

SCA_WS2021_Gruppe105_HA1

R Markdown

- 1) Alle 4 Dateien wurden importiert und wurden als Variablen mit passenden Namen gespeichert. Eine Zusammenfassung von jeder Variable wird ausgegeben.

```
cost = read.csv2("output_cost_8Players_v0010.csv")
prices = read.csv2("output_prices_8Players_v0010.csv")
services = read.csv2("output_services_8Players_v0010.csv")
transactions = read.csv2("output_transactions_8Players_v0010.csv")
```

Zusammenfassung einer Variable anzeigen

```
summary(cost)
```

```
##      Year      Month      Product      Amount
## Min.   :2016   Min.    : 1.00   Length:480   Min.    :25774
## 1st Qu.:2017   1st Qu.: 3.75   Class :character 1st Qu.:31024
## Median :2018   Median : 6.50   Mode  :character Median :32361
## Mean   :2018   Mean    : 6.50           Mean   :32245
## 3rd Qu.:2019   3rd Qu.: 9.25           3rd Qu.:33450
## Max.   :2020   Max.    :12.00          Max.    :36893
```

```
summary(prices)
```

```
##      vendor      service      PriceSpot      PriceContract
## Length:20      Length:20      Min.    :1.181   Min.    :0.7086
## Class :character Class :character 1st Qu.:1.254   1st Qu.:0.7523
## Mode  :character Mode  :character Median :1.282   Median :0.7692
##                                     Mean  :1.284   Mean  :0.7704
##                                     3rd Qu.:1.317 3rd Qu.:0.7900
##                                     Max.   :1.362  Max.   :0.8169
## FeeContractPerUnit PriceChange      UsageLastPeriod
## Min.    :0.1890   Min.    :-0.0507140   Min.    : 7563
## 1st Qu.:0.2006   1st Qu.: -0.0140229 1st Qu.: 9088
## Median :0.2051   Median : -0.0028121 Median : 9650
## Mean    :0.2054   Mean    : 0.0004893 Mean    : 9704
## 3rd Qu.:0.2107   3rd Qu.: 0.0145601 3rd Qu.:10627
## Max.    :0.2178   Max.    : 0.0499259 Max.    :11362
```

```
summary(services)
```

```
##      Year      Month      Day      region
## Min.   :2016   Min.    : 1.000   Min.    : 1.00   Length:105984
## 1st Qu.:2017   1st Qu.: 3.000   1st Qu.: 5.00   Class :character
## Median :2018   Median : 6.000   Median :10.00   Mode  :character
## Mean   :2018   Mean    : 6.494   Mean    :10.48
## 3rd Qu.:2019   3rd Qu.: 9.000   3rd Qu.:15.00
## Max.   :2020   Max.    :12.000   Max.    :20.00
## storename      Product      vendor      service
## Length:105984   Length:105984   Length:105984   Length:105984
```

```
## Class :character Class :character Class :character Class :character
## Mode :character Mode :character Mode :character Mode :character
##
##
##
## DaysScheduled DaysExecuted QScheduled QExecuted cost
## Min. :0 Min. :0.00 Min. : 34.0 Min. : 21.0 Min. :
40.9
## 1st Qu.:0 1st Qu.:0.00 1st Qu.: 86.0 1st Qu.: 74.0 1st
Qu.:110.2
## Median :1 Median :1.00 Median : 97.0 Median : 92.0 Median
:126.8
## Mean :1 Mean :1.38 Mean :112.6 Mean :103.3 Mean
:146.0
## 3rd Qu.:2 3rd Qu.:3.00 3rd Qu.:140.0 3rd Qu.:129.0 3rd
Qu.:181.3
## Max. :2 Max. :4.00 Max. :367.0 Max. :367.0 Max.
:498.4
```

`summary(transactions)`

```
## Year Month Day region
## Min. :2016 Min. : 1.000 Min. : 1.00 Length:295904
## 1st Qu.:2017 1st Qu.: 3.000 1st Qu.: 5.00 Class :character
## Median :2018 Median : 6.000 Median :10.00 Mode :character
## Mean :2018 Mean : 6.488 Mean :10.48
## 3rd Qu.:2019 3rd Qu.: 9.000 3rd Qu.:15.00
## Max. :2021 Max. :12.000 Max. :20.00
## storename Product Sales Received
## Length:295904 Length:295904 Min. : 0.0 Min. : 0.00
## Class :character Class :character 1st Qu.: 0.0 1st Qu.: 0.00
## Mode :character Mode :character Median : 17.0 Median : 0.00
## Mean : 17.2 Mean : 16.91
## 3rd Qu.: 27.0 3rd Qu.: 0.00
## Max. :193.0 Max. :313.00
```

- 2) Extrahieren Sie aus den Transaktionsdaten eine Tabelle aller existierenden Supermärkte, in der jeder Supermarkt nur einmal enthalten ist.

```
df = data.frame(unique(transactions$storename,
                      incomparables = FALSE, fromLast = FALSE))
colnames(df) = "Store Name"
df
```

```
## Store Name
## 1 Shangh-1
## 2 Shangh-2
## 3 Shangh-3
## 4 Shangh-4
## 5 Shangh-5
## 6 Peking-1
## 7 Peking-2
## 8 Peking-3
```

```
## 9    Peking-4
## 10   Peking-5
## 11    Japan-1
## 12    Japan-2
## 13    Japan-3
## 14    Japan-4
## 15    Japan-5
## 16   Skorea-1
## 17   Skorea-2
## 18   Skorea-3
## 19   Skorea-4
## 20   Skorea-5
## 21   Phlppn-1
## 22   Phlppn-2
## 23   Phlppn-3
## 24   Phlppn-4
## 25   Phlppn-5
```

- 3) Extrahieren Sie aus den Transaktionsdaten eine Tabelle aller existierenden Produkte, in der jedes Produkt nur einmal enthalten ist.

```
df = data.frame(unique(subset(transactions, Product != "Lost
Sales")$Product))
colnames(df) = "Products"
df
```

```
##    Products
## 1 Gruppe101
## 2 Gruppe102
## 3 Gruppe103
## 4 Gruppe104
## 5 Gruppe105
## 6 Gruppe106
## 7 Gruppe107
## 8 Gruppe108
```

- 4) Extrahieren Sie aus den Servicedaten eine Tabelle aller 20 Logistikdienstleister mitsamt deren Dienstleistungen. Jeder Logistikdienstleister soll in der Liste nur einmal enthalten sein. Sortieren Sie die Tabelle nach Shipping-DL und Warehousing-DL.

```
df = unique(services[c("vendor", "service")])
df = df[order(df$service, df$vendor),]
df
```

```
##           vendor      service
## 12    AHL Express Shipping Shipping
## 10    Bange+Hammer Shipping Shipping
## 24           CPS Shipping Shipping
## 2      DWL Shipping Shipping
## 36          EPD Shipping Shipping
## 14  Flying Mercury Shipping Shipping
## 4           Gifter Shipping Shipping
## 28           HCX Shipping Shipping
```

```
## 6          IntEx Shipping    Shipping
## 18          JNT Shipping    Shipping
## 27    AHL Express Warehousing Warehousing
## 3    Bange+Hammer Warehousing Warehousing
## 43          CPS Warehousing Warehousing
## 1          DWL Warehousing Warehousing
## 71          EPD Warehousing Warehousing
## 15 Flying Mercury Warehousing Warehousing
## 5          Gifter Warehousing Warehousing
## 33          HCX Warehousing Warehousing
## 9          IntEx Warehousing Warehousing
## 57          JNT Warehousing Warehousing
```

- 5) Berechnen Sie fuer Ihre Produkte/Gruppe die verkauften Mengen (Sales) je Region. Nutzen Sie eine einzige verkettete Anweisung fuer diese Aufgabe.

```
df = data.frame(aggregate(Sales ~ region, data = subset(transactions,
                                                         Product == "Gruppe105"),
                        sum))
head(df)

##    region Sales
## 1  Japan 119480
## 2 Peking  82691
## 3 Phlppn 128612
## 4 Shangh 147368
## 5 Skorea 138546
```

- 6) Berechnen Sie fuer jede Region den Anteil Ihres Produkts an der tatsaechlich verkauften Menge (in %).

```
df = aggregate(Sales ~ region, data = subset(transactions,
                                              Product=="Gruppe105"), sum)
df$SalesInPercent = round(df$Sales / (aggregate(Sales ~ region,
                                              data=transactions, sum))$Sales * 100,
2)
colnames(df) = c("Region", "Sales", "Sales in %")
df

##    Region Sales Sales in %
## 1  Japan 119480      11.72
## 2 Peking  82691       8.19
## 3 Phlppn 128612      12.46
## 4 Shangh 147368      14.47
## 5 Skorea 138546      13.73
```

cat("Um den Anteil unserer Produkte der Gruppe105 zu berechnen, haben wir alle Sales nach den jeweiligen Regionen aggregiert und sie nach Gruppe105 gefiltert. Anschließend haben wir unseren prozentualen Anteil der verkauften Menge berechnet.")

Um den Anteil unserer Produkte der Gruppe105 zu berechnen, haben wir alle Sales nach den jeweiligen Regionen aggregiert und sie nach Gruppe105

gefiltert. Anschließend haben wir unseren prozentualen Anteil der verkauften Menge berechnet.

- 7) Berechnen Sie die durchschnittliche Nachfrage an verkauften Produkten pro Tag nach Ihren Produkten/Gruppe je Region. Schätzen Sie anschließend die Monatsnachfrage mit einer Rechnung. Geben Sie alles in einer Tabelle aus. Bewertungsrelevant: Output.

```
df = aggregate(cbind(Sales, Day) ~ region, data = subset(transactions,
                                                         Product == "Gruppe105"),
sum)
df$AvgSalesPerDay = round(df$Sales / df$Day, 2)

# Für die Schätzung der Monatsnachfrage rechnen wir mit 21 Arbeitstage im
# Monat (laut www.arbeitsreche.de), die wir mit AvgSalesPerDay multiplizieren.
df$MonthlyDemand = df$AvgSalesPerDay * 21
colnames(df) = c("Region", "Sales", "Day",
                 "AvgSalesperDay", "EstimatedAvgMonthlySales")
df
```

	Region	Sales	Day	AvgSalesperDay	EstimatedAvgMonthlySales
## 1	Japan	119480	76707	1.56	32.76
## 2	Peking	82691	77058	1.07	22.47
## 3	Phlppn	128612	77017	1.67	35.07
## 4	Shangh	147368	76363	1.93	40.53
## 5	Skorea	138546	76829	1.80	37.80

```
cat("Um die durchschnittliche Nachfrage an verkauften Produkten für Gruppe105
pro Tag zu berechnen, haben wir die Summe der Sales und die Summe der
Arbeitstage nach Regionen aggregiert, um anschließend die jeweiligen Summen
miteinander zu dividieren. In der vierten Spalte ergibt sich die Anzahl
durchschnittlicher Nachfrage. Diese Zahlen haben wir mit 21 Arbeitstage
multipliziert, um eine Abschaetzung der Monatsnachfrage aufzuzeigen.")

## Um die durchschnittliche Nachfrage an verkauften Produkten für Gruppe105
pro Tag zu berechnen, haben wir die Summe der Sales und die Summe der
Arbeitstage nach Regionen aggregiert, um anschließend die jeweiligen Summen
miteinander zu dividieren. In der vierten Spalte ergibt sich die Anzahl
durchschnittlicher Nachfrage. Diese Zahlen haben wir mit 21 Arbeitstage
multipliziert, um eine Abschaetzung der Monatsnachfrage aufzuzeigen.
```

- 8) Geben Sie den Logistikdienstleister aus, fuer den Ihre Gruppe im Jahr 2018 die niedrigsten Kosten zu verzeichnen hat.

```
MinCost = data.frame(aggregate(cost ~ vendor, data = subset(services,
                                                             services$Product == "Gruppe105" & services$Year == 2018),
sum))
MinCost <- MinCost[order(MinCost$cost, decreasing = FALSE),]
head(MinCost, 1)
```

	vendor	cost
## 8	DWL Warehousing	14865.11

- 9) Berechnen Sie fuer die vergangenen 5 Jahre den Profit je Jahr. Bewertungsrelevant: Output.

```
df = data.frame(aggregate(Sales ~ Year, data = subset(transactions,
                                                       transactions$Product == "Gruppe105"), sum))
head(df, 5)

##   Year  Sales
## 1 2016 122920
## 2 2017 123350
## 3 2018 123554
## 4 2019 122838
## 5 2020 124035
```

10) Berechnen Sie den Profit des letzten halben Jahres im Datensatz (07/2020 - 12/2020) je Monat. Erzeugen Sie hierfür ein DataFrame profit, welche den Profit je Monat als Variablen speichert. Berechnen Sie ausserdem die Veraenderung des Profits (in %) von Monat zu Monat ueber die letzten 6 Monate als zusaetzliche Spalte fuer das DataFrame profit. Dabei sollte die Veraenderung fuer die erste Zeile 0 sein. Geben Sie das DataFrame aus. Bewertungsrelevant: Code, Output.

```
profit = data.frame(aggregate(Sales ~ Month,
                              data = subset(transactions,
                                              transactions$Month >= 7 &
                                              transactions$Month <= 12 &
                                              transactions$Year == 2020 &
                                              transactions$Product == "Gruppe105"), sum))

# Erklärung zu Column3:
# Dritte Spalte, erste Zeile ist 0.
# Die Funktion tail(Sales,-1) zeigt auf die zweite Sales-Zeile.
# Die Funktion head(Sales,-1) zeigt auf die erste Sales-Zeile.
# Diese beiden Zeilen werden miteinander dividiert und mit round() auf die
# zweite Nachkommastelle gerundet.

profit = cbind(profit, with(profit, {
  data.frame( #Columnx = Sales/Sales[1],
              Column3 = c(0, (round(1-(tail(Sales,-1)/
                                         head(Sales,-1)), 2) *100)
            )
}))
colnames(profit) = c("Month", "Sales", "Profit in %")
profit
```

	Month	Sales	Profit in %
## 1	7	10240	0
## 2	8	9641	6
## 3	9	10746	-11
## 4	10	10340	4
## 5	11	9561	8
## 6	12	10193	-7

11) Berechnen Sie, wie viel fuer Ihre Produkte/Gruppe ueber den gesamten Zeitraum fuer Transportdienstleistungen ausgegeben wurde. Berechnen Sie dazu 2 Kennzahlen (einzeln zu berechnen):

- (1) Wie viel wurde fuer verspaeetete Transportdienstleistungen ausgegeben und
 - (2) wie viel wurde fuer puenktliche Transportdienstleistungen ausgegeben.
- Bewertungsrelevant: Output.

Hier wird DaysScheduled mit 0 aussortiert, da kein Zeitraum zu ermitteln ist.

```
df1 = data.frame(aggregate(cost ~ vendor,
                           data = subset(services,
                                           Product == "Gruppe105" &
                                           DaysScheduled != 0 &
                                           DaysExecuted > DaysScheduled), sum))
colnames(df1) = c("vendor", "cost of late service")
df1
```

```
##           vendor cost of late service
## 1    AHL Express Shipping      60022.63
## 2  Bange+Hammer Shipping      73199.56
## 3         CPS Shipping      68791.91
## 4         DWL Shipping      60158.30
## 5         EPD Shipping      74749.08
## 6  Flying Mercury Shipping      61165.63
## 7         Gifter Shipping      63408.30
## 8         HCX Shipping      65977.54
## 9        IntEx Shipping      58626.59
## 10        JNT Shipping      62641.65
```

```
df2 = data.frame(aggregate(cost ~ vendor,
                           data = subset(services,
                                           Product == "Gruppe105" &
                                           DaysScheduled != 0 &
                                           DaysScheduled <= DaysExecuted), sum))
colnames(df2) = c("vendor", "cost of timely service")
df2
```

```
##           vendor cost of timely service
## 1    AHL Express Shipping      87459.01
## 2  Bange+Hammer Shipping     110238.16
## 3         CPS Shipping      94539.51
## 4         DWL Shipping      91954.49
## 5         EPD Shipping      99618.02
## 6  Flying Mercury Shipping      99202.63
## 7         Gifter Shipping      94772.56
## 8         HCX Shipping      93010.29
## 9        IntEx Shipping      95401.17
## 10        JNT Shipping      91731.74
```

- 12) Berechnen Sie, wie viel Sie die reale Ausfuehrung einer *Lagerdienstleistung* tatsaechlich kostet. D.h. rechnen Sie die Gesamtkosten, die Ihr Produkt/Gruppe fuer Qscheduled erzeugt hat, auf die tatsaechliche Lagerleistung (QExecuted) um.

- (1) Geben Sie die Kosten pro Stueck an.

- (2) Berechnen Sie anschliessend, wie viel Prozent Sie mehr bezahlt haben (bei einem Vergleich von Kosten QExecuted pro Steuck zu Kosten QScheduled pro Stueck).
Bewertungsrelevant: Output.

```
df1 = aggregate(cbind(QScheduled, QExecuted, cost) ~ Product,
                data = subset(services,
                              Product == "Gruppe105" &
                              service == "Warehousing"),
                sum)
# Hier werden die Kosten pro geplantes Stueck berechnet
df1$ScheduledItem = round(df1$cost/df1$QScheduled, 2)

# Hier werden die Kosten pro tatsaechlich abgefertigtes Stueck berechnet
df1$ExecutedItem = round(df1$cost/df1$QExecuted, 2)

# Hier wird die Differenz der Kosten von ExecutedItem und ScheduledItem in
# Prozent berechnet
df1$Differ = round(df1$ExecutedItem - df1$ScheduledItem, 2)*100

colnames(df1) = c("Product", "Scheduled in total", "Executed in total",
                  "Cost",
                  "Scheduled Item", "Executed Item", "Difference in %")
head(df1)

##      Product Scheduled in total Executed in total      Cost Scheduled Item
## 1 Gruppe105           738250           615608 956469.3           1.3
##      Executed Item Difference in %
## 1           1.55           25

cat("Die reale Ausfuehrung der Warehousing bzw. Lagerdienstleistung kostete
insgesamt", df1[1,3], "€", wobei", df1[1,2], "€ geplant waren. Zudem werden
Kosten pro Stück von", df1[1,5], "€ geplant. Jedoch betrugen die
tatsaechlichen Kosten pro Stück", df1[1,6], "€. Demzufolge ergibt sich eine
prozentuale Differenz von", df1[1,7], "%", die wir mehr bezahlt haben.")

## Die reale Ausfuehrung der Warehousing bzw. Lagerdienstleistung kostete
insgesamt 615608 €, wobei 738250 € geplant waren. Zudem werden Kosten pro
Stück von 1.3 € geplant. Jedoch betrugen die tatsaechlichen Kosten pro Stück
1.55 €. Demzufolge ergibt sich eine prozentuale Differenz von 25 %, die wir
mehr bezahlt haben.
```

- 13) Waehlen Sie eine geeignete Kennzahl zur Bewertung Ihrer Shipping-Dienstleister.
Beachten Sie dabei, was die Qualiteat der Shipping-Dienstleister ausmacht.
Begrunden Sie die Wahl der Kennzahl kurz. Berechnen Sie diese Kennzahl zunaechst
fuer alle Dienstleistungen als zusaetzliche Variable der Services Tabelle, d.h. fuer jede
einzelne Dienstleistung. Berechnen Sie anschliessend die durchschnittliche Kennzahl
der Shipping-Dienstleister fuer die Dienstleistungen an Ihrem Produkt ueber die
gesamte Laufzeit (5 Jahre). Geben Sie Ihre Ergebnisse in einer Tabelle aus, in der die
Kennzahl-Werte aufsteigend sortiert sind. Bewertungsrelevant: Begrueundung, Code,
Output.

#Berechnen Sie OTD für alle Dienstleistungen als zusätzliche Variable der Services Tabelle.

```
services$OTD = services$DaysScheduled >= services$DaysExecuted
```

#ODBDD = Orders Delivered on or Before Due Date

```
shipping_105_vendors_ODBDD <- subset(services, service == "Shipping" &  
                                     Product == "Gruppe105") %>%  
                                     aggregate(OTD ~ vendor, data = ., sum)  
%>%  
                                     setNames(., c("Vendor", "ODBDD"))
```

#Anzahl an Bestellungen pro Lieferant.

```
shipping_105_vendors_total_orders <- subset(services, service == "Shipping" &  
                                             Product == "Gruppe105") %>%  
                                             aggregate(OTD ~ vendor, data = ., length)  
%>%  
                                             setNames(., c("Vendor", "total_orders"))
```

#Beide Dataframes mergen und OTD berechnen.

```
shipping_105_vendors <- merge(x = shipping_105_vendors_ODBDD,  
                              y = shipping_105_vendors_total_orders, by = "Vendor", all =  
TRUE)  
shipping_105_vendors$OTB <- shipping_105_vendors$ODBDD /  
                              shipping_105_vendors$total_orders
```

#Zu Prozent umrechnen

```
shipping_105_vendors$OTB <- (shipping_105_vendors$OTB * 100) %>% round(., 2)
```

#Ausgabe, Kennzahl-Werte aufsteigend sortiert.

```
shipping_105_vendors <- subset(shipping_105_vendors,  
                               select = -c(ODBDD,total_orders))  
shipping_105_vendors[order(-shipping_105_vendors$OTB),]
```

```
##           Vendor  OTB  
## 6  Flying Mercury Shipping 38.69  
## 9           IntEx Shipping 38.62  
## 4             DWL Shipping 37.48  
## 2  Bange+Hammer Shipping 34.44  
## 7           Gifter Shipping 32.37  
## 10            JNT Shipping 31.97  
## 8            HCX Shipping 30.50  
## 1      AHL Express Shipping 29.46  
## 3            CPS Shipping 27.84  
## 5            EPD Shipping 23.37
```

#Clean up

```
rm(shipping_105_vendors_ODBDD, shipping_105_vendors_total_orders,  
   shipping_105_vendors)
```

cat("Begründung zu Aufgabe 13: Schlechte Shipping-DL führen dazu, dass Waren später als erwartet wieder aufgefüllt werden können. Dies führt zu

nicht releasierten Verkaeufen. Da bekannt ist, dass die Shipping-DL die vereinbarte Lieferzeit haeufig ueberschreiten, verwenden wir On-Time Delivery Rate (OTD-Rate) als Kennzahl zur Bewertung. Sie misst die Zuverlaessigkeit, ob Auftraege rechtzeitig geliefert werden. Das Ergebnis gibt in Prozent an wie hoch der Anteil an Lieferungen ist welche rechtzeitig angekommen sind. Z.B. sind 38,69% aller Lieferungen von Flying Mercury Shipping rechtzeitig angekommen. Diese Berechnung erfolgt mit der folgenden Formel: $OTD\text{-}Rate = \frac{\text{count}(\text{DaysScheduled} \geq \text{DaysExecuted})}{\text{count}(\text{Orders})}$)

Begruendung zu Aufgabe 13: Schlechte Shipping-DL fuehren dazu, dass Waren spaeter als erwartet wieder aufgefuellt werden koennen. Dies fuehrt zu nicht releasierten Verkaeufen. Da bekannt ist, dass die Shipping-DL die vereinbarte Lieferzeit haeufig ueberschreiten, verwenden wir On-Time Delivery Rate (OTD-Rate) als Kennzahl zur Bewertung. Sie misst die Zuverlaessigkeit, ob Auftraege rechtzeitig geliefert werden. Das Ergebnis gibt in Prozent an wie hoch der Anteil an Lieferungen ist welche rechtzeitig angekommen sind. Z.B. sind 38,69% aller Lieferungen von Flying Mercury Shipping rechtzeitig angekommen. Diese Berechnung erfolgt mit der folgenden Formel: $OTD\text{-}Rate = \frac{\text{count}(\text{DaysScheduled} \geq \text{DaysExecuted})}{\text{count}(\text{Orders})}$

- 14) Waehlen Sie eine geeignete Kennzahl zur Bewertung Ihrer Warehousing-Dienstleister. Beachten Sie dabei, was die Qualiteat der Warehousing-Dienstleister ausmacht. Begrunden Sie die Wahl der Kennzahl kurz. Berechnen Sie diese Kennzahl zunaechst fuer alle Dienstleistungen als zusaetzliche Variable der Services-Tabelle, d.h. fuer jede einzelne Dienstleistung. Berechnen Sie anschliessend die durchschnittliche Kennzahl fuer die Warehousing-Dienstleister fuer die Dienstleistungen an Ihrem Produkt ueber die gesamte Laufzeit (5 Jahre). Geben Sie Ihre Ergebnisse in einer Tabelle aus, in der die Kennzahl-Werte aufsteigend sortiert sind. Bewertungsrelevant: Begruendung, Code, Output.

#Berechnen Sie diese Kennzahl zunaechst fuer alle Dienstleistungen als zusaetzliche Variable der Services-Tabelle.

```
services$IFR = services$QExecuted / services$QScheduled
```

#IFR aggregiert fuer jeden Warehousing-DL berechnen.

```
warehousing_105_vendors <- subset(services, service == "Warehousing" &
                                Product == "Gruppe105") %>%
  aggregate(IFR ~ vendor, data = ., mean)
```

#Zu Prozent umrechnen.

```
warehousing_105_vendors$IFR <- (warehousing_105_vendors$IFR *
                                100) %>% round(., 2)
```

#Sortieren und ausgeben.

```
warehousing_105_vendors[order(-warehousing_105_vendors$IFR),]
```

```
##           vendor    IFR
## 3      CPS Warehousing 85.65
## 4      DWL Warehousing 85.46
## 9      IntEx Warehousing 84.13
## 6  Flying Mercury Warehousing 84.08
## 1      AHL Express Warehousing 83.34
```

```
## 10      JNT Warehousing 83.18
## 8       HCX Warehousing 82.32
## 7       Gifter Warehousing 82.14
## 2       Bange+Hammer Warehousing 81.95
## 5       EPD Warehousing 81.68
```

#Clean up

```
rm(warehousing_105_vendors)
```

cat("Begründung zu Aufgabe 14: Schlechte Warehousing-Dienstleister führen dazu, dass die Waren früher als erwartet nicht verfügbar sind. Aus der Aufgabenbeschreibung geht hervor, dass die Warehousing-DL häufig falsche Mengen liefern, was aber erst im nachhinein auffällt. Unsere gewählte Kennzahl ist die Item Fill Rate (IFR). Sie untersucht wie genau eine Lieferung war gemessen wurde an der bestellten Menge. So liefert z. B. CPS Warehousing im Durchschnitt 85,65% von der bestellten Menge und hat damit den besten Wert. Diese Berechnung erfolgt mit der folgenden Formel: Item Fill Rate (IFR) = Number of Items delivered to customers / Number of Items ordered by customer")

Begründung zu Aufgabe 14: Schlechte Warehousing-Dienstleister führen dazu, dass die Waren früher als erwartet nicht verfügbar sind. Aus der Aufgabenbeschreibung geht hervor, dass die Warehousing-DL häufig falsche Mengen liefern, was aber erst im nachhinein auffällt. Unsere gewählte Kennzahl ist die Item Fill Rate (IFR). Sie untersucht wie genau eine Lieferung war gemessen wurde an der bestellten Menge. So liefert z. B. CPS Warehousing im Durchschnitt 85,65% von der bestellten Menge und hat damit den besten Wert. Diese Berechnung erfolgt mit der folgenden Formel: Item Fill Rate (IFR) = Number of Items delivered to customers / Number of Items ordered by customer

- 15) Visualisieren Sie in geeigneter Form den Marktanteil (tatsächliche verkaufte Menge) aller Produkte im Markt in einem ggplot. Es bietet sich an, einen aggregierten Datensatz zu nutzen. Bewertungsrelevant: Begründung, Code, Output.

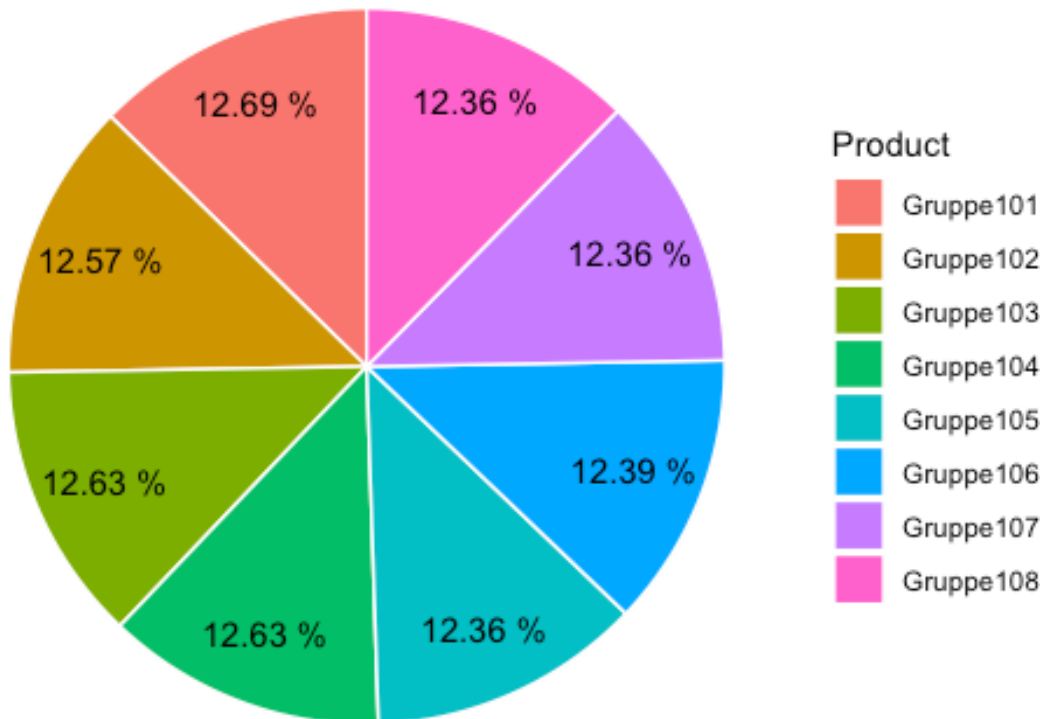
#Entferne Lost Sales, berechne Marktanteil in Prozent und Runde auf zwei Nachkommastellen.

```
marktanteil_tortendiagramm <- transactions[!(transactions$Product ==
                                             "Lost Sales"),]
%>%
  aggregate(Sales ~ Product, data = ., sum) %>%
  mutate(Percent = Sales/sum(Sales)*100)
marktanteil_tortendiagramm$Percent =
  round(marktanteil_tortendiagramm$Percent,
2)
```

#Tortendiagramm zeichnen lassen.

```
ggplot(marktanteil_tortendiagramm, aes(x = "", y = Percent, fill = Product))
+
  geom_bar(width = 1, stat = "identity", color = "white") +
  #Weiße Linie zwischen Feldern
  coord_polar("y", start = 0) +
```

```
geom_text(
  aes(label = paste(Percent, "%"),
    x = 1.3), #x = Distanz zur Kreismitte
  position = position_stack(vjust = 0.5)) +
theme_void()
```



`cat("Begründung zu Aufgabe 15: Wir haben uns fuer einen Tortendiagramm entschieden, da es den Marktanteil auf einfache und verstaendliche Weise darstellt. Zusaetzlich wurde der Marktanteil in Prozent umgerechnet, um die genauen Unterschiede im Nachkommerbereich einfacher zu erfassen. Das Diagramm zeigt deutlich das alle Teilnehmer beinahe gleiche Marktanteile haben. Mit 12,69% ist Gruppe101 Marktführer. Gruppe105, Gruppe107, und Gruppe108 haben mit 12,36% jeweils den niedrigsten Marktanteil.")`

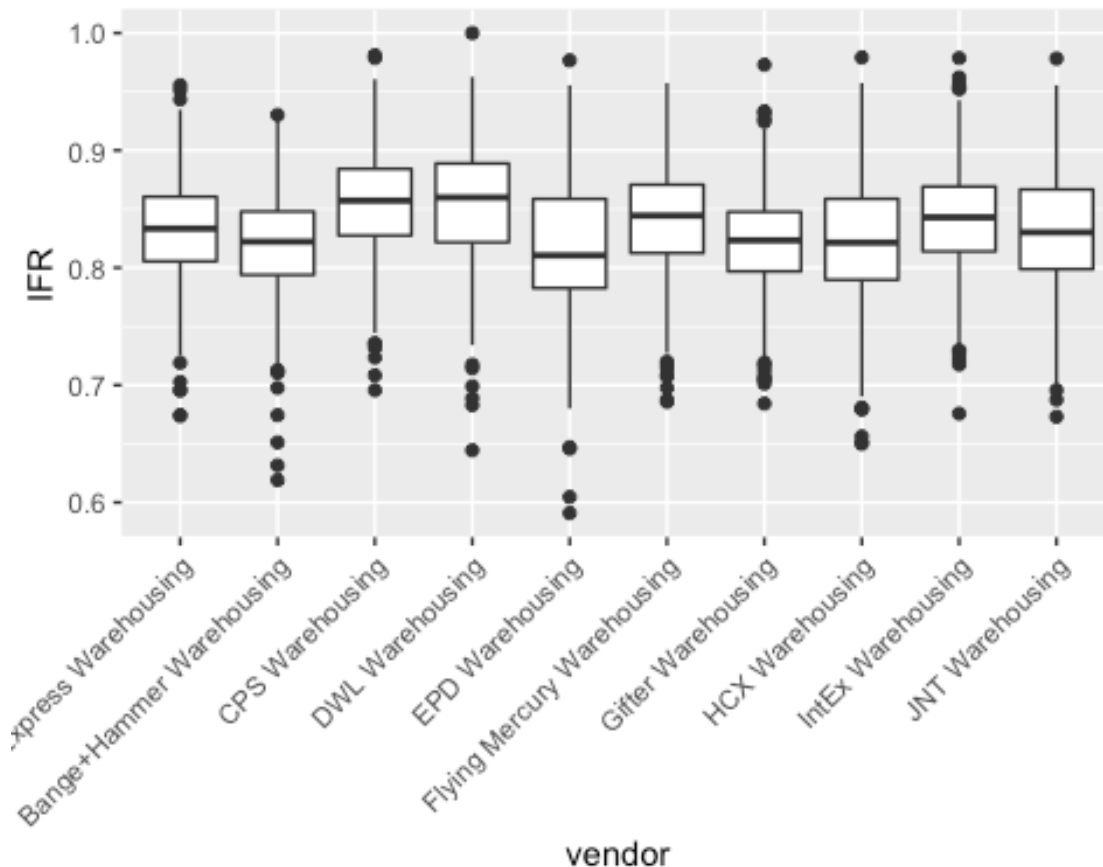
`## Begründung zu Aufgabe 15: Wir haben uns fuer einen Tortendiagramm entschieden, da es den Marktanteil auf einfache und verstaendliche Weise darstellt. Zusaetzlich wurde der Marktanteil in Prozent umgerechnet, um die genauen Unterschiede im Nachkommerbereich einfacher zu erfassen. Das Diagramm zeigt deutlich das alle Teilnehmer beinahe gleiche Marktanteile haben. Mit 12,69% ist Gruppe101 Marktführer. Gruppe105, Gruppe107, und Gruppe108 haben mit 12,36% jeweils den niedrigsten Marktanteil.`

- 16) Visualisieren Sie in geeigneter Form die gewaehlte Qualitaetskennzahl der Warehouse-Dienstleister in einem ggplot. Durch die Visualisierung soll eine Vergleichbarkeit der Dienstleister moeglich sein. Unterstuetzen die abgebildeten Daten

die Entscheidung fuer einen WH-DL? Kommentieren Sie. Bewertungsrelevant:
Begrueundung, Code, Output, Kommentar.

#Boxplot zeichnen lassen für WH-DL und Gruppe105. Unsere gewählt Kennzahl ist IFR.

```
subset(services, service=="Warehousing" & Product=="Gruppe105") %>%  
  ggplot(., aes(x = vendor, y = IFR)) +  
  geom_boxplot() +  
  theme(axis.text.x = element_text(angle=45,hjust=1,vjust=1))
```



#X-Achsenbezeichnung anpassen, damit es lesbar ist.

```
cat("Begrueundung zu Aufgabe 16: Wir entscheiden uns fuer einen Boxplot.  
Dieser gibt uns ein genaueres Bild ueber den Median, Ausreißer und die  
Standardabweichung. Bevorzugt sollte der WH-DL sein, der eine moeglichst  
geringe Standardabweichung hat und gleichzeitig am naechsten bei IFR=1  
liegt.\n\n")
```

```
## Begrueundung zu Aufgabe 16: Wir entscheiden uns fuer einen Boxplot. Dieser  
gibt uns ein genaueres Bild ueber den Median, Ausreißer und die  
Standardabweichung. Bevorzugt sollte der WH-DL sein, der eine moeglichst  
geringe Standardabweichung hat und gleichzeitig am naechsten bei IFR=1 liegt.
```

```
cat("Kommentar: Beim Ablesen des Boxplots faellt uns auf wie klein die  
Unterschiede zwischen den WH-DL sind. DWL Warehousing hat den an IFR=1  
naechsten Median, jedoch hat OPS Warehousing eine geringere
```

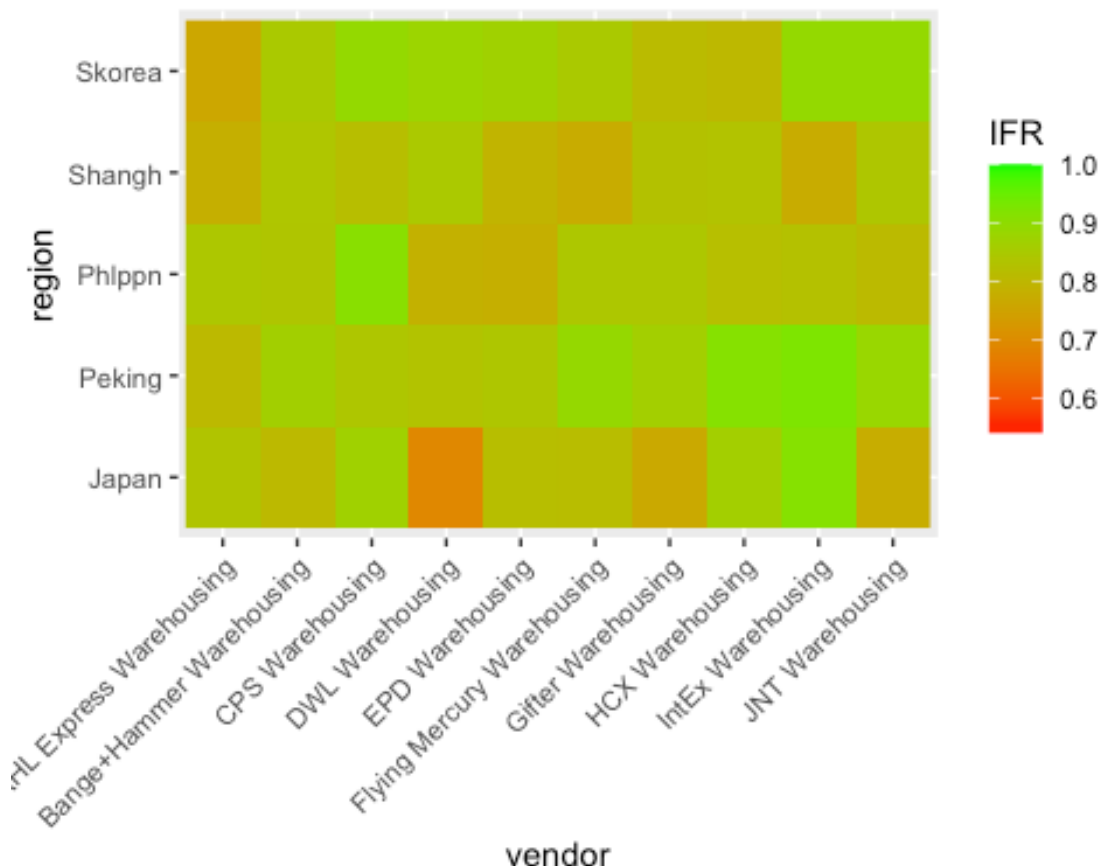
Standardabweichung und liegt fast gleich auf. Eine eindeutige Entscheidung kann nicht getroffen werden.")

Kommentar: Beim Ablesen des Boxplots faellt uns auf wie klein die Unterschiede zwischen den WH-DL sind. DWL Warehousing hat den an IFR=1 naechsten Median, jedoch hat OPS Warehousing eine geringere Standardabweichung und liegt fast gleich auf. Eine eindeutige Entscheidung kann nicht getroffen werden.

- 17) Visualisieren Sie in geeigneter Form die gewaehlte Qualitaetskennzahl der Warehouse-Dienstleister im Vergleich ueber die verschiedenen Regionen in einem ggplot. Wie bewerten Sie die Qualitaet der WH-DL insgesamt? Kommentieren Sie. Bewertungsrelevant: Begrueundung, Code, Output, Kommentar.

#Erzeuge Subset welcher nur Warehousing-DL enthält.

```
subset(services, service=="Warehousing") %>% ggplot(.,  
  aes(x = vendor, y = region, fill = IFR)) +  
  geom_raster() +  
  scale_fill_gradient(low = "red", high="green") +  
  theme(axis.text.x = element_text(angle=45,hjust=1,vjust=1))
```



cat("Begrueundung zu Aufgabe 17: Wir entscheiden uns fuer eine Heat-Map, wobei die Farbe gruen besser und rot schlechter darstellt. Da es bei unseren Daten keine zu groeuen Lieferungen gab, eignet sich dieses Verfahren. Falls es

übermäßig große Lieferungen gegeben hätte, müssten die Daten vorher aufbereitet werden.\n\n")

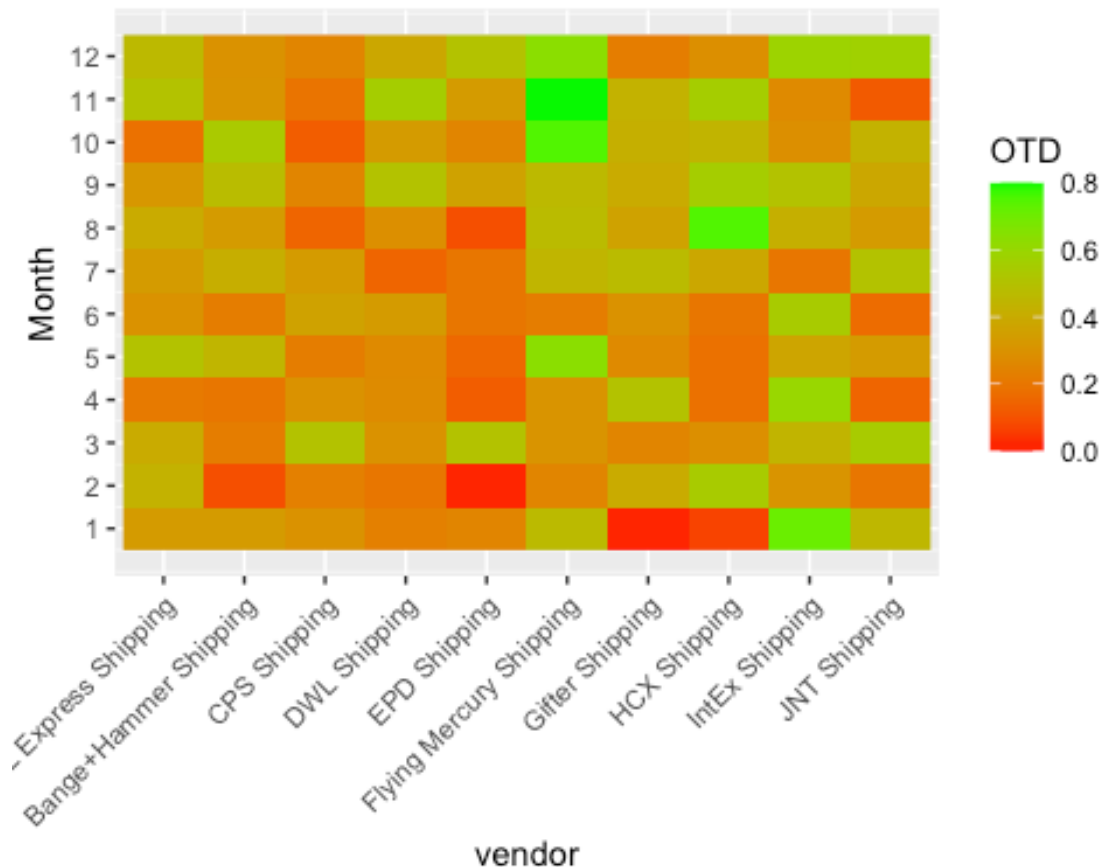
Begründung zu Aufgabe 17: Wir entscheiden uns für eine Heat-Map, wobei die Farbe grün besser und rot schlechter darstellt. Da es bei unseren Daten keine zu großen Lieferungen gab, eignet sich dieses Verfahren. Falls es übermäßig große Lieferungen gegeben hätte, müssten die Daten vorher aufbereitet werden.

cat("Kommentar: Es fällt auf das DWL Warehousing besonders schlecht in Japan liefert, dort sollte man auf IntEx Warehousing umgesteigen. Generell lässt die Heat-Map leicht ablesen, welcher WH-DL in welcher Region gut/schlecht liefern, gemessen an IFR.")

Kommentar: Es fällt auf das DWL Warehousing besonders schlecht in Japan liefert, dort sollte man auf IntEx Warehousing umgesteigen. Generell lässt die Heat-Map leicht ablesen, welcher WH-DL in welcher Region gut/schlecht liefern, gemessen an IFR.

18) Visualisieren Sie in geeigneter Form die Qualitätskennzahl der Shipping-Dienstleister je Monat für das Jahr 2017 in einem ggplot (über alle Regionen zusammen) Wie bewerten Sie die Qualität der Shipping-DL insgesamt? Kommentieren Sie. Bewertungsrelevant: Begründung, Code, Output, Kommentar.

```
```r
#Heat-Map zeichnen lassen für Shipping-DL, Jahr=2017 und unser Produkt
Gruppe105.
subset(services, service=="Shipping" & Year=="2017" & Product=="Gruppe105")
%>%
 aggregate(OTD ~vendor+Month, data=., mean) %>%
 ggplot(., aes(x = vendor, y = Month, fill = OTD)) +
 geom_raster() +
 scale_fill_gradient(low = "red", high="green") +
 theme(axis.text.x = element_text(angle=45,hjust=1,vjust=1)) +
 #Y-Achsen Bezeichnung ändern
 scale_y_continuous(breaks=seq(1,12,1))
```



`cat("Begründung zu Aufgabe 18: Wir wählen hier eine Heat-Map die durch unterschiedliche Farbschattierungen und einer klaren Struktur einen schnellen Überblick gibt. Auch in diesem Fall ist grün besser als rot. Ein Liniendiagramm eignet sich nicht, da es zu viele Linien enthält, die durcheinander gehen und somit schwer zu lesen sind.\n\n")`

`## Begründung zu Aufgabe 18: Wir wählen hier eine Heat-Map die durch unterschiedliche Farbschattierungen und einer klaren Struktur einen schnellen Überblick gibt. Auch in diesem Fall ist grün besser als rot. Ein Liniendiagramm eignet sich nicht, da es zu viele Linien enthält, die durcheinander gehen und somit schwer zu lesen sind.`

`cat("Kommentar: Das Diagramm zeigt, dass alle Lieferanten mal gut und mal schlecht abschneiden. Besonders schlechte OTD-Werte eines Lieferanten stechen sofort durch ein kräftiges rot ins Auge, während besonders gute OTD-Werte durch ein helles grün auffallen. Dank der Struktur des Diagramms kann man schnell Handlungsempfehlungen ableiten. So sollte man z. B. im Februar EPD-Shipping meiden und stattdessen auf HCX-Shipping setzen. Es würde Sinn machen nicht nur das Jahr 2017, sondern alle 5 Jahre zu betrachten, um auf eine fundiertere Lösung zukommen. Beim lesen sollte auf die Legende geachtet werden, beispielsweise ist 0.8 das Maximum, da es der höchste erreichbare Wert ist. Das Optimum wäre jedoch der Wert 1.")`

`## Kommentar: Das Diagramm zeigt, dass alle Lieferanten mal gut und mal schlecht abschneiden. Besonders schlechte OTD-Werte eines Lieferanten stechen`



sofort durch ein kraeftiges rot ins Auge, waehrend besonders gute OTD-Werte durch ein helles gruen auffallen. Dank der Struktur des Diagramms kann man schnell Handlungsempfehlungen ableiten. So sollte man z. B. im Februar EPD-Shipping meiden und stattdessen auf HCX-Shipping setzen. Es wuerde Sinn machen nicht nur das Jahr 2017, sondern alle 5 Jahre zu betrachten, um auf eine fundiertere Loesung zukommen. Beim lesen sollte auf die Legende geachtet werden, beispielsweise ist 0.8 das Maximum, da es der hoechste erreichbare Wert ist. Das Optimum waere jedoch der Wert 1.

**Aufgabe 2.1.) Projektbeschreibung + Stellungnahme 2.2:** Das Unternehmen X setzt sich langfristig das Ziel wettbewerbsfähiger zu werden, um ihre aktuelle Marktposition im asiatischen Raum zu gewährleisten und ihr Marktpotenzial auszuschöpfen. Insbesondere widmet sich das Unternehmen X unterschiedlichen logistischen Fragestellungen, wie die Waren besser in den Markt und an die Kunden gelangen. Im Zuge dessen hat die Data Analytics Abteilung das Projekt AnalyticsX ins Leben gerufen, um sich mit diesen Fragestellungen zu befassen und um einen Mehrwert für das Unternehmen zu schaffen. Im Projekt AnalyticsX wird mittels Data Analytics Methoden Handlungsempfehlungen ausgearbeitet, die der Geschäftsführung helfen sollen, faktenbasierte Entscheidungen zu treffen und Handlungen einzuleiten sowie durchzuführen. Die Data Analytics Abteilung untergliedert ihren Projektablauf in die folgenden sechs Schritte:

**1. Identifikation des Geschäftsproblems:** Zuerst wird ein Forschungsziel gesetzt, die ein Geschäftsproblem adressieren soll, um den Umfang des AnalyticsX Projektes zu bestimmen. Da das Geschäftsproblem bereits vom Geschäftsführer identifiziert wurde, setzt sich die Data Analytics Abteilung daran die nächsten Schritte für das AnalyticsX Projekts zu konkretisieren.

**2. Datenbeschaffung und Datenquellen:** Im zweiten Schritt erfolgt eine Datenbeschaffung durch die Anbindung an verschiedenen Datenbanken, Data Warehouses, Data Marts und Data Lakes im Unternehmen, um relevante Daten zu extrahieren. Dabei sollte darauf geachtet werden, dass es sich bei der Extrahierung um strukturierte Daten in Form von CSV-Dateien handelt. Da dies jedoch nicht immer gewährleistet werden kann, können auch unstrukturierte Daten bezogen werden. Hier ist es abzurufen Daten in Ordern von lokalen Servern sowie Excel Dateien von Stakeholdern anzunehmen, da diese nicht auf Korrektheit überprüft werden können.

**3. Datenvorbereitung und Datenexploration:** Im dritten Schritt bietet es sich für die Modellierung der Daten an, dass strukturierte Daten bereits in Form von CSV-Dateien vorliegen. Falls dies nicht der Fall ist, sollten die Daten zuerst in ein gemeinsames Format überführt werden. Diese Daten werden an das RStudio Environment verknüpft. Dies setzt voraus, dass das AnalyticsX Projekt-Team bereits R und RStudio auf ihren Rechnern installiert haben. Anschließend wird eine Datenexploration durchgeführt, um eine grobe Übersicht über die vorliegenden Daten zu verschaffen. Dies hat den Zweck ein besseres Verständnis über die Daten zu gewinnen.

**4. Modellierung und Ausführung der Analyse:** Für die Modellierung verwenden wir vergangene Daten, um die aktuelle Unternehmenssituation zu beschreiben sowie Vorhersagen abzuleiten, um dann zukünftige Handlungsempfehlungen auszuarbeiten. Die Daten werden gemeinsam interpretiert und ihre Ergebnisse validiert. Wir erwarten zudem durch die Zusammenarbeit mit verschiedenen Abteilungen Rückmeldungen über fehlende Features und Verbesserungsvorschläge, die wir bei unserer Modellierung und Analysen miteinbeziehen. Dieser Prozess erfolgt iterativ, wobei Optimierungen bei der Daten Modellierung und Analysen immer wieder vorgenommen werden.

**5. Ergebnisbewertung und Ergebniskommunikation:** Die Bewertung der Ergebnisse erfolgt durch die Entnahme von Stichproben aus den Daten, die im Detail ausgewertet werden, um Fehler oder Unregelmäßigkeiten zu identifizieren. Zudem sollten die

Ergebnisse kritisch hinterfragt werden, ob das Modell die Geschäftsziele erreicht und ob die Daten genügend aussagekraft haben. Es sollten qualitative und quantitative Bewertungen vorgenommen werden, um eine sichere Aussage treffen zu können. Zum Schluss werden die Ergebnisse an wichtigen Entscheidungsträgern wie beispielsweise dem Geschäftsführer kommuniziert. Hierbei können die Ergebnisse zusammengefasst und visualisiert werden, um die Vermittlung der Ergebnisse zu vereinfachen, die aber gleichzeitig genügend aussagekraft enthalten muss. Für weitere Verbesserungen werden mehrere Feedback-Gespräche mit Geschäftsführung und verschiedenen Abteilungen benötigt. Je nach Bedarf und Ressourcen kann die Vorangehensweise und Ergebnisse durch Seminare, Trainings und Einweisungen vermittelt werden.

**6. Bereitstellung und Nutzengenerierung:** Neben der Bereitstellung der Ergebnisse des AnalyticsX Projektes sollen zudem Code und technische Dokumentation ausreichend bereitgestellt werden, die für zukünftige Wartungsarbeiten und Weiterentwicklungen notwendig sind. Das Ergebnismodell wird in der Produktion zur Verfügung gestellt, wobei es wichtig ist, dass die Daten geschützt werden, um Manipulationen und Löschungen zu verhindern. Die Ergebnisse sollen zentral abgespeichert werden und Veränderungen an Daten sollen dokumentiert werden. Die Ergebnisse sind insbesondere für die Geschäftsführung vorgesehen, da sie anhand der umfangreichen Analyse Entscheidungen und Handlungen treffen können, die dem Unternehmen X dabei helfen soll sich auf dem asiatischen Markt besser zu positionieren. Zudem können bestimmte Abteilungen Zugang zu dem Modell erhalten, die dieses Modell für ihre Analysen gebrauchen können, um einen Mehrwert für das Unternehmen zu schaffen.

**Stellungnahme zu Aufgabe 1.1:** Die verkaufte Menge (Sales) des Produkts Gruppe105 in den asiatischen Regionen hat gezeigt, dass Peking mit 8,19% im Vergleich zu den anderen Regionen am umsatzschwächsten ist und Shanghai mit 14,47% am umsatzstärksten. Anhand des Datenmodells erkennen wir, dass ein Optimierungspotenzial in der Logistik in Peking besteht. Der Logistikdienstleister (LDL) „Bange+Hammer Shipping“ verzeichnet im Vergleich zu anderen LDL die höchsten Kosten für verspätete Transportdienstleistungen. Vorschläge zu Vertragsänderungen mit diesem LDL könnten vorgenommen werden, um verspätete Services und damit Kosten zu reduzieren. Zur Bewertung der Shipping-DLs wurde die On-Time Delivery (OTD) verwendet. Im Durchschnitt der letzten 5 Jahre kommt „Flying Mercury Shipping“ auf eine OTD-Rate von 38,69 %, was bedeutet, dass ca. 39 % der Lieferungen puenktlich ankommen. „IntEx Shipping“ (38,62) und „DWL Shipping“ (37,48) sind nur unwesentlich schlechter. In Zukunft sollten diese 3 Shipping-DL bevorzugt behandelt werden. Vor allem sollte „EPD Shipping“ mit einer OTD von 23,37 % gemieden werden, da dies deutlich schlechter ist als alle anderen Anbieter. Die Warehousing-DLs wurden mithilfe der Item-Fill-Rate (IFR) bewertet. Sie untersucht wie genau eine Lieferung war, gemessen an der bestellten Menge. So ist im Durchschnitt „CPS Warehousing“ mit 85,65 % der beste Warehousing-DL, was bedeutet das bei diesem Anbieter 85,65 % der Bestellenmenge tatsächlich ankommt. Durch eine Heat-Map wird offensichtlich, dass die Warehousing-DLs in unterschiedlichen Regionen unterschiedlich gute Arbeit leisten. In den Phillipinen ist „OPS Warehousing“ die beste Wahl ist, in Japan ist „IntEx Warehousing“ und in Peking sind „IntEx Warehousing“ oder „HCX Warehousing“ zu empfehlen. Da bei der Bewertung der Warehousing-DL 5 Jahre berücksichtigt wurden, können die Ergebnisse auch für konkrete Handlungen verwendet werden.