Univariate,Bivariate and Multivariate Analysis

In [2]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
import math
```

In [3]:
```
card_approval_df=pd.read_csv('clean_dataset.csv')
print(card_approval_df.head())
```

```
   Gender    Age   Debt  Married  BankCustomer     Industry Ethnicity  \
0       1  30.83  0.000        1             1  Industrials     White
1       0  58.67  4.460        1             1    Materials     Black
2       0  24.50  0.500        1             1    Materials     Black
3       1  27.83  1.540        1             1  Industrials     White
4       1  20.17  5.625        1             1  Industrials     White

   YearsEmployed  PriorDefault  Employed  CreditScore  DriversLicense  \
0           1.25             1         1            1               0
1           3.04             1         1            6               0
2           1.50             1         0            0               0
3           3.75             1         1            5               1
4           1.71             1         0            0               0

        Citizen  ZipCode  Income  Approved
0       ByBirth      202       0         1
1       ByBirth       43     560         1
2       ByBirth      280     824         1
3       ByBirth      100       3         1
4  ByOtherMeans      120       0         1
```

In [4]:
```
print(card_approval_df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 690 entries, 0 to 689
Data columns (total 16 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   Gender          690 non-null    int64  
 1   Age             690 non-null    float64
 2   Debt            690 non-null    float64
 3   Married         690 non-null    int64  
 4   BankCustomer    690 non-null    int64  
 5   Industry        690 non-null    object 
 6   Ethnicity       690 non-null    object 
 7   YearsEmployed   690 non-null    float64
 8   PriorDefault    690 non-null    int64  
 9   Employed        690 non-null    int64  
 10  CreditScore     690 non-null    int64  
 11  DriversLicense  690 non-null    int64  
 12  Citizen         690 non-null    object 
 13  ZipCode         690 non-null    int64  
 14  Income          690 non-null    int64  
 15  Approved        690 non-null    int64  
dtypes: float64(3), int64(10), object(3)
memory usage: 86.4+ KB
None
```

In [5]:
```
card_approval_df.duplicated().sum()
```

Out[5]:
```
0
```

Univariate Analysis of continuous Variables

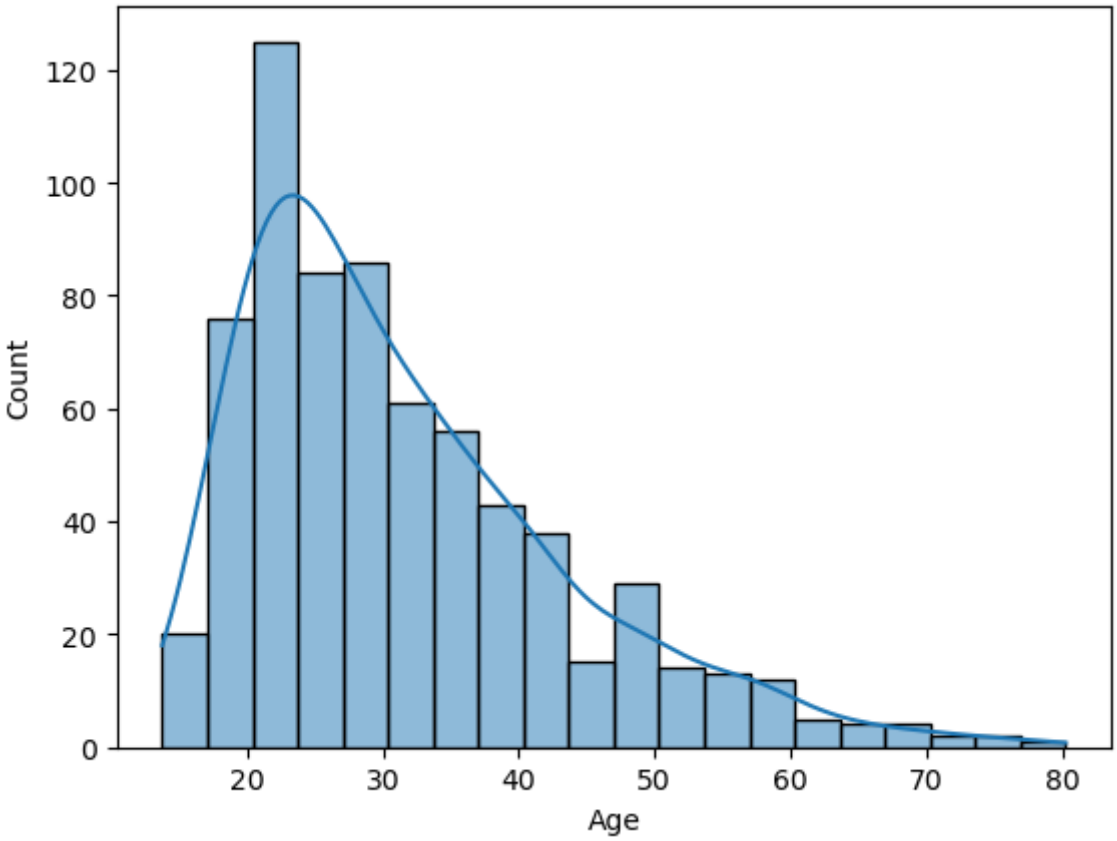
In [6]:
```
card_approval_df[['Age','Debt','YearsEmployed','CreditScore','Income']].describe()
```

Out[6]:

	Age	Debt	YearsEmployed	CreditScore	Income
count	690.000000	690.000000	690.000000	690.00000	690.000000
mean	31.514116	4.758725	2.223406	2.40000	1017.385507
std	11.860245	4.978163	3.346513	4.86294	5210.102598
min	13.750000	0.000000	0.000000	0.00000	0.000000
25%	22.670000	1.000000	0.165000	0.00000	0.000000
50%	28.460000	2.750000	1.000000	0.00000	5.000000
75%	37.707500	7.207500	2.625000	3.00000	395.500000
max	80.250000	28.000000	28.500000	67.00000	100000.000000

In [7]:
```
sns.histplot(card_approval_df.Age,kde=True)
```

Out[7]:
```
<AxesSubplot:xlabel='Age', ylabel='Count'>
```

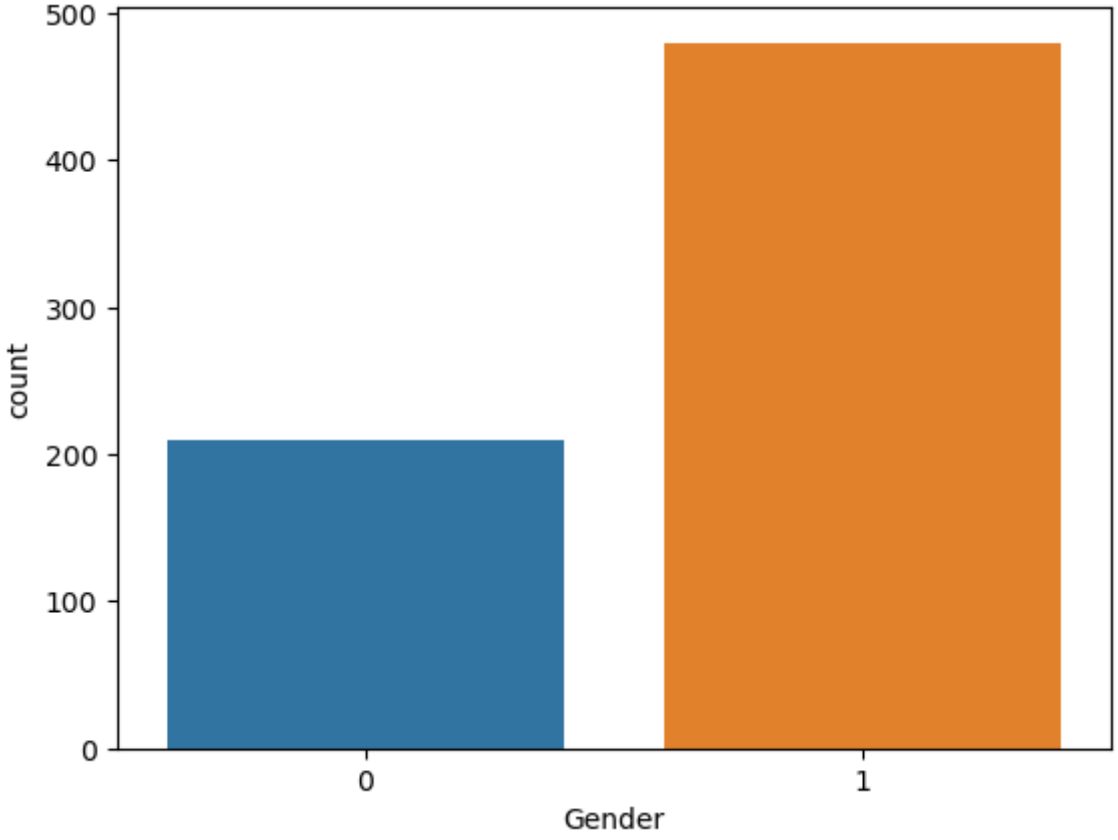


Univariate Analysis of categorical Variables

In [8]:
```
sns.countplot(card_approval_df.Gender)
```

```
C:\Users\gptkgf\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarni
ng: Pass the following variable as a keyword arg: x. From version 0.12, the only v
alid positional argument will be `data`, and passing other arguments without an ex
plicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

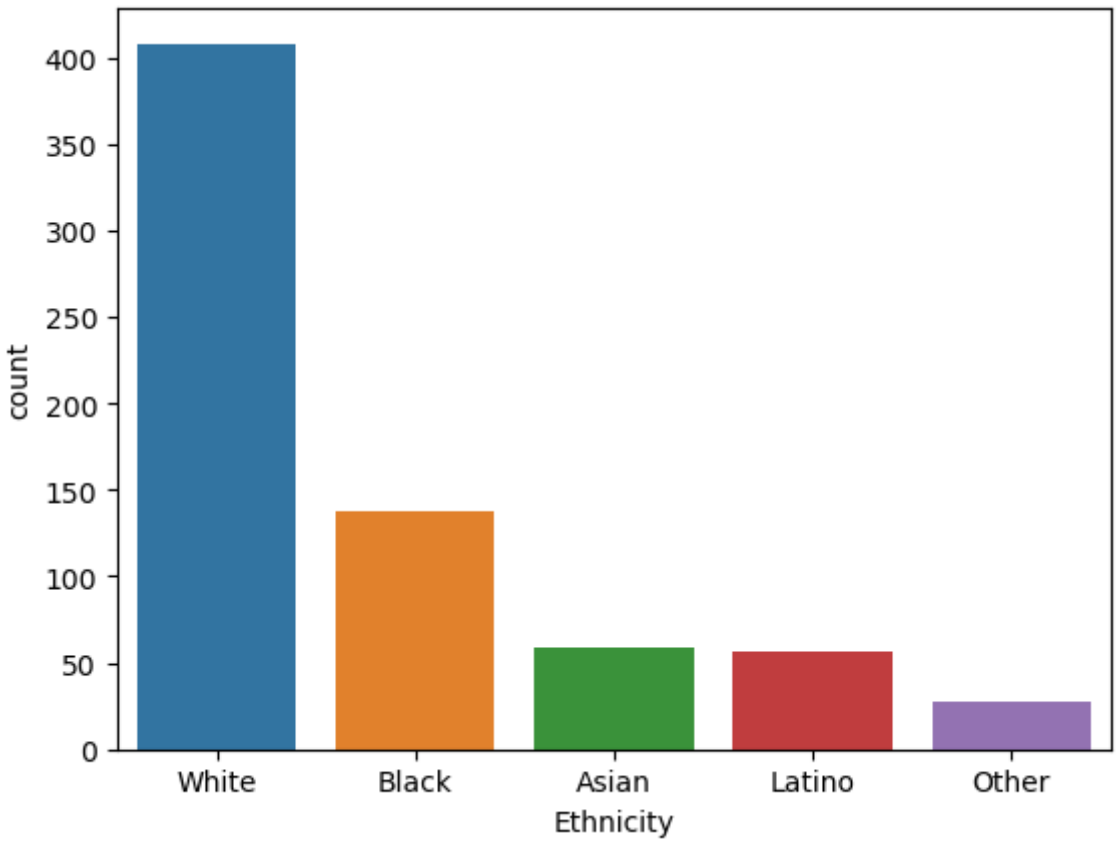
Out[8]:
```
<AxesSubplot:xlabel='Gender', ylabel='count'>
```



In [9]:
```
sns.countplot(card_approval_df.Ethnicity)
```

```
C:\Users\gptkgf\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

Out[9]:
```
<AxesSubplot:xlabel='Ethnicity', ylabel='count'>
```



Bivariate analysisof continuous variable

In [10]:
```
card_approval_df[['Age','Debt','YearsEmployed','CreditScore','Income']].corr()
```

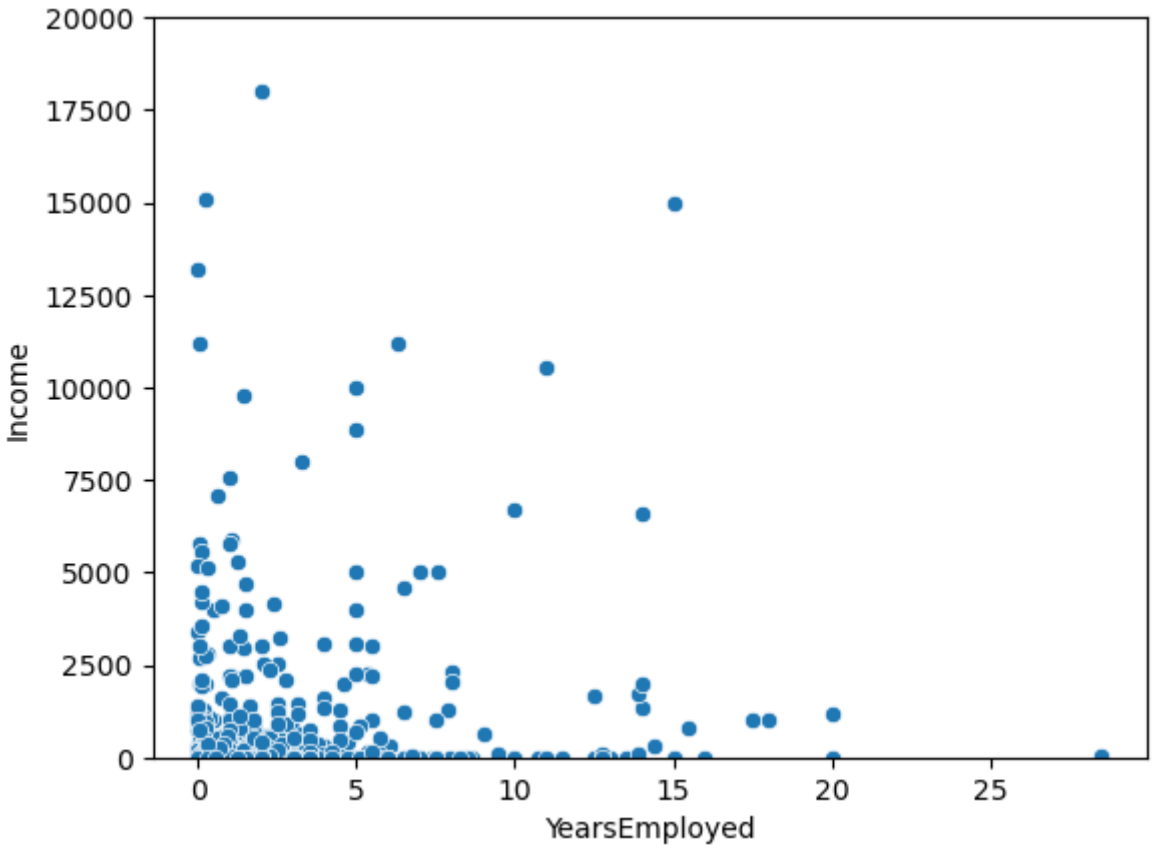
Out[10]:

	Age	Debt	YearsEmployed	CreditScore	Income
Age	1.000000	0.202177	0.391464	0.187327	0.018719
Debt	0.202177	1.000000	0.298902	0.271207	0.123121
YearsEmployed	0.391464	0.298902	1.000000	0.322330	0.051345
CreditScore	0.187327	0.271207	0.322330	1.000000	0.063692
Income	0.018719	0.123121	0.051345	0.063692	1.000000

In [12]:
```
sns.scatterplot(card_approval_df.YearsEmployed,card_approval_df.Income)
plt.ylim(0,20000)
```

```
C:\Users\gptkgf\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarni
ng: Pass the following variables as keyword args: x, y. From version 0.12, the onl
y valid positional argument will be `data`, and passing other arguments without an
explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

Out[12]:
```
(0.0, 20000.0)
```



Bivariate Analysis of Categorical Variables vs Continuous Variables

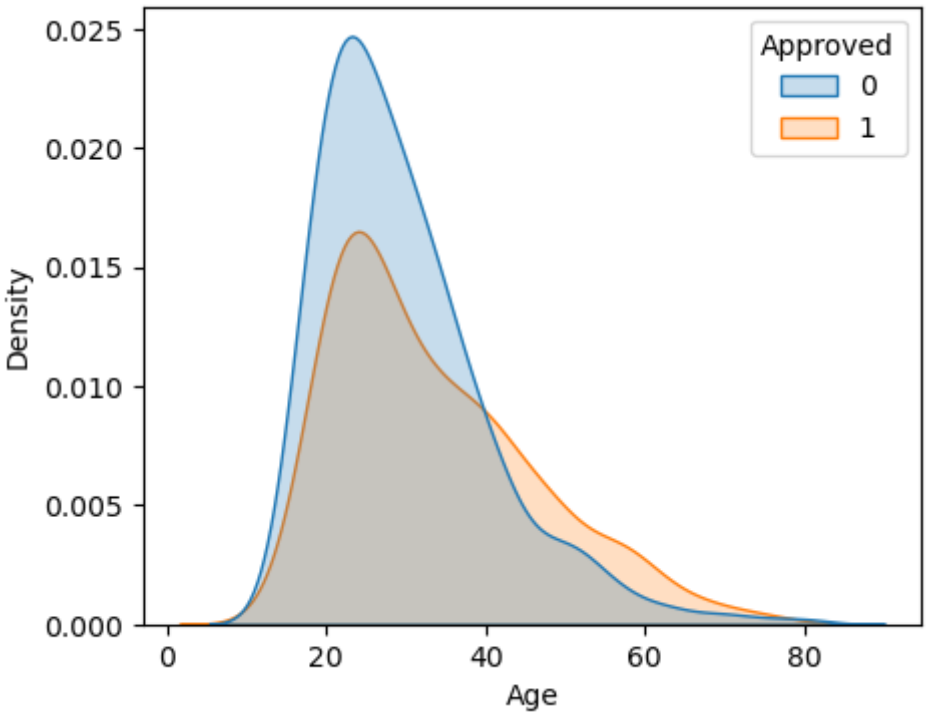
In [13]:
```
card_approval_df.groupby(by='Approved').agg('mean')[['Age','Debt','YearsEmployed',
```

Out[13]:

Approved	Age	Debt	YearsEmployed	CreditScore	Income
0	29.773029	3.839948	1.257924	0.631854	198.605744
1	33.686221	5.904951	3.427899	4.605863	2038.859935

In [19]:
```
plt.figure(figsize=(5,4))
sns.kdeplot(data=card_approval_df,x='Age',hue='Approved',fill=True)
```

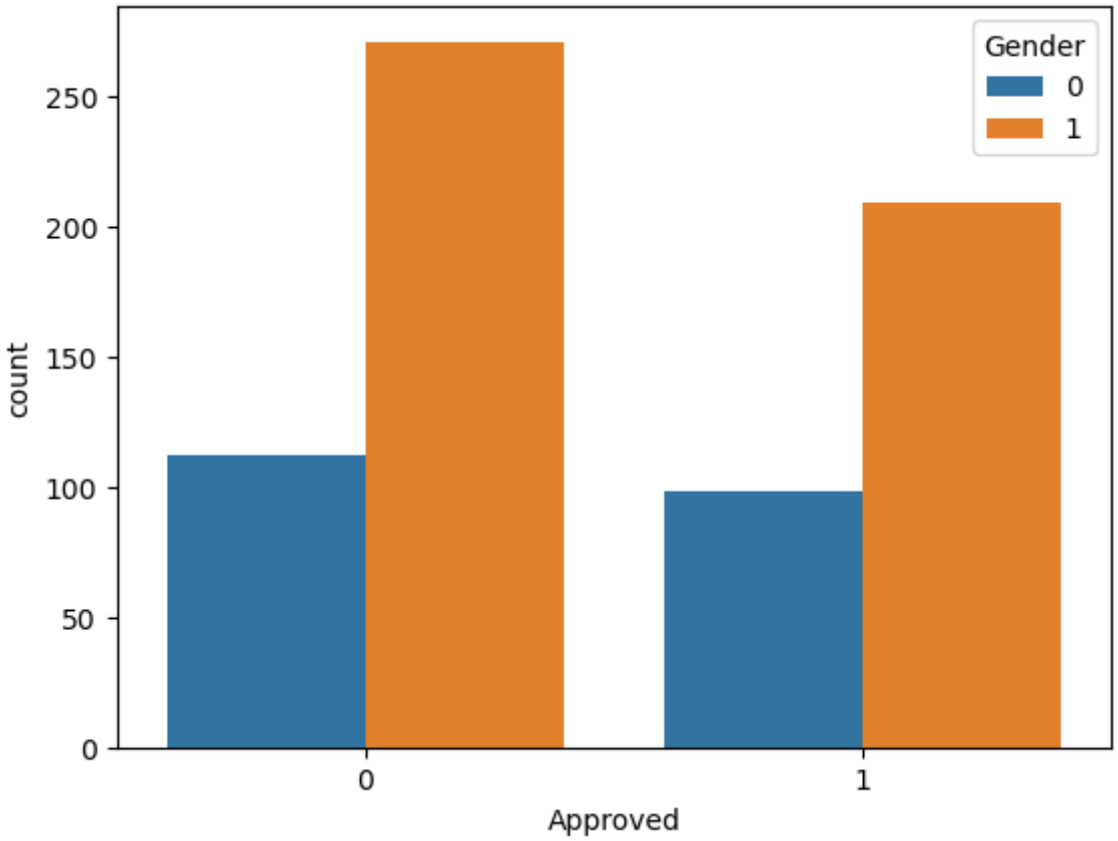
Out[19]:
```
<AxesSubplot:xlabel='Age', ylabel='Density'>
```



Bivariate Analysis of Categorical Variables vs Categorical Variables

In [20]:
```
sns.countplot(data=card_approval_df,x='Approved',hue='Gender')
```

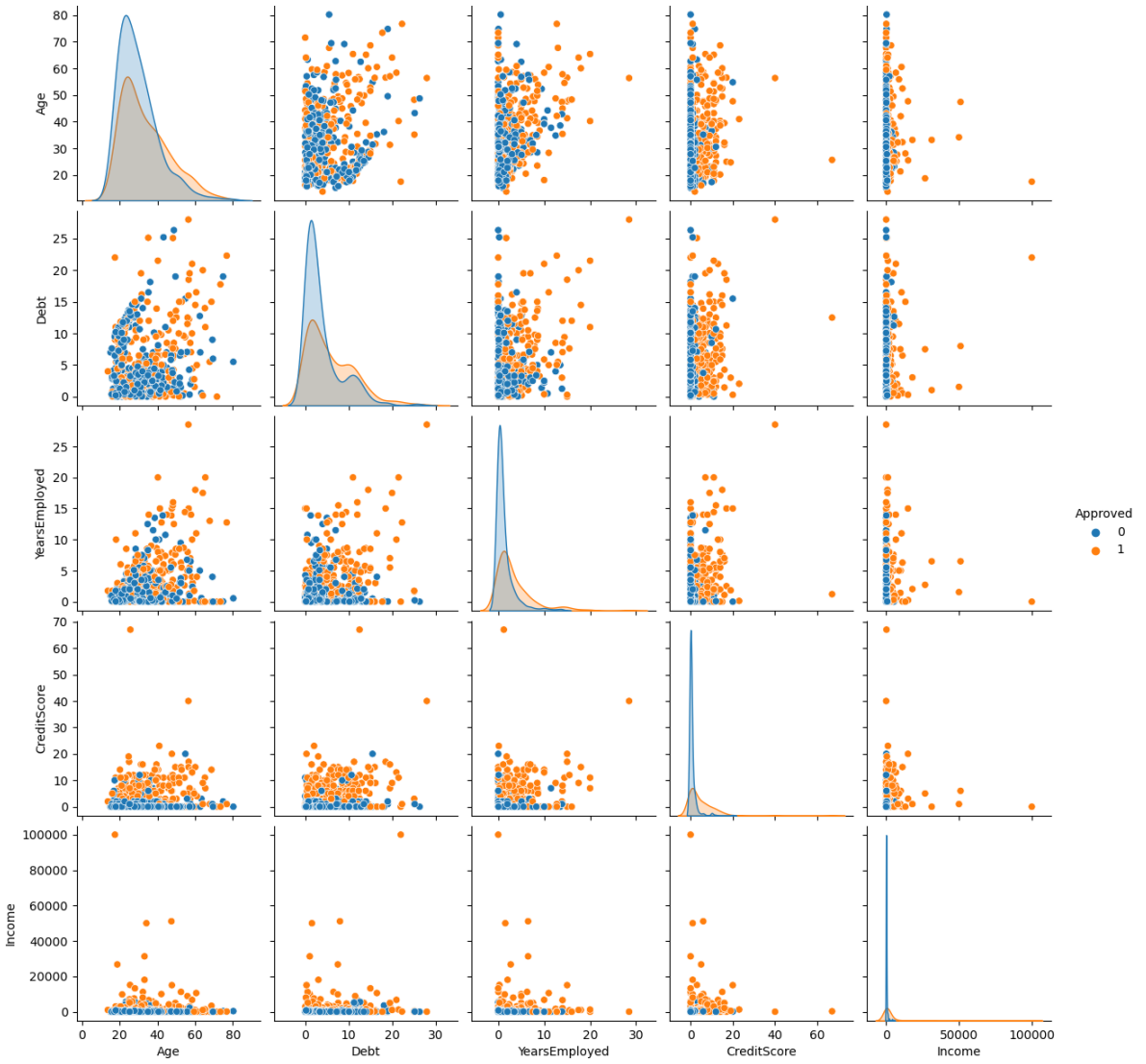
Out[20]:
```
<AxesSubplot:xlabel='Approved', ylabel='count'>
```



Multivariate Analysis

In [21]:
```
sns.pairplot(data=card_approval_df[['Age','Debt','YearsEmployed','CreditScore','In
```

Out[21]:
```
<seaborn.axisgrid.PairGrid at 0x23b342f37c0>
```



In [ ]: