8 - Pandas-inspect_clean

March 22, 2023

Table of Contents

Ingest

Inspect and Clean

Looking for duplicates

Change datatypes

Change column names

Manage missing data

Imputing missing values

Check categorical data

Using visuals to get a sense of the data

Categorical data

Examine categorical data a little more closely

Dropping columns and rows

Quantitative data

Removing outliers

Exercise - 30 minutes

See Beer Notebook - Part 1

1 Ingest

[3]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib as mpl
import seaborn as sns
from numpy.random import randn
```

[4]:

```
df = pd.read_csv('https://raw.githubusercontent.com/jimcody2014/python-data/
 ↪main/diabetes_inspect.csv')
```

```
df.head()
```

[4]:

```
   encounter_id  patient_nbr             race  gender      age weight  \
0       2278392      8222157        Caucasian  Female      xyz      ?
1        149190     55629189        Caucasian  Female      NaN      ?
2         64410     86047875  AfricanAmerican  female  [20-30)      ?
3        500364     82442376        Caucasian     Mle  [30-40)      ?
4         16680     42519267        Caucasian       M  [40-50)      ?

   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  … glipizide glyburide  tolbutamide  miglitol  insulin  \
0                 1  …        No        No           No        No       No
1                 3  …        No        No           No        No       Up
2                 2  …    Steady        No           No        No       No
3                 2  …        No        No           No        No       Up
4                 1  …    Steady        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

  diabetesMed readmitted
0          No         NO
1         Yes        >30
2         Yes         NO
3         Yes         NO
4         Yes         NO

[5 rows x 33 columns]
```

[5]:

```
# A cursory look at the data
df.shape
```

[5]:

```
(101767, 33)
```

2 Inspect and Clean

2.0.1 Looking for duplicates

[6]:

```
# checking for duplicates
df.loc[df.duplicated()]# This will drop all duplicate rows
```

[6]:

```
        encounter_id  patient_nbr       race gender      age weight  \
101766     443867222    175429310  Caucasian   Male  [70-80)      ?

        admission_type_id  discharge_disposition_id  admission_source_id  \
101766                  1                         1                    7

        time_in_hospital  … glipizide glyburide  tolbutamide  miglitol  \
101766                 6  …        No        No           No        No

        insulin  glyburide-metformin  glipizide-metformin  \
101766       No                   No                   No

        glimepiride-pioglitazone diabetesMed readmitted
101766                        No          No         NO

[1 rows x 33 columns]
```

[7]:

```
df.drop_duplicates(keep = 'first', inplace = True)

# keep - which duplicate to keep, default is none!

df.loc[df.duplicated()]
```

[7]:

```
Empty DataFrame
Columns: [encounter_id, patient_nbr, race, gender, age, weight,
admission_type_id, discharge_disposition_id, admission_source_id,
time_in_hospital, payer_code, medical_specialty, num_lab_procedures,
num_procedures, num_medications, number_outpatient, number_emergency,
number_inpatient, diag_1, max_glu_serum, A1Cresult, metformin, glimepiride,
glipizide, glyburide, tolbutamide, miglitol, insulin, glyburide-metformin,
glipizide-metformin, glimepiride-pioglitazone, diabetesMed, readmitted]
Index: []

[0 rows x 33 columns]
```

2.0.2 Change datatypes

[8]:

```
# Are we ok with the data types?
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
Int64Index: 101766 entries, 0 to 101765
Data columns (total 33 columns):
 #   Column                    Non-Null Count   Dtype
---  ------                    --------------   -----
 0   encounter_id              101766 non-null  int64
 1   patient_nbr               101766 non-null  int64
 2   race                      101766 non-null  object
 3   gender                    101766 non-null  object
 4   age                       101765 non-null  object
 5   weight                    101766 non-null  object
 6   admission_type_id         101766 non-null  int64
 7   discharge_disposition_id  101766 non-null  int64
 8   admission_source_id       101766 non-null  int64
 9   time_in_hospital          101766 non-null  int64
 10  payer_code                101766 non-null  object
 11  medical_specialty         101766 non-null  object
 12  num_lab_procedures        101766 non-null  int64
 13  num_procedures            101766 non-null  int64
 14  num_medications           101757 non-null  float64
 15  number_outpatient         101766 non-null  int64
 16  number_emergency          101766 non-null  int64
 17  number_inpatient          101766 non-null  int64
 18  diag_1                    101766 non-null  object
 19  max_glu_serum             101766 non-null  object
 20  A1Cresult                 101766 non-null  object
 21  metformin                 101766 non-null  object
 22  glimepiride               101766 non-null  object
 23  glipizide                 101766 non-null  object
 24  glyburide                 101766 non-null  object
 25  tolbutamide               101766 non-null  object
 26  miglitol                  101766 non-null  object
 27  insulin                   101766 non-null  object
 28  glyburide-metformin       101766 non-null  object
 29  glipizide-metformin       101766 non-null  object
 30  glimepiride-pioglitazone  101766 non-null  object
 31  diabetesMed               101766 non-null  object
 32  readmitted                101766 non-null  object
dtypes: float64(1), int64(11), object(21)
memory usage: 26.4+ MB
```

[9]:

```python
# Change data type

df['encounter_id'] = df['encounter_id'].astype(str)
df['patient_nbr'] = df['patient_nbr'].astype(str)
df['admission_type_id'] = df['admission_type_id'].astype(str)
df['discharge_disposition_id'] = df['discharge_disposition_id'].astype(str)
df['admission_source_id'] = df['admission_source_id'].astype(str)
```

```python
df.info()

# vaccines['series_complete_pop_pct'] = pd.
 ↪to_numeric(vaccines['series_complete_pop_pct']).astype(int)
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 101766 entries, 0 to 101765
Data columns (total 33 columns):
 #   Column                    Non-Null Count   Dtype
---  ------                    --------------   -----
 0   encounter_id              101766 non-null  object
 1   patient_nbr               101766 non-null  object
 2   race                      101766 non-null  object
 3   gender                    101766 non-null  object
 4   age                       101765 non-null  object
 5   weight                    101766 non-null  object
 6   admission_type_id         101766 non-null  object
 7   discharge_disposition_id  101766 non-null  object
 8   admission_source_id       101766 non-null  object
 9   time_in_hospital          101766 non-null  int64
 10  payer_code                101766 non-null  object
 11  medical_specialty         101766 non-null  object
 12  num_lab_procedures        101766 non-null  int64
 13  num_procedures            101766 non-null  int64
 14  num_medications           101757 non-null  float64
 15  number_outpatient         101766 non-null  int64
 16  number_emergency          101766 non-null  int64
 17  number_inpatient          101766 non-null  int64
 18  diag_1                    101766 non-null  object
 19  max_glu_serum             101766 non-null  object
 20  A1Cresult                 101766 non-null  object
 21  metformin                 101766 non-null  object
 22  glimepiride               101766 non-null  object
 23  glipizide                 101766 non-null  object
 24  glyburide                 101766 non-null  object
 25  tolbutamide               101766 non-null  object
 26  miglitol                  101766 non-null  object
 27  insulin                   101766 non-null  object
 28  glyburide-metformin       101766 non-null  object
 29  glipizide-metformin       101766 non-null  object
 30  glimepiride-pioglitazone  101766 non-null  object
 31  diabetesMed               101766 non-null  object
 32  readmitted                101766 non-null  object
dtypes: float64(1), int64(6), object(26)
memory usage: 26.4+ MB
```

2.0.3 Change column names

[10]:

```
# Rename a few columns

short_names = {'admission_type_id':'admin_type', # creating a dict of the names␣
 ↪to be changed
               'discharge_disposition_id':'discharge_dispo',
               'admission_source_id':'admin_source',
               'num_lab_procedures':'lab_procedures',
               'num_procedures':'procedures'}

df.rename(columns=short_names, inplace=True) # passing the dict to the rename␣
 ↪method
                                             # inplace=True
df.info()

######## Appendix A has an explanation of 'inplace'
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 101766 entries, 0 to 101765
Data columns (total 33 columns):
 #   Column                    Non-Null Count   Dtype
---  ------                    --------------   -----
 0   encounter_id              101766 non-null  object
 1   patient_nbr               101766 non-null  object
 2   race                      101766 non-null  object
 3   gender                    101766 non-null  object
 4   age                       101765 non-null  object
 5   weight                    101766 non-null  object
 6   admin_type                101766 non-null  object
 7   discharge_dispo           101766 non-null  object
 8   admin_source              101766 non-null  object
 9   time_in_hospital          101766 non-null  int64
 10  payer_code                101766 non-null  object
 11  medical_specialty         101766 non-null  object
 12  lab_procedures            101766 non-null  int64
 13  procedures                101766 non-null  int64
 14  num_medications           101757 non-null  float64
 15  number_outpatient         101766 non-null  int64
 16  number_emergency          101766 non-null  int64
 17  number_inpatient          101766 non-null  int64
 18  diag_1                    101766 non-null  object
 19  max_glu_serum             101766 non-null  object
 20  A1Cresult                 101766 non-null  object
 21  metformin                 101766 non-null  object
 22  glimepiride               101766 non-null  object
 23  glipizide                 101766 non-null  object
 24  glyburide                 101766 non-null  object
 25  tolbutamide               101766 non-null  object
 26  miglitol                  101766 non-null  object
 27  insulin                   101766 non-null  object
 28  glyburide-metformin       101766 non-null  object
 29  glipizide-metformin       101766 non-null  object
 30  glimepiride-pioglitazone  101766 non-null  object
 31  diabetesMed               101766 non-null  object
 32  readmitted                101766 non-null  object
dtypes: float64(1), int64(6), object(26)
memory usage: 26.4+ MB
```

2.0.4 Manage missing data

[11]:

```
# Just listing the columns and how many rows
# for each have a missing value.

df.isnull().sum()
```

[11]:

```
encounter_id            0
patient_nbr             0
race                    0
gender                  0
age                     1
weight                  0
admin_type              0
discharge_dispo         0
admin_source            0
time_in_hospital        0
payer_code              0
medical_specialty       0
lab_procedures          0
procedures              0
num_medications         9
number_outpatient       0
number_emergency        0
number_inpatient        0
diag_1                  0
max_glu_serum           0
A1Cresult               0
metformin               0
glimepiride             0
glipizide               0
glyburide               0
tolbutamide             0
miglitol                0
insulin                 0
glyburide-metformin     0
```

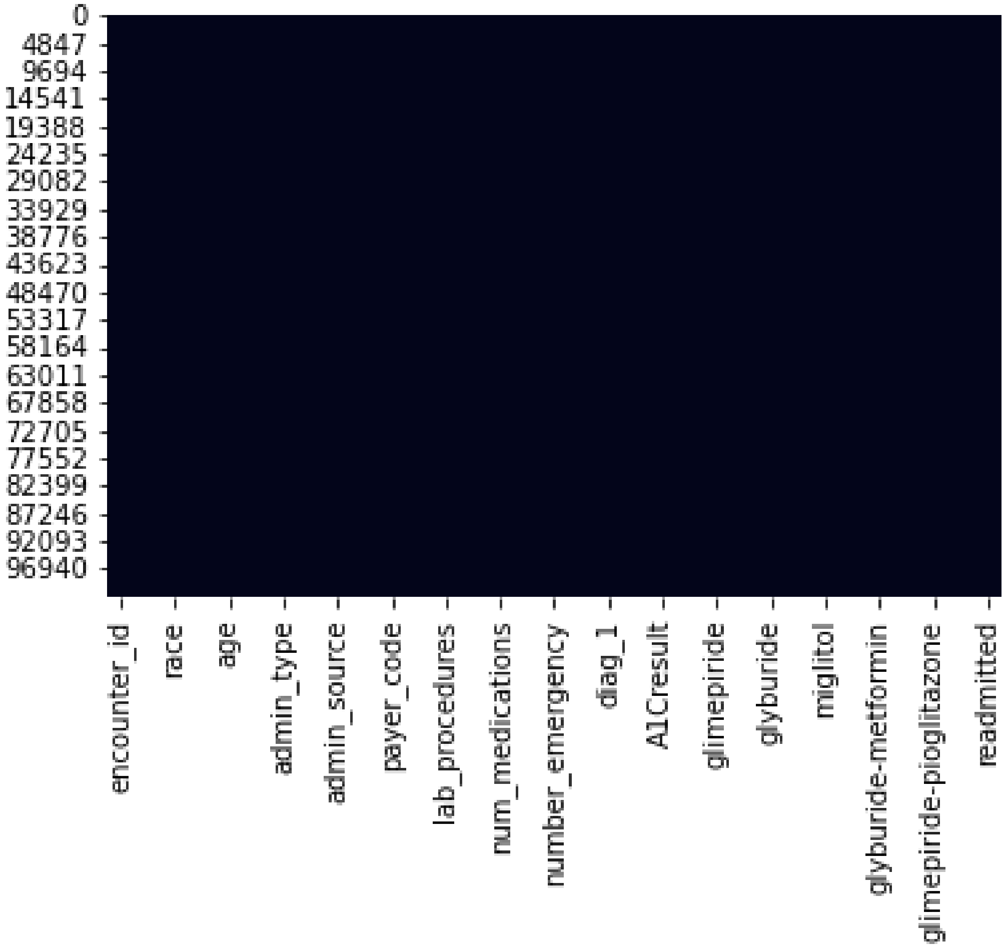
```
glipizide-metformin         0
glimepiride-pioglitazone    0
diabetesMed                 0
readmitted                  0
dtype: int64
```

[12]:
```
df_null = df.isna().mean().round(4) * 100

df_null.sort_values(ascending=False).head()
```

[12]:
```
num_medications    0.01
encounter_id       0.00
glyburide          0.00
max_glu_serum      0.00
A1Cresult          0.00
dtype: float64
```

[13]:
```
# Plotting missing values

sns.heatmap(df.isnull(), cbar=False)
```

[13]:
```
<AxesSubplot:>
```



2.0.5 Imputing missing values

[14]:

```
a = df['num_medications'].describe()
b = df['num_medications'].median()
c = df['num_medications'].mode()
print(a)
print()
print(b)
print()
print(c)
```

```
count    101757.000000
mean         16.021964
std           8.127864
min           1.000000
25%          10.000000
50%          15.000000
```

```
75%           20.000000
max           81.000000
Name: num_medications, dtype: float64

15.0

0    13.0
Name: num_medications, dtype: float64
```

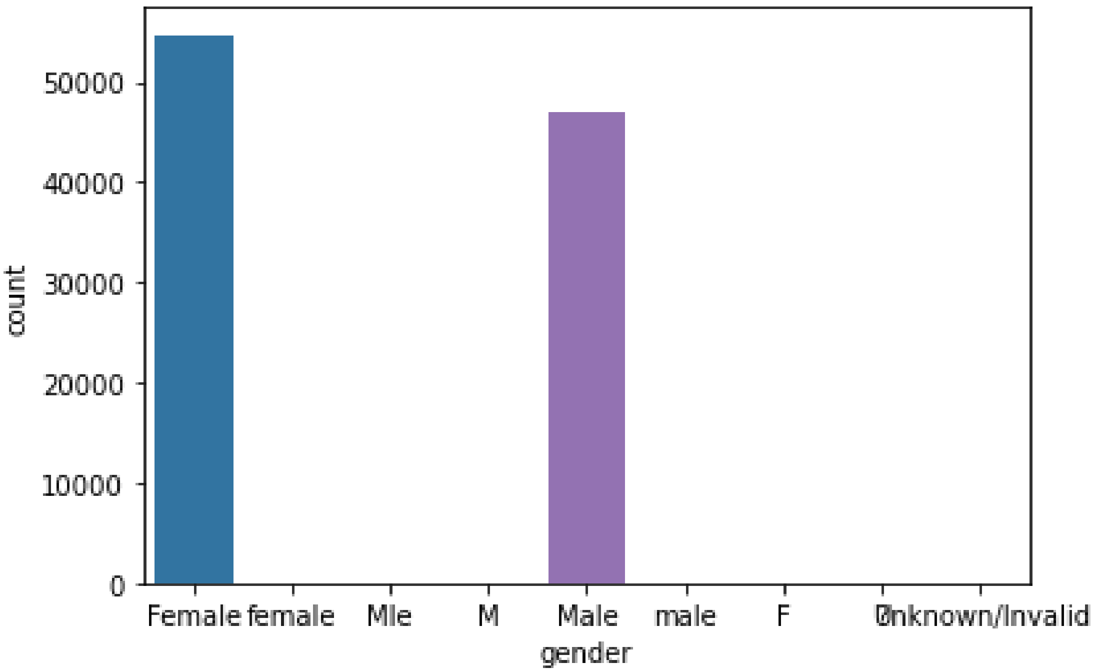
[15]:
```
# Fill missing values of num_medications with the average of num_medications␣
 ↪(mean)

#df[ 'num_medications' ] = df.num_medications.fillna( df.num_medications.mean()␣
 ↪)

df.num_medications.fillna( df.num_medications.mean(),inplace=True )

df_null = df.isna().mean().round(4) * 100
df_null.sort_values(ascending=False).head()

# Can be filled with an arbitrary number
# df.num_medications.fillna( 101,inplace=True )

# backward, forward ->  df.fillna(method='bfill') , df.fillna(method='ffill')
```

[15]:
```
encounter_id                0.0
number_inpatient            0.0
diabetesMed                 0.0
glimepiride-pioglitazone    0.0
glipizide-metformin         0.0
dtype: float64
```

2.1 Check categorical data

[16]:
```
sns.countplot(x='gender', data=df)
```

[16]:
```
<AxesSubplot:xlabel='gender', ylabel='count'>
```



```
[17]: df['gender'].nunique()
```

```
[17]: 9
```

```
[18]: df['gender'].unique()
```

```
[18]: array(['Female', 'female', 'Mle', 'M', 'Male', 'male', 'F', '?',
              'Unknown/Invalid'], dtype=object)
```

```
[19]: df['gender'].value_counts()
```

```
[19]: Female             54706
      Male               47051
      Unknown/Invalid        3
      female                 1
      Mle                    1
      M                      1
      male                   1
      F                      1
      ?                      1
      Name: gender, dtype: int64
```

```
[20]: df.loc[df.gender == 'M','gender']='Male'
      df.head()
```

[20]:

```
  encounter_id patient_nbr             race  gender      age weight  \
0      2278392     8222157        Caucasian  Female      xyz      ?
1       149190    55629189        Caucasian  Female      NaN      ?
2        64410    86047875  AfricanAmerican  female  [20-30)      ?
3       500364    82442376        Caucasian     Mle  [30-40)      ?
4        16680    42519267        Caucasian    Male  [40-50)      ?

  admin_type discharge_dispo admin_source  time_in_hospital  … glipizide  \
0          6              25            1                 1  …        No
1          1               1            7                 3  …        No
2          1               1            7                 2  …    Steady
3          1               1            7                 2  …        No
4          1               1            7                 1  …    Steady

  glyburide  tolbutamide  miglitol  insulin  glyburide-metformin  \
0        No           No        No       No                   No
1        No           No        No       Up                   No
2        No           No        No       No                   No
3        No           No        No       Up                   No
4        No           No        No   Steady                   No

   glipizide-metformin  glimepiride-pioglitazone diabetesMed readmitted
0                   No                        No          No         NO
1                   No                        No         Yes        >30
2                   No                        No         Yes         NO
3                   No                        No         Yes         NO
4                   No                        No         Yes         NO

[5 rows x 33 columns]
```

[21]:

```
# Change/Fix some of the data values

df['gender'] = df['gender'].replace({'M':'Male', 'Mle':'Male', 'F':'Female'})
df.head()
```

[21]:

```
  encounter_id patient_nbr             race  gender      age weight  \
0      2278392     8222157        Caucasian  Female      xyz      ?
1       149190    55629189        Caucasian  Female      NaN      ?
2        64410    86047875  AfricanAmerican  female  [20-30)      ?
3       500364    82442376        Caucasian    Male  [30-40)      ?
4        16680    42519267        Caucasian    Male  [40-50)      ?

  admin_type discharge_dispo admin_source  time_in_hospital  … glipizide  \
0          6              25            1                 1  …        No
1          1               1            7                 3  …        No
2          1               1            7                 2  …    Steady
3          1               1            7                 2  …        No
```

```
4           1               1            7                 1  ...    Steady

  glyburide  tolbutamide  miglitol  insulin  glyburide-metformin  \
0        No           No        No       No                   No
1        No           No        No       Up                   No
2        No           No        No       No                   No
3        No           No        No       Up                   No
4        No           No        No   Steady                   No

   glipizide-metformin  glimepiride-pioglitazone diabetesMed readmitted
0                   No                        No          No         NO
1                   No                        No         Yes        >30
2                   No                        No         Yes         NO
3                   No                        No         Yes         NO
4                   No                        No         Yes         NO

[5 rows x 33 columns]
```

[22]:

```
# Inconsistent capitalization
# Apply a function along an axis of the DataFrame.

df['gender'] = df['gender'].apply(lambda x:x.lower())
df.head()
```

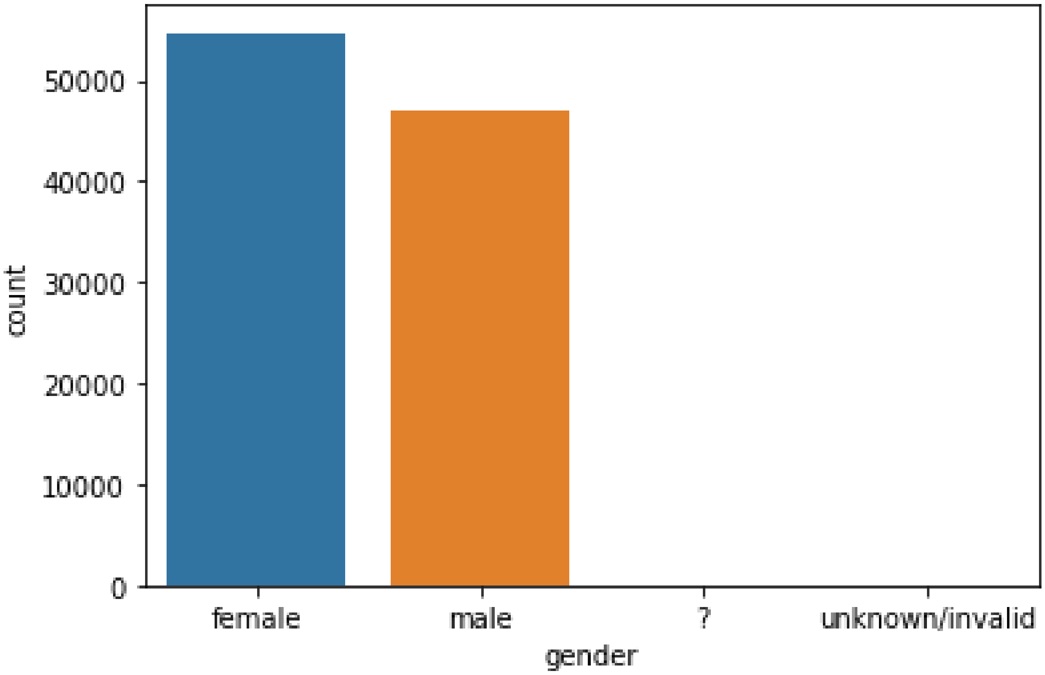
[22]:

```
  encounter_id patient_nbr             race  gender      age weight   \
0      2278392     8222157        Caucasian  female      xyz      ?
1       149190    55629189        Caucasian  female      NaN      ?
2        64410    86047875  AfricanAmerican  female  [20-30)      ?
3       500364    82442376        Caucasian    male  [30-40)      ?
4        16680    42519267        Caucasian    male  [40-50)      ?

  admin_type discharge_dispo admin_source  time_in_hospital  ... glipizide  \
0          6              25            1                 1  ...        No
1          1               1            7                 3  ...        No
2          1               1            7                 2  ...    Steady
3          1               1            7                 2  ...        No
4          1               1            7                 1  ...    Steady

  glyburide  tolbutamide  miglitol  insulin  glyburide-metformin  \
0        No           No        No       No                   No
1        No           No        No       Up                   No
2        No           No        No       No                   No
3        No           No        No       Up                   No
4        No           No        No   Steady                   No

   glipizide-metformin  glimepiride-pioglitazone diabetesMed readmitted
0                   No                        No          No         NO
```

```
1                    No                         No          Yes         >30
2                    No                         No          Yes          NO
3                    No                         No          Yes          NO
4                    No                         No          Yes          NO

[5 rows x 33 columns]
```

[23]:
```
sns.countplot(x='gender', data=df)
```

[23]:
```
<AxesSubplot:xlabel='gender', ylabel='count'>
```



[24]:
```
x = df.loc[df.gender == 'unknown/invalid','gender']
y = df.loc[df.gender == '?','gender']
print(x)
print(y)
```

```
30506    unknown/invalid
75551    unknown/invalid
82573    unknown/invalid
Name: gender, dtype: object
11    ?
Name: gender, dtype: object
```

[25]:
```
df.iloc[11]
```

```
[25]: encounter_id                           36900
      patient_nbr                         77391171
      race                         AfricanAmerican
      gender                                     ?
      age                                  [60-70)
      weight                                     ?
      admin_type                                 2
      discharge_dispo                            1
      admin_source                               4
      time_in_hospital                           7
      payer_code                                 ?
      medical_specialty                          ?
      lab_procedures                            62
      procedures                                 0
      num_medications                         11.0
      number_outpatient                          0
      number_emergency                           0
      number_inpatient                           0
      diag_1                                   157
      max_glu_serum                           None
      A1Cresult                               None
      metformin                                 No
      glimepiride                               No
      glipizide                                 No
      glyburide                                 Up
      tolbutamide                               No
      miglitol                                  No
      insulin                               Steady
      glyburide-metformin                       No
      glipizide-metformin                       No
      glimepiride-pioglitazone                  No
      diabetesMed                              Yes
      readmitted                               <30
      Name: 11, dtype: object
```

```
[26]: df['gender'] = df['gender'].replace({'?':'male', 'unknown/invalid':'male'})
      df.head()
```

```
[26]:   encounter_id patient_nbr             race  gender      age weight  \
      0      2278392     8222157        Caucasian  female      xyz      ?
      1       149190    55629189        Caucasian  female      NaN      ?
      2        64410    86047875  AfricanAmerican  female  [20-30)      ?
      3       500364    82442376        Caucasian    male  [30-40)      ?
      4        16680    42519267        Caucasian    male  [40-50)      ?

        admin_type discharge_dispo admin_source  time_in_hospital  … glipizide  \
      0          6              25            1                 1  …        No
```

```
1          1               1            7                 3  …        No
2          1               1            7                 2  …    Steady
3          1               1            7                 2  …        No
4          1               1            7                 1  …    Steady

  glyburide  tolbutamide  miglitol  insulin  glyburide-metformin  \
0        No           No        No       No                   No
1        No           No        No       Up                   No
2        No           No        No       No                   No
3        No           No        No       Up                   No
4        No           No        No   Steady                   No

   glipizide-metformin  glimepiride-pioglitazone diabetesMed readmitted
0                   No                        No          No         NO
1                   No                        No         Yes        >30
2                   No                        No         Yes         NO
3                   No                        No         Yes         NO
4                   No                        No         Yes         NO

[5 rows x 33 columns]
```

[27]:
```
# Change a value for an entire column
#df.loc[:,'discharge_dispo'] = 99
#df.loc[64410] = 99   # Change a value for an entire row
#df.head()
```

2.1.1 Using visuals to get a sense of the data

[28]:
```
df.info()
```

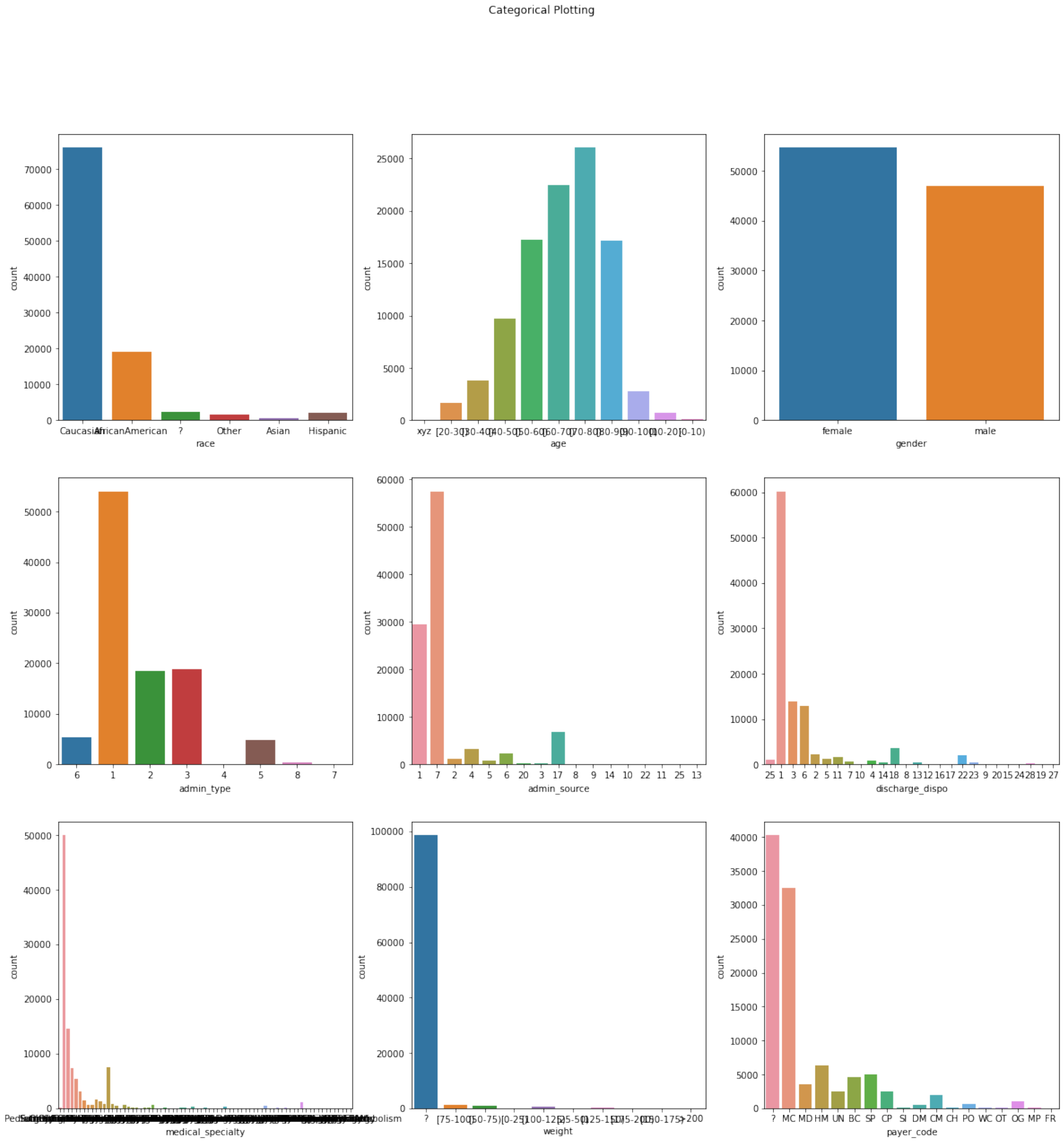
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 101766 entries, 0 to 101765
Data columns (total 33 columns):
 #   Column                    Non-Null Count   Dtype
---  ------                    --------------   -----
 0   encounter_id              101766 non-null  object
 1   patient_nbr               101766 non-null  object
 2   race                      101766 non-null  object
 3   gender                    101766 non-null  object
 4   age                       101765 non-null  object
 5   weight                    101766 non-null  object
 6   admin_type                101766 non-null  object
 7   discharge_dispo           101766 non-null  object
 8   admin_source              101766 non-null  object
 9   time_in_hospital          101766 non-null  int64
 10  payer_code                101766 non-null  object
 11  medical_specialty         101766 non-null  object
 12  lab_procedures           101766 non-null  int64
 13  procedures               101766 non-null  int64
 14  num_medications          101766 non-null  float64
 15  number_outpatient        101766 non-null  int64
 16  number_emergency         101766 non-null  int64
 17  number_inpatient         101766 non-null  int64
 18  diag_1                   101766 non-null  object
 19  max_glu_serum            101766 non-null  object
 20  A1Cresult                101766 non-null  object
 21  metformin                101766 non-null  object
 22  glimepiride              101766 non-null  object
 23  glipizide                101766 non-null  object
 24  glyburide                101766 non-null  object
 25  tolbutamide              101766 non-null  object
 26  miglitol                 101766 non-null  object
 27  insulin                  101766 non-null  object
 28  glyburide-metformin      101766 non-null  object
 29  glipizide-metformin      101766 non-null  object
 30  glimepiride-pioglitazone  101766 non-null  object
 31  diabetesMed              101766 non-null  object
 32  readmitted               101766 non-null  object
dtypes: float64(1), int64(6), object(26)
memory usage: 26.4+ MB
```

Categorical data

[29]:

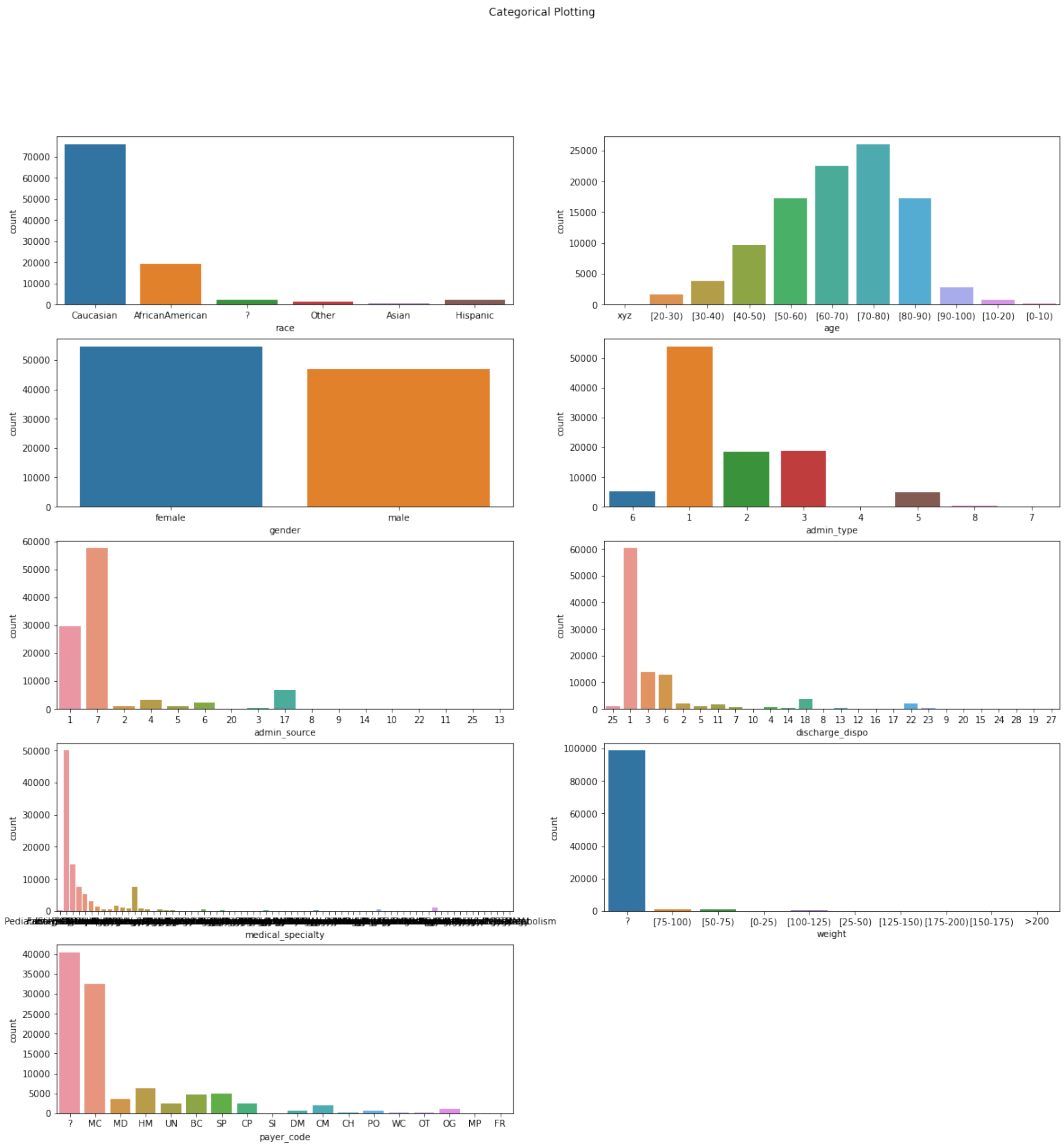
```python
# Create a bar chart for each categorical variables to see the distribution of the data
plt.figure(figsize = (20,20))
plt.subplot(3,3,1)
sns.countplot(x="race", data=df)
plt.subplot(3,3,2)
sns.countplot(x="age", data=df)
plt.subplot(3,3,3)
sns.countplot(x="gender", data=df)
plt.subplot(3,3,4)
sns.countplot(x="admin_type", data=df)
plt.subplot(3,3,5)
sns.countplot(x="admin_source", data=df)
plt.subplot(3,3,6)
sns.countplot(x="discharge_dispo", data=df)
plt.subplot(3,3,7)
sns.countplot(x="medical_specialty", data=df)
plt.subplot(3,3,8)
sns.countplot(x="weight", data=df)
plt.subplot(3,3,9)
sns.countplot(x="payer_code", data=df)
```

```
plt.suptitle('Categorical Plotting')
plt.show()
```



[30]:
```
# Create a bar chart for each categorical variables to see the distribution of
↪the data
plt.figure(figsize = (20,20))
plt.subplot(521)
sns.countplot(x="race", data=df)
plt.subplot(522)
sns.countplot(x="age", data=df)
```

```
plt.subplot(523)
sns.countplot(x="gender", data=df)
plt.subplot(524)
sns.countplot(x="admin_type", data=df)
plt.subplot(525)
sns.countplot(x="admin_source", data=df)
plt.subplot(526)
sns.countplot(x="discharge_dispo", data=df)
plt.subplot(527)
sns.countplot(x="medical_specialty", data=df)
plt.subplot(528)
sns.countplot(x="weight", data=df)
plt.subplot(529)
sns.countplot(x="payer_code", data=df)

plt.suptitle('Categorical Plotting')
plt.show()
```



2.1.2 Examine categorical data a little more closely

[31]:
```
for column in df.columns:         # df.columns is a data frame attribute
    print(f"{column}: Number of unique values {df[column].nunique()}")
    print("==========================================================")



# f means Formatted string literals
```

```
encounter_id: Number of unique values 101766
==========================================================
patient_nbr: Number of unique values 71518
==========================================================
race: Number of unique values 6
==========================================================
gender: Number of unique values 2
==========================================================
age: Number of unique values 11
==========================================================
weight: Number of unique values 10
==========================================================
admin_type: Number of unique values 8
==========================================================
discharge_dispo: Number of unique values 26
==========================================================
admin_source: Number of unique values 17
==========================================================
time_in_hospital: Number of unique values 14
==========================================================
payer_code: Number of unique values 18
==========================================================
medical_specialty: Number of unique values 73
==========================================================
lab_procedures: Number of unique values 118
==========================================================
procedures: Number of unique values 7
==========================================================
num_medications: Number of unique values 76
==========================================================
number_outpatient: Number of unique values 39
==========================================================
number_emergency: Number of unique values 33
==========================================================
number_inpatient: Number of unique values 21
==========================================================
diag_1: Number of unique values 717
==========================================================
max_glu_serum: Number of unique values 4
==========================================================
A1Cresult: Number of unique values 4
==========================================================
metformin: Number of unique values 4
==========================================================
glimepiride: Number of unique values 4
==========================================================
glipizide: Number of unique values 4
==========================================================
glyburide: Number of unique values 4
==========================================================
tolbutamide: Number of unique values 2
==========================================================
miglitol: Number of unique values 4
==========================================================
insulin: Number of unique values 4
==========================================================
glyburide-metformin: Number of unique values 4
==========================================================
glipizide-metformin: Number of unique values 2
==========================================================
glimepiride-pioglitazone: Number of unique values 2
==========================================================
diabetesMed: Number of unique values 2
==========================================================
readmitted: Number of unique values 3
==========================================================
```

[53]:

```python
object_col = []
for column in df.columns:
    if df[column].dtype == object and len(df[column].unique()) <= 30:
        object_col.append(column)
        print(f"{column} : {df[column].unique()}")
        print(df[column].value_counts())
        print("====================================")
```

```
race : ['AfricanAmerican' 'Caucasian' '?' 'Other' 'Asian' 'Hispanic']
Caucasian          76097
AfricanAmerican    19210
?                   2273
Hispanic            2037
Other               1506
Asian                641
Name: race, dtype: int64
====================================
gender : ['female' 'male']
female    54706
male      47058
Name: gender, dtype: int64
====================================
age : ['[20-30)' '[30-40)' '[40-50)' '[50-60)' '[60-70)' '[70-80)' '[80-90)'
 '[90-100)' '[10-20)' '[0-10)']
[70-80)     26068
[60-70)     22483
[50-60)     17256
[80-90)     17197
[40-50)      9685
[30-40)      3775
[90-100)     2793
```

```
[20-30)      1657
[10-20)       690
[0-10)        160
Name: age, dtype: int64
====================================
admin_type : ['1' '2' '3' '6' '4' '5' '8' '7']
1    53989
3    18869
2    18480
6     5290
5     4785
8      320
7       21
4       10
Name: admin_type, dtype: int64
====================================
discharge_dispo : ['1' '3' '6' '2' '5' '11' '7' '25' '10' '4' '14' '18' '8' '13'
'12' '16'
 '17' '22' '23' '9' '20' '15' '24' '28' '19' '27']
1     60233
3     13954
6     12902
18     3691
2      2128
22     1993
11     1642
5      1184
25      988
4       815
7       623
23      412
13      399
14      372
28      139
8       108
15       63
24       48
9        21
17       14
16       11
19        8
10        6
27        5
12        3
20        2
Name: discharge_dispo, dtype: int64
====================================
admin_source : ['7' '2' '4' '1' '5' '6' '20' '3' '17' '8' '9' '14' '10' '22'
'11' '25'
 '13']
7     57493
1     29564
17     6781
4      3187
6      2264
2      1104
5       855
3       187
20      161
9       125
8        16
22       12
10        8
14        2
11        2
25        2
13        1
Name: admin_source, dtype: int64
====================================
max_glu_serum : ['None' '>300' 'Norm' '>200']
None    96418
Norm     2597
>200     1485
>300     1264
Name: max_glu_serum, dtype: int64
====================================
A1Cresult : ['None' '>7' '>8' 'Norm']
None    84746
>8       8216
Norm     4990
>7       3812
Name: A1Cresult, dtype: int64
====================================
metformin : ['No' 'Steady' 'Up' 'Down']
No        81776
Steady    18346
Up         1067
Down        575
Name: metformin, dtype: int64
====================================
glimepiride : ['No' 'Steady' 'Down' 'Up']
No        96573
Steady     4670
Up          327
Down        194
Name: glimepiride, dtype: int64
```

```
====================================
glipizide : ['Steady' 'No' 'Up' 'Down']
No        89078
Steady    11356
Up          770
Down        560
Name: glipizide, dtype: int64
====================================
glyburide : ['No' 'Steady' 'Up' 'Down']
No        91114
Steady     9274
Up          812
Down        564
Name: glyburide, dtype: int64
====================================
tolbutamide : ['No' 'Steady']
No        101741
Steady        23
Name: tolbutamide, dtype: int64
====================================
miglitol : ['No' 'Steady' 'Down' 'Up']
No        101726
Steady        31
Down           5
Up             2
Name: miglitol, dtype: int64
====================================
insulin : ['No' 'Up' 'Steady' 'Down']
No        47382
Steady    30849
Down      12218
Up        11315
Name: insulin, dtype: int64
====================================
diabetesMed : ['Yes' 'No']
Yes    78362
No     23402
Name: diabetesMed, dtype: int64
====================================
readmitted : ['NO' '>30' '<30']
NO     54863
>30    35544
<30    11357
Name: readmitted, dtype: int64
====================================
```

[33]:

```
df['payer_code'].nunique()
```

```
[33]: 18
```

```
[34]: df['payer_code'].value_counts()
```

```
[34]: ?     40256
      MC    32439
      HM     6274
      SP     5007
      BC     4655
      MD     3532
      CP     2533
      UN     2448
      CM     1937
      OG     1033
      PO      592
      DM      549
      CH      146
      WC      135
      OT       95
      MP       79
      SI       55
      FR        1
      Name: payer_code, dtype: int64
```

```
[35]: df['medical_specialty'].nunique()
```

```
[35]: 73
```

```
[36]: df['medical_specialty'].value_counts()
```

```
[36]: ?                                 49949
      InternalMedicine                  14635
      Emergency/Trauma                   7565
      Family/GeneralPractice             7440
      Cardiology                         5352
                                        …
      SportsMedicine                        1
      Speech                                1
      Perinatology                          1
      Neurophysiology                       1
      Pediatrics-InfectiousDiseases         1
      Name: medical_specialty, Length: 73, dtype: int64
```

```
[37]: df['weight'].nunique()
```

```
[37]: 10
```

[38]:

```
df['weight'].value_counts()
```

[38]:

```
?            98569
[75-100)      1336
[50-75)        897
[100-125)      625
[125-150)      145
[25-50)         97
[0-25)          48
[150-175)       35
[175-200)       11
>200             3
Name: weight, dtype: int64
```

2.1.3 Dropping columns and rows

[39]:

```
df.shape
```

[39]:

```
(101766, 33)
```

[40]:

```
# Remove a single column
df = df.drop('payer_code',axis=1)  # Axis=1 means drop the column
df = df.drop('weight',axis=1)

# inplace=True not used so columns still exist. Just not in this instance.
# Fix that.
```

[41]:

```
# Remove multiple columns

# glyburide-metformin
# glipizide-metformin
# glimepiride-pioglitazone

drop_columns = {'medical_specialty','glyburide-metformin','glipizide-metformin',
                'glimepiride-pioglitazone'}
df = df.drop(columns = drop_columns) # inplace=True not used so columns still␣
 ↪exist.
                                     # Just not in this instance.
#df.head()
```

[42]:

```
# Delete by selecting rows not equal to the condition
df = df.loc[df['age']!= 'xyz']
df = df.loc[df.gender != '?']
#df = df.loc[df['gender']!='?']
#df.shape
```

[43]:

```
no_age = df[df['age'].isnull()].index
#no_age
df = df.drop(no_age, axis = 0)    # axis = 0 means drop the row
df.shape
```

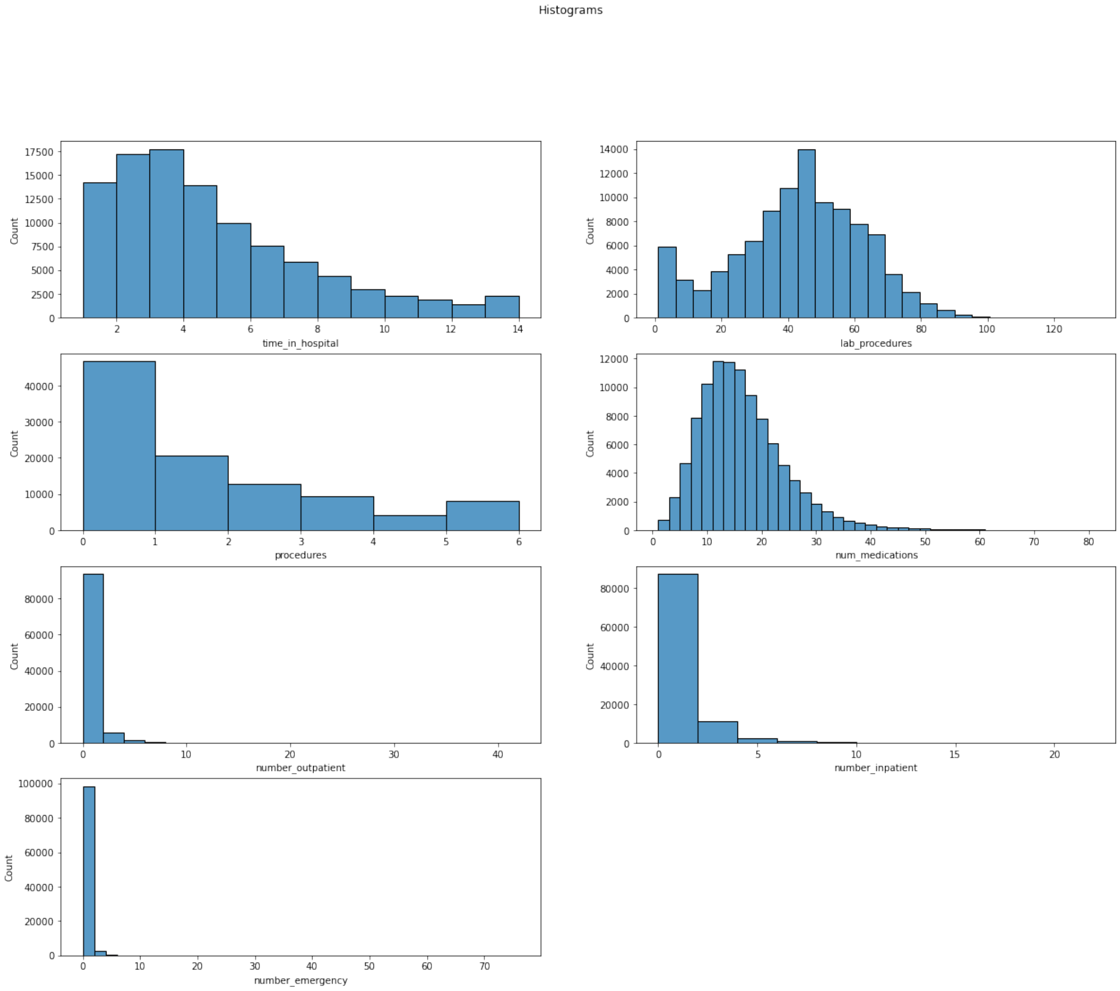
[43]:

```
(101764, 27)
```

Quantitative data

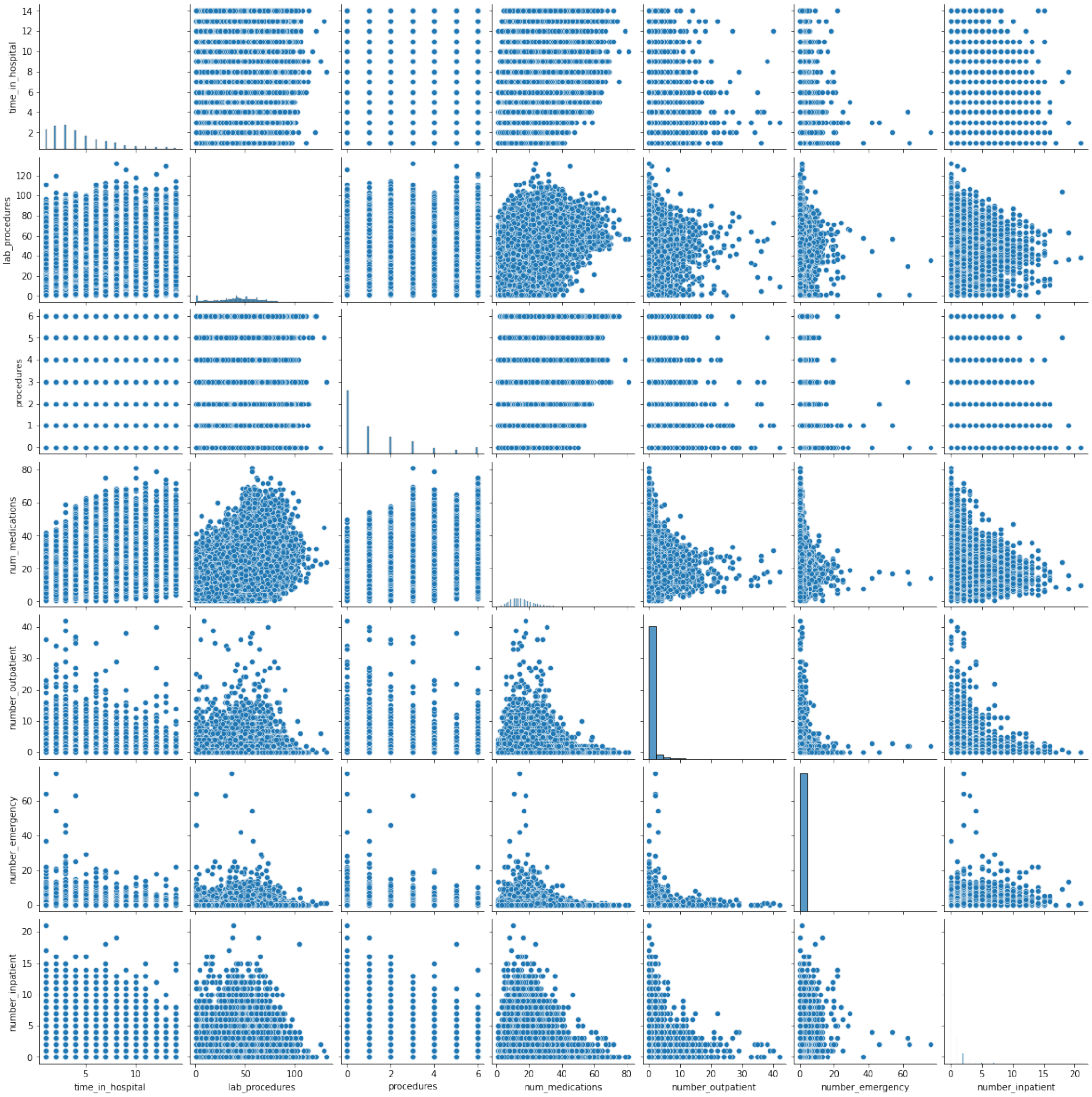
[44]:

```
# Histograms

plt.figure(figsize = (20,20))
plt.subplot(521)
sns.histplot(data=df, x='time_in_hospital', binwidth = 1)
plt.subplot(522)
sns.histplot(data=df, x='lab_procedures', bins=25)
plt.subplot(523)
sns.histplot(data=df, x='procedures', binwidth = 1)
plt.subplot(524)
sns.histplot(data=df, x='num_medications', binwidth = 2)
plt.subplot(525)
sns.histplot(data=df, x='number_outpatient', binwidth = 2)
plt.subplot(526)
sns.histplot(data=df, x='number_inpatient', binwidth = 2)
plt.subplot(527)
sns.histplot(data=df, x='number_emergency', binwidth = 2)


plt.suptitle('Histograms')
plt.show()
```



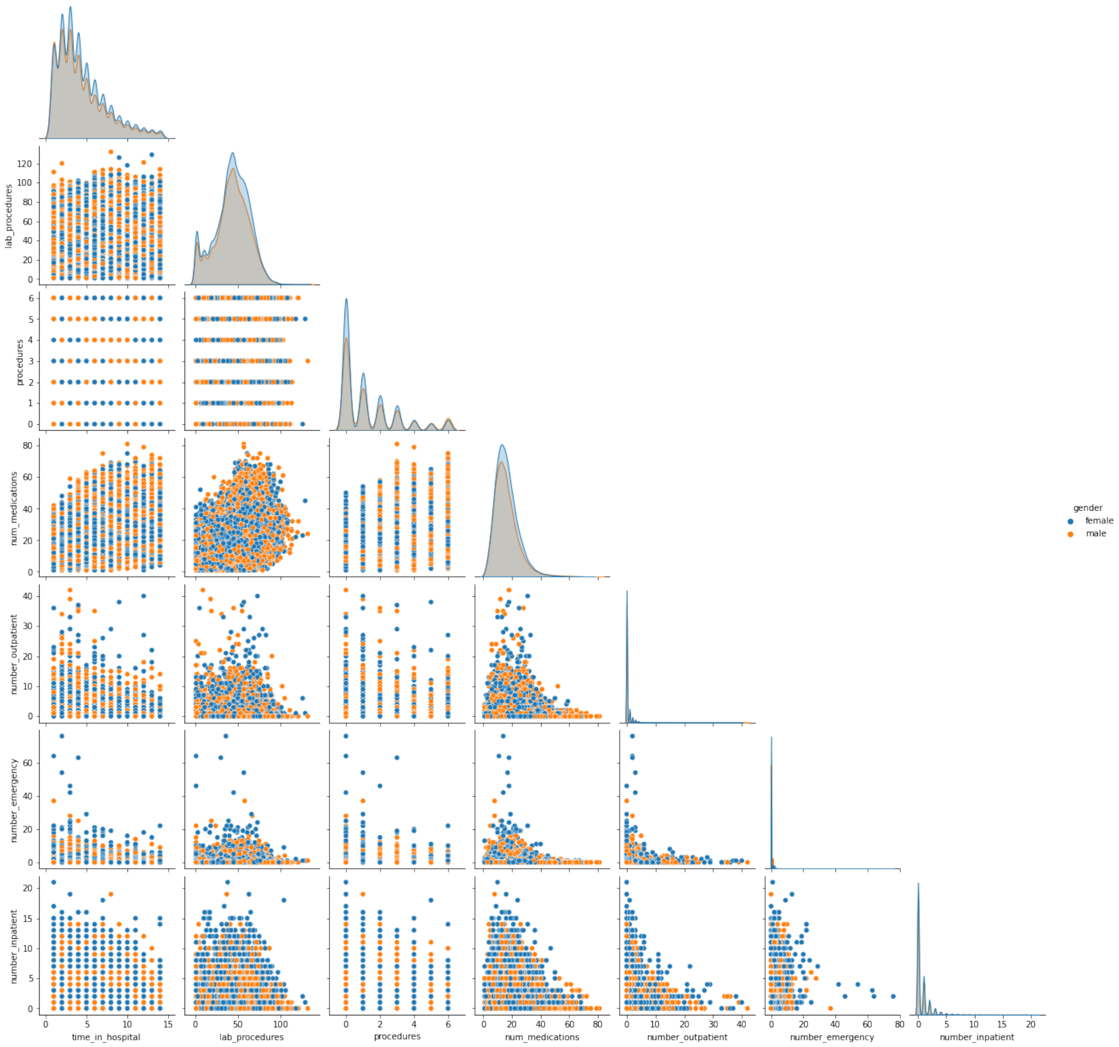
[45]:
```
# Pairplot to see the big picture
sns.pairplot(df)
```

[45]:
```
<seaborn.axisgrid.PairGrid at 0x121f2bee0>
```



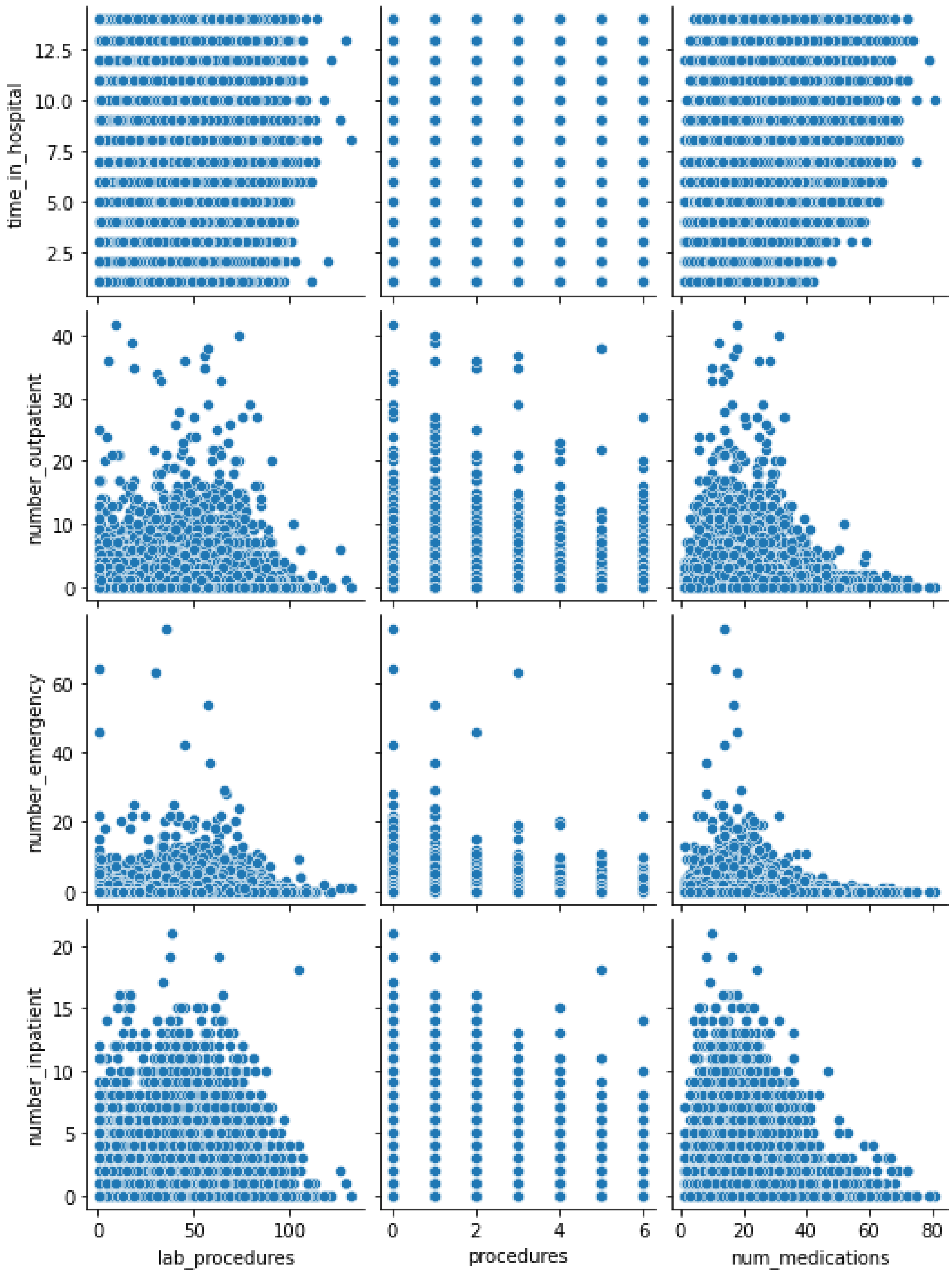
```
[46]: sns.pairplot(df, hue = 'gender', corner = True)
```

```
[46]: <seaborn.axisgrid.PairGrid at 0x122d2d370>
```



```
[47]: sns.pairplot(df,
                   x_vars=['lab_procedures', 'procedures', 'num_medications'],
                  ␣
       ↪y_vars=['time_in_hospital','number_outpatient','number_emergency','number_inpatient'␣
       ↪])
```

```
[47]: <seaborn.axisgrid.PairGrid at 0x123c3dee0>
```



[48]:

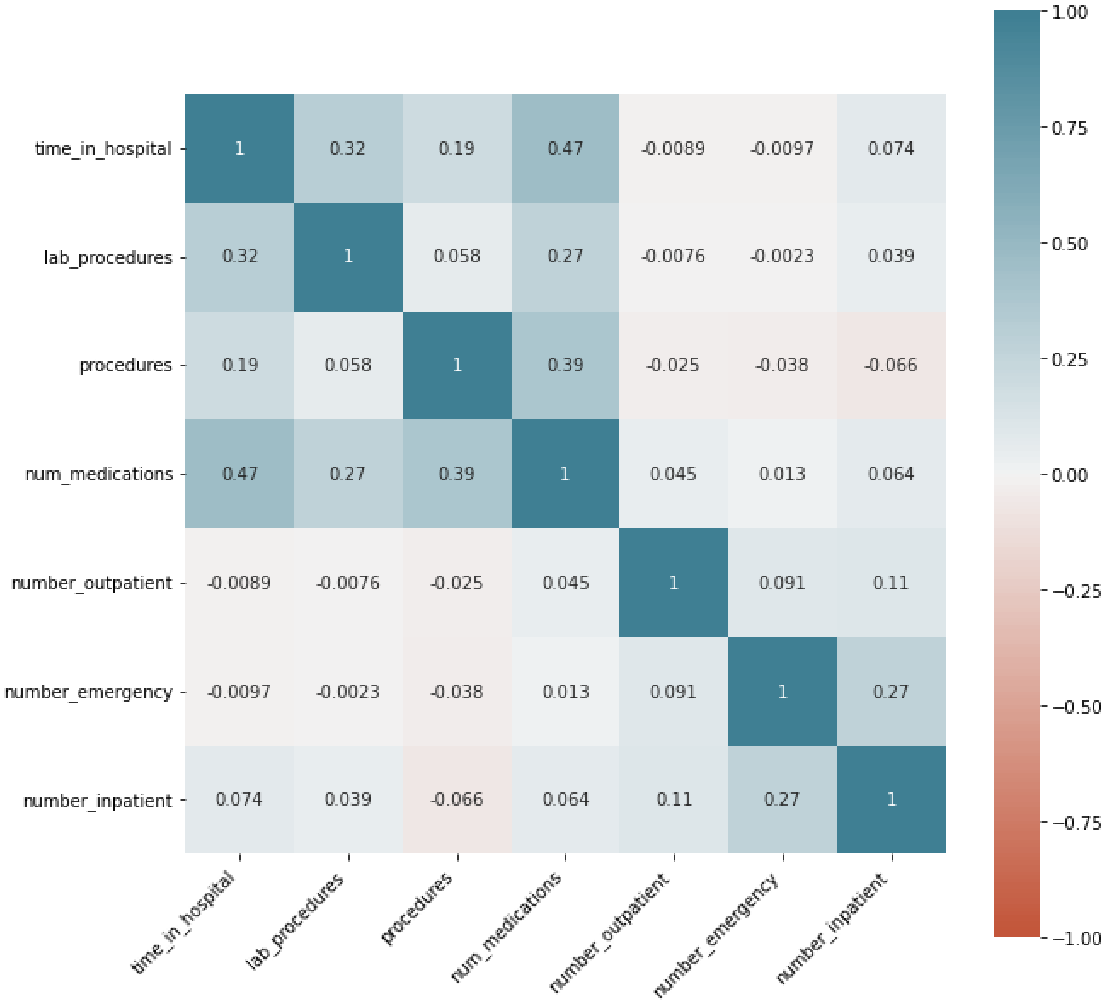
```
# Correlations
df2 = df.corr()
df2
```

[48]:

```
                   time_in_hospital  lab_procedures  procedures  \
time_in_hospital           1.000000        0.318456    0.191462
lab_procedures             0.318456        1.000000    0.058072
procedures                 0.191462        0.058072    1.000000
num_medications            0.466121        0.268152    0.385765
number_outpatient         -0.008921       -0.007600   -0.024823
number_emergency          -0.009684       -0.002278   -0.038183
number_inpatient           0.073615        0.039235   -0.066244

                   num_medications  number_outpatient  number_emergency  \
time_in_hospital          0.466121          -0.008921         -0.009684
lab_procedures            0.268152          -0.007600         -0.002278
procedures                0.385765          -0.024823         -0.038183
num_medications           1.000000           0.045189          0.013175
number_outpatient         0.045189           1.000000          0.091458
number_emergency          0.013175           0.091458          1.000000
number_inpatient          0.064180           0.107335          0.266558

                   number_inpatient
time_in_hospital           0.073615
lab_procedures             0.039235
procedures                -0.066244
num_medications            0.064180
number_outpatient          0.107335
number_emergency           0.266558
number_inpatient           1.000000
```

[49]:

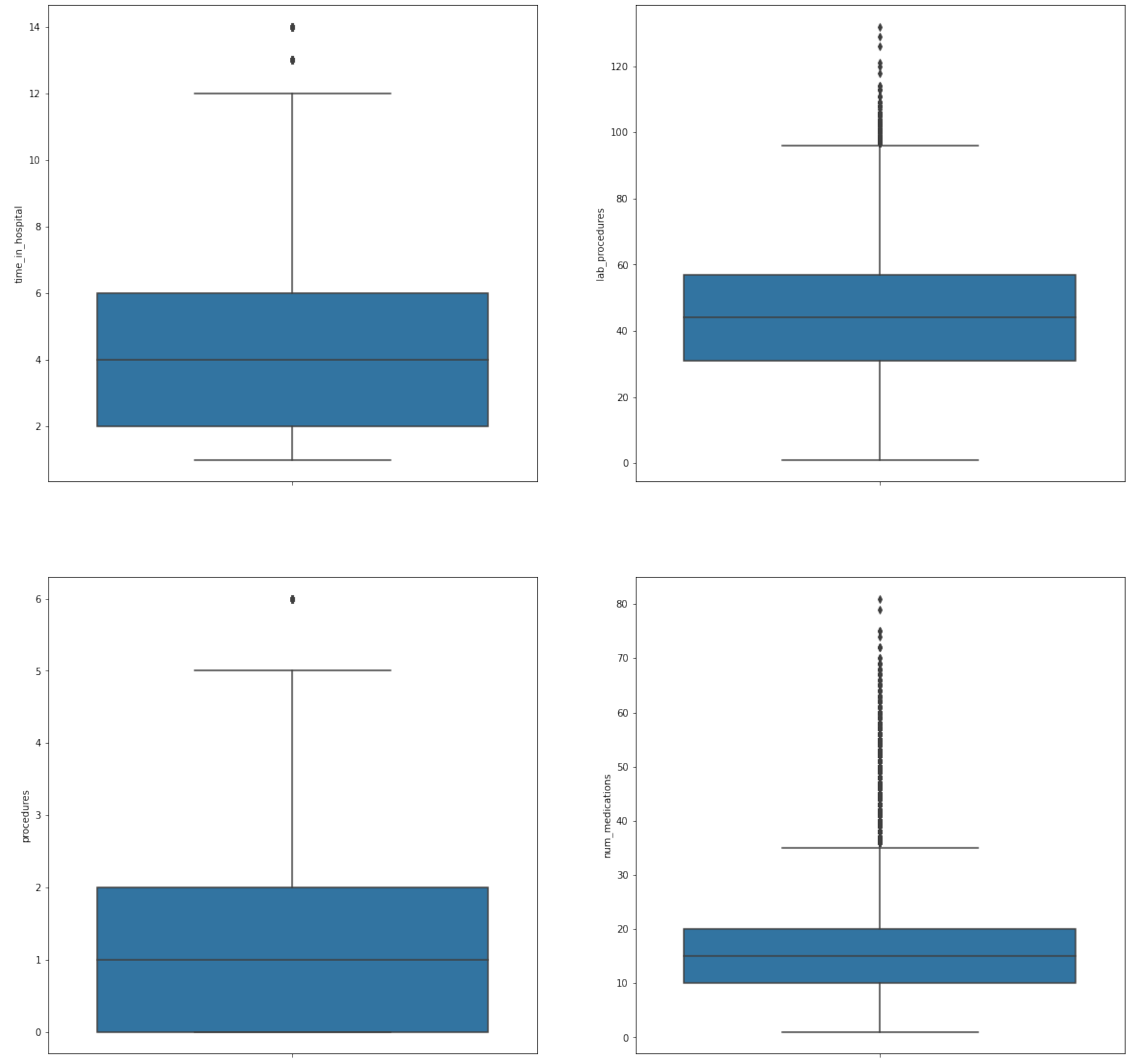
```python
plt.figure(figsize=(10,10))
corr = df2.corr()
ax = sns.heatmap(
    df2,
    vmin=-1, vmax=1, center=0,
    cmap=sns.diverging_palette(20, 220, n=200),
    square=True,
    annot=True, annot_kws={"size":10}
)
ax.set_xticklabels(
    ax.get_xticklabels(),
    rotation=45,
    horizontalalignment='right')
plt.show()
```



	time_in_hospital	lab_procedures	procedures	num_medications	number_outpatient	number_emergency	number_inpatient
time_in_hospital	1	0.32	0.19	0.47	-0.0089	-0.0097	0.074
lab_procedures	0.32	1	0.058	0.27	-0.0076	-0.0023	0.039
procedures	0.19	0.058	1	0.39	-0.025	-0.038	-0.066
num_medications	0.47	0.27	0.39	1	0.045	0.013	0.064
number_outpatient	-0.0089	-0.0076	-0.025	0.045	1	0.091	0.11
number_emergency	-0.0097	-0.0023	-0.038	0.013	0.091	1	0.27
number_inpatient	0.074	0.039	-0.066	0.064	0.11	0.27	1

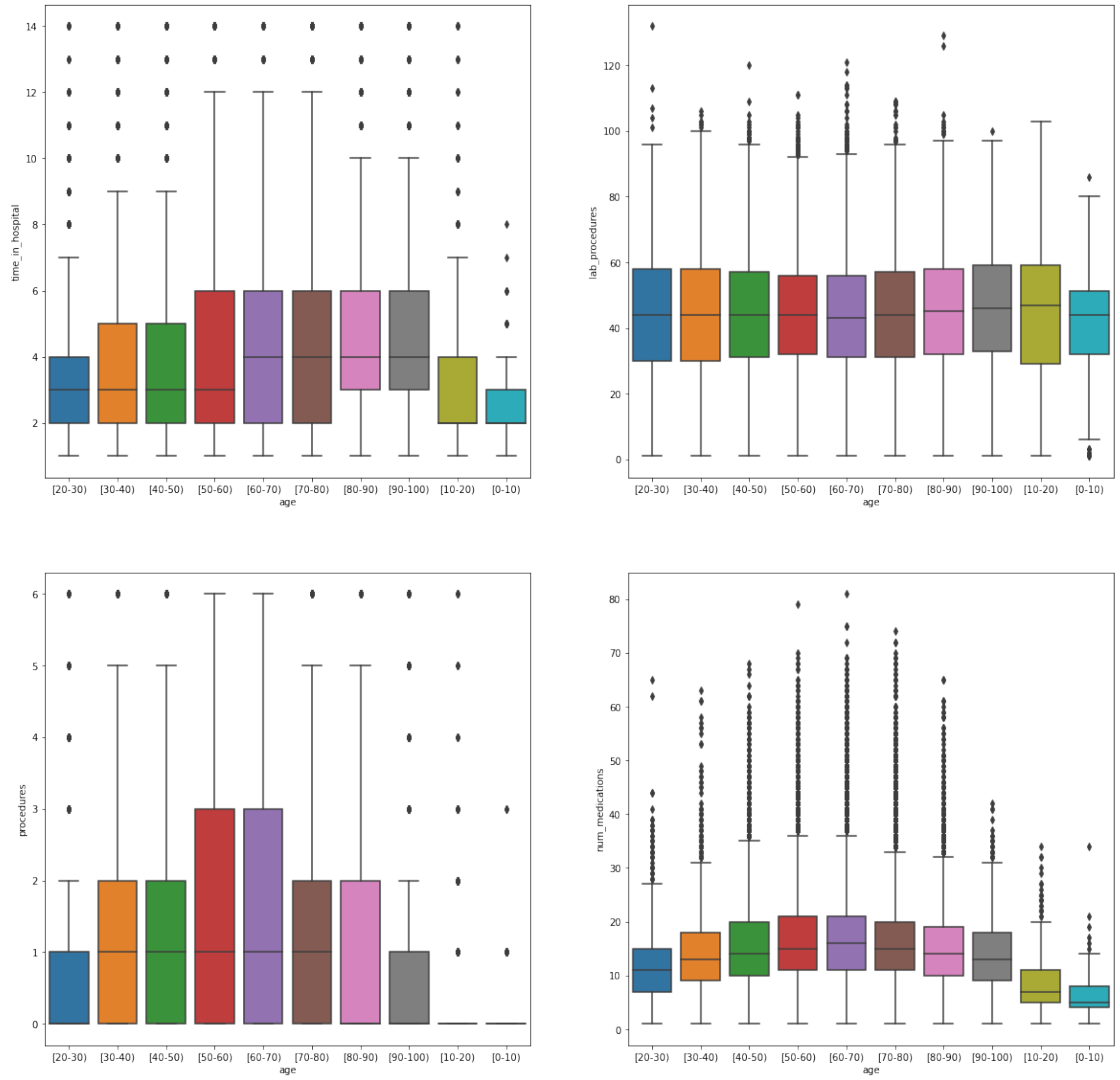
[50]:
```
# Focusing on a few variables

plt.figure(figsize = (20,20))
plt.subplot(221)
sns.boxplot(data=df, y="time_in_hospital")
plt.subplot(222)
sns.boxplot(data=df, y="lab_procedures")
plt.subplot(223)
sns.boxplot(data=df, y="procedures")
plt.subplot(224)
sns.boxplot(data=df, y="num_medications")
```

[50]:
```
<AxesSubplot:ylabel='num_medications'>
```



```
[51]: # Focusing on a few variables

      plt.figure(figsize = (20,20))
      plt.subplot(221)
      sns.boxplot(data=df, x='age', y="time_in_hospital")
      plt.subplot(222)
      sns.boxplot(data=df, x='age', y="lab_procedures")
      plt.subplot(223)
      sns.boxplot(data=df, x='age', y="procedures")
      plt.subplot(224)
      sns.boxplot(data=df, x='age', y="num_medications")
```

```
[51]: <AxesSubplot:xlabel='age', ylabel='num_medications'>
```



2.1.4 Removing outliers

[52]:
```
#outliers
dfoutliers = df[(df['num_medications']>70)]
dfoutliers.shape
#filtering outliers out
#df_movie = df_movie[(df_movie['minute']>43) & (df_movie['minute']<158)]
```

[52]:
```
(8, 27)
```

3 Exercise - 30 minutes

3.0.1 See Beer Notebook - Part 1