

# Churn Analysis Dashboard Installation and Navigation

William Stults

1/12/2023

Western Governors University

---

The purpose of this document is to aid the user in the installation of the Tableau Software and the Churn Analysis dashboard.

## Download and install Tableau

Begin by downloading the Tableau Installer, found at this web address:

<https://www.tableau.com/products/desktop/download>. Once downloaded, run the installer and accept the defaults.

---

## Obtain data sources and prepare

We will need two sources of data to install the churn analysis dashboard. The first data source, the "churn" database, is provided by Western Governors University as part of the Lab On Demand environment, accessible via PgAdmin. The second data source, NST-EST2022-POPCHG2020\_2022.csv, can be downloaded from the following location:

[https://www2.census.gov/programs-surveys/popest/datasets/2020-2022/state/totals/NST-EST2022-POPCHG2020\\_2022.csv](https://www2.census.gov/programs-surveys/popest/datasets/2020-2022/state/totals/NST-EST2022-POPCHG2020_2022.csv).

Before utilizing the second data source, we will need to do some preparation. This will involve creating a table in the churn database to house the data from NST-EST2022-POPCHG2020\_2022.csv, importing the data, and then executing SQL statements to prepare the data for use.

The create table script for the new population table is shown below.

```
-- Table: public.population

-- DROP TABLE public.population;

CREATE TABLE public.population
(
    "SUMLEV" integer NOT NULL,
    "REGION" text COLLATE pg_catalog."default" NOT NULL,
    "DIVISION" text COLLATE pg_catalog."default" NOT NULL,
    "STATE" integer NOT NULL,
```

```

    "NAME" text COLLATE pg_catalog."default" NOT NULL,
    "ESTIMATESBASE2020" integer NOT NULL,
    "POPESTIMATE2020" integer NOT NULL,
    "POPESTIMATE2021" integer NOT NULL,
    "POPESTIMATE2022" integer NOT NULL,
    "NPOPCHG_2020" integer NOT NULL,
    "NPOPCHG_2021" integer NOT NULL,
    "NPOPCHG_2022" integer NOT NULL,
    "PPOPCHG_2020" numeric NOT NULL,
    "PPOPCHG_2021" numeric NOT NULL,
    "PPOPCHG_2022" numeric NOT NULL,
    "NRANK_ESTBASE2020" text COLLATE pg_catalog."default" NOT NULL,
    "NRANK_POPEST2020" text COLLATE pg_catalog."default" NOT NULL,
    "NRANK_POPEST2021" text COLLATE pg_catalog."default" NOT NULL,
    "NRANK_POPEST2022" text COLLATE pg_catalog."default" NOT NULL,
    "NRANK_NPCHG2020" text COLLATE pg_catalog."default" NOT NULL,
    "NRANK_NPCHG2021" text COLLATE pg_catalog."default" NOT NULL,
    "NRANK_NPCHG2022" text COLLATE pg_catalog."default" NOT NULL,
    "NRANK_PPCHG2020" text COLLATE pg_catalog."default" NOT NULL,
    "NRANK_PPCHG2021" text COLLATE pg_catalog."default" NOT NULL,
    "NRANK_PPCHG2022" text COLLATE pg_catalog."default" NOT NULL,
)
TABLESPACE pg_default;

ALTER TABLE public.population
OWNER to postgres;

```

--

Once created, you may right-click the new table and select "import/export" from the menu to utilize PgAdmin's built in tool to import the data from NST-EST2022-POPCHG2020\_2022.csv. Note that the SQL command for this is shown below, but attempting this in the Lab On Demand environment generates a permissions related error when attempting to read the file.

```

COPY public.population ("SUMLEV", "REGION", "DIVISION", "STATE", "NAME",
"ESTIMATESBASE2020", "POPESTIMATE2020", "POPESTIMATE2021",
"POPESTIMATE2022", "NPOPCHG_2020", "NPOPCHG_2021", "NPOPCHG_2022",
"PPOPCHG_2020", "PPOPCHG_2021", "PPOPCHG_2022", "NRANK_ESTBASE2020",
"NRANK_POPEST2020", "NRANK_POPEST2021", "NRANK_POPEST2022",
"NRANK_NPCHG2020", "NRANK_NPCHG2021", "NRANK_NPCHG2022",
"NRANK_PPCHG2020", "NRANK_PPCHG2021", "NRANK_PPCHG2022") FROM
'C:\Users\LabUser\Desktop\NST-EST2022-POPCHG2020_2022.CSV' DELIMITER ','
CSV HEADER;
```

Trim the table by deleting rows containing data on regions rather than states.

```
DELETE FROM public.population WHERE "STATE" = 0
```

Create a new column "STATE\_ABB" and assign it the values in the "NAME" column.

```
UPDATE population SET "STATE_ABB" = "NAME"
```

Execute the update statements below to transform state names in the STATE\_ABB column to state name abbreviations.

```
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Alabama', 'AL');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Alaska', 'AK');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Arizona', 'AZ');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Arkansas', 'AR');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'California', 'CA');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Canal Zone', 'CZ');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Colorado', 'CO');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Connecticut', 'CT');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Delaware', 'DE');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'District of Columbia', 'DC');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Florida', 'FL');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Georgia', 'GA');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Guam', 'GU');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Hawaii', 'HI');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Idaho', 'ID');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Illinois', 'IL');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Indiana', 'IN');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Iowa', 'IA');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Kansas', 'KS');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Kentucky', 'KY');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Louisiana', 'LA');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Maine', 'ME');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Maryland', 'MD');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Massachusetts', 'MA');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Michigan', 'MI');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Minnesota', 'MN');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Mississippi', 'MS');
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Missouri', 'MO');
```

```
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Montana',  
'MT');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Nebraska',  
'NE');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Nevada', 'NV');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'New Hampshire',  
'NH');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'New Jersey',  
'NJ');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'New Mexico',  
'NM');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'New York',  
'NY');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'North  
Carolina', 'NC');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'North Dakota',  
'ND');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Ohio', 'OH');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Oklahoma',  
'OK');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Oregon', 'OR');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Pennsylvania',  
'PA');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Puerto Rico',  
'PR');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Rhode Island',  
'RI');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'South  
Carolina', 'SC');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'South Dakota',  
'SD');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Tennessee',  
'TN');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Texas', 'TX');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Utah', 'UT');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Vermont',  
'VT');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Virgin  
Islands', 'VI');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Virginia',  
'VA');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Washington',  
'WA');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'West VA',  
'WV');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Wisconsin',  
'WI');  
UPDATE population SET "STATE_ABB" = replace("STATE_ABB", 'Wyoming',  
'WY');
```

---

# Import data sources

Launch the Tableau application. In the left-hand panel of the screen which is colored blue, look for the section named "To a Server". Click the More... option and select PostgreSQL. Enter the appropriate details and click "Sign In". Drag the "customer" table to the right hand pane.

With the "customer" table selected in the right pane, open the "Data" menu on the top toolbar and select "Convert to Custom SQL". Paste in the SQL shown below, then click "Ok".

```
SELECT "customer"."age" AS "age",
"customer"."bandwidth_gp_year" AS "bandwidth_gp_year",
"customer"."children" AS "children",
CAST("customer"."churn" AS TEXT) AS "churn",
"customer"."contacts" AS "contacts",
"customer"."contract_id" AS "contract_id",
CAST("customer"."customer_id" AS TEXT) AS "customer_id",
"customer"."email" AS "email",
CAST("customer"."gender" AS TEXT) AS "gender",
"customer"."income" AS "income",
"customer"."job_id" AS "job_id",
"customer"."lat" AS "lat",
"customer"."lng" AS "lng",
"customer"."location_id" AS "location_id",
CAST("customer"."marital" AS TEXT) AS "marital",
"customer"."monthly_charge" AS "monthly_charge",
"customer"."outage_sec_week" AS "outage_sec_week",
"customer"."payment_id" AS "payment_id",
"customer"."population" AS "population",
CAST("customer"."port_modem" AS TEXT) AS "port_modem",
CAST("customer"."tablet" AS TEXT) AS "tablet",
CAST("customer"."techie" AS TEXT) AS "techie",
"customer"."tenure" AS "tenure",
"customer"."yearly_equip_faiure" AS "yearly_equip_faiure",
CAST("location"."city" AS TEXT) AS "city",
CAST("location"."county" AS TEXT) AS "county",
"location"."location_id" AS "location_id (location)",
CAST("location"."state" AS TEXT) AS "state",
"location"."zip" AS "zip",
"population"."POPESTIMATE2022" AS "POPESTIMATE2022",
CAST("population"."STATE_ABB" AS TEXT) AS "STATE_ABB"
FROM "public"."customer" "customer"
INNER JOIN "public"."location"
ON "location"."location_id" = "customer"."location_id"
INNER JOIN "public"."population"
ON "population"."STATE_ABB" = "location"."state"
```

Click "Update Now" in the lower right pane to view the results of the join.

---

# Create calculated fields

Once all the data is imported we will create some calculated fields in Tableau. To create each calculated field, begin by clicking the analysis menu at the top of the screen and selecting "create calculated field". Name each calculated field according to the instructions below.

For each calculated field, a brief description and the code needed to create the field will be included.

## Churn Percent

The churn percent field is a calculation of the percentage of customers that churned.

```
SUM([Churn Yes Count])/COUNT([Churn])
```

## Churn Yes Count

The churn yes count field gives us a count of the customers that churned.

```
IF [Churn]='Yes' THEN 1  
END
```

## Pct of State

The pct of state field gives us a percent value for how many customers in that state are subscribers compared to the total population estimate.

```
COUNT([State])/MAX([POPESTIMATE2022])
```

---

## Build Worksheets

For each new worksheet (four total), begin by clicking the "New Worksheet" button at the bottom.

### Worksheet 1 - Lowest Subscriber to Population Ratios

1. Drag "State" to the columns shelf. From its dropdown menu, select Measure > Count.
2. Drag "Pct of State" to the first position on the columns shelf.
3. Drag "State" to the rows shelf.
4. Drag "State" to the Filters shelf. From the "Top" tab, enable the "By field:" radio button. From the first dropdown menu, select bottom. In the field to the right, type "20". From the second dropdown menu, select "Pct of State". Then click Apply, and OK.
5. Drag "State" to the Color tile on the marks shelf. Click the Color tile and choose "Edit Colors". From the "Select Color Palette" dropdown, choose "Color Blind". Click "Assign Palette", then Apply, and then OK.

The underlying SQL for Worksheet 1 is shown below.

```
SELECT COUNT(CAST("Custom SQL Query1"."state" AS TEXT)) AS
"TEMP(Calculation_242912952012296194)(858378402)(0)",
CAST("Custom SQL Query1"."state" AS TEXT) AS "state"
FROM (
    SELECT CAST("location"."city" AS TEXT) AS "city",
        CAST("location"."county" AS TEXT) AS "county",
        "location"."location_id" AS "location_id (location)",
        CAST("location"."state" AS TEXT) AS "state",
        "location"."zip" AS "zip"
    FROM "public"."location" "location"
) "Custom SQL Query1"
INNER JOIN (
    SELECT "t0"."state" AS "state",
        (CASE WHEN "t2"."__measure_1" = 0 THEN NULL ELSE
        CAST(COALESCE("t1"."__measure_0", 0) AS DOUBLE PRECISION) /
        "t2"."__measure_1" END) AS "$__alias_0"
    FROM (
        SELECT CAST("Custom SQL Query1"."state" AS TEXT) AS "state"
        FROM (
            SELECT CAST("location"."city" AS TEXT) AS "city",
                CAST("location"."county" AS TEXT) AS "county",
                "location"."location_id" AS "location_id (location)",
                CAST("location"."state" AS TEXT) AS "state",
                "location"."zip" AS "zip"
            FROM "public"."location" "location"
        ) "Custom SQL Query1"
        GROUP BY 1
    ) "t0"
    LEFT JOIN (
        SELECT CAST("Custom SQL Query1"."state" AS TEXT) AS "state",
            COUNT(CAST("Custom SQL Query1"."state" AS TEXT)) AS "__measure_0"
        FROM (
            SELECT CAST("location"."city" AS TEXT) AS "city",
                CAST("location"."county" AS TEXT) AS "county",
                "location"."location_id" AS "location_id (location)",
                CAST("location"."state" AS TEXT) AS "state",
                "location"."zip" AS "zip"
            FROM "public"."location" "location"
        ) "Custom SQL Query1"
        GROUP BY 1
    ) "t1" ON ("t0"."state" IS NOT DISTINCT FROM "t1"."state")
    LEFT JOIN (
        SELECT CAST("Custom SQL Query1"."state" AS TEXT) AS "state",
            MAX("Custom SQL Query"."POPESTIMATE2022") AS "__measure_1"
        FROM (
            SELECT "population"."POPESTIMATE2022" AS "POPESTIMATE2022",
                CAST("population"."STATE_ABB" AS TEXT) AS "STATE_ABB"
            FROM "public"."population" "population"
        ) "Custom SQL Query"
        LEFT JOIN (
```

```

        SELECT CAST("location"."city" AS TEXT) AS "city",
               CAST("location"."county" AS TEXT) AS "county",
               "location"."location_id" AS "location_id (location)",
               CAST("location"."state" AS TEXT) AS "state",
               "location"."zip" AS "zip"
        FROM "public"."location" "location"
    ) "Custom SQL Query1" ON (CAST("Custom SQL Query"."STATE_ABB" AS
TEXT) = CAST("Custom SQL Query1"."state" AS TEXT))
        GROUP BY 1
    ) "t2" ON ("t0"."state" IS NOT DISTINCT FROM "t2"."state")
ORDER BY 2 ASC NULLS FIRST,
        1 ASC NULLS FIRST
LIMIT 20
) "t3" ON (CAST("Custom SQL Query1"."state" AS TEXT) IS NOT DISTINCT FROM
"t3"."state")
GROUP BY 2

```

```

SELECT MAX("Custom SQL Query"."POPESTIMATE2022") AS
"TEMP(Calculation_242912952012296194)(3105768215)(0)",
       CAST("Custom SQL Query1"."state" AS TEXT) AS "state"
FROM (
    SELECT "population"."POPESTIMATE2022" AS "POPESTIMATE2022",
           CAST("population"."STATE_ABB" AS TEXT) AS "STATE_ABB"
    FROM "public"."population" "population"
) "Custom SQL Query"
LEFT JOIN (
    SELECT CAST("location"."city" AS TEXT) AS "city",
           CAST("location"."county" AS TEXT) AS "county",
           "location"."location_id" AS "location_id (location)",
           CAST("location"."state" AS TEXT) AS "state",
           "location"."zip" AS "zip"
    FROM "public"."location" "location"
) "Custom SQL Query1" ON (CAST("Custom SQL Query"."STATE_ABB" AS TEXT) =
CAST("Custom SQL Query1"."state" AS TEXT))
        INNER JOIN (
            SELECT "t0"."state" AS "state",
                   (CASE WHEN "t2"."__measure_1" = 0 THEN NULL ELSE
CAST(COALESCE("t1"."__measure_0", 0) AS DOUBLE PRECISION) /
"t2"."__measure_1" END) AS "$__alias_0"
        FROM (
            SELECT CAST("Custom SQL Query1"."state" AS TEXT) AS "state"
        FROM (
            SELECT CAST("location"."city" AS TEXT) AS "city",
                   CAST("location"."county" AS TEXT) AS "county",
                   "location"."location_id" AS "location_id (location)",
                   CAST("location"."state" AS TEXT) AS "state",
                   "location"."zip" AS "zip"
            FROM "public"."location" "location"
        ) "Custom SQL Query1"
        GROUP BY 1
    ) "t0"

```

```

LEFT JOIN (
    SELECT CAST("Custom SQL Query1"."state" AS TEXT) AS "state",
        COUNT(CAST("Custom SQL Query1"."state" AS TEXT)) AS "__measure_0"
    FROM (
        SELECT CAST("location"."city" AS TEXT) AS "city",
            CAST("location"."county" AS TEXT) AS "county",
            "location"."location_id" AS "location_id (location)",
            CAST("location"."state" AS TEXT) AS "state",
            "location"."zip" AS "zip"
        FROM "public"."location" "location"
    ) "Custom SQL Query1"
    GROUP BY 1
) "t1" ON ("t0"."state" IS NOT DISTINCT FROM "t1"."state")
LEFT JOIN (
    SELECT CAST("Custom SQL Query1"."state" AS TEXT) AS "state",
        MAX("Custom SQL Query". "POPESTIMATE2022") AS "__measure_1"
    FROM (
        SELECT "population"."POPESTIMATE2022" AS "POPESTIMATE2022",
            CAST("population"."STATE_ABB" AS TEXT) AS "STATE_ABB"
        FROM "public"."population" "population"
    ) "Custom SQL Query"
    LEFT JOIN (
        SELECT CAST("location"."city" AS TEXT) AS "city",
            CAST("location"."county" AS TEXT) AS "county",
            "location"."location_id" AS "location_id (location)",
            CAST("location"."state" AS TEXT) AS "state",
            "location"."zip" AS "zip"
        FROM "public"."location" "location"
    ) "Custom SQL Query1" ON (CAST("Custom SQL Query". "STATE_ABB" AS TEXT) = CAST("Custom SQL Query1"."state" AS TEXT))
    GROUP BY 1
) "t2" ON ("t0"."state" IS NOT DISTINCT FROM "t2"."state")
ORDER BY 2 ASC NULLS FIRST,
    1 ASC NULLS FIRST
LIMIT 20
) "t3" ON (CAST("Custom SQL Query1"."state" AS TEXT) IS NOT DISTINCT FROM
"t3"."state")
GROUP BY 2

```

```

SELECT "t0"."state" AS "state",
    (CASE WHEN "t2"."__measure_1" = 0 THEN NULL ELSE
    CAST(COALESCE("t1"."__measure_0", 0) AS DOUBLE PRECISION) /
    "t2"."__measure_1" END) AS "$__alias_0"
FROM (
    SELECT CAST("Custom SQL Query1"."state" AS TEXT) AS "state"
    FROM (
        SELECT CAST("location"."city" AS TEXT) AS "city",
            CAST("location"."county" AS TEXT) AS "county",
            "location"."location_id" AS "location_id (location)",
            CAST("location"."state" AS TEXT) AS "state",
            "location"."zip" AS "zip"
    )
)
```

```

        FROM "public"."location" "location"
    ) "Custom SQL Query1"
    GROUP BY 1
) "t0"
LEFT JOIN (
    SELECT CAST("Custom SQL Query1"."state" AS TEXT) AS "state",
        COUNT(CAST("Custom SQL Query1"."state" AS TEXT)) AS "__measure_0"
    FROM (
        SELECT CAST("location"."city" AS TEXT) AS "city",
            CAST("location"."county" AS TEXT) AS "county",
            "location"."location_id" AS "location_id (location)",
            CAST("location"."state" AS TEXT) AS "state",
            "location"."zip" AS "zip"
        FROM "public"."location" "location"
    ) "Custom SQL Query1"
    GROUP BY 1
) "t1" ON ("t0"."state" IS NOT DISTINCT FROM "t1"."state")
LEFT JOIN (
    SELECT CAST("Custom SQL Query1"."state" AS TEXT) AS "state",
        MAX("Custom SQL Query". "POPESTIMATE2022") AS "__measure_1"
    FROM (
        SELECT "population". "POPESTIMATE2022" AS "POPESTIMATE2022",
            CAST("population". "STATE_ABB" AS TEXT) AS "STATE_ABB"
        FROM "public". "population" "population"
    ) "Custom SQL Query"
    LEFT JOIN (
        SELECT CAST("location". "city" AS TEXT) AS "city",
            CAST("location". "county" AS TEXT) AS "county",
            "location". "location_id" AS "location_id (location)",
            CAST("location". "state" AS TEXT) AS "state",
            "location". "zip" AS "zip"
        FROM "public". "location" "location"
    ) "Custom SQL Query1" ON (CAST("Custom SQL Query". "STATE_ABB" AS TEXT)
= CAST("Custom SQL Query1". "state" AS TEXT))
    GROUP BY 1
) "t2" ON ("t0". "state" IS NOT DISTINCT FROM "t2". "state")
ORDER BY 2 ASC NULLS FIRST,
    1 ASC NULLS FIRST
LIMIT 20

```

## Worksheet 2 - Churn % By State

1. Drag "Churn" to the "Columns" shelf.
2. Drag "Churn" to the "Rows" shelf. From its dropdown menu, highlight "Measure" and select "Count".
3. Click "Show Me" in the top right corner, and select "pie charts".
4. In the "Marks" panel, click the "Size" icon (two circles) next to the second "CNT(Churn)" and choose Label. Open that same "CNT(Churn)" dropdown menu, highlight "Quick Table Calculation" and choose "Percent of Total".

5. Drag "Churn" to the "Label" tile in the "Marks" panel.
6. Drag "State" to the "Filters" shelf in the first position. From the filter's settings menu, choose "Single Value (dropdown)".

The underlying SQL for Worksheet 2 is shown below.

```

SELECT CAST("Custom SQL Query2"."churn" AS TEXT) AS "churn",
       COUNT(CAST("Custom SQL Query2"."churn" AS TEXT)) AS "cnt:churn:ok"
  FROM (
    SELECT "customer"."age" AS "age",
           "customer"."bandwidth_gp_year" AS "bandwidth_gp_year",
           "customer"."children" AS "children",
           CAST("customer"."churn" AS TEXT) AS "churn",
           "customer"."contacts" AS "contacts",
           "customer"."contract_id" AS "contract_id",
           CAST("customer"."customer_id" AS TEXT) AS "customer_id",
           "customer"."email" AS "email",
           CAST("customer"."gender" AS TEXT) AS "gender",
           "customer"."income" AS "income",
           "customer"."job_id" AS "job_id",
           "customer"."lat" AS "lat",
           "customer"."lng" AS "lng",
           "customer"."location_id" AS "location_id",
           CAST("customer"."marital" AS TEXT) AS "marital",
           "customer"."monthly_charge" AS "monthly_charge",
           "customer"."outage_sec_week" AS "outage_sec_week",
           "customer"."payment_id" AS "payment_id",
           "customer"."population" AS "population",
           CAST("customer"."port_modem" AS TEXT) AS "port_modem",
           CAST("customer"."tablet" AS TEXT) AS "tablet",
           CAST("customer"."techie" AS TEXT) AS "techie",
           "customer"."tenure" AS "tenure",
           "customer"."yearly_equip_faiure" AS "yearly_equip_faiure",
           CAST("location"."city" AS TEXT) AS "city",
           CAST("location"."county" AS TEXT) AS "county",
           "location"."location_id" AS "location_id (location)",
           CAST("location"."state" AS TEXT) AS "state",
           "location"."zip" AS "zip",
           "population"."POPESTIMATE2022" AS "POPESTIMATE2022",
           CAST("population"."STATE_ABB" AS TEXT) AS "STATE_ABB"
  FROM "public"."customer" "customer"
  INNER JOIN "public"."location"
  ON "location"."location_id" = "customer"."location_id"
  INNER JOIN "public"."population"
  ON "population"."STATE_ABB" = "location"."state"
 ) "Custom SQL Query2"
 GROUP BY 1

```

```

SELECT CAST("Custom SQL Query1"."state" AS TEXT) AS "state"
FROM (
    SELECT "customer"."age" AS "age",
        "customer"."bandwidth_gp_year" AS "bandwidth_gp_year",
        "customer"."children" AS "children",
        CAST("customer"."churn" AS TEXT) AS "churn",
        "customer"."contacts" AS "contacts",
        "customer"."contract_id" AS "contract_id",
        CAST("customer"."customer_id" AS TEXT) AS "customer_id",
        "customer"."email" AS "email",
        CAST("customer"."gender" AS TEXT) AS "gender",
        "customer"."income" AS "income",
        "customer"."job_id" AS "job_id",
        "customer"."lat" AS "lat",
        "customer"."lng" AS "lng",
        "customer"."location_id" AS "location_id",
        CAST("customer"."marital" AS TEXT) AS "marital",
        "customer"."monthly_charge" AS "monthly_charge",
        "customer"."outage_sec_week" AS "outage_sec_week",
        "customer"."payment_id" AS "payment_id",
        "customer"."population" AS "population",
        CAST("customer"."port_modem" AS TEXT) AS "port_modem",
        CAST("customer"."tablet" AS TEXT) AS "tablet",
        CAST("customer"."techie" AS TEXT) AS "techie",
        "customer"."tenure" AS "tenure",
        "customer"."yearly_equip_faiure" AS "yearly_equip_faiure",
        CAST("location"."city" AS TEXT) AS "city",
        CAST("location"."county" AS TEXT) AS "county",
        "location"."location_id" AS "location_id (location)",
        CAST("location"."state" AS TEXT) AS "state",
        "location"."zip" AS "zip",
        "population"."POPESTIMATE2022" AS "POPESTIMATE2022",
        CAST("population"."STATE_ABB" AS TEXT) AS "STATE_ABB"
    FROM "public"."customer" "customer"
    INNER JOIN "public"."location"
    ON "location"."location_id" = "customer"."location_id"
    INNER JOIN "public"."population"
    ON "population"."STATE_ABB" = "location"."state"
) "Custom SQL Query2"
LEFT JOIN (
    SELECT CAST("location"."city" AS TEXT) AS "city",
        CAST("location"."county" AS TEXT) AS "county",
        "location"."location_id" AS "location_id (location)",
        CAST("location"."state" AS TEXT) AS "state",
        "location"."zip" AS "zip"
    FROM "public"."location" "location"
) "Custom SQL Query1" ON ("Custom SQL Query2"."location_id" = "Custom SQL
Query1"."location_id (location)")
WHERE ("Custom SQL Query1"."location_id (location)" IS NULL)
LIMIT 1

```

```

SELECT CAST("Custom SQL Query2"."churn" AS TEXT) AS "churn"
FROM (
    SELECT "customer"."age" AS "age",
        "customer"."bandwidth_gp_year" AS "bandwidth_gp_year",
        "customer"."children" AS "children",
        CAST("customer"."churn" AS TEXT) AS "churn",
        "customer"."contacts" AS "contacts",
        "customer"."contract_id" AS "contract_id",
        CAST("customer"."customer_id" AS TEXT) AS "customer_id",
        "customer"."email" AS "email",
        CAST("customer"."gender" AS TEXT) AS "gender",
        "customer"."income" AS "income",
        "customer"."job_id" AS "job_id",
        "customer"."lat" AS "lat",
        "customer"."lng" AS "lng",
        "customer"."location_id" AS "location_id",
        CAST("customer"."marital" AS TEXT) AS "marital",
        "customer"."monthly_charge" AS "monthly_charge",
        "customer"."outage_sec_week" AS "outage_sec_week",
        "customer"."payment_id" AS "payment_id",
        "customer"."population" AS "population",
        CAST("customer"."port_modem" AS TEXT) AS "port_modem",
        CAST("customer"."tablet" AS TEXT) AS "tablet",
        CAST("customer"."techie" AS TEXT) AS "techie",
        "customer"."tenure" AS "tenure",
        "customer"."yearly_equip_faiure" AS "yearly_equip_faiure",
        CAST("location"."city" AS TEXT) AS "city",
        CAST("location"."county" AS TEXT) AS "county",
        "location"."location_id" AS "location_id (location)",
        CAST("location"."state" AS TEXT) AS "state",
        "location"."zip" AS "zip",
        "population"."POPESTIMATE2022" AS "POPESTIMATE2022",
        CAST("population"."STATE_ABB" AS TEXT) AS "STATE_ABB"
    FROM "public"."customer" "customer"
    INNER JOIN "public"."location"
    ON "location"."location_id" = "customer"."location_id"
    INNER JOIN "public"."population"
    ON "population"."STATE_ABB" = "location"."state"
) "Custom SQL Query2"
GROUP BY 1

```

```

SELECT CAST("Custom SQL Query1"."state" AS TEXT) AS "state"
FROM (
    SELECT CAST("location"."city" AS TEXT) AS "city",
        CAST("location"."county" AS TEXT) AS "county",
        "location"."location_id" AS "location_id (location)",
        CAST("location"."state" AS TEXT) AS "state",
        "location"."zip" AS "zip"
    FROM "public"."location" "location"
) "Custom SQL Query1"
GROUP BY 1

```

ORDER BY 1 ASC NULLS FIRST

```
SELECT COUNT(CAST("Custom SQL Query2"."churn" AS TEXT)) AS "cnt:churn:ok"
FROM (
    SELECT "customer"."age" AS "age",
    "customer"."bandwidth_gp_year" AS "bandwidth_gp_year",
    "customer"."children" AS "children",
    CAST("customer"."churn" AS TEXT) AS "churn",
    "customer"."contacts" AS "contacts",
    "customer"."contract_id" AS "contract_id",
    CAST("customer"."customer_id" AS TEXT) AS "customer_id",
    "customer"."email" AS "email",
    CAST("customer"."gender" AS TEXT) AS "gender",
    "customer"."income" AS "income",
    "customer"."job_id" AS "job_id",
    "customer"."lat" AS "lat",
    "customer"."lng" AS "lng",
    "customer"."location_id" AS "location_id",
    CAST("customer"."marital" AS TEXT) AS "marital",
    "customer"."monthly_charge" AS "monthly_charge",
    "customer"."outage_sec_week" AS "outage_sec_week",
    "customer"."payment_id" AS "payment_id",
    "customer"."population" AS "population",
    CAST("customer"."port_modem" AS TEXT) AS "port_modem",
    CAST("customer"."tablet" AS TEXT) AS "tablet",
    CAST("customer"."techie" AS TEXT) AS "techie",
    "customer"."tenure" AS "tenure",
    "customer"."yearly_equip_faiure" AS "yearly_equip_faiure",
    CAST("location"."city" AS TEXT) AS "city",
    CAST("location"."county" AS TEXT) AS "county",
    "location"."location_id" AS "location_id (location)",
    CAST("location"."state" AS TEXT) AS "state",
    "location"."zip" AS "zip",
    "population"."POPESTIMATE2022" AS "POPESTIMATE2022",
    CAST("population"."STATE_ABB" AS TEXT) AS "STATE_ABB"
    FROM "public"."customer" "customer"
    INNER JOIN "public"."location"
    ON "location"."location_id" = "customer"."location_id"
    INNER JOIN "public"."population"
    ON "population"."STATE_ABB" = "location"."state"
) "Custom SQL Query2"
HAVING (COUNT(1) > 0)
```

## Worksheet 3 - States: Churn % With Total Population

1. Drag "Churn Percent" to the "Columns" shelf.
2. Drag "POPESTIMATE2022" to the "Rows" shelf.

3. Drag "State" to the "Color" tile on the "Marks" shelf. If prompted, choose "Add all members".

Click the "Color" tile and choose "Edit Colors". From the "Select Color Palette" dropdown, choose "Color Blind". Click "Assign Palette", then Apply, and then OK.

4. From the main dropdown menu in the "Marks" shelf, choose "Circle".

5. Click the "Size" tile in the "Marks" shelf and slide the slider to the halfway point.

The underlying SQL for Worksheet 3 is shown below.

```
SELECT "t1"."state" AS "state",
       "t1"."sum:POPESTIMATE2022:ok" AS "sum:POPESTIMATE2022:ok"
    FROM (
        SELECT CAST("t0"."state" AS TEXT) AS "state",
               SUM("Custom SQL Query"."POPESTIMATE2022") AS "sum:POPESTIMATE2022:ok"
            FROM (
                SELECT "population"."POPESTIMATE2022" AS "POPESTIMATE2022",
                       CAST("population"."STATE_ABB" AS TEXT) AS "STATE_ABB"
                  FROM "public"."population" "population"
            ) "Custom SQL Query"
        LEFT JOIN (
            SELECT CAST("Custom SQL Query1"."state" AS TEXT) AS "state"
            FROM (
                SELECT CAST("location"."city" AS TEXT) AS "city",
                       CAST("location"."county" AS TEXT) AS "county",
                       "location"."location_id" AS "location_id (location)",
                       CAST("location"."state" AS TEXT) AS "state",
                       "location"."zip" AS "zip"
                  FROM "public"."location" "location"
            ) "Custom SQL Query1"
        GROUP BY 1
        ) "t0" ON (CAST("Custom SQL Query"."STATE_ABB" AS TEXT) = "t0"."state")
    GROUP BY 1
) "t1"
INNER JOIN (
    SELECT CAST("Custom SQL Query1"."state" AS TEXT) AS "state",
           SUM((CASE WHEN (CAST("Custom SQL Query2"."churn" AS TEXT) = 'Yes')
           THEN 1 ELSE NULL END)) AS "__measure_0",
           COUNT(CAST("Custom SQL Query2"."churn" AS TEXT)) AS "__measure_1"
    FROM (
        SELECT "customer"."age" AS "age",
               "customer"."bandwidth_gp_year" AS "bandwidth_gp_year",
               "customer"."children" AS "children",
               CAST("customer"."churn" AS TEXT) AS "churn",
               "customer"."contacts" AS "contacts",
               "customer"."contract_id" AS "contract_id",
               CAST("customer"."customer_id" AS TEXT) AS "customer_id",
               "customer"."email" AS "email",
               CAST("customer"."gender" AS TEXT) AS "gender",
               "customer"."income" AS "income",
               "customer"."job_id" AS "job_id",
               "customer"."lat" AS "lat",
```

```

"customer"."lng" AS "lng",
"customer"."location_id" AS "location_id",
CAST("customer"."marital" AS TEXT) AS "marital",
"customer"."monthly_charge" AS "monthly_charge",
"customer"."outage_sec_week" AS "outage_sec_week",
"customer"."payment_id" AS "payment_id",
"customer"."population" AS "population",
CAST("customer"."port_modem" AS TEXT) AS "port_modem",
CAST("customer"."tablet" AS TEXT) AS "tablet",
CAST("customer"."techie" AS TEXT) AS "techie",
"customer"."tenure" AS "tenure",
"customer"."yearly_equip_faiure" AS "yearly_equip_faiure",
CAST("location"."city" AS TEXT) AS "city",
CAST("location"."county" AS TEXT) AS "county",
"location"."location_id" AS "location_id (location)",
CAST("location"."state" AS TEXT) AS "state",
"location"."zip" AS "zip",
"population"."POPESTIMATE2022" AS "POPESTIMATE2022",
CAST("population"."STATE_ABB" AS TEXT) AS "STATE_ABB"
FROM "public"."customer" "customer"
INNER JOIN "public"."location"
ON "location"."location_id" = "customer"."location_id"
INNER JOIN "public".population
ON "population"."STATE_ABB" = "location"."state"
) "Custom SQL Query2"
LEFT JOIN (
SELECT CAST("location"."city" AS TEXT) AS "city",
CAST("location"."county" AS TEXT) AS "county",
"location"."location_id" AS "location_id (location)",
CAST("location"."state" AS TEXT) AS "state",
"location"."zip" AS "zip"
FROM "public"."location" "location"
) "Custom SQL Query1" ON ("Custom SQL Query2"."location_id" = "Custom
SQL Query1"."location_id (location)")
GROUP BY 1
HAVING (NOT ((CASE WHEN COUNT(CAST("Custom SQL Query2"."churn" AS
TEXT)) = 0 THEN NULL ELSE CAST(SUM((CASE WHEN (CAST("Custom SQL
Query2"."churn" AS TEXT) = 'Yes') THEN 1 ELSE NULL END)) AS DOUBLE
PRECISION) / COUNT(CAST("Custom SQL Query2"."churn" AS TEXT)) END) IS
NULL))
) "t2" ON ("t1"."state" IS NOT DISTINCT FROM "t2"."state")
)

```

```

SELECT "t0"."state" AS "state"
FROM (
  SELECT CAST("Custom SQL Query1"."state" AS TEXT) AS "state"
  FROM (
    SELECT CAST("location"."city" AS TEXT) AS "city",
    CAST("location"."county" AS TEXT) AS "county",
    "location"."location_id" AS "location_id (location)",
    CAST("location"."state" AS TEXT) AS "state",
    "location"."zip" AS "zip"
  )
)
```

```

        FROM "public"."location" "location"
    ) "Custom SQL Query1"
    GROUP BY 1
) "t0"
INNER JOIN (
    SELECT CAST("Custom SQL Query1"."state" AS TEXT) AS "state",
        SUM((CASE WHEN (CAST("Custom SQL Query2"."churn" AS TEXT) = 'Yes')
    THEN 1 ELSE NULL END)) AS "__measure_0",
        COUNT(CAST("Custom SQL Query2"."churn" AS TEXT)) AS "__measure_1"
    FROM (
        SELECT "customer"."age" AS "age",
            "customer"."bandwidth_gp_year" AS "bandwidth_gp_year",
            "customer"."children" AS "children",
            CAST("customer"."churn" AS TEXT) AS "churn",
            "customer"."contacts" AS "contacts",
            "customer"."contract_id" AS "contract_id",
            CAST("customer"."customer_id" AS TEXT) AS "customer_id",
            "customer"."email" AS "email",
            CAST("customer"."gender" AS TEXT) AS "gender",
            "customer"."income" AS "income",
            "customer"."job_id" AS "job_id",
            "customer"."lat" AS "lat",
            "customer"."lng" AS "lng",
            "customer"."location_id" AS "location_id",
            CAST("customer"."marital" AS TEXT) AS "marital",
            "customer"."monthly_charge" AS "monthly_charge",
            "customer"."outage_sec_week" AS "outage_sec_week",
            "customer"."payment_id" AS "payment_id",
            "customer"."population" AS "population",
            CAST("customer"."port_modem" AS TEXT) AS "port_modem",
            CAST("customer"."tablet" AS TEXT) AS "tablet",
            CAST("customer"."techie" AS TEXT) AS "techie",
            "customer"."tenure" AS "tenure",
            "customer"."yearly_equip_faiure" AS "yearly_equip_faiure",
            CAST("location"."city" AS TEXT) AS "city",
            CAST("location"."county" AS TEXT) AS "county",
            "location"."location_id" AS "location_id (location)",
            CAST("location"."state" AS TEXT) AS "state",
            "location"."zip" AS "zip",
            "population"."POPESTIMATE2022" AS "POPESTIMATE2022",
            CAST("population"."STATE_ABB" AS TEXT) AS "STATE_ABB"
        FROM "public"."customer" "customer"
        INNER JOIN "public"."location"
        ON "location"."location_id" = "customer"."location_id"
        INNER JOIN "public"."population"
        ON "population"."STATE_ABB" = "location"."state"
    ) "Custom SQL Query2"
    LEFT JOIN (
        SELECT CAST("location"."city" AS TEXT) AS "city",
            CAST("location"."county" AS TEXT) AS "county",
            "location"."location_id" AS "location_id (location)",
            CAST("location"."state" AS TEXT) AS "state",
            "location"."zip" AS "zip"
    )

```

```

        FROM "public"."location" "location"
    ) "Custom SQL Query1" ON ("Custom SQL Query2"."location_id" = "Custom
SQL Query1"."location_id (location)")
    GROUP BY 1
    HAVING (NOT ((CASE WHEN COUNT(CAST("Custom SQL Query2"."churn" AS
TEXT)) = 0 THEN NULL ELSE CAST(SUM((CASE WHEN (CAST("Custom SQL
Query2"."churn" AS TEXT) = 'Yes') THEN 1 ELSE NULL END)) AS DOUBLE
PRECISION) / COUNT(CAST("Custom SQL Query2"."churn" AS TEXT)) END) IS
NULL))
) "t1" ON ("t0"."state" IS NOT DISTINCT FROM "t1"."state")

```

```

SELECT COUNT(CAST("Custom SQL Query2"."churn" AS TEXT)) AS
"TEMP(Calculation_242912952012042241)(3605431932)(0)",
SUM((CASE WHEN (CAST("Custom SQL Query2"."churn" AS TEXT) = 'Yes') THEN
1 ELSE NULL END)) AS "TEMP(Calculation_242912952012042241)(670882048)
(0)",
CAST("Custom SQL Query1"."state" AS TEXT) AS "state"
FROM (
    SELECT "customer"."age" AS "age",
    "customer"."bandwidth_gp_year" AS "bandwidth_gp_year",
    "customer"."children" AS "children",
    CAST("customer"."churn" AS TEXT) AS "churn",
    "customer"."contacts" AS "contacts",
    "customer"."contract_id" AS "contract_id",
    CAST("customer"."customer_id" AS TEXT) AS "customer_id",
    "customer"."email" AS "email",
    CAST("customer"."gender" AS TEXT) AS "gender",
    "customer"."income" AS "income",
    "customer"."job_id" AS "job_id",
    "customer"."lat" AS "lat",
    "customer"."lng" AS "lng",
    "customer"."location_id" AS "location_id",
    CAST("customer"."marital" AS TEXT) AS "marital",
    "customer"."monthly_charge" AS "monthly_charge",
    "customer"."outage_sec_week" AS "outage_sec_week",
    "customer"."payment_id" AS "payment_id",
    "customer"."population" AS "population",
    CAST("customer"."port_modem" AS TEXT) AS "port_modem",
    CAST("customer"."tablet" AS TEXT) AS "tablet",
    CAST("customer"."techie" AS TEXT) AS "techie",
    "customer"."tenure" AS "tenure",
    "customer"."yearly_equip_faiure" AS "yearly_equip_faiure",
    CAST("location"."city" AS TEXT) AS "city",
    CAST("location"."county" AS TEXT) AS "county",
    "location"."location_id" AS "location_id (location)",
    CAST("location"."state" AS TEXT) AS "state",
    "location"."zip" AS "zip",
    "population"."POPESTIMATE2022" AS "POPESTIMATE2022",
    CAST("population"."STATE_ABB" AS TEXT) AS "STATE_ABB"
FROM "public"."customer" "customer"
INNER JOIN "public"."location"

```

```
ON "location"."location_id" = "customer"."location_id"
INNER JOIN "public"."population"
ON "population"."STATE_ABB" = "location"."state"
) "Custom SQL Query2"
LEFT JOIN (
SELECT CAST("location"."city" AS TEXT) AS "city",
CAST("location"."county" AS TEXT) AS "county",
"location"."location_id" AS "location_id (location)",
CAST("location"."state" AS TEXT) AS "state",
"location"."zip" AS "zip"
FROM "public"."location" "location"
) "Custom SQL Query1" ON ("Custom SQL Query2"."location_id" = "Custom SQL
Query1"."location_id (location)")
GROUP BY 3
HAVING (NOT ((CASE WHEN COUNT(CAST("Custom SQL Query2"."churn" AS TEXT))
= 0 THEN NULL ELSE CAST(SUM((CASE WHEN (CAST("Custom SQL Query2"."churn"
AS TEXT) = 'Yes') THEN 1 ELSE NULL END)) AS DOUBLE PRECISION) /
COUNT(CAST("Custom SQL Query2"."churn" AS TEXT)) END) IS NULL))
```

---

## Create the Dashboard

We are now ready to create the dashboard. Click the "new dashboard" button along the bottom toolbar. You may drag each worksheet from the sheets panel over to the right side to fill the space in the dashboard.

If you would like floating filters, open the settings dropdown for each filter and choose "floating", then drag each filter wherever you would like it to appear in the dashboard. You may also resize the filters if you wish.

---

## Navigating the Dashboard

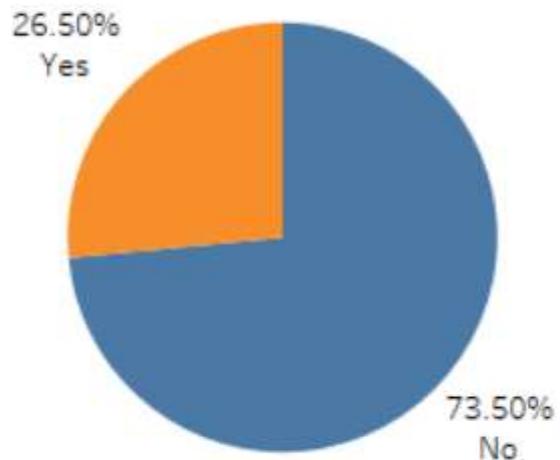
All portions of the dashboard are immediately visible, so no actions need to be taken to reveal hidden sheets or views. There is one sheet in the dashboard which utilizes a dropdown filter: "Churn % By State".

The "Churn % By State" sheet can be filtered by state to narrow data presented to an individual state's churn percentage.

## Churn % By State

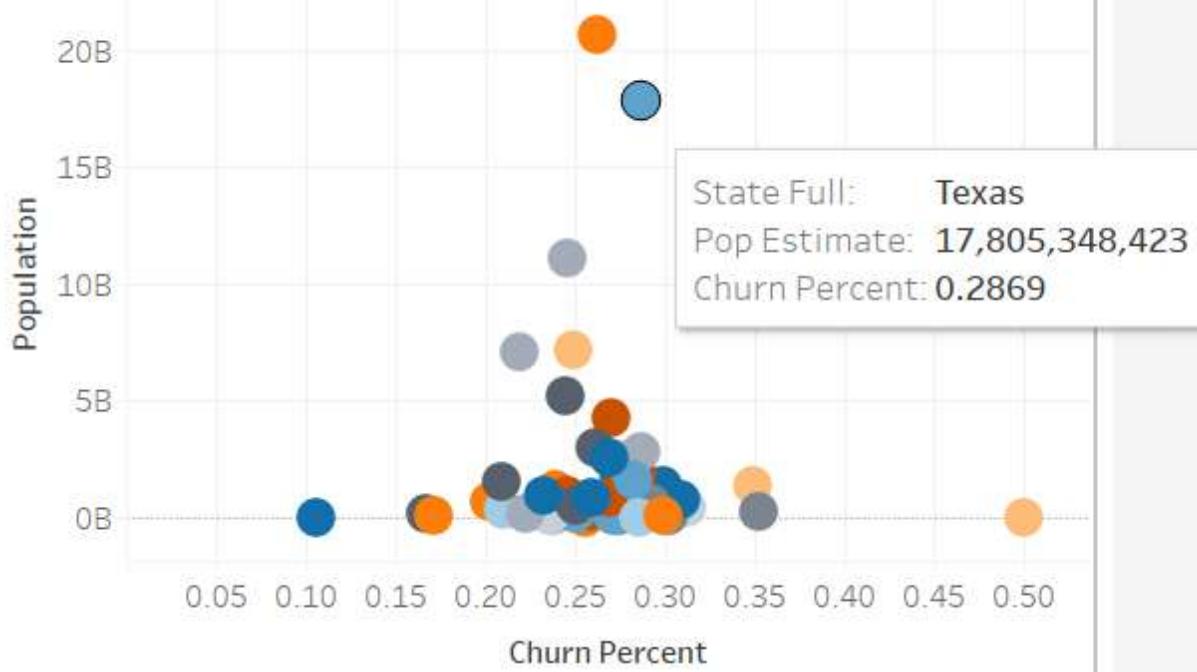
State

(All)



All sheets have elements that will reveal more detailed data (churn percent, population estimate, etc.) when the mouse is hovered over.

## States: Churn % With Total Population



## Web Sources

<https://www.census.gov/data/datasets/time-series/demo/popest/2020s-state-total.html>

<https://blog.openbridge.com/7-steps-to-export-sql-statements-from-tableau-7e51a2fd4277>

<https://stackoverflow.com/questions/6308594/how-can-i-copy-data-from-one-column-to-another-in-the-same-table>