The Data-Powered Leader

Building Intelligent Operations with ML & AI

Site LinkedIn GitHub Kaggle kmcguinness33@gmail.com

Introduction

Who I am & what's this about...

About Me: **I'm Killian, I'm passionate about Data, Customer Operations & LGBTQI+ Representation in Leadership.**

For the past decade, I've built and led operations teams, always looking for ways to use data to make things run more smoothly. This interest in data analysis helped me lead BI teams, focusing on key operational areas like workforce and demand forecasting, crafting insightful KPIs & financial planning.

My love for data goes way back (I studied Advanced Maths in university for a while) and it's driving my current career pivot towards data science and machine learning. After a period of independent consulting, where I helped SMEs build better customer support processes, I'm eager to join a team with a strong mission and a passion for data.

With 12 years of experience building strong KPI frameworks, scaling and mentoring large teams in Europe's Fintech/Foodtech scene, and honing my technical skills in machine learning, generative AI, and data visualization, I'm confident I'd be a great addition to your team.

About This Portfolio: **This document accompanies 3 projects demonstrating my work at the intersection of data science & operations. Find the code links below.**

Project 1

Time Series Forecasting GitHub Kaggle

Find the most accurate model for predicting future call volumes at the San Francisco 999 contact centre.

TimeSeries | sklearn | PyCaert | Prophet

Project 2

Topic Modelling GitHub Kaggle

Turn user generated Reviews & Comments into actionable Insights.

Text Classification | BERT | NLP | XGBoost

Project 3

Data Visualisation GitHub Kaggle

Turn Typical Call centre Data into effective Insights to drive performance.

Data Visualization | Mathplotlib | NumPy | Pandas

Time Series Forecasting

Introduction & Explainer

Project Code

GitHub Kaggle

Objective: Find the most accurate model for predicting future call volumes at the San Francisco 999 contact centre.

This project aims to identify the most effective time series forecasting model for predicting call volume patterns at the San Francisco 999 contact centre.
Accurate forecasting of future call volumes is crucial for any successful customer operations centre. By identifying the optimal forecasting model, we can determine the required number of agents at any given time, enabling efficient resource allocation and optimized service levels.

Key Concepts & Terminology

Before jumping into the project details, let's define some key terminology.

1 A Model

Imagine a model as a student learning from examples. We feed it historical call data, & it analyzes patterns like busy times & seasonal changes. Like the student, it uses this knowledge to predict future call volumes.

2 Time Series Forecast

Time series forecasting is a technique used to predict future values based on past data. Imagine you're trying to predict the weather. You would look at historical weather data, like temperature & rainfall patterns, to make an educated guess about what the weather will be like tomorrow.

3 Residuals, MSE & R^2

Imagine predicting call volumes. The model's predictions won't always match the actual calls. This difference is called a residual. To assess the model's overall performance, we analyze these residuals using Mean Squared Error (MSE) & R-squared (R^2). Lower MSE indicates better predictions whereas higher R^2 is a good sign for the accuracy of the Model.

About the Data

This project utilises call data from the San Francisco 999 contact centre, spanning from January 1st, 2021, to November 28th, 2023. The data mirrors what you would find in any typical contact centre, containing details such as call timestamps and types. This data was sourced from a publicly available repository on the Google Cloud Console here.
At the project's outset, I used Google Cloud BigQuery to retrieve the relevant data and store it within a pandas DataFrame for further analysis.

Site LinkedIn GitHub Kaggle kmcguinness33@gmail.com

Time Series Forecasting

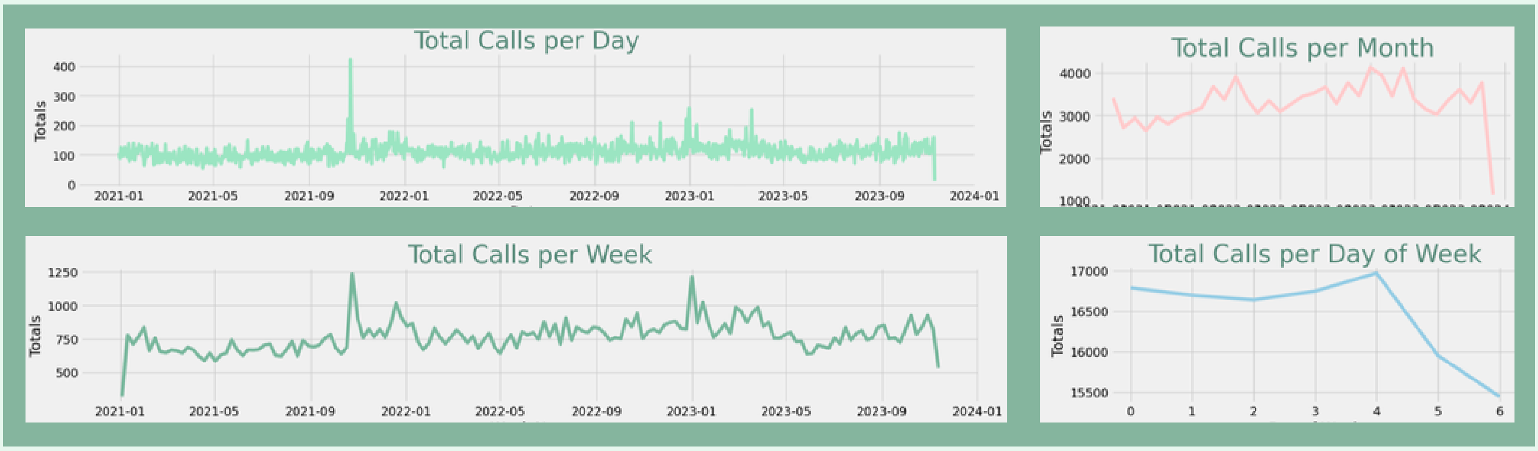
The Process: Smoothing & Preprocessing

Project Code

[GitHub] [Kaggle]

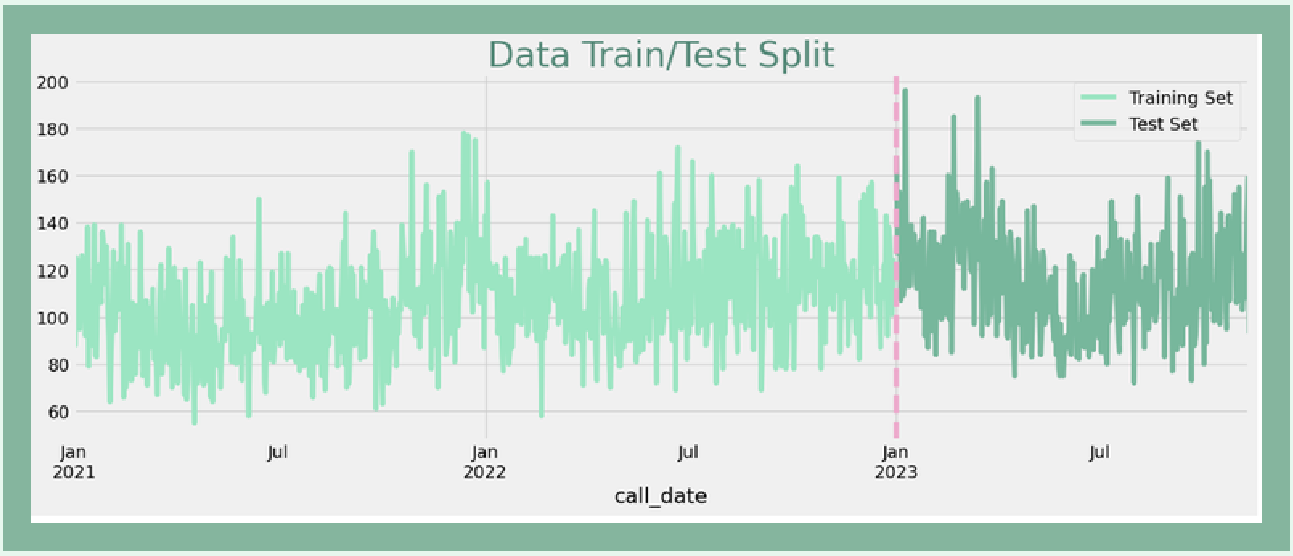
1 Explore the Data Set

I started exploring the dataset by visualizing the call volume trends over time. I created plots for daily, weekly, and monthly views, as well as a specific focus on day-of-week patterns. This helped identify any clear peaks or trends, such as busier days of the week or seasonal fluctuations. The aim here was to spot any clear outliers – unusually high or low call volumes that might require further investigation and potential smoothing to avoid biasing my models. I decided to smooth any daily volumes above 200 or below 50 to avoid overfitting the model.



2 Create a Test Train Split

The call centre data was split into training and testing sets using a time-based approach. This ensures the training set only includes historical data before a specific date (approximately 70% of the data), preventing the model from being influenced by future trends. Data from 2023 onwards will then be used to evaluate the model's performance in predicting call volume.



Site LinkedIn GitHub Kaggle kmcguinness33@gmail.com

Time Series Forecasting

The Process: Training

Project Code

[GitHub] [Kaggle]

3 Train the Models

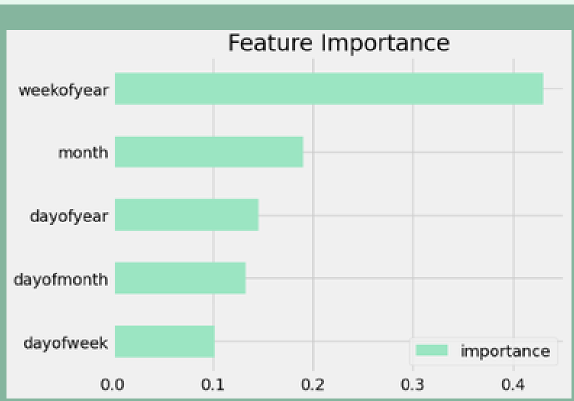
I decided to evaluate four different models. Each model was trained on the same training data and then tested on unseen data held out during the train-test split. The performance of each model was assessed using R-squared (R^2) and Mean Squared Error (MSE) metrics, and we'll review these scores later in the project.

1 Simple Linear Regression using sklearn

This is a basic statistical method that establishes a baseline for comparison with more advanced models.

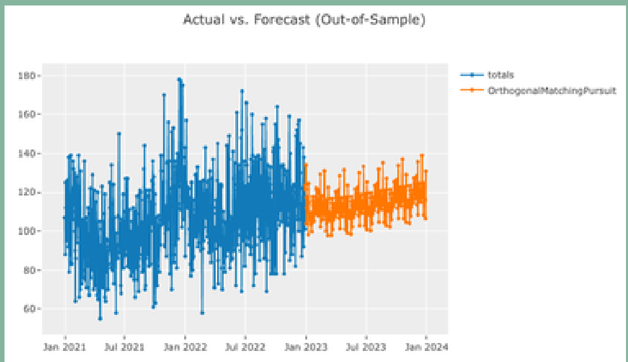
2 XGBoost

This is a more advanced machine learning model that is often used for regression tasks. This model offers a cool extra: feature importance analysis. XGBoost not only helps predict call volumes but also sheds light on the most important factors influencing those predictions.



3 Pycaret

PyCaret is a library that automates training and evaluating various machine learning models. This allowed me to efficiently compare numerous regression techniques and select the most suitable one for predicting call volumes. After analyzing model performance scores and visualizing their predictions, I chose the Orthogonal Matching Pursuit (OMP) model. While not scoring the absolute best on R-squared, it demonstrated strong performance and offered a valuable balance between accuracy and interpretability.



	Model	MASE	RMSSE	MAE	RMSE	MAPE	SMAPE	R2	TT (Sec)
croston	Croston	0.7452	0.6348	16.5166	17.6006	0.1529	0.1431	-1.8361	0.0300
omp_cds_dt	Orthogonal Matching Pursuit w/ Cond. Deseasonalize & Detrending	0.7759	0.7276	17.1943	20.1701	0.1606	0.1471	-2.2895	0.2980
br_cds_dt	Bayesian Ridge w/ Cond. Deseasonalize & Detrending	0.7807	0.7338	17.3020	20.3453	0.1634	0.1476	-2.6326	0.2840
lasso_cds_dt	Lasso w/ Cond. Deseasonalize & Detrending	0.7632	0.7277	16.9150	20.1777	0.1603	0.1444	-2.6687	0.3240
llar_cds_dt	Lasso Least Angular Regressor w/ Cond. Deseasonalize & Detrending	0.7632	0.7277	16.9150	20.1777	0.1603	0.1444	-2.6687	0.2920
ridge_cds_dt	Ridge w/ Cond. Deseasonalize & Detrending	0.7722	0.7306	17.1139	20.2595	0.1619	0.1462	-2.6913	0.2840
lr_cds_dt	Linear w/ Cond. Deseasonalize & Detrending	0.7711	0.7303	17.0909	20.2512	0.1618	0.1460	-2.6915	0.2920
en_cds_dt	Elastic Net w/ Cond. Deseasonalize & Detrending	0.7654	0.7284	16.9639	20.1980	0.1608	0.1449	-2.6924	0.3480

4 Prophet

This is a specialised time series forecasting model that is designed to capture trends & seasonality in data. It was super easy to set up & Run & proved very effective predicting call centre volumes.

Time Series Forecasting

The Results & Conclusion

Project Code

GitHub Kaggle

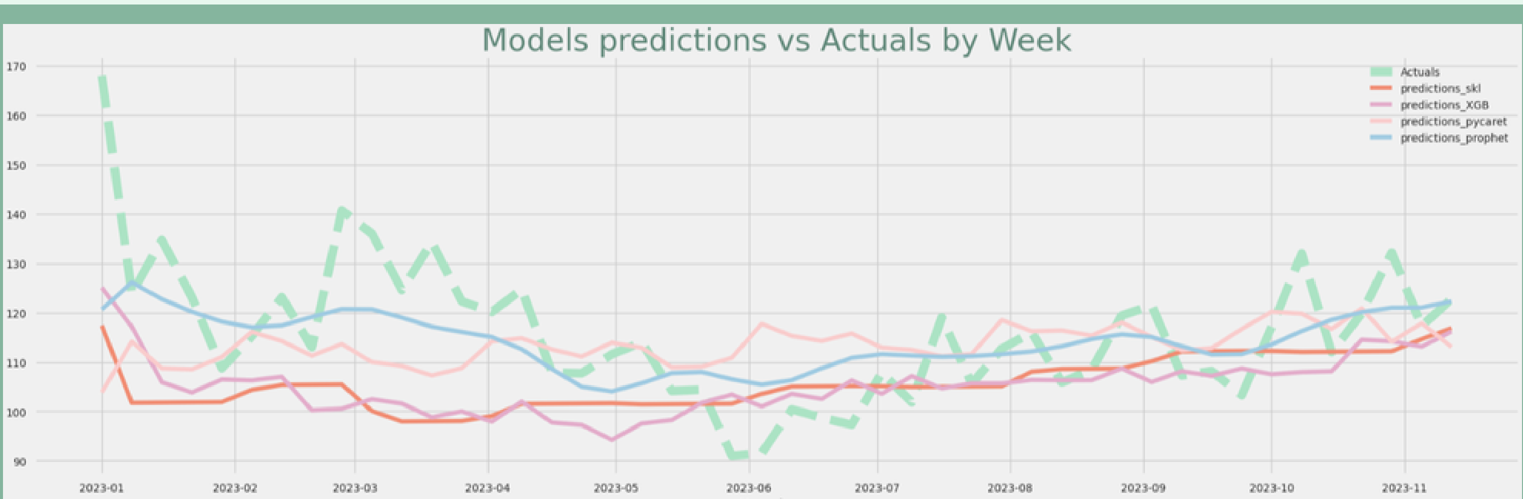
4 Compare MSE & R²

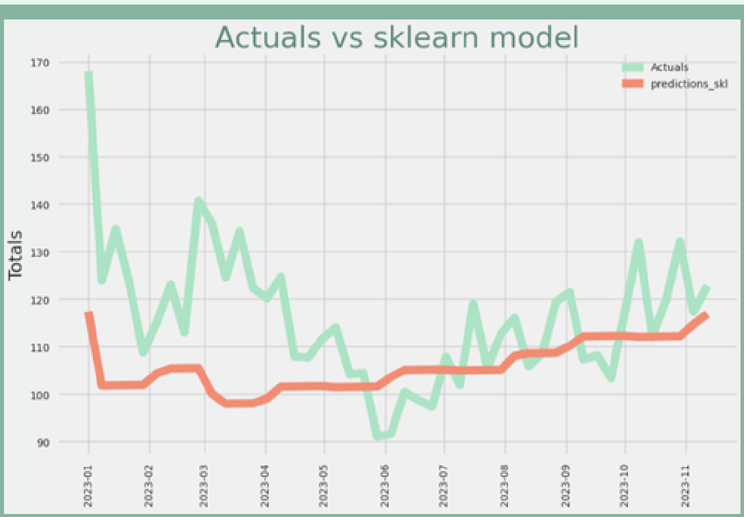
After training and testing each model, we saved an R-squared (R^2) and Mean Squared Error (MSE). R^2 indicates how well the model fits the data, and MSE measures the average squared difference between the predicted and actual values. These scores are printed below.

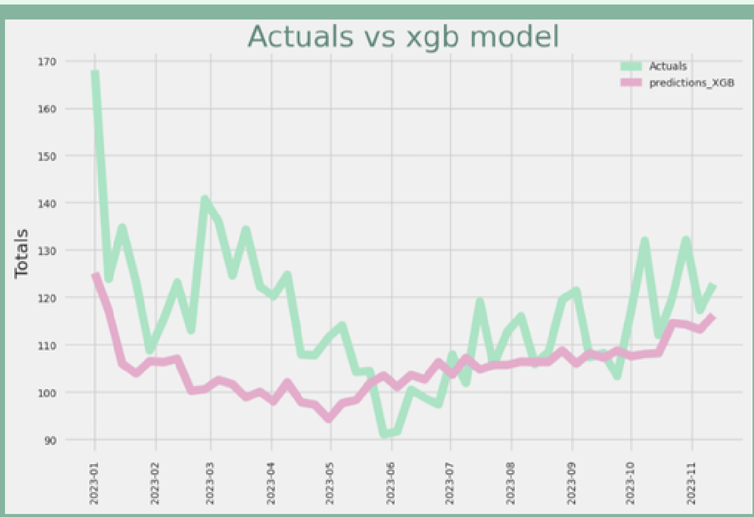
	score_type	sklearn	XGB	PYC	PRO
0	mse	568.0612818157224	558.8320207804576	530.8869795249364	383.74970311564795
1	r2	-0.20800391811985186	-0.1883775435563282	-0.12895135062730279	0.18394166302040993

5 Visualize the Models

Finally, I plotted each model against the actual call volume to get a sense of which model seems to fit the data the best. It's clear that the Prophet, PyCaret Orthogonal Matching Pursuit (OMP), and XGB models do a pretty great job of predicting call volumes and clearly follow the trends of the actual data. However, Prophet wins the day by scoring best on KPIs and clearly following the trend line more closely.





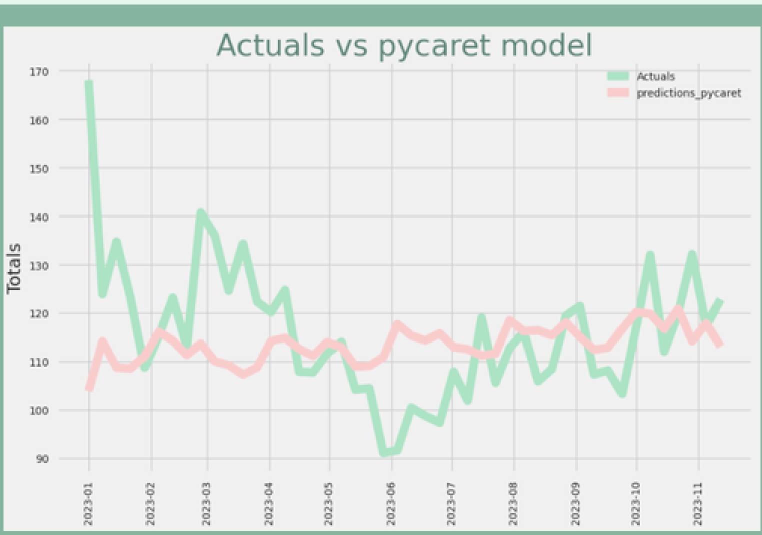


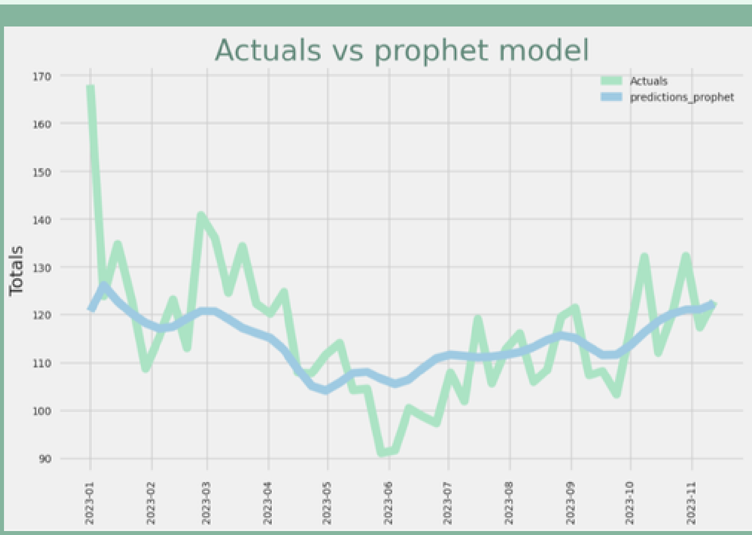
Time Series Forecasting

Conclusion & Additional Reading

Project Code

GitHub Kaggle





Project Conclusion

In conclusion, this project successfully evaluated several time series forecasting models to predict call volumes at the San Francisco 999 contact centre. The exploration revealed distinct daily, weekly, and seasonal trends in call volume data. By comparing various models using R-squared (R^2) and Mean Squared Error (MSE), we identified Prophet and PyCaret's Orthogonal Matching Pursuit (OMP) as the most accurate models for predicting future call volumes. These models effectively captured the underlying trends and patterns within the data, demonstrating their potential to improve call centre operations through better forecasting and resource allocation

Additional Resources on Dem& Forecasting

To read more from me on the Topic of Demand Forecasting feel free have have a look at the following resources I've put together.

1. **Code Repo to Accompany this project** GitHub Kaggle

   Find the most accurate model for predicting future call volumes at the San Francisco 999 contact centre.

   TimeSeries | sklearn | PyCaert | Prophet

2. **Blog Posts or LinkedIn Articles I've published on Topic**

- Mastering Capacity Planning [Blog Post]
- Maximizing Resources: Cost Optimization Audits 💰 [Blog Post]
- Capacity Planning: 5 Metrics to manage [Linkedin Article]

Topic Modeling

Introduction & Explainer

Project Code

GitHub Kaggle

Objective: Turn user generated Reviews & Comments into actionable Insights.

User feedback through comments and reviews offers businesses a window into customer sentiment. However, extracting insights from this vast, unstructured data can be difficult. This project tackles this challenge by leveraging Topic Modeling, a technique that automatically identifies key themes within text data. We'll utilize BERTopic, a cutting-edge approach, to analyze user reviews for four major airlines, uncovering valuable insights into customer perceptions.

Key Concepts & Terminology

Before jumping into the project details, let's define some key terminology.

1 Topic Modeling

Imagine a library with thousandss of books. Topic modelling finds the main categories (history, fiction, etc.) by analysing them & identifying hidden themes based on frequently co-occurring words.

2 Language Model

Think of a language learner trained on massive amounts of text. It can predict the next word, understand text meaning, & even generate text. BERTopic uses a powerful language model called BERT.

3 Vectorization

Comparing large amounts of text is like comparing entire books. Vectorization creates a "fingerprint" (vector) summarizing the key information, making it easier & faster for computers to analyze & compare the text, especially when saving these vectors for later use.

About the Data

This project utilizes a dataset encompassing user-generated reviews for five prominent airlines: Ryanair, EasyJet, Singapore Airlines, Qatar Airways, and Emirates. The data originates from a public comparison website where users share their travel experiences and can potentially rate airline services. The dataset covers a 12-month period from January 2023 to January 2024

NOTE - If you would like to see how I create the Dataset you can see here on my GitHub.

Site LinkedIn GitHub Kaggle kmcguinness33@gmail.com

Topic Modeling

The Process Overview

Project Code

GitHub Kaggle

Training a Model process overview

On a very General level I used the following process to test & train our Topic Model. I'll dive into the specifics of this approach later on.

Data Preprocessing

- **Cleaning**: This involves removing irrelevant information like punctuation, stop words (common words like "the" & "and"), & potentially URLs or HTML tags.
- **Tokenization**: Breaking down the text into individual words or phrases (tokens) for easier processing.
- **Normalization**: Converting all text to lowercase & potentially applying techniques like stemming or lemmatization to reduce words to their root form.

Hyperparameter Tuning

This involves experimenting with different settings within BERTopic to optimize its performance. These settings can include the number of topics to identify, the learning rate, & the batch size for training. This is often an iterative process, requiring testing different configurations & evaluating their effectiveness.

Model Training

Once the data is preprocessed & hyperparameters are tuned, the actual training process begins. BERTopic uses the prepared data to learn the underlying relationships between words & identify distinctive topic clusters.

Topic Labeling

After training, BERTopic assigns descriptive labels to each identified topic cluster. This helps human users understand the meaning & thematic focus of each topic.

Evaluation & Refinement

This final stage involves evaluating the quality & relevance of the identified topics. It's typical to review the topics & their associated words, potentially adjusting hyperparameters or data pre-processing steps for further refinement if needed.

Topic Modeling

The Process: Hyper - Parameter Testing

Project Code

GitHub Kaggle

1 Data Preprocessing

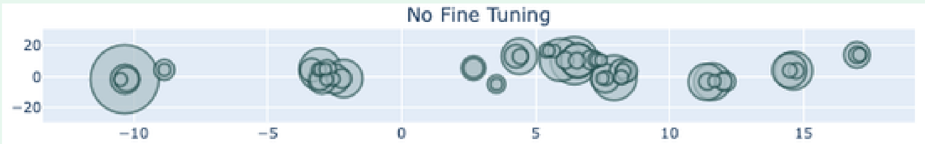
Data cleaning and standardisation were crucial initial steps. Dates were particularly challenging, requiring identification and conversion of expressions like '2 days ago' & 'Updated' into standard date-time formats. To mitigate potential bias towards specific airlines & enhance dataset size, all airline references were replaced with generic placeholders, rendering the comments 'airline-agnostic' for analysis.

2 Hyper Parameter Tuning

To expedite processing, both the original and anonymized comments were converted into numerical vectors in a single instance, avoiding unnecessary repetition during hyperparameter exploration. Five model iterations were then evaluated, each testing the influence of different hyperparameter settings on the topic modeling process. The goal was to identify distinct clusters, where the representative documents within each cluster primarily discuss a common theme.

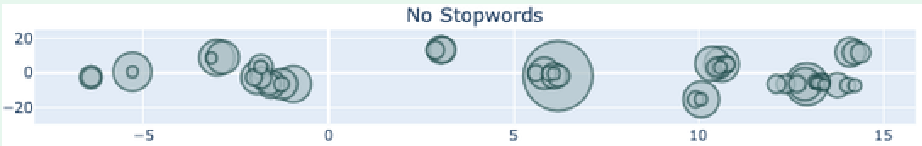
1 No Fine Tuning

No fine Tuning.



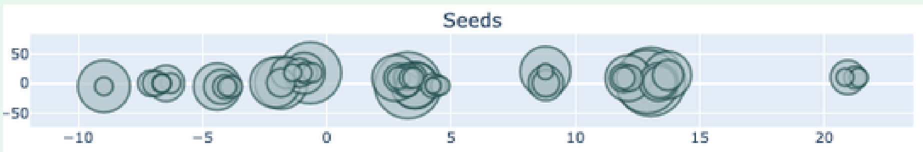
2 Stop words Removed

Common words like "the" or "&" are removed before topic modeling.



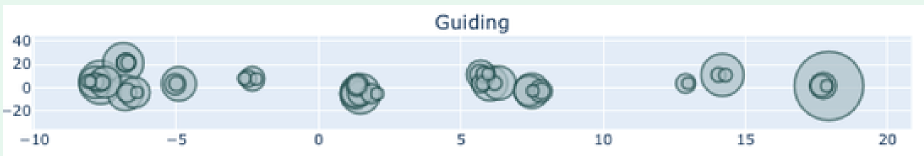
3 Seed Words [seed_words]

Pre-defined words or phrases used to guide topic modeling & improve the quality of topics identified.



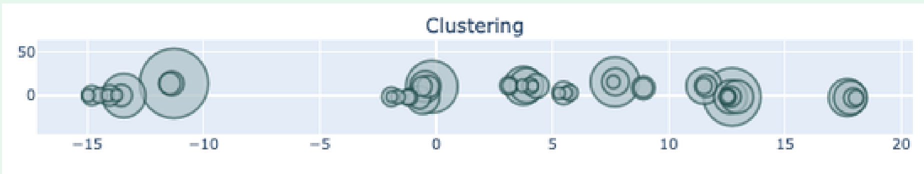
4 Guiding [seed_topic_list]

Similar to Seed Words but lists of seeds representing Groups used.



5 Clustering [hdbscan_model]

Grouping similar documents or data points together to identify distinct topics or themes.



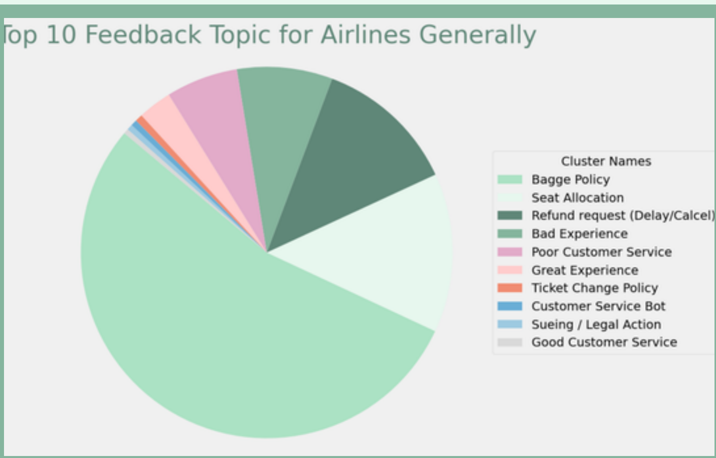
Topic Modeling

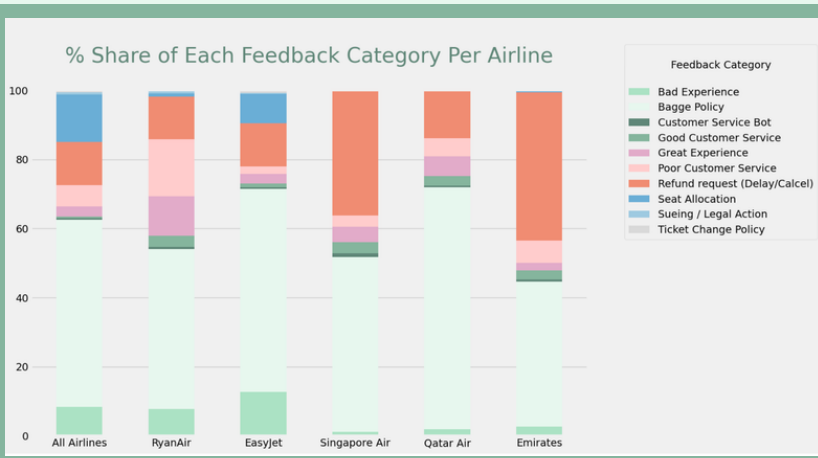
The Process: Model Training

Project Code

GitHub Kaggle

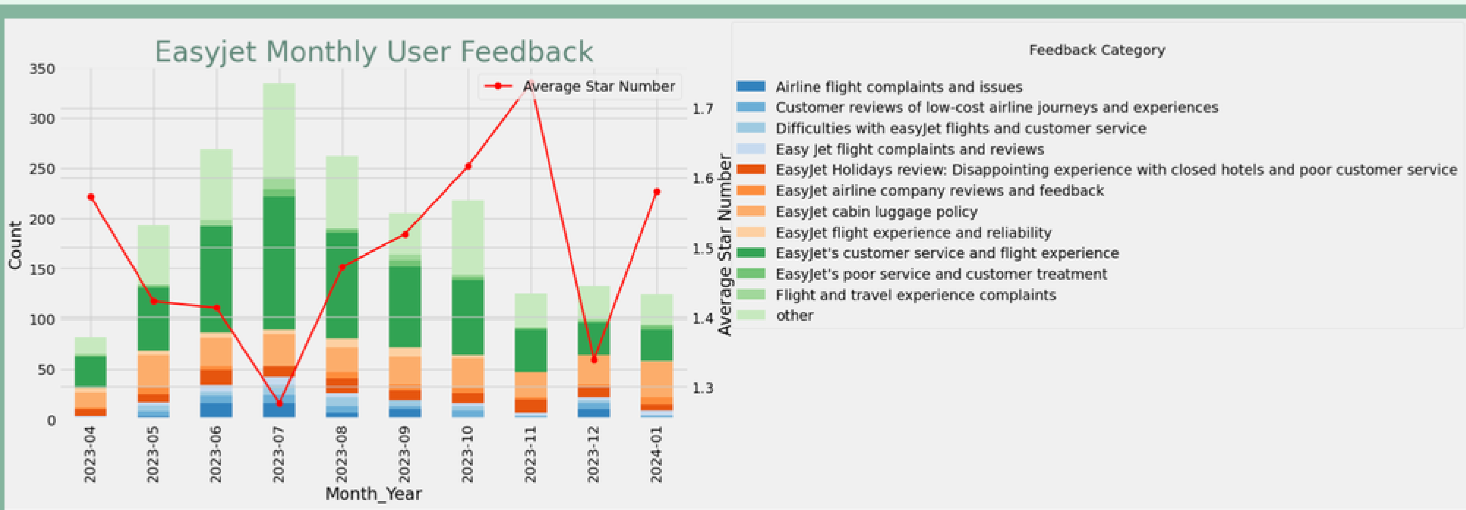
Evaluation of different clustering algorithms revealed that Models using the HDBSCAN clustering paramater yielded the most distinct clusters with representative documents consistently focusing on the same theme. Having established the optimal clustering approach, I sought to gain insights into broader consumer feedback within the airline industry. To facilitate comprehension, custom labels were assigned to these clusters before delving deeper into the specific insights they offered regarding both the general airline landscape and the performance of our own airlines.





3 Model Training

Following hyperparameter selection, I applied the chosen model to my airline-specific datasets. To account for the smaller corpus size of these datasets, I fine-tuned the HDBSCAN model by reducing the minimum cluster size setting from 20 to 5. Additionally, for two datasets, I utilized Cohere's API to generate topic labels using their large language model (LLM). Below is an example of the findings of our Topic Modelling.



Site LinkedIn GitHub Kaggle kmcguinness33@gmail.com

Topic Modeling

The Process: Labelling & Refinment

Project Code

GitHub Kaggle

4 Topic Labeling

To improve the comprehensibility of the automatically generated labels from Cohere's API for two datasets, I additionally performed manual topic labeling and merging using BERTopics' set_topic_labels and reduce_topics functions.

5 Evaluation & Refinment

Finally, the model's results were thoroughly evaluated to understand customer sentiment towards each airline. To ensure efficient resource utilisation, trained models were saved to the Hugging Face Hub for potential future use. Additionally, the potential of Cohere's API was explored to further enrich the analysis. I created two functions. One function used the top 3 most common Documents per topic to create a tl:dr summary of what customers were specifically talking about, function two read the most common Document per topic to summaries all feedback into a single tl:dr of consumer sentiment & suggested next best actions.

1 tl:dr Summary

Based on customer reviews, it seems that you could face issues related to additional fees or unexpected charges when using services through Ryanair.

This could be due to misinformation about the coverage provided by the insurance they purchased through Ryanair or frustration experienced when trying to resolve these issues. To address this situation, you can consider implementing the following steps:

1. Provide more transparent and comprehensive information about the optional insurance policies and any potential additional charges.
2. Make sure that customer service teams are well-equipped to handle such situations by undergoing regular training.
3. Ensure that all relevant information and policies are clearly communicated to customers, including any applicable terms and conditions.
4. Conduct thorough research and fact-finding and quality assurance for customer service, ensuring that your team provides excellent support and solutions to address customers' concerns.
5. Take into account that customers are dissatisfied with the unexpected additional fees they experienced and could face. You could also reach out opportunities for partnership

2 Topic : Document Summary

1). EasyJet cabin luggage policy.

EasyJet's customer service and cabin baggage policy leave a lot to be desired, as detailed in these customer complaints. The airline cancelled a flight and another flight was 21 hours delayed, enforcing an unbalanced refund and compensation policy. The baggage policy is misleading, with many customers paying twice for the same bag, as they are incorrectly guic when onboarding. EasyJet risks losing customers permanently over these antics, leaving relationships tactically disregard in favour of immediate charge increases. To improve these relationships, employ courteous and pragmatic customer supp teams, provide transparent, simple instructions, and execute damage control refunds tactfully.

2). EasyJet Holidays review: Disappointing experience with closed hotels and poor customer service.

EasyJet needs to improve its customer service, especially when it comes to dealing with delays and flight cancellations. Currently, when a flight is cancelled, customers are given a phone number to call for assistance; however, that number is often unreachable. Additionally, staff members are rarely seen in person to help customers, and when they are, they are n adequately trained to deal with such situations. For customers, this results in paying for their own transportation to and fr a hotel (if they are entitled to one), losing out on money for their scheduled activities, and spending valuable time trying tc reach someone for help.

To improve this process, EasyJet should ensure that customers have access to a working phone number, perhaps by posti it on their website or in an email confirmation, and having customer service reps available to answer questions and help w rescheduling flights. EasyJet should also have a clear policy in place for how to handle customer care in the event of a cancellation or delay, and staff should be well-trained in upholding that policy.

47). EasyJet terrible customer service and flight experience..

Common issues amongst these customer reviews are the difficulty in obtaining refunds for cancelled flights and the lack of customer support. These issues could be addressed by making refunds more accessible to customers and ensuring that there is adequate staff and support provided at the airport in the event of a flight cancellation. Improving online claim processes and customer support across different platforms could also help mitigate these issues going forward.

48). Poor airline customer service.

Customer complaints about lengthy flight delays, cancels, and general disorganization from airline staff. Flight from Lisbon to Madrid was delayed several times before ultimately being canceled; customers were left to rebook on their own through the airline app. Additional complaint about a flight to Funchal, Madeira that had to abort takeoff twice and made the customers wait over six hours. Suggested flight itinerary was hilly and treacherous. Urges others not to book with this airline in the future.

49). Poor customer service from an airline company concerning flight cancellations and refunds.

EasyJet's customer service and flight management get poor reviews in these testimonials. In the first case, a flight was cancelled, and the customer was unable to get compensated because EasyJet blamed the customer for not taking a flight that was cancelled. The second testimonial reflects badly on EasyJet for rescheduling flights poorly and providing poor customer service when resolving issues, including compensation. The third testimonial highlights issues with overbooking and poor compensation process with poor customer service.

Based on these reviews, EasyJet would benefit from thoroughly staffing its customer service division with well-trained staff who can respond adeptly to customer issues, especially with compensation claims. Customer complaints should be assessed respectfully and efficiently, with feedback from easyJet staff on how they will address issues going forward.

Topic Modeling

Project Conclusion & Additional Reading

Project Code
GitHub Kaggle

Project Conclusion

This project utilized BERTopic Topic Modeling to transform user-generated comments from a CSV into actionable insights. The approach delivered monthly topic breakdowns with sentiment analysis, key findings & recommendations, & detailed topic explanations. This methodology can be adapted to analyze any user-generated content.

Key Findings:

- Data cleaning & pre-processing optimized model performance (e.g., date clean-up, embedding saving).
- Model testing on generic comments with hyperparameter tuning & custom labels led to improved accuracy.
- Fine-tuning on airline-specific data involved adjusting cluster size & utilizing Cohere's API for topic labeling.
- Cohere's LLM API generated summaries, action suggestions, & topic-specific summaries for further analysis.
- Monthly topic share with sentiment Visualisation provided valuable insights.
- Overall, this project demonstrates the effectiveness of BERTopic & Cohere's API in extracting actionable insights from user-generated data. It showcases the potential for this approach to be applied across various domains involving user feedback analysis.

Additional Resources on Dem& Forecasting

To read more from me Topic Modelling feel free have have a look at the following resources

1. Code Repo to Accompany this project GitHub Kaggle

   Turn user generated Reviews & Comments into actionable Insights.
   Text Classification | BERT | NLP | XGBoost

2. Blog Posts or LinkedIn Articles I've published on Topic

- Customer Analytics: The Secrets of Customer Behavior 🌟📊 [Blog Post]
- Customer Segmentation for Personalised Support 🎯💼💰 [Blog Post]
- Safeguarding Your Brand: Brand Protection Strategies 🛡️ [Blog Post]

Site LinkedIn GitHub Kaggle kmcguinness33@gmail.com

Data Visualisation

Call centre KPIs

Project Code

GitHub Kaggle

Objective: Turn Typical Call centre Data into effective Insights to drive performance.

Optimising call centre performance can be challenging, but key metrics and clear visualisations can make it a whole lot easier. This project will analyse data from the City of Cincinnati's Citizen's Information Centre (Q4 2022), which is representative of data commonly exported from softphone or CRM systems. By leveraging this data, we aim to generate actionable insights that can be used to identify areas for improvement and ultimately enhance call centre efficiency and effectiveness.

Key Concepts & Terminology

Before jumping into the project details, let's define some key terminology.

1. **Call Acceptance Rate (CAR)**

   Measures the percentage of calls answered by agents before abandonment.

2. **Service Level Agreement (SLA)**

   Represents the percentage of calls answered within a predetermined timeframe.

3. **Average H&le Time (AHT)**

   Represents the average time spent by agents resolving a call, including talk time, hold time, & post-call work.

4. **Capacity**

   Indicates the maximum number of calls a specific number of agents can handle efficiently.

5. **Concurrency**

   Refers to the number of calls h&led simultaneously by agents.

About the Data

This dataset comprises typical call centre data sourced from the City of Cincinnati, Ohio's Citizen's Information Centre. It specifically focuses on incoming calls to the contact centre, representing information commonly exported from a Customer Relationship Management (CRM) or Soft-phone service.

Period: October 1st 2022 tth December 31st 2022 (Q4 2022).

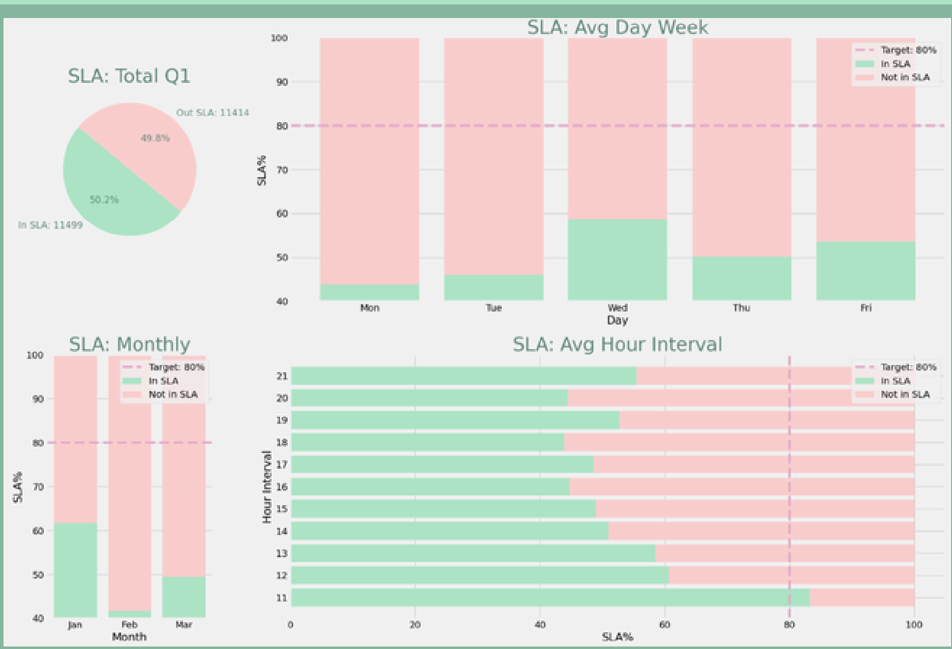
Site LinkedIn GitHub Kaggle kmcguinness33@gmail.com

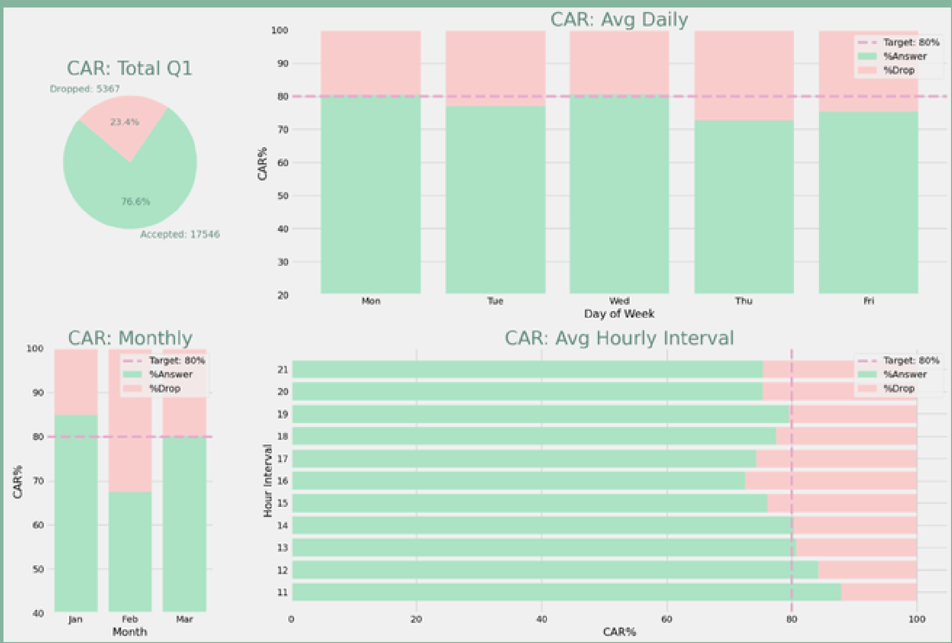
Data Visualisation

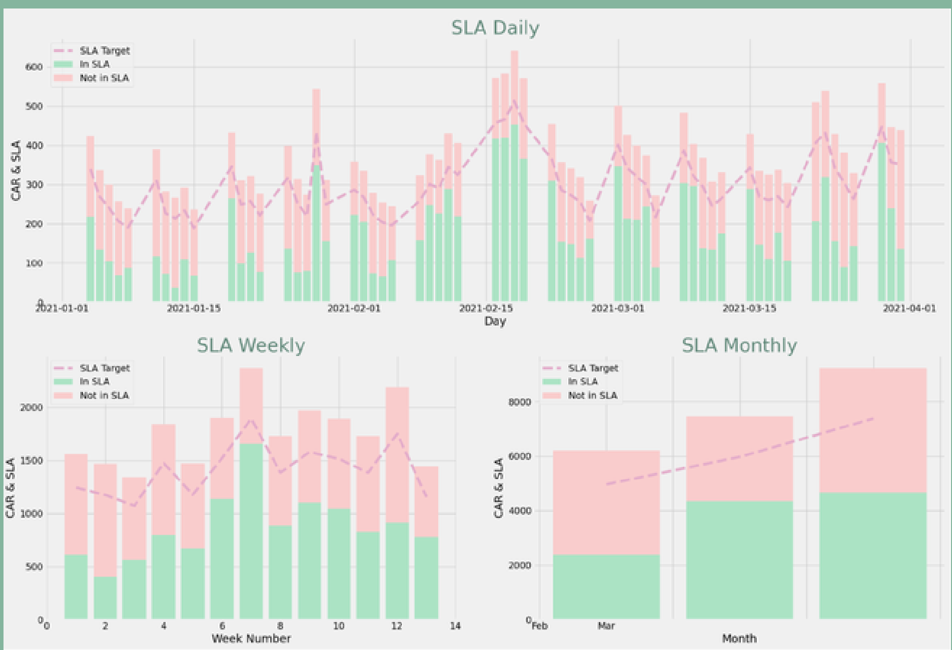
CAR & SLA

Project Code

GitHub Kaggle







1 Service Level Agreement

First, I calculated the overall rate of abandoned calls and how many calls met the SLA. Then I grouped the data by day, hour, and month to see how performance fluctuates across different timeframes. Finally I visualised SLA performance over these timeperiods.

2 Call Acceptance Rate

Similar to SLA, I grouped data by day, hour, and month to explore CAR trends. Then I plotted the results using a bar chart to show the percentage of calls that were answered ("Yes") and abandoned ("No"). Note the substantially higher CAR than SLA results indicating that callers mostly manage to speak to an agent but wait times might be higher than we want.

3 SLA on Actual Volumes

Further analysis plotting actual call volume vs. SLA targets across different time frames revealed a peak in SLA performance coinciding with peak volumes (week 7). This suggests an inverse relationship, potentially indicating that concurrency and scheduling, rather than capacity, are areas for improvement.

Data Visualisation

Call Concurrency & Agent Capacity

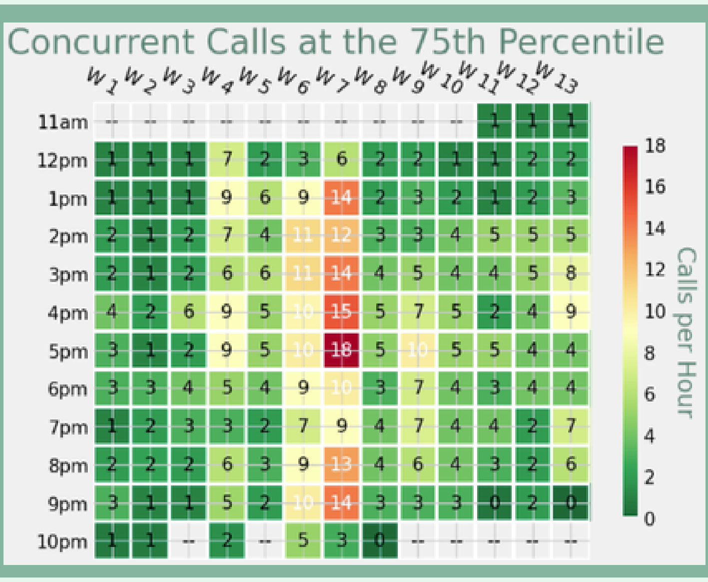
Project Code

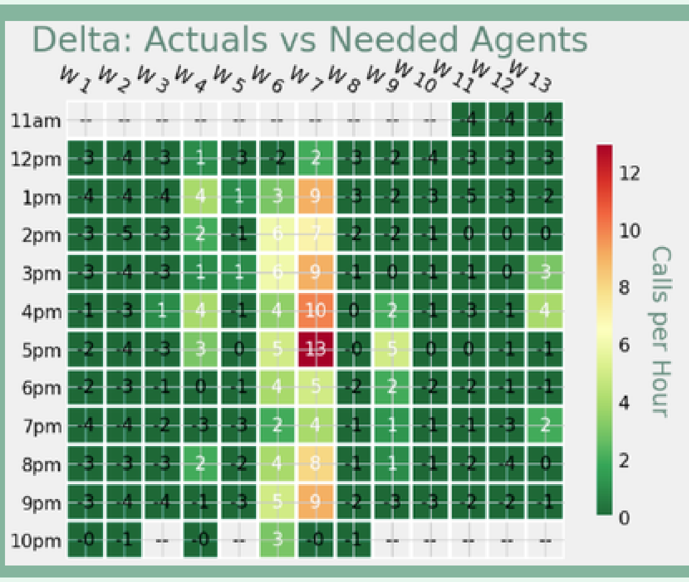
GitHub Kaggle

Underst&ing Call Volume & Agent Availability

Analyzing both call concurrency (simultaneous calls) and agent capacity (available agents) is crucial for optimizing call centre operations. This step focuses on cleaning and formatting call time data, calculating active calls throughout the day, and visualizing hourly and weekly patterns of both metrics. Additionally, we calculate the difference between needed and actual agents, identifying potential staffing gaps. Some key findings:

- Heatmaps reveal significant staffing deficiencies, particularly in Week 7.
- Staffing challenges are most pronounced between 2 PM & 6 PM.

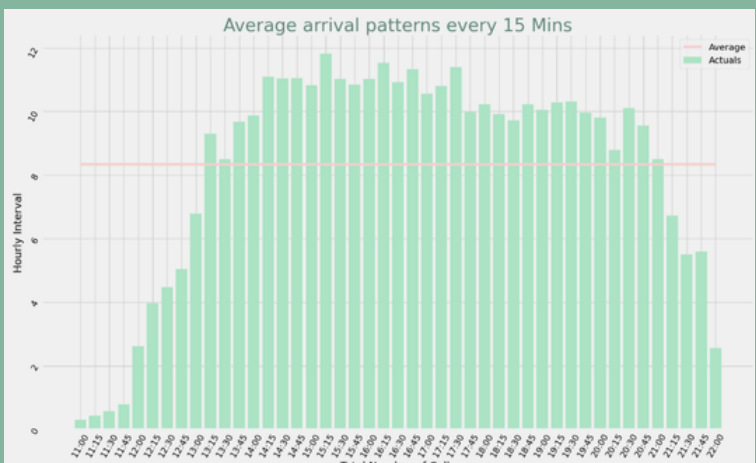


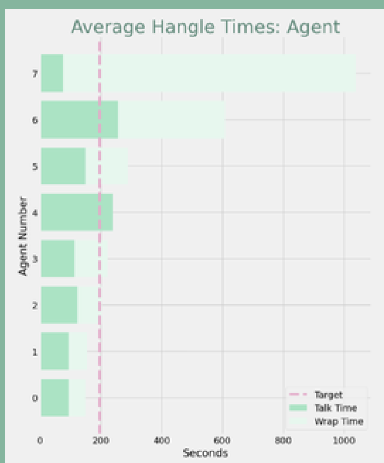


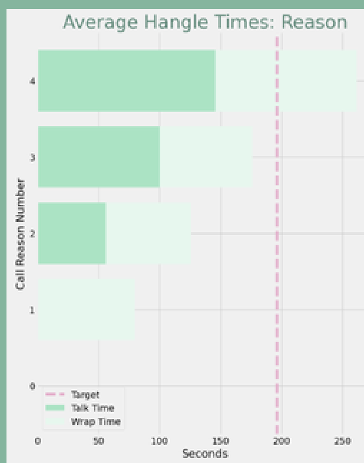
Dissecting Call Patterns & Efficiency

This step analyses both call arrival patterns and average handle times to gain insights into call centre activity and efficiency. Call time data was organised into 15 minute Intervals to highlight peak call periods. I then calculated Average call handle time grouping by Agent & Call Type to understand what factors determine different call times. Some key findings:

- Call reason: We identified that reason 7 takes five times longer to handle than the average call.
- Agent: Agent number 4 consistently exceeds the average handle time.







Data Visualisation

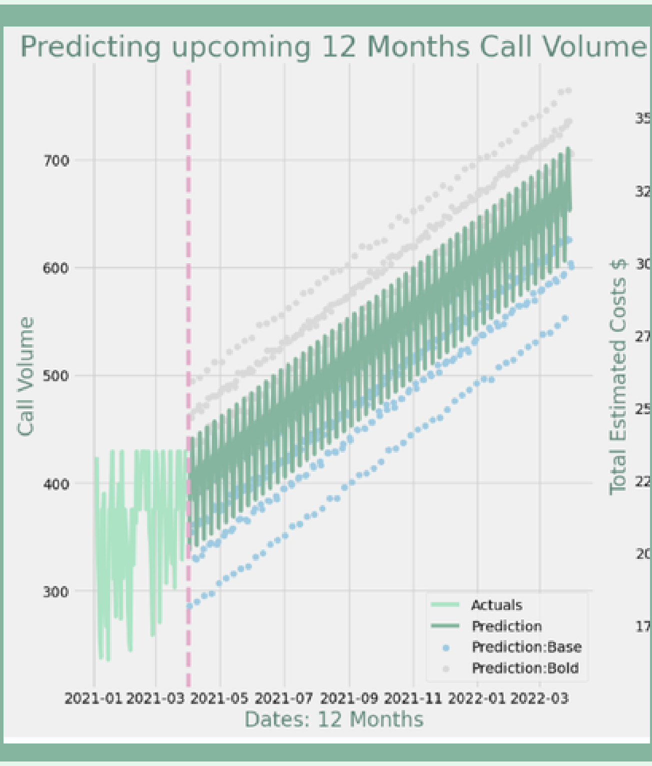
Forecasting & Conclusion

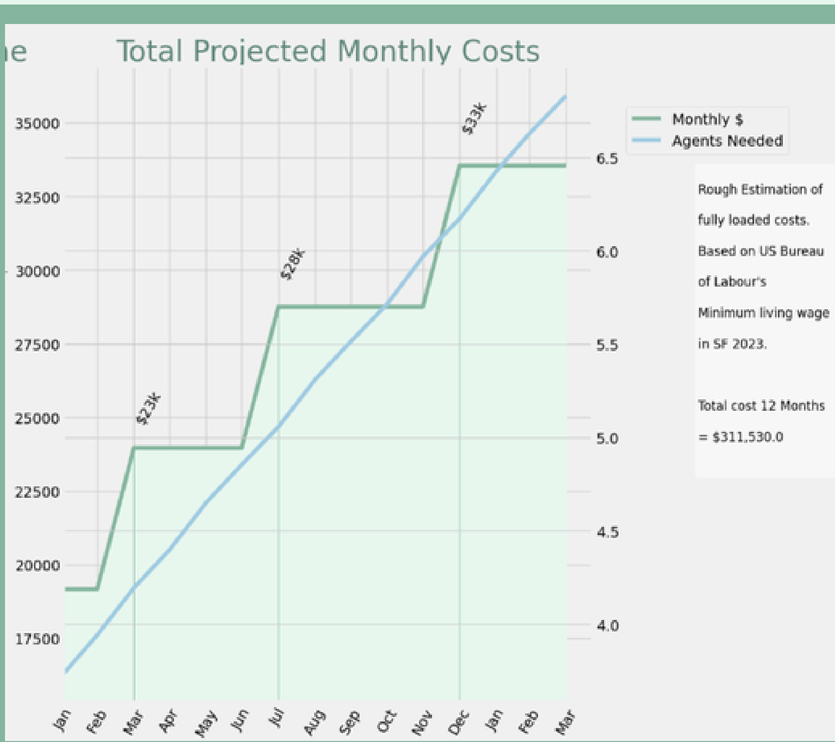
Project Code

GitHub Kaggle

Dissecting Call Patterns & Efficiency

To optimize resource management, I forecasted call volume (including uncertainty bounds) for the next 12 months using the Prophet model. Historical call data informed estimates of future staffing needs, accounting for shrinkage. Leveraging San Francisco's base labor costs, I calculated monthly loaded costs. This comprehensive forecast empowers proactive planning for both staffing and budget allocation.





Project Conclusion

This project successfully transformed raw data into actionable insights for our contact centre. Visualisations & time series forecasting revealed opportunities to improve call handling efficiency, capacity, & future resource allocation. Moving forward, these insights will guide strategic decision-making & enhance the customer experience. Read more of what I've written on the Topic below.

1. Code Repo to Accompany this project GitHub Kaggle

   Turn Typical Call centre Data into effective Insights to drive performance.
   Data Vis | Mathplotlib | NumPy | Pandas

2. Blog Posts or LinkedIn Articles I've published on Topic

- KPI Design & Visualisation in Support Operations 📈🌟📊 [Blog Post]
- Customer Support & Advanced Data Analytics 📈🚀 [Blog Post]

The Data-Powered Leader

Appendix

Project 1	**Time Series Forecasting** GitHub Kaggle Find the most accurate model for predicting future call volumes at the San Francisco 999 contact centre. **TimeSeries | sklearn | PyCaert | Prophet**
Project 2	**Topic Modelling** GitHub Kaggle Turn user generated Reviews & Comments into actionable Insights. **Text Classification | BERT | NLP | XGBoost**
Project 3	**Data Visualisation** GitHub Kaggle Turn Typical Call centre Data into effective Insights to drive performance. **Data Vis | Mathplotlib | NumPy | Pandas**

Articles & Blog Posts I've written

- Mastering Capacity Planning [Blog Post]
- Maximizing Resources: Cost Optimization Audits 💰 [Blog Post]
- Capacity Planning: 5 Metrics to manage [Linkedin Article]
- Customer Analytics: The Secrets of Customer Behavior 🌟📊 [Blog Post]
- Customer Segmentation for Personalized Support 🎯💼💰 [Blog Post]
- Safeguarding Your Brand: Brand Protection Strategies 🛡️ [Blog Post]
- KPI Design & Visualisation in Support Operations 📈🌟📊 [Blog Post]
- Customer Support & Advanced Data Analytics 📈🚀 [Blog Post]

Site LinkedIn GitHub Kaggle kmcguinness33@gmail.com