Bank Marketing Dataset

Mineria de dades

Participantes:
Fèlix Bosch Bosch (E1)
David Botín Garcia-Planas (E2)
Marcel Canals Codina (E1)
Andreu Companys Rufian (E1)
Andrés Déniz Barnés (E2)
Marc Larraz (E2)
Joan Martí Gensana (E1)
Adrià Pérez García (E2)
Ivan Varea Abarca (E2)

Fecha de entrega:
16 de diciembre del 2021

ÍNDICE DE CONTENIDOS

Diagrama de Gantt 6

Distribución de tareas 6

Plan de riesgos 8

Análisis descriptivo de los datos iniciales 10
- Características del cliente: 10
- Contacto con el cliente en las campañas de marketing 13
- Contexto socioeconómico 18
- Otros atributos 21

Preprocessing 21
- Corrección de etiquetas 22
- Balanceamiento de los datos 22
- Importación como NA's los "unknown" 23
- Transformación de la variable "pdays" 24
- Eliminación de variables 24
- Tratamiento de NA's 25
  - Método "KNN" para valores perdidos 25
  - Método "MICE" (Multivariate Imputation by Chained Equations) 25
- Estandarización de variables numéricas 25
- Detección de outliers 26
- Estudio de la correlación entre variables predictoras 26

Análisis descriptivo de los datos preprocesados 27
- Características del cliente 28
- Contacto con el cliente en las campañas de marketing 32
- Contexto socioeconómico 32
- Otros atributos 35
- Variable respuesta (Y) 36

Diseño de los dos procesos de minería de datos 36
- Metodologia 37
- Equipo 1 38
  - ACP: 38
    - Tría del número de componentes principales 38
    - Proyección de las variables numéricas 39
    - Proyección de la variable respuesta 43
    - Proyección de todas las variables 43
  - Árbol de decisión 44
  - KNN 45
  - Naive Bayes 46
- Equipo 2 48

ACM 48
Clasificación de las variables 48
Support Vector Machine (SVM) 53
XGBoost 57
Random Forest 64

Análisis comparativo de Modelos/ Conclusiones 67
Todos los modelos ajustados tienen un acierto en la predicción superior al nivel basal (línea discontinua). El modelo XGBoost consigue el accuracy promedio más alto, seguido muy de cerca por SVM y Random Forest. Para determinar si las diferencias entre ellos son significativas, sería necesario recurrir a un análisis de varianza. 67

Conclusiones 68
Métodos supervisados 68
Métodos no supervisados 69

Plan de trabajo real 69

Distribución de tareas real 71

Anexo 72
Descripción completa de la variables 73

Fuente del repositorio

Repositori github:
https://github.com/99adria99/Mineria.git

Fuente:

[Moro et al., 2014] S. Moro, P. Cortez and P. Rita. A Data-Driven Approach to Predict the Success of Bank Telemarketing. Decision Support Systems, Elsevier, 62:22-31, June 2014

https://archive.ics.uci.edu/ml/datasets/Bank+Marketing

Descripción de la base de datos

Los datos incluidos en este dataset pertenecen a campañas de marketing de una institución bancaria portuguesa. Estas campañas de marketing se hicieron en base a llamadas telefónicas, frecuentemente más de una, para conocer si el producto (depósito a plazo) fué finalmente suscrito o no. Para ello existe una variable regresora “y”, variable binaria que toma “si” en el caso en el que el cliente suscriba el depósito a plazo, y “no” en el caso en el que el cliente no lo suscriba.

El objetivo del trabajo es mediante un modelo de clasificación, predecir si un cliente suscribirá un depósito a plazo o no.

La base de datos utilizada es “bank-additional.csv”, esta base se ha extraído de una base más grande llamada “bank-additional-full.csv”, la cuál contiene 41.188 elementos de 20 variables distintas. De estos datos se extrae de forma aleatoria nuestra muestra, que representa solamente un 10% de los elementos, de forma que el dataset consolidado consta de 4.118 observaciones.

Output Variable (Variable Respuesta)
Y : El cliente se ha suscrito a un depósito a plazo? (categórica binaria)

Estructura básica de la matriz de datos
El dataset está compuesto por 4.119 observaciones(clientes) y 21 variables, de las cuáles 10 son de tipo numéricas y 11 de tipo categóricas.

Hemos clasificado las variables en grupos distintos según sus características. En azul representamos los niveles de las variables que son de tipo categóricas

- **Características del cliente**
  - **age** [numérica]: edad del cliente

- **job** [categórica]: tipo de trabajo
'admin.', 'blue-collar', 'entrepreneur', 'housemaid', 'management', 'retired', 'self-employed', 'services', 'student', 'technician', 'unemployed', 'unknown'
- **marital** [categórica]: estado civil
'divorced', 'married', 'single', 'unknown'; Note: 'divorced' means divorced or widowed
- **education** [categorical]: nivel de estudios
'basic.4y', 'basic.6y', 'basic.9y', 'high.school', 'illiterate', 'professional.course', 'university.degree', 'unknown'
- **default** [categórica]: ¿tiene crédito pendiente de pago?
'no', 'yes', 'unknown'
- **housing** [categórica]: ¿tiene un préstamos hipotecarios?
'no', 'yes', 'unknown'
- **loan** [categórica]: ¿tiene un préstamo personal? *'no', 'yes', 'unknown'*

- **Contacto con el cliente en las campañas de Marketing**
  - **contact** [binaria]: tipo de comunicación mediante la que se ha contactado
'cellular', 'telephone'
  - **month** [categórica]: mes del año del último contacto
'jan', 'feb', 'mar', …, 'nov', 'dec'
  - **day_of_week** [categórica]: día de la semana del último contacto
'mon', 'tue', 'wed', 'thu', 'fri'
  - **duration** [numérica]: duración de la última comunicación, en segundos
  - **campaign** [numérica]: cuántas veces que se ha contactado con este cliente durante la campaña actual (incluída la última llamada)
  - **pdays** [numérica]: días que han pasado desde la última vez que se contactó al cliente por una campaña anterior.
  - **previous** [numérica]: cuántas veces se ha contactado con este cliente con anterioridad a esta campaña por cualquier motivo.

- **Contexto socioeconómico**
  - **emp.var.rate** [numérica]: tasa de variación de empleo (indicador trimestral)
  - **cons.price.idx** [numérica]: Índice de precios al consumidor (indicador mensual)
  - **cons.conf.idx** [numérica]: Índice de confianza del consumidor (indicador mensual)
  - **euribor3m** [numérica]: euribor a 3 meses (indicador diario)
  - **nr.employed** [numérica]: número de empleados (indicador trimestral)

- **Otros atributos**
  - **poutcome** [categórica]: resultado de la anterior campaña de marketing
'failure', 'nonexistent', 'success'

- **Variable Respuesta**
  - **y** [binaria]: ¿ha suscrito el cliente el depósito a plazo ofertado?

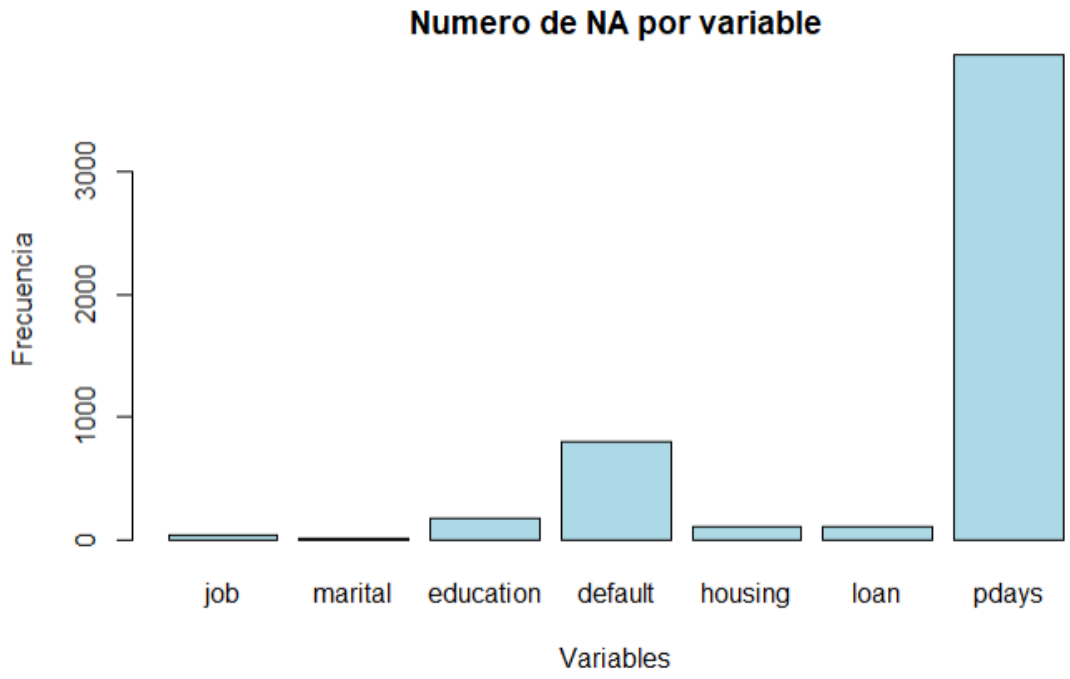
'yes', 'no'

Número de casillas missing:

La base de datos tenía los missings como 'unknown' y en el caso de "Pdays" '999' y lo hemos cambiado por 'NA'.

Número de casillas missing por variable:

Variable	Job	Marital	Education	Default	Housing	Loan	Pdays
Nº Na's	39	11	167	803	105	105	3959
Relativo	0.946%	0.267%	4.054%	19.495%	2.549%	2.549%	96.115%

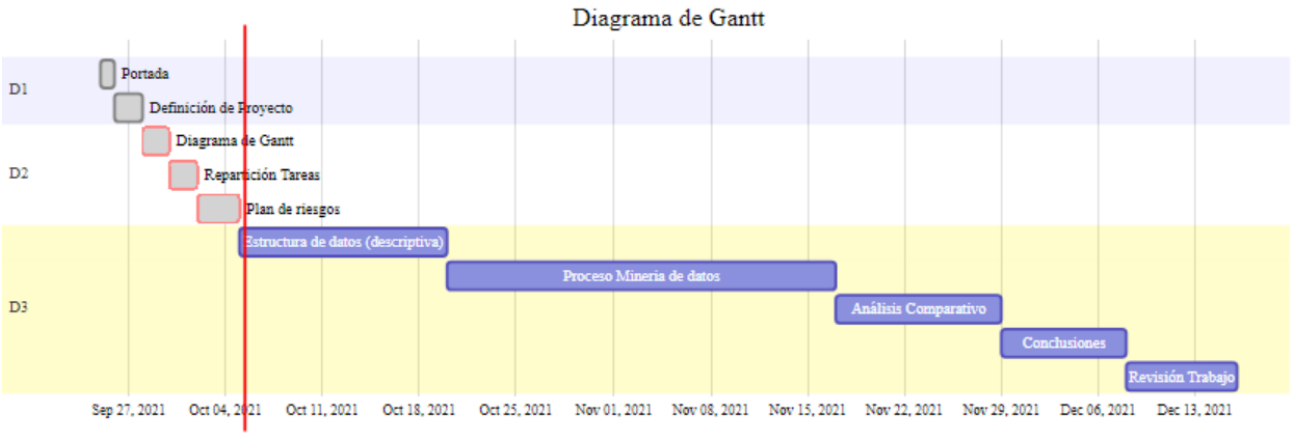


La tabla anterior muestra la cantidad de missings por variable que aparece en la base de datos. La variable "pdays" contiene un 96,11% de Missings, que representan un 76,3% de todos los missings de la base de datos.

En total, todos los NA's representan un 5,99% de la base de datos.

Diagrama de Gantt

El diagrama de Gantt es un herramienta que nos ayudará a organizarnos combinando la parte de distribución de tareas y la distribución temporal. La línea vertical roja marca el día en el que nos encontramos para tener una referencia y así facilitar la comprensión.



Distribución de tareas

La distribución de tareas es una tarea que nos servirá para poder organizarnos y tener una perspectiva más amplia que tenga en cuenta todos los puntos del trabajo.

Recordemos que la bipartición del grupo es la siguiente:
E1 (equipo 1): Fèlix Bosch, Marcel Canals, Andreu Companys y Joan Martí
E2 (equipo 2): David Botín, Andrés Déniz, Marc Larraz, Adrià Pérez y Ivan Varea

Tareas:	Encargado:
Estructura de los datos y descriptiva	
Un párrafo explicando la motivación del trabajo	Ivan Varea
Descripción formal de la estructura de datos.	Marc Larraz y Andrés Déniz
Análisis descriptivo univariante inicial de todas las variables.	Fèlix Bosch y Joan Martí
Descripción detallada del proceso de preprocessing de datos con una	Marcel Canals y Andreu Companys

justificación de todas las decisiones decididas.	
Análisis descriptivo univariante de los datos preprocesados y discusión sobre la aleatoriedad de los datos faltantes.	David Botín y Adrià Pérez
Diseño de los dos procesos de minería de datos a seguir	
Representación gráfica de las técnicas de minería de datos que se enlazarán a lo largo del proceso. Este diagrama tiene que representar el proceso completo, incluyendo el preprocesamiento. Cada división tiene que poner en su diseño las operaciones que haya tenido en cuenta y si trabaja con las mismas variables o no que con la otra división	E1 y E2
Justificación del flujo representado	E1 y E2
Proceso de minería de datos de la división 1	
Resultados de aplicar al menos un método de cada família vista en clase (profiling, asociativo, discriminante, predictivo)	E1
Procés de mineria de dades de la divisió 2	
Resultados de aplicar al menos un método de cada família vista en clase (profiling, asociativo, discriminante, predictivo)	E2
Análisis comparativa	Todos
Conclusions generals	Todos
Plan de trabajo REAL	E1 y E2
Scripts d'R utilizados	Todos

Plan de riesgos

Posible riesgo	Com evitarlo	Mitigación	Grado de peligrosidad
Miembro o miembros inoperativos por enfermedad, empleo o exámenes	El riesgo es bastante imprevisible. Depende de distintos factores no controlables.	-Valoración del tiempo de baja -Repartición de las tareas de los miembros inoperativos entre los que sí están -Asignación entre los miembros operativos en función de la carga de trabajo	**ALTO**
Mala comunicación	Se intenta crear un clima de confianza y de comunicación adecuado	-Valorar otros canales de comunicación -Intentar asignar a alguien responsable de gestionar el grupo	**LEVE**
Mala coordinación	El diagrama de Gantt y la repartición de tareas nos ayudará	-Evitar duplicados de archivos -Cada miembro tiene que tener presente que parte del trabajo realiza cada uno para saber que tiene que pedir y a quién en cada momento -Mejorar la comunicación -Hacer grupos más pequeños para cada tarea	**LEVE**
Tareas mal asignadas	La distribución de tareas realizada se intenta ajustar a cada uno de los miembros	-Se puede considerar reorganizar las tareas -El grupo puede ayudar a un miembro a realizar mejor una tarea	**MEDIO**
Tareas mal realizadas	Trabajar con más cuidado y preparar mejor las tareas	-Se puede considerar repasar los apuntes para evitar errores -Preguntar al profesor que nos guíe para conseguir nuestros objetivos	**LEVE**
Alguien cambia a	Inevitable	-Reasignación de las tareas para compensar la baja de uno de los miembros	**MEDIO**

evaluación única			
No entregar una tarea a tiempo	Planificar de mejor manera y controlar que todo el mundo haga su tarea	-Buscar cual es la fuente y ponerle remedio -Mejorar y comprobar el realizamiento de las tareas de cada uno de los miembros	**ALTO**

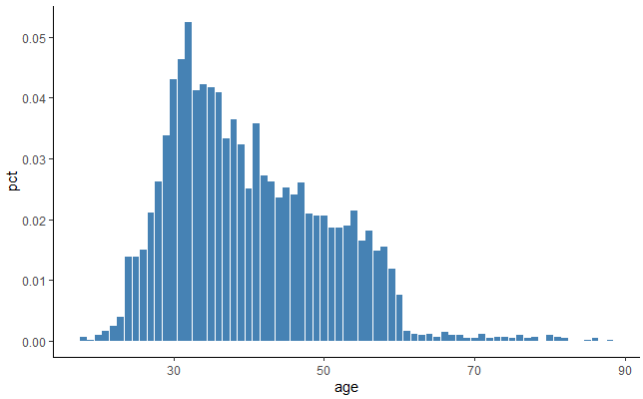
Análisis descriptivo de los datos iniciales

Para todas las variables hemos realizado un análisis bivariante entre dicha variable en función de la variable respuesta “y” y un análisis univariante.

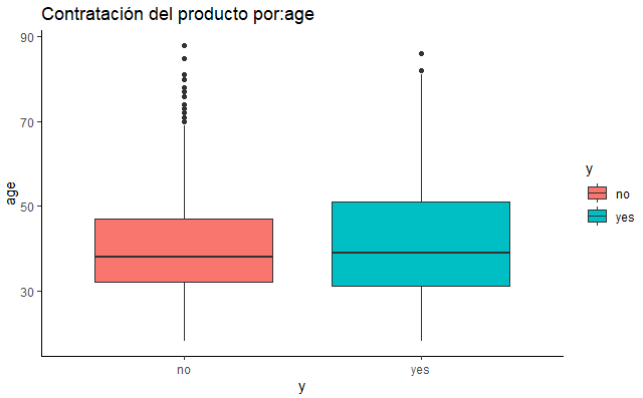
Características del cliente:

- **Númericas:**

1. **age**



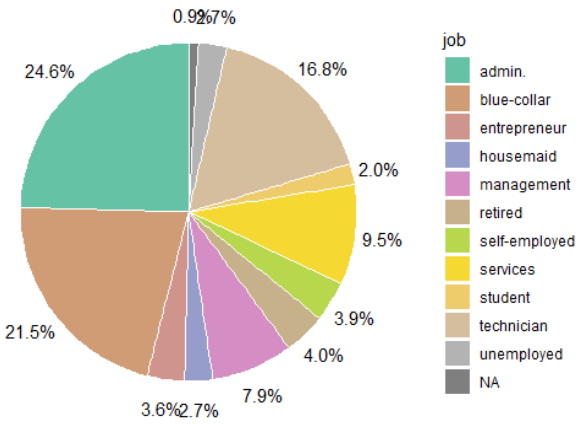
La mayoría de los clientes presenta edades comprendidas entre los 32 y 47 años, con un valor mínimo de 18 (mayoría de edad) y una máxima de 88.



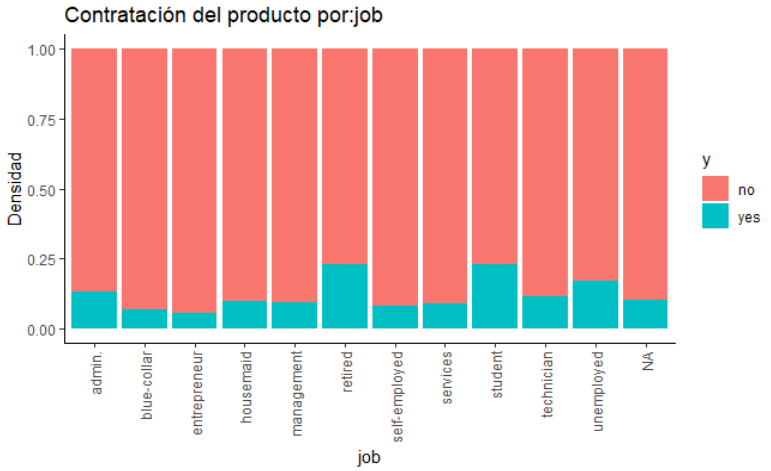
No existen grandes diferencias en la edad de los clientes según si se suscriben al depósito.

- **Categóricas:**

1. **job**

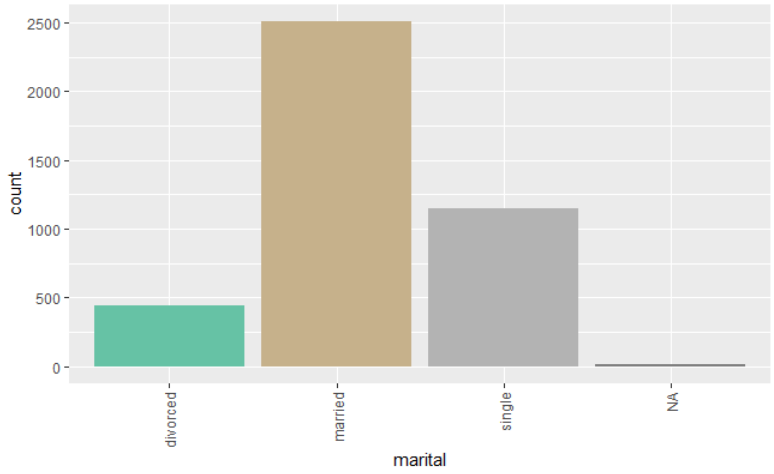


La mayoría de clientes trabajan en la sección de la administración o són trabajadores “blue collar” anglicismo que hace referencia a aquellas personas que desempeñan labores manuales (agricultura, construcción…)

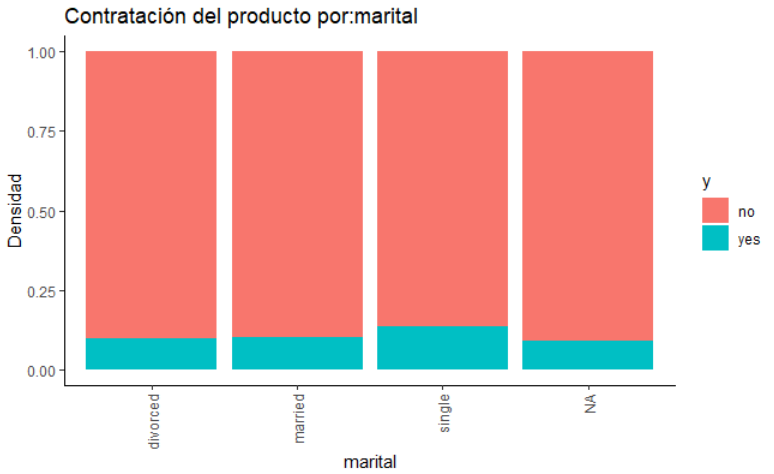


Destacamos los jubilados y los estudiantes como las ocupaciones que más tasa de suscripción al depósito presentan.

2. marital

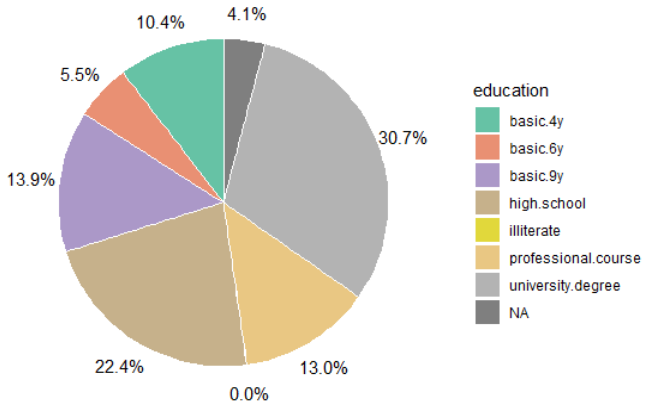


Más de la mitad de los clientes figuran como casados.

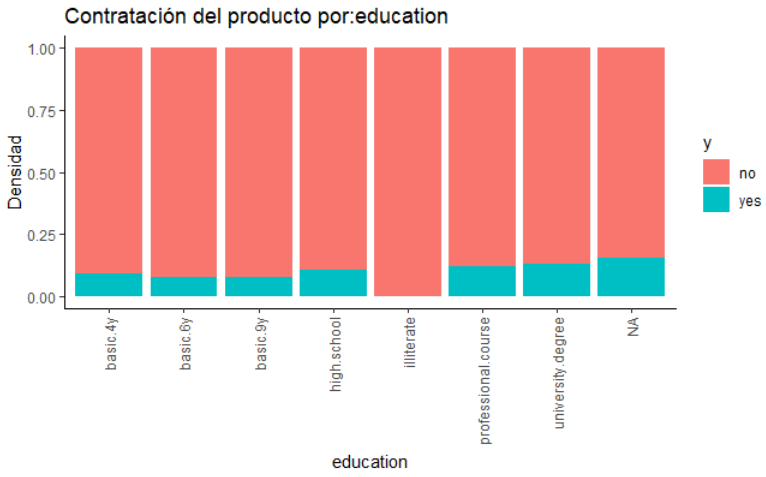


Pese a la leve tendencia de mayor suscripción al depósito para los solteros, no existen diferencias notorias.

3. education

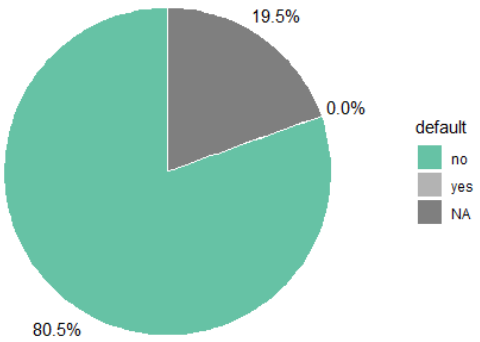


Casi un tercio de la muestra posee estudios universitarios.

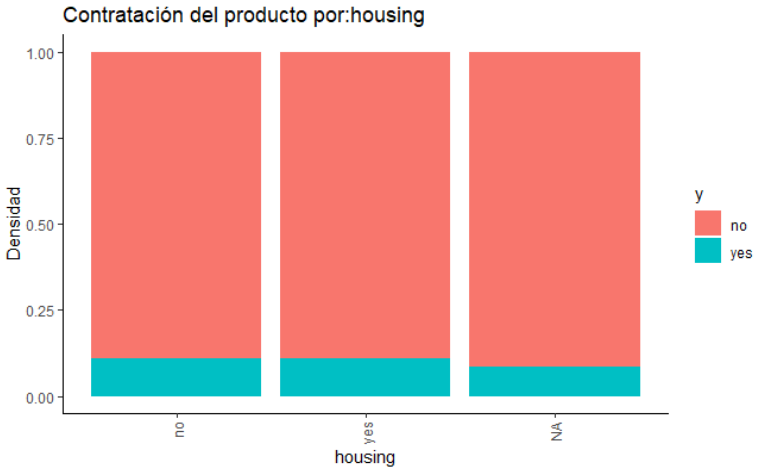


La categoría “Not Assigned” presenta la mayor tasa de suscripción.

4. housing



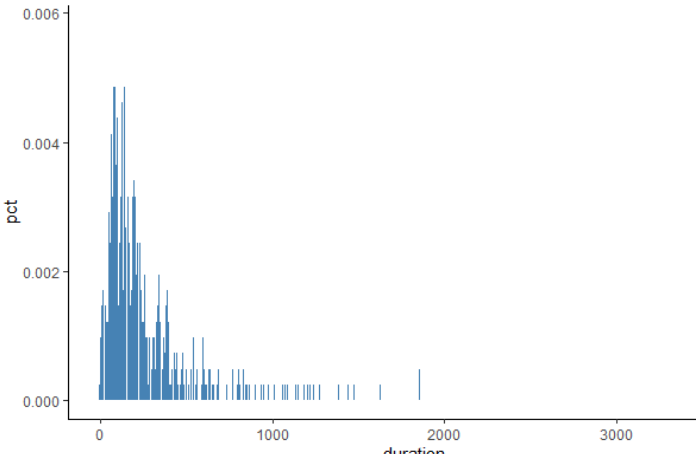
La gran mayoría de clientes de la muestra no tienen contratado un préstamo hipotecario.



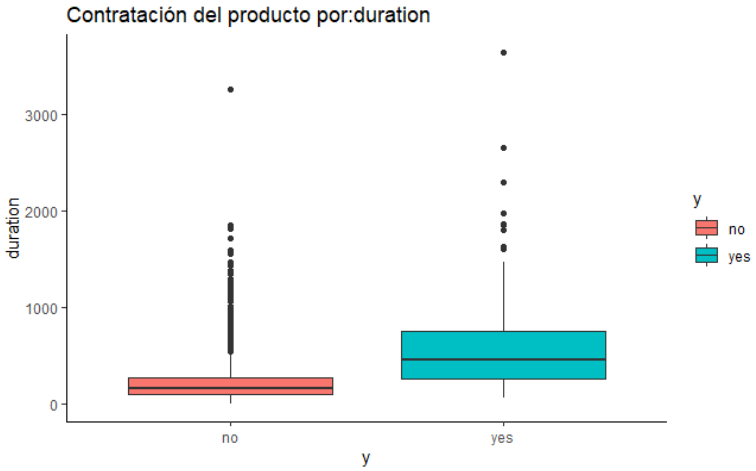
Observamos que poseer un préstamo hipotecario no influye en la decisión de suscribirse al depósito.

Contacto con el cliente en las campañas de marketing

- **Númericas:**
  1. **duration**

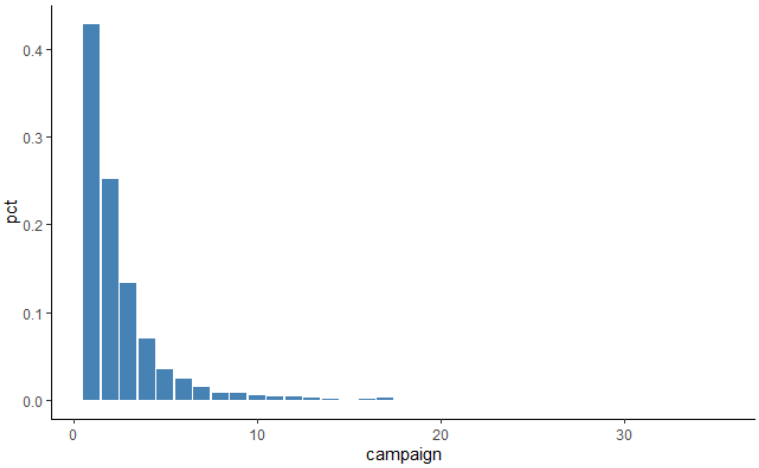


La media de duración de las llamadas está cerca de los 4 minutos, alargándose raramente por encima de los 6 minutos.

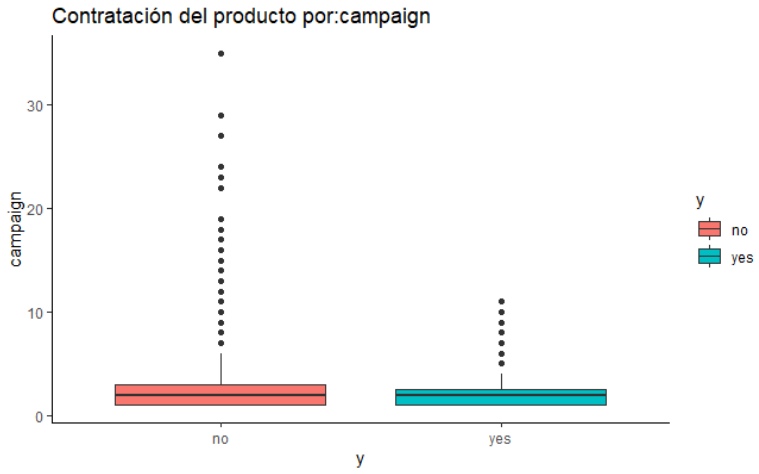


Tiene sentido que la duración de las llamadas para los clientes que se suscriben al depósito sea superior a la de los que no.

2. campaign



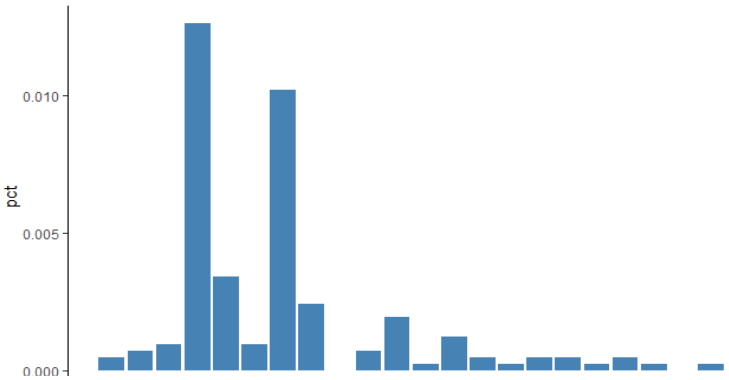
La mayoría de los clientes contactan 1 o 2 veces con el banco durante la duración de la campaña.



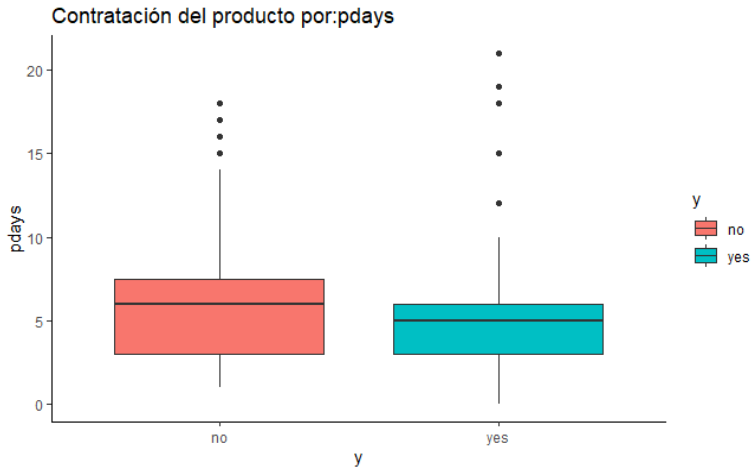
No notamos grandes diferencias entre el volumen de suscripciones o no al depósito según el número de veces que contactan con el banco.

3. pdays

Para la variable pdays solo hemos usado los clientes con los que se ha contactado para una campaña anterior, ya que la mayoría de clientes no han sido contactados. Solo hay 70 observaciones.

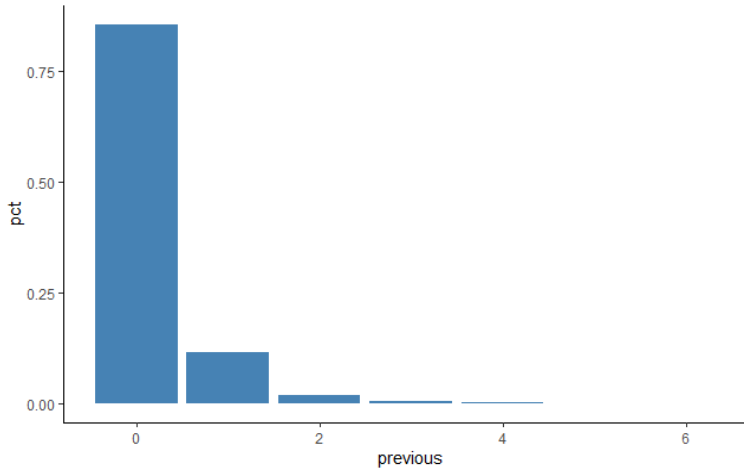


Como podemos ver en el gráfico normalmente pasan entre 3 o 6 días desde el último contacto al cliente por una campaña anterior.

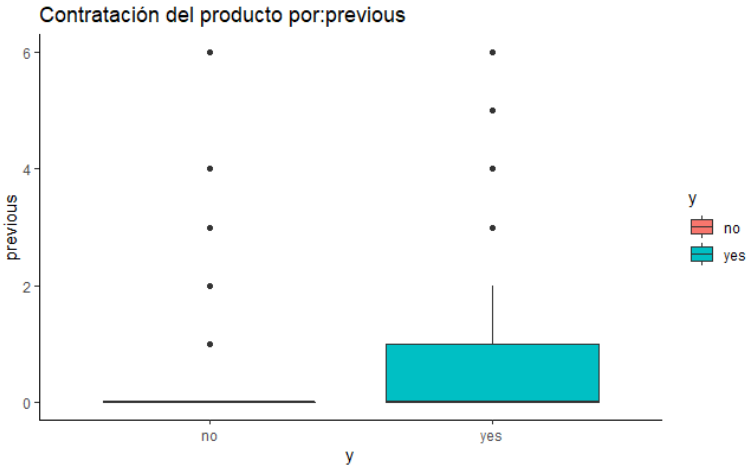


A más días transcurren desde la última vez que se contacta con los clientes, mayor es la tasa de rechazo a suscripciones al depósito.

4. previous



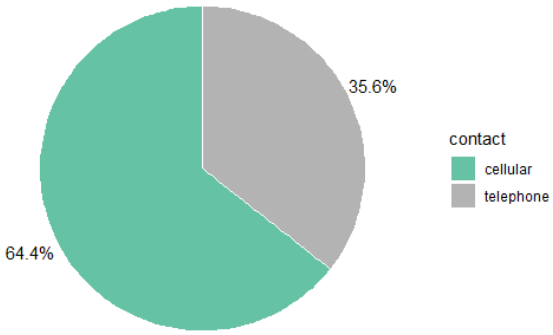
Para la mayoría de clientes actuales, no se ha contactado con anterioridad con ellos.



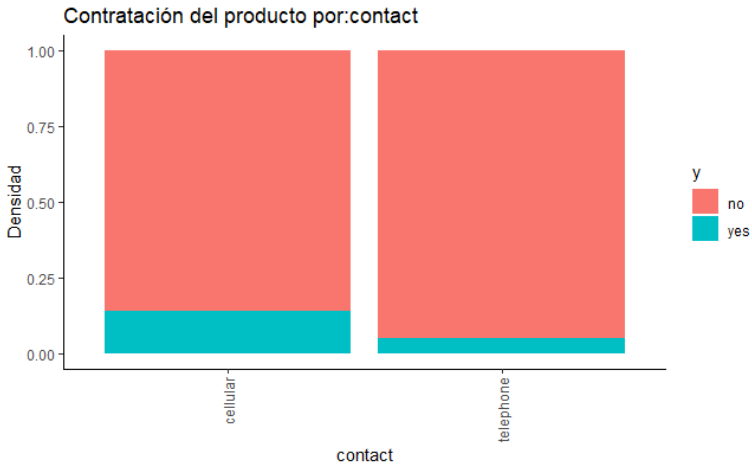
Los clientes contactados con anterioridad muestran una mayor tendencia a suscribirse al depósito.

- **Categóricas:**

1. contact

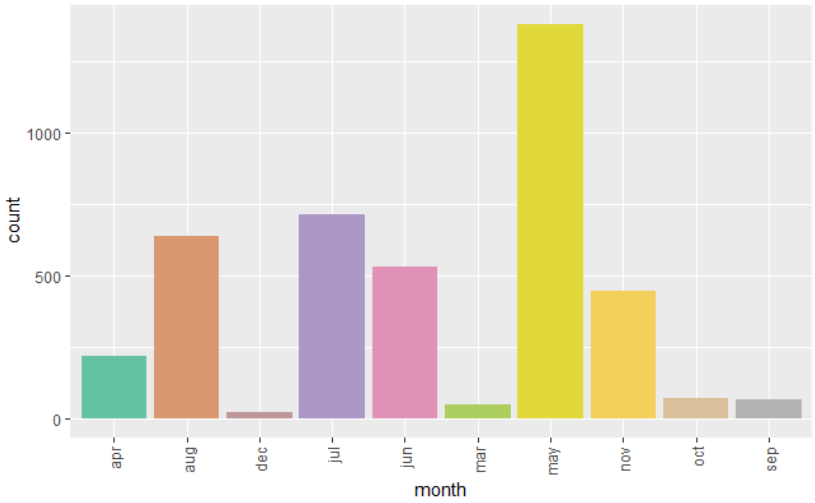


Cerca de dos terceras partes de los clientes son contactas a su móvil, mientras que el tercio restante fué a través del fijo.

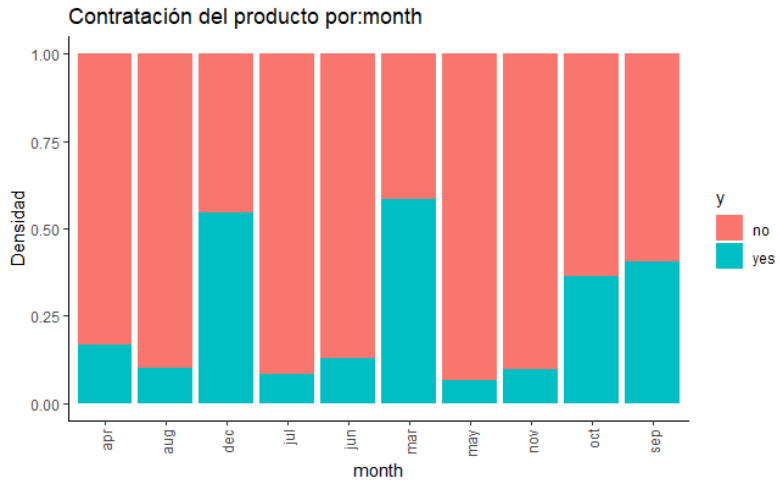


Existe una mayor tasa de suscripciones en aquellos clientes contactos a su celular.

2. month

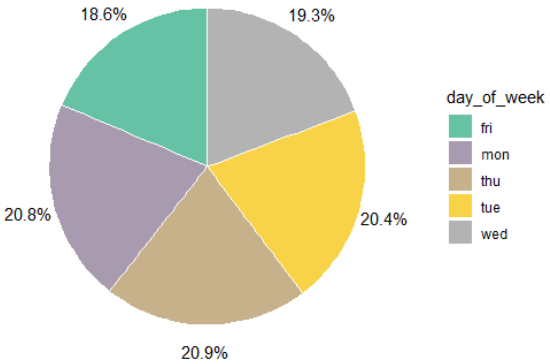


Mayo es el mes donde más clientes fueron contactados por última vez

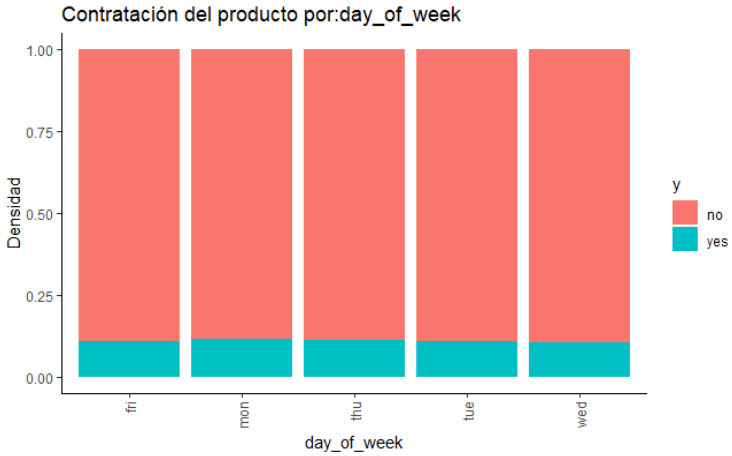


Los clientes contactados por última vez durante los meses de marzo, diciembre, septiembre y octubre presentan las mayores tasas de suscripciones al depósito respectivamente.

3. day_of_week



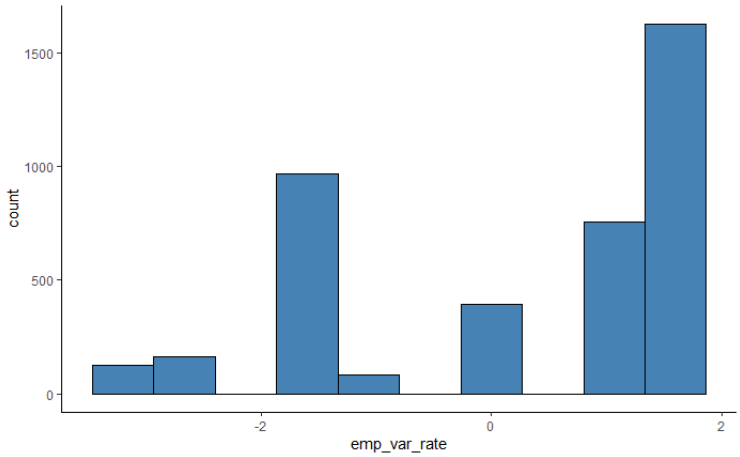
Se contacta con los clientes de forma equitativa durante los 5 días laborables de la semana.



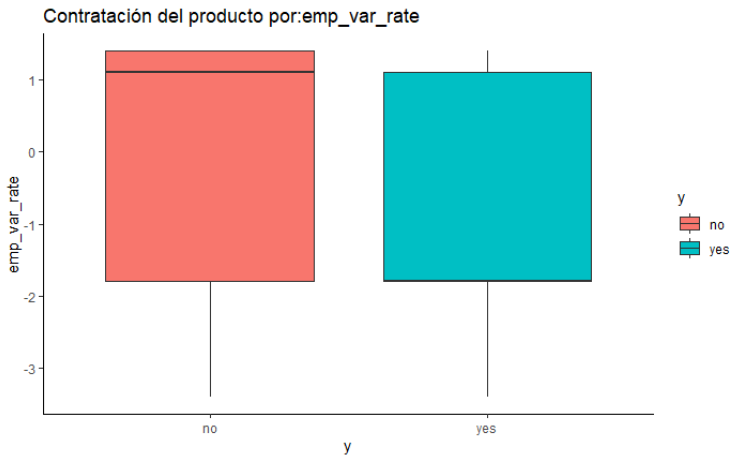
Las tasas de efectividad de suscripción de los clientes, no varían según el día de la semana en el que se ha establecido el contacto con el banco.

Contexto socioeconómico

- **Númericas:**
  1. **emp.var.rate**

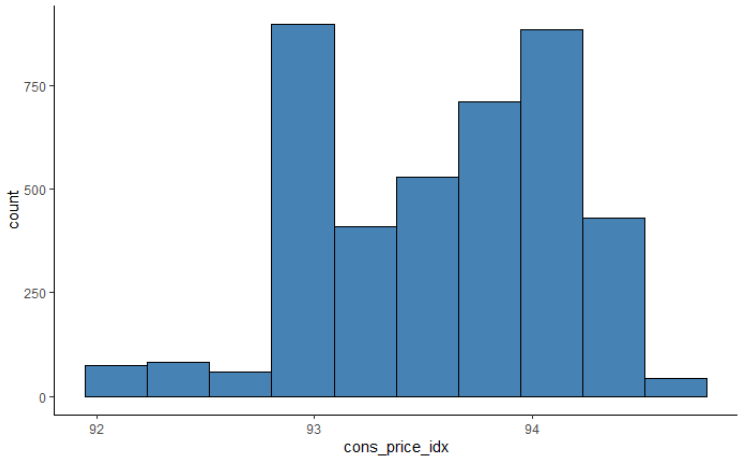


La tasa de variación se mueve entre el intervalo (-2, 2) con pocos valores en el centro.

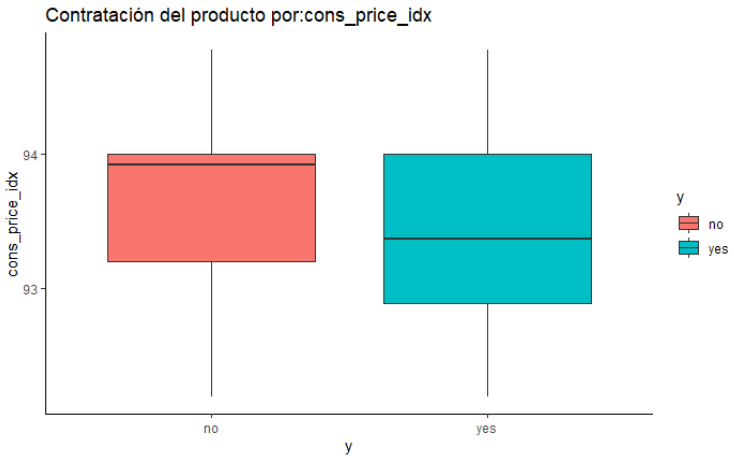


Observamos que la tasa de variación de empleo para la gente que rechaza como para la que contrata es muy similar

2. **cons.price.idx**

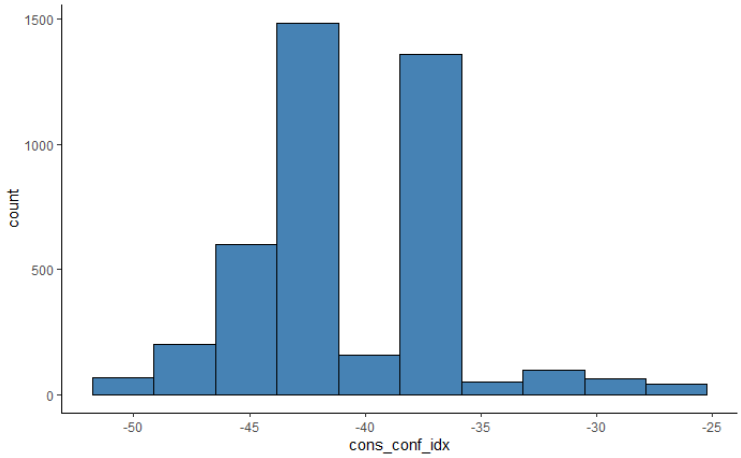


El índice de precios al consumidor normalmente se encuentra sobre 93 y 94.

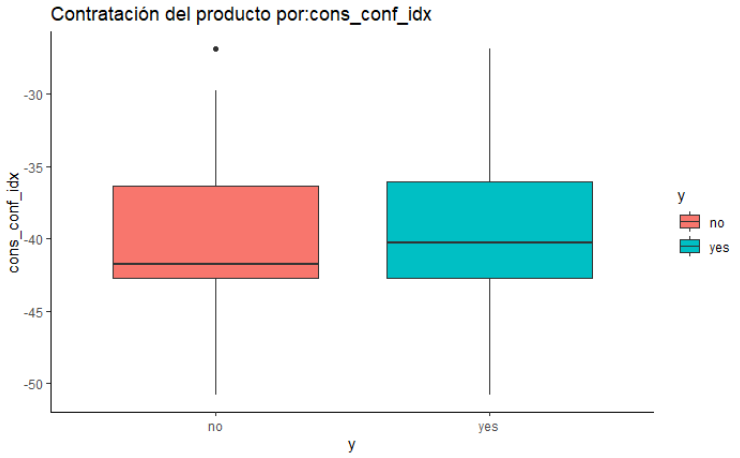


Si el índice de precios al consumidor es bajo es se observa que hay más posibilidades de que un individuo contrate.

3. cons.conf.idx

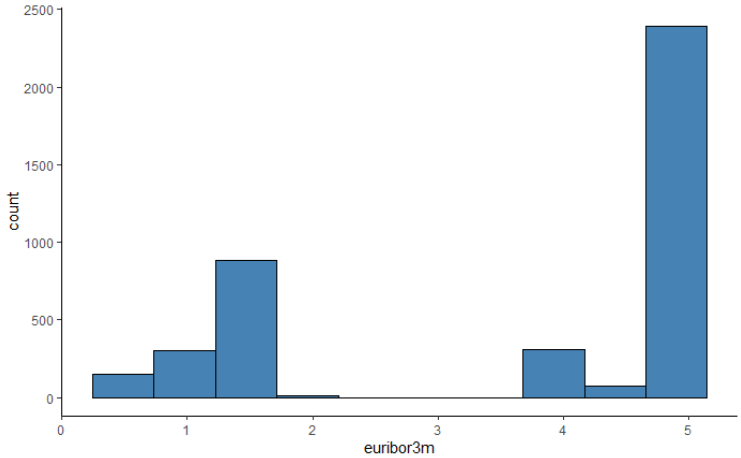


Gran parte del índice de confianza del consumidor se encuentra en un intervalo de (-45, -35).

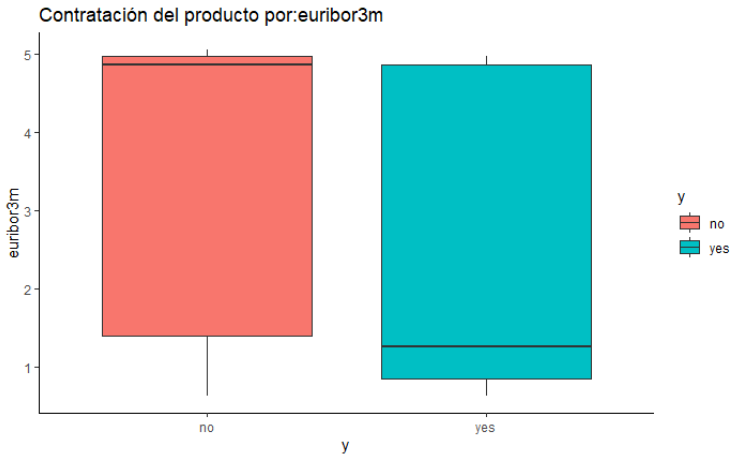


No se observan diferencias significativas respecto a la suscripción al depósito según el índice de confianza del consumidor.

4. **euribor3m**



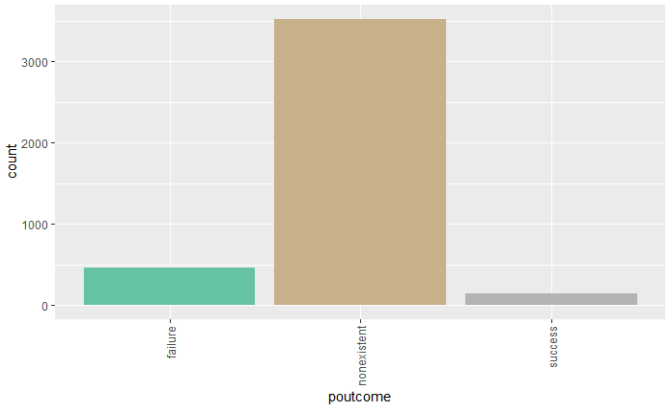
El euribor se sitúa cerca del 1,5 y cerca del 5.



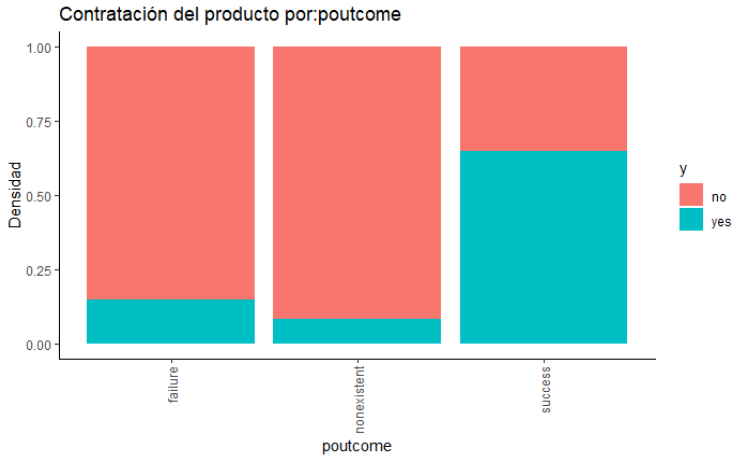
Claramente cuanto más bajo sea el índice euribor los clientes más se suscriben al depósito.

Otros atributos

- **Categóricas:**
  1. **poutcome:** resultado de la anterior campaña de marketing

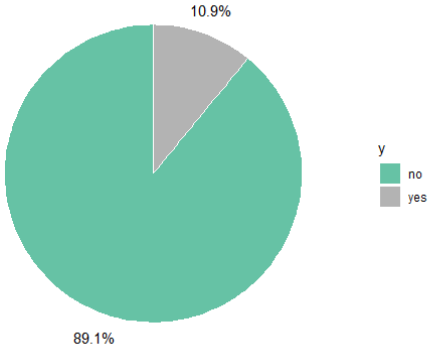


La gran mayoría de clientes no ha participado en las anteriores campañas de marketing. De los que sí han participado, la mayoría no ha contratado



La mayoría de clientes que más se suscriben al depósito ya han participado en campañas de marketing anteriores.

Variable respuesta (Y):



Solo un 10% de los clientes contratan el depósito

Preprocessing

Esta parte del trabajo es una de las más importantes porque para poder realizar buenas predicciones y hacer interpretación de la base que tratamos tiene que estar en el formato correcto y adecuado.

Este preprocesamiento se basa en herramientas que mejoran, clarifican y pulen la base de datos para llevar a cabo los análisis posteriores correctos y nos servirán tanto para los métodos que realizarán tanto el Equipo 1 como el Equipo 2.

Nuestro preprocesamiento tratará los siguientes temas:

- Corrección de etiquetas
- Balanceamiento de datos
- Importación como NA's los "Unknown"
- Transformación de la variable "p.days"
- Eliminación de algunas variables
- Tratamiento de NA's
- Estandarización de variables numéricas
- Tratamiento de outliers
- Tratamiento de la correlación

Corrección de etiquetas

Hemos corregido las etiquetas de algunas variables de la base de datos. Dentro de nuestra base de datos tenemos "characters" (observaciones no numéricas que describen información sobre individuos) que debemos transformar a factores para una correcta codificación. La mayoría de funciones y comandos que usamos requieren que los datos no numéricos sean factores. También existen variables "integer" (datos numéricos enteros como la edad) y variables "numeric" (datos numéricos decimales como el índice de precios del consumidor). Es importante que cada variable esté etiquetada correctamente ya sea "numeric", "integer" o "factor".

Balanceamiento de los datos

Uno de los problemas más importantes que hemos tenido ha sido la diferencia descomunal dentro de la variable respuesta "y" donde un cliente se ha suscrito al depósito a plazos "si" o no lo ha hecho "no". Nos encontramos en que tenemos muchísimos más observaciones "no" que "si" y por ende la probabilidad de ocurrencia de la suscripción a plazo es muy baja.

Y (v.respuesta)	No	Si
Observaciones	3668	451
%	89,05	10,94

Efectivamente existe un problema con el balanceamiento porque el 89% de la variable respuesta son “no” y tan solo un 11% son clientes que se suscriben a plazo.
Esta propiedad desfavorable implica que los modelos que construimos tienen menos oportunidades de reconocer diferencias que si fuera una base de datos balanceada.

¿Cuál es la solución? Existen tres posibles métodos que nos permiten pasar de una muestra no balanceada a una balanceada. Nuestro objetivo es obtener una base de datos que tenga aproximadamente 40% “si” y 60% “no”.

Submuestreo (undersampling)
Reducción aleatoria de aquellas observaciones con variable respuesta negativa reduciendo la muestra total.

Ponderación (weighting)
Asignamos pesos a las observaciones en función de si el cliente se suscribe o no dando mayor fuerza a los clientes con variable respuesta positiva. Se crearía una nueva variable llamada “pesos” y a la hora de estimar modelos se ponderaría la variable en función de dichos pesos.

Sobremuestreo (oversampling)
Este método, que es el que hemos utilizado, se basa en añadir observaciones que tengan la variable respuesta positiva aumentando así el % de casos positivos y a su vez disminuyendo el % de casos negativos.
Al disponer de la base de datos original hemos podido crear una subbase solo con respuestas positivas y a partir de esta base coger una muestra aleatoria que añadimos a nuestra base. Por último hemos eliminado las observaciones duplicadas y el resultado es el siguiente:

Y (v.respuesta)	No	Si
Observaciones	3668	2421
%	60,24	39,76

Cumplimos el objetivo y ahora podemos afirmar que nuestra nueva base de datos tiene 6089 valores y está balanceada.

Importación como NA’s los “unknown”

En general los datos perdidos se suelen codificar como NA’s pero existen casos en que están codificados como “unknown”, “999” o simplemente espacios en blanco y nuestra obligación para no cometer errores es transformar dichas codificaciones a la establecida en general que es NA (Numerical Aperture).
Los valores perdidos de nuestro banco portugués se han codificado todos como “Unknown” menos la variable p-days que tiene sus missings como “999”.

Una vez realizada la transformación podemos proceder al siguiente paso.

Variables	Codificación inicial	Codificación final
p.days	“999”	“NA”
Todas las demás	“Unknown”	“NA”

Transformación de la variable “pdays”

La variable “pdays” nos indica el número de días que han pasado desde que se contactó con el cliente por última vez. Para tratarla creamos una variable binaria adicional llamada “contacted” que sea “Sí” en caso en que se haya podido contactar con el cliente (sin tener en cuenta los días que han pasado) o “No” en caso contrario. Por último sustituimos los NA de la variable “pdays” por valores nulos y así los eliminamos.
De ahora en adelante trabajaremos tanto con la variable numérica de los días como la variable categórica que nos indica si ha habido contacto o no.

Eliminación de variables

En cualquier análisis o modelización de unos datos siempre se debe saber el significado real de las variables con las que se trabaja. En caso contrario el uso de una o varias variables desconocidas en un modelo podrían derivar en un problema de interpretación y por ende se deben eliminar.
En el caso que tratamos existen dos variables cuyo significado no hemos podido esclarecer.

“default”
Esta variable binaria que indica si un cliente tiene o no crédito por defecto. Sí que podemos entender su significado pero la hemos eliminado porque solo tiene una única observación positiva y por lo tanto se podría considerar toda ella no como variable sino como un único valor negativo. La eliminamos y la quitamos de la base de datos.

“n_employees”
Esta variable representa el número de empleados. Esta variable sí que no la entendemos porque no sabemos si la variable se refiere a la cantidad de empleados que asesoran a un cliente o la cantidad de empleados que hay en el banco al entrar ese empleado en él. Si consideramos la primera opción basándonos en los gráficos descriptivos del primer apartado vemos que es imposible que un cliente tenga tantos empleados a su disposición (estaríamos hablando de una media de 5166 empleados).

```
Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
4964    5099    5191    5166    5228    5228
```

Por otra parte si consideramos que es la cantidad de empleados del banco cuando el cliente entra no vemos la relación con la variable respuesta y por lo tanto decidimos que esta variable también la eliminamos.

Tratamiento de NA's

Esta parte del preprocesamiento es la más importante porque la substitución de valores perdidos es fundamental para realizar buenas predicciones.
Recordemos que variables tienen NA's:

Variable	Job	Marital	Education	Housing	Loan
Clase	Factor	Factor	Factor	Factor	Factor
Nº Na's	39	11	167	105	105
Relativo	0.946%	0.267%	4.054%	2.549%	2.549%

Aunque en un principio decidimos tratar los NA con el método KNN, después de considerarlo y hablar con el profesor nos dimos cuenta que el método MICE era la mejor opción porque las variables que contenían missings eran todo categóricas.

Método "KNN" para valores perdidos

Hemos aplicado el método "knn" para tratar las siguientes variables: "job", "marital", "education", "housing" y "load".
Este método se basa en separar la base de datos en dos sub-bases, la que contiene observaciones con NA's y la que no tiene. Estas variables con "missings" serán introducidas en un vector de menos a más NA's que será el que se utilizará en el algoritmo de Knn.
La idea fundamental del Knn es buscar aquellas observaciones que son más parecidas a las observaciones con valores NA y asignar el valor más común. Así estima cada uno de los valores perdidos y finaliza el proceso.

Método "MICE" (Multivariate Imputation by Chained Equations)

Este método se basa en la imputación múltiple por ecuaciones encadenadas donde se reemplazan valores perdidos por valores "plausibles" .Estos valores siguen una distribución de probabilidad basada en la máxima verosimilitud de la cadena de Markov de Monte Carlo. La idea es simple, a partir de una sola base de datos se imputan m datasets que se analizan por separado y luego se mezclan para tener una sola base de datos con unas estimaciones que son medias de las m imputaciones que se han realizado.

A través de la regresión "polytomous", una regresión polinómica, realizamos la imputación siempre teniendo en cuenta que la semilla que fijamos es la 007.
Una vez ejecutado el método eliminamos todos los valores perdidos y podemos seguir con el siguiente paso del preprocesamiento.

Estandarización de variables numéricas

Todos los métodos que hemos visto en clase (KNN, Naive Bayes, ANN…) necesitan que las variables numéricas con las que trabajan estén escrupulosamente transformadas de manera que sigan las propiedades de estandarización y/o normalización.

La estandarización es un método de centralización que consiste en restar para cada observación la media de la variable y dividirla por la desviación típica.

$$z_i = \frac{x_i - \bar{X}}{S_X}$$

De esa manera conseguimos que las variables estén todas en una misma escala y facilite los métodos de minería de datos. Con la función scale() de R se puede estandarizar la variable numérica que se desee.

Detección de outliers

Los "outliers" són valores extraños dentro de una base de datos que difieren significativamente respecto a otras observaciones. La existencia de dichos valores se puede deber a diversas causas; Errores de tipificación, errores de introducción de datos o simplemente casos extremadamente raros. Sea como fuere estos outliers se deben estudiar en detalle y si fuera necesario, eliminarlos.

Los outliers normalmente se observan en las variables numéricas y por eso el proceso de detección de outliers empieza con una subbase de solo variables numéricas. Para estudiar los outliers nos ayudaremos de la distancia de Mahalanobis.

$$d_m(\vec{x}, \vec{y}) = \sqrt{(\vec{x} - \vec{y})^T \Sigma^{-1} (\vec{x} - \vec{y})}$$

La distancia de Mahalanobis es una medida de distancia que determina la similitud de dos variables aleatorias multidimensionales que tiene en cuenta la correlación de dichas variables.
Así pues, creamos una variable con la distancia de Mahalanobis y la añadimos a la subbase. Para saber qué observaciones son significativamente extrañas realizamos un test de hipótesis evaluando la distancia de Mahalanobis con una chi-cuadrado con 8 grados de libertad porque tenemos 9 parámetros que son las variables numéricas y por lo tanto los grados de libertad son p-1 = 9-1=8.
Consideramos que los "outliers" serán aquellas observaciones cuyo pvalor sea inferior a 0.001.

Test de Mahanalobis

χ_8^2

si pvalor <0.001 consideramos "outlier"
si pvalor >0.001 consideramos observación normal

Al realizar dicho test nos encontramos con 245 observaciones que se consideran outliers y por lo tanto el paso final es eliminarlas.

Ahora la base de datos no tendrá ninguna observación que pueda influenciar los modelos de manera negativa ni descentrar los resultados. De esta manera nos quedamos con 5844 observaciones.

Estudio de la correlación entre variables predictoras

La correlación es el grado de asociación entre dos variables y tiene casos en que es una buena propiedad, la fuerte correlación entre la variable respuesta y una variable predictora indica que para explicar el modelo dicha variable es muy necesaria y esclarecedora, y casos

en que puede suponer un problema. Una gran correlación entre dos variables explicativas puede provocar errores de confusión y estimaciones erróneas de parámetros.

La correlación sólo se puede dar en variables numéricas y aceptaremos que existe “un problema significativo de correlación” si el valor absoluto de la correlación de dos variables supera el *0.7*.

Por lo tanto creamos una subbase de solo numéricas y calculamos la matriz de correlaciones entre todas ellas. Hay tres variables que tienen valores de correlación notablemente altos, son las siguientes.

MATRIZ DE CORRELACIONES

	emp.var.rate	cons.price.idx	euribor3m
emp.var.rate	1.0000000	0.7407112	0.9637242
cons.price.idx	0.7407112	1.0000000	0.6153711
euribor3m	0.9637242	0.6153711	1.0000000

Obviamos la diagonal y nos fijamos en el 0.9637 que indica una gran correlación entre el euribor trimestral y el ratio de variación del empleado (“emp.var.rate”). También observamos bastante correlación (0.74) entre el ratio de variación de empleo y el índice del precio del consumidor (“cons.price.idx”) . Por tanto tenemos que decidir cuál eliminar.

Como las dos asociaciones problemáticas han sido relacionadas con una variable común, “emp.var.rate”, decidimos quitar dicha variable y realizar otra vez la matriz añadiendo otras variables para ver si el cambio es positivo y si se ha eliminado el problema de correlación.

MATRIZ DE CORRELACIONES

	age	cons.price.idx	euribor3m	duration
age	1.000000000	-0.02118686	-0.055128965	-0.006031435
cons.price.idx	-0.021186860	1.00000000	0.615371101	0.028916882
euribor3m	-0.055128965	0.61537110	1.000000000	0.002099853
duration	-0.006031435	0.02891688	0.002099853	1.000000000

Una vez realizada la eliminación no vemos que ningúna variable más presente problemas y por tanto cerramos el apartado del estudio de correlación.

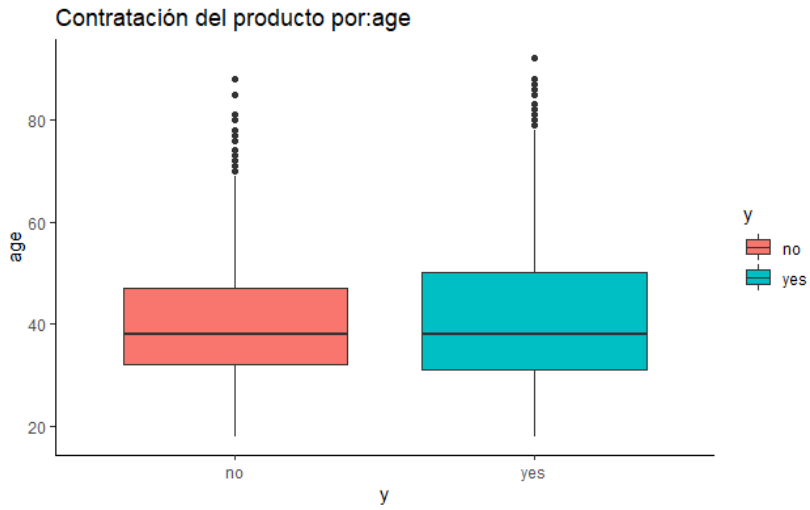
Con este último estudio concluimos el preprocesamiento y los datos ya están listos para poder ser utilizados con los métodos correspondientes a la minería de datos. Pero antes haremos un breve análisis descriptivo de la base ya preprocesada.

Análisis descriptivo de los datos preprocesados

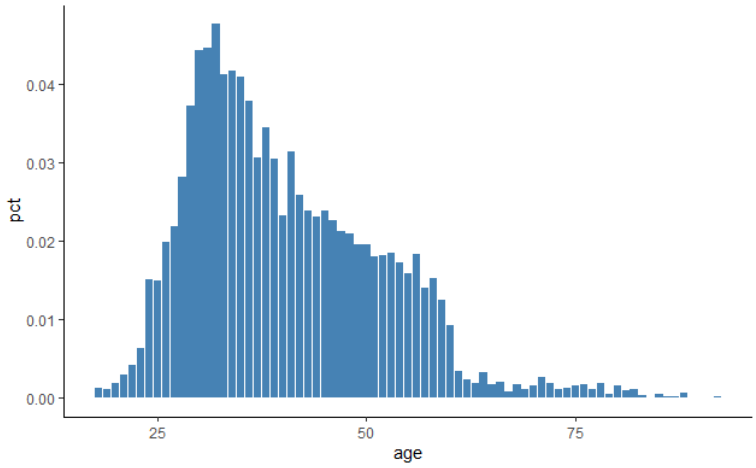
Para todas las variables hemos realizado un análisis bivariante entre dicha variable en función de la variable respuesta “y” y un análisis univariante.

Características del cliente

- **Númericas:**
  1. **age**

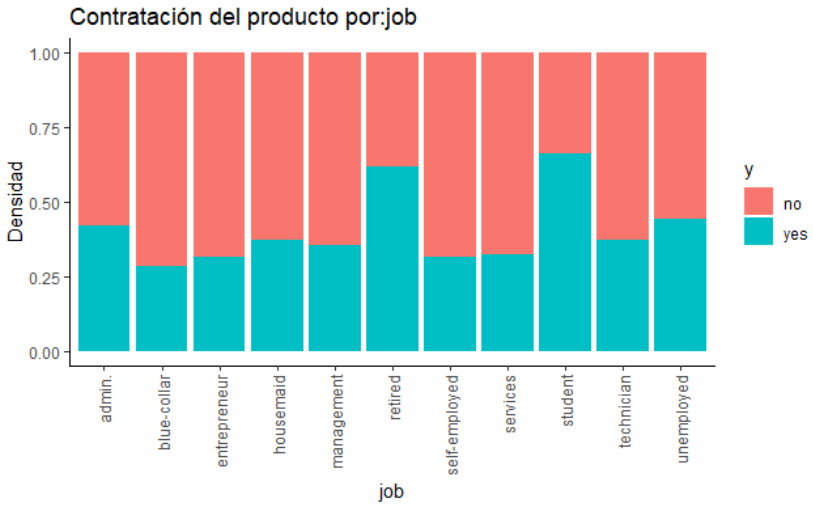


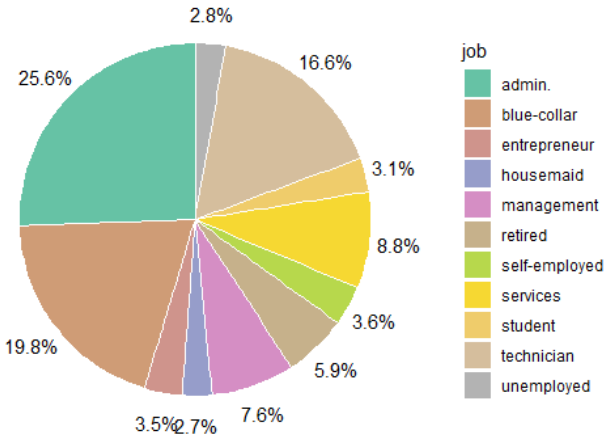
No se observan muchas diferencias en la edad de los clientes según si se suscriben al depósito.



La mayoría de los clientes tiene entre 35 y 45 años. Por razones legales nadie tiene menos de 18 años, también observamos una bajada notable a partir de los 60 años

- **Categóricas:**
  1. **job**

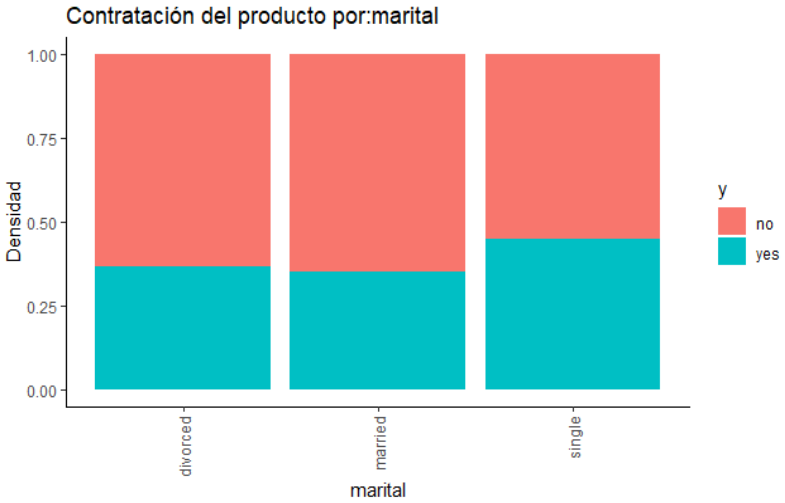




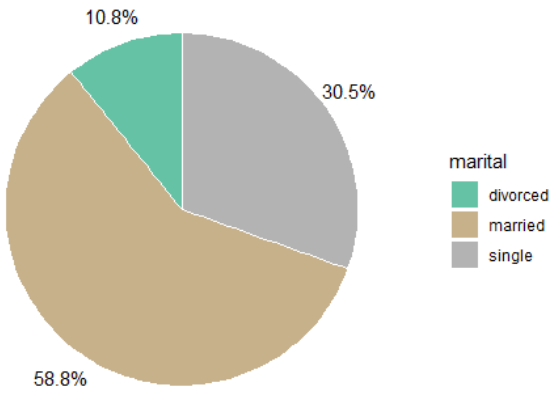
Vemos que los clientes que más se suscriben al depósito son retirados o estudiantes. Entre los demás trabajos no se ven diferencias significativas.

Podemos observar que la mayoría de clientes trabajan en el sector de la administración o son trabajadores de cuello azul, que son individuos que forman la parte más baja de la jerarquía de las empresas.

  2. **marital**

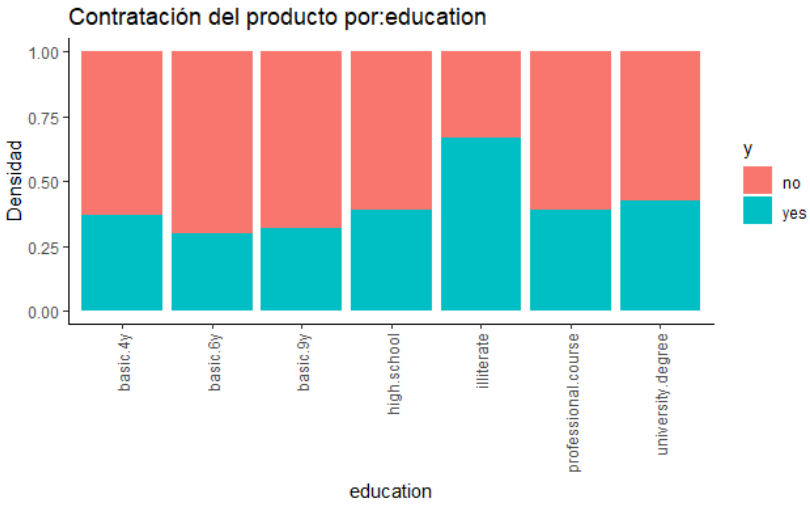


Los clientes solteros tienen mayor tendencia a suscribirse al depósito, aunque no hay grandes diferencias.



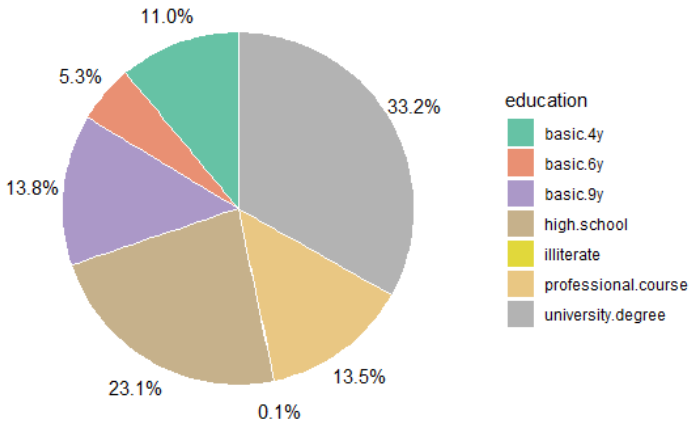
Más de la mitad de los clientes están casados

3. **education**

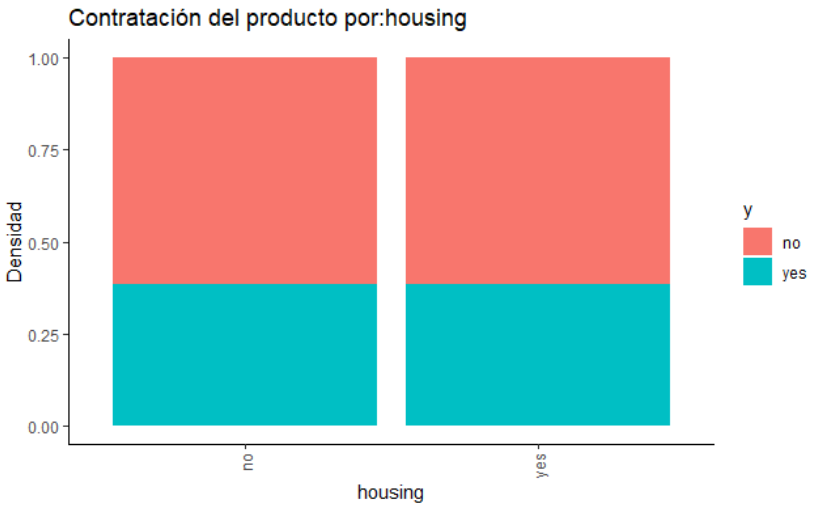


Se puede ver en el gráfico bivariante que los clientes con menos estudios son los que más se suscriben al depósito.

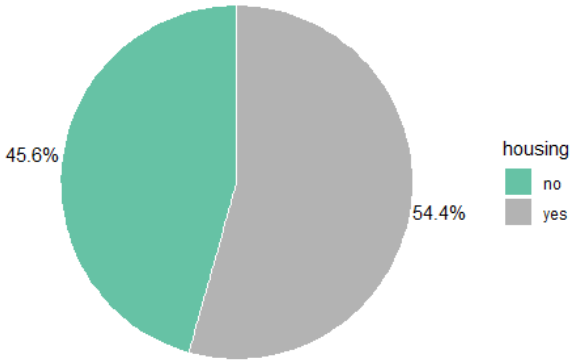
En el gráfico circular podemos observar que una gran parte de los clientes tienen estudios universitarios.



4. **housing**

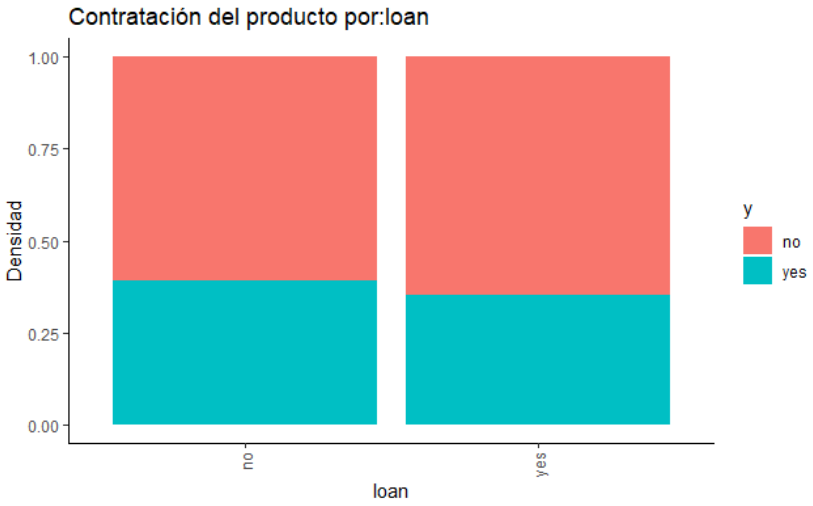


No se aprecia ninguna diferencia entres si el cliente tiene préstamos hipotecarios o no a la hora de suscribirse al depósito.

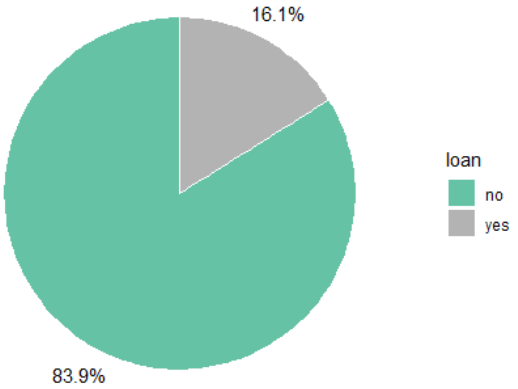


La cantidad de clientes que tiene algún préstamo hipotecario es similar a la que no tiene ningún préstamo.

5. **loan**



Por lo que respecta a si el cliente tiene un préstamo personal, como podemos ver en el gráfico bivariante no se aprecian diferencias significativas entre si el cliente tiene un préstamo o no.

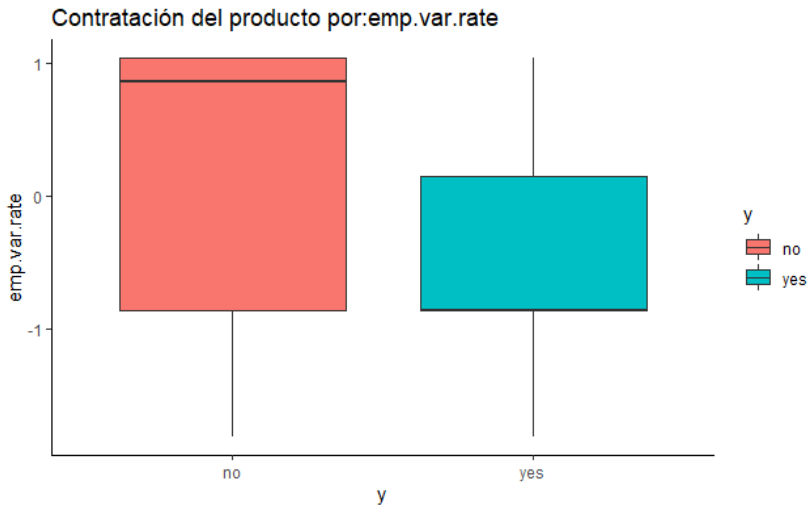


Gran parte de los clientes tiene un préstamo personal.

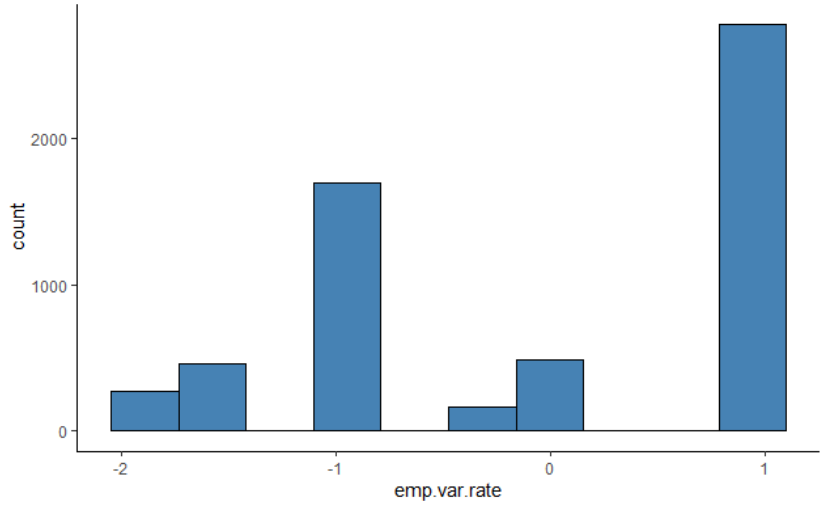
Contacto con el cliente en las campañas de marketing

Contexto socioeconómico

- **Númericas:**
  1. **emp.var.rate**

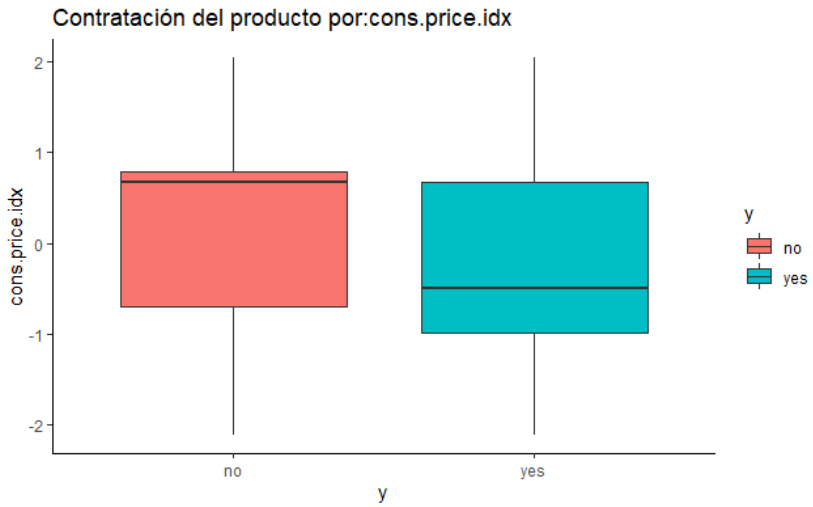


Como se puede observar en el gráfico los clientes que tienen una tasa de variación de empleo más alta rechazan más subscrivires al depósito.

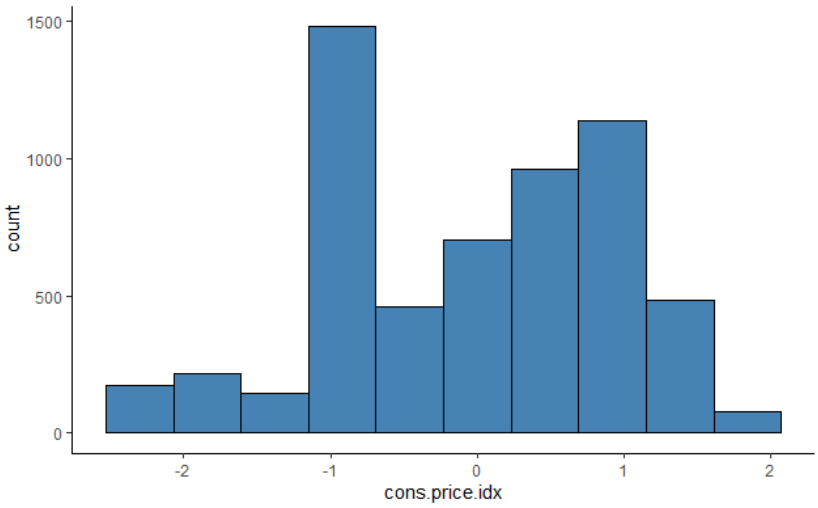


Gran parte de los clientes tienen una tasa de variación de empleo de 1 o de -1.

2. **cons.price.idx**

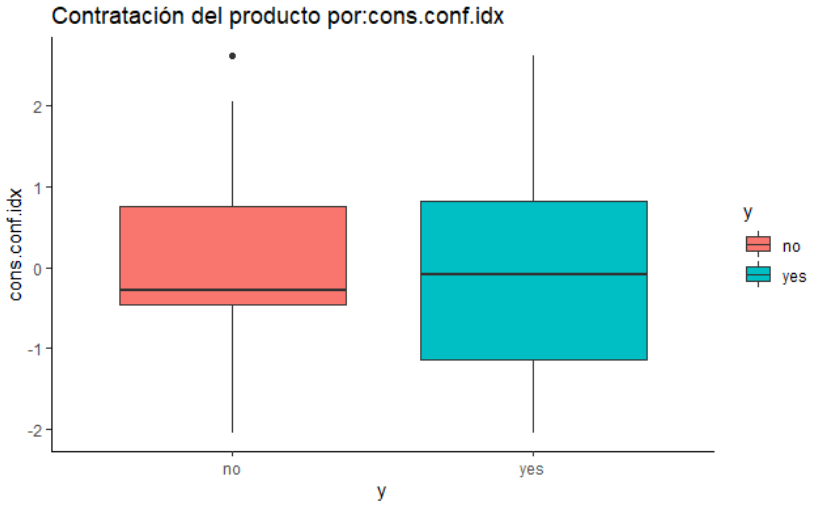


Cuando el índice de precios al consumidor es superior se observa una ligera tendencia a rechazar la suscripción a plazo.

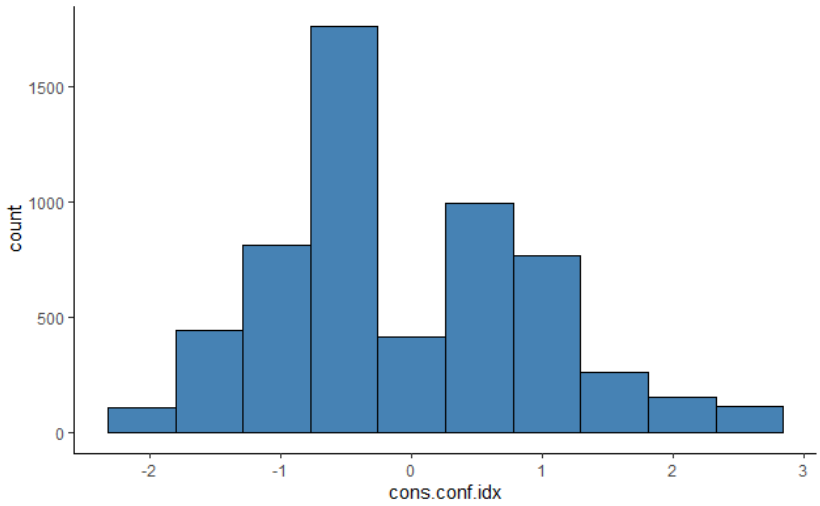


En el gráfico se observa que el índice de precios al consumidor normalmente se encuentra sobre -1, por debajo de -1 baja drásticamente, al igual que por encima de 1.5.

3. **cons.conf.idx**

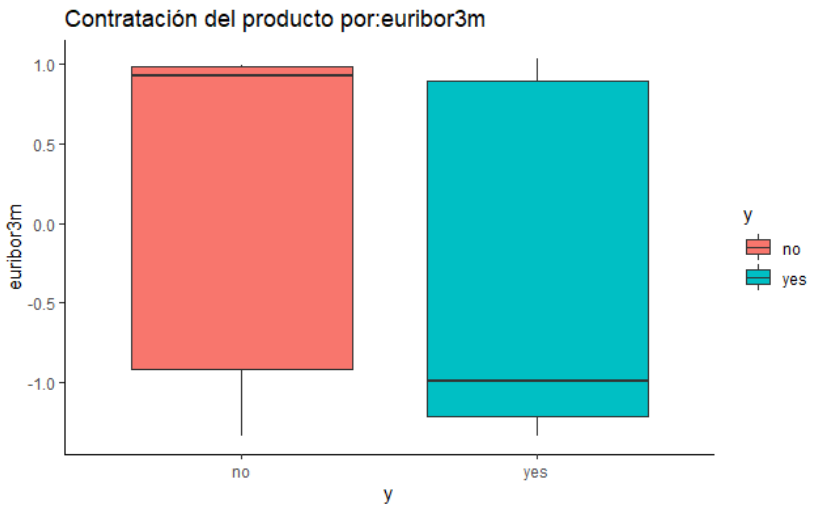


No se observan diferencias significativas respecto a la suscripción al depósito según el índice de confianza del consumidor.

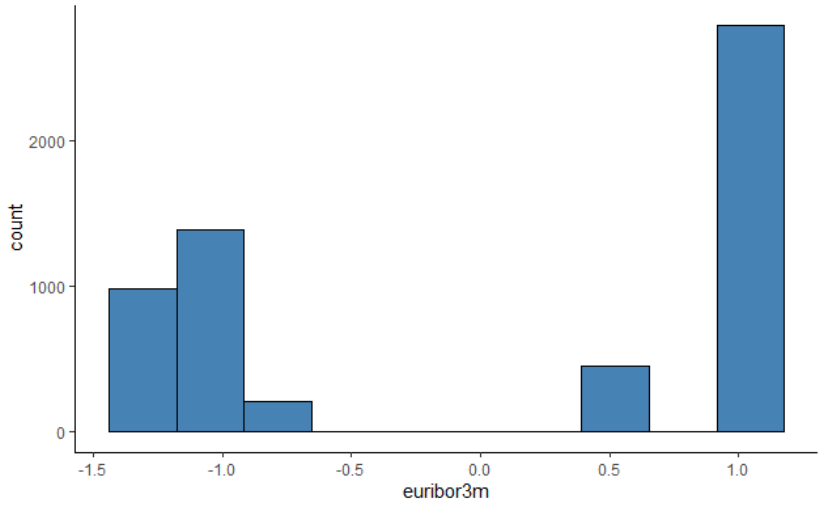


En el gráfico se puede ver que gran parte del índice de confianza del consumidor se encuentra ligeramente por debajo de 0.

euribor3m



Los clientes se suscriben más al depósito cuando el índice del euribor es bajo como se observa en el gráfico bivariante.

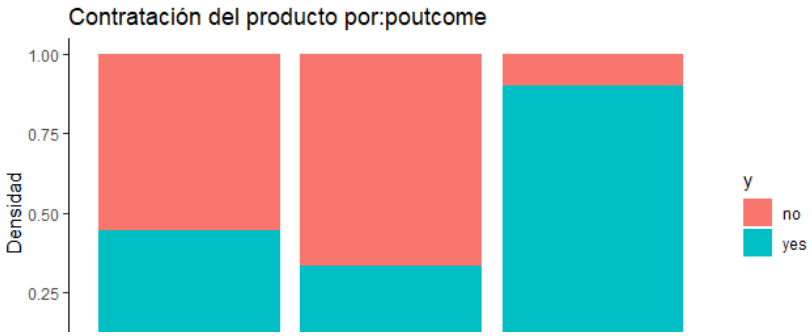


En el gráfico anterior vemos que normalmente el euribor se sitúa alrededor de 1 y de -1, pero no hay valores cercanos a 0.

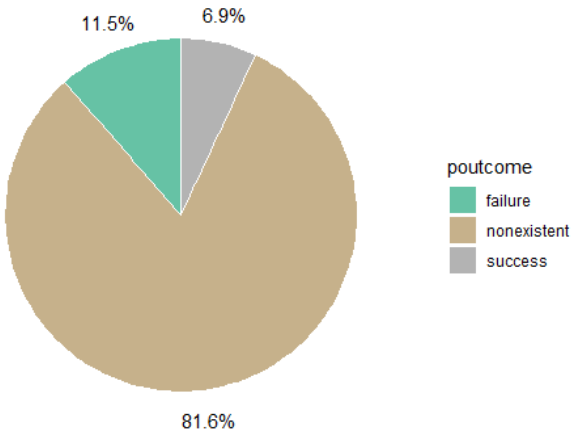
Otros atributos

- **Categóricas:**

1. **poutcome**

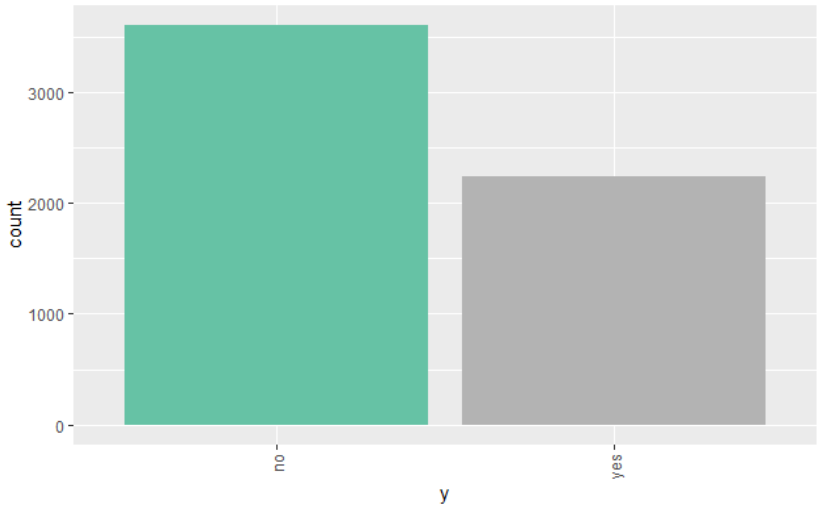


Los clientes que más se suscriben al depósito son los clientes que ya han contratado algún producto anterior en el banco en alguna campaña anterior.



A la mayoría de clientes no se les ha contactado en campañas anteriores, a los clientes con los que se contactó la campaña fracasó un 12% y tuvo éxito un 7%.

Variable respuesta (Y)



La mayoría de los clientes no contratan el depósito.

Diseño de los dos procesos de minería de datos

Una vez que los datos han sido preprocesados y las variables significativas seleccionadas, el siguiente paso es emplear un algoritmo de machine learning que permita crear un modelo capaz de representar los patrones presentes en los datos de entrenamiento y generalizarlos a nuevas observaciones. Existen multitud de algoritmos, cada uno con unas características propias y con distintos parámetros que deben ser ajustados. Por lo general, las etapas seguidas para obtener un buen modelo son:

Ajuste/entrenamiento: consiste en aplicar un algoritmo de machine learning a los datos de entrenamiento para que el modelo aprenda.

Evaluación/validación: el objetivo de un modelo predictivo no es ser capaz de predecir observaciones que ya se conocen, sino nuevas observaciones que el modelo no ha visto. Para poder estimar el error que comete un modelo es necesario recurrir a estrategias de validación. Entre todas las técnicas, se ha utilizado el método de K-Fold-Cross-Validation(CV).

Optimización de hiperparámetros: muchos algoritmos de machine learning contienen en sus ecuaciones uno o varios parámetros que no se aprenden con los datos, a estos se les conoce como hiperparámetros. No existe forma de conocer de antemano cuál es el valor exacto de un hiperparámetro que da lugar al mejor modelo, por lo que se tiene que recurrir a estrategias de validación ejecutadas en bucle para comparar distintos valores.

Predicción: una vez creado el modelo, este se emplea para predecir nuevas observaciones

Metodologia

Se han realizado dos tipos de algoritmos; Los supervisados, en el que el objetivo es predecir una variable respuesta, y los no supervisados en el que el objetivo es puramente descriptivo, buscar relaciones entre variables. Se ha dividido el trabajo en 2 equipos, cada uno ha aplicado al menos un método supervisado, y 3 métodos no supervisados.

No supervisados: Análisis de Componentes Principales(ACP) y Análisis de Correspondencia Múltiple(ACM)
Supervisados: Árbol de decisión, Naive Bayes, K vecinos más proximos(KNN), Support Vector Machine(SVM), Extreme Gradient Boosting(XGBoost) y Random forest.

Evaluar la capacidad predictiva de un modelo consiste en comprobar cómo de próximas son sus predicciones a los verdaderos valores de la variable respuesta. Para poder cuantificar de forma correcta este error, se necesita disponer de un conjunto de observaciones, de las que se conozca la variable respuesta, pero que el modelo no haya “visto”, es decir, que no hayan participado en su ajuste. Con esta finalidad, se dividen los datos disponibles en un conjunto de entrenamiento y un conjunto de test. El tamaño adecuado de las particiones depende en gran medida de la cantidad de datos disponibles y la seguridad que se necesite en la estimación del error. En el presenta trabajo, se ha decidido hacer una partición de 80% para datos de entrenamiento y 20% para datos de test. De esta forma la base de datos está estructurada en “train” y “test”.

Es muy importante que la partición sea de forma aleatoria, y también que se haga de forma estratificada respecto a la variable respuesta, es decir, que la distribución de las categorías de la variable sea la misma en los datos de “train” y los datos de “test”. La función que se ha utilizado en nuestro código (createDataPartition) garantiza este supuesto.

Para que un modelo predictivo sea útil, debe de tener un porcentaje de acierto superior a lo esperado por azar o a un determinado nivel basal. En problemas de clasificación, el nivel basal es el que se obtiene si se asignan todas las observaciones a la clase mayoritaria de la variable respuesta. En nuestro caso se tiene un 61,5% de valores “no” en la variable respuesta, por lo tanto, este es el porcentaje mínimo que hay que se debe superar con los modelos predictivos.

Equipo 1

El equipo 1, que recordamos lo forman Andreu, Félix, Joan y Marcel se encargará de realizar los siguientes métodos: ACP, árbol de decisión, Naive Bayes y KNN.

ACP:

El ACP o PCA en inglés (principal component analysis) es un método no supervisado que reduce el número de dimensiones con el objetivo de conservar la variabilidad de las variables numéricas. Por tanto en este apartado nos centraremos en las variables “age”, “cons.conf.idx”, ”euribor3m”, ”previous”, “campaign”, “duration”, “pdays” y “cons.price.idx**”**.

El primer paso es crear una sub-base con las variables numéricas con la que trabajaremos. Aplicamos el método factorial de ACP a través de la función prcomp() y visualizamos el resultado que nos muestra la transformación siguiente:

```
Standard deviations (1, .., p=8):
[1] 1.4819425 1.1241790 1.0522544 1.0020724 0.9620158 0.9418473 0.5904466 0.5172057

Rotation (n x k) = (8 x 8):
                    PC1        PC2         PC3         PC4         PC5         PC6          PC7         PC8
age             0.07988050  0.2063970  0.52789481 -0.45270795  0.36803722 -0.57321243  0.029000147 -0.05036959
duration       -0.02391421 -0.1024046 -0.35855270 -0.82282349  0.17330487  0.39039805 -0.027342488 -0.01224870
campaign       -0.20450890  0.2204268 -0.05386759  0.31889182  0.84812228  0.29190099 -0.003033542  0.02162416
pdays           0.45592569  0.4903717 -0.24914473  0.01379143 -0.03972982  0.01911123  0.688835820 -0.11386630
previous        0.52707437  0.3109093 -0.25282558  0.05456055  0.04043390 -0.07809742 -0.699332422 -0.24865729
cons.price.idx -0.39484173  0.5074656 -0.35413518 -0.06883936 -0.13887488 -0.29075207 -0.130826021  0.57921467
cons.conf.idx   0.06804607  0.4301908  0.58168195 -0.07990205 -0.23603023  0.58534762 -0.132289000  0.22299083
euribor3m      -0.55228384  0.3443996 -0.01028718 -0.04510836 -0.19233501  0.01332812 -0.015024193 -0.73269007
```

Las 8 componentes principales con sus respectivas desviaciones estándar y los valores que asociamos a las variables numéricas en cada componente principal.

Tría del número de componentes principales

Aunque hayamos encontrado las 8 componentes, no es necesario conservarlas todas puesto que solo necesitaremos seleccionar aquellos que conformen un 80% de variabilidad total de los datos.

Esta selección se puede realizar considerando los valores de su inercia, los valores propios y su variabilidad acumulada.

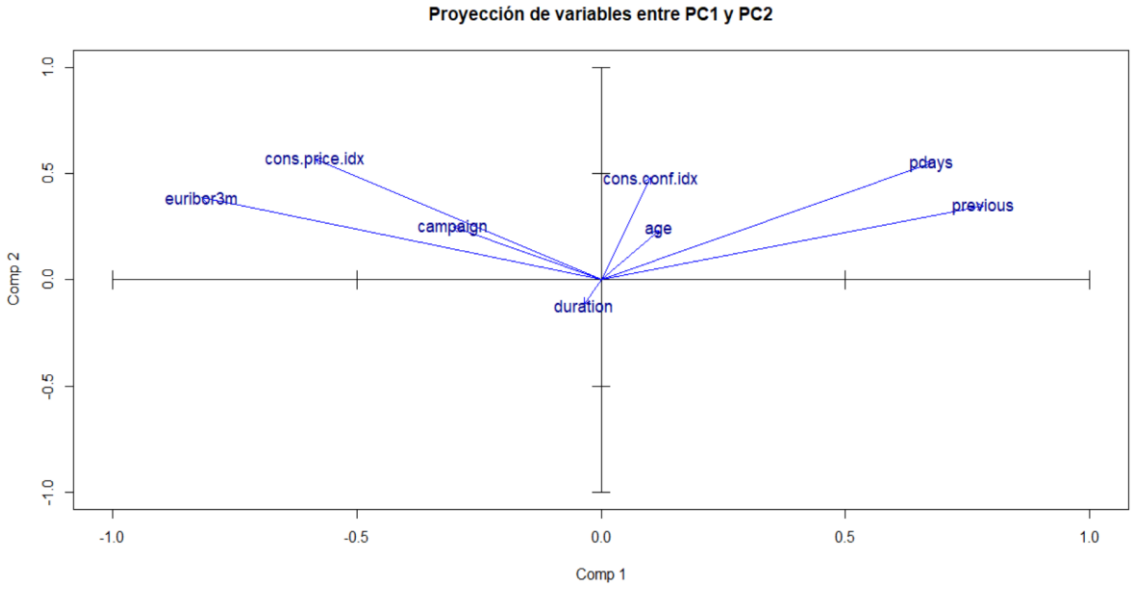
	PCA 1	PCA 2	PCA 3	PCA 4	PCA 5	PCA 6	PCA 7	PCA 8
val.propios	2,19	1,26	1,11	1,00	0,92	0,89	0,34	0,27
variabilidad	27,45	15,78	13,84	12,55	11,56	11,08	4,36	3,34
% acum	27,45	43,25	57,09	69,64	81,21	92,30	96,65	100

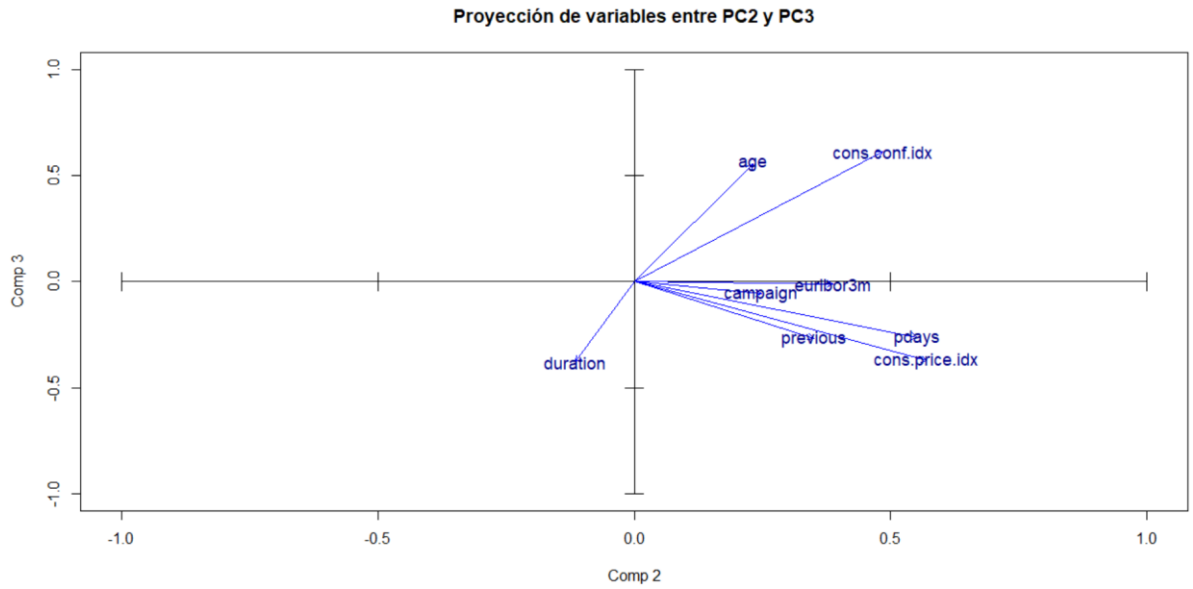
Por lo tanto si elegimos las primeras 5 componentes principales ya tenemos el 81,21% de variabilidad de los datos numéricos.

Proyección de las variables numéricas

Con los 5 ejes seleccionados podemos hacer diversos planos y combinaciones. Esto supone un problema porque si quisiéramos estudiar todas ellas tendríamos que realizar demasiados gráficos.
Con el fin de sintetizar y visualizar los resultados solo escogeremos estudiar las componentes 1, 2 y 3 y graficar los planos primera componente con segunda y tercera componente con tercera.
Consideramos que elegir las 3 componentes con más inercia es la opción a seguir más indicada.

En primer lugar realizaremos gráficos de los dos planos considerando las correlaciones de las variables numéricas:





Interpretación de las proyecciones:

- **Age:** La variable edad está más relacionada con la tercera componente que con las componentes 1 y 2, aunque sus proyecciones en ambos gráficos no són paralelas a las componentes y no se acercan al valor absoluto 1.

  De esta manera sabemos que la edad está relacionada positivamente, aunque no con mucha fuerza (sobre todo con PC3), con las tres componentes. Dicho hecho significa que a medida que la edad crece las componentes también crecen, aunque en poca medida.

- **Duration:** La duración de la última comunicación tiene un comportamiento similar a la edad, puesto que está muy poco relacionada con las dos primeras componentes y relacionada inversamente con la tercera (ya que se sitúa en el tercer cuadrante y el eje de las Y es la PC3).

- **Previous:** La variable, que indica cuántas veces se ha contactado con el cliente en campañas anteriores, está muy relacionada con la componente primera positivamente y poco con las otras dos. Eso supone que a medida en que un cliente es contactado más veces, la componente 1 crece.

- **Euribor3m:** La tasa euribor a tres meses está muy relacionada negativamente con la PC1, poco relacionada positivamente con PC2 y nada relacionada con PC3 (el segundo gráfico muestra la proyección paralela en el eje de las X que equivale a la componente 2). Eso significa que el incremento de la tasa euribor supone un decrecimiento significativo en la componente 1 y un breve crecimiento en la PC2.

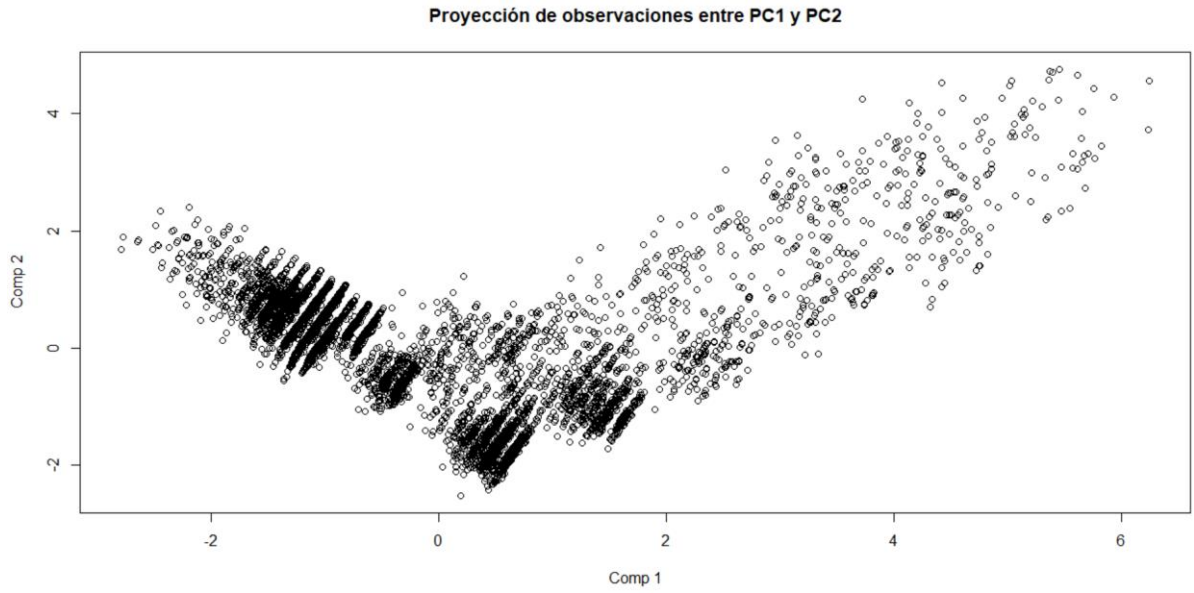
- **Campaign:** Esta variable indica las veces que un cliente ha sido contactado en esta campaña. De todas las variables, es la que está menos relacionada con cualquiera de las componentes principales.
- **Pdays y Cons.Price.idx:** Describen la cantidad de días que han pasado desde el último contacto con el cliente y el índice de precios del consumidor respectivamente.

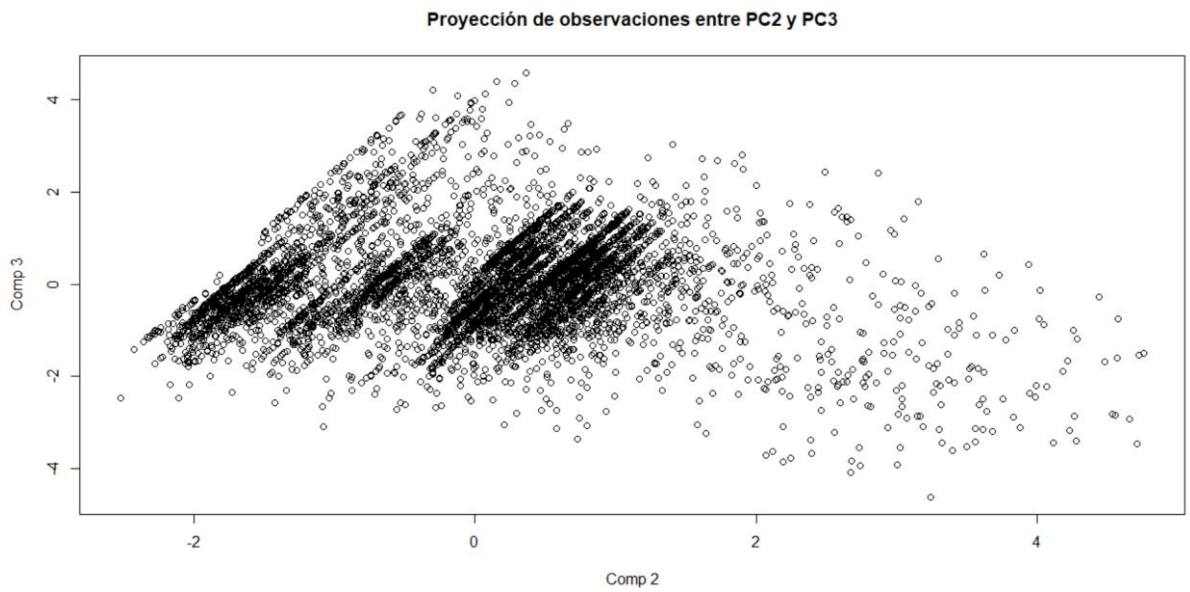
Ambas presentan relaciones similares, pero describen proyecciones inversas. Están poco relacionadas con PC3 y algo relacionadas con PC2. Lo interesante de ellas es que están muy relacionadas con PC1, “pdays” positivamente y ”Cons.Price.idx” negativamente. Por este motivo, la primera hace crecer la componente al aumentar el día de no contacto y la otra la hace decrecer al aumentar el IPC.

- **Cons.Conf.indx:** Nuestra última variable es el índice de confianza del consumidor y es la que más relacionada está, positivamente, con PC3 respecto las otras. Su grado de relación con PC1 es casi nulo y está poco relacionada con PC2. Una crecida del índice, hace aumentar la componente.

Una vez proyectadas las variables, realizamos una proyección de los individuos:

Proyección de individuos





El gráfico que muestra las componentes 1 y 2 indica un comportamiento parabólico convexo de los datos, con el mínimo cerca del valor nulo y donde la mayoría de las observaciones pertenecen a la región central [-2,2] de ambas componentes.

Paralelamente, también podemos considerar un comportamiento lineal de la siguiente forma:

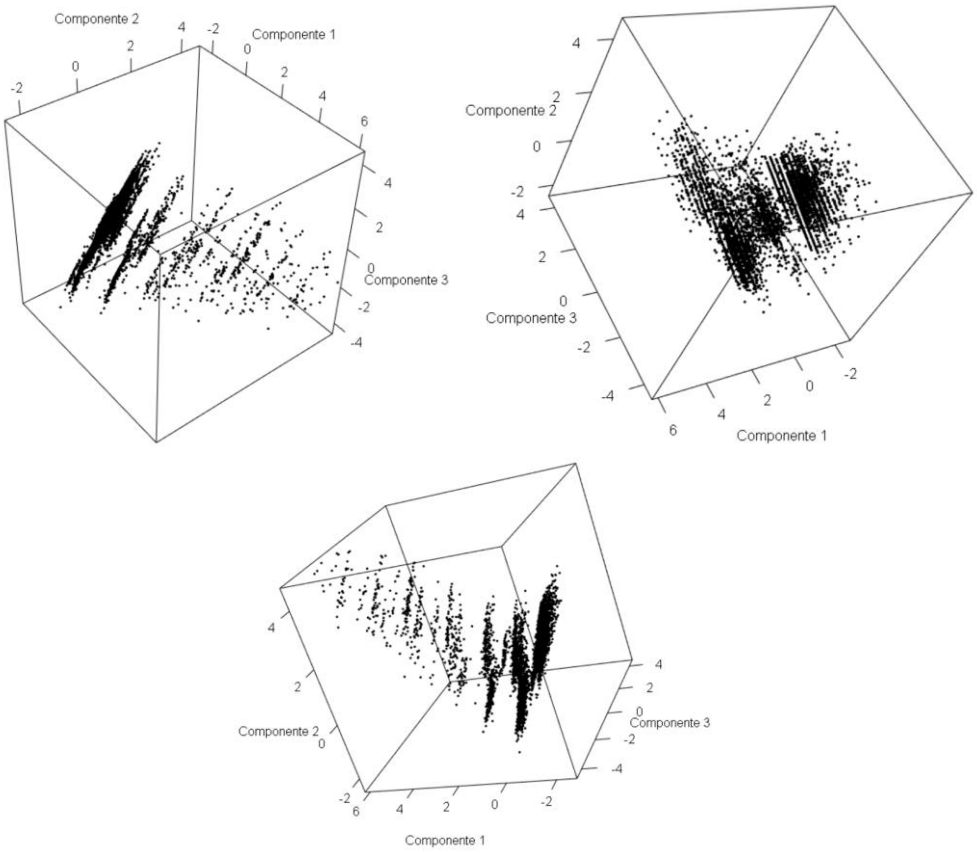
$$PC2 = |PC1| - 2$$

Ya que parece una forma lineal que se le ha aplicado el valor absoluto y que es lineal con pendiente 1. Las observaciones más alejadas comprenden valores de hasta 5 y 6 en la primera componente.

Por otra parte, la proyección de observaciones de las componentes 2 y 3 no resulta de fácil interpretación, pues la nube de valores forma una masa unida, sin tendencia ni linealidad clara. Sí podemos comentar que la mayoría de observaciones rodean el origen de coordenadas (0,0) y se esparcen en la región central [-2,2] tanto en PC2 como PC3.

Gráfico 3D

R también permite crear un gráfico que muestre las 3 componentes juntas, aunque su interpretación es muy difícil de realizar.

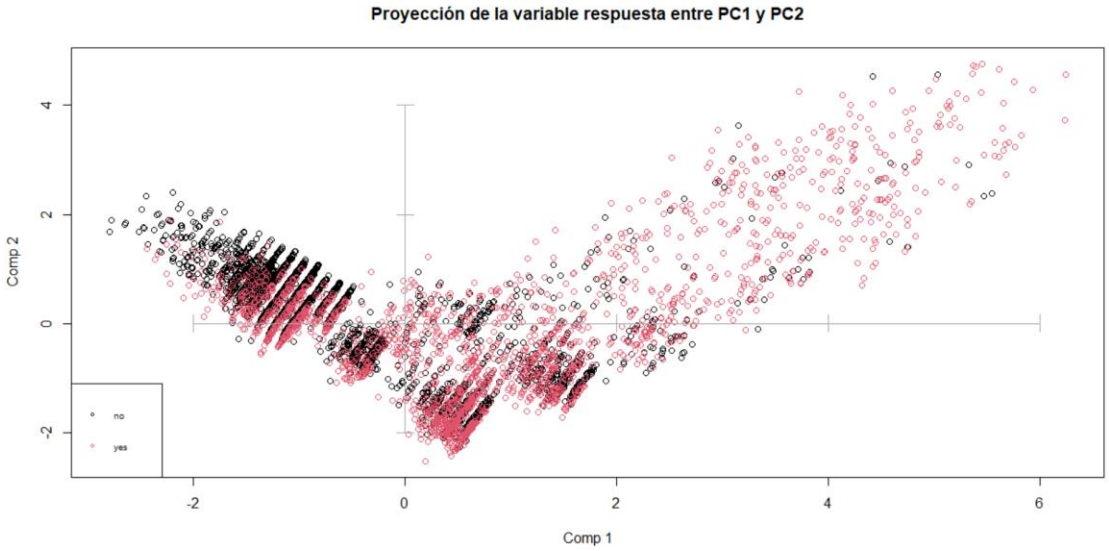


Dependiendo del punto de vista, las observaciones adoptan una u otra forma, impidiendo concretar en una única que se pueda explicar a simple vista. Se tendría que estudiar a fondo o simplemente hacer todas las combinaciones en 2D y sacar conclusiones.

Proyección de la variable respuesta

Usaremos solo la proyección de los componentes PC1 y PC2 porque la otra proyección resulta difícil de interpretar y no aporta demasiada información.

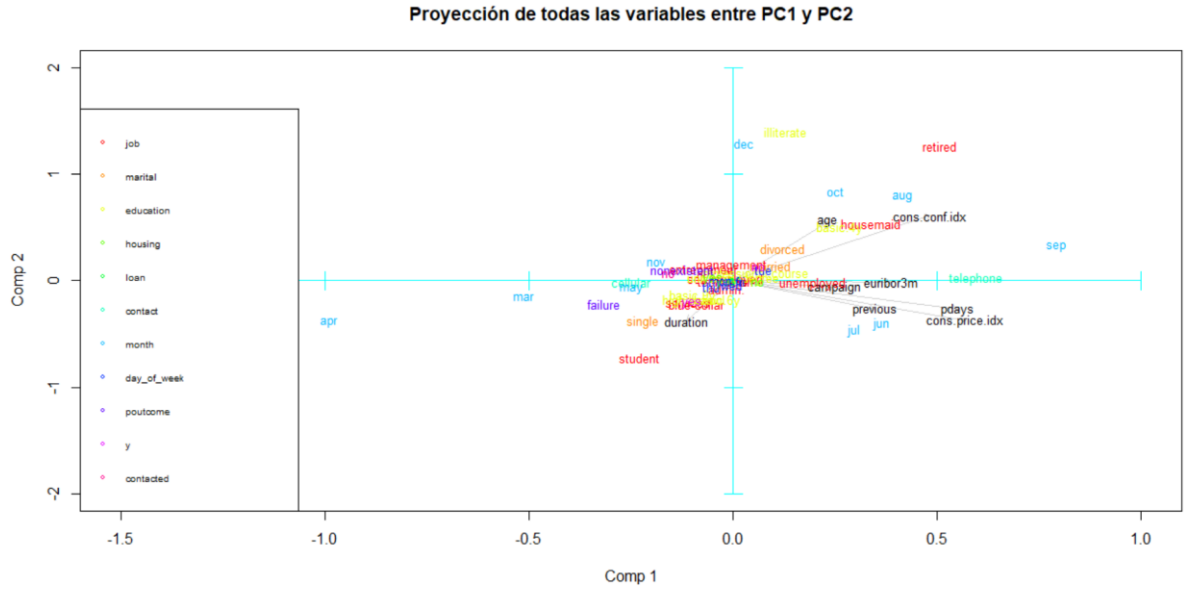
Crearemos un gráfico de los individuos, mostrando de color rojizo aquellos que se hayan suscrito a un depósito a plazo y de negro en caso contrário.



No se observa una clara diferenciación entre los dos niveles de la variable, aunque sí se puede ver como las observaciones positivas comprenden valores más bajos que las negativas, principalmente en la región central.
Aunque no podamos ver una clara diferencia entre las dos opciones, más adelante en esta memoria veremos métodos predictivos que pueden hacerlo y los estudiaremos a fondo.

Proyección de todas las variables

Finalmente, el último gráfico que realizaremos en el apartado de ACP será un gráfico general de todas las variables, tanto las numéricas (en color negro), como las categóricas (en distintos colores).



Su entera interpretación no es posible debido a la confusión de etiquetas y multitud de categorías en la región central, aunque sí podemos comentar que los meses de la variable "month" y los tipos de trabajo de la variable "job" son las categóricas más relacionadas con las componentes principales.

Para concluir el apartado del análisis de componentes principales, es necesario añadir que una forma alternativa de continuar el trabajo sería utilizando las componentes principales a modo de variables numéricas. Si hiciéramos eso, al ser dimensiones ortogonales entre sí, tendríamos buenas propiedades para realizar modelos predictivos.
Consecuentemente el "accuracy" de cada uno de ellos podría ser incluso superior que si utilizáramos las variables de la forma original.
Aun así, hemos decidido realizar los siguientes modelos con las variables estándar y el primer modelo que veremos es el árbol de decisión.

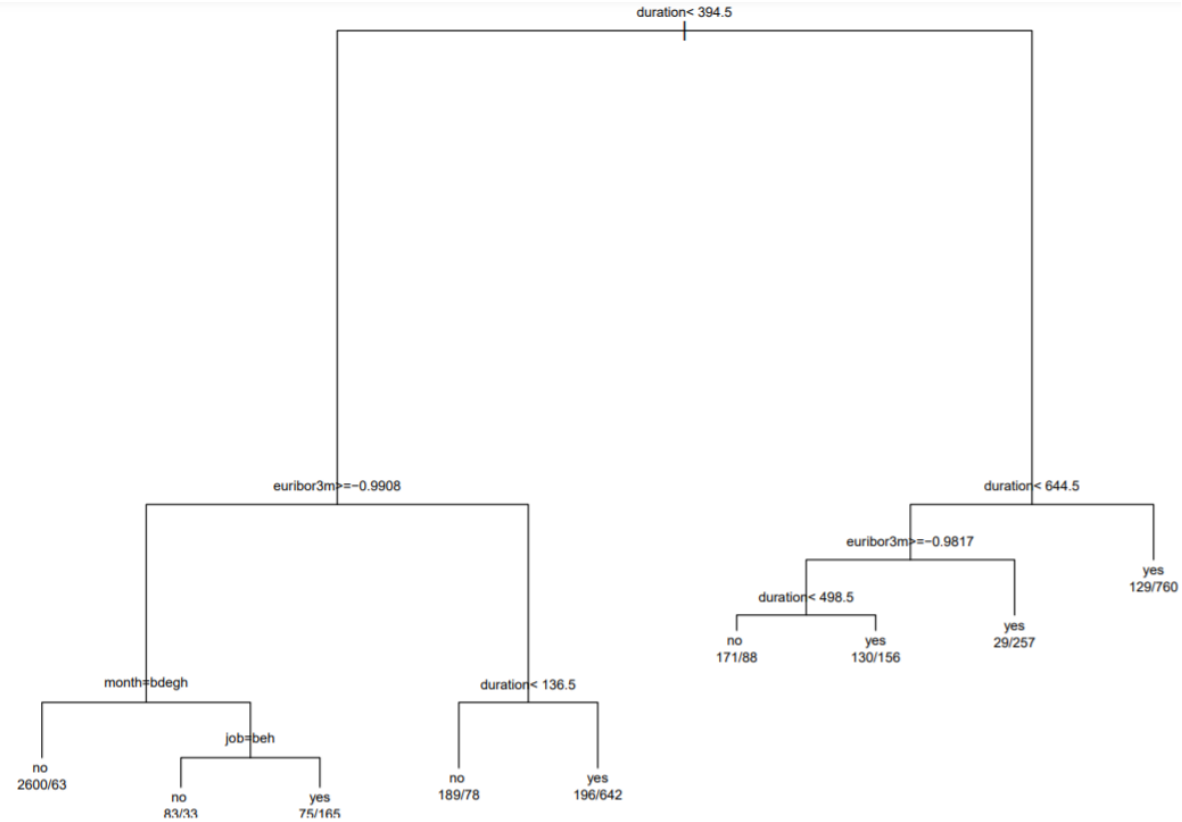
Árbol de decisión

Los árboles de decisión són métodos de predicción que nos sirven tanto para variables numéricas como para cualitativas, dónde a través de diagramas de construcciones lógicas se predice una variable categórica. Este método es fácilmente interpretativo y computacionalmente eficiente, aunque sea peor clasificador que otros métodos y su "accuracy" normalmente sea inferior a los otros algoritmos.
La idea básica es que se construye un árbol lógico formado por nodos, vectores de números, flechas y etiquetas. Todos los individuos empiezan en el nodo raíz y a partir de normas impuestas en los nodos se sigue un criterio de división y un criterio de "stop" (fin del algoritmo).

Para la validación de este modelo utilizaremos la k-fold cross validation con k = 10. Por lo tanto tendremos un total de 10 árboles de decisión de los cuales extraeremos la "accuracy" de cada uno para así obtener una mejor estimación de ella.

Una vez realizada la validación obtenemos una "accuracy" de 0.7108, la cuál parece bastante buena (ya que sabemos que los árboles de decisión no suelen ser un modelo muy potente).

Realizamos ahora un árbol de decisión con todas las observaciones, para obtener así el árbol final con el que trabajaríamos en un futuro.

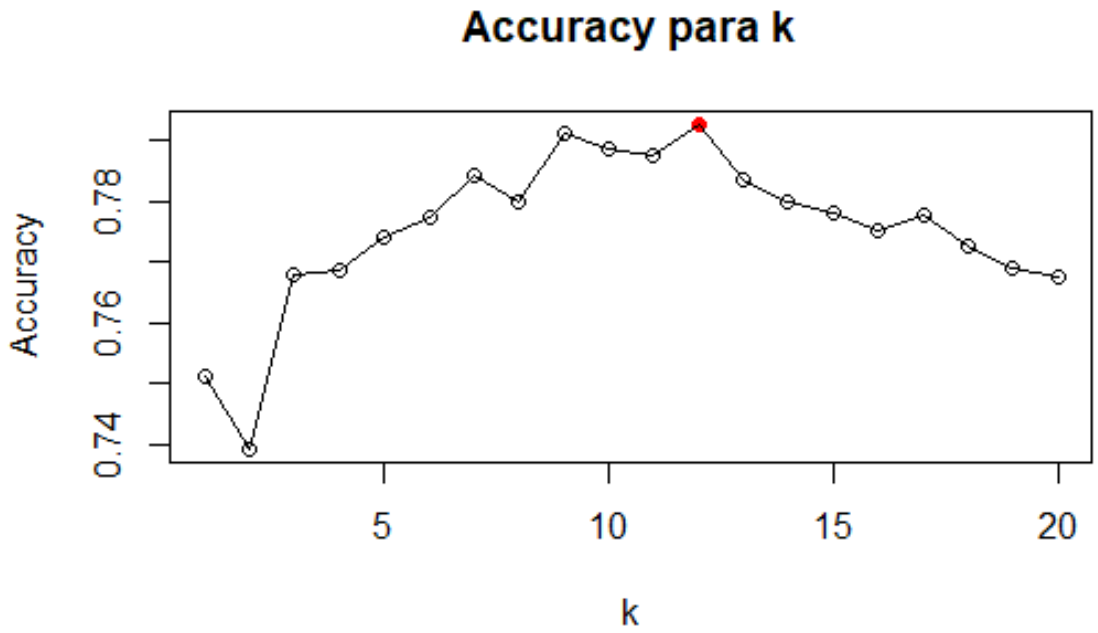


Para la variable month: bdegh = aug, jul, jun, may, nov.
Para la variable job: beh = blue-collar, management, services.

KNN

El método KNN visto anteriormente como algoritmo de estimación para valores perdidos, lo utilizaremos ahora como clasificador. Es un método no supervisado (no es necesaria una variable respuesta) que a través de patrones, clasifica el valor predicho teniendo en cuenta los k puntos que hay alrededor. Básicamente, la idea consiste en buscar a los vecinos más cercanos y ver cómo se comportan. En nuestro caso, haremos la clasificación según si el cliente se ha suscrito a un depósito a plazo o no.

En primer lugar, debemos decidir qué valor de k es el más óptimo para nuestro conjunto de datos. En nuestro caso realizamos una primera prueba con los valores de k de 1 a 10 y, como vimos que aún tenía tendencia a ascender, lo ampliamos para k de 1 a 20. Resumimos mediante el siguiente gráfico:



Podemos ver como la k que maximiza el “accuracy” es 12. Ahora lo que resta es hacer el modelo con k = 12, aplicando la k-fold cross validation para tener una mejor estimación de la “accuracy”.

Obtenemos una “accuracy” de 0.7798, la cual es mucho mejor que la observada anteriormente en el modelo de Decision Trees.

Naive Bayes

El método Naive Bayes es un algoritmo de clasificación. Para cada variable predictora, se calcula su probabilidad según la variable respuesta, utilizando el Teorema de Bayes que permite calcular la probabilidad de dar un préstamo (H) dadas las características del individuo (D):

$$P(H|D) = \frac{P(D|H)P(H)}{P(D)}$$

En nuestro caso, según el individuo y sus características, calculamos qué probabilidad es más alta, la de dar un préstamo o no.

Se dice que el método es ingenuo (naive) ya que asume que las características no están relacionadas entre sí. Al asumir independencia entre las características, los cálculos son mucho más simples. Cuando es apropiada la asunción de independencia, este algoritmo se comporta mejor que otros modelos de clasificación.

En nuestro caso hemos hecho dos modelos de entrenamiento con diez iteraciones. El primero usando la estimación de densidad de Kernel (estimación no paramétrica de funciones de densidad) y otro en el que no. Hemos calculado la media de los 'accuracy' según si los modelos usaban la estimación Kernel o no. La media de 'accuracy' más alta era la de los modelos sin la estimación de densidad de Kernel (78,61%).

Con el método de Naive Bayes, podemos observar las probabilidades de las diferentes variables para entender así, cómo son los individuos que contratan el depósito y los que no. A primera vista, sólo observamos diferencias importantes entre la probabilidad de contratar o no un depósito según los valores de las siguientes variables:

- Contact: si se contacta a un cliente por teléfono, la probabilidad de que contrate el depósito es de 83,58%. Si se contacta por teléfono fijo 16,41%

- Duration: vemos que como más largas sean las llamadas a los clientes, más probabilidad hay de que contraten.

- Campaign: cuantas más veces se contacte con el cliente, más probabilidades hay de que contrate.

- Contacted: si se contacte con el cliente, obviamente, es más probable que contrate el depósito

- Cons.price.idx: como más bajo es el índice, más probable es que el cliente contrate

- Con.conf.idx: como más alto es el índice, más probable es que el cliente contrate

- Euriobor3m: como más bajo es el índice, más probable es que el cliente contrate

```
---------------------------------------
 ::: cons.price.idx (Gaussian)
---------------------------------------

cons.price.idx         no        yes
          mean  0.1399194 -0.2699937
          sd    0.9032169  1.0663706

---------------------------------------
 ::: cons.conf.idx (Gaussian)
---------------------------------------

cons.conf.idx          no         yes
         mean -0.05063356  0.07493928
         sd    0.85508689  1.18368601

---------------------------------------
 ::: euribor3m (Gaussian)
---------------------------------------

euribor3m         no        yes
     mean  0.3653940 -0.5205236
     sd    0.8728939  0.9306161
```

En esta imagen, mostramos la salida de R en la cuál vemos la distribución de cada índice, según si el cliente contrata o no.

Equipo 2

Este equipo se encarga de realizar los métodos ACM, SVM, Random Forest y XGBoosting.

ACM

El Análisis de correspondencias múltiples (ACM) es una técnica de análisis de datos para datos categóricos nominales. ACM Puede ser visto como una extensión del análisis de correspondencias simples (AC) para un conjunto grande de variables categóricas.

Este método se lleva a cabo aplicando el algoritmo de AC a la matriz de indicadores o la tabla de Burt formada a partir de estas variables. Una matriz de indicadores es una matriz de individuos x variables, donde las filas representan a los individuos y las columnas son indicadores binarios que representan a las categorías de las variables. Analizar la matriz de indicadores permite la representación directa de los individuos como puntos en espacio geométrico. La tabla de Burt es la matriz simétrica que contiene las tabulaciones cruzadas para cada pareja de variables categóricas y es el análogo de la matriz de covarianzas para variables continuas. El análisis de la tabla de Burt es una generalización más natural del Análisis de Correspondencias simple, y los individuos, o las medias de los grupos.

En la aproximación mediante la matriz indicadora, las asociaciones entre variables son representadas gráficamente, facilitando la interpretación de la estructura de los datos. Igual que en el Análisis de Componentes Principales (ACP), el primer eje es la dimensión más importante, el segundo eje la segunda más importante, y así sucesivamente, en relación la cantidad de varianza explicada.

Clasificación de las variables

Para realizar el estudio del método del ACM hemos de utilizar las nuevas codificaciones de nivel de las variables categóricas. A continuación se presentan las variables activas y las suplementarias.

- Variables Activas (Categóricas):
  - *Job*
  - *Marital*
  - *Housing*
  - *Loan*
  - *Contact*
  - *Month*
  - *Poutcome*
  - *Y*
  - *Contacted*

- Variables Suplementarias (Categóricas)
  - *Education*
  - *Day_of_week*

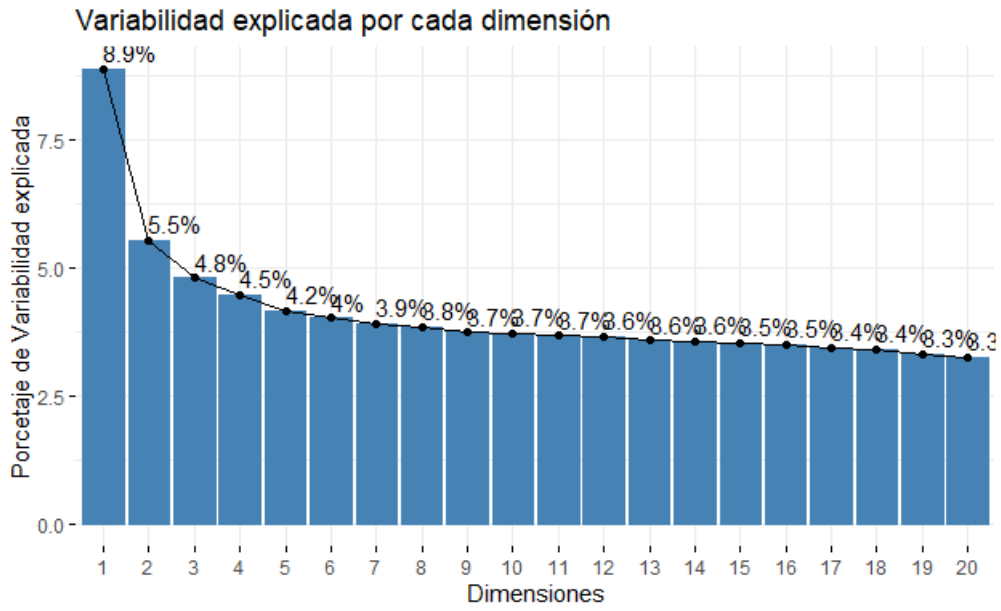
- Variables Suplementarias (Numéricas)
  - *Age*
  - *Duration*
  - *Campaign*

- *Pdays*
- *Previous*
- *Cons.price.idx*
- *Cons.conf.inx*
- *Euribor3m*

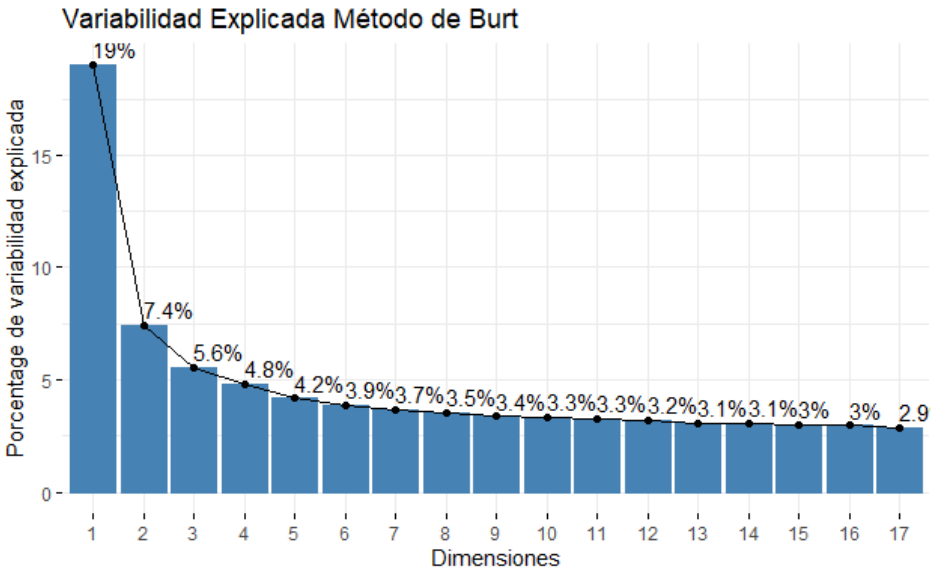
Elección de método

Vamos a comparar en R los dos métodos del ACM, el indicator, que es el método por defecto en R, y el Burt. Para comparar los dos métodos utilizaremos la varianza explicada en las 10 primeras dimensiones de cada método.

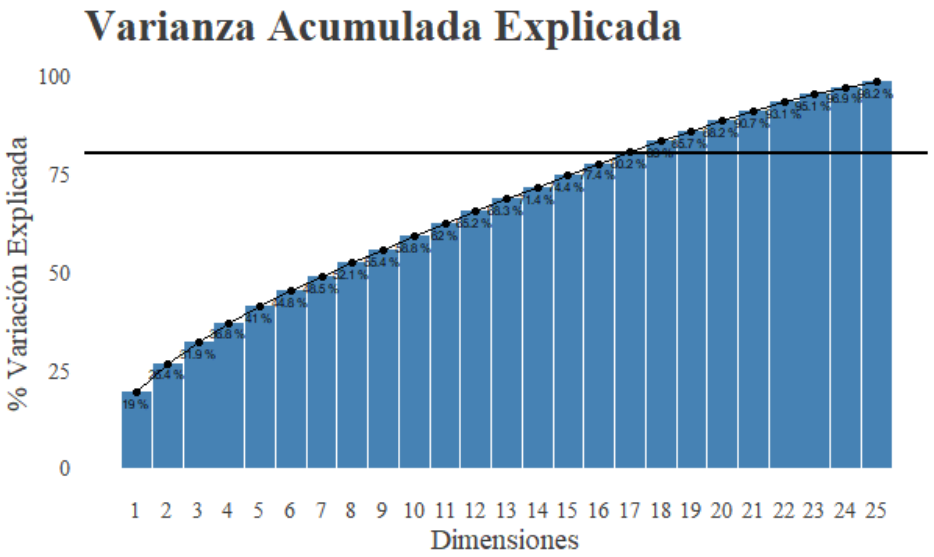
El gráfico scree plot muestra el porcentaje de la varianza explicada por cada dimensión. Para poder visualizar bien el gráfico se han seleccionado sólo las 20 primeras dimensiones en el gráfico del método por defecto y las 17 dimensiones para el gráfico del método Burt .



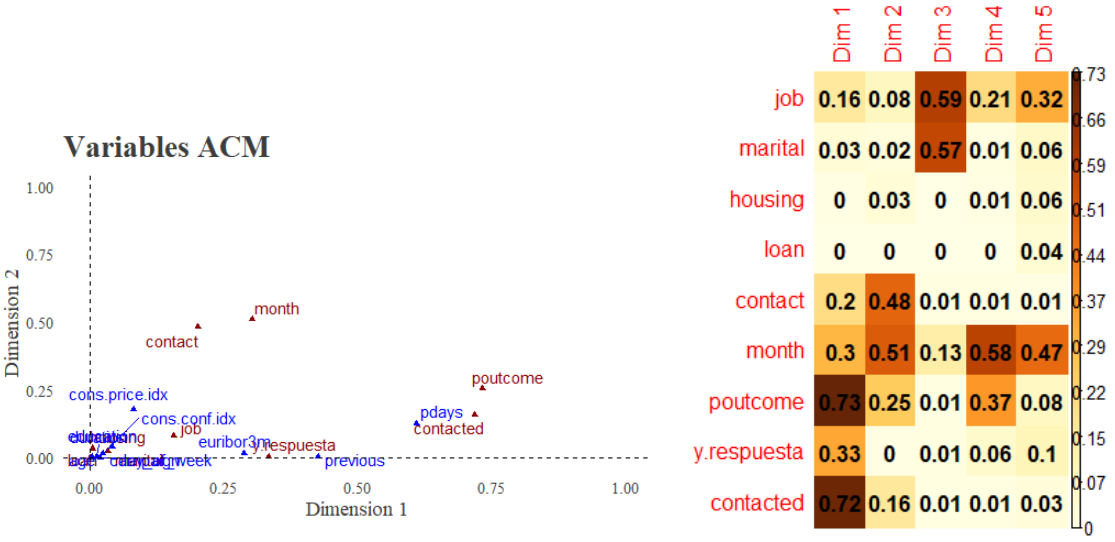
En el gráfico de la varianza explicada del método por defecto vemos que en las diez primeras dimensiones explican el 47% de la varianza.



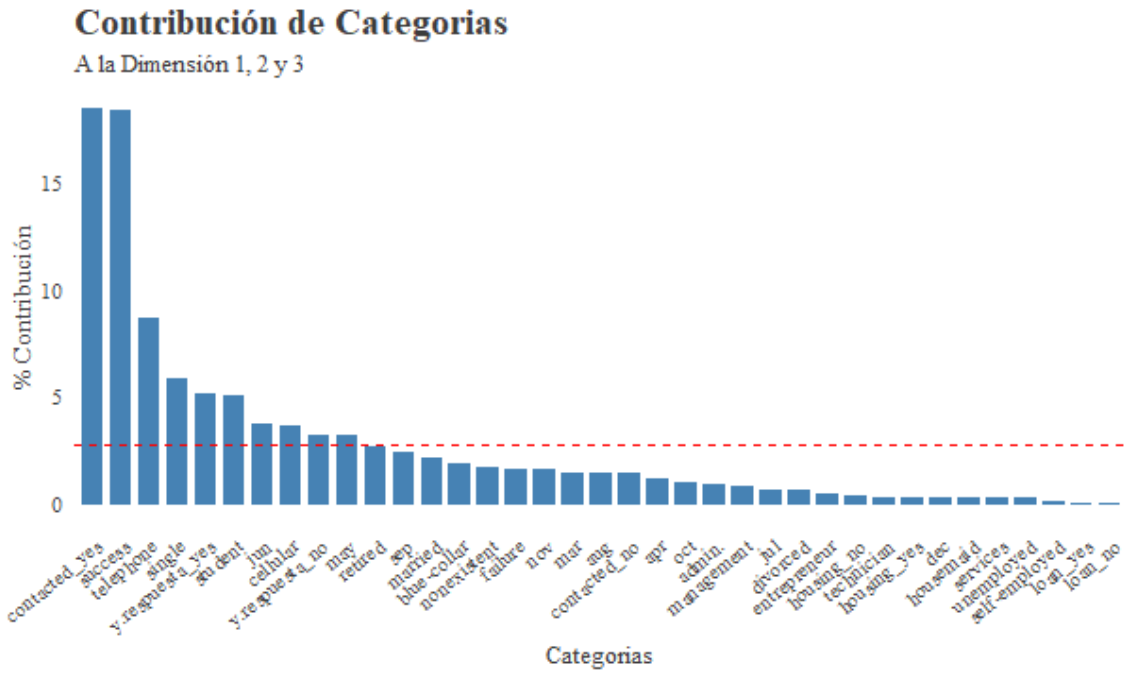
En el método Burt podemos ver que el total de la varianza explicada en las 10 primeras dimensiones es del 58.8%.
Por lo tanto escogemos el método de Burt, ya que tiene mayor varianza explicada en las primeras dimensiones. Aún teniendo un valor mayor en el método de Burt, los resultados que obtengamos no van a ser muy precisos ya que la variabilidad explicada es baja, lo cuál es normal teniendo en cuenta el número de observaciones que tenemos y la suma total de niveles de cada variable.
En el siguiente gráfico vemos la varianza explicada acumulada del método Burt según el número de dimensiones. Cuando se tienen en cuenta las 25 dimensiones la varianza acumulada es del 98.2%.



Contribución de las variables



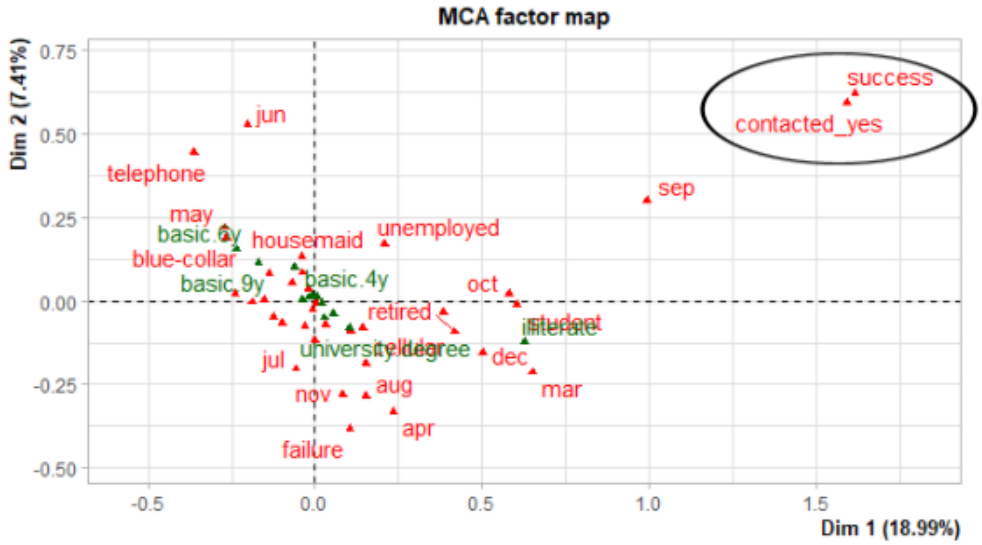
En la gráfica de arriba podemos observar que tanto pesa cada una de las 7 variables en las primeras 5 dimensiones. Vemos que en la dimensión 1 las dos variables que más pesan son *Poutcome* y *Contacted*. Las variables que más pesan en la dimensión 2 son *Contact* y *Month*. En la dimensión 3 *Job* y *Marital* y la que más aporta en la dimensión 4 y 5 es *Month*.
Las variables *Housing* y *Loan* pesan prácticamente 0 en las cinco primeras dimensiones.



Este gráfico de barras muestra las 37 categorías con una mayor contribución porcentual a cada eje. La línea horizontal roja indica la contribución porcentual promedio de todas las categorías.

Analizando el gráfico, se puede observar que las categorías referentes son (“Contacted_yes”, “Success” y “Telephone”), són las que definen mejor las dimensiones 1, 2 i 3. Por lo tanto, se puede ver que en las dimensiones de la uno a la tres está definida

mayoritariamente por si el cliente ha sido contactado, la campaña anterior tuvo éxito y si fue llamado por el teléfono fijo.

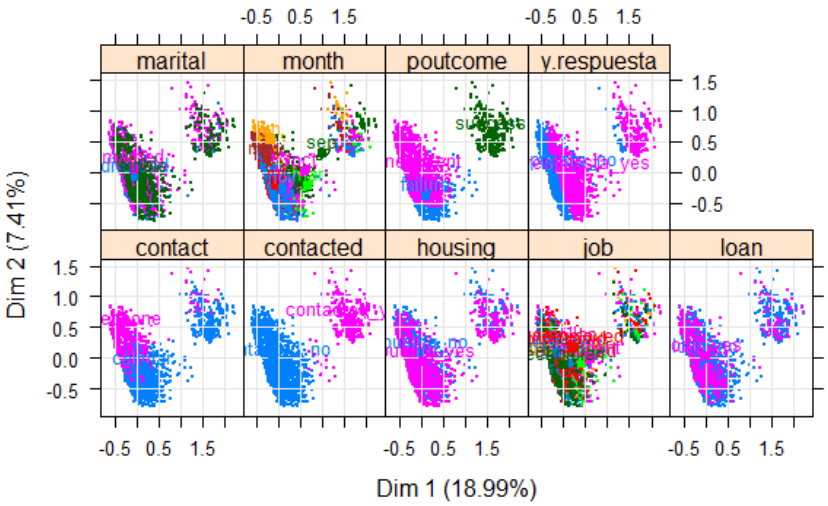


En el factor map queda reflejada la relación que hay entre las diferentes categorías de las variables. Las categorías de las variables con un perfil similar están agrupadas de forma conjunta.

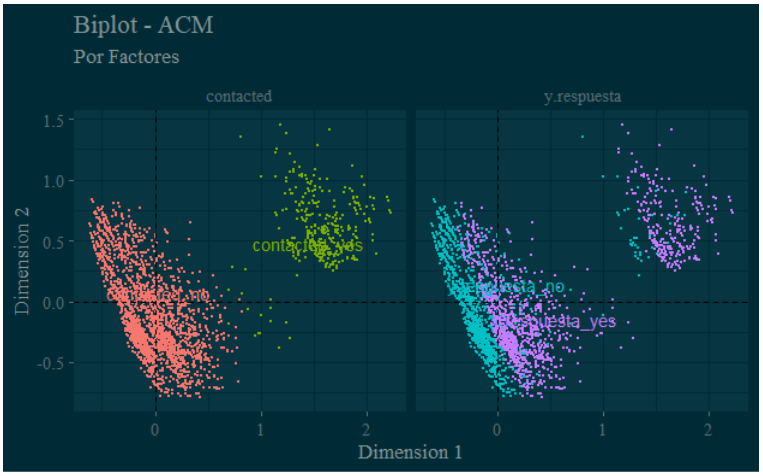
Por ejemplo, se puede observar que las categorías “Succes” y “Contacted_yes” de las variables *Poutcome* y *Contacted* respectivamente son muy cercanas entre sí, como era de esperar.

Las categorías de las variables negativamente correlacionadas están situadas en cuadrantes opuestos.

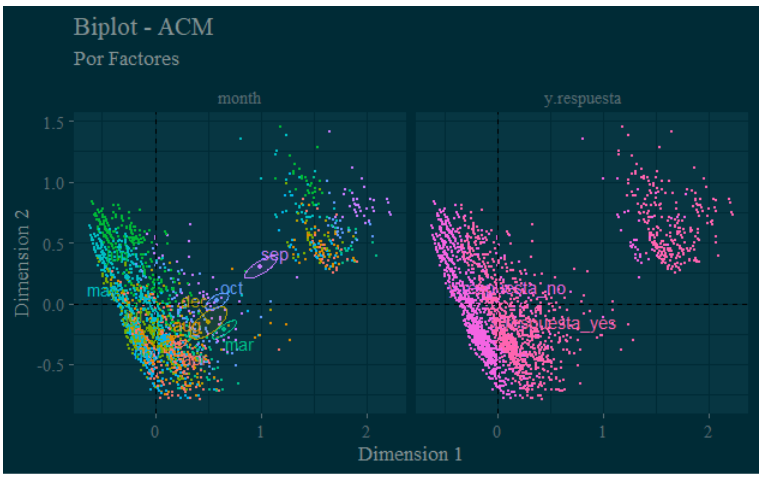
Por último, para ver con más detalle las relaciones entre algunas variables y categorías, podemos observar los gráficos de elipses, primero todos conjuntamente, y seguidamente algunos que interesen por separado.



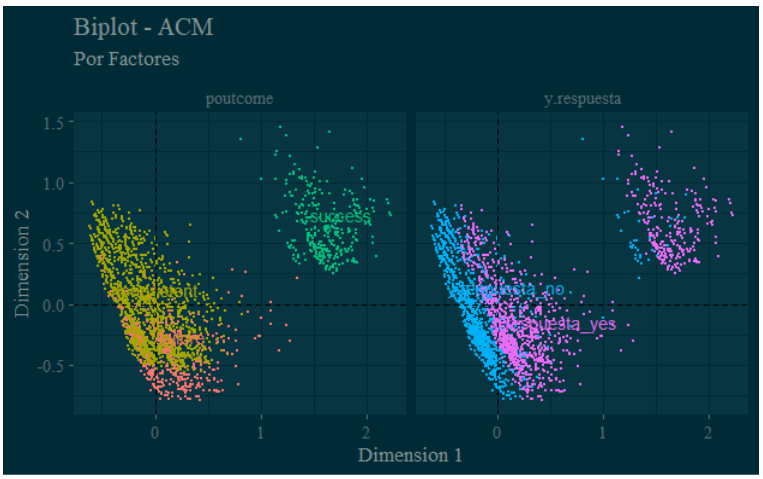
Comparación de alguna variable con la variable respuesta “*y*”.



Como podemos observar, el hecho de que un cliente haya sido contactado aporta una mayor proporción de clientes que se suscriben al depósito que si no se hubiese llamado.



En este gráfico no podemos deducir exactamente que més es mejor para la contratación del servicio, pero sí que podemos decir que el peor mes es el de mayo, ya que se puede apreciar que la posición del color turquesa (que representa al mes de mayo) representa la no contratación del servicio en la variable Y



Se observa que la categoría “succes” de la variable *Poutcome* y “yes” de la variable *respuesta* están agrupadas, es lógico pensar que si una campaña anterior ha tenido éxito esos clientes son las más propensos a contratar un servicio de nuevo.

Support Vector Machine (SVM)

El SVM (Support Vector Machines) es un algoritmo de aprendizaje supervisado capaz de resolver problemas de clasificación, tanto lineales como no lineales. Este método, en lugar

de buscar hipótesis que cometan pocos errores (minimización del riesgo empírico), se basa en la minimización del riesgo estructural, es decir, busca construir modelos que estructuralmente tengan poco riesgo de cometer errores ante clasificaciones futuras.

La idea básica detrás de los SVM es la siguiente: si la frontera definida entre dos regiones con elementos de clases diferentes es compleja, en lugar de construir un clasificador complejo que reproduzca dicha frontera, lo que se intenta es «doblar» el espacio de datos en un espacio de mayor dimensionalidad de forma que con un único corte (es decir, un clasificador muy sencillo) se puedan separar fácilmente ambas regiones.

Implementación del algoritmo:

1) Paso previo al entrenamiento del modelo

En primer lugar, dividiremos los datos en dos bases diferentes:

1. La base de datos “training” tendrá el 80% de los datos y la usaremos para entrenar nuestro modelo.
2. La base de datos “testing” tendrá el 20% restante y se usará para validar la precisión del modelo y para observar si este presenta un problema de “overfitting”.

2) Entrenamiento del modelo

Iniciamos aplicando el algoritmo del SVM a nuestros datos de entrenamiento con los parámetros que nos da por defecto R (SVM-Kernel = radial y Cost = 1).

A continuación, se muestra la matriz de confusión del modelo y su “Accuracy” para los datos de entrenamiento:

```
Training set confusion matrix :
     y_train_pred
        no  yes
  no  2507  375
  yes  210 1584
Success ratio on training set :  87.4893071000855 %
```

3) Optimización de hiperparámetros

El algoritmo Support Vector Machine tiene varios parámetros a tener en cuenta:

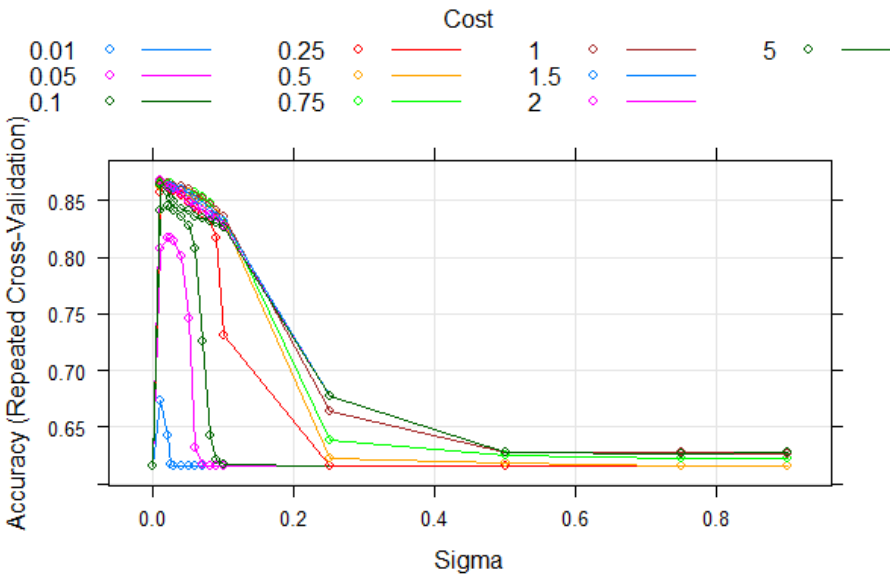
- Funciones Kernel: utilizadas para transformar un problema no lineal en el espacio original X en un problema lineal en el espacio transformado F.

- Cost: controla el equilibrio bias-varianza y la capacidad predictiva del modelo, ya que determina la severidad permitida respecto a las violaciones sobre el margen. En otras palabras, se necesita fijar un margen de separación entre observaciones a priori.

- Sigma: controla el nivel de no linealidad introducido en el modelo.

Aplicamos el algoritmo con diferentes hiperparámetros junto a la técnica de validación cruzada. De esta forma, se calculan todos los distintos modelos, uno por cada combinación, aplicando la técnica “k fold cross validation”. La medida que utilizamos para verificar el ajuste es el “Accuracy” (exactitud).

En el siguiente gráfico observamos cómo evoluciona el “accuracy” en función de diferentes valores de los hiperparámetros:



Los parámetros elegidos para nuestro modelo final son:

Cost = 1.5 y Sigma = 0.1

4) Ajuste del modelo

Una vez seleccionados los hiperparámetros y habiendo validado nuestro modelo, ejecutamos el método con los parámetros óptimos y hacemos las predicciones con nuestros datos de test, los cuales no han visto en ningún momento el modelo. Obtenemos los siguientes resultados:

Para los datos de entrenamiento:

```
Training set confusion matrix :
     y_train_pred
        no  yes
  no  2565  317
  yes  199 1595
Accuracy on training set :  88.9649272882806 %
```

Para los datos de test:

```
Test set confusion matrix :
     y_test_pred
       no yes
  no  615 105
  yes  56 392
Accuracy on test set :  86.2157534246575 %
Error on test set :  13.7842465753425 %
```

Como vemos, habiendo optimizado los parámetros hemos aumentado nuestra precisión para los datos de entrenamiento, hasta el 89% de “accuracy”, lo que aplicado a nuestros datos de test nos da una “accuracy” del 86.2%. Esta será la precisión final.

Consideramos también otras medidas de precisión:

Recall (Sensitivity): Indica la proporción de casos positivos (“yes” o 1) que son detectados.

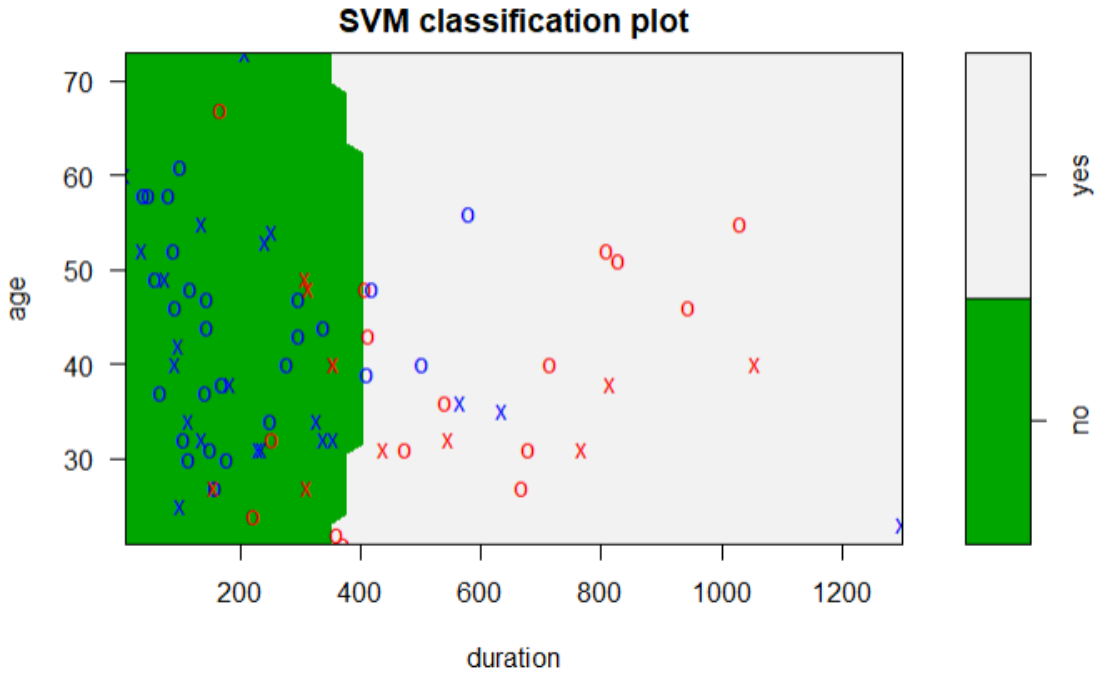
Specificity: Es la proporción de casos positivos respecto al número total de casos predichos positivos.

Precisión: Es la proporción de casos negativos respecto al número total de casos predichos negativos.

F-Measure: Media armónica de Recall y Precisión. ($\frac{2PR}{P+R}$)

Sensivity	Recall	Specifity	F1
85.42	85.42	87.5	88.43

A continuación, se muestra un gráfico de Support Vector Machine en dos dimensiones (“age” y “duration”), en el que observamos una muestra de 75 de las predicciones para la base de datos de test:



5) Validación del modelo

La validación cruzada o cross-validation es una técnica utilizada para evaluar los resultados de un análisis estadístico y garantizar que son independientes de la partición entre datos de entrenamiento y prueba. Consiste en repetir y calcular la media aritmética obtenida de las medidas de evaluación sobre diferentes particiones. Se utiliza en entornos donde el objetivo principal es la predicción y se quiere estimar la precisión de un modelo que se llevará a cabo a la práctica

Para nuestro modelo calculamos el “accuracy” en 10 ocasiones para los datos de entrenamiento y obtenemos como resultado un 86.71% de exactitud.

XGBoost

El “Extreme Gradient Boosting” (XGBoost) es un algoritmo predictivo supervisado que utiliza el principio de “boosting”. Se trata de uno de los algoritmos de “machine learning” más usados en la actualidad, ya que obtiene buenos resultados de predicción con relativamente poco esfuerzo, en muchos casos, mejores que los devueltos por modelos más complejos computacionalmente. Su eficiencia destaca particularmente para problemas con datos heterogéneos.

La idea detrás del boosting es generar múltiples modelos de predicción “débiles” de forma secuencial, y que cada uno de estos tome los resultados del modelo anterior, para generar un modelo más “fuerte”, con una mayor capacidad predictiva y estabilidad en sus resultados. Para conseguirlo se emplea un algoritmo de optimización, en este caso Gradient Descent (descenso de gradiente).

Durante el entrenamiento, los parámetros de cada modelo débil son ajustados iterativamente tratando de encontrar el mínimo de una función objetivo, que puede ser la proporción de error en la clasificación, la raíz del error cuadrático medio (RMSE), la log verosimilitud negativa (Logloss) o alguna otra. Para este problema se ha decidido utilizar la métrica Logloss, que es la que aplica la función de “caret” por defecto.

Cada modelo es comparado con el anterior. Si un nuevo modelo tiene mejores resultados, entonces se toma este como base para realizar nuevas modificaciones. Si, por el contrario, tiene peores resultados, se regresa al mejor modelo anterior y se modifica ese de una manera diferente. Este proceso se repite hasta llegar a un punto en el que la diferencia entre modelos consecutivos es insignificante, lo cual nos indica que hemos encontrado el mejor modelo posible, o cuando se llega al número de iteraciones máximas definido por el usuario.

XGBoost usa como sus modelos débiles árboles de decisión de diferentes tipos, que pueden ser usados para tareas de clasificación y de regresión.

Creación de un modelo predictivo mediante XGBoost

1. Conversión de variables categóricas a numéricas

Antes de aplicar el algoritmo debemos hacer algunos cambios en nuestro dataset para poder ejecutarlo. XGboost requiere matrices numéricas para su implementación, por lo que será necesario transformar nuestras variables categóricas a numéricas.

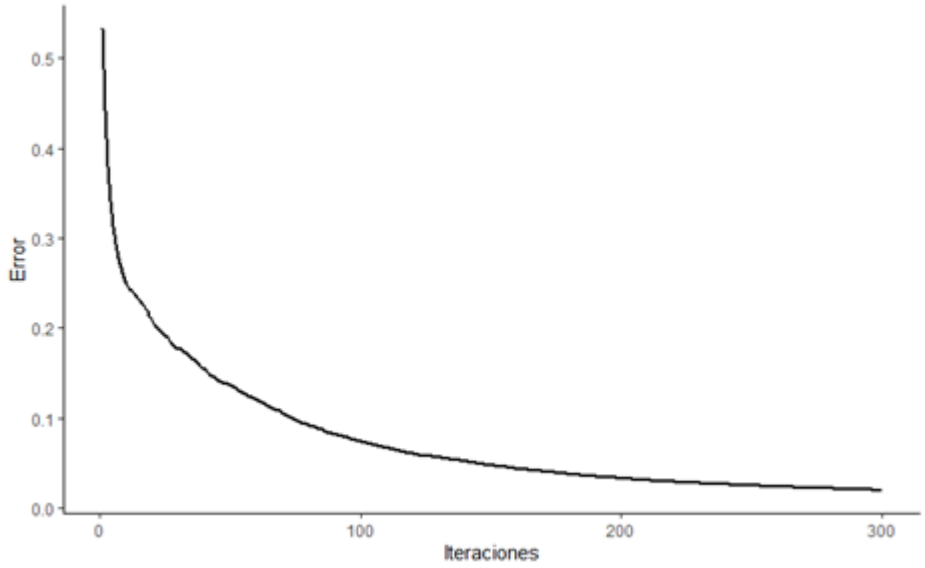
Tomamos como ejemplo de la transformación realizada con la variable “education”:

Education_Categorica	Education_Numerica
basic.4y	0
basic.6y	1
basic.9y	2
high.school	3
illiterate	4
professional.course	5
university.degree	6

2. Entrenamiento del modelo predictivo

Una vez tenemos preparados los datos, aplicamos el algoritmo Xgboost para nuestros datos de entrenamiento. Inicialmente, asignamos los argumentos más conservadores, los que suele asignar el algoritmo por defecto.

eta = 0.3 / max_depth= 6 / gamma = 0 / subsample = 0.5 / min_child_weight = 1



El error del modelo converge en 0 cuando las iteraciones tienden a infinito, ya que cada iteración el algoritmo mejora el modelo.

A raíz de nuestro modelo, generamos predicciones de nuestra variable respuesta con los valores observados. Las predicciones del modelo tienen una precisión del 100%, lo que es normal dado que hemos entrenado el modelo con los mismos datos con los que hemos realizado la predicción. Teniendo una precisión tan grande es muy importante comprobar el sobreajuste, es posible que estemos realizando un modelo que predice muy bien los datos del entrenamiento, pero no datos todavía no observados.

3. Validación Cruzada

Como se ha comentado previamente, el error o la precisión que muestra el modelo entrenado se corresponde con el error que comete el modelo al predecir observaciones que sí ha "visto", por lo tanto, no es una estimación realista de cómo se comporta el modelo ante nuevas observaciones. Para conseguir una estimación más certera, y antes de recurrir al conjunto de test, se pueden emplear estrategias de validación basadas en *resampling.* Existen muchas técnicas de validación, cada uno funciona internamente de forma distinta, pero todos ellos se basan en la idea de ajustar y evaluar el modelo múltiples veces con distintos subconjuntos creados a partir de los datos de entrenamiento, obteniendo en cada repetición una estimación del error. El promedio de todas las estimaciones tiende a converger en el valor real del error de test. En nuestro análisis utilizaremos la técnica de validación cruzada(CrossValidation).

La validación cruzada consiste en repetir y calcular la media aritmética obtenida de las medidas de evaluación sobre diferentes particiones. Es decir, realizamos una partición de los datos en entrenamiento y test k veces, y para cada partición diferente ejecutamos el modelo obteniendo una precisión diferente. Una vez completadas, se calcula la media de todas ellas, obteniendo así el "Accuracy" definitivo.

nrounds	eta	max_depth	gamma	subsample	min_child_weight	colsample_bytree	Accuracy
300	0.3	6	0	0.5	1	1	0.8541445

Con la validación cruzada se obtiene un 85,9% de precisión, esto es indicador de sobreajuste del modelo, ya que para el conjunto de entrenamiento tenemos una precisión mucho mayor. Este sobreajuste puede ser debido a los parámetros de nuestro modelo.

4. Optimización de Hiperparámetros

El algoritmo XGboost tiene múltiples hiperparámetros que pueden ser sintonizados, estos se pueden clasificar en tres:

- *Parámetros generales*: Guían el funcionamiento general
- *Parámetros de refuerzo*: Guían al reforzador individual (árbol/regresión) en cada paso
- *Parámetros de la tarea de aprendizaje*: Guían la optimización realizada

Para este ejemplo, por motivos computacionales, solo haremos uso de los que hemos considerado más relevantes:

- **eta**: controla la tasa de aprendizaje. Escala la contribución de cada árbol en un intervalo . Se utiliza para evitar el sobreajuste.
- **gamma**: es otro parámetro de regularización para la poda de árboles. Especifica la reducción de pérdida mínima requerida para hacer crecer un árbol. Cuanto mayor sea, más conservador será el algoritmo.
- **max_depth**: limita la profundidad a la que puede crecer cada árbol

- **min_child_weight**: si el paso de partición del árbol da como resultado un nodo hoja con la suma del peso de la instancia menor que min_child_weight, entonces el proceso de construcción renunciará a seguir partiendo.

- **subsample**: representa la proporción de submuestras de la muestra de entrenamiento. Una submuestra = 0.5 significa que el 50% de los datos de entrenamiento se usa antes de hacer crecer un árbol.

Para la búsqueda del parámetro hemos hecho una selección de los valores más comunes que pueden tomar estos, de forma que el algoritmo genere todas las combinaciones posibles y nos devuelva la óptima.

Niveles de los parámetros utilizados:

eta = (0.01,0.05,0.1,0.3)

max_depth = (3,6)

gamma = (0,1)

subsample = (0.5,0.75,1)

min_child_weight = (1,3)

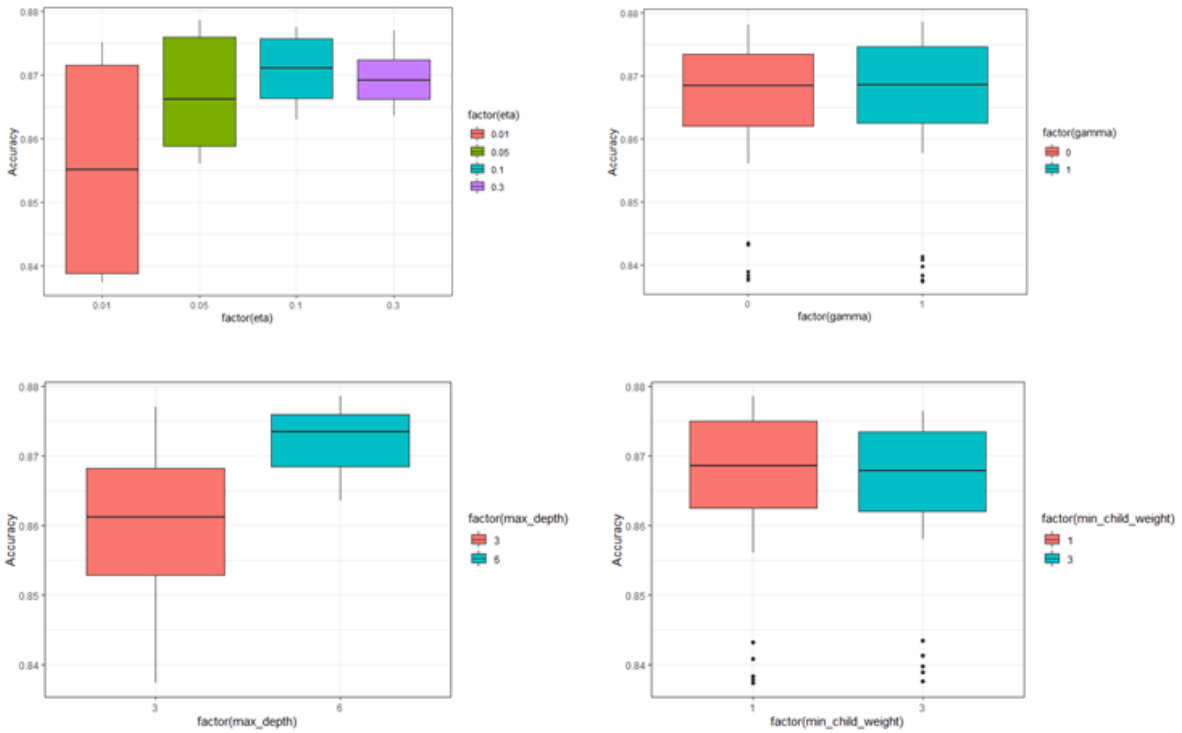
eta	max_depth	gamma	subsample	min_child_weight	colsample_bytree
0.01	3	0	0.5	1	1
0.05	3	0	0.5	1	1
0.10	3	0	0.5	1	1
0.30	3	0	0.5	1	1
0.01	6	0	0.5	1	1
0.05	6	0	0.5	1	1

Una vez creadas todas las combinaciones posibles de los hiperparámetros, aplicamos el algoritmo, junto a la técnica de validación cruzada. De esta forma, se calculan todos los distintos modelos, uno por cada combinación, aplicando la técnica de validación cruzada. La medida que utilizamos para verificar el ajuste es el “Accuracy”.

eta	max_depth	gamma	colsample_bytree	min_child_weight	subsample	nrounds	Accuracy
0.05	6	1	1	1	0.75	50	0.8791669

Con la optimización de parámetros hemos mejorado la precisión del modelo respecto a los datos no observados, además, hemos eliminado el sobreajuste del modelo.

A continuación, se presentan gráficos de la precisión de los modelos según algunos de los parámetros.

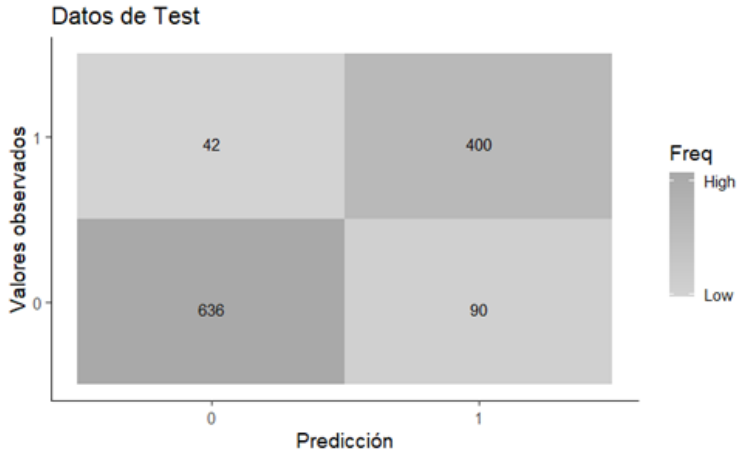


5. Ajuste del Modelo

Una vez seleccionados los mejores hiperparámetros y habiendo comprobado la validez de nuestro modelo, ejecutamos el algoritmo con los parámetros óptimos y hacemos las predicciones respecto a los datos de test. De esta forma podremos ver la precisión real de nuestro modelo.

Accuracy_Train	Accuracy_Test	Error_Train	Error_Test
90.21 %	88.7 %	9.8 %	11.3 %

Como vemos, habiendo hecho las transformaciones hemos disminuido la precisión del modelo respecto a los datos de entrenamiento hasta el 90,2%, y, al mismo tiempo, hemos aumentado la precisión del test hasta un 88,7%. Esta será la precisión final. A continuación, se muestra la matriz de confusión del modelo respecto a los datos de test:



Consideramos también otras medidas de precisión;

Recall: Indica la proporción de casos positivos(“yes”) que son detectados.

Specificity: Es la proporción de casos positivos respecto al número total de casos predichos positivos.

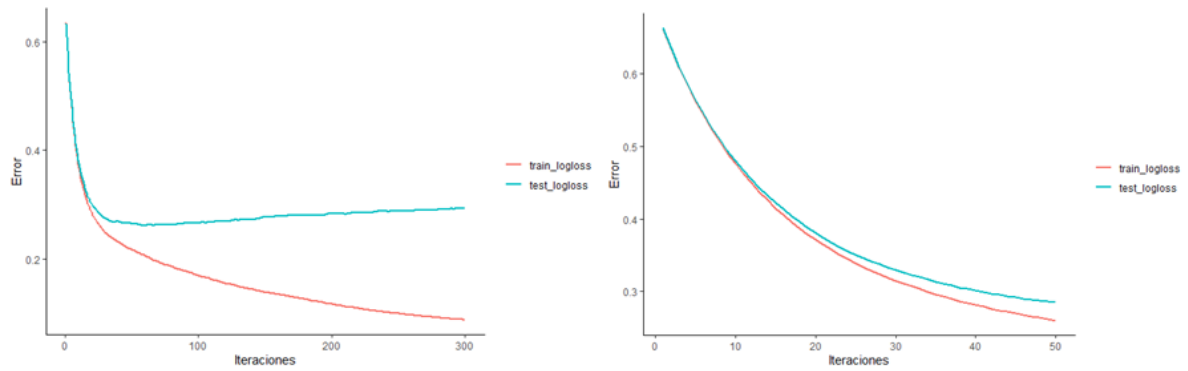
Precision: Es la proporción de casos negativos respecto al número total de casos predichos negativos.

F-Measure: Media armónica de Recall y Precision.

Specificity	Recall	Sensitivity	F1
0.876	0.905	0.905	0.8584

El Accuracy obtenido de nuestro modelo ha sido de un 88.7%. Si se comprueban las otras medidas de precisión, se pueden ver valores similares. El 90% de casos positivos son detectados, el 87,6% de los casos positivos predichos son realmente positivos.

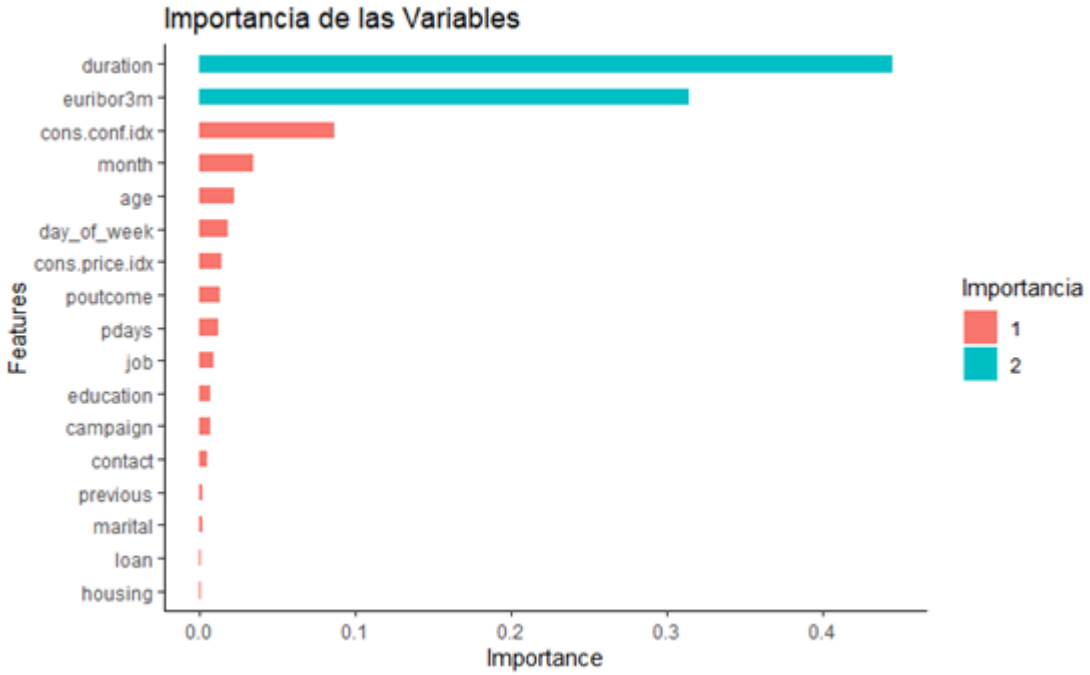
A partir de las curvas de aprendizaje, podemos ver que, como es normal, el rendimiento del modelo en el conjunto de datos de entrenamiento (línea roja) es mejor o tiene una pérdida menor que el rendimiento del modelo en el conjunto de datos de prueba (línea azul).



Otro detalle muy importante que podemos observar con esta gráfica es la evolución del error del modelo con los datos de test. A partir de las 50 iteraciones, el modelo comienza a aumentar el error cuando se aplica a los datos de test, es decir, a partir de las 50 iteraciones, el modelo tiende a una menor precisión . Con tantas iteraciones el modelo se vuelve demasiado específico con respecto a los datos de entrenamiento, de forma que cuando se aplica a datos más generales se pierde precisión, o lo que es lo mismo se aumenta el error. Por este motivo en el modelo final se disminuyen las iteraciones.

6. Importancia de las variables en el modelo

Una vez tenemos el modelo formulado y conocemos su precisión, es interesante analizar la importancia de las variables de nuestro dataset. El gráfico que se muestra revela qué variables han influido más en nuestro modelo.

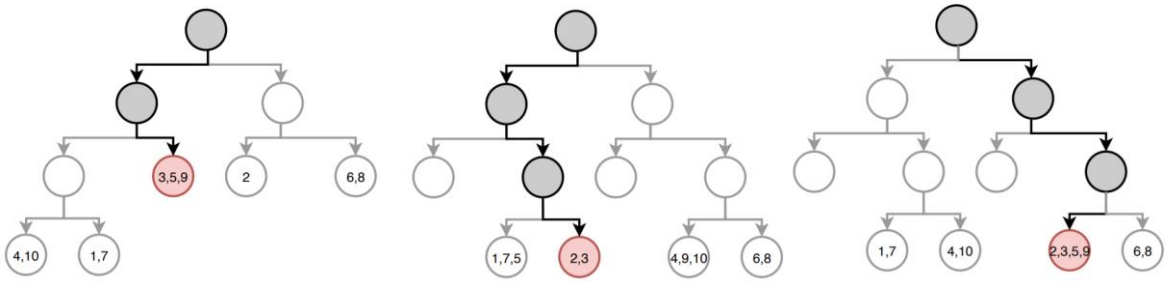


Podemos ver que hay dos variables en concreto que destacan sobre las demás. Estas son “duration” y “euribor3m”, lo que significa que estas dos variables son las que tienen mayor peso en la variable respuesta.

Random Forest

Mediante el uso del bootstrapping, el modelo Random Forest forma un conjunto de árboles de decisión individuales siendo entrenado cada uno de ellos con una muestra aleatoria extraída de los datos. Las observaciones se van distribuyendo en nodos generando la estructura de árbol en forma de ramificaciones. Las predicciones de una nueva observación se obtienen agregando las predicciones de todos los árboles individuales que forman el modelo.

La predicción es el valor de la media de las predicciones de los árboles que lo forman:



Por ejemplo en el primer árbol se llega a los identificadores 3, 5 y 9, entonces el valor respuesta Y saldrá de calcular la media de los tres valores asignados para esas muestras observadas.

Una vez sabiendo esto pasamos a ver el cálculo del Random Forest que hemos realizado.

1. Proceso inicial

Para nuestro modelo se ha usado como variable respuesta si se ha suscrito a un depósito a plazo y como predictores todas las variables disponibles.

Primero de todo dividimos la base de datos en dos diferentes. La base llamada de entrenamiento que tendrá el 80% de los datos y la del test con el 20% restante.

2. Optimización de Hiperparámetros

Una vez hecho esto tenemos que elegir los parámetros e hiperparámetros para poder crear el modelo. Nos basamos en la función ranger del paquete ranger y nos permite cambiar lo siguiente:

1) Formula
2) Data
3) Mode
4) Importance
5) Seed, para que los resultados sean reproducibles
6) num.trees: Número de árboles incluidos en el modelo
7) max.depth: Profundidad máxima que pueden alcanzar los árboles

8) mtry: Número de predictores considerados en cada división (si no se especifica se usa la raíz cuadrada del número total de predictores disponibles)

En nuestro caso hacemos dos casos distintos y nos quedamos con el óptimo. En uno definimos 50 árboles, 3 predictores y como profundidad máxima 1 y en el otro 100, 5 y 3.

3. Ajuste del Modelo

Con los dos hechos nos quedamos con el óptimo. En Random Forest no se sufre de overfit así que no por aumentar el número de árboles vamos a mejorar mucho más ya que alcanzado cierto número el error se estabiliza, sino haríamos pruebas con mayor número de árboles.

Ahora solo queda ver los resultados del modelo. Empezamos por la accuracy:

.metric <chr>	.estimator <chr>	.estimate <dbl>
accuracy	binary	0.8160821

Vemos que el valor es del 81% indicándonos que es un buen modelo.

La matriz de confusión es la siguiente:

```
          Truth
Prediction   no  yes
       no  2544  475
       yes  343 1314
```

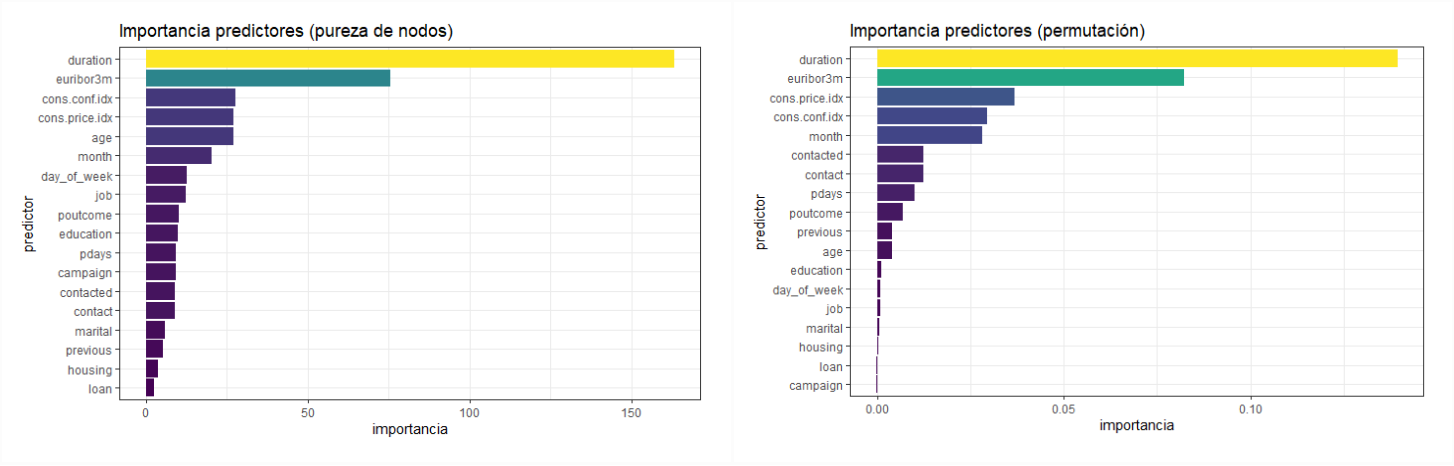
La mayor cantidad de errores se encuentran en casos en los que se da el crédito pero predecimos que no.

Vemos también la predicción de las probabilidades:

.pred_no <dbl>	.pred_yes <dbl>
0.4735927	0.5264073
0.8415559	0.1584441
0.8716616	0.1283384
0.8725356	0.1274644

4. Importancia de las variables en el modelo

Y finalmente para saber cuáles son las variables más importantes en cuanto a influir en la variable respuesta hacemos dos gráficos. Uno sobre la importancia de los predictores por pureza de los nodos y otro sobre la importancia de los predictores por permutación:

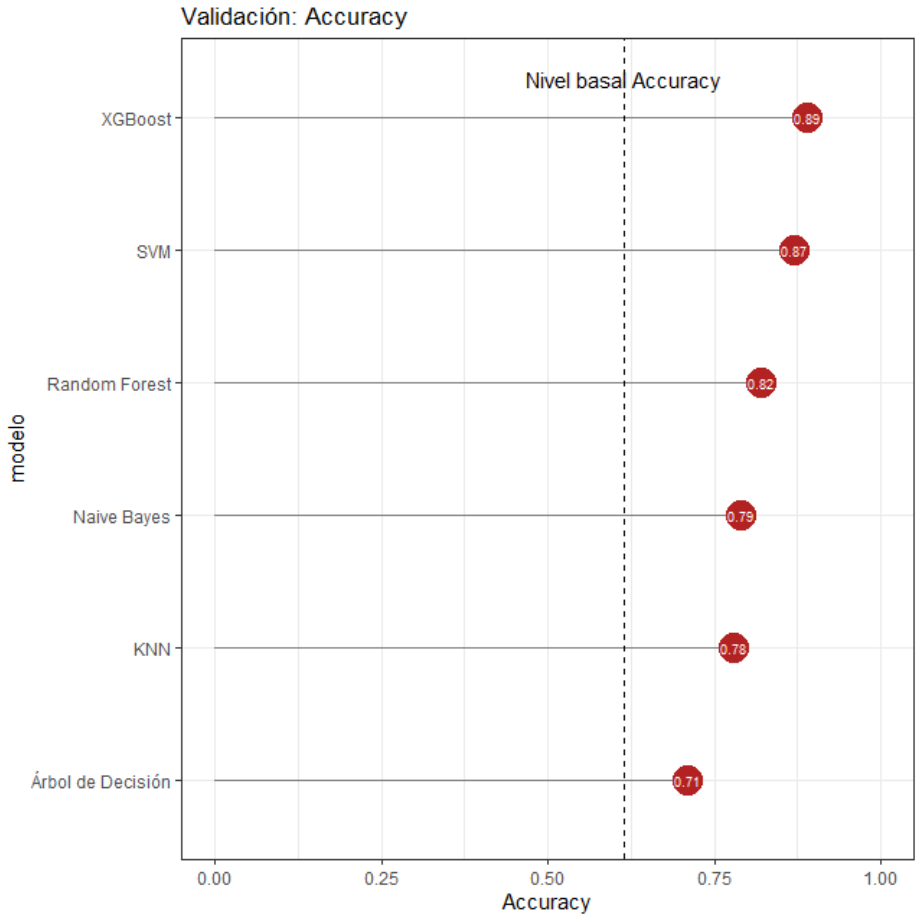


Vemos que hay dos variables que influyen sobre la variable respuesta de manera muy clara: duration y euribor3m. Destacar también a un menor nivel el índice de precio al consumidor y el índice de precio de confianza del consumidor.

Finalmente podríamos decir que estamos frente a un modelo que coincide con los anteriormente vistos en lo que a qué variables son las más influyentes y su precisión es elevada. Random Forest se podría considerar como una muy buena opción de clasificación y predicción.

Análisis comparativo de Modelos/ Conclusiones

Una vez que se han entrenado y optimizado distintos modelos, se tiene que identificar cuál de ellos consigue mejores resultados para el problema en cuestión, en este caso, predecir si se concede un crédito o no.
Dado que la variable respuesta del problema está balanceada, solo se ha tenido en cuenta la medida “Accuracy”. Esta medida es la que nos proporciona información de mayor relevancia.



Todos los modelos ajustados tienen un acierto en la predicción superior al nivel basal (línea discontinua). El modelo XGBoost consigue el accuracy promedio más alto, seguido muy de cerca por SVM y Random Forest. Para determinar si las diferencias entre ellos son significativas, sería necesario recurrir a un análisis de varianza.

Si la prioridad del modelo es obtener una precisión máxima, entonces los modelos óptimos serían el XGBoost o SVM, mientras que si la interpretabilidad es nuestra prioridad, entonces los Árboles de decisión o el método KNN serían los métodos utilizados.

Por otra parte, el coste computacional es otro factor que se debe tener muy en cuenta. En este ejemplo no se ha medido directamente, ya que nuestra base de datos es poco pesada. Sin embargo, los modelos con mayor precisión, como es lógico, son los modelos que se han percibido más lentos, especialmente el método de SVM.

Conclusiones

Para finalizar extraemos las ideas principales de cada apartado y las resumimos.

La primera conclusión que podemos extraer es que la base de datos con la que trabajamos no ha sido la óptima. Hay algunos conceptos económicos con los que no estamos muy familiarizados, podríamos haber escogido una base de datos sobre algún tema sobre el cual tuviéramos más conocimiento.
Por otro lado, algunas variables eran difíciles de interpretar y pensamos que no aportan mucha información sobre los clientes que contratan o no depósitos. Es decir, tenían muy poca relación sobre la variable respuesta.
Al tratarse de un banco podríamos haber escogido una variable respuesta que tuviera más connotación numérica o económica, o quizás algún tema que nos hubiera interesado más a todos.

Métodos supervisados

Para sacar las conclusiones de los métodos no supervisados lo que haremos es crear una tabla de los 6 métodos realizados y nos fijamos en su “accuracy”

Modelo	**Accuracy**
Árbol de Decisión	0.7108
KNN	0.7798
Naive-Bayes	0.7861
Random Forest	0.816
SVM	0.862
XGBoost	0.887

A la vista de los resultados el método ideal basándonos en la precisión del modelo es el “XGBoost“ con una “accuracy” del 89%.

Por otra parte, es importante mencionar que para la optimalidad de los modelos no se ha tenido en cuenta el coste computacional, un factor que con grandes bases de datos puede ser muy relevante.

Al comparar el Árbol de decisión con el XGBoost o el Random forest es bastante lógico que estos dos últimos tengan un ‘accuracy’ más alto ya que son un modelo evolucionado a partir del Árbol de decisión. También tenemos que considerar por contra que computacionalmente exigen más. Las conclusiones que vemos en estos dos modelos van muy de la mano en cuanto a resultados se refiere. Las variables que se ha visto que influyen más en la variable respuesta son “duration” y “euribor3m”.

Vemos que el accuracy del KNN es mejor que el del Árbol de decisión, pero sigue siendo de las más bajas, debido a que es uno de los modelos más simples tratados en este trabajo.

En algunos modelos como el Naive Bayes hemos observado que son bastante claros para sacar conclusiones de la base de datos y hacer interpretaciones.

Métodos no supervisados

El ACP nos ha indicado que de las 8 variables numéricas que partimos quedan reducidas a 5 dimensiones que engloban el 80% de la variancia de los datos. Teniendo en cuenta el estudio de las tres primeras componentes la variable “pdays” y “cons.price.idx” están muy relacionadas entre sí. Las variables “age” y “cons.conf.idx” también están muy relacionadas mutuamente y las variables que menos se pueden explicar con las tres primeras componentes son “campaign” y “duration”.

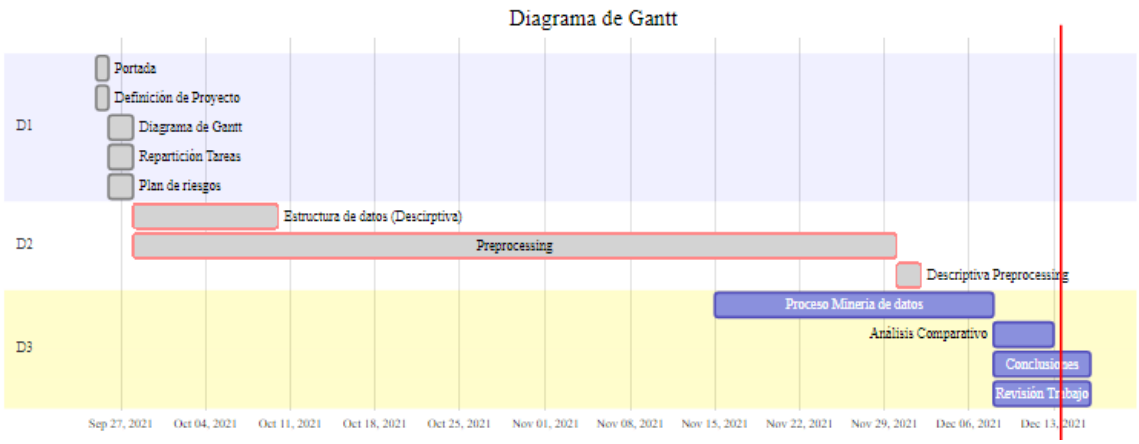
En el ACM vemos que con las 10 primeras dimensiones solo se explica el 59% de la varianza y con las 25 dimensiones se explica el 98% de la varianza. Las variables que más peso tienen en las primeras dimensiones son “poutcome”, “contact” y “month” y las que menos son “housing” y “loan”. Vemos que las variables que están más relacionadas con la variable respuesta “y” son la variable “contacted” y la variable “poutcome”. Otro aspecto que podemos observar es que en la variable “month” se puede ver un patrón, en el mes de mayo donde los clientes se suscriben menos al depósito.

Plan de trabajo real

Como equipo nos hemos coordinado con mucha facilidad y todo el mundo se ha involucrado con el trabajo. Al hacer reuniones de grupo, la gente presente siempre ha estado participativa y dando sus ideas y sus diferentes puntos de vista. También pensamos que siempre hemos tenido buen ambiente entre nosotros lo cual ha favorecido a la hora de crear un buen ambiente de trabajo.

Hemos tenido algunos contratiempos en algunos apartados pero en ningún momento han perjudicado de una manera directa al grupo. Por ejemplo, la descripción de variables antes del 'preprocessing' no se hizo según el calendario establecido y sufrió un leve retrasó. Se notificó con tiempo al equipo y no hubo ningún problema

Para graficar dichos resultados compararemos el Diagrama de Gantt inicial con el real con lo cual se podrán apreciar algunas diferencias.



Distribución de tareas real

La distribución real de las tareas ha sido la siguiente:

Tareas:	Encargado:
Estructura de los datos y descriptiva	
Un párrafo explicando la motivación del trabajo	Ivan Varea
Descripción formal de la estructura de datos.	Marc Larraz y Andrés Déniz
Análisis descriptivo univariante inicial de todas las variables.	Fèlix Bosch, Andreu Companys, Joan Martí
Descripción detallada del proceso de preprocessing de datos con una justificación de todas las decisiones decididas.	Marcel Canals, Andreu Companys, Fèlix Bosch y Joan Martí
Análisis descriptivo univariante de los datos preprocesados y discusión sobre la aleatoriedad de los datos faltantes.	David Botín y Adrià Pérez
Diseño de los dos procesos de minería de datos a seguir	
Representación gráfica de las técnicas de minería de datos que se enlazarán a lo largo del proceso. Este diagrama tiene que representar el proceso completo, incluyendo el preprocesamiento. Cada división tiene que poner en su diseño las operaciones que haya tenido en cuenta y si trabaja con las mismas variables o no que con la otra división	E1 y E2
Justificación del flujo representado	E1 y E2
Proceso de minería de datos de la división 1	
Resultados de aplicar al menos un método	E1

de cada família vista en clase (profiling, asociativo, discriminante, predictivo)	
Procés de mineria de dades de la divisió 2	
Resultados de aplicar al menos un método de cada família vista en clase (profiling, asociativo, discriminante, predictivo)	E2
Análisis comparativa	Todos
Conclusions generals	Todos
Plan de trabajo REAL	David Botín
Scripts d'R utilizados	Andreu Companys

Anexo

Descripción completa de la variables

Variable	Descripción	Tipo	Ejemplo
Age	Edad	Numérica	30
Job	Tipo de trabajo	Categórica	Admin
Marital	Estado civil	Categórica	Married
Education	Tipo de estudios	Categórica	University.degree
Default	¿Tiene crédito por defecto?	Categórica Binaria	No
Housing	¿Tiene un préstamo para la vivienda?	Categórica Binaria	Yes
Loan	¿Tiene préstamos personales?	Categórica Binaria	No
Contact	Tipo de comunicación para contactar	Categórica Binaria	Cellular
Month	Último mes de contacto	Categórica	May
Day_of_week	Último día de contacto en la semana	Categórica	Fri
Duration	Duración (en segundos) del último contacto establecido	Numérica	487
Campaign	Número de contactos establecidos con el cliente esta campaña	Numérica	2
Pdays	Número de días que han pasado desde el último contacto con el cliente	Numérica	12
Previous	Número de contactos realizados antes de esta campaña para este cliente	Numérica	0
Poutcome	Resultado de la campaña previa	Categórica	Noexistent
Emp.var.rate	Ratio de variación del empleado	Numérica	-1.8
Cons.price.idx	Índice de precios al consumidor, indicador mensual	Numérica	92.893

Cons.conf.idx	Índice de confianza del consumidor, indicador mensual	Numérica	-46.2
Euribor3m	Tasa 3 meses euribor, indicador diario	Numérica	1.313
Nr.employed	Número de empleados	Numérica	50099
Y (Variable respuesta)	¿El cliente se ha suscrito a un depósito a plazo?	Categórica Binaria	No