

Einfluss der Wettbewerbsstruktur auf Gehaltsniveaus im Data Science-Bereich:

Eine Social Network Analyse

Contents

1	Einleitung	2
1.1	Requirements	2
1.2	Motivation und Zielsetzung	2
1.3	Forschungsfrage	3
1.4	Datengrundlage	3
1.4.1	CSV einlesen	3
1.4.2	Erste Ansicht der Daten	4
2	Analysestrategie	8
3	Analyse	9
3.1	Datenbereinigung	9
3.1.1	Bereinigung für die geografische Analyse	9
3.1.2	Überprüfung auf weitere fehlende Werte	10
3.1.3	Entfernen irrelevanter Spalten	11
3.2	Netzwerkbildung und Visualisierung	12
3.2.1	Geografische Vorbetrachtung	12
3.3	Wettbewerbsnetzwerk	16
3.3.1	Bloß direkte Wettbewerber	19
3.4	Zentralitätsanalyse innerhalb des Netzwerkes	24
3.4.1	Betweenness-Zentralität	24
3.4.2	Degree-Zentralität	25
3.4.3	Eigenvector-Zentralität	26
3.5	Cluster-Analyse	26
3.6	Ergänzung zu den Zentralitätsanalysen	27
4	Conclusion	30
5	Literaturverzeichnis	31

1 Einleitung

1.1 Requirements

Zunächst müssen die benötigten Bibliotheken installiert werden:

- `$ install.packages("tidyverse")`
- `$ install.packages("igraph")`
- `$ install.packages("visNetwork")`
- `$ install.packages("dplyr")`
- `$ install.packages("tidyr")`
- `$ install.packages("kableExtra")`
- `$ install.packages("webshot")`
- `$ install.packages("knitr")`
- `$ install.packages("ggplot2")`
- `$ install.packages("RColorBrewer")`

Und anschließend geladen werden:

```
library(tidyverse)
library(igraph)
library(visNetwork)
library(dplyr)
library(tidyr)
library(knitr)
library(kableExtra)
library(webshot)
library(ggplot2)
library(RColorBrewer)
```

1.2 Motivation und Zielsetzung

In ihrem Artikel “Data Scientist: The Sexiest Job of the 21st Century” betonen Davenport und Patil, dass Data Scientists durch ihre Fähigkeiten in Informatik, Statistik und ihr Fachwissen allgemein einen erheblichen Mehrwert für Unternehmen schaffen.¹ Die Fähigkeit, aus komplexen, unstrukturierten Daten wertvolle Erkenntnisse zu gewinnen, macht Data Scientists in vielen Branchen zu einer unverzichtbaren Ressource.² Die Nutzung ihrer Kompetenzen verschafft Unternehmen einen Wettbewerbsvorteil, da sie datengetriebene Entscheidungen, Produktinnovationen und Effizienzsteigerungen ermöglicht.³

Darüber ob Data Scientists immer noch the “Sexiest Job” des 21. Jahrhunderts sind, lässt sich streiten. Fakt ist jedoch, dass die Nachfrage nach Data Scientists in den letzten Jahren stark gestiegen ist und voraussichtlich immer weiter steigen wird. Dieser Trend ist auch in den Google-Suchanfragen zu den Begriffen erkenntlich:⁴

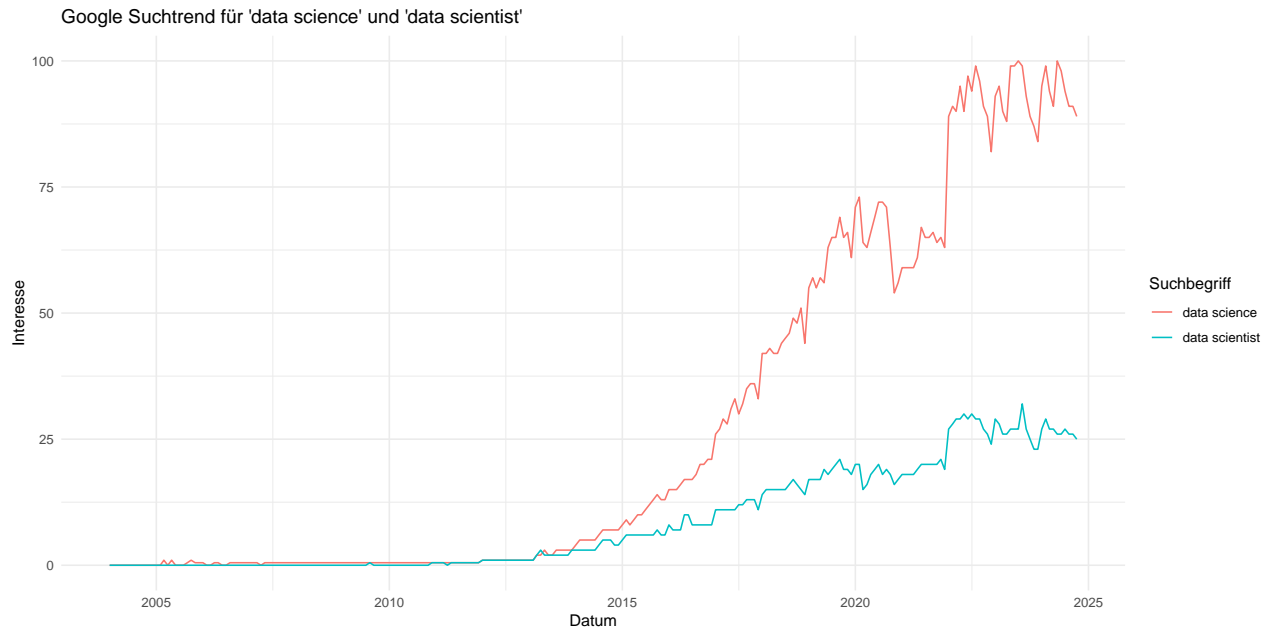
```
ggplot(data, aes(x = Monat)) +
  geom_line(aes(y = `data science`, color = "data science")) +
  geom_line(aes(y = `data scientist`, color = "data scientist")) +
  labs(title = "Google Suchtrend für 'data science' und 'data scientist'",
        x = "Datum",
        y = "Interesse",
        color = "Suchbegriff") +
  theme_minimal()
```

¹Davenport, Patil 2012

²Davenport, Patil 2012

³Davenport, Patil 2012

⁴Google Trends, abgerufen am 30.10.2024



Das wachsende Interesse an Data Science stellt eine große Chance für Arbeitnehmer dar. Ziel dieser Arbeit ist es einen Überblick über den Data-Science-Jobmarkt zu geben, um Arbeitnehmern bei der Jobsuche zu helfen und andererseits einen Überblick über die Gehälter und die Rolle von Geographie und Wettbewerb bei Jobangeboten und Gehältern zu geben.

1.3 Forschungsfrage

Im Rahmen der vorliegenden Arbeit wird die folgende Forschungsfrage bearbeitet:

Inwiefern beeinflusst die geografische Nähe von Unternehmen das Gehaltsniveau und die Verfügbarkeit von Data-Science-Jobs? Lässt sich eine signifikante Variation der Einkommen innerhalb regionaler Cluster feststellen, und wie kann diese durch Netzwerkzentralität erklärt werden?

Zur Beantwortung dieser Forschungsfrage soll zudem analysiert werden, inwiefern das Wettbewerbsumfeld zwischen Unternehmen die Gehaltsstruktur im Bereich Data Science beeinflusst und welche Rolle zentrale Unternehmen bei der Bestimmung des Gehaltsniveaus spielen.

1.4 Datengrundlage

Nachdem die Daten in Python extern als Vorbereitung aufbereitet wurden, kann nun die Datengrundlage für diese Arbeit in R eingelesen werden. Dabei wurde sich an <https://www.kaggle.com/code/fahadrehman07/data-science-job-salary-prediction-glassdoor> orientiert.

1.4.1 CSV einlesen

```
data <- read_csv("data/Glassdoor_DataScience_Salary.csv")

## Rows: 742 Columns: 28
## -- Column specification -----
## Delimiter: ","
## chr (14): Job Title, Job Description, Company Name, Location, Headquarters, ...
## dbl (14): Salary Estimate, Rating, Founded, Min_Salary, Max_Salary, Same Sta...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Die vorliegende Arbeit basiert auf einem Datensatz von Kaggle, der Informationen über Data Science Jobs in verschiedenen Unternehmen für den US-amerikanischen Markt enthält. Der Datensatz umfasst 742 Zeilen und 28 Spalten, was auf eine Anzahl von 742 verschiedenen Jobangeboten hindeutet. Diese Anzahl ist kann für die Zwecke dieser Arbeit als ausreichend zu betrachten, auch wenn eine höhere Zahl an Beobachtungen möglicherweise zu präziseren Schlussfolgerungen geführt hätte.

Der Datensatz beruht auf Daten, die von Glassdoor extrahiert wurden, eine für Stellenanzeigen und Unternehmensbewertung bekannte Website, und bietet detaillierte Informationen über Data-Science-Jobs sowie deren Gehälter. Der Datensatz beinhaltet wesentliche Informationen, darunter Jobtitel, geschätzte Gehälter, Stellenbeschreibungen, Unternehmensbewertungen sowie relevante Unternehmensdaten wie Standort, Größe und Branche. Eine detaillierte Beschreibung dieser Daten erfolgt im späteren Verlauf. Der Datensatz eignet sich in besonderem Maße für den Zweck dieser Arbeit, aber auch für Analysen des Arbeitsmarktes, beispielsweise zur Untersuchung von Gehaltstrends oder zur Identifizierung der am besten bewerteten Unternehmen.

Der Datensatz umfasst konkret die folgenden Spalten:

1.4.2 Erste Ansicht der Daten

```
head(data, 5)

## # A tibble: 5 x 28
##   `Job Title` `Salary Estimate` `Job Description` Rating `Company Name` Location
##   <chr>          <dbl> <chr>          <dbl> <chr>          <chr>
## 1 Data Scien~          72  "Data Scientist\~    3.8 Tecolote Rese~ Albuquerque~
## 2 Healthcare~         87.5  "What You Will D~    3.4 University of~ Linthic~
## 3 Data Scien~          85  "KnowBe4, Inc. i~    4.8 KnowBe4      Clearwa~
## 4 Data Scien~         76.5  "*Organization a~    3.8 PNNL          Richlan~
## 5 Data Scien~        114.  "Data Scientist\~    2.9 Affinity Solu~ New Yor~
## # i 22 more variables: Headquarters <chr>, Size <chr>, Founded <dbl>,
## #   `Type of ownership` <chr>, Industry <chr>, Sector <chr>, Revenue <chr>,
## #   Competitors <chr>, Min_Salary <dbl>, Max_Salary <dbl>, State <chr>,
## #   `Same State` <dbl>, Age <dbl>, Python_yn <dbl>, `R Studio` <dbl>,
## #   Spark <dbl>, AWS_yn <dbl>, Excel_yn <dbl>, Job_simp <chr>, job_state <chr>,
## #   desc_len <dbl>, Num_comp <dbl>
```

```
spec(data)

## cols(
##   `Job Title` = col_character(),
##   `Salary Estimate` = col_double(),
##   `Job Description` = col_character(),
##   Rating = col_double(),
##   `Company Name` = col_character(),
##   Location = col_character(),
##   Headquarters = col_character(),
##   Size = col_character(),
##   Founded = col_double(),
##   `Type of ownership` = col_character(),
##   Industry = col_character(),
##   Sector = col_character(),
##   Revenue = col_character(),
##   Competitors = col_character(),
##   Min_Salary = col_double(),
##   Max_Salary = col_double(),
##   State = col_character(),
##   `Same State` = col_double(),
```

```
## Age = col_double(),
## Python_yn = col_double(),
## `R Studio` = col_double(),
## Spark = col_double(),
## AWS_yn = col_double(),
## Excel_yn = col_double(),
## Job_simp = col_character(),
## job_state = col_character(),
## desc_len = col_double(),
## Num_comp = col_double()
## )
```

```
summary(data)
```

```
## Job Title      Salary Estimate Job Description      Rating
## Length:742     Min.   : 13.5   Length:742         Min.   :-1.000
## Class :character 1st Qu.: 73.5   Class :character   1st Qu.: 3.300
## Mode  :character Median : 97.5   Mode  :character   Median : 3.700
##                  Mean   :100.6                Mean   : 3.619
##                  3rd Qu.:122.5                3rd Qu.: 4.000
##                  Max.    :254.0                Max.    : 5.000
## Company Name     Location      Headquarters      Size
## Length:742       Length:742     Length:742       Length:742
## Class :character Class :character Class :character Class :character
## Mode  :character Mode  :character Mode  :character Mode  :character
##
##
##
## Founded         Type of ownership  Industry          Sector
## Min.   : -1     Length:742       Length:742       Length:742
## 1st Qu.:1939    Class :character Class :character Class :character
## Median :1988    Mode  :character Mode  :character Mode  :character
## Mean   :1837
## 3rd Qu.:2007
## Max.    :2019
## Revenue         Competitors      Min_Salary        Max_Salary
## Length:742       Length:742       Min.   : 15.00    Min.   : 16.0
## Class :character Class :character 1st Qu.: 52.00    1st Qu.: 96.0
## Mode  :character Mode  :character Median : 69.50    Median :124.0
##                  Mean   : 74.72    Mean   :127.2
##                  3rd Qu.: 91.00    3rd Qu.:155.0
##                  Max.    :202.00    Max.    :306.0
## State           Same State      Age              Python_yn
## Length:742       Min.   :0.000    Min.   : -1.00    Min.   :0.0000
## Class :character 1st Qu.:0.000    1st Qu.: 14.00    1st Qu.:0.0000
## Mode  :character Median :1.000    Median : 27.00    Median :1.0000
##                  Mean   :0.558    Mean   : 49.39    Mean   :0.5283
##                  3rd Qu.:1.000    3rd Qu.: 62.00    3rd Qu.:1.0000
##                  Max.    :1.000    Max.    :279.00    Max.    :1.0000
## R Studio        Spark          AWS_yn            Excel_yn
## Min.   :0.000000 Min.   :0.0000    Min.   :0.0000    Min.   :0.0000
## 1st Qu.:0.000000 1st Qu.:0.0000    1st Qu.:0.0000    1st Qu.:0.0000
## Median :0.000000 Median :0.0000    Median :0.0000    Median :1.0000
## Mean   :0.002695 Mean   :0.2251    Mean   :0.2372    Mean   :0.5229
## 3rd Qu.:0.000000 3rd Qu.:0.0000    3rd Qu.:0.0000    3rd Qu.:1.0000
```

```
## Max.      :1.000000  Max.      :1.0000  Max.      :1.0000  Max.      :1.0000
## Job_simp   job_state      desc_len      Num_comp
## Length:742  Length:742      Min.       : 407  Min.       :0.000
## Class :character  Class :character  1st Qu.: 2801  1st Qu.:0.000
## Mode  :character  Mode  :character  Median  : 3731  Median :0.000
##                                     Mean   : 3870  Mean   :1.054
##                                     3rd Qu.: 4740  3rd Qu.:3.000
##                                     Max.    :10051  Max.    :4.000
```

Im Folgenden wird eine Übersicht der wesentlichen Spalten präsentiert:

- **Job Title:** Die Berufsbezeichnung, sie gibt Aufschluss über die Tätigkeit.
- **Salary Estimate:** Die geschätzte Gehalt, in tausend Dollar pro Jahr. Es basiert auf dem Durchschnitt von dem minimalen und maximalen Gehalt.
- **Job Description, Job_simp:** Die Beschreibung der Stelle, die Aufgaben und Anforderungen enthält. Auch die vereinfachte Version der Berufsbezeichnung.
- **Rating:** Die Bewertung des Unternehmens, sie weist eine Spannbreite von 1 bis 5 auf, wobei die Bewertung "1" bei jeder Spalte für fehlende Bewertungen steht.
- **Company Name, Location, Headquarters, Size, Founded:** Unternehmensbezogene Daten wie Name, Standort, Sitz, Größe und Gründungsjahr des Unternehmens.
- **Type of ownership, Industry, Sector, Revenue:** Weitere Unternehmensmerkmale, diese umfassen die Eigentumsart, die Branche, den Sektor sowie die Einnahmen.
- **Competitors:** Die Wettbewerber des Unternehmens, die im Zusammenhang dieser Arbeit von besonderer Bedeutung sind.
- **Skills (Python_yn, R Studio, Spark, AWS_yn, Excel_yn):** Spalten, aus denen hervorgeht, ob die betreffende Kompetenz in der Stellenbeschreibung verlangt wird (0 = nein, 1 = ja).
- **Min_salary, Max_salary:** Minimale und maximale Gehaltsschätzungen.
- **State, Same State, job_state, Age, desc_len, Num_comp:** Zusätzliche Informationen wie Standort der Stelle, Alter des Unternehmens, Länge der Stellenbeschreibung und Anzahl der Mitbewerber.

Es zeigt sich, dass eine Vielzahl von Spalten für die vorliegende Untersuchung irrelevant ist. Infolgedessen werden in einem späteren Teil der Arbeit irrelevante Spalten, wie beispielsweise die Kenntnisse in Python, R Studio, Spark und ähnlichen Programmen, welche ursprünglich aus der Jobbeschreibung extrahiert wurden, entfernt.

Nachdem die Daten in Python mit Hilfe von Pandas bereinigt, ergänzt und bearbeitet wurden, können sie nun in R eingelesen werden. Dabei wurde sich an <https://www.kaggle.com/code/maxzeitler/data-science-job-salary-prediction-glassdoor/edit> orientiert.

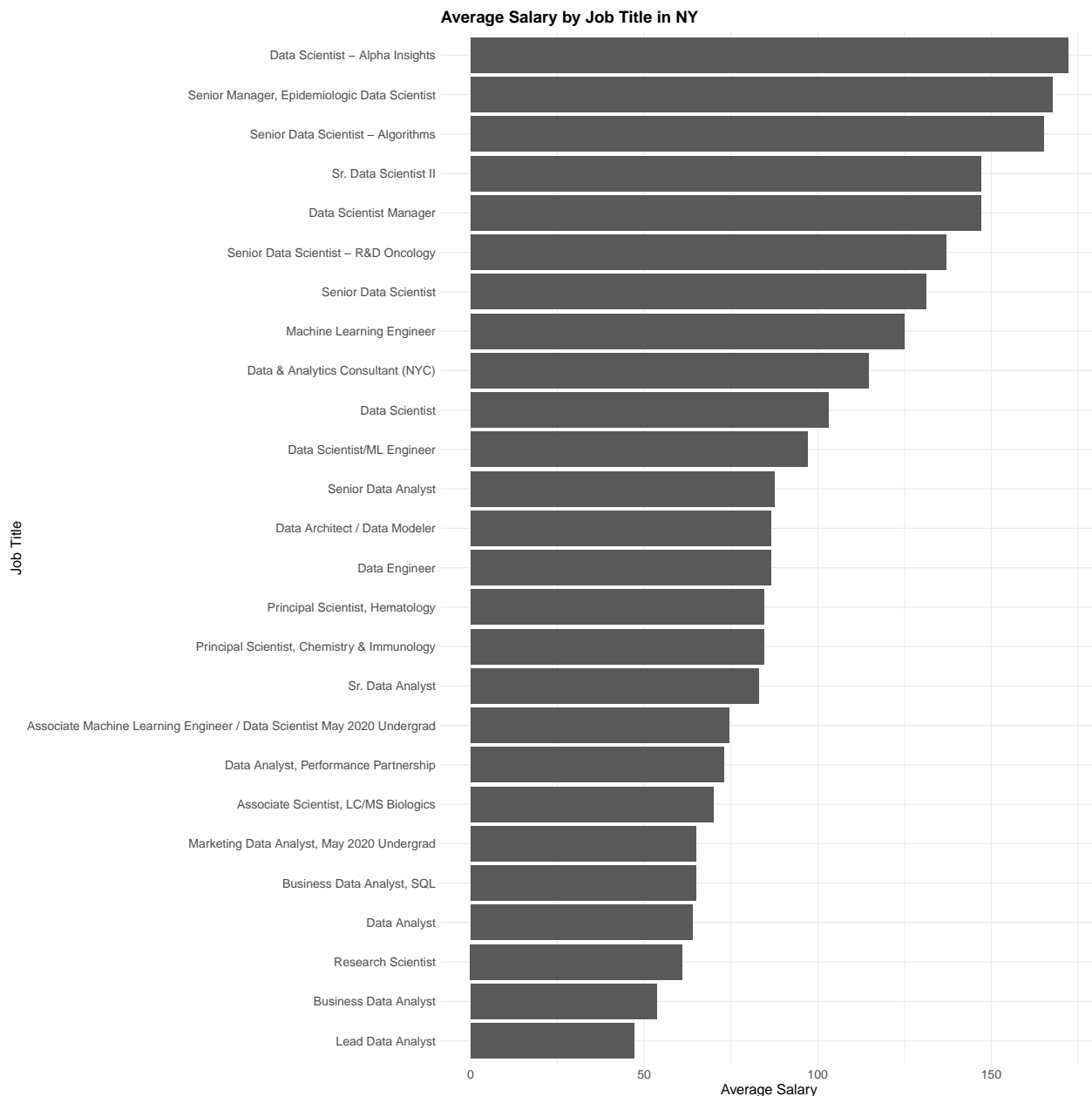
Im Folgenden wird eine erste Betrachtung der Daten vorgenommen. Zu diesem Zweck werden die Jobs in New York nach ihren jeweiligen Vergütungen geordnet und in Form eines Balkendiagramms dargestellt.

```
# Filterung der Daten für New York
data_ny <- data %>%
  filter(State == "NY")

# Durchschnittsgehalt nach Berufsbezeichnung
avg_salary_by_job_ny <- data_ny %>%
  group_by(`Job Title`) %>%
  summarise(Average_Salary = mean(`Salary Estimate`, na.rm = TRUE)) %>%
  arrange(desc(Average_Salary))

# Bar Plot
ggplot(avg_salary_by_job_ny,
  aes(x = reorder(`Job Title`, Average_Salary), y = Average_Salary)) +
  geom_bar(stat = "identity") +
  coord_flip() +
```

```
labs(title = "Average Salary by Job Title in NY",
     x = "Job Title",
     y = "Average Salary") +
theme_minimal() +
theme(
  axis.title = element_text(size = 14),
  axis.text = element_text(size = 12),
  plot.title = element_text(size = 16, face = "bold")
)
```



todo ... Insights aus dem Plot ziehen

Da die Datengrundlage nicht in einem igraph-Objekt vorliegt und ungerichtet ist, ist es notwendig Knoten, Kanten sowie relevante Attribute wie beispielsweise Gewichtungen zu definieren, um überhaupt Netzwerkvisualisierungen in R durchführen zu können. Doch dazu mehr im nächsten Kapitel.

2 Analysestrategie

1. Geografisches Netzwerk

Das Ziel besteht in der Erstellung eines Netzwerkes, welches auf der räumlichen Nähe von Unternehmen basiert. Auf diese Weise soll untersucht werden, inwiefern regional bedingte Faktoren die Gehälter beeinflussen. Die Bildung von Kanten erfolgt nach dem Kriterium der räumlichen Nähe. Dabei werden Unternehmen, die im gleichen Ort angesiedelt sind, durch Kanten verbunden.

2. Wettbewerbsnetzwerk

Die vorliegende Untersuchung zielt darauf ab, den Einfluss des Wettbewerbs auf die Gestaltung von Gehaltsstrukturen zu analysieren. Dazu werden die Beziehungen zwischen konkurrierenden Unternehmen als Netzwerk dargestellt. Die Bildung von Kanten durch Konkurrenzen erfolgt wie folgt: Die in der Spalte "Competitors" gelisteten Unternehmen werden als Knoten verbunden. In Bezug auf die Gewichtung sind verschiedene Optionen denkbar. Beispielsweise könnte die direkte Konkurrenz mit dem Wert "1" und die indirekte Konkurrenz mit dem Wert "0,5" bewertet werden. Dabei würde die indirekte Konkurrenz eine Branche umfassen, in der das Unternehmen zwar nicht als direkter Konkurrent aufgeführt ist, jedoch potenziell in Konkurrenz stehen könnte. Im Rahmen der Netzwerkmetriken erfolgt eine Analyse der folgenden Aspekte: Im Rahmen der Analyse von hierarchischen Beziehungen und unterschiedlichen Zentralitäten erfolgt eine Untersuchung der Wichtigkeit eines Unternehmens im Wettbewerbsnetzwerk sowie der Gehaltshöhen in Relation zur Konkurrenz.

3. Vergleich der Gehälter innerhalb der Netzwerke

Im Rahmen der Analyse werden die Gehälter innerhalb der beiden Netzwerke miteinander verglichen. Ziel ist die Identifikation von Unternehmen, die zentral in einem der beiden Netzwerke liegen, und solchen, die am Rand oder isoliert sind, um festzustellen, ob die zentralen Unternehmen höhere Gehälter anbieten. Zur Durchführung des Gehaltsvergleichs werden Korrelationen zwischen dem Gehalt und verschiedenen Zentralitätsmaßen innerhalb der geografischen und wettbewerbsbezogenen Netzwerke herangezogen. Darüber hinaus werden Cluster-Analysen durchgeführt, um Unternehmen, die geografisch und wettbewerbsbedingt vernetzt sind, miteinander zu vergleichen.

4. Zusammenführung und Vergleich der Netzwerke

Im Rahmen der Zusammenführung und des Vergleichs der Netzwerke erfolgt eine Gegenüberstellung der jeweiligen Strukturen, um etwaige Gemeinsamkeiten und Unterschiede zu identifizieren. Das Ziel dieser Untersuchung besteht in der Analyse der Interaktion beider Netzwerke sowie der Identifikation von Regionen, in denen eine besonders hohe Gehaltskonkurrenz zu beobachten ist. Im Rahmen des Vergleichs der Netzwerke hinsichtlich der Gehälter und des Wettbewerbs erfolgt zunächst eine Gegenüberstellung der Gehaltsverteilung in sogenannten "Hotspot-Regionen" und geografisch isolierten Regionen. Darüber hinaus werden gemeinsame Unternehmen in beiden Netzwerken sowie die Gehaltsstrukturen innerhalb der Überschneidungsbereiche analysiert.

3 Analyse

3.1 Datenbereinigung

3.1.1 Bereinigung für die geografische Analyse

Bei der Durchsicht des Datensatzes viel auf, dass die Spalten "Same State" und "job_state" von der Logik her ähnlich sind. Dies soll nun näher untersucht werden, um spätere Fehler vorzubeugen, vor allem bei den geografischen Netzwerken vorzubeugen.

```
# Auswahl der "State" und "job_state" Spalten
selected_data <- data %>%
  select(State, job_state)

# Heading der ausgewählten Spalten
head(selected_data, 15)
```

```
## # A tibble: 15 x 2
##   State job_state
##   <chr> <chr>
## 1 NM    NM
## 2 MD    MD
## 3 FL    FL
## 4 WA    WA
## 5 NY    NY
## 6 TX    TX
## 7 MD    MD
## 8 CA    CA
## 9 NY    NY
## 10 NY   NY
## 11 CA    CA
## 12 VA    VA
## 13 TX    TX
## 14 WA    WA
## 15 MA    MA
```

Sieht so aus, als wäre beide Spalten identisch. Dies soll jedoch zur Probe gestellt werden:

```
if (all(selected_data$State == selected_data$job_state, na.rm = TRUE)) {
  print("Alle Werte in 'State' und 'job_state' sind identisch.")
} else {
  print("Es gibt Unterschiede zwischen 'State' und 'job_state'.")
}
```

```
## [1] "Es gibt Unterschiede zwischen 'State' und 'job_state'."
```

Jedoch trügt der Schein, da es Unterschiede gibt.

```
# Auswahl der Zeilen, in denen "State" und "job_state" unterschiedlich sind
different_states <- selected_data %>%
  filter(State != job_state)

print(different_states, n = Inf)
```

```
## # A tibble: 1 x 2
##   State      job_state
##   <chr>      <chr>
## 1 Los Angeles CA
```

Es fällt auf, das LA und Los Angeles nicht einheitlich verwendet werden. Außerdem ist Los Angeles kein eigener Bundesstaat, sonder ein Teil von Kalifornien(CA). Dies soll nun korrigiert werden.

Außerdem sollte bei weieren Vorgehen beachtet werden, dass Werte wie “Na” oder “-1” vor den Analysen entfernt werden sollten.

```
# Ersetzen von "Los Angeles" durch "LA" und "LA" durch "CA"
data <- data %>%
  mutate(State = ifelse(State == "Los Angeles", "LA", State),
         job_state = ifelse(job_state == "Los Angeles", "LA", job_state))

data <- data %>%
  mutate(State = ifelse(State == "LA", "CA", State),
         job_state = ifelse(job_state == "LA", "CA", job_state))

# Erneute Überprüfung
selected_data <- data %>%
  select(State, job_state)

if (all(selected_data$State == selected_data$job_state, na.rm = TRUE)) {
  print("Alle Werte in 'State' und 'job_state' sind identisch.")
} else {
  print("Es gibt Unterschiede zwischen 'State' und 'job_state'.")
}
```

```
## [1] "Alle Werte in 'State' und 'job_state' sind identisch."
```

3.1.2 Überprüfung auf weitere fehlende Werte

```
# Überprüfen auf NA-Werte

na_counts <- colSums(is.na(data))
print("Anzahl der NA-Werte pro Spalte:")
```

```
## [1] "Anzahl der NA-Werte pro Spalte:"
```

```
print(na_counts)
```

```
##      Job Title  Salary Estimate  Job Description      Rating
##           0           0           0           0
## Company Name      Location  Headquarters      Size
##           0           0           0           0
##      Founded Type of ownership      Industry      Sector
##           0           0           0           0
##      Revenue    Competitors  Min_Salary  Max_Salary
##           0           0           0           0
##      State    Same State      Age  Python_yn
##           0           0           0           0
##      R Studio      Spark  AWS_yn  Excel_yn
##           0           0           0           0
##      Job_simp    job_state  desc_len  Num_comp
##           0           0           0           0
```

```
# Überprüfen auf -1-Werte
neg_one_counts <- sapply(data, function(x) sum(x == -1, na.rm = TRUE))
print("Anzahl der -1-Werte pro Spalte:")
```

```
## [1] "Anzahl der -1-Werte pro Spalte:"
```

```
print(neg_one_counts)
```

##	Job Title	Salary Estimate	Job Description	Rating
##	0	0	0	11
##	Company Name	Location	Headquarters	Size
##	0	0	1	1
##	Founded	Type of ownership	Industry	Sector
##	50	1	10	10
##	Revenue	Competitors	Min_Salary	Max_Salary
##	1	460	0	0
##	State	Same State	Age	Python_yn
##	0	0	50	0
##	R Studio	Spark	AWS_yn	Excel_yn
##	0	0	0	0
##	Job_simp	job_state	desc_len	Num_comp
##	0	0	0	0

Es zeigt sich, dass es keine NA-Werte gibt, jedoch einige -1-Werte, die entfernt werden sollten.

```
# Entfernen von Zeilen mit -1 Werten
```

```
data <- data %>%
```

```
  filter_all(all_vars(. != -1))
```

```
# Überprüfen auf -1-Werte nach Entfernung
```

```
neg_one_counts <- sapply(data, function(x) sum(x == -1, na.rm = TRUE))
```

```
print("Anzahl der -1 Werte pro Spalte:")
```

```
## [1] "Anzahl der -1 Werte pro Spalte:"
```

```
print(neg_one_counts)
```

##	Job Title	Salary Estimate	Job Description	Rating
##	0	0	0	0
##	Company Name	Location	Headquarters	Size
##	0	0	0	0
##	Founded	Type of ownership	Industry	Sector
##	0	0	0	0
##	Revenue	Competitors	Min_Salary	Max_Salary
##	0	0	0	0
##	State	Same State	Age	Python_yn
##	0	0	0	0
##	R Studio	Spark	AWS_yn	Excel_yn
##	0	0	0	0
##	Job_simp	job_state	desc_len	Num_comp
##	0	0	0	0

3.1.3 Entfernen irrelevanter Spalten

Basierend auf der Analysestrategie und den geplanten Analysen werden jetzt noch die Spalten, die nicht für die anfängliche geografische Analyse und die nachfolgende Wettbewerbsanalyse benötigt werden, entfernt.

```
# Entfernen irrelevanter Spalten
```

```
# Job Description, Rating, Headquarters, Size, Founded, Type of ownership, Sector, Revenue und Skills
```

```
data <- data %>%
```

```
  select(-c(`Job Description`, Rating, Headquarters, Size, Founded,
```

```

`Type of ownership`, Sector, Revenue,
Python_yn, `R Studio`, Spark, AWS_yn, Excel_yn))

# Ausgeben der noch enthaltenen Spalten
print(data %>% names())

## [1] "Job Title"          "Salary Estimate" "Company Name"    "Location"
## [5] "Industry"          "Competitors"     "Min_Salary"      "Max_Salary"
## [9] "State"             "Same State"      "Age"             "Job_simp"
## [13] "job_state"         "desc_len"        "Num_comp"

```

Nachdem die Bereinigung des Datensatzes abgeschlossen ist, kann mit der Analyse begonnen werden.

3.2 Netzwerkbildung und Visualisierung

3.2.1 Geografische Vorbetrachtung

Da bei der Betrachtung der Wettbewerbsstruktur die geografische Nähe von Unternehmen auch eine Rolle spielen kann, soll zunächst ein Netzwerk erstellt werden, das auf der geografischen Nähe von Unternehmen basiert. Diese Annahme beruht darauf, dass Unternehmen in derselben Region wahrscheinlich ähnliche Gehälter anbieten. Dies soll überprüft werden um diese Arbeit um eine weitere Dimension zu erweitern.

3.2.1.1 Erstellung eines Geografischen Netzwerkes Die Gewichtung erfolgt linear, wobei jeder Standort eine Grundgröße von 3 hat, und für jedes Unternehmen an diesem Standort wird die Größe um 0.5 erhöht. Ab einer Größe von 4.5 wird die Farbe des Standorts geändert, um die Standorte mit mehreren Unternehmen hervorzuheben.

```

# Aus Gründen der Sichtbarkeit, werden bloß Locations mit mehr als einem
# Unternehmen dargestellt.

# Extract relevant columns for geographic visualization
edges_geo <- data %>%
  select(Company = `Company Name`, Location = `Location`) %>%
  distinct()

# Calculate the number of companies per location and filter for locations
# with more than one company
location_counts <- edges_geo %>%
  group_by(Location) %>%
  summarise(Company_Count = n()) %>%
  filter(Company_Count > 1) # Keep only locations with more than one company

# Filter edges to include only connections for locations with more than
# one company
filtered_edges <- edges_geo %>%
  filter(Location %in% location_counts$Location)

# Create an igraph object for geographic visualization
network_geo <- graph_from_data_frame(filtered_edges, directed = FALSE)

# Set vertex colors based on whether the node is a company or a location
company_colors <- "blue"
location_colors <- rainbow(nrow(location_counts))

# Set vertex size based on the number of companies at each location

```

```

vertex_sizes <- ifelse(V(network_geo)$name %in% location_counts$Location,
                      3 + location_counts$Company_Count[
                        match(V(network_geo)$name, location_counts$Location)
                      ] * 0.5, # Linear scaling factor with minimum size 3
                      3) # Default size for companies

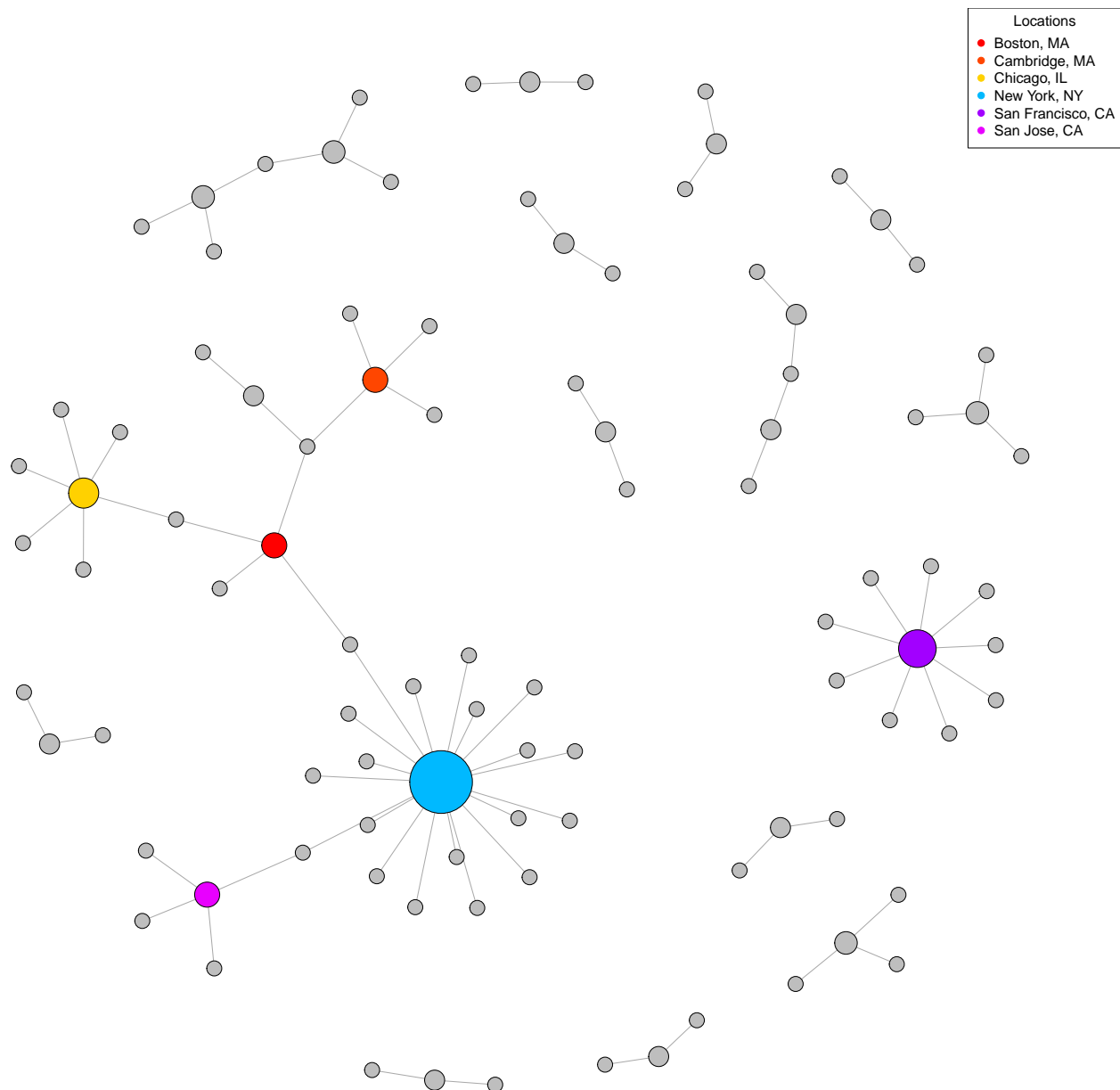
# Assign colors and sizes to vertices
V(network_geo)$size <- vertex_sizes
V(network_geo)$color <- ifelse(V(network_geo)$name %in% location_counts$Location &
                              vertex_sizes > 4.5,
                              location_colors[match(V(network_geo)$name, location_counts$Location)],
                              "grey")

# Plot the network
plot(network_geo,
     vertex.label = NA, # Remove labels from the plot
     vertex.size = V(network_geo)$size,
     vertex.color = V(network_geo)$color,
     edge.arrow.size = 0.3,
     layout = layout_with_fr,
)

# Add legend for locations with size > 4.5
location_indices <- match(location_counts$Location, V(network_geo)$name)
large_locations <- location_counts$Location[vertex_sizes[location_indices] > 4.5]

large_location_colors <- location_colors[
  match(large_locations, location_counts$Location)
]
legend("topright",
     legend = large_locations,
     col = large_location_colors,
     pch = 19,
     title = "Locations")

```



Wie zu erwarten war, sind die meisten Unternehmen Ballungszentren wie New York, Chicago und San Francisco angesiedelt.

```
# Ausgabe der farbigen Standorte
print(large_locations)
```

3.2.1.2 Vergleich der Gehälter zwischen den Hotspot- und den anderen Regionen

```
## [1] "Boston, MA"          "Cambridge, MA"      "Chicago, IL"
## [4] "New York, NY"        "San Francisco, CA"  "San Jose, CA"
```

```
# Filterung der Daten für die Hotspot-Regionen
data_hotspots <- data %>%
  filter(`Location` %in% large_locations)
```

```

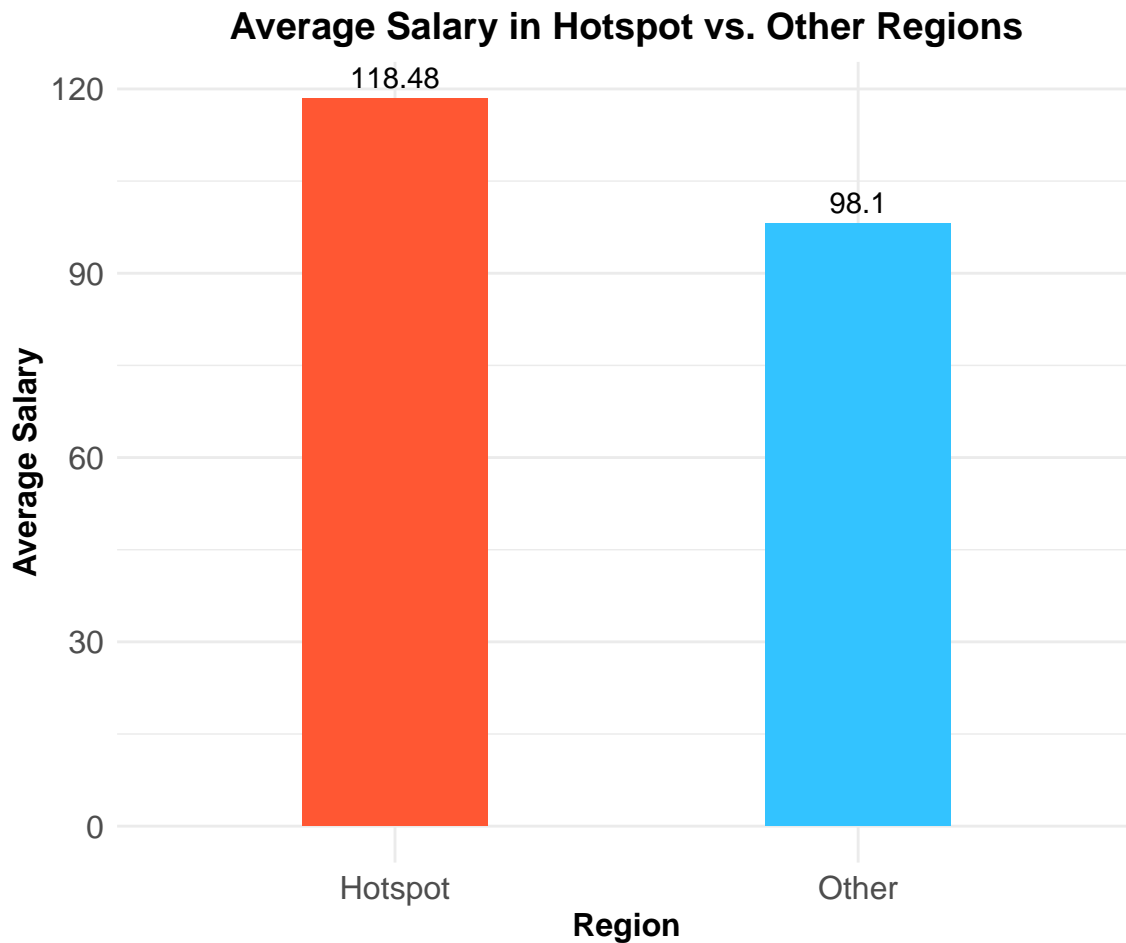
# Filterung der Daten für die anderen Regionen
data_other <- data %>%
  filter(!`Location` %in% large_locations)

# Durchschnittsgehalt in den Hotspot-Regionen
avg_salary_hotspots <- mean(data_hotspots$`Salary Estimate`, na.rm = TRUE)

# Durchschnittsgehalt in den anderen Regionen
avg_salary_other <- mean(data_other$`Salary Estimate`, na.rm = TRUE)

# Erstellung eines Balkendiagramms
ggplot(data = data.frame(Region = c("Hotspot", "Other"),
                          Average_Salary = c(avg_salary_hotspots,
                                              avg_salary_other)),
       aes(x = Region, y = Average_Salary, fill = Region)) +
  geom_bar(stat = "identity", width = 0.4) +
  scale_fill_manual(values = c("Hotspot" = "#FF5733", "Other" = "#33C3FF")) +
  labs(title = "Average Salary in Hotspot vs. Other Regions",
       x = "Region",
       y = "Average Salary") +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5, size = 14, face = "bold"),
    axis.title.x = element_text(size = 12, face = "bold"),
    axis.title.y = element_text(size = 12, face = "bold"),
    axis.text.x = element_text(size = 12),
    axis.text.y = element_text(size = 12),
    legend.position = "none"
  ) +
  geom_text(aes(label = round(Average_Salary, 2)), vjust = -0.5, size = 4)

```



```
# Berechnung der Gehaltsunterschiede
salary_diff <- avg_salary_hotspots - avg_salary_other

# Ausgabe der Gehaltsunterschiede
print(paste("Durchschnittsgehalt in Hotspot-Regionen:", avg_salary_hotspots))

## [1] "Durchschnittsgehalt in Hotspot-Regionen: 118.475247524752"
print(paste("Durchschnittsgehalt in anderen Regionen:", avg_salary_other))

## [1] "Durchschnittsgehalt in anderen Regionen: 98.1"
print(paste("Durchschnittlicher Gehaltsunterschied:", salary_diff))

## [1] "Durchschnittlicher Gehaltsunterschied: 20.3752475247525"
```

Das Ergebnis zeigt, dass entsprechend der vorher getroffenen Annahme, die Gehälter in den Hotspot-Regionen im Durchschnitt höher sind als in anderen Regionen. Dies impliziert eine Korrelation zwischen geografischer Nähe und Gehaltsniveau.

Deswegen sollen am Ende dieser Arbeit die Ergebnisse der Wettbewerbsanalyse mit den Ergebnissen der geografischen Analyse verglichen und in Bezug gesetzt werden.

3.3 Wettbewerbsnetzwerk

In diesem Abschnitt wird mit der eigentlichen Analyse, dem Ziel dieser Arbeit, der Erstellung einer Wettbewerbsanalyse begonnen.

Zu diesem Zweck wird ein Netzwerk erstellt, das auf den Wettbewerbsbeziehungen zwischen Unternehmen basiert.

Die Wettbewerbsbeziehungen werden anhand der in der Spalte "Competitors" aufgeführten Unternehmen definiert. Die Punkte im Netzwerk repräsentieren die Unternehmen, während die Kanten die Wettbewerbsbeziehungen zwischen ihnen darstellen.

Die Gewichtung der Kanten erfolgt wie folgt:

- Direkte Wettbewerber erhalten eine Gewichtung von 1.
- Unternehmen in derselben Branche, jedoch nicht als direkte Wettbewerber aufgeführt, erhalten eine Gewichtung von 0.5.

Branchenbezogene Wettbewerbsbeziehungen sind in blau dargestellt, während direkte Wettbewerber in rot hervorgehoben sind.

```
# Extrahiere Unternehmen und ihre Wettbewerber
edges <- data %>%
  filter(!is.na(Competitors) & Competitors != "-1") %>%
  separate_rows(Competitors, sep = ", ") %>%
  select(`Company Name`, Competitors) %>%
  rename(from = `Company Name`, to = Competitors) %>%
  mutate(weight = 1) # Gewichtung für direkte Wettbewerber

# Füge Unternehmen in derselben Branche mit Gewichtung 0.5 hinzu
industry_edges <- data %>%
  filter(!is.na(Industry)) %>%
  select(`Company Name`, Industry) %>%
  inner_join(
    data %>% select(`Company Name`, Industry),
    by = "Industry",
    relationship = "many-to-many"
  ) %>%
  filter(`Company Name.x` != `Company Name.y`) %>%
  select(from = `Company Name.x`, to = `Company Name.y`) %>%
  mutate(weight = 0.5) # Gewichtung für gleiche Branche

# Kombiniere beide Datensätze
all_edges <- bind_rows(edges, industry_edges)

# Erstelle den Graphen
g_competitors <- graph_from_data_frame(all_edges, directed = FALSE)

# Entferne mehrere Kanten zwischen denselben Punkten
g_competitors <- simplify(g_competitors, remove_multiple = TRUE,
  edge.attr.comb = "first")

# Setze die Farben der Kanten basierend auf der Gewichtung
E(g_competitors)$color <- ifelse(E(g_competitors)$weight == 1, "red", "blue")

# Visualisiere das Netzwerk mit kleineren Knoten
plot(g_competitors, vertex.label = NA,
  vertex.size = 2, # Kleinere Knoten
  edge.width = E(g_competitors)$weight, # Gewichtung der Kanten
  edge.arrow.size = 0.5, # Kleinere Pfeile
  main = "Unternehmensnetzwerk basierend auf Wettbewerbern und Branchen",
```

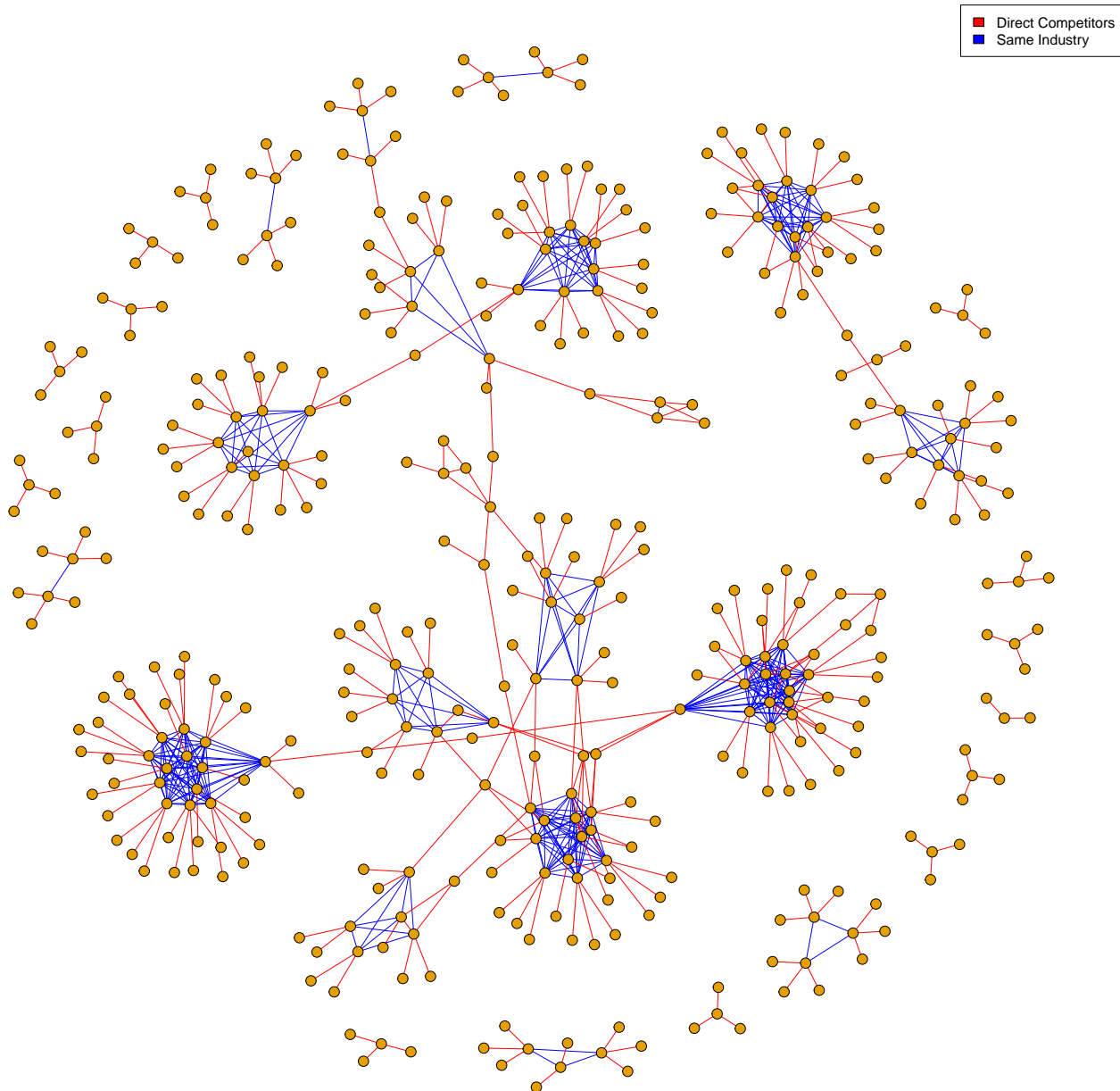
```

layout = layout_with_fr)

# Legende für Kantenfarben
legend("topright", legend = c("Direct Competitors", "Same Industry"),
      fill = c("red", "blue"))

```

Unternehmensnetzwerk basierend auf Wettbewerbern und Branchen



Es lassen sich einige interessante Beobachtungen aus dem Netzwerk ziehen. Einerseits sind ganz eindeutig Branchencluster zu erkennen, die auf die Branchenzugehörigkeit der Unternehmen hinweisen.

```

# Ausgeben der 10 häufigsten Branchen im Netzwerk
top_industries <- data %>%
  count(Industry, sort = TRUE) %>%
  head(10)

```

```
# Welche Unternehmen sind in mehreren Branchen vertreten?
multi_industry_companies <- data %>%
  group_by(`Company Name`) %>%
  summarise(Num_Industries = n_distinct(Industry)) %>%
  filter(Num_Industries > 1) %>%
  arrange(desc(Num_Industries))

# Ausgabe der Unternehmen, die in mehreren Branchen vertreten sind
print(multi_industry_companies)
```

```
## # A tibble: 0 x 2
## # i 2 variables: Company Name <chr>, Num_Industries <int>
```

Jedoch sind keine Unternehmen in mehreren Branchen vertreten, was darauf hindeutet, dass die Branchenzugehörigkeit eindeutig ist. Aber es gibt einige Unternehmen, die mit direkter Konkurrenz die verschiedenen Branchen verbinden. Dies könnte auf eine Diversifikation der Geschäftsfelder hindeuten, die eine breitere Wettbewerbsbasis schafft.

Da aber wie oberhalb dargestellt, die Branchenzugehörigkeit eindeutig ist, und somit die Branchenzugehörigkeit keinen Mehrwert für die Analyse bietet, wird diese nicht weiter verfolgt.

3.3.1 Bloß direkte Wettbewerber

Aus diesem Grund wird das Netzwerk auf direkte Wettbewerber beschränkt, um die Analyse zu vereinfachen und die Relevanz der Wettbewerbsbeziehungen zu erhöhen.

```
# Extrahiere Unternehmen und ihre Wettbewerber
edges <- data %>%
  filter(!is.na(Competitors) & Competitors != "-1") %>%
  separate_rows(Competitors, sep = ", ") %>%
  select(`Company Name`, Competitors) %>%
  rename(from = `Company Name`, to = Competitors) %>%
  mutate(weight = 1) # Gewichtung für direkte Wettbewerber

# Summiere die Gewichtungen für mehrere Kanten zwischen denselben Punkten
edge_weights <- edges %>%
  group_by(from, to) %>%
  summarise(weight = sum(weight), .groups = 'drop')

# Erstelle den Graphen nur mit direkten Wettbewerbern
g_direct_competitors <- graph_from_data_frame(edge_weights, directed = FALSE)

# Setze die Gewichtungen der Kanten im Graphen
E(g_direct_competitors)$weight <- edge_weights$weight

# Füge die Gehaltsdaten hinzu und berechne das durchschnittliche Gehalt pro Unternehmen
salary_data <- data %>%
  group_by(`Company Name`) %>%
  summarise(AverageSalary = mean(`Salary Estimate`, na.rm = TRUE))

# Füge die Gehaltsdaten zu den Knoten des Graphen hinzu
V(g_direct_competitors)$salary <- salary_data$AverageSalary[match(V(g_direct_competitors)$name, salary_data$name)]

# Setze die Farben der Knoten basierend auf den Gehältern
salary_quantiles <- quantile(V(g_direct_competitors)$salary, probs = seq(0, 1, length.out = 6), na.rm = TRUE)
```

```

color_palette <- brewer.pal(5, "YlOrRd")
V(g_direct_competitors)$color <- cut(V(g_direct_competitors)$salary,
                                     breaks = salary_quantiles,
                                     labels = FALSE,
                                     include.lowest = TRUE)
V(g_direct_competitors)$color <- color_palette[V(g_direct_competitors)$color]

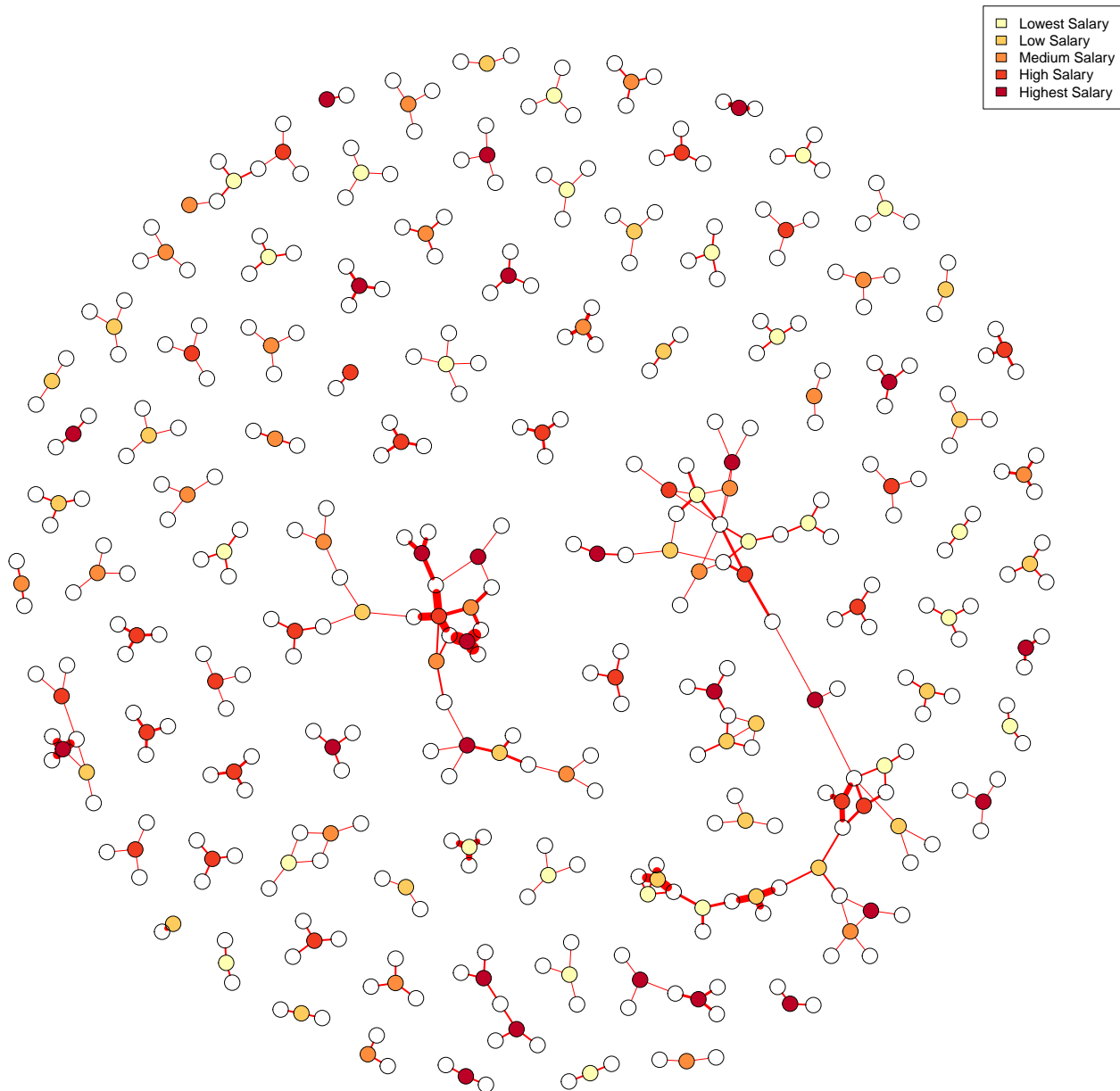
# Setze die Farben der Kanten basierend auf der Gewichtung
E(g_direct_competitors)$color <- "red"

# Visualisiere das Netzwerk mit kleineren Knoten
plot(g_direct_competitors, vertex.label = NA,
     vertex.size = 3, # Kleinere Knoten
     edge.width = 1 * E(g_direct_competitors)$weight, # Reduzierte Gewichtung der Kanten
     edge.arrow.size = 1,
     main = "Unternehmensnetzwerk basierend auf direkten Wettbewerbern",
     layout = layout_with_fr
)

# Legende für Knotenfarben
legend("topright", legend = c("Lowest Salary", "Low Salary", "Medium Salary", "High Salary", "Highest Salary"),
      fill = color_palette)

```

Unternehmensnetzwerk basierend auf direkten Wettbewerbern



Ausgabe der Gehälter der Unternehmen

```
salary_data <- data %>%
  group_by(`Company Name`) %>%
  summarise(AverageSalary = mean(`Salary Estimate`, na.rm = TRUE)) %>%
  arrange(desc(AverageSalary))
```

Ausgabe aller Unternehmen zusammen mit ihren Gehältern

```
print(salary_data, n = Inf)
```

```
## # A tibble: 127 x 2
##   `Company Name`
##   <chr>
## 1 Gallup
## 2 Credit Sesame
```

```
AverageSalary
<dbl>
238.
205
```

##	3 The Climate Corporation	194
##	4 Samsung Research America	177
##	5 Nektar Therapeutics	174
##	6 BioMarin Pharmaceutical	168
##	7 Adobe	162
##	8 Glassdoor	162
##	9 Netskope	154.
##	10 Liberty Mutual Insurance	154.
##	11 Factual	153
##	12 Visa Inc.	153.
##	13 Sunovion	151.
##	14 Sumo Logic	150.
##	15 Western Digital	147.
##	16 Capgemini	147
##	17 Tapjoy	147.
##	18 Mitsubishi Electric Research Labs	144.
##	19 1904labs	144.
##	20 <intent>	140
##	21 Demandbase	139.
##	22 Walmart	139
##	23 Red Ventures	134
##	24 Takeda Pharmaceuticals	132.
##	25 Johns Hopkins University Applied Physics Laboratory	130
##	26 CBS Interactive	128
##	27 Novetta	128.
##	28 Information Builders	125
##	29 AstraZeneca	124.
##	30 TriNet	124.
##	31 Swiss Re	123
##	32 L.A. Care Health Plan	120.
##	33 Equity Residential	118.
##	34 PennyMac	117.
##	35 Affinity Solutions	114.
##	36 Pactera	114.
##	37 Genesys	112.
##	38 New England Biolabs	112.
##	39 IQVIA	112.
##	40 Assurant	110.
##	41 GNY Insurance Companies	110.
##	42 The Integer Group	110
##	43 The Buffalo Group	108.
##	44 Echo Global Logistics	108.
##	45 Johns Hopkins Health Care	108.
##	46 Mentor Graphics	107
##	47 NCSOFT	106.
##	48 Porch	106.
##	49 Medidata Solutions	105
##	50 PA Consulting	104
##	51 Maximus Real Estate Partners	104.
##	52 Blueprint Medicines	102
##	53 Genworth	102
##	54 Blue Cross & Blue Shield of Rhode Island	100
##	55 Centro	100
##	56 PatientPoint	100

## 57 Plymouth Rock Assurance	98.5
## 58 Carmeuse	98
## 59 Pilot Flying J Travel Centers LLC	98
## 60 Eventbrite	97.7
## 61 Strategic Financial Solutions	97.5
## 62 comScore	96.5
## 63 CapTech	95.5
## 64 CyrusOne	95
## 65 Object Partners	95
## 66 Remedy BPCI Partners, LLC.	95
## 67 Exelixis	93.8
## 68 23andMe	92
## 69 L&T Infotech	90
## 70 Lockheed Martin	89
## 71 Productive Edge	88.8
## 72 Sauce Labs	88
## 73 Fivestars	87.5
## 74 GSK	87.5
## 75 Moda Operandi	87.5
## 76 Trilogy Ed	87.5
## 77 PNNL	87.0
## 78 AVANADE	87
## 79 Biz2Credit Inc	87
## 80 Mteq	87
## 81 Centauri	86.5
## 82 Esri	85.8
## 83 Rapid Response Monitoring	85.5
## 84 DICK'S Sporting Goods - Corporate	85
## 85 First Command Financial Services, Inc.	85
## 86 ICW Group	83
## 87 Edgewell Personal Care	82.5
## 88 MITRE	82.1
## 89 Pharmavite	81.5
## 90 General Dynamics Information Technology	81.2
## 91 Caterpillar	81
## 92 IHS Markit	80.5
## 93 Trace3	80
## 94 Dayton Freight Lines, Inc.	79
## 95 Sapphire Digital	78
## 96 Vermeer	77.5
## 97 SullivanCotter	76
## 98 ExecOnline	75.5
## 99 Audentes Therapeutics	73
## 100 Beckman Coulter Diagnostics	73
## 101 United BioSource	72.5
## 102 IZEA	71.5
## 103 CALIBRE Systems	71
## 104 Saama Technologies Inc	68.5
## 105 Southwest Research Institute	67.7
## 106 Pacific Northwest National Laboratory	67
## 107 AmeriHealth Caritas	66.5
## 108 AXION Healthcare Solutions	66
## 109 RTI International	65.5
## 110 Guidepoint	64.5

## 111 Fareportal	64.4
## 112 Infosys	62.5
## 113 Boys Town Hospital	61.5
## 114 National Student Clearinghouse	61.5
## 115 Motorola Solutions	61
## 116 WK Dickson	60.5
## 117 Associated Banc-Corp	60
## 118 DECISIVE ANALYTICS Corporation	59
## 119 COUNTRY Financial	58.5
## 120 C Space	56.5
## 121 CentralReach	56.5
## 122 DentaQuest	56.5
## 123 Boys Town	52.5
## 124 Citadel Federal Credit Union	49
## 125 Icon Health and Fitness	47
## 126 Greenway Health	37.5
## 127 Catholic Health Initiatives	25

Kommentar!!!.....

3.4 Zentralitätsanalyse innerhalb des Netzwerkes

```
# Calculate network metrics
betweenness centrality <- betweenness(g_direct_competitors)
degree centrality <- degree(g_direct_competitors)
eigenvector centrality <- eigenvector_centrality(g_direct_competitors)$vector

closeness centrality <- closeness(g_direct_competitors)
clustering_coeff <- transitivity(g_direct_competitors, type = "local")
```

3.4.1 Betweenness-Zentralität

Jetzt soll das igraph-Paket in R verwendet werden, um die Betweenness-Zentralität für jeden Knoten zu berechnen. Dies zeigt, wie oft ein Unternehmen auf dem kürzesten Weg zwischen anderen Unternehmen liegt.

Unternehmen mit hoher Betweenness-Zentralität könnten als Brücke zwischen verschiedenen Netzwerken fungieren, was einen Wettbewerbsvorteil und möglicherweise höhere Gehälter zur Folge hat.

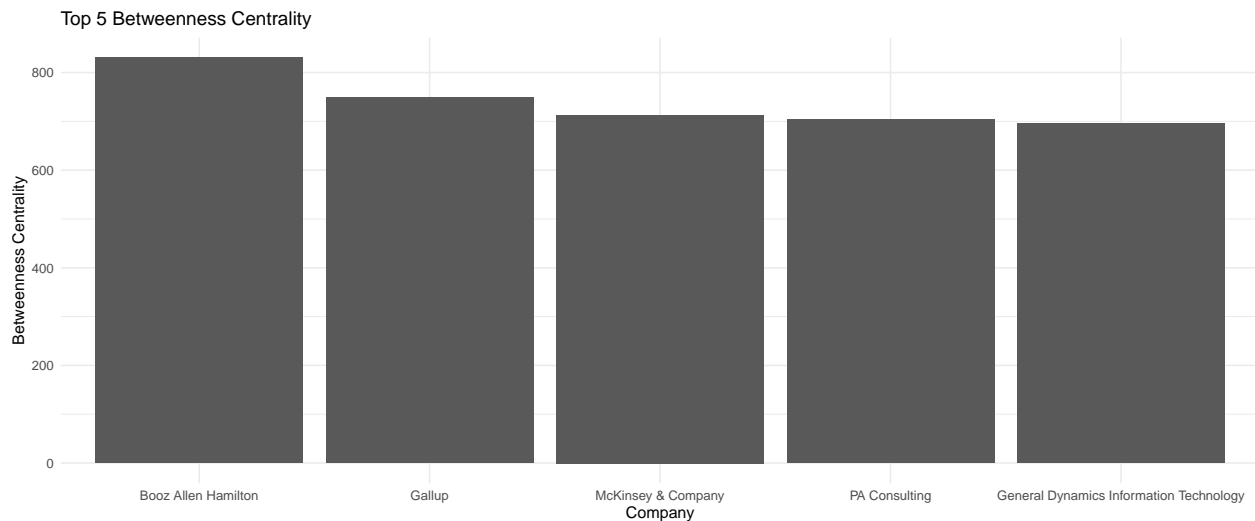
```
# Berechne die Betweenness-Centrality und sortiere sie absteigend
top_betweenness <- head(sort(betweenness centrality, decreasing = TRUE), 5)

# Erstelle ein DataFrame mit den Namen der Unternehmen und ihrer Betweenness-Centrality
top_betweenness_df <- data.frame(
  Company = names(top_betweenness),
  Betweenness = as.numeric(top_betweenness),
  stringsAsFactors = FALSE
)

# Erstelle die Tabelle und zentriere sie links
kable(top_betweenness_df, format = "latex", booktabs = TRUE, align = "l") %>%
kable_styling(latex_options = c("striped", "hold_position"), position = "left")
```


Company	Betweenness
Booz Allen Hamilton	830
Gallup	749
McKinsey & Company	713
PA Consulting	704
General Dynamics Information Technology	695

```
# Create bar plot
ggplot(top_betweenness_df, aes(x = reorder(Company, -Betweenness), y = Betweenness)) +
  geom_bar(stat = "identity") +
  labs(title = "Top 5 Betweenness Centrality",
       x = "Company",
       y = "Betweenness Centrality") +
  theme_minimal()
```



3.4.2 Degree-Zentralität

Hier wird die Anzahl der Kanten gezählt, die an jedem Knoten hängen. Hohe Werte können auf starke Verbindungen zu anderen Unternehmen hinweisen.

Ein Unternehmen mit einer hohen Degree-Zentralität ist in der Regel gut vernetzt und könnte in der Lage sein, bessere Gehälter zu zahlen, um Talente anzuziehen.

```
# Berechne die Degree-Centrality und sortiere sie absteigend
top_degree <- head(sort(degree centrality, decreasing = TRUE), 5)

# Erstelle ein DataFrame mit den Namen der Unternehmen und ihrer Degree-Centrality
top_degree_df <- data.frame(
  Company = names(top_degree),
  Degree = as.numeric(top_degree),
  stringsAsFactors = FALSE
)

# Erstelle die Tabelle und zentriere sie links
kable(top_degree_df, format = "latex", booktabs = TRUE, align = "l") %>%
  kable_styling(latex_options = c("striped", "hold_position"), position = "left")
```

Company	Degree
Accenture	7
AstraZeneca	5
Infosys	5
Booz Allen Hamilton	5
BioMarin Pharmaceutical	4

3.4.3 Eigenvector-Zentralität

```
# Berechne die Eigenvector-Centrality und sortiere sie absteigend
top_eigenvector <- head(sort(eigenvector_centrality, decreasing = TRUE), 5)

# Erstelle ein DataFrame mit den Namen der Unternehmen und ihrer Eigenvector-Centrality
top_eigenvector_df <- data.frame(
  Company = names(top_eigenvector),
  Eigenvector = as.numeric(top_eigenvector),
  stringsAsFactors = FALSE
)

# Erstelle die Tabelle und zentriere sie links
kable(top_eigenvector_df, format = "latex", booktabs = TRUE, align = "l") %>%
  kable_styling(latex_options = c("striped", "hold_position"), position = "left")
```

Company	Eigenvector
Takeda Pharmaceuticals	1.0000000
Novartis	0.6885416
Pfizer	0.5750772
Baxter	0.5510613
AstraZeneca	0.3694600

3.5 Cluster-Analyse

Klassifizierung der Unternehmen: Unternehmen werden basierend auf ihrer Netzwerkposition in zentrale (innerhalb von Netzwerken) und periphere (am Rand der Netzwerke) Gruppen klassifiziert. Gehaltsvergleich: Verwende Boxplots oder Histogramme, um die Gehaltsverteilung in beiden Gruppen zu vergleichen. Beispiel: „Die Boxplots zeigen, dass die medianen Gehälter in zentralen Positionen signifikant höher sind als in peripheren Positionen.“

Abschließend soll noch eine Clusteranalyse durchgeführt werden, die die Gehalt in Bezug auf Standort und Wettbewerb kombiniert betrachtet. **Cluster-Analyse** Clusteranalyse innerhalb des geografischen und Wettbewerbsnetzwerks. Cluster von Unternehmen nach Gehalt und Standort innerhalb des Wettbewerbsnetzwerks: Korrelation zwischen Gehältern und der Stärke von Wettbewerb und regionaler Vernetzung. Regionale Gehaltsklassen: Visualisierung der regionalen Gehaltsspreizung in Bezug auf Netzwerkcluster, z.B. ob Cluster in wirtschaftsstarken Regionen höhere Durchschnittsgehälter bieten.

```
# Detect communities
communities <- cluster_louvain(g_direct_competitors)
```

Vlt: bevor wir mit den Clustern loslegen TODO “Arc Diagramm” zu den regionalen Clustern von der geografischen Netzwerkanalyse

3.6 Ergänzung zu den Zentralitätsanalysen

```
# Prepare data for visNetwork
nodes <- data.frame(id = V(g_direct_competitors)$name,
                    label = V(g_direct_competitors)$name,
                    group = membership(communities),
                    value = degree Centrality,
                    title = paste("Degree:", degree Centrality,
                                   "<br>Betweenness:", betweenness Centrality,
                                   "<br>Closeness:", closeness Centrality,
                                   "<br>Eigenvector:", eigenvector Centrality))

edges <- data.frame(from = as.character(edges$from), to = as.character(edges$to))

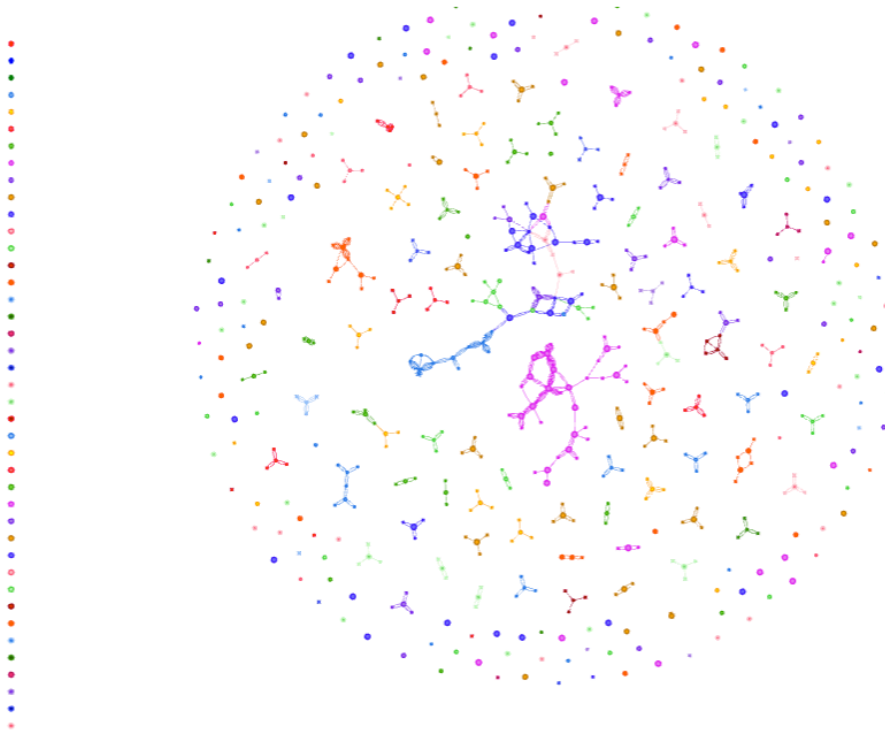
# Create interactive network visualization
visNetwork(nodes, edges) %>%
  visOptions(highlightNearest = TRUE, nodesIdSelection = TRUE) %>%
  visGroups(groupname = "1", color = "red") %>%
  visGroups(groupname = "2", color = "blue") %>%
  visGroups(groupname = "3", color = "green") %>%
  visGroups(groupname = "4", color = "yellow") %>%
  visGroups(groupname = "5", color = "purple") %>%
  visGroups(groupname = "6", color = "orange") %>%
  visGroups(groupname = "7", color = "pink") %>%
  visLayout(randomSeed = 123) %>%
  visLegend()
```

Select by id



```
# Fügt ein Bild der interaktiven Netzwerkvisualisierung hinzu
knitr::include_graphics("interaktive_Netzwerke_Bilder/Übersicht.png")
```

Select by id



```
knitr::include_graphics("interaktive_Netzwerke_Bilder/NVDIA.png")
```

NVIDIA



Degree: 1
Betweenness: 0
Closeness: 0.000630318310746927
Eigenvector: 4.32887877922348e-06

Zugriff auf die interaktive Visualisierung über das Repository (Dateiname: network.html): <https://github.com/Mzaex7/SNA>

4 Conclusion

Zusammenfassung der zentralen Ergebnisse:

Bedeutung: Diskutiere, wie wichtig geografische Nähe und Wettbewerbsumfeld für die Karriereentwicklung von Data Scientists sind.

Praktische Implikationen: Gib Empfehlungen für Jobuchende, wie sie Standorte und Unternehmen auswählen sollten, um die besten Gehaltsaussichten zu haben. Dies könnte auch für Unternehmen von Interesse sein, um zu verstehen, wie sie ihre Position im Markt verbessern können.

5 Literaturverzeichnis

Davenport, Thomas H.; Patil, D. J. 2012. »Data Scientist: The Sexiest Job of the 21st Century«, in Harvard Business Review vom 1. Oktober 2012. <https://hbr.org/2012/10/data-scientist-the-sexiest-job-of-the-21st-century> (Zugriff vom 30.10.2024).

Google Trends, <https://trends.google.com/trends/explore?date=all&q=%22data%20science%22,%22data%20scientist%22> (Zugriff vom 30.10.2024).