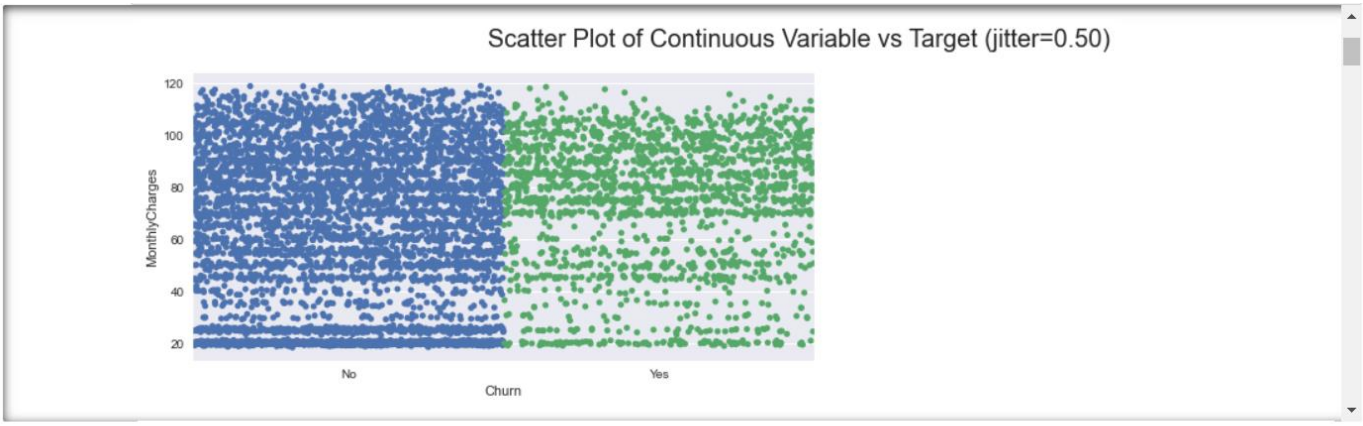
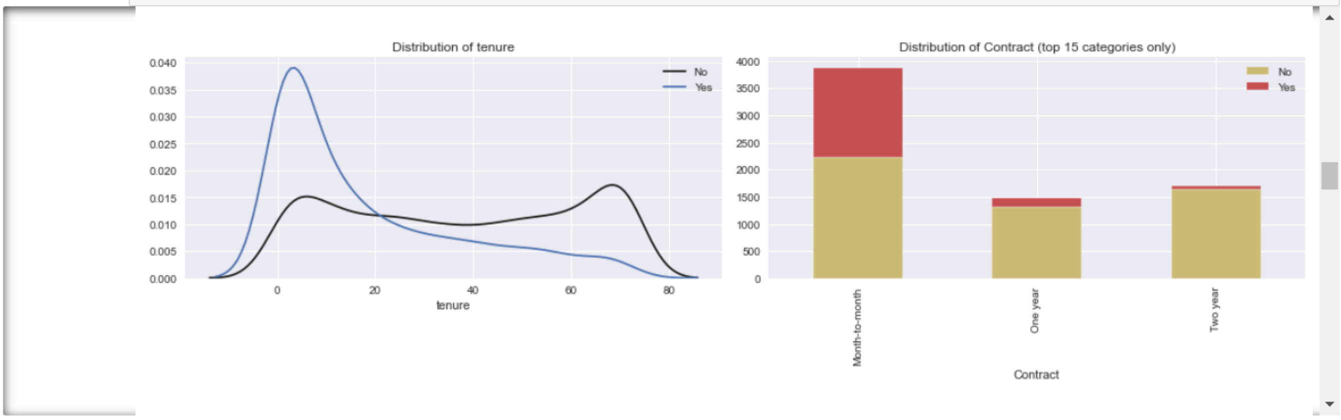
# Project Name: Visualization With Autoviz in 1 line - 4

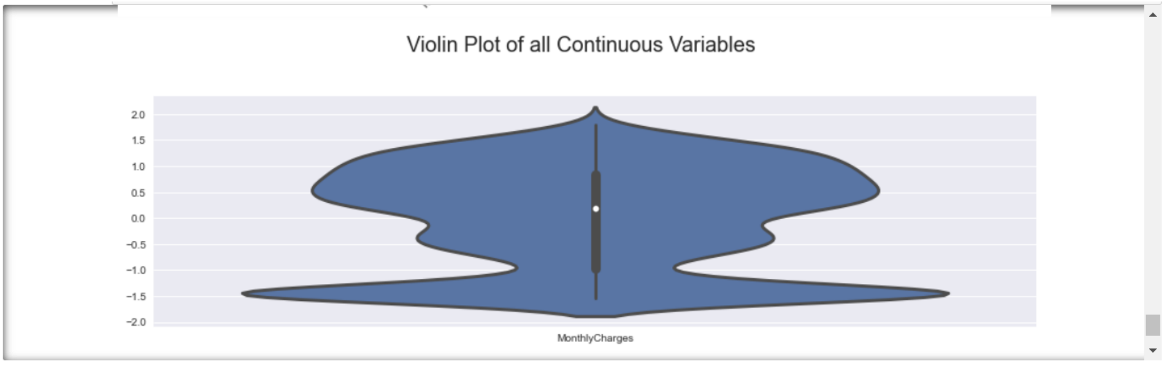
## Table of Contents

Demo
Overview
Motivation
Technical Aspect
Installation
Run/How to Use/Steps
Directory Tree/Structure of Project
To Do/Future Scope
Technologies Used/System Requirements/Tech Stack
Credits

## Demo







## Overview

This is an Exploratory Data Analysis with Visualization using Titanic Dataset.

EDA is essentially a type of storytelling for statisticians.

It allows us to uncover patterns and insights, often with visual methods, within data.

EDA is often the first step of the data modelling process.

This repository contains the code for EDA along with Visualization using python's various libraries.

It used numpy, pandas, xgboost, os, sys, urillib and autoviz libraries.

These libraries help to perform individually one particular functionality.

Numpy is used for working with arrays. It stands for Numerical Python.

Pandas objects rely heavily on Numpy objects.

os means miscellaneous operating system interfaces.

sys means system-specific parameters and functions.

urllib is a package that collects several modules for working with URLs.

Autoviz means Automatically Visualize any dataset, any size with a single line of code.

The purpose of creating this repository is to gain insights into EDA with Visualization.

These python libraries raised knowledge in discovering these libraries with practical use of it.

It leads to growth in my ML repository.

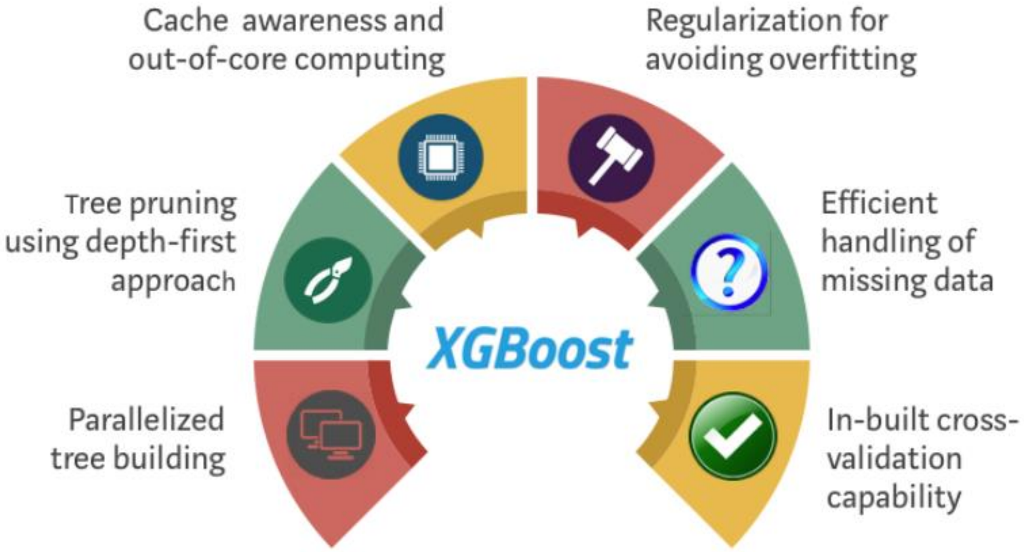
These above screenshots will help you to understand flow of output.

## Motivation

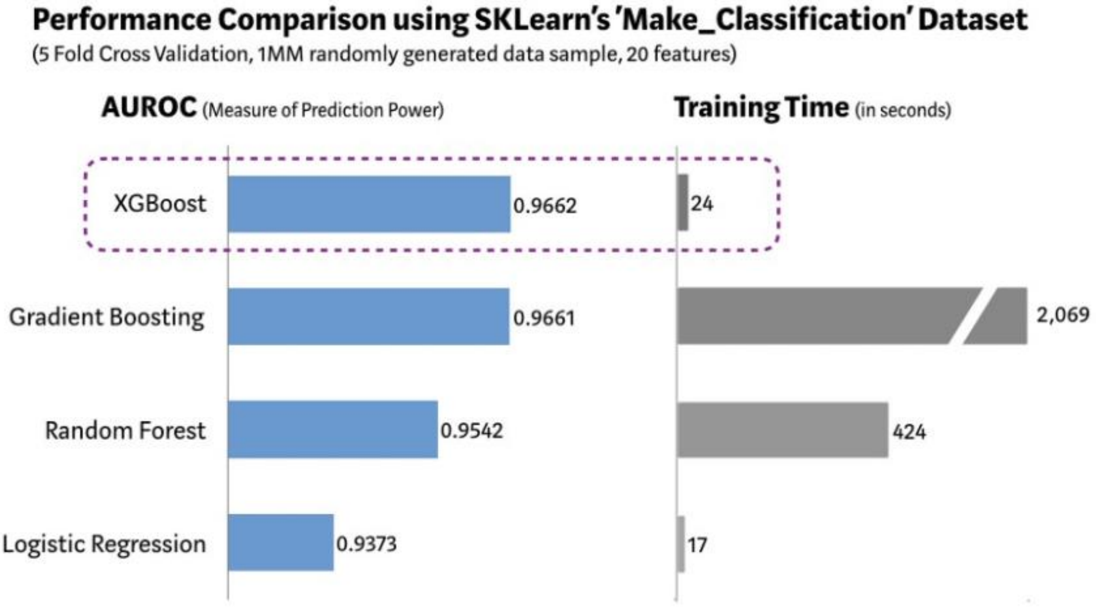
The reason behind building this is, to maximize I as analyst's insight into a data set and into the underlying structure of a data set, while providing all of the specific items that an analyst would want to extract. It is a way of visualizing, summarizing and interpreting the information that is hidden in rows and column format. EDA is understanding the data sets by summarizing their main characteristics often plotting them visually. This step is very important for me especially when I arrive at modelling the data in order to apply Machine learning. Another motive is, as a master's student I have learnt Data Mining Subject which has somewhere led me to also learn about EDA and Visualization. Data visualization is the discipline of trying to understand data by placing it in a visual context so that patterns, trends and correlations that might not otherwise be detected can be exposed. Visualization through visual imagery has been an effective way to communicate both abstract and concrete ideas since the dawn of humanity. A good visualization tells a story, removing the noise from data and highlighting the useful information. Effective data visualization is a delicate balancing act between form and function. Even statistically, it is said that child from 0-5 years of age can remember 92% of things that have seen in form of image as cartoons rather than only read as text such as dialogues of cartoon characters. For example, I do not remember all dialogues of Tom-Jerry Cartoon but I definitely remember how they look and that is because I saw their visual picture. I obtained grasp on this abstraction. It is a reward for me after practicing it manually. I will always recommend to practice manually first then once we get hang on it then can try with other libraries. I am trying many distinct techniques for training my visualization concept clarity because I also prefer viewing and retaining information rather than reading therefore this will also help while presenting in front of company's client and they will have finer perception of it. Hence, I continue to gain knowledge while practicing the same and spread intellectual wings in tech-heaven.

## Technical Aspect

XGBoost is an implementation of gradient boosted decision trees designed for speed and performance. It is recently dominating applied ML industry. XGBoost uses more regularized model formalization to control over-fitting which gives it better performance than GBM. XGBoost is like 'steroids for developers. It is so good because It is a perfect combination of software and hardware optimization techniques to yield superior results using fewer computing resources in the shortest amount of time.


Cache awareness and out-of-core computing
Regularization for avoiding overfitting
Tree pruning using depth-first approach
Efficient handling of missing data
XGBoost
Parallelized tree building
In-built cross-validation capability

How XGBoost optimizes standard GBM algorithm



XGBoost vs. Other ML Algorithms using SKLearn's Make_Classification Dataset

Reason for selecting xgboost here is, when it comes to small-to-medium structured/tabular data, decision tree-based algorithms that is xgboost are considered best-in-class right now. And because the number of features < number of training samples. Or When You have a mixture of categorical and numeric features.

Numpy contains a multi-dimensional array and matrix data structures. It works with the numerical data. Numpy is faster because is densely packed in memory due to its homogeneous type. It also frees the memory faster.

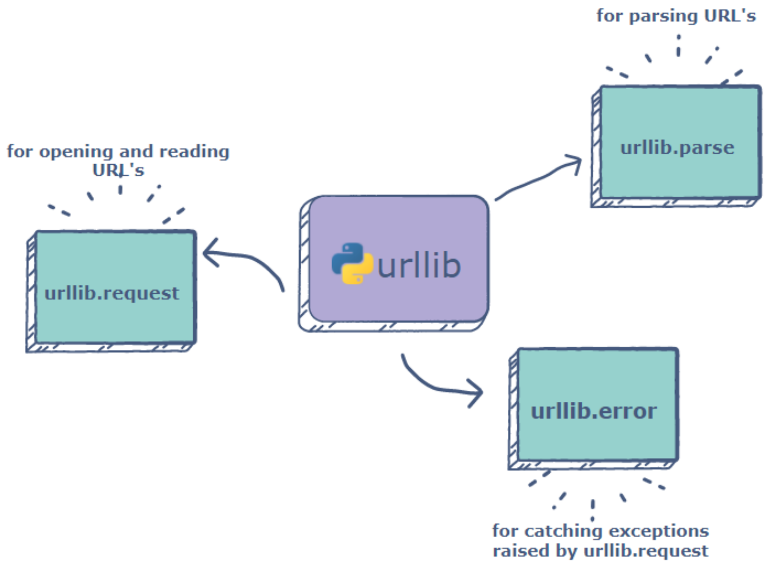
Pandas module mainly works with the tabular data. It contains Data Frame and Series. Pandas is 18 to 20 times slower than Numpy.  Pandas is seriously a game changer when it comes to cleaning, transforming, manipulating and analyzing data.

os provides a portable way of using operating system dependent functionality. It makes it possible to automatically perform many operating system tasks. It provides functions for creating and removing a folder, fetching its contents, changing and identifying the current folder etc.

Autoviz is a one-click visualization engine. It creates powerful charts that anyone from beginner to to an expert can use. It can create charts from any flat file format be it csv or excel or txt. Just upload your data and Autoviz will send you the right charts that help you derive insights within minutes. It mainly works on visualizing the relationship of the data, it can find the most impactful features and plot creative visualization in just one line of code. Autoviz is incredibly fast and highly useful.

sys module provides access to some variables used by interpreter and to functions that interact strongly with the interpreter. Import sys loads the module named sys into the current namespace so that you can access the functions and anything else defined within the module using the module name. On the most common items is the list of arguments created when the program was called.  sys module provides information about constants, functions and methods of the python interpreter.

Urllib uses the urlopen function and is able to fetch URLs using a variety of different protocols. Through urllib, you can access websites, download data, parse data, modify your headers, and do any GET and POST requests you might need to do.


for parsing URL's
urllib.parse
for opening and reading URL's
urllib.request
urllib
urllib.error
for catching exceptions raised by urllib.request

## Installation

Using intel core i5 9th generation with NVIDIA GFORECE GTX1650.
Windows 10 Environment Used.
Already Installed Anaconda Navigator for Python 3.x
The Code is written in Python 3.8.
If you don't have Python installed then please install Anaconda Navigator from its official site.
If you are using a lower version of Python you can upgrade using the pip package, ensuring you have the latest version of pip, *python* -m pip install --*upgrade pip and press Enter*.

## Run/How to Use/Steps

Keep your internet connection on while running or accessing files and throughout too.
Follow this when you want to perform from scratch.

Open Anaconda Prompt, Perform the following steps:

```
cd <PATH>
pip install numpy
pip install pandas
pip install autoviz
pip install xgboost
```

You can also create requirement.txt file as, pip freeze > requirements.txt

run files.

Follow this when you want to just perform on local machine.
Download ZIP File.
Right-Click on ZIP file in download section and select Extract file option, which will unzip file.
Move unzip folder to desired folder/location be it D drive or desktop etc.
Open Anaconda Prompt, write cd <PATH> and press Enter.
eg: cd C:\Users\Monica\Desktop\Projects\Python Projects 1\12)Visualization\Project_1_Visualization_with_autoviz_in_1line

In Anconda Prompt, pip install -r requirements.txt to install all packages.
Open in Jupyter Notebook, <filename>.ipynb
That is,

Open in Jupyter Notebook, 1)Visualization_with_Autoviz.ipynb
This takes churn_data.csv file as input dataset.
Please be careful with spellings or numbers while typing filename and easier is just copy filename and then run it to avoid any silly errors.
Note: cd <PATH>

[Go to Folder where file is. Select the path from top and right-click and select copy option and paste it next to cd one space <path> and press enter, then you can access all files of that folder]   [cd means change directory]

## Directory Tree/Structure of Project

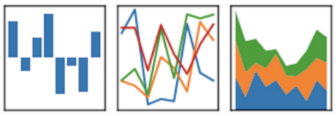
Folder: 12)Visualization > Project_1_Visualization_with_autoviz_in_1line

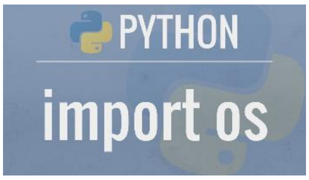
1)Visualization_with_Autoviz.ipynb

## To Do/Future Scope

Can make it End to end project.

## Technologies Used/System Requirements/Tech Stack





## Credits

https://analyticsindiamag.com/tips-for-automating-eda-using-pandas-profiling-sweetviz-and-autoviz-in-python/

AIEngineering Channel