---

title: “Analyzing Financial Data to Predict Net Income of a Building Services Company (2019–2023)” author: “Shelley Wright” date: “2025-04-26” format: pdf: documentclass: article number-sections: true toc: true lof: true lot: true execute: echo: false code-fold: true —

Introduction

Predicting Net Income accurately is vital for operational planning in business industries. This study analyzes financial data from a building services company collected between 2019 and 2023, aiming to model and predict net Income using variables such as class, year, cleaning supplies, paper goods, and payroll expenses. Multiple linear regression models were developed in analysis attempting to find the model with the best fit. During the analysis, log transformation was conducted as well as pivoting to robust regression in an attempt to address issues of non-linearity, outliers, and heteroscedasticity. Although transformations improved model fit, further refinement is needed.

Setup

Data Cleaning and Preparation

The dataset underwent extensive cleaning to correct inconsistencies, handle missing values, and prepare the variables for analysis. Data was exported to Excel by specific years. The exported data had finacial data on all accounts listed. The account names were removed and coded into one of four classes. The classes determined were commercial, industrial, medical and university affiliated. A year factor was added to the datasets accordingly and each dataset had to be transposed. Merging of the indivisual sets to create one final dataset for anaysis was the last step in the data cleaning process.

#Demographics of the Data Demographic breakdowns of the data were created to better understand the company’s factors that will be used in the regression models. The pie chart indicates the number of different accounts in the four classes cleaned during 2019-2023. The majority of the accounts were commercial (279) and the least class of accounts cleaned was medical (44).

```
library(plotrix)

# Create the class counts table
class_counts <- table(PandL$Class)
```

```r
# Define custom labels
labels <- c("Commercial", "Industrial", "Medical", "University")

# Create labels with counts included
labels_with_counts <- paste0(labels, "\n(", class_counts, ")")

# Create the exploding 3D pie chart
pie3D(
  class_counts,
  labels = labels_with_counts,
  explode = 0.3,
  height = 0.2,
  main = "Number of Different Accounts from 2019-2023",
  labelcex = 1.1,
  col = rainbow(length(class_counts))
)
```

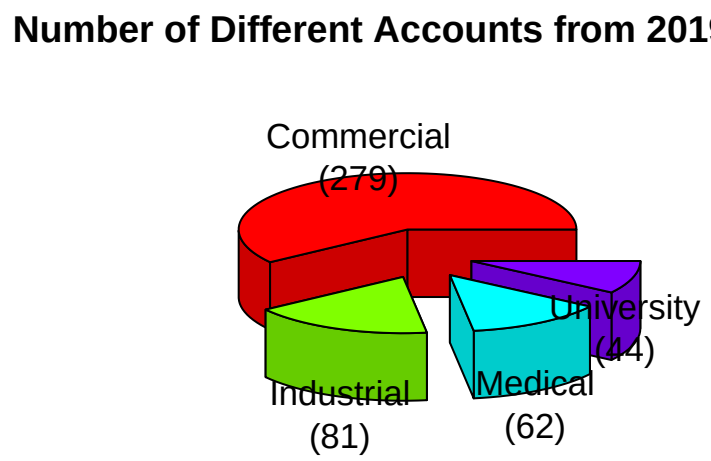

Figure 1: Class Distribution of Accounts (2019–2023) with Counts

Methods

Initial Regression Model

The initial multiple linear regression model was specified as:

$$Net\ Income \sim Class + Year + Cleaning\ Supplies + Paper\ Goods + Payroll\ Expenses$$

Key model results:

- **Residual Standard Error**: 9498
- **Multiple R-squared**: 0.8172
- **Adjusted R-squared**: 0.814
- **F-statistic**: 255.4 ($p < 2.2e\text{-}16$)

```
# Placeholder for model code
# model <- lm(Net.Income ~ Class + Year + Cleaning.Supplies + Paper.Goods + Payroll.Expens
# summary(model)
```

Model Diagnostics

Residual Analysis

- **Residual vs Fitted Plot**: Most points centered around 0.
- **Q-Q Plot**: Heavy deviation in tails, indicating non-normality.
- **Scale-Location Plot**: Upward trend suggests heteroscedasticity.
- **Residuals vs Leverage**: Few high-leverage points identified.

These diagnostic plots suggested violations of linear regression assumptions.

Transformation

Log transformation of Net Income was applied to mitigate non-linearity and heteroscedasticity.

- **Residuals vs Fitted**: Improved, but some curvature remains.
- **Q-Q Plot**: S-shape remains, tails improved.
- **Scale-Location Plot**: Spread reduced but not eliminated.
- **Leverage Plot**: Influential points persist.

Robust Regression Analysis

Given persistent issues, a **Robust Regression** was employed to reduce the influence of outliers.

Reasons for using robust regression: - Presence of outliers distorting OLS estimates - Non-normal residual distribution - Heteroscedasticity (non-constant variance)

The robust model used **M-estimation** to downweight extreme residuals.

```
# Placeholder for robust model code
# library(MASS)
# robust_model <- rlm(log(Net.Income) ~ Class + Year + Cleaning.Supplies + Paper.Goods + P
# summary(robust_model)
```

Results

Robust regression showed improved fit:

- Less sensitivity to outliers
- More stable fitted lines
- Reduced influence of high-leverage observations

However, some non-linearity and heteroscedasticity issues persisted, suggesting the need for further advanced modeling techniques (e.g., spline regression, Generalized Additive Models).

Discussion

While model performance improved with log transformation and robust techniques, residual diagnostics indicated areas needing further refinement. Future work may incorporate non-linear methods or variable interaction terms.

Conclusion

Multiple modeling strategies were explored to predict Net Income. Although transformations and robust regression improved model performance, residual analysis highlighted persistent issues requiring future research.

References

- Fox, J. (2016). *Applied Regression Analysis and Generalized Linear Models*. Sage Publications.
- Huber, P. J. (1981). *Robust Statistics*. Wiley.

Quarto

Quarto enables you to weave together content and executable code into a finished document. To learn more about Quarto see https://quarto.org.

Running Code

When you click the **Render** button a document will be generated that includes both content and the output of embedded code. You can embed code like this:

```
1 + 1
```

```
[1] 2
```

You can add options to executable code like this

```
[1] 4
```

The `echo: false` option disables the printing of code (only output is displayed).