

**MACHINE LEARNING UNSUPERVISED LEARNING,  
AND FEATURE ENGINEERING  
CASE STUDY**

**STUDENT: HUGO ALBUQUERQUE COSME DA SILVA**

**ID:92126125**

## TABLE OF CONTENTS

<b>1 – Introduction</b>	<b>2</b>
<b>2 - Methodology</b>	<b>2</b>
2.1.1 - Imports and Initial Exploratory Data Analysis	2
2.1.2 - Columns renaming	3
2.1.3 - Columns relabeling	4
2.1.4 - Missing Values Investigation	6
2.1.5 - Further manipulations	7
2.1.6 - Encoding	9
<b>2.2 - Categorizing survey respondents.</b>	<b>9</b>
<b>2.3 - Cluster interpretation</b>	<b>11</b>
<b>3 - Conclusion</b>	<b>14</b>
<b>4 - Bibliography</b>	<b>15</b>

## 1 – INTRODUCTION

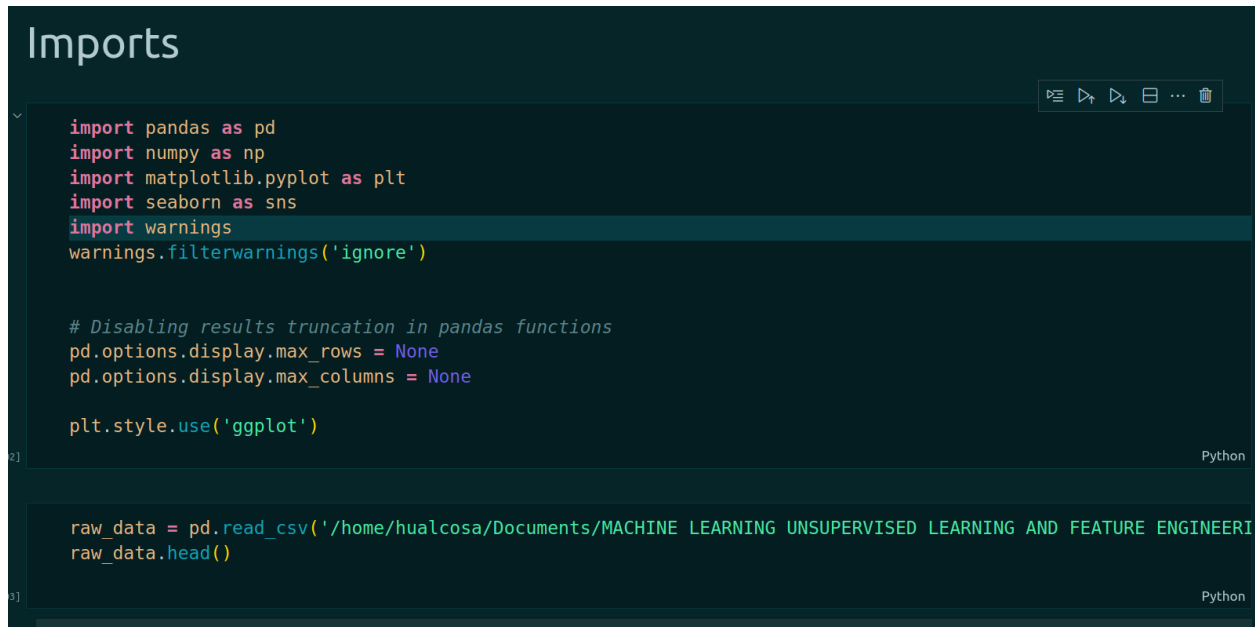
THIS CASE STUDY SIMULATES THE SCENARIO WHERE I WORK FOR TECHNOLOGY COMPANY AS A DATA SCIENTIST, AND THE HUMAN RESOURCES DEPARTMENT HAS CONTACTED ME TO HELP THEM PROVIDE SOME INSIGHTS ABOUT A SURVEY DATA RELATED TO MENTAL HEALTH IN THE WORKPLACE. THE GOAL IS TO CATEGORIZE SURVEY PARTICIPANTS, PROVIDING INTERPRETATIONS AND VISUALIZATIONS ABOUT EACH CLUSTER'S CHARACTERISTICS. THE DATA OFFER SOME CHALLENGES LIKE HIGH DIMENSIONALITY, MISSING VALUES, NON STANDARD TEXT INPUTS AND INCORRECT DATA TYPES THAT WILL BE ADDRESSED DURING THE ANALYSIS

## 2 - METHODOLOGY

### 2.1.1 - IMPORTS AND INITIAL EXPLORATORY DATA ANALYSIS

THE ANALYSIS WAS CONDUCTED USING PYTHON. IT STARTED BY IMPORTING THE NECESSARY PACKAGES, SETTING SOME CONFIGURATIONS, AND LOADING THE DATA.

FIG. 1 - IMPORTS AND READING THE DATA



```
Imports

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')

# Disabling results truncation in pandas functions
pd.options.display.max_rows = None
pd.options.display.max_columns = None

plt.style.use('ggplot')

raw_data = pd.read_csv('/home/hualcosa/Documents/MACHINE LEARNING UNSUPERVISED LEARNING AND FEATURE ENGINEERING')
raw_data.head()
```

USING PANDAS, IT CAN BE SEEN THAT THE DATA CONSISTS OF **1433** ROWS AND **63** COLUMNS, MOST COLUMNS ARE IN A GENERIC "OBJECT" FORMAT AND THE COLUMN NAMES ARE IN THE FORMAT OF QUESTIONS, WHICH CONTAINS SPACES AND ARE OFTEN LONG.

### 2.1.2 - COLUMNS RENAMING

THE NEXT STEP WAS TO RENAME COLUMNS TO ABBREVIATIONS THAT WILL MAKE THE ANALYSIS EASIER. IT WAS ALSO CREATED A DICTIONARY THAT WILL ALLOW ME TO CONSULT THE ORIGINAL COLUMN NAMES AFTER I APPLY THE CLUSTERING TECHNIQUES

FIG.2 - COLUMNS RENAMING

## Renaming cols

```
old_cols = raw_data.columns.tolist()
new_cols = ['self_empl_flag', 'comp_no_empl', 'tech_comp_flag', 'tech_role_flag', 'mh_coverage_flag',
'mh_coverage_awareness_flag', 'mh_employer_discussion', 'mh_resources_provided', 'mh_anonymity_fl',
'mh_medical_leave', 'mh_discussion_neg_impact', 'ph_discussion_neg_impact', 'mh_discussion_cowork',
'mh_discussion_supervis', 'mh_eq_ph_employer', 'mh_conseq_coworkers', 'mh_coverage_flag2', 'mh_on',
'mh_diagnosed&reveal_clients_flag', 'mh_diagnosed&reveal_clients_impact', 'mh_diagnosed&reveal_co',
'mh_prod_impact', 'mh_prod_impact_perc', 'prev_employers_flag', 'prev_mh_benefits', 'prev_mh_bene',
'prev_mh_discussion', 'prev_mh_resources', 'prev_mh_anonymity', 'prev_mh_discuss_neg_conseq', 'pr',
'prev_mh_discussion_cowork', 'prev_mh_discussion_supervisor', 'prev_mh_importance_employer', 'pre',
'future_ph_specification', 'why/why_not', 'future_mh_specification', 'why/why_not2', 'mh_hurt_on_',
'mh_sharing_friends/fam_flag', 'mh_bad_response_workplace', 'mh_for_others_bad_response_workplace',
'mh_disorder_past', 'mh_disorder_current', 'yes:what_diagnosis?', 'maybe:whats_your_diag', 'mh_di',
'yes:condition_diagnosed', 'mh_sought_proffes_treatm', 'mh_eff_treat_impact_on_work', 'mh_not_eff',
'age', 'sex', 'country_live', 'live_us_teritory', 'country_work', 'work_us_teritory', 'work_posit']

old_to_new_cols = {k:v for k, v in zip(old_cols, new_cols)}
```

### 2.1.3 - COLUMNS RELABELING

THERE ARE SOME COLUMNS IN THE DATASET THAT NEED TO BE STANDARDIZED. I STARTED WITH THE 'SEX' COLUMN, WHICH CONTAINED A LOT OF NON STANDARD VALUES

FIG.3 - NON STANDARD TEXT INPUTS IN SEX COLUMN

```
tmp['sex'].str.lower().value_counts().sort_index()

female
afab
agender
androgynous
bigender
cis female
cis male
cis man
cis-woman
cisdude
cisgender female
dude
enby
f
fem
female
female
female (props for making this a freeform field, though)
female assigned at birth
female or multi-gender femme
female-bodied; no feelings about gender
female/woman
fluid
fm
genderfluid
...
transgender woman
transitioned, m2f
unicorn
woman
```

AFTER NORMALIZATION, VALUES PRESENT WERE: 'MALE', 'FEMALE' OR 'OTHER'.

THE VALUES IN THE COLUMN 'COMP\_NO\_EMPL' ALSO WERE STANDARDIZED.

FIG. 4 - COMP\_NO\_EMPL STANDARDIZATION

```
tmp['comp_no_empl'].value_counts()

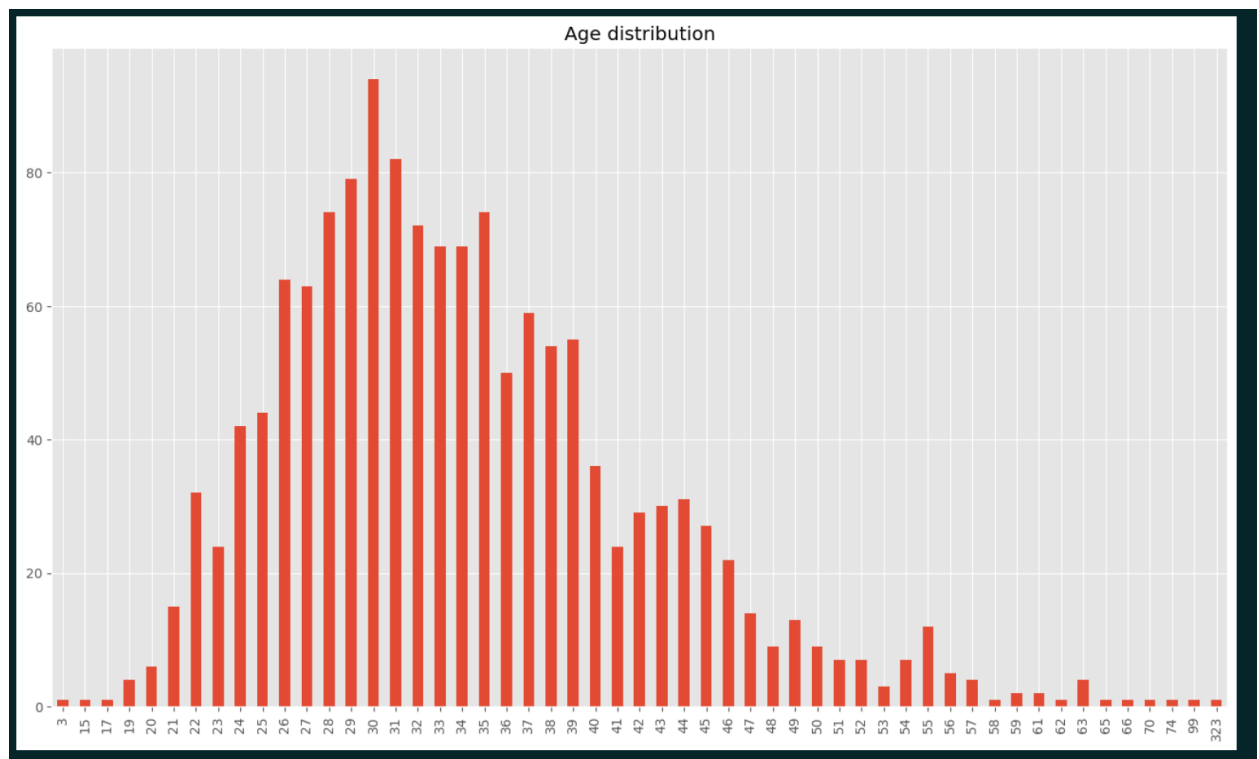
111]
... 26-100          292
     More than 1000  256
     100-500        248
     6-25           210
     500-1000        80
     1-5            60
     Name: comp_no_empl, dtype: int64

tmp['comp_no_empl'].replace(to_replace = ['More than 1000'], value = '>1000', inplace = True)

112]
```

ANOTHER NECESSARY MODIFICATION WAS THE REPLACEMENT OF OUTLIER VALUES IN THE AGE COLUMN BY THE MEAN AGE VALUE

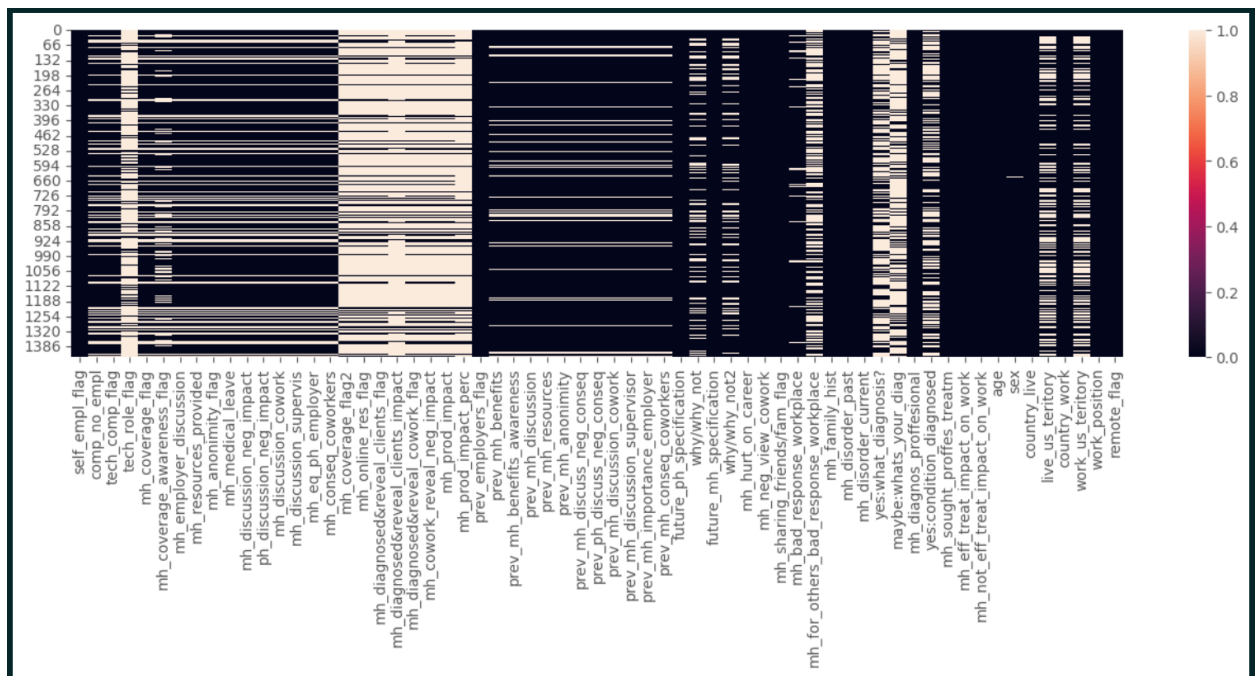
FIG. 5 - AGE COLUMN ORIGINALLY CONTAINED OUTLIERS



## 2.1.4 - MISSING VALUES INVESTIGATION

WITH THE HELP OF PANDAS ISNA() AND SEABORN HEATMAP FUNCTION, I DID A QUICK VISUALIZATION TO HAVE A GLIMPSE OF MISSING VALUES ACROSS THE DIFFERENT COLUMNS:

FIG. 6 - MISSING VALUES



THE NEXT STEP WAS TO REMOVE THE COLUMNS WHERE MORE THAN 50% OF THE ROWS HAD DATA MISSING IN THAT SPECIFIC COLUMN. FOR THE REMAINING CASES, THE STRATEGY USED WAS TO

INSTACIATE A **SIMPLEIMPUTER** OBJECT TO IMPUTE MISSING VALUES WITH THE MOST FREQUENT VALUE FOR THAT COLUMN.

FIG.7 - IMPUTING MISSING VALUES

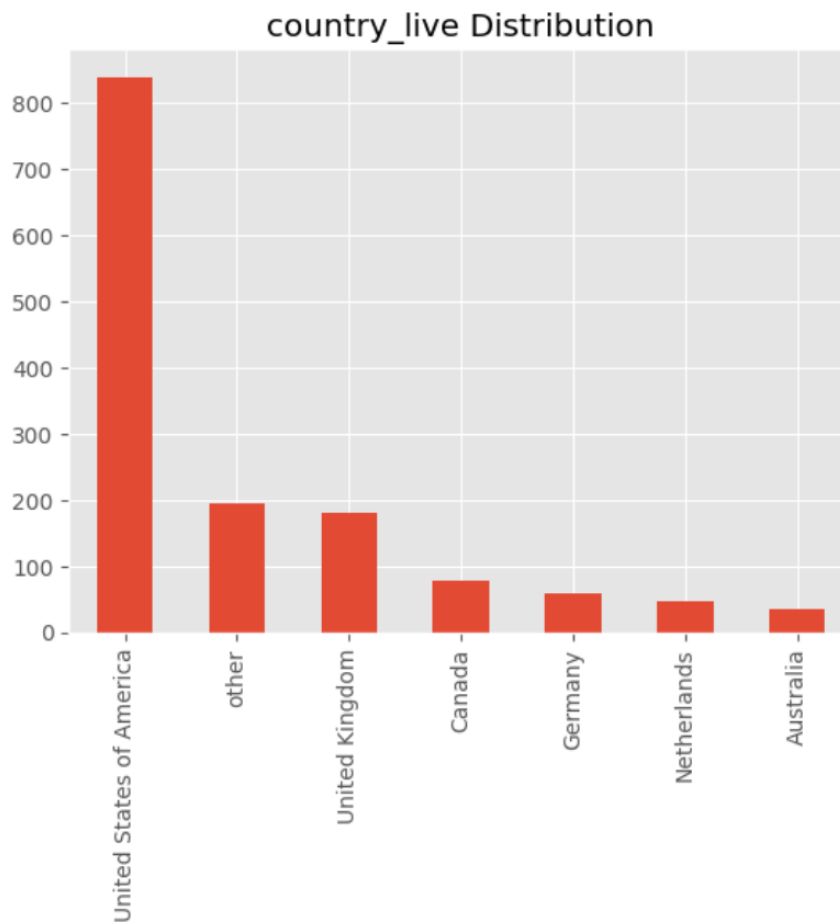
```
from sklearn.impute import SimpleImputer
imputer = SimpleImputer(missing_values=np.nan, strategy='most_frequent')
tmp = pd.DataFrame(imputer.fit_transform(tmp), columns=tmp.columns)
tmp.isna().sum()
```

Python

### 2.1.5 - FURTHER MANIPULATIONS

THE COLUMNS RELATED TO LOCATION ('COUNTRY\_LIVE', 'LIVE\_US\_TERRITORY', 'COUNTRY\_WORK', 'WORK\_US\_TERRITORY') HAD VERY LITTLE REPRESENTATIVENESS. SINCE IN ORDER FOR STATISTICS ABOUT A POPULATION TO BE SIGNIFICANT, NORMALLY AT LEAST 30 DATA POINS ARE NECESSARY, I HAVE GROUPED THESE MINORITY CATEGORIES INTO A SINGLE CATEGORY CALLED 'OTHER'. AS AN ILLUSTRATION, HERE IS THE 'COUNTRY\_LIVE' COLUMN DISTRIBUTION AFTER THE REGROUPING:

FIG. 8 - EXAMPLE OF REGROUPING UNDERREPRESENTED GROUPS IN LOCATION COLUMNS



THE ORIGINAL COLUMN THAT IDENTIFIED WHETHER THE PERSON WORKED IN A TECH ROLE HAD TOO MANY MISSING DATA AND WAS FILTERED OUT. A NEW COLUMN WAS ENGINEERED BASED ON KEYWORDS THAT WERE PRESENT IN THE 'WORK\_POSITION' COLUMN

FIG 9 - ENGINEERING 'IS\_TECH\_ROLE' COLUMN

```
def is_tech_role(name):  
    l_name = name.lower()  
    for cat in ['back-end', 'front-end', 'devops']:  
        if cat in l_name:  
            return 1  
    return 0  
tmp['tech_role_flag'] = tmp['work_position'].apply(is_tech_role)  
tmp['tech_role_flag'].value_counts()
```

```
1    1007  
0     426  
Name: tech_role_flag, dtype: int64
```



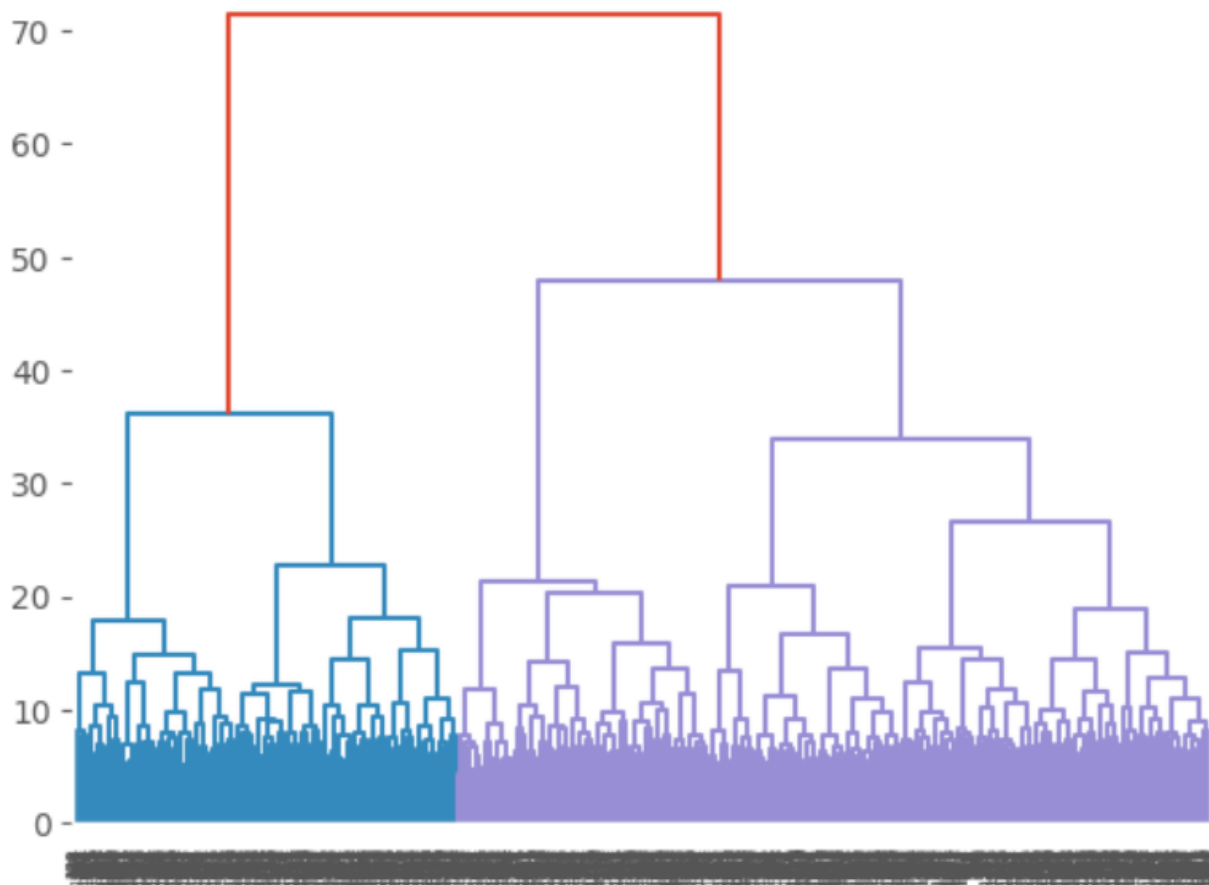
### 2.1.6 - ENCODING

IN ORDER TO PERFORM DIMENSIONALITY REDUCTION, THE DATA NEEDS TO BE IN A NUMERICAL FORMAT. FOR THIS, THE DATA NEEDED TO BE ENCODED. NOMINAL CATEGORICAL COLUMNS, THE MAJORITY OF COLUMNS IN THE DATASET, WERE ENCODED USING PANDAS GET\_DUMMIES FUNCTION. COMP\_NO\_EMPL, A CATEGORICAL ORDINAL COLUMN, WERE ENCODED USING LABEL ENCODER.

## 2.2 - CATEGORIZING SURVEY RESPONDENTS.

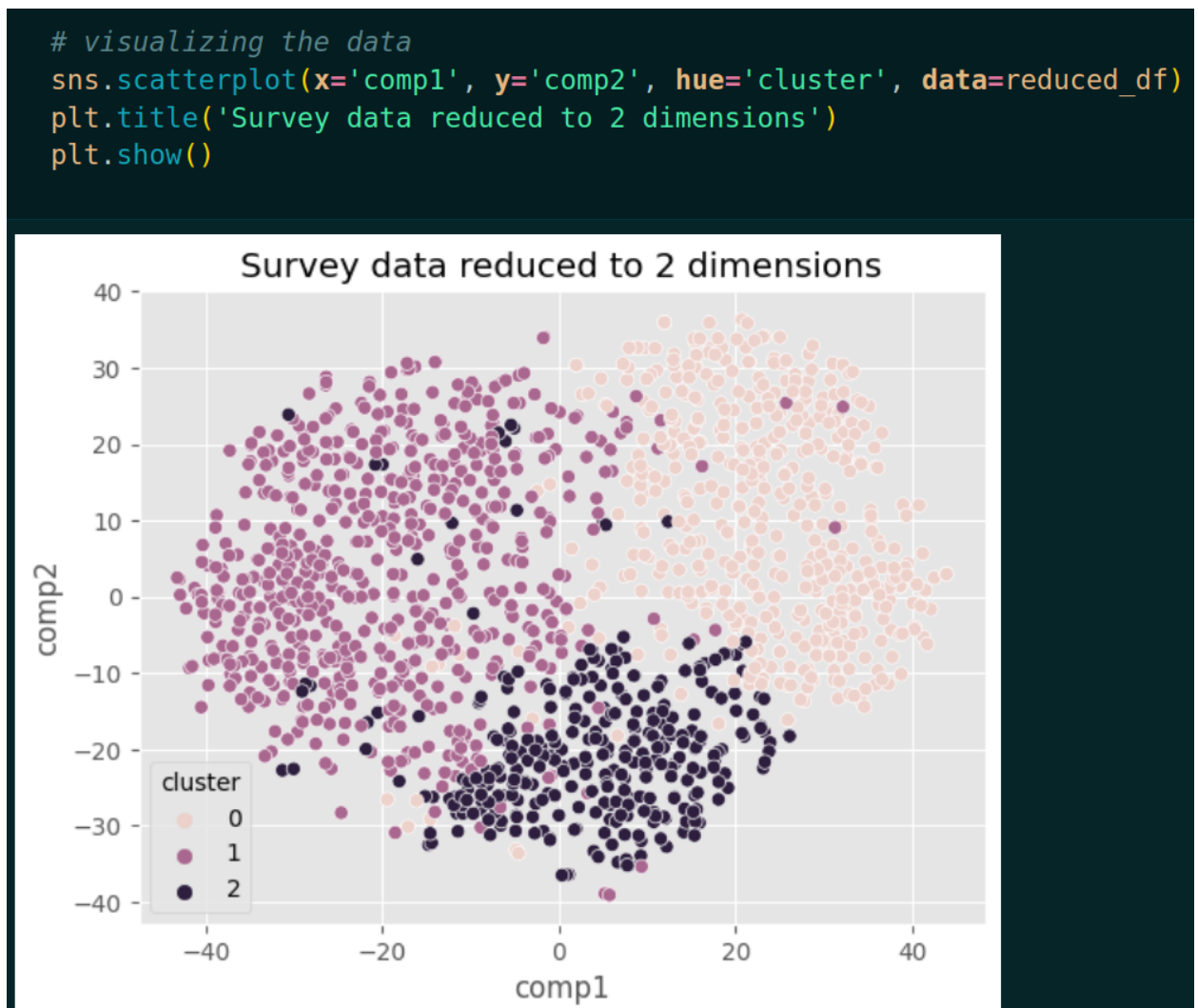
IN ORDER TO CATEGORIZE SURVEY RESPONDENTS, I EXPERIMENTED WITH DIFFERENT CLUSTERING TECHNIQUES, BUT ENDED UP CHOOSING HIERARCHICAL CLUSTERING, SINCE IT WAS THE TECHNIQUE THAT SHOWED THE MOST PROMISING RESULTS. THE WARD ALGORITHM WAS USED, SINCE IT MINIMIZES WITHIN CLUSTER VARIANCE, AND IT WORKED WELL IN THIS CASE.

FIG. 10 - HIERARCHICAL CLUSTERING



BASED ON VISUAL INSPECTION, IT WAS CHOSEN A CUTOFF POINT JUST ABOVE 40, WHICH MEANS THERE IS GOING TO BE 3 CLUSTERS. SKLEARN'S `AgglomerativeClustering` WAS USED TO CLUSTER AND LABEL THE DATA POINTS. TO VISUALIZE THE RESULTS, I APPLIED `T-SNE`, A DIMENSIONALITY REDUCTION TECHNIQUE THAT PERFORMS WELL FOR DATASETS WITH A LARGE NUMBER OF CATEGORICAL COLUMNS. SINCE THE DATASET IS NOT BIG, THE COMPUTATIONAL COST WAS NOT A PROBLEM. THE REDUCED DATASET WAS CONCATENATED WITH THE CLUSTER LABELS FROM THE HIERARCHICAL CLUSTERING PROCESS, WHICH WAS THEN USED TO PRODUCE THE FOLLOWING VISUALIZATION:

FIG. 11 - REDUCE DATA HUED BY CLUSTER



## 2.3 - CLUSTER INTERPRETATION

AS WE CAN SEE IN THE PREVIOUS PICTURE, ALTHOUGH THERE ARE SOME NOISY REGIONS, THE LABELING SEEMS TO GROUP THE DATA WELL. I FIRST INVESTIGATE THE CLUSTER SIZES:

FIG. 12 - CLUSTER SIZES

```
[170] cluster_sizes = encoded_data['cluster'].value_counts().values

[171] cluster_sizes

... array([633, 485, 315])
```

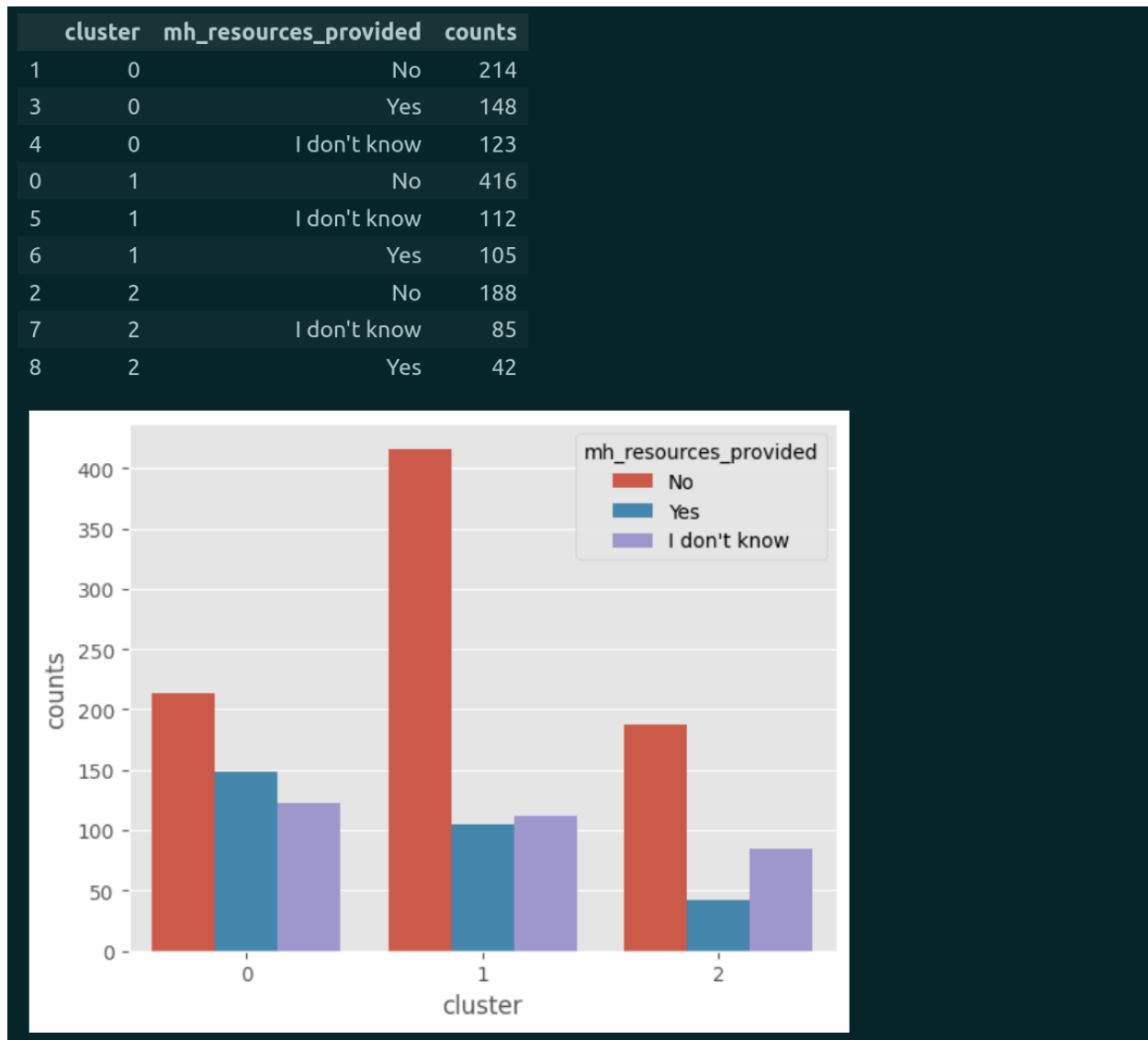
TO ANALYZE THE DATA BEHIND THE CLUSTERS BETTER, I HAVE ADDED THE LABEL COLUMN TO THE DATASET VERSION BEFORE ENCODING. I DEVELOPED A UTILITY FUNCTION TO HELP VISUALIZING THE DISTRIBUTION OF A A SPECIFIC COLUMN ACROSS THE DIFFERENT CLUSTERS. I ALSO PRINTED A TABLE OF COUNTS TO COMPLEMENT THE ANALYSIS

FIG. 13 - UTILITY FUNCTION

```
def compare_clusters_across_dimension(c):
    """
    display the table of frequency for categorical variable across different clusters AND
    Plot a barplot showing the distribution of categorical variable across different clusters
    """
    vis_data = df_cat[['cluster', c]].value_counts().reset_index().sort_values(by='cluster')
    vis_data.columns = ['cluster', c, 'counts']
    display(vis_data)
    # plot the data
    sns.barplot(x='cluster', y='counts', hue=c, data=vis_data)
    plt.show()
    print('\n\n')
```

FOR EACH COLUMN IN THE DATASET, I CALLED THE FUNCTION AND ITS OUTPUTS HELPED ME ANALYZE THE CLUSTERS SIMILARITIES/DIFFERENCES ACROSS THAT DIMENSIONS. AS AN EXAMPLE, HERE IS THE RESULT FOR THE 'MH\_RESOURCES\_PROVIDED' COLUMN:

FIG. 11 - UTILS FUNCTION UTILIZATION EXAMPLE



THE GRAPHS FOR EACH COLUMN CAN BE CHECKED IN THE NOTEBOOK THAT IS AVAILABLE IN THE CASE STUDY'S GITHUB REPOSITORY. BASED ON THOSE GRAPHS, THE FOLLOWING CAN BE INFERRED ABOUT THE CLUSTERS:

## CLUSTER 0

- COMPRISED MOSTLY OF EMPLOYEES
- FORMED MOSTLY BY EMPLOYEES WHO WORK AT COMPANIES WITH MORE THAN 1000 EMPLOYEES
- FORMED BY EMPLOYEES WHO WORK AT COMPANIES WHERE IS NOT SOMEWHAT EASY TO ASK FOR A LEAVE BECAUSE OF MENTAL HEALTH ISSUES
- MOST PEOPLE IN THIS CLUSTER WOULD NOT FEEL SAFE TALKING ABOUT MENTAL HEALTH ISSUE WITH A COWORKER

## CLUSTER 1

- PEOPLE WHO HAVE HAD MENTAL ISSUES ON THE PAST
- PEOPLE WHO CURRENTLY HAVE A MENTAL DISORDER
- MOST PEOPLE WERE DIAGNOSED AS HAVING A MENTAL DISORDER
- FAMILY OF THIS CLUSTER MEMBERS HAVE BIGGER MENTAL HEALTH ISSUES HISTORY
- MOST PEOPLE IN CLUSTER 1 SOUGHT PROFESSIONAL TREATMENT
- MOST TECH EMPLOYEES
- FORMED MOSTLY BY EMPLOYEES WHO WORK AT COMPANIES WITH 26-100 EMPLOYEES
- EMPLOYEES WHO WORK AT COMPANIES THAT MOSTLY OFFER MENTAL HEALTH BENEFITS
- EMPLOYEES ARE HIGHLY UNCONFIDENT WHETHER THEY WILL SUFFER NEGATIVE CONSEQUENCES IF THEY DISCUSS ABOUT MENTAL HEALTH ISSUES
- THE MAJORITY OF PEOPLE FEEL COMFORTABLE DISCUSSING A MENTAL HEALTH ISSUE WITH THEIR BOSSES
- PEOPLE IN THIS CLUSTER ARE MORE OPEN TO SHARE WITH FRIENDS AND FAMILY THAT THEY SUFFER MENTAL HEALTH
- EXPERIENCED/OBSERVE MORE NEGATIVE RESPONSES IN THEIR PREVIOUS WORKPLACE COMPARED TO OTHER CLUSTERS
- PEOPLE WITH MENTAL HEALTH FEW THAT THEIR PROBLEM CONTRIBUTES TO NOT BEING TREATED EFFECTIVELY

## CLUSTER 2

- MOST PEOPLE IN THIS CLUSTER DIDN'T HAVE MENTAL HEALTH PROBLEMS IN THE PAST
- MOST PEOPLE DOESN'T HAVE A MENTAL DISORDER
- MOST PEOPLE WERE NOT DIAGNOSED WITH MENTAL DISORDER
- FAMILY OF THESE CLUSTER MEMBER MOSTLY DON'T HAVE MENTAL HEALTH ISSUES
- FORMED MOSTLY BY EMPLOYEE WHO WORK AT COMPANIES WITH 26-500 EMPLOYEES
- FORMED BY EMPLOYEES PARTICULARLY UNAWARE OF THE MENTAL HEALTH BENEFITS THE COMPANY PROVIDES
- EMPLOYEES UNDER THIS CATEGORY HAVE LESS MENTAL HEALTH RESOURCES PROVIDED
- HAS A HIGHER PROPORTION OF PEOPLE WHO NEVER HAD DISCUSSIONS ABOUT MENTAL HEALTH WITH THEIR SUPERVISORS
- HAS A HIGHER PROPORTION OF PEOPLE WHOSE PREVIOUS EMPLOYERS DIDN'T TAKE MENTAL HEALTH ISSUES AS SERIOUS AS PHYSICAL HEALTH

- ONLY CLUSTER WHERE THE PROPORTION OF PEOPLE WHO WOULD BE WILLING TO BRING UP A PHYSICAL HEALTH PROBLEM TO AN EMPLOYEE IS BIGGER THAN THOSE WHO DOESN'T
- PEOPLE IN THIS CLUSTER ARE PROPORTIONATELY LESS AFRAID TO BRING UP MENTAL HEALTH ISSUES WITH A POTENTIAL EMPLOYER
- PEOPLE IN THIS CLUSTER ARE LESS INCLINED TO BELIEVE THAT BEING IDENTIFIED WITH MENTAL HEALTH ISSUES WILL HURT THEIR CAREER
- PEOPLE IN THIS CLUSTER ARE LESS INCLINED TO BELIEVED THAT THEY WOULD BE VIEWED NEGATIVELY IF THEIR COWORKERS KNEW THEY SUFFER FROM MENTAL HEALTH

#### INTERESTING FINDINGS(COMMON AMONG CLUSTERS):

- MOST SURVEY PARTICIPANTS ARE MALE, WHO LIVE IN USA OR UNITED KINGDOM
- MOST RESPONDENTS WHO LIVE IN US ARE FROM CALIFORNIA
- MOST RESPONDENTS WORK AS EITHER DEVOPS/SYSADMIN OR BACKEND DEVELOPERS
- MOST PEOPLE WORK REMOTELY SOMETIMES OR ALWAYS
- MOST COMPANIES DOESN'T OFFER MENTAL HEALTH COVERAGE BENEFITS
- MOST PEOPLE WHO ANSWERED THE SURVEY AND WORK IN THE US ARE FROM CALIFORNIA
- THE MOST PREVALENT WORKING POSITION OF PEOPLE WHO ANSWERED THE SURVEY IS BACKEND ENGINEER
- MOST EMPLOYEES DON'T DISCUSS ABOUT MENTAL HEALTH ISSUES WITH THEIR BOSSES
- EMPLOYEES IN GENERAL DON'T KNOW WHETHER THEIR ANONYMITY IS PROTECTED IN CASE THEY DISCLOSE MENTAL HEALTH ISSUES
- MOST RESPONDENTS DIDN'T DISCUSSED ABOUT MENTAL HEALTH ISSUES WITH THEIR EMPLOYERS
- MOST PEOPLE WORKED IN PLACES WHERE THEIR PREVIOUS EMPLOYER DIDN'T PROVIDE ANY MENTAL HEALTH ASSISTANCE

## 3 - CONCLUSION

THIS CASE STUDY PRESENTED AN ANALYSIS OF A SURVEY DATA RELATED TO THE MENTAL HEALTH OF PEOPLE WORKING IN THE TECH SPACE. SEVERAL DATA PROCESSING STEPS WERE APPLIED, LIKE COLUMN VALUES NORMALIZATION, OUTLIER FILTERING AND MISSING VALUES FILTERING AND IMPUTATION.

FEATURE ENGINEERING TECHNIQUES LIKE COLUMNS ENCODING AND DIMENSIONALITY REDUCTION WERE APPLIED TO HELPING MAKING SENSE OF THE DATA. HIERARCHICAL CLUSTER HELPED IDENTIFYING 3 DISTINCT CLUSTERS, WHOSE MEANING WERE CLARIFIED WITH THE HELP OF VISUALIZATIONS AND FREQUENCY TABLES. IT WAS ALSO HIGHLIGHTED COMMON CHARACTERISTICS AMONG THE 3 CLUSTERS.

IT DEPICTED HOW USEFUL FEATURE ENGINEERING, DATA CLEANING AND UNSUPERVISED LEARNING CAN BE WHEN IT IS NECESSARY TO MAKE SENSE OF UNLABELED, HIGH-DIMENSIONAL, MESSY DATA.

## 4 - BIBLIOGRAPHY

SILVA, H. A. C. DA. (2023, JUNE 21). HUALCOSA/IU\_UNSUPERVISED\_LEARNING. GITHUB.  
[HTTPS://GITHUB.COM/HUALCOSA/IU\\_UNSUPERVISED\\_LEARNING](https://github.com/HualCosa/IU_Unsupervised_Learning)

SKLEARN.IMPUTE.SIMPLEIMPUTER — SCIKIT-LEARN 0.24.1 DOCUMENTATION. (N.D.).  
SCIKIT-LEARN.ORG.  
[HTTPS://SCIKIT-LEARN.ORG/STABLE/MODULES/GENERATED/SKLEARN.IMPUTE.SIMPLEIMPUTER.HTML](https://scikit-learn.org/stable/modules/generated/sklearn.impute.SimpleImputer.html)

PANDAS.GET\_DUMMIES — PANDAS 1.2.4 DOCUMENTATION. (N.D.). PANDAS.PYDATA.ORG.  
[HTTPS://PANDAS.PYDATA.ORG/DOCS/REFERENCE/API/PANDAS.GET\\_DUMMIES.HTML](https://pandas.pydata.org/docs/reference/api/pandas.get_dummies.html)

SKLEARN.PREPROCESSING.LABELENCODER — SCIKIT-LEARN 0.22.1 DOCUMENTATION. (2019).  
SCIKIT-LEARN.ORG.  
[HTTPS://SCIKIT-LEARN.ORG/STABLE/MODULES/GENERATED/SKLEARN.PREPROCESSING.LABELENCODER.HT  
ML](https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.LabelEncoder.html)

SCIPY.CLUSTER.HIERARCHY.WARD — SciPY v1.10.1 MANUAL. (N.D.). DOCS.SCIPIY.ORG. RETRIEVED  
JUNE 21, 2023, FROM  
[HTTPS://DOCS.SCIPIY.ORG/DOC/SCIPY/REFERENCE/GENERATED/SCIPY.CLUSTER.HIERARCHY.WARD.HTML](https://docs.scipy.org/doc/scipy/reference/generated/scipy.cluster.hierarchy.ward.html)

SKLEARN.CLUSTER.AGGLOMERATIVECLUSTERING. (N.D.). SCIKIT-LEARN.  
[HTTPS://SCIKIT-LEARN.ORG/STABLE/MODULES/GENERATED/SKLEARN.CLUSTER.AGGLOMERATIVECLUSTERIN  
G.HTML](https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html)

SKLEARN.MANIFOLD.TSNE — SCIKIT-LEARN 0.21.3 DOCUMENTATION. (2014). SCIKIT-LEARN.ORG.  
[HTTPS://SCIKIT-LEARN.ORG/STABLE/MODULES/GENERATED/SKLEARN.MANIFOLD.TSNE.HTML](https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html)