Data Mining Tutorial

Clusteranalyse – Teil I

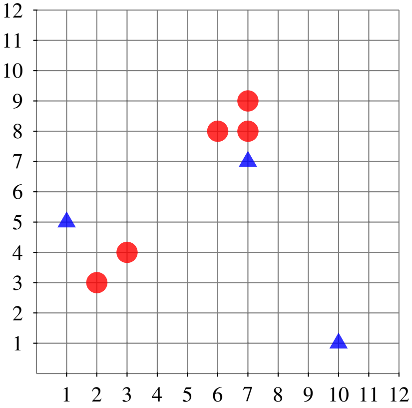
Erich Schubert, Arthur Zimek

Ludwig-Maximilians-Universität München

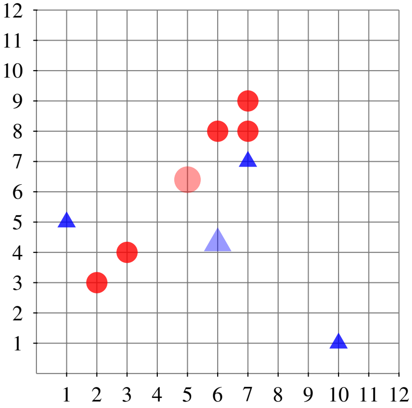
2015-05-13 — KDD Übung

k-Means Clustering – Lloyd/Forgy Algorithmus

E. Schubert, A. Zimek



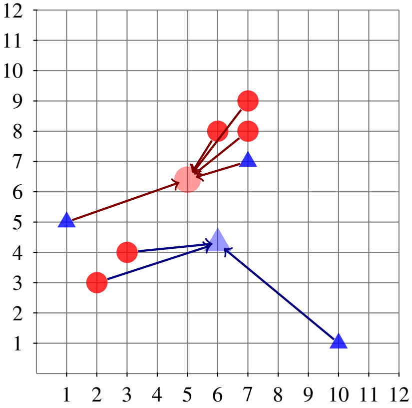
k-Means Clustering – Lloyd/Forgy Algorithmus



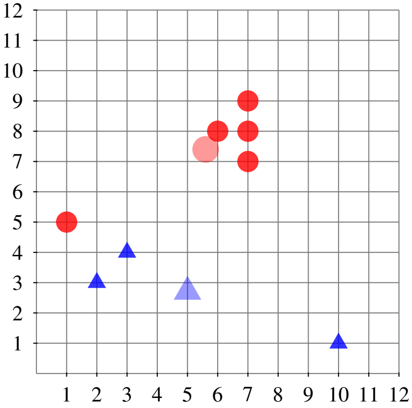
Zentroide neu berechnen:

$\mu \approx (6.0, 4.3)$

$\mu \approx (5.0, 6.4)$



Punkte neu zuordnen

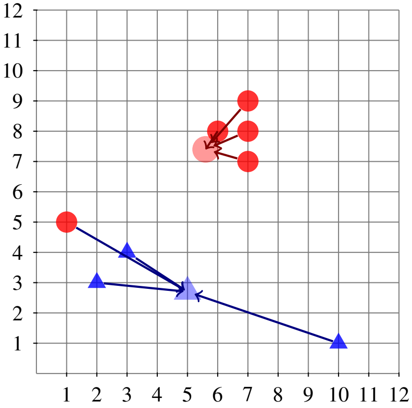


Zentroide neu berechnen:

$$\mu \approx (5.0, 2.7)$$

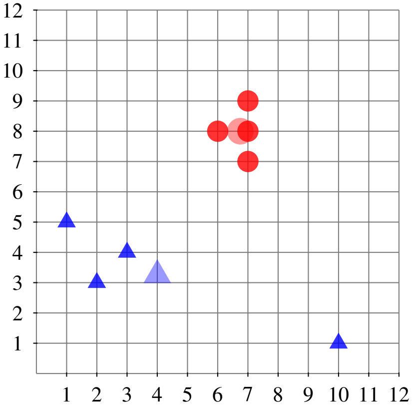
$$\mu \approx (5.6, 7.4)$$

k-Means Clustering – Lloyd/Forgy Algorithmus



Punkte neu zuordnen

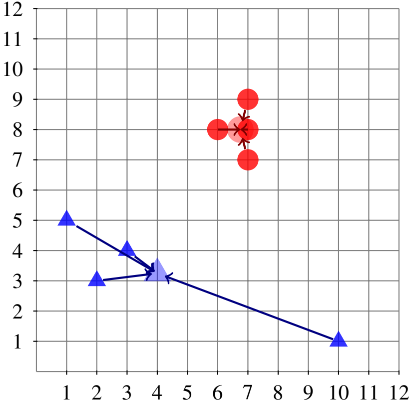
k-Means Clustering – Lloyd/Forgy Algorithmus



Zentroide neu berechnen:

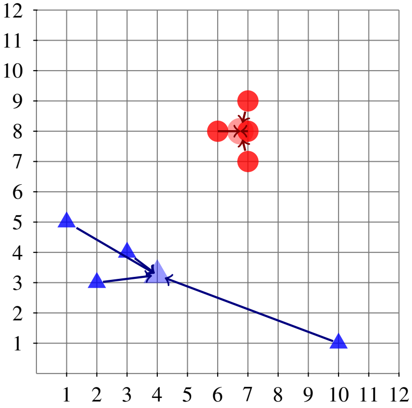
$$\mu \approx (4.0, 3.25)$$

$$\mu \approx (6.75, 8.0)$$



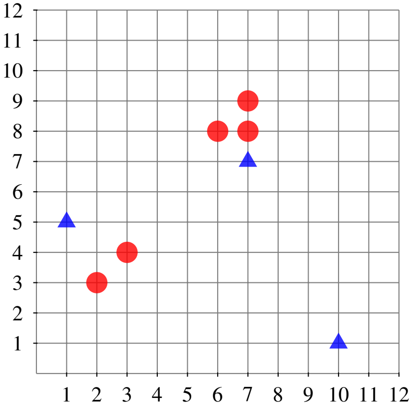
Punkte neu zuordnen

k-Means Clustering – Lloyd/Forgy Algorithmus

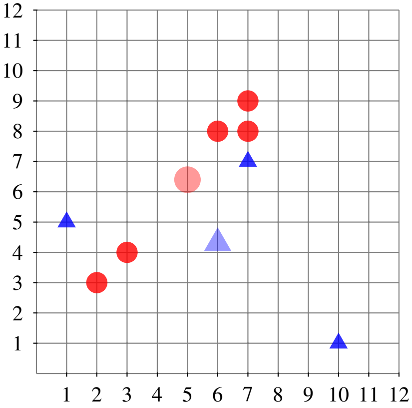


Punkte neu zuordnen
Keine Änderung
Konvergenz!

k-Means Clustering – MacQueen Algorithmus



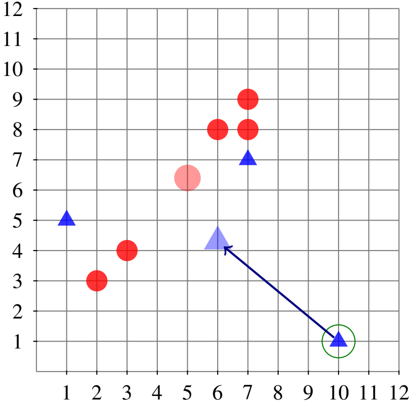
k-Means Clustering – MacQueen Algorithmus



Zentroide (z.B.: aus vorheriger Iteration):

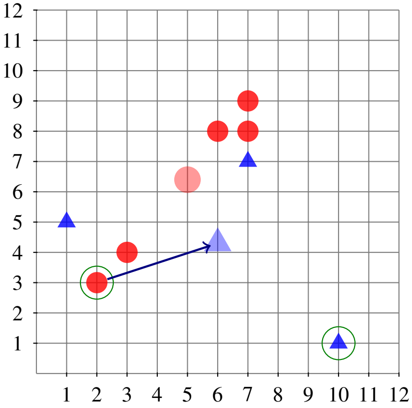
$$\mu \approx (6.0, 4.3)$$

$$\mu \approx (5.0, 6.4)$$



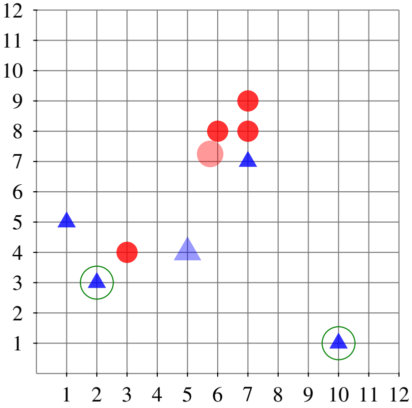
Ersten Punkt zuordnen

k-Means Clustering – MacQueen Algorithmus



Zweiten Punkt zuordnen

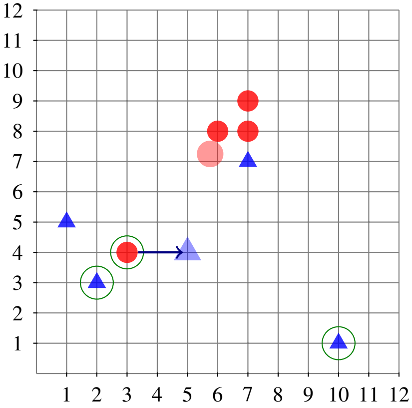
k-Means Clustering – MacQueen Algorithmus



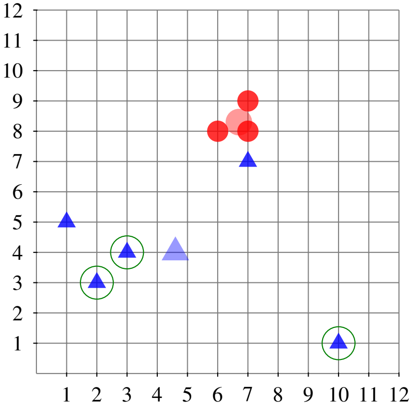
Zentroide aktualisieren:

$\mu \approx (5.0, 4.0)$

$\mu \approx (5.75, 7.25)$



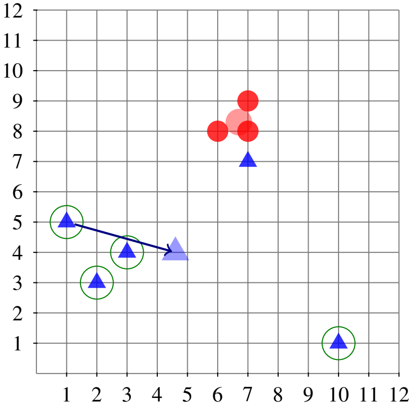
Dritten Punkt zuordnen



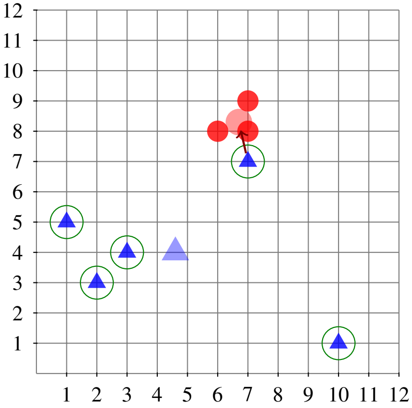
Zentroide aktualisieren:

$$\mu \approx (4.6, 4.0)$$

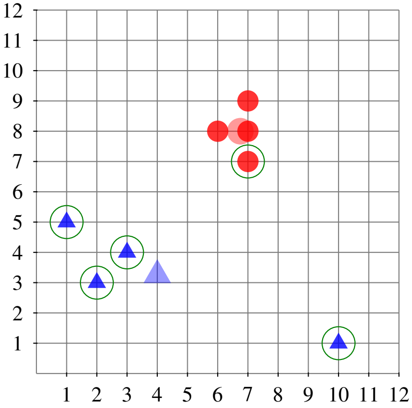
$$\mu \approx (6.7, 8.3)$$



Vierten Punkt neu zuordnen



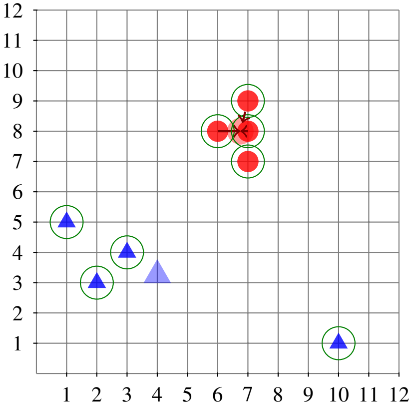
Fünften Punkt neu zuordnen



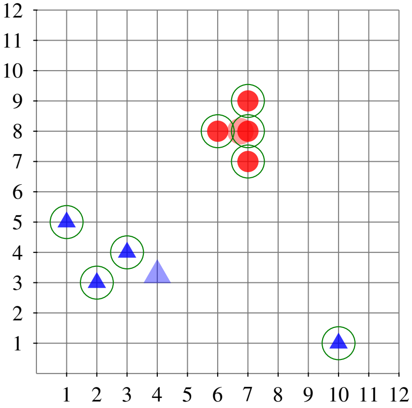
Zentroide aktualisieren:

$\mu \approx (4.0, 3.25)$

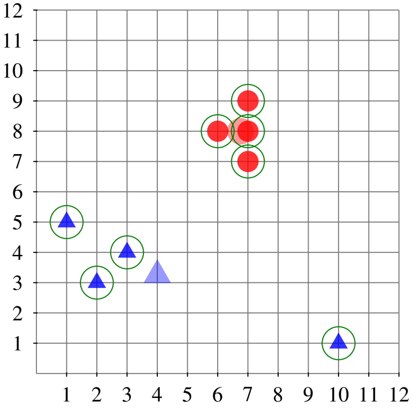
$\mu \approx (6.75, 8.0)$



Weitere Punkte neu zuordnen



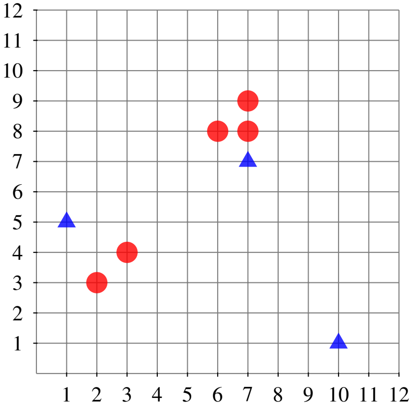
Weitere Punkte neu zuordnen
ggf. Weitere Iterationen



Weitere Punkte neu zuordnen
ggf. Weitere Iterationen
Konvergenz

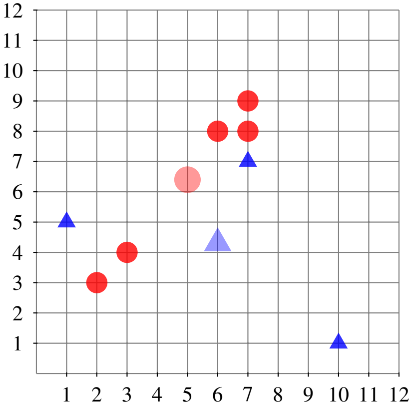
k-Means Clustering – MacQueen Algorithmus

Alternativer Ablauf – andere Reihenfolge



k-Means Clustering – MacQueen Algorithmus

Alternativer Ablauf – andere Reihenfolge



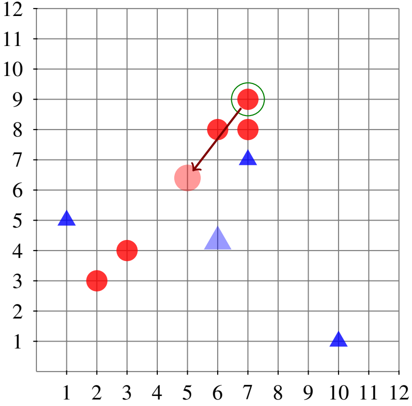
Zentroide (z.B.: aus vorheriger Iteration):

$$\mu \approx (6.0, 4.3)$$

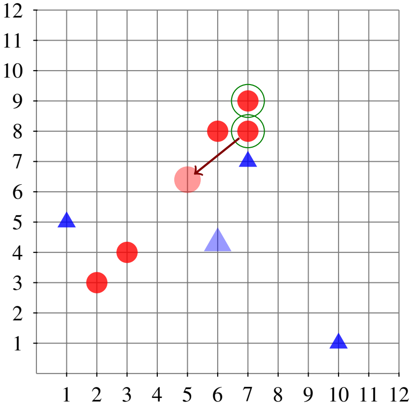
$$\mu \approx (5.0, 6.4)$$

k-Means Clustering – MacQueen Algorithmus

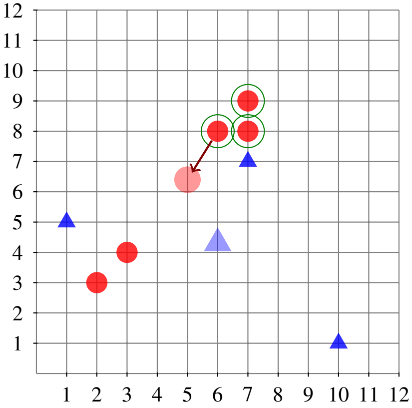
Alternativer Ablauf – andere Reihenfolge



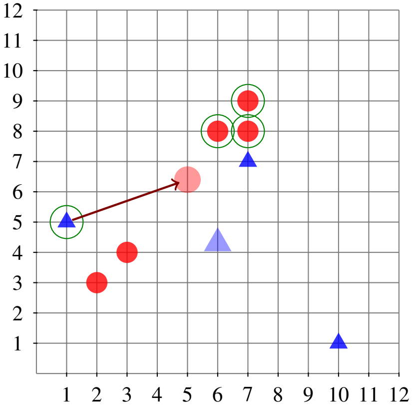
Ersten Punkt zuordnen



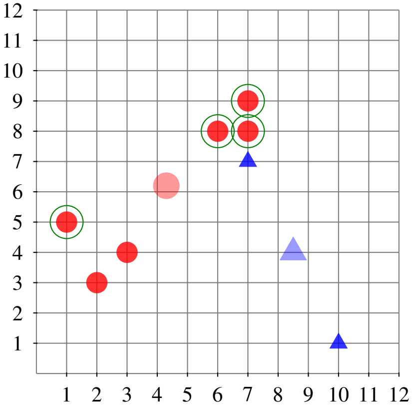
Zweiten Punkt zuordnen



Dritten Punkt zuordnen



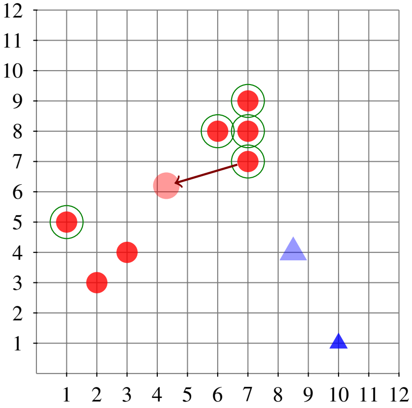
Vierten Punkt zuordnen



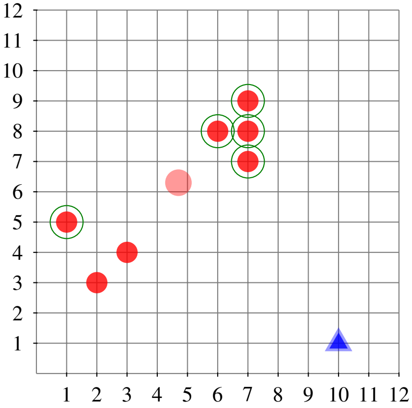
Zentroide aktualisieren:

$\mu \approx (4.0, 8.5)$

$\mu \approx (4.3, 6.2)$



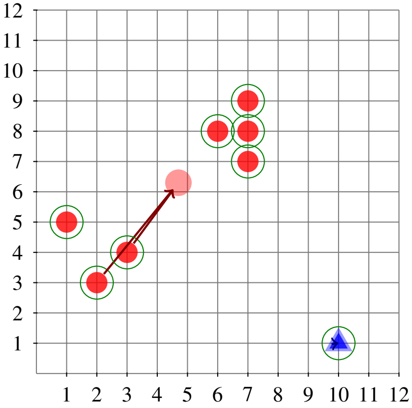
Fünften Punkt zuordnen



Zentroide aktualisieren:

$\mu \approx (10.0, 1.0)$

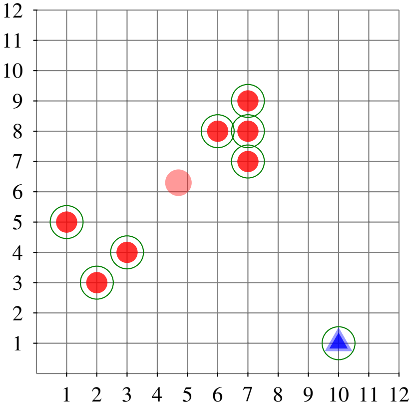
$\mu \approx (4.7, 6.3)$



Weitere Punkte neu zuordnen

k-Means Clustering – MacQueen Algorithmus

Alternativer Ablauf – andere Reihenfolge

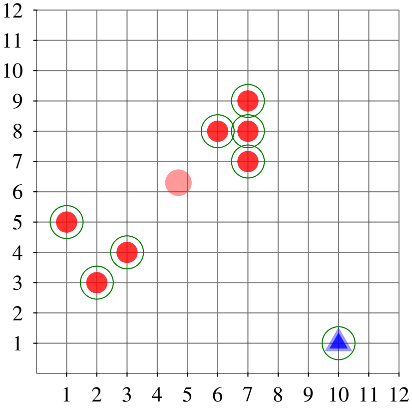


Weitere Punkte neu zuordnen

ggf. Weitere Iterationen

k-Means Clustering – MacQueen Algorithmus

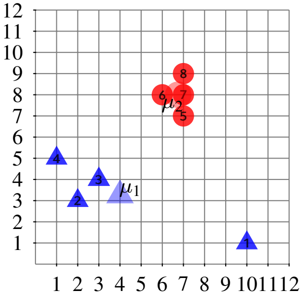
Alternativer Ablauf – andere Reihenfolge



Weitere Punkte neu zuordnen

ggf. Weitere Iterationen

Konvergenz

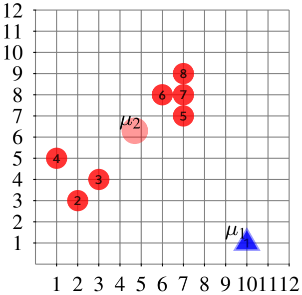


$SSQ(\mu_1, p_1) = |4-10|^2 + |3.25-1|^2 = 36 + 5\frac{1}{16} = 41\frac{1}{16}$
$SSQ(\mu_1, p_2) = |4-2|^2 + |3.25-3|^2 = 4 + \frac{1}{16} = 4\frac{1}{16}$
$SSQ(\mu_1, p_3) = |4-3|^2 + |3.25-4|^2 = 1 + \frac{9}{16} = 1\frac{9}{16}$
$SSQ(\mu_1, p_4) = |4-1|^2 + |3.25-5|^2 = 9 + 3\frac{1}{16} = 12\frac{1}{16}$
$TD^2(C_1) = 58\frac{3}{4}$

$SSQ(\mu_2, p_5) = |6.75-7|^2 + |8-7|^2 = \frac{1}{16} + 1 = 1\frac{1}{16}$
$SSQ(\mu_2, p_6) = |6.75-6|^2 + |8-8|^2 = \frac{9}{16} + 0 = \frac{9}{16}$
$SSQ(\mu_2, p_7) = |6.75-7|^2 + |8-8|^2 = \frac{1}{16} + 0 = \frac{1}{16}$
$SSQ(\mu_2, p_8) = |6.75-7|^2 + |8-9|^2 = \frac{1}{16} + 1 = 1\frac{1}{16}$
$TD^2(C_2) = 2\frac{3}{4}$

Erste Lösung: $TD^2 = 61\frac{1}{2}$

Beachte: $SSQ(\mu, p) = Euclidean(\mu, p)^2 = L_2^2(\mu, p)$.

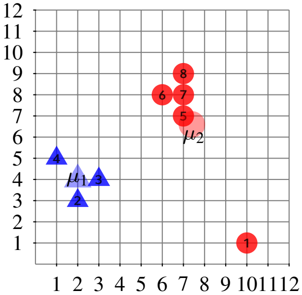


$SSQ(\mu_1, p_1) = |10 - 10|^2 + |1 - 1|^2 = 0$
$TD^2(C_1) = 0$

$SSQ(\mu_2, p_2) \approx |4.7 - 2|^2 + |6.3 - 3|^2 \approx 18.2$
$SSQ(\mu_2, p_3) \approx |4.7 - 3|^2 + |6.3 - 4|^2 \approx 8.2$
$SSQ(\mu_2, p_4) \approx |4.7 - 1|^2 + |6.3 - 5|^2 \approx 15.4$
$SSQ(\mu_2, p_5) \approx |4.7 - 7|^2 + |6.3 - 7|^2 \approx 5.7$
$SSQ(\mu_2, p_6) \approx |4.7 - 6|^2 + |6.3 - 8|^2 \approx 4.6$
$SSQ(\mu_2, p_7) \approx |4.7 - 7|^2 + |6.3 - 8|^2 \approx 8.2$
$SSQ(\mu_2, p_7) \approx |4.7 - 7|^2 + |6.3 - 9|^2 \approx 12.6$
$TD^2(C_2) \approx 72.86$

Erste Lösung: $TD^2 = 61\frac{1}{2}$
Zweite Lösung: $TD^2 \approx 72.68$

Beachte: $SSQ(\mu, p) = Euclidean(\mu, p)^2 = L_2^2(\mu, p)$.



$$SSQ(\mu_1, p_2) = |2-2|^2 + |4-3|^2 = 0 + 1 = 1$$
$$SSQ(\mu_1, p_3) = |2-3|^2 + |4-4|^2 = 1 + 0 = 1$$
$$SSQ(\mu_1, p_4) = |2-1|^2 + |4-5|^2 = 1 + 1 = 2$$
$$TD^2(C_1) = 4$$

$$SSQ(\mu_2, p_1) = |7.4-10|^2 + |6.6-1|^2 = 6\tfrac{19}{25} + 31\tfrac{9}{25} = 38\tfrac{3}{25}$$
$$SSQ(\mu_2, p_5) = |7.4-7|^2 + |6.6-7|^2 = \tfrac{4}{25} + \tfrac{4}{25} = \tfrac{8}{25}$$
$$SSQ(\mu_2, p_6) = |7.4-6|^2 + |6.6-8|^2 = 1\tfrac{24}{25} + 1\tfrac{24}{25} = 3\tfrac{23}{25}$$
$$SSQ(\mu_2, p_7) = |7.4-7|^2 + |6.6-8|^2 = \tfrac{4}{25} + 1\tfrac{24}{25} = 2\tfrac{3}{25}$$
$$SSQ(\mu_2, p_8) = |7.4-7|^2 + |6.6-9|^2 = \tfrac{4}{25} + 5\tfrac{19}{25} = 5\tfrac{23}{25}$$
$$TD^2(C_2) = 50\tfrac{2}{5}$$

Erste Lösung: $TD^2 = 61\frac{1}{2}$
Zweite Lösung: $TD^2 \approx 72.68$
Optimale Lösung: $TD^2 = 54\frac{2}{5}$

Beachte: $SSQ(\mu, p) = \mathit{Euclidean}(\mu, p)^2 = L_2^2(\mu, p)$.

Fazit k-Means Clustering

Merke:

- K-means konvergiert nur gegen ein lokales Minimum
- K-means ist abhängig von den Startparametern
- K-means nach MacQueen ist reihenfolgeabhängig
- K-means ist anfällig gegen Rauschen
  - Degenerierte 1-Element "Cluster"
  - Dadurch Reduktion von effektivem k
- K-means minimiert Varianzen, ist also eigentlich nur für Euklidische Distanz korrekt (oftmals aber auch Konvergenz bei anderen Distanzen)
- K-means (nach Lloyd) ist dennoch das beliebteste Verfahren, da es sehr einfach und schnell ist und mit geringem Aufwand implementiert werden kann!

Konvergenz von PAM

Sei:

- M_i = Modell (= Menge der Medoide) nach Iteration i.
- $TD(M_i)$ = Kosten von Modell M_i.

Konvergenz von PAM

Sei:

- M_i = Modell (= Menge der Medoide) nach Iteration i.
- $TD(M_i)$ = Kosten von Modell M_i.

PAM konvergiert, wenn

$$TD(M_i) = TD(M_{i+1})$$

Konvergenz von PAM

Sei:

- M_i = Modell (= Menge der Medoide) nach Iteration i.
- $TD(M_i)$ = Kosten von Modell M_i.

PAM konvergiert nicht, wenn

$$\forall i \in \mathbb{N} : TD(M_i) > TD(M_{i+1})$$

Konvergenz von PAM

Sei:

- M_i = Modell (= Menge der Medoide) nach Iteration i.
- $TD(M_i)$ = Kosten von Modell M_i.

PAM konvergiert nicht, wenn

$$\forall i \in \mathbb{N} : TD(M_i) > TD(M_{i+1})$$

Es gibt $\binom{n}{k}$ mögliche Modelle,

Konvergenz von PAM

Sei:

- M_i = Modell (= Menge der Medoide) nach Iteration i.
- $TD(M_i)$ = Kosten von Modell M_i.

PAM konvergiert nicht, wenn

$$\forall i \in \mathbb{N} : TD(M_i) > TD(M_{i+1})$$

Es gibt $\binom{n}{k}$ mögliche Modelle, aber $\mathbb{N}$ Iterationen.

Konvergenz von PAM

Sei:

- M_i = Modell (= Menge der Medoide) nach Iteration i.
- $TD(M_i)$ = Kosten von Modell M_i.

PAM konvergiert nicht, wenn

$$\forall i \in \mathbb{N} : TD(M_i) > TD(M_{i+1})$$

Es gibt $\binom{n}{k}$ mögliche Modelle, aber $\mathbb{N}$ Iterationen.

$$\Rightarrow \exists_{t_1 < t_2} M_{t_1} = M_{t_2}$$

Konvergenz von PAM

Sei:

- M_i = Modell (= Menge der Medoide) nach Iteration i.
- $TD(M_i)$ = Kosten von Modell M_i.

PAM konvergiert nicht, wenn

$$\forall i \in \mathbb{N} : TD(M_i) > TD(M_{i+1})$$

Es gibt $\binom{n}{k}$ mögliche Modelle, aber $\mathbb{N}$ Iterationen.

$$\Rightarrow \exists_{t_1 < t_2} M_{t_1} = M_{t_2}$$

$$\Rightarrow TD(M_{t_1}) = TD(M_{t_2})$$

Konvergenz von PAM

Sei:

- M_i = Modell (= Menge der Medoide) nach Iteration i.
- $TD(M_i)$ = Kosten von Modell M_i.

PAM konvergiert nicht, wenn

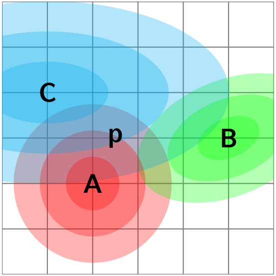
$$\forall i \in \mathbb{N} : TD(M_i) > TD(M_{i+1})$$

Es gibt $\binom{n}{k}$ mögliche Modelle, aber $\mathbb{N}$ Iterationen.

$$\Rightarrow \exists_{t_1 < t_2} M_{t_1} = M_{t_2}$$

$$\Rightarrow TD(M_{t_1}) = TD(M_{t_2}) \quad \text{Wiederspruch!}$$

Multivariate Normalverteilung

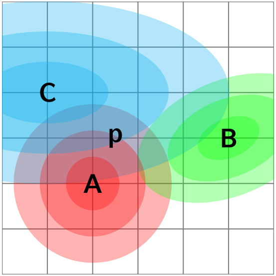


Cluster können sein:

- Kugelförmig (A)
- Ellipsoid (C)
- Rotierter Ellipsoid (B)

Aber: es ist immer die gleiche Formel!

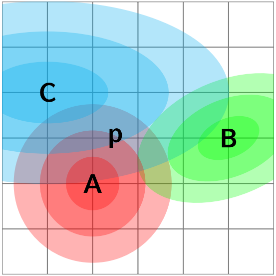
Multivariate Normalverteilung



Cluster können sein:

- Kugelförmig (A)
- Ellipsoid (C)
- Rotierter Ellipsoid (B)

Aber: es ist immer die gleiche Formel!

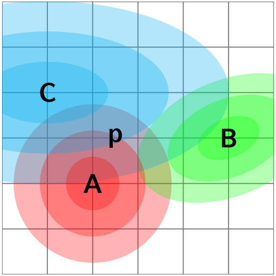


Wahrscheinlichkeitsdichte (PDF) der multivariaten Normalverteilung:

$$pdf(p, \mu, \Sigma) := \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \cdot e^{-\frac{1}{2}\left((p-\mu)^T \Sigma^{-1} (p-\mu)\right)}$$

Die wichtigste mehrdimensionale Verteilungsfunktion!

Multivariate Normalverteilung



Wahrscheinlichkeitsdichte (PDF) der multivariaten Normalverteilung:

$$pdf(p, \mu, \Sigma) := \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \cdot e^{-\frac{1}{2}\left((p-\mu)^T \Sigma^{-1} (p-\mu)\right)}$$

Die wichtigste mehrdimensionale Verteilungsfunktion!

Das sollten wir uns genauer anschauen!

Multivariate Normalverteilung:

$$pdf(p,\mu,\Sigma) := \frac{1}{\sqrt{(2\pi)^d|\Sigma|}} \cdot e^{-\frac{1}{2}\left((p-\mu)^T\Sigma^{-1}(p-\mu)\right)}$$

Multivariate Normalverteilung:

$$pdf(p, \mu, \Sigma) := \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \cdot e^{-\frac{1}{2}\left((p-\mu)^T \Sigma^{-1} (p-\mu)\right)}$$

1-dimensionale Normalverteilung

$$pdf(x, \mu, \sigma) := \frac{1}{\sqrt{(2\pi)}\sigma} \cdot e^{-\frac{1}{2}\left(\frac{(x-\mu)}{\sigma}\right)^2}$$

Multivariate Normalverteilung:

$$pdf(p,\mu,\Sigma) := \frac{1}{\sqrt{(2\pi)^d|\Sigma|}} \cdot e^{-\frac{1}{2}\left((p-\mu)^T\Sigma^{-1}(p-\mu)\right)}$$

1-dimensionale Normalverteilung

$$pdf(x,\mu,\sigma) := \frac{1}{\sqrt{(2\pi)}\sigma} \cdot e^{-\frac{1}{2}\left(\frac{(x-\mu)}{\sigma}\right)^2}$$

Multivariate Normalverteilung:

$$pdf(p,\mu,\Sigma) := \frac{1}{\sqrt{(2\pi)^d|\Sigma|}} \cdot e^{-\frac{1}{2}\left((p-\mu)^T\Sigma^{-1}(p-\mu)\right)}$$

1-dimensionale Normalverteilung

$$pdf(x,\mu,\sigma) := \frac{1}{\sqrt{(2\pi)}\sigma} \cdot e^{-\frac{1}{2}\left(\frac{(x-\mu)}{\sigma}\right)^2}$$

Normalisierung (auf Gesamtvolumen 1!) und quadrierte Abweichung vom Mittelwert

Multivariate Normalverteilung:

$$pdf(p, \mu, \Sigma) := \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \cdot e^{-\frac{1}{2}\left((p-\mu)^T \Sigma^{-1} (p-\mu)\right)}$$

1-dimensionale Normalverteilung

$$pdf(x, \mu, \sigma) := \frac{1}{\sqrt{(2\pi)\sigma^2}} \cdot e^{-\frac{1}{2}\left((x-\mu)\sigma^{-2}(x-\mu)\right)}$$

Normalisierung (auf Gesamtvolumen 1!) und quadrierte Abweichung vom Mittelwert

Multivariate Normalverteilung:

$$pdf(p,\mu,\Sigma) := \frac{1}{\sqrt{(2\pi)^d|\Sigma|}} \cdot e^{-\frac{1}{2}\left((p-\mu)^T\Sigma^{-1}(p-\mu)\right)}$$

1-dimensionale Normalverteilung

$$pdf(x,\mu,\sigma) := \frac{1}{\sqrt{(2\pi)\sigma^2}} \cdot e^{-\frac{1}{2}\left((x-\mu)\sigma^{-2}(x-\mu)\right)}$$

Mahalanobis-Distanz:

$$d_{Mahalanobis}(x,\mu,\Sigma) := \sqrt{(x-\mu)^T\Sigma^{-1}(x-\mu)}$$

Multivariate Normalverteilung:

$$pdf(p, \mu, \Sigma) := \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \cdot e^{-\frac{1}{2}\left((p-\mu)^T \Sigma^{-1} (p-\mu)\right)}$$

1-dimensionale Normalverteilung

$$pdf(x, \mu, \sigma) := \frac{1}{\sqrt{(2\pi)\sigma^2}} \cdot e^{-\frac{1}{2}\left((x-\mu)\sigma^{-2}(x-\mu)\right)}$$

Mahalanobis-Distanz:

$$d_{Mahalanobis}(x, \mu, \Sigma)^2 := (x-\mu)^T \Sigma^{-1} (x-\mu)$$

Multivariate Normalverteilung:

$$pdf(p, \mu, \Sigma) := \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \cdot e^{-\frac{1}{2}\left((p-\mu)^T \Sigma^{-1} (p-\mu)\right)}$$

1-dimensionale Normalverteilung

$$pdf(x, \mu, \sigma) := \frac{1}{\sqrt{(2\pi)\sigma^2}} \cdot e^{-\frac{1}{2}\left((x-\mu)\sigma^{-2}(x-\mu)\right)}$$

Mahalanobis-Distanz:

$$d_{Mahalanobis}(x, \mu, \Sigma)^2 := (x-\mu)^T \Sigma^{-1} (x-\mu)$$

Was ist die Rolle von Σ^{-1}?

Kovarianzmatrizen sind symmetrisch und auf der Diagonalen nicht negativ, und können daher normalerweise invertiert werden (es gibt degenerierte Fälle, auch die kann man handhaben!)

Kovarianzmatrizen sind symmetrisch und auf der Diagonalen nicht negativ, und können daher normalerweise invertiert werden (es gibt degenerierte Fälle, auch die kann man handhaben!)
Die Matrix kann zerlegt werden:

$$V \Lambda V^{-1} = \Sigma \quad \equiv \quad V \Lambda^{-1} V^{-1} = \Sigma^{-1}$$

wobei V die Eigenvektoren und Λ die Eigenwerte enthält.

Kovarianzmatrizen sind symmetrisch und auf der Diagonalen nicht negativ, und können daher normalerweise invertiert werden (es gibt degenerierte Fälle, auch die kann man handhaben!)
Die Matrix kann zerlegt werden:

$$V \Lambda V^{-1} = \Sigma \quad \equiv \quad V \Lambda^{-1} V^{-1} = \Sigma^{-1}$$

wobei V die Eigenvektoren und Λ die Eigenwerte enthält.
$V \cong$ Drehung, $\Lambda \cong$ Skalierung^2!
(Das ist die Kernidee der Hauptachsentransformation=PCA)

Konstruiere Ω als $\omega_i = 1/\sqrt{\lambda_i} = \lambda_i^{-\frac{1}{2}}$. Dann gilt $\Omega\Omega = \Lambda^{-1}$.

Konstruiere Ω als $\omega_i = 1/\sqrt{\lambda_i} = \lambda_i^{-\frac{1}{2}}$. Dann gilt $\Omega\Omega = \Lambda^{-1}$.

$$\Sigma^{-1} = V\Lambda^{-1}V^{-1} = V\Omega\Omega^T V^T = V\Omega(V\Omega)^T$$

Konstruiere Ω als $\omega_i = 1/\sqrt{\lambda_i} = \lambda_i^{-\frac{1}{2}}$. Dann gilt $\Omega\Omega = \Lambda^{-1}$.

$$\Sigma^{-1} = V\Lambda^{-1}V^{-1} = V\Omega\Omega^T V^T = V\Omega(V\Omega)^T$$

$$d^2_{\text{Mahalanobis}} = (x-\mu)^T\Sigma^{-1}(x-\mu)$$

Konstruiere Ω als $\omega_i = 1/\sqrt{\lambda_i} = \lambda_i^{-\frac{1}{2}}$. Dann gilt $\Omega\Omega = \Lambda^{-1}$.

$$\Sigma^{-1} = V\Lambda^{-1}V^{-1} = V\Omega\Omega^T V^T = V\Omega(V\Omega)^T$$

$$d^2_{\text{Mahalanobis}} = (x-\mu)^T V\Omega(V\Omega)^T(x-\mu)$$

Konstruiere Ω als $\omega_i = 1/\sqrt{\lambda_i} = \lambda_i^{-\frac{1}{2}}$. Dann gilt $\Omega\Omega = \Lambda^{-1}$.

$$\Sigma^{-1} = V\Lambda^{-1}V^{-1} = V\Omega\Omega^T V^T = V\Omega(V\Omega)^T$$

$$\begin{aligned} d^2_{\text{Mahalanobis}} &= (x-\mu)^T V\Omega(V\Omega)^T(x-\mu) \\ &= \left\langle (V\Omega)^T(x-\mu), (V\Omega)^T(x-\mu) \right\rangle \end{aligned}$$

Konstruiere Ω als $\omega_i = 1/\sqrt{\lambda_i} = \lambda_i^{-\frac{1}{2}}$. Dann gilt $\Omega\Omega = \Lambda^{-1}$.

$$\Sigma^{-1} = V\Lambda^{-1}V^{-1} = V\Omega\Omega^T V^T = V\Omega(V\Omega)^T$$

$$\begin{aligned} d^2_{\text{Mahalanobis}} &= (x-\mu)^T V\Omega(V\Omega)^T(x-\mu) \\ &= \left\langle (V\Omega)^T(x-\mu), (V\Omega)^T(x-\mu) \right\rangle \\ &= L_2((V\Omega)^T(x-\mu))^2 \end{aligned}$$

Konstruiere Ω als $\omega_i = 1/\sqrt{\lambda_i} = \lambda_i^{-\frac{1}{2}}$. Dann gilt $\Omega\Omega = \Lambda^{-1}$.

$$\Sigma^{-1} = V\Lambda^{-1}V^{-1} = V\Omega\Omega^T V^T = V\Omega(V\Omega)^T$$

$$\begin{aligned} d^2_{\text{Mahalanobis}} &= (x-\mu)^T V\Omega(V\Omega)^T(x-\mu) \\ &= \left\langle (V\Omega)^T(x-\mu), (V\Omega)^T(x-\mu) \right\rangle \\ &= L_2((V\Omega)^T(x-\mu))^2 \end{aligned}$$

L_2 ist die L_2-Norm (Euclidische Distanz $d(x,y) = L_2(x-y)$!)

Multivariate Normalverteilung

Inverse Kovarianzmatrix – Σ^{-1}

Konstruiere Ω als $\omega_i = 1/\sqrt{\lambda_i} = \lambda_i^{-\frac{1}{2}}$. Dann gilt $\Omega\Omega = \Lambda^{-1}$.

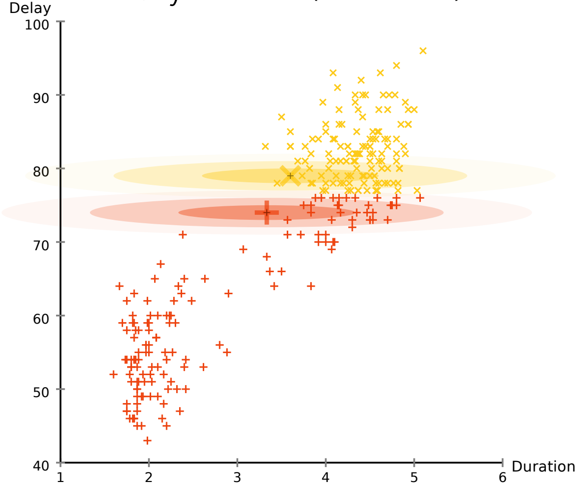
$$\Sigma^{-1} = V\Lambda^{-1}V^{-1} = V\Omega\Omega^T V^T = V\Omega(V\Omega)^T$$

$$\begin{aligned} d^2_{\text{Mahalanobis}} &= (x-\mu)^T V\Omega(V\Omega)^T(x-\mu) \\ &= \left\langle (V\Omega)^T(x-\mu), (V\Omega)^T(x-\mu) \right\rangle \\ &= L_2((V\Omega)^T(x-\mu))^2 \end{aligned}$$

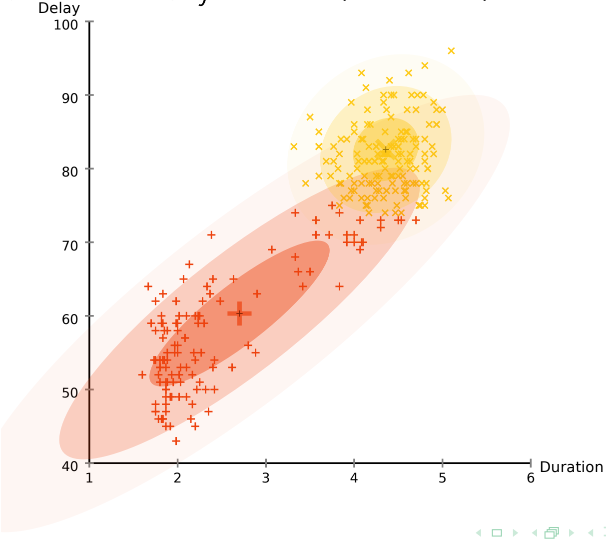
L_2 ist die L_2-Norm (Euclidische Distanz $d(x,y) = L_2(x-y)$!)

Mahalanobis $\approx$ Euclidische Distanz nach PCA!

Old Faithful Geysir Daten (Iteration 0):

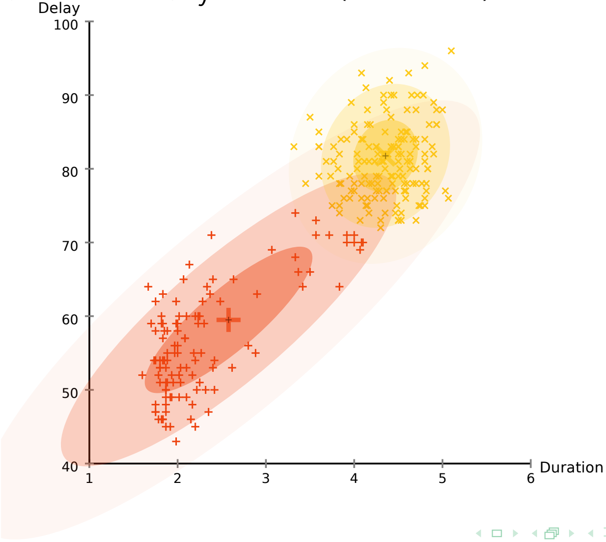


Old Faithful Geysir Daten (Iteration 1):

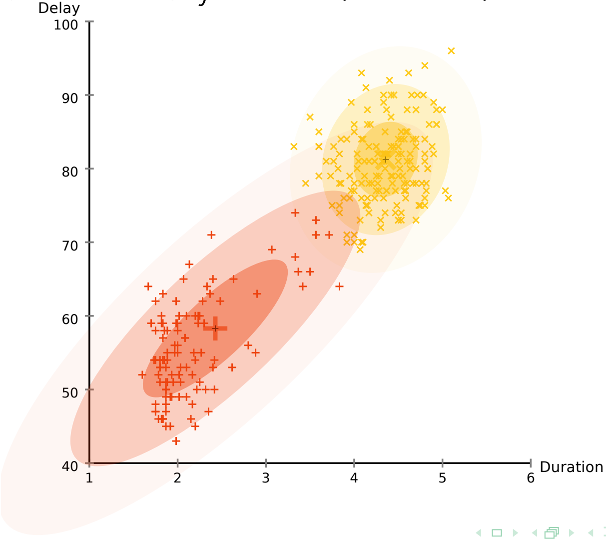


EM Demonstration

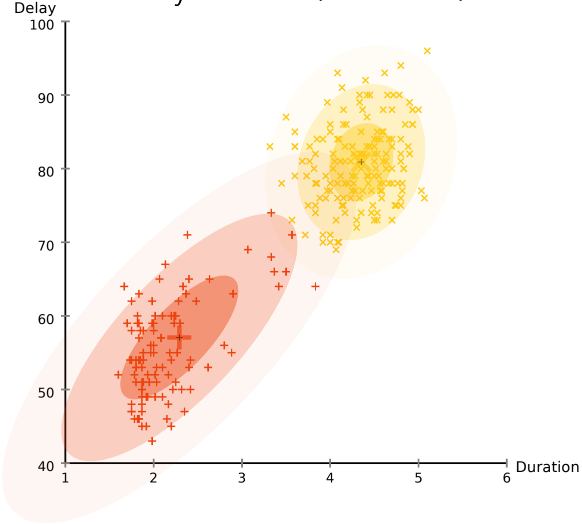
Old Faithful Geysir Daten (Iteration 2):



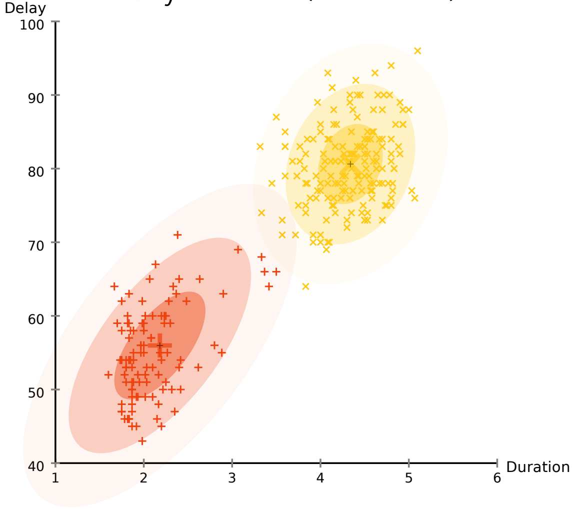
Old Faithful Geysir Daten (Iteration 3):



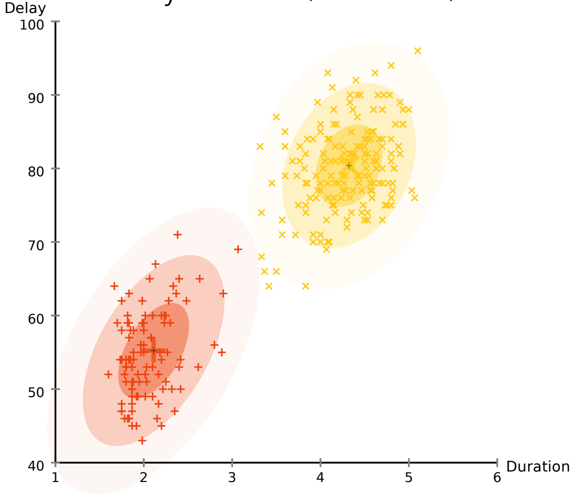
Old Faithful Geysir Daten (Iteration 4):



Old Faithful Geysir Daten (Iteration 5):

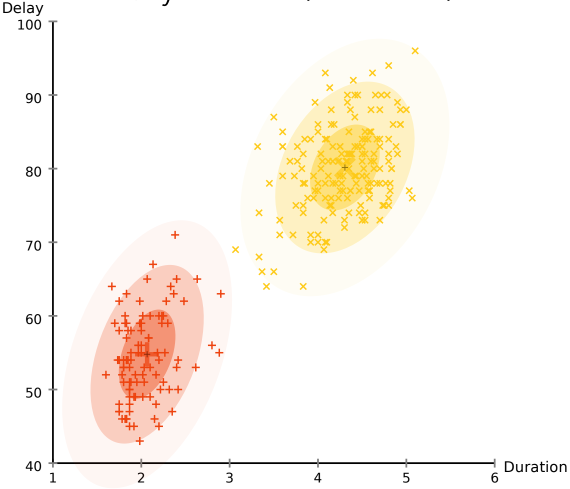


Old Faithful Geysir Daten (Iteration 6):

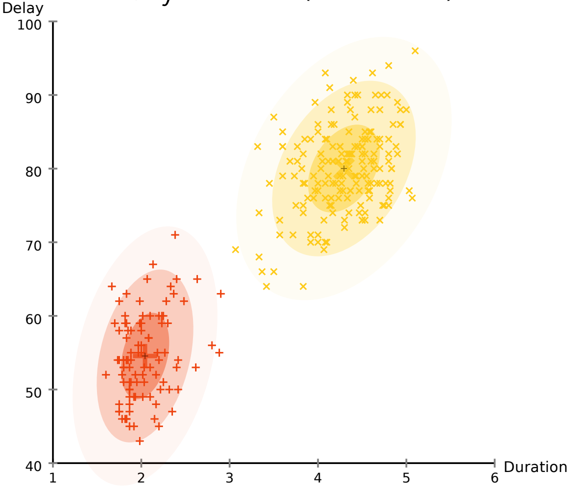


EM Demonstration

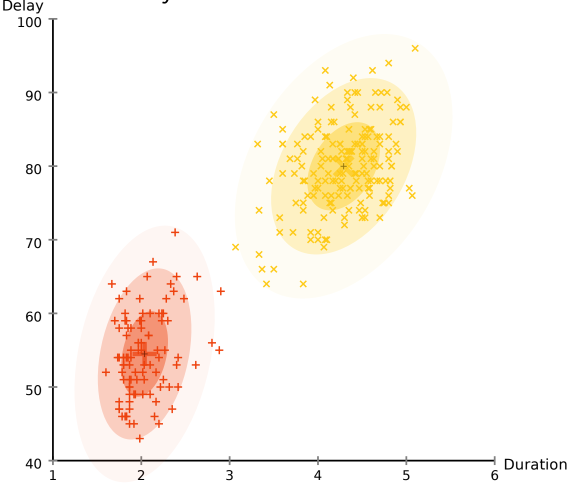
Old Faithful Geysir Daten (Iteration 7):



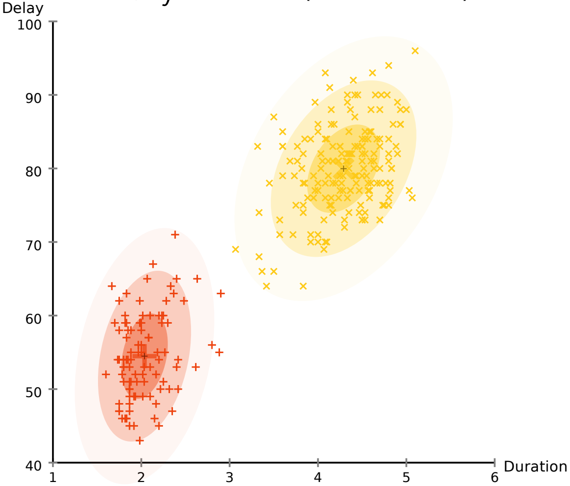
Old Faithful Geysir Daten (Iteration 8):



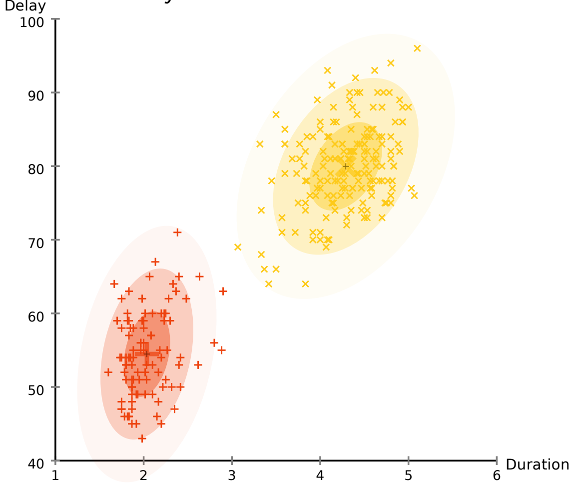
Old Faithful Geysir Daten (Iteration 9):



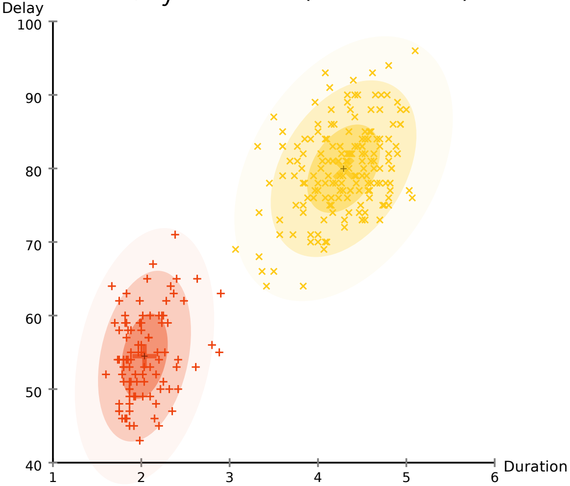
Old Faithful Geysir Daten (Iteration 10):



Old Faithful Geysir Daten (Iteration 11):

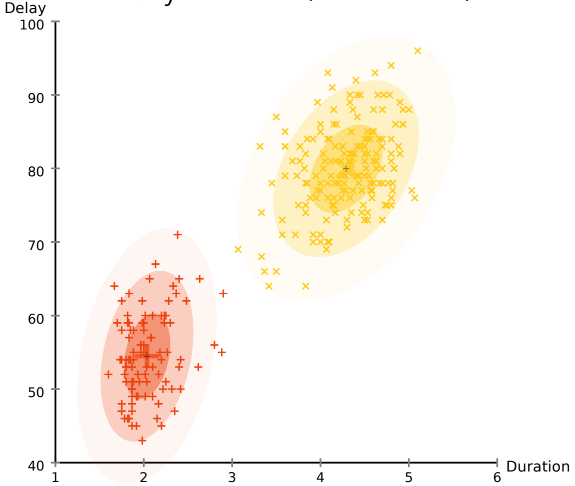


Old Faithful Geysir Daten (Iteration 12):

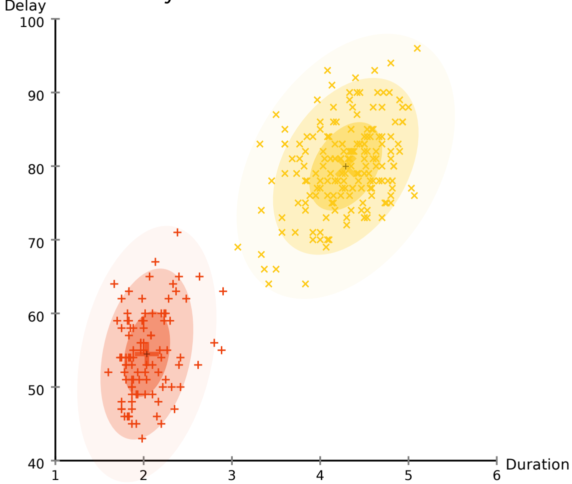


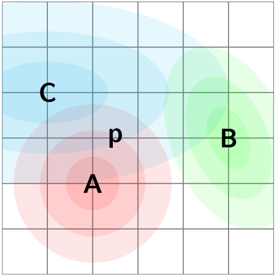
EM Demonstration

Old Faithful Geysir Daten (Iteration 13):

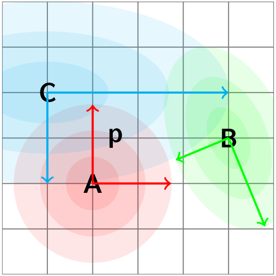


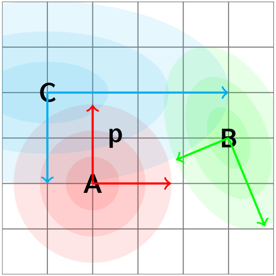
Old Faithful Geysir Daten (Iteration 14):



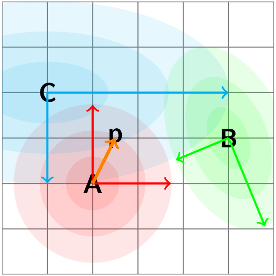

C
p
B
A

EM Hausaufgabe



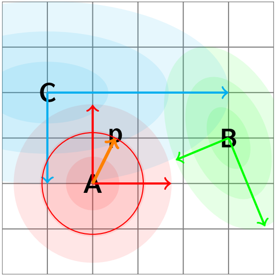


$$\Sigma_A = \begin{pmatrix} 3 & 0 \\ 0 & 3 \end{pmatrix} \quad \Sigma_A^{-1} = \begin{pmatrix} \frac{1}{3} & 0 \\ 0 & \frac{1}{3} \end{pmatrix}$$



$$\Sigma_A = \begin{pmatrix} 3 & 0 \\ 0 & 3 \end{pmatrix} \quad \Sigma_A^{-1} = \begin{pmatrix} \frac{1}{3} & 0 \\ 0 & \frac{1}{3} \end{pmatrix}$$

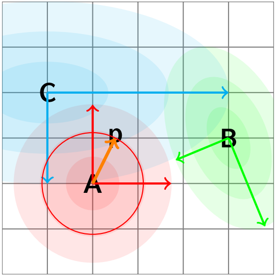
$$p - \mu_A = (0.5, 1)$$



$$\Sigma_A = \begin{pmatrix} 3 & 0 \\ 0 & 3 \end{pmatrix} \quad \Sigma_A^{-1} = \begin{pmatrix} \frac{1}{3} & 0 \\ 0 & \frac{1}{3} \end{pmatrix}$$

$$p - \mu_A = (0.5, 1)$$

$$dist^2 = (p-\mu)^T \Sigma^{-1} (p-\mu) \approx 0.41666$$



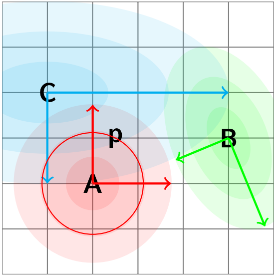
$$\Sigma_A = \begin{pmatrix} 3 & 0 \\ 0 & 3 \end{pmatrix} \quad \Sigma_A^{-1} = \begin{pmatrix} \frac{1}{3} & 0 \\ 0 & \frac{1}{3} \end{pmatrix}$$

$$p - \mu_A = (0.5, 1)$$

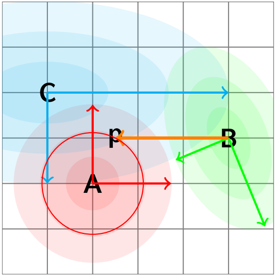
$$dist^2 = (p-\mu)^T \Sigma^{-1} (p-\mu) \approx 0.41666$$

$$prob_A \approx \frac{1}{\sqrt{(2\pi)^2 \cdot 9}} e^{-\frac{1}{2}0.41666}$$

$$\approx 0.04307456$$

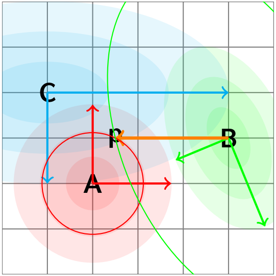


$$\Sigma_B^{-1} \approx \begin{pmatrix} 0.571428 & -0.142857 \\ -0.142857 & 0.285714 \end{pmatrix}$$



$$\Sigma_B^{-1} \approx \begin{pmatrix} 0.571428 & -0.142857 \\ -0.142857 & 0.285714 \end{pmatrix}$$

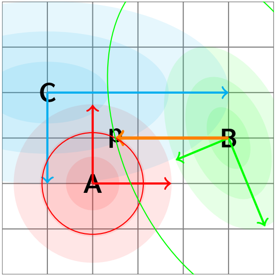
$$p - \mu_B = (-2.5, 0)$$



$$\Sigma_B^{-1} \approx \begin{pmatrix} 0.571428 & -0.142857 \\ -0.142857 & 0.285714 \end{pmatrix}$$

$$p - \mu_B = (-2.5, 0)$$

$$dist^2 = (p-\mu)^T \Sigma^{-1} (p-\mu) \approx 3.5714285$$



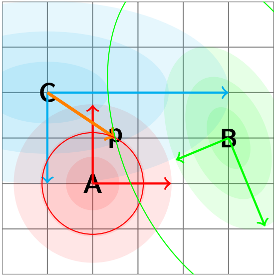
$$\Sigma_B^{-1} \approx \begin{pmatrix} 0.571428 & -0.142857 \\ -0.142857 & 0.285714 \end{pmatrix}$$

$$p - \mu_B = (-2.5, 0)$$

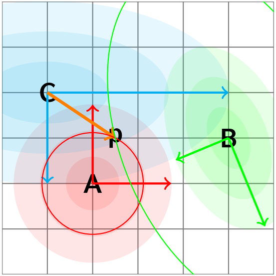
$$dist^2 = (p-\mu)^T \Sigma^{-1} (p-\mu) \approx 3.5714285$$

$$prob_B \approx \frac{1}{\sqrt{(2\pi)^2 \cdot 7}} e^{-\frac{1}{2} 3.5714285}$$

$$\approx 0.01008661$$

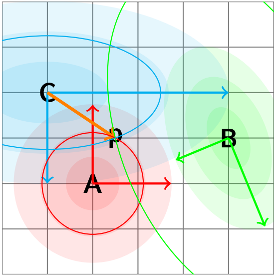


$$\Sigma_C = \begin{pmatrix} 16 & 0 \\ 0 & 4 \end{pmatrix} \qquad \Sigma_C^{-1} = \begin{pmatrix} \frac{1}{16} & 0 \\ 0 & \frac{1}{4} \end{pmatrix}$$



$$\Sigma_C = \begin{pmatrix} 16 & 0 \\ 0 & 4 \end{pmatrix} \quad \Sigma_C^{-1} = \begin{pmatrix} \frac{1}{16} & 0 \\ 0 & \frac{1}{4} \end{pmatrix}$$

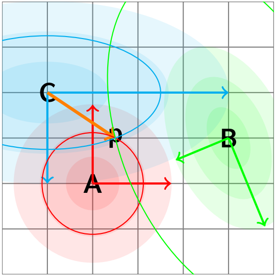
$$p - \mu_C = (1.5, -1)$$



$$\Sigma_C = \begin{pmatrix} 16 & 0 \\ 0 & 4 \end{pmatrix} \quad \Sigma_C^{-1} = \begin{pmatrix} \frac{1}{16} & 0 \\ 0 & \frac{1}{4} \end{pmatrix}$$

$$p - \mu_C = (1.5, -1)$$

$$dist^2 = (p-\mu)^T \Sigma^{-1} (p-\mu) \approx 0.390625$$



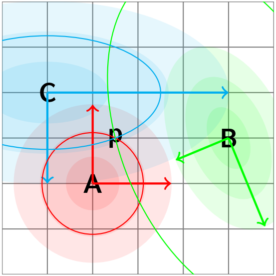
$$\Sigma_C = \begin{pmatrix} 16 & 0 \\ 0 & 4 \end{pmatrix} \quad \Sigma_C^{-1} = \begin{pmatrix} \frac{1}{16} & 0 \\ 0 & \frac{1}{4} \end{pmatrix}$$

$$p - \mu_C = (1.5, -1)$$

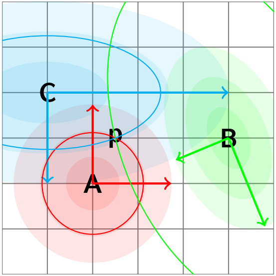
$$dist^2 = (p-\mu)^T \Sigma^{-1} (p-\mu) \approx 0.390625$$

$$prob_C \approx \frac{1}{\sqrt{(2\pi)^2 \cdot 64}} e^{-\frac{1}{2}0.390625}$$

$$\approx 0.01636466$$

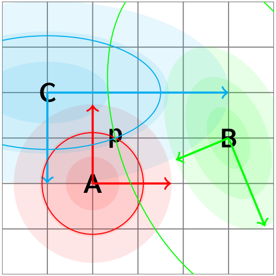


	A	B	C
prob	0.043075	0.010087	0.016365
size	30%	20%	50%

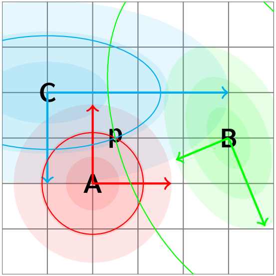


	A	B	C
prob	0.043075	0.010087	0.016365
size	30%	20%	50%
score	0.012922	0.002017	0.008182

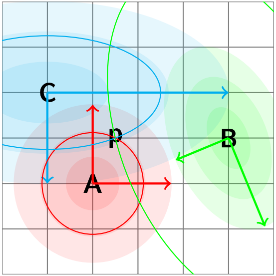
EM Hausaufgabe



	A	B	C
prob	0.043075	0.010087	0.016365
size	30%	20%	50%
score	0.012922	0.002017	0.008182
sum	dividiert durch 0.023122		



	A	B	C
prob	0.043075	0.010087	0.016365
size	30%	20%	50%
score	0.012922	0.002017	0.008182
sum	dividiert durch 0.023122		
weight	$\approx 55.9\%$	$\approx 8.2\%$	$\approx 35.4\%$



	A	B	C
prob	0.043075	0.010087	0.016365
size	30%	20%	50%
score	0.012922	0.002017	0.008182
sum	dividiert durch 0.023122		
weight	$\approx 55.9\%$	$\approx 8.2\%$	$\approx 35.4\%$

Punkt p gehört primär zu Cluster A!