Implementation of Exploratory Data Analysis (EDA) in Python

Ahmad Farhan AlShammari
Department of Computer and Information Systems
College of Business Studies, PAAET
Kuwait

ABSTRACT
The goal of this research is to develop an exploratory data analysis model in Python. Exploratory Data Analysis (EDA) is used to understand the nature of data. It helps to identify the main characteristics of data (patterns, trends, and relationships). The application of exploratory data analysis helps to build a solid foundation for more advanced analysis.

The basic steps of exploratory data analysis are explained: importing libraries, reading data, displaying data, displaying general information, computing descriptive statistics, cleaning data (duplicates, missing values, and outliers), and analyzing data (univariate, bivariate, and multivariate).

The developed model was tested on an experimental dataset. The model successfully performed the basic steps of exploratory data analysis and provided the required results.

Keywords
Artificial Intelligence, Machine Learning, Data Science, Data Analysis, Exploratory Data Analysis, EDA, Univariate, Bivariate, Multivariate, Python, Programming.

1. INTRODUCTION
In recent years, machine learning has played a major role in the development of computer systems. Machine learning (ML) is a branch of Artificial Intelligence (AI) which is focused on the study of computer algorithms to improve the performance and efficiency of computer programs [1-14].

Exploratory data analysis is extremely important in the field of machine learning. It is sharing knowledge with other fields like: programming, data science, mathematics, statistics, and numerical methods [15-19].

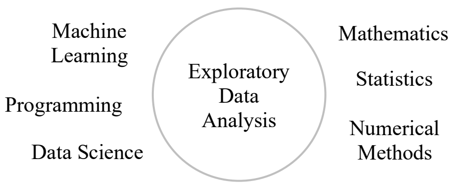


Fig 1: Area of Exploratory Data Analysis

Exploratory data analysis is used to understand the nature of data (structure and content). It helps to identify the main characteristics of data (patterns, trends, and relationships). The better understanding of data is crucial for data analysts to apply the appropriate statistical methods, leading to more accurate results.

Exploratory data analysis is widely used in the applications of machine learning, for example: regression, prediction, classification, clustering, etc.

2. LITERATURE REVIEW
The review of literature provided a comprehensive overview of the basic concepts, steps, and methods of exploratory data analysis [20-33].

Exploratory data analysis is very essential in machine learning. It is the first step in any data analysis process. It was developed by John Tukey in the 1970s to help statisticians understand data and identify the potential issues before going into more complex analysis [34].

The better understanding of data will always help to improve the performance and efficiency of the applied model.

The fundamental concepts of exploratory data analysis are explained in the following section.

Exploratory Data Analysis:
Exploratory Data Analysis (EDA) is the process of studying data to understand its nature and identify its characteristics. First, the original data is cleaned from errors (duplicates, missing values, and outliers). Then, the cleaned data is analyzed using the appropriate statistical methods to find out the patterns, trends, and relationships within data.

The concept of exploratory data analysis is illustrated in the following diagram:

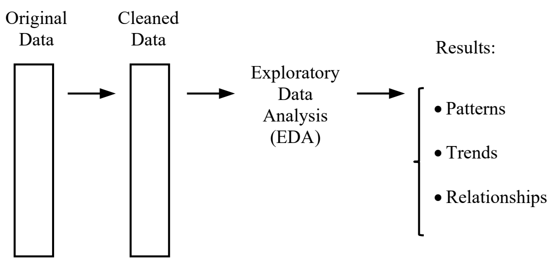


Fig 2: Concept of Exploratory Data Analysis

Types of Data Analysis:
In general, there are three types of data analysis: univariate, bivariate, and multivariate.

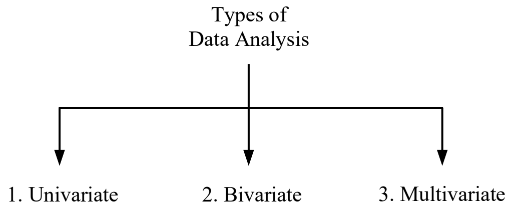


Fig 3: Types of Data Analysis

The univariate analysis involves studying one variable, for example: insurance cost. The bivariate analysis involves studying two variables, for example: insurance cost and age. The multivariate analysis involves studying three or more variables, for example: insurance cost, age, and sex.

Types of Variables:

Simply, variables are classified into two types: numerical, and categorical.

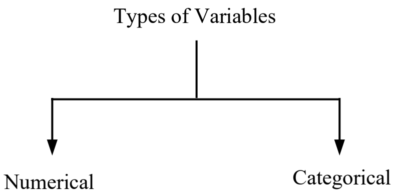


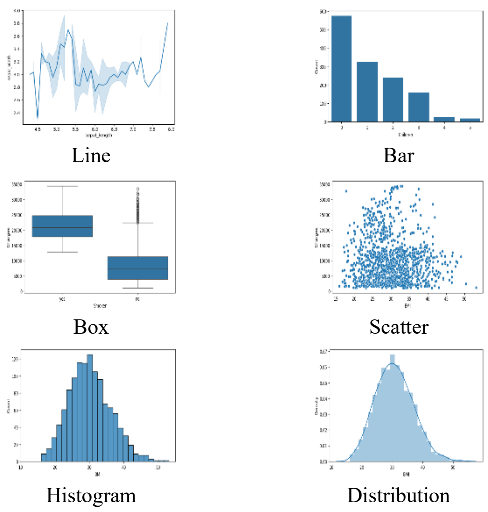
Fig 4: Types of Variables

Numerical variables include numbers like integers and reals. For example: age (30), temperature (40), score (80.5), salary (1000), etc.

On the other hand, categorical variables include categories like types and groups. For example: sex (male, female), region (north, south, east, west), smoking (yes, no), subject (math, science, history, …), etc.

Methods of Data Analysis:

There are different methods used in data analysis where each method has a specific visualization. For example: line, bar, box, scatter, histogram, distribution, pairplot, and heatmap.



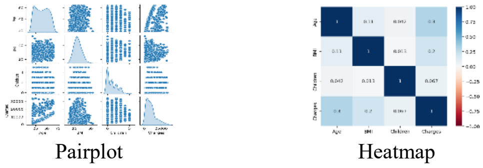


Fig 5: Methods of Data Analysis

Actually, selecting a method depends on both the type of analysis (univariate, bivariate, or multivariate) and the type of variables (numerical or categorical).

Exploratory Data Analysis Model:

The exploratory data analysis model is summarized in the following description:

Input: Original data.
Output: Results (patterns, trends, and relationships).
Processing: First, the original data is cleaned from errors (duplicates, missing values, and outliers). Then, the general information is displayed. Next, the descriptive statistics are computed. After that, the cleaned data is analyzed (univariate, bivariate, and multivariate) using the appropriate statistical methods. Finally, the required results (patterns, trends, and relationships) are obtained.

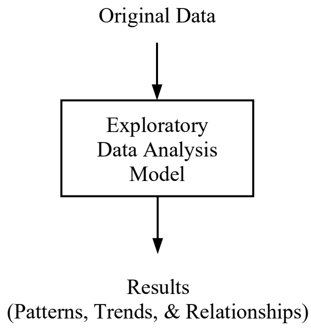


Fig 6: Exploratory Data Analysis Model

Python:

Python [35] is an open-source general-purpose programming language. It is very simple to code, easy to learn, and powerful. It is the most popular programming language for the development of machine learning applications.

Python provides additional libraries for different purposes for example: Numpy [36], Pandas [37], Matplotlib [38], Seaborn [39], NLTK [40], SciPy [41], and SK Learn [42].

3. RESEARCH METHODOLOGY

The basic steps of exploratory data analysis are: (1) importing libraries, (2) reading data, (3) displaying data, (4) displaying general information, (5) computing descriptive statistics, (6) cleaning data (duplicates, missing values, and outliers), and (7) analyzing data (univariate, bivariate, and multivariate).

1. Importing Libraries
2. Reading Data
3. Displaying Data
4. Displaying General Information
5. Computing Descriptive Statistics
6. Cleaning Data
7. Analyzing Data

Fig 7: Steps of Exploratory Data Analysis

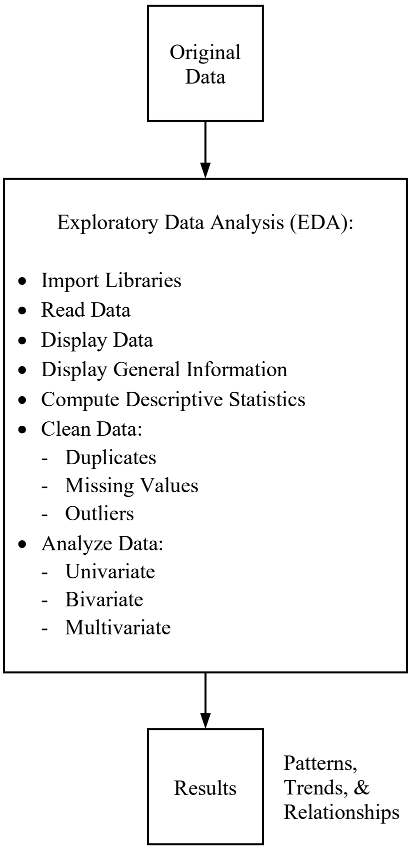


Fig 8: Flowchart of Exploratory Data Analysis

The basic steps of exploratory data analysis are explained in the following section.

1. Importing Libraries:

The required libraries (Pandas, Matplotlib, and Seaborn) are imported by the following code:

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

2. Reading Data:

The original data is read from the csv file source and converted into data frame by the following code:

```
df = pd.read_csv("dataset.csv")
```

3. Displaying Data:

The data is displayed by the following code:

```
df.head()      # first rows
df.tail()      # last rows
```

4. Displaying General Information:

The general information about data is displayed by the following code:

```
df.info()
```

It shows information about column names, non-null counts, data types, and shape.

For specific information, the following commands are used as shown here:

```
df.columns     # column names
df.dtypes      # data types
df.shape       # number of rows and columns
```

5. Computing Descriptive Statistics:

The descriptive statistics of data are computed by the following code:

```
df.describe()
```

It shows the count, mean, standard deviation, min, max, and percentiles (25%, 50%, and 75%) of numerical columns.

6. Cleaning Data:

The original data should be cleaned from errors (duplicates, missing values, and outliers). Cleaning data is done by the following steps:

6.1. Duplicates:

The duplicates are checked by the following code:

```
df[df.duplicated()]
```

Then, they are deleted by the following code:

```
df = df.drop_duplicates()
```

6.2. Missing Values:

The missing values are checked by the following code:

```
df.isna().sum()
```

Then, they are deleted by the following code:

```
df = df.dropna()
```

6.3. Outliers:

The outliers are extreme values that exceed the normal range of data. This range is defined by the lower and upper limits. They are calculated by the following code:

```
Q1 = df.col.quantile(0.25)
Q3 = df.col.quantile(0.75)
IQR = Q3 - Q1
lower_limit = Q1 - 1.5*IQR
```

```
upper_limit = Q3 + 1.5*IQR
```

Then, the values that exceed the lower and upper limits are deleted. It is done by the following code:

```
df = df[(df.col >= lower_limit) &
        (df.col <= upper_limit)]
```

7. Analyzing Data:

The data analysis is performed in three levels: univariate, bivariate, and multivariate. For each level, the appropriate statistical methods are used according to the type of variables (numerical or categorical). The different methods are explained in the following section.

7.1. Line:

The line is used to analyze a numerical variable. It is plotted by the following code:

```
sns.lineplot(df.col)
```

7.2. Bar:

The bar is used to analyze a categorical variable. It is plotted by the following code:

```
sns.countplot(df.col)
```

7.3. Box:

The box is used to analyze a numerical variable. It is plotted by the following code:

```
sns.boxplot(df.col)
```

7.4. Scatter:

The scatter is used to analyze numerical variables. It is plotted by the following code:

```
sns.scatterplot(df.col1, df.col2)
```

7.5. Histogram:

The histogram is used to analyze a numerical variable. It is plotted by the following code:

```
sns.histplot(df.col)
```

7.6. Distribution:

The distribution is used to analyze a numerical variable. It is plotted by the following code:

```
sns.distplot(df.col)
```

7.7. Pairplot:

The pairplot is used to analyze numerical variables. It is plotted by the following code:

```
sns.pairplot(df)
```

7.8. Correlation Matrix:

The correlation matrix is used to measure the correlation between numerical variables. It is computed by the following code:

```
cm = df.corr()
```

7.9. Heatmap:

The heatmap is used to display the correlation matrix. It is plotted by the following code:

```
sns.heatmap(cm)
```

4. RESULTS AND DISCUSSION

The developed model was tested on an experimental dataset from Kaggle [43]. The model performed the basic steps of exploratory data analysis and provided the required results. The output is explained in the following section.

Note: The data analysis is done using Jupyter Notebook [44].

Displaying Data:

The original data is loaded from the csv file and displayed as shown in the following view:

	Age	Sex	BMI	Children	Smoker	Region	Charges
0	19	female	27.900	0	yes	southwest	16884.92400
1	18	male	33.770	1	no	southeast	1725.55230
2	28	male	33.000	3	no	southeast	4449.46200
3	33	male	22.705	0	no	northwest	21984.47061
4	32	male	28.880	0	no	northwest	3866.85520

The data contains (1338) rows and (7) columns. The columns are: age, sex, BMI, children, smoker, region, and charges.

Displaying General Information:

The general information about data is displayed as shown in the following view:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1338 entries, 0 to 1337
Data columns (total 7 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Age       1338 non-null   int64
 1   Sex       1338 non-null   object
 2   BMI       1338 non-null   float64
 3   Children  1338 non-null   int64
 4   Smoker    1338 non-null   object
 5   Region    1338 non-null   object
 6   Charges   1338 non-null   float64
dtypes: float64(2), int64(2), object(3)
memory usage: 73.3+ KB
```

Computing Descriptive Statistics:

The descriptive statistics of data are computed and displayed as shown in the following view:

	Age	BMI	Children	Charges
count	1338.000000	1338.000000	1338.000000	1338.000000
mean	39.207025	30.663397	1.094918	13270.422265
std	14.049960	6.098187	1.205493	12110.011237
min	18.000000	15.960000	0.000000	1121.873900
25%	27.000000	26.296250	0.000000	4740.287150
50%	39.000000	30.400000	1.000000	9382.033000
75%	51.000000	34.693750	2.000000	16639.912515
max	64.000000	53.130000	5.000000	63770.428010

Cleaning Data:

The original data is cleaned from errors (duplicates, missing values, and outliers). About (10.5%) of data is deleted using the steps explained in the previous section. Now, the data is cleaned and contains (1198) rows.

For example, to remove the outliers in the target variable (charges), the lower and upper limits are calculated as shown in the following view:

```
Charges Outliers:

Q1 = 4746.344
Q3 = 16657.71745
IQR = 11911.37345
Lower Limit = -13120.716175
Upper Limit = 34524.777625
```

The following charts show the charges boxplot before and after removing outliers (values above upper limit):

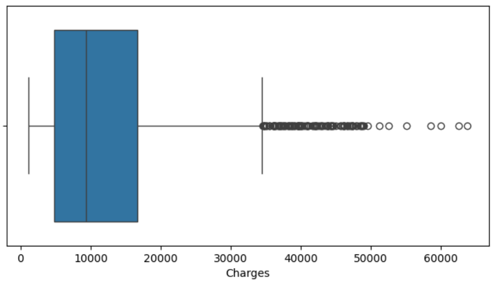


Fig 9: Charges Before Removing Outliers

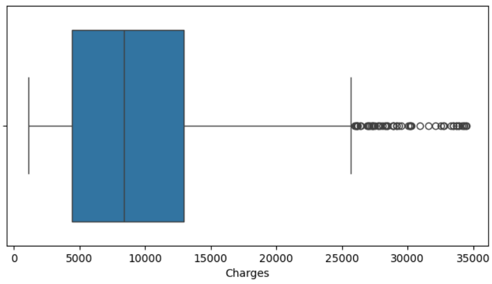


Fig 10: Charges After Removing Outliers

Analyzing Data:

The three levels of data analysis (univariate, bivariate, and multivariate) are explained in the following section.

1. Univariate Analysis:

The univariate analysis is performed for each variable according to the type of variable (numerical or categorical).

The univariate analysis is illustrated in the following charts:

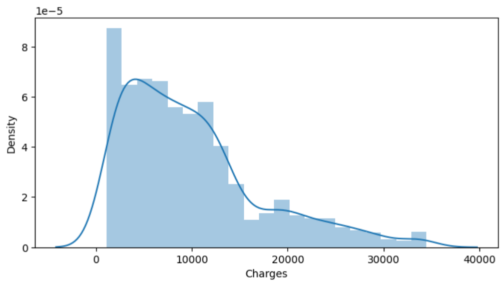


Fig 11: Charges Distribution

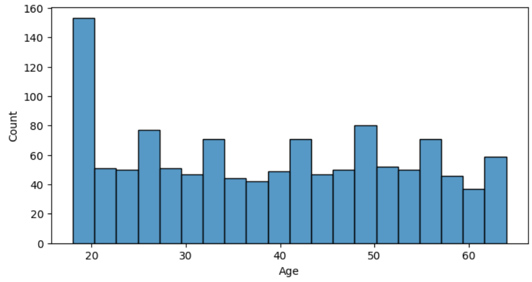


Fig 12: Age Histogram

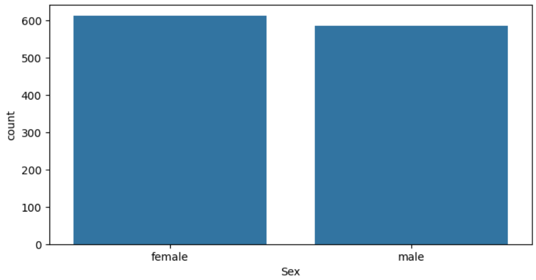


Fig 13: Sex Count

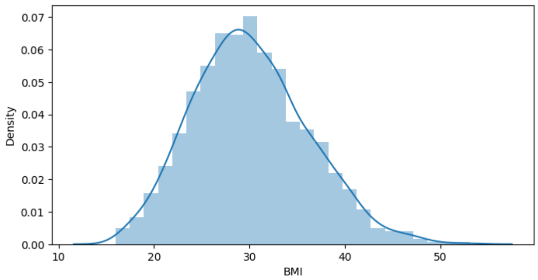


Fig 14: BMI Distribution

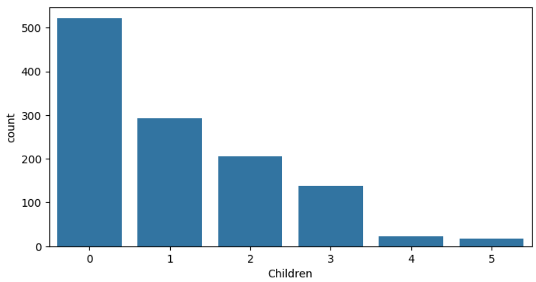


Fig 15: Children Count

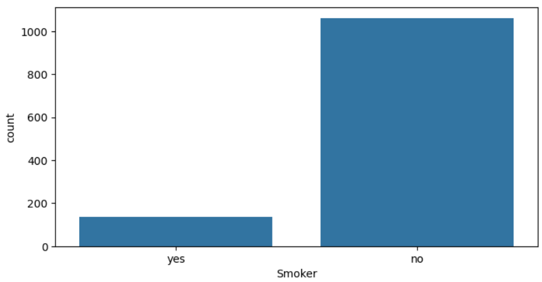


Fig 16: Smoker Count

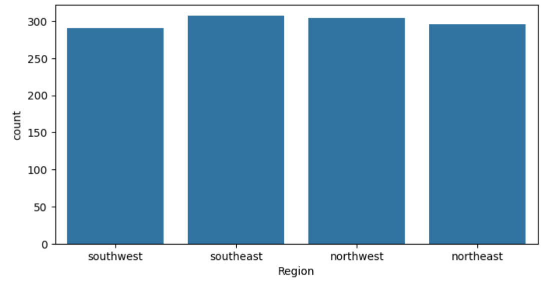


Fig 17: Region Count

2. Bivariate Analysis:

The bivariate analysis is performed between the target variable (charges) and the other variables individually.

The bivariate analysis is illustrated in the following charts:

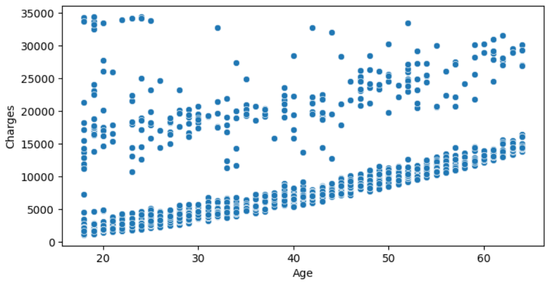


Fig 18: Age/Charges Scatter

The plot shows a positive relation between age and charges.

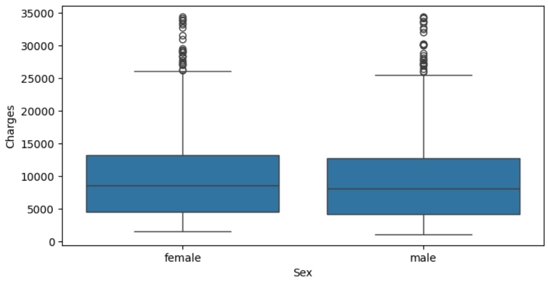


Fig 19: Sex/Charges Boxplot

The plot shows no difference in charges for sex.

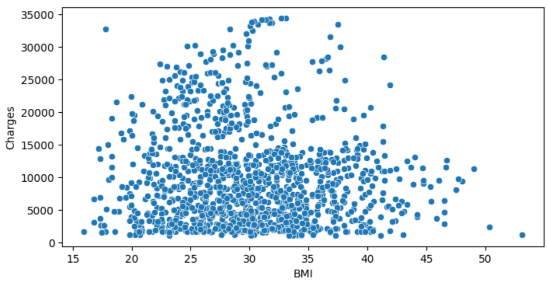


Fig 20: BMI/Charges Scatter

The plot shows a weak relation between charges and BMI.

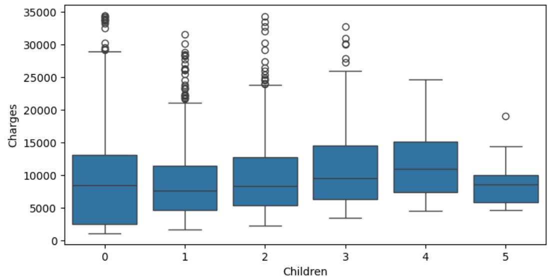


Fig 21: Children/Charges Scatter

The plot shows small differences in charges for children.

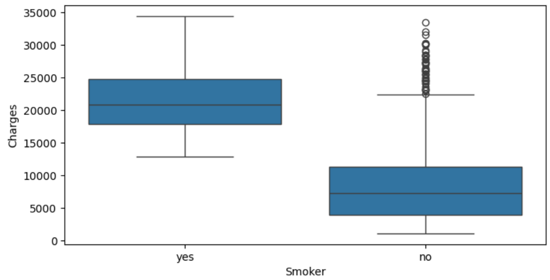


Fig 22: Smoker/Charges Boxplot

The plot shows higher charges for smokers than non-smokers.

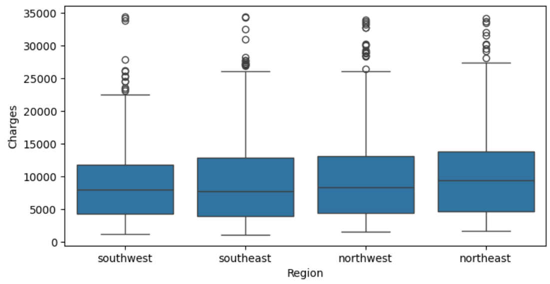


Fig 23: Region/Charges Boxplot

The plot shows no differences in charges for region.

3. Multivariate Analysis:

The multivariate analysis is performed between the target variable (charges) and the other variables collectively.

The pairplot of variables is plotted as shown in the following chart:

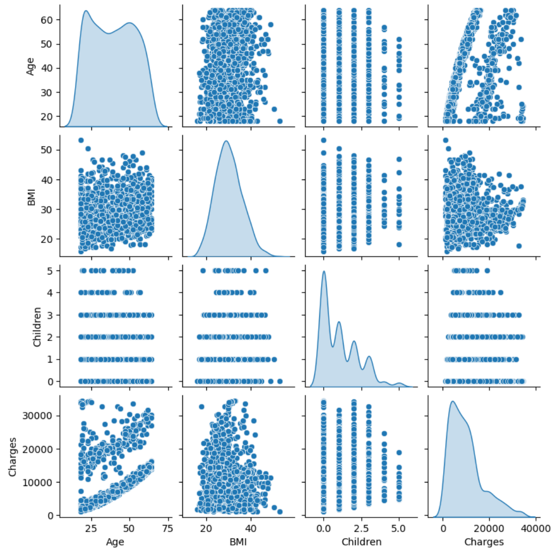


Fig 24: Pairplot of Variables

The plot shows the pairwise relationships between the variables: age, BMI, children, and charges. The diagonal charts are displayed as distribution plots and the other charts are displayed as scatter plots.

The scatter and box plots are also used in multivariate analysis to examine the relationship between two variables against the third variable.

For example, the target variable (charges) is examined with age, BMI, and children against smoker (yes, no) which is displayed in different colors. They are shown in the following charts:

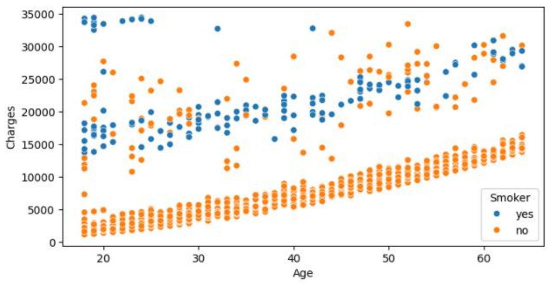


Fig 25: Age-Charges Scatter Against Smoker

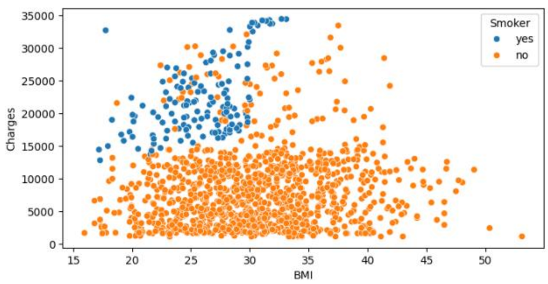


Fig 26: BMI-Charges Scatter Against Smoker

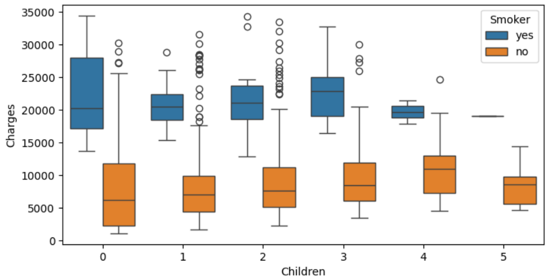


Fig 27: Children-Charges Scatter Against Smoker

The plots show that non-smokers are charged less than smokers.

Now, the correlation matrix is used to measure the correlation between variables. It is computed and displayed as shown in the following view:

	Age	BMI	Children	Charges
Age	1.000000	0.119704	0.039201	0.436891
BMI	0.119704	1.000000	0.002798	-0.066453
Children	0.039201	0.002798	1.000000	0.082932
Charges	0.436891	-0.066453	0.082932	1.000000

The strength of correlation can be described by the following scale:

Absolute Value	Meaning
0 – 0.249	Very Weak
0.25 – 0.49	Weak
0.5 – 0.749	Strong
0.75 - 1	Very Strong

Then, the heatmap of correlation matrix is plotted as shown in the following chart:

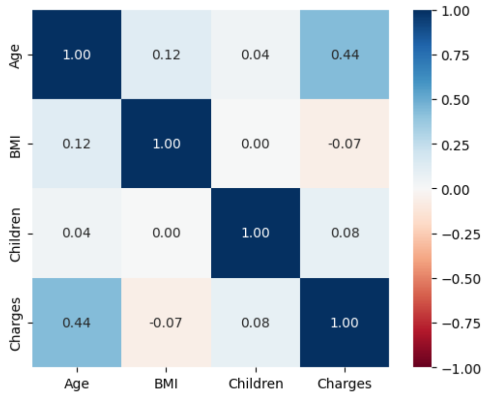


Fig 28: Heatmap of Correlation Matrix

The heatmap shows that the target variable (charges) has a weak positive correlation (0.44) with age, a very weak negative correlation (-0.07) with BMI, and a very weak positive correlation (0.08) with children.

In summary, the output shows that the model has successfully performed the basic steps of exploratory data analysis and provided the required results.

5. CONCLUSION

Machine Learning is playing a major role in the development of computer systems. It helps to improve the performance and efficiency of computer programs.

Exploratory data analysis is extremely important in machine learning. It is the first step in any data analysis process. It is used to understand the nature of data. It helps to identify the main characteristics of data (patterns, trends, and relationships).

In this research, the author developed a model to perform exploratory data analysis in Python. The basic steps of exploratory data analysis are: importing libraries, reading data, displaying data, displaying general information, computing descriptive statistics, cleaning data (duplicates, missing values, and outliers), and analyzing data (univariate, bivariate, and multivariate).

The developed model was tested on an experimental dataset and provided the required results: general information, descriptive statistics, univariate analysis, bivariate analysis, and multivariate analysis.

In the future, more work is really needed to improve the current methods of exploratory data analysis. In addition, they should be more investigated on different fields, domains, and datasets.

6. REFERENCES

[1] Sammut, C., & Webb, G. I. (2011). "Encyclopedia of Machine Learning". Springer.

[2] Jung, A. (2022). "Machine Learning: The Basics". Springer.

[3] Kubat, M. (2021). "An Introduction to Machine Learning". Springer.

[4] Li, H. (2023). "Machine Learning Methods". Springer.

[5] Dey, A. (2016). "Machine Learning Algorithms: A Review". International Journal of Computer Science and Information Technologies, 7 (3), 1174-1179.

[6] Bonaccorso, G. (2018). "Machine Learning Algorithms: Popular Algorithms for Data Science and Machine Learning". Packt Publishing.

[7] Jo, T. (2021). "Machine Learning Foundations: Supervised, Unsupervised, and Advanced Learning". Springer.

[8] Jordan, M. I., & Mitchell, T. M. (2015). "Machine Learning: Trends, Perspectives, and Prospects". Science, 349 (6245), 255-260.

[9] Forsyth, D. (2019). "Applied Machine Learning". Springer.

[10] Chopra, D., & Khurana, R. (2023). "Introduction to Machine Learning with Python". Bentham Science Publishers.

[11] Müller, A. C., & Guido, S. (2016). "Introduction to Machine Learning with Python: A Guide for Data Scientists". O'Reilly Media.

[12] Zollanvari, A. (2023). "Machine Learning with Python: Theory and Implementation". Springer.

[13] Raschka, S. (2015). "Python Machine Learning". Packt Publishing.

[14] Sarkar, D., Bali, R., & Sharma, T. (2018). "Practical Machine Learning with Python". Apress.

[15] Igual, L., & Seguí, S. (2017). "Introduction to Data Science: A Python Approach to Concepts, Techniques and Applications". Springer.

[16] VanderPlas, J. (2017). "Python Data Science Handbook: Essential Tools for Working with Data". O'Reilly Media.

[17] Yale, K., Nisbet, R., & Miner, G. D. (2018). "Handbook of Statistical Analysis and Data Mining Applications". Academic Press.

[18] Unpingco, J. (2022). "Python for Probability, Statistics, and Machine Learning". Springer.

[19] James, G., Witten, D., Hastie, T., Tibshirani, R., & Taylor, J. (2023). "An Introduction to Statistical Learning: With Applications in Python". Springer.

[20] Navlani, A., Fandango, A., & Idris, I. (2021). "Python Data Analysis". Packt Publishing.

[21] Unpingco, J. (2021). "Python Programming for Data Analysis". Springer.

[22] McKinney, W. (20128). "Python for Data Analysis". O'Reilly Media.

[23] Embarak, O. (2018). "Data Analysis and Visualization using Python ". Apress.

[24] Denis, D. J. (2021). "Applied Univariate, Bivariate, and Multivariate Statistics Using Python: A Beginner's Guide to Advanced Data Analysis". John Wiley & Sons.

[25] Mukhiya, S. K., & Ahmed, U. (2020). "Hands-On Exploratory Data Analysis with Python". Packt Publishing.

[26] Chen, D. (2018). "Pandas for Everyone: Python Data Analysis". Addison-Wesley.

[27] Molin, S. (2019). "Hands-On Data Analysis with Pandas". Packt Publishing.

[28] Myatt, G. J. (2014). "Making Sense of Data: A Practical Guide to Exploratory Data Analysis and Data Mining". John Wiley & Sons.

[29] Vigni, M. L., Durante, C., & Cocchi, M. (2013). "Exploratory Data Analysis". In Data Handling in Science and Technology. 28, 55-126. Elsevier.

[30] Komorowski, M., Marshall, D. C., Salciccioli, J. D., & Crutain, Y. (2016). "Exploratory Data Analysis". In Secondary Analysis of Electronic Health Records, pp. 185-203, Springer.

[31] Sahoo, K., Samal, A. K., Pramanik, J., & Pani, S. K. (2019). "Exploratory Data Analysis using Python". International Journal of Innovative Technology and Exploring Engineering, 8 (12), 4727-4735.

[32] Heydt, M. (2017). "Learning Pandas: High-Performance Data Manipulation and Analysis in Python". Packt Publishing.

[33] Miller, C. (2018). "Hands-On Data Analysis with Numpy and Pandas". Packt Publishing.

[34] Tukey, J. W. (1977). "Exploratory Data Analysis". Addison-Wesley.

[35] Python: http://www.python.org

[36] Numpy: http://www.numpy.org

[37] Pandas: http://pandas.pydata.org

[38] Matplotlib: http://www. matplotlib.org

[39] Seaborn: http://seaborn.pydata.org

[40] NLTK: http://www.nltk.org

[41] SciPy: http://scipy.org

[42] SK Learn: http://scikit-learn.org

[43] Kaggle: http://www.kaggle.com

[44] Jupyter: http://www.jupyter.org