local bakery data analysis

Yufenyuy

2025-07-17

```
#setwd("C:/gitrepos/ranalytics/r_data_analysis")
```

Hier werden R-Packete für die Daten Manipulation bzw. Auswertung beladen

```
library(conflicted)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
```

```
library(DBI)
```

```
## Warning: Paket 'DBI' wurde unter R Version 4.4.3 erstellt
```

```
library(RPostgres)
```

```
## Warning: Paket 'RPostgres' wurde unter R Version 4.4.3 erstellt
```

```
library(ggplot2)
library(lubridate)
library(dplyr)
library(tidyr)
library(stringr)
```

Erstellte Umgebung Variablen werden für die Datenbank Verbindung eingelesen

```
con <- dbConnect(
  Postgres(),
  dbname = Sys.getenv("DBNAME"),
  host = Sys.getenv("HOST"),
  user = Sys.getenv("USER"),
  password = Sys.getenv("PASSWORD"),
  port = Sys.getenv("PORT")
)
```

Die gezielte Daten Tabelle liegt in einer spezifisches Schema in PostgreSQL. Die Tabellen enhalten in diesem Schema werden aufgelisted und geprüft, ob die gewünschte Tabelle drin liegt

```
schema_name <- "baker_yelp_dbt_prod"

# SQL query to list tables in the given schema
tables <- dbGetQuery(con, paste0("
```

```
  SELECT table_name
  FROM information_schema.tables
  WHERE table_schema = '", schema_name, "'
    AND table_type = 'BASE TABLE';
"))

print(tables)
```

```
##                         table_name
## 1                        pdtn_fuel
## 2                monthly_ts_issales
## 3           weekly_product_pdtn_ts
## 4                          sales_t1
## 5           monthly_ts_should_sales
## 6                products_weekly_ts
## 7                           pdtn_t1
## 8                           feeding
## 9                          sales_t2
## 10             business_reviews_t1
## 11       weekly_timeseries_issales
## 12                    pdtn_oldstuck
## 13             business_reviews_t2
## 14       weekly_ts_is_should_sales
## 15                    businesses_t1
## 16          weekly_ts_should_sales
## 17                    businesses_t2
## 18 daily_expected_production_dates
## 19          daily_product_pdtn_amt
## 20                    businesses_t3
## 21      monthly_ts_is_should_sales
## 22                           items
## 23                       datetable
```

Hier werden die Daten mittels SQL selektiert

```
product_ts <- dbGetQuery(con, "SELECT * FROM baker_yelp_dbt_prod.products_weekly_ts")
```

Datentypen und Werte einsehen

```
str(product_ts)
```

```
## 'data.frame':    425 obs. of  15 variables:
##  $ endofweek     : Date, format: "2023-07-09" "2023-07-02" ...
##  $ banana50_amt  : num  9 3 0 0 0 0 0 0 0 0 ...
##  $ square50_amt  : num  148 142 124 136 148 ...
##  $ local50_amt   : num  7.5 12.5 17.5 17.5 22.5 22.5 25 15 30 30 ...
##  $ banana100_amt : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ local100_amt  : num  190 185 177 201 190 ...
##  $ special100_amt: num  675 1732 1674 1752 1657 ...
##  $ special150_amt: num  1319 128 149 158 163 ...
##  $ local200_amt  : num  253 244 250 234 269 ...
##  $ special200_amt: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ local250_amt  : num  50.4 50.4 42 50.4 49 42 40.6 33.6 42 50.4 ...
##  $ local300_amt  : num  569 519 524 580 543 ...
##  $ special400_amt: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ special500_amt: num  706 624 623 697 630 ...
```

```
##  $ special800_amt: num  60.5 21.6 32.7 68.7 44.1 ...
```

Statistik der Daten einsehen, um die Zentrale Tendenz und die Streuung der Daten zu verstehen.

Beobachtungen:

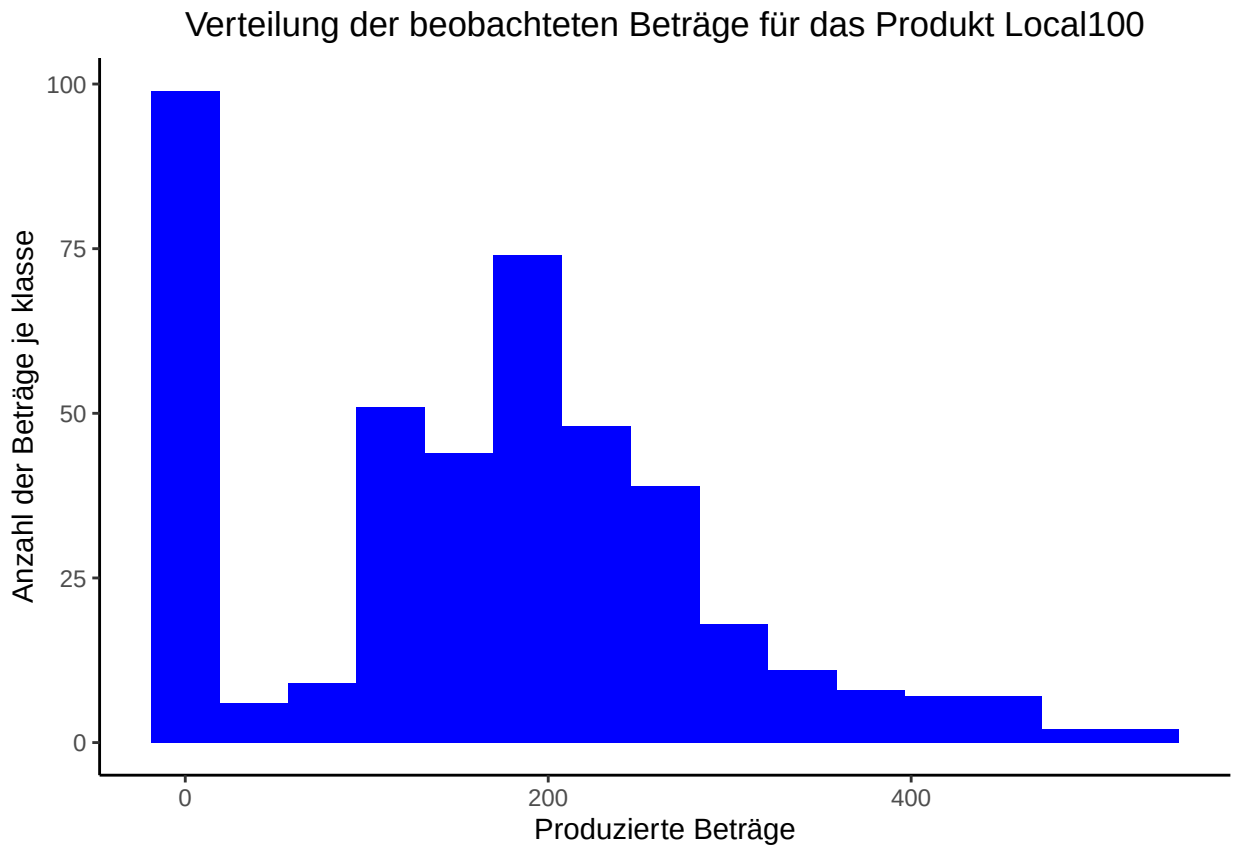
- Die Daten wurden von mitte 2015 bis mitte 2023 gesammelt.
- Bei viele Produkte(spezial150_amt) liegt der mittleren Wert bei 0. D.H Bis zu 50% ihre Beträge liegen bei 0 Geld Einheiten(GE).
- Der minimale produzierte Betrag bei alle Produkte liegt bei 0 GE.
- Der höchste durchschnittliche produzierte Betrag über den ganzen Zeitraum ist bei dem Produkt special200_amt beobachtet und dasselbe Produkt hat der höchste produzierte Betrag unter alle Produkten.Allerding liegt sein Mittelwert bei 0 GE.
- Bei der Produkte local100_amt, local300_amt und special100_amt liegen der Mittelwert und der Durchschnitt nah beieinander.Diese weißt auf einer quasi Normal Verteilung hin spricht die Daten sind quasi gleichverteilt.

```r
summary(product_ts)
```

```
##    endofweek           banana50_amt    square50_amt     local50_amt
##  Min.   :2015-05-24   Min.   :  0.0   Min.   :  0.00   Min.   : 0.000
##  1st Qu.:2017-06-04   1st Qu.:  0.0   1st Qu.:  0.00   1st Qu.: 0.000
##  Median :2019-06-16   Median : 48.0   Median :  0.00   Median : 0.000
##  Mean   :2019-06-16   Mean   :135.8   Mean   : 66.88   Mean   : 1.688
##  3rd Qu.:2021-06-27   3rd Qu.:276.0   3rd Qu.:135.00   3rd Qu.: 0.000
##  Max.   :2023-07-09   Max.   :542.4   Max.   :450.00   Max.   :40.000
##  banana100_amt      local100_amt   special100_amt   special150_amt
##  Min.   :  0.000   Min.   :  0.0   Min.   :   0.0   Min.   :   0.0
##  1st Qu.:  0.000   1st Qu.: 66.3   1st Qu.: 196.3   1st Qu.:   0.0
##  Median :  0.000   Median :170.0   Median : 973.5   Median :   0.0
##  Mean   :  7.916   Mean   :161.9   Mean   : 889.5   Mean   : 164.8
##  3rd Qu.:  0.000   3rd Qu.:238.0   3rd Qu.:1440.5   3rd Qu.: 301.0
##  Max.   :107.500   Max.   :528.7   Max.   :2352.9   Max.   :1318.6
##   local200_amt   special200_amt     local250_amt    local300_amt
##  Min.   :  0.0   Min.   :  0.000   Min.   :  0.0   Min.   :  0.0
##  1st Qu.:  0.0   1st Qu.:  0.000   1st Qu.: 15.4   1st Qu.: 72.1
##  Median : 93.2   Median :  0.000   Median : 56.0   Median :349.3
##  Mean   :113.5   Mean   :  8.179   Mean   :143.2   Mean   :319.5
##  3rd Qu.:231.6   3rd Qu.:  0.000   3rd Qu.:237.3   3rd Qu.:518.7
##  Max.   :351.6   Max.   :134.940   Max.   :607.6   Max.   :777.7
##  special400_amt   special500_amt   special800_amt
##  Min.   :  0.00   Min.   :   0.0   Min.   :  0.00
##  1st Qu.:  0.00   1st Qu.: 100.4   1st Qu.:  2.40
##  Median :  0.00   Median : 435.2   Median : 19.20
##  Mean   : 13.17   Mean   : 384.1   Mean   : 25.08
##  3rd Qu.:  0.00   3rd Qu.: 581.0   3rd Qu.: 38.40
##  Max.   :534.48   Max.   :1004.7   Max.   :110.40
```

```r
g1 <- product_ts %>%
  ggplot(aes(x = local100_amt)) +
  geom_histogram(fill = "blue", bins = 15) +
  xlab("Produzierte Beträge") +
  ylab("Anzahl der Beträge je klasse") +
  theme_classic() +
  ggtitle("Verteilung der beobachteten Beträge für das Produkt Local100") +
  theme(
```

```
    plot.title = element_text(hjust = 0.5)
  )
g1
```




```
product_ts <- product_ts %>% pivot_longer(cols = ends_with("amt"), names_to = "products", values_to = "
```

```
str(product_ts)
```

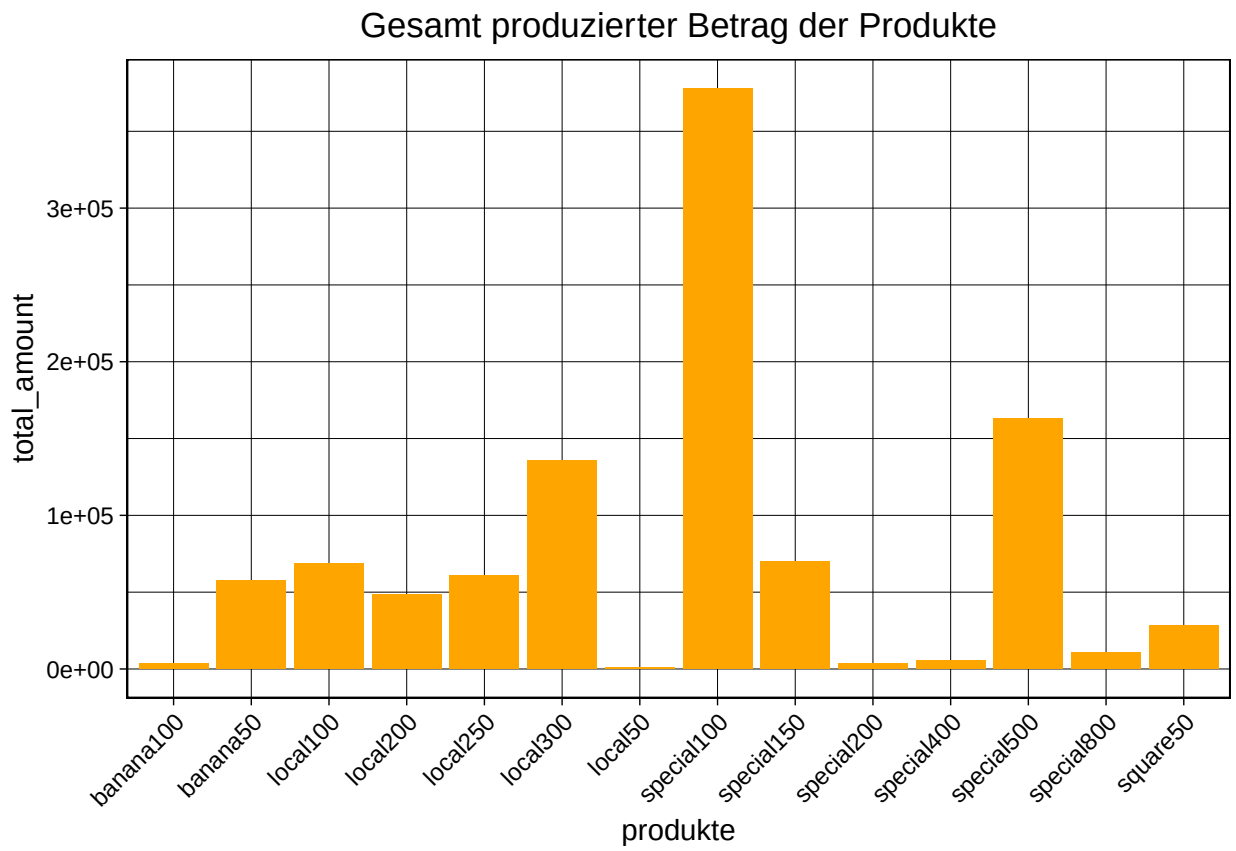
```
## tibble [5,950 x 3] (S3: tbl_df/tbl/data.frame)
##  $ endofweek    : Date[1:5950], format: "2023-07-09" "2023-07-09" ...
##  $ products     : chr [1:5950] "banana50_amt" "square50_amt" "local50_amt" "banana100_amt" ...
##  $ weekly_amount: num [1:5950] 9 148.5 7.5 0 190.4 ...
```

```
summary(product_ts)
```

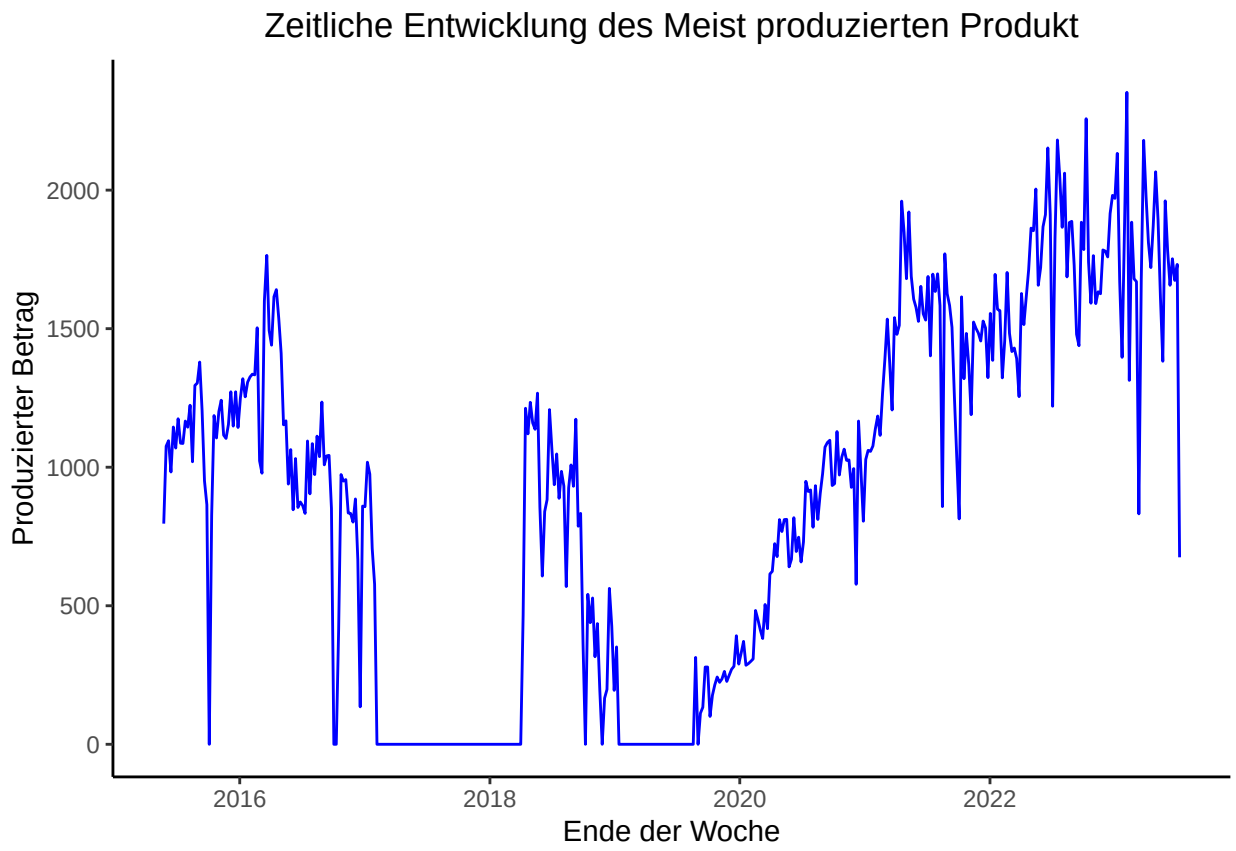
```
##    endofweek            products         weekly_amount
##  Min.   :2015-05-24   Length:5950        Min.   :   0.0
##  1st Qu.:2017-06-04   Class :character   1st Qu.:   0.0
##  Median :2019-06-16   Mode  :character   Median :  13.8
##  Mean   :2019-06-16                      Mean   : 174.0
##  3rd Qu.:2021-06-27                      3rd Qu.: 229.9
##  Max.   :2023-07-09                      Max.   :2352.9
```

```
gplot_1 <- product_ts %>%
  mutate(
    produkte = str_sub(products, 1, str_length(products) - 4)
    ) %>%
  group_by(produkte) %>%
```

```
  summarise(total_amount = sum(weekly_amount)) %>%
  ggplot(aes(x = produkte, y = total_amount)) +
  geom_col(fill = "orange") +
  ggtitle("Gesamt produzierter Betrag der Produkte") +
  theme_linedraw() +
  theme(
    plot.title = element_text(hjust = 0.5),
    axis.text.x = element_text(angle = 45, hjust = 1)
  )
gplot_1
```




```
gplot_2 <- dplyr::filter(product_ts, products == "special100_amt") %>%
  ggplot(aes(x = endofweek, y = weekly_amount)) +
  geom_line(color = "blue") +
  xlab("Ende der Woche") +
  ylab("Produzierter Betrag") +
  ggtitle("Zeitliche Entwicklung des Meist produzierten Produkt") +
  theme_classic() +
  theme(plot.title = element_text(hjust = 0.5))
gplot_2
```


Zeitliche Entwicklung des Meist produzierten Produkt
Produzierter Betrag
2000
1500
1000
500
0
2016
2018
2020
2022
Ende der Woche