```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

df = pd.read_csv(r"C:\Users\susmi\Desktop\test_counter\ml_code\
HepatitisCdata.csv")
df.head()

   Unnamed: 0       Category  Age Sex   ALB   ALP   ALT   AST   BIL
CHE  \
0           1  0=Blood Donor   32   m  38.5  52.5   7.7  22.1   7.5
6.93
1           2  0=Blood Donor   32   m  38.5  70.3  18.0  24.7   3.9
11.17
2           3  0=Blood Donor   32   m  46.9  74.7  36.2  52.6   6.1
8.84
3           4  0=Blood Donor   32   m  43.2  52.0  30.6  22.6  18.9
7.33
4           5  0=Blood Donor   32   m  39.2  74.1  32.6  24.8   9.6
9.15

   CHOL   CREA   GGT  PROT
0  3.23  106.0  12.1  69.0
1  4.80   74.0  15.6  76.5
2  5.20   86.0  33.2  79.3
3  4.74   80.0  33.8  75.7
4  4.32   76.0  29.9  68.7

df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 615 entries, 0 to 614
Data columns (total 14 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   Unnamed: 0  615 non-null    int64
 1   Category    615 non-null    object
 2   Age         615 non-null    int64
 3   Sex         615 non-null    object
 4   ALB         614 non-null    float64
 5   ALP         597 non-null    float64
 6   ALT         614 non-null    float64
 7   AST         615 non-null    float64
 8   BIL         615 non-null    float64
 9   CHE         615 non-null    float64
 10  CHOL        605 non-null    float64
 11  CREA        615 non-null    float64
 12  GGT         615 non-null    float64
 13  PROT        614 non-null    float64
```

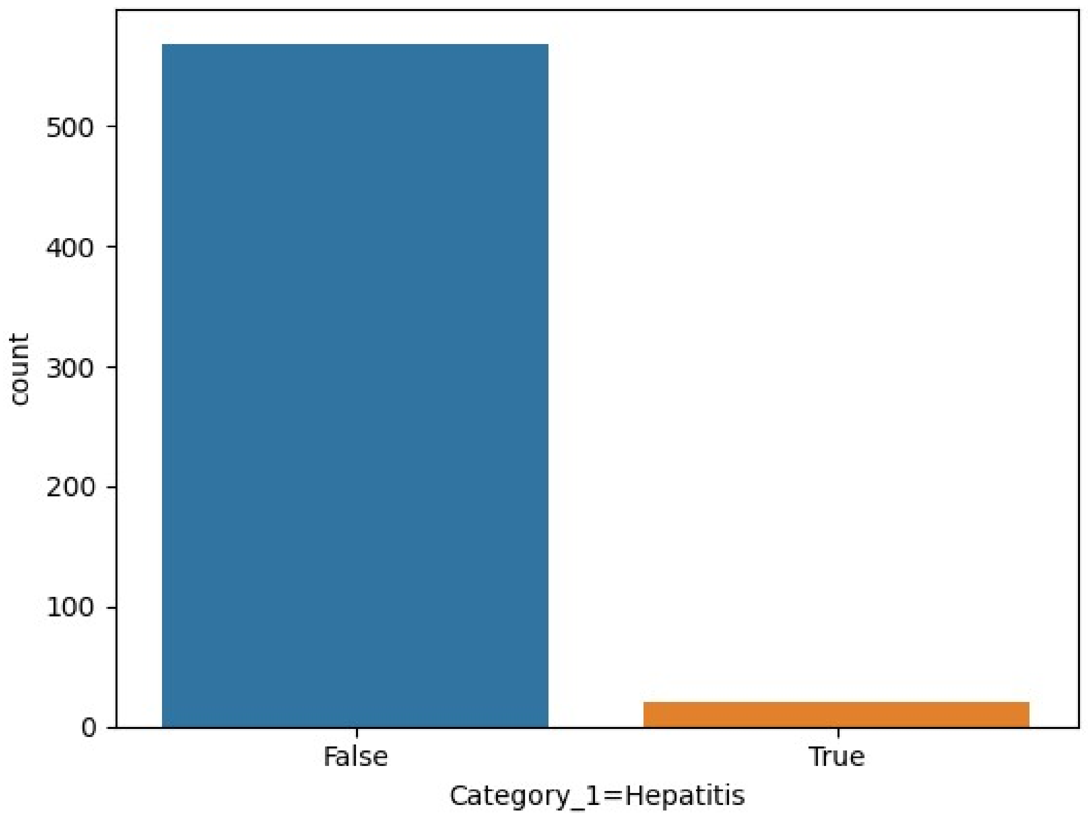
```
dtypes: float64(10), int64(2), object(2)
memory usage: 67.4+ KB

df = df.drop('Unnamed: 0', axis =1)

df = df.dropna()

df = pd.get_dummies(df, columns=['Category', 'Sex'])

df.head()

   Age   ALB   ALP   ALT   AST   BIL    CHE  CHOL   CREA   GGT
PROT  \
0   32  38.5  52.5   7.7  22.1   7.5   6.93  3.23  106.0  12.1  69.0

1   32  38.5  70.3  18.0  24.7   3.9  11.17  4.80   74.0  15.6  76.5

2   32  46.9  74.7  36.2  52.6   6.1   8.84  5.20   86.0  33.2  79.3

3   32  43.2  52.0  30.6  22.6  18.9   7.33  4.74   80.0  33.8  75.7

4   32  39.2  74.1  32.6  24.8   9.6   9.15  4.32   76.0  29.9  68.7


   Category_0=Blood Donor  Category_0s=suspect Blood Donor  \
0                    True                            False
1                    True                            False
2                    True                            False
3                    True                            False
4                    True                            False

   Category_1=Hepatitis  Category_2=Fibrosis  Category_3=Cirrhosis
Sex_f  \
0                 False                False                 False
False
1                 False                False                 False
False
2                 False                False                 False
False
3                 False                False                 False
False
4                 False                False                 False
False

   Sex_m
0   True
1   True
2   True
3   True
4   True
```

```
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report,
confusion_matrix
```

hepatitis

```
x =
df.drop(['Category_1=Hepatitis','Category_2=Fibrosis','Category_3=Cirr
hosis'], axis =1)
y = df['Category_1=Hepatitis']

x_train, x_test, y_train, y_test = train_test_split(x,y, test_size =
0.2, random_state =42)

clf = RandomForestClassifier(random_state = 42)
clf.fit(x_train, y_train)

RandomForestClassifier(random_state=42)

y_pred = clf.predict(x_test)

accuracy = accuracy_score(y_test, y_pred)
print(f'accuracytest:{accuracy}')

accuracytest:0.9491525423728814

report = classification_report(y_test, y_pred)
print(f'Classification Report:\n{report}')

Classification Report:
              precision    recall  f1-score   support

       False       0.95      1.00      0.97       110
        True       1.00      0.25      0.40         8

    accuracy                           0.95       118
   macro avg       0.97      0.62      0.69       118
weighted avg       0.95      0.95      0.93       118


cm = confusion_matrix(y_test, y_pred)
print(f'Confusion Matrix:\n{cm}')

Confusion Matrix:
[[110   0]
 [  6   2]]

sns.countplot(x = 'Category_1=Hepatitis', data = df)
```

```
<Axes: xlabel='Category_1=Hepatitis', ylabel='count'>
```



```
df['Category_1=Hepatitis'].value_counts()

Category_1=Hepatitis
False    569
True      20
Name: count, dtype: int64

affected_rate = 20/589

affected_rate

0.03395585738539898
```

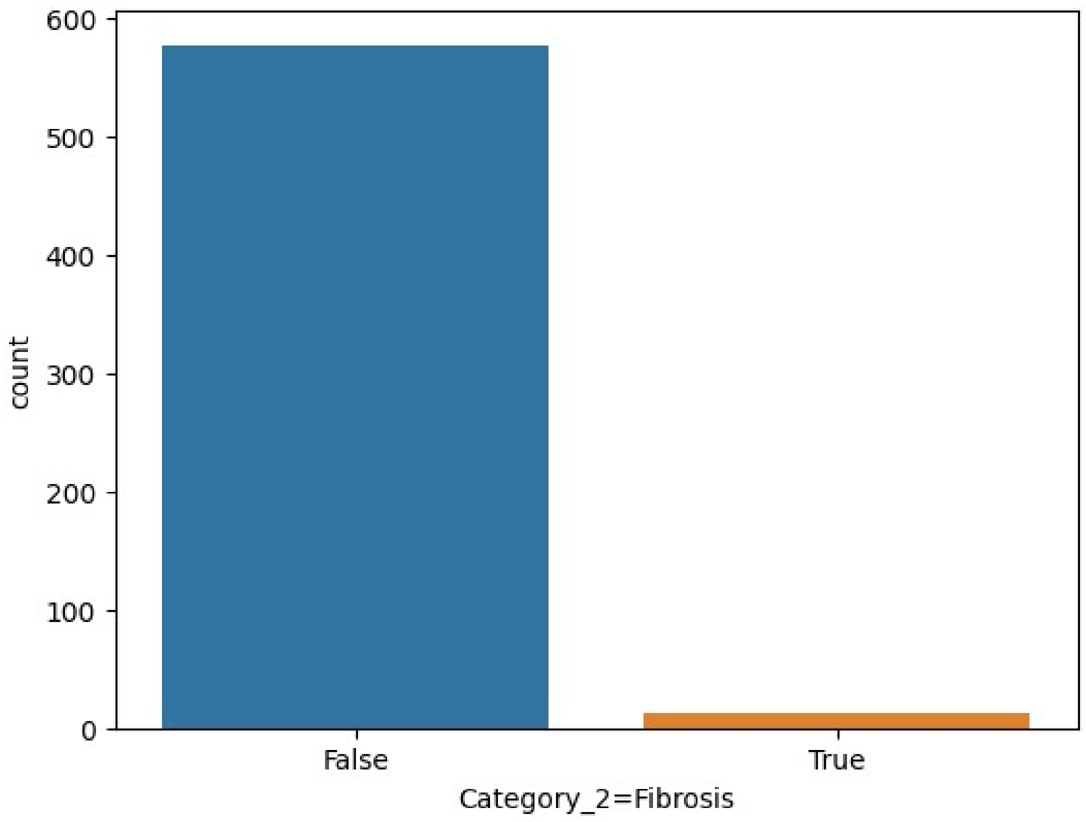
fibrosis

```
x =
df.drop(['Category_1=Hepatitis','Category_2=Fibrosis','Category_3=Cirr
hosis'], axis =1)
y = df['Category_2=Fibrosis']
```

```
x_train, x_test, y_train, y_test = train_test_split(x,y, test_size =
0.2, random_state =42)

clf = RandomForestClassifier(random_state = 42)
clf.fit(x_train, y_train)

RandomForestClassifier(random_state=42)

y_pred = clf.predict(x_test)

accuracy = accuracy_score(y_test, y_pred)
print(f'accuracytest:{accuracy}')

accuracytest:0.9745762711864406

report = classification_report(y_test, y_pred)
print(f'Classification Report:\n{report}')

Classification Report:
              precision    recall  f1-score   support

       False       0.97      1.00      0.99       115
        True       0.00      0.00      0.00         3

    accuracy                           0.97       118
   macro avg       0.49      0.50      0.49       118
weighted avg       0.95      0.97      0.96       118


C:\Users\susmi\anaconda3\lib\site-packages\sklearn\metrics\
_classification.py:1469: UndefinedMetricWarning: Precision and F-score
are ill-defined and being set to 0.0 in labels with no predicted
samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
C:\Users\susmi\anaconda3\lib\site-packages\sklearn\metrics\
_classification.py:1469: UndefinedMetricWarning: Precision and F-score
are ill-defined and being set to 0.0 in labels with no predicted
samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
C:\Users\susmi\anaconda3\lib\site-packages\sklearn\metrics\
_classification.py:1469: UndefinedMetricWarning: Precision and F-score
are ill-defined and being set to 0.0 in labels with no predicted
samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))

cm = confusion_matrix(y_test, y_pred)
print(f'Confusion Matrix:\n{cm}')

Confusion Matrix:
[[115   0]
 [  3   0]]
```

```
sns.countplot(x = 'Category_2=Fibrosis',data = df)
```

```
<Axes: xlabel='Category_2=Fibrosis', ylabel='count'>
```



```
df['Category_2=Fibrosis'].value_counts()
```

```
Category_2=Fibrosis
False    577
True      12
Name: count, dtype: int64
```

```
affected_rate = 12/589
```

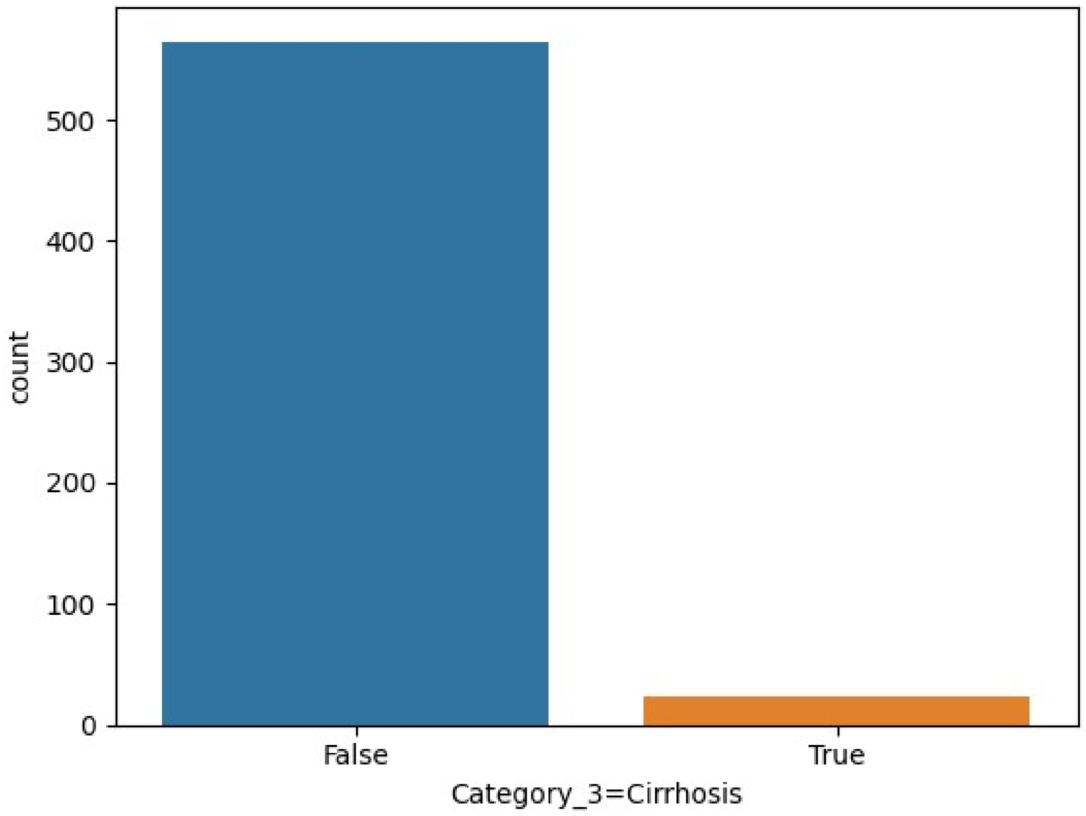
```
affected_rate
```

```
0.020373514431239389
```

cirrhosis

```
x = 
df.drop(['Category_1=Hepatitis','Category_2=Fibrosis','Category_3=Cirr
hosis'], axis =1)
y = df['Category_3=Cirrhosis']
```

```
x_train, x_test, y_train, y_test = train_test_split(x,y, test_size =
0.2, random_state =42)

clf = RandomForestClassifier(random_state = 42)
clf.fit(x_train, y_train)

RandomForestClassifier(random_state=42)

y_pred = clf.predict(x_test)

accuracy = accuracy_score(y_test, y_pred)
print(f'accuracytest:{accuracy}')

accuracytest:0.9830508474576272

report = classification_report(y_test, y_pred)
print(f'Classification Report:\n{report}')

Classification Report:
              precision    recall  f1-score   support

       False       0.98      1.00      0.99       112
        True       1.00      0.67      0.80         6

    accuracy                           0.98       118
   macro avg       0.99      0.83      0.90       118
weighted avg       0.98      0.98      0.98       118


cm = confusion_matrix(y_test, y_pred)
print(f'Confusion Matrix:\n{cm}')

Confusion Matrix:
[[112   0]
 [  2   4]]

sns.countplot(x ='Category_3=Cirrhosis', data = df )

<Axes: xlabel='Category_3=Cirrhosis', ylabel='count'>
```



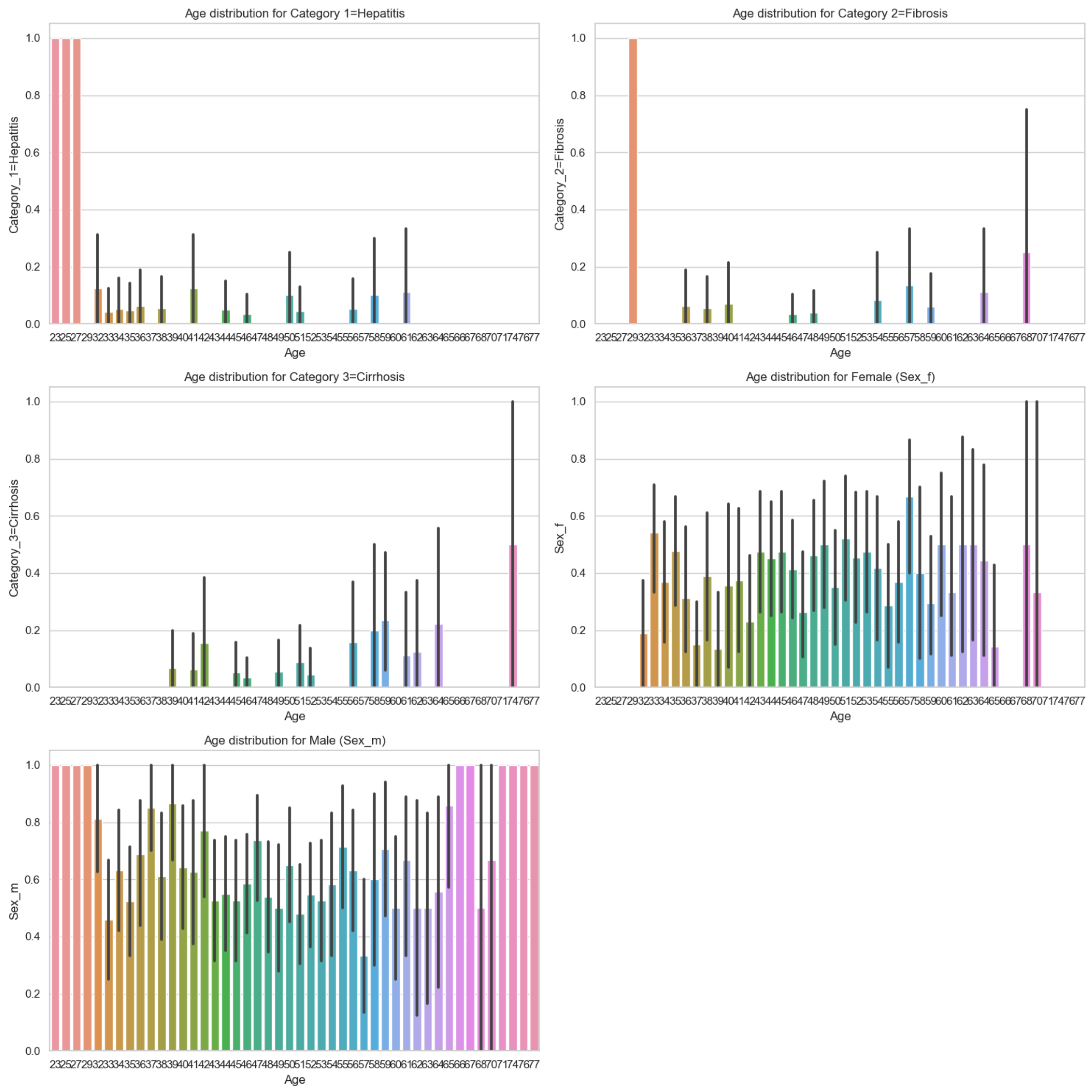
```
df['Category_3=Cirrhosis'].value_counts()

Category_3=Cirrhosis
False    565
True      24
Name: count, dtype: int64

affected_rate = 24/589

affected_rate

0.040747028862478778
```

data_visualization

```
df.info()

<class 'pandas.core.frame.DataFrame'>
Index: 589 entries, 0 to 612
Data columns (total 18 columns):
 #   Column                                  Non-Null Count  Dtype
---  ------                                  --------------  -----
 0   Age                                     589 non-null    int64
 1   ALB                               589 non-null    float64
 2   ALP                               589 non-null    float64
 3   ALT                               589 non-null    float64
 4   AST                               589 non-null    float64
 5   BIL                               589 non-null    float64
 6   CHE                               589 non-null    float64
 7   CHOL                              589 non-null    float64
 8   CREA                              589 non-null    float64
 9   GGT                               589 non-null    float64
 10  PROT                              589 non-null    float64
 11  Category_0=Blood Donor            589 non-null    bool
 12  Category_0s=suspect Blood Donor   589 non-null    bool
 13  Category_1=Hepatitis              589 non-null    bool
 14  Category_2=Fibrosis               589 non-null    bool
 15  Category_3=Cirrhosis              589 non-null    bool
 16  Sex_f                             589 non-null    bool
 17  Sex_m                             589 non-null    bool
dtypes: bool(7), float64(10), int64(1)
memory usage: 59.2 KB
```

```
sns.set(style="whitegrid")


fig, axes = plt.subplots(nrows=3, ncols=2, figsize=(15, 15))

# Plot Age distribution for Category 1=Hepatitis
sns.barplot(x="Age", y="Category_1=Hepatitis", data=df, ax=axes[0, 0])
axes[0, 0].set_title("Age distribution for Category 1=Hepatitis")

# Plot Age distribution for Category 2=Fibrosis
sns.barplot(x="Age", y="Category_2=Fibrosis", data=df, ax=axes[0, 1])
axes[0, 1].set_title("Age distribution for Category 2=Fibrosis")

# Plot Age distribution for Category 3=Cirrhosis
sns.barplot(x="Age", y="Category_3=Cirrhosis", data=df, ax=axes[1, 0])
axes[1, 0].set_title("Age distribution for Category 3=Cirrhosis")

# Plot Age distribution for Female (Sex_f)
sns.barplot(x="Age", y="Sex_f", data=df, ax=axes[1, 1])
axes[1, 1].set_title("Age distribution for Female (Sex_f)")

# Plot Age distribution for Male (Sex_m)
sns.barplot(x="Age", y="Sex_m", data=df, ax=axes[2, 0])
axes[2, 0].set_title("Age distribution for Male (Sex_m)")

# Remove empty subplot
fig.delaxes(axes[2, 1])

# Adjust layout
```

```
plt.tight_layout()
plt.show()
```



```
def func_plot(col):
    fig, axes = plt.subplots(nrows=3, ncols=2, figsize=(15, 15))

    # Plot Age distribution for Category 1=Hepatitis
    sns.barplot(x=col, y="Category_1=Hepatitis", data=df, ax=axes[0,
0])
    axes[0, 0].set_title(col + " distribution for Category
1=Hepatitis")
```

```
    # Plot Age distribution for Category 2=Fibrosis
    sns.barplot(x=col, y="Category_2=Fibrosis", data=df, ax=axes[0,
1])
    axes[0, 1].set_title(col + " distribution for Category
2=Fibrosis")

    # Plot Age distribution for Category 3=Cirrhosis
    sns.barplot(x=col, y="Category_3=Cirrhosis", data=df, ax=axes[1,
0])
    axes[1, 0].set_title(col + " distribution for Category
3=Cirrhosis")

    # Plot Age distribution for Female (Sex_f)
    sns.barplot(x=col, y="Sex_f", data=df, ax=axes[1, 1])
    axes[1, 1].set_title(col + " distribution for Female (Sex_f)")

    # Plot Age distribution for Male (Sex_m)
    sns.barplot(x=col, y="Sex_m", data=df, ax=axes[2, 0])
    axes[2, 0].set_title(col + " distribution for Male (Sex_m)")

    # Remove empty subplot
    fig.delaxes(axes[2, 1])

    # Adjust layout
    plt.tight_layout()
    plt.show()

df.head(1)

   Age   ALB   ALP  ALT   AST  BIL   CHE  CHOL   CREA   GGT  PROT  \
0   32  38.5  52.5  7.7  22.1  7.5  6.93  3.23  106.0  12.1  69.0

   Category_0=Blood Donor  Category_0s=suspect Blood Donor  \
0                    True                            False

   Category_1=Hepatitis  Category_2=Fibrosis  Category_3=Cirrhosis
Sex_f  \
0                 False                False                 False
False

   Sex_m
0   True
```

Age (int64):

Represents the age of the individuals in the dataset. It is stored as an integer (int64). ALB (float64):

Stands for Albumin, a protein in the blood. It is measured in grams per deciliter (g/dL). ALP (float64):

Alkaline Phosphatase, an enzyme found in the liver and other tissues. The measurement is in international units per liter (IU/L). ALT (float64):

Alanine Transaminase, an enzyme that can indicate liver damage. It is measured in IU/L. AST (float64):

Aspartate Transaminase, an enzyme found in the liver and other tissues. Elevated levels may indicate liver damage. Measured in IU/L. BIL (float64):

Bilirubin, a yellowish substance that can be a marker for liver function. Measured in milligrams per deciliter (mg/dL). CHE (float64):

Cholinesterase, an enzyme that can be related to liver function. The unit of measurement is IU/L. CHOL (float64):

Cholesterol level in the blood, measured in mg/dL. CREA (float64):

Creatinine, a waste product in the blood that can indicate kidney function. Measured in mg/dL. GGT (float64):

Gamma-Glutamyl Transferase, an enzyme found in the liver. Elevated levels may indicate liver or bile duct issues. Measured in IU/L. PROT (float64):

Total protein level in the blood, measured in g/dL. Category_0=Blood Donor (bool):

Boolean indicating whether the individual falls into the category of a Blood Donor (True/False). Category_0s=suspect Blood Donor (bool):

Boolean indicating whether the individual falls into the category of a suspected Blood Donor (True/False). Category_1=Hepatitis (bool):

Boolean indicating whether the individual falls into the category of having Hepatitis (True/False). Category_2=Fibrosis (bool):

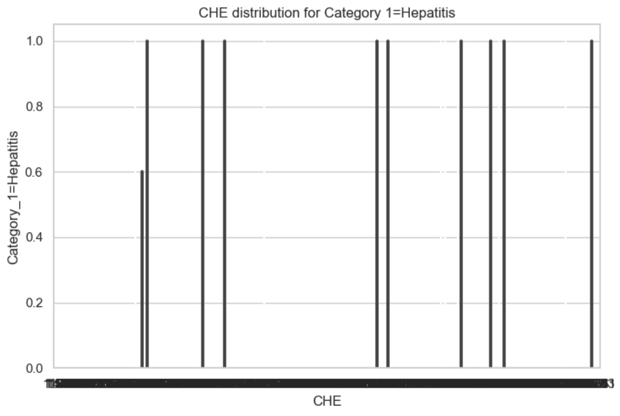
Boolean indicating whether the individual falls into the category of having Fibrosis (True/False). Category_3=Cirrhosis (bool):

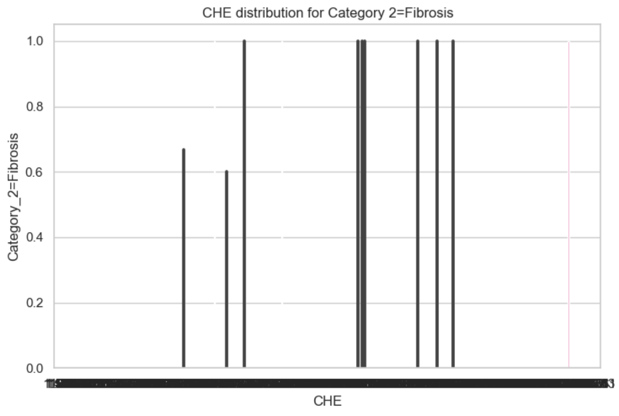
Boolean indicating whether the individual falls into the category of having Cirrhosis (True/False). Sex_f (bool):

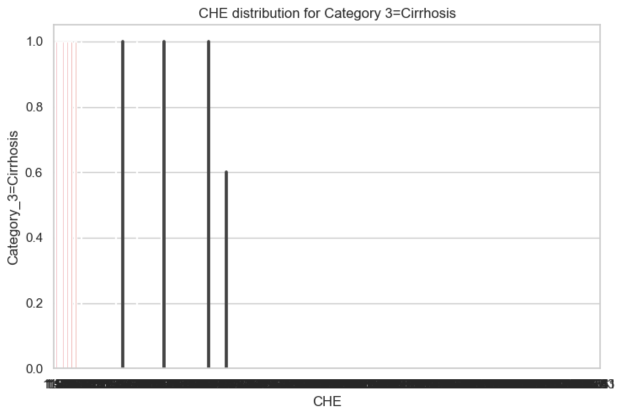
Boolean indicating the gender as female (True/False). Sex_m (bool):

Boolean indicating the gender as male (True/False)

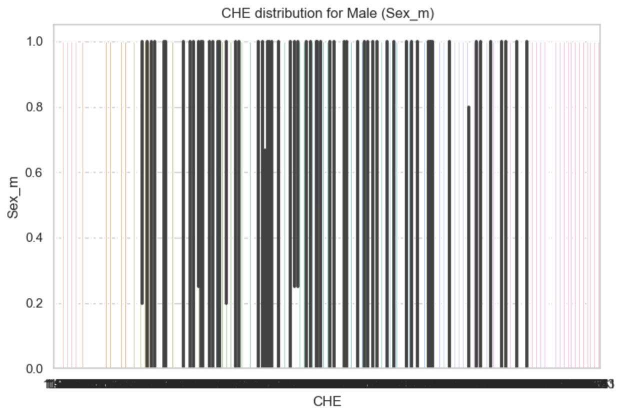
```
heading = ["Age", "Category_0=Blood Donor", "Category_0s=suspect Blood Donor"]

blood_content= ["ALB", "ALP", "ALT", "AST","BIL"]

func_plot("CHE")
```



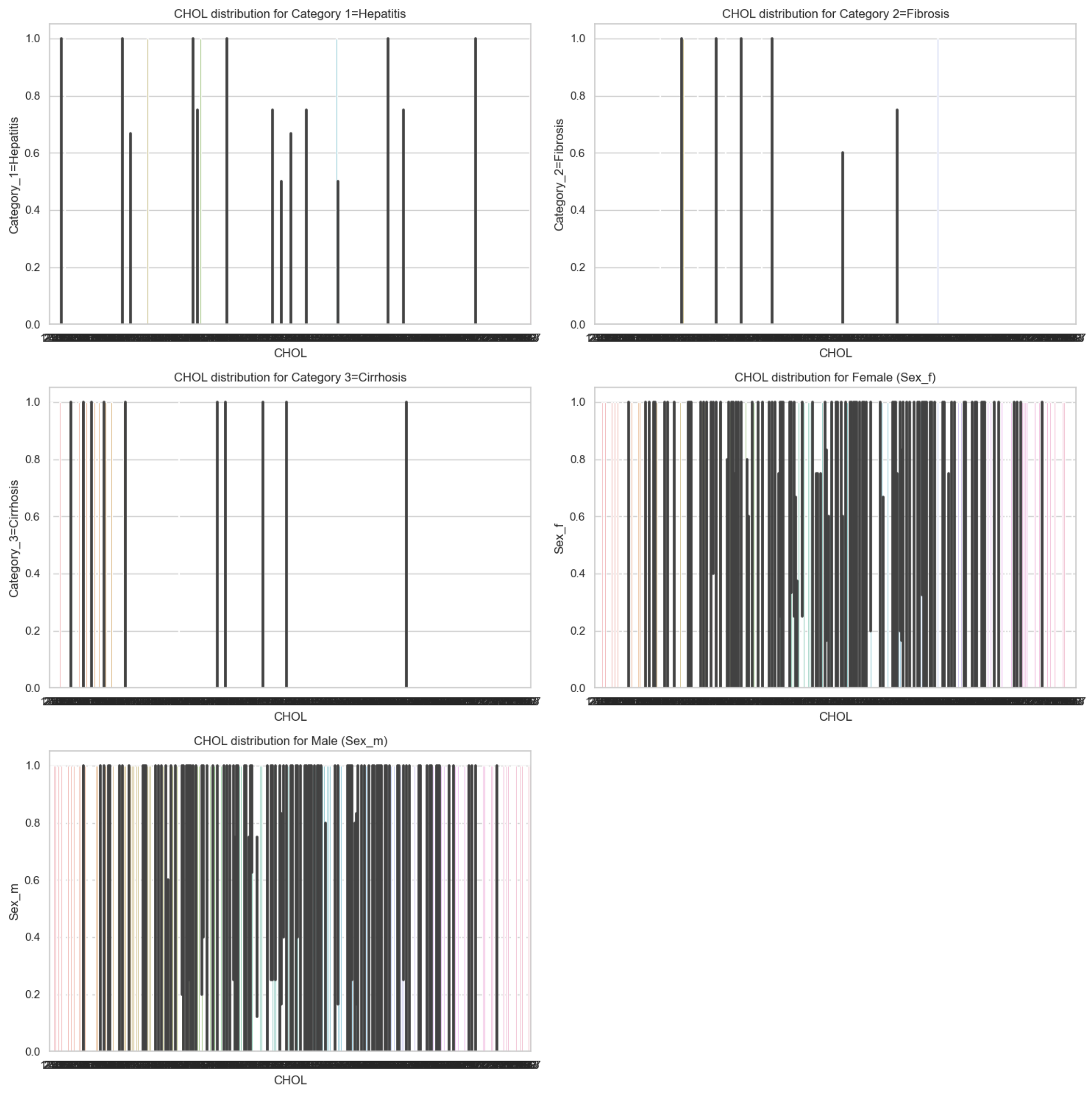




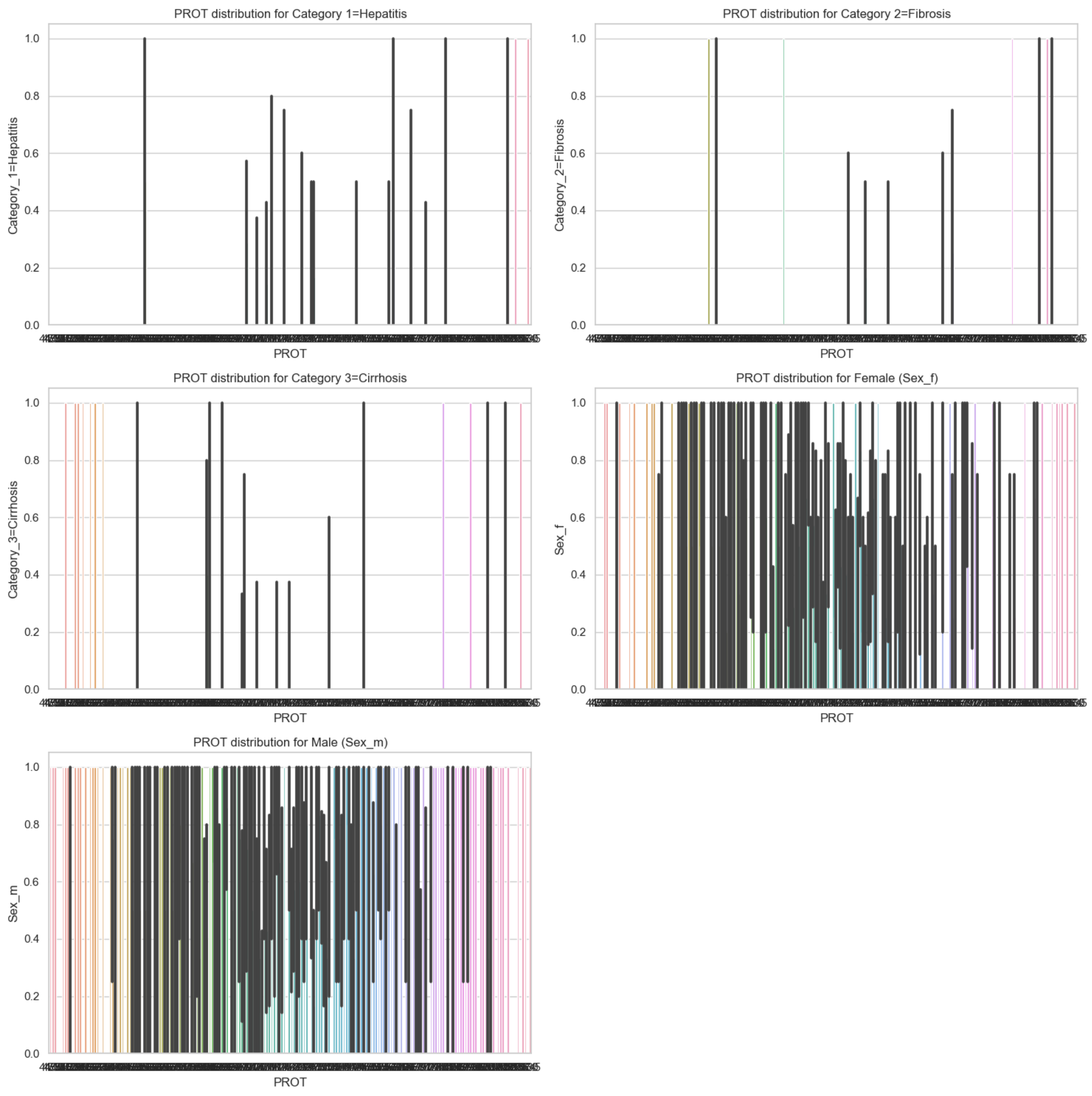
CHE distribution for Female (Sex_f)
Sex_f
CHE



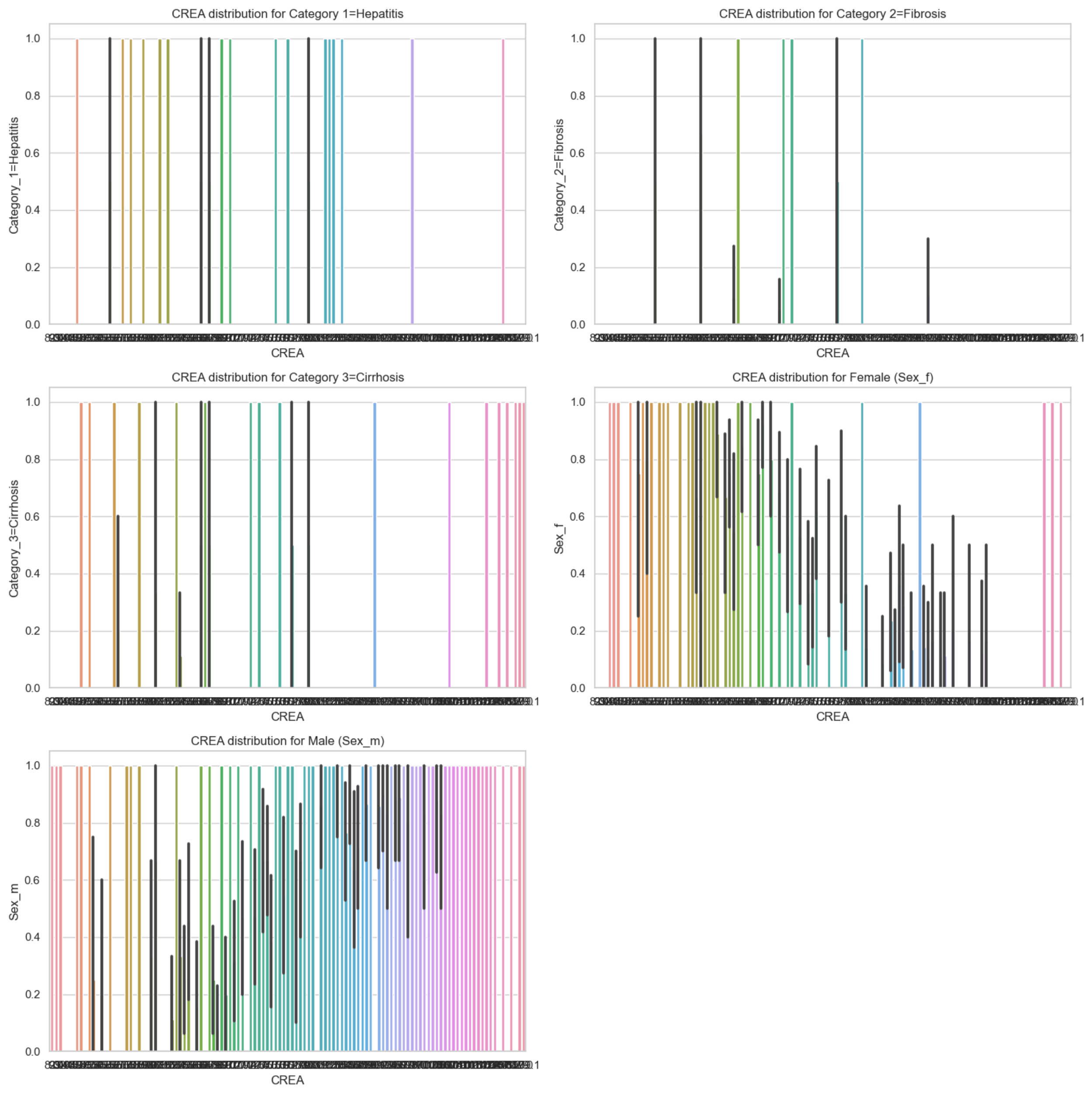
```
func_plot("CHOL")
```



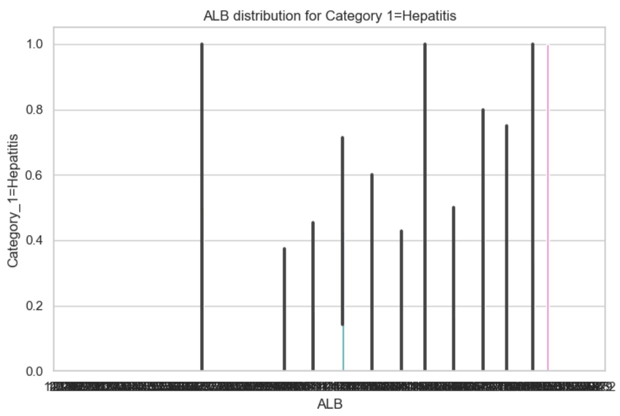
```
heading_3 = ["CREA", "CGT", "PROT"]

func_plot("PROT")
```

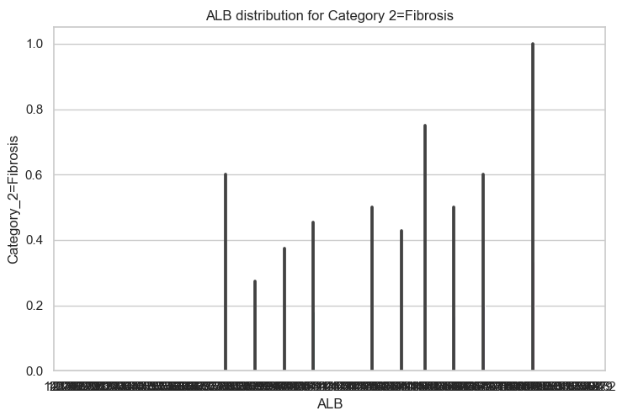


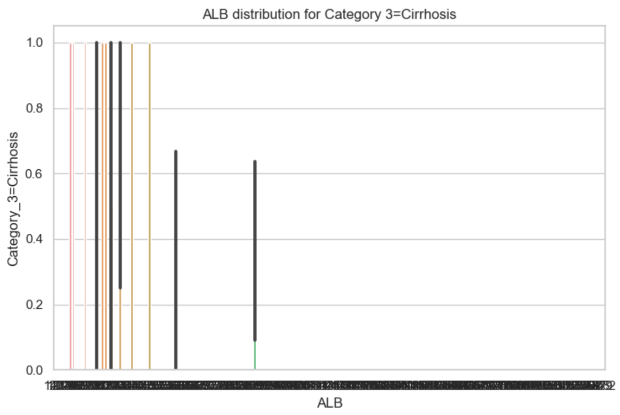
```python
func_plot("CREA")
```

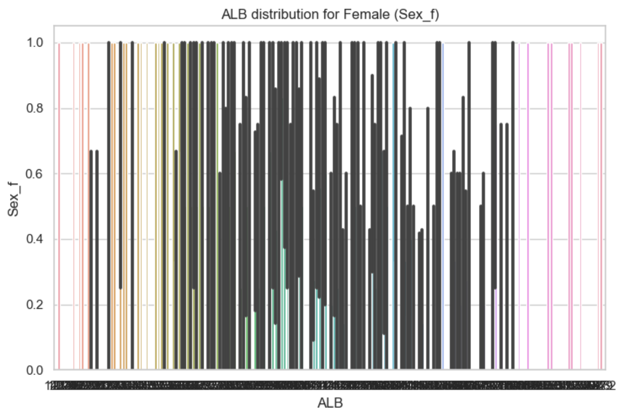


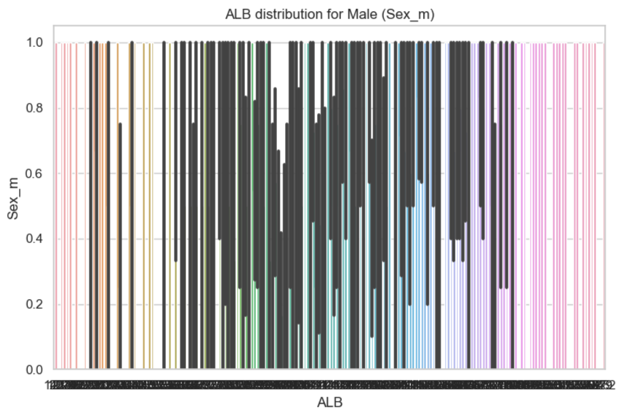
```
for head in blood_content:
    func_plot(head)
```

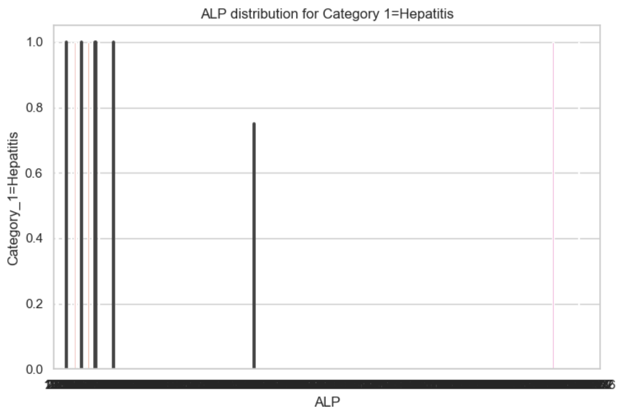

ALB distribution for Category 1=Hepatitis
Category_1=Hepatitis
0.0
0.2
0.4
0.6
0.8
1.0
ALB

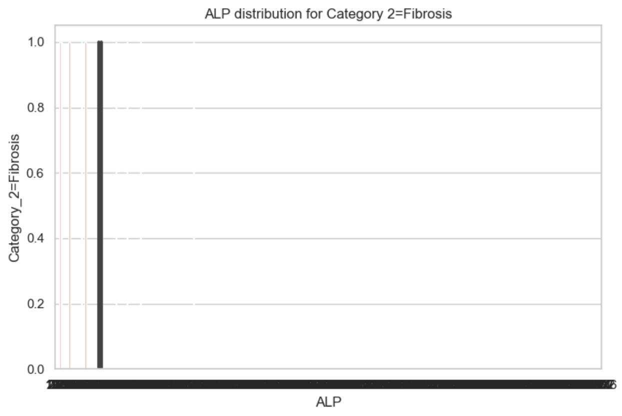

ALB distribution for Category 2=Fibrosis
Category_2=Fibrosis
0.0
0.2
0.4
0.6
0.8
1.0
ALB

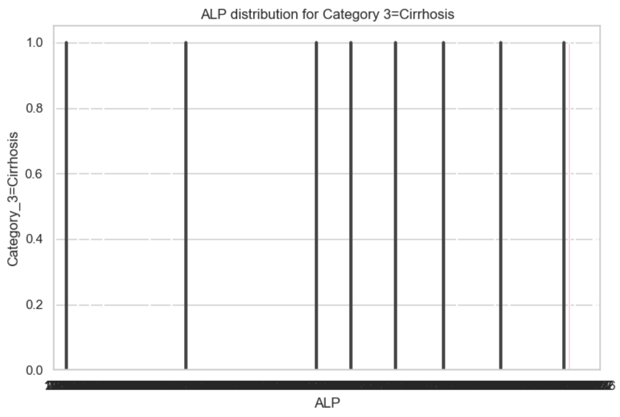

ALB distribution for Category 3=Cirrhosis
Category_3=Cirrhosis
0.0
0.2
0.4
0.6
0.8
1.0
ALB


ALB distribution for Female (Sex_f)
Sex_f
0.0
0.2
0.4
0.6
0.8
1.0
ALB


ALB distribution for Male (Sex_m)
Sex_m
0.0
0.2
0.4
0.6
0.8
1.0
ALB

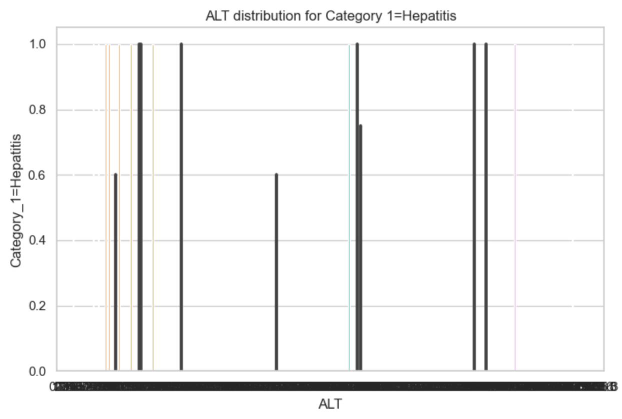

ALP distribution for Category 1=Hepatitis
Category_1=Hepatitis
ALP

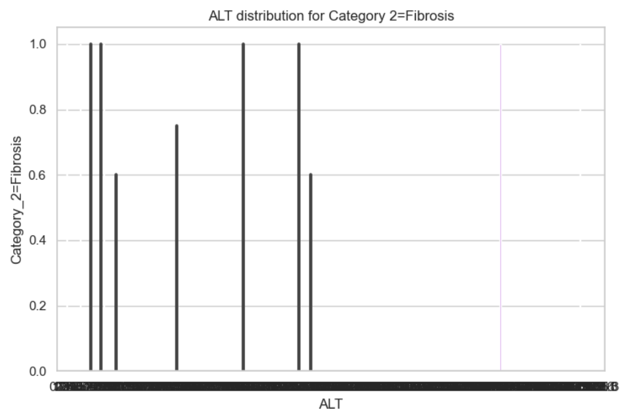

ALP distribution for Category 2=Fibrosis
Category_2=Fibrosis
ALP

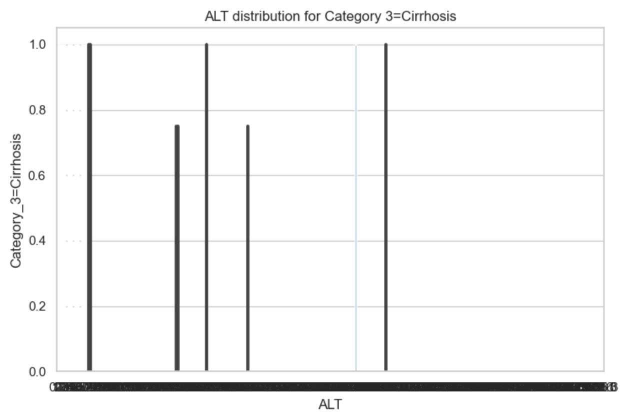

ALP distribution for Category 3=Cirrhosis
Category_3=Cirrhosis
ALP

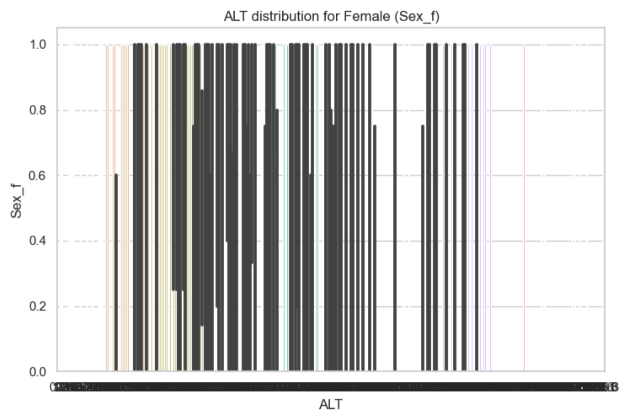
ALP distribution for Female (Sex_f)
Sex_f
ALP

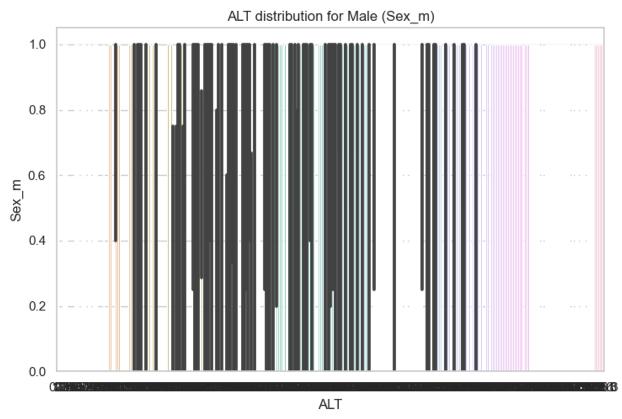
ALP distribution for Male (Sex_m)
Sex_m
ALP

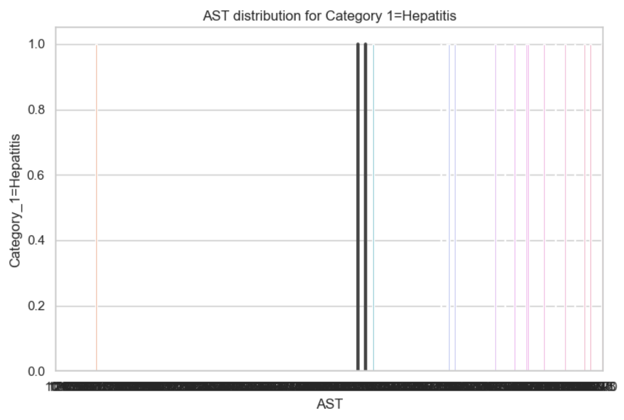

ALT distribution for Category 1=Hepatitis
Category_1=Hepatitis
ALT
1.0
0.8
0.6
0.4
0.2
0.0

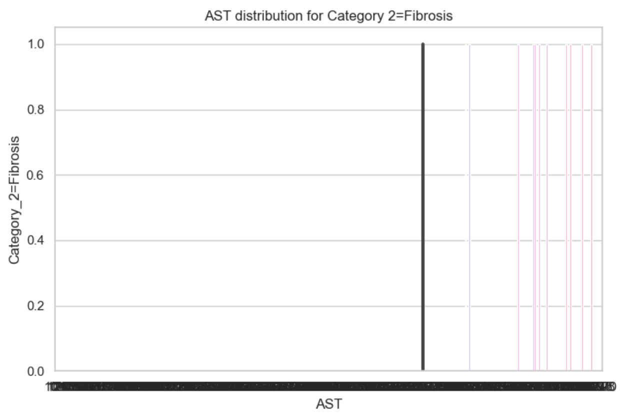

ALT distribution for Category 2=Fibrosis
Category_2=Fibrosis
ALT
1.0
0.8
0.6
0.4
0.2
0.0

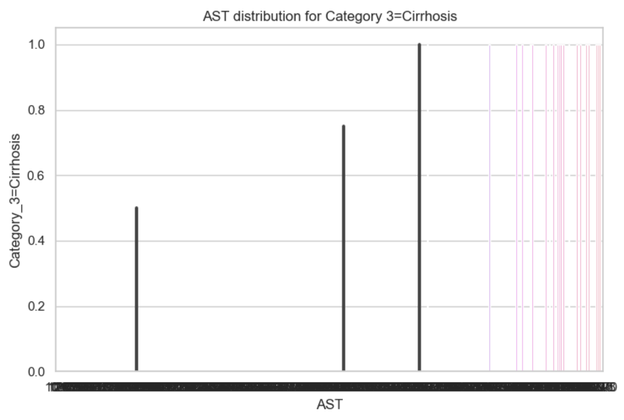

ALT distribution for Category 3=Cirrhosis
Category_3=Cirrhosis
ALT
1.0
0.8
0.6
0.4
0.2
0.0

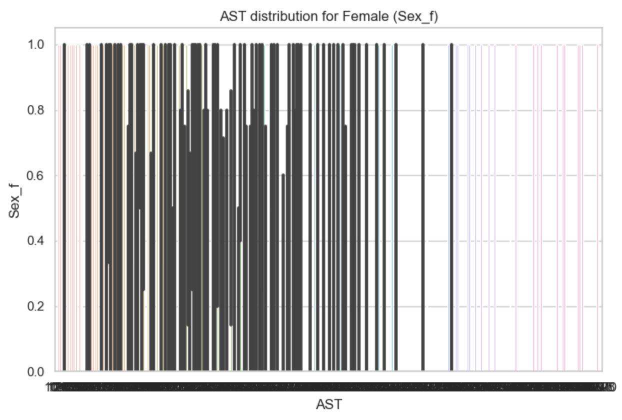

ALT distribution for Female (Sex_f)
Sex_f
ALT
1.0
0.8
0.6
0.4
0.2
0.0

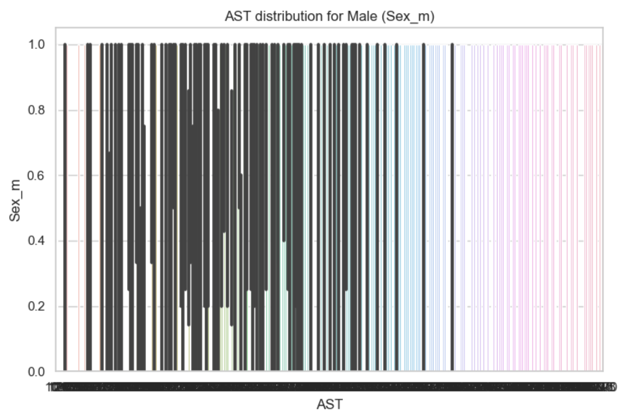

ALT distribution for Male (Sex_m)
Sex_m
ALT
1.0
0.8
0.6
0.4
0.2
0.0

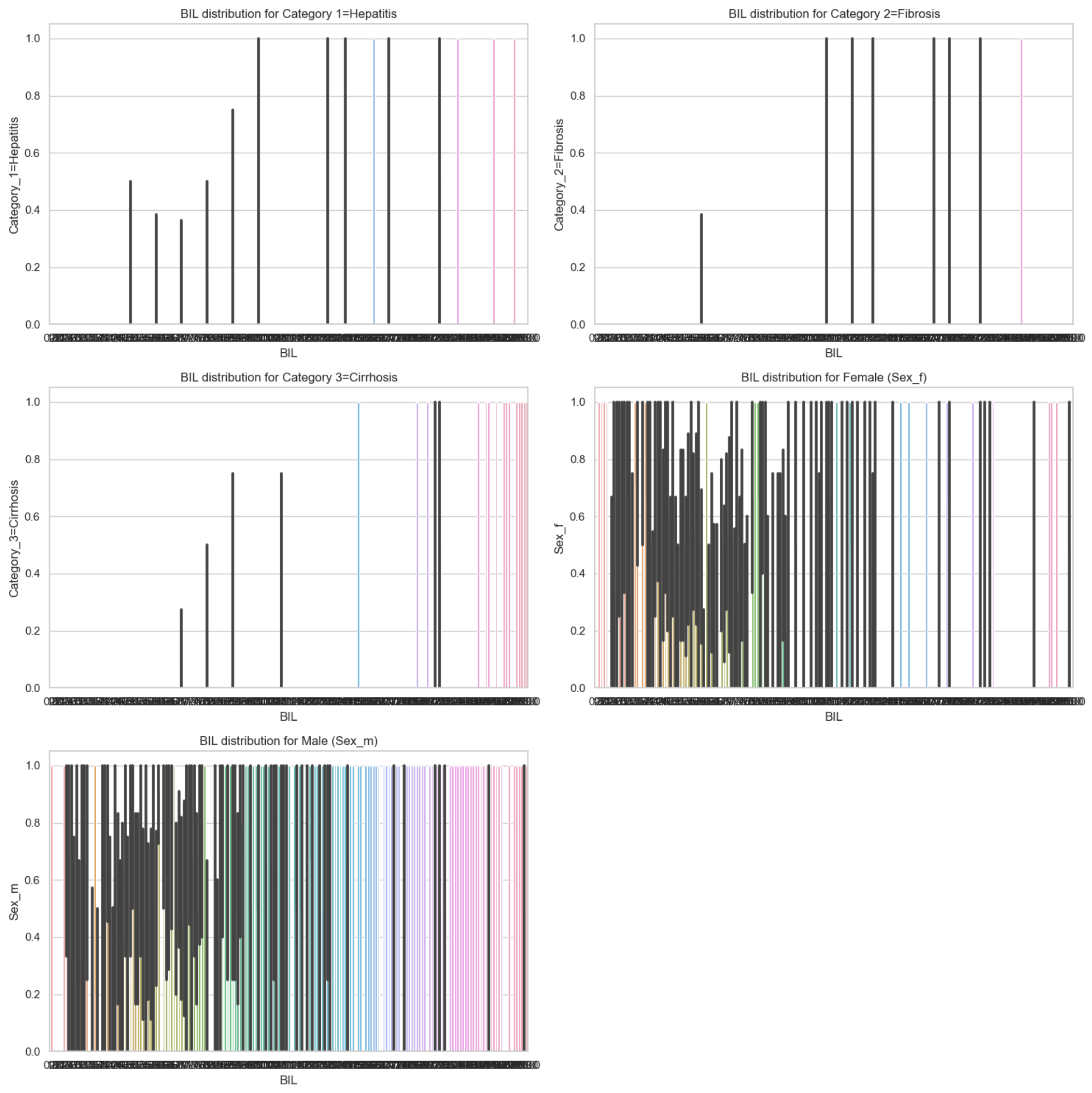

AST distribution for Category 1=Hepatitis
Category_1=Hepatitis
AST
0.0
0.2
0.4
0.6
0.8
1.0


AST distribution for Category 2=Fibrosis
Category_2=Fibrosis
AST
0.0
0.2
0.4
0.6
0.8
1.0

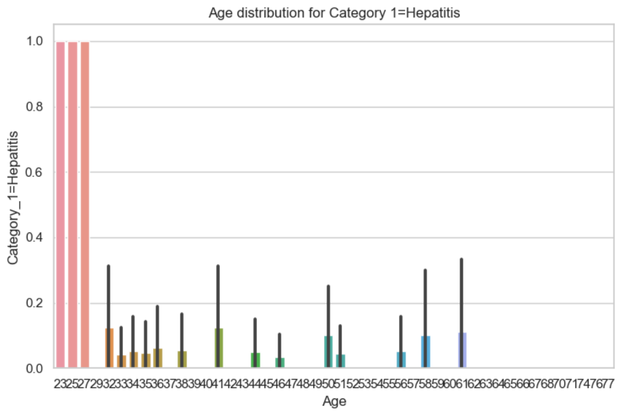

AST distribution for Category 3=Cirrhosis
Category_3=Cirrhosis
AST
0.0
0.2
0.4
0.6
0.8
1.0

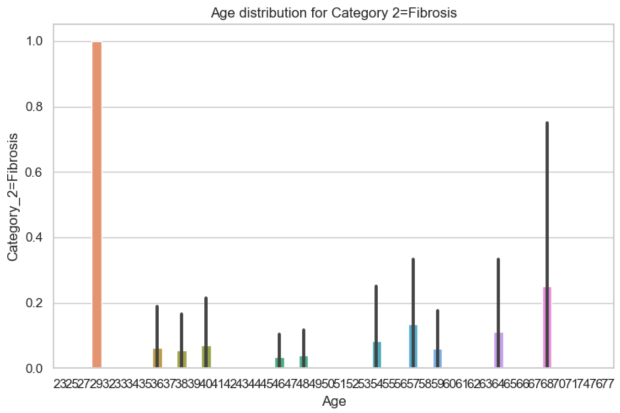

AST distribution for Female (Sex_f)
Sex_f
AST
0.0
0.2
0.4
0.6
0.8
1.0

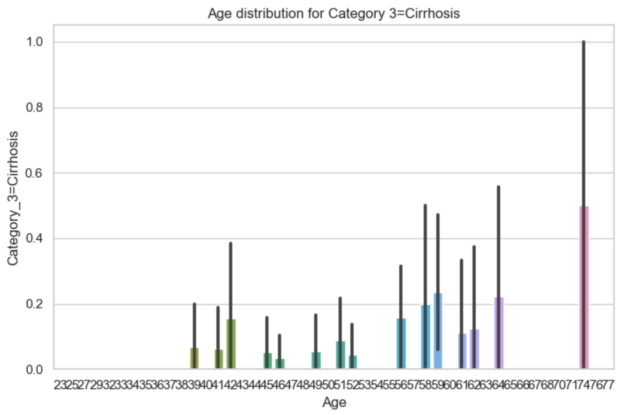

AST distribution for Male (Sex_m)
Sex_m
AST
0.0
0.2
0.4
0.6
0.8
1.0

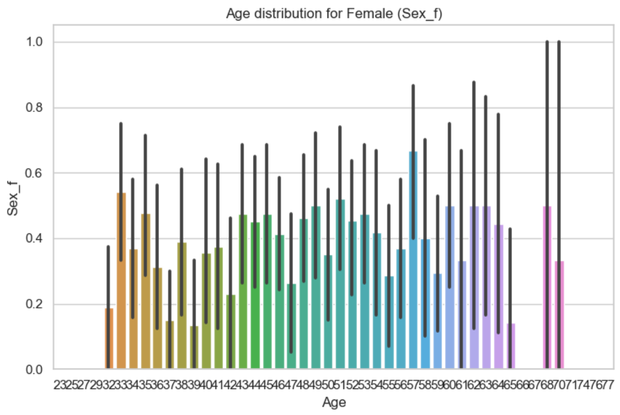


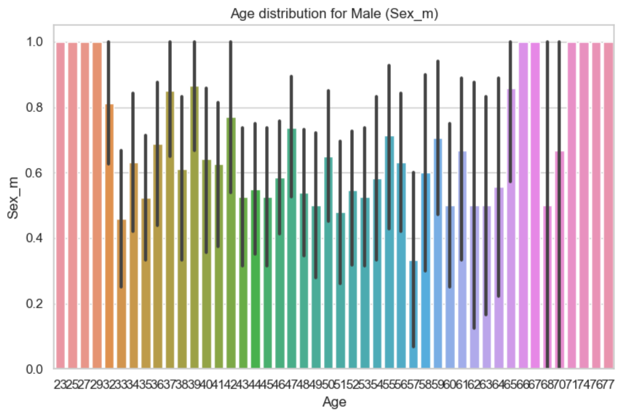
```
for head in heading:
    func_plot(head)
```

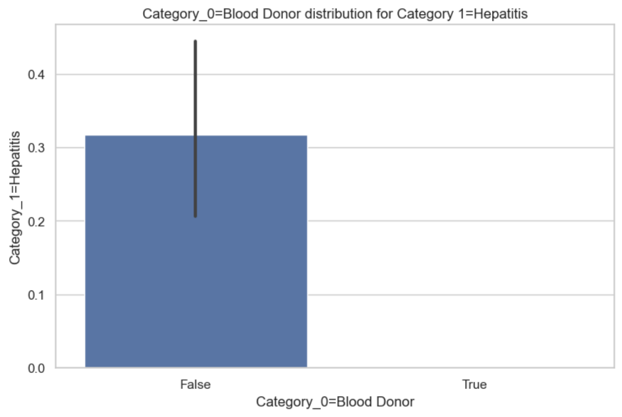

Age distribution for Category 1=Hepatitis
Category_1=Hepatitis
Age

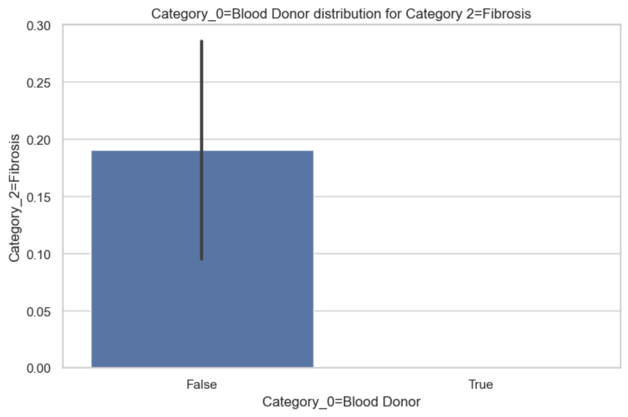

Age distribution for Category 2=Fibrosis
Category_2=Fibrosis
Age

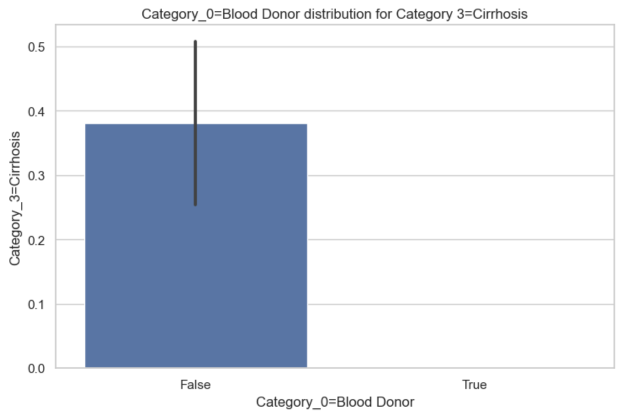

Age distribution for Category 3=Cirrhosis
Category_3=Cirrhosis
Age

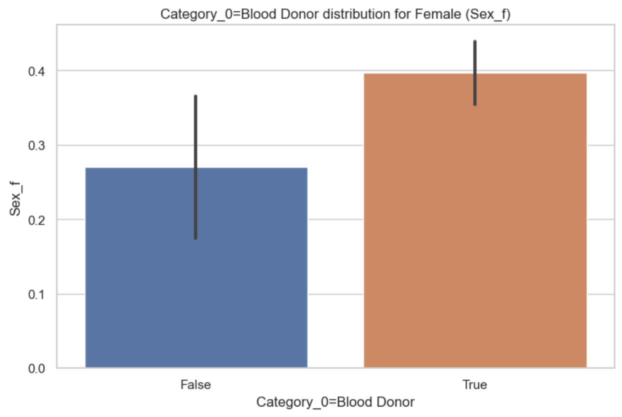

Age distribution for Female (Sex_f)
Sex_f
Age

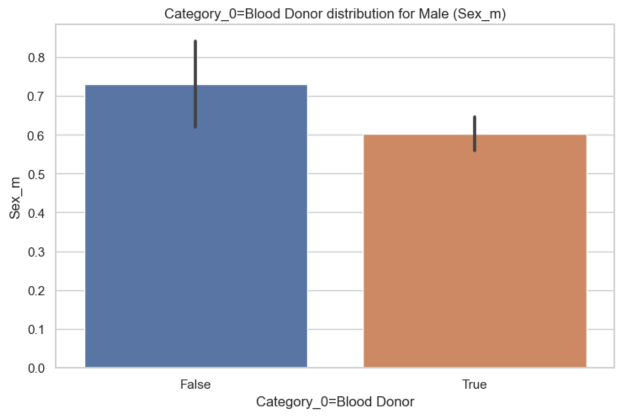

Age distribution for Male (Sex_m)
Sex_m
Age

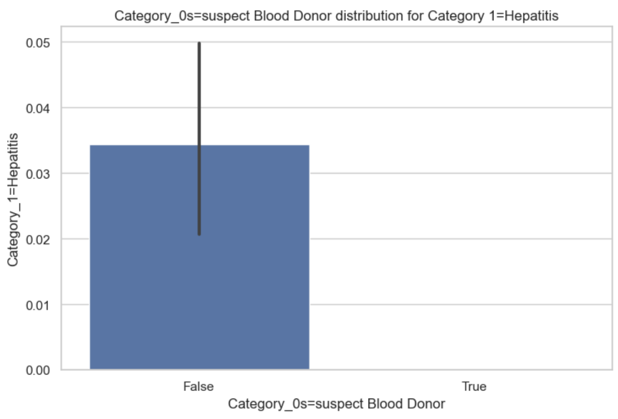

Category_0=Blood Donor distribution for Category 1=Hepatitis
Category_1=Hepatitis
0.0
0.1
0.2
0.3
0.4
False
True
Category_0=Blood Donor

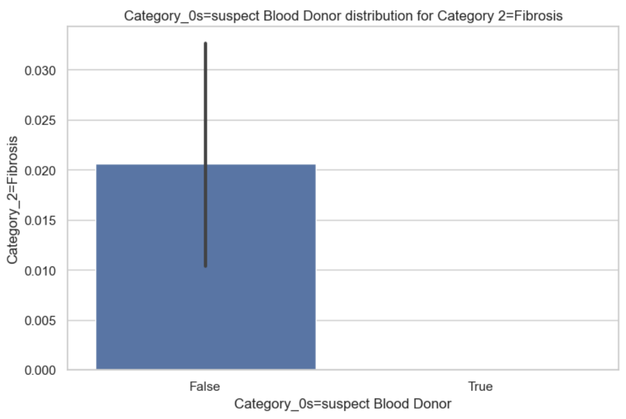

Category_0=Blood Donor distribution for Category 2=Fibrosis
Category_2=Fibrosis
0.00
0.05
0.10
0.15
0.20
0.25
0.30
False
True
Category_0=Blood Donor

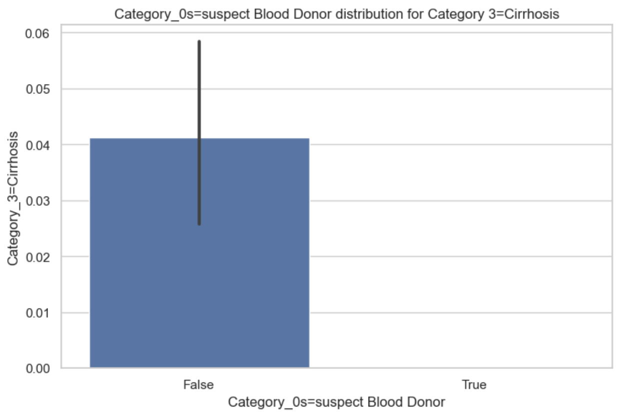

Category_0=Blood Donor distribution for Category 3=Cirrhosis
Category_3=Cirrhosis
0.0
0.1
0.2
0.3
0.4
0.5
False
True
Category_0=Blood Donor

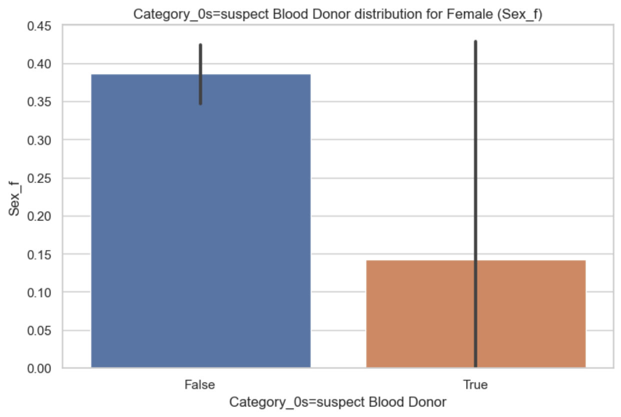

Category_0=Blood Donor distribution for Female (Sex_f)
Sex_f
0.0
0.1
0.2
0.3
0.4
False
True
Category_0=Blood Donor

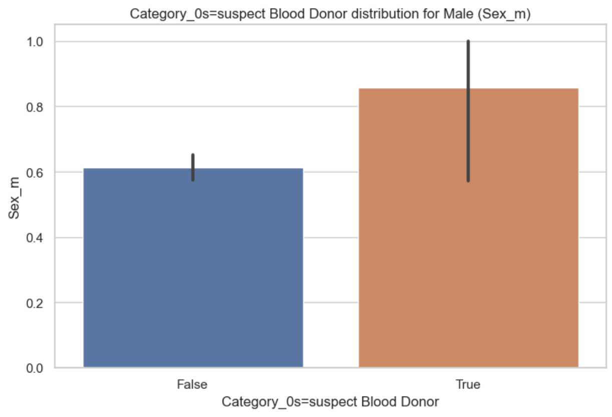

Category_0=Blood Donor distribution for Male (Sex_m)
Sex_m
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
False
True
Category_0=Blood Donor


Category_0s=suspect Blood Donor distribution for Category 1=Hepatitis
Category_1=Hepatitis
0.00
0.01
0.02
0.03
0.04
0.05
False
True
Category_0s=suspect Blood Donor


Category_0s=suspect Blood Donor distribution for Category 2=Fibrosis
Category_2=Fibrosis
0.000
0.005
0.010
0.015
0.020
0.025
0.030
False
True
Category_0s=suspect Blood Donor


Category_0s=suspect Blood Donor distribution for Category 3=Cirrhosis
Category_3=Cirrhosis
0.00
0.01
0.02
0.03
0.04
0.05
0.06
False
True
Category_0s=suspect Blood Donor


Category_0s=suspect Blood Donor distribution for Female (Sex_f)
Sex_f
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0.40
0.45
False
True
Category_0s=suspect Blood Donor


Category_0s=suspect Blood Donor distribution for Male (Sex_m)
Sex_m
0.0
0.2
0.4
0.6
0.8
1.0
False
True
Category_0s=suspect Blood Donor