ENFOCAMENT BASAT EN DADES PER PREDIR L'ÈXIT DEL TELEMARKETING BANCARI

Aprenentatge Automàtic 1

David Bergés; Alex Carrillo; Roser Cantenys Sabà

Primavera 2019

Índex

INTRODUCCIÓ

En aquest treball s'estudia la venta de dipòsits bancaris a través de trucades de telemàrqueting. Durant aquestes campanyes, els agents realitzen trucades telefòniques a una llista de clients per vendre el producte financer (*outbound*) o, altrament, el client truca al centre de contacte per algun motiu aliè a la campanya i se l'informa sobre la subscripció del dipòsit (*inbound*). El resultat, així doncs, és un diàleg amb èxit o no de la venda.

OBJECTIUS DEL TREBALL

Aquest projecte té diversos objectius. El primer, de caire més general, es basa en la recerca i investigació de les estratègies de les campanyes de comercialització de vendes i la segmentació de clients per assolir un objectiu específic de negoci. És a dir, es vol estudiar la influència dels diferents paràmetres a l'hora de comprar o no un dipòsit bancari.

El segon objectiu consisteix en la teoria portada a la pràctica. Això és replantejar-se el màrqueting centrant-se en minimitzar el nombre de trucades (*outbound*) maximitzant el nombre de clients que contracten el dipòsit bancari. En aquest projecte, es vol potenciar la tasca de seleccionar el millor conjunt de clients, el més probable de subscriure's a un producte.

DADES

L'estudi considera dades reals recollides d'una institució bancaria portuguesa de maig de 2008 al juny de 2013, amb un total de 41.188 contactes telefònics. Cada registre inclou la variable objectiu de sortida amb el resultat del contacte ({"fracàs", "èxit"}) i les característiques d'entrada dels candidats. Aquestes inclouen atributs de telemàrqueting (p. ex. el mitjà de comunicació), detalls del producte (p. ex. el tipus d'interès ofert) i informació del client (p. ex. l'edat). A més, la llista conté certa informació d'influència social i econòmica (p. ex. la taxa de variació de l'atur) extreta de la web del Banc Central de la República Portuguesa (aquí). La fusió d'ambdues fonts de dades constitueix un gran conjunt de característiques potencialment útils, amb un total de 21 atributs, que es revisen a continuació.

VARIABLES

Variables d'entrada

Dades dels clients

- *Age:* edat dels clients, enter.
- *Job:* tipus d'ocupació (categòriques amb 12 nivells: 'admin.', 'blue-collar', 'entrepreneur', 'housemaid', 'management', 'retired', 'self-employed', 'services', 'student', 'technician', 'unemployed', 'unknown').
- *Marital:* estat civil (categòrica amb 4 nivells: 'divorced', 'married', 'single', 'unknown'; amb: 'divorced' significant divorciat o viudo).
- *Education*: nivell d'educació (categòrica amb 8 nivells: 'basic.4y', 'basic.6y', 'basic.9y', 'high.school', 'illiterate', 'professional.course', 'university.degree', 'unknown').
- *Default*: considera si es té algún impagament (categòrica amb 3 nivells: 'no', 'yes', 'unknown').
- *Housing*: considera si es té un préstec hipotecari (categòrica amb 3 nivells: 'no', 'yes', 'unknown').
- *Loan*: considera si es té un préstec personal (categòrica de 3 nivells: 'no', 'yes', 'unknown').

Relacionades amb l'últim contacte

- *Contact*: tipus de contacte (categòrica de 2 nivells: 'cellular','telephone')
- *Month*: mes de l'últim contacte (categòrica de 12 nivells: 'jan', 'feb', 'mar', ..., 'nov', 'dec')
- *Day_of_week*: dia de l'últim contacte(categòrica de 5 nivells: 'mon','tue','wed','thu','fri')

- *Duration:* duració de l'últim contacte, en segons (numèrica). Aquest atribut afecta directament a la variable de sortida ja que si la duración és 0, no es contractarà el dipòsit. A més a més, aquesta variable no ens servirà per fer modelar ja que no la podem saber fins que s'ha afectuat la trucada.

Altres atributs

- *Campaign*: nombre de contactes efectuats durant la campanya de màrqueting (numèrica)
- *Pdays*: nombre de dies transcorreguts des de l'última vegada que el client va ser contactat en una altra campanya (numèrica)
- *Previous:* nombre de contactes efectuats abans de la campanya (numèrica)
- *Poutcome*: resultat de l'anterior campanya de màrqueting (categòrica de 3 nivells: 'failure','nonexistent','success')

Variables socials i econòmiques

- *Emp.var.rate*: taxa de variació trimestral de la població activa (numèrica)
- *Cons.price.idk*: índex del preu del consum mensual (numèrica)
- *Cons.conf.idx:* índex de confiança del consumidor, indicadora mensual (numèrica)
- *Euribor3m*: índex de "Euro Interbank Offered Rate", indicador mensual (numèrica)
- *Nr.employed*: nombre d'empleats, indicador trimestral (numèrica)

Variable resposta ("target")

- *Y*: el client se subscriu o no al dipòsit bancari (binària: 'yes','no')

PROCÉS D'EXPLORACIÓ DE DADES

Primerament, es considera la variable resposta. Pel que fa aquest atribut ('*y*'), s'observa que està molt desequilibrat: hi ha un 89% de clients que no soliciten el dipòsit i només un 11% que el contracta. A priori doncs, es pot dir que serà difícil obtenir un model de predicció molt bo.

Tot seguit, es prossegueix a analitzar les dades personals dels clients. L'edat ('***age***') mínima és de 17 anys i la màxima de 98 anys, és a dir, el rang de valors sembla ser raonable. Tanmateix, l'anàlisi dels *boxplots* reflecteix que l'edat no sembla ser una variable que influeixi en el resultat de la classificació. En la variable '***job***' s'observa que existeixen 330 persones de les quals no se sap la seva ocupació i es troben indexades amb el valor NA. En la comparació dels clients que contracten el dipòsit en funció de la seva feina, s'aprecia que el percentatge d'èxit és més alt en els grups 'retired', ja que un 25% dels jubilats contracten el dipòsit, i 'students', el 31% d'aquests també el contracten. En la base de dades hi ha 80 persones de les quals no es coneix el seu estat civil '***marital***' i es troben indexades amb el valor NA. En un principi, no s'observa cap diferència entre els estats civils pel que fa a la variable resposta. En quant a la variable '***education***', existeixen 1731 dades faltants indexades amb NA, la qual cosa podria ser un problema. Els clients amb estudis '*university degree*' i '*professional curse'* tenen un percentatge d'èxit més alt, d'un 13% i un 11% respectivament dintre de cada grup, tot i que la categoria 'illiterate' és la més exitosa, el 22% d'aquest grup contracten el dipòsit. Ara bé, es té una representació molt baixa com per extreure conclusions definitives. En la variable '***default***' existeix un gran nombre de dades faltants (8597) i únicament hi ha 3 de positives ('yes'). Aquestes tres persones que tenien algun tipus d'impagament no accedeixen al dipòsit. Es tenen molt poques dades per fer inferència i dir que les que tenen un impagament no contracten el dipòsit, no es poden treure conclusions concloents a partir d'aquestes. En els atributs '***housing***' i '***loan***' es comptabilitzen un total de 990 valors faltants NA en exactament les mateixes observacions i estan estretament correlacionades. Aparentment, no s'observen diferències significatives entre aquestes variables respecte a la contractació del dipòsit.

En canvi, la variable '***contact***' sí que sembla tenir una certa influència, ja que s'observa que el percentatge d'èxit de les trucades realitzades a telèfons mòbils és del 14,7%, gairebé el triple que les efectuades a fixes les

quals només tenen un 5% d'èxit. En la variable '***month***' no es troben tots els mesos: falten gener i febrer. Sembla ser que les trucades realitzades al desembre, març, octubre i setembre tenen un percentatge d'èxit més alt (48,9%, 50,5%, 43,9%, 44,9% respectivament) que les efectuades en altres mesos. En l'atribut '***day-of-week***' s'observa que les trucades es duen a terme en dies laborables i no en festius. No obstant això, no es troben diferències substancials pel que fa al percentatge d'èxit de les trucades respecte el dia de la setmana. Per últim, la variable '***duration***' no resulta massa útil considera-la per obtenir a un model predictiu real, ja que la duració de la trucada no se sap a priori, tot i que s'observa que aquesta sí que influencia molt la resposta.

En altres atributs, en '***campaign***', el nombre de vegades que es truca a un client es troba entre 1 i 10 principalment. Tot i així, es percep una cua molt llarga de persones a les quals per alguna raó han estat trucades moltíssimes vegades. En la variable '***pdays***', els valors marcats amb un 999 representen aquells clients que mai han estat contactats, els qual hi ha 39.673. No obstant això, les variables '***previous***' i '***poutcome***' indiquen que d'aquests, realment 35.563 no havien estat contactats. Per tant, és possible que la diferència de clients restants (4110) fossin persones les quals van estar contactades fa més de 999 dies (~2,7 anys). L'atribut '*poutcome*' influeix molt en el resultat de la variable resposta amb un 65% d'èxit en aquells clients que ja havien accedit a l'oferta.

Pel que fan les variables socials i econòmiques s'observa que estan altament correlades entre elles, per tant, es podria prescindir d'unes quantes i es guanyaria interpretabilitat. Les variables que es consideren treure són les '***nr.employed***' i la '***emp.var.rate***' ja que estan molt correlades amb la '***euribor3m***', concretament estan correlacionades un 90%, per tant, per modelar i simplificar es podria prescindir d'elles. Els models que es generen més endavant es proven amb totes les variables i sense les altament correlades per tal de veure com obtenir els millors resultats. Figura 1 i Figura 2.

PREPROCESSAMENT DE DADES

En aquest apartat es du a terme una transformació de les variables per tal de poder, en un futur, aplicar els diferents models a les dades.

En primer lloc, pel que fa la variable '***duration***', la que mostra la duració de la trucada, es considera treure-la de la base de dades ja que es veu que té una influencia important en la resposta però no serveix per modelar ja que aquesta no se sap a priori, no se sap fins que no s'ha efectuat la trucada, moment en què es coneix, empíricament, quin ha sigut el resultat.

En segon lloc, s'ha fet una categorització de la variable 'pdays'. S'ha observat que aquesta variable pren valors en [0,27]U{999}. S'observa que les entrades amb valor 999 tenen associat un nombre de vegades contactades prèviament més gran o igual a 0. Aleshores el que s'ha fet és categoritzar aquesta variable en tres nivells. El primer nivell correspon a les persones contactades en el darrer mes, concretament en els últims 27 dies i es troben 1515 persones en aquest grup; el segon, a les persones mai contactades on hi trobem 35563 individus i l'últim a persones contactades fa més de 2,7 anys on es troben 4110 individus.

IMPUTACIÓ

Tot seguit s'ha dut a terme una imputació a les dades mancants, és a dir, a les variables: 'default', 'education', 'housing', 'loan', 'job' i 'marital'. Aquesta imputació s'ha decidit fer amb l'ajuda de MICE (*Multivariate Imputation via Chained Equations*) de R. Aquest paquet assumeix que les dades mancants són MAR (*Missing at Random*), cosa que significa que la probabilitat que falta un valor depèn únicament en el valor observat.

Per defecte, s'utilitza regressió lineal per predir valors mancants continus i regressió logística per les variables categòriques. Un cop completat el cicle, es generen múltiples conjunts de dades que només difereixen dels valors imputats. Es considera que és una bona pràctica construir models sobre aquests conjunts per separat i escollir el més adequat.

PARTICIONAT DE LES DADES

Les dades originals es parteixen en dos conjunts, un de *re-train*, que conté aproximadament un 80% de les dades, 32.950 observacions, que s'utilitza per re-entrenar els models que han donat bons resultats i un de *final test* que conté el 20% restant, 8.238 individus, que s'utilitza per testejar els models bons.

Per entrenar els models no es dur a terme un *k-fold cross validation* ja que no és viable perquè es té un conjunt de dades molt desbalancejat. Si es portés a terme, caldria que es balancegessin les dades per a cada partició de *training*. Això implicaria generar copies de les dades per tal de tenir les dades originals (a balancejar) al següent pas i no haver d'equilibrar de nou les balancejades anteriorment.

Per evitar aquest problema es du a terme el *validation set approach.* El conjunt *re-train* està format per dos subconjunts, un de *trainig* que conté un 64% de les observacions i un de *testing* que conté un 16% dels individus. Aquests s'utilitzen per provar els diferents models. Les dades del *training* són les que cal balancejar. Una vegada escollits els millors models per les dades amb les que es tracta s'utilitza tot el conjunt *re-train* per entrenar de nou els models i el conjunt *final test* per testejar-los. Obtenint així un millor resultat dels models emprats. Per revalidar els resultats es podria repetir la partició diverses vegades, imitant, en certa manera, la tècnica de *cross validation.*

BALANCEJAMENT DE LES DADES

Les dades amb les que es tracta estan molt desequilibrades. Només 46640 observacions de les 41188 totals es classifiquen com a sí, és a dir, només un 11% dels usuaris contracten el servei.

La classificació no balancejada és un problema d'aprenentatge supervisat on una classe supera en gran mesura l'altra classe. El terme desequilibrat fa referència a la disparitat trobada en la variable dependent (*target*). Per tant, un problema de classificació desequilibrat és aquell en què la variable dependent depèn de la proporció de classes desequilibrada.

Els algoritmes ML tenen problemes de precisió a causa de la distribució desigual de la variable dependent cosa que provoca que el rendiment dels classificadors existents estigui esbiaixat cap a la classe majoritària ja que els algorismes són controlats amb precisió, és a dir, pretenen minimitzar l'error global al qual la classe minoritària contribueix molt poc. Els algorismes ML assumeixen que el conjunt de dades té distribucions de classes equilibrades. També assumeixen que els errors obtinguts de diferents classes tenen el mateix cost. Per tant, s'ha de fer front a aquest problema per tal de lluitar cintra la reducció de la precisió dels algorismes ML en conjunts desequilibrats. Per intentar equilibrar les dades s'han dut a terme cinc mètodes diferents (*oversampling, undersampling, both sampling, rose, smote*).

OVERSAMPLING

Aquest mètode s'utilitza per sobremostrejar la classe minoritària fins que arriba al mateix nombre d'observacions que la classe majoritària. Amb aquesta tècnica s'aconsegueix balancejar les dades obtenint 29239 respostes afirmatives i 29239 respostes negatives.

UNDERSAMPLING

Prenem menys mostres de la classe majoritària sense substituir-la aconseguint balancejar les dades obtenint 3712 respostes afirmatives i 3712 negatives. Com que no es tenen observacions de la classe 'sí' per a la variable 'default' cosa que portaria a errors en la regressió logística i en altres algorismes, de manera que s'afegeix una observació al conjunt de dades.

BOTH SAMPLING

Aquest mètode és una combinació del *undersampling* i el *oversampling*. La classe majoritària és *undersampled* sense reemplaçament i la minoritària és *oversampled* amb reemplaçament.

ROSE

Aquest mètode genera dades sintèticament i proveeix una millor estimació de les dades originals.

SMOTE (Synthetic Minority Over-Sampling Technique)

La tècnica de SMOTE s'utilitza per evitar el sobreajust en afegir rèpliques exactes de casos minoritaris al conjunt de dades principal.

Amb tots els mètodes proposats anteriorment s'aconsegueix balancejar la variable resposta. En els mètodes de classificació que s'utilitzaran posteriorment es faran servir totes les tècniques proposades i s'acabarà considerant, en cada cas, la que proporcioni millors resultats.

TRANSFORMACIÓ VARIABLES CATEGÒRIQUES

Per fer diferents anàlisis cal que totes les variables siguin numèriques, però en el cas de la base de dades amb la que es tracta, més de la meitat són categòriques. Així doncs, cal fer-ne una transformació per poder utilitzar certs models. Una de les tècniques més usades i la que s'utilitza en aquest projecte és la transformació de les variables categòriques en *dummies* ja que són les que utilitzen la majoria de les funcions ja implementades com LDA{MASS}, QDA{MASS} o SVM{e1071}. Aquestes, entre d'altres, ja contemplen la possibilitat de rebre paràmetres factors i tractar internament la transformació a *dummies*.

Hi ha alguns mètodes que s'implementen com el KNN o SVM, entre d'altres, on cal escalar les variables per tal de comparar les seves magnituds, la majoria d'ells però ja ho fan per defecte.

CLASSIFICACIÓ

CLUSTER

En aquesta secció es realitza un anàlisis clúster al conjunt de dades, per tal d'intentar agrupar-les seguint un cert criteri de similitud, i poder extreure alguna conclusió de cara a la distribució de la variable resposta respecte a aquests grups.

La base de dades està formada per variables numèriques i categòriques, per tant no pes pot dur a terme un anàlisis clúster clàssic com el famós *k-means*, ja que aquest té en compte les distàncies Euclidianes entre observacions. S'ha de prendre algun algorisme més robust que *k-means* en el sentit de que es pugui tractar amb variables no només de tipus numèric. Una distància que pot tractar amb tot tipus de variables és la distància de Gower, que es calcula com la mitjana de les dissimilituds parcials entre individus. Aquesta distància, encaixa molt bé amb l'algorisme *k-medoids*, que és una clàssica tècnica de partició de clustering que separa les dades en k clústers diferents (k com a hiperparàmetre).

Un problema d'aquest algorisme és que és molt costós a nivell de temps i a nivell de computació de CPU, ja que el temps d'execució i el cost en memòria són quadràtics. Per tant, com que es tenen bastantes dades i limitacions computacionals, es pren una mostra aleatòria de les dades (aproximadament del 20% de les nostres dades), per tal de realitzar l' anàlisis. Després de realitzar vàries proves, amb diferents conjunts aleatoris per comprovar que tots donen el mateix, s'obtenen els resultats exposats a continuació.

Per tal de poder determinar quin és el nombre de clústers adequat per l'anàlisi, s'ha repetit l'anàlisi per *k = 2,...,10*, i s'han pres els millor representats per la corba de *'Silhouette'*, que és un índex que contrasta la distància mitjana dels elements en un mateix clúster entre la distància a elements d'altres clústers. Es prova de fer l' anàlisi amb *k = 2, 4 i 8*, que són els pics de la Figura 3 de l'annex.

Per tal d'interpretar els resultats, bàsicament hi ha dues maneres: observar el *summary* de cada un dels clústers o visualitzar-los.

Per visualitzar els resultats s'utilitza una tècnica anomenada '*T-distributed Stochastic Neighbor Embedding' (t-SNE)*, que és un algorisme d'aprenentatge automàtic que s'utilitza per poder visualitzar dades d'un espai dimensional gran amb només 2 o 3 dimensions. Més específicament, modela cada observació com a un punt 2-3 dimensional de tal manera que les observacions similars estan representades per punts propers, i les observacions diferents per punts llunyans amb una probabilitat alta. Figura 4. Figura 5. Figura 6.

Els *summaries* dels clústers es poden consultar a l'annex, però en particular interessa com de bé està distribuïda la variable resposta en els diferents grups. Es pot veure aquest contrast utilitzant taules de contingència, i en tots els anàlisis (per *k = 2,4,8*) el percentatge d'observacions classificades com a 'yes' de la nostra base de dades és molt més gran per uns grups en concret. En la Taula 1, Taula 2 es mostra per exemple, per k = 4.

Es pot observar com clarament en els clústers 3 i 4 és molt més probable que una observació sigui de la classe 'yes'. Per tal d'interpretar millor aquest resultat, es pot intentar representar les diferències d'aquests clústers entre sí per a alguna variable, com per exemple l'euríbor o el nombre d'empleats com es mostra en la Figura 7. D'aquests dos gràfics es pot deduir que és molt més probable que un individu contracti el servei bancari si el dia en el qual van contactar amb ell telefònicament hi havia un euríbor baix i el nr.employed també era més baix.

Curiosament, i com a fet a destacar, si es prova de modelar dos arbres de decisió (CART trees) que només tinguin en compte aquestes variables es veuen representats els fets mencionats anteriorment. Efectivament, es pot veure en la Figura 8 i Figura 9 que si l'euribor és alt, és molt més probable que l'individu no contracti el servei i viceversa. També a partir d'aquests, es pot veure com és important que el nr.employed sigui baix per tal de maximitzar la probabilitat de que l'individu contracti el servei ofert per la campanya publicitària.

KNN (k-nearest neighbours)

KNN és un mètode que funciona per distàncies euclidianes. Com que s'està treballant amb factors aquests s'han hagut de binaritzar prèviament. Així doncs, a priori sembla que no té gaire sentit dur a terme aquest anàlisis. No obstant, s'ha dut a terme el KNN i els resultats obtinguts es mostren a la Taula 3.

El millor rendiment a partir de KNN s'obté quan s'usa el conjunt d'entrenament *undersampling* amb 20 veïns, ja que l'equilibri entre ambdues *accuracies* és força adequat. En el millor cas s'obté un 80.57% de la *total accuracy,* un 64.29% *de la positive* accuracy i 1015 true negatives. Cal destacar també que, aquest model no es considerarà com a model pertinent pel tipus d'anàlisi sobre les dades que hi ha disponibles.

DISCRIMINANT ANALYSIS

En dur a terme l'anàlisi discriminant es troben diferents problemes. Per una banda, hi ha problemes de multicol·linealitat en el LDA i en el QDA on es produeix una deficiència de rang a causa de convertir les variables en *dummies*. Aquest problema no es troba en el RDA ja que aquest mètode és més estable en aquest sentit. Per altra banda, el fet de poder realitzar o no el LDA i QDA depèn de la partició inicial de les variables.

LDA (Linear Discriminant Analysis)

El LDA té com a assumpció fonamental que les variables independents segueixen una distribució normal multivariada. A priori sembla ser que no té gaire sentit fer aquesta assumpció en les dades amb les que es treballa ja que algunes d'aquestes s'han convertit a *dummies*, variables binàries que indiquen la presència o absència d'algun efecte categòric i que no segueixen cap distribució. Això es trasllada a que les variables que són categòriques no es tracten òptimament amb un anàlisi lineal discriminant. Així doncs, semblaria millor utilitzar *logistic regression*, mètode que no fa cap assumpció de la distribució de variables. No obstant, es du a terme el LDA.

Després de realitzar vàries proves, s'ha obtingut una partició que permet dur a terme el LDA. Aquesta proporciona resultats molt similars als obtinguts amb els altres mètodes.

QDA (Quadratic Discriminant Analysis)

A priori, aquest mètode tampoc té sentit aplicar-lo per les mateixes raons que s'han donat en el LDA. A diferència del primer, aquest no assumeix igualtat de la matriu de covariàncies però assumeix que les variables provenen d'una normal multivariada.

El QDA no s'ha pogut dur a terme ja que dóna problemes de deficiència de rang, se n'esperaven uns resultats molt similars al LDA però.

RDA (Regularization Discriminant Analysis)

El RDA és una generalització del LDA i el QDA, du a terme una regla de classificació fent servir matrius de covariància de grups regularitzats cosa que aporta més estabilitat i robustesa contra els problemes de multicol·linealitat però en el cas de les dades amb les que es treballa, teòricament, continua sense tenir sentit per les seves assumpcions.

Efectivament, sense balancejar les dades s'obté una *total accuray* alta però s'obté un *positive rate* molt baix, per tant, aquests resultats no interessen. Una vegada equilibrades les dades s'obté un millor *positive rate* però es perd *total accuracy*. Els mètodes de balancejament de dades que proporcionen millors resultats són: *ROSE, Undersampling, Oversampling i Both* (en aquest ordre) com es pot veure en la Taula 4.

El LDA és millor en el sentit que classifica millor en àmbit general, comet menys errors. El RDA per aconseguir una millora del 9% repecte el LDA comet el doble d'errors, s'efectuen el doble de trucades *true negative*, però s'obté una millor classificació de la classe desitjada. Per tant, s'obtenen més clients.

REGRESSIÓ LOGÍSTICA (GLM)

La regressió logística no fa cap assumpció de normalitat de cap tipus, ni pels predictors ni per la variable resposta; és un model de probabilitats directe i no requereix usar la regla de Bayes per convertir els resultats a probabilitats com fa el LDA. Així doncs, a priori sembla que té més sentit fer un fit amb aquest model que amb els vistos fins ara.

Després de fer varis experiments amb els diferents mètodes de balancejament de dades, com es pot veure en la Taula 5, i jugant amb les *priors*, s'observa que els diversos mètodes obtenen resultats molt similars, no obstant, el mètode de *both* sembla ser el que proporciona els millors. Aquest dóna un 81.12% de la *total accuracy*, un 65.90% de *true accuracy* i 991 *true negatives*.

NAIVE BAYES CLASSIFIER

El *naive Bayes classifier* és una tècnica per construir classificadors que assumeix independència entre les diferents variables. Així doncs, a priori, sembla que no té gaire sentit dur a terme aquest anàlisis directament amb la base de dades ja que les variables de les quals es disposen no estan incorrelades, no són independents entre elles.

No obstant, es du a terme l'anàlisi amb totes les variables i com era d'esperar, segons les proves realitzades, s'aprecia que *l'accuracy total* és, en general, lleugerament més baixa en comparació amb altres models utilitzats. Aquest fet es manifesta en la distribució de cadascuna de les *accuracies* que es troba equilibrada en pràcticament tots els conjunts d'entrenament. No obstant això, dels models provats, es considera el millor el del conjunt de training original (tot i no ser realment massa bo i tal i com es pot veure en la Taula 6 de l'annex): 81.99 *d'accuracy total*, 60.78 de positiva i 896 *true negatives*.

A més a més, s'ha provat de dur a terme, per solucionar el problema de la no independència entre variables, aquesta tècnica sense les variables altament correlades. Ara bé, els resultats segueixen sent dolents comparat amb els dels altres models, no s'obtenen gaires millores.

DECISION TREES

En utilitzar els decision trees s'obtenen resultats similars en diversos conjunts d'entrenament. En particular, tant en el model amb priors, com en els de oversampling, undersampling i both s'aconsegueixen els mateixos valors i són els que ofereixen *l'accuracy positiva* més alta amb un 61.73, mantenint a la vegada una bona *accuracy total* del 84.09 i obtenint 764 *true negatives*. No obstant això, s'escull el conjunt d'undersampling, el qual manté un bon ajust i ha mostrat un bon rendiment en models emprats anteriorment. Taula 7.

RANDOM FORESTS

En els *random forests* s'utilitza els cinc conjunt de dades balancejades i el conjunt de *training* original. Cal destacar que, en el cas del conjunt original, es proven diferents plantejaments: D'una banda, es juga amb les probabilitats a priori i amb la creació d'un nombre d'arbres més gran (1000 *trees*). D'altra banda, es modifica el nombre de variables disponibles per dividir en cada node de l'arbre (*mtry* = 15) i la mida mínima del node que defineix implícitament la profunditat dels arbres (*nodesize* = 60).

Segons els resultats obtinguts, es considera el conjunt de training balancejat amb undersampling (un dels que millor resultat dona amb altres models com es pot veure en la Taula 8 de l'annex) el més adequat per a les dades, ja que s'aconsegueix un bon equilibri amb un 81.45 d'accuracy total i un 65.63 de positive i 967 *true negatives*.

ANN (Artificial Neural Network)

En primer lloc, la funció 'nnet' fa la transformació entre les variables categòriques (*factor*) a variables *dummy* internament i no cal tenir present cap modificació. En les NN s'utilitza el conjunt de *training* original i els cinc conjunts de dades d'entrenament que s'han creat per intentar balancejar les dades.

Per tal de trobar la millor arquitectura de xarxa, es poden considerar dos enfocaments: a) explorar diferents nombres d'unitats ocultes en una capa oculta, sense regularització b) fixar un nombre H d'unitats ocultes en una capa oculta i explorar diferents valors de regularització (recomanat i utilitzat aquí). Cal destacar que realitzar ambdós alhora sol ser una pèrdua de recursos computacionals.

Nombre de neurones ocultes: s'estableix el nombre de nodes ocults utilitzant la heurística $H = round(M/2)$ (on M és el nombre d'entrades) esmentada en l'informe. Realment no es coneix quantes entrades *M* crea la NN per a les dades (a causa de la implementació interna de variables *dummy*).

Valors de regularització: s'estableixen set valors de regularització (de 0.001 a 1) per provar sense utilitzar cap mètode de remostreig CV i el millor s'utilitza per construir un model final.

Tal i com es preveia, amb el conjunt de *training* original no s'obtenen bons resultats, ja que en tots els casos les *accuracies* positives són molt baixes. Cal destacar que, en general, el valor de regularització 1 és el que ha generat millors rendiments. A més, en els altres conjunts d'entrenament es troben certes similituds. Tant en *oversampling* com en *both* s'obtenen *accuracies* semblants per a cadascun dels valors de regularització amb valors al voltant de 80 per *accuracy* total i 62 per la positiva. Anàlogament, en els conjunts de *smote i rose* els resultats són similars, encara que més baixos, amb un 80 de total i un 50 de positiva. Per tant, el millor conjunt esdevé de nou el *undersampling* amb uns resultats molt bons usant el *decay* 1: 82.74 d'*accuracy* total i 64.82 de positiva amb 876 *true negatives*. Taula 9.

SVM (Support Vector Machine)

Per modelar amb Support Vector Machines s'utilitza la funció ksvm{ksvm} de R que permet provar amb vàries funcions de kernel diferents i jugar amb tots els seus paràmetres. Després de realitzar vàries proves amb tots els models s'ha arribat a la conclusió que els que millor s'adapten a les dades són el kernel radial Gaussià i el Laplacià, mentre que els altres, ja sigui el lineal, el polinòmic o d'altres, obtenen resultats bastant pobres.

La funció ksvm és especialment bona per a aquests dos tipus de kernel mencionats anteriorment, ja que per defecte implementa una heurística que determina per si sol el valor del paràmetre sigma necessari. El mètode proporciona uns resultats molt similars als obtinguts amb altres mètodes, tot i que al ser un mètode d'optimització convex d'ordre superior a quadràtic, té uns temps d'entrenament bastant elevat per a aquelles bases de dades amb més observacions.

Com es pot veure en la Taula 10 de l'annex, com que s'ha decidit prendre una *total accuracy* major del 80%, el model de SVM proposat és el model amb kernel RBF i C=1 que aconsegueix un 83.70% de *total accuracy*, un 63.07 de *positive accuracy* i 800 *true negatives.*

CLASSIFICACIÓ AMB *FEATURE SELECTION*

A partir del *Random forest* s'implementa una funció per mirar la importància de cada variable per tal de poder escollir les més rellevants. Aquesta, principalment es basa en l'índex de *Gini*. Una vegada implementada, es fa un estudi de les que tenen un *MeanDecreaseGini* més alt i es combinen les set primeres (es poden veure en la Figura 10 de l'annex) exhaustivament fins obtenir la combinació de variables més rellevant. En aquest cas, formada per l'indicador mensual, *euríbor,* i l'indicador trimestral de nombre d'empleats*, nr.employed*.

Es duen a terme, de nou, els algorismes d'aprenentatge automàtic exposats anteriorment. En casos com NN, KNN, Bayes... donen resultats molt similars però lleugerament inferiors. Ara bé, a l'aplicar *Random forests* i *SVM* s'obtenen uns resultats millors, com es pot observar en la Taula 11 i Taula 12 de l'annex. És a dir, amb dues variables s'aconsegueixen resultats molt similars i en alguns casos inclòs millors que a l'utilitzar totes les variables. El millor cas és el del *Random Forest* amb *undersampling* i les dues variables més importants en el qual s'aconsegueix quasi un 80% *de total accuracy i la positive accuracy* més alta obtinguda amb aquest percentatge de total, un 69.41% amb 1120 *true negatives*.

VALIDACIÓ DELS MILLORS MODELS

En aquesta part, es dur a terme la segona part del *re-train* explicada en l'apartat de particionat de les dades.

Primer de tot, es fa un recopilatori dels millors models seleccionats a cada mètode fins ara els quals es poden observar a la taula 13. Una vegada seleccionats, són re-entrenats amb la partició *re-train* del 80% de les dades originals. Seguidament, i com s'ha mencionat anteriorment, es validen amb el conjunt de validació del 20% del total. Aquests resultats es mostren en la taula 14.

Comparant la taula dels models entrenats amb un 64% del total de les dades i dels models entrenats amb el 80% es pot observar que hi ha fluctuacions petites, en algun cas de fins un 4%. Aquestes poder ser degudes a la component aleatòria que té seleccionar particions aleatòries de dades. Per tal de consolidar els resultats obtinguts es podria repetir el mateix experiment per diferents particions de dades, per varis conjunts de validació final i finalment promitjar tots els experiments per així obtenir una estimació més aproximada als valors reals.

CONCLUSIONS

Durant el transcurs d'aquest projecte s'han trobat diferents adversitats. Principalment, en la importància del preprocessat de les dades entenent com a pre-processat de les dades la imputació de les dades mancants, el balancejament de les dades i per últim el que es titula *feature selection o feature engineering.*

La selecció del millor model dependrà totalment dels interessos del banc en qüestió. Per poder-ho determinat correctament s'hauria de saber què li reporta en beneficis a un banc obtenir un client nou i què li costa a aquest fer una trucada a un possible client obtenint una negativa. Si el fet de realitzar trucades que no acaben amb un client nou no li suporta al banc una pèrdua significativa potser interessaria prendre un model que prioritzés més la *true accuracy* i no tingués tant en compte la *total accuracy* ja que el creixement disparat de *true negatives* no importaria. En canvi, si al banc li suposa moltes pèrdues realitzar moltes trucades no satisfactòries es prioritzaria al màxim mantenir el balanç, l'equilibri, entre les dues *accuracies.*

Un banc podria acabar trucant a tots els clients potencials obtenint el 100% de la *true accuracy* però una *total accuracy* molt baixa, és a dir, podria fer infinites trucades obtenint així tots els clients possibles. Ara bé, en aquest projecte aquest no és l'objectiu. En aquest, s'ha prioritzat en tot moment mantenir l'equilibri entre les *accuracies*, intentat obtenir el màxim de precisió en la predicció de la classe desitjada sense perdre gaire precisió global, és a dir, tenint en tot moment una *total accuracy* per sobre del 80%.

Seguidament, fent una valoració global de tots els algorismes d'aprenentatge automàtic utilitzats en aquest projecte, es pot descartar clarament que els millors models siguin els que funcionen sota l'assumpció de normalitat de les dades com els de *Discriminant Analysis, tant LDA com QDA i RDA,* o sota l'assumpció d'independència de les variables com el *Naive Bayes Analysis* o sota les distàncies euclidianes com en el cas del KNN.

Els resultats obtinguts i plantejats a continuació poden variar degut a la component aleatòria afegida al realitzar la imputació de les dades mancants amb un algorisme pròpiament aleatori per ser, el *random forest*. No s'observa que cap model destaqui significativament sobre cap dels altres, sense assumpcions, en tots els experiments. Així doncs, se seleccionarien principalment: *random forest, SVM* i *NN.*

Després de provar tots els models observem que s'obté com a molt al voltant del 65-67% de precisió en la predicció de la classe desitjada. Per poder millorar aquestes xifres, una opció seria demanar més dades, o la inclusió d'alguna variable que no està present en la base de dades actual, que potser ens pogués ajudar a explicar més satisfactòriament aquesta variabilitat de les dades que s'és capaç de modelar.

WEBGRAFIA

- **R-bloggers** (29 novembre 2018). *Linear, Quadratic, and Regularized Discriminant Analysis.* Recuperat des de: https://www.r-bloggers.com/linear-quadratic-and-regularized-discriminant-analysis/
- **Tormod Næs and Bjørn-Helge Mevik** (2001). *Understanding the collinearity problem in regression and discriminant analysis, university of Oslo, Blindern, Oslo, Norway*. Recuperat des de: http://mevik.net/work/publications/understanding_collinearity.pdf
- **Analytics Vidhya** (4 març 2016). *Tutorial on 5 Powerful R Packages used for imputing missing values*. Recuperat des de: https://www.analyticsvidhya.com/blog/2016/03/tutorial-powerful-packages-imputing-missing-values/
- **Analytics Vidhya** (19 agost 2018). *A Guide to Machine Learning in R for Begginers: logistic regression*. Recuperat des de: https://medium.com/analytics-vidhya/a-guide-to-machine-learning-in-r-for-beginners-part-5-4c00f2366b90
- **RDocumentation** (2015). *Classification and Regression Training (caret)*. Recuperat des de: https://www.rdocumentation.org/packages/caret/versions/6.0-84
- **RDocumentation** (2014). Create Elegant Data Visaulisations Using the Grammar of Graphics (ggplot2). Recueprat des de: https://www.rdocumentation.org/packages/ggplot2/versions/3.2.0

BIBLIOGRAFIA

- Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani (2013). *An Introduction to Statistical Learning with Applications in R*. New York, USA. Springer.E.
- Alpaydin (2010). *Introduction to Machine Learning*. MIT, USA. The MIT press.
- S. Moro, P. Cortez, P. Rita (2013). *A data-driven approach to predict the succes of bank telemarketing*. http://media.salford-systems.com/video/tutorial/2015/targeted_marketing.pdf

ANNEX I: TAULES DE RESULTATS I GRÀFICS

EXPLORACIÓ DADES

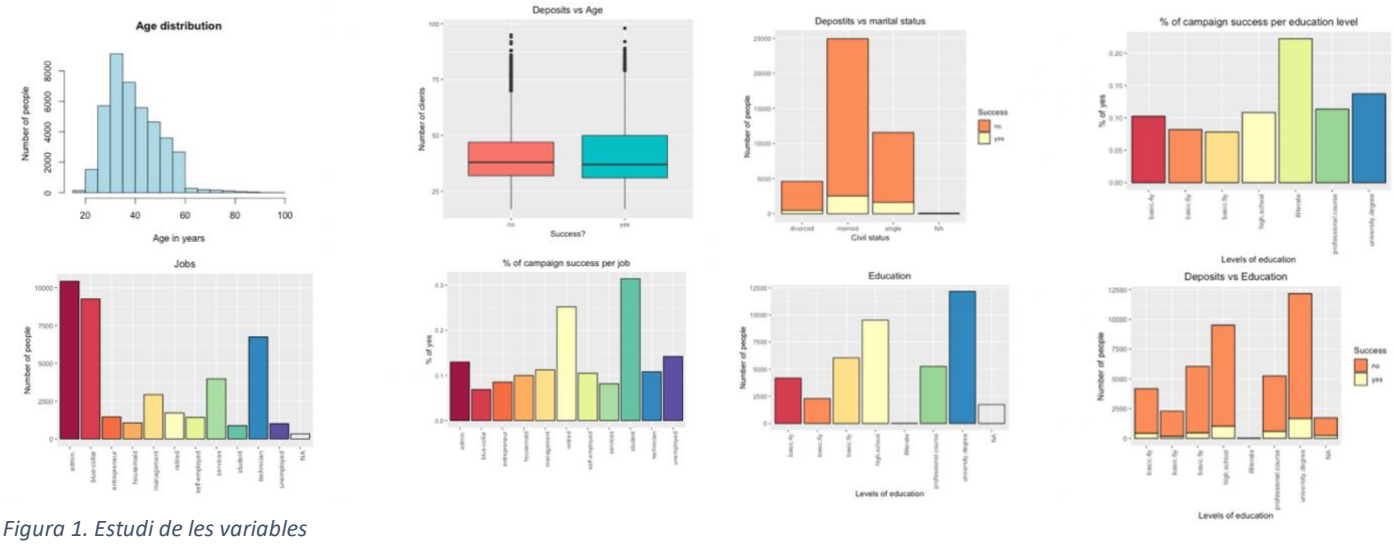


Figura 1. Estudi de les variables

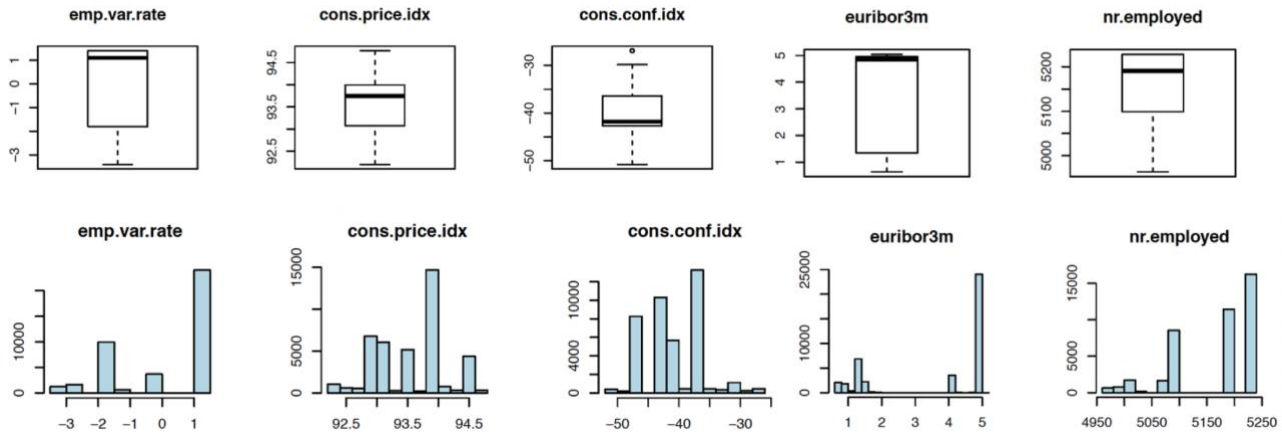


Figura 2. Estudi de les variables

CLUSTER

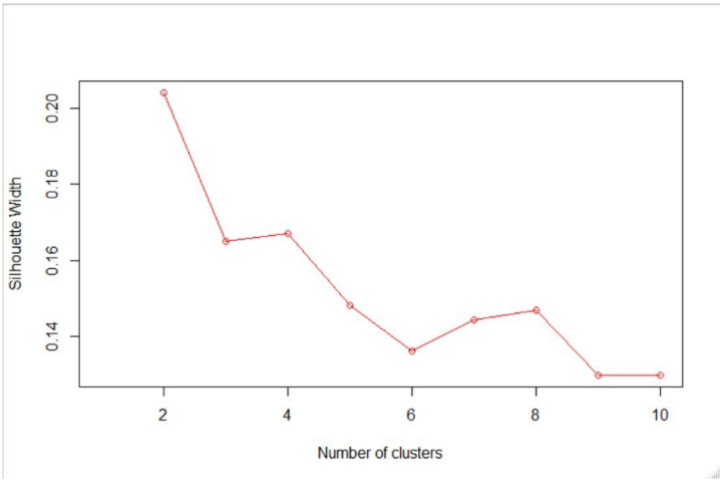


Figura 3. Número de clusters

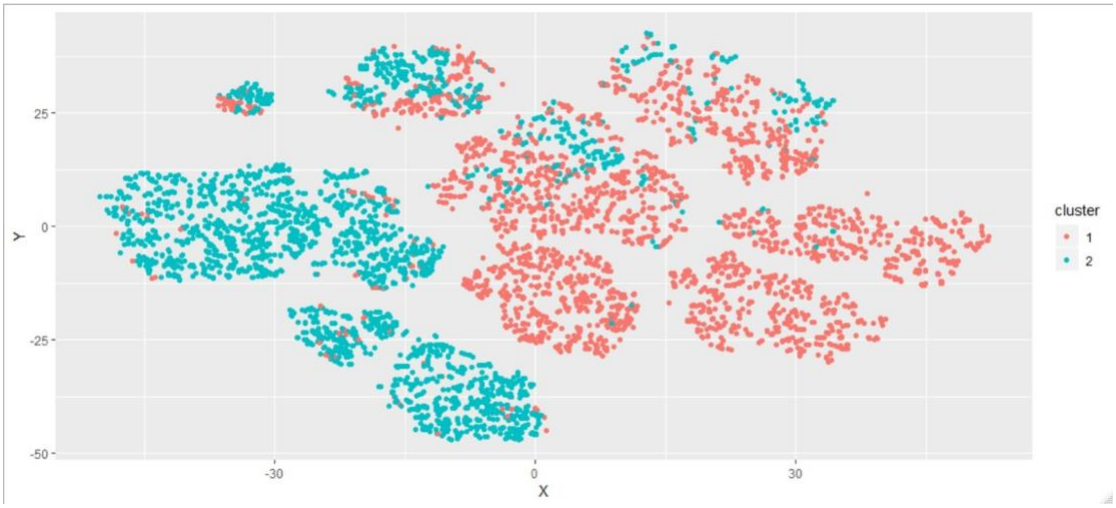


Figura 4. Clustering amb 2 classes en 2D

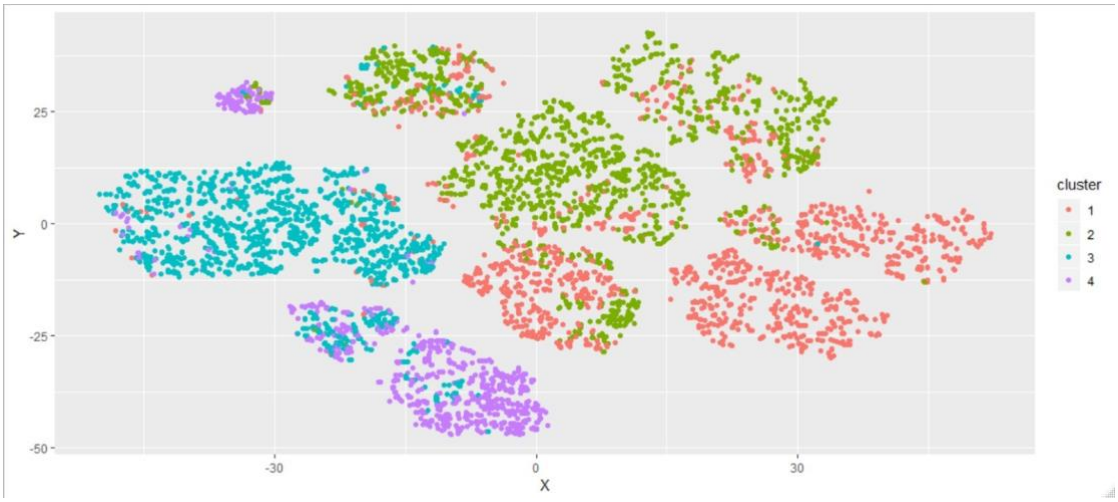


Figura 5. Clustering amb 4 classes en 2D

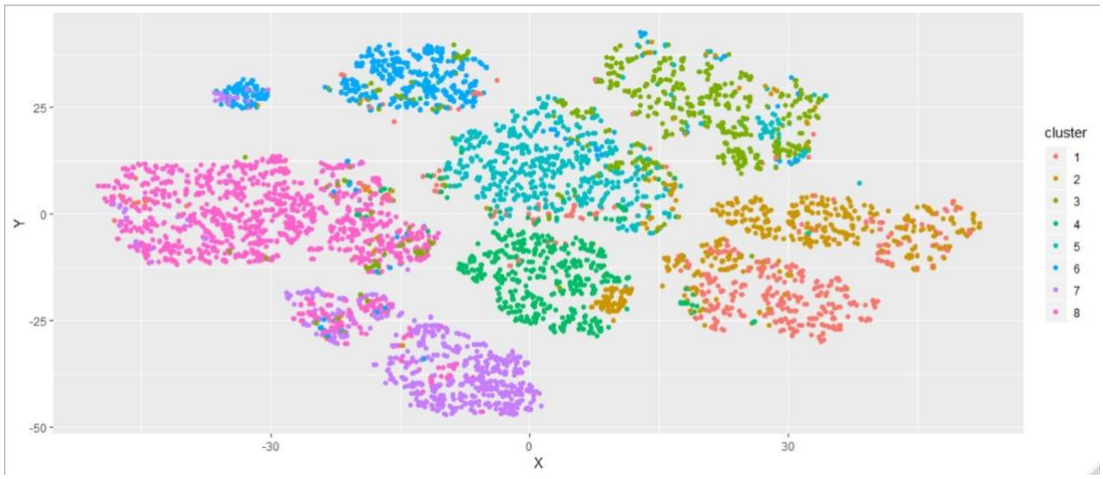


Figura 6. Clustering amb 8 classes en 2D

	NO	YES
1	2449	124
2	2621	176
3	1492	428
4	748	200

Taula 1. Clusters k=4

1	2	3	4
4.819	6.292	22.292	21.097

Taula 2. Clusters k=4

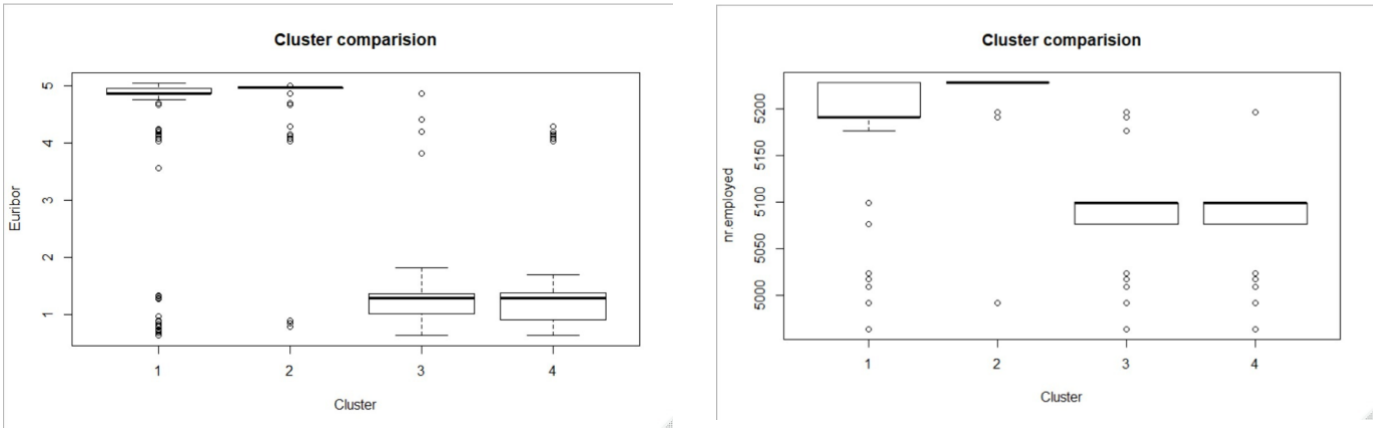


Figura 7. Comparació clusters

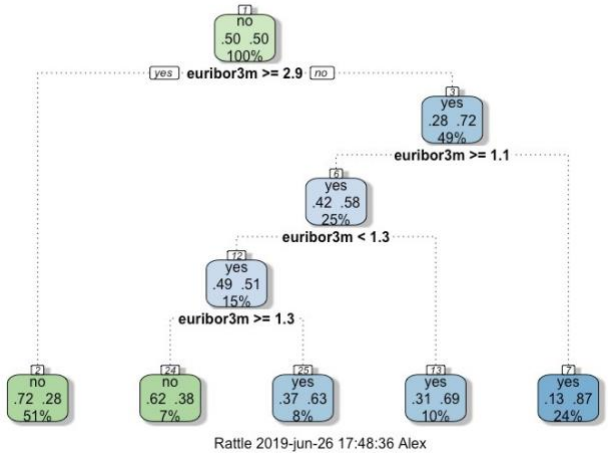


Figura 8. Decision trees Euribor

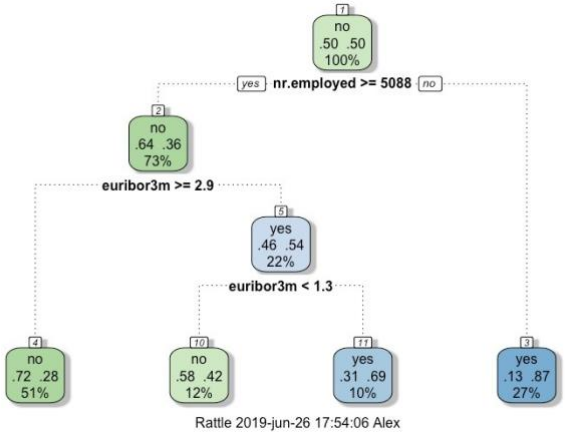


Figura 9. Decision trees employed

TAULA DE RESULTATS DE KNN

		KNN		
	K	**Total**	**Positive**	**True negative**
(·)	20	88.10	27.22	244
	10	89.35	23.85	137
	3	89.63	23.18	113
OVERSAMPLING	3	77.84	55.26	1128
	10	64.23	70.22	2136
UNDERSAMPLING	3	70.09	65.63	1716
	10	76.32	66.04	1308
	20	80.57	64.29	1015
BOTH	3	69.71	64.42	1732
	10	67.32	66.31	1903
ROSE	3	78.08	52.02	1088
	10	75.84	60.38	1298
SMOTE	3	78.08	52.02	1088
	10	75.72	60.51	1307

Taula 3. Resultats KNN

TAULA DE RESULTATS DEL DISCRIMINANT ANALYSIS

	LDA			RDA		
	Total	**Positive**	**True negative**	**Total**	**Positive**	**True negative**
(·)	89.09	36.66	249	89.30	19.68	109
(·)PRIOR	86.84	59.30	565	73.44	72.37	1545
OVERSAMPLING	83.38	63.21	822	Amb tots els mètodes de balancejament de dades obtenim els mateixos resultats que amb l'opció prior de la funció		
UNDERSAMPLING	82.47	64.82	894			
BOTH	83.58	63.07	808			
ROSE	82.68	64.66	878			
SMOTE	75.67	49.87	1231			

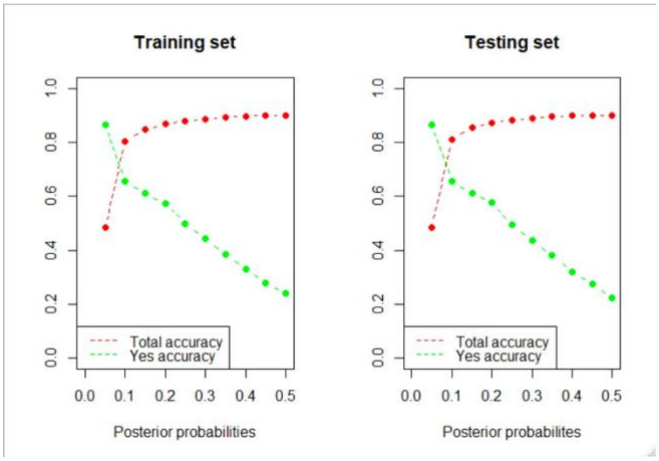
Taula 4. Resultats analisi discriminant

REGRESSIÓ LOGÍSTICA

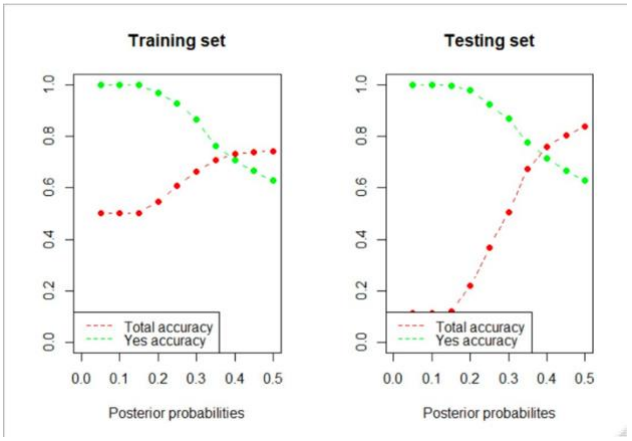
En aquest apartat, es poden veure les taules dels resultats, a l'esquerra, i a la dreta es mostren els gràfics dels resultat de modificar les *priors* fins a 0.5 amb passos de 0.05 per les dades no balancejades i les dades balancejades pels diferents mètodes.

Taula 5. Resultats GLM

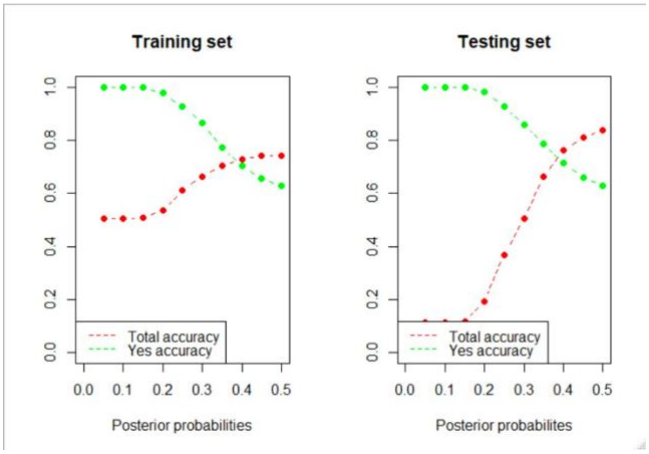
	REGRESSIÓ LOGÍSTICA		
	(·)		
POSTERIORS	**Total**	**Positive**	**True negative**
0.50	89.88	22.23	90
0.45	90.01	27.35	119
0.40	89.88	32.07	163
0.35	89.48	38.01	233
0.30	88.88	43.53	314
0.25	88.27	49.32	397
0.20	87.36	57.55	518
0.15	85.63	61.05	664
0.10	81.17	65.50	985
0.05	48.59	86.52	3288



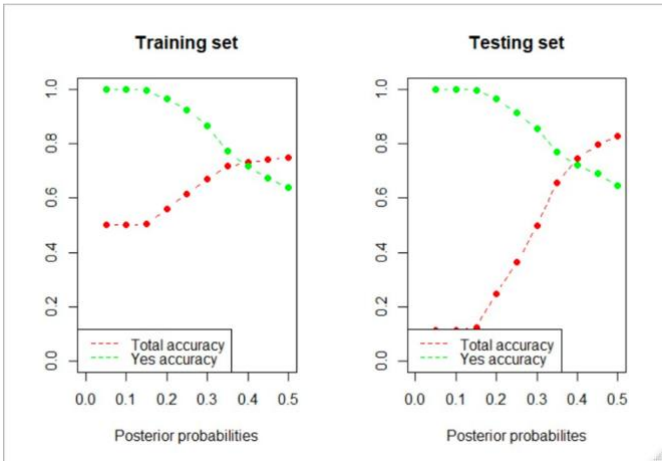
	REGRESSIÓ LOGÍSTICA		
	OVERSAMPLING		
POSTERIORS	**Total**	**Positive**	**True negative**
0.50	83.88	62.80	786
0.45	80.35	66.71	1048
0.40	76.05	71.56	1367
0.35	67.23	77.49	1992
0.30	50.49	86.79	3164
0.25	36.60	92.31	4120
0.20	22.09	97.98	5118
0.15	11.88	99.73	5804
0.10	11.27	100.00	5846
0.05	11.27	100.00	5846



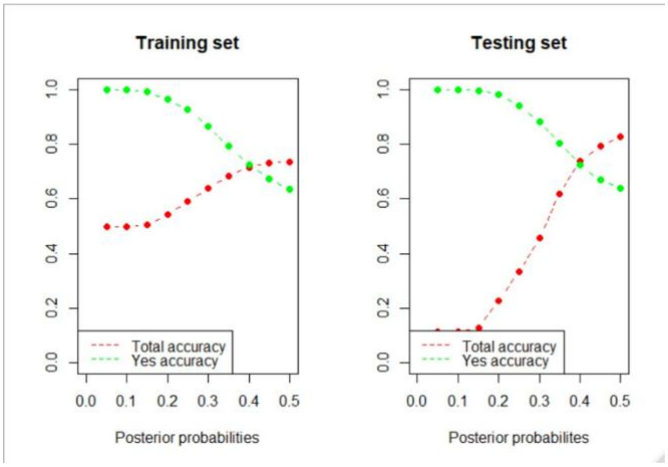
	REGRESSIÓ LOGÍSTICA		
	UNDERSAMPLING		
POSTERIORS	**Total**	**Positive**	**True negative**
0.50	82.92	64.69	863
0.45	79.53	68.87	1118
0.40	74.63	71.97	1463
0.35	65.73	76.81	2086
0.30	49.89	85.58	3195
0.25	36.33	91.37	4131
0.20	24.70	96.50	4935
0.15	12.32	99.73	5775
0.10	11.28	100.00	5846
0.05	11.28	100.00	5846



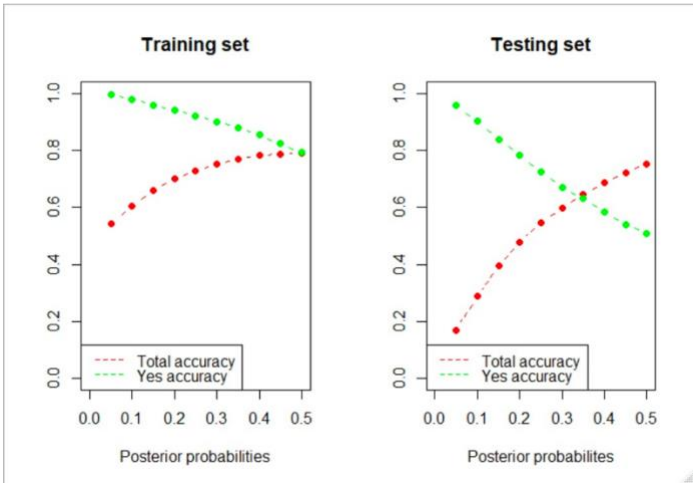
	REGRESSIÓ LOGÍSTICA		
	BOTH		
POSTERIORS	**Total**	**Positive**	**True negative**
0.50	83.72	62.94	798
0.45	81.12	65.90	991
0.40	76.15	71.29	1358
0.35	66.23	78.71	2067
0.30	50.60	85.71	3149
0.25	36.88	92.72	4105
0.20	19.32	98.38	5304
0.15	11.79	99.87	5811
0.10	11.28	100.00	5846
0.05	11.28	100.00	5846



	REGRESSIÓ LOGÍSTICA		
	ROSE		
POSTERIORS	**Total**	**Positive**	**True negative**
0.50	83.82	64.02	865
0.45	79.41	66.85	1111
0.40	73.79	72.51	1523
0.35	61.94	80.46	2363
0.30	45.55	88.41	3502
0.25	33.38	94.20	4347
0.20	22.63	98.38	5086
0.15	12.66	99.73	5753
0.10	11.31	100.00	5844
0.05	11.28	100.00	5846



	REGRESSIÓ LOGÍSTICA		
	SMOTE		
POSTERIORS	**Total**	**Positive**	**True negative**
0.50	75.37	50.81	1258
0.45	72.23	54.04	1489
0.40	68.66	58.49	1757
0.35	64.55	63.07	2062
0.30	59.64	67.12	2062
0.25	54.73	72.51	2779
0.20	47.85	78.30	3275
0.15	39.60	83.96	3861
0.10	28.67	90.16	4627
0.05	16.98	95.82	5439



TAULA DE RESULTATS DE BAYES

	BAYES		
	Total	**Positive**	**True negative**
(·)	81.99	60.78	896
(·)PRIOR	74.40	71.43	1475
OVERSAMPLING	74.43	71.43	1473
UNDERSAMPLING	74.49	71.43	1469
BOTH	74.72	71.43	1474
ROSE	76.95	68.73	1287
SMOTE	72.71	61.19	1510

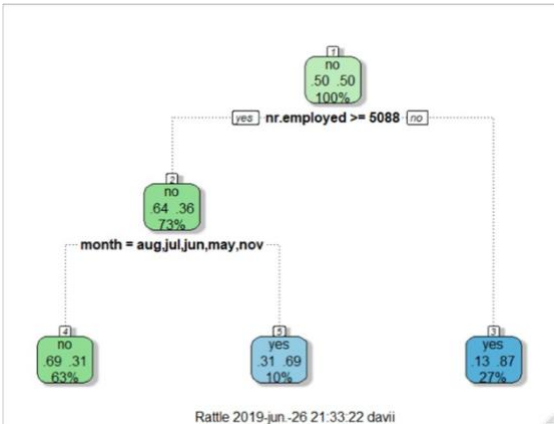
Taula 6. Resultats Bayes

TAULA DE RESULTATS DE DECISION TREES

	DECISION TREES		
	Total	**Positive**	**True negative**
(·)	89.82	18.46	66
(·)PRIOR	84.09	61.73	764
(·)LOSS	87.89	55.93	471
(·)LOSS+PRIOR	73.44	72.37	1545
OVERSAMPLING	Mateixos resultats que els obtinguts amb (·) prior		
UNDERSAMPLING			
BOTH			
ROSE	83.94	61.05	768
SMOTE	88.50	44.34	345
SMOTE + LOSS	82.55	59.84	852

Taula 7. Resultats decision trees

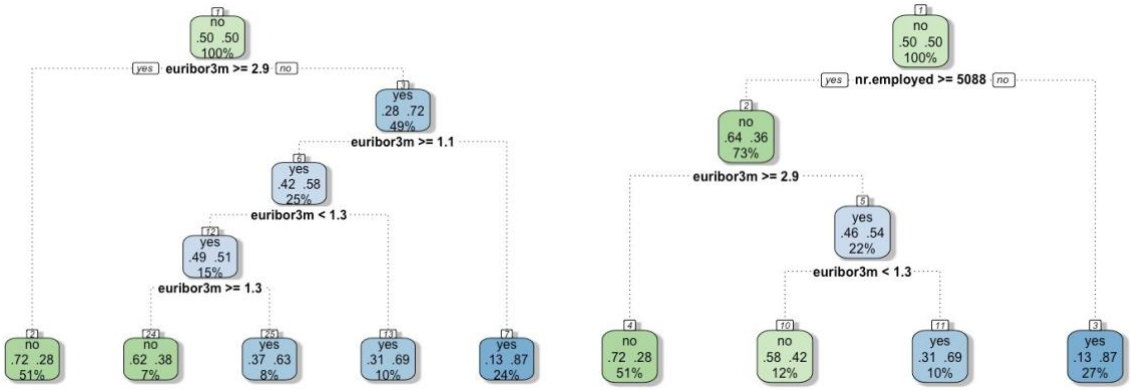
CART de tree amb undersampling



RANDOM FORESTS

	RANDOM FORESTS		
	Total	**Positive**	**True negative**
(.)	89.73	29.65	155
(.)PRIOR	88.78	37.74	277
(.)MORE TREES	88.72	37.06	276
(.)MTRY	89.30	30.32	188
(.)NODESIZE	86.04	61.99	638
OVERSAMPLING	87.68	45.96	411
UNDERSAMPLING	81.45	65.63	967
BOTH	86.33	53.77	558
ROSE	88.40	45.69	361
SMOTE	88.71	44.34	331

Taula 8. Resultats random forests



NEURAL NETWORK

Taula 9. Resultats NN

	NEURAL NETWORK		
	(·)		
POSTERIORS	**Total**	**Positive**	**True negative**
0.001	89.77	29.38	150
0.003	90.09	24.53	93
0.010	89.73	24.53	117
0.031	89.88	25.47	114
0.100	89.95	26.68	118
0.316	90.01	25.61	106
1.000	90.06	21.56	73

	NEURAL NETWORK		
	OVERSAMPLING		
POSTERIORS	**Total**	**Positive**	**True negative**
0.001	83.43	63.21	819
0.003	84.57	60.92	727
0.010	84.40	63.61	758
0.031	83.15	64.02	843
0.100	83.61	62.94	805
0.316	82.85	64.42	866
1.000	82.76	64.15	870

	NEURAL NETWORK		
	UNDERSAMPLING		
POSTERIORS	**Total**	**Positive**	**True negative**
0.001	76.45	67.66	1312
0.003	82.53	63.21	878
0.010	82.55	63.21	877
0.031	80.92	66.17	1006
0.100	81.71	64.82	944
0.316	79.42	64.56	1093
1.000	82.74	64.82	876

	NEURAL NETWORK		
	BOTH		
POSTERIORS	Total	Positive	True negative
0.001	74.72	71.02	1451
0.003	86.01	59.30	620
0.010	83.05	62.40	838
0.031	84.55	60.65	726
0.100	81.67	63.48	937
0.316	84.19	62.40	763
1.000	83.52	63.88	818

	NEURAL NETWORK		
	SMOTE		
POSTERIORS	Total	Positive	True negative
0.001	86.42	51.89	538
0.003	81.58	31.13	703
0.010	87.37	47.84	445
0.032	87.24	43.26	420
0.100	59.64	21.97	335
0.316	54.73	51.07	550
1.000	47.85	52.70	846

	NEURAL NETWORK		
	ROSE		
POSTERIORS	Total	Positive	True negative
0.001	87.74	33.83	317
0.003	87.25	55.12	507
0.010	88.30	46.77	376
0.032	88.34	47.04	375
0.100	88.16	43.26	359
0.316	85.13	60.65	688
1.000	86.51	58.49	581

SVM

		SVM, UNDERSAMPLING		
KERNEL	C	Total	Positive	True negative
RBF	1	83.70	63.07	800
LAPLACIÀ	0.5	79.34	67.52	1120

Taula 10. Resultats SVM

FEATURE SELECTION

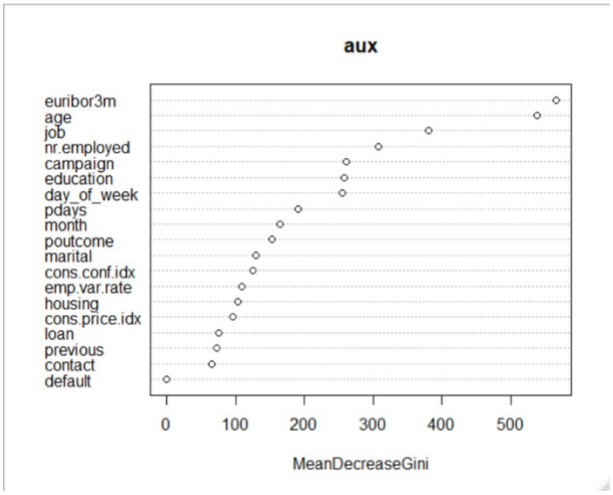


Figura 10. Feature selection

SVM

Taula 11. Feature selection SVM

		SVM, UNDERSAMPLING		
KERNEL	C	Total	Positive	True negative
LAPLACIÀ	0.3	82.41	65.90	906
LAPLACIÀ	2	82.47	65.36	898

RANDOM FOREST

Taula 12. Feature selection random forest

	RANDOM FORESTS		
	Total	**Positive**	**True negative**
UNDER (VAR 7)	83.29	63.88	833
UNDER (VAR 4)	83.15	64.02	843
UNDER (18 19)	79.56	69.41	1120

COMPARACIÓ DE MODELS

	MÈTODE	**TOTAL**	**POSITIVE**	**TRUE NEGATIVE**
LOGISTIC REGRESSION	BOTH	81.12	65.90	991
BAYES	(·)	81.99	60.78	896
DECISION TREES	UNDERSAMPLING	84.09	61.73	764
RANDOM FORESTS	UNDERSAMPLING	81.45	65.63	967
RF, FEATURE SELECTION	UNDERSAMPLING, 2 VARIABLES	79.56	69.41	1120
NEURAL NETWORK	UNDERSAMPLING, DECAY=1	82.74	64.82	876
SUPORT VECTOR MACHINE	UNDERSAMPLING,RBF, C=1	83.70	63.07	800
SVM, FEATURE SELECTION	UNDERSAMPLING, LAPLACIÀ, C=2	82.47	65.36	898

Taula 13. Comparació de models

RE-ENTRENAMENT DELS MODELS

	MÈTODE	**TOTAL**	**POSITIVE**	**TRUE NEGATIVE**
LOGISTIC REGRESSION	BOTH	82.44	62.07	1095
BAYES	(·)	81.52	60.34	1154
DECISION TREES	UNDERSAMPLING	83.61	58.62	966
RANDOM FORESTS	UNDERSAMPLING	82.06	62.39	1129
RF, FEATURE SELECTION	UNDERSAMPLING, 2 VARIABLES	80.53	65.41	1283
NEURAL NETWORK	UNDERSAMPLING, DECAY=1			
SUPORT VECTOR MACHINE	UNDERSAMPLING,RBF, C=1	84.30	60.34	925
SVM, FEATURE SELECTION	UNDERSAMPLING, LAPLACIÀ, C=2	84.86	59.59	872

Taula 14. re-entrenament dels models