

# Data Science und maschinelles Lernen mit R

Gruppe 1: Moritz Larsen, Ina Schicke, Janina Güntzel

# Inhalt

---

- ▶ 1. Vorstellung des Datensatzes und erstellte Variablen
- ▶ 2. Balkendiagramme mit Konfidenzintervallen
- ▶ 3. SVM für den 01.06.2019
- ▶ 4. Neuronales Netz für den 01.06.2019



# 1. Vorstellung des Datensatzes und erstellte Variablen

---

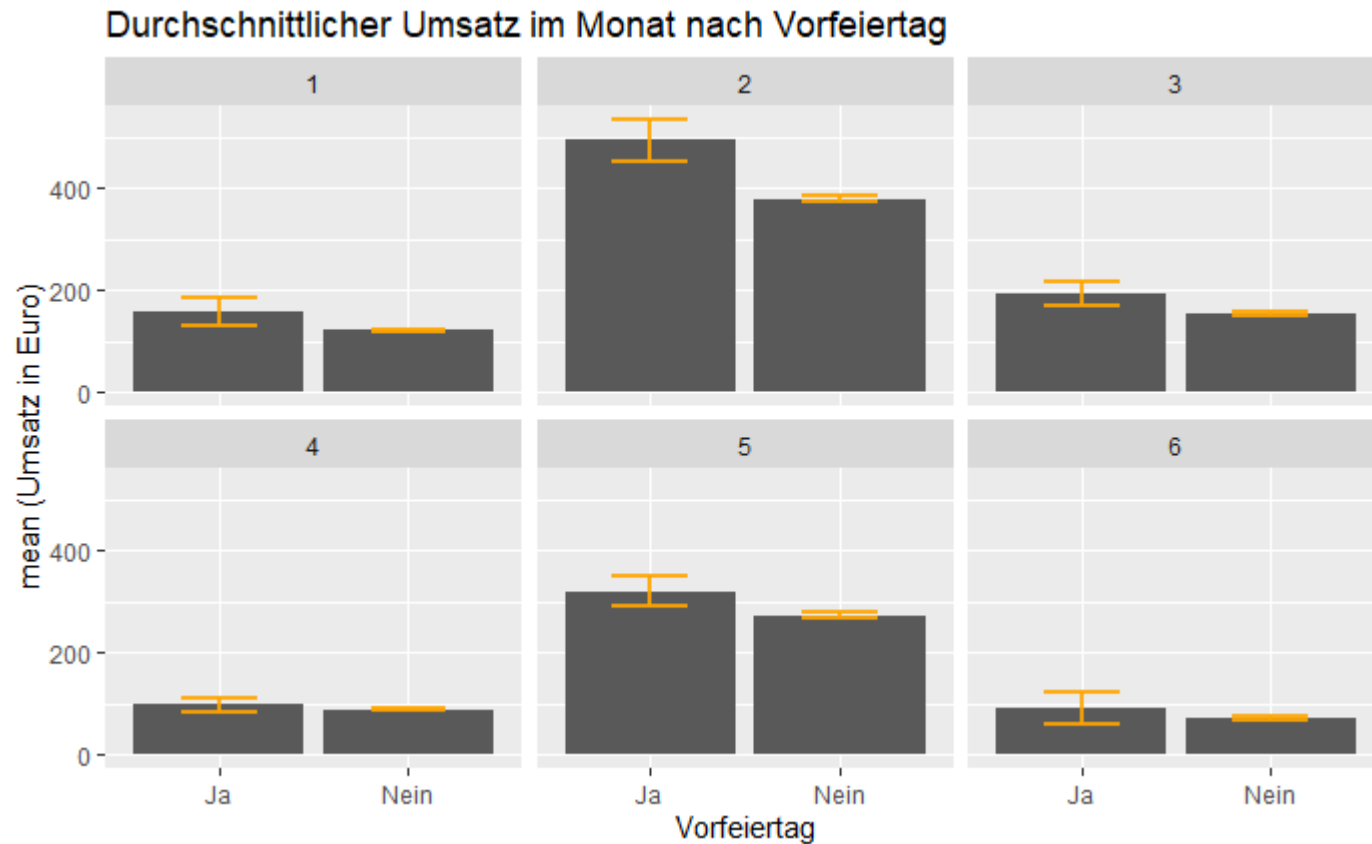
## ► Erstellte Variablen:

- Umsatz\_Vormonat, Umsatz\_Vortag, Umsatz\_Monat, vorfeiertage, EndOfMonth, BeginningOfMonth, holidays, dayOfMonth, Wochentag, Monat
- Für die Vorhersage: Temperatur, Warengruppe, Umsatz\_Vormonat, Umsatz\_Vortag, Wochentag, Kieler Woche, Monat, Holiday, Monatsende



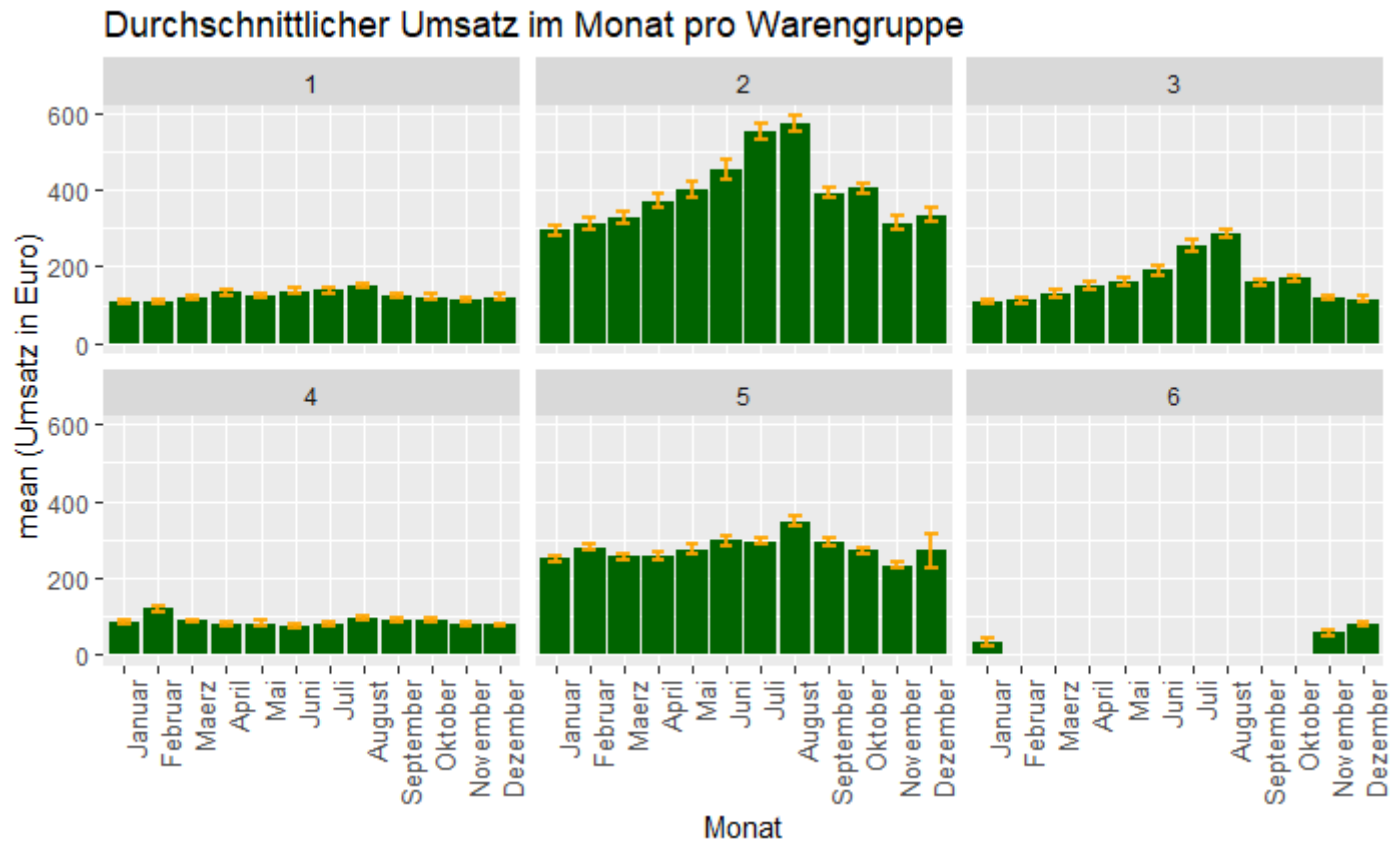
## 2. Balkendiagramme mit Konfidenzintervallen

### ► Selbst erstellte Variablen: Vorfeiertage



## 2. Balkendiagramme mit Konfidenzintervallen

### ► Selbst erstellte Variablen: Month



### 3. SVM

---

```
> svm_tune$best.model

call:
best.tune(method = svm, train.x = Umsatz ~ as.factor(Warengruppe) +
  Temperatur + Umsatz_Vormonat + Umsatz_Vortag + as.factor(weekday) +
  as.logical(Kielerwoche) + month + holiday + EndOfMonth, data = umsatzdatenSVM_train,
  ranges = list(epsilon = seq(0.2, 1, 0.1), cost = 2^(2:3)))

Parameters:
  SVM-Type:  eps-regression
SVM-Kernel:  radial
      cost:    8
      gamma: 0.03448276
  epsilon:   0.2

Number of Support Vectors: 2225
```



# Ergebnisse SVM

---

Warengruppe	MAPE Training	MAPE Test
1. Brot	19.44	21.40
2. Brötchen	8.70	9.99
3. Croissant	15.01	15.63
4. Konditorei	21.68	23.46
5. Kuchen	11.60	13.45
6. Saisonbrot	45.06	30.43
Gesamt	16.01	17.23



# Ergebnisse SVM

---

Warengruppe	Umsatz in Euro
1. Brot	190
2. Brötchen	584
3. Croissant	259
4. Konditorei	75
5. Kuchen	337
6. Saisonbrot	NA





## 4. Neuronales Netz

---

```
# Definition der Form des tiefen neuronalen Netzes (Deep Neural Nets)
model = keras.Sequential([
    layers.Dense(22, activation='relu', input_shape=[len(r.train_dataset.keys())]),
    layers.Dense(22, activation='relu'),
    layers.Dense(1)
])

# Definition der Kosten-(Loss-)Funktion und der Optimierungsfunktion mit seinen Hyperparametern
model.compile(loss="mse",
              optimizer=tf.keras.optimizers.SGD(lr=1e-4, momentum=0.9))
```



# Ergebnisse Neuronales Netz

---

Warengruppe	MAPE Training	MAPE Test
1. Brot	23.78	24.76
2. Brötchen	11.59	12.45
3. Croissant	17.57	17.46
4. Konditorei	27.22	26.93
5. Kuchen	14.34	15.97
6. Saisonbrot	66.43	40.33
Gesamt	20.08	20.15



# Ergebnisse Neuronales Netz

---

Vorhergesagter Umsatz pro Warengruppe	Umsatz in Euro
1. Brot	177
2. Brötchen	565
3. Croissant	240
4. Konditorei	79
5. Kuchen	385
6. Saisonbrot	NaN



# Link vom Repository

---

- ▶ <https://github.com/moritzlarsen/gl/tree/master/Final>

