AUTO INSURANCE FRAUD DETECTION

Submitted to

JAWAHARLAL NEHRU TECNOLOGICAL UNIVERSITY, HYDERABAD

In partial fulfilment of the requirements for the award of the degree of

MASTER OF COMPUTER APPLICTION

In

COMPUTER SCIENCE AND ENGINEERING (MCA)

Submitted By

NARRA SAIRAM **23UK1F0008**

Under the guidance of

Mrs S ANOOSHA

Assistant Professor

DEPARTMENT OF MASTER OF COMPUTER APPLICATION
VAAGDEVI ENGINEERING COLLEGE

Affiliated to JNTUH, HYDERABAD

BOLLIKUNTA, WARANGAL (T.S) – 506005

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING (MCA)
VAAGDEVI ENGINEERING COLLEGE (AUTONOMOUS)

CERTIFICATE OF COMPLETION PROJECT WORK REVIEW-I

This is to certify that the PG Project Phase-1 entitled **“AUTO INSURANCE FRAUD DETECTION”** is being submitted by **NARRA SAIRAM (23UK1F0008)** in partial fulfilment of the requirements for the award of the degree of master of computer applications in Computer Science and Engineering to Jawaharlal Nehru Technological University Hyderabad during the academic year 2023- 2024.

Project Guide	HOD
Mrs S ANOOSHA	**Dr. R NAVEEN KUMAR**
(Assistant Professor)	(Professor)

External

ACKNOWLEDGEMENT

We wish to take this opportunity to express our sincere gratitude and deep sense of respect to our beloved **Dr. SYED MUSTHAK AHMED,** Principal, Vaagdevi Engineering College for making us available all the required assistance and for his support and inspiration to carry out this PG Project Phase-1 in the institute.

We extend our heartfelt thanks to **Dr. R NAVEEN KUMAR** Head of the Department of computer science and engineering (MCA), Vaagdevi Engineering College for providing us necessary infrastructure and thereby giving us freedom to carry out the PG Project Phase-1.

We express heartfelt thanks to Smart Bridge Educational Services Private Limited, for their constant supervision as well as for providing necessary information regarding the PG Project Phase-1 and for their support in completing the PG Project Phase-1.

We express heartfelt thanks to the guide, **MRs S ANOOSHA,** Assistant professor, Department of CSE for his constant support and giving necessary guidance for completion of this PG Project Phase-1.

Finally, we express our sincere thanks and gratitude to my family members, friends for their encouragement and outpouring their knowledge and experience throughout the thesis.

NARRA SAIRAM **(23UK1F0008)**

ABSTRACT

- The detection of auto insurance fraud is a critical task in the insurance industry, aimed at reducing financial losses and maintaining trust among stakeholders. This paper explores various methods and technologies employed in the detection of fraudulent auto insurance claims. Key techniques include data mining, machine learning algorithms, and anomaly detection, which are utilized to analyse large datasets containing claim information and identify suspicious patterns.

- Additionally, the paper examines the role of advanced technologies such as artificial intelligence and predictive analytics in enhancing fraud detection accuracy and efficiency. Case studies and real-world examples are presented to illustrate the application and effectiveness of these methods in different scenarios

- The findings highlight the importance of continuous innovation and adaptation of detection strategies to stay ahead of evolving fraudulent activities in the auto insurance sector.

- Auto insurance fraud is a significant issue impacting insurers globally, leading to financial losses and increased premiums for policyholders. Effective detection methods are crucial to mitigate these losses and maintain trust in the insurance industry.
- This document presents an overview of current techniques, challenges, and proposed solutions for auto insurance fraud detection.

TABLE OF CONTENTS: -

1. **INTRODUCTION 1**

1.1 OVERVIEW… 1-8

1.2 PURPOSE 9

2. **LITERATURE SURVEY 9**

2.1 EXISTING PROBLEM 10

2.2 PROPOSED SOLUTION 11

3. **THEORITICAL ANALYSIS… 12**

3.1 BLOCK DIAGRAM 12

3.2 HARDWARE /SOFTWARE DESIGNING 13

4. **FLOWCHART… 14**

5. **RESULTS… 15-17**

6. **ADVANTAGES AND DISADVANTAGES… 18-19**

7. **APPLICATIONS 20**

8. **CONCLUSION 21**

9. **FUTURE SCOPE… 22**

10. **APPENDIX (SOURCE CODE) & CODE SNIPPETS 23-40**

1.INTRODUCTION

1.1. OVERVIEW

- Auto insurance fraud detection is a critical process within the insurance industry aimed at identifying and mitigating fraudulent activities related to auto insurance claims. Fraud in auto insurance can take various forms, from staged accidents and inflated claims to false reports of vehicle theft. These fraudulent activities not only impact insurance companies financially but also contribute to higher premiums for honest policyholders

★ **Purpose and Importance**

- The primary goal of auto insurance fraud detection is to ensure fair treatment for policyholders and maintain the financial stability of insurance providers. Detecting fraud helps in preventing illegitimate claims from being paid out, thereby reducing overall costs and preserving the integrity of insurance systems. It also helps in combating organized crime rings that exploit insurance loopholes for financial gain.

★ **Types of Auto Insurance Fraud**

Auto insurance fraud can be broadly categorized into two main types

- **Hard Fraud:** This involves deliberate acts of deception, such as staging accidents or filing false claims for accidents or vehicle damage that did not occur. Examples include deliberately causing a collision with another vehicle or submitting forged documents to support a claim.
- **Soft Fraud:** Also known as opportunistic fraud, soft fraud occurs when legitimate claims are exaggerated to obtain larger payouts. This can include inflating the cost of repairs or falsely attributing pre-existing damage to an insured incident.

★ **Detection Methods**

- Detecting auto insurance fraud involves a combination of investigative techniques and advanced technologies:
- **Data Analysis:** Insurers analyze large volumes of data, including claim histories, policyholder information, and external databases, to identify suspicious patterns and anomalies that may indicate fraud.
- **Pattern Recognition:** Statistical models and machine learning algorithms are used to detect unusual claim patterns or behaviors that deviate from typical claim profiles. This includes analyzing the frequency of claims, geographical patterns, and claimant behavior.
- **Claim Investigation:** Experienced investigators review claims thoroughly, verifying details through interviews, inspections, and documentation to ensure consistency and accuracy. This may involve collaborating with law enforcement agencies and other experts to uncover fraudulent activities.
- **Technology Integration:** Emerging technologies such as artificial intelligence (AI) and predictive analytics are increasingly being employed to enhance fraud detection capabilities. AI algorithms can analyze vast datasets in real-time, identifying complex fraud patterns and adapting to new fraud tactics.
- **Collaboration and Information Sharing:** Insurance companies often collaborate with industry associations, law enforcement agencies, and regulatory bodies to share information and best practices for detecting and preventing fraud. This collective effort strengthens fraud detection capabilities across the insurance sector.

★ **Challenges and Future Directions:**

- Despite advancements in fraud detection technology, challenges persist, including the adaptability of fraudsters to new detection methods and the need for balancing fraud prevention with customer service and claims processing efficiency. Future directions in auto insurance fraud detection include leveraging blockchain technology for secure data management and implementing more sophisticated AI-driven fraud detection systems.

- Auto insurance fraud detection is a vital component of maintaining the trust and reliability of insurance systems. By employing a combination of investigative expertise, advanced technologies, and collaborative efforts, insurers can effectively combat fraud and protect the interests of honest policyholders. Continual innovation and adaptation to emerging fraud tactics will be crucial in staying ahead in the ongoing battle against auto insurance fraud.

➢ Auto insurance fraud detection employs a variety of techniques and technologies to combat fraudulent activities. By leveraging data analytics, AI, and specialized software, insurers can detect and prevent fraud effectively while balancing the need for accuracy and customer privacy. Continuous adaptation and improvement in detection methods are crucial in staying ahead of increasingly sophisticated fraudulent schemes.

➢ Auto insurance fraud is a significant issue impacting insurers globally, leading to financial losses and increased premiums for policyholders. Effective detection methods are crucial to mitigate these losses and maintain trust in the insurance industry. This document presents an overview of current techniques, challenges, and proposed solutions for auto insurance fraud detection.

1.2 PURPOSE

- The purpose of auto insurance fraud detection is to identify and prevent fraudulent activities related to auto insurance claims. Insurance fraud occurs when individuals or groups deliberately deceive insurance companies for financial gain. In the context of auto insurance, fraud can take various forms, such as:

- **Staged Accidents:** Deliberately causing accidents or exaggerating the extent of damage to claim insurance benefits.
- **False Claims:** Submitting claims for accidents or damage that did not occur or exaggerating the severity of injuries sustained.
- **Policy Fraud:** Providing false information when purchasing an insurance policy (e.g., misrepresenting driving history or vehicle condition) to obtain lower premiums.
- **Identity Theft:** Using someone else's identity to obtain insurance coverage or make claims.

- **Vehicle Fraud:** Falsifying vehicle details (e.g., mileage, condition) to manipulate insurance coverage or claim payouts.

- Detecting and preventing these fraudulent activities is crucial for insurance companies to maintain fair premiums for honest policyholders and to minimize financial losses. Auto insurance fraud detection typically involves using advanced analytics, machine learning algorithms, and data mining techniques to analyze large volumes of data. Suspicious patterns and anomalies in claims data, customer information, and historical trends are flagged for further investigation by fraud specialists. This proactive approach helps insurance companies mitigate risks associated with fraudulent claims and maintain the integrity of their operations.

2.LITERATURE SURVEY

2.1 EXISTING PROBLEM

- Auto insurance fraud detection faces several challenges and existing problems, despite advancements in technology and methodologies. Some of the key issues include:

- **Sophisticated Fraud Schemes:** Fraudsters continually evolve their tactics to evade detection. They may employ sophisticated methods such as staged accidents with multiple participants, complex networks of accomplices, or using advanced technology to create false documentation.
- **Data Quality and Integration:** Insurance fraud detection relies heavily on data from various sources, including claims history, policyholder information, and external databases. Ensuring the accuracy, completeness, and timely integration of these data sets can be challenging. Inconsistent or incomplete data can lead to missed detection opportunities or false alarms.
- **Real-Time Processing:** Detecting fraud in real-time is crucial to prevent payouts for fraudulent claims. However, the sheer volume of data and the need for rapid analysis pose significant challenges. Delayed detection can result in higher financial losses for insurance companies.
- **Privacy Concerns:** Gathering and analyzing large amounts of personal data raise privacy concerns. Striking a balance between effective fraud detection and protecting policyholders' privacy rights is essential but challenging.
- **Regulatory Compliance:** Insurance companies must comply with regulatory requirements regarding data privacy, consumer rights, and fraud investigation procedures. These regulations can vary by jurisdiction and add complexity to fraud detection efforts.
- **Cost of Detection:** Implementing robust fraud detection systems and maintaining skilled fraud detection teams can be costly. Balancing the investment in fraud detection technology and resources with the potential savings from fraud prevention is a continual challenge.

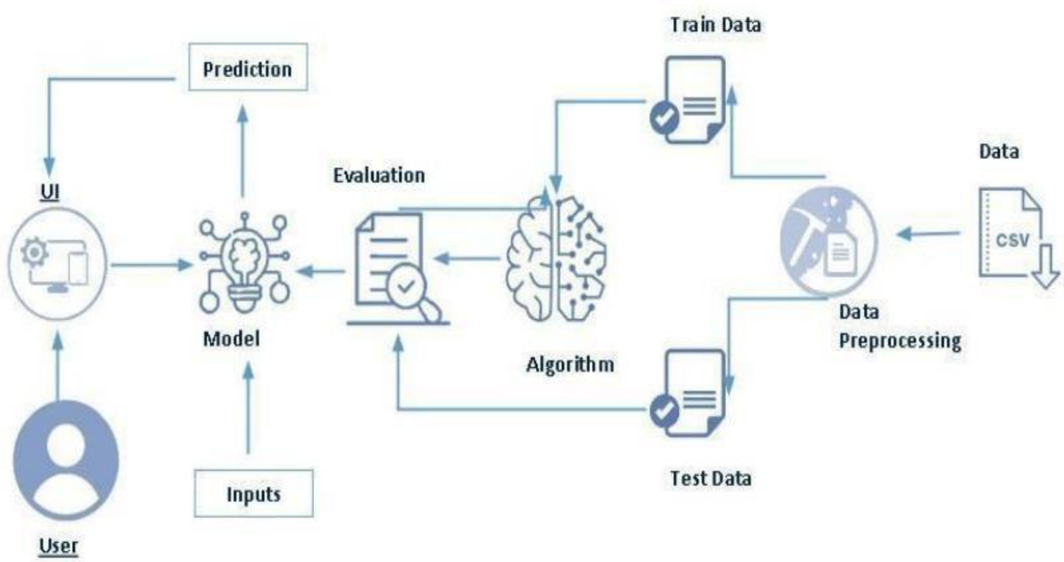
2.2 PROPOSED SOLLUTION

- Proposed solutions for improving auto insurance fraud detection involve leveraging advanced technologies, enhancing data analytics capabilities, fostering industry collaboration, and implementing robust processes. Here are key solutions:

- **Advanced Analytics and AI:** Utilize artificial intelligence (AI) and machine learning algorithms to analyze large volumes of data in real-time. These technologies can detect patterns, anomalies, and suspicious behavior that human analysts might overlook. AI can continuously learn from new data and adapt to evolving fraud schemes.
- **Predictive Modeling:** Develop predictive models that assess risk factors and detect potential fraud before claims are processed. Predictive analytics can identify high-risk claims based on historical data, fraud indicators, and external factors.
- **Integrated Data Systems:** Implement integrated data systems that consolidate and validate information from multiple sources (e.g., claims history, policy details, external databases). This ensures data accuracy and completeness, enabling more effective fraud detection.
- **Behavioral Analysis:** Utilize behavioral analytics to monitor and detect unusual behavior patterns among policyholders, claimants, and third parties. Behavioral analysis can identify deviations from typical behavior that may indicate fraudulent activity.
- **Collaboration and Information Sharing:** Foster collaboration among insurance companies, law enforcement agencies, and industry associations to share fraud intelligence, best practices, and industry standards. Collaborative efforts can enhance fraud detection capabilities and improve response times to emerging fraud threats.
- **Real-Time Monitoring and Alerts:** Implement real-time monitoring systems that generate alerts for suspicious activities or anomalies. This allows insurers to intervene promptly and investigate potentially fraudulent claims before payouts are made.
- **Fraud Detection Tools and Software:** Invest in specialized fraud detection tools and software designed to automate fraud detection processes, streamline investigations, and reduce false positives. These tools can incorporate rules-based systems, anomaly detection, and network analysis techniques.
- **Training and Education:** Provide ongoing training to employees and agents on recognizing fraud indicators, ethical practices, and compliance with anti-fraud policies. Educated staff can play a critical role in early detection and prevention of fraud.
- **Regulatory Compliance:** Ensure adherence to regulatory requirements and collaborate with regulatory bodies to align fraud detection practices with legal frameworks. Compliance with regulations enhances credibility and trustworthiness in fraud prevention efforts.
- By implementing these proposed solutions, insurance companies can enhance their ability to detect and prevent auto insurance fraud effectively. These strategies not only mitigate financial losses but also uphold fairness in premiums for honest policyholders and maintain the integrity of the insurance industry as a whole.

3.THEORITICAL ANALYSIS

3.1. BLOCK DIAGRAM



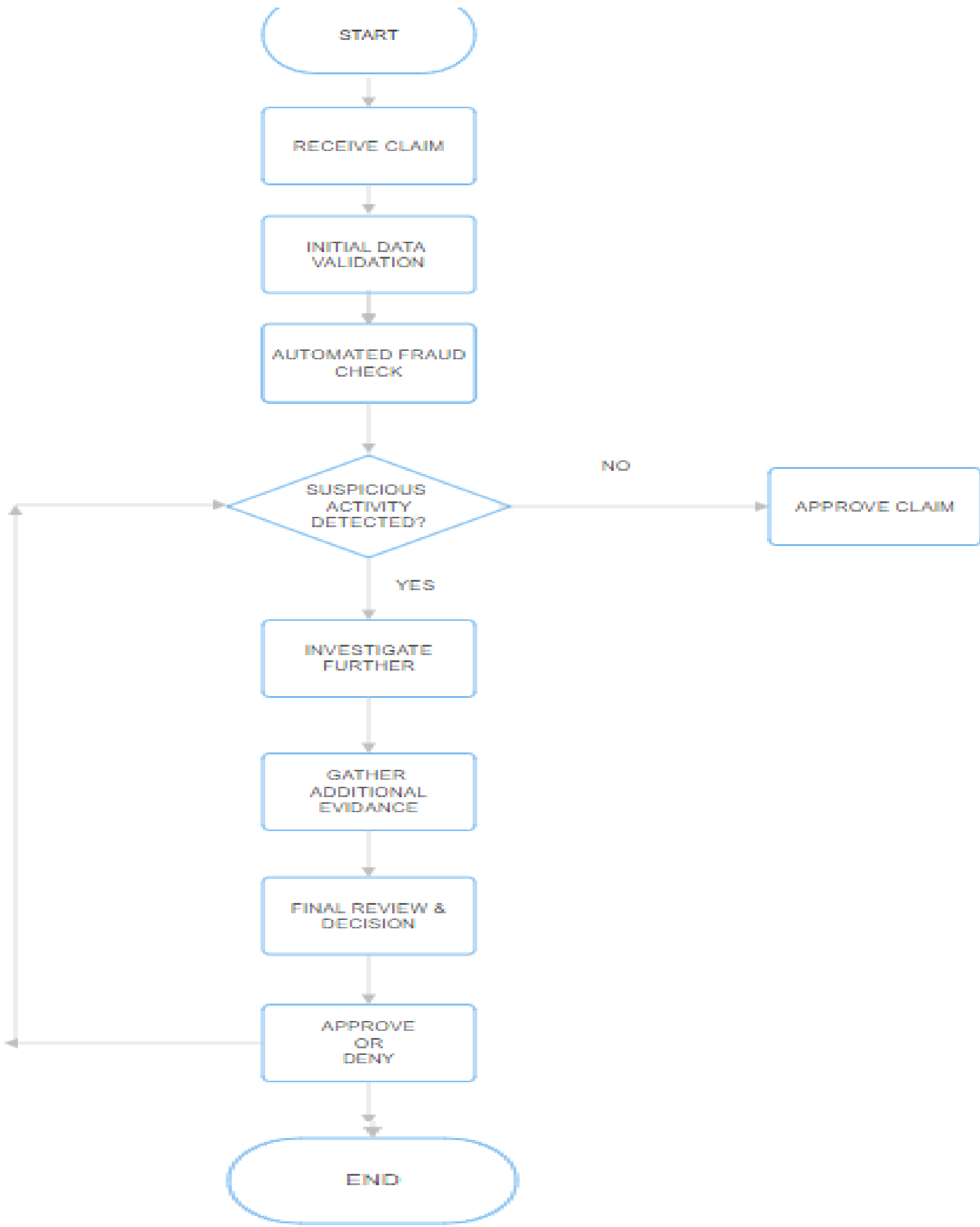
3.2. SOFTWARE DESIGNING

The following is the Software required to complete this project:

- **JUPYTER NOTEBOOK**: Google Colab will serve as the development and execution environment for your predictive modeling, data preprocessing, and model training tasks. It provides a cloud-based Jupyter Notebook environment with access to Python libraries and hardware acceleration.

- **Dataset (CSV File)**: The dataset in CSV format is essential for training and testing your predictive model. It should include historical air quality data, weather information, pollutant levels, and other relevant features.
- **Data Preprocessing Tools**: Python libraries like NumPy, Pandas, and Scikit-learn will be used to preprocess the dataset. This includes handling missing data, feature scaling, and data cleaning.
- **Feature Selection/Drop**: Feature selection or dropping unnecessary features from the dataset can be done using Scikit-learn or custom Python code to enhance the model's efficiency.
- **Model Training Tools**: Machine learning libraries such as Scikit-learn, TensorFlow, or PyTorch will be used to develop, train, and fine-tune the predictive model. Regression or classification models can be considered, depending on the nature of the AQI prediction task.
- **Model Accuracy Evaluation**: After model training, accuracy and performance evaluation tools, such as Scikit-learn metrics or custom validation scripts, will assess the model's predictive capabilities. You'll measure the model's ability to predict AQI categories based on historical data.
- **UI Based on Flask Environment**: Flask, a Python web framework, will be used to develop the user interface (UI) for the system. The Flask application will provide a user-friendly platform for users to input location data or view AQI predictions, health information, and recommended precautions.
- Jupyter NoteBook will be the central hub for model development and training, while Flask will facilitate user interaction and data presentation. The dataset, along with data preprocessing, will ensure the quality of the training data, and feature selection will optimize the model. Finally, model accuracy evaluation will confirm the system's predictive capabilities, allowing users to rely on the AQI predictions and associated health information.

<u>4.FLOW CHART</u>



5.RESULT

HOME PAGE

ABOUT

FRAUD DETECTION

1.Data Collection and Preprocessing

The first step involves collecting Insurance data and preprocessing it to handle missing values, Handling Categorical data and outliers, and inconsistencies.

2.Feature Engineering and Model Selection

The second step involves selecting relevant features and transforming them into a format suitable for building a machine learning model, as well as selecting an appropriate algorithm such as **KNN, Naive Balyse,Decision Trees, Random Forest, or SVM.**

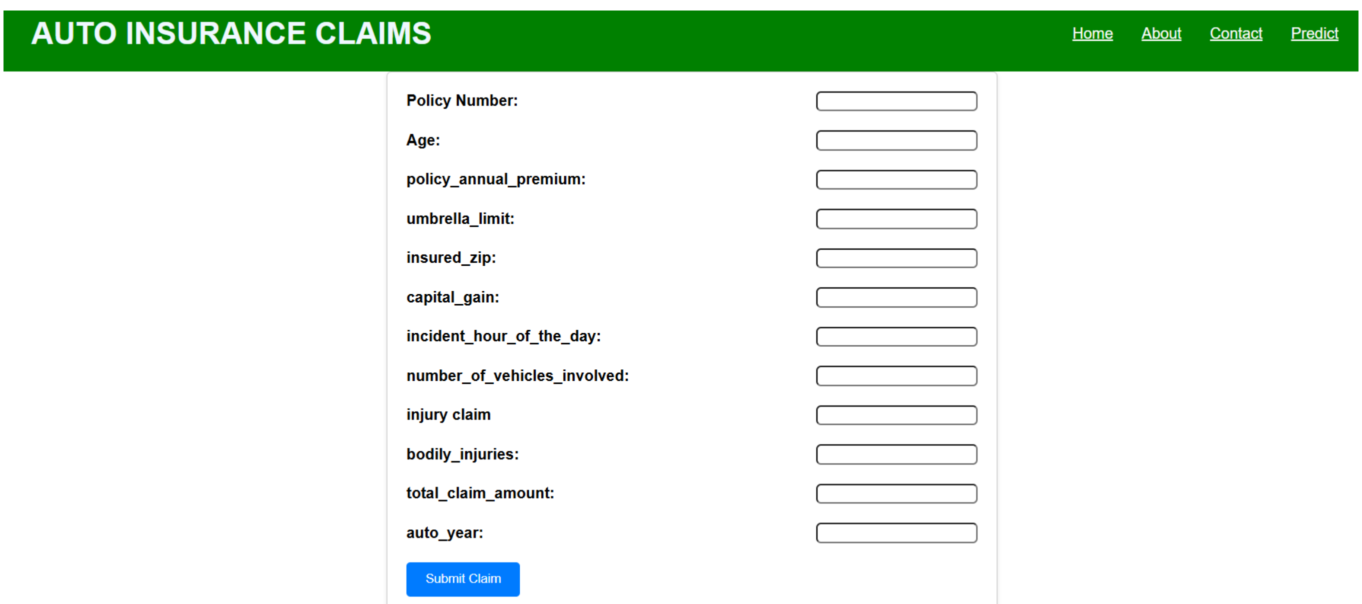
3.Model Training and Evaluation

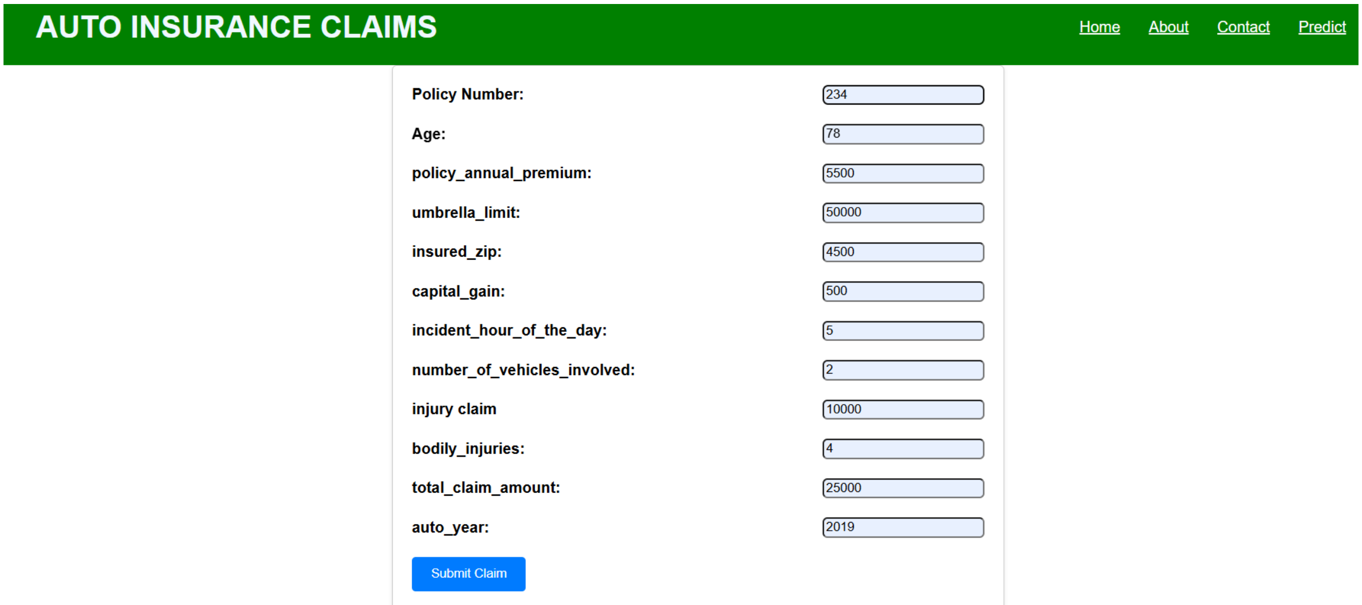
The third step involves training the selected model using the preprocessed data and evaluating its performance using metrics such as accuracy, precision.

4.Model Deployment

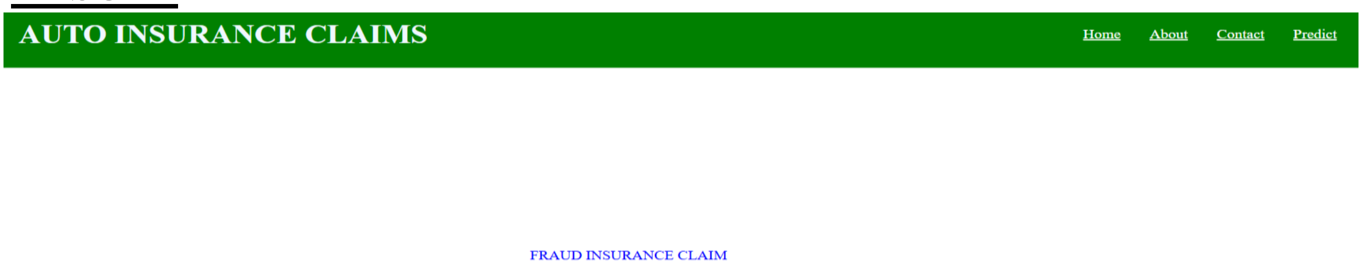
The final step involves deploying the model in a real world scenario to classify Fraud claims in real-time, so that No Frauds can be happened in insurance Claims

PREDICTIONS




AUTO INSURANCE CLAIMS
Home
About
Contact
Predict
Policy Number:
234
Age:
78
policy_annual_premium:
5500
umbrella_limit:
50000
insured_zip:
4500
capital_gain:
500
incident_hour_of_the_day:
5
number_of_vehicles_involved:
2
injury claim
10000
bodily_injuries:
4
total_claim_amount:
25000
auto_year:
2019
Submit Claim

RESULT


AUTO INSURANCE CLAIMS
Home
About
Contact
Predict
FRAUD INSURANCE CLAIM

7.ADVANTAGES AND DISADVANTAGES

ADVANTAGES:

- Auto insurance fraud detection offers several advantages, which are crucial for both insurance companies and their clients:

1. **Cost Reduction**: Detecting and preventing fraud helps insurance companies save money by minimizing payouts on fraudulent claims. This, in turn, can lead to lower premiums for honest policyholders.
2. **Improved Risk Assessment**: Fraud detection systems often analyze data to identify patterns and anomalies that indicate potential fraud. This data analysis helps insurers better understand and assess risk, leading to more accurate pricing and underwriting decisions.
3. **Enhanced Customer Trust**: By effectively detecting and preventing fraud, insurance companies demonstrate their commitment to fairness and integrity. This can improve trust and satisfaction among customers who appreciate knowing that their premiums are not subsidizing fraudulent claims.
4. **Streamlined Claims Processing**: Fraud detection systems can automate the initial screening of claims, flagging suspicious cases for further review. This helps streamline the claims process for legitimate claimants by reducing delays caused by fraudulent claims.
5. **Compliance and Legal Protection**: Insurance companies face regulatory requirements to combat fraud. Implementing robust fraud detection systems ensures compliance with these regulations, reducing the risk of legal and regulatory penalties.
6. **Early Intervention**: Detecting fraud early allows insurers to intervene promptly, potentially stopping ongoing fraudulent activities and preventing additional losses.
7. **Data-Driven Insights**: The data collected and analyzed for fraud detection purposes can provide valuable insights into trends and patterns of fraudulent behavior. This information can be used to refine fraud detection algorithms and improve overall risk management strategies.
8. **Reduction in False Positives**: Advanced fraud detection systems are designed to minimize false positives (legitimate claims flagged as fraudulent). This helps avoid unnecessary inconvenience for honest policyholders.
9. **Adaptability and Scalability**: Modern fraud detection technologies can adapt to evolving fraud schemes and scale as the insurance business grows. This flexibility allows insurers to stay ahead of new and emerging threats.
10. Auto insurance fraud detection contributes significantly to the financial health of insurance companies, the satisfaction of honest policyholders, and the integrity of the insurance industry as a whole.

DISADVANTAGES:

- While auto insurance fraud detection offers numerous advantages, there are also some potential disadvantages and challenges associated with its implementation:

1. **Cost of Implementation**: Setting up and maintaining effective fraud detection systems can be expensive. This includes costs associated with acquiring and integrating advanced technology, hiring skilled analysts, and ongoing system maintenance.
2. **False Positives**: One of the challenges of fraud detection is the risk of false positives—legitimate claims mistakenly flagged as fraudulent. This can lead to delays and frustrations for honest policyholders who have to prove the validity of their claims.
3. **Complexity of Data Analysis**: Analyzing vast amounts of data to detect fraud requires sophisticated algorithms and computing power. Ensuring the accuracy and reliability of these algorithms can be challenging, especially in the face of rapidly evolving fraud tactics.
4. **Privacy Concerns**: Fraud detection systems often rely on extensive data collection and analysis, which can raise concerns about privacy and data security. Insurance companies must carefully navigate regulatory requirements and customer expectations regarding data usage.
5. **Adaptation to New Fraud Schemes**: Fraudsters are continually evolving their tactics to circumvent detection systems. Keeping fraud detection methods up-to-date and effective against new fraud schemes requires ongoing investment and innovation.
6. **Impact on Customer Experience**: Overly aggressive fraud detection measures can create a negative customer experience. Lengthy investigations, additional paperwork, and delays in claims processing can frustrate policyholders, even if they understand the need for fraud prevention.
7. **Ethical Considerations**: There are ethical considerations around the use of data analytics and algorithms in fraud detection. Ensuring fairness, transparency, and accountability in how fraud detection systems operate is essential to maintain trust with customers and stakeholders.
8. **Resource Intensive Investigations**: Investigating suspected fraud cases can be resource-intensive, requiring skilled personnel and significant time and effort. This can strain the resources of insurance companies, particularly smaller firms with limited budgets.
9. **Legal and Regulatory Challenges**: Compliance with various legal and regulatory requirements related to fraud detection can be complex and costly. Insurance companies must navigate different laws and regulations across jurisdictions, which adds another layer of complexity.
10. Auto insurance fraud detection systems are crucial for mitigating financial losses and maintaining the integrity of insurance operations, they also present challenges related to cost, accuracy, privacy, customer experience, and regulatory compliance. Balancing these factors is essential for insurers seeking to effectively combat fraud while maintaining positive relationships with their policyholders.

8.APPLICATIONs

- Auto insurance fraud detection has several practical applications across different stages of insurance operations. Some key applications include:

1. **Claims Processing Automation**: Automated fraud detection systems can analyze incoming claims data in real-time. They identify suspicious patterns or anomalies that may indicate potential fraud, allowing insurers to prioritize high-risk claims for manual review while expediting legitimate claims.
2. **Anomaly Detection**: Using advanced analytics and machine learning algorithms, insurers can detect unusual patterns in data that suggest fraudulent activities. These anomalies may include sudden spikes in claims from specific geographic areas, unusual claim types, or discrepancies in policyholder information.
3. **Social Network Analysis**: Fraud detection systems can analyze relationships between policyholders, service providers, and other stakeholders to identify networks of potentially fraudulent activity. This helps uncover organized fraud rings that coordinate fraudulent claims across multiple policies or individuals.
4. **Behavioral Analytics**: By analyzing historical data and current behavior patterns, insurers can identify deviations from normal behavior that may indicate fraud. For example, sudden changes in claim frequency, severity, or timing can signal fraudulent activity.
5. **Image and Text Analysis**: Insurers can utilize image and text analysis technologies to verify claim documentation, such as photos of vehicle damage or medical reports. This helps detect forged documents or misleading information submitted to support fraudulent claims.
6. **Predictive Modeling**: Predictive modeling techniques can forecast the likelihood of fraud based on various risk factors and historical data. This allows insurers to proactively monitor high-risk policies or claimants and take preventive actions before fraudulent activities occur.
7. **Post-Claim Investigation Support**: Fraud detection systems provide valuable support during post-claim investigations by flagging suspicious cases and providing evidence-based insights. Investigators can use this information to conduct thorough inquiries and gather additional evidence to support fraud prosecution.
8. **Compliance and Regulatory Reporting**: Fraud detection systems help insurers comply with regulatory requirements related to fraud prevention and reporting. They enable accurate documentation and reporting of suspected fraud cases to regulatory authorities, demonstrating due diligence in fraud management.
9. **Fraud Awareness and Training**: Insurers can use data from fraud detection systems to enhance fraud awareness among employees and policyholders. Training programs can educate stakeholders about common fraud schemes, warning signs, and preventive measures.
10. **Continuous Improvement**: By continuously analyzing and learning from data patterns, fraud detection systems improve over time. Insurers can refine algorithms, update rules, and adapt strategies to stay ahead of evolving fraud tactics and enhance detection accuracy.

- Auto insurance fraud detection applications help insurers mitigate financial losses, improve operational efficiency, enhance regulatory compliance, and maintain trust with policyholders by ensuring fair and transparent claims processing.

9.CONCLUSION

- ✓ In conclusion, auto insurance fraud detection plays a critical role in safeguarding insurers, policyholders, and the overall integrity of the insurance industry. By leveraging advanced technologies such as data analytics, machine learning, and artificial intelligence, insurers can effectively identify and prevent fraudulent activities at various stages of the insurance process.

- ✓ The benefits of auto insurance fraud detection are manifold. It reduces financial losses by minimizing payouts on fraudulent claims, thereby potentially lowering premiums for honest policyholders. It enhances the accuracy of risk assessment and underwriting decisions, leading to fairer pricing and improved operational efficiency. Moreover, fraud detection systems contribute to regulatory compliance, ensuring insurers meet legal obligations while maintaining trust and transparency with stakeholders.
- ✓ However, implementing fraud detection systems also presents challenges, such as the cost of technology and data privacy concerns. Balancing these challenges with the benefits requires careful consideration of ethical implications, customer experience, and regulatory requirements.
- ✓ Ultimately, the continuous evolution and adoption of sophisticated fraud detection technologies are essential for insurers to stay ahead of increasingly complex fraud schemes. By investing in robust fraud prevention strategies, insurers can foster a more secure and sustainable insurance environment for all parties involved.

10.FUTURE SCOPE

- The future of auto insurance fraud detection holds significant promise, driven by advancements in technology and evolving fraud tactics. Several key areas represent the future scope of auto insurance fraud detection:

- ✓ **Artificial Intelligence and Machine Learning**: AI and ML will continue to play a crucial role in enhancing fraud detection capabilities. These technologies can analyze large volumes of data in real-time, identify complex patterns, and adapt to new fraud schemes more effectively than traditional methods.
- ✓ **Predictive Analytics**: Predictive modeling will become more sophisticated, allowing insurers to anticipate fraudulent behavior before it occurs. By analyzing historical data and identifying predictive indicators, insurers can proactively mitigate risks and prevent fraudulent activities.
- ✓ **Integration of Big Data**: The integration of big data sources—from IoT devices in vehicles to social media data—will provide insurers with more comprehensive insights into policyholder behavior and potential fraud indicators. This holistic approach enhances the accuracy and depth of fraud detection efforts.
- ✓ **Enhanced Digital Verification**: Technologies such as blockchain and digital identities will improve the verification of claim documents and policyholder information, reducing the risk of identity theft and document fraud.
- ✓ **Real-time Monitoring and Alerts**: Continuous monitoring of transactions and interactions will enable insurers to detect suspicious activities in real-time, triggering immediate alerts for further investigation and intervention.
- ✓ **Collaborative Intelligence**: Sharing data and insights across insurers and industry stakeholders will strengthen fraud detection capabilities. Collaborative platforms and networks can facilitate the exchange of information on fraud trends, patterns, and prevention strategies.
- ✓ **Behavioral Biometrics**: Utilizing behavioral biometrics—such as keystroke dynamics and voice recognition—can add an additional layer of authentication and fraud detection, particularly in digital interactions and claims processing.
- ✓ **Regulatory Compliance and Transparency**: As regulatory requirements around data privacy and fraud prevention evolve, future systems will need to ensure compliance while maintaining transparency in their operations and decision-making processes.

- ✓ **Enhanced Customer Experience**: Future fraud detection systems will strive to minimize false positives and streamline legitimate claims processing, enhancing overall customer satisfaction and trust in insurers' fraud prevention measures.
- ✓ **Adaptation to Emerging Threats**: With fraudsters continually evolving their tactics, future fraud detection systems will need to be agile and adaptive. Continuous monitoring, learning from new data patterns, and rapid response to emerging threats will be critical.
- ✓ The future scope of auto insurance fraud detection lies in leveraging advanced technologies to enhance accuracy, efficiency, and proactive prevention of fraudulent activities. By embracing innovation and collaboration, insurers can stay ahead of fraudsters and create a more secure and resilient insurance ecosystem for policyholders and stakeholders alike.

11.APPENDIX

Model building :

1)Dataset
2)Jupyter Notebook and VS code Application Building

1. HTML file (Index file, Predict file)
1. CSS file
2. Models in pickle format

SOURCE CODE:

INDEX.HTML

```
 <!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Document</title>
   <style>
      body{
   margin: 0;
   border: 0;
   padding: 0;
   background-color: transparent;
}
.home{
   background-image: url('https://res.cloudinary.com/dn0jqytyw/image/upload/v1720071894/image1_ywg7rb.jpg');
   background-size: cover;
   background-repeat: no-repeat;
   background-position: center;
   width: 100%;
   height:100vh;
   padding: 0;
   margin: 0;
   border: 0;
}
.navbar{
   display: flex;
   flex-direction: row;
   justify-content: space-between;
   background-color: transparent;
}
.navbar-left{
   color:aliceblue;
```

```
   padding-left: 50px;
    }.navbar-right{
    color: aliceblue;
    padding-right:50px;
    display: flex;
    flex-direction: row;
    align-items: center;
}
.a{
    padding: 20px;
    margin: 10px;
    border: 2px;
    color:white;
}
.para{
    align-items: center;
    color:whitesmoke;
    padding-left: 550px;
    padding-right:550px;
padding-top: 230px;
}
.button{
    color: aqua;
}
    </style>
</head>
<body>
    <div>
        <div class="home">
            <div class="navbar">
                <div class="navbar-left">
                    <h1>AUTO INSURANCE CLAIMS</h1>
                </div>
                <div class="navbar-right">
                    <a class="a"href="{{url_for('index')}}" alt="/index">Home</a>
                    <a class="a"href="{{url_for('about')}}" alt="/about">About</a>
                    <a class="a"href="{{url_for('contact')}}" alt="/contact">Contact</a>
                    <a class="a"href="{{url_for('predict')}}" alt="/predict">Predict</a>
                </div>
            </div>
            <div class="para">
                <h1>Insurance Claims</h1>
                <p Insurance Fraud Detection can be prevented by <br>
                    analyzing the previous Fraud Data and detecting same<br>
                    trends
                </p>
            </div>
        </div>
        <form action="/predict" method="post">
            <button type="submit">Click me!</button>
        </form>
    </div>
</body>
</html>
```

PREDICT.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Predict</title>
<style>
  body{
    margin: 0;
    border: 0;
    padding: 0;
    background-color: transparent;
}.navbar{
    display: flex;
    flex-direction: row;
    justify-content: space-between;
    background-color:green;
}
.navbar-left{
    color:aliceblue;
    padding-left: 50px;
}
.navbar-right{
    color: aliceblue;
    padding-right:50px;
    display: flex;
    flex-direction: row;
    align-items: center;
}
.a{
    padding: 15px;
    color: white;
}
.form{
    display: flex;
    flex-direction: row;
    justify-content: space-between;
    background-color: transparent;
}
.form-left{
```

```
   padding-left: 40px;
   padding-top: 20px;
}
.form-right{
   padding-right: 100px;
   padding-top: 20px;
}
.button{
        display: flex;
        align-items: center;
        padding-left: 5x;
        padding-right: 200px;
      }
</style>
</head>
<body>
  <div class="home">
   <div class="navbar">
      <div class="navbar-left">
        <h1>AUTO INSURANCE CLAIMS</h1>
      </div>
      <div class="navbar-right">
       <a class="a"href="{{url_for('index')}}" alt="/index">Home</a>
       <a class="a"href="{{url_for('about')}}" alt="/about">About</a>
       <a class="a"href="{{url_for('contact')}}" alt="/contact">Contact</a>
       <a class="a"href="{{url_for('predict')}}" alt="/predict">Predict</a>
       </div>
    </div>
    <div class="form" >
     <div class="form-left">
      <form action="{{ url_for('predict') }}" method="POST">
    <label for="months_as_customer">Months as Customer:</label>
    <input type="number" id="months_as_customer" name="months_as_customer"><br><br>
    <label for="age">Age</label>
    <input type="number" id="age" name="age"><br><br>
    <label for="policy_number">Policy Number      :</label>
    <input type="number" id="policy_number" name="policy_number"><br><br>
    <label for="policy_csl">policy_csl:</label>
    <input type="number" id="policy_csl" name="policy_csl"><br><br>
    <label for="policy_deductable">policy_deductable:</label>
    <input type="number" id="policy_deductable" name="policy_deductable"><br><br>
    <label for="policy_annual_premium">policy_annual_premium:</label>
    <input type="number" id="policy_annual_premium" name="policy_annual_premium"><br><br>
    <label for="insured_zip">insured_zip:</label>
    <input type="number" id="insured_zip" name="insured_zip"><br><br>
```

```
<label for="insured_sex">insured_sex:</label>
<input type="text" id="insured_sex" name="insured_sex"><br><br>
 <label for="insured_hobbies">insured_hobbies:</label>
 <input type="text" id="insured_hobbies" name="insured_hobbies"><br><br>
<label for="insured_relationship">insured_relationship:</label>
 <input type="text" id="insured_relationship" name="insured_relationship"><br><br>
<label for="capital_gain">capital_gain:</label>
<input type="number" id="capital_gain" name="capital_gain"><br><br>
 <label for="capital_loss">capital_loss:</label>
 <input type="number" id="capital_loss" name="capital_loss"><br><br>
 </div>
 <div class="form-right">
  <label for="collision_type">collision_type:</label>
  <input type="number" id="collision_type" name="collision_type"><br><br>
 <label for="incident_severity">incident_severity:</label>
  <input type="number" id="incident_severity" name="incident_severity"><br><br>
<label for="authorities_contacted">authorities_contacted:</label>
 <input type="text" id="authorities_contacted" name="authorities_contacted"><br><br>
<label for="incident_hour_of_the_day">incident_hour_of_the_day:</label>
  <input type="number" id="incident_hour_of_the_day" name="incident_hour_of_the_day"><br><br>
 <label for="number_of_vehicles_involved">number_of_vehicles_involved:</label>
  <input type="number" id="number_of_vehicles_involved" name="number_of_vehicles_involved"><br><br>
<label for="property_damage">property_damage:</label>
 <input type="number" id="property_damage" name="property_damage"><br><br>
 <label for="injury_claim">injury claim</label>
  <input type="number" id="injury_claim" name="injury_claim"><br><br>
<label for="property_claim">property_claim</label>
  <input type="number" id="property_claim" name="property_claim"><br><br>
  <label for="bodily_injuries">bodily_injuries:</label>
  <input type="number" id="bodily_injuries" name="bodily_injuries"><br><br>
<label for="witnesses">witnesses:</label>
     <input type="number" id="witnesses" name="witnesses"><br><br>
 <label for="police_report_available">police_report_available:</label>
      <input type="number" id="police_report_available" name="police_report_available"><br><br>
<label for="total_claim_amount">total_claim_amount:</label>
   <input type="number" id="total_claim_amount" name="total_claim_amount"><br><br>
      <label for="auto_year">auto_year:</label>
      <input type="number" id="auto_year" name="auto_year"><br><br></div>
      <div class="button">
        <button type="submit">SUBMIT</button> </form></div>
  </div>
</body>
</html>
```

APP.PY

```
from flask import Flask,render_template,request
import os
import pandas as pd
import numpy as np
import pickle

app=Flask(__name__)

encoders_path=os.path.dirname(os.path.abspath(__file__))

model=pickle.load(open('dtc_model.pkl','rb'))


@app.route('/')
def index():
   return render_template('index.html')
@app.route('/about')
def about():
   return render_template('about.html')
@app.route('/contact')
def contact():
   return render_template('contact.html')

@app.route('/predict',methods=['POST','GET'])
def predict():
   print(request.method)
   if request.method=='POST':
      months_as_customer=float(request.form['months_as_customer'])
      age=float(request.form['age'])
      policy_number=float(request.form['policy_number'])
      policy_csl=float(request.form['policy_csl'])
      policy_deductable=float(request.form['policy_deductable'])
      policy_annual_premium=float(request.form['policy_annual_premium'])
      insured_zip=float(request.form['insured_zip'])
      insured_sex=float(request.form['insured_sex'])
      insured_hobbies=float(request.form['insured_hobbies'])
      insured_relationship=float(request.form['insured_relationship'])
      capital_gain=float(request.form['capital_gain'])
      capital_loss=float(request.form['capital_loss'])
      collision_type=float(request.form["collision_type"])
      incident_severity=float(request.form["incident_severity"])
      authorities_contacted=float(request.form["authorities_contacted"])
      incident_hour_of_the_day=float(request.form["incident_hour_of_the_day"])
```

```
   number_of_vehicles_involved=float(request.form["number_of_vehicles_involved"])
    property_damage=float(request.form["property_damage"])
    injury_claim=float(request.form['injury_claim'])
    property_cliam=float(request.form['property_claim'])
    bodily_injuries=float(request.form["bodily_injuries"])
    witnesses=float(request.form["witnesses"])
    police_report_available=float(request.form['police_report_available'])
    total_claim_amount=float(request.form['total_claim_amount'])
    auto_year=float(request.form['auto_year'])


pred=[[months_as_customer,age,policy_number,policy_csl,policy_deductable,policy_annual_premium,insured_zi
p,insured_sex,
insured_hobbies,insured_relationship,capital_gain,capital_loss,collision_type,incident_severity,
authorities_contacted,incident_hour_of_the_day,number_of_vehicles_involved,property_damage,injury_claim,
bodily_injuries,witnesses,police_report_available,total_claim_amount,auto_year]]
    prediction=model.predict(pred)

    result="Legal Insurance Claim" if prediction==0 else "Fraud Insurance Claim"
    return render_template('result.html',prediction_text=result)
    #print(result)
    return render_template("predict.html")

if __name__=='__main__':
  app.run(debug=True)
```

CODE SNIPPETS

MODEL BUILDING

jupyter project Last Checkpoint: 56 seconds ago

File Edit View Run Kernel Settings Help — Trusted

Code — JupyterLab — Python 3 (ipykernel)

[59]:
```
import pandas as pd
import seaborn as sns
import numpy as np
import matplotlib.pyplot as plt
```

[60]:
```
data=pd.read_csv('insurance_claims.csv')
```

[61]:
```
data
```

[61]:

	months_as_customer	age	policy_number	policy_bind_date	policy_state	policy_csl	policy_deductable	policy_annual_premium	umbrella_limit	insured_zip	...	poli
0	328	48	521585	2014-10-17	OH	250/500	1000	1406.91	0	466132	...	
1	228	42	342868	2006-06-27	IN	250/500	2000	1197.22	5000000	468176	...	
2	134	29	687698	2000-09-06	OH	100/300	2000	1413.14	5000000	430632	...	
3	256	41	227811	1990-05-25	IL	250/500	2000	1415.74	6000000	608117	...	
4	228	44	367455	2014-06-06	IL	500/1000	1000	1583.91	6000000	610706	...	
...	...	...	...	...	...	...	...	...	...	...	...	
995	3	38	941851	1991-07-16	OH	500/1000	1000	1310.80	0	431289	...	
996	285	41	186934	2014-01-05	IL	100/300	1000	1436.79	0	608177	...	
997	130	34	918516	2003-02-17	OH	250/500	500	1383.49	3000000	442797	...	
998	458	62	533940	2011-11-18	IL	500/1000	2000	1356.92	5000000	441714	...	
999	456	60	556080	1996-11-11	OH	250/500	1000	766.19	0	612260	...	

1000 rows × 40 columns

[62]:
```
data.head()
```

[62]:

	months_as_customer	age	policy_number	policy_bind_date	policy_state	policy_csl	policy_deductable	policy_annual_premium	umbrella_limit	insured_zip	...	police_
0	328	48	521585	2014-10-17	OH	250/500	1000	1406.91	0	466132	...	
1	228	42	342868	2006-06-27	IN	250/500	2000	1197.22	5000000	468176	...	
2	134	29	687698	2000-09-06	OH	100/300	2000	1413.14	5000000	430632	...	
3	256	41	227811	1990-05-25	IL	250/500	2000	1415.74	6000000	608117	...	
4	228	44	367455	2014-06-06	IL	500/1000	1000	1583.91	6000000	610706	...	

5 rows × 40 columns

[63]:
```
data.tail()
```

[63]:

	months_as_customer	age	policy_number	policy_bind_date	policy_state	policy_csl	policy_deductable	policy_annual_premium	umbrella_limit	insured_zip	...	poli
995	3	38	941851	1991-07-16	OH	500/1000	1000	1310.80	0	431289	...	
996	285	41	186934	2014-01-05	IL	100/300	1000	1436.79	0	608177	...	
997	130	34	918516	2003-02-17	OH	250/500	500	1383.49	3000000	442797	...	
998	458	62	533940	2011-11-18	IL	500/1000	2000	1356.92	5000000	441714	...	
999	456	60	556080	1996-11-11	OH	250/500	1000	766.19	0	612260	...	

5 rows × 40 columns

[64]:
```
data.replace('?',np.nan,inplace=True)
data.head()
```

[64]:

	months_as_customer	age	policy_number	policy_bind_date	policy_state	policy_csl	policy_deductable	policy_annual_premium	umbrella_limit	insured_zip	...	police_
0	328	48	521585	2014-10-17	OH	250/500	1000	1406.91	0	466132	...	
1	228	42	342868	2006-06-27	IN	250/500	2000	1197.22	5000000	468176	...	
2	134	29	687698	2000-09-06	OH	100/300	2000	1413.14	5000000	430632	...	
3	256	41	227811	1990-05-25	IL	250/500	2000	1415.74	6000000	608117	...	
4	228	44	367455	2014-06-06	IL	500/1000	1000	1583.91	6000000	610706	...	

5 rows × 40 columns

[65]:
```
data.columns
```

Create a duplicate of this cell

[65]:
```
Index(['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', 'insured_occupation', 'insured_hobbies',
       'insured_relationship', 'capital-gains', 'capital-loss',
       'incident_date', 'incident_type', 'collision_type', 'incident_severity',
       'authorities_contacted', 'incident_state', 'incident_city',
       'incident_location', 'incident_hour_of_the_day',
       'number_of_vehicles_involved', 'property_damage', 'bodily_injuries',
       'witnesses', 'police_report_available', 'total_claim_amount',
       'injury_claim', 'property_claim', 'vehicle_claim', 'auto_make',
       'auto_model', 'auto_year', 'fraud_reported', '_c39'],
      dtype='object')
```

[66]:
```
data.shape
```

[66]:
```
(1000, 40)
```

[67]:
```
data.dtypes
```

[67]:
```
months_as_customer               int64
age                              int64
policy_number                    int64
policy_bind_date                object
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                   object
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
```

```
 15  capital-gains                1000 non-null   int64
 16  capital-loss                 1000 non-null   int64
 17  incident_date                1000 non-null   object
 18  incident_type                1000 non-null   object
 19  collision_type               822 non-null    object
 20  incident_severity            1000 non-null   object
 21  authorities_contacted        909 non-null    object
 22  incident_state               1000 non-null   object
 23  incident_city                1000 non-null   object
 24  incident_location            1000 non-null   object
 25  incident_hour_of_the_day     1000 non-null   int64
 26  number_of_vehicles_involved  1000 non-null   int64
 27  property_damage              640 non-null    object
 28  bodily_injuries              1000 non-null   int64
 29  witnesses                    1000 non-null   int64
 30  police_report_available      657 non-null    object
 31  total_claim_amount           1000 non-null   int64
 32  injury_claim                 1000 non-null   int64
 33  property_claim               1000 non-null   int64
 34  vehicle_claim                1000 non-null   int64
 35  auto_make                    1000 non-null   object
 36  auto_model                   1000 non-null   object
 37  auto_year                    1000 non-null   int64
 38  fraud_reported               1000 non-null   object
 39  _c39                         0 non-null      float64
dtypes: float64(2), int64(17), object(21)
memory usage: 312.6+ KB
```

[69]:
```
data.describe()
```

[69]:

	months_as_customer	age	policy_number	policy_deductable	policy_annual_premium	umbrella_limit	insured_zip	capital-gains	capital-loss	incic
count	1000.000000	1000.000000	1000.000000	1000.000000	1000.000000	1.000000e+03	1000.000000	1000.000000	1000.000000	
mean	203.954000	38.948000	546238.648000	1136.000000	1256.406150	1.101000e+06	501214.488000	25126.100000	-26793.700000	
std	115.113174	9.140287	257063.005276	611.864673	244.167395	2.297407e+06	71701.610941	27872.187708	28104.096686	
min	0.000000	19.000000	100804.000000	500.000000	433.330000	-1.000000e+06	430104.000000	0.000000	-111100.000000	
25%	115.750000	32.000000	335980.250000	500.000000	1089.607500	0.000000e+00	448404.500000	0.000000	-51500.000000	
50%	199.500000	38.000000	533135.000000	1000.000000	1257.200000	0.000000e+00	466445.500000	0.000000	-23250.000000	
75%	276.250000	44.000000	759099.750000	2000.000000	1415.695000	0.000000e+00	603251.000000	51025.000000	0.000000	
max	479.000000	64.000000	999435.000000	2000.000000	2047.590000	1.000000e+07	620962.000000	100500.000000	0.000000	

[72]:
```
data.isna().sum()
```

[72]:
```
months_as_customer             0
age                            0
policy_number                  0
policy_bind_date               0
policy_state                   0
policy_csl                     0
policy_deductable              0
policy_annual_premium          0
umbrella_limit                 0
insured_zip                    0
insured_sex                    0
insured_education_level        0
insured_occupation             0
insured_hobbies                0
insured_relationship           0
capital-gains                  0
capital-loss                   0
incident_date                  0
incident_type                  0
collision_type                 0
incident_severity              0
authorities_contacted          0
incident_state                 0
incident_city                  0
incident_location              0
incident_hour_of_the_day       0
number_of_vehicles_involved    0
property_damage                0
bodily_injuries                0
witnesses                      0
police_report_available        0
total_claim_amount             0
injury_claim                   0
property_claim                 0
```

```
vehicle_claim                   0
auto_make                       0
auto_model                      0
auto_year                       0
fraud_reported                  0
_c39                         1000
dtype: int64
```

[73]:
```
data.head()
```

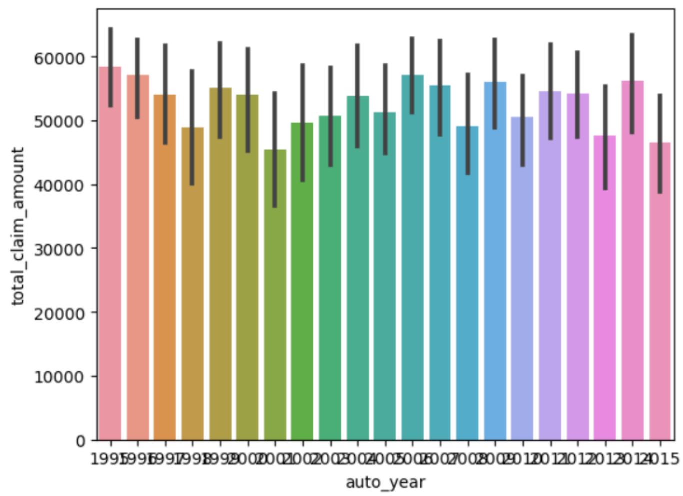
[73]:

	months_as_customer	age	policy_number	policy_bind_date	policy_state	policy_csl	policy_deductable	policy_annual_premium	umbrella_limit	insured_zip	...	police_
0	328	48	521585	2014-10-17	OH	250/500	1000	1406.91	0	466132	...	
1	228	42	342868	2006-06-27	IN	250/500	2000	1197.22	5000000	468176	...	
2	134	29	687698	2000-09-06	OH	100/300	2000	1413.14	5000000	430632	...	
3	256	41	227811	1990-05-25	IL	250/500	2000	1415.74	6000000	608117	...	
4	228	44	367455	2014-06-06	IL	500/1000	1000	1583.91	6000000	610706	...	

5 rows × 40 columns

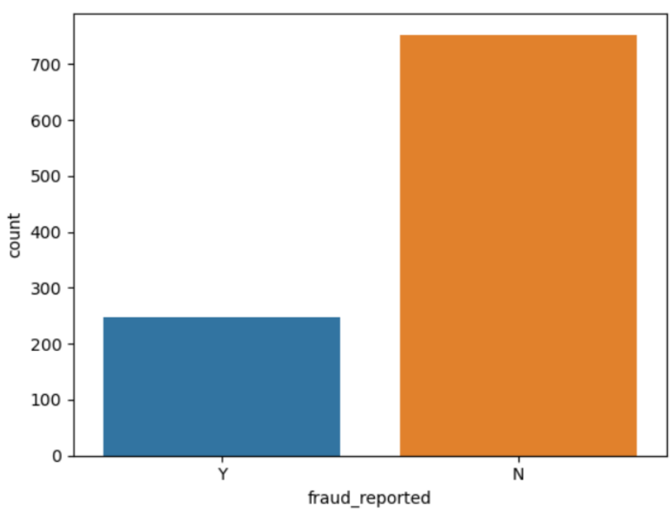
[74]:
```
sns.barplot(x='auto_year',y='total_claim_amount',data=data)
```

[74]: <Axes: xlabel='auto_year', ylabel='total_claim_amount'>

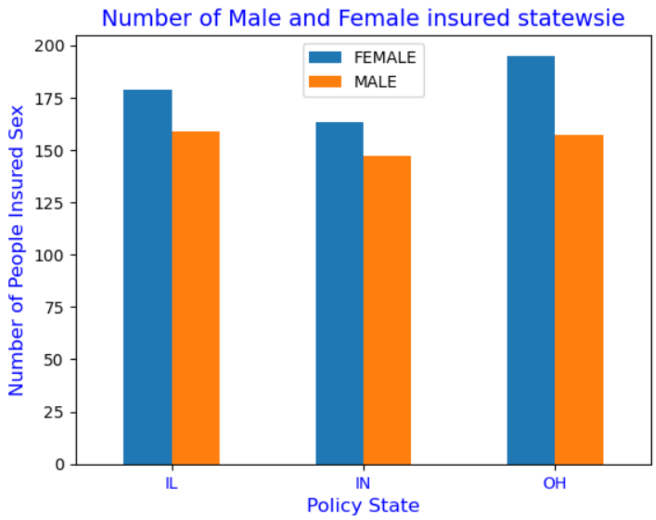


[75]:
```
sns.countplot(x='fraud_reported',data=data)
```

[75]: <Axes: xlabel='fraud_reported', ylabel='count'>



[76]:
```
insurance_state=pd.crosstab(data['policy_state'],data['insured_sex'])
insurance_state.plot(kind='bar',grid=False)
plt.xticks(rotation=0,fontsize=10,color='blue')
plt.legend(fontsize=10)
plt.xlabel('Policy State',fontsize=12,color='blue')
plt.ylabel('Number of People Insured Sex',fontsize=12,color='blue')
plt.title('Number of Male and Female insured statewsie',fontsize=14,color='blue')
plt.show()
```



```
77]: data.drop(columns=['policy_bind_date','policy_state','umbrella_limit','auto_model','insured_education_level','auto_make','incident_severity','policy_csl
           'incident_state','incident_type','incident_date','incident_city', '_c39','insured_occupation','incident_location'],inplace = True,axis=1)
```

```
78]: data.head()
```

78]:

	months_as_customer	age	policy_number	policy_deductable	policy_annual_premium	insured_zip	insured_sex	insured_hobbies	insured_relationship	capital-gains	...	p
0	328	48	521585	1000	1406.91	466132	MALE	sleeping	husband	53300	...	
1	228	42	342868	2000	1197.22	468176	MALE	reading	other-relative	0	...	
2	134	29	687698	2000	1413.14	430632	FEMALE	board-games	own-child	35100	...	
3	256	41	227811	2000	1415.74	608117	FEMALE	board-games	unmarried	48900	...	
4	228	44	367455	1000	1583.91	610706	MALE	board-games	unmarried	66000	...	

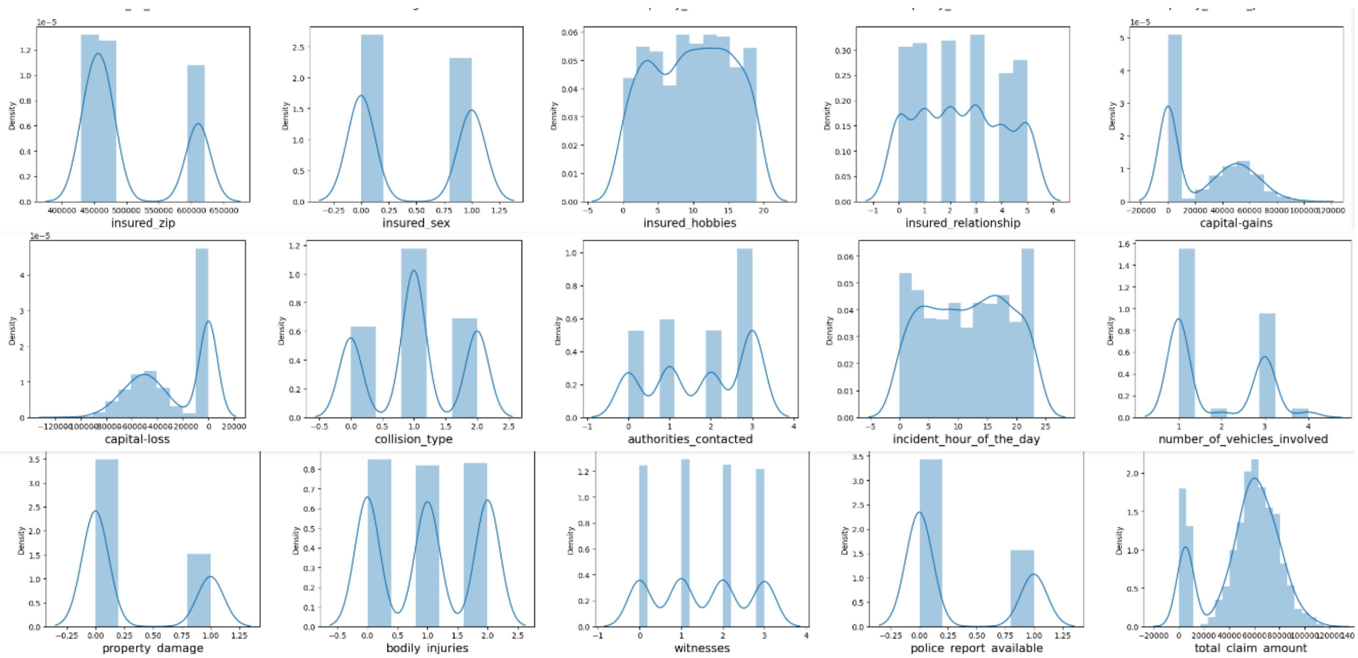
5 rows × 25 columns

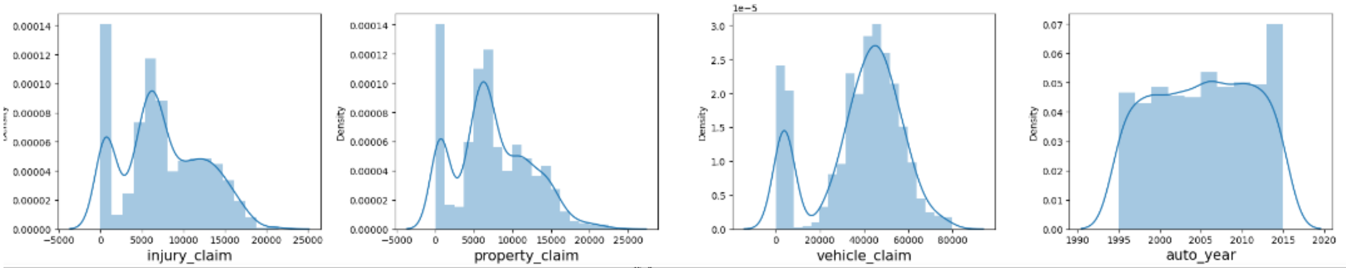
```
[79]: from sklearn.preprocessing import LabelEncoder
      le=LabelEncoder()
      for i in data.columns:
          if data[i].dtypes=='object':
              data[i]=le.fit_transform(data[i])
```

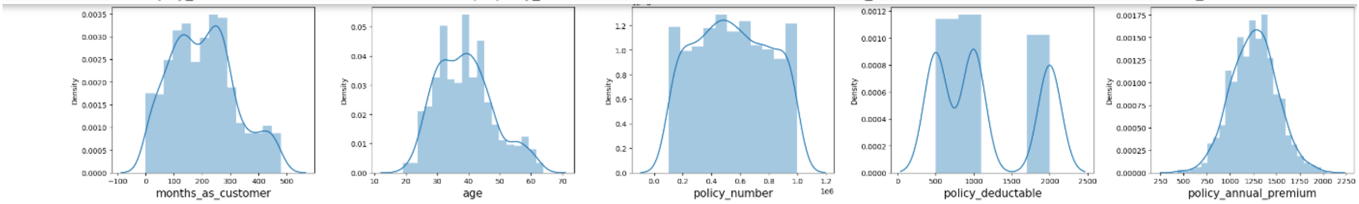
```
[80]: x=data.iloc[:,:-1]
```

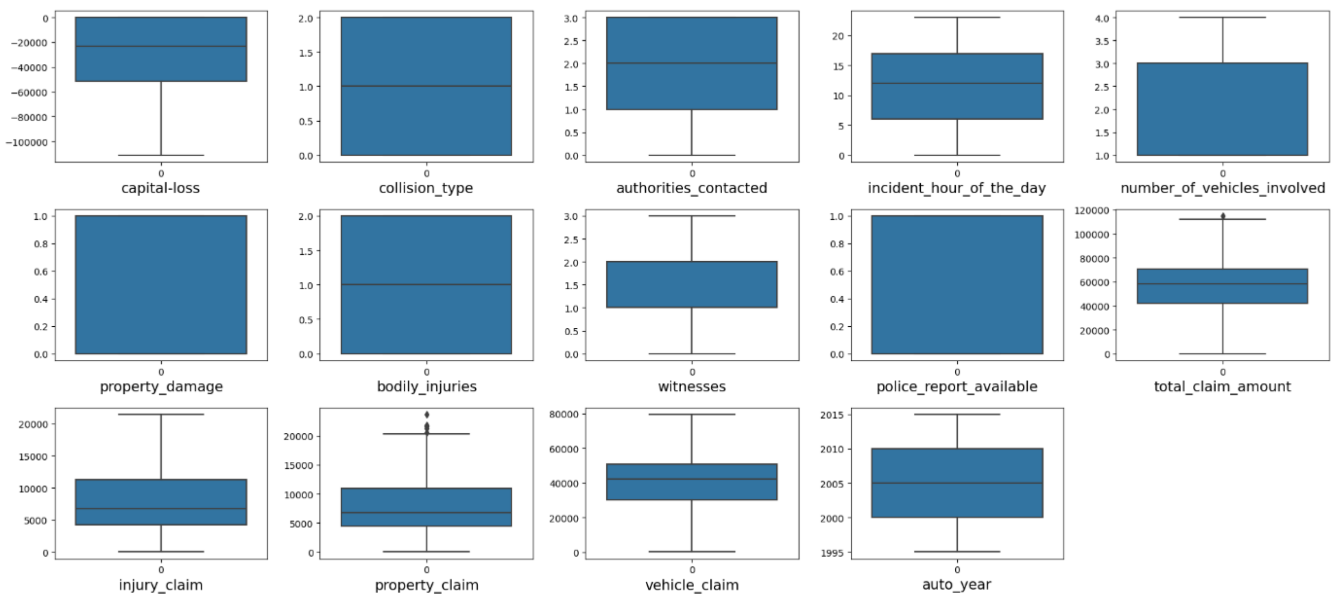
```
[81]: y=data.iloc[:,-1]
```

```
[82]: plt.figure(figsize = (25, 20))
      plotnumber = 1

      for col in x.columns:
          if plotnumber <= 24:
              ax = plt.subplot(5, 5, plotnumber)
              sns.distplot(x[col])
              plt.xlabel(col, fontsize = 15)

          plotnumber += 1

      plt.tight_layout()
      plt.show()
```










```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=12)
```

[85]:
```
x_train.head()
```

[85]:

red_sex	insured_hobbies	insured_relationship	capital-gains	...	number_of_vehicles_involved	property_damage	bodily_injuries	witnesses	police_report_available	total_claim_a
0	5	4	36100	...	3	0	2	3	0	
1	12	4	61900	...	1	1	0	0	0	
1	7	3	36900	...	1	0	2	3	1	
0	11	3	0	...	1	0	0	0	1	
0	15	0	0	...	3	0	1	3	0	

[86]:
```
x_test.head()
```

[86]:

	months_as_customer	age	policy_number	policy_deductable	policy_annual_premium	insured_zip	insured_sex	insured_hobbies	insured_relationship	capital-gains	...	r
518	196	41	246435	2000	1800.76	441499	1	4	2	0	...	
871	133	34	467841	500	1074.07	440833	0	3	0	70900	...	
797	136	33	804608	1000	855.14	458582	0	13	1	37900	...	
274	217	39	522506	2000	1399.85	605490	0	16	2	49900	...	
325	399	55	984948	2000	995.56	464665	1	17	1	0	...	

5 rows × 24 columns

[87]:
```
from imblearn.over_sampling import SMOTE
smt=SMOTE()
x_train,y_train=smt.fit_resample(x_train,y_train)
```

[88]:
```
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
x_train = scaler.fit_transform(x_train)
x_train= pd.DataFrame(x_train, columns =x.columns)
x_test=scaler.fit_transform(x_test)
x_test= pd.DataFrame(x_test, columns =x.columns)
```

[89]:
```
#Decision Tree
```

[90]:
```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
dtc=DecisionTreeClassifier()
dtc.fit(x_train,y_train)
y_pred=dtc.predict(x_test)
dtc_train_acc=accuracy_score(y_train,dtc.predict(x_train))
dtc_test_acc=accuracy_score(y_test,y_pred)
```

[91]:
```
y_pred
```

[91]:
```
array([1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 0, 1, 0, 0, 1, 0,
       1, 0, 1, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1, 1, 1, 0, 1, 0, 1, 1, 0, 1,
       0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 1, 1, 0, 0, 1,
       1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1, 0,
       1, 1, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1,
       0, 1, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 1, 0,
       0, 1, 0, 0, 1, 0, 0, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 0,
       0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 1, 0,
       0, 1, 1, 0, 1, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 1,
       0, 0])
```

[92]:
```
print(f"Training accuracy of Decision Tree is : {dtc_train_acc}")
print(f"Test accuracy of Decision Tree is : {dtc_test_acc}")

print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, y_pred))
```

```
Training accuracy of Decision Tree is : 1.0
Test accuracy of Decision Tree is : 0.58
[[88 64]
 [20 28]]
              precision    recall  f1-score   support

           0       0.81      0.58      0.68       152
           1       0.30      0.58      0.40        48

    accuracy                           0.58       200
   macro avg       0.56      0.58      0.54       200
weighted avg       0.69      0.58      0.61       200
```

[93]:
```
#RandomForest
```

[94]:
```
from sklearn.ensemble import RandomForestClassifier
rfc = RandomForestClassifier(criterion= 'entropy', max_depth= 10, max_features= 'sqrt', min_samples_leaf= 1, min_samples_split= 3, n_estimators= 140)
rfc.fit(x_train, y_train)
y_pred = rfc.predict(x_test)

from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
rand_clf_train_acc = accuracy_score(y_train, rfc.predict(x_train))
rand_clf_test_acc = accuracy_score(y_test, y_pred)

print(f"Training accuracy of Random Forest is : {rand_clf_train_acc}")
print(f"Test accuracy of Random Forest is : {rand_clf_test_acc}")
```

```
print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, y_pred))
```

```
Training accuracy of Random Forest is : 0.9900166389351082
Test accuracy of Random Forest is : 0.53
[[83 69]
 [25 23]]
              precision    recall  f1-score   support

           0       0.77      0.55      0.64       152
           1       0.25      0.48      0.33        48

    accuracy                           0.53       200
   macro avg       0.51      0.51      0.48       200
weighted avg       0.64      0.53      0.56       200
```

[95]:
```
#KNN
```

[96]:
```
from sklearn.neighbors import KNeighborsClassifier

knn = KNeighborsClassifier(n_neighbors = 30)
knn.fit(x_train, y_train)
y_pred = knn.predict(x_test)

from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
knn_train_acc = accuracy_score(y_train, knn.predict(x_train))
knn_test_acc = accuracy_score(y_test, y_pred)

print(f"Training accuracy of KNN is : {knn_train_acc}")
print(f"Test accuracy of KNN is : {knn_test_acc}")

print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, y_pred))
```

```
Training accuracy of KNN is : 0.6414309484193012
Test accuracy of KNN is : 0.385
[[ 36 116]
 [  7  41]]
              precision    recall  f1-score   support

           0       0.84      0.24      0.37       152
           1       0.26      0.85      0.40        48

    accuracy                           0.38       200
   macro avg       0.55      0.55      0.38       200
weighted avg       0.70      0.39      0.38       200
```

[103]:
```
#Naive Bayes
```

[104]:
```
from sklearn.naive_bayes import CategoricalNB,GaussianNB
from sklearn.metrics import confusion_matrix,accuracy_score,classification_report
```

[105]:
```
gnb=GaussianNB()
```

[106]:
```
model=gnb.fit(x_train,y_train)
y_pred=model.predict(x_test)
```

[107]:
```
gnb_train_acc = accuracy_score(y_train, gnb.predict(x_train))
gnb_test_acc = accuracy_score(y_test, y_pred)

print(f"Training accuracy of NaiveBayes is : {gnb_train_acc}")
print(f"Test accuracy of NaiveBayes is : {gnb_test_acc}")

print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, y_pred))
```

```
Training accuracy of NaiveBayes is : 0.6921797004991681
Test accuracy of NaiveBayes is : 0.45
[[59 93]
 [17 31]]
              precision    recall  f1-score   support

           0       0.78      0.39      0.52       152
           1       0.25      0.65      0.36        48

    accuracy                           0.45       200
   macro avg       0.51      0.52      0.44       200
weighted avg       0.65      0.45      0.48       200
```

[111]:
```
print('Decision Tree      :',100*dtc_train_acc)
print('Random Forest      :',100*(rand_clf_train_acc))
print('KNN                :',100*knn_train_acc)
print('LogisticRegression:',100*lg_train_acc)
print('Naive Bayes        :',100*gnb_train_acc)
print('Svm                :',100*(svc_train_acc))
```

```
Decision Tree     : 100.0
Random Forest     : 99.00166389351082
KNN               : 64.14309484193012
LogisticRegression: 69.13477537437605
Naive Bayes       : 69.21797004991681
Svm               : 89.26788685524126
```

[112]:
```
import pickle
```

[113]:
```
filename='dtc_model.pkl'
pickle.dump(dtc,open(filename,'wb'))
```

[114]:
```
import os
os.getcwd()
```