

**Realindex – Graduate Quantitative Developer   
Technical Assessment**

*The objective of this task is to write code, using either R or Python, to calculate some simple metrics relating to the carbon footprint of a portfolio of listed companies. As a rough guide you should expect this task to take approximately 30-60 minutes, though it will not be timed.*

Carbon Intensity (CI) is a popular metric for assessing the carbon footprint of a company. It is calculated as the ratio of carbon emitted by the company (over a given fiscal year, in tons of CO2), to the revenue of the company (over the same period in $million USD). The carbon emitted by a company is typically calculated by summing the carbon emitted directly as a result of their business operations (Scope 1), and the carbon emitted when generating energy, such as electricity, that is consumed by the company (Scope 2).

Investors are often interested in the aggregate carbon footprint for a portfolio of companies that they are invested in. The weighted average Carbon Intensity (WACI) of all the companies in the portfolio can be used for this purpose. The weights that are used in this calculation represent the proportional dollar value of each investment in the portfolio on a given date.

WACI can also be calculated for a benchmark portfolio (such as the ASX 20, which tracks the performance of the largest 20 companies listed on the Australian Stock Exchange). In this case the weights reflect the proportional total market value of each company in the benchmark on a given date.

To formalise this, we can define WACI as

*Where:*

***n*** is the total number of companies in the portfolio, on a given date

***units*** is the number of units of a company held in the portfolio, on a given date

***price*** is the price of one unit of the company, on a given date, in a given currency

***C02scope1*** is the last reported annual Scope 1 carbon emissions for a company (in CO2 T), on a given date

***C02scope2***is the last reported annual Scope 2 carbon emissions for company (in CO2 T), on a given date

***revenue*** is the last reported annual revenue for a company (in $million USD), on a given date

**Using the information above, please implement code to answer the following questions:**

Using the provided data for the RI\_20 portfolio and the ASX\_20 benchmark

1. Calculate the Weighted Average Carbon Intensity (WACI) for both the portfolio and the benchmark.
2. For both the portfolio and benchmark, calculate the WACI within each category group, and the contribution of each category to the total WACI calculated in part 1.
3. For both the portfolio and benchmark, use a bar graph to visualise the category data calculated in part 2.
4. State any assumptions you have made (if any).

You must use either R or Python to calculate the results, and submit this code with your response. You can make use of contributed libraries/packages in your code if you choose to. We will be assessing the quality of your code as well as the accuracy of your answers. If you are selected to proceed to the interview stage of the recruitment process be prepared to explain your code and answer questions related to your solution.

If you have any questions about the assessment please contact:  
Layla St Clair - [layla.stclair@firstsentier.com](mailto:layla.stclair@firstsentier.com)

**Answers**

1. Portfolio WACI Answer: 198.41  
   Benchmark WACI Answer: 161.14
2. Category Group Breakdown

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Portfolio** | | **Benchmark** | |
| **Category Group** | **Weighted Average** | **Contribution to Total** | **Weighted Average** | **Contribution to Total** |
| **1** | 36.11 | 18.20% | 32.24 | 20.01% |
| **2** | 54.61 | 27.52% | 44.01 | 27.31% |
| **3** | 107.69 | 54.28% | 84.89 | 52.68% |

1. Graph  
   A graph with blue and orange squares

   Description automatically generated
2. Assumptions:

1. There is no missing data but there can be mismatched Tickers between data.

2. In case of mismatched data (NaN values), these rows are not considered.

3. The weight for portfolio is the multiplication of Units and Price for each stock, but benchmark's weight is from the IndexWeight column.

4. REVENUE\_USD is given in dollar.

**Code** – please paste your code below and also include in a zipped file (some email systems block python / R files when attached).

Language: Python

Code:

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

# Load CSV files

id\_map = pd.read\_csv("IDMap.csv")

portfolio\_holdings = pd.read\_csv("PortfolioHoldings.csv")

benchmark\_holdings = pd.read\_csv("BenchmarkHoldings.csv")

carbon\_data = pd.read\_csv("CarbonData.csv")

# Merging files

portfolio\_df = portfolio\_holdings.merge(id\_map, left\_on='Ticker', right\_on='ticker', how='left').merge(carbon\_data, left\_on='sedol', right\_on='SEDOL', how='left')

benchmark\_df = benchmark\_holdings.merge(id\_map, on='ticker', how='left').merge(carbon\_data, left\_on='sedol', right\_on='SEDOL', how='left')

# Handle Mismatched value

# Drop rows with NaN values

benchmark\_df.dropna(subset=['EMISSIONS\_SCOPE\_1', 'EMISSIONS\_SCOPE\_2', 'REVENUE\_USD'], inplace=True)

portfolio\_df.dropna(subset=['EMISSIONS\_SCOPE\_1', 'EMISSIONS\_SCOPE\_2', 'REVENUE\_USD'], inplace=True)

#revenue to Millon

portfolio\_df['REVENUE\_USD'] = portfolio\_df['REVENUE\_USD'] / 1e6

benchmark\_df['REVENUE\_USD'] = benchmark\_df['REVENUE\_USD'] / 1e6

# Calculate Weights for Portfolio

TotalCap = portfolio\_df['Units'] \* portfolio\_df['Price'].sum()

portfolio\_df['Weight'] = portfolio\_df['Units'] \* portfolio\_df['Price'] / TotalCap

# Calculate WACI for Portfolio and Benchmark

portfolio\_df['WACI'] = portfolio\_df['Weight'] \* (portfolio\_df['EMISSIONS\_SCOPE\_1'] + portfolio\_df['EMISSIONS\_SCOPE\_2']) / portfolio\_df['REVENUE\_USD']

benchmark\_df['WACI'] = benchmark\_df['IndexWeight'] \* (benchmark\_df['EMISSIONS\_SCOPE\_1'] + benchmark\_df['EMISSIONS\_SCOPE\_2']) / benchmark\_df['REVENUE\_USD']

portfolio\_WACI = portfolio\_df['WACI'].sum()

benchmark\_WACI = benchmark\_df['WACI'].sum()

# Output the results

print(f"\nPortfolio WACI: {portfolio\_WACI:.2f}")

print(f"Benchmark WACI: {benchmark\_WACI:.2f}")

# Group by CategoryGroup and sum up WACI

portfolio\_grouped = portfolio\_df.groupby('CategoryGroup')['WACI'].sum()

benchmark\_grouped = benchmark\_df.groupby('CategoryGroup')['WACI'].sum()

print("\nCategory Group Breakdown:")

print(f"{'Category Group':<20} | {'Portfolio weighted average':<30} | {'Contribution to Portfolio WACI':<30} | {'Benchmark weighted average':<30} | {'Contribution to Benchmark WACI':<30}")

for category in portfolio\_grouped.index:

    portfolio\_value = portfolio\_grouped[category]

    benchmark\_value = benchmark\_grouped.get(category, 0)

    print(f"{category:<20} | {portfolio\_value:30.2f} | {portfolio\_value/portfolio\_WACI\*100:30.2f}% | {benchmark\_value:30.2f} | {benchmark\_value/benchmark\_WACI\*100 if benchmark\_WACI != 0 else 0:30.2f}%")

# Bar graph visualization

categories = portfolio\_grouped.index

plt.bar(categories, portfolio\_grouped, alpha=0.7, label='Portfolio')

plt.bar(categories, benchmark\_grouped, alpha=0.7, label='Benchmark', bottom=portfolio\_grouped)

plt.xlabel('Category Group')

plt.ylabel('WACI')

plt.title('Category Group WACI')

plt.legend()

plt.show()

result:  
