cs109a_MS3_EDA

October 31, 2023

1 cs109a Final Project Milestone 3: EDA

[1]:

```
# Import libraries
import os
import time
import numpy as np
import pandas as pd
%matplotlib inline
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.decomposition import PCA
from sklearn.preprocessing import StandardScaler
from sklearn.impute import SimpleImputer, KNNImputer
from sklearn.linear_model import LinearRegression, LogisticRegression,␣
  ↪LogisticRegressionCV
from sklearn.neighbors import KNeighborsRegressor
from sklearn.model_selection import train_test_split, cross_validate
from sklearn.metrics import r2_score, confusion_matrix, classification_report,␣
  ↪roc_curve
from sklearn.metrics import roc_auc_score, precision_recall_curve,␣
  ↪average_precision_score
import warnings
warnings.filterwarnings("ignore")
plt.style.use('seaborn-notebook')
sns.set_style('darkgrid')
# pandas tricks for better display
pd.options.display.max_columns = 50
pd.options.display.max_rows = 500
pd.options.display.max_colwidth = 100
pd.options.display.precision = 3
```

1.1 Helper Functions

[2]:

```
# helper functions

def visualize_class_imbalance(series, partition='Training', figsize=(5, 2)):
    # Visualize class imbalance
    plt.figure(figsize=figsize)
    series.value_counts().plot(kind='barh')
    plt.title(f'Class Distribution for {partition} Data')
    plt.ylabel('Class Label')
    plt.xlabel('Number of Samples')
    plt.show()

def create_A1Cencounters(df):
    # 1. create copy of 'A1Cresult' in new column
    df['A1Cencounters'] = df['A1Cresult']
    # 2. convert 'Norm' and '>7' to 'norm'
    df['A1Cencounters'] = df['A1Cencounters'].replace(['normal', '>7'],␣
 ↪'normal')
    # 3. convert '>8' with 'change' == 'No'
    df.loc[(df['A1Cencounters'] == '>8') & (df['change'] == 'No'),␣
 ↪'A1Cencounters'] = '>8_no_change'
    # 4. convert '>8' with 'change' == 'Ch'
    df.loc[(df['A1Cencounters'] == '>8') & (df['change'] == 'Ch'),␣
 ↪'A1Cencounters'] = '>8_yes_change'
    return df

def visualize_missingness(df, partition='Training', figsize=(6, 7)):
    # calculate missing data percentages
    missing_data_percentage = (df.isnull().sum() / len(df)) * 100
    # sort by percentage
    missing_data_percentage = missing_data_percentage.
 ↪sort_values(ascending=True)
    # visualize using horizontal barplot
    plt.figure(figsize=figsize)
    missing_data_percentage.plot(kind='barh')
    plt.title(f'Percentage of Missing Data by Column for {partition} Data')
    plt.xlabel('Percentage Missing (%)')
    plt.ylabel('Columns')
    plt.show()

def missing_values_table(df, partition='training'):
    # count the missing values for each column
    missing_values = df.isnull().sum()
    # calculate the percentage of missing values
    missing_percentage = (100 * df.isnull().sum() / len(df))
    # create a table with the results
    missing_values_table = pd.concat([missing_values, missing_percentage],␣
 ↪axis=1)
    # rename the columns
    missing_values_table_columns = missing_values_table.rename(
        columns = {0 : 'Missing Values', 1 : '% of Total Values'})
    # sort the table by percentage of missing in descending order
    missing_values_table_columns = missing_values_table_columns[
        missing_values_table_columns.iloc[:,1] != 0].sort_values(
        '% of Total Values', ascending=False).round(1)
    # print a summary
    print(f"The {partition} data have " + str(df.shape[1]) + " columns.\n"
            "There are " + str(missing_values_table_columns.shape[0]) +
            " columns that have missing values.\n")
    # return the dataframe with missing info
    return missing_values_table_columns

def handle_missing_data(df):
    # keep track of the original columns
    original_columns = df.columns.tolist()
    # create dummy columns only for columns with missing data
    for col in original_columns:
        if df[col].isna().any():
            df[col + "_is_missing"] = df[col].isna().astype(int)
    # define imputers
    numeric_imputer = SimpleImputer(strategy='median')
    categorical_imputer = SimpleImputer(strategy='most_frequent')
    # impute median values for numeric columns and most frequent value for␣
 ↪non-numeric columns
    for col in original_columns:
        if df[col].dtype in [np.float64, np.int64]:
            df[col] = numeric_imputer.fit_transform(df[[col]]).flatten()
        else:
            df[col] = categorical_imputer.fit_transform(df[[col]]).flatten()
    return df

def scale_data(df, columns_to_scale):
    scaler = StandardScaler()
    df[columns_to_scale] = scaler.fit_transform(df[columns_to_scale])
    return df
```

1.2 Load Data

[3]:

```
# load data
df = pd.read_csv('../data/diabetic_data.csv', na_values='?')

# examine first 5 rows of dataframe
print(f'Shape of diabetic data: {df.shape}\n')
display(df.head())
```

```
Shape of diabetic data: (101766, 50)


   encounter_id  patient_nbr             race  gender      age weight  \
0       2278392      8222157        Caucasian  Female   [0-10)    NaN
```

```
1        149190     55629189        Caucasian  Female  [10-20)    NaN
2         64410     86047875  AfricanAmerican  Female  [20-30)    NaN
3        500364     82442376        Caucasian    Male  [30-40)    NaN
4         16680     42519267        Caucasian    Male  [40-50)    NaN

   admission_type_id  discharge_disposition_id  admission_source_id  \
0                  6                        25                    1   
1                  1                         1                    7   
2                  1                         1                    7   
3                  1                         1                    7   
4                  1                         1                    7   

   time_in_hospital payer_code         medical_specialty  num_lab_procedures  \
0                 1        NaN  Pediatrics-Endocrinology                  41   
1                 3        NaN                       NaN                  59   
2                 2        NaN                       NaN                  11   
3                 2        NaN                       NaN                  44   
4                 1        NaN                       NaN                  51   

   num_procedures  num_medications  number_outpatient  number_emergency  \
0               0                1                  0                 0   
1               0               18                  0                 0   
2               5               13                  2                 0   
3               1               16                  0                 0   
4               0                8                  0                 0   

   number_inpatient  diag_1  diag_2 diag_3  number_diagnoses max_glu_serum  \
0                 0  250.83     NaN    NaN                 1           NaN   
1                 0     276  250.01    255                 9           NaN   
2                 1     648     250    V27                 6           NaN   
3                 0       8  250.43    403                 7           NaN   
4                 0     197     157    250                 5           NaN   

  A1Cresult metformin repaglinide nateglinide chlorpropamide glimepiride  \
0       NaN        No          No          No             No          No   
1       NaN        No          No          No             No          No   
2       NaN        No          No          No             No          No   
3       NaN        No          No          No             No          No   
4       NaN        No          No          No             No          No   

  acetohexamide glipizide glyburide tolbutamide pioglitazone rosiglitazone  \
0            No        No        No          No           No            No   
1            No        No        No          No           No            No   
2            No    Steady        No          No           No            No   
3            No        No        No          No           No            No   
4            No    Steady        No          No           No            No   

  acarbose miglitol troglitazone tolazamide examide citoglipton insulin  \
0       No       No           No         No      No          No      No
1       No       No           No         No      No          No      Up
2       No       No           No         No      No          No      No
3       No       No           No         No      No          No      Up
4       No       No           No         No      No          No  Steady

  glyburide-metformin glipizide-metformin glimepiride-pioglitazone  \
0                  No                  No                       No
1                  No                  No                       No
2                  No                  No                       No
3                  No                  No                       No
4                  No                  No                       No

  metformin-rosiglitazone metformin-pioglitazone change diabetesMed readmitted
0                      No                     No     No          No         NO
1                      No                     No     Ch         Yes        >30
2                      No                     No     No         Yes         NO
3                      No                     No     Ch         Yes         NO
4                      No                     No     Ch         Yes         NO
```

1.3 Recode Some Key Features

[4]:

```
# recode target to binary
df['readmitted'] = df['readmitted'].map({'NO': 0, '>30': 0, '<30': 1})
df['readmitted'].value_counts() # sanity check
```

[4]:

```
readmitted
0    90409
1    11357
Name: count, dtype: int64
```

[5]:

```
# change A1c test result values (Hemoglobin A1c)
df['A1Cresult'] = df['A1Cresult'].fillna('none')
df['A1Cresult'] = df['A1Cresult'].replace('Norm', 'normal')
df['A1Cresult'].value_counts() # sanity check
```

[5]:

```
A1Cresult
none      84748
>8         8216
normal     4990
>7         3812
Name: count, dtype: int64
```

[6]:

```
# create new HbA1c encounters feature (see paper)
df = create_A1Cencounters(df)
df['A1Cencounters'].value_counts() # sanity check
```

[6]:

```
A1Cencounters
none             84748
normal            8802
>8_yes_change     5349
>8_no_change      2867
Name: count, dtype: int64
```

[7]:

```
# change some glucose serum test result values
df['max_glu_serum'] = df['max_glu_serum'].fillna('none')
df['max_glu_serum'] = df['max_glu_serum'].replace('Norm', 'normal')
df['max_glu_serum'].value_counts() # sanity check
```

[7]:

```
max_glu_serum
none      96420
normal     2597
>200       1485
>300       1264
Name: count, dtype: int64
```

1.4 Partition Data

[8]:

```
X_train, X_test, y_train, y_test = train_test_split(
    df.drop('readmitted', axis=1),
    df['readmitted'],
    train_size=0.8,
    random_state=109
)

X_train.shape, X_test.shape, y_train.shape, y_test.shape
```

[8]:

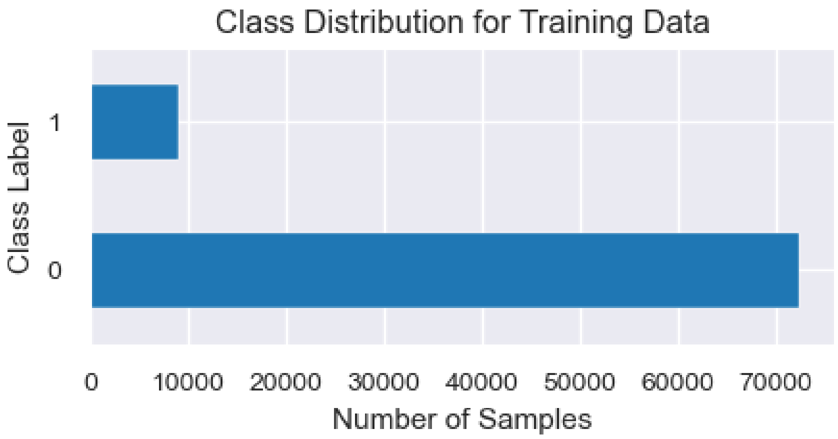
```
((81412, 50), (20354, 50), (81412,), (20354,))
```

1.5 EDA

1.5.1 Class Imbalance

[9]:

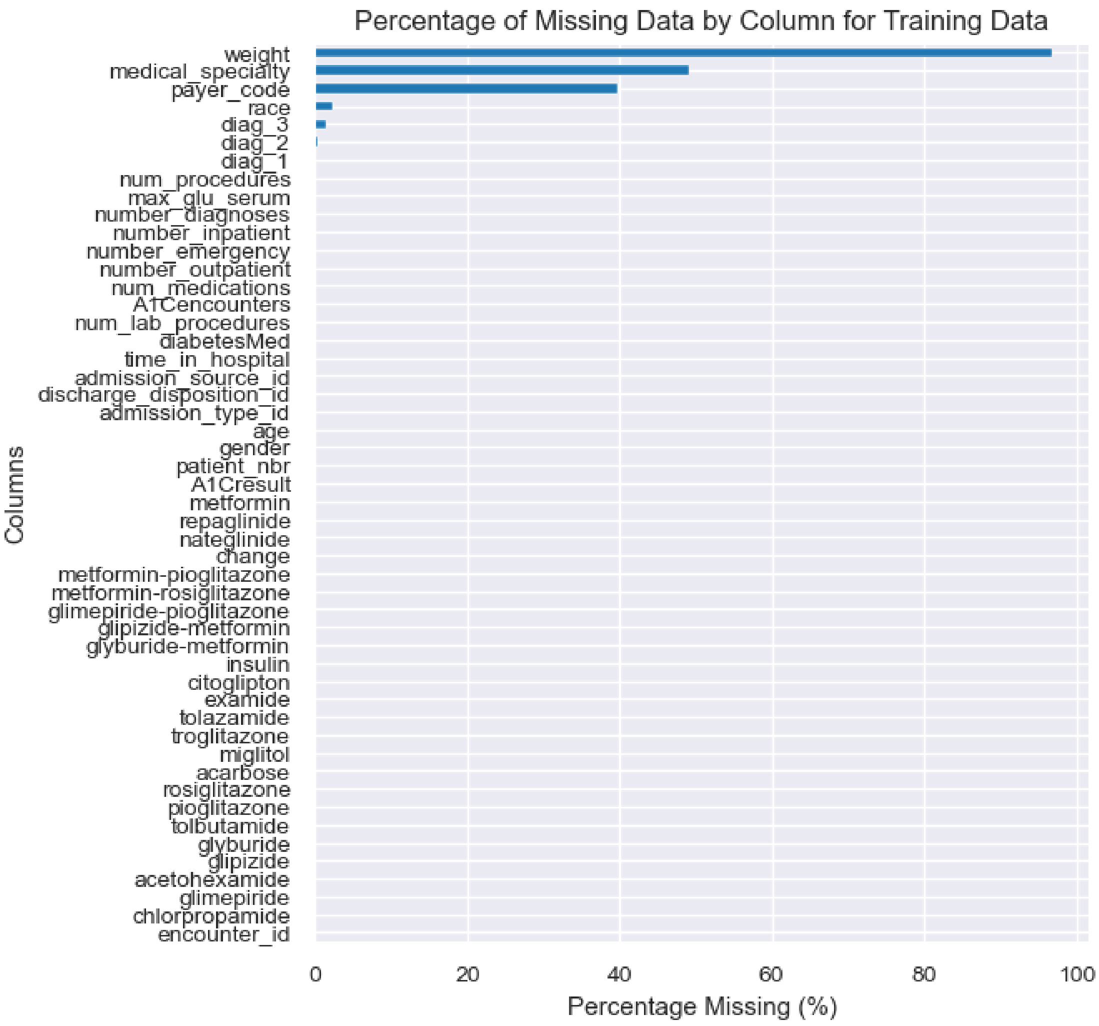
```
# class imbalance
visualize_class_imbalance(y_train)
```



Question: Can we use SMOTE (synthetic minority oversampling)? Seems like it's only available in `imblearn`, which isn't in the list of class packages. If not, should we just use classifiers that can adjust class weights, or roll our own procedure to oversample the minority class?

1.5.2 Missing Data

[10]:
```
# visualize missingness
visualize_missingness(X_train)
```



[11]:

```
# missing values table
missing_values_table(X_train)
```

```
The training data have 50 columns.
There are 7 columns that have missing values.
```

[11]:

```
                   Missing Values  % of Total Values
weight                      78862               96.9
medical_specialty           39927               49.0
payer_code                  32265               39.6
race                         1846                2.3
diag_3                       1129                1.4
diag_2                        282                0.3
diag_1                         18                0.0
```

[12]:
```
X_train[['weight', 'medical_specialty', 'payer_code']].head(10)
```

[12]:
```
         weight medical_specialty payer_code
98299       NaN               NaN         MC
40063       NaN               NaN        NaN
25045       NaN       Orthopedics        NaN
2178        NaN           Urology        NaN
69063       NaN               NaN         MC
82824       NaN       Orthopedics         MC
29895       NaN               NaN         HM
64777       NaN   InternalMedicine        MC
78362       NaN   Emergency/Trauma        PO
45146  [75-100)               NaN         MC
```

[13]:
```
X_train['payer_code'].value_counts()
```

[13]:
```
payer_code
MC    25973
HM     4993
SP     4022
BC     3702
MD     2804
CP     2037
UN     1934
CM     1534
OG      829
PO      458
DM      443
CH      123
WC      105
OT       80
MP       68
SI       41
FR        1
Name: count, dtype: int64
```

[14]:
```
X_train['medical_specialty'].value_counts()
```

[14]:
```
medical_specialty
InternalMedicine                       11824
Emergency/Trauma                        6061
Family/GeneralPractice                  5943
Cardiology                              4232
Surgery-General                         2480
Nephrology                              1286
Orthopedics                             1105
Orthopedics-Reconstructive               976
```

Radiologist	909
Pulmonology	699
Psychiatry	692
Urology	540
Surgery-Cardiovascular/Thoracic	535
ObstetricsandGynecology	521
Gastroenterology	452
Surgery-Vascular	414
Surgery-Neuro	384
PhysicalMedicineandRehabilitation	319
Oncology	276
Pediatrics	196
Hematology/Oncology	156
Neurology	152
Pediatrics-Endocrinology	125
Otolaryngology	108
Endocrinology	91
Surgery-Thoracic	90
Psychology	85
Podiatry	79
Surgery-Cardiovascular	77
Pediatrics-CriticalCare	72
Hematology	68
Gynecology	46
Radiology	44
Hospitalist	44
Surgeon	33
InfectiousDiseases	33
Osteopath	31
Surgery-Plastic	31
Ophthalmology	29
SurgicalSpecialty	25
Pediatrics-Pulmonology	22
Obsterics&Gynecology-GynecologicOnco	20
Obstetrics	16
Anesthesiology-Pediatric	15
Pathology	15
Rheumatology	13
OutreachServices	11
Surgery-Colon&Rectal	11
Anesthesiology	10
PhysicianNotFound	10
Pediatrics-Neurology	9
Surgery-Pediatric	8
AllergyandImmunology	7
Surgery-Maxillofacial	7
Psychiatry-Child/Adolescent	6

```
Endocrinology-Metabolism                    6
Cardiology-Pediatric                        6
DCPTEAM                                     6
Dentistry                                   4
Pediatrics-Hematology-Oncology              4
Pediatrics-AllergyandImmunology             3
Resident                                    2
Pediatrics-EmergencyMedicine                2
Pediatrics-InfectiousDiseases               1
Proctology                                  1
Psychiatry-Addictive                        1
SportsMedicine                              1
Speech                                      1
Perinatology                                1
Dermatology                                 1
Neurophysiology                             1
Surgery-PlasticwithinHeadandNeck            1
Name: count, dtype: int64
```

1.5.3 Single Imputation

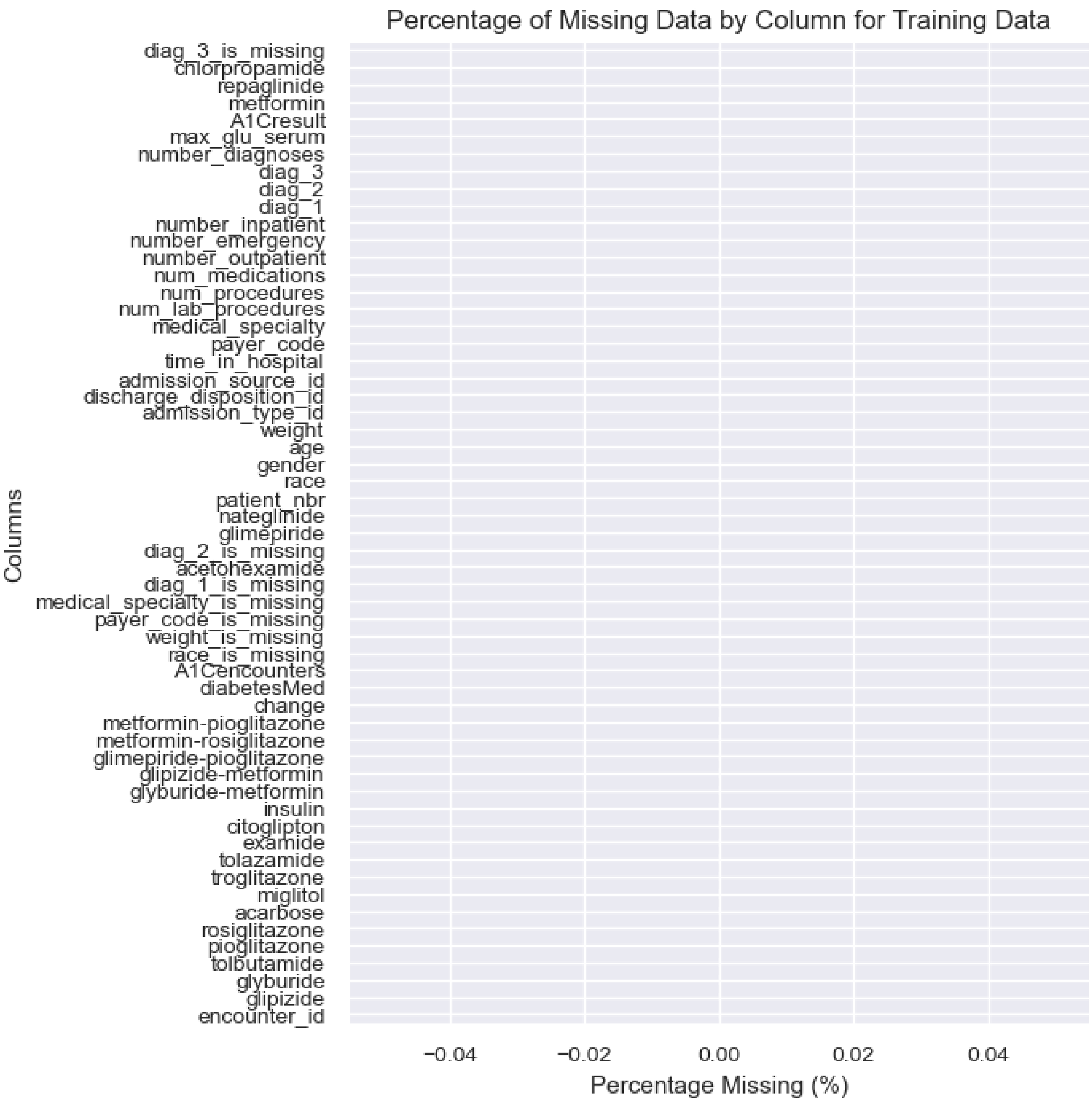
Question: Should we even bother doing imputation? Maybe just for `race` and the diagnoses feature and exclude those 3 with high levels of missingness?

[15]:

```
# deal with missing data
X_train_imp = handle_missing_data(X_train)
```

[16]:

```
# look at imputed data
visualize_missingness(X_train_imp, figsize=(6, 8))
```



1.5.4 Scaling

Question: Should we bother scaling? I think normalization would work better than standardization here for the various count variables (with the `num_` prefix) because they're probably quite right-skewed. They are mostly on similar scales already though.

```
[17]: # standardize data (just an example)
      X_train_imp_std = scale_data(X_train_imp, ['time_in_hospital'])
      X_train_imp_std.head()
```

```
[17]:        encounter_id  patient_nbr             race  gender       age    weight  \
      98299     3.991e+08    1.313e+08  AfricanAmerican  Female  [90-100)  [75-100)
      40063     1.246e+08    5.370e+07        Caucasian  Female   [50-60)  [75-100)
```

```
25045     8.417e+07    2.134e+07        Caucasian    Male   [80-90)  [75-100)
2178      1.416e+07    3.345e+06  AfricanAmerican  Female   [70-80)  [75-100)
69063     1.957e+08    5.873e+07        Caucasian  Female   [50-60)  [75-100)

       admission_type_id  discharge_disposition_id  admission_source_id  \
98299                1.0                      14.0                  7.0
40063                6.0                       1.0                  7.0
25045                3.0                       3.0                  1.0
2178                 2.0                       1.0                  1.0
69063                1.0                       3.0                  7.0

       time_in_hospital payer_code medical_specialty  num_lab_procedures  \
98299            -0.803         MC  InternalMedicine                21.0
40063            -0.134         MC  InternalMedicine                36.0
25045             0.201         MC       Orthopedics                33.0
2178              0.201         MC           Urology                54.0
69063             1.204         MC  InternalMedicine                62.0

       num_procedures  num_medications  number_outpatient  number_emergency  \
98299             0.0              7.0                0.0               1.0
40063             0.0             16.0                0.0               0.0
25045             1.0             36.0                0.0               0.0
2178              2.0             13.0                0.0               0.0
69063             1.0             24.0                0.0               0.0

       number_inpatient diag_1  diag_2 diag_3  number_diagnoses max_glu_serum  \
98299               0.0    428     599    411               9.0          none
40063               7.0    493  250.02    244               6.0          >300
25045               0.0    996     427    413               8.0          none
2178                1.0    592     599    427               9.0          none
69063               1.0    562     438    401               8.0          none

      A1Cresult metformin  … tolbutamide pioglitazone rosiglitazone  \
98299      none        No  …          No           No            No
40063      none        No  …          No           No            No
25045      none        No  …          No           No            Up
2178       none        No  …          No           No            No
69063      none        No  …          No           No            No

      acarbose miglitol troglitazone tolazamide examide citoglipton insulin  \
98299       No       No           No         No      No          No      No
40063       No       No           No         No      No          No      No
25045       No       No           No         No      No          No  Steady
2178        No       No           No         No      No          No  Steady
69063       No       No           No         No      No          No    Down

      glyburide-metformin glipizide-metformin glimepiride-pioglitazone  \
98299                  No                  No                       No
40063                  No                  No                       No
25045                  No                  No                       No
2178                   No                  No                       No
69063                  No                  No                       No

      metformin-rosiglitazone metformin-pioglitazone change diabetesMed  \
98299                      No                     No     No          No
40063                      No                     No     No          No
25045                      No                     No     Ch         Yes
2178                       No                     No     No         Yes
69063                      No                     No     Ch         Yes

      A1Cencounters race_is_missing weight_is_missing payer_code_is_missing  \
98299          none               0                 1                     0
40063          none               0                 1                     1
25045          none               0                 1                     1
2178           none               0                 1                     1
69063          none               0                 1                     0

      medical_specialty_is_missing diag_1_is_missing diag_2_is_missing  \
98299                            1                 0                 0
40063                            1                 0                 0
25045                            0                 0                 0
2178                             0                 0                 0
69063                            1                 0                 0

      diag_3_is_missing
98299                 0
40063                 0
25045                 0
2178                  0
69063                 0

[5 rows x 57 columns]
```

```
[ ]:
```