ACME Case Study

Team 14

4 6 2021

Einführung

Datensatz lesen und generellen Überblick verschaffen.

Zusammenfassung ausgeben

```
summary(event_log)
```

```
##    CASE_ID            ACTIVITY           TIMESTAMP
##  Length:178078      Length:178078      Min.   :2013-05-22 10:39:39
##  Class :character   Class :character   1st Qu.:2018-06-11 09:41:52
##  Mode  :character   Mode  :character   Median :2018-10-31 10:17:36
##                                        Mean   :2018-10-16 14:14:51
##                                        3rd Qu.:2019-02-23 10:12:57
##                                        Max.   :2019-06-28 08:39:30
##  REPAIR_IN_TIME_5D  DEVICETYPE        SERVICEPOINT
##  Min.   :0.000     Length:178078      Length:178078
##  1st Qu.:0.000     Class :character   Class :character
##  Median :0.000     Mode  :character   Mode  :character
##  Mean   :0.326
##  3rd Qu.:1.000
##  Max.   :1.000
```

Ersten 10 Datensätzen ausgeben

```
head(event_log, n=10)
```

```
## # A tibble: 10 x 6
##    CASE_ID ACTIVITY TIMESTAMP           REPAIR_IN_TIME_~ DEVICETYPE SERVICEPOINT
##    <chr>   <chr>    <dttm>                         <dbl> <chr>      <chr>
##  1 Case10  Creation 2018-01-02 13:39:47                0 AB52       E
##  2 Case10  Letter   2018-01-05 00:00:00                0 AB52       E
##  3 Case10  DeviceR~ 2018-01-05 16:45:34                0 AB52       E
##  4 Case10  StockEn~ 2018-01-17 00:00:00                0 AB52       E
##  5 Case10  InDeliv~ 2018-01-17 00:00:00                0 AB52       E
##  6 Case10  NoteWor~ 2018-01-17 07:37:19                0 AB52       E
##  7 Case10  Complet~ 2018-01-17 09:34:32                0 AB52       E
##  8 Case100 Creation 2018-01-02 15:43:48                0 AB41       E
##  9 Case100 NoteHot~ 2018-01-02 15:44:41                0 AB41       E
## 10 Case100 Letter   2018-01-08 00:00:00                0 AB41       E
```

Wertebereich für interessante Spalten ausgeben

```
unique(event_log$ACTIVITY)
```

```
##  [1] "Creation"       "Letter"         "DeviceReceived" "StockEntry"
##  [5] "InDelivery"     "NoteWorkshop"   "Completed"      "NoteHotline"   
##  [9] "StatusRequest"  "Transmission"   "Approved"       "FreeticketCust"
## [13] "FreeticketComp"
```

```
unique(event_log$DEVICETYPE)
```

```
##  [1] "AB52" "AB41" "AB47" "AB22" "AB49" "AB62" "AB29" "AB63" "AB20" "AB53"
## [11] "AB50" "AB44" "AB45" "AB36" "AB61" "AB16" "AB34" "AB25" "AB40" "AB8" 
## [21] "AC68" "AB38" "AB65" "AB60" "AB31" "AB27" "AB10" "AB19" "AB59" "AB21"
## [31] "AB56" "AB26" "AB55" "AB9"  "AB58" "AB39" "AB14" "AB43" "AB24" "A07" 
## [41] "AB57" "AB23" "AB28" "AB64" "AB32" "AB15" "AB30" "AF3"  "AB33" "AG5" 
## [51] "AB12" "AB51" "AB54" "AB18" "AB17" "AB35" "AB46" "AB37" "AB48" NA    
## [61] "AB42" "AG4"  "AB66" "AB67" "AB13"
```

```
unique(event_log$SERVICEPOINT)
```

```
##  [1] "E" "G" "J" "L" NA  "C" "H" "I" "K" "D" "B" "A"
```

```
unique(event_log$REPAIR_IN_TIME_5D)
```

```
## [1] 0 1
```

Datenreinigung

Datensätze ohne Angabe zu Servicepoint oder Gerät ausschließen.

```
clean_events <- na.omit(event_log)
```

Datensatz aus unvollständigen Sätzen abspalten.

```
corrupted_events <- subset(event_log,is.na(DEVICETYPE) | is.na(SERVICEPOINT))
```

Datenanalyse

Grundlagen

Wie viele verschiedene Bearbeitungsfälle gibt es?

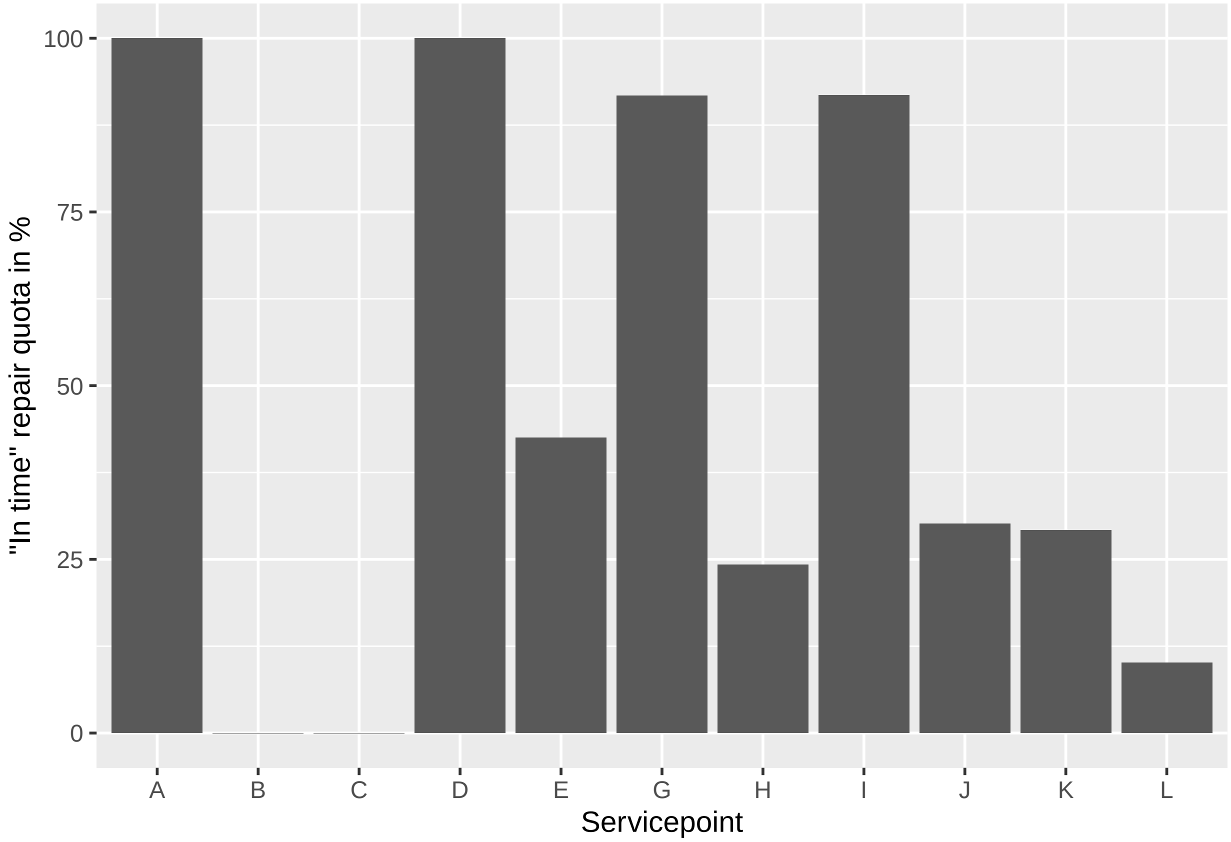
```
unique(clean_events$CASE_ID) %>% length()
```

```
## [1] 21931
```

Visualisierung

5-Tage-Reperatur-Quote nach Servicepoint

```
# das reperaturzeit flag steht in jedem eintrag eines CASEs, uns langt aber ein
# Eintrag je CASE
distinct_cases <- distinct(clean_events, CASE_ID, .keep_all = TRUE)

# nach Servicepoint gruppieren und die flags aufsummieren
by_servicep <- group_by(distinct_cases, SERVICEPOINT) %>%
  summarise(fivedaysum = sum(REPAIR_IN_TIME_5D), all = n())
# quote berechnen mit anzahl der "schnellen" CASEs
by_servicep$quota = by_servicep$fivedaysum / by_servicep$all * 100

# plot zeichnen
plot <- ggplot(data = by_servicep, aes(x=SERVICEPOINT, y=quota)) +
  geom_bar(stat="identity") + ylab("\"In time\" repair quota in %")+
  xlab("Servicepoint")
plot
```

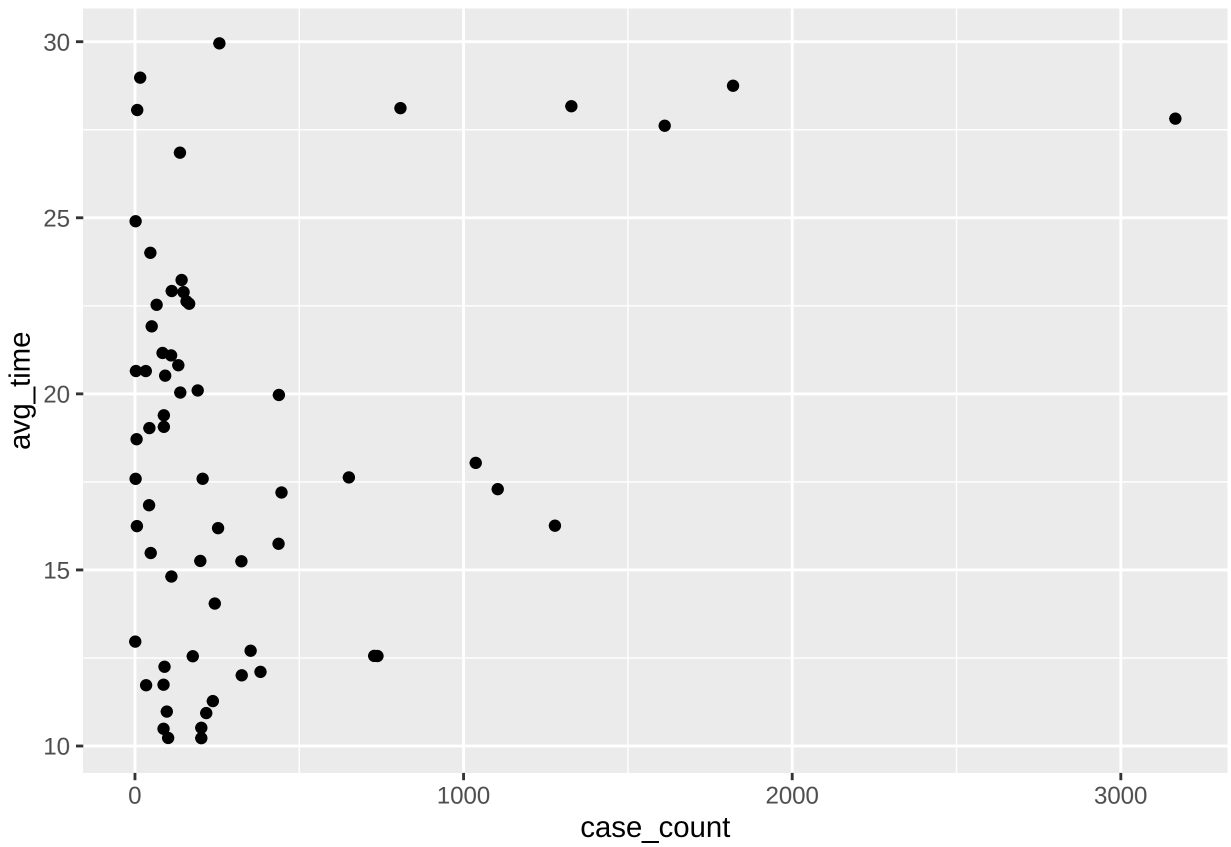


Clusteranalyse Anfälligkeit der Geräte

```
case_total_duration <- group_by(clean_events, CASE_ID) %>%
  group_by(DEVICETYPE, .add = TRUE) %>%
  summarise(timemax = max(TIMESTAMP), timemin = min(TIMESTAMP), duration = timemax - timemin)
```

```
## `summarise()` has grouped output by 'CASE_ID'. You can override using the `.groups` argument.
```

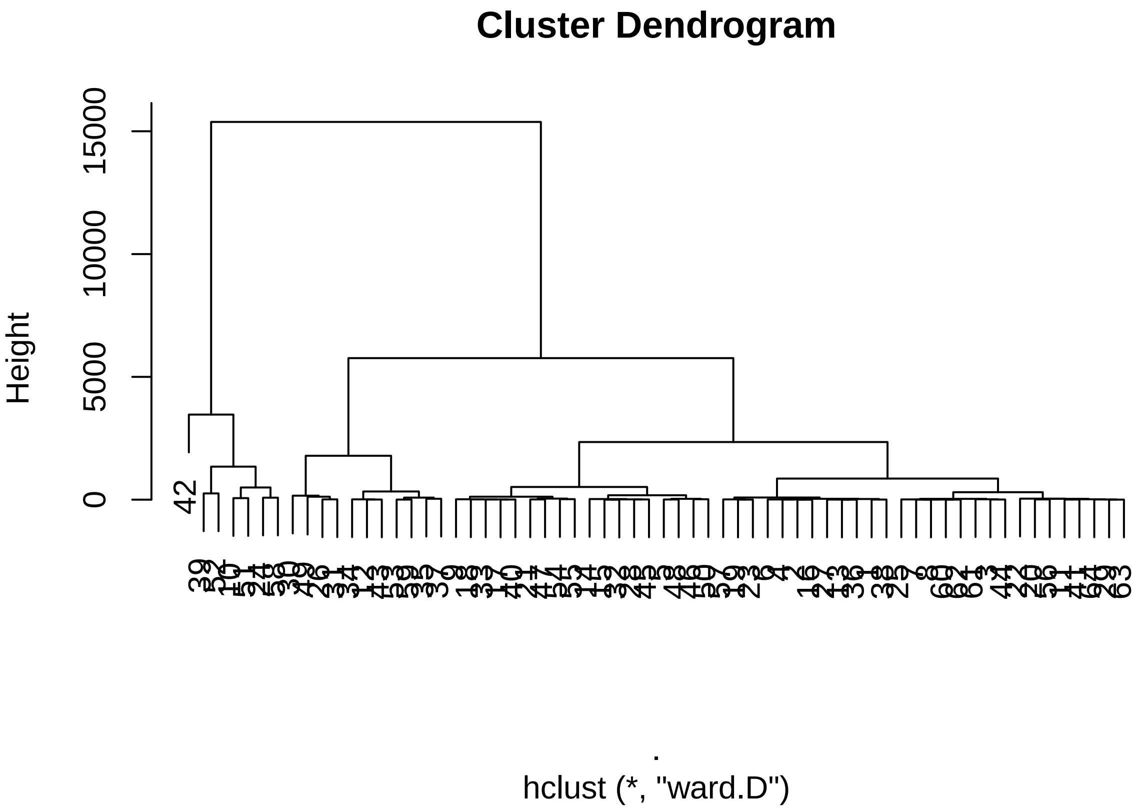
```
avg_duration_by_device <- case_total_duration %>%
  group_by(DEVICETYPE) %>%
  summarize(case_count = n(), avg_time = mean(duration))

avg_duration_by_device$avg_time <- as.numeric(avg_duration_by_device$avg_time, units="days")

ggplot(avg_duration_by_device, aes(case_count, avg_time)) + geom_point()
```



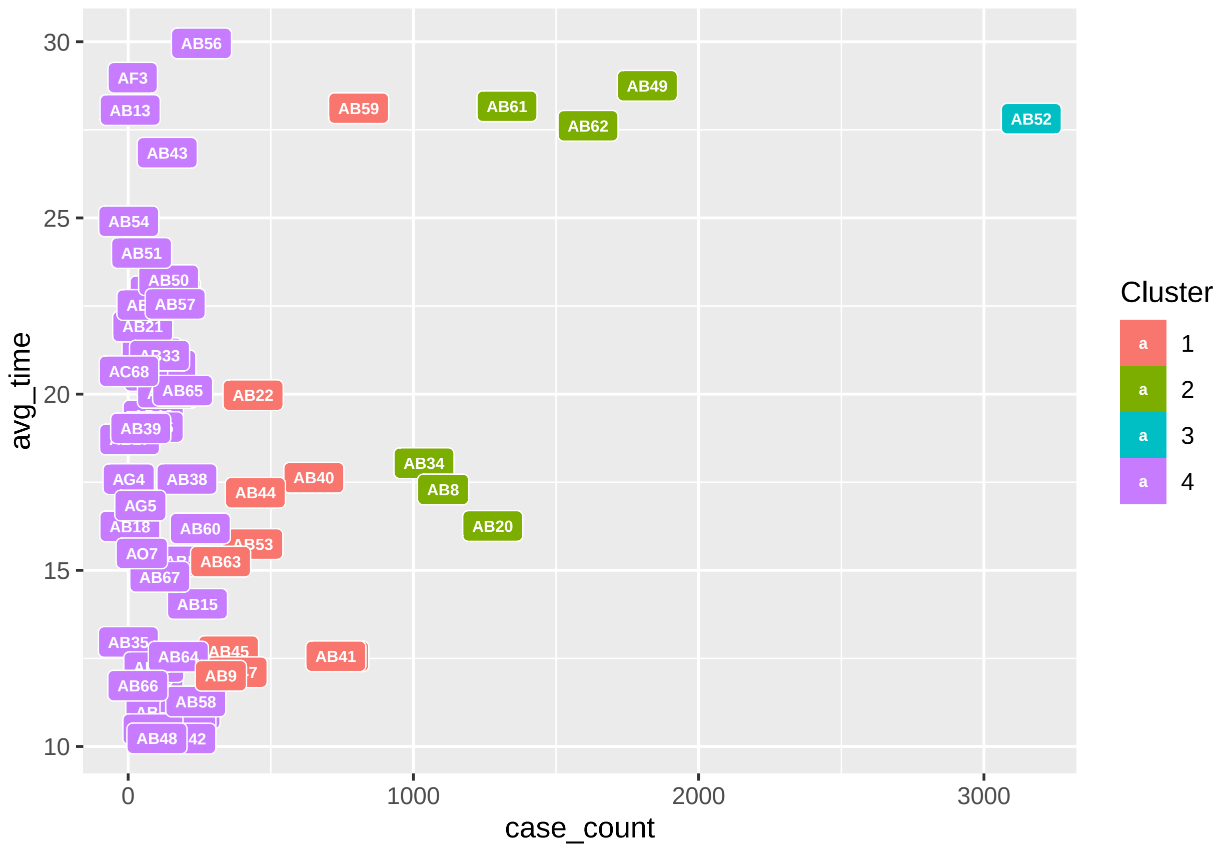
```
h.cluster <- avg_duration_by_device %>% dist(., method = "euclidean") %>% hclust(., method="ward.D")
```

```
## Warning in dist(., method = "euclidean"): NAs durch Umwandlung erzeugt
```

```
plot(h.cluster)
```

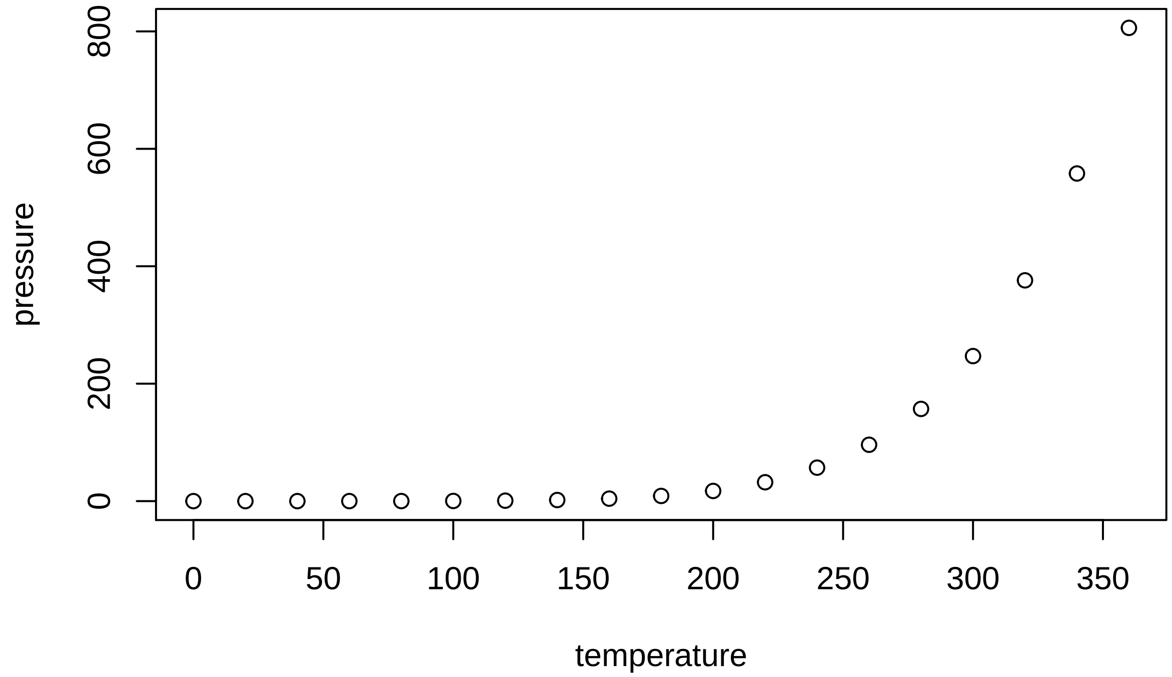


```
p.cluster <- avg_duration_by_device %>% select(avg_time, case_count) %>% kmeans(., 4)
p.cluster$cluster <- as.factor(p.cluster$cluster)

ggplot(avg_duration_by_device, aes(case_count, avg_time, label = DEVICETYPE)) +
  scale_fill_discrete(name = "Cluster") +
  geom_label(aes(fill = p.cluster$cluster), colour = "white", fontface = "bold", size = 2)
```



Including Plots

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.