Hierarchische Clusteranalyse

Daniel Stepanovic

Ausgangssituation

Ein Hardware-Onlineshop bietet seinen Kund:innen bei höherpreisigen Produkten die Möglichkeit, eine Ratenzahlung zu vereinbaren. Dabei kann die Laufzeit der Ratenzahlung flexibel gewählt werden. Um das Angebot kundenfreundlicher zu gestalten, soll analysiert werden, ob typische Gruppen (Cluster) von Kund:innen anhand ihrer gewählten Kreditlaufzeit und -höhe existieren.

Datenmanagement

Die Daten wurden mit `read.table()` eingelesen:

```
rohdaten = read.table("wi23b095.txt", sep = ";", header = TRUE)
head(rohdaten)
```

```
  laufzeit hoehe
1        7  2576
2       48 10297
3       48  6110
4       24  3552
5       48  6758
6        6  1343
```

Wir extrahieren die interessierenden Variablen **laufzeit** und **hoehe** und überprüfen, ob es Beobachtungseinheiten mit fehlenden Werten gibt. Außerdem bestimmen wir die Stichprobengröße:

```r
summary(rohdaten)
```

```
    laufzeit        hoehe
 Min.   : 6.0   Min.   :  708
 1st Qu.:12.0   1st Qu.: 1416
 Median :24.0   Median : 2348
 Mean   :23.5   Mean   : 3857
 3rd Qu.:31.5   3rd Qu.: 5195
 Max.   :48.0   Max.   :14179
                NA's   :1
```

```r
nrow(rohdaten)
```

```
[1] 36
```

Die Zusammenfassung zeigt einen fehlenden Wert in der Spalte hoehe. Insgesamt enthält der Datensatz 36 Beobachtungen, eine davon unvollständig.

```r
daten = na.omit(rohdaten)
nrow(daten)
```

```
[1] 35
```

Der bereinigte Datensatz umfasst nun 35 Beobachtungen.

Standardisierung

Um die unterschiedlichen Wertebereiche der zwei metrischen Variablen auszugleichen, werden diese standardisiert:

```r
daten.s = scale(daten)
```

Distanzmatrix

Im nächsten Schritt berechnen wir die Distanzmatrix, die als Input für den eigentlichen Clusteralgorithmus übergeben wird. Da wir nur metrische Variablen verwenden, können wir das voreingestellte Distanzmaß (Euklidische Distanz) beibehalten:

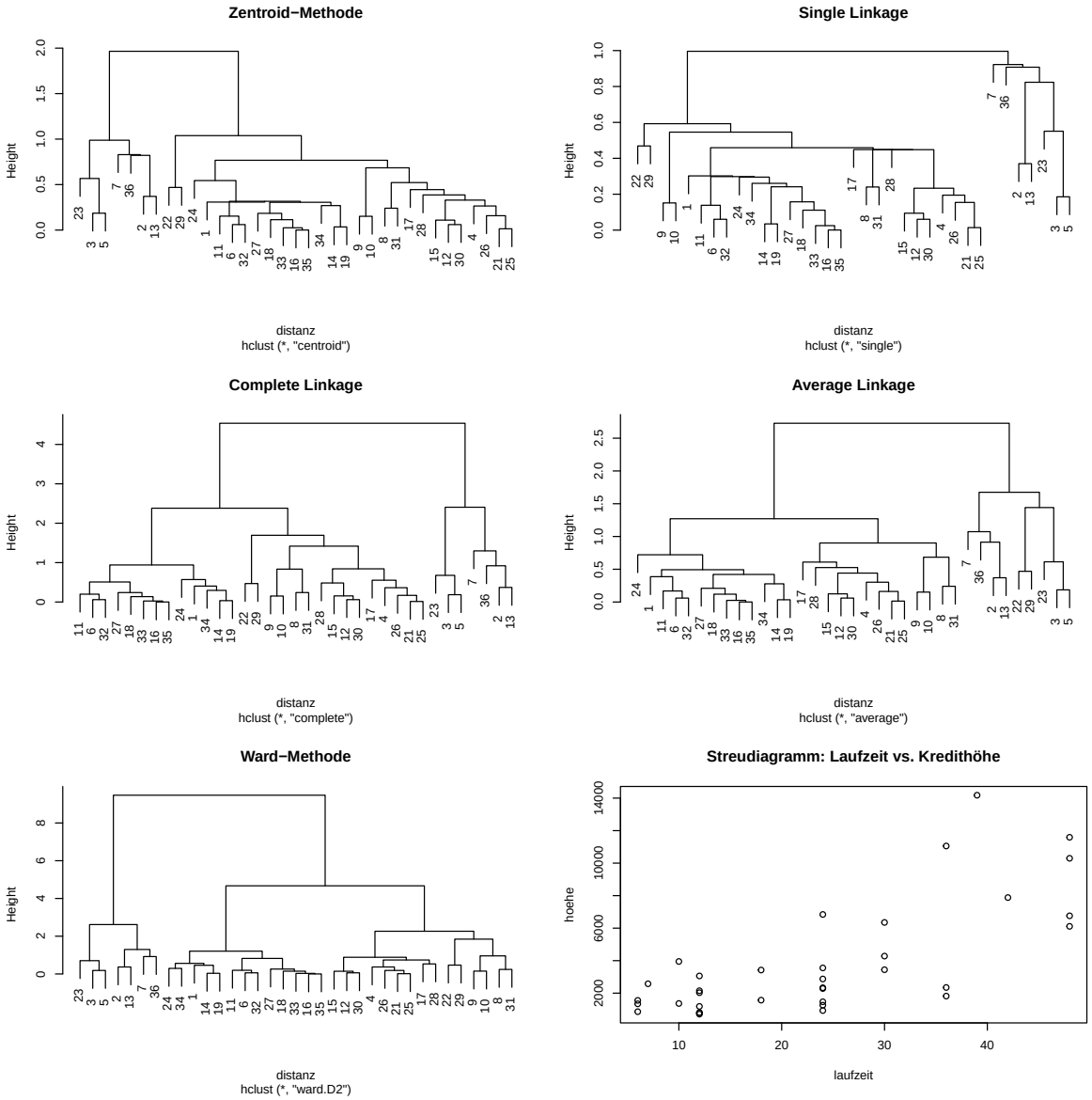
```
distanz = dist(daten.s)
```

Durchführung der hierarchischen Clusteranalyse

Als Clustermethode verwenden wir agglomeratives hierarchisches Clustering. Wir verwenden fünf verschiedene Methoden (Zentroid-Methode, Single Linkage, Complete Linkage, Average Linkage und Ward), deren Ergebnisse wir danach vergleichen wollen:

```
hc.centroid <- hclust(distanz, method = "centroid")
hc.single   <- hclust(distanz, method = "single")
hc.complete <- hclust(distanz, method = "complete")
hc.average  <- hclust(distanz, method = "average")
hc.ward     <- hclust(distanz, method = "ward.D2")
```

Wir visualisieren die Fusionierungsschritte für alle fünf Methoden mit Dendrogrammen und stellen diese dem Streudiagramm gegenüber:

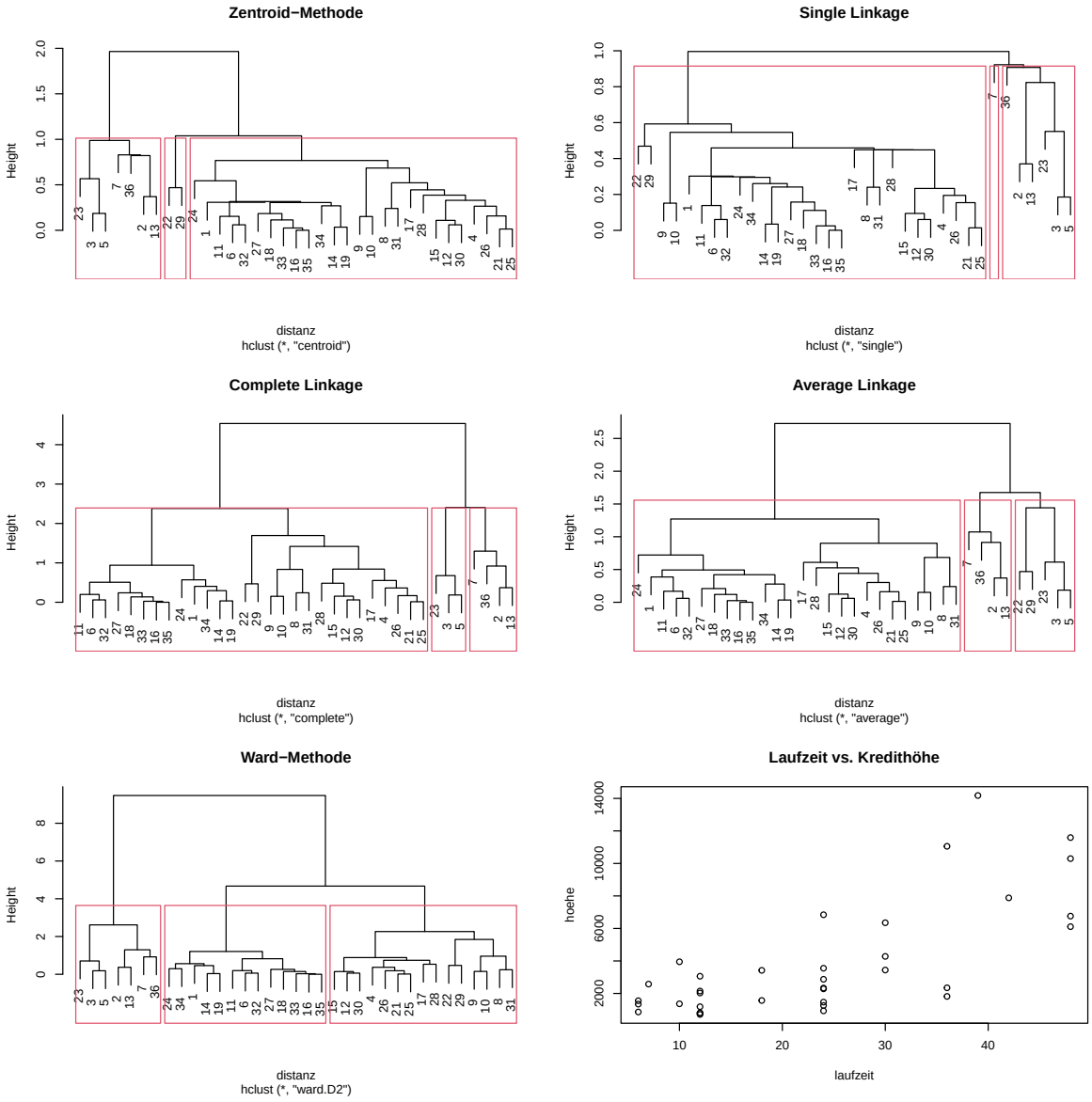


Die fünf Dendrogramme zeigen die Ergebnisse hierarchischer Clusteranalysen mit unterschiedlichen Fusionskriterien. Rechts unten ist ein Streudiagramm dargestellt, das den Zusammenhang zwischen Laufzeit (x-Achse) und Kreditbetrag (y-Achse) visualisiert.

- Die **Zentroid-Methode** zeigt einige markante Fusionen mit deutlichem Anstieg der Heterogenität. Besonders die Beobachtungen 3, 5 und 23 bleiben lange isoliert und werden erst sehr spät mit anderen Gruppen verschmolzen. Die Clusterstruktur wirkt insgesamt unausgewogen. Die Methode ist außerdem anfällig für Inversionen, was die Interpretation zusätzlich erschwert. Eine 3-Cluster-Lösung erscheint anhand des Dendrogramms am sinnvollsten, da sie die Ausreißergruppe (3, 5, 23) und (22, 29) klar von den übrigen Beobachtungen trennt und gleichzeitig eine interpretable Struktur erhält.
- Die **Single-Linkage-Methode** zeigt das typische Kettenverhalten: Viele Beobachtungen werden nacheinander verbunden, was auf eine geringe Trennschärfe hindeutet. Auffällige Fälle wie 3, 5 und 23 fusionieren sehr spät, was auf mögliche Ausreißer hinweist. Die Methode liefert keine klar abgegrenzten Gruppen, weshalb eine sinnvolle Clusterlösung nur schwer erkennbar ist. Eine 3-Cluster-Lösung wäre möglich.
- Die **Complete-Linkage-Methode** zeigt eine stabile und klar strukturierte Clusterlösung. Die Beobachtungen gruppieren sich auf ähnlicher Höhe, was für eine ausgewogene Struktur spricht. Auffällige Fälle wie 5, 23 und 3 bleiben lange isoliert. Eine 3-Cluster-Lösung erscheint hier am sinnvollsten, da sie die Hauptgruppen gut voneinander trennt und zugleich kompakte Cluster bildet.
- Die **Average-Linkage-Methode** zeigt eine klare, aber weichere Struktur als Complete Linkage. Auch hier bleiben Beobachtungen wie 3, 5 und 23 lange eigenständig. Eine 3-Cluster-Lösung ist sinnvoll, da sich drei Hauptgruppen deutlich abgrenzen lassen, ohne die Struktur zu stark aufzubrechen.
- Die **Ward-Methode** zeigt eine klar strukturierte Clusterlösung mit deutlichem Anstieg der Heterogenität bei der Reduktion auf zwei Cluster. Ein Schnitt bei ca. Höhe 5 liefert eine sinnvolle 3-Cluster-Lösung: Eine Gruppe mit hohen Kreditbeträgen und langen Laufzeiten (z.B. 3, 5, 23), eine heterogene Mittelgruppe, und ein dritter Cluster mit kürzeren Laufzeiten und geringeren Kreditbeträgen. Die Gruppen sind gut getrennt und inhaltlich interpretierbar.

Das begleitende Streudiagramm der beiden Variablen laufzeit und hoehe zeigt bereits eine sichtbare Gruppierung in drei Bereiche: kurze Laufzeit mit niedriger Kredithöhe, mittlere Laufzeit mit mittlerem Betrag und lange Laufzeit mit hohen Kreditbeträgen. Besonders Beobachtungen mit hohen Werten in beiden Dimensionen heben sich klar ab und stützen die Entscheidung für eine 3-Cluster-Lösung.

Clustervisualisierung mit Rechtecken (k = 3)



Vier der Methoden ergeben weitgehend ähnliche 3-Cluster-Lösungen. Besonders die Beobachtungen 3, 5 und 23 bilden in allen Verfahren eine klar abgegrenzte Gruppe. Single Linkage weicht ab: Aufgrund des Kettenverhaltens entstehen unscharfe Cluster. Die Zentroid-Methode ist weniger zuverlässig, da sie anfällig für Inversionen ist. Die **Ward-Methode** liefert die stabilste und am besten interpretierbare Struktur.

Agglomerative Koeffizienten

```
library(cluster)
methoden = c("single", "complete", "average", "ward.D2")
sapply(methoden, function(m) coef(hclust(distanz, method = m)))
```

```
   single  complete   average   ward.D2
0.7652802 0.9391793 0.9014946 0.9723563
```

Ward und Complete Linkage liefern hier die besten Ergebnisse (ACs von 0.97 bzw. ~0.94), auch die Performance von Average Linkage ist mit einem Wert von 0.90 noch sehr gut. Single Linkage fällt mit einem Koeffizienten von ~0.77 deutlich ab. Wir entscheiden uns daher für die Verwendung der Ward-Methode. Die daraus resultierende Clusterlösung wird anschließend als neue Variable clust dem ursprünglichen Datensatz hinzugefügt.

```
clusterzuordnung = cutree(hc.ward, k = 3)
daten$clust = clusterzuordnung
```

Interpretation der Clusterlösung

Wir sehen uns für jeden der drei Cluster die Zentroide an, um die Lösung inhaltlich zu interpretieren. Dabei betrachten wir nur die beiden Variablen, die im Clustering berücksichtigt wurden: **laufzeit** und **hoehe**.

```
aggregate(cbind(laufzeit, hoehe) ~ clust, data = daten, mean)
```

```
  clust  laufzeit    hoehe
1     1  9.923077 1720.538
2     2 44.142857 9695.714
3     3 26.000000 2984.467
```

Interpretation:

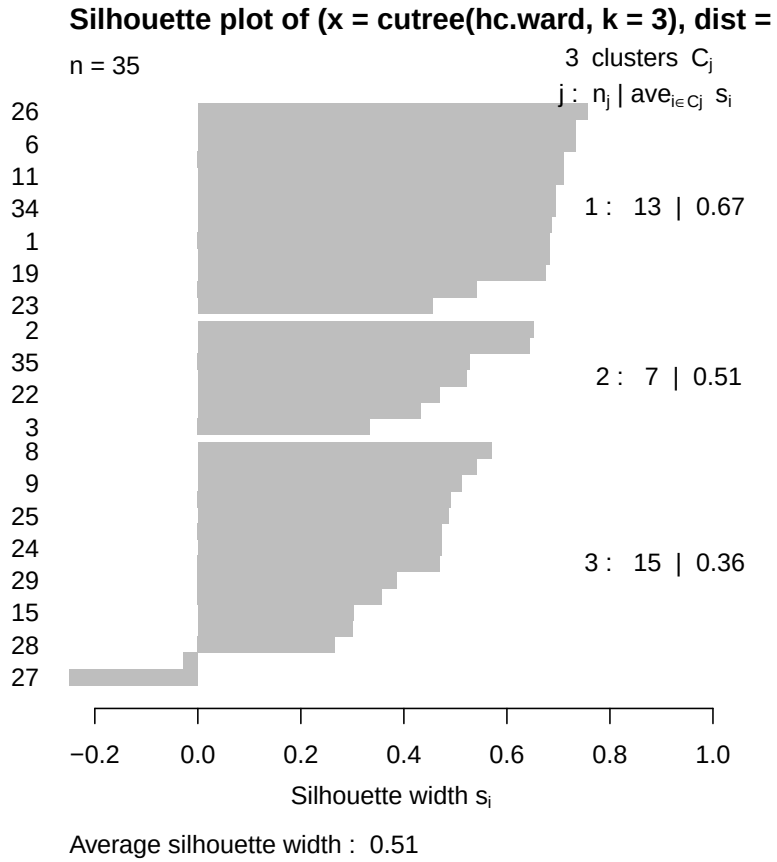
Cluster 1 umfasst Kund:innen mit sehr kurzen Laufzeiten (Ø 9,9 Monate) und niedrigen Kreditbeträgen (Ø 1.720 €). Diese Gruppe steht vermutlich für kurzfristige, kleinere Finanzierungen

Cluster 2 zeigt mit Abstand die höchsten Durchschnittswert, lange Laufzeiten (Ø 44,1 Monate) und hohe Kredithöhen (Ø 9.696 €). Diese Gruppe könnte für hochpreisige Produkte mit langfristiger Finanzierung stehen

Cluster 3 liegt im mittleren Bereich mit durchschnittlicher Laufzeit von 26 Monaten und Kredithöhen von ca. 2.984 €. Hier handelt es sich vermutlich um Standardkredite für mittelgroße Investitionen mit mittlerer Laufzeit.

Abschließend überprüfen wir die Qualität der Clusterlösung mithilfe eines Silhouettenplots:

```
library(cluster)
plot(silhouette(cutree(hc.ward, k = 3), distanz))
```



- Insgesamt beträgt der durchschnittliche Silhouettenkoeffizient 0.51, die Clusterlösung weist somit eine mittelstarke Struktur auf.
- Cluster 1 hat mit 0.67 den höchsten Durchschnittswert, das spricht für eine klar abgegrenzte und kompakte Gruppe. Die Zuordnungen in diesem Cluster sind weitgehend überzeugend.
- Cluster 2 hat mit 0.51 ebenfalls eine akzeptable Struktur. Die Silhouettenhöhen sind hier etwas variabler, aber es gibt keine stark negativen Ausreißer.

- Cluster 3 zeigt mit 0.36 den schwächsten Wert, die Struktur ist hier deutlich weniger kompakt, einige Beobachtungen (z.B. Nr. 27) liegen nahe an der Clustergrenze bzw. wurden nur schwach zugeordnet.
- Es gibt eine leicht negative Silhouette (Beobachtung 27), was auf eine mögliche Fehlzuordnung hinweist, aber insgesamt bleibt die Qualität der Lösung noch akzeptabel.