Generative Topographic Mapping
GTM

明治大学 理工学部 応用化学科
データ化学工学研究室　金子 弘昌

GTM とは？

- ✓データを可視化・見える化するための非線形手法
- ✓主成分分析などとは異なり、はじめに二次元平面の座標を作ってしまい、それを実際の多次元空間のサンプルに合わせ込むというスタンス
- ✓ゴム状のシート (二次元平面) を曲げたり伸び縮みさせたりしながら、多次元空間にあるサンプルを通るようにシートを置き、そのシートにサンプルを射影するような手法
- ✓自己組織化マップ (Self-Organizing Map, SOM) のいろいろな問題点を解決した、上位互換の手法
- ✓ハイパーパラメータの数が多いため、設定の際には注意が必要
- ✓２次元平面において近いところにあるサンプル同士は、多次元空間においても近い

GTMで解決できたSOMの問題点

✓元の多次元空間においてサンプル同士が似ているからといって、SOMの2次元平面上でも似ているとは限らない

✓二次元平面への可視化結果の良し悪しを評価するための関数(評価関数 or コスト関数) がない

✓学習回数を大きくしたとしても、計算が収束するとは限らない

✓二次元平面への可視化結果として、SOMにおいて発火したニューロンのどれか、だけあり確率で表わされるわけではない

✓学習率や近傍関数をどのように設定すればよいかわからない

GTMの大まかな流れ

✓① 二次元平面のサイズを決める
- 10×10とか、20×20とか

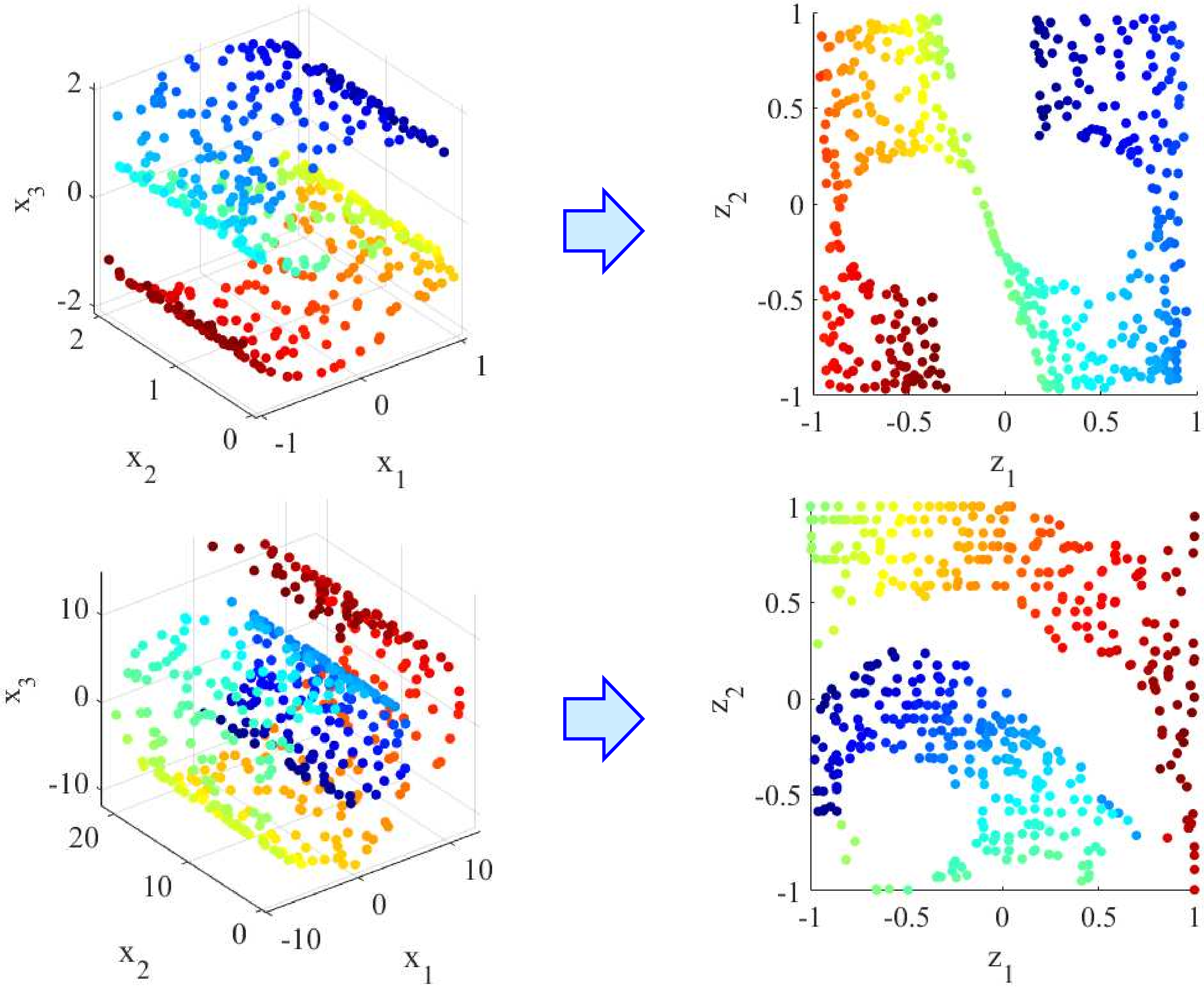
✓② 二次元平面から多次元空間へ変換する関数を決める

✓③ 多次元空間におけるデータセットになるべく当てはまるように
②の関数のパラメータを最適化する

✓④ サンプルごとの、二次元平面上の各グリッド (格子点) に存在する
確率を求める

✓⑤ 確率から、二次元平面上の位置を決める

GTMによる可視化の例



こんなデータセットがあるとする

		変数						
		1	2	$\cdots$	i	$\cdots$	m-1	m
	1							
	2							
	$\vdots$							
サンプル	j				$x_i^{(j)}$			
	$\vdots$							
	n-1							
	n							

１つのサンプル、全サンプル

✓ １つのサンプルを $\mathbf{x}^{(i)}$、

✓全サンプルを $\mathbf{X}$

とする

GTMを誤解なく理解するための発想の転換

✓GTMは一見、多次元 (m次元) → 2次元 の低次元化手法と思われがち

✓発想を転換して、多次元 (m次元) → [グリッド数] 次元 への変換手法と考えたほうが、GTMを理解しやすい

- たとえば、二次元平面のサイズが 10×10 のとき、[グリッド数] ＝ 100

① 二次元平面のサイズを決める

✓縦のサイズと横のサイズは同じにすることが多い

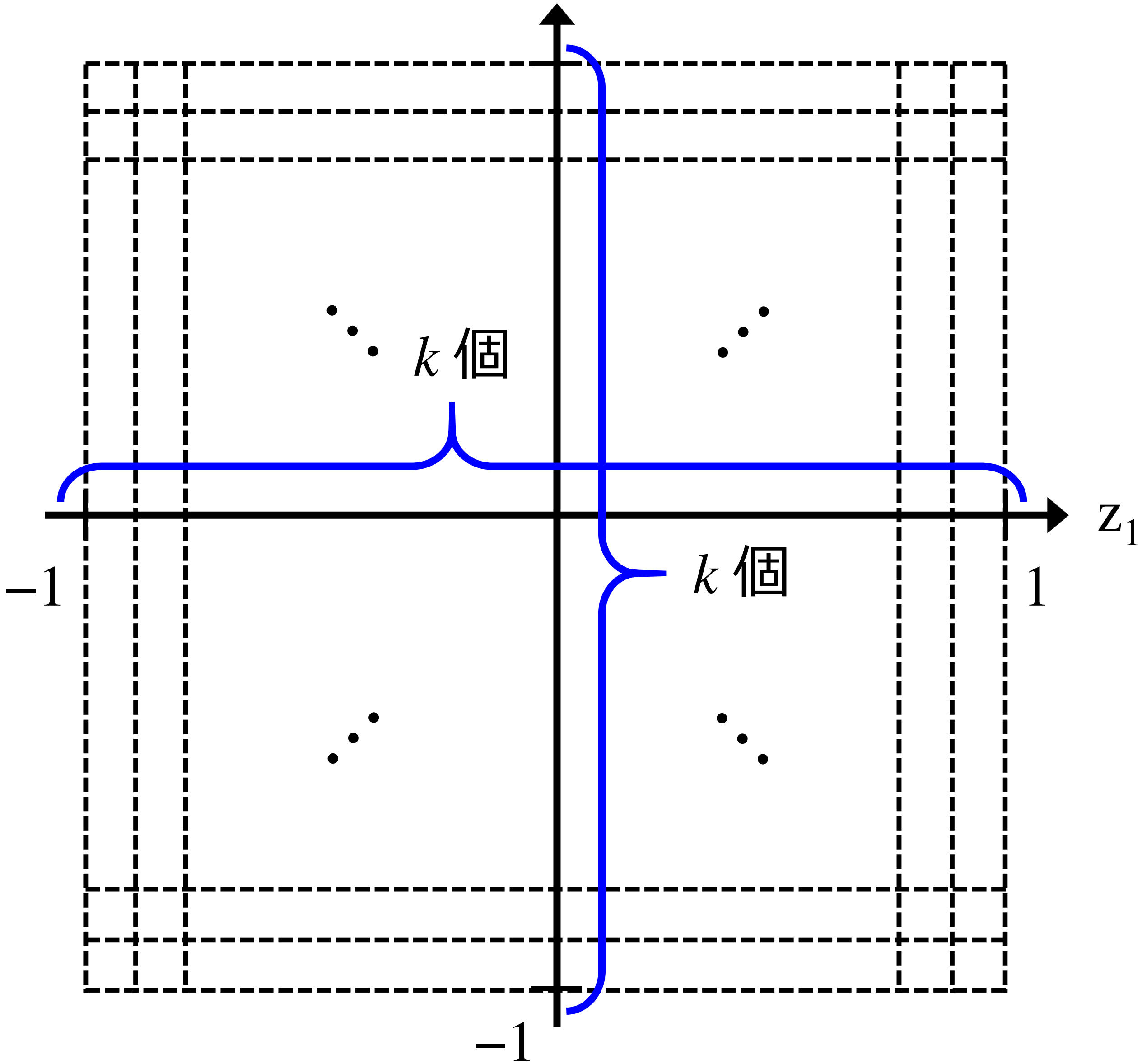
- $k \times k$ とする

✓縦も横も、-1 から 1 の間でグリッド (格子点) を決める

✓k を大きめに設定したとしても、計算時間が多めにかかるだけで、オーバーフィッティングへの影響はほとんどない

- のちに設定する基底関数の数のほうが、オーバーフィッティングへの関係が強い

① 二次元平面



① グリッド (格子点) の座標

✓グリッド (格子点) の座標を

$$\mathbf{Z} = \begin{bmatrix} z_1^{(1)} & z_2^{(1)} \\ \vdots & \vdots \\ z_1^{(j)} & z_2^{(j)} \\ \vdots & \vdots \\ z_1^{\left(k^2\right)} & z_2^{\left(k^2\right)} \end{bmatrix}$$

とする

✓また、 $\mathbf{z}^{(j)} = \begin{bmatrix} z_1^{(j)} & z_2^{(j)} \end{bmatrix}$ とする

② 二次元 → 多次元 の変換

✓二次元から多次元へ変換する非線形関数を f とする

$$\mathbf{x}^{(i)} = f\left(\mathbf{z}^{(j)}\right)$$

② 基底関数

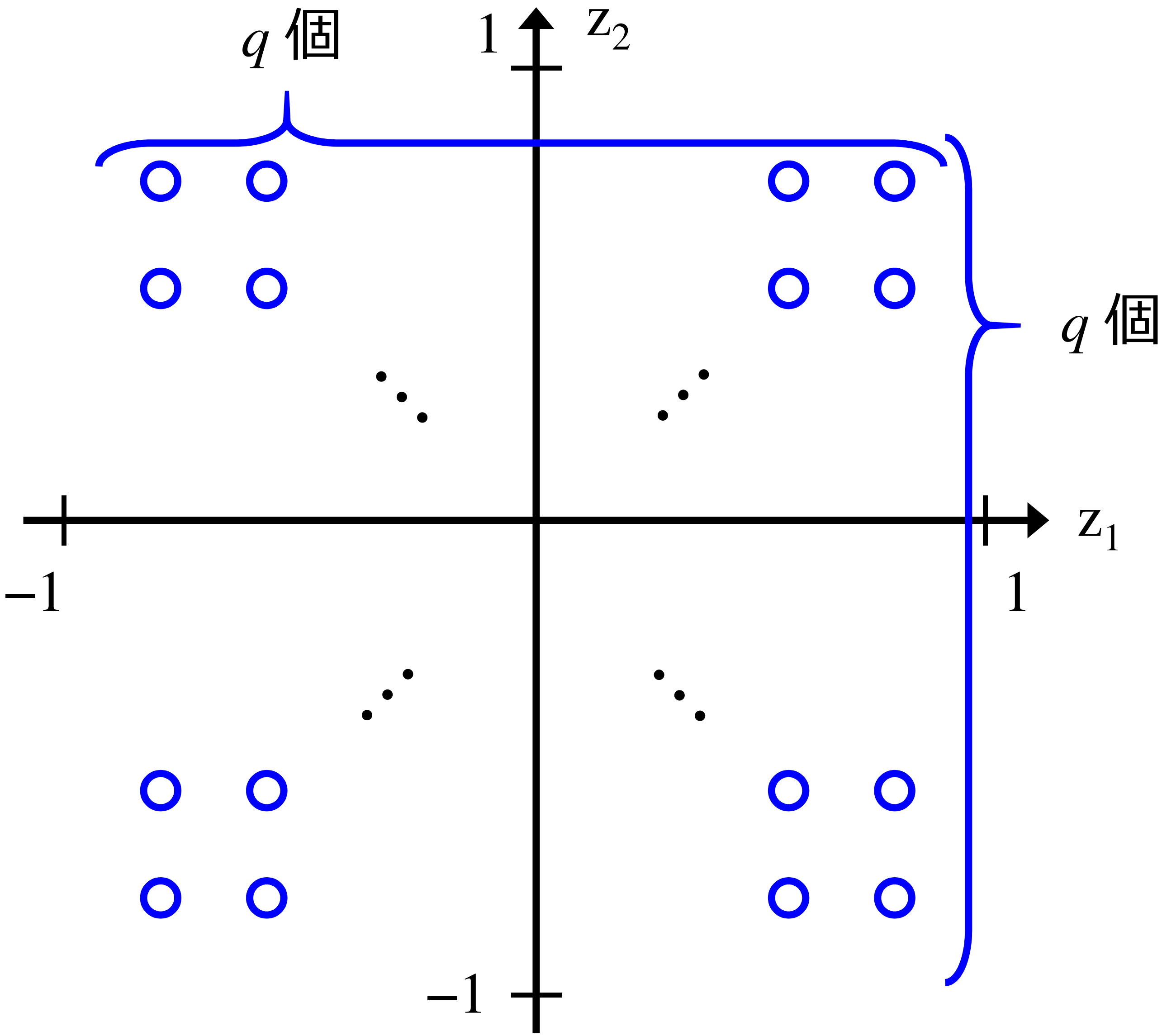
✓f を、p 個の基底関数という非線形関数の線形結合とする

✓i 番目の基底関数 $\phi_i(\mathbf{z}^{(j)})$ は、二次元平面上における、
中心 $\mathbf{t}^{(i)} = [\ t_1^{(i)}\ t_2^{(i)}]$ から同心円状に値が小さくなる、以下の
放射基底関数 (radial basis function)

$$\phi_i\left(\mathbf{z}^{(j)}\right) = \exp\left\{-\frac{1}{2\sigma^2}\left\|\mathbf{t}^{(i)} - \mathbf{z}^{(j)}\right\|^2\right\}$$

✓$p = q \times q$ として、基底関数の中心を二次元平面上にまんべんなく
置くことが多い

② 基底関数の中心の配置



② 重み W

✓f を、p 個の基底関数という非線形関数の線形結合とする

$$
\begin{aligned}
f\left(\mathbf{z}^{(j)}\right) &= \phi_1\left(\mathbf{z}^{(j)}\right)\mathbf{w}_1 + \phi_2\left(\mathbf{z}^{(j)}\right)\mathbf{w}_2 + \cdots + \phi_{q^2}\left(\mathbf{z}^{(j)}\right)\mathbf{w}_{q^2} \\
&= \Phi\left(\mathbf{z}^{(j)}\right)\mathbf{W}
\end{aligned}
$$

✓ただし、

$$\mathbf{w}_i = \begin{bmatrix} w_i^{(1)} & w_i^{(2)} & \cdots & w_i^{(m)} \end{bmatrix}$$

$$\mathbf{W} = \begin{bmatrix} \mathbf{w}_1{}^{\mathrm{T}} & \mathbf{w}_2{}^{\mathrm{T}} & \cdots & \mathbf{w}_{q^2}{}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$$

$$\Phi\left(\mathbf{z}^{(j)}\right) = \begin{bmatrix} \phi_1\left(\mathbf{z}^{(j)}\right) & \phi_2\left(\mathbf{z}^{(j)}\right) & \cdots & \phi_{q^2}\left(\mathbf{z}^{(j)}\right) \end{bmatrix}$$

② 二次元→多次元 の変換は分布をもつ

✓二次元平面の各グリッドから、多次元空間への変換は、
$f(\mathbf{z}^{(j)}) = \Phi(\mathbf{z}^{(j)})\mathbf{W}$ を中心とした正規分布

$$p\left(\mathbf{x} \,|\, \mathbf{z}^{(j)}, \mathbf{W}, \beta\right) = \left(\frac{\beta}{2\pi}\right)^{\frac{m}{2}} \exp\left\{-\frac{\beta}{2}\left\|\Phi\left(\mathbf{z}^{(j)}\right)\mathbf{W} - \mathbf{x}\right\|^2\right\}$$

↑

β : 正規分布の分散の逆数

$\mathbf{W}$、β が決まったあとに、$\mathbf{z}^{(j)}$ を入力したときの多次元空間における確率分布 (probability distribution) という意味

✓確率分布なので、値をすべて足すと 1 になる

② すべてのグリッド(格子点)からの変換

✓二次元平面におけるすべてのグリッドを多次元空間に変換する

✓多次元空間における確率分布 $p(\mathbf{x} \mid \mathbf{W}, \beta)$ は、各グリッドにおける $p(\mathbf{x} \mid \mathbf{z}^{(j)}, \mathbf{W}, \beta)$ をすべてのグリッドで足し合わせ、最後にグリッド数で割ることで与えられる

- グリッド数で割るのは、足して 1 にするため

$$p\left(\mathbf{x} \mid \mathbf{W}, \beta\right) = \frac{1}{k^2} \sum_{j=1}^{k^2} p\left(\mathbf{x} \mid \mathbf{z}^{(j)}, \mathbf{W}, \beta\right)$$

③ 最適化のための準備

✓確率分布 $p(\mathbf{x} \mid \mathbf{W}, \beta)$ が、多次元空間における実際のデータ分布を表現していればよい

✓つまり、各サンプル $\mathbf{x}^{(i)}$ の確率 $p(\mathbf{x}^{(i)} \mid \mathbf{W}, \beta)$ をすべてのサンプルでかけあわせたものが、大きければよい

↓

$$\prod_{i=1}^{n} p\left(\mathbf{x}^{(i)} \mid \mathbf{W}, \beta\right)$$

③ 尤度関数 L

- ✓ $\prod_{i=1}^{n} p\left(\mathbf{x}^{(i)} \mid \mathbf{W}, \beta\right)$ について、対数をとったものを尤度関数 L とする

- ✓ L は W と β の関数

$$L(\mathbf{W}, \beta) = \sum_{i=1}^{n} \ln\left\{p\left(\mathbf{x}^{(i)} \mid \mathbf{W}, \beta\right)\right\}$$

$$= \sum_{i=1}^{n} \ln\left\{\frac{1}{k^2}\sum_{j=1}^{k^2} p\left(\mathbf{x}^{(i)} \mid \mathbf{z}^{(j)}, \mathbf{W}, \beta\right)\right\}$$

$$= \sum_{i=1}^{n} \ln\left\{\frac{1}{k^2}\left(\frac{\beta}{2\pi}\right)^{\frac{m}{2}} \sum_{j=1}^{k^2} \exp\left\{-\frac{\beta}{2}\left\|\Phi\left(\mathbf{z}^{(j)}\right)\mathbf{W} - \mathbf{x}^{(i)}\right\|^2\right\}\right\}$$

- ✓ L の最大化を試みる

③ EMアルゴリズム

✓一般的には、L を $\mathbf{W}$ と β で微分して、勾配法・ニュートン法などで L を最大化し、そのときの $\mathbf{W}$ や β を求める

✓今回は微分が難しい

✓ L の最大化を、$\mathbf{W}$ と β が不明という不完全データ問題としてとらえ、EM (Expectation–Maximization) アルゴリズムを用いて L が最大となる $\mathbf{W}$ と β を求める

- EステップとMステップとを繰り返して $\mathbf{W}$ と β を求める
  - Eステップ：今の $\mathbf{W}$ と β のもとで、尤度関数の条件付き確率に関する期待値を計算
  - Mステップ：期待値を最大化する $\mathbf{W}$ と β を計算

EMアルゴリズム
https://ja.wikipedia.org/wiki/EM%E3%82%A2%E3%83%AB%E3%82%B4%E3%83%AA%E3%82%BA%E3%83%A0

③ Responsibility (R)

✓Responsibility

- あるサンプル $\mathbf{x}^{(i)}$ に対応する あるグリッド $\mathbf{z}^{(j)}$ の確率
- Eステップにおける条件付き確率
  - ベイズの定理により計算

$$R_{\mathbf{x}^{(i)},\mathbf{z}^{(j)}}\left(\mathbf{W},\beta\right) = p\left(\mathbf{z}^{(j)} \mid \mathbf{x}^{(i)},\mathbf{W},\beta\right)$$

$$= \frac{p\left(\mathbf{x}^{(i)} \mid \mathbf{z}^{(j)},\mathbf{W},\beta\right)}{\sum_{r=1}^{k^2} p\left(\mathbf{x}^{(i)} \mid \mathbf{z}^{(r)},\mathbf{W},\beta\right)}$$

③ Mステップで最大化する関数 $\mathrm{L}_{\mathrm{comp}}$

✓EMアルゴリズムにおいて繰り返し計算 a 回目の $\mathbf{W}, \beta$ をそれぞれ $\mathbf{W}_a, \beta_a$ とする

$$R_{\mathbf{x}^{(i)},\mathbf{z}^{(j)}}\left(\mathbf{W}_a,\beta_a\right)=\frac{p\left(\mathbf{x}^{(i)}\mid\mathbf{z}^{(j)},\mathbf{W}_a,\beta_a\right)}{\sum_{r=1}^{k^2}p\left(\mathbf{x}^{(i)}\mid\mathbf{z}^{(r)},\mathbf{W}_a,\beta_a\right)}$$

✓Mステップにおいて、W, β について最大化する関数 $L_{\mathrm{comp}}(\mathbf{W},\beta)$ は、

$$L_{\mathrm{comp}}\left(\mathbf{W},\beta\right)=\sum_{i=1}^{n}\sum_{j=1}^{k^2}R_{\mathbf{x}^{(i)},\mathbf{z}^{(j)}}\left(\mathbf{W}_a,\beta_a\right)\ln\left\{p\left(\mathbf{x}^{(i)}\mid\mathbf{z}^{(j)},\mathbf{W},\beta\right)\right\}$$

③ L$_{comp}$を最大化させるW

✓$L_{\mathrm{comp}}(\mathbf{W}, \beta)$ を $\mathbf{W}$ で微分して、0 とする

✓そのときの $\mathbf{W}$ を $\mathbf{W}_{a+1}$ とする

$$\sum_{i=1}^{n}\sum_{j=1}^{k^2} R_{\mathbf{x}^{(i)},\mathbf{z}^{(j)}}\left(\mathbf{W}_a, \beta_a\right)\left\{\Phi\left(\mathbf{z}^{(j)}\right)\mathbf{W}_{a+1} - \mathbf{x}^{(i)}\right\}\Phi^{\mathrm{T}}\left(\mathbf{z}^{(j)}\right) = 0$$

✓これを満たす $\mathbf{W}_{a+1}$ を計算

③ L_{comp}を最大化させる β

✓$L_{\mathrm{comp}}(\mathbf{W}, \beta)$ を β で微分して、0 とする

✓そのときの β を β_{a+1} とする

$$\frac{1}{\beta_{a+1}} = \frac{1}{mn} \sum_{i=1}^{n} \sum_{j=1}^{k^2} R_{\mathbf{x}^{(i)},\mathbf{z}^{(j)}}(\mathbf{W}_a, \beta_a) \left\| \Phi\left(\mathbf{z}^{(j)}\right) \mathbf{W}_{a+1} - \mathbf{x}^{(i)} \right\|^2$$

✓これを満たす β_{a+1} を計算

③ W の大きさに制約

✓正則化係数 λ を導入

$$p(\mathbf{W} \mid \lambda) = \left(\frac{\lambda}{2\pi} \right)^{\frac{k^2 m}{2}} \exp\left\{ -\frac{\lambda}{2} \sum_{i=1}^{m} \sum_{j=1}^{k^2} w_j^{(i)} \right\}$$

③ $\mathbf{W}_1$ と β_1

✓主成分分析 (Principal Component Analysis, PCA) を用いて $\mathbf{W}$ と β との初期値 $\mathbf{W}_1, \beta_1$

✓下の式が最小になるような $\mathbf{W}_1$ を計算

$$\frac{1}{2}\sum_{i=1}^{k^2}\left\|\Phi\left(\mathbf{z}^{(i)}\right)\mathbf{W}_1-\mathbf{z}^{(i)}\mathbf{U}\right\|^2$$

$\mathbf{U}$：最初の固有ベクトルと二番目の固有ベクトル

✓β_1 は三番目の固有値を逆数にしたもの

③ W と β の計算

✓p.24 で $\mathbf{W}_1, \beta_1$ を計算

✓下の３つを、尤度関数が収束するまで繰り返す

- p.20 の Responsibility (R) の計算
- p.21 の $\mathbf{W}_a, \beta_a \rightarrow \mathbf{W}_{a+1}$ の計算
- p.22 の $\mathbf{W}_a, \beta_a \rightarrow \beta_{a+1}$ の計算

④ 二次元平面上での確率

✓サンプルごとの、二次元平面上の各グリッド (格子点) に存在する確率は、p.19のResponsibility (R) によって計算できる

✓サンプルごとに、全グリッドの R を足すと 1 になる

⑤ 二次元平面上の位置

✓サンプルごとに、確率分布から二次元平面上の位置を決める方法が主に２通りある

- Mode (最頻値): すべてのグリッドの中で R が最大の位置
- Mean (平均値): Rの値を重みとした重み付き平均

GTMのハイパーパラメータのその意味合い

✓マップサイズ：k

- 大きめに設定したとしても、計算時間が多めにかかるだけで、オーバーフィッティングへの影響はほとんどない

✓基底関数の数：q

- 基底関数を多くすると、柔軟に可視化・見える化できるが、オーバーフィッティングの危険も高くなる

✓基底関数の標準偏差：σ

- 大きいほど滑らかな二次元平面になる
- データセットが多次元空間に
  - まんべんなく広がっているときは、大きい方がよい
  - 局所的にいくつか かたまって存在していれば、小さい方がよい

✓重み $\mathbf{W}$ の大きさを決めるパラメータ：λ

- 0 からスタートし、問題があれば、0.0001, 0.001, …と、少しずつ大きくする

GTMのハイパーパラメータの最適化の方法

✓クロスバリデーションの”誤差”が最小になるパラメータの組み合わせ

✓サンプルごとの、最もユークリッド距離の小さい k 個のサンプルとの中点をテストサンプルとしてあつかい、テストサンプルの”誤差”が最小となるパラメータの組み合わせ

- ”誤差”とは、多次元空間上のサンプル点と、二次元平面に写像したあとに、もう一度多次元空間に戻した点との RMSE (Root-Mean-Squared Error)

逆写像

- ✓GTMにより、あるサンプルを二次元平面上に写像できる
- ✓二次元平面上に写像された点を、元の空間に戻してあげることを逆写像という
- ✓元のサンプル点と逆写像された点との距離を見ることで、サンプル点が二次元平面とどれくらい近いかが分かる
- ✓離れているサンプルは、適切に写像されていない、外れ値である、などの議論ができる

逆写像のしかた

✓ある $\mathbf{z}^{(j)}$ に対して、$\Phi\left(\mathbf{z}^{(j)}\right)\mathbf{W}$ により多次元平面上に逆写像できる

✓Responsibility (R) を考慮した逆写像

- あるサンプルに対して、p.20 の Responsibility (R) を計算する
- j 番目のグリッド点に対応する R を $\Phi\left(\mathbf{z}^{(j)}\right)\mathbf{W}$ の重みとみなして、

  $\Phi\left(\mathbf{z}^{(j)}\right)\mathbf{W}$ を重み付き平均した座標が、対象としたサンプルを

  逆写像した点である