HEALTHCARE TRENDS ANALYSIS PROJECT

- SRIJA BASAK

OVERVIEW OF THE GIVEN DATASET

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 55500 entries, 0 to 55499
Data columns (total 20 columns):
 #   Column              Non-Null Count  Dtype         
---  ------              --------------  -----         
 0   Name                55500 non-null  object        
 1   Age                 55500 non-null  int64         
 2   Gender              55500 non-null  object        
 3   Blood Type          55500 non-null  object        
 4   Medical Condition   55500 non-null  object        
 5   Date of Admission   55500 non-null  datetime64[ns]
 6   Doctor              55500 non-null  object        
 7   Hospital            55500 non-null  object        
 8   Insurance Provider  55500 non-null  object        
 9   Billing Amount      55500 non-null  float64       
 10  Room Number         55500 non-null  int64         
 11  Admission Type      55500 non-null  object        
 12  Discharge Date      55500 non-null  datetime64[ns]
 13  Medication          55500 non-null  object        
 14  Test Results        55500 non-null  object        
 15  Days in Hospital    55500 non-null  timedelta64[ns]
 16  Admission Year      55500 non-null  int32         
 17  Admission Month     55500 non-null  int32         
 18  Admission Day       55500 non-null  int32         
 19  Admission Weekday   55500 non-null  int32         
dtypes: datetime64[ns](2), float64(1), int32(4), int64(2), object(10), timedelta64[ns](1)
memory usage: 7.6+ MB
```

DESCRIPTIVE STATISTICS OF EACH COLUMN OF THE DATASET

	Age	Date of Admission	Billing Amount	Room Number	Discharge Date	Days in Hospital	Admission Year	Admission Month	Admission Day	Admission Weekday
count	55500.000000	55500	55500.000000	55500.000000	55500	55500	55500.000000	55500.00000	55500.000000	55500.000000
mean	51.539459	2021-11-01 01:02:22.443243008	25539.316097	301.134829	2021-11-16 13:15:20.821621504	15 days 12:12:58.378378378	2021.334631	6.52845	15.679081	2.998955
min	13.000000	2019-05-08 00:00:00	-2008.492140	101.000000	2019-05-09 00:00:00	1 days 00:00:00	2019.000000	1.00000	1.000000	0.000000
25%	35.000000	2020-07-28 00:00:00	13241.224655	202.000000	2020-08-12 00:00:00	8 days 00:00:00	2020.000000	4.00000	8.000000	1.000000
50%	52.000000	2021-11-01 00:00:00	25538.069380	302.000000	2021-11-17 00:00:00	15 days 00:00:00	2021.000000	7.00000	16.000000	3.000000
75%	68.000000	2023-02-03 00:00:00	37820.508432	401.000000	2023-02-18 00:00:00	23 days 00:00:00	2023.000000	9.25000	23.000000	5.000000
max	89.000000	2024-05-07 00:00:00	52764.276740	500.000000	2024-06-06 00:00:00	30 days 00:00:00	2024.000000	12.00000	31.000000	6.000000
std	19.602454	NaN	14211.454431	115.243069	NaN	8 days 15:49:49.456115772	1.497310	3.43689	8.824412	1.997530

VALUE-COUNTS OF CATEGORICAL COLUMNS

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

```
Gender
Male      27774
Female    27726
Name: count, dtype: int64
```

```
df['Admission Type'].value_counts()
```

```
Admission Type
Elective     18655
Urgent       18576
Emergency    18269
Name: count, dtype: int64
```

```
[ ] df['Blood Type'].value_counts()
```

```
Blood Type
A-     6969
A+     6956
AB+    6947
AB-    6945
B+     6945
B-     6944
O+     6917
O-     6877
Name: count, dtype: int64
```

```
[ ] df['Test Results'].value_counts()
```

```
Test Results
Abnormal        18627
Normal          18517
Inconclusive    18356
Name: count, dtype: int64
```

PRINTING THE NAME OF THE DOCTOR HAVING MOST NO. OF PATIENTS

Printing the name of the doctor having most no of patients

```
[ ]  #Doctor having most no of patients
     doctor_max_admission=df['Doctor'].value_counts().idxmax()
     print("Doctor with most no of patients -" ,doctor_max_admission)
```

```
Doctor with most no of patients - Michael Smith
```

Double-click (or enter) to edit

```
[ ]  df1=df.copy()
```

PRINTING THE DETAILS OF THE OLDEST PATIENT IN THE DATASET

```
[ ]  df1=df.copy()
```

Printing the details of the oldest patient in the dataset

```
[ ]  #oldest patient in the dataset, age, medical condition, admission type and test result
     oldest_patient_age=df1['Age'].max()
     oldest_patient_name=df1[df['Age']==oldest_patient_age]['Name'].iloc[0].upper()
     print(f"Oldest patient in the dataset: {oldest_patient_name}\nAge: {oldest_patient_age}")
     med_oldest_patient_name=df1[df['Age']==oldest_patient_age]['Medical Condition'].iloc[0]
     print(f"Medical Condition: {med_oldest_patient_name}")
     admission_oldest_patient_name=df1[df['Age']==oldest_patient_age]['Admission Type'].iloc[0]
     print(f"Admission Type: {admission_oldest_patient_name}")
     test_oldest_patient_name=df1[df['Age']==oldest_patient_age]['Test Results'].iloc[0]
     print(f"Test Result: {test_oldest_patient_name}")
```

```
Oldest patient in the dataset: DAVID NEWTON
Age: 89
Medical Condition: Arthritis
Admission Type: Elective
Test Result: Inconclusive
```

PRINTING THE DETAILS OF THE YOUNGEST PATIENT IN THE DATASET

Printing the details of the youngest patient in the dataset

```
[ ] #youngest patient in the dataset, age, medical condition, admission type and test result
    youngest_patient_age=df1['Age'].min()
    youngest_patient_name=df1[df['Age']==youngest_patient_age]['Name'].iloc[0].upper()
    print(f"Youngest patient in the dataset: {youngest_patient_name}\nAge: {youngest_patient_age}")
    med_youngest_patient_name=df1[df['Age']==youngest_patient_age]['Medical Condition'].iloc[0]
    print(f"Medical Condition: {med_youngest_patient_name}")
    admission_youngest_patient_name=df1[df['Age']==youngest_patient_age]['Admission Type'].iloc[0]
    print(f"Admission Type: {admission_youngest_patient_name}")
    test_youngest_patient_name=df1[df['Age']==youngest_patient_age]['Test Results'].iloc[0]
    print(f"Test Result: {test_youngest_patient_name}")
```

```
Youngest patient in the dataset: JAMES BASS PHD
Age: 13
Medical Condition: Asthma
Admission Type: Emergency
Test Result: Inconclusive
```

MONTHLY ADMISSIONS YEARWISE

Monthly admissions yearwise

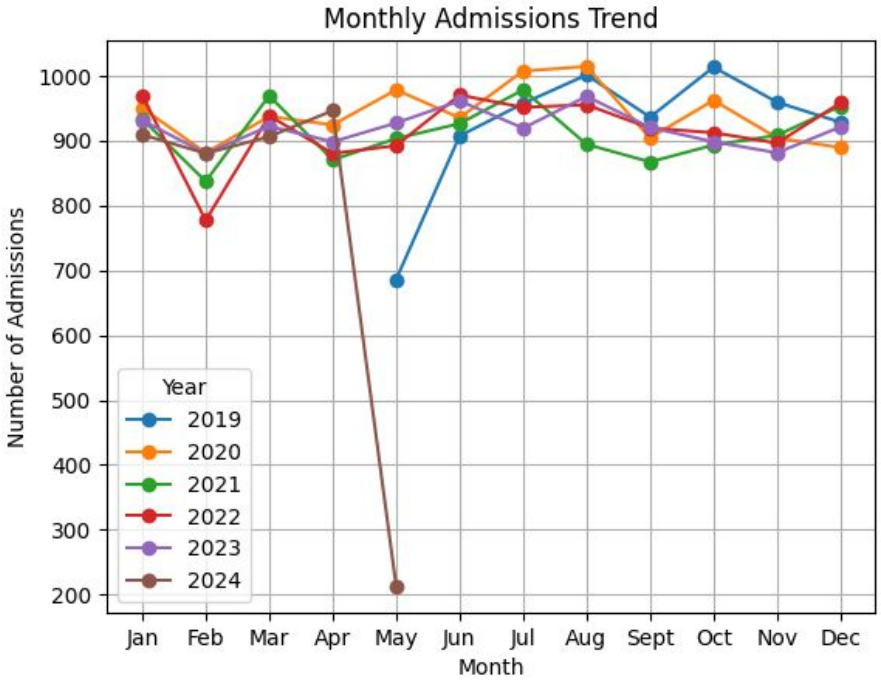
```
[ ] monthly_admissions=df.groupby(['Admission Year','Admission Month']).size().reset_index(name='Counts')

    monthly_admissions_pivot=monthly_admissions.pivot(index='Admission Month',columns='Admission Year',values='Counts')
    monthly_admissions_pivot
```

Admission Year	2019	2020	2021	2022	2023	2024
Admission Month						
1	NaN	950.0	933.0	969.0	931.0	909.0
2	NaN	881.0	837.0	777.0	880.0	880.0
3	NaN	937.0	969.0	938.0	922.0	906.0
4	NaN	924.0	870.0	880.0	898.0	946.0
5	686.0	978.0	903.0	892.0	927.0	213.0
6	907.0	935.0	926.0	970.0	961.0	NaN
7	957.0	1007.0	978.0	951.0	919.0	NaN
8	1001.0	1014.0	894.0	955.0	968.0	NaN
9	936.0	904.0	867.0	919.0	920.0	NaN
10	1013.0	962.0	893.0	912.0	898.0	NaN
11	959.0	904.0	908.0	896.0	881.0	NaN
12	928.0	889.0	953.0	958.0	921.0	NaN

DATA VISUALIZATIONS USING MATPLOTLIB, SEABORN AND PLOTLY

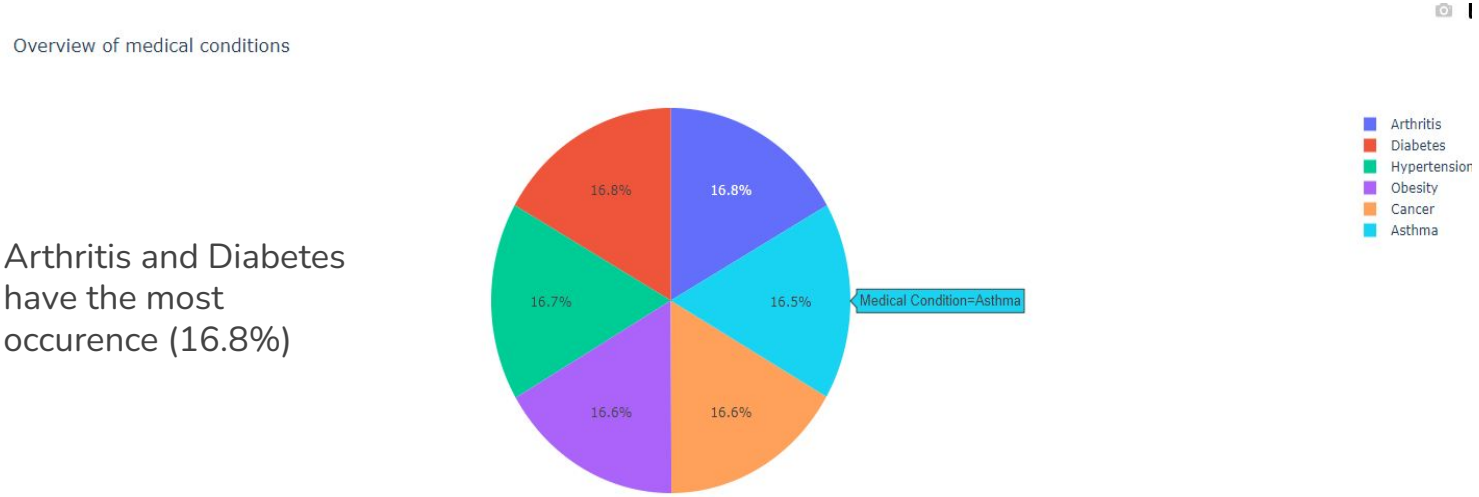
TRENDS OF MONTHLY ADMISSIONS YEARWISE



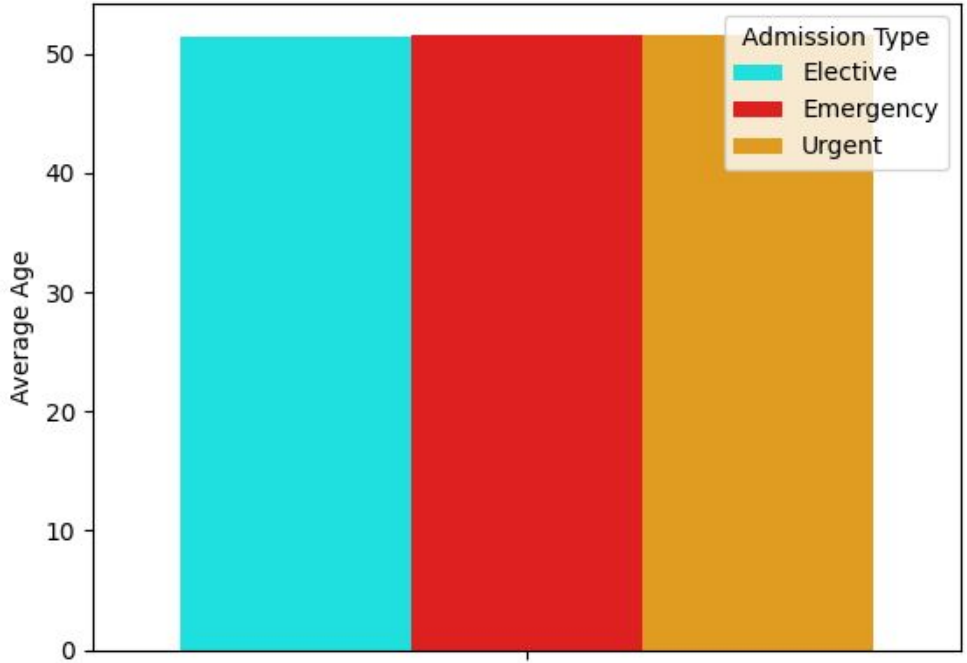
As per the given data, the admissions have been the least in May 2024 and the admissions have been the most in October 2019

PIE CHART FOR PERCENTAGE OF PEOPLE FOR A PARTICULAR MEDICAL CONDITION

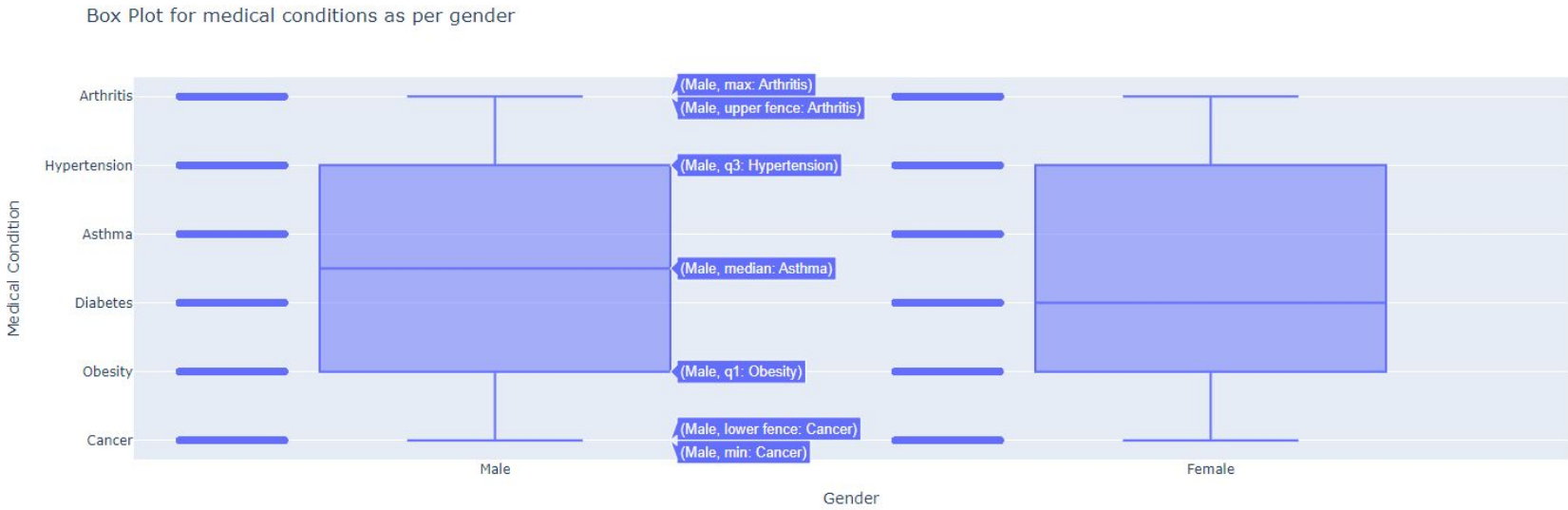
Arthritis and Diabetes have the most occurence (16.8%)



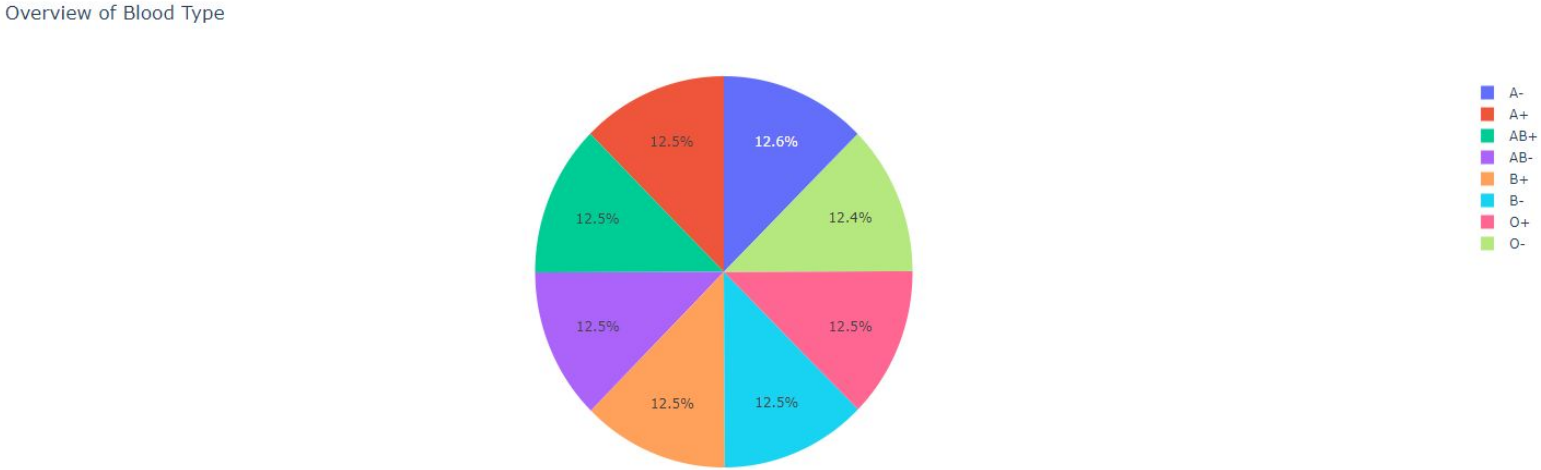
BAR GRAPH FOR AVERAGE AGE FOR EACH ADMISSION TYPE



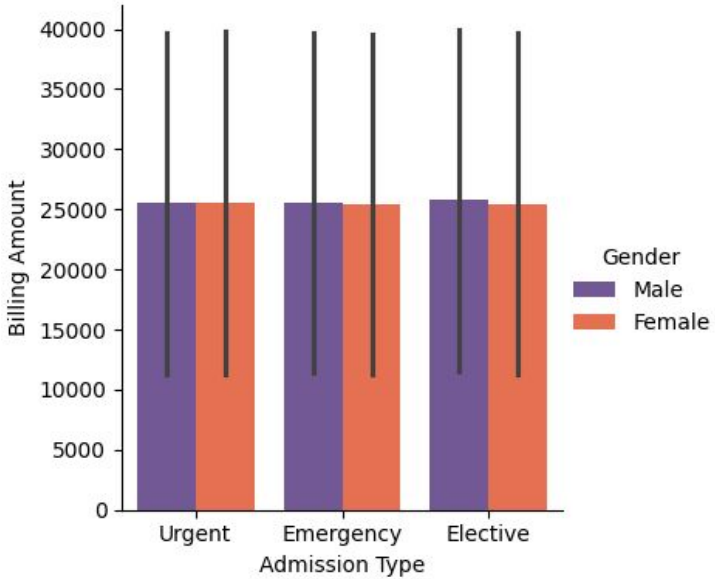
BOX PLOT REPRESENTING STATISTICS OF MEDICAL CONDITIONS AS PER GENDER



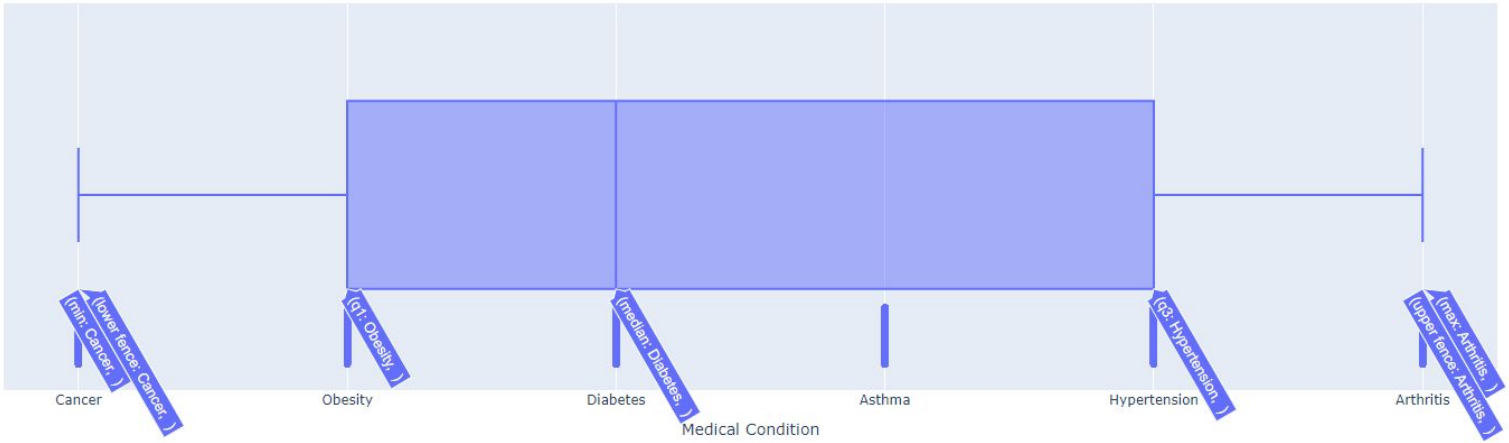
PIE CHART DEPICTING PERCENTAGE OF PEOPLE HAVING EACH BLOOD TYPE



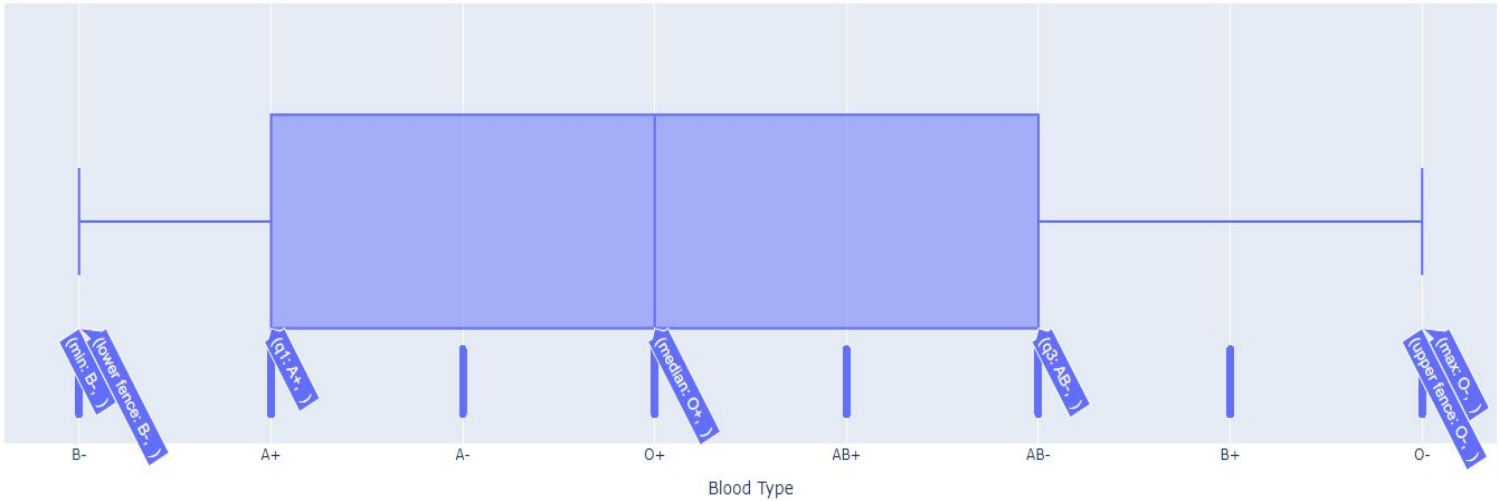
CATEGORICAL PLOT DEPICTING BILLING AMOUNT FOR EACH ADMISSION TYPE AS PER GENDER



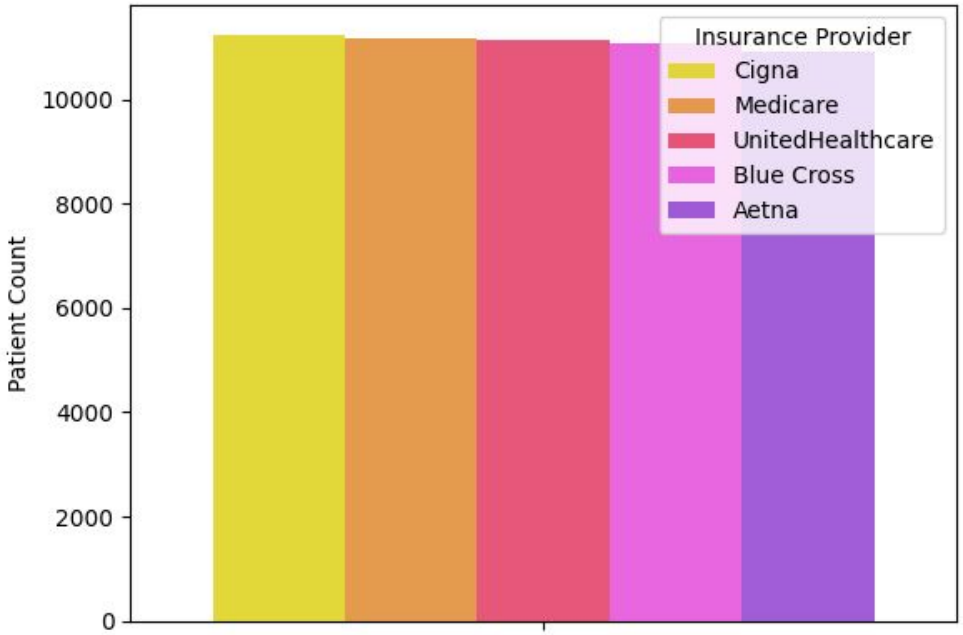
BOX PLOT DEPICTING STATISTICS OF MEDICAL CONDITIONS IN PATIENTS



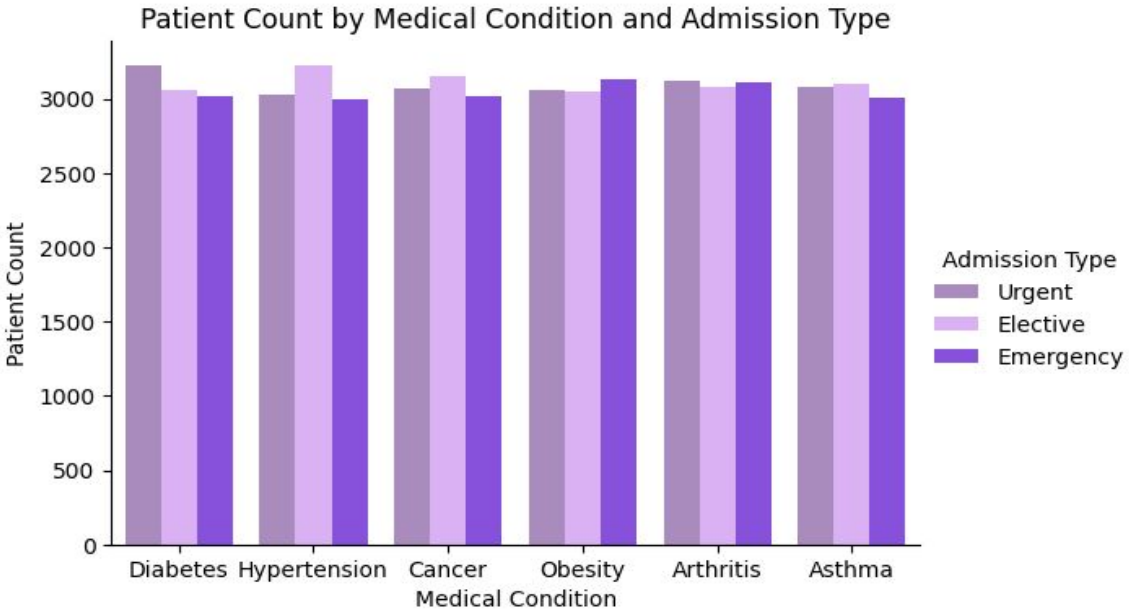
BOX PLOT DEPICTING STATISTICS OF BLOOD TYPE IN PATIENTS



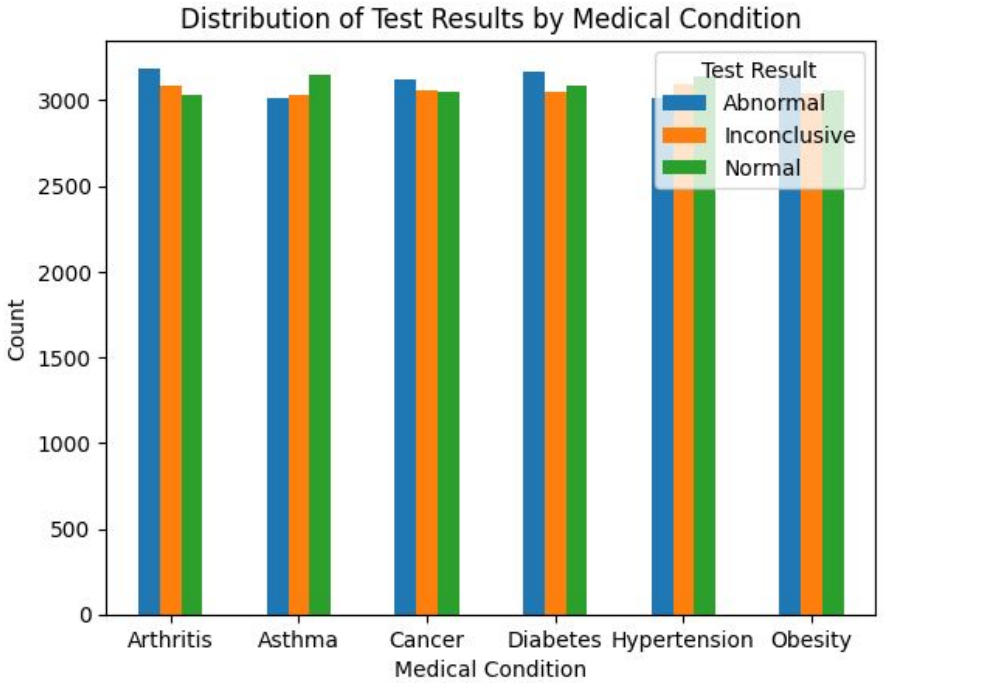
BAR GRAPH DEPICTING PATIENT COUNT AS FOR EACH INSURANCE PROVIDER



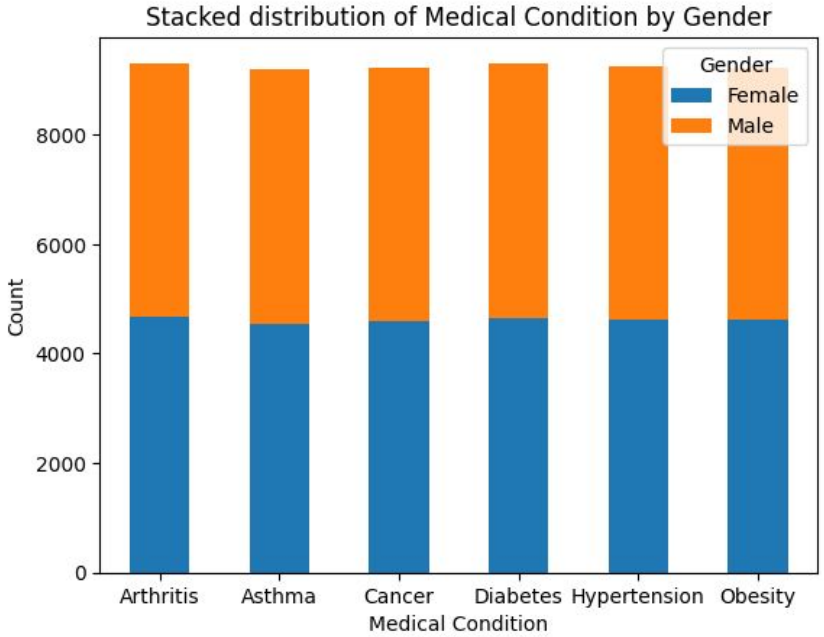
CATEGORICAL PLOT FOR PATIENT COUNT AS PER ADMISSION TYPE FOR EACH MEDICAL CONDITION



DISTRIBUTION OF TEST RESULTS AS PER MEDICAL CONDITION



STACKED BAR PLOT DEPICTING DISTRIBUTION OF MEDICAL CONDITIONS AS PER GENDER



STACKED BAR PLOT DEPICTING DISTRIBUTION OF TEST RESULTS PER ADMISSION TYPES