test

August 9, 2023

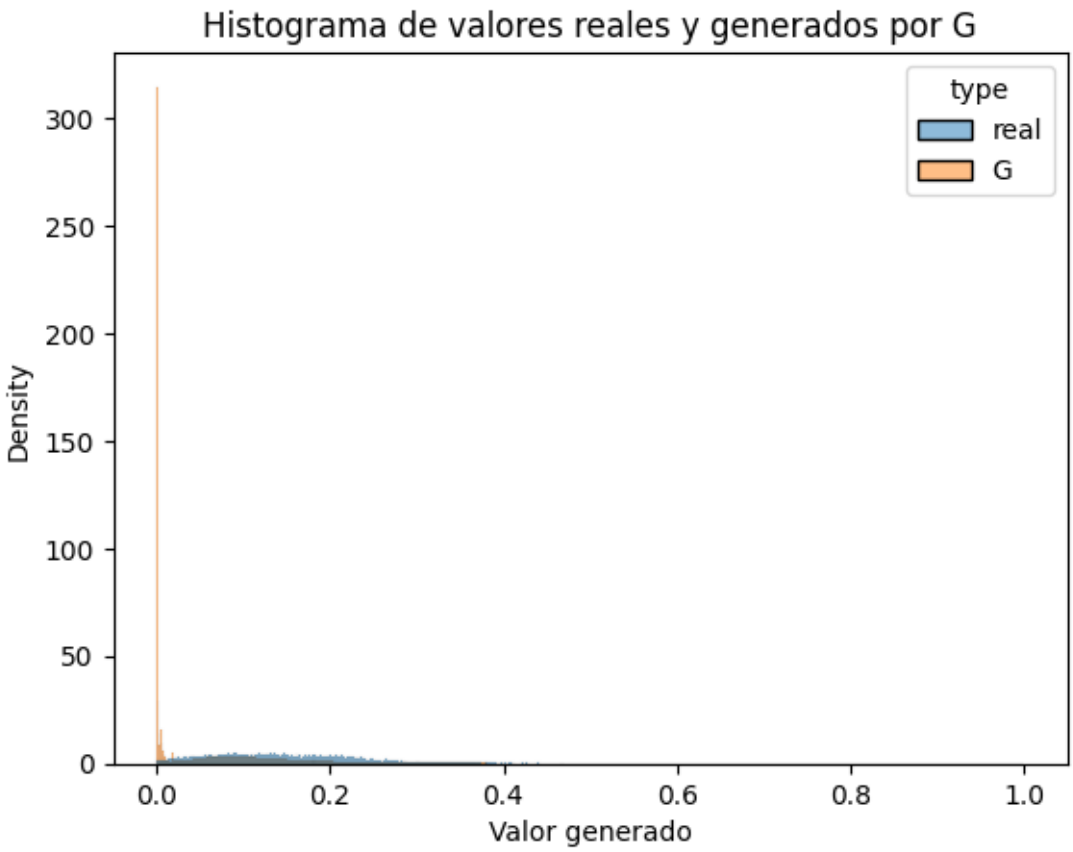
[145]:
```
import pickle
import seaborn as sns
import matplotlib.pyplot as plt
import pandas as pd
import torch
```

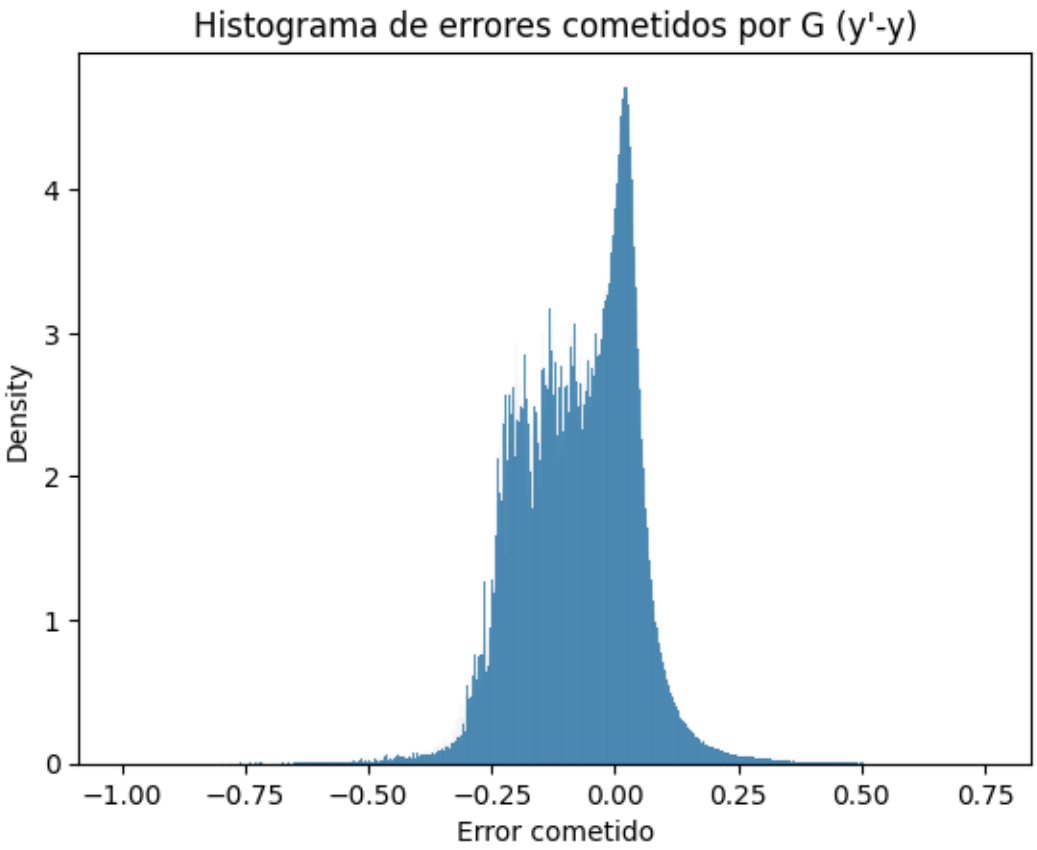
[146]:
```
def load_tensors(file_name):
    with open(file_name, 'rb') as f:
        outputs = pickle.load(f)

    for i in range(len(outputs)):
        if i == 0:
            x_real = outputs[i]['x_real']
            x_fake = outputs[i]['x_fake']
            d_pred = outputs[i]['d_pred']
            imputation = outputs[i]['imputation']
            input_mask_int = outputs[i]['input_mask_int']
            input_mask_bool = outputs[i]['input_mask_bool']
            known_values = outputs[i]['known_values']
        else:
            x_real = torch.cat([x_real, outputs[i]['x_real']], dim=0)
            x_fake = torch.cat([x_fake, outputs[i]['x_fake']], dim=0)
            d_pred = torch.cat([d_pred, outputs[i]['d_pred']], dim=0)
            imputation = torch.cat([imputation, outputs[i]['imputation']],␣
 ↪dim=0)
            input_mask_int = torch.cat([input_mask_int,␣
 ↪outputs[i]['input_mask_int']], dim=0)
            input_mask_bool = torch.cat([input_mask_bool,␣
 ↪outputs[i]['input_mask_bool']], dim=0)
            known_values = torch.cat([known_values,␣
 ↪outputs[i]['known_values']], dim=0)

    return x_real, x_fake, d_pred, imputation, input_mask_int, input_mask_bool,␣
 ↪known_values


def prepare_data(x_real, x_fake, imputation, input_mask_bool):
```

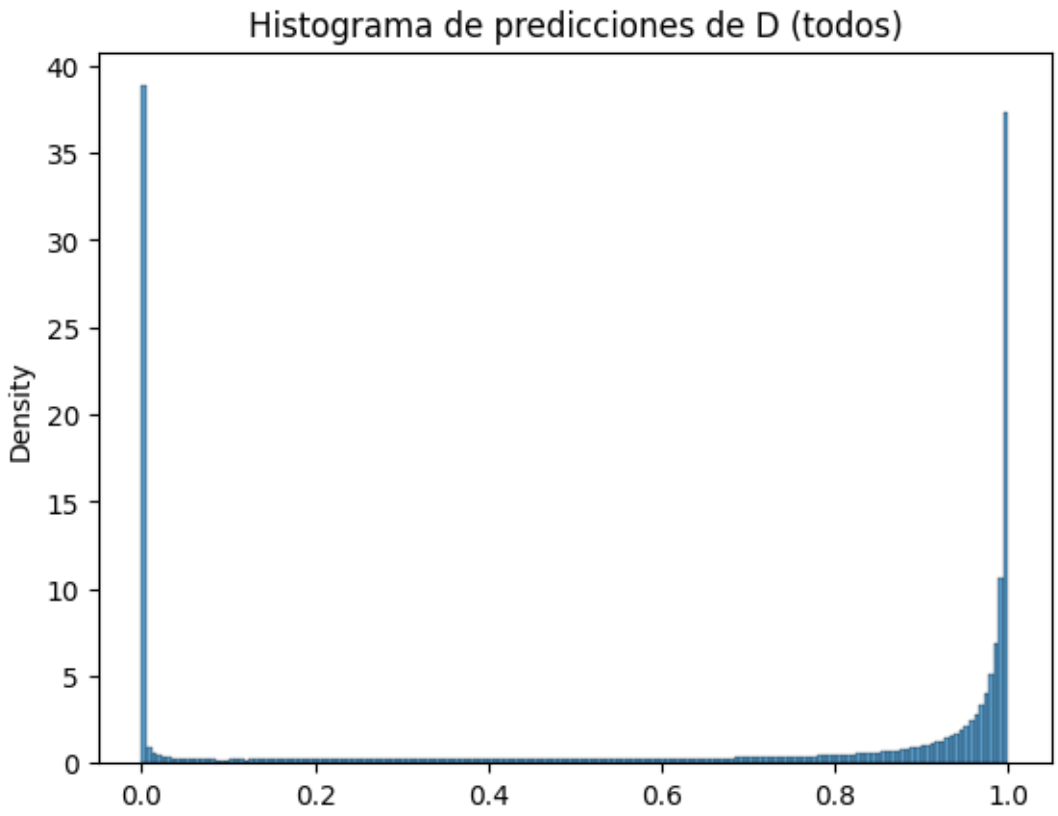
```
    x1 = x_real[~input_mask_bool].flatten().cpu().detach().numpy()
    x2 = x_fake[~input_mask_bool].flatten().cpu().detach().numpy()

    df1 = pd.DataFrame({'x': x1, 'type': ['real' for _ in range(len(x1))]})
    df2 = pd.DataFrame({'x': x2, 'type': ['G' for _ in range(len(x2))]})

    diff = imputation[~input_mask_bool].flatten().cpu().detach().numpy()-␣
 ↪x_real[~input_mask_bool].flatten().cpu().detach().numpy()
    return pd.concat([df1, df2]), diff

def plot_figures(df, diff, d_pred, input_mask_bool):
    plt.figure()
    sns.histplot(data=df, x='x', hue='type', stat='density', common_norm=False)
    plt.title('Histograma de valores reales y generados por G')
    plt.xlabel('Valor generado')
    plt.show()

    plt.figure()
    plt.title("Histograma de errores cometidos por G (y'-y)")
    plt.xlabel('Error cometido')
    sns.histplot(data=diff, stat='density')
    plt.show()

    plt.figure()
    sns.histplot(data=d_pred.flatten().cpu().detach().numpy(), stat='density')
    plt.title('Histograma de predicciones de D (todos)')
    plt.show()

    plt.figure()
    sns.histplot(data=d_pred[input_mask_bool].flatten().cpu().detach().numpy(),␣
 ↪stat='density')
    plt.title('Histograma de predicciones de D (reales)')
    plt.show()

    plt.figure()
    sns.histplot(data=d_pred[~input_mask_bool].flatten().cpu().detach().
 ↪numpy(), stat='density')
    plt.title('Histograma de predicciones de D (falsos)')
    plt.show()
```

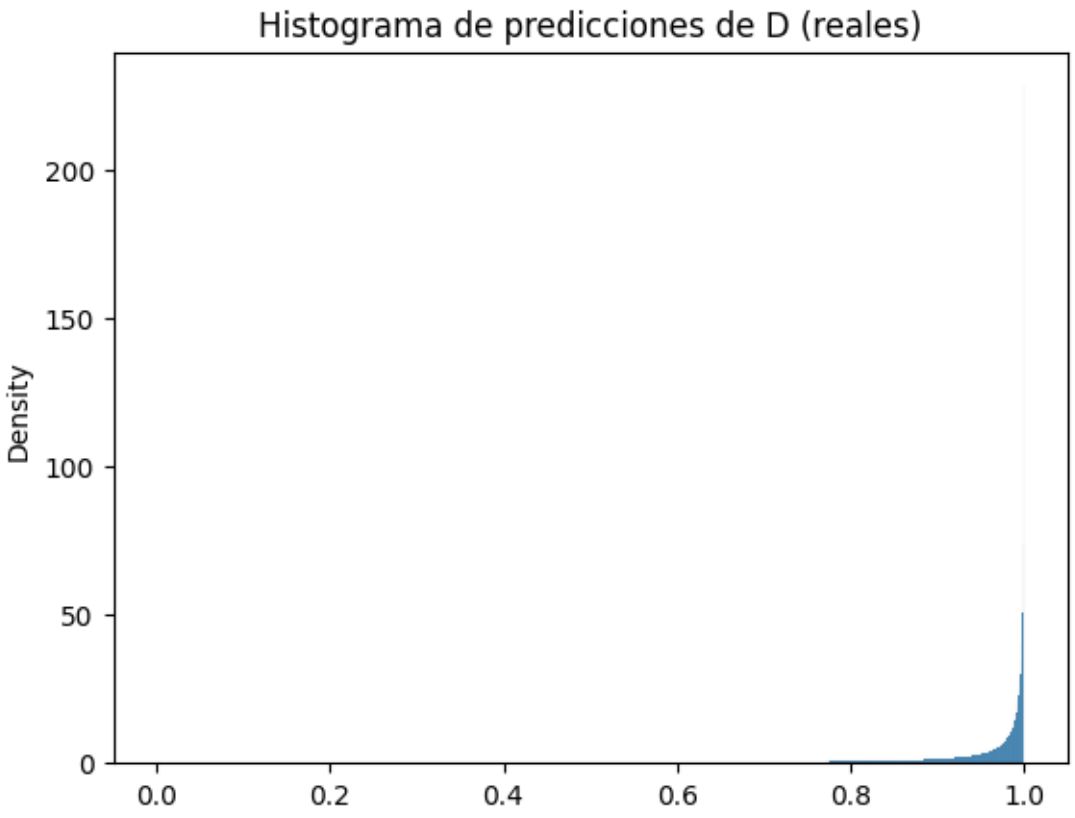
1 Análisis de resultados con H=0.9

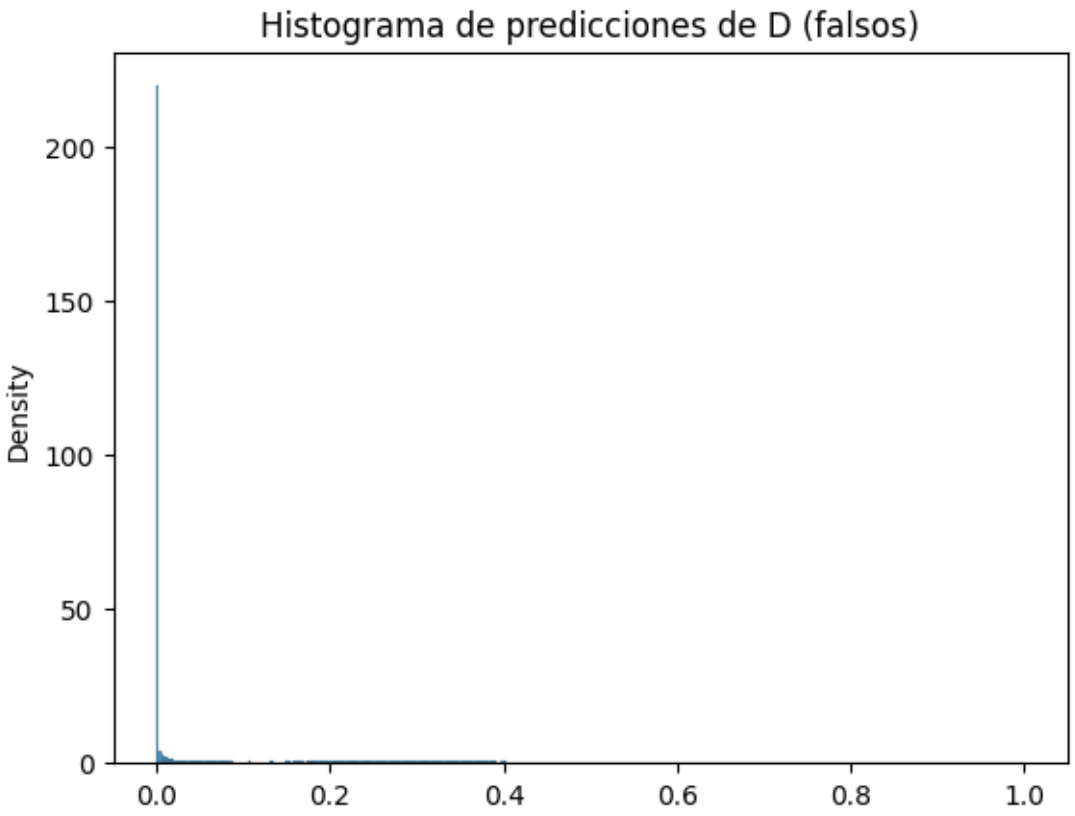
[147]:
```
x_real, x_fake, d_pred, imputation, input_mask_int, input_mask_bool,␣
 ↪known_values = load_tensors('outputs_test_h_0.9.pkl')
df, diff = prepare_data(x_real, x_fake, imputation, input_mask_bool)
plot_figures(df, diff, d_pred, input_mask_bool)
```


Histograma de valores reales y generados por G
type
real
G
Density
300
250
200
150
100
50
0
0.0
0.2
0.4
0.6
0.8
1.0
Valor generado


Histograma de errores cometidos por G (y'-y)
Density
0
1
2
3
4
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
Error cometido

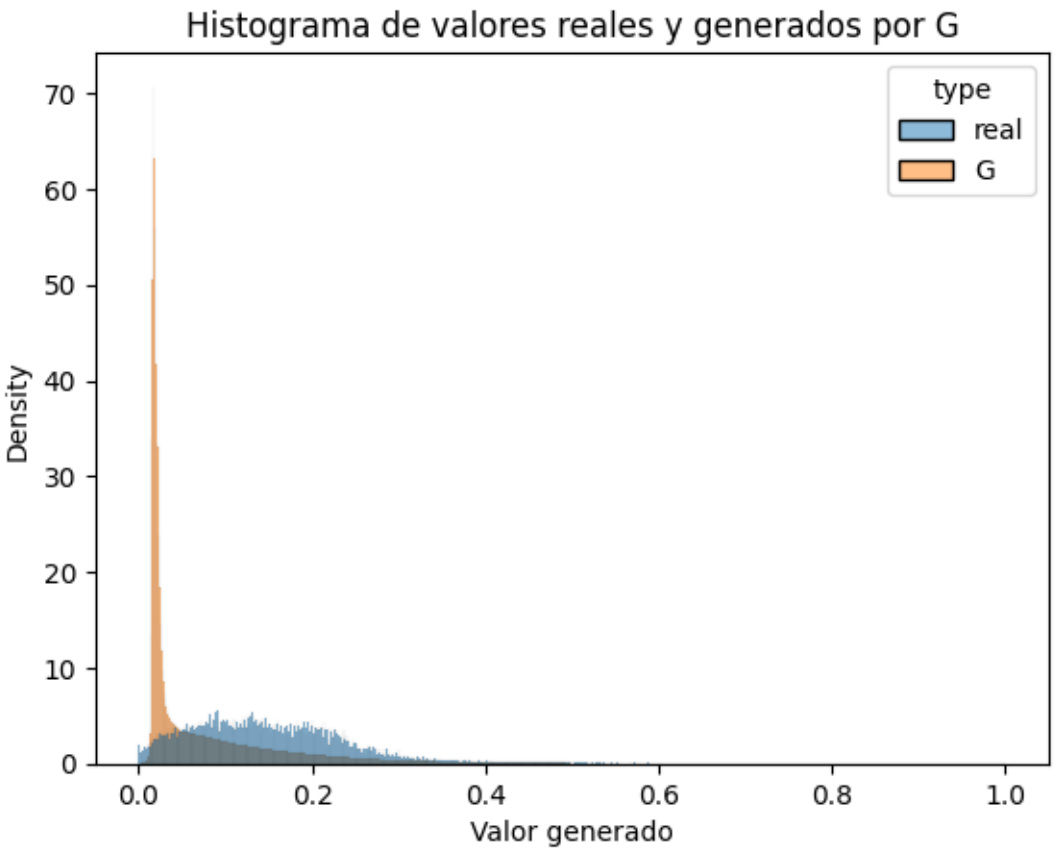

Histograma de predicciones de D (todos)
Density
40
35
30
25
20
15
10
5
0
0.0
0.2
0.4
0.6
0.8
1.0

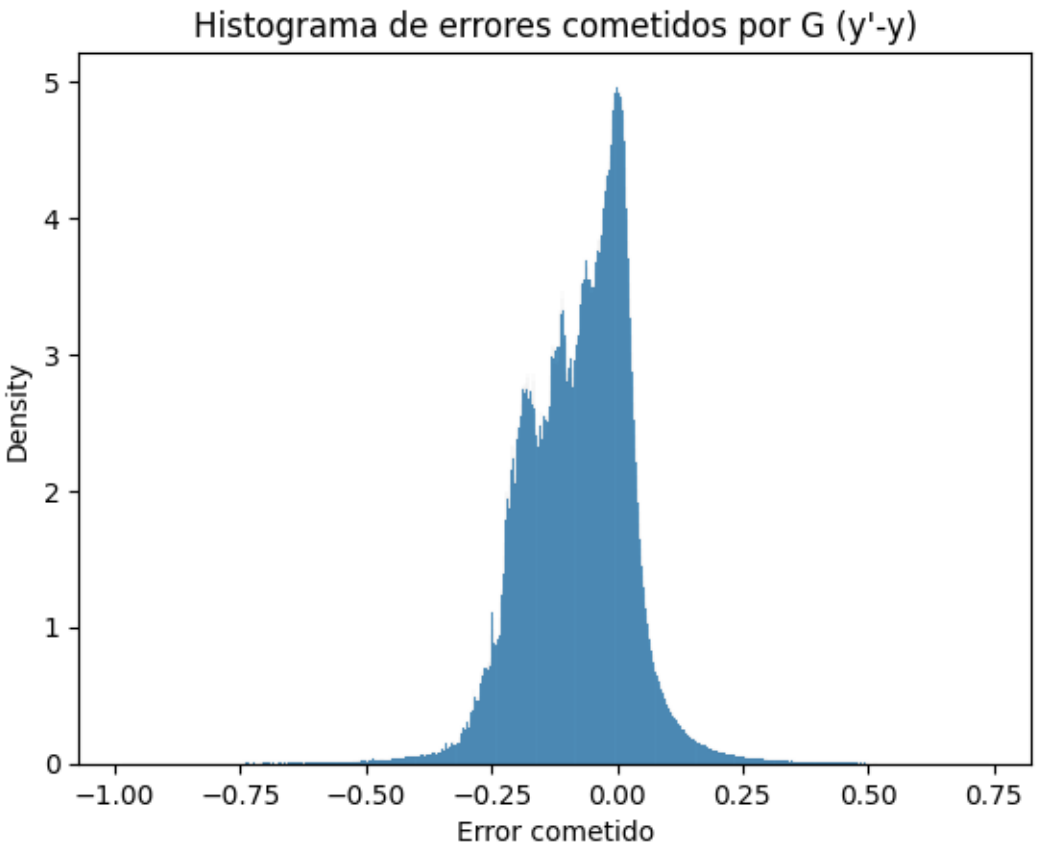

Histograma de predicciones de D (reales)
Density
200
150
100
50
0
0.0
0.2
0.4
0.6
0.8
1.0

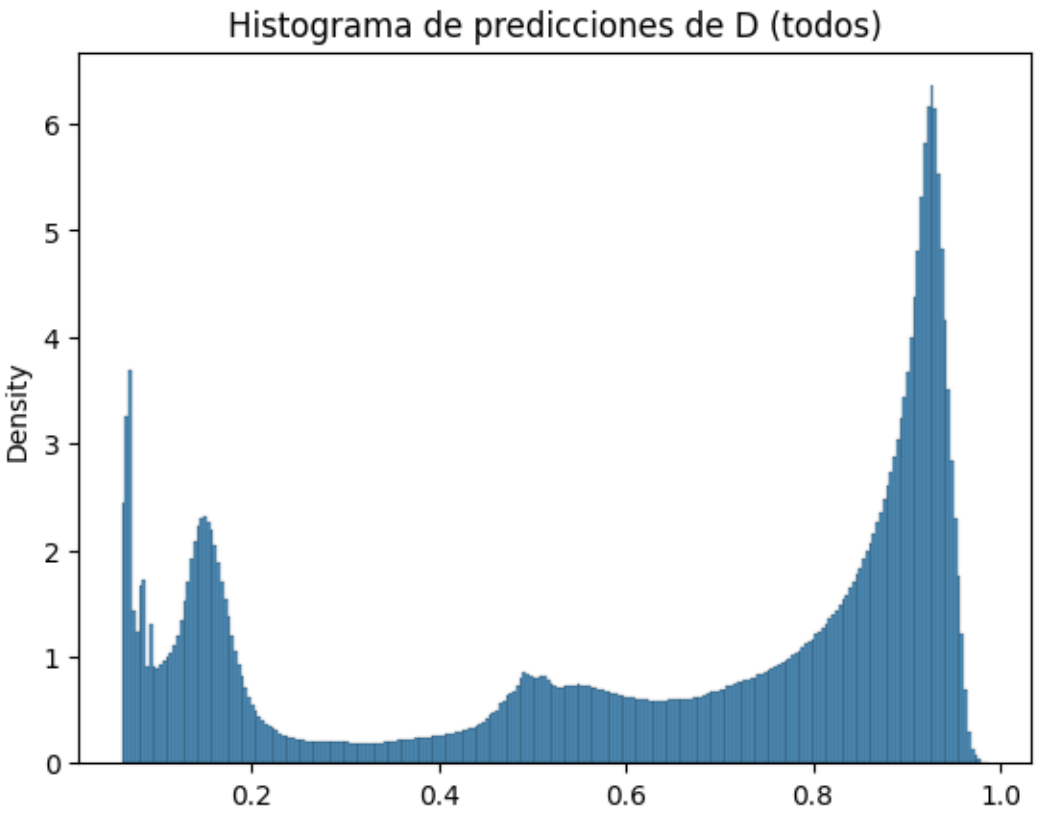


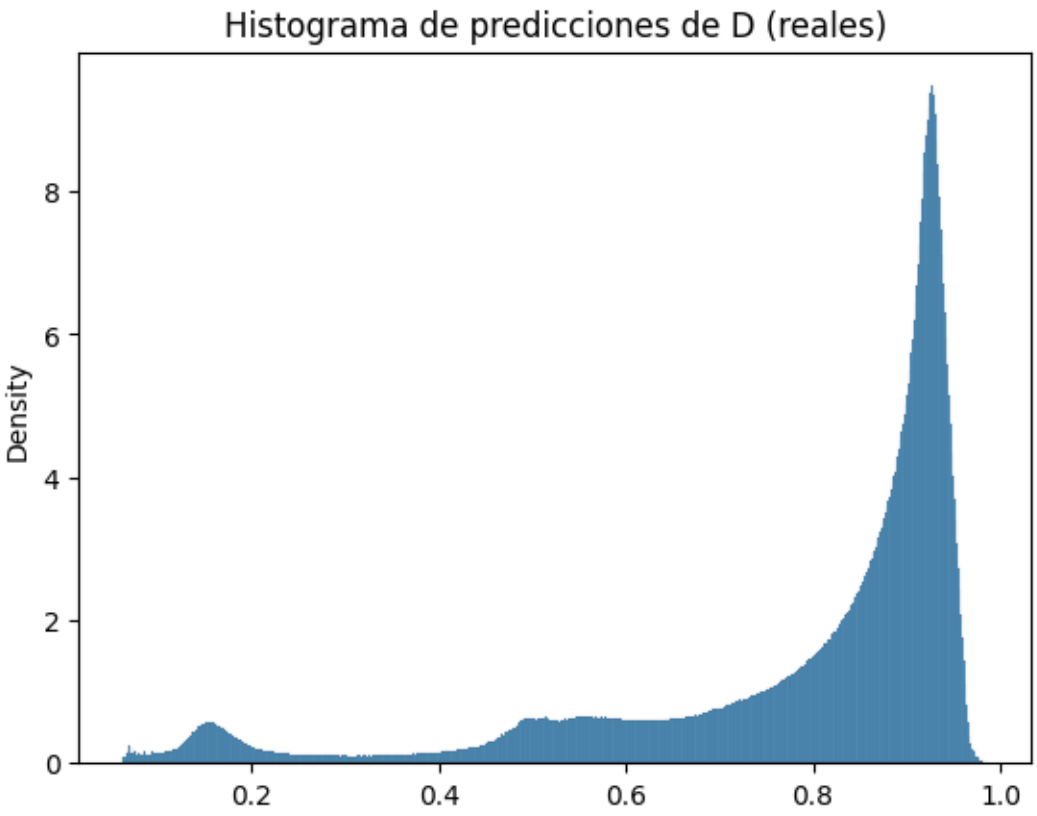
2 Análisis de resultados con H=0.1

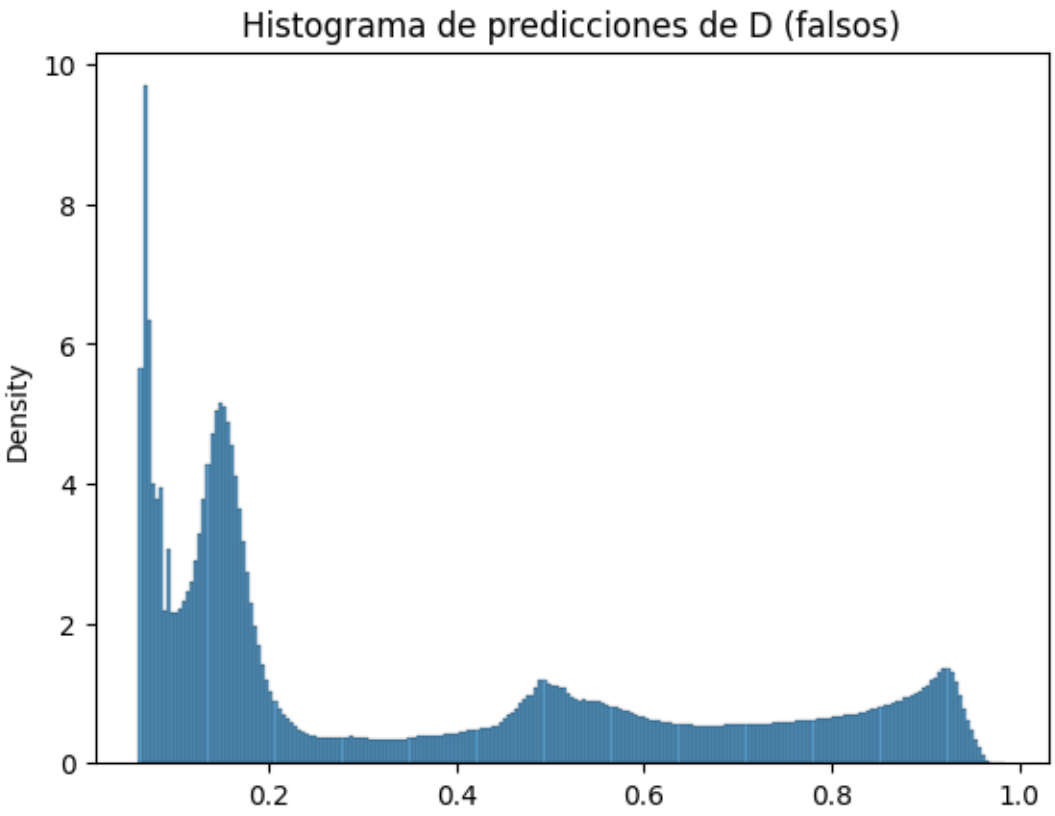
[148]:
```
x_real, x_fake, d_pred, imputation, input_mask_int, input_mask_bool,␣
  ↪known_values = load_tensors('outputs_test_h_0.1.pkl')
df, diff = prepare_data(x_real, x_fake, imputation, input_mask_bool)
plot_figures(df, diff, d_pred, input_mask_bool)
```


Histograma de valores reales y generados por G
type
real
G
Density
Valor generado
0
10
20
30
40
50
60
70
0.0
0.2
0.4
0.6
0.8
1.0


Histograma de errores cometidos por G (y'-y)
Density
Error cometido
0
1
2
3
4
5
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75


Histograma de predicciones de D (todos)
Density
0
1
2
3
4
5
6
0.2
0.4
0.6
0.8
1.0


Histograma de predicciones de D (reales)
Density
0
2
4
6
8
0.2
0.4
0.6
0.8
1.0


Histograma de predicciones de D (falsos)
Density
10
8
6
4
2
0
0.2
0.4
0.6
0.8
1.0