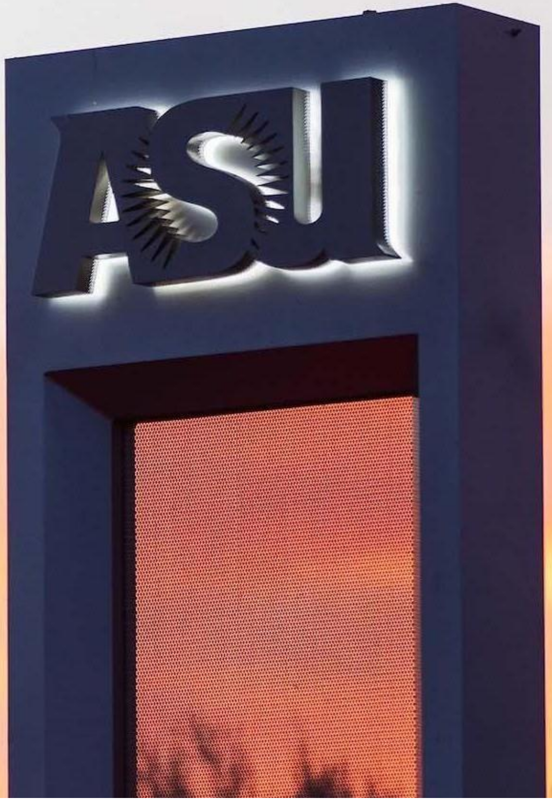
PROJECT PORTFOLIO



ASU Arizona State University

- Ayush Trivedi
- MS Business Analytics (Big Data)

About Me

- **Name:** Ayush Trivedi
- **Current Role:** Graduate Research Assistant in Department of Supply Chain at ASU
- **Interest Areas:** Analytics, Machine Learning, Data Science
- **Professional Experience:**
    - Business Analyst at MedAire Inc., Global Aviation and Maritime Medical Company
    - Business Analyst at Physics Wallah, India's top EdTech Company
    - Business Analyst at Merkle Sokrati, India’s leading digital marketing and CX firm
- **Highlight Courses at ASU:**
    - Enterprise Data Analytics
    - Machine Learning
    - Analytics for Unstructured Data
    - AI and Data Analytics Strategy

Research Interests

- **Predictive Modeling:** Utilizing labeled datasets to develop models, forecasting future outcomes and trends.
- **Data Management:** Ensuring optimal data quality through rigorous cleaning and preprocessing methodologies.
- **In-depth Statistical Analysis:** Employing statistical methods to decipher patterns, correlations, and insights within the data.
- **Time Series Exploration:** Conducting a thorough analysis of time-sequenced data to enhance forecasting accuracy.
- **Computer Vision:** Implementing ML models to interpret and make decisions based on visual data.
- **Customer & Market Insights:** Analyzing customer behavior, feedback and engagement data to drive segmentation, positioning, and data-backed strategic decisions.

Technical Skills

Programming & Tools

- **Languages**: Python, SQL
- **Cloud & Data Platforms:** AWS, Azure, Salesforce
- **Additional:** GIT, Github, Bash, MS Excel (Advanced), Alteryx, Minitab

Python Libraries

- **Data & ML:** Pandas, NumPy, Sklearn, PyTorch, TensorFlow
- **Visualization & Time Series:** Matplotlib, Seaborn, Prophet
- **ETL & Management:** PySpark, Apache Airflow, mlflow

ML & AI

- **Applied Techniques:** Regression, SVM, KNN, Decision Trees
- **Neural Networks:** CNN, RNN
- **Forecasting:** Time Series Analysis, ARIMA, fbProphet
- **Model Management:** MLflow, evidently
- **NLP & Computer Vision:** OpenCV, YOLO V8, NLTK

Databases & Dashboards

MSSQL, MySQL, Looker Studio, PowerBI, Tableau, Google Analytics, Firebase, Appsflyer, MixPanel

My Projects

- **SCM 517:** Design of Experiments for Lego Car Race optimization
- **CIS 508 Research:** RSNA Breast Cancer Detection
- **CIS 509:** Customer Sentiment & Topic Analysis of Restaurant Reviews
- **SCM 593:** Spring 2025 Internship Project
- **CIS 515:** Automated Wait-Time Estimation at Campus Eateries

Design of Experiments for Lego Car Race optimization

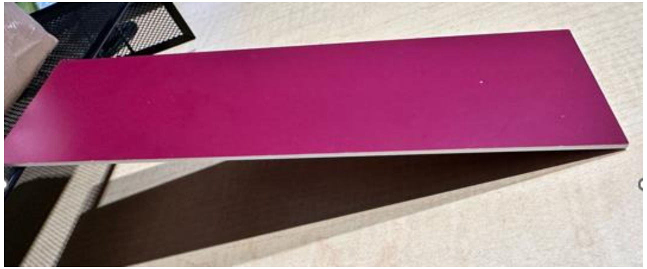
SCM 517 Business Process Analytics

Github: https://github.com/Relostar-Devil/Design-of-Experiments-DOE.git

Project Scope

- **Project Objective:** Design and optimize a Lego-based race car using Design of Experiments (DOE) techniques to maximize the distance traveled under controlled experimental conditions.
- **Response Variable (Y):** Distance traveled by Lego race car.
- **Design Factors Considered:** Tire size, windscreen size, axle length, and car slant configuration.
- **Experimental Methodology:** Applied a full factorial Design of Experiments (2^4) approach to systematically evaluate the impact of multiple design parameters on car performance.
- **Statistical Strategy:** Utilized Analysis of Variance (ANOVA) and regression modeling to identify statistically significant main effects and interaction effects influencing the response variable.
- **Optimization Goal:** Determine the optimal combination of design factors that maximize performance while considering cost constraints derived from Bill of Materials (BOM).

Car made from Legos

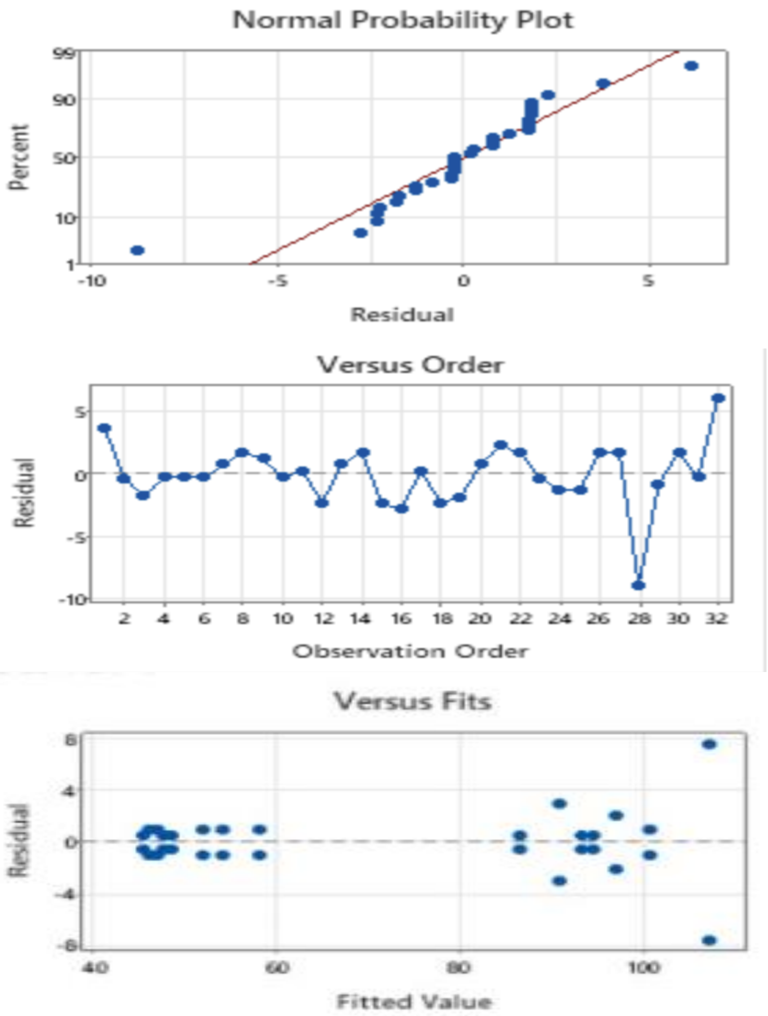


Ramp used to run the car down from

Continued..

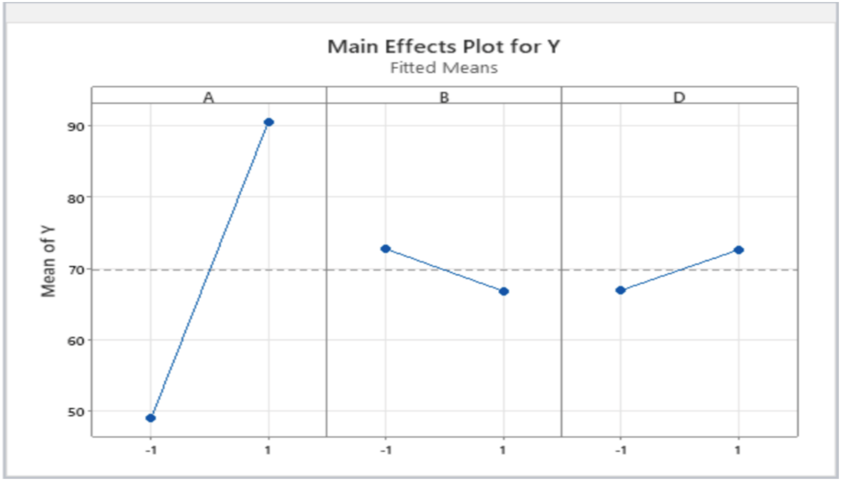
Data Analysis & Modeling

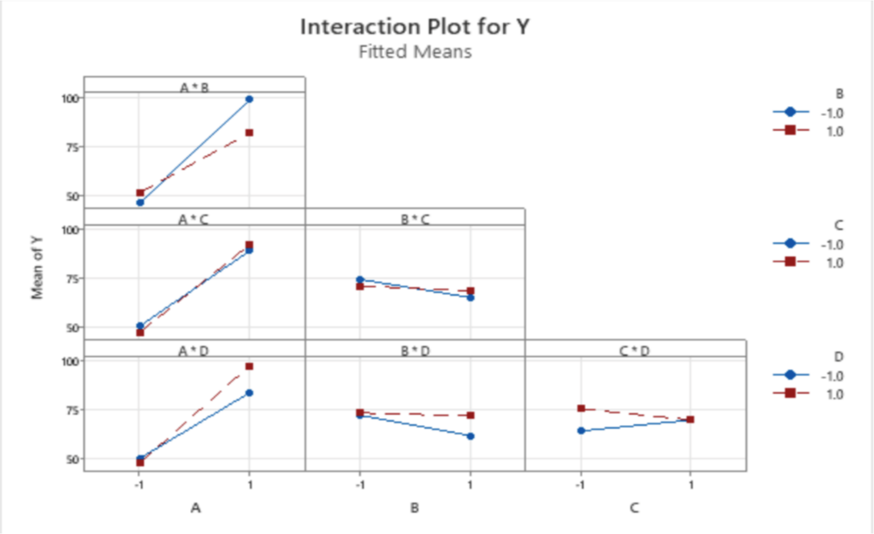
- **Statistical Analysis:** Conducted ANOVA to quantify the significance of main effects and interaction terms across all experimental factors.
- **Model Performance:** Achieved an R-squared value of 99.13%, indicating strong explanatory and predictive capability of the fitted model.
- **Model Validation:** Residual analysis confirmed normality and constant variance, validating the assumptions of the regression model.
- **Interaction Effects:** Notable interaction observed between the size and car slant, demonstrating combined influence on performance outcomes.



Result

- **Main Effects Analysis:** Tire size was identified as the most influential factor affecting the distance traveled by race car.
- **Aerodynamics & Structural Effects:** Smaller windscreen configurations and slanted car designs reduced drag and improved overall performance.
- **Interaction Insights:** A significant interaction between tire size and car slant highlighted the importance of combined factor selection rather than independent optimization.
- **Optimal configuration:** The best-performing design consisted of large tires, a small windscreen, a slanted body, and a shorter axle length.
- **Cost-Performance Trade-off:** The optimized configuration achieved maximum performance at a total cost of $13,200, demonstrating an effective balance between efficiency and material cost.





Tech Stack Used

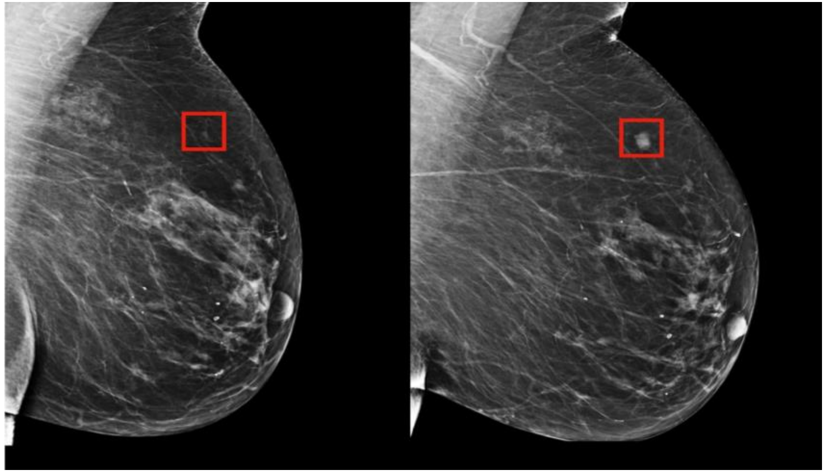
RSNA Breast Cancer Detection

Research Aide – W.P. Carey School of Business

Github: https://github.com/Relostar-Devil/Breast-Cancer-Detection.git

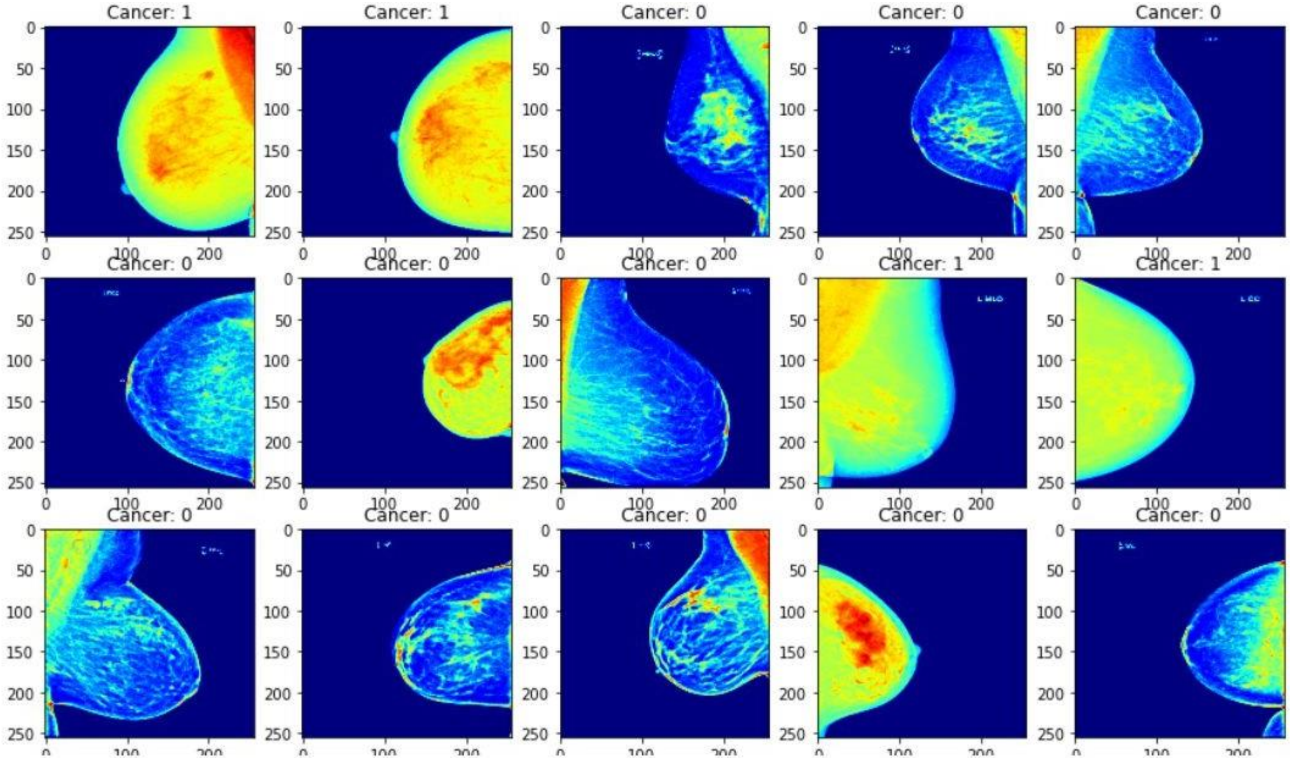
Project Scope

- **Primary Goal:** Develop a model to accurately identify breast cancer using screening mammograms, thereby enhancing the efficiency and precision of radiologists.
- **Challenges in Current Detection Methods:**
  - Dependency on highly-trained radiologists, making the screening process expensive.
  - High incidence of false positives, leading to unnecessary stress and additional medical procedures for patients.
- **Impact of Automation through ML:**
  - Facilitate early detection and treatment, crucial for reducing cancer fatalities.
  - Potentially streamline radiologists' evaluation process of screening mammograms.
- **Potential Outcomes:**
  - Enhance the quality and safety of patient care by improving detection automation.
  - Possibly reduce costs and curtail unnecessary medical procedures.



Mammogram Example

Sample set



Cancer: 1
Cancer: 1
Cancer: 0
Cancer: 0
Cancer: 0
Cancer: 0
Cancer: 0
Cancer: 0
Cancer: 1
Cancer: 1
Cancer: 0
Cancer: 0
Cancer: 0
Cancer: 0
Cancer: 0

Sample Random Images

```
majority_class_df=train_df[train_df['cancer'] == 0].sample(50)
minority_class_df=train_df[train_df['cancer'] == 1].sample(100)

final_df = pd.concat([majority_class_df, minority_class_df])
final_df=final_df.reset_index(drop=True)
final_df.head()
```

- Experiments were conducted to determine the optimal split of the data.
- The following options were evaluated:
  - Balanced dataset with equal number of healthy (500) and cancer (500) patients
  - Unbalanced dataset with fewer healthy patients (50) and more cancer patients (100)
  - Unbalanced dataset with more healthy patients (200) and fewer cancer patients (400)

➢ The unbalanced dataset worked better, and gave better validation accuracy

Normalize the images

After splitting the images into 0: healthy & 1: cancer sub-folders, we normalize the images to ensure that the input data is in a standardized format and scale.

Next we did some data integrity checks before passing the images to our model.

- Checked the min & max values of images
- Checked shapes of images (2 channel (224x224) grayscale image)

```
Cancer images check ===========
Image normalized: 0 1 (224, 224) /kaggle/working/train_images/cancer/60653_2052987229.png
Image normalized: 0 1 (224, 224) /kaggle/working/train_images/cancer/64439_84747386.png
Image normalized: 0 1 (224, 224) /kaggle/working/train_images/cancer/28989_1880776532.png
Image normalized: 0 1 (224, 224) /kaggle/working/train_images/cancer/7053_888903661.png
Image normalized: 0 1 (224, 224) /kaggle/working/train_images/cancer/38311_300211801.png
Image normalized: 0 1 (224, 224) /kaggle/working/train_images/cancer/31582_435931040.png
```

Data Augmentation

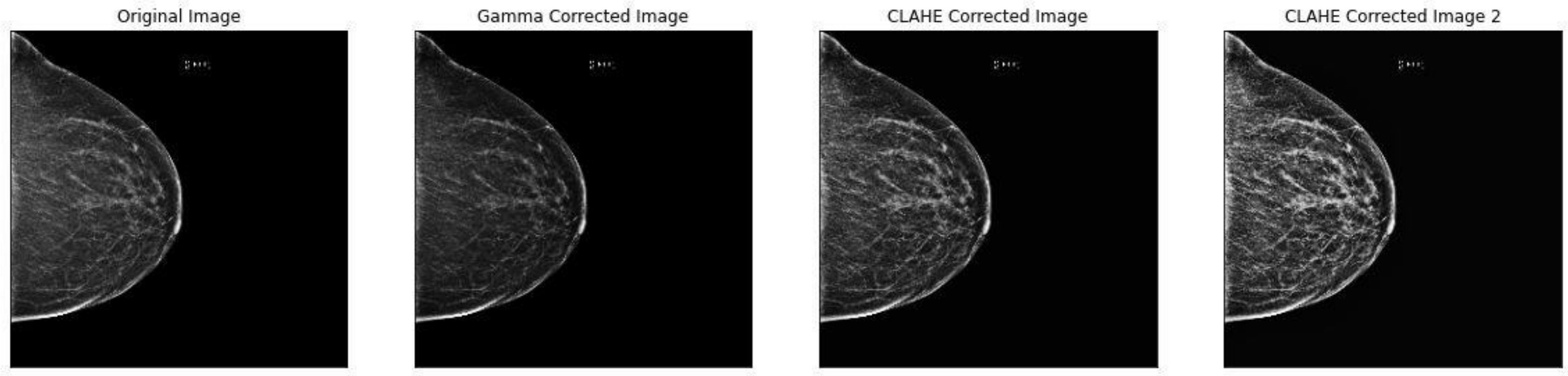
```
majority_class_df=train_df[train_df['cancer'] == 0].sample(50)
minority_class_df=train_df[train_df['cancer'] == 1].sample(100)

final_df = pd.concat([majority_class_df, minority_class_df])
final_df=final_df.reset_index(drop=True)
final_df.head()
```

- Experiments were conducted to determine the optimal split of the data.
- The following options were evaluated:
  - Balanced dataset with equal number of healthy (500) and cancer (500) patients
  - Unbalanced dataset with fewer healthy patients (50) and more cancer patients (100)
  - Unbalanced dataset with more healthy patients (200) and fewer cancer patients (400)

➢ The unbalanced dataset worked better, and gave better validation accuracy

Image Pre-Processing Flow & Results

For this model, we kept the image in the original 3 channel rgb format, and applied 3 filters i.e Gamma, CLAHE 1 & CLAHE 2



Thanks to the image pre-processing steps and augmentations, I was able to get an **accuracy of 84%** on the test dataset, an improvement from the previous 63% when I joined the team.

Tech Stack Used

TensorFlow

OpenCV

GitHub

Customer Sentiment & Topic Analysis of Restaurant Reviews

CIS 509 Analytics Unstructured Data

Github: https://github.com/Relostar-Devil/CIS-509-Analytics-Unstructured-Data-Yelp-Data-Analysis.git

Project Scope

- Analyzing large-scale Yelp restaurant reviews to identify key drivers of customer sentiment, star ratings, and regional dining preferences using unstructured text analytics.

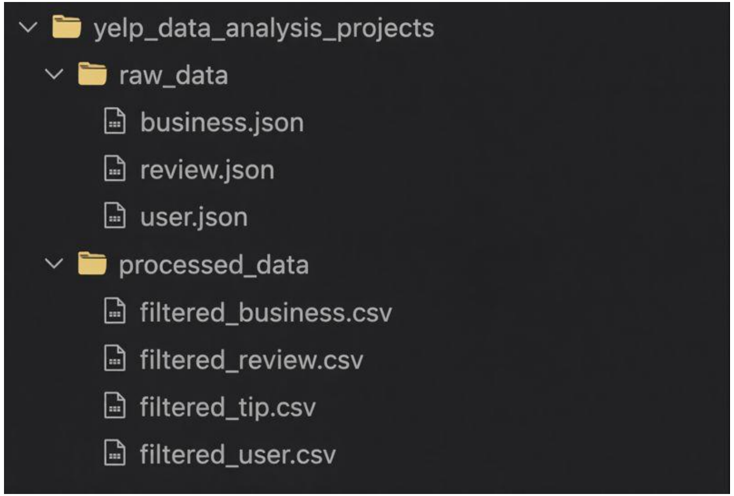
Key Focus Areas:

- Unstructured Data Analytics & Natural Language Processing
- Sentiment Analysis and Topic Modeling
- Business and Regional Insights Generation

Most Common Words in Reviews

Data Collection & Preparation

- Data Source: Yelp Open Dataset (raw JSON format)
- Uploaded raw JSON files to AWS S3 for scalable storage and access.
- Processed and filtered data from S3 based on:
  - Geography: Florida (FL) and Pennsylvania (PA)
  - Cuisine: American, Italian and Chinese
- Converted raw JSON files into structured CSV datasets:
  - Business metadata
  - Reviews
  - Users
  - Tips
- Performed text cleaning, normalization, and dataset validation to ensure analysis-ready data.



Data Pipeline Structure

Exploratory & Sentiment Analysis

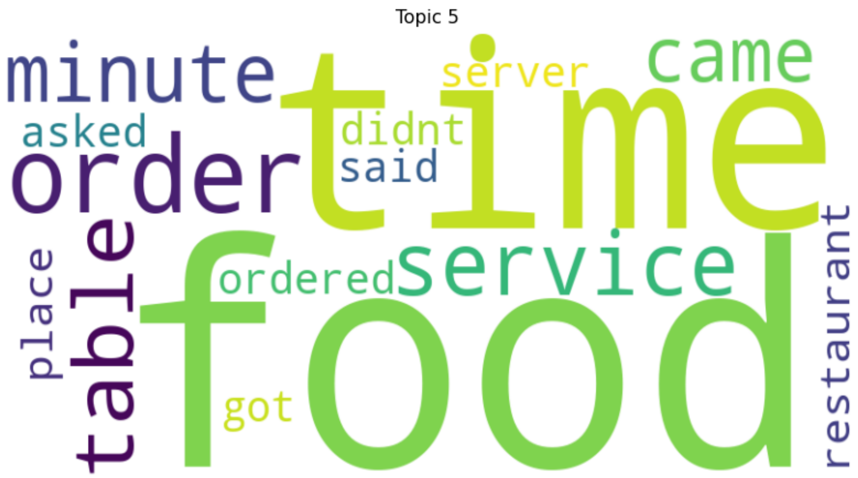
- Conducted Exploratory Data Analysis (EDA) to examine:
  - Review volume and star ratings distributions
  - Cuisine-wise and region-wise trends
- Performed sentiment analysis by comparing:
  - 1-star reviews highlighting service issues, delays, and poor experiences
  - 5-star reviews emphasizing food quality, ambiance, and positive dining experiences
- Applied bigram and trigram analysis along with word cloud visualizations to identify dominant themes.

1-Star Review Word Cloud

5-Star Review Word Cloud

Topic Modeling & Regional Analysis

- Applied BERTopic to extract latent topics from review text data.
- Identified cuisine-specific themes:
    - American: Wings, brunch, happy hour specials
    - Chinese: Authentic dishes, dim sum, service-related complaints.
    - Italian: Pizza, pasta, wine pairings, gluten-free options.
- Regional insights revealed:
    - Florida customers favor Italian cuisine, seafood, outdoor dining, and brunch.
    - Pennsylvania customers show higher preference for wings and pancakes, with concerns around parking and tipping.



Topic modeling highlights service delays and order-handling issues as dominant drivers of customer experience

Insights

Key Statistics:

- Total Reviews Analyzed: 845,306
- Unique Users: 351,921
- Businesses: 8,642

Business Insights:

- Food quality and service consistency are the strongest drivers of positive ratings.
- Operational inefficiencies such as long wait times and poor service contribute to negative sentiment across all cuisines.
- Regional customization of offerings can significantly improve customer satisfaction.

```
Total Review Count: 845306
```

```
Number of Unique Customers: 351921
```

Exploratory output validating dataset scale

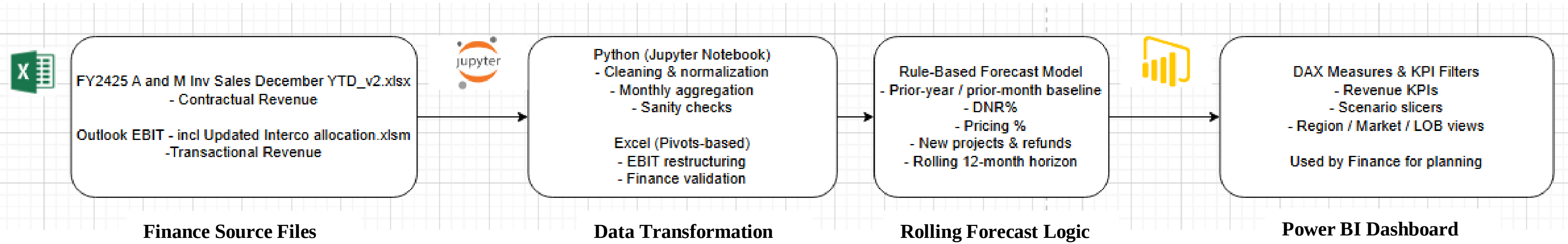
Tech Stack Used

Spring 2025 Internship Project

Rolling Revenue Forecast Model for MedAire Inc.

- **Background:** MedAire Inc. is a global provider of medical and security support services for the maritime and aviation industries, generating revenue through contractual subscription and transactional product sales.
- **Challenge:** Finance relied on static, spreadsheet-based forecasts that lacked flexibility, scenario analysis, and month-over-month adaptability.
- **Objective:** Develop a production-deployable rolling 12-month revenue forecast integrating contractual and transactional revenue streams.
- **Model:** Built a transparent, rule-based forecasting framework aligned with Finance requirements and adaptability.
- **Impact:** Enabled Finance to actively forecast and scenario-plan revenue across 357 projects using a unified model.

MedAire®
An International SOS Company



- **Objective:** Integrate multiple finance data sources into a single forecasting pipeline.
- **Data Sources:**
  - Invoiced Sales data for contractual revenue
  - Outlook EBIT data for transactional revenue
- **Challenge:** Data existed across large, multi-sheet Excel files with different structures and aggregation logic.
- **Approach:** Designed structured ETL workflows using Python and Excel to clean, standardize, and align data for forecasting and reporting.

REQUIREMENT	CHALLENGE	SOLUTION

REQUIREMENT

- Support a rolling 12-month forward forecast
- Enable Finance-driven scenario adjustments
- Maintain transparency and auditability
- Deliver outputs through an interactive Power BI dashboard

CHALLENGE

- Contractual and transactional revenue follow fundamentally different financial behaviors
- Transactional revenue is more volatile and pricing-sensitive
- Forecast logic needed to remain explainable and Finance-controlled for adoption

SOLUTION

- Designed a rule-based rolling revenue forecast model integrating contractual and transactional revenue streams
- Built structured ETL workflows using Python and Excel to prepare finance-grade datasets
- Implemented forecasting and scenario logic in Power BI using DAX for transparency and real-time recalculation

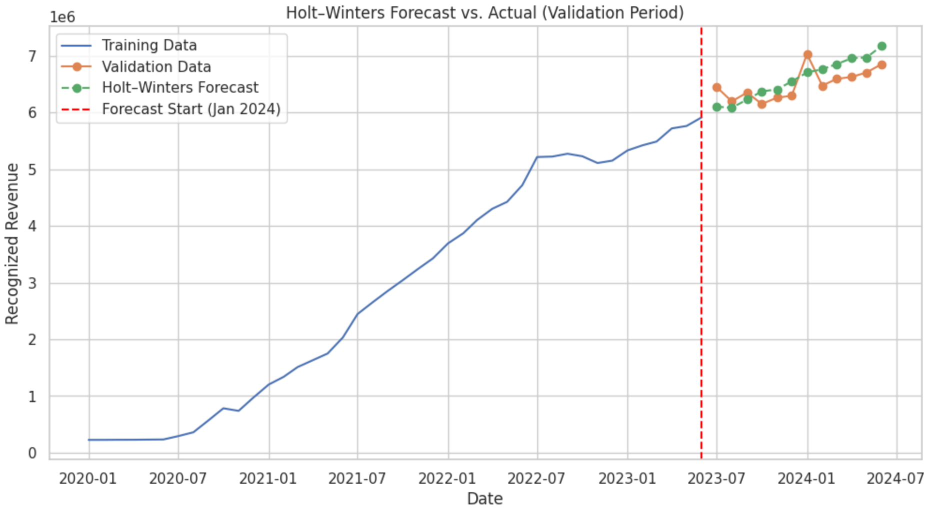
```
df['Finance Period'] = pd.to_numeric(df['Finance Period'], errors='coerce')

# Performing aggregation - Fin year and Finance Period
revenue_fy = df[df['Fin Year'].isin(['2021-2022','2022-2023','2023-2024','2024-2025'])].groupby(['Fin Year','Finance Period'])
['Price'].sum().reset_index()
revenue_fy.sort_values(by=['Fin Year','Finance Period'])

revenue_fy
```

Contractual invoiced sales and transactional EBIT data are cleaned, standardized, and aggregated to monthly project-level revenue.

Prior-year and prior-month revenue baselines are recalculated using rule-based financial logic applied consistently across forecast periods.

Revenue projections are dynamically adjusted using DNR rates, pricing changes, new project additions, and refund assumptions.

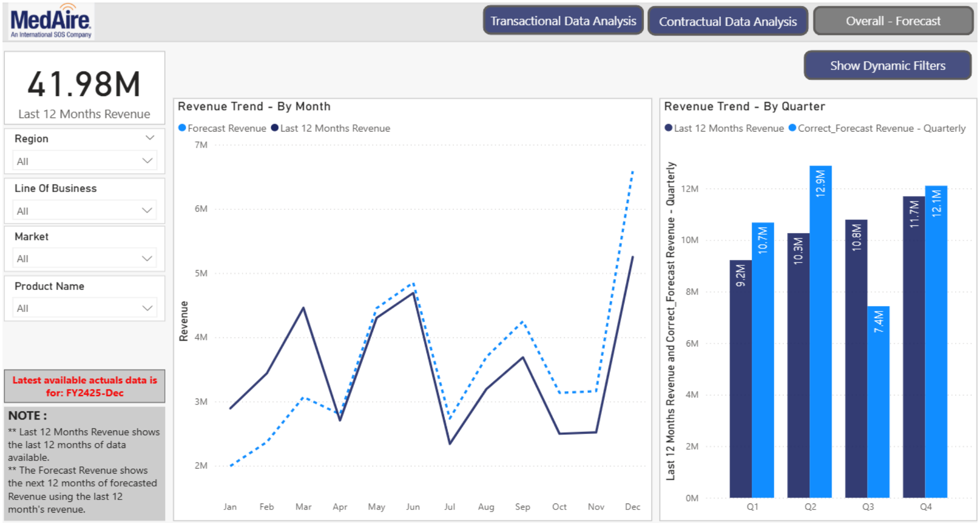
```
hw_model = ExponentialSmoothing
           (train_data['Recognized Revenue'],
            trend='add',
            seasonal='add',
            seasonal_periods=12).fit()
```

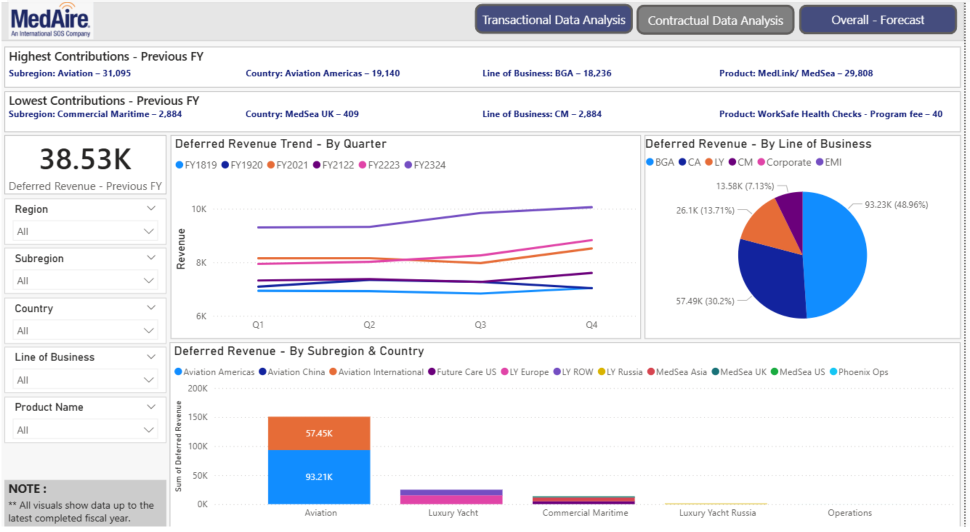


Conclusion

Successful Project Completion: Key Highlights

- Built a production-ready rolling revenue forecast model for aviation and maritime verticals integrating contractual and transactional revenue.
- Enabled Finance-led scenario planning across 357 projects with a 12-month forward horizon.
- Delivered a transparent, auditable forecasting framework aligned with real financial drivers.
- Delivered an interactive Power BI dashboard, enabling informed decision-making and planning for Finance team.





Tech Stack Used

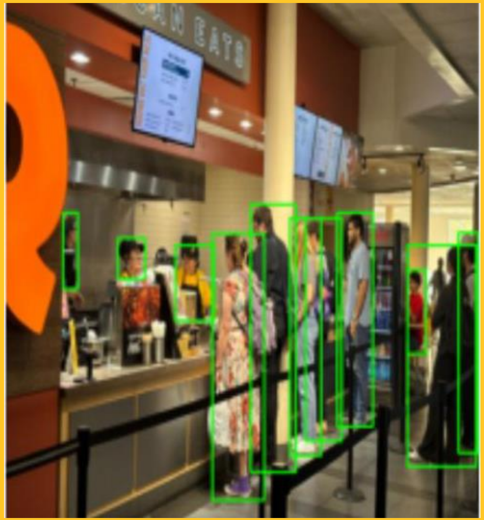
Automated Wait-Time Estimation at Campus Eateries

CIS 515 AI and Data Analytics Strategy

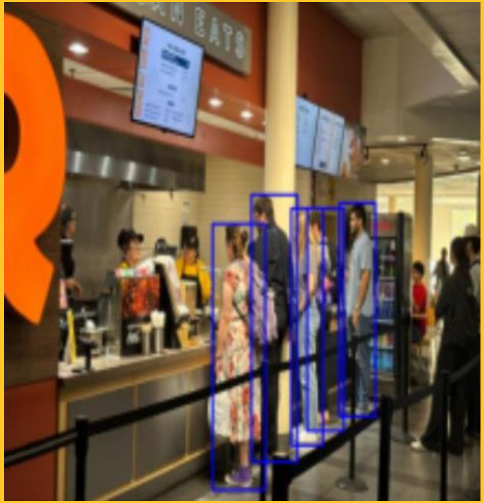
Github: https://github.com/Relostar-Devil/ Real-Time-Queue-Monitoring-at-ASU-Campus-Eateries-using-Computer-Vision-YOLOv8.git

Automated Queue Length Estimation using Computer Vision

- **Objective:** Develop a computer vision-based model to detect and count individuals in cafeteria queues and estimate wait times.
- **Dataset:** Utilized ~3000 custom-labeled cafeteria queue images captured across multiple campus dining locations.
- **Data Filtering:** Removed blurred, low-resolution, and occluded images to improve label quality and model stability.
- **Annotation:** Images annotated using bounding boxes for a single class (person) to preserve privacy.
- **Preprocessing:** Resized images to 640x640 resolution and normalized inputs for model training.
- **YOLOv8 Model:** Implemented and fine-tuned a YOLOv8 object detection model for people detection in queue scenarios.
- **Queue Estimation:** Counted detected individuals within queue regions to estimate average wait time per customer.
- **Results:** Model successfully detected queue lengths and produced consistent wait-time estimates during pilot testing.



All detected



Filtered customers


Thank You!
ASU