Hands-on Activity 1.1 Introduction to Machine Learning

Name: Viktor Angelo B. Apuyan

Section: CPE31S3

Intended Learning Outcomes (ILOs):

- Demonstrate how to use different toolsets in machine learning.
- Demonstrate how to import, manipulate and analyze data using pandas and numpy.
- Demonstrate how to visualize data in graphs using matplotlib and seaborn.

Data Exploration

In [2]:
```
#Importing libararies
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [19]:
```
#Importing the csv
diabetes = pd.read_csv('/content/diabetes.csv')
diabetes
```

Out[19]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFu
0	6	148	72	35	0	33.6	
1	1	85	66	29	0	26.6	
2	8	183	64	0	0	23.3	
3	1	89	66	23	94	28.1	
4	0	137	40	35	168	43.1	
...	...	...	...	...	...	...	
763	10	101	76	48	180	32.9	
764	2	122	70	27	0	36.8	
765	5	121	72	23	112	26.2	
766	1	126	60	0	0	30.1	
767	1	93	70	31	0	30.4	

768 rows × 9 columns

In [20]:
```
diabetes.head()
```

Out[20]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunc
0	6	148	72	35	0	33.6	0
1	1	85	66	29	0	26.6	0
2	8	183	64	0	0	23.3	0
3	1	89	66	23	94	28.1	0
4	0	137	40	35	168	43.1	2

In [21]:
```
df = pd.DataFrame(diabetes)
df
```

Out[21]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFu
0	6	148	72	35	0	33.6	
1	1	85	66	29	0	26.6	
2	8	183	64	0	0	23.3	
3	1	89	66	23	94	28.1	
4	0	137	40	35	168	43.1	
...	...	...	...	...	...	...	
763	10	101	76	48	180	32.9	
764	2	122	70	27	0	36.8	
765	5	121	72	23	112	26.2	
766	1	126	60	0	0	30.1	
767	1	93	70	31	0	30.4	

768 rows × 9 columns

In [22]:
```
df.describe()
```

Out[22]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	Dia
count	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	
mean	3.845052	120.894531	69.105469	20.536458	79.799479	31.992578	
std	3.369578	31.972618	19.355807	15.952218	115.244002	7.884160	
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
25%	1.000000	99.000000	62.000000	0.000000	0.000000	27.300000	
50%	3.000000	117.000000	72.000000	23.000000	30.500000	32.000000	
75%	6.000000	140.250000	80.000000	32.000000	127.250000	36.600000	
max	17.000000	199.000000	122.000000	99.000000	846.000000	67.100000	

Data pre-processing

In [23]:
```
#Finding null values
diabetes.isnull().sum()
```

Out[23]:

	0
Pregnancies	0
Glucose	0
BloodPressure	0
SkinThickness	0
Insulin	0
BMI	0
DiabetesPedigreeFunction	0
Age	0
Outcome	0

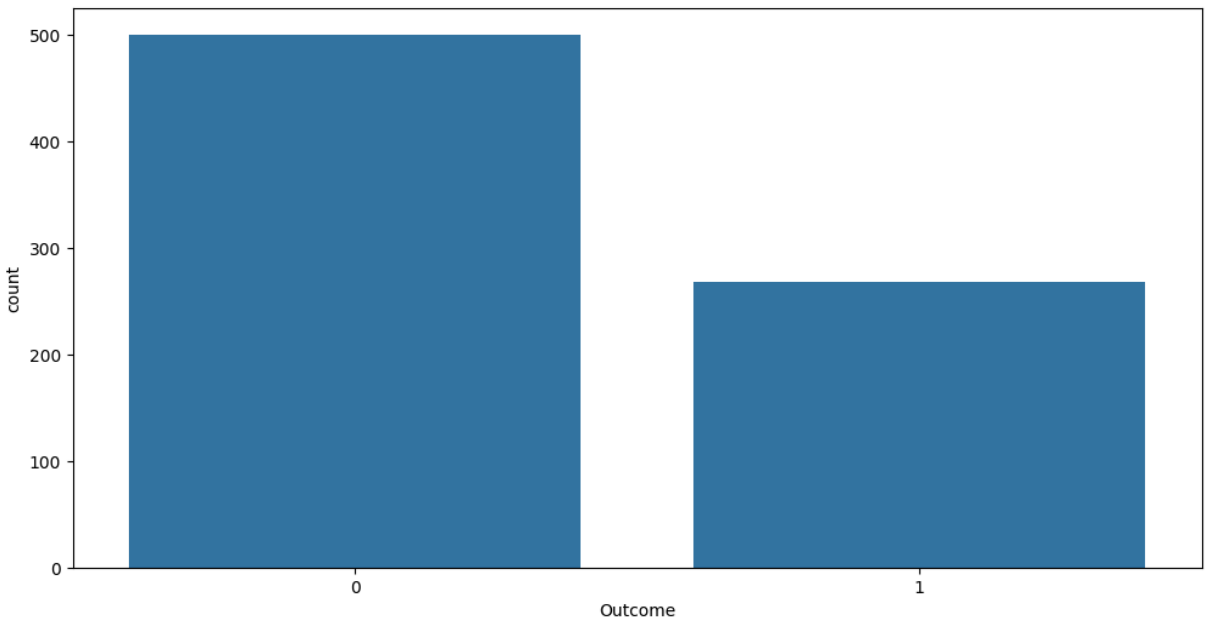
dtype: int64

In [24]:
```
#Finding duplicate values
diabetes = diabetes.duplicated().any()
diabetes
```

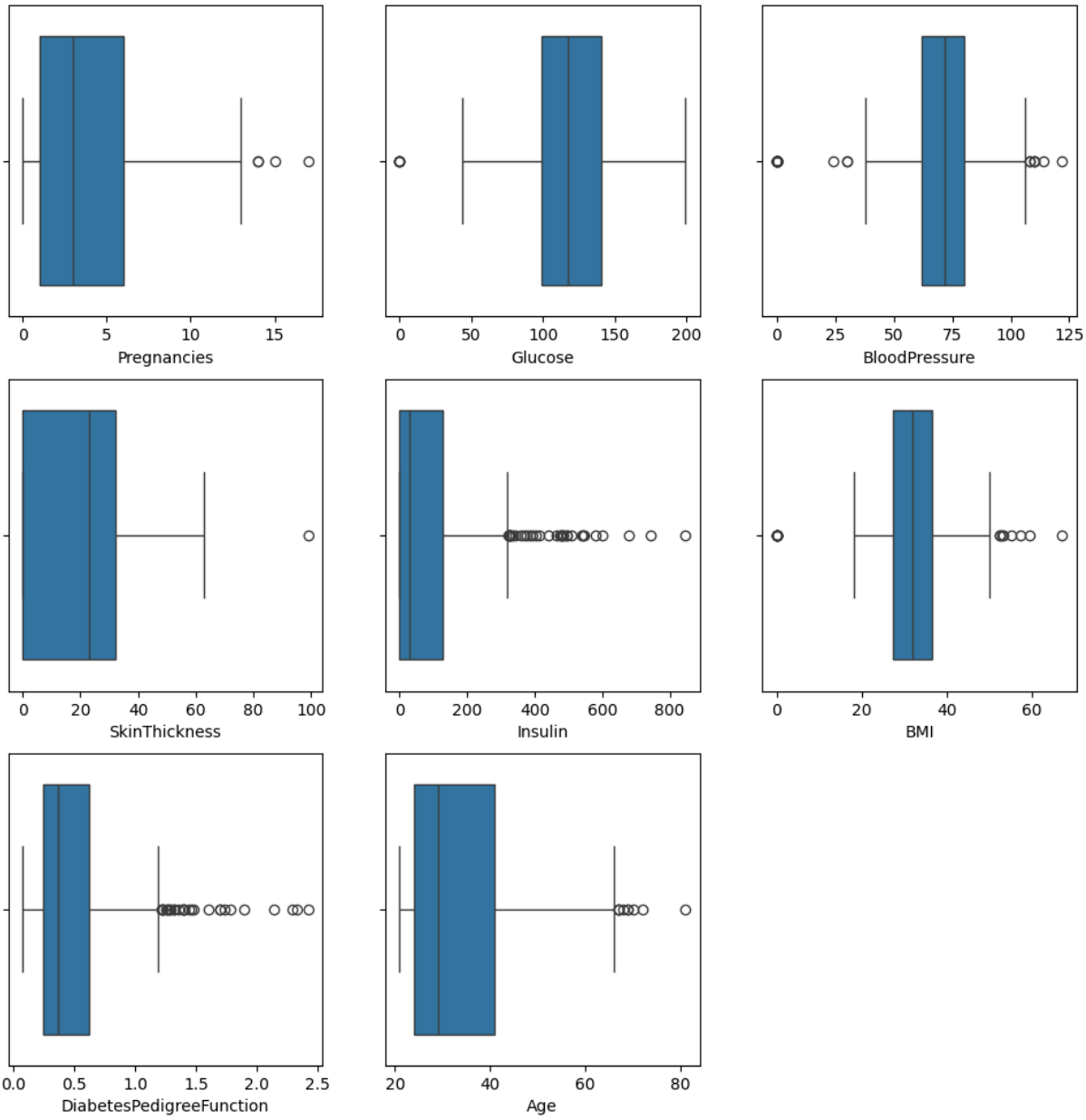
Out[24]: False

Data Visualization

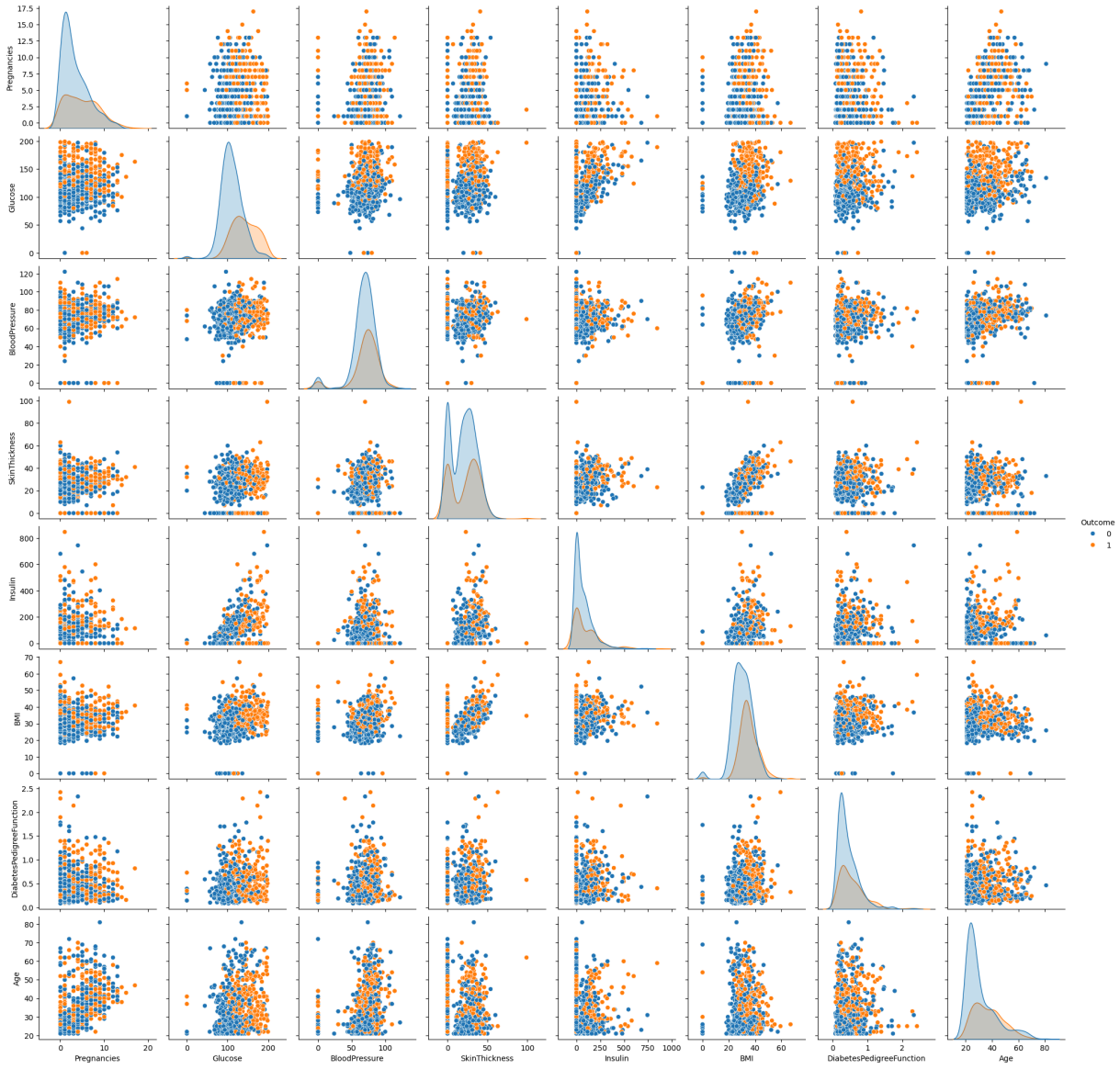
In [26]:
```
plt.figure(figsize = (12,6))
sns.countplot(x ='Outcome' , data = df)
plt.show()
```



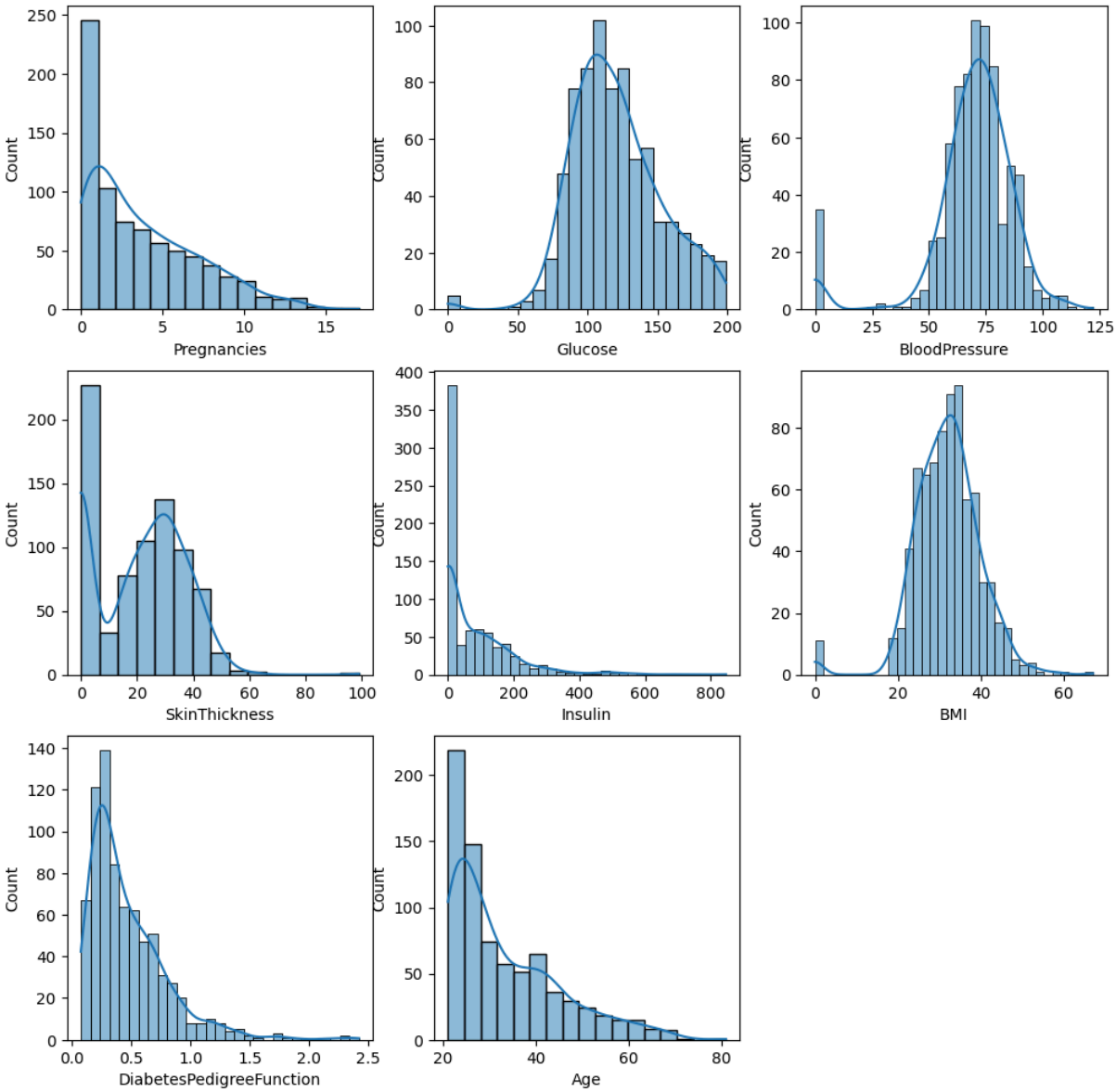
In [27]:
```
plt.figure(figsize = (12,12))
for i,col in enumerate(['Pregnancies','Glucose','BloodPressure','SkinThickness','In
   plt.subplot(3,3,i+1)
   sns.boxplot(x = col ,data = df)
```



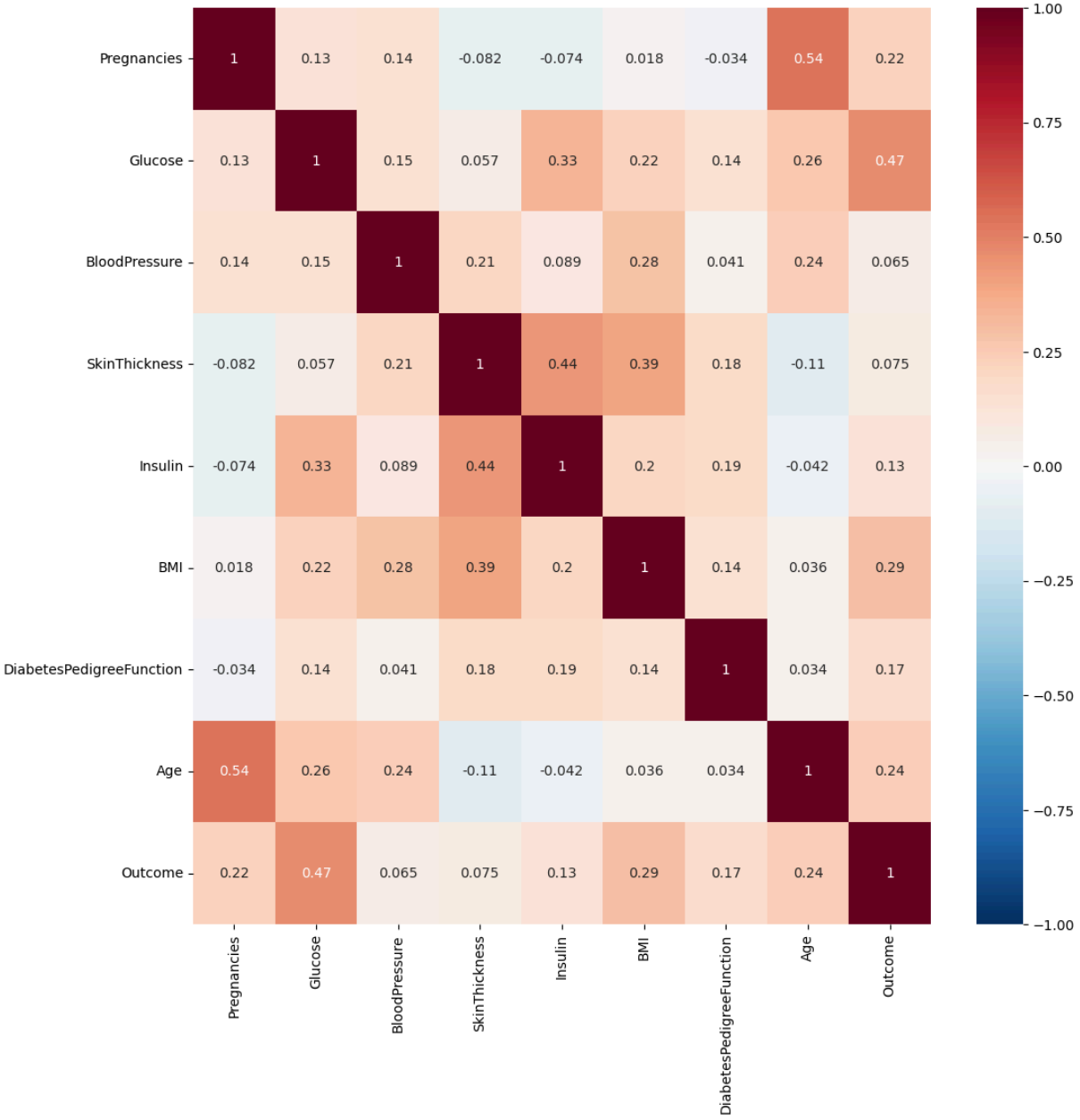
In [28]:
```
sns.pairplot(df , hue = 'Outcome')
plt.show()
```



In [29]:
```
#shows a histoplot of the features of the diabetes dataset
plt.figure(figsize = (12,12))
for i,col in enumerate(['Pregnancies','Glucose','BloodPressure','SkinThickness','In
    plt.subplot(3,3,i+1)
    sns.histplot(x = col ,data = df ,kde = True)
```



In [30]:
```
#shows a heatmap correlation between the features
plt.figure(figsize=(12, 12))
sns.heatmap(df.corr(), vmin=-1.0, center=0, cmap='RdBu_r', annot=True)
plt.show()
```


Pregnancies 1 0.13 0.14 -0.082 -0.074 0.018 -0.034 0.54 0.22
Glucose 0.13 1 0.15 0.057 0.33 0.22 0.14 0.26 0.47
BloodPressure 0.14 0.15 1 0.21 0.089 0.28 0.041 0.24 0.065
SkinThickness -0.082 0.057 0.21 1 0.44 0.39 0.18 -0.11 0.075
Insulin -0.074 0.33 0.089 0.44 1 0.2 0.19 -0.042 0.13
BMI 0.018 0.22 0.28 0.39 0.2 1 0.14 0.036 0.29
DiabetesPedigreeFunction -0.034 0.14 0.041 0.18 0.19 0.14 1 0.034 0.17
Age 0.54 0.26 0.24 -0.11 -0.042 0.036 0.034 1 0.24
Outcome 0.22 0.47 0.065 0.075 0.13 0.29 0.17 0.24 1
Pregnancies
Glucose
BloodPressure
SkinThickness
Insulin
BMI
DiabetesPedigreeFunction
Age
Outcome
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00