|  |  |
| --- | --- |
|  |  |
| Exploratory Data Analysis on Uber Rides Dataset in Python using Anaconda and Jupiter |  |
|  | ABSTRACT |
|  | BYALI AHMED CHOHAN 18SW07SAMEER AKHTARI 18SW107 |
|  |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  | IntroThis is a complete Exploratory Data Analysis on one of the uber personal rides dataset. Which include 1000+ Rows and 7 Columns. *The dataset is only consisted of 1 year data.*  *In this project we went through these four steps:*  *1. Business Problem Definition*  *2. Data Discovery*  *3. Data Preparation*  *4. Data Analysis* Business Problem Definition Requesting and downloading your personal dataset  Uber’s data download feature provides you in-depth information about your rides. You can request access to your data through the following link: <https://myprivacy.uber.com/privacy/exploreyourdata/download>  After your request is done, an email with the download link will be sent to you (usually in the same day). [Unfortunately our uber data was just 3 rows long so we had to download dataset from internet]  **What? When? Where? Who? Which? How? How many? How much?**.  Depending on how many data and features you have, analysis could go to the infinite and beyond. So that’s why (after thinking process) I decided to focus on the following questions: | |  |
|  | #A). How many trips I did over the years?  #B). How many trips were Completed on what Purpose?  #C). For What reason Went to what place...?!? | Image |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | |  |
|  | Data Discovery Importing libraries and dataset.    Checking basic dataset information (data types and dimensions)    ***.rename( )*** method allows you to rename axis labels (indexes and columns). In this case I decided to normalize column names to clean up coding, as long you can columns by typing **<data\_frame>.<column>**    **Use *.head( )*** method to gain more sensibility around data formatting and understand the overall structure of the dataset values.    Taking a look on the **continuous variables**, we notice the presence of some outliers. However **these outliers do not seem to reflect any abnormal value** (e.g. miles), which may let us a little bit more comfortable. | |  |
|  |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| 3. Data Preparation |
|  |
|  | let’s check existence of ***missing values***.    Despite empty values (29 total), there were found 1 records without end date and category. As shown below, these records are insignificant to my dataset, since practically no columns are fulfilled.    So now, let’s get rid of these 3 records before proceeding.       3.1 Data Cleansing: Categorical features Checking categories in product\_type column   3.2 Data Transformation: Handling dates **Dates** usually increase a lot your power of analysis, since you can break it down to different parts and generate insights from different perspectives. As previously shown, our dates features are in fact **object data types**, so we need to **convert them into datetime format**.    Now, let’s break down <start\_date and end\_date> feature into different date parts.   3.3 Feature Engineering: Creating new features #Now Finding ride time   4. Data Analysis In order to organize better my analysis, I will create an additional dataframe, **removing all trips with status CANCELED and DRIVER\_CANCELED**, since they should be disconsider in some questions.    IT’S SHOW TIME…..!  #A). How many trips I did over the years?  #B). How many trips were Completed on what Purpose?      #C). For What reason Went to what place...?!?        🙂  THANK YOU |  |