D206 Data Cleaning: Performance Assessment

Arjun Gupta

Student ID: 012296064

MSDA, Western Governors University

A.

1. What factors affect Diabetes? This is an important question to ask due to the magnitude of people affected by diabetes, and this would allow doctors to find preventative measures to reduce the risk of diabetes.
2. All the types of variables in the data set are:-
    - CaseOrder- (quantitative data type)
        - It is the ID variable that is used to keep the same order found in the original file.
        - Ex- 1
    - Customer_id- (qualitative data type)
        - This ID variable is used as a unique identifier for patients.
        - Ex- C412403
    - Interaction-(qualitative data type)
        - Both this and UID variables are related to patient transactions, procedures, and admissions.
        - Ex- 8cd49b13-f45a-4b47-a2bd-173ffa932c2f
    - UID-(qualitative data type)
        - Like Interaction, the UID variable is related to patient transactions, procedures, and admissions.
        - Ex- 3a83ddb66e2ae73798bdf1d705dc0932
    - City-(qualitative data type)
        - The city in which the patient resides in as written on the billing statement.
        - Ex- Eva

- State-(qualitative data type)
    - The state in which the patient resides in as written on the billing statement.
    - AL
- County-(qualitative data type)
    - The county in which the patient resides in as written on the billing statement.
    - Morgan
- Zip-(qualitative data type)
    - The zip code in which the patient resides in as written on the billing statement.
    - Ex- 35621
- Lat-(quantitative data type)
    - The GPS coordinates in which the patient resides in as written on the billing statement as latitude.
    - Ex- 34.3496
- Lng-(quantitative data type)
    - The GPS coordinates in which the patient resides in as written on the billing statement as longitude.
    - Ex- 34.3496
- Population-(quantitative data type)
    - The number of people within a 1-mile radius of the patient based on census data
    - Ex- 2951

- Area-(qualitative data type)
    - The type of area (urban, suburban, or rural), based on the unofficial census.
    - Ex- Suburban
- Timezone-(qualitative data type)
    - The time zone that the patient resides in. It based on sign-up information.
    - Ex- America/Chicago
- Job-(qualitative data type)
    - The job that the patient or primary insurance holder does.
    - Ex- Psychologist, sport and exercise
- Children-(quantitative data type)
    - The number of children in the patient's house.
    - Ex- 1
- Age-(quantitative data type)
    - The patient's age.
    - Ex- 53
- Education-(qualitative data type)
    - The highest degree earned by the patient
    - Ex- Some College, Less than 1 Year
- Employment-(qualitative data type)
    - This variable tells us the current employment status of the patient.
    - Ex- Full Time

- Income-(quantitative data type)
    - Annual income of the patient or primary insurance holder.
    - Ex- 86575.93
- Marital-(qualitative data type)
    - Tells us whether the patient or primary insurance holder is married or not.
    - Ex- Divorced
- Gender-(qualitative data type)
    - It tells us if the patient is male, female, or nonbinary.
    - Ex- Male
- ReAdmis-(qualitative data type)
    - It tells us if the patient got readmitted within a month of release. (yes, or no)
    - Ex- No
- VitD_levels-(quantitative data type)
    - The vitamin D levels of the patient in ng/ml
    - 17.80233049
- Doc_visits-(quantitative data type)
    - Number of times the physician visited the patient during the initial hospitalization.
    - Ex- 6
- Full_meals_eaten-(quantitative data type)
    - The number of complete meals the patient ate. If the meals were partially eaten, then the count is 0.

    - Ex- 0
- VitD_supp-(quantitative data type)
    - The number of vitamin D supplements that were administered into the patient.
    - Ex- 0
- Soft_drink-(qualitative data type)
    - Whether the patient drinks 3 or more soft drinks a day.
    - Ex- No
- Initial_admin-(qualitative data type)
    - The reason why the patient was initially admitted into the hospital. (emergency admission, elective admission, or observation).
    - Ex- Emergency Admission
- HighBlood-(qualitative data type)
    - Whether the patient has high blood pressure or not.
    - Ex- Yes
- Stroke-(qualitative data type)
    - Whether the patient has had a stroke or not.
    - Ex- No
- Complication_risk-(qualitative data type)
    - The risk the patient has which was assessed by a primary patient assessment (it can be low, medium, or high).
    - Ex- Medium

- Overweight-(qualitative data type)
    - Is the patient considered overweight based on age, gender, and height.
    - Ex- 0
- Arthritis-(qualitative data type)
    - Whether the patient has arthritis or not.
    - Ex- Yes
- Diabetes-(qualitative data type)
    - Whether the patient has diabetes or not.
    - Ex- Yes
- Hyperlipidemia- (qualitative data type)
    - Whether the patient has hyperlipidemia or not.
    - Ex- No
- BackPain-(qualitative data type)
    - Whether the patient has back pain or not.
    - Ex- Yes
- Anxiety-(qualitative data type)
    - Whether the patient has an anxiety disorder.
    - Ex- 1
- Allergic_rhinitis-(qualitative data type)
    - Whether the patient has allergic rhinitis or not.
    - Ex- Yes
- Reflux_esophagitis-(qualitative data type)
    - Whether the patient has reflux esophagitis or not.

    - Ex- No
- Asthma-(qualitative data type)
    - Whether the patient has asthma or not.
    - Ex- Yes
- Services-(qualitative data type)
    - What service did the patient receive when hospitalized (CT scan, MRI, etc.)
    - Ex- Blood Work
- Initial_days-(quantitative data type)
    - The number of days the patient stayed when they initially visited the hospital.
    - Ex- 10.58576971
- TotalCharge-(quantitative data type)
    - This is the amount that the patient is charged daily. It is calculated by taking the total charge divided by the number of days they have been hospitalized. This amount only shos the typical amount charged to patients and does not include specialized treatments.
    - Ex- 3191.048774
- Additional_charges-(quantitative data type)
    - The amount charged to the patients for miscellaneous procedures, treatments, medicines, anesthesiology, etc.
    - Ex- 17939.40342

The following “Item” variables are responses to a survey from a scale of 1 to 8 with 1 being the most important and 8 being the least important

- Item1-(qualitative data type)
    - How important is timely admission?
    - Ex- 3
- Item2-(qualitative data type)
    - How important is timely treatment?
    - Ex- 3
- Item3-(qualitative data type)
    - How important is timely visits?
    - Ex- 2
- Item4-(qualitative data type)
    - How important is reliability in hospitals?
    - Ex- 2
- Item5-(qualitative data type)
    - How important is it to have options for the patients in hospitals?
    - Ex- 4
- Item6-(qualitative data type)
    - Is it important to have better hours of treatment?
    - Ex- 3
- Item7-(qualitative data type)
    - How important is it for the patients to have a courteous staff?
    - Ex- 3

- Item8-(qualitative data type)
    - How important is it for the doctor to show that they are actively listening?
    - Ex- 4

B

1. The plan that I will use to clean the data requires a few steps. First, I will check the data for duplicates. This will be done by checking the duplicates of variables that are key in the data (this includes CaseOrder, Customer_id, Interaction, and UID). I am using the keys labeled as such in the Medical Data Dictionary provided.

    After checking for duplicates, I will check for missing data. First, I will check how many rows are empty in each variable; if there are empty rows, then I will use a histogram chart to find the shape of the graph and, depending on the shape, will use the mean, median, or mode to impute data into variables with missing data.

    Upon completion, I will look at the numerical variables to see if there are any outliers in each variable. I will create an array using the data dictionary provided to us for all the numeric variables and make a boxplot for each variable. Then accordingly, I will implement a technique suitable for treating outliers (imputation, retain, exclude, or remove the outliers).

    Then I will adjust the categorical variables. For boolean variables (Yes or No), I will use Ordinal Encoding to change them to (1 or 0 respectively).

    Then I will check for any other issues, such as the number of decimal places, or any other issues I may want to change.

2. The reason that I am using this method for checking for duplicates is if I were to look for duplicates on any variables that are not unique (such as a boolean variable), there would be a huge number of duplicates which if I were to make any changes to these duplicates this would cause a significant change in the final result of the data due to the large number of changes being made.

    Using this method for missing data is the best because when creating histograms for each variable with missing data, we can get a better idea of the data distribution. The reason knowing the distribution is essential is that dictates whether we impute the missing data based on the mean, median, or mode. The mean will be used if the shape of the data is a standard or uniform distribution, the median will be used if the shape is skewed or a bi-modal distribution and the mode will be used if the data is categorical or bi-modal. This ensures that the data distribution does not change before and after imputing the data. This method is also great for checking the distribution afterward to see if it changes. We use imputation of the data to retain the entire dataset. By imputing the missing values using this method, we can prevent the loss of information which would allow my analysis to be more robust and create a better representation of the data. This method is also helpful as imputing details based on these measures of central tendency instead of imputing randomly prevents bias from being created by maintaining the original shape of the dataset, which leads to more reliable predictions.

    The method for checking outliers was also chosen due to the flexibility of the technique. Depending on the data found, we can use different methods to ensure the data's overlying meaning does not change. It was also chosen as a boxplot to distribute the data visually. The reason I decided to use imputation of the outliers is because by

imputing, we can reduce the noise, allowing us to create a more accurate representation of the data. I also chose this method as it can improve our understanding of central trends by imputing the outliers, we can highlight and find more relevant patterns and trends in the data.

I chose to use Ordinal encoding for the boolean values found to change all boolean variables into similar data types for ease of understanding and better potential use in the next steps in the data analytics lifecycle.

I also want to adjust the decimal places due to the data in some variables having too many decimals, and that much information hurts readability.

3. The programming language that I used to clean my data was Python. Due to having used Python in previous projects, the syntax was easier to follow and apply. Another reason is the ability to create scripts and loops, which help speed up repetitive processes. I used the pandas, numpy, matplotlib, and seaborn libraries for data cleaning. I used pandas to import and work with data sets, numpy for arrays and other numerical functions, matplotlib to create histograms, and seaborn to create boxplots which further helped analyze and clean the dataset.

C.

1. When cleaning the data, I found no duplicates in any of the key variables in the dataset. Due to this, I did not need to remove any duplicate values.
   For the missing values, I found that there were 2588 missing rows in the children variable, 2414 rows in the Age variable, 2464 rows missing in the Income, 2467 missing

in the Soft_drink, 982 missing in the Overweight, 984 missing in Anxiety, and 1056 missing in Initial_days.

When it came to outliers, I found that variable Lat contained the range of outliers are from 15 to 25 and 50 to 75. The variable Lng contained the range of outliers from -180 to -120. The variable Population contained the range of outliers from 35000 to 130000. The variable Children contained the range of outliers from 7 to 11. The variable Income contained the range of outliers from 75000 to 225000. The variable Population contained the range of outliers from 35000 to 130000. The variable VitD_levels contained a range of outliers from 40 to 60. The variable Full_meals_eaten contained a range of outliers from 6 to 7. The variable VitD_supp contained a range of outliers from 3 to 5. The variable TotalChange contained the range of outliers from 14000 to 22500. The variable Additional_charges contained the range of outliers from 27500 to 30000. The variable Item1 contained the range of outliers from 6 to 8 and outliers at 1.

I also found that there were several categorical variables that needed to be converted to numerical. These included HighBlood, Stroke, Arthritis, Diabetes, Hyperlipidemia, BackPain, Anxiety, Allergic_rhinitis, Asthma, and Reflux_esophagitis.

There were several numeric variables that had too many decimal points. So I converted VitD_levels, Initial_days, TotalCharge, and Additional_charge to 2 decimal places.

2. Since there were no duplicates to be treated, there was no treatment method for this step. I did, however find this by using the duplicated() method on CaseOrder, Customer_id, Interaction, and UID to see if the variables which were keys had any duplicates.

When it came to missing values, I used the isnull() method with the sum() method to count the number of missing values in each variable. I found 2588 missing values in the Children variable. I used matplotlib for the histograms as this was a convenient way of making histograms. Judging by the shape of the distribution of the graph, I found that the data was skewed to the right. Due to this, I imputed the missing data using the median. I found 2414 missing values in the Age variable. Judging by the shape of the distribution of the graph, I found that the data was uniform. Due to this, I imputed the missing data using the mean. I found 2464 missing values in the Income variable. Judging by the shape of the distribution of the graph, I found that the data was skewed to the right. Due to this, I imputed the missing data using the median. I found 2467 missing values in the Soft_drink variable. Judging by the shape of the distribution of the graph, I found that the data was bimodal. Due to this, I imputed the missing data using the mode. I found 982 missing values in the Overweight variable. Judging by the shape of the distribution of the graph, I found that the data was bimodal. Due to this, I imputed the missing data using the mode.I found 984 missing values in the Anxiety variable. Judging by the shape of the distribution of the graph, I found that the data was bimodal. Due to this, I imputed the missing data using the mode. I found 1056 missing values in the Initial_days variable. Judging by the shape of the distribution of the graph, I found that the data was bimodal. Due to this, I imputed the missing data using the mode.

When it comes to the outliers, for the Lat, Lng, Population, Income, VitD_levels, and TotalCharge variables on the histogram, I treated the outliers using imputation as I wished to preserve the sample size. For the Children, Full_meals_eaten, and

Additional_charges, I retained the outliers as I wanted to maintain the diversity of the dataset.

For the boolean variables, I used ordinal encoding to convert categorical variables to numeric ones. I created an array with all the variables that needed to be replaced and then changed them to numerical variables. I also changed the VitD_levels, Initial_days, TotalCharge, and Additional_charge to 2 decimal places using the round() function.

3. By cleaning the data, we now get a more accurate representation of the data, which will allow us to prepare for the next step in the data analysis lifecycle. I tested for duplicates but did not find any, so no adjustments were made. the missing data was imputed into each variable that was present based on the shape of the histogram. The outliers were adjusted to give a better representation of each variable by either imputing the outliers as the median or by retaining the original outliers. Other categorical variables were adjusted to make the data more accessible for data exploration, such as changing booleans to numerical data and adjusting decimal points.

In [66]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 54 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   Unnamed: 0           10000 non-null  int64
 1   CaseOrder            10000 non-null  int64
 2   Customer_id          10000 non-null  object
 3   Interaction          10000 non-null  object
 4   UID                  10000 non-null  object
 5   City                 10000 non-null  object
 6   State                10000 non-null  object
 7   County               10000 non-null  object
 8   Zip                  10000 non-null  int64
 9   Lat                  10000 non-null  float64
 10  Lng                  10000 non-null  float64
 11  Population           10000 non-null  float64
 12  Area                 10000 non-null  object
 13  Timezone             10000 non-null  object
 14  Job                  10000 non-null  object
 15  Children             10000 non-null  float64
 16  Age                  10000 non-null  float64
 17  Education            10000 non-null  object
 18  Employment           10000 non-null  object
 19  Income               10000 non-null  float64
 20  Marital              10000 non-null  object
 21  Gender               10000 non-null  object
 22  ReAdmis              10000 non-null  object
 23  VitD_levels          10000 non-null  float64
 24  Doc_visits           10000 non-null  int64
 25  Full_meals_eaten     10000 non-null  int64
 26  VitD_supp            10000 non-null  int64
 27  Soft_drink           10000 non-null  object
 28  Initial_admin        10000 non-null  object
 29  HighBlood            10000 non-null  object
 30  Stroke               10000 non-null  object
 31  Complication_risk    10000 non-null  object
 32  Overweight           10000 non-null  float64
 33  Arthritis            10000 non-null  object
 34  Diabetes             10000 non-null  object
 35  Hyperlipidemia       10000 non-null  object
 36  BackPain             10000 non-null  object
 37  Anxiety              10000 non-null  float64
 38  Allergic_rhinitis    10000 non-null  object
 39  Reflux_esophagitis   10000 non-null  object
 40  Asthma               10000 non-null  object
 41  Services             10000 non-null  object
 42  Initial_days         10000 non-null  float64
 43  TotalCharge          10000 non-null  float64
 44  Additional_charges   10000 non-null  float64
 45  Item1                10000 non-null  int64
 46  Item2                10000 non-null  int64
 47  Item3                10000 non-null  int64
 48  Item4                10000 non-null  int64
 49  Item5                10000 non-null  int64
 50  Item6                10000 non-null  int64
 51  Item7                10000 non-null  int64
 52  Item8                10000 non-null  int64
 53  numeric              10000 non-null  int64
dtypes: float64(12), int64(15), object(27)
memory usage: 4.1+ MB
```

4. One of the disadvantages of these methods that I used for duplicates is that even though I looked at all the keys for removing duplicates, I did not take a deeper dive into the columns to see if there may be any underlying duplicates that may not be apparent at first. An underlying issue with my method of imputing the missing data is the addition of bias by imputing the mean, median, or mode. Since we are imputing with one of these three values, it could add some bias toward the data, especially for the larger number of needed rows. My method of imputing or retaining outliers has some limitations as well. One of which is similar to the restriction in imputing the missing data step where the removal of outliers and replacing them with the mean, median or mode would underestimate the variability of the data or would introduce bias towards whatever value we impute the outliers as. When decreasing the number of decimal points on the variables, we may not get the exact value, and in a way, it is an approximation. This would mean that some data is lost by removing some decimals, and the data in such variables is not exact.

5. The limitations summarized in the last part could affect the question, “What factors affect diabetes?”. If there were duplicates that were not correctly removed, then we would create an inaccurate representation of the data by inflating counts. Similarly, the method we used to impute data for the missing data may not reflect the true values of the dataset. To the question, if some variables now affect diabetes more than before, this would mean that the underlying relationship between diabetes and the variable changed. By using the current data with the outliers being imputed, would give us an inaccurate measure of dispersion as well as inaccurate results of data. To the question, if the amount of meals a

person may eat is taken as a variable, then if by removing outliers which may be 4 or more meals, we are taking away what could potentially be a reason for diabetes.

D.

1. (CSV file is in a separate link)
2. (PDF for code is attached as a separate link)

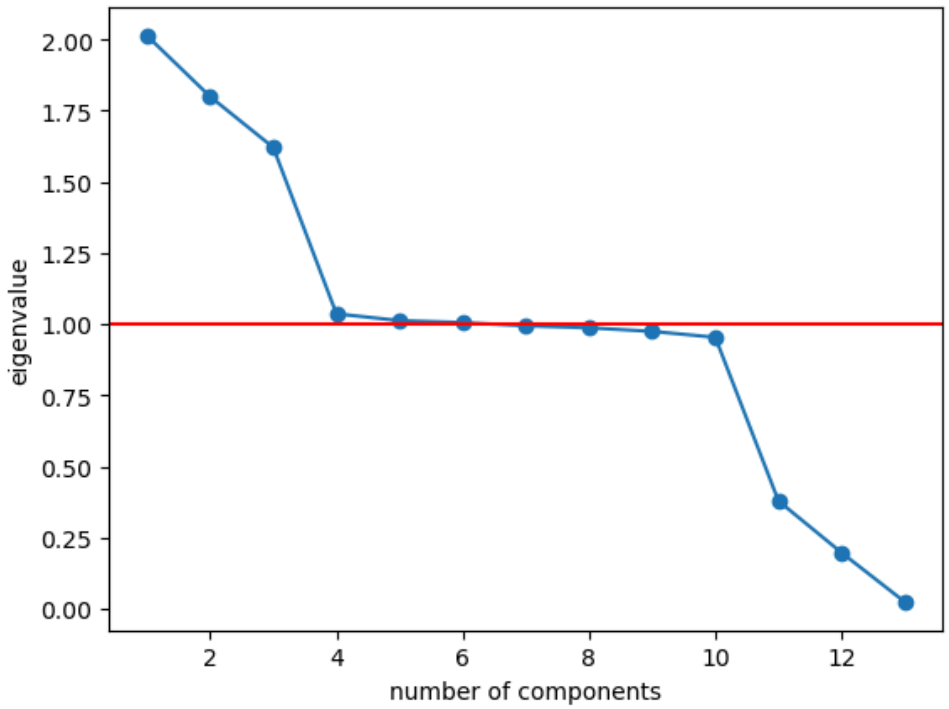
E.

1. The variables used for PCA are Lat, Lng, Population, Children, Age, Income, VitD_levels, Doc_visits, Full_meals_eaten, VitD_supp, Initial_days, TotalCharge, Additional_charges. The reason these variables were used is because they are continuous variables.

[94]:

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10	PC11	PC12	PC13
Lat	-0.688881	0.090087	0.005902	-0.020168	0.011733	0.028989	-0.052836	0.023869	-0.011976	0.115469	0.004047	0.002006	0.706434
Lng	-0.689853	0.091700	0.007406	-0.017607	0.008570	0.029425	-0.047062	0.024887	-0.013409	0.102658	0.000399	0.001961	-0.707696
Population	0.181661	-0.003183	-0.025145	-0.169625	0.043939	0.193604	-0.371285	0.133892	-0.075381	0.858259	0.002701	0.000311	-0.009627
Children	0.007908	0.005243	0.011403	0.123882	0.032642	0.890027	-0.267451	0.043538	0.190943	-0.284955	-0.008735	0.007528	0.001788
Age	0.018017	0.076730	0.701364	-0.009192	0.038900	-0.008535	-0.005971	-0.008175	-0.025770	0.013793	-0.704842	0.049416	0.002402
Income	0.002998	-0.001692	0.001013	-0.486053	-0.347674	-0.117513	-0.236218	0.709034	0.038344	-0.261704	-0.023492	-0.003240	0.001797
VitD_levels	0.005916	0.048511	0.015771	0.299371	-0.695813	0.230925	0.365242	0.100968	-0.446533	0.145230	-0.014477	-0.077205	0.002978
Doc_visits	-0.011559	-0.009666	0.014245	-0.279159	-0.541270	-0.084112	-0.430212	-0.649551	0.087597	-0.082732	-0.006433	-0.005707	-0.000855
Full_meals_eaten	-0.004441	-0.024515	0.037558	0.556597	-0.286246	-0.213568	-0.122433	0.183892	0.698452	0.154495	-0.009132	0.005123	-0.000854
VitD_supp	-0.001700	0.038639	0.012404	-0.490496	-0.106798	0.217701	0.627518	-0.100154	0.508193	0.192022	-0.004680	-0.001072	-0.000276
Initial_days	0.085627	0.693399	-0.088100	0.002814	0.055428	-0.033248	-0.047994	-0.008802	0.044410	-0.027961	-0.057379	-0.701064	0.000686
TotalCharge	0.090274	0.697039	-0.071925	0.024273	-0.027377	-0.016387	-0.007879	-0.000141	-0.009504	-0.013374	0.054516	0.704204	0.000692
Additional_charges	0.021820	0.079633	0.701630	-0.017189	0.010465	-0.001133	-0.012558	0.012501	-0.013370	0.001794	0.704220	-0.063786	-0.001093

2. To decide which PCs should be retained, I used the Kaiser rule. This method entails that only the PCs with a variance greater than 1 will be retained. The reason why this method uses PCs with a variance greater than 1 is because any PC with a lower variance will contain less information than the others, so it is not worth retaining. Judging by the graph below, there are seven components that will be used, as several of them contain eigenvalues above 1.



3. The main way that the organization can benefit from PCAs is that they decrease the number of variables used, which makes them easier to interpret and visualize. This creates simpler models and can lead to faster and more efficient data analysis. Another way the organization can benefit from PCAs is noise reduction since most of the data is spread along the x-axis, while the data on the y-axis is mostly dominated by noise. Applying PCA projects the data to a 1 dimensional space which can help eliminate that noise and create a cleaner signal.

F. (Panopto recording)

G.

1. *Rudolph, A., Krois, J., Hartmann, K. (2023): Statistics and Geodata Analysis using Python (SOGA-Py). Department of Earth Sciences, Freie Universitaet Berlin,*

https://www.geo.fu-berlin.de/en/v/soga-py/Advanced-statistics/Multivariate-Approaches/Principal-Component-Analysis/PCA-the-basics/Choose-Principal-Components/index.html#:~:text=The%20Kaiser's%20rule%20

2. "What Is Principal Component Analysis (PCA) & How to Use It?" *Bigabid*, 8 Feb. 2023, www.bigabid.com/what-is-pca-and-how-can-i-use-it/.

3. Middleton, Keiona. *Https://Westerngovernorsuniversity.Sharepoint.Com/Sites/DataScienceTeam/Shared%20Documents/Forms/AllItems.Aspx?id=%2Fsites%2FDataScienceTeam%2FShared%20Documents%2FGraduate%20Team%2FD206%2FStudent%20Facing%20Resources%2FD206%20Course%20Guide%2DV3%2Epdf&parent=%2Fsites%2FDataScienceTeam%2FShared%20Documents%2FGraduate%20Team%2FD206%2FStudent%20Facing%20Resources*, WGU, westerngovernorsuniversity.sharepoint.com/sites/DataScienceTeam/Shared%20Documents/Forms/AllItems.aspx?id=%2Fsites%2FDataScienceTeam%2FShared%20Documents%2FGraduate%20Team%2FD206%2FStudent%20Facing%20Resources%2FD206%20Course%20Guide%2DV3%2Epdf&parent=%2Fsites%2FDataScienceTeam%2FShared%20Documents%2FGraduate%20Team%2FD206%2FStudent%20Facing%20Resources. Accessed 29 July 2024.

H. No sources cited