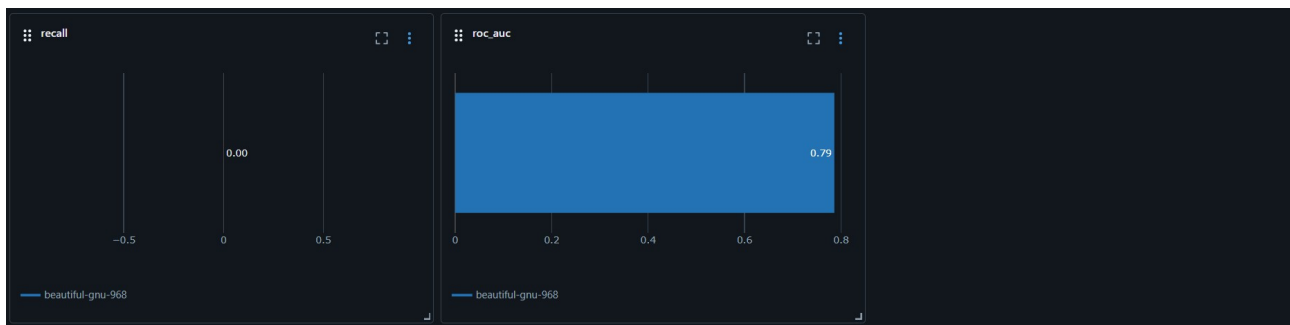


En el primer modelo presentado había un desajuste (0,04) de cantidad de variables objetivo:

Primer modelo



.

Para obtener un mejor resultado usamos un resampling de datos para equiparar las cantidades de variables dependiente e independientes:

```
from sklearn.utils import resample

# Separar clases
df_minority = df[df['stroke'] == 1]
df_majority = df[df['stroke'] == 0]

# Aumentar la clase minoritaria con Bootstrap (muestreo con reemplazo)
df_minority_boot = resample(df_minority,
                             replace=True,
                             n_samples=1000, # o igualar al tamaño de la clase mayoritaria
                             random_state=42)

# Combinar con clase mayoritaria
df_balanced = pd.concat([df_majority, df_minority_boot])

# Verificar nuevo balance
print(df_balanced['stroke'].value_counts(normalize=True))
```

```
stroke
0    0.824561
1    0.175439
Name: proportion, dtype: float64
```

Clase	Proporción Original	Proporción Nueva
stroke = 1	~4.9%	17.5%
stroke = 0	~95.1%	82.5%

Modelo balanceado

