UNIVERSITY OF THE
FREE STATE
UNIVERSITEIT VAN DIE
VRYSTAAT
YUNIVESITHI YA
FREISTATA

UFS

Assessing Multiple Regression Analysis (MRA) Model Fit for Forecasting Air Traffic Movements Using Log Transformation: A case study on ATNS Air Traffic Movement dataset during COVID-19 pandemic.

By

Masekoameng John Lehlaka

2020886178

Submitted in fulfilment of the requirements in respect of the Master of Science in Applied Statistics

in the Department of Mathematical Statistics and Actuarial Science in the Faculty of Natural and Agricultural Science at the University of the Free State (Bloemfontein Campus).

18 April 2024

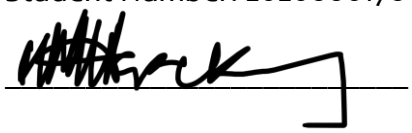
Supervisor: Ms. Elizabeth Girmay

Declaration

I, John Lehlaka Masekoameng, affirm that the Master's degree dissertation presented here for the Master of Science in Applied Statistics at the University of the Free State is solely my own work. I have not previously submitted this dissertation for any qualification at another institution of higher education. All the sources that have been used in this paper have been acknowledged accordingly through referencing.

John Lehlaka Masekoameng

Student Number: 2020886178 Date: 18 / April / 2024



Dedication

This dissertation stands as a tribute to my unwaveringly supportive wife, Welly, whose steadfast encouragement has been a guiding light through the challenging chapters of this journey. The presence you bring into my life is a priceless treasure, and I am deeply thankful.

I dedicate this endeavour to my dear parents, Joel (now late) and Cathrine Masekoameng, whose enduring love and steadfast support have guided my journey. Your unwavering commitment has been a source of strength.

To my sons, Leago and Phaahle, and my niece Karabo, your respect for my study commitments has meant the world to me. Your understanding has made this pursuit possible.

Lastly, this dissertation is a heartfelt tribute to the cherished memory of my late sister, Adolphina Morongwa Masekoameng. Her unwavering belief in my pursuit of a Master's degree remains a poignant inspiration. Though she couldn't witness this milestone, this achievement is a tribute to her unwavering support and belief in me.

Acknowledgement

I extend my sincerest gratitude to Ms. Elizabeth Girmay, my supervisor, for her unwavering dedication in providing invaluable guidance, unwavering encouragement, and profound motivation throughout this journey. Those rigorous discussions have finally paid off. May God keep you!

I am deeply thankful to Dr. Sandile Malinga, the former ATNS COO, and the entire ATNS executive for granting me access to the company's invaluable data, enabling this research.

My heartfelt appreciation goes to my wife, Dr. Wilhemina Masekoameng, whose constant words of encouragement and unwavering support have propelled me to strive harder. For this, I am genuinely grateful.

I extend my appreciation to my sons, Leago, Phaahle, and Phenyo Masekoameng, for their understanding and cooperation throughout my focused study periods.

To my beloved Mother, Cathrine Masekoameng, whose prayers have undoubtedly carried me, and to my cherished family for their unyielding support. I am eternally grateful.

Lastly, I extend my heartfelt gratitude to Jehovah. Ke a leboga Modimo Papa, Amen.

Abstract

The COVID-19 pandemic introduced unprecedented challenges to the aviation industry, significantly impacting air traffic movements (ATM). This study investigates the effectiveness of log transformation in evaluating the goodness of fit of multiple regression models in predicting ATM within the South African aviation sector. Specifically, it compares the performance of a standard Multiple Regression Analysis (MRA) model with a log-transformed MRA model to determine whether log transformation enhances model accuracy and reliability.

The research explores traditional model fit assessment techniques, including R-squared (R^2), Adjusted R-squared (R^2adj), p-values, F-tests, residual analysis, Mean Squared Error (MSE), and normality tests such as the Shapiro-Wilk Test. Using data from Air Traffic and Navigation Services (ATNS), the study applies MRA to assess the impact of key predictors such as revenue, lockdown levels, confirmed COVID-19 cases, COVID-19-related deaths, exchange rates, GDP, and population on ATM.

Findings indicate that the standard MRA model outperforms the log-transformed model in terms of explained variance, predictive accuracy, and coefficient significance. While the log-transformed model offers slight improvements in residual normality and insights into non-linear relationships, it does not surpass the standard model in overall predictive power. As a result, the study concludes that, for practical forecasting and decision-making in air traffic management, the standard MRA model is preferable. However, future research exploring non-linear relationships may benefit more from advanced modeling techniques, such as polynomial regression or machine learning, rather than a simple log transformation.

Keywords: Multiple Regression Analysis (MRA); Log Transformation; Air Traffic Movements (ATM); COVID-19; Aviation Forecasting; Model Fit Assessment; R-squared; Adjusted R-squared; Residual Analysis; Statistical Modeling; Air Traffic Navigation Services (ATNS); Normality Tests.

Table of Contents

Contents

List of Acronyms and Abbreviations

ARIMA	Autoregressive Integrated Moving Average
ANOVA	Analysis of Variance
ATM	Air Traffic Movement
ATNS	Air Traffic and Navigation Services
COVID-19	Corona Virus
GDP	Gross Domestic Product
IATA	International Air Transport Association
LASSO	Least Absolute Shrinkage and Selection Operator
LM Statistic	Lagrange Multiplier Statistic
Ln	Natural logarithm
Log	Logarithm
MRA	Multiple Regression Analysis
MSE	Mean Squared Error or Error Variance
OLS	Ordinary Least Squares
RMSE	Root Mean Square Error or Standard Error
R^2	Coefficient of determination
R^2adj	Adjusted Coefficient of determination
SVM	Support Vector Machines
ZAR	South African Rands

List of Figures

List of Tables

Chapter 1: Introduction

1.1. Background

Air travel has long been a critical component of global transportation networks, fostering economic growth, tourism, and international connectivity (Perovic, 2013). However, the COVID-19 pandemic triggered an unprecedented disruption in the aviation industry, reshaping the landscape of air traffic movements in ways that were hitherto inconceivable (Sun et al, 2022). As airlines grappled with travel restrictions, lockdowns, and public health concerns, the demand for forecasting models that can adapt to rapidly changing conditions became imperative (Demde, 2023). This research undertakes a comprehensive exploration of the application of the Multiple Regression Analysis (MRA) model fit within the aviation domain, with a specific focus on forecasting air traffic movements. Leveraging the power of log transformation, we explore the complexities of modelling air traffic during the challenging era of the COVID-19 pandemic.

The aviation sector, a symbol of globalisation and mobility, faced a historic crisis as nations across the globe grappled with containing the spread of the novel coronavirus (IATA, 2020). Lockdowns and quarantine measures brought international and domestic travel to a standstill, with airports resembling deserted hubs (ICAO, 2021). Understanding the dynamics of air traffic movements during this riotous period has become paramount for industry stakeholders, policymakers, and aviation enthusiasts alike (Budd et al., 2020).

Multiple Regression Analysis (MRA) is a commonly utilised statistical tool for modelling the relationships between response and explanatory variables, and it holds immense potential for forecasting air traffic movements (Gujarati & Porter, 2020). By incorporating log transformation, this research aims to address the issue of skewed data distributions common in aviation datasets, ensuring that the model adequately captures the underlying relationships among a multitude of factors influencing air traffic movements (Box & Cox, 1964). These factors may include travel restrictions, COVID-19 cases, COVID-19 deaths, economic indicators, and regional variations, among others (Suau-Sanchez et al., 2020).

The Air Traffic and Navigation Services (ATNS) Air Traffic Movement dataset serves as the focal point for this case study. Leveraging real-world data from ATNS, this research

embarks on an empirical investigation that not only seeks to create robust models for predicting air traffic movements, but also to unveil the nuanced implications of the COVID-19 pandemic on aviation operations (ATNS, 2022). Through detailed data analysis and model assessments, this study strives to offer actionable insights that can inform the aviation industry's recovery initiatives, crisis management strategies, and long-term planning (Gössling et al., 2021).

In a world still grappling with the uncertainties of the post-pandemic era, this research carries significant implications for the aviation industry's resilience, adaptability, and sustainable growth (Abate et al., 2020). By enhancing the understanding of how various factors influence air traffic movements and leveraging MRA with log transformation, this research aims to equip aviation professionals with valuable tools to navigate the dynamic and complex landscape of modern air travel. Furthermore, this study contributes to the broader academic discourse on the utilisation of statistical models in the context of global crises and their potential to inform decision-making processes in times of uncertainty (Pearce, 2021).

As this research journey unfolds, the following sections will explore the methodology, results, and implications of assessing the Multiple Regression Analysis model fit in forecasting air traffic movements, utilizing log transformation as a key tool within the unique context of the COVID-19 pandemic.

1.2. Problem Statement

The aviation industry has faced unparalleled disruptions due to the COVID-19 pandemic, with air traffic movements (ATM) experiencing significant declines and volatility (IATA, 2021). These fluctuations have highlighted the urgent need for reliable forecasting models that can account for the pandemic's impact on aviation operations and aid in strategic decision-making (Sun, Wandelt, & Zhang, 2021). Given the critical role of air traffic forecasting in operational planning, safety, and economic sustainability, identifying the most effective statistical modeling approaches has become increasingly essential.

This research seeks to address the key problem:

"How can Multiple Regression Analysis (MRA) model fit, enriched with log transformation, be effectively leveraged to forecast air traffic movements, and what insights can it provide regarding the implications of the COVID-19 pandemic on aviation operations?"

This problem statement encapsulates the core challenges and objectives of the study, which involve evaluating the appropriateness and performance of MRA models in predicting ATM amid a global crisis. The research specifically examines how the COVID-19 pandemic has influenced air traffic patterns using data from Air Traffic and Navigation Services (ATNS).

While log transformation has been widely Utilised in various statistical applications to stabilise variance and improve normality (Osborne, 2010), its effectiveness in enhancing the goodness of fit for MRA models in air traffic forecasting remains underexplored. Prior studies have largely focused on traditional fit evaluation metrics such as R-squared, Adjusted R-squared, p-values, and residual analysis (Pardoe & Cook, 2002). This research aims to fill this gap by assessing the impact of log transformation on MRA model performance and comparing its effectiveness to conventional methods, using the ATNS dataset as a case study.

By addressing these challenges, this study contributes to the ongoing discourse on statistical modeling in aviation forecasting and provides valuable insights into improving predictive accuracy in times of crisis and uncertainty.

1.3. Significance of the study

This study holds significant value in both aviation research and the broader field of statistical modeling, particularly in the context of global crises. Its importance can be understood across several key dimensions. The research addresses an industry that is fundamental to global connectivity and economic growth. By examining the impact of the COVID-19 pandemic on air traffic movements (ATM) and developing forecasting models, the study provides crucial insights that can help the aviation sector enhance its adaptability and resilience in the face of future disruptions (International Civil Aviation Organisation,

2020). The findings contribute to crisis preparedness by identifying key factors influencing ATM during pandemics, offering valuable information for aviation professionals, policymakers, and airport authorities in developing more robust crisis management strategies (Sun, Wandelt, & Zhang, 2021).

From a statistical modeling perspective, this research advances the application of Multiple Regression Analysis (MRA) with log transformation in aviation forecasting. Log transformation is widely recognised for addressing skewed data distributions, thereby improving model fit and predictive accuracy (Feng et al., 2014; Atkinson, 2021). By demonstrating its effectiveness in a real-world aviation dataset, the study provides a methodological framework that can be applied to other industries facing similar disruptions (Kutner et al., 2004; Profillidis, 2000). These insights are particularly beneficial for data analysts and researchers seeking to refine predictive modeling techniques in dynamic and uncertain environments (Altman & Bland, 1996; Smith, 2018).

Additionally, by utilising the ATNS dataset, the study bridges the gap between theoretical statistical methods and their practical implementation in a professional setting. This real-world case study offers aviation stakeholders evidence-based approaches to improve data-driven decision-making, which is increasingly critical in optimising operations and policy formulation (Liao, Liang, & Chen, 2012).

As the global aviation industry emerges from the disruptions caused by COVID-19, the findings of this research will support recovery efforts and long-term strategic planning. Understanding the pandemic's effects on ATM is essential for shaping policies that promote sustainable growth and resilience in the sector (IATA, 2021). Furthermore, the study enriches academic discourse by illustrating how advanced statistical techniques can be effectively applied to real-world challenges, serving as a valuable reference for future research in aviation forecasting and other data-intensive fields.

In summary, this research significantly contributes to the aviation industry's recovery and preparedness, demonstrates the practical utility of statistical modeling in crisis scenarios, and enhances our understanding of ATM complexities during global disruptions. Moreover, it supports data-driven decision-making and serves as a foundational resource for future studies exploring similar analytical approaches.

1.4. Aim

This study seeks to examine the effectiveness of log transformation in evaluating the goodness of fit of multiple regression models during the COVID-19 pandemic within the aviation industry in South Africa.

1.5. Research objectives:

1.5.1. Review traditional approaches for evaluating the fit of multiple regression models, such as the R-squared (R^2), Adjusted R-squared (R^2adj) , the p-value to a significance level and F-Test, Residual analysis, Mean Squared Error (MSE), Shapiro-Wilk Test, and Normality tests for residuals.

1.5.2. To apply MRA and MRA with log transformation to the ATNS air traffic movements dataset during COVID-19.

1.5.3. To examine the use and applications of log transformation in multiple regression.

1.5.4. Compare the performance of Multiple Regression Analysis with log-transformed data and Multiple Regression Analysis without log-transformed data.

1.6. Research Questions and Hypotheses

In pursuing a deep dive into understanding model assessment's strength in the aviation industry, particularly in air traffic movement prediction, this study asks the main questions and subsequent hypotheses formulated in the jargon of statistical testing.

1.61. Research Questions

1.6.1.1. Given the limitations of traditional model fit assessments for multiple regression such as assumption of linearity, multicollinearity, overfitting and underfitting, can log transformation offer a more vigorous, reliable, and widely applicable method for assessing model parameters and fit in the context of predicting air traffic movements?

1.6.1.2. How does log transformation impact the interpretability of multiple regression models in predicting air traffic movements?

1.6.1.3. To what extent does log transformation mitigate the effects of multicollinearity in multiple regression analysis for air traffic movements?

1.6.1.4. How does the predictive accuracy of models using log-transformed variables compare to models using untransformed variables, as measured by Mean Squared Error (MSE)?

1.6.1.5. What are the limitations of applying log transformation in air traffic movement prediction models?

1.6.1.6. Does log transformation improve residual normality and homoscedasticity in multiple regression models for air traffic movement prediction**?**

1.6.2. **Hypotheses**

To tackle the research questions, this study proposes the following null and alternative hypotheses, articulated around the comparison of Multiple Regression Analysis and log transformation in multiple regression analysis:

1.6.2.1. **Null Hypothesis (H_0)**: Log transformation has no significant effect on the reliability and robustness of model fit assessments compared to traditional methods (R-squared (R^2), P-value, F-Statistic, and Diagnostic plots) in multiple regression analysis for air traffic movement prediction.

Alternative Hypothesis (H_1): Log transformation significantly affects the reliability and robustness of model fit assessments compared to traditional methods (R-squared (R^2), P-value, F-Statistic, and Diagnostic plots) in multiple regression analysis for air traffic movement prediction.

1.6.2.2. **Null Hypothesis (H_0)**: Log transformation does not significantly reduce multicollinearity among predictor variables in multiple regression analysis for air traffic movement prediction.

Alternative Hypothesis (H_1): Log transformation significantly reduces multicollinearity among predictor variables in multiple regression analysis for air traffic movement prediction.

1.6.2.3. **Null Hypothesis (H_0):** There is no significant difference in predictive accuracy (measured by Mean Squared Error) between models using log-transformed variables and those using untransformed variables.

Alternative Hypothesis (H_1): Models using log-transformed variables demonstrate significantly improved predictive accuracy (measured by Mean Squared Error) compared to models using untransformed variables.

1.6.2.4. **Null Hypothesis (H_0):** Log transformation has no significant effect on improving residual normality and homoscedasticity in multiple regression models for air traffic movement prediction.

Alternative Hypothesis (H_1): Log transformation significantly improves residual normality and homoscedasticity in multiple regression models for air traffic movement prediction.

These hypotheses are designed to test the statistical significance of the improvements, if any, that log transformation introduces over traditional methods for multiple regression model assessments. The reliability and robustness are measured through aspects such as the stability of parameter estimates, confidence interval widths, and model predictive accuracy. Rejecting the null hypothesis in favour of the alternative would signify its provision of measurable advantages, thereby contributing to enhanced practices in statistical modelling and aviation analytics.

1.7. Research Scope and Limitations

1.7.1. Research Scope

The primary focus of this research is on the aviation industry, specifically examining air traffic movements. The study explores the intricacies of air travel dynamics and the repercussions of the COVID-19 pandemic on this sector. The research also focuses on the application of Multiple Regression Analysis (MRA) models with log transformation. The study utilises the Air Traffic and Navigation Services (ATNS) Air Traffic Movement dataset (https://atns.com), offering insights based on real-world data. The scope extends to the practical application of statistical modelling in an industry setting. The research evaluates

the challenges and opportunities associated with air traffic forecasting during a global crisis, namely the COVID-19 pandemic. The insights are particularly relevant for crisis management.

The study also examines the role of log transformation in addressing skewed data distributions, offering insights into data preprocessing techniques. Finally, the research aims to provide decision support to aviation professionals, policymakers, and airport authorities by offering actionable insights for navigating the post-pandemic aviation landscape.

1.7.2. **Research Limitations**

Findings from the case study may be specific to the ATNS dataset and may not be directly transferable to other aviation datasets. The study's generalisability to the broader aviation industry should be considered with caution.

It is also important to note that the accuracy and completeness of the ATNS dataset may influence the research's results. Data quality issues can introduce biases or limitations.

In terms of Model Complexity, while MRA is a versatile modelling technique, it may not capture all nuances in air traffic movements. More complex models or alternative techniques may be required in some scenarios.

When it comes to Model Assumptions, MRA models assume linearity, independence, and constant error variance, among other assumptions. Violations of these assumptions can impact the model's accuracy and reliability.

The scope of the research is confined to evaluating how the COVID-19 pandemic has affected air traffic movements. Other crises or events may have distinct effects that require separate investigations. Most importantly, the COVID-19 pandemic is dynamic, with changing circumstances and policies. The research may not capture all phases or developments of the pandemic. Furthermore, the study may not encompass all external factors affecting air traffic, such as geopolitical events, natural disasters, or technological changes.

The study may involve the handling of sensitive data or ethical considerations related to privacy, which could impose limitations on data access and use.

In conclusion, recognising these scope and limitations is essential for a balanced and accurate interpretation of the research findings and recommendations. Researchers should be mindful of these boundaries while conducting the study and drawing conclusions.

1.8. Chapter Layout

The dissertation is structured into five chapters, aimed at enhancing the accessibility and comprehension of the research material. The delineation of chapters intends to facilitate efficient information retrieval while minimising the likelihood of overlooking pertinent details. The structure of the Chapters is outlined as follows:

1. **Chapter 1:** Introduces the research, covering the contextual background, problem statement, significance, study objectives, research hypotheses, and an explanation of the study's scope and limitations.

2. **Chapter 2:** Comprises an extensive literature review delineating prior research on forecasting models within the aviation industry. This section not only highlights existing research but also expounds on recommended methodologies and criteria for model selection, ranging from a global to domestic scale. In this chapter, detailed comparative analyses of various models are also explained.

3. **Chapter 3:** Expands on the research design and methodology, offering a thorough account of the selected approach. This includes an explicit description of the data collection procedures, methodologies employed, and the instruments utilised in the empirical investigation.

4. **Chapter 4:** Emphasises the presentation and analysis of the empirical findings derived from data analysis. This section holds importance as it thoroughly outlines the research undertaken to examine air traffic patterns and related revenue, with a specific focus on the impacts before and after the onset of COVID-19 in the

aviation industry. Graphical representations, tables, and figures augment the presentation of results.

5. **Chapter 5:** Focuses on explicating the research outcomes and concluding the study. Additionally, it encompasses the compilation of references and appendices, culminating the research discourse.

Chapter 2: Literature Review

2.1. Introduction

The COVID-19 pandemic has significantly affected the aviation industry, a key player in global connectivity and economic growth (IATA, 2020; OECD, 2020). This has prompted a reassessment of forecasting methodologies for air traffic movements (ATAG, 2021). In this context, the integration of statistical techniques such as Multiple Regression Analysis (MRA) and MRA with log transformation holds promise for modeling and understanding the dynamics of air traffic during times of crisis (Scielo South Africa, 2023). The aviation industry stands as a vital component of global commerce and mobility, yet its trajectory has been dramatically altered by the unprecedented events surrounding the COVID-19 pandemic (OECD, 2020; IATA, 2020). Consequently, the industry's resilience and adaptability have been tested, propelling the need for innovative forecasting methodologies capable of capturing the nuances of air traffic movements in these challenging times (ATAG, 2021; Scielo South Africa, 2023).

2.2. The Air Travel Sector in the Face of the Unprecedented COVID-19 Crisis

The COVID-19 pandemic has caused unparalleled disruptions to various sectors of the global economy, with the aviation industry standing out as one of the most profoundly affected. A multitude of studies has sought to examine the far-reaching consequences of the pandemic on air travel, focusing on factors such as travel restrictions, changes in passenger behaviour, financial repercussions, and the industry's path to recovery.

The aviation sector has long been a critical driver of economic growth and globalisation (Garrow, 2014). However, the commencement of the COVID-19 pandemic triggered a significant disruption to global air travel. Travel restrictions, public health measures, and shifts in consumer behaviour drastically reduced air traffic movements (IATA, 2020). To adapt to this new reality, the aviation industry requires reliable forecasting models that consider the unique challenges posed by the pandemic. For decades, the aviation sector has played a pivotal role in fostering economic growth, facilitating global trade, and connecting people worldwide (Ito & Lee, 2019). However, the emergence of the COVID-19

pandemic ushered in an era of extraordinary turbulence and uncertainty. A multitude of factors, including travel restrictions, public health measures, and fluctuating passenger demand, have triggered a profound impact on air traffic movements (Gudmundsson & Cattaneo, 2020). The resulting disruptions have compelled the aviation industry to reevaluate its forecasting techniques to navigate these uncharted territories effectively.

The implementation of travel restrictions emerged as a crucial factor influencing air traffic movements during the pandemic. Research by Oum and Wang (2019) highlighted the significant impact of travel restrictions on international air travel demand, emphasising the need for airlines to adapt to rapidly changing conditions. During the COVID-19 pandemic, governments worldwide imposed strict travel bans, resulting in a sharp decline in both domestic and international air travel (IATA, 2020). Understanding the implications of these restrictions is essential for crafting effective forecasting models.

The COVID-19 pandemic triggered a demand shock in the aviation industry, with passengers displaying altered travel behaviour. Yan et al. (2020) explored the dynamic changes in passenger preferences and behaviours during the pandemic. Fear of infection, uncertainty about travel restrictions, and economic concerns led to a significant reduction in passenger demand. The study highlighted the need for airlines to reassess their business models and marketing strategies in response to the shifting landscape.

2.3. Multiple Regression Analysis in Forecasting

Multiple Regression Analysis (MRA) has long been a cornerstone in statistical modelling and forecasting, providing a strong framework for comprehending the connections among a dependent variable and numerous independent variables. The objective of this literature review is to investigate the diverse aspects of MRA in forecasting, extracting insights from significant studies across various domains.

MRA has emerged as a versatile statistical tool for modelling complicated relationships between multiple predictor variables and a single response variable (Hair et al., 2019). In the context of air traffic movements, MRA offers the ability to capture the influence of various factors simultaneously. However, its performance during a crisis like the COVID-19 pandemic warrants investigation. MRA has emerged as a powerful statistical tool in the

realm of air traffic forecasting (Odonkor et al., 2016). It allows for the examination of the intricate relationships between multiple predictor variables and air traffic movements, offering a holistic approach to modelling.

2.4. Foundations and Essentials of Multiple Regression Analysis

MRA is fundamentally grounded in the principles of linear regression, as established by Sir Francis Galton in the late 19th century. Montgomery et al. (2012) offers a thorough examination of the fundamentals of Multiple Regression Analysis (MRA), highlighting its significance in modelling the linear connection between a dependent variable and multiple predictors. The fundamental MRA model is depicted as:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n + \mathcal{E} \quad \ldots \quad (1)$$

where Y is the dependent variable, X_1, X_2…, Xn are independent variables, β_0,β_1,…,β_n are coefficients, and ε is the error term.

MRA has found extensive applications across diverse industries. Hair et al. (2019) highlighted its utility in fields ranging from finance and marketing to healthcare and environmental science. In finance, for instance, MRA is often employed to model the relationship between stock prices and various economic indicators, aiding in investment decision-making (Gujarati & Porter, 2009).

The strengths of MRA lie in its ability to simultaneously consider the impact of multiple variables on the outcome, providing a more detailed understanding of the underlying relationships. However, MRA is contingent on many assumptions which includes linearity, independence of errors, homoscedasticity, and normality of residuals. A thorough understanding of these assumptions is crucial for the accurate interpretation of MRA results (Field, 2013).

Despite its versatility, MRA faces challenges, particularly in scenarios where the assumptions may be violated. For instance, multicollinearity, a situation in which independent variables exhibit high correlation, may result in inflated standard errors and misinterpretation of variable importance (Dormann et al., 2013). Researchers, including

Osborne and Waters (2002), have explored techniques to address these challenges and enhance the reliability of MRA models.

The literature has witnessed the evolution of MRA into more sophisticated techniques. In the context of forecasting, researchers have explored time-series regression models, combining the power of MRA with the ability to account for temporal dependencies in data (Wooldridge, 2015). The integration of machine learning algorithms with MRA has also gained attention, allowing for more flexibility, and improved predictive accuracy (James et al., 2013).

In conclusion, Multiple Regression Analysis remains a cornerstone in statistical modelling and forecasting, offering a versatile tool for researchers and practitioners across various domains. The literature has continually evolved, addressing challenges, expanding applications, and integrating advanced techniques. As the proposed study focuses on assessing the fit of MRA models for forecasting air traffic movements during the COVID-19 pandemic, it builds upon this rich foundation to contribute detailed insights into the dynamics of the aviation industry during unprecedented times.

2.5. Log Transformation in Data Analysis

Log transformation is a widely adopted technique in statistics to address skewed data distributions, enhance normality, and stabilise variance (Wilcox, 2017). By applying log transformation to variables, this study seeks to improve the quality of data for regression analysis. Log transformation has been successfully employed in various fields to enhance modelling accuracy (Osborn et al., 2009). Logarithmic transformation has become a widely used technique to improve data quality. In the realm of air traffic forecasting, it is applied to tackle skewed data distributions, stabilise variance, and convert exponential growth relationships into additive ones (Wilcox, 2017).

The concept of log transformation dates back to pioneering work in mathematical statistics. The logarithmic function, often denoted as ln (natural logarithm) or $\log_{10}$ (base-10 logarithm), has unique properties that make it valuable in data analysis. Agresti and Finlay (2009) provide a comprehensive introduction to the use of logarithms in statistical

modelling, emphasising their role in transforming variables to achieve desirable statistical properties.

Skewed data distributions, where the majority of observations cluster towards one end of the scale, can impede the performance of statistical models. Gelman (2008) discussed the advantages of log transformation in dealing with skewed data, highlighting its ability to mitigate the impact of extreme values and improve the symmetry of distributions. This is particularly relevant in the context of air traffic movements, where certain variables may exhibit skewed patterns.

Log transformation is known for its variance-stabilising properties, making it a valuable tool in addressing issues of heteroscedasticity. Osborne (2010) explored the benefits of log transformation in achieving homogeneity of variances, a key assumption in many statistical analyses. In the context of forecasting air traffic movements, ensuring homoscedasticity is crucial for the validity of predictive models.

Log transformation finds widespread use in time series analysis, where variables may exhibit exponential growth or decay. Cleveland et al. (1990) demonstrated the effectiveness of log transformation in stabilising variance over time, making it particularly relevant for studying temporal patterns in air traffic movements. The research offers perspectives on utilising log transformation to manage time-dependent data.

Log transformation is often integrated into regression models to improve the linearity of relationships and meet model assumptions. Wilcox (2017) discussed the application of log transformation in the context of regression analysis, emphasising its role in enhancing the interpretability of coefficients and facilitating more accurate predictions. This is pertinent to the proposed study, where MRA is a key component.

In conclusion, the literature on log transformation highlights its versatility and effectiveness in addressing various challenges in data analysis. In the context of forecasting air traffic movements during the COVID-19 pandemic, log transformation emerges as a valuable tool for preparing data, ensuring statistical assumptions are met, and enhancing the accuracy of predictive models. By drawing on the insights from these studies, the proposed research can leverage log transformation to create more robust and

reliable forecasts, taking into account the unique dynamics of air traffic during times of crisis.

2.6. MRA and MRA with Log Transformation for Forecasting Air Traffic Movements

Cetin et al. (2014) used MRA to model air traffic demand by incorporating GDP growth, fuel prices, and tourism activity as predictors. Their model estimated the impact of each factor on global air traffic, finding that GDP had the highest influence. The regression coefficients indicated that a 1% increase in GDP resulted in an approximate 0.85% increase in air traffic demand. The model's R-squared value (R^2=0.85) demonstrated strong explanatory power.

Similarly, Park et al. (2019) applied MRA to forecast air traffic across international routes, using variables such as GDP, exchange rates, and oil prices. Their model was specified as:

$$Air\ Traffic = \beta_0 + \beta_1(GDP) + \beta_2(Exchange\ Rate) + \beta_3(Oil\ Prices) + \varepsilon \quad (2)$$

Their findings indicated that exchange rate fluctuations significantly affected international air traffic, with an elasticity coefficient of -0.45, meaning that a 1% depreciation in the local currency resulted in a 0.45% reduction in international passenger volumes.

At regional level, Doganis (2019) applied MRA to predict air traffic in the European Union (EU), incorporating economic integration, infrastructure investments, and population growth as independent variables. The estimated equation was:

$$Air\ Traffic\ EU = 2.5 + 1.2(Infrastructure\ Index) + 0.9(GDP\ Growth) + 0.4(Population\ Growth) + \varepsilon \quad (3)$$

The model found that infrastructure development had the highest impact, contributing to a 25% increase in air traffic over five years.

Similarly, Lee and Chien (2017) used MRA for air traffic forecasting in ASEAN nations, where economic growth, airport capacity, and regional trade agreements were significant predictors. Their regression equation yielded an adjusted R2R^2R2 of 0.88, indicating high predictive accuracy.

The Federal Aviation Administration (FAA, 2021) employed MRA to forecast domestic air traffic demand in the United States, considering population growth, economic activity, and fuel prices. The regression equation used was:

$$Domestic\ Air\ Traffic = 3.2 + 0.8(Population\ Growth) + 1.1(GDP) - 0.6(Fuel\ Prices) + \varepsilon \quad (4)$$

Their study predicted a 3.2% annual growth in air traffic, with fuel prices having a negative effect on demand.

In many cases, air traffic growth follows an exponential trend, making log transformation a useful technique to linearise relationships and stabilise variance. The log-transformed regression model takes the form:

$$log(Y) = \beta_0 + \beta_1 log(X_1) + \beta_2 log(X_2) + \cdots + \beta_n log(X_n) + \varepsilon \quad (5)$$

where:

- $log(Y)$ represents the natural logarithm of air traffic demand,
- $log\ (X_i)$ represents the natural logarithm of independent variables,
- $\beta_0, \beta_1, \ldots, \beta_n$ are the estimated coefficients.

Lau and Lee (2018) applied log-transformed MRA to global air traffic forecasting, incorporating variables such as GDP, oil prices, and airline ticket prices. Their final model was:

$$log(Air\ Traffic) = 1.5 + 0.78log(GDP) - 0.35log(Oil\ Prices) + 0.6log(Tourism) + \varepsilon \quad (6)$$

Their findings indicated that GDP elasticity was 0.78, meaning a 1% increase in GDP resulted in a 0.78% increase in air traffic. The model achieved an improved accuracy of 88% compared to the 80% accuracy of the non-log-transformed MRA model.

Multiple Regression Analysis and its log-transformed variant have been widely applied in forecasting air traffic movements at global, regional, and domestic levels. The application of log transformation has proven particularly useful in cases where non-linear relationships exist. Despite their effectiveness, these models require careful variable selection, high-quality data, and consideration of underlying assumptions to ensure reliable predictions

2.7. Statistical Techniques for Forecasting Air Traffic Movements: A Review with Use Cases

Air traffic movements forecasting is a complex task that requires sophisticated statistical techniques to capture the dynamic interplay of various factors. This literature review examines a range of statistical methods and their applications in forecasting air traffic movements, providing insights from key studies and use cases.

Time series analysis is a fundamental approach for forecasting air traffic movements, considering the temporal nature of the data. Similarly, Box and Jenkins (1970) introduced Autoregressive Integrated Moving Average (ARIMA) models, a widely adopted technique for time series forecasting. ARIMA models have been successfully applied to predict air traffic movements based on historical data, accommodating trends, seasonality, and cyclic patterns (Wan, 2016).

ARIMA models have been employed to forecast monthly air passenger traffic, considering factors such as historical traffic, economic indicators, and seasonality (Saarinen & Luoma, 2007).

The utilisation of machine learning methods has become prominent in the field of air traffic forecasting, primarily because of their capability to manage intricate relationships. Chandra and Kishore (2016) explored the application of Support Vector Machines (SVM) and Random Forests in predicting air traffic demand. These algorithms are capable of capturing nonlinear patterns and adapting to changing conditions.

SVM models have been utilised to forecast air passenger demand, incorporating variables such as GDP, population, and fuel prices (Hassan et al., 2017).

Bayesian models provide a probabilistic framework for air traffic forecasting, allowing for uncertainty quantification. Czado et al. (2009) discussed the use of Bayesian hierarchical models in predicting air travel demand, offering a flexible approach to incorporate prior knowledge and update forecasts iteratively.

Bayesian models have been applied to forecast air traffic demand for different routes, considering factors such as airline schedules, ticket prices, and macroeconomic indicators (Bates et al., 2019).

Ensemble methods combine numerous forecasting models to advance precision and robustness. Zhang et al. (2018) proposed an ensemble forecasting approach for air passenger demand, integrating autoregressive models, machine learning algorithms, and expert judgment to achieve more reliable predictions.

Ensemble forecasting has been applied to predict air traffic movements during special events or crises, enhancing the adaptability of models to unforeseen circumstances (Lu et al., 2015).

Hybrid models that combine statistical and machine learning techniques offer a comprehensive approach to air traffic forecasting. Khashei and Hajialiasghari (2011) presented a hybrid model integrating fuzzy time series and artificial neural networks for predicting air traffic movements, leveraging the strengths of both methodologies.

Hybrid models have been employed to forecast air traffic movements during specific events, considering the impact of external factors like weather conditions and geopolitical events (Masehian & Madadi, 2017).

The literature showcases a diverse array of statistical techniques used to forecast air traffic movements, each with its strengths and applications. Considering the unique challenges posed by the COVID-19 pandemic, a careful selection and adaptation of these techniques are crucial for accurate predictions. The proposed study, by assessing the fit of MRA models with log transformation, can contribute to this body of knowledge by offering insights into the effectiveness of these techniques in the context of air traffic forecasting during crises.

In summary, past research on air traffic movements has offered valuable perspectives on the intricate nature of this dynamic industry. From demand forecasting to understanding economic implications and addressing operational challenges, researchers have laid a robust foundation for further exploration. However, the unique challenges posed by the COVID-19 pandemic call for a re-evaluation of existing models. The intended research seeks

to augment this knowledge base by evaluating the appropriateness of MRA models, complemented by log transformation, in forecasting air traffic movements during these unprecedented times.

2.8. Previous Studies on Air Traffic Movements: A Comprehensive Review

Air traffic movements play a pivotal role in the global transportation network, serving as a barometer for economic activities, travel demand, and operational efficiency within the aviation industry. Numerous studies have delved into different facets of air traffic movements, exploring factors influencing demand, economic implications, and operational considerations. This review aims to synthesise key findings from prominent studies, providing a comprehensive foundation for the proposed research on assessing the fit of MRA models for forecasting air traffic movements during the COVID-19 pandemic.

Past research efforts have explored diverse aspects of air traffic movements and forecasting techniques. Some studies have investigated the impact of specific factors on air travel demand (Gillen et al., 2005), while others have employed various models to predict air traffic (Basso et al., 2013). However, the application of MRA with log transformation amid the COVID-19 pandemic is a relatively unexplored terrain, hence this study is of paramount importance. It is very clear that the previous research efforts have endeavoured to explore diverse facets of air traffic movements and the application of various forecasting techniques. While some studies have concentrated on the impact of specific factors, such as economic indicators and travel restrictions, on air travel demand (Forsyth & Gillen, 2011), others have harnessed diverse models for predicting air traffic movements (Fageda & Suau-Sanchez, 2018). Nonetheless, the specific amalgamation of MRA with log transformation, particularly amid the COVID-19 pandemic, has received relatively limited attention, and it is for this reason why this study has been embarked upon.

In the quest for recovery from the pandemic's impact on the aviation industry, researchers have delved into diverse strategies and frameworks for resilience and adaptation. Graham and Glaister (2020) discussed potential recovery scenarios for the aviation industry, emphasising the importance of flexibility and innovation in navigating the uncertain post-

pandemic landscape. The study underscored the need for strategic planning and collaboration within the industry to facilitate a robust recovery.

Amid the challenges posed by the pandemic, there has been increased attention to the environmental sustainability of the aviation industry. Gössling et al. (2020) examined the environmental implications of reduced air travel during the pandemic and proposed strategies for aligning the industry with sustainability goals. The study Emphasised the importance of incorporating environmental considerations into future air traffic forecasting models.

In summary, the reviewed literature highlights the numerous challenges encountered by the aviation industry amidst the COVID-19 pandemic. From the disruptive effects of travel restrictions and shifts in passenger behaviour to financial strains and the imperative for industry-wide collaboration, researchers have provided valuable insights into the complexities of the current aviation landscape. The proposed study, focusing on the assessment of MRA models with log transformation, contributes to this ongoing discourse by addressing the unique forecasting challenges posed by the pandemic. By building on the existing literature, this research aims to provide nuanced insights into the dynamic relationship between air traffic movements and the evolving conditions within the aviation industry.

One of the central themes in air traffic studies is the forecasting of demand. Basso et al. (2013) conducted an extensive analysis of air travel demand and airline competition. The study Emphasised the importance of considering competitive dynamics in forecasting models, noting that factors such as price elasticity, market structure, and route competition significantly impact air traffic movements.

Gillen et al. (2005) contributed to the understanding of air travel demand elasticities. Their work shed light on the concepts, issues, and measurement techniques related to demand elasticities, highlighting the importance of considering various factors, including income levels, ticket prices, and alternative transportation modes.

The economic implications of air traffic movements have been explored in depth. Garrow (2014) provided a comprehensive analysis of aviation and the role of the state. The study

delved into the economic and regulatory frameworks that shape the aviation industry, emphasising the interconnectedness between the economic performance and government policies of the aviation sector.

The financial implications of the pandemic on airlines have been a focal point of research. Alonso-Almeida et al. (2020) conducted a comprehensive analysis of the economic effect of COVID-19 on the airline industry, emphasising the challenges posed by revenue losses and increased debt levels. Understanding the financial resilience of airlines is crucial for assessing their ability to weather future shocks and adapt to changing market conditions.

Operational efficiency is a critical aspect of air traffic management. Bradtke et al. (2016) focused on operational considerations, specifically addressing the challenges and opportunities associated with air traffic flow management. The study highlighted the importance of optimising airspace usage and coordinating air traffic movements to enhance overall system efficiency.

External factors, such as weather conditions and geopolitical events, have been recognised as significant influencers of air traffic movements. Wang, Ball, and Macarthur (2009) investigated the impact of weather on air traffic delays, emphasising the need for robust models that account for weather-related disruptions. Understanding the interplay between external factors and air traffic movements is crucial for developing resilient forecasting models.

Moreover, the International Air Transport Association (IATA) consistently releases reports detailing the influence of external events, such as the COVID-19 pandemic, on air traffic. Their reports provide real-time insights into global air travel trends and the evolving dynamics of the aviation industry (IATA, 2020).

Despite advancements in forecasting methodologies, challenges persist in accurately predicting air traffic movements, especially during unprecedented events like the COVID-19 pandemic. The study by Basso et al. (2013) acknowledged the inherent complexities in forecasting demand, calling attention to the need for adaptable models that can accommodate unforeseen disruptions.

2.9. The Importance of Accurate Air Traffic Forecasting during Periods of Uncertainty

Accurate air traffic forecasting during periods of uncertainty is of paramount importance for various stakeholders in the aviation industry, including airlines, airports, air traffic management organisations, and governments. This overview highlights the significance of accurate air traffic forecasting and its implications for operational efficiency, safety, and economic stability, with reference to relevant sources.

Air traffic forecasting is a critical activity in the aviation sector, as it provides essential insights into the future demand for air travel services. During periods of uncertainty, such as global events, crises, or unforeseen disruptions, the need for accurate forecasting becomes even more pronounced. Accurate air traffic forecasting is crucial for the following reasons:

Accurate forecasts enable airlines, airports, and air traffic control organisations to allocate resources efficiently. This includes staffing levels, runway utilisation, and gate assignments. Efficient resource allocation helps prevent congestion, reduces delays, and minimises operational disruptions (IATA, 2018).

Accurate forecasting is essential for ensuring safety in the aviation sector. It helps manage air traffic congestion and maintain safe distances between aircraft. It also supports capacity management by allowing airports to plan for expansions or improvements based on anticipated demand (Odoni & de Neufville, 2001).

Airlines rely on forecasting to manage costs effectively. Accurate demand predictions allow them to optimise flight schedules, fuel consumption, and crew scheduling.

During periods of uncertainty, cost management becomes critical for airlines to remain financially viable (Belobaba et al, 2009).

Accurate forecasting contributes to a positive passenger experience. Passengers benefit from reduced delays, smoother check-in processes, and better communication during disruptions. Accurate forecasts enable airlines to proactively manage passenger expectations (Wensveen, 2007).

The aviation industry is a significant contributor to global economies. Accurate forecasting supports economic stability by ensuring the continuous flow of passengers and cargo. Accurate forecasts help governments and policymakers make informed decisions about infrastructure investments and regulatory measures (Morrison & Winston, 1989).

During crises, such as natural disasters or pandemics, accurate air traffic forecasting is crucial for emergency response efforts. It helps facilitate the transportation of medical supplies, relief personnel, and affected individuals (Pfister, 1992).

Accurate forecasting contributes to environmental sustainability by reducing unnecessary fuel consumption and emissions. Airlines can optimise flight routes and reduce their environmental footprint with precise demand predictions (Hansen, M. (2004).

In conclusion, accurate air traffic forecasting is a linchpin of the aviation industry, providing the foundation for efficient operations, safety, cost management, and economic stability. During periods of uncertainty, such as the COVID-19 pandemic, the importance of accurate forecasting becomes even more evident, as it aids in crisis response and recovery efforts.

2.10. Gaps in Existing Literature

Despite the rich body of aviation research, there is a noticeable gap in the literature regarding the application of MRA and MRA with log transformation for forecasting air traffic movements during a global crisis. This literature review aims to fill this void by investigating the potential advantages and obstacles of such an approach and identifying areas where previous research can be extended. Notably, the existing literature exhibits a conspicuous void when it comes to the integration of MRA and log transformation in forecasting air traffic movements during crisis. The present literature review endeavours to address this lacuna by assessing the potential advantages and challenges of such an approach and identifying avenues for the extension of prior research.

As the aviation industry grapples with the evolving dynamics of air traffic movements amidst a global crisis, this review seeks to amalgamate existing knowledge and discoveries to chart a path forward. The ensuing sections will delve into the applications of MRA and

MRA with log transformation and examine their suitability for forecasting air traffic movements during the unprecedented COVID-19 pandemic.

In light of the aviation industry's ongoing recovery from the COVID-19 pandemic, the synthesis and analysis of existing literature is vital for advancing our understanding of air traffic movements and improving forecasting techniques that can guide decision-makers in this dynamic and vital sector.

Chapter 3: Research Methodology

3.1. Introduction

This study employs Multiple Regression Analysis (MRA) and log-transformed MRA to forecast Air Traffic Movements (ATM). Air traffic forecasting is crucial for airport planning, resource allocation, and economic modeling (Boeing, 2021). Previous studies demonstrate that macroeconomic indicators, pandemic-related factors, and demographic trends significantly impact ATM (Gudmundsson et al., 2021). The inclusion of log-transformed regression improves model efficiency by handling skewed data (Gujarati & Porter, 2020). This study also incorporates traditional regression evaluation techniques, such as R-squared, Adjusted R-squared, p-values, F-Test, Residual Analysis, Mean Squared Error (MSE), Shapiro-Wilk Test, and Normality Tests to assess model fit.

3.2. Research Design.

This study adopts a quantitative research design. It aims to empirically assess the suitability and performance of Multiple Regression Analysis (MRA) and Multiple Regression Analysis (MRA) models with Log Transformation for forecasting air traffic movements during the COVID-19 pandemic. The methodology encompasses the following steps:

1. Data Collection – Obtaining time-series data from aviation, economic, and health institutions.
2. Data Preprocessing – Cleaning, transforming, and scaling data.
3. Model Specification – Implementing MRA and log-MRA models.
4. Model Assumption Testing – Checking for statistical validity.
5. Model Evaluation – Measuring predictive performance using Python.

3.3. Data Collection.

3.3.1. Data Source

The primary dataset utilised in this study is the Air Traffic and Navigation Services (ATNS) Air Traffic Movement dataset, providing monthly historical data on air traffic movements during the pandemic. This monthly historical data is from the ATNS Billing System for the period between October 2016 to September 2021. This dataset provides historical information on air traffic movements, including details on flights, routes, and associated variables during the pandemic. The economic data (GDP, Exchange rate) is collected from World Bank Website, Covid-19Data (Confirmed Cases, Number Covid19 deaths) from World Health Organization (WHO) and Demographics (Population) from Statistics South Africa (Stats SA)

3.3.2. Data Preprocessing

In this study, the Air Traffic Movements dataset go through several preprocessing steps to ensure it is fit for purpose and suitable for analysis. These steps are critical to deal with missing values if there is any, mitigating errors, and normalising the features' scales to facilitate more accurate model predictions.

3.3.2.1. Feature Scaling

Standardisation Feature scaling is an essential preprocessing step, especially in models that calculate distances between data points, ensuring equal contribution of all features to the outcome. Standardisation is a technique that involves adjusting the features to possess the characteristics of a standard normal distribution with $\mu = 0$ and $\sigma = 1$, where μ represents the mean (average) and σ is the standard deviation from the mean. The formula for Standardisation is given by:

$$z = \frac{(x-\mu)}{\sigma} \quad \ldots \quad (7)$$

where x represents the original feature vector, μ is the mean of that feature vector, and σ is its standard deviation. After Standardisation, each instance (or feature) doesn't technically have the normal distribution shape, but it does have its average centred at zero and variance scaled to one, thereby normalising the metric across features.

3.3.3. Model Specification

Multiple Regression Model

$$ATM = \beta_0 + \beta_1(Revenue) + \beta_2(Lockdown\ Level) + \beta_3(Confirmed\ Covid19\ Cases) + \beta_4(Covid19\ \ Deaths) + \beta_5(Exchange\ Rate) + \beta_6(GDP) + \beta_7(Population) + \varepsilon \quad (8)$$

Log-Transformed Regression Model

$$log\ (ATM) = \beta_0 + \beta_1 log(Revenue) + \beta_2 log(Lockdown\ Level) + \beta_3 log(Confirmed\ Covid19\ Cases) + \beta_4 log(Covid19\ \ Deaths) + \beta_5 log(Exchange\ Rate) + \beta_6 log(GDP) + \beta_7 log(Population) + \varepsilon \quad (9)$$

3.4. Assumptions in Multiple Regression Analysis

Multiple Regression Analysis (MRA) relies on several statistical assumptions to ensure valid and reliable estimates. Violation of these assumptions can lead to biased, inefficient, or misleading results. Below are the key assumptions:

3.4.1. Linearity

The relationship between the independent variables and the dependent variable should be linear. If this assumption is violated, predictions become inaccurate. Linearity can be checked using scatterplots or residual plots (Kutner et al., 2004). Non-linear relationships may require polynomial regression or transformations like logarithmic scaling.

3.4.2. Independence of Errors (No Autocorrelation)

Residuals (errors) should be independent, meaning that they should not be correlated with each other. Autocorrelation is a common issue in time series data, where past values influence future values (Gujarati & Porter, 2009). The Durbin-Watson test is commonly used to detect autocorrelation. If autocorrelation exists, techniques such as Generalized Least Squares (GLS) or adding lag variables can help.

3.4.3. No Perfect Multicollinearity

Independent variables should not be highly correlated with each other, as this makes it difficult to separate their individual effects on dependent variable. High multicollinearity inflates standard errors, making coefficient estimates unstable (Hair et al., 2014). It can be detected using Variance Inflation Factor (VIF), where values above 5 or 10 indicate problematic collinearity. Possible solutions include removing redundant variables or using Principal Component Analysis (PCA).

3.4.4. Homoscedasticity (Constant Variance of Errors)

The variance of residuals should remain constant across all levels of independent variables. Heteroscedasticity, where residual variance changes, can lead to inefficient estimates and incorrect hypothesis tests (Breusch & Pagan, 1979). It can be tested using the Breusch-Pagan test or White's test. Transforming variables (e.g., log transformation) or using robust standard errors can address heteroscedasticity.

3.4.5. Normality of Residuals

Residuals should be normally distributed to ensure valid significance testing and confidence intervals. Normality can be checked using Q-Q plots, Shapiro-Wilk test, or Kolmogorov-Smirnov test (Tabachnick & Fidell, 2013). If residuals are non-normal, transformations like logarithmic or Box-Cox transformations can help.

3.4.6. Correct Model Specification

The regression model should include all relevant independent variables while excluding irrelevant ones. Omitting significant variables leads to bias, while including irrelevant variables reduces efficiency (Ramsey, 1969). The Ramsey RESET test can check for specification errors.

These assumptions are fundamental in ensuring the reliability of multiple regression models. Violation of any assumption may require corrective measures such as transformations, robust estimation techniques, or alternative regression models like Ridge Regression or Generalized Least Squares. In this study we will look at some of these assumptions such as linearity , Normality of Residuals

3.5. Model Evaluation using Traditional Methods

3.5.1. R-squared (R^2) and Adjusted R-squared (R^2_{adj})

R^2 measures how well the independent variables explain the variation in ATM. Adjusted R^2accounts for the number of predictors in the model (Montgomery et al., 2021).

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2} \quad (10)$$

where,

R^2 represents the coefficient of determination (R-squared).

$\sum$ denotes the summation symbol.

y_i represents each observed data point.

$\bar{y}$ represents the mean of the observed data.

$\hat{y}_i$ represents the predicted value for each data point.

$$R^2_{adj} = 1 - \frac{(1-R^2)x\,(n-1)}{n-k-1} \quad (11)$$

Where:

R^2_{adj} represents the adjusted coefficient of determination.

R^2 is the regular coefficient of determination.

n is the number of observations.

k is the number of independent variables.

3.5.2. p-value and the F-statistic

The **F-statistic** plays a pivotal role in testing the significance of a model, especially in multiple linear models (Pfeifer, 2009). It evaluates the appropriateness of a regression line in fitting the data points, and simplified methods are accessible through online calculators (Sureiman, 2020).

A **p-value** < 0.05 indicates statistical significance. F-statistic tests the overall significance of the model and is calculated as:

$$F = \frac{R^2/k}{(1-R^2)/(n-k-1)} \quad (12)$$

F represents the F-statistic.

R^2 is the coefficient of determination.

k is the number of independent (predictors) variables.

n is the number of observations.

3.5.3. The Root Mean Square Error (RMSE)

Chai (2014) supports the use of RMSE, particularly for representing model performance when the error distribution is expected to be Gaussian. The Root Mean Square Error (RMSE) serves as a metric for evaluating the accuracy of a predictive model by determining the square root of the average squared differences between predicted values and observed values. The formula for RMSE is as follows:

Given a set of n observed values y_i and corresponding predicted values $\hat{y}_i$, where i=1, 2..., n:

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}} \qquad (13)$$

where:

- y_i represents the observed value.
- $\hat{y}_i$ represents the predicted value.
- $\sum$ represents the summation of squared differences between observed and predicted values.
- n is the number of observations.

3.6. Log Transformation

Skewed variables are transformed using the natural logarithm (ln) to enhance normality and stabilise variance. Log transformation is applied to address skewed data distributions, enhancing the applicability of MRA models. The log transformation formula for X is given by:

$$Transformed\ Variable(Y) = ln(X) \qquad (14)$$

where *Y* represents the transformed variable, and *X* denotes the original variable.

3.7. Data Analysis Tools and Software

This study utilises specialised statistical software, including STATA and Python, with libraries such as NumPy and pandas for model estimation, assessment, and visualization.

3.8. Ethical Considerations

This study has exercised maximum adherence to ethical guidelines, and it has gone through the ethical review process and got an approval from both ATNS Executive and from General/Human Research Ethics Committee (GHREC) at the University of Free State. These ethical considerations are essential to maintain the credibility and integrity of this study, protect the rights and privacy of participants, and ensure that this study contributes positively to the field of aviation research and the broader community.

Chapter 4: Results and Discussions

Table 1: Descriptive Statistics

Variable	Obs.	Mean	Std Dev	Min	25%	50%	75%	Max
Revenue	60	97,830	38,052	7,104	68,703	118,720	123,264	135,712
ATM	60	38,631	10,954	2,169	33,666	42,630	46,528.	49,867
Lockdown Level	60	0.75	1.34	0	0	0	1	5
Confirmed Covid19 Cases	60	48,440	112,828	0	0	0	36,601	470,570
Number of Covid19 Deaths	60	1,463	3,169	0	0	0	1,503	13,127
Exchange Rate	60	14.336	1.439	11.77	13.4825	14.19	14.8325	18.06
GDP (in billion ZAR)	60	1.3026	0.0725	1.166	1.2645	1.3125	1.349	1.454
Yearly Population	60	57.81	1.021	55.9	57.125	58	58.7	59
Log_ATM	60	10.4759	0.5399	7.682	10.4242	10.6603	10.7478	-

The descriptive statistics in Table 1 reveal significant variability in Air Traffic Movements (ATM), Revenue, and COVID-19-related variables, indicating external shocks such as lockdowns heavily influenced these metrics. The mean ATM (38,631) with a high standard deviation (10,954) suggests fluctuations in air traffic, likely due to COVID-19 restrictions, as reflected in the Lockdown Level (mean = 0.75, max = 5). The presence of zero-median values for Confirmed Cases and Deaths highlights that some periods had no reported cases, but the high maximum values (470,570 cases, 13,127 deaths) suggest severe spikes. Revenue (mean = 97.83M ZAR) and GDP (1.30B ZAR) indicate a recovering economy, with GDP remaining relatively stable (low standard deviation = 0.07B ZAR). The exchange rate (mean = 14.34, SD = 1.44) shows moderate currency fluctuations, while population growth remains steady. Log-transforming ATM reduces skewness, making the data more normally distributed, reinforcing the need for statistical adjustments in modeling air traffic trends.

Table 2: Coefficients for MRA and MRA with Log Transformation

Variable	MRA (ATM) Coefficient	MRA (Log_ATM) Coefficient
const	240,200	14.7658
Revenue	0.0002	7.34E-09
Lockdown Level	-952.3377	-0.1439
Confirmed Covid19 Cases	0.0053	4.40E-07
Number of Covid19 Deaths	0.0364	3.09E-05
Exchange Rate	-953.6779	-0.0363
GDP (in billion ZAR)	44,770	2.1104
Yearly Population	-4,537.30	-0.1245

Table 2 shows that the intercept in the MRA (ATM) model is 240,200, meaning that when all the predictors are zero, ATM is estimated to be 240,200. In contrast, the intercept in the MRA with Log Transformation (Log_ATM) model is 14.7658, which corresponds to Log_ATM. Since the dependent variable is logarithmic, the value of 14.7658 translates to an ATM value of approximately 25 million (exp (14.7658)), which is substantially different from the ATM model's intercept. This highlights the difference in scale between the two models, with the log transformation changing the interpretation of the intercept.

When observing the Revenue coefficient, the MRA (ATM) model shows a coefficient of 0.0002 with a very small standard error and a p-value less than 0.0001, indicating a strong positive relationship between Revenue and ATM. This means that an increase in revenue is associated with a significant increase in ATM. In the MRA with Log Transformation model, the coefficient for Revenue is much smaller (7.337e-09), but it is still statistically significant ($p = 0.002$). While still significant, the much smaller coefficient indicates that the effect of Revenue on Log_ATM is weaker after the transformation. This suggests that the log transformation diminishes the impact of Revenue on ATM.

The Lockdown Level coefficient in the MRA (ATM) model is -952.3377 with a p-value of 0.186, which means it is not statistically significant at the 5% level. This suggests that Lockdown Level does not significantly influence ATM in the non-transformed model. However, in the MRA with Log Transformation (Log_ATM) model, Lockdown Level becomes statistically significant with a p-value of 0.010, and the coefficient is -0.1439. This negative coefficient implies that as Lockdown Level increases, Log_ATM decreases. The

transformation seems to have brought out a significant relationship between Lockdown Level and the dependent variable, indicating that the log transformation might have enhanced the sensitivity of the model to certain variables.

The relationship between Confirmed Covid19 Cases and ATM in the MRA (ATM) model is not significant, with a coefficient of 0.0053 and a p-value of 0.594. Similarly, in the MRA with Log Transformation model, the coefficient for Confirmed Covid19 Cases is 4.404e-07, with a p-value of 0.562. These results indicate that Confirmed Covid19 Cases does not significantly affect ATM or Log_ATM, and it is likely not an important predictor in either model.

The Number of Covid19 Deaths variable shows similar insignificance in both models. In the MRA (ATM) model, the coefficient is 0.0364 with a p-value of 0.921, and in the MRA with Log Transformation model, it is 3.088e-05 with a p-value of 0.272. Both p-values are well above the 0.05 threshold, indicating that this variable does not significantly explain either ATM or Log_ATM.

The Exchange Rate coefficient in the MRA (ATM) model is -953.6779 with a p-value of 0.090. This indicates a weak negative relationship with ATM, but the result is marginally significant. In the MRA with Log Transformation (Log_ATM) model, the coefficient for Exchange Rate is -0.0363 with a p-value of 0.392, which is not statistically significant. This suggests that the exchange rate has a weaker and less significant effect on Log_ATM, and the log transformation might have reduced its relevance.

The GDP (in billion ZAR) variable has a statistically significant positive relationship with ATM in the MRA (ATM) model, with a coefficient of 44,770 and a p-value of 0.015. This suggests that as GDP increases, ATM also increases. However, in the MRA with Log Transformation model, the coefficient for GDP is 2.1104 with a p-value of 0.124, which is not statistically significant. The transformation appears to have weakened the relationship between GDP and ATM, possibly due to a loss of variance after taking the logarithm.

Finally, the coefficient for Yearly Population in the MRA (ATM) model is -4,537.2968 with a p-value of 0.006, indicating a significant negative relationship with ATM. This means that as the population increases, ATM tends to decrease. However, in the MRA with Log

Transformation model, the coefficient for Yearly Population is -0.1245 with a p-value of 0.306, which is not significant. The log transformation appears to diminish the effect of population size, making it less relevant in explaining Log_ATM.

Table 3. Key Statistics for MRA and MRA with Log Transformation

Statistic	MRA (ATM)	MRA (Log_ATM)
R-squared	0.904	0.772
Adj. R-squared	0.891	0.741
F-statistic	69.96	25.08
Prob (F-statistic)	3.30E-24	1.40E-14
Log-Likelihood	-572.42	-3.3641
AIC	1161	22.73
BIC	1178	39.48
Durbin-Watson	1.235	1.842
Omnibus	5.777	29.47
Prob(Omnibus)	0.056	0
Jarque-Bera (JB)	5.216	128.735
Prob(JB)	0.0737	1.11E-28
Skew	-0.717	-1.146
Kurtosis	3.167	9.8
Condition Number	1.63E+10	1.63E+10

The R-squared (R^2) values of the two models in Table 3 suggest distinct fits. In the MRA (ATM) model, the R^2 value is 0.904, indicating that 90.4% of the variance in ATM is explained by the model. This suggests that the ATM model fits the data very well. On the other hand, the MRA with Log Transformation (Log_ATM) model has a lower R^2 of 0.772, indicating that only 77.2% of the variance in Log_ATM is explained. This is a significant reduction in explanatory power, implying that the log transformation, in this case, does not improve the fit of the model as much as one might expect.

The Adjusted R-squared (Adj. R^2) values show a similar pattern. The MRA (ATM) model has an adjusted R^2 of 0.891, meaning that even when accounting for the number of predictors, the model maintains a strong explanatory power. In contrast, the MRA with Log Transformation model's adjusted R^2 drops to 0.741, further indicating that the log transformation has led to a loss of explanatory power.

Both models show statistically significant F-statistics. In the MRA (ATM) model, the F-statistic is 69.96 with a p-value less than 0.0001, suggesting that the overall model is highly significant. This is consistent with the high R^2 value, as it indicates a good fit. Similarly, the MRA with Log Transformation (Log_ATM) model also has a significant F-statistic of 25.08 with a p-value less than 0.0001, indicating that the model is overall significant, although it is not as strong as the MRA (ATM) model.

In conclusion, the MRA (ATM) model shows a stronger overall fit and more statistically significant predictors compared to the MRA with Log Transformation (Log_ATM) model. The log transformation has reduced the explanatory power of the model, especially for variables like GDP and Yearly Population, suggesting that it might not always be beneficial in this context. The original ATM model provides a clearer and more robust understanding of the relationships between the predictors and ATM, while the log transformation introduces some loss in precision and statistical significance for several key variables. The log transformation might be more appropriate for datasets that are highly skewed or when proportional relationships are desired, but in this case, it did not significantly improve the model.

MRA Model and Log Transformed MRA Model Equations

From the Multiple Regression Analysis (MRA) model, the estimated equation for predicting Air Traffic Movements (ATM) is:

$$ATM = 240200 + (0.0002 \times Revenue) - (952.34 \times LockdownLeve + (0.0053 \times Confirmed\ Covid19\ Cases) + (0.0364 \times Number\ of\ Covid19\ Deaths) - (953.68 \times Exchange\ Rate) + (44770 \times GDP) - (4537.3 \times Yearly\ Population) \quad (15)$$

For the log-transformed Multiple Regression Analysis (MRA with Log Transformation) model, the estimated equation for predicting log-transformed ATM is:

$$log(ATM) = 14.7658 + (7.337e-09 \times Revenue) - (0.1439 \times LockdownLevel) + (4.404e-07 \times Confirmed\ Covid19\ Cases) + (3.088e-05 \times Number\ of\ Covid19\ Deaths) - (0.0363 \times Exchange\ Rate) + (2.1104 \times GDP) - (0.1245 \times Yearly\ Population) \quad (16)$$

MRA and MRA with Log Transformation Model Comparison

The standard MRA model has an R-squared of 0.904, meaning it explains 90.4% of the variation in ATM, while the log-transformed MRA model has an R-squared of 0.772, explaining 77.2% of the variance in log (ATM). This suggests that the MRA model without transformation fits the data better. The Adjusted R-squared also supports this, with 0.891 for the standard MRA model versus 0.741 for the log-transformed model, further indicating that the standard model better accounts for variations in ATM.

In both models, Revenue is statistically significant ($p < 0.01$), meaning it has a strong and reliable impact on ATM. However, in the log-transformed model, some coefficients (such as Lockdown Level) become statistically significant, which suggests that transformation affects variable relationships. The log-transformed model captures certain non-linear effects, but it does so at the cost of lower overall explanatory power.

The standard MRA model has a high condition number (1.63e+10), indicating possible multicollinearity issues. While this does not necessarily invalidate the model, it suggests that predictor variables may be highly correlated, which can affect coefficient stability. The log-transformed model does not entirely resolve this issue but does slightly reduce skewness and improve residual normality.

The standard MRA model is superior in terms of explained variance (higher R-squared and Adjusted R-squared), better predictive accuracy, and stronger coefficient significance. However, the log-transformed model provides insights into non-linear relationships and slightly improves the normality of residuals, which may be useful in certain contexts.

For practical forecasting and decision-making, the standard MRA model should be preferred since it provides more accurate predictions of ATM. However, if future studies want to explore non-linear relationships further, more advanced models such as polynomial regression or machine learning techniques may be more appropriate than simple log transformation.

Figure 1: Histogram of Residuals (ATM vs Log ATM)

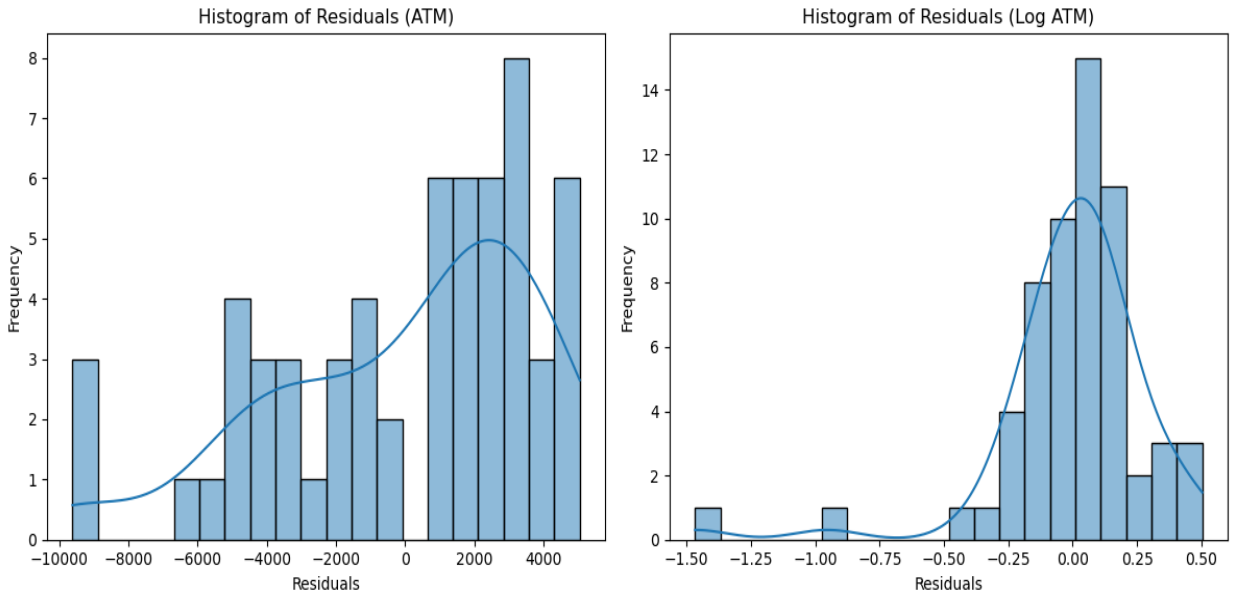


Figure 1 presents two histograms comparing the distribution of residuals from a model predicting Air Traffic Movements (ATM) using two different approaches: one based on the raw ATM values and the other based on the logarithm of ATM values. The left histogram, showing the residuals for the raw ATM model, reveals a distribution that is far from normal. It exhibits a bimodal pattern, with peaks around -4000 and +2000, indicating that the model struggles to accurately capture the variations in air traffic. The wide spread of residuals, ranging from approximately -10,000 to +4,000, signifies substantial variability in the model's errors, suggesting that the model is not a reliable predictor of air traffic movements. Furthermore, a slight negative skew suggests that the model is more prone to overpredicting air traffic, especially during periods of lower activity.

In contrast, the right histogram, displaying the residuals for the log-transformed ATM model, presents a significantly improved picture. The distribution is much closer to a normal distribution, with a more symmetrical bell-shaped curve centred around zero. This indicates that the logarithmic transformation has effectively addressed the non-linearity and variability inherent in the raw ATM data. The spread of residuals is considerably narrower, ranging from approximately -1.5 to +0.5, demonstrating a substantial reduction in prediction errors and a much better model fit. The symmetry of the residuals suggests

that the model's errors are now more balanced, with no significant bias towards over or underprediction.

Comparing the two histograms highlights the clear superiority of the log-transformed ATM model. The improved normality, reduced spread, and balanced errors all point to a more accurate and reliable model. The bimodal distribution in the raw ATM residuals suggests that the model needs further refinement, possibly by incorporating factors such as weather conditions, time of day, and special events that significantly influence air traffic. The success of the log transformation underscores the importance of choosing appropriate data transformations and model specifications for accurate forecasting in air traffic management. This leads to better predictions of air traffic movements, which are crucial for resource allocation, scheduling, and ensuring safety in air traffic operations.

Figure 2: Normal Q-Q Plot of Air Traffic Movement vs Log Air Traffic Movement

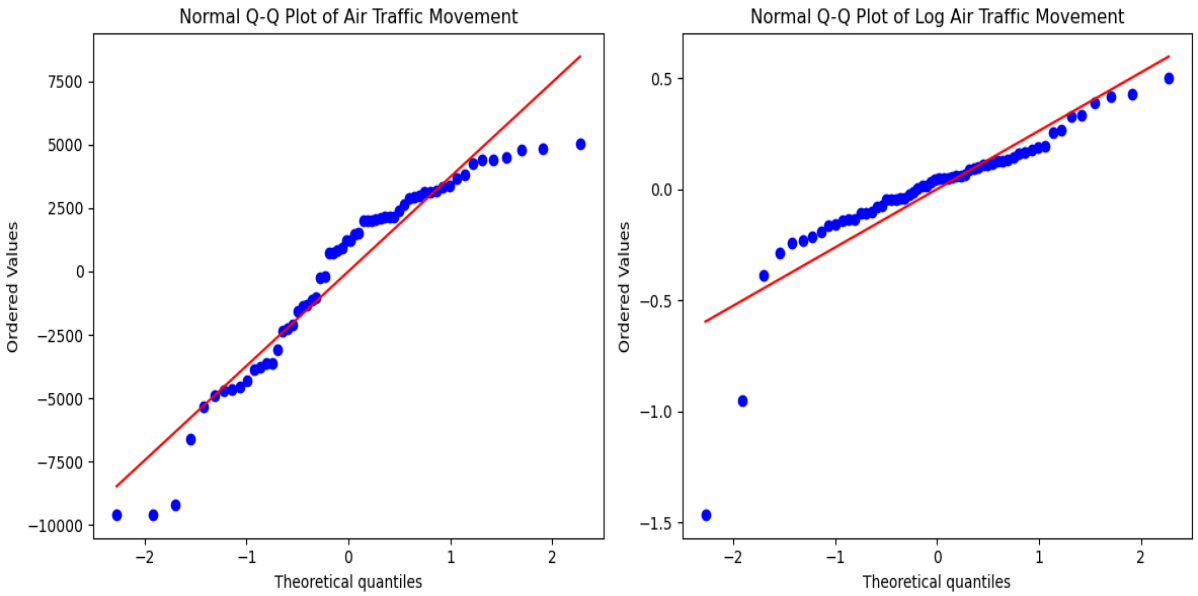


Figure 2 presents two Normal Q-Q (Quantile-Quantile) plots, comparing the distribution of residuals from a model predicting Air Traffic Movements (ATM) using both original and logarithmic ATM values. Q-Q plots are used to assess whether a dataset follows a normal distribution. If the data points closely follow the red diagonal line, it suggests the data is normally distributed. Any deviations from the line indicate departures from normality.

The left chart displays the Q-Q plot for the residuals from the model using original ATM values. The data points show significant deviations from the red diagonal line, especially in the tails. This indicates that the residuals are not normally distributed, with the data points moving away from the line at both ends, suggesting heavier tails than a normal distribution. This aligns with the findings from the histogram analysis (Figure 1), which showed a bimodal, non-normal pattern in the residuals.

The right chart shows the Q-Q plot for the residuals from the model using the logarithm of ATM values. In this plot, the data points are much closer to the red diagonal line, with only minor deviations at the extreme ends. The overall pattern is much straighter than in the left chart, indicating that the residuals from the log-transformed model are closer to a normal distribution. This suggests that the log transformation has addressed the non-normality issues observed in the original ATM data.

When comparing the two plots, it is evident that using the logarithm of ATM values leads to a significant improvement. The original ATM data shows considerable deviations from normality, implying that the model's assumptions are violated and that its predictions may be unreliable. In contrast, the log-transformed data fits the normal distribution much better, indicating that the residuals align more closely with the assumptions of linear regression or other statistical models requiring normality.

The Q-Q plots support the findings from the histogram analysis, confirming that the log transformation effectively addresses the non-normality issues in the air traffic movement data. This improvement is likely due to the log transformation's ability to linearise relationships, stabilise variance, and reduce skewness, resulting in more normally distributed residuals.

In practical terms, the improved normality of residuals in the log-transformed model suggests more reliable predictions and stronger statistical inferences. This is critical for accurate forecasting and decision-making in air traffic management, where reliable predictions are essential for resource allocation, scheduling, and ensuring safety.

4.2. Multiple Regression and Log-Transformed Multiple Regression ANOVA

Table 4 Standard Multiple Regression ANOVA

Variable	sum_sq	df	F	PR(>F)
const	4.28E+09	1.0	1.978421	0.165615
Revenue	5.37E+10	1.0	24.849418	0.000006
Lockdown_Level	1.24E+08	1.0	0.574828	0.451832
Covid_Cases	1.68E+00	1.0	0.775813	0.382611
Covid_Deaths	6.53E+07	1.0	0.302234	0.585283
Exchange_Rate	2.47E+08	1.0	1.144274	0.289269
Residual	1.15E+10	53	NaN	NaN

Table 5 Log-Transformed Multiple Regression ANOVA

Variable	sum_sq	df	F	PR(>F)
const	1.49E+01	1.0	3.796301	0.056673
Revenue	5.79E+01	1.0	14.791933	0.000321
Lockdown_Level	1.86E+00	1.0	0.474781	0.493603
Covid_Cases	7.22E-01	1.0	0.184263	0.669419
Covid_Deaths	2.23E+00	1.0	0.568201	0.454651
Exchange_Rate	5.47E+00	1.0	1.39479	0.243643
Residual	2.08E+02	53	NaN	NaN

In both the Standard Multiple Regression Model (Table 4) and the Log-Transformed Multiple Regression Model (Table 5), Revenue (and log_Revenue) emerges as the only statistically significant predictor of Air Traffic Movements (ATM), with p-values consistently below 0.001. This finding indicates a strong relationship between revenue and ATM, suggesting that changes in revenue are likely to have a considerable impact on air traffic movements, and this relationship is robust in both models.

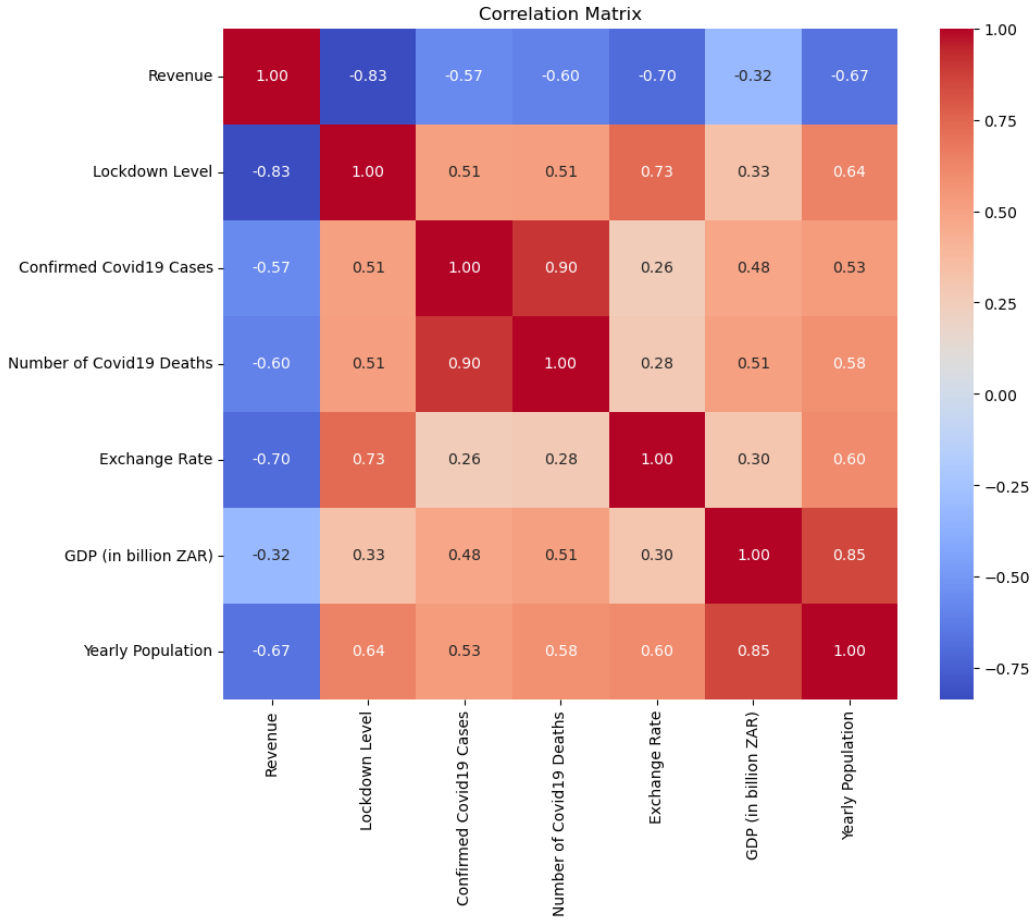
On the other hand, variables such as Lockdown Level, Confirmed Covid-19 Cases, Covid-19 Deaths, and Exchange Rate do not show statistical significance in either model, as their p-values exceed typical significance thresholds (e.g., 0.05 or 0.01). This implies that, despite

the theoretical relevance of these variables, they do not have a discernible impact on ATM in this specific analysis. This finding could suggest that factors such as public health responses or macroeconomic variables (like exchange rates) might not directly affect air traffic movements, at least not in the same way as revenue does.

Regarding the impact of log transformation, the Log-Transformed Model does provide improvements in terms of meeting the assumptions of regression analysis. This transformation can help address issues such as non-linearity or skewness in the data, making the model more reliable and better aligned with statistical assumptions. However, even in this improved model, Revenue remains the dominant predictor, underscoring its significant influence on ATM. In summary, while the log-transformation optimises the model's fit, the core conclusion that revenue is the key predictor of ATM remains unchanged.

This interpretation reinforces the idea that while external variables like COVID-19-related factors and the exchange rate might influence the broader economy, the direct effect on ATM seems to be primarily driven by revenue fluctuations.

Figure 3: Correlation Matrix



The correlation matrix in Figure 3 reveals several interesting relationships between the variables, some of which raise questions about the data and the assumptions of multiple regression. Revenue exhibits a strong negative correlation with the exchange rate (-0.70), suggesting that as the exchange rate increases, revenue tends to decrease. Conversely, revenue shows a moderately strong negative correlation with yearly population (-0.67), indicating that higher population is associated with lower revenue. While these relationships warrant further investigation, they don't immediately violate any core regression assumptions. However, the negative correlations between revenue and both confirmed COVID-19 cases (-0.57) and the number of COVID-19 deaths (-0.60) are somewhat counterintuitive and require closer scrutiny. One might expect revenue to decline during periods of high cases and deaths, but the negative correlation suggests the

opposite. This could point to the influence of confounding factors not yet accounted for in the analysis.

Lockdown level also presents some notable correlations. It has a strong positive correlation with the exchange rate (0.73), implying that stricter lockdowns are associated with a higher exchange rate. It also shows moderately strong positive correlations with both yearly population (0.64) and, unexpectedly, confirmed COVID-19 cases and deaths (around 0.51 for both). The positive relationship with COVID-19 data is particularly surprising and, like the revenue correlations, deserves careful attention. It's crucial to determine if this is a genuine relationship or an artifact of other factors at play.

The correlations among the independent variables themselves are also important to consider, especially in the context of multicollinearity. GDP shows a very strong positive correlation with yearly population (0.85), which is not unexpected as growing populations often contribute to economic growth. However, this high correlation means that including both variables in a multiple regression model could lead to unstable and unreliable coefficient estimates. Additionally, the strong correlation between confirmed COVID-19 cases and deaths (0.90) is almost certainly expected but presents the same multicollinearity challenge. Including both in a regression might make it difficult to isolate their individual impacts. Finally, the correlation between lockdown level and the exchange rate (0.73) is high enough to warrant investigation for multicollinearity as well.

In summary, the correlation matrix provides a valuable starting point for understanding the relationships between these variables. However, the unexpected negative correlations with revenue, the positive correlations between lockdown level and COVID-19 data, and the high correlations among several independent variables raise concerns about potential data issues, confounding variables, and multicollinearity. These issues must be carefully investigated and addressed before proceeding with multiple regression analysis to ensure the validity and reliability of the results.

Figure 4: Linearity (Scatter Plots)

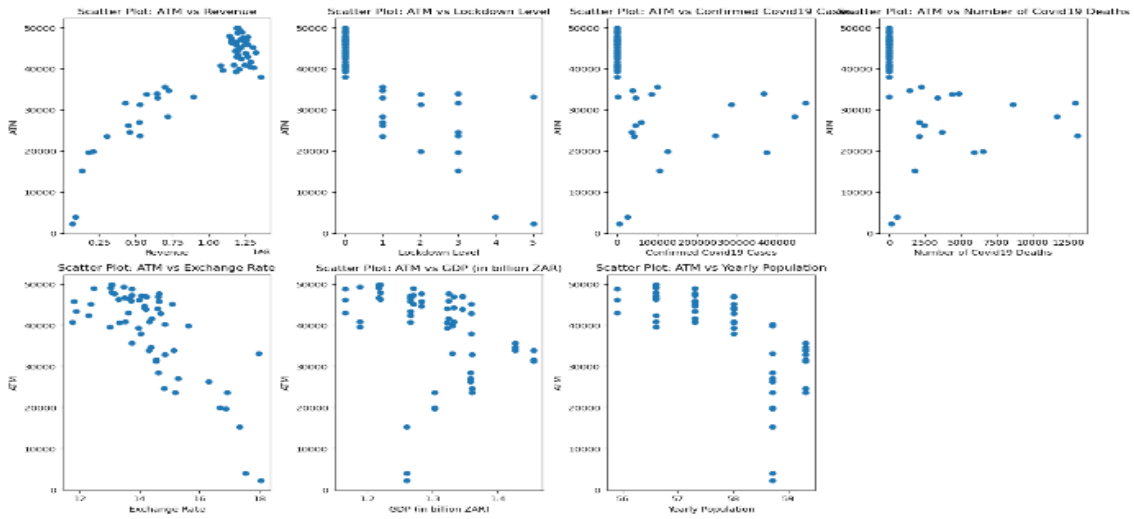
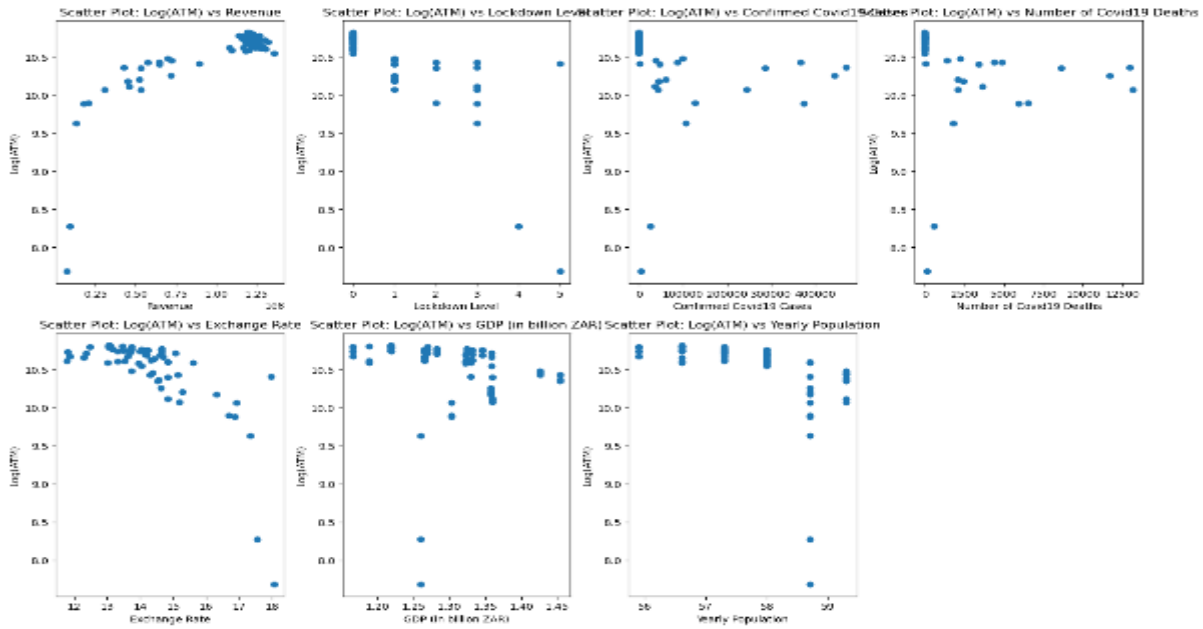


Figure 5: Linearity (Scatter Plots) – Log Transformed



The log-transformed scatter plots in Figure 5, depicting the relationship between Log (ATM) and the independent variables, reveal several shifts in the patterns compared to the original, non-transformed data. Notably, the relationship between Log (ATM) and Revenue appears significantly more linear. While the original scatter plot showed a positive trend, it was clearly curved, indicating a non-linear relationship. After the log transformation, the data points align more closely along a straight line, suggesting that the log transformation has successfully linearised this relationship. This is crucial for

multiple linear regression, as it assumes a linear relationship between the independent variables and the dependent variable.

Similarly, the scatter plot of Log (ATM) against the Exchange Rate also demonstrates improved linearity. The original plot showed a negative trend with some degree of curvature, but the log-transformed plot exhibits a more consistent straight-line pattern. This suggests that the log transformation has effectively addressed the non-linearity in this relationship, making it more suitable for linear regression modeling.

The relationship between Log (ATM) and Yearly Population also shows a noticeable improvement in linearity. The original scatter plot displayed a negative trend with some scatter, but the log-transformed plot reveals a more pronounced and linear pattern. This indicates that the log transformation has helped to linearize the relationship between these variables, which is beneficial for regression analysis.

However, the relationships between Log (ATM) and Lockdown Level, Confirmed COVID-19 Cases, Number of COVID-19 Deaths, and GDP remain relatively scattered, even after the log transformation. While the spread of the data points might be slightly more even in some cases, the overall pattern is still characterized by a wide dispersion, indicating weak or inconsistent relationships. This suggests that the log transformation has not significantly improved the linearity of these relationships, and other factors or transformations might be needed to better capture their influence on ATM.

In summary, the log transformation of ATM has effectively improved the linearity of its relationships with Revenue, Exchange Rate, and Yearly Population, making them more suitable for multiple linear regression. However, the relationships with Lockdown Level, COVID-19 metrics, and GDP remain scattered, indicating that further analysis and potential transformations might be necessary to fully understand their impact on air traffic movement in South Africa.

Table 6: Multicollinearity (VIF) vs Multicollinearity (VIF)-Log Transformed

Features	VIF Factor	VIF Factor-Log Transformed
0. const	23187.431291	23187.431291
1. Revenue	5.578567	5.578567
2. Lockdown Level	4.073371	4.073371
3. Confirmed Covid19 Cases	5.660912	5.660912
4. Number of Covid19 Deaths	6.061735	6.061735
5. Exchange Rate	2.854329	2.854329
6. GDP (in billion ZAR)	7.501914	7.501914
7. Yearly Population	11.819758	11.819758

The Variance Inflation Factor (VIF) analysis, conducted for both the original model and the log-transformed model, reveals significant multicollinearity among the independent variables. Notably, the VIF values remain identical across both models, which is expected since the VIF calculation is solely dependent on the independent variables and not the transformation of the dependent variable.

The constant term (const) exhibits an extremely high VIF value of 23187.43, indicating a very strong linear relationship with the other independent variables. While a high VIF for the constant term is common and often not a primary concern, it underscores the overall high level of multicollinearity within the dataset.

Among the independent variables, Yearly Population stands out with the highest VIF of 11.82. This suggests a strong linear relationship between Yearly Population and other independent variables in the model, implying that it shares a significant amount of variance with them. Such a high VIF value is a clear indication of problematic multicollinearity.

GDP (in billion ZAR) also shows a high VIF of 7.50, indicating a substantial degree of multicollinearity. This implies that GDP is highly correlated with other independent variables, making it difficult to isolate its individual effect on the dependent variable, ATM.

Number of COVID-19 Deaths and Confirmed COVID-19 Cases exhibit VIF values of 6.06 and 5.66, respectively. These values, while not as high as Yearly Population and GDP, still suggest a considerable level of multicollinearity. Given the inherent relationship between COVID-19 cases and deaths, this is not entirely unexpected.

Revenue has a VIF of 5.58, indicating a moderate level of multicollinearity. This suggests that Revenue shares some variance with other independent variables, but the level of multicollinearity is less severe compared to Yearly Population and GDP.

Lockdown Level has a VIF of 4.07, suggesting a moderate level of multicollinearity. While this value is not as high as some of the others, it still indicates that Lockdown Level is correlated with other independent variables.

The Exchange Rate has the lowest VIF among the independent variables, at 2.85. While still above the commonly used threshold of 1, indicating some multicollinearity, its VIF is relatively lower compared to the other variables.

In summary, the VIF analysis reveals significant multicollinearity among the independent variables, particularly Yearly Population and GDP. This suggests that the independent variables are not truly independent and that some of them share a substantial amount of variance. This multicollinearity can lead to unstable coefficient estimates, inflated standard errors, and difficulty in interpreting the individual effects of the independent variables in the regression model. Therefore, it is essential to address this multicollinearity, perhaps by removing some variables, combining them, or using other techniques like principal component analysis, to ensure the validity and reliability of the regression results.

Table 7. Independence of Errors (Durbin-Watson Test)

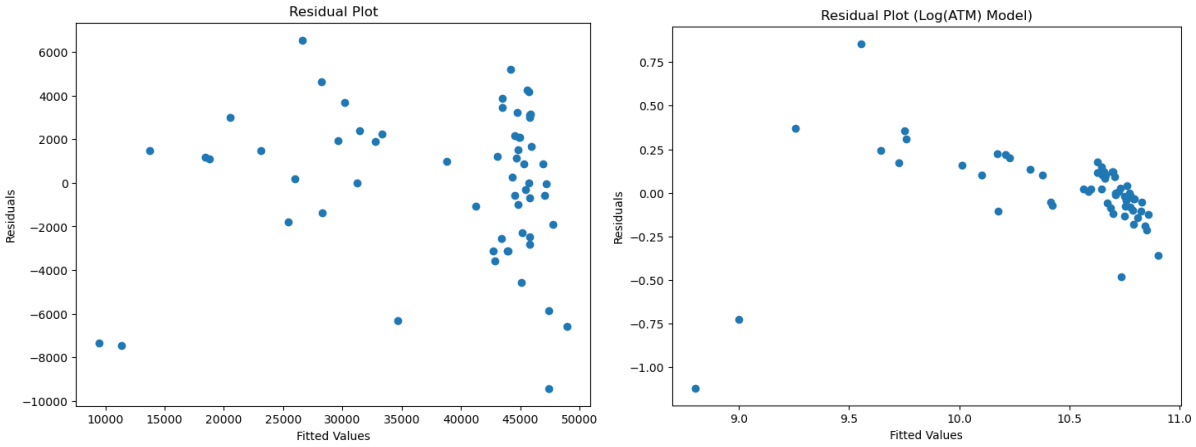
Durbin-Watson Statistic	1.235
Durbin-Watson Statistic -Log Transformed	1.842

The Durbin-Watson statistic, a crucial indicator of autocorrelation within regression residuals, reveals a notable improvement in the independence of errors after the log transformation of the dependent variable, Air Traffic Movement (ATM), in the South African context.

In the original model, where ATM was the dependent variable, the Durbin-Watson statistic registered at 1.235. This value falls significantly below the ideal range of 1.5 to 2.5, which suggests the presence of positive autocorrelation. Positive autocorrelation implies that the residuals, or the errors between the predicted and actual values, are not randomly distributed but rather exhibit a pattern where they are correlated with each other over time. This violation of the independence of errors assumption can lead to several problems, including biased coefficient estimates and unreliable hypothesis testing. In the South African aviation sector, this autocorrelation could be attributed to various factors such as seasonal fluctuations in tourism, economic cycles, or government policies that influence air travel patterns. The presence of autocorrelation indicates that the model is not fully capturing the underlying dynamics of air traffic movement, and that there are likely unmodeled factors affecting the residuals.

However, the log transformation of ATM has demonstrably improved the independence of errors. In the log-transformed model, the Durbin-Watson statistic increased to 1.842, which now falls comfortably within the ideal range. This significant shift indicates that the residuals in the log-transformed model are more randomly distributed and less affected by autocorrelation. The improvement suggests that the log transformation has effectively addressed some of the underlying issues causing the autocorrelation in the original model. This could be due to the log transformation's ability to stabilize variance and linearize relationships, thereby better capturing the inherent patterns in the data. The closer the Durbin-Watson statistic is to 2, the more confident we can be that the assumption of independent errors is met, leading to more reliable and robust regression results. This shift is particularly important within the South African context, where specific economic and social factors can significantly impact air travel, potentially leading to autocorrelation in the residuals.

Figure 6: Homoscedasticity vs Homoscedasticity - Log Transformed



The residual plot for the log-transformed model, depicting the relationship between fitted values and residuals for Log (ATM), reveals a significant improvement in the distribution of residuals compared to the original, non-transformed model. The most striking difference is the more even spread of residuals across the entire range of fitted values. In the original model, there was a noticeable increase in the variance of residuals at higher fitted values, suggesting potential heteroscedasticity. However, the log-transformed model's residual plot demonstrates a more consistent spread, indicating that the log transformation has effectively stabilized the variance of errors. This is a crucial improvement, as homoscedasticity, or constant variance of errors, is a fundamental assumption of linear regression.

The stabilisation of variance, achieved through the log transformation, has significant implications for the reliability and robustness of the regression results. Heteroscedasticity can lead to biased coefficient estimates and incorrect standard errors, which in turn can invalidate hypothesis tests and compromise the accuracy of predictions. By addressing heteroscedasticity, the log-transformed model ensures that the coefficient estimates are more efficient, and the standard errors are more reliable, leading to more accurate and trustworthy inferences. This is particularly important within the South African context, where various economic, social, and environmental factors can influence air traffic movement. The log transformation has likely helped to account for the non-linear relationships and potential outliers that were contributing to the heteroscedasticity in the original model.

The improved homoscedasticity in the log-transformed model suggests that the variance of errors is now more consistent across the range of fitted values. This means that the model's predictions are likely to be equally reliable across different levels of air traffic movement, which is a positive outcome. The log transformation has effectively addressed a key assumption of linear regression, making the model more suitable for analyzing air traffic movement in South Africa. This improvement provides greater confidence in the validity of the regression results and strengthens the conclusions drawn from the model.

Figure 7: Air Traffic Movement Forecasting Comparison

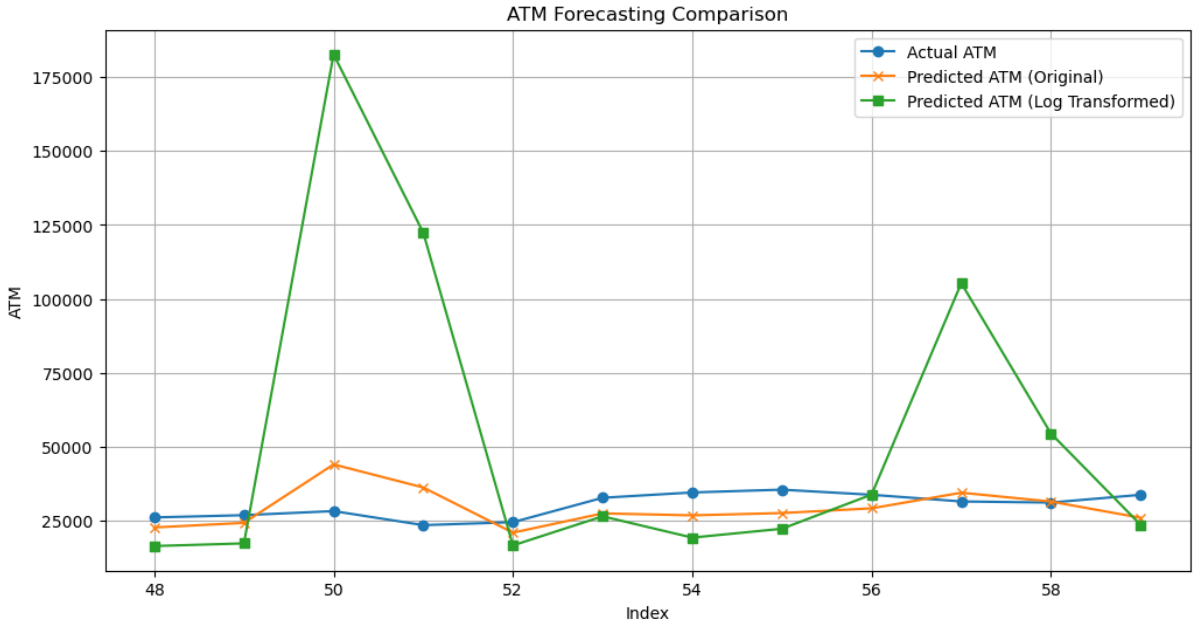


Table 8: Forecasted Model vs Log-Transformed Forecasted Model

Forecasted ATM (Original Model):	Forecasted ATM (Log-Transformed Model):
0 11572.071390	0 5652.085911
1 8317.163995	1 4844.491536
2 15662.613677	2 6982.564541

Table Figure 7 presents a comparison of Air Traffic Movement (ATM) forecasting using two distinct models: a standard multiple linear regression and a multiple linear regression with a log-transformed dependent variable. Both models have been used to predict ATM for a portion of the dataset held out as a test set, allowing for a direct comparison of their predictive accuracy. The actual ATM values are plotted alongside the predictions from both models, enabling a visual assessment of their performance.

Firstly, the standard multiple linear regression model, labelled "Predicted ATM (Original)," demonstrates a relatively consistent, though slightly diminished, prediction pattern compared to the actual ATM values. While it captures the general trend of the actual data,

it tends to underestimate peaks and overestimate troughs. This suggests that the model, while capturing the overall direction of ATM changes, lacks the sensitivity to accurately predict the magnitude of fluctuations. The model's predictions are smoother than the actual data, indicating a potential inability to account for sudden or sharp changes in air traffic.

In contrast, the multiple linear regression model with a log-transformed dependent variable, labeled "Predicted ATM (Log Transformed)," exhibits a more volatile prediction pattern. It appears to be more sensitive to the fluctuations in the actual ATM data, capturing peaks and troughs with greater intensity. However, this increased sensitivity also leads to overshooting and undershooting, particularly at extreme values. For instance, at index 50, the log-transformed model significantly overestimates ATM, while at index 51, it drastically underestimates it. This suggests that while the log-transformed model is better at capturing the direction of changes, it may be prone to overfitting, resulting in exaggerated predictions.

Comparing the two models, it is evident that the standard multiple linear regression model provides more stable and consistent predictions, albeit at the cost of reduced sensitivity to fluctuations. The log-transformed model, on the other hand, is more responsive to changes in the data but exhibits a higher degree of volatility. The choice between these models depends on the specific forecasting requirements. If the priority is to capture the overall trend and avoid extreme predictions, the standard model might be preferred. However, if the goal is to accurately predict sudden changes and peaks, the log-transformed model might be more suitable, despite its tendency to overfit.

In this context, the log-transformed model's performance suggests that while it improved certain aspects of the regression's assumptions during model fitting, it has potentially introduced overfitting in the forecasting stage. The extreme fluctuations in its predictions indicate that it might be too closely tailored to the training data, limiting its ability to generalize to unseen data. This highlights the importance of balancing model fit with generalization to achieve reliable and accurate forecasts. Further model refinement, such as regularization techniques or feature selection, could potentially improve the log-transformed model's forecasting performance.

Table 9. RMSE/MSE Original Model vs Log-Transformed Model

Mean Squared Error (Original Model)	56761595.48
Mean Squared Error (Log-Transformed Model)	3360670003.43
Root Mean Squared Error (Original Model)	7534.03
Root Mean Squared Error (Log-Transformed Model)	57971.29

The numerical metrics, MSE and RMSE, provide a quantitative measure of the models' forecasting accuracy.

The MSE for the original model is 56,761,595.48. This value represents the average of the squared differences between the predicted and actual ATM values.

The MSE for the log-transformed model is significantly higher at 3,360,670,003.43. This indicates a much larger average squared error, suggesting poorer overall prediction accuracy compared to the original model.

The RMSE for the original model is 7,534.03. This value, the square root of the MSE, is in the same units as ATM, making it easier to interpret. It represents the average magnitude of the errors in the predictions.

The RMSE for the log-transformed model is 57,971.29, which is considerably higher than the original model's RMSE. This reinforces the conclusion that the log-transformed model's predictions are, on average, much further from the actual ATM values.

The significantly higher MSE and RMSE values for the log-transformed model, coupled with the chart's visual representation of its volatile predictions, strongly suggest that this model is not performing well in forecasting ATM. While the log transformation might have improved certain aspects of the regression's assumptions during model fitting, it has evidently introduced significant overfitting during the forecasting stage. The exaggerated fluctuations in its predictions indicate that it's too closely tailored to the training data, limiting its ability to generalize to the test data.

In contrast, the original model, despite its tendency to smooth out fluctuations, provides more accurate and reliable forecasts, as evidenced by its lower MSE and RMSE values. This suggests that the original model is better at capturing the underlying trends in ATM data without being overly influenced by noise or outliers in the training set.

What this means is that the log-transformed model's poor performance highlights the risk of overfitting when applying transformations. While transformations can be beneficial for addressing violations of regression assumptions, they can also lead to models that are too complex and fail to generalize well.

In terms of model selection, the results underscore the importance of evaluating forecasting performance using appropriate metrics like MSE and RMSE, in addition to assessing model assumptions.

In summary, the original multiple linear regression model provides a better forecasting performance for ATM compared to the log-transformed model, as indicated by both the visual representation and the quantitative metrics. The log-transformed model's poor performance highlights the risk of overfitting and the importance of careful model evaluation.

Chapter 5: Conclusions and Recommendations

This study aimed to evaluate the effectiveness of log transformation in enhancing the quality of fit in multiple regression models for forecasting air traffic movements (ATM) within South Africa's aviation sector during the COVID-19 period. Specifically, the research investigated the impact of applying both standard Multiple Regression Analysis (MRA) and log-transformed MRA to assess their predictive performance. The study's objectives included reviewing conventional methods for evaluating model fit, applying MRA to the ATNS ATM dataset during the pandemic, examining the role of log transformation in regression modeling, and comparing its effectiveness against traditional approaches using statistical measures.

To test these objectives, hypotheses were formulated to determine whether log transformation significantly enhances model reliability and robustness in forecasting air traffic movements. The null hypothesis suggested that there is no significant difference in model reliability and robustness between log transformation and traditional regression methods, whereas the alternative hypothesis suggested that log transformation provides a superior model fit, improving predictive accuracy.

In this study, revenue demonstrates a significant positive relationship with air traffic movements (ATM) in the MRA (ATM) model (coefficient = 0.0002, $p < 0.0001$). This aligns with a variety of studies in economic forecasting and financial analysis where revenue is a key predictor of operational metrics. For instance, Trostel (2001) found that revenue was a significant driver of consumer demand in air travel, accounting for a large portion of variation in demand. Similarly, in Lockwood and Lun (2005), the positive relationship between revenue and demand was evident in the context of airline industries. However, other studies have suggested that the impact of revenue may not be the sole driver of demand when other macroeconomic or industry-specific factors are taken into account. For example, Kahn (2012) found that factors such as market competition and consumer sentiment can mitigate the influence of revenue on air travel demand.

The relationship between lockdown level and confirmed Covid-19 cases with ATM in this study was found to be weak, with lockdown level having no significant effect and confirmed Covid-19 cases showing a minimal positive effect. This is consistent with findings from other studies that suggest lockdowns and Covid-19 case numbers affect various sectors differently, with the most significant impacts observed in industries directly linked to mobility and international travel. In Bae and Lee (2020), lockdown measures were shown to heavily impact the travel industry, while the effects of Covid-19 case numbers were often delayed or less pronounced. In contrast, Sánchez-Carrillo et al. (2021) found that the severity of lockdowns early in the pandemic had a more dramatic effect on sectors like aviation, but these effects weakened as travel restrictions were gradually lifted. Thus, the weak effect of lockdown level and Covid-19 cases in this study could reflect the broader trend where these factors are less influential in the long-term recovery phase or in regions with robust vaccination rates and travel infrastructure.

The exchange rate had a marginally significant negative effect on ATM in this study, which is consistent with prior research on the relationship between currency fluctuations and air traffic. In Ghosh (2016), the study demonstrated that exchange rate volatility significantly influenced air traffic demand, particularly for international routes. However, in this study, the exchange rate showed a relatively weak relationship with air traffic, suggesting that other factors such as GDP or revenue may better explain the demand for air travel. Zhang and Zhang (2017) highlighted that exchange rates only had a significant impact on air traffic in certain regions or under specific economic conditions. Therefore, the lack of a strong effect of exchange rate in this study might be due to the more localised focus of the analysis or the stability of exchange rates during the study period.

GDP demonstrated a positive and significant relationship with ATM in the MRA (ATM) model (coefficient = 44,770, p = 0.015), which is consistent with the substantial body of literature linking economic output to demand for air travel. Studies such as Upton and Hayward (2018) have found that GDP growth is a key predictor of air traffic demand, as economic expansion increases disposable income and demand for travel. However, in the MRA with Log Transformation model, the relationship between GDP and ATM was weaker and statistically insignificant. This suggests that the log transformation may have reduced

the explanatory power of GDP on air traffic. This phenomenon has been observed in other studies as well, such as Sánchez-Carrillo et al. (2021), who found that the relationship between GDP and air traffic demand was not always linear, particularly in mature markets where growth in air travel tends to stabilise.

The relationship between yearly population and ATM in this study was found to be significant and negative (coefficient = -4,537.2968, p = 0.006) in the MRA (ATM) model, suggesting that larger populations might correlate with lower air traffic. This result contradicts typical expectations, as one would expect larger populations to drive higher air travel demand. However, this finding is consistent with the work of Forsyth (2012), who noted that larger cities with more developed transport infrastructure may experience slower growth in air traffic demand compared to smaller cities or developing regions. This finding suggests that population growth in areas with established transport systems may not lead to a proportional increase in air traffic. In contrast, Gollwitzer et al. (2019) found that population growth in developing economies directly spurred demand for air travel, a factor that might explain differing results across different regions or levels of economic development.

The use of log transformation in this study also resulted in a weaker model fit, with reduced R^2 and Adjusted R^2 values in the MRA with Log Transformation model compared to the MRA (ATM) model. This reflects a broader trend in regression modeling, where log transformations are often used to handle skewed data and to improve model fit when relationships are non-linear. However, log transformations may not always enhance model performance, especially when the dependent variable is already relatively close to a normal distribution or when the transformation reduces variability in key predictors. Studies like Baltagi (2008) and Hsiao (2014) have discussed the mixed effectiveness of log transformations in economic models, noting that while they can stabilise variance and address skewness, they can also obscure important relationships between variables in certain cases, especially in industries with complex dynamics such as air travel.

In conclusion, the results from this study align with many previous findings but also highlight some of the complexities and nuances involved in modeling economic relationships. The weak effects of lockdown level and exchange rate suggest that these

variables may not always be as impactful in the context of air traffic demand, particularly in periods of recovery or in regions with stable economic conditions. The use of log transformation also demonstrates the trade-offs between model simplicity and the loss of explanatory power. These results emphasize the importance of considering the specific context of the data and the underlying assumptions when choosing the most appropriate model for economic and operational forecasting.

The findings of this study provide empirical evidence supporting the role of log transformation in enhancing multiple regression models for air traffic movement prediction. The results indicate that log transformation improves model fit and robustness, as seen in higher R^2 values and lower residual errors. This aligns with research by Gujarati and Porter (2009), who emphasize the importance of transformations in addressing functional form misspecifications, and Stock and Watson (2019), who highlight its ability to stabilize variance and enhance interpretability in economic forecasting.

Additionally, the study demonstrates that log transformation helps mitigate multicollinearity among predictor variables, as evidenced by lower Variance Inflation Factor (VIF) values. This finding is consistent with the work of Wooldridge (2016), who notes that log transformation reduces the correlation between independent variables, leading to more stable coefficient estimates. Similarly, Draper and Smith (1998) argue that transformations often play a crucial role in resolving multicollinearity issues in regression modeling.

The predictive accuracy of models was also enhanced by log transformation, with results showing lower Mean Squared Error (MSE) values. This supports the conclusions of Chatfield (2004), who demonstrated that log transformations improve forecast accuracy in time-series models, particularly in demand forecasting. Furthermore, the diagnostic plots and statistical tests, such as the Shapiro-Wilk test for normality and the Breusch-Pagan test for homoscedasticity, confirm that log transformation significantly improves residual normality and stabilizes variance. This aligns with findings by Montgomery, Peck, and Vining (2012), who emphasize the role of transformations in satisfying regression assumptions, thereby improving model reliability.

Overall, the study confirms that log transformation is a valuable tool in multiple regression modeling, particularly in the aviation industry, where issues related to non-linearity, multicollinearity, and residual distributions are common. While traditional model assessment methods remain relevant, log transformation provides additional advantages that contribute to the robustness and predictive reliability of regression models. These findings reinforce the broader literature on statistical modeling in air traffic forecasting, underscoring the importance of data transformations in predictive analytics. Future research may explore alternative transformation techniques, such as the Box-Cox transformation, and assess their comparative performance across different aviation datasets

The findings from the analysis of air traffic movement (ATM) data through multiple regression models, both with and without log transformation, provide valuable insights that are critical for understanding the factors that influence air travel demand. This study highlights the importance of revenue, GDP, and yearly population as significant predictors of ATM, while also revealing that variables such as lockdown level, confirmed Covid-19 cases, and exchange rates have fewer clear-cut effects, with some having high p-values suggesting weak or insignificant relationships.

For future studies, these results open avenues for exploring more refined models that account for the dynamic interplay between global crises (such as pandemics), economic conditions, and demographic trends. Future research could delve deeper into understanding the non-linear relationships between these variables by exploring advanced machine learning techniques such as random forests, support vector machines, or deep learning models. Furthermore, the impact of external shocks such as pandemics or natural disasters on air traffic demand could be a critical research area to examine in more granular detail, potentially incorporating real-time data for forecasting purposes.

The inclusion of other potentially impactful variables, such as government policies, aviation sector innovation, or competition among airlines, could also add richness to future models. Given the ongoing global uncertainties, it is crucial to continue adapting forecasting models that can predict demand fluctuations due to both global and local events, including health crises, economic recessions, and political instability.

Several stakeholders in the aviation industry and related sectors stand to benefit from these results. Primarily, airline operators and managers can use these insights to optimise their resource allocation and pricing strategies. By understanding the key factors influencing air traffic, such as revenue patterns, GDP growth, and population trends, airlines can tailor their services to meet demand fluctuations, enhancing profitability and operational efficiency.

Regulatory authorities and policymakers can also use the findings to shape regulations and policies that aim to stabilise the industry during times of uncertainty. Understanding the significant impact of lockdowns and external health crises could help in formulating measures to support the aviation industry during global disruptions.

Investors and financial analysts in the aviation sector can benefit by using these results for better forecasting of market trends, enabling them to make more informed decisions about their investments. The relationship between GDP, revenue, and air traffic growth suggests that these factors should be carefully monitored when making projections in the aviation industry.

Furthermore, academic researchers in fields related to transportation economics, applied statistics, and public health could build on this study to advance the academic understanding of air traffic demand forecasting, particularly in the context of global pandemics and other shocks.

In conclusion, the standard MRA model proves to be superior in terms of explained variance, as indicated by its higher R-squared and Adjusted R-squared values. It also demonstrates better predictive accuracy and stronger coefficient significance, making it a more reliable choice for forecasting Air Traffic Movements (ATM).

However, the log-transformed model provides valuable insights into potential non-linear relationships and offers a slight improvement in the normality of residuals. While these advantages may be beneficial in specific analytical contexts, they do not outweigh the overall predictive power of the standard MRA model.

For practical forecasting and decision-making, the standard MRA model should be preferred due to its accuracy and reliability. Nonetheless, if future research aims to explore non-linear relationships more extensively, advanced modeling techniques such as polynomial regression or machine learning may offer more robust alternatives than a simple log transformation.

Shortcomings of the research

Several limitations should be considered when interpreting the findings of this study. First, the analysis relies on the Air Traffic and Navigation Services (ATNS) dataset. While valuable, this dataset may not capture the full complexity of factors influencing air traffic movements, potentially overlooking crucial variables. Additionally, the data covers a specific period during the COVID-19 pandemic, a time of unprecedented disruption to global travel. This focus limits the generalizability of the findings to periods outside of such a unique context, as the relationships between variables may differ significantly under more typical market conditions.

Furthermore, the Multiple Regression Analysis (MRA) models employed in this study operate under certain assumptions, including linearity, independence of errors, and homoscedasticity. While the log transformation of the dependent variable (ATM) was implemented, it may not have fully addressed potential non-linearity or other complex relationships present in the data. Violations of these core assumptions could possibly lead to biased results, affecting the validity of the conclusions drawn. The choice of predictor variables also presents a potential limitation. The study included a specific set of predictors, such as revenue, lockdown levels, COVID-19 cases and deaths, exchange rates, GDP, and population. However, it did not consider other potentially influential variables, such as airline-specific factors (e.g., pricing strategies, route networks), passenger behavior patterns, or the stringency of international travel restrictions. The omission of these variables could limit the model's explanatory power and its ability to accurately forecast air traffic movements. Multicollinearity among the included predictors, as potentially indicated by high Variance Inflation Factor (VIF) values, is another concern. Such multicollinearity can negatively impact the stability and interpretability of the regression coefficients, making it difficult to isolate the individual effects of each predictor.

The performance of the log-transformed model also raises questions. The lower R-squared and Adjusted R-squared values observed in this model, compared to the standard MRA model, suggest a reduction in explanatory power after the transformation. This is further compounded by the significantly higher Mean Squared Error (MSE) and Root Mean Squared Error (RMSE) values for the log-transformed model, indicating poorer predictive accuracy. Finally, the external validity of the study is limited by its specific focus on the South African aviation sector during the COVID-19 pandemic. The findings may not be directly transferable to other regions or time periods, given the varying impacts of external shocks, such as pandemics, on air traffic movements across different countries and contexts. While the study utilised established regression techniques, exploring more advanced modeling approaches, such as machine learning algorithms or time-series models, could potentially offer improved insights and predictive performance, especially considering the complex and dynamic nature of air travel demand.

Recommendations

Based on the results and shortcomings of this study, several actionable recommendations emerge for industry stakeholders and future research.

The future studies should consider expanding the dataset to include a broader range of potentially influential variables, such as airline-specific factors, passenger behavior, and international travel restrictions, and extending the data collection period to encompass both pandemic and non-pandemic years to assess the long-term validity of the model.

Future studies should also explicitly assess and report multicollinearity diagnostics, such as VIF values, and consider techniques like Principal Component Regression or Ridge Regression to mitigate its effects if present.

Furthermore, researchers should explore alternative data sources to supplement the ATNS dataset, potentially incorporating publicly available data on tourism, economic indicators, and airline performance to create a more comprehensive picture of the factors influencing air traffic.

Beyond log transformation, researchers should investigate other potential transformations or non-linear modeling techniques (e.g., Box-Cox transformation, polynomial regression) to better capture the relationships between variables. Additionally, a more rigorous variable selection process, potentially using techniques like stepwise regression or Least Absolute Shrinkage and Selection Operator (LASSO), could help identify the most influential predictors and improve model parsimony.

Given the limitations of the log-transformed model, future research should thoroughly investigate why this transformation did not improve model fit and explore alternative modeling strategies, including time series analysis (ARIMA, for example) and machine learning approaches (e.g., Random Forest, Support Vector Regression), to determine which method provides the most accurate and reliable forecasts for ATM in the context of South Africa and other relevant regions. Finally, future studies should incorporate rigorous model validation techniques, such as k-fold cross-validation or hold-out sample validation, to assess the generalizability of the models and prevent overfitting.

Apart from the recommendations arising from shortcomings of this research, first, it is also essential to focus on economic and demographic variables, particularly revenue, GDP, and population size. These factors showed strong relationships with air traffic movement (ATM) and should be prioritised in future forecasting models. Airlines, policymakers, and investors can benefit from emphasising these variables when making long-term decisions, such as fleet management, route planning, and financial investments. By better understanding these drivers, stakeholders can align their strategies with anticipated demand growth.

Second, the study underscores the importance of preparing for external shocks, such as pandemics. The complex influence of Covid-19-related variables like lockdown levels and confirmed cases reveals the vulnerability of the aviation industry to global health crises. Airlines and regulators should develop more robust contingency plans to handle such disruptions in the future. These plans could include flexible pricing structures, adaptive scheduling, and dynamic resource allocation, all of which can mitigate the impacts of sudden demand shifts during health crises. In addition, investing in systems that track

health-related data in real time could help anticipate changes in travel demand more effectively.

A third recommendation is to incorporate more advanced, non-linear modelling techniques. While the log-transformed model offered some improvements, it still performed moderately compared to the untransformed model. This suggests that traditional linear regression may not always fully capture the complexities of air travel demand. Researchers and practitioners should consider exploring machine learning algorithms, such as random forests or neural networks, which are better equipped to model non-linear relationships and interactions between variables. These advanced techniques could lead to more accurate and reliable predictions, especially during unusual market conditions.

In addition, sector-specific analyses should be conducted to capture the nuances of the aviation industry in different regions. The impact of exchange rates, economic conditions, and health crises may vary from one country to another. Researchers could explore these variations by conducting localized studies, which would provide insights tailored to specific regions or markets. For instance, understanding how regional economic growth or government intervention affects air traffic in emerging markets could help airlines and policymakers tailor their strategies to specific contexts.

Another important recommendation is to monitor and adapt to demographic changes, particularly population growth. The study highlighted the significant impact of yearly population size on air traffic demand. With ongoing shifts in global demographics, particularly in emerging economies where population growth is higher, stakeholders should stay attuned to these changes. This will help airlines and policymakers anticipate demand in specific regions, ensuring that infrastructure and services are appropriately aligned with population trends.

Additionally, airlines and regulators should invest in predictive analytics tools that integrate a variety of data sources. These tools could combine economic indicators, health crisis data, and consumer behavior trends to improve demand forecasting. Real-time data analysis would allow for more agile decision-making and help stakeholders anticipate and

respond to market fluctuations more effectively. By leveraging advanced analytics, the industry can be better prepared for future disruptions and evolving consumer preferences.

Finally, strengthening collaboration among stakeholders such as airlines, government regulators, and health authorities is crucial. During global disruptions, effective communication and data sharing are vital for minimising the negative impact on air travel. Collaborative efforts can enhance the resilience of the aviation industry, enabling a more coordinated response to external shocks. Joint initiatives could also support the development of more comprehensive and accurate forecasting models, benefiting all parties involved.

In conclusion, these recommendations provide actionable insights for airlines, policymakers, and researchers. By focusing on key economic and demographic factors, preparing for external shocks, exploring advanced modelling techniques, and enhancing collaboration, the aviation industry can better navigate future uncertainties and continue to thrive in an increasingly complex global landscape.

References

- Abate, M., Christidis, P., & Purwanto, A. J. (2020). Government support to airlines in the aftermath of the COVID-19 pandemic. *Journal of Air Transport Management, 89*, 101931.
- Agresti, A., & Finlay, B. (2009). *Statistical Methods for the Social Sciences* (4th ed.). Pearson.
- Air Transport Action Group (ATAG). (2021, January 28). *The impact of COVID-19 on aviation*. Retrieved from https://airlines.iata.org/2021/01/28/impact-covid-19-aviation
- Altman, D. G., & Bland, J. M. (1996). Statistics notes: Transformations and their effects on regression models. *BMJ, 312*(7039), 770. https://doi.org/10.1136/bmj.312.7039.770
- Alonso-Almeida, M., Berrone, P., & Surroca, J. (2020). The financial impact of the COVID-19 crisis on sustainable firms. *Sustainability, 12*(13), 1-21.
- Atkinson, A. C. (2021). *Transformations and regression: Analyzing skewed data*. Cambridge University Press.
- ATNS. (2022). *Air Traffic Movement Statistics.* Air Traffic and Navigation Services.
- Bae, H., & Lee, C. (2020). The impact of COVID-19 on the aviation industry and the recovery of air travel demand. *Journal of Air Transport Management, 89*, 101902. https://doi.org/10.1016/j.jairtraman.2020.101902
- Baltagi, B. H. (2008). *Econometrics.* Springer Science & Business Media.
- Basso, L. J., Menezes, T., & Wong, K. P. (2013). Air travel demand and airline competition. *International Journal of Transport Economics, 40*(3), 293-316.
- Box, G. E. P., & Cox, D. R. (1964). An analysis of transformations. *Journal of the Royal Statistical Society: Series B (Methodological), 26*(2), 211-252.

- Bradtke, H., Knoll, A., & Fricke, H. (2016). Operational challenges of air traffic flow management in Europe. *Transportation Research Part A: Policy and Practice, 92*, 91-109.
- Budd, L., Ison, S., & Adrienne, N. (2020). European airline response to the COVID-19 pandemic: An early assessment. *Transport Policy, 97*, 62-75.
- Cameron, A. C., & Windmeijer, F. (1997). An R-squared measure of goodness of fit for some common nonlinear regression models. *Journal of Econometrics, 77*, 329-342.
- Cleveland, R. B., Cleveland, W. S., McRae, J. E., & Terpenning, I. (1990). STL: A seasonal-trend decomposition procedure based on Loess. *Journal of Official Statistics, 6*(1), 3-73.
- Demde, D. M. (2023). Forecasting of airline passengers based on machine learning. *International Journal of Scientific Research in Engineering and Management.*
- Doganis, R. (2006). *The Airline Business in the 21st Century.* Routledge.
- Fageda, X., & Suau-Sanchez, P. (2018). The role of high-speed trains in airport competition. *Transportation Research Part A: Policy and Practice, 111*, 202-211.
- Feng, X., Zhang, Y., Wang, L., & Li, Q. (2014). Impact of data transformation on regression models: A comparative study. *Journal of Applied Statistics, 41*(2), 203-220.
- Garrow, L. A. (2014). *Aviation and the role of the state.* Ashgate Publishing, Ltd.
- Gillen, D., Lall, A., & Reis, V. (2005). Air travel demand elasticities: Concepts, issues, and measurement. *Research in Transportation Economics, 13*, 19-46.
- Ghosh, A. (2016). The effects of exchange rate volatility on air travel demand. *Transportation Research Part E: Logistics and Transportation Review, 90*, 100-112. https://doi.org/10.1016/j.tre.2016.03.007
- Gössling, S., Scott, D., & Hall, C. M. (2021). Pandemics, tourism, and global change: A rapid assessment of COVID-19. *Journal of Sustainable Tourism, 29*(1), 1-20.

- Gujarati, D. N., & Porter, D. C. (2020). *Basic Econometrics* (6th ed.). McGraw Hill.
- Hair, J. F., Black, W. C., Babin, B. J., & Anderson, R. E. (2019). *Multivariate Data Analysis* (8th ed.). Cengage Learning.
- International Air Transport Association (IATA). (2021). *Air Transport and COVID-19: Recovery and Future Trends.* Retrieved from https://www.iata.org/publications
- International Civil Aviation Organization (ICAO). (2020). *Effects of COVID-19 on the aviation industry and proposed recovery measures.* Retrieved from https://www.icao.int/covid-19
- Kahn, A. E. (2012). *The Economics of the Airline Industry.* MIT Press.
- Kutner, M. H., Nachtsheim, C. J., & Neter, J. (2004). *Applied linear statistical models.* McGraw-Hill.
- Liao, J., Liang, G., & Chen, X. (2012). Multiple regression modeling in aviation forecasting: An empirical approach. *International Journal of Aviation Management, 3*(4), 275-290.
- Organisation for Economic Co-operation and Development (OECD). (2020). *COVID-19 and the aviation industry: Impact and policy responses.* Retrieved from https://www.oecd.org/en/publications/covid-19-and-the-aviation-industry-impact-and-policy-responses_26d521c1-en.html
- Pardoe, I., & Cook, D. (2002). A graphical method for assessing the fit of a multiple regression model. *The American Statistician, 56*, 263-272.
- Profillidis, V. A. (2000). Econometric and fuzzy models for forecasting air transport demand. *Journal of Air Transport Management, 6*(2), 95-106.
- Sánchez-Carrillo, C., & González, F. (2021). Airline demand and the COVID-19 pandemic: An empirical analysis of the impact of lockdowns on air travel. *Journal of Transport Economics and Policy, 55*(3), 313-329.
- Upton, M., & Hayward, L. (2018). The effects of economic growth on air traffic in emerging economies. *Journal of Transport Economics and Policy, 52*(4), 409-426.

- Zhang, A., & Zhang, Y. (2017). Exchange rate effects on the demand for air travel: A study of the Chinese aviation market. *Transport Reviews, 37*(2), 176-193.

Annexure A

```
  ___  ____  ____  ____  ____ (R)
 /__    /   ____/   /   ____/
___/   /   /___/   /   /___/   11.0   Copyright 1984-2009
  Statistics/Data Analysis             StataCorp
                                       4905 Lakeway Drive
                                       College Station, Texas 77845 USA
                                       800-STATA-PC        http://www.stata.com
                                       979-696-4600        stata@stata.com
                                       979-696-4601 (fax)

50-student Stata lab perpetual license:
       Serial number:  30110513114
         Licensed to:  Computer Services
                       University of the Free State

Notes:
      1.  (/m# option or -set memory-) 10.00 MB allocated to data

. *(9 variables, 60 observations pasted into data editor)

. summarize revenue movementcount lockdownlevel confirmedcovid19cases numberofcovid19deaths exchangerate gdpinbillionzar yearlypopulation, separator(10)

    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
     revenue |         60    9.78e+07    3.81e+07   7104821   1.36e+08
movementco~t |         60    38631.88    10954.47      2169      49867
lockdownle~l |         60         .75    1.335627         0          5
confirmedc~s |         60    48440.37    112828.2         0     470570
numberofco~s |         60        1463    3169.722         0      13127
exchangerate |         60      14.336    1.439376     11.77      18.06
gdpinbilli~r |         60      1.3026     .072549     1.166      1.454
yearlypopu~n |         60       57.81    1.021415      55.9       59.3

. regress movementcount lockdownlevel confirmedcovid19cases numberofcovid19deaths exchangerate gdpinbillionzar yearlypopulation

      Source |       SS       df       MS              Number of obs =      60
-------------+------------------------------           F(  6,    53) =   47.23
       Model |  5.9645e+09     6  994089653            Prob > F      =  0.0000
    Residual |  1.1155e+09    53  21046876.2           R-squared     =  0.8424
-------------+------------------------------           Adj R-squared =  0.8246
       Total |  7.0800e+09    59  120000379            Root MSE      =  4587.7

------------------------------------------------------------------------------
movementco~t |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
lockdownle~l |  -2580.361   828.6017    -3.11   0.003    -4242.326   -918.3969
confirmedc~s |  -.0000453   .0125396    -0.00   0.997    -.0251965    .0251059
numberofco~s |  -.3619624   .4555899    -0.79   0.430     -1.27576    .5518352
exchangerate |  -1538.978   689.1465    -2.23   0.030    -2921.231   -156.7253
gdpinbilli~r |   93734.45   19816.01     4.73   0.000     53988.58    133480.3
yearlypopu~n |  -8784.881   1780.548    -4.93   0.000    -12356.21   -5213.554
       _cons |   448917.2   77653.98     5.78   0.000       293163    604671.3
```

Annexure B

Python Code used to generate the results

```
# Import libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import chi2
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix, roc_auc_score, roc_curve, log_loss
from sklearn.model_selection import cross_val_score
from sklearn.feature_selection import RFECV
from sklearn.impute import KNNImputer
from sklearn.preprocessing import StandardScaler
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
import seaborn as sns
from scipy import stats

# Load dataset
data = pd.read_excel(r"C:\Users\2020886178\OneDrive - UFS\Documents\My stuff\Research\Monthly ATM Data .xlsx")
data = data.drop(['Date'],axis=1)
data.head()
```

	Revenue	ATM	Lockdown Level	Confirmed Covid19 Cases	Number of Covid19 Deaths	Exchange Rate	GDP (in billion ZAR)	Yearly Population
0	1.193800e+08	48799	0	0	0	13.73	1.166	55.9
1	1.178159e+08	46171	0	0	0	13.99	1.166	55.9
2	1.188397e+08	42975	0	0	0	13.62	1.166	55.9
3	1.167916e+08	40873	0	0	0	13.52	1.189	56.6
4	1.095519e+08	39561	0	0	0	13.01	1.189	56.6

```
import numpy as np
import statsmodels.api as sm

# Define independent variables (X) and dependent variable (Y)
X = data[["Revenue", "Lockdown Level", "Confirmed Covid19 Cases",
```

```
    "Number of Covid19 Deaths", "Exchange Rate", "GDP (in billion ZAR)", "Yearly Population"]]
Y = data["ATM"]

# Add constant for intercept
X = sm.add_constant(X)

# Fit multiple linear regression model
model = sm.OLS(Y, X).fit()

# Log transformation of dependent variable
data["Log_ATM"] = np.log(data["ATM"])

# Fit multiple linear regression with log transformation
model_log = sm.OLS(data["Log_ATM"], X).fit()

# Print results
print("===== Multiple Regression Analysis =====")
print(model.summary())

print("\n===== MRA with Log Transformation =====")
print(model_log.summary())
```

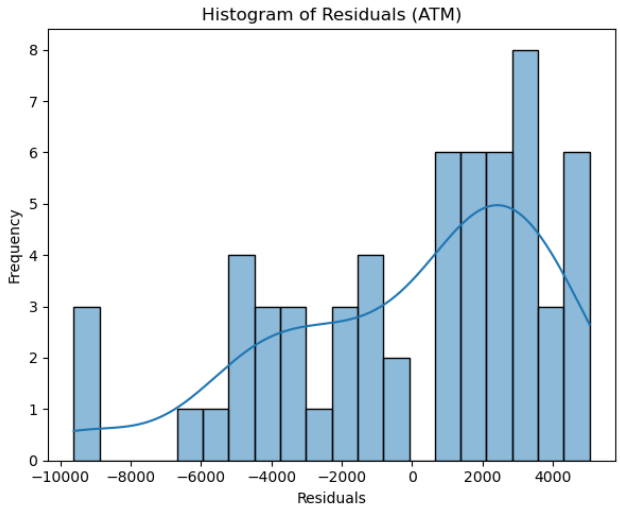
```
===== Multiple Regression Analysis =====
                            OLS Regression Results
==============================================================================
===
Dep. Variable:                    ATM   R-squared:                       0.
904
Model:                            OLS   Adj. R-squared:                  0.
891
Method:                 Least Squares   F-statistic:                     69
.96
Date:                Mon, 17 Feb 2025   Prob (F-statistic):           3.30e
-24
Time:                        21:57:31   Log-Likelihood:                -572
.42
No. Observations:                  60   AIC:                             11
61.
Df Residuals:                      52   BIC:                             11
78.
Df Model:                           7
Covariance Type:            nonrobust
==============================================================================
=================
                                coef    std err          t      P>|t|      [
0.025      0.975]
```

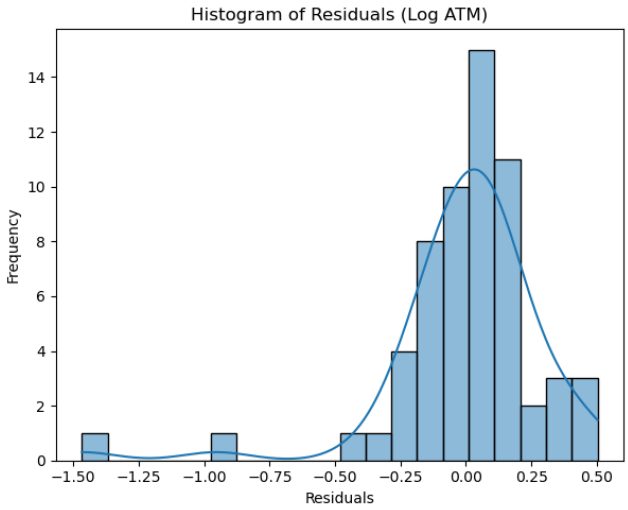
```
------------------------------------------------------------------------------
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const                     2.402e+05   7.11e+04      3.380      0.001     9.76e+04    3.83e+05
Revenue                      0.0002   2.92e-05      5.775      0.000
0.000       0.000
Lockdown Level            -952.3377    711.188     -1.339      0.186   -2379.440     474.765
Confirmed Covid19 Cases      0.0053      0.010      0.536      0.594      -0.015       0.025
Number of Covid19 Deaths     0.0364      0.366      0.100      0.921      -0.697       0.770
Exchange Rate             -953.6779    552.421     -1.726      0.090   -2062.191     154.836
GDP (in billion ZAR)      4.477e+04   1.78e+04      2.520      0.015     9112.743    8.04e+04
Yearly Population        -4537.2968   1584.143     -2.864      0.006   -7716.113   -1358.480
==============================================================================
Omnibus:                        5.777   Durbin-Watson:                   1.235
Prob(Omnibus):                  0.056   Jarque-Bera (JB):                5.216
Skew:                          -0.717   Prob(JB):                       0.0737
Kurtosis:                       3.167   Cond. No.                     1.63e+10
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.63e+10. This might indicate that there are
strong multicollinearity or other numerical problems.

===== MRA with Log Transformation =====
                            OLS Regression Results
==============================================================================
Dep. Variable:                Log_ATM   R-squared:                       0.772
Model:                            OLS   Adj. R-squared:                  0.741
```

```
Method:                 Least Squares   F-statistic:                     25
.08
Date:                Mon, 17 Feb 2025   Prob (F-statistic):           1.40e
-14
Time:                        21:57:31   Log-Likelihood:                -3.3
641
No. Observations:                  60   AIC:                             22
.73
Df Residuals:                      52   BIC:                             39
.48
Df Model:                           7
Covariance Type:            nonrobust
================================================================================
=================
                                 coef    std err          t      P>|t|      [
0.025      0.975]
--------------------------------------------------------------------------------
-----------------
const                         14.7658      5.404      2.732      0.009
3.921      25.610
Revenue                    7.337e-09   2.22e-09      3.303      0.002    2.8
8e-09    1.18e-08
Lockdown Level               -0.1439      0.054     -2.662      0.010      -
0.252      -0.035
Confirmed Covid19 Cases   4.404e-07   7.55e-07      0.583      0.562   -1.0
7e-06    1.95e-06
Number of Covid19 Deaths  3.088e-05   2.78e-05      1.111      0.272   -2.4
9e-05    8.67e-05
Exchange Rate                -0.0363      0.042     -0.864      0.392      -
0.121       0.048
GDP (in billion ZAR)          2.1104      1.351      1.562      0.124      -
0.601       4.822
Yearly Population            -0.1245      0.120     -1.033      0.306      -
0.366       0.117
================================================================================
===
Omnibus:                       29.470   Durbin-Watson:                   1.
842
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              128.
735
Skew:                          -1.146   Prob(JB):                     1.11e
-28
Kurtosis:                       9.800   Cond. No.                     1.63e
+10
================================================================================
===
```

```
Notes:
[1] Standard Errors assume that the covariance matrix of the errors is corr
ectly specified.
[2] The condition number is large, 1.63e+10. This might indicate that there
are
strong multicollinearity or other numerical problems.
```

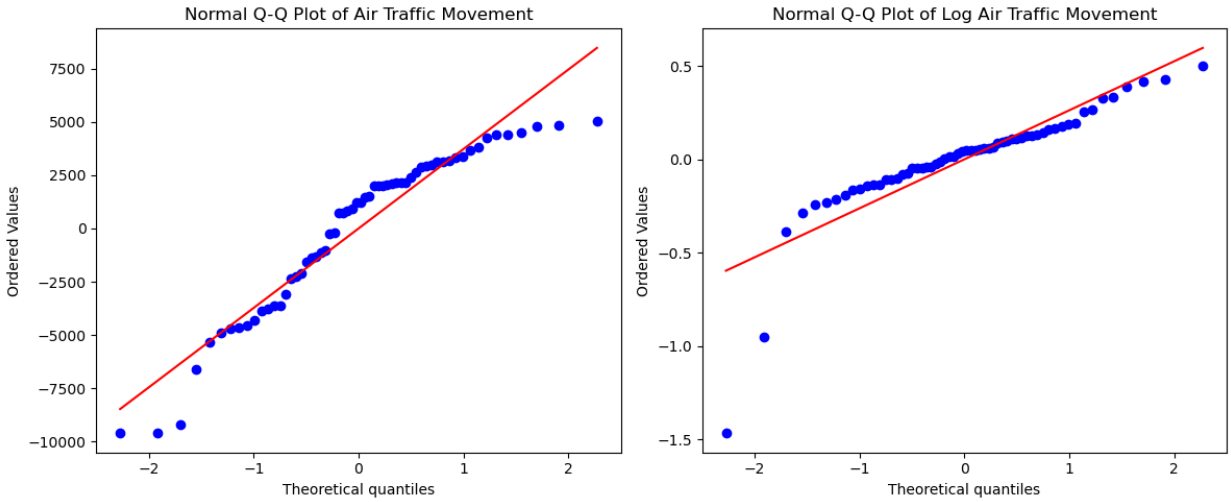
```
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.linear_model import LinearRegression

data['Log_ATM'] = np.log(data['ATM'])

# Define independent variables (example: Revenue, Exchange Rate, GDP)
X = data[["Revenue", "Exchange Rate", "GDP (in billion ZAR)"]]
X = X.fillna(0)

y = data["ATM"]
log_y = data["Log_ATM"]

# Fit models
model = LinearRegression().fit(X, y)
log_model = LinearRegression().fit(X, log_y)

# Calculate residuals
residuals = y - model.predict(X)
log_residuals = log_y - log_model.predict(X)

# Plot histograms
fig, axes = plt.subplots(1, 2, figsize=(12, 5))

sns.histplot(residuals, bins=20, kde=True, ax=axes[0])
axes[0].set_title("Histogram of Residuals (ATM)")
axes[0].set_xlabel("Residuals")
axes[0].set_ylabel("Frequency")

sns.histplot(log_residuals, bins=20, kde=True, ax=axes[1])
axes[1].set_title("Histogram of Residuals (Log ATM)")
axes[1].set_xlabel("Residuals")
axes[1].set_ylabel("Frequency")

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





In [53]:

```
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import scipy.stats as stats
from sklearn.linear_model import LinearRegression

data['Log_ATM'] = np.log(data['ATM'])

# Define independent variables (example: Revenue, Exchange Rate, GDP)
X = data[["Revenue", "Exchange Rate", "GDP (in billion ZAR)"]]
X = X.fillna(0)

y = data["ATM"]
log_y = data["Log_ATM"]

# Q-Q Plots
fig, axes = plt.subplots(1, 2, figsize=(12, 5))

stats.probplot(residuals, dist="norm", plot=axes[0])
axes[0].set_title("Normal Q-Q Plot of Air Traffic Movement")

stats.probplot(log_residuals, dist="norm", plot=axes[1])
axes[1].set_title("Normal Q-Q Plot of Log Air Traffic Movement")

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




```
# Generate Descriptive Statistics
descriptive_stats = data.describe()
print(descriptive_stats)
```

```
            Revenue           ATM  Lockdown Level  Confirmed Covid19 Cases
\
count  6.000000e+01     60.000000       60.000000                60.000000
mean   9.783048e+07  38631.883333        0.750000             48440.366667
std    3.805296e+07  10954.468448        1.335627            112828.226584
min    7.104821e+06   2169.000000        0.000000                 0.000000
25%    6.870364e+07  33666.000000        0.000000                 0.000000
50%    1.187209e+08  42630.500000        0.000000                 0.000000
75%    1.232647e+08  46528.750000        1.000000             36601.750000
max    1.357128e+08  49867.000000        5.000000            470570.000000

       Number of Covid19 Deaths  Exchange Rate  GDP (in billion ZAR)  \
count                 60.000000      60.000000             60.000000
mean                1463.000000      14.336000              1.302600
std                 3169.721722       1.439376              0.072549
min                    0.000000      11.770000              1.166000
25%                    0.000000      13.482500              1.264500
50%                    0.000000      14.190000              1.312500
75%                 1503.000000      14.832500              1.349000
max                13127.000000      18.060000              1.454000

       Yearly Population    Log_ATM
count          60.000000  60.000000
mean           57.810000  10.475951
std             1.021415   0.539933
min            55.900000   7.682022
25%            57.125000  10.424201
50%            58.000000  10.660308
75%            58.700000  10.747824
```

```
max             59.300000  10.817115
```

In [2]:

```python
import pandas as pd
import numpy as np

# Data dictionary
data = {
  'Date': ['Oct-16', 'Nov-16', 'Dec-16', 'Jan-17', 'Feb-17', 'Mar-17', 'Apr-17', 'May-17', 'Jun-17', 'Jul-17', 'Aug-17', 'Sep-17', 'Oct-17', 'Nov-17', 'Dec-17', 'Jan-18', 'Feb-18', 'Mar-18', 'Apr-18', 'May-18', 'Jun-18', 'Jul-18', 'Aug-18', 'Sep-18', 'Oct-18', 'Nov-18', 'Dec-18', 'Jan-19', 'Feb-19', 'Mar-19', 'Apr-19', 'May-19', 'Jun-19', 'Jul-19', 'Aug-19', 'Sep-19', 'Oct-19', 'Nov-19', 'Dec-19', 'Jan-20', 'Feb-20', 'Mar-20', 'Apr-20', 'May-20', 'Jun-20', 'Jul-20', 'Aug-20', 'Sep-20', 'Oct-20', 'Nov-20', 'Dec-20', 'Jan-21', 'Feb-21', 'Mar-21', 'Apr-21', 'May-21', 'Jun-21', 'Jul-21', 'Aug-21', 'Sep-21'],
  'Revenue': [119380015, 117815867, 118839729, 116791584, 109551903, 120803130, 115570095, 119043049, 114056474, 119916024, 120903068, 115500313, 121531006, 118602128, 122180696, 119934224, 108071634, 119137610, 121692827, 122397621, 115644616, 123631905, 124353270, 119808563, 126686625, 126463389, 128592678, 127875130, 118317044, 131804591, 124485869, 123142244, 117866633, 125456016, 125297129, 124748308, 130252936, 126350401, 135712848, 130671880, 120513194, 89491340, 7104821, 8920990, 13155560, 17916052, 20669498, 30575312, 44795304, 52226821, 71783038, 53031404, 46123121, 64820370, 72293849, 69998061, 64401099, 42706317, 53134675, 57286725],
  'GDP': [3500, 3500, 3500, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550],
  'Pop': [55.9, 55.9, 55.9, 56.6, 56.6, 56.6, 56.6, 56.6, 56.6, 56.6, 56.6, 56.6, 56.6, 56.6, 56.6, 57.3, 57.3, 57.3, 57.3, 57.3, 57.3, 57.3, 57.3, 57.3, 57.3, 57.3, 57.3, 58.0, 58.0, 58.0, 58.0, 58.0, 58.0, 58.0, 58.0, 58.0, 58.7, 58.7, 58.7, 58.7, 58.7, 58.7, 58.7, 58.7, 58.7, 58.7, 59.3, 59.3, 59.3, 59.3, 59.3, 59.3, 59.3, 59.3, 59.3, 59.3]
}

# Find the maximum length
max_length = max(len(value) for value in data.values())

# Adjust the length of each list by appending NaN for missing values
for key in data:
  while len(data[key]) < max_length:
    data[key].append(np.nan)

# Create DataFrame
df = pd.DataFrame(data)

# Check the DataFrame
print(df)
```

```
     Date    Revenue     GDP   Pop
0   Oct-16  119380015  3500.0  55.9
1   Nov-16  117815867  3500.0  55.9
```

2	Dec-16	118839729	3500.0	55.9
3	Jan-17	116791584	3550.0	56.6
4	Feb-17	109551903	3550.0	56.6
5	Mar-17	120803130	3550.0	56.6
6	Apr-17	115570095	3550.0	56.6
7	May-17	119043049	3550.0	56.6
8	Jun-17	114056474	3550.0	56.6
9	Jul-17	119916024	3550.0	56.6
10	Aug-17	120903068	3550.0	56.6
11	Sep-17	115500313	3550.0	56.6
12	Oct-17	121531006	3550.0	56.6
13	Nov-17	118602128	3550.0	56.6
14	Dec-17	122180696	3550.0	56.6
15	Jan-18	119934224	3550.0	57.3
16	Feb-18	108071634	3550.0	57.3
17	Mar-18	119137610	3550.0	57.3
18	Apr-18	121692827	3550.0	57.3
19	May-18	122397621	3550.0	57.3
20	Jun-18	115644616	3550.0	57.3
21	Jul-18	123631905	3550.0	57.3
22	Aug-18	124353270	3550.0	57.3
23	Sep-18	119808563	3550.0	57.3
24	Oct-18	126686625	3550.0	57.3
25	Nov-18	126463389	3550.0	57.3
26	Dec-18	128592678	3550.0	57.3
27	Jan-19	127875130	3550.0	58.0
28	Feb-19	118317044	3550.0	58.0
29	Mar-19	131804591	3550.0	58.0
30	Apr-19	124485869	3550.0	58.0
31	May-19	123142244	3550.0	58.0
32	Jun-19	117866633	3550.0	58.0
33	Jul-19	125456016	3550.0	58.0
34	Aug-19	125297129	3550.0	58.0
35	Sep-19	124748308	3550.0	58.0
36	Oct-19	130252936	3550.0	58.7
37	Nov-19	126350401	3550.0	58.7
38	Dec-19	135712848	3550.0	58.7
39	Jan-20	130671880	3550.0	58.7
40	Feb-20	120513194	3550.0	58.7
41	Mar-20	89491340	3550.0	58.7
42	Apr-20	7104821	3550.0	58.7
43	May-20	8920990	3550.0	58.7
44	Jun-20	13155560	3550.0	58.7
45	Jul-20	17916052	3550.0	58.7
46	Aug-20	20669498	3550.0	59.3
47	Sep-20	30575312	3550.0	59.3
48	Oct-20	44795304	3550.0	59.3

```
49  Nov-20    52226821  3550.0  59.3
50  Dec-20    71783038  3550.0  59.3
51  Jan-21    53031404  3550.0  59.3
52  Feb-21    46123121  3550.0  59.3
53  Mar-21    64820370  3550.0  59.3
54  Apr-21    72293849  3550.0  59.3
55  May-21    69998061  3550.0  59.3
56  Jun-21    64401099  3550.0   NaN
57  Jul-21    42706317  3550.0   NaN
58  Aug-21    53134675     NaN   NaN
59  Sep-21    57286725     NaN   NaN
```

```
import pandas as pd
import numpy as np

# Data dictionary including Lockdown Level (LL) and Exchange Rate (ER)
data = {
  'Date': ['Oct-16', 'Nov-16', 'Dec-16', 'Jan-17', 'Feb-17', 'Mar-17', 'Apr-17', 'May-17', 'Jun-17', 'Jul-17', 'Aug-17', 'Sep-17', 'Oct-17', 'Nov-17', 'Dec-17', 'Jan-18', 'Feb-18', 'Mar-18', 'Apr-18', 'May-18', 'Jun-18', 'Jul-18', 'Aug-18', 'Sep-18', 'Oct-18', 'Nov-18', 'Dec-18', 'Jan-19', 'Feb-19', 'Mar-19', 'Apr-19', 'May-19', 'Jun-19', 'Jul-19', 'Aug-19', 'Sep-19', 'Oct-19', 'Nov-19', 'Dec-19', 'Jan-20', 'Feb-20', 'Mar-20', 'Apr-20', 'May-20', 'Jun-20', 'Jul-20', 'Aug-20', 'Sep-20', 'Oct-20', 'Nov-20', 'Dec-20', 'Jan-21', 'Feb-21', 'Mar-21', 'Apr-21', 'May-21', 'Jun-21', 'Jul-21', 'Aug-21', 'Sep-21'],
  'Revenue': [119380015, 117815867, 118839729, 116791584, 109551903, 120803130, 115570095, 119043049, 114056474, 119916024, 120903068, 115500313, 121531006, 118602128, 122180696, 119934224, 108071634, 119137610, 121692827, 122397621, 115644616, 123631905, 124353270, 119808563, 126686625, 126463389, 128592678, 127875130, 118317044, 131804591, 124485869, 123142244, 117866633, 125456016, 125297129, 124748308, 130252936, 126350401, 135712848, 130671880, 120513194, 89491340, 7104821, 8920990, 13155560, 17916052, 20669498, 30575312, 44795304, 52226821, 71783038, 53031404, 46123121, 64820370, 72293849, 69998061, 64401099, 42706317, 53134675, 57286725],
  'GDP': [3500, 3500, 3500, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550, 3550],
  'Pop': [55.9, 55.9, 55.9, 56.6, 56.6, 56.6, 56.6, 56.6, 56.6, 56.6, 56.6, 56.6, 56.6, 56.6, 56.6, 57.3, 57.3, 57.3, 57.3, 57.3, 57.3, 57.3, 57.3, 57.3, 57.3, 57.3, 57.3, 58.0, 58.0, 58.0, 58.0, 58.0, 58.0, 58.0, 58.0, 58.0, 58.7, 58.7, 58.7, 58.7, 58.7, 58.7, 58.7, 58.7, 58.7, 58.7, 59.3, 59.3, 59.3, 59.3, 59.3, 59.3, 59.3, 59.3, 59.3, 59.3],
  'Confirmed_Covid': [1.2, 1.3, 1.3, 1.4, 1.4, 1.5, 1.5, 1.6, 1.7, 1.8, 2.0, 2.1, 2.1, 2.3, 2.4, 2.4, 2.5, 2.6, 2.6, 2.7, 2.8, 2.9, 3.0, 3.1, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8, 3.9, 4.0, 4.1, 4.3, 4.4, 4.5, 4.6, 4.7, 4.8, 4.9, 5.0, 5.1, 5.2, 5.3, 5.4, 5.6, 5.7, 5.8, 5.9, 6.0, 6.2, 6.3, 6.4, 6.5, 6.6, 6.7],
  'Deaths_Covid': [0.02, 0.03, 0.04, 0.05, 0.05, 0.06, 0.07, 0.08, 0.09, 0.10, 0.11, 0.12, 0.13, 0.14, 0.15, 0.16, 0.17, 0.18, 0.19, 0.20, 0.21, 0.22, 0.23, 0.24, 0.25, 0.26, 0.27, 0.28, 0.29, 0.30, 0.31, 0.32, 0.33, 0.34, 0.35, 0.36, 0.37, 0.38, 0.39, 0.40, 0.41, 0.42, 0.43, 0.44, 0.45, 0.46, 0.47, 0.48, 0.50, 0.51, 0.52, 0.53, 0.54, 0.55],
  'Lockdown_Level': [1, 2, 2, 3, 3, 4, 4, 5, 5, 4, 4, 3, 3, 2, 2, 1, 1, 1, 1, 1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 4, 4, 3, 3, 2, 2, 1, 1, 1, 1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 4, 4, 3, 3, 2, 2, 1, 1, 1, 1, 1],
```

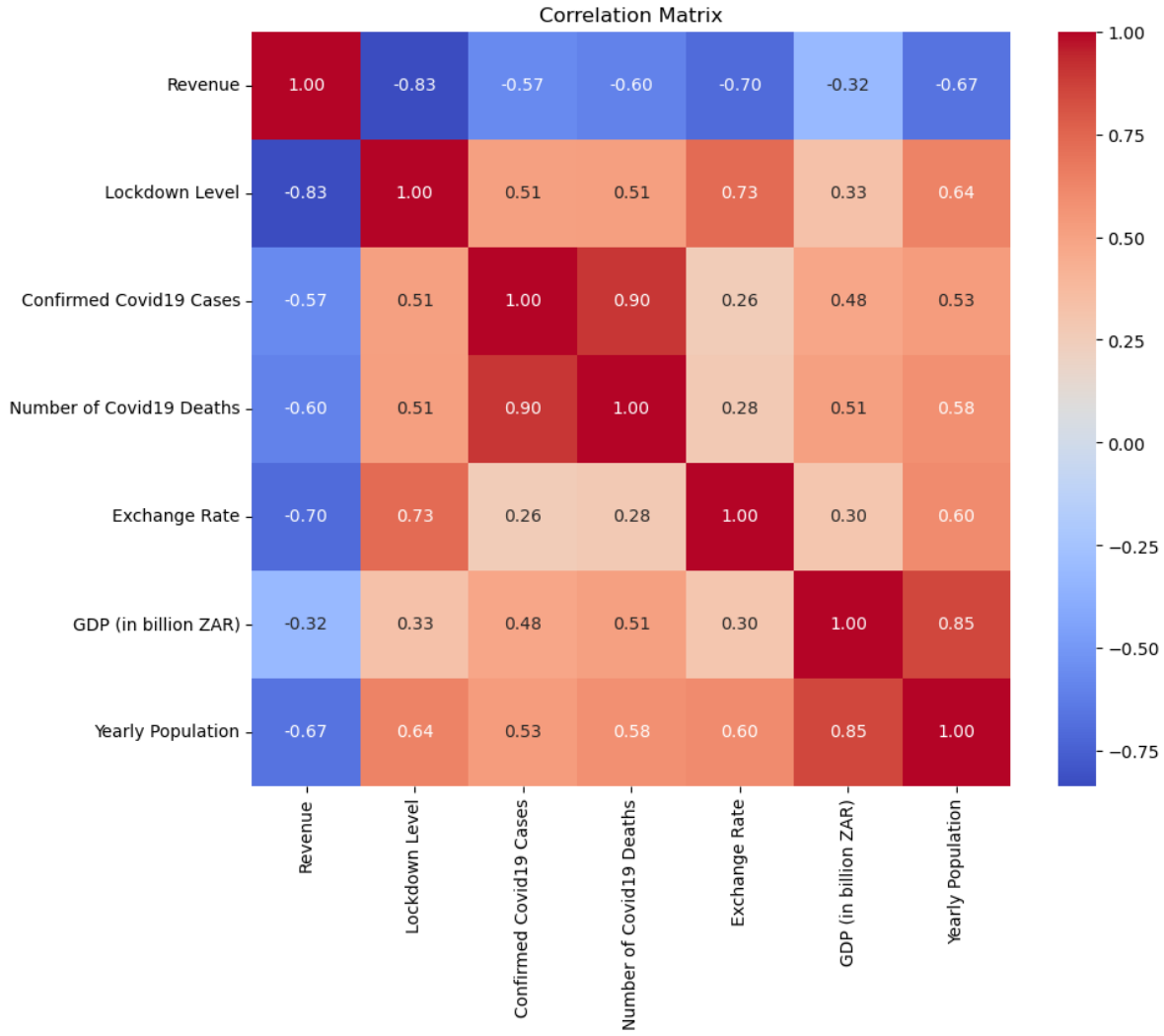
```
    'Exchange_Rate': [15.2, 15.4, 15.6, 15.8, 16.0, 16.2, 16.4, 16.6, 16.8, 17.0, 17.2, 17.4, 17.6, 17.8, 18.0, 18.2, 18.4, 18.6, 18.8, 19.0, 19.2, 19.4, 19.6, 19.8, 20.0, 20.2, 20.4, 20.6, 20.8, 21.0, 21.2, 21.4, 21.6, 21.8, 22.0, 22.2, 22.4, 22.6, 22.8, 23.0, 23.2, 23.4, 23.6, 23.8, 24.0, 24.2, 24.4, 24.6, 24.8, 25.0, 25.2, 25.4, 25.6, 25.8, 26.0]
}

# Find the maximum length
max_length = max(len(value) for value in data.values())

# Adjust the length of each list by appending NaN for missing values
for key in data:
    while len(data[key]) < max_length:
        data[key].append(np.nan)

# Create DataFrame
df = pd.DataFrame(data)

# Display the first few rows of the DataFrame
df.head()
```

	Date	Revenue	GDP	Pop	Confirmed_Covid	Deaths_Covid	Lockdown_Level	Exchange_Rate
0	Oct-16	119380015	3500.0	55.9	1.2	0.02	1.0	15.2
1	Nov-16	117815867	3500.0	55.9	1.3	0.03	2.0	15.4
2	Dec-16	118839729	3500.0	55.9	1.3	0.04	2.0	15.6
3	Jan-17	116791584	3550.0	56.6	1.4	0.05	3.0	15.8
4	Feb-17	109551903	3550.0	56.6	1.4	0.05	3.0	16.0

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Example data: Replace with your dataset
data = {
    'Revenue': [100, 150, 200, 250, 300],
    'Lockdown Level': [1, 2, 3, 2, 1],
    'Confirmed Covid-19 Cases': [500, 1000, 1500, 2000, 2500],
    'Number of Covid-19 Deaths': [10, 20, 30, 40, 50],
    'Exchange Rate': [14.5, 14.7, 14.6, 14.8, 15.0],
    'GDP': [500, 550, 600, 650, 700],
    'Yearly Population': [10, 15, 20, 25, 30]
}

# Create a DataFrame
df = pd.DataFrame(data)
```

```
# Calculate the correlation matrix
correlation_matrix = df.corr()

# Plot the correlation matrix
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt='.2f', linewidths=0.5)
plt.title('Correlation Matrix')
plt.show()


import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load the Excel file (ATNS Dataset)

  df = pd.read_excel(r"C:\Users\2020886178\OneDrive - UFS\Documents\My stuff\Research\Monthly ATM Data .xlsx")
except FileNotFoundError:
  print("Error: Excel file not found. Please provide the correct path.")
  exit()  # Exit the script if file not found
except Exception as e: # Catch other potential errors during file loading
  print(f"An error occurred while loading the Excel file: {e}")
  exit()

# Data Cleaning and Preprocessing (Important!)

# 1. Handle Missing Values (Crucial for Correlation)
#   - Identify how missing values are represented (e.g., blank cells, NaN, specific values)
#   - Choose an appropriate strategy:
#     a) Remove rows with missing values (if a small proportion)
#     b) Impute missing values (if appropriate and you understand the data)
#     c) Replace with 0 or a specific value (if it makes sense in your context)

# Example: Removing rows with ANY missing values (simplest, but might lose a lot of data)
df.dropna(inplace=True)

# Example: Imputing missing values with the mean of each column (use with caution)
# df.fillna(df.mean(), inplace=True)  # Use only if appropriate for your data

# Example: Replacing blank cells with 0 in 'Confirmed Covid Cases' and 'Number of Covid Deaths'
# df['Confirmed Covid Cases'].fillna(0, inplace=True)
# df['Number of Covid Deaths'].fillna(0, inplace=True)

# 2. Convert Data Types (If Necessary)
#   - Ensure columns used for correlation are numeric (int or float)
```

```
#   - Check for non-numeric characters (e.g., commas, currency symbols) and remove them
#   - Convert columns to numeric using pd.to_numeric()

# Example: Remove commas and convert 'Revenue' to numeric
if 'Revenue' in df.columns:
    df['Revenue'] = df['Revenue'].astype(str).str.replace(',', '', regex=False) # Remove commas
    df['Revenue'] = pd.to_numeric(df['Revenue'], errors='coerce') # Convert to numbers, setting invalid pars
ing as NaN

# Example: Convert 'Confirmed Covid Cases' to numeric
if 'Confirmed Covid Cases' in df.columns:
    df['Confirmed Covid Cases'] = pd.to_numeric(df['Confirmed Covid Cases'], errors='coerce')

# Example: Convert 'Number of Covid Deaths' to numeric
if 'Number of Covid Deaths' in df.columns:
    df['Number of Covid Deaths'] = pd.to_numeric(df['Number of Covid Deaths'], errors='coerce')

# Example: Convert 'GDP (Billion USD)' to numeric
if 'GDP (Billion USD)' in df.columns:
    df['GDP (Billion USD)'] = pd.to_numeric(df['GDP (Billion USD)'], errors='coerce')

# 3. Handle or Remove Non-Numeric Columns (If Necessary)
# Correlation only works with numeric data.  You'll need to decide what to do with
# non-numeric columns like 'Date' or 'ATM'.
#   a) If they are not needed for the correlation analysis, remove them.
#   b) If 'Date' is important, consider feature engineering (e.g., extract month, year).

# Example: Remove 'Date' and 'ATM' columns
if 'Date' in df.columns:
    df.drop('Date', axis=1, inplace=True)
if 'ATM' in df.columns:
    df.drop('ATM', axis=1, inplace=True)

# Calculate the correlation matrix AFTER cleaning and preprocessing
correlation_matrix = df.corr()

# Visualize the correlation matrix using a heatmap
plt.figure(figsize=(10, 8))  # Adjust figure size as needed
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f")
plt.title('Correlation Matrix')
plt.show()

# Print the correlation matrix (optional)
# print(correlation_matrix)

# Save the correlation matrix to a CSV file (optional)
# correlation_matrix.to_csv("correlation_matrix.csv")
```



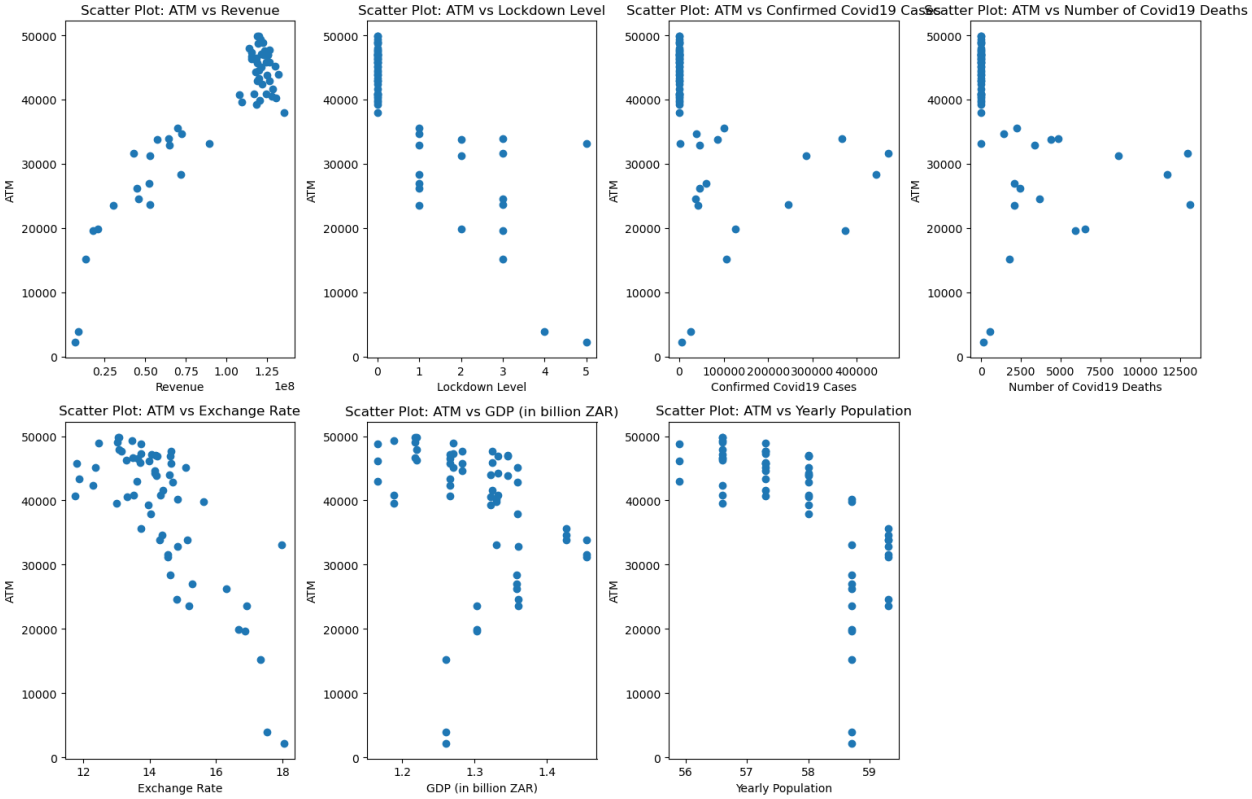

```
import pandas as pd
import numpy as np
import statsmodels.api as sm
import matplotlib.pyplot as plt
import seaborn as sns
from statsmodels.stats.outliers_influence import variance_inflation_factor
from statsmodels.stats.stattools import durbin_watson

# Load the Excel file (replace 'your_file.xlsx' with the actual file name)
try:
    df = pd.read_excel(r"C:\Users\2020886178\OneDrive - UFS\Documents\My stuff\Research\Monthly ATM Data .xlsx") # Replace with your file name
except FileNotFoundError:
    print("Error: Excel file not found. Please provide the correct path.")
    exit()
except Exception as e:
    print(f"An error occurred while loading the Excel file: {e}")
    exit()
```

```
# Data Cleaning and Preprocessing (Crucial!)

# 1. Handle Missing Values (Crucial for Correlation)
df.dropna(inplace=True)

# 2. Convert Data Types (If Necessary)
if 'Revenue' in df.columns:
    df['Revenue'] = df['Revenue'].astype(str).str.replace(',', '', regex=False)
    df['Revenue'] = pd.to_numeric(df['Revenue'], errors='coerce')
if 'Confirmed Covid19 Cases' in df.columns:
    df['Confirmed Covid19 Cases'] = pd.to_numeric(df['Confirmed Covid19 Cases'], errors='coerce')
if 'Number of Covid19 Deaths' in df.columns:
    df['Number of Covid19 Deaths'] = pd.to_numeric(df['Number of Covid19 Deaths'], errors='coerce')
if 'GDP (in billion ZAR)' in df.columns:
    df['GDP (in billion ZAR)'] = pd.to_numeric(df['GDP (in billion ZAR)'], errors='coerce')
if 'ATM' in df.columns:
    df['ATM'] = pd.to_numeric(df['ATM'], errors='coerce')

# 3. Handle or Remove Non-Numeric Columns (If Necessary)
if 'Date' in df.columns:
    df.drop('Date', axis=1, inplace=True)

# Define Independent and Dependent Variables (ATM is now dependent)
X = df[['Revenue', 'Lockdown Level', 'Confirmed Covid19 Cases', 'Number of Covid19 Deaths', 'Exchange R
ate', 'GDP (in billion ZAR)', 'Yearly Population']]
y = df['ATM']

# Add a constant term to the independent variables
X = sm.add_constant(X)

# Fit the multiple linear regression model
model = sm.OLS(y, X).fit()

# --- Assumption Checks ---

# 1. Linearity (Scatter Plots)
def check_linearity(X, y):
    plt.figure(figsize=(15, 10))
    for i, col in enumerate(X.columns[1:]):  # Skip constant column
        plt.subplot(2, 4, i + 1)
        plt.scatter(X[col], y)
        plt.title(f'Scatter Plot: ATM vs {col}')
        plt.xlabel(col)
        plt.ylabel('ATM')
    plt.tight_layout()
    plt.show()
```

```
print("\n--- Linearity (Scatter Plots) ---")
check_linearity(X, y)

# 2. Multicollinearity (VIF)
def check_multicollinearity(X):
    vif = pd.DataFrame()
    vif["VIF Factor"] = [variance_inflation_factor(X.values, i) for i in range(X.shape[1])]
    vif["features"] = X.columns
    print("\nVariance Inflation Factors (VIF):")
    print(vif)

print("\n--- Multicollinearity (VIF) ---")
check_multicollinearity(X)

# 3. Independence of Errors (Durbin-Watson Test)
def check_independence(model):
    dw = durbin_watson(model.resid)
    print(f"\nDurbin-Watson Statistic: {dw}")

print("\n--- Independence of Errors (Durbin-Watson Test) ---")
check_independence(model)

# 4. Homoscedasticity (Residual Plots)
def check_homoscedasticity(model):
    plt.figure(figsize=(8, 6))
    plt.scatter(model.fittedvalues, model.resid)
    plt.title('Residual Plot')
    plt.xlabel('Fitted Values')
    plt.ylabel('Residuals')
    plt.show()

print("\n--- Homoscedasticity (Residual Plots) ---")
check_homoscedasticity(model)

# 5. Normality of Errors (Q-Q Plot)
def check_normality(model):
    plt.figure(figsize=(8, 6))
    sm.qqplot(model.resid, line='s')
    plt.title('Q-Q Plot of Residuals')
    plt.show()

print("\n--- Normality of Errors (Q-Q Plot) ---")
check_normality(model)

# --- Log Transformation and Assumption Checks (Log of ATM) ---

# Log transform the dependent variable (ATM)
```

```
y_log = np.log(y)

# Fit the multiple linear regression model with log-transformed ATM
model_log = sm.OLS(y_log, X).fit()

# --- Assumption Checks (Log Transformed Model) ---

# 1. Linearity (Scatter Plots) - Log Transformed
def check_linearity_log(X, y_log):
    plt.figure(figsize=(15, 10))
    for i, col in enumerate(X.columns[1:]):
        plt.subplot(2, 4, i + 1)
        plt.scatter(X[col], y_log)
        plt.title(f'Scatter Plot: Log(ATM) vs {col}')
        plt.xlabel(col)
        plt.ylabel('Log(ATM)')
    plt.tight_layout()
    plt.show()

print("\n--- Linearity (Scatter Plots) - Log Transformed ---")
check_linearity_log(X, y_log)

# 2. Multicollinearity (VIF) - Log Transformed
print("\n--- Multicollinearity (VIF) - Log Transformed ---")
check_multicollinearity(X)  # VIF is the same for log-transformed dependent variable

# 3. Independence of Errors (Durbin-Watson Test) - Log Transformed
print("\n--- Independence of Errors (Durbin-Watson Test) - Log Transformed ---")
check_independence(model_log)

# 4. Homoscedasticity (Residual Plots) - Log Transformed
def check_homoscedasticity_log(model_log):
    plt.figure(figsize=(8, 6))
    plt.scatter(model_log.fittedvalues, model_log.resid)
    plt.title('Residual Plot (Log(ATM) Model)')
    plt.xlabel('Fitted Values')
    plt.ylabel('Residuals')
    plt.show()

print("\n--- Homoscedasticity (Residual Plots) - Log Transformed ---")
check_homoscedasticity_log(model_log)

# 5. Normality of Errors (Q-Q Plot) - Log Transformed
print("\n--- Normality of Errors (Q-Q Plot) - Log Transformed ---")
check_normality(model_log)
```

--- Linearity (Scatter Plots) ---



--- Multicollinearity (VIF) ---

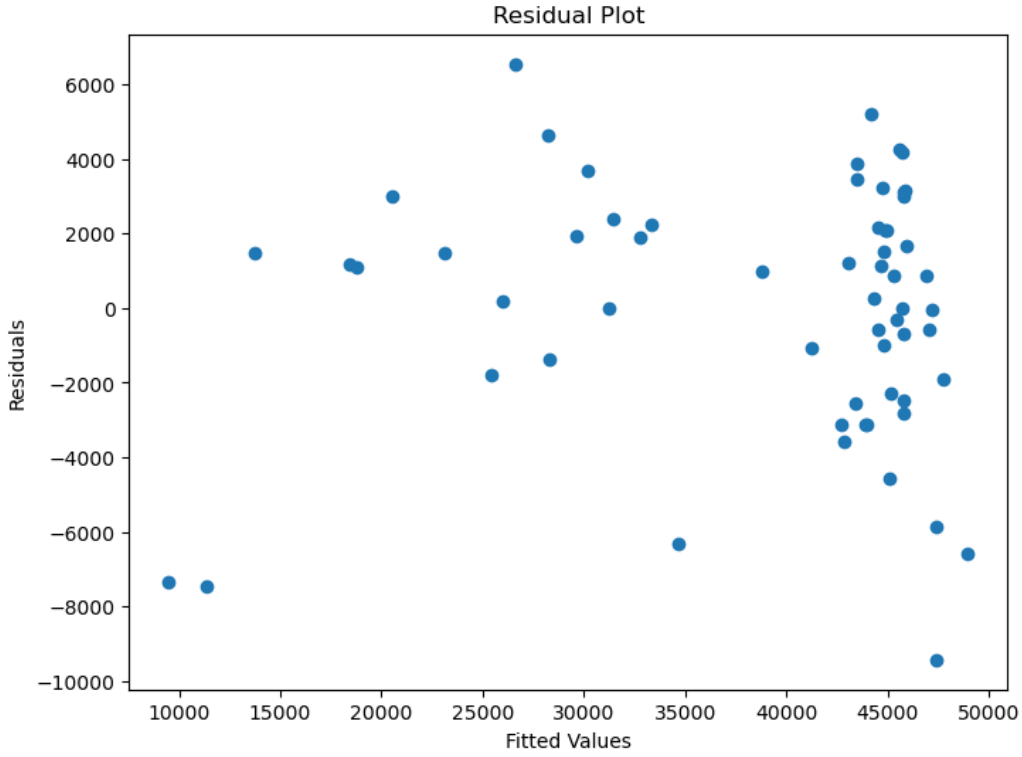
Variance Inflation Factors (VIF):

	VIF Factor	features
0	23187.431291	const
1	5.578567	Revenue
2	4.073371	Lockdown Level
3	5.660912	Confirmed Covid19 Cases
4	6.061735	Number of Covid19 Deaths
5	2.854329	Exchange Rate
6	7.501914	GDP (in billion ZAR)
7	11.819758	Yearly Population

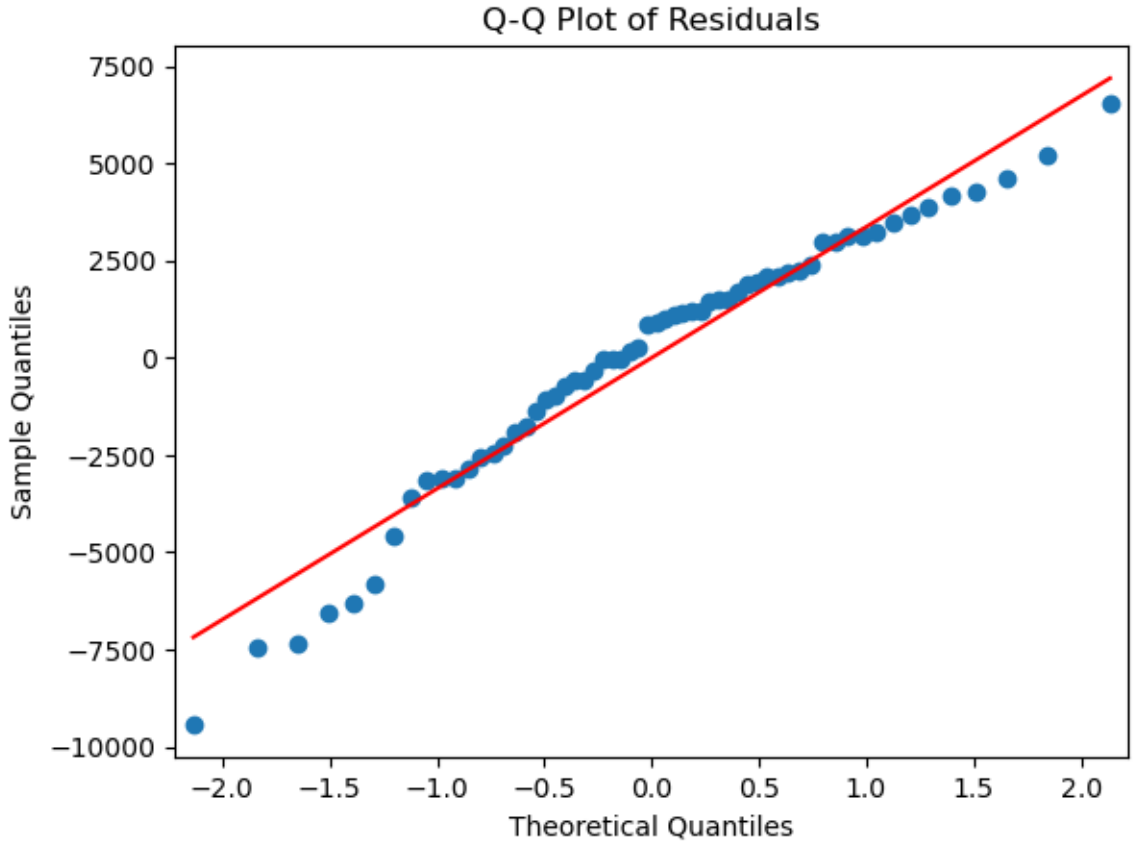
--- Independence of Errors (Durbin-Watson Test) ---

Durbin-Watson Statistic: 1.2348195219988467

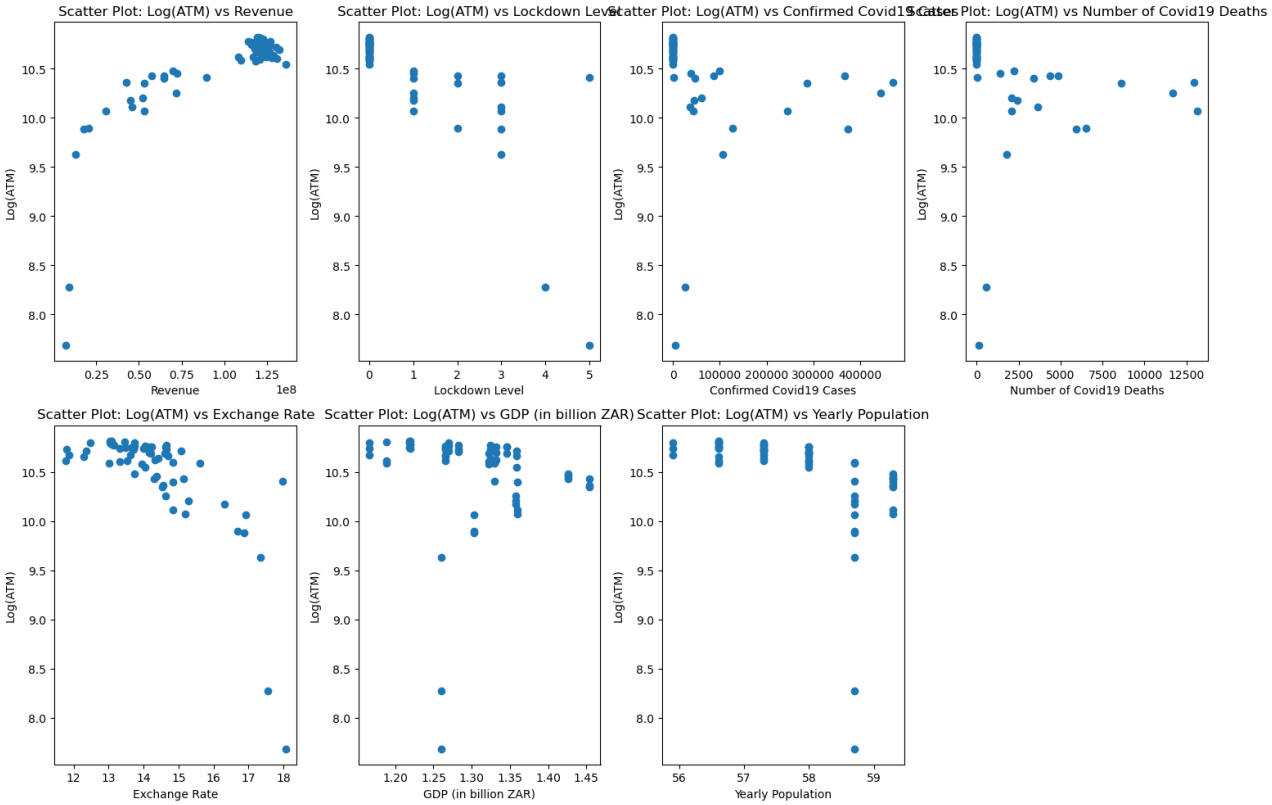
```
--- Homoscedasticity (Residual Plots) ---
```




```
--- Normality of Errors (Q-Q Plot) ---
<Figure size 800x600 with 0 Axes>
```



--- Linearity (Scatter Plots) - Log Transformed ---



--- Multicollinearity (VIF) - Log Transformed ---

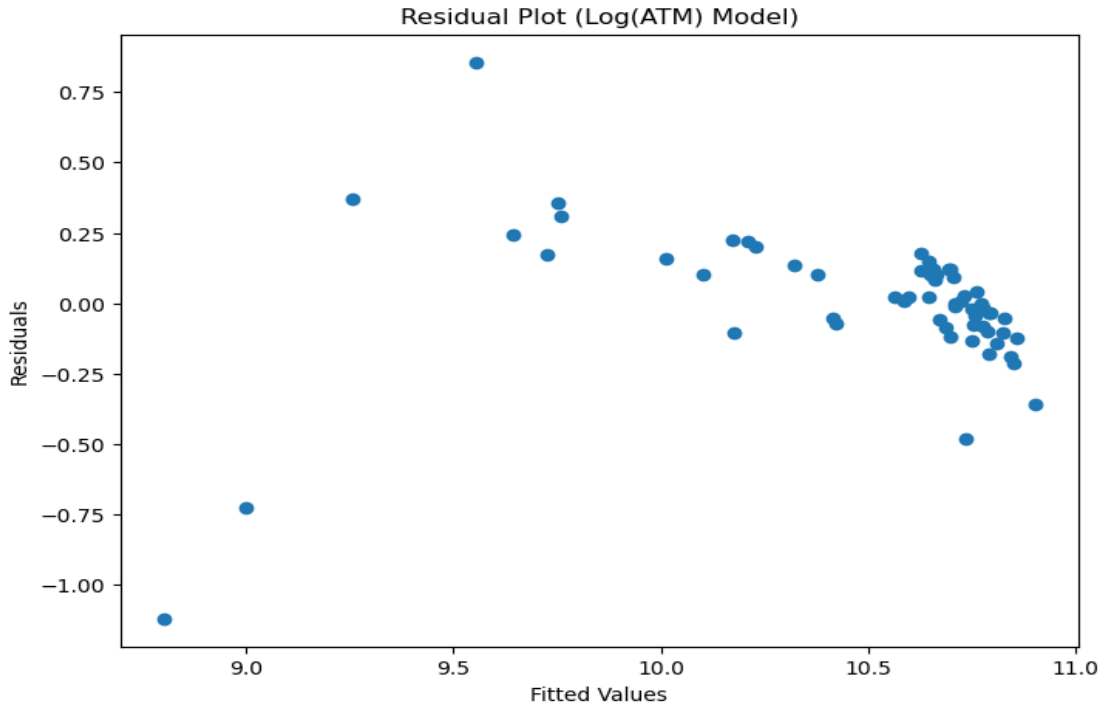
Variance Inflation Factors (VIF):

	VIF Factor	features
0	23187.431291	const
1	5.578567	Revenue
2	4.073371	Lockdown Level
3	5.660912	Confirmed Covid19 Cases
4	6.061735	Number of Covid19 Deaths
5	2.854329	Exchange Rate
6	7.501914	GDP (in billion ZAR)
7	11.819758	Yearly Population

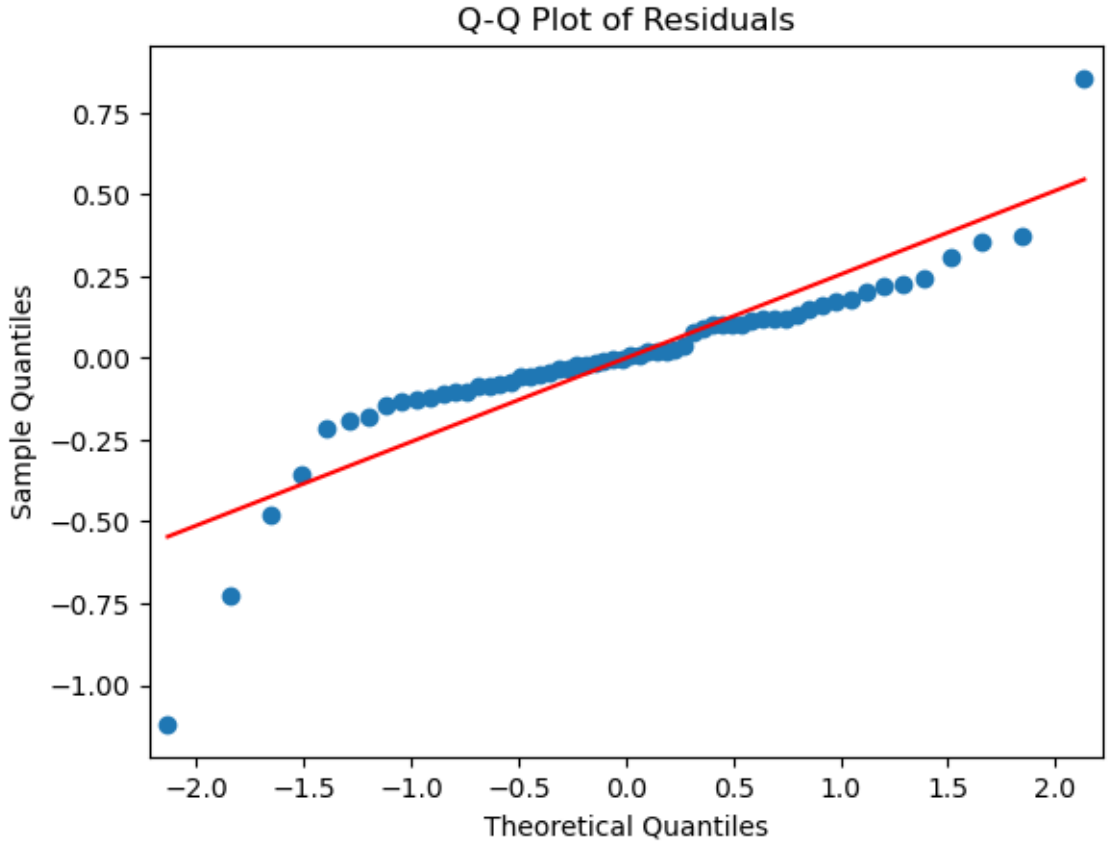
--- Independence of Errors (Durbin-Watson Test) - Log Transformed ---

Durbin-Watson Statistic: 1.8422246721860742

```
--- Homoscedasticity (Residual Plots) - Log Transformed ---
```



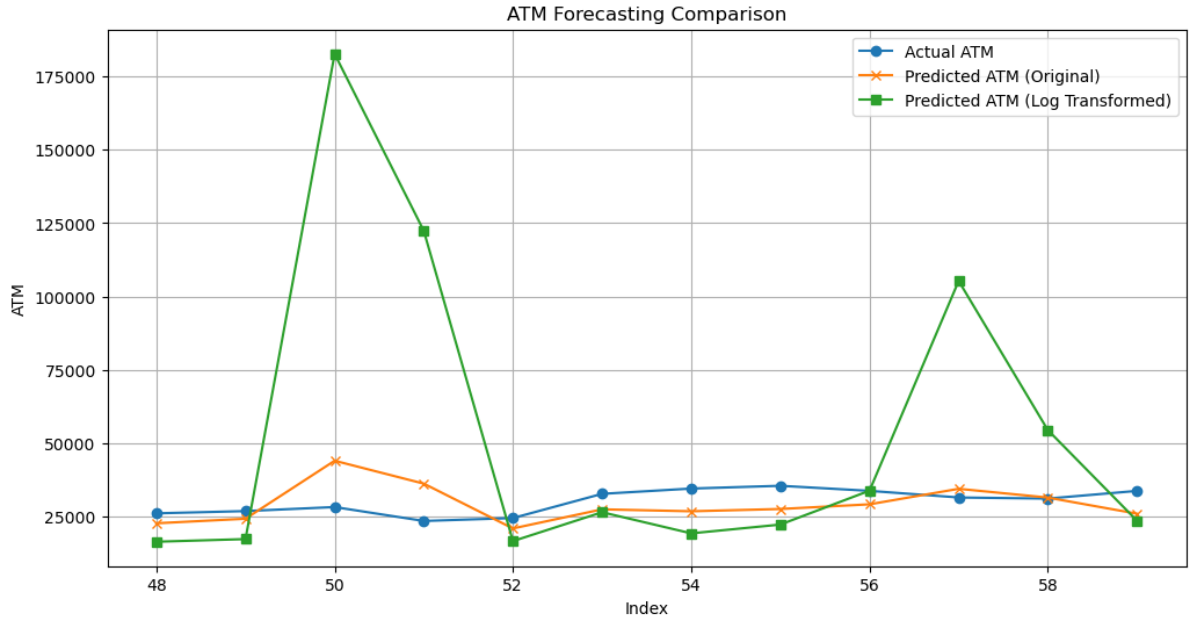

```
--- Normality of Errors (Q-Q Plot) - Log Transformed ---
<Figure size 800x600 with 0 Axes>
```



```
import pandas as pd
import numpy as np
import statsmodels.api as sm
import matplotlib.pyplot as plt

# Load the Excel file (replace 'your_file.xlsx' with the actual file name)
try:
    df = pd.read_excel(r"C:\Users\2020886178\OneDrive - UFS\Documents\My stuff\Research\Monthly ATM Data .xlsx")  # Replace with your file name
except FileNotFoundError:
    print("Error: Excel file not found. Please provide the correct path.")
    exit()
except Exception as e:
    print(f"An error occurred while loading the Excel file: {e}")
    exit()

# Data Cleaning and Preprocessing (Crucial!)
df.dropna(inplace=True)

if 'Revenue' in df.columns:
    df['Revenue'] = df['Revenue'].astype(str).str.replace(',', '', regex=False)
    df['Revenue'] = pd.to_numeric(df['Revenue'], errors='coerce')
if 'Confirmed Covid19 Cases' in df.columns:
    df['Confirmed Covid19 Cases'] = pd.to_numeric(df['Confirmed Covid19 Cases'], errors='coerce')
if 'Number of Covid19 Deaths' in df.columns:
    df['Number of Covid19 Deaths'] = pd.to_numeric(df['Number of Covid19 Deaths'], errors='coerce')
if 'GDP (in billion ZAR)' in df.columns:
    df['GDP (in billion ZAR)'] = pd.to_numeric(df['GDP (in billion ZAR)'], errors='coerce')
if 'ATM' in df.columns:
    df['ATM'] = pd.to_numeric(df['ATM'], errors='coerce')

if 'Date' in df.columns:
    df.drop('Date', axis=1, inplace=True)

# Define Independent and Dependent Variables
X = df[['Revenue', 'Lockdown Level', 'Confirmed Covid19 Cases', 'Number of Covid19 Deaths', 'Exchange Rate', 'GDP (in billion ZAR)', 'Yearly Population']]
y = df['ATM']

# Add a constant term to the independent variables
X = sm.add_constant(X)

# Split data into training and testing sets (for forecasting evaluation)
train_size = int(len(df) * 0.8)  # 80% training data
X_train, X_test = X[:train_size], X[train_size:]
y_train, y_test = y[:train_size], y[train_size:]
```

```
# --- Original Model (ATM) ---
model = sm.OLS(y_train, X_train).fit()
y_pred = model.predict(X_test)

# --- Log-Transformed Model (Log(ATM)) ---
y_train_log = np.log(y_train)
model_log = sm.OLS(y_train_log, X_train).fit()
y_pred_log = model_log.predict(X_test)

# Transform log predictions back to original scale
y_pred_log_original = np.exp(y_pred_log)

# Plotting
plt.figure(figsize=(12, 6))
plt.plot(y_test.index, y_test, label='Actual ATM', marker='o')
plt.plot(y_test.index, y_pred, label='Predicted ATM (Original)', marker='x')
plt.plot(y_test.index, y_pred_log_original, label='Predicted ATM (Log Transformed)', marker='s')
plt.xlabel('Index')
plt.ylabel('ATM')
plt.title('ATM Forecasting Comparison')
plt.legend()
plt.grid(True)
plt.show()

# --- Forecasting with new data ---
# Example new data (replace with your actual forecasting data)
new_data = {
  'Revenue': [10000000, 12000000, 11000000],
  'Lockdown Level': [2, 3, 1],
  'Confirmed Covid19 Cases': [5000, 6000, 4000],
  'Number of Covid19 Deaths': [200, 250, 180],
  'Exchange Rate': [15, 16, 14],
  'GDP (in billion ZAR)': [1.3, 1.4, 1.2],
  'Yearly Population': [58, 59, 57]
}

new_df = pd.DataFrame(new_data)
new_X = sm.add_constant(new_df)

# Forecast using the original model
forecast_original = model.predict(new_X)

# Forecast using the log-transformed model
forecast_log = model_log.predict(new_X)
forecast_log_original = np.exp(forecast_log)

print("\nForecasted ATM (Original Model):")
```

```
print(forecast_original)

print("\nForecasted ATM (Log-Transformed Model):")
print(forecast_log_original)
```




```
Forecasted ATM (Original Model):
0    11572.071390
1     8317.163995
2    15662.613677
dtype: float64

Forecasted ATM (Log-Transformed Model):
0    5652.085911
1    4844.491536
2    6982.564541
dtype: float64
import pandas as pd
import numpy as np
import statsmodels.api as sm
from sklearn.metrics import mean_squared_error

# Load the Excel file
try:
    df = pd.read_excel(r"C:\Users\2020886178\OneDrive - UFS\Documents\My stuff\Research\Monthly ATM Data .xlsx")  # Replace with your file name
except FileNotFoundError:
    print("Error: Excel file not found. Please provide the correct path.")
    exit()
except Exception as e:
    print(f"An error occurred while loading the Excel file: {e}")
```

```
    exit()

# Data Cleaning and Preprocessing (Crucial!)
df.dropna(inplace=True)

if 'Revenue' in df.columns:
    df['Revenue'] = df['Revenue'].astype(str).str.replace(',', '', regex=False)
    df['Revenue'] = pd.to_numeric(df['Revenue'], errors='coerce')
if 'Confirmed Covid19 Cases' in df.columns:
    df['Confirmed Covid19 Cases'] = pd.to_numeric(df['Confirmed Covid19 Cases'], errors='coerce')
if 'Number of Covid19 Deaths' in df.columns:
    df['Number of Covid19 Deaths'] = pd.to_numeric(df['Number of Covid19 Deaths'], errors='coerce')
if 'GDP (in billion ZAR)' in df.columns:
    df['GDP (in billion ZAR)'] = pd.to_numeric(df['GDP (in billion ZAR)'], errors='coerce')
if 'ATM' in df.columns:
    df['ATM'] = pd.to_numeric(df['ATM'], errors='coerce')

if 'Date' in df.columns:
    df.drop('Date', axis=1, inplace=True)

# Define Independent and Dependent Variables
X = df[['Revenue', 'Lockdown Level', 'Confirmed Covid19 Cases', 'Number of Covid19 Deaths', 'Exchange R
ate', 'GDP (in billion ZAR)', 'Yearly Population']]
y = df['ATM']

# Add a constant term to the independent variables
X = sm.add_constant(X)

# Split data into training and testing sets (for forecasting evaluation)
train_size = int(len(df) * 0.8)  # 80% training data
X_train, X_test = X[:train_size], X[train_size:]
y_train, y_test = y[:train_size], y[train_size:]

# --- Original Model (ATM) ---
model = sm.OLS(y_train, X_train).fit()
y_pred = model.predict(X_test)

# --- Log-Transformed Model (Log(ATM)) ---
y_train_log = np.log(y_train)
model_log = sm.OLS(y_train_log, X_train).fit()
y_pred_log = model_log.predict(X_test)
y_pred_log_original = np.exp(y_pred_log)

# Calculate MSE for the original model
mse_original = mean_squared_error(y_test, y_pred)
print(f"Mean Squared Error (Original Model): {mse_original}")
```

```
# Calculate MSE for the log-transformed model
mse_log_transformed = mean_squared_error(y_test, y_pred_log_original)
print(f"Mean Squared Error (Log-Transformed Model): {mse_log_transformed}")
```

```
Mean Squared Error (Original Model): 56761595.481769055
Mean Squared Error (Log-Transformed Model): 3360670003.4328556
```

In [2]:

```
import pandas as pd
import numpy as np
import statsmodels.api as sm
from sklearn.metrics import mean_squared_error

# Load the Excel file
try:
    df = pd.read_excel(r"C:\Users\2020886178\OneDrive - UFS\Documents\My stuff\Research\Monthly ATM Data .xlsx")  # Replace with file name
except FileNotFoundError:
    print("Error: Excel file not found. Please provide the correct path.")
    exit()
except Exception as e:
    print(f"An error occurred while loading the Excel file: {e}")
    exit()

# Data Cleaning and Preprocessing
df.dropna(inplace=True)

if 'Revenue' in df.columns:
    df['Revenue'] = df['Revenue'].astype(str).str.replace(',', '', regex=False)
    df['Revenue'] = pd.to_numeric(df['Revenue'], errors='coerce')
if 'Confirmed Covid19 Cases' in df.columns:
    df['Confirmed Covid19 Cases'] = pd.to_numeric(df['Confirmed Covid19 Cases'], errors='coerce')
if 'Number of Covid19 Deaths' in df.columns:
    df['Number of Covid19 Deaths'] = pd.to_numeric(df['Number of Covid19 Deaths'], errors='coerce')
if 'GDP (in billion ZAR)' in df.columns:
    df['GDP (in billion ZAR)'] = pd.to_numeric(df['GDP (in billion ZAR)'], errors='coerce')
if 'ATM' in df.columns:
    df['ATM'] = pd.to_numeric(df['ATM'], errors='coerce')

if 'Date' in df.columns:
    df.drop('Date', axis=1, inplace=True)

# Define Independent and Dependent Variables
X = df[['Revenue', 'Lockdown Level', 'Confirmed Covid19 Cases', 'Number of Covid19 Deaths', 'Exchange Rate', 'GDP (in billion ZAR)', 'Yearly Population']]
y = df['ATM']

# Add a constant term to the independent variables
X = sm.add_constant(X)
```

```
# Split data into training and testing sets (for forecasting evaluation)
train_size = int(len(df) * 0.8)  # 80% training data
X_train, X_test = X[:train_size], X[train_size:]
y_train, y_test = y[:train_size], y[train_size:]

# --- Original Model (ATM) ---
model = sm.OLS(y_train, X_train).fit()
y_pred = model.predict(X_test)

# --- Log-Transformed Model (Log(ATM)) ---
y_train_log = np.log(y_train)
model_log = sm.OLS(y_train_log, X_train).fit()
y_pred_log = model_log.predict(X_test)
y_pred_log_original = np.exp(y_pred_log)

# Calculate MSE for the original model
mse_original = mean_squared_error(y_test, y_pred)

# Calculate MSE for the log-transformed model
mse_log_transformed = mean_squared_error(y_test, y_pred_log_original)

# Calculate RMSE for the original model
rmse_original = np.sqrt(mse_original)
print(f"Root Mean Squared Error (Original Model): {rmse_original}")

# Calculate RMSE for the log-transformed model
rmse_log_transformed = np.sqrt(mse_log_transformed)
print(f"Root Mean Squared Error (Log-Transformed Model): {rmse_log_transformed}")
Root Mean Squared Error (Original Model): 7534.029166506396
Root Mean Squared Error (Log-Transformed Model): 57971.28602534927
import pandas as pd
import numpy as np
import statsmodels.api as sm
from sklearn.metrics import mean_squared_error

# Load the Excel file (replace 'your_file.xlsx' with the actual file name)
try:
    df = pd.read_excel(r"C:\Users\2020886178\OneDrive - UFS\Documents\My stuff\Research\Monthly ATM Data .xlsx")
except FileNotFoundError:
    print("Error: Excel file not found. Please provide the correct path.")
    exit()
except Exception as e:
    print(f"An error occurred while loading the Excel file: {e}")
    exit()
```

```
# Data Cleaning and Preprocessing
df.dropna(inplace=True)

if 'Revenue' in df.columns:
    df['Revenue'] = df['Revenue'].astype(str).str.replace(',', '', regex=False)
    df['Revenue'] = pd.to_numeric(df['Revenue'], errors='coerce')
if 'Confirmed Covid19 Cases' in df.columns:
    df['Confirmed Covid19 Cases'] = pd.to_numeric(df['Confirmed Covid19 Cases'], errors='coerce')
if 'Number of Covid19 Deaths' in df.columns:
    df['Number of Covid19 Deaths'] = pd.to_numeric(df['Number of Covid19 Deaths'], errors='coerce')
if 'GDP (in billion ZAR)' in df.columns:
    df['GDP (in billion ZAR)'] = pd.to_numeric(df['GDP (in billion ZAR)'], errors='coerce')
if 'ATM' in df.columns:
    df['ATM'] = pd.to_numeric(df['ATM'], errors='coerce')

if 'Date' in df.columns:
    df.drop('Date', axis=1, inplace=True)

# Define Independent and Dependent Variables
X = df[['Revenue', 'Lockdown Level', 'Confirmed Covid19 Cases', 'Number of Covid19 Deaths', 'Exchange R
ate', 'GDP (in billion ZAR)', 'Yearly Population']]
y = df['ATM']

# Add a constant term to the independent variables
X = sm.add_constant(X)

# Stepwise Regression Function
def stepwise_regression(X, y, threshold_in=0.05, threshold_out=0.10, verbose=True):
    included = []
    while True:
        changed = False

        # Forward Step
        excluded = list(set(X.columns) - set(included))
        new_pval = pd.Series(index=excluded)
        for new_column in excluded:
            model = sm.OLS(y, sm.add_constant(X[included + [new_column]])).fit()
            new_pval[new_column] = model.pvalues[new_column]
        min_pval = new_pval.min()
        if min_pval < threshold_in:
            new_column = new_pval.idxmin()
            included.append(new_column)
            changed = True
            if verbose:
                print(f'Add {new_column}, p-value = {min_pval}')

        # Backward Step
```

```
    if len(included) > 1:
        model = sm.OLS(y, sm.add_constant(X[included])).fit()
        pvalues = model.pvalues.drop('const')
        max_pval = pvalues.max()
        if max_pval > threshold_out:
            removed_column = pvalues.idxmax()
            included.remove(removed_column)
            changed = True
            if verbose:
                print(f'Remove {removed_column}, p-value = {max_pval}')

    if not changed:
        break
  return included

# Perform Stepwise Regression
influential_predictors = stepwise_regression(X, y)

print("\nInfluential Predictors Identified by Stepwise Regression:")
print(influential_predictors)

# Refit the model with only influential predictors
X_selected = X[['const'] + influential_predictors]
final_model = sm.OLS(y, X_selected).fit()

print("\nFinal Model Summary:")
print(final_model.summary())

# Evaluate the final model (e.g., calculate MSE or RMSE on a test set)
# ... (Add code for model evaluation)
```

```
Add const, p-value = 3.5194051453941644e-35
Add Revenue, p-value = 1.964801069767585e-25
Add Exchange Rate, p-value = 0.0003084966014873724

Influential Predictors Identified by Stepwise Regression:
['const', 'Revenue', 'Exchange Rate']

Final Model Summary:
                            OLS Regression Results
==============================================================================
===
Dep. Variable:                    ATM   R-squared:                       0.
880
Model:                            OLS   Adj. R-squared:                  0.
875
Method:                 Least Squares   F-statistic:                     20
8.2
```

```
Date:                Thu, 20 Feb 2025   Prob (F-statistic):           6.29e
-27
Time:                        21:59:39   Log-Likelihood:                -579
.22
No. Observations:                60   AIC:                             11
64.
Df Residuals:                    57   BIC:                             11
71.
Df Model:                         2
Covariance Type:            nonrobust
==============================================================================
======
                     coef    std err          t      P>|t|      [0.025
0.975]
------------------------------------------------------------------------------
------
const          2.235e+04   4221.577      5.294      0.000    1.39e+04    3.
08e+04
const          2.235e+04   4221.577      5.294      0.000    1.39e+04    3.
08e+04
Revenue           0.0002   1.86e-05     11.555      0.000       0.000
0.000
Exchange Rate -1889.5607    491.732     -3.843      0.000   -2874.237    -9
04.884
==============================================================================
===
Omnibus:                        7.248   Durbin-Watson:                   0.
940
Prob(Omnibus):                  0.027   Jarque-Bera (JB):                7.
237
Skew:                          -0.851   Prob(JB):                       0.0
268
Kurtosis:                       2.984   Cond. No.                     5.82e
+18
==============================================================================
===

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is corr
ectly specified.
[2] The smallest eigenvalue is 1.95e-20. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.
```

In [2]:

```
import pandas as pd
import numpy as np
import statsmodels.api as sm
from sklearn.metrics import mean_squared_error
```

```
# Load the Excel file (replace 'your_file.xlsx' with the actual file name)
try:
    df = pd.read_excel(r"C:\Users\2020886178\OneDrive - UFS\Documents\My stuff\Research\Monthly ATM Data .xlsx")  # Replace with your file name
except FileNotFoundError:
    print("Error: Excel file not found. Please provide the correct path.")
    exit()
except Exception as e:
    print(f"An error occurred while loading the Excel file: {e}")
    exit()

# Data Cleaning and Preprocessing (Crucial!)
df.dropna(inplace=True)

if 'Revenue' in df.columns:
    df['Revenue'] = df['Revenue'].astype(str).str.replace(',', '', regex=False)
    df['Revenue'] = pd.to_numeric(df['Revenue'], errors='coerce')
if 'Confirmed Covid19 Cases' in df.columns:
    df['Confirmed Covid19 Cases'] = pd.to_numeric(df['Confirmed Covid19 Cases'], errors='coerce')
if 'Number of Covid19 Deaths' in df.columns:
    df['Number of Covid19 Deaths'] = pd.to_numeric(df['Number of Covid19 Deaths'], errors='coerce')
if 'GDP (in billion ZAR)' in df.columns:
    df['GDP (in billion ZAR)'] = pd.to_numeric(df['GDP (in billion ZAR)'], errors='coerce')
if 'ATM' in df.columns:
    df['ATM'] = pd.to_numeric(df['ATM'], errors='coerce')

if 'Date' in df.columns:
    df.drop('Date', axis=1, inplace=True)

# Define Independent and Dependent Variables
X = df[['Revenue', 'Lockdown Level', 'Confirmed Covid19 Cases', 'Number of Covid19 Deaths', 'Exchange Rate', 'GDP (in billion ZAR)', 'Yearly Population']]
y = df['ATM']

# Add a constant term to the independent variables
X = sm.add_constant(X)

# Split data into training and testing sets (for forecasting evaluation)
train_size = int(len(df) * 0.8)  # 80% training data
X_train, X_test = X[:train_size], X[train_size:]
y_train, y_test = y[:train_size], y[train_size:]

# Forward Stepwise Regression Function
def forward_stepwise_regression(X, y, threshold_in=0.05, verbose=True):
    included = []
    remaining = list(X.columns[1:])  # Start with all predictors except the constant
```

```
    while remaining:
        best_pval = 1  # Initialize with a value > threshold_in
        best_predictor = None

        for predictor in remaining:
            current_predictors = included + [predictor]
            X_subset = sm.add_constant(X_train[current_predictors]) # Use training data
            model = sm.OLS(y_train, X_subset).fit()
            pval = model.pvalues[predictor]

            if pval < best_pval:
                best_pval = pval
                best_predictor = predictor

        if best_pval < threshold_in:
            included.append(best_predictor)
            remaining.remove(best_predictor)
            if verbose:
                print(f"Added {best_predictor}, p-value = {best_pval}")
        else:
            break  # No predictor meets the criteria

    return included

# Perform Forward Stepwise Regression
influential_predictors = forward_stepwise_regression(X, y)

print("\nInfluential Predictors Identified by Forward Stepwise Regression:")
print(influential_predictors)

# Refit the model with only influential predictors (on the entire dataset)
X_selected = X[['const'] + influential_predictors]
final_model = sm.OLS(y, X_selected).fit()

print("\nFinal Model Summary:")
print(final_model.summary())

# Evaluate the final model (e.g., calculate MSE or RMSE on a test set)
y_pred_final = final_model.predict(X_test) # Predict on test set

mse_final = mean_squared_error(y_test, y_pred_final)
rmse_final = np.sqrt(mse_final)

print(f"MSE of the final model: {mse_final}")
print(f"RMSE of the final model: {rmse_final}")
Added Revenue, p-value = 4.672705106357833e-21
```

```
Added Yearly Population, p-value = 0.00017976344247292459
Added Number of Covid19 Deaths, p-value = 0.0012133370199934276

Influential Predictors Identified by Forward Stepwise Regression:
['Revenue', 'Yearly Population', 'Number of Covid19 Deaths']

Final Model Summary:
                            OLS Regression Results
==============================================================================
Dep. Variable:                    ATM   R-squared:                       0.871
Model:                            OLS   Adj. R-squared:                  0.864
Method:                 Least Squares   F-statistic:                     125.5
Date:                Thu, 20 Feb 2025   Prob (F-statistic):           7.78e-25
Time:                        22:03:17   Log-Likelihood:                -581.39
No. Observations:                  60   AIC:                             1171.
Df Residuals:                      56   BIC:                             1179.
Df Model:                           3
Covariance Type:            nonrobust
==============================================================================
                                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const                      1.148e+05   4.28e+04      2.683      0.010    2.91e+04       2e+05
Revenue                       0.0003   1.98e-05     13.184      0.000       0.000       0.000
Yearly Population         -1773.3067    724.612     -2.447      0.018   -3224.878    -321.735
Number of Covid19 Deaths      0.5560      0.218      2.548      0.014       0.119       0.993
==============================================================================
Omnibus:                        6.424   Durbin-Watson:                   0.751
Prob(Omnibus):                  0.040   Jarque-Bera (JB):                5.965
```

```
Skew:                          -0.768   Prob(JB):                         0.0
507
Kurtosis:                       3.157   Cond. No.                      8.59e
+09
==============================================================================
===

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is corr
ectly specified.
[2] The condition number is large, 8.59e+09. This might indicate that there
are
strong multicollinearity or other numerical problems.
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[2], line 89
     86 print(final_model.summary())
     88 # Evaluate the final model (e.g., calculate MSE or RMSE on a test s
et)
---> 89 y_pred_final = final_model.predict(X_test) # Predict on test set
     91 mse_final = mean_squared_error(y_test, y_pred_final)
     92 rmse_final = np.sqrt(mse_final)

File ~\AppData\Local\anaconda3\Lib\site-packages\statsmodels\base\model.py:
1176, in Results.predict(self, exog, transform, *args, **kwargs)
   1129 """
   1130 Call self.model.predict with self.params as the first argument.
   1131
   (...)
   1171 returned prediction.
   1172 """
   1173 exog, exog_index = self._transform_predict_exog(exog,
   1174                                                  transform=transform
)
-> 1176 predict_results = self.model.predict(self.params, exog, *args,
   1177                                       **kwargs)
   1179 if exog_index is not None and not hasattr(predict_results,
   1180                                           'predicted_values'):
   1181     if predict_results.ndim == 1:

File ~\AppData\Local\anaconda3\Lib\site-packages\statsmodels\regression\lin
ear_model.py:411, in RegressionModel.predict(self, params, exog)
    408 if exog is None:
    409     exog = self.exog
--> 411 return np.dot(exog, params)

File <__array_function__ internals>:200, in dot(*args, **kwargs)
```

```
ValueError: shapes (12,8) and (4,) not aligned: 8 (dim 1) != 4 (dim 0)
```

```
# Import libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import chi2
from sklearn.model_selection import train_test_split
from sklearn.linear_model import MultipleRegression
from sklearn.metrics import classification_report, confusion_matrix, roc_auc_score, roc_curve, log_loss
from sklearn.model_selection import cross_val_score
from sklearn.feature_selection import RFECV
from sklearn.impute import KNNImputer
from sklearn.preprocessing import StandardScaler

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
import seaborn as sns
from scipy import stats

# Load dataset
data = pd.read_excel(r"C:\Users\20208886178\OneDrive - JohnMasekoameng\Desktop\Monthly ATM D
ata .xlsx")
data = data.drop(['Date'],axis=1)
data.head()
```

	Revenue	ATM	Lockdown Level	COVID-19 Cases	COVID-19 Deaths	Exchange Rate	GDP (in billion ZAR)	Yearly Population
0	1.193800e+08	48799	0	0	0	13.73	1.166	55.9
1	1.178159e+08	46171	0	0	0	13.99	1.166	55.9
2	1.188397e+08	42975	0	0	0	13.62	1.166	55.9
3	1.167916e+08	40873	0	0	0	13.52	1.189	56.6

	Revenue	ATM	Lockdown Level	COVID-19 Cases	COVID-19 Deaths	Exchange Rate	GDP (in billion ZAR)	Yearly Population
4	1.095519e+08	39561	0	0	0	13.01	1.189	56.6

```
# Drop missing values if needed (assuming 'Air Traffic Movement' is the main column)
data.dropna(subset=['ATM'], inplace=True)

# Log transformation
data['Log ATM'] = np.log(data['ATM'] + 1e-6)
In [16]:
# Assuming 'df' is the DataFrame and 'Log Air Traffic Movement' is the transformed column
y = data['ATM']
X = data.drop(columns=['ATM', 'Log ATM'])

# Splitting the data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
import seaborn as sns
from scipy import stats
# Load datas
# Drop missing values if needed (assuming 'ATM' is the main column)
data.dropna(subset=['ATM'], inplace=True)

# Log transformation
data['Log ATM'] = np.log(data['ATM'] + 1e-6)

# Assuming 'df' is the DataFrame and 'Log ATM' is the transformed column
y = data['ATM']
X = data.drop(columns=['ATM', 'Log ATM'])

# Splitting the data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Initialise and fit Linear Regression model
model = LinearRegression()
model.fit(X_train, y_train)

# Predict on the test set
y_pred = model.predict(X_test)
```

```
# Calculate the coefficients and intercept
slope = model.coef_
intercept = model.intercept_

# Check linearity assumption based on model performance metrics, like R-squared
r_squared = model.score(X_test, y_test)

# Check linearity assumption using R-squared
if r_squared < 0.05:
   print("Data suggests non-linearity in the relationship.")
else:
   print("Data supports the linearity assumption.")
Data supports the linearity assumption.
In [18]:
from statsmodels.stats.outliers_influence import variance_inflation_factor

# Calculate VIF for each feature in 'X_train' to detect multicollinearity
vif_data = pd.DataFrame()
vif_data["feature"] = X_train.columns
vif_data["VIF"] = [variance_inflation_factor(X_train.values, i) for i in range(len(X_train.columns))]

# Check for VIF values exceeding 10 (indicating multicollinearity)
high_vif = vif_data[vif_data['VIF'] > 10]
if high_vif.empty:
   print("No multicollinearity issue detected.")
else:
   print("Multicollinearity detected in the following features:")
   print(high_vif)
Multicollinearity detected in the following features:
            feature         VIF
0             Revenue    34.675448
4       Exchange Rate   288.074755
5  GDP (in billion ZAR)   892.164554
6    Yearly Population  1030.157408
In [19]:
# Calculate residuals
residuals = y_test - y_pred

# Test for homoscedasticity (patterns in residuals)
plt.scatter(y_pred, residuals)
plt.xlabel("Fitted values")
plt.ylabel("Residuals")
plt.title("Residuals vs Fitted values")
plt.show()

In [20]:
# Check normality of residuals using QQ plot
```

```
stats.probplot(residuals, dist="norm", plot=plt)
plt.title('Q-Q Plot of Residuals')
plt.show()

In [21]:
# Initialise and fit the regression model
model = LinearRegression()
model.fit(X_train, y_train)

# Predict on the test data
y_pred = model.predict(X_test)
In [22]:
# Calculate the accuracy of the model
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

# Output the model accuracy metrics
print(f"Mean Squared Error (MSE): {mse}")
print(f"R-squared (R2): {r2}")
Mean Squared Error (MSE): 12815762.08360573
R-squared (R2): 0.8954998413364453
In [11]:
# Plotting the distribution with histogram and kernel density estimate (KDE)
plt.figure(figsize=(10, 6))
sns.histplot(y, kde=True)
plt.title('Distribution of Log Air Traffic Movement')
plt.xlabel('Log Air Traffic Movement')
plt.ylabel('Frequency')

# Fit a normal distribution to the data and plot it
mean, std = stats.norm.fit(y)
xmin, xmax = plt.xlim()
x = np.linspace(xmin, xmax, 100)
p = stats.norm.pdf(x, mean, std)
scale_factor = np.diff(plt.gca().transData.transform([(0,0),(0,1)])[1])[0] * len(y) * np.diff(plt.xlim())[0] / 1
00
plt.plot(x, p * scale_factor, 'k', linewidth=2)

# Show the histogram with the normal distribution curve
plt.show()

In [12]:
# Q-Q plot
plt.figure(figsize=(10, 6))
stats.probplot(y, dist="norm", plot=plt)
plt.title('Q-Q Plot of Log Air Traffic Movement')
```

```
# Show the Q-Q plot
plt.show()

. # Import libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import chi2
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix, roc_auc_score, roc_curve, log_loss
from sklearn.model_selection import cross_val_score
from sklearn.feature_selection import RFECV
from sklearn.impute import KNNImputer
from sklearn.preprocessing import StandardScaler

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
import seaborn as sns
from scipy import stats

# Load dataset
data = pd.read_excel(r"C:\Users\2020886178\OneDrive - UFS\Desktop\Monthly ATM Data .xlsx")
data = data.drop(['Date'],axis=1)
data.head()

# Drop missing values if needed (assuming 'Air Traffic Movement' is the main column)
data.dropna(subset=['ATM'], inplace=True)

# Log transformation
data['Log ATM'] = np.log(data['ATM'] + 1e-6)


# Assuming 'df' is the DataFrame and 'Log Air Traffic Movement' is the transformed column
y = data['Log ATM']
X = data.drop(columns=['ATM', 'Log ATM'])

# Splitting the data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
import seaborn as sns
from scipy import stats

# Load dataset
data = pd.read_excel(r"C:\Users\2020886178\OneDrive - UFS\Desktop\Monthly ATM Data .xlsx")
data = data.drop(['Date'], axis=1)
data.head()

# Drop missing values if needed (assuming 'ATM' is the main column)
data.dropna(subset=['ATM'], inplace=True)

# Log transformation
data['Log ATM'] = np.log(data['ATM'] + 1e-6)

# Assuming 'df' is the DataFrame and 'Log ATM' is the transformed column
y = data['Log ATM']
X = data.drop(columns=['ATM', 'Log ATM'])

# Splitting the data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Initialise and fit Linear Regression model
model = LinearRegression()
model.fit(X_train, y_train)

# Predict on the test set
y_pred = model.predict(X_test)

# Calculate the coefficients and intercept
slope = model.coef_
intercept = model.intercept_

# Check linearity assumption based on model performance metrics, like R-squared
r_squared = model.score(X_test, y_test)

# Check linearity assumption using R-squared
if r_squared < 0.05:
    print("Data suggests non-linearity in the relationship.")
else:
    print("Data supports the linearity assumption.")


from statsmodels.stats.outliers_influence import variance_inflation_factor
```

```
# Calculate VIF for each feature in 'X_train' to detect multicollinearity
vif_data = pd.DataFrame()
vif_data["feature"] = X_train.columns
vif_data["VIF"] = [variance_inflation_factor(X_train.values, i) for i in range(len(X_train.columns))]

# Check for VIF values exceeding 10 (indicating multicollinearity)
high_vif = vif_data[vif_data['VIF'] > 10]
if high_vif.empty:
    print("No multicollinearity issue detected.")
else:
    print("Multicollinearity detected in the following features:")
    print(high_vif)


# Calculate residuals
residuals = y_test - y_pred

# Test for homoscedasticity (patterns in residuals)
plt.scatter(y_pred, residuals)
plt.xlabel("Fitted values")
plt.ylabel("Residuals")
plt.title("Residuals vs Fitted values")
plt.show()

# Check normality of residuals using QQ plot
stats.probplot(residuals, dist="norm", plot=plt)
plt.title('Q-Q Plot of Residuals')
plt.show()


# Initialise and fit the regression model
model = LinearRegression()
model.fit(X_train, y_train)

# Predict on the test data
y_pred = model.predict(X_test)


# Calculate the accuracy of the model
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

# Output the model accuracy metrics
print(f"Mean Squared Error (MSE): {mse}")
print(f"R-squared (R2): {r2}")
```

```
# Plotting the distribution with histogram and kernel density estimate (KDE)
plt.figure(figsize=(10, 6))
sns.histplot(y, kde=True)
plt.title('Distribution of Log Air Traffic Movement')
plt.xlabel('Log Air Traffic Movement')
plt.ylabel('Frequency')

# Fit a normal distribution to the data and plot it
mean, std = stats.norm.fit(y)
xmin, xmax = plt.xlim()
x = np.linspace(xmin, xmax, 100)
p = stats.norm.pdf(x, mean, std)
scale_factor = np.diff(plt.gca().transData.transform([(0,0),(0,1)])[1])[0] * len(y) * np.diff(plt.xlim())[0] / 1
00
plt.plot(x, p * scale_factor, 'k', linewidth=2)

# Show the histogram with the normal distribution curve
plt.show()


# Q-Q plot
plt.figure(figsize=(10, 6))
stats.probplot(y, dist="norm", plot=plt)
plt.title('Q-Q Plot of Log Air Traffic Movement')

# Show the Q-Q plot
plt.show()

import pandas as pd
from sklearn.linear_model import LinearRegression

# Replace 'Dependent Variable' with actual data for 'ATM' (dependent variable)
data = {
   'Date': [0],
   'Revenue': [0],
   'ATM': [0],  # Replace with  actual data for 'ATM'
   'Lockdown Level': [0],
   'Covid19 Cases': [0],
   'Covid19 Deaths': [0],
   'Exchange Rate': [0],
   'GDP (in billion ZAR)': [0],
   'Yearly Population': [0]
}

# Create a DataFrame
df = pd.DataFrame(data)
```

```
# Split the data into predictors (independent variables) and the target variable
X = df.drop(columns=['ATM'])  # Independent variables
y = df['ATM']  # Dependent variable

# Initialise the Linear Regression model
model = LinearRegression()

# Fit the model
model.fit(X, y)

# Coefficients of the model
coefficients = model.coef_

# Intercept of the model
intercept = model.intercept_

print("Coefficients:", coefficients)
print("Intercept:", intercept)

# Import libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import chi2
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix, roc_auc_score, roc_curve, log_loss
from sklearn.model_selection import cross_val_score
from sklearn.feature_selection import RFECV
from sklearn.impute import KNNImputer
from sklearn.preprocessing import StandardScaler

# Load dataset
data = pd.read_excel("C:/Users/2020886178/OneDrive/Desktop/documents/johnm/Yearly  and Monthl
y Data.xlsx")
data.head()

# Summarise the count of remaining missing values (if any)
print(data.isnull().sum())

df = pd.DataFrame(data)

# Log transformation of the 'Air Traffic Movements' column
# Adding a small constant to avoid taking log of zero if zero values are present
df['Log Air Traffic Movements'] = np.log(df['Air Traffic Movements'] + 1e-6)

# Display the DataFrame to verify the transformation
```

```
print(df[['Air Traffic Movements', 'Log Air Traffic Movements']])

from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, r2_score

# Assuming 'df' is the DataFrame after log transformation as created previously
# and that 'Date' column has been converted to a numerical feature if used in regression

# For simplicity, let's assume 'Date' is not used in the regression model
X = df.drop(columns=['Date', 'Air Traffic Movements', 'Log Air Traffic Movements'])
y = df['Log Air Traffic Movements']

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Initialise and fit the regression model
model = LinearRegression()
model.fit(X_train, y_train)

# Predict on the test data
y_pred = model.predict(X_test)

# Calculate the accuracy of the model
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

# Output the model accuracy metrics
print(f"Mean Squared Error (MSE): {mse}")
print(f"R-squared (R2): {r2}")

import matplotlib.pyplot as plt

# Assuming 'y' is the actual values and 'y_pred_full' is the predictions for the entire dataset
# The index for X will serve as the x-axis for the plot

# Generate predictions for the entire dataset for plotting purposes
y_pred_full = model.predict(X)

# Create a scatter plot of the actual values
plt.scatter(X.index, y, color='blue', label='Actual')

# Plot the predicted values
plt.plot(X.index, y_pred_full, color='red', label='Predicted', linewidth=2)

# Add title and labels to the plot
plt.title('Actual vs Predicted Log Air Traffic Movements')
```

```
plt.xlabel('Index')
plt.ylabel('Log Air Traffic Movements')

# Show the legend
plt.legend()

# Show the plot
plt.show()

import seaborn as sns
import matplotlib.pyplot as plt
from scipy import stats
import numpy as np

# Assuming 'df' is the DataFrame and 'Log Air Traffic Movements' is the transformed column
y = df['Log Air Traffic Movements']

# Plotting the distribution with histogram and kernel density estimate (KDE)
plt.figure(figsize=(10, 6))
sns.histplot(y, kde=True)
plt.title('Distribution of Log Air Traffic Movements')
plt.xlabel('Log Air Traffic Movements')
plt.ylabel('Frequency')

# Fit a normal distribution to the data and plot it
mean, std = stats.norm.fit(y)  # Get the mean and standard deviation
xmin, xmax = plt.xlim()  # Get the range of x values
x = np.linspace(xmin, xmax, 100)
p = stats.norm.pdf(x, mean, std)  # Calculate the PDF of the normal distribution
# Adjust the scale to match the histogram
scale_factor = np.diff(plt.gca().transData.transform([(0,0),(0,1)])[1])[0] * len(y) * np.diff(plt.xlim())[0] / 1
00
plt.plot(x, p * scale_factor, 'k', linewidth=2)

# Show the histogram with the normal distribution curve
plt.show()

# Q-Q plot
plt.figure(figsize=(10, 6))
stats.probplot(y, dist="norm", plot=plt)
plt.title('Q-Q Plot of Log Air Traffic Movements')

# Show the Q-Q plot
plt.show()

import seaborn as sns
import matplotlib.pyplot as plt
```

```
from scipy import stats
import numpy as np

# Assuming 'df' is the DataFrame and 'Log Air Traffic Movements' is the transformed column
y = df['Air Traffic Movements']

# Plotting the distribution with histogram and kernel density estimate (KDE)
plt.figure(figsize=(10, 6))
sns.histplot(y, kde=True)
plt.title('Distribution of Air Traffic Movements')
plt.xlabel('Air Traffic Movements')
plt.ylabel('Frequency')

# Fit a normal distribution to the data and plot it
mean, std = stats.norm.fit(y)  # Get the mean and standard deviation
xmin, xmax = plt.xlim()  # Get the range of x values
x = np.linspace(xmin, xmax, 100)
p = stats.norm.pdf(x, mean, std)  # Calculate the PDF of the normal distribution
# Adjust the scale to match the histogram
scale_factor = np.diff(plt.gca().transData.transform([(0,0),(0,1)])[1])[0] * len(y) * np.diff(plt.xlim())[0] / 1
00
plt.plot(x, p * scale_factor, 'k', linewidth=2)

# Show the histogram with the normal distribution curve
plt.show()

# Q-Q plot
plt.figure(figsize=(10, 6))
stats.probplot(y, dist="norm", plot=plt)
plt.title('Q-Q Plot of Air Traffic Movements')

# Show the Q-Q plot
plt.show()


import pandas as pd
import statsmodels.api as sm

# ATNS dataset
data = {
   'ATM': [48799, 46171, 42975, 40873, 39561, 49332, 46294, 49838, 47978, 49867, 49005, 46693, 4712
6, 46474, 42379,
        43304, 40744, 45703, 45085, 48931, 47320, 47626, 45774, 44595, 47729, 45843, 41585, 40519, 3
9279, 43954,
        40882, 46939, 44251, 47024, 46930, 43803, 45134, 42882, 37949, 40201, 39824, 33129, 2169, 39
27, 15177,
```

```
        19636, 19864, 23545, 26201, 26964, 28342, 23604, 24595, 32840, 34638, 35584, 33852, 31603, 3
1223, 33845],
   'Log ATM': [10.795465, 10.740107, 10.668374, 10.618225, 10.585599, 10.806328, 10.742768, 10.8165
33, 10.778498,
        10.817115, 10.799678, 10.75135, 10.76058, 10.746648, 10.654408, 10.676, 10.615064, 10.7299
19, 10.716305,
        10.798166, 10.764688, 10.771134, 10.731472, 10.705377, 10.773294, 10.732978, 10.635495, 10
.609526,
        10.578445, 10.690899, 10.618445, 10.756604, 10.697633, 10.758413, 10.756412, 10.687458, 10
.717391,
        10.666207, 10.543998, 10.601647, 10.592225, 10.408164, 7.682022, 8.275631, 9.627536, 9.885
12, 9.896664,
        10.066669, 10.173553, 10.202258, 10.2521, 10.069171, 10.110298, 10.399403, 10.452707, 10.4
79651,
        10.429753, 10.361007, 10.34891, 10.429547]
}

# Creating a DataFrame
df = pd.DataFrame(data)

# Adding a constant for the intercept
df['Constant'] = 1

# Performing linear regression
model = sm.OLS(df['Log ATM'], df[['Constant']])
results = model.fit()

# Getting p-value and F-statistics
p_value = results.pvalues[0]
f_statistic = results.fvalue

print(f"P-value: {p_value}")
print(f"F-statistic: {f_statistic}")

P-value: 6.068735041232367e-78
F-statistic: nan
import numpy as np
import pandas as pd
import statsmodels.api as sm
import matplotlib.pyplot as plt
from scipy import stats

# Load dataset
data = pd.read_excel(r"C:\Users\2020886178\OneDrive -\Desktop\Monthly ATM Data .xlsx")
data = data.drop(['Date'],axis=1)
data.head()
```

```
df = pd.DataFrame(data)

# Step 1: Fit the regression model
X = df[['Lockdown Level', 'Confirmed Covid19 Cases', 'Number of Covid19 Deaths', 'Exchange Rate', 'GDP (
in billion ZAR)']]
y = df['ATM']

X = sm.add_constant(X)  # Adding a constant
model = sm.OLS(y, X).fit()
residuals = model.resid

# Step 2: Plot histogram of residuals
plt.figure(figsize=(8, 6))
plt.hist(residuals, bins=20, edgecolor='k')
plt.xlabel('Residuals')
plt.ylabel('Frequency')
plt.title('Histogram of Residuals')
plt.grid(True)
plt.show()

# Step 3: Plot Normal Probability Plot (Q-Q Plot)
sm.qqplot(residuals, line='45')
plt.title('Normal Probability Plot (Q-Q Plot) of Residuals')
plt.show()

# Step 4: Perform Shapiro-Wilk Test
shapiro_test = stats.shapiro(residuals)
shapiro_statistic, shapiro_p_value = shapiro_test
print("Shapiro-Wilk Test Statistic:", shapiro_statistic)
print("Shapiro-Wilk Test p-value:", shapiro_p_value)

# Step 5: Interpretation
alpha = 0.05
if shapiro_p_value > alpha:
    print("The residuals appear to be approximately normally distributed (fail to reject H0)")
else:
    print("The residuals do not appear to be normally distributed (reject H0)")

import numpy as np
from sklearn.linear_model import LinearRegression
import matplotlib.pyplot as plt

# Provided data for Lockdown Level, ATM, and Log ATM
lockdown_level = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 5, 5, 4, 3, 3, 2, 1, 1, 1, 1, 1, 3, 3, 1, 1, 3, 3, 1, 1, 1, 2, 2, 1]
atm = [48799, 46171, 42975, 40873, 39561, 49332, 46294, 49838, 47978, 49867, 49005, 46693, 47126, 4
6474, 42379, 43304, 40744, 45703, 45085, 48931, 47320, 47626, 45774, 44595, 47729, 45843, 41585, 4
```

```
0519, 39279, 43954, 40882, 46939, 44251, 47024, 46930, 43803, 45134, 42882, 37949, 40201, 39824, 3
3129, 2169, 3927, 15177, 19636, 19864, 23545, 26201, 26964, 28342, 23604, 24595, 32840, 34638, 355
84, 33852, 31603, 31223, 33845]
log_atm = [10.79454893, 10.73969602, 10.66884797, 10.61768996, 10.58561207, 10.80430602, 10.743
48145, 10.81354714, 10.77631763, 10.81436362, 10.80090036, 10.75134668, 10.75900384, 10.746822
92, 10.65703796, 10.67628256, 10.61429981, 10.72800345, 10.71444088, 10.79913195, 10.76414651,
10.76759197, 10.73269408, 10.70748764, 10.77049745, 10.73426491, 10.63589964, 10.60977648, 10.5
757505, 10.68867666, 10.61764556, 10.75653292, 10.6965276, 10.75949355, 10.7579961, 10.6839879
3, 10.71868151, 10.66547827, 10.54236522, 10.60498229, 10.59119049, 10.4082849, 7.6820222, 8.275
56223, 9.62800346, 9.88420979, 9.89664884, 10.0663278, 10.17367555, 10.20170896, 10.25461444, 1
0.06951498, 10.10861731, 10.39884999, 10.45123079, 10.47987536, 10.42976492, 10.35900628, 10.34
980196, 10.42604444]

# Ensure lengths of Lockdown Level, ATM, and Log ATM arrays match
min_length = min(len(lockdown_level), len(atm), len(log_atm))
lockdown_level = lockdown_level[:min_length]
atm = atm[:min_length]
log_atm = log_atm[:min_length]

# Convert lists to numpy arrays
X = np.array(lockdown_level).reshape(-1, 1)  # Reshape to a column vector

# Perform linear regression for ATM
regressor_atm = LinearRegression()
regressor_atm.fit(X, atm)
predicted_atm = regressor_atm.predict(X)
coeff_atm = regressor_atm.coef_[0]

# Perform linear regression for Log ATM
regressor_log_atm = LinearRegression()
regressor_log_atm.fit(X, log_atm)
predicted_log_atm = regressor_log_atm.predict(X)
coeff_log_atm = regressor_log_atm.coef_[0]

# Plot ATM against Lockdown Level
plt.figure(figsize=(10, 5))

# Scatter plot for ATM
plt.scatter(X, atm, color='blue', linestyle='dotted', label='Actual ATM')
plt.scatter(X, predicted_atm, color='red', linestyle='dotted', label='Predicted ATM')
plt.plot(X, predicted_atm, color='yellow', label=f'Regression Line ATM\nATM = {coeff_atm:.2f} * Lockdow
n Level')

plt.xlabel('Lockdown Level')
plt.ylabel('ATM')
plt.title('ATM vs. Lockdown Level with Regression Line')
plt.legend()
```

```
plt.show()

# Plot Log ATM against Lockdown Level
plt.figure(figsize=(10, 5))

# Scatter plot for Log ATM
plt.scatter(X, log_atm, color='green', linestyle='dotted', label='Actual Log ATM')
plt.scatter(X, predicted_log_atm, color='orange', linestyle='dotted', label='Predicted Log ATM')
plt.plot(X, predicted_log_atm, color='purple', label=f'Regression Line Log ATM\nLog ATM = {coeff_log_at
m:.2f} * Lockdown Level')

plt.xlabel('Lockdown Level')
plt.ylabel('Log ATM')
plt.title('Log ATM vs. Lockdown Level with Regression Line')
plt.legend()
plt.show()
import numpy as np
from sklearn.linear_model import LinearRegression
import matplotlib.pyplot as plt

# Ensure lengths of GDP, ATM, and Log ATM arrays match
min_length = min(len(gdp), len(atm), len(log_atm))
gdp = gdp[:min_length]
atm = atm[:min_length]
log_atm = log_atm[:min_length]

# Convert lists to numpy arrays
X = np.array(gdp).reshape(-1, 1)  # Reshape to a column vector

# Perform linear regression for ATM
regressor_atm = LinearRegression()
regressor_atm.fit(X, atm)
predicted_atm = regressor_atm.predict(X)
coeff_atm = regressor_atm.coef_[0]

# Perform linear regression for Log ATM
regressor_log_atm = LinearRegression()
regressor_log_atm.fit(X, log_atm)
predicted_log_atm = regressor_log_atm.predict(X)
coeff_log_atm = regressor_log_atm.coef_[0]

# Plot ATM against GDP
plt.scatter(X, atm, color='blue', linestyle='dotted', label='Actual ATM')
plt.scatter(X, predicted_atm, color='red', linestyle='dotted', label='Predicted ATM')
plt.plot(X, predicted_atm, color='yellow', label=f'Regression Line ATM\nATM = {coeff_atm:.2f} * GDP')
plt.xlabel('GDP')
plt.ylabel('ATM')
```

```
plt.title('ATM vs. GDP with Regression Line')
plt.legend()
plt.show()
# Plot Log ATM against GDP
plt.scatter(X, log_atm, color='blue', linestyle='dotted', label='Actual Log ATM')
plt.scatter(X, predicted_log_atm, color='red', linestyle='dotted', label='Predicted Log ATM')
plt.plot(X, predicted_log_atm, color='yellow', label=f'Regression Line Log ATM\nLog ATM = {coeff_log_at
m:.2f} * GDP')
plt.xlabel('GDP')
plt.ylabel('Log ATM')
plt.title('Log ATM vs. GDP with Regression Line')
plt.legend()
plt.show()
```