Allocation of Supervisory Resources for Insurance Firms
Bank of England Assessment

Shubhani Jain

March 2024

Contents

1 Introduction

To determine which insurance firms need the most attention, this report will analyze data and offer insights in response to the need for an efficient allocation of supervisory resources. The analysis considers several metrics mentioned in the assessment file, such as outliers from the norm, changing business profiles, and firm size.

Here's a quick overview of the key metrics in the dataset:

Gross Written Premium (GWP): This metric measures an insurer's total revenue generated from premiums. In non-insurance terms, it is comparable to turnover. The GWP provides insight into an insurer's overall operational scale and revenue-generation capacity.

Net Written Premium (NWP): The NWP is calculated by subtracting the reinsurance cost from the GWP. It represents the percentage of risk the insurer retained after transferring some risk to reinsurers. The NWP-to-GWP ratio indicates how much risk the insurer offloads to reinsurers.

SCR Coverage Ratio: The SCR (Solvency Capital Requirement) coverage ratio is an important indicator of an insurer's compliance with prudential capital requirements. A ratio greater than 100% indicates that the insurer has sufficient capital to meet regulatory requirements. The surplus over 100% acts as a buffer, which is important when assessing the insurer's financial resilience.

Gross Claims Incurred: This metric represents an insurer's total costs incurred as a result of policyholder claims. Monitoring variations in gross claims over time is critical for determining the financial impact of claims on an insurer's operations and profitability.

Net combined ratio: This is calculated as the ratio of incurred losses plus expenses to earned premiums. It is a key indicator of an insurer's profitability. A net combined ratio below 100% indicates that the insurer is profiting from its underwriting activities.

Understanding and analyzing these metrics and outliers from the norm yields valuable insights into insurance companies' financial performance, risk management practices, and overall viability. Using these metrics, stakeholders can make informed decisions about investment, risk assessment, and regulation.

The layout of the report is as follows: In Section 2, we analyse the dataset to identify firms that need immediate attention. In Section 3, we implement the clustering machine learning technique to further gain insight into the relevant metrics driving the decision-making of the supervisor. In Section 4, we give a thorough, step-by-step explanation of implementing Microsoft Azure for the same. In Section 5, we present our conclusion.

2 Task I: Data Analysis and Firm Attention Assessment

Analysis Summary:
In this section, we analysed the provided dataset using Python libraries such as Pandas, Numpy, Matplotlib, and Seaborn. The dataset includes metrics for various firms over multiple years, such as gross written premium (GWP), net written premium (NWP), SCR coverage ratio, gross claims incurred, and net combined ratio. Our analysis sought to identify firms that warranted special attention based on these metrics.

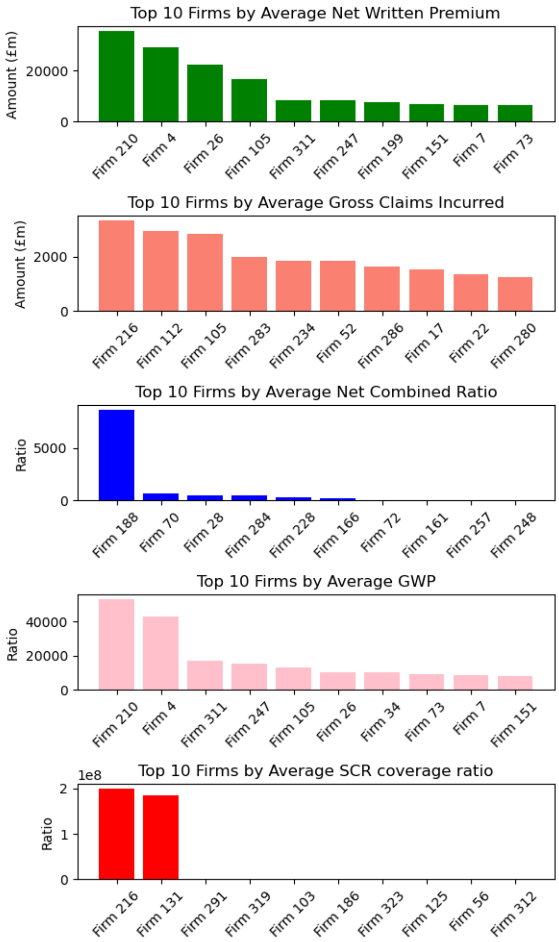


Figure 1: Top 10 Insurance Firms outliers (General overview): Average Net Written Premium, Average Gross Claims Incurred, Average Net Combined Ratio, Average GWP, and Average SCR coverage ratio.

We began our analysis by loading the dataset and running basic descriptive statistics to fully understand the data's characteristics. This included calculating summary statistics such as the mean, standard deviation, minimum, maximum, and quartiles for each metric. Subsequently, we used various charts to visualise the average metrics data and outliers data (average here means that mean for all years per metric), allowing us to gain insights and highlight potential areas of interest. Our visualisations included scatter plots, histograms, and bar plots, which allowed us to explore the relationships between various metrics and identify patterns or outliers.

Key Findings: Our analysis identified firms with significant deviations from the norm in key metrics like average net written premium, average GWP, average SCR coverage ratio, average gross claims incurred, and average net combined ratio. These identified firms may require closer scrutiny and the allocation of resources for additional investigation.

In Fig. 1, we plot the top 10 firms that need special consideration from the supervisor based on each category: average NWP, average gross claims incurred, average net combined ratio, average GWP, and average SCR coverage ratio. It appears that firms do not overlap across multiple lists. For instance, Firm 210 holds the highest average net written premium at £35,000. Similarly, Firm 216 ranks highest in average gross claims incurred at £3,200. Finally, Firm 188 boasts the highest net combined ratio at £8,000. Supervisors can use these metrics to guide investment decisions in the insurance industry. Supervisors can strategically allocate resources by analyzing the top ten firms based on these metrics, focusing on firms with high revenue potential, high-risk areas, and potential profitability challenges. This integrated approach enables proactive regulatory oversight, ensuring that resources are directed where they are most needed to reduce risks and improve the insurance industry's stability.

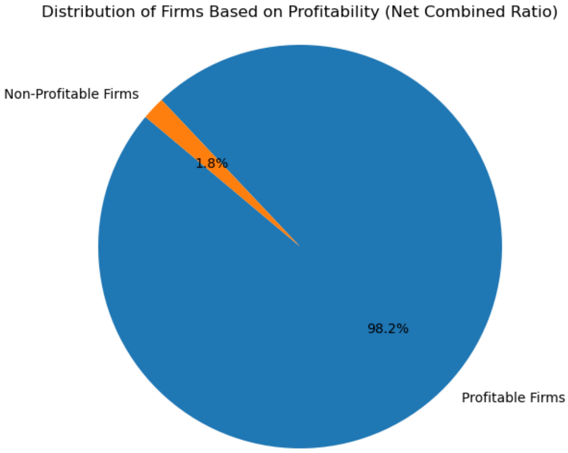


Figure 2: Distribution of firms based on profitability using Average Net combined ratio.

Fig. 2 depicts the distribution of firms according to their profitability, as measured by the average net combined ratio. We consider all those firms as profitable for which the average net combined ratio is less than 100% and others as non-profitable. It was found that around 98.2% firms were profitable. Certain firms (1.8%) demonstrate high net combined ratios, potentially indicating unprofitability or recent losses. Based on this figure, the supervisor can strategically assist firms by identifying profitable firms for commendation and knowledge-sharing, providing targeted assistance to the majority of unprofitable firms, and allocating resources to them. Assisting non-profit organizations with financial analysis, operational reviews, training, and strategic planning can help to address inefficiencies and promote long-term profitability. By monitoring progress and encouraging resilience and innovation, the supervisor can help firms navigate challenges and thrive in a competitive environment.

Fig. 3 shows the relationship between net written premiums and net combined ratios in the insurance industry. Despite a discernible weak positive correlation, this plot indicates that firms with higher premiums typically have higher combined ratios and significant deviations indicating unprofitability. On the other hand, other firms showed disproportionately low premiums but high combined ratios, indicating nuanced risk profiles or recent significant losses.

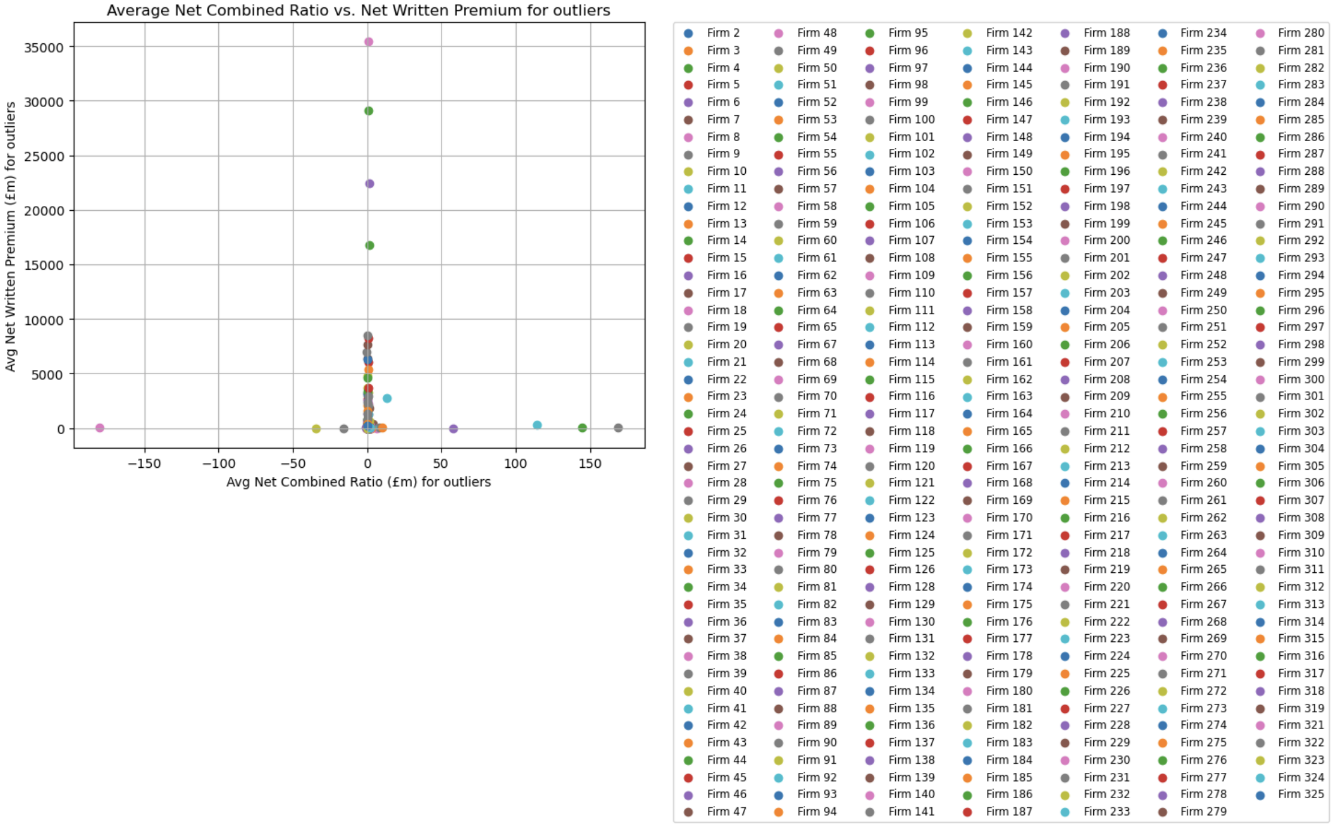
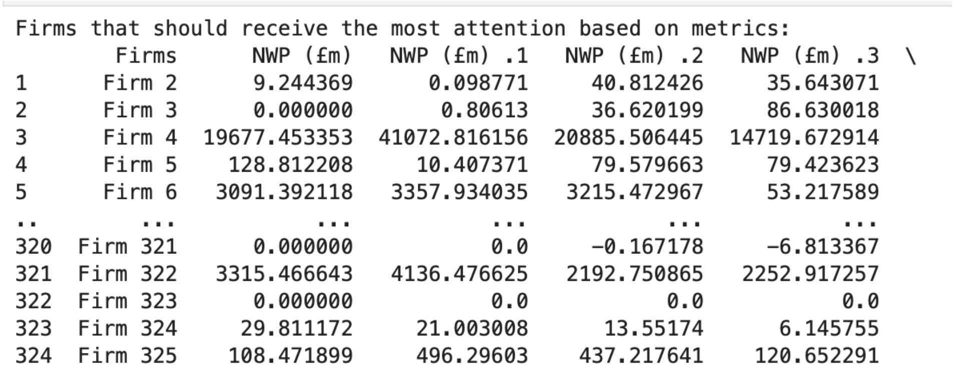


Figure 3: Distribution for Average Net combined ratio vs average net written premium.

For completeness, we also present a table consisting of outlier firms that need more attention from the supervisor in Fig. 4.

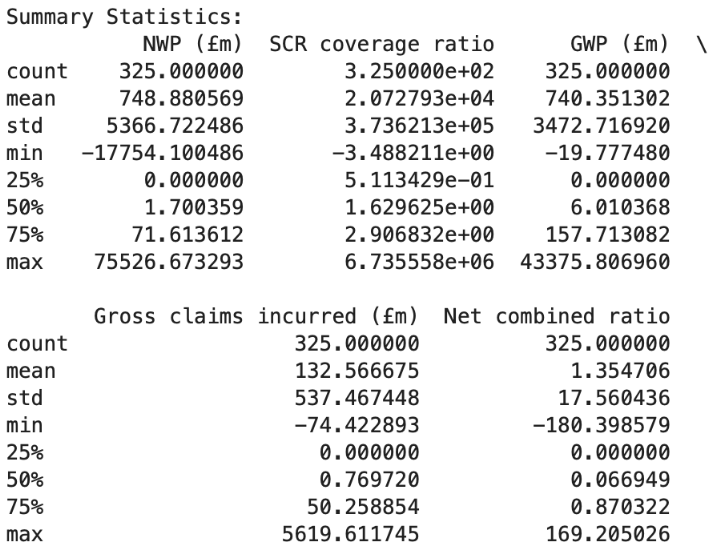


```
Firms that should receive the most attention based on metrics:
        Firms       NWP (£m)    NWP (£m) .1    NWP (£m) .2    NWP (£m) .3  \
1      Firm 2       9.244369       0.098771      40.812426      35.643071
2      Firm 3       0.000000        0.80613      36.620199      86.630018
3      Firm 4   19677.453353   41072.816156   20885.506445   14719.672914
4      Firm 5     128.812208      10.407371      79.579663      79.423623
5      Firm 6    3091.392118    3357.934035    3215.472967      53.217589
..        ...            ...            ...            ...            ...
320  Firm 321       0.000000            0.0      -0.167178      -6.813367
321  Firm 322    3315.466643    4136.476625    2192.750865    2252.917257
322  Firm 323       0.000000            0.0            0.0            0.0
323  Firm 324      29.811172      21.003008       13.55174       6.145755
324  Firm 325     108.471899      496.29603     437.217641     120.652291
```

Figure 4: Firms that should receive the most attention based on metrics.

In Fig. 5, we analyse summary statistics that yield valuable insights into the distribution and characteristics of key metrics in the dataset, shedding light on the variability and central tendencies observed across insurance firms. The presence of negative values, including the minimum NWP of -£17,754.10 million, indicates potential anomalies or instances of financial losses that warrant further investigation. Instances of negative ratios in the average SCR coverage ratio raise regulatory concerns and highlight the need for increased scrutiny. The same can be seen for the other three metrics, suggesting potential issues with underwriting profitability or data integrity.

These findings highlight the complex interplay of financial metrics in assessing insurer performance,



```
Summary Statistics:
             NWP (£m)  SCR coverage ratio      GWP (£m)  \
count      325.000000        3.250000e+02    325.000000
mean       748.880569        2.072793e+04    740.351302
std       5366.722486        3.736213e+05   3472.716920
min     -17754.100486       -3.488211e+00    -19.777480
25%          0.000000        5.113429e-01      0.000000
50%          1.700359        1.629625e+00      6.010368
75%         71.613612        2.906832e+00    157.713082
max      75526.673293        6.735558e+06  43375.806960

       Gross claims incurred (£m)  Net combined ratio
count                  325.000000          325.000000
mean                   132.566675            1.354706
std                    537.467448           17.560436
min                    -74.422893         -180.398579
25%                      0.000000            0.000000
50%                      0.769720            0.066949
75%                     50.258854            0.870322
max                   5619.611745          169.205026
```

Figure 5: Summary statistics for all the metrics.

emphasizing the importance of nuanced analysis in deciphering underlying trends and informed strategic decision-making within the industry. Other metrics and more thorough analysis can be used to gain a better understanding of the data and the allocation of resources more effectively.

3 Task II: Machine Learning Insights

In Task II, we used relevant machine-learning techniques to extract additional insights from the datasets. We used clustering analysis to find additional patterns and relationships in the data.

Clustering Analysis: We used K-means clustering to group firms according to their performance metrics. By categorizing firms, we hoped to identify similarities and differences between them, which would aid in understanding their behavior and potential risk profiles. K-means clustering was applied with three clusters, and labels were added to the data frame. Centroids were calculated to understand central tendencies within clusters.

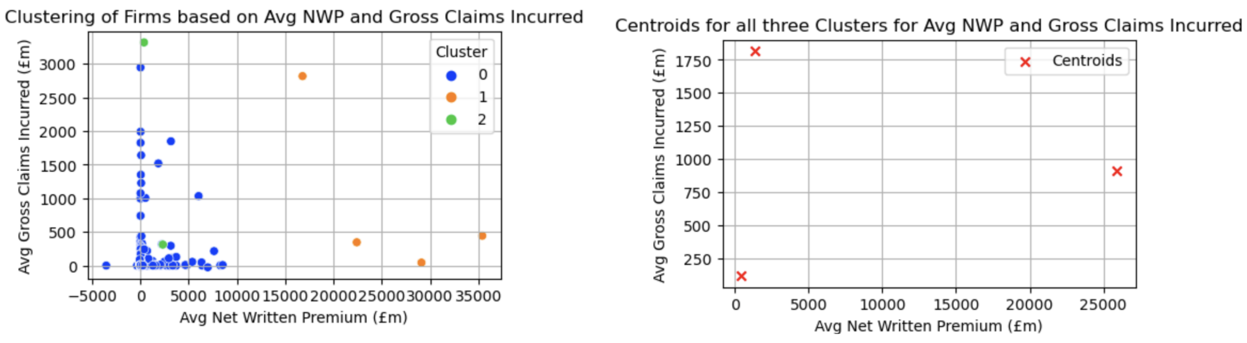


Figure 6: Right: Distribution for clustering of firms based on average NWP and gross claims incurred. Left: centroids information for average NWP and gross claims incurred.

The data reveals that Cluster 0 is the most populated group, comprising 319 firms, while Clusters 1 and 2 exhibit smaller sizes, housing only 4 and 2 entities, respectively. From Fig. 6, we can see that clusters in the bottom-left and top-right quadrants are more concentrated, indicating distinct groupings of firms based on their average net written premium and gross claims incurred. Clusters in the bottom-left quadrant are likely to represent firms with less financial activity, whereas those in the top-right quadrant represent entities with more financial engagement. Centroids within each cluster provide additional insights into the typical characteristics of firms, allowing for the identification of common attributes and targeted interventions.

Other machine learning techniques that can be applied to this problem are regression, classification, and anomaly detection. Implementing all these techniques is beyond the scope of this report.

The takeaway message is that supervisors can use these insights to prioritize supervision efforts, focusing on clusters with a higher firm density to ensure thorough oversight and regulatory compliance. We gained valuable insights into the dataset by using machine learning techniques, allowing us to identify firms that needed to be addressed and understand the underlying factors influencing their performance. These insights can help supervisory teams make strategic decisions and allocate resources effectively.

4 Task III: Cloud-based Data Processing and Analytics Pipeline using Microsoft Azure

Task III involves transitioning the data processing and analytics pipeline to cloud technologies, specifically using Microsoft Azure. This means designing a system that can ingest, process, analyze, and visualize data, with an emphasis on batch processing since the data is processed daily.

To accomplish this, we would use Azure services such as Azure Blob Storage for raw and processed data storage, Azure Data Factory for pipeline orchestration and automation, Azure Databricks for data transformation and analytics, Azure Machine Learning for training and deploying ML models, Azure Synapse Analytics for data warehousing and querying, Azure Analysis Services for BI model development, and Azure Monitor for monitoring and compliance.

Let's break down this task based on the assessment's requirements and how it relates to Microsoft Azure:

- Data Ingestion Strategy on Azure Platform:

  1. Azure Blob Storage: Utilise Azure Blob Storage to store data files containing insurance firm information. The reason for using blob storage is to store large amounts of structured and unstructured data in a scalable and cost-effective manner.

  2. Azure Data Factory: Utilise Azure Data Factory to build pipelines that ingest data from various sources into Blob Storage. Data Factory enables the orchestration of data movement and transformation activities, making it a dependable and scalable solution for data ingestion. One can also refer to a Python notebook in the Databricks workspace to perform data cleaning, feature engineering, analysis, and output generation.

- Data Preparation and Processing: Utilise Azure Databricks to perform data preparation tasks. Databricks provides a collaborative environment for users to clean, transform, and preprocess data using big data tools like Apache Spark. Databricks notebooks can be used to interactively explore and preprocess insurance data with Python libraries such as Pandas and Spark.

  1. Perform data cleaning to address missing values, outliers, and reporting errors (like mentioned in Task I). This can include techniques like outlier detection and removal, imputation, and error correction.

  2. Implement feature engineering, which creates new features from existing ones. For example, create new features such as the NWP/GWP ratio, year-over-year changes, and other metrics useful for supervisory analysis.

  3. Implement the analysis and visualization functionalities in the code examples (described in Task I). Optionally, incorporate machine learning techniques (for example, Task II) into the notebook using libraries such as SKLearn.

4. Write the processed and enriched data to an Azure Synapse Analytics workspace.

- Machine Learning Model Implementation: Utilise the Azure Machine Learning service to build and deploy machine learning models. Azure AutoML can be used to train models, automate model selection, and tune hyperparameters. This service assists in quickly determining the best-performing model for a given dataset and problem. As an alternative, create unique machine learning models with well-known frameworks like TensorFlow or Scikit-Learn. For model training, Azure Databricks clusters or Azure Machine Learning compute instances can be utilized. The trained models can then be deployed within the working pipeline.

- Data Consumption and Warehouse: Create an Azure Synapse Analytics workspace to use as a data warehouse for storing processed and enriched data. Synapse Analytics provides a unified platform for data warehousing, integration, and analysis. One can configure the Databricks notebook to save the final DataFrame with prioritized firms and insights to a dedicated table in the Synapse workspace.

- Visualisation and Reporting: Utilise Power BI for data visualization and reporting, if using Matplotlib or Seaborn is not enough. One can connect Power BI to Azure Synapse Analytics to create interactive dashboards and reports to visualise key metrics and insights gleaned from data analysis and machine learning models. Power BI provides a wide range of visualization options, as well as easy report sharing and collaboration.

- Maintenance and Monitoring: Set up monitoring and logging in Azure Monitor to track the performance and health of the data pipeline and machine learning models. Set up alerts for pipeline anomalies or failures.

In Fig. 7, we depict the flowchart to show a step-by-step procedure to follow for implementing data analysis techniques. on Microsoft Azure.

In choosing Microsoft Azure for the implementation of our data processing and analytics pipeline, several factors have been considered to ensure an optimal solution. Azure's comprehensive suite of cloud services provides a tailored solution for efficiently and effectively managing and analysing large datasets. With Azure's robust infrastructure and wide range of services, we can seamlessly integrate various pipeline components, from data ingestion to machine learning model deployment, all within a unified and scalable environment.

Azure was selected primarily because of its ability to offer comprehensive support for data pipeline requirements. Any dataset can be securely and scalablely stored with Azure Blob Storage, and managing the tasks of data ingestion, transformation, and movement can be made easier with Azure Data Factory. Furthermore, we can use Azure Databricks as a scalable and collaborative platform for data processing and analytics. Additionally, we can easily create, train, and implement machine learning models with the Azure Machine Learning service, which makes it easier to generate insights and predictive analytics from our data.

In addition, Azure's dedication to security and compliance standards was a major factor in our choice. Azure maintains industry-leading security certifications and practices, guaranteeing our sensitive insurance company data is always protected. All things considered, Microsoft Azure is the best option for our data processing and analytics pipeline since it provides a complete and safe platform that satisfies our requirements for compliance, scalability, and dependability.

For the assessment, supervisors can efficiently allocate resources to insurance companies based on their size, business profile changes, and outlier detection by utilizing advanced analytics techniques and cloud-based technologies on Microsoft Azure. This approach allows for proactive supervision and risk management, which contribute to the stability and resilience of the insurance industry.

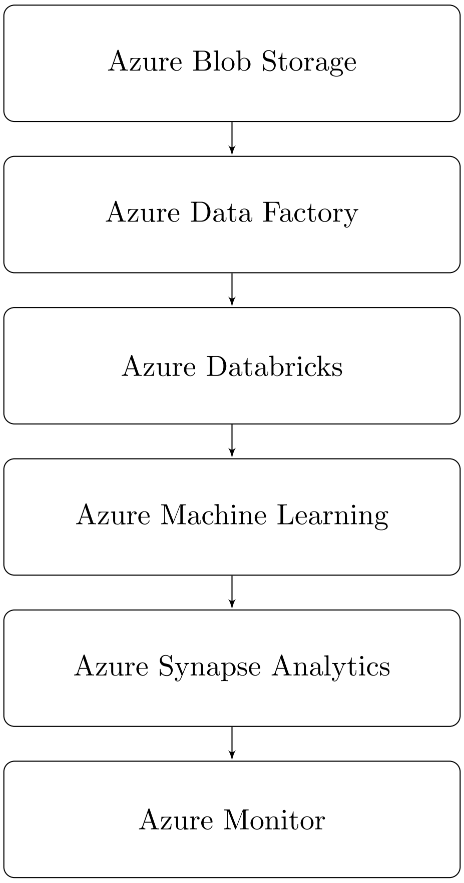


Figure 7: Flowchart to show a step-by-step procedure to follow for implementing data analysis techniques on Microsoft Azure.

5 Conclusion

In summary, our analysis has shed important light on the traits and operations of insurance industry businesses. The first task was to determine which companies were worthy of notice by looking at metrics like size, changes in the company's profile, and anomalies from the norm. We showed the distribution of businesses based on size, profitability, and other important parameters using graphics like pie charts and histograms. This analysis informs the allocation of resources and support to maximize impact and foster industry resilience.

Task 2 used machine learning techniques to extract more detailed insights from the data. We discovered patterns and relationships in the dataset using regression models, clustering algorithms, and other machine-learning approaches, allowing for more targeted interventions and strategic decision-making. These findings pave the way for more tailored strategies to improve firm performance and competitiveness.

Finally, Task 3 aimed to provide insight into Microsoft Azure for the implementation of our data processing and analytics pipeline. Leveraging Microsoft Azure for building an end-to-end data processing and analytics pipeline offers numerous benefits, including flexibility, scalability, reliability, and comprehensive tooling. Azure's integrated ecosystem of services, coupled with its strong focus on security and compliance, makes it a compelling choice for organizations looking to harness the power of cloud technologies for their data-driven initiatives. By cultivating an innovative and resilient culture, the insurance industry can navigate uncertainty and thrive in an ever-changing environment.

Overall, this report provides a roadmap for supervisors and stakeholders to use data-driven insights to guide interventions, promote industry-wide improvement, and ensure the long-term success of insurance firms.

A Task II: Code snippets for reference

Here's an approach using clustering analysis to group firms based on their performance metrics. This code carries out the subsequent actions:

- Chooses pertinent features to be clustered.
- Ensures that the features have the same scale by standardizing them.
- Utilises three clusters for KMeans clustering.
- Calculate the centroids of each cluster
- Utilises a scatter plot to display the clusters.
- Prints each cluster's number of businesses.

```
'''
Here I have selected the mean values of all the metrics for all years to be
my features. One can change that according to their preference or necessary
requirements.
'''


# Select relevant features for clustering
rel_features = df[['avg_NWP', 'avg_SCR_coverage_ratio', 'avg_GWP',

                   'avg_Gross_claims_incurred', 'avg_Net_combined_ratio']]

# Standardize the features
scaler = StandardScaler()
scaled_features = scaler.fit_transform(rel_features)

# Apply K-means clustering with arbitrary k (number of clusters)
kmeans = KMeans(n_clusters=3, random_state=42)
kmeans.fit(scaled_features)
cluster_labels = kmeans.labels_

# Add cluster labels to DataFrame
df['Cluster'] = cluster_labels

# Calculate the centroid of each cluster to understand the central tendencies
#of the data points within the cluster
centroids = scaler.inverse_transform(kmeans.cluster_centers_)
cluster_centers = pd.DataFrame(centroids, columns=rel_features.columns)

# Visualise clusters to displays the clusters formed by the algorithm, enabling
#a visual interpretation of patterns within the data
plt.figure(figsize=(5, 3))
sns.scatterplot(x='avg_NWP', y='avg_Gross_claims_incurred', hue='Cluster',
                data=df, palette='bright', legend='full')
plt.title('Clustering␣of␣Firms␣based␣on␣Avg␣NWP␣and␣Gross␣Claims␣Incurred')
plt.xlabel('Avg␣Net␣Written␣Premium␣(\textsterling␣m)')
plt.ylabel('Avg␣Gross␣Claims␣Incurred␣(\textsterling␣m)')
plt.grid(True)
plt.savefig('example_plot1_clus.png')
plt.show()

plt.figure(figsize=(5, 3))
```

```
sns.scatterplot(x='avg_SCR_coverage_ratio', y='avg_SCR_coverage_ratio',
                 hue='Cluster', data=df, palette='bright', legend='full')
plt.title('Clustering of Firms based on Avg SCR Coverage Ratio and Net
           Combined Ratio')
plt.xlabel('Avg SCR Coverage Ratio')
plt.ylabel('Avg Net Combined Ratio')
plt.grid(True)
plt.savefig('example_plot2_clus.png')
plt.show()

# Visualisation of cluster characteristics for identifying distinct centroids
plt.figure(figsize=(5, 3))
plt.scatter(cluster_centers['avg_NWP'], cluster_centers['avg_Gross_claims_incurred
             '],color='red', marker='x', label='Centroids')
plt.xlabel('Avg Net Written Premium')
plt.ylabel('Avg Gross Claims Incurred')
plt.title('Centroids for all three Clusters for Avg NWP and Gross Claims
          Incurred')
plt.grid(True)
plt.legend()
plt.savefig('example_plot3_clus.png')
plt.show()

plt.figure(figsize=(5, 3))
plt.scatter(cluster_centers['avg_SCR_coverage_ratio'],
            cluster_centers['avg_SCR_coverage_ratio'], color='red', marker='x',
            label='Centroids')
plt.xlabel('Avg SCR Coverage Ratio')
plt.ylabel('Avg Net Combined Ratio')
plt.title('Centroids for all three Clusters for Avg SCR Coverage Ratio and
          Net Combined Ratio')
plt.grid(True)
plt.legend()
plt.savefig('example_plot4_clus.png')
plt.show()

# Print number of firms in each cluster
print(df['Cluster'].value_counts())

#Output
Cluster
0    319
1      4
2      2
Name: count, dtype: int64
```

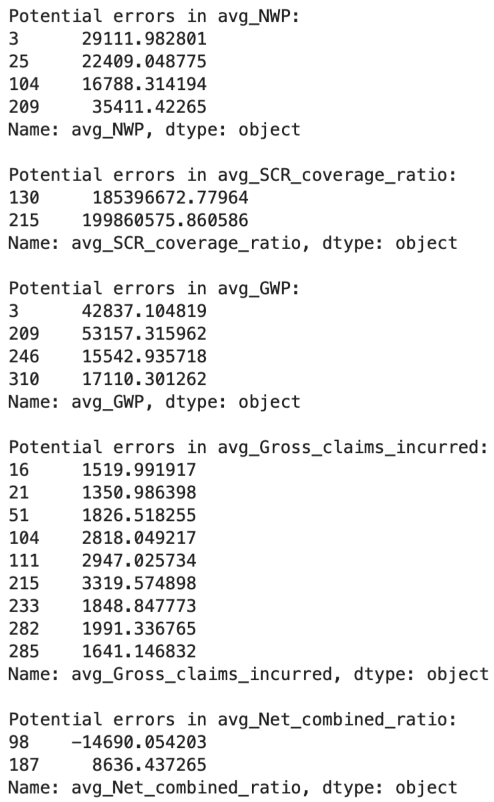
For the code, Python libraries such as pandas, numpy, scipy, matplotlib, seaborn, and sklearn were used.

B Distinguish between genuine outliers and potential errors in reporting

This code defines the identify errors() function, which uses Z-scores to identify possible reporting errors. Every data point in the designated DataFrame df has its Z-score calculated, which is then compared to a predetermined threshold (in this case, set to 3). Any data point that has an absolute Z-score greater than the cutoff is regarded as possibly erroneous. For every column, the function yields a data frame with possible errors. After that, it applies this function to the designated DataFrame columns, storing any possible errors in an err dictionary. To shed light on any possible inconsistencies or anomalies in the data about average Net Written Premium, average SCR coverage ratio, average Gross Written Premium, average Gross Claims Incurred, and average Net Combined Ratio, it finally prints out the possible errors for each column.

```
# Define a function to identify potential errors in reporting based on Z-score

def identify_errors(df):
    # apply the function to each relevant column to identify potential error
    threshold_zsc = 3  # Define the threshold for Z-score

    # Calculate Z-scores for each data point
    zsc = np.abs((df - df.mean()) / df.std())

    # Identify potential errors in reporting based on Z-score
    err = df[zsc > threshold_zsc]

    return err

# Columns to consider for outlier detection
columns_int = ['avg_NWP', 'avg_SCR_coverage_ratio', 'avg_GWP',
               'avg_Gross_claims_incurred', 'avg_Net_combined_ratio']

# Create a dictionary to store potential errors for each column
err_dict = {}

for col in columns_int:
    err = identify_errors(df[col])
    err_dict[col] = err
# Print potential errors for each column
for col, errors in err_dict.items():
    print(f"Potential␣errors␣in␣{col}:")
    print(errors)
    print()
```

Fig. 8 contains the potential errors using Z-scores to identify possible reporting errors.



```
Potential errors in avg_NWP:
3      29111.982801
25     22409.048775
104    16788.314194
209     35411.42265
Name: avg_NWP, dtype: object

Potential errors in avg_SCR_coverage_ratio:
130     185396672.77964
215    199860575.860586
Name: avg_SCR_coverage_ratio, dtype: object

Potential errors in avg_GWP:
3      42837.104819
209    53157.315962
246    15542.935718
310    17110.301262
Name: avg_GWP, dtype: object

Potential errors in avg_Gross_claims_incurred:
16     1519.991917
21     1350.986398
51     1826.518255
104    2818.049217
111    2947.025734
215    3319.574898
233    1848.847773
282    1991.336765
285    1641.146832
Name: avg_Gross_claims_incurred, dtype: object

Potential errors in avg_Net_combined_ratio:
98    -14690.054203
187     8636.437265
Name: avg_Net_combined_ratio, dtype: object
```

Figure 8: Result of the potential errors gained from the above code snippet.