



## Tutor Marked Assignment (TMA) -Spring 2021

### TM351 Data management and analysis

(Solution document)

Please complete the information below	
Student Name:	Nancy Al Aswad
Student ID:	2180385
Section Number:	
Tutor:	

(Submit a word document of the complete TMA with all answers including your coding work. You can download the notebooks directly to a AsciiDoc file. Also submit all the specified Jupyter notebooks in the form of a "solved" .rar or .zip file)

## Question (1)

-----

1. )

I use Anaconda Framework for run and compel my codes my codes but I need to update Anaconda Prompt with Pip commands such as:

- python -m pip install --upgrade pip
- pip install -U pandasql

After these steps I run all my notebooks in the solved assets folder and added comments as required using Anaconda Navigator

2. )

[I download Open Refine](#): which is a powerful tool for working with messy data: cleaning it; transforming it from one format into another; and extending it with web services and external data.<sup>1</sup>

I open the powerful tool then I choose file from my solved assets folder and access the notebook (4 ) to use data folder and open the sales book file .

Then I took the required screenshots and save it in the solved assets folder

---

<sup>1</sup> <https://openrefine.org/>

## Question (2)

-----

### 1. Question Summary

#### Introduction for data set content:

This dataset is originally from the National Institute of Diabetes and Digestive and Kidney Diseases. The objective of the dataset is to diagnostically predict whether a patient has diabetes, based on certain diagnostic measurements included in the dataset. Several constraints were placed on the selection of these instances from a larger database. In particular, all patients here are females at least 21 years old of Pima Indian heritage.<sup>2</sup>

From the data set in the (.csv) File We can find several variables, some of them are independent (several medical predictor variables) and only one target dependent variable (Outcome).

So I think the proper way to describe our dataset maybe from its rows (768 observations) and its columns (9 attributes) as below:

#### Information about dataset Attributes

- Pregnancies: To express the Number of pregnancies
- Glucose: To express the Glucose level in blood
- BloodPressure: To express the Blood pressure measurement
- SkinThickness: To express the thickness of the skin
- Insulin: To express the Insuline level in blood
- BMI: To express the Body mass index
- DiabetesPedigreeFunction: To express the Diabetes percentage
- Age: To express the age
- Outcome: To express the final result 1 is YES o is NO

---

<sup>2</sup> <https://www.kaggle.com/uciml/pima-indians-diabetes-database>

## 2. Question Assessment for each data set Quality:

=====

Type of quality	Assessment	The Amount of dirtiness
<b>Validity</b>	We can find all data values are valid	Here No invalid data found
<b>Accuracy</b>	We can find all data attributes are accurate	There are no inaccurate data
<b>Completeness</b>	We can find a lot of missing data in different attributes:  1- Glucose  2- Blood Pressure  3- Skin Thickness  4- Insulin  5- BMI	We can also find the same amount of dirtiness in the python code in zip folder as Jupiter notebook  1. Glucose got 5 missing values. <ul style="list-style-type: none"><li>amount of dirtiness = <math>5/768 = 0.65 \%</math></li></ul> 2- Blood Pressure got 35 missing values. <ul style="list-style-type: none"><li>amount of dirtiness = <math>35/768 = 4.5 \%</math></li></ul> 3- Skin condition got 227 missing values. <ul style="list-style-type: none"><li>amount of dirtiness = <math>227/768 = 29.5 \%</math></li></ul> 4- Insulin got 374 missing values. <ul style="list-style-type: none"><li>amount of dirtiness = <math>374/768 = 48.7 \%</math></li></ul> 5- BMI got 11 missing values. <ul style="list-style-type: none"><li>amount of dirtiness = <math>11/768 = 1.4 \%</math></li></ul>
<b>Consistency</b>	We can find all data are consistent	Here No inconsistency data found
<b>Uniformity</b>	We can find all dataset has uniform values for each attribute	Here No ununiformed data found

## Question (3)

-----

### ***1) Executive Summary:***

In this project we discuss the Pima Indians Diabetes Dataset, by investigating and discussing the relationships between selected dependent & independent variables.

The primary Source of data set is the [Kaggle](#) Community which is made up of data scientists and machine learners from all over the world with a variety of skills and backgrounds. We strongly believe that our community and the future of the field are brighter when we embrace differences<sup>3</sup>.

To show the relationships between selected variables we use appropriate visualizations with Pandas data frame and use proper measure for correlation, So, we make (5) conclusions by making (5) different questions on the data set.

### ***2) Aims and objectives:***

- Aims:
  - 1.1. Display different correlation between columns.
  - 1.2. Display statistics for each data sets.
  - 1.3. Visualizing the relation between variable.
- Objectives:
  - a) Using ANACONDA frame work to import CSV files into PANDAs data frame.
  - b) Importing required libraries into ANACONDA.
  - c) Using SQL language and Pandas data frame
  - d) Using different visualization models to show the correlations between variables in specific conditions or criteria.

---

<sup>3</sup> <https://www.kaggle.com/community-guidelines>

### ***3) The Research questions:***

#### **Identifying the dependent and independent variables:**

**1. Independent Variables:** Pregnancies, Glucose, Blood Pressure, Skin Thickness,  
Insulin, BMI, Diabetes Pedigree Function, Age

**2. Dependent Variable: Outcome** and we can describe it

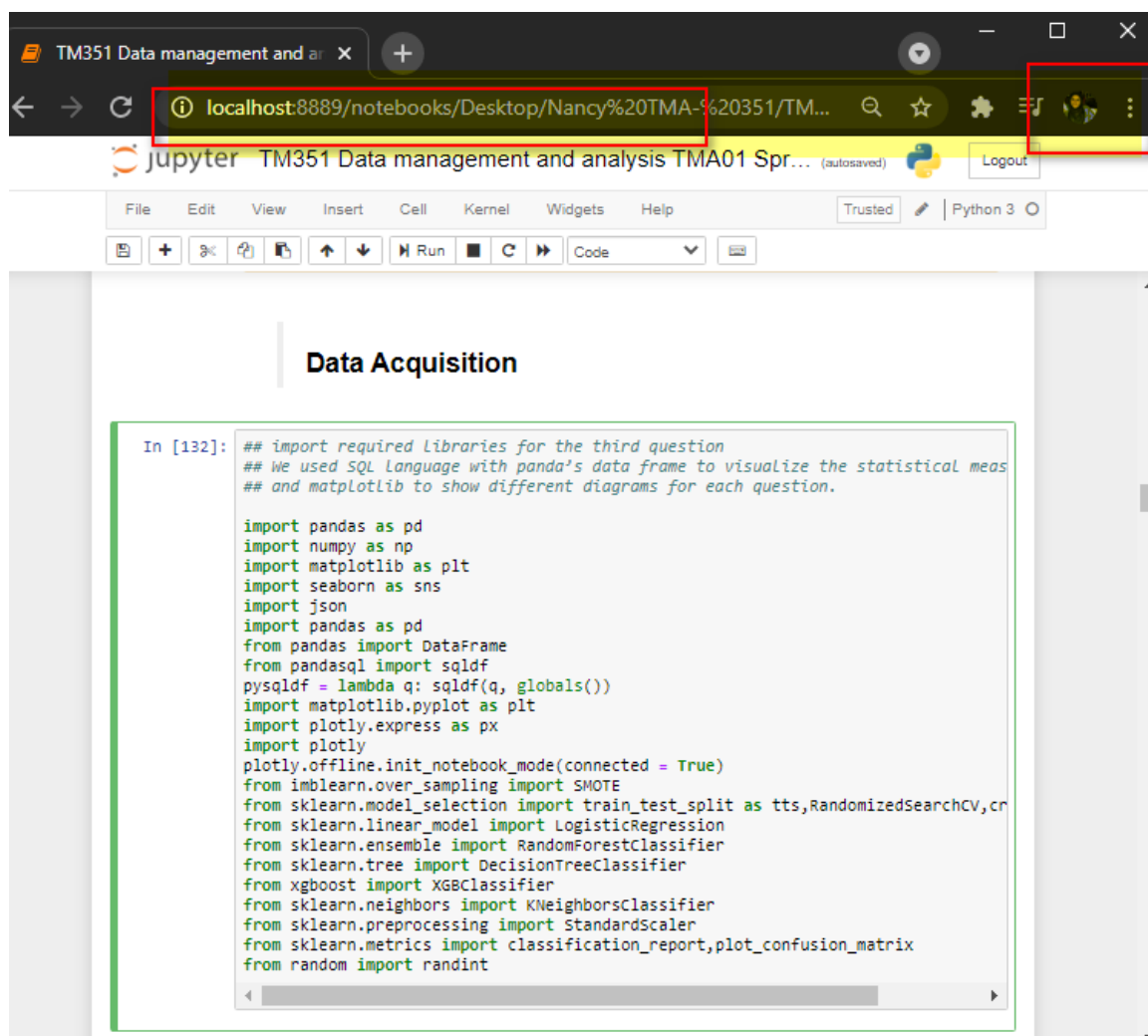
**as** ( 0 = Do not have Diabetes // 1 = Do have Diabetes)

- **I choose to investigate the relation between** (Pregnancies, BMI, Glucose, Insulin, Blood Pressure) variables for patients **according to following questions:**

- 1- Investigate the relation between Pregnancies and diabetes with display average Pregnancies for patients.
- 2- Investigate the relation between BMI and diabetes and average BMI for patients.
- 3- Investigate the relation between Glucose and diabetes and average Glucose level for patients.
- 4- Investigate the relation between Insulin and diabetes and average Insulin for patients.
- 5- Investigate the relation between Blood pressure and diabetes and average Blood pressure for patients.

#### 4) The Analysis and findings:

- The questions analysis: We used panda's data frame with SQL language to visualize the statistical measures and matplotlib for display different diagrams for each question, So, we begin with import different libraries to ANACONDA:

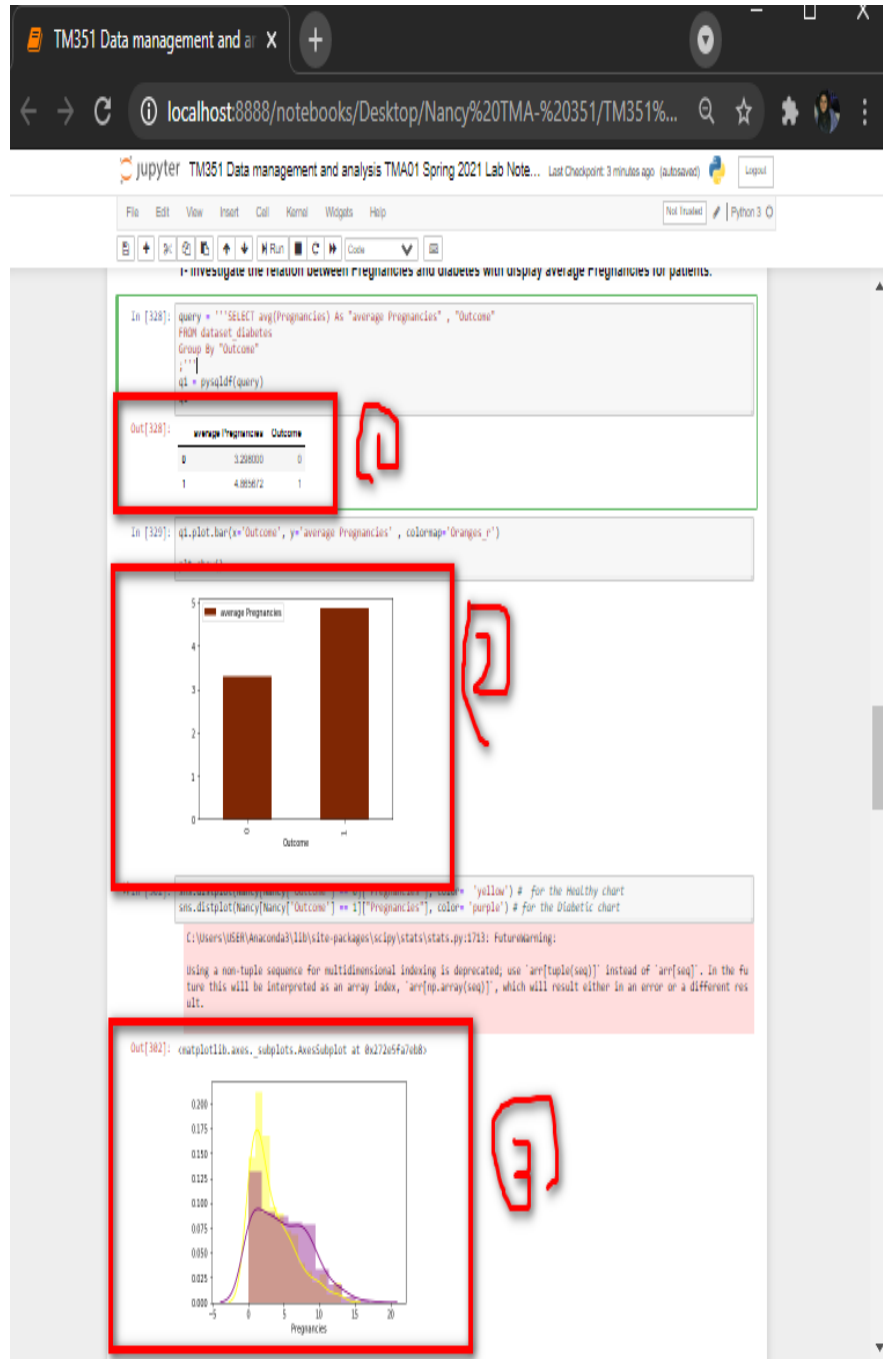


The screenshot displays a Jupyter Notebook window titled "TM351 Data management and analysis TMA01 Spr... (autosaved)". The browser address bar shows the URL "localhost:8889/notebooks/Desktop/Nancy%20TMA-%20351/TM...". The notebook interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with icons for file operations, running code, and viewing output. The main content area is titled "Data Acquisition" and contains a code cell labeled "In [132]:". The code cell contains the following Python code:

```
## import required Libraries for the third question
## We used SQL Language with panda's data frame to visualize the statistical meas
## and matplotlib to show different diagrams for each question.

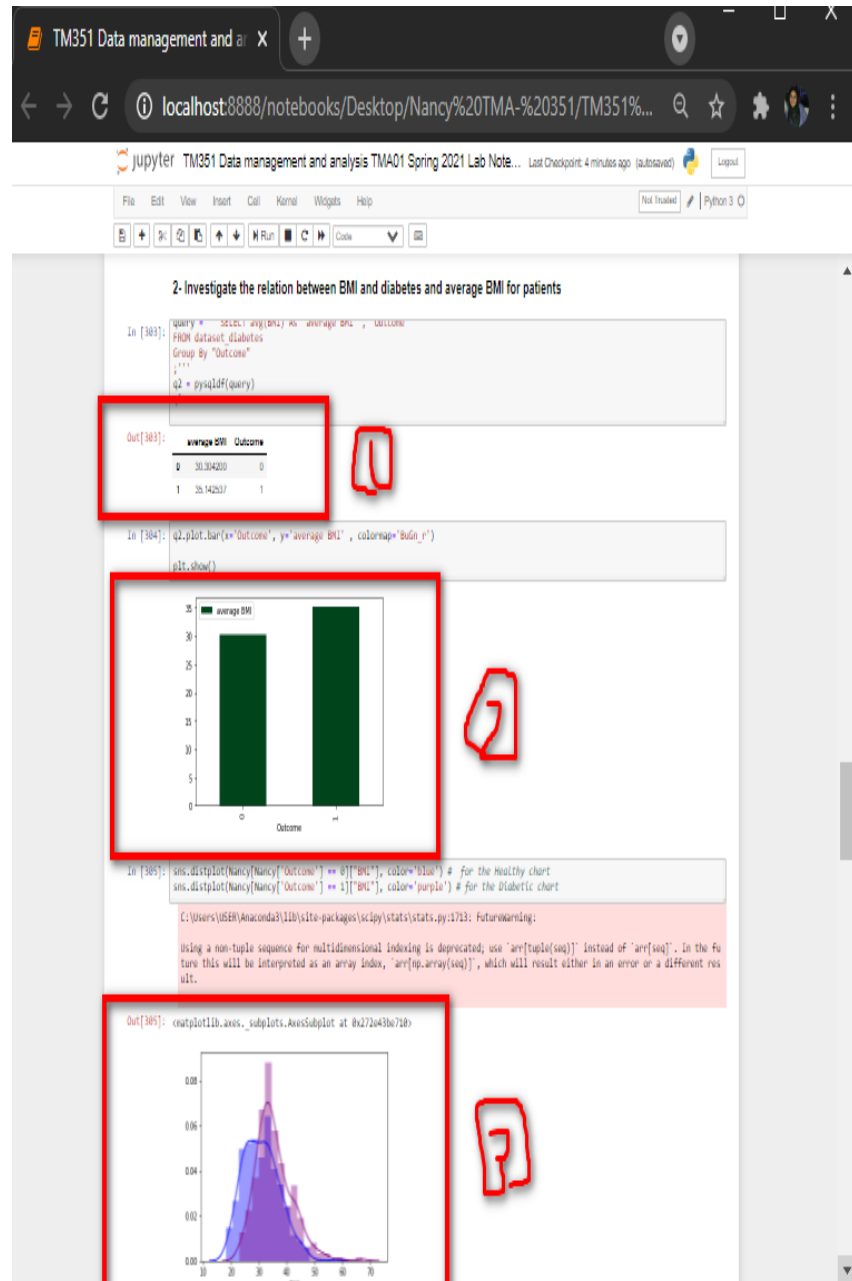
import pandas as pd
import numpy as np
import matplotlib as plt
import seaborn as sns
import json
import pandas as pd
from pandas import DataFrame
from pandasql import sqldf
pysqldf = lambda q: sqldf(q, globals())
import matplotlib.pyplot as plt
import plotly.express as px
import plotly
plotly.offline.init_notebook_mode(connected = True)
from imblearn.over_sampling import SMOTE
from sklearn.model_selection import train_test_split as tts, RandomizedSearchCV, cr
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.tree import DecisionTreeClassifier
from xgboost import XGBClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import classification_report, plot_confusion_matrix
from random import randint
```

1. Investigate the relation between Pregnancies and diabetes with display average Pregnancies for patients.

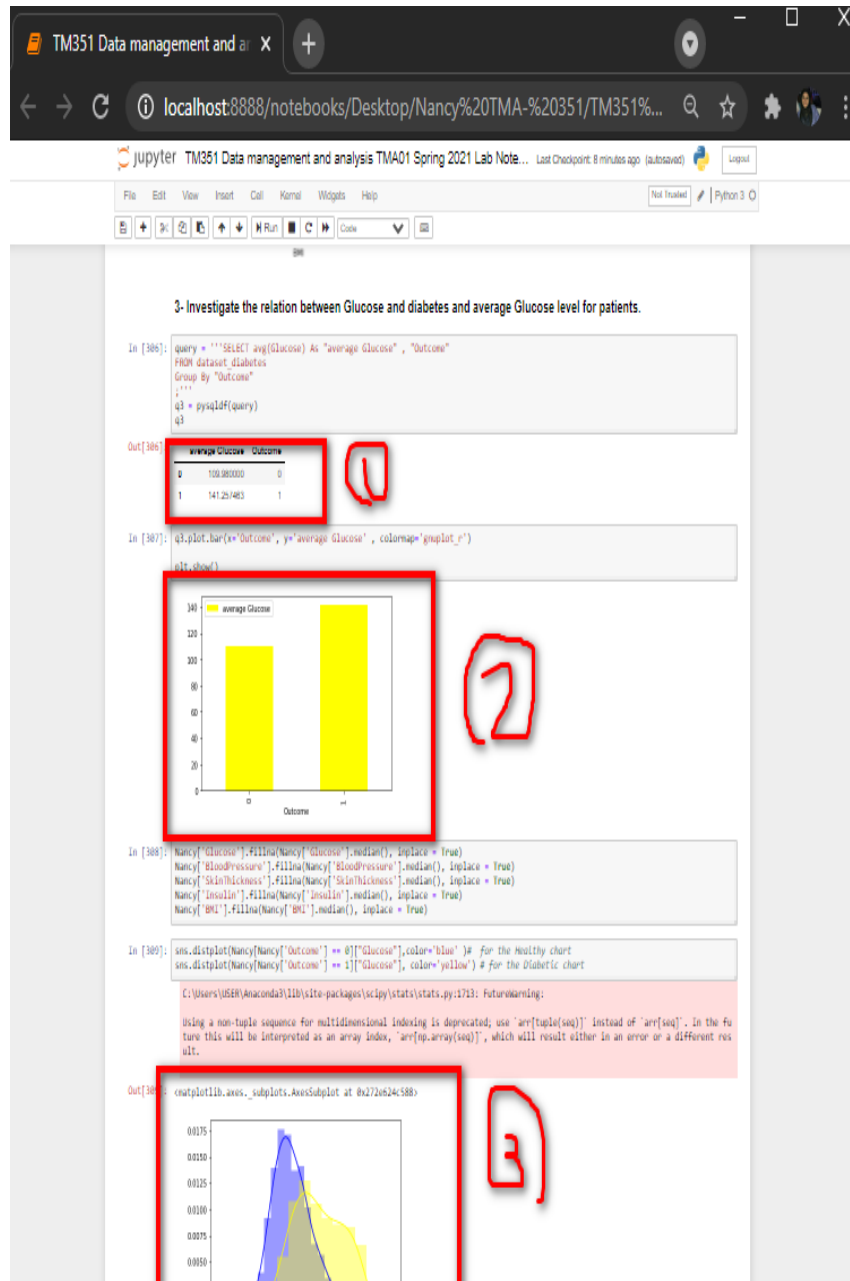




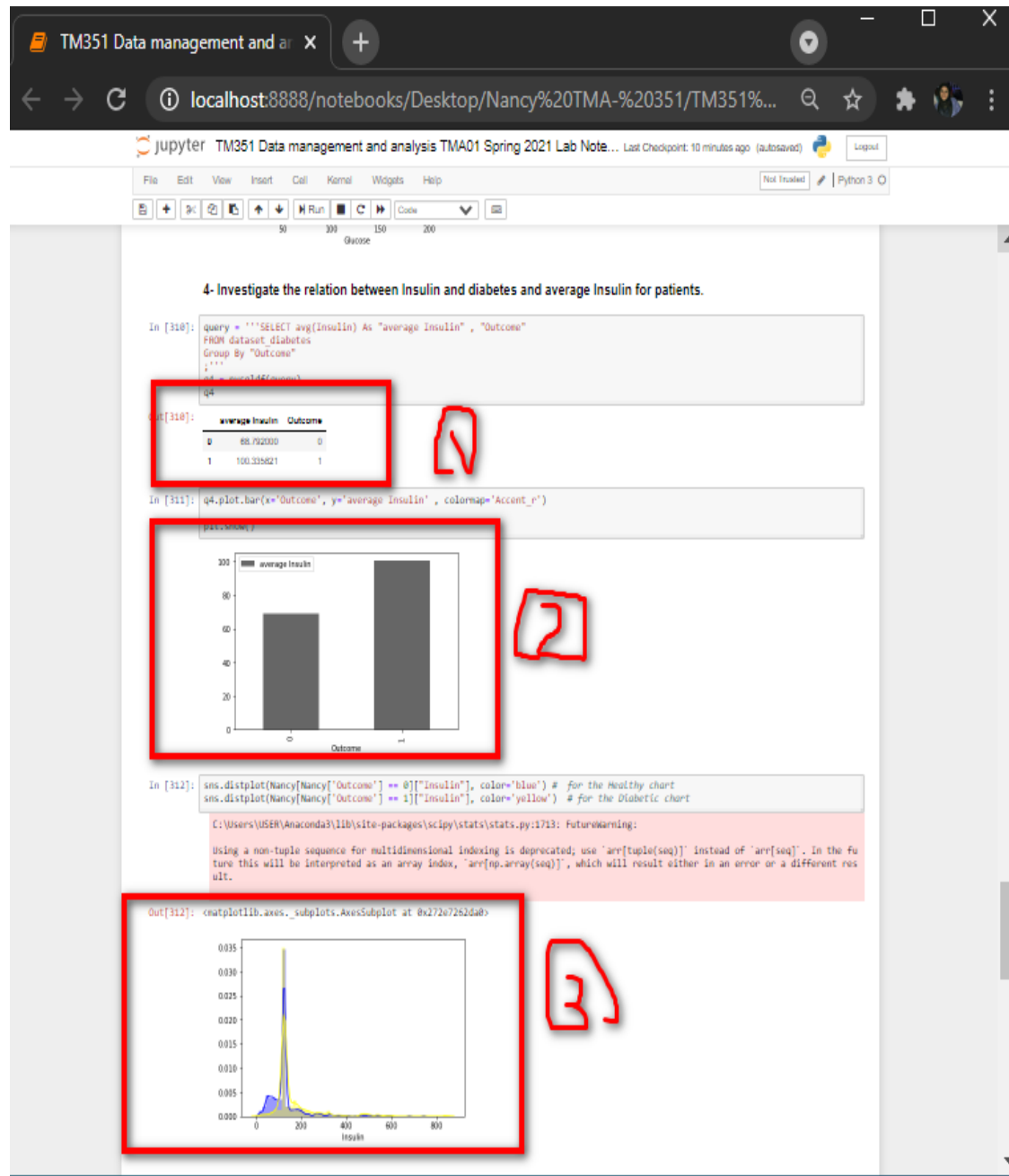
- Investigate the relation between BMI and diabetes and average BMI for patients.



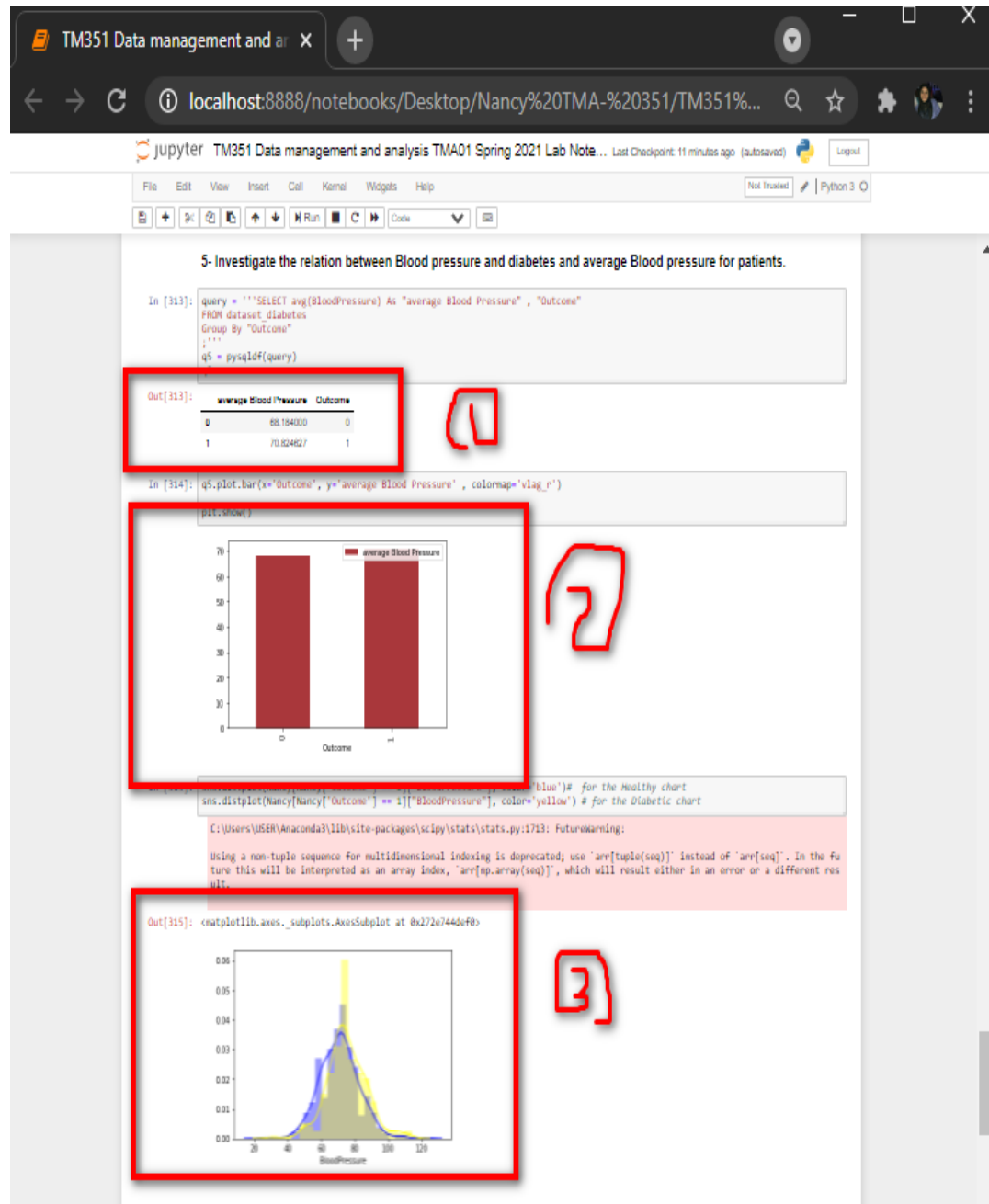
3. Investigate the relation between Glucose and diabetes and average Glucose level for patients.



4. Investigate the relation between Insulin and diabetes and average Insulin for patients.



- Investigate the relation between Blood pressure and diabetes and average Blood pressure for patients.



## ***Conclusion for Finding:***

---

In the conclusion section we will explain our Data Analysis for each independent variable and display its effect on Average level of Diabetic patients

1. **Pregnancies**: We can conclude that pregnancy may not be a prime factor to consider.
2. **BMI**: We can conclude that fat people suffers from diabetes more than thin people.
3. **Glucose**: Glucose range over 145 mg/dl in the population sampled indicates the presence of diabetes.
4. **Insulin**: Diabetic patient have a sharp peak between ( $\sim 160$  -180)  $\mu$ U/ml range.
5. **Blood Pressure**: The peak values for the diabetic patients occurs at  $\sim 72$  to 76 mm Hg and the range of diabetic range falls between 62 to 95 mm Hg while healthy people have till 80 mmHg

## ***5) The Reflection:***

---

- Project Experience: Good experience in working with jupyter note books, looking for relationships between different variables, Filtration of data to minimize the number of records, Splitting the work into different tasks.
- 1- Learning Outcomes: Using Anaconda framework, Python coding, Open Refine, Mongo DB, VM TM351 software
- What went Well: All tasks done
- What went wrong: The VM TM351 software was hard to work with and some codes did not running as desired
- Future benefits: Learning how to analysis using Python, SQL.

## 6) The References:

=====

- Pihlak, Martin. ["PostgreSQL @Skype" \(PDF\)](#). [wiki.postgresql.org](#). Retrieved January 16, 2019.
- ["postgresql-client-10.5p1 – PostgreSQL RDBMS \(client\)"](#). [OpenBSD ports](#). October 4, 2018.  
  
Retrieved October 10, 2018.
- Kenny Gorman. ["MongoDB 3.0 WiredTiger Compression and Performance"](#).  
  
[Objectrocket.com/](#). [Archived](#) from the original on June 16, 2017. Retrieved June 28, 2017.
- [scalegrid.io. "Atomicity, isolation & concurrency in MongoDB"](#). [scalegrid.io](#). [Archived](#) from the original  
  
on September 10, 2017. Retrieved July 5, 2017.
- Jackson, Joab (Feb 5, 2013). ["Python gets a big data boost from DARPA"](#). [networkworld](#).  
  
Retrieved October 30, 2014.

## Question (4)

-----

*a) Compare any three trending data visualization tools.*

=====

- **Plotly<sup>4</sup>**

Plotly enables more complex and sophisticated visualizations, thanks to its integration with analytics-oriented programming languages such as Python, R and Matlab. It is built on top of the open source d3.js visualization libraries for JavaScript, but this commercial package (with a free non-commercial license available) adds layers of user-friendliness and support as well as inbuilt support for APIs such as Salesforce.

- **Tableau<sup>5</sup>**

Tableau is often regarded as the grand master of data visualization software and for good reason. It is particularly well suited to handling the huge and very fast-changing datasets which are used in Big Data operations, including artificial intelligence and machine learning applications, thanks to integration with a large number of advanced database solutions including Hadoop, Amazon AWS, My SQL, SAP and Teradata. Extensive research and testing have gone into enabling Tableau to create graphics and visualizations as efficiently as possible, and to make them easy for humans to understand.

- **Fusion Charts<sup>6</sup>**

This is a very widely-used, JavaScript-based charting and visualization package that has established itself as one of the leaders in the paid-for market. It can produce 90 different chart types and integrates with a large number of platforms and frameworks giving a great deal of flexibility.

---

<sup>4</sup> <https://plotly.com/>

<sup>5</sup> <https://www.tableau.com/>

<sup>6</sup> <https://www.fusioncharts.com/>

## ***b) What is data Pre-Processing?***

=====

It is technique for a data mining that involves transforming raw data into an understandable format because in the Real-world data is not always complete, consistent, and often it is lacking of certain behaviours or trends, and it might contains errors, So, Data pre-processing method help us resolving many issues.

**And to make Data pre-processing method we need to work in a series of steps such as:**

- 1- **Data Cleaning:** Here we make some smoothing for noisy data, filling in deleting rows or missing values, to resolving the inconsistencies in the data.
- 2- **Data Integration:** Here we make representations and put different data together to conflicts within the data are resolved.
- 3- **Data Transformation:** Here we make normalized and generalized. to ensures that no data is redundant, and it is all stored in a single place, and all the dependencies are logical.
- 4- **Data Reduction:** Data reduction step aims to present a reduced representation of the data in a data warehouse.
- 5- **Data Discretization:** This step involves the reduction of a number of values of a continuous attribute by dividing the range of attribute intervals.
- 6- **Data Sampling:** Sampling techniques can be used to select and work with just a subset of the dataset, provided that it has approximately the same properties of the original one.



### c) *Briefly explain Google Colab.*

=====

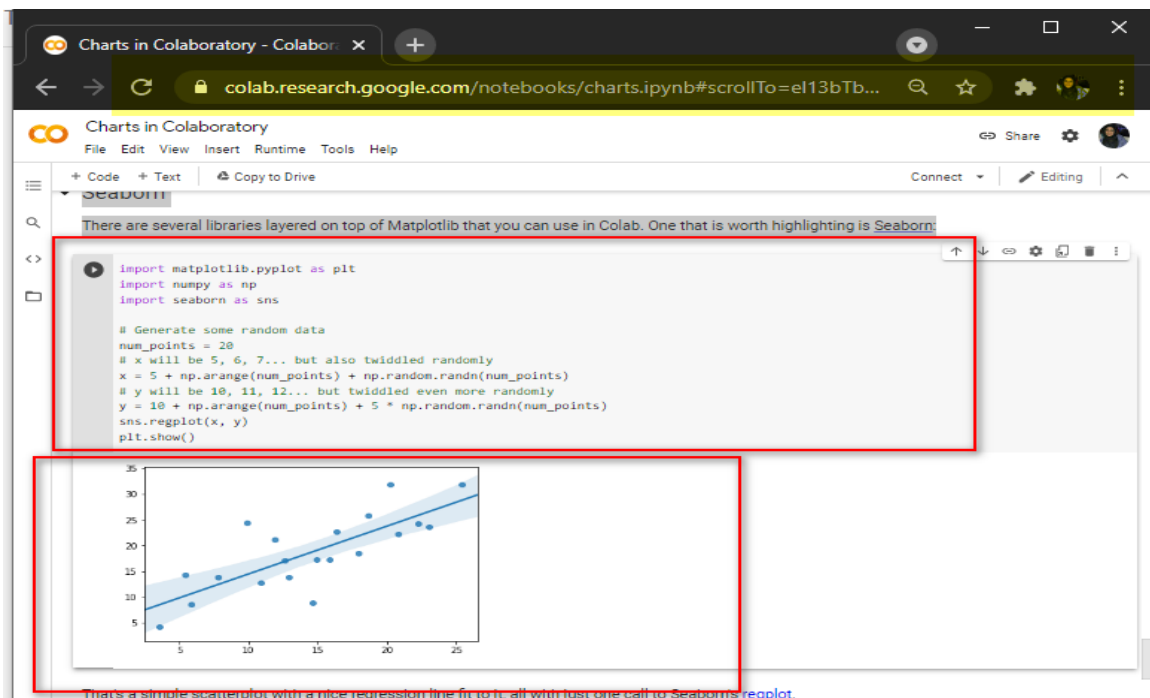
- Google Colaboratory<sup>7</sup> is a hosted Jupyter notebook environment that is free to use and requires no setup. It's a free cloud service and it supports free GPU!

You can simply improve your **Python** programming language coding skills **OR** develop deep learning applications using popular libraries such as **Keras**, **TensorFlow**, **PyTorch**, and **OpenCV**.

Now all these libraries are already installed for you. So there's no need to go through the hassle of installing the libraries. Just get right onto coding!

The most important feature that distinguishes Colab from other free cloud services is that it provides GPU and is totally free. You need no membership or a credit card to use it. Colab provides you a free **CPU** and a **GPU**.

- Here is **example** I use with it Google Colab and **I also use and run it in my Jupyter notebook environment**



<sup>7</sup> [https://medium.com/@iHrishi\\_mane/what-is-google-colab-eb1e718646ce](https://medium.com/@iHrishi_mane/what-is-google-colab-eb1e718646ce)



### Question 4 - Project (15 marks)

I use the below code as example of using numpy to generate some random data, and using matplotlib to visualise it.

```
In [336]: ## The site : https://colab.research.google.com/notebooks/charts.ipynb#scrollTo=e113bTbyPRw4

import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns

# Generate some random data
num_points = 20
# x will be 5, 6, 7... but also twiddled randomly
x = 5 + np.arange(num_points) + np.random.randn(num_points)
# y will be 10, 11, 12... but twiddled even more randomly
y = 10 + np.arange(num_points) + 5 * np.random.randn(num_points)
sns.regplot(x, y)
plt.show()
```

C:\Users\USER\Anaconda3\lib\site-packages\scipy\stats\stats.py:1713: FutureWarning:

Using a non-tuple sequence for multidimensional indexing is deprecated; use `arr[tuple(seq)]` instead of `arr[seq]`. In the future this will be interpreted as an array index, `arr[np.array(seq)]`, which will result either in an error or a different result.

