

Gruppe 105 Hausaufgabe 3

Supply Chain Analytics SS23

Cordelia Mena Hernandez, Daniel Glatter

2023-06-07

Daten für die Modellierung vorbereiten

Aufgabe 1

```
# Laden der Daten in die Dataframes 'services' und 'externals'
externals <- read.csv("externals20.csv", sep=";", dec=",")
services <- read.csv("../HA1/data/output_services_8Players_v0020.csv", sep=";", dec=",")

# Tabelle Externals aufbereiten
# Characters in Factors umwandeln
externals$region <- as.factor(externals$region)

# Periode anpassen
externals$Period <- as.yearmon(make_date(year=externals$Year, month = externals$Month))
externals <- subset(externals, select=-c(Year, Month))

# Tabelle Services aufbereiten
# Umwandlung ins Datumsformat
services$Date <- make_date(services$Year, services$Month, services$Day)
services <- subset(services, select=-c(Year, Month, Day))

# Characters in Factors umwandeln
services$region = as.factor(services$region)
services$storename = as.factor(services$storename)
services$Product = as.factor(services$Product)
services$vendor = as.factor(services$vendor)
services$service = as.factor(services$service)

# Periode hinzufügen
services$Period <- as.yearmon(services$Date)

# OTD, IFR berechnen
services$IFR <- services$QExecuted / services$QScheduled
services$OTD <- services$DaysExecuted <= services$DaysScheduled # true/false

# LDL nach OTD aggregieren
ldl_otd <- services %>%
  group_by(Logistikdienstleister=vendor, service) %>%
  summarize(OTD = mean(OTD)) %>%
  arrange(OTD)
```

```
ldl_otd %>% kable(caption="Logistikdienstleister nach On-Time Delivery Rate", digits=3)
↪ %>% kable_styling(latex_options = "hold_position")
```

Table 1: Logistikdienstleister nach On-Time Delivery Rate

Logistikdienstleister	service	OTD
JNT Shipping	Shipping	0.277
AHL Express Shipping	Shipping	0.295
CPS Shipping	Shipping	0.298
HCX Shipping	Shipping	0.330
IntEx Shipping	Shipping	0.334
Gifter Shipping	Shipping	0.356
Flying Mercury Shipping	Shipping	0.357
DWL Shipping	Shipping	0.379
EPD Shipping	Shipping	0.380
Bange+Hammer Shipping	Shipping	0.431
AHL Express Warehousing	Warehousing	1.000
Bange+Hammer Warehousing	Warehousing	1.000
CPS Warehousing	Warehousing	1.000
DWL Warehousing	Warehousing	1.000
EPD Warehousing	Warehousing	1.000
Flying Mercury Warehousing	Warehousing	1.000
Gifter Warehousing	Warehousing	1.000
HCX Warehousing	Warehousing	1.000
IntEx Warehousing	Warehousing	1.000
JNT Warehousing	Warehousing	1.000

```
# Tabelle nach IFR aggregieren
ldl_ifr <- services %>%
  group_by(Logistikdienstleister=vendor, service) %>%
  summarize(IFR = mean(IFR)) %>%
  arrange(IFR)

ldl_ifr %>% kable(caption="Logistikdienstleister nach In-Full Rate", digits=3) %>%
  ↪ kable_styling(latex_options = "hold_position")
```

Table 2: Logistikdienstleister nach In-Full Rate

Logistikdienstleister	service	IFR
IntEx Warehousing	Warehousing	0.819
EPD Warehousing	Warehousing	0.819
Bange+Hammer Warehousing	Warehousing	0.824
CPS Warehousing	Warehousing	0.825
DWL Warehousing	Warehousing	0.825
AHL Express Warehousing	Warehousing	0.825
HCX Warehousing	Warehousing	0.829
JNT Warehousing	Warehousing	0.830
Gifter Warehousing	Warehousing	0.837
Flying Mercury Warehousing	Warehousing	0.837
AHL Express Shipping	Shipping	1.000
Bange+Hammer Shipping	Shipping	1.000
CPS Shipping	Shipping	1.000
DWL Shipping	Shipping	1.000
EPD Shipping	Shipping	1.000
Flying Mercury Shipping	Shipping	1.000
Gifter Shipping	Shipping	1.000
HCX Shipping	Shipping	1.000
IntEx Shipping	Shipping	1.000
JNT Shipping	Shipping	1.000

Aufgabe 2

```
# Daten nach Monat zusammenfassen
ldl_ifr_warehousing <- services %>%
  filter(service == "Warehousing") %>%
  group_by(Logistikdienstleister=vendor, region, Period) %>%
  summarize(IFR = mean(IFR)) %>%
  arrange(IFR)

head(ldl_ifr_warehousing) %>% kable(caption='In-Full Rate der Warehousing-LDL nach Region
  ↪ und Periode', digits=3) %>% kable_styling(latex_options = "hold_position")
```

Table 3: In-Full Rate der Warehousing-LDL nach Region und Periode

Logistikdienstleister	region	Period	IFR
IntEx Warehousing	Shangh	Dec 2018	0.750
IntEx Warehousing	Skorea	Aug 2018	0.759
IntEx Warehousing	Shangh	Jan 2018	0.761
AHL Express Warehousing	Shangh	Aug 2022	0.763
AHL Express Warehousing	Shangh	Oct 2022	0.764
IntEx Warehousing	Shangh	Nov 2021	0.765

```
# Insgesamt bester Warehousing-LDL
ldl_ifr_warehousing %>%
  group_by(Logistikdienstleister) %>%
  summarize(IFR=mean(IFR)) %>%
  arrange(desc(IFR)) %>%
  head(1) %>%
  kable(caption="Bester LDL nach IFR, über Regionen und Perioden hinweg", digits=3) %>%
  ↪ kable_styling(latex_options = "hold_position")
```

Table 4: Bester LDL nach IFR, über Regionen und Perioden hinweg

Logistikdienstleister	IFR
Flying Mercury Warehousing	0.837

Als besten Warehousing-LDL betrachten wir hier den LDL mit der höchsten durchschnittlichen IFR über alle Regionen und Perioden hinweg. Das ist Flying Mercury Warehousing mit einer durchschnittlichen IFR von 83,7%.

```
# Schlechtester IFR-Wert (und Periode) dieses LDL
ldl_ifr_warehousing %>%
  filter(Logistikdienstleister == "Flying Mercury Warehousing") %>%
  filter(region == "Shangh") %>%
  arrange(IFR) %>%
  head(1) %>%
  kable(caption="Schlechtester IFR-Wert und Periode von Flying Mercury Warehousing in der
  ↪ Region Shanghai", digits=3) %>% kable_styling(latex_options = "hold_position")
```

Table 5: Schlechtester IFR-Wert und Periode von Flying Mercury Warehousing in der Region Shanghai

Logistikdienstleister	region	Period	IFR
Flying Mercury Warehousing	Shangh	May 2022	0.784

Der schlechteste IFR-Wert von Flying Mercury Warehousing in der Region Shanghai beträgt 78,4% und war im Mai 2022.

Aufgabe 3

```
# Daten nach Monat zusammenfassen
ldl_ifr_shipping <- services %>%
  filter(service == "Shipping") %>%
```

```

group_by(Logistikdienstleister=vendor, region, Period=Period) %>%
summarize(OTD = mean(OTD)) %>%
# Sortieren nach OTD Rate
arrange(OTD)

head(ldl_ifr_shipping) %>% kable(caption='On-Time Delivery Rate der Shipping-LDL nach
↪ Region und Periode', digits=3) %>% kable_styling(latex_options = "hold_position")

```

Table 6: On-Time Delivery Rate der Shipping-LDL nach Region und Periode

Logistikdienstleister	region	Period	OTD
AHL Express Shipping	Japan	Aug 2019	0
AHL Express Shipping	Japan	Jan 2020	0
AHL Express Shipping	Japan	Apr 2022	0
AHL Express Shipping	Shangh	Jun 2019	0
CPS Shipping	Japan	Mar 2018	0
CPS Shipping	Peking	Apr 2021	0

```

# Schlechtester OTD-Wert in Japan im April 2019
ldl_ifr_shipping %>%
  filter(Period == "Apr 2019") %>%
  filter(region == "Japan") %>%
  arrange(OTD) %>%
  # Wir können den ersten Eintrag nehmen, weil das Dataframe bereits sortiert ist
  head(1) %>%
  kable(caption="Niedrigster OTD-Wert und entsprechender LDL in der Region Japan im April
↪ 2019", digits=3) %>% kable_styling(latex_options = "hold_position")

```

Table 7: Niedrigster OTD-Wert und entsprechender LDL in der Region Japan im April 2019

Logistikdienstleister	region	Period	OTD
CPS Shipping	Japan	Apr 2019	0.071

Der schlechteste OTD-Wert eines Shipping-LDL in der Region Japan im April 2019 beträgt 7.1%. Betroffen ist der Dienstleister CPS Shipping.

Modellierung: Warehousing

Aufgabe 4

```

cps_warehousing <- merge(subset(ldl_ifr_warehousing, Logistikdienstleister == "CPS
↪ Warehousing"),
                        externals,
                        by=c("Period", "region"))

# Index X aus Externals-Teil löschen
cps_warehousing <- subset(cps_warehousing, select=-c(X))

outputable <- head(cps_warehousing)

list(outputable[,1:7],outputable[,8:13], outputable[,14:19], outputable[,20:24],
↪ outputable[,25:26]) %>% kable(caption="IFR Rate sowie externe Einflussfaktoren von
↪ CPS Warehousing nach Region und Periode" )%>% kable_styling(latex_options =
↪ "hold_position")

```

Table 8: IFR Rate sowie externe Einflussfaktoren von CPS Warehousing nach Region und Periode

Period	region	Logistikdienstleister	IFR	Temperature_C	Rain_mm	Sunshine_h
Apr 2018	Japan	CPS Warehousing	0.8363984	9.020	42.559	153.421
Apr 2018	Peking	CPS Warehousing	0.8280104	5.633	36.033	166.449
Apr 2018	Phlppn	CPS Warehousing	0.8076996	14.748	21.673	243.125
Apr 2018	Shangh	CPS Warehousing	0.8091017	12.880	68.116	241.760
Apr 2018	Skorea	CPS Warehousing	0.8842718	3.731	54.879	182.586
Apr 2019	Japan	CPS Warehousing	0.8416323	9.890	37.908	154.331
Humidity	Congestion	InternetStability	PowerGridStability	ParkingSpaceAvailability	RoadCondition	
61.031	54.16	1589	0.27	0.0327874	5.94	
79.192	29.23	1733	1.66	0.0594874	3.97	
75.300	76.68	637	2.81	0.0226809	2.50	
73.540	59.30	323	1.69	0.0161623	5.94	
69.773	44.61	62	3.71	0.0337854	2.42	
57.897	44.43	1676	0.27	0.0300179	5.70	
PoliticalStability	AvgHealth	Criminality	AirPollution	WaterQuality	leisureAndSocialInteractions	
1.76	79.67	27.83	232.18	0.94	6.49	
1.48	82.52	17.93	10.51	0.98	7.20	
2.77	80.30	23.00	59.91	0.80	6.00	
2.43	78.45	35.67	61.85	0.88	6.08	
4.98	53.84	50.44	57.27	0.81	7.07	
2.57	76.47	27.76	246.98	0.92	6.50	
SkilledLaborAvailability		UnskilledLaborAvailability		WorkerMotivation	Overtime	Inflation
71.00		49.38		6.80	0.214	0.0011542
37.56		80.51		6.54	0.062	-0.0008377
36.96		60.52		6.22	0.000	0.0004873
48.50		60.31		5.98	0.276	0.0014327
69.45		20.45		6.82	0.205	0.0132490
69.09		51.28		7.47	0.000	0.0010815
			BusinessConfidence	FuelPrice		
			101.0724	2.442		
			99.2389	2.372		
			100.5846	2.593		
			100.9373	2.662		
			101.6482	2.368		
			100.4598	2.354		

Aufgabe 5

a) Externe Effekte und Korrelation zur IFR

```
correlations_IFR <- cor(cps_warehousing$IFR, cps_warehousing[, 4:ncol(cps_warehousing)])

# Besser lesbarer Output und Sortierung nach Stärke der Korrelation
correlations_IFR <- data.frame(IFR=t(correlations_IFR)) %>%
  arrange(desc(abs(IFR)))

# Ausgabe
correlations_IFR %>%
  kable(caption="Korrelation aller Variablen zur In-Full Rate", digits=4) %>%
  ↪ kable_styling(latex_options = "hold_position")
```

Table 9: Korrelation aller Variablen zur In-Full Rate

	IFR
IFR	1.0000
SkilledLaborAvailability	0.6799
UnskilledLaborAvailability	-0.6500
Criminality	0.4634
Inflation	0.4588
AirPollution	0.4501
AvgHealth	-0.3913
WorkerMotivation	0.3792
Sunshine_h	-0.3612
Temperature_C	-0.3540
BusinessConfidence	0.3523
PoliticalStability	0.3032
Overtime	-0.1976
Congestion	-0.1970
ParkingSpaceAvailability	0.1505
Rain_mm	-0.1107
leisureAndSocialInteractions	-0.1020
FuelPrice	-0.1012
RoadCondition	-0.0837
Humidity	0.0783
PowerGridStability	-0.0673
WaterQuality	0.0603
InternetStability	0.0446

b) Die 5 am stärksten korrelierenden Effekte und ihre Korrelationen

```
# 5 stärkste Effekte (neben IFR selbst); 'correlations_IFR' ist schon nach Korrelation
↪ sortiert
strongest_five_effects <- rownames(head(correlations_IFR, 6))

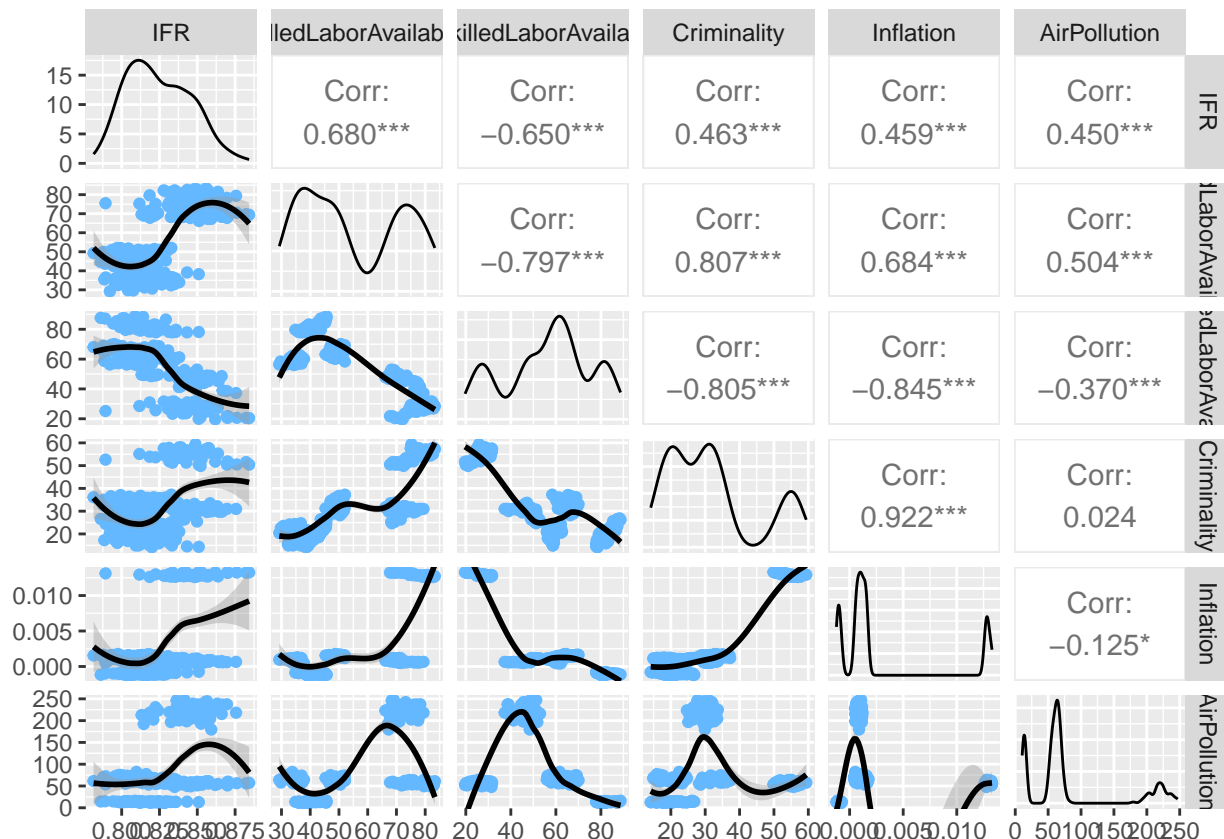
cor(subset(cps_warehousing, select=strongest_five_effects)) %>%
  kable(caption="IFR und die 5 stärksten Effekte darauf", digits=4) %>%
  kable_styling(latex_options=c("scale_down", "hold_position"))
```

Table 10: IFR und die 5 stärksten Effekte darauf

	IFR	SkilledLaborAvailability	UnskilledLaborAvailability	Criminality	Inflation	AirPollution
IFR	1.0000	0.6799	-0.6500	0.4634	0.4588	0.4501
SkilledLaborAvailability	0.6799	1.0000	-0.7969	0.8071	0.6844	0.5035
UnskilledLaborAvailability	-0.6500	-0.7969	1.0000	-0.8053	-0.8455	-0.3697
Criminality	0.4634	0.8071	-0.8053	1.0000	0.9222	0.0240
Inflation	0.4588	0.6844	-0.8455	0.9222	1.0000	-0.1252
AirPollution	0.4501	0.5035	-0.3697	0.0240	-0.1252	1.0000

c) Korrelations-Plot für die 5 stärksten Effekte

```
ggpairs(subset(cps_warehousing, select=strongest_five_effects),
  progress = FALSE,
  lower = list(continuous = wrap("smooth_loess", colour = "steelblue1")))
```



Aufgabe 6

```
cps_warehousing$Baseline <- mean(cps_warehousing$IFR)

head(cps_warehousing[, c("Period", "region", "IFR", "Baseline")]) %>%
  kable(caption="Tabellenkopf für LDL CPS Warehousing mit Baseline", digits=4) %>%
  ↪ kable_styling(latex_options = "hold_position")
```

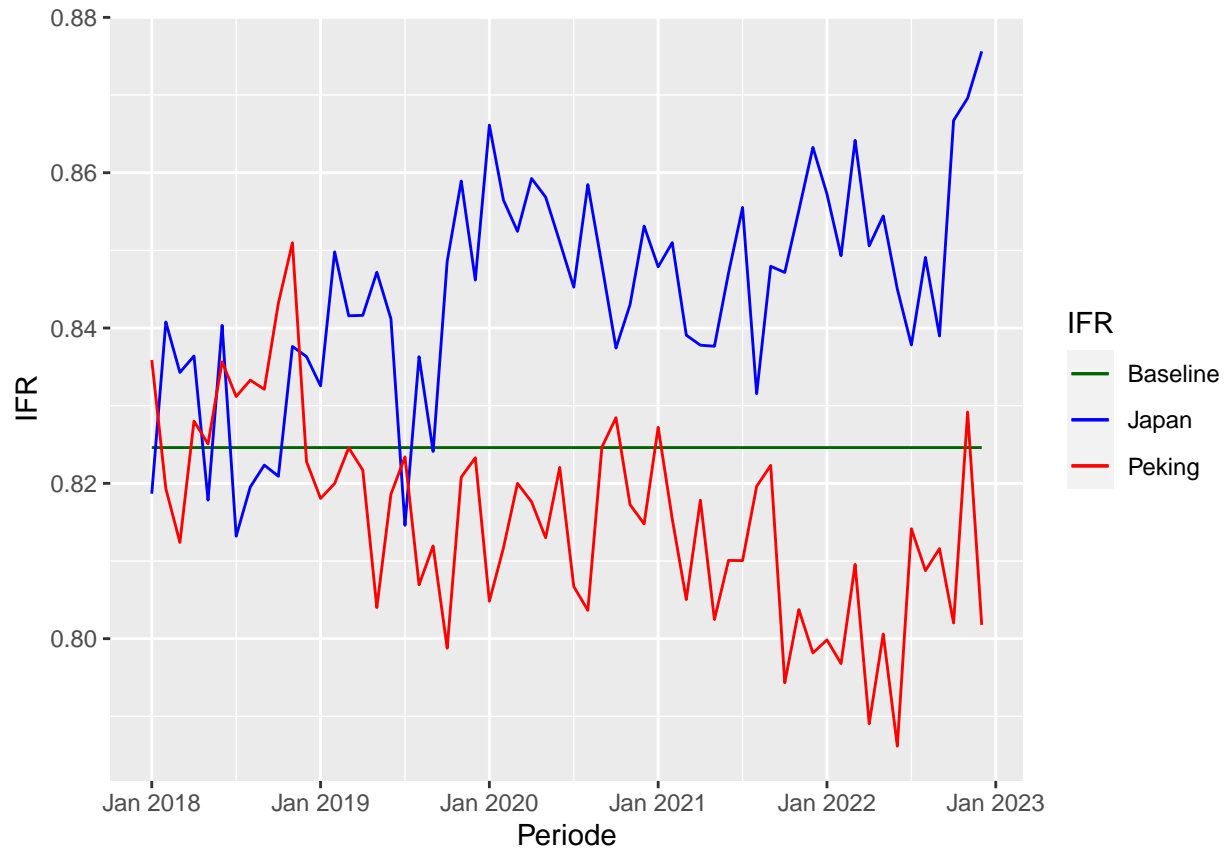
Table 11: Tabellenkopf für LDL CPS Warehousing mit Baseline

Period	region	IFR	Baseline
Apr 2018	Japan	0.8364	0.8246
Apr 2018	Peking	0.8280	0.8246
Apr 2018	Phlppn	0.8077	0.8246
Apr 2018	Shangh	0.8091	0.8246
Apr 2018	Skorea	0.8843	0.8246
Apr 2019	Japan	0.8416	0.8246

Eine sehr einfache Baseline stellt der Mittelwert der vorherzusagenden Variable (IFR) dar. Dieser ist leicht zu berechnen und liegt per Definition “im Schnitt richtig”. Natürlich kann der Mittelwert allerdings keine Nuancen basierend auf den anderen verfügbaren externen Informationen aufgreifen.

Aufgabe 7

```
ggplot() +
  # Baseline
  geom_line(data=filter(cps_warehousing, year(Period) <= 2022), aes(x=Period, y=Baseline,
  ↪ color="Baseline")) +
  # IFR Japan
  geom_line(data = filter(cps_warehousing, year(Period) <= 2022 & region == "Japan"),
  ↪ aes(x=Period, y=IFR, color="Japan")) +
  # IFR Peking
  geom_line(data = filter(cps_warehousing, year(Period) <= 2022 & region == "Peking"),
  ↪ aes(x=Period, y=IFR, color="Peking")) +
  xlab("Periode") +
  ylab("IFR") +
  scale_color_manual(name='IFR',
    breaks=c('Baseline', 'Japan', 'Peking'),
    values=c('Baseline'='darkgreen', 'Japan'='blue', 'Peking'='red'))
```



Aufgabe 8

```
# Data Frame 'Evaluation' erstellen
evaluation <- data.frame(Model = "Baseline",
  MAE = numeric(1),
  MAPE = numeric(1),
  Rsquared = numeric(1),
  Rsquared_adj = numeric(1))

# MAE berechnen
evaluation[evaluation$Model == "Baseline",]$MAE = mean(abs(cps_warehousing$IFR -
  ↪ cps_warehousing$Baseline))

# MAPE berechnen (SMAPE nicht benötigt, weil IFR immer >0)
evaluation[evaluation$Model == "Baseline",]$MAPE =
  ↪ mean(abs((as.numeric(cps_warehousing$IFR -
  ↪ cps_warehousing$Baseline)/as.numeric(cps_warehousing$IFR))*100))

# R² berechnen
evaluation[evaluation$Model == "Baseline",]$Rsquared = NA

# Adjusted R² berechnen
evaluation[evaluation$Model == "Baseline",]$Rsquared_adj = NA

# Tabelle anzeigen
```

```
evaluation %>% kable(caption="Bewertung der Baseline nach MAE und MAPE", digits=3) %>%
  ↪ kable_styling(latex_options = "hold_position")
```

Table 12: Bewertung der Baseline nach MAE und MAPE

Model	MAE	MAPE	Rsquared	Rsquared_adj
Baseline	0.018	2.152	NA	NA

Aufgabe 9

```
# Seed setzen
set.seed(4141)

# 80% der Daten für Trainingsset sampeln
training_sample_indices <- sample(1:nrow(cps_warehousing), nrow(cps_warehousing) * 0.8)

# Trainingsset
training_data <- cps_warehousing[training_sample_indices, ]
tr_datahead<- head(training_data)
list(tr_datahead[,1:7],tr_datahead[,8:13], tr_datahead[,14:19], tr_datahead[,20:24],
  ↪ tr_datahead[,25:27]) %>% kable(caption="Trainingsdaten") %>%
  kable_styling(latex_options = "hold_position")
```

Table 13: Trainingsdaten

	Period	region	Logistikdienstleister		IFR	Temperature_C	Rain_mm	Sunshine_h
52	Dec 2018	Peking	CPS Warehousing		0.8227964	-2.218	93.504	8.122
295	Sep 2021	Skorea	CPS Warehousing		0.8354299	12.898	59.955	190.065
281	Sep 2019	Japan	CPS Warehousing		0.8241058	13.180	44.828	138.760
39	Aug 2020	Shangh	CPS Warehousing		0.7911334	30.785	6.175	352.352
60	Dec 2019	Skorea	CPS Warehousing		0.8313408	0.354	46.088	32.743
89	Feb 2020	Shangh	CPS Warehousing		0.7970938	16.414	97.308	159.699
	Humidity	Congestion	InternetStability		PowerGridStability	ParkingSpaceAvailability		RoadCondition
52	70.180	31.70	1557		1.61	0.0574459		3.27
295	69.603	45.69	62		3.25	0.0323625		2.28
281	64.343	48.99	1895		0.19	0.0309810		5.81
39	56.185	64.81	314		1.36	0.0157669		6.80
60	88.765	45.48	73		3.32	0.0361720		2.52
89	83.882	67.36	303		1.25	0.0168114		6.23
	PoliticalStability		AvgHealth	Criminality	AirPollution	WaterQuality	leisureAndSocialInteractions	
52	1.25		86.06	15.92	13.47	0.98	7.04	
295	5.23		52.37	55.89	52.08	0.81	7.00	
281	2.79		77.21	30.88	222.99	0.90	6.07	
39	3.48		76.23	33.75	60.43	0.75	6.56	
60	3.86		53.86	57.81	62.24	0.83	7.02	
89	3.47		76.93	31.64	74.07	0.85	6.18	
	SkilledLaborAvailability			UnskilledLaborAvailability		WorkerMotivation	Overtime	Inflation
52	37.61			78.00		5.73	0.175	-0.0007581
295	79.11			29.70		2.95	0.117	0.0126387
281	70.26			49.64		8.02	0.129	0.0008812
39	49.64			67.12		4.40	0.223	0.0014827
60	75.19			27.80		3.41	0.000	0.0129941
89	52.02			69.30		4.07	0.275	0.0015266
			BusinessConfidence		FuelPrice	Baseline		
52			99.2765		2.535	0.8246065		
295			100.7971		2.854	0.8246065		
281			100.7918		2.178	0.8246065		
39			100.7295		2.273	0.8246065		
60			101.2981		2.147	0.8246065		
89			100.7778		2.300	0.8246065		

```

# Testset
test_data <- cps_warehousing[-training_sample_indices, ]

test_datahead <- head(test_data)
list(test_datahead[,1:7], test_datahead[,8:13], test_datahead[,14:19],
  ↪ test_datahead[,20:24], test_datahead[,25:27]) %>% kable(caption="Testdaten") %>%
  ↪ kable_styling(latex_options = "hold_position")

```

Table 14: Testdaten

	Period	region	Logistikdienstleister	IFR	Temperature_C	Rain_mm	Sunshine_h
3	Apr 2018	Phlppn	CPS Warehousing	0.8076996	14.748	21.673	243.125
24	Apr 2022	Shangh	CPS Warehousing	0.8074963	16.411	59.059	215.524
27	Aug 2018	Peking	CPS Warehousing	0.8332893	13.629	89.989	184.054
29	Aug 2018	Shangh	CPS Warehousing	0.8044351	21.662	5.684	312.734
35	Aug 2019	Skorea	CPS Warehousing	0.8117526	21.389	55.289	253.626
42	Aug 2021	Peking	CPS Warehousing	0.8196131	17.421	97.523	167.323
	Humidity	Congestion	InternetStability	PowerGridStability	ParkingSpaceAvailability	RoadCondition	
3	75.300	76.68	637	2.81	0.0226809	2.50	
24	70.353	52.37	337	1.11	0.0178755	6.29	
27	86.600	27.56	1989	1.72	0.0597833	3.40	
29	53.573	58.31	292	1.21	0.0180470	6.18	
35	70.584	46.74	75	3.06	0.0343809	2.10	
42	78.679	27.46	2004	1.84	0.0553904	4.99	
	PoliticalStability	AvgHealth	Criminality	AirPollution	WaterQuality	leisureAndSocialInteractions	
3	2.77	80.30	23.00	59.91	0.80	6.00	
24	3.08	66.29	34.07	66.33	0.88	5.84	
27	1.48	82.77	14.93	13.46	0.95	7.66	
29	2.33	77.97	33.31	64.18	0.84	5.80	
35	3.62	56.33	55.14	60.28	0.67	6.68	
42	1.74	81.62	19.80	14.04	0.98	7.30	
	SkilledLaborAvailability	UnskilledLaborAvailability	WorkerMotivation	Overtime	Inflation		
3	36.96	60.52	6.22	0.000	0.0004873		
24	49.50	65.12	3.71	0.229	0.0014817		
27	36.72	77.95	6.39	0.126	-0.0008384		
29	48.61	61.07	6.06	0.139	0.0014731		
35	75.28	28.56	3.39	0.000	0.0129481		
42	43.11	81.16	4.79	0.000	-0.0010801		
	BusinessConfidence	FuelPrice	Baseline				
3	100.5846	2.593	0.8246065				
24	100.0720	2.738	0.8246065				
27	99.2156	2.428	0.8246065				
29	101.0271	2.494	0.8246065				
35	101.4334	2.195	0.8246065				
42	99.9786	2.553	0.8246065				

Aufgabe 10

Iteration 1: Univariate Modelle

```

# Modelle erstellen
model1 = lm(IFR ~ SkilledLaborAvailability, data = training_data)
model2 = lm(IFR ~ UnskilledLaborAvailability, data = training_data)
model3 = lm(IFR ~ Criminality, data = training_data)
model4 = lm(IFR ~ Inflation, data = training_data)
model5 = lm(IFR ~ AirPollution, data = training_data)

# Zusammenfassung ausgeben zur Betrachtung der statistischen Signifikanz
cat("\nModell 1 -----")

```

```
##
## Modell 1 -----
summary(model1)

##
## Call:
## lm(formula = IFR ~ SkilledLaborAvailability, data = training_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.053315 -0.010353 -0.000873  0.010783  0.046878
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      7.770e-01  3.427e-03  226.70  <2e-16 ***
## SkilledLaborAvailability 8.698e-04  6.021e-05   14.45  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01602 on 238 degrees of freedom
## Multiple R-squared:  0.4673, Adjusted R-squared:  0.465
## F-statistic: 208.7 on 1 and 238 DF, p-value: < 2.2e-16

cat("\nModell 2 -----")

##
## Modell 2 -----
summary(model2)

##
## Call:
## lm(formula = IFR ~ UnskilledLaborAvailability, data = training_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.058902 -0.011944 -0.000332  0.012344  0.036682
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      8.676e-01  3.337e-03  260.04  <2e-16 ***
## UnskilledLaborAvailability -7.697e-04  5.608e-05  -13.73  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0164 on 238 degrees of freedom
## Multiple R-squared:  0.4418, Adjusted R-squared:  0.4395
## F-statistic: 188.4 on 1 and 238 DF, p-value: < 2.2e-16

cat("\nModell 3 -----")

##
## Modell 3 -----
```

```
summary(model3)
```

```
##
## Call:
## lm(formula = IFR ~ Criminality, data = training_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.051972 -0.012850 -0.001866  0.012725  0.051744
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.990e-01   3.310e-03  241.353   < 2e-16 ***
## Criminality  8.047e-04   9.773e-05   8.234   1.2e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01937 on 238 degrees of freedom
## Multiple R-squared:  0.2217, Adjusted R-squared:  0.2184
## F-statistic: 67.8 on 1 and 238 DF, p-value: 1.202e-14
```

```
cat("\nModell 4 -----")
```

```
##
## Modell 4 -----
```

```
summary(model4)
```

```
##
## Call:
## lm(formula = IFR ~ Inflation, data = training_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.055566 -0.013302 -0.002961  0.012680  0.056169
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.81812     0.00146  560.169   < 2e-16 ***
## Inflation    2.04214     0.25170   8.113 2.63e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01943 on 238 degrees of freedom
## Multiple R-squared:  0.2167, Adjusted R-squared:  0.2134
## F-statistic: 65.82 on 1 and 238 DF, p-value: 2.627e-14
```

```
cat("\nModell 5 -----")
```

```
##
## Modell 5 -----
```

```
summary(model5)
```

```
##
```

```
## Call:
## lm(formula = IFR ~ AirPollution, data = training_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.039697 -0.013710 -0.002144  0.012259  0.063765
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  8.122e-01  1.937e-03  419.381  < 2e-16 ***
## AirPollution 1.459e-04  1.789e-05   8.153 2.03e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01941 on 238 degrees of freedom
## Multiple R-squared:  0.2183, Adjusted R-squared:  0.215
## F-statistic: 66.48 on 1 and 238 DF, p-value: 2.028e-14

# Berechnung der Fehlerkennzahlen und einfügen in DataFrame
# Data Frame erweitern
evaluation = rbind(evaluation, data.frame(Model = c("Model1", "Model2", "Model3",
  ↪ "Model4", "Model5"),
                                MAE = numeric(5),
                                MAPE = numeric(5),
                                Rsquared=numeric(5),
                                Rsquared_adj=numeric(5)))

# Funktionen definieren für MAE und MAPE
MAE <- function(model) {
  return (mean(abs(model$residuals)))
}

MAPE <- function(model) {
  return (
    mean(abs((as.numeric(model$residuals)/
              ↪ as.numeric(model$fitted.values + model$residuals))*100))
  )
}

# MAE berechnen
evaluation[evaluation$Model == "Model1",]$MAE = MAE(model1)
evaluation[evaluation$Model == "Model2",]$MAE = MAE(model1)
evaluation[evaluation$Model == "Model3",]$MAE = MAE(model1)
evaluation[evaluation$Model == "Model4",]$MAE = MAE(model1)
evaluation[evaluation$Model == "Model5",]$MAE = MAE(model1)

# MAPE berechnen (SMAPE nicht benötigt, weil IFR immer >0)
evaluation[evaluation$Model == "Model1",]$MAPE =
  ↪ mean(abs((as.numeric(model$residuals)/as.numeric(training_data$IFR))*100))
evaluation[evaluation$Model == "Model2",]$MAPE =
  ↪ mean(abs((as.numeric(model$residuals)/as.numeric(training_data$IFR))*100))
evaluation[evaluation$Model == "Model3",]$MAPE =
  ↪ mean(abs((as.numeric(model$residuals)/as.numeric(training_data$IFR))*100))
evaluation[evaluation$Model == "Model4",]$MAPE =
  ↪ mean(abs((as.numeric(model$residuals)/as.numeric(training_data$IFR))*100))
evaluation[evaluation$Model == "Model5",]$MAPE =
  ↪ mean(abs((as.numeric(model$residuals)/as.numeric(training_data$IFR))*100))
```



```

evaluation[evaluation$Model == "Model5",]$MAPE =
  ↳ mean(abs((as.numeric(model5$residuals)/as.numeric(training_data$IFR))*100))

# R2 berechnen
evaluation[evaluation$Model == "Model1",]$Rsquared = summary(model1)$r.squared
evaluation[evaluation$Model == "Model2",]$Rsquared = summary(model2)$r.squared
evaluation[evaluation$Model == "Model3",]$Rsquared = summary(model3)$r.squared
evaluation[evaluation$Model == "Model4",]$Rsquared = summary(model4)$r.squared
evaluation[evaluation$Model == "Model5",]$Rsquared = summary(model5)$r.squared

# Adjusted R2 berechnen
evaluation[evaluation$Model == "Model1",]$Rsquared_adj = summary(model1)$adj.r.squared
evaluation[evaluation$Model == "Model2",]$Rsquared_adj = summary(model2)$adj.r.squared
evaluation[evaluation$Model == "Model3",]$Rsquared_adj = summary(model3)$adj.r.squared
evaluation[evaluation$Model == "Model4",]$Rsquared_adj = summary(model4)$adj.r.squared
evaluation[evaluation$Model == "Model5",]$Rsquared_adj = summary(model5)$adj.r.squared

# Tabelle anzeigen
evaluation %>% kable(caption="Bewertung der Baseline und der univariaten Modelle nach
  ↳ MAE, MAPE, R2 und adjusted R2", digits=3) %>% kable_styling(latex_options =
  ↳ "hold_position")

```

Table 15: Bewertung der Baseline und der univariaten Modelle nach MAE, MAPE, R² und adjusted R²

Model	MAE	MAPE	Rsquared	Rsquared_adj
Baseline	0.018	2.152	NA	NA
Model1	0.013	1.532	0.467	0.465
Model2	0.013	1.630	0.442	0.439
Model3	0.013	1.885	0.222	0.218
Model4	0.013	1.874	0.217	0.213
Model5	0.013	1.882	0.218	0.215

Wir sehen zunächst, dass alle univariaten Modelle Verbesserungen gegenüber der Baseline darstellen, sowohl nach MAE als auch MAPE. Wir entscheiden uns hier für Modell 1 (Variable “SkilledLaborAvailability”), da dieses konsistent die besten Werte für MAE, MAPE, R² und Adjusted R² hat. Darüber hinaus sind die Parameter statistisch hoch signifikant (und auch signifikanter als bei den Modellen 3 bis 5 - Modell 2 liegt hier gleich auf).

Iteration 2: Bivariate Modelle

Alle anderen Variablen korrelieren mit mehr als 0,5 mit SkilledLaborAvailability (siehe Aufgabe 5), was schon relativ hoch ist, das heißt Multikollinearität tritt auf. Um dennoch eine weitere Iteration der Forward Selection zu ermöglichen, wählen wir als Schwellenwert eine betragsweise Korrelation von 0,7 mit SkilledLaborAvailability. Darunter liegen die Variablen “Inflation” und “AirPollution”, welche wir nun also weiter betrachten werden. “UnskilledLaborAvailability” und “Criminality” liegen darüber.

```

# Modelle erstellen
model1_1 = lm(IFR ~ SkilledLaborAvailability + Inflation, data = training_data)
model1_2 = lm(IFR ~ SkilledLaborAvailability + AirPollution, data = training_data)

# Zusammenfassung ausgeben zur Betrachtung der statistischen Signifikanz
cat("\nModell 1.1 -----")

```

```
##
## Modell 1.1 -----

summary(modell1_1)

##
## Call:
## lm(formula = IFR ~ SkilledLaborAvailability + Inflation, data = training_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.053449 -0.010365 -0.000868  0.010816  0.046716
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      7.772e-01  4.063e-03 191.271  <2e-16 ***
## SkilledLaborAvailability 8.655e-04  8.197e-05 10.559  <2e-16 ***
## Inflation          2.227e-02  2.826e-01   0.079    0.937
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01606 on 237 degrees of freedom
## Multiple R-squared:  0.4673, Adjusted R-squared:  0.4628
## F-statistic: 103.9 on 2 and 237 DF, p-value: < 2.2e-16

cat("\nModell 1.2 -----")

##
## Modell 1.2 -----

summary(modell1_2)

##
## Call:
## lm(formula = IFR ~ SkilledLaborAvailability + AirPollution, data = training_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.049694 -0.009788 -0.001021  0.009770  0.049694
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      7.785e-01  3.415e-03 227.962  < 2e-16 ***
## SkilledLaborAvailability 7.665e-04  6.906e-05 11.100  < 2e-16 ***
## AirPollution      4.939e-05  1.694e-05   2.916  0.00389 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01578 on 237 degrees of freedom
## Multiple R-squared:  0.4857, Adjusted R-squared:  0.4814
## F-statistic: 111.9 on 2 and 237 DF, p-value: < 2.2e-16

# Berechnung der Fehlerkennzahlen und einfügen in DataFrame
# Data Frame erweitern
evaluation = rbind(evaluation, data.frame(Model = c("Modell1_1", "Modell1_2"),
                                             MAE = numeric(2),
```

```

MAPE = numeric(2),
Rsquared=numeric(2),
Rsquared_adj=numeric(2)))

# MAE berechnen
evaluation[evaluation$Model == "Modell1_1",]$MAE = MAE(model1_1)
evaluation[evaluation$Model == "Modell1_2",]$MAE = MAE(model1_2)

# MAPE berechnen (SMAPE nicht benötigt, weil IFR immer >0)
evaluation[evaluation$Model == "Modell1_1",]$MAPE = MAPE(model1_1)
evaluation[evaluation$Model == "Modell1_2",]$MAPE = MAPE(model1_2)

# R² berechnen
evaluation[evaluation$Model == "Modell1_1",]$Rsquared = summary(model1_1)$r.squared
evaluation[evaluation$Model == "Modell1_2",]$Rsquared = summary(model1_2)$r.squared

# Adjusted R² berechnen
evaluation[evaluation$Model == "Modell1_1",]$Rsquared_adj =
  ↳ summary(model1_1)$adj.r.squared
evaluation[evaluation$Model == "Modell1_2",]$Rsquared_adj =
  ↳ summary(model1_2)$adj.r.squared

# Tabelle anzeigen
evaluation %>% kable(caption="Bewertung aller Modelle (inkl. bivariat) nach MAE, MAPE, R²
  ↳ und adjusted R²", digits=3) %>% kable_styling(latex_options = "hold_position")

```

Table 16: Bewertung aller Modelle (inkl. bivariat) nach MAE, MAPE, R^2 und adjusted R^2

Model	MAE	MAPE	Rsquared	Rsquared_adj
Baseline	0.018	2.152	NA	NA
Model1	0.013	1.532	0.467	0.465
Model2	0.013	1.630	0.442	0.439
Model3	0.013	1.885	0.222	0.218
Model4	0.013	1.874	0.217	0.213
Model5	0.013	1.882	0.218	0.215
Modell1_1	0.013	1.532	0.467	0.463
Modell1_2	0.012	1.532	0.486	0.481

Bei Modell1_1 (mit der Variable Inflation) stellen wir keine Verbesserung in den Kennzahlen fest, beim adjusted R^2 sogar eine Verschlechterung. Darüber hinaus ist der Parameter nicht statistisch signifikant. Bei Modell1_2 (mit AirPollution) gibt es eine leichte Verbesserung in den Kennzahlen bei immer noch guter statistischer Signifikanz. Die Verbesserung kann die Erhöhung der Modellkomplexität gerade noch rechtfertigen, und wir testen noch ein trivariates Modell.

Iteration 3: Trivariates Modell

Die Korrelation zwischen AirPollution und Inflation ist sehr gering, daher erstellen wir ein Modell aus den Variablen SkilledLaborAvailability, AirPollution und Inflation (Korrelation mit SkilledLaborAvailability haben wir schon in der zweiten Iteration betrachtet).

```

# Modelle erstellen
modell1_2_1 = lm(IFR ~ SkilledLaborAvailability + AirPollution + Inflation, data =
  ↳ training_data)

```

```

# Zusammenfassung ausgeben zur Betrachtung der statistischen Signifikanz
cat("\nModell 1.2.1 -----")

##
## Modell 1.2.1 -----

summary(model1_2_1)

##
## Call:
## lm(formula = IFR ~ SkilledLaborAvailability + AirPollution +
##     Inflation, data = training_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.053300 -0.009119 -0.001210  0.010438  0.043337
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    7.907e-01  4.939e-03 160.077 < 2e-16 ***
## SkilledLaborAvailability 3.774e-04  1.345e-04   2.806 0.005432 **
## AirPollution    1.095e-04  2.446e-05   4.479 1.17e-05 ***
## Inflation      1.341e+00  4.008e-01   3.346 0.000953 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01545 on 236 degrees of freedom
## Multiple R-squared:  0.509, Adjusted R-squared:  0.5028
## F-statistic: 81.55 on 3 and 236 DF, p-value: < 2.2e-16

# Berechnung der Fehlerkennzahlen und einfügen in DataFrame
# Data Frame erweitern
evaluation = rbind(evaluation, data.frame(Model = c("Modell1_2_1"),
                                             MAE = numeric(1),
                                             MAPE = numeric(1),
                                             Rsquared=numeric(1),
                                             Rsquared_adj=numeric(1)))

# MAE berechnen
evaluation[evaluation$Model == "Modell1_2_1",]$MAE = MAE(model1_2_1)

# MAPE berechnen (SMAPE nicht benötigt, weil IFR immer >0)
evaluation[evaluation$Model == "Modell1_2_1",]$MAPE = MAPE(model1_2_1)

# R² berechnen
evaluation[evaluation$Model == "Modell1_2_1",]$Rsquared = summary(model1_2_1)$r.squared

# Adjusted R² berechnen
evaluation[evaluation$Model == "Modell1_2_1",]$Rsquared_adj =
  ↳ summary(model1_2_1)$adj.r.squared

# Tabelle anzeigen
evaluation %>% kable(caption="Bewertung aller Modelle (inkl. trivariat) nach MAE, MAPE,
  ↳ R² und adjusted R²", digits=3) %>% kable_styling(latex_options = "hold_position")

```

Table 17: Bewertung aller Modelle (inkl. trivariat) nach MAE, MAPE, R^2 und adjusted R^2

Model	MAE	MAPE	Rsquared	Rsquared_adj
Baseline	0.018	2.152	NA	NA
Model1	0.013	1.532	0.467	0.465
Model2	0.013	1.630	0.442	0.439
Model3	0.013	1.885	0.222	0.218
Model4	0.013	1.874	0.217	0.213
Model5	0.013	1.882	0.218	0.215
Model1_1	0.013	1.532	0.467	0.463
Model1_2	0.012	1.532	0.486	0.481
Model1_2_1	0.012	1.532	0.509	0.503

Zwar ändern sich im trivariaten Modell MAE und MAPE nicht mehr, aber wir sehen eine Verbesserung von R^2 und adjusted R^2 . Die Signifikanz der einzelnen Parameter ist immer noch gut bis sehr gut. Wir übernehmen also Modell1_2_1 und beenden hier die Forward Selection, da keine weiteren (unkorrelierten) Variablen mehr zur Verfügung stehen.

Test auf Overfitting

Um auf Overfitting zu testen, wenden wir das Modell auf die Testdaten an:

```
# Testdaten vorhersagen
predictions_test <- predict(model1_2_1, newdata =
  ↪ test_data[,c("SkilledLaborAvailability", "AirPollution", "Inflation")])

# Kennzahlen berechnen
data.frame(MAE=mean(abs(test_data$IFR - predictions_test)),
  MAPE=mean(abs((as.numeric(test_data$IFR -
    ↪ predictions_test)/as.numeric(test_data$IFR)*100))) %>% kable(caption="MAE
    ↪ und MAPE des Modells auf den Testdaten", digits=4) %>%
    ↪ kable_styling(latex_options = "hold_position")
```

Table 18: MAE und MAPE des Modells auf den Testdaten

MAE	MAPE
0.0109	1.3177

Wir sehen hier keine Anzeichen für Overfitting. MAE und MAPE sind auch auf den Testdaten ähnlich (hier sogar geringer) als auf den Trainingsdaten. Das Modell scheint also gut zu generalisieren.

Aufgabe 11

```
evaluation[evaluation$Model == 'Baseline' | evaluation$Model == 'Model1_2_1', c("MAE",
  ↪ "MAPE")] %>% kable(caption="Vergleich von Baseline zum ausgewählten Modell nach
  ↪ Forward Selection", digits=3) %>% kable_styling(latex_options = "hold_position")
```

Im Vergleich zur Baseline hat das trivariate Modell einen deutlich geringeren mittleren absoluten Fehler (0,012 vs. 0,018, also 33% weniger) sowie eine geringere prozentuale Abweichung von 1,524% vs. 2,152%. Für die genaue Vorhersage der In-Full Rate kann allerdings auch eine Abweichung 1,2 Prozentpunkten relevant sein, zumal die Baseline selbst nur 1,8% abweicht. Insgesamt ist unser Modell also für den vorliegenden Zweck

Table 19: Vergleich von Baseline zum ausgewählten Modell nach Forward Selection

	MAE	MAPE
1	0.018	2.152
9	0.012	1.532

nur mäßig gut. Das sehen wir auch am eher geringen R^2 (Anteil der erklärten Varianz in den Daten) von 0,513.

Aufgabe 12

```
# Werte für die Vorhersage zusammenstellen
externals_SH_0423 <- externals[externals$Period=='Apr 2023' &
  externals$region=='Shangh',
  c("SkilledLaborAvailability", "AirPollution",
    ↪ "Inflation")]
externals_BJ_0423 <- externals[externals$Period=='Apr 2023' &
  externals$region=='Peking',
  c("SkilledLaborAvailability", "AirPollution",
    ↪ "Inflation")]

# Vorhersage
data.frame(Region=c("Shanghai", "Peking"),
  Periode=c("Apr 2023", "Apr 2023"),
  Vorhersage_IFR=c(predict(model1_2_1, newdata = externals_SH_0423),
    predict(model1_2_1, newdata = externals_BJ_0423))) %>%
kable(caption="Vorhersage des IFR-Wertes in Shanghai und Peking im April 2023",
  ↪ digits=4) %>% kable_styling(latex_options = "hold_position")
```

Table 20: Vorhersage des IFR-Wertes in Shanghai und Peking im April 2023

	Region	Periode	Vorhersage_IFR
316	Shanghai	Apr 2023	0.8182
317	Peking	Apr 2023	0.8076

Nach der Modellvorhersage liegt die In-Full Rate im April 2023 in Shanghai höher als in Peking, die Aussage der Chefin wäre also richtig und wir würden die Wette nicht annehmen.

Der Unterschied beträgt allerdings nur 1,05 Prozentpunkte. Darüber hinaus müssen wir verschiedene Unsicherheitsfaktoren in der Vorhersage berücksichtigen.

Zunächst herrscht Unsicherheit im Modell selbst. Wir könnten beispielsweise Parameter fälschlich aufgenommen haben. Diese Unsicherheit wird durch den p-Wert quantifiziert und ist hier sehr gering. Daneben kann aber auch Multikollinearität zu Unsicherheit in den geschätzten Parametern führen. Sowohl AirPollution (0.50) als auch Inflation (0.68) haben mit SkilledLaborAvailability eine Korrelation, die nicht von der Hand zu weisen ist.

Ein anderes Problem ist, dass relevante Variablen im Modell fehlen könnten. Dies ist nicht unwahrscheinlich, da wir eine eher niedrige erklärte Varianz (R^2) haben. Gründe dafür sind beispielsweise dass wir nur die 5 am stärksten mit IFR korrelierten Variablen bei der Forward Selection überhaupt betrachtet haben. Ggf. müssten hier weitere Variablen betrachtet werden. Auch eine Unterscheidung der Regionen könnte relevant sein, man könnte diese beispielsweise als Dummy-Variable in das Modell mit aufnehmen (derzeit ist das Modell sozusagen der Schnitt über alle Regionen). Außerdem haben wir kein Feature Engineering betrieben,

sodass Variablen ggf. nur geringe Linearität und damit Tauglichkeit für die Regression aufweisen. Darüber hinaus könnten auch solche Variablen relevant sein, die in unserem Datensatz gänzlich fehlen.

Schließlich kommt eine weitere Unsicherheit aus der Vorhersage der externen Effekte. Wir wissen, dass diese für 2023 auf Vorhersagen beruhen, deren Unsicherheit jedoch unklar ist, da sie aus einer externen Quelle kommen.

In Anbetracht dieser Unsicherheiten, dem ohnehin geringen Abstand der Vorhersage, dem hohen Wetteinsatz und dem Fakt, dass unsere Chefin nach Vorhersage ohnehin Recht hat, würden wir die Wette dankend ablehnen.

Entscheidung

Aufgabe 13

Mit dem vorliegenden Modell können wir die In-Full Rate von “CPS Warehousing” basierend auf externen Einflussfaktoren vorhersagen. Entsprechende Modelle könnten wir nun auch für alle weiteren Warehousing-Dienstleister erstellen, oder ein großes Modell für alle Dienstleister (mit den einzelnen WH-DL als Dummy-Variablen).

Beeinflusste Prozesse

Das Modell kann potenziell unseren gesamten Order-to-Cash-Prozess beeinflussen. Am unmittelbarsten ist natürlich das Warehousing selbst betroffen. Hier können wir mithilfe des Modells wo nötig weitere WH-Dienstleistungen frühzeitig buchen (etwa um übrig gebliebene Flaschen zu packen und laden), proaktiv mit schlecht performenden WH-DL an Verbesserungen arbeiten, oder andere WH-DL auswählen. Änderungen im Warehousing betreffen auch das Order Management, eine höhere IFR im Warehousing sollte in einer höheren IFR beim Kunden resultieren, womit wir die Kundenzufriedenheit erhöhen und im Idealfall den Auftragseingang und damit den Umsatz steigern können. Neben direkten Verbesserungen kann auch verbesserte Kommunikation die Kundenzufriedenheit steigern, etwa wenn wir Warnungen bspw. bei Extremwetter aussprechen, dass sich voraussichtlich die Ware nicht vollständig liefern lässt. Auch auf Upstream-Prozesse wie der Produktion oder dem Shipping von der Produktion zu den Großlagern können sich indirekte Auswirkungen ergeben, etwa wenn sich der Ordereingang mittelfristig ändert oder wir sehen, dass Ware an anderen Orten benötigt wird. Schließlich hat eine bessere Kenntnis der IFR unserer WH-DL auch Einfluss auf die Finanzprozesse, konkret die Debitorenbuchhaltung (Accounts Receivable). Wenn wegen zu niedriger IFR im Lager Ware beim Kunden fehlt, können wir zur Steigerung der Kundenzufriedenheit Rechnungen über unvollständige Lieferungen vorausschauend zurückhalten bzw. Mahnungen unterlassen.

Nutzer:innen

Wir können verschiedene Nutzer:innengruppen identifizieren: - Interne Auftraggeber der WH-DL: Diese können das Modell zur Bewertung und Steuerung von WH-DL nutzen und ggf. frühzeitig andere Dienstleister beauftragen. - Vertrieb: Diese können Frühwarnungen aussprechen und damit Kundenerwartungen steuern. - Angestellte der Finanzabteilung: Diese können wir oben beschrieben zur Steigerung der Kundenzufriedenheit Rechnungen zurückhalten oder Mahnungen unterlassen.

Bereitstellung

Das Modell bzw. dessen Ergebnisse können den Nutzer:innen auf verschiedenen Wegen bereitgestellt werden: - Bereitstellung als Web App/Dashboard: Mithilfe von Tools wie Shiny für R können schnell Web Apps gebaut werden, die einfachen Zugriff auf die Modellvorhersagen von Browser aus ermöglichen. - Integration in Business Intelligence-Dashboards/Reports: Falls schon BI-Lösungen wie Tableau oder Power BI verwendet werden, könnten Modellvorhersagen hier integriert werden. - Integration in CRM-Tools: Bei Verwendung von Customer Relationship Management-Tools wie Salesforce könnten Modell-Vorhersagen direkt hier eingespielt werden, damit beispielsweise Vertriebsmitarbeitende den Kunden frühzeitig warnen können. - Integration in

ERP-System: Ggf. könnten Vorhersagen auch ins SAP-System integriert werden, um an zentraler Stelle zur Verfügung zu stehen. Dies ist jedoch in den meisten Fällen deutlich aufwändiger.

Je nach Situation bietet sich eine andere Lösung an. Wir bereits ein BI-Tool verwendet, präferieren viele Nutzer:innen vermutlich eine Integration in bestehende Dashboards oder Reports. Für einen schnellen Prototypen bietet sich eine Shiny App eher an.

Datenbeschaffung

Um relevant zu bleiben, sollte das Modell regelmäßig aktualisiert werden. Dazu sollte eine Data Pipeline gebaut werden, die dem Modell regelmäßig Daten zuführt und typischerweise aus den Schritten Extraktion aus Quellsystemen, Transformation in nützliche Datenformate, und Laden in die Analyseumgebung besteht. Eine solche Datenpipeline kann je nach Bedarf mehr oder weniger automatisiert laufen.

Als Quellsysteme sind zu betrachten: - ERP-System für Daten zu in Anspruch genommenen Services, um IFR zu berechnen. Diese können wir über Schnittstellen des ERP-Systems abfragen. - Externe Datenbanken für Einflussfaktoren. Diese sollten wir in den meisten Fällen über REST APIs abfragen können. Ggf. müssen dazu Lizenzen für nicht-öffentliche Datenbanken gekauft werden.