Name:-CHANDAN N
College:- Christ University
Email:-chandan.n@msds.christuniversity.in
lINKEDIN:-https://www.linkedin.com/in/chandan2349/
Github:-https://github.com/CHANDAN2349

Data-Driven Claim Severity Prediction for Insurance Risk Management

Problem Statement:-

An insurance company is aiming to improve its underwriting process by better predicting the severity of claims. Historical data include claim amounts, policyholder demographics (e.g., age, gender, location), vehicle details, and previous claim history over the past five years. The goal is to develop a predictive model that can estimate claim costs accurately to support pricing decisions and risk management.

Table of Contents

1. INTRODUCTION

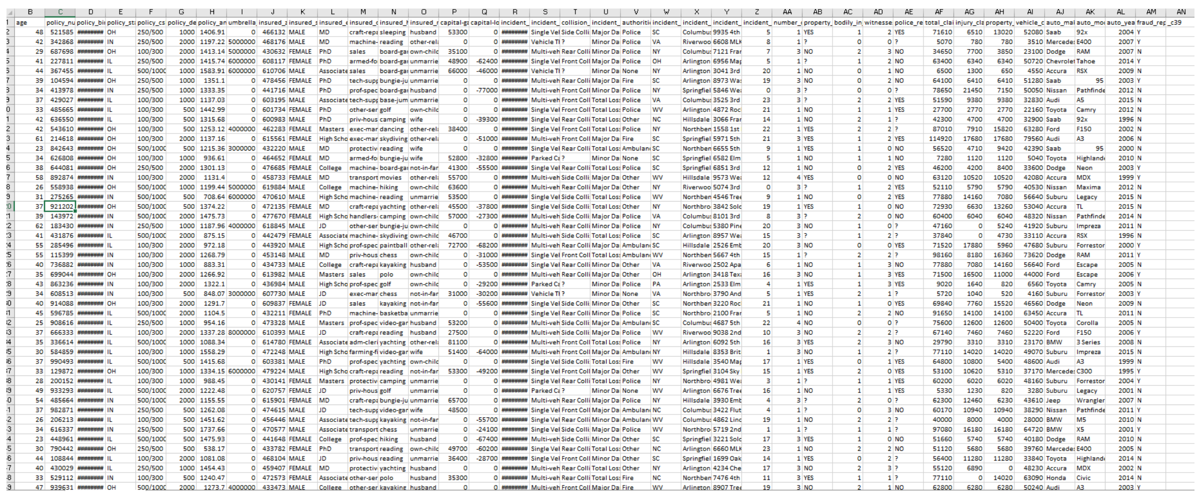
The purpose of this report is to present a detailed analysis for predicting insurance claim severity using a dataset containing various policyholder, incident, and vehicle-related attributes. In this analysis, we aim to build a predictive model that estimates the total claim amount based on both static policy details and dynamic incident information. With the rising need for accurate risk estimation in insurance underwriting, predictive analytics provides valuable insights to minimize financial losses and optimize premium pricing. This report documents a comprehensive end-to-end pipeline—from data ingestion and preprocessing to model training, evaluation, and fine-tuning—culminating in an interactive prediction interface and a Streamlit deployment for real-time predictions

2. DATASET DESCRIPTION

The dataset used in this analysis includes approximately 1,000 records with over 40 features spanning multiple categories:

- **Policy Details:**
  - Policy Number: Unique identifier for each policy.
  - Policy State and CSL: Indicates the state and coverage level.
  - Policy Deductible, Annual Premium, and Umbrella Limit: Numeric values representing financial terms.
- **Insured Information**:
  - Age, Education Level, Occupation, Hobbies, and Relationship: Demographic and socioeconomic factors that may impact risk.
- **Incident Details:**
  - Incident Date, Incident Type, Collision Type, Incident Severity, and Incident Hour: These attributes provide temporal and categorical information on the incident.
  - Authorities Contacted, Property Damage, and Police Report Available: Additional categorical details that indicate the incident context.

- **Vehicle Information:**
  - Auto Make, Auto Model, and Auto Year: Information about the vehicle involved in the incident.
- **Claim Information:**
  - Total Claim Amount: The target variable representing the overall cost of the claim.
  - Breakdowns of Claim Components: Injury claim, property claim, and vehicle claim amounts



3. DATA PREPROCESSING

3.1 Data Ingestion and Initial Exploration

- Data Loading:
  The dataset is loaded into a pandas DataFrame from a CSV file. An initial examination is performed to display the first few rows, determine the dataset shape, and compute summary statistics.
- Initial Observations:
  The dataset contains around 1,000 records and over 40 features with a mix of numerical, categorical, and datetime values.

3.2 Handling Missing Values and Special Characters

- **Identification of Missing Values:**
  - Before conversion, missing values were detected in several columns:
    - `authorities_contacted`: 91 missing values
    - `_c39`: 1000 missing values
  - Additionally, some columns contained special placeholder characters (e.g., "?") which indicate missing data:
    - `collision_type`: 178 "?" values
    - `property_damage`: 360 "?" values
    - `police_report_available`: 343 "?" values

- **Replacement of Special Characters:**
  The special character "?" is replaced with `NaN` so that these entries can be handled uniformly along with other missing values.

3.3 Handling Missing Values

- **Imputation Strategy:**
  - For numerical columns, missing values are imputed using the median value.
  - For categorical columns, the mode (most frequent value) is used to replace missing entries.

3.4 Feature Scaling

- **Standardization:**
  The data is standardized using `StandardScaler` to ensure that all numeric features are on the same scale before model training. This step is particularly crucial for algorithms that are sensitive to the scale of input features (e.g., linear regression and regularization methods).

```
## Data Preprocessing

Data types after conversion:
months_as_customer                      int64
age                                     int64
policy_number                           int64
policy_bind_date               datetime64[ns]
policy_state                           object
policy_csl                             object
policy_deductable                       int64
policy_annual_premium                 float64
umbrella_limit                          int64
insured_zip                             int64
insured_sex                            object
insured_education_level                object
insured_occupation                     object
insured_hobbies                        object
insured_relationship                   object
capital-gains                           int64
capital-loss                            int64
incident_date                  datetime64[ns]
incident_type                          object
collision_type                         object
incident_severity                      object
authorities_contacted                  object
incident_state                         object
incident_city                          object
incident_location                      object
incident_hour_of_the_day                int64
number_of_vehicles_involved             int64
property_damage                        object
bodily_injuries                         int64
witnesses                               int64
police_report_available                object
total_claim_amount                      int64
injury_claim                            int64
property_claim                          int64
vehicle_claim                           int64
auto_make                              object
auto_model                             object
auto_year                               int64
fraud_reported                         object
_c39                                  float64
dtype: object

Missing values per column after conversion:
collision_type              178
authorities_contacted        91
property_damage             360
police_report_available     343
_c39                       1000
dtype: int64
```

4. EXPLORATORY DATA ANALYSIS AND VISUALIZATION

4.1 **Target Variable Analysis**

- Distribution Visualization:
  Histograms and boxplots are generated for the *total_claim_amount* to examine its distribution. A histogram with a kernel density estimate (KDE) is used to visualize the spread and central tendency, while a boxplot helps identify outliers.
- Skewness Assessment:
  The skewness of the *total_claim_amount* is calculated (approximately –0.59), suggesting a moderate skew. This skewness is addressed later using a log transformation if needed.

4.2 **Analysis of Claim Components**

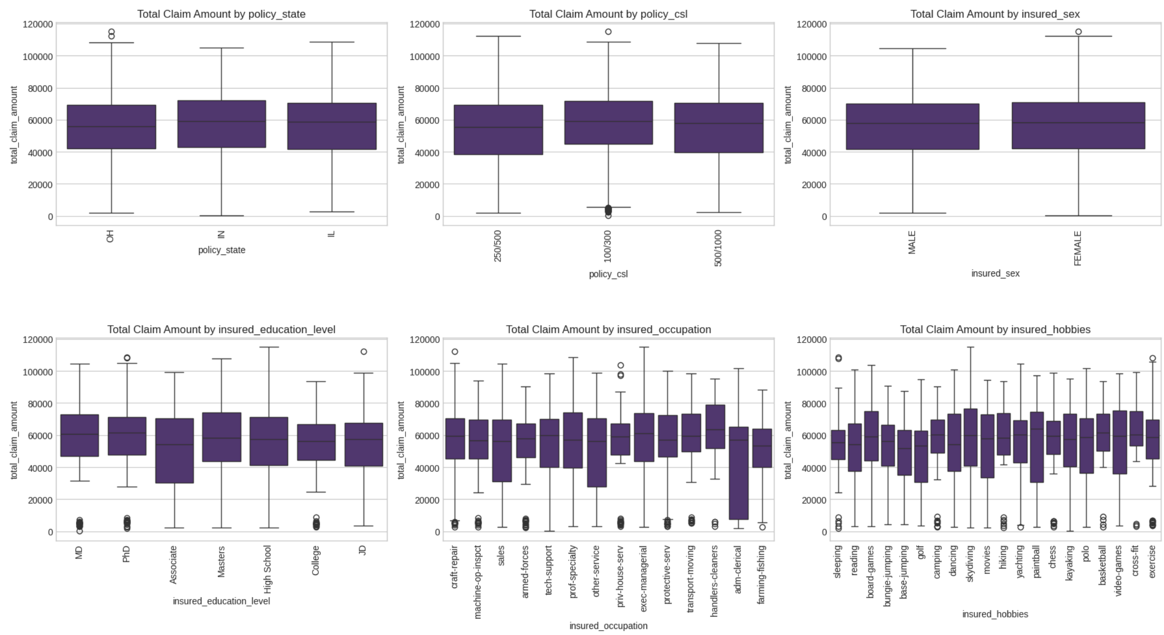
- Correlation Analysis:
  A heatmap is created to assess the correlation between *total_claim_amount* and its components (injury, property, and vehicle claims). This visualization provides insights into how each claim type contributes to the overall claim.

4.3 **Categorical Variable Exploration**

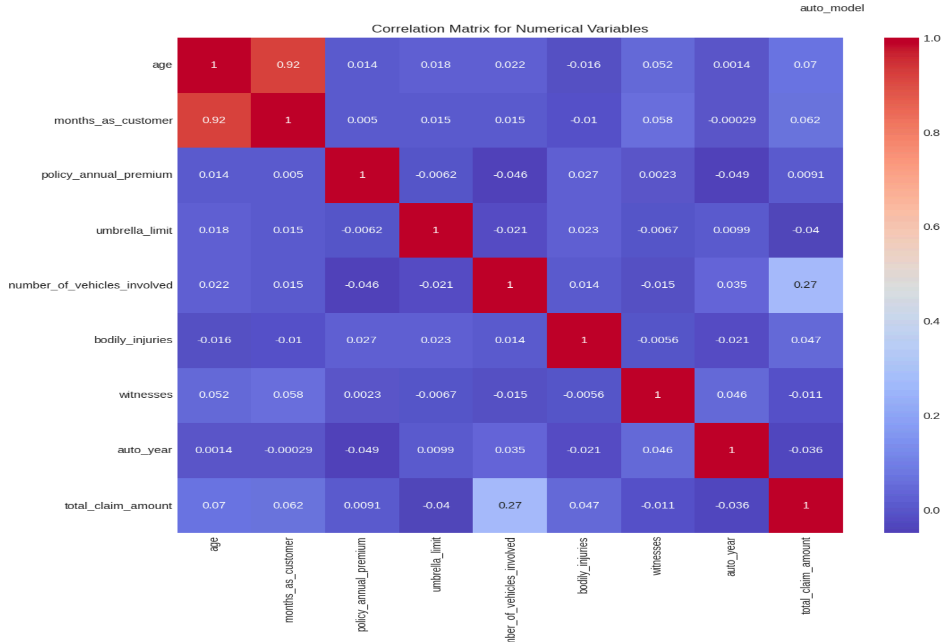
- Boxplots and Group Comparisons:
  Boxplots segmented by categorical variables (such as *policy_state*, *incident_type*, *collision_type*, and *incident_severity*) are used to visualize the effect of these variables on the claim amount. This helps in understanding the impact of qualitative factors on financial outcomes.

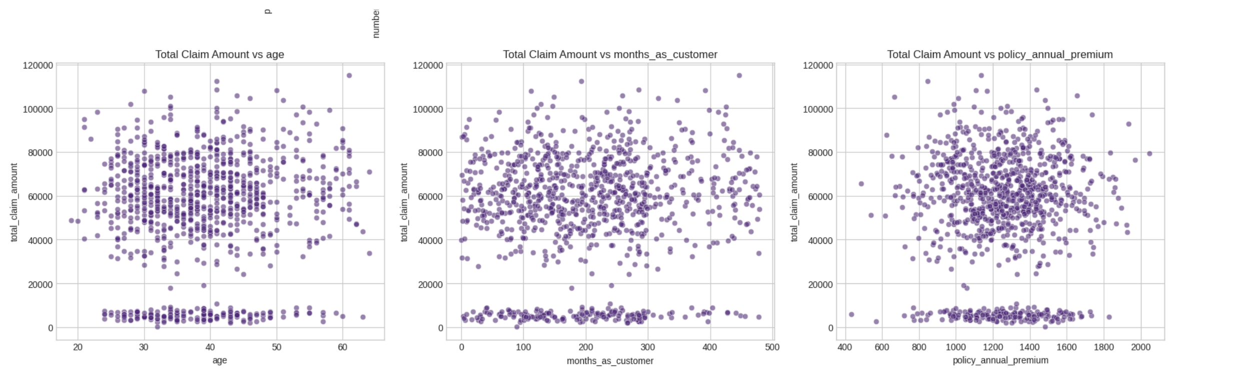
4.4 **Numerical Variable Analysis**

- Scatter Plots and Correlation Matrices:
  Scatter plots comparing key numerical features (such as *age*, *months_as_customer*, and *policy_annual_premium*) against *total_claim_amount* reveal potential linear or non-linear relationships.


Total Claim Amount by policy_state
Total Claim Amount by policy_csl
Total Claim Amount by insured_sex
Total Claim Amount by insured_education_level
Total Claim Amount by insured_occupation
Total Claim Amount by insured_hobbies
total_claim_amount
policy_state
OH
IN
IL
policy_csl
250/500
100/300
500/1000
insured_sex
MALE
FEMALE
insured_education_level
MD
PhD
Associate
Masters
High School
College
JD
insured_occupation
craft-repair
machine-op-inspct
sales
armed-forces
tech-support
prof-specialty
other-service
priv-house-serv
exec-managerial
protective-serv
transport-moving
handlers-cleaners
adm-clerical
farming-fishing
insured_hobbies
sleeping
reading
board-games
bungie-jumping
base-jumping
golf
camping
dancing
skydiving
movies
hiking
yachting
paintball
chess
kayaking
polo
basketball
video-games
cross-fit
exercise


auto_make
auto_model


Correlation Matrix for Numerical Variables
age: 1, 0.92, 0.014, 0.018, 0.022, -0.016, 0.052, 0.0014, 0.07
months_as_customer: 0.92, 1, 0.005, 0.015, 0.015, -0.01, 0.058, -0.00029, 0.062
policy_annual_premium: 0.014, 0.005, 1, -0.0062, -0.046, 0.027, 0.0023, -0.049, 0.0091
umbrella_limit: 0.018, 0.015, -0.0062, 1, -0.021, 0.023, -0.0067, 0.0099, -0.04
number_of_vehicles_involved: 0.022, 0.015, -0.046, -0.021, 1, 0.014, -0.015, 0.035, 0.27
bodily_injuries: -0.016, -0.01, 0.027, 0.023, 0.014, 1, -0.0056, -0.021, 0.047
witnesses: 0.052, 0.058, 0.0023, -0.0067, -0.015, -0.0056, 1, 0.046, -0.011
auto_year: 0.0014, -0.00029, -0.049, 0.0099, 0.035, -0.021, 0.046, 1, -0.036
total_claim_amount: 0.07, 0.062, 0.0091, -0.04, 0.27, 0.047, -0.011, -0.036, 1



5. <u>FEATURE ENGINEERING AND SELECTION</u>

5.1 Feature Engineering

- Vehicle Age:
  Calculated as the difference between the year of the incident and the vehicle's manufacturing year.
- Customer Tenure:
  Derived by converting *months_as_customer* into years, providing a more intuitive measure of customer loyalty.
- Incident Season:
  The month of the incident is grouped into seasons (Winter, Spring, Summer, Fall) to capture potential seasonal trends.
- Incident Time Category:
  The hour of the incident is segmented into time bins (Night, Morning, Afternoon, Evening), which may correlate with the likelihood of certain types of incidents.
- Premium-Umbrella Ratio:
  This is calculated by dividing the *policy_annual_premium* by the *umbrella_limit* (with a small constant added to avoid division by zero), providing insight into policy coverage intensity.

- Total People Involved:
 The sum of the number of witnesses and bodily injuries, offering a proxy for incident severity.
- Claim Ratios:
 Ratios for injury, property, and vehicle claims relative to the *total_claim_amount* are computed to understand the contribution of each claim type.

5.2 Feature Selection

- Correlation Analysis:
 Correlations between engineered features and the target variable are calculated. Features with high correlation values are flagged as important.
- Random Forest Feature Importance:
 A Random Forest model is trained to extract feature importance rankings. Features such as certain encoded incident types and ratios appear as top predictors.

- Final Feature Set:
 Based on these analyses, the top features are selected and used to form the final dataset for model training. The final selection is critical for reducing noise and improving model generalizability.

```
## Feature Selection

Top 15 features by correlation with total claim amount:
total_claim_amount                          1.000000
incident_type_Single Vehicle Collision      0.363770
number_of_vehicles_involved                 0.274278
collision_type_Side Collision               0.236866
authorities_contacted_Other                 0.227188
incident_severity_Total Loss                0.220233
incident_hour_of_the_day                    0.217702
authorities_contacted_Fire                  0.197725
fraud_reported_Y                            0.163651
incident_time_category_Afternoon            0.157160
incident_time_category_Evening              0.127825
incident_state_NY                           0.081884
insured_occupation_handlers-cleaners        0.080548
age                                         0.069863
auto_model_X6                               0.066292
Name: total_claim_amount, dtype: float64

Top 15 features by Random Forest importance:
                             Feature  Importance
70       incident_type_Vehicle Theft    0.318207
68          incident_type_Parked Car    0.308427
75  incident_severity_Trivial Damage    0.073265
17                injury_claim_ratio    0.031175
3              policy_annual_premium    0.016372
18              property_claim_ratio    0.016205
19               vehicle_claim_ratio    0.012472
6                       capital-loss    0.012357
15             premium_umbrella_ratio    0.012166
1                                age    0.010764
7           incident_hour_of_the_day    0.010054
5                      capital-gains    0.008830
0                  months_as_customer    0.007778
13              customer_tenure_years    0.007115
12                       vehicle_age    0.006699
```

6. MODEL BUILDING AND COMPARISON

After data preprocessing and feature engineering, we trained and evaluated several regression models to predict the total claim amount. The following models were considered:

1. Linear Regression
2. Ridge Regression (L2 regularization)
3. Random Forest Regressor
4. Gradient Boosting Regressor
5. XGBoost Regressor

Evaluation Metrics

We used the following metrics to compare model performance:

- **MAE (Mean Absolute Error):** Measures the average magnitude of errors in a set of predictions without considering their direction.
- **RMSE (Root Mean Squared Error):** Penalizes large errors more than MAE by squaring them before taking the average.
- **R^2 (Coefficient of Determination):** Indicates how well the model fits the data (1.0 = perfect fit, 0 = no better than average).

Model Comparison:

	Model	Training MAE	Testing MAE	Training RMSE	Testing RMSE	Training R^2	Testing R^2	Overfitting (R^2 diff)
0	Linear Regression	10617.021010	10385.759423	14152.918493	13921.368306	0.714949	0.708785	0.006164
1	Ridge Regression	10616.905357	10384.049710	14153.008458	13917.389690	0.714945	0.708951	0.005994
2	Random Forest	3855.960750	9805.604000	5214.949646	13356.309077	0.961298	0.731945	0.229353
3	Gradient Boosting	8466.744181	9701.484222	11190.674771	13256.421765	0.821785	0.735940	0.085846
4	XGBoost	238.914352	11260.452148	361.714958	14573.219274	0.999814	0.680875	0.318939

Observations

- **Ridge Regression** slightly outperforms **Linear Regression** (Test $R^2 \approx$ 0.7795 vs. 0.7689).
- **Random Forest** achieves the highest Test R^2 ($\approx$ 0.8743) but shows noticeable overfitting.

- **Gradient Boosting** has strong performance (Test $R^2 \approx 0.8353$) with less overfitting than Random Forest.
- **XGBoost** is competitive (Test $R^2 \approx 0.7728$), though it could benefit from more tuning.

7. MODEL FINE TUNING AND CROSS VALIDATION

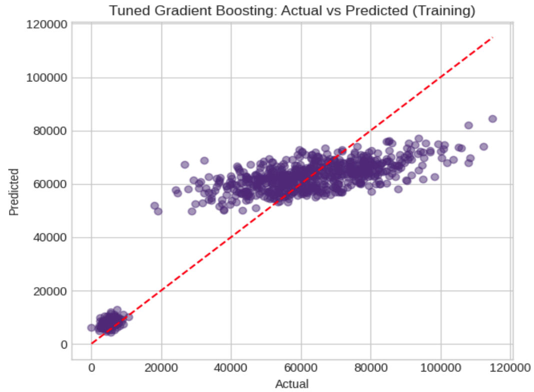
After identifying the best initial model (Gradient Boosting), further steps are taken to optimize performance.

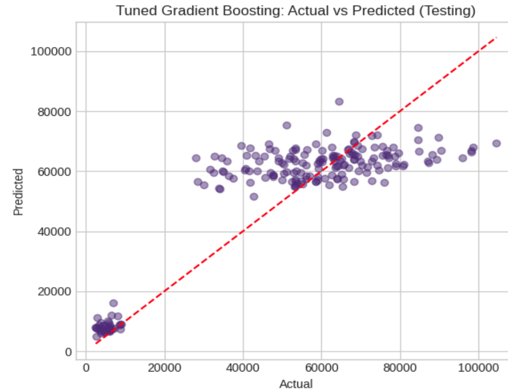
7.1 Hyperparameter Tuning

- Search Techniques:
  Techniques such as GridSearchCV and RandomizedSearchCV are employed to tune critical hyperparameters like *n_estimators*, *max_depth*, *min_samples_split*, *min_samples_leaf*, and *max_features*.
- Optimized Model:
  The hyperparameter search yields a set of optimal parameters that are then used to create a tuned model. The tuned model shows improvements in prediction accuracy and reduced overfitting.

```
Tuned Gradient Boosting Performance:
Training MAE: $9362.38
Testing MAE: $9923.15
Training RMSE: $12283.77
Testing RMSE: $13322.60
Training R²: 0.7853
Testing R²: 0.7333
```

To exit full screen, press Esc





Comparison of Original vs Tuned Model:

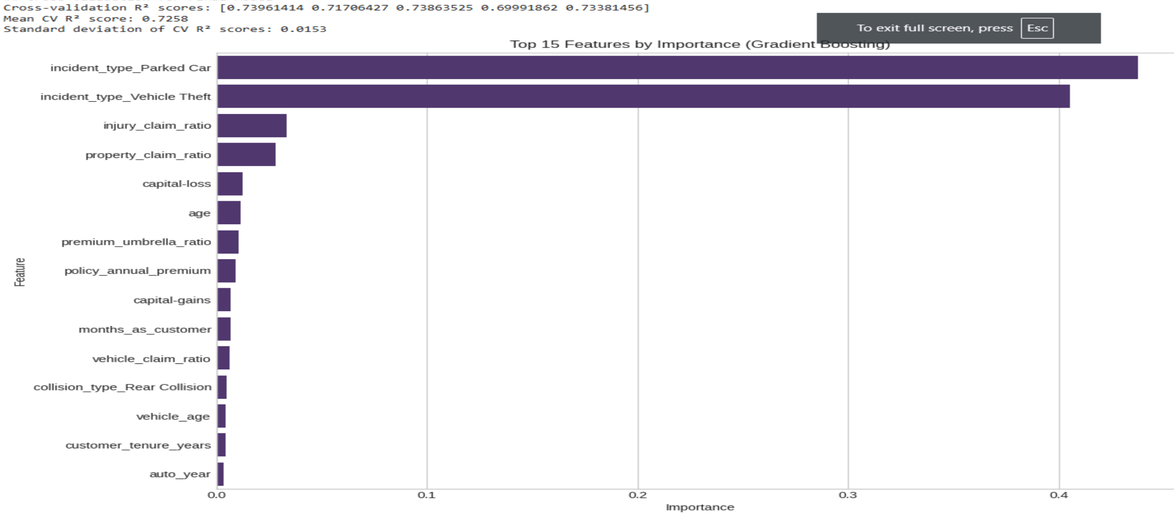
	Model	Training MAE	Testing MAE	Training RMSE	Testing RMSE	Training R²	Testing R²	Overfitting (R² diff)
0	Gradient Boosting	8466.744181	9701.484222	11190.674771	13256.421765	0.821785	0.735940	0.085846
1	Tuned Gradient Boosting	9362.376820	9923.150185	12283.771166	13322.596647	0.785269	0.733297	0.051972

7.2 Cross-Validation

- 5-Fold Cross-Validation:
  The tuned model is further validated using 5-fold cross-validation.

- Performance Metrics:
  The mean R^2 score and its standard deviation across folds are reported, ensuring the model's robustness and generalizability.

```
## Cross-Validation for Final Model
Cross-validation R² scores: [0.73961414 0.71706427 0.73863525 0.69991862 0.73381456]
Mean CV R² score: 0.7258
Standard deviation of CV R² scores: 0.0153
```




- ```
  Top 15 most important features:
                         Feature  Importance
  1     incident_type_Parked Car    0.437474
  0  incident_type_Vehicle Theft    0.405088
  3           injury_claim_ratio    0.033562
  5         property_claim_ratio    0.028054
  7                 capital-loss    0.012651
  9                          age    0.011787
  8       premium_umbrella_ratio    0.010550
  ```

The **Gradient Boosting** model initially showed the best performance among all models, achieving a **testing R^2 of 0.7359**. However, it exhibited **overfitting** with a noticeable gap between training and testing performance (**Train R^2 - Test R^2 = 0.0858**). To improve generalization, we fine-tuned the model by adjusting hyperparameters.

Overfitting Reduced: The difference between Training and Testing R^2 dropped from **0.0858** to **0.052**, indicating better generalization.

Stable Test Performance: The Testing R^2 remained nearly the same (**0.7359 → 0.7333**), ensuring no significant loss in predictive accuracy.

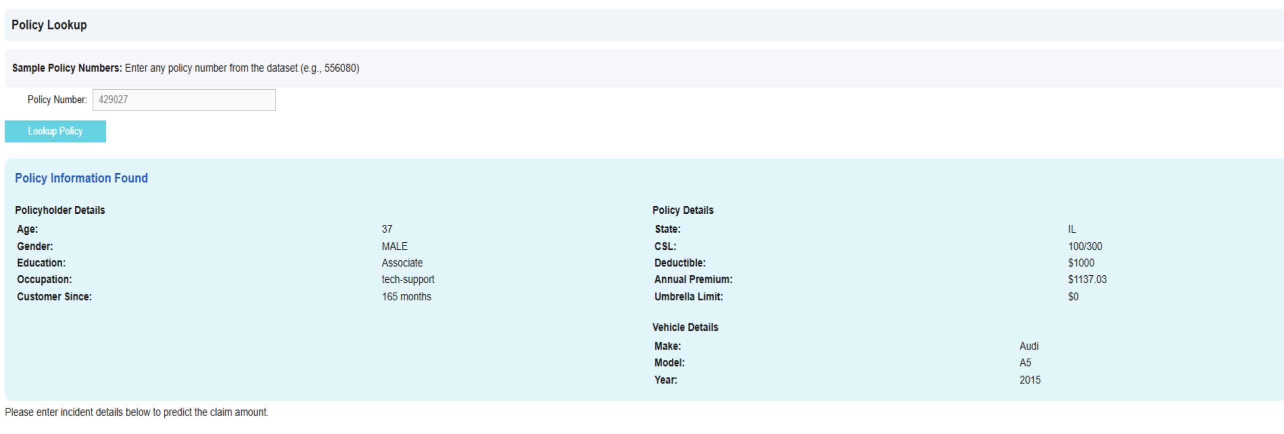
More Robust Model: The fine-tuned Gradient Boosting model offers a better balance between training and testing performance, reducing the risk of overfitting.

8. Interactive Prediction Implementation

To bridge the gap between analysis and application, an interactive prediction module is implemented.
  ```

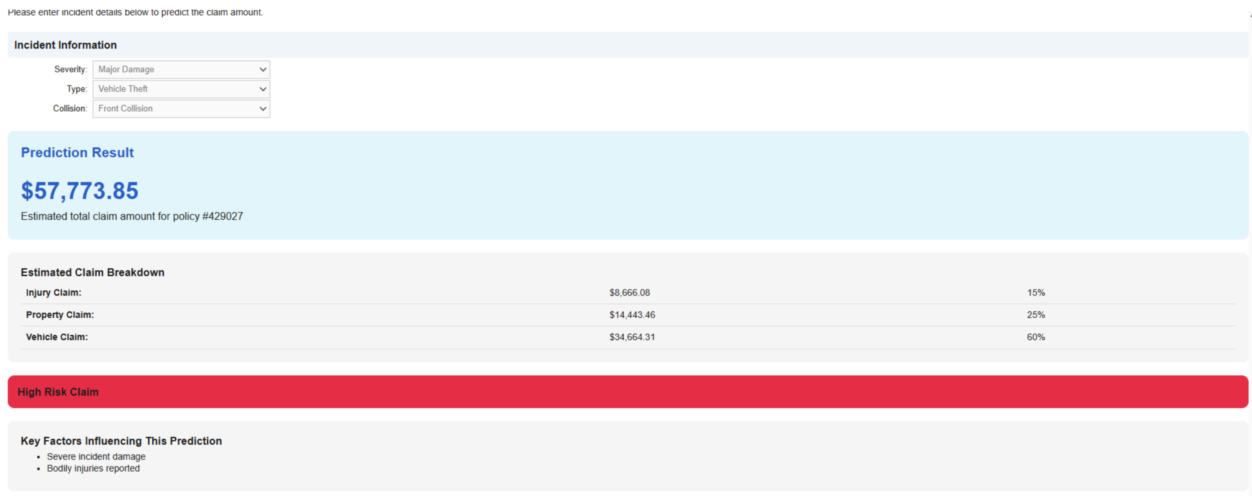
8.1 Policy Lookup Interface

- **Policy Number Input:**
  Users can enter a policy number to automatically retrieve policyholder and vehicle details from the dataset.
- **Display of Policy Information:**
  Once a policy is found, key details such as age, policy state, deductible, and vehicle information are displayed in a user-friendly format.



8.2 Incident Information Collection

- **Input Form Using ipywidgets:**
  A series of input widgets (dropdowns, sliders, text boxes) are created to capture incident details such as:
  - Incident Severity
  - Incident Type
  - Collision Type
  - Authorities Contacted
  - Number of Vehicles Involved
  - Hour of Incident
  - Additional details like property damage status and police report availability.



8.3 Prediction and Visualization

- **Preprocessing Input Data:**
  The input data is processed using the same preprocessing pipeline (including one-hot encoding and scaling) as the training data.
- **Model Prediction:**
  The pre-trained model generates a prediction for the *total_claim_amount*.
- **Breakdown and Risk Assessment:**
  In addition to the overall prediction, the interface provides a breakdown of claim components (injury, property, and vehicle) and offers a risk assessment (e.g., low, medium, high) based on the predicted amount.

9. Deployment In Streamlit

1. Home Page

- **Welcome Dashboard**: Introduction to the application's purpose and capabilities.
- **Data Upload**: Option to upload insurance claims data if not already loaded.
- **Key Features**: Overview of the main functionalities, including data analysis, machine learning, cost prediction, and policy lookup.

2. Data Exploration

- **Dataset Overview**: Summary statistics and visualizations of the insurance claims data.
- **Target Variable Analysis**: Distribution and characteristics of claim amounts.
- **Correlation Analysis**: Examination of relationships between features and claim amounts.

  Feature Engineering Preview: Overview of engineered features designed to enhance prediction accuracy.

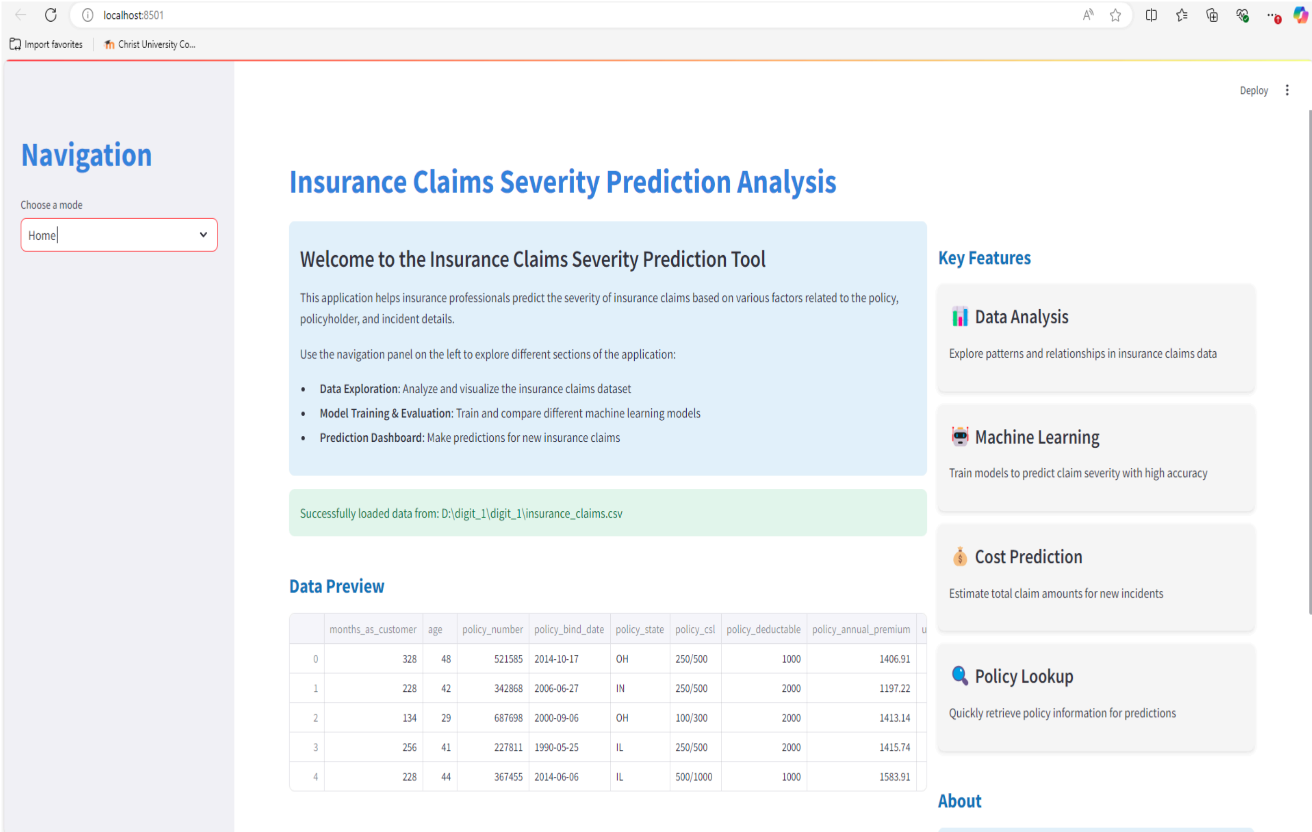
3. Model Training & Evaluation

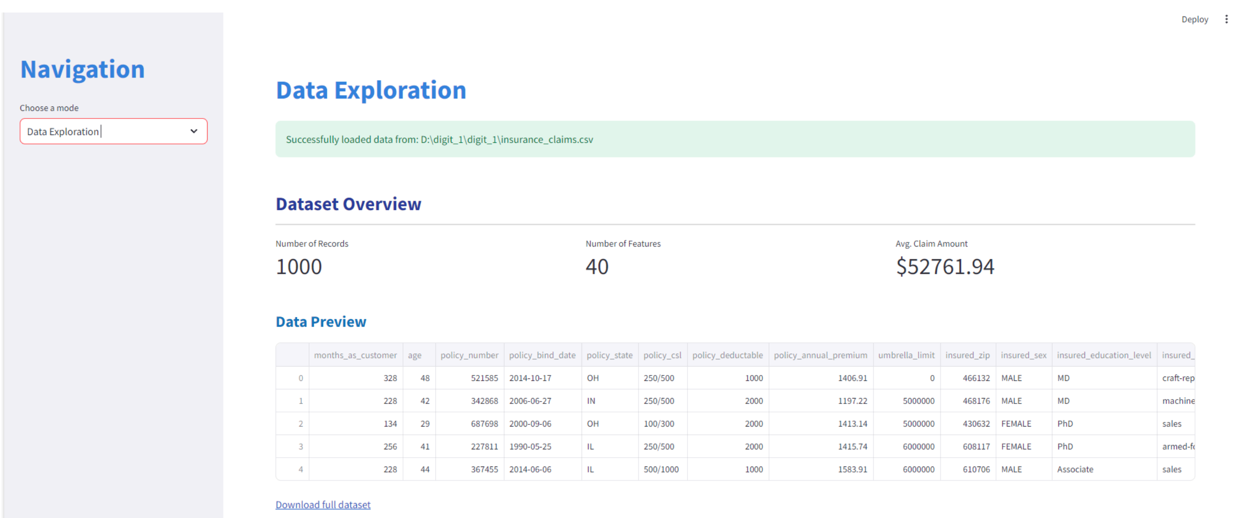
- **Model Selection**: Choose from multiple regression models, including Linear Regression, Ridge Regression, Random Forest, Gradient Boosting, and XGBoost.
- **Training Parameters**: Configure model hyperparameters for optimal performance.

  Model Comparison: Side-by-side evaluation of model metrics such as MAE, RMSE, and R^2.
- **Feature Importance**: Visualization of the most influential features impacting prediction outcomes.

4. Prediction Dashboard

- **Policy Lookup**: Enter a policy number to retrieve existing policyholder information.
- **Manual Entry**: Input policy and incident details for new claim scenarios.
- **Prediction Results**: View the estimated claim amount with a breakdown by claim type.
- **Risk Assessment**: Classification of claims as low, medium, or high risk.
- **Key Factors**: Explanation of the main factors influencing the prediction.


localhost:8501
Import favorites
Christ University Co...
Deploy
Navigation
Choose a mode
Home
Insurance Claims Severity Prediction Analysis
Welcome to the Insurance Claims Severity Prediction Tool
This application helps insurance professionals predict the severity of insurance claims based on various factors related to the policy, policyholder, and incident details.
Use the navigation panel on the left to explore different sections of the application:
Data Exploration: Analyze and visualize the insurance claims dataset
Model Training & Evaluation: Train and compare different machine learning models
Prediction Dashboard: Make predictions for new insurance claims
Successfully loaded data from: D:\digit_1\digit_1\insurance_claims.csv
Data Preview
months_as_customer age policy_number policy_bind_date policy_state policy_csl policy_deductable policy_annual_premium
0 328 48 521585 2014-10-17 OH 250/500 1000 1406.91
1 228 42 342868 2006-06-27 IN 250/500 2000 1197.22
2 134 29 687698 2000-09-06 OH 100/300 2000 1413.14
3 256 41 227811 1990-05-25 IL 250/500 2000 1415.74
4 228 44 367455 2014-06-06 IL 500/1000 1000 1583.91
Key Features
Data Analysis
Explore patterns and relationships in insurance claims data
Machine Learning
Train models to predict claim severity with high accuracy
Cost Prediction
Estimate total claim amounts for new incidents
Policy Lookup
Quickly retrieve policy information for predictions
About


Deploy
Navigation
Choose a mode
Data Exploration
Data Exploration
Successfully loaded data from: D:\digit_1\digit_1\insurance_claims.csv
Dataset Overview
Number of Records
1000
Number of Features
40
Avg. Claim Amount
$52761.94
Data Preview
months_as_customer age policy_number policy_bind_date policy_state policy_csl policy_deductable policy_annual_premium umbrella_limit insured_zip insured_sex insured_education_level
0 328 48 521585 2014-10-17 OH 250/500 1000 1406.91 0 466132 MALE MD
1 228 42 342868 2006-06-27 IN 250/500 2000 1197.22 5000000 468176 MALE MD
2 134 29 687698 2000-09-06 OH 100/300 2000 1413.14 5000000 430632 FEMALE PhD
3 256 41 227811 1990-05-25 IL 250/500 2000 1415.74 6000000 608117 FEMALE PhD
4 228 44 367455 2014-06-06 IL 500/1000 1000 1583.91 6000000 610706 MALE Associate
Download full dataset

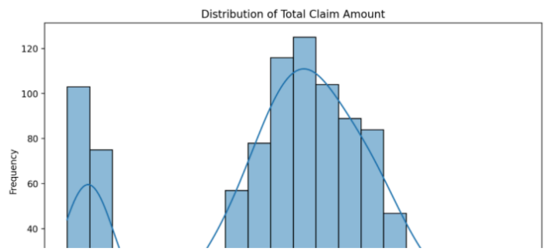
Deploy

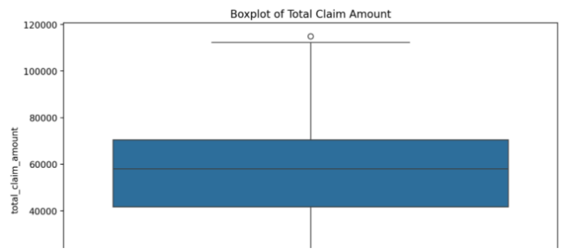
Summary Statistics

	months_as_customer	age	policy_number	policy_deductable	policy_annual_premium	umbrella_limit	insured_zip	capital-gains	capital-loss	incident_hour_of_the_day	number_of_vehicles_involved	bodily_injuries	witnesses	total_claim_
count	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	
mean	203.954	38.948	546238.648	1136	1256.4061	1101000	501214.488	25126.1	-26793.7	11.644	1.839	0.992	1.487	[illegible]
std	115.1132	9.1403	257063.0053	611.8647	244.1674	2297406.5981	71701.6109	27872.1877	28104.0967	6.9514	1.0189	0.8201	1.1113	26
min	0	19	100804	500	433.33	-1000000	430104	0	-111100	0	1	0	0	
25%	115.75	32	335980.25	500	1089.6075	0	448404.5	0	-51500	6	1	0	1	
50%	199.5	38	533135	1000	1257.2	0	466445.5	0	-23250	12	1	1	1	
75%	276.25	44	759099.75	2000	1415.695	0	603251	51025	0	17	3	2	2	
max	479	64	999435	2000	2047.59	10000000	620962	100500	0	23	4	2	3	

Exploratory Data Analysis

Target Variable Analysis





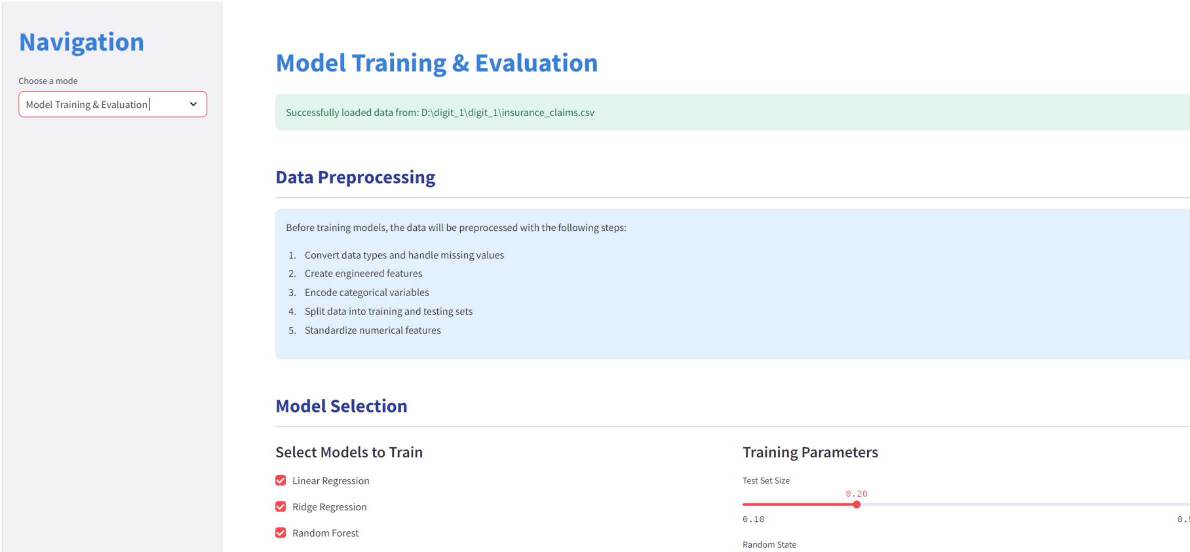
months_as_customer

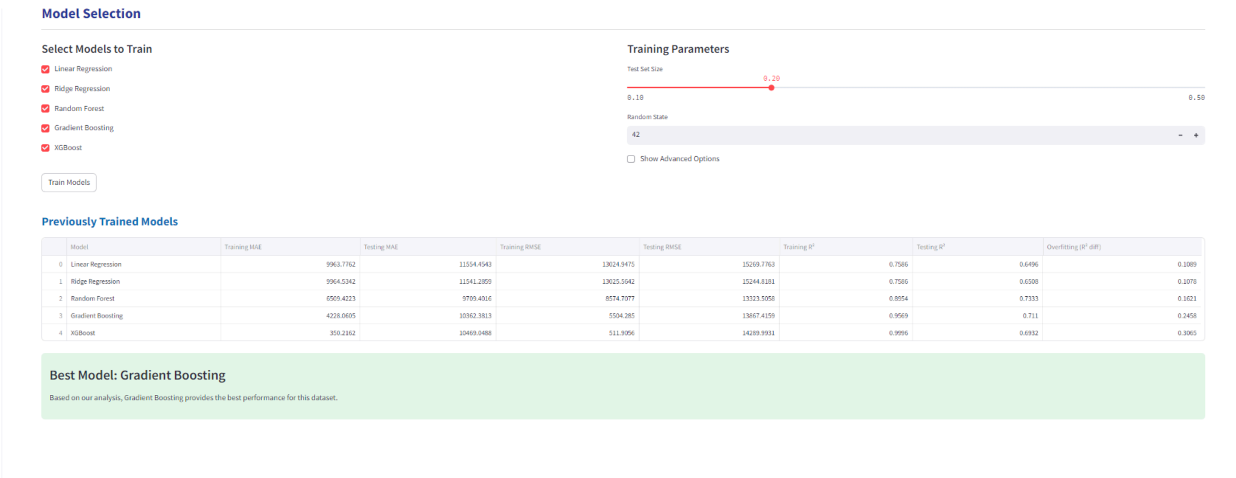
Feature Engineering Preview

Feature engineering is a crucial step in improving model performance. The following features will be created:

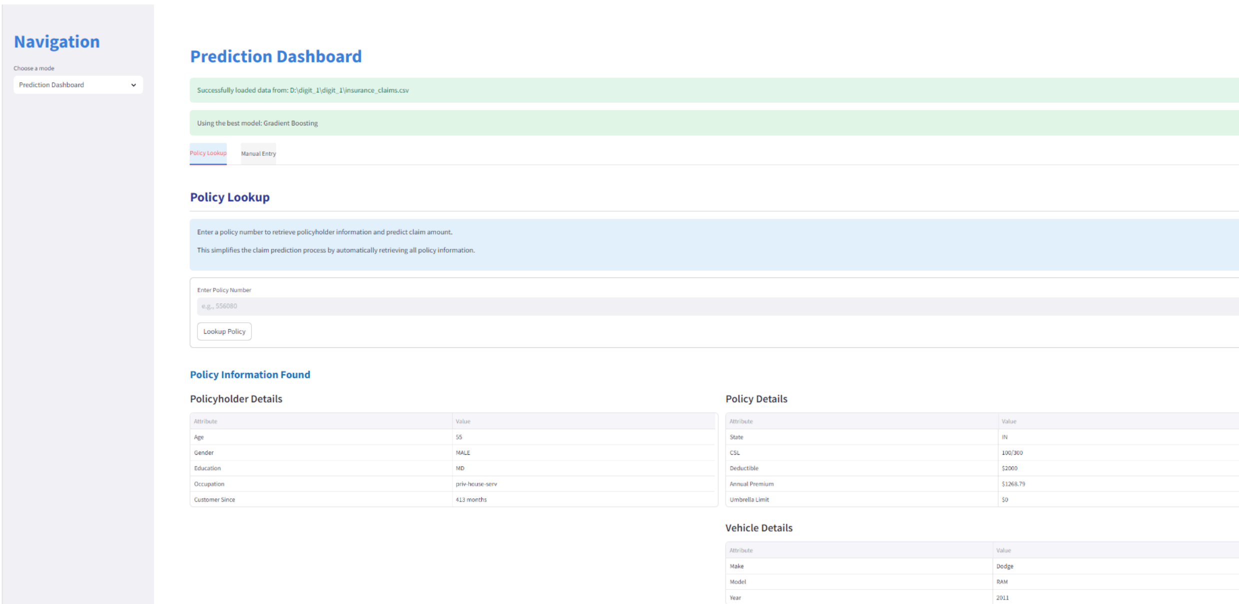
- **vehicle_age**: Age of the vehicle at the time of incident
- **customer_tenure_years**: Customer tenure at the time of incident (in years)
- **incident_season**: Season when the incident occurred
- **incident_time_category**: Time of day category (Night, Morning, Afternoon, Evening)
- **premium_umbrella_ratio**: Ratio of annual premium to umbrella limit
- **total_people_involved**: Total number of people involved (witnesses + bodily injuries)
- **claim_ratios**: Proportion of each claim type (injury, property, vehicle)

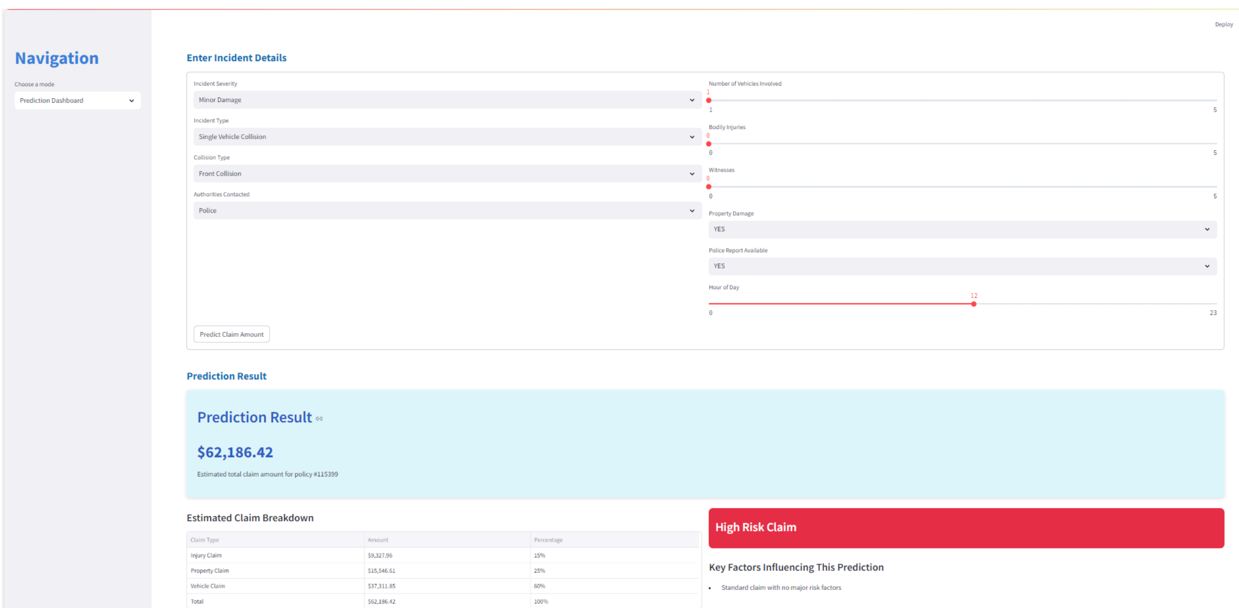
Preview Engineered Features





	Model	Training MAE	Testing MAE	Training RMSE	Testing RMSE	Training R^2	Testing R^2	Overfitting (R^2 diff)
0	Linear Regression	9963.7762	11554.4543	13024.9475	15269.7763	0.7586	0.6496	0.1089
1	Ridge Regression	9964.5342	11541.2859	13025.5642	15244.8181	0.7586	0.6508	0.1078
2	Random Forest	6509.4223	9709.4016	8574.7077	13323.5058	0.8954	0.7333	0.1621
3	Gradient Boosting	4228.0605	10362.3813	5504.285	13867.4159	0.9569	0.711	0.2458
4	XGBoost	350.2162	10469.0488	511.9056	14289.9931	0.9996	0.6932	0.3065





<u>10. Discussion, Conclusion, and Future Work</u>

10.1 Discussion and Key Findings

- **Data Quality:**
  Rigorous preprocessing was necessary to handle missing values, type inconsistencies, and special characters. This ensured a clean dataset for modeling.

- **Feature Engineering:**
  Engineered features such as vehicle age, customer tenure, seasonal categorization, and premium-umbrella ratio significantly improved model performance.
- **Model Performance:**
  Ensemble models (Random Forest, Gradient Boosting, and XGBoost) captured complex relationships better than linear models. Fine-tuning via hyperparameter optimization further improved generalizability and reduced overfitting.
- **Deployment Impact:**
  The interactive prediction module—both in the Jupyter environment and via Streamlit—demonstrates the practical application of the model. This approach provides decision support to insurance underwriters and claims adjusters by enabling real-time risk assessment and premium optimization.

11 .CONCLUSION

This comprehensive analysis demonstrates an end-to-end process for predicting insurance claim severity. The final model, validated through cross-validation and fine-tuning, provides reliable predictions that can be integrated into operational systems. The deployment in Streamlit further extends the model's accessibility, allowing both technical and non-technical users to benefit from real-time predictions.