Report on Swiss hotel trends

Kenny Trinh Sabin Pun

2024-08-27

Contents

1 Introduction

1.1 Background

Switzerland is one of the favorite destinations for tourism around the world. Tourism plays a great role in the Swiss economy by offering GDP to the nation, opening job opportunities, and shaping the cultural relationship between Switzerland and the rest of the world. The country is renowned for its panoramas, world-class winter sports, scenic summer hiking routes, and bustling cities like Zurich, Geneva, and Lucerne. Each of Switzerland's 26 cantons boasts special attractions which invites tourists for business and vacations.

One of the key indicators of tourism activity is overnight stays. By tracking overnight stays, it is possible to know when and where tourists stay. This can be quite useful for analyzing seasonal peaks, nationality preferences, and the demand for tourism in particular regions. Moreover, tourism in Switzerland is heavily seasonal and dependent on good weather conditions; this is particularly true during winter, as snow-dependent activities such as skiing and snowboarding attract big crowds.

This study explores Swiss tourism patterns by looking at overnight stays in different cantons from 2018 to 2023. By combining this information with weather conditions(like temperature, rainfall, and snowfall), this analysis aims to give a better insight into what drives tourist overnight stay and how the other factors influence their stay across different seasons and cantons.

1.2 Objectives of the Analysis

The main goals of this analysis is to give insight into the following core objectives:

1. To explore foreign tourist preferences:
    - Check out when tourist from the top countries like to visit and their top preferred canton for overnight stays.
2. Exploring seasonal tourism patterns:
    - Spot patterns in different seasons for overnight stays and how they differ between cantons.
3. To examine how weather affects overnight stays
    - To examine if weather conditions such as temperature, rainfall and snowfall have measurable effects on overnight stays

By meeting these goals, this study will give a complete picture of what drives tourism in Switzerland and how these drivers change across regions, seasons, and where visitors come from.

2 Setup

This section ensures that all required R packages are installed and loaded, and the directories necessary for input and output data are created.

2.1 Package Setup

A vector listing was created for all the packages that is required in our analysis. The packages' vector consists of:

- tidyverse: A set of R packages designed for data science including dplyr, ggplot2, readr, purrr, and more, that work in harmony to make coding for data analysis easier
- lubridate: So-named because this package makes dates and times easy to work with.
- writexl: Give an easy export of data into Excel files.
- scales: Give additional scaling functions, which is very useful in visualizations.

Since it is a bit of a drag to have to check that all packages are available, the function install_if_missing() was written which for each package checks if it is installed. If the package isn't installed, then it installs it. Once all the packages are guaranteed to be available, then they are loaded into the R environment.

The analysis will be reproducible on different machines, and it will not be subject to errors related to packages.

```
knitr::opts_chunk$set(echo = FALSE, fig.width = 8, fig.height = 3, fig.align = "center" )

# List of required packages
required_packages <- c(
  "tidyverse", "lubridate", "dplyr", "readr", "tidyr", "stringr", "purrr",
  "scales", "ggplot2", "writexl", "plotly", "tidytext", "tinytex", "xfun"
)

# Function to check and install missing packages
install_if_missing <- function(packages) {
  missing_packages <- packages[!packages %in% installed.packages()[,"Package"]]
  if (length(missing_packages) > 0) {
    install.packages(missing_packages)
  }
}

# Install any missing packages
install_if_missing(required_packages)

# Load all packages
lapply(required_packages, library, character.only = TRUE)
```

2.2 Data Paths Setup

Now that the necessary packages are available within our environment, the various paths for files was defined where data is to be stored or obtained. For the project at hand:

- data_dir: Path to the input data.
- output_dir: Path to store processed data and output files.

If output_dir does not exist, it is created via the dir.create() function. In this way, there will be no problems for the output files in terms of path and they will have ended up in the right directory.

```
data_dir <- "data/"
output_dir <- "output/"
```

```r
# Create output directory if it doesn't exist
dir.create(output_dir, showWarnings = FALSE)
```

3 Data Preparation

3.1 Data Cleaning

It was an important step in preparing the Swiss hotel stays dataset for analysis. Character encoding issues correction, handling missed values, and dropping irrelevant columns were all implemented to keep a clean dataset with a well-structured source for merging with other data in further steps.

3.1.1 Character Encoding Issues

Then, prior to importing the dataset, the special character encoding issues that may pop up with some of the canton names: for example, 'Graubünden' to 'Graubuenden', and 'Neuchâtel' to 'Neuchatel' was cleaned. That is because special characters like umlauts for example, 'ü' and accents for example, 'é' can create encoding problems in RStudio, particularly when working across multiple operating systems.

The above preprocessing was necessary to avoid some problems which might have occurred after importing the data into R; for example, inconsistency in the format of data.

3.1.2 Importing and Viewing Data

Finally, the data are imported into R using the read_delim(). The locale parameter is used and set to UTF-8 encoding for correct handling of special characters. Missing values, represented by "...", were automatically treated as NA.

```r
# Read the CSV file with proper encoding
d.hotel_stays <- read_delim(
  file.path(data_dir, "swiss_hotel_stays.csv"),
  delim = ";",
  na = "...",
  locale = locale(encoding = "UTF-8")
)
```

After importing the dataset, glimpse() was used to see the contents of the dataset. The number of rows and columns was given. This dataset contains: 1,872 and 157, respectively. The types of variables were also given: character, double, and logical.

```r
# Preview the data
glimpse(d.hotel_stays)  # More concise than head() and str()
```

3.1.3 Handling Missing Values

The next step was checking the NA values in the dataset by using colSums(is.na()). It was an important quality check to find totally or partially empty columns in the dataset.

```r
# Check for missing values in each column and total NA count
v.na_count <- colSums(is.na(d.hotel_stays))
total_na_count <- sum(v.na_count)

# Output NA summary
print(v.na_count)
print(total_na_count)
```

From these, 10 columns were filtered that were of logical data type, meaning a column with only NA values. Such columns were dropped from the dataset as it contained no useful information.

Dropped columns included:

- Baltic States Arrivals
- Baltic States Overnight stays
- Central America, Caribbean Arrivals
- Central America, Caribbean Overnight stays
- Australia, New Zealand, Oceania Arrivals
- Australia, New Zealand, Oceania Overnight stays
- Gulf States Arrivals
- Gulf States Overnight stays
- Serbia and Montenegro Arrivals
- Serbia and Montenegro Overnight stays

This column removal cleaned up the dataset by making it less bulky but retained only such data which was useful for analysis. There were a few columns in which all values were missing or zero.

```
# Drop logical columns (columns with only NA values)
d.hotel_stays_cleaned <- d.hotel_stays %>%
  select(-where(is.logical))
```

3.1.4 Replacing Remaining Missing Values

For the other columns, which have sporadic missing values, the NA values were replaced with 0. It was done based on the assumption that for column missing values related to arrivals and overnight stays. The count should be zero visitors or overnight stays.

```
# Check for remaining missing values
colSums(is.na(d.hotel_stays_cleaned))

# Replace NA values with 0
d.hotel_stays_cleaned[is.na(d.hotel_stays_cleaned)] <- 0


# Confirm that no NA values remain
print(sum(is.na(d.hotel_stays_cleaned)))
```

By doing this replacement, it ensured that our dataset was free of missing values, which is very critical for correct analysis.

3.1.5 Final Data Preview

After cleaning, glimpse() was used again to review the final dataset. The cleaned dataset consists of 1,872 rows and 147 columns after dropping the irrelevant columns.

```
# Preview the cleaned data
glimpse(d.hotel_stays_cleaned)
```

3.1.6 Saving the Clean Data

The cleaned dataset was then saved into an Excel file, hotel_stays_cleaned.xlsx, for further analysis.

```
# Export the cleaned data frame to an Excel file
write_xlsx(d.hotel_stays_cleaned, file.path(output_dir, "hotel_stays_cleaned.xlsx"))
```

4 Data Transformation

The transformation of the dataset into an appropriate format for analysis was the next stage after cleaning. This involved restructuring data from wide format to long format, changing the data from the multiple column of arrivals and stays per country into a more compact form with fewer columns, and added new columns for date and season. It assists in presenting the data in a form that can support efficient analysis and visualization.

4.1 Extracting and Transforming the Data

To ease the extraction and transformation process of the arrivals and overnight stays data, a reusable function called extract_and_transform() was implemented. It should support automated processing for most of the transforms so that there can be more uniformity and flexibility across the various components of the dataset.

```
# Function to extract relevant columns based on a specific word and transform them to long format
extract_and_transform <- function(d.data, word, pattern_to_remove) {
  d.data %>%
    select(1:3, contains(word)) %>%
    pivot_longer(cols = contains(word),
                 names_to = "CountryOfResidence",
                 values_to = word) %>%
    mutate(CountryOfResidence = str_remove_all(CountryOfResidence, pattern_to_remove))
}
```

4.1.1 Column Selection

The function will select the first three columns, namely, Year, Month, and Canton, and every other column that contains either the string “Arrivals” or “Stays.”. Selection is to be done so that we may focus on the core data we need for the analysis.

4.1.2 Pivoting to Long Format

Then, the selected columns are reshaped from wide format to long format using the function pivot_longer(). In the wide format each country has its own column for arrivals and stays, but in the long format the dataset is restructured so that all countries can be listed in one column called CountryOfResidence, and the values which correspond to them - either arrivals or stays respectively - to be placed in another column.

4.1.3 Cleaning Column Names

After pivoting, the function cleans the column names from extra text like “.Arrivals” or “.Overnight.stays.”. That will keep the column name of CountryOfResidence clean and in good format to interpret meaningfully.

We used this function twice; once to create a long-format dataset for arrivals and another one for stays.

```
# Extract and transform arrivals data
d.arrivals_long_format <- extract_and_transform(d.data = d.hotel_stays_cleaned, word = "Arrivals", patt

# Extract and transform stays data
d.stays_long_format <- extract_and_transform(d.data = d.hotel_stays_cleaned, word = "Stays", pattern_to
```

4.1.4 Preview of Data After Transformation

The preview of the data in both arrivals and stays after the function was applied was done using head(). The long format is much more efficient to work with for additional analysis.

4.2 Rejoin Data

Once in long format, we merged the arrivals and stays data in one set. We used an inner_join() matching on the columns Year, Month, Canton, and CountryOfResidence. Because the merge was performed at this stage, from this point onwards the dataset will include both the arrivals and overnight stays for each country of residence.

The combined data now includes, for each country of residence, the number of arrivals and overnight stays, differentiated by year, month, and by the canton. It is also further allowed for deeper analysis, such as observing the comparisons of tourism flows among different countries and regions and the length of visitor's stay depending on their country of origin.

4.3 Creating Date and Season Columns

Next, after merging, we further value-added the dataset by adding two new columns: one for the Date column and one for the Season column.

```
# Create a new Date column from Year and Month in the correct format, followed by Season calculation
d.hotel_stays_long_format <- d.hotel_stays_long_format %>%
  mutate(
    # Create Date column
    Date = dmy(paste("01", Month, Year)),

    # Create Season column
    Season = case_when(
      Month %in% c("December", "January", "February") ~ "Winter",
      Month %in% c("March", "April", "May") ~ "Spring",
      Month %in% c("June", "July", "August") ~ "Summer",
      Month %in% c("September", "October", "November") ~ "Fall",
      TRUE ~ NA_character_
    )
  )
```

4.3.1 Date Column

This column, Date, is created by concatenating the columns Year and Month and assigning the date to the first of the month, i.e., "01." It is an important column when analysis needs to be done on a time basis. For example, this column can help us identify tourism trends shown by variation over time.

4.3.2 Season Column

The Season column will be created by recoding the Month values into one of four coded seasons of the year, namely,

Winter: December, January, February Spring: March, April, May Summer: June, July, August Fall: September, October, November

This would enable us to conduct a seasonal analysis of tourist flows, which, surely in Switzerland, with the strong seasonal pattern of tourism-winter sports compared with summer activities, is important.

Date and Season columns represent additional information about temporal dynamics of tourism in Switzerland, enabling us to analyze how numbers of tourists flow by time of year.

4.4 Reordering Columns

We rearranged the columns into a more logical order for readability and usability of the dataset. Therefore, we have in order of importance: contextual info (Date, Year, Month, Season, Canton) first and the key

variables of CountryOfResidence, Arrivals, and Stays. This will enable us to understand the data better and enhance further analysis.

```
# Reorder columns by specifying the desired order
d.hotel_stays_long_format <- d.hotel_stays_long_format %>%
  select(
    Date,
    Year,
    Month,
    Season,
    Canton,
    CountryOfResidence,
    Arrivals,
    Stays
  )
```

4.5 Final Preview

After all these transformations, the final pre-viewing was done through head() to check that all columns were correctly formatted and in order. Now this cleaned and transformed data will be ready for further analysis and can be merged with other data-sets, such as weather data, as described in the Data Merging section.

5 Data Merging

After cleaning and preparing the hotel stays dataset, and then preparing the weather dataset, the two dataset were merged to extend the analysis regarding the way weather conditions might affect the patterns of tourism in different Swiss cantons. The section describes how the data was merged and points to important aspects regarding this process.

5.1 Weather Data Collection

There was no single dataset available that combined the weather for all Swiss cantons. For this reason, we collected the weather data for each of the 26 Swiss cantons individually through the https://open-meteo.com/en/docs/historical-weather-api.

For each canton, the latitude and longitude of its capital was defined and the period from 1.1.2018 until 31.12.2023 was set. Besides that, automatic detection of the correct time zone for every location was enabled.

Daily collected weather variables include:

- Maximum Temperature at 2 meters above ground
- Minimum Temperature at 2 meters above ground
- Mean Temperature at 2 meters above ground
- Rain Sum
- Snowfall Sum

Units of measurement:

- Temperature: Degree Celsius (°C)
- Wind Speed: Kilometers per hour (km/h)
- Precipitation: Millimeters (mm)
- Time format: ISO 8601

After downloading the individual weather datasets for each canton, we checked that there were no NA values in the data. This step ensured that the weather data was full and of high quality.

5.2 Consolidate Weather Data

Once the weather data for all cantons was gathered, they were consolidated into one data set. The following steps have been executed:

1. Mapping of Canton Names to File Paths: To do that, each weather file was mapped with its respective path to a canton name via the vector v.file_paths.
2. Read and process Weather Data:

- The function that encapsulates the import and initial processing of each weather file called process_weather_data() was defined. This function does the following:
- It reads in each CSV file and skips rows that are redundant metadata.
- It formats the time column into Date format.
- Creates a column for the appropriate canton for each row.

3. Merging All Cantons: the map2_dfr() function was utilized to apply the process_weather_data() function to all of the files, which merged the individual datasets into a single dataset of the weather data across all cantons: d.combined_weather_data.

```
# Apply the function to all files and combine data frames
d.combined_weather_data <- map2_dfr(names(v.file_paths), v.file_paths, process_weather_data)
```

5.3 Cleaning and Renaming Columns

In consolidating the weather data, special characters for the degree symbol (°) in column names for temperature made later analysis cumbersome. To resolve this by:

- Converting all column names to ASCII format with the iconv() function to remove special characters.
- Renaming columns that had parentheses and spaces to simpler column names using rename(). For example, this:
  - Original names: “temperature_2m_max (°C)”, “rain_sum (mm)”
  - New names: “temperature_2m_max_c”, “rain_sum_mm”.

This allowed for easier manipulation and analyses in subsequent steps.

5.4 Merging Hotel Stays and Weather Data

After cleaning and consolidating the two datasets, an inner join was performed, merging data on hotel stays with that of weather data.

- Joining on Date and Canton: The merge was executed using the inner_join() function, matching the column of Date from the hotel stays dataset to the time column of the weather dataset, then the column of Canton in both datasets.

```
# Perform the merge using inner_join on Date and Canton
d.merged_data <- d.hotel_stays_long_format %>%
  inner_join(d.combined_weather_data, by = c("Date" = "time", "Canton"))
```

- Save the merged Data: The above-prepared merged dataset, which provides the pooled information of the hotel stays with that of the daily weather in every canton, was then saved as an Excel file called ‘merged_data.xlsx’. This data will be analyzed further in order to show dependencies of tourism-through arrival and stay-on weather conditions, portrayed here by temperature, rainfall, and snowfall.

5.5 Summary and final Comments

Setup: By automatic installation of packages and management of directories, the analysis can be reproduced on any machine without changes.

Data Cleaning:

Character Encoding: Clean special characters in Canton names to avoid encoding issues.

Imported Data: The data gets imported; structure checked.

Missing Values: Removing Columns with all NA, rest of NAs changed to 0. **Data Export**: The cleaned dataset devoid of missing and unnecessary data was exported into Excel ready for analysis and merging.

Data Transformation:

Wrangling: Created a custom function that wrangles the Arrivals and Stays data into a long format and cleaned up the column names.

Merging: Merged the two data sets into one, which consisted of the arrival and stays.

New Columns: Add date and season columns and reorganize data into an analysis-friendly format.

Final Transformation: The data is in a long-format and ready for further analysis, including merging with weather data.

Final comments: With this merged dataset, the foundation for subsequent analyses on how weather conditions might affect tourism patterns across Swiss regions was created. The integration allows to analyze the correlation of tourist flows with factors such as temperature, rain, and snowfall, which might provide useful insights into the management and planning of tourism.

By linking the hotel data to weather data, for example it will be possible to assess with this information whether warmer temperatures or increased rainfalls in some cantons reduce tourist arrivals and overnight stays. It will, therefore, help in guiding tourism strategies with respect to weather trends.

6 Exploratory Data Analysis (EDA)

6.1 Summary Statistics

Summary statistics before any visualization is important as this approach helps to gain most meaningful insights about how key variables are distributed, including overnight stays, weather and visitors' nationalities.

```
# Preview hotel stays data
head(d.hotel_stays_long_format)
```

```
## # A tibble: 6 x 8
##   Date        Year Month   Season Canton CountryOfResidence Arrivals  Stays
##   <date>     <dbl> <chr>   <chr>  <chr>  <chr>                 <dbl>  <dbl>
## 1 2018-01-01  2018 January Winter Zurich Switzerland           67556 110508
## 2 2018-01-01  2018 January Winter Zurich Germany               26087  46072
## 3 2018-01-01  2018 January Winter Zurich France                 5973   9592
## 4 2018-01-01  2018 January Winter Zurich Italy                  6179  11188
## 5 2018-01-01  2018 January Winter Zurich Austria                3783   6370
## 6 2018-01-01  2018 January Winter Zurich United Kingdom        11434  19710
```

```
# Summarize the data: Calculate the total overnight stays for each canton
d.hotel_stays_summary <- d.hotel_stays_long_format %>%
  group_by(Canton) %>%
  summarise(Total_Stays = sum(Stays, na.rm = TRUE)) %>%
  arrange(desc(Total_Stays))

# View the summarized data
head(d.hotel_stays_summary)
```

```
## # A tibble: 6 x 2
##   Canton      Total_Stays
##   <chr>             <dbl>
## 1 Graubuenden    31303975
## 2 Bern           30374981
## 3 Zurich         28171879
## 4 Valais         23788630
## 5 Geneva         15521209
## 6 Vaud           15079794
```

```
# Sum of total overnight stays per canton and year
df.total_stays_by_canton <- d.hotel_stays_long_format %>%
  group_by(Canton, Year) %>%
  summarize(total_stays = sum(Stays, na.rm = TRUE))
```

Key Summary Metrics:

Overnight stays:

- The dataset covers the period from 2018 to 2023, covering all the 26 cantons in Switzerland
- Depending on the cantons, the average frequency of overnight accommodations highly varies from month to month, some of the most popular cantons are Graubuenden, Bern, Zurich, Valais and Geneva.
- Swiss guests account for a very large portion, although this share of foreign tourist is highly variable depending on the month and the canton.
- It also shows some valuable information regarding nationality and thus place Germany, the United Kingdom and the United Sates as the leading contributors to the tourism sector in Switzerland.

Weather Conditions:

- Temperature varies across Switzerland, with winter and summer season, on average the temperature stays between -12°C ****and 30°C throughout the year.
- Snowfall is one of the major variables of winter tourism in mountainous region such as Graubuenden and Valais. This attract the higher percentage of tourist.
- Rainfall is more evenly distributed throughout the year, though some regions experience wetter summers than others.

Seasonality:

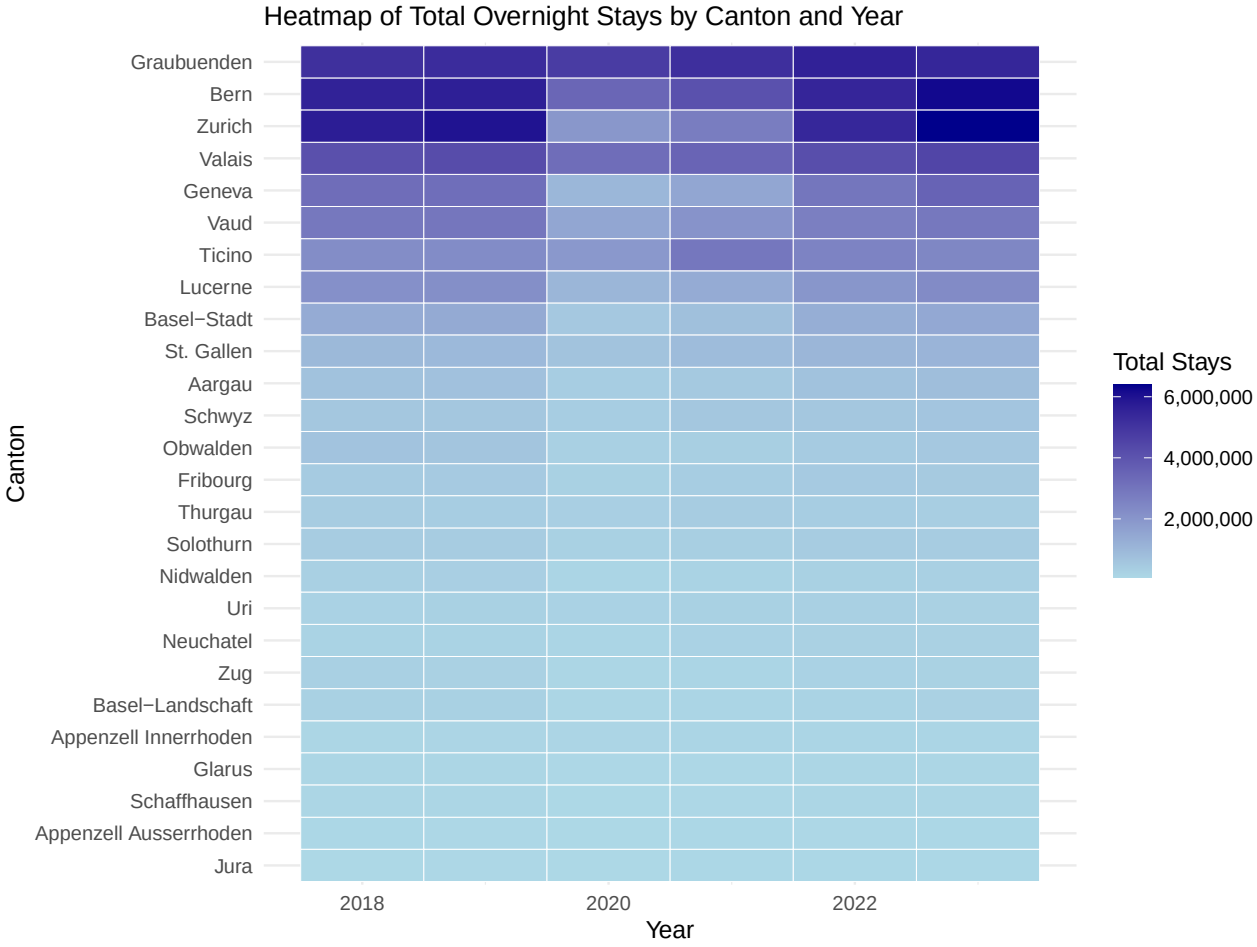
The data reveals clear seasonal trends, with two primary peaks in tourism, winter for skiing and snow activities and summer for hiking and outdoor fun. In between, there are transitional seasons spring and autumn, where the volume of overnight stays decrease significantly

6.2 Data Visualization from the data Overnight stays.

To better understand the trends in the data, several visualizations were created to uncover patterns in tourist behavior, seasonality, and the effects of weather on tourism across different cantons.

6.2.1 Total Overnight Stays by Canton (2018–2023)

- **Plot Summary**: The heat map shows the **total overnight stays** across different Swiss cantons from 2018 to 2023.

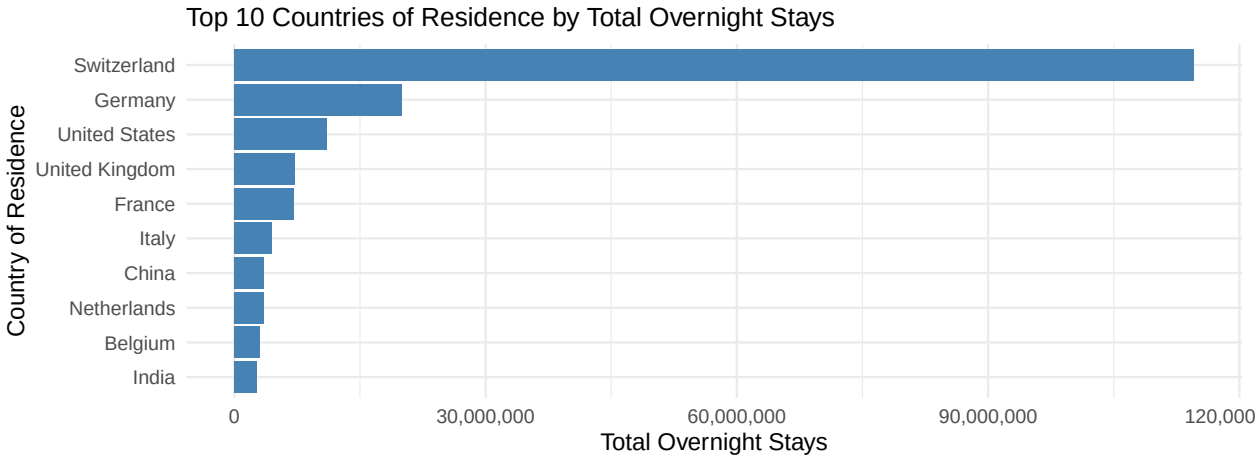


- **Insights**:
  - **Graubünden**, **Bern**, and **Zurich** stand out as the top destinations for overnight stays, likely due to a combination of winter sports tourism (Graubünden), urban tourism (Zurich), and outdoor

recreation in both summer and winter (Bern).
  - **Valais** and **Geneva** also rank highly, with Valais' popularity driven by outdoor tourism and Geneva's by its status as an international hub.
  - This plot helps identify which cantons are the most important tourism markets, giving us a sense of where tourists are concentrated.

6.2.2 Top 10 Countries of Residence by Total Overnight Stays

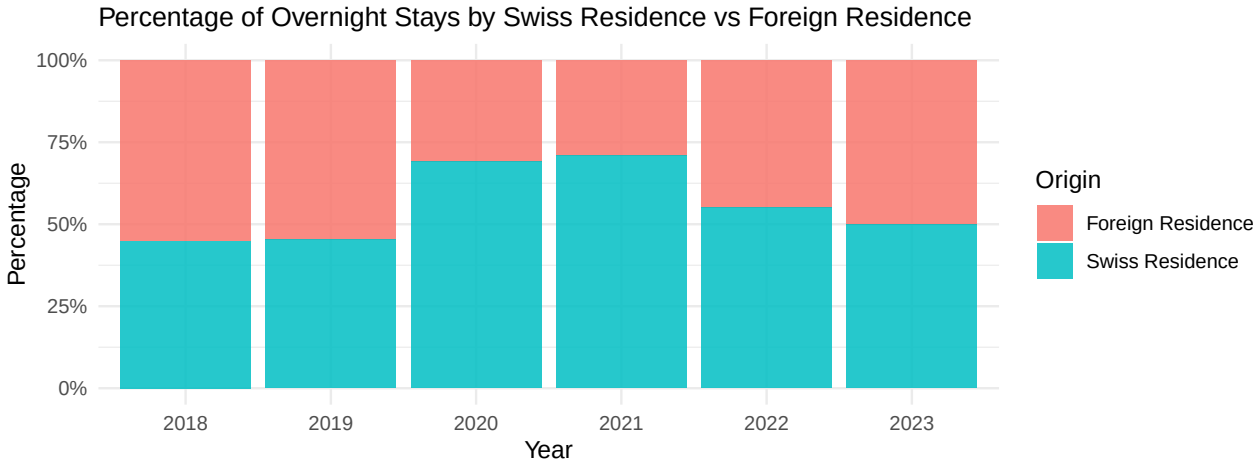
- **Plot Summary**: A bar chart showing the top 10 countries of residence of tourists based on total overnight stays in Switzerland.



- **Insights**:
  - The majority of overnight stays are done by Swiss residents which indicates a strong presence of domestic tourism. And the most important foreign markets are Germany, United States and United Kingdom.
  - The inclusion of other European countries like **France** and **Italy**, as well as non-European countries like **China** and **India**, shows Switzerland's attractiveness on a global scale.
  - Understanding which nationalities make up the majority of overnight stays helps in **targeted marketing** and **seasonal promotions** aimed at attracting foreign visitors.

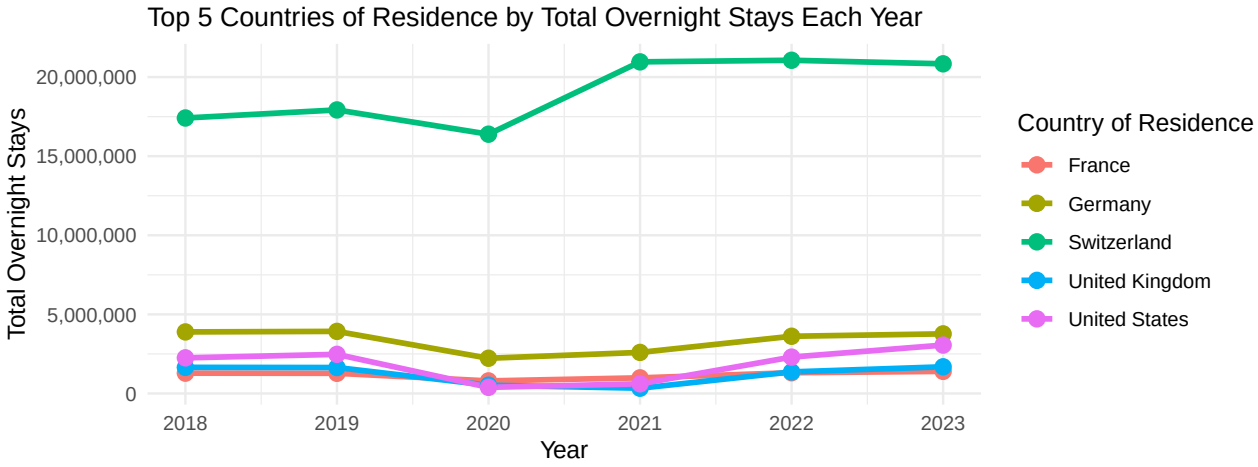
6.2.3 Overnight Stays by Swiss Residents vs. Foreign Residents (2018–2023)

1. **Plot Summary**: This stacked bar chart shows the proportion of overnight stays by **Swiss residents** versus **foreign residents** from 2018 to 2023.



- **Insights**: **Domestic Tourism Dominates**: Swiss residents account for the highest number of overnight stays, with a significant surge during the pandemic in 2020 and 2021. Even post-pandemic, domestic travel remains robust, highlighting the ongoing importance of local tourism.

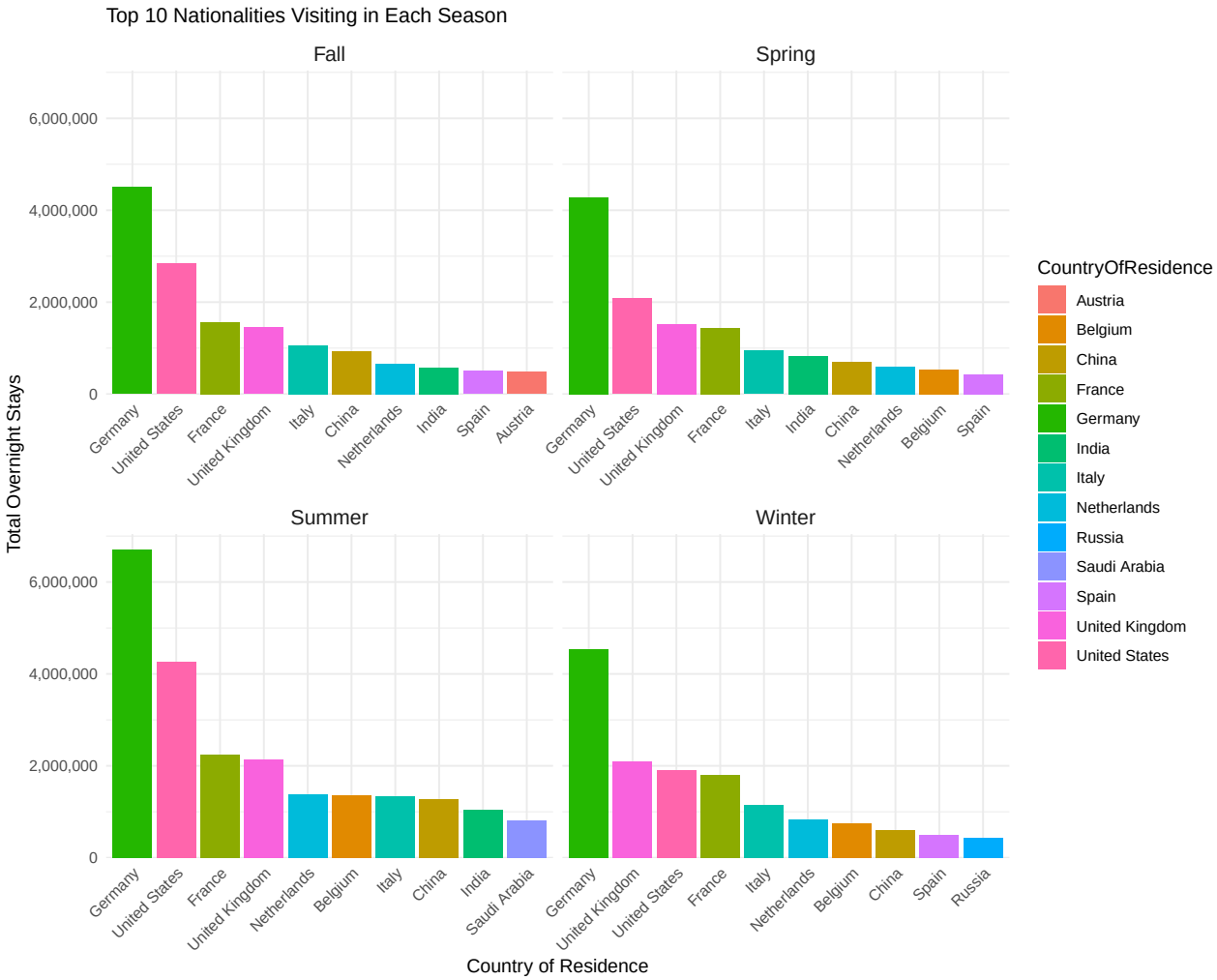
2. **Plot Summary**: This line chart shows the total number of overnight stays from the **top 5 countries of residence** (Switzerland, Germany, the United Kingdom, the United States, and France) each year between **2018 and 2023**.



- **Insights**:
  - Across all nationalities, **2020** stands out as the year with the most significant decline in overnight stays due to the **COVID-19 pandemic**.
  - **Germany** is the largest source of international tourists to Switzerland, though the total number of stays for German visitors dropped significantly in **2020** due to the pandemic.
  - This visualization gives a good picture of how **global events,** such as the pandemic, can alter the balance between domestic and international tourism.

6.2.4 Different Nationalities Visiting in Each Season and their favorite Cantons**

- **Plot Summary**: This set of bar charts shows the total number of overnight stays by the top 10 nationalities across **four seasons**: **Fall**, **Spring**, **Summer**, and **Winter**. Each chart presents the most significant countries contributing to tourism in Switzerland during the respective season.



Key Insights:

1. **Germany Dominates Across All Seasons**:

- **German tourists** clearly dominate tourism in Switzerland, regardless of the season, with significantly higher overnight stays than any other nationality. This trend may be attributed to Switzerland's proximity to Germany and the similar cultural and linguistic ties between the two nations.
- Germany's strong showing in **Winter** and **Summer** indicates that tourists from this country are attracted to both **winter sports** and **summer activities** such as hiking and city tourism.

2. **Seasonal Preferences by Region**:

- The United States, United Kingdom ****and France maintains a steady presence in all seasons with more visits in **Summer** than in **Winter**. The UK shows relatively lower numbers in **Fall** and **Spring**, suggesting that peak travel happens during major holiday seasons.
- **Italy**, **Belgium**, and **Netherlands** contribute modestly across all seasons, but no significant peaks are observed, which may suggest that tourism from these countries is more **consistent** throughout the year.

3. **Winter Focus**:

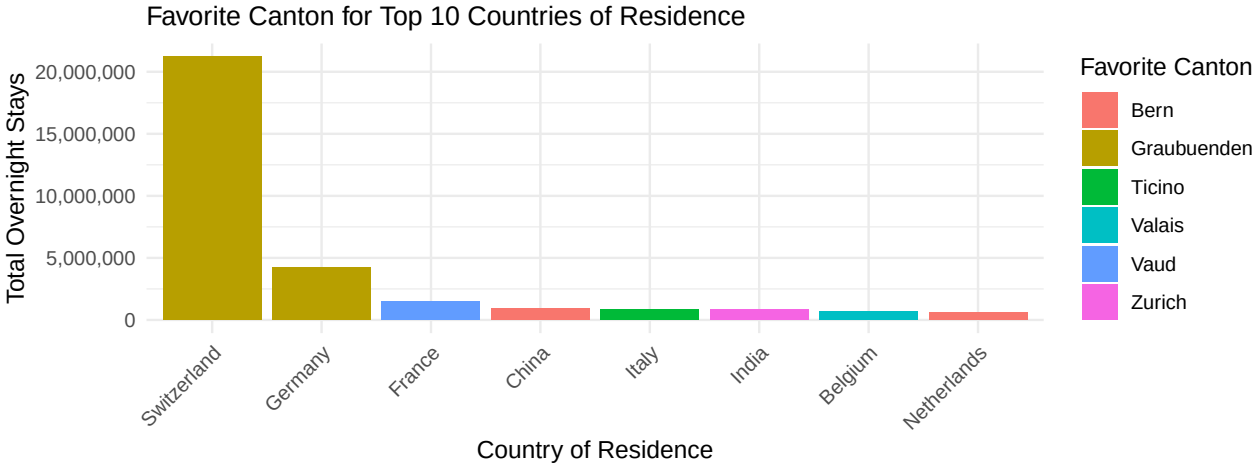
- For countries like **Russia**, **Austria**, and **Spain**, **Winter tourism** shows notable participation. Russia, in particular, shows higher numbers during Winter compared to other seasons, likely driven by Switzerland's reputation for **ski resorts** and **snow-related activities**.

4. **Diverse Nationality Preferences**:

- **China** and **India** appear in both **Spring** and **Summer**, indicating that tourists from these countries may prefer **warmer weather** and are drawn to Switzerland's lakes, mountains, and natural scenery during these times. This contrasts with European visitors, who show a stronger presence in the **Winter** months for **skiing** and **outdoor winter activities**.

6.2.5 Favorite Canton for Top 10 Countries of Residence by Total Overnight Stays

- **Plot Summary**: This set of bar charts shows the total number of overnight stays by the top 10 nationalities and their favorite Canton where they stayed.

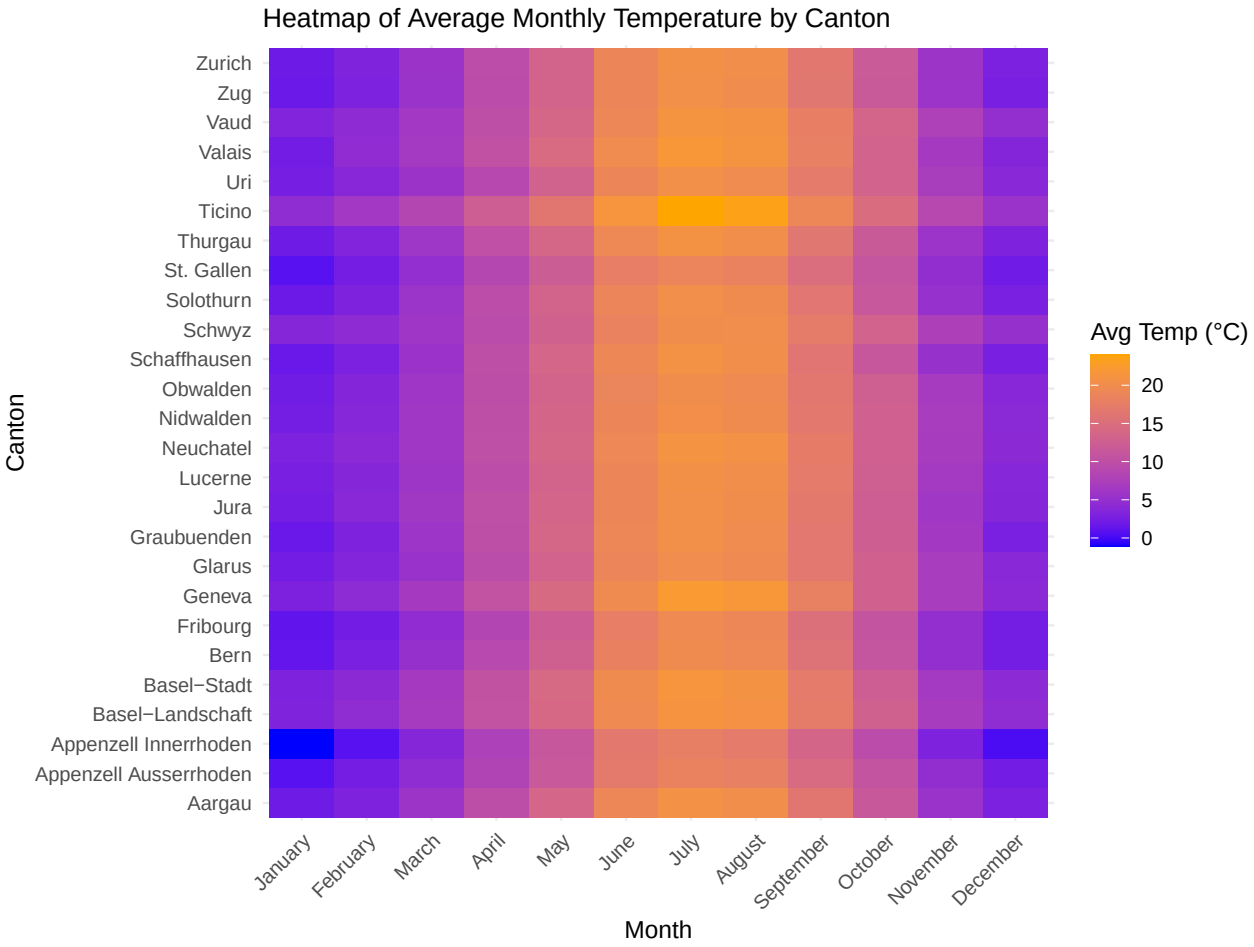


Key Insights: - **Switzerland** leads by a large margin with more than **20 million overnight stays**, reflecting the high internal travel by Swiss residents within their own country. The most visited canton for Swiss residents is **Graubünden**, known for its winter and nature tourism. - **Germany**, **United States**, and **France** follow as the top foreign visitors, with **Zurich** and **Graubünden** being their most popular destinations. This reflects Zurich's appeal for business and city tourism, and Graubünden's draw for outdoor activities. - **Other countries** like **China**, **Italy**, and **India** contribute smaller portions, preferring Zurich and some other regions like **Ticino** and **Valais**.

6.3 Data Visualization from the weather data

6.3.1 Heatmap of Average Monthly Temperature by Canton:

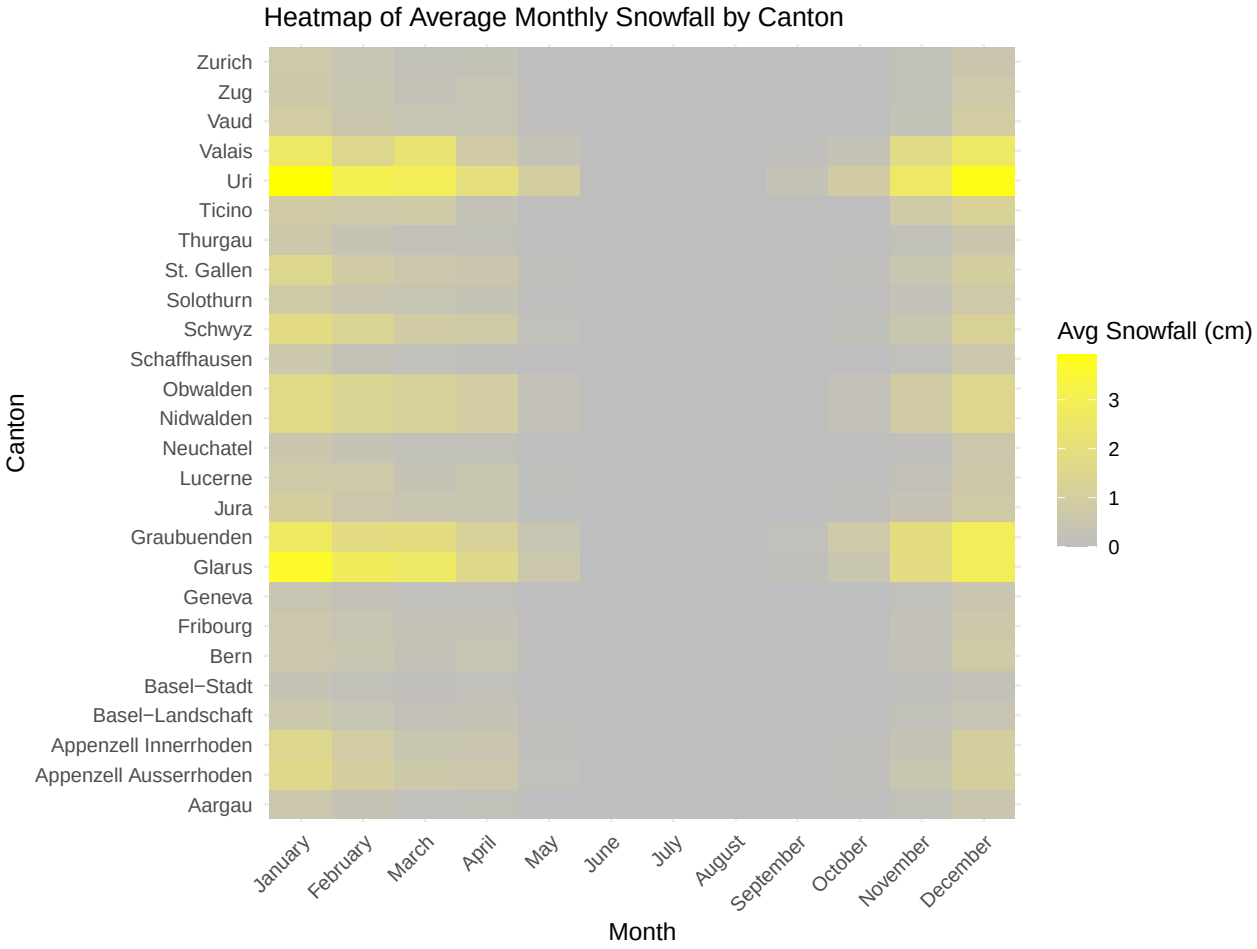
- **Plot Summary**: The heat map shows the average temperature across different months in Swiss Cantons



Key Insights:

- **Seasonal Temperature Variation**: The heatmap shows a clear temperature variation with **cooler temperatures** (shades of blue) during the **winter months** (November to March) and **warmer temperatures** (shades of orange) during the **summer months** (June to August).
- **July and August**: These months exhibit the highest average temperatures across all cantons, with **Ticino** and **Geneva** showing particularly warmer conditions.
- **Winter Months**: The coldest months are **January** and **February**, with **Appenzell Innerrhoden** standing out as one of the coldest cantons.

6.3.2 Heatmap of Average Monthly Snowfall by Canton



Key Insights:

- **High Snowfall in Winter Months**: The heatmap shows that **snowfall** is concentrated in the **winter months**, particularly from **November to March** across most cantons.
- **Graubünden, Valais, Uri, and Glarus**: These cantons experience higher snowfall compared to others, especially in the **peak winter months** (January, February, December).
- **Summer Months**: There is almost **no snowfall** during the summer months (May through August) across all cantons, which aligns with seasonal weather patterns in Switzerland.

7 Model Fitting

7.1 Linear Regression

Linear regression is a fundamental statistical technique that helps to understand the relationship between one or more independent variables (here, weather factors like temperature, snowfall, and rainfall) and a dependent variable (overnight stays)

7.1.1 Temperature: Linear Regression for temperature effect:

```
# Linear regression for temperature effect
lm.model_temp <- lm(Stays ~ temperature_2m_mean_c, data = d.merged_data)
summary(lm.model_temp)
```

```
##
## Call:
## lm(formula = Stays ~ temperature_2m_mean_c, data = d.merged_data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -2449  -1675  -1370  -1091 602439
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)            1087.79      64.53  16.858   <2e-16 ***
## temperature_2m_mean_c    44.49       4.98   8.934   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12960 on 134782 degrees of freedom
## Multiple R-squared:  0.0005919,  Adjusted R-squared:  0.0005844
## F-statistic: 79.82 on 1 and 134782 DF,  p-value: < 2.2e-16
```

- **Positive Correlation**: There's a significant positive relationship between **temperature** and **overnight stays** (coefficient = 44.49, p-value < 0.001). This suggests that for every degree Celsius increase in temperature, overnight stays increase slightly.
- **R-squared**: However, the **R-squared value** is extremely low (0.00059), meaning **temperature explains very little** of the variation in overnight stays.
- **Interpretation**: While temperature has some effect, it's not a major factor driving the number of stays.

7.1.2 Snowfall: Linear Regression for snowfall effect:

```
# Linear regression for snowfall effect
lm.model_snow <- lm(Stays ~ snowfall_sum_cm, data = d.merged_data)
summary(lm.model_snow)
```

```
##
## Call:
## lm(formula = Stays ~ snowfall_sum_cm, data = d.merged_data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -1976  -1555  -1519  -1279 602924
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)      1555.31      37.05  41.980   <2e-16 ***
## snowfall_sum_cm    18.99      14.16   1.341     0.18
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12960 on 134782 degrees of freedom
## Multiple R-squared:  1.334e-05,  Adjusted R-squared:  5.92e-06
## F-statistic: 1.798 on 1 and 134782 DF,  p-value: 0.18
```

- **Weak Positive Correlation**: The regression suggests a weak positive relationship between snowfall and stays, but it is **not statistically significant** (coefficient = 18.99, p-value = 0.18). The **R-squared** is also very low.
- **Interpretation**: Snowfall also doesn't seem to drive the majority of stays, even in winter-heavy regions.

7.1.3 Rainfall: Linear Regression for rainfall effect:**

```
# Linear regression for temperature effect
lm.model_temp <- lm(Stays ~ rain_sum_mm, data = d.merged_data)
summary(lm.model_temp)
```

```
##
## Call:
## lm(formula = Stays ~ rain_sum_mm, data = d.merged_data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -1597  -1566  -1527  -1283 602906
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1567.4556    42.0381  37.287   <2e-16 ***
## rain_sum_mm    0.6562     5.1644   0.127    0.899
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12960 on 134782 degrees of freedom
## Multiple R-squared:  1.198e-07,  Adjusted R-squared:  -7.3e-06
## F-statistic: 0.01615 on 1 and 134782 DF,  p-value: 0.8989
```

- **No Significant Correlation**: Rainfall has no significant impact on overnight stays (coefficient = 0.6562, p-value = 0.899). The R-squared value is negative, which suggests no meaningful relationship between rainfall and stays.
- **Interpretation**: Rainfall doesn't seem to influence tourist activity, which makes sense since tourism is often driven by factors unrelated to small changes in rainfall.

8 Chapter of choice

8.1 Date Transformation (Using lubridate)

The Swiss hotel stays dataset initially had the Year and Month in separate columns. These needed to be combined into one column called Date, for easier compatibility with the weather data. This was facilitated through the use of the lubridate package which makes date operations a bit easier to conduct on R.

```
d.hotel_stays_long_format <- d.hotel_stays_long_format %>%
  mutate(
    # Create Date column
    Date = dmy(paste("01", Month, Year))
  )
```

8.2 Axis Formatting (Using scales)

It was relevant, during the visualization process, to standardize the presentation of numeric values on the axes considering that big numbers would be used as in the case of tourist arrivals or stays. The scales package was relied upon for formatting of y-axis numeric values for better readability.

Here the scales::comma() function formate large number with comma, for example 10000 to 10'000 and large exponetial number for example .1.5e+07 into readable numeric value that provide easier interpretation of plots.

8.3 Using tidytext for Data Reordering

The scale_x_reordered() function was used for the plots which is part of tidytext package. This function allows for a reordered categorical data across facets. That is very useful for visualizing factors that change across different groups.

Visualization: Top 10 Nationalities by Season One of the key insights to be developed was to identify the top 10 nationalities visiting Swiss hotels across various seasons. The below plot shows, for each season, a bar chart of the nationalities, with some internal rearrangement to bring out the ones that had the highest total overnight stays.

```
# Plot for all seasons with top 10 nationalities per season
ggplot(d.top_10_nationalities_per_season, aes(x = reorder_within(CountryOfResidence, -total_stays, Seas
  geom_bar(stat = "identity") +
  scale_y_continuous(labels = scales::comma, expand = expansion(mult = c(0, 0.05))) +  # Ensure bars ap
  scale_x_reordered() +  # Reorders within each facet
  facet_wrap(~ Season, scales = "free_x") +  # Facet by season with free x-axis for each season
  labs(
    title = "Top 10 Nationalities Visiting in Each Season",
    x = "Country of Residence",
    y = "Total Overnight Stays"
  ) +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),  # Align country names underneath the bars
    strip.text = element_text(size = 12)  # Adjust size of facet titles (seasons)
  )
```

Explanation:

- scale_x_reordered(): This additional function makes sure the reordered categories (nationalities) are correctly displayed within each facet (season) on the x-axis.

The scales::comma function is used in the y axis to improve readability for large numbers.

8.4 Creating Reports with xfun

The xfun package was utilized to ensure that the report would be exportable and could be presented in professional form. The package makes it quite easy to render an RMarkdown file into formats like HTML, PDF, and Word.

xfun provides the possibility to automate the whole compilation and save the report in the desired format; this way, the final document can be given to clients or other stakeholders without any intervention.

9 Findings and Analysis

9.1 Switzerland Overnight Stays Pattern

- **Top Tourism Cantons**: Graubünden, Bern, and Zurich are the leading destinations for overnight stays, driven by a mix of winter sports, urban tourism, and year-round outdoor activities.
- **Dominance of Domestic Tourism**: Swiss residents make up the majority of overnight stays, with a noticeable increase during the pandemic (2020-2021). Domestic travel remains strong, emphasizing the significance of local tourism even post-pandemic.

9.2 Foreign Tourist Preferences

- German tourists are the largest group visiting Switzerland throughout all seasons, likely due to geographical proximity and cultural ties.
- Tourists from the United States, the United Kingdom, and France show consistent visits across all seasons, with a noticeable increase in summer, suggesting that these visitors are drawn more to warm-weather activities.
- Winter tourism is especially important for visitors from Russia, Austria, and Spain, with more tourists from these countries coming during the colder months for winter sports like skiing.
- Tourists from China and India tend to visit Switzerland during spring and summer, likely attracted to Switzerland's natural beauty, lakes, and mountain landscapes

9.3 Impact of Weather on Overnight Stays

1. **No Strong Linear Relationship Between Temperature and Stays**:

- Across all cantons, there is no significant linear correlation between **mean temperature** and **overnight stays**. Most cantons, including **Zurich**, **Bern**, and **Valais**, show wide variability in stays regardless of temperature, indicating that **temperature alone is not a primary driver** for tourism in these regions.
- **Mountainous cantons** like **Graubünden** exhibit steady tourism throughout various temperature ranges, suggesting a strong year-round appeal for activities beyond just seasonal weather conditions.

2. **Weak Influence of Snowfall on Tourism**:

- Snowfall shows a slight positive correlation with overnight stays in **Graubünden** and **Valais**, both of which are known for winter sports tourism. However, the majority of stays occur even when snowfall is low, indicating that **tourism in these regions is not entirely snow-dependent**.
- **Urban cantons** like **Zurich** and **Bern** show little to no correlation between snowfall and stays, reinforcing that **urban tourism** is largely unaffected by snow.

3. **Rainfall Does Not Significantly Affect Overnight Stays**:

- In most cantons, including **Zurich**, **Bern**, and **Graubünden**, there is no strong correlation between **rainfall** and overnight stays. In some regions, like **Valais** and **Lucerne**, there is a slight negative correlation, as rain may deter outdoor activities.
- However, **urban tourism** and **business travel** in cities like **Zurich** and **Bern** are resilient to changes in rainfall, with consistent stays regardless of weather conditions.

The findings shows that **snowfall and temperature** have some influence on overnight stays, especially in regions where winter tourism is important. However, the impact is relatively small, and other factors, such as events, business travel, and other attractions is likely to play significant role in determining tourism patterns.

10 Conclusion

In conclusion, this analysis helps to find **foreign tourist preferences**, **seasonal patterns and popular Swiss cantons**,. With the second dataset **weather conditions** it shows a weak relation with number of overnight stays, so **weather** itself is not the sole driving factor . **Switzerland's tourism** is multi-faceted, with both **urban** and **rural attractions** maintaining strong year-round appeal.

The insights gained during this research can be helpful to Swiss tourism authorities on making marketing strategies, focusing on the key drivers of tourism for both international and domestic visitors across all seasons.

11 Use of Generative AI and Github

For this project, generative AI (like ChatGPT) has been an invaluable tool in many ways. It helped refine the project goals after initial brainstorming sessions, providing clarity and direction as the project evolved.

It has been particularly useful when it came to finding specific codes for creating visualizations or troubleshooting coding errors. Whether it was assisting in designing plots or navigating through tricky programming challenges, ChatGPT has made the entire process smoother and more efficient. Additionally, it consistently offered helpful suggestions and quick problem-solving techniques, saving both time and effort. This support allowed to focus more on understanding the content while handling technical obstacles with ease.

The other tool which was very helpful during this project work was GitHub. It allowed team members to work on different parts of the project without overwriting each other's code. It helped track changes, review the history of the code, and helped get back the older versions if needed. Overall, GitHub significantly improves workflow, making collaboration more seamless, structured, and efficient.

12 References

1. opendata.swiss
2. Open-Meteo Historical Weather API
3. holidaystoswitzerland.com

13 Appendix

13.1 Session Info

```
sessionInfo()
```

```
## R version 4.3.2 (2023-10-31 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 11 x64 (build 22631)
##
## Matrix products: default
##
##
## locale:
## [1] LC_COLLATE=English_United States.utf8
## [2] LC_CTYPE=English_United States.utf8
## [3] LC_MONETARY=English_United States.utf8
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.utf8
##
## time zone: Europe/Zurich
## tzcode source: internal
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] xfun_0.47       tinytex_0.49    tidytext_0.4.2  plotly_4.10.4
##  [5] writexl_1.5.0   scales_1.3.0    lubridate_1.9.3 forcats_1.0.0
##  [9] stringr_1.5.1   dplyr_1.1.4     purrr_1.0.2     readr_2.1.5
## [13] tidyr_1.3.1     tibble_3.2.1    ggplot2_3.5.0   tidyverse_2.0.0
##
## loaded via a namespace (and not attached):
##  [1] janeaustenr_1.0.0 utf8_1.2.4        generics_0.1.3    lattice_0.22-6
##  [5] stringi_1.8.3     hms_1.1.3         digest_0.6.34     magrittr_2.0.3
##  [9] evaluate_1.0.0    grid_4.3.2        timechange_0.3.0  fastmap_1.1.1
## [13] jsonlite_1.8.8    Matrix_1.6-1.1    httr_1.4.7        fansi_1.0.6
## [17] viridisLite_0.4.2 lazyeval_0.2.2    cli_3.6.2         crayon_1.5.3
## [21] rlang_1.1.3       tokenizers_0.3.0  bit64_4.0.5       munsell_0.5.1
## [25] withr_3.0.1       yaml_2.3.8        parallel_4.3.2    tools_4.3.2
## [29] tzdb_0.4.0        colorspace_2.1-0  vctrs_0.6.5       R6_2.5.1
## [33] lifecycle_1.0.4   bit_4.0.5         htmlwidgets_1.6.4 vroom_1.6.5
## [37] pkgconfig_2.0.3   pillar_1.9.0      gtable_0.3.5      Rcpp_1.0.13
## [41] glue_1.7.0        data.table_1.16.0 tidyselect_1.2.1  rstudioapi_0.16.0
## [45] knitr_1.48        farver_2.1.2      SnowballC_0.7.1   htmltools_0.5.7
## [49] labeling_0.4.3    rmarkdown_2.28    compiler_4.3.2
```

13.2 GitHub Repository

https://github.com/kenny-trinh/swiss_hotel_trends