

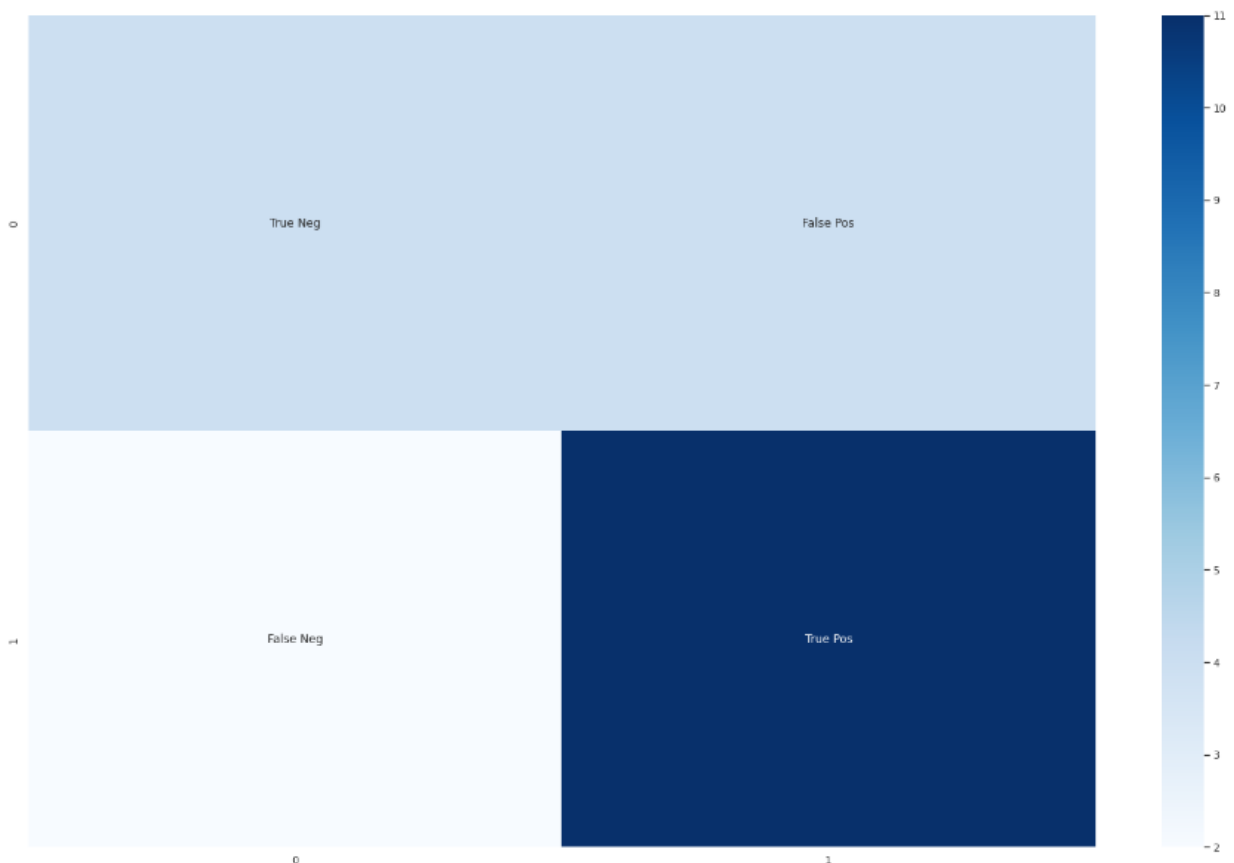
# Implementación de un clasificador con **Regresión logística** para predecir el sexo de zarigüeya.

Del paquete DAAG R: "El dataset de la zarigüeya consta de nueve medidas morfométricas en cada una de las 104 zarigüeyas cola de cepillo de montaña, atrapadas en siete sitios desde el sur de Victoria hasta el centro de Queensland".

## Comparación entre modelos

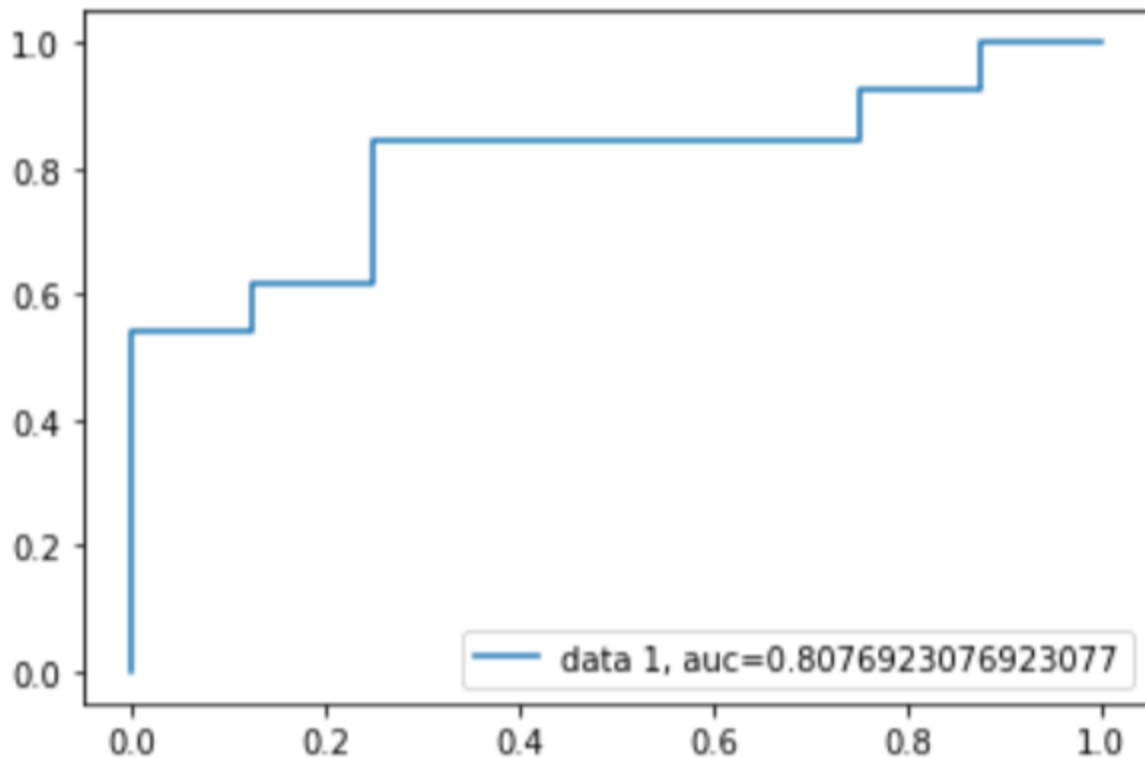
**Modelo 1.** (Precisión de 0.71).

*Matriz de confusión*



Vemos que este modelo nos da valores falsos positivos al igual que verdaderos negativos, lo cual indica una mala predicción. (Un buen modelo tendría mayores valores y por ende un color azul más oscuro en los cuadros de “True Neg” y “True Pos” mientras que los valores de “False Pos” y “False Neg” tendrían que ser cerca a 0 y de color transparente o blanco.

*Gráfica ROC*



La puntuación AUC (área bajo la curva) para el modelo es 0,80. La puntuación AUC 1 representa un clasificador perfecto y 0,5 representa un clasificador sin valor.

*Bias y Varianza*

```
Model 1: Average Bias : 0.19005833333333333
Model 1: Average Variance : 0.14494166666666666
```

Vemos que existe un **bias alto relativo** en comparación con la varianza, más que nada debido a que estamos utilizando una regresión logística. El valor de la **varianza es relativamente poco**, ya que la suma de cuadrados es similar a diferentes datasets. Esto refleja que nuestro modelo dará buenas predicciones, más no excelentes. Además, refleja un **underfitting**, ya que el bias es alto

en comparación con la varianza, lo que significa que el modelo no es lo suficientemente complejo como para capturar bien el patrón en los datos de entrenamiento y, por lo tanto, también sufre de bajo rendimiento en datos no vistos.

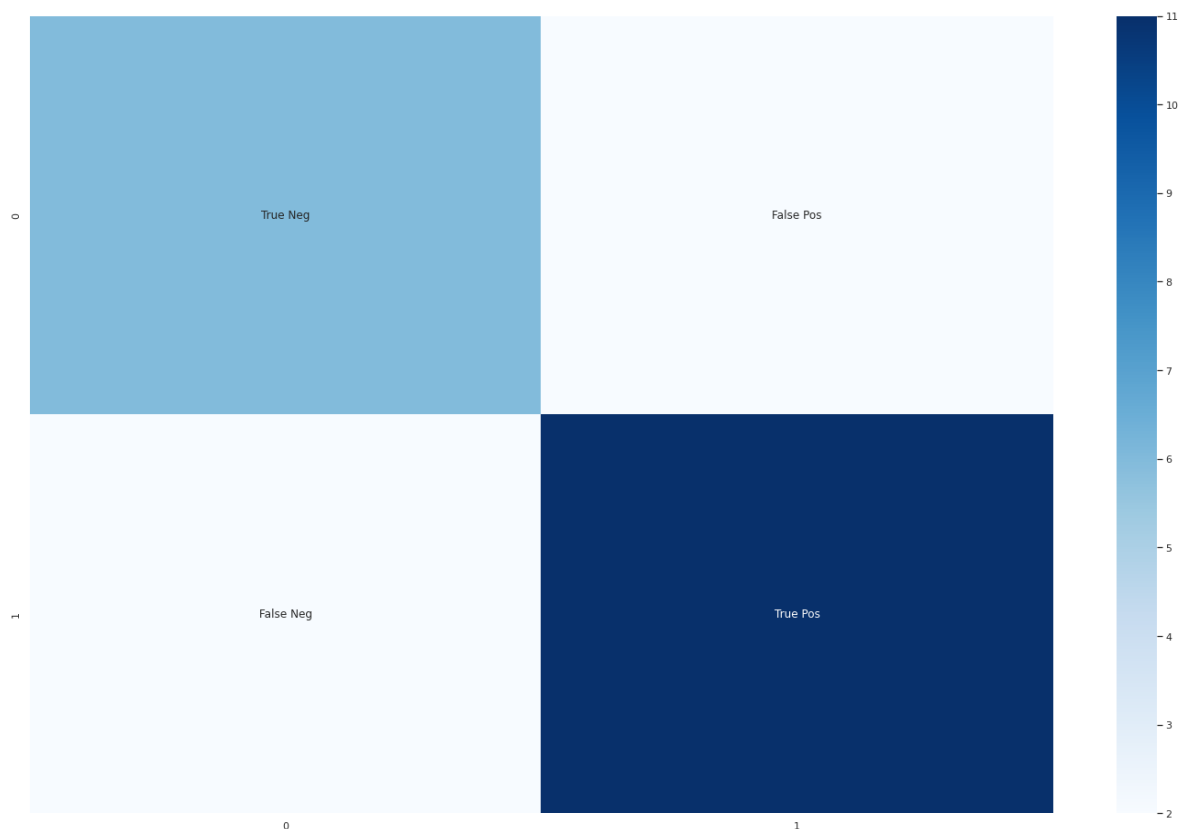
## **Modelo 2.** (Precisión de 0.81).

Debido a que es regresión logística, realmente no se tienen hiperparámetros críticos para ajustar. (Gusarov, 2022). A veces, pueden ver diferencias útiles al modificar los diferentes solucionadores (solver). De los que se encuentran en scikit learn; ['newton-cg', 'lbfgs', 'liblinear', 'sag', 'saga']. En nuestro caso se probaron todos y 'newton-cg' nos dio el mejor resultado.

Uno de los otros parametros importantes para la mejora del modelo son la penalización (o regularización) que pretende reducir el error de generalización del modelo y pretende desincentivar y regular el overfitting. En nuestro caso, no tenemos overfitting, sino underfitting, por lo cual se descarta. Sin embargo, se modificó el dataset para mejorar nuestras predicciones.

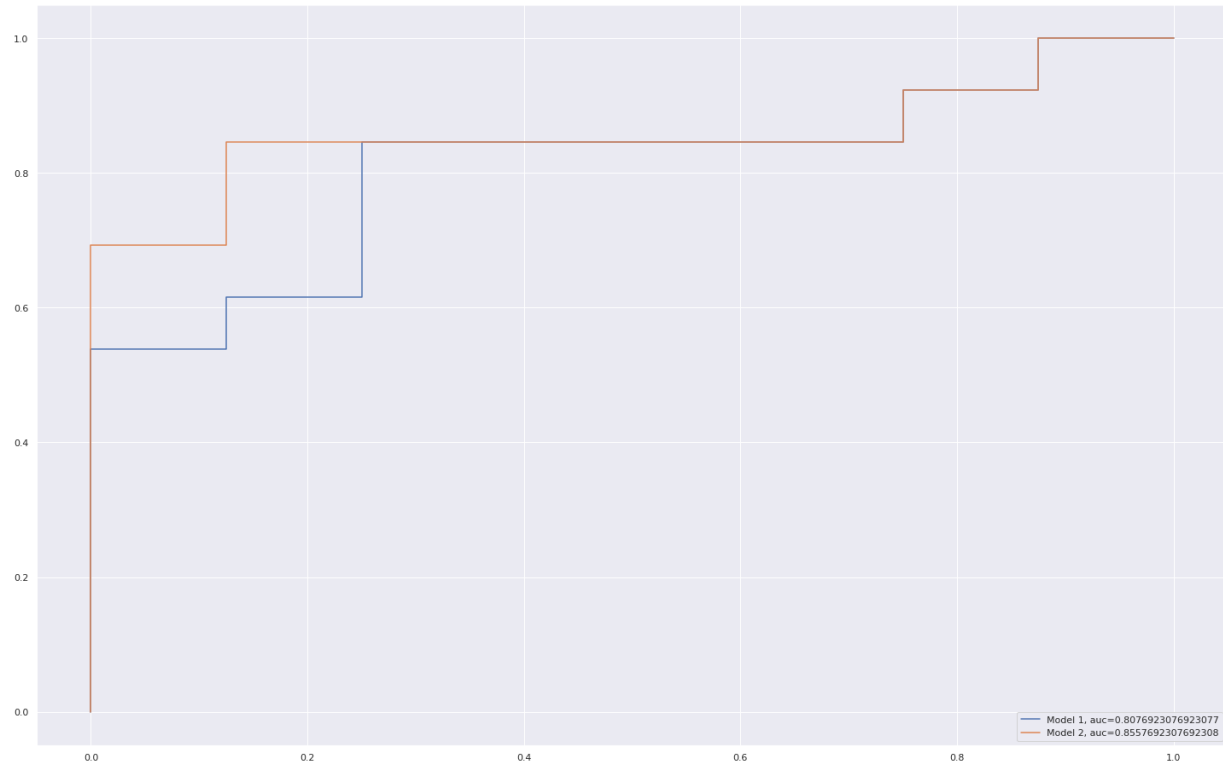
Se realizó un GridSearch con ayuda de la librería de scikit learn para él tuneo de hiperparámetros; sin embargo, no mejoro el modelo.

*Matriz de confusión*



Se mejora el modelo y sus predicciones para cada cuadro de la matriz. Vemos que el valor de Falso Positivo disminuye en gran cantidad y que el valor de Verdadero Negativo incrementa en valor y por ende en su tono fuerte de color.

*Gráfica ROC*



La puntuación AUC para el modelo nuevo es 0,85, una mejora de 5% en comparación del modelo anterior. La puntuación AUC 1 representa un clasificador perfecto y 0,5 representa un clasificador sin valor.

*Bias y Varianza*

```
Model 1: Average Bias : 0.19005833333333333
Model 1: Average Variance : 0.14494166666666666
Model 2: Average Bias : 0.15202261904761907
Model 2: Average Variance : 0.13916785714285718
```

Vemos que **el bias disminuyo** y que **no existió mucho cambio en la varianza**. Sin embargo, mejoro bastante el modelo en cuanto a precisión. (10 décimas).