

# **ISODATA Algorithmus**

**Beleg für das Fach  
„Mustererkennung“**

**Kay Wokoun**  
Matrikelnr.: 211186  
Studiengang Informatik

**Steffen Dubetz**  
Matrikelnr.: 210819  
Studiengang Informatik

## **Inhaltsverzeichnis**

1. Der ISODATA Algorithmus .....	4
1.1 Algorithmus.....	4
1.2 Beispiel.....	9
2. Bedienungsanleitung .....	18
2.1 Startbildschirm .....	18
2.2 Muster manuell anlegen .....	19
2.3 Initialisierungsparameter .....	21
2.4 Visualisierung ändern.....	23
2.5 Daten laden und speichern .....	25
2.6 Debugging .....	27
2.7 Algorithmus starten .....	28

## **Abbildungsverzeichnis**

Abbildung 1: Beispiel Initial-Cluster .....	9
Abbildung 2: Beispiel Clusterergebnis .....	17
Abbildung 3: Startbildschirm .....	18
Abbildung 4: Muster setzen .....	19
Abbildung 5: Prototypen setzen .....	20
Abbildung 6: Muster wechseln .....	20
Abbildung 7: Musterkoordinaten .....	21
Abbildung 8: Falsche Koordinateneingabe .....	21
Abbildung 9: Initialisierungsparameter.....	21
Abbildung 10: Anzeige (Defaulteinstellung) .....	23
Abbildung 11: Formular (Defaulteinstellung).....	23
Abbildung 12: Anzeige (ohne Muster).....	24
Abbildung 13: Formular (ohne Muster) .....	24
Abbildung 14: Anzeige (ohne Linien) .....	25
Abbildung 15: Formular (ohne Linien).....	25
Abbildung 16: Daten speichern.....	26
Abbildung 17: Daten laden .....	26
Abbildung 18: Debugging .....	27
Abbildung 19: Debugfenster (leer) .....	27
Abbildung 20: Debugfenster (gefüllt) .....	28
Abbildung 21: Algorithmus starten.....	28

# 1. Der ISODATA Algorithmus

Der ISODATA Algorithmus (Iterative Selbst-Organisierende Analyse Techniken) wird sehr oft in vielen Applikationen genutzt. Wie beim K-Means Algorithmus werden die Clusterzentren iterativ aus der Durchschnittsentfernung ihrer Muster ermittelt. Zusätzlich werden verschiedenen heuristischen Funktionen mit einbezogen, welche erfolgreich in einer Vielzahl von Applikationen implementiert wurden. Der Benutzer des Algorithmus muss eine klare Vorstellung von der gewünschten Anzahl der Cluster haben. Die maximale Clusteranzahl wird diesen Parameter nicht mehr als das doppelte und nicht weniger als die Hälfte übersteigen bzw. unterschreiten.

Betrachtet werden die Muster  $X = \{x_1, x_2, \dots, x_m\}$  mit  $c$  Initialen Clusterzentren  $y_1, y_2, \dots, y_c$

## 1.1 Algorithmus

### Eingabe:

$n$  - Anzahl der Dimensionen

$m$  - Anzahl der gegebenen Muster

$X = \{x_i\}, 1 \leq i \leq m$  - die  $m$  Muster in  $R^n$

$Y = \{y_i\}, Z = \{z_i\}, 1 \leq i \leq c$  - zwei identische Sequenzen die die Initialclusterzentren enthalten

$k$  - die gewünschte Anzahl der Cluster

$m_0$  - minimal erlaubte Größe eines Clusters

$\sigma_0$  - Standardabweichungsschwelle (Teilung)

$\lambda$  - Teilungsbruch:  $0 < \lambda \leq 1$

$d_0$  - Vereinigungsschwelle

$l$  - maximale Anzahl von Clusterpaaren, die simultan vereinigt werden können

$\varepsilon$  - Toleranz

$N$  - maximale Iterationsschritte

$S, L$  - Vektoren der Größe  $N$ .

- Initial:  $S(i) = L(i) = 2, 1 \leq i \leq N$

- Nach der  $i$ -ten Iteration wird  $S(i) = 0$  oder  $L(i) = 0$  gesetzt, wenn das Teilen oder Vereinigen beginnt.
- Wenn die Teilung oder die Vereinigung erfolgreich abgeschlossen wurde, wird  $S(i) = 1$  oder  $L(i) = 1$  gesetzt

$NC$  - zeigt einen Wechsel in der Anzahl der Clusterzentren während der Klassifikation: Schritt 2 – Schritt 4

**Ausgabe:**

$Y = \{y_j\}, \quad 1 \leq j \leq c$  - die endgültigen Clusterzentren

$it$  - Anzahl der Iterationsschritte

**Schritt 1:**

Setze  $it = 0$  und  $S(i) = L(i) = 2, \quad 1 \leq i \leq N$

**Schritt 2:**

Setze  $c' = c$  und  $z_j = y_j, \quad 1 \leq j \leq c$  and  $NC = 1$

Unter Nutzung der existierenden Clusterzentren und der minimalen Distanz werden die Muster klassifiziert:

$$x \in C_j \text{ iff } \|x - y_j\| \leq \|x - y_i\|, \quad 1 \leq i \leq c, \quad i \neq j$$

für alle  $x \in X$ ,

**Schritt 3:**

Jedes Clusterzentrum mit weniger als  $m_0$  Mustern wird gelöscht. Seine Muster werden unter den übrigen Clustern verteilt. Setze  $c = c - 1$  falls ein Cluster gelöscht werden muss.

**Schritt 4:**

Aktualisiere die existierenden Clusterzentren durch:

$$y_j = \frac{1}{m_j} \sum_{i=1}^{m_j} x_{lij}, \quad 1 \leq j \leq c$$

Wenn  $c = c'$  und  $\sum_{i=1}^c \|y_j - z_j\| < \varepsilon$  dann setze  $NC = 0$

**Schritt 5:**

Für  $1 \leq j \leq c$  berechne den Durchschnittlichen Abstand von  $x_{lij}$ ,  $1 \leq i \leq m_j$  aus  $y_j$ :

$$\bar{d}_j = \frac{1}{m_j} \sum_{i=1}^{m_j} \|x_{lij} - y_j\|$$

**Schritt 6:**

Berechne die globale Durchschnittliche Entfernung  $\bar{d}$  von allen  $m$  Mustern zu deren zugehörigen Clusterzentren:

$$\bar{d} = \frac{1}{m_j} \sum_{j=1}^c m_j \bar{d}_j$$

Das ist das Ende einer Iteration. Setze  $it = it + 1$

**Schritt 7:**

Wenn  $it = N$ , dann gehe zu Schritt 13. Anderenfalls:

- (a) wenn  $c \leq \left\lceil \frac{k+1}{2} \right\rceil$ , dann gehe zu Schritt 8 (Teilung)
- (b) wenn  $\left\lceil \frac{k+1}{2} \right\rceil < c < 2k$  und  $it$  ungerade ist, dann gehe zu Schritt 8 (Teilung)
- (c) wenn  $c \geq 2k$ , dann gehe zu Schritt 10 (Vereinigung)
- (d) wenn  $\left\lceil \frac{k+1}{2} \right\rceil < c < 2k$  und  $it$  ist gerade, dann gehe zu Schritt 10 (Vereinigung)

**Schritt 8:**

Teilungsversuch: Setze  $S(it) = 0$ . Bezeichne für jedes Cluster das Clusterzentrum und die Clustermuster durch:

$$y_j = (y_j^{(1)}, y_j^{(2)}, \dots, y_j^{(n)})^T, \quad 1 \leq j \leq c$$

$$x_{ljk} = (x_{lkj}^{(1)}, x_{lkj}^{(2)}, \dots, x_{lkj}^{(n)})^T, \quad 1 \leq j \leq c, \quad 1 \leq k \leq m_j$$

Berechne den Standardabweichungsvektor:

$$\sigma_j = (\sigma_j^{(1)}, \sigma_j^{(2)}, \dots, \sigma_j^{(n)})^T, \quad 1 \leq j \leq c$$

wobei,

$$\sigma_j^{(i)} = \left( \frac{\sum_{k=1}^{m_j} (x_{lkj}^{(i)} - y_j^{(i)})^2}{m_j} \right)^{\frac{1}{2}}, \quad 1 \leq j \leq c, \quad 1 \leq i \leq n$$

Jedes  $\sigma_j^{(i)}$  ist die Standardabweichung des j-ten Clusters entlang der i-ten Koordinate.

Bezeichne  $\sigma_j^{(i0)} = \max \sigma_j^{(i)}, \quad 1 \leq i \leq n$

### Schritt 9:

Für  $1 \leq j \leq c$ , wenn  $\sigma_j^{(i0)} \leq \sigma_0$ , dann teile nicht das j-te Cluster, ansonsten teile es, vorausgesetzt das mindestens eine der folgenden Bedingungen erfüllt wird:

$$(a) \quad c \leq \left\lceil \frac{k+1}{2} \right\rceil$$

$$(b) \quad \bar{d}_j > \bar{d} \quad \text{und} \quad m_j \geq 2m_0$$

Die Teilung des j-ten Cluster geschieht wie folgt. Das Clusterzentrum  $y_j$  wird gelöscht, da 2 neue Clusterzentren  $y_{j+}, y_{j-}$  entstehen:

$$y_{j+} = (y_j^{(1)}, \dots, y_j^{(i0-1)}, y_j^{(i0)} + \lambda \sigma_j^{(i0)}, y_j^{(i0+1)}, \dots, y_j^{(n)})$$

$$y_{j-} = (y_j^{(1)}, \dots, y_j^{(i0-1)}, y_j^{(i0)} - \lambda \sigma_j^{(i0)}, y_j^{(i0+1)}, \dots, y_j^{(n)})$$

Setze  $c = c + 1$ . Auf diese Weise wurde  $y_i$  entlang der i-ten Koordinate geteilt. Die Teilung wird durch den Parameter  $\lambda$  kontrolliert, welcher einen erkennbaren aber nicht dramatischen Wechsel in der Clusterzentren Anordnung sicherstellt. Wenn die Teilung erfolgt ist setze  $S(it) = 1$  und gehe zu Schritt 2. Anderenfalls:

(a) wenn  $it > 1, L(it-1) = 0$  und  $NC = 0$ , dann gehe zu Schritt 12

(b) wenn  $it > 1, L(it-1) = 0$  und  $NC = 1$ , dann gehe zu Schritt 2

(c) wenn  $it > 1, L(it-1) \neq 0$ , dann mache einfach weiter

(d) wenn  $it = 1$ , dann mache einfach weiter

**Schritt 10:**

Vereinigung. Setze  $L(it) = 0$ .

- (a) wenn  $c < 2$ ,  $S(it) = 0$  und  $NC = 0$ , dann gehe zu Schritt 12
- (b) wenn  $c < 2$ ,  $S(it) = 0$  und  $NC = 1$ , dann gehe zu Schritt 2
- (c) wenn  $c < 2$ ,  $S(it) = 2$ , dann gehe zu Schritt 2

Anderenfalls berechne alle Distanzen zwischen 2 willkürlichen Clusterzentren.

$$d_{ij} = \|y_i - y_j\|, \quad 1 \leq i \leq c-1, \quad i+1 \leq j \leq c$$

Ordne  $\{d_{ij}\}$  monoton ansteigend neu und bezeichne durch  $l'$  die Anzahl der  $d_{ij}$ 's welche  $d_0$  nicht übersteigen.  $l^* = \min(l, l')$

$$d_{i_1 j_1} \leq d_{i_2 j_2} \leq \dots \leq d_{i^* j^*} \leq d_0$$

Wenn  $l^* = 0$  kommt keine Vereinigung zustande.

- (a) wenn  $S(it) = 2$ , dann gehe zu Schritt 2
- (b) wenn  $S(it) = 0$ , dann gehe zu Schritt 12
- (c) wenn  $l^* \neq 0$ , dann setze  $L(it) = 1$  und mache einfach weiter

**Schritt 11:**

Die Vereinigung beginnt mit dem Clusterpaar  $(i_1, j_1)$  und endet mit dem

Clusterpaar  $(i^*, j^*)$ . Jedes der beiden Clusterzentren wird vereinigt. Wenn ein gegebenes

Paar  $(i_r, j_r)$  so beschaffen ist, dass entweder das  $i_r$ -te oder das  $j_r$ -te Clusterzentrum schon

vereinigt wurde, wird diese Paar ignoriert. Die Vereinigung wird abgeschlossen durch die

Ersetzung der  $i_r$ -ten und  $j_r$ -ten Clusterzentrum durch deren Schwerpunktzentrum basierend auf deren Population:

$$y_{(i_r, j_r)} = \frac{m_{i_r} y_{i_r} + m_{j_r} y_{j_r}}{m_{i_r} + m_{j_r}}$$

Wenn  $y_{i_r}$  und  $y_{j_r}$  gelöscht wurden, setzt man  $c = c - 1$ . Wenn die Vereinigung fertig ist geht man weiter zu Schritt 2.

**Schritt 12:**

Gebe  $\{y_j\}$ ,  $1 \leq j \leq c$  und  $it$  aus und beende den Algorithmus.

### Schritt 13:

Gebe  $y_j$ ,  $1 \leq j \leq c$  und „Anzahl der Iterationen überschritten“ aus und beende den Algorithmus.

### 1.2 Beispiel

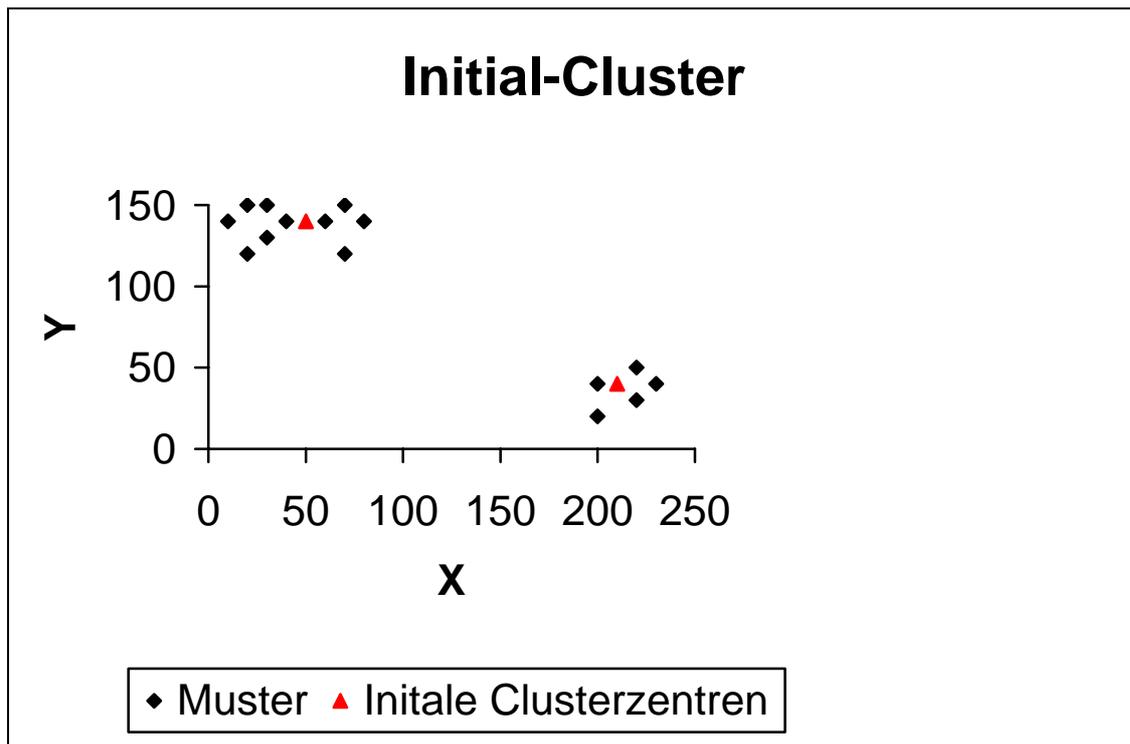


Abbildung 1: Beispiel Initial-Cluster

Gegebene Muster:

$$X = \begin{pmatrix} 10 \\ 140 \end{pmatrix}, \begin{pmatrix} 20 \\ 120 \end{pmatrix}, \begin{pmatrix} 20 \\ 150 \end{pmatrix}, \begin{pmatrix} 30 \\ 130 \end{pmatrix}, \begin{pmatrix} 30 \\ 150 \end{pmatrix}, \begin{pmatrix} 40 \\ 140 \end{pmatrix}, \begin{pmatrix} 60 \\ 140 \end{pmatrix}, \begin{pmatrix} 70 \\ 120 \end{pmatrix}, \\ \begin{pmatrix} 70 \\ 150 \end{pmatrix}, \begin{pmatrix} 80 \\ 140 \end{pmatrix}, \begin{pmatrix} 200 \\ 20 \end{pmatrix}, \begin{pmatrix} 200 \\ 40 \end{pmatrix}, \begin{pmatrix} 220 \\ 30 \end{pmatrix}, \begin{pmatrix} 220 \\ 50 \end{pmatrix}, \begin{pmatrix} 230 \\ 40 \end{pmatrix}$$

Initiale Clusterzentren:

$$Y = \begin{pmatrix} 50 \\ 140 \end{pmatrix}, \begin{pmatrix} 210 \\ 40 \end{pmatrix}$$

weitere Eingabeparameter:

$$k = 3, \quad m_0 = 2, \quad \sigma_0 = 20, \quad \lambda = 0.5, \quad d_0 = 40, \quad l = 2, \quad N = 10, \quad \varepsilon = 40$$

$$n = 2, \quad m = 15. \quad c = 2$$

**Schritt 1:**

$$it = 1, \quad S(i) = L(i) = 2, \quad 1 \leq i \leq 10$$

**Schritt 2:**

$$c' = c, \quad z_j = y_j, \quad 1 \leq j \leq 2, \quad NC = 1$$

Punkte zu Clustern zuordnen durch Berechnung der minimalen Distanz

$Y_1 (50,140)$

$$X_1 = (10,140)$$

$$X_2 = (20,120)$$

$$X_3 = (20,150)$$

$$X_4 = (30,130)$$

$$X_5 = (30,150)$$

$$X_6 = (40,140)$$

$$X_7 = (60,140)$$

$$X_8 = (70,120)$$

$$X_9 = (70,150)$$

$$X_{10} = (80,140)$$

$Y_2 (210,40)$

$$X_{11} = (200,20)$$

$$X_{12} = (200,40)$$

$$X_{13} = (220,30)$$

$$X_{14} = (220,50)$$

$$X_{15} = (230,40)$$

**Schritt 3:**

alle Cluster mehr als  $m_0 = 2$  Muster  $\rightarrow$  keine Cluster wird gelöscht

**Schritt 4:**

Für  $1 \leq j \leq 2$  alle Clusterzentren neu berechnen (Mittelwert)

$$Y_1 = \begin{pmatrix} 43 \\ 138 \end{pmatrix}, \quad Y_2 = \begin{pmatrix} 214 \\ 36 \end{pmatrix}$$

$$c = c' \text{ und } \sum_{i=1}^2 \|y_j - z_j\| < 40, \text{ also wird } NC = 0 \text{ gesetzt}$$

**Schritt 5:**

Für  $1 \leq j \leq 2$  die Durchschnittsentfernung der Muster zu dem zugehörigen Cluster berechnen.

$$Y_1 = \begin{pmatrix} 43 \\ 138 \end{pmatrix}, \quad X_1 = \begin{pmatrix} 10 \\ 140 \end{pmatrix} \Rightarrow \text{Entfernung} = \sqrt{(43-10)^2 + (138-140)^2} = 33,1$$

$$X_2 = 29,2$$

$$X_3 = 25,9$$

$$X_4 = 15,3$$

$$X_5 = 17,7$$

$$X_6 = 3,6$$

$$X_7 = 17,1$$

$$X_8 = 32,4$$

$$X_9 = 29,5$$

$$\underline{X_{10} = 37,1}$$

$$\underline{\underline{\sum = 240,9 / 10 \Rightarrow d_1 = 24,1}}$$

$$\underline{\underline{d_2 = 15,2}}$$

### Schritt 6:

Die globale Durchschnittsdistanz berechnen

$$\bar{d} = \frac{10 * 24,1 + 5 * 15,2}{15} = \underline{\underline{21,1}}$$

$$it = 1$$

### Schritt 7:

$$c = 2, \quad \frac{k+1}{2} = 2 \Rightarrow c \leq \frac{k+1}{2} \Rightarrow \text{weiter mit Schritt 8}$$

### Schritt 8:

$$S(1) = 0$$

Standartabweichungsvektoren für die Cluster berechnen:

Standartabweichung in X-Richtung des Clusters  $Y_1$ :

$$C_1 = \begin{pmatrix} 43 \\ 138 \end{pmatrix}$$

Quadrierte Entfernung (X-Richtung) zu Punkt  $X_1 = \begin{pmatrix} 10 \\ 140 \end{pmatrix} : = (43 - 10)^2 = 1089$

$$X_2 = 529$$

$$X_3 = 529$$

$$X_4 = 169$$

$$X_5 = 169$$

$$X_6 = 9$$

$$X_7 = 289$$

$$X_8 = 729$$

$$X_9 = 729$$

$$\underline{X_{10} = 1369}$$

$$\sum = \sqrt{5610 / 10} \Rightarrow \underline{\underline{23,7}}$$

Quadrierte Entfernung (Y-Richtung) zu Punkt  $X_1 = \begin{pmatrix} 10 \\ 140 \end{pmatrix} : = (4 - 10)^2 = 1089$

$$X_2 = 324$$

$$X_3 = 144$$

$$X_4 = 64$$

$$X_5 = 144$$

$$X_6 = 4$$

$$X_7 = 4$$

$$X_8 = 324$$

$$X_9 = 144$$

$$\underline{X_{10} = 4}$$

$$\sum = \sqrt{1160 / 10} \Rightarrow \underline{\underline{10,8}}$$

Daraus ergibt sich der Standardabweichungsvektor für  $Y_1$  :

$$\underline{\underline{\sigma_1 = \begin{pmatrix} 23,7 \\ 10,8 \end{pmatrix}}}$$

Standartabweichungsvektor für  $Y_2$ :

$$\underline{\underline{\sigma_2 = \begin{pmatrix} 12 \\ 10,2 \end{pmatrix}}}$$

Aus jedem Standartabweichungsvektor den maximalen Wert ermitteln

$\sigma_1^x = 23,7$  (X-Wert), d.h. das Cluster streut mehr in X-Richtung

$\sigma_2^x = 12$  (X-Wert), d.h. das Cluster streut mehr in X-Richtung

### Schritt 9:

$\sigma_1^x > \sigma_0 = 20 \Rightarrow$  Cluster  $Y_1$  teilen

$\sigma_2^x \leq \sigma_0 = 20 \Rightarrow$  Cluster  $Y_2$  nicht teilen

Prüfen ob  $C_1$  eine der Bedingungen  $c \leq \left\lceil \frac{k+1}{2} \right\rceil$  oder  $\bar{d}_1 > \bar{d}$  und  $m_1 \geq 2m_0$  erfüllt

$\Rightarrow$  es werden beide Bedingungen erfüllt, d.h. Cluster  $Y_1$  darf geteilt werden

Berechnung der beiden neuen Clusterzentren  $Y_{1+}$  und  $Y_{1-}$  aus  $Y_1$ :

$$Y_{1+} = \begin{pmatrix} 43 \\ 138 \end{pmatrix} + \begin{pmatrix} 0,5 * 23,7 \\ 0 \end{pmatrix} = \begin{pmatrix} 54,9 \\ 138 \end{pmatrix}$$

$$Y_{1-} = \begin{pmatrix} 43 \\ 138 \end{pmatrix} - \begin{pmatrix} 0,5 * 23,7 \\ 0 \end{pmatrix} = \begin{pmatrix} 31,2 \\ 138 \end{pmatrix}$$

Die Teilung ist nun beendet, man setzt  $c = 3$  und  $S(1) = 1$ , und geht weiter zu Schritt 2.

### Schritt 2:

$c' = c, \quad z_j = y_j, \quad 1 \leq j \leq 3, \quad NC = 1$

Punkte zu Clustern zuordnen durch Berechnung der minimalen Distanz

$Y_1 (54,9,138)$

$X_7 = (60,140)$

$X_8 = (70,120)$

$X_9 = (70,150)$

$X_{10} = (80,140)$

$Y_2 (31,2,138)$

$X_1 = (10,140)$

$X_2 = (20,120)$

$X_3 = (20,150)$

$X_4 = (30,130)$

$X_5 = (30,150)$

$X_6 = (40,140)$

$Y_3 (214,36)$

$X_{11} = (200,20)$

$X_{12} = (200,40)$

$X_{13} = (220,30)$

$X_{14} = (220,50)$

$X_{15} = (230,40)$

**Schritt 3:**

alle Cluster mehr als  $m_0 = 2$  Muster  $\rightarrow$  keine Cluster wird gelöscht

**Schritt 4:**

Für  $1 \leq j \leq 3$  alle Clusterzentren neu berechnen (Mittelwert)

$$Y_1 = \begin{pmatrix} 70 \\ 137,5 \end{pmatrix}, \quad Y_2 = \begin{pmatrix} 25 \\ 138,3 \end{pmatrix}, \quad Y_3 = \begin{pmatrix} 214 \\ 36 \end{pmatrix}$$

$c = c'$  und  $\sum_{i=1}^2 \|y_j - z_j\| < 40$ , also wird  $NC = 0$  gesetzt

**Schritt 5:**

$$\bar{d}_1 = 12,6, \quad \bar{d}_2 = 14,1, \quad \bar{d}_3 = 15,2,$$

**Schritt 6:**

$$\bar{d} = 14,1 \quad it = 2$$

**Schritt 7:**

$$c = 3, \quad \frac{k+1}{2} = 2 \Rightarrow \frac{k+1}{2} < c < 2k \Rightarrow it \text{ ist gerade, also weiter mit Schritt 10}$$

**Schritt 10:**

$$L(2) = 0$$

$$d_{1,2} = 45 \leq d_{1,3} = 176,4 \leq d_{2,3} = 214,8$$

alle Clusterdistanzen übersteigen die Vereinigungsschwelle  $d_0 = 40 \Rightarrow l' = 0$

$$l^* = \min(l, l') = \min(2, 0) = 0 \Rightarrow \text{keine Vereinigung findet statt}$$

$$S(2) = 2 \Rightarrow \text{weiter zu Schritt 2}$$

**Schritt 2:**

$$c' = c, \quad z_j = y_j, \quad 1 \leq j \leq 3, \quad NC = 1$$

Punkte zu Clustern zuordnen durch Berechnung der minimalen Distanz

<u><math>Y_1 (70, 137,5)</math></u>	<u><math>Y_2 (25, 138,3)</math></u>	<u><math>Y_3 (214,36)</math></u>
$X_7 = (60,140)$	$X_1 = (10,140)$	$X_{11} = (200,20)$
$X_8 = (70,120)$	$X_2 = (20,120)$	$X_{12} = (200,40)$
$X_9 = (70,150)$	$X_3 = (20,150)$	$X_{13} = (220,30)$
$X_{10} = (80,140)$	$X_4 = (30,130)$	$X_{14} = (220,50)$
	$X_5 = (30,150)$	$X_{15} = (230,40)$
	$X_6 = (40,140)$	

**Schritt 3:**

alle Cluster mehr als  $m_0 = 2$  Muster  $\rightarrow$  keine Cluster wird gelöscht

**Schritt 4:**

Für  $1 \leq j \leq 3$  alle Clusterzentren neu berechnen (Mittelwert)

$$Y_1 = \begin{pmatrix} 70 \\ 137,5 \end{pmatrix}, \quad Y_2 = \begin{pmatrix} 25 \\ 138,3 \end{pmatrix}, \quad Y_3 = \begin{pmatrix} 214 \\ 36 \end{pmatrix}$$

$$c = c' \text{ und } \sum_{i=1}^2 \|y_j - z_j\| < 40, \text{ also wird } NC = 0 \text{ gesetzt}$$

**Schritt 5:**

$$\bar{d}_1 = 12,6, \quad \bar{d}_2 = 14,1, \quad \bar{d}_3 = 15,2,$$

**Schritt 6:**

$$\bar{d} = 14,1 \quad it = 3$$

**Schritt 7:**

$$c = 3, \quad \frac{k+1}{2} = 2 \Rightarrow \frac{k+1}{2} < c < 2k \Rightarrow it \text{ ist ungerade, also weiter mit Schritt 8}$$

**Schritt 8:**

$$S(3) = 0$$

$$\underline{\underline{\sigma_1 = \begin{pmatrix} 7,1 \\ 10,9 \end{pmatrix}}}, \quad \underline{\underline{\sigma_2 = \begin{pmatrix} 9,6 \\ 10,7 \end{pmatrix}}}, \quad \underline{\underline{\sigma_3 = \begin{pmatrix} 12 \\ 10,2 \end{pmatrix}}}$$

Aus jedem Standardabweichungsvektor den maximalen Wert ermitteln:

$\sigma_1^y = 10,9$  (Y-Wert), d.h. das Cluster streut mehr in Y-Richtung

$\sigma_2^y = 10,7$  (Y-Wert), d.h. das Cluster streut mehr in Y-Richtung

$\sigma_3^x = 10,2$  (X-Wert), d.h. das Cluster streut mehr in X-Richtung

**Schritt 9:**

$$\sigma_1^y \leq \sigma_0 = 20 \Rightarrow \text{Cluster } Y_1 \text{ nicht teilen}$$

$$\sigma_2^y \leq \sigma_0 = 20 \Rightarrow \text{Cluster } Y_2 \text{ nicht teilen}$$

$$\sigma_3^x \leq \sigma_0 = 20 \Rightarrow \text{Cluster } Y_3 \text{ nicht teilen}$$

$$it > 1, \quad L(2) = 0, \quad NC = 0 \Rightarrow \text{weiter zu Schritt 12}$$

**Schritt 12:**

Ausgabe:

$$Y_1 = \begin{pmatrix} 70 \\ 137,5 \end{pmatrix}, \quad Y_2 = \begin{pmatrix} 25 \\ 138,3 \end{pmatrix}, \quad Y_3 = \begin{pmatrix} 214 \\ 36 \end{pmatrix}, \quad it = 2$$

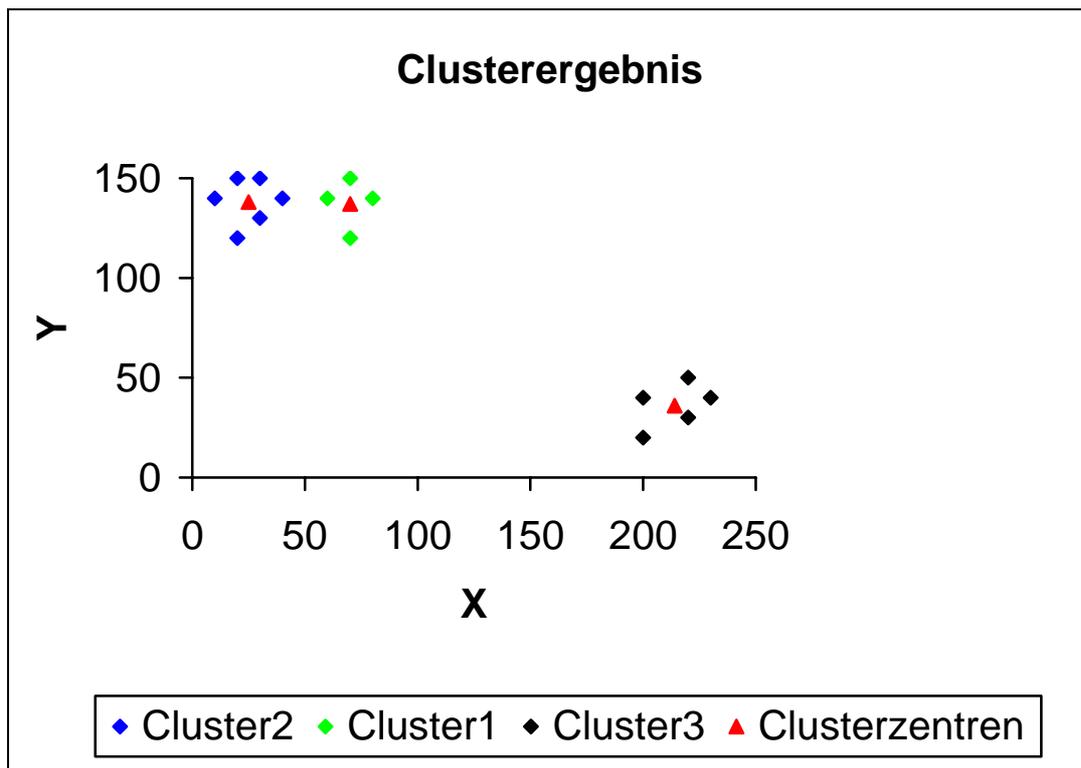


Abbildung 2: Beispiel Clusterergebnis

## 2. Bedienungsanleitung

### 2.1 Startbildschirm

Beim starten der Anwendung erscheint folgender Startbildschirm (Abb. 3).

Im linken Abschnitt ist das Formular (Koordinatensystem) der Muster zu sehen.

Im rechten Abschnitt werden Daten geladen, die Initialisierungsparameter angegeben und alle sonstigen Einstellungen vorgenommen. Durch betätigen des „Hilfe“ Buttons erscheint eine kurze Anleitung des Programms.

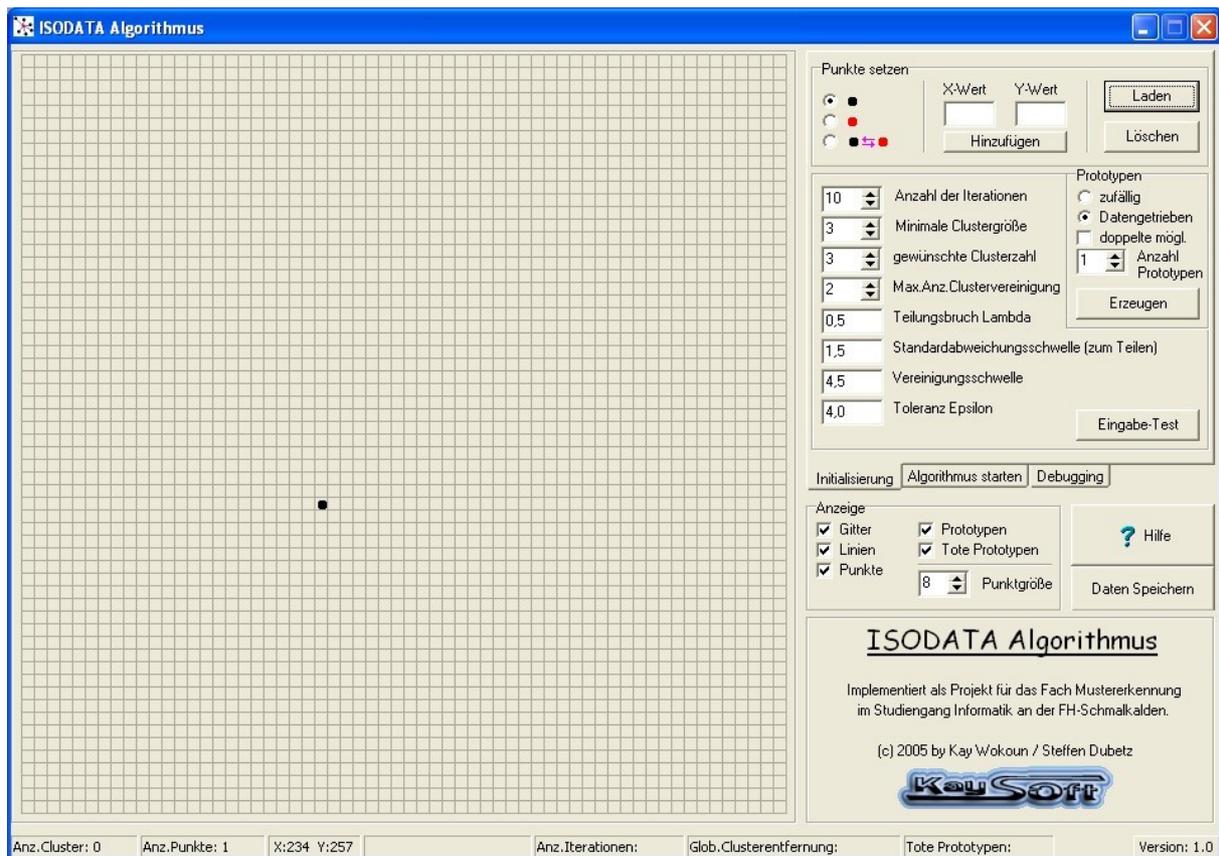


Abbildung 3: Startbildschirm

Im unteren Abschnitt befindet sich die Statusleiste in der man folgende Anzeigen finden kann:

Anz.Cluster: 3

Zeigt die aktuelle Anzahl der gebildeten Cluster

Anz.Punkte: 15

Zeigt die Anzahl der angelegten Muster

X:214 Y:36

Zeigt die aktuelle Position der Maus im Feldbereich, oder falls ein Eingabe- oder Prototypvektor ausgewählt wurde, dessen Position

Anz.Clusterpunkte: 5

Wenn ein Prototyp ausgewählt wird, wird hier die Anzahl der Muster angezeigt, die zu diesem die kürzeste Entfernung besitzen.

Anz.Iterationen: 3

Wenn der Algorithmus gestartet wurde, wird hier die aktuelle Anzahl der Iterationen angezeigt

Glob.Clusterentfernung:

Wenn der Algorithmus läuft wird hier die aktuelle globale Clusterentfernung angezeigt.

Tote Prototypen: 0

Zeigt die Anzahl der Toten Prototypen an, die z.B. durch Clustervereinigung entstanden sind.

## 2.2 Muster manuell anlegen

Durch die Auswahl im Optionsfeld „Punkte setzen“ kann man die Muster bzw. Prototypen manuell im Formular setzen. Um einzelne Muster manuell zu setzen, wählt man das Optionsfeld neben dem „schwarzen Punkt“ (Abb. 4) und klickt anschließend auf das Formular um ein neues Muster zu setzen. In der unteren Statusleiste sieht man dabei die aktuelle X- und Y-Koordinate des Mauszeigers. Diesen Vorgang kann man wiederholen bis die gewünschten Muster erzeugt wurden sind.

Anschließend kann man durch den Wechsel im Optionsfeld auf den „roten Punkt“ (Abb. 5) einzelne Prototypen manuell im Koordinatensystem setzen. Die Prototypen werden durch „rote Punkte“ dargestellt.

Durch Optionsfeld 3 (Abb. 6) kann man ein angelegtes Muster in einen Prototype wandeln. Dafür einfach auf das Muster klicken welches in einen Prototype gewandelt werden soll



Abbildung 4: Muster setzen

### Linke Maustaste:

Ein Muster hinzufügen, oder einen ausgewählten (Muster färbt sich gelb) verschieben (dabei muss die Maustaste dauerhaft gedrückt werden).

### Rechte Maustaste:

Es wird ebenfalls ein Muster hinzugefügt, aber wenn die Maus dabei nicht bewegt wird, dann wird dieser nicht ausgewählt (gelb markiert) und durch erneutes drücken der Maustaste wird einen weiterer Eingabevektor auf dieselbe Position hinzugefügt. Ein ausgewählter Eingabevektor kann durch anklicken gelöscht werden.



**Abbildung 5: Prototypen setzen**

Linke Maustaste:

Einen Prototyp hinzufügen, oder einen ausgewählten (Prototyp färbt sich gelb) verschieben (dabei muss die Maustaste dauerhaft gedrückt werden).

Rechte Maustaste:

Es wird ebenfalls ein Prototyp hinzugefügt, aber wenn die Maus dabei nicht bewegt wird, dann wird dieser nicht ausgewählt (gelb markiert) und durch erneutes drücken der Maustaste wird einen weiterer Prototyp auf dieselbe Position hinzugefügt. Ein ausgewählter Prototyp kann durch anklicken gelöscht werden.

Info:

Es werden immer die kürzesten Entfernungen zwischen den Mustern zu allen Prototypen berechnet und die kürzesten Verbindungen eingezeichnet. Alle so entstandenen Zugehörigkeiten von Eingabevektoren zu Prototypen nennt man Cluster.



**Abbildung 6: Muster wechseln**

Linke Maustaste:

Ein Muster hinzufügen, oder einen ausgewähltes Muster in einen Prototypvektor umwandeln. Dieser kann, nachdem er wieder ausgewählt wurde, verschoben werden (dabei muss die Maustaste dauerhaft gedrückt sein).

Rechte Maustaste:

Es wird ebenfalls ein Muster hinzugefügt, aber wenn die Maus dabei nicht bewegt wird, dann wird dieses nicht ausgewählt (gelb markiert) und durch erneutes drücken der Maustaste wird ein weiteres Muster auf dieselbe Position hinzugefügt. Ein ausgewähltes Muster wird durch anklicken gelöscht und ein ausgewählter Prototyp wird durch anklicken in ein Muster umgewandelt

Wenn das Anlegen der Muster mit dem Mauszeiger zu ungenau ist, kann man im Fenster (Abb. 7) auch die X- und Y-Koordinaten der jeweiligen Muster angeben, und mit einem Klick auf den Button „Hinzufügen“ oder durch drücken der Entertaste, zum Formular hinzufügen. Als X- und Y-Koordinate sind nur ganze Zahlen zwischen 0 und 600 erlaubt. Sollten hier andere Werte eingegeben werden, erscheint eine Fehlermeldung (Abb. 8)



Abbildung 7: Musterkoordinaten



Abbildung 8: Falsche Koordinateneingabe

## 2.3 Initialisierungsparameter

Im Feld „Initialisierungsparameter“ können sämtliche Eingabeparameter angegeben werden, die für den ISODATA Algorithmus notwendig sind (Abb. 9). Dabei können immer nur positive Werte verwendet werden. Durch einen Klick auf den Button „Eingabe-Test“ kann überprüft werden ob alle Parameter im gültigen Intervall liegen.

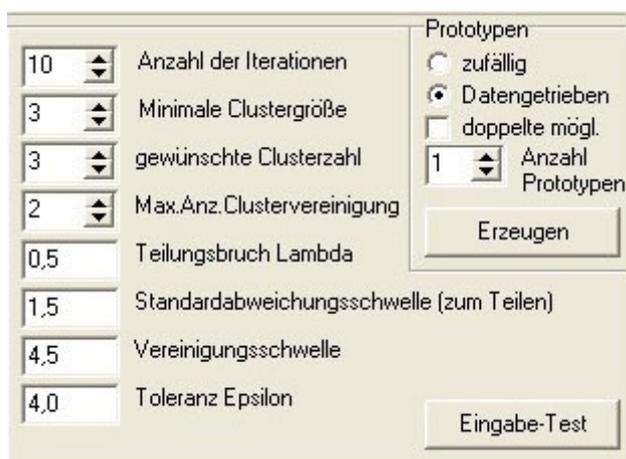


Abbildung 9: Initialisierungsparameter

Anzahl der Iterationen:

Legt die maximal Anzahl der Iterationen fest, nach dem der Algorithmus spätestens beendet wird.

Minimale Clustergröße:

Legt fest, wie viele Muster mindestens zu einem Prototyp gehören müssen, damit ein Cluster entsteht. Sind es weniger, wird der entsprechende Prototyp gelöscht und die Muster anderen Prototypen zugeordnet.

gewünschte Clusterzahl:

Hier stellen Sie die gewünschte Clusteranzahl ein, die möglichst am Ende des Algorithmus herauskommen soll. An diesem Wert orientiert sich der Algorithmus allerdings nur, es können am Ende durchaus mehr oder weniger Cluster entstanden sein.

Max. Anzahl Clustervereinigungen:

Legt fest, wie viele Prototypenpaare (Clusterpaare) bei einer Iteration gleichzeitig vereint werden können.

Teilungsbruch Lambda:

Dieser Wert besitzt einen Wertebereich der zwischen 0 und 1 liegt. Wird Lambda groß gewählt, werden bei der Clusterteilung die 2 neuen Cluster weiter auseinander, bei einem kleineren Lambda weiter zusammen angelegt.

Standardabweichungsschwelle (zum Teilen):

Legt fest, ab welcher Schwelle (Wert) ein Cluster geteilt werden soll. Der optimale Wert muss durch probieren ermittelt werden. Eine gute Hilfe stellt dabei der Debug-Modus dar, wo man genau den Wert ablesen kann, der zum Teilen der Cluster führen würde.

Vereinigungsschwelle:

Dieser Wert ist genau das Gegenstück zur vorherigen Einstellung und legt fest, wann 2 Cluster vereinigt werden. Auch hier ist zum ermitteln des optimalen Wertes der Debug-Modus eine große Hilfe.

Toleranz Epsilon:

Die Clusterzentren verschieben sich von Schritt zu Schritt des Algorithmus. Wenn diese Verschiebung aller Cluster unterhalb von Epsilon liegt, dann wird in Schritt 4 die Variable  $NC = 0$  gesetzt. Wenn  $NC$  in Schritt 9 = 0 ist, dann wird der Algorithmus abgebrochen und ist beendet.

Unter „Prototypen“ kann man die Anzahl der automatisch zu erzeugenden Prototypen angeben. Auch hier sind nur positive ganze Zahlen erlaubt. Wählt man das Optionsfeld „zufällig“, werden die Prototypen rein zufällig in dem Bereich der kleinsten/größten X/Y - Koordinate der Eingabevektoren gesetzt. Wählt man die Option „Datengetrieben“ werden die einzelnen Prototypen auf bereits vorhandene Muster gelegt.

Man kann außerdem durch die Option „doppelte möglich“ mehrere Prototypen für dieselbe Koordinate ermöglichen.

Durch einen Klick auf den Button „Erzeugen“ werden die Prototypen automatisch, mit den gewählten Optionen, im Formular erzeugt.

## 2.4 Visualisierung ändern

Im Feld „Anzeige“ kann man die Visualisierung des Formulars ändern und anpassen. (Abb. 10) Defaultmäßig sind hier alle Optionen gesetzt, d.h. es werden alle möglichen Visualisierungen im Formular angezeigt. (Abb. 11) Diese Variante verschlechtert allerdings die Performance des Algorithmus, da alle visuellen Objekte bei jedem Schritt neu gezeichnet werden müssen.



Abbildung 10: Anzeige (Defaulteinstellung)

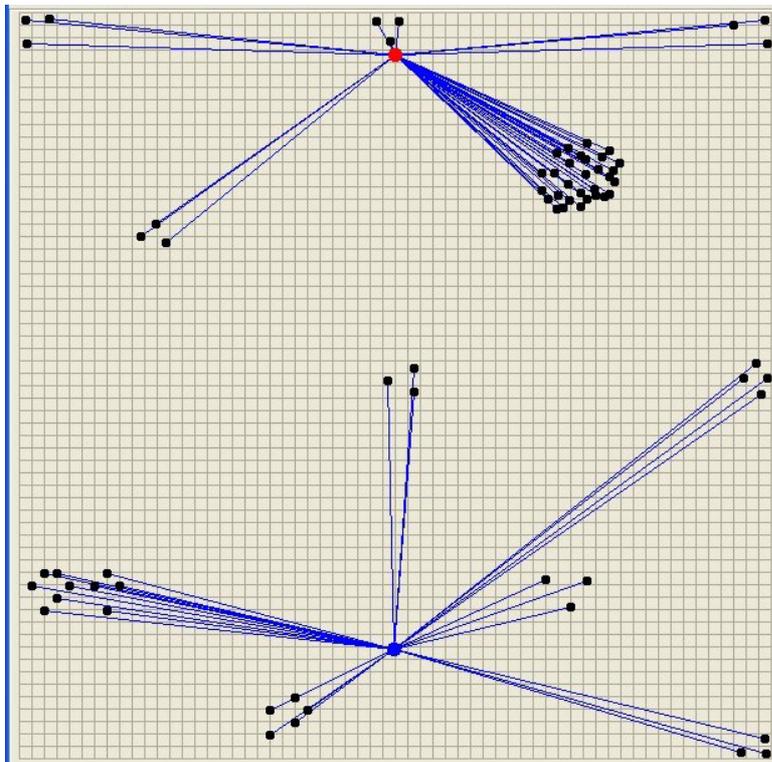


Abbildung 11: Formular (Defaulteinstellung)

Wenn man die Option „Punkte“ (Muster) deaktiviert (Abb. 12), wird das Formular ohne die gegebenen Muster dargestellt. Diese Variante bringt einen erheblichen Performancegewinn gegenüber der Darstellung aller Muster während der Ausführung des ISODATA Algorithmus.

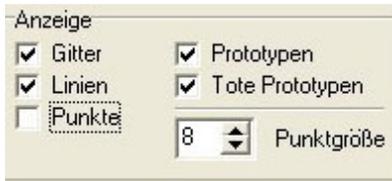


Abbildung 12: Anzeige (ohne Muster)

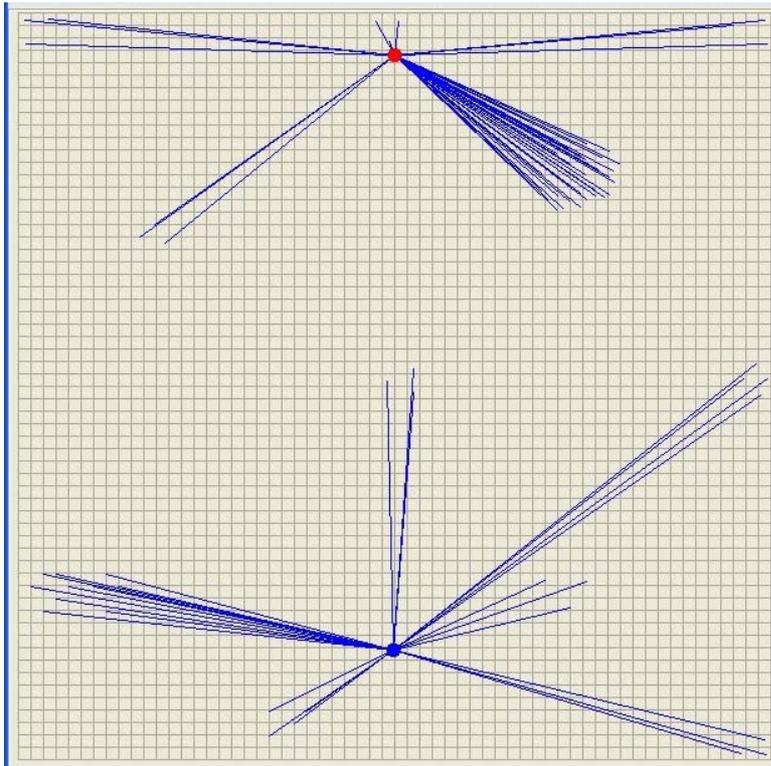


Abbildung 13: Formular (ohne Muster)

Mit der Option „Linien“ kann man die blauen Verbindungslinien zwischen Muster und den zugehörigen Prototypen ein- bzw. ausblenden. Die Linien verdeutlichen die Zugehörigkeit eines bestimmten Musters zu dem jeweiligen Prototypen (Cluster).

Sollte man die Linien ausblenden, färben sich alle Muster eines Cluster in derselben Farbe. Die Muster von verschiedenen Clustern werden durch unterschiedliche Farben dargestellt, z.B. Cluster 1 = rot und Cluster 2 = blau (Abb. 15).

So kann man auch ohne Verbindungslinien die Clusterzugehörigkeit eines Musters eindeutig erkennen.



Abbildung 14: Anzeige (ohne Linien)

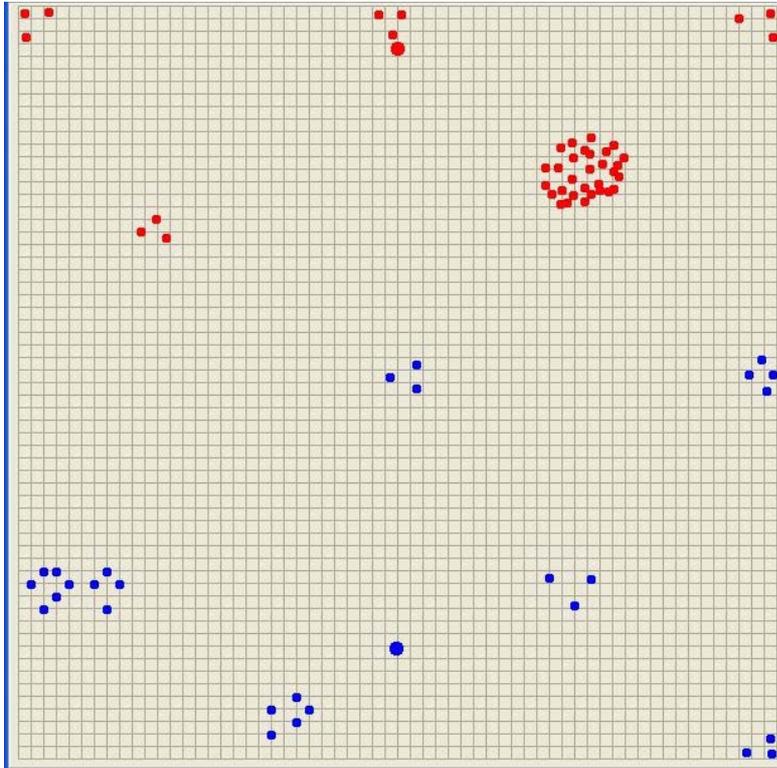


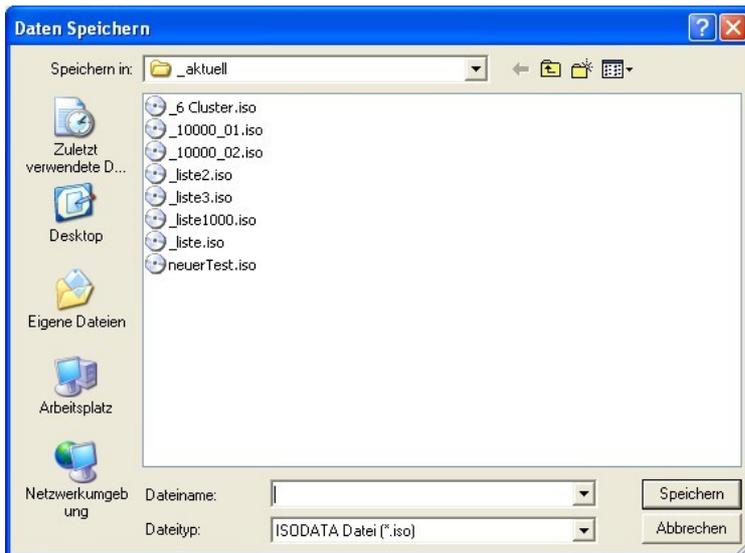
Abbildung 15: Formular (ohne Linien)

Des Weiteren kann man im Feld „Anzeige“ das Gitter, die Prototypen und die toten Prototypen aus- bzw. einblenden.

Tote Prototypen entstehen durch die Vereinigung zweier Cluster und werden durch die Farbe „grau“ dargestellt.

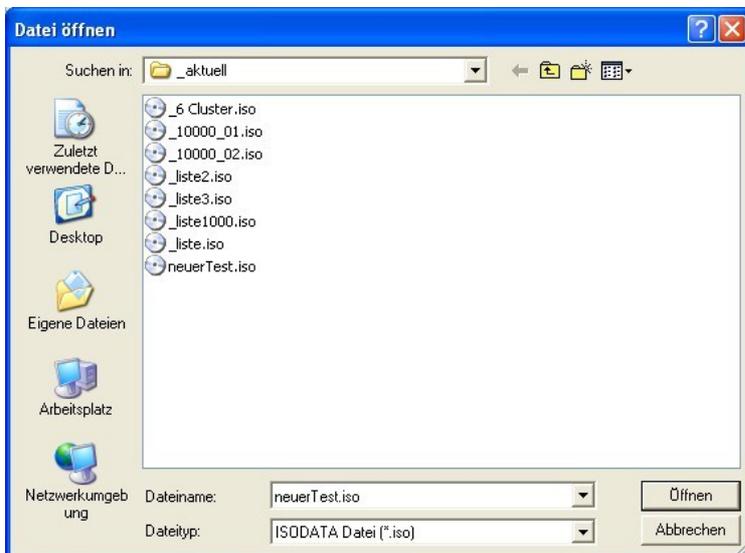
## 2.5 Daten laden und speichern

Durch einen Klick auf den Button „Speichern“, kann man alle angelegten Muster und Prototypen, sowie alle Initialisierungs- und Anzeigeparameter speichern. Dabei öffnet sich folgender Dialog (Abb. 16). Dabei kann man nur eine Datei mit der Endung \*.iso speichern.



**Abbildung 16: Daten speichern**

Durch einen Klick auf den Button „Laden“ öffnet sich folgender Dialog (Abb. 17). Man kann hier nicht nur Dateien mit der Endung \*.iso laden sondern auch jede andere Datei (Format: x-Koordinate y-Koordinate und dazwischen kann jedes Trennzeichen stehen). Durch das Laden werden alle evtl. vorher angelegten Muster, Prototypen sowie alle Initialisierungs- und Anzeigeparameter überschrieben.



**Abbildung 17: Daten laden**

## 2.6 Debugging

Wenn man den ISODATA Algorithmus Schritt für Schritt nachvollziehen möchte, und alle wichtigen Rechenergebnisse angezeigt bekommen möchte, der kann dies im „Debugging-Mode“ tun. Hierfür einfach auf den Tab „Debugging“ wechseln (Abb. 18)



Abbildung 18: Debugging

Durch die Optionsfelder (Schritt 1 - Schritt 13) kann man wählen welche Schritte des Algorithmus „debugged“ werden sollen. Dabei kann man durch einen Klick auf den Button „Aktivieren“ / „Deaktivieren“ alle Schritte gleichzeitig ein- bzw. ausschalten. Im rechten Fenster wird dazu nochmals der wesentliche Ablauf jedes Schrittes erläutert.

Wichtig ist, dass Sie einen Haken bei Debuggen setzen! Dies wird automatisch gemacht, wenn das Debugfenster durch klicken auf den „Show Debug“ - Button geöffnet wird.

Durch einen Klick auf den Button „Show Debug“ öffnet sich ein neues Fenster (Abb. 19) in dem die Debug-Informationen während der Ausführung des ISODATA Algorithmus angezeigt werden (Abb. 20).

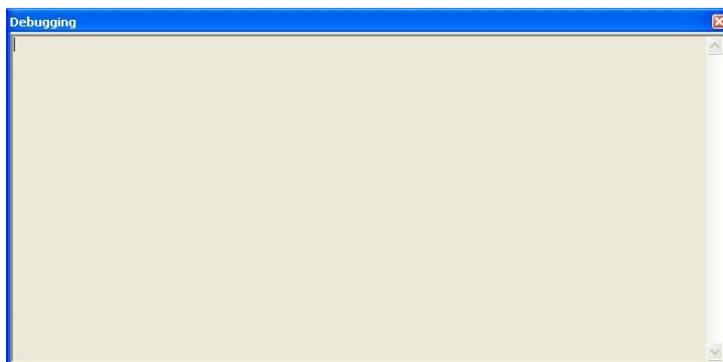
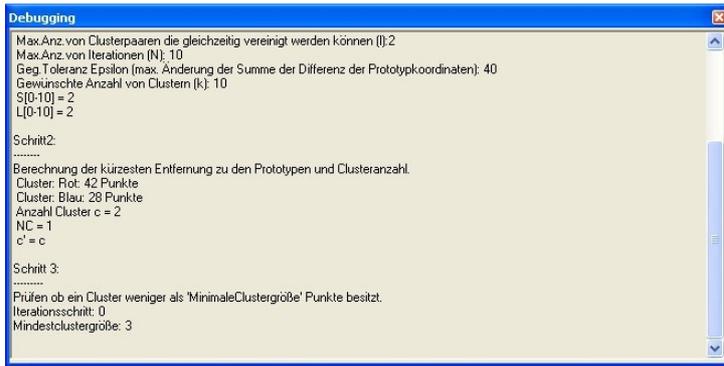


Abbildung 19: Debugfenster (leer)



**Abbildung 20: Debugfenster (gefüllt)**

Durch einen Klick auf den Button „Lösche Inhalt“ werden alle Debug-Informationen aus dem Fenster gelöscht.

Durch klicken des Buttons „Inhalt speichern“ wird der gesamte Inhalt des Debugfenster in eine txt-Datei gespeichert.

## 2.7 Algorithmus starten

Wenn man nach der Initialisierung den Algorithmus starten möchte, muss man auf den Tab „Algorithmus starten“ wechseln (Abb. 21)



**Abbildung 21: Algorithmus starten**

Die Optionsfelder (Schritt 1 – Schritt 13) zeigen an, bei welchem Schritt sich der Algorithmus gerade befindet.

Durch einen Klick auf den Button „Go“ wird der Algorithmus gestartet. Sollte hierbei die Option „Auto“ deaktiviert sein, wird nur 1 Schritt des Algorithmus pro Klick ausgeführt.

Wenn die Option „Auto“ aktiviert ist, läuft der komplette Algorithmus automatisch bis zum Ende durch. Die Dauer zwischen den einzelnen Arbeitsschritten kann dabei durch den Regler verändert werden. Je höher der Regler steht, desto schneller läuft der Algorithmus durch.

Im Feld „Reset Optionen“ stehen 4 Resetmöglichkeiten zur Auswahl.

Bei der Option „Prototypen löschen“ werden alle vorher angelegten oder geladenen Prototypen nach einem Klick auf den Button „Reset“ aus dem Formular gelöscht. Die Muster bleiben aber erhalten.

Die Option „Prototypen wie nach Laden“ stellt den Algorithmus auf die Konstellation zurück, wie sie nach dem Laden war. Man kann z.B. einen Datensatz laden und diesen dann verändern. Wenn man die Änderungen rückgängig machen möchte, braucht man hierzu nur auf den Button „Reset“ klicken, und braucht nicht den ganzen Datensatz erneut laden.

Durch die Option „Prototypen wie vor Algorithmus“ kann man nach Ausführung des ISODATA Algorithmus, die Muster und Prototypen in ihre Position, wie sie vor dem Starten des Algorithmus waren, bringen.

Die Option „nur Algorithmus zurücksetzen“ bringt nur den ISODATA Algorithmus wieder in die Initiale Position, von der aus er wieder ausgeführt werden kann.

Wenn man z.B. den Algorithmus mit 10 Iterationen durchführt und anschließend die Option „nur Algorithmus zurücksetzen“ wählt und den Algorithmus erneut mit 10 Iterationen durchführt, entspricht das Clusterergebnis der beiden Durchläufe, denselben, als wenn man den Algorithmus mit 20 Iterationen gestartet hätte.