|  |
| --- |
| Miguel Ángel Castaño Ibáñez |

# Minería de datos y Modelización Predictiva

A continuación, se muestran la información que contienen las variables de nuestro conjunto de datos:

1. Introducción al objetivo del problema y las variables implicadas.

Elegimos nuestras variables objetivo entre las propuestas en el ejercicio:

* **AbstentionPtge**, variable objetivo continua. Representa el porcentaje de abstención en una población.
* **AbstencionAlta**, variable objetivo binaria. Variable dicotómica que toma el valor 1 si el porcentaje de abstención es superior al 30 % y, 0, en otro caso.

En el archivo adjunto a este trabajo *(“DatosEleccionesEspaña.xlsx”)* podemos ver todas las variables de nuestro dataframe y que representan cada una de estas.

1. Importación del conjunto de datos y asignación correcta de los tipos de variables.

Importamos todas las librerías necesarias en esta práctica (líneas 17-31) y el paquete de funciones *“FuncionesRosa.R”* (línea 42).

Cargamos los datos con la función: “*read\_excel”* y de la hoja de calculo que se muestra en la imagen a continuación. Una vez importado los datos comprobamos los tipos de las variables y cambiaremos a factor todas aquellas variables cualitativas.

Texto

Descripción generada automáticamente

El paso siguiente será analizar las variables numéricas y en caso de que alguna de estas contemple pocos valores recodificarla a tipo factor. En este conjunto datos no se da el caso, ya que algunas variables numéricas fueron tipadas a factor en líneas de código anteriores y la línea 55 nos muestra que ninguna variable numérica tiene pocos valores para poder tiparla a factor.

1. Análisis descriptivo de datos en el conjunto de training. Número de observaciones, número y naturaleza de variables, datos erróneos etc.

Este apartado hemos procedido ha ejecutar la función *summary(datos),* *describe(Filter(is.numeric, datos))* para variablesnuméricas y *freq(datos$variableCuant)* para variables cualitativas. Esto nos ayudara a tener una visión más general de los datos, y en base a esta observación procederemos a corregir errores detectados en el apartado 4.

1. Corrección de los errores detectados.

En este apartado vamos a empezar por analizar la frecuencia de las observaciones en las variables cualitativas y en caso de obtener alguna poco representadas, unirlas en subgrupos. En este caso contemplamos tres (líneas 57-86):

* **ActividadPpal:** En el caso de la actividad presencial obtenemos esta representación muy baja por parte de la industria y construcción, por ello he considerado agrupar Industria y Construcción (sector secundario) junto con Servicios (sector terciario), ya que son sectores fuera del sector primario y la hostelería, comercio y restauración, estas últimas actividades principales en España.

**Texto

Descripción generada automáticamente**

Tras ser agrupado:

****

* **CodigoProvincia y CCAA**: En este apartado solo he considerado agrupar Ceuta y Melilla en la Provincia de Cádiz (Andalucía). Ya que estas suponían solo una observación para cada una de ellas. Por otro lado, Murcia o las Islas Baleares, pese a tener una representación < 1.00 % he creído conveniente dejarla separada ya que al ser regiones “grandes” y uniprovinciales, tienen una representación política más grande que las de las ciudades autónomas. Además, debido al número elevado de provincias y CCAA no tiene porcentajes tan dispares (salvo con las comunidades de más de 5 provincias donde si tenemos más porcentaje de representación).

**Datos no declarados, desconocidos o erróneos.**

Podemos empezar a comprobar si en el caso de las variables AbstencionAlta se cumple 1 cuando la variable AbstencionPtge es superior al 30%, este siempre se cumple, por lo tanto, no hay ningún error (líneas 88-94).

En la variable densidad, encontramos observaciones *“?”* donde procederemos a recodificarla a NA, ya que no presenta un porcentaje importante de representación respecto a las otras variables (línea 97).

Otro paso es recodificar a valores NA aquellos valores atípicos o erróneos en variables cuantitativas sin limite, donde podemos ver observaciones máximas iguales a 99999 (línea 105).

Los valores fuera de rango son otra posible errata al anotar las observaciones y entre ella podemos ver algunas variables que representa porcentaje con valores superiores al 100% e incluso inferiores a 0% (quedando excluida en este ultimo caso el *“PobChange\_pct”).* Estos cambios pueden verse en las líneas 107-110.

Una variable interesante de analizar seria si *“totalPercentVotes”* = 100%, esta es una variable creada a partir de la operación:



Este caso también he decidido no tenerlo encuentra, debido al alto numero de veces donde el *“totalPercentVotes”* != 100%.

Tampoco apreciamos errores de escritura en las observaciones donde las variables son de tipo factor, por lo que este caso no lo vamos a tratar (líneas 117-120).

1. Análisis de valores atípicos. Decisiones.

En este apartado no considero que ninguna de las variables tenga un elevado valor de atípicos con un máximo de un 10% en las que más le afecta. Por lo tanto, he decidido transformar esos valores a NA (missing). En la imagen siguiente podemos ver el porcentaje de atípicos para todas las variables. El código resultante de este apartado podemos obtenerlos en las líneas 133-137.

Texto

Descripción generada automáticamente

1. Análisis de valores perdidos. Imputaciones.

En este apartado vamos a analizar si existe algún patrón en los missing que nos ayude a entender la falta de datos en estos y tomar decisiones en la imputación respecto a estos resultados.

Gráfico, Gráfico de barras

Descripción generada automáticamente

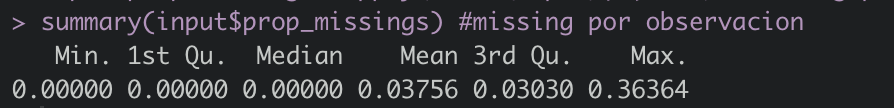
Podemos ver que si que existe un patrón en las variables:

* ***Pob2010, Population, TotalCensu, TotalEmpresas, ComercicTTEEHosteleria, Inmuebles*:** esto se debe a que todas estas variables están relacionadas con el numero de habitantes, por lo tanto, si que es normal que presenten una fuerte relación de valores perdidos. Es decir, ciertas localidades cuando no tengamos el numero de habitantes, es probable que desconozcamos el censo de la población votante y inmuebles y el porcentaje de la gente que trabaja en el comercio y hostelería (principal actividad en la población española). Por otro lado, ***Industria, Construcción*** están menos relacionadas, aunque hay que tener en cuenta que estos sectores no representan tanto porcentaje en la población.

Esta relación en las variables puede influir a la hora de imputar, por ello yo he decidido utilizar la moda para variables cualitativas y mediana para la variable cuantitativa, que más adelante veremos.

Otra cosa a tener en cuenta a la hora de imputar será la proporción de missing por observación y por variables y consideraremos si es necesario eliminar alguna de estas cuando presenten una alta proporción de missings.

* **Observaciones**. Vamos a calcular el porcentaje de missing por observación (líneas 144-145) y analizar estas proporciones para considerar si es necesario eliminaciones en estas.



Como podemos ver, no obtenemos una gran cantidad de observaciones con elementos perdidos con un máximo de un 36% de missings por observación, una media de 3% y una mediana del 0%. Por ello no será necesario aplicar las líneas (153-154) para eliminar las observaciones superiores al 50%.

* **Variables**. Para las variables podemos obtener también la proporción de missing en las observaciones (línea 147).

Texto

Descripción generada automáticamente

Como podemos ver en la imagen anterior ninguna proporción de missings es superior al 50% (máximo del 12.6% para *Servicios*), por ello el código de la línea 155 no será necesario de ejecutar.

Otro paso anterior a la imputación será la recategorización en las variables categóricas, pero puesto que ninguna de estas variables da un gran numero de missings. Por ello, he considerado no recategorizar ninguna de estas variables, creando una nueva categoría *Missing* o *Desconocido*.

El último paso para tratar los valores missings (NA) es la imputación. Para ello debemos de tener en cuenta dos variantes:

* Las variables **cuantitativas**. En vista del análisis de datos he considerado utilizar el método de imputación simple sustituyendo estos missings por la mediana. Podemos ver esta función utilizada en la siguiente imagen.



* Las variables **cualitativas**. En este caso he decidido utilizar también el método de imputación simple sustituyendo los missings por la moda. Podemos ver esta función utilizada en la siguiente imagen.



Después de efectuar estos cambios, tendremos que considerar cambiar a factor todos las variables cualitativas que han sido tipadas a carácter, tras imputar los missings.

Por último, toca comprobar estos valores (líneas 175-182), y guardar estos datos depurados en el archivo *“datosEleccionesDep”* (línea 185).

1. Transformaciones de variables y relaciones con las variables objetivo.

Una vez depurado los datos, vamos a proceder con las relaciones del las variables objetivos con las variables de entrada.

El primer paso es cargar los datos (líneas 208-212)Analizamos la relación de las variables de entradas con las variable objetivo continua y binaria. Obtenemos los siguientes gráficos.

* Relación objetivo binaria:

Imagen que contiene Gráfico

Descripción generada automáticamente

* Relación objetivo continua

Un conjunto de letras negras en un fondo blanco

Descripción generada automáticamente con confianza media

Analizando estos datos he decidido descartar CodProvincia debido a la multicolinealidad que se da con las variables CodigoProvincia y CCAA. Ambas aportan la misma información y a pesar de que cuando calculamos la V de Cramer este tiene un coeficiente mayor, no presenta una gran diferencia, sin embargo, si que puede agilizar los cálculos en las funciones posteriores significativamente debido sus 50 categorías de esta frente a las 30 de CCAA (líneas 214-224).

Las demás variables vemos como van perdiendo correlación, a demás de variar el orden de las variables en los distintos gráficos.

**Relación sobre la variable objetivo binaria**

* **Variables cualitativas**

Para este caso podemos utilizar un diagrama de mosaico donde nos muestra la influencia de la variable binaria sobre las variables cualitativas. En le ejemplo siguiente tenemos la densidad y la CCAA. Pudiendo confirmar el resultado obtenido con la V de Cramer donde la importancia de la densidad no será tan grande como el de la CCAA con respecto a la variable binaria (líneas 228-229).

Gráfico, Gráfico de barras

Descripción generada automáticamente**Gráfico, Gráfico de barras

Descripción generada automáticamente**

Otra forma de representarlas podría ser el caso de un diagrama de barras, aun que creo que sería menos efectiva para el efecto de la variable binaria sobre las cualitativas, en un solo “golpe de vista” (líneas 230-231).

* **Variables cuantitativas**

En este caso podemos utilizar dos tipos de gráficos que nos permitirá conocer muy bien el efecto de la variable binaria sobre las variables cuantitativas.

En el siguiente ejemplo podemos observar como la variable binaria no afecta mucho sobre la variable que representa el numero de parados menores de 25 años (líneas 235-236). Sin embargo, el numero de persona por inmuebles de una localidad si que es afectado por la variable objetivo binaria, la cual que representa una gran Abstención (líneas 237-238).

Gráfico, Gráfico de cajas y bigotes

Descripción generada automáticamenteGráfico

Descripción generada automáticamente

**Relación entre las variables continuas**

Para conocer el efecto de la variable objetivo continua sobre las numéricas podemos aplicar la función: *graficoCorrelacion(varObjCont,input),* observando con especial atención en las líneas rojas, que representan la relación con esta. Esta función no se ha podido aplicar a todo el conjunto de datos debido al gran numero de variables. Una solución parcial podría ser aplicarla en varias veces con diferentes variables (línea 242). Un ejemplo de esta grafica lo podemos ver en las primeras 10 variables de entrada:

Un conjunto de letras negras en un fondo blanco

Descripción generada automáticamente con confianza media

Podemos obtener la matriz de correlación entre las variables continuas (línea 244), donde como ya comentamos en apartados anteriores existe una fuerte correlación entre las variables que implican población. Por ejemplo, existe una fuerte correlación positiva entre el total de Population y el TotalCensus, o en caso inverso entre la población joven (menores de 19) y la población más anciana, esto podría ser lógico debido a la pirámide demográfica de una población de un país avanzado (inversa).

Imagen que contiene Escala de tiempo

Descripción generada automáticamente

**Transformaciones para las variables numéricas**

El siguiente paso antes de comenzar con los modelos de regresión es buscar las mejores transformaciones de las variables numéricas de nuestro modelo. Esto significa que transformamos las variables para mejorar la representación con las variables objetivos. Las variables transformadas se muestran en la imagen siguiente. Tras ello guardamos ambos dataframes. Estas operaciones las podemos ver en las líneas 247-254.

**Objetivo continua:**

Texto

Descripción generada automáticamente

**Objetivo binario:**

**Texto

Descripción generada automáticamente**

1. Detección de las relaciones entre las variables input y objetivo.

Este apartado se hizo con antes que el apartado anterior (*7. Transformaciones de variables y relaciones con las variables objetivo*), debido a la simplicidad de trabajar en depuración solo con los datos de entrada (input). Es decir, este tendría lugar en el final del apartado *4. Corrección de los errores detectados.*

Vamos a borrar de las variables objetivo descartadas, y separar las elegidas (*“AbstentionPtge* y *AbstencionAlt”*) de las variables de entrada (input). También han sido descartado el nombre de la localidad (*“Name”*) que posteriormente junto a *“CodigoProvincia”* han pasado a formar el ID de cada observación. Para formar el identificador he utilizado estos dos campos, a causa de que algunas localidades coincidían en nombre, pero estaban situadas en diferentes provincias.

Texto

Descripción generada automáticamente

9. Construcción del modelo de regresión lineal.

En este modelo de regresión lineal estará implicada la variable objetivo continua, cuyo objetivo será predecir esta con más o menos certeza dependiendo de las variables de entrada de datos.

Cargamos los datos depurados con la variable objetivo continua con la función que se muestra en la imagen a continuación (línea 273). Además, estableceremos la semilla aleatoria para separar los datos de entrenamiento para obtener el modelo (80% de los datos) y los datos test para probar el modelo obtenido (20%) (líneas 279- 283).

Adema he eliminado las variables que presentaban NA cuando creamos un modelo con todas las variables (líneas 276-277)

Texto

Descripción generada automáticamente

**Selección de modelos manuales**

* **Modelo con todas las variables (modelo1)**

Construyo un modelo un modelo preliminar con todas las variables de entrada (línea 287) y analizo este modelo con la función *summary(modelo1)* de esta manera podre comprobar las variables con mayor significación en mi modelo, conocer el error, el ajuste de *Rsq* (R2),entre otros parámetros (línea 288). A continuación, observamos el *Rsq* (R2), de nuestro modelo tras aplicarlo en el conjunto de datos train y test, de esta forma podremos comprobar el sesgo que se da en este ajuste (líneas 289-290).

Texto

Descripción generada automáticamente

La línea 292 nos calcula el AIC del modelo (41316.52) y en la línea 292 podemos calcular el numero de variables que contempla el modelo (75)

Por último, podemos plotear en una grafica la importancia de las variables para Rsq (R2) y claramente destaca la Comunidad autónoma (CCAA) sobre el resto.

Imagen que contiene Tabla

Descripción generada automáticamente

* **Modelo con variables más significativas (modelo2)**

Con las mismas funciones del apartado anterior calcularemos el *modelo2* y posteriormente probarlo en los datos de entrenamiento y test, para hallar la *Rsq (R2)* en ambos (líneas 299-304). Para este modelo se seleccionaron las variables: CCAA, *ActividadPpal, prop\_missings, ComercTTEHosteleria, Construcción, Servicios, Industria, Population, TotalEmpresas, TotalCensus.* Obtenemos los siguientes resultados:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**Selección de modelos clásicos (automático)**

Estos modelos a continuación los vamos a calcular de forma automática, aplicando los algoritmos del paquete *FuncionesRosa.R.* Este modelo se calcula en base a la cada iteración de nuestros modelos primitivo (*null*, modelo minino [línea 317] y *full*, modelo máximo, línea 318]) evaluándose el AIC/SBC del modelo resultante.

* **Modelo Step AIC**

Este modelo se calcula en base al coeficiente AIC (Criterio de información de Akaike). Partes desde el modelo mínimo y se añaden variables en cada iteración. Permite eliminar variables en caso de no mejorar este coeficiente, diferenciándose del método forward en el cual no se puede sacar variables del modelo ya calculado. Podemos ver el resultado del modelo en la siguiente imagen (líneas 322-327). Con la función *coef(modeloStepAIC)* podemos comprobar los coeficientes del modelo.

Texto

Descripción generada automáticamente

* **Modelo Back AIC**

Este modelo también se calcula en base al coeficiente de AIC. Partes del modelo máximo y en cada iteración se van eliminando las variables, sin opción de volver a incluir esa variable en el modelo. Los resultados los podemos ver en la siguiente imagen (líneas 329-333).

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

* **Modelo Step SBC**

Este método funciona en cada iteración igual que el anteriormente mencionado en el AIC, pero con la diferencia de que modelo se calcula en base al coeficiente SBC (Criterio bayesiano de Schwarz). El resultado lo podemos ver en la imagen a continuación (líneas 335-339).

**Texto

Descripción generada automáticamente con confianza media**

* **Modelo Back SBC**

Este método al igual que el modelo anterior se calcula igual que su similar en el método AIC (Modelo Back AIC), pero estableciendo como criterio el coeficiente SBC. El resultado lo vemos ver en la imagen siguiente (líneas 341-345).

**Texto

Descripción generada automáticamente con confianza media**

* **Modelo Step AIC con todas las posibles interacciones**

Este utilizará el mismo método que el que genera *Modelo Step AIC*, con la diferencia de que este incluirá todas las posibles iteraciones entre las variables del dataframe. Con este método a pesar de ser más ajustado, estaremos sobreparametrizando. El resultado lo podemos ver a continuación (líneas 351-355).

Texto

Descripción generada automáticamente

* **Modelo Step SBC con todas las posibles interacciones**

Para generar el modelo en este método trataremos de utilizar el mismo método dispuesto en el *Modelo Step SBC* con todas las interacciones posibles entre las variables de nuestro dataframe. En la imagen siguiente podemos ver el resultado (líneas 357-361).

Texto

Descripción generada automáticamente

* **Modelo Step AIC con todas las posibles interacciones, añadiendo variables transformadas**

Por último, además de probar toda la posible interacción de nuestras variables, vamos a añadir las variables numéricas transformadas y de nuevo calcular todas las interacciones y posteriormente aplicar el método *Step AIC,* para hallar nuestro modelo. Los resultados de este modelo los podemos ver en la imagen a continuación (líneas 367-371).

**Texto

Descripción generada automáticamente**

* **Modelo Step SBC con todas las posibles interacciones, añadiendo variables transformadas**

Finalmente aplicar lo mismo que el modelo anterior, pero usando el método *Step SBC,* para crear nuestro modelo. En la siguiente imagen podemos ver los resultados obtenidos (líneas 373-378).

**Texto

Descripción generada automáticamente**

**Modelo aleatorio**

El siguiente método es el aleatorio para conseguir el modelo más estable para nuestros datos. Este consiste repetir el método de selección de variables clásico sobre diferentes submuestras de nuestros datos. En base a todos estos modelos generados elegiremos el más repetido, y por tanto el más estable (ya que aparecerán para diferente subconjunto de datos).

Tras ejecutar el algoritmo de las líneas 381-392, observamos que ningún modelo se repitió ni una sola vez, es decir todos los modelos propuestos, solo fueron generados una vez. Este algoritmo se aplicará solo al 70% de los datos para agilizar el algoritmo, ya que el tiempo computacional es bastante alto. *(Por longitud de los datos no puedo enseñar todos los resultados obtenidos por consola).*

Texto

Descripción generada automáticamente

**Modelo híbrido**

He intentado generar un modelo con ayuda de los modelos generados con el método clásico, pero no he podido mejorarlo ya que esta suficientemente optimizado. He intentado añadir variables, pero va aumentando el sesgo y el numero de variable, además el coeficiente de *Rsq (R2).* Por ello he decidido no incluirlo en la validación cruzada para optimizar el algoritmo. El mejor modelo que he podido crear se encuentra en las líneas 394-400.

**Selección del modelo ganador**

Para la selección del modelo ganador vamos a utilizar el algoritmo de la validación cruzada con los métodos que a priori yo he considerado como posibles ganadores. Estos modelos que yo he seleccionado son: *modelo1, modeloStepAIC (mismo resultado que modeloBackAIC), modeloStepAIC\_int, modeloStepBIC\_int, modeloStepAIC\_transInt, modeloStepBIC\_transInt.* En este orden se representan los modelos a continuación en la imagen. El algoritmo para poder representar estos modelos se encuentra en las líneas 402-424.

**Gráfico, Gráfico de cajas y bigotes

Descripción generada automáticamente**

Viendo estos resultados junto a sus estadísticas en la imagen inferior. He decidido elegir el Modelo 5 (*modeloStepBIC\_transInt*) ya que, pese a presentar un R2 inferior es el que menor variabilidad presenta, a demás de tener 43 variables frente a las 167 y 163 de los Modelos 4 y 2 respectivamente.

**Texto

Descripción generada automáticamente**

Por ultimo evaluamos el modelo ganador podemos examinar los coeficientes (línea 209) con la función *coef(ModeloGanador)* y ver el R2 para los modelos de entrenamiento y test, 0.4193538 (train) y 0.4111261 (test). A continuación se muestra el summary del modelo ganador y los coeficientes seleccionados para su interpretación (línea 429)

Texto

Descripción generada automáticamenteTexto

Descripción generada automáticamenteTexto

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente con confianza media

* **Variable continua:** Elijo la variable **Age\_over65\_pct,** quePosee un valor de -1.975186e-01, esto quiere decir que, por cada aumento unitario en la variable predictora,variable objetivo continua (**AbstentionPtge**) va a disminuir -0.1975186.
* **Variable binaria:** Utilizamos la variable **CCAA** y la categoría **Canarias (**la categoría de referencia es **Andalucía**). El valor del coeficiente seleccionado para esta categoría es de 4.486972e-00. Esto nos dice que la media de **AbstentionPtge** en Canarias esde4.486972 unidades superior que en Andalucía

10. Construcción del modelo de regresión logística.

Este modelo se centra en la variable objetivo binaria. Vamos a utilizar este modelo ganador para predecir esta variable objetivo con más o menos certeza dependiendo de las variables de entrada.

Una vez cargado los datos y (línea 449) procedemos a hacer la partición entre los datos de entrenamiento y test (líneas 458-462). La forma de calcular estos modelos es igual a la del apartado anterior (regresión lineal), pero esta vez, en lugar de usar la función *lm(),* usaremos la función *glm()* del paquete de funciones *FuncionesRosa.R.*

**Selección de modelos manuales**

* **Modelo con todas las variables (modelo1)**

Modelo un modelo preliminar con todas las variables, a excepción de las transformadas (líneas 465-470). Obtenemos los siguientes resultados.

Texto

Descripción generada automáticamente

* **Modelo con variables más significativas (modelo2)**

Para este modelo he seleccionado las variables más significativas al calcular V de Cramer (línea 472-477). Los resultados obtenidos se muestran a continuación.

Texto

Descripción generada automáticamente

**Selección de modelos clásicos (automático)**

Aplicando los algoritmos del paquete *FuncionesRosa.R,* como en la regresión lineal*,* los modelos se calcula en base a la cada iteración de nuestros modelos primitivo (*null*, modelo minino [línea 317] y *full*, modelo máximo, línea 317]) evaluándose el AIC/SBC del modelo resultante.

Los algoritmos son iguale para cada uno de los modelos que su homónimo en el ya explicado apartado anterior. Por ello mostrare solamente los resultados y las líneas donde podemos encontrarlos.

* **Modelo Step AIC**

Líneas 484-489

Texto

Descripción generada automáticamente

* **Modelo Back AIC**

Líneas 491-495

Texto

Descripción generada automáticamente

* **Modelo Step SBC**

Líneas 497-501

**Texto

Descripción generada automáticamente con confianza media**

* **Modelo Back SBC**

Líneas 503-507

**Texto

Descripción generada automáticamente**

* **Modelo Step AIC con todas las posibles interacciones**

Líneas 513-517

**Texto

Descripción generada automáticamente**

* **Modelo Step SBC con todas las posibles interacciones**

Líneas 519-524

Texto

Descripción generada automáticamente con confianza media

* **Modelo Step AIC con todas las posibles interacciones, añadiendo variables transformadas**

Líneas 545-549

**Texto

Descripción generada automáticamente**

* **Modelo Step SBC con todas las posibles interacciones, añadiendo variables transformadas**

Líneas 551-555

**Texto

Descripción generada automáticamente**

**Modelo aleatorio**

Este modelo, no he sido capaz de calcularlo debido al tiempo de procesamiento para estos cálculos. Mi ordenador ha estado 3 horas aproximadamente de cálculos y no ha sido capaz de procesarlo, por ello he decidido abortar el calculo. En vista del resultado con la regresión lineal y teniendo en cuenta la gran cantidad de variables es probable que no se repitiera ningún modelo, o apenas tuviera repeticiones sin proporcionar un modelo realmente estable que se sea elegido un gran numero de veces sobre los demás. Tras abortar los cálculos observamos que los primeros modelos aportados solo se repiten una vez.

Texto

Descripción generada automáticamente

**Selección del modelo ganador**

Para la selección de este modelo ganador he decidido enfrentar todos los modelos menos los manuales, ya que no creo que puedan igualar a priori a los calculados de forma automática con los métodos clásicos.

**Texto

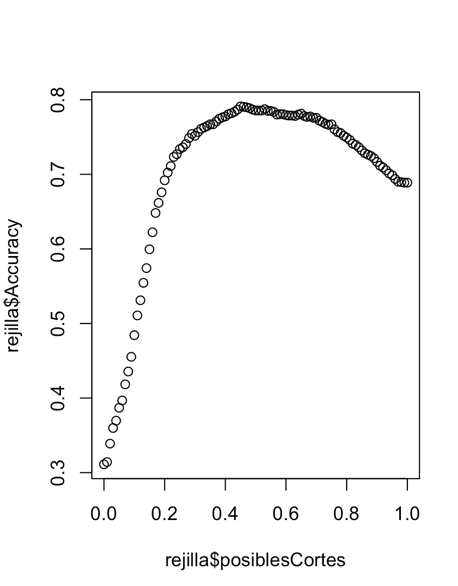
Descripción generada automáticamente**

Gráfico, Gráfico de cajas y bigotes

Descripción generada automáticamente

Viendo estos resultados junto a sus estadísticas en la imagen inferior, observamos que el área sobre la curva ROC es mayor en los modelos 5 y 7 (ambos pese a estar calculados sobre diferentes métodos, son el mismo modelo) y por otro lado los modelos 6 y 8 (ambos también el mismo modelo). He decidido seleccionar estos últimos Modelo 6 y Modelo 8 como el ganador ya que pese a tener una variabilidad mayor, esta no es mu elevada a sus modelos competidores 5 y 7, además he tenido muy en cuenta los 30 parámetros del modelo ganador frente a los 155 del modelo 5 y 7.

El siguiente paso una vez seleccionado el modelo ganador, será generar una rejilla para entre 0 y 1 con una división de 0.1 y mediante el índice de Youden y la Accuracy (precisión) decidir cual de ellos será nuestro punto de corte. Es importante destacar que estos resultados para nuestros datos solo se darán en la semilla de aleatoriedad dada en el que hacemos el análisis.

Gráfico

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Creo que para la obtención resulta más importante maximizar los verdaderos negativos, es decir saber con mayor exactitud en que poblaciones no se dará una obtención alta con una exactitud del 90%. Para ello me fio en la Specifity y selecciono el punto 0.45, Al elegir el punto con mayor Accuracy podemos asegurar que este también tiene una mayor tasa de aciertos locales demás. A continuación, enseñamos el summary del modelo ganador.

Texto

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto

Descripción generada automáticamente**Texto

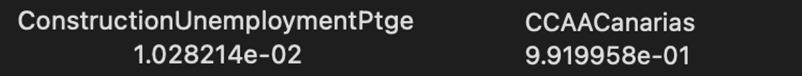
Descripción generada automáticamente**

Para este modelo de regresión logística, al igual que en la lineal la variable CCAA adquiere una gran importancia sobre el resto de las variables del modelo. Podemos ver el grafico ejecutando la línea 632. Podemos evaluar este modelo con los datos test (línea 638).

Gráfico

Descripción generada automáticamente

Por último, interpretar los coeficientes del modelo ganador



* **Variable continua:** Utilizamos la variable **ConstructionUnemploymentPtge.** Adquiere un valor de 1.028214e-02 => *e* 0.01028214 = 1.010335, esto quiere decir que por cada unidad que aumenta la variable seleccionada predictora (ConstructionUnemploymentPtge), el OR (odds-ratio) de **AbstentionPtge** aumenta en 1.010335.
* **Variable binaria:** Utilizamos la variable **CCAA** y la categoría **Canarias (**la categoría de referencia es **Andalucía**). El valor del coeficiente seleccionado para esta categoría es de 9.919958e-01 => *e* 0.9919958 = 2.696611, esto quiere decir que la OR de la probabilidad de abstención en Canarias es 2.696611 veces mayor que en Andalucía.

11. Conclusión

Esta práctica considero que ha sido bastante interesante ya que me ha ayudado a entender mejor los modelos de regresión logística y lineal, pilares fundamentales para el Machine Learning y Deep Learning y procesamiento de redes neuronales. También, me he sentido bastante cómodo en esta practica, gracias al paquete proporcionado por la profesora *“FuncionesRosa.R”,* el cual nos ha ayudado a ahorrar tiempo sus algoritmos.