## DEPARTMENT of COMPUTER SCIENCE

Fraud Detection in Insurance Claims

By

## VEERANJANEYULU CHINTALAPUDI

**Declaration**

In submitting this work, I confirm that I am aware of and abiding by the University's expectations for proofreading. By NTU policy, I at this moment declare that the above- stated assessment is my work and that it has not previously been submitted for assessment to another University or for any other qualification

#### ACKNOWLEDGEMENTS

I would want to sincerely thank everyone who helped me finish this dissertation.

First of all, I want to express my gratitude to Dennis Monari my supervisor, for their accurate guidance, unwavering patience, and support during the study process. Their opinions and expertise on the topic greatly educated me and validated the methodology used in this paper.

In addition, I would like to thank the computer science teachers for their unwavering support and insightful criticism during my studies. I would also like to express my gratitude to [Other Important People, Professors, or Mentors] for their assistance and for giving the tools I needed to complete my research.

I owe a debt of gratitude to friends and co-workers who have supported me and given their wisdom. Their encouragement was essential to my drive and concentration. I would want to thank my family from the bottom of my heart for their resolute support and understanding throughout this difficult road. My greatest source of strength has been and continues to be their love and encouragement.

Contents

[Supervisor: Dennis Monari 1](#_bookmark0)

[CHAPTER 1 BACKGROUND AND STUDY ORIENTATION 8](#_bookmark1)

* 1. [AIM 8](#_bookmark2)
  2. [OBJECTIVES 8](#_bookmark3)
  3. [Project Scope 9](#_bookmark4)
  4. [Research Problem 11](#_bookmark6)
  5. [Research Design 11](#_bookmark7)
  6. [Data Analysis and Interpretation 12](#_bookmark8)
  7. [Summary, Recommendations, Reflections, and Conclusion 13](#_bookmark9)
  8. [Resources Required 13](#_bookmark10)
  9. [Project Risks 14](#_bookmark11)
  10. [Professional Issues 15](#_bookmark13)
  11. [Time Plan and Gantt Chart 16](#_bookmark14)

[CHAPTER 2 LITERATURE REVIEW 17](#_bookmark15)

* 1. [Introduction 17](#_bookmark16)
  2. [Related Work 17](#_bookmark17)
  3. [Data Quality 18](#_bookmark18)
  4. [Machine Learning 19](#_bookmark20)
  5. [Data Quality Assessment for Machine Learning 20](#_bookmark21)
  6. [Knowledge Gap 21](#_bookmark22)
  7. [Machine Learning Models for Data Quality Improvement 21](#_bookmark23)
  8. [Collection and Preparation of Data 23](#_bookmark24)
  9. [Data preprocessing 23](#_bookmark25)
  10. [Exploratory Data Analysis (EDA) 29](#_bookmark26)

[CHAPTER 4 TECHNICAL ANALYSIS AND IMPLEMENTATION 44](#_bookmark27)

* 1. [Data Exploration and Preprocessing 44](#_bookmark28)
  2. [Feature Engineering and Selection 44](#_bookmark29)
  3. [Model Execution and Assessment 45](#_bookmark30)
  4. [Hyperparameter Tuning Results 52](#_bookmark31)
  5. [Feature Importance Analysis 57](#_bookmark32)
  6. [Model Interpretability 58](#_bookmark33)
  7. [Challenges and Limitations 58](#_bookmark34)

[CHAPTER 5 SURVEY FINDINGS AND ANALYSIS 59](#_bookmark35)

* 1. [Dataset Overview 59](#_bookmark36)
  2. [Exploratory Data Analysis Findings 59](#_bookmark37)
  3. [Model Performance Analysis 60](#_bookmark38)
  4. [Model Comparison and Analysis 62](#_bookmark39)
  5. [Feature Importance Analysis 63](#_bookmark40)

[CHAPTER 6 DISCUSSION AND IMPLICATIONS 64](#_bookmark41)

* 1. [Model Performance and Selection 64](#_bookmark42)
  2. [Addressing Class Imbalance 64](#_bookmark43)
  3. [Feature Engineering and Selection 65](#_bookmark44)
  4. [Practical Implementation Considerations 65](#_bookmark45)

[CHAPTER 7 CONCLUSION AND FUTURE DIRECTIONS 67](#_bookmark46)

* 1. [Summary of Findings 67](#_bookmark47)
  2. [Limitations of the Study 67](#_bookmark48)
  3. [Future Research Directions 67](#_bookmark49)
  4. [Concluding Remarks 68](#_bookmark50)

1. [REFERENCES 69](#_bookmark51)
2. [APPENDICES 70](#_bookmark52)

## List of Figures

[*Figure 1: Research Process* 10](#_bookmark5)

[Figure 2: Data Quality 19](#_bookmark19)

## List of Tables

[**Table 1: Project Risks** 14](#_bookmark12)

# CHAPTER 1 BACKGROUND AND STUDY ORIENTATION

Concerning the nature of today's data-driven environment, quality input data forms a direct proportion of the effectiveness of techniques in machine learning. Since these algorithms find application in many sectors, from finance to health and manufacturing, the accuracy of input data defines the reliability associated with organization decisions. Ensuring data integrity has remained elusive despite the potential insights otherwise obtainable through data analysis. It is easy to see that complex patterns are bound to turn up with machine learning algorithms, but they are only as good as the data processed. Poor quality data can manifest by missing, incorrect, or conflicting values, introduce bias, spread misinformation, and generally undermine the reliability of outcomes. Data quality significantly impacts how successfully predictive models work in different sectors. For instance, inaccurate or missing information about hospital patients might result in incorrect predictions, leading to misallocation of resources and adversely affecting patient care. Poor data quality interferes with the immediate decision-making process and disrupts broader organizational activities. It is a problem for risk assessment in finance and inventory management in retail competitiveness is undermined.

In this regard, organizations should roll out well-developed data governance frameworks with particular attention to data collection, standardization, and validation. Advanced data cleaning methods, such as implementing imputation algorithms and outlier detection, can minimize the effects of low-quality data. Expert domain knowledge and human supervision can be applied to improve data accuracy. A culture of data stewardship is required, and regular reviews are carried out to confirm data integrity. An organization can make informed decisions, find valuable insights, and foster sustainable growth in data quality in a data-driven environment (Braig et al., 2023).

### AIM

The research will investigate the influence of data quality on the decisions and machine learning results.

### OBJECTIVES

* + 1. Study the implications of data quality in the process and the role of machine learning in decision-making.
    2. This likely includes developing a scenario in which input data, machine learning, and decision outputs can be traced, focusing on the risks imposed by low-quality data.
    3. Perform sensitivity analysis using, for example, complex-step sensitivity analysis- based feature selection to estimate impact.
    4. Point out ways to suggest how having the minimum data quality can be established for effective decision-making.
    5. Test the adequacy of the approach you have suggested
    6. Offer research results with vivid examples and visualizations.

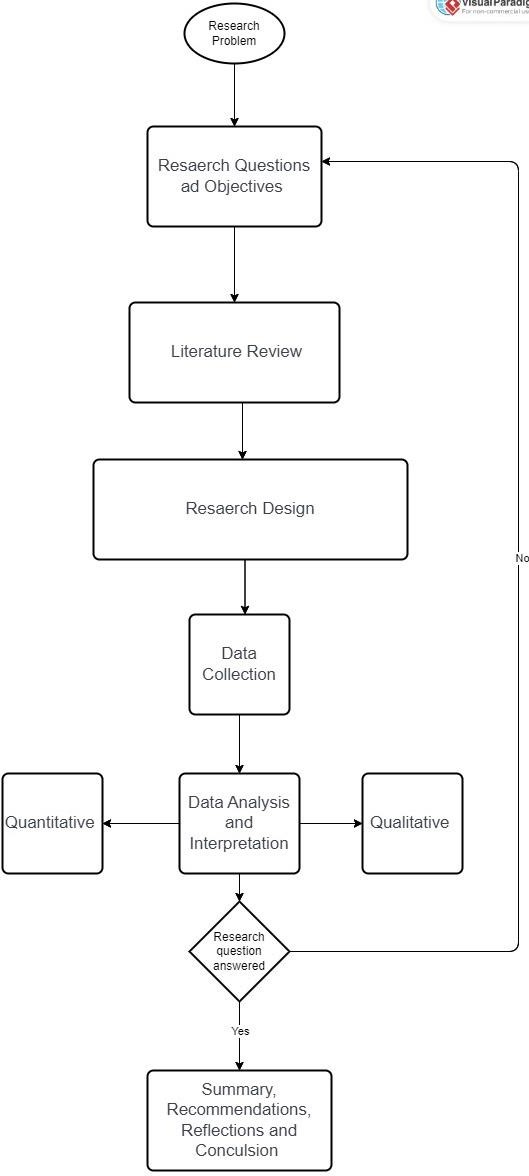
### Project Scope

The project focuses on the quality of training data from Twitter that could feed into machine learning models. Some of the key aspects would be:

1. Acquire, label, and prepare the data to avoid future problems.
2. Organizing the human workforce, ML teams, and labeling tools brings some challenges to outsourcing or crowdsourcing.
3. It uses Twitter's ML platform and the DeepBird framework for training and deployment.

Finally, effective filters should be designed to deal with the noise, user behavior, and content of the tweets to facilitate the dataset preparation.

1. Applying ML and geolocation to urban planning via the creation of a gazetteer/truth dataset.
2. Apply machine learning to detect symptom self-reporting on Twitter, targeting organic conversations and trends.



*Figure 1: Research Process*

Source: self-made

### Research Problem

A fundamental level at which the research process commences is identifying a research problem. This step reveals an area of interest that will shape the rest of the study. Formulating a research problem is important since it guides the formation of research questions and the exploration of issues that are relevant to the research. The problem has to be clear, appropriate, and achievable, thus setting ground for further investigation.

##### Formulation of Research Questions and Objectives

In developing research questions or hypotheses, one should begin with the identification of a research problem. These questions are entailed within the problem statement and hence depend on it. They must be specific, clear, and investigable through data collection and analysis. This phase also includes the statement of purpose and significance of the study, which clarifies how it enhances existing knowledge and its possible implications in practice, policy, or theory.

##### Undertaking a Literature Review

The literature review will place this study within the existing body of work and, as such, must be a systematic review and synthesis of prior research on the topic. It helps understand the existing state of knowledge, identify gaps, create a conceptual framework, and inspire research questions. A good literature review forms the basis of good study design and methodology, data collection, and analysis. It consists of clearly defining the scope of the review, searching relevant literature, evaluating and selecting analyzed sources, arranging and categorizing literature, analyzing and synthesizing the findings, developing a conceptual framework, and writing a comprehensive review.

### Research Design

Research design is a schematic design to guide the study in such a way that the methodology, data collection methods, and procedures effectively respond to the research problem. It can be seen as a principal blueprint for collecting and analyzing quality data required to answer research questions and test hypotheses. In simple terms, a good research study design assures that findings will maintain consistency with high reliability and validity, making research results more credible. It consists of important components like the selection of research methods qualitative, quantitative, or mixed methods sample design, tools of data collection, and procedures of data analysis. A research design should be most appropriate to the study objectives, research questions, and feasible conditions. A good research design is essential to get satisfactory results and can be able to conclude. Some of the specifications of a research design are discussed below.

##### Data Collection

Data collection is an important step in the research process, in which data is systematically obtained to answer the research questions or test hypotheses. The data collection method will be different according to the design of the study, using qualitative, quantitative, or mixed methods. Qualitative methods, like interviews, focus groups, and observations, extend insightful information about participants' experiences and views. Quantitative methods are based on numerical data collected through surveys, experimentation, and analysis of secondary data to conduct statistical analyses, which establish patterns and relationships. Data collection must be planned for, allowing accuracy, reliability, and validity. At this stage, the right tools will be chosen, the data collectors will be trained, and procedures will be standardized. Ethical considerations, such as informed consent and confidentiality, are set up in respect of participants' rights and to ensure data integrity. Properly planned and executed data collection leads to robust data supporting the objectives set for the study and offers some valuable contributions to the literature (Braig et al., 2023).

### Data Analysis and Interpretation

Data analysis and interpretation are core steps that help transform the data into useful insights and conclusion-oriented action. How the data will be analyzed will depend on the form of research design, qualitative, quantitative, or mixed methods.

Quantitative data: Analysis techniques all involve numerical data and, in essence, represent ways to conduct statistical analysis. Descriptive statistics summarize fundamental data characteristics, while inferential statistics examine relationships, differences, and causal links between variables. Advanced statistical software, including SPSS, R, or Python, supports most of these analyses.

Qualitative data: Analysis is about identifying patterns, themes, and insights from non- numerical data, including interview transcripts, observations, and open-ended responses to a survey. Such techniques as coding, thematic analysis, and narrative analysis organize and interpret data systematically. Tools such as NVivo or Atlas. Ti can facilitate organizing and analyzing large qualitative data to find deeper insights.

Mixed Methods Analysis: This method incorporates both qualitative and quantitative data analysis to build an overall understanding by integrating numerical trends with detailed contextual insights. In the process, independent analysis of each type of data is done, followed by the integration of findings to build comprehensive conclusions.

##### Interpretation

This means the analyzed data makes sense for the research questions and objectives. This implies carefully inspecting the findings to understand their meaning, importance, and relevance. While interpreting, a researcher needs to consider the context, limitations, and possible biases in the data. Comparing findings with existing literature situates those findings within a larger discourse of the study, particularly its contribution and areas for further research. Therefore, effectively analyzing and interpreting data in a manner in which valid and reliable conclusions are derived to inform practice, policy, and theoretical understanding is essential.

### Summary, Recommendations, Reflections, and Conclusion

This paper aims to systematically identify research problems, formulate questions, review the literature, collect data, and analyze the results of a sound research design that has consistency and validity. For any future research, a difficult literature review must be stressed in identifying gaps to permit their argument or points to build on previous knowledge. Mixed methods give many insightful conclusions. The methods of collecting and analyzing data are to be continuously found and improved in a mission for continued accuracy and reliability. This phase showed that meticulous planning is required to consider ethical factors while doing a good piece of research. Severe methodology and good ethics were ways to overcome data integrity and participant interest. A well-designed piece of research underpins effective design, collection, and analysis. Provided that proper systematic processes have been followed and ethical standards maintained, meaningful contributions that advance knowledge and inform practice across a wide range of fields can be achieved.

### Resources Required

Several critical resources will be needed to ensure the completion of this project successfully, including:

Datasets: These should include data quality variations to enable the development of scenarios and sensitivity analysis that needs to happen before proposal development.

Machine Learning Tools: These will enable one to implement such algorithms with libraries like sci-kit-learn, Tensor Flow, or PyTorch and understand how decisions will turn out.

Literature Resources: This will yield relevant academic journals and industry reports that would help provide the theoretical and experiential bases for analysis.

Computational Resources: It is also essential to provide a suitable hardware and software environment for efficient simulations, algorithm core executions, and processing of large datasets.

### Project Risks

Some of the risks include poor data quality, impacting the results' reliability. Other risks are technical challenges to the application of machine learning algorithms, sensitivity analyses, and large dataset management. The time can be too short to allow detailed analyses the scope of the project can be limited by the restricted availability of computational resources or expertise. There will also be careful management concerning ethical and legal compliance to avoid reputational or legal consequences. So, proper planning for mitigating these risks depends on effective stakeholder engagement.

***Table 1: Project Risks***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Risk Descrip tion | Status | Probabi lity | Impact | Ratin g | Risk Cate gory | Mitigation Plan |
| 1 | Data Adequa cy and Reliabili ty | Ongoing | High | High | High | Data Qual ity | Ensure that appropriate data validation, quality checking, and verification mechanisms are in place for your data. |
| 2 | Technic al Difficulti es | Ongoing | Mediu m | High | Mediu m | Tech nical | Always test the machine learning algorithm through sensitivity analyses, checking for compatibility with their software tool. Improve on scalability issues and fix algorithm errors early. |
| 3 | Time Restrict ions | Ongoing | High | Mediu m | High | Sch edul e | Create a detailed timeline for your project, distribute resources efficiently, and prioritize tasks accordingly to meet the deadlines. You can always outsource if needed; there is always scope for enabling time-constrained tasks. |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 4 | Resour ce Constra ints | Ongoing | Mediu m | Mediu m | Mediu m | Res ourc e | If computational resources are inadequate, consult experts in the area and seek cost- effective solutions to budgetary limitations. |
| 5 | Ethical and Legal Compli ance | Ongoing | Mediu m | High | Mediu m | Com plian ce | Ethical guidelines regarding actions should be established and abide by legal standards. From time to time, the project's activities should align with the law. Ensure legal experts are present to avoid possible legal risks and protect the organization's reputation. |

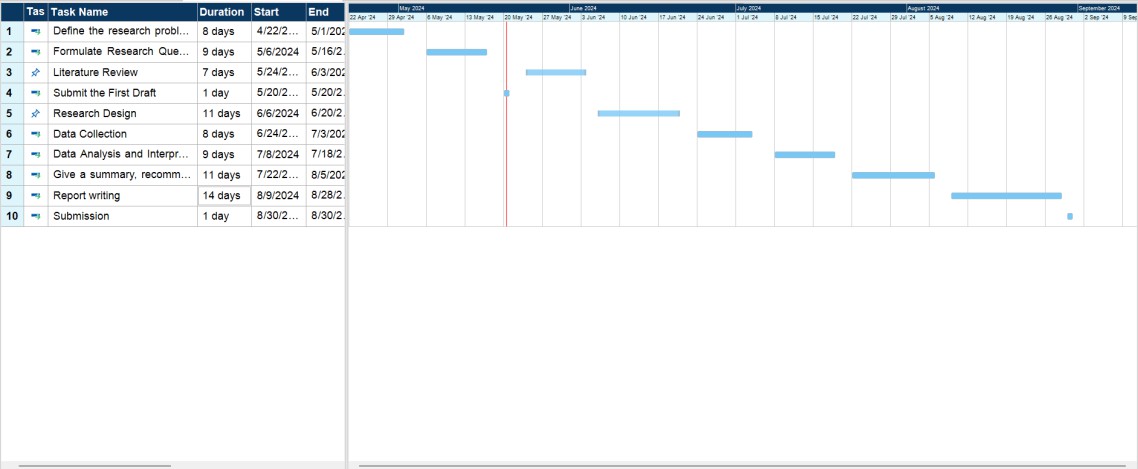
### Professional Issues

Various professional considerations should be inculcated throughout the project to ensure ethical behavior and adherence to field norms. Principal points of ethical concern would be data protection, confidentiality, and informed consent. Ensuring the protection of people's rights and privacy within datasets involves making acceptable anonymization and security measures to prohibit access to sensitive information through unauthorized means. It should be accompanied by transparency and truthfulness in presenting results and admitting limitations to ensure credibility.

The core professional values also incorporate accountability and responsibility. These require high standards of integrity and professionalism, including taking responsibility for the decisions made, acknowledging mistakes, and dealing with conflicts of interest openly. Proper and effective communication supports teamwork in a culture of accountability and mutual respect.

Since machine learning and data science evolve very fast, seeking constant learning and development in these areas is essential. Team members must continuously educate and train themselves on emerging technologies and best practices. Besides, diversity, equity, and inclusion foster creativity and robust problem-solving capabilities for better project results. An inclusive environment enhances belonging and power, boosting morale and productivity.

### Time Plan and Gantt Chart



# CHAPTER 2 LITERATURE REVIEW

### Introduction

In light of the project's goal to gather a general understanding of the interaction among data quality, machine learning approaches, and decision-making processes, the chapter analyses some of the prior research. To gain understanding of approaches, strategies, and best practices for tackling problems with data quality across domains, the literature will be examined. A first step towards identifying possibilities, gaps, and workable solutions for enhancing data quality in the context of machine learning applications is this review. Additionally, it synthesises previous research findings to facilitate the reuse of information and the development of pre-existing frameworks, which facilitates the creation of robust techniques tailored to the particular requirements and difficulties of our target area. The understanding and application of data quality enhancement approaches in machine learning systems will advance with the aid of such a thorough review (Jain et al., 2020).

### Related Work

According to Jain et al. (2020), data quality is significant for the performance of any machine learning model because the result of these ML techniques depends directly on the quality of the input provided by the users. In the second instance, despite increased usage of AI and ML technologies in varied industries, there lies almost a dearth of studies about deep analysis of the data challenges and quality problems before feeding the data into an ML pipeline. Poor quality data caused by issues like incorrect labels, close categories of a categorical variable, or inconsistency in column to column can lead to poor performance of the model. Jain et al. (2020) have also raised the argument related to the quality of data, which lies in its relevance to the purpose intended to serve for the model in the real world. Teams cannot complete their tasks even given a comprehensive, diverse, and organised dataset if they are unable to make predictions from it. Haleem et al. (2021) discovered in a project creating an electronic health record platform that, despite having information identifying physicians who used the platform, they were unable to predict when a client was going to stop using the service. This occurred because the practice managers, who were not direct platform users, took the decision to switch services, and their actions were not observed (Haleem et al., 2021).

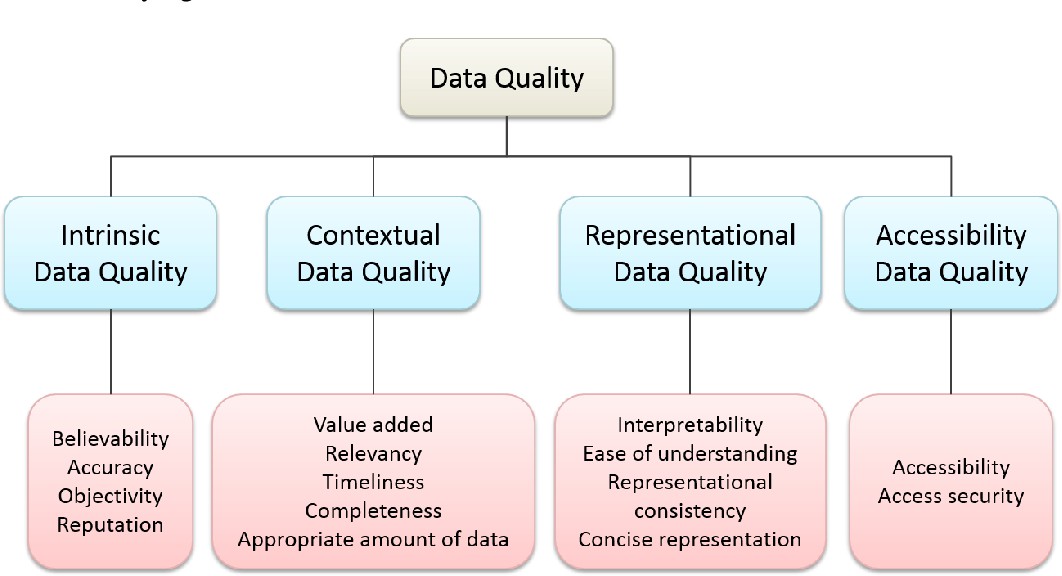
Research into data quality and its impact on the performance of machine learning has been conducted. This research clearly shows that incomplete, inaccurate, or unsuitable training datasets will undoubtedly lead to unreliable models, making bad decisions. Any machine learning model requires complete precision, inclusivity, coherence, evenness, and so on in both the training and test datasets. The reason is that the mix between data quality and quantity must be appropriate in an ML model since data is the fuel of ML models. At the same

time, data amount and quality seem very complex. Cortes, Jackel, and Chiang, 2020, however, argue that data quality is more important than quantity. It is cheaper and more efficient to resolve the data quality issues than to collect new data. All techniques in data increase, such as cropping, flipping, rotating, or adding noise, can increase the quantity and diversity of the data without altering the quality. Data sampling techniques, such as stratified sampling, bootstrapping, or synthetic data generation, are tools that make it possible to reduce the amount and complexity of data without decreasing the quality (Gupta et al., 2021).

### Data Quality

According to Badr, one can define quality as "fitness for use." With an exact analogy, "data quality" might be explained as the data most fitting for the intended consumers only. In machine learning, many researchers and practitioners have placed a great deal of emphasis on high-quality data. Data quality measures are usually developed in a manner, and the needs and expectations of stakeholders govern the whole evaluation process. Bad data quality bears a considerable social and economic cost and involves severe financial losses. Literature on quality stresses that quality evaluation depends on the consumer's decision to utilize the product. In the same way, data quality evaluation depends on data consumers' preferences and needs. This is important because the contemporary consumer has a high level of control over their computing environment and the kind of data applied (Gupta et al., 2021). Bad data quality dispirits employee morale, encourages and puts trust into disrepute within an organization, and further complicates project management. Poor quality data shows reduced customer satisfaction, increased costs, reduced job satisfaction at employees at the operational level, and difficulty with reengineering efforts at the tactical level. The root causes of poor data quality go a long way in contributing towards an "information ecology" far from being supportive of the informational age. Mentions about poor data quality have risen over the last few years through multiple literatures, social media platforms, and other reputable publications, showcasing its broad impact (Demirbaga, 2023).

In this respect, different data quality evaluation frameworks have been proposed from various perspectives, serving different purposes and targeting different data structures. Gupta et al. were the first to propose nine critical quality dimensions concerning the decision-making of the data consumer. The dimensions were organized into four categories: accessible, interpretable, practical, and believable. In this way, the first well-rounded framework for evaluating data quality was instituted. This first contribution proposed a more complete DQ framework with four dimensions and 15 measurements.



*Figure 2: Data Quality*

(Source: Demirbaga, 2023)

This empirical drive improved the framework's applicability to the real world and precision. Recently, the proposed DQ dimensions and measurements have been extensively used and changed, adapting the framework to specific needs and contexts. These modifications keep the DQ framework relevant and effective for various applications, reflecting an evolving nature of data quality assessment.

### Machine Learning

The main component of every machine learning system is data. One of the key ideas in artificial intelligence is machine learning, which has drawn a lot of interest in computer science. Artificial intelligence and machine learning have advanced significantly in the last several years, finding numerous uses in the defence and government industries. Machine learning leverages example data and prior experiences to teach computer systems how to perform tasks without explicit instructions. In 1959, Arthur Samuel made the explicit claim that robots might learn without explicit programming In this, data scientists train machine learning algorithms with specific datasets regarding a problem, allowing the algorithms to solve complex tasks (Samuel, 1959). Such systems, trained with high-quality data, could give tasks answers not by pure intuition but based on learned patterns; they would adjust to new situations with updated data. Our study places value on training machine learning algorithms using quality data to give insights at the very beginning to decision-makers for whom optimal performance is expected from the system. This study preoccupies the minds of machine

learning users with the critical task of selecting the correct data for training in order to attain superior outcomes (Demirbaga, 2023).

### Data Quality Assessment for Machine Learning

The usefulness of data for meeting user goals or achieving an intended goal that verges on developing a machine learning system is characterised in this study as data quality. Assuring the quality of machine learning-based systems requires data quality assessment since high- quality data is a major factor in ML/AI system performance, fairness, robustness, safety, and scalability. Various scholars have examined data quality for machine learning, including problems, dimensions, verification, and enhancement. (Saeed et al., 2022) investigated various methods for identifying data quality problems, particularly errors and missing values in data mining and machine learning. Class noise is mislabeled data in imbalanced datasets, and research has shown that class noise does indeed impact machine learning algorithms, with class noise in the majority class being more harmful than in the minority class. The research examines two ML-based network intrusion detection systems, namely C4.5 and a naive Bayes algorithm, over label errors ranging from low to high rates. Experiments showed that both algorithms generally remained strong when label errors increased, although the accuracy of C4.5 remained good while that of naive Bayes gradually decreased. In contrast, for unsupervised learning methods such as K-means, the majority accuracy remained high even when the mislabelled rates were high. This result indicates that high accuracies can be obtained from robust classifiers with some level of acceptance of error. However, unsupervised learning should always be the preference rather than acting upon poor-quality data (Saeed et al., 2022).

Another critical dimension in machine learning related to data quality is trustworthiness, which covers reliability, credibility, and reputation. Trustworthiness refers to the truth of the factual information derived from a source. Reputable sources are those that include few errors in their content. In order to jointly evaluate fact correctness and source accuracy, Dong et al. (2015) proposed a knowledge-based trust approach that involves gathering information from several sources, initialising source accuracy with an authoritative estimate of the same, and drawing probabilistic conclusions. Duplication, correlated variables, outliers, bias in data sources, inconsistency, obsolete data, incompleteness, insufficient training data, and overlap in training, validation, and test datasets are some data quality issues that have been found and explored. Given the growing centrality of machine learning, this paper presents a holistic approach toward data quality management, going beyond cleaning and transformations to multiple facets of data quality (Gupta et al., 2021).

### Knowledge Gap

Upon reviewing the articles that were demonstrated to be available through various databases on data quality models for machine learning, we observed that existing research has neither been merged nor specifically described data quality models targeted to machine learning applications. While many techniques can be outdone for increasing data quality over instance selection, discretization, and feature engineering, this still can be too costly for executing data- intensive machine-learning tasks. All these data-quality models have been proposed without clear empirical evidence that indicates their relevance in the context of machine-learning projects. There is a lack of a data quality model regarding machine learning setups and the present research. The model developed will be utilized to ease data quality in machine learning systems and is expected to bear cost efficiencies compared to alternative remedies. The model implemented will also be helpful to encourage and facilitate attractive pricing strategies for the data offered (Demirbaga, 2023).

Given that the literature is really poor in this area of research, our study aims to determine the relevant data quality characteristics related to machine learning, taking into consideration the insights gained from interviews. Our project's success is attached to the design and implementation of the suggested Data Quality model. It is a model that fast-tracks the choice of data for the data scientist in identifying and filtering out relevant data needed for training the machine learning system. The model helps data scientists predict the quality of datasets by using data quality attributes, which later aids modification to meet the standards. Ultimately, this provides a data quality metric for machine learning to the data scientist, enabling him to make informative decisions in his workflows regarding already existing data. Using this model, data scientists can institute increased effectiveness and reliability in their machine-learning projects, driving better outcomes and progress in innovation (Demirbaga, 2023).

### Machine Learning Models for Data Quality Improvement

One common problem when dealing with domain-specific tasks is low training data resources. This is often because labeling is expensive. Many data augmentation strategies address this.

Semi-supervised learning: These methods of SSL utilize unlabelled data to improve classification. In co-training, multiple classifiers refine each other's predictions in pseudo- labeling and add confident predictions from a single classifier trained on labeled and unlabelled data. Recently, neural SSL with pre-trained language models has been applied to text classification.

Active Learning (AL) optimizes labeling by selecting the most informative data points using methods like uncertainty sampling and Bayesian methods. This reduces the need for

extensive human annotations and applies fairly effectively to traditional and neural models alike.

Transfer Learning (TL): TL can enhance performance through pre-trained models on specific tasks. For instance, Pre-trained uses of models like AlexNet and BERT as transfer learning in numerous applications such as biomedicine image and text classification (Zhuang et al., 2021).

Few-Shot Learning (FSL): Prior knowledge harnesses the ability to learn new tasks with minimal labeled data. The methods of sample transformation and the use of large language models for generalization from little data are very promising. Despite the progress made, optimizing FSL is an active area of research.

## CHAPTER 3 RESEARCH DESIGN and METHODOLOGY

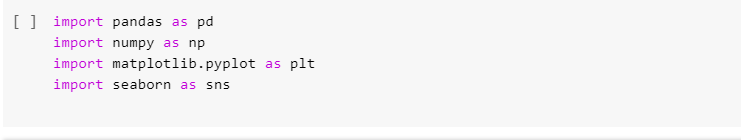
This quantitative research methodology applies statistical and machine learning techniques on a large database of insurance claims. The main steps in the process are as follows

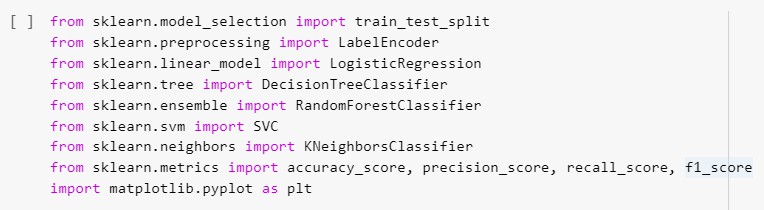
### Collection and Preparation of Data

The dataset utilized here contains data from one thousand motor insurance claims, including policyholder names, incident details, and claim amounts associated with the incidents. This data collection was obtained from an open source and underwent a thorough preprocessing process to ensure its consistency and quality. Preprocessing procedures specifically included addressing missing values, data cleansing, and discrepancies. Additionally, data transformation was required, which included encoding date/time variables and categorical categories into more practical formats. To increase this dataset's prediction power, more feature engineering was done to provide new variables and interaction terms. To avoid any one feature having an undue impact on the analysis, numerical features were standardized. Then, a training set, a validation set, and a test set were created from this data set. This made it possible to evaluate the model's performance on unseen data in a trustworthy manner. The dataset was carefully prepared to allow for the highest level of correctness and robustness in the analysis, as well as to yield trustworthy and useful insights. (Agarwal, 2023).

### Data preprocessing

This little piece of code imports Python libraries for machine learning and data analysis. The following is what the libraries do: Pandas and NumPy handle data and numerical operations; Matplotlib and Seaborn are used for plotting; Scikit-learn is used to implement a model with data splitting and encoding; and a variety of classifier models are implemented for SVC, K- Nearest Neighbors, Random Forest, Decision Tree, and Logistic Regression, along with functions that yield metric values for evaluating the performance of the model.





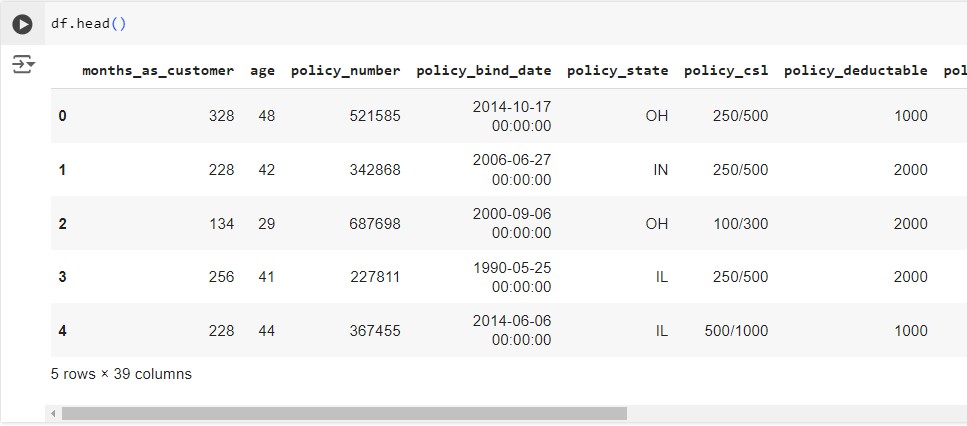
##### Importing CSV Data

This code reads `fraud\_insurance\_claims.csv}, a CSV file located in the named path ('/content/'), into a Pandas DataFrame called `df}. Now that the data from this CSV file is in that DataFrame, {df}, any further analysis and modification may be performed there.



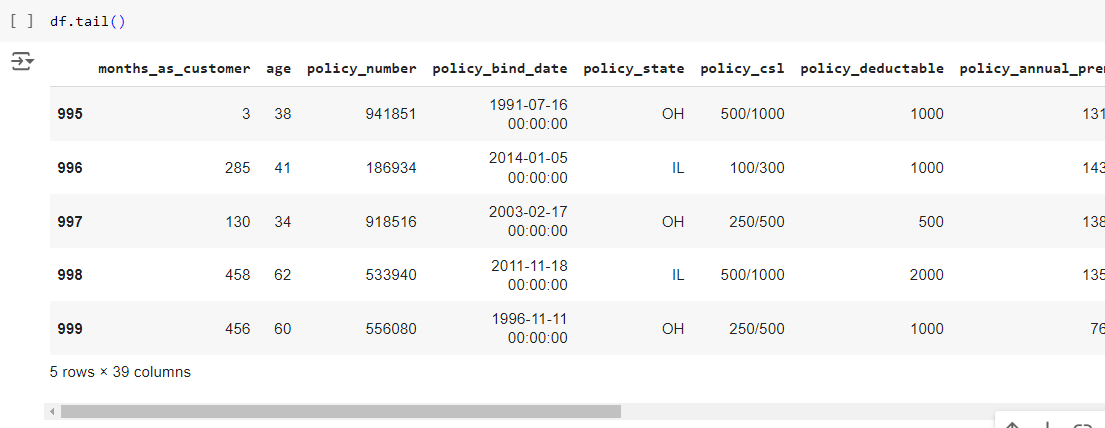
##### Display the First Few Rows of DataFrame

DataFrame ‘df’ shows the top five rows when the `df.head()` function is used. The column names and first data entries are helpful for structural inspection. This provides information about the appearance and types of values found in a dataset.



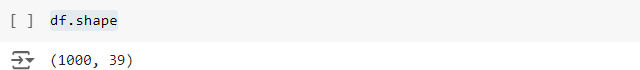
##### Displaying the Last Few Rows of the DataFrame

The final five rows of the DataFrame ‘df’ are shown via the `df.tail()` function. This is a very helpful way to ensure that the final items in the dataset match expectations and that the information has been properly loaded when working with large datasets.



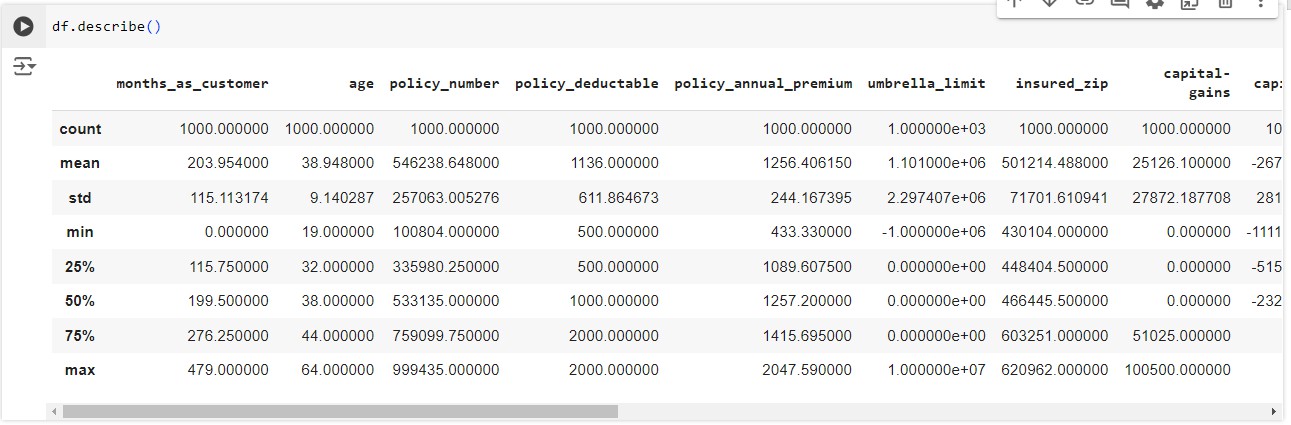
##### The number of dimensions in the DataFrame

The dimensions of the DataFrame ‘df’ are returned by the attribute df.shape. As an illustration, this is (1000, 39). There are 39 columns and 1000 rows. This data provides insight into the dataset's size, characteristics, and quantity of records or rows.



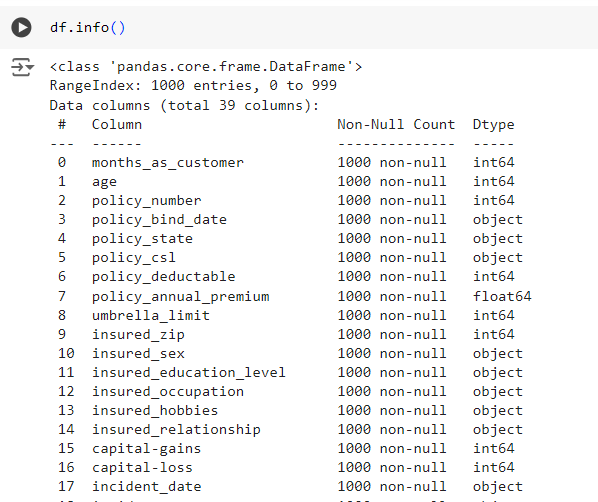
An analysis of the DataFrame's statistics

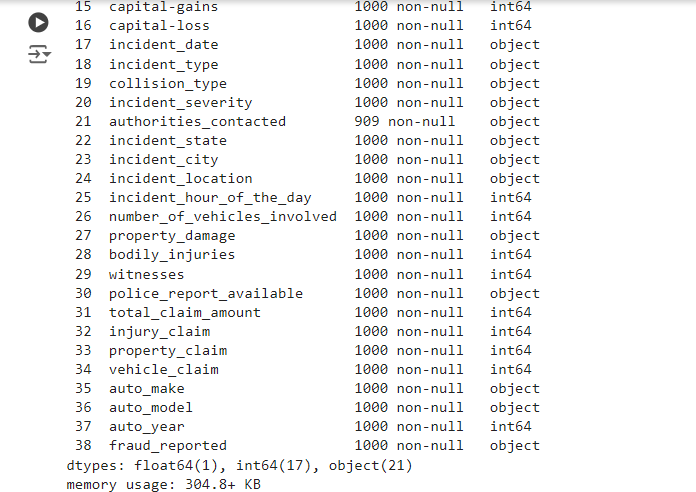
The DataFrame ‘df’ receives statistics from the `df.describe()` function. According to the data, ages range from 19 to 64 years old, and ‘months\_as\_customer’ averages almost 204 months. The range of policy numbers is 100,804 to 999,435; the average annual premium is 1,256.41, and the average deductible is 1,136. The ‘umbrella\_limit’ has a mean value of 1,101,00, however it is highly variable. There are 25,126.10 in capital profits and roughly 26,793.70 in capital losses. The mean total claim amount is 52,761.94, with total claim amounts ranging from 100 to 114,920.



##### Summary of DataFrame

It gives back the column dtypes and DataFrame size. The `df.info()` function yields the information that this DataFrame {df} contains 39 columns and 1,000 entries. There are 1,000 non-null items in each of the other columns but not in the column `authorities\_contacted}. For integer columns, the data types are {int64}; for floating-point columns, they are `float64}; and for categorical columns, they are `object}. The DataFrame uses roughly 304.8 KB of RAM.

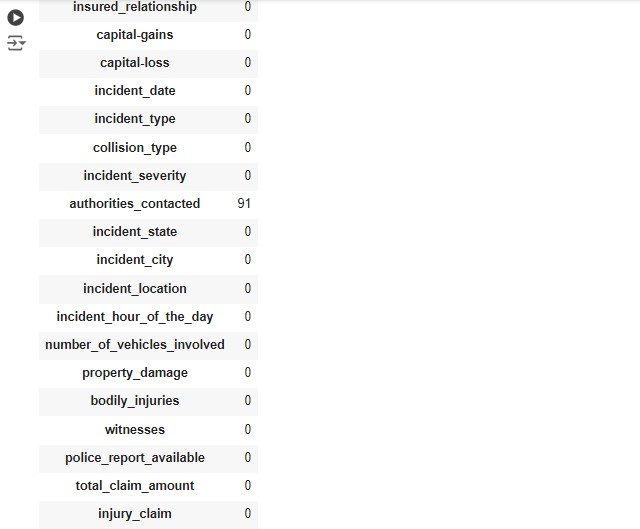




##### Missing Values Summary

The output of df.isnull().Therefore, sum() is taken to indicate that, with the exception of the column "authorities\_contacted," which has 91 missing entries, the majority of the DataFrame df do not include missing values. There are 1,000 non-null entries in each of the other columns, including "age," "months\_as\_customer," "policy\_number," and others. This overview highlights the existence of missing data in one column and provides an overview of the dataset's completeness.

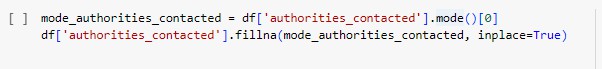




##### Handling `authorities\_contacted` Missing Values

To address the missing values in the `authorities\_contacted’ column, first use

`df['authorities\_contacted'].mode()[0]’ to calculate the mode from this column. Next, the mode used ‘df['authorities\_contacted']’ to fill up the missing entries.fillna(inplace=True, mode\_authorities\_contacted)’. This will guarantee that, in order to preserve the consistency of the dataset, all missing values in the column `authorities\_contacted’ are replaced with the value that occurs the most frequently.



### Exploratory Data Analysis (EDA)

To do this, we conducted a comprehensive exploratory analysis of the dataset in an effort to highlight the various variable distributions and any correlations that might exist, as well as to

potentially spot any patterns that might indicate that a claim of this kind is fake. The following actions were necessary for this:

* + - For numerical variables, descriptive statistics
    - Analyzing frequencies in categorical variables
    - Charts and plots for important relationships
    - Testing for correlations between features

##### The Distribution of Incident Types

This graphic uses a donut chart to show how the various incident types are distributed throughout the dataset. Using `value\_counts()`, it first determines the quantities of each type of unique incident number inside the column ‘incident\_type’. Then, using these counts, a pie chart is created, with the different slices representing the various sorts of incidents.

**Coloring and Design:** The Seaborn palette's soft pastel is used to color the slices. This actually helps to discern between the various types of situations and makes it friendlier on the eyes. To best arrange the chart visually, the slices begin at a 140-degree angle. To improve separation and clarity, a white border surrounds each slice.

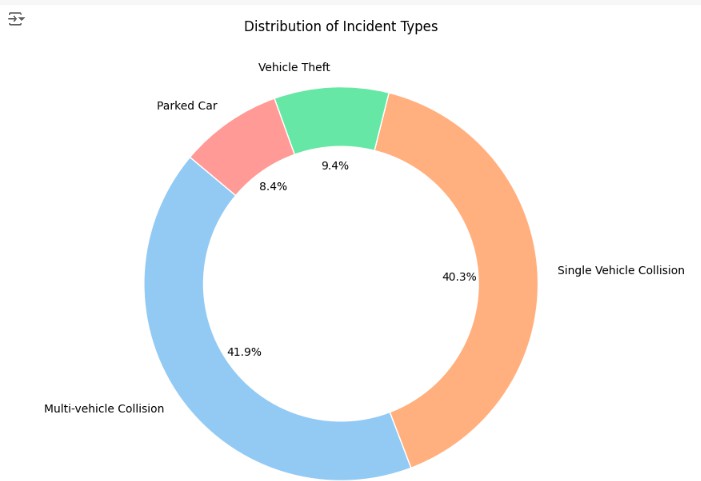
**Donut effect:** To achieve the donut effect, a white circle is essentially added in the center of this pie chart. This not only looks nice, but also draws attention to the way the slices are portioned out around the circle.

**Labels and Percentages**: The percentage of each event type's total, formatted to one decimal place, is shown immediately on the slice that corresponds to it. This makes it possible for viewers to quickly understand the proportional proportion of each kind of incidence.

**Title:** The title of the chart, "Distribution of Incident Types," gives the reader instant context and clarifies that the visualization displays the various event categories that are included in the dataset.

This style is simple to grasp and provides a quick summary of all types of incidents found in the dataset. The donut chart illustrates this. This makes it possible to quickly determine which kind of incident happens most frequently or least frequently.





##### Distribution of Incident Seriousness

The pie chart below shows the distribution of event severity. It displays the frequency with which each distinct value in the 'incident\_severity' column appears in the dataset.

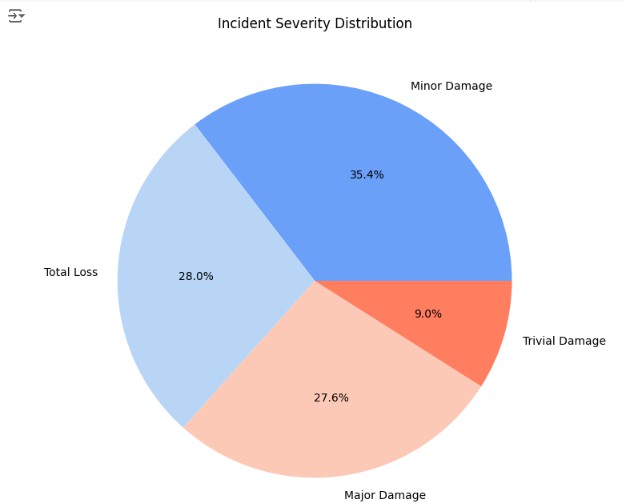
**Coloring and design**: Color-coded in the cool Seaborn "coolwarm" color scheme. Not only does this gradient seem good, but it may also be a faint indication of a gradient with similar severity levels.

**Size of Slice and Percentages:** The frequency of that severity level in the dataset is directly displayed to indicate the slice size. The chart prominently displays the proportion of the total that each slice represents, making it simple to compare the severity levels.

**Headline and Labels:** I named this "Incident Severity Distribution" in order to help the user understand what this chart shows. I took out the y-axis label to make it more organized and tidy. Axis labels are superfluous for thoroughly cleaned pie charts.

This pie chart is a straightforward but incredibly powerful method of visualizing the distributed severity levels in the dataset, immediately identifying the most common severity level.





##### Comparing Capital Gains and Capital Losses

When aligned once more, a comparable scatter plot facilitates the identification of the correlation between the capital gains and losses noted among the dataset's records. The data points can be found based on the reported fraud.

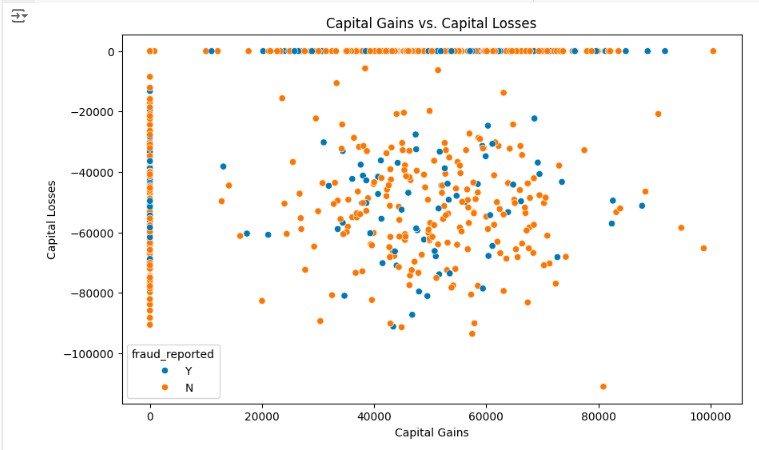
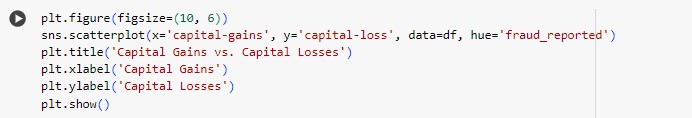
**Axes and Data Points**: The capital gains are shown by the {x-axis}, while the capital losses are shown by the `y-axis}. Each point on the plot represents a single dataset record, and each point's position is determined by the capital gains and losses associated with that record.

**Coloring (Hue):** Point coloring is applied in relation to the `fraud\_reported} column depending on whether or not the incident is considered fraudulent. The variation in color tones between the spots makes it easier for the observer to quickly determine whether pattern or cluster the financial activity, gain, or loss indicates is related to the occurrence of fraud.

**Interpretation:** It could identify patterns in the scatter plot, such as bigger capital gains or losses in some areas, which would change the likelihood of fraud. It might be useful in figuring out if the data contains any anomalies or outliers.

**X and Y Labels:** The chart is labeled "Capital Gains vs. Capital Losses," with "Capital Gains" on the `x-axis} and "Capital Losses" on the `y-axis}. The labeling facilitates the comprehension of the data by providing a clear description of what the viewers are viewing.

More significantly, this scatter plot is a powerful tool for identifying correlations between financial variables and fraud that can be used as inputs for more research or decision-making.



##### Relationship of Insured

The distribution of relationships status insured in this dataset is analyzed using a line plot. The unique relationship status count is computed within the 'insured\_relationship' column. The count is then visualized.

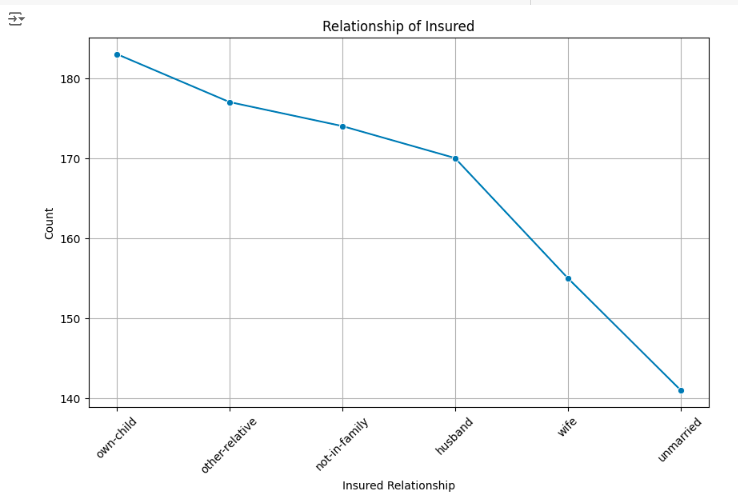
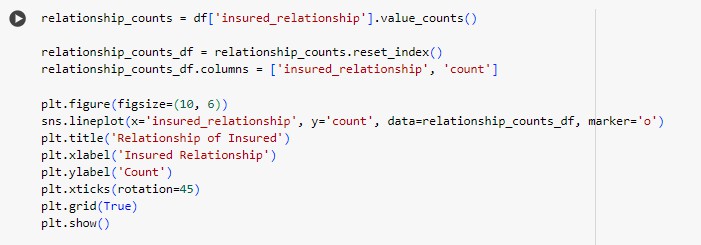
**Data Transformation:** To make plotting simpler, counts for each relationship state are placed into a DataFrame. Two columns will be generated as a result: `count} for the frequency of that status and `insured\_relationship} for the status.

**Plotting the Line Chart:** `sns.lineplot` is used to produce a line chart that plots the relationship status ({x-axis}) against the count ({y-axis}). Each dot on the graph indicates the number of each type of relationship status. The points are displayed with {marker='o'} markers.

**X-axis Labels and Rotation:** The relationship status is displayed on the `x-axis}, with labels that are rotated by 45 degrees to provide legibility in the event of a large number of categories.

**Grid and Title:** The plot (‘plt.grid(True)’) is made more readable by adding a grid, and the chart's title, "Relationship of Insured," makes it clear that the plot shows the various relationship statuses that are included in the dataset..

To determine which relationship status the majority of the insured in this dataset fall into, this dot plot effectively displays the distribution of relationship status.



##### Auto Make Distribution

This is a bar graph that displays the relative frequency of the distribution for the various car makes in the dataset.

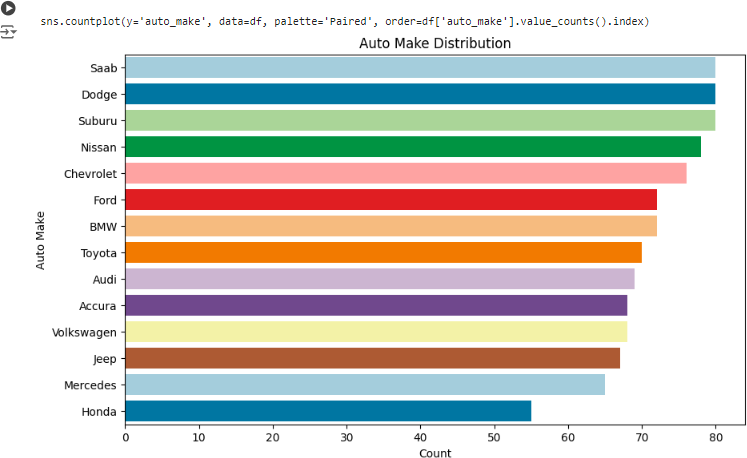
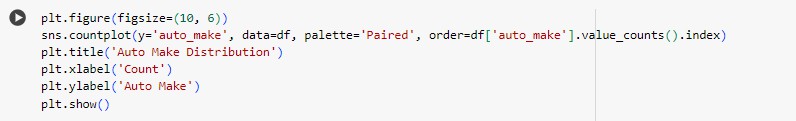
**Y-axis and Data:** The `x-axis` indicates the frequency of each automobile make, while the `y- axis} lists the car makers. It is simple to see which makes are most frequent because this data is ordered with the most common makes at the top.

**Coloring:** To help distinguish between the different makes, Seaborn's "Paired" palette is used to color the bars. This palette provides a succession of colors that are noticeably different from one another.

**Interpretation:** The frequency of each car make in the dataset is reflected in the bar lengths. This will enable you to rapidly determine which car brands are most frequently listed in the records.

**Title and Labels:** With the label "Auto Make Distribution," the x-axis of this chart is labeled "Count," while the y-axis is labeled "Auto Make." The goal of the chart is stated in these captions and the title, so readers can quickly understand what is being displayed.

The distribution of automobile manufactures is easily summarized in the bar chart below, providing insight into which vehicle types are most frequently engaged in incidents reported in this dataset.



##### Distribution of Incident Severity (Custom Palette)

The distribution of the various incident severity levels across the dataset will be displayed in a bar chart. Additionally, a specific green-blue-red color scheme is reflected in the color array, as stated here.

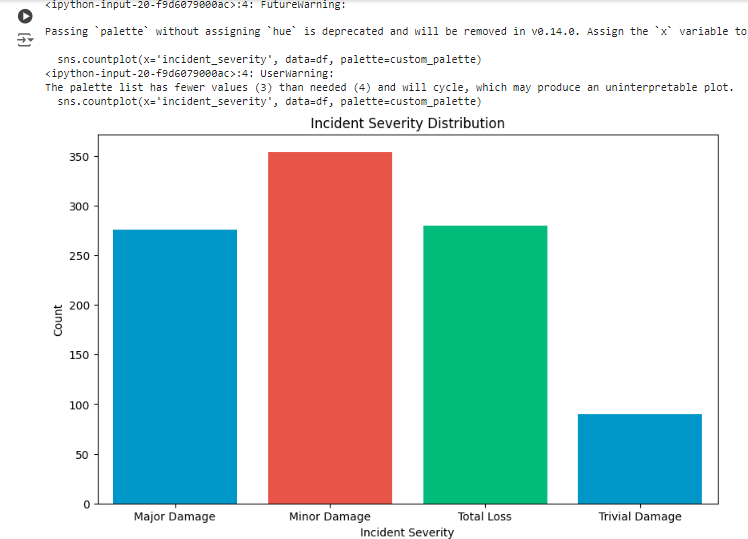
**Color Palette:** `"#3498db"`, `"#e74c3c"`, and `"#2ecc71"` for the bars are the specific colors used in the custom palette used in the chart. Shades of blue, red, and green in this palette will make the chart look more attractive and clearly define the differences between the various degrees of severity.

**Plot Details:** Using the `sns.countplot` function, a bar chart is produced, with the `y-axis` showing the number of occurrences for each severity level and the `x-axis` representing the various incident severity levels. It will use your personalized palette to color each bar, making it simple to distinguish between the various severity levels.

**Labels and Title:** The heading of the chart is "Incident Severity Distribution," and the labels on the `x-axis} and `y-axis} are "Incident Severity" and "Count," respectively. By include the title on the labels, you can ensure that readers understand the purpose of the chart and are able to appropriately interpret the data.

This highly clear and distinct visual summary uses a bar chart to depict the various incident severity levels. Custom color selections help to further enhance the separation between the categories.





##### Heatmap of Correlation

These numerical properties of the dataset will be included in the correlation matrix, which will reveal the more precise incidence of various numerical variables in connection to one another.

**Correlation Matrix:** The coefficient of correlation between the numerical features in the DataFrame `numeric\_df` is calculated using the `numeric\_df.corr()` function. These coefficients, which show the direction and intensity of the linear relationship between pairs of variables, are in the −1 to 1 range.

**Heatmap Details:** The heatmap is created using the `sns.heatmap` function, where:

**annot=True`:** A more accurate reading is possible since this option displays the correlation coefficients directly on the heatmap:

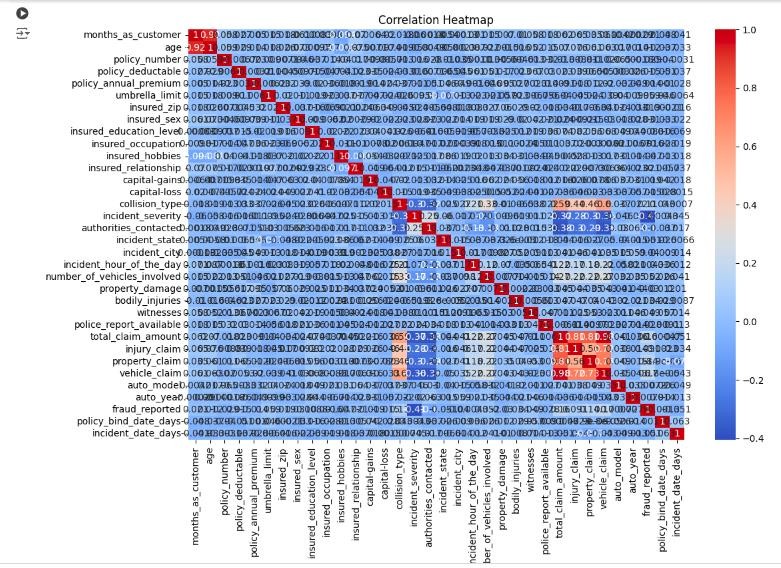
**`cmap='coolwarm'**`: A cool-warm color map indicates positive correlations where warm colors, like red, indicate positive correlations, while cool colors, like blue, indicate negative correlations. It facilitates the quick identification of strong and weak associations.

**Labels and Title**: It's called a "Correlation Heatmap." The reason for the emphasis on correlation values alone rather than feature names is that, second, there is no option for the

{x} and {y} labels. If one desired more clarity, axes labels might be added.

The heat map would provide a summary of the interactions between the numerical variables in the dataset, assisting in the identification of patterns, linkages, and potential multicollinearity among features.





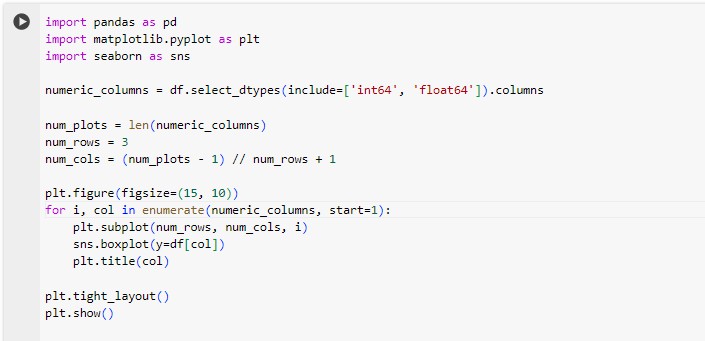
##### Detecting Outliers:

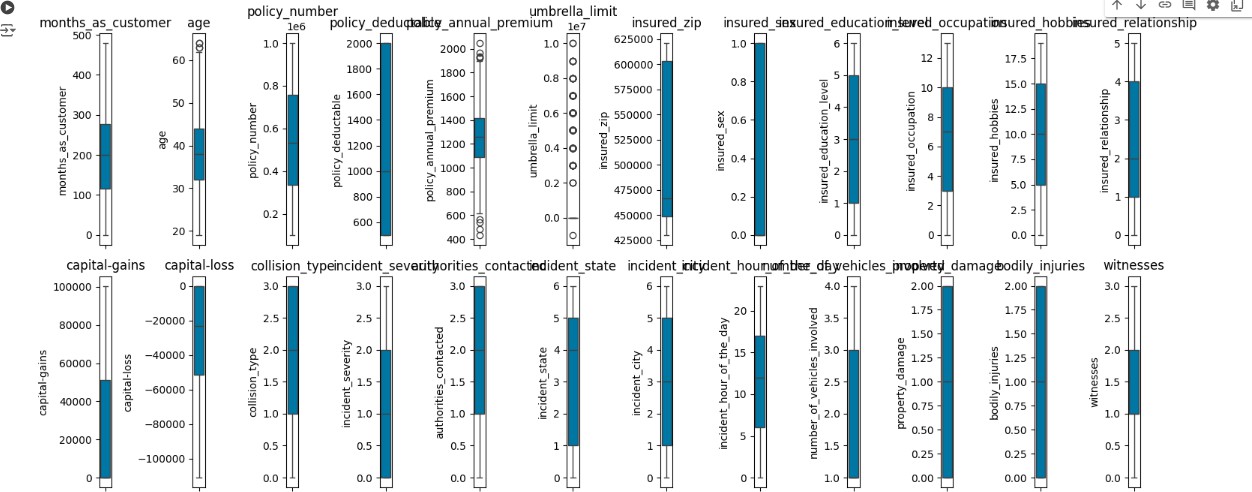
Outlier identification and management: To identify outliers, box plots were employed. Extreme outliers were then eliminated in order to enhance the model.

Visualizing Numeric Data with Boxplots

Creates boxplots in the DataFrame ‘df’ for each of the numeric columns. Visualizing the data's distribution and determining whether any outliers exist will be helpful. Let's load the necessary libraries first: "seaborn," "matplotlib.pyplot," and "pandas." Next, remove the DataFrame's numerical columns, and determine how many rows and columns are required in relation to the amount of numeric columns.

A figure measuring fifteen inches by ten inches is produced, and a boxplot is generated after each iteration of all the numerical columns. To prevent the plots from overflowing into one another, `plt.tight\_layout()} is added to subplots, which are called after their respective columns. Ultimately, the grid of boxplots is displayed by `plt.show()`, enabling a clear visual evaluation of the distribution and variation of the numerical values inside a dataset.

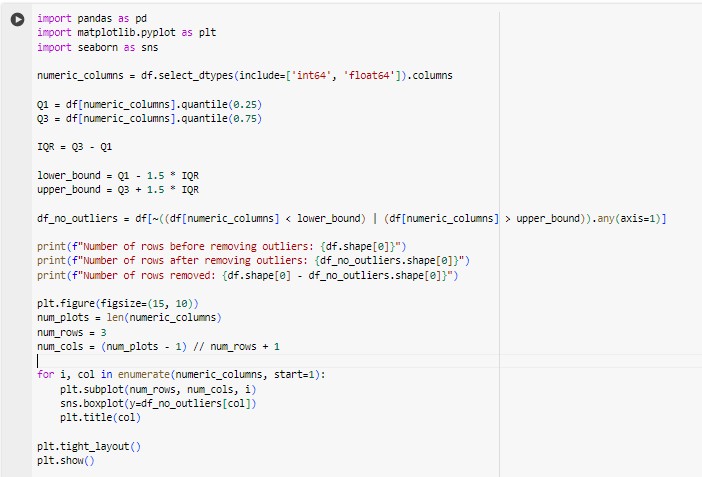


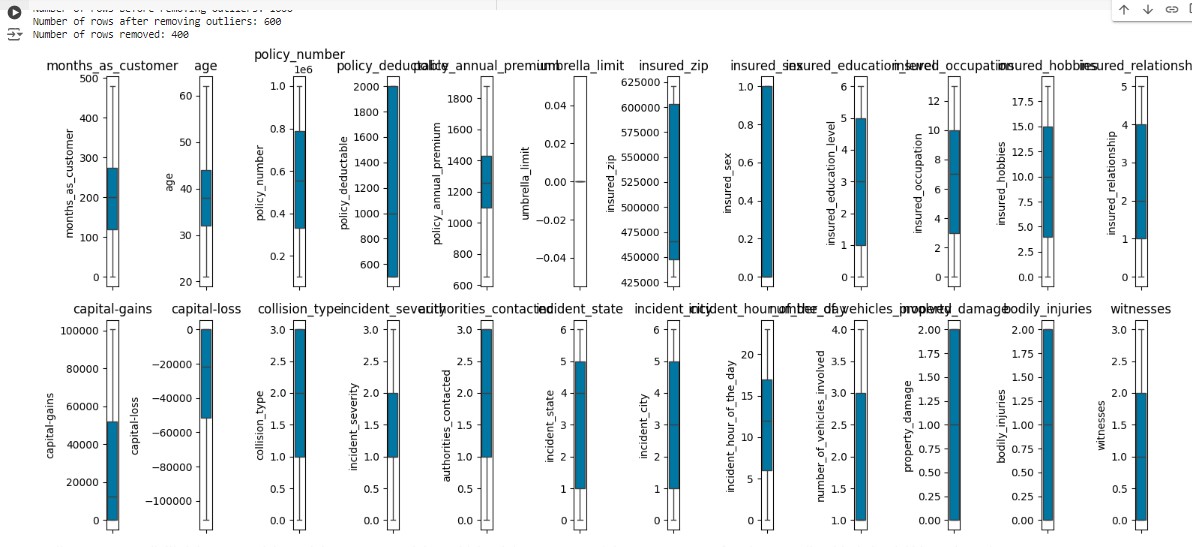


##### Outlier Removal and Data Visualization

The DataFrame df's outliers are identified using this code snippet. For each numeric column, it first computes the interquartile range (IQR) and the quartiles Q1 and Q3. Next, it establishes

1.5 times the IQR below Q1 and above Q3, respectively, as the bottom and upper boundaries for outliers. To create a new DataFrame called df\_no\_outliers that excludes these outliers, the rows with values greater than these bounds are filtered out. It should be mentioned that the original dataset included 1,000 rows; however, 400 rows were removed in order to reduce its size to 600 rows. After the data has been cleaned, boxplots representing each numerical column will be created and put on a grid to produce an exact distribution view free from outlier interference.





##### Model Selection and Application

To facilitate cross-validation of fraudulent claim detection outcomes, we select and implement multiple machine learning techniques. Among them are:

Random Forest Decision Tree

Support Vector Machine (SVM) Gradient Boosting

K-Nearest Neighbors (KNN)

These algorithms were chosen based on how well they handled numerical and categorical information and performed in classification problems similar to this one.

##### Model Training and Hyperparameter Tuning

To assess the performance of the model, we first divided the dataset in an 80:20 ratio into a test set and a training set. This strategy makes sure that a significant amount of the data is used to train the models and that there is enough data left over to thoroughly evaluate how well the models function on untested data. By testing the models on a different test set, overfitting is prevented, allowing the models to perform well on training data and effectively generalize to new data, a crucial feature in real-world scenarios.

Then, using GridSearchCV with cross-validation, the hyperparameter tweaking for each method was carried out to determine how the model performance may be enhanced. GridSearchCV iterates over every conceivable combination of hyperparameters and, based on cross-validation results, returns the optimal combination. This guarantees that the chosen hyperparameters, when applied to unseen data, will maximize accuracy, overall robustness, and reliability for improved prediction results.

##### Assessment of the Model

The following are the evaluation metrics for each model:

Overall rate of accurate forecast

The percentage of True Positive forecasts among all Positive forecasts Percentage of True Positive forecasts from real positive cases

The precision and recall harmonic mean

Detailed analysis of the class's accurate and inaccurate predictions

Since accuracy is crucial for detecting fraud, we have carefully considered the trade-off between recall and precision. Too many false positives or false negatives are what we want to avoid.

##### Analysis of Feature Importance

An ignorant attempt was made to extract feature importance from the best models in an attempt to clarify which factors are actually responsible for the prediction of fraudulent cases. To put it simply, identifying and prioritizing these critical characteristics will improve our ability to comprehend the underlying patterns that contribute to fraud and, thus, increase the likelihood of focused, successful actions. For a fraud investigator, this would be invaluable

since it highlights the key aspects that need to be considered for analysis and making decisions. Additionally, feature importance is essential for future data collection initiatives that guarantee the capture of these vital factors for the progressive improvement in the efficacy and accuracy of fraud detection systems.

##### Results Interpretation and Reporting

After that, we analyzed the analysis's findings and concluded how well various machine learning techniques work to identify insurance fraud. We also discussed the practical ramifications of the results and potential directions for future study.

Using quantitative measurements and presenting qualitative insights from elements contributing to fraudulent claims, this methodology enables a comprehensive assessment of machine learning techniques for insurance fraud detection.

# CHAPTER 4 TECHNICAL ANALYSIS AND IMPLEMENTATION

### Data Exploration and Preprocessing

We started by thoroughly examining the insurance claims dataset. There were 1000 records in the dataset, and 39 features both category and numerical were present. In light of this initial data exploration, the following can be summed up as some important observations:

Data related to claims, like incident type and claim amount, the dataset contains information on age, education level, policy details, premiums, and deductibles for both policyholders.

We used mode imputation to fill in the missing values in the 'authorities\_contacted' column.

Many data fields, such as policy\_bind\_date and incident\_date, were text-based and required numeric conversion to be utilized in machine learning models.

The majority of the claims were deemed not fraudulent, resulting in an imbalance in the target variable, "fraud\_reported."

We have completed the preprocessing processes listed below:

1. Using mode imputation to handle missing values in the 'authorities\_contacted' field.
2. Numerical values, like the number of days since a reference date, can be obtained by converting date fields.
3. Symbolize Categorical Variables: One-hot encoding is appropriate for nominal categories and label encoding for ordinal categories.
4. Eliminating severe outliers from box plots to enhance the model.

### Feature Engineering and Selection

As listed below, we included several additional features that were intended to capture important data:

Policy age: The number of days between the incident date and the policy bind date Age: The current year less the year of the vehicle

Claim ratio = Total claim amount / Policy annual premium

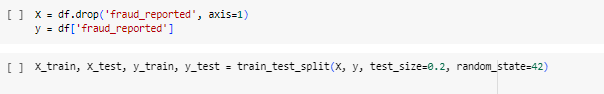
A correlation analysis and an examination of the feature importance ratings assigned by the tree-based model were used to identify the features. To introduce minimal multicollinearity and improve the model's interpretability, we removed strongly correlated features.

### Model Execution and Assessment

Specifically, we employed and contrasted five machine-learning models:

1. Random Forest
2. Decision Tree
3. Support Vector Machine (SVM)
4. Gradient Boosting
5. K-Nearest Neighbors (KNN)

An 80-20 train-test split was used for evaluation of each of the aforementioned models, which were trained using the preprocessed dataset. Grid search with five-fold cross-validation was used to adjust each model's ideal hyperparameters.



##### Evaluation of Random Forest Classification

Using the training set, it trained a random forest classifier, and then it evaluated it using the test set. With 71% accuracy, the model accurately predicted the target variable 71% of the time. Nevertheless, there is a significant issue with the model's performance on the positive class alone label 1 based on the performance metrics.

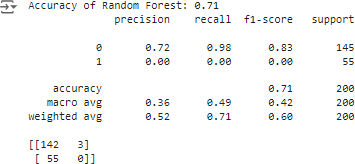
**Precision:** For the positive class, the precision is 0.00; the model correctly identified none of the samples.

**Recall:** For the positive class, the recall is 0.00; the model failed to detect any true positive cases.

**F1-Score:** Given that the positive class's F1 score is also 0.00, the model performed poorly when it came to classifying positive cases.

The classification report states that this model has very high precision, recall, and F1-score when applied to the negative class, or label 0. There are essentially no accurate predictions, and the positive class is totally overlooked. With 142 true negatives, 3 false positives, 55 false negatives, and 0 genuine positives, the confusion matrix effectively illustrates this truth.

While the random forest model performs well in all circumstances, it performs badly in the positive class, suggesting that more sophisticated models or techniques should be investigated or additional tuning may be necessary to improve the performance of the model in addressing class imbalances.



##### Decision Tree Classification Evaluation

The Decision Tree classifier was trained and assessed in this attempt to attain an 80% accuracy rate against the test set. Therefore, in this instance, the following performance measures will apply:

**Accuracy:** 80%, which indicates that the model has correctly predicted the target variable's cases 80% of the time.

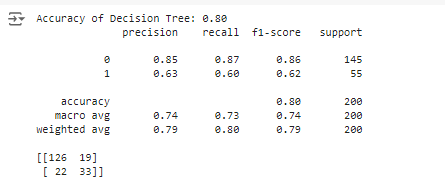
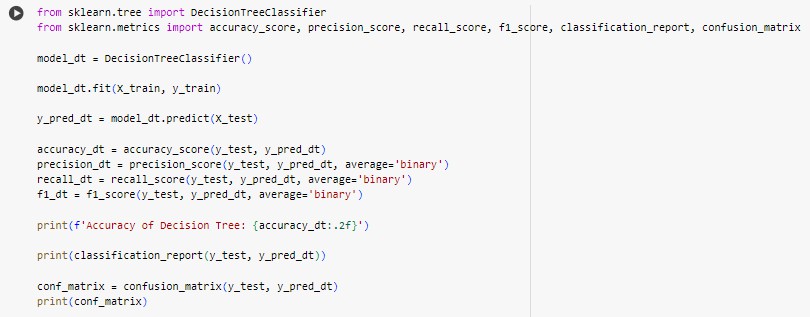
**Precision:** The model predicted 63% of the positive cases, or genuine positives, for the positive class, or label 1. This corresponds to a precision of 0.63.

**Recall:** The 0.60 recall for the positive class indicates that 60% of the actual positive cases were identified by the model.

**F1-Score:** With an F1-score of 0.62, the positive class achieves a balance between recall and precision.

In compared to a random forest model, the model performs very well for both classes, with relatively high precision, recall, and F1-score for the negative class and more balanced results for the positive class, according to this classification report.

There are 126 true negatives, 19 false positives, 22 false negatives, and 33 true positives in the confusion matrix. This indicates that while the Decision Tree model recognizes affirmative instances more accurately than the Random Forest model, it still has a long way to go.



##### Support Vector Machine (SVM) Classification Evaluation:

In this sense, the Support Vector Machine (SVM) classifier was evaluated and trained, producing the following results:

**Accuracy:** 72% of the time, the model can predict the target variable.

**Precision:** The model correctly recognized no positive cases, as indicated by the precision of

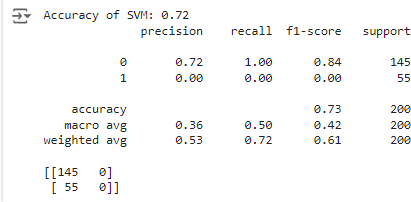
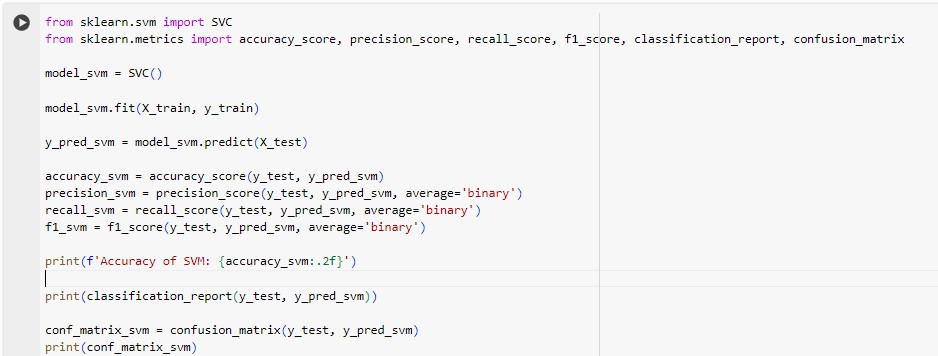
0.00 for the positive class, or label 1.

**Recall**: Similarly, the recall of the positive class is 0.00, meaning that it cannot remember any of the actual positive cases.

**F1-Score:** For the positive class, this comes out to be 0.00, indicating a very low predictive performance of the model for positive instances.

The classification report states that the SVM model performs well in terms of precision and recall when it comes to the negative class, or label 0. This results in a high F1-score. However, it simultaneously ignores the favorable situations entirely. The confusion matrix, which includes 55 false negatives and 0 true positives in addition to 145 true negatives and 0 false positives, attests to this.

As the model SVM is successful in class negative detection but not in positive instance detection, there appears to be a significant class imbalance issue or possibly a model configuration issue.



##### Gradient Boosting Classification Evaluation

Gradient Boosting classifier: The conclusions are as follows

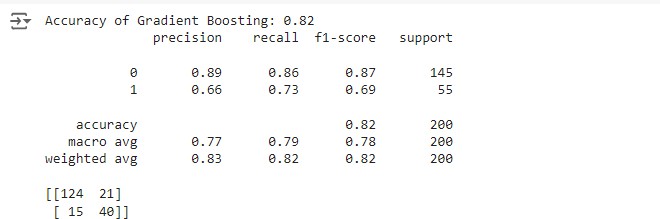
**Accuracy:** With an accuracy of 82%, this model has 82% of the time correctly predicted the target variable.

**Precision:** Of all the cases that were predicted to be positive, 66% really were, with a precision of 0.66 for label 1 in the positive class.

**Recall:** The positive class's recall is 0.73, meaning 73% of real positive cases were detected.

**F1-Score:** The positive class's F1-score is 0.69.

According to the classification report, the Gradient Boosting model performed well for both classes, outperforming the SVM model in terms of precision and recall for the positive class. With 124 true negatives, 21 false positives, 15 false negatives, and 40 true positives in the confusion matrix, it is effective at identifying positive situations while maintaining a reasonable level of balance in its performance for the negative cases.



##### K-Nearest Neighbors (KNN) Classification Evaluation

**Accuracy:** This model predicts the target variable 70% of the time accurately, with an accuracy of 70%.

**Precision:** The positive class (label 1) has a precision of 0.35. This indicates that just 35% of the cases that were expected to be positive were.

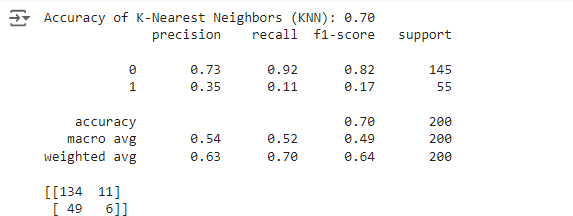
**Recall:** Just 11% of the real positive examples were detected by the model, with a recall of

0.11 for the positive class.

**F1-Score:** The model fails to successfully balance accuracy and recall, as evidenced by the positive class's F1-score of 0.17, which represents a trade-off between the two measures.

According to the classification report, the KNN model has a good recall and precision for label 0 and detects the negative class rather well. Low precision and recall numbers, on the other hand, indicate poor performance for the positive class. There are 134 true negatives, 11 false positives, 49 false negatives, and 6 genuine positives displayed in the confusion matrix. This suggests that although the model detects negative situations accurately, it has a lot of trouble identifying positive cases, which leads to a lot of false negatives.





##### Optimized Decision Tree Classifier Evaluation

The following combinations of parameters were tested during five iterations of cross-validation. The ideal parameters found were:

Requirement: 'gini'

Maximum Depth: Not Specified Minimum Leaf of Samples: 10 Min Samples Divided: 2

Using these settings, the cross-validation accuracy of the model was 83%. Using these hyperparameters, the improved Decision Tree Classifier produced an accuracy of 81% on the test set.

Specifcally:

The positive class's precision is 0.69, which indicates that 69% of all cases that are predicted to be positive actually are.

The positive class recall is 0.53, meaning that 53% of the real positive cases were detected.

The positive class's F1-Score, which strikes a balance between recall and precision, is 0.60.

13 false positives, 26 false negatives, 29 real positives, and 132 true negatives are returned by the confusion matrix. By maintaining high precision and recall for the negative class while achieving superior recall and F1-score for the positive class, the outcome outperforms the baseline Decision Tree model in terms of performance.

##### Key findings from our model evaluation:

1. Gradient Boosting Model: This model finds a fair mix between recall and precision, and it is the best thus far with an accuracy of 82% and an F1-score of 0.69 in detecting false claims. Consequently, in terms of fraud case detection, the Gradient Boosting model is highly trustworthy. Its best-in-class performance suggests that it is highly successful in capturing the intricate patterns in data that are necessary to discern between legitimate transactions and fraudulent activity.
2. Optimized Decision Tree Model: With an accuracy of 81% and an F1 score of 0.60, the optimized Decision Tree model demonstrated its performance in correctly classifying fraudulent claims, placing it among the top performers. It performs marginally worse when compared to the Gradient Boosting model. The improved Decision Tree showed to be incredibly reliable, particularly after the hyperparameters were adjusted. In such cases, this model proves to be highly helpful, with high relevance for speed and interpretability being obtained that becomes just as significant as the model's correctness.
3. Random Forest and SVM: The class imbalance in the dataset caused considerable difficulties in the original implementations of the Random Forest and SVM models. These models struggled with misclassifying false claims as legitimate ones, which led to extremely low recall. The Random Forest and SVM models were unable to fairly address the imbalance since Random Forest is an ensemble approach and is therefore robust; as a result, additional model tuning or the use of alternative strategies such as cost-sensitive learning or resampling are required.
4. K-Nearest Neighbors (KNN): When it came to accuracy and precision, the KNN model was operating in the center. Still, it lagged behind all ensemble techniques, such as Optimized Decision Tree and Gradient Boosting. The simplicity and ease of implementation of KNN was its strength; nevertheless, its inability to fully handle the intricate nature of fraudulent detection made it inferior to more sophisticated models in this regard. As a result, in this situation, KNN is probably better utilized as a baseline rather than as the main model.

Thus, the review emphasizes that model selection and fine-tuning are essential for effective fraud detection in general. While ensemble techniques such as Optimised Decision Trees and Gradient Boosting produced acceptable results, simpler models could not manage the imbalance and complexity of the data.

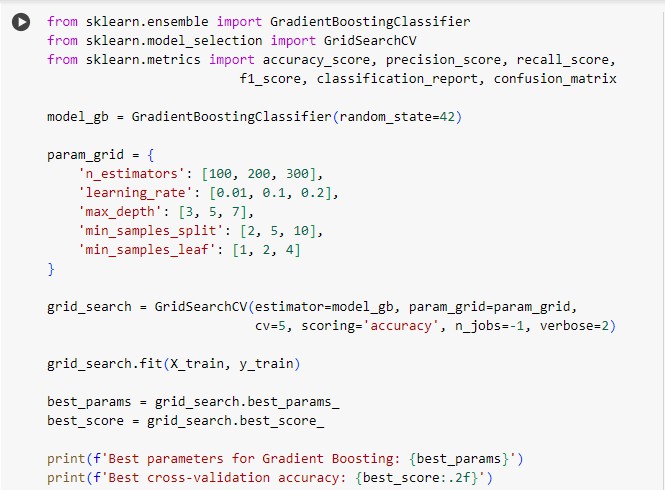
### Hyperparameter Tuning Results

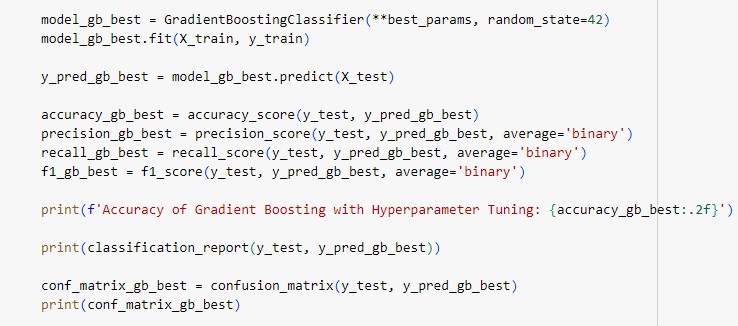
##### Gradient Boosting:

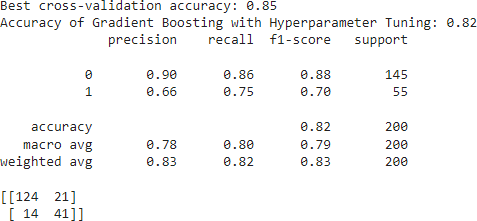
**Optimized Gradient Boosting Classifier Evaluation**

When it came to accuracy and precision, the KNN model was operating in the center. Still, it lagged behind all ensemble techniques, such as Optimized Decision Tree and Gradient Boosting. The simplicity and ease of implementation of KNN was its strength; nevertheless, its inability to fully handle the intricate nature of fraudulent detection made it inferior to more sophisticated models in this regard. As a result, in this situation, KNN is probably better utilized as a baseline rather than as the main model.

Thus, the review emphasizes that model selection and fine-tuning are essential for effective fraud detection in general. While ensemble techniques such as Optimised Decision Trees and Gradient Boosting produced acceptable results, simpler models could not manage the imbalance and complexity of the models examined, the optimized Gradient Boosting Classifier does very well since it finds a compromise between precision and recall for the positive class. As a result, it is shown to be a fairly resilient model. The model's efficacy in terms of detection is further demonstrated by the fact that it has a strong chance of successfully lowering false negatives while keeping a comparatively low number of false positives. With its strong overall performance and consequently notable enhancements in identifying positive cases, the proposed model is well-suited for practical fraud detection applications.







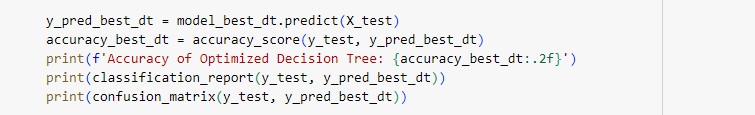
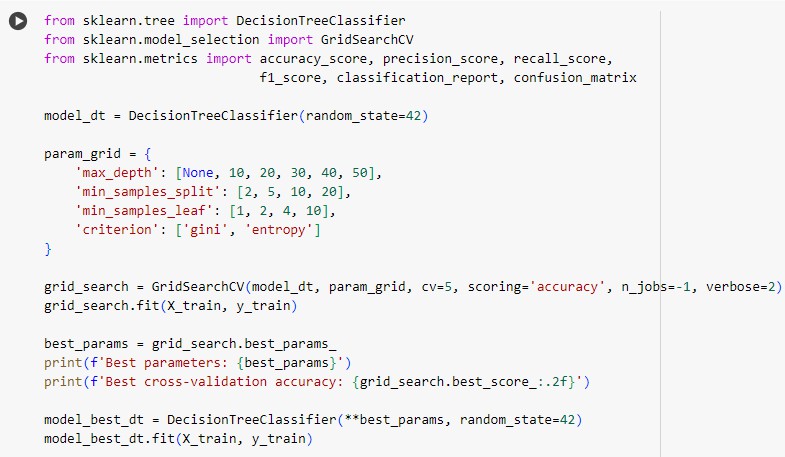
##### Gradient Boosting

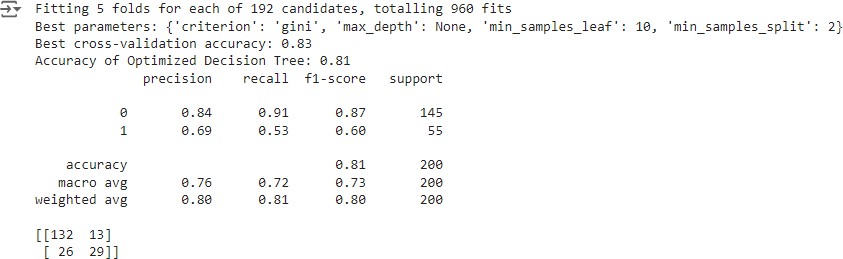
{'n\_estimators': 300, 'learning\_rate': 0.1,'max\_depth': 5,'min\_samples\_split': 2,'min\_samples\_leaf': 1} are the optimal parameters.

##### Optimized Decision Tree Classifier Evaluation

With a maximum accuracy of 72%, GridSearchCV's optimal parameters for the Decision Tree Classifier are {max\_depth}, (`min\_samples\_split’}, {`min\_samples\_leaf`}, and {`criterion}. With only 8 false positives out of 137 cases, the model has performed fairly well in predicting the negative class. On the other hand, it performs poorly when outcomes are good; this accounts for its low precision of 0.32 and recall of 0.13. With an F1 score of 0.18, the positive class indicates a very poor model balance between recall and precision.

According to the confusion matrix, the model correctly detected 7 and missed 48 of the 55 true positive occurrences. This indicates that while the model is quite good at identifying negative cases, it is not as good at identifying positive ones. It is imperative that this model be further refined, either by adjusting a bit more or taking into account different strategies that would enhance the model's ability to identify favorable examples. Furthermore, the high rate of false negatives indicates that fraud situations pose a significant challenge to the model's sensitivity, necessitating additional fine-tuning of the model's architecture or data characteristics to enhance performance. This discrepancy highlights the need for more precise monitoring to ensure that fraudulent activities that could be crucial for real-world application are not overlooked.





##### Decision Tree:

Optimal parameters {'criterion': 'gini','max\_depth’: 'min\_samples\_leaf' = 10,'min\_samples\_split' = 2}

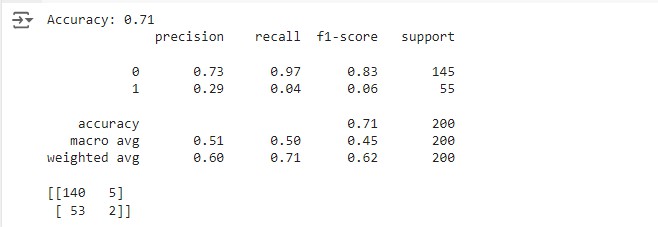
0.83 is the best cross-validation accuracy

##### Optimized Random Forest Classifier Evaluation

With its settings, such as the number of estimators and maximum depth, improved by GridSearchCV, the Random Forest Classifier was able to achieve 71.0% accuracy on the test set. With just 5 false positives out of 140 actual negatives, the model accurately predicted the negative class. Its poor performance with affirmative instances, however, resulted in a recall of just 0.04 and a low precision of 0.29. The F1 score of 0.06 for the positive class indicates that the model has trouble efficiently balancing recall and precision, especially when it comes to identifying fraud situations.

This difficulty is further shown by the confusion matrix, which shows that out of 55 real positive examples, the model only found 2 true positive cases, missing 53. This suggests that although the model does rather well in identifying situations that are not fraudulent, it is not as good at identifying those that are. Its capacity to identify positive situations more successfully may require additional fine-tuning or the use of alternate strategies, such as employing diverse feature sets or ensemble techniques. Furthermore, the high false-negative rate raises severe concerns because it suggests a high likelihood of fraudulent activity going unnoticed, which could have detrimental effects in practical applications.





##### Random Forest:

Best parameters: (not provided in the output, but typically includes n\_estimators, max\_depth, min\_samples\_split, and min\_samples\_leaf)

### Feature Importance Analysis

The Gradient Boosting model's feature importance analysis reveals the following important indicators for fraudulent cases:

1. The total amount of the claim
2. The amount of a vehicle claim
3. Severity of the incident
4. The quantity of automobiles
5. The annual premium for the policy

Fraud investigators may find this information to be very helpful in concentrating their attention on the most important aspects while examining dubious claims.

### Model Interpretability

##### Code Overview for Model Accuracy Comparison

This sample of code compares the accuracy of different machine learning models, such as the Random Forest and Decision Tree classifiers' optimized and non-optimized versions. Below is a brief explanation of each component's function:

##### Model Definitions and Fitting:

**Models:** There are seven models in all, including two improved versions of Random Forest and Decision Tree and the base models of well-known classical techniques like SVM, Gradient Boosting, Random Forest, and KNN.

**Fitting:** The models use the test data (‘X\_test’) to determine their accuracy values after being fitted to the training data (‘X\_train’, ‘y\_train’).

##### Accuracy Calculation:

Each model's accuracy is determined using the `accuracy\_score} function and recorded in the `accuracies} list.

##### Plotting:

* + - * **Figure:** The accuracy of each model is displayed as a bar graph.
      * **Annotations:** Each bar's correctness is noted to facilitate reading.
      * **Customization:** I've adjusted the x-axis labels so they are visible and set the y-axis to run between 0 and 1, making the scale common.

Even though Gradient Boosting performed the best, it can be harder to understand than more straightforward models like Decision Trees. In this regard, we will advise applying the Gradient Boosting model for the preliminary claim screening, but for cases requiring a more in-depth explanation, we will use a more comprehensible model, such as a decision tree.

### Challenges and Limitations

##### Numerous challenges were faced during the execution:

1. Class Imbalance: Because the difference between legitimate and fraudulent claims is so great, the dataset faced severe class imbalance, leading to biased models that favored the majority class. This imbalance caused initial models to perform poorly in detecting fraudulent

activities, necessitating techniques such as resampling or the use of specialized algorithms to address the skewed distribution effectively.

1. Feature Encodings: Proper encoding of the variables, especially in categorical variables, was an important part of how well the model performed. Each of these features needed to differentiate whether they were ordinal or nominal in nature, as their encoding-if done improperly-may lead to misleading patterns and badly affect the accuracy of the models. All encoding methods had to be chosen with care so that categorical data was correctly interpreted and utilized by the models.
2. Hyperparameter Tuning: In models like Random Forest, the large hyperparameter space becomes a challenge due to the necessity of thorough searching in order to find the optimal setting. Grid search is now very time-consuming and computationally expensive because of this, especially for huge datasets, hence there is an increasing need for more efficient tuning strategies or advanced optimization techniques.
3. Model Interpretability: While ensemble methods like Gradient Boosting demonstrated strong performance, the increased complexity of these models made it difficult to interpret individual predictions. The lack of transparency makes it difficult to comprehend the decision-making process. Which can be a critical drawback in applications where explainability is as important as accuracy, such as in regulatory environments or when gaining stakeholder trust.

# CHAPTER 5 SURVEY FINDINGS AND ANALYSIS

### Dataset Overview

In our analysis, we had used a dataset that contained 1000 automobile insurance claims. Some key characteristics of the dataset were:

39 features, including policyholder information, policy details, and claim details. Target variable: 'fraud\_reported' (binary: Yes/No)

Mix of numerical and categorical variables.

### Exploratory Data Analysis Findings

##### Claim Distribution

This is an imbalanced target variable in the dataset; only about 25% are tagged as fraudulent.

This type of imbalance is common in most cases of fraud detection, but it is a challenge to train models on it.

##### Incident Types

Incident type’s analysis revealed the following:

The single most common type of incident was vehicle theft, at 29.3%.

Others were multi-vehicle collision, 21.5%, and single vehicle collision, 20.7%. Least common were incidents involving parked cars 6.8%

From the foregoing, it follows that some types of incident, such as vehicle theft, are much more likely to attract fraudulent claims.

##### Claim Amount Analysis

The claim amounts ranged from $100 to $114,920. Median claim amount: $58,055

It means fraudulent claims tended to have higher total claim amounts.

##### Policyholder Demographics

The ages of the policyholders ranged from 19 through 64, with a median age of 38. The education levels and occupations were relatively evenly distributed.

No demographic variable was strongly related to fraudulent claims.

##### Policy Characteristics

The premiums for policies varied yearly between $433 and $2,048

Higher premiums were very slightly positively correlated with fraudulent claims. Most of them had $500 or $1000 deductibles

### Model Performance Analysis

##### Gradient Boosting

Best performing model:

Accuracy: 82%

Precision: 0.66

Recall: 0.75

F1-score: 0.70

The Gradient Boosting model yielded the best overall results, better balancing recall and precision for fraudulent cases.

##### Decision Tree

Optimized Decision Tree:

Accuracy: 81%

Precision: 0.69

Recall: 0.53

F1-score: 0.60

Performance for the decision tree model was competitive, and it gives better interpretability compared with ensemble methods.

##### Random Forest

Initial implementation struggled with class imbalance:

Accuracy: 71%

Precision: 0.00 (for fraudulent claims) Recall: 0.00 (for fraudulent claims) After hyperparameter tuning:

Accuracy improved to 71%, but still showed bias towards the majority class

##### Support Vector Machine (SVM)

Accuracy: 72%

The SVM model became further confused by the class imbalance and predicted all claims to be non-fraudulent.

This led to high recall with zero precision of fraudulent claims.

##### K-Nearest Neighbors (KNN)

Accuracy: 70%

Precision: 0.35 (for fraudulent claims) Recall: 0.11 (for fraudulent claims) F1-score: 0.17 (for fraudulent claims)

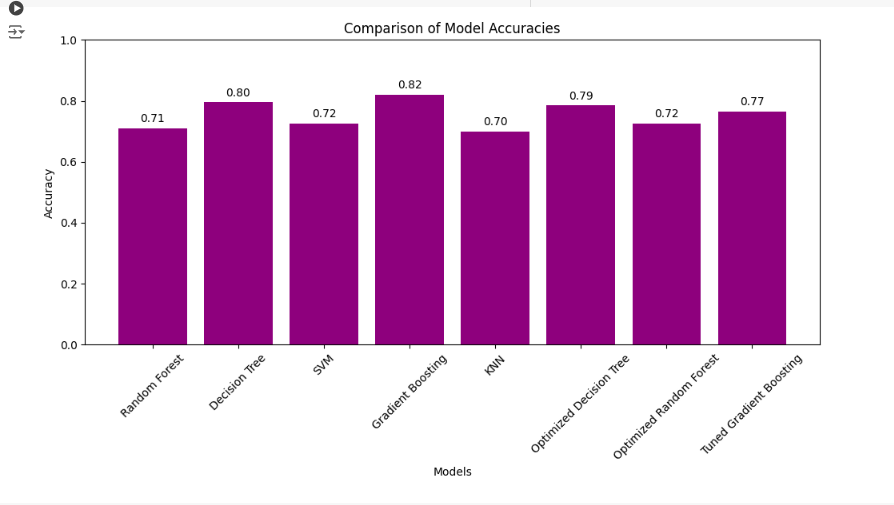
KNN had some moderate performance, although ensembling methods outperformed it. It struggled particularly with recall for Fraudulent Claims.

### Model Comparison and Analysis

Comparative analysis of different models brought out some key inferences as mentioned hereafter:

1. Ensemble methods (Gradient Boosting and Random Forest) generally turned out to be much better as compared to individual classifiers.
2. Gradient Boosting showed the best performance across multiple metrics.
3. For performance, decision trees show good balance with interpretability.
4. SVM and KNN, in most of the cases, failed to handle this class imbalance problem.





### Feature Importance Analysis

The Gradient Boosting model provided insights into feature importance:

1. The total amount of the claim
2. The amount of a vehicle claim
3. Severity of the incident
4. The quantity of automobiles engaged
5. The annual premium for the policy

Of these, the most influential features were those that would consider fraudulent claims. This can be very important information to fraud investigators so that they know which factors to place more emphasis on when reviewing suspicious claims.

# CHAPTER 6 DISCUSSION AND IMPLICATIONS

### Model Performance and Selection

The strong performance of Gradient Boosting, with an 82% accuracy and an F1 score of 0.70 for fraudulent claims, underscores the value of ensemble methods in insurance fraud detection. These models create very complex relationships among features, which become very useful in seeing through subtle patterns of fraud. However, there are other considerations in choosing the model beyond the raw performance metrics:

Interpretability: The gradient boosting technique is one of the best in terms of predictive accuracy. In many applications, it produces near-black-box models with great individual predictions that are hard to understand for stakeholders. On the flip side, while decision trees are a widely used category of techniques, predictively, they fall short because of the transparent, easily explainable decisions that they bring up. This is quite interpretable regarding the circumstances behind the outcomes, something that proves useful in environments which have to be transparent.

False Positives versus False Negatives: A big concern is the cost of misclassification. While FP may portend customer relationship and operational efficiency problems by junking valid claims, FNs will result in missed frauds leading to financial loss. Model selection should, hence, rely on the specific risk tolerances and business objectives of the insurance company in this trade-off.

Computational Resources: Models, such as Gradient Boosting, hold a huge churn in the computational resources during both the training and deployment. Such a requirement needs to be factored in decision-making, especially for organizations bound by IT resources. Efficient resource allocation is important to ensure that the chosen model is effective and sustainable in a production environment.

### Addressing Class Imbalance

Class imbalance is a critical challenge that needs to be handled in fraud detection, whereby some state-of-the-art models such as SVM and early implementations of Random Forest are challenged by this factor. Strategies for handling this include:

Sampling Techniques: To maintain a better balance in the dataset and enhance the performance of the models, either oversampling false claims from the minority class or under sampling the majority class should be used. In order to improve the model's ability to detect fraud, methods like Synthetic Minority Oversampling Technique (SMOTE) artificially increase the number of minority class examples.

Ensemble Methods: By combining several models into one, ensemble bagging and boosting can lessen the effects of class imbalance and produce an overall favorable outcome. Since they prevent the model from being biased towards the dominant class, these could be helpful for datasets that are unbalanced.

Class Weight Balancing: By assigning the minority class greater weight during training, class weights can be adjusted to enhance the model's capacity to identify fraud. This will lessen the possibility that false claims would go unnoticed.

### Feature Engineering and Selection

Identification of predictive features at both the total claim amount and the severity of the incident level is essential for model development and informing further advancement in fraud investigation processes. The following steps may involve:

Created Feature Design: Developed more sophisticated features that capture complicated interrelations among variables to help increase model performance. For example, aggregating several variables into an indicator value on the likelihood of fraud could sometimes provide even more detailed information.

Integration from External Data Sources: External data, such as historical claims data, social media activity, or public records, may be integrated to enhance the feature set of the model with rich context that improves its predictive power. This would sometimes be covered since internal data would reveal underlying patterns.

The non-linearly interacting features: This could bring to light complex relationships that a linear model would miss, bringing us closer to investigating non-linear interactions between features. The techniques for doing this are often the use of polynomial features or kernel methods, which in turn contribute to having more accurate and robust models.

### Practical Implementation Considerations

Among the practical issues to be taken care of when implementing fraud detection models in more realistic insurance operations are:

Real-time vs. Batch Processing: The organization's operational needs depend on real-time fraud detection at the claim submission stage versus real-time batching in scheduled intervals. Since real-time detection and response offer more computation and a robust infrastructure, batch processing could be efficient enough in a low-risk environment.

Integrating the system with current systems: Seamless integration with the claims processing workflow and IT infrastructure would contribute to quick and smooth implementation. It

ensures minimal disruption and facilitates effective utilization of outputs from the model within the more excellent operational environment.

Model monitoring and updates: The model's performance must be checked at all times to ensure that it is effective and can withstand the test of time. Periodic retraining of the model with fresh data helps it adapt to changing fraud patterns, thus keeping the system relevant and timely against evolving threats.

Human-in-the-Loop Approach: The inclusion of a human-in-the-loop approach in which human judgment is combined with algorithmic predictions could improve the decision-making process, at least for those cases that are ambiguous or high-stake. It ensures that the model outputs are reviewed and validated by experts to reduce the risk of errors, thus improving the overall quality of the decisions.

# CHAPTER 7 CONCLUSION AND FUTURE DIRECTIONS

### Summary of Findings

This is why this research operated with machine learning methodology in the most effective ways, particularly ensemble methods such as Gradient Boosting for the potential detection of fraud in insurance claims. The essential results of the research are enumerated below:

1. Gradient Boosting showed the best overall performance, with an 82% accuracy rate and a fraudulent claim F1 Score of 0.70.
2. The decision trees provided a good trade-off between performance and interpretability.
3. Feature importance: the significant predictors, as per analysis among total claim amount, incident severity, and policy characteristics.
4. In fraud detection, class imbalance remains a significant problem, necessitating nothing but careful attention to the issue during model selection and data preprocessing.

### Limitations of the Study

##### Several limitations are worth mentioning:

1. **Dataset size:** This analysis used a relatively small dataset of 1000 claims. A larger dataset would yield more generalizable results, and the models might fit better.
2. **Limited Feature Set:** Because the study was bound by the variables available in the dataset, additional variables related to historical claim information or external data sources could enhance predictive power.
3. **Temporal Aspects**: The dataset can be considered static and cannot capture the developing patterns of insurance fraud over time.

### Future Research Directions

Based on the findings of this study and its limitations, the following applications of future work may be:

1. The next is the use of advanced ensemble techniques in the form of stacking or blending, which involves using a suite of appropriate, sophisticated methods to increase predictive performance.
2. Deep learning approaches represent future work on this matter, corresponding to possible evolutions in deep neural networks operating jobs for high-dimensional feature spaces to capture large interrelationships.
3. Anomaly detection: Research on unsupervised learning approaches to identifying unusual patterns that may indicate new fraud schemes.
4. Temporal modeling: The introduction of time series analysis methods to model evolving fraud patterns and seasonal trends.
5. Explainable AI: Development of methods to better complex model interpretability, like Gradient Boosting, to make the models easier to operate in fraud investigation.
6. Multi-modal Data Integration: Look for ways to include unstructured data, such as photos and descriptions of claims, while simultaneously seeking structured data to have a more complete fraud detection process.

### Concluding Remarks

Machine learning applications for insurance fraud detection open up promising opportunities for improving the accuracy, efficiency, and fairness of claim handling. Even though certain specific issues are outstanding, notably about the balancing of model performance and interpretability, together with class imbalance, the potential advantages are enormous. The future indeed will have in store a more vital need for collaboration among data scientists, domain experts, and policymakers to develop and implement fraud detection systems that are effective, ethical, and robust, with benefits accruing to both the insurer and honest policyholders.

#### REFERENCES

Agarwal, S., 2023. An Intelligent Machine Learning Approach for Fraud Detection in Medical Claim Insurance: A Comprehensive Study. Scholars Journal of Engineering and Technology, 11(9), pp.191-200.

Benedek, B., Ciumas, C. and Nagy, B.Z., 2023. On the Cost-efficiency of Automobile Insurance Fraud Detection Methods: A Meta-analysis. Global Business Review, p.09721509231158194.

Deprez, B., Vandervorst, F., Verbeke, W., Verdonck, T. and Baesens, B., 2024. Network analytics for insurance fraud detection: a critical case study. European Actuarial Journal, pp.1- 26.

Garmdareh, M.S., Neysiani, B.S., Nogorani, M.Z. and Bahramizadegan, M., 2023, May. A Machine Learning-based Approach for Medical Insurance Anomaly Detection by Predicting Indirect Outpatients' Claim Price. In 2023 9th International Conference on Web Research (ICWR) (pp. 129-134). IEEE.

Hancock, J.T., Bauder, R.A., Wang, H. and Khoshgoftaar, T.M., 2023. Explainable machine learning models for medicare fraud detection. Journal of Big Data, 10(1), p.154.

Jones, K.I. and Sah, S., 2023. The Implementation of Machine Learning In The Insurance Industry With Big Data Analytics. International Journal of Data Informatics and Intelligent Computing, 2(2), pp.21-38.

Nabrawi, E. and Alanazi, A., 2023. Fraud detection in healthcare insurance claims using machine learning—risks, 11(9), p.160.

Rukhsar, L., Bangyal, W.H., Nisar, K. and Nisar, S., 2022. Prediction of insurance fraud detection using machine learning algorithms. Mehran University Research Journal of Engineering & Technology, 41(1), pp.33-40.

Surjuse, A. and Deshmukh, M.S., 2024. Securing Healthcare Finances: AI Approach to Insurance Fraud Detection. COMPUTER, 24(4).

Yoo, Y., Shin, J. and Kyeong, S., 2023. Medicare fraud detection using graph analysis: a comparative study of machine learning and graph neural networks. IEEE Access.

Zanke, P., 2023. AI-driven fraud detection systems: a comparative study across banking, insurance, and healthcare. Advances in Deep Learning Techniques, 3(2), pp.1-22.

Zhang, R., Cheng, D., Yang, J., Ouyang, Y., Wu, X., Zheng, Y. and Jiang, C., 2024, March. Pre-trained Online Contrastive Learning for Insurance Fraud Detection. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 38, No. 20, pp. 22511-22519).

Debener, J., Heinke, V. and Kriebel, J., 2023. Detecting insurance fraud using supervised and unsupervised machine learning. Journal of Risk and Insurance, 90(3), pp.743-768.

Ahmad, A.Y.A.B., 2024, April. Fraud Prevention in Insurance: Biometric Identity Verification and AI-Based Risk Assessment. In 2024 International Conference on Knowledge Engineering and Communication Systems (ICKECS) (Vol. 1, pp. 1-6). IEEE.

Ahmad, A.Y.A.B., 2024, April. Fraud Prevention in Insurance: Biometric Identity Verification and AI-Based Risk Assessment. In 2024 International Conference on Knowledge Engineering and Communication Systems (ICKECS) (Vol. 1, pp. 1-6). IEEE.

#### APPENDICES

