**NTAsset Internship Assignment Logic Explanation**

Paramaa Sawanpanyalert

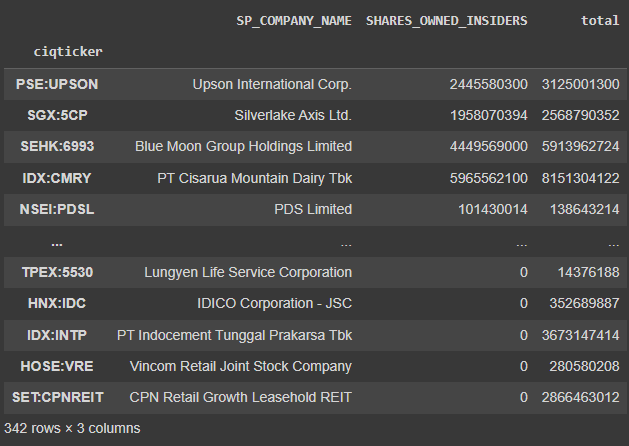
1. **Alignment of Ownership Analysis**

A screenshot of a computer

Description automatically generatedWe download the ownership information from GitHub. The first dataframe (df) contains the provided data for task 1 (Alignment of Ownership), while the second dataframe (df2) is utilized for task 2 (Insider Activity), which consists of 5 sheets of CSV.

The next five steps will outline the process of aligning ownership analysis.

1. Since the data is stored in Python object format, we need to convert it to an integer (int64) type using the convert\_to\_int function. Also, NaN values should be replaced with 0 to make calculations easier.
2. We find out the total number of companies’ shares by adding up how many each shareholder has.

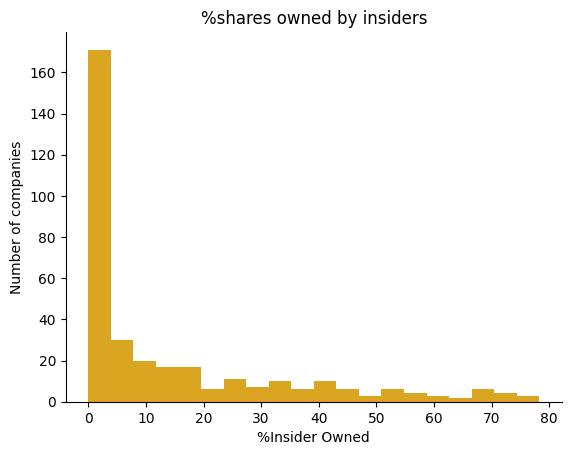
* *Creating new column name ‘total’ to collect number of shares owned by every stakeholder*

1. To figure out the insider ownership percentage, we divide the number of shares owned by insiders by the total number of shares issued.

1. Then, we make these percentages easier to compare by putting them on a scale from 1 to 10 using the score\_convertor function.

**Score convertor function**

With this formula we can normalize ownership percentage to score range from 1 to 10, which derives from typical normalization formula, as shown in figure below.

The reason why we use **normalization** over standardization is that the data is not normally distributed, as shown in the figure below.

Standardization is useful when the distribution of the data is Gaussian (normal distribution) or when the algorithm used in the analysis assumes that the data is normally distributed, as it helps to center the data around 0 with a standard deviation of 1.

Normalization is useful when the distribution of the data does not follow a Gaussian distribution (non-normal distribution) or when the algorithm used in the analysis does not assume any specific distribution of the data.

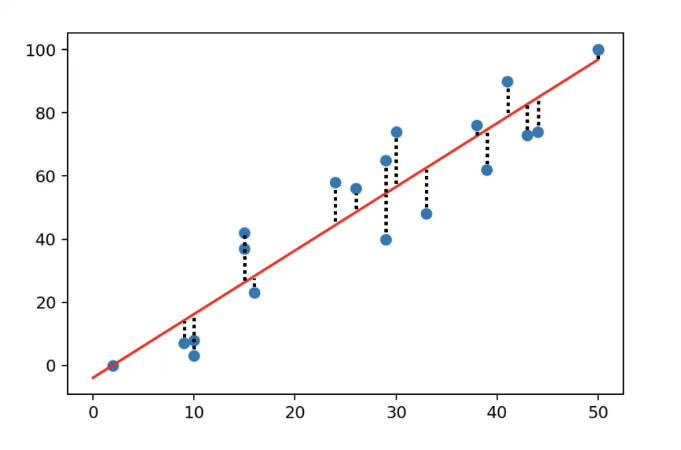
1. A screenshot of a computer

   Description automatically generatedAfter that, we calculate the ownership score and add it to the last column of the dataframe. We set up DFtask1 with just the ticker, company name, and ownership score for later merging.
2. **Insider Activity Analysis**

We can explain the entire analysis in 8 main steps, which are:

1. The ownership information is imported into dataframe 2 (df2).
2. Data cleaning involves replacing NaN values with 0 to facilitate calculations.
3. A screenshot of a computer

   Description automatically generatedInsider holdings for each company across multiple time periods are aggregated, treating all insiders collectively as if they exchange shares among themselves, while still maintaining a constant total share count.
4. Holding position data from five distinct time periods are integrated into a unified dataframe (df3) to enhance clarity and visualization.
5. The insider\_activity function facilitates the examination of the holding position value for a specific company.
6. To recognize trends in insider activity, the slope of the **best fit line** of provided numerical data serves as a reliable indicator, obtained through the ***"polyfit" function***, which determines the optimal fit line for the given dataset.

**Polyfit function**

In NumPy, the polyfit function is used to fit a polynomial of a specified degree to a set of data points using the method of least squares. It returns the coefficients of the polynomial that best fits the data.

Here's the syntax of the polyfit function:

* ***x:*** The independent variable, or the x-coordinates of the data points.
* ***y:*** The dependent variable, or the y-coordinates of the data points.
* ***deg:*** The degree of the polynomial to be fitted.
* ***rcond:*** Relative condition number of the fit. Default is len(x)\*eps, where eps is the machine precision.
* ***full:*** If True, returns additional outputs. Default is False.
* ***w:*** Weights to apply to the y-coordinates of the data points.
* ***cov:*** If True, return the covariance matrix.

A mathematical equation with numbers and symbols

Description automatically generatedThe solution minimizes the squared error

in the polynomial equations

x**[0]\*\***n **\*** p**[0]** **+** **...** **+** x**[0]** **\*** p**[**n**-1]** **+** p**[**n**]** **=** y**[0]**

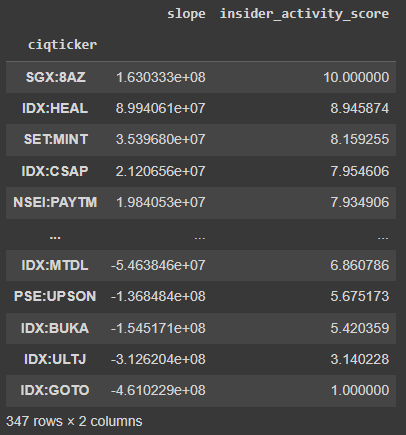
x**[1]\*\***n **\*** p**[0]** **+** **...** **+** x**[1]** **\*** p**[**n**-1]** **+** p**[**n**]** **=** y**[1]**

**...**

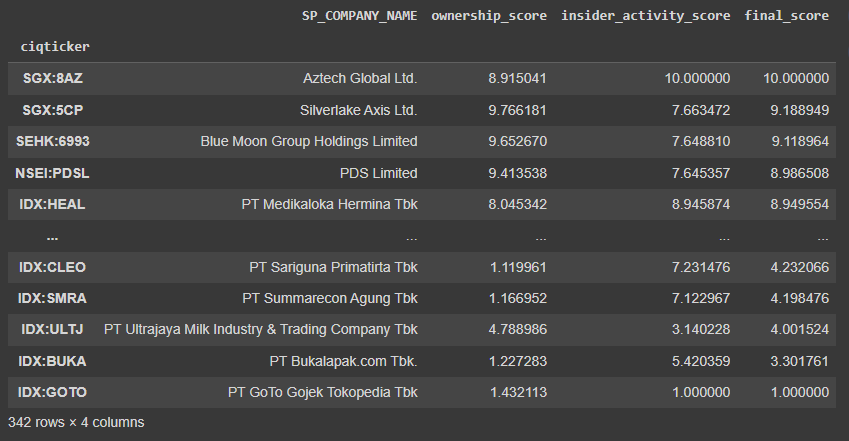
x**[**k**]\*\***n **\*** p**[0]** **+** **...** **+** x**[**k**]** **\*** p**[**n**-1]** **+** p**[**n**]** **=** y**[**k**]**

In this case, we use only 1- degree linear equations fitting the given points

We focus on ***m*** as being a slope of the best fit line.

1. Subsequently, the slope is normalized to a scale ranging from 1 to 10 using the score\_convertor function.
2. The insider score is then assigned to dftask2, preparing it for integration with dftask1 into the final dataframe.
3. **Aggregation of Ownership Factors:**

* The final scores are aggregated using the **median** method.
* It is important to regard the score as ordinal data due to the inconsistent intervals between each score, albeit retaining an ordered structure. Given that the rating or score represents ordinal data, the suitable statistical measures include mode and median.
* The selection of the median over the mode is justified as it offers a more robust measure of central tendency. Then, we also apply normalization to the final score column making it ranged from 1 to 10 using the score\_convertor function.

Here’s the final result table, sorted by final score(ranget right column), consisting of 5 columns, including

1. Company name

2. CIQ ticker

3. Alignment of ownership score

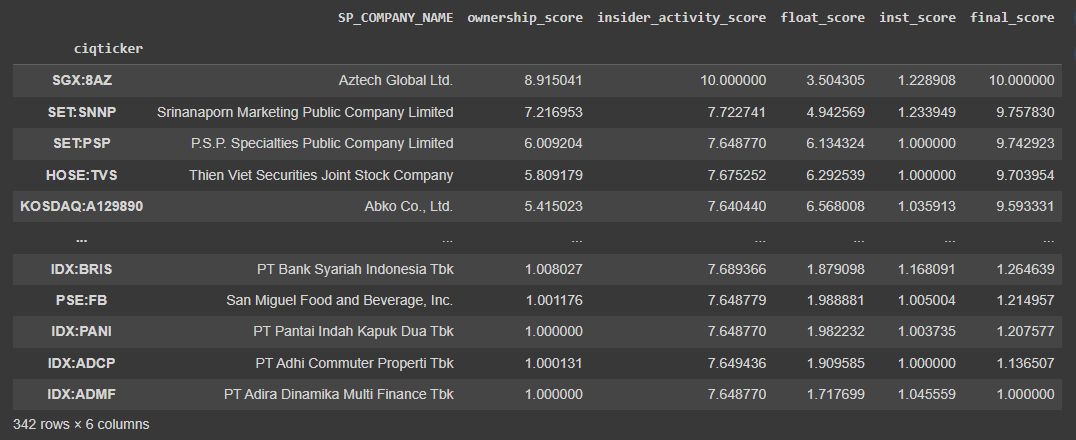
4. Insider activity score

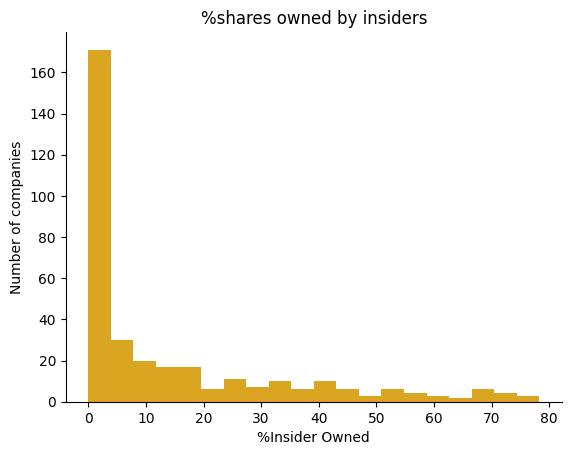
5. Ownership score

1. **Other factors need to be considered for LT investment perspective:**
   1. **Share Buybacks**: When a company repurchases its own shares from the market. Share buybacks can signal confidence from management and may indicate that the company believes its stock is undervalued. This can be calculated by ***comparing number of outstanding shares*** in different period of time.

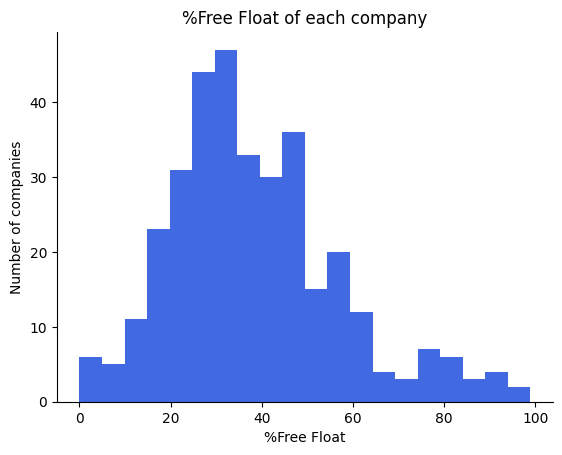
* Number of outstanding shares of 2 different periods of time are required for compute this factors.
  1. **Short Interest**: The percentage of a company's shares that have been sold short by investors betting that the stock price will decline. High short interest can indicate bearish sentiment, while low short interest may suggest bullish sentiment. It derives from number of the number of shorted shares divided by the number of shares outstanding
* This will require number of shares sold short for calculating short interest value.
  1. **Float**: The number of shares available for trading by the public. Low float stocks can be more volatile due to limited supply, while high float stocks may have more stable prices
* This value can be calculated based on given data.
* For long-term investments, we might prefer price stability over short spikes in price due to a lack of float.
  1. **Ownership of institutional investors**: The percentage of a company's shares held by institutional investors, such as mutual funds, pension funds, and hedge funds. High institutional ownership can indicate confidence from professional investors and may influence stock price movements.
* This value can be calculated based on given information.

In addition, I will show calculations of both float and institutional holding score also in the python notebook.

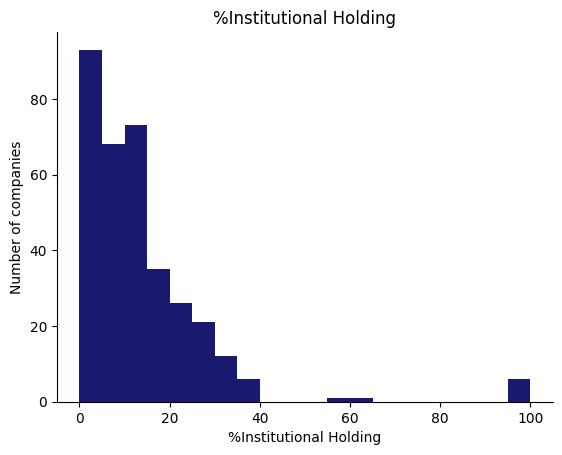
Here’s a result of combining those 4 factors into final score:



Distribution of number of company categorized by percentage of insider ownership.



Distribution of number of company categorized by percentage of free float



Distribution of number of company seperated by percentage of institutional holding