CCT College Dublin

Assessment Cover Page

Module Title:	Strategically Thinking
Assessment Title:	Company Reviews
Lecturer Name:	James Garza
Student Full Name:	Team 2: Ana Isabel Nieves Barcenas Daniela Daia Magdalene Ejiro Awaritefe Rochana da Silva Matos Karla Carolina Alvarado Minon
Student Number:	2022455 - Ana Isabela Nieves Barcenas 2017207 – Daniela Daia 2022151 – Magdalene Ejiro Awaritefe 2022175 – Rochana da Silva Matos 2022461 - Karla Carolina Alvarado Minon
Assessment Due Date:	3 January 2022
Date of Submission:	3 January 2022

Declaration

By submitting this assessment, I confirm that I have read the CCT policy on Academic Misconduct and understand the implications of submitting work that is not my own or does not appropriately reference material taken from a third party or other source. I declare it to be my own work and that all material from third parties has been appropriately referenced. I further confirm that this work has not previously been submitted for assessment by myself or someone else in CCT College Dublin or any other higher education institution.

The Impact of reviews: What are the major factors employees look for in Company Reviews?

by,

2022455 - Ana Isabel Ana Isabel Nieves Barcenas

2017207 – Daniela Daia

2022151 – Magdalene Ejiro Awaritefe

2022175 – Rochana da Silva Matos

2022461 - Karla Carolina Alvarado Minon

Higher Diploma in Science in Data Analytics for Business Strategic Thinking

James Garza

cct College Dublin
Computing • IT • Business

Abstract

We were tasked with finding a topic that interested us as a team. After two meetings, we decided that company reviews were an intriguing subject for all of us, as is our next step for the second semester when we finish college and start to look for a job in our area. Looking for a company review will undoubtedly be a tool to build our cover letter, prepare us for the interview, and find a perfect match with our values as individuals and as an organisation. From there, we came up with a question to which we would need to find the answer: "What are the major factors employees look for in Company Reviews?"

A description of the business problem and an explanation of the project goal will be shown. The characterisation of the data, applying Exploratory Data Analyses (EDA) by analysing the better fit for the imputation of the missing values and the outliers is developed and using the feature selection model to extract the influencing factors of company reviews by using machine learning score. And other techniques will be applied using machine learning approaches and a comparison. And finally, the results were obtained based on different regression models.

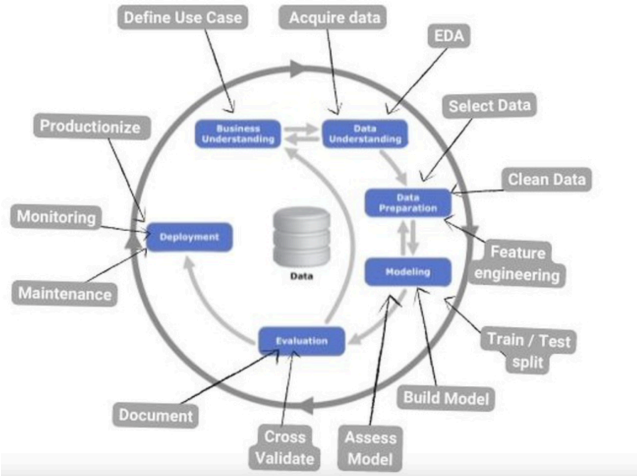


Figure 1. CRISP-DM Methodology.

Keywords: *consumer research, company reviews, CRISP-DM.*

Table of Contents

Table of Figures

Introduction

Reviews are essential to your job search because they provide valuable first-hand information. Rather than relying on the curated brand, a company offers online. Reviews provide insight into how a company looks from the perspective of current and former employees (Indeed Editorial Team, 2022).

This analysis is based on a dataset scraped from the Indeed.com website containing information about companies and their employees' ratings and Happiness, location, revenue, salaries and so on. The company reviews database raises the question, "The Impact of reviews: What are the major factors employees look for in Company Reviews?" and is available at Kaggle. All the project info was reunited in Google Drive so all members could proactively access work. The code was developed on Google Collab, also available on GitHub, following the Cross-Industry Standard Process for Data Mining (CRISP-DM) methodology.

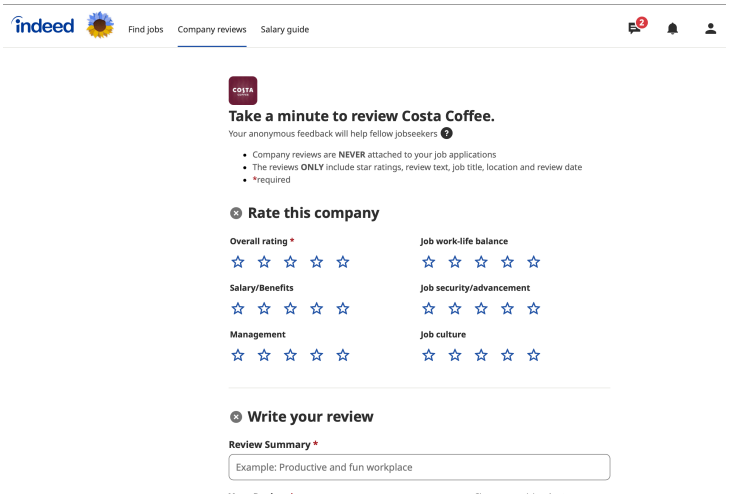


Figure 2. Example of a review page at Indeed.com

Business Understanding

To determine whether a candidate would genuinely want to work for a particular company, check the reviews of the organisation before deciding whether to apply by reading through the company reviews will also provide a better understanding of the work environment and other details about the business to aid in job interview preparation. In that sense, company reviews can also help during the job-search process in various ways, including Informing the decision-making ability, improving the understanding of a workplace, allowing to create of the content of the cover letter and to be prepared for an interview (Indeed Editorial Team, 2021).

Based on the above challenges, this study focuses on the company's review information and rating survey results. Finally, this study is also a reference for companies to use employee evaluation-driven reviews methodologies to improve their services and competitiveness.

Data Understanding

The goals of data understanding are: to understand the attributes of the data. Summarise the data by identifying key characteristics such as data volume and the total number of variables. Understand the issues with the data, such as missing values, inaccuracies, and outliers (Luna, 2021).

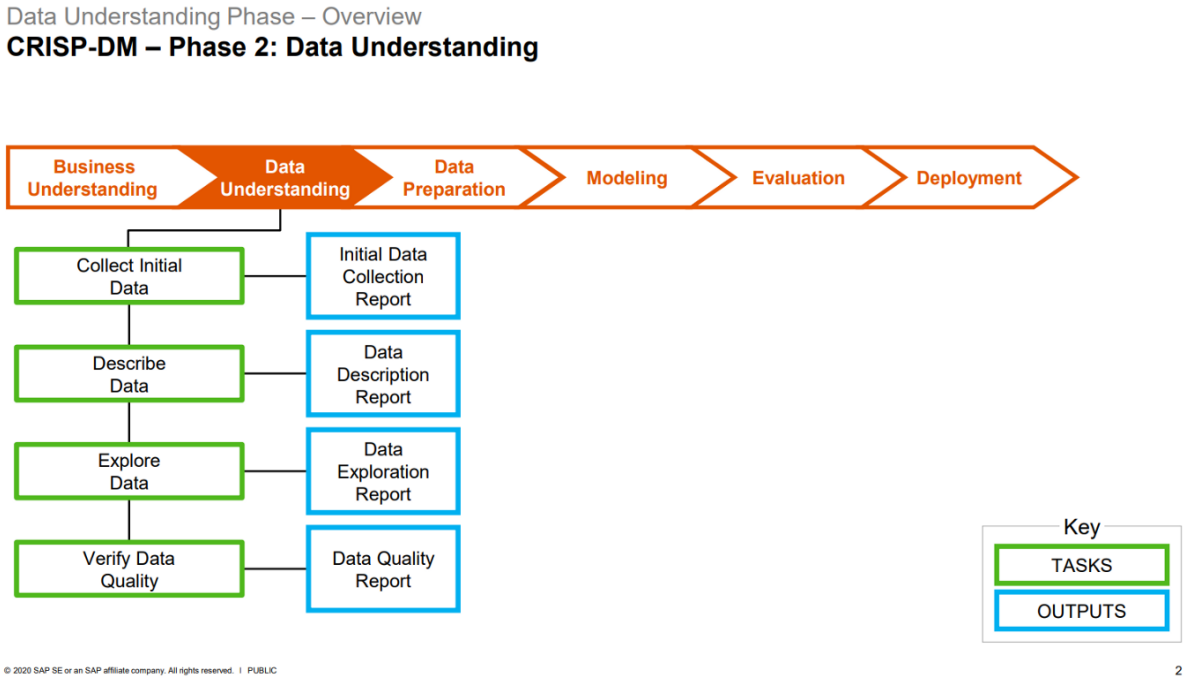


Figure 3. Data Understanding phase

But first, it is necessary to understand what each variable means by the data dictionary.

Data Dictionary

The first thing is to understand the data variables by looking in the dictionary is a file or set of files that contain information about objects in the database. It is a repository of data names, definitions, and attributes used to describe the data. It allows users to understand even the most complex databases without examining every column. Data dictionaries ensure that everyone in the company is on the same page regarding metrics and critical definitions used in the company (Wertz, 1993).

Name: Name of the company .
Ratings: The index is the result of crossing information on the size and financial performance of each company in its respective sector. Rating is calculated based on 4.6K reviews and is evolving.
Reviews: The reviews section lists all the reviews of the company. You can filter the reviews by job title and location. You can also see an overall star rating (out of five stars) and a diversity score (out of 100). The diversity score indicates how positive ratings are spread over different positions and departments. You can click on the "Ask a question" button if you would like to learn more about working in the company.
Description: The company description is a summary of the most important points of the Company: its history, the management team, where it is located, what it does and what it hopes to achieve, the mission statement and the legal structure.
Happiness: Index that indicates the levels of happiness of employees within a company.
Ceo_approval: Ratings are based on the percentage of employees who approve their CEO. Users have the option to rate their CEO. All users do not choose to review their CEO, though, which means this rating only reflects those reviewers who did review their CEO.
Ceo_count: Ratings are based on messages and employee counts approving of their CEO.
Ratings: count index the result of crossing information on the size and financial performance of each CEO.
Locations: Where the company is located.
Roles: Company Roles means serving on board of directors, serving in executive management roles or performing the functional equivalent of such roles.
Salary: This section displays the average salary after aggregating the salaries reported by employees working in different positions. You can look at the salary satisfaction score on the right side of the section to find out what percentage of employees think they are fairly paid. Common benefits at the company are listed below the salary satisfaction score.
Interview_experience: displays feedback about the job interview process here. the interview steps and interview questions listed.
Interview_count: find feedback about the job. They also rate interview difficulty from easy to difficult.
Interview_duration: Feedback with the duration of the interview process.
Headquarters: Company addresses with all descriptions.
Employees: average number of employees working for the company.
Industry: describe the sector of activity of the company.
Revenue: The total amount of money from a business's sales of goods or services related to normal business operations, with gross revenue specifically being what a business generates before any expenses are taken out.
Website: informs what type of website the companies have.

Figure 4. Data dictionary

Characterisation of the data set

Characterisation generates condensed representations of whatever information content is hidden within data (Ray, 2019). Will be imported the required packages such as Pandas, Seaborn, Numpy, Matplotlib, and Missingno for initial installation, allowing running all the analyses in the Colab notebook code and, further, other libraries were applied when necessary.

```
[1]  import pandas as pd
     import numpy as np
     import matplotlib.pyplot as plt
     import seaborn as sns
     import missingno as msno


     pd.set_option('display.max_columns', 50) # Help me to visualice all the columns


     import warnings
     warnings.filterwarnings(action='ignore')

     ##et you check if a particular string matches a given regular expression
     import re
```

Figure 5. Required Libraries

After loading, by using the "head" function, it can be seen the five rows of the data set and with the ".shape" function, it is possible to see that the data size consists of 17050 rows and 20 columns (The pandas development team, 2020).

```
[2] # Read the csv
company_reviews= pd.read_csv("/content/company_reviews.csv")
```

```
#Check the first five Rows
company_reviews.head()
```

	name	rating	reviews	description	happiness	ceo_approval	ceo_count	ratings	locations	roles	salary	interview_experience	interview_difficulty	interview_duration	intervi
0	Sitel	NaN	NaN	Sitel Group's 75,000 people across the globe c...	{'Work Happiness Score': '55', 'Achievement': ...	70%	CEO Approval is based on 4,612 ratings	{'Work/Life Balance': '3.4', 'Compensation/Ben...	{'Paradise, NV': '5.0', 'Pioneer, OH': '4.7', ...	{'Tier 1 Agent': '5.0', 'Director of Operation...	{'Customer Service Representative': '$14.48 pe...	Favorable	Easy	About a day or two	Ba
1	Meadowbrook Rehabilitation	3.7	21 reviews	You'll work with the most experienced and loya...	{}	NaN	NaN	{'Work/Life Balance': '4.1', 'Compensation/Ben...	{}	{}	{}	Favorable	Easy	NaN	
2	Intermountain	4.0	23 reviews	Why Intermountain? \n\nWe Bring Hope\n\nWith ou...	{}	88%	CEO Approval is based on 17 ratings	{'Work/Life Balance': '3.5', 'Compensation/Ben...	{}	{}	{'Mental Health Technician': '$13.16 per hour'...	Favorable	Medium	About a day or two	
3	Smith & Nephew	NaN	NaN	It's more than business at Smith+Nephew - it's...	{'Work Happiness Score': '65', 'Purpose': '71'...	76%	CEO Approval is based on 374 ratings	{'Work/Life Balance': '3.5', 'Compensation/Ben...	{'Largo, FL': '4.5', 'Chicago, IL': '4.3', 'Sa...	{'Packaging Technician': '5.0', 'Senior Associ...	{'Packager': '$30,006 per year', 'Finisher': '...	Favorable	Medium	About a week	Ba
4	Reverse Mortgage Funding	4.1	19 reviews	Reverse Mortgage Funding LLC is committed to e...	{}	NaN	NaN	{'Work/Life Balance': '4.2', 'Compensation/Ben...	{}	{}	{}	Favorable	NaN	NaN	

```
company_reviews.shape
```

```
(17050, 20)
```

Figure 6. Head and Shape of the Dataset

More information about the dataset, including its shape, kind of variables, and memory usage, is displayed using the info() function:

```
[4] company_reviews.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 17050 entries, 0 to 17049
Data columns (total 20 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   name                  16712 non-null  object
 1   rating                15616 non-null  float64
 2   reviews               15494 non-null  object
 3   description           17049 non-null  object
 4   happiness             17050 non-null  object
 5   ceo_approval          11222 non-null  object
 6   ceo_count             11222 non-null  object
 7   ratings               17050 non-null  object
 8   locations             17050 non-null  object
 9   roles                 17050 non-null  object
 10  salary                17050 non-null  object
 11  interview_experience  11499 non-null  object
 12  interview_difficulty  11462 non-null  object
 13  interview_duration    10322 non-null  object
 14  interview_count       11499 non-null  object
 15  headquarters          15230 non-null  object
 16  employees             15023 non-null  object
 17  industry              15204 non-null  object
 18  revenue               10177 non-null  object
 19  website               16129 non-null  object
dtypes: float64(1), object(19)
memory usage: 2.6+ MB
```

Figure 7. The function .info()

As can be seen, the data consists primarily of categorical variables (texts and dictionaries). At first look, it can be seen that the variable "ceo_aproval" is a percentage, and it should be a number. Also," "salary" should be a numerical variable and is recorded as categorical. Only one numerical variable, the "rating", is recorded as a float. In that sense, it will be needed to pre-process the data for further machine learning training.

Pre-processing

Any processing done on raw data to prepare it for another data processing operation is referred to as data pre-processing, which is a part of data preparation. It has historically been a crucial first stage in data mining (Lawton, 2022). In that sense, it will remove the irrelevant variables for this analysis. The reason is written in front of the variable below:

- Company name – ID unique variable.
- Description – About the company type.
- Locations – Addresses were removed because the data provided does not give much information.
- Interviews-related data – number of people who gave reviews about the company.
- Website – All the companies have social media addresses irrelevant to this analysis.
- Roles – It will be analysed the companies and not the positions.
- Salary – It was used the revenue variable instead, as the salary varies according to the role.
- Headquarters - Addresses were removed because the provided data needs more information.
- Industry – was removed because the data provided does not give much information.

```
[6]  company_reviews = company_reviews.drop(["name", "description","locations", "interview_duration",
           "interview_count","website", "roles", "salary","headquarters","industry"],axis=1)
```

Figure 8. Dropping irrelevant variables

First, it will have a look at the missing values. As seen below, the target variable" "review" has a low percentage of them. In that sense, it was decided to drop the rows with missing values. The reason for not imputing the missing values with other technics is to avoid creating fake data.

```
In [7]: company_reviews.isnull().mean()*100
Out[7]: rating                   8.410557
        reviews                  9.126100
        happiness                0.000000
        ceo_approval            34.181818
        ceo_count               34.181818
        ratings                  0.000000
        interview_experience    32.557185
        interview_difficulty    32.774194
        employees               11.888563
        revenue                 40.310850
        dtype: float64
```

Figure 9. Missing Values percentage

```
In [8]: company_reviews["reviews"].isnull().sum()
Out[8]: 1556
```

Figure 10. Sum of "review" Missing values

As seen earlier, the entire database contains 17,050 records, thus after deleted rows that don't have a review (this is the parameter that will be predicted). Now, the remaining data contains 15494 records.

```
In [9]: company_reviews = company_reviews.dropna(subset = ['reviews'])

In [10]: company_reviews.shape
Out[10]: (15494, 10)
```

Figure 11. Dropping "review" missing values

To get a clean dataset, it will need to convert the dictionary inside the rating variable to new columns. The dictionary is detailed rating in its five aspects: Compensation / Benefits, Culture, Job Security/ Advancement Management, and Work/life balance: all this data is embedded within one column as a dictionary.

```
In [11]: ratings = company_reviews.ratings.tolist()
         rating_dict = []
         for i in range(len(ratings)):
             rating_dict.append(eval(ratings[i]))


         keys = ['Compensation/Benefits', 'Job Security/Advancement', 'Management', 'Culture', 'Work/Life Balance']

         for key in keys:
             company_reviews[key] = [x.get(key, np.nan) for x in rating_dict]
             company_reviews[key] = pd.to_numeric(company_reviews[key], errors='coerce')

         # Removing the column ratings
         company_reviews = company_reviews.drop(['ratings'], axis=1)
```

Figure 12. Detailed "Rating" columns

Another column that is needed to convert too is the happiness column. First, It will convert the data from a dictionary to separate columns. Something to note is that the "happiness" rating has a lot of missing values (NaN). "happiness" has 13 different statements about belonging, energy, learning, purpose etc. The code below will bring those statements as columns:

```
In [12]: happiness=company_reviews.happiness.tolist()
         happiness_dict = []
         full_dict = {}

         for i in range(len(happiness)):
             new_entry = eval(happiness[i])
             key_list = list(new_entry.keys())
             new_key_list = ["happiness " + s for s in key_list]
             for count, key in enumerate(key_list):
                 new_entry[new_key_list[count]] = new_entry.pop(key)

             full_dict.update(new_entry)
             happiness_dict.append(new_entry)

         for key in full_dict.keys():
             company_reviews[key] = [x.get(key, np.nan) for x in happiness_dict]
             company_reviews[key] = pd.to_numeric(company_reviews[key], errors='coerce')

         company_reviews= company_reviews.drop(['happiness'], axis=1)
```

Figure 13. Detailed "Rating" columns

```
In [13]: company_reviews.head()
```

ure	Work/Life Balance	happiness Work Happiness Score	happiness Appreciation	happiness Purpose	happiness Learning	happiness Support	happiness Achievement	happiness Flexibility	happiness Trust	happiness Energy	happiness Inclusion	happiness Belonging	happiness Management	happiness Compensation
3.6	4.1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4.1	3.5	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4.3	4.2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
3.2	3.4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

Figure 14. Detailed "Happiness" columns

After converting the data from a dictionary to separate columns, there is a total of 26 columns and 15494 rows, as seen in the **figure.6** many of the columns still need pre-processing as is an object, such as the case of "review", "ceo_approval", and " "ceo_count" it will be converted to a numerical variable:

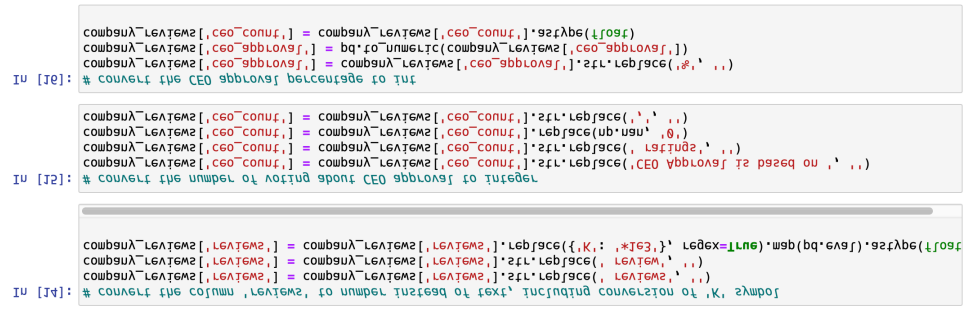


Figure 15. Transforming categorical variables to numerical

In [17]:
```
company_reviews.head()
```

Out[17]:

	rating	reviews	ceo_approval	ceo_count	interview_experience	interview_difficulty	employees	revenue	Compensation/Benefits	Security/Advancement	Job
1	3.7	21.0	NaN	0.0	Favorable	Easy	NaN	NaN	3.6	3.4	
2	4.0	23.0	88.0	17.0	Favorable	Medium	201 to 500	$5M to $25M (USD)	3.4	4.0	
4	4.1	19.0	NaN	0.0	Favorable	NaN	11 to 50	$5M to $25M (USD)	3.9	3.6	
5	3.4	437.0	57.0	199.0	Favorable	Medium	5,001 to 10,000	$1B to $5B (USD)	3.5	2.8	
6	3.5	45.0	67.0	27.0	Favorable	Medium	51 to 200	NaN	3.4	3.4	

In [18]:
```
company_reviews.tail()
```

Out[18]:

	rating	reviews	ceo_approval	ceo_count	interview_experience	interview_difficulty	employees	revenue	Compensation/Benefits	Security/Advancem
17045	3.2	92.0	69.0	51.0	Favorable	Medium	1,001 to 5,000	NaN	3.4	
17046	3.0	7.0	NaN	0.0	Average	Medium	201 to 500	NaN	4.0	
17047	2.5	11.0	NaN	0.0	NaN	NaN	NaN	NaN	2.5	
17048	3.9	122.0	88.0	33.0	Average	Easy	51 to 200	$25M to $100M (USD)	3.1	
17049	3.8	30.0	NaN	0.0	NaN	NaN	51 to 200	$5M to $25M (USD)	3.9	

Figure 16.Head and Tail of the data set pre-processed

In [19]:
```
company_reviews.shape
```

Out[19]:
```
(15494, 26)
```

In [20]:
```
company_reviews.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 15494 entries, 1 to 17049
Data columns (total 26 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   rating                          15494 non-null  float64
 1   reviews                         15494 non-null  float64
 2   ceo_approval                    10639 non-null  float64
 3   ceo_count                       15494 non-null  float64
 4   interview_experience            10896 non-null  object
 5   interview_difficulty            10858 non-null  object
 6   employees                       13732 non-null  object
 7   revenue                         9504 non-null   object
 8   Compensation/Benefits           15142 non-null  float64
 9   Job Security/Advancement        15141 non-null  float64
 10  Management                      15141 non-null  float64
 11  Culture                         15141 non-null  float64
 12  Work/Life Balance               15142 non-null  float64
 13  happiness Work Happiness Score  4030 non-null   float64
 14  happiness Appreciation          4030 non-null   float64
 15  happiness Purpose               4030 non-null   float64
 16  happiness Learning              4030 non-null   float64
 17  happiness Support               4030 non-null   float64
 18  happiness Achievement           4030 non-null   float64
 19  happiness Flexibility           4030 non-null   float64
 20  happiness Trust                 4030 non-null   float64
 21  happiness Energy                4030 non-null   float64
 22  happiness Inclusion             4030 non-null   float64
 23  happiness Belonging             4030 non-null   float64
 24  happiness Management            4030 non-null   float64
 25  happiness Compensation          4030 non-null   float64
dtypes: float64(22), object(4)
memory usage: 3.2+ MB
```

Figure 17. Pre-processed data shape and .info()

After the pre-processing, It can see that all the variables are in the correct data type.

Now the data is ready for the next phase, where an Exploratory Data Analysis will be realised to extract more information about the data and feature engineering for further training data application.

Exploratory Data Analysis

Data analysis utilising visual methods is called exploratory data analysis (EDA). With the use of statistical summaries and graphical representations it is used to identify trends and patterns or to verify assumptions (GeeksforGeeks, 2022). To get the statistics for the numerical values in the data frame, use the ".describe" function, which is responsible for generating descriptive statistics that summarise the central tendency, dispersion and shape of the dataset's distribution (McKinney, 2017).

```
In [21]: company_reviews.describe()
```

Out[21]:

	rating	reviews	ceo_approval	ceo_count	Compensation/Benefits	Job Security/Advancement	Management	Culture	Work/Life Balance	4(
count	15494.000000	15494.000000	10639.000000	15494.000000	15142.000000	15141.000000	15141.000000	15141.000000	15142.000000	
mean	3.520524	457.265070	69.961557	191.199045	3.342980	3.115950	3.170610	3.371164	3.450581	
std	0.606745	3112.588072	14.745167	1359.699482	0.566901	0.559117	0.602943	0.589443	0.563345	
min	1.000000	1.000000	6.000000	0.000000	1.000000	1.000000	1.000000	1.000000	1.000000	
25%	3.200000	16.000000	61.000000	0.000000	3.000000	2.800000	2.800000	3.000000	3.100000	
50%	3.500000	48.000000	72.000000	22.000000	3.400000	3.100000	3.100000	3.400000	3.500000	
75%	3.900000	179.000000	81.000000	79.000000	3.700000	3.400000	3.500000	3.700000	3.800000	
max	5.000000	215500.000000	100.000000	96314.000000	5.000000	5.000000	5.000000	5.000000	5.000000	

Figure 18. Statistical Summaries

The measures of dispersion evaluate how distributed the collected data are. They are standard deviation, variation and interquartile range; the code is taken from Pandas (The pandas development team, 2020).

As it is known, a data set is very spread out when the standard deviation value is high. That's the case of the variables "review" and "ceo_count", for example, which looks like they could have an error such as a human typing error when analysing it because the difference between the min and the max value is very high. For the "rating", for example, the standard deviation is lower than the mean and thus is likely to be a symmetrical distribution because the mean (3.520) is very close to the median (3.500). Also, the same behaviour in other numerical variables can be observed (see **figure.18***).*

A histogram of the numerical variables will be plotted to analyse its distribution, and a boxplot will be used to visualise its outliers.

```
In [22]: company_reviews.hist(bins = 15 , layout = (6,4), figsize = (20,17)  ,column = company_reviews.columns, grid =
         plt.show()
```

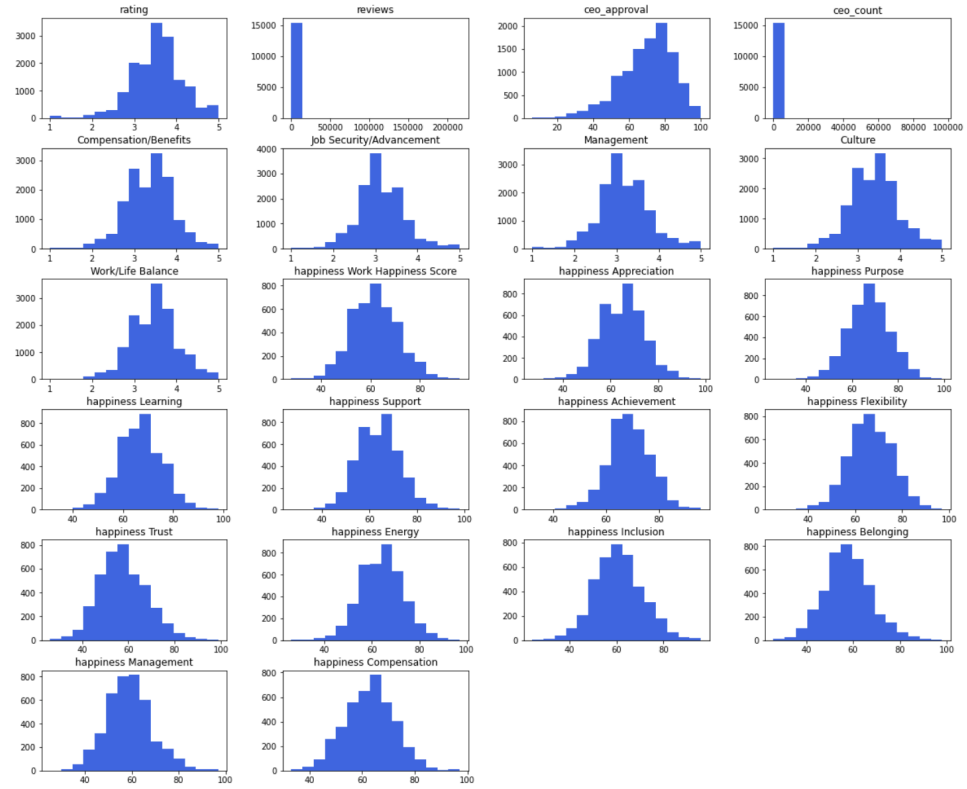


Figure 19. Numerical variables histogram

```
[ ] company_reviews.plot(subplots =True, kind = 'box', layout = (6,4), figsize = (26,20),colormap="BrBG")
    plt.subplots_adjust(wspace = 0.5, hspace= 0.5)
```

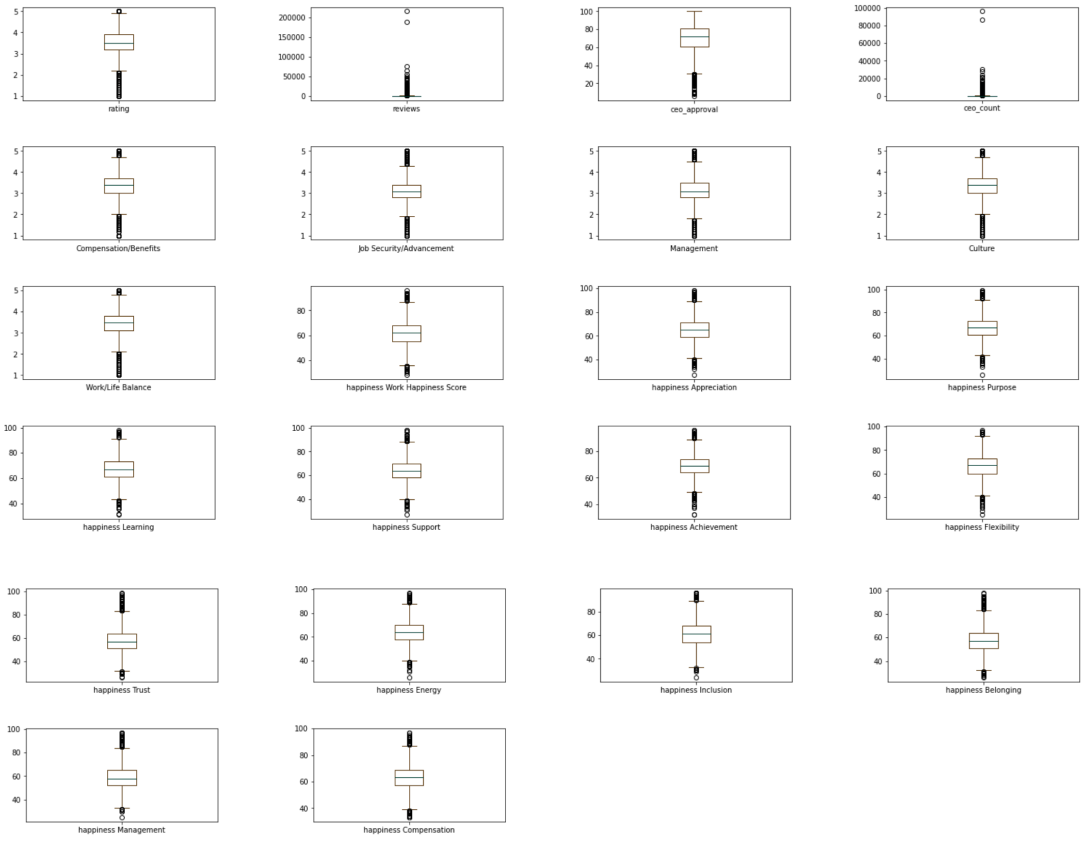


Figure 20. Numerical variables boxplot

From the boxplot, it can state the presence of outliers in all the variables. Outliers are data that is abnormal in its distribution. This means that they are reviews from employees who have a different perspective from most others concerning a particular variable.

Now, it will be analysed the statistical sum of the categorical variables:

```
In [24]: company_reviews.describe(include=object)
```

Out[24]:

	interview_experience	interview_difficulty	employees	revenue
count	10896	10858	13732	9504
unique	3	3	9	9
top	Favorable	Medium	1,001 to 5,000	$100M to 500M (USD)
freq	8597	5779	3009	1970

Figure 21. Categorical variables statistical summaries

The only descriptive statistics that can be extracted from the categorical variables is the mode, represented as "top" in the above statistical sum. It can be seen that the mode in the "interview_experience" variable is "Favorable", and the level of interview_dificulty is "Medium". In other words, the mode is the most repeated data. The frequency ladled as "freq" is the time this variable is repeated in the data.

```
[23] Columns= ['interview_experience', 'interview_difficulty','interview_difficulty','employees',"revenue"  ]
     for col in Columns:
         plt.figure(figsize=(10,7))
         sns.countplot(x=col,data=company_reviews,palette="Blues")
         plt.show()
```

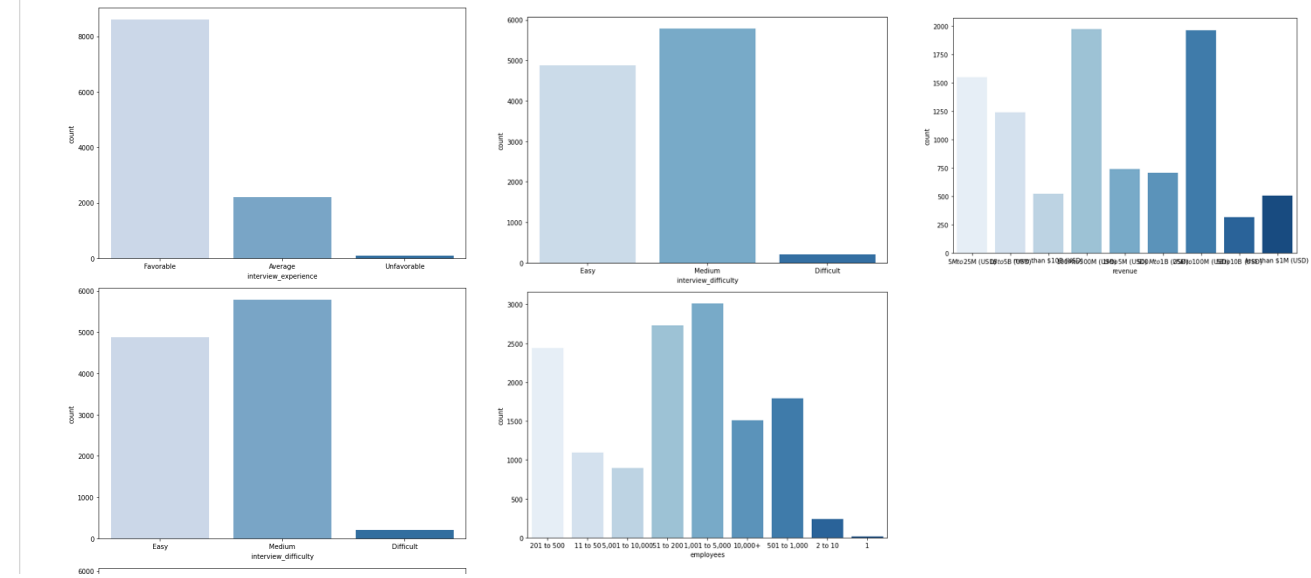


Figure 22. Categorical variables bar plot

From the plot above, the" "Interview_experience" has the "favourable" most repeated with 8597 data values. The most repeated number in the "employee" variable has 1001 to 5000 data values. The level of "interview_difficulty" is Medium, with almost 6000 thousand of employees having the same opinion.

Below is a heatmap showing the correlation between all the variables in the dataset. It can be seen that the "ceo_count" with the "review" variable is strongly correlated." "Culture" and "management" is highly correlated. The "happiness" variables are correlated with each other. And lastly, The "happiness Compensation" has a weak correlation with the "ceo_count".

```
In [26]: # shows the correlations between the values
plt.figure(figsize=(20, 9))

mask = np.triu(np.ones_like(company_reviews.corr(), dtype=np.bool))
heatmap = sns.heatmap(company_reviews.corr(), mask=mask, vmin=-1, vmax=1, annot=True, cmap="Blues")
heatmap.set_title('Triangle Correlation Heatmap', fontdict={'fontsize':18}, pad=16);
```

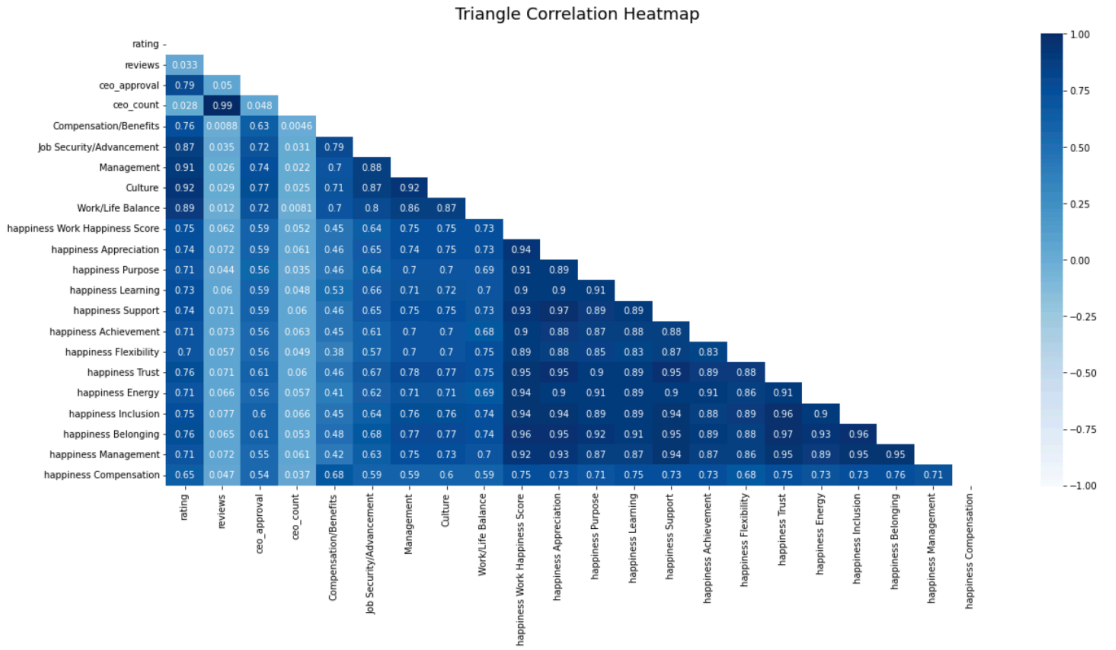


Figure 23. Dataset Heatmap

Handling missing Values

As already seen in the data previously, this data set contains some missing values that need to be imputed. The pandas library provides the easiest way to check for missing values. They are isnull() and not null () are functions that always return a binary value, in this case, false or true, indicating whether the value of the argument passed contain missing values (The Pandas Development Team).

```
In [27]: #check how many values are missing (NaN)
company_reviews.isnull().sum()
```

```
Out[27]: rating                              0
reviews                             0
ceo_approval                     4855
ceo_count                           0
interview_experience             4598
interview_difficulty             4636
employees                        1762
revenue                          5990
Compensation/Benefits             352
Job Security/Advancement          353
Management                        353
Culture                           353
Work/Life Balance                 352
happiness Work Happiness Score   11464
happiness Appreciation           11464
happiness Purpose                11464
happiness Learning               11464
happiness Support                11464
happiness Achievement            11464
happiness Flexibility            11464
happiness Trust                  11464
happiness Energy                 11464
happiness Inclusion              11464
happiness Belonging              11464
happiness Management             11464
happiness Compensation           11464
dtype: int64
```

Figure 24. Missing values

Below is the visualisation of them:

```
In [28]: #you can see the missing values
         msno.bar(company_reviews,color="royalblue")
```

Out[28]: <AxesSubplot:>

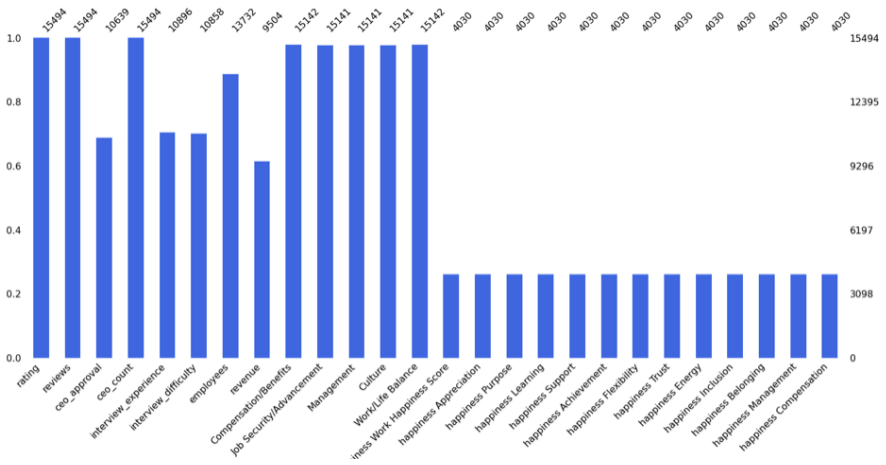


Figure 25.Missing values plot

Many values need to be added to this dataset. It will be checked the fraction of them in each variable:

```
In [29]: # We can also use the mean() method after isnull()
         # to obtain the fraction of missing values:

         company_reviews.isnull().mean()*100
```

Out[29]:

```
rating                            0.000000
reviews                           0.000000
ceo_approval                     31.334710
ceo_count                         0.000000
interview_experience             29.676004
interview_difficulty             29.921260
employees                        11.372144
revenue                          38.660127
Compensation/Benefits             2.271847
Job Security/Advancement          2.278301
Management                        2.278301
Culture                           2.278301
Work/Life Balance                 2.271847
happiness Work Happiness Score   73.989932
happiness Appreciation           73.989932
happiness Purpose                73.989932
happiness Learning               73.989932
happiness Support                73.989932
happiness Achievement            73.989932
happiness Flexibility            73.989932
happiness Trust                  73.989932
happiness Energy                 73.989932
happiness Inclusion              73.989932
happiness Belonging              73.989932
happiness Management             73.989932
happiness Compensation           73.989932
dtype: float64
```

Figure 26. Missing values fraction

Happiness has about 74% of the missing values; there needs to be more information to talk about this variable in the project. For that, it was decided to drop.

```
In [30]: df_new= company_reviews.drop(["happiness Work Happiness Score","happiness Appreciation","happiness Purpose","happin
```

Figure 27. Dropping "Happiness"

Imputation of Missing Values

For the categorical variables, it was decided to input with" "unknown" in order not to have fake data. The reason is that the frequency will change if input with the mode.

Categorical Variable

```
In [31]: df_new["interview_experience"] = df_new["interview_experience"].fillna(value='Unknown')

In [32]: df_new["interview_difficulty"] = df_new["interview_difficulty"].fillna(value='Unknown')

In [33]: df_new["employees"] = df_new["employees"].fillna(value='Unknown')

In [34]: df_new["revenue"] = df_new["revenue"].fillna(value='Unknown')
```

Figure 28. Imputation of categorical variables

For the numerical variables, It was decided to fill with the median. When the data is likely symmetrical, it is good to consider using the median value to replace them. And after visualisation of the variables with the imputation of the missing values:

Imput Numerical Missing Values with Median

```
In [35]: df_new.fillna(company_reviews.median(), inplace=True)

In [36]: df_new.isnull().sum()

Out[36]: rating                        0
         reviews                       0
         ceo_approval                  0
         ceo_count                     0
         interview_experience          0
         interview_difficulty          0
         employees                     0
         revenue                       0
         Compensation/Benefits         0
         Job Security/Advancement      0
         Management                    0
         Culture                       0
         Work/Life Balance             0
         dtype: int64
```

Figure 29. imputation of missing values

Below is the plot of the distribution of each variable after the imputation of the missing values that, after input with the median, do not have a considerable change (see **Figure 19**).

```
In [37]: df_new.hist(bins = 15 , layout = (6,4), figsize = (20,17)  ,column = df_new.columns, grid = False,color="royalblue"
         plt.show()
```

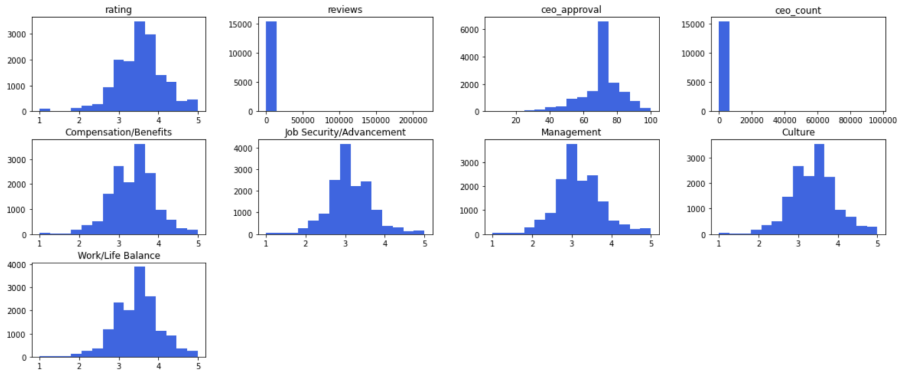


Figure 30. The plot of the new distribution

Feature Engineering

In feature engineering, variables from raw data are extracted and transformed to be used as training and prediction features (Sreemany, 2021).

Encoding

Encoding generally converts data from one form (for example: categorical) to another (numerical)(Wijaya, 2021).

Employers

The data about the company size (the number of employees) are strings giving the range of number of employees. we will convert it to ordinal data in accordance with the following: 0- Unknown

1 - 1 employee

2 - 2 to 10 employees

3 - 11 to 50 employees

4 - 50 to 200 employees

5 - 201 to 500 employees

6 - 501 to 1000 employees

7 - 1001 to 5000 employees

8 - 5001 to 10000 employees

9 - 10000+ employees

```
In [38]: pd.unique(df_new['employees'])
scale_mapper = {"Unknown":0,"1":1, "2 to 10":2, "11 to 50":3, "51 to 200":4,"201 to 500":5,"501 to 1,000":6,"1,001
df_new['employees']=df_new['employees'].replace(scale_mapper)
```

Figure 31. Encoding" the "employee" variable

revenue

The revenue of the companies are strings giving the range of the revenue. We will convert it to ordinal data in accordance with the following:

0 - Unknown

1 - 'less than $1M (USD)'

2 - 1Mto 5M (USD)

3 - 5Mto 25M (USD)

4 - 25Mto 100M (USD)

5 - 100Mto 500M (USD)

6 - 500Mto 1B (USD)

7 - 1Bto 5B (USD)

8 - 5Bto 10B (USD)

9 - more than $10B (USD)

```
In [39]: scale_mapper = {"Unknown":0,"less than $1M (USD)":1, "$1M to $5M (USD)":2, "$5M to $25M (USD)":3, "$25M to $100M (U
df_new['revenue']=df_new['revenue'].replace(scale_mapper)
```

Figure 32. Encoding" "revenue" variable

Interview experience

0.- Unknown

1.- Favorable

2.-Average

3.-Unfavorabel

```
In [40]: pd.unique(df_new["interview_experience"])
Out[40]: array(['Favorable', 'Unknown', 'Average', 'Unfavorable'], dtype=object)
```

Figure 33. Encoding "Interview Experience"

Interview difficulty

0.- Unknown

1.- Easy

2.- Medium

3.- Difficult

```
In [42]: pd.unique(df_new["interview_difficulty"])
Out[42]: array(['Easy', 'Medium', 'Unknown', 'Difficult'], dtype=object)

In [43]: scale_mapper = {"Unknown":0,"Easy":1, "Medium":2, "Difficult":3}
         df_new["interview_difficulty"]=df_new["interview_difficulty"].replace(scale_mapper)
```

Figure 34. Encoding interview difficulty

```
In [44]: df_new.head(15)
```

Out[44]:

	rating	reviews	ceo_approval	ceo_count	interview_experience	interview_difficulty	employees	revenue	Compensation/Benefits	Job Security/Advancement	Mana
1	3.7	21.0	72.0	0.0	1	1	0	0	3.6	3.4	
2	4.0	23.0	88.0	17.0	1	2	5	3	3.4	4.0	
4	4.1	19.0	72.0	0.0	1	0	3	3	3.9	3.6	
5	3.4	437.0	57.0	199.0	1	2	8	7	3.5	2.8	
6	3.5	45.0	67.0	27.0	1	2	4	0	3.4	3.4	
7	3.5	367.0	78.0	167.0	1	2	7	7	3.3	3.2	
8	2.8	6.0	72.0	0.0	0	0	3	0	2.8	2.7	
9	4.2	803.0	83.0	275.0	1	2	8	7	3.7	3.6	
10	3.4	11.0	72.0	0.0	1	0	0	0	3.1	3.0	
11	4.1	16.0	72.0	0.0	0	0	5	3	4.1	3.7	
12	3.6	117.0	79.0	48.0	1	2	0	0	3.6	3.1	
13	3.2	50.0	70.0	20.0	1	2	5	0	3.1	3.0	
16	4.1	501.0	89.0	181.0	1	2	8	7	3.8	3.7	
17	4.0	183.0	76.0	84.0	1	2	8	9	3.7	3.4	
19	3.3	54.0	67.0	21.0	1	2	5	5	3.3	3.2	

Figure 35. New data set after encoding

Now the data is ready to be scaled after applying the machine learning models.

Scaling

Now the data have to be scaled to reduce the variance of the data set. Scaling the data makes it easy for a model to learn and understand the problem. But before, it will be separated the dependent from the independent variables:

```
In [45]: #Independent variables
         X= df_new.drop("rating", axis=1).values
         X
Out[45]: array([[ 21. ,  72. ,   0. , ...,   3.4,   3.6,   4.1],
                [ 23. ,  88. ,  17. , ...,   4. ,   4.1,   3.5],
                [ 19. ,  72. ,   0. , ...,   4.3,   4.3,   4.2],
                ...,
                [ 11. ,  72. ,   0. , ...,   2.3,   2.4,   2.8],
                [122. ,  88. ,  33. , ...,   3.6,   3.6,   3.8],
                [ 30. ,  72. ,   0. , ...,   3.9,   4. ,   4. ]])

In [46]: #Dependent variables
         y= df_new["rating"].values
         y
Out[46]: array([3.7, 4. , 4.1, ..., 2.5, 3.9, 3.8])
```

Figure 36. Separating dependent from independent variables

The code below will scale the data set using the StandardScaler that standardises the variables by removing the mean and scaling to unit variance (Brownlee, 2020).

```
In [47]: from sklearn.preprocessing import StandardScaler

         scaler= StandardScaler()
         scaler.fit(X)
         X_scaled = scaler.transform(X)
         X_scaled

Out[47]: array([[-0.14016604,  0.1142198 , -0.14062314, ...,  0.38751182,
                  0.39159766,  1.16403374],
                [-0.13952347,  1.41986578, -0.12811998, ...,  1.39403996,
                  1.24969208,  0.08671702],
                [-0.14080861,  0.1142198 , -0.14062314, ...,  1.89730403,
                  1.59292985,  1.34358653],
                ...,
                [-0.14337891,  0.1142198 , -0.14062314, ..., -1.45778977,
                 -1.66782897, -1.17015249],
                [-0.10771611,  1.41986578, -0.11635229, ...,  0.7230212 ,
                  0.39159766,  0.62537538],
                [-0.13727446,  0.1142198 , -0.14062314, ...,  1.22628527,
                  1.0780732 ,  0.98448096]])

In [48]: df_scaled = pd.DataFrame(X_scaled)

In [49]: df_scaled_standar=pd.DataFrame(X_scaled,columns=["reviews","ceo_approval","ceo_count","interview_experience","inter
                                                          "Compensation/Benefits","Job Security/Advancement","Management","C
         df_scaled_standar.head()
```

Out[49]:

	reviews	ceo_approval	ceo_count	interview_experience	interview_difficulty	employees	revenue	Compensation/Benefits	Job Security/Advancement	Manageme
0	-0.140166	0.114220	-0.140623	0.214093	-0.118398	-2.026755	-1.034619	0.456268	0.514591	0.3875
1	-0.139523	1.419866	-0.128120	0.214093	1.064359	-0.051578	0.047808	0.099425	1.600175	1.3940
2	-0.140809	0.114220	-0.140623	0.214093	-1.301154	-0.841649	0.047808	0.991533	0.876453	1.8973
3	-0.006511	-1.109823	0.005737	0.214093	1.064359	1.133527	1.491044	0.277847	-0.570993	-0.4512
4	-0.132455	-0.293795	-0.120765	0.214093	1.064359	-0.446614	-1.034619	0.099425	0.514591	-0.1157

Figure 37. Data set scaled

Now the data is ready for the application of Machine Learning Models.

Modelling

Two supervised machine learning models were applied, Linear regression and Random Forest. These models can be used for prediction problems. But first, it is necessary to split the data into training and Test.

Training Data

The train_test_split function splits the dataset into training and test sets (Pramoditha, 2022). The test set will contain 20% of the data, and the training set will hold the remaining 80%. Here we are trying to predict the column rating, which is assigned, and the other columns, which represent our independent variable, are assignee' 'X'.

Trainig Data

```
In [50]: from sklearn.model_selection import train_test_split


         X_train, X_test, y_train, y_test = train_test_split(X, y,
                                                             test_size = 0.2,
                                                             random_state = 0,
                                                             stratify = y)

In [51]: X_train.shape, X_test.shape, y_train.shape, y_test.shape

Out[51]: ((12395, 12), (3099, 12), (12395,), (3099,))
```

Figure 38. train and Test split

Find below the score from the coefficient of determination "r" 2" regression applied in each six machine learning models. By observing them, the linear regression model represents the highest coefficient of 0.771.

Prediction Models

```
In [52]: #import models from scikit learn module:
         from sklearn.linear_model import LinearRegression
         from sklearn.neighbors import KNeighborsRegressor
         from sklearn.tree import DecisionTreeRegressor
         from sklearn.ensemble import RandomForestRegressor
         from sklearn.svm import SVR
         from sklearn.ensemble import GradientBoostingRegressor

         #Cross validation
         from sklearn.model_selection import cross_val_score

In [53]: models = []
         models.append(('LR', LinearRegression()))
         models.append(('KNN', KNeighborsRegressor()))
         models.append(('DTR', DecisionTreeRegressor()))
         models.append(('RFR', RandomForestRegressor()))
         models.append(('SVR', SVR()))
         models.append(('GBR', GradientBoostingRegressor()))

In [54]: results = []
         names = []
         for name, model in models:
             cv_result = cross_val_score (model, X_train, y_train, cv=10, scoring= "r2")
             results.append(cv_result)
             names.append(name)
             print('%s: %f (%f)' % (name, cv_result.mean(), cv_result.std()))



         LR: 0.770997 (0.025435)
         KNN: 0.552296 (0.042735)
         DTR: 0.652245 (0.037632)
         RFR: 0.743283 (0.028112)
         SVR: 0.163011 (0.018314)
         GBR: 0.767791 (0.027097)
```

Figure 39. Prediction Models

Linear Regression Model

The Linear Regression model was created using scikit-learn's linear regression by training and testing the set using the appropriate method. The model is then used to make predictions on the Test.

The scikit-learn's r2_score function is used to evaluate the performance of a Linear Regression model on the test set, and an r2_score of 0.771 is obtained. The r2_score function calculates the coefficient of determination, which measures the degree to which the input features can explain the target variable.

Linear Regresion

In [55]:
```
#CREAT THE MODEL
#Create a LogisticRegression
LR = LinearRegression()

#Train the model using the training sets
LR.fit(X_train, y_train)

#Predict the response for test dataset
y_pred = LR.predict(X_test)
```

In [56]:
```
# EVALUATING MODEL
#Import scikit-learn metrics module for accuracy calculation
from sklearn.metrics import r2_score

# Model Accuracy, how often is the classifier correct?
LR_r2=r2_score(y_test, y_pred)
print (LR_r2)
```

```
0.7718277428998205
```

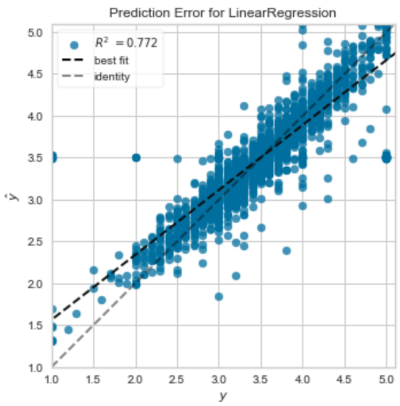
In [57]:
```
# Instantiate the linear model and visualizer
from yellowbrick.regressor import PredictionError
# Instantiate the linear model and visualizer
model = LR
visualizer = PredictionError(LR)

visualizer.fit(X_train, y_train)  # Fit the training data to the visualizer
visualizer.score(X_test, y_test)  # Evaluate the model on the test data
visualizer.show()                 # Finalize and render the figure
```

Figure 40. Linear Regression Model

In [57]:
```
# Instantiate the linear model and visualizer
from yellowbrick.regressor import PredictionError
# Instantiate the linear model and visualizer
model = LR
visualizer = PredictionError(LR)

visualizer.fit(X_train, y_train)  # Fit the training data to the visualizer
visualizer.score(X_test, y_test)  # Evaluate the model on the test data
visualizer.show()                 # Finalize and render the figure
```



Out[57]: <AxesSubplot:title={'center':'Prediction Error for LinearRegression'}, xlabel='y', ylabel='$\\hat{y}$'>

Figure 41. Yellow Brick Regressor: Linear Regression

The above plot demonstrates the Yellow brick's Prediction Error visualiser to evaluate the performance of a Linear Regression model. The Prediction Error visualiser plots the actual target values against the predicted values and includes a line of best fit and a 45-degree reference line to help visualise the error between the predictions and the true values.

Cross Validation

A cross-validation score of 0.77 is obtained. It can be seen that the r2 scores of the two different models: the linear regression model and the linear regression model with cross-validation. Both models have similar r2 scores, as the bar for both models is approximately the same height.

```
In [58]: # CROSS VALIDATION
         LR_cvs = cross_val_score(estimator = LR, X = X_train, y = y_train,scoring='r2', cv = 10).mean()
         print(LR_cvs.mean())

         0.7709974383031973

In [59]: R2 = pd.DataFrame({"R2":[0.7718277428998205,0.7709974383031973], "Models":["LinearRegression",
                    "LinearRegression with Cross Validation"]})

         g = sns.barplot("R2", "Models", data =R2,palette ="Blues")
         g.set_xlabel("Mean r2")
         g.set_title("R2 Scores")

Out[59]: Text(0.5, 1.0, 'R2 Scores')
```

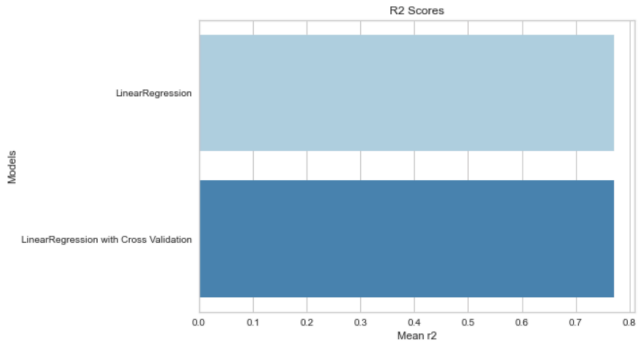


Figure 42. Cross-validation: Linear Regression

Random Forest

Random Forest is a Supervised Machine Learning Algorithm used in prediction problems. It works by building decision trees on different samples, and their majority occurrence is taken for regression (Sruthi, 2021).

Random Forest

```
In [60]: #CREAT THE MODEL
         #Create a LogisticRegression
         from sklearn.ensemble import RandomForestRegressor
         RFR = RandomForestRegressor()

         #Train the model using the training sets
         RFR.fit(X_train, y_train)

         #Predict the response for test dataset
         y_pred = RFR.predict(X_test)

In [61]: # EVALUATING MODEL
         #Import scikit-learn metrics module for accuracy calculation
         from sklearn.metrics import r2_score

         # Model Accuracy, how often is the classifier correct?
         RFR_r2= r2_score(y_test, y_pred)
         print (RFR_r2)

         0.7581560093860794
```

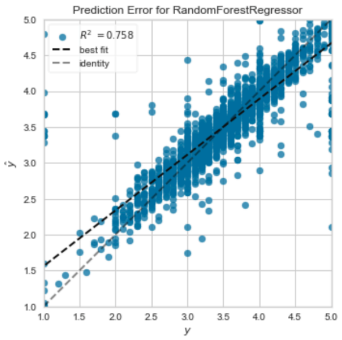
Figure 43. Random Forest Model

Above, a random forest regressor model is created and trained. It applied the r2_score function from the Scikit-learn. Metrics module in evaluating the performance of a random forest regressor model on a test dataset, an r2_score of 0.755 is obtained. A high R2 score indicates that the model can predict the target variable (Nima, 2022).

Above, the Prediction Error visualiser from the Yellow brick library is used to evaluate the performance of a random forest regressor model.

```
In [62]: # Instantiate the linear model and visualizer
         from yellowbrick.regressor import PredictionError
         # Instantiate the linear model and visualizer
         model = RFR
         visualizer = PredictionError(RFR)

         visualizer.fit(X_train, y_train)  # Fit the training data to the visualizer
         visualizer.score(X_test, y_test)  # Evaluate the model on the test data
         visualizer.show()                 # Finalize and render the figure
```




```
Out[62]: <AxesSubplot:title={'center':'Prediction Error for RandomForestRegressor'}, xlabel='$y$', ylabel='$\\hat{y}$'>
```

Figure 44. Yellow Brick Regressor: Random Forest

The cross-validation below is used to evaluate the performance of a random forest regressor model. A cross-validation score of 0.74 is obtained. It can be seen that the r2 scores of two different models: a random forest regressor model and a random forest regressor model with cross-validation, appear that both models have similar r2 scores, as the bar for both models is approximately the same height.

```
In [63]: # CROSS VALIDATION
         RFR_cvs = cross_val_score(estimator = RFR, X = X_train, y = y_train,scoring='r2', cv = 10).mean()
         print(RFR_cvs.mean())

         0.7439786652488398

In [64]: R2 = pd.DataFrame({"R2":[0.759501635139176,0.7435599435281295], "Models":["RandomForestRegressor",
                    "RandomForestRegressor with Cross Validation"]})

         g = sns.barplot("R2", "Models", data =R2,palette ="Blues")
         g.set_xlabel("Mean r2")
         g.set_title("R2 Scores")

Out[64]: Text(0.5, 1.0, 'R2 Scores')
```

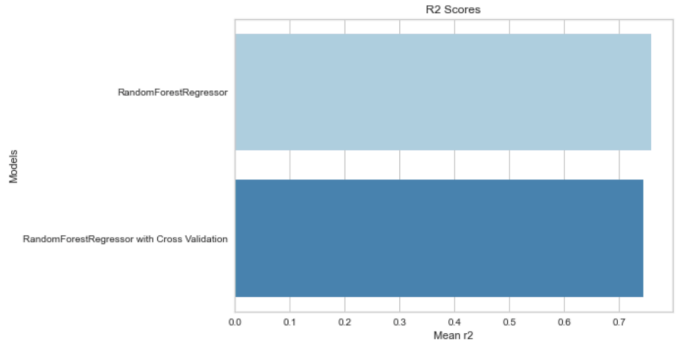


Figure 45. Cross Validation: Random Forest

Evaluation

After obtaining the results of the ML models, it was found that linear regression has a better performance, which is why it was got to extract the essential characteristic.

It will use the **Feature Importance** method to obtain the most critical factor. This method consists of calculating a score for all input features for a given model; practices represent the "importance" of each feature (Shin, 2021)

```
In [65]: # linear regression feature importance
         from sklearn.datasets import make_regression
         from sklearn.linear_model import LinearRegression
         from matplotlib import pyplot
         # define the model
         model = LinearRegression()
         # fit the model
         model.fit(X, y)
         # get importance
         importance = model.coef_

         # summarize feature importance
         for i,v in enumerate(importance):
          print('Feature: %0d, Score: %.5f' % (i,v))

         # plot feature importance
         plt.title('Feature Importance')
         plt.xlabel('Features')
         plt.ylabel('coef')
         pyplot.bar([x for x in range(len(importance))], importance)
         pyplot.show()
```

```
Feature: 0, Score: 0.00000
Feature: 1, Score: 0.00257
Feature: 2, Score: -0.00001
Feature: 3, Score: -0.03106
Feature: 4, Score: 0.02665
Feature: 5, Score: 0.00197
Feature: 6, Score: 0.00499
Feature: 7, Score: 0.08141
Feature: 8, Score: 0.10686
Feature: 9, Score: 0.23061
Feature: 10, Score: 0.30378
Feature: 11, Score: 0.21968
```

Figure 46. Feature importance code

A higher score means that the specific variable will significantly affect the model used to predict a particular variable. From the result, variable: 10 has the highest importance, with a Score of 0.30378, followed by Feature: 9, with a Score: of 0.23061. **Figure 46,** it can see the meaning of each feature. Note that the figure below does not rank the most critical factors.

Feature 0	reviews
Feature 1	ceo_approval
Feature 2	ceo_count
Feature 3	interview_experience
Feature 4	interview_difficulty
Feature 5	employees
Feature 6	revenue
Feature 7	Compensation/Benefits
Feature 8	Job Security/Advancement
Feature 9	Management
Feature 10	Culture
Feature 11	Work/Life Balance

Figure 47.Meaning of each variable

Accordingly, with the analysis done in this project and the results from the Feature Importance method, it can be concluded that the most important factor influencing an employee when analysing a company review is Culture, Management, and Work/life balance.

.

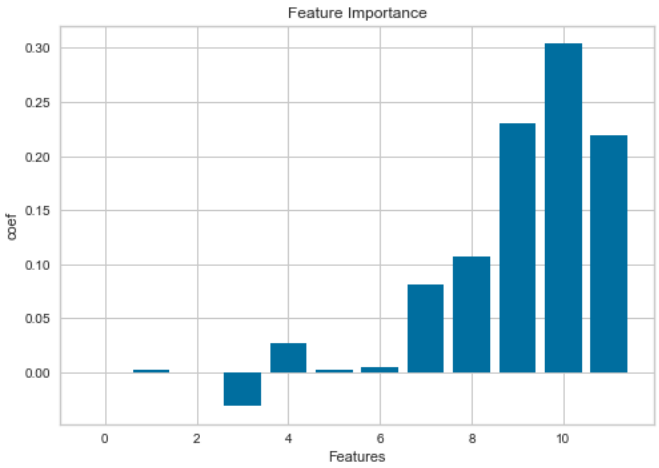


Figure 48. Most important factors

Deployment

This project represented a challenge in different aspects. It was necessary to reorganise the data by identifying irrelevant variables to this analysis. During the cleaning process, most data were categorical. It was disappointing that Happiness had so much missing data; it was impossible to get accurate answers from it, and it had to be dropped.

Before applying EDA and developing the machine-learning model, business understanding and data understanding were crucial to understanding the dataset domain. It was decided to use six machine learning models based on their r2 score. The models to work with were Linear regression and Random Forest.

Even though the results were good, it is essential to know that models can continuously be improved. There is a significant opportunity to find different techniques that will give better results. There is a chance to experiment with new models such as Lasso Regression and Gaussian Process.

Conclusion

Through all the stages of this project, from Business Understanding and after applying the feature importance, it was found that the most critical factor for the employees was the "culture". As it is known, when a company's environment matches an employee's values and ideas, an employee who is happy and comfortable embraces a strong workplace culture and does not have many reasons to leave. (Bloznalis, 2022)

The second factor is "management", which shows how important it is for companies to have good leaders, mentors, and someone who inspires the rest of the team. Effective managers help people stay motivated to do their best work. Understanding the “company reviews” data set gives a better idea of how this can bring opportunities to companies, employees, and their development.

This type of analysis can beneficiate or not the image of an entire company and identify new opportunities to improve internal efficiencies and optimise processes.

Extra Contents

Roles and responsibilities

A multidisciplinary team formed this research to meet the established goals and perform well even within diversities founded during its development. The roles were described as follows:

Team Member	Role	Responsibilities
Daniela Daia	Project Manager /Data Researcher	- Gathering data from Kaggle in the project plan; - Producing questions to analyse the data set - Managing the development team meetings; - Managing tasks in Trello; - Development of Abstract and Introduction in the report - Development of the Data Understanding and EDA report - Development of the PPT desing presentation -Summarize content in the presentation - Revising and closeing the report - Revising, organisinf and closing the PPT file - Revising final code and report compilations to ensure clarity in the reports.
Ana Isabel	Data Researcher/ Data Scientist	- Gathering data from Kaggle in the project plan; - Producing questions to analyse the data set - Project background development - Development of the data understanding codes - Development of the EDA codes - Development of the Modelimg codes -Development the results report. -Development the deployment report. -Summarize content in the presentation - Revising final code and report compilations to ensure clarity in the reports.
Magdalena	Data Researcher/ Data Analyst	- Gathering data from Kaggle in the project plan; - Producing questions to analyse the data set - Developing of the EDA codes - Test solutions to validade objectives; -Summarize content in the presentation - Modeling Undestanding report
Roxana	Data Researcher/Business Analyst	- Gathering data from Kaggle in the project plan; - Producing questions to analyse the data set - Developmento of the data dictionary -Summarize content in the presentation - Development of the Business Understanding report
K. Carolina A. Minon	Data Researcher/Data Analyst	- Gathering data from Kaggle in the project plan; - Producing questions to analyse the data set -Development the evaluation - Development of EDA code -Creation of Insights in the presentation, Summarize content in the Presentation -Support with the final report evaluation to ensure clarity in the reports - Development of the conclusion

Figure 49. Roles and Responsibilities

Team Project Management

Trello is a straightforward, and simple-to-use collaboration application that allows us to organise and track the project on the board; the team decided to use it for project management. It was a terrific tool to help to achieve the assignment on time as necessary, as seen in our board below. For better visualisation access: [TrelloTeam2Board.]

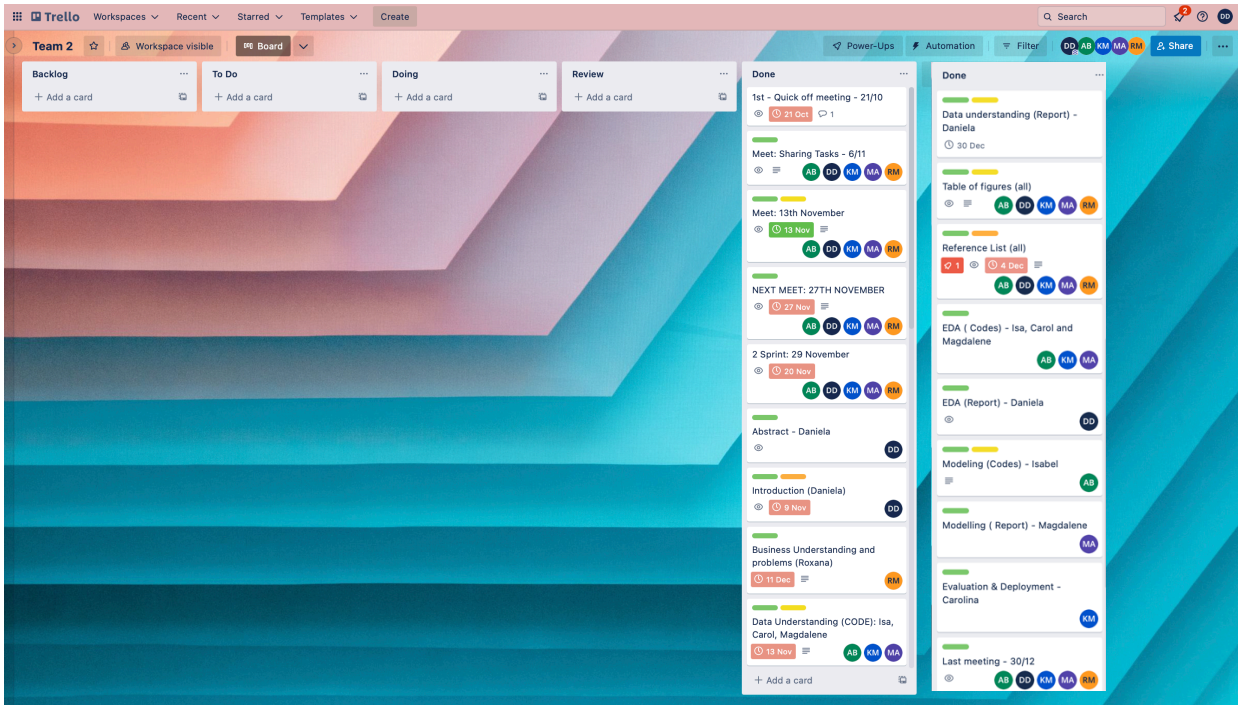


Figure 50. Project management

Individual Reflection Report

Roxana – Student ID: 2022175

The whole project was super challenging, with developing business understanding being the main questioning factor of the companies; it was rewarding and interesting. Throughout the project, I discovered all the steps to have the exact solution!

One of my biggest challenges was communication with the group because my English is not so good, it makes me ashamed to communicate, but this was solved because a member of the team spoke my language and it became easier to carry out the project with mastery.

The team was very important in solving the problem. We had some time constraints because we had other projects to finish, but they all did an excellent job. I am grateful to everyone on the team because everyone had great participation!

So I'm happy to conclude a project as well prepared as ours!

Thanks to the team!

Isabel Nieves – Student ID: 2022455

This personal reflection report provides me with an overview of my growth problems as a team member in the strategic thinking project. I used The reflective cycle model by Gibbs (1988) to share my thoughts.

Description

At the beginning of the project, I needed to organise my time to carry out my responsibilities and the tasks assigned to me in the team. I contributed to different stages of the project; I invested most of the time in understanding the data set and each of the variables we had; carrying out the pre-processing of the data before the modelling part was one of the biggest challenges since I did not have the necessary experience, which helped me to understand better. After applying the different Machine learning models, it was required to identify the metrics for the regression problems. The last part of our project was to find the most important factors for employees when looking for a company. For this part, we had to use different methods and find the one that best suited our needs.

Feelings

I have experienced a growing sense of stress, frustration and worry during the project. I feel frustrated that I spent much time understanding each of the variables but even more, time pre-processing each. I feel incompetent for not being able to share my results with my colleagues quickly. The stress came when communication in the team needed improvement and the project deadline was approaching.

Evaluation

The subsequent study of the mistakes made helped me understand that I need to develop my communication skills with the team and a study habit that will allow me to obtain more knowledge, not just the knowledge I acquire in class.

Another important aspect is to improve the project management because it turned out differently than we expected, having some problems during the execution of the project.

Analysis

To succeed in my personal development, I had to appraise my strengths, weaknesses, and opportunities for future projects using the SWOT Analysis.

Strengths	Opportunities
• I have a good approach with people. • I have a good theoretical understanding about machine learning. • I am good at working as a team.	• Develop better communication skills. • Develop better project management skills.
Weaknesses	**Threats**
• I have no experience with regression models. • English as a second language for communication.	• Practical examples about differents machine learning problems. • Better understanding about differents metrics such as accuracy, r2, mse,etc. • improve communication channels

Conclusions

This experience is precious because it taught me that teamwork is the fundamental piece for the success of any project and, in turn, that communication plays an essential role in achieving objectives.

Action Plan

My actions for future situations may be:

- Prioritise each of the activities according to the delivery date.
- Create an adequate communication channel between team members.
- Develop the ability of the project leader to be able to take responsibility in the team.
- Understand the theory and have more experience working with different ML models.

Karla Carolina A. Miñon – Student ID: 2022461

Finding a good Data Set is challenging, and I spent a certain number of hours comparing datasets and sharing them with the team. Many different ideas came to light, but I ultimately decided that the goal was finding something more relevant but interesting to us at the same time. After comparing a few datasets, “Company Reviews” was the one that kept my attention the most. I’ve had an increasing curiosity for Social Sciences, and when I thought we could measure and predict the happiness of employees, I was fascinated with that. As a communication graduate, it’s been always interested in the ways people tend to behave in certain contexts and how there is always a way to modify and improve those. Unfortunately, it did not work as well as we thought, so we decided to explore the data's various variables. Sometimes it is necessary to take a different route as you go deeper into the data.

I had an idea about the subject as I worked with HR before and how important it Is for them to collect reviews for different departments and purposes. As we started to play and see through the data, we had big code challenge cleaning and reorganizing the data and imputing values. I must admit that some members make it easier to understand as they explain this to the team to make the information more digestible. I contributed to developing the EDA of the final variables chosen to understand the data better and answer the question: What are the most important factors for employees in a company? I always found the EDA one of the most interesting things. I enjoyed it when we found correlations, and the information tended to make more sense to all of us.

CRISP-DM helped the team to make management workflow better. As the project progressed, there were numerous modifications to the plan based on unexpected events, ideas, and a handful of other reasons. There were still more ways in which members could innovate, specifically in Project management and its numerous issues, such as putting the team under pressure. The final solution was to improve communication and mark limits between the members. The creation of this project brings me the opportunity to see how this type of analysis can impact a company, a culture, or a society. The magic of seeing what important today is might be something other than tomorrow. How values change through the years in societies and the decisions that will be made based on that.

The Deployment helped me to clarify the idea I had about what Data Analysis is, is not just answer a question because the answer has power. The answer can be people being fired, a company losing money, a new investment or a development of a new production line in a factory. I believe that the improvement of my skills, particularly in ML, will be extremely helpful in the future, but I’m glad I had a team that supported me. I also know that there is a big opportunity to develop as I now understand my boundaries and limits, so it will be easier for me to set attainable goals and deadlines in future projects.

Awaritefe Magdalene – Student ID:2022151

We were tasked as a team to solve a Machine Learning Problem of our choice and to present findings and reports. At the beginning, it was a difficult task as it was my first attempt to go in the wild, pick and solve a machine learning problem and organise a report on the finding.

After a few moments of deep thinking, I decided to go online and search for resources and materials that handled Machine Learning problems and Build a Report to guide me. After spending many hours and days, I saw a dataset on Kaggle that I was comfortable with about predicting the inflationary rate. I picked that dataset because, having a First Degree in Economics, I find it easy to understand the Dataset. But after having a meeting with the team, we chose another dataset, and as such, we all decided to work with the Company Review Dataset. After that, the task was shared among team members, I and a team member were responsible for the EDA and Machine Learning Model.

We decided to work on it independently so that we would combine our findings at the end of the process. I worked on data cleaning and EDA, and in the process of working, I identified that the dataset had lots of missing values, and some of the missing values were in the target variable, that's why the missing values had to be dropped. The dataset also had a mix of categorical and numerical variables, which made it challenging to work on and as such, Feature Engineering had to be applied in other to convert the categorical variables to numerical variables. Also, some of the variables had their own dictionary, so I had to research python codes to expand the variables, and I concluded my part of the analysis.

In the next meeting as a team, all our individual work was looked into, corrections were made, and from the meeting, I realised that I had to arrange my codes in an orderly manner for easy understanding. I also wrote a report on the modelling explaining the Machine Learning Modelling that was used in the project, the predictions of the model and the results, which I had to do research on their usage and interpretation of their result.

This process has not only been draining but also exciting to me. I have gained a lot of insights on the processes involved in cleaning, exploring, analysing, interpreting the data and also Report Presentation.

Daniela Daia – Student ID: 2017207

This individual reflection aims to summarize what I have done in the project, the challenges I faced as a group and as an individual, the lessons learned and the outcomes for the next group project.

Contribution to the project tasks:

Since the first day the project was released, I created the WhatsApp group and set our first meeting with the objective that everyone could bring at least 1 data to discuss so we could choose one data. That was an intense part as we all have different interests. We took quite a lot of time to decide to work with the "Company Reviews". Meantime, I created a report structure to be easy to understand the project as a whole, and everyone could choose what part would like to contribute. After the tasks were chosen, I created a Trello Board and inserted not only all the agreed tasks with their deadlines but also tracked our meetings and checked the progress of our work.

With the data chosen and the question decided, I started to do my tasks which were to build the abstract and the introduction. When the first part of the codes was done, I started to build the Data Understanding and the Exploratory Data Analyses report.

When I finished, I focused on the design of the ppt and found the best way to organise the presentation by studying the PowerPoint tools to facilitate as each member is remote. Furthermore, I help my colleagues, such as in the Business Understanding report and reviewing and closing all the report grammar, referencing and closing the PPT file.

Project Challenges faced:

- To find a data set
- To write a concise and clear report
- To work on the ppt presentation.
- To put the pieces and bits together

Team Challenges faced:

As a team with different backgrounds and interests, working as a group is certainly a challenge. However, when everyone wants to reach the same goal, in our case, not only learning but marks, should be a great healthy competition, even among the diversity, that brings even more insights to the project overall.

As the deadline was approaching and there was a lack of communication in our group, I decided to play the role of the Project Manager to certify that we were all on the same page with the deadline. And I knew that from that, it would be challenging to ask people what they have to do on key dates.

Firstly, I certify that all the information was clear to everyone in our Whatsapp group, all the support materials were available in the group drive, and I was available for help giving all the

support necessary. As noted, some members did not show up even in our last sprint session. The lack of compromise with some group members will certainly end up in stressful times for everyone in the last three days, resulting in extra work for the reviewer.
However, I must highlight that Isabel is a great team player and we worked well together.

Next steps

The team could have a meeting to review and share all the lessons learnt, the things that happened and were not satisfactory and the things that were satisfactory. This way, we could put differences aside and grow as a team and individuals.
On my behalf, I have to practice my patience and put my efforts into working under pressure without losing good energy, as in the real world things are a even more complicated.

Reference List

Brownlee, J. (2020). Data Preparation for Machine Learning: Data Cleaning, Feature Selection and Data Transforms in Python. Jason Brownlee.

GeeksforGeeks.org (2022). What is Exploratory Data Analysis? Available at: https://www.geeksforgeeks.org/what-is-exploratory-data-analysis/ (Accessed 27 December 2022).

Google Colaboratory (2022) Tutorials. Available at: https://research.google.com/colaboratory/faq.html (Accessed 26 November 2022).

Han, J., Kamber, M. and Pei, J. (2011) Data Mining Concepts and Techniques. 3rd Edition. Burlington: Morgan Kaufmann Publishers.

Hotz, N. (2011). Data Science Process Alliance. What is CRISPR-DM? Available at: https://www.datascience-pm.com/crisp-dm-2/ (Accessed 29 December 2022).

Indeed Editorial Team (2022) What are company reviews, and why are they important? Available at: https://uk.indeed.com/career-advice/career-development/what-are-company-reviews (Accessed 24 December 2022).

Indeed Editorial Team (2021) 5 Things To Look for in Company Reviews. Available at: https://www.indeed.com/career-advice/finding-a-job/checking-company-reviews (Accessed 29 December 2022).

Lawton, G. (2022) Target Tech. What is data preparation? An in-depth guide to data prep/ Available at: https://www.techtarget.com/searchdatamanagement/definition/data-preprocessing (Accessed 26 December 2022).

Luna, Zipporah (2021) Medium. Analytics Vidhya. CRISP-DM Phase 2: Data Understanding. Available at: https://medium.com/analytics-vidhya/crisp-dm-phase-2-data-understanding-b4d627ba6b45 (Accessed 24 December 2022).

McKinney, W. (2017) Python for Data Analysis. 2nd ed. Beijing: O'Reillyllyc.

Pramoditha R. (2022) Why do we set a random state in machine learning models? Medium. Available at: https://towardsdatascience.com/why-do-we-set-a-random-state-in-machine-learning-models-bb2dc68d8431 Accessed 02 January 2023).

Shin, T. (2021) Understanding Feature Importance and How to Implement it in Python. Medium. Available at: https://towardsdatascience.com/understanding-feature-importance-and-how-to-implement-it-in-python-

ff0287b20285#:~:text=Feature%20Importance%20refers%20to%20techniques (Accessed: 02 January 2023).

Sreemany, I. (2021) AnalyticsVidhya. Introduction to Feature Engineering – Everything You Need to Know! Available at: https://www.analyticsvidhya.com/blog/2021/10/a-beginners-guide-to-feature-engineering-everything-you-need-to-know/ (Accessed 28 December 2022).

The pandas development team (2022). pandas: Python Data Analysis Library. Available at: https://pandas.pydata.org/ (Accessed 25 December 2022).

Wertz, C. J. (1993) The Data Dictionary Concepts and Uses. 2nd Edition. New York: QED Information Sciences, Inc.

Wijaya, C. Y. (2021) 4 Categorical Encoding Concepts to Know for Data Scientists. Available at: https://towardsdatascience.com/4-categorical-encoding-concepts-to-know-for-data-scientists-e144851c6383 (Accessed 02 January 2023).