Universität Potsdam
Institut für Informatik
Lehrstuhl Maschinelles Lernen

Clusteranalyse: Gauß'sche Mischmodelle

Niels Landwehr

Überblick

- Problemstellung/Motivation
- Deterministischer Ansatz: K-Means
- Probabilistischer Ansatz: Gaußsche Mischmodelle
- Bayesscher Ansatz: Gaußsche Mischmodelle + Priors

Überblick

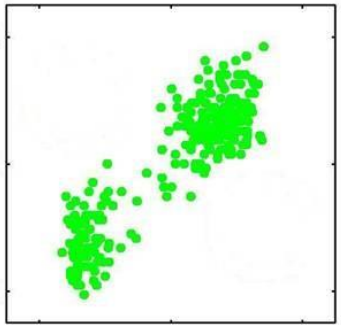
- Problemstellung/Motivation
- Deterministischer Ansatz: K-Means
- Probabilistischer Ansatz: Gaußsche Mischmodelle
- Bayesscher Ansatz: Gaußsche Mischmodelle + Priors

Clusteranalyse: Was ist Clustern?

- Wir haben Datenpunkte $\mathbf{x}_1, ..., \mathbf{x}_N$

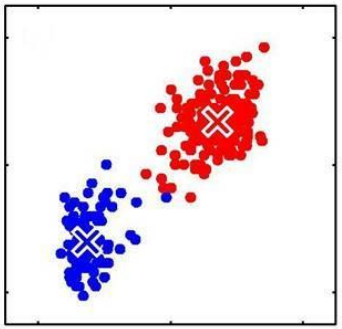
  Merkmalsvektoren $\mathbf{x}_n \in \mathbb{R}^D$

  Beispiel $\mathbb{R}^2$, 272 Datenpunkte



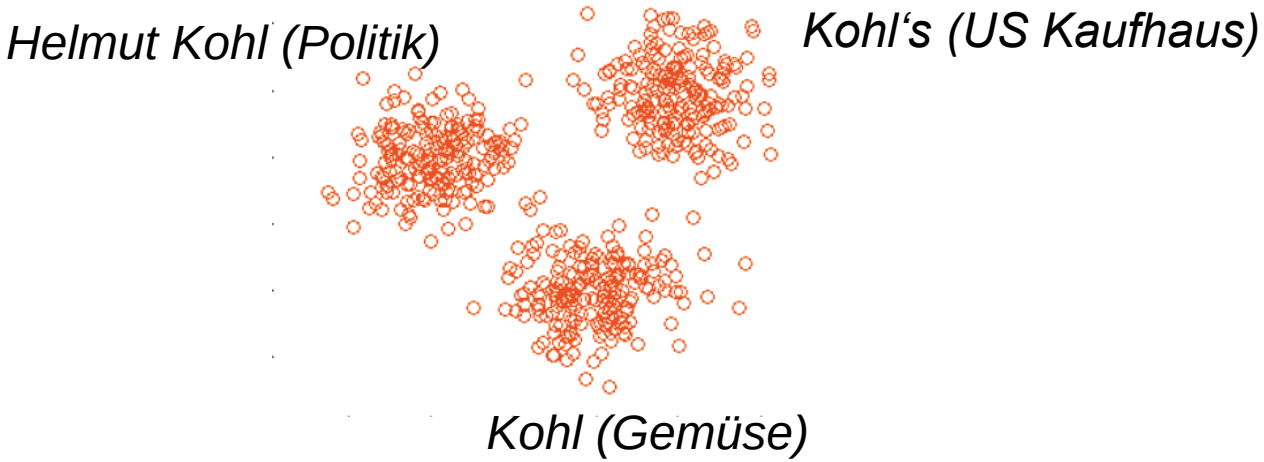
- Wir wollen Einteilung der Datenpunkte in „Cluster“

Jeder Punkt wird entweder Cluster 1
oder Cluster 2 zugewiesen
(im Allgemeinen: $K \geq 2$ Cluster)



Clusteranalyse: Anwendungen

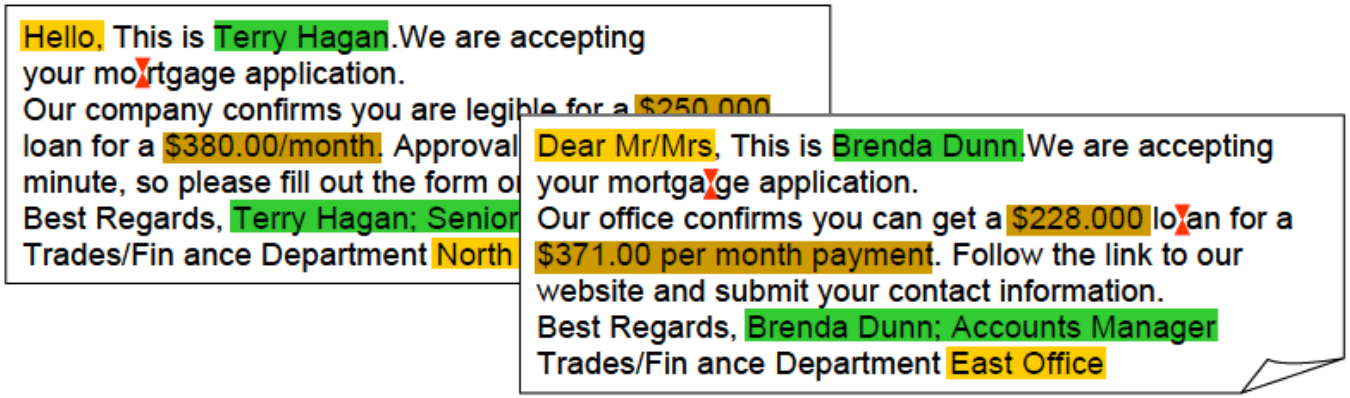
- Überblick über eine Dokumentenkollektion
  - Z.B. Suchmaschine: Suchwort „Kohl“
  - Liefert grosse Menge von Dokumenten



  - Idee: zeige dem Nutzer die Cluster, um genauere Auswahl des Themas zu ermöglichen

Clusteranalyse: Anwendungen

- Spam Kampagnen identifizieren
  - Spam-Kampagne: große Menge ähnlicher (aber nicht gleicher) e-mails



  - Eine Kampagne ist ein deutlicher Cluster ähnlicher e-mails

Überblick

- Problemstellung/Motivation
- Deterministischer Ansatz: K-Means
- Probabilistischer Ansatz: Gaußsches Mischmodell
- Bayesscher Ansatz: Gaußsches Mischmodell + Priors

Problemstellung Clustering (Deterministisch)

- Gegeben
  - Daten $\mathbf{x}_1, ..., \mathbf{x}_N$ mit $\mathbf{x}_n \in \mathbb{R}^D$
  - Anzahl K vermuteter Cluster

Oft problematisch (woher wissen wir K?)

- Gesucht
  - Zuweisung der Daten zu Clustern *1,...,K*

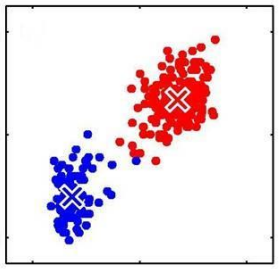
$$\mathbf{r}_n \in \{0,1\}^K \qquad r_{nk} = \begin{cases} 1: & \mathbf{x}_n \text{ in Cluster } k \\ 0: & sonst \end{cases}$$

z.B. $\mathbf{r}_1 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$

x_1 *liegt im 3. Cluster*

  - Clusterzentren

$$\mu_1, ..., \mu_K \in \mathbb{R}^D$$

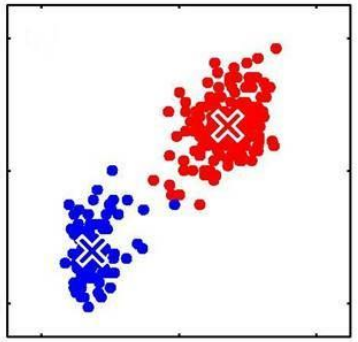


Problemstellung Clustering (Deterministisch)

- Ziel/Optimierungskriterium
  - „Punkte in einem Cluster sollen alle ähnlich sein, d.h. geringen Abstand im Merkmalsraum haben“
  - Minimiere quadratische Abstand zum Clusterzentrum:

$$J = \sum_{n=1}^{N} \underbrace{\sum_{k=1}^{K} r_{nk} \| \mathbf{x}_n - \mu_k \|^2}_{\text{Abstand } \mathbf{x}_n \text{ zu Clusterzentrum}}$$

Minimieren in $\mathbf{r}_1, ..., \mathbf{r}_n$ und $\mu_1, ..., \mu_K$



K-Means Algorithmus

- Gleichzeitiges Min. über $\mu_1, ..., \mu_K$ und $\mathbf{r}_1, ..., \mathbf{r}_N$ schwierig

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \mu_k\|^2$$

- Iterativer Algorithmus: Abwechselnde Minimierung
  - Starte mit zufälligen $\mu_1, ..., \mu_K$
  - Update

$$\mathbf{r}_1^{neu}, ..., \mathbf{r}_N^{neu} = \arg\min_{\mathbf{r}_1,...,\mathbf{r}_N} \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \mu_k\|^2 \qquad \text{„Expectation“}$$

$$\mu_1^{neu}, ..., \mu_K^{neu} = \arg\min_{\mu_1,...,\mu_K} \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk}^{neu} \|\mathbf{x}_n - \mu_k\|^2 \qquad \text{„Maximization“}$$

  - Iteriere bis Konvergenz

- Konvergenz sicher, weil *J* immer sinkt – aber im Allgemeinen nur lokales Optimum

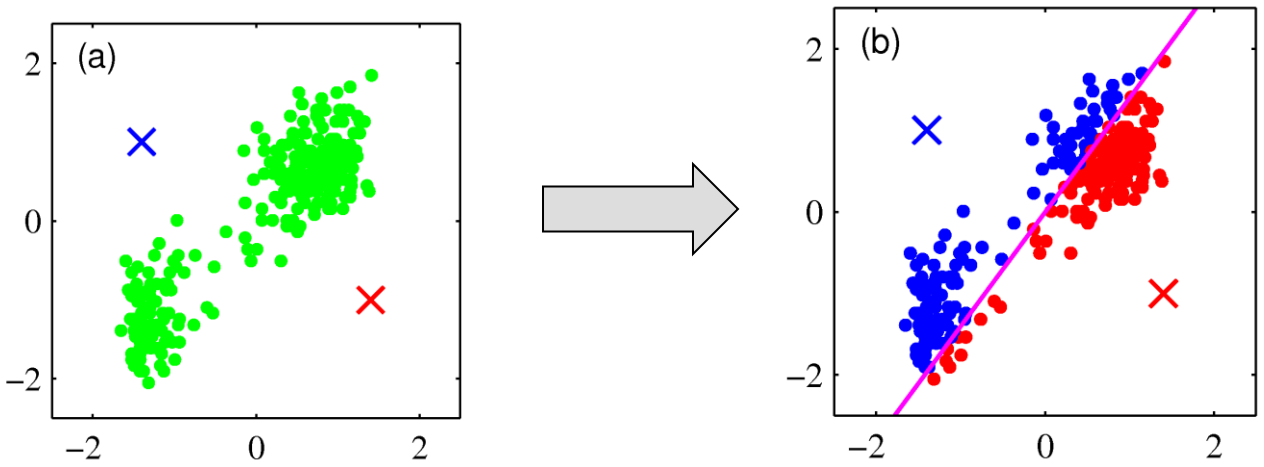
K-Means Algorithmus

- Expectation Schritt

$$\mathbf{r}_1^{neu}, ..., \mathbf{r}_N^{neu} = \underset{\mathbf{r}_1,...,\mathbf{r}_N}{\arg\min} \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \mu_k\|^2$$

  - Einfach: ordne jeden Punkt dem ihm nächsten Cluster(zentrum) zu

$$r_{nk}^{neu} = \begin{cases} 1: & \text{if } k = \arg\min_j \|\mathbf{x}_n - \mu_j\| \\ 0: & sonst \end{cases}$$



K-Means Algorithmus

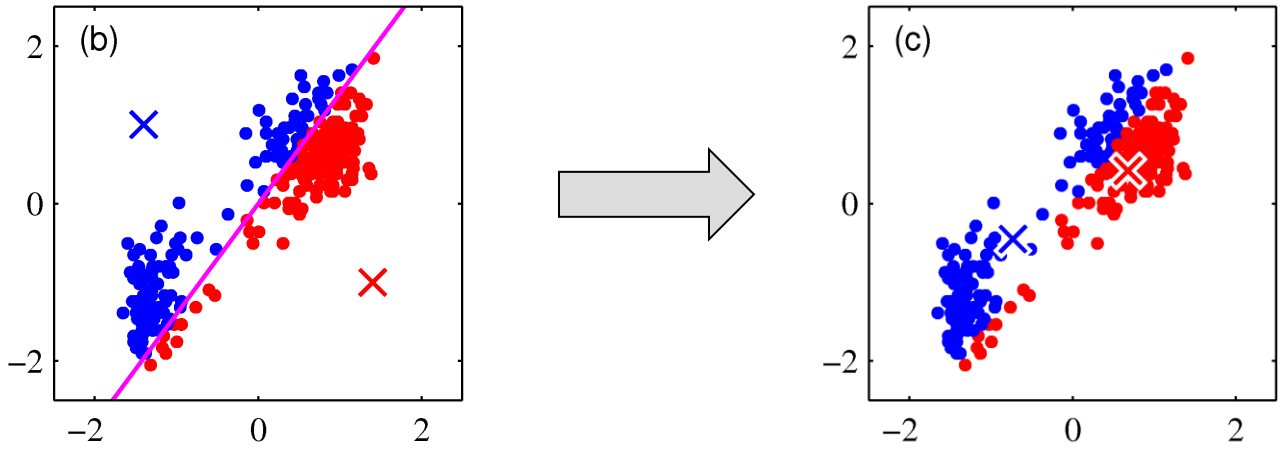
- Maximization Schritt:

$$\mu_1^{neu},...,\mu_K^{neu} = \arg\min_{\mu_1,...,\mu_k} \sum_n \sum_k r_{nk}^{neu} \left\| \mathbf{x}_n - \mu_k \right\|^2$$

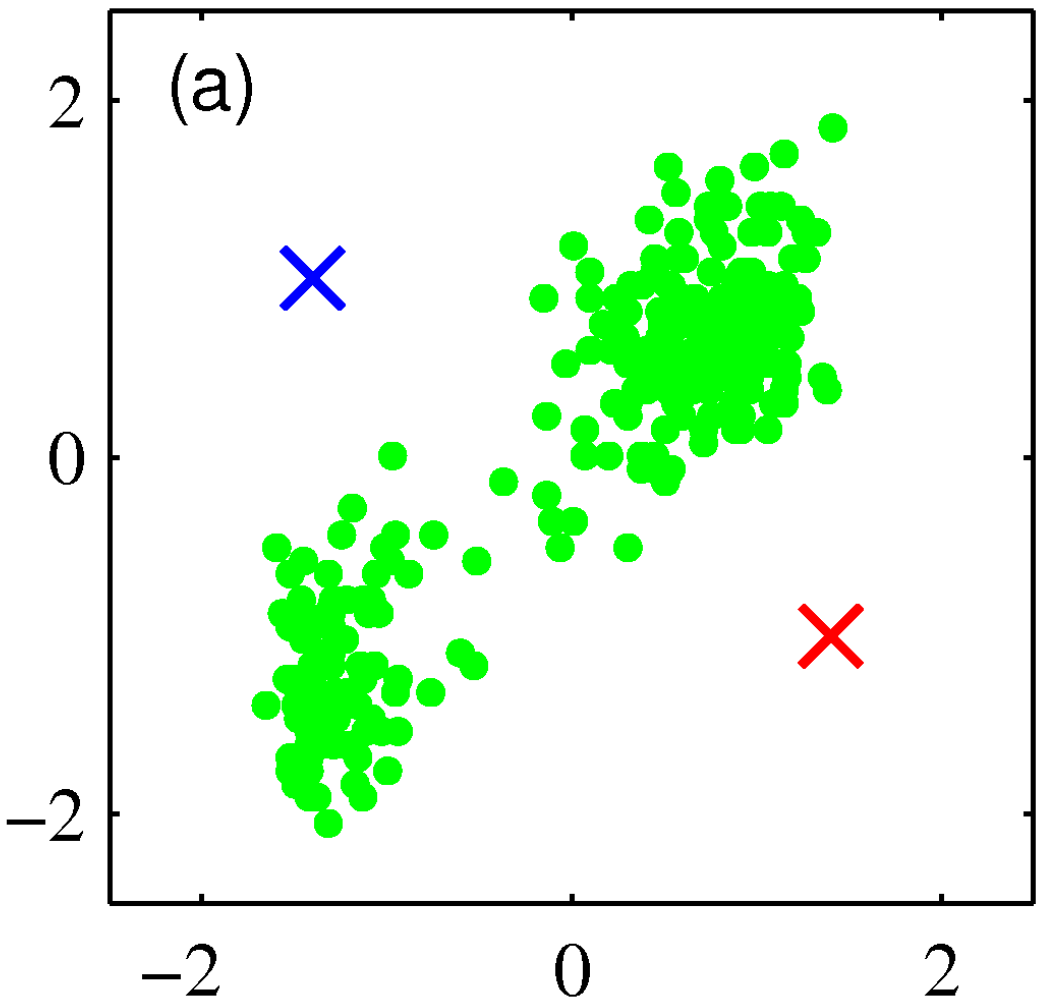
- Ableitung Null setzen:

$$\mu_k^{neu} = \frac{\sum_n r_{nk} \mathbf{x}_n}{\sum_n r_{nk}}$$

„Durchschnitt der Punkte, die in den Cluster fallen“

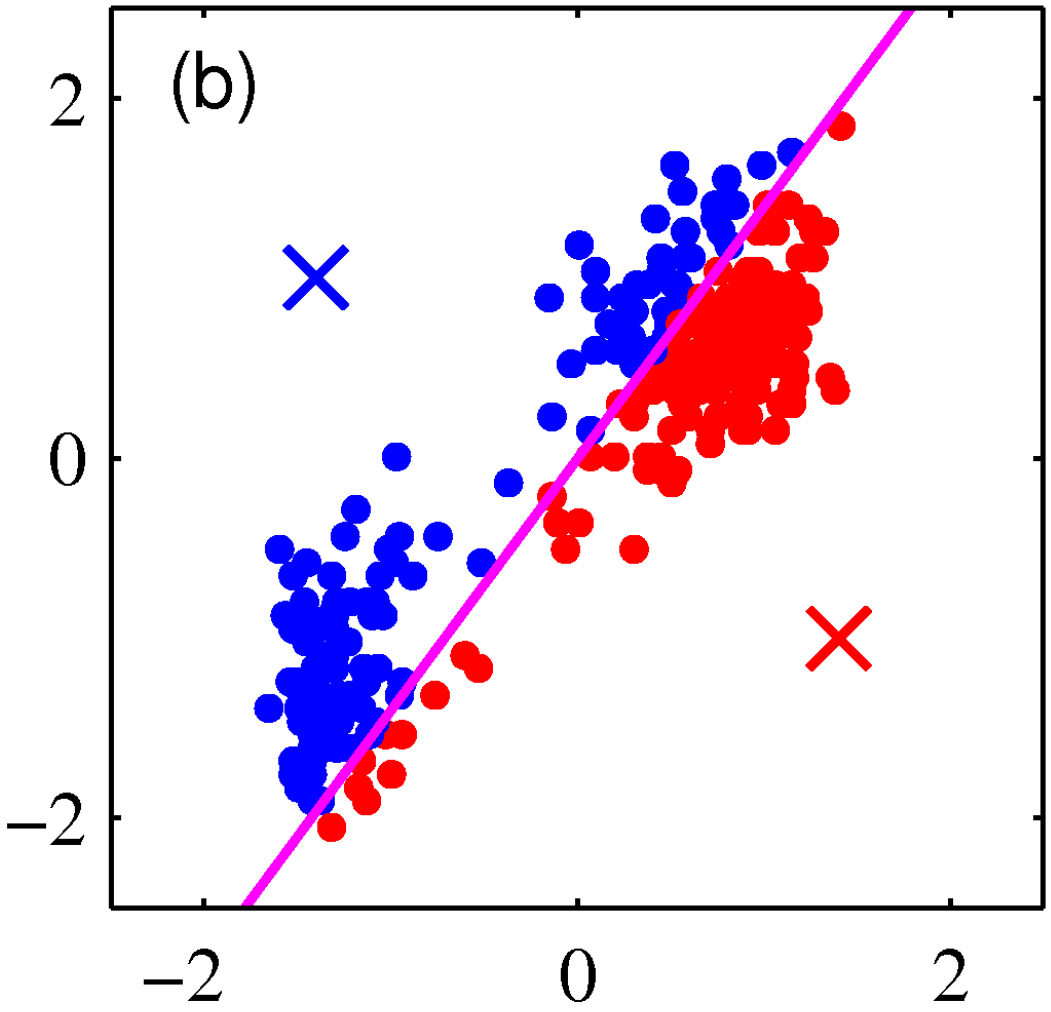


K-Means: Beispiel K = 2



Start:
Zufällige Initialisierung von μ_1, μ_2

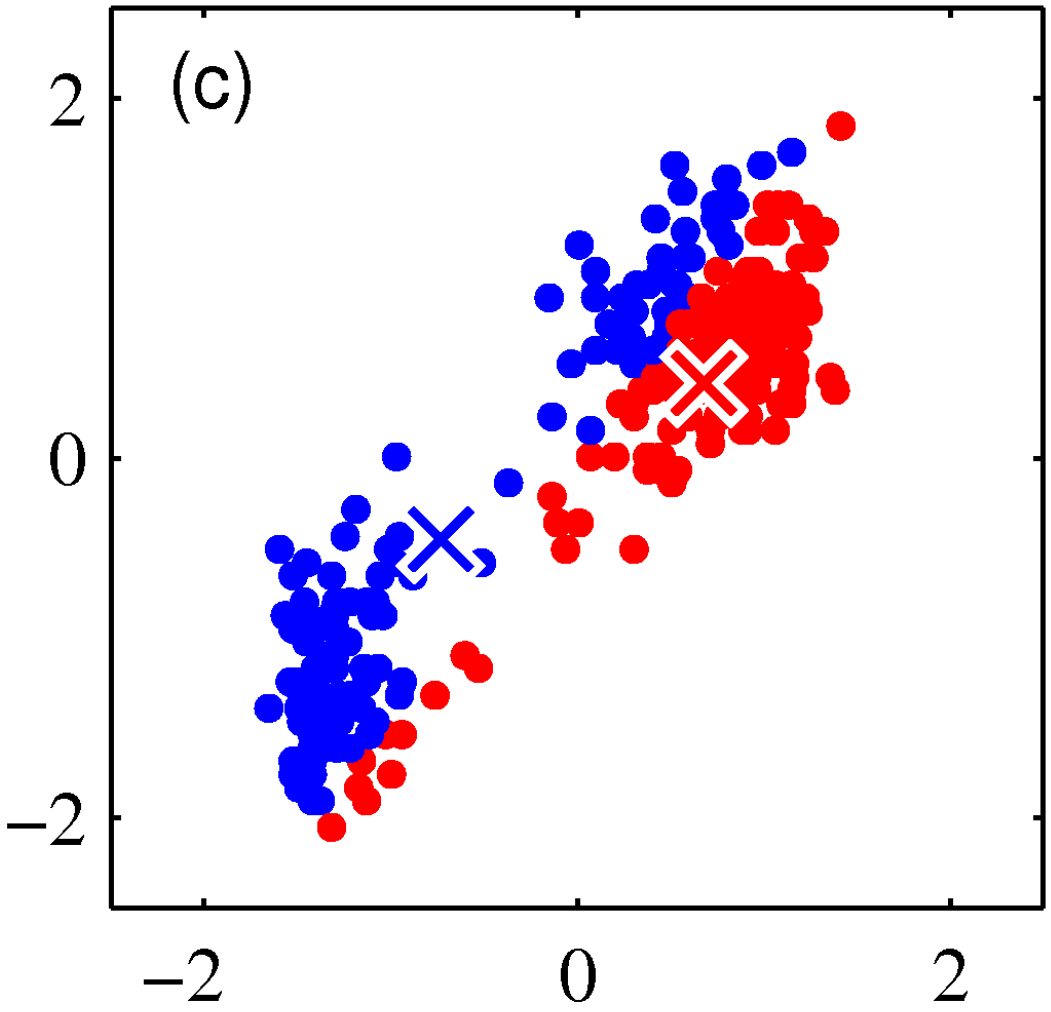
K-Means: Beispiel K = 2



Expectation:

$$r_{nk}^{neu} = \begin{cases} 1: & \text{if } k = \arg\min_j \|\mathbf{x}_n - \mu_j\| \\ 0: & sonst \end{cases}$$

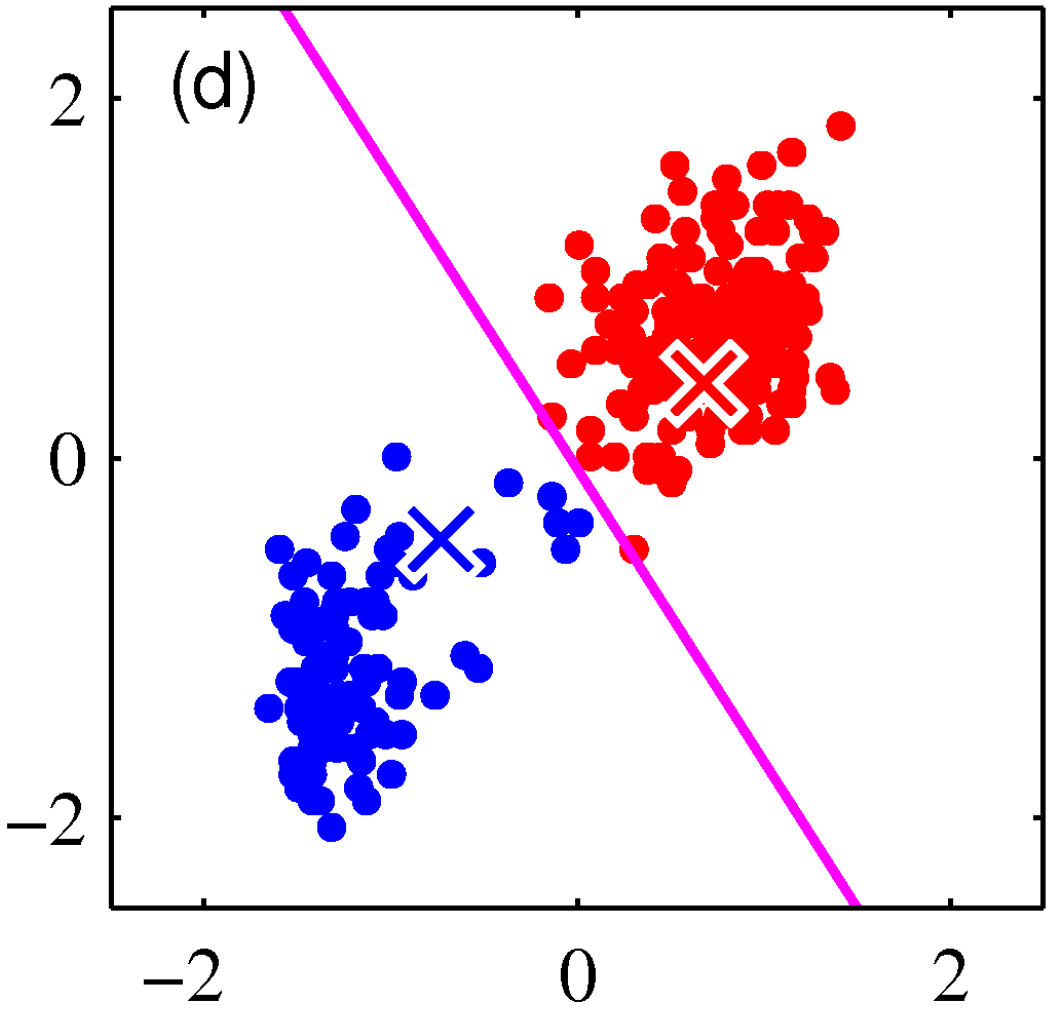
K-Means: Beispiel K = 2



Maximization:

$$\mu_k^{neu} = \frac{\sum_n r_{nk}\mathbf{x}_n}{\sum_n r_{nk}}$$

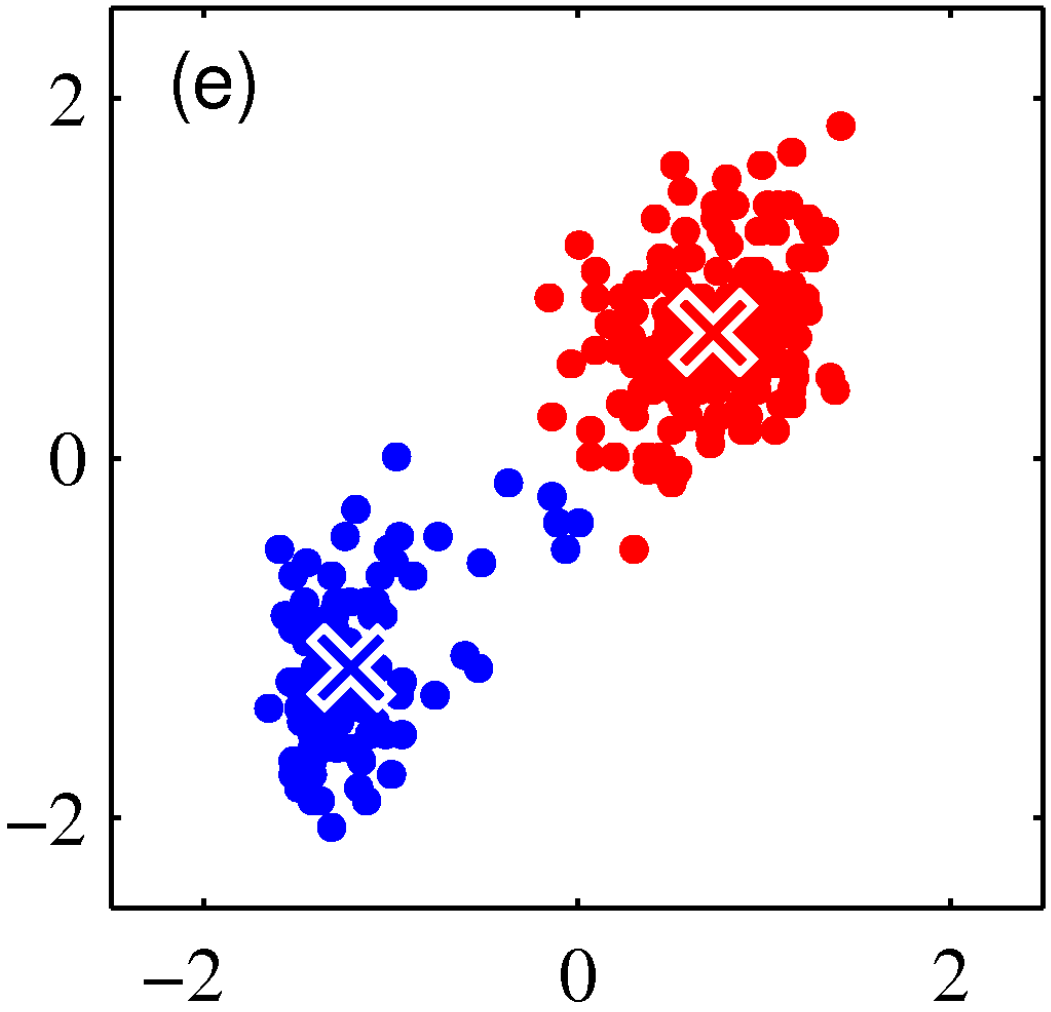
K-Means: Beispiel K = 2



Expectation:

$$r_{nk}^{neu} = \begin{cases} 1: & \text{if } k = \arg\min_j \|\mathbf{x}_n - \mu_j\| \\ 0: & sonst \end{cases}$$

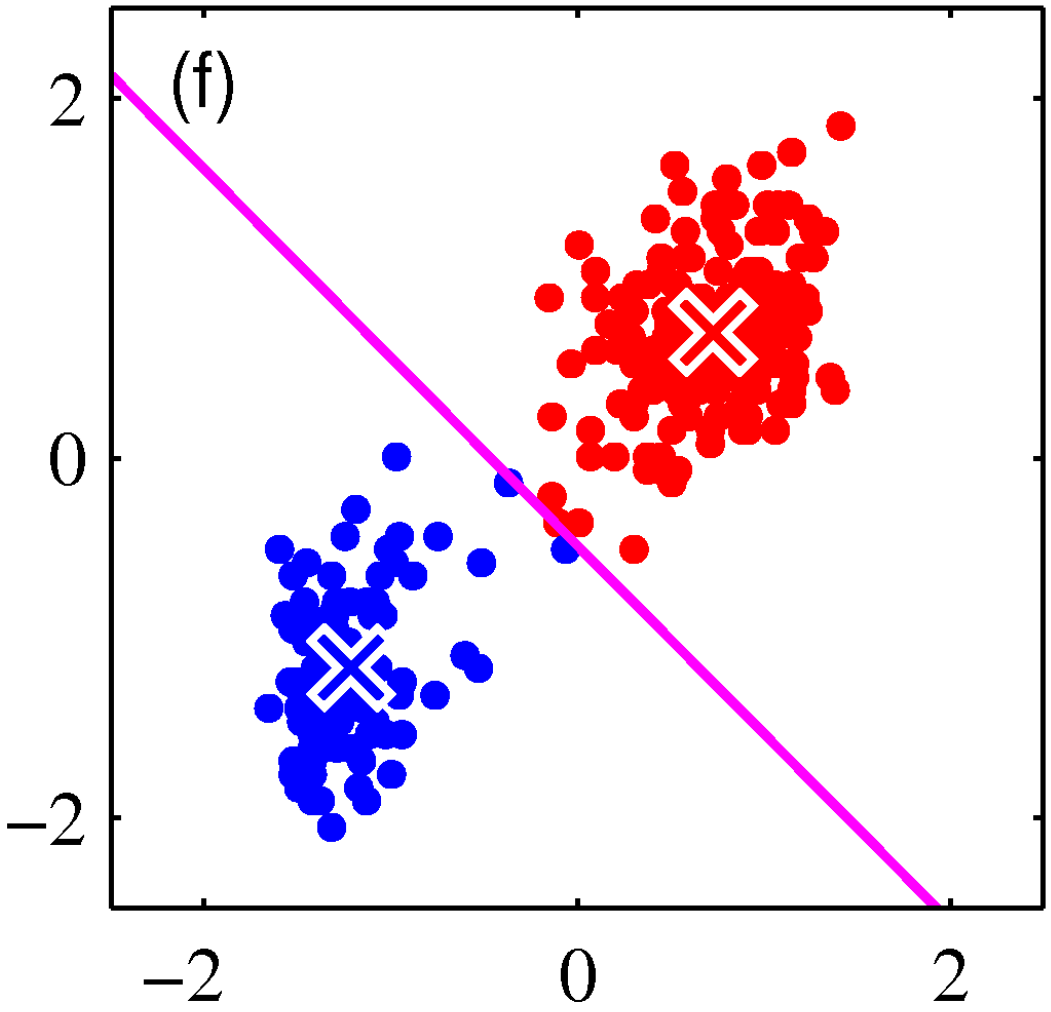
K-Means: Beispiel K = 2



Maximization:

$$\mu_k^{neu} = \frac{\sum_n r_{nk}\mathbf{x}_n}{\sum_n r_{nk}}$$

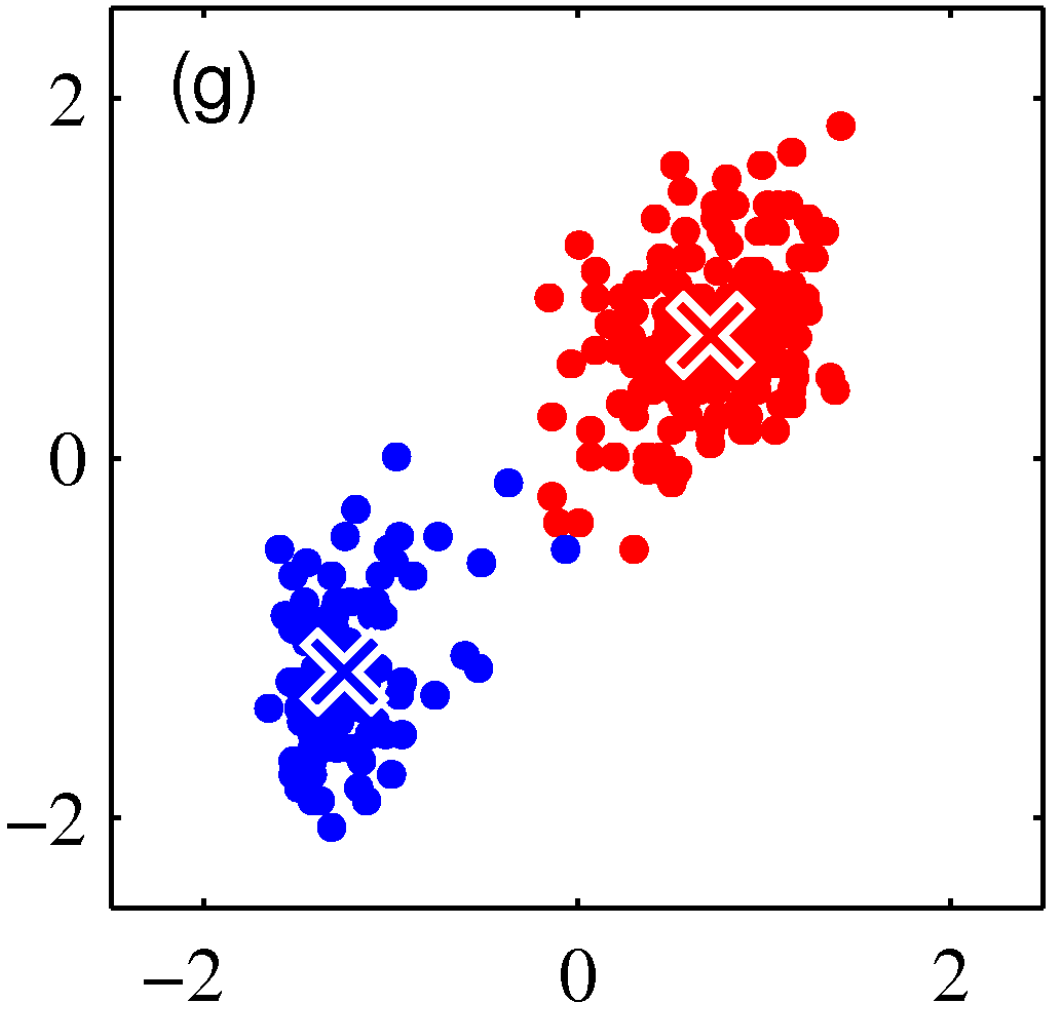
K-Means: Beispiel K = 2



Expectation:

$$r_{nk}^{neu} = \begin{cases} 1: & \text{if } k = \arg\min_j \|\mathbf{x}_n - \mu_j\| \\ 0: & sonst \end{cases}$$

K-Means: Beispiel K = 2



Maximization:

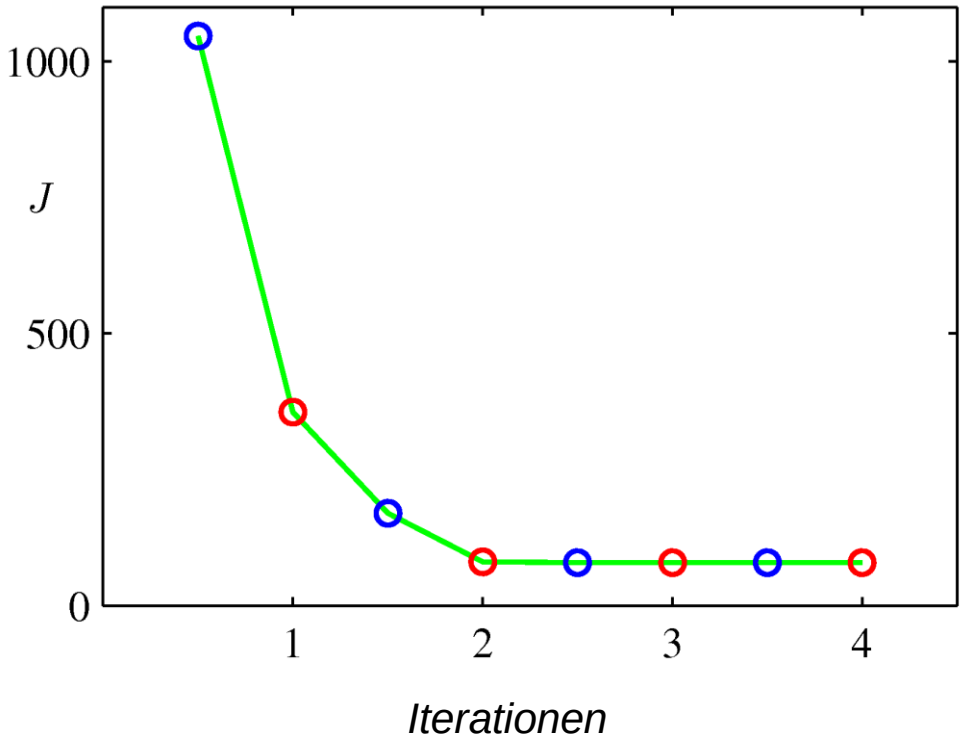
$$\mu_k^{neu} = \frac{\sum_n r_{nk}\mathbf{x}_n}{\sum_n r_{nk}}$$

usw. (Konvergenz in nächster Iteration)

K-Means: Beispiel K = 2

- Kostenfunktion *J* fällt kontinuierlich

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \mu_k\|^2$$



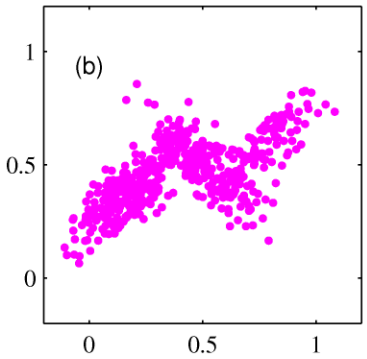
Kommentare K-Means

- ☺ Einfach zu implementieren
- ☺ Relativ schnell: O(NK) per Iteration
- ☹ Nur lokales Optimum garantiert: unterschiedliche Startwerte = unterschiedliche Lösungen
- ☹ Keine Konfidenz für Clusterzugehörigkeit
- ☹ Muss Anzahl Cluster vorgeben

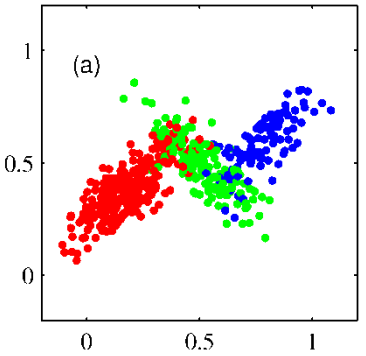
Probabilistisches Clustern besser

- Clustern sollte Konfidenz liefern: für einige Datenpunkte können wir keine sichere Entscheidung treffen!
- Probabilistisches Clustern

Beobachtete Daten

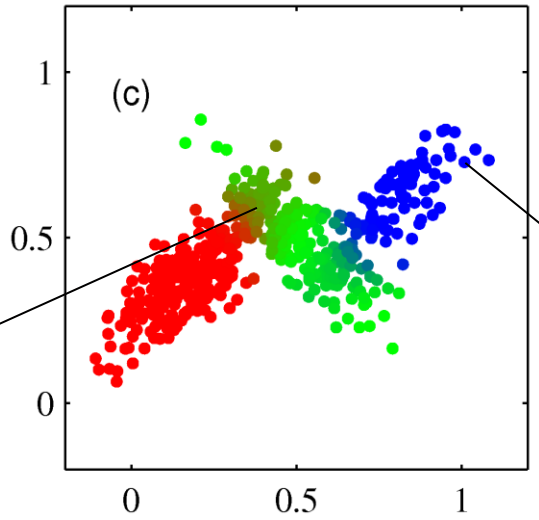


Ursprüngliche Klassen (nicht beobachtet)



Clustern

Probabilistische Cluster



Cluster 1 oder 2?

Sicher Cluster 3!

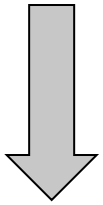
Überblick

- Problemstellung/Motivation
- Deterministischer Ansatz: k-Means
- Probabilistischer Ansatz: Gaußsches Mischmodell
- Bayesscher Ansatz: Gaußsches Mischmodell + Priors

Probabilistisches Clustern mit Generativem Modell

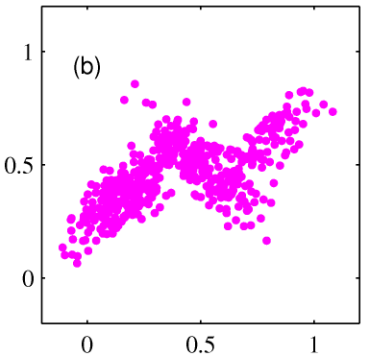
- Idee: Generatives Modell, das die Daten erzeugt haben könnte
- Modell hat Parametervektor $\Theta = (\pi, \mu, \Sigma)$

Modell $\Theta = (\pi, \mu, \Sigma)$



Generativer Prozess

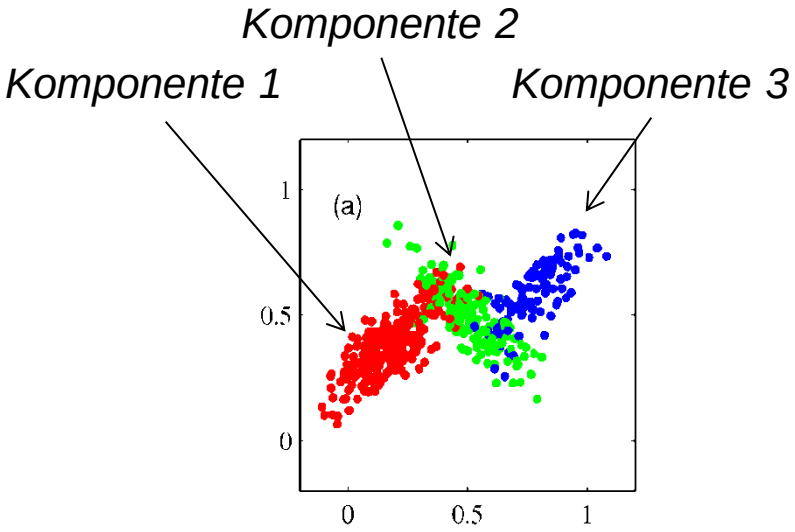
Daten



Form der Daten hängt ab von Parametern $\Theta = (\pi, \mu, \Sigma)$

Probabilistisches Clustern: Gaußsches Mischmodell

- Generativer Prozess:
  - Wähle Clusterkomponente k
  - Generiere einen Datenpunkt zu diesem Cluster



- Zufallsvariablen:
  - Clusterzugehörigkeit **z:** Kodierung wie bei k-Means

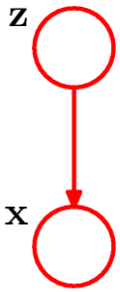
$$\mathbf{z} = \begin{pmatrix} z_1 \\ z_2 \\ \cdots \\ z_K \end{pmatrix} \qquad z_k = \begin{cases} 1 : x \text{ in Cluster } k \\ 0 : \text{sonst} \end{cases} \qquad \text{z.B. } \mathbf{z} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

Datenpunkt im 3. Cluster

  - Datenpunkt **x**

Probabilistisches Clustern: Gaußsches Mischmodell

- Clusterkomponente wählen, anschließend Datenpunkt generieren



Verteilung über Clusterzugehörigkeit $\mathbf{z}$*: multinomial*

Parameter $\pi = (\pi_1, ..., \pi_K), \quad \sum_{i=1}^{K} \pi_i = 1$

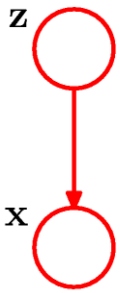
$$p(z_k = 1) = \pi_k$$

$$p(\mathbf{z}) = \prod_{k=1}^{K} \pi_k^{z_k}$$

Nur einer der Faktoren ungleich Eins

Probabilistisches Clustern: Gaußsches Mischmodell

- Clusterkomponente wählen, anschließend Datenpunkt generieren



Verteilung über Datenpunkte gegeben Cluster: Multivariate Normalverteilungen

$$p(\mathbf{x} \mid z_k = 1) = \mathcal{N}(\mathbf{x} \mid \mu_k, \Sigma_k)$$

Cluster-spezifische Parameter: Clusterzentrum, Kovarianzmatrix

$$p(\mathbf{x} \mid \mathbf{z}) = \prod_{k=1}^{K} \mathcal{N}(\mathbf{x} \mid \mu_k, \Sigma_k)^{z_k}$$

Nur einer der Faktoren ungleich Eins

Parameter: $\mu=(\mu_1,...,\mu_K)$ (Clusterzentren); $\Sigma = (\Sigma_1,...,\Sigma_K)$ (Kovarianzmatrizen)

Probabilistisches Clustern: Gaußsches Mischmodell

- Verteilung der Daten in einem Cluster k

Normalverteilung

Clusterzentrum

Clusterkovarianz

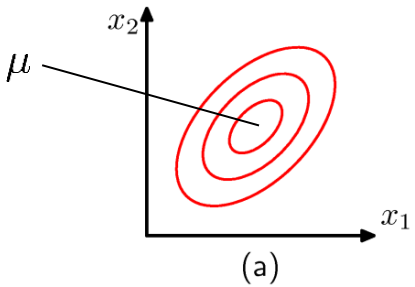
$$p(\mathbf{x} \mid z_k = 1) = \mathcal{N}(\mathbf{x} \mid \mu_k, \Sigma_k)$$

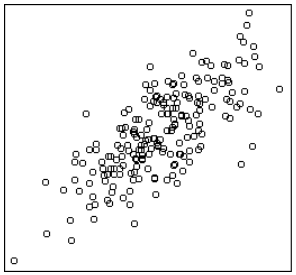
$$= \frac{1}{Z} \exp\left(-\frac{1}{2} (\mathbf{x} - \mu_k)^T \Sigma_k^{-1} (\mathbf{x} - \mu_k) \right)$$

Normalisierer

$$Z = 2\pi^{D/2} \mid \Sigma \mid^{1/2}$$

- Beispiel $D=2$: Dichte, Samples aus Verteilung

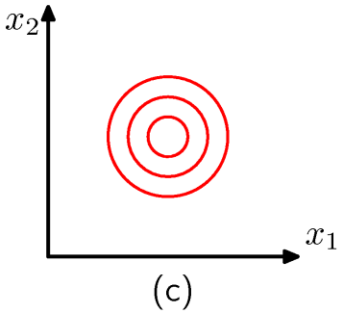




Probabilistisches Clustern: Gaußsches Mischmodell

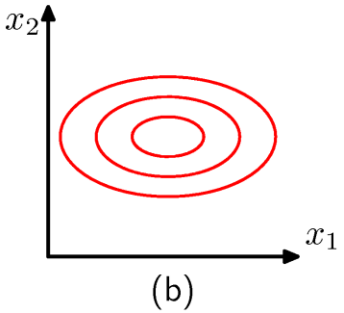
- Interpretation der Parameter μ_k, Σ_k
  - Parameter $\mu_k \in \mathbb{R}^D$ ist der Mittelpunkt des Clusters
  - Kovarianzmatrix $\Sigma_k \in M_{DxD}(\mathbb{R})$ beschreibt die Form des Clusters, d.h. wie Dichte um den Mittelwert streut

$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$



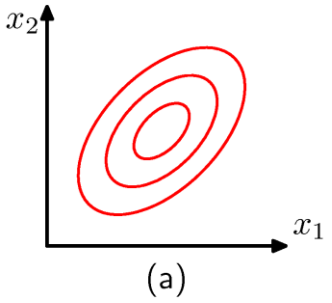
(c)

$$\boldsymbol{\Sigma} = \begin{pmatrix} 4 & 0 \\ 0 & 1 \end{pmatrix}$$



(b)

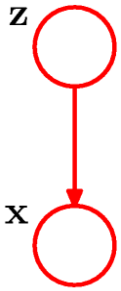
$$\boldsymbol{\Sigma} = \begin{pmatrix} 2.5 & 1.5 \\ 1.5 & 2.5 \end{pmatrix}$$

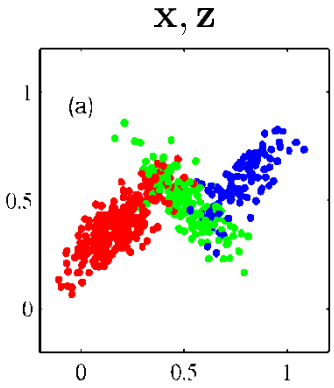


(a)

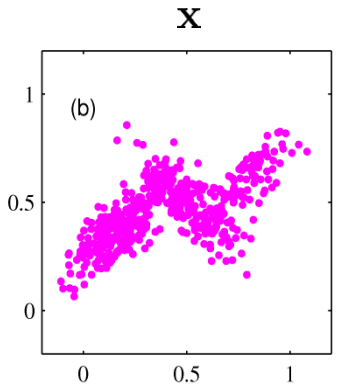
Beispiel Gaußsches Mischmodell

- Gesamtmodell: „Gaußsches Mischmodell“
  - Erzeugt Daten bestehend aus mehreren Clustern
  - Beispiel K = 3, 500 Datenpunkte gezogen





ZV $\mathbf{z}$ nicht beobachtet



Clusterzentren

$$\mu_1 \approx \begin{pmatrix} 0.2 \\ 0.4 \end{pmatrix}$$

$$\mu_2 \approx \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix}$$

$$\mu_3 \approx \begin{pmatrix} 0.8 \\ 0.6 \end{pmatrix}$$

Clusterkovarianzen

Σ_1 Σ_2 Σ_3 *Geben an, wie die Punkte um das Clusterzentrum streuen*

Probabilistisches Clustern: Gaußsches Mischmodell

- Wir ziehen N Datenpunkte aus dem Gaußschen Mischmodell
- Graphisches Modell, Parameter explizit (Parameter keine ZV)

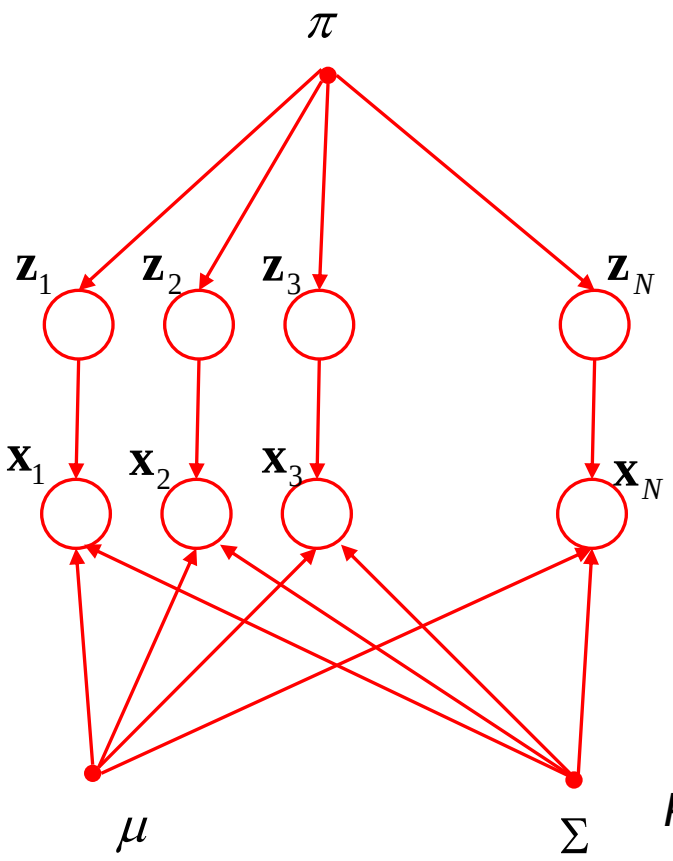
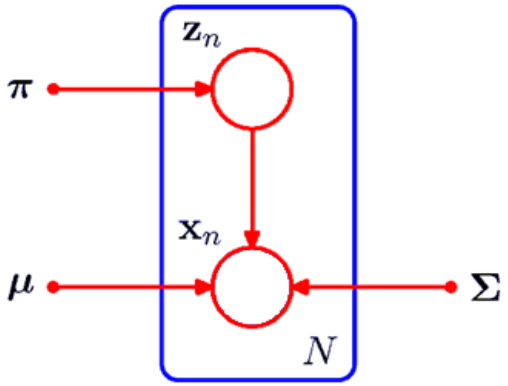


Plate-Notation

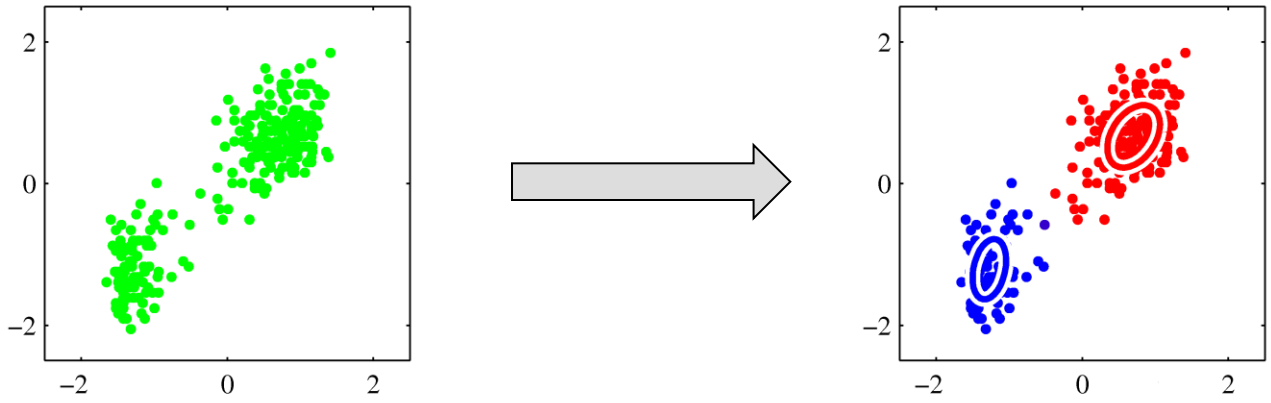


Parameter koppeln Beobachtungen

Clustern mit Gaußschem Mischmodell

- Gauß‘sches Mischmodell definiert Verteilungen über Datenpunkte (als Überlagerung einzelner Cluster)
- Form/Lage der Cluster abhängig von Modellparametern

- Problemstellung in der Praxis: Daten ➔ Cluster
  - Anpassen des Modells an Daten = Parameterlernen
  - Inferieren der Clusterzugehörigkeiten gegeben Modell



Clustern mit Gaußschem Mischmodell (Maximum Likelihood)

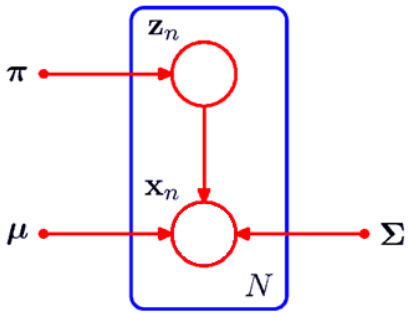
- Parameterlernproblem
  - Gegeben: Daten $X = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$
  - Gesucht: Parameter $\Theta = (\pi, \mu, \Sigma)$

- Optimierungskriterium Likelihood:

$$\arg\max_\Theta \; p(X \mid \Theta) = \arg\max_\Theta \prod_{n=1}^{N} p(\mathbf{x}_n \mid \Theta) \qquad \text{(i.i.d)}$$

$$= \arg\max_\Theta \prod_{n=1}^{N} \sum_{\mathbf{z}_n} p(\mathbf{x}_n, \mathbf{z}_n \mid \Theta)$$

$$= \arg\max_\Theta \prod_{n=1}^{N} \sum_{\mathbf{z}_n} p(\mathbf{z}_n \mid \pi)\, p(\mathbf{x}_n \mid \mathbf{z}_n, \mu, \Sigma)$$



Produkt von Summen: schwierig zu optimieren

Maximum Likelihood: Vollständige Daten

- Zunächst Vereinfachung: vollständig beobachtete Daten

Definiere $Z = \{\mathbf{z}_1, ..., \mathbf{z}_N\}$ (Clusterzugehörigkeiten)

$$\Theta^* = \arg\max_\Theta p(X, Z \mid \Theta)$$

$$= \arg\max_\Theta \prod_{n=1}^{N} p(\mathbf{z}_n \mid \pi)\, p(\mathbf{x}_n \mid \mathbf{z}_n, \mu, \Sigma)$$

$$= \arg\max_\Theta \prod_{n=1}^{N} \prod_{k=1}^{K} \pi_k^{z_{nk}} \mathcal{N}(\mathbf{x}_n \mid \mu_n, \Sigma_n)^{z_{nk}}$$

Produkt von Produkten: leichter zu optimieren (Log!)

$$= \arg\max_\Theta \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} (\log(\pi_k) + \log(\mathcal{N}(\mathbf{x}_n \mid \mu_k, \Sigma_k))$$

Maximum Likelihood: Vollständige Daten

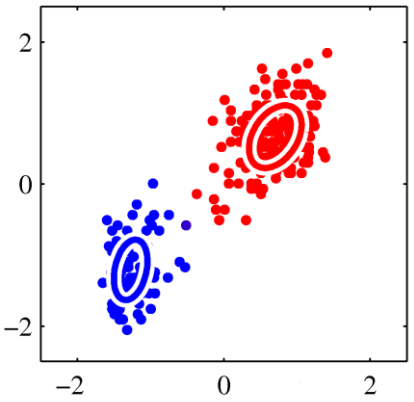
- Likelihood Maximierung ist relativ einfach, wenn wir X und Z kennen (geschlossene Lösung)

$$\pi_k^* = \frac{N_k}{N}$$

Anzahl Punkte in Clusterkomponente k

$$\mu_k^* = \frac{1}{N_k}\sum_{n=1}^{N} z_{nk}\mathbf{x}_n$$

$$\Sigma_k^* = \frac{1}{N_k}\sum_{n=1}^{N} z_{nk}(\mathbf{x}_n - \mu_k^*)(\mathbf{x}_n - \mu_k^*)^T$$



$$N_k = \sum_{n=1}^{N} z_{nk}, \quad z_{nk} \in \{0,1\}$$ Indikator: $\mathbf{x}_n$ in Cluster k?

EM Algorithmus

- Problem: Z nicht beobachtet!
- Wir müssen schwieriges Problem lösen:

$$\Theta^* = \arg\max_\Theta \, p(X \mid \Theta)$$

- Lösung mit dem EM-Algorithmus („Expectation-Maximization“)

EM Algorithmus

- Iteratives Verfahren: bestimme $\Theta_1, \Theta_2, \Theta_3, \ldots$
- Berechnung von Θ_{t+1} als Argmax der Q-Funktion

$$\mathcal{Q}(\boldsymbol{\Theta}, \boldsymbol{\Theta}_t) = \mathbb{E}_{\mathbf{Z}}[\log\, p(\mathbf{X}, \mathbf{Z} \mid \boldsymbol{\Theta}) \mid \mathbf{X}, \boldsymbol{\Theta}_t]$$

Parameterwert im letzten Schritt

Berechnen als Funktion von $\boldsymbol{\Theta}$

- Beginne mit zufälligem Θ_1 . Iteriere:
  - Expectation: $\mathcal{Q}(\boldsymbol{\Theta}, \boldsymbol{\Theta}_t) = \mathbb{E}[\log\, p(\mathbf{X}, \mathbf{Z} \mid \boldsymbol{\Theta}) \mid \mathbf{X}, \boldsymbol{\Theta}_t]$
  - Maximization: $\boldsymbol{\Theta}_{t+1} = \arg\max_{\boldsymbol{\Theta}} \mathcal{Q}(\boldsymbol{\Theta}, \boldsymbol{\Theta}_t)$
- Theorem (Konvergenz): $p(\mathbf{X} \mid \boldsymbol{\Theta}_{t+1}) \geq p(\mathbf{X} \mid \boldsymbol{\Theta}_t)$
  - Allerdings nur lokales Maximum

EM für Gaußsches Mischmodell

- Q-Funktion für Gaußsches Mischmodell

$$
\begin{aligned}
Q(\Theta,\Theta_t) &= \mathbb{E}_Z\left[\log p(X,Z\mid\Theta)\mid X,\Theta_t\right] \\
&= \sum_Z p(Z\mid X,\Theta_t)\log p(X,Z\mid\Theta) \qquad \textit{(Def. Erwartungswert)} \\
&= \sum_Z p(Z\mid X,\Theta_t)\sum_{n=1}^{N}\sum_{k=1}^{K} z_{nk}(\log\pi_k+\log\mathcal{N}(\mathbf{x}_n\mid\mu_k,\Sigma_k)) \\
&= \sum_{n=1}^{N}\sum_{k=1}^{K}\underbrace{\sum_Z p(Z\mid X,\Theta_t)z_{nk}}_{\mathbb{E}[z_{nk}\mid X,\Theta_t]}(\log\pi_k+\log\mathcal{N}(\mathbf{x}_n\mid\mu_k,\Sigma_k)) \\
&= \sum_{n=1}^{N}\sum_{k=1}^{K}\mathbb{E}\left[z_{nk}\mid X,\Theta_t\right](\log\pi_k+\log\mathcal{N}(\mathbf{x}_n\mid\mu_k,\Sigma_k))
\end{aligned}
$$

EM für Gaußsches Mischmodell

- Q-Funktion = Likelihood der vollständigen Daten, wobei Indikatoren z_{nk} ersetzt sind durch ihre Erwartungswerte

$$\log p(X, Z \mid \Theta) = \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} (\log(\pi_k) + \log(\mathcal{N}(\mathbf{x}_n \mid \mu_k, \Sigma_k))$$

$$Q(\Theta, \Theta_t) = \sum_{n=1}^{N} \sum_{k=1}^{K} \underbrace{\mathbb{E}[z_{nk} \mid X, \Theta_t]}_{\text{"Responsibilities" } \gamma(z_{nk})} (\log(\pi_k) + \log(\mathcal{N}(\mathbf{x}_n \mid \mu_k, \Sigma_k))$$

EM für Gaußsches Mischmodell

- Expectation Schritt: Berechnung der „Responsibilities“
- Inferenz im aktuellen Modell, gegeben X

$$\gamma(z_{nk}) := \mathbb{E}[z_{nk} \mid X, \Theta_t] = p(z_{nk} = 1 \mid X, \Theta_t)$$
$$= \frac{\pi_k \mathcal{N}(\mathbf{x}_n \mid \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x}_n \mid \mu_j, \Sigma_j)}$$

$\gamma(z_{nk})$: Wahrscheinlichkeit, mit der Beispiel n in Cluster k fällt

"Weiche" Clusterzugehörigkeit

EM für Gaußsches Mischmodell

- Maximization Schritt: maximiere in $\Theta = (\pi, \mu, \Sigma)$

$$Q(\Theta, \Theta_t) = \mathbb{E}\left[\log p(X, Z \mid \Theta) \mid X, \Theta_t\right]$$

- Ergebnis:

$$\pi_k = \frac{N_k}{N}$$ *„Erwarteter Anteil von Punkten in Cluster k“*

$$\mu_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) \mathbf{x}_n$$ *„Gewichteter Mittelwert für Cluster k“*

$$\Sigma_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk})(x_n - \mu_k)(x_n - \mu_k)^T$$ *„Gewichtete Kovarianz für Cluster k“*

$$N_k = \sum_{n=1}^{N} \gamma(z_{nk}),$$ *„Erwartete Anzahl von Punkten in Cluster k“*

Zusammenfassung EM

- EM Zusammenfassung:
  - Starte mit zufälligen $\mu, \mathbf{\Sigma}, \pi$
  - Expectation: berechne „Responsibilities“

$$\gamma(z_{nk}) = p(z_{nk} = 1 | X, \Theta_t)$$

„weiche“ Clusterzugehörigkeiten

  - Maximization:

$$\pi_k = \frac{N_k}{N}$$

$$\mu_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) \mathbf{x}_n$$

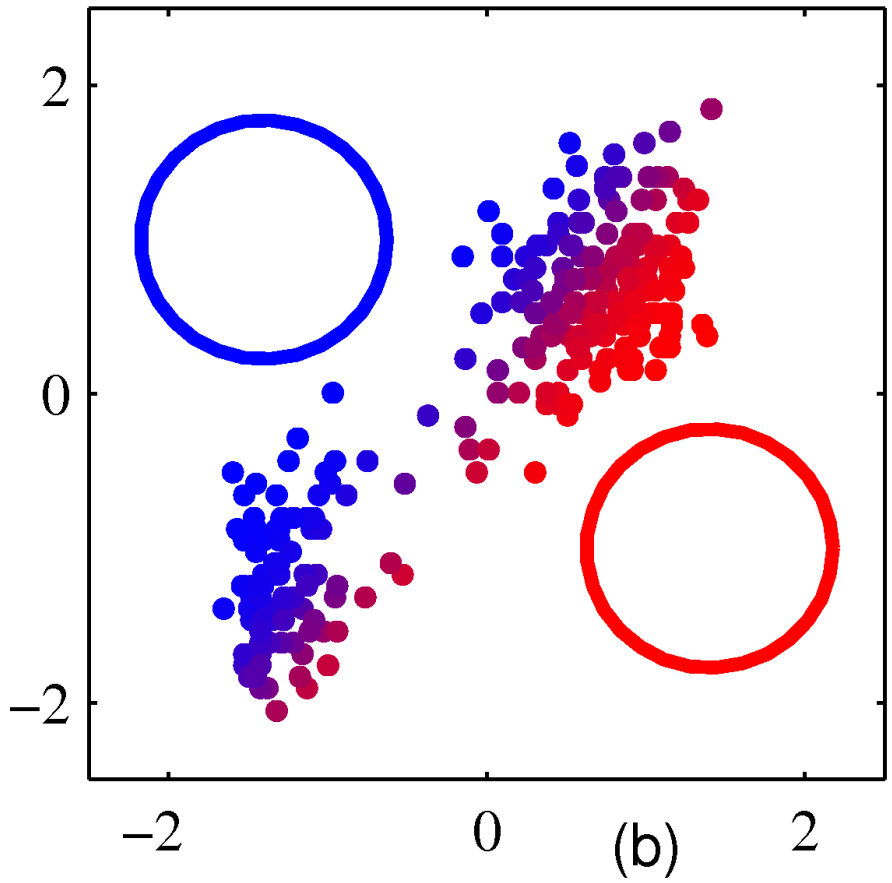
Berechnung der neuen Parameter gegeben weiche Clusterzugehörigkeiten

$$\mathbf{\Sigma}_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nl})(\mathbf{x}_n - \mu_k)(\mathbf{x}_n - \mu_k)^T$$

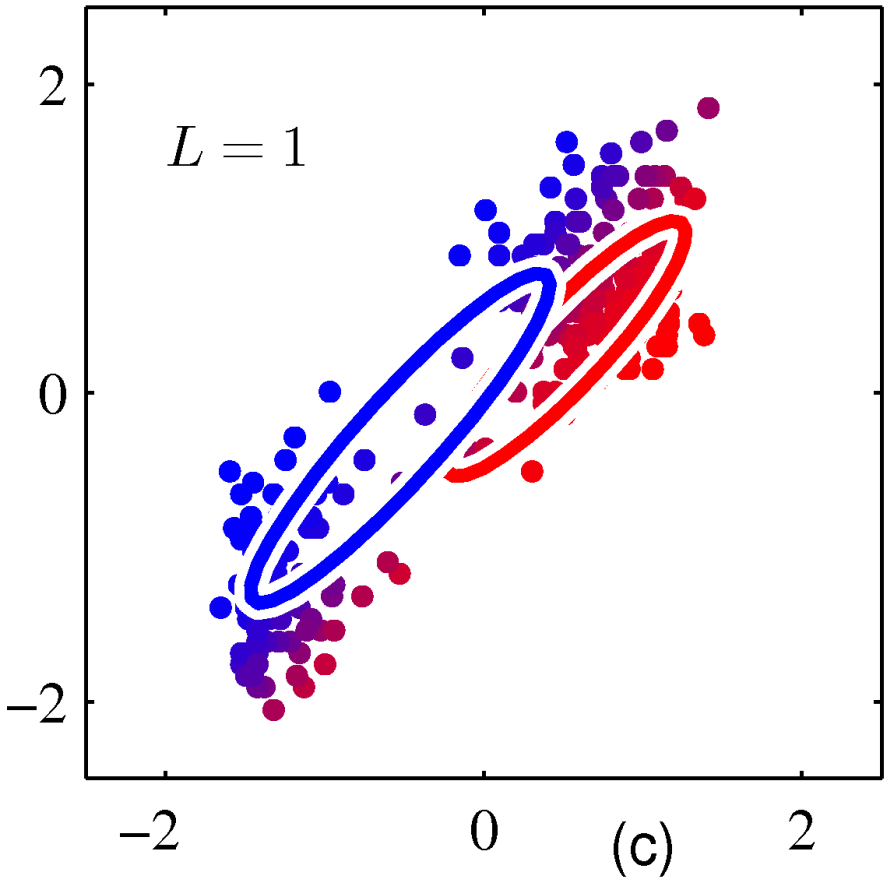
  - Wiederholen bis Konvergenz

- Gaußsches Mischmodell + EM ≈ „Weicher“ K-Means
  - Weiche Clusterzugehörigkeit, weiche Berechnung Clusterzentren

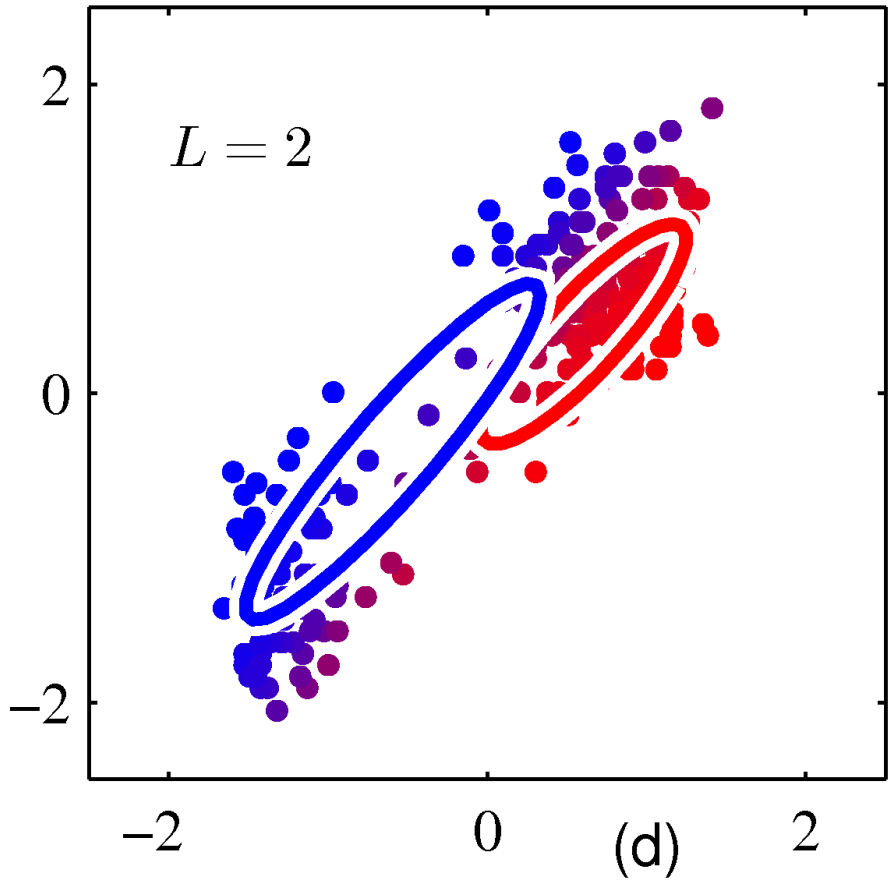
Beispiel Gaußsches Mischmodell Clustering



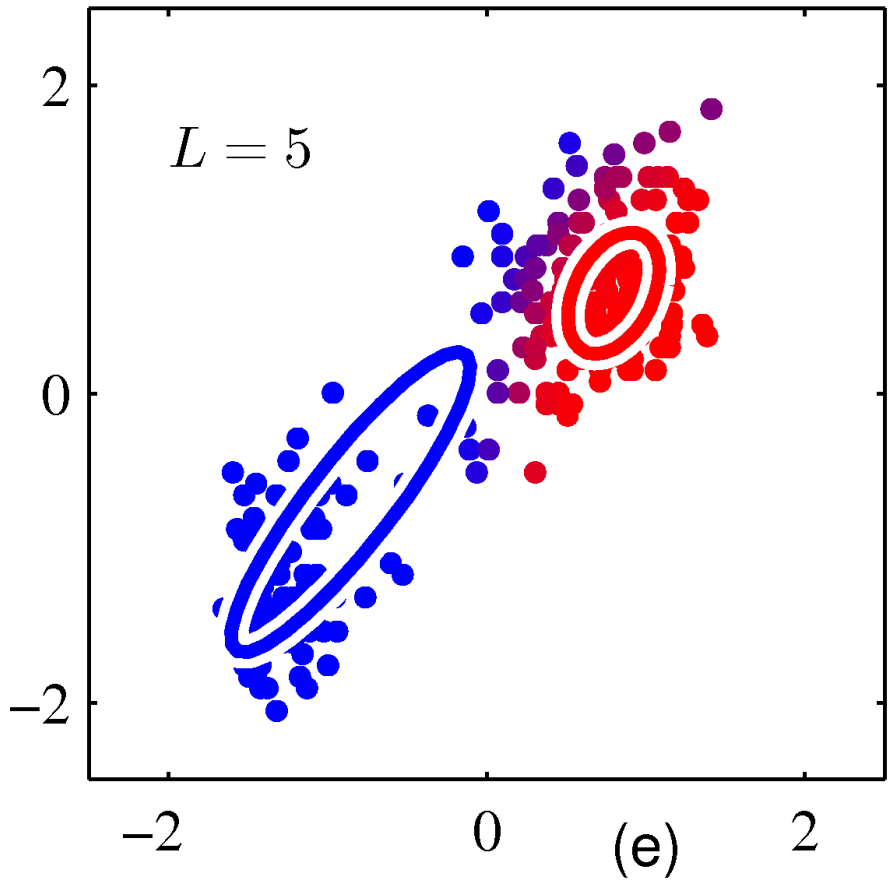
Beispiel Gaußsches Mischmodell Clustering



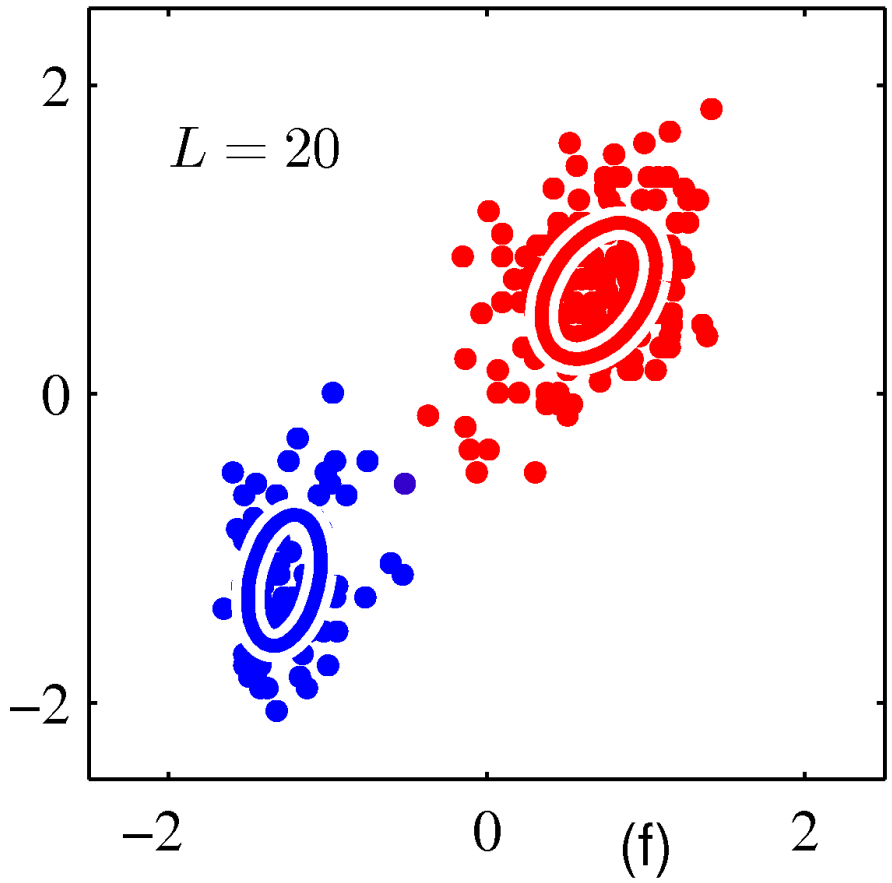
Beispiel Gaußsches Mischmodell Clustering



Beispiel Gaußsches Mischmodell Clustering



Beispiel Gaußsches Mischmodell Clustering



Überblick

- Problemstellung/Motivation
- Deterministischer Ansatz: k-Means
- Probabilistischer Ansatz: Gaußsches Mischmodell
- Bayesscher Ansatz: Gaußsches Mischmodell + Priors

Problem: Singularitäten

- EM maximiert Likelihood
- Problem des Overfittings

- Insbesondere: Singularität für

$$\mu_{\mathbf{k}} = \mathbf{x}_n, \quad \mathbf{x}_n \in \mathbf{X} \qquad \boldsymbol{\Sigma}_k \to \mathbf{0}$$

- Likelihood wird unendlich für $\boldsymbol{\Sigma}_k \to \mathbf{0}$

- Heuristik: Während EM diesen Fall detektieren und entsprechende Clusterkomponente neu initialisieren
- Bessere Lösung: Regularisierung durch Prior

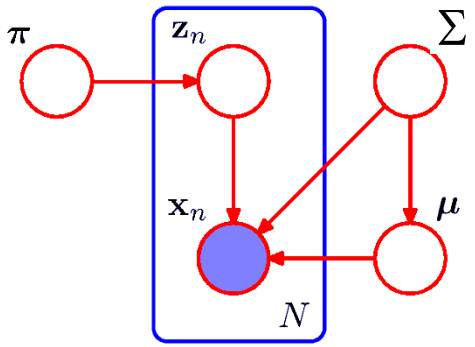
Prior Verteilungen für Gaußsches Mischmodell

- Gaußsches Mischmodell kann durch Prior Verteilungen erweitert werden
  - ZV $\pi, \mu, \mathbf{\Sigma}$
  - Prior-Verteilung $p(\pi, \mu, \Sigma) = p(\pi)\, p(\mu, \Sigma)$

$$= p(\pi)\, p(\mu \mid \Sigma)\, p(\Sigma)$$

„Erwartung für Parameterwerte“ (degenerative Fälle unwahrscheinlich)

  - Gesamtverteilung



MAP Lösung Gaußsches Mischmodell

- Maximum a posteriori Parameterschätzung:

$$\begin{aligned}\text{Suche } \boldsymbol{\Theta}^* &= \arg\max_{\boldsymbol{\Theta}} p(\boldsymbol{\Theta} \mid \mathbf{X}) \\ &= \arg\max_{\boldsymbol{\Theta}} p(\mathbf{X} \mid \boldsymbol{\Theta}) p(\boldsymbol{\Theta})\end{aligned}$$

- Anpassung des EM Algorithmus: maximiere

$$\begin{aligned}\mathcal{R}(\boldsymbol{\Theta}, \boldsymbol{\Theta}_t) &= \mathcal{Q}(\boldsymbol{\Theta}, \boldsymbol{\Theta}_t) + \log\, p(\boldsymbol{\Theta}) \\ &= \mathbb{E}[\log\, p(\mathbf{X}, \mathbf{Z} \mid \boldsymbol{\Theta}) \mid \mathbf{X}, \boldsymbol{\Theta}_t] + \log\, p(\boldsymbol{\Theta})\end{aligned}$$

- Entsprechende Änderung im M-Schritt notwendig (keine Details)

Vorteile von Prior Verteilung

- Löst das Problem der Singularitäten
  - Prior verhindert den Fall $\boldsymbol{\Sigma}_k \to \mathbf{0}$

$$\text{Suchen } \arg\max_{\boldsymbol{\Theta}} p(\mathbf{X} \mid \boldsymbol{\Theta}) p(\boldsymbol{\Theta})$$

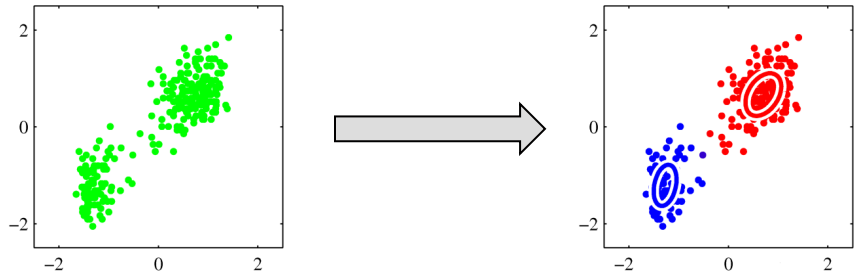
$$\text{Für } \boldsymbol{\Sigma}_k \to \mathbf{0} \text{ und } \mu_k = \mathbf{x}_n$$

$$p(\mathbf{X} \mid \boldsymbol{\Theta}) \to \infty \text{ aber } p(\boldsymbol{\Theta}) \to 0$$

- Für geeignete Wahl der Priorverteilung kann die Anzahl der Clusterkomponenten automatisch bestimmt werden: in der MAP Lösung sind einige π_k Null

Zusammenfassung

- Clusterproblem



- Deterministischer Ansatz: K-Means
  - Schnell, einfach, nicht probabilistisch

- Probabilistischer Ansatz mit Gaußschem Mischmodell
  - Allgemeiner + eleganter als K-Means
  - Training mit EM Algorithmus
  - Prior-Verteilungen auf Parametern um Overfitting zu vermeiden