Übung Extremwerte raushauen

Sebastian Sauer

2022-06-24

Hallo, R! Wie geht’s! Mensch, immer diese STatistik. Wer hat sich das eigentlich ausgedacht? Hätte ich gewusst, dass es so viel Statistik gibt im Studium

Pakete starten:

library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 ──

## ✔ ggplot2 3.3.6 ✔ purrr 0.3.4  
## ✔ tibble 3.1.7 ✔ dplyr 1.0.9  
## ✔ tidyr 1.2.0 ✔ stringr 1.4.0  
## ✔ readr 2.1.2 ✔ forcats 0.5.1

## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

Daten laden:

data("flights", package = "nycflights13")

Was ist der Flug mit der größten Verspätung?

flights %>%   
 arrange(-dep\_delay)

## # A tibble: 336,776 × 19  
## year month day dep\_time sched\_dep\_time dep\_delay arr\_time sched\_arr\_time  
## <int> <int> <int> <int> <int> <dbl> <int> <int>  
## 1 2013 1 9 641 900 1301 1242 1530  
## 2 2013 6 15 1432 1935 1137 1607 2120  
## 3 2013 1 10 1121 1635 1126 1239 1810  
## 4 2013 9 20 1139 1845 1014 1457 2210  
## 5 2013 7 22 845 1600 1005 1044 1815  
## 6 2013 4 10 1100 1900 960 1342 2211  
## 7 2013 3 17 2321 810 911 135 1020  
## 8 2013 6 27 959 1900 899 1236 2226  
## 9 2013 7 22 2257 759 898 121 1026  
## 10 2013 12 5 756 1700 896 1058 2020  
## # … with 336,766 more rows, and 11 more variables: arr\_delay <dbl>,  
## # carrier <chr>, flight <int>, tailnum <chr>, origin <chr>, dest <chr>,  
## # air\_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>, time\_hour <dttm>

Sie könnten synonym auch schreiben:

arrange(flights, -dep\_delay)

## # A tibble: 336,776 × 19  
## year month day dep\_time sched\_dep\_time dep\_delay arr\_time sched\_arr\_time  
## <int> <int> <int> <int> <int> <dbl> <int> <int>  
## 1 2013 1 9 641 900 1301 1242 1530  
## 2 2013 6 15 1432 1935 1137 1607 2120  
## 3 2013 1 10 1121 1635 1126 1239 1810  
## 4 2013 9 20 1139 1845 1014 1457 2210  
## 5 2013 7 22 845 1600 1005 1044 1815  
## 6 2013 4 10 1100 1900 960 1342 2211  
## 7 2013 3 17 2321 810 911 135 1020  
## 8 2013 6 27 959 1900 899 1236 2226  
## 9 2013 7 22 2257 759 898 121 1026  
## 10 2013 12 5 756 1700 896 1058 2020  
## # … with 336,766 more rows, and 11 more variables: arr\_delay <dbl>,  
## # carrier <chr>, flight <int>, tailnum <chr>, origin <chr>, dest <chr>,  
## # air\_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>, time\_hour <dttm>

Synonym:

flights %>%   
 drop\_na(dep\_delay) %>%   
 filter(dep\_delay == max(dep\_delay))

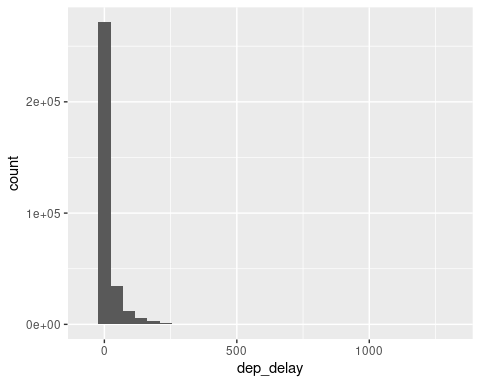
## # A tibble: 1 × 19  
## year month day dep\_time sched\_dep\_time dep\_delay arr\_time sched\_arr\_time  
## <int> <int> <int> <int> <int> <dbl> <int> <int>  
## 1 2013 1 9 641 900 1301 1242 1530  
## # … with 11 more variables: arr\_delay <dbl>, carrier <chr>, flight <int>,  
## # tailnum <chr>, origin <chr>, dest <chr>, air\_time <dbl>, distance <dbl>,  
## # hour <dbl>, minute <dbl>, time\_hour <dttm>

Visualisieren Sie die Verteilung von dep\_delay und entscheiden Sie, ob Extremwerte vorkommen, die besonderer Behandlung bedürfen!

flights %>%   
 ggplot() +  
 aes(x = dep\_delay) +  
 geom\_histogram()

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 8255 rows containing non-finite values (stat\_bin).



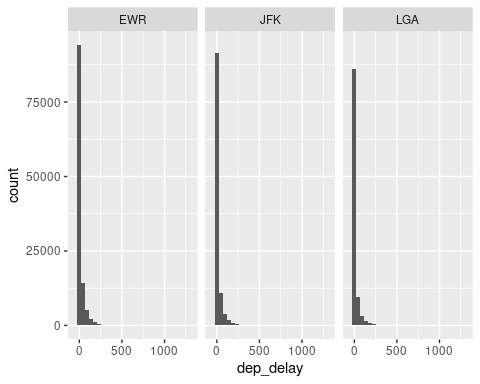
Unterscheiden sich die New Yorker Flughäfen in ihrer (Verteilung) der Verspätung?

Um diese Frage zu beantworten, kann man deskriptive Statistik oder Visualisierung(en) bemühen.

flights %>%   
 ggplot() +  
 aes(x = dep\_delay) +  
 geom\_histogram() +  
 facet\_wrap(~ origin)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

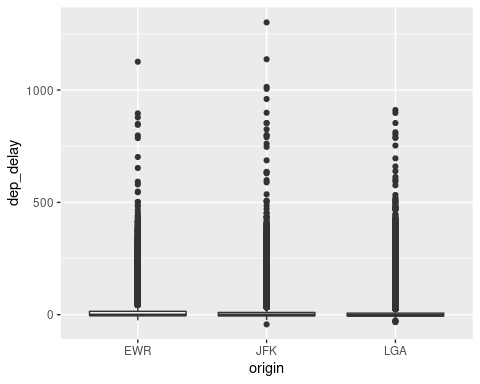
## Warning: Removed 8255 rows containing non-finite values (stat\_bin).



Vielleicht besser mit Boxplots:

flights %>%   
 ggplot() +  
 aes(x = origin, y = dep\_delay) +  
 geom\_boxplot()

## Warning: Removed 8255 rows containing non-finite values (stat\_boxplot).



Geben Sie die typischen deskriptiven Statistiken für dep\_delay an, am besten runtergebrochen pro Abflugsort (origin).

summary(flights)

## year month day dep\_time sched\_dep\_time  
## Min. :2013 Min. : 1.000 Min. : 1.00 Min. : 1 Min. : 106   
## 1st Qu.:2013 1st Qu.: 4.000 1st Qu.: 8.00 1st Qu.: 907 1st Qu.: 906   
## Median :2013 Median : 7.000 Median :16.00 Median :1401 Median :1359   
## Mean :2013 Mean : 6.549 Mean :15.71 Mean :1349 Mean :1344   
## 3rd Qu.:2013 3rd Qu.:10.000 3rd Qu.:23.00 3rd Qu.:1744 3rd Qu.:1729   
## Max. :2013 Max. :12.000 Max. :31.00 Max. :2400 Max. :2359   
## NA's :8255   
## dep\_delay arr\_time sched\_arr\_time arr\_delay   
## Min. : -43.00 Min. : 1 Min. : 1 Min. : -86.000   
## 1st Qu.: -5.00 1st Qu.:1104 1st Qu.:1124 1st Qu.: -17.000   
## Median : -2.00 Median :1535 Median :1556 Median : -5.000   
## Mean : 12.64 Mean :1502 Mean :1536 Mean : 6.895   
## 3rd Qu.: 11.00 3rd Qu.:1940 3rd Qu.:1945 3rd Qu.: 14.000   
## Max. :1301.00 Max. :2400 Max. :2359 Max. :1272.000   
## NA's :8255 NA's :8713 NA's :9430   
## carrier flight tailnum origin   
## Length:336776 Min. : 1 Length:336776 Length:336776   
## Class :character 1st Qu.: 553 Class :character Class :character   
## Mode :character Median :1496 Mode :character Mode :character   
## Mean :1972   
## 3rd Qu.:3465   
## Max. :8500   
##   
## dest air\_time distance hour   
## Length:336776 Min. : 20.0 Min. : 17 Min. : 1.00   
## Class :character 1st Qu.: 82.0 1st Qu.: 502 1st Qu.: 9.00   
## Mode :character Median :129.0 Median : 872 Median :13.00   
## Mean :150.7 Mean :1040 Mean :13.18   
## 3rd Qu.:192.0 3rd Qu.:1389 3rd Qu.:17.00   
## Max. :695.0 Max. :4983 Max. :23.00   
## NA's :9430   
## minute time\_hour   
## Min. : 0.00 Min. :2013-01-01 05:00:00.00   
## 1st Qu.: 8.00 1st Qu.:2013-04-04 13:00:00.00   
## Median :29.00 Median :2013-07-03 10:00:00.00   
## Mean :26.23 Mean :2013-07-03 05:22:54.64   
## 3rd Qu.:44.00 3rd Qu.:2013-10-01 07:00:00.00   
## Max. :59.00 Max. :2013-12-31 23:00:00.00   
##

Oder so:

flights %>%   
 drop\_na(origin, dep\_delay) %>%   
 group\_by(origin) %>%   
 summarise(median(dep\_delay),  
 IQR(dep\_delay),  
 n())

## # A tibble: 3 × 4  
## origin `median(dep\_delay)` `IQR(dep\_delay)` `n()`  
## <chr> <dbl> <dbl> <int>  
## 1 EWR -1 19 117596  
## 2 JFK -1 15 109416  
## 3 LGA -3 13 101509

Nehmen wir mal eine Convenience-Funktion:

library(easystats)

## # Attaching packages: easystats 0.4.3 (red = needs update)  
## ✖ insight 0.17.0 ✖ datawizard 0.4.0   
## ✔ bayestestR 0.12.1 ✖ performance 0.9.0   
## ✖ parameters 0.17.0 ✖ effectsize 0.6.0.1  
## ✖ modelbased 0.8.0 ✖ correlation 0.8.0   
## ✖ see 0.7.0 ✔ report 0.5.1.1  
##   
## Restart the R-Session and update packages in red with 'easystats::easystats\_update()'.

Achtug! easystats kann man nicht wie üblich installieren.

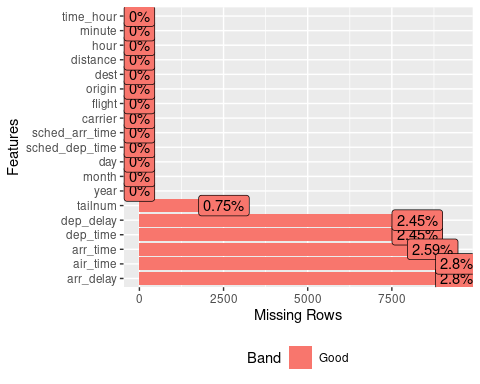
flights %>%   
 select(origin, dep\_delay) %>%   
 group\_by(origin) %>%   
 describe\_distribution()

## # origin=EWR  
##   
## Variable | Mean | SD | IQR | Range | Skewness | Kurtosis | n | n\_Missing  
## ----------------------------------------------------------------------------------------------  
## dep\_delay | 15.11 | 41.32 | 19 | [-25.00, 1126.00] | 4.15 | 30.97 | 117596 | 3239  
##   
## # origin=JFK  
##   
## Variable | Mean | SD | IQR | Range | Skewness | Kurtosis | n | n\_Missing  
## ----------------------------------------------------------------------------------------------  
## dep\_delay | 12.11 | 39.04 | 15 | [-43.00, 1301.00] | 5.45 | 64.44 | 109416 | 1863  
##   
## # origin=LGA  
##   
## Variable | Mean | SD | IQR | Range | Skewness | Kurtosis | n | n\_Missing  
## ---------------------------------------------------------------------------------------------  
## dep\_delay | 10.35 | 39.99 | 13 | [-33.00, 911.00] | 5.02 | 41.52 | 101509 | 3153

Das Paket DataExplorer ist auch ganz cool. Können Sie ja mal ausprobieren.

library(DataExplorer)

plot\_missing(flights)



[Hier](https://cran.r-project.org/web/packages/DataExplorer/vignettes/dataexplorer-intro.html) findet sich eine Einführung in das Paket.

Wie wiele Zeilen ohne fehlende Werte gibt es im Datensatz (absolut und anteilig)?

flights %>%   
 drop\_na() %>%   
 nrow()

## [1] 327346

Identifizieren Sie alle Flüge, deren Verspätung (Abflug) mehr als 3 SD über dem Mittelwert (aller Flüge) liegt!