Alexander Kowarik
Statistik Austria

Einführung in die statistische Geheimhaltung

Mikrodaten für Forscher

- Wir werden immer häufiger mit Datenanfragen von Forschern und Institutionen konfrontiert.
- Die Datenweitergabe ist kritisch wegen bestehender Gesetze über Datenschutz.

```
R> x[12876, c(3,21,6)]
          Wohnort  Geschlecht Beruf
12867       Vorau  m\"annlich Univ.-Prof.
```

- $\longrightarrow$ Notwendigkeit, Daten zu anonymisieren

- **direkte Identifizierungsvariablen** (z.B. *Sozialversicherungsnummer*, *Name*, usw.)
- **indirekte Identifizierungsvariablen**: alle kategorielle Variable aus̈ser direkten Identifizierungsvariablen
- **Schlüsselvariablen** für die Geheimhaltung:
  $\longrightarrow$ jene indirekte Identifizierungsvariablen über die *Datenangreifer* Informationen besitzen könnten.
- durch **Verkreuzung** entsteht das **Geheimhaltungsproblem** (siehe Beispiel mit *Univ.-Prof.*, *männlich*, *Vorau*, von vorhin).
- **Beispiele** für Schlüsselvariablen sind *Nationalität*, *Beruf*, *NACE*, *Forschungsausgaben von Unternehmen*, etc.

Grundbegriffe

- Konzept der **uniqueness**: Durch Kombination mehrerer Variablen kann ein Individuum eindeutig im Datensatz identifiziert werden ($f_k = 1$).
- Konzept der k-**Anonymität**: Jeder Ausprägungskombination können zumindest k Beobachtungen zugeordnet werden ($f_k \geq 3$).
- Konzept des **Re-Identifizierungsrisikos**: Suche rare Kombinationen in der Population (F_k) durch Berücksichtigung des Stichprobengewichtes. Schätzung des Riskios über Verteilungsannahmen bzgl. des Quotionten $\frac{f_k}{F_k}$.
- Eine Stichprobe an sich trägt schon zur Geheimhaltung bei: Der Datenangreifer kann sich nicht sicher sein, ob eine Person oder ein Unternehmen in der Stichprobe ist.

Beispieldaten

```
R> library(sdcMicro)
R> data(francdat)
R> x <- francdat[ ,c(1,3,7,2,4,5,6,8)]; x
```

	Num1	Num2	Num3	Key1	Key2	Key3	Key4	w
1	0.30	0.40	4	1	2	5	1	18.0
2	0.12	0.22	22	1	2	1	1	45.5
3	0.18	0.80	8	1	2	1	1	39.0
4	1.90	9.00	91	4	3	1	5	17.0
5	1.00	1.30	13	4	3	1	4	541.0
6	1.00	1.40	14	4	3	1	1	8.0
7	0.10	0.01	1	6	3	1	5	5.0
8	0.15	0.50	5	1	2	5	1	92.0

Re-Identifizierungsrisiko

```
R> f1 <- freqCalc(x, keyVars = 4:7, w = 8)
R> cbind(x[,4:8], f1$fk, f1$Fk, indivRisk(f1)$rk)
```

	Key1	Key2	Key3	Key4	w	f1$fk	f1$Fk	indivRisk(f1)$rk
1	1	2	5	1	18.0	2	110.0	0.01705402
2	1	2	1	1	45.5	2	84.5	0.02195396
3	1	2	1	1	39.0	2	84.5	0.02195396
4	4	3	1	5	17.0	1	17.0	0.17686217
5	4	3	1	4	541.0	1	541.0	0.01112028
6	4	3	1	1	8.0	1	8.0	0.29690573
7	6	3	1	5	5.0	1	5.0	0.40223299
8	1	2	5	1	92.0	2	110.0	0.01705402

Recoding von Kategorien

```
R> x[7,4] <- 4
R> f1 <- freqCalc(x, keyVars = 4:7, w = 8)
R> cbind(x[,4:8], f1$fk, f1$Fk, indivRisk(f1)$rk)
```

	Key1	Key2	Key3	Key4	w	f1$fk	f1$Fk	indivRisk(f1)$rk
1	1	2	5	1	18.0	2	110.0	0.01705402
2	1	2	1	1	45.5	2	84.5	0.02195396
3	1	2	1	1	39.0	2	84.5	0.02195396
4	4	3	1	5	17.0	2	22.0	0.07594707
5	4	3	1	4	541.0	1	541.0	0.01112028
6	4	3	1	1	8.0	1	8.0	0.29690573
7	4	3	1	5	5.0	2	22.0	0.07594707
8	1	2	5	1	92.0	2	110.0	0.01705402

Lokales Unterdrücken

```
R> x[5,7] <- NA
R> f1 <- freqCalc(x, keyVars = 4:7, w = 8)
R> cbind(x[,4:8], f1$fk, f1$Fk, indivRisk(f1)$rk)
```

	Key1	Key2	Key3	Key4	w	f1$fk	f1$Fk	indivRisk(f1)$rk
1	1	2	5	1	18.0	2	110.0	0.017054016
2	1	2	1	1	45.5	2	84.5	0.021953961
3	1	2	1	1	39.0	2	84.5	0.021953961
4	4	3	1	5	17.0	3	563.0	0.002607482
5	4	3	1	NA	541.0	4	571.0	0.002296465
6	4	3	1	1	8.0	2	549.0	0.003484248
7	4	3	1	5	5.0	3	563.0	0.002607482
8	1	2	5	1	92.0	2	110.0	0.017054016

Generelle Herangehensweise bei kat. Schlüsselvariablen

- Explorativ für das Recoding. Ziel: Minimales Recoding.
- Optimal bei Lokaler Unterdrückung (komplexe Optimierungsprobleme). Ziel: Nur in bestimmten Variablen so wenig wie möglich Werte Unterdrücken.
- Beispiele mit Echtdaten, siehe z.B. Meindl & Templ, 2007

Geheimhaltung bzgl. stetig skalierter Variablen

- Jede Beobachtung ist *unique*.
- Wenn ein Datenangreifer Information über einen Wert einer numerischen Variablen hat kann er die Beobachtung identifizieren und dadurch weit mehr Information über diese Person in Erfahren bringen.
- Datenangreifer können zur Verlinkung ihrer Information mit den Daten *Matching* Verfahren anwenden.
- $\rightarrow$ Auch numerische Variablen müssen geschützt werden.
- Die multivariate Struktur der Daten sollte erhalten bleiben und gleichzeitig das Re-Identifizierung minimiert werden.

Mikroaggregation

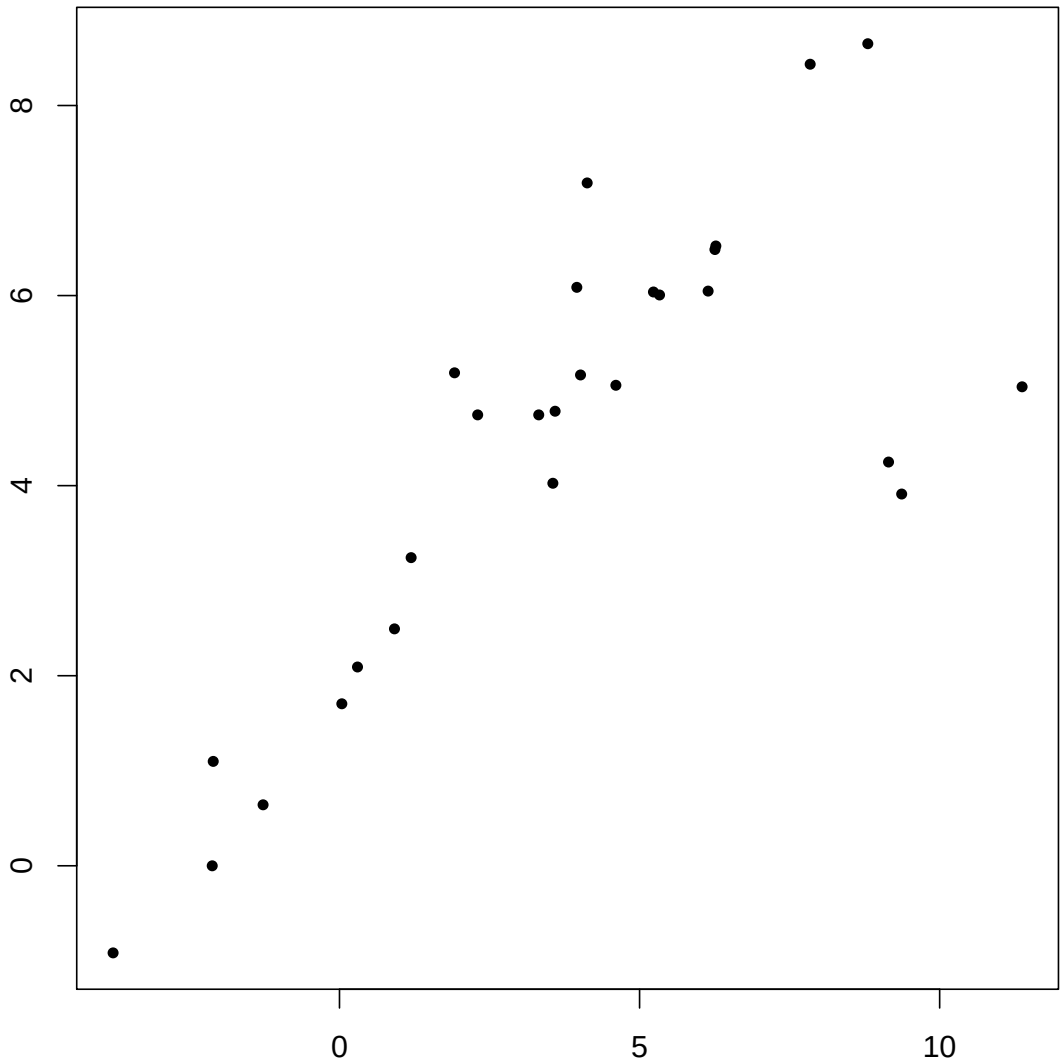
Suche ähnliche Beobachtungen, aggregiere diese mit dem arithm. Mittel, und ersetzte die Werte mit dieser Aggregation.

```
cbind(x[,1:3], microaggregation(x[,1:3], aggr=2,method="rmd"))
```

	Num1	Num2	Num3	Num1	Num2	Num3
1	0.30	0.40	4	0.200	0.205	2.5
2	0.12	0.22	22	1.010	4.610	56.5
3	0.18	0.80	8	0.165	0.650	6.5
4	1.90	9.00	91	1.010	4.610	56.5
5	1.00	1.30	13	1.000	1.350	13.5
6	1.00	1.40	14	1.000	1.350	13.5
7	0.10	0.01	1	0.200	0.205	2.5
8	0.15	0.50	5	0.165	0.650	6.5

Algorithm 1: simple

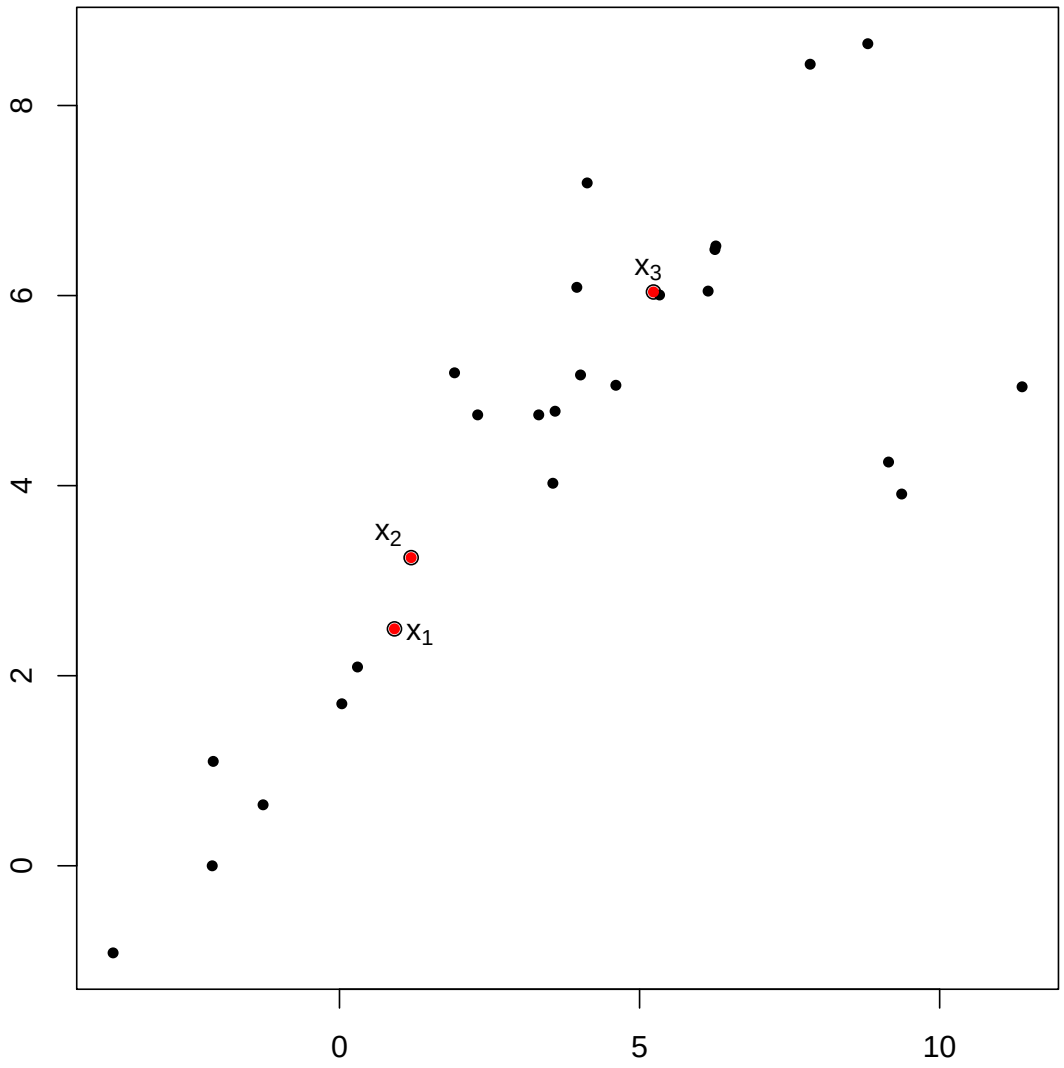
Zufallsauswahl



Algorithm 1: simple

Zufallsauswahl

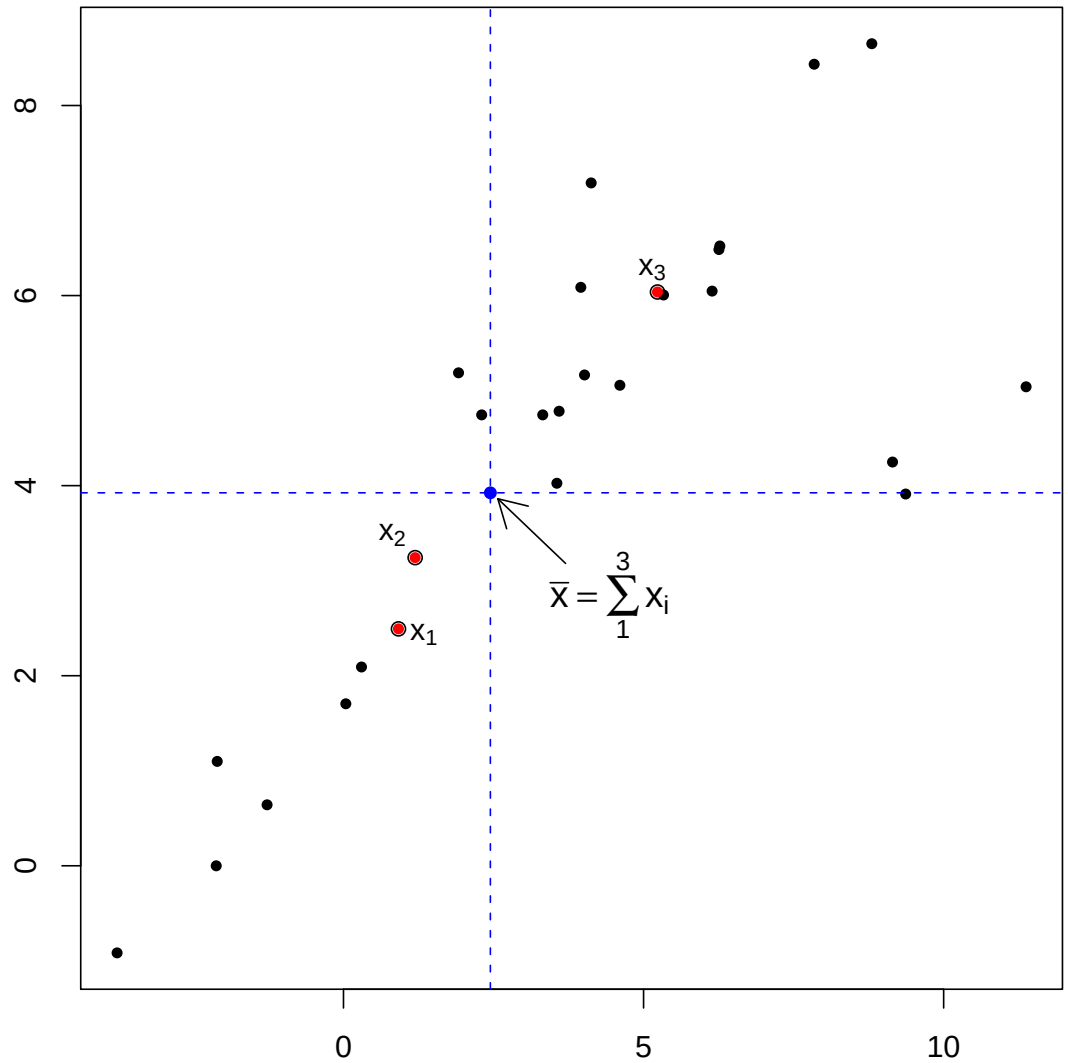
- wähle (z.B) 3 Zufallspunkte



Algorithm 1: simple

Zufallsauswahl

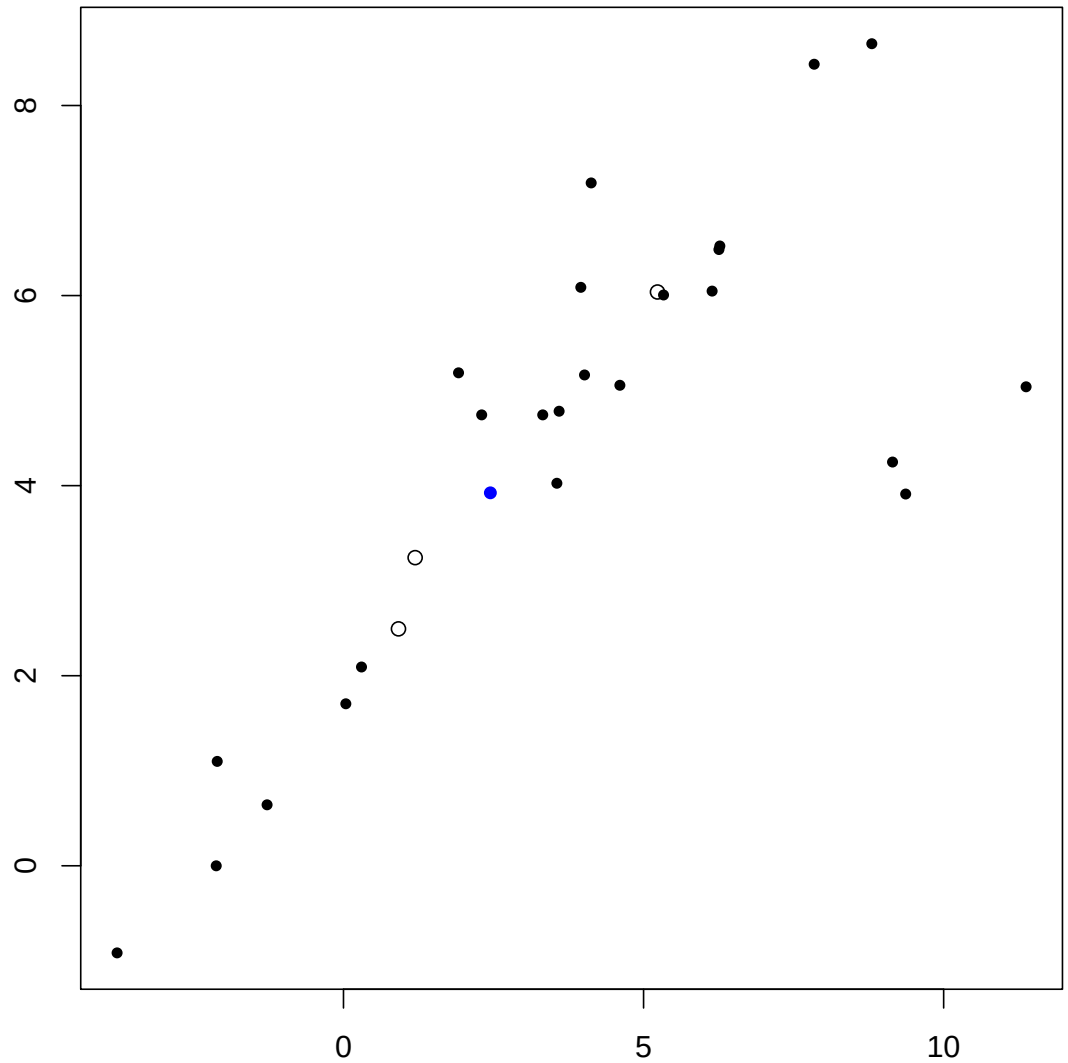
- wähle (z.B) 3 Zufallspunkte
- berechne Lagemaß (z.B Mittelwert)



Algorithm 1: simple

Zufallsauswahl

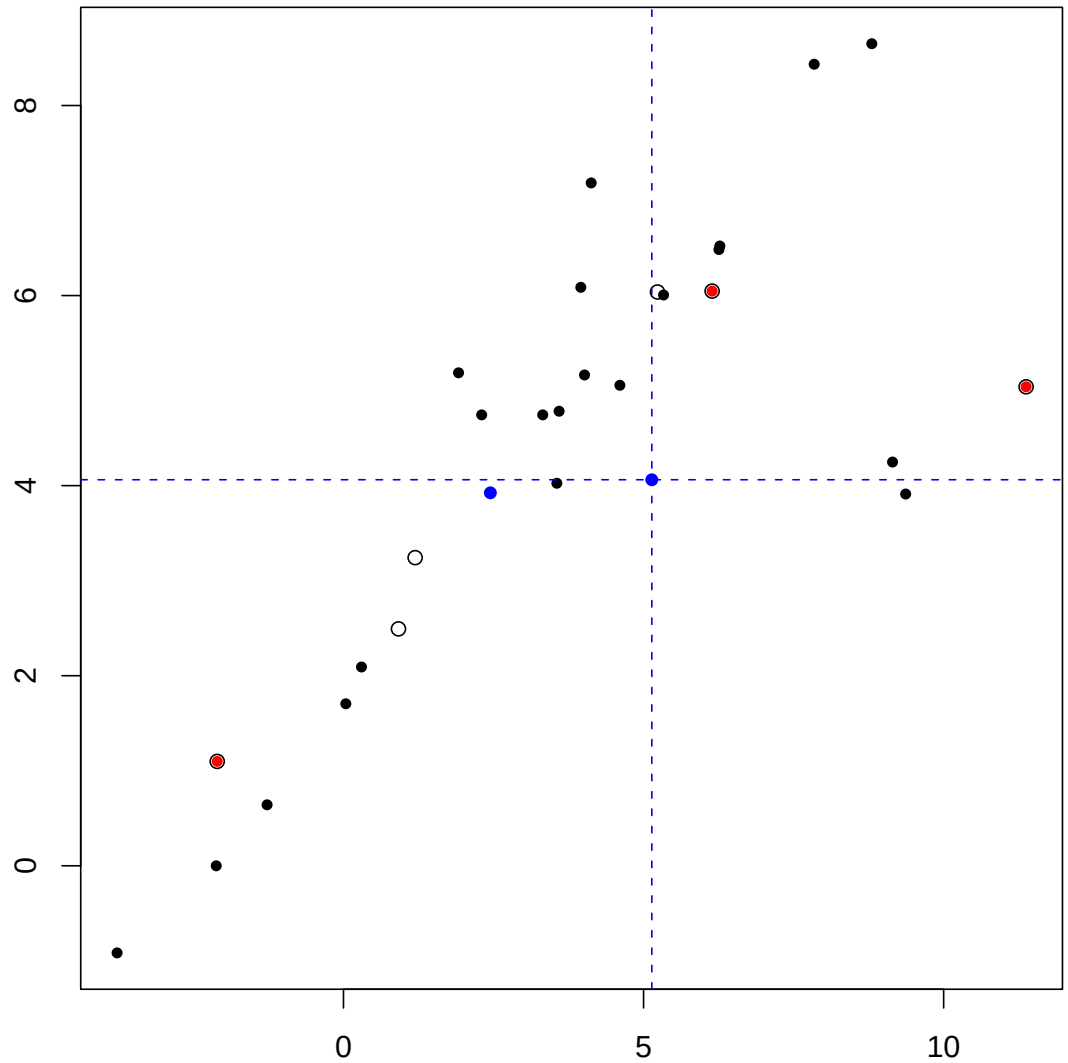
- wähle (z.B) 3 Zufallspunkte
- berechne Lagemaß (z.B Mittelwert)
- ersetze



Algorithm 1: simple

Zufallsauswahl

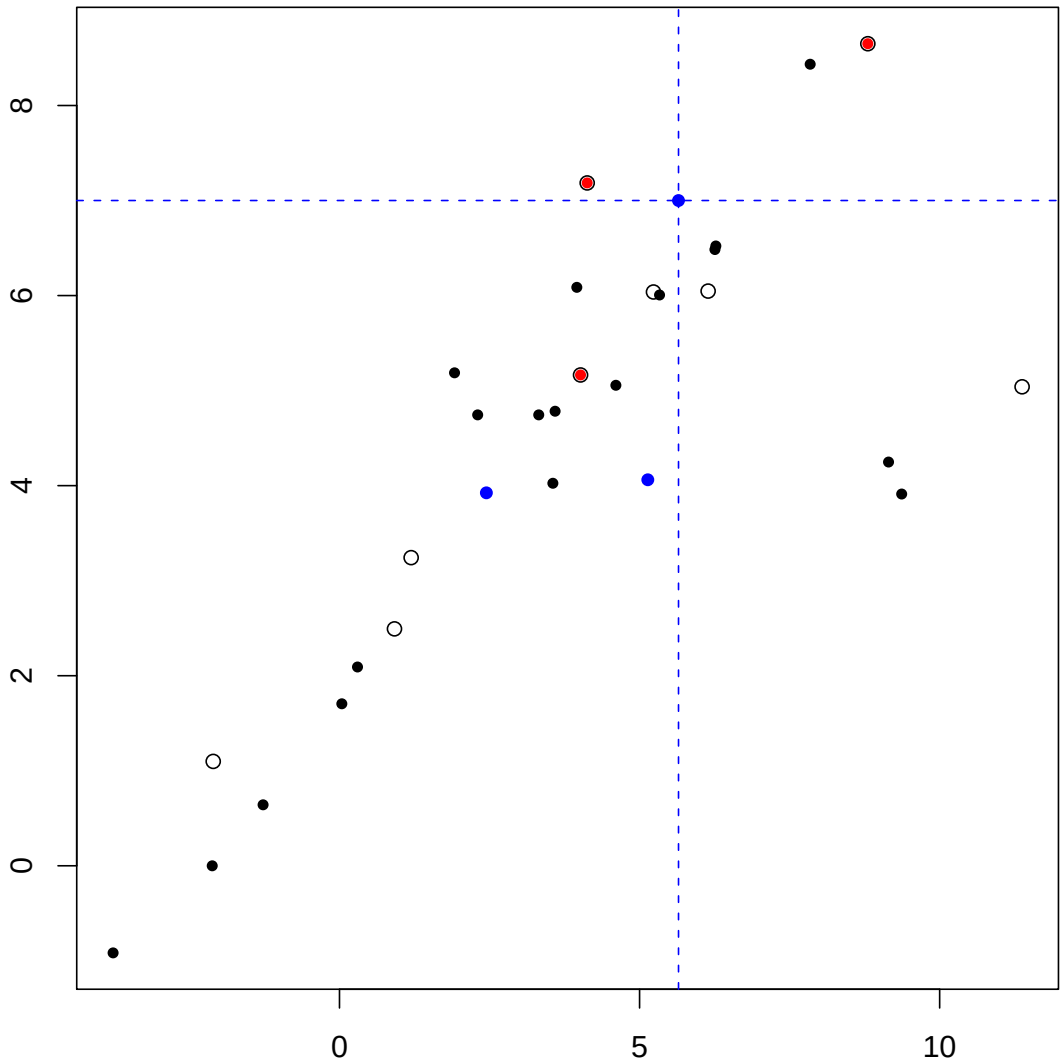
- wähle (z.B) 3 Zufallspunkte
- berechne Lagemaß (z.B Mittelwert)
- ersetze
- bis alle Punkte mikroaggregiert sind



Algorithm 1: simple

Zufallsauswahl

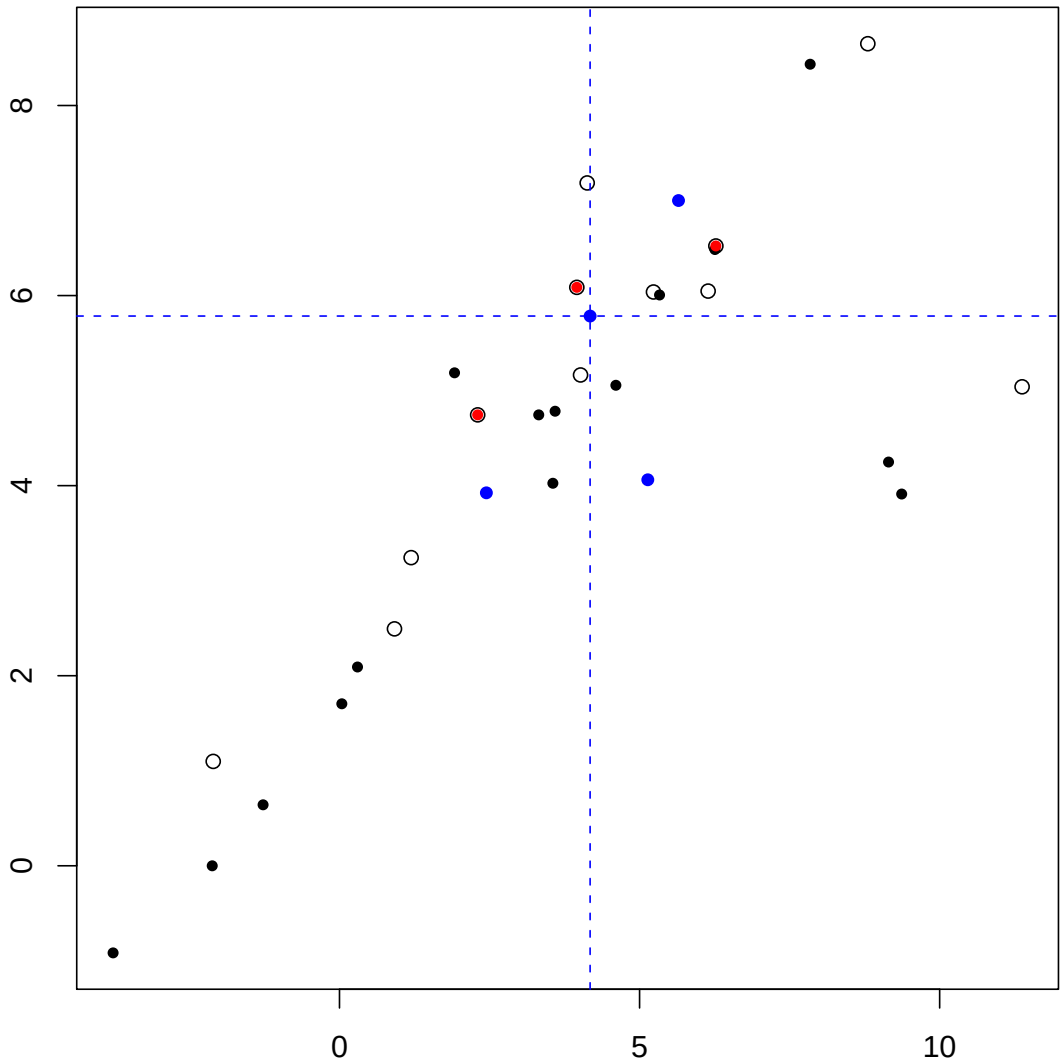
- wähle (z.B) 3 Zufallspunkte
- berechne Lagemaß (z.B Mittelwert)
- ersetze
- bis alle Punkte mikroaggregiert sind



Algorithm 1: simple

Zufallsauswahl

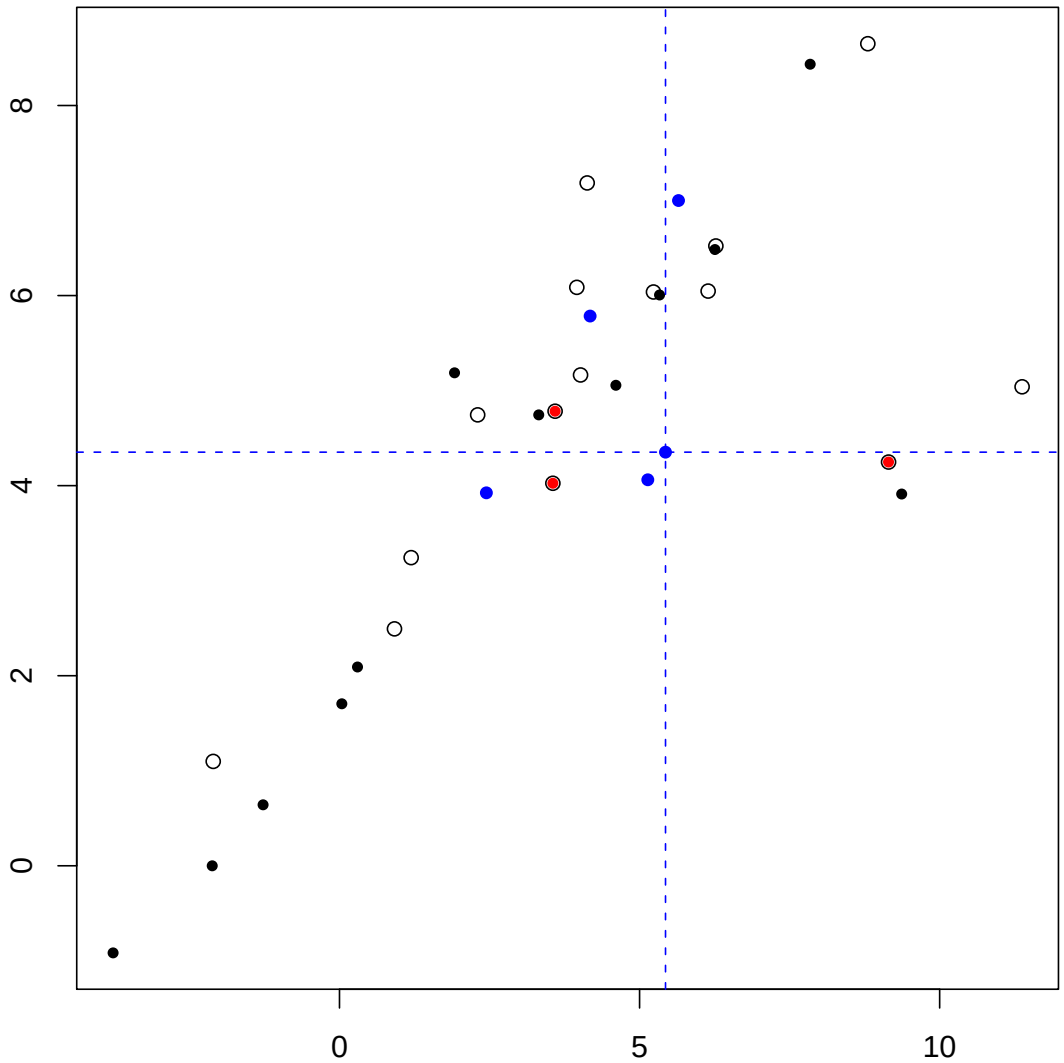
- wähle (z.B) 3 Zufallspunkte
- berechne Lagemaß (z.B Mittelwert)
- ersetze
- bis alle Punkte mikroaggregiert sind



Algorithm 1: simple

Zufallsauswahl

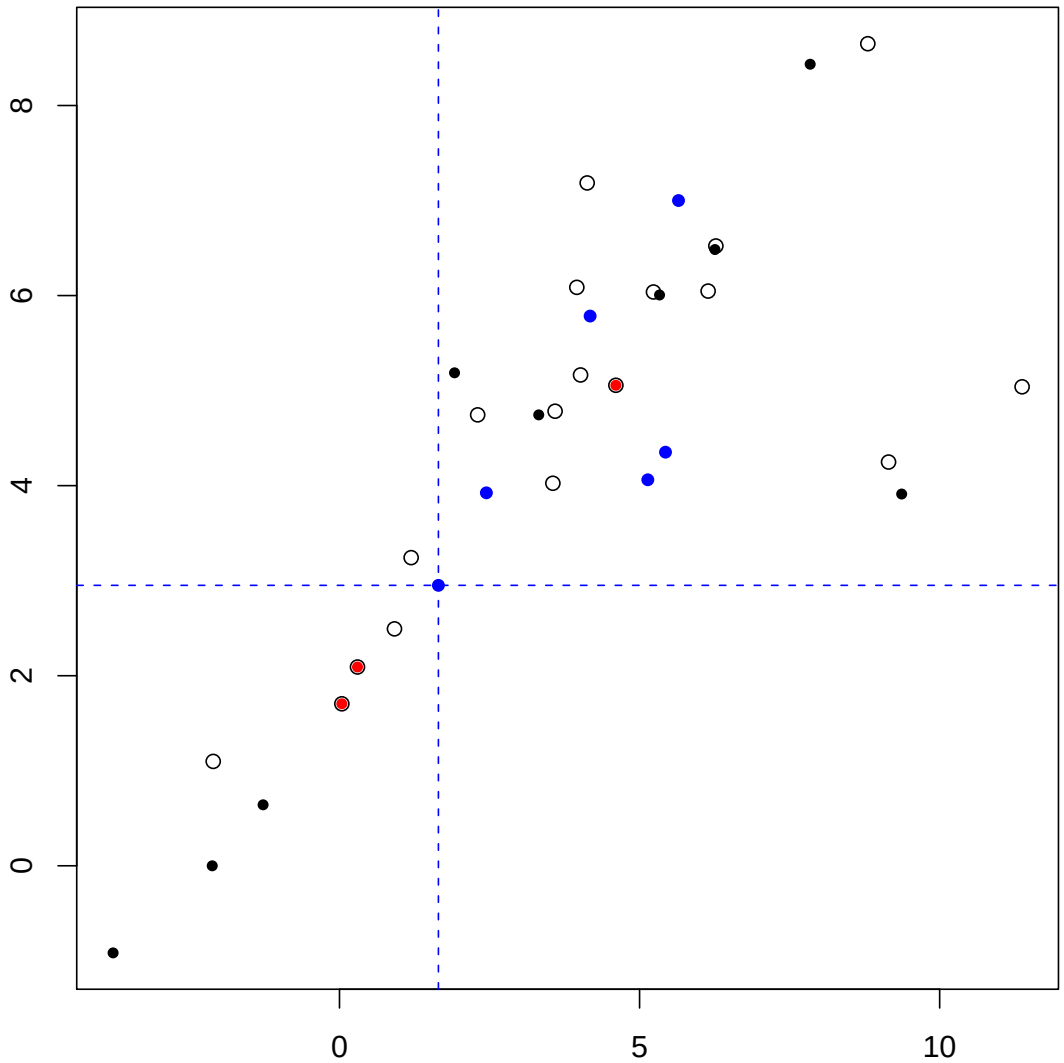
- wähle (z.B) 3 Zufallspunkte
- berechne Lagemaß (z.B Mittelwert)
- ersetze
- bis alle Punkte mikroaggregiert sind



Algorithm 1: simple

Zufallsauswahl

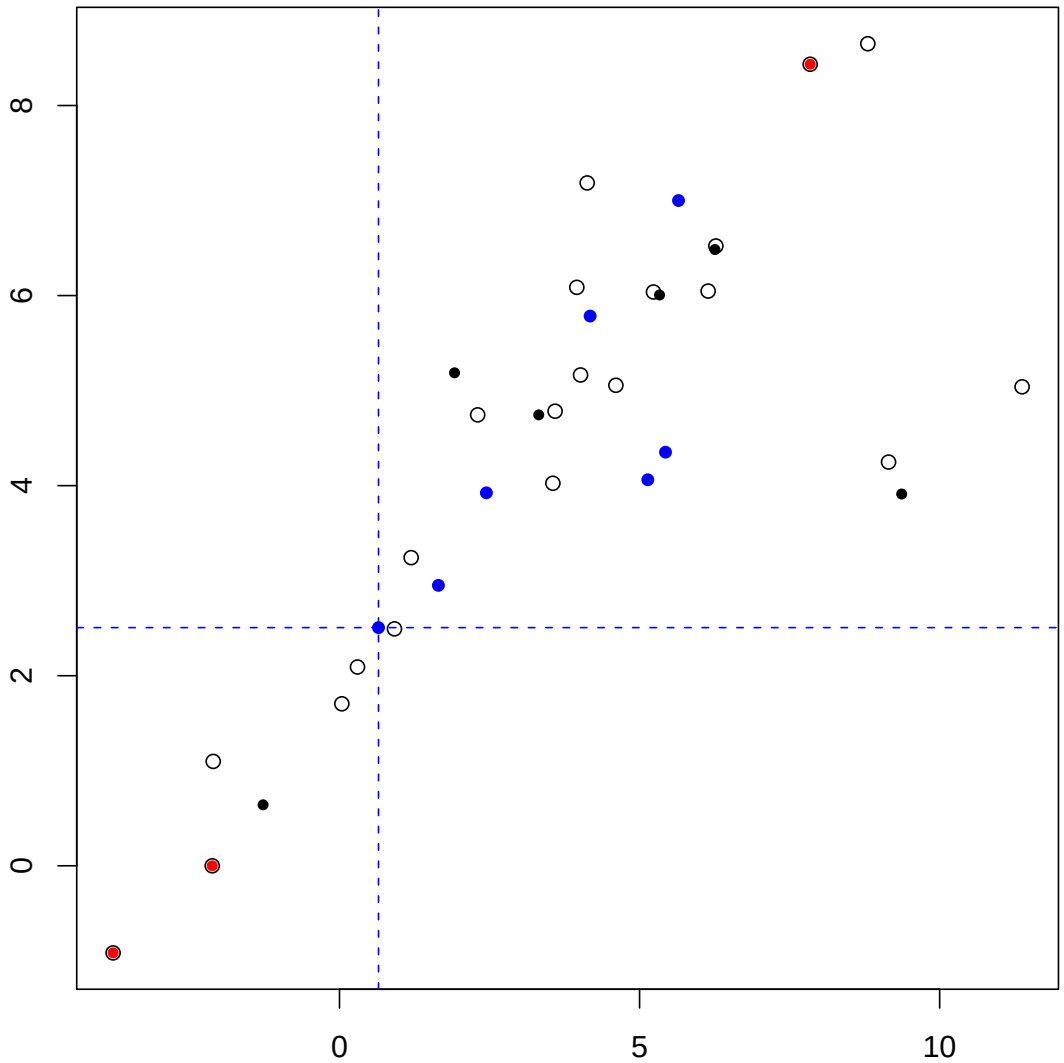
- wähle (z.B) 3 Zufallspunkte
- berechne Lagemaß (z.B Mittelwert)
- ersetze
- bis alle Punkte mikroaggregiert sind



Algorithm 1: simple

Zufallsauswahl

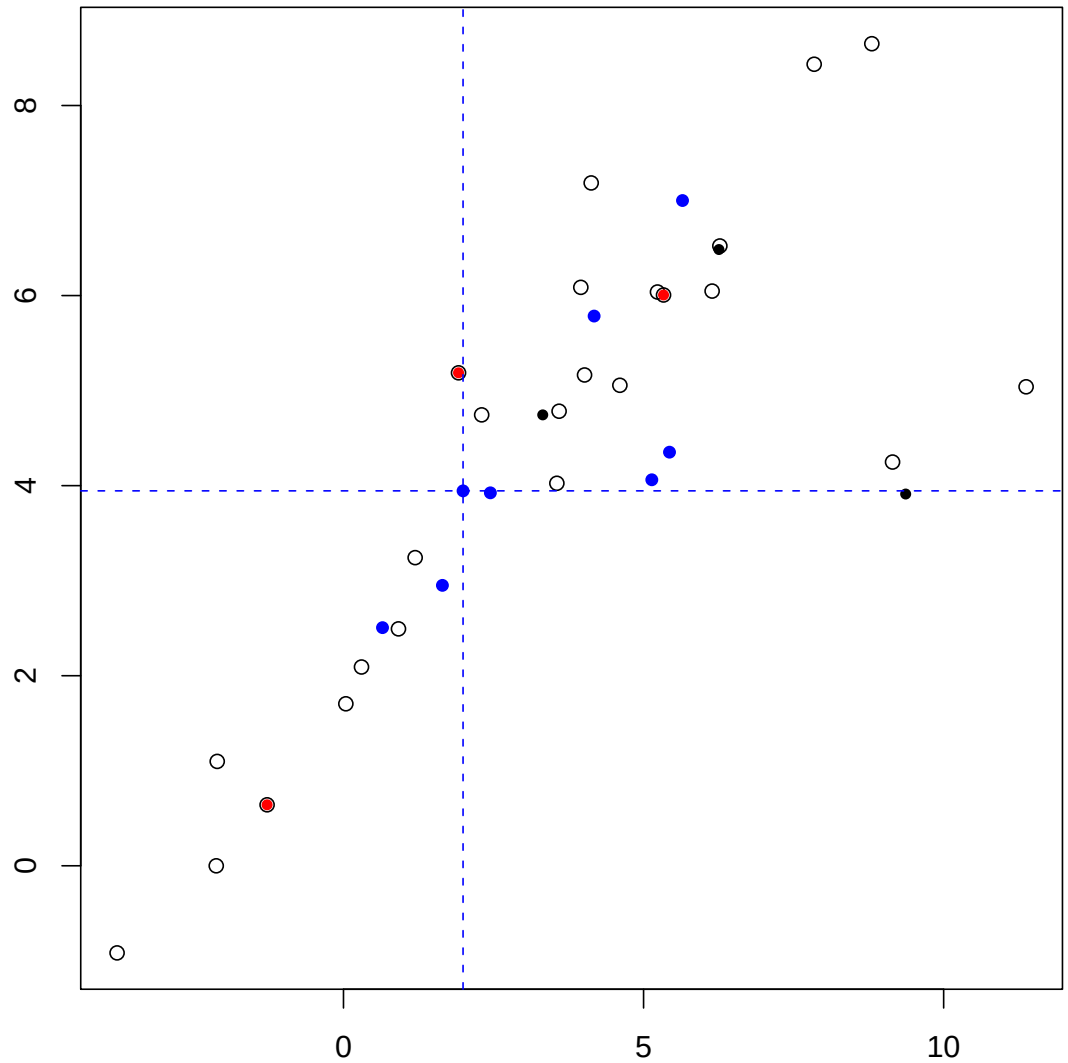
- wähle (z.B) 3 Zufallspunkte
- berechne Lagemaß (z.B Mittelwert)
- ersetze
- bis alle Punkte mikroaggregiert sind



Algorithm 1: simple

Zufallsauswahl

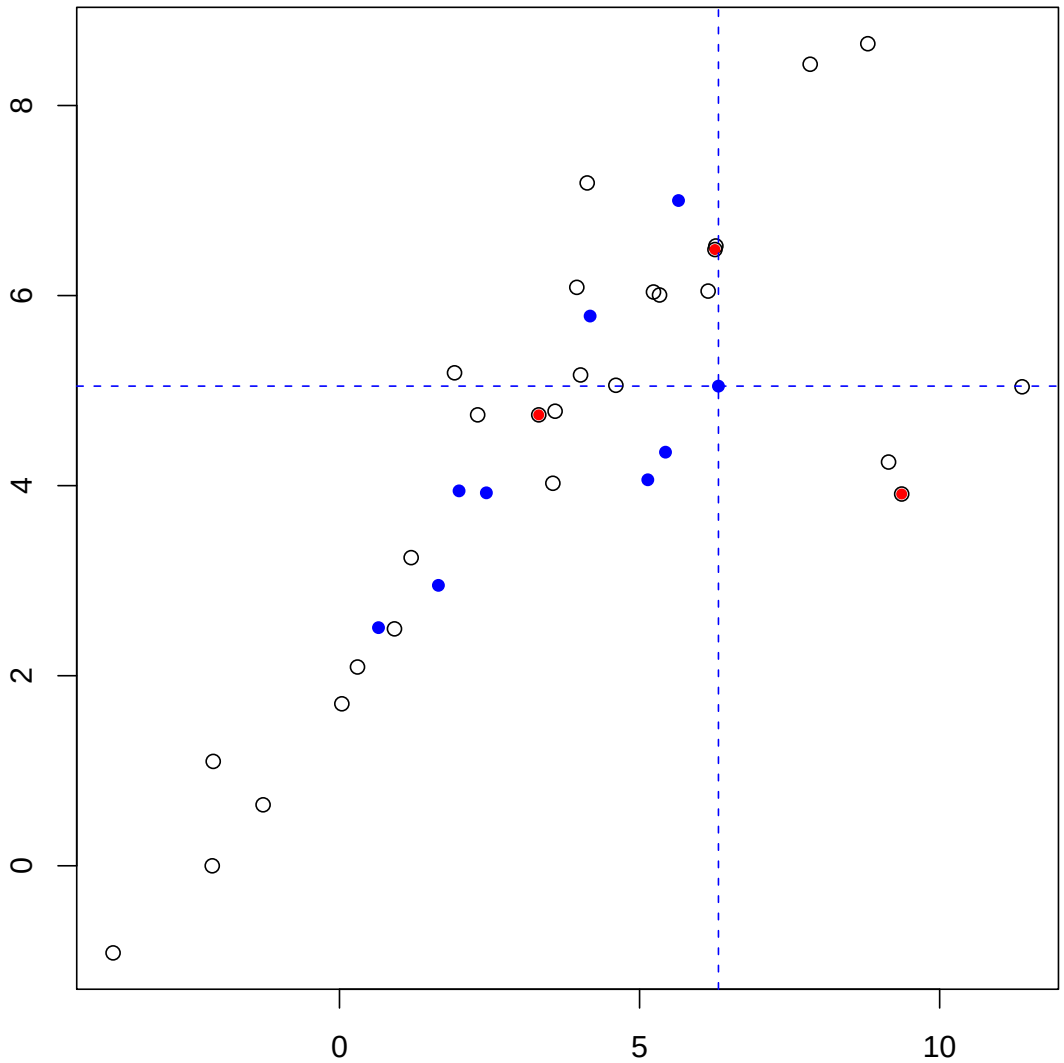
- wähle (z.B) 3 Zufallspunkte
- berechne Lagemaß (z.B Mittelwert)
- ersetze
- bis alle Punkte mikroaggregiert sind



Algorithm 1: simple

Zufallsauswahl

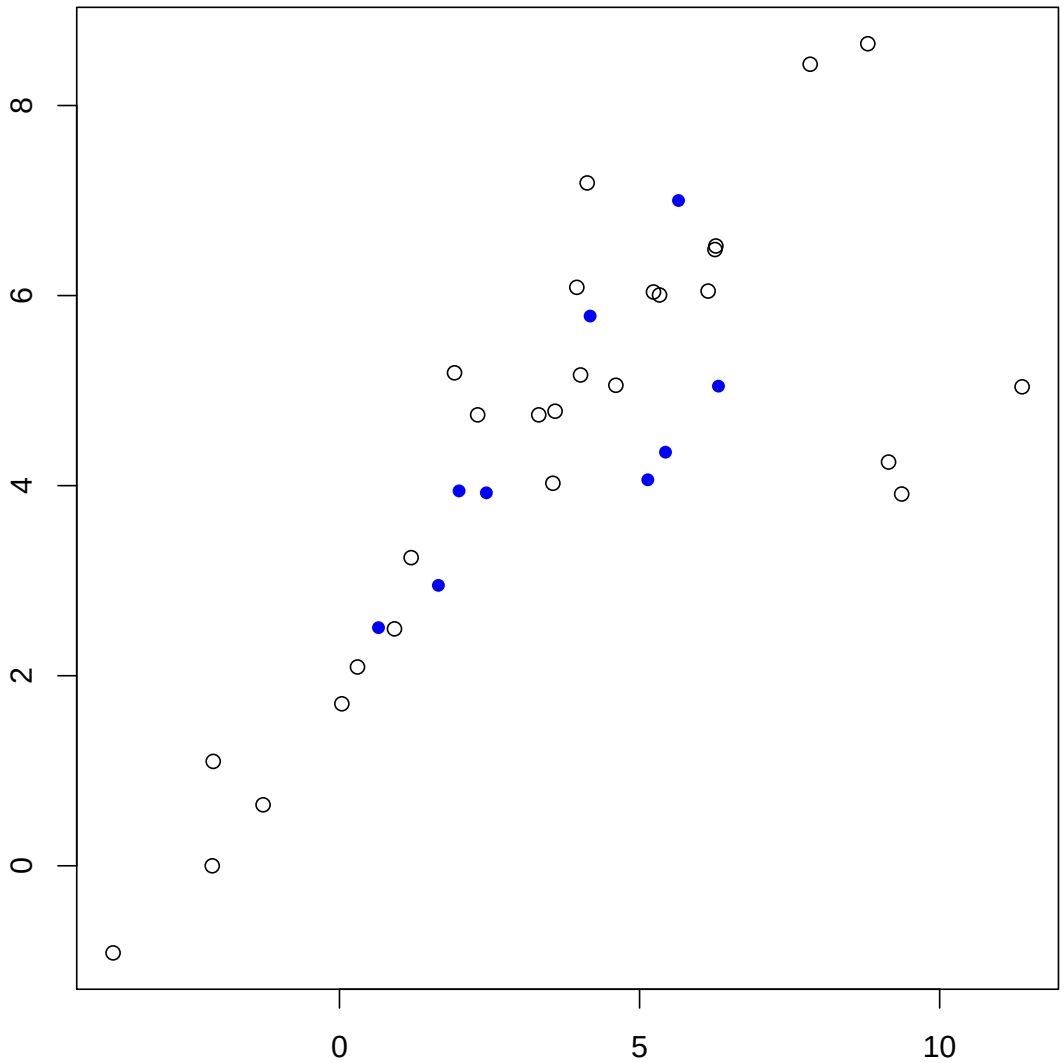
- wähle (z.B) 3 Zufallspunkte
- berechne Lagemaß (z.B Mittelwert)
- ersetze
- bis alle Punkte mikroaggregiert sind



Algorithm 1: simple

Zufallsauswahl

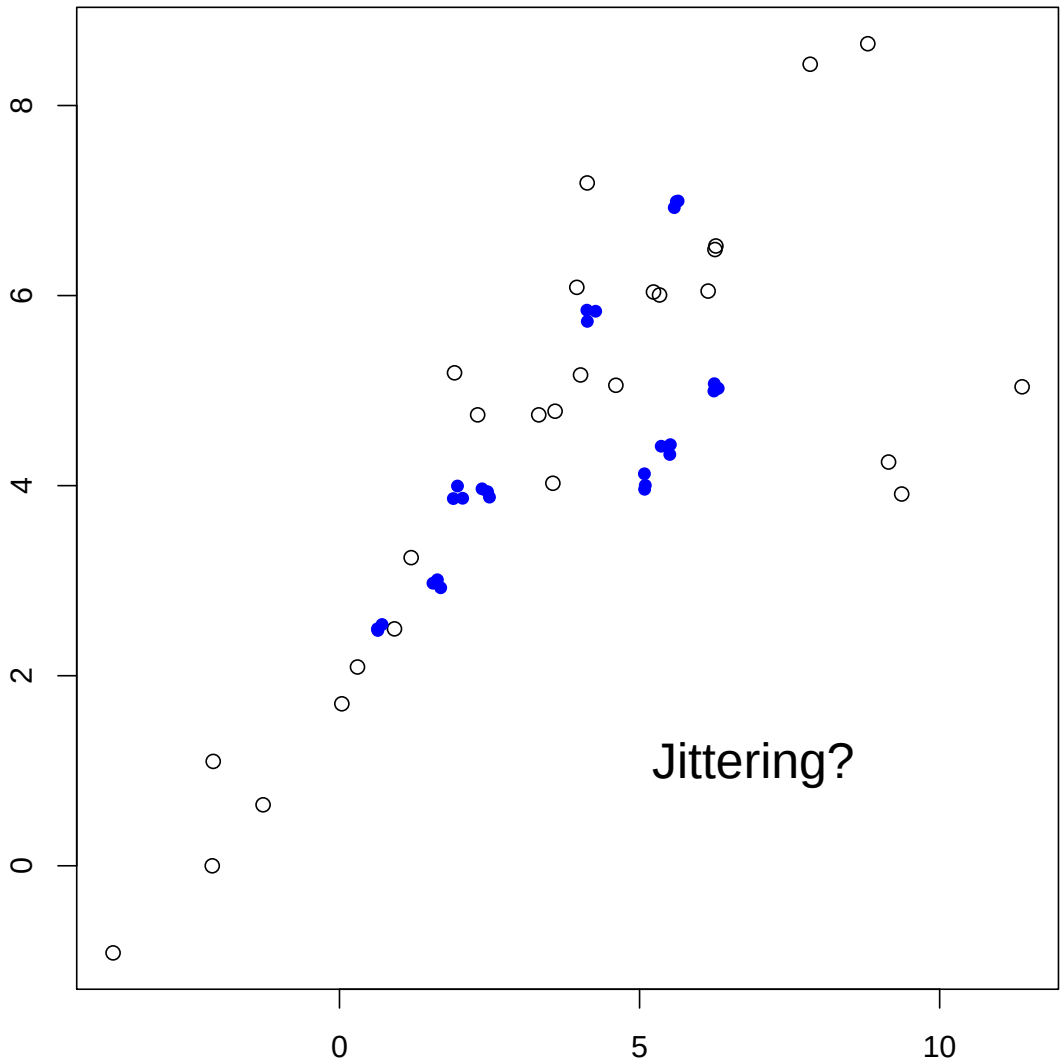
- wähle (z.B) 3 Zufallspunkte
- berechne Lagemaß (z.B Mittelwert)
- ersetze
- bis alle Punkte mikroaggregiert sind
- $\rightarrow$ Fertig!



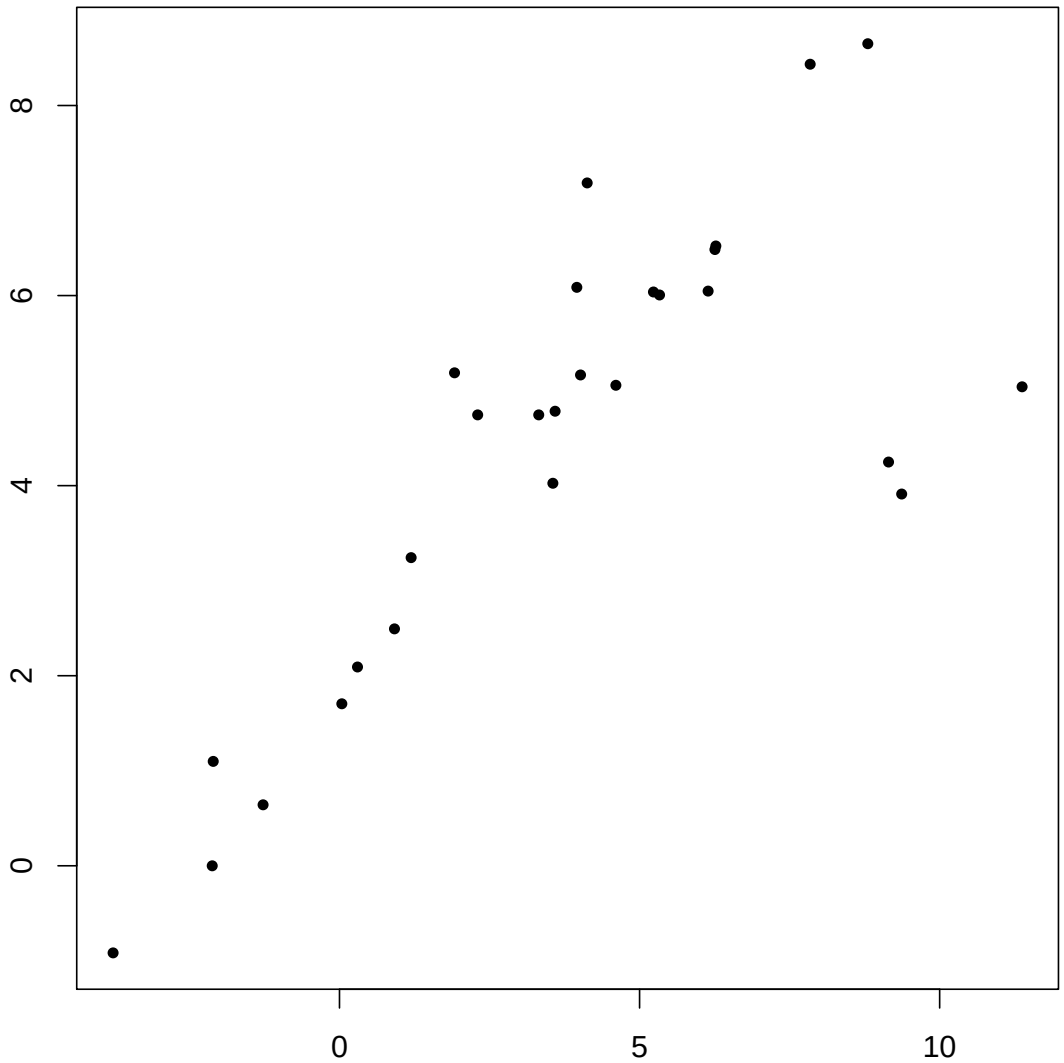
Algorithm 1: simple

Zufallsauswahl

- wähle (z.B) 3 Zufallspunkte
- berechne Lagemaß (z.B Mittelwert)
- ersetze
- bis alle Punkte mikroaggregiert sind
- $\rightarrow$ Fertig!

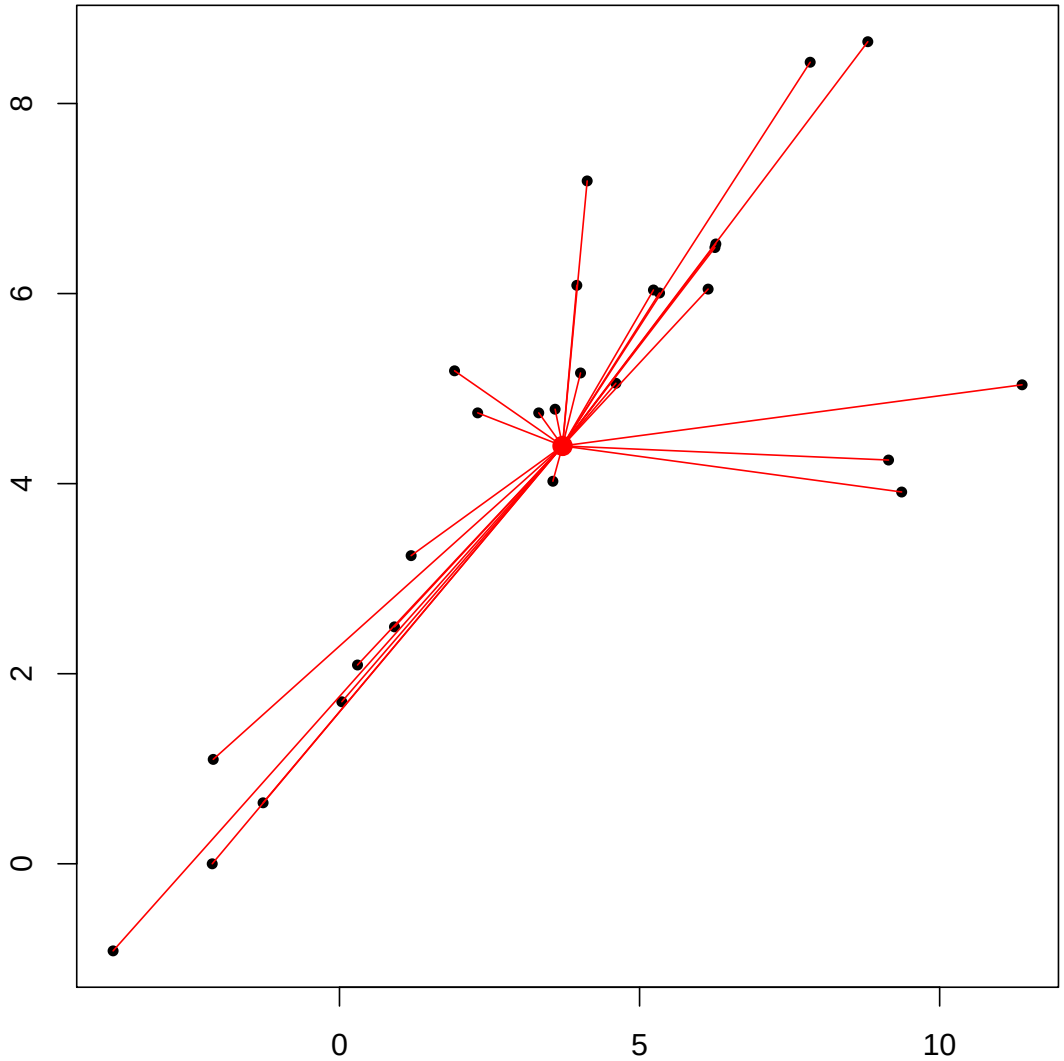


Algorithmus



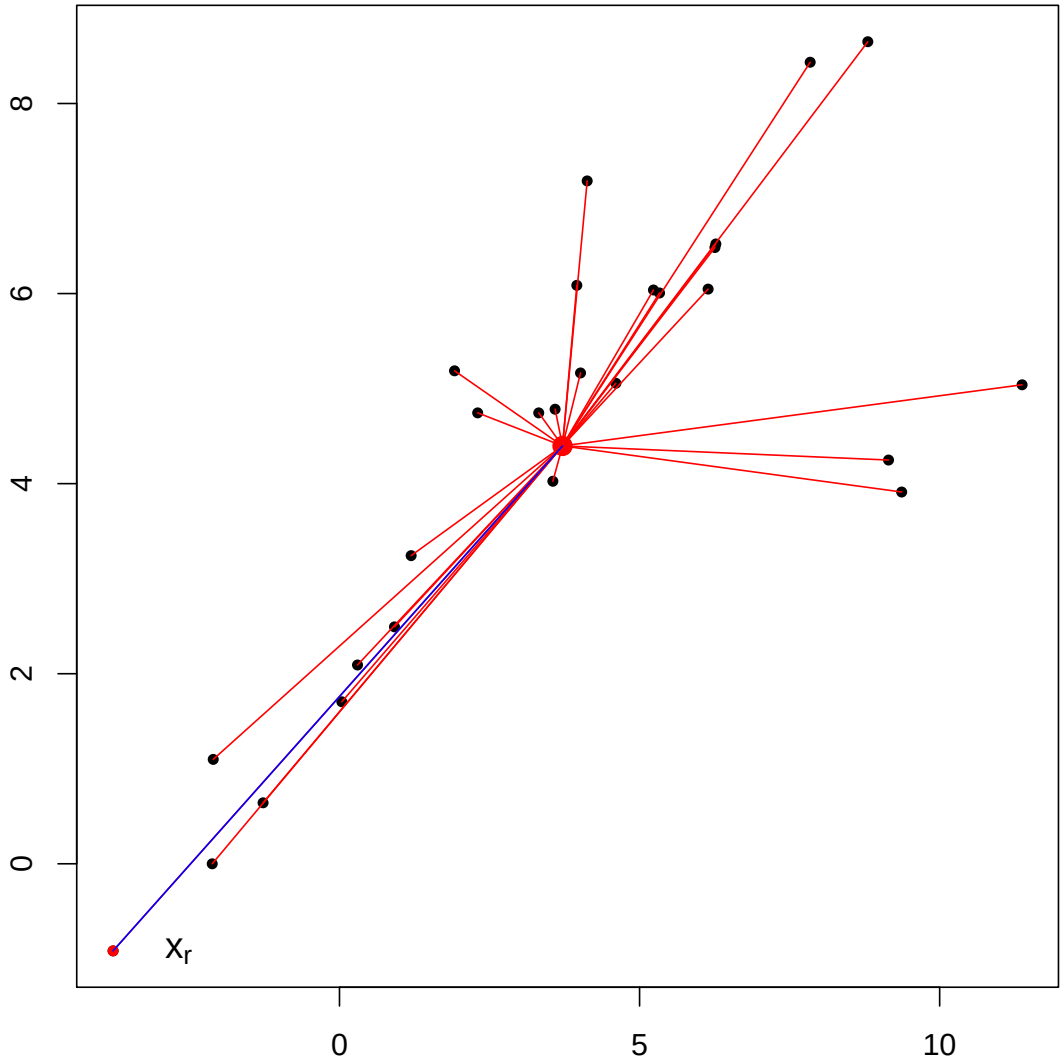
Algorithmus

- berechne Mittelwert



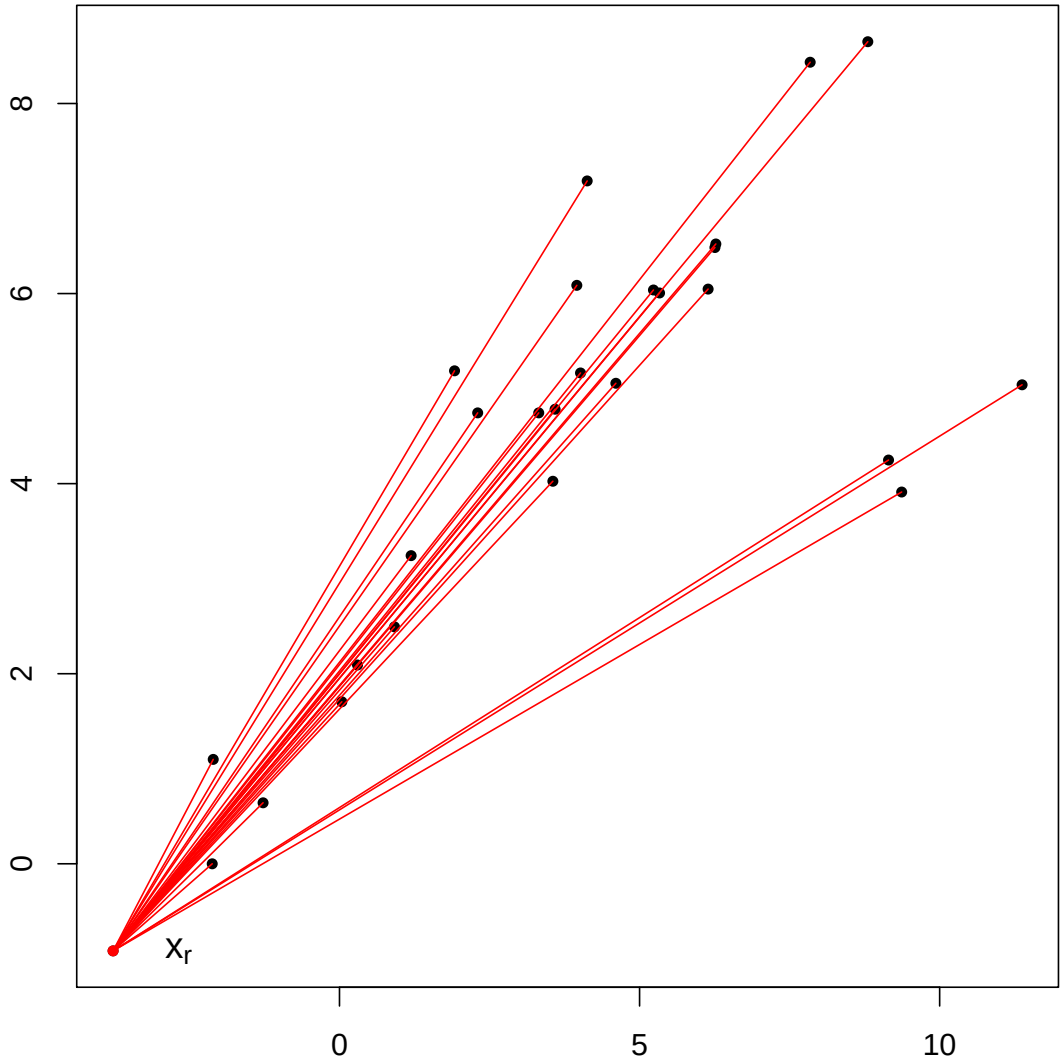
Algorithmus

- berechne Mittelwert
- finde Beobachtung x_r mit max. Abstand



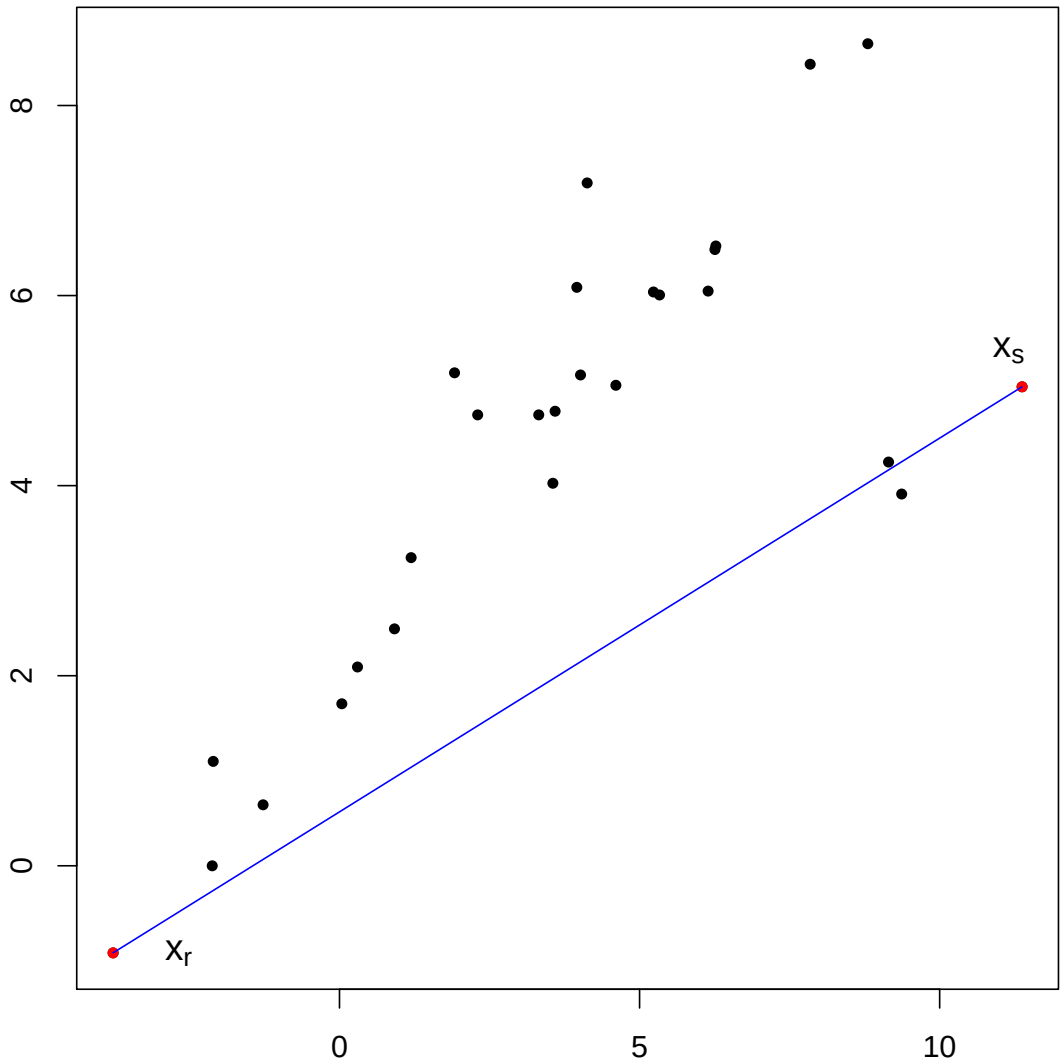
Algorithmus

- berechne Mittelwert
- finde Beobachtung x_r mit max. Abstand
- berechne Distanzen aller Punkte zu x_r



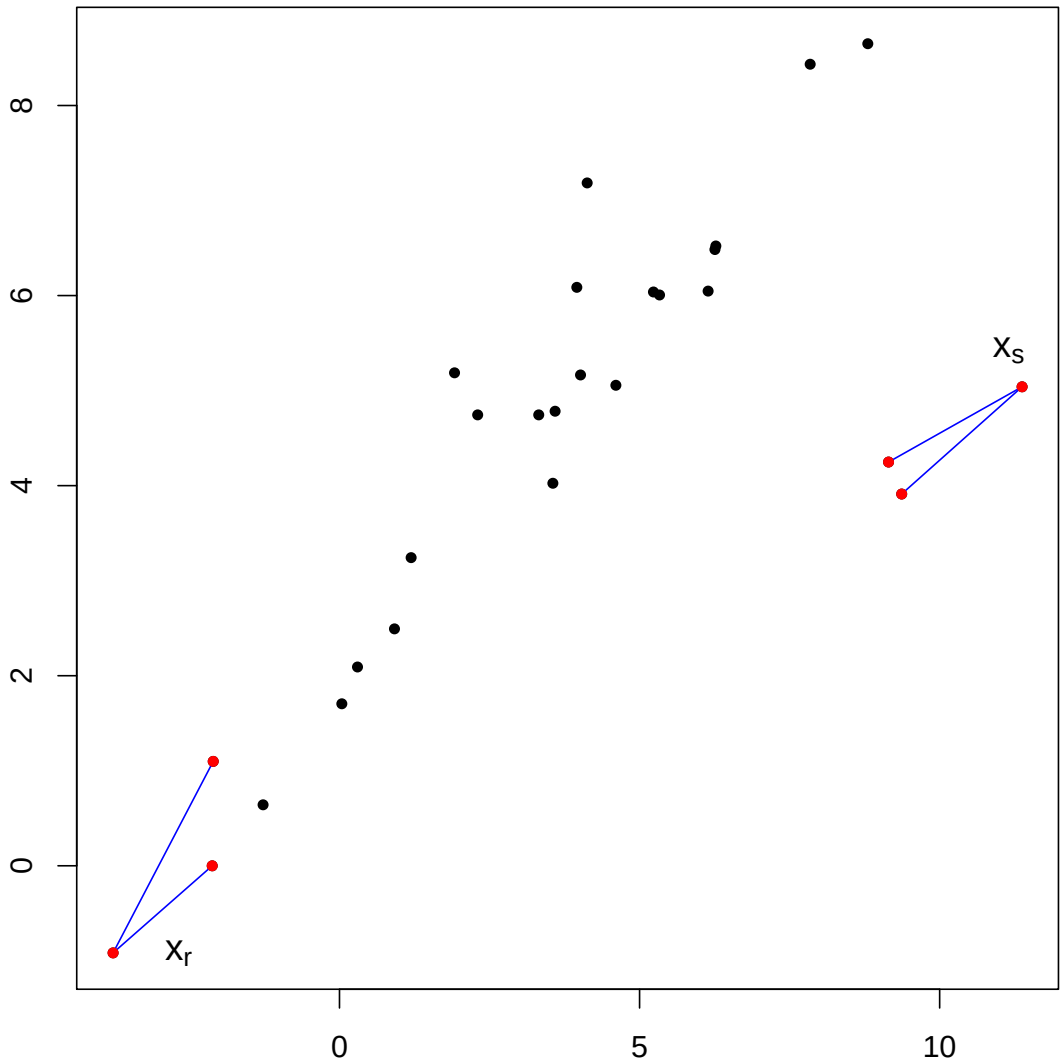
Algorithmus

- berechne Mittelwert
- finde Beobachtung x_r mit max. Abstand
- berechne Distanzen aller Punkte zu x_r
- wähle Beobachtung x_s mit max. Distanz zu x_r



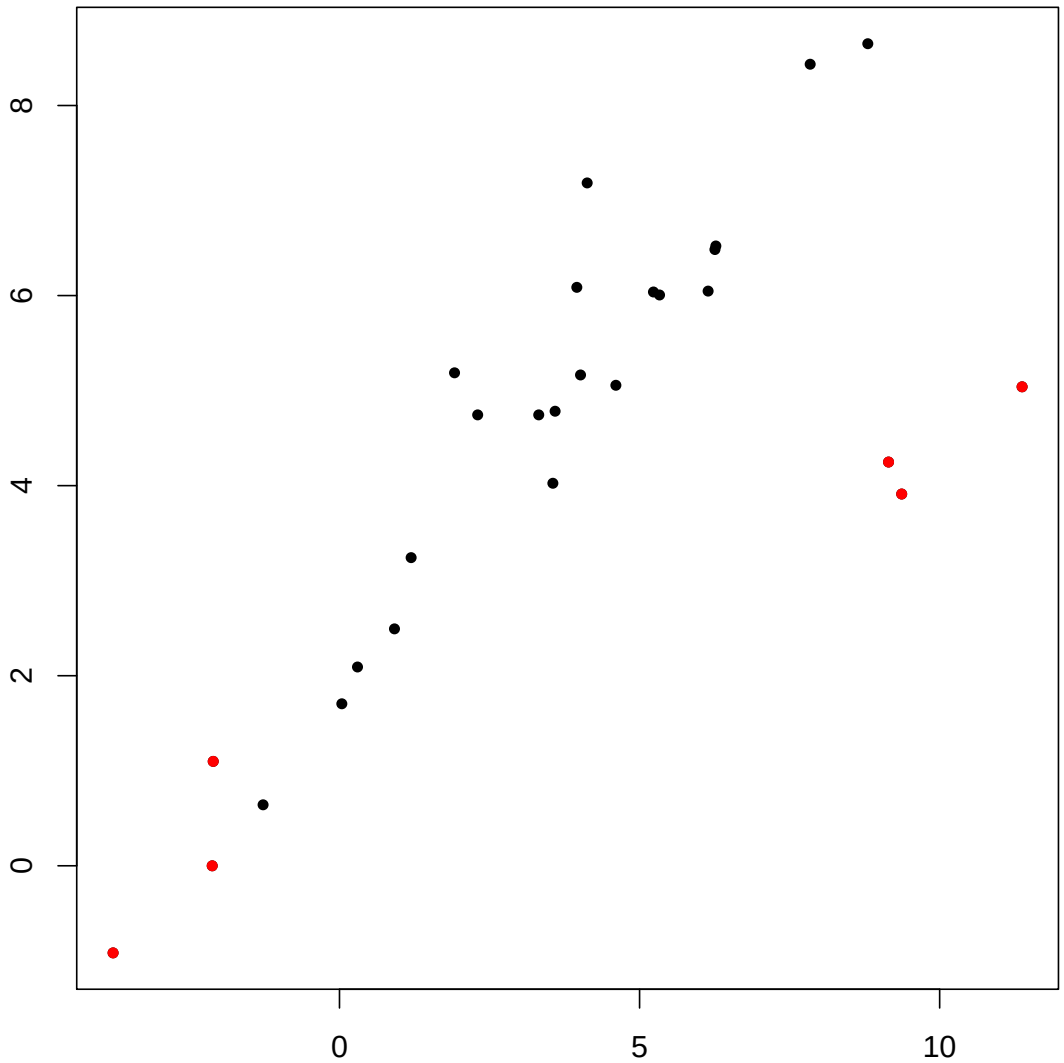
Algorithmus

- berechne Mittelwert
- finde Beobachtung x_r mit max. Abstand
- berechne Distanzen aller Punkte zu x_r
- wähle Beobachtung x_s mit max. Distanz zu x_r
- suche die 2 nächsten Nachbarn von x_r und x_s



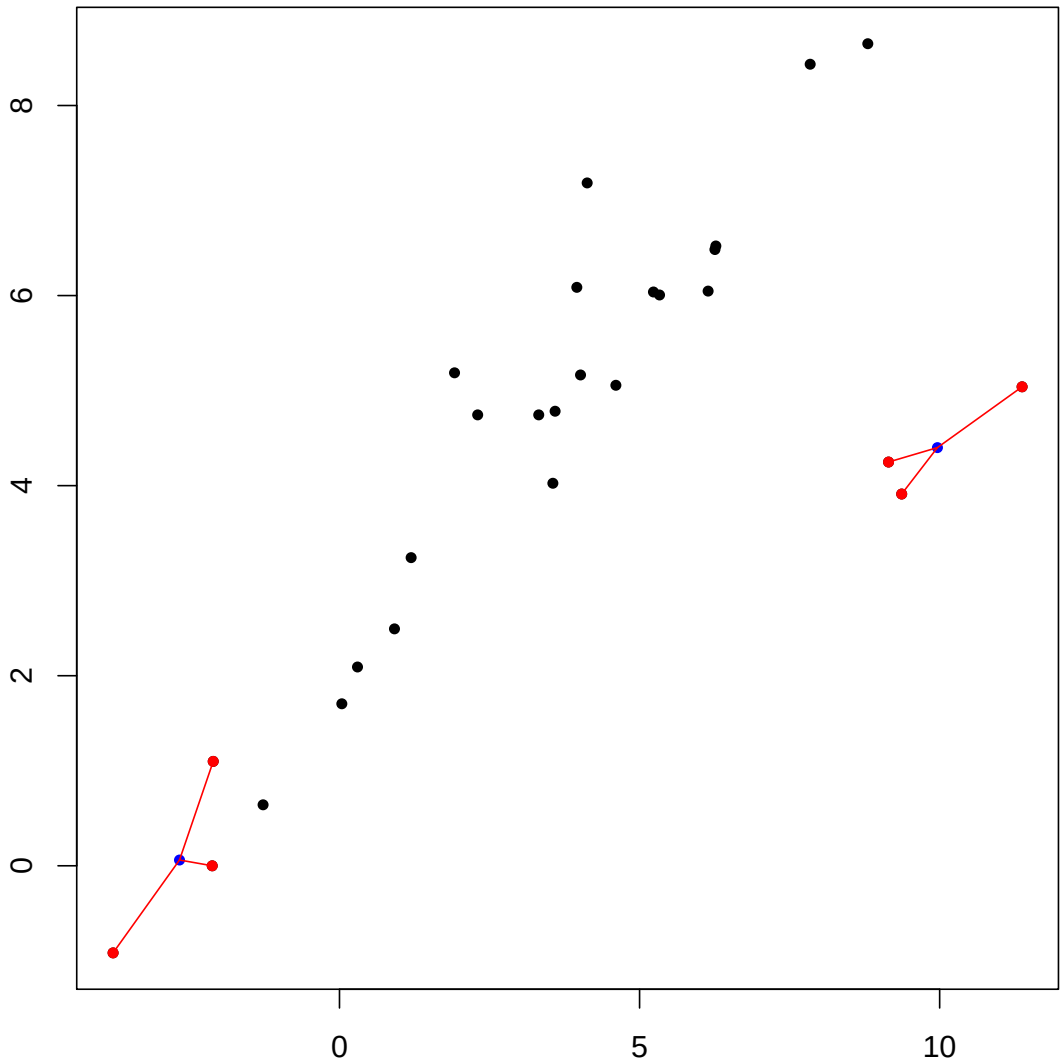
Algorithmus

- berechne Mittelwert
- finde Beobachtung x_r mit max. Abstand
- berechne Distanzen aller Punkte zu x_r
- wähle Beobachtung x_s mit max. Distanz zu x_r
- suche die 2 nächsten Nachbarn von x_r und x_s



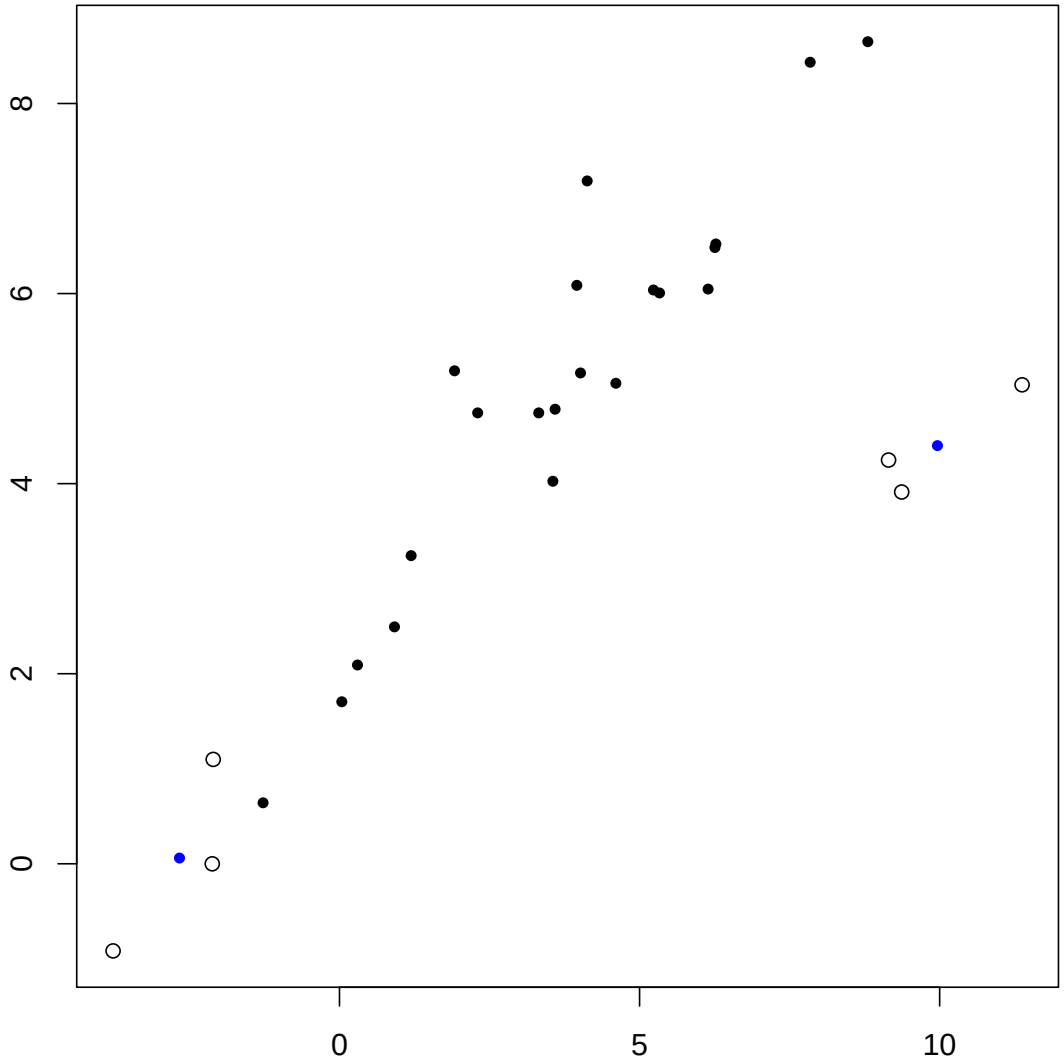
Algorithmus

- berechne Mittelwert
- finde Beobachtung x_r mit max. Abstand
- berechne Distanzen aller Punkte zu x_r
- wähle Beobachtung x_s mit max. Distanz zu x_r
- suche die 2 nächsten Nachbarn von x_r und x_s
- Aggregiere diese mit dem Mittelwert



Algorithmus

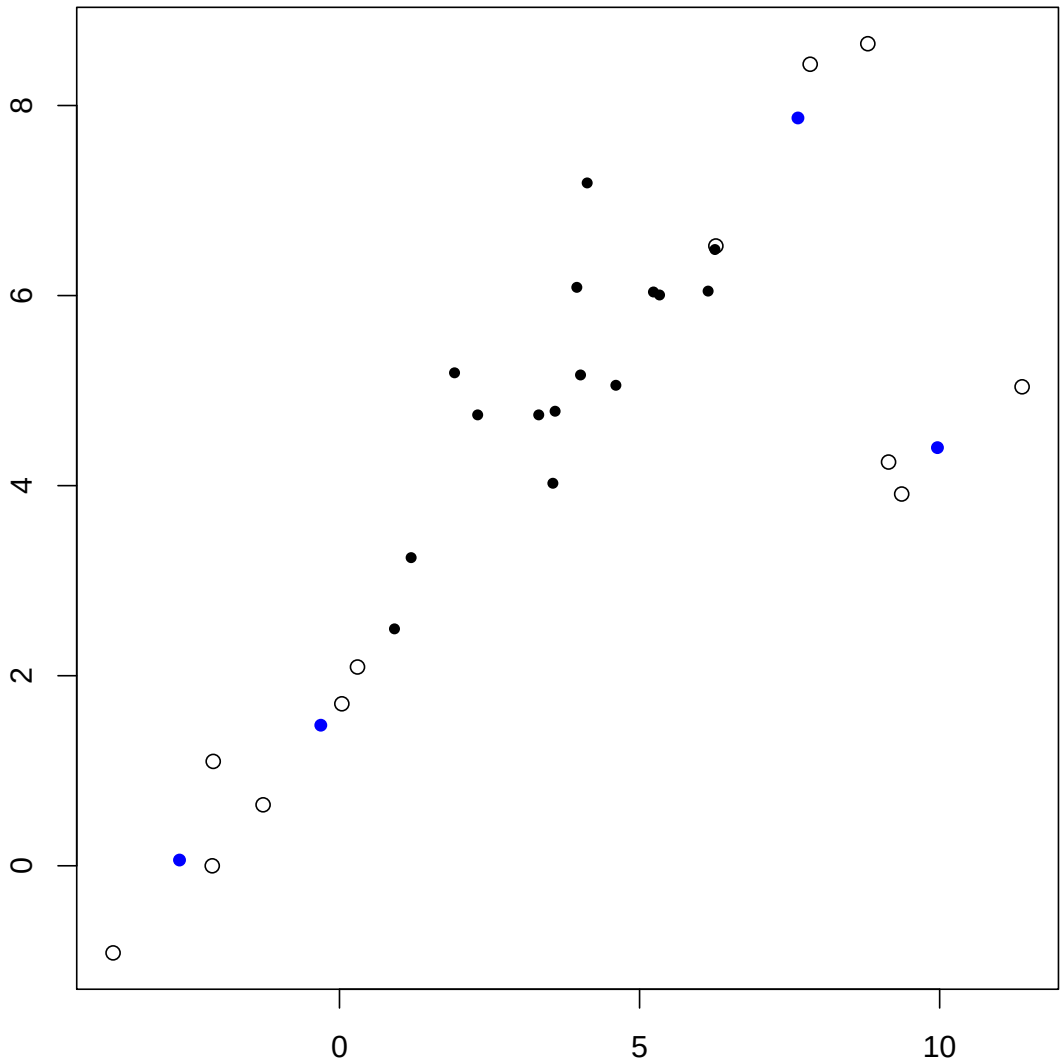
- berechne Mittelwert
- finde Beobachtung x_r mit max. Abstand
- berechne Distanzen aller Punkte zu x_r
- wähle Beobachtung x_s mit max. Distanz zu x_r
- suche die 2 nächsten Nachbarn von x_r und x_s
- Aggregiere diese mit dem Mittelwert



Algorithmus 2: mdav

Algorithmus

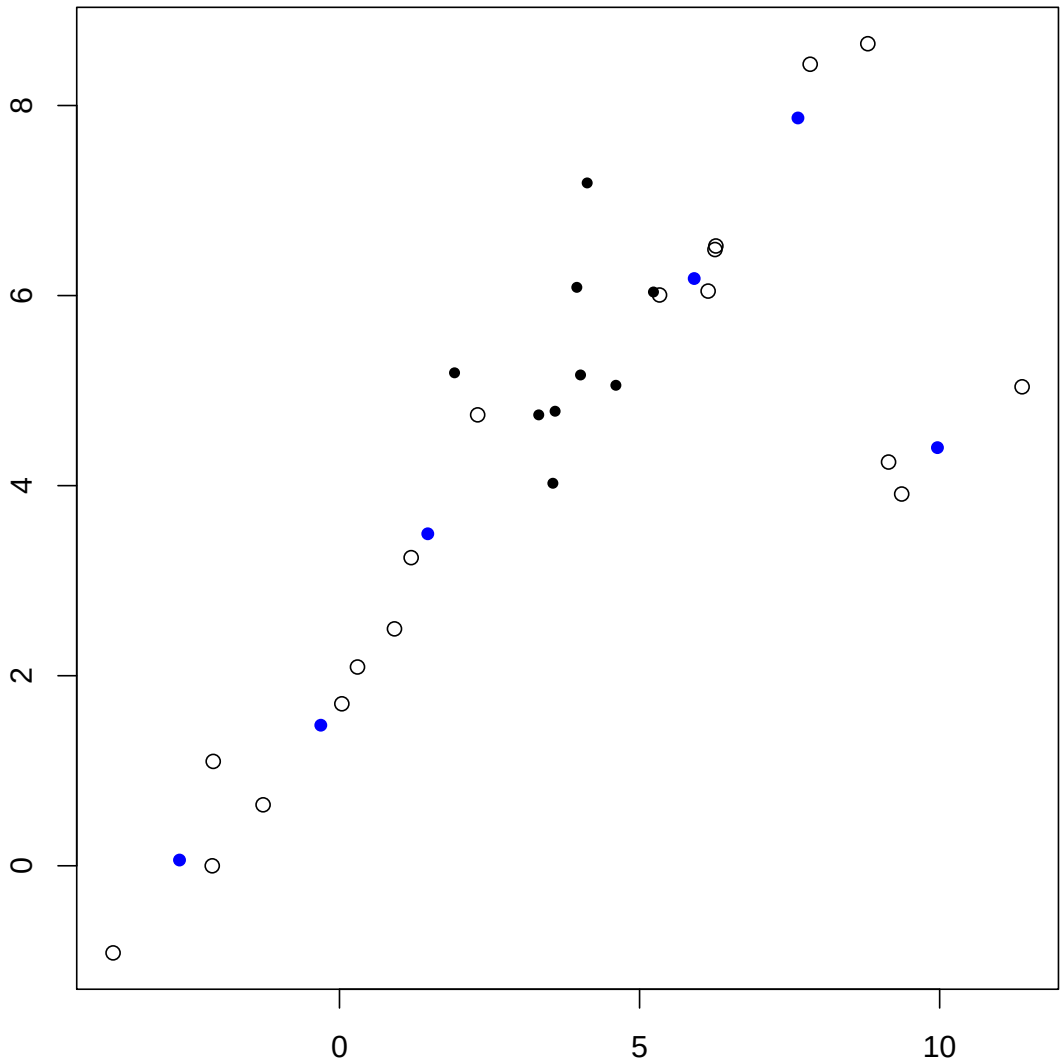
- berechne Mittelwert
- finde Beobachtung x_r mit max. Abstand
- berechne Distanzen aller Punkte zu x_r
- wähle Beobachtung x_s mit max. Distanz zu x_r
- suche die 2 nächsten Nachbarn von x_r und x_s
- Aggregiere diese mit dem Mittelwert
- fahre fort, bis alle Beobachtungen aggregiert sind (Spezialregeln am Ende)



Algorithmus 2: mdav

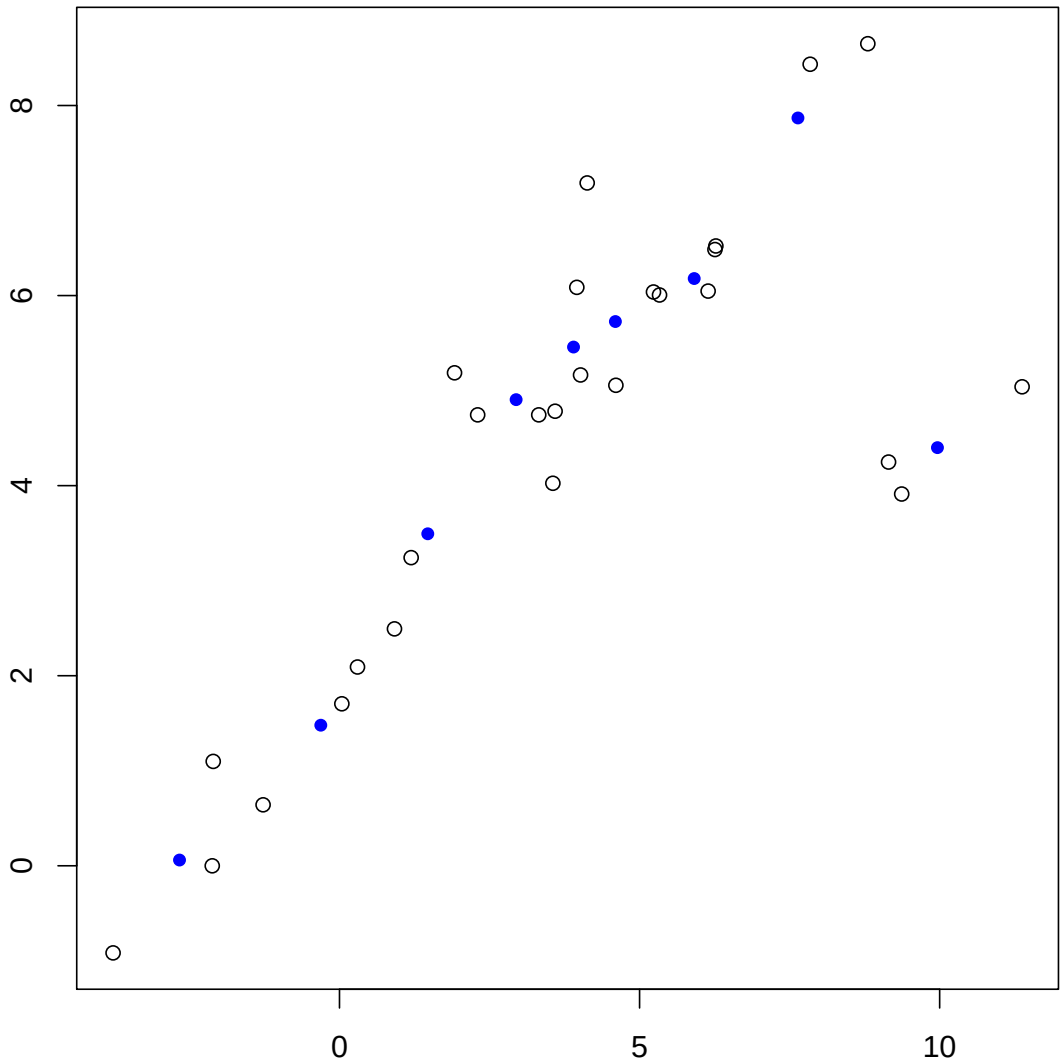
Algorithmus

- berechne Mittelwert
- finde Beobachtung x_r mit max. Abstand
- berechne Distanzen aller Punkte zu x_r
- wähle Beobachtung x_s mit max. Distanz zu x_r
- suche die 2 nächsten Nachbarn von x_r und x_s
- Aggregiere diese mit dem Mittelwert
- fahre fort, bis alle Beobachtungen aggregiert sind (Spezialregeln am Ende)



Algorithmus

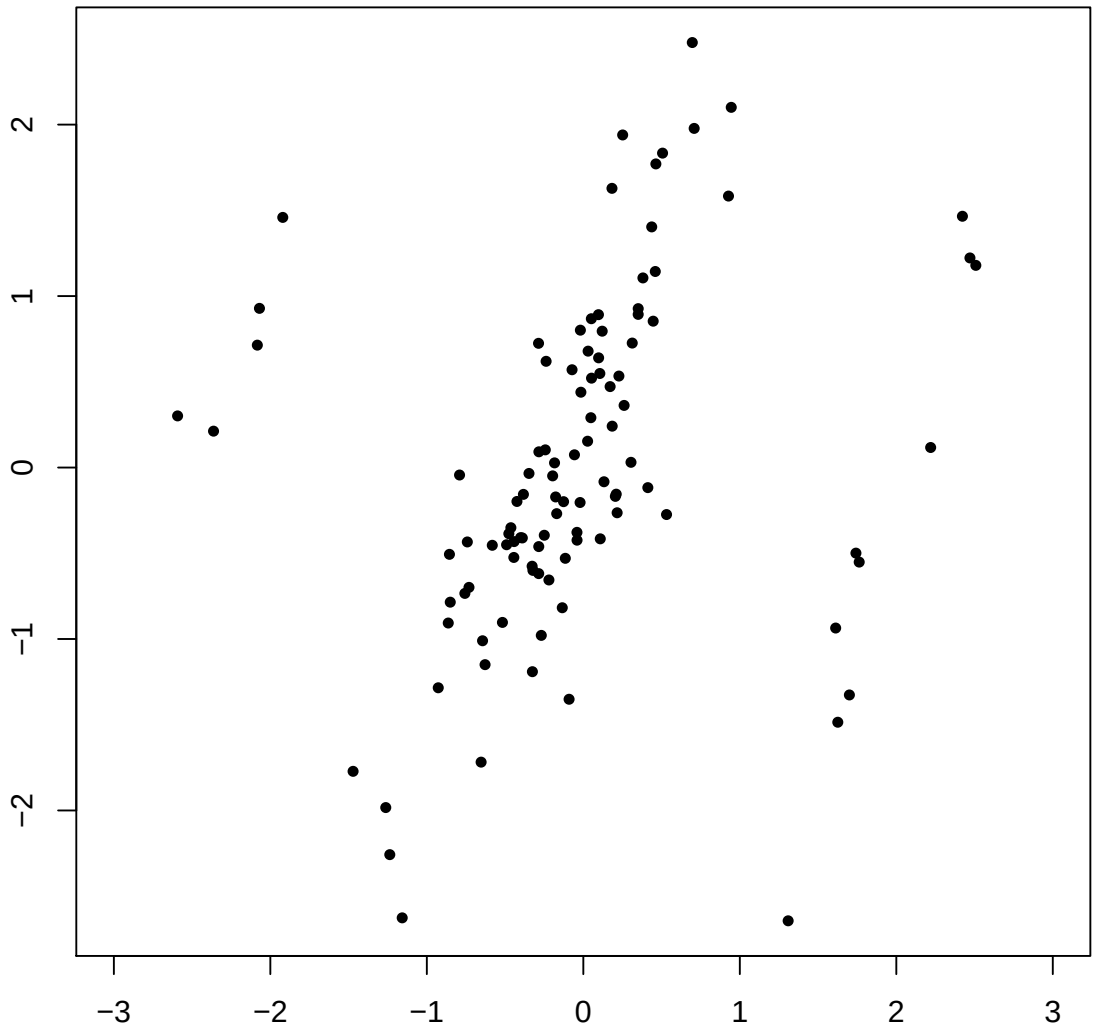
- berechne Mittelwert
- finde Beobachtung x_r mit max. Abstand
- berechne Distanzen aller Punkte zu x_r
- wähle Beobachtung x_s mit max. Distanz zu x_r
- suche die 2 nächsten Nachbarn von x_r und x_s
- Aggregiere diese mit dem Mittelwert
- fahre fort, bis alle Beobachtungen aggregiert sind (Spezialregeln am Ende)



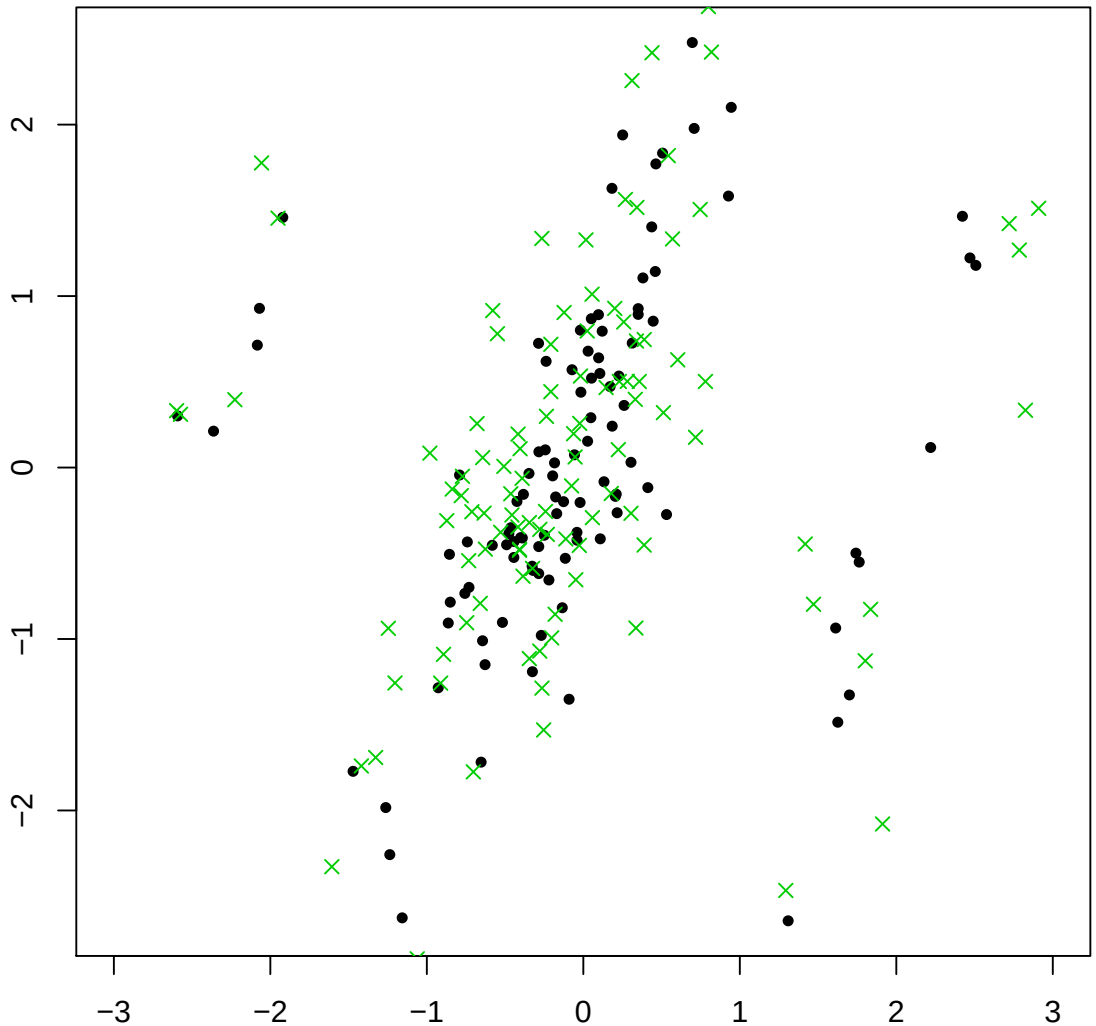
Messung des Re-Identifizierungsrisikos

- Distanzbasierte *Record Linkage* Methoden vergleichen die Originalwerte mit den perturbierten Werten mit Hilfe einer Metrik (z.B. Euklidische Distanz).
- Überprüfung, ob der Originalwert in einem Intervall um dem perturbierten Wert liegt.

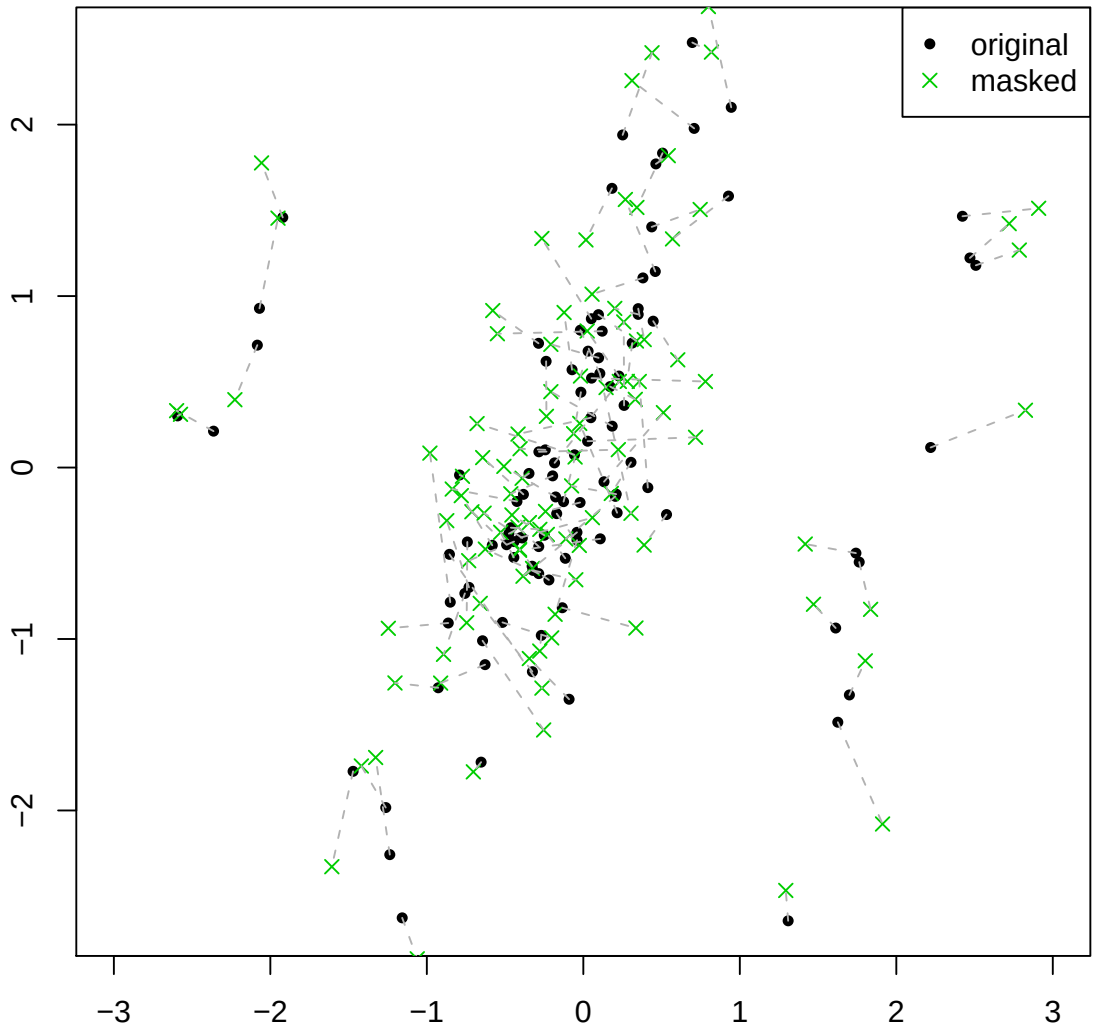
Beispiel



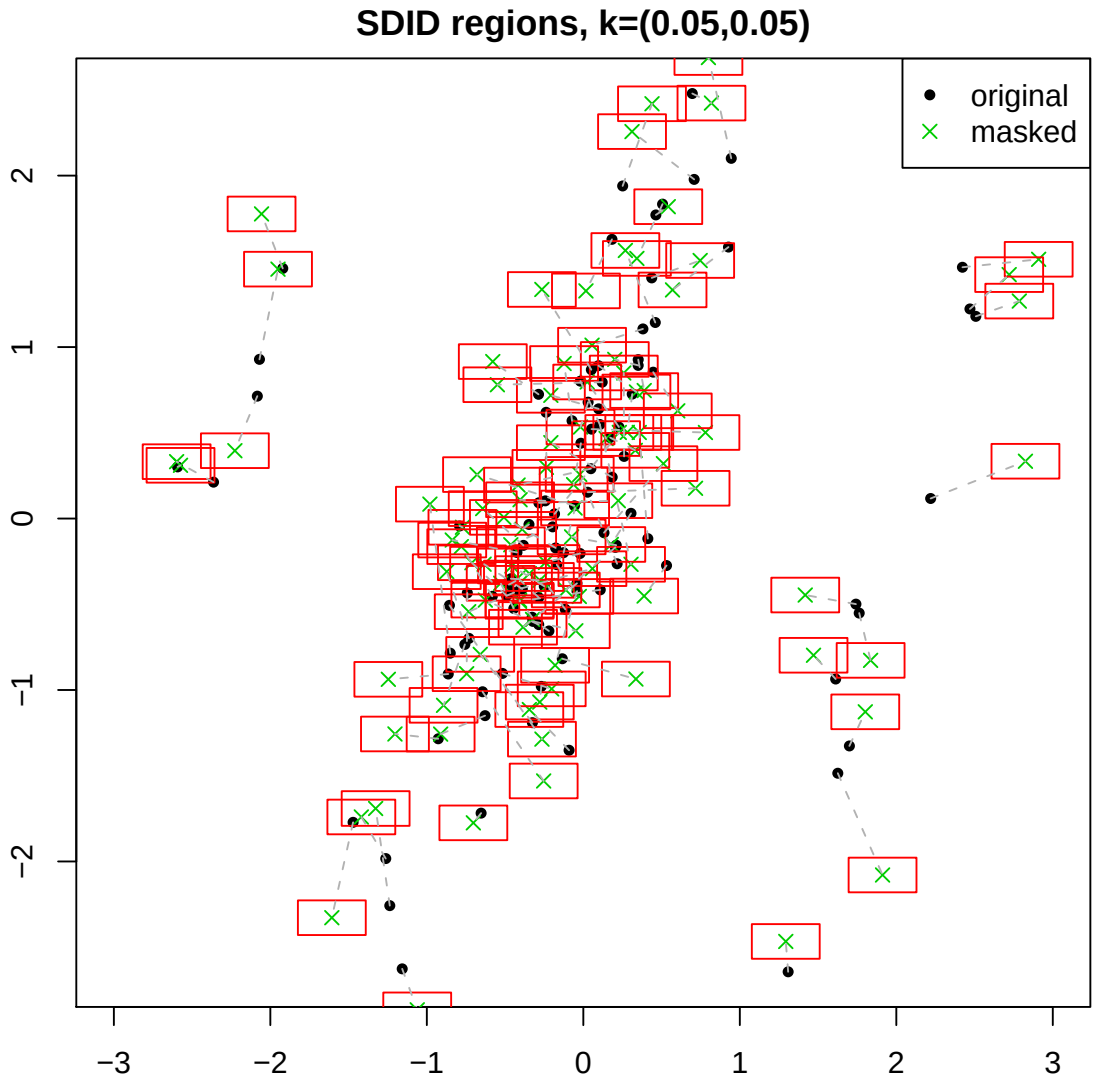
Beispiel



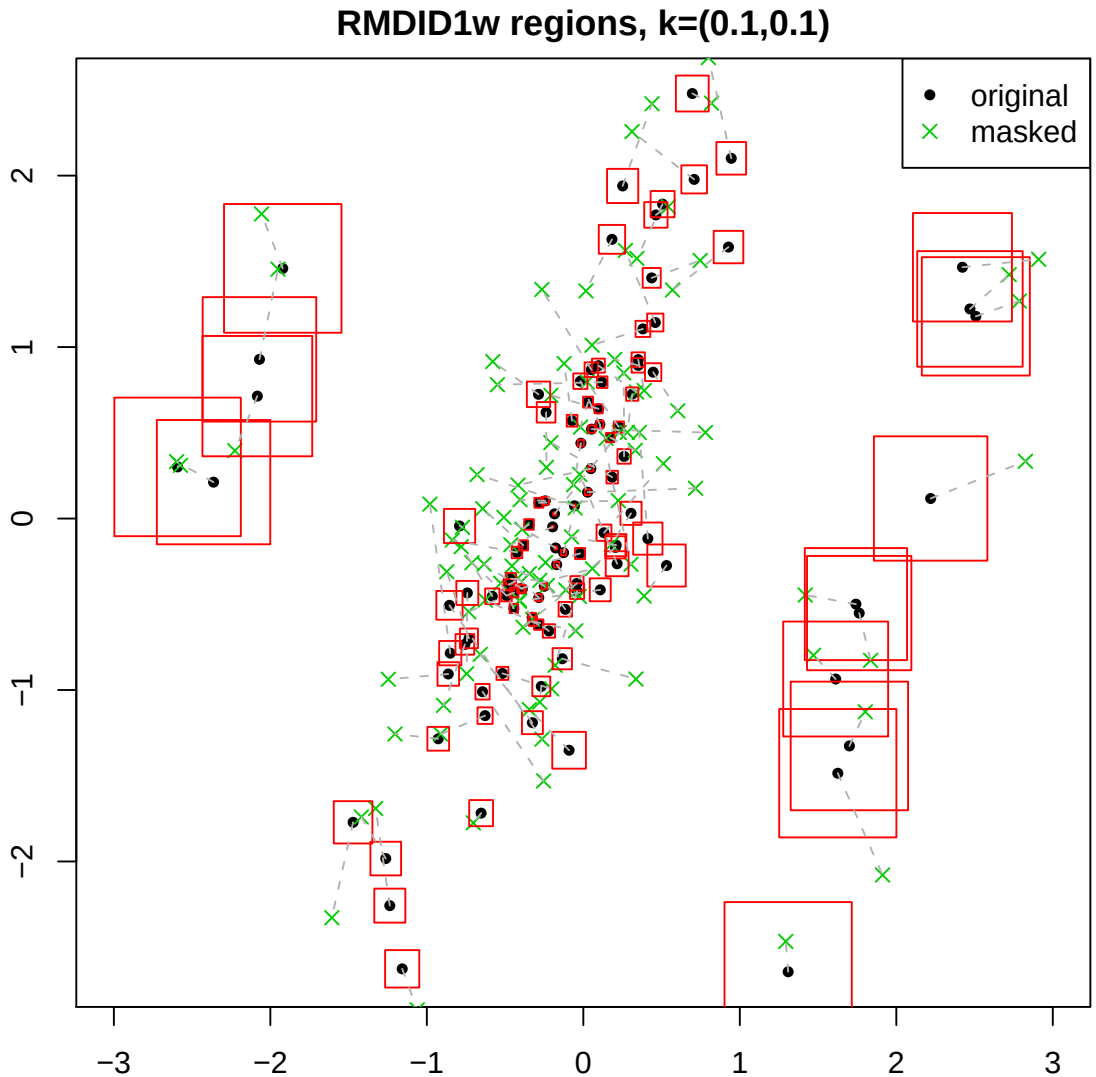
Beispiel



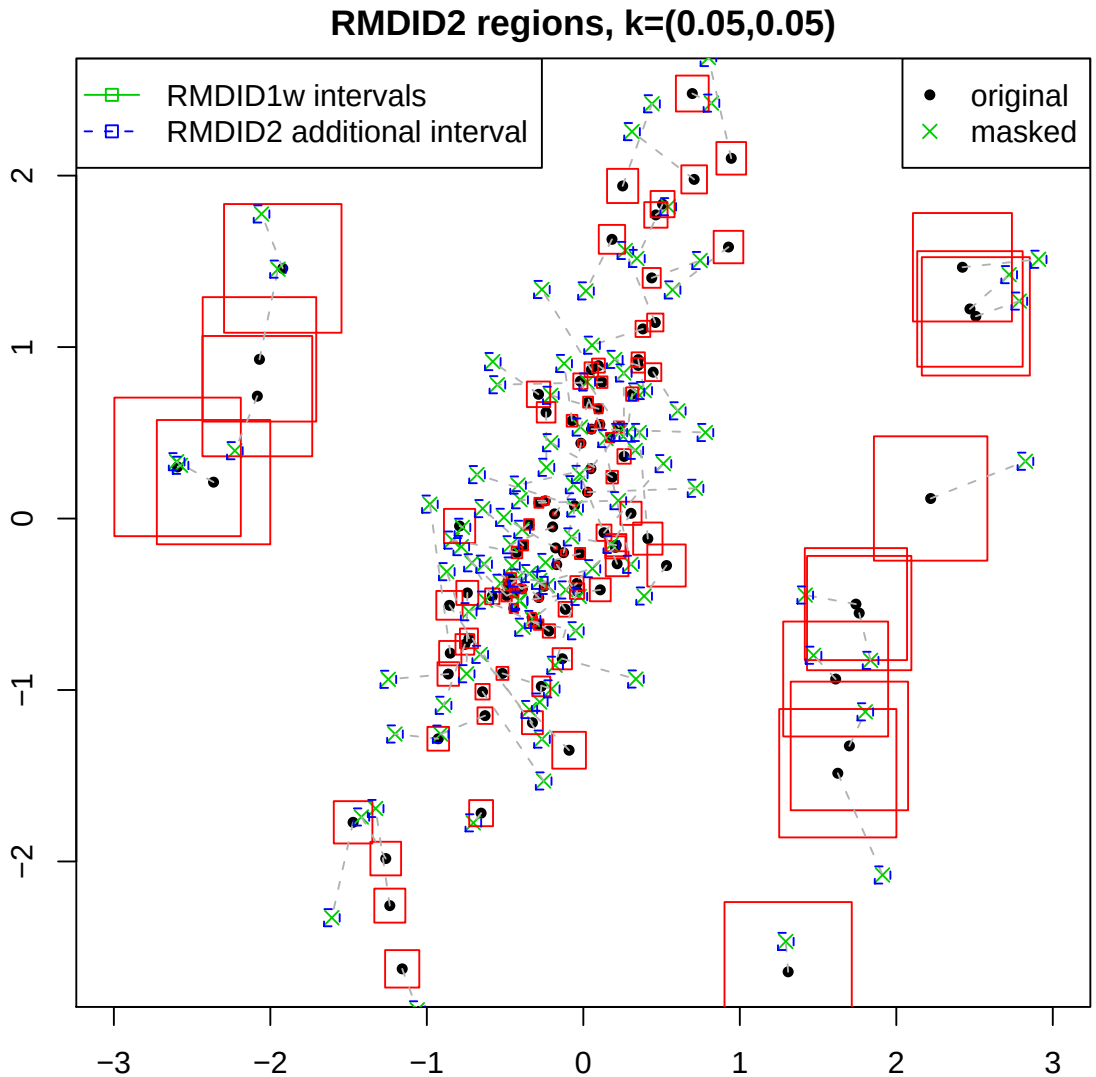
Beispiel



Beispiel (besser)



Beispiel (besser)



Anonymisierung von Tabellen

STATISTIK AUSTRIA
Die Informationsmanager

- Statistische Institutionen veröffentlichen ihre Daten zumeist in aggregierter Form (Tabellen)
- auch in Tabellen muss eine mögliche Reidentifizierung statistischer Einheiten unterbunden werden.
- die Anonymisierung von hierarchisch aufgebaute Tabellen ist höchstgradig komplex

Start: Mikrodaten

- Berechnung von Tabellen mit Randsummen
- Ein Datenangreifer kann Informationen über statistische Einheiten erlangen, falls zu wenige Einheiten zu einer Zelle beitragen
- Zellsperrung als Standardmethode (es gibt auch andere Methoden)

	1	2	3	sum
1	20	50	10	80
2	8	19	22	49
3	17	32	12	61
sum	45	101	44	190

Anonymisieren von Tabellen - Primärsperrung

- k **Threshold Regel**: Sperre jene Zellwerte, wo $< k$ statistische Einheiten beitragen (in der STAT zumeist 3er oder 4er Regel)
- **Problem** z.B. in Branchen mit "Monopolisten"
- **Abhilfe:** p-Prozent Regel (Sperren, wenn der Zellwert minus der beiden größten Beitragenden mehr als p Prozent des größten Beitrags ausmacht)
- **weitere Regel ((n, k) Regel)**: Sperren, wenn n statistische Einheiten mehr als k Prozent zum Zellenwert beitragen ($n < 3$ problematisch - Konkurrenz weiß die Information vom Mitbewerber!)

Anonymisieren von Tabellen - Primärsperrung

	1	2	3	sum
1	20	50	10	80
2	8	19	22	49
3	17	32	12	61
sum	45	101	44	190

Anonymisieren von Tabellen - Primärsperrung

	1	2	3	sum
1	20	50	10	80
2	NA	19	22	49
3	17	32	12	61
sum	45	101	44	190

Anonymisieren von Tabellen - Primärsperrung

	1	2	3	sum
1	20	50	10	80
2	NA	19	22	49
3	17	32	12	61
sum	45	101	44	190

- jede Tabelle mit Randsummen besitzt lineare Abhängigkeiten
- primär unterdrückter Wert kann leicht berechnet werden
- $\longrightarrow$ weitere Sperrungen sind notwendig!

Anonymisieren von Tabellen - Sekundärsperrungen

	1	2	3	sum
1	20	50	10	80
2	NA	19	NA	49
3	NA	32	NA	61
sum	45	101	44	190

Anonymisieren von Tabellen - Sekundärsperrungen

	1	2	3	sum
1	20	50	10	80
2	NA	NA	22	49
3	NA	NA	12	61
sum	NA	101	NA	190

Anonymisieren von Tabellen - Sekundärsperrungen

	1	2	3	sum
1	NA	50	10	NA
2	NA	19	22	NA
3	17	32	12	61
sum	45	101	44	190

Anonymisieren von Tabellen - Sekundärsperrungen

	1	2	3	sum
1	NA	50	10	NA
2	NA	19	22	NA
3	17	32	12	61
sum	45	101	44	190

- welches Sperrmuster ist optimal?
- gibt es überhaupt optimale Sperrmuster?
- **Beurteilungskriterien:**
  - minimale Anzahl an zusätzlich unterdrückten Zellen
  - minimale Summe der zusätzlich gesperrten Zellwerte

Anonymisieren von Tabellen - Sekundärsperrungen

STATISTIK AUSTRIA
Die Informationsmanager

- Problem der möglichst optimalen Sekundärsperrung wird NP-hard im Falle von hierarchischen (und/oder verlinkten) Tabellen
- statt Werte zu sperren kann man Zellen auch (geschickt) Runden um ausreichende Geheimhaltung zu erreichen (mehr dazu in ST35!)
- die besten Methoden (aber auch die komplexesten) basieren auf **linearer Optimierung**
- **Ziel**: verändere die Tabelle sowenig als möglich (durch Sperrung, Runden, etc.) unter der Nebenbedingung, dass Werte von primär gesperrten Zellen nicht in einem bestimmten Intervall vorausgesagt werden können (mehr dazu in ST35!)
- **Hinweis**: Sekundärunterdrückung ist eine (versteckte) Form von Intervallpublikation!

Anonymisieren von Tabellen - Intervallpublikation

	1	2	3	sum
1	NA	50	NA	80
2	NA	19	NA	49
3	17	32	12	61
sum	45	101	44	190

Anonymisieren von Tabellen - Intervallpublikation

	1	2	3	sum
1	[0,28]	50	[2,30]	80
2	[0,28]	19	[2,30]	49
3	17	32	12	61
sum	45	101	44	190

- was ist ein ausreichendes Schutzintervall?
- oft werden dafür Prozentwerte des originalen Zellwertes herangezogen
- reicht Schutz gegen exakte Rückrechenbarkeit (Diskussion?)

Anonymisieren von Tabellen - hierarchische Tabellen

	A	B	C	Total
6211	20	50	10	80
6212	8	19	22	49
6214	17	32	12	61
621	45	101	44	190

	A	B	C	Total
6222	40	50	20	110
6223	2	20	18	40
6224	20	30	25	75
622	62	100	53	225

Anonymisieren von Tabellen - hierarchische Tabellen

	A	B	C	Total
6211	20	50	10	80
6212	8	19	22	49
6214	17	32	12	61
621	45	101	44	190

	A	B	C	Total
6222	40	50	20	110
6223	S	20	S	40
6224	20	30	25	75
622	S	100	S	225

Anonymisieren von Tabellen - hierarchische Tabellen

	A	B	C	Total
6211	20	50	10	80
6212	8	19	22	49
6214	17	32	12	61
621	45	101	44	190

	A	B	C	Total
6222	40	50	20	110
6223	S	20	S	40
6224	20	30	25	75
622	S	100	S	225

	A	B	C	Total
621	45	101	44	190
622	62	100	53	225
62	107	201	97	415

Anonymisieren von Tabellen - hierarchische Tabellen

- wird Sekundärunterdückung durchgeführt, muss immer die gesamte, publizierte Hierarchiestruktur berücksichtigt werden
- je komplexer (verschachtelter) die Hierarchie ist, ...
  - desto mehr Sekundärsperrungen sind notwendig
  - desto mehr lineare Abhängigkeiten besitzt das Problem
  - desto rechenintensiver ist es, das Unterdrückungsproblem zu lösen
- analoges gilt auch für verlinkte Tabellen

Geschafft

STATISTIK AUSTRIA
Die Informationsmanager

Geschafft! :)