Week 8 Deliverables

- Group Name: Gamma Group
- Name: Matthew Iversen, Jonathan Charles Allen
- Email: matt.w.iversen@gmail.com
- Country: USA
- College/Company: N/A
- Specialization: Data Science
- Github Repo Link: Github/Week8

Problem Description

This project revolves around understanding the persistence of a drug based on physician prescriptions. The objective is to automate predictions of a persistency flag using a classification machine learning model. After this analysis, we will identify the most important factors in predicting persistence.

Data Understanding

Through this notebook, we will come to understand the limitations, data types, and issues with the data.

Imports

In [ ]:
```
import pandas as pd

# working with excel files
%pip install openpyxl
```

```
Requirement already satisfied: openpyxl in c:\users\matthew iversen\appdata\local\programs\pyth
on\python310\lib\site-packages (3.1.2)
Requirement already satisfied: et-xmlfile in c:\users\matthew iversen\appdata\local\programs\py
thon\python310\lib\site-packages (from openpyxl) (1.1.0)
Note: you may need to restart the kernel to use updated packages.
[notice] A new release of pip is available: 23.2.1 -> 23.3.1
[notice] To update, run: python.exe -m pip install --upgrade pip
```

Util File

In [ ]:
```
%%writefile testutility.py
import logging
import os
import subprocess
import yaml
import pandas as pd
import datetime
import gc
import re
import difflib
```

```python
# summary of a data file
def summary(df: pd.DataFrame, file_path: str) -> None:
    # filesize in mb
    file_size_bytes = os.path.getsize(file_path)
    file_size_mb = file_size_bytes / (1024 * 1024)

    # get dimensions
    total_rows = len(df)
    total_columns = len(df.columns)

    print(f"Total number of rows: {total_rows}")
    print(f"Total number of columns: {total_columns}")
    print(f"File size: {file_size_mb:.2f} MB")


# prints the number of nans in each column
def show_nan_all_columns(df: pd.DataFrame) -> None:
    nan_counts = df.isnull().sum().sort_values(ascending=False)
    print(f"NaN Counts:\n{nan_counts}")


# prints the number of nans in columns with nans
def show_nan_columns(df: pd.DataFrame) -> None:
    nan_counts = df.isnull().sum().sort_values(ascending=False)
    nan_counts = nan_counts[nan_counts > 0]
    print(f"NaN Counts:\n{nan_counts}")


# returns what features have nans
def find_nan_columns(df: pd.DataFrame) -> pd.Index:
    nan_features = df.isnull().sum()
    non_zero_nans = nan_features[nan_features > 0]
    return non_zero_nans.index


# changes the number of columns seen on output
def set_pd_max_columns(max_columns: int | None) -> None:
    pd.set_option("display.max_columns", max_columns)


# changes the number of rows seen on output
def set_pd_max_rows(max_rows: int | None) -> None:
    pd.set_option("display.max_rows", max_rows)


def detect_outliers_iqr(data: pd.DataFrame) -> pd.DataFrame:
    """
    Detects and returns any outliers for a given dataframe.
    """
    Q1 = data.quantile(0.25)
    Q3 = data.quantile(0.75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR

    # filter for outliers
    outliers = data[(data < lower_bound) | (data > upper_bound)]
    return outliers


def show_spelling_errors(
    df: pd.DataFrame, similarity_threshold: float, exclude_columns: list[str]
```

```python
) -> None:
    """This prints all of the observations in a column that are similar above a threshold

    Args:
        df (pd.DataFrame): Pandas DataFrame
        similarity_threshold (float): Decimal of how similar of results we want to see (0.0-1.0
        exclude_columns (list[str]): List of columns you want to exclude from spelling check
    """

    spelling_errors = {}

    if exclude_columns is None:
        exclude_columns = []

    # find potential spelling errors for object columns
    for column in df.select_dtypes(include="object"):
        if column not in exclude_columns:
            unique_values = df[column].dropna().unique()
            potential_errors = []

            for i, value1 in enumerate(unique_values):
                for value2 in unique_values[i + 1 :]:
                    similarity = difflib.SequenceMatcher(None, value1, value2).ratio()
                    if similarity > similarity_threshold:
                        potential_errors.append((value1, value2))

            if potential_errors:
                spelling_errors[column] = potential_errors

    # print the errors
    for column, errors in spelling_errors.items():
        print(f"Potential spelling errors in column '{column}':")
        for error in errors:
            print(f"- '{error[0]}' might be similar to '{error[1]}'")


def remove_duplicates(df: pd.DataFrame) -> pd.DataFrame:
    """Prints info about and removes duplicate columns and rows

    Args:
        df (pd.DataFrame): Incoming Pandas DataFrame

    Returns:
        pd.DataFrame: Pandas DataFrame with no duplicate rows/columns
    """

    # count and remove duplicate rows
    duplicate_rows = df[df.duplicated()]
    num_duplicate_rows = len(duplicate_rows)
    df = df.drop_duplicates()

    # count and remove duplicate columns
    duplicate_columns = df.columns[df.columns.duplicated()]
    num_duplicate_columns = len(duplicate_columns)
    df = df.loc[:, ~df.columns.duplicated()]

    print(f"Number of duplicate rows removed: {num_duplicate_rows}")
    print(f"Number of duplicate columns removed: {num_duplicate_columns}")

    return df
```

```
Overwriting testutility.py
```

In [ ]:
```
# import util file for use
import testutility as util
```

Read the Data

In [ ]:
```
file_path = "../week-7/Healthcare_dataset.xlsx"
df = pd.read_excel(file_path, sheet_name=1)  # data is on the second sheet of the file

util.set_pd_max_columns(None)
util.set_pd_max_rows(None)
df.head()
```

Out[ ]:

	Ptid	Persistency_Flag	Gender	Race	Ethnicity	Region	Age_Bucket	Ntm_Speciality	Ntm
0	P1	Persistent	Male	Caucasian	Not Hispanic	West	>75	GENERAL PRACTITIONER	
1	P2	Non-Persistent	Male	Asian	Not Hispanic	West	55-65	GENERAL PRACTITIONER	
2	P3	Non-Persistent	Female	Other/Unknown	Hispanic	Midwest	65-75	GENERAL PRACTITIONER	
3	P4	Non-Persistent	Female	Caucasian	Not Hispanic	Midwest	>75	GENERAL PRACTITIONER	
4	P5	Non-Persistent	Female	Caucasian	Not Hispanic	Midwest	>75	GENERAL PRACTITIONER	

Summarize the File

In [ ]:
```
# use util summary
util.summary(df, file_path)
```

```
Total number of rows: 3424
Total number of columns: 69
File size: 0.88 MB
```

Look at Feature Data Types

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3424 entries, 0 to 3423
Data columns (total 69 columns):
 #   Column                                                          Non-Null Count  Dtype
---  ------                                                          --------------  -----
 0   Ptid                                                            3424 non-null   object
 1   Persistency_Flag                                                3424 non-null   object
 2   Gender                                                          3424 non-null   object
 3   Race                                                            3424 non-null   object
 4   Ethnicity                                                       3424 non-null   object
 5   Region                                                          3424 non-null   object
 6   Age_Bucket                                                      3424 non-null   object
 7   Ntm_Speciality                                                  3424 non-null   object
 8   Ntm_Specialist_Flag                                             3424 non-null   object
 9   Ntm_Speciality_Bucket                                           3424 non-null   object
 10  Gluco_Record_Prior_Ntm                                          3424 non-null   object
 11  Gluco_Record_During_Rx                                          3424 non-null   object
 12  Dexa_Freq_During_Rx                                             3424 non-null   int64
 13  Dexa_During_Rx                                                  3424 non-null   object
 14  Frag_Frac_Prior_Ntm                                             3424 non-null   object
 15  Frag_Frac_During_Rx                                             3424 non-null   object
 16  Risk_Segment_Prior_Ntm                                          3424 non-null   object
 17  Tscore_Bucket_Prior_Ntm                                         3424 non-null   object
 18  Risk_Segment_During_Rx                                          3424 non-null   object
 19  Tscore_Bucket_During_Rx                                         3424 non-null   object
 20  Change_T_Score                                                  3424 non-null   object
 21  Change_Risk_Segment                                             3424 non-null   object
 22  Adherent_Flag                                                   3424 non-null   object
 23  Idn_Indicator                                                   3424 non-null   object
 24  Injectable_Experience_During_Rx                                 3424 non-null   object
 25  Comorb_Encounter_For_Screening_For_Malignant_Neoplasms          3424 non-null   object
 26  Comorb_Encounter_For_Immunization                               3424 non-null   object
 27  Comorb_Encntr_For_General_Exam_W_O_Complaint,_Susp_Or_Reprtd_Dx 3424 non-null   object
 28  Comorb_Vitamin_D_Deficiency                                     3424 non-null   object
 29  Comorb_Other_Joint_Disorder_Not_Elsewhere_Classified            3424 non-null   object
 30  Comorb_Encntr_For_Oth_Sp_Exam_W_O_Complaint_Suspected_Or_Reprtd_Dx 3424 non-null   object
 31  Comorb_Long_Term_Current_Drug_Therapy                           3424 non-null   object
 32  Comorb_Dorsalgia                                                3424 non-null   object
 33  Comorb_Personal_History_Of_Other_Diseases_And_Conditions        3424 non-null   object
 34  Comorb_Other_Disorders_Of_Bone_Density_And_Structure            3424 non-null   object
 35  Comorb_Disorders_of_lipoprotein_metabolism_and_other_lipidemias 3424 non-null   object
 36  Comorb_Osteoporosis_without_current_pathological_fracture       3424 non-null   object
 37  Comorb_Personal_history_of_malignant_neoplasm                   3424 non-null   object
 38  Comorb_Gastro_esophageal_reflux_disease                         3424 non-null   object
 39  Concom_Cholesterol_And_Triglyceride_Regulating_Preparations     3424 non-null   object
 40  Concom_Narcotics                                                3424 non-null   object
 41  Concom_Systemic_Corticosteroids_Plain                           3424 non-null   object
 42  Concom_Anti_Depressants_And_Mood_Stabilisers                    3424 non-null   object
 43  Concom_Fluoroquinolones                                         3424 non-null   object
 44  Concom_Cephalosporins                                           3424 non-null   object
 45  Concom_Macrolides_And_Similar_Types                             3424 non-null   object
 46  Concom_Broad_Spectrum_Penicillins                               3424 non-null   object
 47  Concom_Anaesthetics_General                                     3424 non-null   object
 48  Concom_Viral_Vaccines                                           3424 non-null   object
 49  Risk_Type_1_Insulin_Dependent_Diabetes                          3424 non-null   object
 50  Risk_Osteogenesis_Imperfecta                                    3424 non-null   object
 51  Risk_Rheumatoid_Arthritis                                       3424 non-null   object
 52  Risk_Untreated_Chronic_Hyperthyroidism                          3424 non-null   object
 53  Risk_Untreated_Chronic_Hypogonadism                             3424 non-null   object
 54  Risk_Untreated_Early_Menopause                                  3424 non-null   object
 55  Risk_Patient_Parent_Fractured_Their_Hip                         3424 non-null   object
 56  Risk_Smoking_Tobacco                                            3424 non-null   object
```

```
 57  Risk_Chronic_Malnutrition_Or_Malabsorption                          3424 non-null   object
 58  Risk_Chronic_Liver_Disease                                          3424 non-null   object
 59  Risk_Family_History_Of_Osteoporosis                                 3424 non-null   object
 60  Risk_Low_Calcium_Intake                                             3424 non-null   object
 61  Risk_Vitamin_D_Insufficiency                                        3424 non-null   object
 62  Risk_Poor_Health_Frailty                                            3424 non-null   object
 63  Risk_Excessive_Thinness                                             3424 non-null   object
 64  Risk_Hysterectomy_Oophorectomy                                      3424 non-null   object
 65  Risk_Estrogen_Deficiency                                            3424 non-null   object
 66  Risk_Immobilization                                                 3424 non-null   object
 67  Risk_Recurring_Falls                                                3424 non-null   object
 68  Count_Of_Risks                                                      3424 non-null   int64
dtypes: int64(2), object(67)
memory usage: 1.8+ MB
```

The data is all objects, aside from 2 int64 columns.

Checking for Outliers

In [ ]:
```
df.Dexa_Freq_During_Rx.unique(), df.Count_Of_Risks.unique()
```

Out[ ]:
```
(array([  0,   2,   7,   3,   5,  20,  13,   1,   6,  12,   4,  10,  25,
         11,  18,  21,  15,  28,  22,  37,  14,   8,   9,  17,  81,  42,
         16,  30,  19,  45,  27,  24,  58,  26,  23,  33, 110,  36,  34,
         88,  66,  32, 118,  48,  69,  38,  40,  68,  52,  50, 146,  44,
         35,  39, 108,  54,  72,  29], dtype=int64),
 array([0, 2, 1, 3, 4, 5, 6, 7], dtype=int64))
```

In the context of these features, neither appear to have outliers. The remainder of the features are categorical and cannot be analyzed for outliers.

Checking the Spelling of the Data

In [ ]:
```
util.show_spelling_errors(df, 0.80, ['Ptid']) # excluding PTID since they are all similar
```

```
Potential spelling errors in column 'Persistency_Flag':
- 'Persistent' might be similar to 'Non-Persistent'
Potential spelling errors in column 'Ntm_Speciality':
- 'UROLOGY' might be similar to 'NEUROLOGY'
- 'NEUROLOGY' might be similar to 'NEPHROLOGY'
- 'RADIOLOGY' might be similar to 'CARDIOLOGY'
```

No spelling issues found as these are intentional.

Checking for Duplicates

In [ ]:
```
df = util.remove_duplicates(df)
```

```
Number of duplicate rows removed: 0
Number of duplicate columns removed: 0
```

Check for NaN Values

In [ ]:
```
util.show_nan_all_columns(df)
```

```
NaN Counts:
Ptid                                                                  0
Concom_Cephalosporins                                                 0
Risk_Osteogenesis_Imperfecta                                          0
Risk_Type_1_Insulin_Dependent_Diabetes                                0
Concom_Viral_Vaccines                                                 0
Concom_Anaesthetics_General                                           0
Concom_Broad_Spectrum_Penicillins                                     0
Concom_Macrolides_And_Similar_Types                                   0
Concom_Fluoroquinolones                                               0
Comorb_Disorders_of_lipoprotein_metabolism_and_other_lipidemias       0
Concom_Anti_Depressants_And_Mood_Stabilisers                          0
Concom_Systemic_Corticosteroids_Plain                                 0
Concom_Narcotics                                                      0
Concom_Cholesterol_And_Triglyceride_Regulating_Preparations           0
Comorb_Gastro_esophageal_reflux_disease                               0
Comorb_Personal_history_of_malignant_neoplasm                         0
Risk_Rheumatoid_Arthritis                                             0
Risk_Untreated_Chronic_Hyperthyroidism                                0
Risk_Untreated_Chronic_Hypogonadism                                   0
Risk_Untreated_Early_Menopause                                        0
Risk_Patient_Parent_Fractured_Their_Hip                               0
Risk_Smoking_Tobacco                                                  0
Risk_Chronic_Malnutrition_Or_Malabsorption                            0
Risk_Chronic_Liver_Disease                                            0
Risk_Family_History_Of_Osteoporosis                                   0
Risk_Low_Calcium_Intake                                               0
Risk_Vitamin_D_Insufficiency                                          0
Risk_Poor_Health_Frailty                                              0
Risk_Excessive_Thinness                                               0
Risk_Hysterectomy_Oophorectomy                                        0
Risk_Estrogen_Deficiency                                              0
Risk_Immobilization                                                   0
Risk_Recurring_Falls                                                  0
Comorb_Osteoporosis_without_current_pathological_fracture             0
Comorb_Other_Disorders_Of_Bone_Density_And_Structure                  0
Persistency_Flag                                                      0
Ntm_Speciality_Bucket                                                 0
Frag_Frac_During_Rx                                                   0
Frag_Frac_Prior_Ntm                                                   0
Dexa_During_Rx                                                        0
Dexa_Freq_During_Rx                                                   0
Gluco_Record_During_Rx                                                0
Gluco_Record_Prior_Ntm                                                0
Ntm_Specialist_Flag                                                   0
Comorb_Personal_History_Of_Other_Diseases_And_Conditions              0
Ntm_Speciality                                                        0
Age_Bucket                                                            0
Region                                                                0
Ethnicity                                                             0
Race                                                                  0
Gender                                                                0
Risk_Segment_Prior_Ntm                                                0
Tscore_Bucket_Prior_Ntm                                               0
Risk_Segment_During_Rx                                                0
Tscore_Bucket_During_Rx                                               0
Change_T_Score                                                        0
Change_Risk_Segment                                                   0
Adherent_Flag                                                         0
Idn_Indicator                                                         0
Injectable_Experience_During_Rx                                       0
Comorb_Encounter_For_Screening_For_Malignant_Neoplasms                0
```

```
Comorb_Encounter_For_Immunization                                       0
Comorb_Encntr_For_General_Exam_W_O_Complaint,_Susp_Or_Reprtd_Dx         0
Comorb_Vitamin_D_Deficiency                                             0
Comorb_Other_Joint_Disorder_Not_Elsewhere_Classified                    0
Comorb_Encntr_For_Oth_Sp_Exam_W_O_Complaint_Suspected_Or_Reprtd_Dx      0
Comorb_Long_Term_Current_Drug_Therapy                                   0
Comorb_Dorsalgia                                                        0
Count_Of_Risks                                                          0
dtype: int64
```

No NaNs found.

Conclusion

- No outliers were detected in the 2 numerical features
- No NaN values were found in any features
- No duplicate rows were found
- No duplicate columns were found
- No spelling errors were detected in object columns

Due to the cleanliness of this data, there is no need to make any changes at this time.