

# Das Berufsbild des Data Scientisten aufgeschlüsselt

## Contents

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Requirements . . . . .	1
1.2	Motivation und Zielsetzung . . . . .	1
1.3	Forschungsfrage . . . . .	2
1.4	Datengrundlage . . . . .	3
1.4.1	CSV einlesen . . . . .	3
1.4.2	Erste Ansicht der Daten . . . . .	3
<b>2</b>	<b>Analysestrategie</b>	<b>7</b>
<b>3</b>	<b>Analyse</b>	<b>8</b>
3.1	Datenbereinigung . . . . .	8
3.2	Visualisierung . . . . .	10
3.3	Datenvisualisierung . . . . .	11
3.4	Zweite Copilot iteration . . . . .	13
<b>4</b>	<b>Conclusion</b>	<b>18</b>
<b>5</b>	<b>Literaturverzeichnis</b>	<b>18</b>

## 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")
```

```
# Bibliotheken laden
library(tidyverse)
library(igraph)
library(visNetwork)
library(dplyr)
library(tidyr)
```

### 1.2 Motivation und Zielsetzung

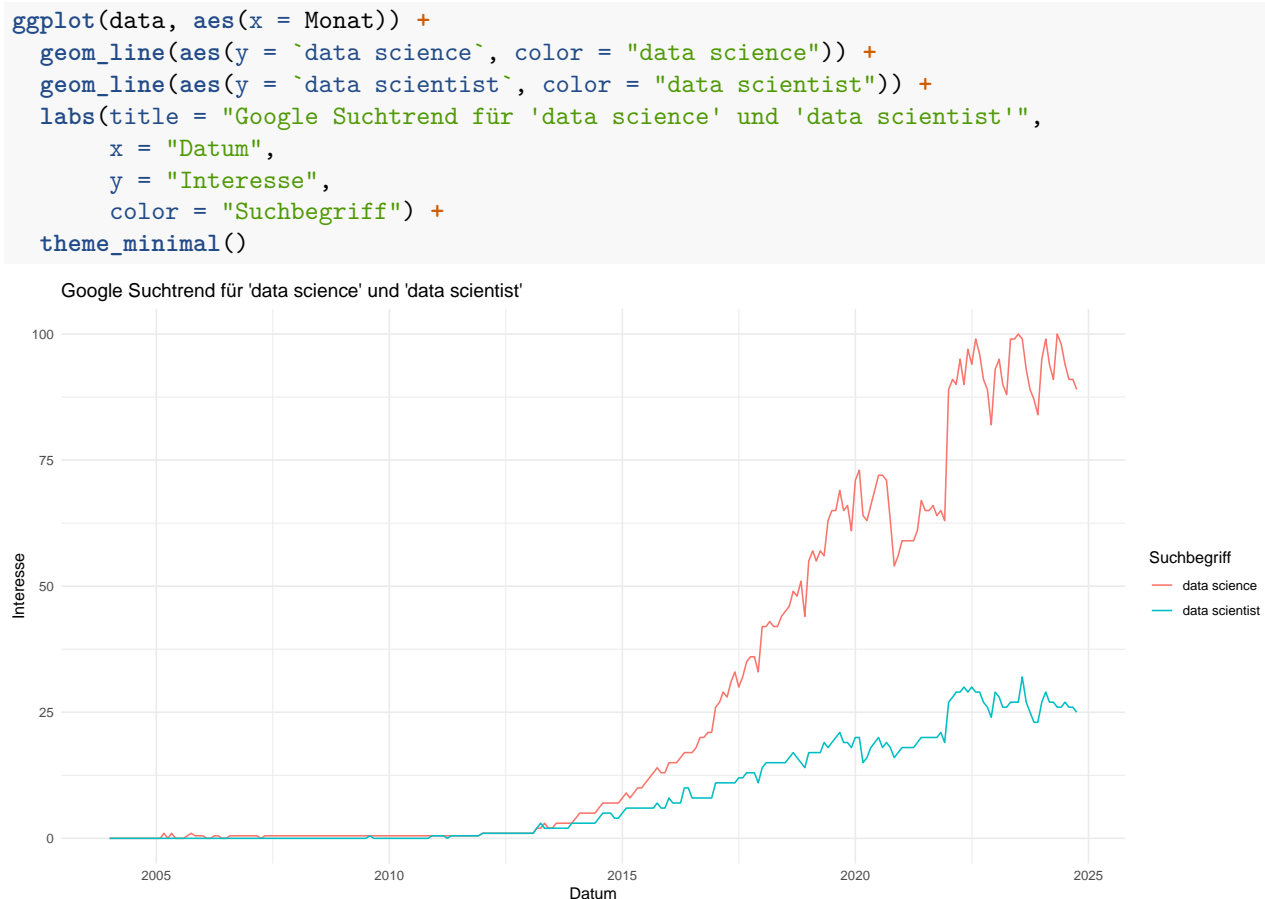
In ihrem Artikel “Data Scientist: The Sexiest Job of the 21st Century” betonen Davenport und Patil, dass Data Scientists durch ihre Fähigkeiten in Informatik, Statistik und ihr Fachwissen allgemein einen erheblichen Mehrwert für Unternehmen schaffen.<sup>1</sup> Die Fähigkeit, aus komplexen, unstrukturierten Daten wertvolle

---

<sup>1</sup>Davenport, Patil 2012

Erkenntnisse zu gewinnen, macht Data Scientists in vielen Branchen zu einer unverzichtbaren Ressource.<sup>2</sup> Die Nutzung ihrer Kompetenzen verschafft Unternehmen einen Wettbewerbsvorteil, da sie datengetriebene Entscheidungen, Produktinnovationen und Effizienzsteigerungen ermöglicht.<sup>3</sup>

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:<sup>4</sup>



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

<sup>2</sup>Davenport, Patil 2012

<sup>3</sup>Davenport, Patil 2012

<sup>4</sup>Google Trends, abgerufen am 30.10.2024

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/maxzeitler/data-science-job-salary-prediction-glassdoor/edit> 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, der von Kaggle stammt und Informationen zu Data-Science-Jobs in verschiedenen Unternehmen 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~ Albuque~
## 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.

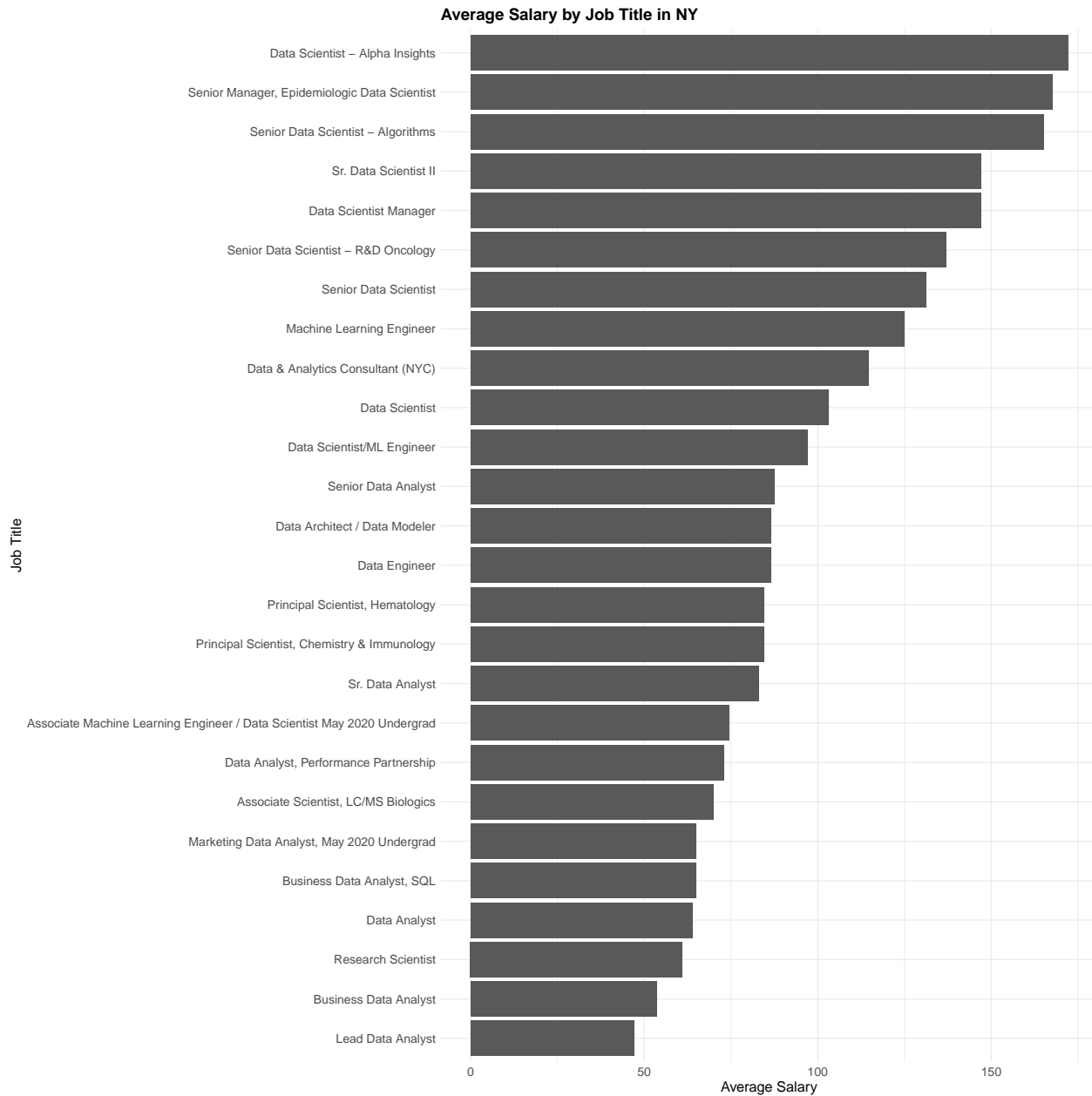
Nachdem die Daten in Python mit Hilfe von Pandas bereinigt, ergänzt und bearbeitet wurden, können sie nun in R eingelesen werden. Dabei wurde sich an <https://www.kaggle.com/code/maxzeitler/data-science-job-salary-prediction-glassdoor/edit> orientiert.

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

```
# Filter data for the state of New York (NY)
data_ny <- data %>%
  filter(State == "NY")

# Calculate average salary by job title for NY
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 of average salary by job title for NY
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")
  )
```



Kanten und Knoten ergänzen

## 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

Bei der Durchsicht des Datensatzes fiel 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.

```
# Select "State" and "job_state" columns
selected_data <- data %>%
  select(State, job_state)

# Display the first few rows of the selected data
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.

```
# Check if the "State" and "job_state" columns are identical
if (all(selected_data$State == selected_data$job_state, na.rm = TRUE)) {
  print("All values in 'State' and 'job_state' columns are identical.")
} else {
  print("There are differences between 'State' and 'job_state' columns.")
}
```

```
## [1] "There are differences between 'State' and 'job_state' columns."
```

Jedoch trügt der Schein, da es Unterschiede gibt.

```
# Filter rows where State is not equal to job_state
different_states <- selected_data %>%
  filter(State != job_state)

# Display all rows where State is not equal to 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.

```
# First step: Replace "Los Angeles" with "LA"
data <- data %>%
  mutate(State = ifelse(State == "Los Angeles", "LA", State),
         job_state = ifelse(job_state == "Los Angeles", "LA", job_state))

# Second step: Replace "LA" with "CA"
data <- data %>%
  mutate(State = ifelse(State == "LA", "CA", State),
         job_state = ifelse(job_state == "LA", "CA", job_state))

# Re-check for differences after correction
selected_data <- data %>%
  select(State, job_state)

# Check if the "State" and "job_state" columns are identical
if (all(selected_data$State == selected_data$job_state, na.rm = TRUE)) {
  print("All values in 'State' and 'job_state' columns are identical.")
} else {
  print("There are differences between 'State' and 'job_state' columns.")
}
```

```
}
```

```
## [1] "All values in 'State' and 'job_state' columns are identical."
```

## 3.2 Visualisierung

```
# AAus 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,  
  sqrt(location_counts$Company_Count[match(V(network_geo)$name, location_counts$Location)  
  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)$name %in% filtered_edges$Company,  
  company_colors,  
  ifelse(vertex_sizes > 5, location_colors[match(V(network_geo)$name, location_counts$Location)],  
  company_colors))
```

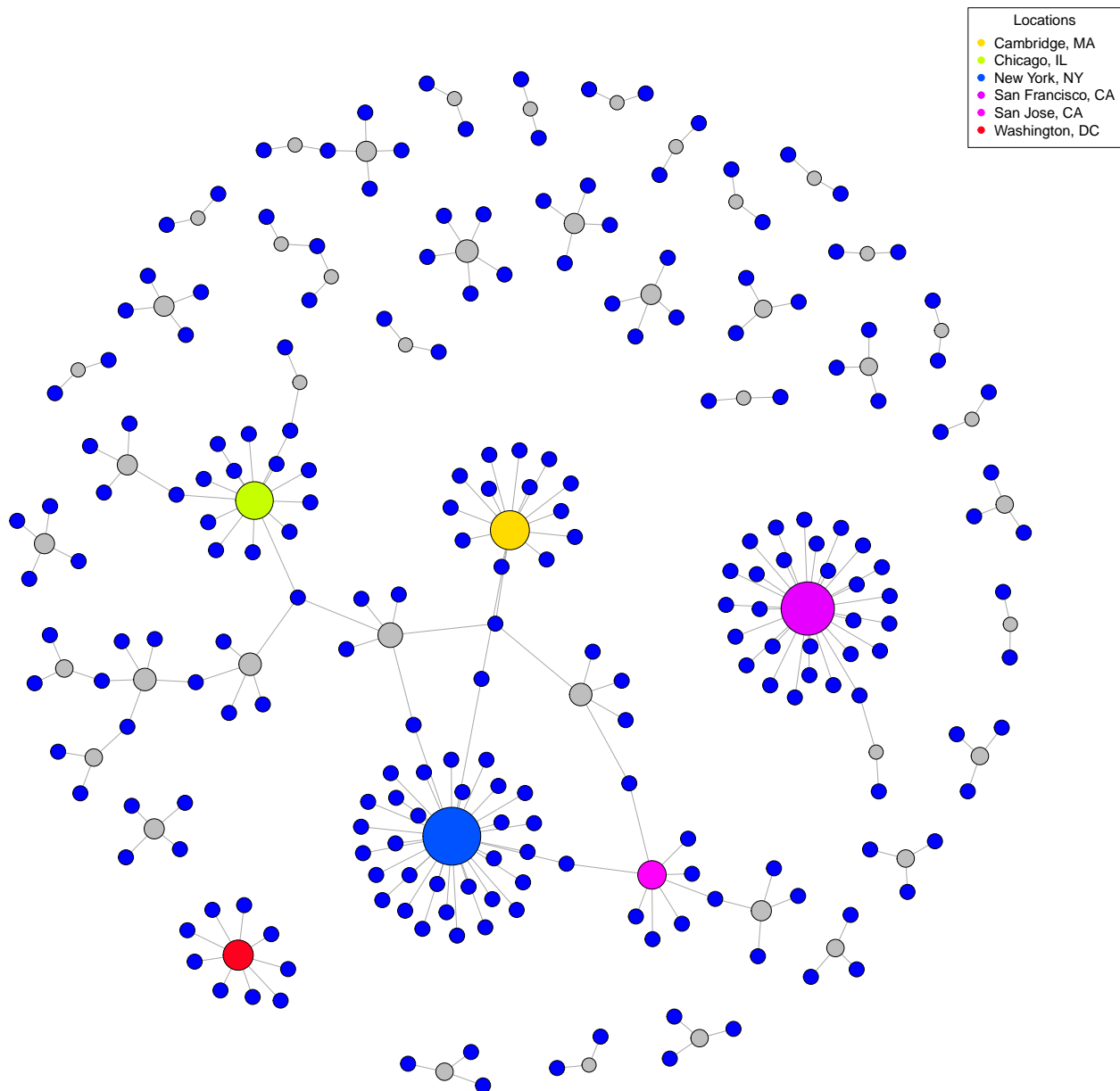
```
# Plot the network
```

```
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 > 5
```

```
large_locations <- location_counts$Location[vertex_sizes[match(location_counts$Location, V(network_geo)$name)] > 5]  
large_location_colors <- location_colors[match(large_locations, location_counts$Location)]
```

```
legend("topright", legend = large_locations, col = large_location_colors, pch = 19, title = "Locations")
```



### 3.3 Datenvisualisierung

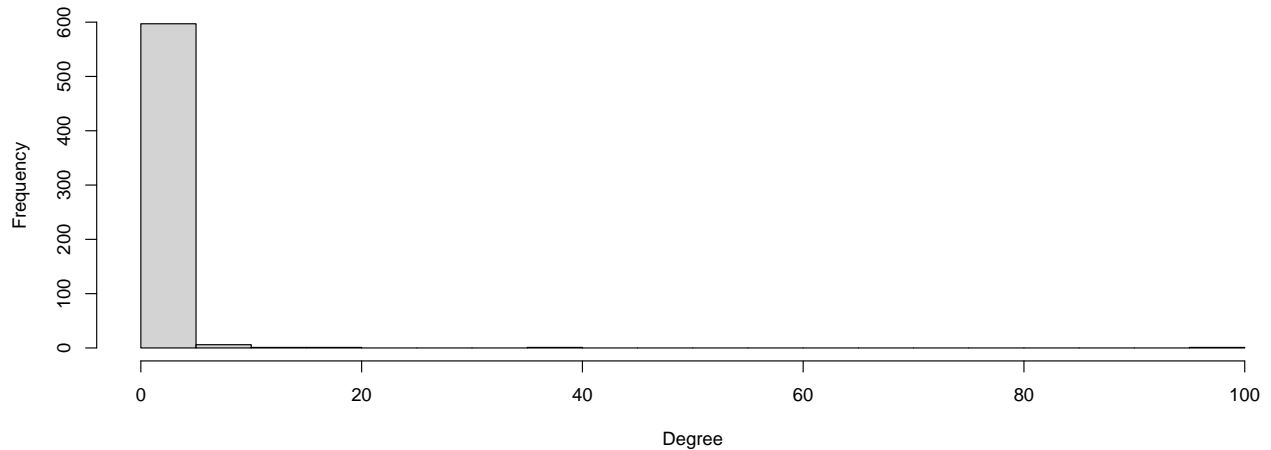
#Netzwerk von Jobtiteln und Unternehmen: #Visualisierung des Netzwerks, das zeigt, welche Unternehmen die meisten unterschiedlichen Jobtitel anbieten. #Interpretation: Zentralität der Unternehmen und welche Rolle sie im Jobmarkt spielen. #Degree distribution

```
# Create the network object
edges <- data %>%
  select(`Job Title`, `Company Name`) %>%
  distinct() %>%
  rename(from = `Job Title`, to = `Company Name`)

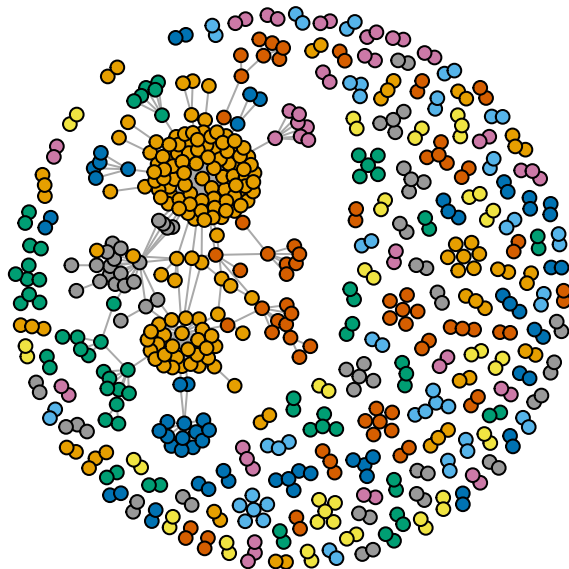
network_job_company <- graph_from_data_frame(edges, directed = FALSE)
```

```
degree_distribution <- degree(network_job_company)
hist(degree_distribution, breaks = 30, main = "Degree Distribution",
     xlab = "Degree", ylab = "Frequency")
```

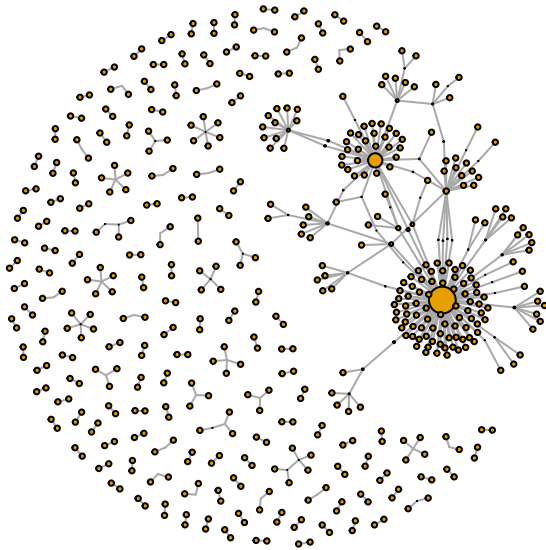
Degree Distribution



```
# Community detection using the Louvain method
communities <- cluster_louvain(network_job_company)
plot(network_job_company, vertex.label = NA, vertex.size = 5,
     vertex.color = communities$membership)
```



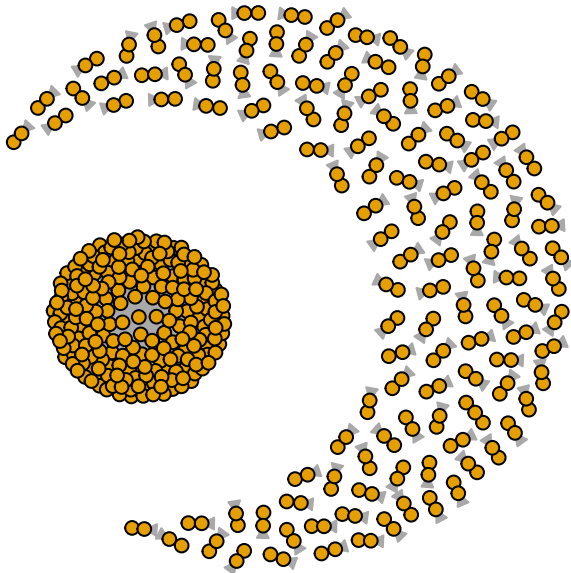
```
# Calculate betweenness centrality
betweenness centrality <- betweenness(network_job_company)
V(network_job_company)$size <- betweenness centrality /
  max(betweenness centrality) * 10 # Scale sizes
plot(network_job_company, vertex.label = NA,
     vertex.size = V(network_job_company)$size)
```



```
# Extract relevant columns for competition analysis
edges_competition <- data %>%
  select(Company_Name = `Company Name`, Competitor = `Competitors`) %>%
  distinct()

# Create an igraph object for competition analysis
network_competition <- graph_from_data_frame(edges_competition, directed = TRUE)

# Plot the competition network
plot(network_competition, vertex.label = NA, vertex.size = 5, edge.arrow.size = 0.5)
```



### 3.4 Zweite Copilot iteration

```
# Extrahiere Unternehmen und ihre Wettbewerber
edges <- data %>%
  filter(!is.na(Competitors) & Competitors != "-1") %>%
  separate_rows(Competitors, sep = ", ") %>%
```

```

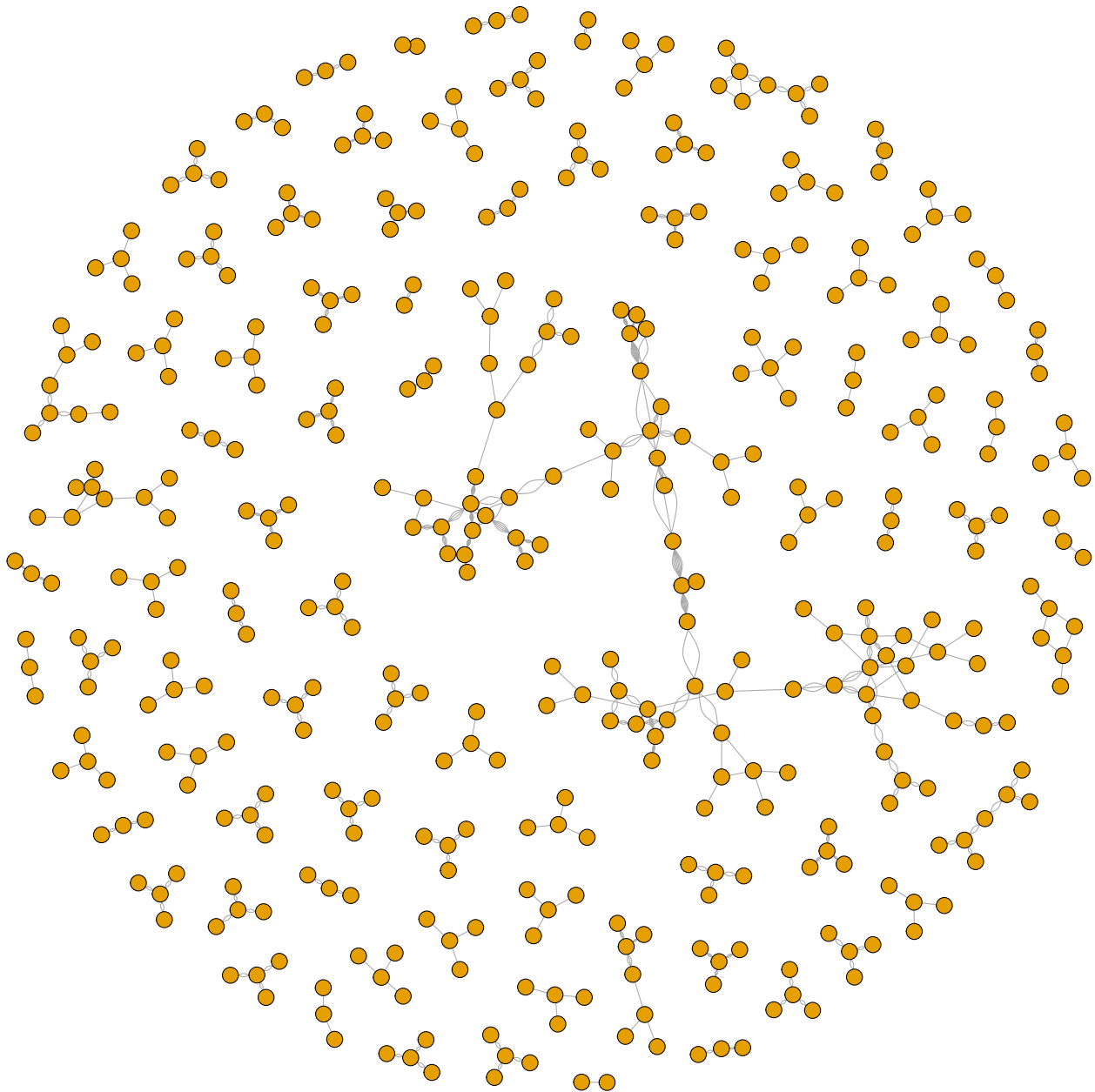
select(`Company Name`, Competitors) %>%
  rename(from = `Company Name`, to = Competitors)

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

# Visualisiere das Netzwerk mit kleineren Knoten
plot(g_competitors, vertex.label = NA,
     vertex.size = 3, # Kleinere Knoten
     edge.arrow.size = 0.5, # Kleinere Pfeile
     main = "Unternehmensnetzwerk basierend auf Wettbewerbern",
     ) # Höhe der Grafik in Pixeln

```

## Unternehmensnetzwerk basierend auf Wettbewerbern



```
# Calculate network metrics
degree_centrality <- degree(g_competitors)
betweenness_centrality <- betweenness(g_competitors)
closeness_centrality <- closeness(g_competitors)
eigenvector_centrality <- eigen_centrality(g_competitors)$vector
clustering_coeff <- transitivity(g_competitors, type = "local")

# Detect communities
communities <- cluster_louvain(g_competitors)

# Prepare data for visNetwork
```

```

nodes <- data.frame(id = V(g_competitors)$name,
                    label = V(g_competitors)$name,
                    group = membership(communities),
                    value = degree centrality,
                    title = paste("Degree:", degree centrality, "<br>Betweenness:", betweenness_central

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") %>%
  visLayout(randomSeed = 123) %>%
  visLegend()

# Print top nodes for each centrality measure
print("Top 5 nodes by degree centrality:")

## [1] "Top 5 nodes by degree centrality:"
print(head(sort(degree centrality, decreasing = TRUE), 5))

##      Takeda Pharmaceuticals      AstraZeneca Liberty Mutual Insurance
##                42                33                31
##                PNNL                Novartis
##                30                25

print("Top 5 nodes by betweenness centrality:")

## [1] "Top 5 nodes by betweenness centrality:"
print(head(sort(betweenness centrality, decreasing = TRUE), 5))

##                Booz Allen Hamilton                Gallup
##                841.8485                749.0000
##                PA Consulting                McKinsey & Company
##                715.4191                713.0000
## General Dynamics Information Technology
##                695.0000

print("Top 5 nodes by closeness centrality:")

## [1] "Top 5 nodes by closeness centrality:"
print(head(sort(closeness centrality, decreasing = TRUE), 5))

##                Esri                CareDx Medidata Solutions                Factual
##                1                1                1                1
##                Pitney Bowes
##                1

print("Top 5 nodes by eigenvector centrality:")

## [1] "Top 5 nodes by eigenvector centrality:"
print(head(sort(eigenvector centrality, decreasing = TRUE), 5))

```



```
## Takeda Pharmaceuticals      Novartis      Pfizer
##      1.0000000      0.6885416      0.5750772
##      Baxter      AstraZeneca
##      0.5510613      0.3694600
```

```
print("Top 5 nodes by clustering coefficient:")
```

```
## [1] "Top 5 nodes by clustering coefficient:"
```

```
print(head(sort(clustering_coef, decreasing = TRUE), 5))
```

```
## CNH Industrial      Vermeer      L&T Infotech      Caterpillar      Pactera
##      1.0000000      0.6666667      0.6666667      0.3333333      0.3333333
```

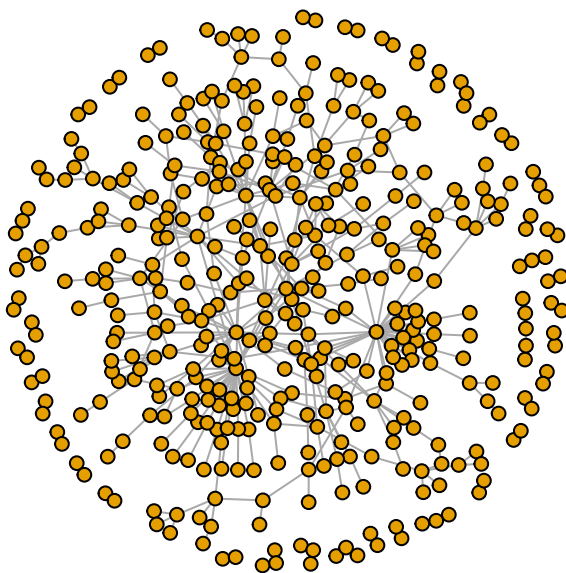
```
# Standort-Cluster für Gehälter und Bewertungen
edges_location_salary <- data %>%
  select(Location, `Salary Estimate`) %>%
  distinct() %>%
  mutate(`Salary Estimate` = as.numeric(gsub("[^0-9]", "", `Salary Estimate`))) %>%
  drop_na() %>%
  rename(from = Location, to = `Salary Estimate`)

# Remove duplicate edges
edges_location_salary <- edges_location_salary %>%
  distinct(from, to, .keep_all = TRUE)

# Erstelle den Graphen
g_location_salary <- graph_from_data_frame(edges_location_salary, directed = FALSE)

# Visualisiere das Netzwerk
plot(g_location_salary, vertex.label = NA, vertex.size = 5,
     edge.arrow.size = 0.5, main = "Standort-Cluster für Gehälter und Bewertungen")
```

## Standort-Cluster für Gehälter und Bewertungen



## 4 Conclusion

.....

## 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).