Universidad del Valle de Guatemala
Minería de Datos
Lynette García
01/04/2019

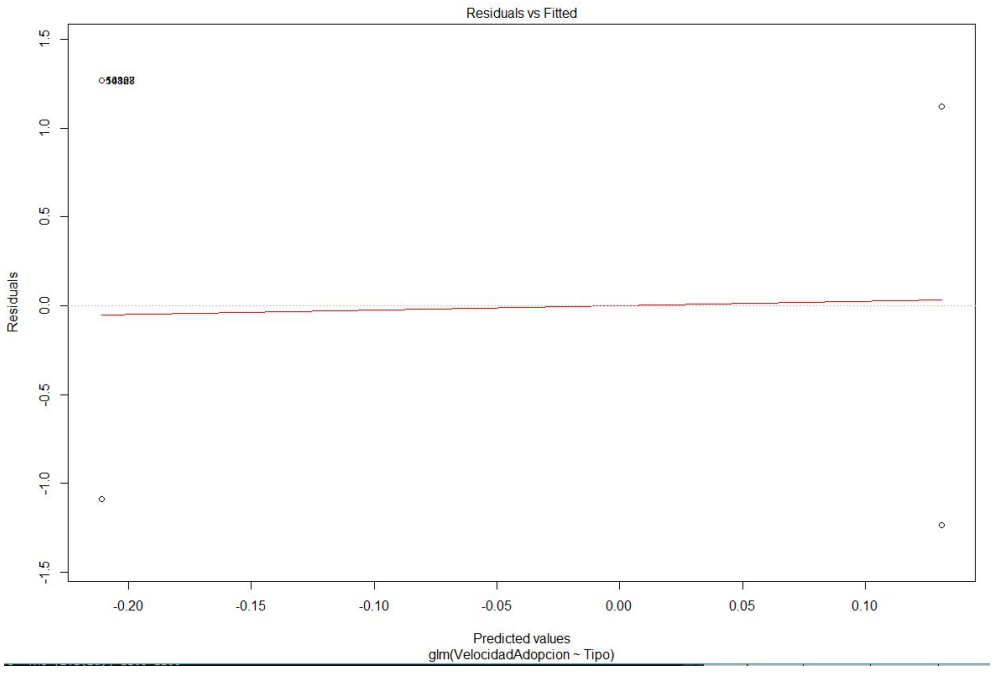
Andrea María Cordón Mayén, 16076
Cristopher Sebastian Recinos Ramírez, 16005
Fecha de Entrega:

Hoja de Trabajo 6
"Regresión Logística"

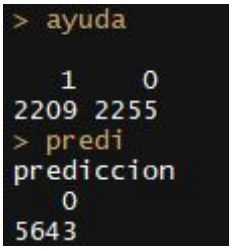
1. **Transformar la variable respuesta:**
    a. Se transformó la variable AdoptionSpeed a factor debido a que se necesita para realizar la regresión logística. Esto se realizó con el siguiente código:

```
datos$AdoptionSpeed <- factor(datos$AdoptionSpeed)
datos$AdoptionSpeed
```

2. **Elabore un modelo de regresión logística utilizando el conjunto de entrenamiento y explique los resultados a los que llega. *Mostrar gráfico***
    a. Se elaboró el modelo de regresión logística con los datos de entrenamiento. Este modelo se realizó utilizando las variables AdoptionSpeed y Type ya que fueron los que mejor resultado dieron en las pruebas anteriores. A continuación se mostrará el gráfico que este modelo generó. Como se puede observar, el modelo parece que se adaptó bastante a los datos y va a generar una buena predicción.



3. **Analizar el modelo. Determinar si hay multicolinealidad en las variables y cuáles son las que aportan al modelo por su valor de significación. Hacer un análisis de correlación de las variables del modelo y especifique si el modelo se adapta bien a los datos. Explicar si hay sobreajuste.**
    a. Como se pudo observar en el inciso anterior, pareciera que existe multicolinealidad en las variables AdoptionSpeed y Type. Esto debido a que la relación que obtuvieron dentro del modelo fue muy buena y parece ser que es un modelo casi perfecto. Debido a esto, se puede afirmar que existe sobreajuste ya que un modelo casi perfecto significa que intentó acercarse lo más posible a la realidad.
4. **Utilice el modelo con el conjunto de prueba y determine la eficiencia del algoritmo para clasificar o predecir en dependencia de las características de la variable respuesta.**
    a. Después de comparar las predicciones realizadas con el modelo, se puede determinar que el algoritmo no fue tan bueno como se esperaba. Como se podrá notar en las siguientes imágenes, este algoritmo predijo que todo sería 0 cuando en realidad habían cantidades iguales de 1 y 0 dentro de la data inicial.



5. **Haga un análisis de la eficiencia del algoritmo usando una matriz de confusión.**
    a. Lamentablemente, debido a cómo genera los datos de predicción de este modelo la función predict, no se puede generar la matriz de confusión. Sin embargo, podemos notar que tiene una eficiencia de 39.96% ya que predijo bien cierta cantidad de los datos (como se pudo observar en las imágenes del inciso anterior).

6. **Compare la eficiencia del algoritmo al aplicar a los datos modelos de árbol de decisión y naive bayes. ¿Cuál es le mejor para predecir? ¿Cuál se demoró más en procesar?**
    a. Si se utiliza la versión de este modelo que utiliza la función cbind, entonces este sería el modelo que más se demora en procesar los datos. Si no, entonces sigue siendo el algoritmo de árboles de decisión el más tardado. En cuanto cual es el mejor para predecir, sigue siendo el modelo de regresión lineal con una accuracy de 55.51%.
7. **Actualizar el kernel en kaggle:** https://www.kaggle.com/rec16005/petterinosnaivebayes