**Project Report: Customer Tier Classification**

**1. Problem Statement**

The goal of this project is to classify customers into marketing tiers such as **Gold, Silver, and Bronze** based on **demographic and policy-related data**. In addition, textual data from customer interactions or applications is incorporated using **Natural Language Processing (NLP)** techniques to enrich the structured dataset.

The problem is a **multi-class classification** task where accurate segmentation helps insurance companies target customers effectively for marketing campaigns and personalized services.

Challenges include:

* Handling both **structured and textual data**.
* Ensuring **balanced representation** across tiers for accurate model predictions.

**2. Steps Involved**

**Step 1: Synthetic Data Creation using Faker**

To supplement the COIL 2000 dataset and incorporate realistic textual features, the **Faker library** is used to generate synthetic data such as:

* Customer inquiries (e.g., "Customer inquired about bundling home and auto insurance").
* Names, locations, demographics, and policy details.

This ensures sufficient training data for both structured and text-based features.

**Step 2: Data Preprocessing**

* **Handling missing values** using mean/median imputation for numerical data and mode for categorical.
* **Encoding categorical variables** using One-Hot Encoding or Label Encoding.
* **Text vectorization** using TF-IDF or embeddings to transform text into numerical features.
* **Feature engineering** to combine structured and text-derived features (e.g., sentiment scores, topic labels).

**Step 3: Exploratory Data Analysis (EDA)**

EDA helps in understanding the dataset and patterns:

* **Distribution of customer tiers** visualized using bar plots.
* **Correlation analysis** between features.
* **Text analytics** including most frequent words, word clouds, and sentiment distribution.
* Identification of class imbalances and outliers for proper handling during modeling.

**Step 4: Model Training**

The project explored several **multi-class classification models**:

1. **Random Forest** – Ensemble method suitable for high interpretability and handling structured features.
2. **Gradient Boosting** – Captures complex patterns and improves performance through boosting.
3. **Logistic Regression** – Baseline linear model for structured data.
4. **SVC (Support Vector Classifier)** – Effective for high-dimensional feature spaces.
5. **Neural Network** – Captures non-linear interactions between structured and text-derived features.

**Pipeline setup**:

* Combine structured features and text embeddings.
* Split dataset into training and testing sets.
* Hyperparameter tuning using grid search or random search for optimal performance.

**Step 6: Model Evaluation**

Performance metrics used:

* **Accuracy** – Overall correctness of predictions.
* **F1-score (macro average)** – Evaluates balance across classes, especially for imbalanced datasets.

Cross-validation ensures **robust performance estimates** and prevents overfitting.

**Step 7: Saving Best Model**

The model with the highest F1-score was **serialized using joblib/pickle** for future deployment:

import joblib

joblib.dump(best\_model, "customer\_tier\_classifier.pkl")

This allows the model to be loaded directly into a **production environment or frontend application**.

**Step 8: Frontend Creation using Streamlit**

A **Streamlit web application** was developed to interact with the model:

* Users can **input customer details** (demographics, policy type, textual inquiry).
* Model predicts the **customer tier** instantly.
* Visualizations of **feature importance** and **tier distribution** are displayed.
* Easy deployment and accessibility for non-technical users.

**3. Conclusion**

This project demonstrates a **comprehensive approach** to customer tier classification by:

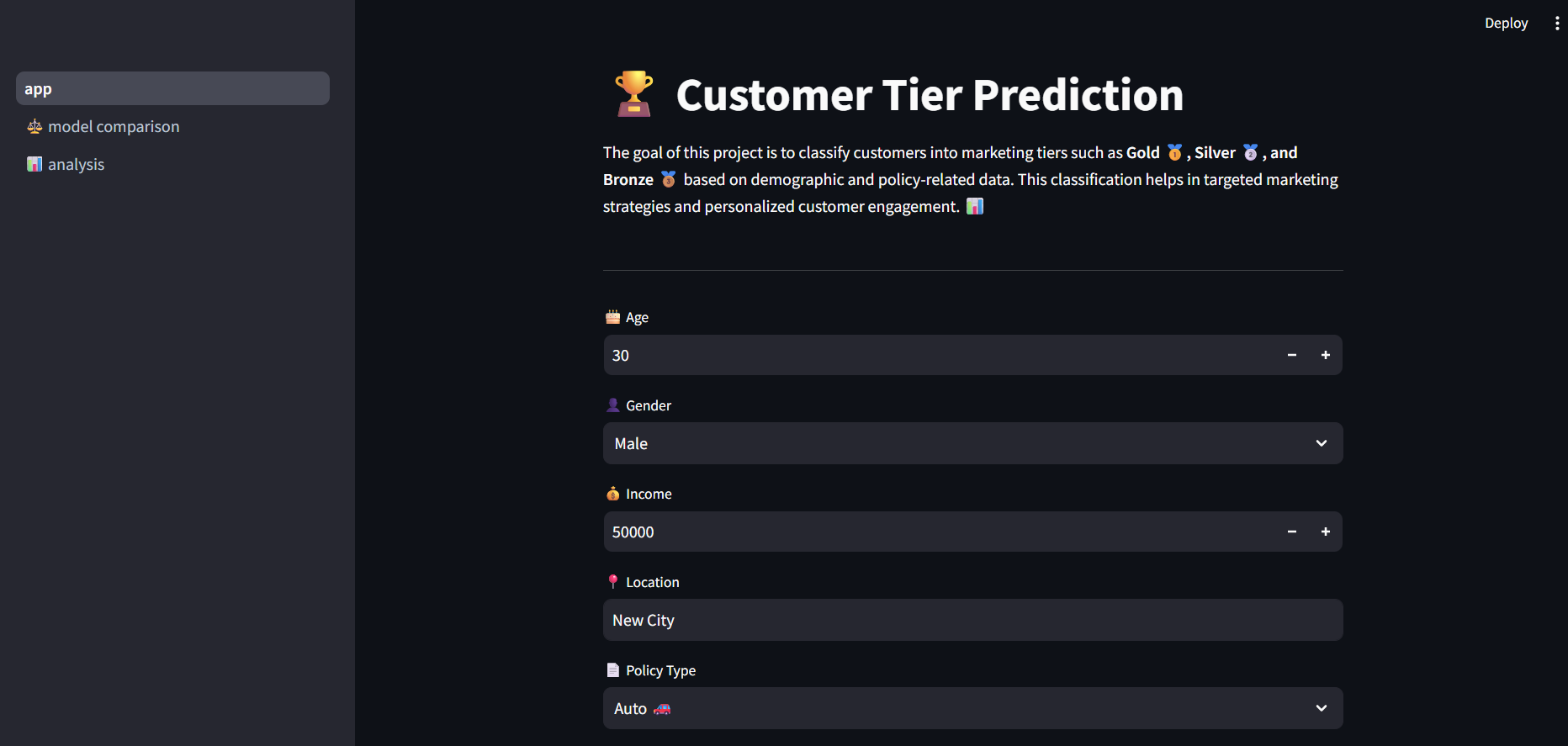
* Combining **structured data and text features**.
* Leveraging **synthetic data generation** to enrich NLP features.
* Using **ensemble and neural network models** for accurate predictions.
* Building an **interactive frontend** for practical deployment.

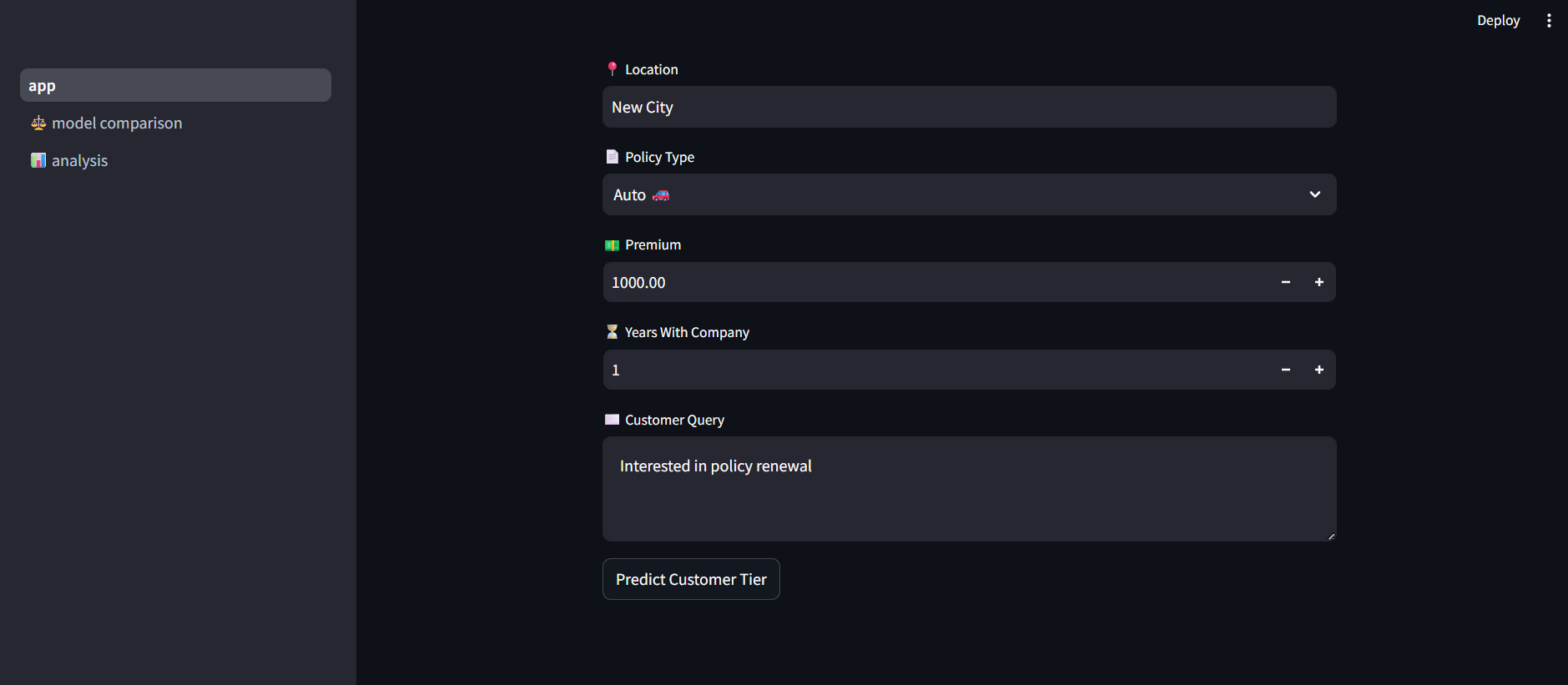
The methodology ensures:

* Enhanced marketing strategies.
* Better customer targeting.
* Scalability for real-world insurance datasets.

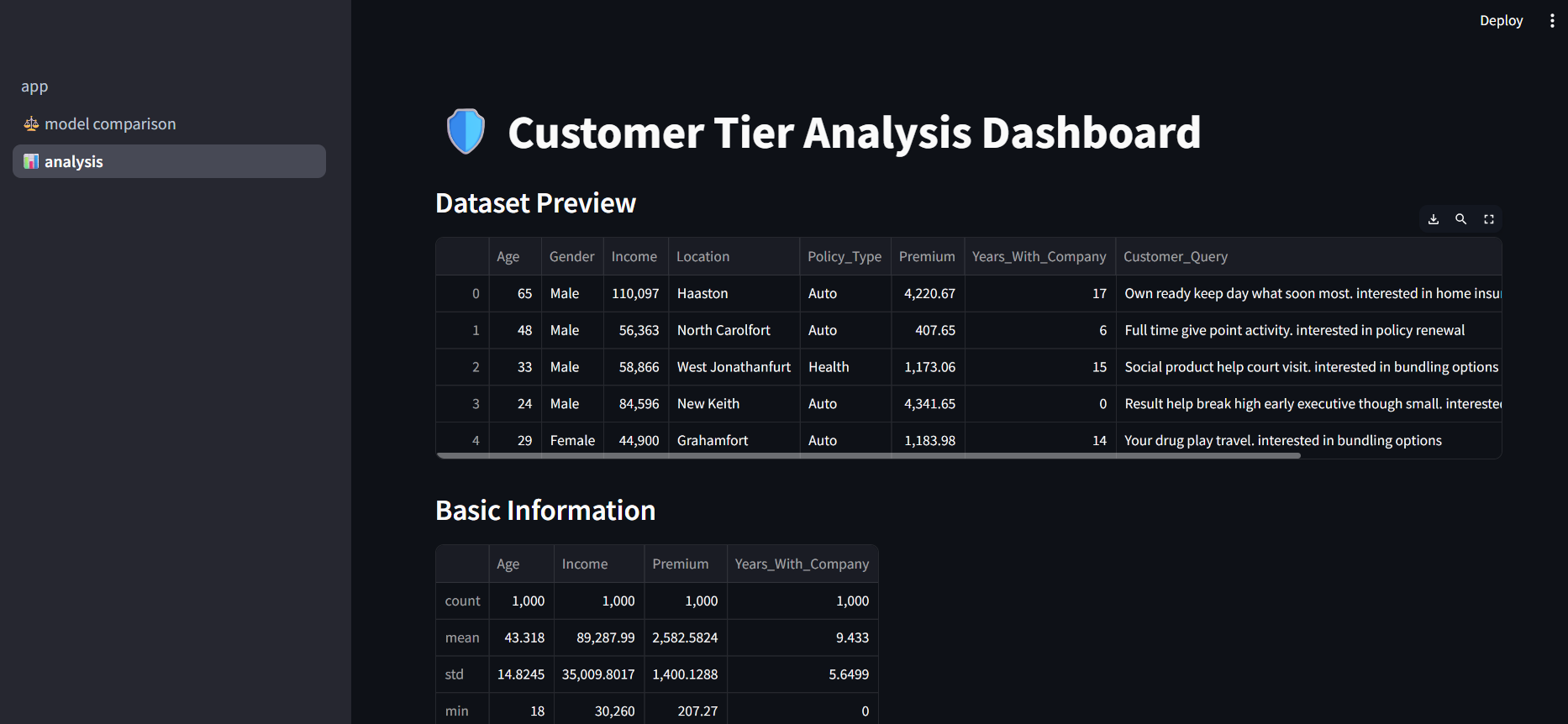
**Frontend Outputs:**

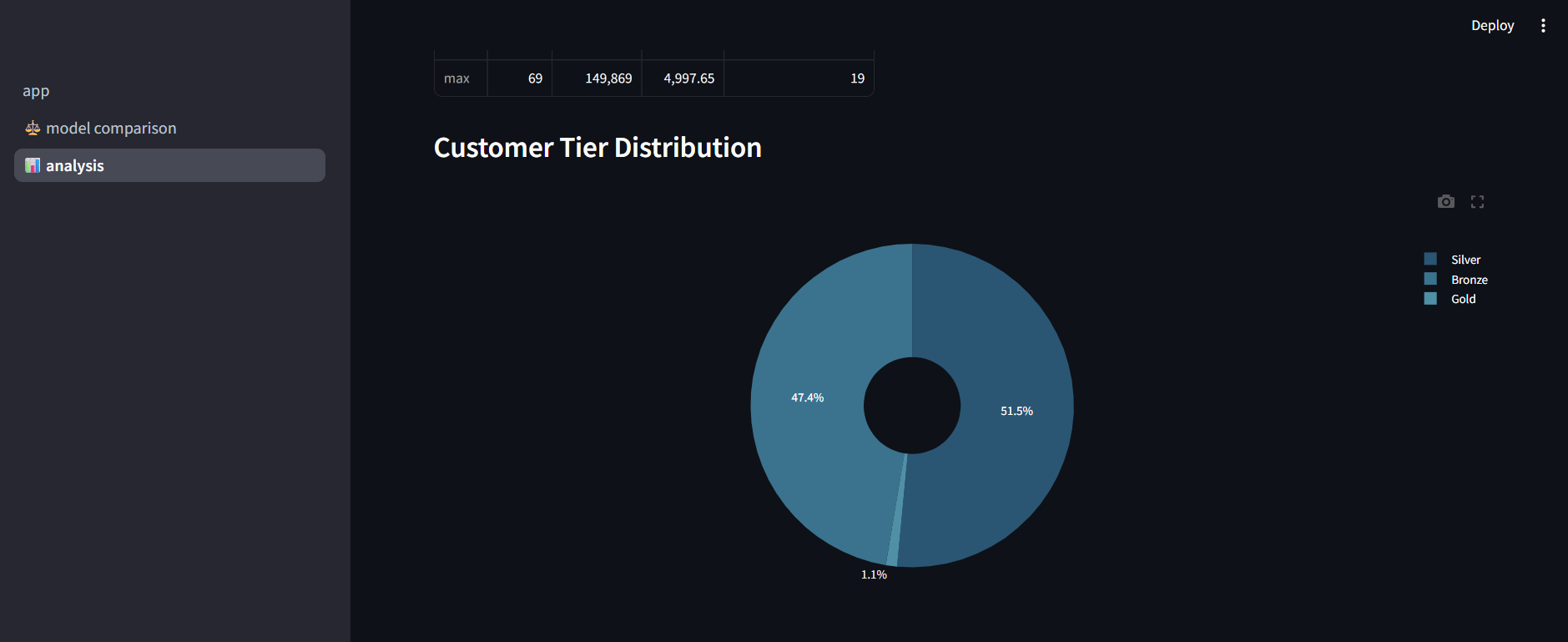
* 1. **New Data Predictions**

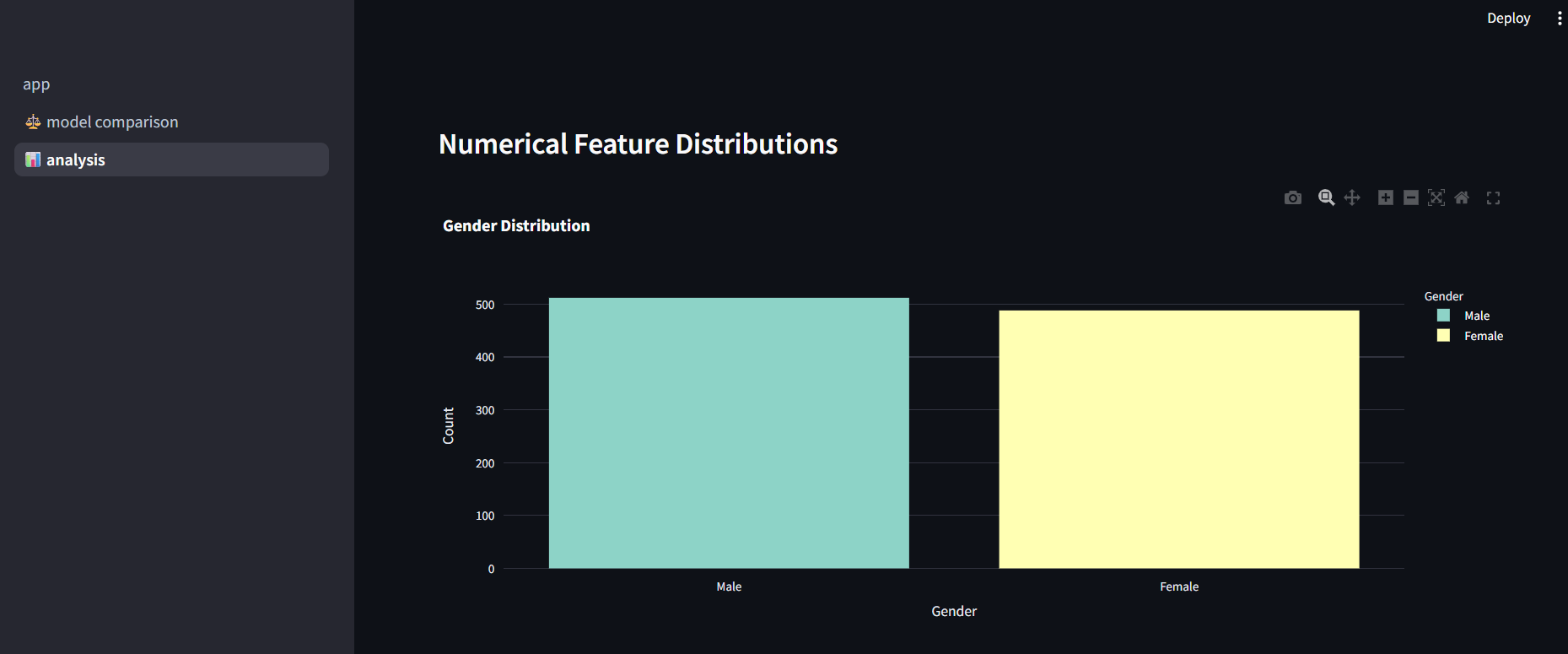
****

****

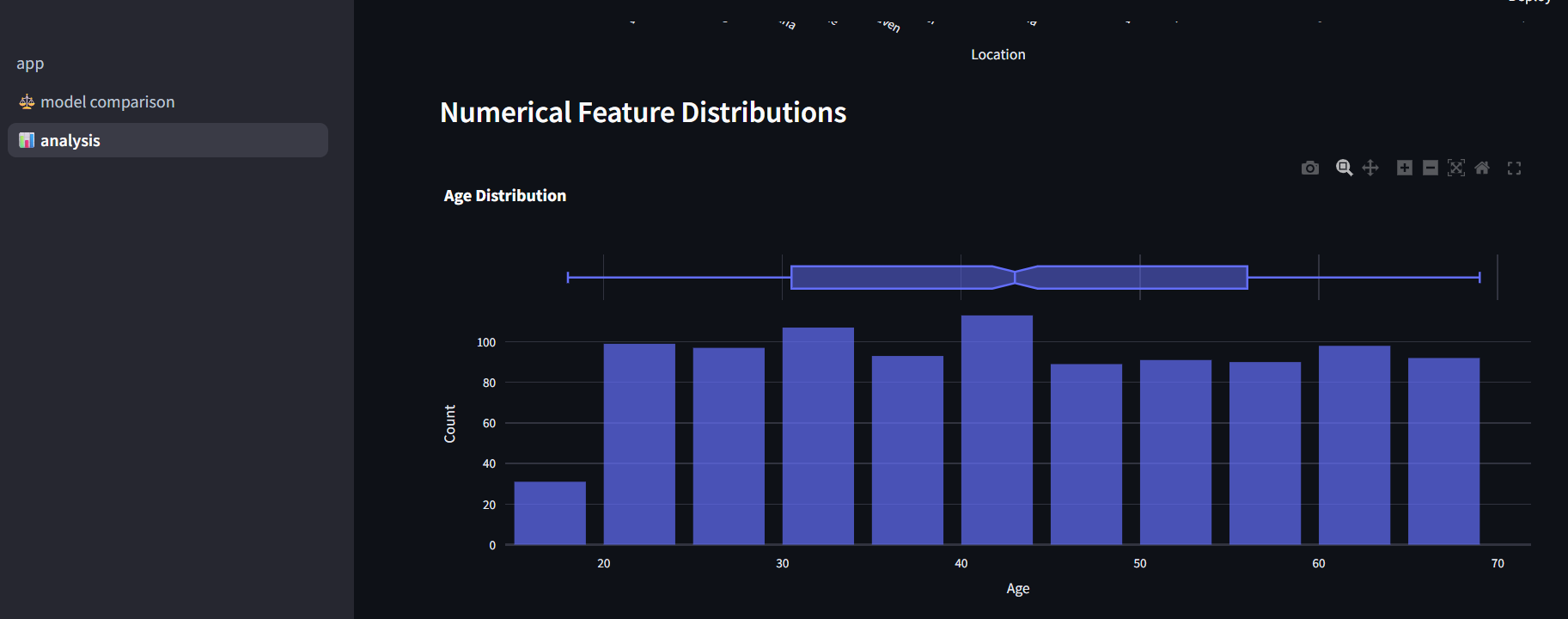
* 1. **Exploratory Data Analysis**

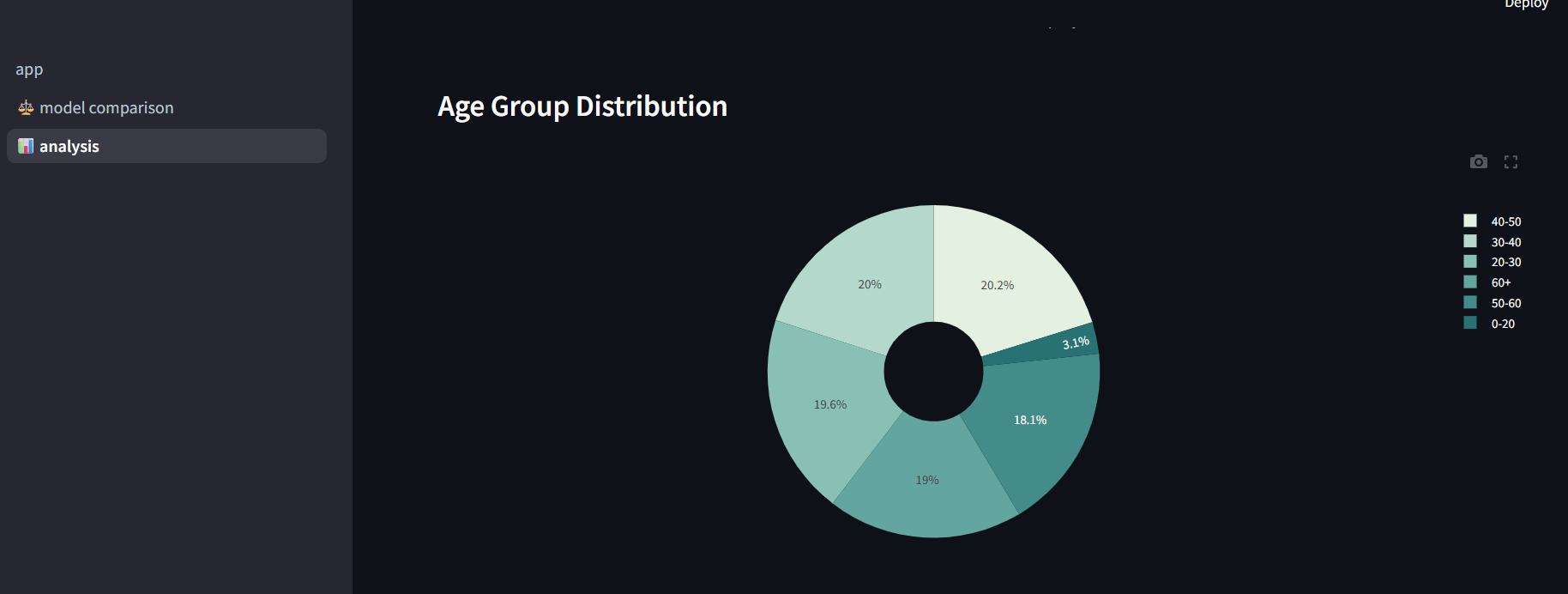
****

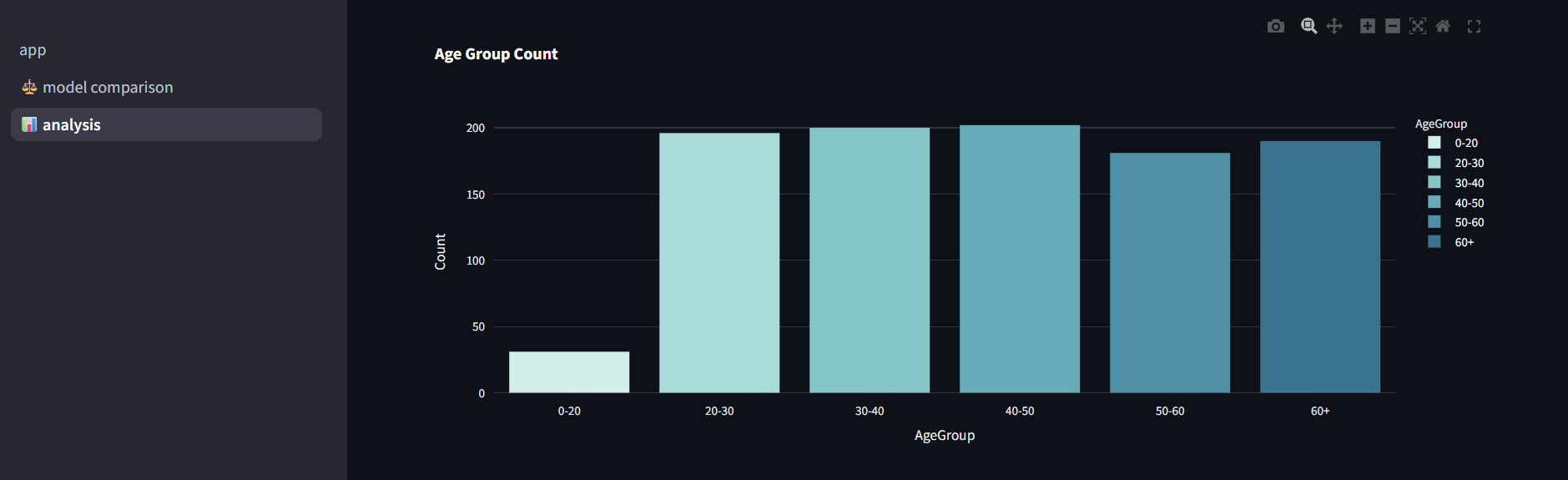
****

****

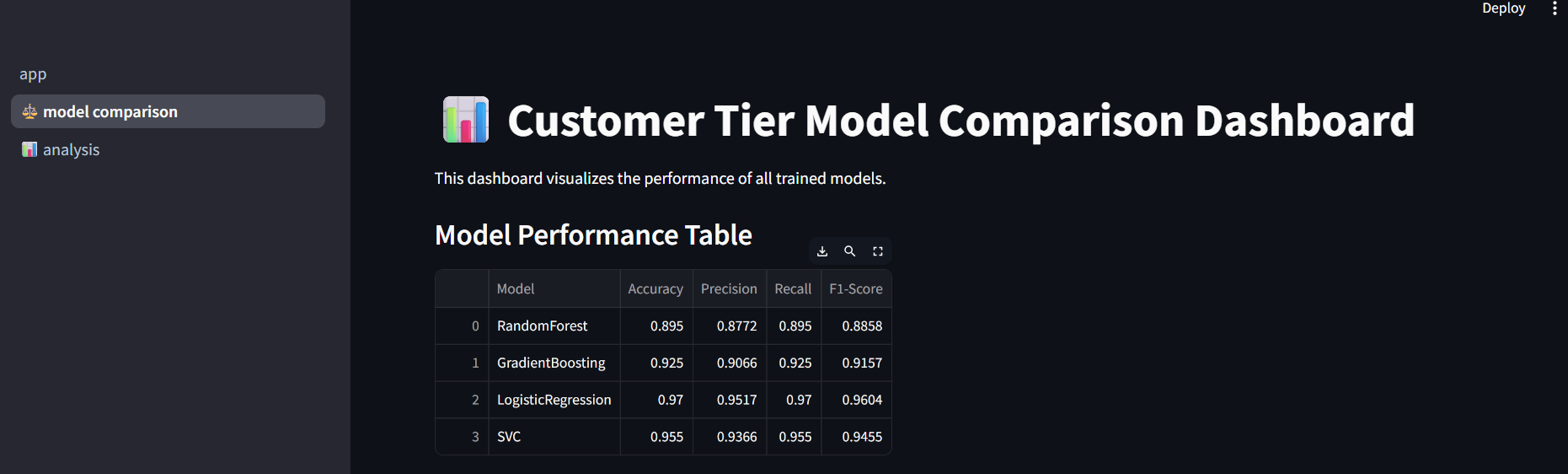
****

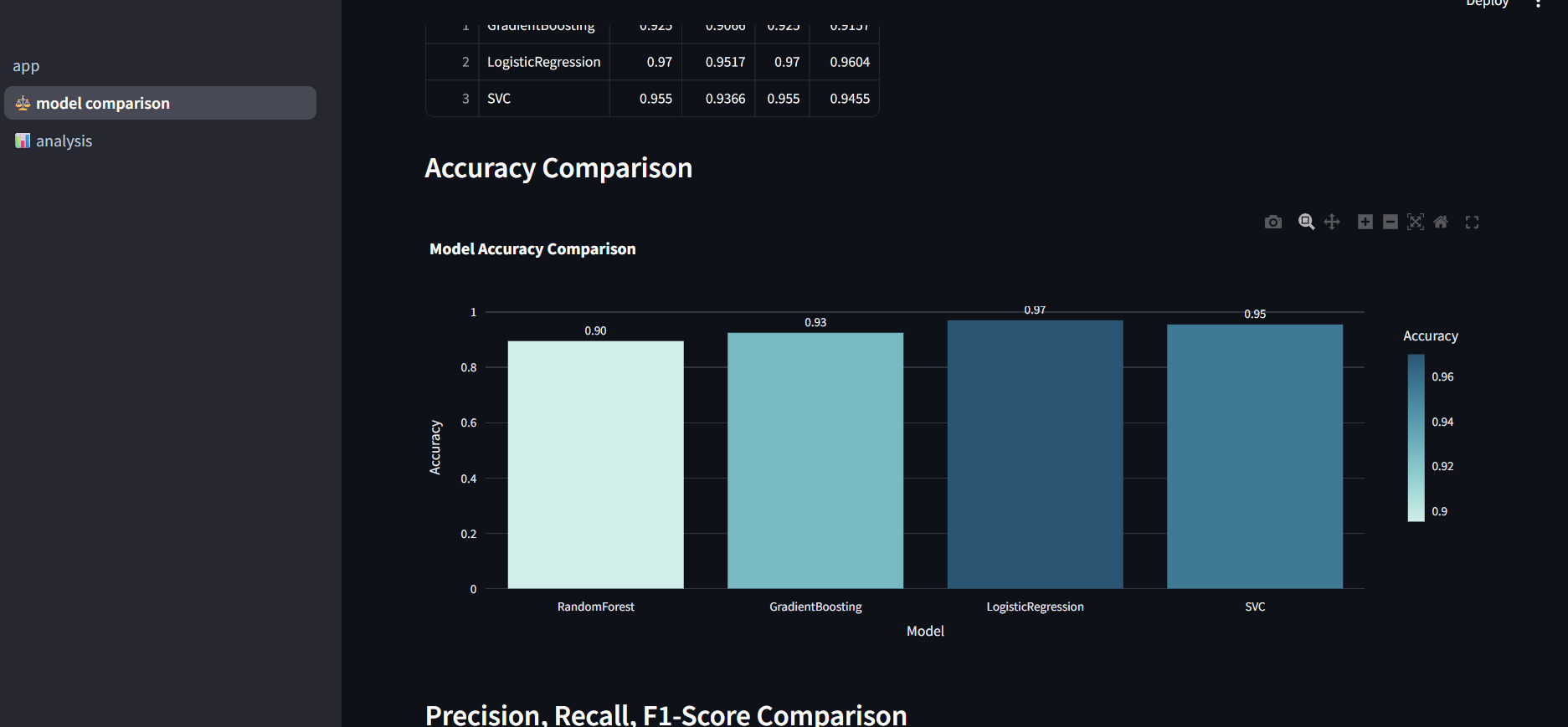
****

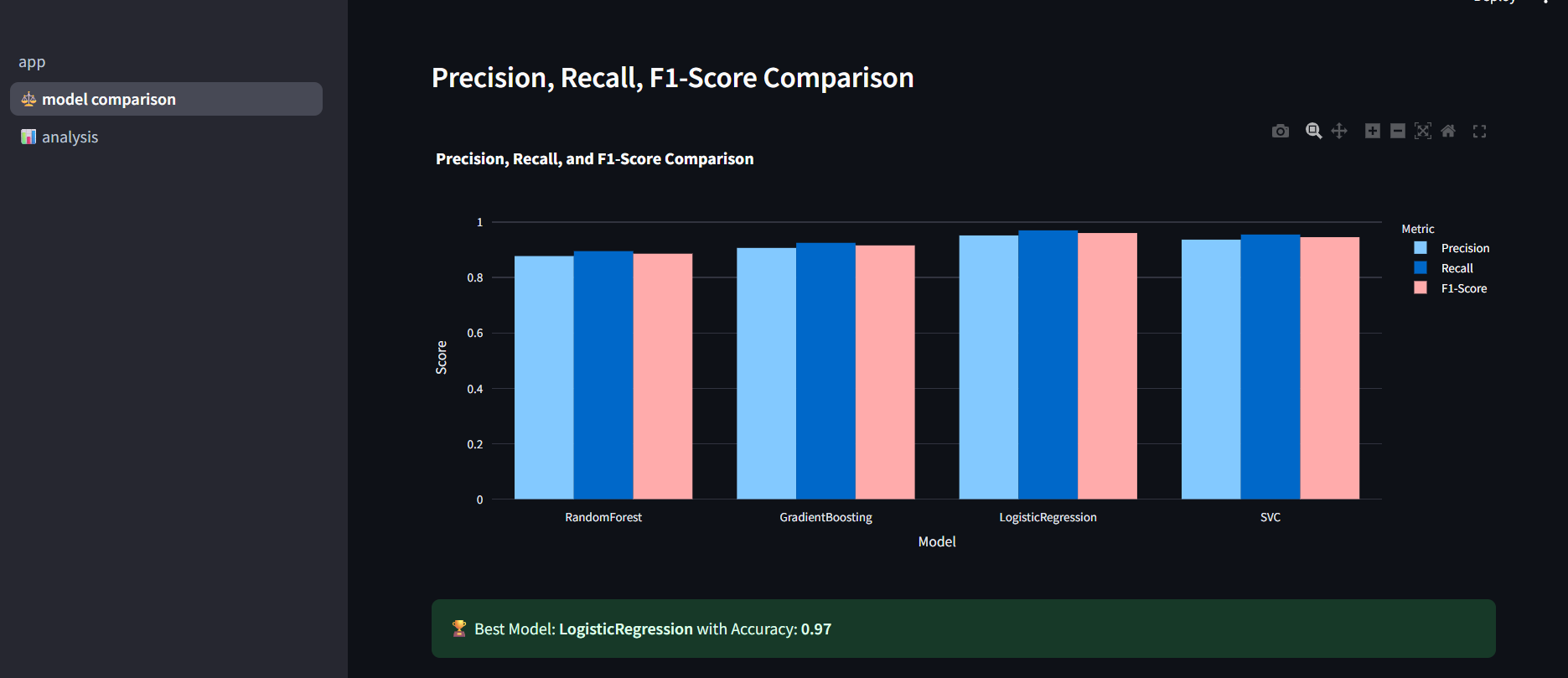
****

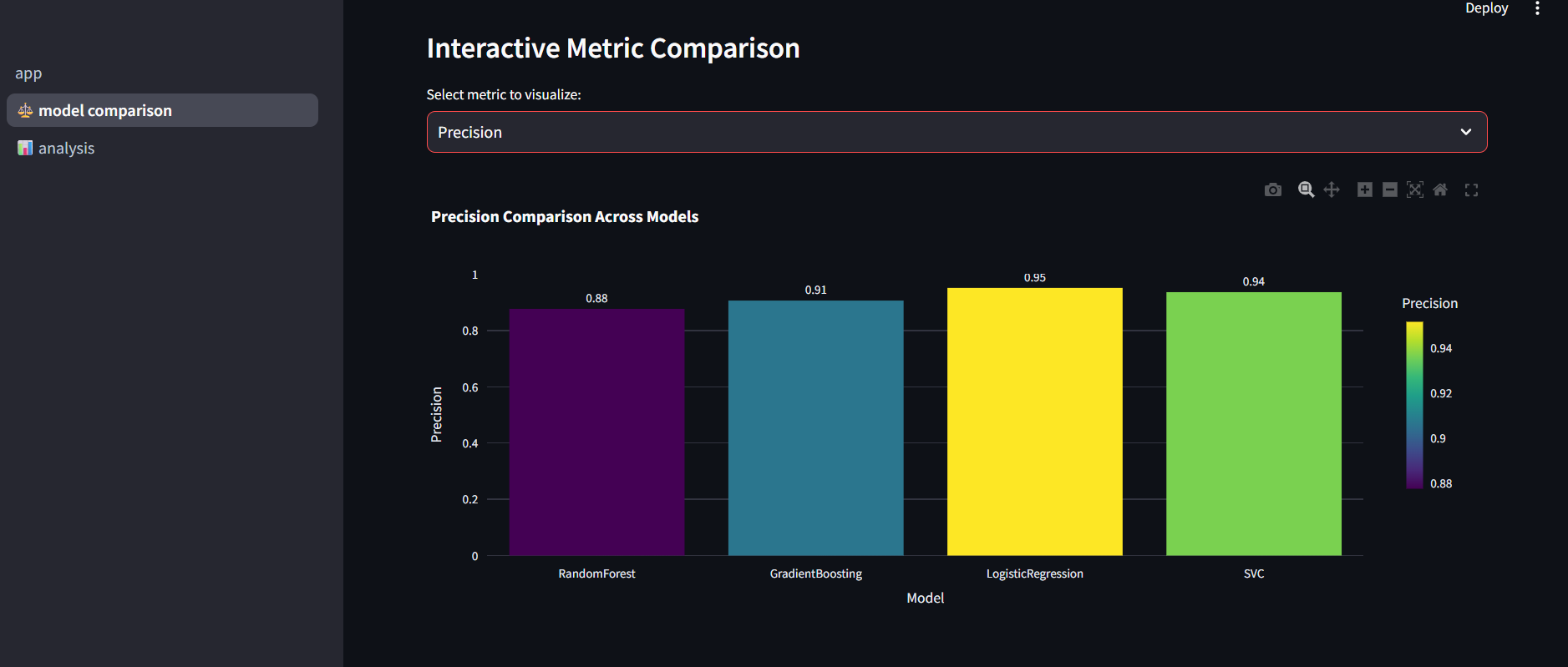
****

* 1. **Trained Model Comparisons**

****

****

****

****

**Extra Feature added: Voice Call Data Extraction**

To automate insights from sales calls, this feature converts call recordings into structured data. Key fields extracted include:

* Name, Age, Existing Customer – Identified via speech-to-text and NLP.
* Short Summary – Concise overview of the conversation.
* Sentiment – Positive, neutral, or negative sentiment of the call.
* Additional Info – Insurance\_Type.

Implementation:

* Speech-to-Text: Convert audio to text using tools like Whisper or Google Speech-to-Text.
* NLP & GenAI: Perform entity extraction, sentiment analysis, and summary generation.
* Data Structuring: Save extracted information into a CSV or database for easy integration.

Benefits:

* Captures insights automatically from calls.
* Enhances customer dataset for tier classification.
* Enables targeted marketing and better follow-ups.