資料枚数50枚

講演2：14:20～14:50

データサイエンスと品質管理

～観測空間の異常を捉まえるには～

(株)デンソー生産技術部　　吉野　睦

デンソーのF-IoT

F-IoTの目指す姿（ダントツのF-IoT構想）

図表割愛

ファクトリーIoTは、着々と進行している

F-IoTによって実現される品質管理

- 工程が正常状態から逸脱し始めたことを**検知**し、**是正**を行うことによって、正常状態を**維持**すること。
- 工程ロスのゼロ化・・・カーボンニュートラルに貢献

  - **検知**：製品個々にQRコードが印字され、全工程履歴が把握されているので、多次元空間で検出可。

  - **是正**：異常を検知したら、レイジー・ラーニングによる自律適応制御で工程条件を適正値に是正。

  - **維持**：異常・逸脱が抑制されているので、空間内部の状態変化(いつもと違う)を監視し対処。

F-IoTの進化

進化のステップ	人に例えるなら	IoTでは	検出する上での特徴
異常検知 (自覚症状あり)	熱が出た。咳が出た。これらから風邪のひきはじめを検知する。	何らかの異常値・変化を検知する。	正常空間から逸脱する**裾野データ(低頻度側データ：アウトライア)**を監視。
自律適応制御	寝込まないように、総合感冒薬でも飲んで処置をする。	根本原因は取り除けないが、制御できるパラメータを調整して、特性値を是正する。	
状態監視 (自覚症状なし)	一見元気そうでも(自覚症状がなくても)、無理は禁物。健康管理は必要。	是正が入っているので、低頻度側データ(異常値)は殆ど出現しない。どうするか。	正常である**高頻度側データ**を用いて監視する。
源流改善	もしかすると、ガンかもしれない。ガンであれば早期に取り除くべき。	材料ロットなど根本原因を突き止め対策する。	ここで『因果分析』が登場する。

今回は状態監視の視点で、裾野データに限らず検討する

異常検知・状態監視とは

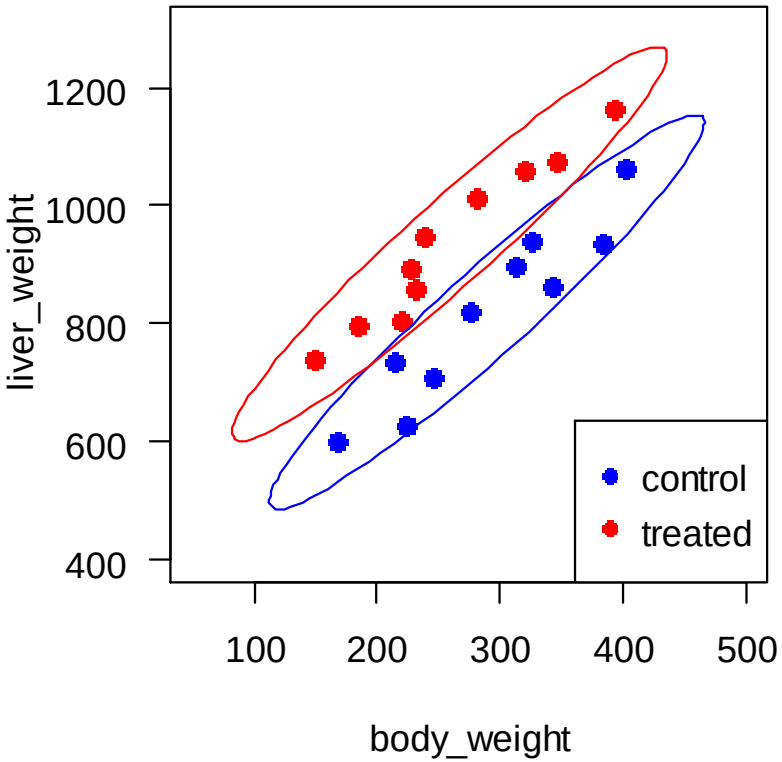
空間の異常をどうやって見つけるか

多次元を監視することの重要性

ラットの肝重量と体重(芳賀ら(1989)「SASによる実験データの解析」東大出版会)

表7.1 ラットの肝重量と体重

薬物投与	体重g	肝重量mg
1	245	710
1	224	627
1	342	865
1	403	1064
1	214	736
1	325	940
1	384	935
1	276	822
1	168	603
1	313	897
2	347	1074
2	220	806
2	282	1013
2	227	893
2	185	798
2	231	861
2	393	1162
2	320	1060
2	148	741
2	238	947



単独項目では、２群に分かれていることは見抜けない

古典的な解決策がヒントになる

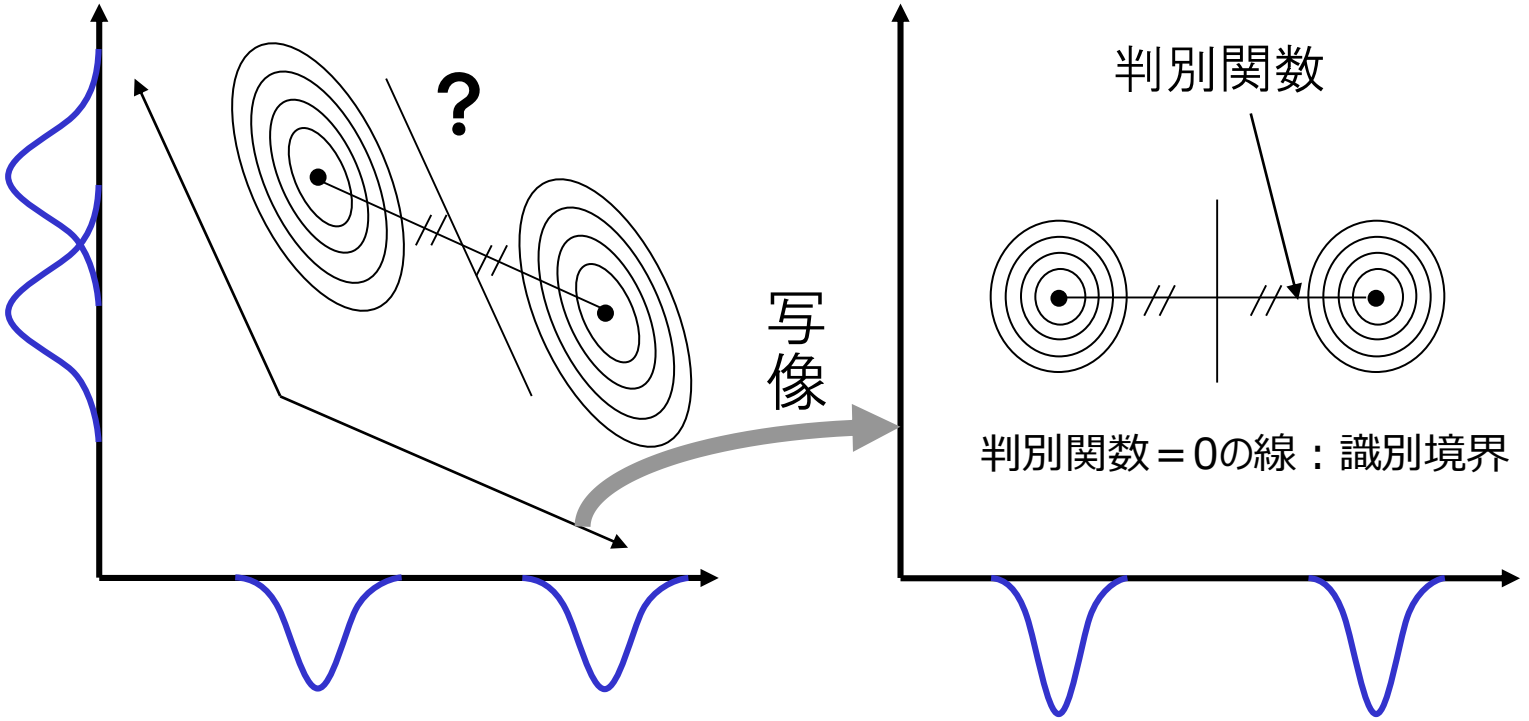
- **多変量分散分析**（E:群内変動，H:群間変動）

ウィルクのラムダ検定	$\Lambda = \det\left(\frac{E}{E+H}\right) = \det\left(\frac{I}{I+E^{-1}H}\right) = \prod_i \frac{1}{1+\lambda_i}$
ローリー・ホテリングの検定	$V = tr\left(E^{-1}\cdot H\right) = \sum_i \lambda_i$
ピライのトレース検定	$\Lambda = tr\left(\left(E+H\right)^{-1}\cdot H\right) = \sum_i \frac{\lambda_i}{1+\lambda_i}$
ロイの最大根検定	$\Theta = \lambda_1$　　最大固有値

ホテリングは何をやっているか

判別分析：$E^{-1} \cdot H$

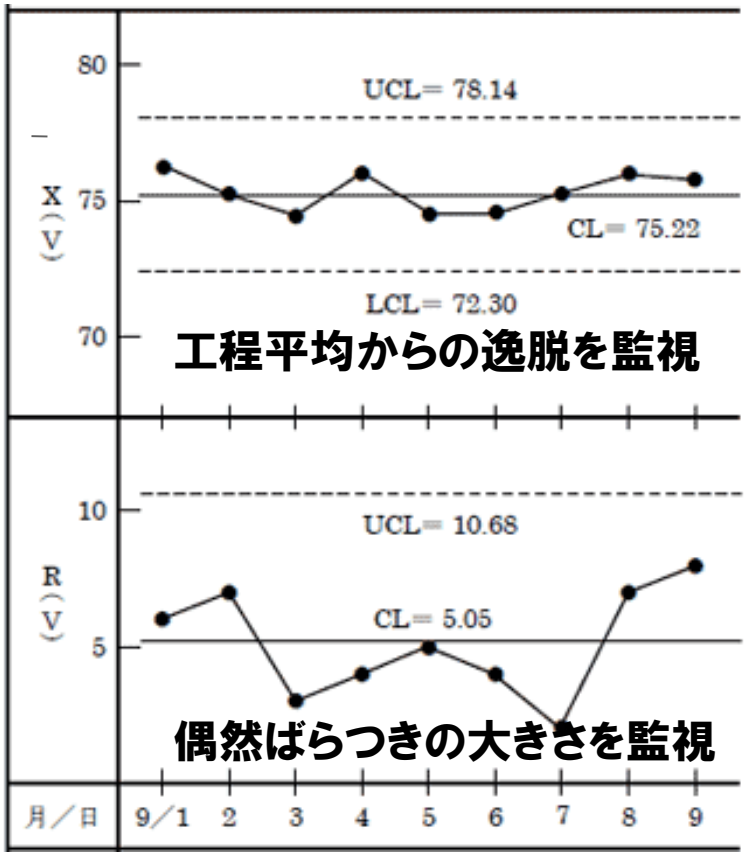
- 群内変動（合併分散）の逆行列を掛けた群間変動をみる。



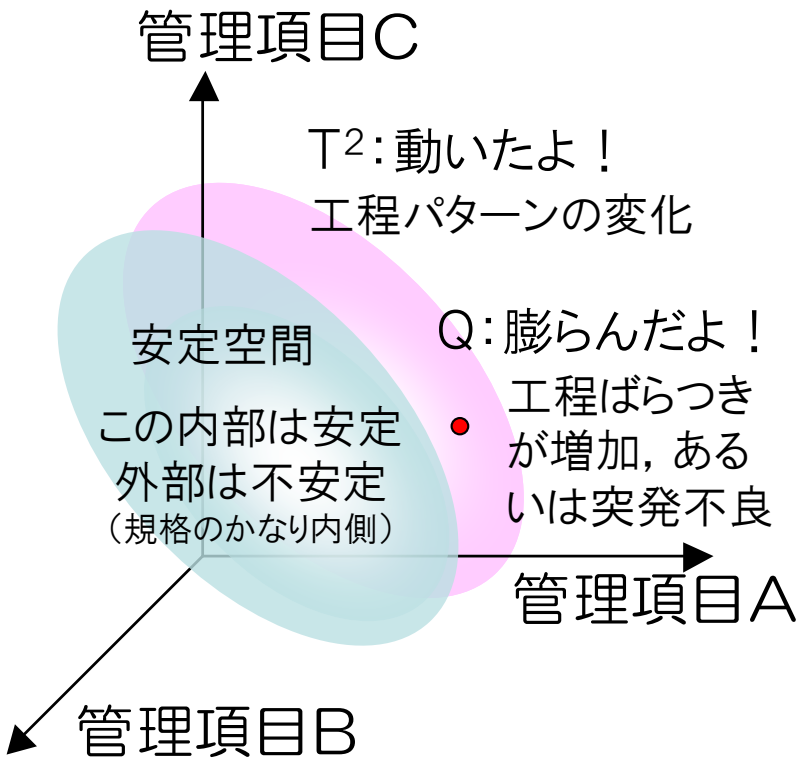
異常検知では、正常群の分散共分散の逆行列を掛ける

さらに発展させたのが、T²-Q

従来の管理図(シューハート管理図)の概念	T²-Q管理図の概念



T²-Q管理図は、系統誤差と偶然誤差を区別



注意すべきは、多次元正規分布が前提で、その向きが変わらないこと

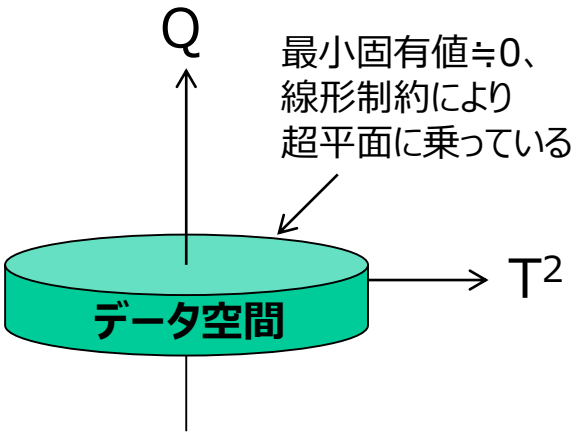
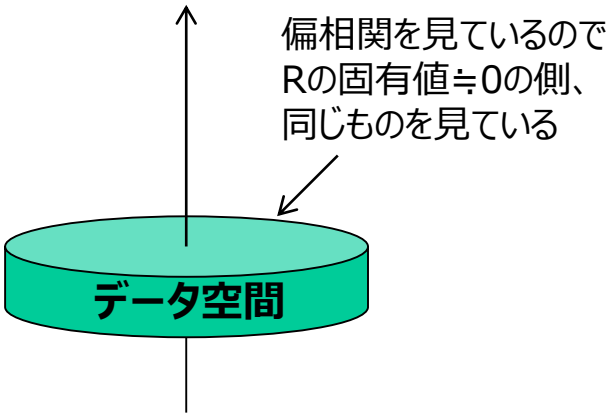
計算式

T^2で系統的変動を見ている。z は主成分スコア。

	計算式
T^2	$T^2 = \frac{1}{\lambda_1} z_1^{\,2} + \frac{1}{\lambda_2} z_2^{\,2} + \cdots + \frac{1}{\lambda_m} z_m^{\,2}$
Q	$Q = \sum_{k=m+1}^{p} z_k^{\,2}$

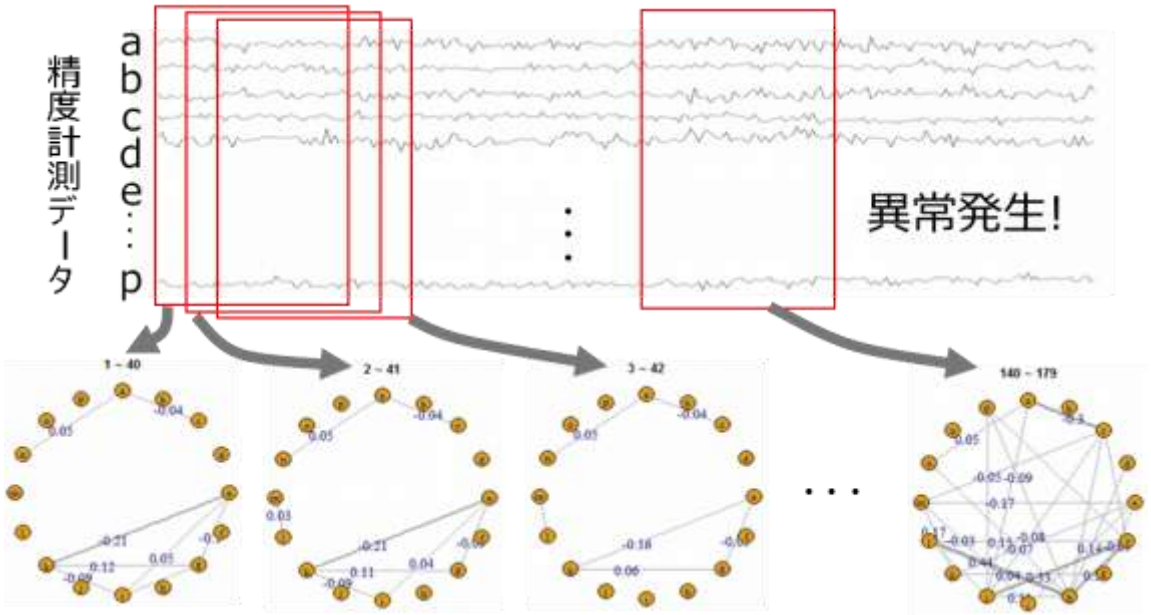
- λ（固有値）などは、前もって安定空間から求めておく。
- 境界となる m は固有値１以上となる主成分数とするなど、現在、明確な基準はなく種々試されている。

厳密には・・・

	原理（何を見ているか）	メリット・デメリット
T^2 Q 管理図	 超平面＝良品成立条件	•工程平均(重心)が動いている(T^2)のか、良品成立条件からの逸脱(Q)なのかが、分離して分かる。 •個々のサンプルについて分かる。
		•円盤の直径は基準化されているので、重心の動きしか分からない。 •円盤の直径の大きさは、高次元では球面集中しているためほぼ一定。
ANACONDA	良品成立条件からの逸脱 	•個々のサンプルではなく、ある区間の群に関する異常度が分かる。 •逸脱原因が項目毎に分かる。
		•相関係数行列(の逆行列)を見るために平均が引かれている。そのため、**重心移動は検出できない**。

ANACONDAとは

- IBMのANACONDAは、良品成立条件(線形制約＝超平面)からの逸脱を監視している。



グラフィカル・ラスー

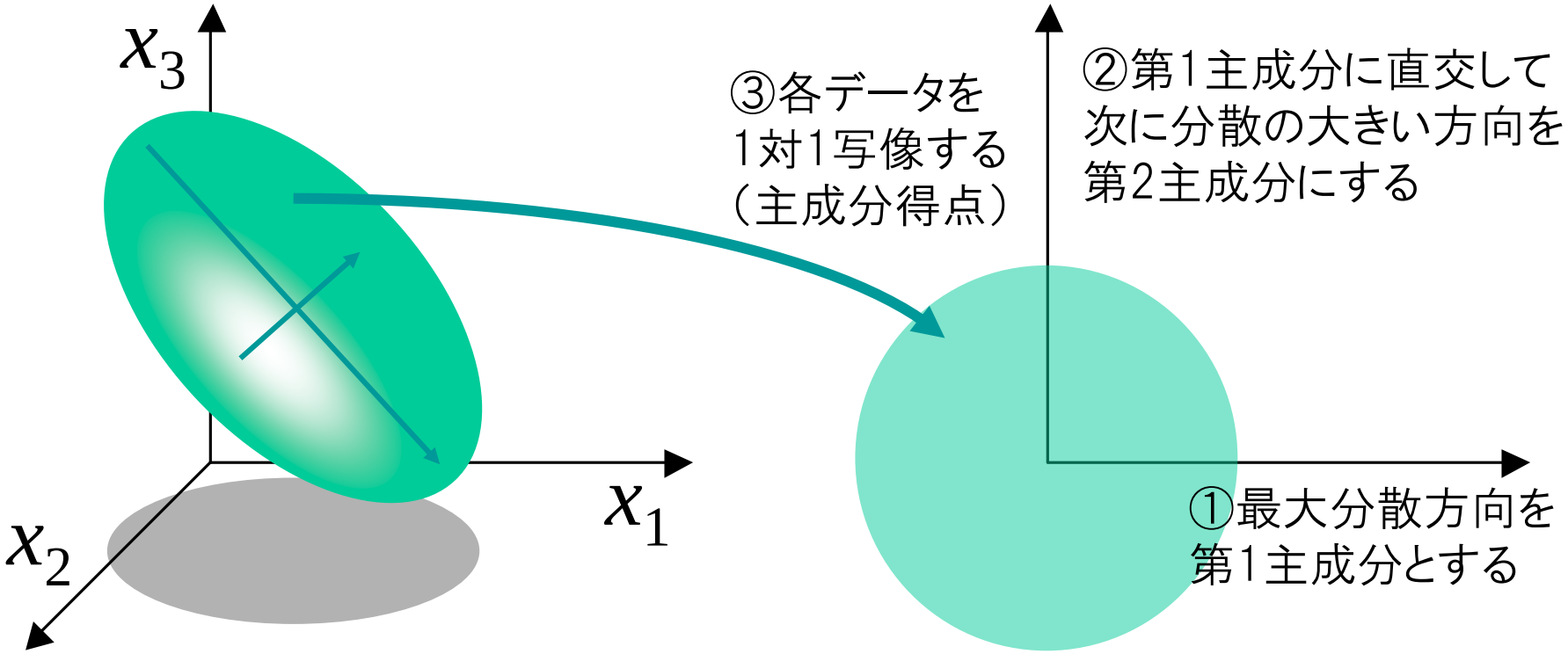
「相関破壊の検知」と呼ばれる

- 弱点は、偏相関係数を使っているので、窓毎に重心は常に0となり、窓間の重心変化など超平面上の動きは検出できない。そこが弱点である。

ピライは何をやっているか

主成分分析：$(E+H)^{-1} \cdot H$

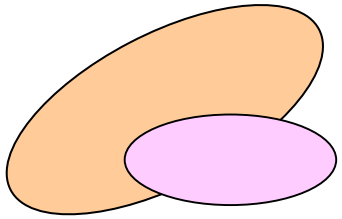
【元のデータ空間】　　【主成分空間】



大福餅のような形状　→　基準化するので球になる

異常群が検出できる

本来のデータ空間

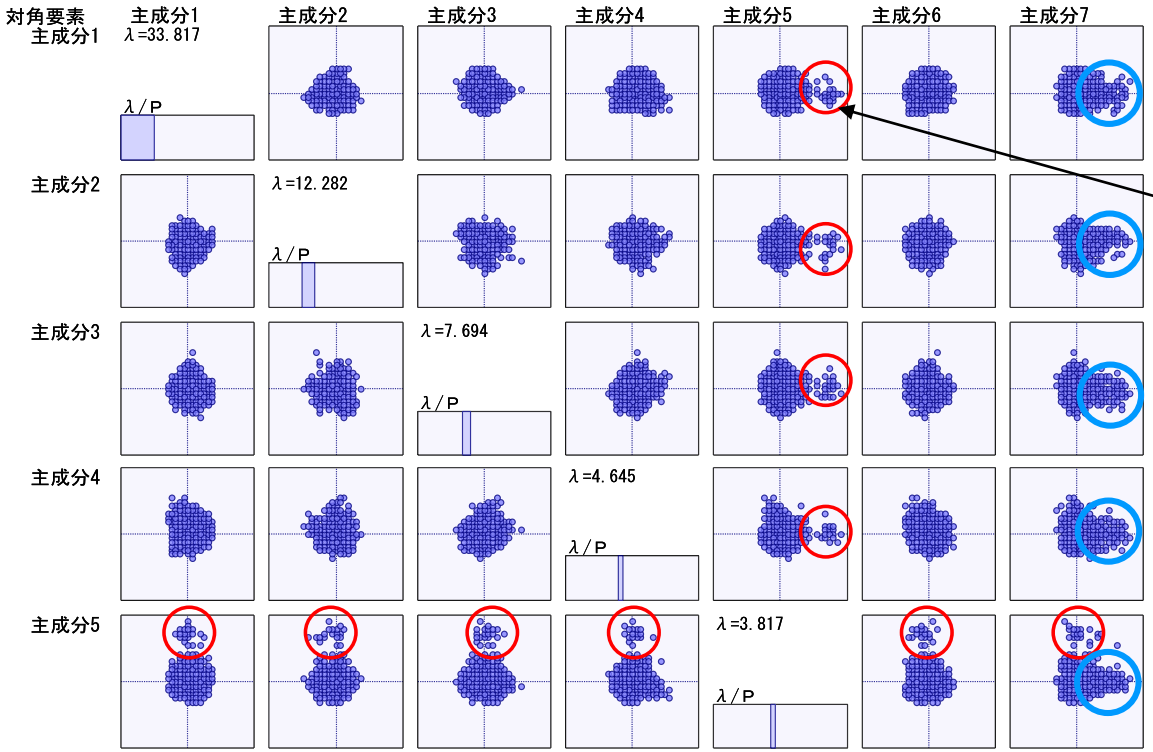


異種状態の混在

もし微妙に重なっていたら、散布図では見えない

変化の大きい軸を取り出してみれば良い

主成分得点のグラフなら「集団の構造」が分かる

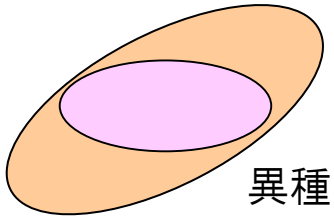


このような、異種状態が混在していることがある

発見することが可能

外れ値が出なくても検知したい

本来のデータ空間



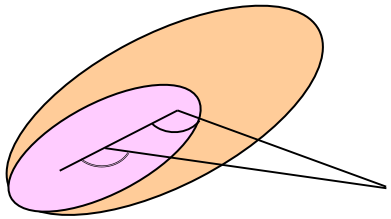
異種状態の混在

主軸の向きが変わったことを検知すれば良い

主軸の向きの変化は「因子負荷量」で見る

だが、主成分分析は**平均を引く**ので、検知できないケースがある。

主軸の変化なし



観測の起点を各重心でなく、外部に置けばよい

移動窓毎に起点を変えるのではなく、常に外部の**定点**に固定する

特異スペクトル変換法

主軸の向きの変化を「ユニタリ行列」で見る

異常検知の実装

異常検知を実際の工程データで確認。

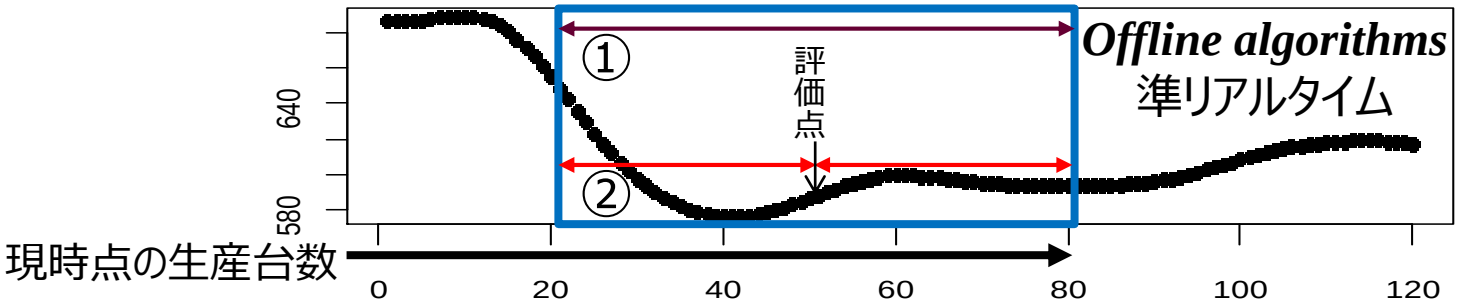
異常検知とは=外れ値検出＋変化点検出

タスク	未来データ要否	概要	主な手法
外れ値検出	× 静的 リアルタイム	• データ点やデータ群が、**教師や基準となる**分布に対して、閾値や確率範囲を越えて外れているか否か。	• マハラノビスの汎距離 • 期待確率からの逸脱(QQplot)
		• 飛び値(デンソー用語)になっていないか＝ギャップ検出。	• 何らかの距離でクラスター分析し、GAP関数で判定
	○ 動的	• 刃具交換などの系統的変動を加味して検出。Moving Windowで外れ値評価、多数決で判定。	
変化点検出	○ 動的	• 時系列データに変化が起きたか否か、また、それはどの時点か。出来れば、リアルタイムに知りたい。	• 各種フィルタ • SDAR • ChangeFinder • SST ・・・
異常検知	**○ 動的**	• **外れ値，変化点等の情報を元に今のデータの状態は、正常か異常か。どの程度か。を知りたい。**	• SVT • EWMA管理図

管理図はリアルタイム、未来データは使用しない ⇔ 準リアルタイム

動的検出とはMoving Window

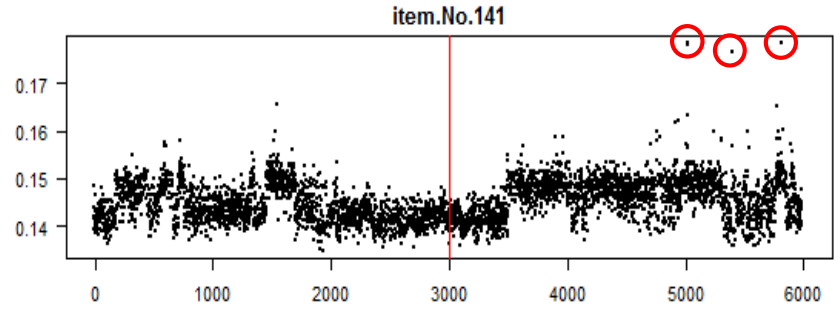
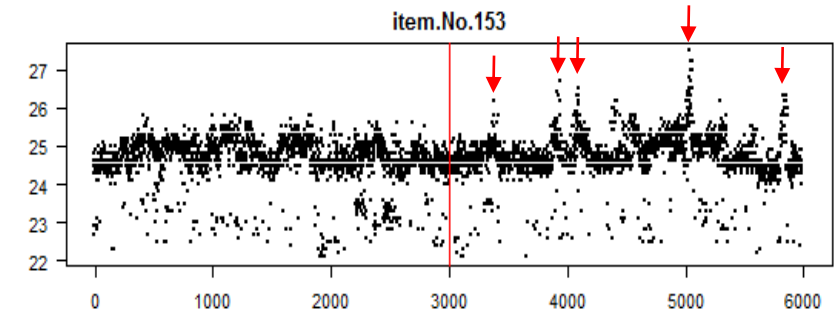
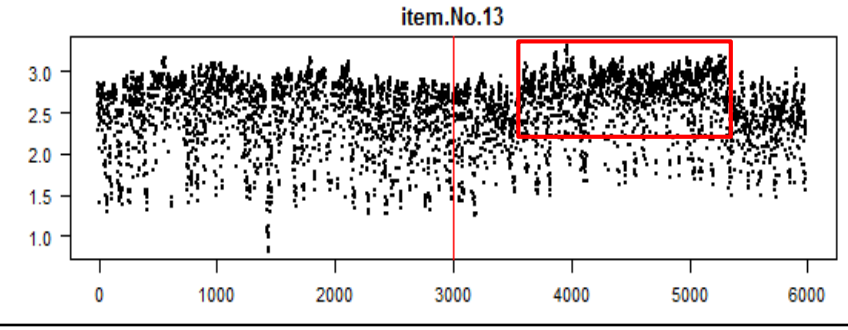
- 窓を設定する。その中央時刻のデータについて評価する。
  - 異常値は、窓内の変化(ベースライン)を差し引いて調べる。
  - 変化点は、①全区間の時系列モデルから実データとのMSEを計算。
  - ②前半と後半で、別々に時系列モデルを作り実データとのMSEを計算。
  - 変化点では、①が最大になり、②が最小になる。



- 両者の差を評価点のスコアとして、**窓をずらしながら**プロット。
- 何を見ているか、
  - **前・後は、同一モデルに従っているか、違うモデルに従っているか。**
  - 違うモデルに従うと仮定した方がMSEが小さいのであれば「変化点」
  - 多項回帰(時間の関数)でも自己回帰でも良い。

データクリーニングの必要性

- データの異常は色々ある。多次元ではそれらが混在している。

予知したい異常	事例	異常ではない点
外れ値		変化点 (刃具交換)
上側の 集団外れ値(尾)		慢性的な 下側の外れ値
平均変動 (変化点)		変化点を 埋もれさす 大きなばらつき

外れ値(アウトライア)検出

静的検出から動的検出まで

多次元アウトライアの検出方法

	異常検知手法	基準	監視している値
排他識別 （アノマリ型） anomaly⇔normaly	期待確率からの逸脱	外れ頻度	別群の混入
	カーネル密度関数法	〃	外れ値
	１クラスSVM	〃	外れ値
	関連性からの逸脱 （ANACONDA™）	KL情報量	超平面に 乗っているかどうか
相対識別 （シグネチャ型）	線形識別器	遷移頻度	状態遷移、または、 負荷量ベクトル
	種々のCクラス分類器	遷移頻度	状態遷移

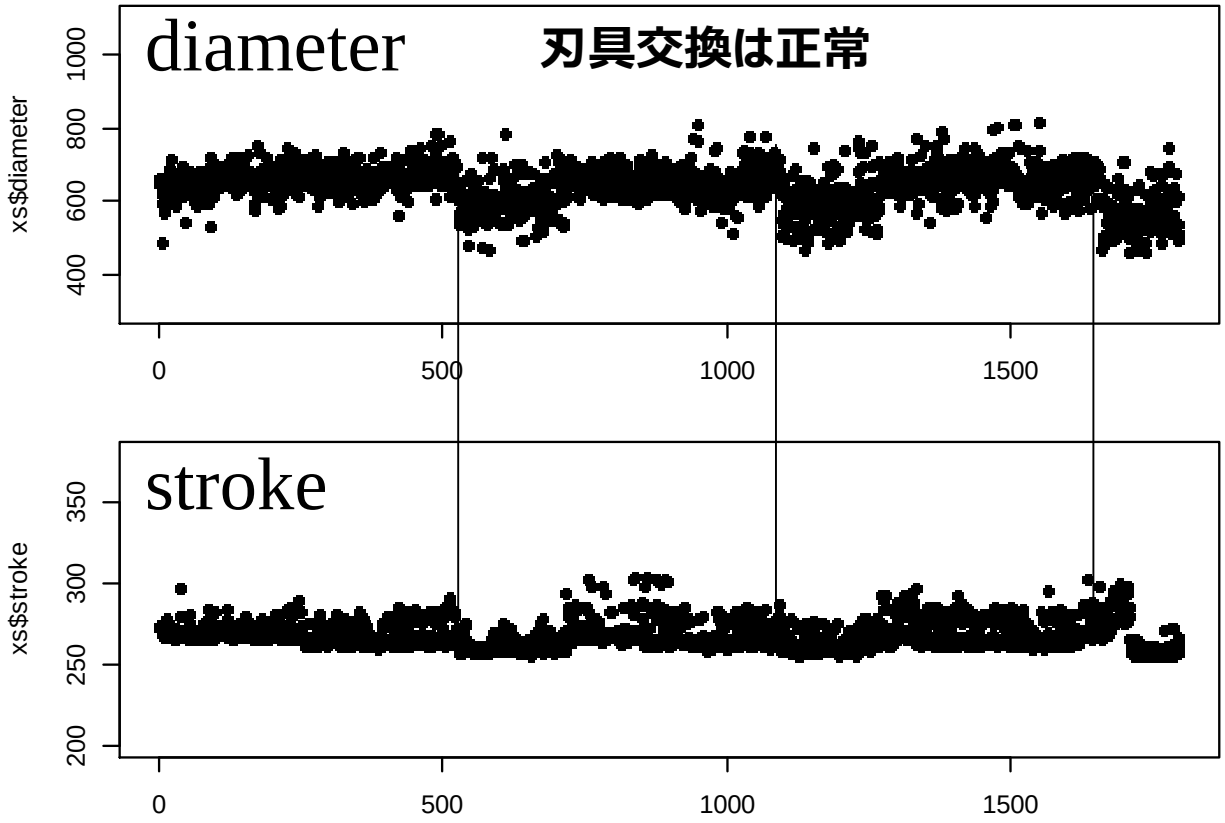
複数の『識別手法』を併用して検出したい異常を発見する

カーネル密度関数法（多次元）

監視対象のデータ空間が『いびつ』で、
等高楕円での境界設定が破綻する時は
データドリブンで境界を引き『外れ値』を監視

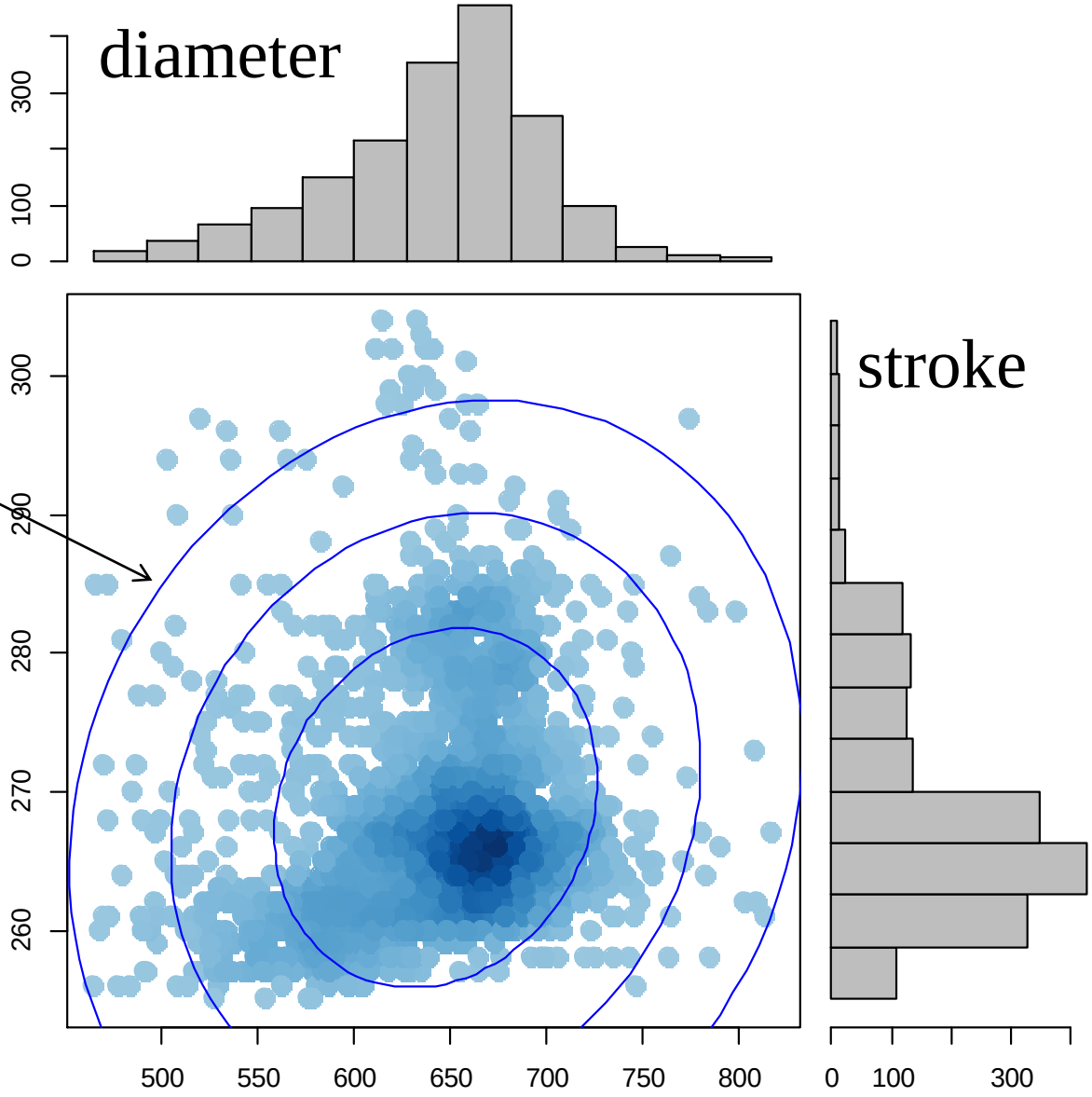
取り上げる事例

- 下記は、連続生産中のマシンニングセンタのデータ。
- 加工品の直径，バイトのストローク。いずれも規格内。



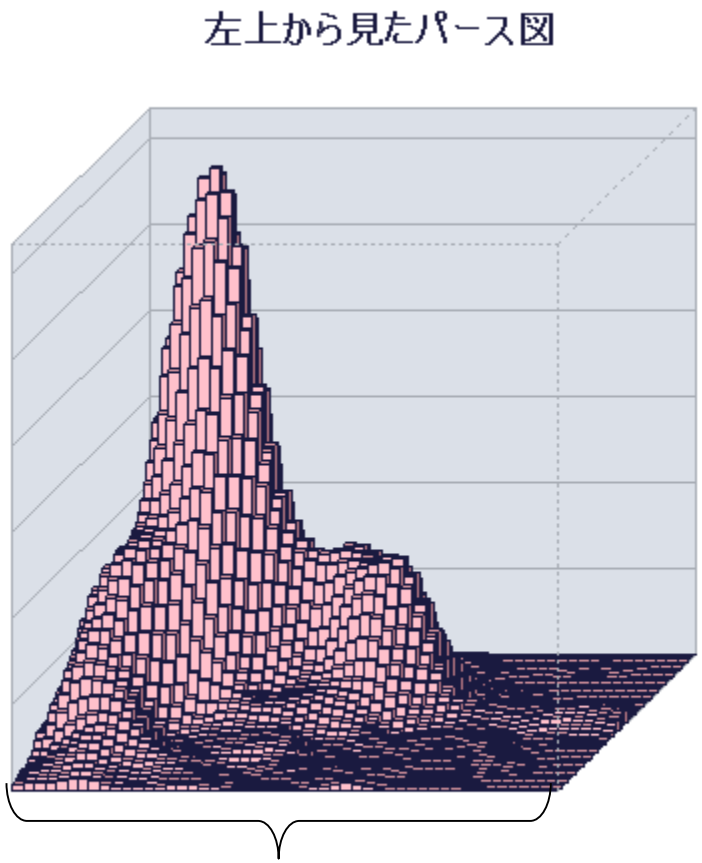
空間密度をプロットしてみる

空間がいびつ
等高楕円は
適用不可



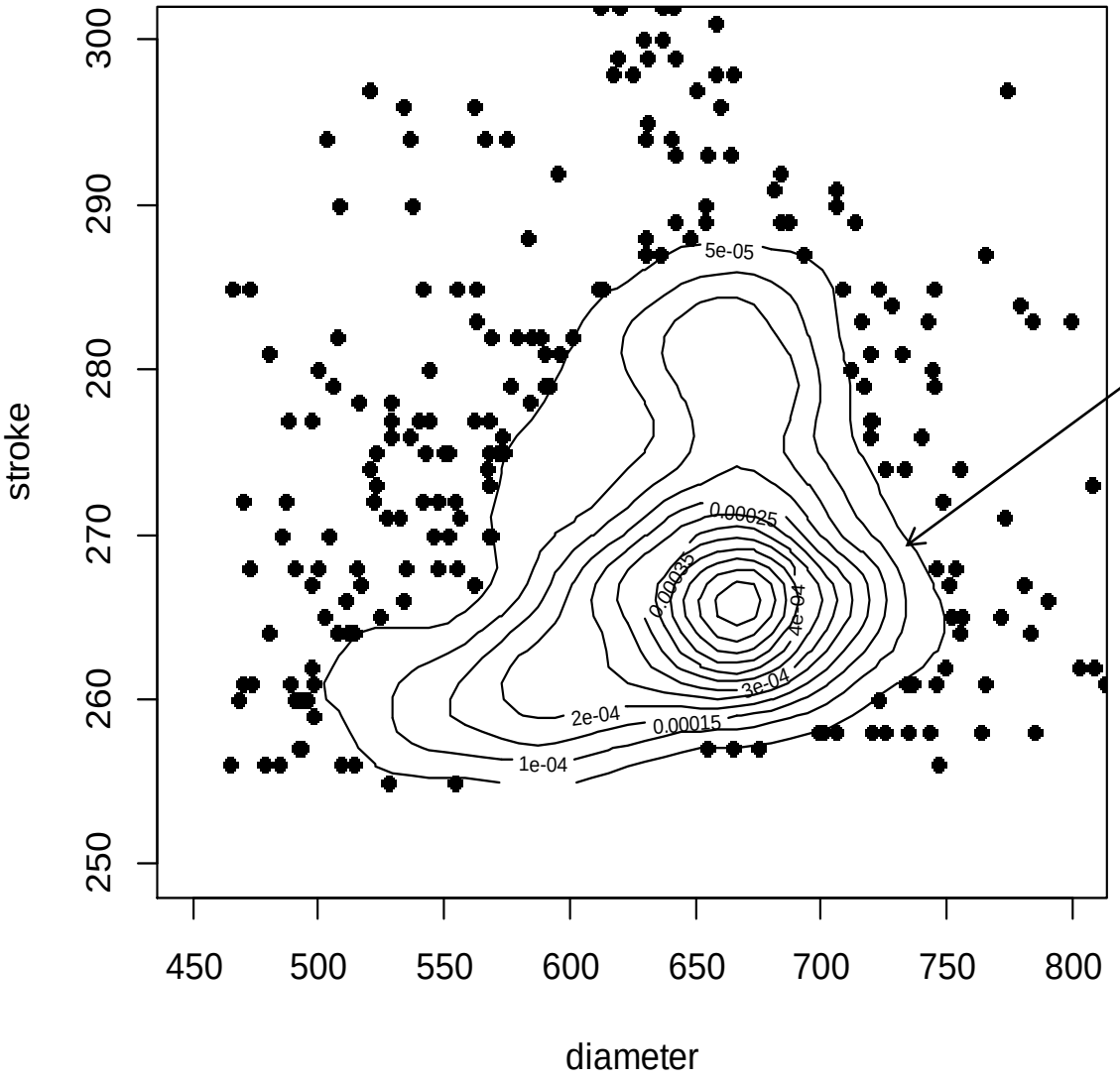
カーネル密度関数法で推定

- 図のようなヒストグラムにRBF（等方的に減衰するガウスの誤差関数）をフィットさせる。



一辺が50分割されている ⇒ 高次元だと膨大化

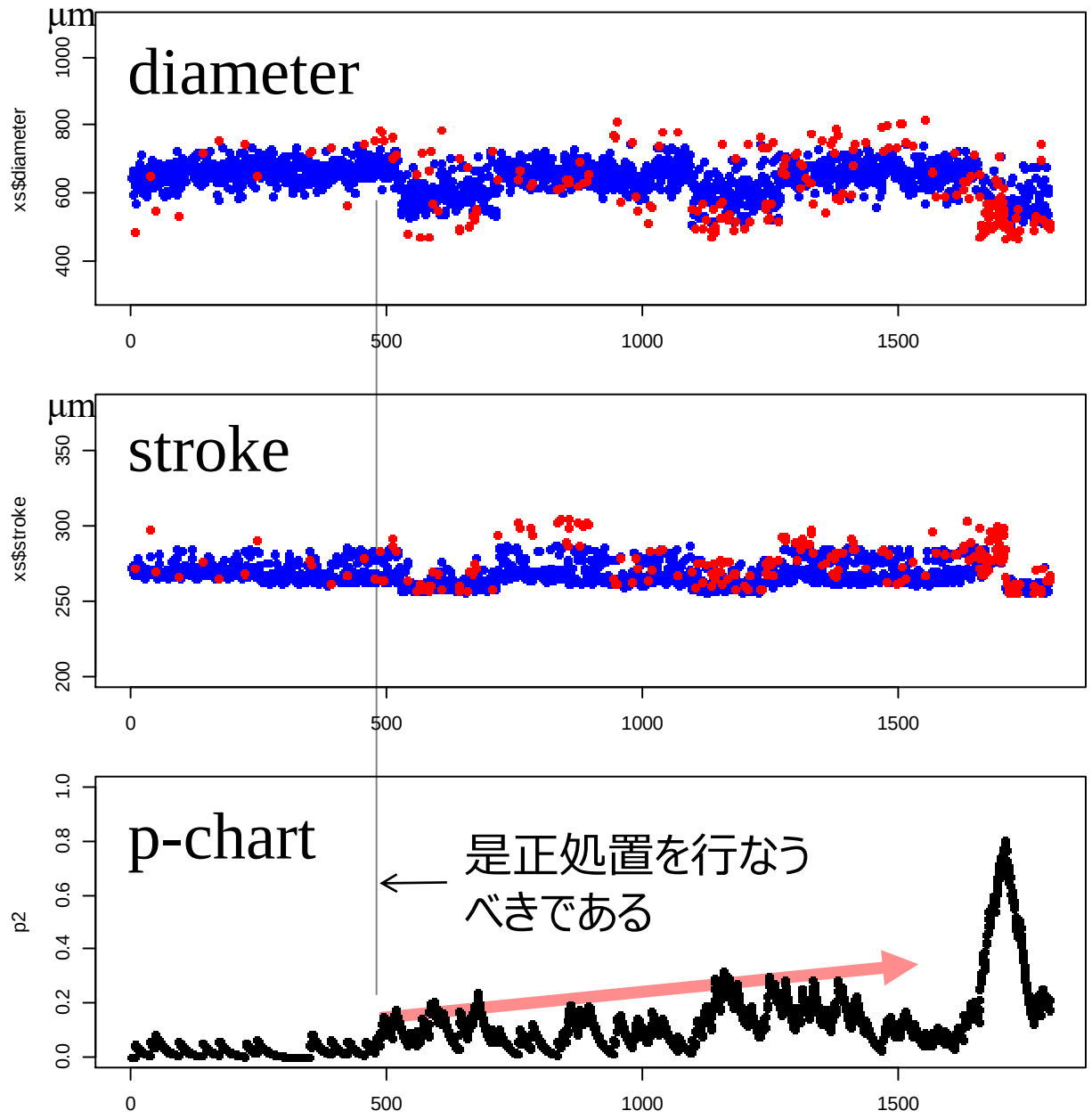
推定関数を用いて境界を線引き



正常時の
95%範囲で
線引き
(約±2σ)

5%外れて
いるときは
正常

重み付き移動Pチャート



重み付き移動平均とは

- EWMA（Exponetialy Weighted Moving Average）
  - 過去に指数減衰する重みを付けて、平均を算出する。
- 次の式で算出される。x_tは0・1データでも不適合率でも良い。
  - t=0のとき、$E_t = \lambda \cdot x_t$
  - t>1のとき、$E_t = (1-\lambda) \cdot E_{(t-1)} + \lambda \cdot x_t$
- λの値
  - λ＝0.05　にすれば、過去20点（と言うか、無限に減衰する指数曲線でその面積が20）に対する平均となる。

密度関数は、次数に限界あり

- カーネル密度関数法のプログラム。ここに限界あり。

```
library("MASS")
dens <- kde2d(xs$diameter,xs$stroke,n=50)
```

- 1辺を50分割してヒストグラム化。
  - 2次元では、2500マス目
  - 3次元では、125000マス目
  - 4次元では、625万マス目（組合せ爆発！）
  - データは1800個だったので、2次元で精一杯。

- １クラスSVMが用いられるが、実装は手間がかかる。

外れ値・変化点を使わない 状態監視

各変数間の関連性からの逸脱

線形制約からの逸脱を監視する

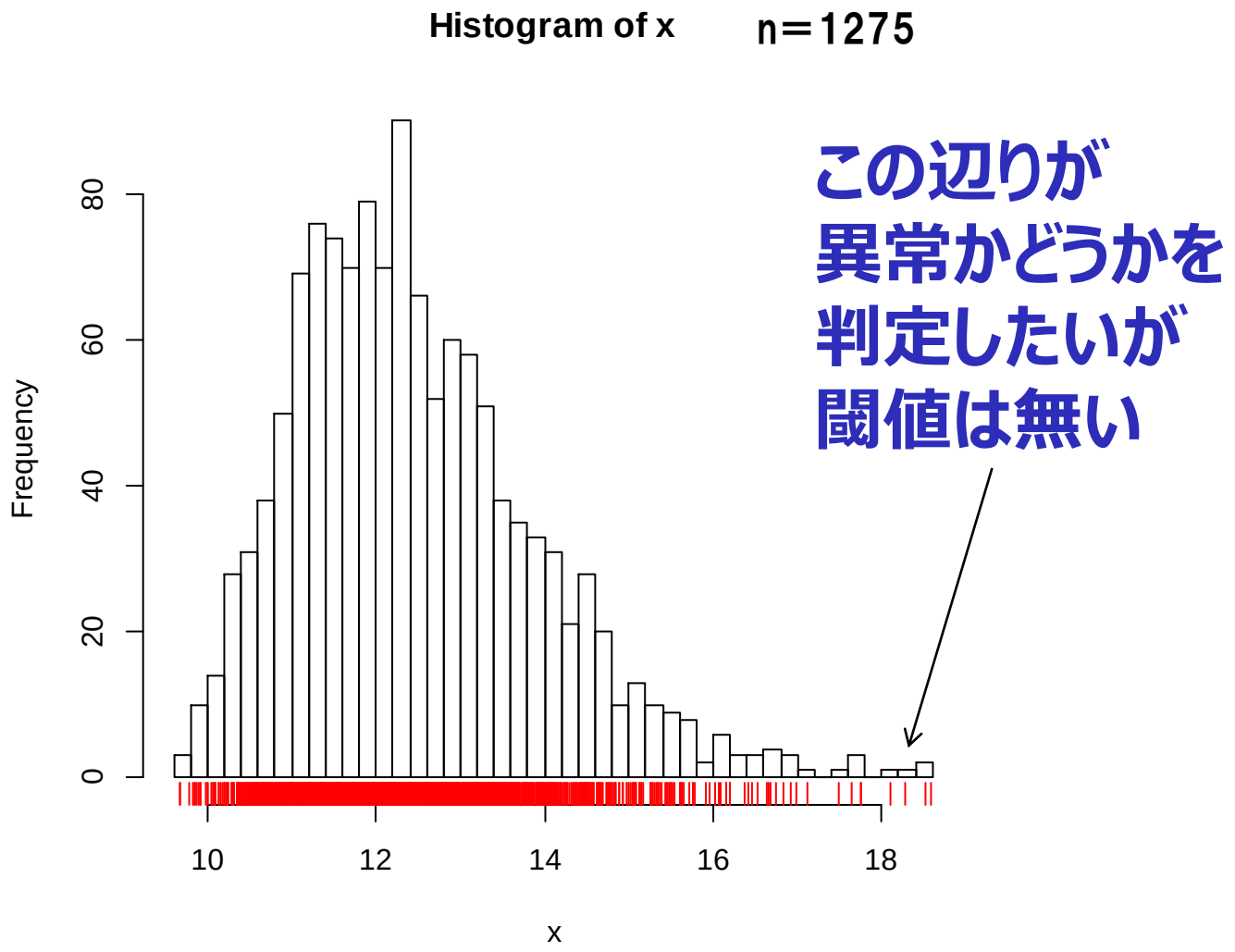
空間の歪を監視する

期待確率からの逸脱監視

監視対象がQQプロットの信頼限界の外に出る

ロット全滅ではなく離れ小島出現

- 「集団外れ値」が出始めるのを監視したい。



ジョンソン変換にて正規分布へ

【うれしさ】

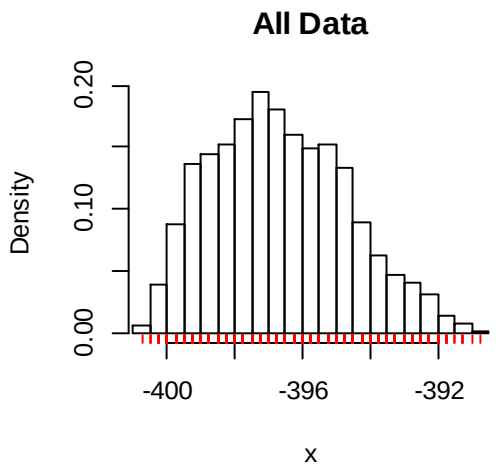
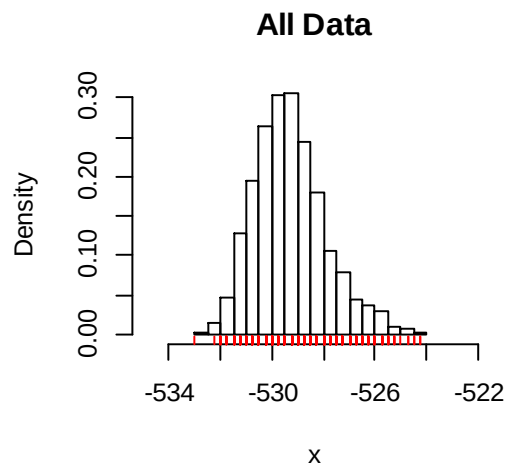
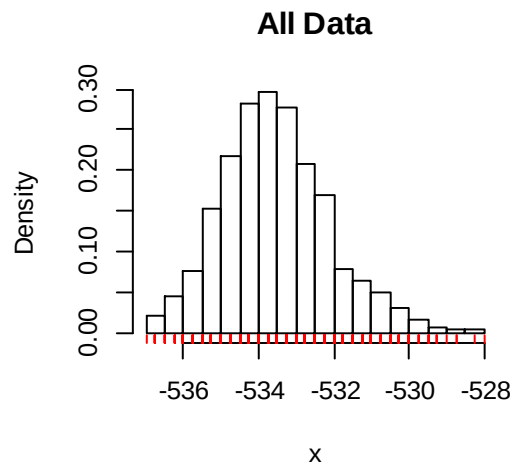
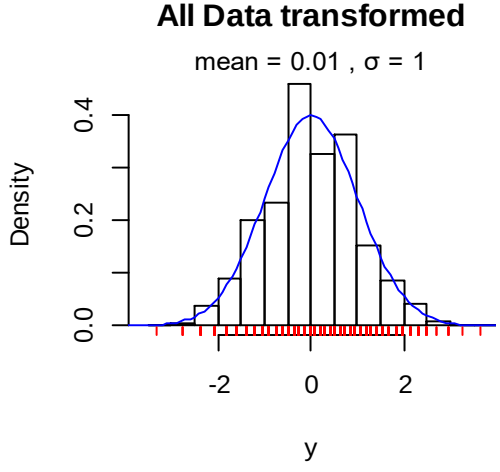
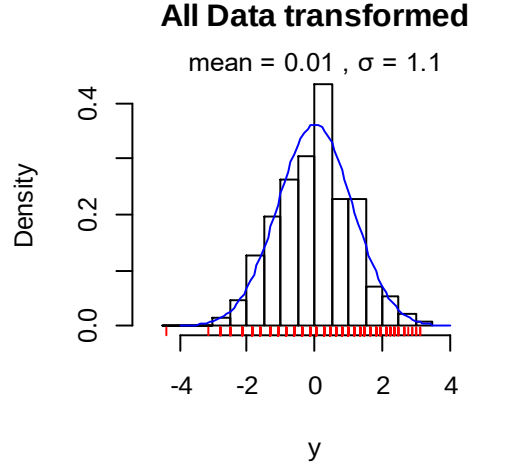
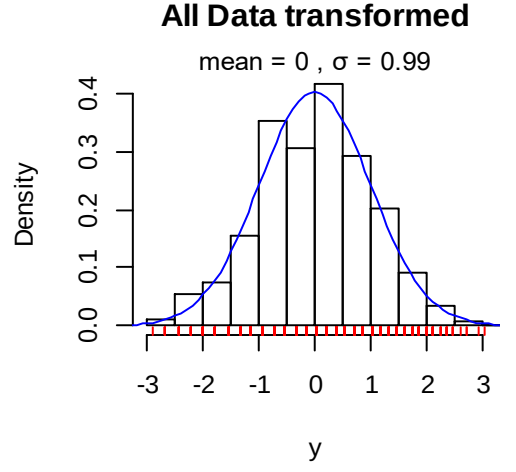
- 負値を含んでも可能
- 3つの変換から一番フィットするものを自動選択

Type	変換式
SB	y <- g + e * log((x-p)/(l+p-x))
SL	y <- g + e * log((x-p)/l)
SU	y <- g + e * asinh((x-p)/l)

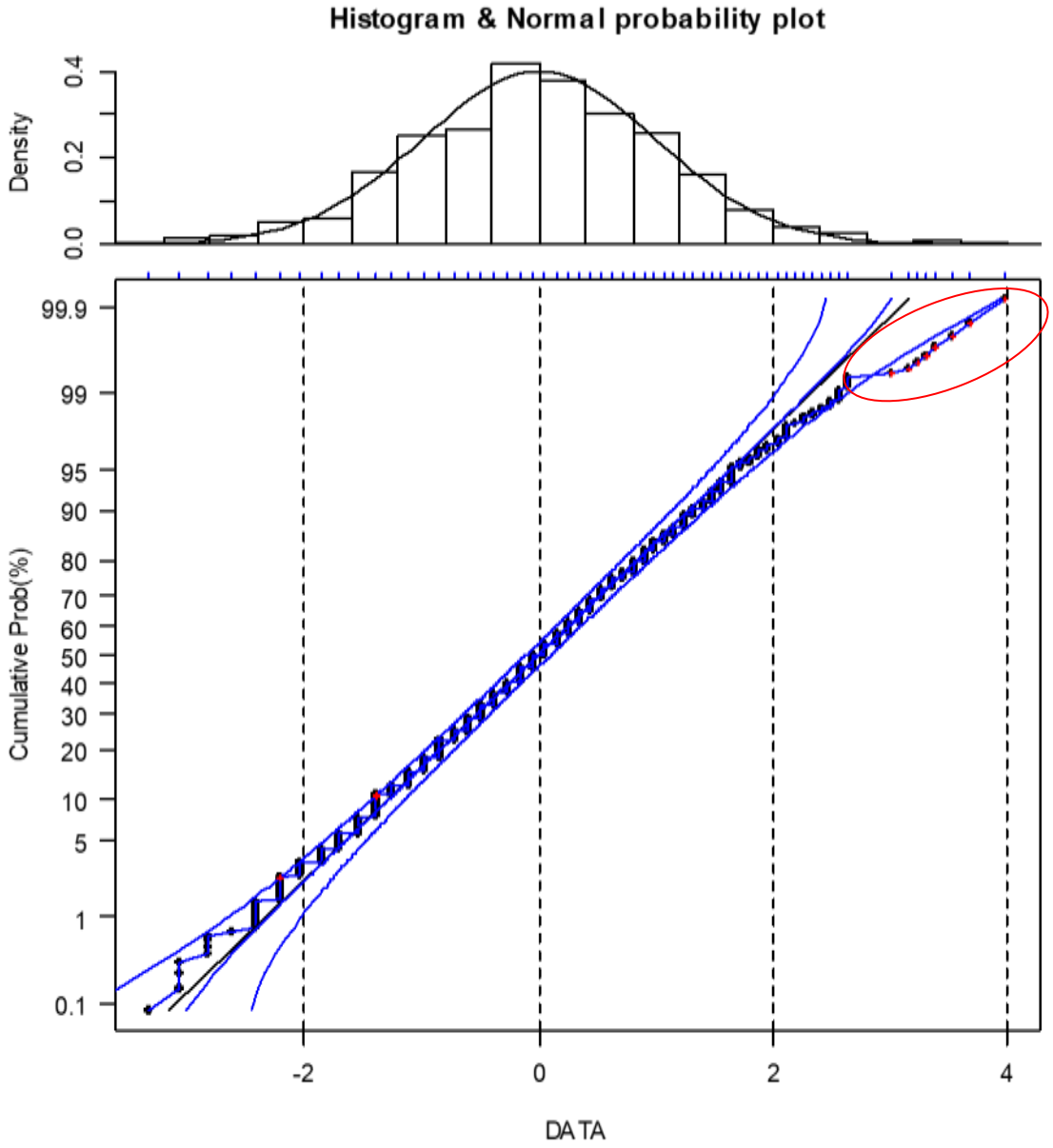
【弱点】

- 少量データや量子化(飛び飛びの)データは、変換できない。
  - jitter で僅かな誤差を加えて連続化する
  - データ増量は、ブートストラップ・オーバー・サンプリングなど

ジョンソン変換の能力

	SB	SL	SU
変換前			
変換後			

正規確率プロットで判定



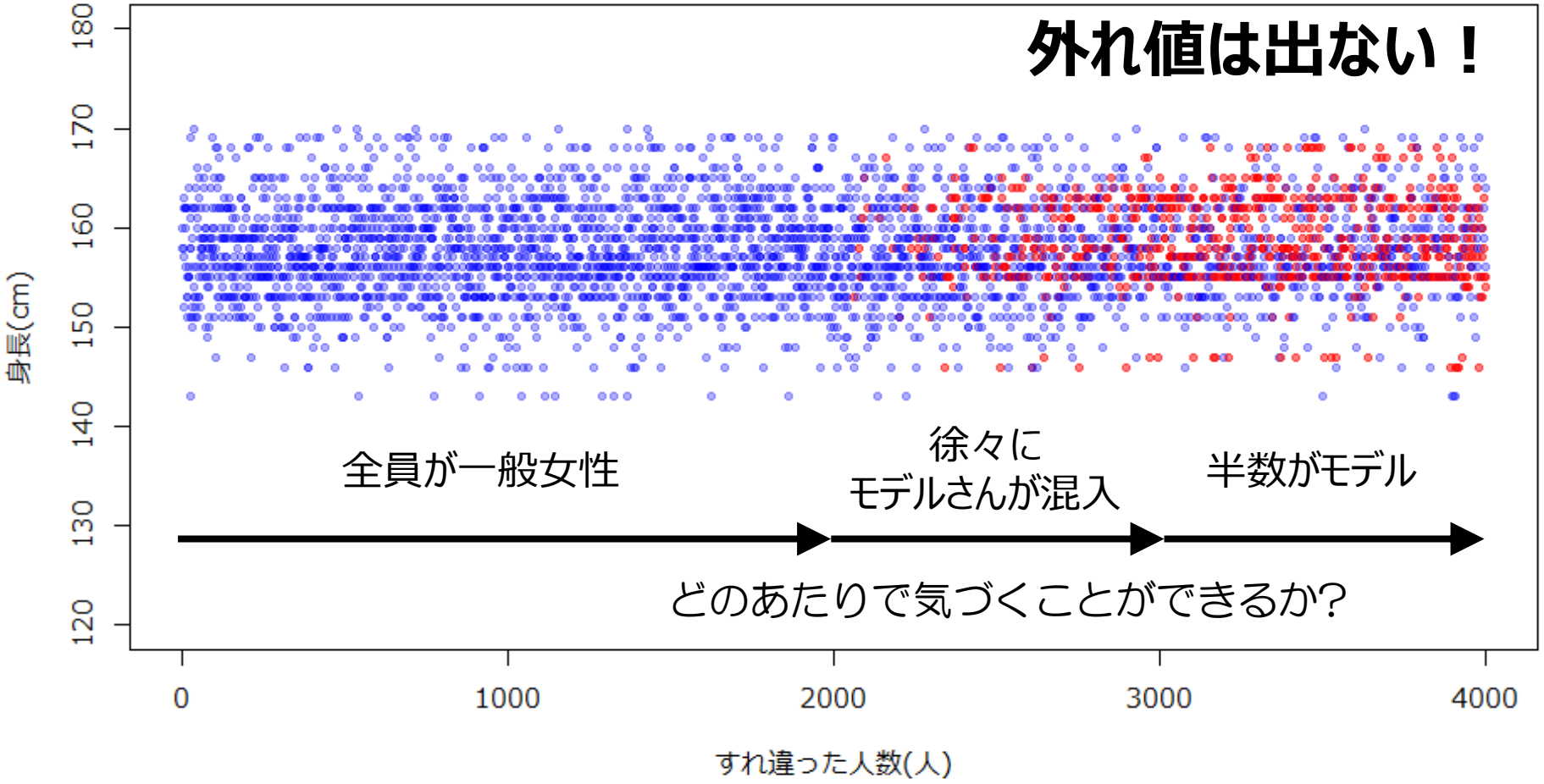
期待確率からの逸脱を監視すれば良い

分布の歪を使った異常検知

外れ値は期待確率からの逸脱監視で検出できる。
では、分布の中に『別物』が出現してくるときは
監視が可能か。静的＋動的のあいのこ。

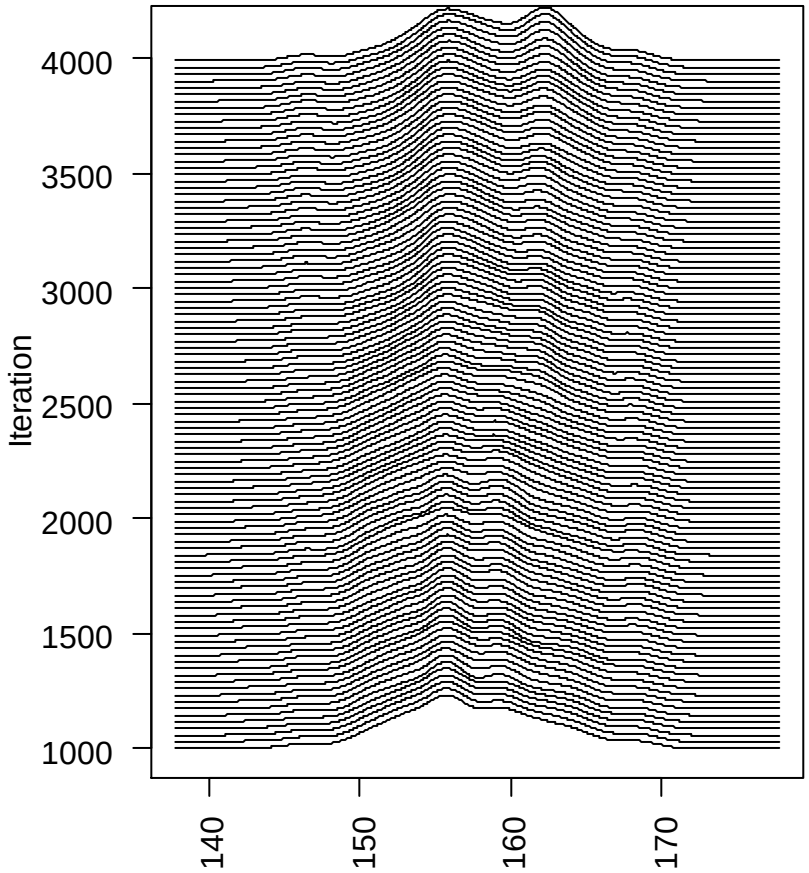
分布の内部で異常が発生！

- 工程データの異常は、どこから始まっていますか?



データ分布にはこんな変化が！

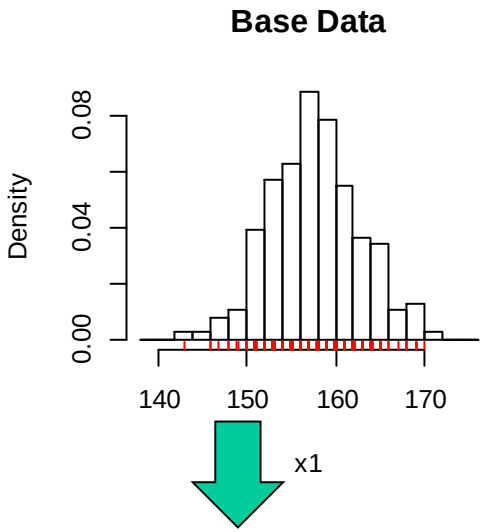
一山だった分布が、二山に変化！

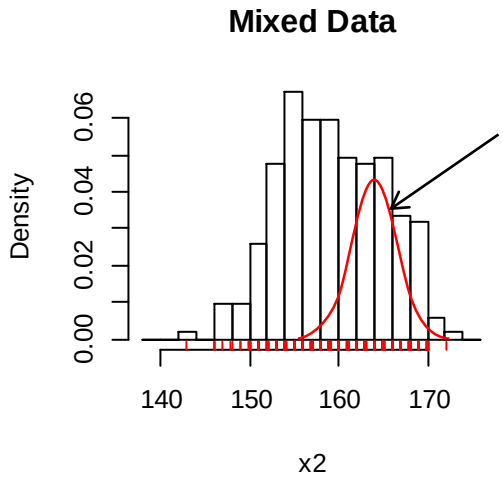


このようなケースでも・・・

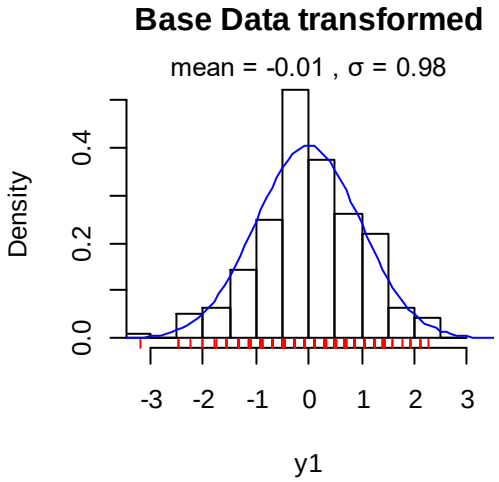
- 一般女性200人の身長を正常データとし（クリーニングあり）、
- ファッションモデルの身長をランダムに混入させ、プロットを観察。

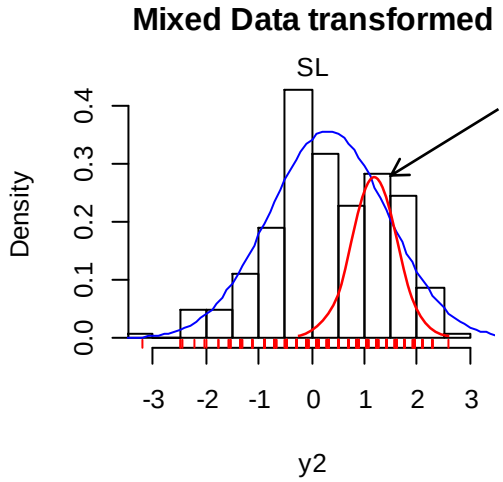
教師データの
正規分布への
Johnson変換の
式を記憶する





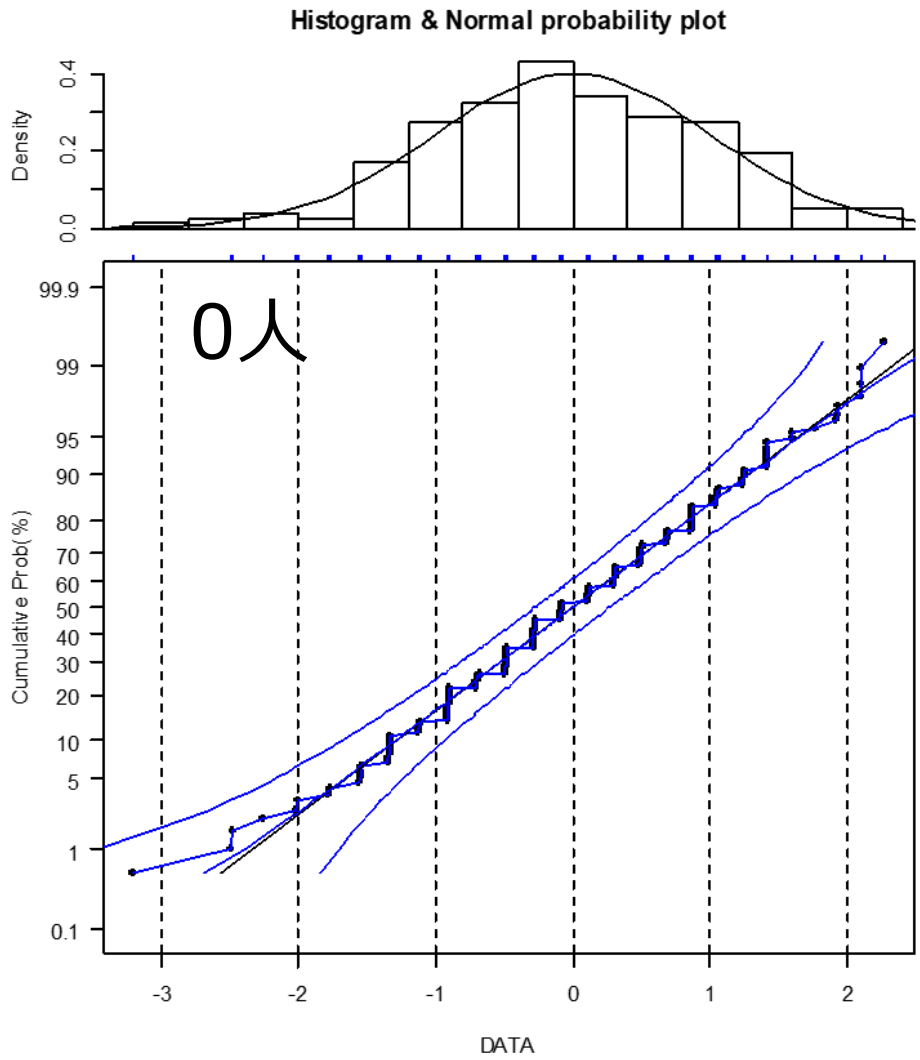
これらが徐々に
混入していく

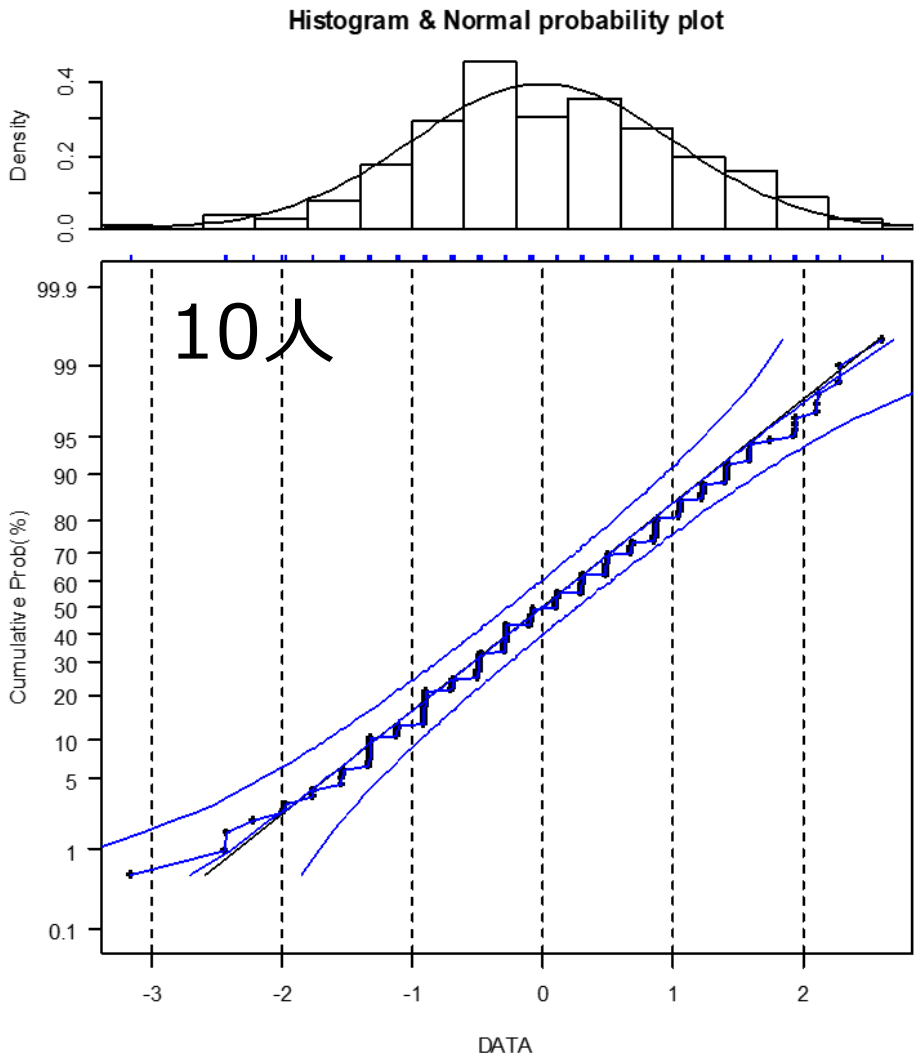




50人混入すると
明らかに分かる

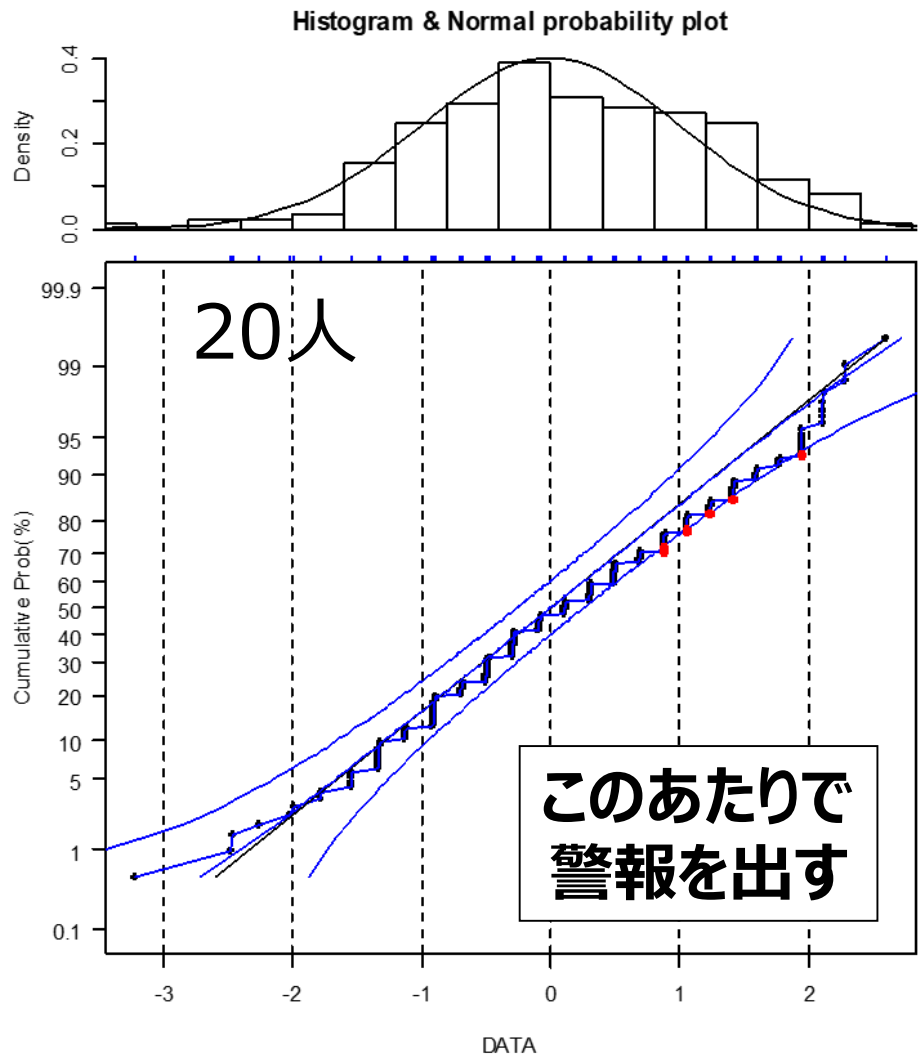
期待確率からの逸脱を見ると

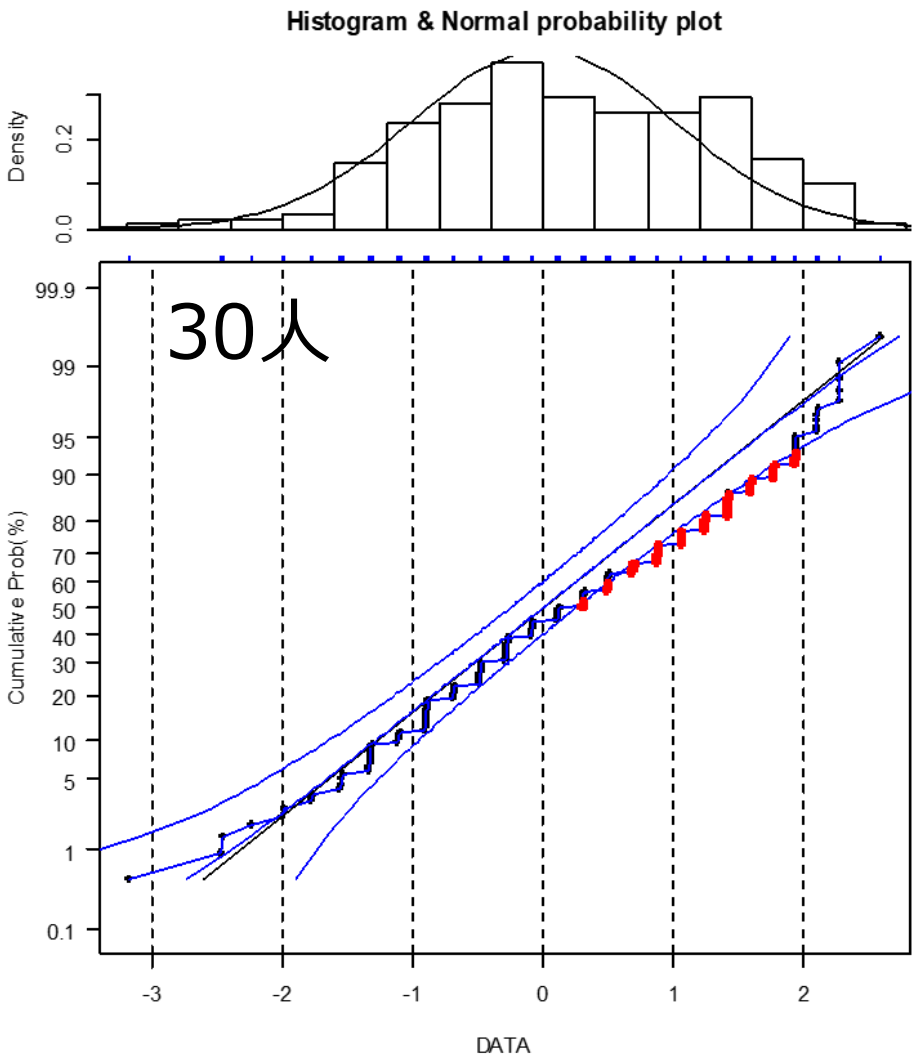




200人中10人の混入は、検出できなかったが・・・

さらに混入が進むと



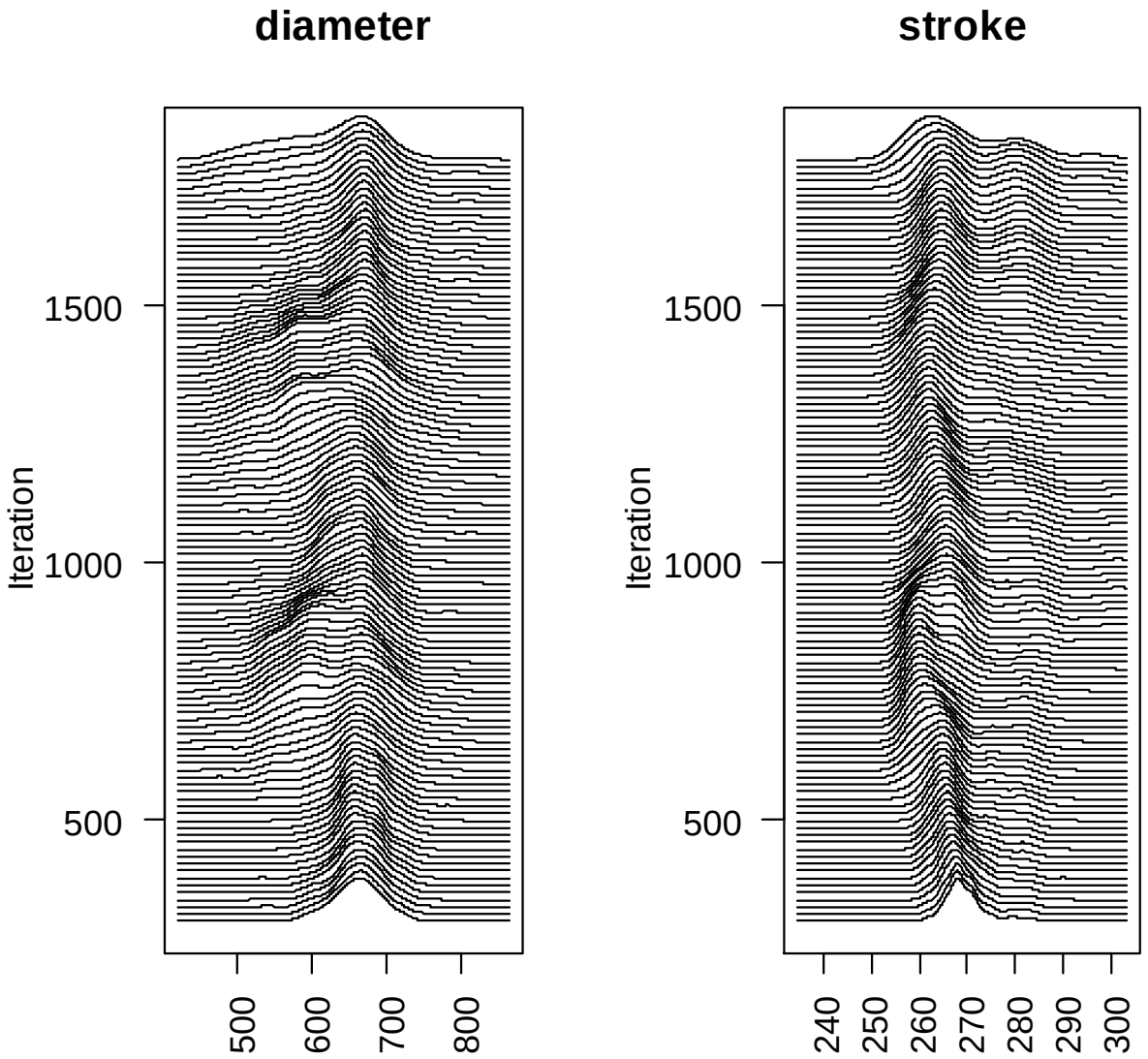


１台生産の都度、窓を移動しながら連続監視すればよい

空間歪を使った異常検知

自律適応制御を導入すると、
異常値は出なくなり、排他識別が効かない。
そのとき、多次元の空間異常をどう検出するか

マシニング・センタのデータ分布



空間情報を使わないと異常は発見できない。１次元じゃダメ。

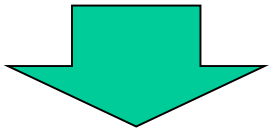
方法

- **直線ならぬ一様分布空間に写像して一様性を判定**

①ジョンソン変換（正規分布に）

②正規直交系へ写像（相関の排除）

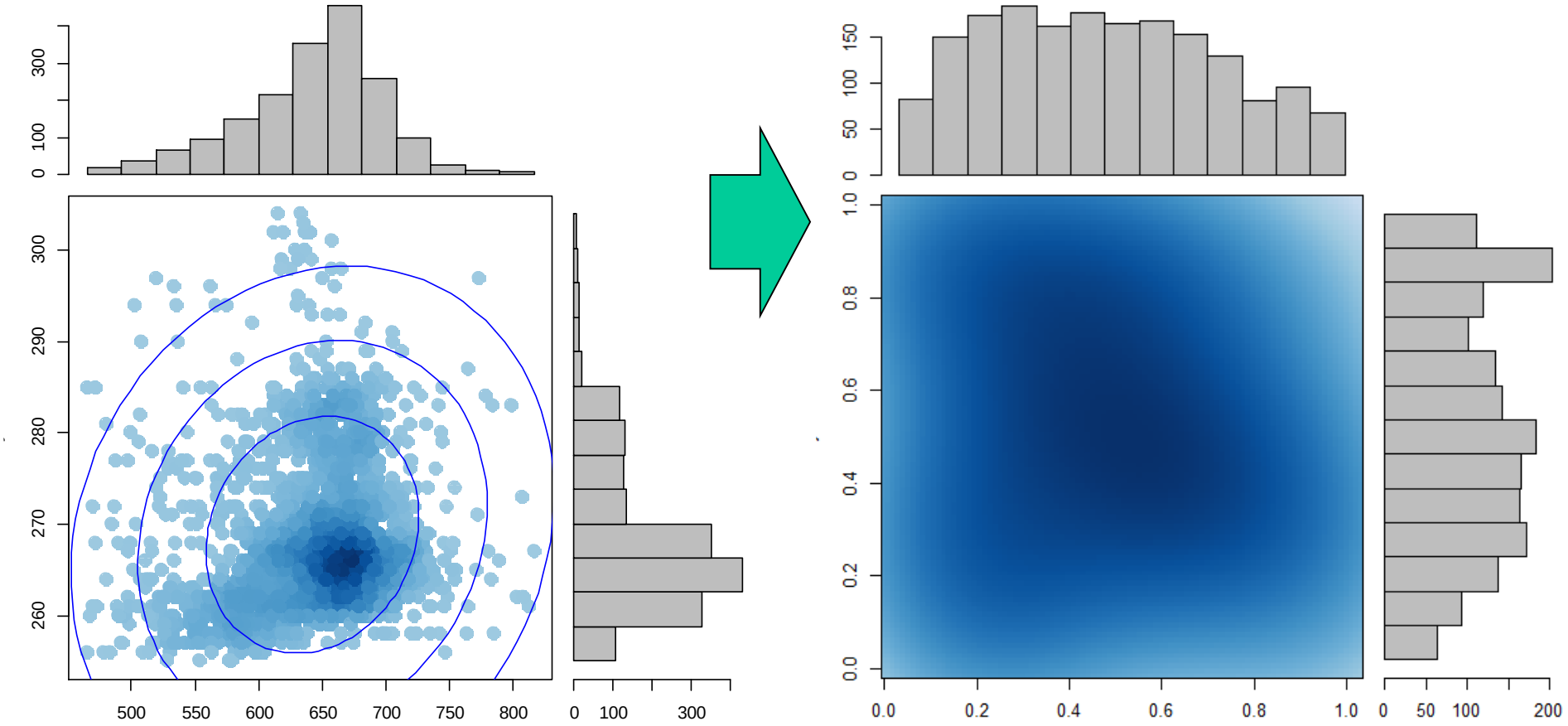
③シグモイド関数で変換（一様分布に）



ディスクレパンシで一様性を判定
(一様に変換した基準空間と比較)

一様分布空間へ写像

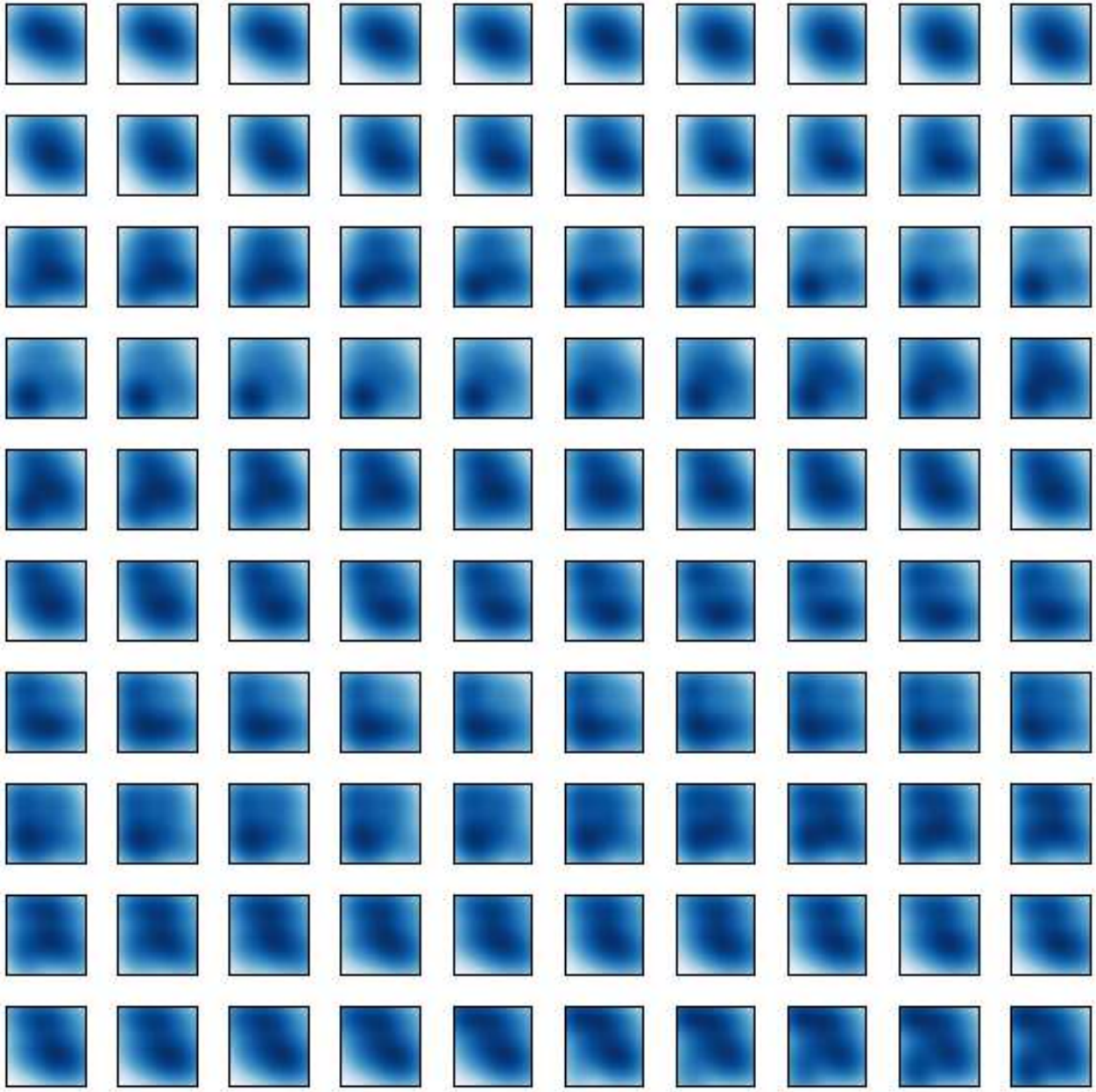
・ 前出のマシニング･センタの例



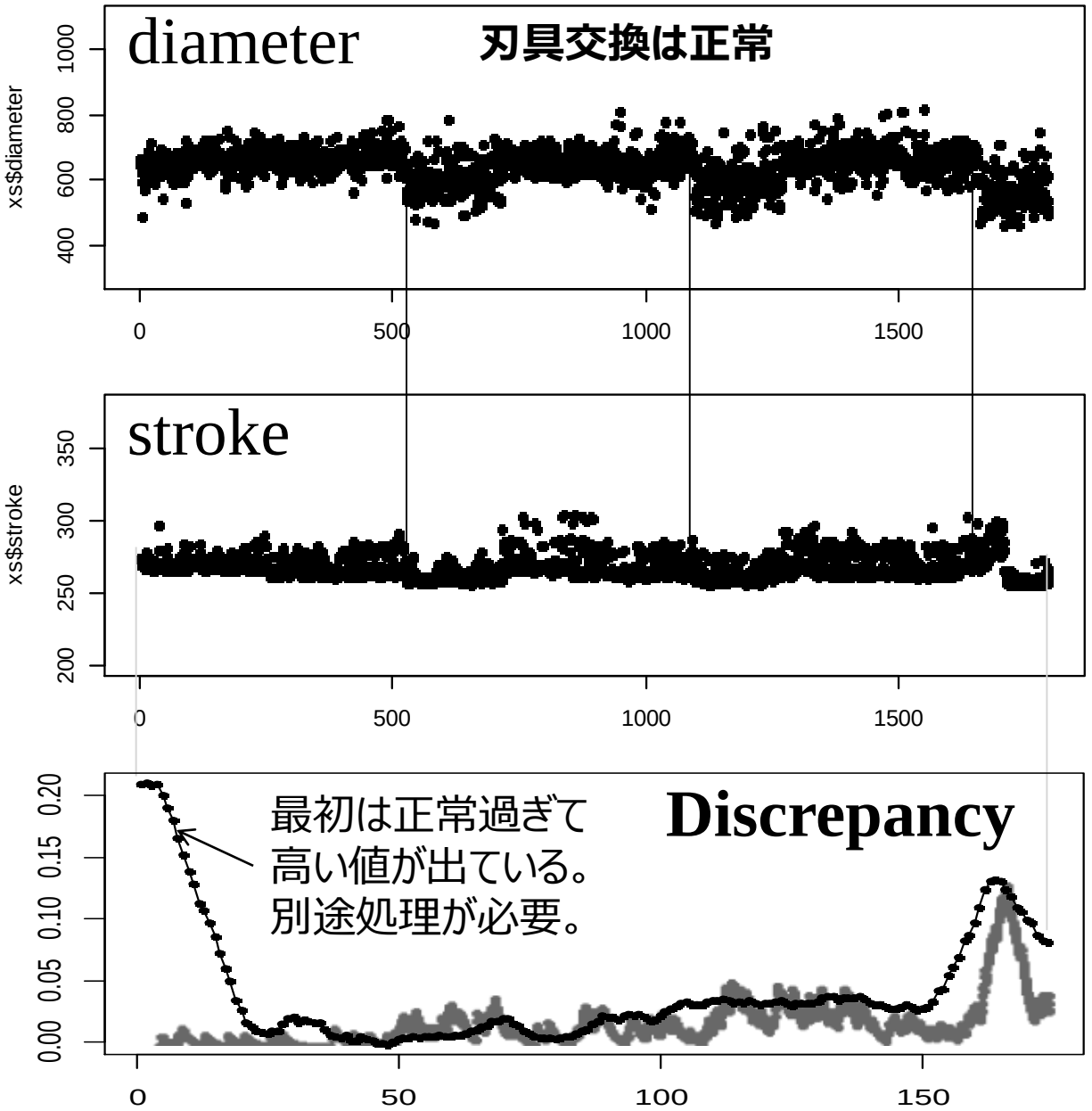
歪があると一様化しても濃淡が出るハズ

空間歪の時系列変化 事例

- 前ページの密度を基準とし、それからの乖離度を測る。
- ディスクレパンシでスカラー化



良く似た検出をしている



まとめ

データサイエンスと品質管理

- 品質管理の考え方は従来とは変わらない。
  - 「アウトライアやアノマリ」「変化点」を機に起動する。
  - 「検知」「是正」「維持」の活動である。

- 状態監視とは、自覚症状を用いない「健康診断」。
  - 近年、「外れ値」「変化点」を用いない方法が出てきた。
  - それらは「線形制約からの逸脱」「空間歪」を監視している。

- ビッグデータ化に対応して、データサイエンスが活用される。
  - 「予期せぬ線形制約」に対して、正則化を用いる。
  - 次元削減（Sufficient Dimension Reduction）をしてから状態監視に掛ける。DBSCAN，UMAP，SIRなど。

ご清聴ありがとうございました