IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

ISO 3297:2007 Certified ⁝⁝ Impact Factor 8.066 ⁝⁝ Peer-reviewed / Refereed journal ⁝⁝ Vol. 10, Issue 6, June 2023

DOI: 10.17148/IARJSET.2023.10631

# Time Series Analysis: Financial Engineering

**Charan Pote[1], Rahul Gaherwar[2], Kartik Shrivas[3], Priya Tripathi[4], Suraj Bhanarkar[5]**

Guide, Priyadarshini College of Engineering, Nagpur[1]

Priyadarshini College of Engineering, Nagpur[2-5]

**Abstract:** Time-series analysis is a statistical method of analyzing data from repeated observations on a single unit or individual at regular intervals over a large number of observations. Time-series analysis can be viewed as the exemplar of longitudinal designs. The most widely employed approach is based on the class of models known as Autoregressive Integrated Moving Average (ARIMA) models. ARIMA models can address several major classesof research questions, including an analysis of basic processes, intervention analysis, and analysis of the pattern of treatment effects over time. Technical aspects of ARIMA models are described, including definitions of important terms, statistical estimation of parameters, and the model identification process. Examples are employed to clarify the technical discussion. Recent extensions of ARIMA modeling techniques include multiunit time-series designs, multivariate time-series analysis, the inclusion of covariates, and the analysis of patterns of intra-individual differences across time. Keywords: ARIMA models; autocorrelation; longitudinal designs; single-subject research; time-series analysis; within-subject research

**Keywords:** Time Series, Frequency, Recency, Monetary Value, Trends, Seasonality, Anomaly Detection,Annual Recurrent Revenue, Lifetimes, Prophet, Streamlit, Product-Based Prediction

## INTRODUCTION

Companies across every industry are generating massive amounts of data covering all sorts of metrics. By analyzing the data and finding patterns, these organizations can discover interesting insights about their business. Much of the focus has been on using data insights to improve day to day operations and cybersecurity. Now companies are turning their attention to utilizing business metrics to support better business decision making.

This Project is in session with the purpose of creating a commercial system for real-time analytics and automated outlier detection. To that we shall add the means to do autonomous forecasting to predict businessgrowth and demand. It is AI-powered forecasting in a turn-key experience, meaning the solution can be used without needing to have a data scientist to forecast the time series metrics. Our Autonomous Forecast automatically manages the machine learning required to create, train, tune and deploy a customized forecasting model.

## LITERATURE SURVEY

**Subscription Economy IndexFighting Churn with Data Revenue Forecast Accuracy Journey to Usership**
**The Essential Guide to Time Series ForecastingAnomaly Detection System**

## APPLICATIONS

### Trend Estimation
Time series methods can be conducted to discover trends, for example, these methods inspect data observations to identify when measurements reflect a decrease or increase in sales of a particular product.

### Seasonal Patterns
Recorded data points variances could unveil seasonal patterns & fluctuations that act as a base for data forecasting. The obtained information is significant for markets whose products fluctuateseasonally and assist organizations in planning product development and delivery requirements.

### Financial Decision Making
The accuracy of prediction of business failure is a very crucial issue in financial decision-making. Therefore, different ensemble classifiers are proposed to predict financial crises and financial distress.Because ensemble learning improves the robustness of the normal behavior modelling, ithas been proposed as an efficient technique to detect customer fraudulent cases and activities

DOI: 10.17148/IARJSET.2023.10631

**Encompassing Multitude of Platforms (Human Resource)**

The current project encompasses multitude of platforms spanning all three phases covering scope of plethora of techniques resulting in significant improvement of learning curve as machine learning engineer aspirant. These platforms has their own discrete applications, hence preparing an individual for various prolific roles in engineering, expanding the knowledge basis and chances of getting recruited on industry's rarest projects.

**Implement Trustworthy Artificial Intelligence**

Address explainability, fairness, robustness, transparency and privacy as part of the AI lifecycle. Mitigate model drift, bias and risk in AI and machine learning. Validate and monitor models toverify that AI and machine learning performance meets business goals. Help meet corporate social responsibility (CSR) and environmental social governance (ESG).

## BENEFITS TO THE SOCIETY

Satisfied customers (associated persons) are the best advocates. Their positive word-of-mouth lends brand credibility, and popularity helps acquire new customers. The mentioned aspects save brands money that they would spend on marketing and promotional campaigns.

Brand credibility: building up well-aware and connected communities in society.

Popularity: connected web of persons sharing common values and experiences of improved lifestyle through redefined precisional services.

Customer acquisition: building a dense web of interconnectedness.Investment Savings:
Can be redirected to bonus and additional income provision for employees.Can be used to donate to needful funding societies for brand promotion.

Can be used to create needful strengthening of the existing process.To channelise new innovative process flows.
A portion as give-back to customers under a reward scheme .

## ADVANTAGES

Capture more revenue with automated operations that **scale across the business**.

Address the obstacles and opportunities in the **quote-to-cash** process with the industry’s most comprehensive set of **low to no-code tools** to maximize our monetization strategy.

**Resolved variances** in revenue recognition proactively with accounting analysis.Quickly forecast and plan revenue directly from open pipeline.

**Identify revenue gaps** and improve future revenue forecasts by relating actual financial performance.

A **single code base**, enabling every customer to run their business on the latest release without disruption.

**Provides effective methods** For identifying churn’s underlying factors and prescriptive tools for addressing it. Increase in sales through improved accept rates.

**Reveal behavioural patterns** of customers who have already stopped using the services or buying productsagainst existing customers. Contributes to early exposure management.

**Building prediction models** building a model to predict how likely a customer will churn by analysing characteristics. Reduction in processing time through automated decisions.

## LIMITATIONS

**Sparse Representation of Dense Data**

Despite choosing an appropriate sample of the population data, the representation always remains limited to organisational scope as data of overall population is discretised with rival companies, shielded on their autonomised encrypted servers.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

ISO 3297:2007 Certified ⋇ Impact Factor 8.066 ⋇ Peer-reviewed / Refereed journal ⋇ Vol. 10, Issue 6, June 2023

DOI: 10.17148/IARJSET.2023.10631

**Automated Benchmark Outcompetes Manual Data Models**
Benchmarking results in accessing valuable automated services of third party giant vendors, putting the parent organisational data susceptible to inter server security breach as well as contract crossover

**Requirement of Holistic Cross Venture Dataset Amalgamation And Analysis**
Summed up as Alternate Scoring in general has become a norm so as to take into account isolated feautures of datasets like roaming, geo location, social network, app usage and top-ups, calling behaviour etc for contributing to the customer churn scores, i.e requires a broad and wide mega project scope and industry body for amalgamation of datasets on contract basis.

**Volatile Markets**
Some markets are highly volatile which can be hard to forecast even when following a data-driven approach

## IMPLEMENTATION

**BASIC FREQUENCY/RECENCY ANALYSIS USING THE BG/NBD MODEL**
For small samples sizes, the parameters can get implausibly large, so by adding an l2 penalty the likelihood, we can control how large these parameters can be. This is implemented as setting as positive penalizer_coef in the initialization of the model. In typical applications, penalizers on the order of 0.001 to 0.1 are effective.

**Visualizing Frequency/Recency Matrix**
Consider: a customer bought from us every day for three weeks straight, and we haven't heard from them in months. What are the chances they are still "alive"? Pretty small. On the other hand, a customer who historically buys from us once a quarter, and bought last quarter, is likely still alive.

We can visualize this relationship using the **Frequency/Recency matrix**, which computes the expected number of transactions an artificial customer is to make in the next time period, given his or her recency (age at last purchase) and frequency (the number of repeat transactions he or she has made).

**Ranking customers from best to worst**
Rank customers from "highest expected purchases in the next period" to lowest. Models expose a method that will predict a customer's expected purchases in the next period using their history.

**Customer Probability Histories**
Given a customer transaction history, we can calculate their historical probability of being alive, according to our trained model.

**Estimating customer lifetime value using the Gamma-Gamma model**
Taking into account the economic value of each transaction, we focus mainly on transactions' occurrences. To estimate this we can use the Gamma-Gamma submodel. But first we need to create summary data from transactional data also containing economic values for each transaction (i.e. profits or revenues).

**Estimating customer lifetime value using the Gamma-Gamma model**
For this whole time we didn't take into account the economic value of each transaction and we focused mainly on transactions' occurrences.

To estimate this we can use the Gamma-Gamma submodel. But first we need to create summary data from transactional data also containing economic values for each transaction (i.e. profits or revenues)

**Workflow**
The usual workflow of using the Lifetimes library can be represented through the following fluxogram :

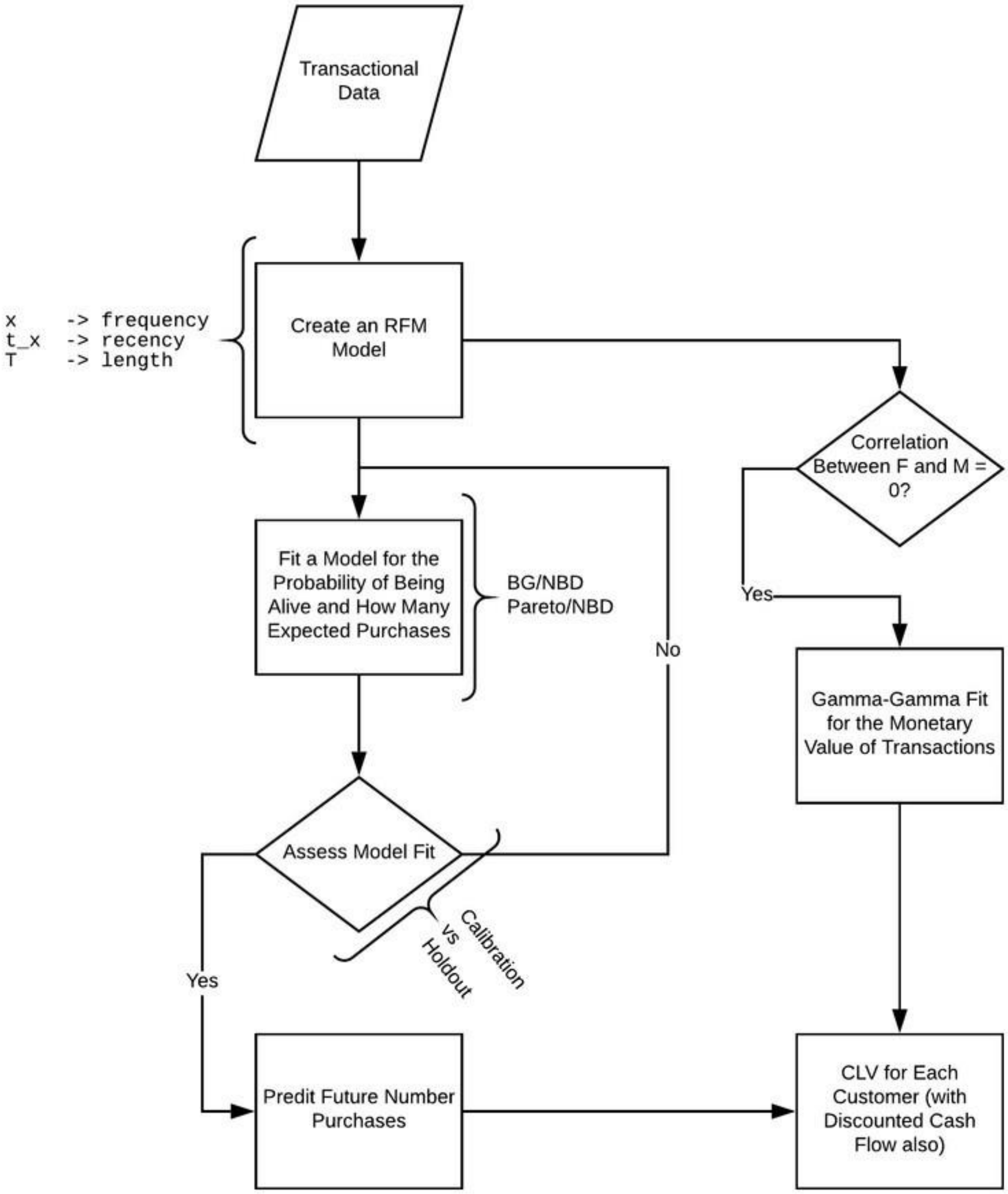
  

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

ISO 3297:2007 Certified ⁜ Impact Factor 8.066 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 10, Issue 6, June 2023

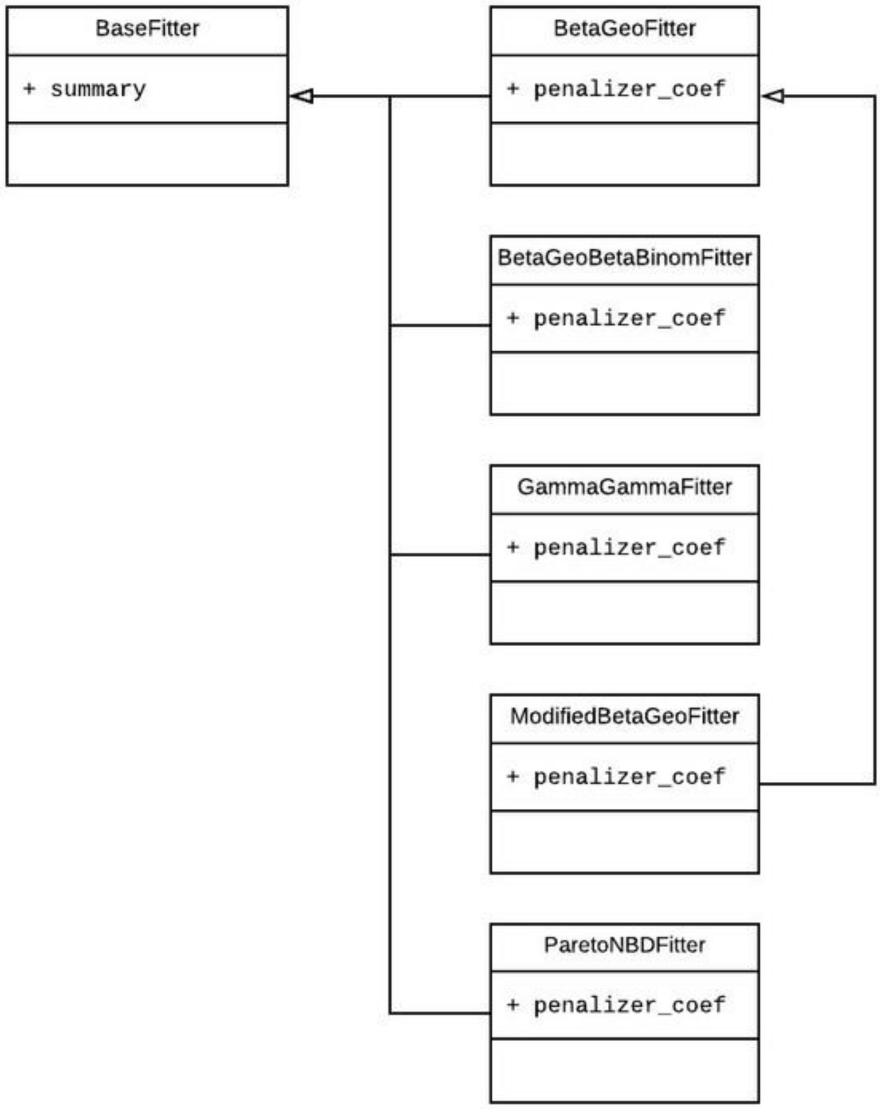
DOI: 10.17148/IARJSET.2023.10631




**Fitters**

The core fitter is the BaseFitter class is inside the __init__.py, which serves as a superclass for most of the the other fitters. So far, only the ModifiedBetaGeoFitter is set on a higher layer, inheriting from the BetaGeoFitter. The following image shows the simplified interaction of the main fitter classes.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
ISO 3297:2007 Certified ⁙ Impact Factor 8.066 ⁙ Peer-reviewed / Refereed journal ⁙ Vol. 10, Issue 6, June 2023
DOI: 10.17148/IARJSET.2023.10631




**GRAPHS**

Graphs are plotted with functions coming from the plotting.py file. The main functions are cited below, alongside a brief description of how they are created.

plotting.py

```
coalesce(*args)
plot_period_transactions(model, max_frequency, title, xlabel,  ylabel, **kwargs)
plot_calibration_purchases_vs_holdout_purchases(model, calibration_holdout_matrix, kind, n, **kwargs)
plot_frequency_recency_matrix(model, T, max_frequency, max_recency, title, xlabel, ylabel, **kwargs)
plot_probability_alive_matrix(model, max_frequency, max_recency, title, xlabel, ylabel, **kwargs)
plot_expected_repeat_purchases(model, title, xlabel, ax, label, **kwargs)
plot_history_alive(model, t, transactions, datetime_col, freq, start_date, ax, **kwargs)
plot_cumulative_transactions(model, transactions, datetime_col, customer_id_col, t, t_cal, datetime_format, freq, set_index_date, title,  xlabel, ylabel, ax, **kwargs)
plot_incremental_transactions(model, transactions, datetime_col, customer_id_col, t, t_cal, datetime_format, freq, set_index_date, title, xlabel, ylabel, ax, **kwargs)
plot_transaction_rate_heterogeneity(model, suptitle, xlabel, ylabel, suptitle_fontsize, **kwargs)
plot_dropout_rate_heterogeneity(model, suptitle, xlabel, ylabel, suptitle_fontsize, **kwargs)
forceAspect(ax, aspect)
```

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

ISO 3297:2007 Certified ⁑ Impact Factor 8.066 ⁑ Peer-reviewed / Refereed journal ⁑ Vol. 10, Issue 6, June 2023

DOI: 10.17148/IARJSET.2023.10631

**PROPHET: (HANDLING SHORTCOMINGS OF LIFETIMES)**

Prophet is a procedure for forecasting time series data based on an additive model where non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects. It works best with time series that have strong seasonal effects and several seasons of historical data. Prophet is robust to missing data and shifts in the trend, and typically handles outliers well.

Prophet is open source software released by Facebook's Core Data Science team. It is available for download on CRAN and PyPI.

**Accurate and fast.**

Prophet is used in many applications across Facebook for producing reliable forecasts for planning and goal setting. We've found it to perform better than any other approach in the majority of cases. It fit models in Stanso that we get forecasts in just a few seconds.

**Fully automatic.**

Get a reasonable forecast on messy data with no manual effort. Prophet is robust to outliers, missing data, and dramatic changes in our time series.

**Tunable forecasts.**

The Prophet procedure includes many possibilities for users to tweak and adjust forecasts. We can use human-interpretable parameters to improve our forecast by adding our domain knowledge.

Forecasting is a common data science task that helps organizations with capacity planning, goal setting, and anomaly detection. Despite its importance, there are serious challenges associated with producing reliable and high quality forecasts – especially when there are a variety of time series and analysts with expertise in timeseries modeling are relatively rare. To address these challenges, prophet describe a practical approach to forecasting "at scale" that combines configurable models with analyst-in-the-loop performance analysis. It propose a modular regression model with interpretable parameters that can be intuitively adjusted by analystswith domain knowledge about the time series. It describe performance analyses to compare and evaluate forecasting procedures, and automatically flag forecasts for manual review and adjustment. Tools that help analysts to use their expertise most effectively enable reliable, practical forecasting of business time series.

**STREAMLIT: (FULL STACK)**

Streamlit is a framework that turns Python scripts into interactive apps, giving data scientists the ability to quickly create data and model-based apps for the entire company.

A simple Streamlit app is:

import streamlit as st

number = st.slider("Pick a number: ", min_value=1, max_value=10)st.text("our number is " + str(number))

When we streamlit run my_app.py, we start a web server that runs the interactive application on our local

computer at

http://localhost:8501. This is great for local development. When we want to share with our colleagues, Streamlit Community Cloud enables us to deploy and run these applications in the cloud. Streamlit Community Cloud

handles all

dtheepldoeytaeidlsaopfps. caling, reliability, and security as well as providing us an interface for easily managing our

**Running Streamlit script**

Working with Streamlit is simple. First we sprinkle a few Streamlit commands into a normal Python script, and then we run it. We list few ways to run our script, depending on our use case.

**Use streamlit run**

Once we've created our script, say our_script.py, the easiest way to run it is with streamlit run:

streamlit run our_script.py As soon as we run the script as shown above, a local Streamlit server will spin up andour app will open in a new tab in our default web browser.

**MACHINE LEARNING WEB-SERVICE**

With the Machine Learning Web service, an external application communicates with a Machine Learning workflow scoring model in real time. A Machine Learning Web service call returns prediction results to an external application. To make a Machine Learning Web service call, we pass an API key that is created when wedeploy a

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

ISO 3297:2007 Certified ⁜ Impact Factor 8.066 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 10, Issue 6, June 2023

DOI: 10.17148/IARJSET.2023.10631

prediction. The Machine Learning Web service is based on REST, a popular architecture choice for web programming projects.

**TIME SERIES ANOMALY DETECTION**

Embed time-series anomaly detection capabilities into app to help users identify problems quickly.

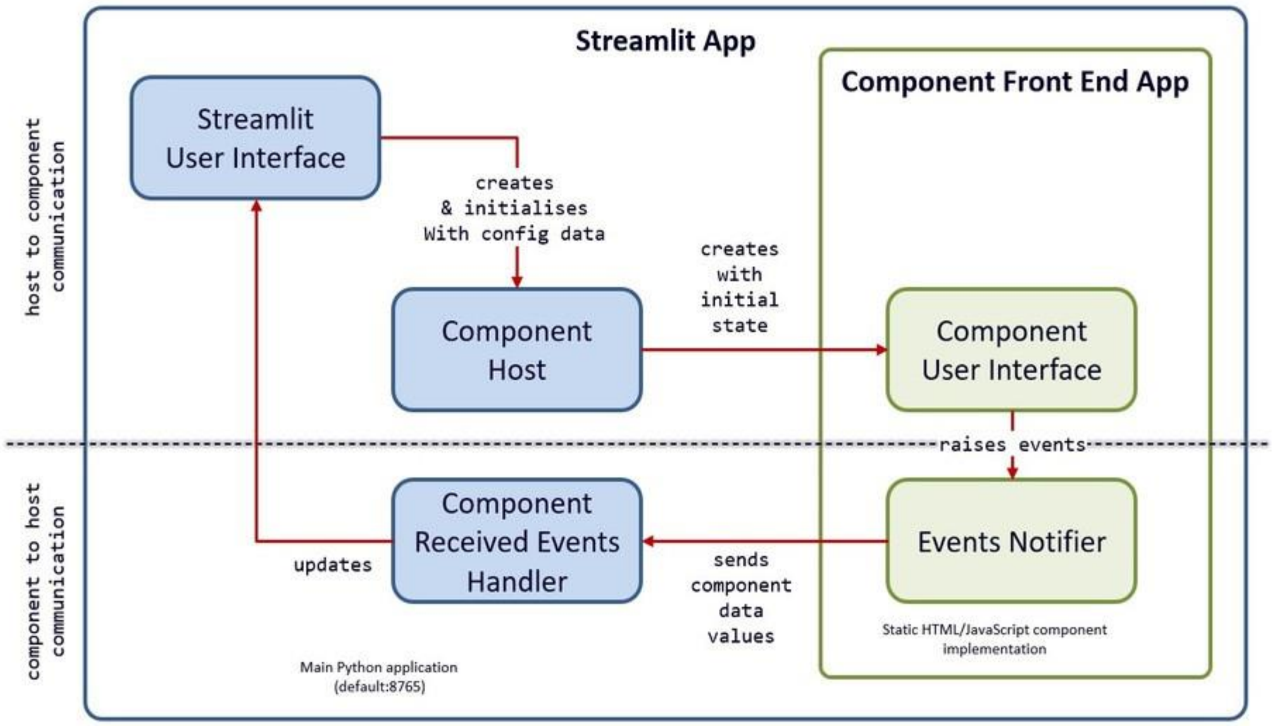
Detector ingests time-series data of all types and selects the best anomaly detection algorithm for data to ensure high accuracy. Detect spikes, dips, deviations from cyclic patterns and trend changes through both univariate and multivariate APIs. Using multivariate anomaly detection to evaluate multiple signals and the correlations betweenthem to find sudden changes in data patterns before they affect business.

**BATCH SCORING**

involves invoking an endpoint with a reference to data. The batch endpoint runs jobs asynchronously to process data in parallel on compute clusters and store the data for further analysis.

**REAL-TIME SCORING**

involves invoking an endpoint with one or more model deployments and receiving a response in near-real-time via HTTPs. Traffic can be split across multiple deployments, allowing for testing new model versions by diverting some amount of traffic initially and increasing once confidence in the new model is established.




**REFERENCES**

[1] (99+) Amanda Iglesias Moreno | LinkedIn AileenNielsen/TimeSeriesAnalysisWithPython (github.com) Practical Time Series Analysis [Book] (oreilly.com)

[2] Google Wikipedia

[3] Advances in Time Series Data Methods in Applied Economic Research: International Conference on Applied Economics (ICOAE) 2018 - PDF Drive

[4] Search Results – Springer

DOI: 10.17148/IARJSET.2023.10631

[5] All: time series : Search (acm.org)

[6] Credit Risk Score - Credit Scoring Model Solutions | Experian Subscription Economy Index Fighting Churn with Data

[7] Revenue Forecast Accuracy

[8] Journey to Usership

[9] The Essential Guide to Time Series Forecasting Anomaly Detection System

[10] THE SUBSCRIBED STRATEGY GROUP

[11] The Key Principles of a Successful Time Series Forecasting System for Business The Variable Time Series

[12] Advancing Computing as a Science & Profession Learning platform your tech teams need to stay ahead Python's reduce(): From Functional to Pythonic Style Lifetimes Package Prophet

[13] Streamlit documentation

**9**