EXPLANATORY DATA ANALYSIS – ADULT INCOME

Monika, Gena, Mustapha, Fazal

CONTENTS:

EXPLANATORY DATA ANALYSIS – ADULT INCOME

SECTION 1: INTRODUCTION

The Adult dataset is from the Census Bureau and the task is to predict whether a given adult makes more than $50,000 a year based attributes such as education, hours of work per week, etc.

Goal

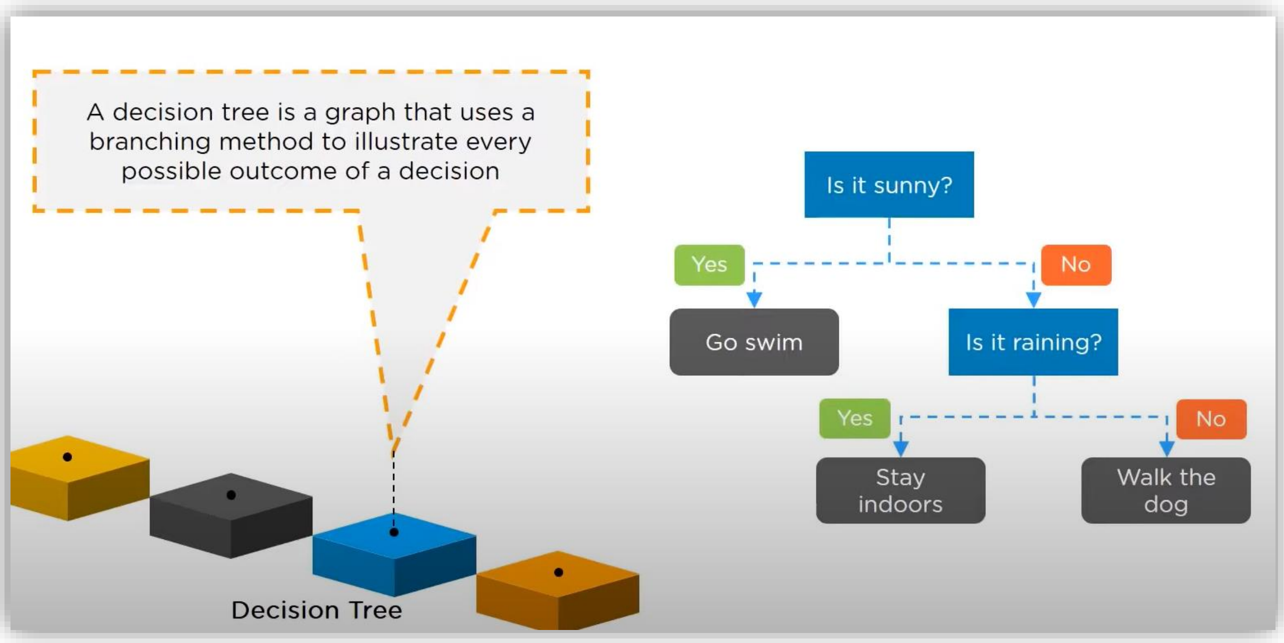
By applying this easy-to-use model, you can predict if a given individual earns more than $50,000 a year or not.

The model is based on the Decision Tree Classification technique and it achieves accuracy of around 95%. Our objective is to deliver the model, which may quickly provide the answer of your question and that way saves time and cost to your company.

The functionality of this model was carefully tested, before being delivered to you. We will proceed with further details about the model itself, the applied method and detailed analysis.

Decision Tree Classification Technique

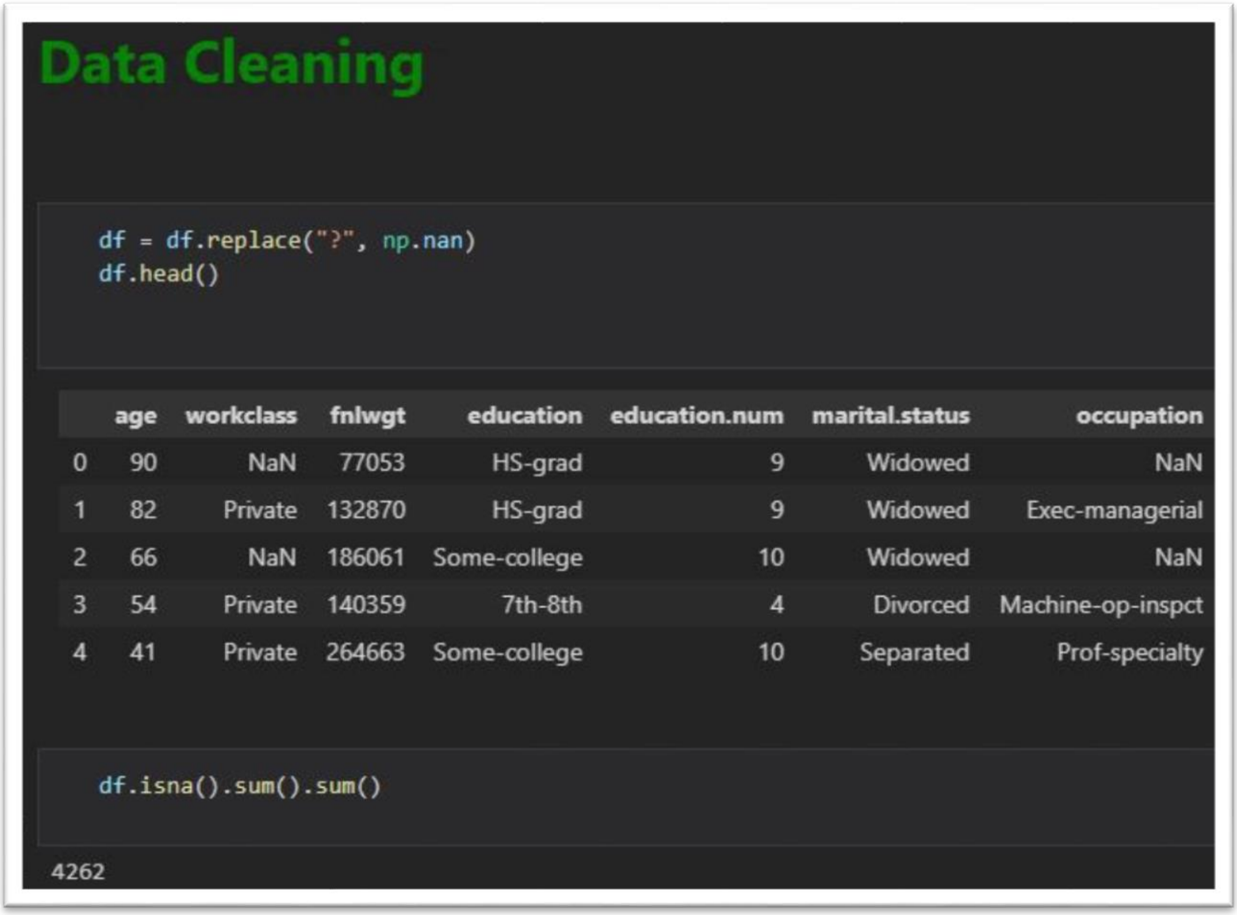
Decision tree learning or induction of decision trees is one of the predictive modelling approaches used in statistics, data mining and machine learning. In our case, we use that technique with the Machine Learning, which builds a model based on sample data, known as training data, in order to make predictions or decisions without being explicitly programmed to do so. The Decision Tree Classifier go from observations about an item to conclusions about the item's target value.


A decision tree is a graph that uses a branching method to illustrate every possible outcome of a decision

SECTION 3: FEEDBACK ABOUT THE PROVIDED INFORMATION

In this section is explained about the difficulties we encountered while working with the provided dataset adult.csv

1. While working with the dataset we encountered a lot missing values with question marks. The total is 4262 missing values (see at the bottom of the screenshot bellow) in which 1836 are in work class column and 583 are in native country column. We dealt with these values by deleting the rows where they were contained and this resulted in decreasing our dataset.



Data Cleaning

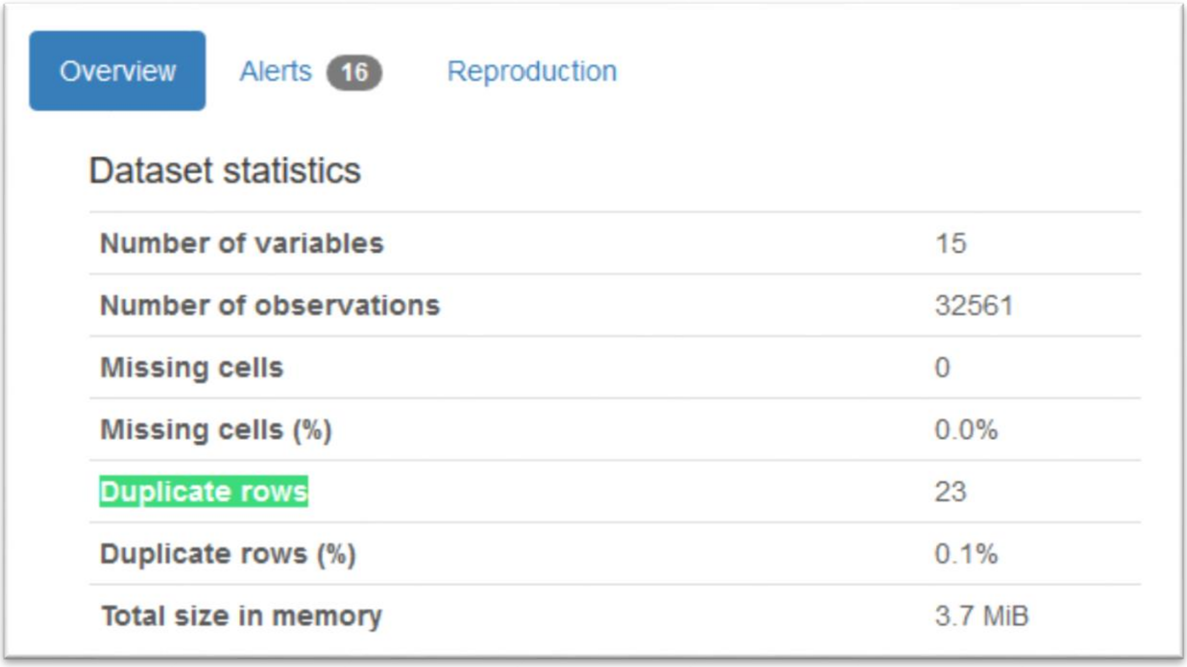
```
df = df.replace("?", np.nan)
df.head()
```

	age	workclass	fnlwgt	education	education.num	marital.status	occupation
0	90	NaN	77053	HS-grad	9	Widowed	NaN
1	82	Private	132870	HS-grad	9	Widowed	Exec-managerial
2	66	NaN	186061	Some-college	10	Widowed	NaN
3	54	Private	140359	7th-8th	4	Divorced	Machine-op-inspct
4	41	Private	264663	Some-college	10	Separated	Prof-specialty

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

```
4262
```

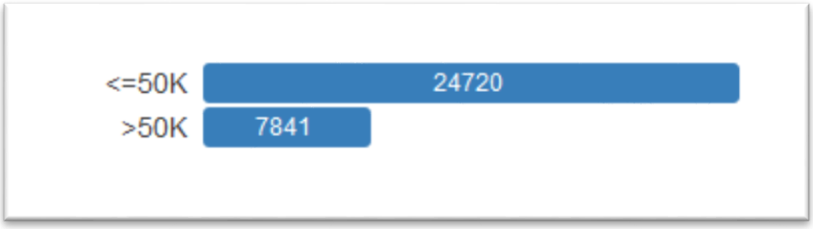
2. There are 23 duplicated rows which is 0.1 % of total dataset, which negatively affects the prediction of our model.



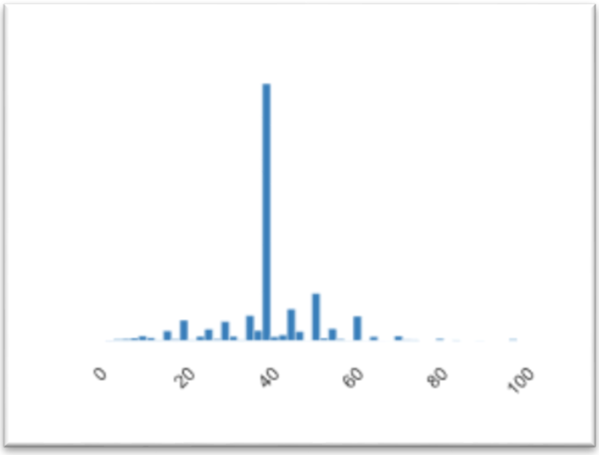
Dataset statistics

Number of variables	15
Number of observations	32561
Missing cells	0
Missing cells (%)	0.0%
Duplicate rows	23
Duplicate rows (%)	0.1%
Total size in memory	3.7 MiB

3. We have a total number of 32561observations, out of which 24720 people are making less than $50,000 per year and the rest of 7841 are making more than $50,000 per year.



4. Our overview also clearly showed that people working 40 hours per week appears more often than people working different hours a week.



SECTION 4: EXPLANATION OF THE CODE

1. Required packages and libraries are being imported at the start of the code. Which includes Pandas, Seaborn, Matplotlib,Keras, Scikitlearn, and Numpy.
2. Read the “adult.csv” file with the Pandas and saved it as a data frame.
3. Data Cleaning → We prepared the data by finding the empty values with question marks and taking them out.
4. Matplotlib and Seaborn are then used to plot and visualize the dataset.
5. The most important part of visualization is to select the features with the highest correlation with the target column. We plotted a correlation chart to have the insights.
6. As some of the features are categorical. We have converted them to dummy variables, added them to our data frame and then removed the original columns.
7. We defined X (features) and y (targets) to feed it to our model.
8. The model is created using “DecisionTreeClassifier()” command in Sci-kit learn.
9. After training the model, it reached accuracy of 95%.
10. With the “prediction” function in training_module, we made a prediction with some data.