

README R-Projekt Gruppe 38 Bundesliga

David Schwefel, Len Bauer

2023-02-02

1. Übersicht

Dieses Paket enthält die Ordner “Csv”, “Python” und “R” sowie dieses README “README_Gruppe_38”. Der hier betrachtete Datensatz beinhaltet Informationen über die letzten 10 Saisons der aktuell besten fünf europäischen Fußball-Ligen (nach dem aktuellsten Update der UEFA-5-Jahreswertung). Dabei speichert der Datensatz die Variablen “Liga”, “Saison” (es wird nur das kleinere Jahr einer Saison angegeben; z.B wird die Saison 2021/22 unter 2021 abgespeichert), “Team”, “Marktwert”, “Platzierung” und “Punkte”. Beschafft wurden diese Variablen, indem sie mithilfe eines Pythonskripts von der Webiste “transfermarkt.de” gecrawlt wurden. Ziel des Projektes ist es die Variablen “Platzierung” und “Marktwert” bezüglich der Saison 2021/22 mithilfe eines Chi-Quadrat-Tests basierend auf der Resampling-Methode auf Unabhängigkeit zu untersuchen. Da diese zwei Variablen im Datensatz nicht kategorisch vorliegen und es somit schwierig wäre diese Variablen auf Unabhängigkeit zu untersuchen wurde die Methode `categorize_data()` bereitgestellt, die diese Variablen kategorisiert. Dabei wurden die Tabellenplatzierungen der Teams in die Kategorien “front” (Platz 1-6), “mid” (Platz 7-12) und “end” (Platz 13-18/20) eingeteilt. Der Marktwert der Teams wurde jeweils durch den Durchschnittsmarktwert der jeweiligen Liga des Teams in der Saison 2021/22 geteilt. Anschließend wurden die Teams anhand ihres Marktwertsquotienten in die Kategorien “high” (1.5,inf), “avg” (0.5,1.5) und “low” (0,0.5) überführt. Falls Unklarheiten zu `categorize_data()` oder anderen im Projekt enthaltenen Methoden besteht, ist es möglich mit dem “help”-Befehl die Dokumentation zu öffnen. Mit einem Python interpreter und den benötigten Modules kann mit den Methoden in “load_new_data.R” oder direkt über die Konsole ein eigener Datensatz von “www.transfermarkt.de” gecrawlet werden und einen Chi-Quadrat-Test mit diesen Daten durchführen. Beachte hierzu Bezeichnung von Team und Liga Namen auf der Website.

2. Installation

Das Paket kann direkt in R Studio mithilfe folgenden Befehls installiert werden:

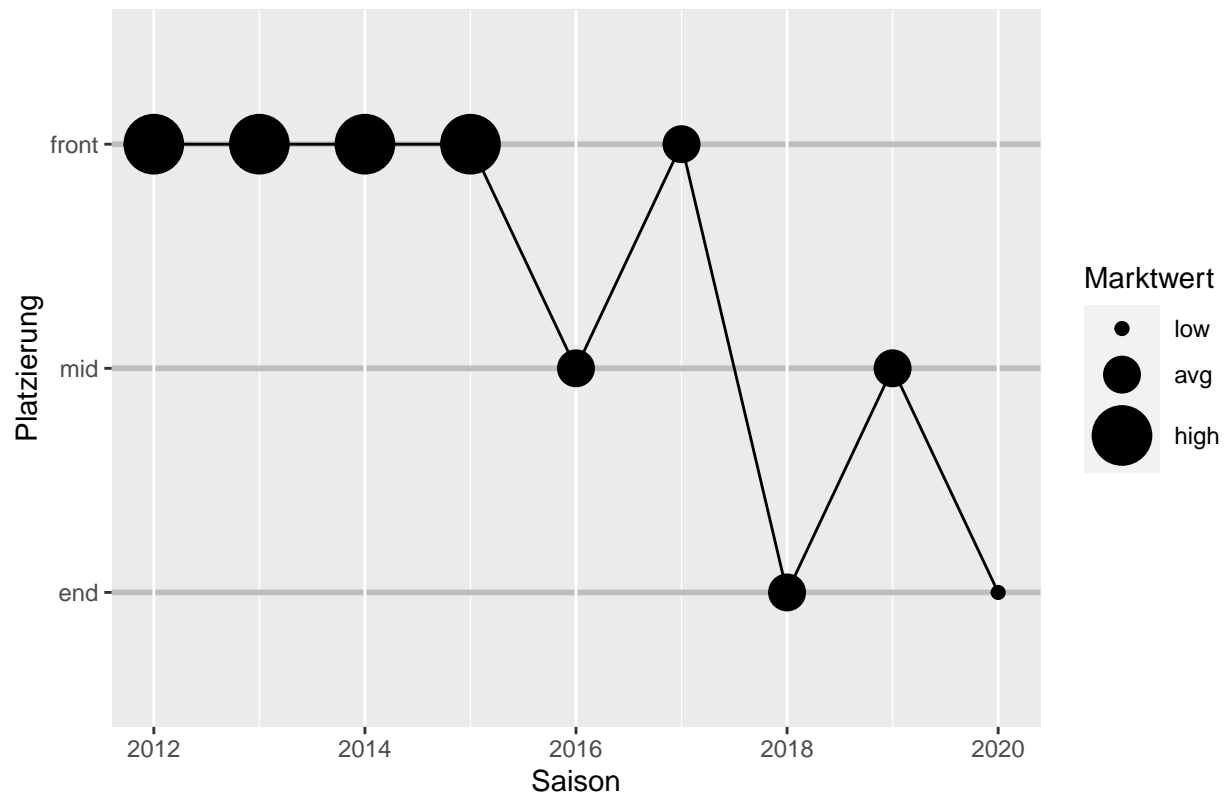
```
```r
remotes::install_gitlab("len1218/bundesliga_r_gruppe_38",host = "gitlab.lrz.de", auth_token = "glpat-jD
```
```

2. Explorative Analyse

Um den Datensatz zu genauer zu analysieren und sich ein Verständnis von den Daten zu verschaffen wurden mehrere graphische und numerische Funktionen erstellt. Dabei lässt sich oft ein Zusammenhang zwischen Marktwert und Platzierung der einzelnen Teams erahnen.

```
library(Bundesliga)
library(tidyverse)
plot_last10years(bigFive,"FC Schalke 04")
```

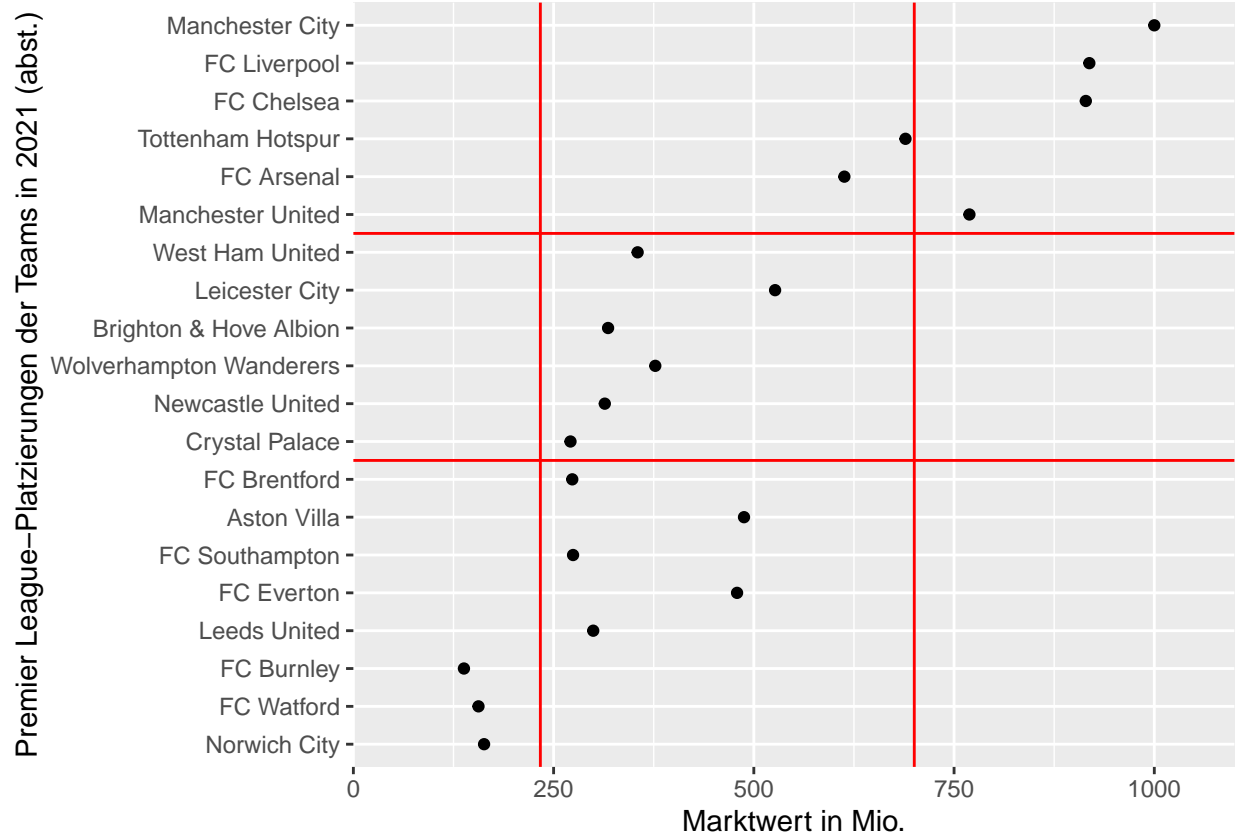
FC Schalke 04s Platzierungen der letzten 10 Jahre



Die Funktion “plot_last10years(bigFive,”FC Schalke 04”)” plottet die Saison-Endplatzierungen von “FC Schalke 04” über die vergangenen 10 Jahre (bzw. die Anzahl an Jahren in denen es in einer Top-5-Liga gespielt hat (hier also alle Saisons bis auf 2021/22)). Dabei ist es möglich zwischen einem kategorischen Plot der Variablen und einem nichtkategorischen Plot mithilfe der Variable “categoric” zu wählen. Dieser Beispielplot legt einen direkten Zusammenhang zwischen den im Test zu untersuchenden Variablen Marktwert und Platzierung nahe. Dennoch darf die Aussage dieses Graphen nicht überbewertet werden, da die Variablen hier in diesem Beispiel abhängig voneinander sind.

“plot_oneleague()” dagegen plottet den Marktwert und die Endplatzierung einer bestimmten Saison aus einer Liga des Datensatzes (also von unabhängigen Variablen).

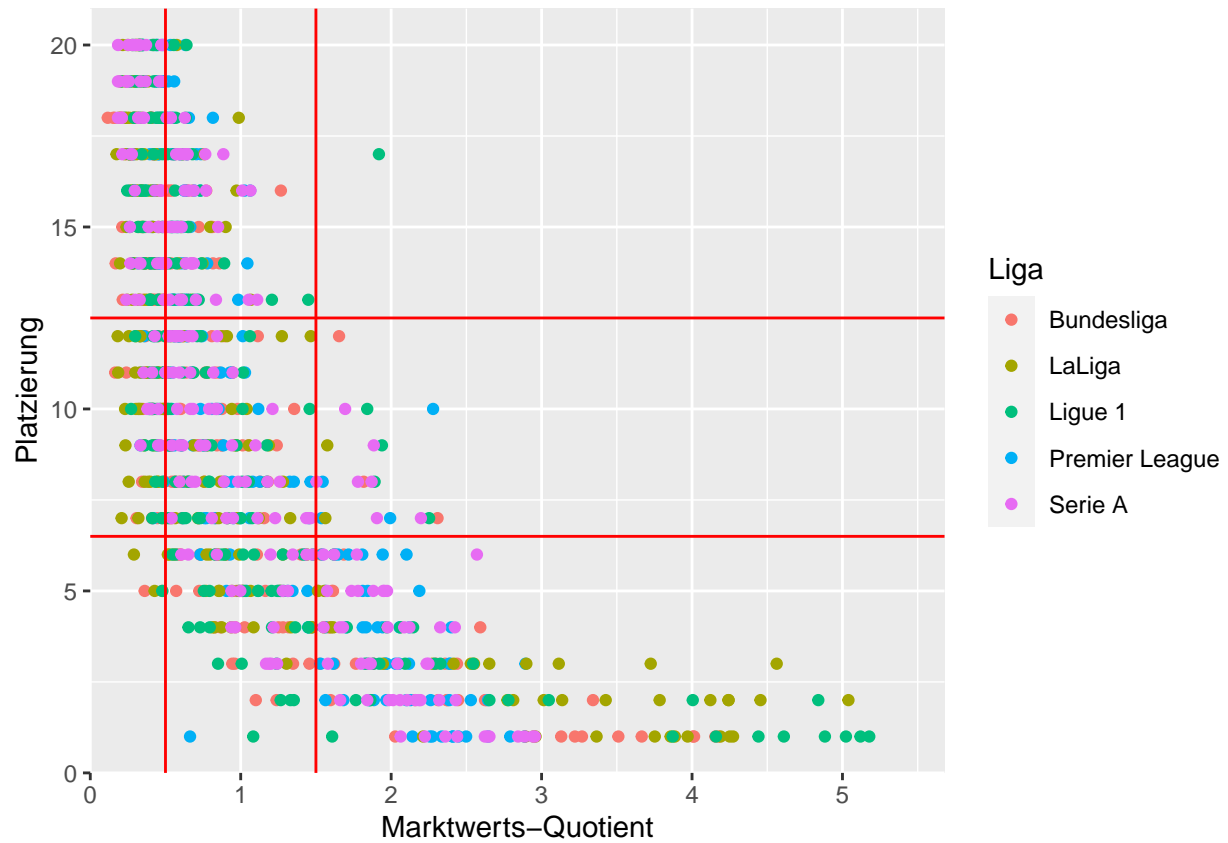
```
plot_oneleague(data=bigFive,liga="Premier League",saison=2021)
```



Bei den im Plot zu erkennenden roten Linien handelt es sich um die Trenngrenzen der einzelnen Kategorien. Im rechten, oberen Quadranten befinden sich also alle Teams der Premier League-Saison 2021/22 die eine Endplatzierung der Kategorie “front” erreicht haben und zugleich einen “high”-Marktwert besitzen. Auch dieser Plot legt einen Zusammenhang zwischen Marktwert und Platzierung nahe.

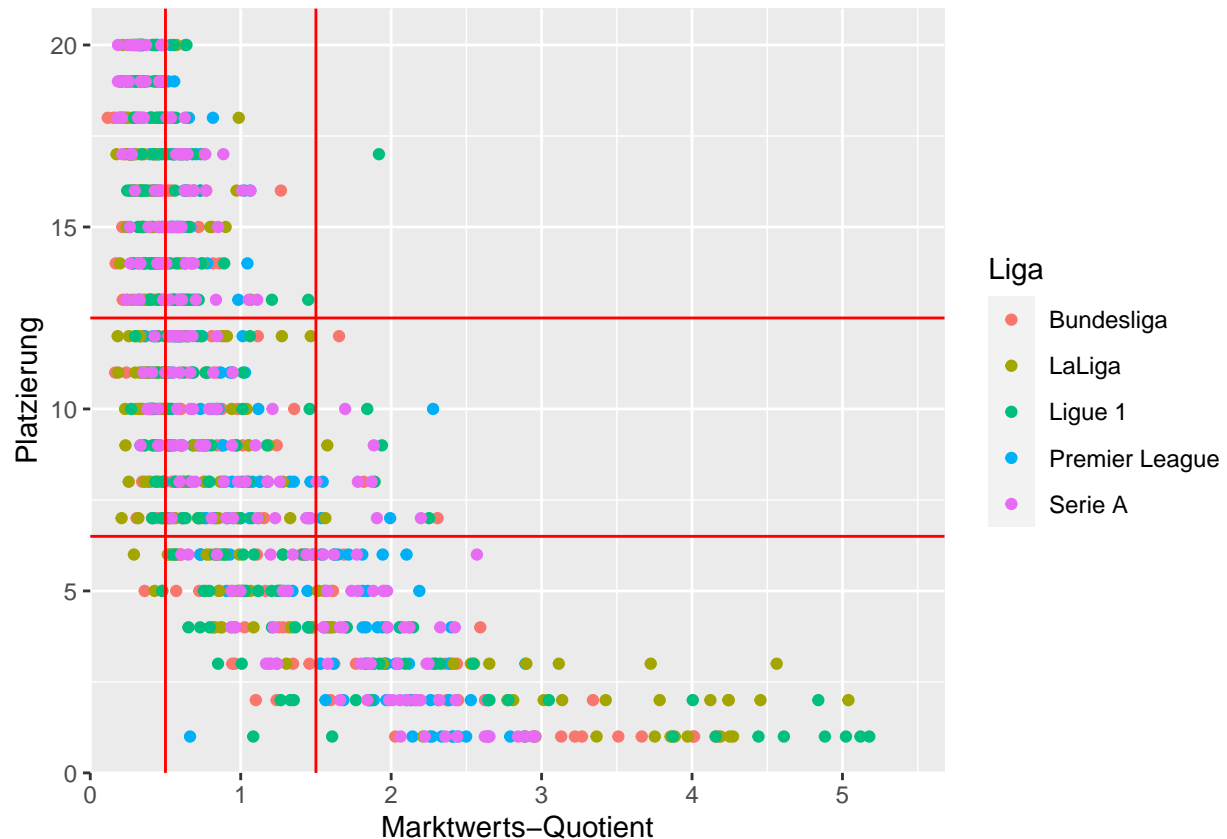
Ein ähnlicher, aber weitaus größerer Plot wird durch “plot_data_placements()” erzeugt. Hier wird für alle im übergebenen Datensatz enthaltenen Reihen die Platzierung gegen den Marktwertsquotient aufgetragen:

```
plot_data_placements(data=bigFive)
```



Zur besseren Übersichtlichkeit wird mit “plot_data_points” noch eine Funktion bereitgestellt die die Punk-
tanzahl gegen den Marktwertsquotienten plottet:

```
plot_data_placements(data=bigFive)
```



Auch diese Plots unterstützen die Aussagen der anderen Plots, die einen Zusammenhang zwischen Marktwert und Platzierung nahe legen.

Zuletzt wurden mit “filter_data()” und frame_performance() zwei numerische Methoden bereitgestellt mit den man “BigFive” durchforsten kann. Mit “filter_data()” ist es möglich den zu analysierenden Datensatz nach Saisons, Teams, Ligen, Punkten und Marktwerten zu filtern und den so produzierten neuen Datensatz beispielsweise mit “plot_data_placements()” zu plotten:

```
filter_data(data=bigFive, saison_von=2015, saison_bis=2019, teams=c("FC Augsburg"))
```

```
## # A tibble: 5 x 6
##   Liga    Saison Team      Marktwert Platzierung Punkte
##   <chr>    <dbl> <chr>      <dbl>         <dbl>   <dbl>
## 1 Bundesliga 2015 FC Augsburg 75930000         12     38
## 2 Bundesliga 2016 FC Augsburg 65850000         13     38
## 3 Bundesliga 2017 FC Augsburg 100450000        12     41
## 4 Bundesliga 2018 FC Augsburg 139250000        15     32
## 5 Bundesliga 2019 FC Augsburg 112950000        15     36
```

Für genauere Informationen gerne “?filter_data” aufrufen! Mit “frame_performance()” ist es möglich einen Datensatz bezüglich einer bestimmten Leistung zu filtern. Dabei kann von “überdurchschnittlich schlecht” (-2) über “schlecht” (-1) und “Durchschnitt” (0) bis zu “überdurchschnittlich gut” (2) gefiltert werden:

```
frame_performance(data=bigFive,-2)
```

```
## # A tibble: 1 x 6
##   Liga    Saison Team      Marktwert Platzierung Punkte
##   <chr>    <dbl> <chr>      <fct>         <fct>   <fct>
## 1 Ligue 1 2018 AS Monaco high      end      end
```

Die AS Monaco ist also das einzige Team aus dem Datensatz “bigFive”, das mit einem “high”-Kaderwert eine “end”-Platzierung erreichte. Auch diese Tatsache untermauert eine Korrelation zwischen den zu untersuchenden Variablen.

Induktive Analyse

Um nun letztendlich die Abhängigkeit tatsächlich zu überprüfen, haben wir einen Chi-Quadrat Unabhängigkeitstest durchgeführt. Dieser kann wahlweise mit einem theoretischen Ansatz oder durch resampling durchgeführt werden.

Konkret bedeutet dies folgende Hypothese:

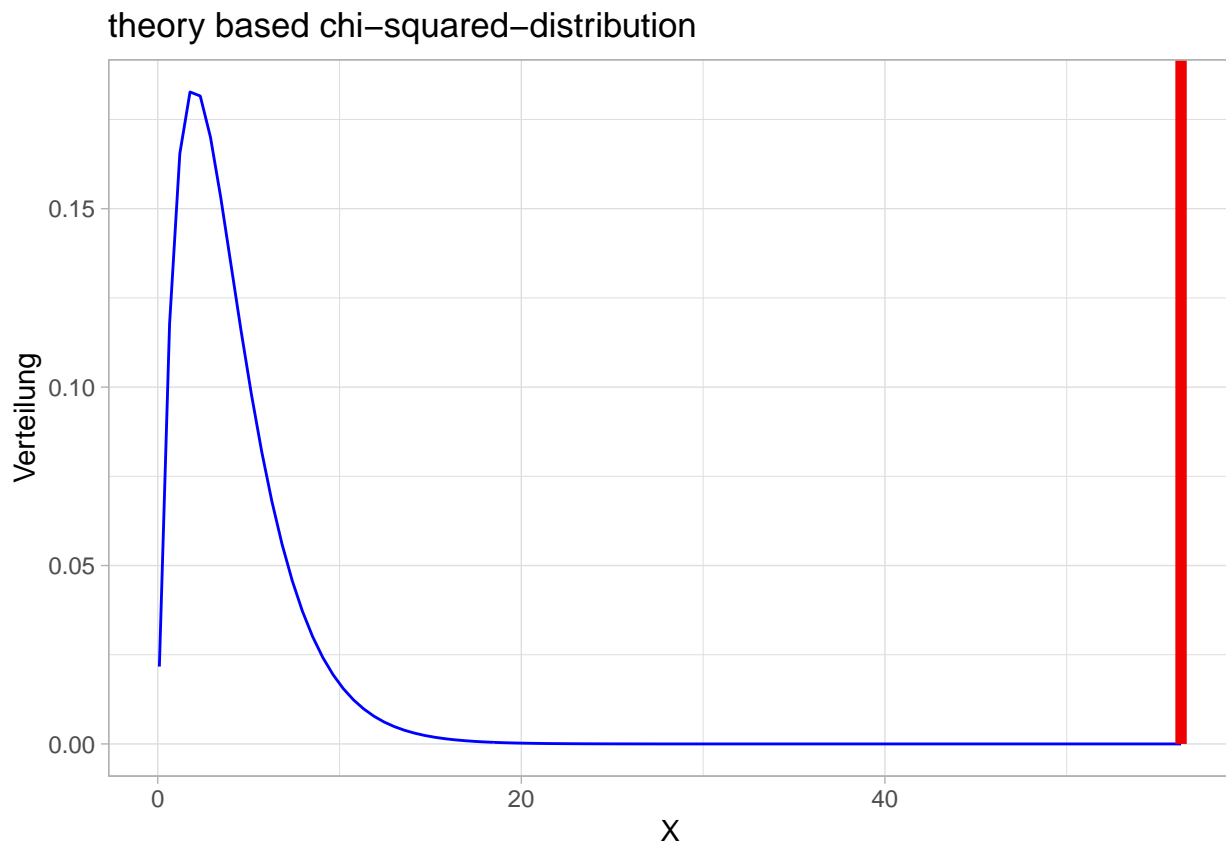
H_0 : \text{Die Zufallsvariablen Marktwert und Platzierung sind unabhängig.} \quad H_1: \text{Die Zufalls}

Die Chi-Quadrat Statistik nimmt bei grosser Abhängigkeit ebenfalls grosse Werte an und bei perfekter Unabhängigkeit dementsprechend den Wert 0.

Mithilfe der Funktion `Chi_squared_test()`, kann eine Chi-Quadrat Verteilung geplottet werden und der dazugehörige p-Wert berechnet werden. Hierzu wird mithilfe des Parameters ‘method’ eine passende Methode zur Berechnung oder Approximation der Chi-Quadrat Verteilung bestimmt werden.

```
filtered_data <- filter_data(bigFive, saison_von = 2021)
chi_squared_test(filtered_data, "theory")
```

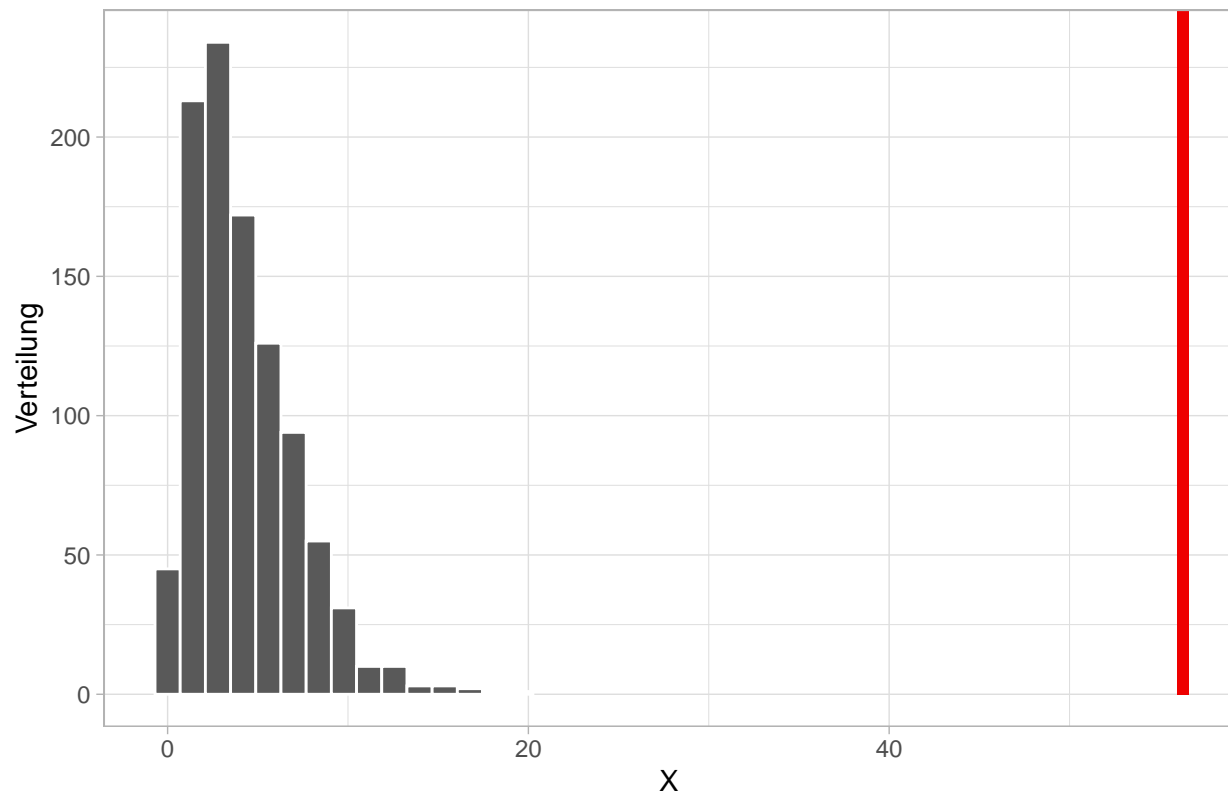
```
## # A tibble: 1 x 1
##   p_value
##   <dbl>
## 1 1.74e-11
```



```
chi_squared_test(filtered_data,"simulation",1000)
```

```
## # A tibble: 1 x 1
##   p_value
##   <dbl>
## 1      0
```

simulation based chi-squared-distribution



Zu sehen sind jeweils die theoretische Chi-Quadrat-Verteilung (blau) , die Balken, als Häufigkeiten der Intervalle der Zufallsvariable X (Chi-Quadrat verteilt) und die rote Linie als beobachteter Wert der Chi-Quadrat Statistik.