****

**Analysis And Comparison Of Low/High Income Developers Through The Use Of Machine Learning & Data Analysis**

**Introduction**

Throughout the duration of the coursework, I will have to investigate and explore the contents of the given data set (2021 Kaggle Machine Learning & Data Science Survey). One of the methods to obtain a good understanding of the data set will be through the implementation of exploratory data analysis aka EDA, this is an approach to analysing data and summarising the main characteristics as well as discovering relationships between features and specific variables.

As well an in-depth description of the characteristics that make up low-income and high-income developers is a key element in this coursework. This is done via determining the most effective features, which is an essential factor in providing a clear understanding of the specific attributes of a high- or low-income developer. To make these features clear and concise, they must be chosen with some thought. Too many features can return a possibly tedious and inconsistent analysis that can be hard to deter a meaning from whilst too little may possibly skip over key elements that give a better understanding if they were to be added to the analysis.

**CRISP-DM Definition**

A data science methodology (CRISP-DM) will be used at full length in this coursework, by paraphrasing the definition “CRISP DM is a methodology for shaping the data mining projects through 6 steps or phases that cycle iterations according to the developer’s requirements”. The six phases consist of: Business/Research Understanding, Data Understanding, Data Preparation, Modelling, Evaluation and Deployment. Each phases is essential to the effectiveness and success of implementing this methodology into a project, adopting this practice will ensure that the workflow and management will proceed with little hinderance.

**CRISP-DM Application and Importance in Data Science**

Summarising each phase of CRISP-DM allows for the further explanation of how it functions as well as how the application in this coursework will affect me by using the methodology.

Business Understanding aim is to assess the pre-existing data and determine objectives from its requirements as well. From this I will gain insight as to the relevance of the data to the set task(s).

Data Understanding phases’ objective is to gather a greater comprehension after initially examining the data, through assessing the data quality via in-depth analysis. This where Exploratory Data Analysis (EDA) element comes into play by creating figures to form comparisons and relationships between specific variables. The Data Understanding phase is vital as I, in this phase can extract the “best value” information aka selecting the most applicable data to create a subset from, allowing them to proceed with a understanding of the benefits of the selected information/data.

Data Preparation is a phase where specific use cases alongside variables are obtained for analysis; data is cleansed and prepared adding/removing any missing and incorrect values from the data. This phase can be quite time constricting, especially with large quantities of data however it is an important phase as it reduces the chance of outliers occurring and possibly increases the accuracy of later applied data to models.

The Modelling phases provides visualisation of the gathered data for comparison and governing how to classify the model as well as training the dataset based on the model effectiveness. Modelling is the lifeblood of a machine learning project the results of any and each model in this phase, it should advance closer to the overall objectives. Further data preparation takes places, to improve upon the given results, calibrate the model settings to optimise it even further.

The Evaluation phase determines the effectiveness of the created models to see if they fit the matching requirements laid out in prior phases; establish if the initially defined objectives have been achieved, this is done by verifying the accuracy of the testing data to the training data; evaluate whether the important problems have been accounted for, if not further corrections will be made. Finally, a decision to use the results is made before implementation as well as proceeding to the next phase.

The Deployment phase is the last of the 6 phases in CRISP-DM, the aim is to make use and present the created code and models in a concise manner. In doing so the project’s requirements and objectives should be met, from this additional documentation can be made such as a report, presentation or spreadsheet etc.

**Data Understanding, Data Pre-Processing, Exploratory Data Analysis (EDA)**

The prior mentioned is a dataset taken place in 2021 to gather responses of a varying and diverse range of software developers globally, the aim of the it was to obtain a wide angled view of how someone becomes a data scientist, their current impacts/priorities/concerns and outlining the industry itself. Responses were obtained via an open survey on the Kaggle notebook website, over 25 thousand responses were collected from 171 countries, participants gave responses to multiple choice questions were recorded in individual columns whilst multiple selection questions were recorded in multiple columns. Bear in mind that this is an online survey of 25,973 responses which isn’t representative of the entire data science industry but moreover an insight into the community itself.

**Features For Data Analysis**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Question | Summary | Type | Example |
| Age | Q1 | The developers’ age | Numerical | 18-25 |
| Education | Q4 | The highest level of education the developer has obtained | Categorical | Doctorates degree |
| Coding Experience | Q6 | How many years has the developer been programming for | Numerical | 5-10 years |
| Industry | Q20 | Current industry that the developer works in | Categorical | Energy/Mining |
| Compensation | Q25 | How much the developer has been compensated for their work on a yearly basis currently | Numerical | $40,000-49,999 |

Total Numerical Features: 4

Total Categorical Features: 4

Total Features: 8

**Dataset Quality:**

To begin, the data must be cleansed (aka remove errors, duplicates , missing values etc) so that it can be used in later analysis as well as increase the accuracy of the prediction model that will be built. The whole scope of the dataset has to be taken into consideration, once the number of responses are counted, the previously stated flawed must be identified so that they can be removed for each of the responses that flaws.

Initially once the features have been chosen before the cleansing process begins, said features must be put into a pandas dataframe for ease of access and manipulating later. This also allows a simple visualisation of the currently stored values.

**Table

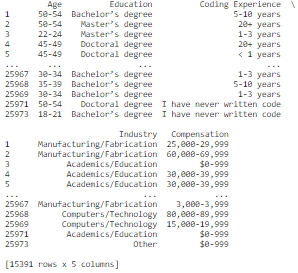
Description automatically generated with low confidence**

Figure 1: Missing Data (Highlighted in red)

Figure 2: Cleansed Data

As you can see in the above figure that there are many flaws, this can occur from the respondents of the survey: not answering a certain question; the question might not be relevant or applicable to the respondent; even only answering part of the survey and withdrawing part way through.

The flaws are shown in the ‘NaN’ label

**EDA Understanding of Chosen Features:**

Exploratory data analysis (EDA) is a popular method of analysis that is used in the machine learning community to analyse and investigate datasets to determine their properties and how they function in line with the dataset. EDA is at its most usefulness when observing the data for outlying, erroneous and flawed data. In this report the use when applying this method will be shown through figures and diagrams that visualise the properties of the selected features.

# **Age**

The first chosen feature to analyse is Age, looking through the feature it has 11 different properties that each respondent could choose from. This can be visualised by putting the data into a horizontal bar chart.(figure []). In this figure, it shown that for the age group ’25-29’ has the highest number of developers, whilst the age group ‘70+’ being the lowest number of developers. The majority of developers tend to be in their early 20s to mid 30s, from this it can be assumed that there’s still a decent amount of people interested in joining the machine learning community.

Chart, bar chart

Description automatically generated

Figure 3: Number Of Developers by Age Group Horizontal Bar Plot

Whilst figure() shows the surface level values of how the respondents interacted with this feature, to get a more in-depth detailed view of the ‘Age’ feature, the data is plotted using a pie chart. This allows us to visualise the percentages/distribution of developers for this feature. The results of the previous figure in comparison with the results of the current figure, make it evident that young to middle aged adults are the lifeblood of the industry. Although not presently shown in this figure() it will be explored through other features as to what exactly is attracting said age groups to this industry.

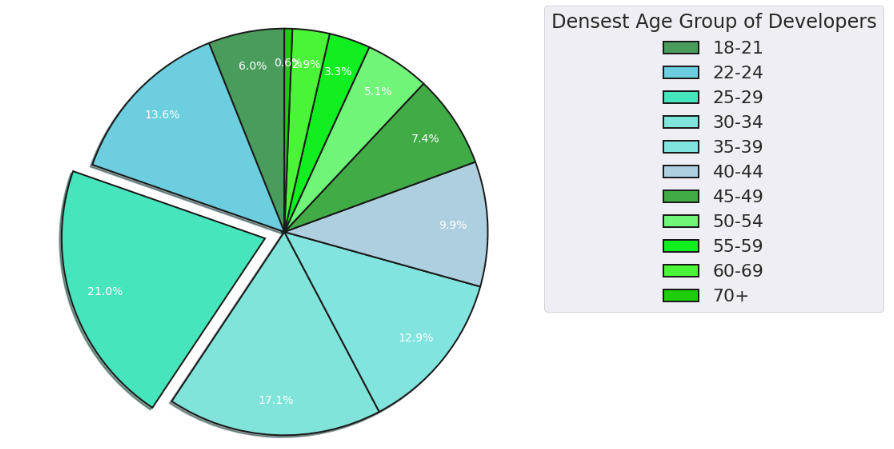


Figure 4: Pie Plot, Showing Percentage of Developers per Age Group

**Education**

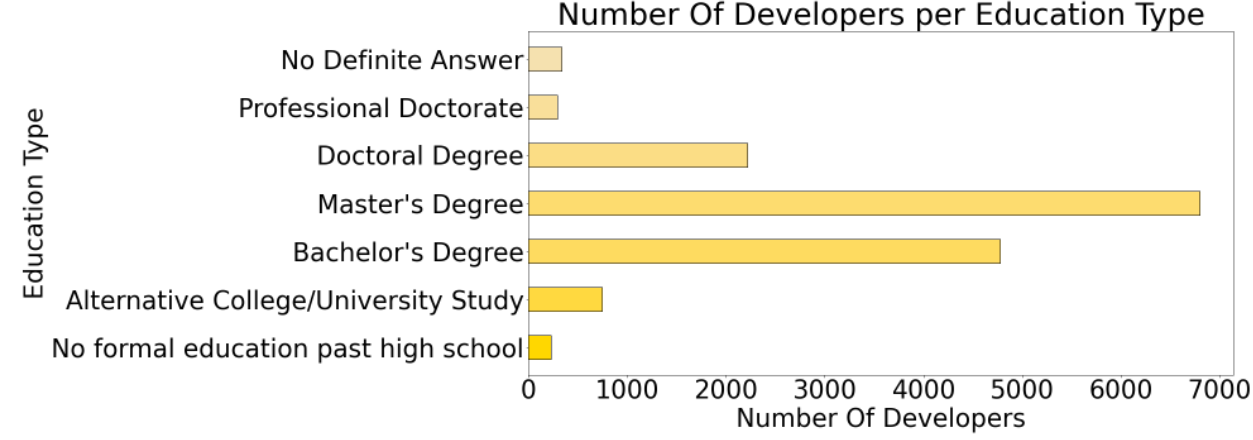
For the feature Education the respondents had a choice of 7 different properties that cover the most applicable forms of education that are considered in a working environment. They were stored as categorical values which makes it difficult to show outlying data however a horizontal bar chart can be used instead to visualise the distribution of data. From (see figure[]) it can be shown that the majority of developers have undergone some for of university course that results in a degree with a master’s being the most common whilst a doctoral degree being the least populated property that is a type of degree.

Figure 5: Number Of Developers per Education Type Horizontal Bar Plot

As the it’s shown in the figure below, there’s 3 possible occurrences of outliers (extreme data that doesn’t fit the main body of data). The 1st column contains a higher count by at least 2000+ in comparison with every other column. The last 3 columns starting with ‘$250k, $300k, $1m’ also contains no values at all, this could possibly throw off any later predictions using data from this plots bar plot. A potential fix could be to introduce a sklearn scaler and to drop the columns that have no values.

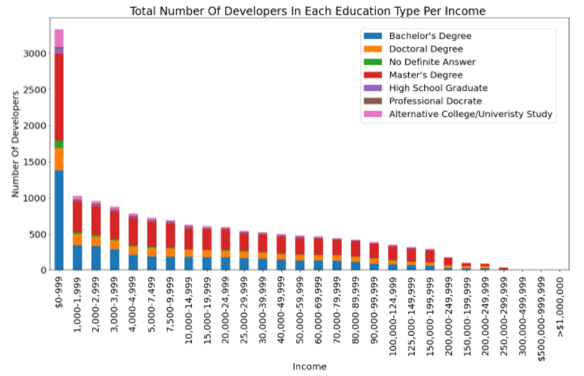


Figure 6: Total Number Of Developers In Each Education Type Per Income Stacked Bar Plot

**Coding Experience**

In the figure below it can be shown that the majority experience of the developers is between 1-3 years, from this it can be suggested that because of the influx of university educated individuals entering the machine learning workplace is a large quantity. And so they may have gained their experience, during the course they have taken, from a placement or even learning in their own time.

Chart, bar chart

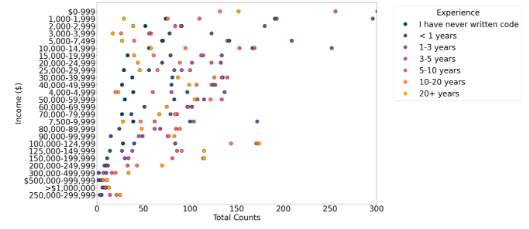
Description automatically generated

Figure 8: Experience and Income Scatter Plot

Analysing the Income and Experience scatter plot, a colormap identified in the legend displays each category of the Experience column. The use of a colormap allows us to see that going from no experience to gaining some a few years of experience the income of the developers increases, but also simultaneously the wage gap between those that have less experience than more experience is shown. Comparing columns **1.**’10,000-14,999’ and **2.**‘100,000-124,000’ for experience categories 1-3 years and 20+ years is evident of the prior statement.

For example: **1.** For this column, the developers that have 1-3 years’ experience they are highest group (250~ counts) that will earn 10,000-14,999 USD, whilst the developers that have 20+ years’ experience are the 2nd smallest group (50~ counts) that earn 10,000-14,999. **2.** For this column the developers that have 1-3 years’ experience they are 3rd highest group (80~ counts) that will earn 100,000-124,999 USD, whilst the developers that have 20+ years’ experience are the highest group (175~ counts) that earn 100,000-124,999.

The wage gap is likely due to graduates taking entry level jobs with low amounts of experience, once they have been in the industry/workplace for a few years their wage increases significantly and can stay at a high level for the rest of their careers. This attractiveness in wage due to experience will also contribute who the characteristics of a high-income developer

Figure 7: Number Of Developers per Coding Experience Horizontal Bar Plot

**Income**

For the Income feature, using a horizontal bar plot, for each individual income level the total count is displayed. This is used to analyse how the income column is distributed, for the variable ‘$0-999’ it can be considered to be an outlier as it most of the variables are between 50-200. This can be further analysed through the use of a boxplot. Although, after taking the former statement into consideration, it’s evident that most developers earn what could be argued to be a ’liveable wage’ (the balance references) of $60k. The majority earning around $80k or above which is a defining characteristic of being a high-income developer.

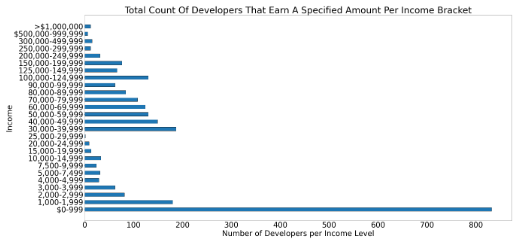


Figure 8: Total Count Of Developers That Earn a Specified Amount Per Income Bracket Horizontal bar plot

**Cluster Analysis**

The dataset in use is split into two subsets: high and low income, defined by the initially chosen features. Not only the features but as well whether the developer earn above or below $80,000 per annum. ‘Cluster Analysis is a technique that involves the grouping of data points’, This can be used to classify each element into groups that have similar properties/features. (towards data science reference) In these subsets the former will be applied, as well as the flow will be observed and/or identified.

K-means is an algorithm used to separate samples of n groups f equal variance, trying to minimize inertia aka ‘within-cluster sum of squares. K-means can be configured in different ways, it’s parameters altered to change the output to fit the desired clustering.

**K-means Parameters:**

**n\_clusters:** By default, it’s 8 clusters (but in this case It’s 3 due to comparison with the elbow plot results)

**init:** Initialisation method that takes in the data to be processed. By default, it’s ‘k-means++’

**n\_init:** By default= 10, defines the number of times the algorithm will be ran with different centroid seeds

**max\_iter:** The Max number of iterations of k-means in one run, will stop after has completed

**random\_state:** Determines random number generation for centroid initialization.Default = None

**copy\_x: Numerically centres the data, if true not modified. Default=True**

(Some Parameters not included for relevancy)

**Low Income Developers Subset Analysis**

Chart

Description automatically generated

Figure 9: Cluster Analysis-Low Income(Experience)

As it is shown in figures 9,10 and 11 the Income (Compensation) feature has been removed. That’s due to the fact that it’s the determining factor in both high and low income developer subsets, to get a better understanding through cluster analysis, that column had to be dropped.

Observing figure 9, it’s identified that the majority developers tend to have a lower income when they inexperienced (cluster 1) rather than when they’ve worked at least half a decade in the machine learning industry.

Chart, bar chart

Description automatically generated

Chart, bar chart

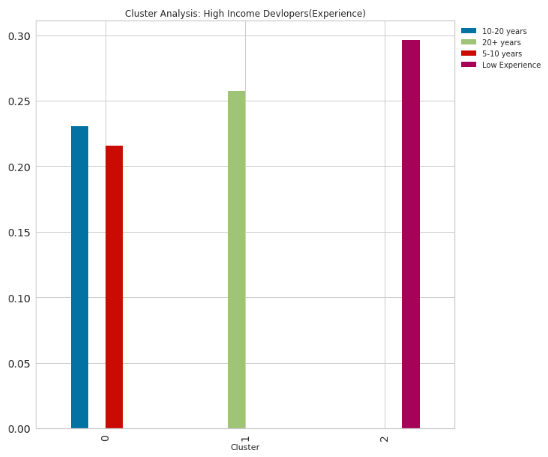
Description automatically generated

Finally as the last figure related to the low income subset, from figure 11 there’s evidence that an University Educated developer is more likely to earn a lower income than a developer who might be self-taught, taken a doctorate or just educated up until high school.

Moving on to figure 10, for the clusters shown except cluster 1, a younger developer is around 2x as less likely to be having a low income than an older developer. This is likely due to the fact developers entering the workplace need more starting years/experience and are more likely to have higher levels of education than older developers, which could in term suggest that that companies’ are giving out higher paid jobs at higher levels in the company to younger developers meaning they won’t have to move up as far up the ladder, to earn a higher income, than when older developers coming straight out as a university graduate(Most likely a Bachelor’s degree) have had to.

Figure 11: Cluster Analysis-Low Income(Experience)

Figure 10: Cluster Analysis-Low Income(Age)

****

**High Income Developers Subset Analysis**

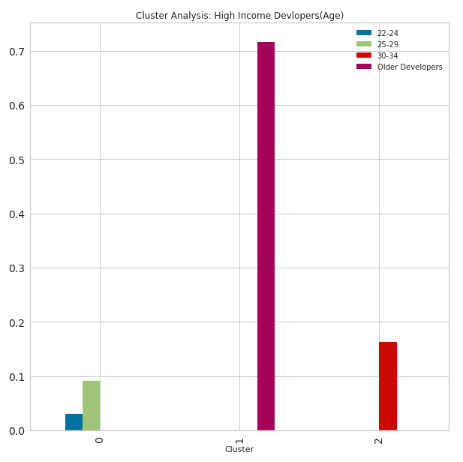
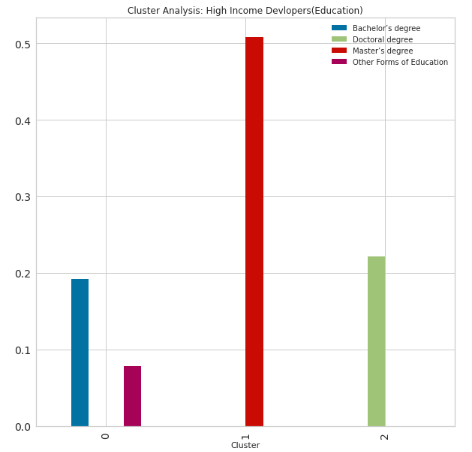


Figure 14: Cluster Analysis-High Income(Education)

Figure 13: Cluster Analysis-High Income(Age)

Those that are younger are less likely to earn those who are older which is likely due to the fact that said developers have a lot more years coding which means they can provide a better service with higher quality leading to be put in high paying company positions

Figure 12: Cluster Analysis-High Income(Experience)

With Figure 12 it can shown that although those with low experience are more likely to earn a higher income, the other experience level have comparatively seen an increase.

**Machine Learning for Classification and Implementation**

The workflow of machine learning for classification is represented in this flow below. Figure 15 aka the flow chart, visualises each necessary step in order to obtain a high level of accuracy for each and ever y model that will be created throughout the duration of this task. First data is either created via a dataframe or loaded in externally. Then the unwanted properties/features are removed from the data. This may cause some errors which in that case any missing data must be cleansed so that for machine learning methods that are prone to outliers can be used. Then the data is split into subsets such as high and low income for example. A specific percentage is taken for the test and the training data this can be altered for a wider or narrower view of the data, an extremity in either way could lead to under-representation of the data throwing off the accuracy. Once the accuracy is determined ensemble learning is used to group larger characteristics of the data which can be used for higher levels of accuracy and helps with the evaluation. Finally some level of the deployment phase (from Crisp-DM is implemented)

A picture containing timeline

Description automatically generatedText

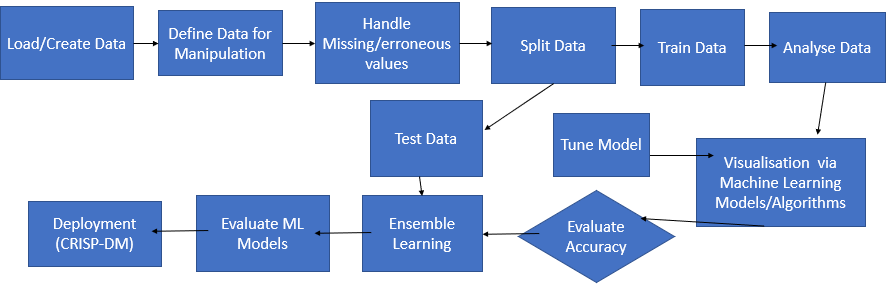
Description automatically generated

Figure 17: Decision Tree Experience

Figure 16:Decision Tree Age

**Decision Tree Model**

A decision tree model is used as the objective for this task requires for the feature to be either low or high income, for this objective a class tree will be used. Decision Trees are use classification and regression, the aim is to predict the value of the target variable .

**Parameters:**

With Setting the max\_depth to 1 the accuracy is 82.26% but increase the test\_size and the max\_deph raises the accuracy to 82.63%

Figure 15: Flow Chart: Machine Learning for Classification

**Conclusions**

Taking a through observation of the each figure and comparing them based on performance it can be assumed that the objective of determining the characteristics of an high income developer. Although the types of evaluation/accuracy models were limited to just decision tree from tuning as well as exploring the details, we can see that if I could improve on the analysis in the future I would add additional models as well a broader view of how the task is determined can be shown. I would say with confidence that the objective was met. I managed to visualise how each feature affects the objective, predicating as to how whether a developer can have a high income based on education, age, experience and compensation.

I think the strengths of my machine learning knowledge and skills were shown during the Exploratory Data Analysis phase of this project. I was less confident on plotting certain graphs effectively to efficiently display the distribution or the relationship or comparison between different features than and the end of this project. Before undertaking the project as a task at hand, research into different aspects of machine learning had to happen as then I would gain a better insight as to what the performance of the data should look like.

Coming away from his project, in the future I would like to advance my skills and knowledge in the machine learning community via undertaking self-tasked projects and partaking in different forms of competitions so that I can make better use of my python programming skills. This would help if I was to do this project again. Overall, this coursework has greatly impacted on the prior statement.