Customer Churn Insights

Prepared By Lisa Totton

March 2025

MAVEN MUSIC

SINCE 1996

Overview

MAVEN MUSIC
SINCE 1996

THE CLIENT: MAVEN MUSIC

- Music streaming service
- Increased customer churn over the past three months

THE ASK:

- Scope the data science project
- Gather the data in Python
- Clean the data
- Explore and visualize the data
- Prepare the data for modeling

MAVEN MUSIC
SINCE 1996

Scoping The Data Science Project

We plan to use **supervised learning** to predict which customers are likely to cancel their subscription, using three months of historical subscription and listening history. This will allow us to:

- Identify the **top predictors for cancellation** and figure out how to address them.
- Use the model to **flag customers who are likely to cancel** and take proactive steps to keep them subscribed.
- Our goal is to **reduce cancellations by 2%** over the next year.

Note: This is the end goal for the project. This presentation covers the data preparation and exploratory data analysis.

Customer Table

	Customer ID	Customer Name	Email	Member Since	Subscription Plan	Subscription Rate	Discount?	Cancellation Date
0	5001	Harmony Greene	Email: harmonious.vibes@email.com	3/13/23	Basic (Ads)	$2.99	NaN	NaN
1	5002	Aria Keys	Email: melodious.aria@email.edu	3/13/23	NaN	$2.99	NaN	NaN
2	5004	Lyric Bell	Email: rhythmical.lyric@email.com	3/13/23	NaN	$2.99	NaN	6/1/23
3	5267	Rock Bassett	Email: groovy.rock@email.com	3/20/23	Basic (Ads)	$2.99	NaN	NaN
4	5338	Rhythm Dixon	Email: beats.by.rhythm@email.edu	3/20/23	NaN	$2.99	NaN	NaN
5	5404	Jazz Saxton	Email: jazzy.sax@email.com	3/20/23	NaN	$2.99	NaN	6/3/23
6	5581	Reed Sharp	Email: sharp.tunes@email.com	3/21/23	Premium (No Ads)	$9.99	NaN	NaN
7	5759	Carol Kingbird	Email: songbird.carol@email.com	3/22/23	Premium (No Ads)	$9.99	NaN	6/2/23
8	5761	Sonata Nash	Email: musical.sonata@email.com	3/28/23	Premium (No Ads)	$9.99	NaN	NaN
9	5763	Jazz Coleman	Email: coleman.jazzmaster@email.com	3/28/23	Basic (Ads)	$2.99	NaN	NaN
10	5826	Chord Hayes	Email: harmonic.chord@email.com	3/28/23	Basic (Ads)	$2.99	NaN	NaN
11	5827	Rhythm Franklin	Email: rhythmic.franklin@email.edu	3/28/23	NaN	$2.99	NaN	NaN

MAVEN MUSIC
SINCE 1996

Listening History & Session Tables

	Customer ID	Session ID	Audio Order	Audio ID	Audio Type
0	5001	100520	1	101	Song
1	5001	100520	2	102	Song
2	5001	100520	3	103	Song
3	5001	100520	4	104	Song
4	5001	100520	5	105	Song
5	5001	100520	6	108	Song
6	5001	100520	7	109	Song
7	5001	100520	8	110	Song
8	5001	100520	9	101	Song
9	5001	100522	1	105	Song
10	5001	100522	2	102	Song
11	5001	100522	3	103	Song

	Session ID	Session Log In Time
0	100520	2023-03-13 18:29:00
1	100522	2023-03-13 22:15:00
2	100525	2023-03-14 10:01:00
3	100527	2023-03-13 14:14:00
4	100538	2023-03-21 12:23:00
5	100542	2023-03-21 19:29:00
6	100549	2023-03-22 00:30:00
7	100556	2023-03-22 07:02:00
8	100579	2023-04-01 22:30:00
9	100589	2023-04-02 17:00:00
10	100789	2023-04-09 09:23:00
11	100823	2023-04-12 15:32:00

MAVEN MUSIC
SINCE 1996

Audio Table

	ID	Name	Genre	Popularity
0	Song-101	Dance All Night	Pop	1
1	Song-102	Unbreakable Beat	Pop	2
2	Song-103	Sunset Boulevard	Pop Music	5
3	Song-104	Glowing Hearts	Pop Music	10
4	Song-105	Pop Rocks	Pop Music	52
5	Song-106	My Old Dog and My True Love	Country	23
6	Song-107	Dirt Road Romance	Country	30
7	Song-108	Chase the Dream	Hip Hop	4
8	Song-109	Rise Above	Hip Hop	9
9	Song-110	Boss Moves	Hip Hop	28
10	Song-111	Moonlit Serenade	Jazz	63
11	Song-112	Midnight Blues	Jazz	80

MAVEN MUSIC
SINCE 1996

Data Cleansing

Checking for:

Missing (NaN) Values

Example:

	Column Name	NaN Count
0	Customer ID	0
1	Customer Name	0
2	Email	0
3	Member Since	0
4	Subscription Plan	5
5	Subscription Rate	0
6	Discount?	0
7	Cancellation Date	17

Inconsistent Text & Typos

Example:

	Genre	count
2	Comedy	3
4	Country	2
1	Hip Hop	3
5	Jazz	2
3	Pop	2
0	Pop Music	3
6	True Crime	2

Outliers

Example:

	Subscription Rate	count
0	2.990000	17
1	7.990000	7
2	9.990000	5
3	99.989998	1

Duplicate Rows

No duplicate rows found in any tables

Data Cleansing – Customer Table

MAVEN MUSIC
SINCE 1996

Email Column

- Removed “Email:” text prefix

Email
Email: harmonious.vibes@email.com
Email: melodious.aria@email.edu
Email: rhythmical.lyric@email.com

Subscription Plan and Subscription Rate Columns

1. Filled NaN values in **Subscription Plan** column with ‘Basic (Ads)’ to align with Subscription Rate column.
2. Removed dollar sign ($) prefix from **Subscription Rate**
3. Replaced **Subscription Rate** Outlier value with ‘$9.99’ to align with Subscription Plan and Discount columns.

	Subscription Plan	Subscription Rate	Discount?
0	Basic (Ads)	$2.99	No
1	NaN	$2.99	No
6	Premium (No Ads)	$9.99	No
15	Premium (No Ads)	$99.99	No
21	Premium (No Ads)	$7.99	Yes

Data Cleansing – Listening History Table

Unique Identifier Column

- Analyzed **Audio ID** and **Audio Type** columns and determine they would need to be concatenated to create a unique identifier for each audio element.

	Audio ID	Audio Type	Audio ID Type
0	101	Song	Song-101
1	101	Podcast	Podcast-101
2	102	Podcast	Podcast-102
3	102	Song	Song-102
4	103	Song	Song-103
5	103	Podcast	Podcast-103
6	104	Song	Song-104
7	105	Song	Song-105
8	106	Song	Song-106
9	107	Song	Song-107
10	108	Song	Song-108
11	109	Song	Song-109
12	110	Song	Song-110

Data Cleansing – Audio Table

Genre Column

- Replaced inconsistent “Pop Music” text with “Pop” to be consistent with other Genres.

	Genre	count
2	Comedy	3
4	Country	2
1	Hip Hop	3
5	Jazz	2
3	Pop	2
0	Pop Music	3
6	True Crime	2

Exploratory Data Analysis Prep

MAVEN MUSIC
SINCE 1996

Add **Cancelled** flag column

```
#Create Cancelled Flag Column
customer_model["Cancelled?"] = np.where(customer_model["Cancellation Date"].isna(), False, True)
customer_model[["Customer ID", "Cancellation Date", "Cancelled?"]].head()
```

	Customer ID	Cancellation Date	Cancelled?
0	5001	NaT	False
1	5002	NaT	False
2	5004	2023-06-01	True
3	5267	NaT	False
4	5338	NaT	False

Exploratory Data Analysis Prep

Add **Membership Duration** column

```
cutoff_date = pd.to_datetime("2023-05-31")

customer_model["Membership Duration (Days)"] = np.where(
    customer_model["Cancelled?"] == False,
    (cutoff_date - customer_model["Member Since"]).dt.days,
    (customer_model["Cancellation Date"] - customer_model["Member Since"]).dt.days
)
customer_model[["Customer ID", "Member Since", "Membership Duration (Days)"]].head()
```

	Customer ID	Member Since	Membership Duration (Days)
0	5001	2023-03-13	79.0
1	5002	2023-03-13	79.0
2	5004	2023-03-13	80.0
3	5267	2023-03-20	72.0
4	5338	2023-03-20	72.0

Exploratory Data Analysis Prep

Add **Session Count** column

```
#Create Session Count Column
customer_sessions = (history
                     .groupby("Customer ID")["Session ID"]
                     .nunique()
                     .reset_index(name="Session Count"))

customer_model = customer_model.merge(customer_sessions, on="Customer ID", how="left")
customer_model[["Customer ID", "Session Count"]].head()
```

	Customer ID	Session Count
0	5001	8
1	5002	4
2	5004	1
3	5267	7
4	5338	4

MAVEN MUSIC
SINCE 1996

Exploratory Data Analysis Prep

Add **Podcasts**, **Songs**, and **Total Audio Count** count columns

```
# Calculate total audio counts and percentage of Podcasts & Songs per Customer
customer_audio_percentage = (history
                             .groupby(["Customer ID", "Audio Type"])
                             .size()
                             .unstack(fill_value=0)  # Convert to wide format
                             .assign(**{"Total Audio": lambda df: df.sum(axis=1)})
                             .assign(**{"Podcast %": lambda df: df["Podcast"] / df["Total Audio"]
                             .assign(**{"Song %": lambda df: df["Song"] / df["Total Audio"]})
                             .drop(["Podcast"], axis=1)
                             .rename(columns={"Song": "Total Songs"})
                             .reset_index())  # Reset index to make it a dataframe

customer_model = customer_model.merge(customer_audio_percentage, on="Customer ID", how="left")
customer_model[["Customer ID", "Podcast %", "Song %", "Total Songs", "Total Audio"]].head()
```

	Customer ID	Podcast %	Song %	Total Songs	Total Audio
0	5001	0.0	1.0	60	60
1	5002	0.0	1.0	22	22
2	5004	0.0	1.0	9	9
3	5267	0.0	1.0	45	45
4	5338	0.0	1.0	18	18
5	5404	0.0	1.0	8	8
6	5581	1.0	0.0	0	5
7	5759	0.0	1.0	15	15
8	5761	1.0	0.0	0	5

Exploratory Data Analysis Prep

Add count columns for each **Genre**

```
#Create columns for genre ratios

# Merge history with audio to get Genre information
genre_counts = (history
                .merge(audio, left_on="New Audio ID", right_on="ID", how="left")[["Customer ID", "Genre"]])

# Group by Customer ID and Genre, then count occurrences
customer_audio_percentage = (genre_counts
                             .groupby(["Customer ID", "Genre"])
                             .size()
                             .unstack(fill_value=0)  # Convert to wide format
                             .rename_axis(columns=None)  # Remove index name
                             .assign(Total=lambda df: df.sum(axis=1))  # Compute total listens per customer
                            )

# Convert to percentages
customer_audio_percentage = (customer_audio_percentage
                             .div(customer_audio_percentage["Total"], axis=0)
                             .drop(columns="Total")
                             .reset_index())

customer_audio_percentage = customer_audio_percentage.rename(columns={
    "Comedy": "Comedy %",
    "Country": "Country %",
    "Hip Hop": "Hip Hop %",
    "Jazz": "Jazz %",
    "Pop": "Pop %",
    "True Crime": "True Crime %"
})

customer_model = customer_model.merge(customer_audio_percentage, on="Customer ID", how="left")
customer_model[["Customer ID", "Podcast %", "Song %", "Total Audio"]].head()
```

	Customer ID	Comedy %	Country %	Hip Hop %	Jazz %	Pop %	True Crime %	Total Audio
0	5001	0.0	0.0	0.433333	0.0	0.566667	0.0	60
1	5002	0.0	1.0	0.000000	0.0	0.000000	0.0	22
2	5004	0.0	0.0	0.000000	0.0	1.000000	0.0	9
3	5267	0.0	0.0	0.488889	0.0	0.511111	0.0	45
4	5338	0.0	1.0	0.000000	0.0	0.000000	0.0	18

Exploratory Data Analysis Prep

MAVEN MUSIC

SINCE 1996

Add dummy variable columns for each **Subscription Plan**

```
#Create dummy variables for subscription plan column
subscription_plan_dummies = pd.get_dummies(customer_model["Subscription Plan"])
customer_model = pd.concat([customer_model, subscription_plan_dummies], axis=1)
customer_model[["Customer ID", "Basic (Ads)", "Premium (No Ads)"]].head()
```

	Customer ID	Basic (Ads)	Premium (No Ads)
0	5001	True	False
1	5002	True	False
2	5004	True	False
3	5267	True	False
4	5338	True	False

MAVEN MUSIC
SINCE 1996

Exploratory Data Analysis Prep

Drop columns to create a data model that is ready for analysis

```
#Drop columns that are now unnecessary
customer_model = customer_model.drop(["Subscription Plan", "Member Since", "Cancellation Date"], axis=1)
customer_model.head()
```

	Customer ID	Discount?	Cancelled?	Membership Duration (Days)	Session Count	Total Songs	Total Audio	Podcast %	Song %	Comedy %	Country %	Hip Hop %	Jazz %	Pop %	True Crime %	Basic (Ads)	Premium (No Ads)
0	5001	False	False	723.0	8	60	60	0.0	1.0	0.0	0.0	0.433333	0.0	0.566667	0.0	True	False
1	5002	False	False	723.0	4	22	22	0.0	1.0	0.0	1.0	0.000000	0.0	0.000000	0.0	True	False
2	5004	False	True	80.0	1	9	9	0.0	1.0	0.0	0.0	0.000000	0.0	1.000000	0.0	True	False
3	5267	False	False	716.0	7	45	45	0.0	1.0	0.0	0.0	0.488889	0.0	0.511111	0.0	True	False
4	5338	False	False	716.0	4	18	18	0.0	1.0	0.0	1.0	0.000000	0.0	0.000000	0.0	True	False

Exploratory Data Analysis

Maven Music would like to explore relationships between **customer cancellations** and the following:

- Customers with/without discounts
- Membership type (basic vs premium)
- Membership duration
- Audio type (podcasts vs songs)
- Song genre

Reminder

Correlation

does not equal

Causation

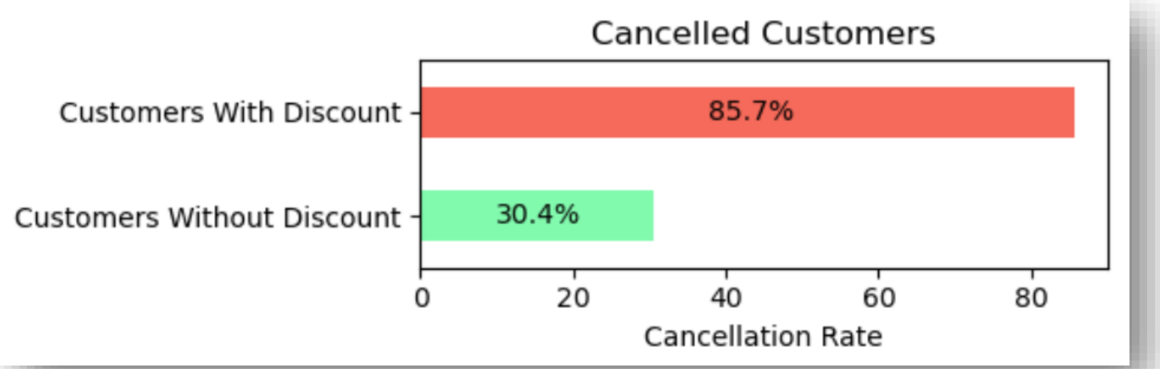
MAVEN MUSIC
SINCE 1996

Exploratory Data Analysis

Membership Discount Analysis

What effect did the membership discount have on membership cancellations?

- Members with a discount cancelled **almost three times more often** than those without a discount.




```python
# Cancellation rate for those who had a discount
customers_with_discount = customer_model[(customer_model["Discount?"] == True)]
cancellation_rate_with_discount = (customers_with_discount["Cancelled?"].sum() /
                                   customers_with_discount["Cancelled?"].count()) * 100

# Cancellation rate for those who did not have a discount
customers_without_discount = customer_model[(customer_model["Discount?"] == False)]
cancellation_rate_without_discount = (customers_without_discount["Cancelled?"].sum() /
                                      customers_without_discount["Cancelled?"].count()) * 100

# Create DataFrame
df = pd.DataFrame([
    ["Customers With Discount", cancellation_rate_with_discount],
    ["Customers Without Discount", cancellation_rate_without_discount]
], columns=["Customer Type", "Cancellation Rate"]).sort_values("Cancellation Rate", ascending=True)

# Define custom colors for each bar
colors = ["#82faae", "#f56a5b"]  # Green for discount, Red for no discount

# Plot with custom colors
fig, ax = plt.subplots(figsize=(6, 2))  # Adjust figure size
chart = df.plot.barh(
    title="Cancelled Customers",
    x="Customer Type",
    y="Cancellation Rate",
    color=colors,  # Assign colors
    xlabel="Cancellation Rate",
    ylabel='',
    legend=False,
    width=0.5,  # Adjust bar width to make bars closer
    ax=ax
)

# Add data labels
for container in chart.containers:
    chart.bar_label(container, fmt="%.1f%%", padding=1, label_type="center")

plt.tight_layout()  # Optimize layout
plt.show()
```

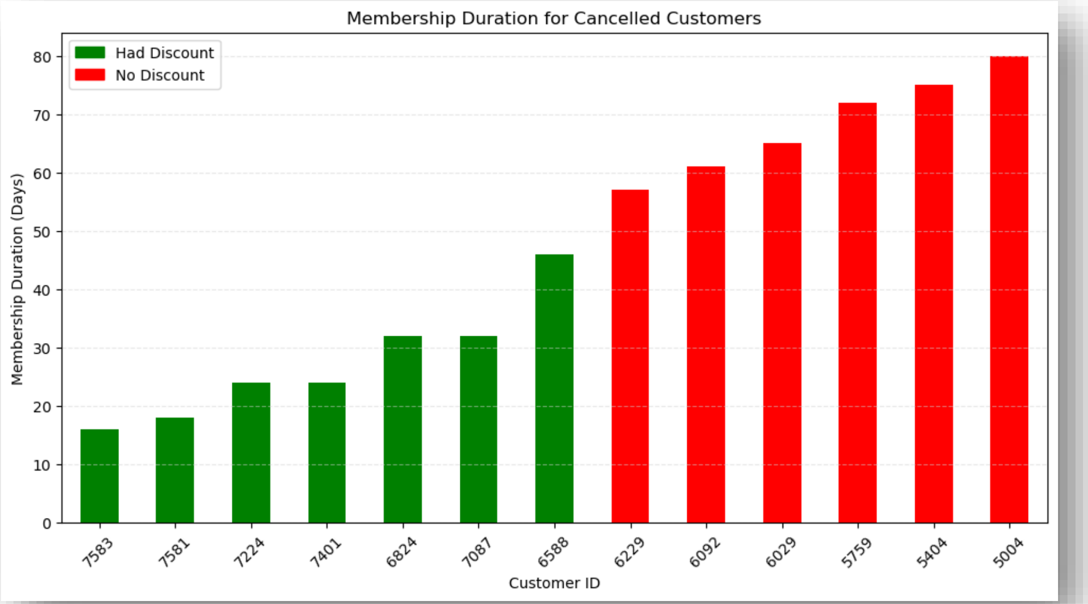
MAVEN MUSIC
SINCE 1996

Exploratory Data Analysis

Membership Discount Analysis

Did the discount affect when customers cancelled?

- Members with a discount cancelled **earlier** than those without a discount.




```
cancelled_customers = customer_model[customer_model["Cancelled?"] == True]

# Map colors based on 'Cancelled?' column
colors = (customer_model
          .loc[cancelled_customers.index, "Discount?"]
          .map({True: "red", False: "green"})
          .tolist()

# Sort customers by Membership Duration
sorted_customers = (cancelled_customers
                    .set_index("Customer ID")["Membership Duration (Days)"]
                    .sort_values(ascending=True))

# Create bar chart
fig, ax = plt.subplots(figsize=(12, 6))
sorted_customers.plot(kind="bar", color=colors, rot=45, ax=ax)

# Add labels and title and gridlines
ax.set_xlabel("Customer ID")
ax.set_ylabel("Membership Duration (Days)")
ax.set_title("Membership Duration for Cancelled Customers")
ax.yaxis.grid(True, linestyle="--", alpha=0.5, color="lightgray")

# Create legend
legend_patches = [
    mpatches.Patch(color="green", label="Had Discount"),
    mpatches.Patch(color="red", label="No Discount"),
]
ax.legend(handles=legend_patches, loc="upper left")

# Show plot
plt.show()
```

Exploratory Data Analysis

Membership Discount Analysis

NOTE:

This chart appears to show a clear separation between discounted and non-discounted customers in terms of when they cancel their memberships:

- Customers **with a discount** tend to cancel **within the first 50 days**.
- Customers **without a discount** appear to cancel **after 50 days.**

This pattern is heavily influenced by a limitation of the dataset:

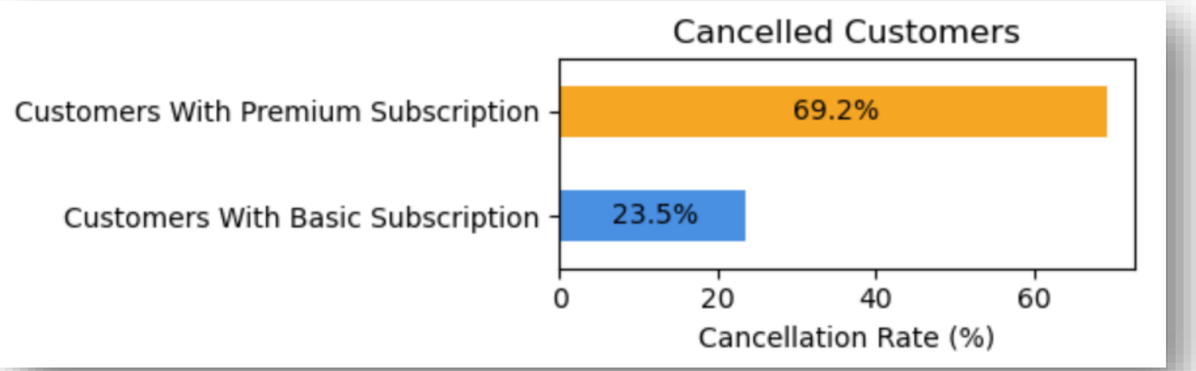
- The dataset is based on records generated on or before **May 31, 2023**.
- While cancelled customers have an associated "Cancellation Date", customers who appear to be "still active" may have cancelled after the dataset was last updated, and their cancellation simply isn't recorded.
- As a result, some of the non-discounted customers with longer membership durations may not actually be long-term retained users (they may have cancelled shortly after the data snapshot ended).

Exploratory Data Analysis

Membership Type Analysis

What effect did the membership type have on membership cancellations?

- Members with the premium subscription cancelled **almost three times more often** than those without a discount.




```python
# Cancellation rate for those who had the basic subscription
basic_customers = customer_model[(customer_model["Basic (Ads)"] == True)]
basic_customer_cancellation_rate = (basic_customers["Cancelled?"].sum() /
                                    basic_customers["Cancelled?"].count()) * 100

# Cancellation rate for those who had the premium subscription
premium_customers = customer_model[(customer_model["Premium (No Ads)"] == True)]
premium_customer_cancellation_rate = (premium_customers["Cancelled?"].sum() /
                                    premium_customers["Cancelled?"].count()) * 100

# Create DataFrame
df = pd.DataFrame([
    ["Customers With Basic Subscription", basic_customer_cancellation_rate],
    ["Customers With Premium Subscription", premium_customer_cancellation_rate]
], columns=["Subscription Type", "Cancellation Rate"]).sort_values("Cancellation Rate", ascending=True)

# Define custom colors for each bar
colors = ["#4A90E2", "#F5A623"]  # Green for discount, Red for no discount

# Plot with custom colors
fig, ax = plt.subplots(figsize=(6, 2))  # Adjust figure size
chart = df.plot.barh(
    title="Cancelled Customers",
    x="Subscription Type",
    y="Cancellation Rate",
    color=colors,  # Assign colors
    xlabel="Cancellation Rate (%)",
    ylabel='',
    legend=False,
    width=0.5,  # Adjust bar width to make bars closer
    ax=ax
)

# Add data labels
for container in chart.containers:
    chart.bar_label(container, fmt="%.1f%%", padding=1, label_type="center")

plt.tight_layout()  # Optimize layout
plt.show()
```

Exploratory Data Analysis

Audio Type Analysis

Did podcasts or songs make up the majority of **total** audio consumption?

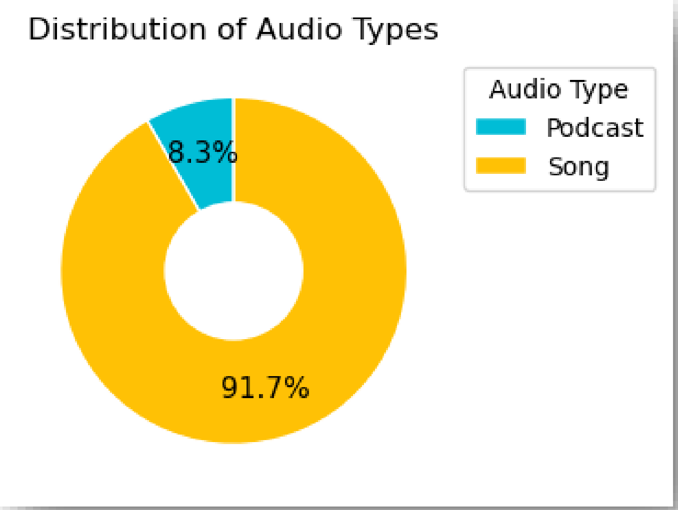
- In the data set, there were 525 instances of a customer listening to an audio track. 92% of the time that audio track was a song, not a podcast.

Note:

Customers who listen more frequently **influence the chart more** than those who listen less. If one power user listens to 1,000 songs and 10 podcasts, their behavior dominates the chart.

```
# Create a new DataFrame with the required columns
customer_audio_summary = (customer_model[["Customer ID", "Total Audio", "Total Songs"]]
                          .assign(**{"Total Podcast": lambda df: df["Total Audio"] - df["Total Songs"]})
                          .drop(["Total Audio", "Customer ID"], axis=1)
                          .sum()
                          .reset_index())

customer_audio_summary.columns = ["Audio Type", "Count"]
customer_audio_summary = customer_audio_summary.sort_values("Count")
customer_audio_summary.head(20)
```



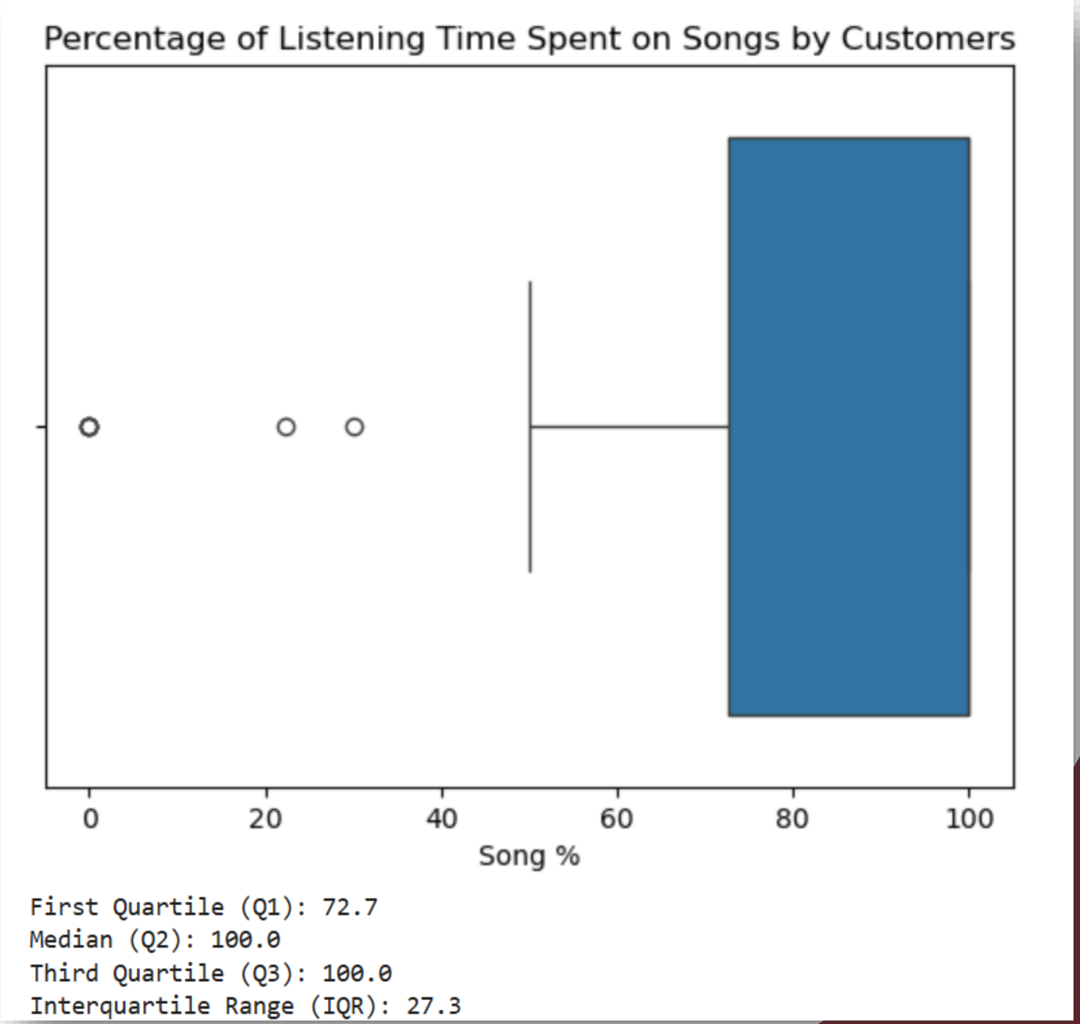
	Audio Type	Count
1	Total Podcast	42
0	Total Songs	463

Exploratory Data Analysis

Audio Type Analysis

How do customers split their listening between songs and podcasts?

- **First Quartile (Q1): 73%**
  - This means that only 25% of customers spend less than 73% of their listening time on songs.
- **Median (Q2): 100%**
  - The median is 1.0, meaning that at least 50% of customers listen exclusively to songs (100% of their listening time is spent on songs).
- **Third Quartile (Q3): 100%**
  - This suggests that 75% of customers spend at most 100% of their listening time on songs (which means many listen exclusively to songs).
- **Interquartile Range (IQR): 27.3%**
  - This measures the spread of the middle 50% of the data. This small IQR means that most customers have similar listening habits, with many listening almost entirely to songs.



Exploratory Data Analysis

MAVEN MUSIC
SINCE 1996

Audio Type Analysis – Code

```
# Calculate quartiles
q1 = (np.percentile(customer_model["Song %"], 25)* 100).round(1) # First quartile (Q1)
q2 = (np.percentile(customer_model["Song %"], 50)* 100).round(1)  # Median (Q2)
q3 = (np.percentile(customer_model["Song %"], 75)* 100).round(1)  # Third quartile (Q3)

# Calculate interquartile range (IQR)
iqr = (q3 - q1).round(1)

# Create boxplot
sns.boxplot(x=(customer_model["Song %"] * 100))
# Add title
plt.title("Percentage of Listening Time Spent on Songs by Customers")
# Show plot
plt.show()

# Print results
print(f"First Quartile (Q1): {q1}")
print(f"Median (Q2): {q2}")
print(f"Third Quartile (Q3): {q3}")
print(f"Interquartile Range (IQR): {iqr}")
```

MAVEN MUSIC
SINCE 1996

Exploratory Data Analysis

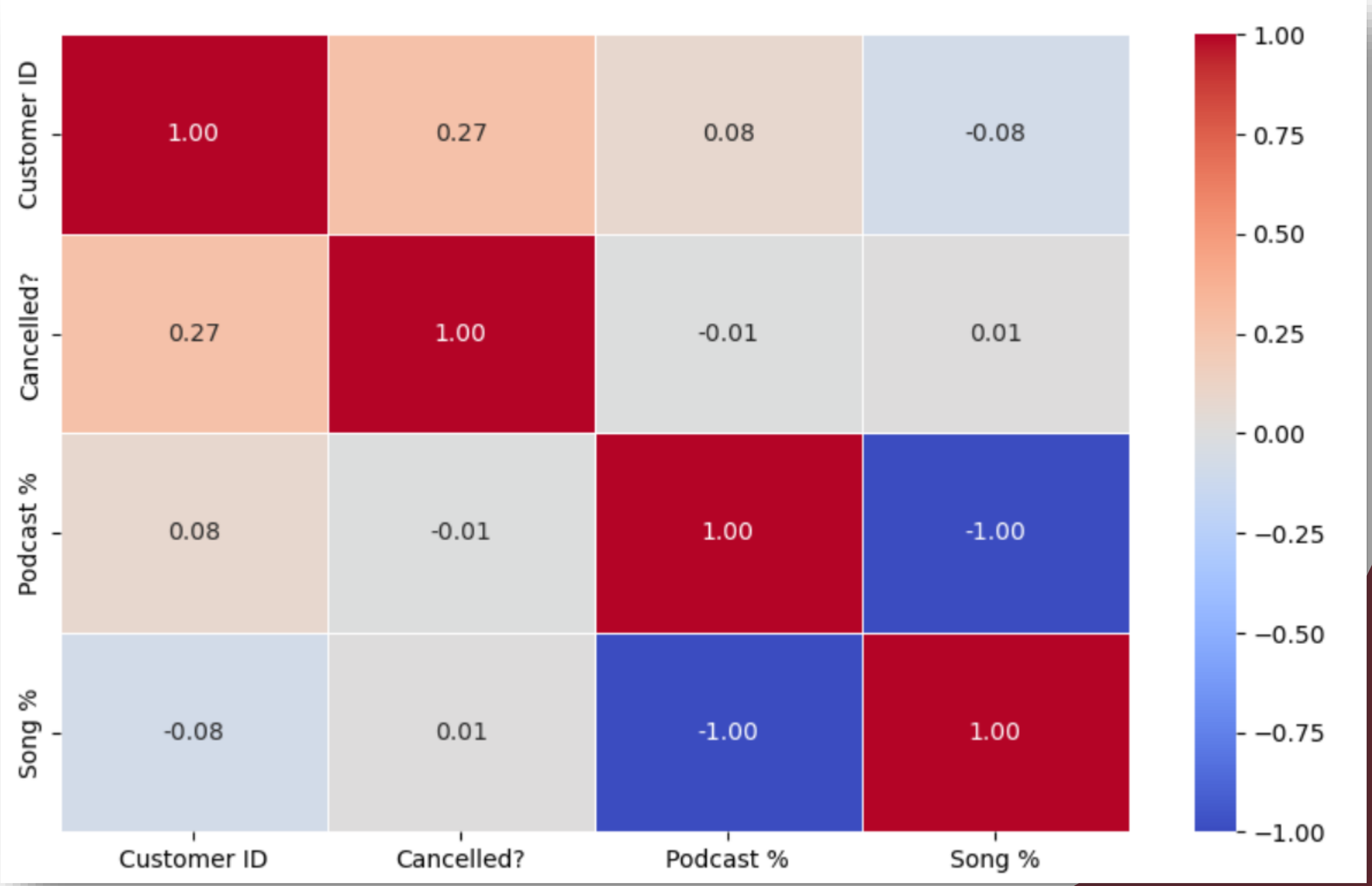
Audio Type Analysis – Correlation Table

Cancellations vs Podcast Listening %

- A value of -0.01 indicates that there is not a strong relationship between cancellations and the percentage of time customers spend listening to podcasts.

Cancellations vs Song Listening %

- A value of -0.01 indicates that there is not a strong relationship between cancellations and the percentage of time customers spend listening to songs.



Exploratory Data Analysis

Audio Type Analysis – Correlation Table Code

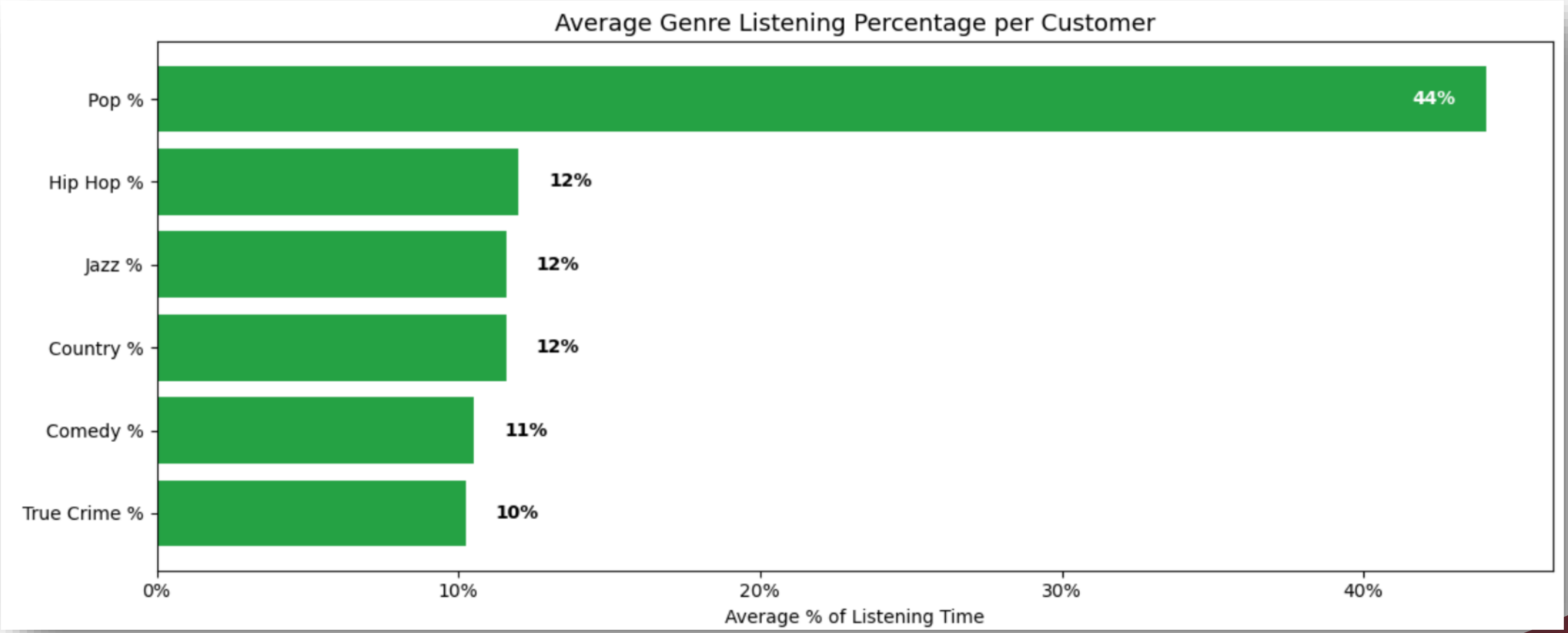
```
customer_model_type = customer_model[["Customer ID", "Cancelled?", "Podcast %", "Song %"]]
plt.figure(figsize=(10, 6))
sns.heatmap(customer_model_type.corr(), annot=True, cmap="coolwarm", fmt=".2f", linewidths=0.5)
plt.show()
```

Exploratory Data Analysis

Song Genre Analysis

How do customers split their listening across song genres?

- Pop music dominates listening behaviour, with customers choosing pop songs (on average) 44% of the time over other genres.



MAVEN MUSIC
SINCE 1996

Exploratory Data Analysis

Song Genre Analysis – Code

```
from matplotlib.ticker import FuncFormatter

genre_cols = ["Comedy %", "Country %", "Hip Hop %", "Jazz %", "Pop %", "True Crime %"]

average_genre_percentages = customer_model[genre_cols].mean().sort_values(ascending=True)

fig, ax = plt.subplots(figsize=(12, 5))
bars = ax.barh(average_genre_percentages.index, average_genre_percentages.values, color="#25A244")
for bar in bars:
    width = bar.get_width()
    label = f"{width:.0%}"
    if width < 0.15:  # If the bar is short, put label outside
        ax.text(width + 0.01, bar.get_y() + bar.get_height() / 2,
                label, va='center', fontsize=10, ha='left', fontweight="bold")
    else:  # Otherwise, put it inside
        ax.text(width - 0.01, bar.get_y() + bar.get_height() / 2,
                label, va='center', fontsize=10, ha='right', color="white", fontweight="bold")

ax.set_xlabel("Average % of Listening Time")
ax.set_title("Average Genre Listening Percentage per Customer", fontsize=13)
ax.xaxis.set_major_formatter(FuncFormatter(lambda x, _: f"{x * 100:.0f}%"))
plt.tight_layout()
plt.show()
```

Exploratory Data Analysis

Audio Type Analysis – Correlation Table

Cancellations vs Pop %

- A value of 0.54 indicates that there is a moderate positive relationship between cancellations and the percentage of time customers spend listening to pop music (as opposed to other genres).

Note:

- We know that, on average, customers choosing pop songs (on average) 44% of the time over other genres.
- Is this why the correlation is so high?

	Customer ID	Cancelled?	Comedy %	Country %	Hip Hop %	Jazz %	Pop %	True Crime %
Customer ID	1.00	0.27	0.34	-0.35	-0.09	0.41	-0.08	-0.17
Cancelled?	0.27	1.00	0.31	-0.34	-0.08	-0.34	0.54	-0.28
Comedy %	0.34	0.31	1.00	-0.17	0.31	-0.17	-0.35	-0.15
Country %	-0.35	-0.34	-0.17	1.00	-0.21	-0.12	-0.35	-0.13
Hip Hop %	-0.09	-0.08	0.31	-0.21	1.00	-0.21	-0.10	-0.24
Jazz %	0.41	-0.34	-0.17	-0.12	-0.21	1.00	-0.35	-0.13
Pop %	-0.08	0.54	-0.35	-0.35	-0.10	-0.35	1.00	-0.35
True Crime %	-0.17	-0.28	-0.15	-0.13	-0.24	-0.13	-0.35	1.00

Exploratory Data Analysis

Song Genre Analysis – Correlation Table Code

```
customer_model_genres = customer_model[["Customer ID", "Cancelled?", "Comedy %", "Country %", "Hip Hop %", "Jazz %", "Pop %", "True Crime %"]]
plt.figure(figsize=(10, 6))
sns.heatmap(customer_model_genres.corr(), annot=True, cmap="coolwarm", fmt=".2f", linewidths=0.5)
plt.show()
```

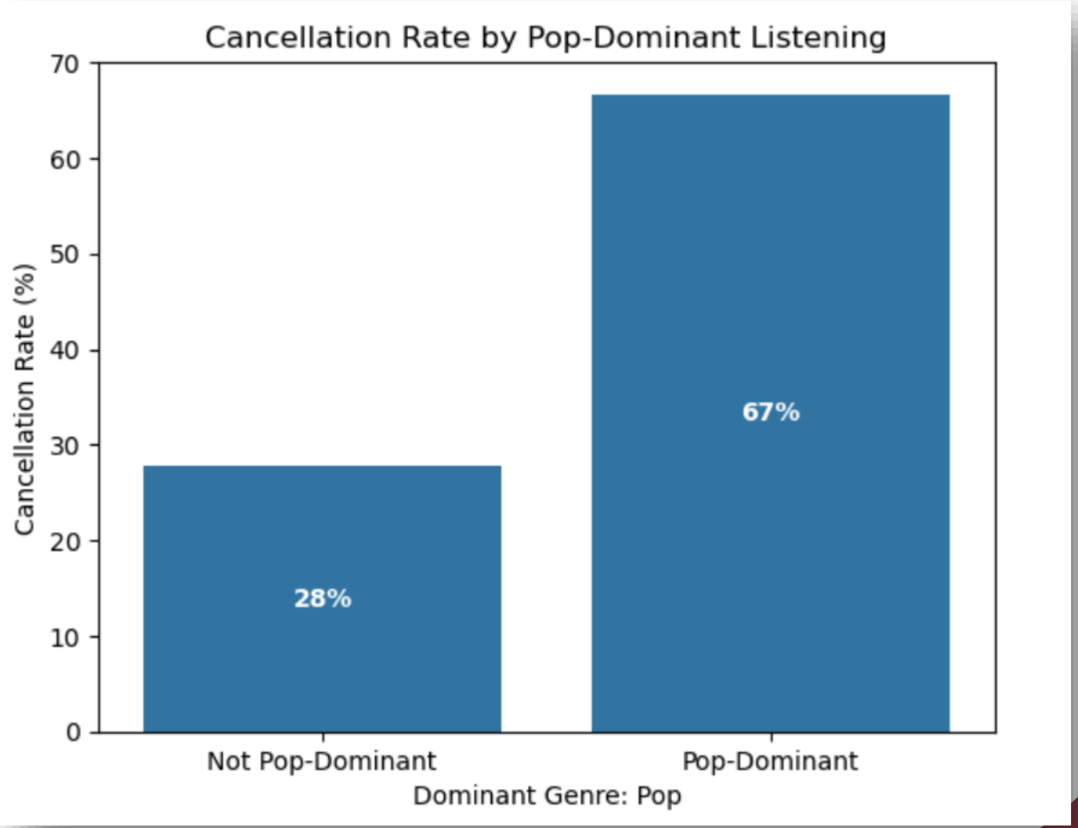
MAVEN MUSIC
SINCE 1996

Exploratory Data Analysis

Audio Type Analysis

Cancellations vs Pop %

- We know that, on average, customers choose pop songs (on average) 44% of the time over other genres. Is this why the correlation is so high?
  - Customers who predominantly listen to Pop music (more than 50% of the time) make up over half of cancellations.
  - This suggests Pop-dominant listening may be a sign of shallow engagement, which could increase the likelihood of cancellations.



Exploratory Data Analysis

Song Genre Analysis – Cancellations vs Pop % Code

```
customer_model["Dominant Genre: Pop"] = customer_model["Pop %"] > 0.50
pop_dominance_churn = customer_model.groupby("Dominant Genre: Pop")["Cancelled?"].mean()
ax = sns.barplot(
    x="Dominant Genre: Pop",
    y="Cancelled?",
    data=customer_model,
    estimator=lambda x: sum(x) / len(x) * 100,  # Convert to %
    errorbar=None
)
plt.ylabel("Cancellation Rate (%)")
plt.title("Cancellation Rate by Pop-Dominant Listening")
plt.xticks([0, 1], ["Not Pop-Dominant", "Pop-Dominant"])
for container in ax.containers:
    for bar in container:
        height = bar.get_height()
        ax.text(
            bar.get_x() + bar.get_width() / 2,
            height / 2,  # Midpoint of the bar
            f'{height:.0f}%',  # Label format
            ha='center', va='center', color='white', fontweight='bold'
        )
plt.show()
```

Exploratory Data Analysis Summary

MAVEN MUSIC
SINCE 1996

Subscription Type

Members with the premium subscription cancelled almost three times more often than those without a discount.

Subscription Discount

Members with a discount cancelled earlier than those without a discount, although this should be verified with a data set that is larger and more current.

Audio Type

Audio Type (song vs podcast) does not appear to have a relationship with cancellations.

Song Genre

Customers who predominantly listen to Pop music (more than 50% of the time) make up over half of cancellations.

MAVEN MUSIC

SINCE 1996

Prepare Data for Modeling

- Drop customerID column and other redundant columns
- Convert Boolean columns to integers

```
bool_cols = customer_model.select_dtypes(include="bool").columns
customer_model[bool_cols] = customer_model[bool_cols].astype(int)
customer_model.drop(columns=["Customer ID", "Song %", "Total Songs", "Dominant Genre: Pop"])
```

	Discount?	Cancelled?	Membership Duration (Days)	Total Audio	Podcast %	Total Sessions	Audio Per Session	Comedy %	Country %	Hip Hop %	Jazz %	Pop %	True Crime %	Basic (Ads)	Premium (No Ads)	Genre Diversity
0	0	0	79.0	60	0.000000	8	7.5	0.000000	0.000000	0.433333	0.000000	0.566667	0.000000	1	0	0.684232
1	0	0	79.0	22	0.000000	4	5.5	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	1	0	0.000000
2	0	1	80.0	9	0.000000	1	9.0	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	1	0	0.000000
3	0	0	72.0	45	0.000000	7	6.4	0.000000	0.000000	0.488889	0.000000	0.511111	0.000000	1	0	0.692900
4	0	0	72.0	18	0.000000	4	4.5	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	1	0	0.000000
5	0	1	75.0	8	0.000000	1	8.0	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	1	0	0.000000
6	0	0	71.0	5	1.000000	3	1.7	0.000000	0.000000	0.000000	0.000000	0.000000	1.000000	0	1	0.000000
7	0	1	72.0	15	0.000000	2	7.5	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0	1	0.000000
8	0	0	64.0	5	1.000000	3	1.7	0.000000	0.000000	0.000000	0.000000	0.000000	1.000000	0	1	0.000000
9	0	0	64.0	31	0.000000	6	5.2	0.000000	0.000000	0.354839	0.000000	0.645161	0.000000	1	0	0.650391
10	0	0	64.0	17	0.000000	3	5.7	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	1	0	0.000000
11	0	0	64.0	7	0.000000	1	7.0	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	1	0	0.000000
12	0	1	65.0	12	0.000000	2	6.0	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0	1	0.000000

The dataset has now been fully prepared and is ready to be used for predictive modeling.