BAKERY SALES FORECASTING WITH MACHINE LEARNING

PRESENTED BY

ANNA PETRI

TILON JÜRGENSEN

LINUS UZOEWULU

SELF-CREATED VARIABLES

Variable	Description
Wochentag	Day of the week (0=Monday, …, 6=Sunday)
Monat	Month of the year (1-12)
IstWochenende	1 if Saturday/Sunday, else 0
KW	Calendar week number
TagSeitWochenstart	Day since start of week (0-6)
Sin_Monat, Cos_Monat	Sine and cosine encoding of month (seasonality)
Wetter_extrem	1 if temperature <0°C or >30°C, else 0
Temp_Step	Categorical binning of temperature (cold, mild, hot)
Temp_Wind	Product of temperature and wind speed

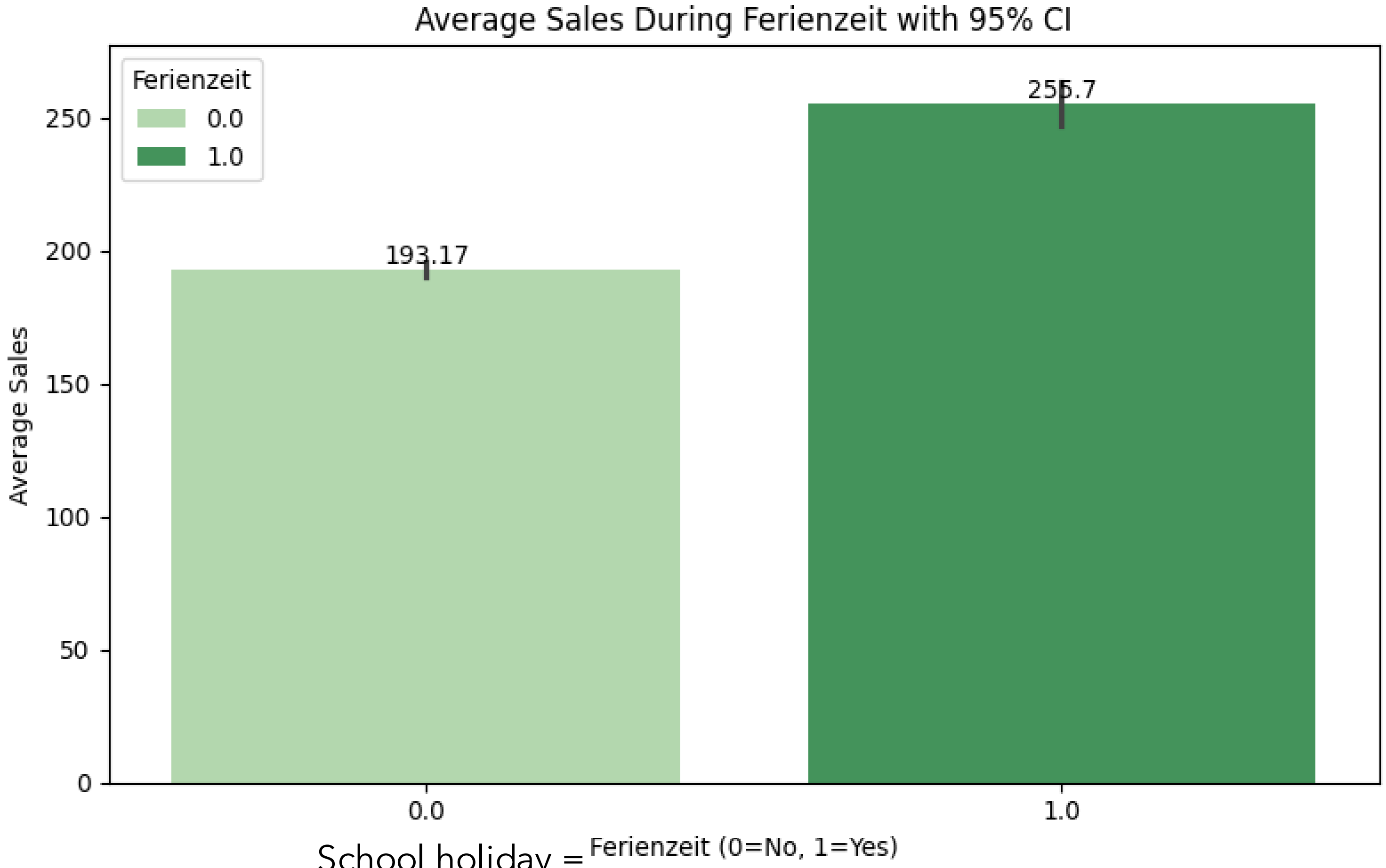
SELF-CREATED VARIABLES

Variable	Description
Ferienzeit	1 if school holiday, else 0
FerienName, FerienName_Code	Name/code of the holiday period
Feiertag	1 if public holiday, else 0
KielerWoche	1 if during Kieler Woche event, else 0
Temperatur_2	Temperatur^2
Umsatz_lag_1	Sales from the previous day (1 day lag)
Umsatz_lag_7	Sales from one week ago (7 days lag)

SELF-CREATED VARIABLES

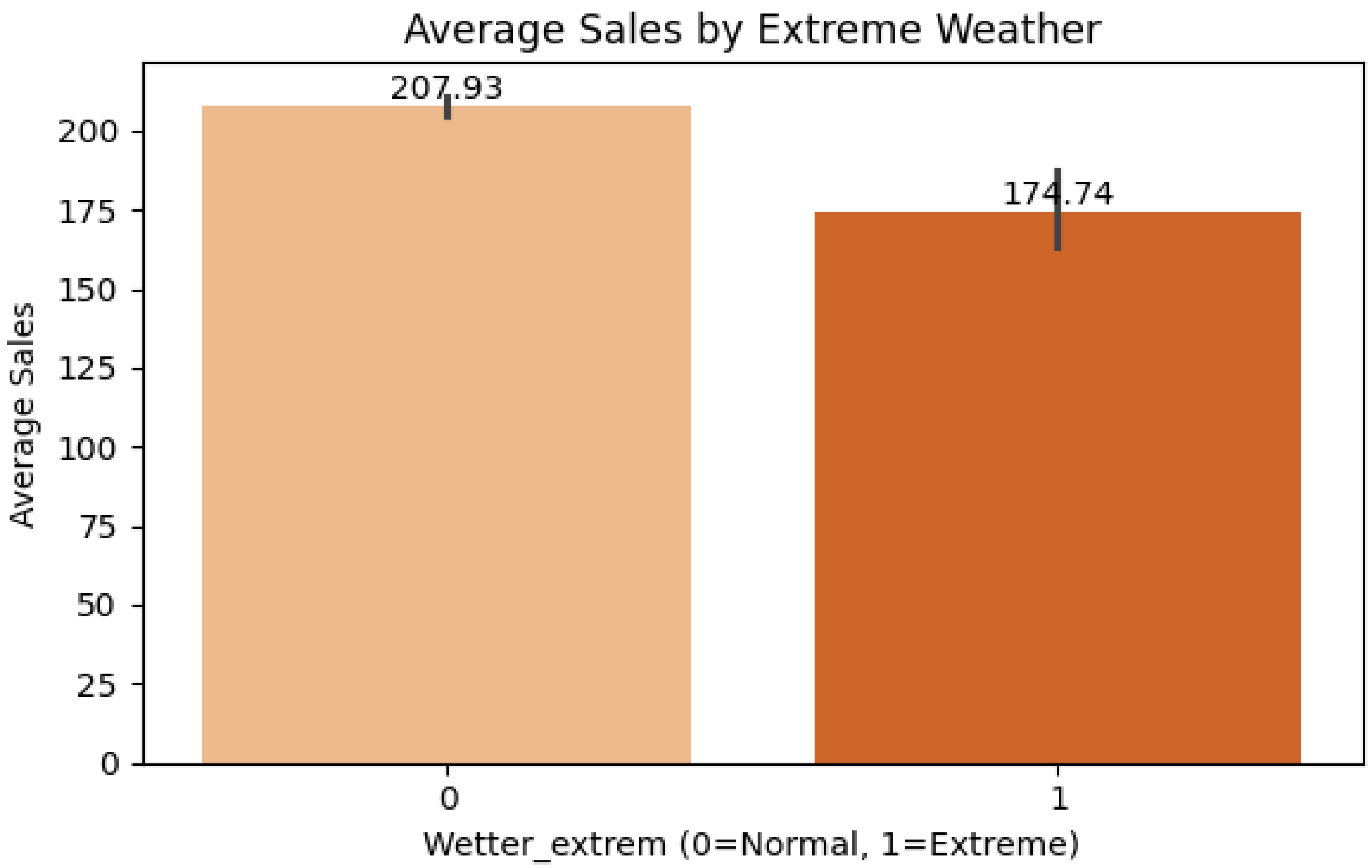
Variable	Description
Weather Code Clustering	
Cluster_0	0; no observation
Cluster_1	1-12, 40-49; clouds and fog
Cluster_2	50-55, 58-63, 65, 80-81; rain and drizzle
Cluster_3	20-23, 25-29; end of precipitation and weather
Cluster_4	13-19, 91-97, 99; thunderstorms and special weather phenomena
Cluster_5	68-79, 83-90; snow, sleet and hail
Cluster_6	30-39, 98; sandstorms and snowstorms
Cluster_7	24, 56-57, 66-67; freezing precipitation & freezing drizzle

BAR CHARTS FOR SELF-CREATED VARIABLES



School holiday =

BAR CHARTS FOR SELF-CREATED VARIABLES



MISSING VALUE IMPUTATION

- Wettercode (about 25% missing):
  - kNN-Imputation (kNNImputer with 5 nearest neighbors)
- Bewoelkung (<1% missing)
  - kNN-Imputation (kNNImputer with 5 nearest neighbors)
- Temperatur:
  - Interpolated
  - kNN
- Windgeschwindigkeit
  - Filled with median
  - kNN

LINEAR MODEL OPTIMIZATION

<u>Used features:</u>

- Warengruppe (large impact, improvement of 70%)
- Temperatur, Temperatur_2 (almost no impact)
- KielerWoche
- Monat
- IstWochenende
- Feiertag
- Ferienzeit
- Umsatz_lag_1 (improvement of 3%)
- Umsatz_lag_2 (almost no impact)
- Wettercode (Cluster_1,… Cluster_6)

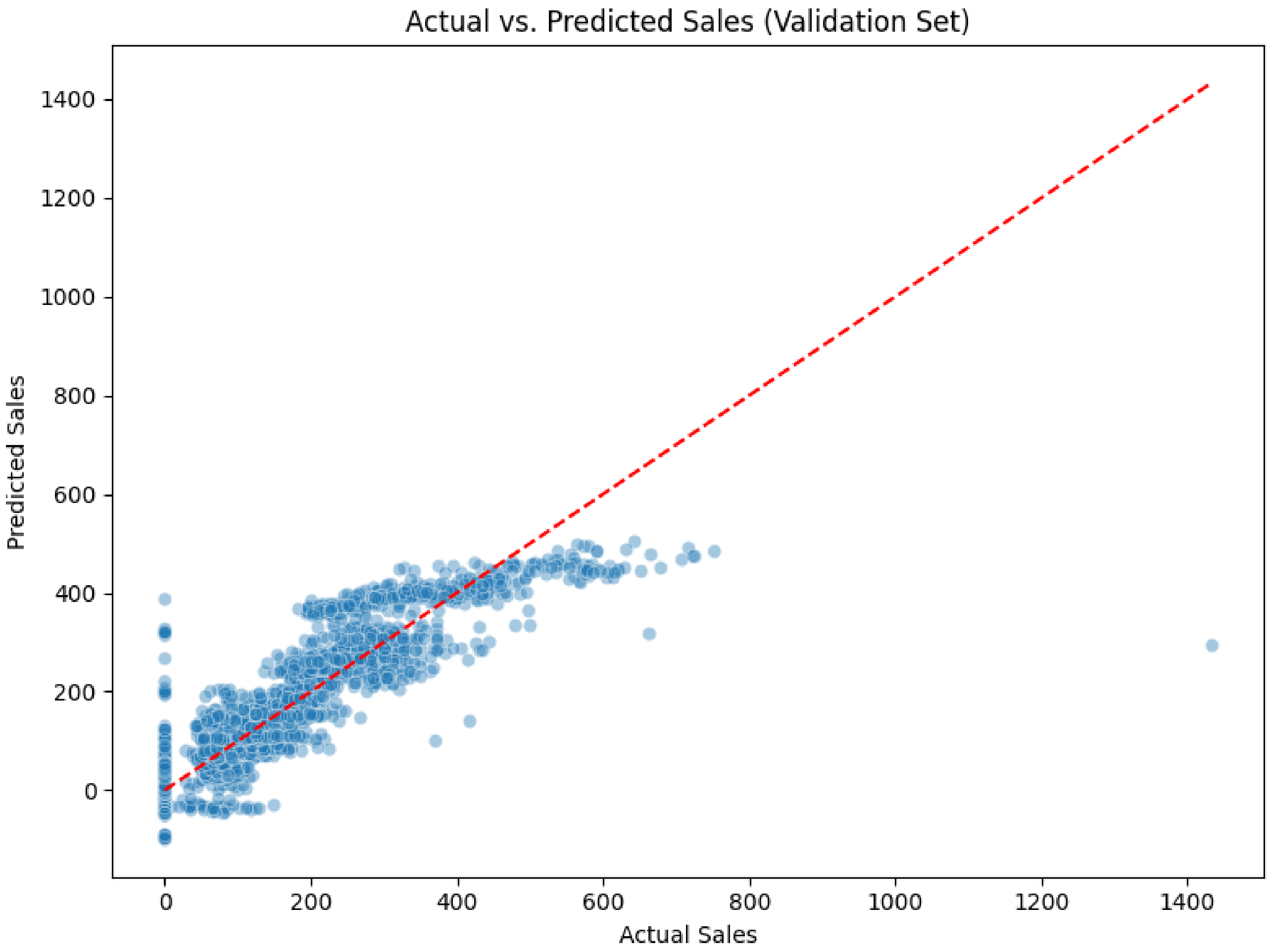
LINEAR MODEL OPTIMIZATION

- Model equation:

Umsatz = 90.5660 + 4.4465 * Temperatur - 0.4527 * Temperatur_2 + 23.0578 * KielerWoche + 36.4669 * IstWochenende - 49.6721 * Feiertag + 19.6123 * Ferienzeit + 24.0831 * Umsatz_lag_1 + 1.0166 * Umsatz_lag_7 + 265.9288 * Warengruppe_2 - 21.1821 * Warengruppe_3 - 59.6554 * Warengruppe_4 + 144.6081 * Warengruppe_5 - 150.9201 * Warengruppe_6 + 17.9822 * Monat_2 + 7.3783 * Monat_3 + 12.2037 * Monat_4 + 24.4876 * Monat_5 + 28.8565 * Monat_6 + 48.2495 * Monat_7 + 63.4480 * Monat_8 + 27.9586 * Monat_9 + 27.2881 * Monat_10 + 13.6632 * Monat_11 + 10.9786 * Monat_12 + 20.0927 * Cluster_1 + 15.6214 * Cluster_2 + 14.0265 * Cluster_3 + 15.7960 * Cluster_4 + 17.7035 * Cluster_5 + 7.3258 * Cluster_6

LINEAR MODEL OPTIMIZATION

- Adjusted R^2: 0.752
- Validation R^2: 0.744



NEURONAL NETWORK OPTIMIZATION

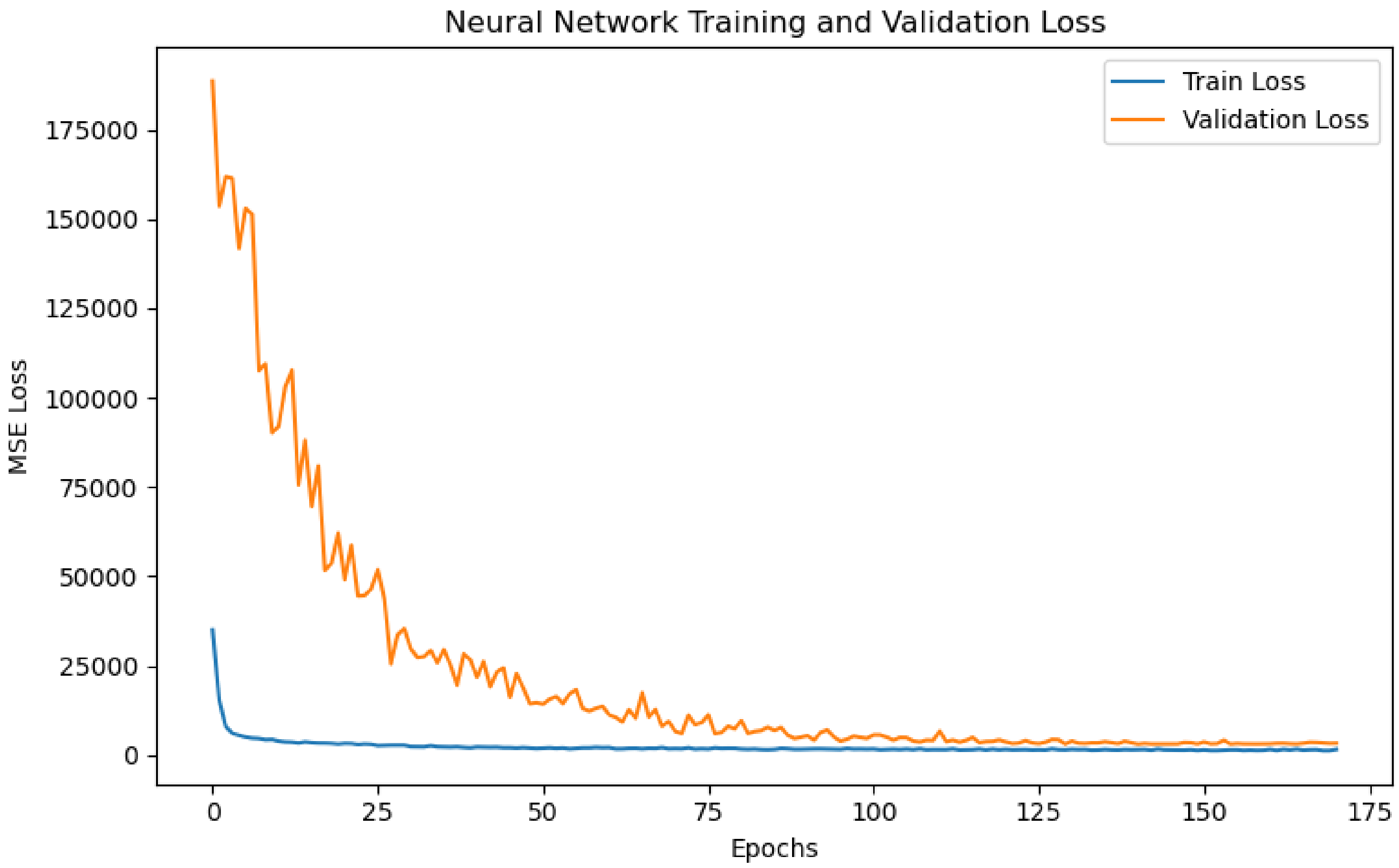
a) Source Code Defining the Neural Network

```
model = Sequential([
    Dense(64, activation='relu', kernel_regularizer=l2(0.0005), input_shape=(X_train_scaled.shape[1],)),
    Dropout(0.2),
    Dense(32, activation='relu', kernel_regularizer=l2(0.0005)),
    Dropout(0.2),
    Dense(1)
])

optimizer = Adam(learning_rate=0.005)
model.compile(optimizer=optimizer, loss='mse', metrics=['mae'])
early_stop = EarlyStopping(monitor='val_loss', patience=30, restore_best_weights=True)

history = model.fit(
    X_train_scaled, y_train,
    validation_data=(X_val_scaled, y_val),
    epochs=200,
    batch_size=32,
    callbacks=[early_stop],
    verbose=2,  # type: ignore
)
```

NEURONAL NETWORK OPTIMIZATION



NEURONAL NETWORK OPTIMIZATION

- Overall Validation MAPE: 21.21%
- <u>MAPE by Product Group (Warengruppe):</u>
  - Warengruppe 1: 24.17%
  - Warengruppe 2: 13.25%
  - Warengruppe 3: 22.27%
  - Warengruppe 4: 23.43%
  - Warengruppe 5: 16.55%
  - Warengruppe 6: 61.81%

RANDOM FOREST OPTIMIZATION

- From scikit-learn
- Key settings:
  - N-estimators=200
  - Max_depth=25
  - Min_samples_split=5
  - Random_state=42
  - Features were not scaled (tree-based models don't require scaling)
- The model was trained on the same feature set as the NN, including all engineered variables

RANDOM FOREST OPTIMIZATION

- Overall Validation MAPE: 16.84% (best overall score)
- <u>MAPE by Product Group (Warengruppe):</u>
  - Warengruppe 1: 18.07%
  - Warengruppe 2: 11.04%
  - Warengruppe 3: 17.31%
  - Warengruppe 4: 18.95%
  - Warengruppe 5: 13.32%
  - Warengruppe 6: 49.88%

WORST FAIL

Linear Model Optimization:

- Weather code clustering had no impact but resulted in a shorter model equation.
- Leaving out Temperatur and Umsatz_lag_7 had no effect on the adjusted R^2.
- Normalization of numerical features (Temperatur, Umsatz_lag_1) had only a minor impact.

In General:

- Generating the submission file with IDs that correctly matched the sample submission was unexpectedly difficult. ChatGPT struggled with this, and we had to adjust the output several times.

BEST IMPROVEMENT

- The product group feature led to a 70% improvement in the linear model's adjusted R^2.
- **Dropout (0.2)** reduced overfitting and led to lower validation loss.
- **EarlyStopping with patience=30** helped by restoring the best weights before overfitting started.
- **Adam optimizer** adaptively adjusted the learning rate for each parameter, which improved convergence stability.

THANK YOU FOR YOUR ATTENTION!