Performace Assessment: D207 Exploratory Data Analysis

Michael Hindes

Department of Information Technology, Western Governors University
D207: Exploratory Data Analysis
Professor David Gagner
December 29, 2023

Task A: Describe a real-world organizational situation or issue in the Data Dictionary you chose, by doing the following:

A1. Provide one question that is relevant to your chosen data set.

Patient readmissions represent a significant concern in the medical sector, impacting patient welfare and hospital operations alike. The seriousness of the issue is underscored by the fact that the Centers for Medicare and Medicaid Services have introduced financial penalties for hospitals experiencing elevated readmission rates (WGU, 2023). Despite the potential economic and image costs, numerous hospitals find themselves ill-equipped to effectively address readmissions. This performance assessment aims to explore this critical gap by looking for factors that might influence a patient's probability of readmission. This kind of data analytics process crucial for fostering better health service outcomes.

Question: Do the medical services a patient receives during their initial hospitalization influence their chances of readmission to hospital?

To tackle this question, the `medical_clean.csv` dataset (WGU, 2023) has been cleaned for analysis and will be used. This dataset contains a variety of variables that could influence a patient's chances of returning to the hospital. After inspecting and understanding the data in this particular context, analyses and statistical tests are performed to try and see if a potential relationship exists between received medical services (`Services`) and patient readmissions (`ReAdmis`). The results of these analyses are then used to try and answer the question posed above.

A2. Explain how stakeholders in the organization could benefit from an analysis of the data.

The stakeholders likely include hospital administrators, healthcare staff, patients, and possibly investors. Administrators and staff can use this analysis to pinpoint patients at higher risk of readmission, allowing them to implement preventive strategies. This approach can not only reduce costs linked to readmissions but also elevates patient care quality. Patients stand to benefit from improved healthcare practices and reduced likelihood of readmissions, enhancing their overall health outcomes. In essence, this analysis could facilitate more efficient hospital operations and foster better patient care.

A3. Identify all of the data in your data set that are relevant to answering your question in part *A1*.

Variable Categories: This dataset contains information on 10,000 patients who were admitted to a hospital. For a general understand of the type of information contained within it, we can break the variables down into the following general categories:

- *Patient Characteristics*: Age, gender, and various health conditions.

- *Hospital Stay Details*: Treatments administered, duration of hospitalization, and the nature of the patient's initial entry. Importantly, the dataset includes a 'ReAdmis' variable, which is a binary indication (Yes, No) of whether the patient was readmitted within a month of discharge.
- *Health Issues*: Hypertension, cerebral incidents, excessive weight, joint inflammation, and sugar imbalances.
- *Demographics*: Sex, age bracket, occupation, and educational background.

Variables: For the first part of the assessment (Section A-B), relationships between categorical variables were explored. For the second part of this assessment (Sections C-D), there were additional continuous variables explored. The continuous variables will be described in sections C and D. Here, I will described the categorical variables that were used and explored as part of exploratory data analysis for sections A and B of this assessment. The main focus will be between `ReAdmis` and `Services`. To narrow the focus, Microsoft Excel was used to reduce `medical_clean.csv` to a clean set of variables relevant to this analysis. This updated dataset is named `medical_clean_forchi.csv` and includes the following variables:

- `Gender`, `HighBlood`, `Stroke`, `Complication_risk`, `Overweight`, `Arthritis`, `Diabetes`, `Hyperlipidemia`, `BackPain`, `Anxiety`, `Allergic_rhinitis`, `Reflux_esophagitis`, `Asthma`, `Services`, `Initial_admin`, `Marital`, `VitD_supp`, `Soft_drink`, `ReAdmis`.

The table below describes all the categorical variables in the original dataset as they were the one relevant to answering the question from A1 (WGU, 2023). The table includes the Variable name, Data type, Variable Type, Description, and an Example for every variable in the dataset.

In [2]:
```
from IPython.display import Image
Image(filename='variable_descriptions_d207.jpg')
```

Out[2]:

Variable name	Data type	Variable Type	Description	Example
Area	object	Categorical - Nominal	Classification of area (suburban, urban, rural) according to unofficial census data.	Suburban
Marital	object	Categorical - Nominal	Patient's marital status (or the primary insurance holder).	Divorced
Gender	object	Categorical - Nominal	Patient's self-identified gender as male, female, or nonbinary.	Male
ReAdmis	object	Categorical - Binary	Indication of whether the patient was readmitted within a month of discharge (Yes, No).	No
Soft_drink	object	Categorical - Binary	Indication of whether the patient regularly consumes three or more sodas per day (Yes, No).	Yes
Initial_admin	object	Categorical - Nominal	The method of initial hospital admission for the patient (emergency admission, elective admission, observation).	Emergency Admission
HighBlood	object	Categorical - Binary	Indication of whether the patient has hypertension (Yes, No).	Yes
Stroke	object	Categorical - Binary	Indication of whether patient has experienced a stroke in past (Yes, No).	No
Complication_risk	object	Categorical - Ordinal	Patient's risk level for complications as determined by a primary patient assessment (high, medium, low).	Medium
Overweight	object	Categorical - Binary	Specifies if patient is deemed overweight based on age, gender, and height (Yes, No).	No
Arthritis	object	Categorical - Binary	Specifies if patient has arthritis (Yes, No).	Yes
Diabetes	object	Categorical - Binary	Specifies if patient has diabetes (Yes, No).	Yes
Hyperlipidemia	object	Categorical - Binary	Specifies if patient has hyperlipidemia (Yes, No).	No
BackPain	object	Categorical - Binary	Specifies if patient suffers from chronic back pain (Yes, No).	Yes
Anxiety	object	Categorical - Binary	Specifies if patient has an anxiety disorder (Yes, No).	No
Allergic_rhinitis	object	Categorical - Binary	Specifies if patient has allergic rhinitis (Yes, No).	Yes
Reflux_esophagitis	object	Categorical - Binary	Specifies if patient has reflux esophagitis (Yes, No).	No
Asthma	object	Categorical - Binary	Specifies if patient has asthma (Yes, No).	Yes
Services	object	Categorical - Nominal	Main service provided to the patient during hospitalization (blood work, intravenous, CT scan, MRI).	Blood Work

---

Task B: Describe the data analysis by doing the following:

B1. Using one of the following techniques, write code in Python to run the analysis of the data set:

- ☑ **chi-square**
- [ ] t-test
- [ ] ANOVA

Python is the chose programming language for this assessment. The code for this assessment was written in Jupyter Notebook using Python 3.12. Some additional code is added for the purposes of formatting and readability. The chi-square test statistic is chosen.

In [3]:
```
# Install and Import dependencies
%pip install scikit-learn
%matplotlib inline
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import scipy.stats as stats
import seaborn as sns
from pandas import DataFrame
from scipy.stats import chi2_contingency
```

```
Requirement already satisfied: scikit-learn in c:\users\hinde\documents\github\master\.venv\lib\site-packages
(1.3.2)
Requirement already satisfied: numpy<2.0,>=1.17.3 in c:\users\hinde\documents\github\master\.venv\lib\site-pa
ckages (from scikit-learn) (1.26.2)
Requirement already satisfied: scipy>=1.5.0 in c:\users\hinde\documents\github\master\.venv\lib\site-packages
(from scikit-learn) (1.11.4)
Requirement already satisfied: joblib>=1.1.1 in c:\users\hinde\documents\github\master\.venv\lib\site-package
s (from scikit-learn) (1.3.2)
Requirement already satisfied: threadpoolctl>=2.0.0 in c:\users\hinde\documents\github\master\.venv\lib\site-
packages (from scikit-learn) (3.2.0)
Note: you may need to restart the kernel to use updated packages.
```

In [4]:
```
# Import the categorical data and read it into a dataframe = df_cat
df_cat = pd.read_csv('medical_clean_forchi.csv')

# Display the first/last five rows of the data
df_cat.head()
```

Out[4]:

	Area	Marital	Gender	ReAdmis	Soft_drink	Initial_admin	HighBlood	Stroke	Complication_risk	Overweight	Arthritis
0	Suburban	Divorced	Male	No	No	Emergency Admission	Yes	No	Medium	No	Yes
1	Urban	Married	Female	No	No	Emergency Admission	Yes	No	High	Yes	No
2	Suburban	Widowed	Female	No	No	Elective Admission	Yes	No	Medium	Yes	No
3	Suburban	Married	Male	No	No	Elective Admission	No	Yes	Medium	No	Yes
4	Rural	Widowed	Female	No	Yes	Elective Admission	No	No	Low	No	No

In [5]:
```
# View the last 5 rows of the dataframe and make sure the data has made it
# all the way to the last expected row
df_cat.tail()
```

Out[5]:

	Area	Marital	Gender	ReAdmis	Soft_drink	Initial_admin	HighBlood	Stroke	Complication_risk	Overweight	Arthritis
9995	Urban	Widowed	Male	No	No	Emergency Admission	Yes	No	Medium	No	No
9996	Urban	Widowed	Male	Yes	No	Elective Admission	Yes	No	Medium	Yes	Yes
9997	Rural	Separated	Female	Yes	Yes	Elective Admission	Yes	No	High	Yes	No
9998	Rural	Divorced	Male	Yes	No	Emergency Admission	No	No	Medium	Yes	No
9999	Urban	Separated	Female	Yes	No	Observation Admission	No	No	Low	Yes	Yes

In [6]:
```
# Understanding the Data
# Check the shape of the DataFrame.
print('Total rows:', df_cat.shape[0])
print('Total columns:', df_cat.shape[1])
```

```
Total rows: 10000
Total columns: 19
```

In [7]:
```
# Check the DataFrame information
df_cat.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 19 columns):
 #   Column              Non-Null Count  Dtype 
---  ------              --------------  ----- 
 0   Area                10000 non-null  object
 1   Marital             10000 non-null  object
 2   Gender              10000 non-null  object
 3   ReAdmis             10000 non-null  object
 4   Soft_drink          10000 non-null  object
 5   Initial_admin       10000 non-null  object
 6   HighBlood           10000 non-null  object
 7   Stroke              10000 non-null  object
 8   Complication_risk   10000 non-null  object
 9   Overweight          10000 non-null  object
 10  Arthritis           10000 non-null  object
 11  Diabetes            10000 non-null  object
 12  Hyperlipidemia      10000 non-null  object
 13  BackPain            10000 non-null  object
 14  Anxiety             10000 non-null  object
 15  Allergic_rhinitis   10000 non-null  object
 16  Reflux_esophagitis  10000 non-null  object
 17  Asthma              10000 non-null  object
 18  Services            10000 non-null  object
dtypes: object(19)
memory usage: 1.4+ MB
```

In [8]:
```
# Check the Descriptive Statistics of the DataFrame,
# including unique values, counts, top values and frequency
print("Columns 1-9 of 19:")
df_cat.iloc[:, :9].describe()
```

```
Columns 1-9 of 19:
```

Out[8]:

	Area	Marital	Gender	ReAdmis	Soft_drink	Initial_admin	HighBlood	Stroke	Complication_risk
count	10000	10000	10000	10000	10000	10000	10000	10000	10000
unique	3	5	3	2	2	3	2	2	3
top	Rural	Widowed	Female	No	No	Emergency Admission	No	No	Medium
freq	3369	2045	5018	6331	7425	5060	5910	8007	4517

In [9]:
```
# Check the Descriptive Statistics of the DataFrame
print("Columns 10-18 of 19:")
df_cat.iloc[:, 9:19].describe()
```

```
Columns 10-18 of 19:
```

Out[9]:

	Overweight	Arthritis	Diabetes	Hyperlipidemia	BackPain	Anxiety	Allergic_rhinitis	Reflux_esophagitis	Asthma	Service
count	10000	10000	10000	10000	10000	10000	10000	10000	10000	10000
unique	2	2	2	2	2	2	2	2	2	4
top	Yes	No	No	No	No	No	No	No	No	Blood Wor
freq	7094	6426	7262	6628	5886	6785	6059	5865	7107	5265

- There are No missing values, duplicate rows, or duplicate columns in the dataset. The number of unique values aligns with the expected data types and values for each column. The data is clean and ready for analysis.
- Now the data structure is understood, a contingency table is created to show the frequencies of the "Services" across the "ReAdmis" categories. Additionally, the expected and observed frequencies will be compared. Then, the chi-square test will be performed to see if there's a statistically significant association between these two variables.

B2. Provide the output and the results of any calculations from the analysis you performed.

- **Null hypothesis H_0:** There is not a significant association between `Services` and `ReAdmis`
- **Alternative hypothesis H_1:** There is a significant association between `Services` and `ReAdmis`

The `alpha`, or the threshold for statistical significance, at `0.05`. This level is widely accepted as the conventional standard in research. (Sewell)

A Chi-square test of independence is conducted since both Services and ReAdmis are categorical variables.

In [10]:
```
# Create a contingency table of the two variables 'ReAdmis' and 'Services'
print("Contingency table for `ReAdmis` and `Services` with row and column totals")
contingency_table = pd.crosstab(df_cat['ReAdmis'], df_cat['Services'])
contingency_table
```

```
Contingency table for `ReAdmis` and `Services` with row and column totals
```

Out[10]:

Services	Blood Work	CT Scan	Intravenous	MRI
ReAdmis				
No	3335	737	2027	232
Yes	1930	488	1103	148

In [11]:
```
# Calculate the chi-square value, p-value, degrees of freedom, and expected frequencies
chi2, p_value, dof, expected = chi2_contingency(contingency_table)
```

```
# Create a DataFrame for the expected frequencies
expected_df = pd.DataFrame(expected, index=contingency_table.index)

# Print the observed and expected frequencies
expected_df.columns = contingency_table.columns
combined_dfinal = pd.concat([contingency_table, expected_df], axis=1)
combined_dfinal.columns = pd.MultiIndex.from_product([['Observed Frequencies', 'Expected Frequencies'], conti

print("The Table below helps to assess whether there's a notable difference between the number of actual outc

combined_dfinal
```

```
The Table below helps to assess whether there's a notable difference between the number of actual outcomes(ob
served frequencies) and the number of times one would expect that outcome to occur purely by chance (expected
frequencies), assuming there is no underlying association between the variables in question. There are slight
differences between some of the observed and expected frequencies.
```

Out[11]:

	Observed Frequencies				Expected Frequencies			
Services	**Blood Work**	**CT Scan**	**Intravenous**	**MRI**	**Blood Work**	**CT Scan**	**Intravenous**	**MRI**
ReAdmis								
No	3335	737	2027	232	3333.2715	775.5475	1981.603	240.578
Yes	1930	488	1103	148	1931.7285	449.4525	1148.397	139.422

In [12]:
```
# Print the Chi-Squared test statistic, degrees of freedom, and p-value

print('Chi-Squared test statistic, degrees of freedom, and p-value:', end='\n\n')
print('Chi-squared value:  ' + str(chi2), end='\n\n')
print('Degrees of freedom:  ' + str(dof), end='\n\n')
print('P-value:  ' + str(p_value), end='\n\n')
```

```
Chi-Squared test statistic, degrees of freedom, and p-value:

Chi-squared value:  8.892645054628435

Degrees of freedom:  3

P-value:  0.03075281113212747
```

B3. Justify why you chose this analysis technique.

Using the Chi-square test for this analysis serves both hypothesis testing and data exploration purposes. The Chi-square test is a nonparametric, distribution-free method, ideal for examining relationships between categorical variables, such as the `Services` provided to patients and their subsequent readmission (`ReAdmis`) status. This test identifies whether a statistically significant relationship exists without making assumptions about population parameters, which makes it well-suited for analyzing the categorical data in our medical dataset. Additionally, the Chi-square test is advantageous over a Z-test as it is applicable to both small and large sample sizes. (Sewell)

It also doesn't require data to follow a normal distribution, a requirements for other tests like the t-test or ANOVA that focus more on comparing mean values. Interest here lies in determining if specific services may result in hospital readmissions, not in comparing averages. The Chi-square test, which evaluates observed frequencies against expected frequencies under an assumption that there is no association, aligns with the research needs. Because it operates on frequency of values, it avoids the need for data to fit a normal distribution. This is particularly beneficial for categorical data in medical research, which can include patient survey data and other categorical information. Using the Chi-square test gives one a direct and effective method for discovering potential dependencies in data, especially regarding patient readmission. (Sewell)

B(cont) Additional exploratory code:

The code below is used to generate a contingency table for each variable against 'ReAdmis' and display it properly. It was part of the early exploratory and discovery process as a tried to better understand the categorical variables. IT was good Python programming practice to create a loop that can iterate through each variable and generate a contingency table for each variable against 'ReAdmis' and then visualize it properly.

In [13]:
```
# The code below is used to generate a contingency table for each variable against 'ReAdmis' and display it p

import pandas as pd
from IPython.display import display, HTML

# Generate contingency table for each variable against 'ReAdmis'
for column in df_cat.columns:
    if column != 'ReAdmis':
        contingency_table = pd.crosstab(df_cat['ReAdmis'], df_cat[column], margins=True, margins_name='Total'
        contingency_table_styled = contingency_table.style.set_table_styles([
            {'selector': 'th', 'props': [('border', '1px solid black')]},
            {'selector': 'td', 'props': [('border', '1px solid black')]},
            {'selector': 'tr:nth-child(even)', 'props': [('background-color', '#e9e9e9')]},
            {'selector': 'tr:nth-child(odd)', 'props': [('background-color', '#F4F4F4')]},
        ])
        display(contingency_table_styled)
        print('\n')
```

Area	Rural	Suburban	Urban	Total
ReAdmis				
No	2150	2106	2075	6331
Yes	1219	1222	1228	3669
Total	3369	3328	3303	10000

Marital	Divorced	Married	Never Married	Separated	Widowed	Total
ReAdmis						
No	1283	1268	1243	1259	1278	6331
Yes	678	755	741	728	767	3669
Total	1961	2023	1984	1987	2045	10000

Gender	Female	Male	Nonbinary	Total
ReAdmis				
No	3205	2995	131	6331
Yes	1813	1773	83	3669
Total	5018	4768	214	10000

Soft_drink	No	Yes	Total
ReAdmis			
No	4717	1614	6331
Yes	2708	961	3669
Total	7425	2575	10000

Initial_admin	Elective Admission	Emergency Admission	Observation Admission	Total
ReAdmis				
No	1608	3156	1567	6331
Yes	896	1904	869	3669
Total	2504	5060	2436	10000

HighBlood	No	Yes	Total
ReAdmis			
No	3747	2584	6331
Yes	2163	1506	3669
Total	5910	4090	10000

Stroke	No	Yes	Total
ReAdmis			
No	5071	1260	6331
Yes	2936	733	3669
Total	8007	1993	10000

Complication_risk	High	Low	Medium	Total
ReAdmis				
No	2135	1343	2853	6331
Yes	1223	782	1664	3669
Total	3358	2125	4517	10000

Overweight	No	Yes	Total
ReAdmis			
No	1821	4510	6331
Yes	1085	2584	3669
Total	2906	7094	10000

Arthritis	No	Yes	Total
ReAdmis			
No	4086	2245	6331
Yes	2340	1329	3669
Total	6426	3574	10000

Diabetes	No	Yes	Total
ReAdmis			
No	4591	1740	6331
Yes	2671	998	3669
Total	7262	2738	10000

Hyperlipidemia	No	Yes	Total
ReAdmis			
No	4206	2125	6331
Yes	2422	1247	3669
Total	6628	3372	10000

BackPain	No	Yes	Total
ReAdmis			
No	3758	2573	6331
Yes	2128	1541	3669
Total	5886	4114	10000

Anxiety	No	Yes	Total
ReAdmis			
No	4301	2030	6331
Yes	2484	1185	3669
Total	6785	3215	10000

Allergic_rhinitis	No	Yes	Total
ReAdmis			
No	3825	2506	6331
Yes	2234	1435	3669
Total	6059	3941	10000

Reflux_esophagitis	No	Yes	Total
ReAdmis			
No	3726	2605	6331
Yes	2139	1530	3669
Total	5865	4135	10000

Asthma	No	Yes	Total
ReAdmis			
No	4462	1869	6331
Yes	2645	1024	3669
Total	7107	2893	10000

Services	Blood Work	CT Scan	Intravenous	MRI	Total
ReAdmis					
No	3335	737	2027	232	6331
Yes	1930	488	1103	148	3669
Total	5265	1225	3130	380	10000

C1: Identify the distribution of two continuous variables and two categorical variables using univariate statistics from your cleaned and prepared data.

Represent your findings in Part C, visually as part of your submission.

For the `univariate` statistics, we will look at the distribution of the following variables contained in the `medical_clean.csv` dataset:

- `VitD_levels` : float64, continuous. A measurement of the patient's vitamin D levels in ng/mL. e.g. 17.80233049
- `Initial_days` : float64, continuous. Duration of the patient's initial hospital stay in days. e.g. 10.58576971
- `Services` : object, nominal. Main service provided to the patient during hospitalization (blood work, intravenous, CT scan, MRI)
- `Overweight` : object, binary. A measurement of the patient's vitamin D levels in ng/mL. (Yes, No)

The data will be read in and quickly checked using pd.read_csv() and df.head() to ensure the data is being read in correctly. Then, the data will be plotted using the `Seaborn` and `Pyplot` packages to create a histogram for the continuous variables and a bar chart for the categorical variables.

The plots are labeled and described, and a density plot has also been added to the histograms of the continuous variables for added visual information.

The code and output for this analysis is shown below.

- C1. Univariate Graphics (4 total): Categorical (2 total), Continuous (2 total)

In [14]:
```
# Read in the cleaned data, assign to the variable `df_med` and display the first five rows
df_med = pd.read_csv('medical_clean.csv')
df_med.head()
```

Out[14]:

	CaseOrder	Customer_id	Interaction	UID	City	State	County	Zip	Lat	
0	1	C412403	8cd49b13-f45a-4b47-a2bd-173ffa932c2f	3a83ddb66e2ae73798bdf1d705dc0932	Eva	AL	Morgan	35621	34.34960	-86.
1	2	Z919181	d2450b70-0337-4406-bdbb-bc1037f1734c	176354c5eef714957d486009feabf195	Marianna	FL	Jackson	32446	30.84513	-85.
2	3	F995323	a2057123-abf5-4a2c-abad-8ffe33512562	e19a0fa00aeda885b8a436757e889bc9	Sioux Falls	SD	Minnehaha	57110	43.54321	-96.
3	4	A879973	1dec528d-eb34-4079-adce-0d7a40e82205	cd17d7b6d152cb6f23957346d11c3f07	New Richland	MN	Waseca	56072	43.89744	-93.
4	5	C544523	5885f56b-d6da-43a3-8760-83583af94266	d2f0425877b10ed6bb381f3e2579424a	West Point	VA	King William	23181	37.59894	-76.

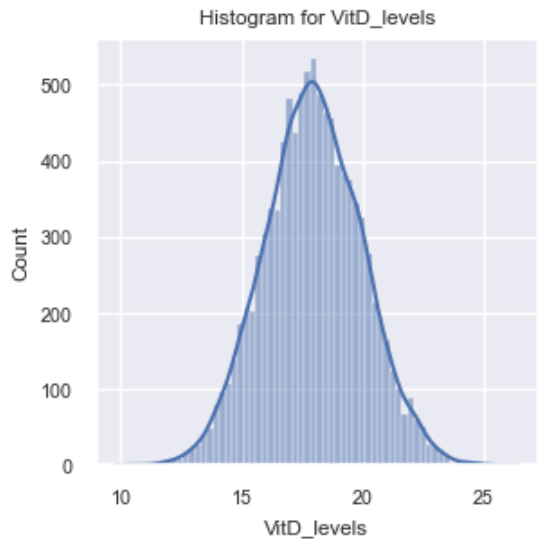
5 rows × 50 columns

In [15]:

```
# Set the font size for all plots
sns.set(font_scale=0.7)

# Histogram for 'VitD_levels'
print("`VitD_levels`: float64, continuous. A measurement of the patient's vitamin D levels in ng/mL. e.g. 17.
print('Normal distribution')
plt.figure(layout='constrained', figsize=(3, 3))
sns.histplot(data=df_med, x='VitD_levels', kde=True)
plt.title('Histogram for VitD_levels')
plt.show()
print('\n')

# Histogram for 'Initial_days'
print("`Initial_days`: float64, continuous. Duration of the patient's initial hospital stay in days. e.g. 10.
print('Bimodal distribution')
plt.figure(layout='constrained', figsize=(3, 3))
sns.histplot(data=df_med, x='Initial_days', kde=True)
plt.title('Histogram for Initial_days')
plt.show()
print('\n')

# Bar chart for 'Services'
print("`Services`: object, nominal. Main service provided to the patient during hospitalization (blood work,
print("'Blood Work' is the most common service, making up over half of all services provided (This is an impo
plt.figure(layout='constrained', figsize=(3, 3))
sns.countplot(data=df_med, x='Services')
plt.title('Bar chart for Services')
plt.show()
print('\n')

# Bar chart for 'Overweight'
print("`Overweight`: object, binary. A measurement of the patient's vitamin D levels in ng/mL. (Yes, No). Mos
print('Binary Categorical')
plt.figure(layout='constrained', figsize=(3, 3))
sns.countplot(data=df_med, x='Overweight')
plt.title('Bar chart for Overweight')
plt.show()
print('\n')
```

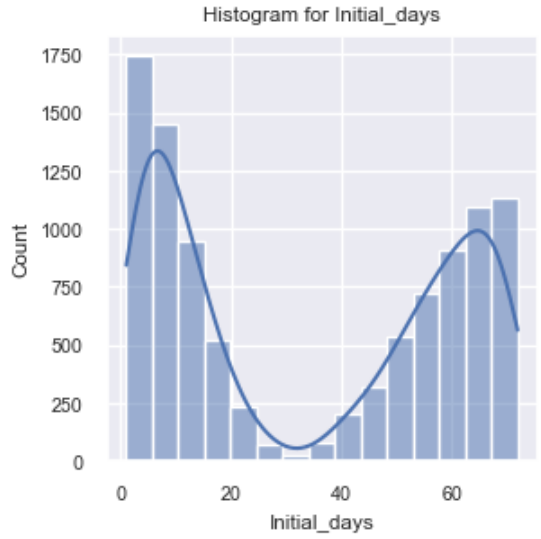
`VitD_levels`: float64, continuous. A measurement of the patient's vitamin D levels in ng/mL. e.g. 17.80233049

Normal distribution



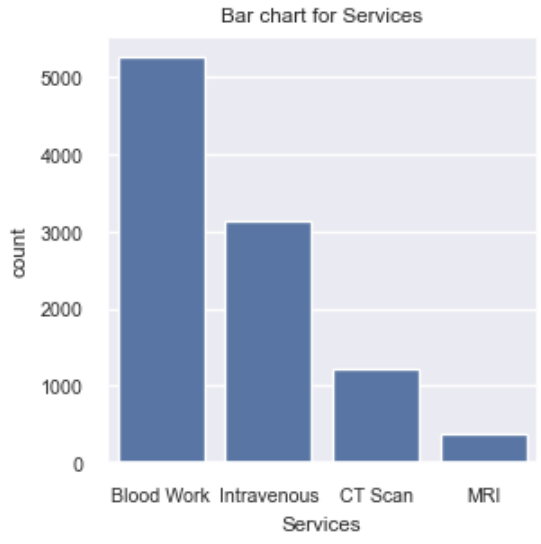
`Initial_days`: float64, continuous. Duration of the patient's initial hospital stay in days. e.g. 10.585769711

Bimodal distribution

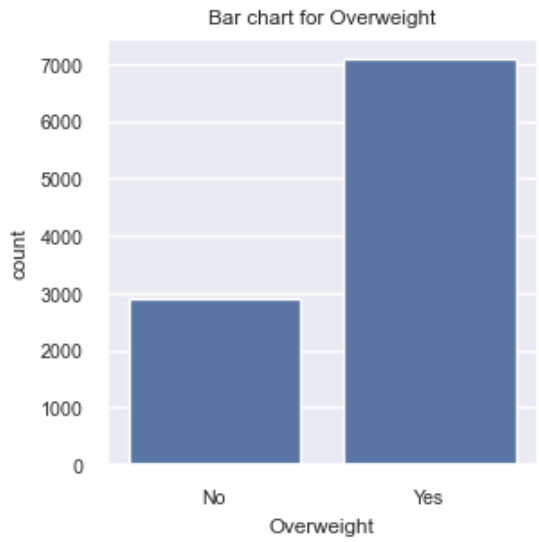


`Services`: object, nominal. Main service provided to the patient during hospitalization (blood work, intravenous, CT scan, MRI)

'Blood Work' is the most common service, making up over half of all services provided (This is an important data point for our Chi-Square analysis and is addressed in the discussion section). 'Intravenous' services are the next most common, followed by 'CT Scan' and 'MRI', which are the least common.



`Overweight`: object, binary. A measurement of the patient's vitamin D levels in ng/mL. (Yes, No). Most patients were considered overweight.
Binary Categorical



D1: Identify the distribution of two continuous variables and two categorical variables using bivariate statistics from your cleaned and prepared data.

Represent your findings in Part D, visually as part of your submission.

For the `bivariate` statistics, we will look at the distribution of the following variables contained in the `medical_clean.csv` dataset, which has already been assigned to the variable `df_med`:

- `Initial_days`: float64, continuous. Yearly income of the patient (or the primary insurance holder). e.g. 86575.93
- `TotalCharge`: object, continuous. Daily charge to the patient. excluding specialized treament. e.g. 3191.05
- `ReAdmis`: object, binary. Indication of whether the patient was readmitted within a month of discharge (Yes, No).
- `Gender`: object, nominal. Patient's self-identified gender as male, female, or non-binary.

The plots are labeled and described.

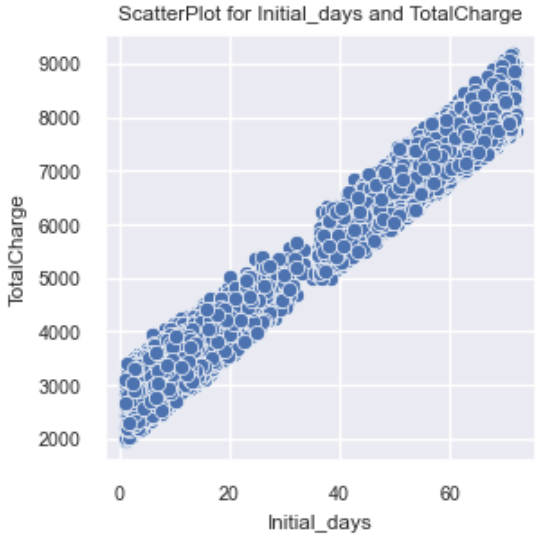
The code and output for this analysis is shown below.

- D1. Bivariate Graphics (3 total): Continuous v Continuous (1 total), Categorical v Continuous (1 total), Categorical v Categorical (1 total)

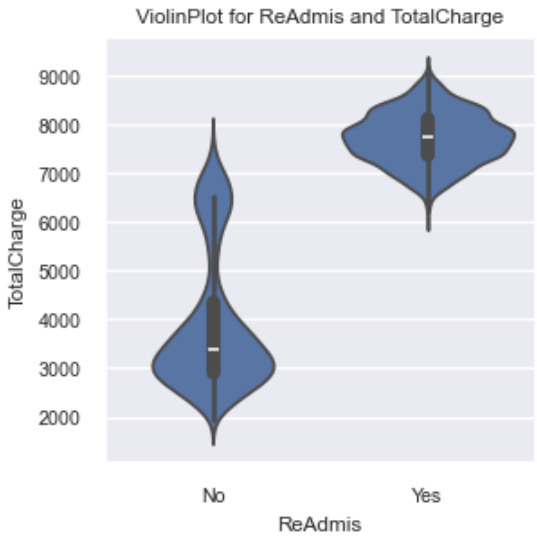
In [16]:
```
# ScatterPlot for 'Initial_days' and 'TotalCharge'
print("Continuous vs. Continuous")
print("There is a positive correlation between 'Initial_days' and 'TotalCharge', which makes sense; They are
plt.figure(layout='constrained', figsize=(3, 3))
sns.scatterplot(data=df_med, x="Initial_days", y="TotalCharge")
plt.title('ScatterPlot for Initial_days and TotalCharge')
# Show the plot
plt.show()
print('\n')

# Violin plot for ReAdmis and 'TotalCharge'
print("Categorical vs. Continuous")
print("Patients with readmitted are charged more than those without a positive 'ReAdmis' status. This makes s
plt.figure(layout='constrained', figsize=(3, 3))
sns.violinplot(data=df_med, x="ReAdmis", y="TotalCharge")
```

```
plt.title('ViolinPlot for ReAdmis and TotalCharge')
# Show the plot
plt.show()
print('\n')

# Count plot for ReAdmis and 'Gender'
print("Categorical vs. Categorical")
print("A countPlot to visualize the count of observations for each combination of `ReAdmis` and `Gender`. The
plt.figure(layout='constrained', figsize=(3, 3))
sns.countplot(data=df_med, x="ReAdmis", hue="Gender")
plt.title('CountPlot for ReAdmis and Gender')
# Show the plot
plt.show()
print('\n')
```

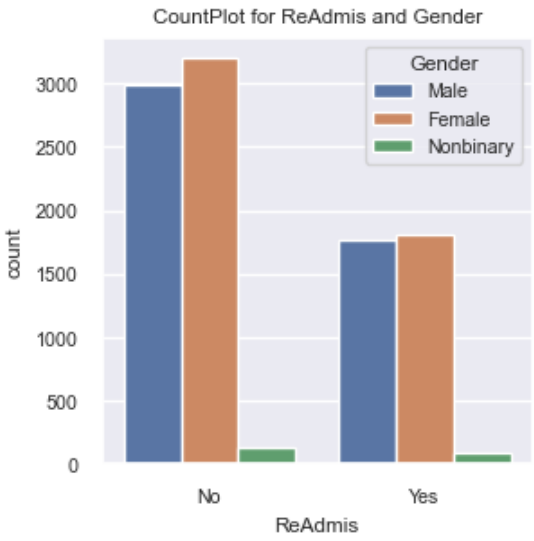
Continuous vs. Continuous
There is a positive correlation between 'Initial_days' and 'TotalCharge', which makes sense; They are describ
ed as 'Daily Charges' so the longer a patient stays in the hospital, the more they will be charged.



Categorical vs. Continuous
Patients with readmitted are charged more than those without a positive 'ReAdmis' status. This makes sense be
cause they are readmitted to the hospital, which means they have already incurred charges. Interestingly, we
can see this in the data; as the lowest `ReAdmis`=Yes value is just under 6000.00.



Categorical vs. Categorical
A countPlot to visualize the count of observations for each combination of `ReAdmis` and `Gender`. There does
not appear to be any significant relationships here as the visualized ratios are similar.



E: Summarize the implications of your data analysis by doing the following:

E1. Discuss the results of the hypothesis test.

In the context of the chi-square analysis we performed, the hypotheses is framed as follows:

- **Null hypothesis H_0:** There is not a significant association between `Services` and `ReAdmis`
- **Alternative hypothesis H_1:** There is a significant association between `Services` and `ReAdmis`

Null hypothesis H_0: This implies that the likelihood of a patient being readmitted does not depend on the type of service they received during their initial visit to the hospital. In other words, whether a patient received Blood Work, a CT Scan, Intravenous therapy, or an MRI does not appear to play a role in their chances of being readmitted.

Alternative hypothesis H_1: This suggests that the type of services a patient receives has some influence on their likelihood of being readmitted to the hospital. The association could mean that certain services are more frequently associated with readmission than others.

- Analysis of Hypothesis Test Based on the chi-square test:

The chi-square statistic was **8.893**. The p-value was **0.03075**. Since the `p-value` (0.03075) is less than the standard `alpha` of 0.05, we reject the null hypothesis H_0, which suggests that there is a statistically significant association between the type of service received (`Services`) and the likelihood of readmission (`ReAdmis`).

Interpretation:

Rejecting the null hypothesis supports the alternative hypothesis H_1. This suggests readmission to the hospital may be influenced by the type of services they received on their first visit to the hospital. In the observed data, certain services are associated with different rates of readmission, which could be higher or lower than what would be expected if there was no association at all.

E2. Discuss the limitations of your data analysis.

The Chi-square test assumes that the cases are independent and that the data fits within categorical variables appropriately. If these assumptions are violated, the results might not be reliable. The results are only as good as the data, and there is much nuance and information one is left wanting as the analysis progressed. The inability to communicate with stakeholders directly places considerable limitation on the analysis process: the more assumption one has to make, the more opportunity there is for error in those assumptions and that can result in lower confidence in the results overall.

It is important to remember that this association does not imply causation. Additionally, the analysis does not tell us which specific services are associated with readmission, or why. The observed association might be influenced by other factors not accounted for in this analysis, such as patient health status, severity of the medical issue, or other demographic and clinical variables. Additionally, the distribution of services provided to patients in the dataset is not uniform, with some services being more common than others, and more expensive than others. This could influence the observed association between services and readmission rates.

For example, blood work is one of the most common diagnostic services provided to patients and is often one of the first tools in the diagnostic tool kit. It can be less expensive when compared to medical imaging, particularly as it relates to time, planning, and resources needs in general. These all contribute to patient well-being and financial costs. Resource intensive modalities are generally not as available as less time consuming tests due the length of time a single machine can be taken up by a single patient. This could influence the observed association between services and readmission rates. In the context of the data, one can see by observing the frequency of the services provided that blood work is the most common service provided to patients, followed by intravenous therapy, CT scans, and MRIs. Therefore, the association that `Services` has with `ReAdmis` could be influenced by the frequency of the services provided. This is something that should be investigated further.

E3. Recommend a course of action based on your results.

In regard to the original question, although this analysis suggests that the type of medical services a patient receives during their initial hospitalization may indeed influence their likelihood of readmission, this answer is the starting point of further investigation, which is required to establish this conclusion with confidence.

From an analysis standpoint, it is important to conduct further investigation to confirm the association between services and readmission rates. Additionally, it would be crucial to fully understand the nature of this relationship, particularly as it relates to the commonality of the services provided to patients. This could help us determine whether the association is driven by the frequency or the type of services.

For stakeholders, if the services are indeed associated with readmission, then it would be important to understand which specific services contribute and why. Investigating which services are more associated with readmission could help identify specific areas for improvement in patient care or follow-up procedures. Moreover, the association might not be with the services themselves but rather with the quality of care in different departments, which might influence readmission rates. Ensuring consistent high-quality care across all services could be a key factor in reducing readmissions. This would help the hospital implement targeted interventions to improve patient outcomes and reduce unnecessary costs.

G & H: References

- Western Governors University. (2023, December 21). D207 - Medical_clean Dataset. Retrieved from https://lrps.wgu.edu/provision/227079957
- Western Governors University IT Department. (2023). R or Python? How to decide which programming language to learn. Retrieved from https://www.wgu.edu/online-it-degrees/programming-languages/r-or-python.html#
- Datacamp. (2023, December 12). D207 - Exploratory Data Analysis. Retrieved from https://app.datacamp.com/learn/custom-tracks/custom-d207-exploratory-data-analysis
- Sewell, Dr. (2023). WGU D207 Exploratory Data Analysis [Webinars]. WGU Webex. Accessed December, 2023. https://wgu.webex.com/webappng/sites/wgu/meeting/info/c4aca2eac546482880f1557c938abf40?siteurl=wgu&MTID=me73470c2eac9e863c6f47a3d5b6d2f26
- Seaborn Developers. (2023). seaborn.scatterplot — seaborn 0.11.2 documentation. Retrieved December 22, 2023, from https://seaborn.pydata.org/generated/seaborn.scatterplot.html