Examen_predicción

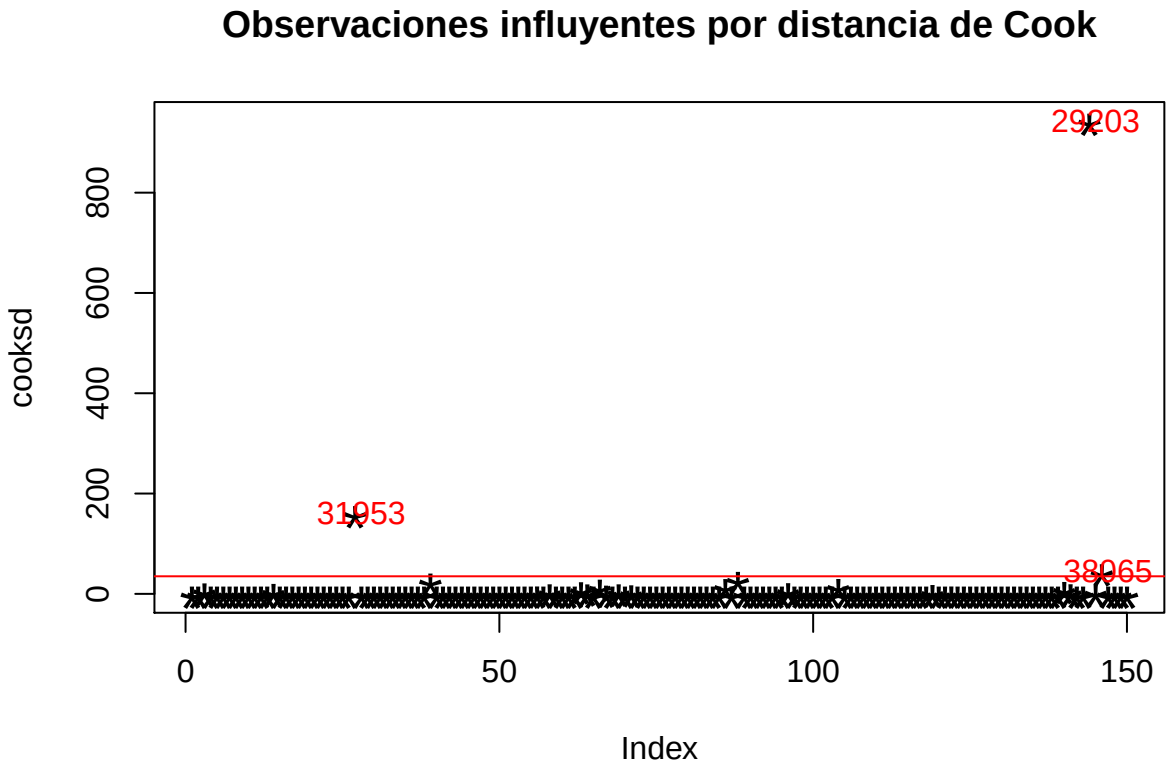
Luis Llera García

1 de febrero de 2019

Executive Summary

Modelo de Regresión Logística Primeramente realizamos la carga de las librerías necesarias para realizar el modelo de Regresión, realizamos la limpieza de datos exigida para el examen, y establecemos como categórica la variable que queremos predecir, que va a ser Valoración medio ambiente y establecemos por debajo de 8 el valor 0 y por encima el valor 1. Y establecemos los primeras 150 observaciones como train que son antes del año 2014 y las otras como test que son después de 2014. Primeramente realizamos le modelo de Regresión Logística, simple, con todas las variables, después de esto realizamos un método de selección de variables que se basará en un criterio de menor AIC, y realizamos un Step, 'por el metodo both', que son forward y backward. Y después hemos realizado con el modelo ANOVA, la selección de variables, y hemos realizado el modelo. Comparandolos con el criterio de AIC, los resultados que nos aporta es que el mejor modelo según este criterio es es que nos ha proporcionado el Step, como era de esperar. El problema del modelo GLM, es que al ser un modelo lineal el algoritmo no converge al 100%, que es el aviso que nos da el R, pero vamos a continuar con el análisis para ver sus resultados finales. Ahora realizamos el análisis de los outliers.

```
cooksd <- cooks.distance(modelo_bueno_training)
plot(cooksd, pch='*', cex = 2, main = 'Observaciones influyentes por distancia de Cook')
abline(h = 4*mean(cooksd, na.rm=T), col="red")
#add cutoff line
text(x=1:length(cooksd)+1, y=cooksd, labels=ifelse(cooksd>4*mean(cooksd, na.rm=T),names(cooksd),""), co
```



Y como podemos observar solo tenemos 3. Ahora realizamos el análisis del cut off que primeramente lo establecimos a mano pero luego lo comprobamos y lo aplicamos, a pesar de arrojarnos los mismos resutados. El intervalo de confianza es 0.25 y 0.117,finalmente nos quedamos con este 0.125. Por tanto lo único que nos queda es ver como se ha realizado la predicción, es decir, comprobar mediante la matriz de confusión que valores se han clasificado bien y cuales se han clasificado mal. Tenemos 36 individuos mal calificados, lo cual, nos arroja un accuracy del 76%. 0 1 0 27 26 1 10 87

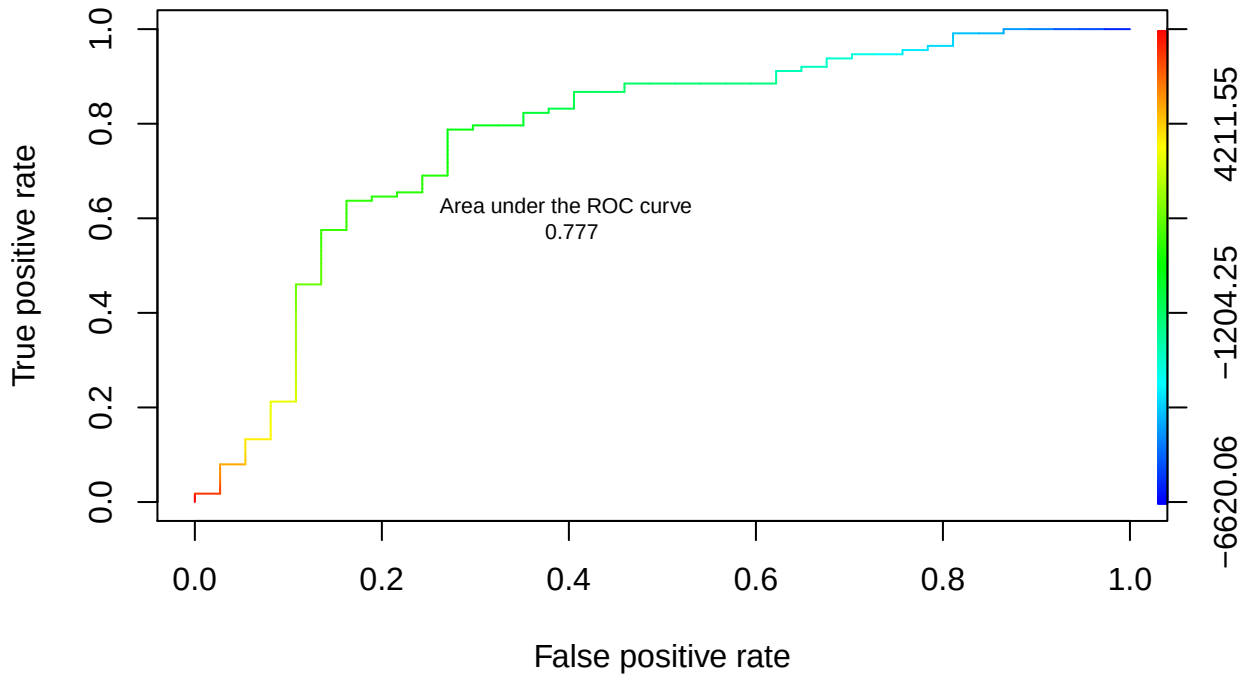
```
roc <- prediction(predict(modelo_bueno_training, test1), test1$VALORACION_MEDIO_AMBIENTE)
AUC <- ROCR::performance(roc, "auc")
AUC@y.name
```

```
## [1] "Area under the ROC curve"
```

```
AUC@y.values
```

```
## [[1]]
## [1] 0.7770868
```

```
perf <- ROCR::performance(roc, "tpr", "fpr")
plot(perf, colorize = TRUE)
text(0.4, 0.6, paste(AUC@y.name, "\n", round(unlist(AUC@y.values), 3)), cex = 0.7)
```



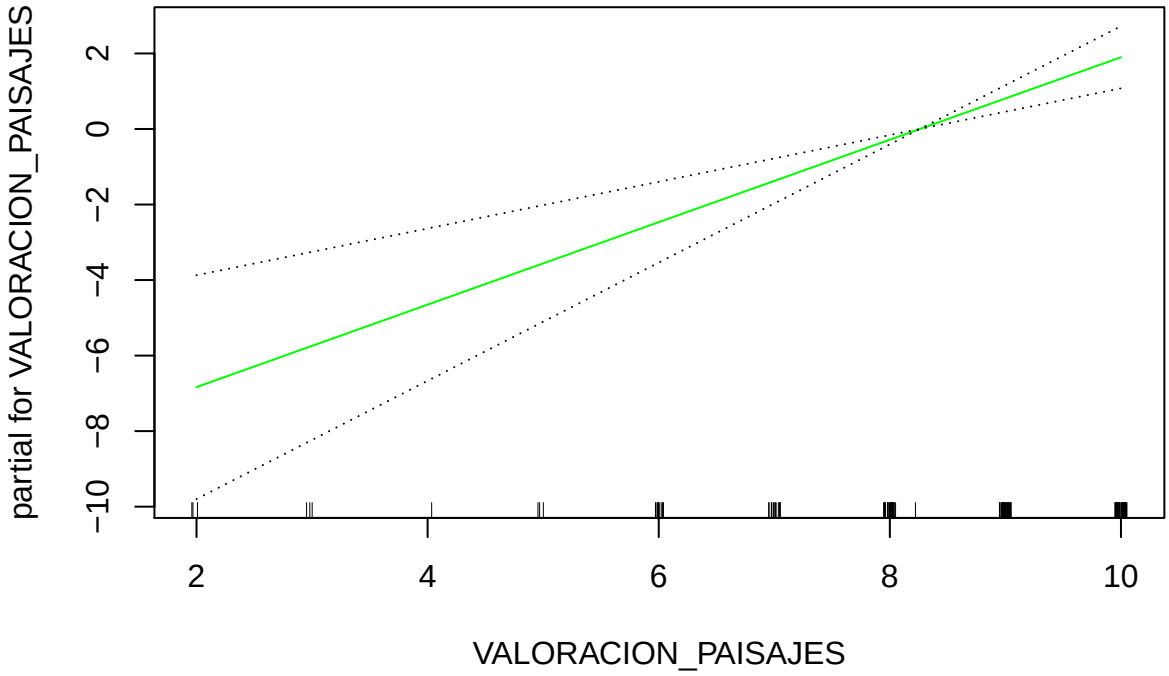
Otra medida que se encarga de medir la especificidad frente a la sensibilidad es lo que se conoce como curva ROC, y mide también la proporción tanto de falsos positivos como falsos negativos. Lo máximo que se puede tener de área debajo de la curva es un 1 que sería un ángulo recto nosotros tenemos un 72%, lo cual no está nada mal. También realizamos los modelos de regularizacion para la seleccion de variables tanto el Lasso, como el Ridge, como el Elastic net pero los resultados que nos aportaban no eran concluyentes por eso lo descartamos.

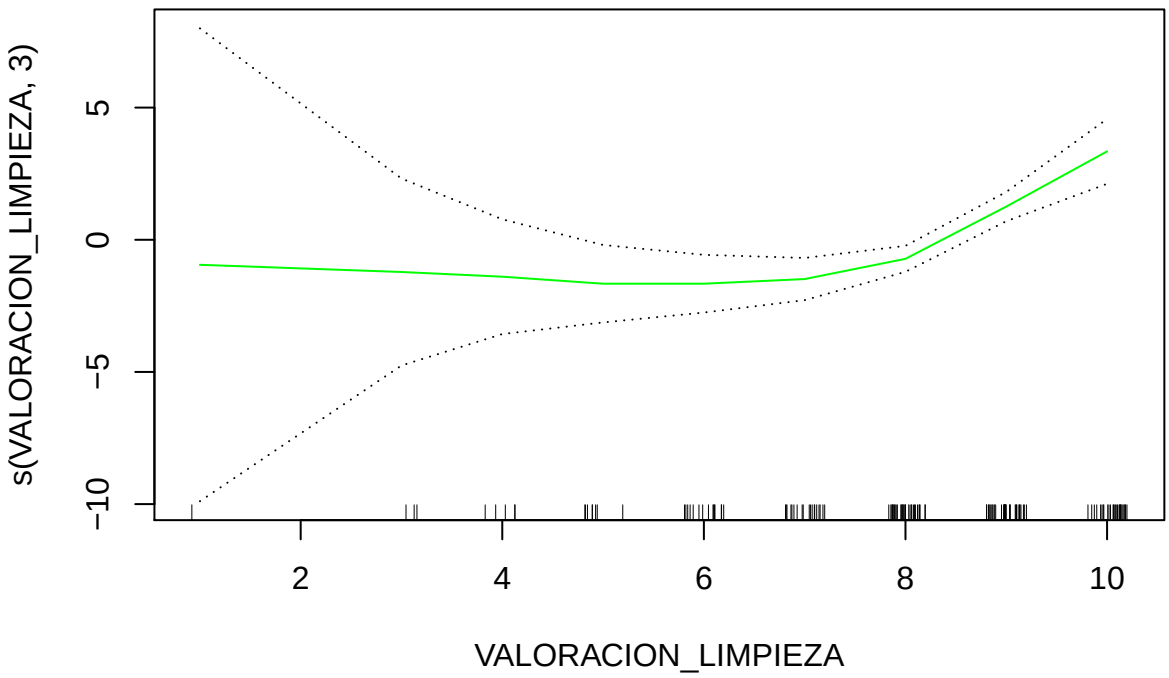
Modelos aditivos generalizados (GAM) Debido a los problemas que nos arrojaba el algoritmo decidimos realizar el Modelo que incluyera un tipo de modelo que incorpora la regresión no-paramétrica y la no-linealidad. Esto se debe a que los residuos no se distribuyen normalmente, por eso el algoritmo no llega a encontrar una solución perfecta ya que no son tan flexibles como los no lineales. Primeramente hemos realizado un análisis de la matriz de correlaciones, para observar problemas de multicolienalidad y ver como se relacionan las variables, luego hemos analizado las numerosas hipótesis para realizar el modelo la normalidad, homocedasticidad... También hemos realizado numerosos boxplos para analizar como se distribuyen las diferentes variables categóricas.

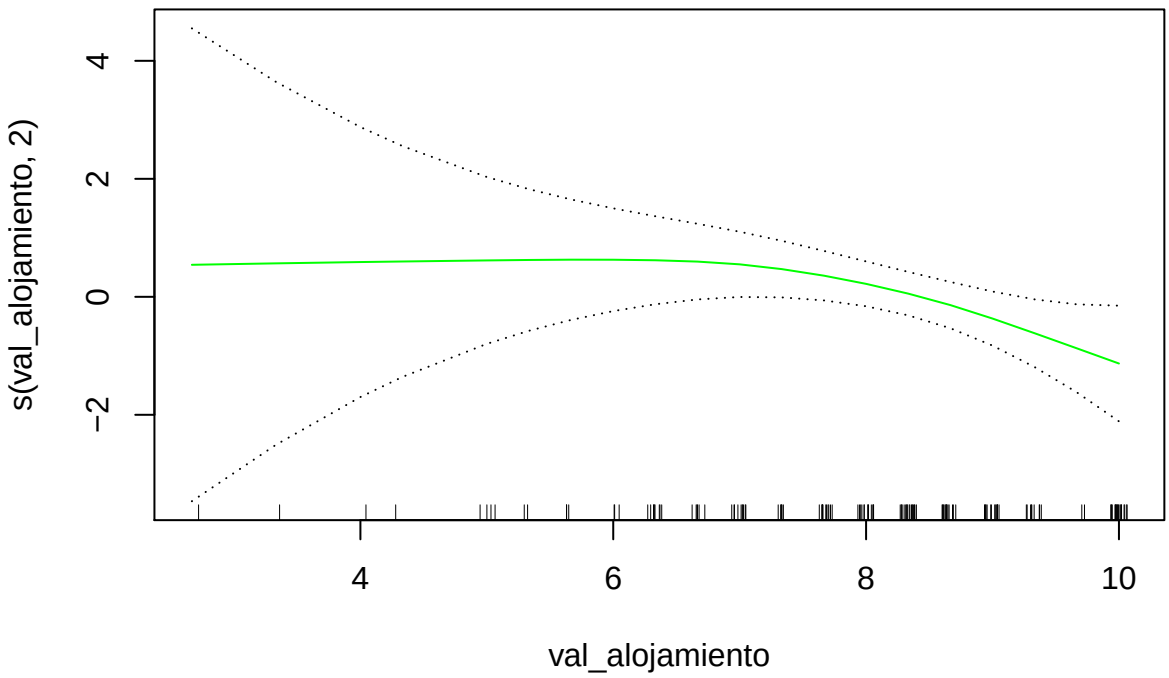
Una vez realizado el modelo Gam logísitco

```
modelo_gam_logit <- gam(target ~ VALORACION_PAISAJES + s(VALORACION_LIMPIEZA, 3) + s(val_alojamiento,2)
                        family = "binomial", data = lg_variables)
resumen <- summary(modelo_gam_logit)


plot(modelo_gam_logit, se = TRUE, col = "green")
```




s(VALORACION_LIMPIEZA, 3)
5
0
−5
−10
2
4
6
8
10
VALORACION_LIMPIEZA



Finalmente si analizamos la matriz de confusión y su accuracy, nos arroja un resltado del 78% y esta es su matriz de confusión, Reference Prediction 0 1 0 66 12 1 20 52 Por tanto podemos concluir que nuestros datos se adaptan mejor mediante un modelo no lineal osea se polinómico aunque mejore el acuracy solamente un 6%, pero aquí entra el debate entre negocio y computación ya que es un modelo algo mas complejo que un simple glm, por tanto en el futuro se debe de decidir entre si nos compensa unas cuantas horas más de trabajo o un aumento del accuracy de un 6%.