

Case Study - Gruppen 105 & 201

Supply Chain Analytics SS23

Benjamin Grünwald, Cordelia Mena Hernandez, Daniel Glatter, Vinzenz Tom Andreas Schaak

2023-06-21

Phase 2: Datenbeschaffung/Einlesen

```
# Daten einlesen
materials <- read_excel("data/MBG_Materialverlauf_20230612.xlsx")

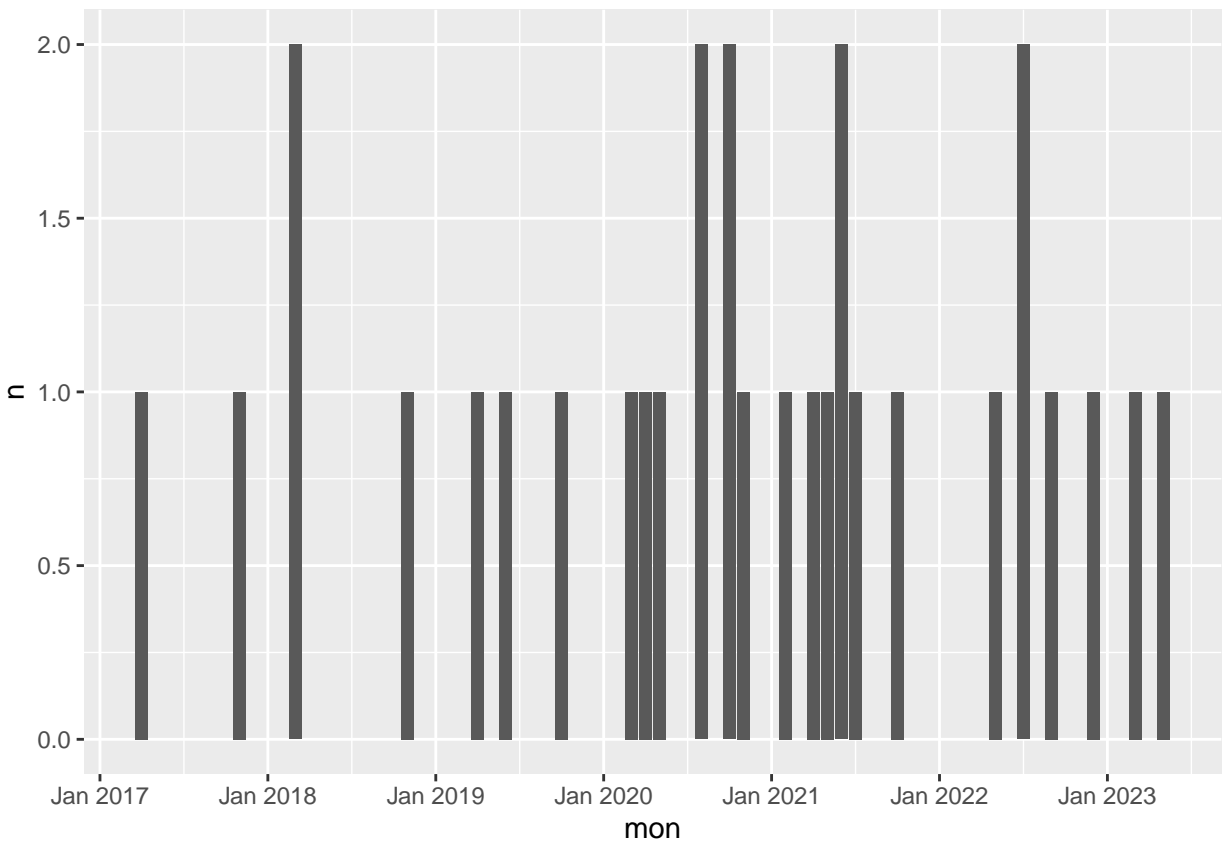
# Datentypen konvertieren
materials$ArtikelNr <- as.factor(materials$ArtikelNr)
materials$`Artikel-Bezeichnung` <- as.factor(materials$`Artikel-Bezeichnung`)
materials$EinAutDatTyp <- as.factor(materials$EinAutDatTyp)
materials$PjNr <- as.factor(materials$PjNr)
materials$PjInfo <- as.factor(materials$PjInfo)
materials$ORGAKz1 <- as.factor(materials$ORGAKz1)
materials$EinAusDat <- as.Date(materials$EinAusDat)
materials$BstDat <- as.Date(materials$BstDat)

# Klassifizierung der Transaktionen: Projekt, Handel, Einkauf - nach Keywords
projekt <- c('PV-Realisierung', 'PV: Realisierung', 'PV: Tankstelle', 'PV: ET', 'PV:
  ↳ EDEKA', 'PV: MBG', 'PV: MO', 'PV: SolarSun', 'PV-Anlage: Realisierung', 'PV: 15,39',
  ↳ 'PV: 175x', 'PV: 99,83', 'PV: EFH')
handel <- c('PV-Materialverkauf', 'PV: Materialverkauf', 'PV Materialverkauf')
einkauf <- c('Einkauf', 'Groß- und Einzelhandel der MBG')

materials <- materials %>%
  mutate(klasse=if_else(grepl(paste(projekt, collapse='|'), PjInfo), "Projekt",
    ↳ if_else(grepl(paste(handel, collapse='|'), PjInfo), "Handel",
    ↳ if_else(grepl(paste(einkauf, collapse='|'), PjInfo), "Einkauf", "Anderes"))))
materials$klasse <- as.factor(materials$klasse)

# Nur relevante Materialien für Gruppe B
materials_group_b <- c("DAS N-TOPcon Bifacial Glas-Glas (2+2mm) Black 415 WP",
  "Rahmenloser Bildhalter Antireflexglas 29,7 x 42",
  "Kabelbinder 280x4,5 mm schwarz",
  "FPKU-EM-F PVC- Rohr grau EN50 L=3m",
  "FINTECH Alu Steckrohr IESR 40 AL (3m)",
  "3d-U-Scheiben      8,4   M 8           DIN 9021   A2")
materials_b <- materials %>% filter(`Artikel-Bezeichnung` %in% materials_group_b)

materials_b %>% filter(`Artikel-Bezeichnung`=="3d-U-Scheiben      8,4   M 8           DIN
  ↳ 9021   A2", EinAutDatTyp=="geliefertAm") %>% group_by(mon=as.yearmon(EinAusDat)) %>%
  ↳ count() %>% ggplot(aes(x=mon, y=n))+geom_col()
```



Phase 3: Explorative Datenanalyse

Deskriptive Statistiken

```
# Anzahl Kunden
materials %>% filter(EinAutDatTyp != "geliefertAm") %>% select(ORGAKz1) %>% distinct()
↳ %>% count()
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1    287
```

```
# Anzahl Lieferanten
materials %>% filter(EinAutDatTyp == "geliefertAm") %>% select(ORGAKz1) %>% distinct()
↳ %>% count()
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1     69
```

```
# Anzahl Projekte
materials %>% select(PjInfo) %>% distinct() %>% count()
```

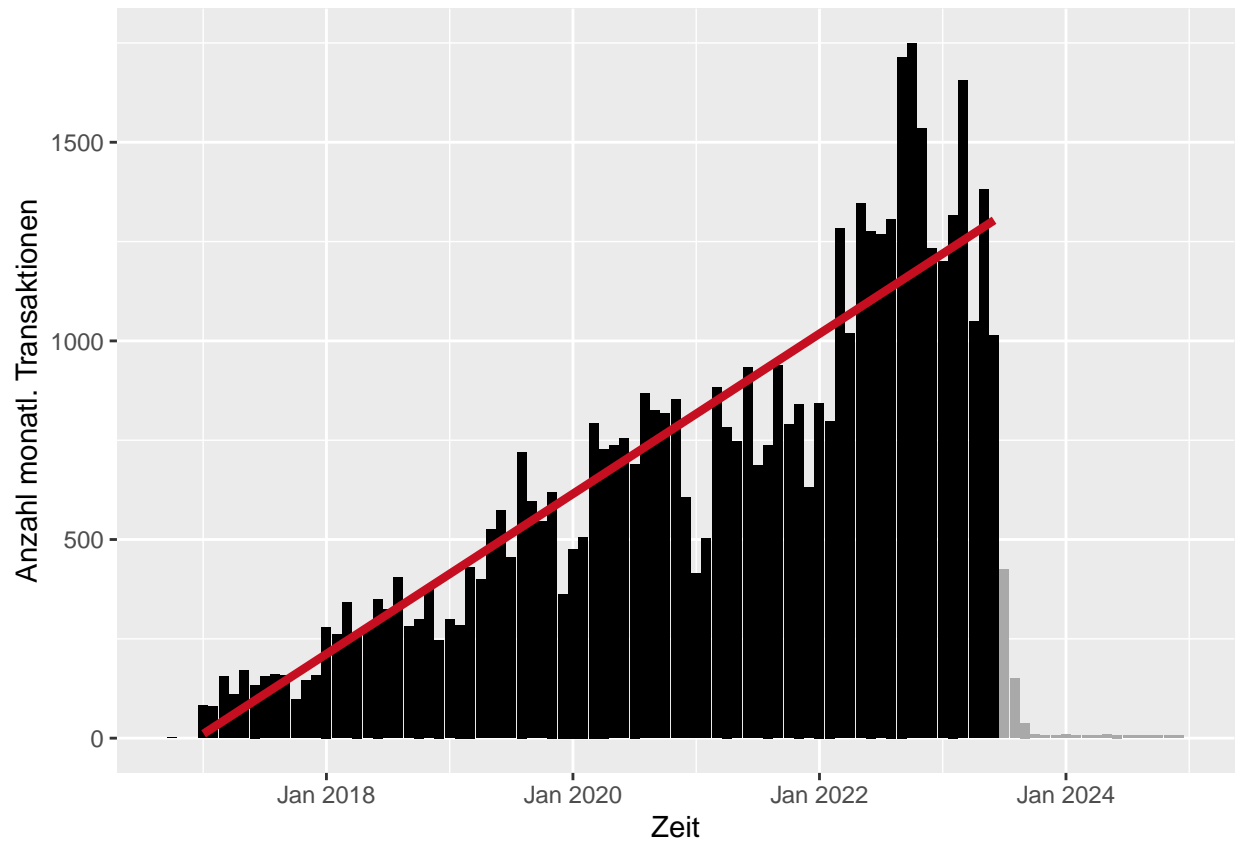
```
## # A tibble: 1 x 1
##       n
##   <int>
## 1   920
```

```
# Anzahl Materialien
materials %>% select(`Artikel-Bezeichnung`) %>% distinct() %>% count()
```

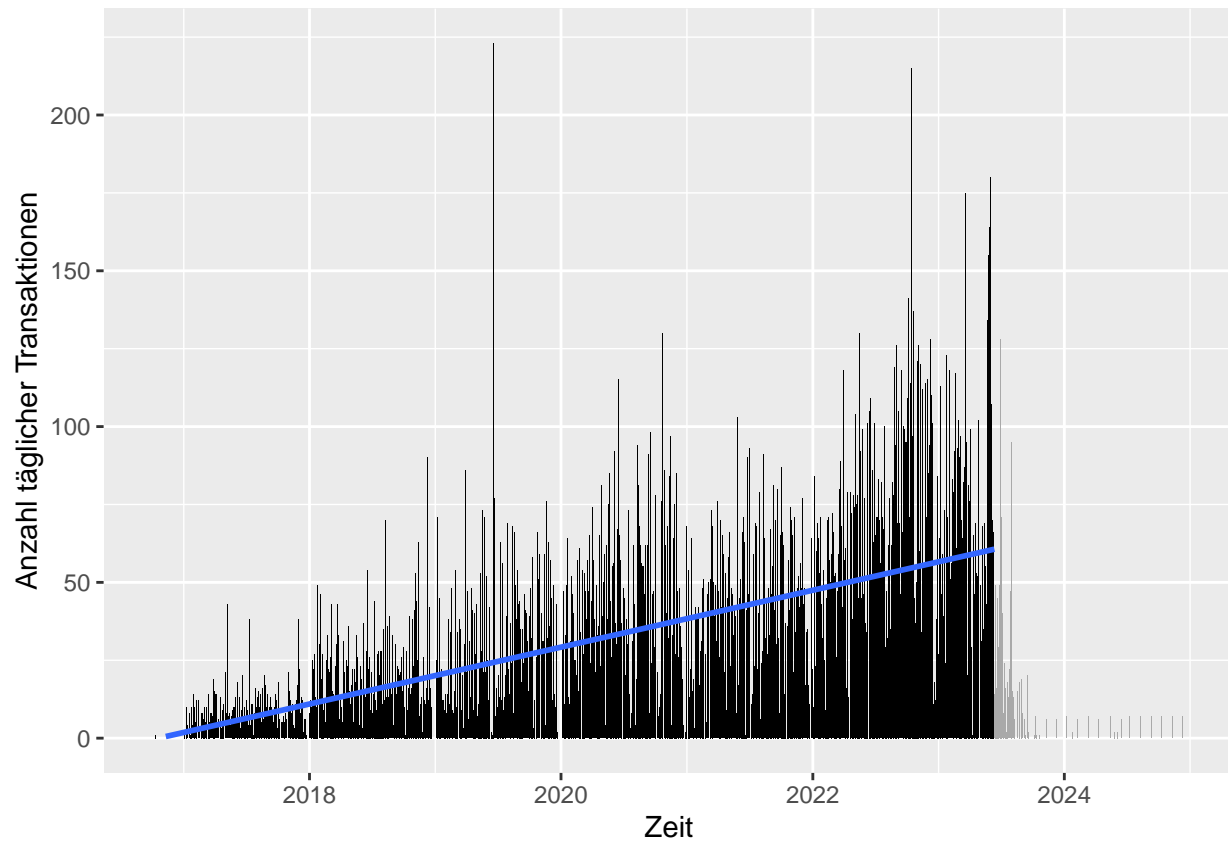
```
## # A tibble: 1 x 1
##       n
##   <int>
## 1   516
```

Plot: Transaktionen nach Zeit

```
# Aggregiert auf Monate
materials %>% group_by(Month=as.yearmon(EinAusDat)) %>% summarize(cnt=n()) %>%
  ↪ mutate(future=ifelse(Month > as.yearmon("2023-06-12"), "Future", "Past")) %>%
  ggplot(aes(x=Month, y=cnt, fill=future)) +
  geom_col() +
  labs(x="Zeit", y="Anzahl monatl. Transaktionen") +
  scale_fill_manual(values=c("darkgrey", "black")) +
  theme(legend.position="none") +
  #geom_vline(aes(xintercept=as.yearmon("2023-06-12")), color="blue", linetype="dashed",
  ↪ size=0.75)
geom_smooth(data=materials %>% filter(EinAusDat <= "2023-06-12") %>%
  ↪ group_by(Month=as.yearmon(EinAusDat)) %>% summarize(cnt=n()) %>%
  ↪ mutate(future=ifelse(Month > as.yearmon("2023-06-12"), "Future", "Past")),
  ↪ method=lm, se=FALSE, color="#C50E1F", size=1.5) +
  ylim(0, NA)
```



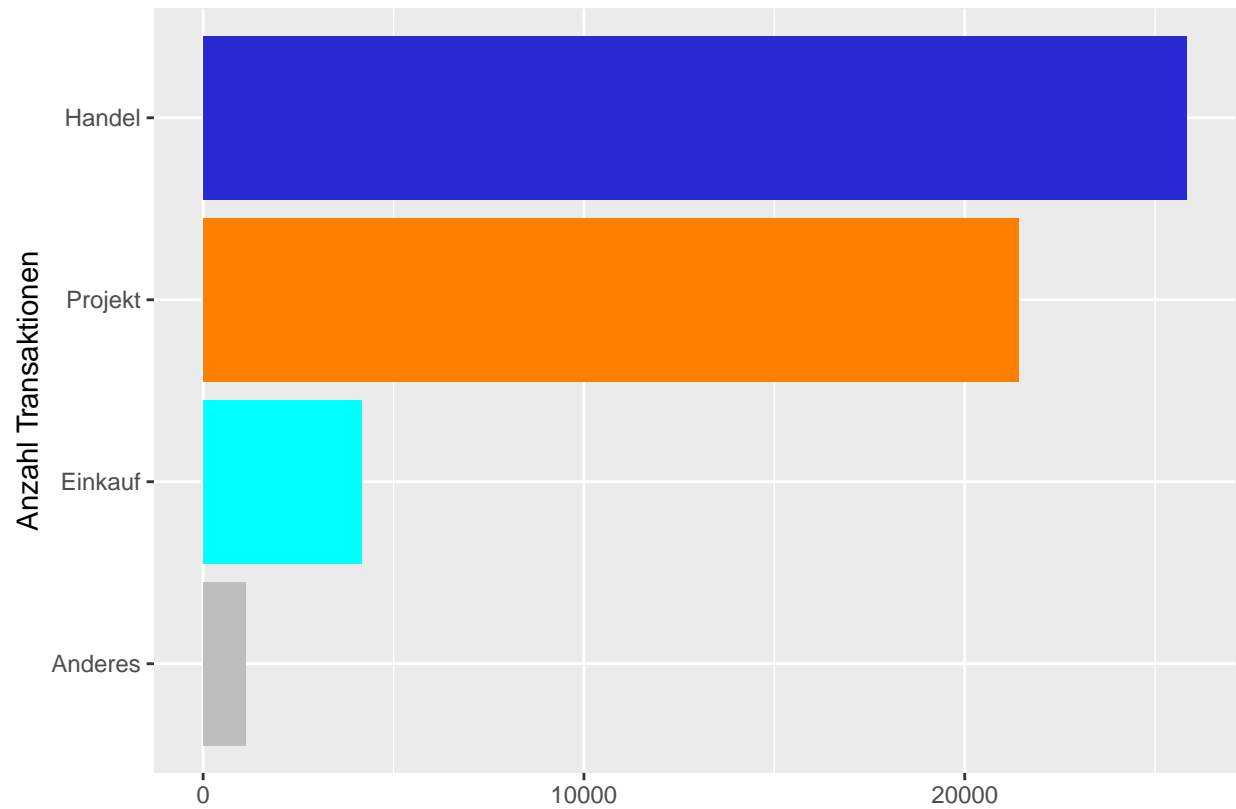
```
# Aggregiert auf Tage
materials %>% group_by(Day=EinAusDat) %>% summarize(cnt=n()) %>% mutate(future=ifelse(Day
  ↪ > "2023-06-12", "Future", "Past")) %>%
  ggplot(aes(x=Day, y=cnt, fill=future)) +
  geom_col()+
  labs(x="Zeit", y="Anzahl täglicher Transaktionen")+
  scale_fill_manual(values=c("darkgrey", "black"))+
  theme(legend.position="none")+
  #geom_vline(aes(xintercept=as.yearmon("2023-06-12")), color="blue", linetype="dashed",
  ↪ size=0.75)
  geom_smooth(data=materials %>% filter(EinAusDat <= "2023-06-12") %>%
  ↪ group_by(Day=EinAusDat) %>% summarize(cnt=n()) %>% mutate(future=ifelse(Day >
  ↪ as.yearmon("2023-06-12"), "Future", "Past")), method=lm, se=FALSE)+
  ylim(0, NA)
```



Plot: Verteilung Projektklassen

```
transaktion_klassiert <- materials %>% group_by(klasse) %>% count()

ggplot(transaktion_klassiert, aes(y=reorder(klasse, n, sum), x=n, fill=klasse)) +
  geom_col() +
  theme(legend.position = "none") +
  ylab("Anzahl Transaktionen") +
  xlab("") +
  scale_fill_manual(values=c("Handel" = "#2A2AD5",
                             "Projekt" = "#FF8000",
                             "Einkauf" = "#00FFFF",
                             "Anderes" = "grey"))
```



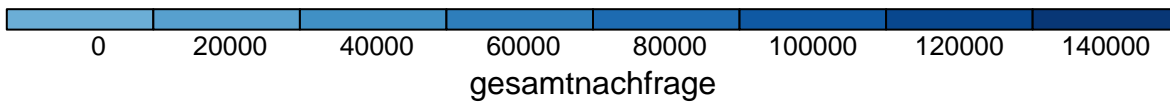
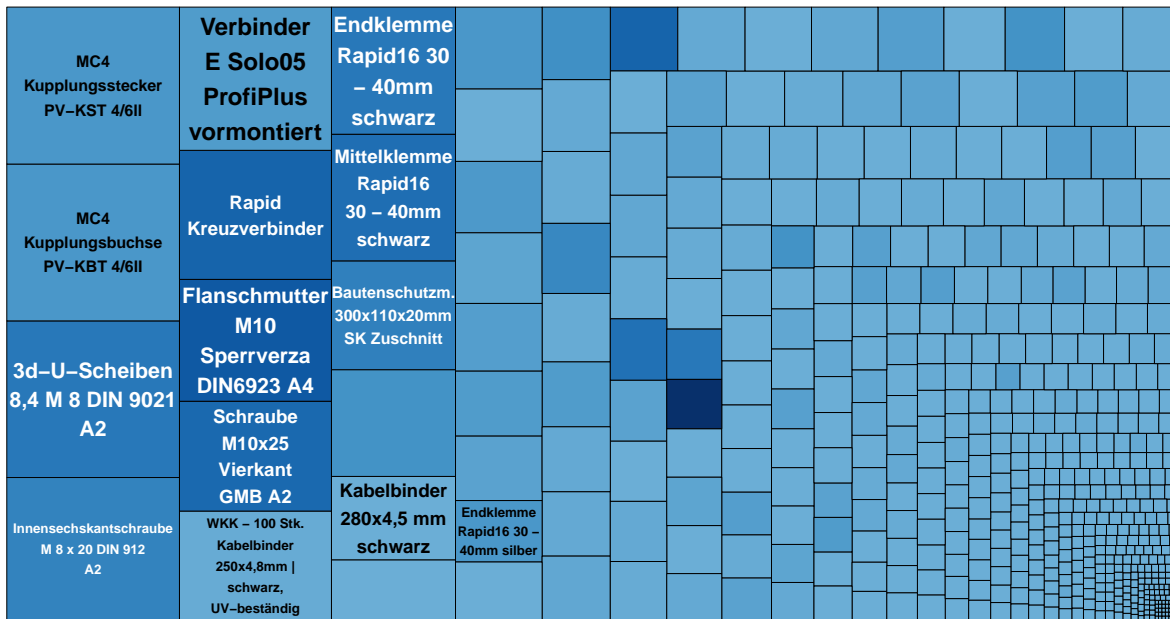
Materialien

Häufigkeit Materialien

```
# Treemap
materials_agg <- materials %>% group_by(`Artikel-Bezeichnung`) %>% summarize(cnt=n(),
  ↳ gesamtnachfrage=sum(ifelse(Menge<0, -Menge, 0)))

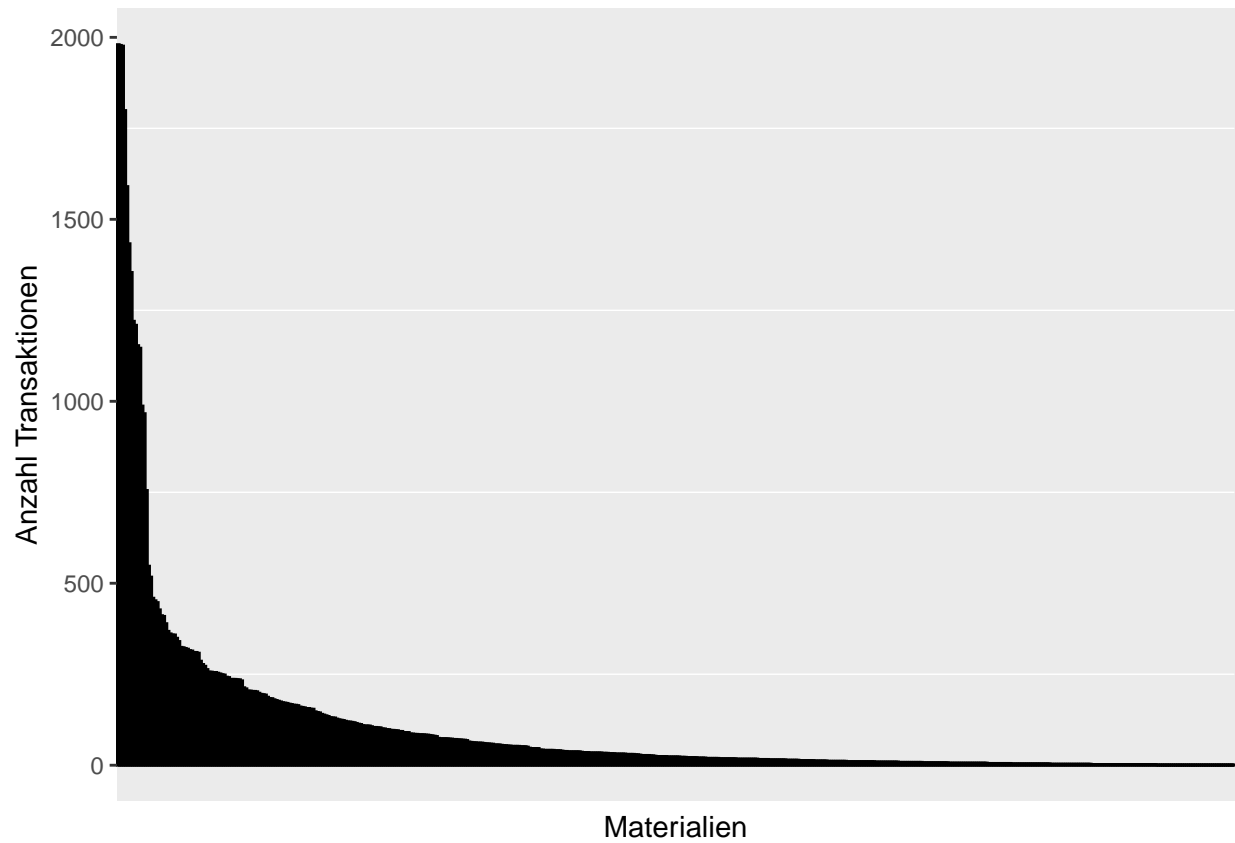
treemap(materials_agg, index="Artikel-Bezeichnung", vSize="cnt",
  ↳ vColor="gesamtnachfrage", palette="Blues", type="value", border.lwds=0.2)
```

cnt

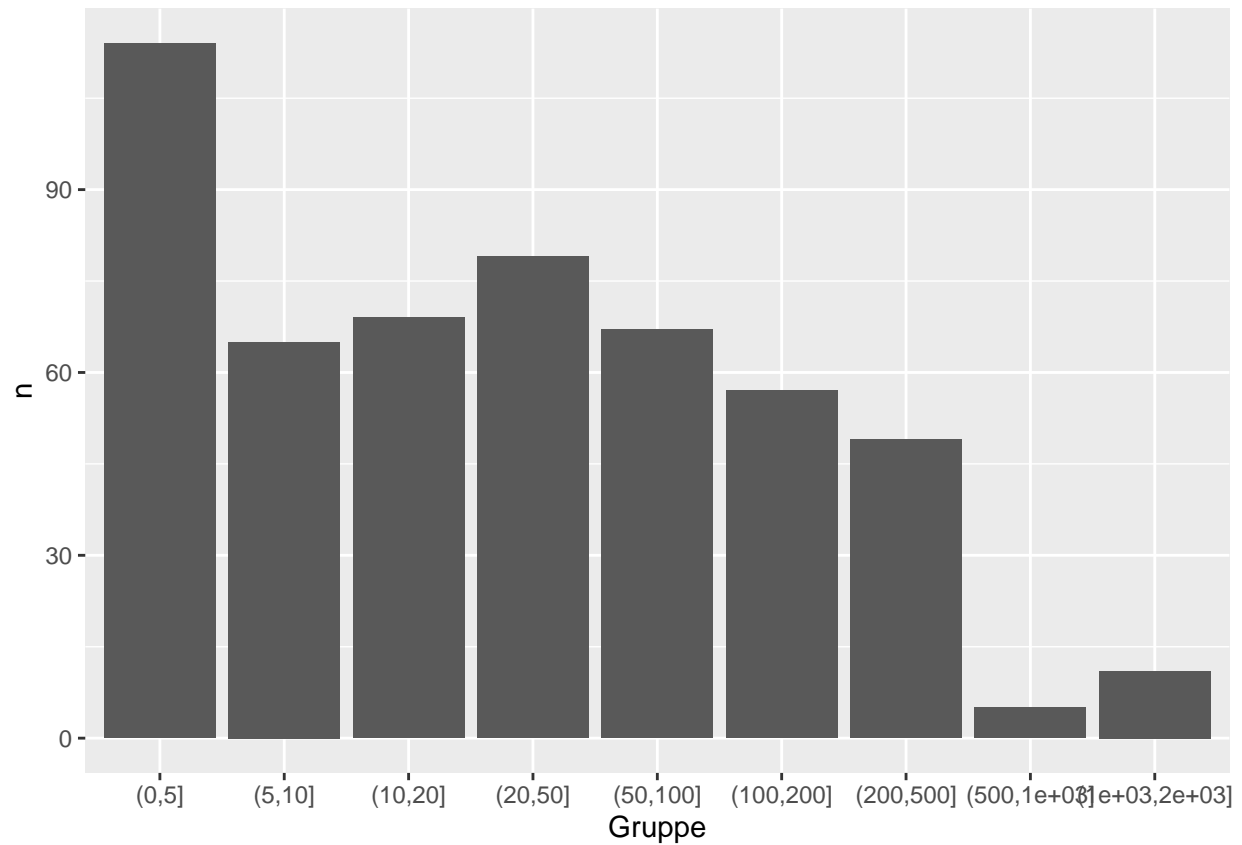


```
# Transaktionen nach Material
agg <- materials %>% group_by(`Artikel-Bezeichnung`) %>% count() %>% arrange(desc(n)) %>%
  ↳ mutate(`Artikel-Bezeichnung` = substring(`Artikel-Bezeichnung`, 0, 50))

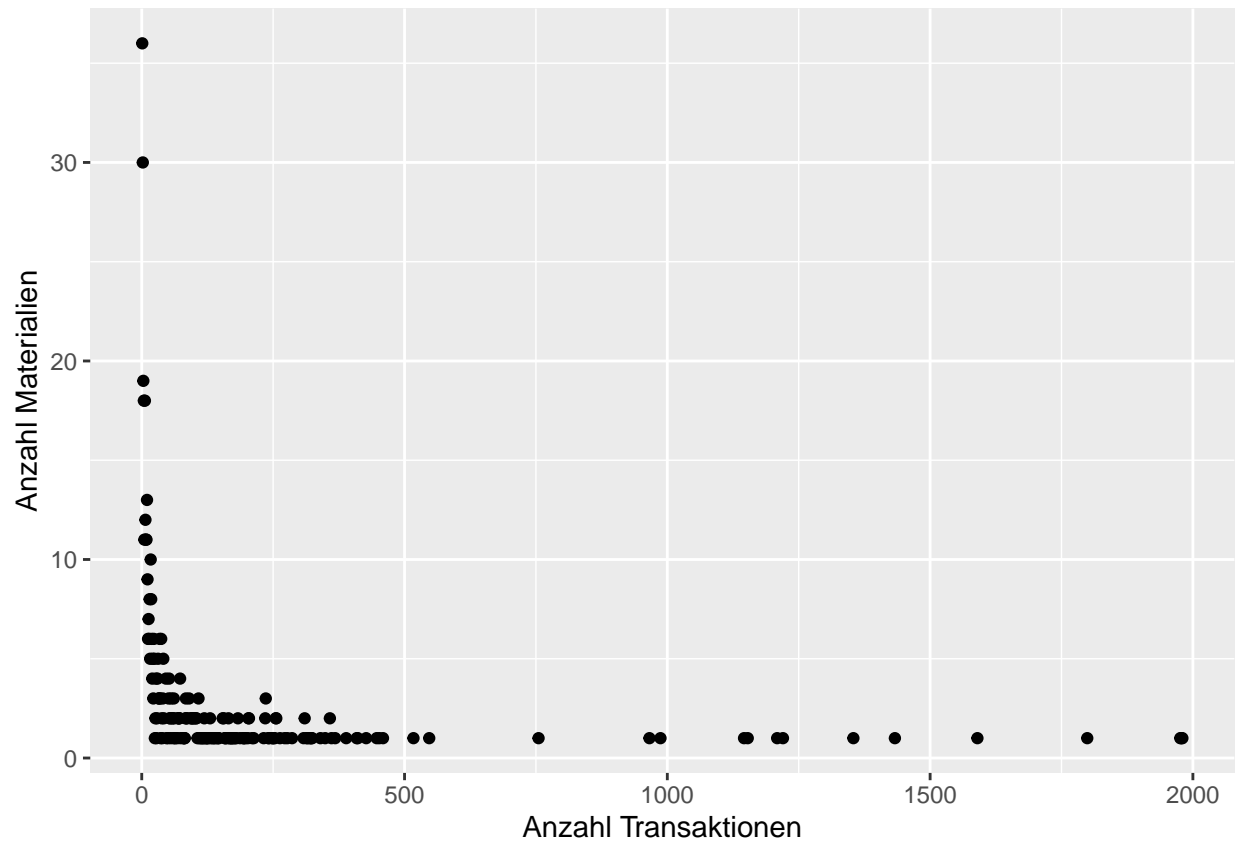
# Plot 1: Materialien nach Anzahl Transaktionen
ggplot(agg, aes(x=reorder(`Artikel-Bezeichnung`, -n), y=n))+
  geom_col(color="black")+
  labs(y="Anzahl Transaktionen", x="Materialien")+
  theme(axis.ticks.x=element_blank(), axis.text.x=element_blank(),
        panel.grid.major.x=element_blank(), panel.grid.major.y=element_blank())
```



```
# Plot 2: In Buckets einsortieren
agg %>% mutate(Gruppe=cut(n, breaks=c(0, 5, 10, 20, 50, 100, 200, 500, 1000, 2000))) %>%
  ↳ group_by(Gruppe) %>% count() %>%
  ggplot(aes(x=Gruppe, y=n))+
  geom_col()
```

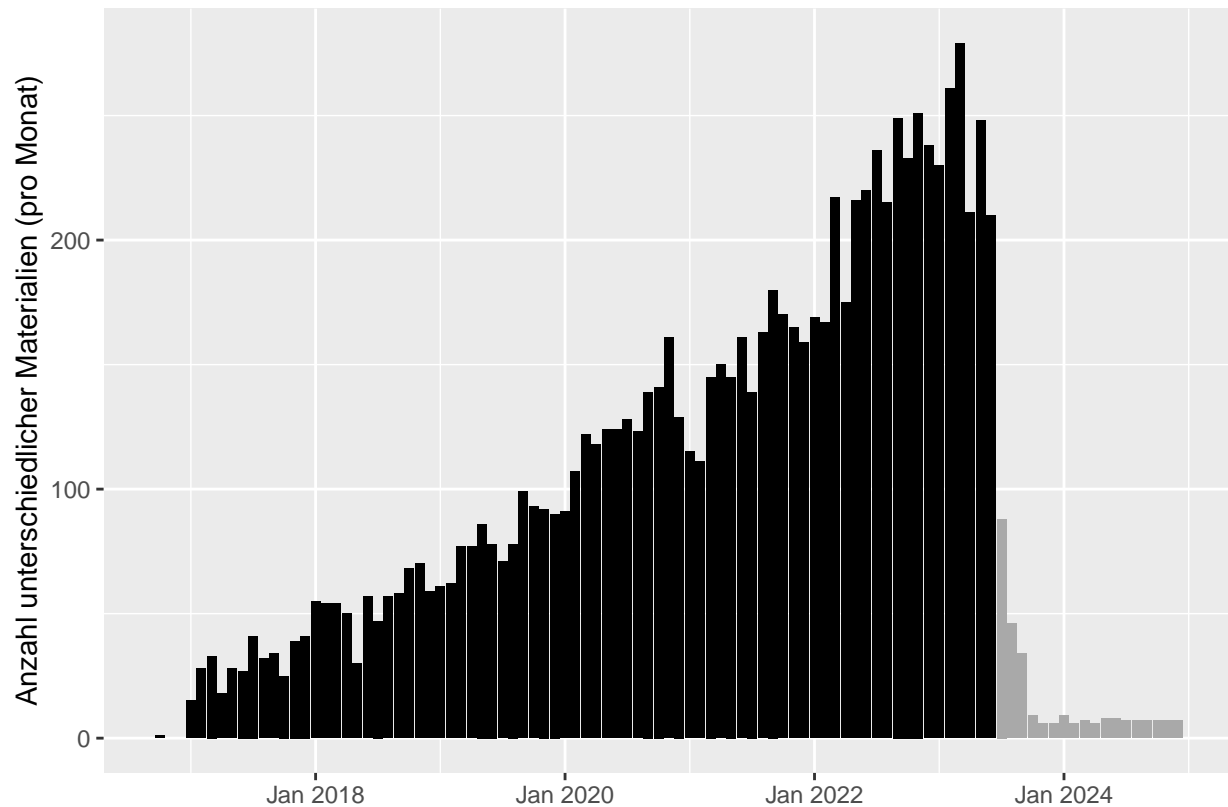
```
# Plot 3: Scatter Plot der Gruppen
materials %>% group_by(`Artikel-Bezeichnung`) %>% summarize(n_trans=n()) %>%
  ↳ group_by(n_trans) %>% summarize(n_materialien_je_n_trans=n(),
  ↳ first_mat=first(`Artikel-Bezeichnung`)) %>%
  arrange(desc(n_trans), n_materialien_je_n_trans) %>%
  ggplot(aes(x=n_trans, y=n_materialien_je_n_trans, label=first_mat))+
  geom_point()+
  labs(x="Anzahl Transaktionen", y="Anzahl Materialien")
```



```
#geom_text(data=materials %>% group_by(`Artikel-Bezeichnung`) %>%
  ↳ summarize(n_trans=n()) %>% group_by(n_trans) %>%
  ↳ summarize(n_materialien_je_n_trans=n(), first_mat=first(`Artikel-Bezeichnung`)) %>%
  ↳ arrange(desc(n_trans), n_materialien_je_n_trans) %>% nth(1), check_overlap = TRUE)
```

Anzahl unterschiedlicher Materialien pro Monat

```
# Anzahl unterschiedlicher Materialien (pro Monat)
materials %>% group_by(mon=as.yearmon(EinAusDat)) %>%
  ↳ summarize(anz_mat=n_distinct(`Artikel-Bezeichnung`)) %>%
  mutate(future=ifelse(mon > as.yearmon("2023-06-12"), "Future", "Past")) %>%
  ggplot(aes(x=mon, y=anz_mat, fill=future))+geom_col()+
  scale_fill_manual(values=c("darkgrey", "black"))+
  theme(legend.position="none")+
  labs(x="", y="Anzahl unterschiedlicher Materialien (pro Monat)")
```



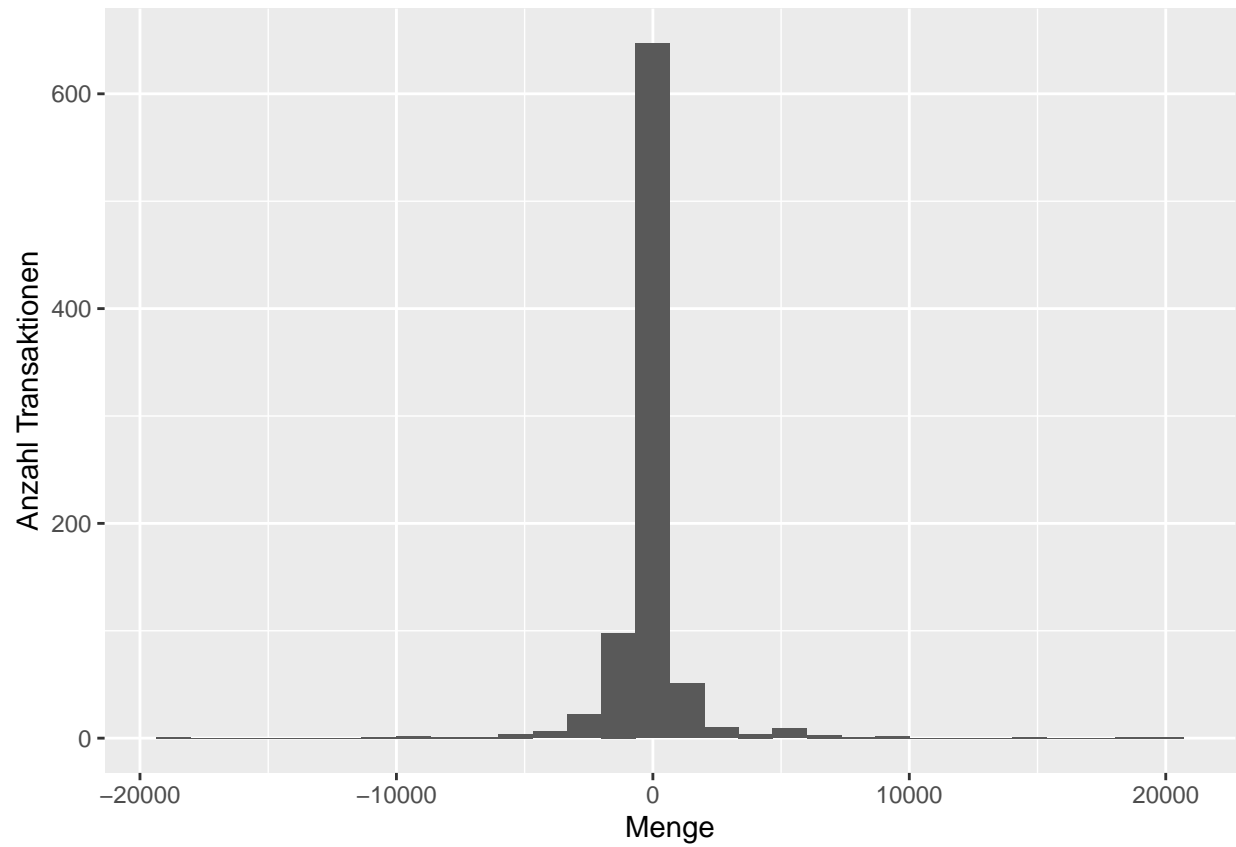
```
# Summary der Menge und Bestand
```

```
summary(materials[, c("Menge", "Bestand")])
```

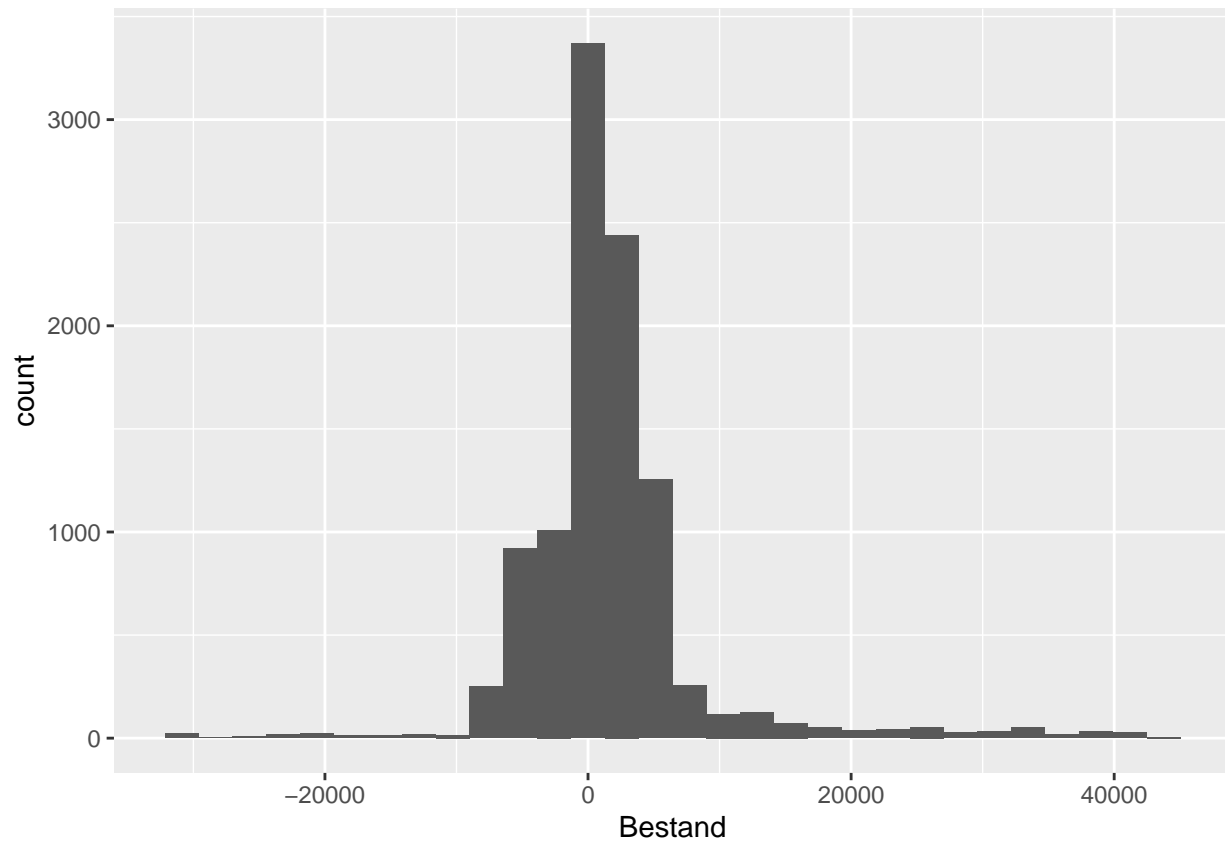
	Menge	Bestand
## Min.	:-18750.000	Min. : -31659.0
## 1st Qu.:	-40.000	1st Qu.: 50.0
## Median :	-12.000	Median : 450.0
## Mean :	2.677	Mean : 927.9
## 3rd Qu.:	-3.000	3rd Qu.: 1392.0
## Max. :	20000.000	Max. : 42991.0

```
# Anzahl aller Transaktionen pro Menge
```

```
materials %>% group_by(Menge) %>% count() %>%
  ggplot(aes(x=Menge))+
  geom_histogram()+
  labs(y="Anzahl Transaktionen")
```



```
# Histogram des Bestands
materials %>% group_by(Bestand) %>% count() %>%
  ggplot(aes(x=Bestand))+
  geom_histogram()
```



EinAutDatTyp

Häufigkeit EinAusDatTyp

```
materials %>% group_by(EinAutDatTyp) %>% count() %>% arrange(desc(n))
```

```
## # A tibble: 7 x 2
## # Groups:   EinAutDatTyp [7]
##   EinAutDatTyp      n
##   <fct>         <int>
## 1 BereitDat     35979
## 2 PjPhaseDat    10447
## 3 geliefertAm   5674
## 4 UmsProg_Dat    198
## 5 LTbestätigt   192
## 6 LT             10
## 7 BstDat         1
```

Häufigkeiten EinAusDatTyp über alle Materialien

```
materials %>% group_by(EinAutDatTyp) %>% count() %>% arrange(desc(n))
```

```
## # A tibble: 7 x 2
## # Groups:   EinAutDatTyp [7]
##   EinAutDatTyp      n
##   <fct>         <int>
```

```
## 1 BereitDat      35979
## 2 PjPhaseDat     10447
## 3 geliefertAm    5674
## 4 UmsProg_Dat    198
## 5 LTbestätigt    192
## 6 LT             10
## 7 BstDat         1
```

Projekte

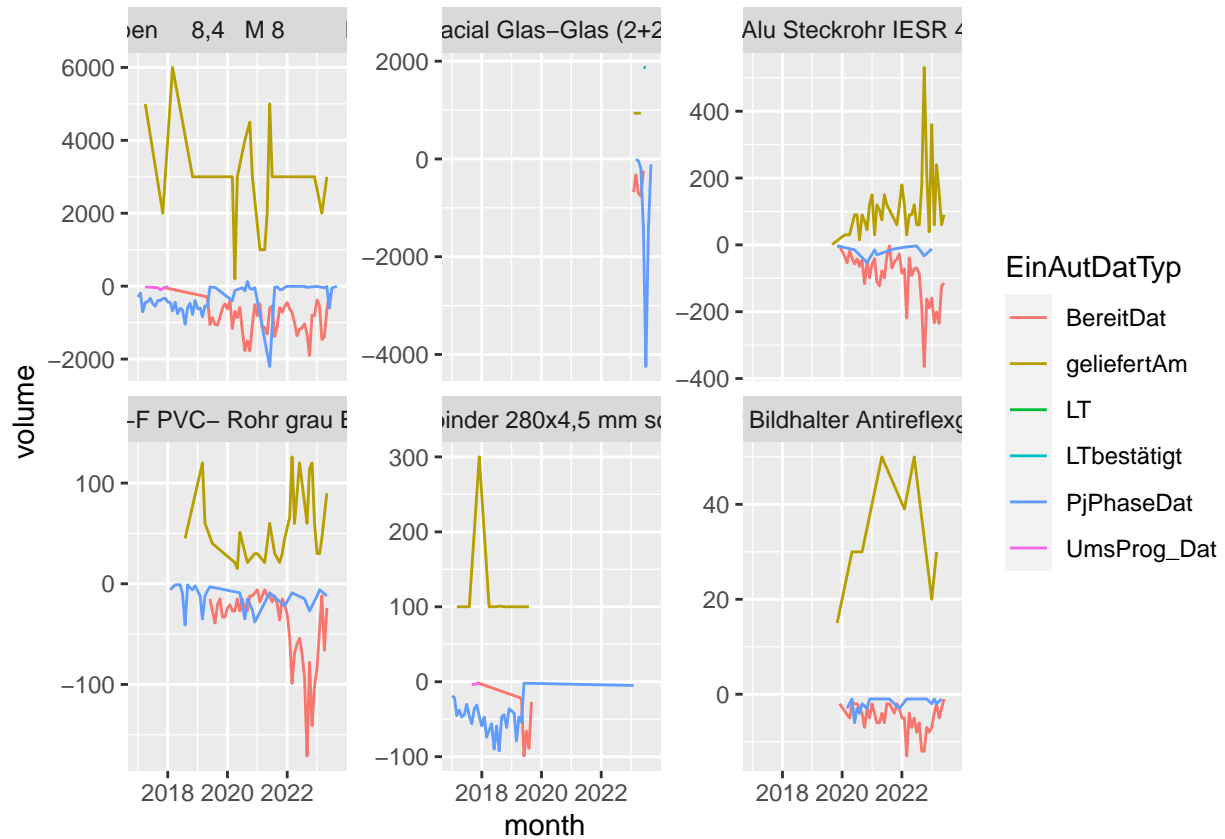
Häufigkeiten Projekte

```
materials %>% group_by(PjInfo) %>% count() %>% arrange(desc(n))
```

```
## # A tibble: 920 x 2
## # Groups:   PjInfo [920]
##   PjInfo                                n
##   <fct>                                <int>
## 1 PV-Materialverkauf: Schletter, DC      9445
## 2 PV-Materialverkauf: SE, Schletter, DC  4435
## 3 PV-Materialverkauf: Schletter          3511
## 4 PV-Materialverkauf: LG, Schletter, DC   747
## 5 PV-Materialverkauf: Schletter, SL, DC   736
## 6 PV-Materialverkauf: SPR, Schletter, DC  583
## 7 PV-Materialverkauf: Schletter, SL      541
## 8 Einkauf 2023 Q1                       410
## 9 Einkauf 2022 Q3                       358
## 10 Einkauf 2022 Q2                      337
## # i 910 more rows
```

Plot: Mengen nach EinAusDatTyp

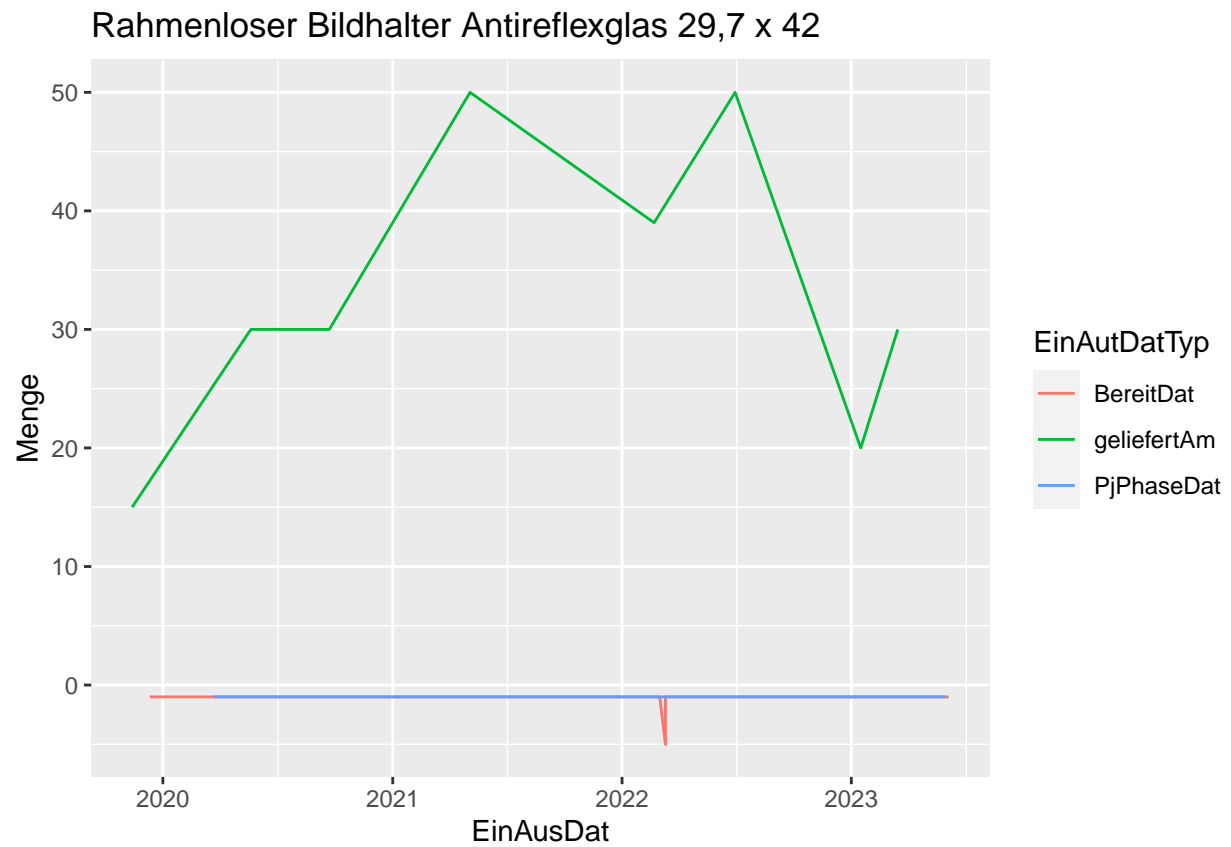
```
materials_b %>% group_by(`Artikel-Bezeichnung`, EinAusDatTyp, month =
  ↳ lubridate::floor_date(EinAusDat, 'month')) %>%
  summarise(volume=sum(Menge)) %>%
  ggplot(aes(x=month, y=volume, group=EinAusDatTyp, color=EinAusDatTyp
    )) +
  geom_line() +
  facet_wrap(~`Artikel-Bezeichnung`, scales = 'free_y')
```



```
# Einzelne Plots pro Artikel für Mengen nach EinAusDatTyp
plots <- materials_b %>% group_by(`Artikel-Bezeichnung`) %>% nest %>%
  mutate(plot = map2(
    data, `Artikel-Bezeichnung`,
    ~ ggplot(data = .x, aes(x=EinAusDat,y=Menge,group=EinAutDatTyp, color=EinAutDatTyp))
    +
    ggtitle(glue("{.y}")) +
    geom_line()))

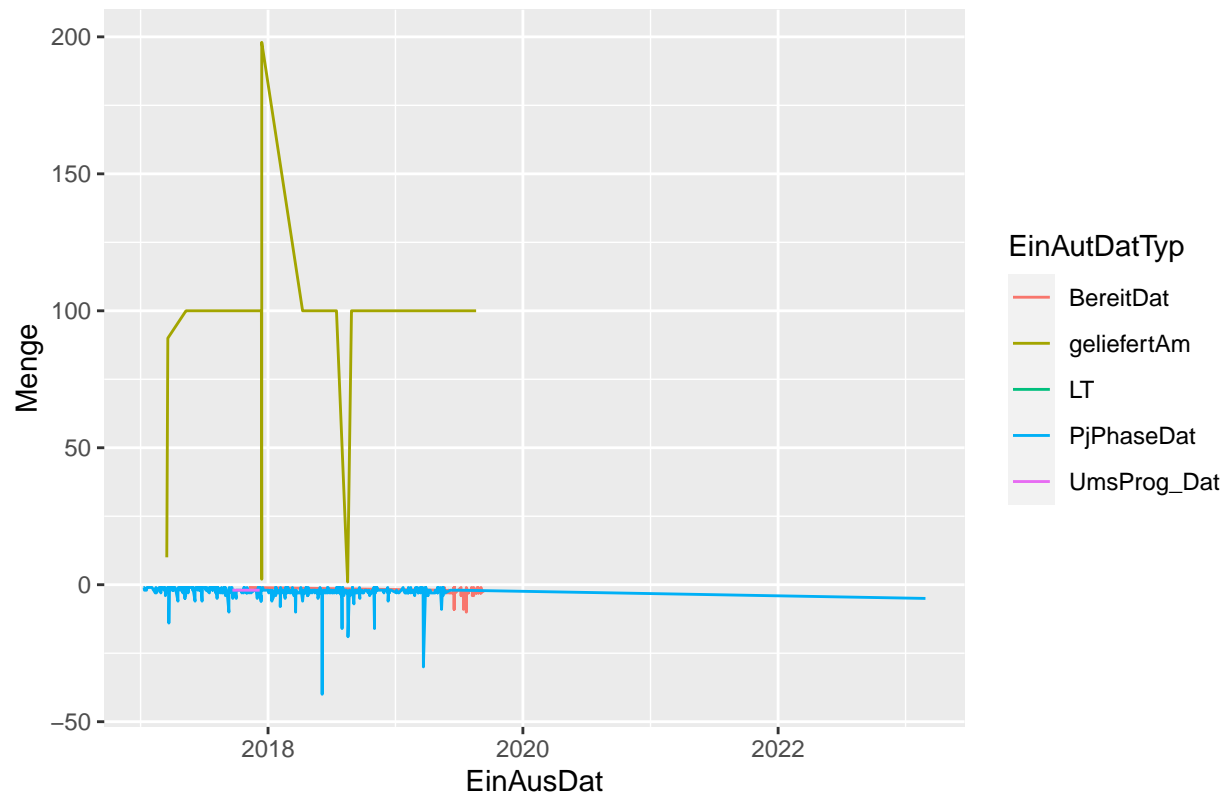
print(plots$plot)
```

```
## [[1]]
```



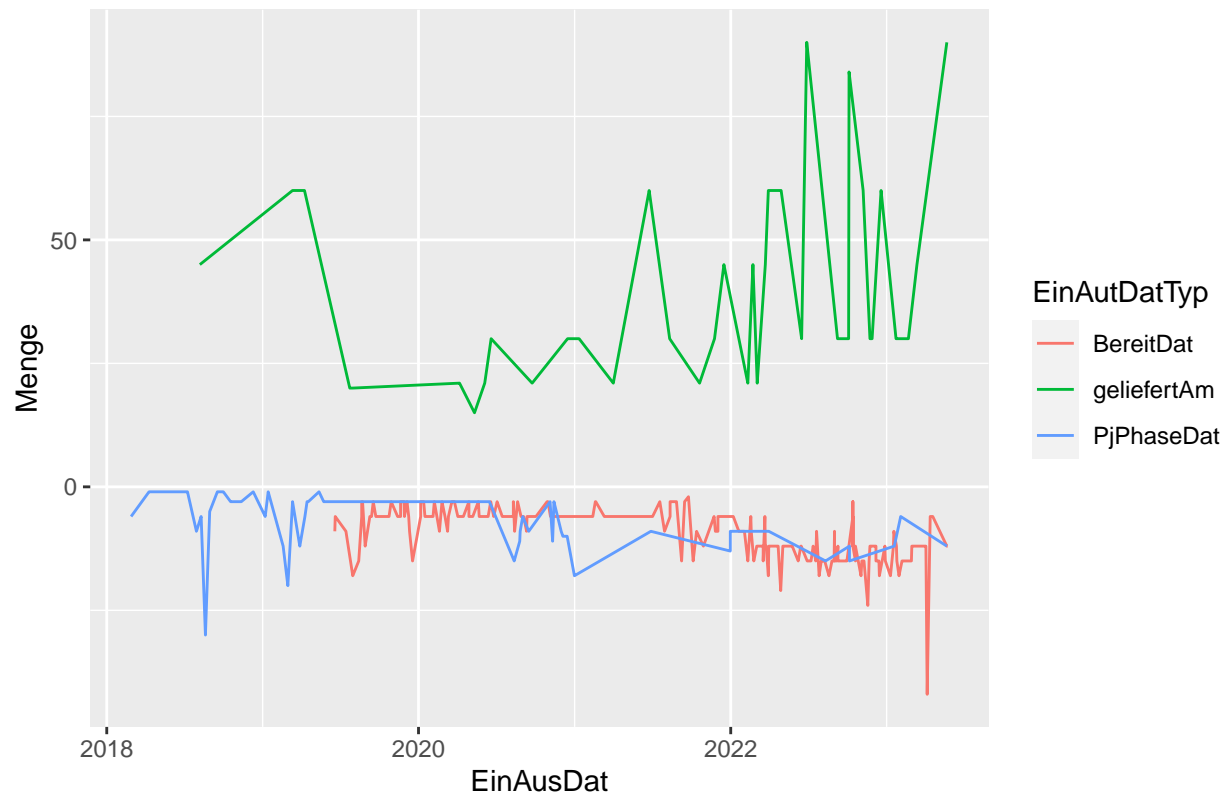
[[2]]

Kabelbinder 280x4,5 mm schwarz



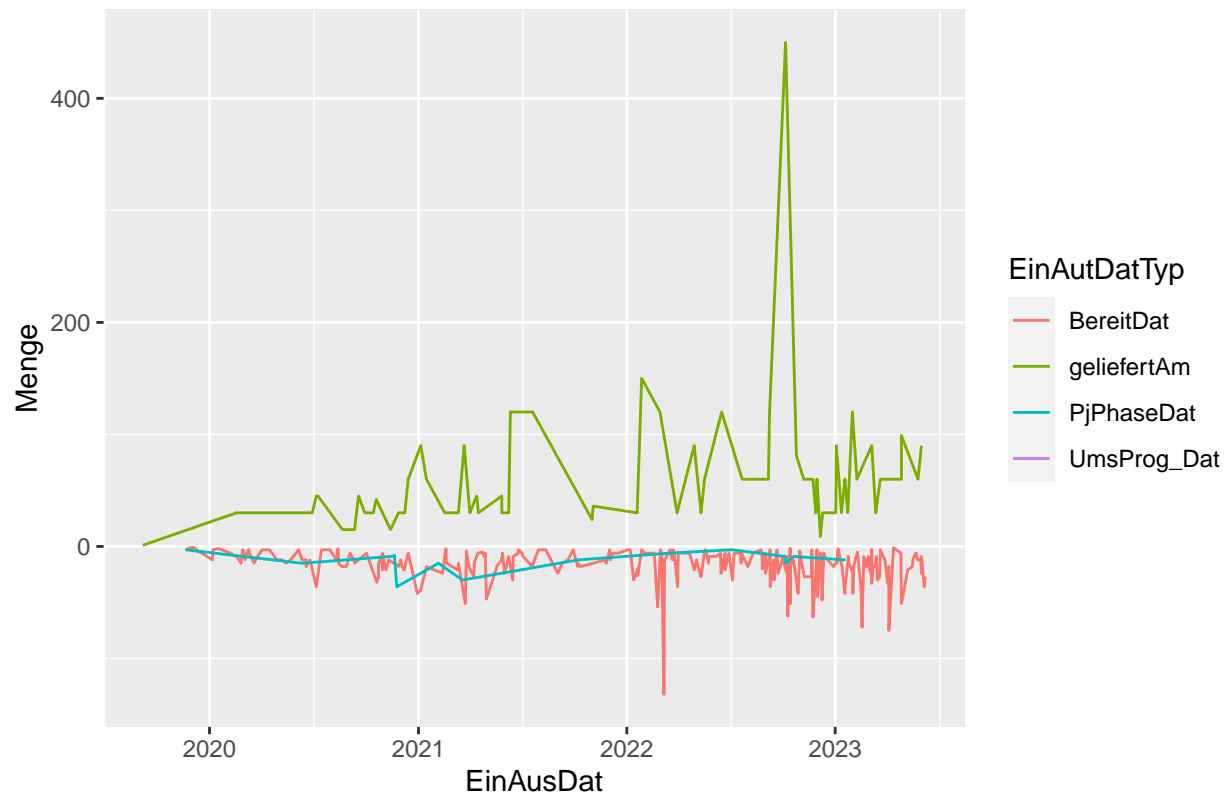
[[3]]

FPKU-EM-F PVC- Rohr grau EN50 L=3m



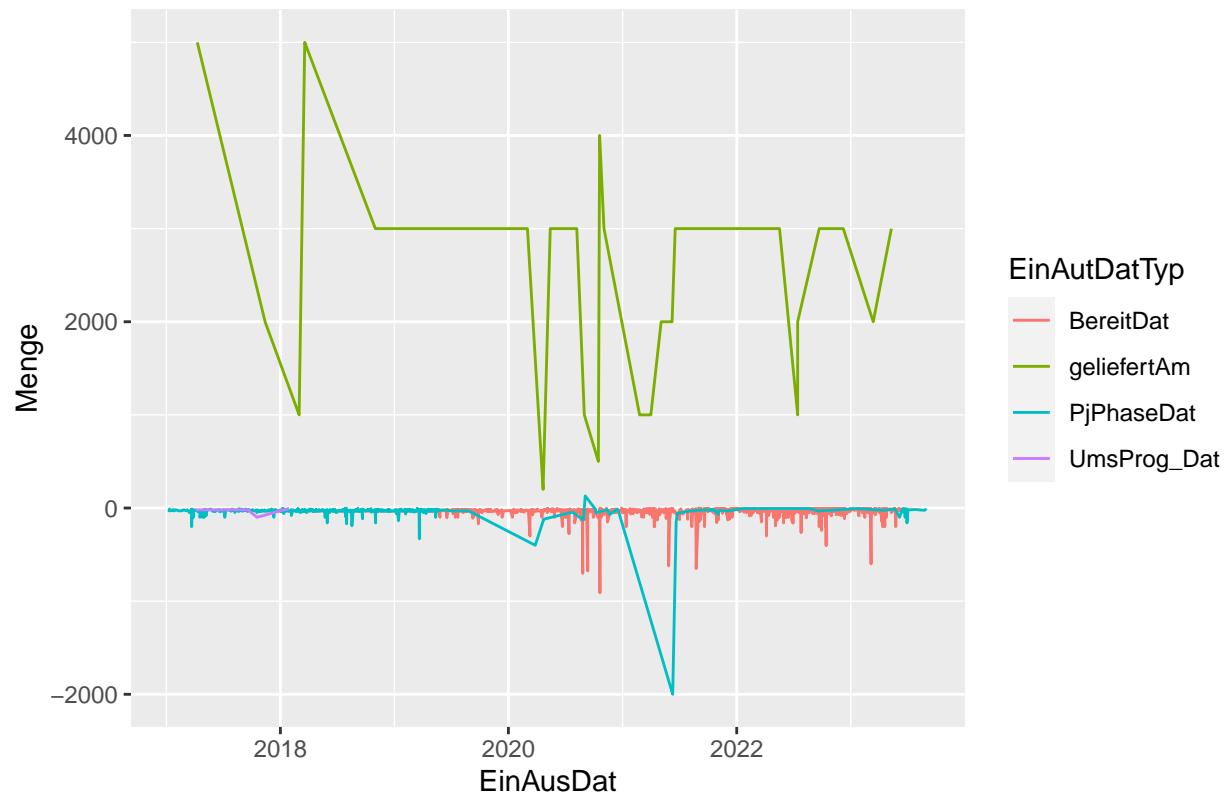
[[4]]

FINTECH Alu Steckrohr IESR 40 AL (3m)



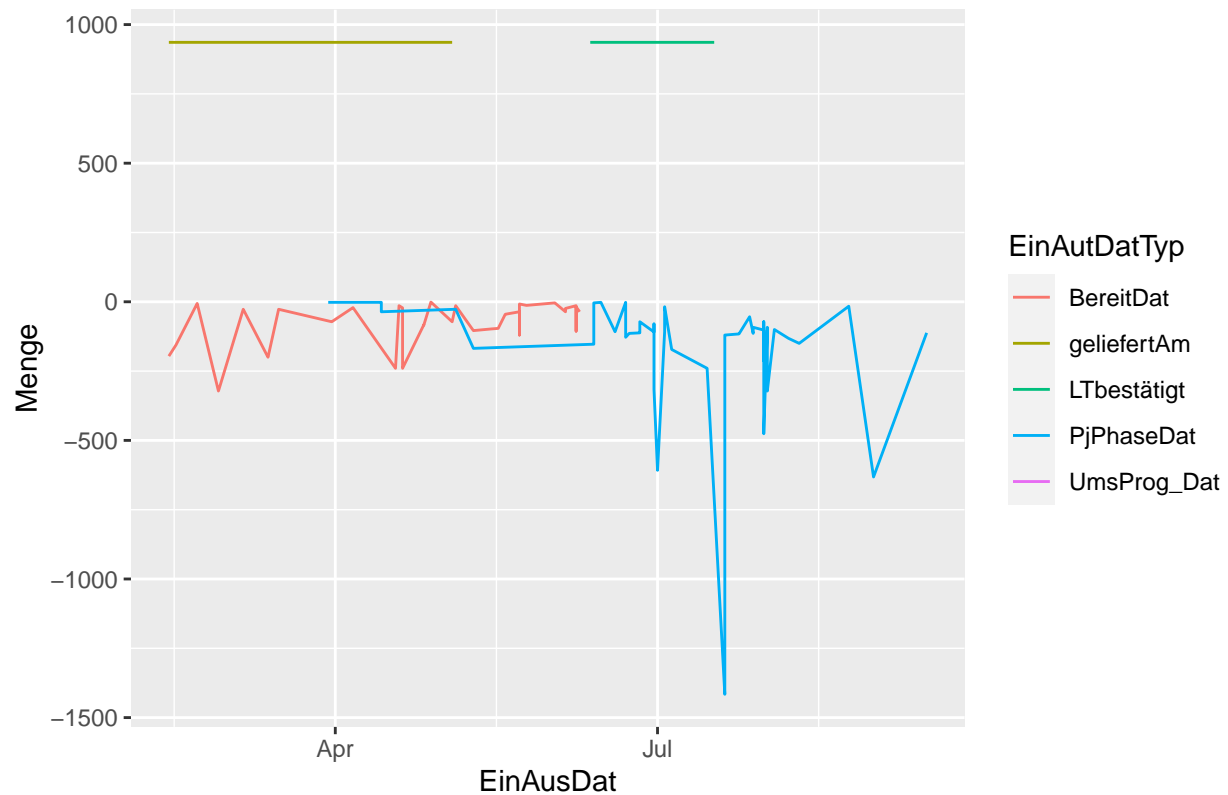
```
##  
## [[5]]
```

3d-U-Scheiben 8,4 M 8 DIN 9021 A2



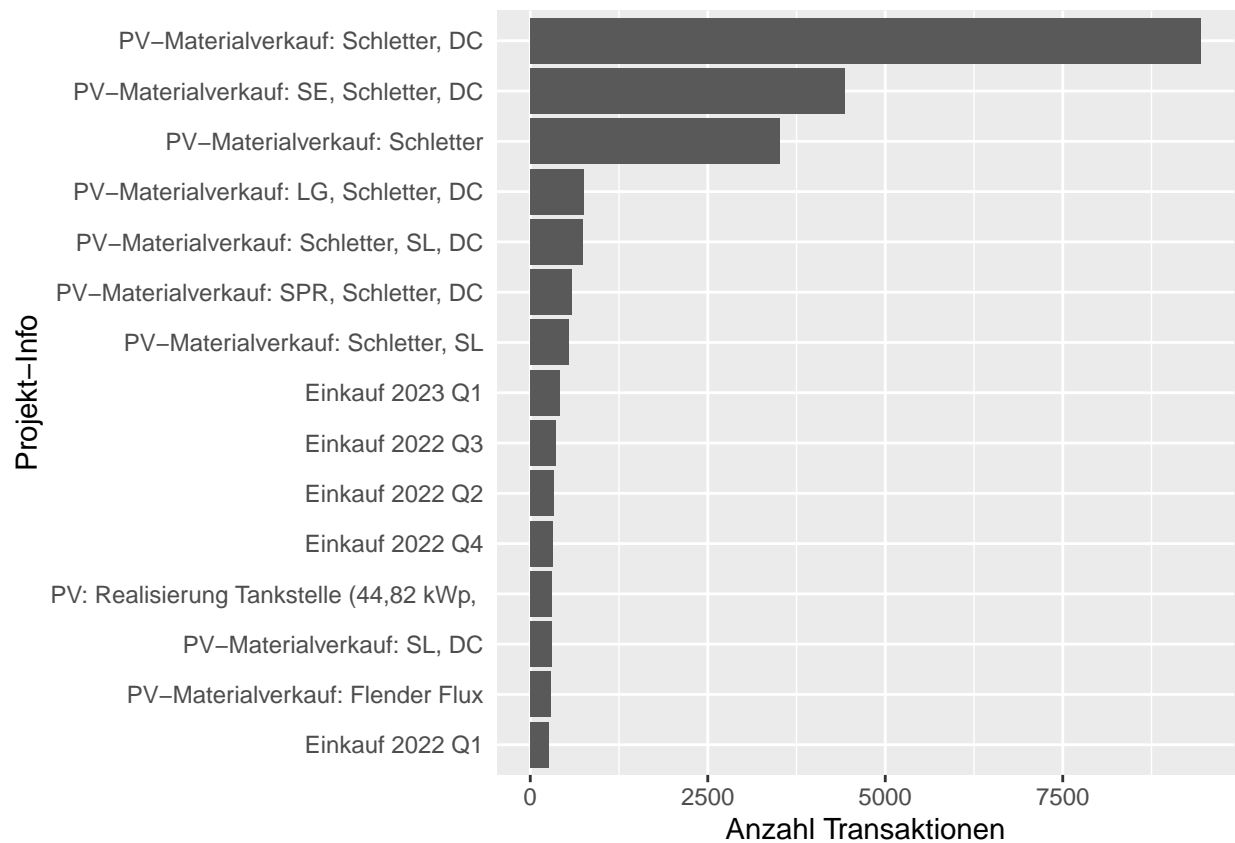
```
##  
## [[6]]
```

DAS N-TOPcon Bifacial Glas-Glas (2+2mm) Black 415 WP



Plot: Top Projekte

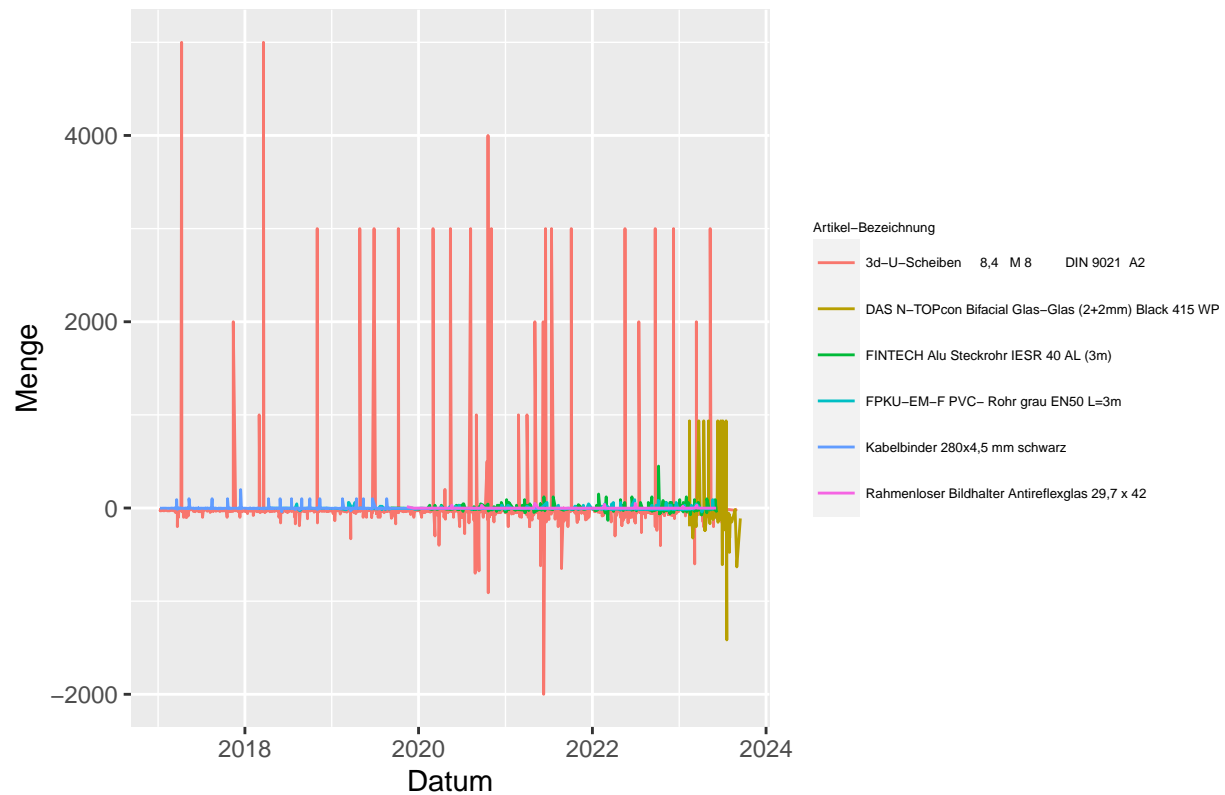
```
# Top 15 Projekte
materials %>% group_by(PjInfo, klasse) %>% count() %>% arrange(desc(n)) %>%
  ↳ mutate(PjInfo=substring(PjInfo, 0, 40)) %>% head(15) %>% ggplot(aes(x=n,
  ↳ y=reorder(PjInfo, n)))+
  geom_col()+
  theme(legend.position="none", axis.ticks.y=element_blank())+
  labs(x="Anzahl Transaktionen", y="Projekt-Info")+
  scale_fill_manual(values=c("Handel" = "#2A2AD5",
                             "Projekt" = "#FF8000",
                             "Einkauf" = "#00FFFF",
                             "Anderes" = "grey"))
```



Menge ### Plot: Menge über Zeit

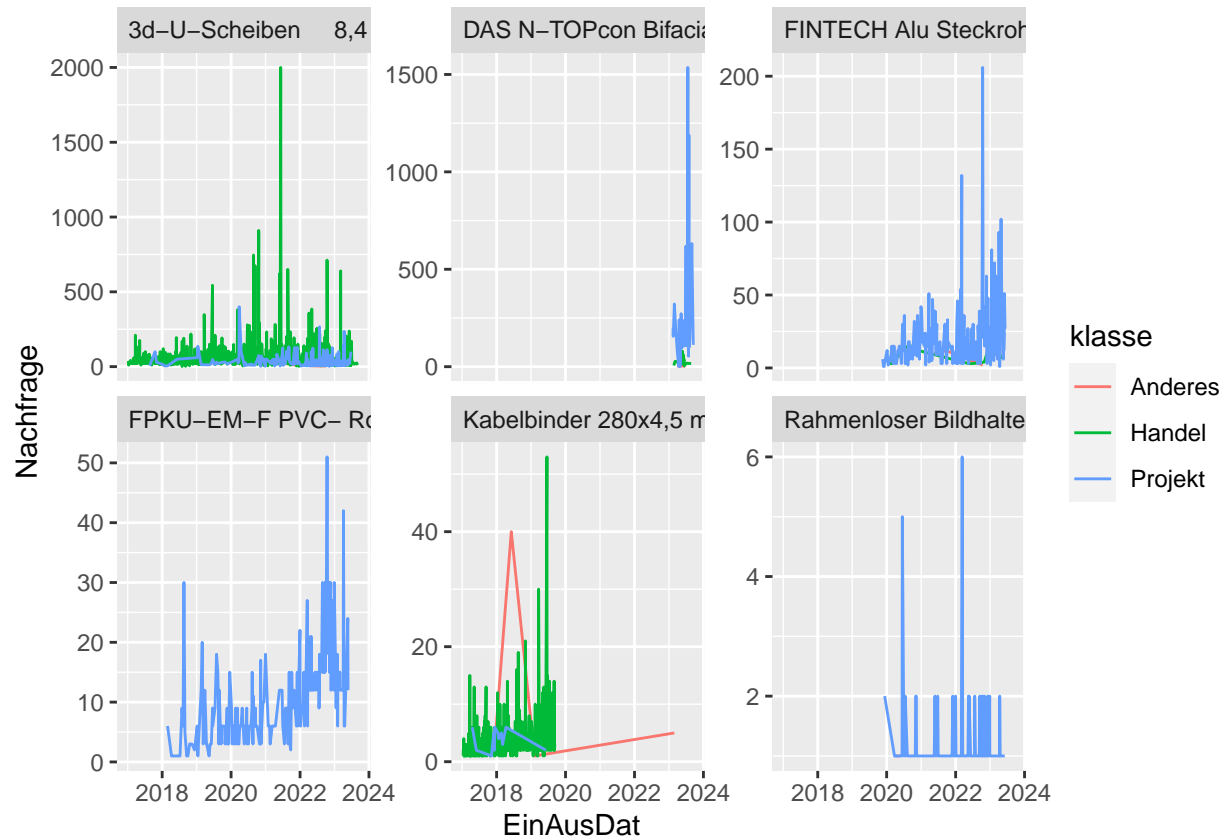
```
materials_b %>%
  ggplot() +
  geom_line(aes(x=EinAusDat,y=Menge,group=`Artikel-Bezeichnung`,
    ↪ color=`Artikel-Bezeichnung`
  ))+
  xlab('Datum')+
  ggtitle('Menge über die Zeit')+
  theme(legend.title = element_text(size = 5),
        legend.text = element_text(size = 5))
```

Menge über die Zeit



Plot: Mengen nach klasse und Material

```
materials_b %>% filter(Menge < 0) %>% group_by(`Artikel-Bezeichnung`, EinAusDat, klasse)
→ %>% summarize(Nachfrage=sum(-Menge)) %>%
  ggplot(aes(x=EinAusDat, y=Nachfrage, group=klasse, color=klasse)) +
  geom_line() +
  # Text linksbündig
  theme(strip.text.x = element_text(hjust=0.0)) +
  facet_wrap(~`Artikel-Bezeichnung`, scales = 'free_y')
```

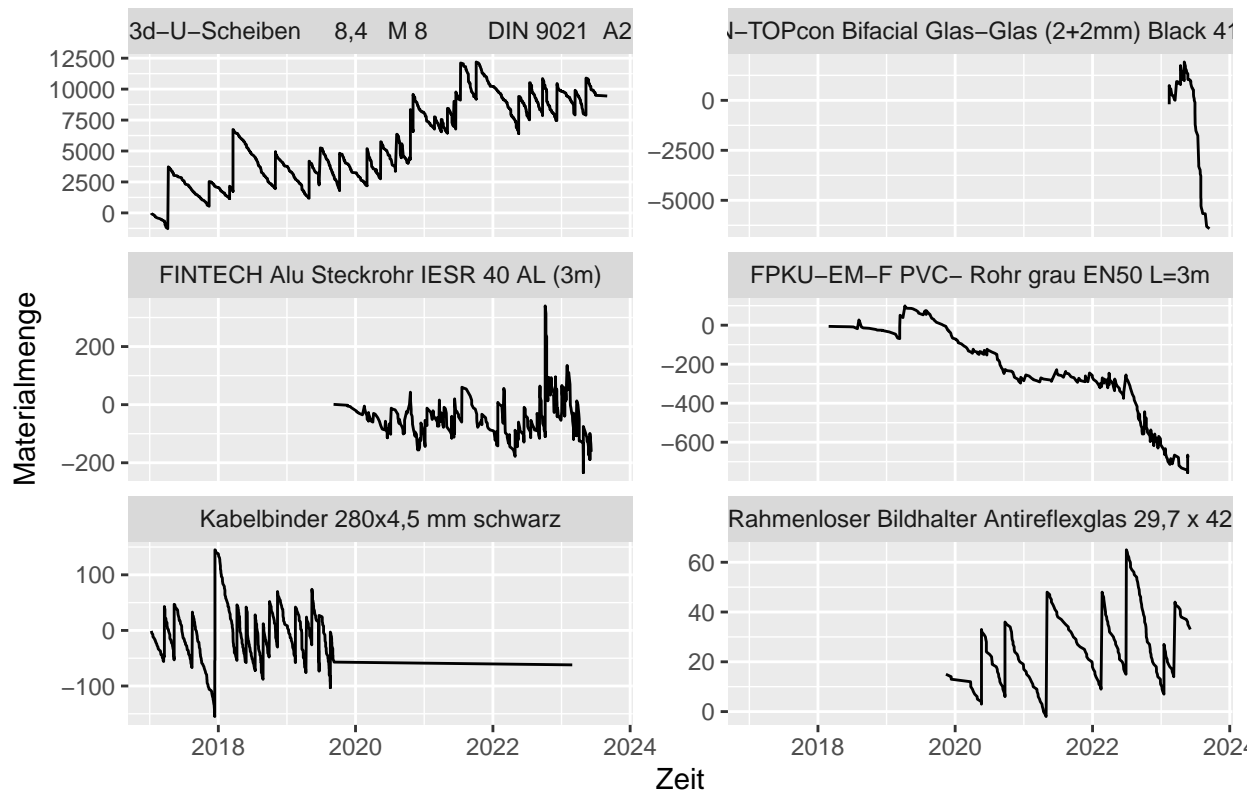


Plot: Korrigierter Bestand (cumsum)

```
mat_verlauf <- materials_b %>%
  filter(EinAutDatTyp == "geliefertAm" | EinAutDatTyp == "BereitDat" | EinAutDatTyp ==
    ↪ "PjPhaseDat") %>%
  group_by(`ArtikelNr`) %>%
  mutate(cum_sum = cumsum(Menge))

ggplot(data=mat_verlauf, aes(x=EinAusDat, y=cum_sum, fill=`ArtikelNr`)) +
  facet_wrap(~`Artikel-Bezeichnung`, nrow=3, scales="free_y") +
  geom_line() +
  xlab("Zeit") +
  ylab("Materialmenge") +
  theme(legend.position = "none") +
  ggtitle("EinAutDatTyp = geliefertAm, BereitDat, PjPhaseDat")
```


EinAusDatTyp = geliefertAm, BereitDat, PjPhaseDat

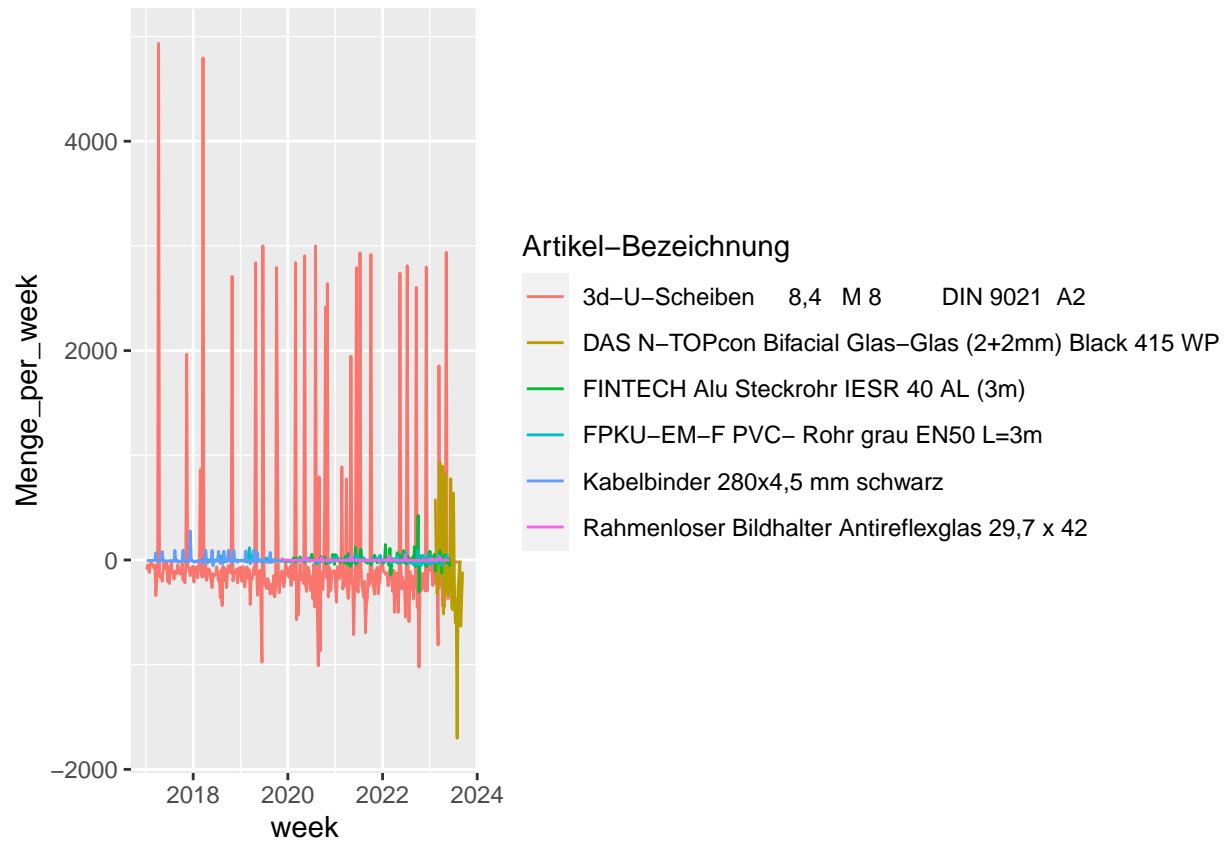


Durchschnitte der Menge (Nachfrage und Lieferungen gemeinsam) ### Plot: Wochendurchschnitt der Menge

```
# Berechnen des Durchschnitts nach Wochen

weekly_Menge <- materials_b %>%
  group_by(`Artikel-Bezeichnung`, week = floor_date(EinAusDat, 'week')) %>%
  summarise(Menge_per_week= sum(Menge))

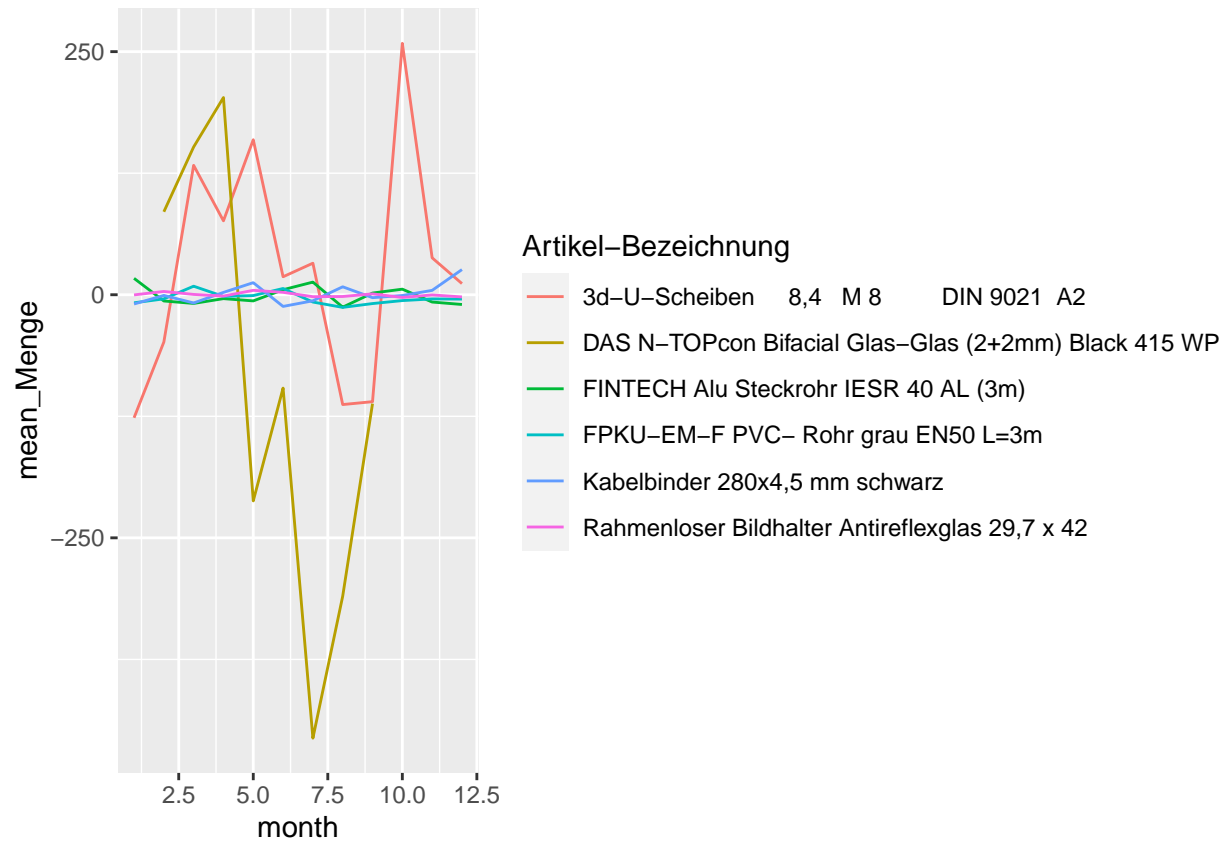
# Plot nach Wochen
weekly_Menge %>%
  ggplot() +
  geom_line(aes(x=week,y=Menge_per_week, group=`Artikel-Bezeichnung`,
    ↪ color=`Artikel-Bezeichnung`
  ))
```



Plot: Monatsdurchschnitt der Menge

```
# Berechnen des Durchschnitts nach Monaten
monthly_mean <- weekly_Menge %>% group_by(`Artikel-Bezeichnung`, month=month(week)) %>%
  mutate(mean_Menge= mean(Menge_per_week))

# Plot Monatsdurchschnitt
monthly_mean %>%
  ggplot() +
  geom_line(aes(x=month,y=mean_Menge, group=`Artikel-Bezeichnung`,
    ↪ color=`Artikel-Bezeichnung`
  )))
```

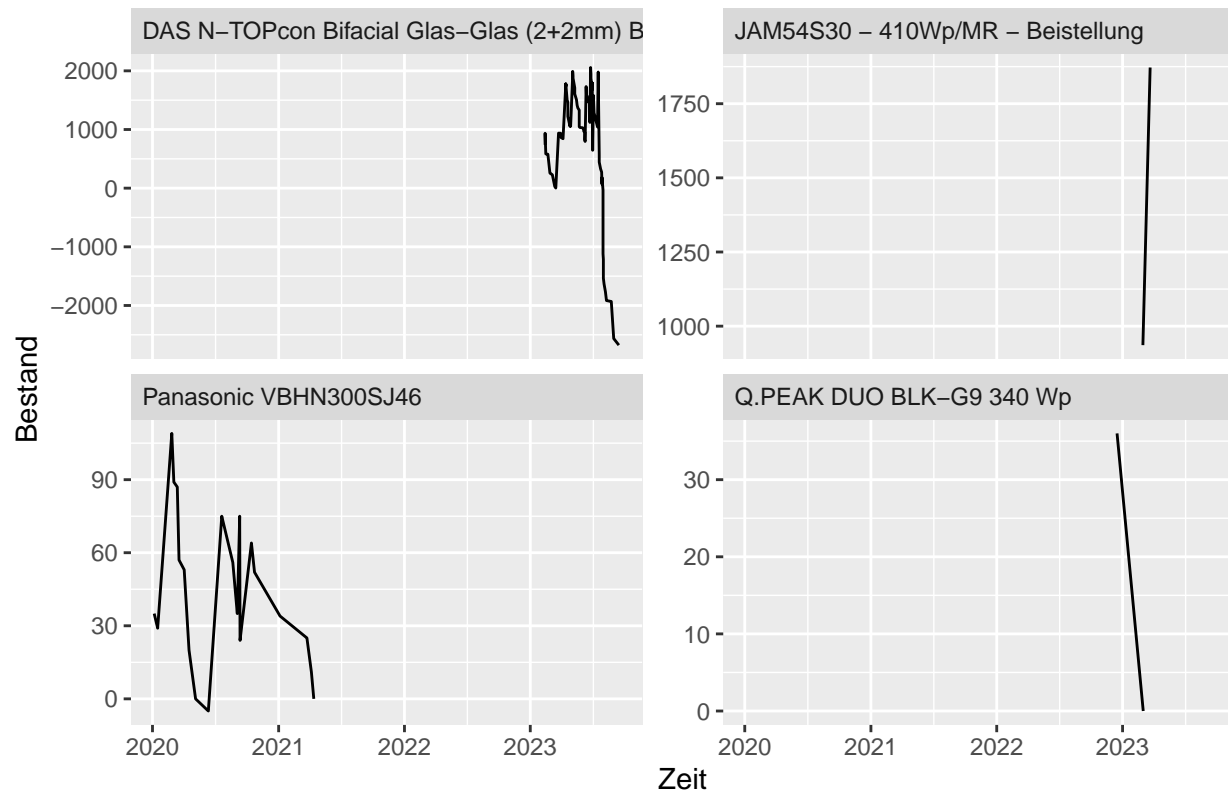


Vergleich Solarmodule

```
solarmodul_artikel <- c("DAS N-TOPcon Bifacial Glas-Glas (2+2mm) Black 415 WP",
  ↳ "Panasonic VBHN300SJ46", "Q.PEAK DUO BLK-G9 340 Wp", "JAM54S30 - 410Wp/MR -
  ↳ Beistellung")

ggplot(data=materials %>% filter(`Artikel-Bezeichnung` %in% solarmodul_artikel),
  aes(x=EinAusDat, y=Bestand, fill=`ArtikelNr`)) +
  facet_wrap(~`Artikel-Bezeichnung`, nrow=2, scales="free_y") +
  geom_line() +
  xlab("Zeit") +
  ylab("Bestand") +
  theme(legend.position = "none") +
  ggtitle("Verläufe Solarmodulbestand")+
  # Text linksbündig
  theme(strip.text.x = element_text(hjust=0.0))
```

Verläufe Solarmodulbestand

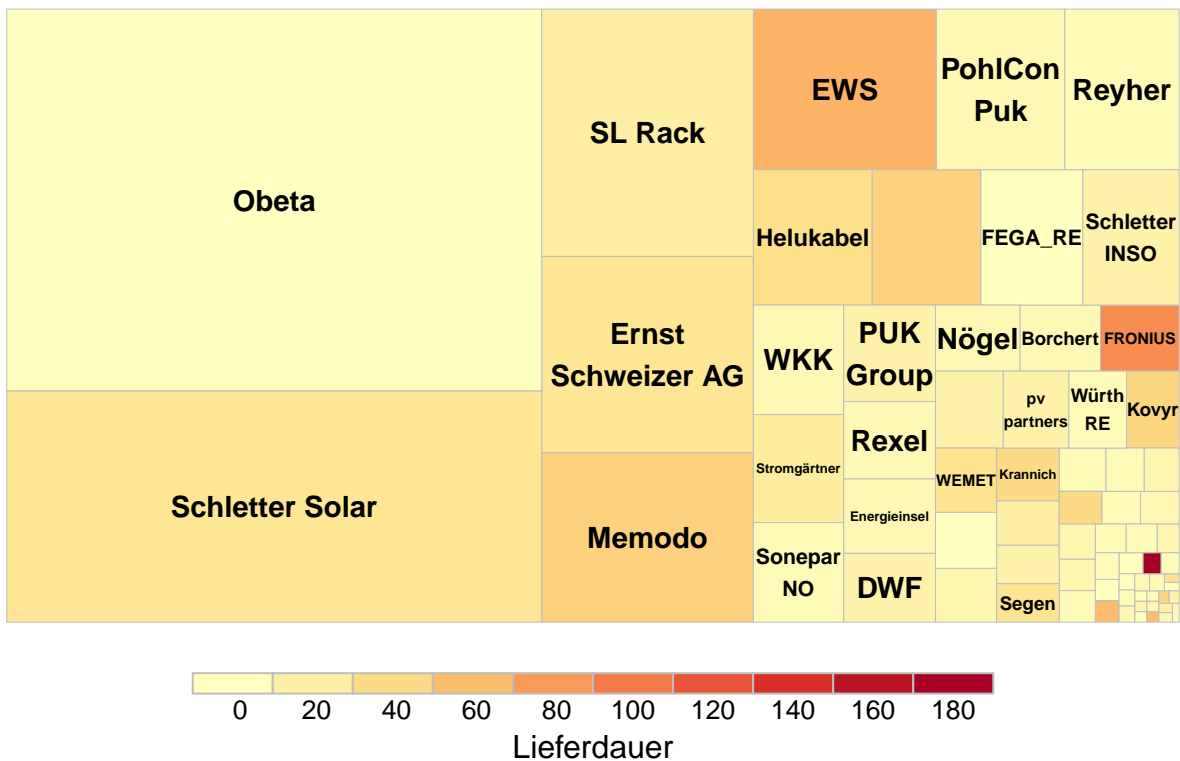


Lieferanten

```
# Lieferanten Dataframe erstellen
lieferanten <- materials %>%
  filter(EinAutDatTyp == "geliefertAm") %>%
  group_by(Lieferant=ORGAKz1) %>%
  summarize(Anzahl_Bestellungen = n(),
            Anzahl_untersch_Materialien = n_distinct(`Artikel-Bezeichnung`),
            Lieferdauer = as.numeric(mean(EinAusDat - BstDat))) %>%
  mutate(Anteil_Bestellungen = Anzahl_Bestellungen/sum(Anzahl_Bestellungen)) %>%
  arrange(desc(Anzahl_Bestellungen))

# Treemap von Lieferanten
treemap(lieferanten, index="Lieferant", vSize="Anzahl_Bestellungen",
  ↪ vColor="Lieferdauer", type="value", palette="-RdYlGn", border.col="grey", border.lwds
  ↪ = 0.2)
```

Anzahl_Bestellungen

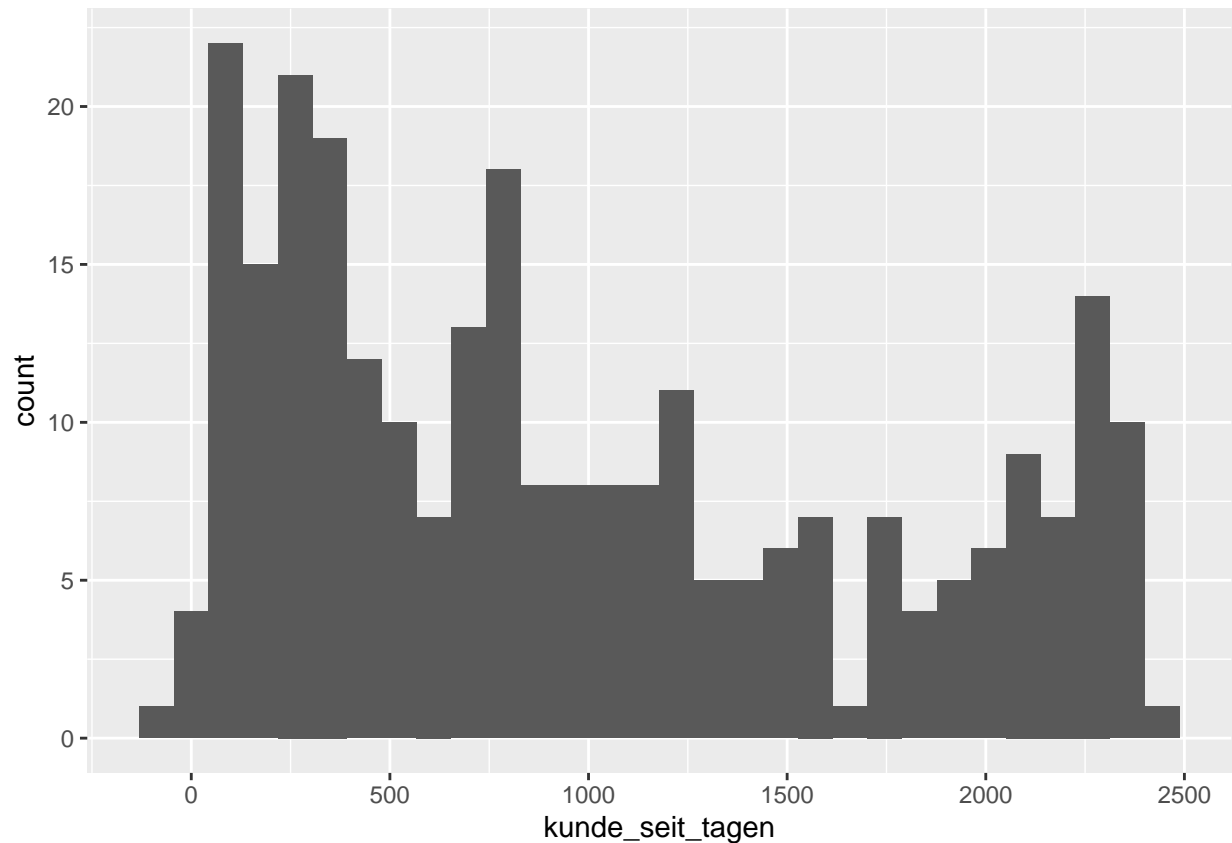


Kunden

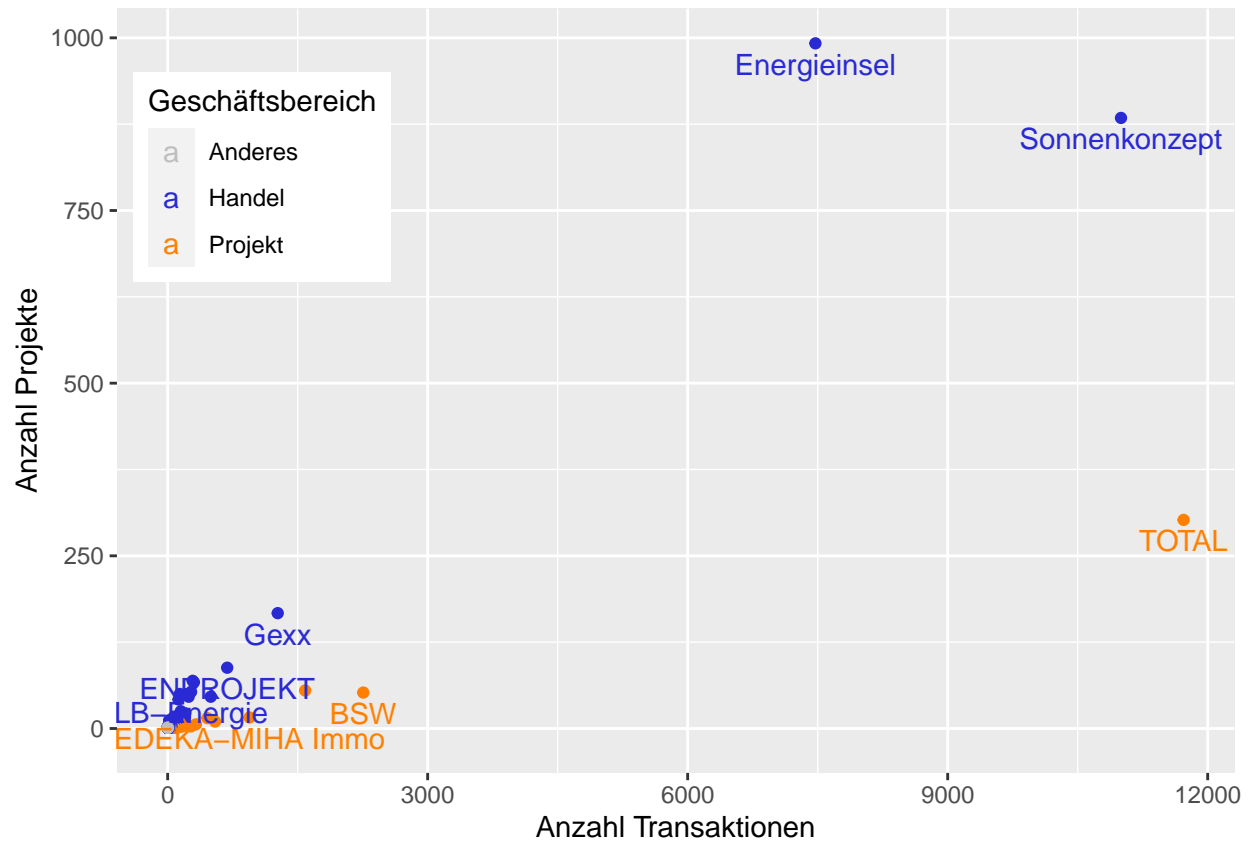
```
# Tabelle
mymode <- function(x) {
  t <- table(x)
  names(t)[ which.max(t) ]
}

kunden <- materials %>%
  filter(Menge < 0) %>%
  group_by(Kunde=ORGAKz1) %>%
  summarize(Anzahl_Transaktionen = n(),
            Anzahl_Projekte = n_distinct(PjNr),
            Kunde_seit = min(EinAusDat, na.rm=TRUE),
            Geschäftsbereich=mymode(klasse)) %>%
  arrange(desc(Anzahl_Transaktionen))

# Plot der Kundenalter
kunden %>% mutate(kunde_seit_tagen=as.numeric(today()-Kunde_seit)) %>%
  ggplot()+geom_histogram(aes(x=kunde_seit_tagen))
```



```
# Plot: Gegenüberstellung der Anzahl der Projekte und Transaktionen
ggplot(kunden, aes(x=Anzahl_Transaktionen, y=Anzahl_Projekte, label=Kunde,
  ↳ color=Geschäftsbereich))+
  geom_point(show.legend = FALSE)+
  geom_text(hjust=0.5, vjust=1, nudge_y=-15, check_overlap=TRUE)+
  labs(x="Anzahl Transaktionen", y="Anzahl Projekte")+
  scale_color_manual(values=c("Handel" = "#2A2AD5",
                              "Projekt" = "#FF8000",
                              "Einkauf" = "#00FFFF",
                              "Anderes" = "grey"))+
  geom_point(size = -1, aes(fill = Geschäftsbereich)) +
  theme(legend.position=c(.13, .78))
```



Datenqualität

```
# Stimmen ArtikelNr und Bezeichnung immer überein? -> Ja
materials %>% group_by(`Artikel-Bezeichnung`) %>% summarize(a=n_distinct(ArtikelNr)) %>%
  ↪ arrange(desc(a))
```

```
## # A tibble: 516 x 2
##   `Artikel-Bezeichnung`      a
##   <fct>                  <int>
## 1 ---NEUE BESTELLUNG---          1
## 2 100m-Ring Solarkabel H1Z2Z2-K EN50618 1x10mm² sw          1
## 3 100m-Ring Solarkabel H1Z2Z2-K EN50618 1x6mm² sw           1
## 4 15m - CAT6 Netzwerkkabel CAT.6 Patchkabel                 1
## 5 20m - CAT6 Netzwerkkabel CAT.6 Patchkabel                 1
## 6 3d-U-Scheiben      6,4   M 6      DIN 9021  A2            1
## 7 3d-U-Scheiben      8,4   M 8      DIN 9021  A2            1
## 8 3d-U-Scheiben     10,5  M10      DIN 9021  A2            1
## 9 3M 2234 Kabelmantelreparaturband                          1
## 10 500m-Trommel PV-Kabel Solarkabel H1Z2Z2-K EN50618 1x6mm² sw  1
## # i 506 more rows
```

```
# Zeile ohne EinAusDat
materials %>% filter(is.na(EinAusDat)) %>% select(EinAusDat, BstDat, ORGAKz1, Menge,
  ↪ `Artikel-Bezeichnung`)
```

```
## # A tibble: 131 x 5
```

```
##      EinAusDat BstDat ORGAKzl      Menge `Artikel-Bezeichnung`
##      <date>    <date> <fct>      <dbl> <fct>
##  1 NA          NA      MBGe        -105 Cat.7 Duplex-Datenleitung Draka UC900 S~
##  2 NA          NA      Q1 Energie AG  -1 Schrumpfschlauch TREDUX-24/8-GNYE      2~
##  3 NA          NA      MBG          -40 Draht Rd-8 - DEHNALU - weich DIN 48801~
##  4 NA          NA      MBG          -70 MV-Klemme AL. 8-10/8-10mm Dehn 390051
##  5 NA          NA      MBG          -70 UNI-Erdungsklemme mit Schr. M8 u. Sperr~
##  6 NA          NA      Q1 Energie AG  -1 Korrosionsschutzbinde 50x10000 mm ~
##  7 NA          NA      MBG          -32 Überbrückungsseil 300 mm Al-Cu
##  8 NA          NA      MBGe         -24 FINTECH Alu Steckrohr IESR 25 AL (3m)
##  9 NA          NA      MBGe         -19 FINTECH Alu Steckrohr IESR 40 AL (3m)
## 10 NA          NA      MBGe         -10 FINTECH Alu Steckmuffe IESM 25 AL
## # i 121 more rows
```

```
# Zeilen mit Ausreißern
materials %>% filter(Menge < -15000)
```

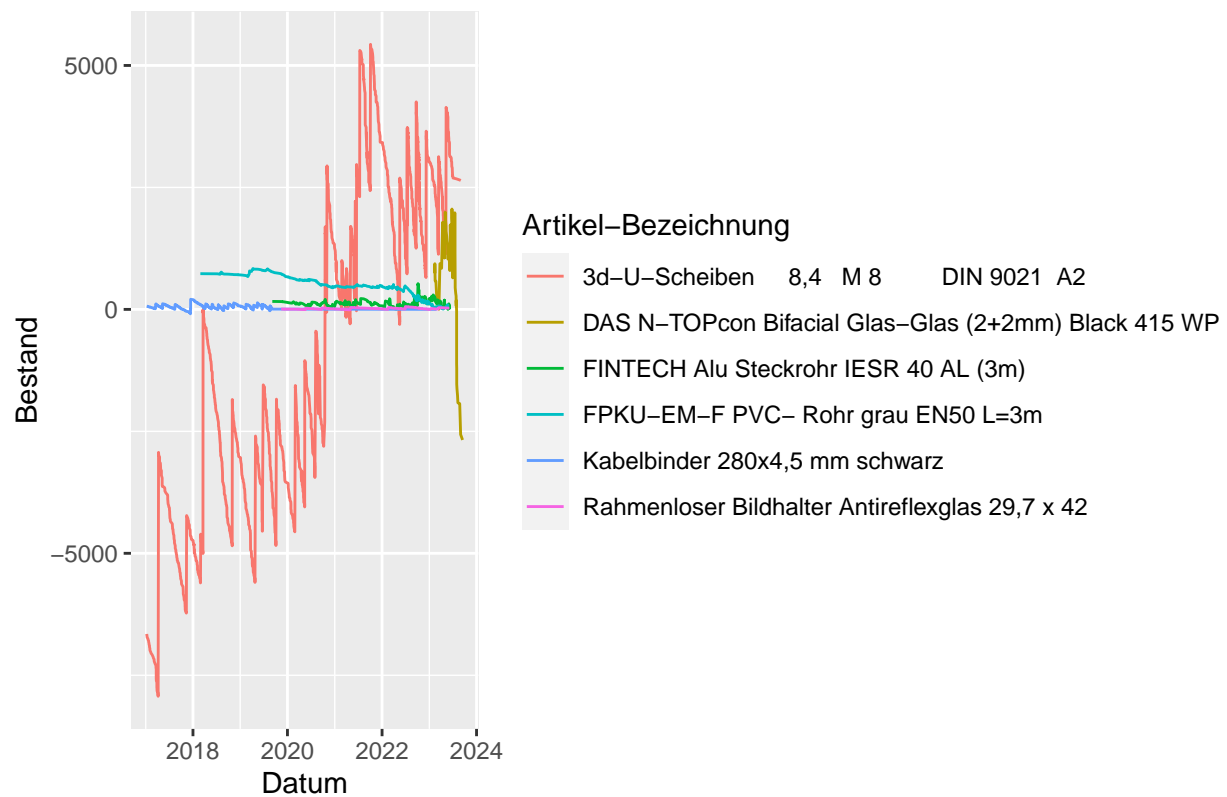
```
## # A tibble: 1 x 11
##   ArtikelNr `Artikel-Bezeichnung` EinAusDat EinAutDatTyp BstDat Menge
##   <fct>      <fct>                <date>    <fct>          <date> <dbl>
## 1 15031      Schraube 8x120 VA Tellerk~ 2022-07-01 PjPhaseDat   NA    -18750
## # i 5 more variables: Bestand <dbl>, PjNr <fct>, PjInfo <fct>, ORGAKzl <fct>,
## #   klasse <fct>
```

Bestand

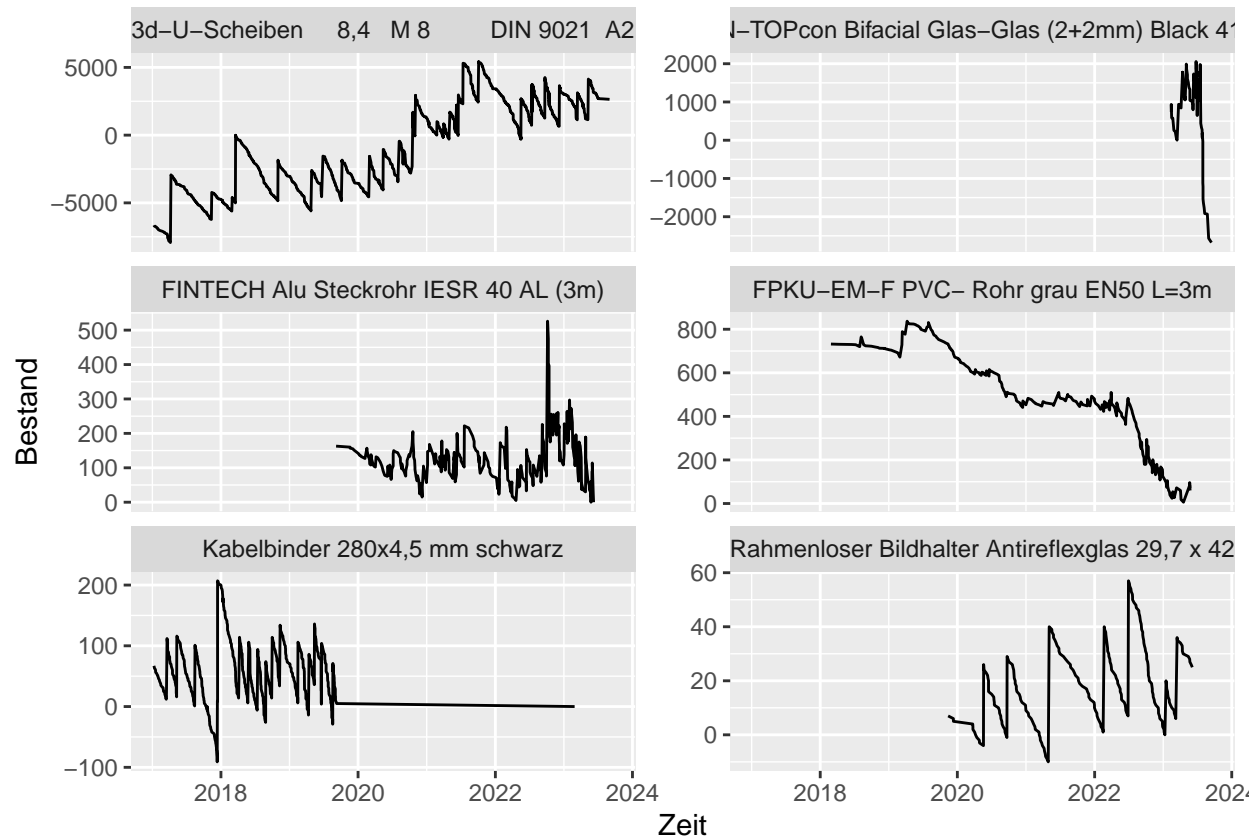
Plot: Bestand über Zeit

```
materials_b %>%
  ggplot() +
  geom_line(aes(x=EinAusDat,y=Bestand,group=`Artikel-Bezeichnung`,
    ↪ color=`Artikel-Bezeichnung`
  ))+
  xlab('Datum')+
  ggtitle('Lagerbestand über die Zeit')
```


Lagerbestand über die Zeit



```
ggplot(data=materials_b, aes(x=EinAusDat, y=Bestand)) +
  facet_wrap(~`Artikel-Bezeichnung`, nrow=3, scales="free_y") +
  geom_line() +
  xlab("Zeit") +
  ylab("Bestand") +
  theme(legend.position = "none")
```

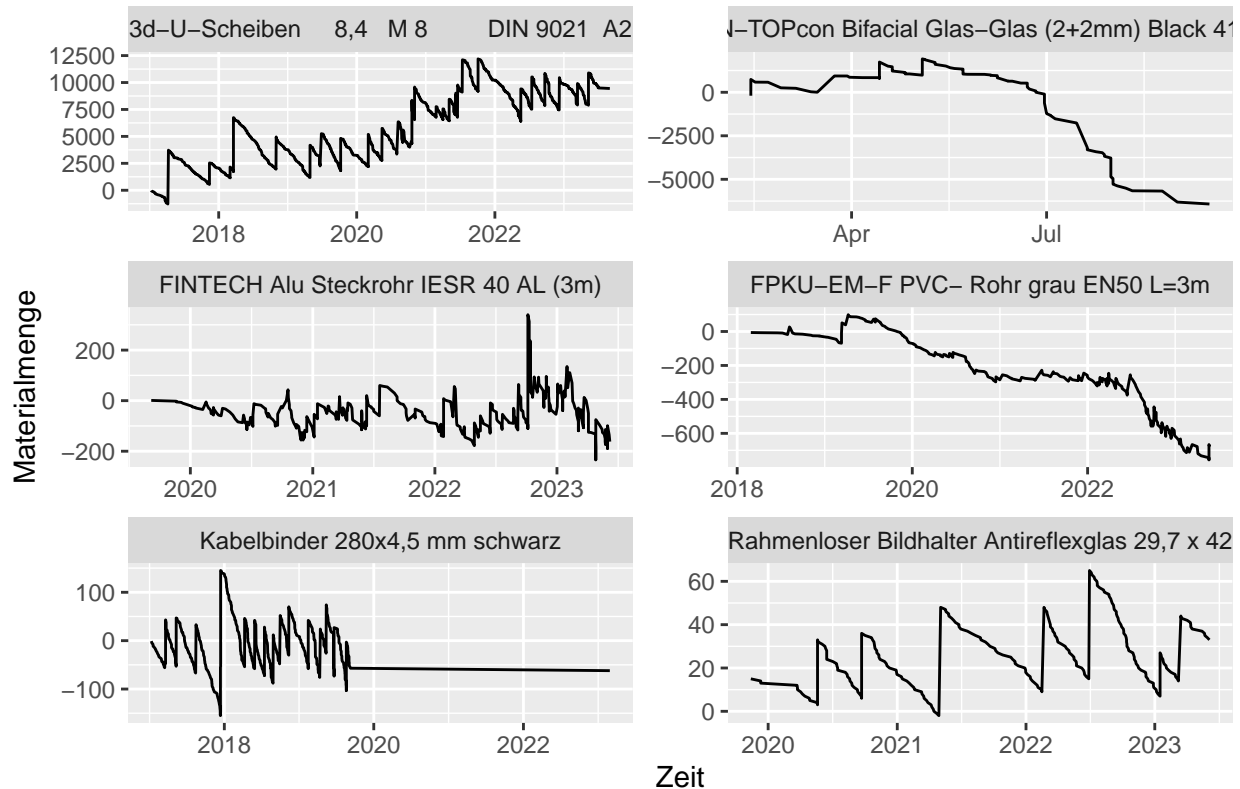


Plot: Menge über Zeit

```
mat_verlauf <- materials_b %>%
  filter(EinAutDatTyp == "geliefertAm" | EinAutDatTyp == "BereitDat" | EinAutDatTyp ==
    ↪ "PjPhaseDat") %>%
  group_by(`ArtikelNr`) %>%
  mutate(cum_sum = cumsum(Menge))

mat_verlauf %>%
  ggplot() +
    geom_line(aes(x=EinAusDat, y=cum_sum, fill=`ArtikelNr`)) +
    facet_wrap(~`Artikel-Bezeichnung`, nrow=3, scales="free") +
    xlab("Zeit") +
    ylab("Materialmenge") +
    theme(legend.position = "none") +
    ggtitle("EinAutDatTyp = geliefertAm, BereitDat, PjPhaseDat")
```

EinAusDatTyp = geliefertAm, BereitDat, PjPhaseDat

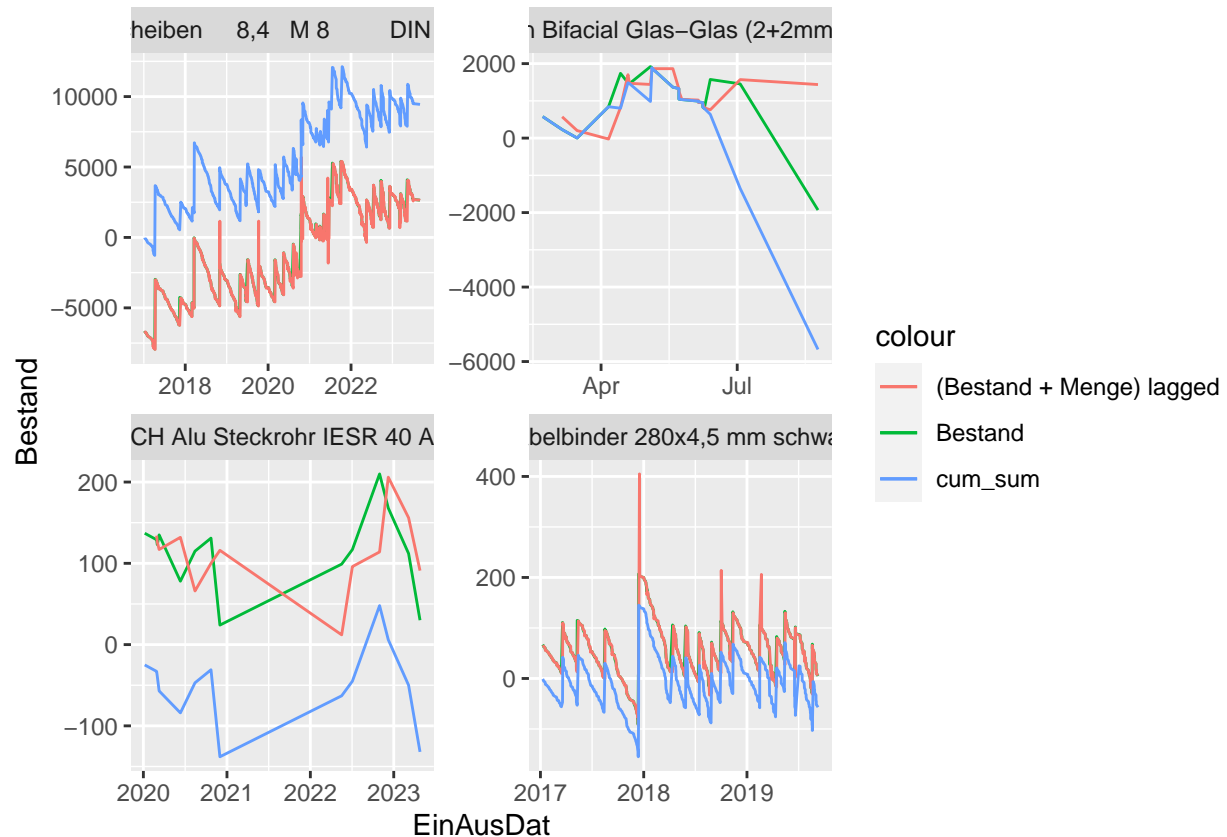


Überprüfung des Bestands auf Integrität

```
b <- mat_verlauf %>% group_by(`Artikel-Bezeichnung`) %>% filter(klasse=='Handel') %>%
  ↳ mutate(bestand_lagged = Menge + Bestand) %>% mutate(bestand_aprox =
  ↳ lag(bestand_lagged))
b %>% summarise(falscher_lagerbestand_prozent = sum(bestand_lagged == bestand_aprox,
  ↳ na.rm = TRUE)/n(), falscher_lagerbestand_prozent_cum_sum = sum(cum_sum ==
  ↳ bestand_aprox, na.rm = TRUE)/n())
```

```
## # A tibble: 4 x 3
##   `Artikel-Bezeichnung`      falscher_lagerbestan~1 falscher_lagerbestan~2
##   <fct>                  <dbl>                  <dbl>
## 1 3d-U-Scheiben      8,4 M 8 ~      0.0154                  0
## 2 DAS N-TOPcon Bifacial Glas-Glas~      0                  0.0435
## 3 FINTECH Alu Steckrohr IESR 40 A~      0                  0
## 4 Kabelbinder 280x4,5 mm schwarz      0.0918                  0
## # i abbreviated names: 1: falscher_lagerbestand_prozent,
## # 2: falscher_lagerbestand_prozent_cum_sum
```

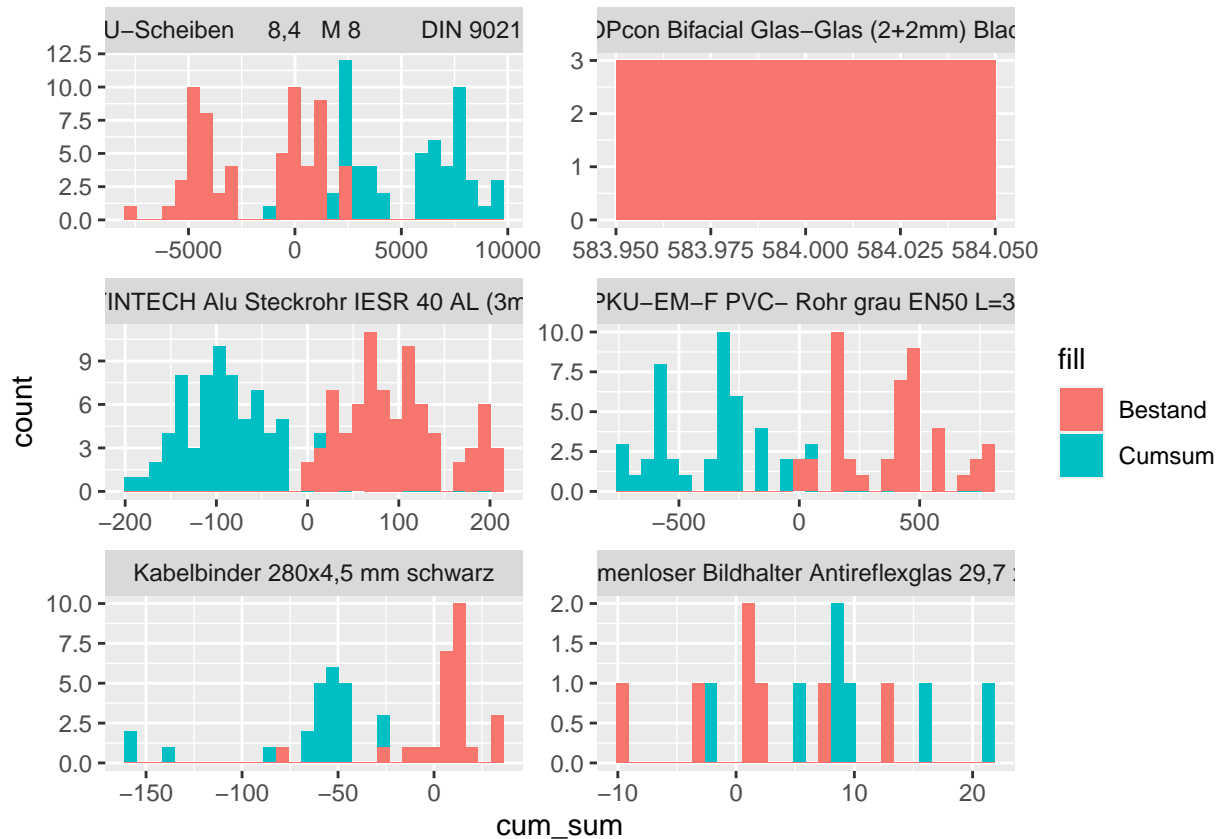
```
b %>% ggplot(aes(x =EinAusDat)) +
  geom_line(aes(y=Bestand, color= "Bestand")) + geom_line(aes(y=bestand_aprox, color=
  ↳ "(Bestand + Menge) lagged")) +
  geom_line(aes(y=cum_sum, color= "cum_sum"))+
  facet_wrap(~`Artikel-Bezeichnung`, nrow=3, scales="free")
```



Wir können sehen dass an gewissen Punkten Lagerbestände falsch angegeben sind. Dies kann man dadurch belegen, dass bei Warenausgang oder Wareneingang, der Bestand in der nächste Transaktion dem vorherigen um die Veränderung entsprechen muss.

Abrufmenge bis zum nächsten Bestellpunkt

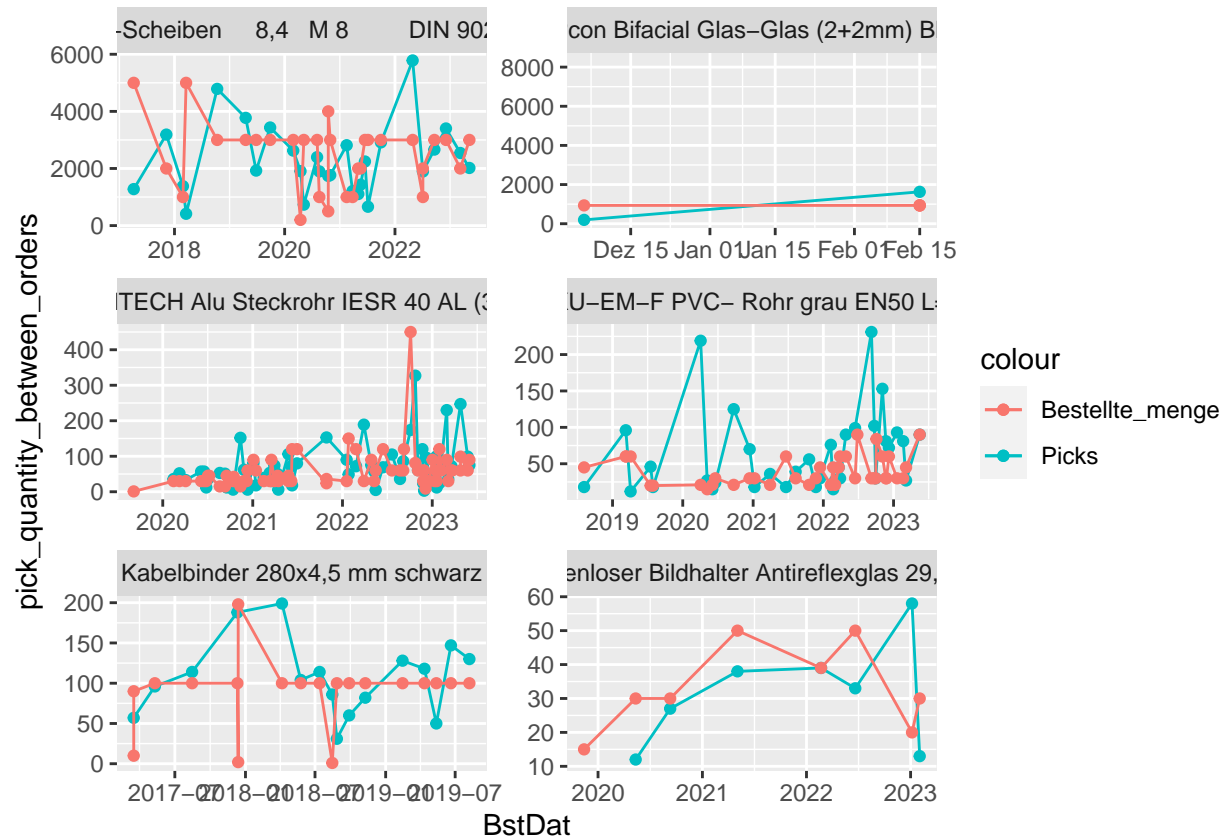
```
# Letzter Abruf vor jeder Bestellung mit as_of_join bestimmt
ordered_quantity <- mat_verlauf %>% filter(EinAusDatTyp == 'geliefertAm')
ordered_quantity %>%
  select(ArtikelNr, BstDat,) %>% left_join(mat_verlauf %>%
    select(EinAusDat, EinAusDatTyp, `Artikel-Bezeichnung`, Menge, Bestand, cum_sum) %>%
    filter(Menge < 0), join_by(ArtikelNr, closest(BstDat >= EinAusDat))) %>%
  drop_na() %>%
  ggplot() + geom_histogram(aes(x=cum_sum, fill="Cumsum")) + geom_histogram(aes(x=
    ↳ Bestand, fill="Bestand")) + facet_wrap(~`Artikel-Bezeichnung`, nrow=3,
    ↳ scales="free")
```



Summe der Picks zwischen zwei Bestellzeitpunkten

```
picks_between_order <- mat_verlauf %>% group_by(`Artikel-Bezeichnung`) %>%
  ↳ arrange(EinAusDat, .by_group = TRUE) %>% fill(BstDat, .direction = "up",) %>%
  ↳ filter(Menge<0) %>% group_by(`Artikel-Bezeichnung`, BstDat) %>%
  ↳ summarise(pick_quantity_between_orders = sum(abs(Menge)))

ggplot() + geom_line(aes(x= BstDat, y = pick_quantity_between_orders ,color="Picks"),
  ↳ data = picks_between_order) + geom_point(aes(x= BstDat, y =
  ↳ pick_quantity_between_orders,color="Picks"), data = picks_between_order, ) +
  geom_line(aes(x= BstDat, y = Menge,color= "Bestellte_menge"), data=ordered_quantity) +
  geom_point(aes(x= BstDat, y = Menge, ,color= "Bestellte_menge"), data=ordered_quantity)
  ↳ +
  facet_wrap(~`Artikel-Bezeichnung`, nrow=3, scales="free")
```



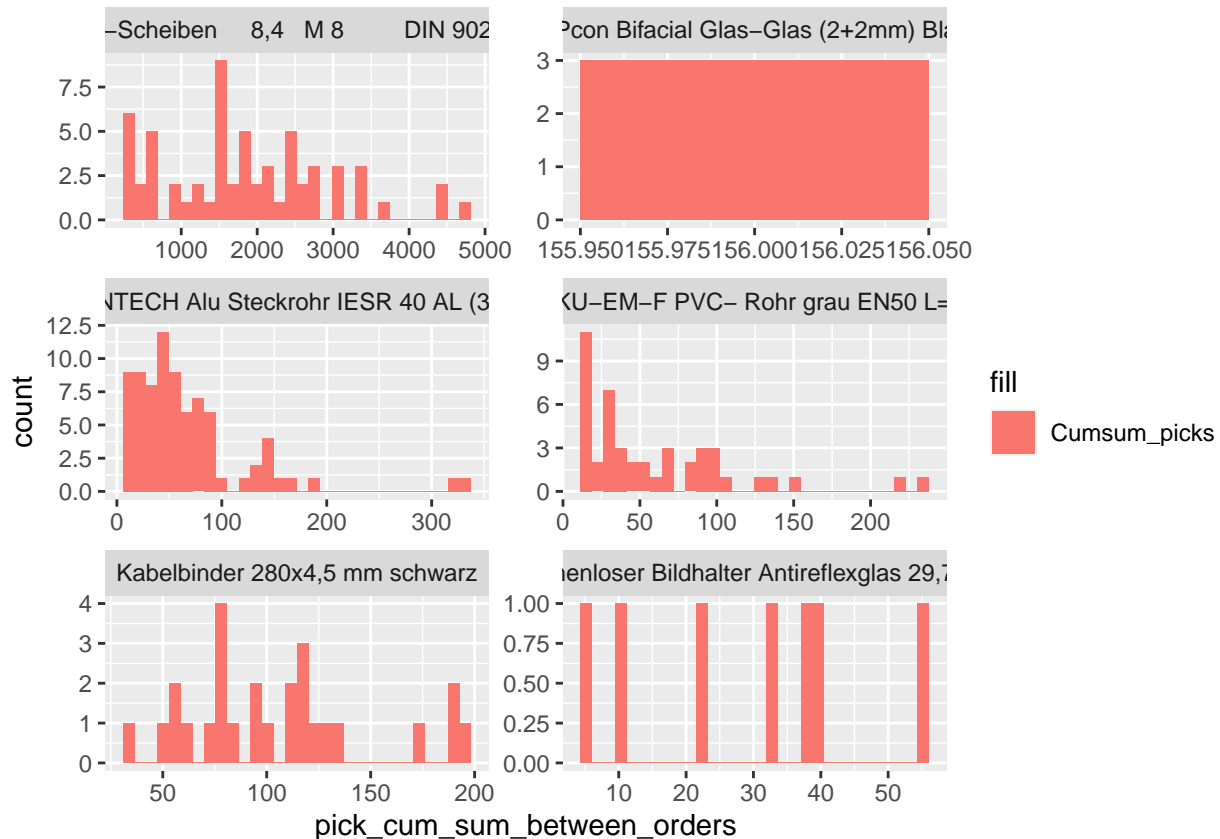
Der Plot stellt die Summe der Picks bevor einer Bestellung in Vergleich zur bestellten Menge dar. Man sieht das die Summe der Picks vor der Bestellung oft weit über der bestellten Menge liegen, das lässt auf größere Infeziente Lagerhaltung schliessen.

Kummulierte Summe der Abrufe zwischen Bestellungen als Histogramm

```
picks_cum_sum_between_order <- mat_verlauf %>%
  ↳ select(ArtikelNr, `Artikel-Bezeichnung`, EinAusDat, BstDat, Menge, ) %>%
  ↳ group_by(`Artikel-Bezeichnung`) %>% arrange(EinAusDat, .by_group = TRUE) %>%
  ↳ fill(BstDat, .direction = "up",) %>% filter(Menge<0) %>%
  ↳ group_by(`Artikel-Bezeichnung`, BstDat) %>% mutate(pick_cum_sum_between_orders =
  ↳ cumsum(abs(Menge)))

nachf_in_WBZ <- ordered_quantity %>% select(ArtikelNr, BstDat, `Artikel-Bezeichnung`) %>%
  ↳ left_join(picks_cum_sum_between_order, join_by(ArtikelNr, closest(BstDat >=
  ↳ EinAusDat)))

ggplot(nachf_in_WBZ)+ geom_histogram(aes(x=pick_cum_sum_between_orders, fill
  ↳ ="Cumsum_picks")) + facet_wrap(~`Artikel-Bezeichnung`, nrow=3, scales="free")
```



```
nachf_in_WBZ %>% group_by(ArtikelNr) %>% summarize(quantile(pick_cum_sum_between_orders,
  ↳ 0.9, na.rm=TRUE))
```

```
## # A tibble: 6 x 2
##   ArtikelNr `quantile(pick_cum_sum_between_orders, 0.9, na.rm = TRUE)`
##   <fct>                                <dbl>
## 1 2628                                116.
## 2 6203                                3378
## 3 21866                               181
## 4 26385                               141.
## 5 26535                                45.8
## 6 29523                               156
```

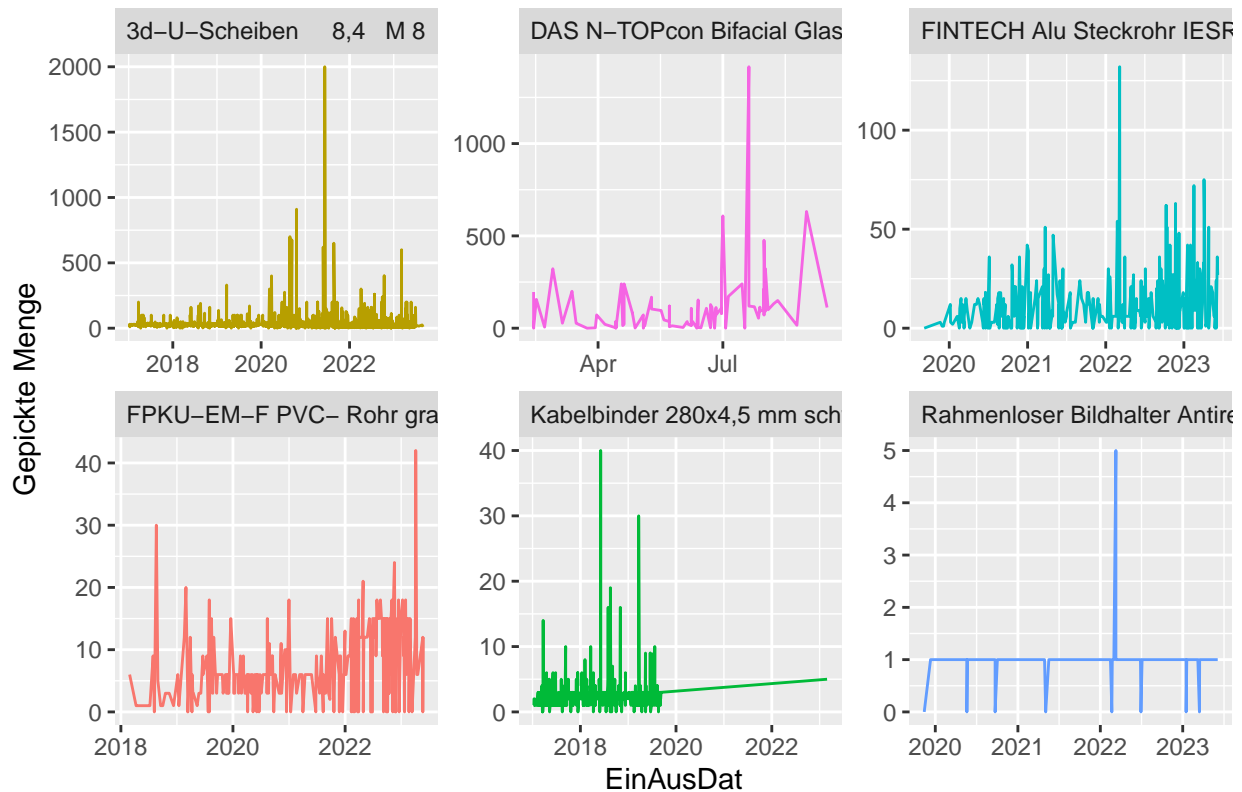
Der Plot zeigt die Verteilung der cumulierten Summe der Picks dar am Bestelltag der nächsten Bestellung. Dadurch dass der Wert nicht eindeutig ist, schließen wir das der Logistiker kein eindeutigen Bestellmenge* hat.

Picks und Losgrößen

```
picks <- materials_b %>% mutate(picks = if_else(Menge < 0, -Menge, 0), lot_size =
  ↳ if_else(Menge >= 0, abs(Menge), 0))
```

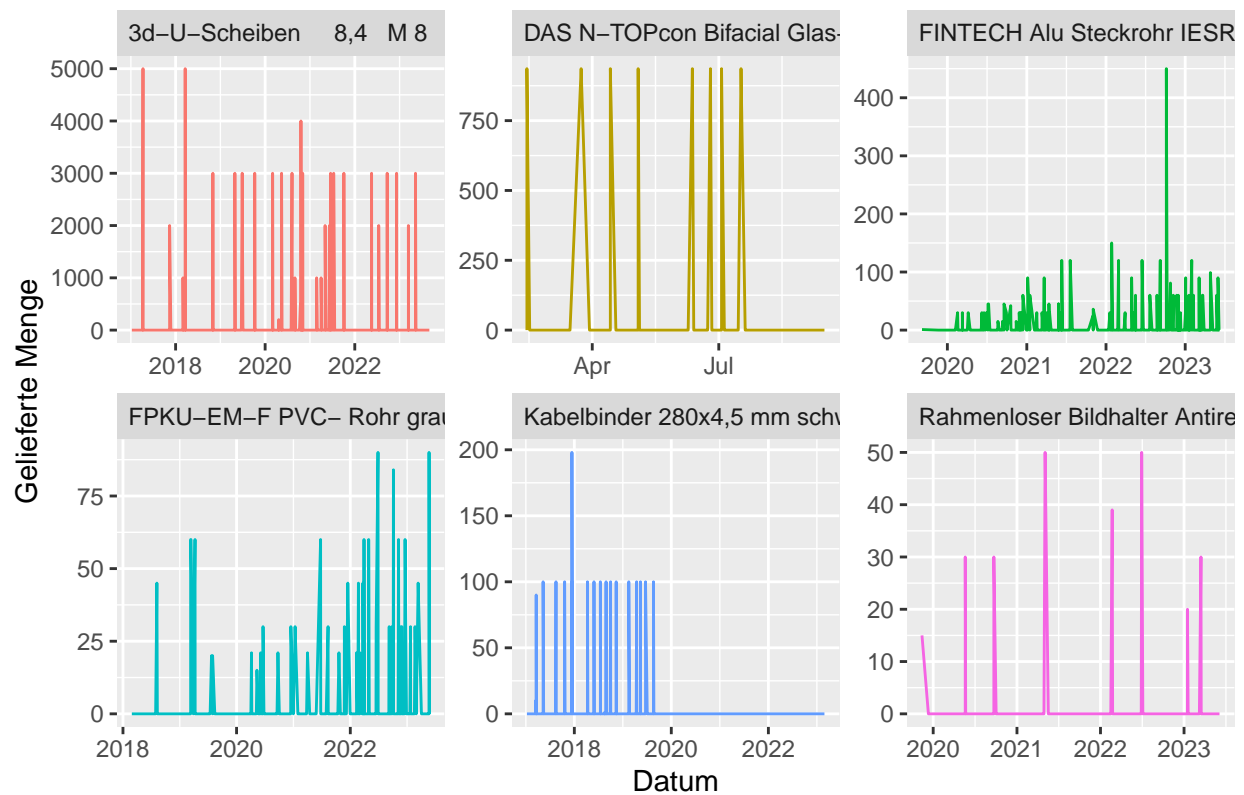
Plot: Gepickte Menge (Nachfrage)

```
picks %>%
  ggplot() +
  geom_line(aes(x=EinAusDat,y=picks, color=ArtikelNr)) +
  ylab("Gepickte Menge")+
  facet_wrap(~`Artikel-Bezeichnung`, scales="free") +
  # Text linksbündig
  theme(strip.text.x = element_text(hjust=0.0)) +
  ggtitle("") +
  theme(legend.position = "none")
```



Plot: Gelieferte Menge/Losgröße (Lieferungen)

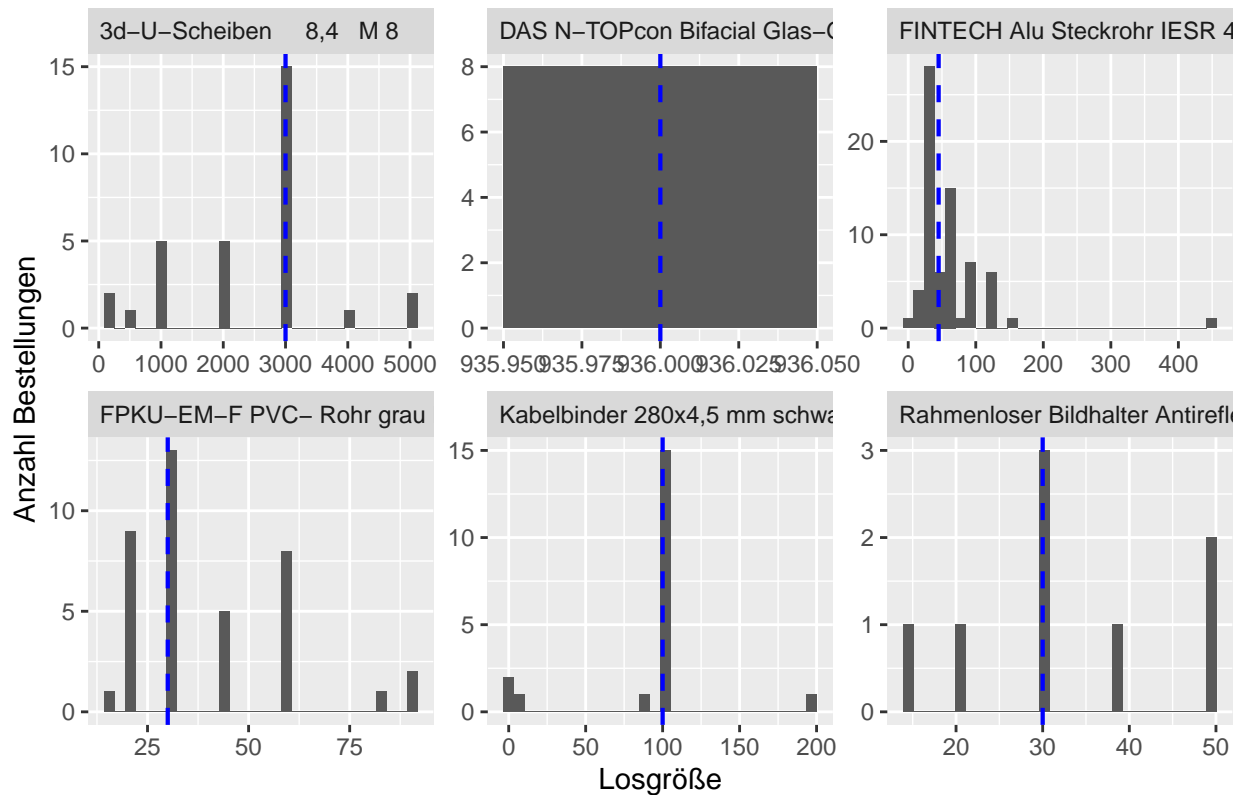
```
picks %>%
  ggplot() +
  geom_line(aes(x = EinAusDat, y = lot_size, color = `Artikel-Bezeichnung`))+
  ylab("Gelieferte Menge")+
  xlab('Datum')+
  facet_wrap(~`Artikel-Bezeichnung`, scales="free") +
  # Text linksbündig
  theme(strip.text.x = element_text(hjust=0.0)) +
  ggtitle("") +
  theme(legend.position = "none")
```

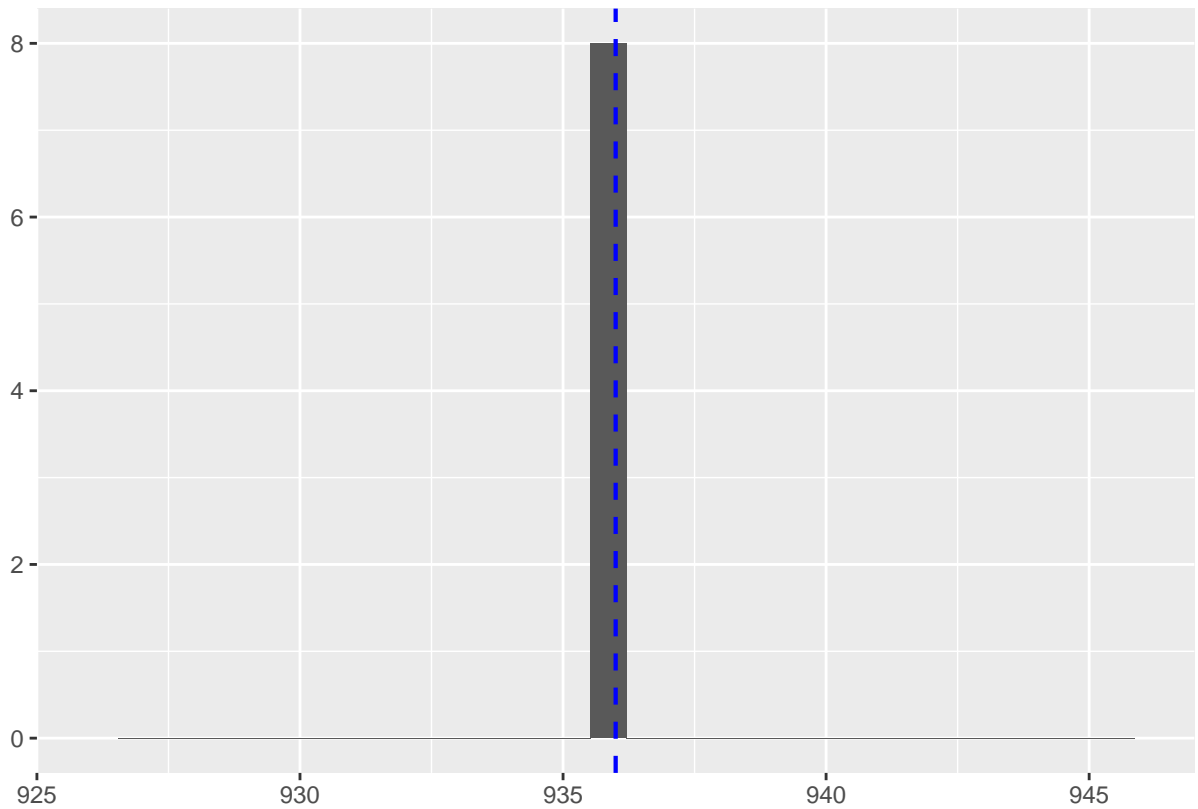
Picks vs. Losgröße nach Artikel

```
picks %>% filter.lot_size != 0) %>% group_by(`Artikel-Bezeichnung`) %>% mutate(med_lot =
  ↪ median.lot_size)) %>%
  ggplot(aes(x=lot_size)) +
  geom_histogram()+
  geom_vline(aes(xintercept=med_lot), color="blue", linetype="dashed", size=0.75)+
  facet_wrap(~`Artikel-Bezeichnung`, scales="free") +
  # Text linksbündig
  theme(strip.text.x = element_text(hjust=0.0)) +
  ggtitle("Verteilung der Bestell-Losgrößen nach Artikel")+
  labs(x="Losgröße", y="Anzahl Bestellungen")
```

Verteilung der Bestell-Losgrößen nach Artikel

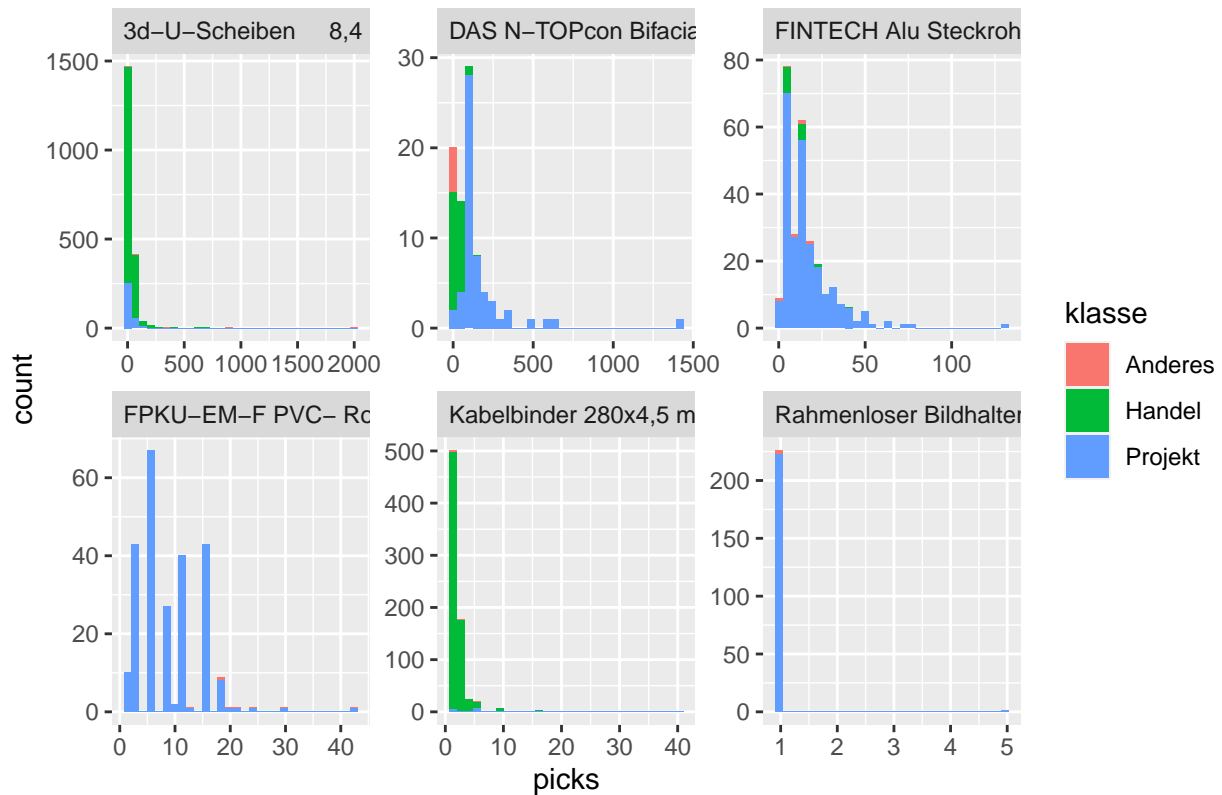


```
# Bifacial Glas separat plotten
picks %>% filter.lot_size != 0, `Artikel-Bezeichnung`=="DAS N-TOPcon Bifacial Glas-Glas
↳ (2+2mm) Black 415 WP") %>% group_by(`Artikel-Bezeichnung`) %>% mutate(med_lot =
↳ median.lot_size)) %>%
  ggplot(aes(x=lot_size)) +
  geom_histogram()+
  geom_vline(aes(xintercept=med_lot), color="blue", linetype="dashed", size=0.75)+
  # Text linksbündig
  theme(strip.text.x = element_text(hjust=0.0)) +
  labs(x="", y="")+
  xlim(936-10, 936+10)
```



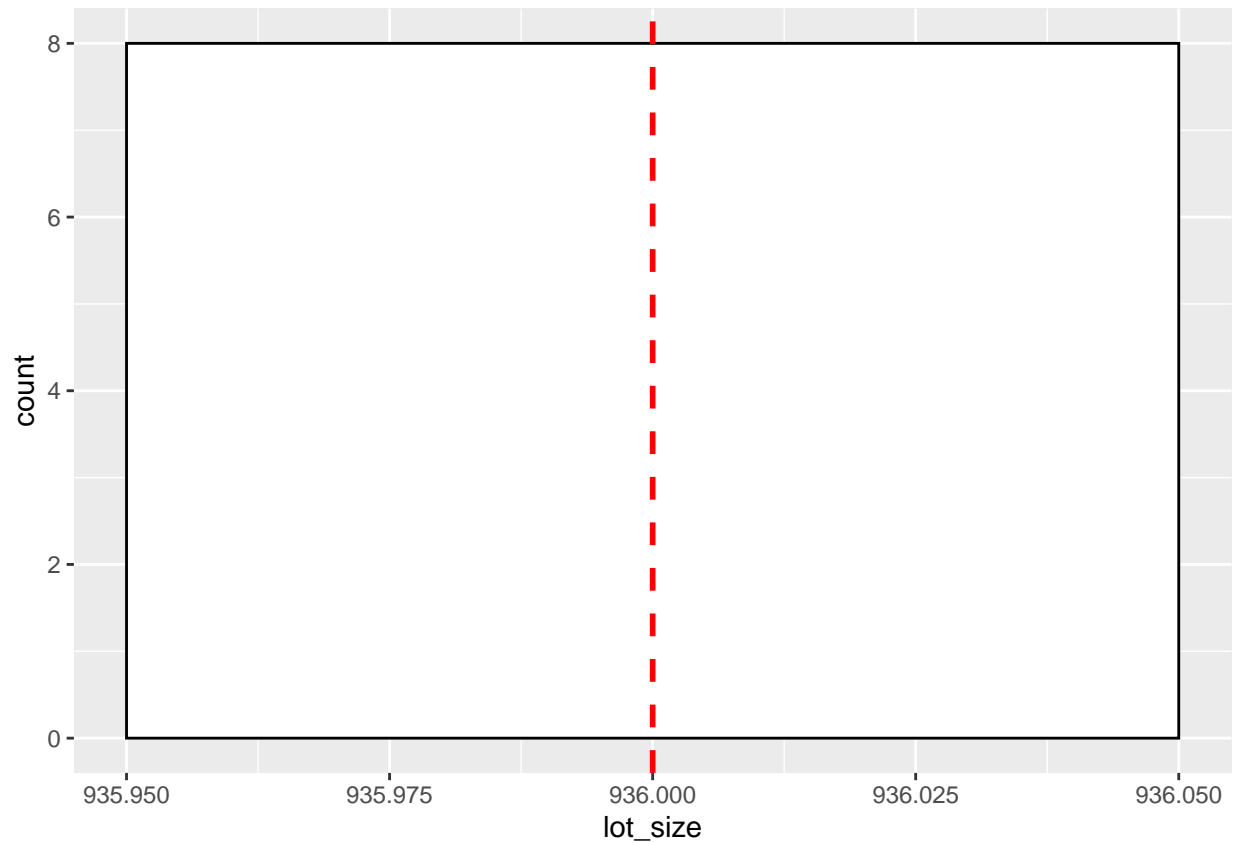
```
# Pick-Losgrößen
picks %>% filter(picks != 0) %>%
  ggplot(aes(x=picks, group=klasse, fill=klasse)) +
  geom_histogram() +
  facet_wrap(~`Artikel-Bezeichnung`, scales="free") +
  # Text linksbündig
  theme(strip.text.x = element_text(hjust=0.0)) +
  ggtitle("Verteilung der Pick-Losgrößen nach Artikel")
```

Verteilung der Pick-Losgrößen nach Artikel

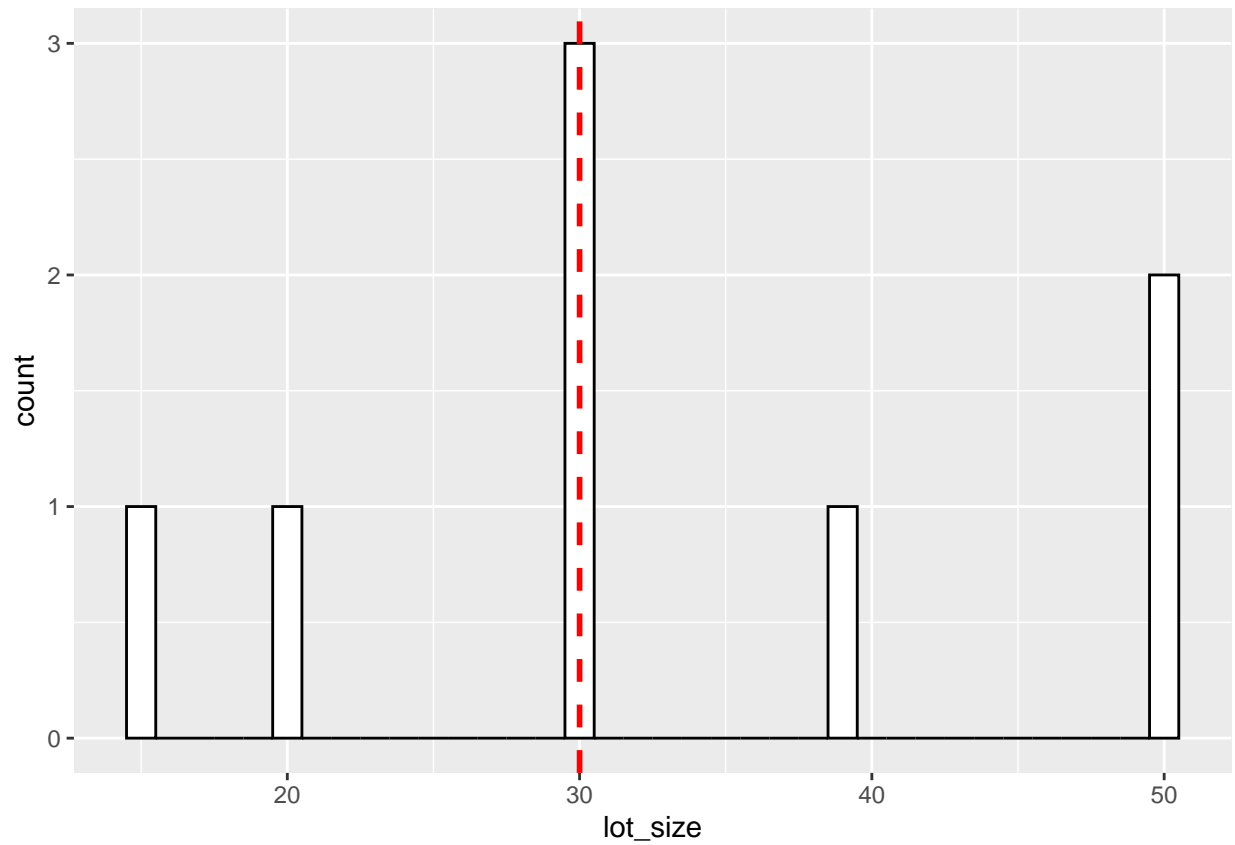


Histogramm und Median der Bestelllosgröße für jedes Produkt

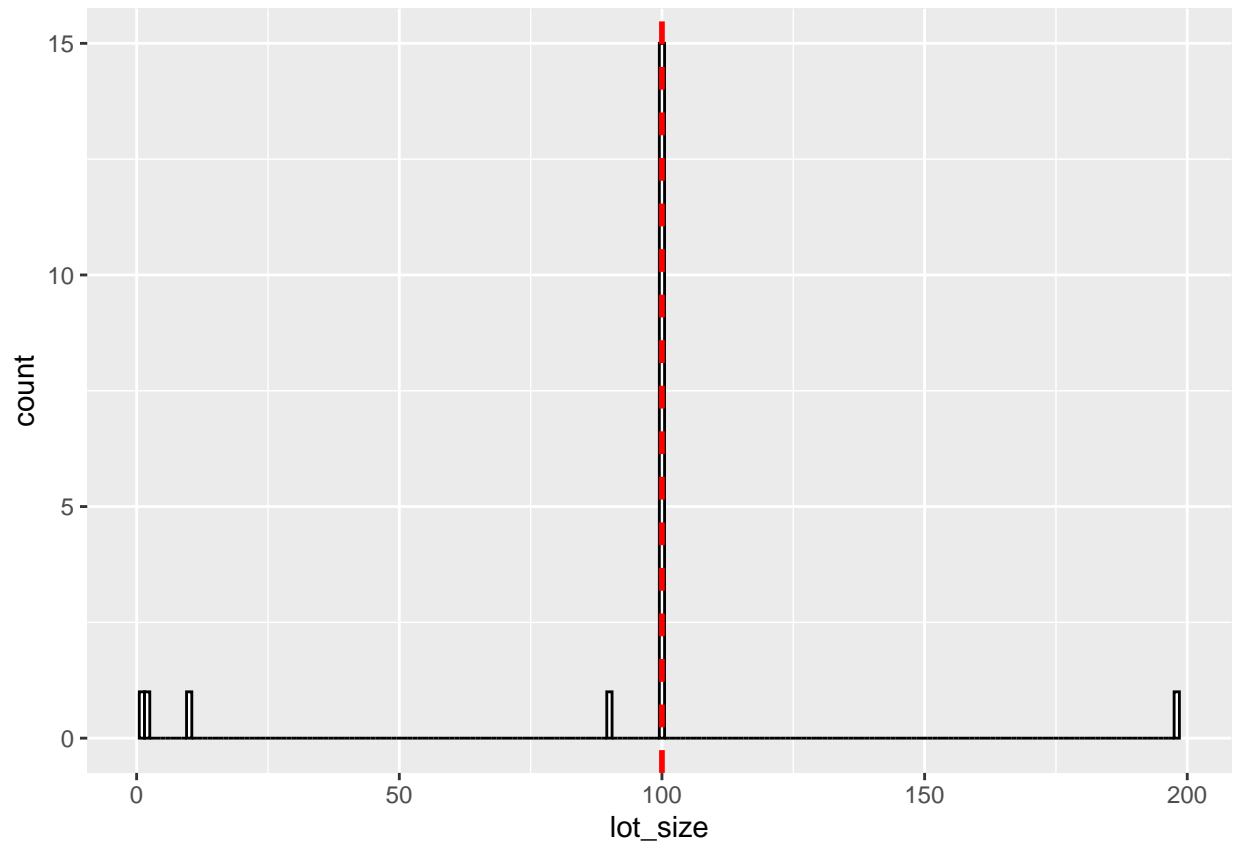
```
picks %>% filter(ArtikelNr == 29523, lot_size != 0) %>%
  ggplot(aes(x=lot_size)) +
  geom_histogram(binwidth=.1, colour="black", fill="white")+
  geom_vline(aes(xintercept=median(lot_size, na.rm=T)), color="red", linetype="dashed",
    size=1)
```



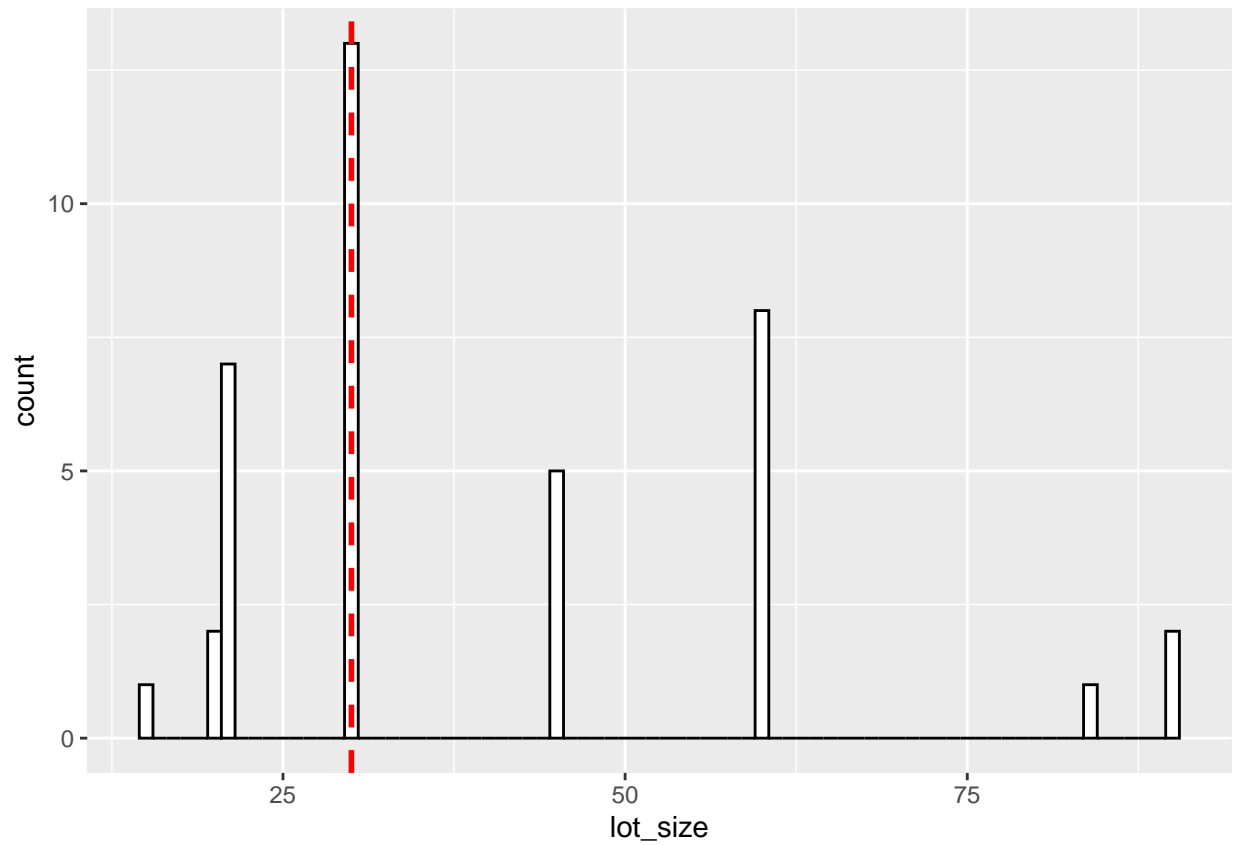
```
picks %>% filter(ArtikelNr == 26535, lot_size != 0 ) %>%
  ggplot(aes(x=lot_size)) +
  geom_histogram(binwidth=1, colour="black", fill="white") +
  geom_vline(aes(xintercept=median(lot_size, na.rm=T)), color="red", linetype="dashed",
    ↪ size=1)
```



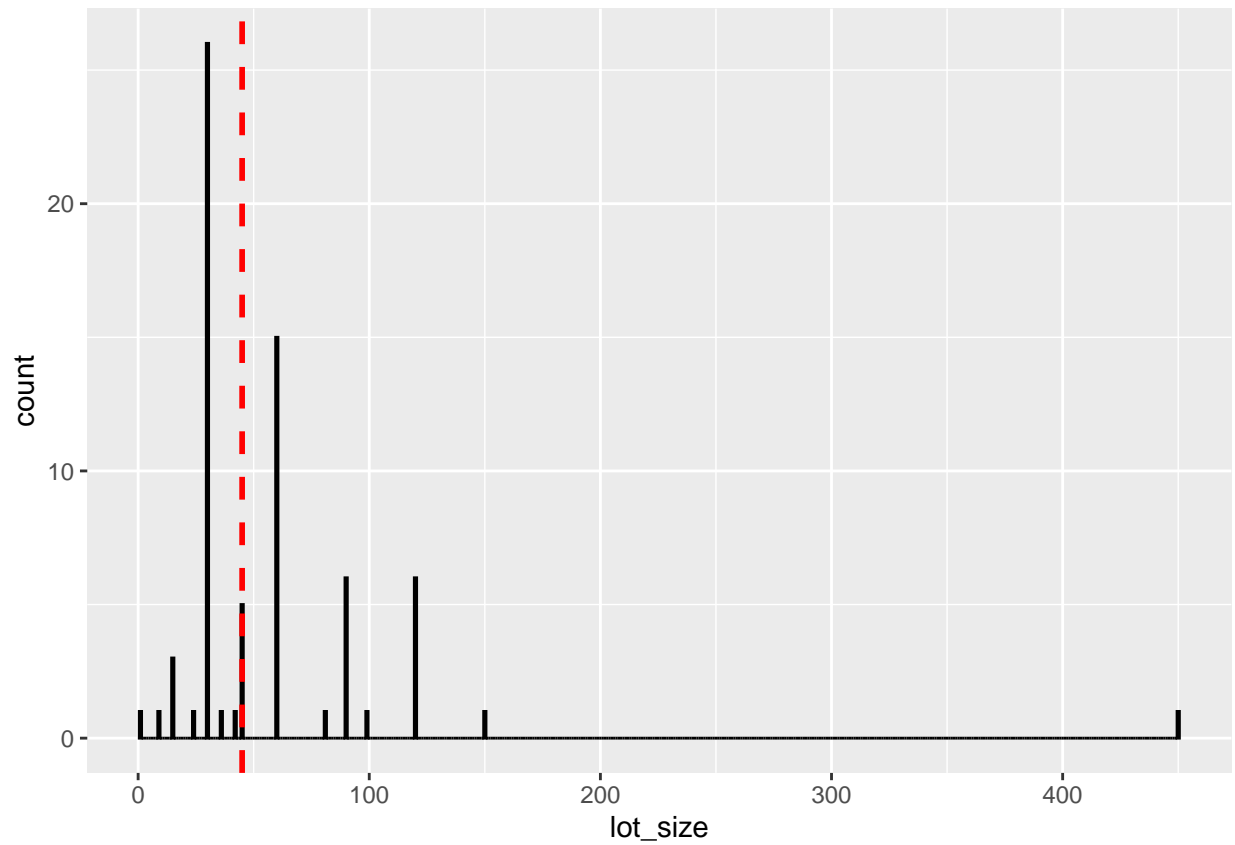
```
picks %>% filter(ArtikelNr == 21866, lot_size != 0 ) %>%  
  ggplot(aes(x=lot_size)) +  
  geom_histogram(binwidth=1, colour="black", fill="white") +  
  geom_vline(aes(xintercept=median(lot_size, na.rm=T)), color="red", linetype="dashed",  
    ↪ size=1)
```



```
picks %>% filter(ArtikelNr == 2628, lot_size != 0 ) %>%
  ggplot(aes(x=lot_size)) +
  geom_histogram(binwidth=1, colour="black", fill="white") +
  geom_vline(aes(xintercept=median(lot_size, na.rm=T)), color="red", linetype="dashed",
    ↪ size=1)
```



```
picks %>% filter(ArtikelNr == 26385, lot_size != 0 ) %>%
  ggplot(aes(x=lot_size)) +
  geom_histogram(binwidth=1, colour="black", fill="white") +
  geom_vline(aes(xintercept=median(lot_size, na.rm=T)), color="red", linetype="dashed",
    ↪ size=1)
```

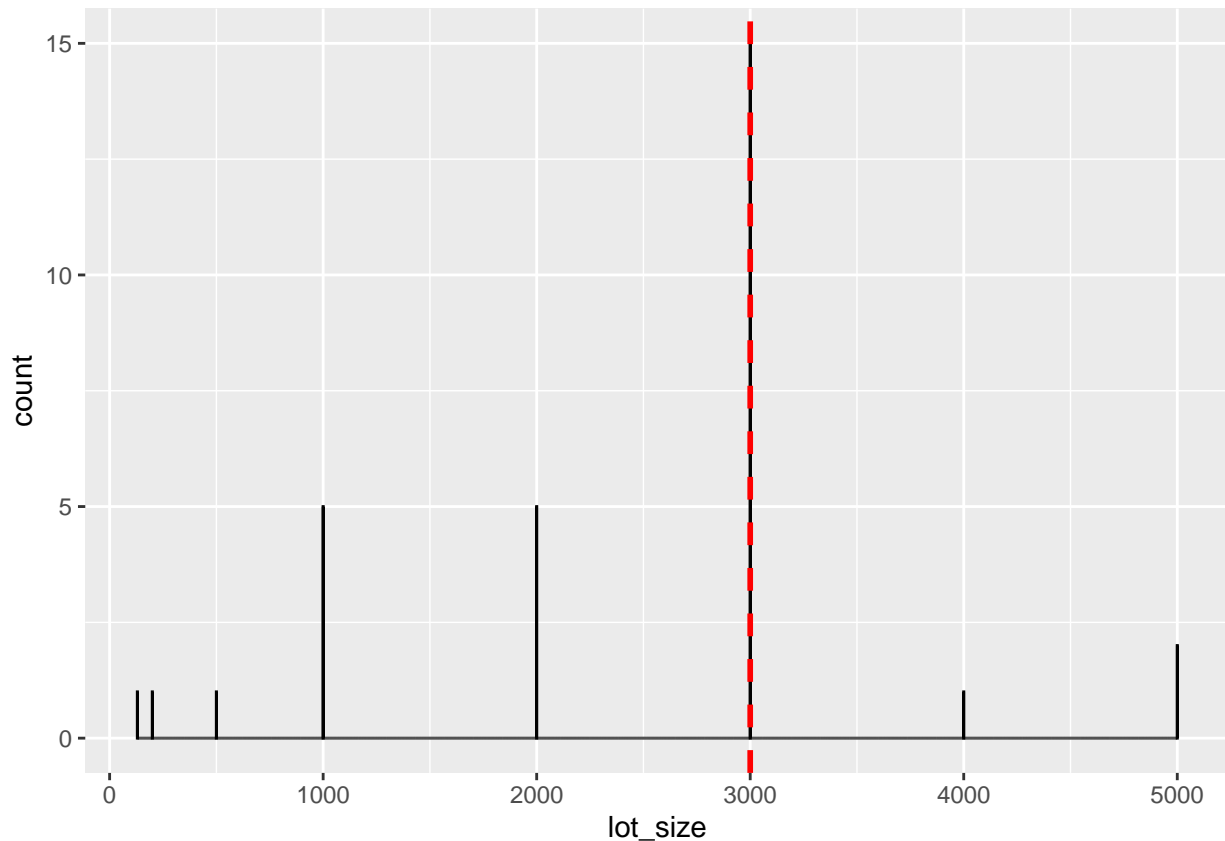


Tabelle des Medians der Bestell-Losgrößen nach Artikel

```
picks %>% filter(lot_size !=0) %>% group_by(`Artikel-Bezeichnung`) %>%
  summarise(median_lot = mean(lot_size))
```

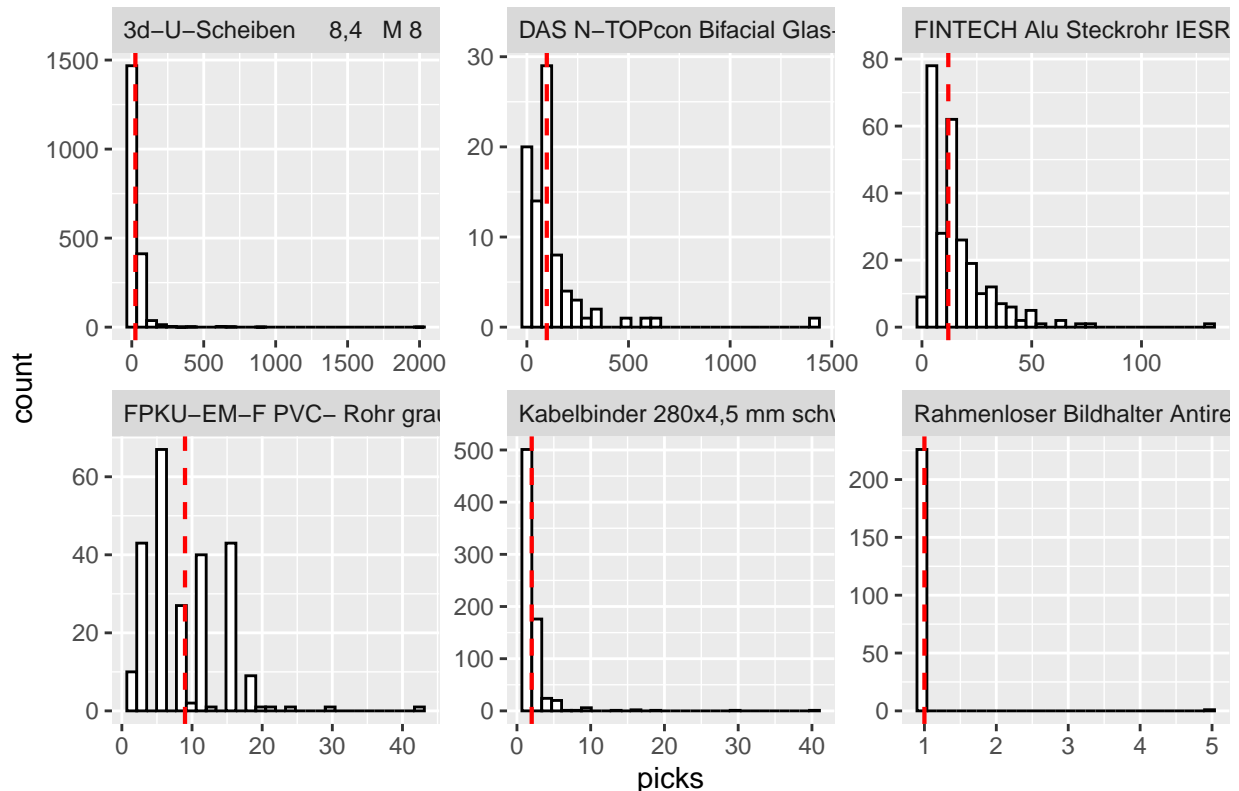
```
## # A tibble: 6 x 2
##   `Artikel-Bezeichnung`      median_lot
##   <fct>                  <dbl>
## 1 3d-U-Scheiben      8,4   M 8      DIN 9021  A2      2414.
## 2 DAS N-TOPcon Bifacial Glas-Glas (2+2mm) Black 415 WP      936
## 3 FINTECH Alu Steckrohr IESR 40 AL (3m)      58.6
## 4 FPKU-EM-F PVC- Rohr grau EN50 L=3m      40.0
## 5 Kabelbinder 280x4,5 mm schwarz      90.0
## 6 Rahmenloser Bildhalter Antireflexglas 29,7 x 42      33
```

Verteilung der Picks nach Artikel

```
picks %>% filter(picks !=0) %>% group_by(`Artikel-Bezeichnung`) %>% mutate(med_pick =
  median(picks)) %>%
  ggplot(aes(x=picks))+
  geom_histogram(colour="black", fill="white")+
  geom_vline(aes(xintercept=med_pick), color="red", linetype="dashed", size=0.75)+
  facet_wrap(~`Artikel-Bezeichnung`, scales="free") +
  # Text linksbündig
  theme(strip.text.x = element_text(hjust=0.0)) +
```

```
ggtitle("Verteilung der Picks nach Artikel")
```

Verteilung der Picks nach Artikel



Median und Durchschnitt der Picks nach Artikel

```
picks %>% filter(picks !=0) %>% group_by(`Artikel-Bezeichnung`) %>%
  summarise(median_pick = median(picks), avg_picks= mean(picks))
```

```
## # A tibble: 6 x 3
##   `Artikel-Bezeichnung`      median_pick avg_picks
##   <fct>                  <dbl>      <dbl>
## 1 3d-U-Scheiben      8,4 M 8      DIN 9021 A2      25      33.7
## 2 DAS N-TOPcon Bifacial Glas-Glas (2+2mm) Black 415 WP      98     124.
## 3 FINTECH Alu Steckrohr IESR 40 AL (3m)      12     15.9
## 4 FPKU-EM-F PVC- Rohr grau EN50 L=3m          9      9.06
## 5 Kabelbinder 280x4,5 mm schwarz          2      2.41
## 6 Rahmenloser Bildhalter Antireflexglas 29,7 x 42          1      1.02
```

Berechnen der kumulierten Nachfrage nach Wochen

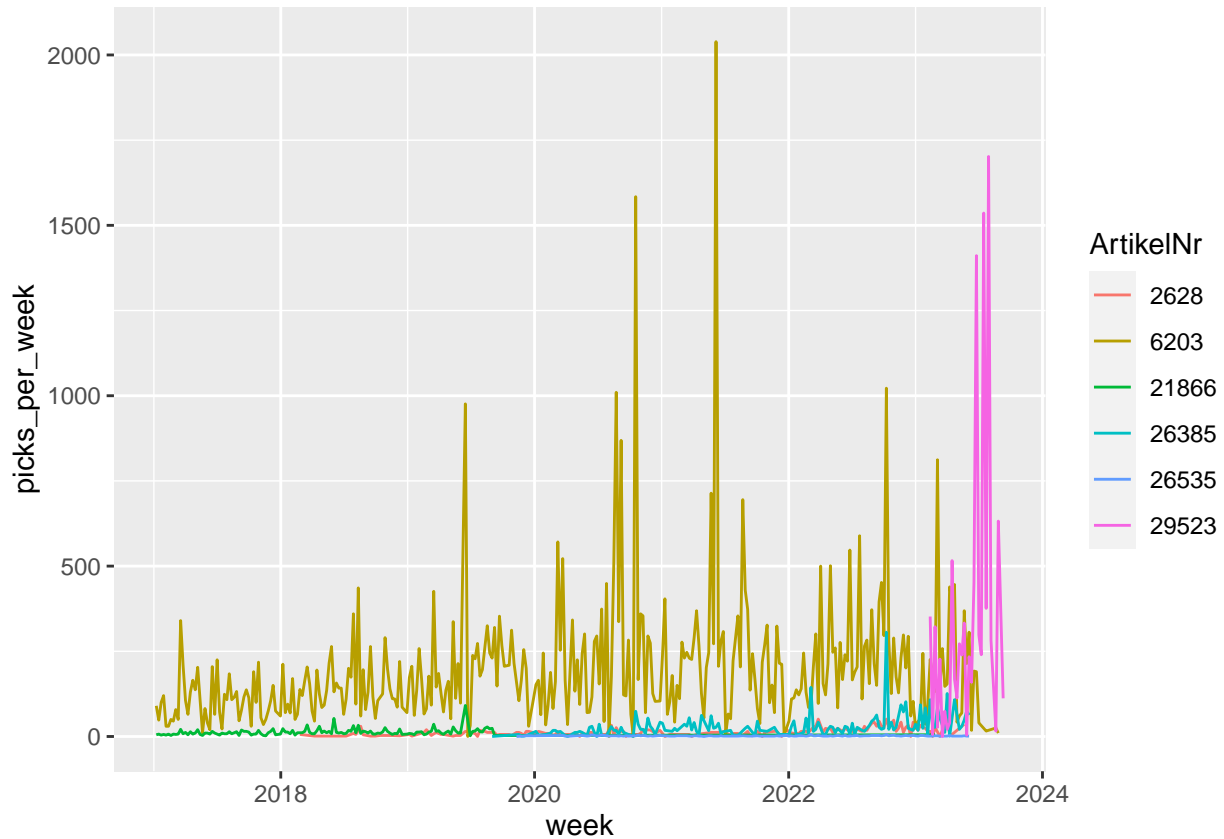
```
weekly_picks <- picks %>%
  group_by(ArtikelNr, week = floor_date(EinAusDat, 'week')) %>%
  summarise(picks_per_week= sum(picks))
```

Berechnen der Durchschnittsnachfrage nach Wochen

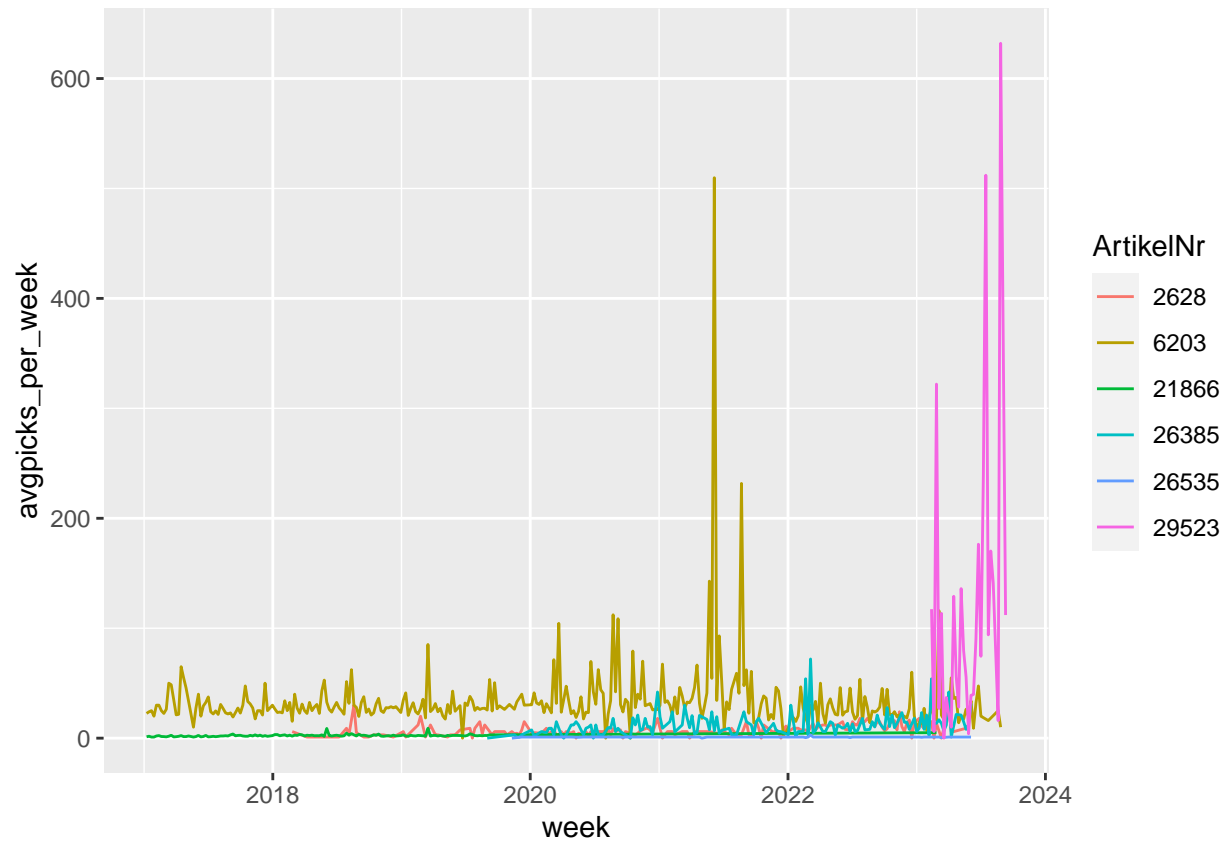
```
weekly_avg_picks <- picks %>%
  group_by(ArtikelNr, week = floor_date(EinAusDat, 'week')) %>%
  summarise(avgpicks_per_week= mean(picks))
```

```
# Berechnen der Bestell-Losgröße nach Wochen
weekly_meanpick <- weekly_picks %>% group_by(ArtikelNr) %>%
  mutate(mean_pick = mean(picks_per_week))
```

```
weekly_picks %>%
  ggplot() +
  geom_line(aes(x=week ,y=picks_per_week, group= ArtikelNr, col= ArtikelNr))
```



```
# Berechnen der Durchschnittlichen Bestell-Losgröße nach Wochen
weekly_avg_picks %>%
  ggplot() +
  geom_line(aes(x=week ,y=avgpicks_per_week, group= ArtikelNr, col= ArtikelNr))
```

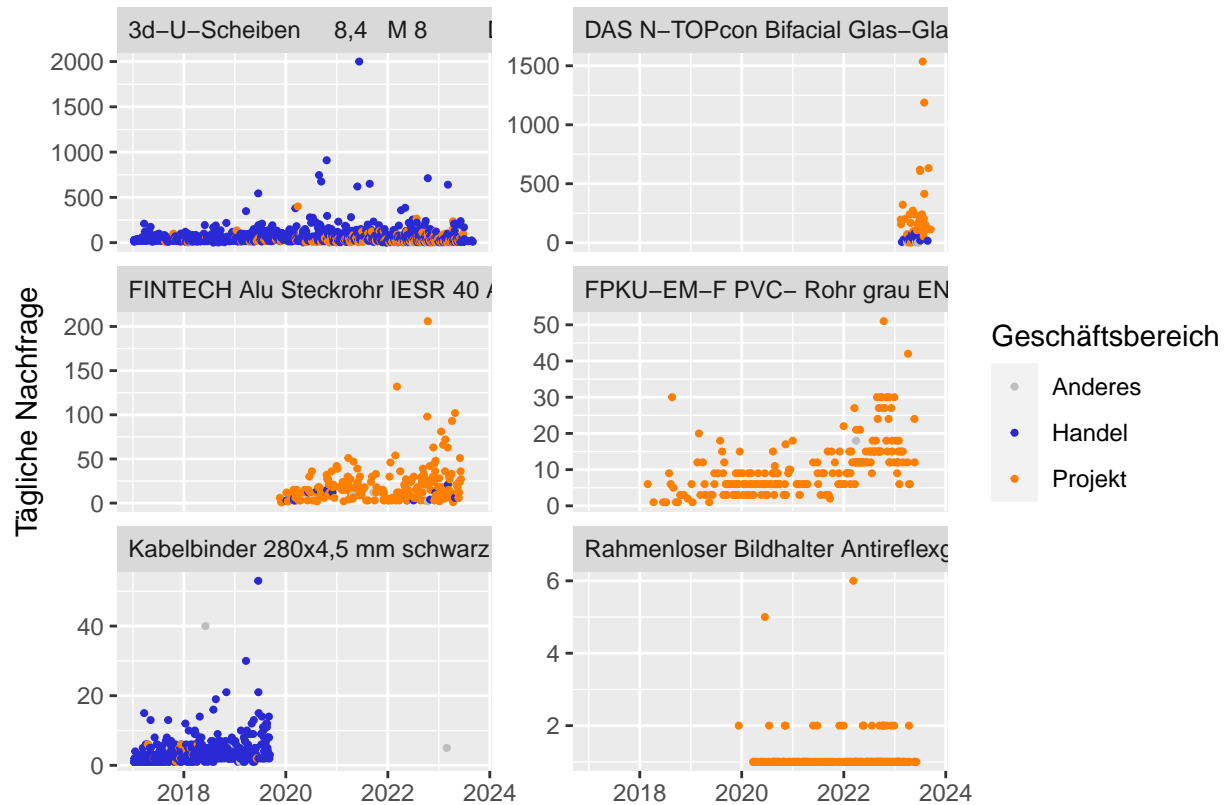


Nachfrage

Tägliche Nachfrage

```
# Nachfrage (Lagerausgänge = negative Werte) pro Tag
materials_b_tägl_menge <- materials_b %>%
  filter(Menge < 0) %>%
  group_by(EinAusDat, `Artikel-Bezeichnung`, klasse) %>%
  summarize(tägl_menge=-sum(Menge))

# Nachfrageverlauf über die Zeit
materials_b_tägl_menge %>% ggplot() +
  geom_point(aes(x=EinAusDat, y=tägl_menge, group=klasse, color=klasse), size=0.75)+
  facet_wrap(~`Artikel-Bezeichnung`, scales="free_y", nrow=3)+
  theme(strip.text.x = element_text(hjust=0.0))+
  scale_color_manual(values=c("Handel" = "#2A2AD5",
                              "Projekt" = "#FF8000",
                              "Einkauf" = "#00FFFF",
                              "Anderes" = "grey"))+
  labs(x="", y="Tägliche Nachfrage", color="Geschäftsbereich")
```



```
# 90% und 95% Perzentile
materials_b_tägl_menge %>%
  group_by(`Artikel-Bezeichnung`) %>%
  summarize(percentile_90 = quantile(tägl_menge, probs=0.9),
            percentile_95 = quantile(tägl_menge, probs=0.95))

## # A tibble: 6 x 3
##   `Artikel-Bezeichnung`      percentile_90 percentile_95
##   <fct>                  <dbl>          <dbl>
## 1 3d-U-Scheiben 8,4 M 8 DIN 9021 A2      120          162.
## 2 DAS N-TOPcon Bifacial Glas-Glas (2+2mm) Black 415~ 336          619
## 3 FINTECH Alu Steckrohr IESR 40 AL (3m)      36           51
## 4 FPKU-EM-F PVC- Rohr grau EN50 L=3m        18           27
## 5 Kabelbinder 280x4,5 mm schwarz              7           10
## 6 Rahmenloser Bildhalter Antireflexglas 29,7 x 42    1.80           2

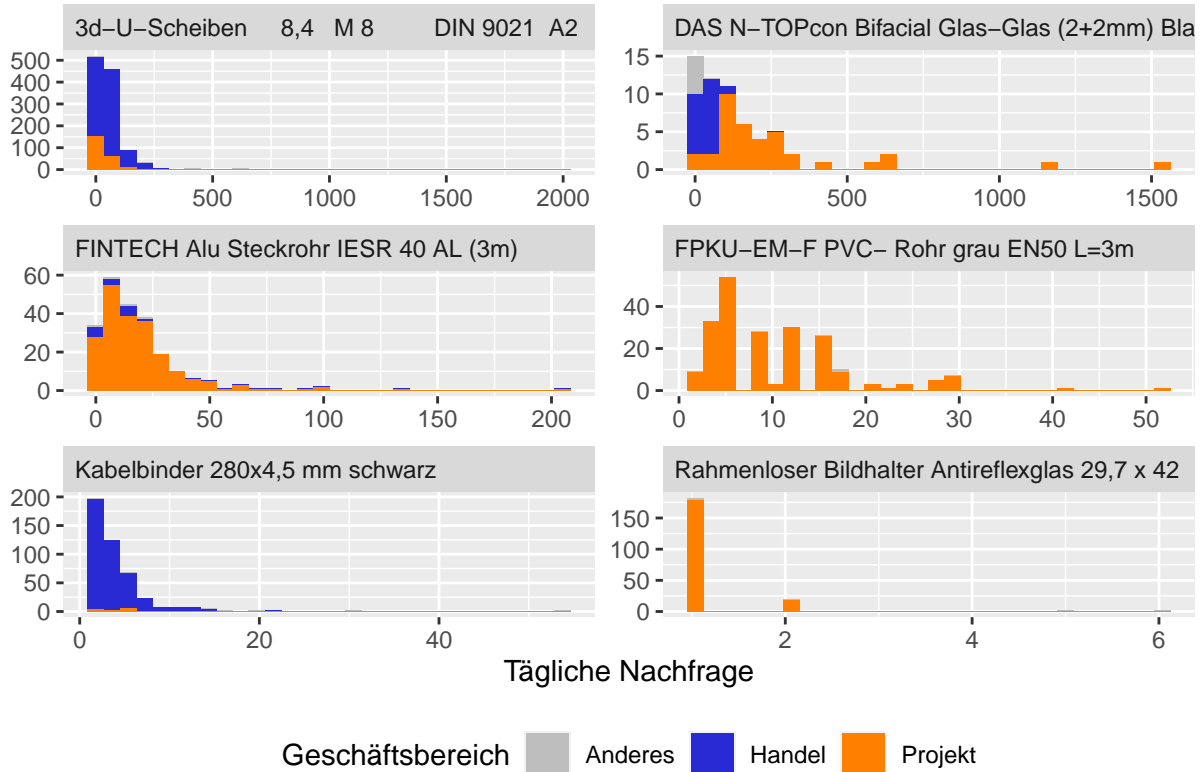
# Histogramm
ggplot(materials_b_tägl_menge, aes(x=tägl_menge, group=klasse, fill=klasse)) +
  geom_histogram() +
  facet_wrap(~`Artikel-Bezeichnung`, scales = 'free', nrow=3) +
  # Text linksbündig
  theme(strip.text.x = element_text(hjust=0.0)) +
  ggtitle("Verteilung der täglichen Ausgangsmengen (Nachfrage)") +
  scale_fill_manual(values=c("Handel" = "#2A2AD5",
                             "Projekt" = "#FF8000",
                             "Einkauf" = "#00FFFF",
```

```

    "Anderes" = "grey"))+
labs(y="", x="Tägliche Nachfrage", fill="Geschäftsbereich")+
theme(legend.position="bottom")

```

Verteilung der täglichen Ausgasmengen (Nachfrage)

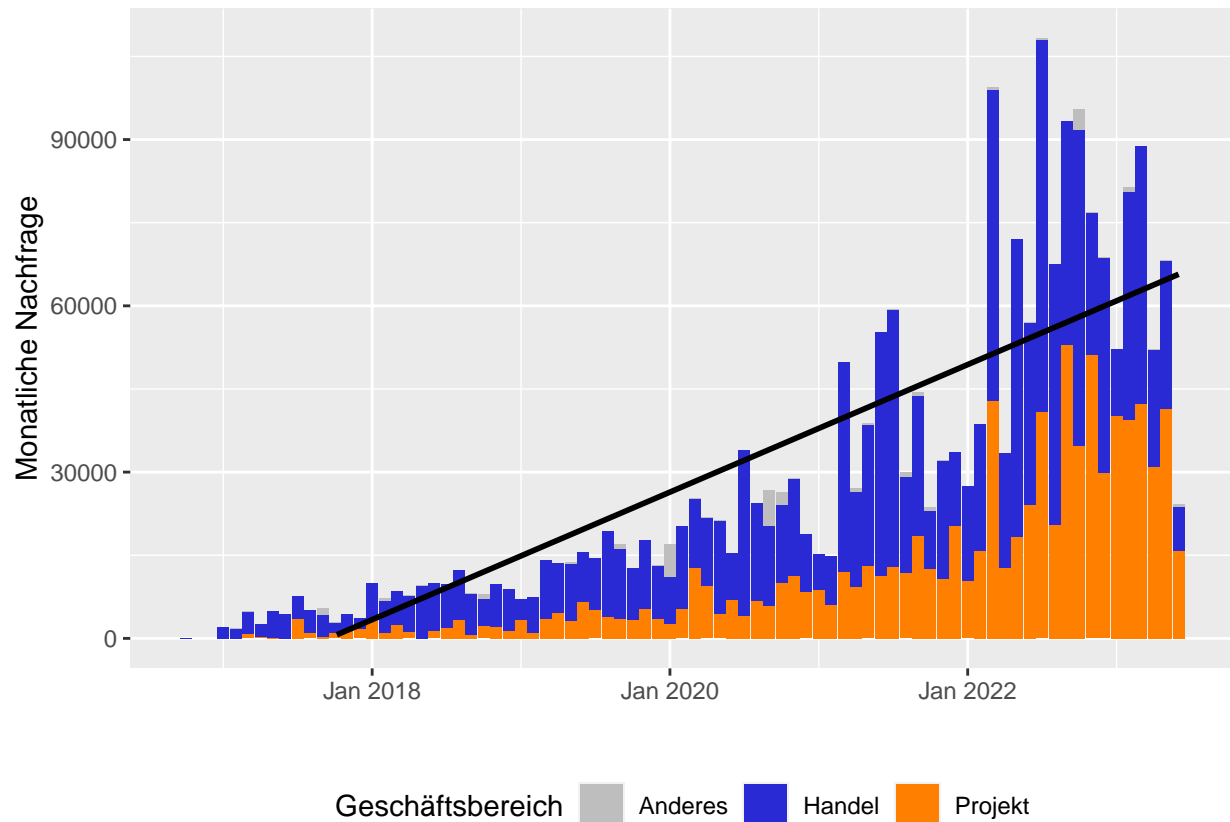


Plots: Monatliche Nachfrage

```

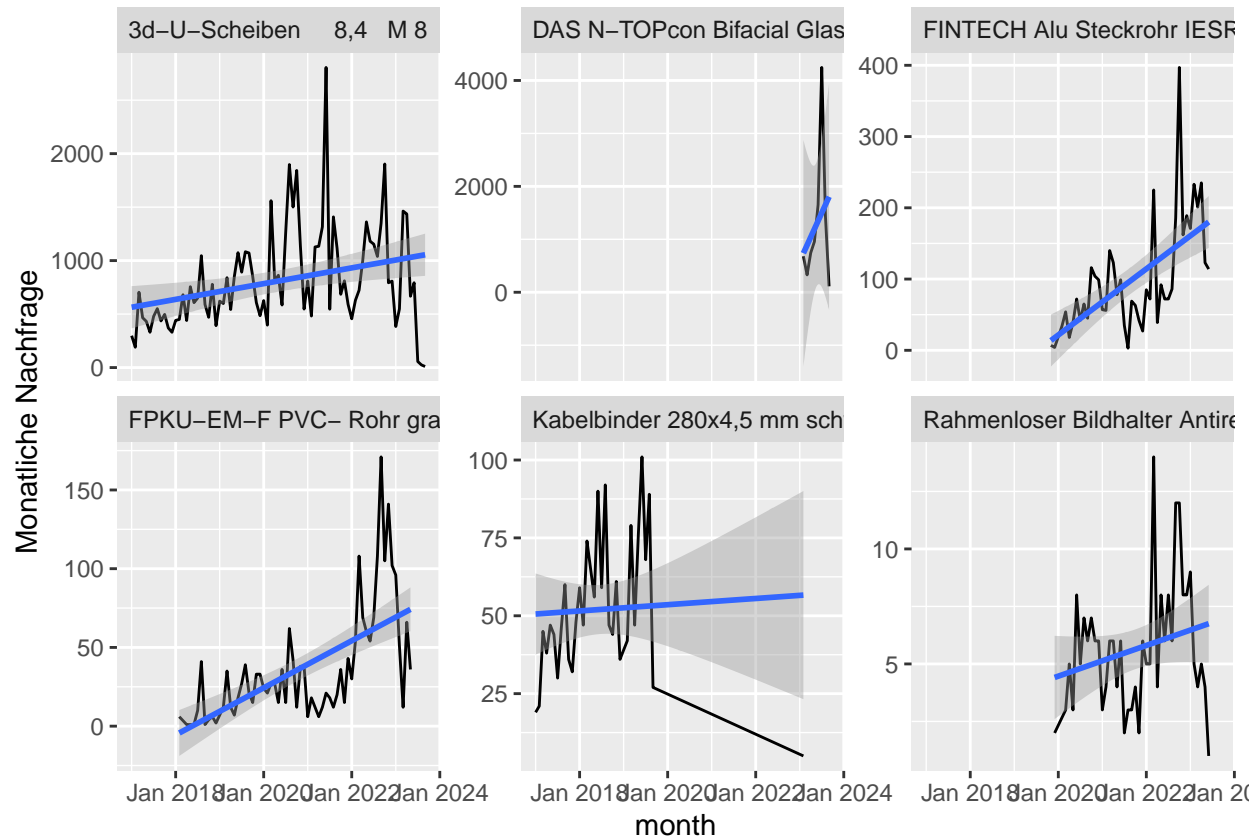
# Nachfrage nach Geschäftsbereich
materials %>% filter(Menge < 0, EinAusDat <= "2023-06-12") %>%
  ↳ group_by(mon=as.yearmon(EinAusDat), klasse) %>% summarize(s=sum(-Menge)) %>%
  ↳ ggplot(aes(x=mon, y=s))+
    geom_col(aes(group=klasse, fill=klasse))+
    scale_fill_manual(values=c("Handel" = "#2A2AD5",
                              "Projekt" = "#FF8000",
                              "Einkauf" = "#00FFFF",
                              "Anderes" = "grey"))+
labs(y="Monatliche Nachfrage", x="", fill="Geschäftsbereich")+
theme(legend.position="bottom")+
geom_smooth(data=materials %>% filter(Menge < 0, EinAusDat <= "2023-06-12") %>%
  ↳ group_by(mon=as.yearmon(EinAusDat)) %>% summarize(s=sum(-Menge)), aes(x=mon, y=s),
  ↳ method=lm, se=FALSE, color="black")+
ylim(0, NA)

```



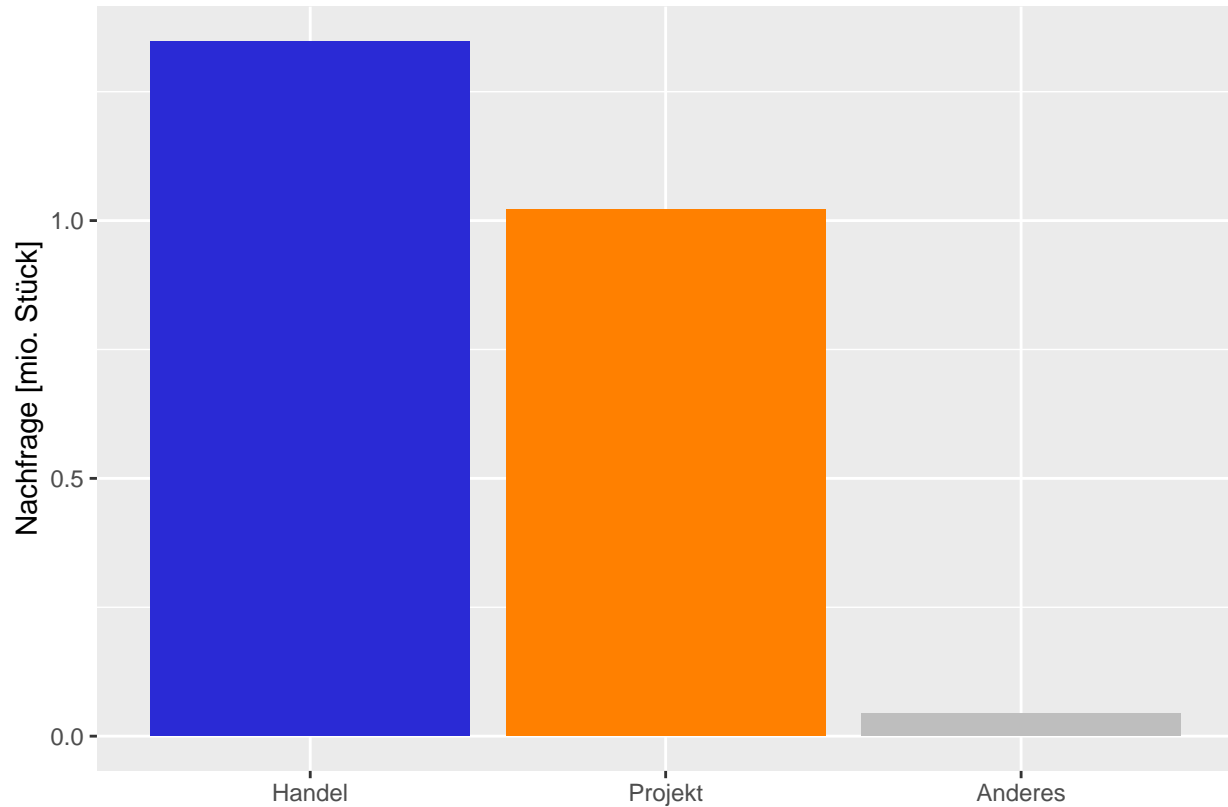
```
# Monatliche Nachfrage nach Artikel
materials_b_monatl_menge <- materials_b %>%
  filter(Menge < 0) %>%
  group_by(month=as.yearmon(EinAusDat), `Artikel-Bezeichnung`) %>%
  summarize(monatl_menge=-sum(Menge))

ggplot(data=materials_b_monatl_menge, aes(x=month, y=monatl_menge)) +
  geom_line() +
  facet_wrap(~`Artikel-Bezeichnung`, scales = 'free_y') +
  # Text linksbündig
  theme(strip.text.x = element_text(hjust=0.0)) +
  # Rotate Axis Ticks
  #theme(axis.text.x = element_text(angle = 90)) +
  ylab("Monatliche Nachfrage") +
  geom_smooth(method=lm)
```

Plot: Nachfrage nach Geschäftsbereich

```
materials %>% filter(Menge < 0) %>% group_by(klasse) %>%
  summarize(Nachfrage = sum(-Menge) / 10 * 6) %>%
  ggplot(aes(x = reorder(klasse, -Nachfrage), y = Nachfrage, fill = klasse)) +
  geom_col() +
  scale_fill_manual(values = c("Handel" = "#2A2AD5",
                              "Projekt" = "#FF8000",
                              "Einkauf" = "#00FFFF",
                              "Anderes" = "grey")) +
  labs(x = "", y = "Nachfrage [mio. Stück]") +
  theme(legend.position = "none")
```



Wiederbeschaffungszeit (WBZ)

```
WBZ_Daten <- materials_b %>% filter(EinAusDatTyp == "geliefertAm")
WBZ_Daten$WBZ <- WBZ_Daten$EinAusDat - WBZ_Daten$BstDat

# Nur positive WBZ (negative basieren auf fehlerhaften Transaktionen)
WBZ_Daten <- WBZ_Daten %>% filter(WBZ > 0)
WBZ_Daten[,c("ArtikelNr", "Artikel-Bezeichnung", "BstDat", "EinAusDat", "WBZ")]
```

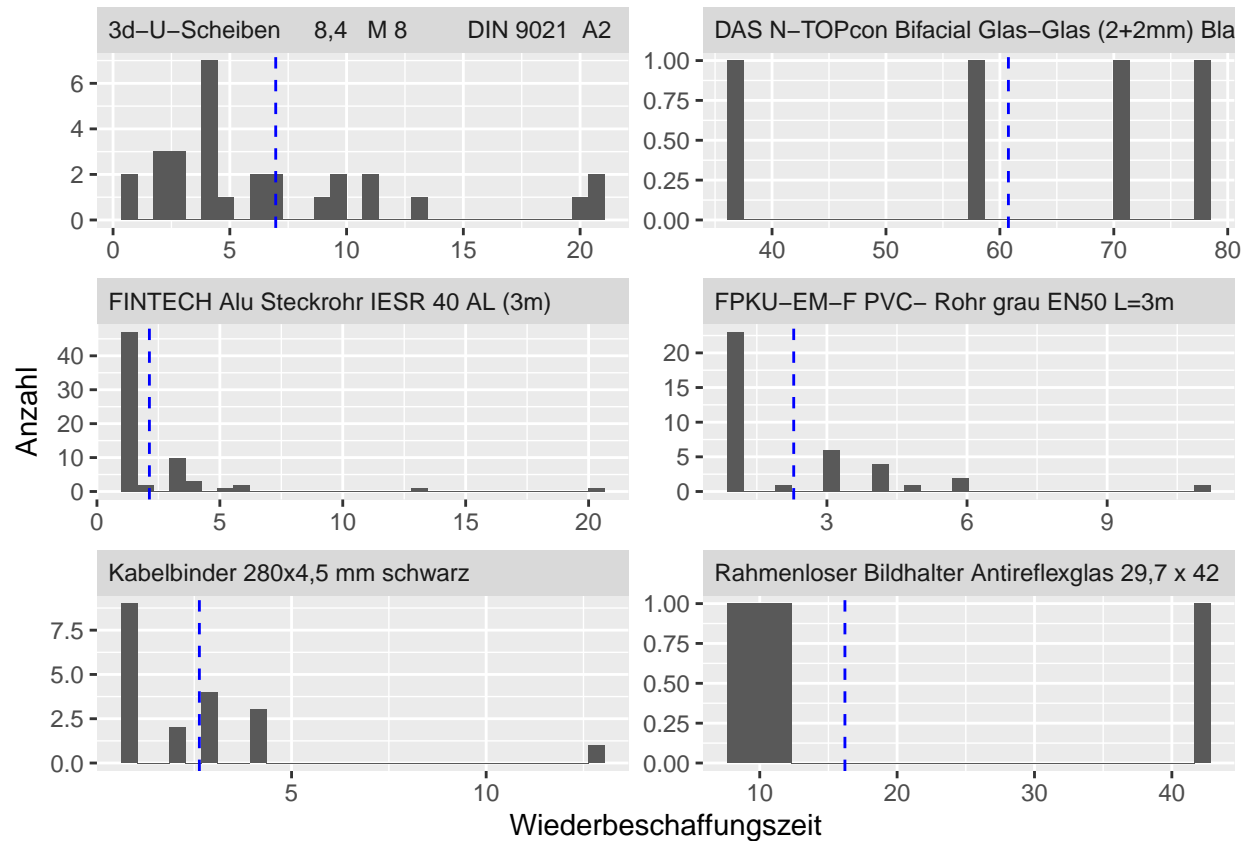
```
## # A tibble: 162 x 5
##   ArtikelNr `Artikel-Bezeichnung` BstDat EinAusDat WBZ
##   <fct>    <fct>                  <date> <date>    <drt>
## 1 26535    Rahmenloser Bildhalter Antireflexglas ~ 2020-05-12 2020-05-20 8 d~
## 2 26535    Rahmenloser Bildhalter Antireflexglas ~ 2020-09-10 2020-09-22 12 d~
## 3 26535    Rahmenloser Bildhalter Antireflexglas ~ 2022-06-21 2022-06-30 9 d~
## 4 26535    Rahmenloser Bildhalter Antireflexglas ~ 2023-01-06 2023-01-16 10 d~
## 5 26535    Rahmenloser Bildhalter Antireflexglas ~ 2023-02-02 2023-03-16 42 d~
## 6 21866    Kabelbinder 280x4,5 mm schwarz 2017-03-16 2017-03-17 1 d~
## 7 21866    Kabelbinder 280x4,5 mm schwarz 2017-03-16 2017-03-20 4 d~
## 8 21866    Kabelbinder 280x4,5 mm schwarz 2017-05-10 2017-05-11 1 d~
## 9 21866    Kabelbinder 280x4,5 mm schwarz 2017-08-15 2017-08-16 1 d~
## 10 21866    Kabelbinder 280x4,5 mm schwarz 2017-12-11 2017-12-13 2 d~
## # i 152 more rows
```

```
# Histogramm
WBZ_Daten %>%
```

```

ggplot(aes(x=WBZ)) +
  geom_histogram() +
  facet_wrap(~`Artikel-Bezeichnung`, scales = "free", nrow=3) +
  labs(x = "Wiederbeschaffungszeit", y = "Anzahl") +
  geom_vline(data=WBZ_Daten %>% group_by(`Artikel-Bezeichnung`) %>%
    ↪ summarize(hist_mean=mean(WBZ)), aes(xintercept=hist_mean), color="blue",
    ↪ linetype="dashed", size=0.5)+
  # Text linksbündig
  theme(strip.text.x = element_text(hjust=0.0))

```



```

# Verteilungsmasse
WBZ_Daten %>%
  group_by(ArtikelNr, `Artikel-Bezeichnung`) %>%
  summarize(mean = mean(WBZ),
            median = median(WBZ),
            sd = sd(WBZ))

```

```

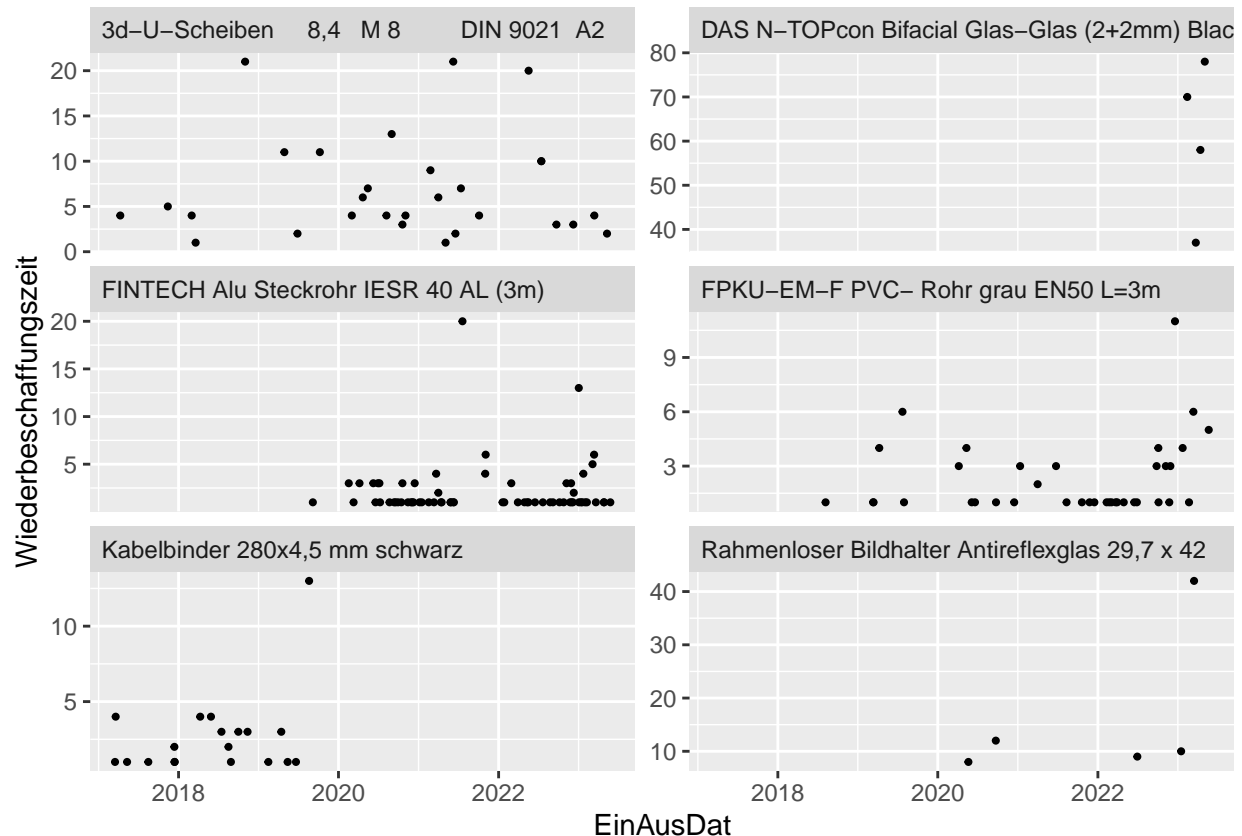
## # A tibble: 6 x 5
## # Groups:   ArtikelNr [6]
##   ArtikelNr `Artikel-Bezeichnung` mean median sd
##   <fct>     <fct>                <dbl> <dbl> <dbl>
## 1 2628      FPKU-EM-F PVC- Rohr grau EN50 L=3m 2.2~ 1 da~ 2.10
## 2 6203      3d-U-Scheiben 8,4 M 8 DIN 9021 A2 6.9~ 4 da~ 5.72
## 3 21866      Kabelbinder 280x4,5 mm schwarz 2.6~ 2 da~ 2.77
## 4 26385      FINTECH Alu Steckrohr IESR 40 AL (3m) 2.1~ 1 da~ 2.90
## 5 26535      Rahmenloser Bildhalter Antireflexglas 29,7 x 42 16.2~ 10 da~ 14.5

```

```
## 6 29523      DAS N-TOPcon Bifacial Glas-Glas (2+2mm) Black 41~ 60.7~ 64 da~ 17.8
```

```
# Im Zeitverlauf - täglich
```

```
WBZ_Daten %>%
  ggplot(aes(x=EinAusDat, y=WBZ)) +
  geom_point(size=0.75) +
  facet_wrap(~`Artikel-Bezeichnung`, scales = "free_y", nrow=3) +
  # Text linksbündig
  theme(strip.text.x = element_text(hjust=0.0)) +
  labs(y="Wiederbeschaffungszeit")
```



Phase 4: Analyse & Modellierung

Nachfrage und WBZ vorbereiten

```
# Zukunftswerte filtern
materials_b <- materials_b %>% filter(EinAusDat <= "2023-06-12")

# Tägliche Nachfrage
tägl_nachfrage <- materials_b %>%
  filter(Menge < 0) %>% # nur Nachfrage-Transaktionen
  group_by(EinAusDat, `Artikel-Bezeichnung`) %>% # Täglich und nach Artikel
  summarize(tägl_menge=-sum(Menge)) %>%
  group_by(`Artikel-Bezeichnung`) %>% # für Artikel über gesamte Zeit aggregieren
  summarize(Konservativ=quantile(tägl_menge, 0.9),
```

```

    Ausgewogen=quantile(tägl_menge, 0.5),
    Risikofreudig=quantile(tägl_menge, 0.5))
tägl_nachfrage

## # A tibble: 6 x 4
##   `Artikel-Bezeichnung`      Konservativ Ausgewogen Risikofreudig
##   <fct>                  <dbl>      <dbl>      <dbl>
## 1 3d-U-Scheiben      8,4    M 8      DIN ~    133.        40        40
## 2 DAS N-TOPcon Bifacial Glas-Glas (2+2mm) ~    265        66        66
## 3 FINTECH Alu Steckrohr IESR 40 AL (3m)      38.4        12        12
## 4 FPKU-EM-F PVC- Rohr grau EN50 L=3m         18          9          9
## 5 Kabelbinder 280x4,5 mm schwarz              8          3          3
## 6 Rahmenloser Bildhalter Antireflexglas 29~    2          1          1

```

```

# Wiederbeschaffungszeit
wbz <- materials_b %>% filter(EinAutDatTyp == "geliefertAm") %>%
  mutate(WBZ=EinAusDat - BstDat) %>%
  filter(WBZ > 0) %>% # Nur positive WBZ (negative basieren auf fehlerhaften
    ↳ Transaktionen)
  group_by(`Artikel-Bezeichnung`) %>%
  summarize(Konservativ=quantile(WBZ, 0.9),
    Ausgewogen=quantile(WBZ, 0.5),
    Risikofreudig=quantile(WBZ, 0.0))
wbz

```

```

## # A tibble: 6 x 4
##   `Artikel-Bezeichnung`      Konservativ Ausgewogen Risikofreudig
##   <fct>                  <drtn>      <drtn>      <drtn>
## 1 3d-U-Scheiben      8,4    M 8      DIN ~ 14.4 days    4 days    1 days
## 2 DAS N-TOPcon Bifacial Glas-Glas (2+2mm) ~ 75.6 days   64 days   37 days
## 3 FINTECH Alu Steckrohr IESR 40 AL (3m)      4.0 days    1 days    1 days
## 4 FPKU-EM-F PVC- Rohr grau EN50 L=3m         4.3 days    1 days    1 days
## 5 Kabelbinder 280x4,5 mm schwarz              4.0 days    2 days    1 days
## 6 Rahmenloser Bildhalter Antireflexglas 29~ 30.0 days   10 days    8 days

```

Bestandverlauf mit neuer Bestellpolitik

```

alle_artikel <- data.frame()

# Loop über alle Bestellpolitiken
for (politik in c("Konservativ", "Ausgewogen", "Risikofreudig")) {
  cat("\n\n\n-----\nVerfolgte Bestellpolitik:", politik)

  # Loop über alle Artikel/Materialien
  for (mat in materials_group_b) {

    # Start-Parameter
    artikel <- mat
    artikelNr <- first(materials_b[materials_b$`Artikel-Bezeichnung` == artikel,
    ↳ "ArtikelNr"])

    # Quantile heraussuchen
    spalte <- ifelse(politik=="Konservativ", "Konservativ", ifelse(politik=="Ausgewogen",
    ↳ "Ausgewogen", "Risikofreudig"))
  }
}

```

```

wbz_quantil <- as.numeric(wbz[wbz$`Artikel-Bezeichnung`==mat, spalte])
nachfrage_quantil <-
↪ as.numeric(tägl_nachfrage[tägl_nachfrage$`Artikel-Bezeichnung`==mat, spalte])

# Losgröße berechnen nach Formel
losgröße <- nachfrage_quantil * wbz_quantil
startbestand <- losgröße

# Reale Nachfragewerte vorbereiten zum Iterieren
nachfrage <- materials_b %>%
  filter(Menge < 0, `Artikel-Bezeichnung`==artikel, !is.na(EinAusDat)) %>%
  mutate(cum_sum=cumsum(Menge))

# Variablen für Loop
bestand <- startbestand
bestelldaten <- c()
bestellmengen <- c()

# Bestelldaten berechnen
for (i in 1:nrow(nachfrage)) {
  row <- nth(nachfrage, i)

  bestand <- bestand + as.numeric(row$Menge)

  if (bestand <= losgröße) {
    # Bestellung auslösen
    bestelldaten <- append(bestelldaten, row$EinAusDat)

    # Bestellmenge = Losgröße plus was zusätzlich fehlt
    bestellmenge <- losgröße + (losgröße-bestand)
    bestellmengen <- append(bestellmengen, bestellmenge)
    bestand <- bestand + bestellmenge
  }
}

# Erste Zeile - Anfangswert
anfang <- data.frame(ArtikelNr=artikelNr,
  `Artikel-Bezeichnung`=artikel,
  EinAusDat=min(nachfrage$EinAusDat, na.rm=TRUE),
  EinAusDatTyp="geliefertAm",
  BstDat=min(nachfrage$EinAusDat, na.rm=TRUE) - wbz_quantil,
  Menge=startbestand,
  Bestand=NA,
  PjNr=NA,
  PjInfo=NA,
  ORGAKzl="Anfangsbestand",
  klasse="Einkauf",
  cum_sum=NA)

# DataFrame mit allen Bestellungen
mengen <- data.frame(ArtikelNr=artikelNr,
  `Artikel-Bezeichnung`=artikel,
  EinAusDat=bestelldaten + wbz_quantil,

```

```

        EinAusDatTyp="geliefertAm",
        BstDat=bestelldaten,
        Menge=bestellmengen,
        Bestand=NA,
        PjNr=NA,
        PjInfo=NA,
        ORGAKz1="Neuer Einkauf",
        klasse="Einkauf",
        cum_sum=NA)

colnames(anfang) <- colnames(nachfrage)
colnames(mengen) <- colnames(nachfrage)

# DataFrame zusammenführen, cum_sum (= Bestand) neu berechnen
ges <- rbind(anfang, mengen, nachfrage) %>%
  arrange(EinAusDat) %>%
  mutate(cum_sum = cumsum(Menge), Bestellpolitik=politik)
# DataFrame mit Gesamt-DF zusammenführen
alle_artikel <- rbind(alle_artikel, ges)

# Statistiken
cat("\n\n", artikel)
cat("\n-----")
cat("\nLosgröße (= Meldebestand) = ", losgröße)
cat("\nAngenommene Wiederbeschaffungszeit = ", wbz_quantil)
cat("\nAnzahl Bestellungen = ", nrow(mengen))
cat("\nBestellungen MBG = ", nrow(materials_b %>% filter(Menge > 0,
  ↳ `Artikel-Bezeichnung`==artikel)))
cat("\nAnzahl Stock-outs = ", nrow(ges %>% filter(cum_sum < 0 & lag(cum_sum) >=0)))
cat("\nAnzahl Stock-outs MBG = ", nrow(materials_b %>%
  ↳ filter(`Artikel-Bezeichnung`==artikel, Bestand < 0 & lag(Bestand) >=0)))

# Individuelle Ergebnisse plotten
#ggplot(data=vergleich) + geom_line(aes(x=EinAusDat,y=cum_sum, group=Bestellpolitik,
  ↳ color=Bestellpolitik))+ggtitle(artikel)
}
}

```

```

##
##
##
## -----
## Verfolgte Bestellpolitik: Konservativ
##
## DAS N-TOPcon Bifacial Glas-Glas (2+2mm) Black 415 WP
## -----
## Losgröße (= Meldebestand) = 20034
## Angenommene Wiederbeschaffungszeit = 75.6
## Anzahl Bestellungen = 1
## Bestellungen MBG = 5
## Anzahl Stock-outs = 0
## Anzahl Stock-outs MBG = 0

```

```

##
## Rahmenloser Bildhalter Antireflexglas 29,7 x 42
## -----
## Losgröße (= Meldebestand) = 60
## Angenommene Wiederbeschaffungszeit = 30
## Anzahl Bestellungen = 4
## Bestellungen MBG = 8
## Anzahl Stock-outs = 0
## Anzahl Stock-outs MBG = 3
##
## Kabelbinder 280x4,5 mm schwarz
## -----
## Losgröße (= Meldebestand) = 32
## Angenommene Wiederbeschaffungszeit = 4
## Anzahl Bestellungen = 54
## Bestellungen MBG = 20
## Anzahl Stock-outs = 1
## Anzahl Stock-outs MBG = 6
##
## FPKU-EM-F PVC- Rohr grau EN50 L=3m
## -----
## Losgröße (= Meldebestand) = 77.4
## Angenommene Wiederbeschaffungszeit = 4.3
## Anzahl Bestellungen = 27
## Bestellungen MBG = 39
## Anzahl Stock-outs = 0
## Anzahl Stock-outs MBG = 0
##
## FINTECH Alu Steckrohr IESR 40 AL (3m)
## -----
## Losgröße (= Meldebestand) = 153.6
## Angenommene Wiederbeschaffungszeit = 4
## Anzahl Bestellungen = 26
## Bestellungen MBG = 70
## Anzahl Stock-outs = 0
## Anzahl Stock-outs MBG = 0
##
## 3d-U-Scheiben      8,4    M 8          DIN 9021  A2
## -----
## Losgröße (= Meldebestand) = 1909.44
## Angenommene Wiederbeschaffungszeit = 14.4
## Anzahl Bestellungen = 33
## Bestellungen MBG = 31
## Anzahl Stock-outs = 1
## Anzahl Stock-outs MBG = 6
##
##
## -----
## Verfolgte Bestellpolitik: Ausgewogen
##
## DAS N-TOPcon Bifacial Glas-Glas (2+2mm) Black 415 WP
## -----
## Losgröße (= Meldebestand) = 4224
## Angenommene Wiederbeschaffungszeit = 64

```



```

## Anzahl Bestellungen = 1
## Bestellungen MBG = 5
## Anzahl Stock-outs = 0
## Anzahl Stock-outs MBG = 0
##
## Rahmenloser Bildhalter Antireflexglas 29,7 x 42
## -----
## Losgröße (= Meldebestand) = 10
## Angenommene Wiederbeschaffungszeit = 10
## Anzahl Bestellungen = 23
## Bestellungen MBG = 8
## Anzahl Stock-outs = 0
## Anzahl Stock-outs MBG = 3
##
## Kabelbinder 280x4,5 mm schwarz
## -----
## Losgröße (= Meldebestand) = 6
## Angenommene Wiederbeschaffungszeit = 2
## Anzahl Bestellungen = 239
## Bestellungen MBG = 20
## Anzahl Stock-outs = 44
## Anzahl Stock-outs MBG = 6
##
## FPKU-EM-F PVC- Rohr grau EN50 L=3m
## -----
## Losgröße (= Meldebestand) = 9
## Angenommene Wiederbeschaffungszeit = 1
## Anzahl Bestellungen = 170
## Bestellungen MBG = 39
## Anzahl Stock-outs = 24
## Anzahl Stock-outs MBG = 0
##
## FINTECH Alu Steckrohr IESR 40 AL (3m)
## -----
## Losgröße (= Meldebestand) = 12
## Angenommene Wiederbeschaffungszeit = 1
## Anzahl Bestellungen = 176
## Bestellungen MBG = 70
## Anzahl Stock-outs = 66
## Anzahl Stock-outs MBG = 0
##
## 3d-U-Scheiben      8,4    M 8          DIN 9021  A2
## -----
## Losgröße (= Meldebestand) = 160
## Angenommene Wiederbeschaffungszeit = 4
## Anzahl Bestellungen = 314
## Bestellungen MBG = 31
## Anzahl Stock-outs = 84
## Anzahl Stock-outs MBG = 6
##
##
## -----
## Verfolgte Bestellpolitik: Risikofreudig
##

```

```

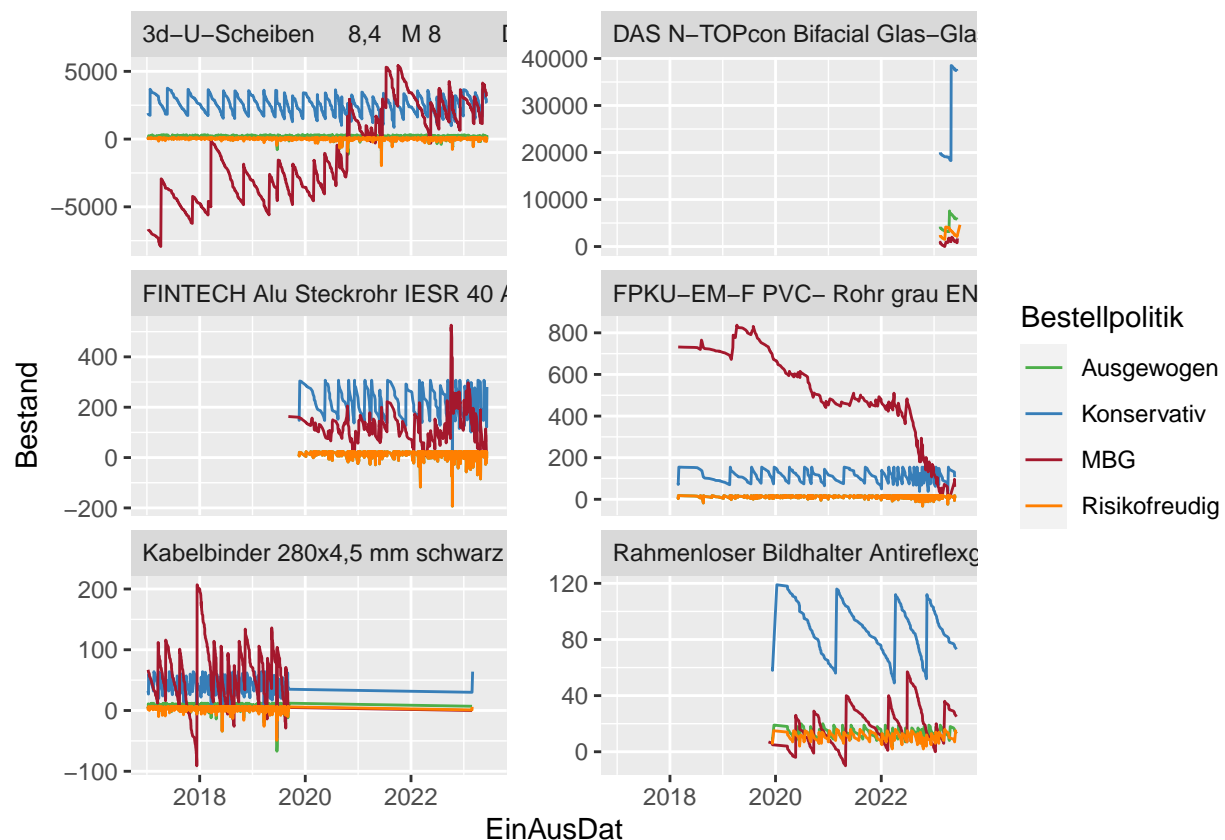
## DAS N-TOPcon Bifacial Glas-Glas (2+2mm) Black 415 WP
## -----
## Losgröße (= Meldebestand) = 2442
## Angenommene Wiederbeschaffungszeit = 37
## Anzahl Bestellungen = 2
## Bestellungen MBG = 5
## Anzahl Stock-outs = 0
## Anzahl Stock-outs MBG = 0
##
## Rahmenloser Bildhalter Antireflexglas 29,7 x 42
## -----
## Losgröße (= Meldebestand) = 8
## Angenommene Wiederbeschaffungszeit = 8
## Anzahl Bestellungen = 29
## Bestellungen MBG = 8
## Anzahl Stock-outs = 0
## Anzahl Stock-outs MBG = 3
##
## Kabelbinder 280x4,5 mm schwarz
## -----
## Losgröße (= Meldebestand) = 3
## Angenommene Wiederbeschaffungszeit = 1
## Anzahl Bestellungen = 427
## Bestellungen MBG = 20
## Anzahl Stock-outs = 86
## Anzahl Stock-outs MBG = 6
##
## FPKU-EM-F PVC- Rohr grau EN50 L=3m
## -----
## Losgröße (= Meldebestand) = 9
## Angenommene Wiederbeschaffungszeit = 1
## Anzahl Bestellungen = 170
## Bestellungen MBG = 39
## Anzahl Stock-outs = 24
## Anzahl Stock-outs MBG = 0
##
## FINTECH Alu Steckrohr IESR 40 AL (3m)
## -----
## Losgröße (= Meldebestand) = 12
## Angenommene Wiederbeschaffungszeit = 1
## Anzahl Bestellungen = 176
## Bestellungen MBG = 70
## Anzahl Stock-outs = 66
## Anzahl Stock-outs MBG = 0
##
## 3d-U-Scheiben      8,4    M 8          DIN 9021  A2
## -----
## Losgröße (= Meldebestand) = 40
## Angenommene Wiederbeschaffungszeit = 1
## Anzahl Bestellungen = 890
## Bestellungen MBG = 31
## Anzahl Stock-outs = 293
## Anzahl Stock-outs MBG = 6

```

```
# DataFrame mit MBG Performance zusammenführen
alle_artikel <- rbind(alle_artikel, materials_b %>% mutate(cum_sum=Bestand,
  ↳ Bestellpolitik="MBG") # Bestand statt cumsum der Menge, um richtige Anfangswerte zu
  ↳ haben
                        #mutate(cum_sum=cumsum(Menge), Bestellpolitik="MBG")
)
```

Plots der Bestandspolitiken

```
# Gesamtplot über alle Materialien
ggplot(data=alle_artikel) +
  geom_line(aes(x=EinAusDat, y=cum_sum, group=Bestellpolitik, color=Bestellpolitik))+
  facet_wrap(~`Artikel-Bezeichnung`, nrow=3, scale="free_y")+
  ylab("Bestand")+
  scale_color_manual(values=c("MBG"="#A4182D", "Konservativ"="#377eb8",
  ↳ "Ausgewogen"="#4daf4a", "Risikofreudig"="#ff7f00"))+
  # Text linksbündig
  theme(strip.text.x = element_text(hjust=0.0))
```



```
# Einzelplots
mat <- "Rahmenloser Bildhalter Antireflexglas 29,7 x 42"

ggplot(data=alle_artikel %>% filter(`Artikel-Bezeichnung`==mat)) +
  geom_line(aes(x=EinAusDat, y=cum_sum, group=Bestellpolitik, color=Bestellpolitik),
  ↳ show.legend=FALSE)+
```

```

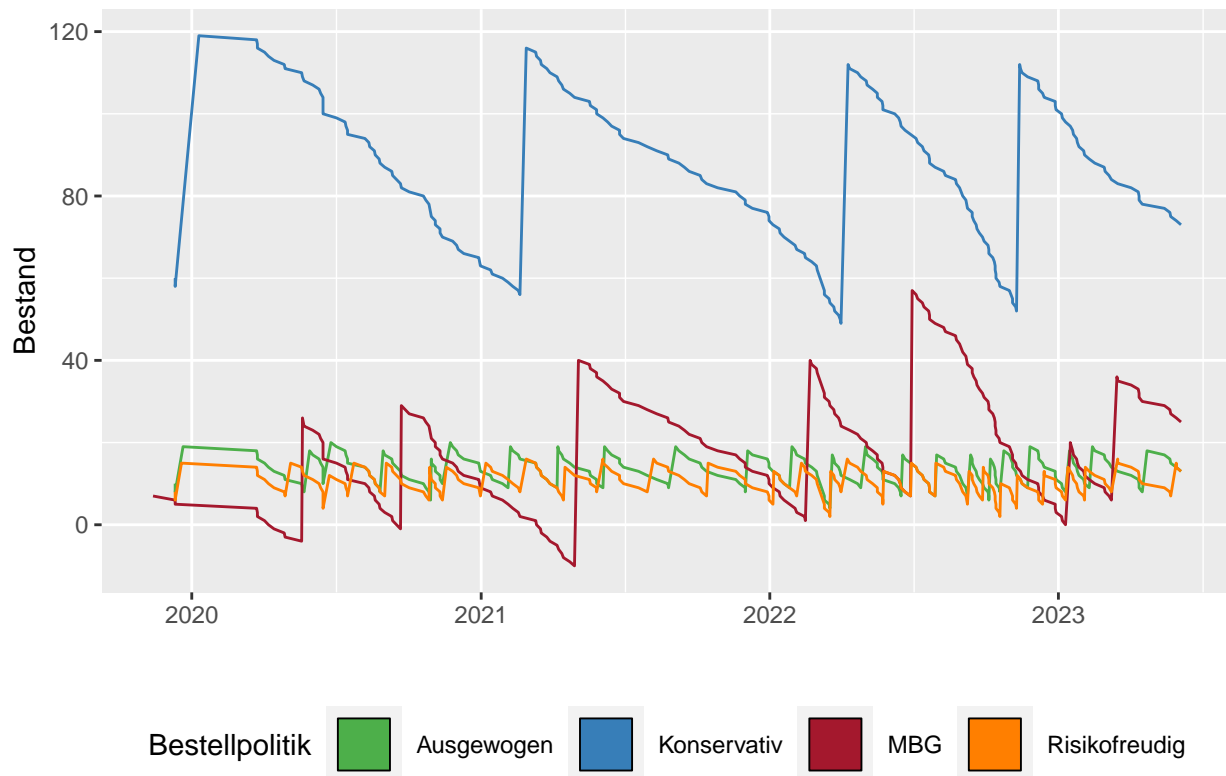
labs(y="Bestand", x="")+
ggtitle(mat)+

# Zusätze für schönere Legende
geom_point(aes(x=EinAusDat, y=cum_sum, fill=Bestellpolitik), size=-1)+
theme(legend.position = "bottom")+
guides(fill = guide_legend(keywidth = unit(1, "cm"),
                           keyheight = unit(1, "cm"),
                           override.aes = list(shape = 22,
                                                size = 10)))+

scale_color_manual(values=c("MBG"="#A4182D", "Konservativ"="#377eb8",
  ↳ "Ausgewogen"="#4daf4a", "Risikofreudig"="#ff7f00"))+
scale_fill_manual(values=c("MBG"="#A4182D", "Konservativ"="#377eb8",
  ↳ "Ausgewogen"="#4daf4a", "Risikofreudig"="#ff7f00"))

```

Rahmenloser Bildhalter Antireflexglas 29,7 x 42



Phase 5: Ergebnisbewertung

Anzahl Bestellungen

```

# Anmerkung: Von den Bestellungen der neuen Bestellpolitiken muss 1 abgezogen werden
↳ (Startwert)
alle_artikel %>% filter(Menge > 0) %>%
  group_by(`Artikel-Bezeichnung`, Bestellpolitik) %>% count() %>%
  pivot_wider(names_from=Bestellpolitik, values_from=n)

```

```
## # A tibble: 6 x 5
## # Groups:   Artikel-Bezeichnung [6]
##   `Artikel-Bezeichnung`      Ausgewogen Konservativ MBG Risikofreudig
##   <chr>                  <int>      <int> <int>      <int>
## 1 3d-U-Scheiben      8,4    M 8      ~      315        34    31      891
## 2 DAS N-TOPcon Bifacial Glas-Glas (2~      2         2     5        3
## 3 FINTECH Alu Steckrohr IESR 40 AL (~      177       27   70      177
## 4 FPKU-EM-F PVC- Rohr grau EN50 L=3m      171       28   39      171
## 5 Kabelbinder 280x4,5 mm schwarz          240      55   20      428
## 6 Rahmenloser Bildhalter Antireflexg~      24        5    8       30
```

Durchschnittlicher Lagerbestand

```
alle_artikel %>% group_by(`Artikel-Bezeichnung`, Bestellpolitik) %>%
  ↳ summarize(mean_bestand=mean(cum_sum)) %>% pivot_wider(names_from=Bestellpolitik,
  ↳ values_from=mean_bestand)
```

```
## # A tibble: 6 x 5
## # Groups:   Artikel-Bezeichnung [6]
##   `Artikel-Bezeichnung`      Ausgewogen Konservativ MBG Risikofreudig
##   <chr>                  <dbl>      <dbl> <dbl>      <dbl>
## 1 3d-U-Scheiben      8,4    M 8      ~      119.      2484. -583.      17.7
## 2 DAS N-TOPcon Bifacial Glas-Glas (~      5484.     28647. 1159.     2911.
## 3 FINTECH Alu Steckrohr IESR 40 AL ~      6.50      216.   132.      6.50
## 4 FPKU-EM-F PVC- Rohr grau EN50 L=3m      10.1      110.   468.     10.1
## 5 Kabelbinder 280x4,5 mm schwarz          3.63      39.6   54.6      1.77
## 6 Rahmenloser Bildhalter Antireflex~      13.2      85.6   18.9     10.2
```

Stock-outs

```
alle_artikel %>% filter(cum_sum < 0 & lag(cum_sum) >=0) %>%
  group_by(`Artikel-Bezeichnung`, Bestellpolitik) %>% count() %>%
  pivot_wider(names_from=Bestellpolitik, values_from=n)
```

```
## # A tibble: 5 x 5
## # Groups:   Artikel-Bezeichnung [5]
##   `Artikel-Bezeichnung`      Ausgewogen Konservativ MBG Risikofreudig
##   <chr>                  <int>      <int> <int>      <int>
## 1 3d-U-Scheiben      8,4    M 8      ~      84         1     6      293
## 2 FINTECH Alu Steckrohr IESR 40 AL (~      66        NA    NA       66
## 3 FPKU-EM-F PVC- Rohr grau EN50 L=3m      24        NA    NA       24
## 4 Kabelbinder 280x4,5 mm schwarz          44         1     6       86
## 5 Rahmenloser Bildhalter Antireflexg~      NA        NA    3       NA
```

Business Impact: Kapitalbindung

```
kosten <- data.frame(`Artikel-Bezeichnung`=materials_group_b,
                     Stückpreis=c(170, 36.58, 2.85*280/100, 9.39, 11.77, 28.1/500))
colnames(kosten) <- c("Artikel-Bezeichnung", "Stückpreis")

alle_artikel %>% group_by(`Artikel-Bezeichnung`, Bestellpolitik) %>%
  ↳ summarize(gesamtbestand=sum(cum_sum)) %>% pivot_wider(names_from=Bestellpolitik,
  ↳ values_from=gesamtbestand) %>%
```

```
mutate(Unterschied=MBG - Risikofreudig) %>%
inner_join(kosten, by="Artikel-Bezeichnung") %>%
mutate(Einsparungen=Unterschied*Stückpreis)
```

```
## # A tibble: 6 x 8
## # Groups:   Artikel-Bezeichnung [6]
##   `Artikel-Bezeichnung` Ausgewogen Konservativ MBG Risikofreudig Unterschied
##   <chr>                <dbl>         <dbl>    <dbl>         <dbl>         <dbl>
## 1 3d-U-Scheiben      8,~      266348    4874032. -1.14e6      49870      -1191699
## 2 DAS N-TOPcon Bifacia~ 235801    1231827   5.33e4      128085       -74780
## 3 FINTECH Alu Steckroh~   2901      64054.   4.48e4       2901        41930
## 4 FPKU-EM-F PVC- Rohr ~   4227      30354.   1.34e5       4227       129737
## 5 Kabelbinder 280x4,5 ~   3538      31250    4.13e4       2061        39190
## 6 Rahmenloser Bildhalt~   3306      19851.   4.44e3       2616        1827
## # i 2 more variables: Stückpreis <dbl>, Einsparungen <dbl>
```

Anhang

Helper-Grafiken

```
# Nachher entfernen
perzentil <- -3
ggplot(data.frame(x=c(-3,3)), aes(x)) +
  stat_function(fun = dnorm,
                args = list(mean = 0, sd = 1),
                geom = "area",
                fill = "#C00000", n = 1001, xlim=c(-3, perzentil)) +
  stat_function(fun = dnorm,
                args = list(mean = 0, sd = 1),
                geom = "area",
                fill = "grey", n = 1001, xlim = c(perzentil, 3)) +
  labs(x="", y="") +
  scale_y_continuous(breaks = NULL)+
  scale_x_continuous(breaks=NULL)+geom_vline(aes(xintercept=-3), color="#C00000", size=2)
```



Zeitreihenanalysen

Zeitreihenanalyse Nachfrage

```
materials_b <- materials_b %>% group_by(ArtikelNr) %>% mutate(Baseline = mean(Menge))

evaluation = data.frame(Model = "Baseline",
                        MFE = numeric(1),
                        MAE = numeric(1),
                        MSE = numeric(1),
                        sMAPE = numeric(1))

# MFE berechnen
evaluation[evaluation$Model == "Baseline",] $MFE = mean(materials_b$Menge -
  ↪ materials_b$Baseline)

evaluation[evaluation$Model == "Baseline",] $MAE = mean(abs(materials_b$Menge -
  ↪ materials_b$Baseline))

evaluation[evaluation$Model == "Baseline",] $MSE = mean((materials_b$Menge -
  ↪ materials_b$Baseline)^2)

evaluation[evaluation$Model == "Baseline",] $sMAPE = smape(materials_b$Menge,
  ↪ materials_b$Baseline)
```

evaluation

Baseline-Modell (Durchschnitt)

##	Model	MFE	MAE	MSE	sMAPE
## 1	Baseline	7.869702e-17	48.08389	67060.18	1.914781

```
# Monatliche Durchschnittsnachfrage nehmen
materials_B_aus <- materials_b %>%
  filter(Menge < 0) %>%
  group_by(ArtikelNr, `Artikel-Bezeichnung`, month=as.yearmon(EinAusDat)) %>%
  summarize(menge_pro_monat= sum(-Menge))

# Gesamt-DataFrame
zeitreihen_alle_materialien <- data.frame()

for (mat in unique(materials_b$`Artikel-Bezeichnung`)) {
  cat("\n\n-----\nCurrent material: ", mat)
  mat_subset <- materials_B_aus %>% filter(`Artikel-Bezeichnung` == mat)
  ts <- ts(mat_subset$menge_pro_monat, frequency = 12)
  #plot.ts(ts_3d_Scheiben)

  # Create model
  model <- ets(ts, model="ZZZ")
  print(model)

  fcast <- forecast(model, 12)

  # DataFrame für Originaldaten
  df_orig = data.frame(
    period = seq(1, length(fcast$x), 1),
    bestand = as.numeric(fcast$x),
    group = rep("Original", length(fcast$x)))

  # DataFrame für Forecast erstellen (sowohl Vorhersage für Zeitreihenwerte als auch ein
  ↪ Jahr in die Zukunft)
  df_fcast = data.frame(
    period = seq(1, length(fcast$fitted)+length(fcast$mean), 1),
    bestand = c(as.numeric(fcast$fitted), as.numeric(fcast$mean)),
    group = rep("Modell", length(fcast$fitted)+length(fcast$mean)))

  # In ein DataFrame zusammenführen
  df = rbind(df_orig, df_fcast)

  # In Gesamt-DataFrame hinzufügen
  zeitreihen_alle_materialien <- rbind(zeitreihen_alle_materialien, df %>%
  ↪ mutate(Artikel=mat))

  # Plot (print() hinzufügen, wenn jeder Plot einzeln ausgegeben werden soll)
  ggplot(df, aes(x = period, y = bestand, colour = group)) +
    geom_line()+
    xlab("Periode") +
```



```

ylab("Nachfrage") +
ggtitle("Gesamtmodell (Projekt + Großhandel) für Artikel:", mat) +
theme(legend.title = element_blank()
)
}

```

Gesamtmodell

```

##
##
## -----
## Current material:  Rahmenloser Bildhalter Antireflexglas 29,7 x 42ETS(A,N,N)
##
## Call:
## ets(y = ts, model = "ZZZ")
##
## Smoothing parameters:
##   alpha = 0.4135
##
## Initial states:
##   l = 3.3117
##
## sigma:  2.66
##
##      AIC      AICc      BIC
## 236.4295 237.0781 241.5702
##
## -----
## Current material:  Kabelbinder 280x4,5 mm schwarzETS(M,N,N)
##
## Call:
## ets(y = ts, model = "ZZZ")
##
## Smoothing parameters:
##   alpha = 0.2188
##
## Initial states:
##   l = 34.5943
##
## sigma:  0.4171
##
##      AIC      AICc      BIC
## 328.1770 328.9770 332.7561
##
## -----
## Current material:  FPKU-EM-F PVC- Rohr grau EN50 L=3mETS(M,A,N)
##
## Call:
## ets(y = ts, model = "ZZZ")
##
## Smoothing parameters:
##   alpha = 0.6227

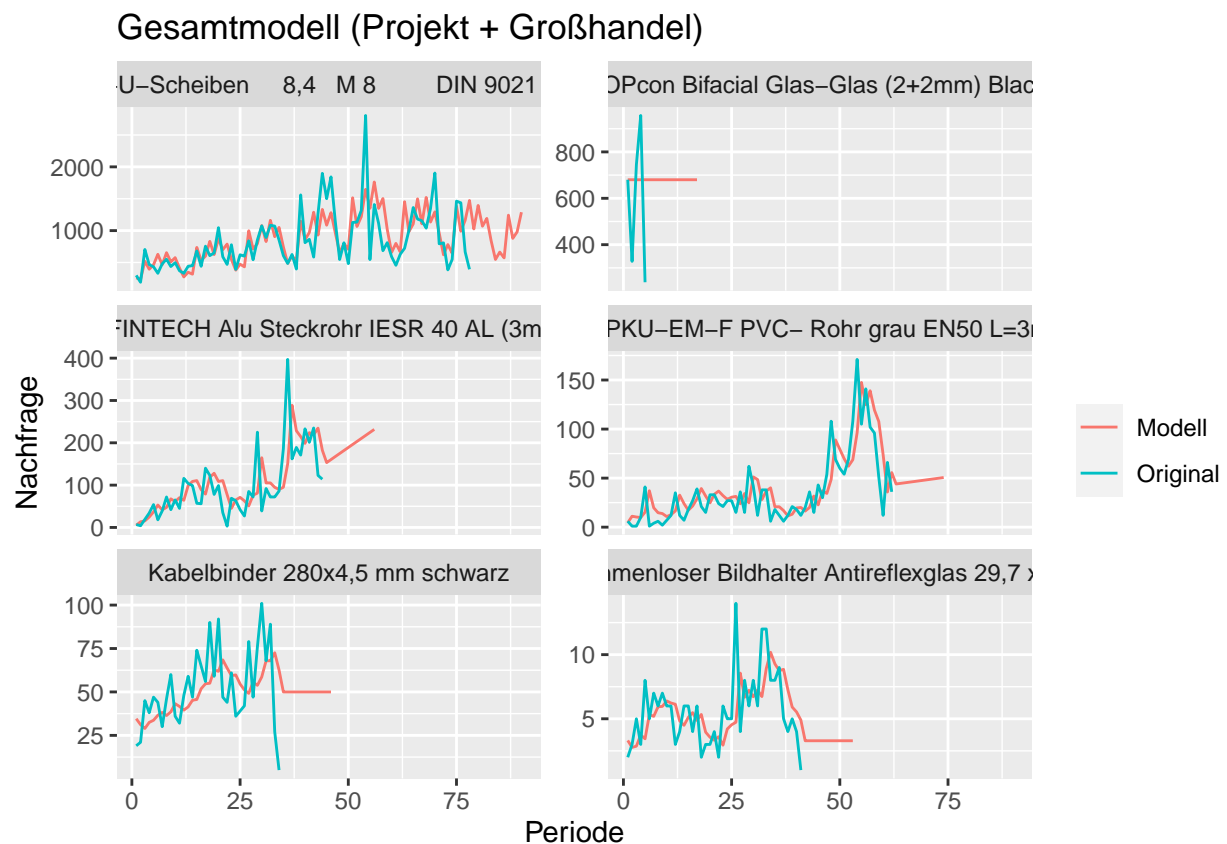
```

```

##      beta  = 0.0211
##
##      Initial states:
##      l = -1.2116
##      b = 5.6891
##
##      sigma: 0.6303
##
##      AIC      AICc      BIC
## 628.0887 629.1601 638.7244
##
## -----
## Current material:  FINTECH Alu Steckrohr IESR 40 AL (3m)ETS(M,A,N)
##
## Call:
## ets(y = ts, model = "ZZZ")
##
##      Smoothing parameters:
##      alpha = 0.5311
##      beta  = 1e-04
##
##      Initial states:
##      l = -1.9594
##      b = 7.1588
##
##      sigma: 0.6049
##
##      AIC      AICc      BIC
## 515.2676 516.8466 524.1886
##
## -----
## Current material:  3d-U-Scheiben      8,4      M 8      DIN 9021  A2ETS(M,A,M)
##
## Call:
## ets(y = ts, model = "ZZZ")
##
##      Smoothing parameters:
##      alpha = 0.0969
##      beta  = 0.0064
##      gamma = 1e-04
##
##      Initial states:
##      l = 365.7297
##      b = 12.6928
##      s = 0.5602 0.8569 1.208 1.0784 1.4002 1.0249
##           1.3602 1.0283 0.9174 1.2909 0.5921 0.6825
##
##      sigma: 0.3298
##
##      AIC      AICc      BIC
## 1223.858 1234.058 1263.922
##

```

```
##
## -----
## Current material:  DAS N-TOPcon Bifacial Glas-Glas (2+2mm) Black 415 WPETS(A,N,N)
##
## Call:
## ets(y = ts, model = "ZZZ")
##
## Smoothing parameters:
##   alpha = 0
##
## Initial states:
##   l = 680
##
## sigma: 282.9212
##
# Für alle Materialien plotten
ggplot(zeitreihen_alle_materialien, aes(x = period, y = bestand, colour = group)) +
  geom_line()+
  facet_wrap(~Artikel, nrow=3, scales="free_y")+
  xlab("Periode") +
  ylab("Nachfrage") +
  ggtitle("Gesamtmodell (Projekt + Großhandel)") +
  theme(legend.title = element_blank())
```



```
# Monatliche Durchschnittsnachfrage - neu gruppieren für richtige Werte
```

```

materials_B_aus <- materials_b %>%
  filter(Menge < 0) %>%
  ## CHANGE MADE HERE ##
  filter(klasse == "Projekt") %>%
  group_by(ArtikelNr, `Artikel-Bezeichnung`, month=as.yearmon(EinAusDat)) %>%
  summarize(menge_pro_monat= sum(-Menge))

# Gesamt-DataFrame
zeitreihen_alle_materialien <- data.frame()

# Projektbetrieb
for (mat in unique(materials_b$`Artikel-Bezeichnung`)) {
  cat("\n\n-----\nCurrent material: ", mat)
  mat_subset <- materials_B_aus %>% filter(`Artikel-Bezeichnung` == mat)
  ts <- ts(mat_subset$menge_pro_monat, frequency = 12)

  # Create model
  model <- ets(ts, model="ZZZ")
  print(model)

  fcast <- forecast(model, 12)

  # DataFrame für Originaldaten
  df_orig = data.frame(
    period = seq(1, length(fcast$x), 1),
    bestand = as.numeric(fcast$x),
    group = rep("Original", length(fcast$x)))

  # DataFrame für Forecast erstellen (sowohl Vorhersage für Zeitreihenwerte als auch ein
  ↪ Jahr in die Zukunft)
  df_fcast = data.frame(
    period = seq(1, length(fcast$fitted)+length(fcast$mean), 1),
    bestand = c(as.numeric(fcast$fitted), as.numeric(fcast$mean)),
    group = rep("Modell", length(fcast$fitted)+length(fcast$mean)))

  # In ein großes DataFrame zusammenführen
  df = rbind(df_orig, df_fcast)

  # In Gesamt-DataFrame hinzufügen
  zeitreihen_alle_materialien <- rbind(zeitreihen_alle_materialien, df %>%
  ↪ mutate(Artikel=mat))

  # Plot
  ggplot(df, aes(x = period, y = bestand, colour = group)) +
    geom_line()+
    xlab("Periode") +
    ylab("Nachfrage") +
    ggtitle("Modell auf Projektdaten für Artikel:", mat) +
    theme(legend.title = element_blank())
}

```

Projektbetrieb

##

```

##
## -----
## Current material:  Rahmenloser Bildhalter Antireflexglas 29,7 x 42ETS(A,N,N)
##
## Call:
## ets(y = ts, model = "ZZZ")
##
## Smoothing parameters:
##   alpha = 0.4128
##
## Initial states:
##   l = 2.7468
##
## sigma: 2.7341
##
##      AIC      AICc      BIC
## 238.6814 239.3301 243.8221
##
## -----
## Current material:  Kabelbinder 280x4,5 mm schwarzETS(A,N,N)
##
## Call:
## ets(y = ts, model = "ZZZ")
##
## Smoothing parameters:
##   alpha = 1e-04
##
## Initial states:
##   l = 5.2227
##
## sigma: 4.1329
##
##      AIC      AICc      BIC
## 49.05493 53.85493 49.64660
##
## -----
## Current material:  FPKU-EM-F PVC- Rohr grau EN50 L=3mETS(M,A,N)
##
## Call:
## ets(y = ts, model = "ZZZ")
##
## Smoothing parameters:
##   alpha = 0.6124
##   beta  = 0.0181
##
## Initial states:
##   l = 13.2664
##   b = 5.4298
##
## sigma: 0.6278
##
##      AIC      AICc      BIC

```

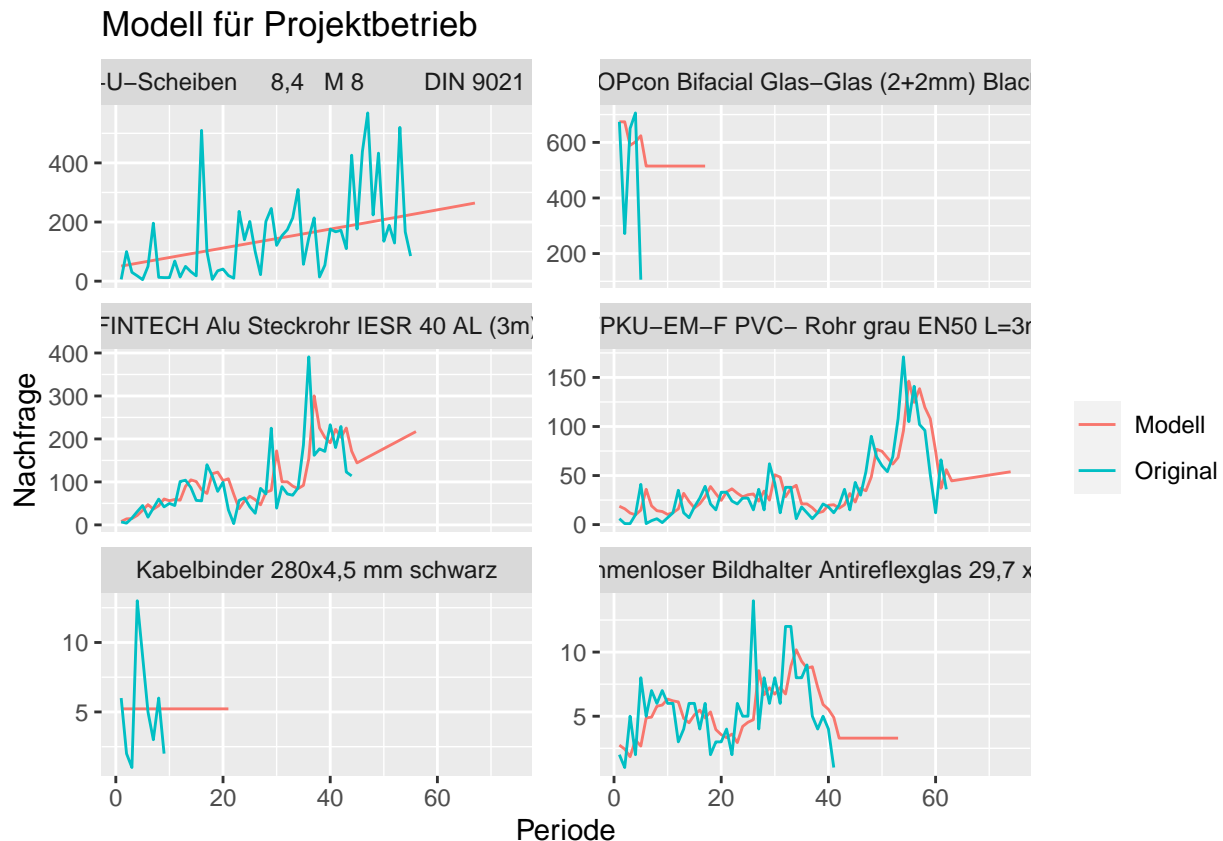
```

## 630.4847 631.5561 641.1203
##
## -----
## Current material:  FINTECH Alu Steckrohr IESR 40 AL (3m)ETS(M,A,N)
##
## Call:
## ets(y = ts, model = "ZZZ")
##
## Smoothing parameters:
##   alpha = 0.5894
##   beta  = 1e-04
##
## Initial states:
##   l = 1.2054
##   b = 6.6495
##
## sigma: 0.5963
##
##      AIC      AICc      BIC
## 509.7666 511.3456 518.6876
##
## -----
## Current material:  3d-U-Scheiben      8,4      M 8      DIN 9021  A2ETS(M,A,N)
##
## Call:
## ets(y = ts, model = "ZZZ")
##
## Smoothing parameters:
##   alpha = 1e-04
##   beta  = 1e-04
##
## Initial states:
##   l = 47.7409
##   b = 3.2325
##
## sigma: 0.9638
##
##      AIC      AICc      BIC
## 755.2122 756.4367 765.2489
##
## -----
## Current material:  DAS N-TOPcon Bifacial Glas-Glas (2+2mm) Black 415 WPETS(A,N,N)
##
## Call:
## ets(y = ts, model = "ZZZ")
##
## Smoothing parameters:
##   alpha = 0.2107
##
## Initial states:
##   l = 674

```

```
##
##   sigma: 298.1249

# Für alle Materialien plotten
ggplot(zeitreihen_alle_materialien, aes(x = period, y = bestand, colour = group)) +
  geom_line()+
  facet_wrap(~Artikel, nrow=3, scales="free_y")+
  xlab("Periode") +
  ylab("Nachfrage") +
  ggtitle("Modell für Projektbetrieb") +
  theme(legend.title = element_blank())
```



```
# Monatliche Durchschnittsnachfrage - neu gruppieren für richtige Werte
materials_B_aus <- materials_b %>%
  filter(Menge < 0) %>%
  ## CHANGE MADE HERE ##
  filter(klasse == "Handel") %>%
  mutate(cum_sum_aus = -cumsum(Menge)) %>%
  group_by(ArtikelNr, `Artikel-Bezeichnung`, month=as.yearmon(EinAusDat)) %>%
  summarize(menge_pro_monat= sum(-Menge))

# Gesamt-DataFrame
zeitreihen_alle_materialien <- data.frame()

for (mat in unique(materials_b$`Artikel-Bezeichnung`)) {
```

```

cat("\n\n-----\nCurrent material: ", mat)
mat_subset <- materials_B_aus %>% filter(`Artikel-Bezeichnung` == mat)
if (nrow(mat_subset) == 0) {
  cat("\nNo data available for ", mat)
  next
}

ts <- ts(mat_subset$menge_pro_monat, frequency = 12)

# Create model
model <- ets(ts, model="ZZZ")
print(model)

fcast <- forecast(model, 12)

# DataFrame für Originaldaten
df_orig = data.frame(
  period = seq(1, length(fcast$x), 1),
  bestand = as.numeric(fcast$x),
  group = rep("Original", length(fcast$x)))

# DataFrame für Forecast erstellen (sowohl Vorhersage für Zeitreihenwerte als auch ein
→ Jahr in die Zukunft)
df_fcast = data.frame(
  period = seq(1, length(fcast$fitted)+length(fcast$mean), 1),
  bestand = c(as.numeric(fcast$fitted), as.numeric(fcast$mean)),
  group = rep("Modell", length(fcast$fitted)+length(fcast$mean)))

# In ein großes DataFrame zusammenführen
df = rbind(df_orig, df_fcast)

# In Gesamt-DataFrame hinzufügen
zeitreihen_alle_materialien <- rbind(zeitreihen_alle_materialien, df %>%
→ mutate(Artikel=mat))

# Plot
ggplot(df, aes(x = period, y = bestand, colour = group)) +
  geom_line()+
  xlab("Periode") +
  ylab("Nachfrage") +
  ggtitle("Modell auf Großhandelsdaten für Artikel:", mat) +
  theme(legend.title = element_blank())
}

```

Großhandel

```

##
##
## -----
## Current material:  Rahmenloser Bildhalter Antireflexglas 29,7 x 42
## No data available for  Rahmenloser Bildhalter Antireflexglas 29,7 x 42
##
## -----
## Current material:  Kabelbinder 280x4,5 mm schwarzETS(M,A,N)

```



```

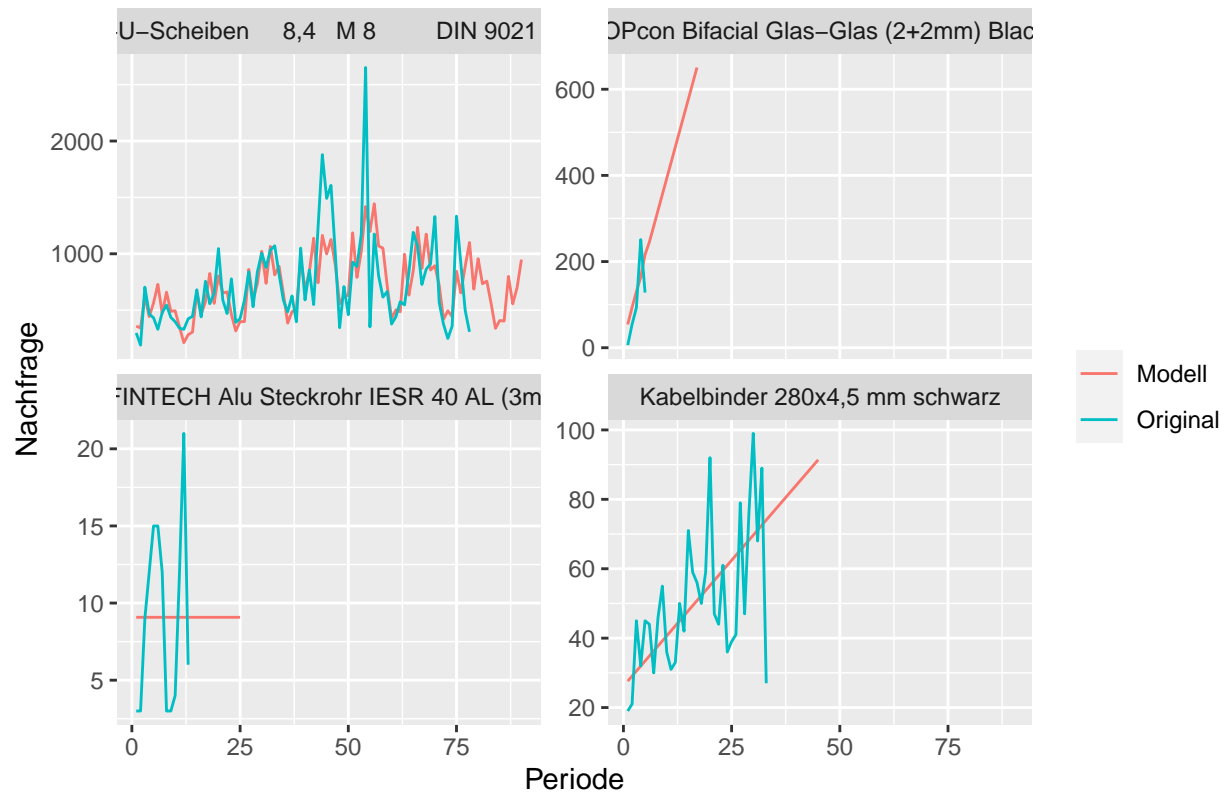
##
## Call:
## ets(y = ts, model = "ZZZ")
##
## Smoothing parameters:
##   alpha = 1e-04
##   beta  = 1e-04
##
## Initial states:
##   l = 26.1664
##   b = 1.4478
##
## sigma: 0.3235
##
##      AIC      AICc      BIC
## 303.2773 305.4995 310.7598
##
## -----
## Current material: FPKU-EM-F PVC- Rohr grau EN50 L=3m
## No data available for FPKU-EM-F PVC- Rohr grau EN50 L=3m
##
## -----
## Current material: FINTECH Alu Steckrohr IESR 40 AL (3m)ETS(A,N,N)
##
## Call:
## ets(y = ts, model = "ZZZ")
##
## Smoothing parameters:
##   alpha = 1e-04
##
## Initial states:
##   l = 9.0768
##
## sigma: 6.1862
##
##      AIC      AICc      BIC
## 84.55315 87.21982 86.24800
##
## -----
## Current material: 3d-U-Scheiben      8,4      M 8      DIN 9021      A2ETS(M,N,M)
##
## Call:
## ets(y = ts, model = "ZZZ")
##
## Smoothing parameters:
##   alpha = 0.191
##   gamma = 1e-04
##
## Initial states:
##   l = 573.6853
##   s = 0.5179 0.8546 1.1579 1.1225 1.4621 1.0496
##       1.4498 1.0721 0.8502 1.2223 0.617 0.624

```

```
##
##   sigma: 0.3712
##
##       AIC       AICc       BIC
## 1214.813 1222.555 1250.163
##
## -----
## Current material:  DAS N-TOPcon Bifacial Glas-Glas (2+2mm) Black 415 WPETS(A,A,N)
##
## Call:
## ets(y = ts, model = "ZZZ")
##
## Smoothing parameters:
##   alpha = 0.0888
##   beta  = 1
##
## Initial states:
##   l = 6
##   b = 48
##
##   sigma: 64.2359

# Für alle Materialien plotten
ggplot(zeitreihen_alle_materialien, aes(x = period, y = bestand, colour = group)) +
  geom_line()+
  facet_wrap(~Artikel, nrow=3, scales="free_y")+
  xlab("Periode") +
  ylab("Nachfrage") +
  ggtitle("Modell für Großhandel") +
  theme(legend.title = element_blank())
```

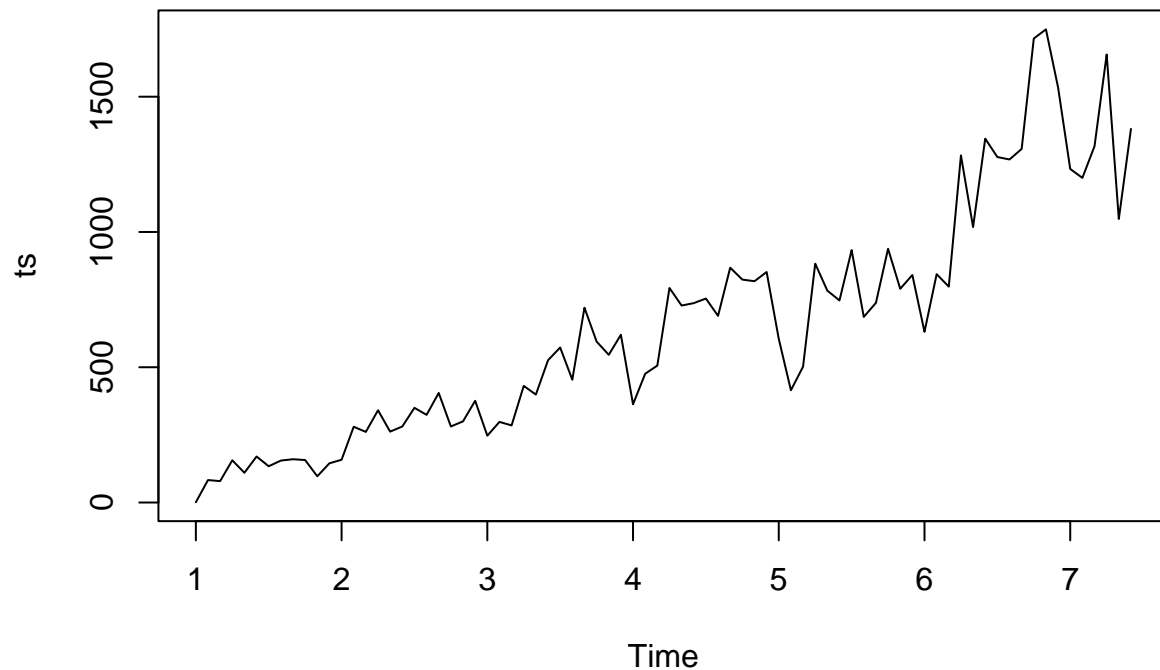
Modell für Großhandel



Zeitreihenanalyse Transaktionsmenge

```
# Monatliche Durchschnittsnachfrage nehmen
transaktionen_monatlich <- materials %>% group_by(Monat=as.yearmon(EinAusDat)) %>%
  ↪ count() %>% filter(Monat < as.yearmon("2023-06"))

ts <- ts(transaktionen_monatlich$n, frequency = 12)
plot.ts(ts)
```



```
# Create model
model <- ets(ts, model="ZZZ")
print(model)
```

```
## ETS(M,N,M)
##
## Call:
## ets(y = ts, model = "ZZZ")
##
## Smoothing parameters:
##   alpha = 0.6115
##   gamma = 1e-04
##
## Initial states:
##   l = 212.0323
##   s = 0.9929 0.9355 1.0884 1.1332 0.9892 1.1195
##       1.1008 0.9353 1.2221 0.829 0.9096 0.7446
##
## sigma: 0.2433
##
##      AIC      AICc      BIC
## 1095.935 1103.677 1131.286
```

```
fcast <- forecast(model, 12)
```

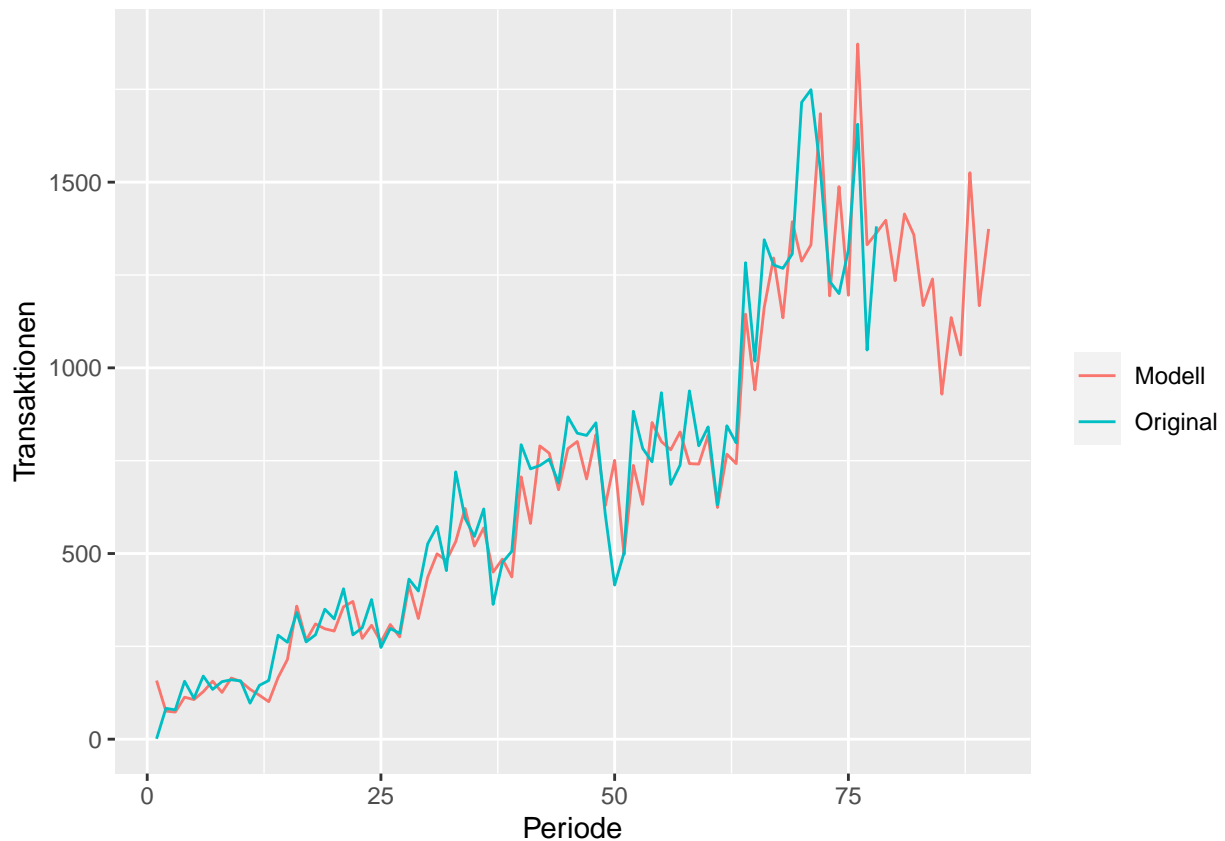
```
# DataFrame für Originaldaten
```

```
df_orig = data.frame(
  period = seq(1, length(fcast$x), 1),
  bestand = as.numeric(fcast$x),
  group = rep("Original", length(fcast$x)))

# DataFrame für Forecast erstellen (sowohl Vorhersage für Zeitreihenwerte als auch ein
#   ↳ Jahr in die Zukunft)
df_fcast = data.frame(
  period = seq(1, length(fcast$fitted)+length(fcast$mean), 1),
  bestand = c(as.numeric(fcast$fitted), as.numeric(fcast$mean)),
  group = rep("Modell", length(fcast$fitted)+length(fcast$mean)))

# In ein DataFrame zusammenführen
df = rbind(df_orig, df_fcast)

# Plot
ggplot(df, aes(x = period, y = bestand, colour = group)) +
  geom_line()+
  xlab("Periode") +
  ylab("Transaktionen") +
  theme(legend.title = element_blank())
```

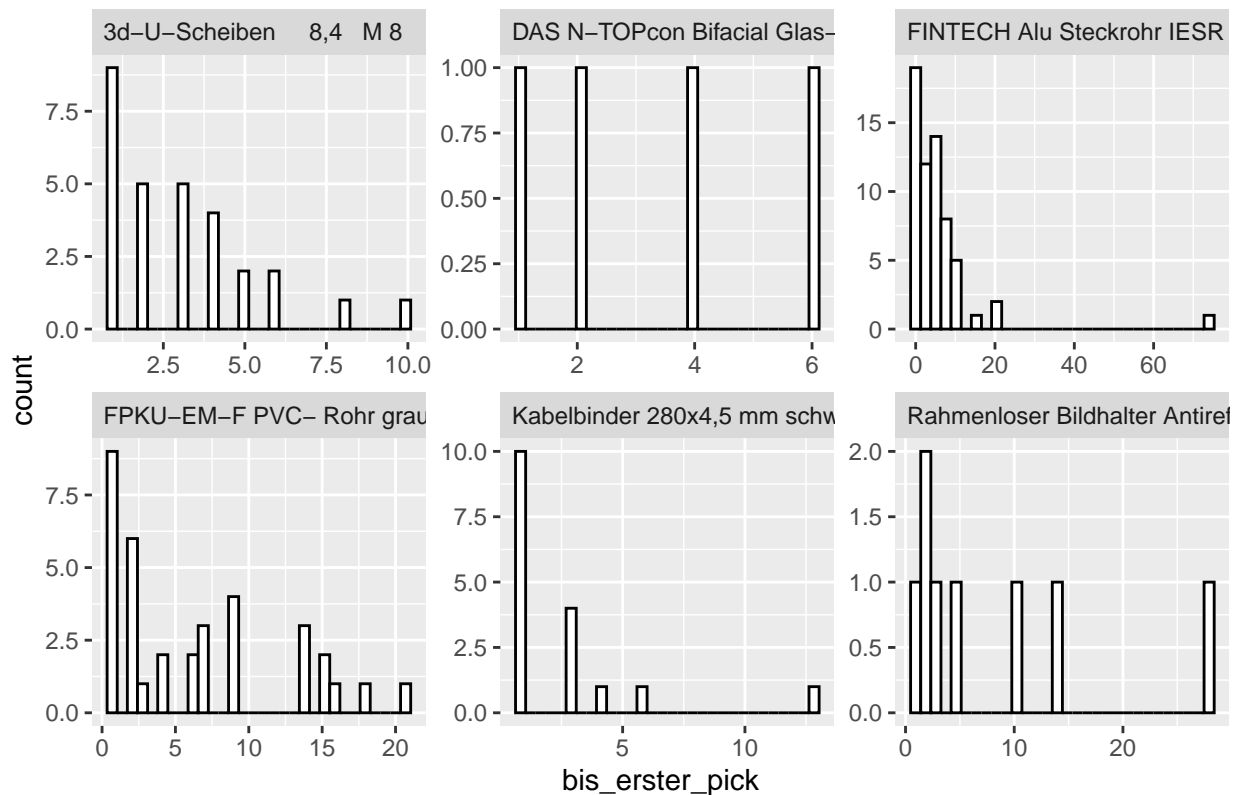


Reichweite

Zeit bis zum ersten Pick

```
# Reichweite bis 1. Pick = Zeit zwischen Bestelleingang (geliefertAm) und erster  
↳ Transaktion (BereitDat, PjPhaseDat)  
materials_b_1st_pick <- materials_b %>%  
  # Nach Datum sortieren  
  arrange(EinAusDat) %>%  
  # Nach Material gruppieren  
  group_by(ArtikelNr, `Artikel-Bezeichnung`) %>%  
  mutate(bis_erster_pick = if_else(lead(EinAusDatTyp) %in% c("BereitDat", "PjPhaseDat"),  
    ↳ as.numeric(difftime(lead(EinAusDat), EinAusDat, units="days")), NA)) %>%  
  filter(EinAusDatTyp == "geliefertAm")  
  # summarize(mean_bis_pick=mean(bis_erster_pick, na.rm=TRUE)) # NA remove for last entry  
  
# Histogramm  
ggplot(materials_b_1st_pick, aes(x=bis_erster_pick)) +  
  geom_histogram(color="black", fill="white") +  
  facet_wrap(~`Artikel-Bezeichnung`, scales = 'free') +  
  # Text linksbündig  
  theme(strip.text.x = element_text(hjust=0.0)) +  
  ggtitle("Verteilung der Zeit von Bestelleingang bis zum ersten Pick [in Tagen]")
```

Verteilung der Zeit von Bestelleingang bis zum ersten Pick [in Tagen]



```
# Zeit zwischen Picks  
# Mit WBZ gegenüberstellen
```

Reichweite bis nächste Bestellung

```
# = Zeit zwischen Bestelleingang und nächster Bestellung (BstDat -> wenn wir bestellen,
↳ nicht Eingang)
materials_b_n_bestellung <- materials_b %>%
  arrange(EinAusDat, ArtikelNr) %>%
  filter(EinAusDatTyp == "geliefertAm") %>%
  group_by(ArtikelNr, `Artikel-Bezeichnung`) %>%
  mutate(bis_n_bestl = as.numeric(difftime(lead(BstDat), EinAusDat, units="days")))
  # summarize(mean_bis_bestl=mean(bis_n_bestl, na.rm=TRUE)) # NA remove for last entry

# Histogramm
ggplot(materials_b_n_bestellung, aes(x=bis_n_bestl)) +
  geom_histogram(color="black", fill="white") +
  facet_wrap(~`Artikel-Bezeichnung`, scales = 'free') +
  # Text linksbündig
  theme(strip.text.x = element_text(hjust=0.0)) +
  ggtitle("Verteilung der Reichweite von Bestelleingang bis zur nächsten Bestellung [in
↳ Tagen]")
```

Verteilung der Reichweite von Bestelleingang bis zur nächsten Bestellung [in

