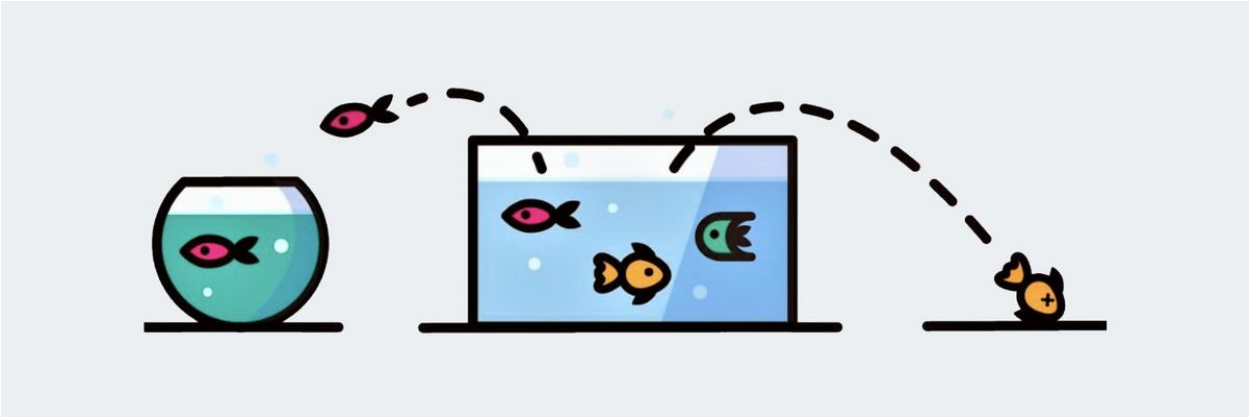
Interactive Dashboard

Chatterbox Telco Pvt Ltd



CS2500 Data Science and Engineering Challenge

Name : Laksika Tharmalingam
Index NO : 190346A
Date : 02.04.2022

Problem definition

Customer churn is the loss of customers by a business for different reasons such as poor service and better prices somewhere else. It is one of the most critical and challenging problems for telecommunication companies, credit card companies, cable service providers, etc.

CEO of Chatterbox Telecom Pvt Ltd in the Banana Republic wants to analyze this customer churn in his/her company.

Motivation & Objectives

Finding solutions to reduce customer churn is important since it is more expensive to acquire new customers than it is to keep the ones you already have. Therefore our aim is

- Analyze data to uncover new insights.
- Develop a dashboard that presents numerous insights into the provided dataset and enables the user to receive a prediction for a new customer.

Solution

Possible Technologies

1. PowerBI

   Power BI is a Microsoft-developed interactive data visualization software application with a primary focus on business intelligence.

   Pros: Outstanding graphics, excellent data connectivity.

   Cons: The interface is not intuitive, and the formulas are inflexible. It is well-suited for a variety of analytics activities.

2. Chart.js

   Chart.js is a lightweight but highly configurable JavaScript charting framework. It's free to use, includes a good selection of chart kinds (eight in all), and supports animation and interaction. Chart.js outputs using HTML5 Canvas, which ensures that charts look great in all modern browsers. Additionally, the charts created are responsive, which is ideal for developing mobile-friendly visualizations.

Pros: Responsive and cross-browser compatible output that is open source.

Cons: In comparison to other tools, this one has a very restricted number of chart kinds. It also has limited support outside of the official documentation.

Chart.js is an excellent choice for designers looking for a straightforward, customizable, and interactive visualization tool.

3. D3.js

D3.js is a JavaScript library for data-driven document manipulation. D3.js requires at least some knowledge of JS, while there are applications available that enable non-programmers to use the framework. Among these apps are NVD3, which provides reusable charts for D3.js; Plotly's Chart Studio, which enables designers to create WebGL and other charts; and Ember Charts, which is likewise built on the Ember.js platform.

Pros: Extremely powerful and adaptable, Numerous chart types are possible. A strong emphasis on web standards Tools for non-programmers to make visualizations It is completely free and open source.

Cons: To be used independently, requires programming knowledge. Support is less available than it is with commercial tools.

D3.js is only ideal for designers who have access to a programmer for assistance or who possess programming experience.

4. Grafana

Grafana is an open-source visualization tool for creating dynamic dashboards. It can handle mixed data sources, annotations, and configurable alerts. Designers can export dashboard snapshots and invite other people to collaborate. Grafana has over 50 plugin data sources. It's free to download or $49/month for cloud hosting. (A free hosted version is available.) Unlike many other open-source programs, the downloadable version includes support plans.

Pros: Open source, free and premium versions, Various data sources available, Chart kinds available, Easy to create dynamic dashboards Use mixed data feeds.

Cons: Excessive for simple visualizations Setup difficult, Not as customizable as other tools, Not the best tool for visualizing data, Individual panels can be embedded in websites, but not dashboards.

Grafana is a great tool for developing internal dashboards with mixed or huge data sources.

5. Dash

Dash is the first low-code framework for rapidly constructing data apps in Python, R, Julia, and F#. It is built on Flask, Plotly.js, and React.js.

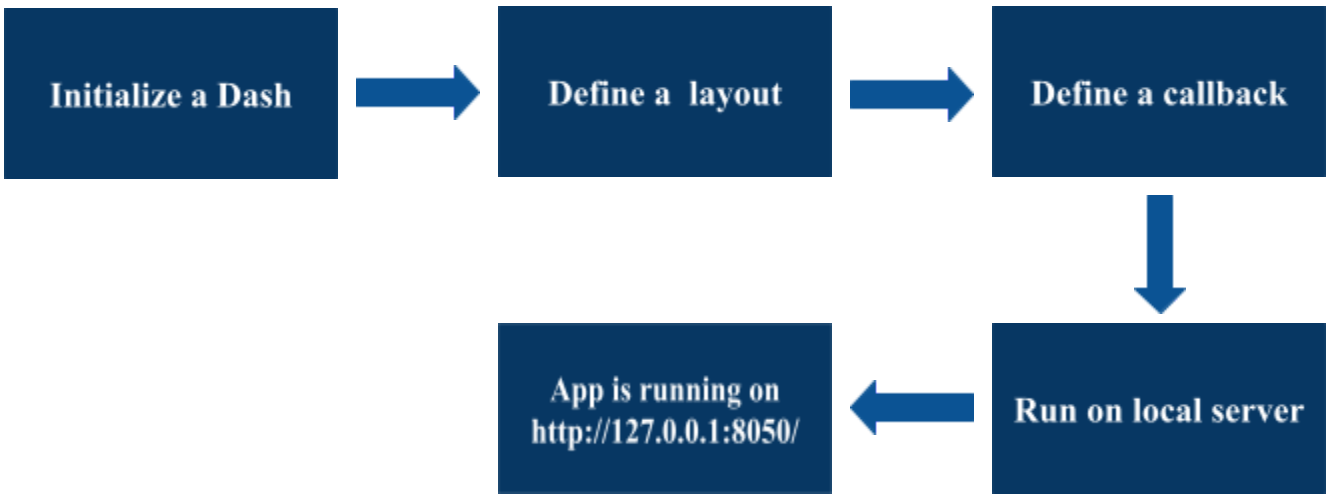
Pros: We may create from scratch. It is more customisable and makes development easier.

Cons: There are no lists in the Dash HTML, customizing the API is tough, and two Python callbacks cannot update the same element.

Selected Technology

Since it is sufficient to be familiar with Python and a little web programming to build an interactive dashboard using Dash, Dash will be a better fit than other related technologies that are mentioned above, and it simplifies the process of plotting graphs and visualizing data. We can use data frames directly to feed them into Dash's graph elements.

Procedure to implement a dashboard with Dash:



Some features of the dashboard

1. Explore the numerous insights.
2. Dashboard chart filter.
3. Time Interval Widget.
4. Dynamic Text Boxes.
5. Predict for a new customer.

Project timeline

Task	Period
Data pre-processing	01.04.2022 - 15.04.2022
Data analyzing	16.04.2022 - 30.04.2022
Model building	01.05.2022 - 15.05.2022
Dashboard building	16.05.2022 - 30.05.2022
Merging & Testing	31.05.2022 - 05.06.2022