­

**Advanced Data Acquisition: Churn Data**

Stephen E. Porter

College of Information Technology, Western Governors University

Dr. William Sewell

September 1, 2022

**Advance Data Acquisition: Churn Data**

In this paper, I will use the provided data set containing cleaned customer data from a fictional telecommunications company. To supplement the provided data, I have included an additional data set that is similar to the one provided to serve as a competitor to the first company. The data will be imported into a database using pgAdmin where the tables can be joined together to create new data sets. The primary purpose is to use these new data sets to create visualizations to highlight differences between the two companies and provide insights to leadership on how to retain customers.

# Part I: Data Dashboards

## A. Data Dashboards

Please see included tableau file.

## A1. Both Data Sets

Please see the included data csv files.

## A2. Dashboard Installation

I installed the dashboard by first connecting to the data sets. Let us call the provided data set Data Set 1 (or *customer*) and the additional data set Data Set 2 (or *competitor*). I connected to my both data sets in tableau through a localhost PostgreSQL connection. Data Set 2 was prepared for the subsequent join statements using Python so the *payment* and *contract* tables could be joined to it.

The Metrics Comparison dashboard has six visualizations on it (three from each data source) called Churn Count, Tenure vs Monthly Charge, and Contract Duration by Payment Type. The individual visualizations serve as filters for the others creating a dynamic page that allows users to answer many questions.

## A3. Dashboard Navigation

With the Tableau workbook opened, confirm the Metrics Comparison tab at the bottom is selected, then press the Presentation Mode button in the ribbon at the top. (See Figure 1).

Graphical user interface, application

Description automatically generated

***Figure 1 – Presentation View***

The data for Our Company is on the left-hand side, and the data for our Competitor is on the right-hand side. Hover your cursor over different parts of the visualizations to get a peek at more detailed information. (See Figure 2).

Timeline

Description automatically generated

***Figure 2 – Hover over unique customer to view details***

Click on different objects with each report to slice the information in different ways. on the right side to see information about a specific state. (See Figure 3)

Chart, scatter chart

Description automatically generated with medium confidence

***Figure 3 – Filter All on Churn = Yes***

You can select multiple states by pressing “CTRL” while clicking the states you’re interested in

A picture containing timeline

Description automatically generated

***Figure 3 – Filter All on Churn = Yes & Duration = Month-to-month***

Click on the Gender Legend to get a better view of each gender. (See Figure 5.)

Graphical user interface, application, PowerPoint

Description automatically generated

***Figure 5 – Scatterplots highlight Gender = Male***

## A4. SQL Code

-- Create Competitor Table (outside data source)

-- Table: public.competitor

-- DROP TABLE IF EXISTS public.competitor;

CREATE TABLE IF NOT EXISTS public.competitor

(

customer\_id text COLLATE pg\_catalog."default" NOT NULL,

gender text COLLATE pg\_catalog."default",

senior\_citizen integer,

partner text COLLATE pg\_catalog."default",

dependents text COLLATE pg\_catalog."default",

tenure integer,

phone\_service text COLLATE pg\_catalog."default",

multiple\_lines text COLLATE pg\_catalog."default",

internet\_service text COLLATE pg\_catalog."default",

online\_security text COLLATE pg\_catalog."default",

online\_backup text COLLATE pg\_catalog."default",

device\_protection text COLLATE pg\_catalog."default",

tech\_support text COLLATE pg\_catalog."default",

streaming\_tv text COLLATE pg\_catalog."default",

streaming\_movies text COLLATE pg\_catalog."default",

contract\_id integer,

paperless\_billing text COLLATE pg\_catalog."default",

payment\_id integer,

monthly\_charge text COLLATE pg\_catalog."default",

total\_charge text COLLATE pg\_catalog."default",

churn text COLLATE pg\_catalog."default",

CONSTRAINT competitor\_pkey PRIMARY KEY (customer\_id)

)

TABLESPACE pg\_default;

ALTER TABLE IF EXISTS public.competitor

OWNER to postgres;

-- Tenure v Monthly Charge (Us)

"SELECT CAST(""customer"".""customer\_id"" AS TEXT) AS ""customer\_id"",

CAST(""customer"".""gender"" AS TEXT) AS ""gender"",

CAST(""payment"".""payment\_type"" AS TEXT) AS ""payment\_type"",

SUM(""customer"".""monthly\_charge"") AS ""sum:monthly\_charge:ok"",

SUM(""customer"".""tenure"") AS ""sum:tenure:ok""

FROM ""public"".""customer"" ""customer""

LEFT JOIN ""public"".""payment"" ""payment"" ON (""customer"".""payment\_id"" = ""payment"".""payment\_id"")

GROUP BY 1,

2,

3"

-- Tenure v Monthly Charge (Them)

“SELECT MIN(CAST(CAST(""competitor"".""monthly\_charge"" AS TEXT) AS DOUBLE PRECISION)) AS ""TEMP(attr:monthly\_charge:qk)(1088457976)(0)"",

MAX(CAST(CAST(""competitor"".""monthly\_charge"" AS TEXT) AS DOUBLE PRECISION)) AS ""TEMP(attr:monthly\_charge:qk)(4083174111)(0)"",

CAST(""competitor"".""customer\_id"" AS TEXT) AS ""customer\_id"",

CAST(""competitor"".""gender"" AS TEXT) AS ""gender"",

CAST(""payment"".""payment\_type"" AS TEXT) AS ""payment\_type"",

SUM(""competitor"".""tenure"") AS ""sum:tenure:ok""

FROM ""public"".""competitor"" ""competitor""

LEFT JOIN ""public"".""contract"" ""contract"" ON (""competitor"".""contract\_id"" = ""contract"".""contract\_id"")

LEFT JOIN ""public"".""payment"" ""payment"" ON (""competitor"".""payment\_id"" = ""payment"".""payment\_id"")

WHERE ((CAST(""contract"".""duration"" AS TEXT) >= 'Month-to-month') AND (CAST(""contract"".""duration"" AS TEXT) <= 'Two Year'))

GROUP BY 3,

4,

5”

-- Payment Type by Contract (Us)

SELECT COUNT(DISTINCT CAST("competitor"."customer\_id" AS TEXT)) AS "ctd:customer\_id:ok",

CAST("contract"."duration" AS TEXT) AS "duration",

CAST("payment"."payment\_type" AS TEXT) AS "payment\_type"

FROM "public"."competitor" "competitor"

LEFT JOIN "public"."contract" "contract" ON ("competitor"."contract\_id" = "contract"."contract\_id")

LEFT JOIN "public"."payment" "payment" ON ("competitor"."payment\_id" = "payment"."payment\_id")

WHERE ((CAST("contract"."duration" AS TEXT) >= 'Month-to-month') AND (CAST("contract"."duration" AS TEXT) <= 'Two Year'))

GROUP BY 2, 3

-- Payment Type by Contract (Them)

SELECT COUNT(DISTINCT CAST("competitor"."customer\_id" AS TEXT)) AS "ctd:customer\_id:ok",

CAST("contract"."duration" AS TEXT) AS "duration",

CAST("payment"."payment\_type" AS TEXT) AS "payment\_type"

FROM "public"."competitor" "competitor"

LEFT JOIN "public"."contract" "contract" ON ("competitor"."contract\_id" = "contract"."contract\_id")

LEFT JOIN "public"."payment" "payment" ON ("competitor"."payment\_id" = "payment"."payment\_id")

WHERE ((CAST("contract"."duration" AS TEXT) >= 'Month-to-month') AND (CAST("contract"."duration" AS TEXT) <= 'Two Year'))

GROUP BY 2, 3

-- Churn Count (Us)

SELECT CAST("customer"."churn" AS TEXT) AS "churn",

COUNT(DISTINCT CAST("customer"."customer\_id" AS TEXT)) AS "ctd:customer\_id:ok"

FROM "public"."customer" "customer"

GROUP BY 1

-- Churn Count (Them)

SELECT CAST("competitor"."churn" AS TEXT) AS "churn",

COUNT(DISTINCT CAST("competitor"."customer\_id" AS TEXT)) AS "ctd:customer\_id:ok"

FROM "public"."competitor" "competitor"

LEFT JOIN "public"."contract" "contract" ON ("competitor"."contract\_id" = "contract"."contract\_id")

WHERE ((CAST("contract"."duration" AS TEXT) >= 'Month-to-month') AND (CAST("contract"."duration" AS TEXT) <= 'Two Year'))

GROUP BY 1

# Part II: Demonstration

## B. Panapto Demonstration

Please see attached Panapto video.

## B1. Technical Environment

[This is the written version of the explanation I plan to give in the video demonstration. It is also the reasoning for C2 in Part III of the performance task.]

Tableau was chosen as the technical environment to create the data visualizations. It is powerful and flexible, allowing the user to connect to a data source and create numerous data visualizations without needing to have a preconstructed plan for the final visualization. (Loth, Vogel, Sparkes 1). This was a key feature that allowed me to create increasingly useful data visualizations through trial and error, and I could then select the best ones for the story I wished to tell with these data.

## B2. Demonstrate Dashboard Functionality

Please see attached Panapto video

## B3. SQL Scripts

The SQL code provided in section A4 demonstrates how the dashboards were created in Tableau. They include casting columns as datatypes, joining tables together on primary and foreign keys, and aggregations with grouping to create totals for various categories.

## B4. Data Streams

The data is brought into tableau directly by the PostgreSQL server. The database connection was then switched to Extract (over Live) as the data in this simulation is static which allows for easier sharing of the dashboard (so the viewer does not need access to the database themselves).

## B5. Data Points

The data points in the scatter plots were aligned with fixed, common axes so their differences can be easily determined. An obvious difference in the data sets is in the range of monthly charge values. For our company, the range is approximately $80 to $300 while the range for our competitor is approximately $18 to $120. By scaling the data to use the same intervals along the axes, the stark difference between the ranges is clearly visible.

## B6. Database Creation

A single Postgres database was created so that a single connection could be made between the PostgreSQL server and Tableau for ease of access to the data. The customer, payment, contract, job, and location tables were provided, and the competitor table was created using the SQL script provided in Section B3. The table was then joined to the provided contract and payment tables using a one-to-many relationship as shown in the ERD in Section B7 below.

## B7. Referential Integrity

The ERD below demonstrates the referential integrity of the database.

Text

Description automatically generated with low confidence

# Part III: Report

## C1. Dashboard Alignment

The dashboard was designed to highlight the key metrics between our company and a competing company. Principal component analysis performed in a previous course identified Tenure and Monthly Charge as two variables worth tracking. The Payment Method by Contract graphs were included because the external data was joined to the *contract* and *payment* tables in the database (as shown in B7). The Churn bar chart is included as that is the main concern of our analysis – which characteristics are most closely tied to a customer’s decision to discontinue service. The visualizations serve as filters for each other (both *customer* and *competitor* data sets) so it is easy to compare the two companies.

## C2. Business Intelligence Tool

Tableau was chosen as the technical environment to create the data visualizations. It is powerful and flexible, allowing the user to connect to a data source and create numerous data visualizations without needing to have a preconstructed plan for the final visualization. (Loth, Vogel, Sparkes 1). This was a key feature that allowed me to create increasingly useful data visualizations through trial and error, and I could then select the best ones for the story I wished to tell with these data.

## C3. Data Cleaning

The Data Set 2 (acquired from Kaggle) was used in my previous course D210. The database contains no null values an only required minor adjustments to allow it to be joined to the tables that make up Data Set 1 (provided by the course). The python script I utilized to prepare the data is given below:

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

import pandas as pd

df = pd.read\_csv('C:\\WGU\\D211 Advanced Data Acquisition\\tables\\WA\_Fn-UseC\_-Telco-Customer-Churn.csv')

print(df.info())

# print list of column names to confirm which columns need to be modified & how they are named.

for col in df.columns:

print(col)

# columns to replace: Contract & PaymentMethod

# print value counts to view values & their spelling

print(df.Contract.value\_counts())

df.Contract.replace(['Month-to-month', 'One year', 'Two year'], [1, 2, 3], inplace=True)

# print new value counts to confirm the replace took correctly

print(df.Contract.value\_counts())

# print value counts to view values & their spelling

print(df.PaymentMethod.value\_counts())

df.PaymentMethod.replace(['Bank transfer (automatic)', 'Credit card (automatic)', 'Electronic check', 'Mailed check'],

[1, 2, 3, 4], inplace=True)

# print new value counts to confirm the replace took correctly

print(df.PaymentMethod.value\_counts())

# export to csv to be imported into database as table

df.to\_csv('C:\\WGU\\D211 Advanced Data Acquisition\\tables\\competitor\_churn\_scraped.csv')

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

## C4. Dashboard Creation

Every worksheet within the dashboard serves as a filter for the other worksheets in the same dashboard so there are many ways that users could manipulate the view of these data. A simple bar chart was created for each company to show the count of users who have churned and those who have not and colored by churn. Seeing characteristics related to customer churn is the main purpose of this analysis, so including a visualization for this variable was critical.

Tenure and Monthly Charge were identified during Principal Component Analysis and fit well as a scatterplot. These variables were plotted with Tenure on the *x*-axis and Monthly Charge on the *y*-axis. These were then colored by Gender to add an additional characteristic to analyze (knowing that the Churn bar chart will be used as a filter for all worksheets on the dashboard).

Payment Type by Contract is another bar chart of counting the customers in each contract (month-to-month, one year, or two year) and then colored by payment type.

Each of these reports was made for both companies for a total of six worksheets. All were added to the same dashboard for easy comparison.

## C5. Data Analysis Results

The ratio of customers who have churned to those who have not for each company is nearly identical so, while the size of each data set is not the same, the distribution should allow for a reasonable comparison to be drawn.

When filtering on those customers who have churned, it is clear that in both companies customers with a low tenure but high monthly charge are more likely to churn than those who have a higher tenure and lower charge. The focus should be on ways to reduce the cost for new customers and sustain that reduced cost over time. Gender does not appear to be a significant factor in customers who have churned – when filtering on the Gender category, the distributions appear similar.

Payment Type by Contract is an interesting bar chart. It not only highlights that the majority of customers are on a month-to-month contract, but also make their payments manually (as opposed to automatic). Analyzing for churn relationship, Month-to-month customers are significantly more likely to churn as are customers who make manual payments (Mailed Check and Electronic Check).

Actionable items would be to look for ways to reduce the monthly charge for our customers, and to get them to join one or two year contracts instead of month-to-month.

## C6. Analysis Limitations

A limitation of this analysis is that it takes a narrow view of only some of the available data in the database. The Principal Component Analysis indicated that Tenure and Monthly Charge were worth analyzing and the scatterplot provided valuable insights. We were also able to review Contract type and Payment Type as an additional factor that could be related to churn. However, with many more variables available there could be other insights left unidentified. A notably useful, but missing, variable would be geographic location (like city/state/area). This data was not available in the external data set so a comparison was not possible. The breakdown of churning customers could vary differently state by state for several factors (strong competition from a local company, poor infrastructure/service for our company). This could provide more targeted action items for our company if a comparison was possible.

## D. Web Sources & E. Sources

**References**

Kaggle.com. 2022. Telco Customer Churn. [online] Available at: <https://www.kaggle.com/datasets/blastchar/telco-customer-churn?resource=download> [Accessed 8 August 2022].

Loth, A., Vogel, N., &amp; Sparkes, S. (2019). Visual analytics with tableau. John Wiley &amp; Sons, Inc.

Spicer, T. (2019, July 12). 7 steps to export SQL statements from Tableau. Medium. Retrieved September 7, 2022, from https://blog.openbridge.com/7-steps-to-export-sql-statements-from-tableau-7e51a2fd4277