx + c$ の係数 a, b, c が格納されます.

戻り値 エラー報告です.

<code>FUNC_OK</code>	正常終了
<code>FUNC_NOT_ENOUGH_MEM</code>	メモリ不足
<code>FUNC_ILLEGAL_ARGUMENT</code>	パラメータが不正
<code>FUNC_FAILED_ROBUST</code>	ロバスト推定失敗

留意事項 特にありません.

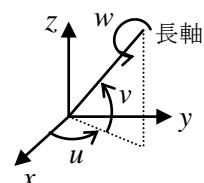
eyemFitEllipsoid

機 能 楕円体近似

形 式

```
#include "eyemLib.h"
int      eyedFitEllipsoid ( int n, EyemOcsDXYZ taPoint[], int iCalcMode,
                           double dRobustCoef, EyemOcsDCRUVW *tpEllipsoid );
```

解 説 3次元点群に対して楕円体を当てはめ、その中心座標、長軸半径、中軸半径、短軸半径および楕円体の姿勢(長軸の xy 平面への射影と x 軸とのなす角 u , 長軸と xy 平面とのなす角 v および長軸まわりの回転角 w)のロバスト推定を行います。



引 数 n データ(座標)点の数です. 9以上を入力してください.
 taPoint[] 各点の3次元座標値です. 9点以上を入力してください.
 iCalcMode 処理選択のフラグです. 最小二乗法の場合は**0**, ロバスト推定の場合は**1**を入力してください.
 dRobustCoef 点群のばらつきを表す係数です. iCalcMode=1 (ロバスト推定を選択)のときのみ有効となります. この係数の目安は, データのばらつきの標準偏差が σ であって $k\sigma$ 以上のデータ点を無視したい場合には, この係数 ' k ' を入力してください(推奨値:2.7).
 *tpEllipsoid 推定した近似楕円体の中心座標, 長軸半径, 中軸半径, 短軸半径および楕円体の姿勢(長軸の xy 平面への射影と x 軸とのなす角 u (rad; $0 \leq u \leq \pi$), 長軸と xy 平面とのなす角 v (rad; $-\pi/2 \leq v \leq \pi/2$) および長軸まわりの回転角 w (rad; $-\pi/2 \leq w \leq \pi/2$))が格納されます.

戻り値 エラー報告です.

FUNC_OK	正常終了
FUNC_NOT_ENOUGH_MEM	メモリ不足
FUNC_ILLEGAL_ARGUMENT	パラメータが不正
FUNC_FAILED_ROBUST	ロバスト推定失敗
FUNC_DET_EQ_ZERO	行列式がゼロ
FUNC_FAILED_EIGEN	固有値計算失敗
FUNC_FAILED_ELLIPSOID	楕円体にならなかった

留意事項 特にありません.

改訂履歴

Version No.	内 容
1.0	• 新規発行
1.1	• 楕円体近似の追加
1.2	• 楕円近似(係数算出)の追加
1.3	• 誤記の修正および説明の追記
1.4	• 3次元空間内の直線近似の新規追加および説明の追記
1.5	• 2次関数当てはめの追加