

# Klimabedingte Verkaufsanalyse: Ventilatoren und Kühleräte

**Einfluss der Temperatur auf Produktverkäufe in Schweizer Städten**

## 1 Executive Summary

Diese Analyse untersucht den Zusammenhang zwischen Durchschnittstemperatur und Verkaufszahlen von Ventilatoren und Kühleräten 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 Kühleräte 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("Temperatur 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, Gaming, Heizlüf
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"),
  !str_detect(tolower(`infos.Category`), "game-|gaming-controller-zubehoer|heizluefter|hei
)
cat(sprintf("Gefilterte Produkte: %d von %d (%.2f%%)\n",
            nrow(cooling_sales),
            nrow(sales_data),
            100 * nrow(cooling_sales) / nrow(sales_data)))

```

Gefilterte Produkte: 1448 von 151185 (0.96%)

```

# 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
lueftersteuerung-38	29
notebook-staender-2062	23
kuehlschrank-freistehend-139	12
elektronikzubehoer-gehaeuse-2630	10
spielkonsole-26	10
interne-kabel-pc-451	8

Kategorie	Anzahl Verkäufe
luftbehandlung-zubehoer-653	8
modding-sleevring-2261	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: 517

```
# 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: 296 von 1448 (20.44%)

## 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	51	9'190.76	180.21	177'571	22.2
Bern	24	5'349.85	222.91	137'995	20.7
Lausanne	43	5'168.10	120.19	144'873	22.4
Lugano	5	1'720.00	344.00	63'629	24.8
Luzern	21	6'261.19	298.15	86'234	21.2
St. Gallen	18	3'537.20	196.51	78'863	19.6
Zürich	134	29'090.92	217.10	436'551	21.1

### 4.2 3.2 Visualisierung: Verkäufe vs. Temperatur

```

# Hauptvisualisierung wird in Abschnitt 3.3 gezeigt (normalisierte Werte)
# Hier nur zur Referenz die Rohdaten
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 (Absolute Zahlen)",
    subtitle = "Hinweis: Größere Städte haben naturgemäß mehr absolute Verkäufe",
    x = "Durchschnittstemperatur des wärmsten Monats (°C)",
    y = "Anzahl Verkäufe (absolut)",
    size = "Bevölkerung",
    color = "Stadt"
  ) +
  theme_minimal() +
  theme(legend.position = "bottom")

# Plot wird nicht direkt angezeigt, nur für Vergleich gespeichert

```

### 4.3 3.3 Normalisierung nach Bevölkerungsgröße

Um faire Vergleiche zwischen Städten unterschiedlicher Größe zu ermöglichen, berechnen wir **Verkäufe pro 5,000 Einwohner**.

```

# Verkäufe pro 5.000 Einwohner
analysis_data <- analysis_data %>%
  mutate(
    Verkaeufe_pro_5k = (Anzahl_Verkaeufe / Bevoelkerung) * 5000,
    Umsatz_pro_5k = (Gesamtumsatz / Bevoelkerung) * 5000
  )

kable(analysis_data %>%
  select(Stadt, Temperatur, Bevoelkerung,
         Verkaeufe_pro_5k, Umsatz_pro_5k) %>%
  arrange(desc(Verkaeufe_pro_5k)),
  digits = 2,
  format.args = list(big.mark = '')),
  caption = "Normalisierte Verkaufsdaten (pro 5,000 Einwohner)")

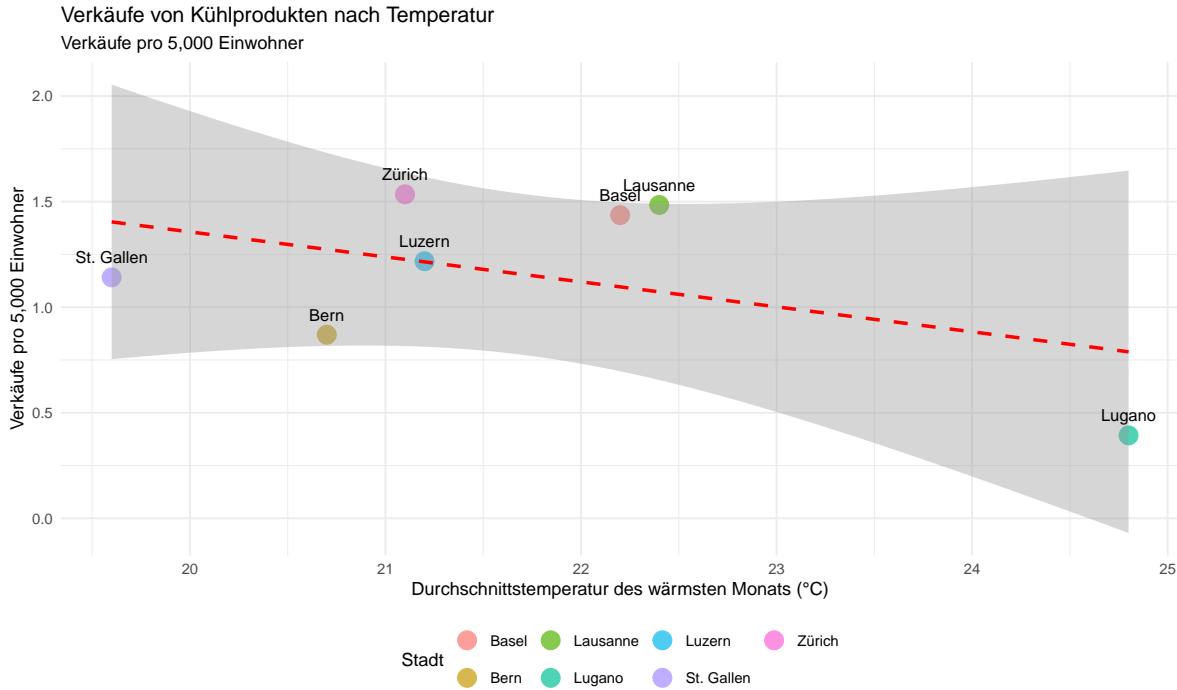
```

Table 4: Normalisierte Verkaufsdaten (pro 5,000 Einwohner)

Stadt	Temperatur	Bevoelkerung	Verkaeufe_pro_5k	Umsatz_pro_5k
Zürich	21.1	436'551	1.53	333.19
Lausanne	22.4	144'873	1.48	178.37
Basel	22.2	177'571	1.44	258.79
Luzern	21.2	86'234	1.22	363.03
St. Gallen	19.6	78'863	1.14	224.26
Bern	20.7	137'995	0.87	193.84
Lugano	24.8	63'629	0.39	135.16

```
# Normalisierte Verkäufe vs. Temperatur
p2 <- ggplot(analysis_data, aes(x = Temperatur, y = Verkaeufe_pro_5k)) +
  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",
    subtitle = "Verkäufe pro 5,000 Einwohner",
    x = "Durchschnittstemperatur des wärmsten Monats (°C)",
    y = "Verkäufe pro 5,000 Einwohner",
    color = "Stadt"
  ) +
  theme_minimal() +
  theme(legend.position = "bottom")

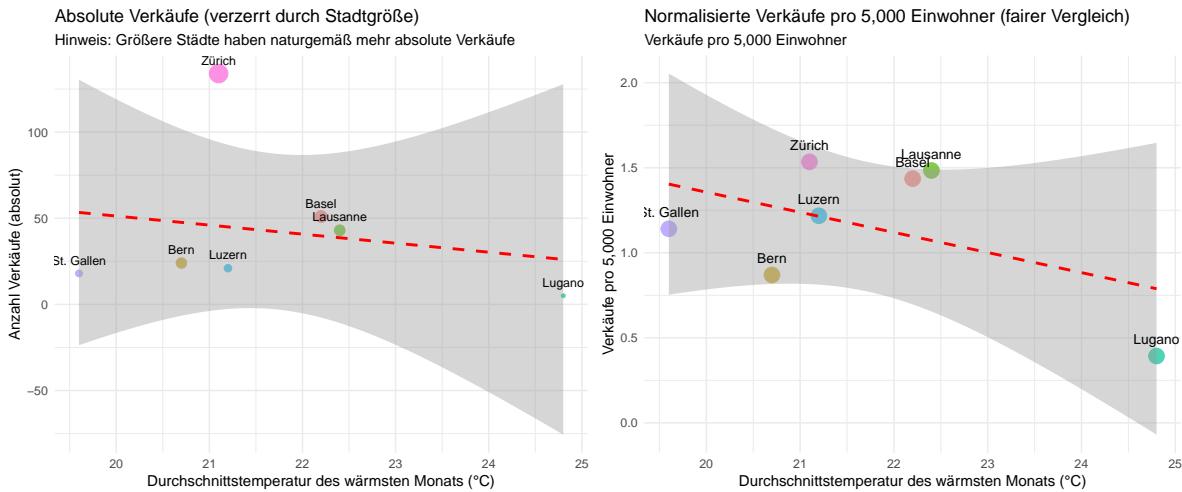
print(p2)
```



#### 4.4 3.4 Vergleich: Absolute vs. Normalisierte Werte

Dieser Vergleich zeigt, warum die Normalisierung wichtig ist:

```
grid.arrange(p1 + theme(legend.position = "none") +
             ggttitle("Absolute Verkäufe (verzerrt durch Stadtgröße)") ,
             p2 + theme(legend.position = "none") +
             ggttitle("Normalisierte Verkäufe pro 5,000 Einwohner (fairer Vergleich)") ,
             ncol = 2)
```



**Interpretation:** Die normalisierten Werte (rechts) ermöglichen einen fairen Vergleich zwischen Städten unterschiedlicher Größe.

## 5 4. Korrelationsanalyse

### 5.1 4.1 Pearson-Korrelation

```
# Korrelation zwischen Temperatur und normalisierten Verkäufen
cor_normalized <- cor.test(analysis_data$Temperatur,
                           analysis_data$Verkäufe_pro_5k,
                           method = "pearson")

# Ergebnisse ausgeben
cat("Korrelationsanalyse:\n")
```

Korrelationsanalyse:

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

```
=====
```

```
cat("Normalisierte Verkäufe (pro 5,000 Einwohner) vs. Temperatur:\\n")
```

Normalisierte Verkäufe (pro 5,000 Einwohner) vs. Temperatur:

```

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

Korrelationskoeffizient (r): -0.479

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

p-Wert: 0.2767

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 ( $p \geq 0.05$ )

## 5.2 4.2 Lineare Regression

```

# Lineares Modell: Normalisierte Verkäufe ~ Temperatur
model <- lm(Verkaeufe_pro_5k ~ Temperatur, data = analysis_data)

# Modellzusammenfassung
summary(model)

```

Call:

```
lm(formula = Verkaeufe_pro_5k ~ Temperatur, data = analysis_data)
```

Residuals:

1	2	3	4	5	6	7
0.339761	-0.404132	0.411433	-0.395813	0.003037	-0.262634	0.308348

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	3.72245	2.10973	1.764	0.138
Temperatur	-0.11830	0.09692	-1.221	0.277

Residual standard error: 0.3921 on 5 degrees of freedom

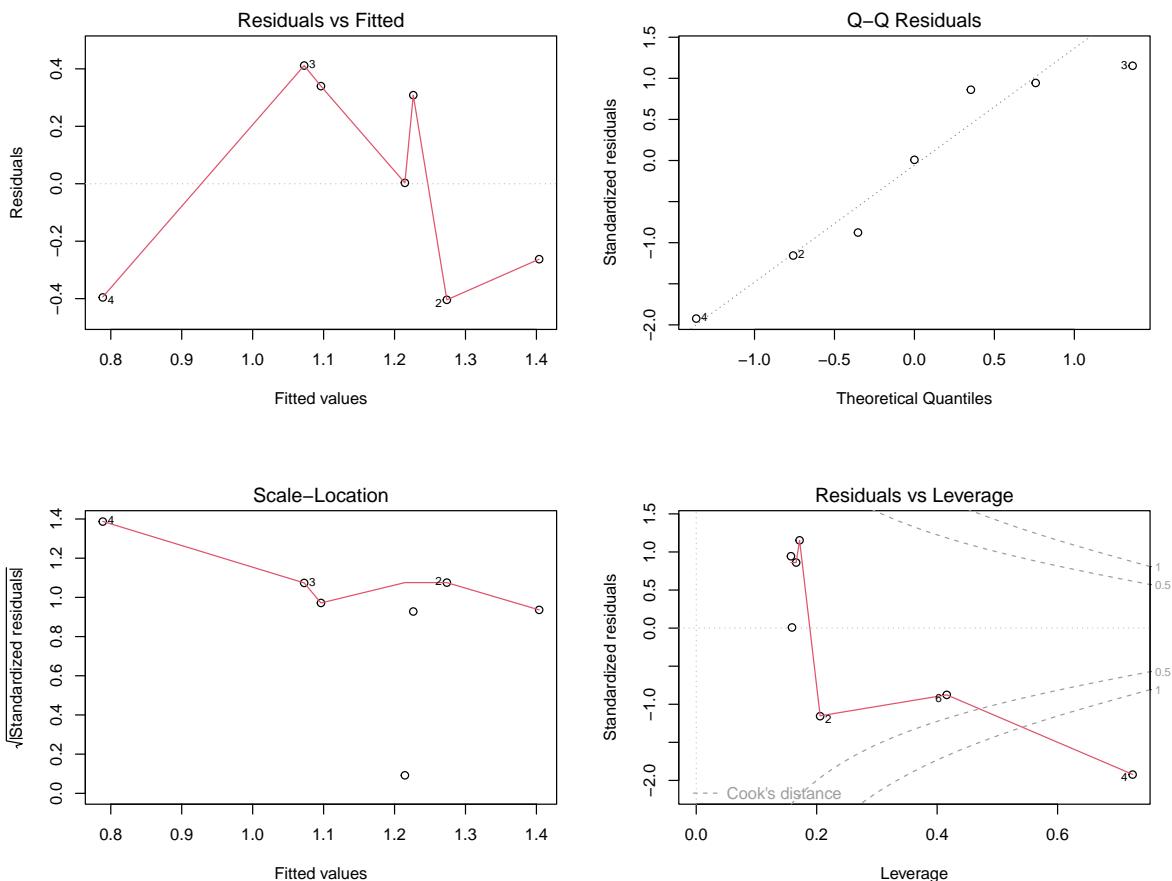
Multiple R-squared: 0.2296, Adjusted R-squared: 0.07547

F-statistic: 1.49 on 1 and 5 DF, p-value: 0.2767

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

2.5 % 97.5 %  
(Intercept) -1.7007868 9.1456905  
Temperatur -0.3674332 0.1308416

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



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

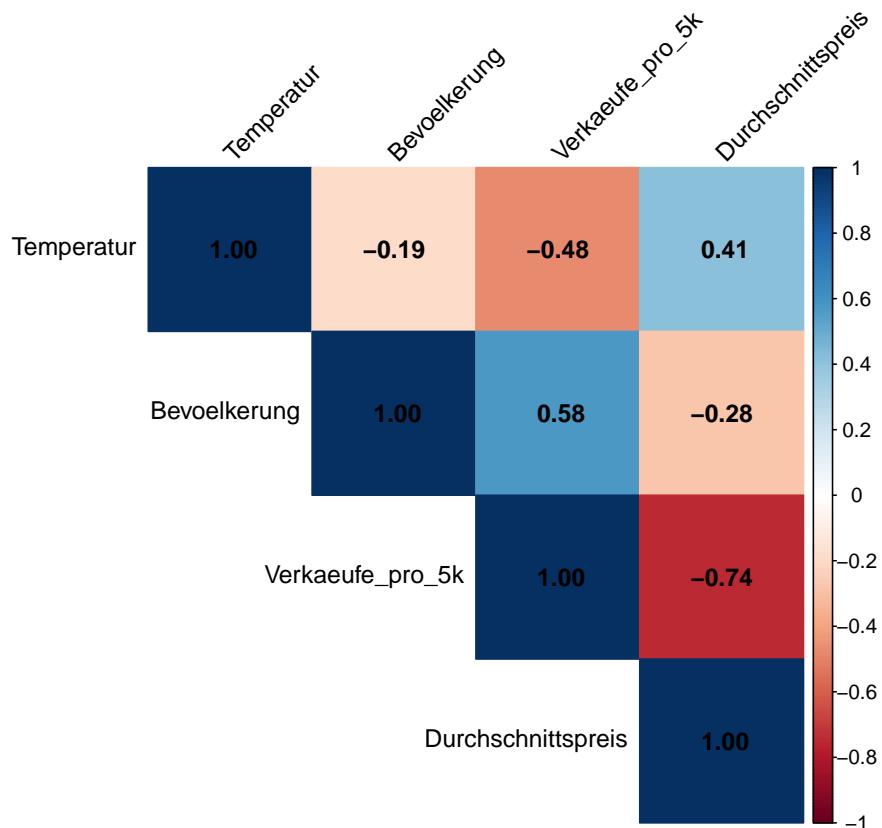
### 5.3 4.3 Korrelationsmatrix (alle Variablen)

```
# Korrelationsmatrix erstellen
cor_data <- analysis_data %>%
  select(Temperatur, Bevoelkerung,
         Verkaeufe_pro_5k, 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))
```

**Korrelationsmatrix: Temperatur und Verkaufsdaten**



## 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_Normalisierte_Verkaeufe = rank(-Verkaeufe_pro_5k)
  ) %>%
  select(Stadt, Temperatur, Rang_Temperatur,
         Verkaeufe_pro_5k, Rang_Normalisierte_Verkaeufe) %>%
  arrange(Rang_Temperatur)
```

```

kable(ranking,
  digits = 2,
  format.args = list(big.mark = "'"),
  col.names = c("Stadt", "Temperatur (°C)", "Rang Temp.",
               "Verkäufe pro 5k", "Rang Verkäufe"),
  caption = "Stadt-Ranking nach Temperatur und normalisierten Verkäufen")

```

Table 5: Stadt-Ranking nach Temperatur und normalisierten Verkäufen

Stadt	Temperatur (°C)	Rang Temp.	Verkäufe pro 5k	Rang Verkäufe
Lugano	24.8	1	0.39	7
Lausanne	22.4	2	1.48	2
Basel	22.2	3	1.44	3
Luzern	21.2	4	1.22	4
Zürich	21.1	5	1.53	1
Bern	20.7	6	0.87	6
St. Gallen	19.6	7	1.14	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_5k = mean(Verkaeufe_pro_5k),
    .groups = 'drop'
  )

kable(region_summary,
  digits = 2,
  col.names = c("Region", "Anzahl Städte", "Ø Temperatur (°C)",
               "Ø Verkäufe pro 5k"),
  caption = "Vergleich: Tessin (Lugano) vs. andere Städte")

```

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

Region	Anzahl Städte	$\bar{\varnothing}$ Temperatur (°C)	$\bar{\varnothing}$ Verkäufe pro 5k
Andere Städte	6	21.2	1.28
Tessin (Lugano)	1	24.8	0.39

```
# 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 5,000 Einwohner\n",
            lugano_data$Verkaeufe_pro_5k))
```

Lugano: 0.39 Verkäufe pro 5,000 Einwohner

```
cat(sprintf("Durchschnitt andere Städte: %.2f Verkäufe pro 5,000 Einwohner\n",
            mean(other_data$Verkaeufe_pro_5k)))
```

Durchschnitt andere Städte: 1.28 Verkäufe pro 5,000 Einwohner

```
cat(sprintf("Differenz: %.2f (%.1f%% %s)\n",
            lugano_data$Verkaeufe_pro_5k - mean(other_data$Verkaeufe_pro_5k),
            abs((lugano_data$Verkaeufe_pro_5k / mean(other_data$Verkaeufe_pro_5k) - 1) * 100),
            ifelse(lugano_data$Verkaeufe_pro_5k > mean(other_data$Verkaeufe_pro_5k), "höher"))
```

Differenz: -0.89 (69.3% niedriger)

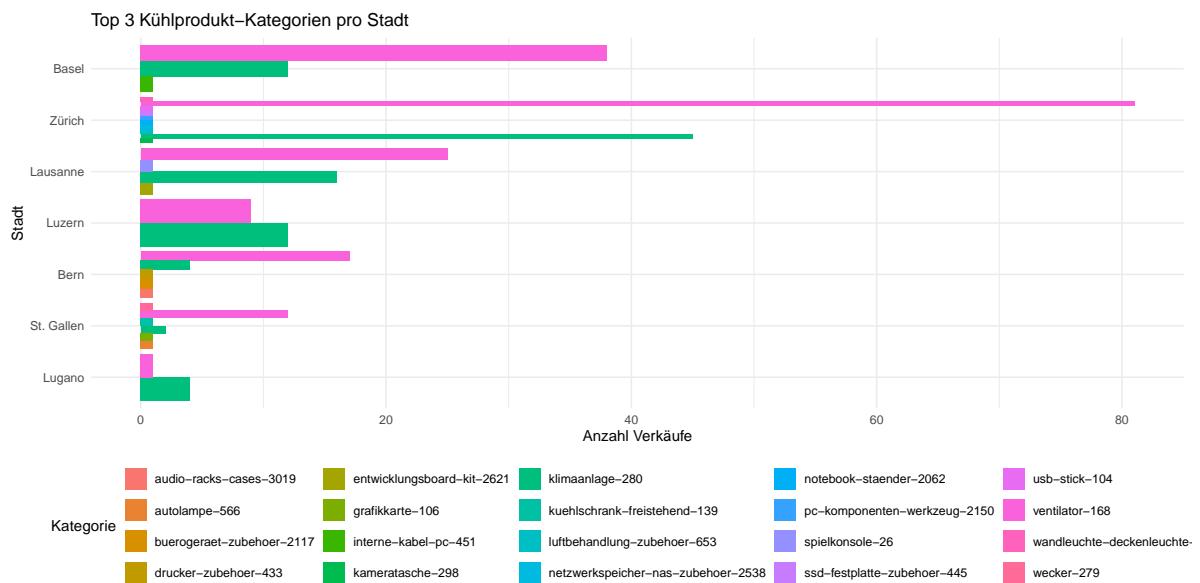
### 6.3 5.3 Verkäufe nach Produktkategorie und Stadt

```

# 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")

```

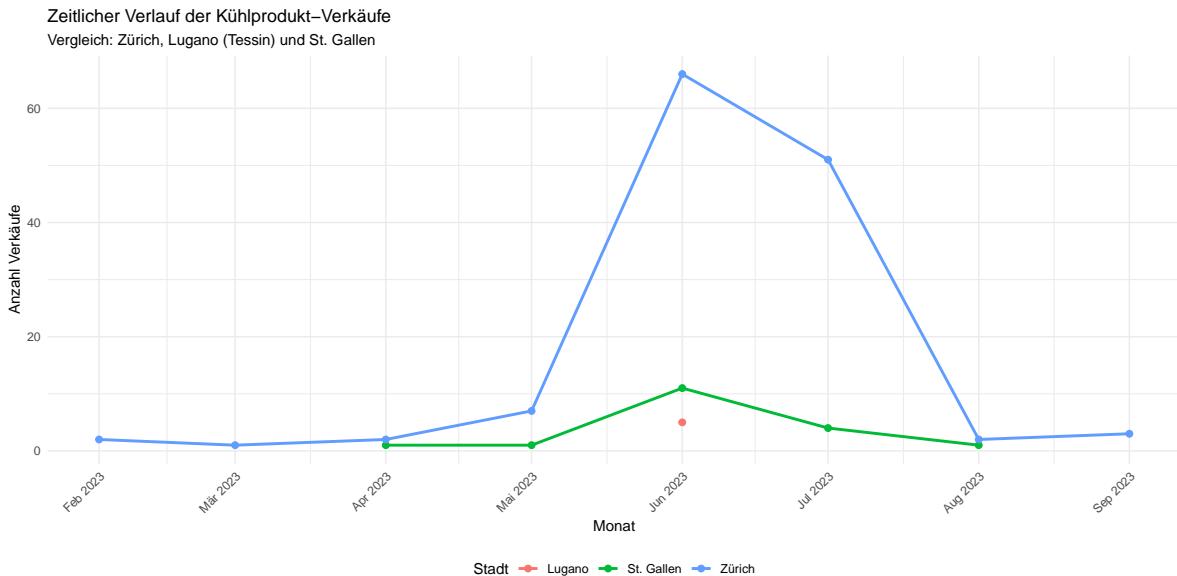


## 7 6. Zeitliche Analyse

```
# 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 durchgeföhrten 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 = -0.479 (p = 0.2767)

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

2. Lugano (wärmste Stadt, 24.8°C):

```
cat(sprintf(" - Verkäufe pro 5k Einwohner: %.2f\n",
            analysis_data %>% filter(Stadt == "Lugano") %>% pull(Verkaeufe_pro_5k)))
```

- Verkäufe pro 5k Einwohner: 0.39

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

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

```
cat(sprintf(" - Verkäufe pro 5k Einwohner: %.2f\n",
            analysis_data %>% filter(Stadt == "St. Gallen") %>% pull(Verkaeufe_pro_5k)))
```

- Verkäufe pro 5k Einwohner: 1.14

```
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. R<sup>2</sup> des Regressionsmodells: 0.230

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

(23.0% 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 moderate 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)
- 

**Erstellt mit:** Quarto, R, tidyverse, ggplot2

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