MIMIC MACHINE LEARNING

In [1]:
```
from sklearn.datasets import load_boston
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression

import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

import statsmodels.api as sm
from statsmodels.stats.outliers_influence import variance_inflation_factor

#import boston_valuation as val

%matplotlib inline
```

In [17]:
```
mimic_data=pd.read_csv("final_mimic.csv",index_col=0)
```

In [18]:
```
mimic_data.head(5)
```

Out[18]:

	person_id	age	gender	Age_T2D_First	Age_AD_First	T2D_OR_AD_FIRST
0	148	78	F	NaN	NaN	NaN
1	463	62	F	NaN	NaN	NaN
2	471	75	F	NaN	NaN	NaN
3	833	0	M	NaN	NaN	NaN
4	1088	68	M	NaN	NaN	NaN

In [19]:
```
mimic_data.tail(5)
```

Out[19]:

	person_id	age	gender	Age_T2D_First	Age_AD_First	T2D_OR_AD_FIRST
52638	96746	75	F	NaN	NaN	NaN
52639	97592	35	M	NaN	NaN	NaN
52640	98417	78	M	NaN	NaN	NaN
52641	99286	57	F	NaN	NaN	NaN
52642	99564	62	M	NaN	NaN	NaN

In [46]:
```
columns = ['person_id']
mimic_data.drop(columns, inplace=True, axis=1)
```

In [47]:
```
mimic_data.shape
```

Out[47]:
```
(52643, 5)
```

In [48]:
```
mimic_data.count()
```

Out[48]:
```
age                52643
gender             52643
Age_T2D_First      12184
Age_AD_First         600
T2D_OR_AD_FIRST      168
dtype: int64
```

In [49]:
```
pd.isnull(mimic_data)
```

Out[49]:

	age	gender	Age_T2D_First	Age_AD_First	T2D_OR_AD_FIRST
0	False	False	True	True	True
1	False	False	True	True	True
2	False	False	True	True	True
3	False	False	True	True	True
4	False	False	True	True	True
...	...	...	...	...	...
52638	False	False	True	True	True
52639	False	False	True	True	True
52640	False	False	True	True	True
52641	False	False	True	True	True
52642	False	False	True	True	True

52643 rows × 5 columns

In [50]:
```
mimic_data.isnull().sum()
```

Out[50]:
```
age                    0
gender                 0
Age_T2D_First      40459
Age_AD_First       52043
T2D_OR_AD_FIRST    52475
dtype: int64
```

In [51]:
```
mimic_data=mimic_data.fillna(" ")
```

In [52]:
```
mimic_data.isnull().sum()
```

Out[52]:
```
age                0
gender             0
Age_T2D_First      0
Age_AD_First       0
T2D_OR_AD_FIRST    0
dtype: int64
```

In [53]:
```
mimic_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 52643 entries, 0 to 52642
Data columns (total 5 columns):
 #   Column           Non-Null Count  Dtype 
---  ------           --------------  ----- 
 0   age              52643 non-null  int64 
 1   gender           52643 non-null  int32 
 2   Age_T2D_First    52643 non-null  object
 3   Age_AD_First     52643 non-null  object
 4   T2D_OR_AD_FIRST  52643 non-null  object
dtypes: int32(1), int64(1), object(3)
memory usage: 2.2+ MB
```

In [54]:
```
mimic_data.count()
```

Out[54]:
```
age                52643
gender             52643
Age_T2D_First      52643
Age_AD_First       52643
T2D_OR_AD_FIRST    52643
dtype: int64
```

In [55]:
```
mimic_data.tail(100)
```

Out[55]:

	age	gender	Age_T2D_First	Age_AD_First	T2D_OR_AD_FIRST
52543	48	1			
52544	70	1			
52545	75	1	75.871		
52546	60	1	60.807	60.807	0
52547	84	0			
...	...	...	...	...	...
52638	75	0			
52639	35	1			
52640	78	1			
52641	57	0			
52642	62	1			

100 rows × 5 columns

In [59]:
```
# Import label encoder
from sklearn import preprocessing

# label_encoder object knows how to understand word labels.
label_encoder = preprocessing.LabelEncoder()

# Encode labels in columns
mimic_data['gender']= label_encoder.fit_transform(mimic_data['gender'])

mimic_data['gender'].unique()
```

Out[59]:
```
array([0, 1], dtype=int64)
```

In [60]:
```
mimic_data["Age_T2D_First"] = pd.to_numeric(mimic_data.Age_T2D_First,errors='
```

In [61]:
```
mimic_data["Age_AD_First"] = pd.to_numeric(mimic_data.Age_AD_First,errors='co
```

In [62]:
```
mimic_data["T2D_OR_AD_FIRST"] = pd.to_numeric(mimic_data.T2D_OR_AD_FIRST,erro
```

In [63]:
```
mimic_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 52643 entries, 0 to 52642
Data columns (total 5 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   age              52643 non-null  int64  
 1   gender           52643 non-null  int64  
 2   Age_T2D_First    12184 non-null  float64
 3   Age_AD_First     600 non-null    float64
 4   T2D_OR_AD_FIRST  168 non-null    float64
dtypes: float64(3), int64(2)
memory usage: 2.4 MB
```

In [64]:
```
mimic_data.describe()
```

Out[64]:

	age	gender	Age_T2D_First	Age_AD_First	T2D_OR_AD_FIRST
count	52643.000000	52643.00000	12184.000000	600.000000	168.000000
mean	63.008909	0.55960	76.365733	133.407660	-0.682845
std	57.034501	0.49644	48.411772	94.835686	1.820989
min	0.000000	0.00000	16.073000	47.651000	-7.951000
25%	42.000000	0.00000	58.715000	78.728000	0.000000
50%	61.000000	1.00000	68.430500	84.253000	0.000000
75%	75.000000	1.00000	77.778250	88.935000	0.000000
max	311.000000	1.00000	308.481000	306.611000	1.046000

In [65]:
```
mimic_data.corr() # Pearson Correlation Coefficients
```

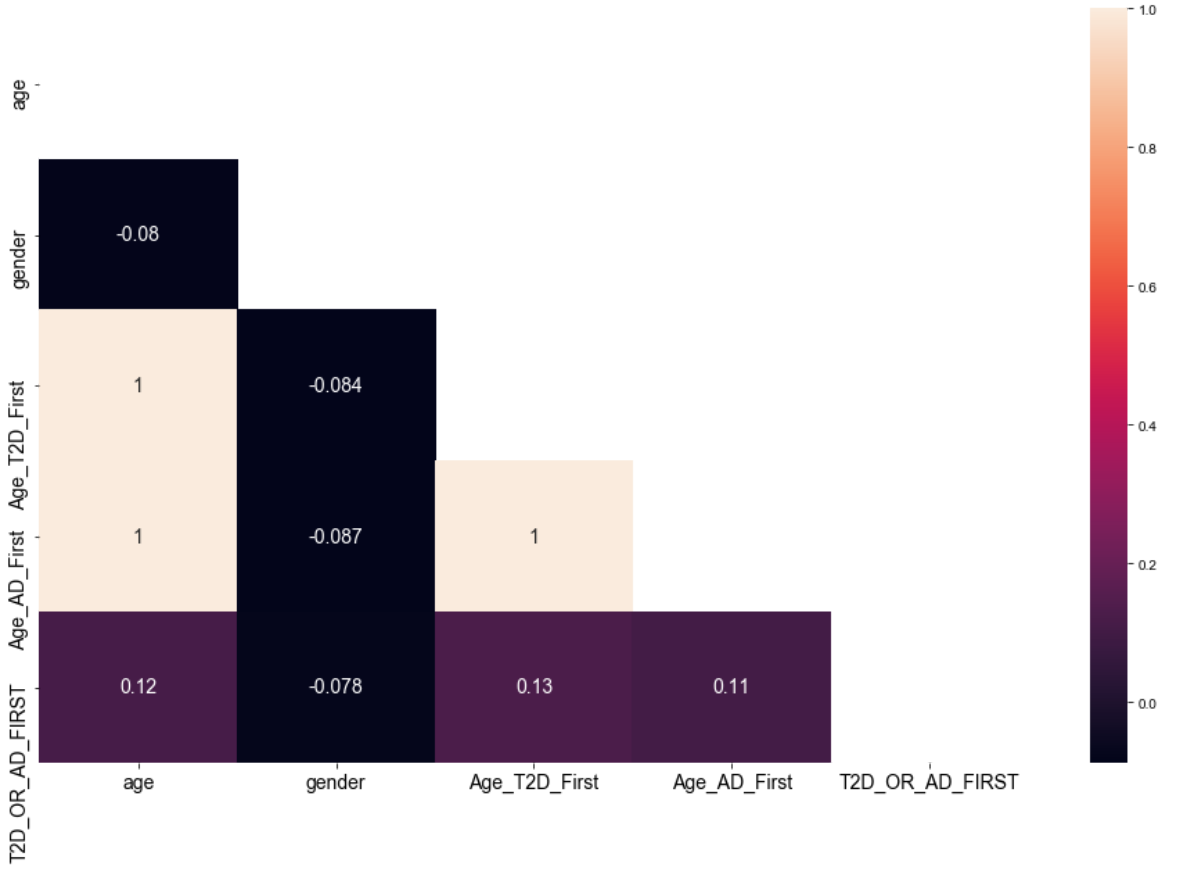
Out[65]:

	age	gender	Age_T2D_First	Age_AD_First	T2D_OR_AD_FIRST
age	1.000000	-0.079924	0.999393	0.999884	0.116262
gender	-0.079924	1.000000	-0.083765	-0.087095	-0.078046
Age_T2D_First	0.999393	-0.083765	1.000000	0.999777	0.127750
Age_AD_First	0.999884	-0.087095	0.999777	1.000000	0.106758
T2D_OR_AD_FIRST	0.116262	-0.078046	0.127750	0.106758	1.000000

In [75]:
```
mask = np.zeros_like(mimic_data.corr())
triangle_indices = np.triu_indices_from(mask)
mask[triangle_indices] = True
mask
```

Out[75]:
```
array([[1., 1., 1., 1., 1.],
       [0., 1., 1., 1., 1.],
       [0., 0., 1., 1., 1.],
       [0., 0., 0., 1., 1.],
       [0., 0., 0., 0., 1.]])
```

In [76]:
```
plt.figure(figsize=(16,10))
sns.heatmap(mimic_data.corr(), mask=mask, annot=True, annot_kws={"size": 14})
sns.set_style('white')
plt.xticks(fontsize=14)
plt.yticks(fontsize=14)
plt.show()
```



In [66]:
```
mimic_data.tail(100)
```

Out[66]:

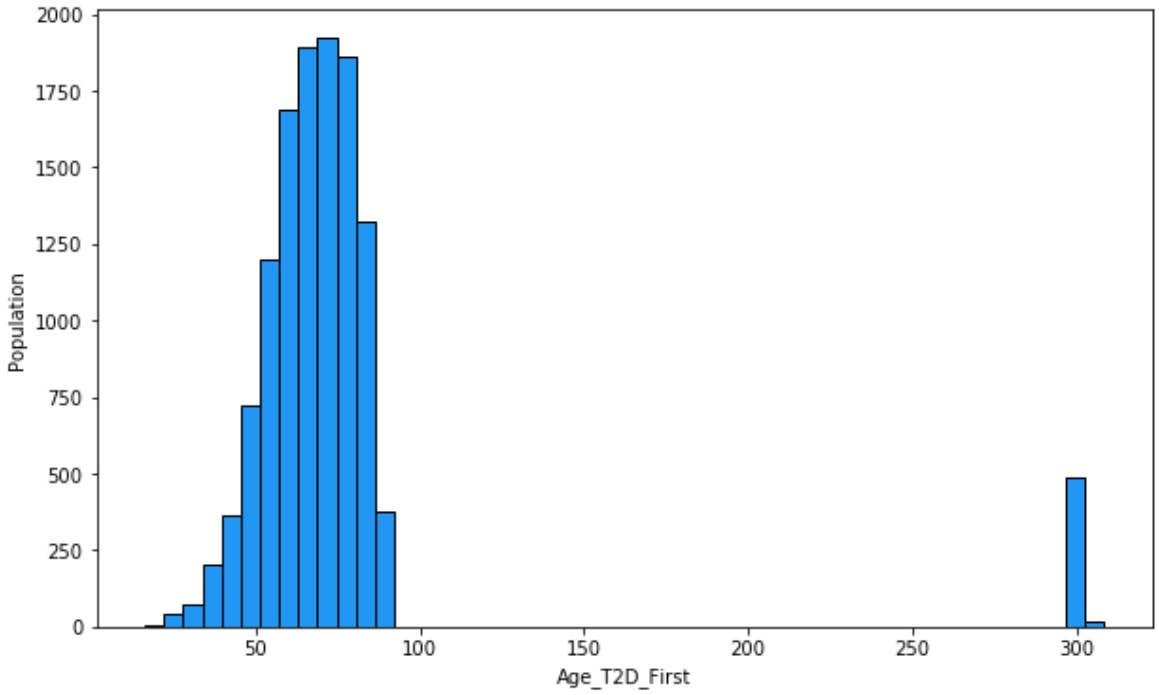
	age	gender	Age_T2D_First	Age_AD_First	T2D_OR_AD_FIRST
52543	48	1	NaN	NaN	NaN
52544	70	1	NaN	NaN	NaN
52545	75	1	75.871	NaN	NaN
52546	60	1	60.807	60.807	0.0
52547	84	0	NaN	NaN	NaN
...	...	...	...	...	...
52638	75	0	NaN	NaN	NaN
52639	35	1	NaN	NaN	NaN
52640	78	1	NaN	NaN	NaN
52641	57	0	NaN	NaN	NaN
52642	62	1	NaN	NaN	NaN

100 rows × 5 columns

Visualising Data - Histograms, Distributions and Bar Charts

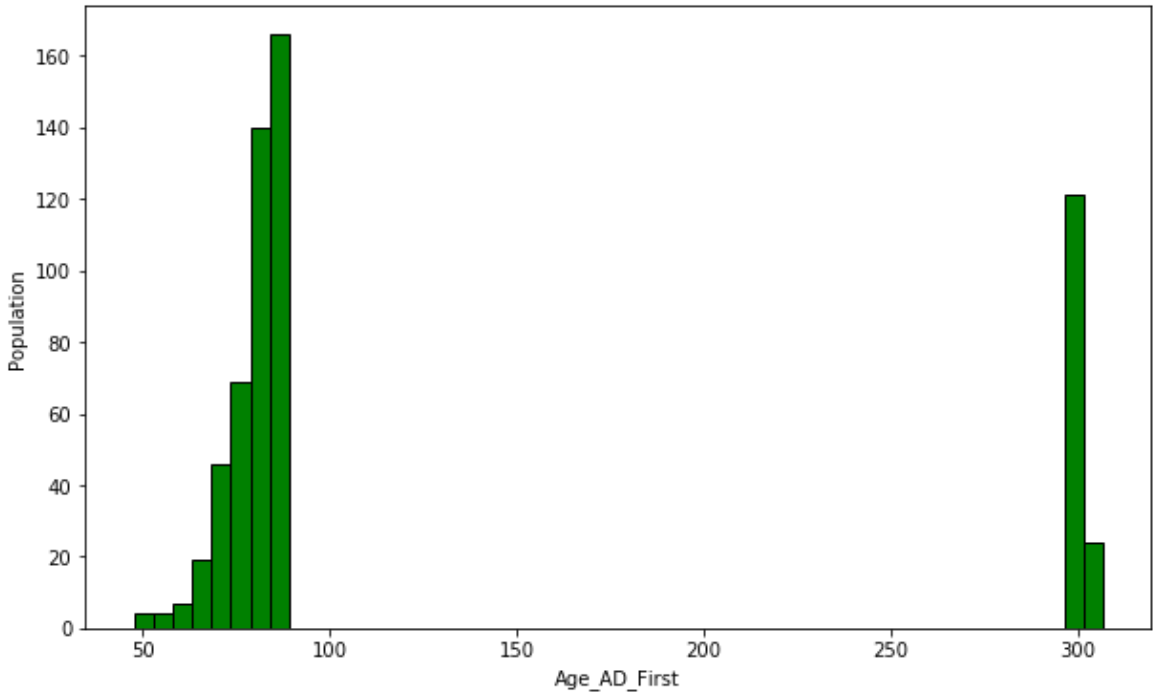
In [67]:
```
plt.figure(figsize=(10, 6))
plt.hist(mimic_data['Age_T2D_First'], bins=50, ec='black', color='#2196f3')
plt.xlabel('Age_T2D_First')
plt.ylabel('Population')
plt.show()
```

```
C:\Users\mayam\anaconda3\lib\site-packages\numpy\lib\histograms.py:839: RuntimeWarning: invalid value encountered in greater_equal
  keep = (tmp_a >= first_edge)
C:\Users\mayam\anaconda3\lib\site-packages\numpy\lib\histograms.py:840: RuntimeWarning: invalid value encountered in less_equal
  keep &= (tmp_a <= last_edge)
```



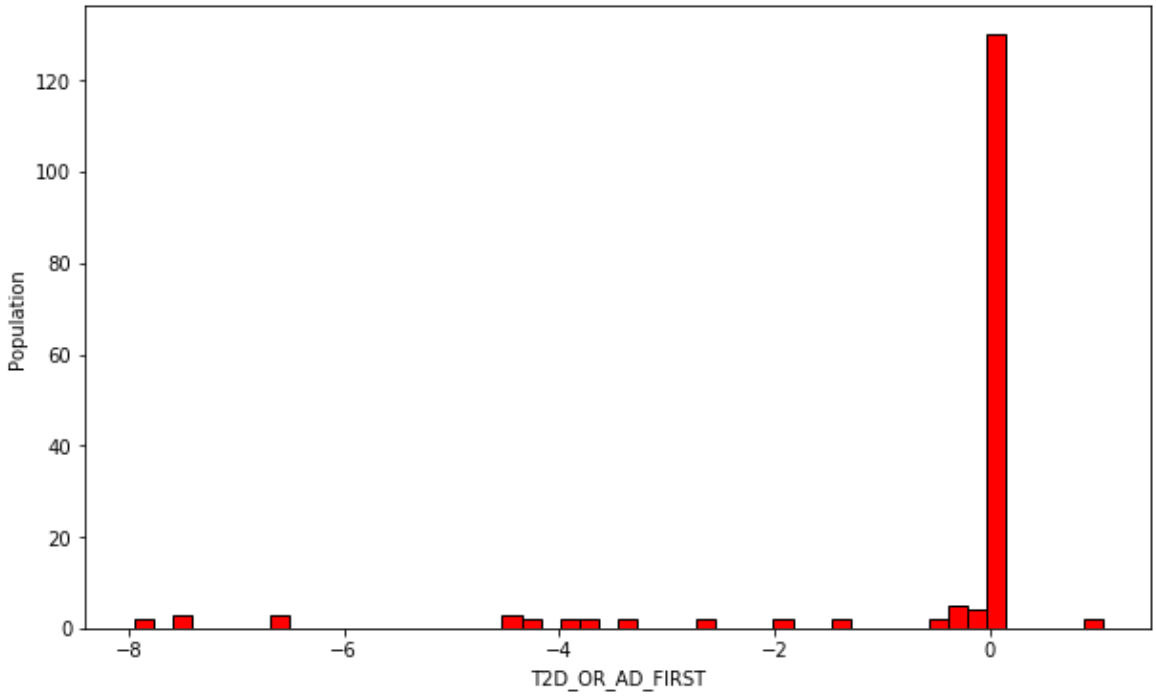
In [68]:

```
plt.figure(figsize=(10, 6))
plt.hist(mimic_data['Age_AD_First'], bins=50, ec='black', color='green')
plt.xlabel('Age_AD_First')
plt.ylabel('Population')
plt.show()
```

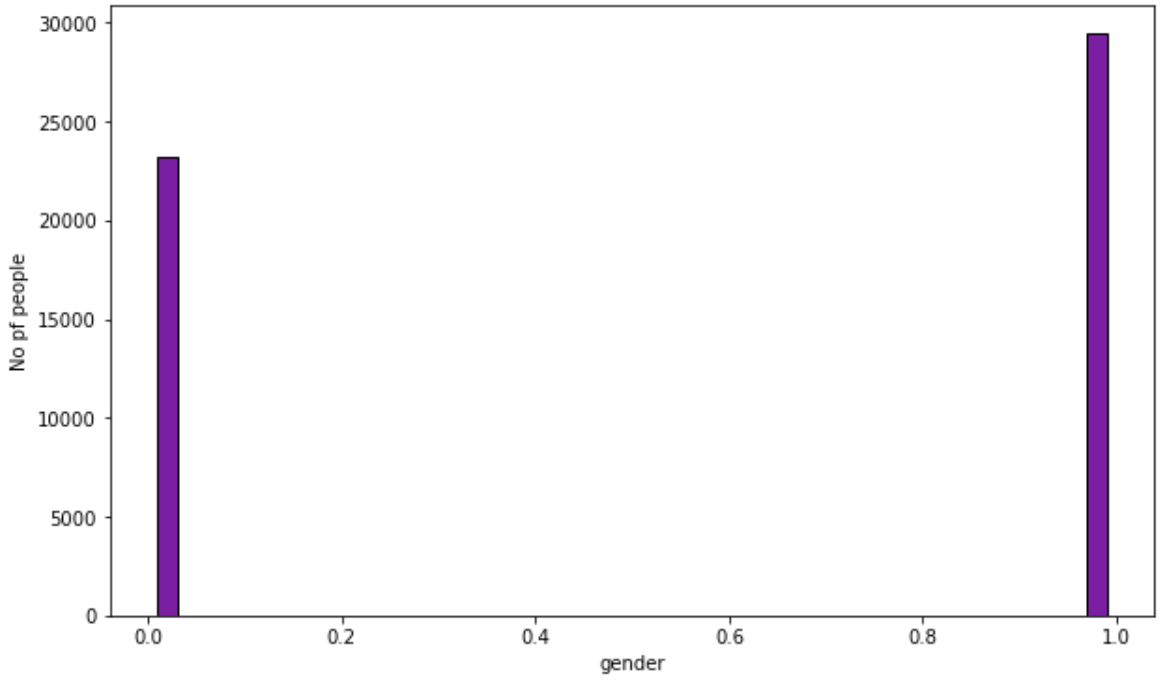


In [69]:

```
plt.figure(figsize=(10, 6))
plt.hist(mimic_data['T2D_OR_AD_FIRST'], bins=50, ec='black', color='red')
plt.xlabel('T2D_OR_AD_FIRST')
plt.ylabel('Population')
plt.show()
```



In [70]:
```
plt.figure(figsize=(10, 6))
plt.hist(mimic_data['gender'], bins=24, ec='black', color='#7b1fa2', rwidth=0
plt.xlabel('gender')
plt.ylabel('No pf people')
plt.show()
```



In [71]:
```
mimic_data.columns
```

Out[71]:
```
Index(['age', 'gender', 'Age_T2D_First', 'Age_AD_First', 'T2D_OR_AD_FIRS
T'], dtype='object')
```

In [72]:
```
mimic_data['Age_T2D_First'].describe()
```

Out[72]:
```
count    12184.000000
mean        76.365733
std         48.411772
min         16.073000
25%         58.715000
50%         68.430500
75%         77.778250
max        308.481000
Name: Age_T2D_First, dtype: float64
```

In [73]:
```
mimic_data['Age_AD_First'].describe()
```

Out[73]:
```
count    600.000000
mean     133.407660
std       94.835686
min       47.651000
25%       78.728000
50%       84.253000
75%       88.935000
max      306.611000
Name: Age_AD_First, dtype: float64
```

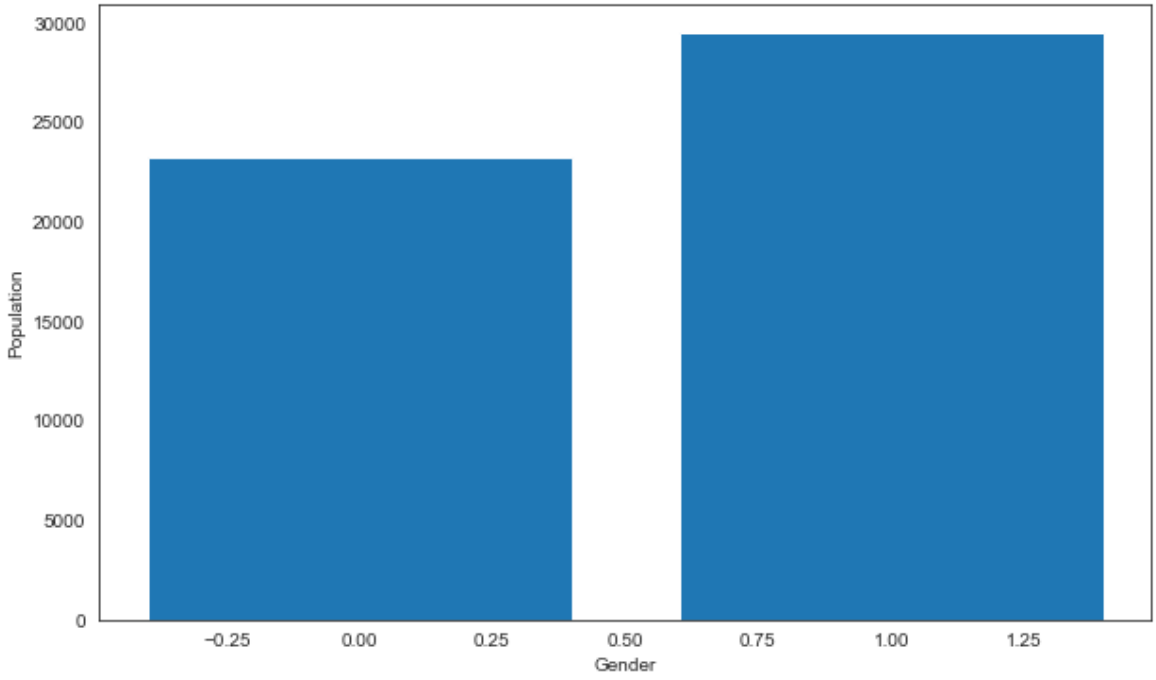
In [74]:
```
mimic_data['T2D_OR_AD_FIRST'].describe()
```

Out[74]:
```
count    168.000000
mean      -0.682845
std        1.820989
min       -7.951000
25%        0.000000
50%        0.000000
75%        0.000000
max        1.046000
Name: T2D_OR_AD_FIRST, dtype: float64
```

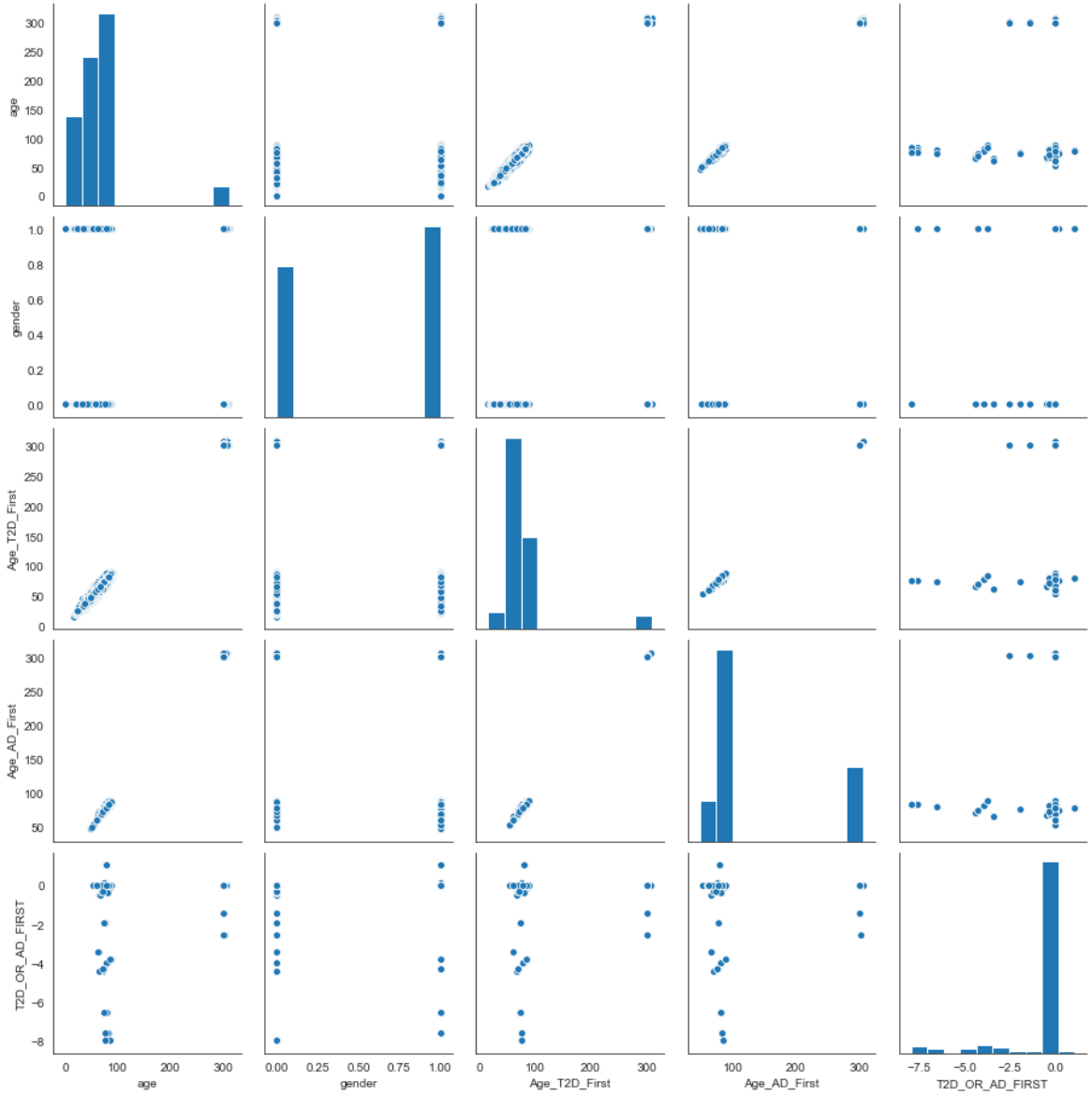
In [77]:
```
mimic_data['gender'].value_counts()
```

Out[77]:
```
1    29459
0    23184
Name: gender, dtype: int64
```

In [78]:
```
frequency = mimic_data['gender'].value_counts()
#type(frequency)
#frequency.index
#frequency.axes[0]
plt.figure(figsize=(10, 6))
plt.xlabel('Gender')
plt.ylabel('Population')
plt.bar(frequency.index, height=frequency)
plt.show()
```



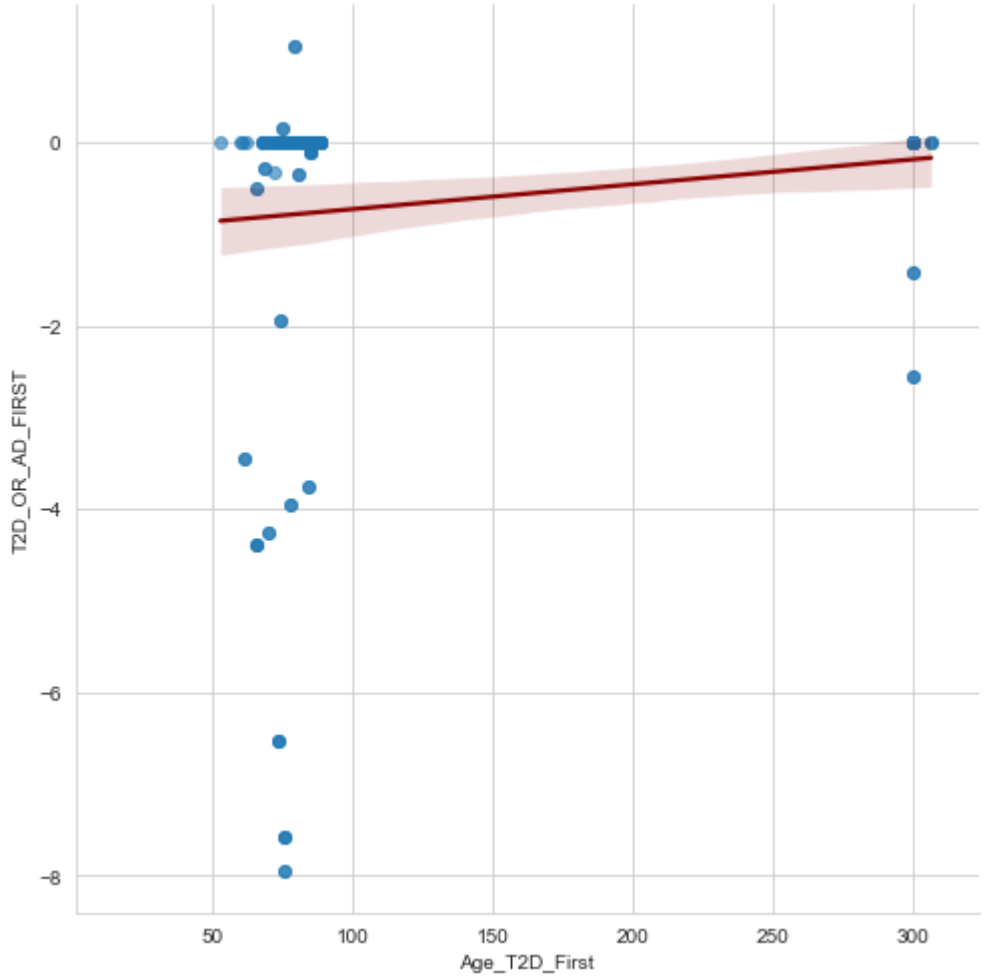
In [79]:
```
%%time

sns.pairplot(mimic_data)
plt.show()
```




```
Wall time: 7.44 s
```

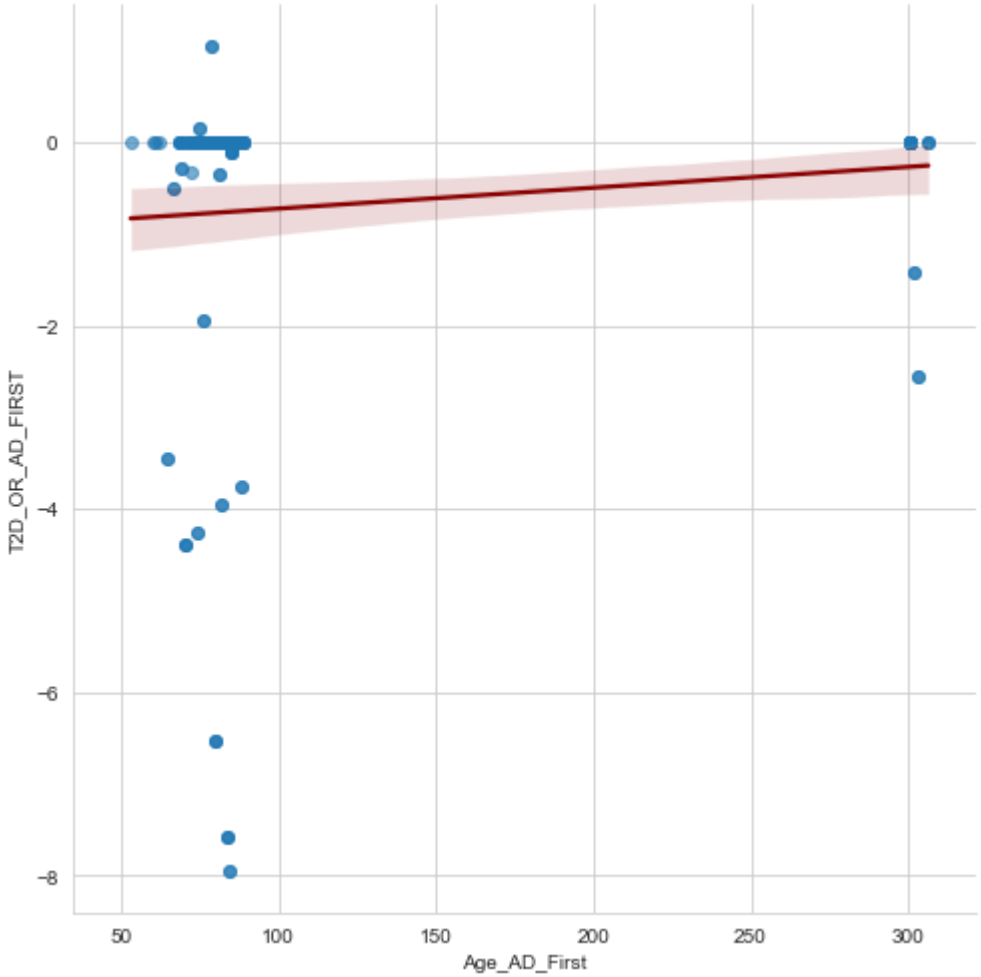
In [83]:
```
style('whitegrid')
ot(x='Age_T2D_First', y='T2D_OR_AD_FIRST', data=mimic_data, size=7, scatter_ku
()
```

```
C:\Users\mayam\anaconda3\lib\site-packages\seaborn\regression.py:574: UserW
arning: The `size` parameter has been renamed to `height`; please update yo
ur code.
  warnings.warn(msg, UserWarning)
```



In [82]:
```
_style('whitegrid')
lot(x='Age_AD_First', y='T2D_OR_AD_FIRST', data=mimic_data, size=7, scatter_kw
w()
```

```
C:\Users\mayam\anaconda3\lib\site-packages\seaborn\regression.py:574: UserW
arning: The `size` parameter has been renamed to `height`; please update yo
ur code.
  warnings.warn(msg, UserWarning)
```



In [ ]:
```

```