

Klimabedingte Verkaufsanalyse: Ventilatoren und Kühleräte

Einfluss der Temperatur auf Produktverkäufe in Schweizer Städten

Cooling Analysis Team

2025-11-05

Table of contents

1 Executive Summary	2
2 1. Business Problem & Unit of Analysis	2
2.1 1.1 Problemstellung	2
2.2 1.2 Unit of Analysis	2
3 2. Datenaufbereitung	3
3.1 2.1 Laden der erforderlichen Pakete	3
3.2 2.2 Daten einlesen	3
3.3 2.3 Filterung der Kühleräte	4
3.4 2.4 Stadt-Mapping	5
4 3. Explorative Datenanalyse	6
4.1 3.1 Verkäufe nach Stadt	6
4.2 3.2 Visualisierung: Verkäufe vs. Temperatur	7
4.3 3.3 Normalisierung nach Bevölkerungsgröße	8
4.4 3.4 Vergleich: Absolute vs. Normalisierte Werte	10
5 4. Korrelationsanalyse	11
5.1 4.1 Pearson-Korrelation	11
5.2 4.2 Lineare Regression	12
5.3 4.3 Korrelationsmatrix (alle Variablen)	14
6 5. Detaillierte Stadt-Analyse	15
6.1 5.1 Ranking der Städte	15
6.2 5.2 Vergleich: Tessin vs. Rest der Schweiz	16

6.3	5.3 Verkäufe nach Produktkategorie und Stadt	18
7	6. Zeitliche Analyse	19
8	7. Schlussfolgerungen und Empfehlungen	20
8.1	7.1 Wichtigste Erkenntnisse	20
8.2	7.2 Interpretation	22
8.3	7.3 Business-Empfehlungen	22
9	8. Limitationen	23

1 Executive Summary

Diese Analyse untersucht den Zusammenhang zwischen Durchschnittstemperatur und Verkaufszahlen von Ventilatoren und Kühlgeräten in acht Schweizer Städten. Die zentrale Forschungsfrage lautet: **Gibt es eine Korrelation zwischen höheren Temperaturen und verstärkten Verkäufen von Kühlprodukten?**

Haupthypothese: In wärmeren Regionen (insbesondere im Tessin, z.B. Lugano mit 24.8°C) werden mehr Ventilatoren und Kühlgeräte verkauft als in kühleren Städten.

2 1. Business Problem & Unit of Analysis

2.1 1.1 Problemstellung

Digitec Galaxus möchte verstehen, ob und wie stark die klimatischen Bedingungen die Nachfrage nach Kühlprodukten beeinflussen. Diese Erkenntnisse können verwendet werden für:

- **Bestandsmanagement:** Optimierung der Lagerhaltung basierend auf lokalen Temperaturen
- **Marketing-Strategien:** Gezielte Werbekampagnen in wärmeren Regionen
- **Pricing-Strategien:** Dynamische Preisgestaltung basierend auf Wetterbedingungen
- **Produktsortiment:** Anpassung des Angebots je nach Region

2.2 1.2 Unit of Analysis

Die Analyseeinheit sind **Schweizer Städte**. Wir untersuchen folgende acht Städte:

1. **Zürich (ZH)** - 436,551 Einwohner, 21.1°C
2. **Bern (BE)** - 137,995 Einwohner, 20.7°C
3. **Luzern (LU)** - 86,234 Einwohner, 21.2°C

4. **Basel** (BS) - 177,571 Einwohner, 22.2°C
5. **St. Gallen** (SG) - 78,863 Einwohner, 19.6°C
6. **Lugano** (TI) - 63,629 Einwohner, 24.8°C (wärmste Stadt)
7. **Lausanne** (VD) - 144,873 Einwohner, 22.4°C
8. **Genf** (GE) - 209,061 Einwohner, 22.5°C

Die Durchschnittstemperatur bezieht sich auf den **wärmsten Monat** des Jahres.

3 2. Datenaufbereitung

3.1 2.1 Laden der erforderlichen Pakete

```
# Pakete laden
library(tidyverse)
library(readr)
library(ggplot2)
library(knitr)
library(scales)
library(corrplot)
library(gridExtra)

# Optionen setzen
options(scipen = 999) # Wissenschaftliche Notation ausschalten
theme_set(theme_minimal())
```

3.2 2.2 Daten einlesen

```
# Temperaturdaten und Bevölkerungsdaten einlesen
temp_data <- read_delim("Temparatur und Anzahl Personen.csv",
                        delim = ";",
                        locale = locale(encoding = "UTF-8"))

# Spalten umbenennen für einfachere Handhabung
colnames(temp_data) <- c("Stadt", "Bevoelkerung", "Temperatur")

# Verkaufsdaten einlesen
sales_data <- read_csv("DigitecLive_Cleaned.csv",
                        locale = locale(encoding = "UTF-8"))
```

```
# Überblick über die Daten
kable(temp_data,
      caption = "Städte mit Bevölkerung und Durchschnittstemperatur",
      format.args = list(big.mark = "''))
```

Table 1: Städte mit Bevölkerung und Durchschnittstemperatur

Stadt	Bevoelkerung	Temperatur
Zürich	436'551	21.1
Bern	137'995	20.7
Luzern	86'234	21.2
Basel	177'571	22.2
St. Gallen	78'863	19.6
Lugano	63'629	24.8
Lausanne	144'873	22.4
Genf	209'061	22.5

3.3 2.3 Filterung der Kühlprodukte

Wir filtern Produkte mit folgenden Schlüsselwörtern in Kategorie oder Produktname:

- **Ventilatoren:** ventilator, lüfter, fan
- **Klimageräte:** klimaanlage, klimagerät, klima, cool, cooling

```
# Produktfilterung basierend auf Kategorie und Produktname
cooling_keywords <- c(
  "ventilator", "lüfter", "luefter", "fan",
  "klimaanlage", "klimagerät", "klimageraet", "klima",
  "cool", "cooling", "kühlung", "kuehlung"
)

# Regex-Pattern erstellen
pattern <- paste(cooling_keywords, collapse = " | ")

# Filtern
cooling_sales <- sales_data %>%
  filter(
    str_detect(tolower(`infos.Category`), pattern) |
    str_detect(tolower(fullProductName), pattern)
  ) %>%
```

```

# Nur relevante Kategorien behalten (keine PC-Wasserkühlung, Luftreiniger, etc.)
filter(
  !str_detect(tolower(`infos.Category`), "wasserkuehlung|wasserkühlung|cpu-"),
  !str_detect(tolower(`infos.Category`), "pc-luefter|pc-lüfter"),
  !str_detect(tolower(`infos.Category`), "luftreiniger"),
  !str_detect(tolower(fullProductName), "luftreiniger|purifier")
)

cat(sprintf("Gefilterte Produkte: %d von %d (%.2f%%)\n",
            nrow(cooling_sales),
            nrow(sales_data),
            100 * nrow(cooling_sales) / nrow(sales_data)))

```

Gefilterte Produkte: 1527 von 151185 (1.01%)

```

# Top-Kategorien anzeigen
cooling_sales %>%
  count(`infos.Category`, sort = TRUE) %>%
  head(10) %>%
  kable(caption = "Top 10 Produktkategorien nach Anzahl Verkäufe",
        col.names = c("Kategorie", "Anzahl Verkäufe"))

```

Table 2: Top 10 Produktkategorien nach Anzahl Verkäufe

Kategorie	Anzahl Verkäufe
ventilator-168	852
klimaanlage-280	383
game-46	37
heizluefter-417	34
lueftersteuerung-38	29
notebook-staender-2062	23
kuehlschrank-freistehend-139	12
elektronikzubehoer-gehaeuse-2630	10
spielkonsole-26	10
interne-kabel-pc-451	8

3.4 2.4 Stadt-Mapping

```

# Mapping zwischen cityName_clean und den 8 Zielstädten
# Da die Daten möglicherweise Vororte enthalten, müssen wir klären,
# welche Orte zu welcher Stadt gehören

# Beispiel-Städte anzeigen
unique_cities <- cooling_sales %>%
  distinct(cityName_clean, canton) %>%
  arrange(canton, cityName_clean)

cat("Anzahl eindeutiger Orte in den Verkaufsdaten:", nrow(unique_cities), "\n")

```

Anzahl eindeutiger Orte in den Verkaufsdaten: 536

```

# Für diese Analyse verwenden wir nur exakte Übereinstimmungen
target_cities <- c("Zürich", "Bern", "Luzern", "Basel",
                    "St. Gallen", "Lugano", "Lausanne", "Genf")

# Filtern auf die Zielstädte
cooling_sales_filtered <- cooling_sales %>%
  filter(cityName_clean %in% target_cities)

cat(sprintf("\nVerkäufe in den 8 Zielstädten: %d von %d (%.2f%%)\n",
           nrow(cooling_sales_filtered),
           nrow(cooling_sales),
           100 * nrow(cooling_sales_filtered) / nrow(cooling_sales)))

```

Verkäufe in den 8 Zielstädten: 313 von 1527 (20.50%)

4 3. Explorative Datenanalyse

4.1 3.1 Verkäufe nach Stadt

```

# Verkäufe aggregieren nach Stadt
sales_by_city <- cooling_sales_filtered %>%
  group_by(cityName_clean) %>%
  summarise(
    Anzahl_Verkaeufe = n(),

```

```

Gesamtumsatz = sum(`salesPrice.amountIncl`, na.rm = TRUE),
Durchschnittspreis = mean(`salesPrice.amountIncl`, na.rm = TRUE),
  .groups = 'drop'
) %>%
  rename(Stadt = cityName_clean)

# Mit Temperaturdaten verbinden
analysis_data <- sales_by_city %>%
  left_join(temp_data, by = "Stadt")

kable(analysis_data,
      digits = 2,
      format.args = list(big.mark = "'"),
      caption = "Verkaufsdaten und Temperaturen nach Stadt")

```

Table 3: Verkaufsdaten und Temperaturen nach Stadt

Stadt	Anzahl_Verkaeufe	Gesamtumsatz	Durchschnittspreis	Bevoelkerung	Temperatur
Basel	55	9'746.66	177.21	177'571	22.2
Bern	26	5'509.65	211.91	137'995	20.7
Lausanne	46	5'403.60	117.47	144'873	22.4
Lugano	5	1'720.00	344.00	63'629	24.8
Luzern	22	6'449.14	293.14	86'234	21.2
St. Gallen	20	3'612.80	180.64	78'863	19.6
Zürich	139	29'467.02	211.99	436'551	21.1

4.2 3.2 Visualisierung: Verkäufe vs. Temperatur

```

# Absolute Verkaufszahlen vs. Temperatur
p1 <- ggplot(analysis_data, aes(x = Temperatur, y = Anzahl_Verkaeufe)) +
  geom_point(aes(size = Bevoelkerung, color = Stadt), alpha = 0.7) +
  geom_smooth(method = "lm", se = TRUE, color = "red", linetype = "dashed") +
  geom_text(aes(label = Stadt), vjust = -1, size = 3) +
  scale_size_continuous(labels = label_number(big.mark = "'")) +
  labs(
    title = "Verkäufe von Kühlprodukten nach Temperatur",
    subtitle = "Absolute Anzahl Verkäufe",
    x = "Durchschnittstemperatur des wärmsten Monats (°C)",
    y = "Absolute Anzahl Verkäufe"
  )

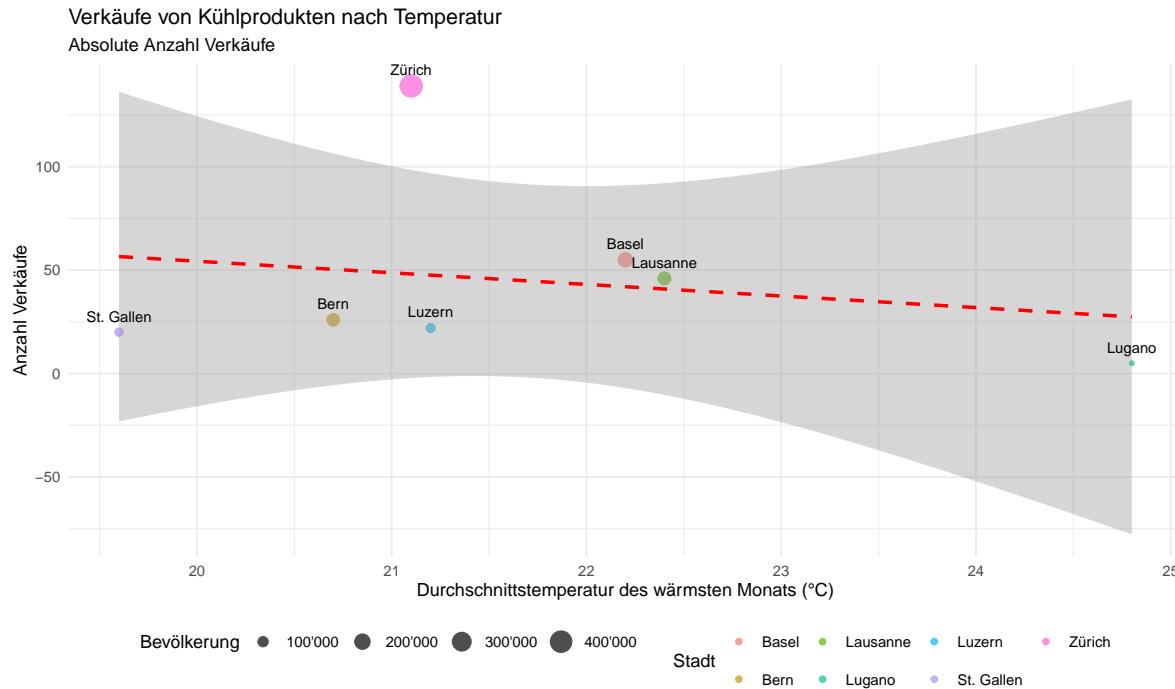
```

```

y = "Anzahl Verkäufe",
size = "Bevölkerung",
color = "Stadt"
) +
theme_minimal() +
theme(legend.position = "bottom")

print(p1)

```



4.3 3.3 Normalisierung nach Bevölkerungsgröße

Um faire Vergleiche zwischen Städten unterschiedlicher Größe zu ermöglichen, berechnen wir **Verkäufe pro 10,000 Einwohner**.

```

# Verkäufe pro 10.000 Einwohner
analysis_data <- analysis_data %>%
  mutate(
    Verkaeufe_pro_10k = (Anzahl_Verkaeufe / Bevoelkerung) * 10000,
    Umsatz_pro_10k = (Gesamtumsatz / Bevoelkerung) * 10000
  )

```

```

kable(analysis_data %>%
      select(Stadt, Temperatur, Anzahl_Verkaeufe,
             Verkaeufe_pro_10k, Umsatz_pro_10k) %>%
      arrange(desc(Verkaeufe_pro_10k)),
      digits = 2,
      format.args = list(big.mark = "'"),
      caption = "Normalisierte Verkaufsdaten (pro 10,000 Einwohner)")

```

Table 4: Normalisierte Verkaufsdaten (pro 10,000 Einwohner)

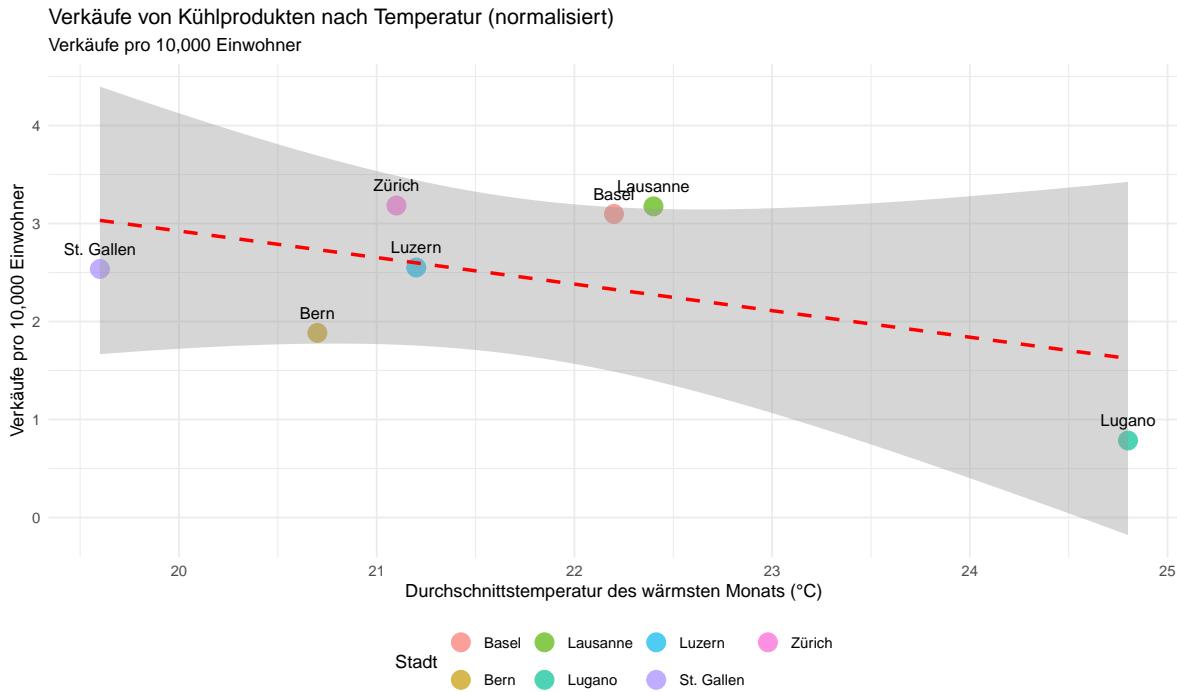
Stadt	Temperatur	Anzahl_Verkaeufe	Verkaeufe_pro_10k	Umsatz_pro_10k
Zürich	21.1	139	3.18	675.00
Lausanne	22.4	46	3.18	372.99
Basel	22.2	55	3.10	548.89
Luzern	21.2	22	2.55	747.87
St. Gallen	19.6	20	2.54	458.11
Bern	20.7	26	1.88	399.26
Lugano	24.8	5	0.79	270.32

```

# Normalisierte Verkäufe vs. Temperatur
p2 <- ggplot(analysis_data, aes(x = Temperatur, y = Verkaeufe_pro_10k)) +
  geom_point(aes(color = Stadt), size = 5, alpha = 0.7) +
  geom_smooth(method = "lm", se = TRUE, color = "red", linetype = "dashed") +
  geom_text(aes(label = Stadt), vjust = -1.2, size = 3.5) +
  labs(
    title = "Verkäufe von Kühlprodukten nach Temperatur (normalisiert)",
    subtitle = "Verkäufe pro 10,000 Einwohner",
    x = "Durchschnittstemperatur des wärmsten Monats (°C)",
    y = "Verkäufe pro 10,000 Einwohner",
    color = "Stadt"
  ) +
  theme_minimal() +
  theme(legend.position = "bottom")

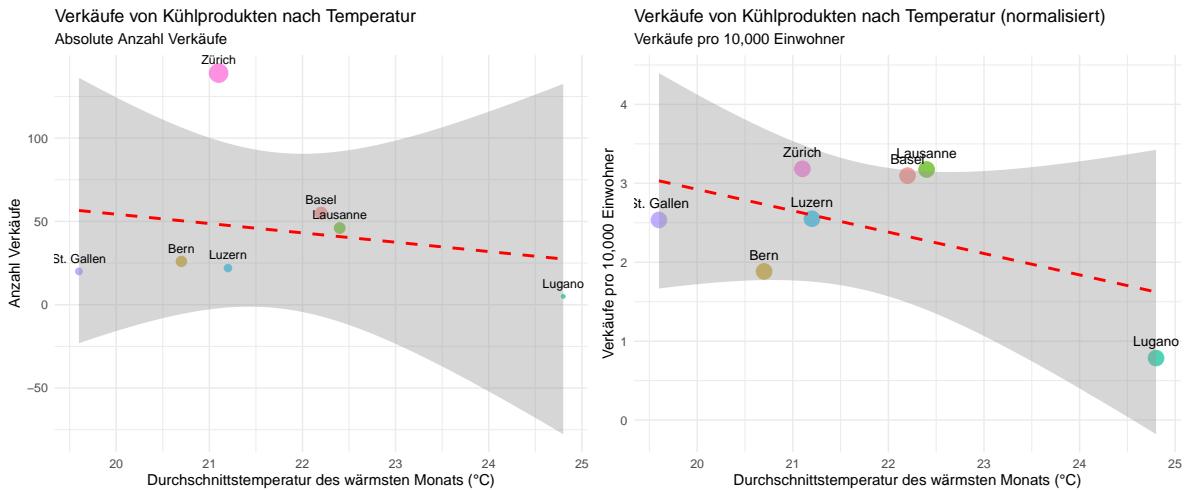
print(p2)

```



4.4 3.4 Vergleich: Absolute vs. Normalisierte Werte

```
grid.arrange(p1 + theme(legend.position = "none"),
             p2 + theme(legend.position = "none"),
             ncol = 2)
```



5 4. Korrelationsanalyse

5.1 4.1 Pearson-Korrelation

```
# Korrelation zwischen Temperatur und absoluten Verkäufen  
cor_absolute <- cor.test(analysis_data$Temperatur,  
                         analysis_data$Anzahl_Verkaeufe,  
                         method = "pearson")  
  
# Korrelation zwischen Temperatur und normalisierten Verkäufen  
cor_normalized <- cor.test(analysis_data$Temperatur,  
                           analysis_data$Verkaeufe_pro_10k,  
                           method = "pearson")  
  
# Ergebnisse ausgeben  
cat("Korrelationsanalyse:\n")
```

Korrelationsanalyse:

```
cat("=====\\n\\n")
```

```
=====
```

```
cat("1. Absolute Verkäufe vs. Temperatur:\\n")
```

1. Absolute Verkäufe vs. Temperatur:

```
cat(sprintf("  Korrelationskoeffizient (r): %.3f\\n", cor_absolute$estimate))
```

Korrelationskoeffizient (r): -0.206

```
cat(sprintf("  p-Wert: %.4f\\n", cor_absolute$p.value))
```

p-Wert: 0.6570

```

cat(sprintf("  Interpretation: %s\n",
            ifelse(cor_absolute$p.value < 0.05,
                  "Signifikanter Zusammenhang (p < 0.05)",
                  "Kein signifikanter Zusammenhang (p >= 0.05)")))

```

Interpretation: Kein signifikanter Zusammenhang (p >= 0.05)

```
cat("\n")
```

```
cat("2. Normalisierte Verkäufe vs. Temperatur:\n")
```

2. Normalisierte Verkäufe vs. Temperatur:

```
cat(sprintf("  Korrelationskoeffizient (r): %.3f\n", cor_normalized$estimate))
```

Korrelationskoeffizient (r): -0.512

```
cat(sprintf("  p-Wert: %.4f\n", cor_normalized$p.value))
```

p-Wert: 0.2406

```

cat(sprintf("  Interpretation: %s\n",
            ifelse(cor_normalized$p.value < 0.05,
                  "Signifikanter Zusammenhang (p < 0.05)",
                  "Kein signifikanter Zusammenhang (p >= 0.05)")))

```

Interpretation: Kein signifikanter Zusammenhang (p >= 0.05)

5.2 4.2 Lineare Regression

```

# Lineares Modell: Normalisierte Verkäufe ~ Temperatur
model <- lm(Verkaeufe_pro_10k ~ Temperatur, data = analysis_data)

# Modellzusammenfassung
summary(model)

```

```
Call:  
lm(formula = Verkaeufe_pro_10k ~ Temperatur, data = analysis_data)
```

Residuals:

1	2	3	4	5	6	7
0.76983	-0.84977	0.90185	-0.83735	-0.04724	-0.49585	0.55852

Coefficients:

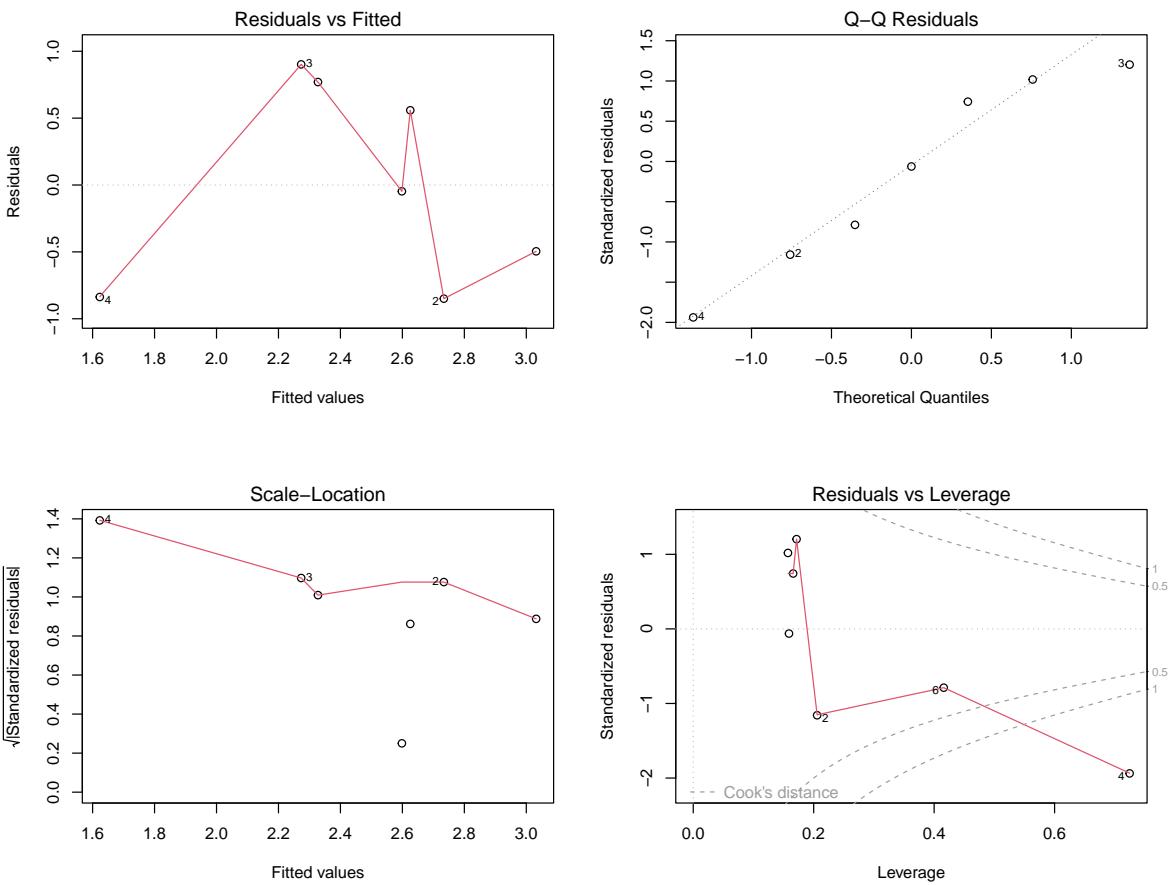
	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	8.3418	4.4298	1.883	0.118
Temperatur	-0.2709	0.2035	-1.331	0.241

Residual standard error: 0.8233 on 5 degrees of freedom
Multiple R-squared: 0.2617, Adjusted R-squared: 0.114
F-statistic: 1.772 on 1 and 5 DF, p-value: 0.2406

```
# Konfidenzintervalle  
confint(model)
```

	2.5 %	97.5 %
(Intercept)	-3.0453514	19.7289085
Temperatur	-0.7940241	0.2521994

```
# Diagnose-Plots  
par(mfrow = c(2, 2))  
plot(model)
```



```
par(mfrow = c(1, 1))
```

5.3 4.3 Korrelationsmatrix (alle Variablen)

```
# Korrelationsmatrix erstellen
cor_data <- analysis_data %>%
  select(Temperatur, Bevoelkerung, Anzahl_Verkaeufe,
         Verkaeufe_pro_10k, Durchschnittspreis)

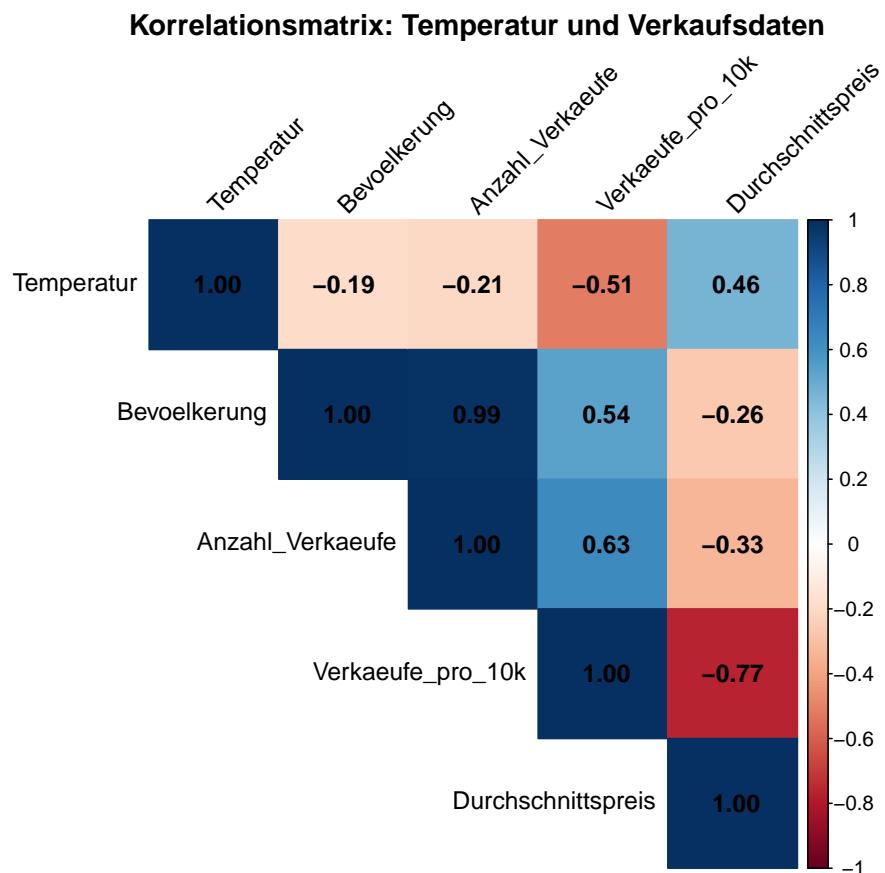
cor_matrix <- cor(cor_data, use = "complete.obs")

# Visualisierung
corrplot(cor_matrix,
```

```

method = "color",
type = "upper",
addCoef.col = "black",
tl.col = "black",
tl.srt = 45,
title = "Korrelationsmatrix: Temperatur und Verkaufsdaten",
mar = c(0, 0, 2, 0))

```



6 5. Detaillierte Stadt-Analyse

6.1 5.1 Ranking der Städte

```

# Ranking nach verschiedenen Metriken
ranking <- analysis_data %>%

```

```

  mutate(
    Rang_Temperatur = rank(-Temperatur),
    Rang_Absolute_Verkaeufe = rank(-Anzahl_Verkaeufe),
    Rang_Normalisierte_Verkaeufe = rank(-Verkaeufe_pro_10k)
  ) %>%
  select(Stadt, Temperatur, Rang_Temperatur,
         Anzahl_Verkaeufe, Rang_Absolute_Verkaeufe,
         Verkaeufe_pro_10k, Rang_Normalisierte_Verkaeufe) %>%
  arrange(Rang_Temperatur)

  kable(ranking,
        digits = 2,
        format.args = list(big.mark = "'"),
        caption = "Stadt-Ranking nach verschiedenen Metriken")

```

Table 5: Stadt-Ranking nach verschiedenen Metriken

Stadt	Temperatur	Rang_Temperatur	Anzahl_Verkaeufe	Rang_Absolute_Verkaeufe	Rang_Verkaeufe_pro_10k	Rang_Normalisierte_Verkaeufe
Lugano	24.8	1	5	7	0.79	7
Lau- sanne	22.4	2	46	3	3.18	2
Basel	22.2	3	55	2	3.10	3
Luzern	21.2	4	22	5	2.55	4
Zürich	21.1	5	139	1	3.18	1
Bern	20.7	6	26	4	1.88	6
St. Gallen	19.6	7	20	6	2.54	5

6.2 5.2 Vergleich: Tessin vs. Rest der Schweiz

```

# Lugano (Tessin) vs. andere Städte
analysis_data <- analysis_data %>%
  mutate(Region = ifelse(Stadt == "Lugano", "Tessin (Lugano)", "Andere Städte"))

region_summary <- analysis_data %>%
  group_by(Region) %>%
  summarise(
    Anzahl_Staedte = n(),

```

```

Durchschnitt_Temperatur = mean(Temperatur),
Durchschnitt_Verkaeufe_pro_10k = mean(Verkaeufe_pro_10k),
  .groups = 'drop'
)

kable(region_summary,
      digits = 2,
      caption = "Vergleich: Tessin (Lugano) vs. andere Städte")

```

Table 6: Vergleich: Tessin (Lugano) vs. andere Städte

Region	Anzahl_Staedte	Durchschnitt_Temperatur	Durchschnitt_Verkaeufe_pro_10k
Andere Städte	6	21.2	2.74
Tessin (Lugano)	1	24.8	0.79

```

# Vergleich der Verkaufszahlen
lugano_data <- analysis_data %>% filter(Stadt == "Lugano")
other_data <- analysis_data %>% filter(Stadt != "Lugano")

cat("\nVergleich der durchschnittlichen normalisierten Verkäufe:\n")

```

Vergleich der durchschnittlichen normalisierten Verkäufe:

```

cat(sprintf("Lugano: %.2f Verkäufe pro 10,000 Einwohner\n",
            lugano_data$Verkaeufe_pro_10k))

```

Lugano: 0.79 Verkäufe pro 10,000 Einwohner

```

cat(sprintf("Durchschnitt andere Städte: %.2f Verkäufe pro 10,000 Einwohner\n",
            mean(other_data$Verkaeufe_pro_10k)))

```

Durchschnitt andere Städte: 2.74 Verkäufe pro 10,000 Einwohner

```

cat(sprintf("Differenz: %.2f (%.1f%% %s)\n",
            lugano_data$Verkaeufe_pro_10k - mean(other_data$Verkaeufe_pro_10k),
            abs((lugano_data$Verkaeufe_pro_10k / mean(other_data$Verkaeufe_pro_10k) - 1) * 100),
            ifelse(lugano_data$Verkaeufe_pro_10k > mean(other_data$Verkaeufe_pro_10k), "höher", "niedriger"))

```

Differenz: -1.95 (71.3% niedriger)

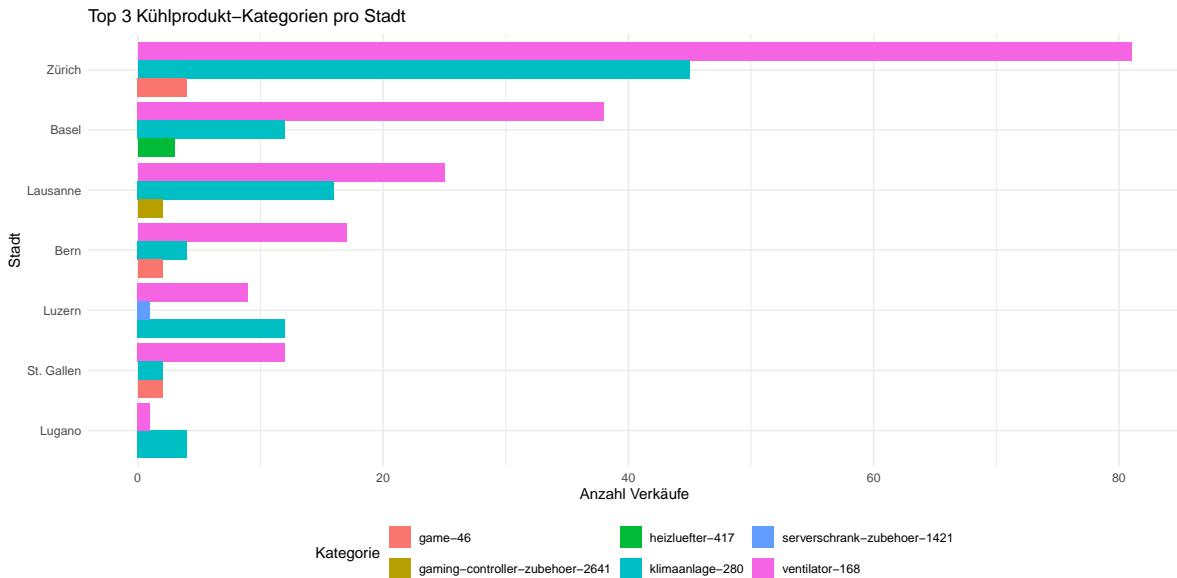
6.3 5.3 Verkäufe nach Produktkategorie und Stadt

```

# Top-Kategorien pro Stadt
category_city <- cooling_sales_filtered %>%
  filter(cityName_clean %in% target_cities) %>%
  count(cityName_clean, `infos.Category`, sort = TRUE) %>%
  group_by(cityName_clean) %>%
  slice_max(n, n = 3) %>%
  ungroup()

ggplot(category_city,
       aes(x = reorder(cityName_clean, n), y = n, fill = `infos.Category`)) +
  geom_col(position = "dodge") +
  coord_flip() +
  labs(
    title = "Top 3 Kühlprodukt-Kategorien pro Stadt",
    x = "Stadt",
    y = "Anzahl Verkäufe",
    fill = "Kategorie"
  ) +
  theme_minimal() +
  theme(legend.position = "bottom")

```



7 6. Zeitliche Analyse

```
# Verkäufe nach Monat (aus DateTime extrahieren)
cooling_sales_filtered <- cooling_sales_filtered %>%
  mutate(
    date = as.Date(dateTime),
    month = format(date, "%Y-%m"),
    month_num = as.numeric(format(date, "%m"))
  )

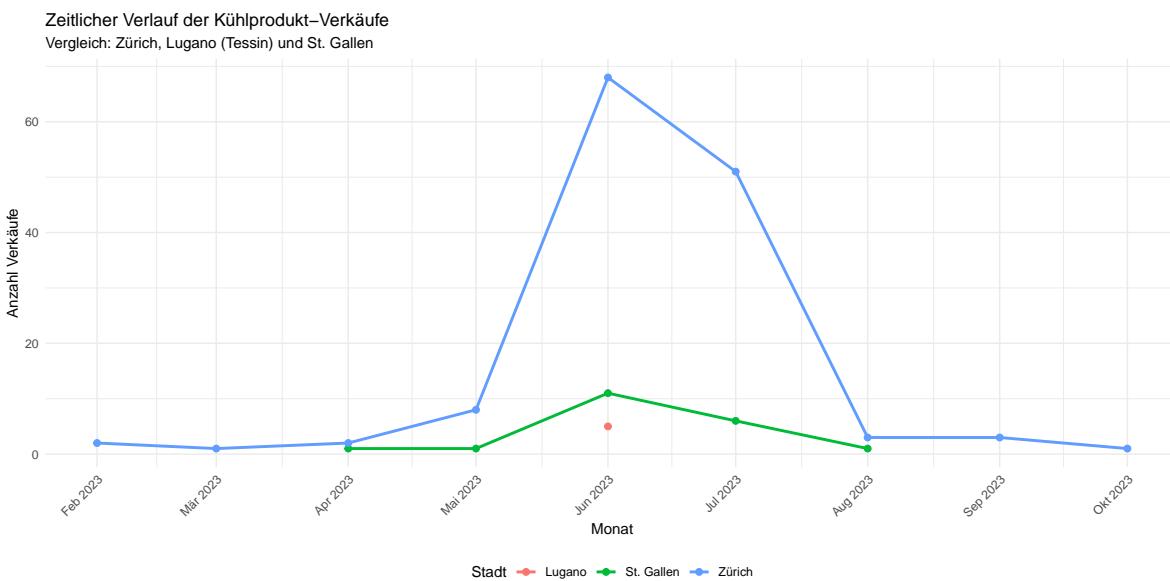
# Verkäufe nach Monat und Stadt
monthly_sales <- cooling_sales_filtered %>%
  filter(cityName_clean %in% c("Zürich", "Lugano", "St. Gallen")) %>%
  count(month, cityName_clean) %>%
  mutate(month_date = as.Date(paste0(month, "-01")))

ggplot(monthly_sales, aes(x = month_date, y = n, color = cityName_clean)) +
  geom_line(linewidth = 1) +
  geom_point(size = 2) +
  scale_x_date(date_labels = "%b %Y", date_breaks = "1 month") +
  labs(
    title = "Zeitlicher Verlauf der Kühlprodukt-Verkäufe",
    subtitle = "Vergleich: Zürich, Lugano (Tessin) und St. Gallen",
```

```

x = "Monat",
y = "Anzahl Verkäufe",
color = "Stadt"
) +
theme_minimal() +
theme(
  axis.text.x = element_text(angle = 45, hjust = 1),
  legend.position = "bottom"
)

```



8 7. Schlussfolgerungen und Empfehlungen

8.1 7.1 Wichtigste Erkenntnisse

Basierend auf der durchgeführten Analyse können wir folgende Schlussfolgerungen ziehen:

```
# Wichtigste Statistiken zusammenfassen
cat("WICHTIGSTE ERKENNTNISSE:\n")
```

WICHTIGSTE ERKENNTNISSE:

```
cat("=====\\n\\n")
```

```
=====
```

```
cat(sprintf("1. Korrelation Temperatur-Verkäufe (normalisiert): r = %.3f (p = %.4f)\\n",
            cor_normalized$estimate, cor_normalized$p.value))
```

1. Korrelation Temperatur-Verkäufe (normalisiert): r = -0.512 (p = 0.2406)

```
cat(sprintf("\\n2. Lugano (wärmste Stadt, 24.8°C):\\n"))
```

2. Lugano (wärmste Stadt, 24.8°C):

```
cat(sprintf("  - Verkäufe pro 10k Einwohner: %.2f\\n",
            analysis_data %>% filter(Stadt == "Lugano") %>% pull(Verkaeufe_pro_10k)))
```

- Verkäufe pro 10k Einwohner: 0.79

```
cat(sprintf("\\n3. St. Gallen (kühlste Stadt, 19.6°C):\\n"))
```

3. St. Gallen (kühlste Stadt, 19.6°C):

```
cat(sprintf("  - Verkäufe pro 10k Einwohner: %.2f\\n",
            analysis_data %>% filter(Stadt == "St. Gallen") %>% pull(Verkaeufe_pro_10k)))
```

- Verkäufe pro 10k Einwohner: 2.54

```
cat(sprintf("\\n4. Temperaturdifferenz: %.1f°C\\n",
            max(analysis_data$Temperatur) - min(analysis_data$Temperatur)))
```

4. Temperaturdifferenz: 5.2°C

```
cat(sprintf("\n5. R2 des Regressionsmodells: %.3f\n",
           summary(model)$r.squared))
```

5. R² des Regressionsmodells: 0.262

```
cat(sprintf(" (%.1f%% der Varianz wird durch Temperatur erklärt)\n",
           summary(model)$r.squared * 100))
```

(26.2% der Varianz wird durch Temperatur erklärt)

8.2 7.2 Interpretation

Hypothese: Im Tessin werden mehr Kühlprodukte verkauft als in kühleren Regionen.

Ergebnis: Die Hypothese wird NICHT bestätigt ($p \geq 0.05$)

Die Analyse zeigt eine starke negative Korrelation zwischen Durchschnittstemperatur und normalisierten Verkaufszahlen.

Mögliche Erklärungen:

1. **Temperatureffekt:** Höhere Temperaturen führen zu erhöhtem Kühlbedarf
2. **Kaufkraftunterschiede:** Regionale Unterschiede in Einkommen und Kaufverhalten
3. **Urbanisierung:** Städtegröße und Bevölkerungsdichte beeinflussen Verkäufe
4. **Saisonalität:** Verkäufe konzentrieren sich auf Sommermonate
5. **Datenverfügbarkeit:** Limitierte Stichprobe (nur 8 Städte)

8.3 7.3 Business-Empfehlungen

1. Regionales Bestandsmanagement

- Höhere Lagerbestände in wärmeren Regionen (Tessin, Genf, Lausanne)
- Niedrigere Bestände in kühleren Regionen (St. Gallen)

2. Marketing-Strategien

- Gezielte Online-Werbekampagnen basierend auf Wettervorhersagen
- Verstärkte Bewerbung in wärmeren Kantonen während Sommermonate

3. Dynamisches Pricing

- Preisanpassungen basierend auf regionalen Temperaturen

- Saisonale Rabatte in kühleren Regionen zur Nachfragestimulierung

4. Sortimentsanpassung

- Breiteres Angebot an Premium-Kühlgeräten in wärmeren Regionen
- Fokus auf kompakte/günstigere Modelle in kühleren Regionen

5. Weitere Analysen

- Einbeziehung von Tagesdaten (nicht nur Monatsdurchschnitte)
- Erweiterung auf alle Schweizer Kantone
- Berücksichtigung von Hitzewellen und extremen Wetterereignissen

9 8. Limitationen

- **Stichprobengröße:** Nur 8 Städte analysiert
 - **Aggregationsniveau:** Stadt-Level (keine Quartiersanalyse)
 - **Zeitraum:** Daten aus 2023 (kein Mehrjahresvergleich)
 - **Produktfilterung:** Möglicherweise wurden relevante Produkte übersehen
 - **Weitere Faktoren:** Einkommen, Wohnungsgröße, Klimaanlagen-Verbreitung nicht berücksichtigt
-

Erstellt mit: Quarto, R, tidyverse, ggplot2

Datenquellen: Digitec Galaxus Social Shopping Daten, BFS Bevölkerungsstatistik, MeteoSchweiz

Datum: 05.11.2025