In [14]:
```
import numpy as nm
import matplotlib.pyplot as mtp
import pandas as pd
```

In [19]:
```
df = pd.read_csv('gender_class.csv')
print (df)
```

```
      long_hair  forehead_width_cm  forehead_height_cm  nose_wide  nose_long  \
0             1               11.8                 6.1          1          0
1             0               14.0                 5.4          0          0
2             0               11.8                 6.3          1          1
3             0               14.4                 6.1          0          1
4             1               13.5                 5.9          0          0
...         ...                ...                 ...        ...        ...
4996          1               13.6                 5.1          0          0
4997          1               11.9                 5.4          0          0
4998          1               12.9                 5.7          0          0
4999          1               13.2                 6.2          0          0
5000          1               15.4                 5.4          1          1

      lips_thin  distance_nose_to_lip_long  gender
0             1                         1       1
1             1                         0       0
2             1                         1       1
3             1                         1       1
4             0                         0       0
...         ...                       ...     ...
4996          0                         0       0
4997          0                         0       0
4998          0                         0       0
4999          0                         0       0
5000          1                         1       1

[5001 rows x 8 columns]
```

In [20]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5001 entries, 0 to 5000
Data columns (total 8 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   long_hair                  5001 non-null   int64
 1   forehead_width_cm          5001 non-null   float64
 2   forehead_height_cm         5001 non-null   float64
 3   nose_wide                  5001 non-null   int64
 4   nose_long                  5001 non-null   int64
 5   lips_thin                  5001 non-null   int64
 6   distance_nose_to_lip_long  5001 non-null   int64
 7   gender                     5001 non-null   int64
dtypes: float64(2), int64(6)
memory usage: 312.7 KB
```

In [29]:
```
cols=["nose_long"]
x=df[cols]
```

```
y=df.gender
```

In [30]:
```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test= train_test_split(x,y, test_size= 0.30, random_sta
```

In [31]:
```
from sklearn.linear_model import LogisticRegression
classifier= LogisticRegression(random_state=0)
classifier.fit(x_train, y_train)
```

Out[31]:
```
LogisticRegression
LogisticRegression(random_state=0)
```

In [32]:
```
y_pred= classifier.predict(x_test)
```

In [33]:
```
y_pred
```

Out[33]:
```
array([0, 1, 1, ..., 0, 0, 0])
```

In [34]:
```
from sklearn.metrics import confusion_matrix
cm= confusion_matrix(y_test,y_pred)
print(cm)
```

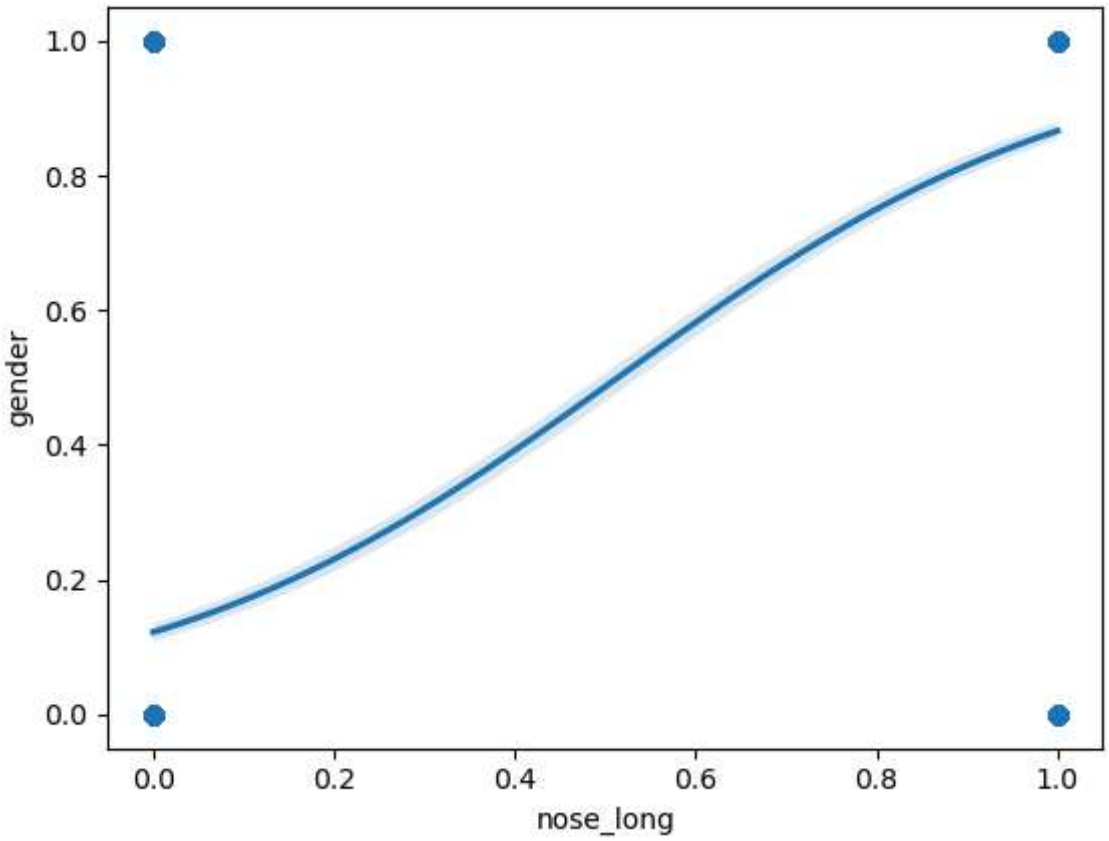
```
[[660 104]
 [ 87 650]]
```

In [35]:
```
from sklearn.metrics import accuracy_score
print(accuracy_score(y_test,y_pred))
```

```
0.8727514990006662
```

In [37]:
```
import seaborn as sns
data=df
sns.regplot(x="nose_long",y="gender",data=data,logistic=True)
```

Out[37]:
```
<Axes: xlabel='nose_long', ylabel='gender'>
```



In [36]:
```
from sklearn.metrics import classification_report
print(classification_report(y_test,y_pred))
```

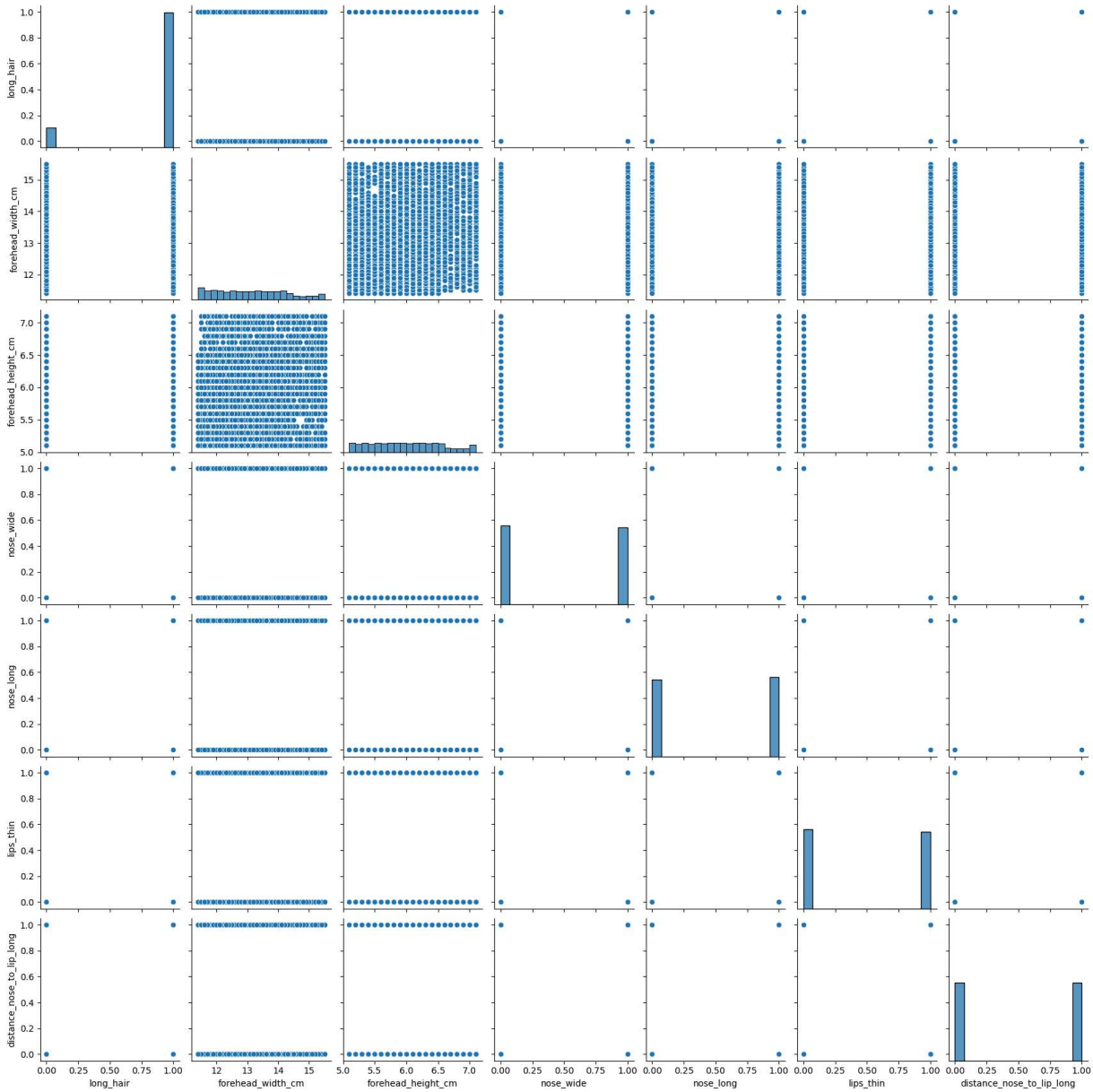
```
              precision    recall  f1-score   support

           0       0.88      0.86      0.87       764
           1       0.86      0.88      0.87       737

    accuracy                           0.87      1501
   macro avg       0.87      0.87      0.87      1501
weighted avg       0.87      0.87      0.87      1501
```

In [43]:
```
sns.pairplot(df.iloc[:,0:-1])
```

Out[43]: <seaborn.axisgrid.PairGrid at 0x7fd2b905c8b0>


long_hair
forehead_width_cm
forehead_height_cm
nose_wide
nose_long
lips_thin
distance_nose_to_lip_long

In [40]:
```
df.corr()
```

Out[40]:

	long_hair	forehead_width_cm	forehead_height_cm	nose_wide	n
long_hair	1.000000	-0.006530	-0.017233	0.001216	
forehead_width_cm	-0.006530	1.000000	0.088596	0.251648	
forehead_height_cm	-0.017233	0.088596	1.000000	0.211655	
nose_wide	0.001216	0.251648	0.211655	1.000000	
nose_long	0.014432	0.257368	0.194120	0.565192	
lips_thin	0.011287	0.258564	0.205441	0.557615	
distance_nose_to_lip_long	-0.025794	0.251328	0.215292	0.569303	
gender	-0.010767	0.334125	0.277190	0.758502	

In [ ]: