TREBALL DE FI DE GRAU

Grau en Estadística

Títol: *Ensemble* de models predictius

Autor: Laura Nòria Santamaria

Director: Catalina Bolancé Losilla

Departament: Econometria, Estadística i Economia Aplicada

Convocatòria: 1

Data i localització: Juny del 2020, Barcelona

UNIVERSITAT DE BARCELONA

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat de Matemàtiques i Estadística

Resum

En aquest treball és demostra com la nova era del Big-data és essencial en el món actual i, concretament, en el sector de les assegurances. Mitjançant la gran quantitat de dades disponibles avui en dia, i mitjançant la modelització predictiva, s'ha creat una model òptim per tal de predir si el client d'una empresa d'assegurances de la llar, renovarà o no la pòlissa que té contractada. Això és un canvi gegant en el món dels negocis i una gran oportunitat per canalitzar la teoria del risc cap a la predicció de pèrdues. S'intenta demostrar en aquest estudi que en aquesta actual i nova economia digital, el Big-data i el Machine Learning són dos factors claus per millorar tant el sector de les assegurances com molt d'altres, i així poder augmentar les vendes, conèixer millor als clients, créixer en l'àrea de negocis, ser més competitius, formar als treballadors per tal que siguin els millors, etc. Ens trobem en una societat on tot el que ens envolta és informació, i aquesta no n'hi ha prou en que sigui recollida sinó que també ha de ser analitzada i tractada per tal de poder-ne extreure prediccions i plans d'acció per poder millorar.

Paraules clau:

Big-data, pòlissa, modelització, anàlisi de dades, llar, matriu de decisió, ensemble, regressió logística, arbre de decisió, xarxes neuronals, svm, predicció

Classificació AMS

91B82 Statistical methods; economic indices and measures

97K80 Applied statistics

Índex

Introducció

A l'hora de decidir si renovarà una pòlissa concreta, el client té en compte diferents factors per tal de prendre la decisió més encertada (renovar la pòlissa o no renovar-la). Aquests factors, sovint estan relacionats amb el preu, el temps que fa que és vigent, el mètode de pagament, el nombre de suplements de la pòlissa, etc.

En aquest estudi, el que es pretén és trobar el millor model que predigui si el client renovarà o no una pòlissa en qüestió. Per fer-ho, s'utilitzaran tan variables relacionades amb la pòlissa com amb el client, i també relacionades amb la llar. La metodologia a seguir serà la següent: es calcularà una predicció de la variable objectiu *policy_status_at_t2* (=1 si el client renovarà la pòlissa, =0 altrament) mitjançant diferents mètodes de predicció com els models Logit i Probit, Arbres de Decisió, Xarxes Neuronals o Màquines de Suport Vectorial. A partir de les prediccions obtingudes, es crearan uns models d'ensemble que seran diferents combinacions de tots els altres. Per últim, es compararà la qualitat de la predicció de tots els models i també la dels models ensemblats i s'escollirà el que millor predigui si el client renovarà o no la pòlissa.

L'ensemble de models naix per la necessitat de millorar la predicció d'un sol model. Existeix una gran quantitat d'estudis que demostren els avantatges de fer un ensemble. Per entendre-ho millor, suposem que volem predir quins restaurants són els que ofereixen una millor qualitat i servei. Una possibilitat és anar a tots els restaurants, provar-los i posar-los una puntuació a cadascun. Tot i així, això no sembla gaire raonable. El que és habitual és preguntar a amics i familiars, revisar els menús, eliminar els que de forma evident no ens agraden i menjar en aquells que pensem que tenen bastanta probabilitat de ser bons (Dietterich, 2000). Si estenem aquesta analogia al camp del Machine Learning, un ensemble es com si un grup de persones busquessin un bon restaurant en una mateixa zona i entre tots escollissin quin és el millor. Amb l'ensemble de models, doncs, no ens conformem en entrenar un model per predir les nostres dades sinó que n'entrenen uns quants per poder comparar-los i escollir quin fa una millor predicció.

Pel que fa a l'estudi, aquest està dividit en tres parts principals. El primer de tots és la preparació de les dades, també anomenat *preprocessing*. Consisteix en definir la classe de dades amb les quals es treballarà (numèriques o factors), la depuració d'errors i imputació de possibles valors mancants *(missings)*. Aquesta part és un aspecte clau per assegurar l'obtenció d'un anàlisi posterior vàlid i de qualitat. També es realitzarà una descripció individual per cadascuna de les variables i una descripció bivariant de cadascuna d'elles amb la variable objectiu.

La segona part fa referència als mètodes de predicció utilitzats, es mostrarà una explicació teòrica i, seguidament, els resultats obtinguts per cada mètode amb les dades de la llar. Per predir la renovació o no de la pòlissa s'implementaran models de classificació per separar les dues poblacions esmentades. Entre molts mètodes de classificació existents s'ha decidit implementar concretament els models Logit i Probit, que són models aleatoris d'elecció discreta i binària; l'arbre de decisió condicional, que estimen una relació de regressió mitjançant particions recursives binàries en un marc d'inferència condicional; les Xarxes Neuronals, que tenen la capacitat de fer prediccions davant la presència de relacions no lineals; o les Màquines

de Suport Vectorial (SVM), que són un conjunt d'algoritmes d'aprenentatge supervisat, també relacionats amb problemes del classificació binària i regressió. Per cadascun d'aquests mètodes es realitza una petita introducció, interpretació dels resultats, validació i avaluació de la capacitat predictiva. Aquesta avaluació ajudarà a concloure quin d'ells és el millor mètode de classificació respecte el context de les nostres dades.

Per avaluar els algoritmes s'ha decidit construir la matriu de confusió amb els quatre tipus de situacions possibles – vertader positiu, vertader negatiu, fals positiu i fals negatiu- i extreure'n l'error de predicció per validació creuada. Per fer-ho s'han dividit les dades en dos subconjunts, el d'entrenament i el de validació. L'objectiu de l'estudi és la classificació entre els clients que tenen la pòlissa activa i els que no prioritzant la proporció de vertaders positius (sensibilitat). Una vegada escollit el model més adient per a l'objectiu de l'estudi s'avaluarà la seva capacitat predictiva amb el conjunt validació.

La tercera part de l'estudi inclou la comparació dels cinc models de predicció individuals i els tres mètodes d'ensemble. Aquests últims també seran comparats amb els models individuals i, d'aquesta manera, es podrà decidir quina és la millor manera per predir si les pòlisses en qüestió seran renovades o no, assolint, així, l'objectiu final de l'estudi.

Les dades d'estudi

Descripció de les dades

L'anàlisi es realitza a partir de les dades d'una empresa d'assegurances que contenen informació sobre les diferents pòlisses contractades durant l'any 2013. La base de dades utilitzada, en concret, conté només la informació de les pòlisses de la llar, que formen un total de 206 812 observacions.

A continuació es mostra una breu descripció de les variables. Es pot observar que hi ha variables relaciones amb la pòlissa, d'altres relacionades amb el client i d'altres relacionades amb la llar en qüestió. Es treballarà amb un total de 13 variables més la variable objectiu (*policy_status_at_t*). Hi han variables de la base de dades inicial que no s'han utilitzat o bé perquè s'han utilitzat per la creació de noves variables o bé perquè contenen una gran quantitat de valors mancants o bé perquè s'ha considerat poc rellevants i/o difícils de tractar. Aquestes seran enumerades al final.

A continuació, es mostra la descripció de les 14 variables:

Nom de la variable	Descripció
sex_customer	Sexe del client
Age_client	Edat del client
HomeType	Tipus d'habitatge (AT=Àtic, PB=Planta Baixa, PI=Pis, RU=Rural, UA=Casa individual, UF=Casa familiar)
nunpol_sum	Nombre total de pòlisses que té el client
Insuredcapital_content_H	Contingut de capital assegurat
Insuredcapital_continent_H	Continent de capital assegurat
Policy_seniority	Antiguitat de la pòlissa
Client_Seniority	Antiguitat del client
Policy_numSupplements	Nombre de suplements de la pòlissa
Policy_PaymentMethod	Mètode de pagament de la pòlissa (A=Anual, S=Semestral, T=Trimestral)

Nom de la variable	Descripció
nclaims	Nombre de sinistres ocorreguts en total
nclaims_A	Nombre de sinistres ocorreguts per culpa de l'assegurat
nclaims_ C	Nombre de sinistres ocorreguts per culpa del contrari
policy_status_at_t	Variable binària que és 1 si la pòlissa està activa en l'any en qüestió (2013) i 0 si està anul·lada

Taula 1: Enumeració i explicació de les variables utilitzades en el model.

Les variables que no s'han inclòs en el model són: *client_id, policy_id, year, Client_DateofLastcancel, Policy_CancelDate, next_renewal_date, nunpol_life_saving, nunpol_accidents, nunpol_InsRetPlan, nunpol_other, previous_to_last_premium_paid, last_premium_paid. dif_current_previous, dif_current_first, yearlast, yearCancel, yearpstart, postalcode, year_last_cancellation, Client_LastStartDate i fault_C.*

Preprocés de les dades

Per poder treballar adequadament amb les dades, en primer lloc s'ha recodificat la variable d'estudi estat de la pòlissa, de manera que les pòlisses vigents siguin un 1 i les anul·lades un 0. A continuació, s'ha estudiat la tipologia de cada variable i s'han definit de forma corresponent com a factors o numèriques per a la seva correcta lectura. Pel que fa als valors mancants o *missings*, aquests han estat eliminats de la base de dades.

Per realitzar l'arbre de decisió condicional i també el *Suport Vector Machine* (SVM) és necessari que totes les variables siguin numèriques o binàries. Com que les variables *HomeType* i *Policy_PaymentMethod* són categòriques, és pertinent crear una variable dicotòmica per a cadascuna de les categories de la variable i que sigui igual a 1 si la característica es manifesta en cada observació o 0 si no. Tot i així, s'observa que en ambdós casos hi ha una categoria que és clarament majoritària respecte a la resta. Es per això que, per simplificar, s'ha recodificat la variable de manera que sigui igual a 1 si aquesta correspon a la categoria majoritària i 0 si no. En el cas de la variable tipus de llar (*HomeType*) la categoria predominant és "PI", és a dir, pis. Per tant, totes les observacions que pertanyin a la categoria pis seran igual a 1 mentre que tota la resta seran 0. Pel que fa a la variable mètode de pagament (*Policy_PaymentMethod*), el més habitual és l'anual ("A") amb una majoria del 0,94%, per tant, serà recodificat com un 1 i la resta com un 0.

A continuació es mostra una taula de la proporció d'observacions per a cada categoria:

Mètode de pagament de la pòlissa		
A	S	T
0.9354	0.0583	0.0063

Taula 2: Proporció d'observacions per les tres categories de la variable Policy_PaymentMethod.

Tipus de llar					
AT	PB	PI	RU	UA	UF
0.0264	0.0049	0.6452	0.0251	0.1595	0.1390

Taula 3: Proporció d'observacions per les tres categories de la variable HomeType.

Després de re codificar ambdues variables, la proporció de 0 i 1 per cada variable és la següent:

Mètode de pagament de la pòlissa	
0	1
0.0646	0.9354

Taula 4: Proporció d'observacions de la variable Policy_PaymentMethod després de la re categorització.

Tipus de llar	
0	1
0.3548	0.6452

Taula 5: Proporció d'observacions de la variable HomeType després de la re categorització.

El software utilitzat ha estat l'RStudio i l'Excel.

Partició de la base de dades i validació creuada

Els mètodes de remostreig són molt útils a l'hora d'avaluar els models estadístics ja que permeten ajustar un model diverses vegades utilitzant subconjunts del *training data set*. El mètode que s'utilitzarà em aquest estudi és el de validació creuada, que és el més senzill. Aquest consisteix en dividir aleatòriament les observacions disponibles en dos grups, un que s'utilitza per entrenar el model i l'altre per avaluar-lo. S'obté així la següent classificació:

- *Training data set* o conjunt d'entrenament: conjunt de dades/observacions amb les quals es genera el model estadístic, està format per dos terceres parts de les dades originals.
- *Test data set* o conjunt de validació: conjunt de dades/observacions que són del mateix tipus que les que formen el *training data set* però que no s'han utilitzat en la creació del model. Aquest conjunt de dades s'utilitza per avaluar el model final.

Una vegada seleccionats els predictors adequats, generat el model y comprovat que es compleixen les condicions necessàries del mètode d'ajust utilitzat, el següent pas és avaluar la capacitat del model per predir la variable resposta.

- *Training error date* o error de predicció: error que comet el model al predir observacions que provenen del *training data set*.
- *Test error rate*: error que comet el model al predir observacions d'un test data set i que per tant el model no ha "tractat".

Cal tenir en compte però que l'aplicació de la validació creuada implica:

- Haver d'ajustar el model repetides vegades, fet que suposa un alt cost computacional. Tot i així, el conjunt de dades estudiat no requereix un temps excessiu per executar els diferents algoritmes.
- L'estimació del *test error date* és altament variable depenent de quines observacions s'inclouen com a conjunt d'entrenament i quines com a conjunt de validació, això pot comportar a un problema de variància.
- A l'excloure una part de les observacions disponibles com a dades d'entrenament, es disposa de menys informació amb la que crear el model i per tant es redueix la seva capacitat. Com a conseqüència, això pot provocar una sobreestimació del *test error* comparat amb el que s'obtindria si s'utilitzessin totes les observacions per l'entrenament, hi hauria un problema de biaix.

Descriptiva

Descriptiva univariant

En aquest apartat es realitza un anàlisi descriptiu de les variables utilitzades en el model de forma individual.

En primer lloc es representa gràficament la variable dependent o variable objectiu per saber la quantitat de pòlisses que estan actives i les que estan anul·lades.

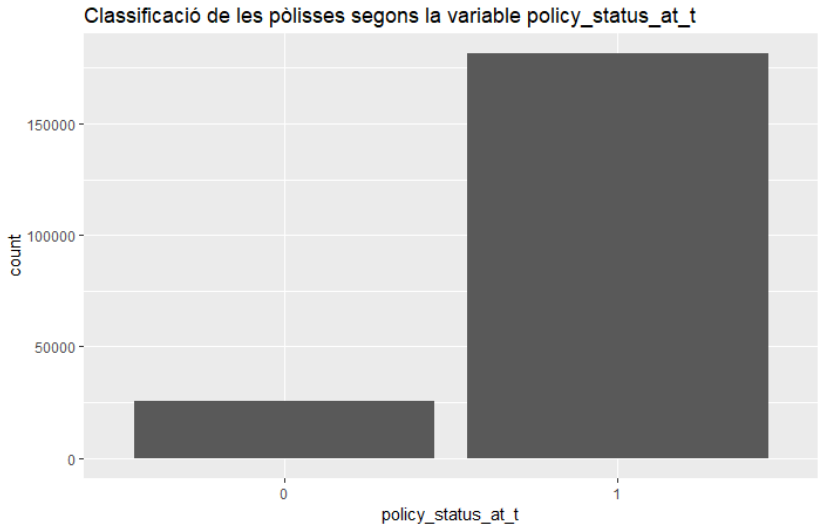


Figura 1: Diagrama de barres que mostra la proporció de pòlisses que estan actives (1) i pòlisses que s'han anul·lat (0).

En concret s'observen 181 365 pòlisses actives i 25 447 d'anul·lades, és a dir, el 87.7% de les pòlisses estan actives mentre que el 12.3% restant, no.

A continuació, s'analitzen les variables categòriques, que s'utilitzaran com a explicatives en els models de predicció. Les variables que presenten poques categories seran representades per un diagrama de sectors per tal de facilitar la seva interpretació.

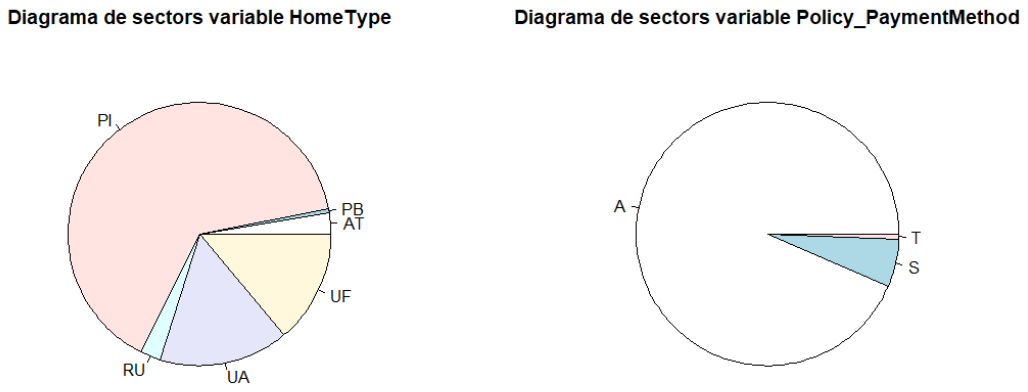


Figura 2: diagrama de sectors de les variables tipus de llar i mètode de pagament de la pòlissa.

S'observa que la majoria de pòlisses de la llar fan referència a pisos, concretament en un 64.5%, però també abunden les cases individuals (15.9%) i les cases familiars (13.9%). Pel que fa al mètode de pagament s'observa que la gran majoria de pòlisses es paguen anualment, exactament un 93.5%.

Les variables que s'estudien a continuació han estat recodificades de forma binària ja que la major part de les observacions són igual a 0 i les restants representen una proporció molt minòritària, es per això que s'ha decidit agrupar totes les minories. Per exemple, la variable *nclaims*, que representa el nombre total de sinistres ocorreguts durant la vigència d'una pòlissa, tant per culpa del contrari com de l'assegurat, 0 significa que no ha ocorregut cap sinistre i 1 que ha ocorregut un sinistre o més. El mateix s'ha realitzat amb les variables nclaims_A i nclaims_C que indiquen el nombre de sinistres ocorreguts per culpa de l'assegurat i per culpa del contrari, respectivament. La variable nclaims és la suma de les variables nclaims_A i nclaims_C. Els gràfics es mostren a continuació, juntament amb la variable sex_customer, que és binària de naturalesa.

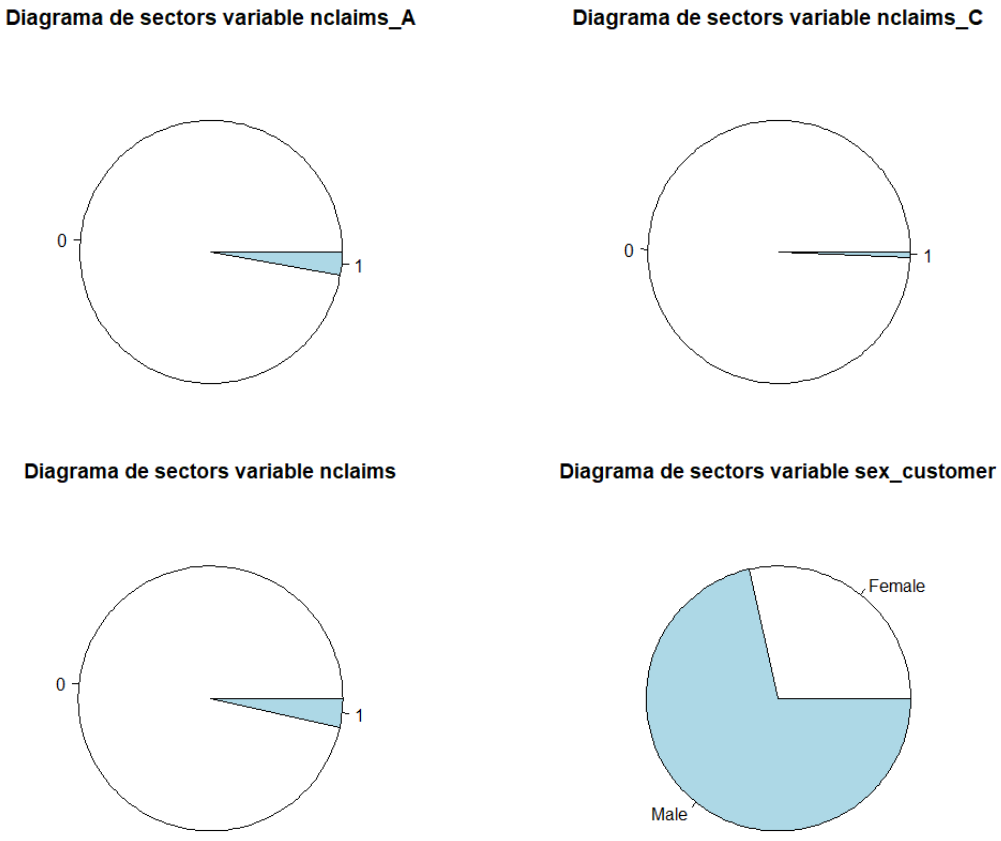


Figura 3: Diagrama de sectors de les variables nombre de sinistres ocorreguts per culpa de l'assegurat, nombre de sinistres ocorreguts per culpa del contrari, nombre de sinistres ocorreguts en total i proporció de pòlisses segons el sexe del client.

Els tres primers gràfics mostren com la major part dels clients no tenen cap sinistre associat a la pòlissa. Pel que fa a la variable sexe del client es pot observar que gairebé tres quartes parts dels titulars de les pòlisses són homes (un 71.4%) i només un 28.5% són dones.

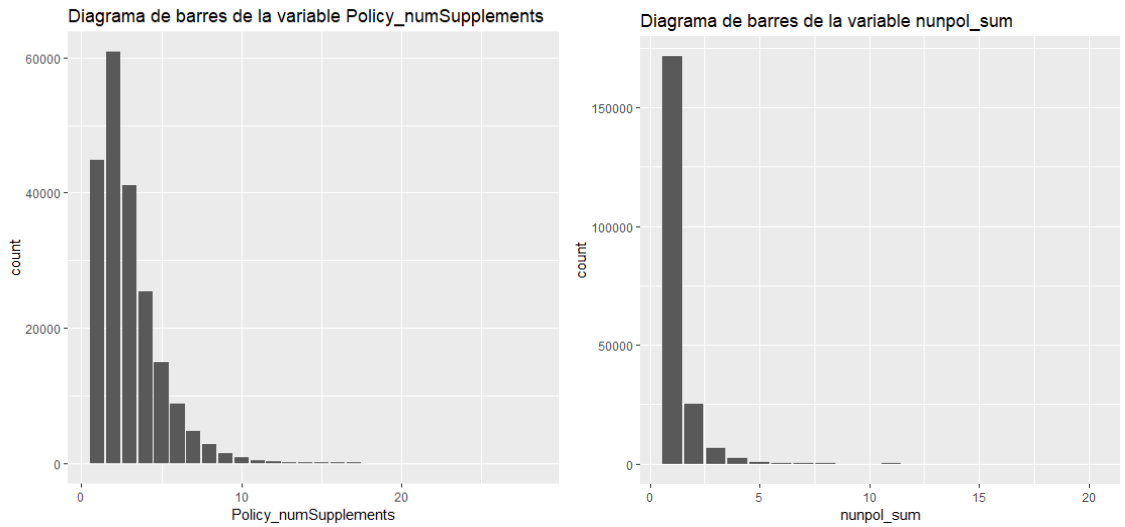


Figura 4: Diagrama de barres de les variables numèriques discretes nombre de suplements de la pòlissa i nombre de pòlisses contractades pel client.

S'observa en el primer gràfic que la majoria de pòlisses o bé no tenen contractades cap suplement, o bé només en tenen un, o bé en tenen dos. Hi ha poques pòlisses que tinguin contractades més de 5 suplements. Pel que fa a la quantitat d'altres pòlisses que té contractades el client a més de la pòlissa en qüestió, observem que gairebé tots els clients no en tenen cap altra de contractada.

L'última variable tractada com a categòrica és l'edat del client. Inicialment era una variable contínua que s'ha decidit dividir en intervals per agilitzar el seu tractament i la seva comprensió. Per escollir el nombre d'intervals s'ha utilitzat el criteri de Sturges que segueix la següent fórmula: $k = [1 + log_2(n)]$, on n =nombre d'observacions de la variable *Age_client*. En total, s'han creat 19 intervals d'una amplada de 3.5. S'observa que l'interval amb més observacions és el comprés entre 63.8 i 67.4 anys. La mitjana d'edat és de 60 anys i mig. A continuació es mostra el diagrama de barres d'aquesta variable.

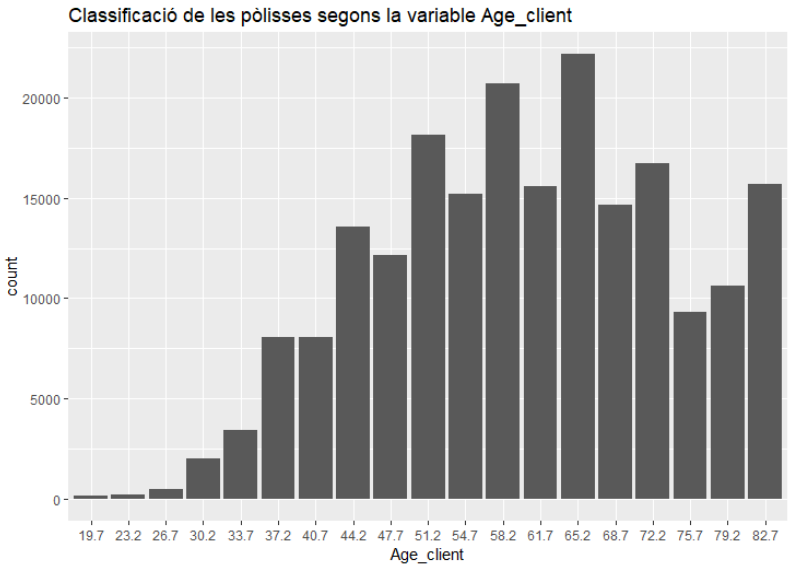


Figura 5: Diagrama de barres de la variable Age_client dividida en intervals.

Pel que fa a les variables numèriques contínues s'ha realitzat una gràfica de densitat per tal d'estudiar la seva distribució.

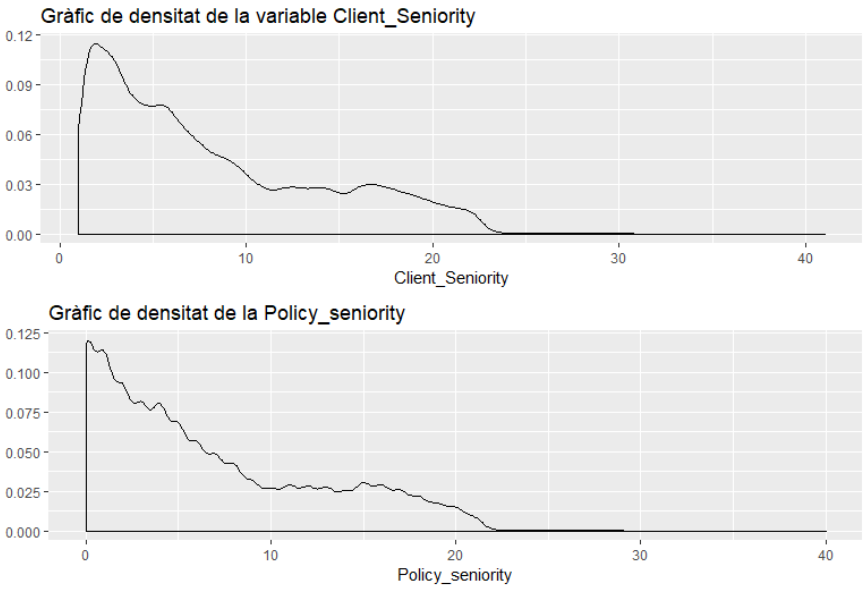


Figura 6: Gràfics de densitat per les variables antiguitat del client i antiguitat de la pòlissa.

S'observa que en cap de les dues variables presenta normalitat. Pel que fa a l'antiguitat del client, la mitjana és d'uns 8 anys formant part de la companyia, tot i que la moda és de 1.5. L'antiguitat de les pòlisses és de mitjana 6 anys i mig, però la seva moda és 0, cosa que indica que moltes de les pòlisses van ser contractades recentment.

A continuació, es mostren els gràfics de densitat de les variables *Insuredcapital_content_H* i *Insuredcapital_continent_H*. Ambdós mostren que en les distribucions s'hi troben molts valors allunyats de la majoria.

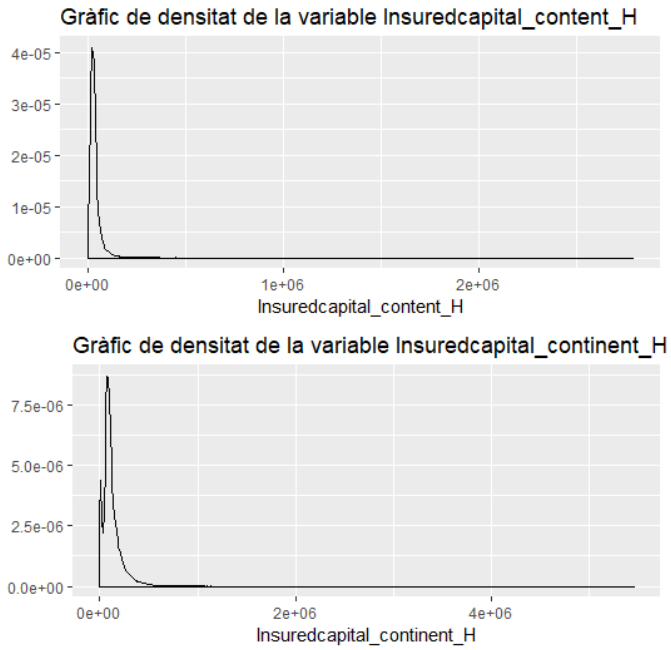


Figura 7: Gràfics de densitat per les variables contingut del capital assegurat i continent del capital assegurat.

	Mínim	1r quartil	Mediana	Mitjana	3r quartil	Màxim
Insuredcapital_content_H	1	18 828	28 705	35 865	41596	2 792 795
Insuredcapital_continent_H	180	65 583	95 490	122 132	147 682	5 453 879

Figura 6: Estadístics descriptius de les variables Insuredcapital_content_H i Insuredcapital_continent_H.

Al calcular els estadístics descriptius s'observa que el 95% dels valors es comprenen entre 18 828 i 41 596 en el cas del contingut del capital i entre 65 583 i 147 682 en el cas del continent.

Per veure també en les altres variables la presència de valors atípics, es representen les variables en *boxplots*, que són de molta ajuda per detectar valors extrems.

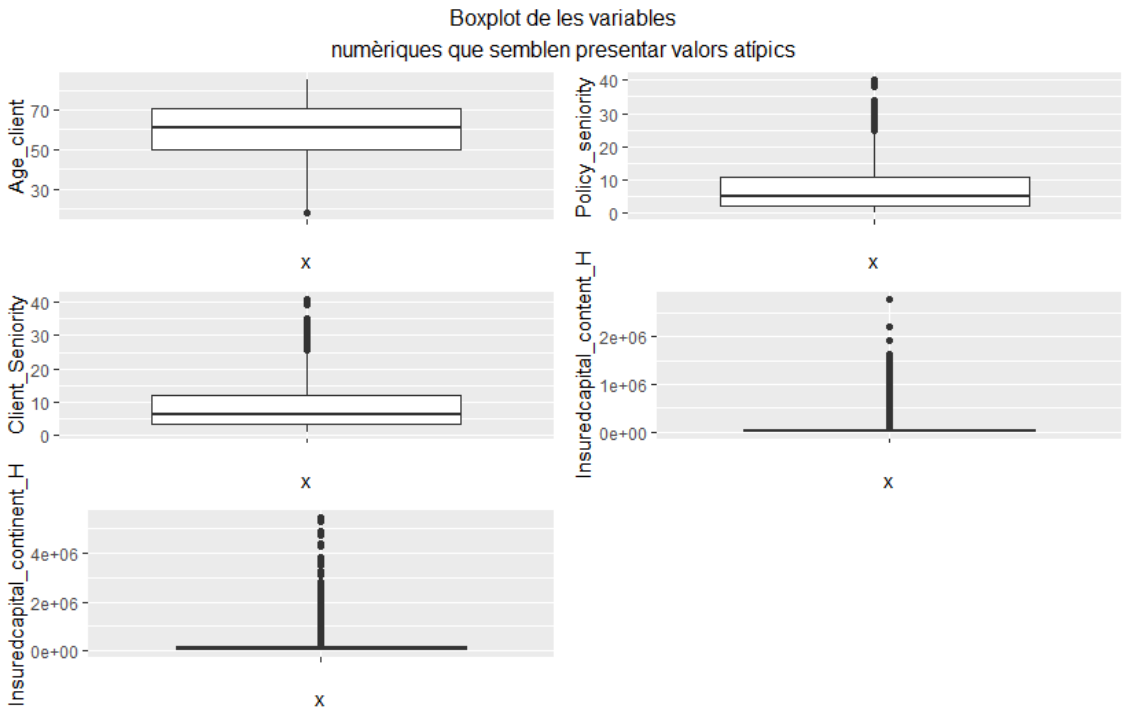


Figura 8: Boxplot per les variables edat del client, antiguitat de la pòlissa, antiguitat del client, contingut del capital assegurat i continent del capital assegurat.

S'observa clarament que les variables presenten valors atípics (la variable edat del client només en presenta un), tot i que només són significatius els valors extrems de les dues variables comentades anteriorment, la resta es mantenen perquè poden contenir informació per als resultats.

Descriptiva bivariant

En aquest apartat s'estudia la distribució de les diferents variables en funció de la variable d'estudi: *policy_status_at_t*. Els gràfics que segueixen a continuació permeten identificar les variables que seran més útils per discriminar les pòlisses actives i les anul·lades.

Figura 9: Gràfiques de densitat de les variables numèriques continues en funció de la variable policy_status_at_t

Segons el que mostren els gràfics, les distribucions són bastant similars tan si la pòlissa està activa com si no ho està, no hi ha cap distribució que sigui significativament diferent quan la pòlissa està anul·lada.

Per altra banda, per comparar la distribució de les observacions en les diferents classes que presenten les variables categòriques segons la variable objectiu, es construeixen taules.

Taula de l'estat de la pòlissa en funció del tipus d'habitatge del client		
	pòlissa anul·lada	pòlissa activa
àtic	0.0254647	0.0265321
planta baixa	0.0048729	0.0048741
pis	0.6466381	0.6450473
rural	0.0228711	0.0253853
casa individual	0.1608048	0.1592810
casa familiar	0.1393484	0.1388802

Taula 7: Taula de la variable categòrica HomeType respecte la variable policy_status_at_t.

S'observa que gairebé no hi ha diferències entre els valors encara que variï l'estat de la pòlissa.

Taula de l'estat de la pòlissa en funció de la forma de pagament		
	pòlissa anul·lada	pòlissa activa
Anual	0.898770	0.940523
Semestral	0.091759	0.053577
Trimestral	0.009471	0.005900

Taula 8: Taula de la variable categòrica Policy_PaymentMethod respecte la variable policy_status_at_t.

En aquesta variable s'observen més diferències entre les proporcions obtingudes segons el mètode de pagament quan la pòlissa està activa o anul·lada però aquestes segueixen sent poc significatives.

Taula de l'estat de la pòlissa en funció del sexe del client		
	pòlissa anul·lada	pòlissa activa
dona	0.120350	0.879649
home	0.124122	0.875878

Taula 9: Taula de la variable categòrica sex_customer respecte la variable policy_status_at_t.

La taula indica, tant els homes com les dones, que el 12% de les pòlisses són nul·les i el 87% estan actives. El sexe del client no influeix significativament en la variable d'estudi.

Pel que fa al nombre de sinistres ocorreguts, tant per culpa de l'assegurat, com per culpa del contrari o la suma dels dos, s'han agrupat en un mateix bloc els casos en que el nombre de sinistres ocorreguts és 2 o major que 2 ja que aquests constitueixen un percentatge molt petit sobre el total.

Taula de l'estat de la pòlissa en funció del nombre de sinistres ocorreguts per culpa de l'assegurat		
	pòlissa anul·lada	pòlissa activa
0 sinistres	0.981176	0.969911
1 sinistre	0.017173	0.028010
2 o més sinistres	0.001650	0.002079

Taula 10: Taula de la variable categòrica nclaims_A respecte la variable policy_status_at_t.

No sembla que hi hagin diferències significatives entre les dos classes.

Taula de l'estat de la pòlissa en funció del nombre de sinistres ocorreguts per culpa del contrari		
	pòlissa anul·lada	pòlissa activa
0 sinistres	0.996306	0.993290
1 sinistre	0.003576	0.006589
2 o més sinistres	0.000118	0.000121

Taula 11: Taula de la variable categòrica nclaims_C respecte la variable policy_status_at_t.

En aquest cas, observem que quan ha ocorregut 1 sinistre per culpa del contrari, la probabilitat que la pòlissa estigui activa és el doble de que estigui anul·lada.

Taula de l'estat de la pòlissa en funció del nombre de sinistres ocorreguts en total		
	pòlissa anul·lada	pòlissa activa
0 sinistres	0.977640	0.963604
1 sinistre	0.020474	0.033832
2 o més sinistres	0.001886	0.002564

Taula 12: Taula de la variable categòrica nclaims respecte la variable policy_status_at_t.

En general, sembla que les diferències entre els dos grups són poc rellevants.

Taula de l'estat de la pòlissa en funció del nombre de suplements contractats		
	pòlissa anul·lada	pòlissa activa
0	0.000000	0.247369
1	0.397925	0.279414
2	0.252878	0.191514
3	0.152395	0.118694
4	0.087869	0.070074
5	0.052383	0.040730
6	0.025936	0.022871
7	0.014618	0.013128
8	0.007899	0.006892
9	0.003772	0.003964
10 o més	0.004323	0.005348

Taula 13: Taula de la variable categòrica Policy_numSumpplements respecte la variable policy_status_at_t.

Poden distingir-se algunes diferències entre pòlissa anul·lada i pòlissa vigent quan el nombre de suplements contractats és de 0, 1 o 2. No hi ha diferències significatives quan el client contracta 3 o més suplements per la pòlissa.

Taula de l'estat de la pòlissa en funció del nombre de pòlisses que el client té contractades		
	pòlissa anul·lada	pòlissa activa
0	0.886313	0.820147
1	0.083350	0.125972
2	0.019216	0.033154
3	0.006170	0.011750
4	0.003380	0.004505
5	0.000590	0.001902
6	0.000236	0.000777
7	0.000157	0.000540
8	0.000078	0.000248
9	0.000078	0.000220
10 o més	0.000432	0.000783

Taula 14: Taula de la variable categòrica nunpol_sum respecte la variable policy_status_at_t.

Similar a la taula anterior, les diferències principals entre quan la pòlissa s'ha anul·lat o segueix activa, són quan el client té contractades 1, 2 o 3 pòlisses. A parir de les 4 pòlisses contractades no hi ha diferències rellevants.

Mètodes predictius

Els mètodes predictius son models que apliquen resultats coneguts amb la finalitat d'entrenar un model per predir valors, amb dades diferents i completament noves, en un procés repetitiu. El model proporciona els resultats en forma de prediccions representades mitjançant el grau de probabilitat de la variable objectiu basat en la significació estimada a partir d'un conjunt de variables d'entrada.

Així doncs, els models predictius són diferents dels descriptius (que ens ajuden a entendre successos passats) o dels de diagnòstic (que ens ajuden a l'hora d'entendre les relacions entre entitats amb la finalitat d'esbrinar per què quelcom ha succeït). Hi ha dos tipus de mètodes predictius: els de classificació i els de regressió. Els models de classificació permeten predir la pertinença a una classe. Com en el nostre cas d'estudi, quines pòlisses seran renovades i quines no. Per això, s'estableixen variables d'entrada com el preu de la pòlissa, l'edat del client, el nombre de pòlisses que té, etc. Els resultats del model són binaris, o un sí o un no (en forma de 0 i 1) amb el seu grau de probabilitat. Els models de regressió, en canvi, ens permeten predir un valor. Per exemple, quin és el benefici de la companyia per un determinat client (o segment) en els pròxims mesos.

Els mètodes d'anàlisi predictiu més aplicats i que tractarem i avaluarem en aquest informe són: regressió lineal i logística (concretament veurem models d'elecció binària), arbres de decisió, xarxes neuronals i màquines de vectors de suport (SVM). L'últim mètode que posarem en pràctica serà el de l'ensemble dels models anteriors ja que ofereix una de les maneres més convincents per construir models predictius altament precisos. La tècnica en sí consistirà en construir un model nou entrenant diversos models similars combinant els resultats per millorar la precisió, reduir el biaix, reduir la variància i identificar el millor models per a utilitzar amb dades noves.

Models d'elecció discreta binària:

Els models d'elecció discreta binària formen part dels models de variable dependent limitada, que són aquells que admeten variables dependents amb valors restringits. Aquests valors a més de binaris també poden ser positius, múltiples restringits, de recompte, etc. Els models d'elecció discreta es caracteritzen per tenir una variable dependent que reflexa decisions individuals de respostes tancades. La resposta pot ser binària o múltiple. Els models Logit i Probit pertanyen al grup de models d'elecció discreta binària, juntament amb els models de probabilitat lineal. Els veiem a continuació.

Descripció dels models:

Ambdues aproximacions (logística i probit) són útils per modelar la probabilitat d'un esdeveniment (variable dependent) que succeeix com a funció d'altres factors (variables dependents o predictores). Formen part dels *Models Lineals Generalitzats (GLM).*

Ambdues metodologies utilitzen funcions d'enllaç (*linkage*) que permeten variables resposta amb distribució d'errors no gaussianes. Són molt útils per variables objectiu amb distribució *Poisson, Binomial* i *Gamma*, entre d'altres.

- Funció logit: funció d'enllaç amb aplicacions en regressió logística.
- Funció probit: funció d'enllaç amb aplicacions en regressió probit. Aquesta funció és la inversa de la funció de distribució normal estàndard.

L'estimació dels paràmetres per ambdós models pot fer-se a través de la Màxima Versemblança.

Encara que les estimacions dels dos models siguin similars, la regressió logística ha estat àmpliament utilitzada en entorns epidemiològics (ciències de la salut), mentre que la regressió probit és més comuna en contextos economètrics.

Observació: modelar una variable dicotòmica (x) amb la regressió lineal clàssica podria no restringir els valors de la resposta entre 0 i 1 (veure figura 10). A més, és altament probable que a l'utilitzar aquest tipus de models s'incompleixin els supòsits de normalitat residuals.

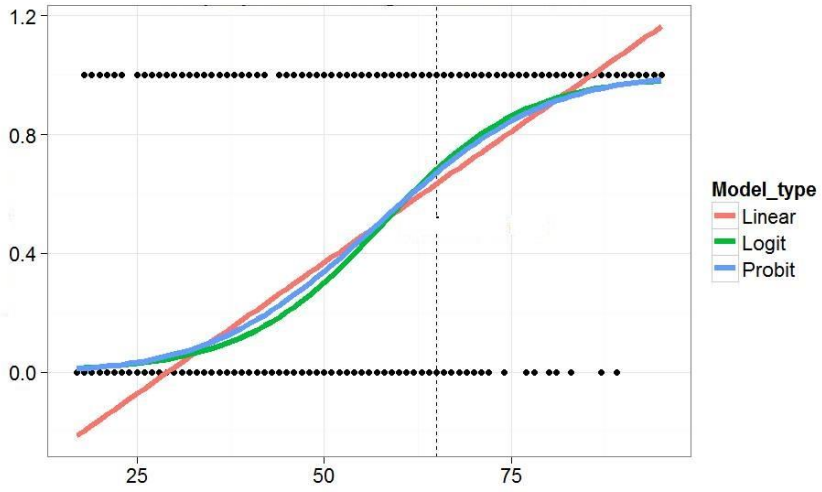


Figura 10: Models Lineal, Logístic i Probit.

Model Logit

El model Logit és, amb diferència, el model d'elecció discreta més senzill i d'ús més estès. La seva popularitat es deu al fet que la fórmula per les probabilitats d'elecció té una expressió tancada i fàcilment interpretable.

Com sorgeix?

Originalment, la fórmula Logit va ser obtinguda per Luce l'any 1959 a partir de certes assumpcions sobre les característiques de les probabilitats d'elecció, de les quals la més important era la independència d'alternatives irrellevants (IIA: *Independence from irrelevant alternatives*). Marschak, l'any següent (1960) va demostrar que aquests axiomes implicaven que el model era consistent amb un comportament del decisor orientat a la maximització de la utilitat. La relació de la fórmula Logit amb la distribució de la utilitat no observada (com a oposada a les característiques de les probabilitats d'elecció) va ser desenvolupada per Marley, tal com ho citen Luce i Suppes (1965), que van ser els que van demostrar que la distribució del valor extrem ens porta a la fórmula Logit. Al 1974, McFadden va completar l'anàlisi mostrant la relació inversa, és a dir, que la fórmula Logit per les probabilitats d'elecció necessàriament implica que la utilitat no observada es distribueix d'acord amb una distribució de valor extrem.

En què consisteix?

El model Logit, a més d'obtenir estimacions de la probabilitat d'un succés, el que ens permet és identificar els factors de risc que determinen aquestes probabilitats, així com la influència o pes relatiu que aquests tenen sobre aquestes mateixes probabilitats.

Aquest tipus de model treu com a resultat un índex, els determinants dels quals són coneguts, que permet efectuar ordenacions que al realitzar-se permeten, mitjançant algun mètode d'estratificació, generar classificacions en les que se li associa a cada element una qualificació.

Pel cas més senzill, el de una única variable explicativa, es tracta de trobar la relació que hi ha entre la variable explicativa i la endògena. Una possibilitat és que la funció que relaciona ambdues variables sigui una funció lineal, l'anomenat model lineal de probabilitat. Tot i així, hi ha molts casos en que aquesta relació entre les dues variables no té un comportament lineal, situació que origina els models de regressió no lineals, dins dels quals es troba el model Logit (i Probit).

La modelització Logit és similar a la regressió tradicional amb l'excepció que utilitza com a funció d'estimació la funció logística en lloc de la lineal. Amb la modelització Logit, el resultat del model és l'estimació de la probabilitat que un nou individu pertanyi a un grup o a un altre, mentre que per altra banda, al tractar-se d'un anàlisi de regressió, també permet identificar les variables més importants que expliquen les diferències entre grups.

Existeixen diferents tipus de models Logit en funció de les característiques que presenten les alternatives que defineixen a la variable endògena. Aquesta variable permet mesurar el nombre de grups existents en l'anàlisi. Els models Logit poden classificar-se segons si són, dicotòmics, de resposta múltiple, amb respostes no ordenades o no ordenades, multinomials, condicionals, etc.

El model Logit que s'utilitzarà en aquest estudi és el dicotòmic. Aquest presenta tres característiques principals:

- *Variable endògena binària*: identifica la pertinència d'un individu a una de les dos possibles categories, identificant amb un 1 i l'individu pertany a la característica d'interès (en aquest estudi, 1= la pòlissa serà renovada) la probabilitat de la qual serà estimada al model. S'identifica com a 0 a l'element que no compleix la característica d'interès (0= la pòlissa no serà renovada) la probabilitat de la qual també s'estimarà al model.
- *Variables exògenes*: són les variables que permeten discriminar entre els grups i que determinen la pertinència d'un element a un grup o a l'altre. Poden estar mesurades en escala nominal, ordinal, d'interval o de raó.
- *Resultat de l'anàlisi*: aquest és un vector de paràmetres amb valors numèrics, que són els coeficients per cadascuna de les variables explicatives que formen part del model. La importància radica en que a cada valor del vector de paràmetres li correspon una variable explicativa; al tenir-se en compte totes en conjunt i donar valors a cadascuna de les variables independents contingudes en el model definitiu, s'obté el valor de la probabilitat de que un individu compleixi la característica d'interès estudiada al model.

Quina forma presenta?

La regressió logística analitza dades amb distribució binomial de la següent forma:

$$Y_i \sim B(p_i, n_i) \quad per\ i = 1, \dots, m$$

En l'expressió anterior p_i fa referència a la probabilitat d'èxit (probabilitat de que succeeixi un esdeveniment sota estudi) y n_i determina el nombre d'assajos de tipus *Bernoulli*. El nombre d'assajos és conegut, tot i així, la probabilitat d'èxit es desconeix.

S'ha de complir que la resposta estigui acotada entre 0 i 1, és a dir, que el resultat sempre ha de ser positiu, a més de ser inferior a 1.

L'exponencial (e) de qualsevol valor (x) és sempre positiu i qualsevol nombre dividit entra la quantitat més u $(x + 1)$ sempre serà menor que 1. Sota aquestes dues premisses es pot expressar la següent probabilitat condicional (funció logística):

$$p(Y = 1|X) = \frac{e^{(\beta_0 + \beta_1 x)}}{e^{(\beta_0 + \beta_1 x)} + 1}$$

Per facilitar els càlculs escrivim $p(Y = 1|X) =$ com $p(X)$:

$$p(X) = \frac{e^{(\beta_0 + \beta_1 x)}}{e^{(\beta_0 + \beta_1 x)} + 1}$$

$$p(e^{(\beta_0 + \beta_1 x)} + 1) = e^{(\beta_0 + \beta_1 x)}$$

$$p \times e^{(\beta_0 + \beta_1 x)} + p = e^{(\beta_0 + \beta_1 x)}$$

$$p = e^{(\beta_0 + \beta_1 x)} - p \times e^{(\beta_0 + \beta_1 x)}$$

$$p = e^{(\beta_0 + \beta_1 x)}(1 - p)$$

$$\frac{p}{1-p} = e^{(\beta_0+\beta_1 x)}$$

Els logits (funció d'enllaç) de les probabilitats binomials desconegudes, és a dir, els logaritmes de *l'oportunitat relativa (odds ratio)* són modelats com una funció lineal dels X_i:

$$ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x$$

Aquesta funció d'enllaç és coneguda amb el nom de *sigmoide* i limita el seu rang de probabilitats entre 0 i 1 (veure figura 11).

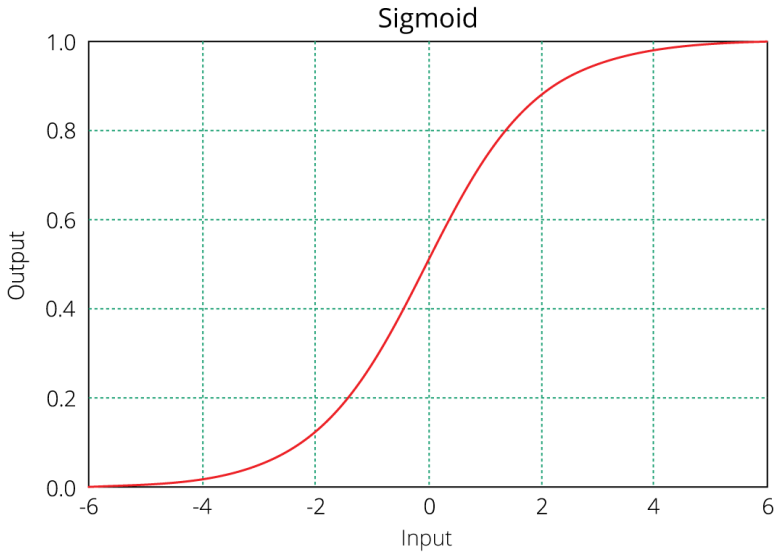


Figura 11: Funció sigmoide.

Model Probit

El model Probit naix arran de certes limitacions importants que presenta el model Logit. Aquestes són tres: per una banda, no permet representar la variació aleatòria de preferències. A més, presenta patrons de substitució restrictius degut a la propietat de IIA (*Independence from irrelevant alternatives*). Per últim, no pot utilitzar-se amb dades de panell quan els factors no observats estan correlacionats en el temps per cada decisor. Els models Probit resolen aquestes tres limitacions. Poden tractar la variació de preferències aleatòries, permeten qualsevol patró de substitució i són aplicables a dades de panell amb errors correlacionats temporalment.

L'única limitació dels models Probit és que requereixen distribucions normals per tots els components no observats d'utilitat. En la majoria de casos, les distribucions normals proporcionen una representació adequada dels comportaments aleatoris. Tot i així, en algunes situacions les distribucions normals són inadequades i poden conduir a prediccions incorrectes.

Com sorgeix?

El model Probit s'obté sota el supòsit que les utilitats no observades segueixen una distribució normal conjunta. La primera formulació d'un Probit binari es va dur a terme per Thurstone, l'any 1927, utilitzant una terminologia d'estímuls psicològics, terminologia que Marchak (1960) va traduir a termes econòmics com utilitat. Hausman i Wise (1978) i Daganzo (1979) va clarificar la generalització de l'especificació per representar diversos aspectes del comportament d'elecció.

Quina forma presenta?

La regressió probit permet analitzar dades amb resposta ordinal o amb distribució binomial (respostes dicotòmiques) de la forma:

$$Y_i \sim B(p_i, n_i) \quad per\ i = 1, \dots, m$$

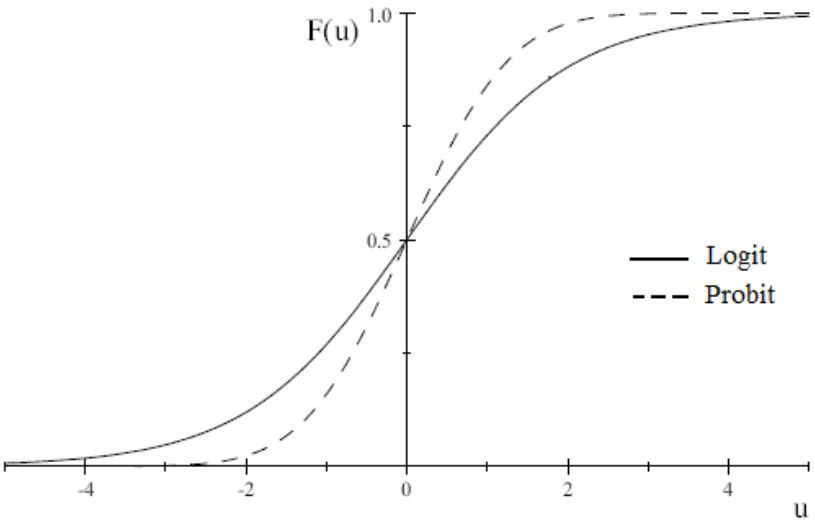
El marc conceptual del model probit pot expressar-se de la següent manera:

$$p(Y = 1|X) = \Phi(\beta_0 + \beta_1 x)$$

On $p(Y = 1|X)$ denota probabilitat, Φ és la funció de distribució acumulativa de la distribució normal estàndard i β són els paràmetres del model, estimats a través de la màxima versemblança.

El model pot ser expressat de la següent manera: $Y = \beta_0 + \beta_1 x + \epsilon \sim N(0,1)$

Les funcions logística i probit difereixen en la manera com defineixen la funció de distribució, mentre que la primera utilitza la funció logística, la segona fa ús de la funció de distribució acumulada de la normal estàndard. Ambdues funcions poden ser comparades en la figura següent:



<u>Figura 12:</u> Funció Logit i Probit.

Arbres de decisió:

En l'àmbit de l'aprenentatge automàtic, els mètodes combinats (mètodes d'ensemble) utilitzen múltiples algoritmes d'aprenentatge per obtenir un rendiment predictiu que millori el que podria obtenir-se mitjançant qualsevol algoritme d'aprenentatge individual dels que el constitueix.

La idea dels mètodes combinats és considerar múltiples hipòtesis simultàniament per formar una hipòtesi que, amb l'ajuda d'alguns teoremes essencials, es comporti millor. El terme "ensemble de models" s'acostuma a utilitzar per aquelles combinacions que fan ús de múltiples hipòtesis que pertanyen a una mateixa família, mentre que el terme més general "sistemes d'aprenentatge múltiples" s'utilitza quan les hipòtesis que es combinen provenen de famílies diferents.

Com que els mètodes combinats utilitzen diverses hipòtesis simultànies, es produeix una elevació dels costos computacionals, i és per això que habitualment com a espai d'hipòtesi base s'utilitzen algoritmes ràpids, com ho són els arbres de decisió.

Els arbres de decisió ("*tree*") són un mètode analític que a través d'una representació esquemàtica d'alternatives disponibles, facilita la presa de decisions, especialment quan existeixen riscos, costos, beneficis i múltiples opcions.

Els arbres de decisió són especialment útils quan:

- Les alternatives estan ben definides
- Les incerteses poden ser quantificades
- Els objectius són clars

A l'hora de desenvolupar un arbre de decisió, la metodologia a utilitzar és la següent:

1. Identificació de les variables del problema central
2. Enumeració de tots els factors que causin el problema o risc identificat
3. És important prioritzar i acotar cada criteri de decisió, per això, és convenient eliminar tots aquells factors que no siguin rellevants
4. Buscar i enumerar els factors de major a menor importància
5. Un cop establertes les variables pertinents, obtenir els factors que continguin fortaleses i debilitats
6. Per cada factor, generar supòsits de manera objectiva per tal de crear ramificacions. A l'hora de fer aquestes particions, poden aplicar-se diferents criteris com:

- *Error de classificació*: fracció d'observacions d'entrenament en una regió o node que no pertany a la classe més freqüent $E = 1 - màx_k(\hat{p}_{mk})$ on $\hat{p}_{mk}$=proporció d'observacions d'entrenament en la regió m que pertany a la classe k. Aquest criteri no acostuma a ser suficientment sensible a l'hora de mesurar la puresa dels nodes per crear l'arbre (millor utilitzar els altres dos criteris), però és preferible a l'hora d'aconseguir la màxima predicció en les prediccions de l'arbre podat final.
- *Índex de Gini*: mesura de la variància total en el conjunt de les K classes, o puresa dels nodes: $G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk})$. Aquest índex serà baix quan tots els nodes $\hat{p}_{mk}$ es trobin propers a 0 o 1. Un valor baix indica que en el node hi predominen observacions d'una sola classe.

- *Cross-entropy*: ve donat per $D = -\sum_{k=1}^{K} \hat{p}_{mk} \log(\hat{p}_{mk})$. La interpretació és la mateixa que per l'índex de Gini.

7. Seleccionar diverses alternatives, les més rellevants pel model
8. Implementar les alternatives d'acord amb els seus possibles problemes i riscos
9. Avaluar l'efectivitat de la decisió

En aquest estudi es treballarà amb els arbres de decisió condicionals.

A continuació veurem un tipus d'arbre de decisió, l'anomenat **CART**.

Els arbres de classificació y regressió, CART (Classification and Regression Trees), són un terme introduït per Leo Breiman en el seu llibre *Classification and Regression Trees (1984)* per referir-se als arbres de decisió que poden utilitzar-se per problemes de modelització predictiva de classificació o regressió. L'algoritme CART proporciona una base per algoritmes importants com els arbres de decisió en bosses, Random Forest i arbres de decisió potenciats.

La representació pel model CART és un arbre binari. Cada node arrel representa una sola variable d'entrada (x) i un punt de divisió en aquesta variable (suposant que la variable sigui numèrica). Els nodes fulla de l'arbre contenen la variable de sortida (y) que s'utilitza per fer la predicció.

La manera que té de fer prediccions és la següent: donada una nova dada, l'arbre es travessa avaluant l'entrada específica iniciada al node arrel de l'arbre. Un arbre binari après és en realitat una partició de l'espai d'entrada. El que fa és dividir aquest espai p-dimensional en rectangles (quan p=2) o híper-rectangles. Les noves dades es filtren a traves de l'arbre i aterren a un dels rectangles. El valor de sortida per aquest rectangle és la predicció realitzada pe model.

En aquest estudi es treballarà amb els arbres de decisió condicionals. S'expliquen a continuació.

Arbres de decisió condicionals

Els arbres d'inferència condicionals, també anomenats particions recursives imparcials, són una classe no paramètrica d'arbres de decisió que utilitza una teoria estadística (selecció mitjançant proves de significació basades en permutació) per tal de seleccionar variables en lloc de seleccionar la variable que maximitza una mesura d'informació (coeficient de Gini o guany d'informació) i, per tant, elimina el biaix potencial en CART o arbres de decisió similars.

Els arbres condicionals estimen una relació de regressió mitjançant particions recursives binàries. Aproximadament, l'algoritme funciona de la manera següent:

1. Provar la hipòtesi nul·la global d'independència entre qualsevol de les variables d'entrada i la resposta. Aturar-se si aquesta hipòtesi no es pot rebutjar. En cas contrari, seleccionar la variable d'entrada amb l'associació més forta a la represa. Aquesta associació es mesura mitjançant el p-valor corresponent a una prova per a la hipòtesi nul·la parcial d'una sola variable d'entrada i la resposta.
2. Implementar una divisió binària a la variable d'entrada seleccionada.
3. Repetir de forma recurrent els passos 1 i 2.

Xarxes neuronals

Com sorgeixen?

Les xarxes neuronals (ANN: Artificial Neural Networks) van sorgir originàriament com una simulació abstracta dels sistemes nerviosos biològics, constituïts per un conjunt d'unitats anomenades neurones connectades les unes amb les altres.

El primer model de xarxa neuronal va ser proposat per McCulloch i Pitts (1943) en termes d'un model computacional d'activitat nerviosa. Aquest model era un model binari, on cada neurona tenia un esglaó o llindar prefixat, i va servir de base pels models posteriors.

Característiques

Les ANN aplicades estan en general inspirades en les xarxes neuronals biològiques, encara que tenen altres funcionalitats i estructures de connexió diferents a les observades des de la perspectiva biològica. Les característiques principals de les ANN són les següents:

- Auto-organització i adaptabilitat: utilitzen algoritmes d'aprenentatge adaptatiu i d'auto-organització, per la qual cosa ofereixen millors possibilitats de processament robust i adaptatiu.
- Processament no lineal: augmenta la capacitat de la xarxa per aproximar funcions, classificar patrons i augmentar la seva immunitat respecte al soroll.
- Processament paral·lel: normalment s'utilitza un gran nombre de nodes de processament, amb un alt nivell de inter-connectivitat.

Quin és el seu funcionament?

L'element bàsic de computació (model de neurona) és un node o unitat. Aquest, rep un input des d'altres unitats o des d'una font externa de dades. Cada input té un pes associat w, que es va modificant en l'anomenat procés d'aprenentatge. Cada unitat aplica una funció donada f de la suma dels inputs ponderats mitjançant els pesos $y_i = \sum_j w_{ij} y_j$. El resultat pot servir com a output d'altres unitats.

Normalment, els pesos òptims s'obtenen optimitzant (minimitzant) alguna funció d'energia. Per exemple, un criteri molt utilitzat és l'anomenat *entrenament supervisat*, que consisteix en minimitzar l'error quadràtic mitjà entre el valor de sortida i el valor real esperat.

Funció d'activació

El valor de xarxa, expressat per la funció de base $u(w;x)$, es transforma mitjançant una funció d'activació no lineal. Les funcions d'activació més comunes són la logística i la tangent hiperbòlica:

- Funció logística: $F(x) = \frac{1}{1+e^{-x}}$
- Funció hiperbòlica: $F(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$

Estructures de connexió de darrera cap a endavant

Una xarxa neuronal està formada per les neurones i la matriu de pesos. Es poden definir tres tipus de capes de neurones: la capa d'entrada, la capa oculta i la capa de sortida. Entre dos capes de neurones existeix una xarxa de pesos de connexió, que pot ser dels següents tipus: cap a endavant, cap a endarrere i de retard.

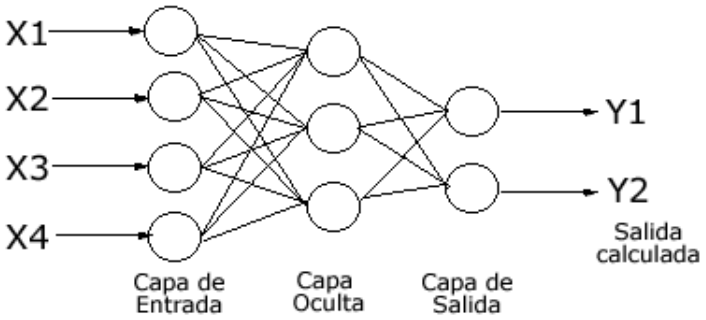


Figura 13: Esquema representatiu d'una xarxa neuronal.

El procés d'aprenentatge de la xarxa s'aconsegueix gràcies a l'algoritme de *propagació cap a endarrere* o *backpropagation*, que consisteix en aplicar les entrades a la xarxa, obtenir una sortida final en l'última capa, comparar aquest resultat amb el resultat esperat (aprenentatge supervisat) i, llavors, anar modificant *cap a endarrere* els pesos de la capa de sortida, de l'oculta i d'entrada, amb la finalitat de minimitzar (a partir del Descens de Gradient) l'error entre la sortida esperada i la sortida general per la xarxa neuronal.

Mida de les xarxes neuronals

En una xarxa multicapa de propagació cap a endavant, pot haver-hi una o més capes ocultes entre la capa d'entrada i la se sortida. La mida de les xarxes depèn del nombre de capes i del nombre de neurones ocultes en cada capa.

El nombre d'unitats ocultes està directament relacionat amb les capacitats de la xarxa. Per a que el comportament de la xarxa sigui correcte, s'ha de determinar de forma apropiada el nombre de neurones de cada capa oculta.

SVM: Support Vector Machine

Les màquines de suport vectorial o SVM (suport Vector Machines) són un conjunt d'algoritmes d'aprenentatge supervisat desenvolupats per Vladimir Vapnik i el seu equip als laboratoris AT&T.

Aquests mètodes estan pròpiament relacionats amb problemes de classificació binaria i regressió. Són molt populars en aplicacions com el processament del llenguatge natura, la parla, el reconeixement d'imatges i la visió artificial. Donat un conjunt d'exemples d'entrenament (de mostres) podem etiquetar les classes i entrenar una SVM per construir un model que predigui la classe d'una nova mostra. Intuïtivament, una SVM és un model que representa els punts de la mostra a l'espai, separant les classes a dos espais lo més amples possible mitjançant un hiperplà de separació definit com el vector entre dos punts, de les dues classes, més propers al que s'anomena vector de suport. Quan les noves mostres es posen en correspondència amb aquest model, en funció dels espais als que pertanyin poden ser classificades a una classe o a l'altra.

Més formalment, una SVM construeix un hiperplà o conjunt d'hiperplans en un espai de dimensionalitat molt alta (o inclús infinita), que pot ser utilitzat en problemes de classificació o regressió. Una bona separació entre les classe permetrà una classificació correcta.

Quin és el seu funcionament?

Com en la majoria de mètodes de classificació supervisada, les dades d'entrada (els punts) són vistos com un vector p-dimensional (una llista ordenada de p nombres). La SVM busca un hiperplà que separi de forma òptima els punts d'una classe i de l'altra, que eventualment han pogut ser prèviament projectats a un espai de dimensionalitat superior.

En aquest concepte de "separació òptima" és on rau la característica fonamental de les SVM: aquest tipus d'algoritmes busquen l'hiperplà que tingui la màxima distància (marge) amb els punts que estiguin més a prop d'ell mateix. És per això que a vegades a les SVM també se les coneix com classificadors de marge màxim. D'aquesta manera, els punts del vector que són etiquetats amb una categoria estaran a un costat de l'hiperplà i els que estiguin en l'altra categoria estaran a l'altre costat.

Els algoritmes dels SVM pertanyen a la família dels classificadors lineals. També poden ser considerats un cas especial de la regularització de Tikhonov.

Pel que fa a la nomenclatura dels SVM, se li anomena "atribut" a la variable predictora i "característica" a un atribut transformat que és utilitzat per definir l'hiperplà. L'elecció de la representació més adequada de l'univers estudiat, es realitza mitjançant un procés anomenat selecció de característiques. El vector format pels punts més propers a l'hiperplà se l'anomena vector de suport.

Els models basats en SVM estan íntimament relacionats amb les xarxes neuronals. Utilitzant la funció Kernel, resulten ser un mètode d'entrenament alternatiu per classificadors polinomials, funcions de base radial i perceptró multicapa. Com que les SVM pertanyen, doncs, als algoritmes de Machine Learning anomenats mètodes de Kernel, es coneixen també com màquines Kernel.

L'entrenament d'una màquina de vectors de suport consta de dos fases:

1. Transformar els predictors (dades d'entrada) en un espai de característiques altament dimensional. En aquesta fase és suficient amb especificar el Kernel; les dades mai es transformen explícitament a l'espai de característiques.
2. Resoldre un problema d'optimització quadràtica que s'ajusti a un hiperplà òptim de característiques transformades en dos classes. El nombre de característiques transformades està determinat pel nombre de vectors de suport.

Per construir la superfície de decisió només es requereixen els vectors de suport seleccionats de les dades d'entrenament. Un cop entrenades, la resta de dades d'entrenament són irrellevants.

A continuació es mostren tres exemples de les transformacions de Kernel més habituals:

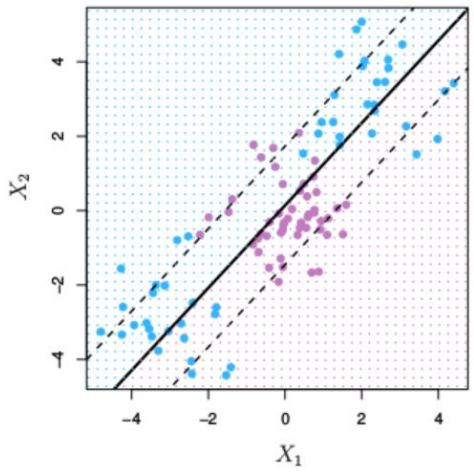


Figura 14: Representació gràfica d'una transformació de Kernel lineal.

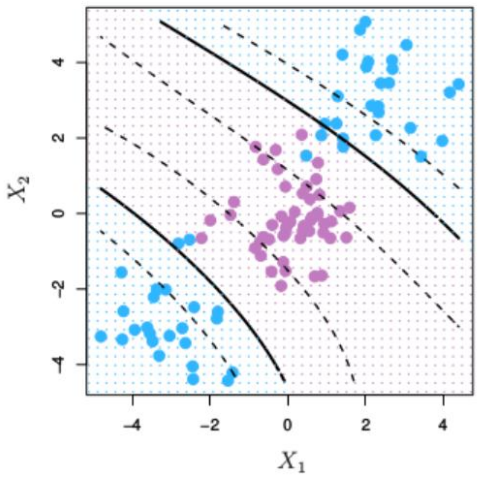


Figura 15: Representació gràfica d'una transformació de Kernel polinòmic de grau 3.

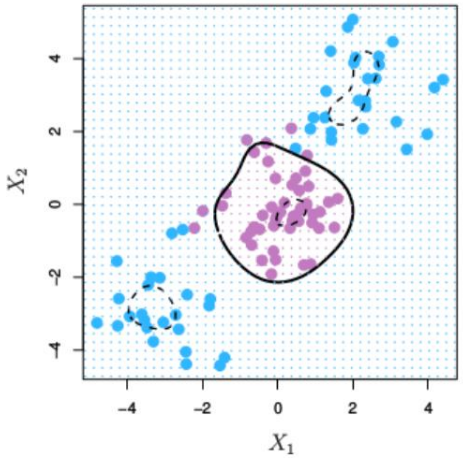


Figura 16: Representació gràfica d'una transformació de Kernel radial.

Aplicació dels mètodes i resultats

Aquest apartat inclou la implementació dels cinc models comentats amb les variables també esmentades. Després de realitzar la partició de les dades en dos subconjunts, el d'entrenament (*train set*), que engloba el 70% de les observacions, i el de proba (*test set*), que inclou el 30% restant, la mida de cada subconjunt és la següent:

- Subconjunt d'entrenament: 144 798 observacions
- Subconjunt de proba: 62 014 observacions

La taula següent mostra la proporció d'observacions amb la pòlissa anul·lada i amb la pòlissa vigent del conjunt de dades d'entrenament.

	0	1
Freqüència	17830	126968
Proporció	0.123	0.877

Taula 15: Freqüència i proporció de la quantitat de pòlisses anul·lades i vigents.

Per tal de valorar la qualitat de la predicció de cada model es mostraran algunes mesures de rendiment per tal de que puguin ser comparats posteriorment. Les mesures tractades seran les següents:

1. **Corba ROC** ("*Receiver Operating Characteristic*")

La corba ROC (corba de característica operativa del recepto) és un gràfic que mostra el rendiment d'un model de classificació en tots els llindars de classificació. Aquesta corba presenta dos paràmetres:

- Taxa de vertaders positius (VP)
- Taxa de falsos positius (FP)

La taxa de vertaders positius (TPR) és sinònim d'exhaustivitat i, per tant, es defineix de la següent forma: $TPR = \frac{VP}{VP+FN}$ on FN són els falsos negatius.

La taxa de falsos positius (FPR) es defineix: $FPR = \frac{FP}{FP+VN}$ on VN són els vertaders negatius.

Una corba ROC representa TPR en front a FPR en diferents llindars de classificació. Reduir el llindar de classificació classifica més elements com positius, per tant, augmentaran tant els falsos positiu com els vertaders positius. En la següent figura es mostra la corba ROC típica:

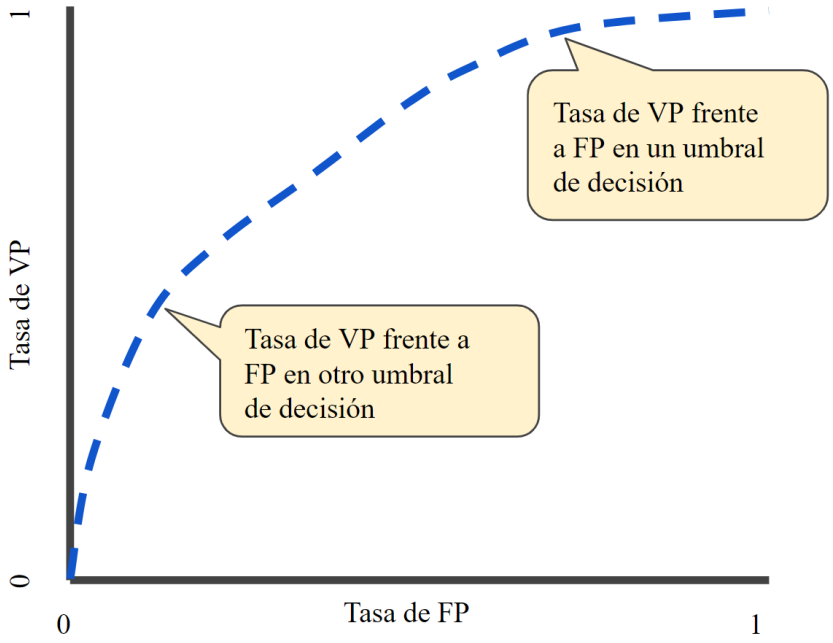


Figura 17: Taxa de VP en front a FP en diferents llindars de classificació.

Informació que conté la corba ROC:

- Si la proba o model fos perfecte, és a dir, estigués el màxim allunyada possible de la diagonal, vol dir que hi ha una regió en la que qualsevol punt de tall té sensibilitat i especificitat iguals a 1: la corba només té el punt (0,1).
- Si la proba o model coincideix amb la diagonal, el contrari que en el cas anterior, indica que la sensibilitat (vertaders positius) és igual a la proporció de falsos negatius. La corba coincidiria amb la diagonal de (0,0) a (1,1).

El més habitual és que la corba es trobi en un punt intermedi.

Com a limitació de la corba ROC, cal destacar que només contempla dos situacions possibles (pòlissa nul·la, pòlissa vigent) i no serveix per situacions on s'ha de discernir entre més de dos possibilitats.

2. AUC (*"Area Under the ROC Curve"*)

Per calcular els punts en la corba ROC, podríem avaluar un model de regressió logística repetidament amb diferents llindars de classificació, per seria ineficient. L'AUC és un algoritme eficient basat en l'ordenament que proporciona aquesta informació.

AUC significa "àrea sota la corba ROC". Això significa que l'AUC mesura tota l'àrea bidimensional per sota de la corba ROC completa de (0,0) a (1,1).

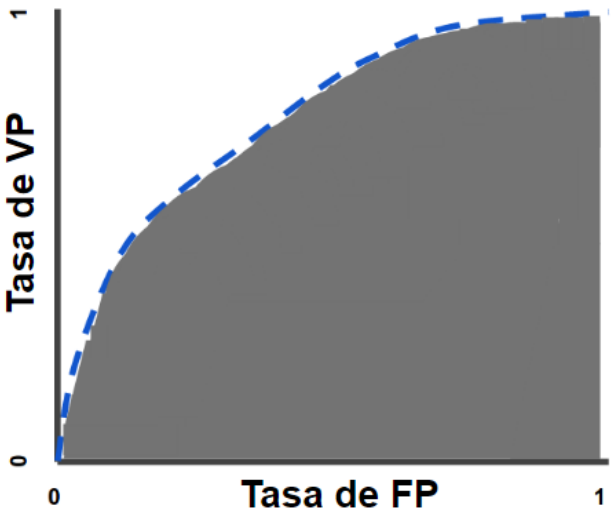


Figura 18: AUC (àrea dota la corba ROC).

L'AUC oscil·la entre valors del 0 a l'1. Un model on les prediccions són un 100% incorrectes té un AUC de 0.0; un altre amb unes prediccions 100% correctes té un AUC de 1.0.

Avantatges de l'AUC:

- És invariable respecte a l'escala. Mesura quant de bé que es classifiquen les prediccions, enlloc dels valors absoluts.
- És invariable respecte el llindar de classificació. Mesura la qualitat de les prediccions del model, sense tenir en compte quin llindar de classificació s'escull.

3. **MSE** ("*Mean Squared Error*")

L'Error quadràtic mig mesura la quantitat d'error que hi ha entre dos conjunts de dades, és a dir, compara un valor predit amb un valor observat o conegut.

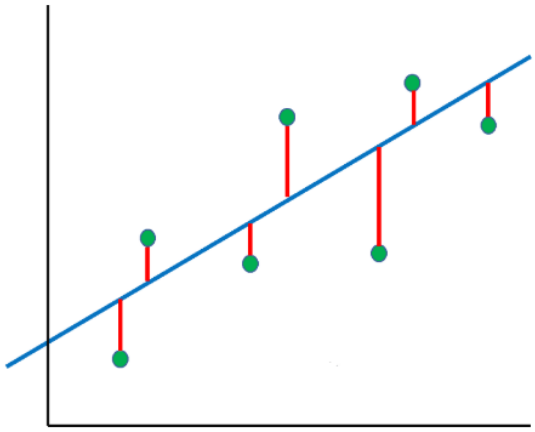


Figura 19: Error en regressió lineal.

La figura anterior mostra com s'utilitza una regressió lineal (en blau) per estimar les dades que es tenen (en verd. El model lineal té un error (en vermell) que es pot definir amb la següent fórmula: $error\ quadràtic = (valor\ observat - valor\ predit)^2$. El valor estimat és el valor que ens dóna el model, en aquest cas la línia blava. L'error es calcula al quadrat perquè aquest sempre sigui positiu, d'aquesta manera se sap que l'error perfecte és 0.

4. Biaix

El biaix és la diferència entra la predicció esperada del nostre model i els valors vertaders. Es podria dir que el biaix és l'error que introduïm en el model a l'intentar explicar un problema del món real, al qual li correspondria un model molt complicat, amb un model bastant més senzill. És a dir, el biaix pot interpretar-se com en el quant generalitzem el nostre model. En general, els models més flexibles, menys generalitzats i més complexes, impliquen menys biaix. El biaix, doncs, no és conseqüència de les nostres dades sinó del model escollit.

5. Matrius de confusió

Per avaluar la predicció de pertinença o no a un grup concret (pòlisses anul·lades o vigents) el que s'utilitza és la matriu de confusió. Aquesta, és una eina fonamental per la valoració predictiva dels models de classificació. Consta d'una taula de doble entrada que conté els valors els valors predits en les files i els reals en les columnes. La matriu de confusió té la següent forma:

	Valor real = 0	Valor real = 1
Predicció = 0 (FALS)	TN	FN
Predicció = 1 (VERTADER)	FP	TP

Taula 16: Esquema d'una matriu de confusió.

On, com ja s'ha esmentat anteriorment:

- TP: vertaders positius *("True Positive* ")
- TN: vertaders negatius *("True Negative*")
- FP: falsos positius *("False Positive*")
- FN: falsos negatius *("False Negative*")

A partir d'aquests valors es poden calcular les següents ràtios:

- Precisió $= (TP + TN)/(P + N)$
- Exactitud $= TP/(TP + FP)$
- Sensibilitat $= TP/P$
- Especificitat $= TN/N$

On P i N són el nombre de positius i negatius del conjunt de dades.

Logit

A l'aplicar a les nostres dades una regressió logística s'obtenen els següents resultats:

Sobre la desviació residual s'obté:

Mínim	1r quartil	Mediana	3r quartil	Màxim
-4.2423	0.3980	0.4708	0.5473	1.8320

Taula 17: Resultats de la desviació residual del model Logit.

La desviació nul·la obtinguda és 108 057 amb 144 797 graus de llibertat. La desviació residual és de 104 969 amb 144 780 graus de llibertat. I, per últim, l'AIC ("*Akaike Information Criterion*") que s'obté és de 105 005.

La matriu de confusió pel conjunt de dades de proba (*test set*) és:

Predicció	Anul·lades	Vigents	Sumatori
0	4660	21999	26659
1	2957	32398	35355
Sumatori	7617	54397	62014

Taula 18: Matriu de confusió en freqüències pel testset obtingut mitjançant el model Logit.

Predicció	Anul·lades	Vigents	Sumatori
0	0.075144	0.354742	0.429887
1	0.047683	0.522430	0.570113
Sumatori	0.122827	0.877173	1.000000

Taula 19: Matriu de confusió en proporcions pel testset obtingut mitjançant el model Logit.

Precisió	Exactitud	Sensibilitat	Especificitat
59,76%	91,64%	59,56%	61,18%

Taula 20: Taula dels ràtios obtinguts mitjançant la regressió logística.

S'observa que la capacitat predictiva del model Logit no és gaire alta, tot i que cal destacar l'elevada exactitud que presenta el model. Per altra banda, pel que fa al percentatge d'encerts per les pòlisses que es renoven (sensibilitat) i per les que no ho fan (especificitat), els resultats estan molt equilibrats.

A continuació es mostra la corba ROC.

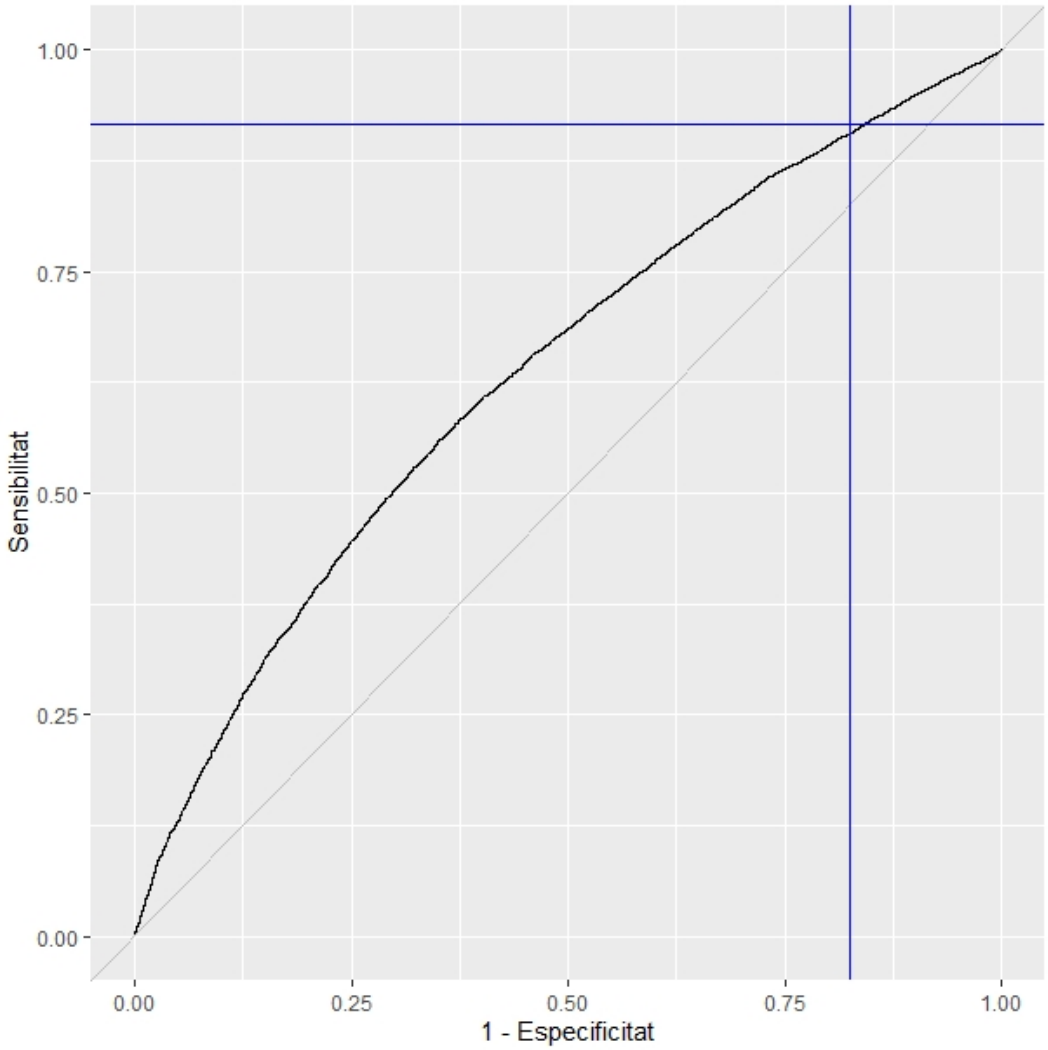


Figura 20: Corba ROC pel model Logit.

Els resultats es mostren en la següent taula:

Conjunt d'entrenament (*train set*)	Conjunt de prova (*test set*)
AUC	
0.6335	0.6357
MSE	
0.1059	0.1055
Biaix	
-1.090483e-11	-9.835344e-05

Taula 21: Taula amb certes mesures de rendiment obtingudes mitjançant la regressió logística.

Tal com demostra el valor de l'AUC i s'observa en la corba ROC, l'àrea sota la corba és d'un 64% aproximadament. Com més elevat és aquest valor, més prediccions correctes presenta el model. Un 64% no és un valor especialment bo però suficient per poder predir el model amb certa confiança d'encerts.

Pel que fa al biaix, en general, els algoritmes paramètrics com la regressió lineal, tenen un biaix elevat que els fa ràpids d'aprendre i més fàcil d'entendre, però generalment, també menys flexibles. A la vegada, tenen un rendiment predictiu menor en problemes més complexos. Al contrari del que s'acostuma a esperar d'un model de regressió lineal, el biaix obtingut és baix. Aquest fet pot ser conseqüència de la grandària de la base de dades utilitzada. En el nostre cas, contràriament al que podria esperar-se, s'observa un error negatiu, que es considerarà com a 0, que representa que el model gairebé no presenta errors.

L'error quadràtic mitjà com més proper a 0 sigui millor. Un MSE de 0.1055 és un valor reduït i per tant, correcte.

Per últim, es calcula la bondat de l'ajust mitjançant un anàlisi de la variància, l'anomenat test Anova. Aquest, compara el model reduït, que només inclou l'*intercept* com a variable explicativa, amb el model complet.

$$H_o = \beta_1 = \beta_2 = \cdots = \beta_{13} = 0$$

$$H_1 = altrament$$

És a dir, la hipòtesi nul·la (H_o) del test Anova estableix que totes les mitjanes del conjunt de dades són iguals mentre que la hipòtesi alternativa (H_1) estableix que almenys una és diferent.

Al realitzar el test amb les nostres dades s'obté un p-valor inferior a 2.2e-16. Al ser clarament menor que 0.05 que és el nostre nivell de significació, es rebutja la hipòtesi nul·la i es conclou que el model complet és estadísticament significatiu respecte el model reduït.

Probit

A l'aplicar a les nostres dades una regressió logística s'obtenen els següents resultats:

Sobre la desviació residual s'obté:

Mínim	1r quartil	Mediana	3r quartil	Màxim
-4.9679	0.3952	0.4708	0.5483	1.8499

Taula 22: Resultats de la desviació residual del model Probit.

La desviació nul·la obtinguda és, igual que l'obtinguda en la regressió logística, 108 057 amb 144 797 graus de llibertat. La desviació residual és de 104 943 amb 144 780 graus de llibertat. I, per últim, l'AIC que s'obté és de 104 979.

La matriu de confusió per les dades del test de proba (*test set*) és:

Predicció	Anul·lades	Vigents	Sumatori
0	4660	21999	26659
1	2957	32398	35355
Sumatori	7617	54397	62014

Taula 23: Matriu de confusió en freqüències pel testset obtingut mitjançant el model Probit.

Predicció	Anul·lades	Vigents	Sumatori
0	0.075144	0.354742	0.429887
1	0.047683	0.522430	0.570113
Sumatori	0.122827	0.877173	1.000000

Taula 24: Matriu de confusió en proporcions pel testset obtingut mitjançant el model Probit.

Pel que fa a les pòlisses predites correctament, la matriu de confusió anterior mostra només un 52% en les pòlisses actives i un 7.5% per les pòlisses anul·lades.

Precisió	Exactitud	Sensibilitat	Especificitat
59.76%	91.64%	59.56%	61.18%

Taula 25: Taula dels ràtios obtinguts mitjançant el model Probit.

El model Probit presenta uns resultats molt similars als del Logit. Per una banda, la precisió és igual de baixa i l'exactitud igual d'elevada. També s'observa equilibri entre la proporció d'encerts per les pòlisses que es renoven i per les que no ho fan.

A continuació es mostra la corba ROC.

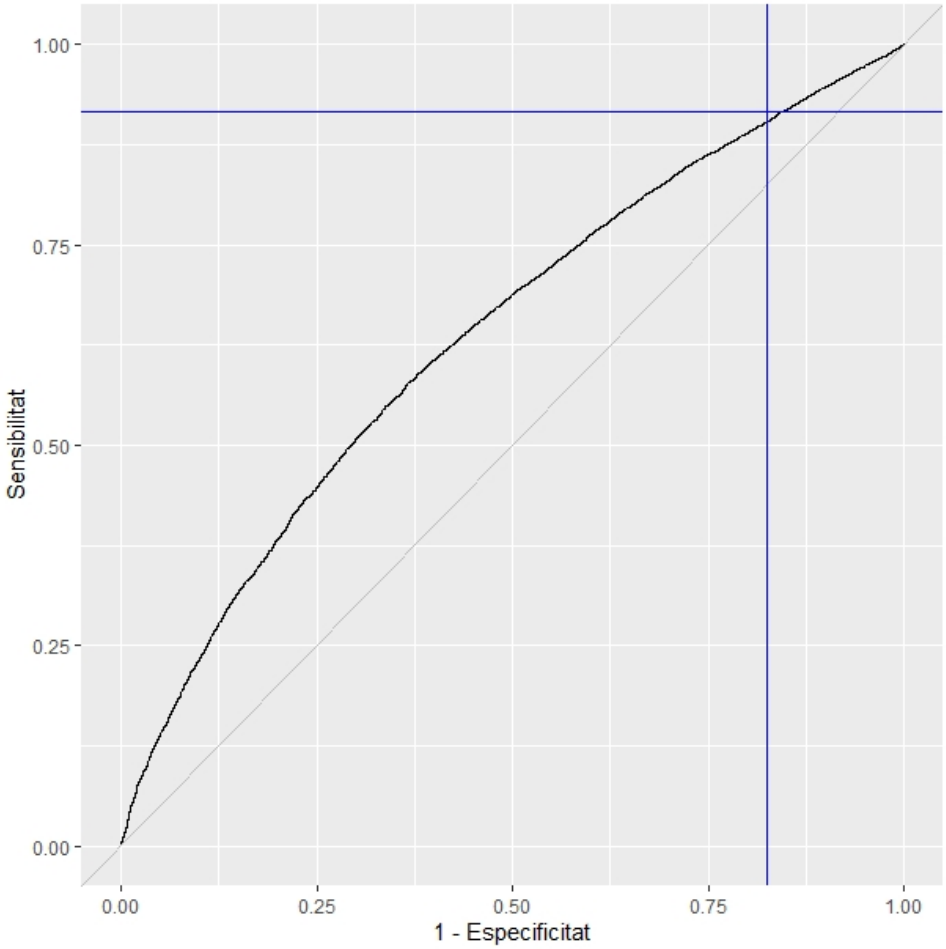


Figura 21: Corba ROC pel model Prodit.

Conjunt d'entrenament (*train set*)	Conjunt de prova (*test set*)
AUC	
0.6345	0.6370
MSE	
0.1058	0.1055
Biaix	
8.1790e-05	-1.8269e-05

Taula 26: Taula amb certes mesures de rendiment obtingudes mitjançant el model Probit.

La corba ROC és pràcticament idèntica a la del model Logit, amb un AUC del gairebé 64%. El biaix també és gairebé nul, cosa que indica que el model explica adequadament el problema real plantejat. L'error quadràtic mitjà com més proper a 0 sigui millor. Un MSE de 0.1055 és un valor reduït i per tant, correcte.

Per últim, al calcular la bondat de l'ajust mitjançant el test Anova. Amb el mateix test d'hipòtesis utilitzat amb el model Logit:

$$H_o = \beta_1 = \beta_2 = \cdots = \beta_{13} = 0$$

$$H_1 = altrament$$

S'obté un p-valor de 2.2e-16, inferior a 0.05, i per tant, es rebutja la hipòtesi nul·la i es conclou que el model complet és estadísticament significatiu respecte el model reduït.

Arbres de decisió condicional

La matriu de confusió per les dades del test de proba (test set) és:

Predicció	Anul·lades	Vigents	Sumatori
0	6355	30074	36429
1	1262	24323	25585
Sumatori	7617	54397	62014

Taula 27: Matriu de confusió en freqüències pel testset obtingut mitjançant l'arbre de decisió condicional.

Predicció	Anul·lades	Vigents	Sumatori
0	0.102477	0.484955	0.587432
1	0.020350	0.392218	0.412568
Sumatori	0.122827	0.877173	1.000000

Taula 28: Matriu de confusió en proporcions pel testset obtingut mitjançant l'arbre de decisió condicional.

La matriu anterior mostra que correctament només s'ha predit un 39% de les pòlisses actives i un 10% de les anul·lades.

Precisió	Exactitud	Sensibilitat	Especificitat
49,47%	95,07%	44,71%	83,43%

Taula 29: Taula dels ràtios obtinguts mitjançant l'arbre de decisió condicional.

S'observa en la taula anterior que l'arbre d'inferència condicional té una capacitat predictiva inferior als dos models anteriors, tot i que alhora obté uns resultats més exactes. Tot i haver desequilibris en entre els encerts per les pòlisses que es renoven i per les que no ho fan, aquest és menor que en el model Probit.

A continuació es mostra la corba ROC.

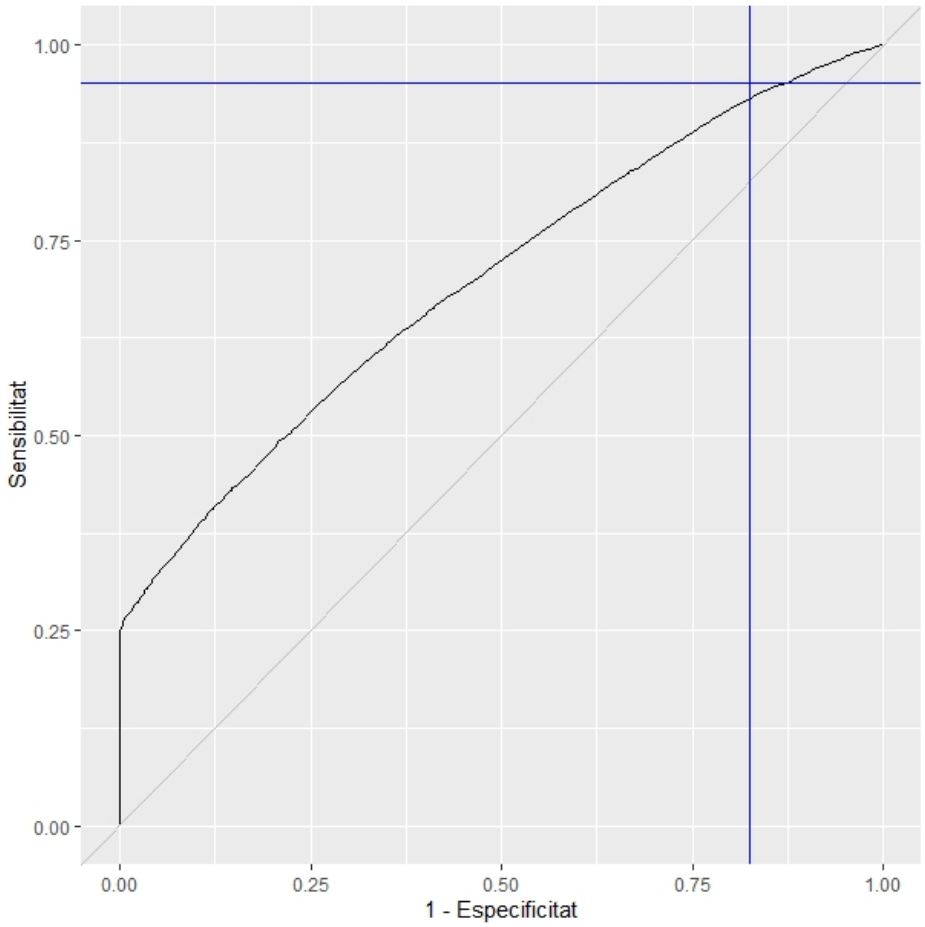


Figura 22: Corba ROC per l'arbre de decisió condicional.

Conjunt d'entrenament (*train set*)	Conjunt de prova (*test set*)
AUC	
0.7195	0.6988
MSE	
0.1007	0.1019
Biaix	
0	-0.0001561

Taula 30: Taula amb certes mesures de rendiment obtingudes amb un arbre condicional.

En aquest cas, l'àrea sota la corba ROC és una mica més elevada, amb un 70% aproximadament. Això indica que l'arbre de decisió condicional ha predit correctament més valors que en els models anteriors. El biaix però no és tan reduït com en el Logit i el Probit però, tot i així, és molt proper a 0 i s'accepta com a bo. L'MSE també és reduït.

Per últim, es mostra l'arbre de decisió obtingut amb les dades de la llar. L'arbre és molt poc comprensible i poc entenedor, però deixa entreveure la magnitud de la complexitat del problema.

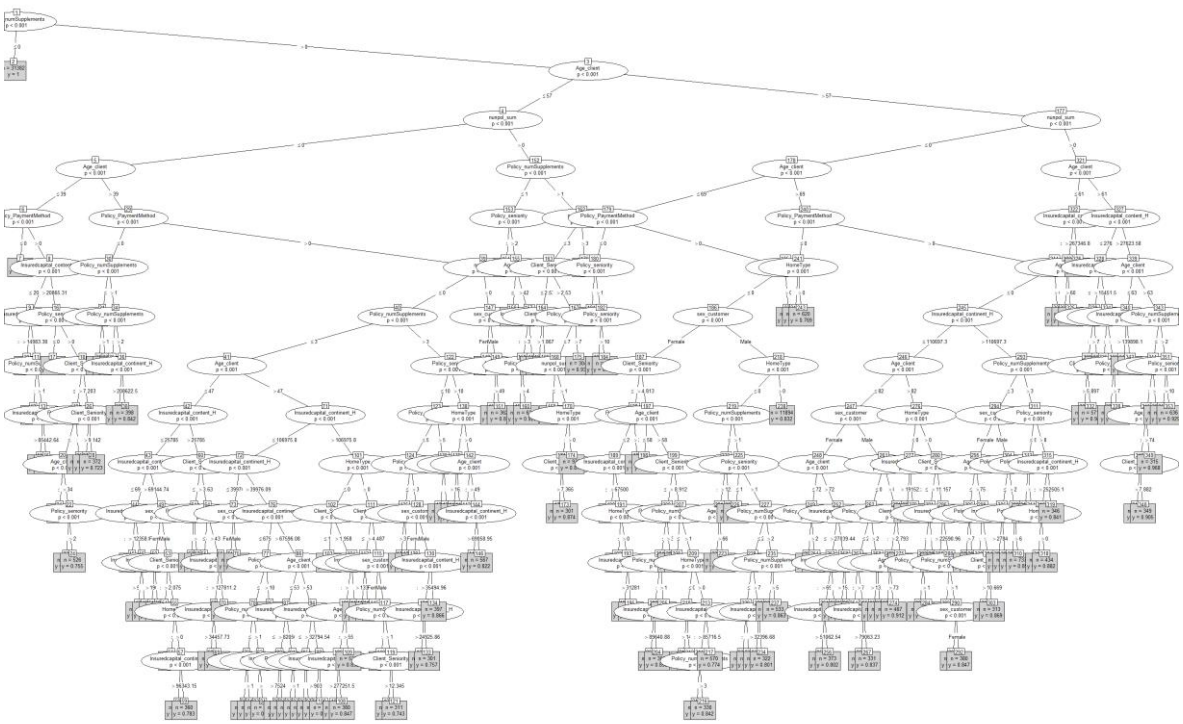


Figura 23: Representació gràfica de l'arbre de decisió condicional.

Xarxes neuronals

La matriu de confusió obtinguda mitjançant una xarxa neuronal, pel subconjunt de dades de proba (*test set*) és:

Predicció	Anul·lades	Vigents	Sumatori
0	6482	45834	52316
1	1135	8563	9698
Sumatori	7617	54397	62014

Taula 31: Matriu de confusió en freqüències pel testset obtingut mitjançant la xarxa neuronal.

Predicció	Anul·lades	Vigents	Sumatori
0	0.104525	0.739091	0.843616
1	0.018302	0.138082	0.156384
Sumatori	0.122827	0.877173	1.000000

Taula 32: Matriu de confusió en proporcions pel testset obtingut mitjançant la xarxa neuronal.

En aquesta matriu de confusió destaca l'alta proporció de falsos negatius, amb un valor de 0.73. En canvi de de vertaders positius només n'ha predit un 13%.

Precisió	Exactitud	Sensibilitat	Especificitat
24,26%	88,30%	15,74%	85,10%

Taula 33: Taula dels ràtios obtinguts mitjançant la xarxa neuronal.

La taula mostra un valor de precisió molt baix, encara que l'exactitud és bastant elevada. Pel que fa a la sensibilitat i l'especificitat, hi ha bastant desequilibri entre ambdues però es pot concloure que aquest model pot ser de molta utilitat si el que es vol és detectar aquells clients que no renovaran la pòlissa.

A continuació es mostra la corba ROC.

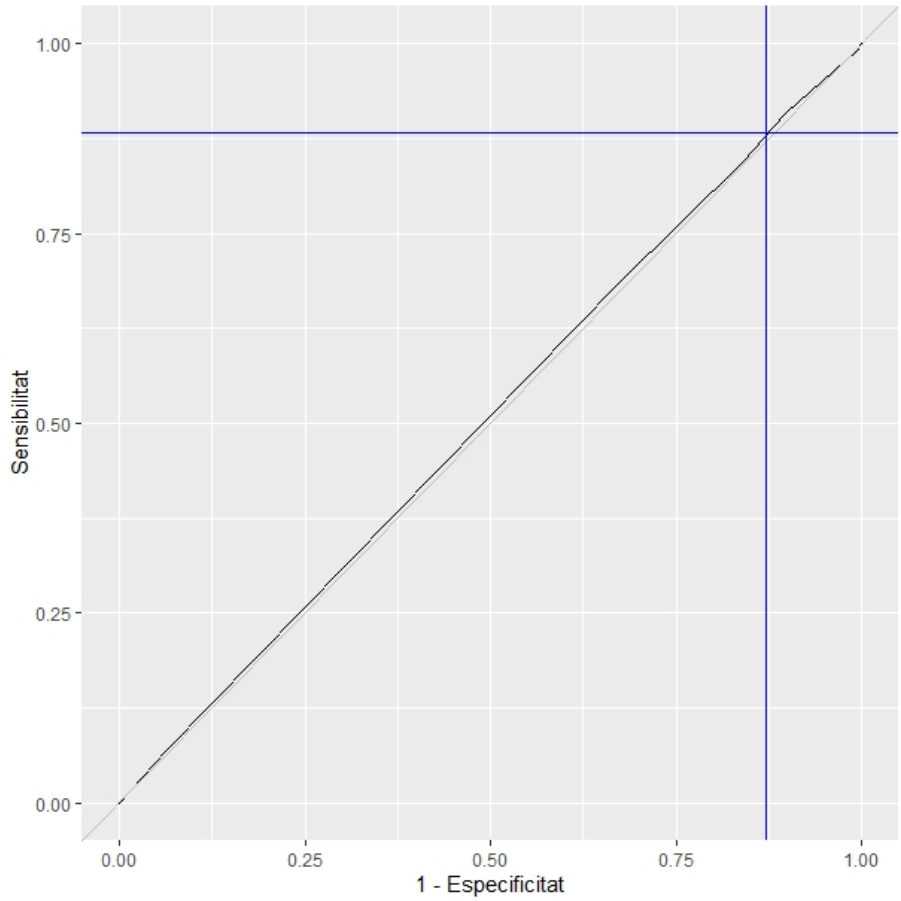


Figura 24: Corba ROC per la xarxa neuronal.

Conjunt d'entrenament (*train set*)	Conjunt de prova (*test set*)
AUC	
0.5073	0.5044
MSE	
0.1079	0.1077
Biaix	
-7.9032e-05	-0.0004181

Taula 34: Taula amb certes mesures de rendiment obtingudes amb una xarxa neuronal.

Tal com es veu en el gràfic, la corba coincideix amb la diagonal de (0,0) a (1,1), per tant, l'àrea sota la corba ROC és del 50%. Això indica que la sensibilitat (vertaders positius) és igual a la proporció de falsos negatius. No és un valor gaire bo. El biaix, però, és molt reduït, igual que en el model anterior. També ho és l'error quadràtic mig. ,

Per acabar, es mostra la gràfica de la xarxa neuronal completa, creada utilitzant 10 nodes intermedis per avaluar la seva capacitat predictiva amb un total de 201 pesos. Com que les dades no estan balancejades, és a dir, hi ha un elevat percentatge de renovacions (més del 87%) tenim en compte aquesta circumstància a l'hora d'establir el criteri de predicció de la categoria de la variable resposta.

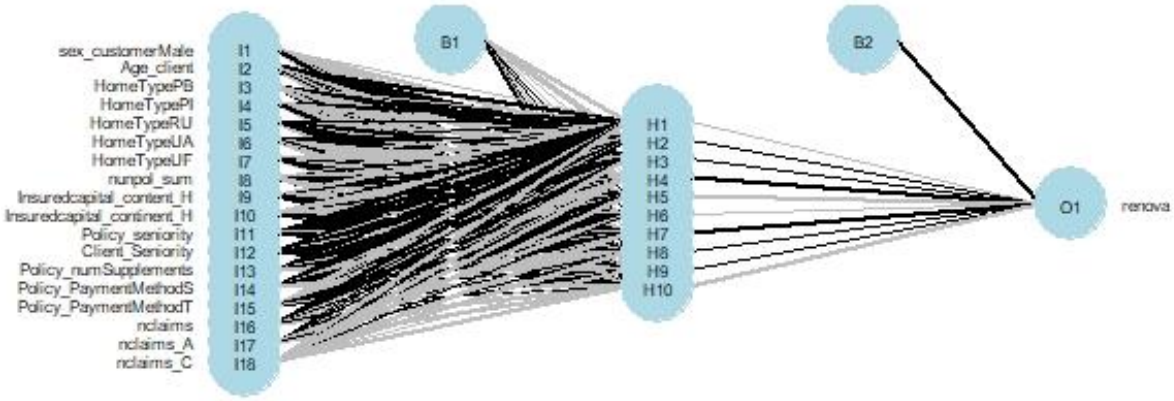


Figura 25: Representació gràfica de la xarxa neuronal.

Svm

La matriu de confusió obtinguda mitjançant la màquina de suport vectorial, pel subconjunt de dades de proba (*test set*) és:

Predicció	Anul·lades	Vigents	Sumatori
0	0	5	5
1	7284	54725	62009
Sumatori	7284	54730	62014

Taula 35: Matriu de confusió en freqüències pel testset obtingut mitjançant l'svm.

Predicció	Anul·lades	Vigents	Sumatori
0	0	0.000083	0.000083
1	0.117462	0.882455	0.999917
Sumatori	0.117462	0.882538	1.000000

Taula 35: Matriu de confusió en proporcions pel testset obtingut mitjançant l'svm.

Cal destacar la nul·la presència de vertaders positius però, en canvi, l'elevat percentatge de vertaders positius, un 88%. S'observa en la matriu que l'svm ha predit un 99% d'1, és a dir, de pòlisses actives.

Precisió	Exactitud	Sensibilitat	Especificitat
88.25%	88.25%	99.99%	0%

Taula 37: Taula dels ràtios obtinguts mitjançant l'svm.

La taula mostra uns valors de precisió i d'exactitud bastant elevats. Pel que fa a la sensibilitat i l'especificitat, hi ha total desequilibri entre ambdues però es pot concloure que aquest model pot ser de molta utilitat si el que es vol és detectar aquells clients que sí que renovaran la pòlissa.

A continuació es mostra la corba ROC.

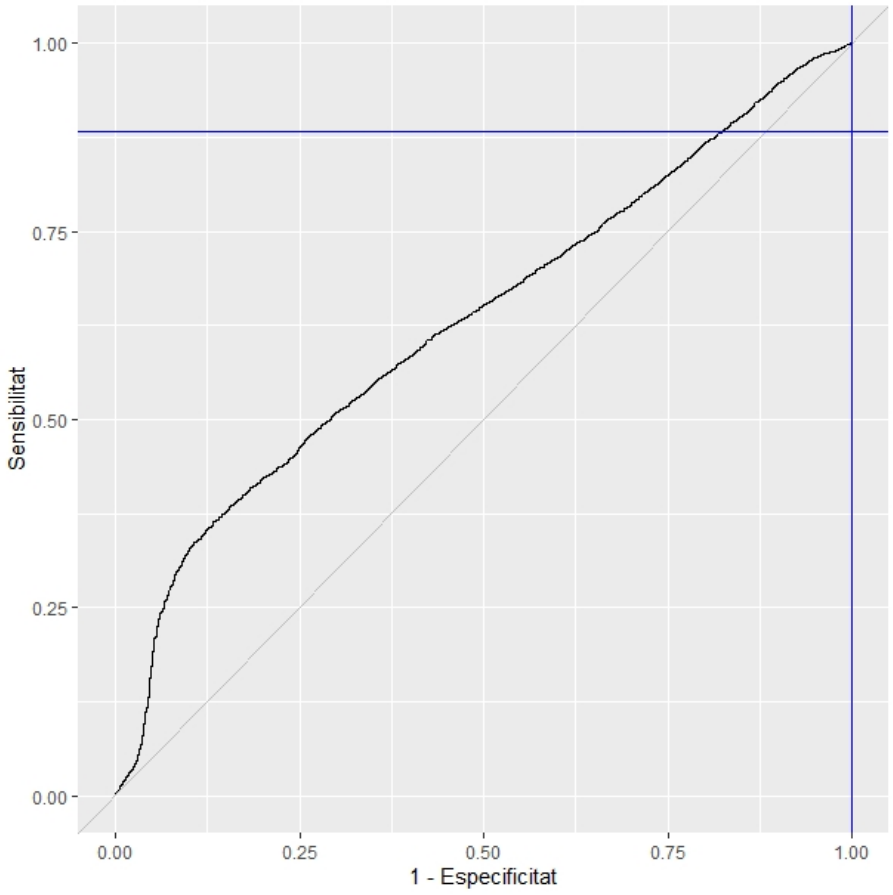


Figura 24: Corba ROC per l'svm.

Conjunt d'entrenament (*train set*)	Conjunt de prova (*test set*)
AUC	
0.7046	0.6307
MSE	
0.111391	0.110893
Biaix	
0.086007	0.0853324

Taula 38: Taula amb certes mesures de rendiment obtingudes amb l'svm.

Observant el gràfic i els valors de la taula, notem que l'àrea sota la corba ROC és del 70%, un valor millor que l'obtingut amb la xarxa neuronal. El biaix i l'error quadràtic mig són baixos, s'accepten com a bons.

Avaluació capacitat predictiva i comparació de resultats

En aquest apartat es pretén fer una comparació de les matrius de confusió obtingudes en cadascun dels cinc models i analitzar el percentatge d'encerts per veure quin model fa una millor predicció. També es realitzaran tres ensemblatges per veure si, aquests, milloren la predicció dels models de forma individual.

Per tal de poder ensemblar els quatre models, un cop obtingudes les probabilitats per cada model, el següent pas és predir si el client renova la pòlissa (variable renova = 1) o no (renova=0). Per fer-ho, s'ha fixat un llindar de probabilitat per tal de classificar la probabilitat en funció de si està per sobre o per sota d'aquest llindar. El llindar o límit coincideix amb la freqüència d'1 de la variable renova, i el seu valor és de 0.877. A continuació, es duran a terme tres mètodes d'ensemble diferents.

Ensemble 1

El primer consisteix en assignar un 1 o un 0 en la predicció final del model ensemblat en funció de la freqüència de 1 i 0 entre tots els models individuals. Es mostra en la següent taula un exemple conceptual del funcionament d'aquest ensemble.

Logit	Probit	Arbre condicional	Xarxa neuronal	SVM	Renova=1	Renova=0	Predicció final
0	0	0	0	1	1	4	0
1	1	1	1	0	4	1	1
1	1	1	0	0	3	2	1
1	1	0	0	0	2	3	0
1	0	1	0	1	3	2	1

Taula 39: Exemple conceptual de l'ensemble.

Aquest mètode fa l'ensemble de la següent manera: si la quantitat d'1 predits per cadascun dels models és major a la quantitat de 0, la predicció final serà 1, i a la inversa; si la quantitat de 0 predits és major que la quantitat d'1, el model final predirà 0. Com que en l'ensemble s'utilitzen cinc models individuals, no pot haver-hi empats.

Aquestes són les freqüències i les proporcions obtingudes mitjançant aquest mètode:

Predicció	Anul·lades	Vigents
0	4617	21208
1	3000	33189

Predicció	Anul·lades	Vigents
0	0.074451	0.341987
1	0.048376	0.535185

Taules 40 i 41: Matriu de confusió en freqüències i en proporcions de l'ensemble 1.

El percentatge de vertaders positius és el més elevat mentre que el de vertaders negatius és bastant baix. Això pot ser provocat pel fet que les dades no estan balancejades, és a dir, hi ha un gran percentatge de renovacions (més del 87%).

Ensemble 2

El segon mètode d'ensemble que s'utilitza es basa en la priorització d'encerts dels 1 (ensemble 2.1) i en la priorització dels 0 (ensemble 2.2). És a dir, el primer ensemble pretén prioritzar l'encert de pòlisses actives i el segon, el de les pòlisses anul·lades. El criteri que utilitza el mètode d'ensemble 2.1 és que només que un model individual hagi predit un 1, la predicció del model ensemblat serà igual a 1. Perquè la predicció de l'ensemble sigui igual a 0 tots els models haurien d'obtenir una predicció igual a 0.

A continuació es mostren les matrius de confusió obtingudes:

Predicció	Anul·lades	Vigents
0	0	3
1	7617	54394

Predicció	Anul·lades	Vigents
0	0	0.00004838
1	0.122827	0.877124

Taules 42 i 43: Matriu de confusió en freqüències i en proporcions de l'ensemble 2.1.

Les matrius de confusió mostren que el 99.9% (0.122827+0.877124) de les prediccions de l'ensemble són iguals a 1. Només el 0.0048% han predit un 0 com a resultat i, tot i així, aquest no ha estat encertat ja que a la realitat aquestes pòlisses estaven actives.

Pel que fa a l'ensemble 2.2, aquest funciona de la següent manera: només que hi hagi un model que predigui que la pòlissa estarà anul·lada (=0), el model ensemblat predirà també un 0. Perquè el model ensemblat predigui un 1 és necessari que tots els models individuals prediguin un 1. A continuació es mostren les matrius de confusió, en freqüència i en proporció.

Predicció	Anul·lades	Vigents
0	7455	51350
1	162	3047

Predicció	Anul·lades	Vigents
0	0.120215	0.828039
1	0.002612	0.049134

Taules 44 i 45: Matriu de confusió en freqüències i en proporcions de l'ensemble 2.2.

Al contrari del que passa en l'ensemble anterior, en aquest cas el 95% de les prediccions de l'ensemble són iguals a 0, encara que només un 12% hagi encertat la predicció.

Utilitzar mètodes d'ensemble amb prevalença de 0 o 1 dependrà de l'objectiu que tingui l'empresa a l'hora de realitzar l'estudi. Ambdós models, el que prioritza els 0 i el que prioritza els 1 són totalment vàlids però cadascun d'ells serà útil en una situació concreta, depenent de si l'empresa vol centrar-se en el clients que seguiran amb la pòlissa o en els clients que tenen pensat no renovar.

Ensemble 3

Per últim, la forma d'ensemblar els quatre models consistirà en calcular la probabilitat mitjana y en aplicar-li el criteri de 0 o 1 en funció de si la probabilitat mitjana està per sobre o per sota del llindar de probabilitat fixat anteriorment, que 0.877.

Tot seguit, es mostren les matrius de confusió obtingudes amb els resultats de l'ensemble:

Predicció	Anul·lades	Vigents
0	3364	12937
1	4253	41460

Taules 46 i 47: Matriu de confusió en freqüències i en proporcions de l'ensemble 3.

La matriu de confusió calculada en proporcions mostra un 5.4% de vertaders negatius i un 66.86% de vertaders positius.

Per acabar, es procedeix a fer una taula comparativa per la proporció d'encerts tan dels 0 ("*True Negative*") com dels 1 ("*True positive*") tan pels quatre models individuals com pels tres models d'ensemble.

Model	Proporció d'encerts de 0	Proporció d'encerts d'1
Logit	0.075144	0.522430
Probit	0.075144	0.522430
Arbre de decisió condicional	0.102477	0.392218
Xarxa neuronal	0.104525	0.138082
SVM	0	0.882455
Ensemble 1: Màxim	0.074451	0.535185
Ensemble 2.1: Prevalença dels 1	0	0.877124
Ensemble 2.2: Prevalença dels 0	0.120215	0.049134
Ensemble 3: Mitjana	0.054246	0.668559

Taula 48: Proporció d'encerts pels diferents models i mètodes d'ensemblatge.

Pel que fa a la proporció d'encerts de les pòlisses que seran renovades, el model que en fa una millor predicció és el de les màquines de suport vectorial (svm), amb un 88% d'encerts. Tot i així, cal destacar que el model d'ensemble 2.1 obté un valor similar. Ambdós valors eren previsibles ja que tan l'svm com l'ensemble 2.1 han obtingut unes prediccions majoritàriament iguals a 1, és per això que han encertat més pòlisses actives que la resta de models. Les probabilitats més baixes s'han obtingut amb la xarxa neuronal i l'arbre de decisió condicional, ambdós inferiors als 50% d'encerts. L'ensemble 2.2 és el que ha obtingut la proporció d'encerts més baixa perquè el seu criteri prioritza la predicció de pòlisses anul·lades. En aquest cas podem dir que els mètodes d'ensemble 1, 2.1 i 3, han millorat les prediccions de l'arbre de decisió condicional i de la xarxa neuronal, però no han millorat el de l'svm.

Per altra banda, si es comparen les probabilitats d'encert dels 0, el model que n'ha fet una millor predicció és, com podia suposar-se, el de l'ensemble 2.2, que dóna prioritat al fet de no renovar la pòlissa. Tot i així, el valor obtingut per aquest model és baix, d'un 12%. Això és conseqüència de que la majoria de les pòlisses són actives i per tant, el model ha predit molts 0 que en realitat haurien de ser 1. De fet el percentatge de falsos negatius és de gairebé el 83%. L'arbre de deicisó condicional i la xarxa neuronal, però, han obtingut un percentatge d'encerts de 0 molt similar al de l'ensemble 2.2, tot i que aquest segueixi sent baix. Cal destacar que l'svm i l'ensemble 2.1 no han predit correctament cap pòlissa anul·lada, conseqüència del que s'ha comentat anteriorment, que és que ambdós models han predit majoritàriament valors iguals a 1. Es podria dir en aquest aspecte que el mètode d'ensemble han la predicció dels models Logit, Probit i svm. Els mètodes d'ensemble restants han obtingut probabilitats molt baixes, no han millorat la capacitat predictiva dels models individuals.

Conclusió

Mitjançant aquest estudi, es pretenia principalment obtenir el millor mètode per predir, a partir d'una base de dades d'assegurances de la llar, si el client en qüestió renovaria o no la pòlissa que té contractada. Per fer-ho, s'ha fet ús de la quantitat d'informació que hi ha disponible avui en dia, l'anomenat Big-data, que com ja s'ha dit anteriorment, no només és necessari tenir-la sinó que fa falta tractar-la perquè arribi a tenir una utilitat. En aquest cas, mitjançant la modelització de cinc mètodes predictius, s'han aconseguit predir les dades que es buscaven amb resultats més o menys bons. A més, s'han realitzat tres mètodes d'ensemble per tal de millorar les prediccions obtingudes mitjançant els models individuals. La pregunta que es plantejava abans de fer l'anàlisi era: podran aquests models d'ensemble millorar la capacitat predictiva de la resta de models? Com ho faran? Si no és així, quins són els millors models per predir la nostre variable objectiu (pòlissa activa o anul·lada)?

Després de realitzar una recerca d'informació per aprendre sobre cada model, i d'haver modelat una predicció amb cadascun d'ells, el que s'ha obtingut és que l'arbre de decisió condicional i la xarxa neuronal són els models que prediuen amb més assertivitat les pòlisses que no es renovaran. Per altra banda, el model que fa una millor predicció de les pòlisses que sí seran renovades, és la màquina de suport vectorial (svm).

Pel que fa als mètodes ensemblats, el que obté una probabilitat més alta d'encerts de pòlisses que no es renovaran és el model 2.2, fet que era fàcil preveure ja que l'ensemble prioitza la predicció de 0 contra dels 1. En aquest aspecte, el model 2.2 millora la probabilitat de predicció dels cinc models individuals. Tot i així, si es tenen en compte els altres ensembles (que no donen tanta impotència als 0) el model que en fa una millor predicció és l'ensemble 1, el qual fa el màxim entre les prediccions obtingudes pels cinc models. Tot i així, la capacitat predictiva d'aquest model és molt similar a la del Logit i Probit i inferior a la de l'arbre de decisió condicional i la xarxa neuronal. Per tant, no millora la capacitat predictiva dels models individuals.

Pel que fa a la probabilitat d'encerts de les pòlisses que es renovaran, dels models ensemblats el que en fa una millor predicció és el 2.1. Aquest, preval la predicció d'1 enlloc de 0, per tant, té sentit que sigui el que mostri una probabilitat més alta. Si no es té en compte aquest ensemble de prevalença dels 1, el model amb la millor capacitat predictiva en aquest aspecte és l'ensemble 3. Aquest, fa una predicció de la variable objectiu utilitzant la mitjana entre les prediccions obtingudes pels cinc models. La predicció que fa és relativament bona i superior als models individuals, però amb una excepció: l'svm. Aquest obté una proporció de vertaders positius del 87% mentre que la de l'ensemble 3 és del 67%.

Les conclusions que s'obtenen doncs, vistos els resultats obtinguts, són que els models d'ensemble tractats en aquest estudi no són els millors per millorar la capacitat predictiva obtinguda mitjançant la regressió logística, la regressió Probit, l'arbre de decisió condicional, la xarxa neuronal i la màquina de suport vectorial. Seria necessari provar i aplicar altres maneres d'ensemblar per tal de poder obtenir millors resultats per predir si el client renovarà o no la pòlissa que té contractada.

Tot i així, a l'hora de realitzar una predicció de renovació o no de pòlisses, és important tenir clar l'objectiu de l'anàlisi, depenent de si l'empresa d'assegurances en qüestió vol saber si les pòlisses seran renovades per tal de recompensar al client i motivar-lo en seguir a la companyia, o saber quines pòlisses es preveu que siguin anul·lades, per tal d'evitar que això passi i oferir millors condicions al client, és recomanable que s'utilitzi un mètode o un altre, en funció de la sensibilitat i l'especificitat obtinguda en cadascun d'ells. O depenent de l'objectiu de l'estudi, es prioritzarà un percentatge major en la precisió o en l'exactitud.

Així, doncs, com a conclusió final, recalcar que tots els models són vàlids per fer prediccions, però que és necessari focalitzar l'estudi en un objectiu concret per tal de decidir quin model és més convenient aplicar a les nostres dades. També es vol destacar que a vegades, mitjançant la realització d'un ensemble entre un conjunt de models, les prediccions poden millorar i poden obtenir-se així, millors resultats que mitjançant un model individual, però que no sempre és així. Com s'ha comprovat en el nostre cas, els models ensemblats han millorat les prediccions d'algun model individual, però amb poca diferència i no de tots els models. Tot és qüestió de buscar la manera d'ensemblar que millor s'adeqüi a l'objectiu final de l'estudi.

Agraïments:

Agraeixo a la meva tutora del Treball de Recerca, Catalina Bolancé, la seva professionalitat, ajuda, paciència i comprensió durant aquests últims mesos. També agreixo tot el suport a la meva família.

Bibliografia

Dietterich, T. (2000). *“Ensemble Methods of Machine Learning”.* Oregon State University.

Amat, J. (2016). *“Validación de modelos de regresión”.* RPubs.

Barber, X. (s.f.). *“Redes Neuronales: Introduciendonos en la Inteligencia Artificial”.* Elche: Universidad Miguel Hernández de Elche.

Brownlee, J. (2016). *"Árboles de clasificación y regressió para el aprendizaje automático".*

Caruana, R., Niculescu-Mizil, A., Crew, G., & Ksikes, A. (n.d.). *"Ensemble Selection from Libraries of Models".* Nova York: Cornell University.

Delgado, R. (8 de juliol de 2018). *“Introducción a la Redes Neuronales Artificiales”*. Obtenido de RPubs: https://rpubs.com/rdelgado/402754

Dietterich, T. (2000). *“Ensemble Methods of Machine Learning”.* Oregon State University.

Geo Turoriales. (7 de març de 2016). *Árbol de decisión*. Obtenido de Gestión de Operaciones: https://www.gestiondeoperaciones.net/procesos/arbol-de-decision/

Jaramillo, E. D. (s.f.). *"Regresión Logística y Regresión Probit".* RPubs.

Llano, L., & Mosquetera, V. (2006). *"El modelo Logit, una alternativa para medir probabilidad de permanencia estudiantil".* Colombia: Facultad Nacional de Colombia.

Máquinas vectoriales de soporte para la clasificación binaria. (2019). Retrieved from MathWork: https://es.mathworks.com/help/stats/support-vector-machines-for-binary-classification.html

Martínez, G. V. (2015). *"Metodología de minería de datos para el estudio de tablas de siniestralidad vial".* Madrid: Universidad Complutense de Madrid.

Padilla-Barreto, A., Guillén, M., & Bolancé, C. (s.f.). *"Big-data Analytics en seguros".* Barcelona.

Train, K. E. (2014). *"Métodos de elección discreta con simulación (Segunda edición)".*

Annex

Codi de l'R

```
# INICIALITZACIo

# Paquets i llibreries

#install.packages("graphics")
library(graphics)
#install.packages("Amelia")#Podemos visualizar los missings de un datafra
me
library(Amelia)
library(gridExtra) #combinacio de grafics en una mateixa finestra
library(caret)
library(car)
library(knitr) # taules
library(kableExtra) #opcions per taula
library(FactoMineR)
library(factoextra)
library(dplyr) #preprocessing
library(ISLR) #eliminacio de missings
library(data.table)
library(readxl) #lectura de la bbdd
#install.packages("gtools")
library(gtools) # Programacio de macros
#install.packages("party")
library(party) #Arbre de decisio condicional
#install.packages("nnet")
library(nnet) #Xarxes neuronals (Info: http://scg.sdsu.edu/ann_r/)
#install.packages("ROCR")
library(ROCR) #corva RO
#install.packages("ggplot2")
library(ggplot2)
#install.packages("pROC")
library(pROC)
#install.packages("e1071")
library(e1071) #SVM

##############################

# Importacio i lectura de la bbdd
data <- read_excel("dades_hogar2.xlsx")
#View(data)
summary(data)
attach(data)
# la bbdd presenta 206 812 observacions i 14 variables
```

```r
# Valors mancants
#str(data)
#sum(is.na(data))
#data <- na.omit(data)

# PREPROCeS DE LES DADES

# Re-categoritzacio de les variables
data$sex_customer=as.factor(data$sex_customer)
data$Policy_PaymentMethod=as.factor(data$Policy_PaymentMethod)
data$HomeType_H=as.factor(data$HomeType)
data$policy_status_at_t=as.factor(data$policy_status_at_t)

# per poder executar l'arbre de decisio i l'svm, es necessari que les var
iables siguin numeriques o dicotomiques
# per aixo ambdos models s'implementaran amb les variables HomeType i Pol
icy_PaymentMethod recodificades com a dicotomiques
data$HomeType1<- ifelse(data$HomeType == "PI",1,0)  #HomeType = 1 si pert
any al grup majoritari Pis ("PI")
data$Policy_PaymentMethod1<- ifelse(data$Policy_PaymentMethod == "A",1,0)
  #Policy_PaymentMethod = 1 si pertany al grup majoritari Anual ("A")
prop.table(table(data$Policy_PaymentMethod1))
prop.table(table(data$HomeType1))


# DESCRIPTIVA

# tractament de la variable Age_client per fer la descriptiva
k2 = ceiling(1+log(n,2))
k2 # hauriem de dividir en 19 intervals

data$Age_client=as.numeric(data$Age_client)
data$Age_client<-
cut(data$Age_client,breaks=19, label=c(19.7,23.2,26.7,30.2,33.7,37.2,40.7
,44.2,47.7,51.2,54.7,58.2,61.7,65.2,68.7,72.2,75.7,79.2,82.7))

ggplot(data,aes(x = Age_client)) +
  geom_bar() + ggtitle("Classificacio de les polisses segons la variable
Age_client")


# Descriptiva univariant

# Proporcio d'anul⁄lades i vigents
table(data$policy_status_at_t)
prop.table(table(data$policy_status_at_t))#percentages

# Proporcio de categories per cada variable re-categoritzada
table(data$sex_customer)
```

```
prop.table(table(data$Policy_PaymentMethod))
prop.table(table(data$HomeType))
table(data$Age_client)

# Variables categoriques

# diagrama de barres per la variable d'estudi
ggplot(data,aes(x = policy_status_at_t)) +
  geom_bar() + ggtitle("Classificacio de les polisses segons la variable
policy_status_at_t")

# Barplots i pies

cat<- c("sex_customer","Policy_PaymentMethod","HomeType","nunpol_sum","Po
licy_numSupplements","nclaims","nclaims_A","nclaims_C")
par(mfrow=c(1,2))
pie(table(data[[(cat[2])]]),main= paste("Diagrama de sectors variable",ca
t[3]))
pie(table(data[[(cat[9])]]),main= paste("Diagrama de sectors variable",ca
t[2]))

par(mfrow=c(1,2))
pie(table(data[[(cat[7])]]),main= paste("Diagrama de sectors variable",ca
t[7]))
pie(table(data[[(cat[8])]]),main= paste("Diagrama de sectors variable",ca
t[8]))

par(mfrow=c(1,2))
pie(table(data[[(cat[4])]]),main= paste("Diagrama de sectors variable",ca
t[6]))
pie(table(data[[(cat[3])]]),main= paste("Diagrama de sectors variable",ca
t[1]))

data$nunpol_sum<-as.numeric(data$nunpol_sum)
data$Policy_numSupplements<-as.numeric(data$Policy_numSupplements)
ggplot(data,aes(x = Policy_numSupplements)) +
  geom_bar() + ggtitle("Diagrama de barres de la variable Policy_numSuppl
ements")
ggplot(data,aes(x = nunpol_sum)) +
  geom_bar() + ggtitle("Diagrama de barres de la variable nunpol_sum")

hist(data$Age_client,main= paste("Histograma variable",Age_client))


# Variables num◆riques

# grafics de densitat
```

```
num<-
c("Client_Seniority","Insuredcapital_content_H","Insuredcapital_continent
_H","Policy_seniority")
p1<-
qplot(data[[num[1]]],data=data, geom = "density", main= paste("Grafic de
densitat de la variable",num[1]),xlab=num[1])
p2<-
qplot(data[[num[4]]],data=data, geom = "density", main= paste("Grafic de
densitat de la variable",num[4]),xlab=num[4])
grid.arrange(p1,p2)
p4<-
qplot(data[[num[2]]],data=data, geom = "density", main= paste("Grafic de
densitat de la variable",num[2]),xlab=num[2])
p5<-
qplot(data[[num[3]]],data=data, geom = "density", main= paste("Grafic de
densitat de la variable",num[3]),xlab=num[3])
grid.arrange(p4,p5)

# resum de les vars contingut i continent del capital assegurat
summary(data$Insuredcapital_content_H)
summary(data$Insuredcapital_continent_H)

# Boxplots
q1<-ggplot(data = data, aes(x = "", y = data$Insuredcapital_content_H)) +
  geom_boxplot() + ylab("Insuredcapital_content_H")  #presenta algun valo
r atipic
q2<-
ggplot(data = data, aes(x = "", y = data$Insuredcapital_continent_H)) +
  geom_boxplot() + ylab("Insuredcapital_continent_H")
q4<-ggplot(data = data, aes(x = "", y = data$Client_Seniority)) +
  geom_boxplot() + ylab("Client_Seniority") #presenta algun valor atipic
q5<-ggplot(data = data, aes(x = "", y = data$Age_client)) +
  geom_boxplot() + ylab("Age_client")
q6<-ggplot(data = data, aes(x = "", y = data$Policy_seniority)) +
  geom_boxplot() + ylab("Policy_seniority")

grid.arrange(q5,q6,q4,q1,q2,nrow=3,ncol=2,top="Boxplot de les variables
             num\ufffdriques que semblen presentar valors atipics")


# Descriptiva bivariant

# Variables num\ufffdriques
num<-
c("Age_client","Client_Seniority","Insuredcapital_content_H","Insuredcapi
tal_continent_H","Policy_seniority")
```

```
#Grafics de densitat de les covariables num�riques per a cada classe de
la variable policy_status_at_t (0= anulada, 1=vigent)
b1<-
qplot(data[[num[1]]],data=data, geom = "density",color =policy_status_at_
t, main= paste("Variable",num[1]),xlab= num[1])
b2<-
qplot(data[[num[2]]],data=data, geom = "density",color =policy_status_at_
t, main= paste("Variable",num[2]),xlab= num[2])
b3<-
qplot(data[[num[5]]],data=data, geom = "density",color =policy_status_at_
t, main= paste("Variable",num[5]),xlab= num[5])
b4<-
qplot(data[[num[3]]],data=data, geom = "density",color =policy_status_at_
t, main= paste("Variable",num[6]),xlab= num[3])
b5<-
qplot(data[[num[4]]],data=data, geom = "density",color =policy_status_at_
t, main= paste("Variable",num[7]),xlab= num[4])
grid.arrange(b1,b2,b3,b4,b5,nrow=3)

# Variables categoriques
cat<- c("policy_status_at_t","sex_customer","Policy_PaymentMethod","HomeT
ype","nunpol_sum","Policy_numSupplements","yearpstar","yearlast","nclaims
","nclaims_A","nclaims_C")

#Taules

#Tipus d'habitatge (HomeType)
t_htype= table(data$HomeType,data$policy_status_at_t)
pt_htype= prop.table(t_htype,2)
colnames(pt_htype)= c("polissa anulada","polissa activa")
rownames(pt_htype)=c("atic","planta baixa","pis","rural","casa individual
","casa familiar")
kable(pt_htype,caption = "Taula de l'estat de la polissa en funcio del ti
pus d'habitatge del client") %>%kable_styling(c("striped", "bordered"))

#M�tode de pagament (Policy_PaymentMethod)
t_pay= table(data$Policy_PaymentMethod,data$policy_status_at_t)
pt_pay= prop.table(t_pay,2)
colnames(pt_pay)= c("polissa anulada","polissa activa")
rownames(pt_pay)= c("Anual", "Semestral","Trimestral")
kable(pt_pay,caption = "Taula de l'estat de la polissa en funcio de la fo
rma de pagament") %>%kable_styling(c("striped", "bordered"))

#quin percentatge de les dones/homes tenen la polissa vigent
t_sex<-table(data$sex_customer,data$policy_status_at_t)
pt<-prop.table(t_sex,1)
colnames(pt)= c("polissa anulada","polissa activa")
rownames(pt)= c("dona","home")
```

```
kable(pt,caption = "Taula de l'estat de la polissa en funcio del sexe del
 client")%>% kable_styling(c("striped", "bordered"))

##Recodifiquem la variable nclaims_A de manera que 0= 0 siniestres ocorre
guts per culpa de l'assegurat, 1= 1 sinistre i 2= 2 o mes sinistres
#data$nclaims_A<-as.numeric(data$nclaims_A)
setDT(data)
data[,"nclaims_A":=ifelse(nclaims_A>=2,2,nclaims_A)]
table(data$nclaims_A)
t_nclaims_A= table(data$nclaims_A,data$policy_status_at_t)
pt_nclaims_A= prop.table(t_nclaims_A,2)
colnames(pt_nclaims_A)= c("polissa anulada","polissa activa")
rownames(pt_nclaims_A) = c("0 sinistres","1 sinistre","2 o mes sinistres"
)
kable(pt_nclaims_A,caption = "Taula de l'estat de la polissa en funcio de
l nombre de sinistres ocorreguts per culpa de l'assegurat") %>%kable_styl
ing(c("striped", "bordered"))


##Recodifiquem la variable nclaims_C de manera que 0= 0 siniestres ocorre
guts per culpa de l'assegurat, 1= 1 sinistre i 2= 2 o mes sinistres
#data$nclaims_C<-as.numeric(data$nclaims_C)
data[,"nclaims_C":=ifelse(nclaims_C>=2,2,nclaims_C)]
table(data$nclaims_C)
t_nclaims_C= table(data$nclaims_C,data$policy_status_at_t)
pt_nclaims_C= prop.table(t_nclaims_C,2)
colnames(pt_nclaims_C)= c("polissa anulada","polissa activa")
rownames(pt_nclaims_C) = c("0 sinistres","1 sinistre","2 o mes sinistres"
)
kable(pt_nclaims_C,caption = "Taula de l'estat de la polissa en funcio de
l nombre de sinistres ocorreguts per culpa del contrari") %>%kable_stylin
g(c("striped", "bordered"))


##Recodifiquem la variable nclaims de manera que 0= 0 siniestres ocorregu
ts per culpa de l'assegurat, 1= 1 sinistre i 2= 2 o mes sinistres
#data$nclaims<-as.numeric(data$nclaims)
data[,"nclaims":=ifelse(nclaims>=2,2,nclaims)]
table(data$nclaims)
t_nclaims= table(data$nclaims,data$policy_status_at_t)
pt_nclaims= prop.table(t_nclaims,2)
colnames(pt_nclaims)= c("polissa anulada","polissa activa")
rownames(pt_nclaims) = c("0 sinistres","1 sinistre","2 o mes sinistres")
kable(pt_nclaims,caption = "Taula de l'estat de la polissa en funcio del
nombre de sinistres ocorreguts en total") %>%kable_styling(c("striped", "
bordered"))

##Recodifiquem la variable Policy_numSupplements de manera que tots els s
inistres amb 10 o mes sumplements quedaran agrupats en un mateix bloc
```

```
data[,"Policy_numSupplements":=ifelse(Policy_numSupplements>=10,10,Policy
_numSupplements)]
table(data$Policy_numSupplements)
t_numSup= table(data$Policy_numSupplements,data$policy_status_at_t)
pt_numSup= prop.table(t_numSup,2)
colnames(pt_numSup)= c("polissa anulada","polissa activa")
rownames(pt_numSup) = c("0","1","2","3","4","5","6","7","8","9","10 o mes
")
kable(pt_numSup,caption = "Taula de l'estat de la polissa en funcio del n
ombre de suplements contractats") %>%kable_styling(c("striped", "bordered
"))


##Recodifiquem la variable numpol_sum de manera que tots els sinistres am
b 10 o mes sumplements quedaran agrupats en un mateix bloc
data[,"nunpol_sum":=ifelse(nunpol_sum>=10,10,nunpol_sum)]
table(data$nunpol_sum)
t_nunpol= table(data$nunpol_sum,data$policy_status_at_t)
pt_nunpol= prop.table(t_nunpol,2)
colnames(pt_nunpol)= c("polissa anulada","polissa activa")
rownames(pt_nunpol) = c("0","1","2","3","4","5","6","7","8","9","10 o mes
")
kable(pt_nunpol,caption = "Taula de l'estat de la polissa en funcio del n
ombre de polisses que el client te contractades") %>%kable_styling(c("str
iped", "bordered"))


# PARTICIo DE LA BBDD PER CROSS-VALIDACIo

#Divisio de la bbdd en un 70 - 30
set.seed(1234)#Es fixa la llavor per despres poder generar numeros aleato
ris

mostra_var<-defmacro(x,expr={
  n<-dim(x)[1] #Definicio d'un vector n amb la dimensio de la bdd
  a<-runif(n,0,1) # genera numeros aleatoris d'una distribucio uniforme
  sampl<-((a>0.70)*0+ (a<=0.70)*1) #Es divideix la mostra en un 70-30
  x$mostra<-sampl
  cat('Tabla de frecuencias \n')
  table(x$mostra)}) #creacio d'una nueva variable en la bdd denominada "m
ostra"
mostra_var(data)

#En la seg?ent macro es crea la variable resposta "renova" a partir de l
a variable policy_status_at_t

#Creacio de la variable renova a partir de la variable "policy_status_at_
t"
```

```
renova <- defmacro(x,expr={x$renova<-
0+(x$policy_status_at_t==1)*1}) #x es la bbdd
renova(data)


# IMPLEMENTACIo DELS M?TODES

### REGRESSo LOGiSTICA

model_train_test<-defmacro(x,expr={
  train<-subset(x,mostra==1)
  attach(train)
  cat('Resultados del ajuste del modelo simple \n')
  tabs<-
rbind(round(table(train$renova),digits=0),round(prop.table(table(train$re
nova)),digits = 3))
  rownames(tabs) <- c("Freq","Propr")

  ## Formula ##
  formula=renova ~ sex_customer+Age_client+HomeType+nunpol_sum+Insuredcap
ital_content_H+Insuredcapital_continent_H+Policy_seniority+Client_Seniori
ty+Policy_numSupplements+Policy_PaymentMethod+nclaims+nclaims_A+nclaims_C

  #Model
  model <- glm(formula,data=train, family=binomial(link=logit))

  #prediccio
  predTest<- predict(model, newdata=train, type = "response") #LR

  #Taules de classificacio

  qualitat<-matrix(0,20,8)
  colnames(qualitat) <-  paste(c("encerts","Falsos positius","Falsos nega
tius","Threshold","Sensitivity","1-Specificity","Max(TP-FN)","Max(TN-
FP)"), sep = "")

  for(i in 1:20) {

    thresh  <- (0.7+i/100)
    predFac <- cut(predTest, breaks=c(-
Inf, thresh, Inf), labels=c("anulades", "vigents"))
    cTab    <- table(train$renova, predFac, dnn=c("actual", "predicted"))
    addmargins(cTab)

    qualitat[i,1]=(cTab[1,1]+cTab[2,2])/sum(cTab)# encerts
    qualitat[i,2]<-cTab[1,2]/sum(cTab[1,2]+cTab[2,2])#Falsos positius
    qualitat[i,3]<-cTab[2,1]/sum(cTab[1,1]+cTab[2,1])#Falsos negatius
    qualitat[i,4]<-thresh
```

```
    qualitat[i,5]<-cTab[2,2]/sum(cTab[2,1]+cTab[2,2]) # sensitivity
    qualitat[i,6]<-1 - cTab[1,1]/sum(cTab[1,1]+cTab[1,2]) #1-Specificity
    qualitat[i,7]<-max(cTab[2,2]-cTab[2,1]) #Max(TP-FN)
    qualitat[i,8]<-max(cTab[1,1]-cTab[1,2])#Max(TN-FP)
  }
  qualitat

  #### Avaluacio del model amb el testset ###
  testset<- subset(data,mostra==0)

  #Prediccions del model
  predTest2<- predict(model, newdata=testset, type = "response") #LR

  #Threshold
  thresh  <- 0.8769559 #LR

  predFac <- cut(predTest2, breaks=c(-
Inf, thresh, Inf), labels=c("anulades", "vigents"))
  cTab    <- prop.table(table(predFac,testset$renova, dnn=c("predicted",
"actual")))

  #Calculs addicionals
  correctos=(cTab[1,1]+cTab[2,2])/sum(cTab)
  correctos

  fn<-cTab[2,1]/sum(cTab[1,1]+cTab[2,1])#Falsos negatius
  fn

  sensitivity<- cTab[2,2]/sum(cTab[2,1]+cTab[2,2]) # sensitivity
  onemSpecificity<- 1 - cTab[1,1]/sum(cTab[1,1]+cTab[1,2]) #1-Specificity

  list("Tables" = tabs, "Summary_results" = summary(model), "Quality_tabl
e"=qualitat,"Threshold"= thresh,"Classification_table_testset"=addmargins
(cTab), "Correctos"=correctos,"False_negatives"=fn, "Sensitivity"= sensit
ivity, "1-specificity"=onemSpecificity)
  })

model_train_test(data)

#write.table(cbind(testset$renova,predTest2), file = "LR_test.csv", sep =
 ",",col.names=c("testset_renova","predTest2"),row.names = FALSE)

#### Corba ROC del conjunt de dades test ####

prob <- prediction(predTest2,  testset$renova, label.ordering = c('0', '1
'))
tprfpr <- performance(prob, "tpr", "fpr")
tpr <- unlist(slot(tprfpr, "y.values")) # TP
```

```
fpr <- unlist(slot(tprfpr, "x.values")) # FP
roc <- data.frame(tpr, fpr)
#X11()
ggplot(roc) + geom_line(aes(x = fpr, y = tpr)) +geom_abline(intercept = 0
, slope = 1, colour = "gray") +   ylab("Sensibilitat") +   xlab("1 - Espe
cificitat") +  geom_hline(yintercept=sensitivity,  colour = "blue") +  ge
om_vline(xintercept=onemSpecificity,  colour = "blue")

#Mesures de rendiment

#(a) AUC

auc(train$renova,as.numeric(predTest))
auc(testset$renova,as.numeric(predTest2))

#(b) Error quadratic mig (MSE)

mse_train<-sum((train$renova-as.numeric(predTest))^2)/nrow(train)
mse_train
mse_test<-sum((testset$renova-as.numeric(predTest2))^2)/nrow(testset)
mse_test

#(c) Biaix

bias_train<-mean(as.numeric(predTest))-mean(train$renova)
bias_train
bias_test<-mean(as.numeric(predTest2))-mean(testset$renova)
bias_test

#### Bondat de l'ajust ####

# Anova

#Model redu?t (model nomes amb l'intercept com a variable explicativa)
formula_rm=renova ~1
model_r<- glm(formula_rm,data=train,family=binomial(link=logit))

# Test per comparar el model complet amb el redu?t

#Ho: [Beta_1=Beta_2=...=Beta_10=0 / Beta_0]

anova(model_r, model, test="Chisq")
#Pr(>Chi) < 2.2e-
16 *** es rebutja Ho i es conclou que  LR_Model es estadisticament signif
icatiu respecte al LR_Model_red


### REGRESSo PROBIT
```

```
model_train_test<-defmacro(x,expr={
  train<-subset(x,mostra==1)
  attach(train)
  cat('Resultados del ajuste del modelo simple \n')
  tabs<-
rbind(round(table(train$renova),digits=0),round(prop.table(table(train$re
nova)),digits = 3))
  rownames(tabs) <- c("Freq","Propr")

  ## Formula ##
  formula=renova ~ sex_customer+Age_client+HomeType+nunpol_sum+Insuredcap
ital_content_H+Insuredcapital_continent_H+Policy_seniority+Client_Seniori
ty+Policy_numSupplements+Policy_PaymentMethod+nclaims+nclaims_A+nclaims_C

  #Model
  model <- glm(formula,data=train, family=binomial(link=probit))

  #prediccio
  predTest<- predict(model, newdata=train, type = "response")

  #Taules de classificacio

  qualitat<-matrix(0,20,8)
  colnames(qualitat) <-  paste(c("encerts","Falsos positius","Falsos nega
tius","Threshold","Sensitivity","1-Specificity","Max(TP-FN)","Max(TN-
FP)"), sep = "")

  for(i in 1:20) {

    thresh  <- (0.7+i/100)
    predFac <- cut(predTest, breaks=c(-
Inf, thresh, Inf), labels=c("anulades", "vigents"))
    cTab    <- table(train$renova, predFac, dnn=c("actual", "predicted"))
    addmargins(cTab)

    qualitat[i,1]=(cTab[1,1]+cTab[2,2])/sum(cTab)# encerts
    qualitat[i,2]<-cTab[1,2]/sum(cTab[1,2]+cTab[2,2])#Falsos positius
    qualitat[i,3]<-cTab[2,1]/sum(cTab[1,1]+cTab[2,1])#Falsos negatius
    qualitat[i,4]<-thresh
    qualitat[i,5]<-cTab[2,2]/sum(cTab[2,1]+cTab[2,2]) # sensitivity
    qualitat[i,6]<-1 - cTab[1,1]/sum(cTab[1,1]+cTab[1,2]) #1-Specificity
    qualitat[i,7]<-max(cTab[2,2]-cTab[2,1]) #Max(TP-FN)
    qualitat[i,8]<-max(cTab[1,1]-cTab[1,2])#Max(TN-FP)
  }
  qualitat

  #### Avaluacio del model amb el testset ###
```

```
  testset<- subset(data,mostra==0)

  #Prediccions del model
  predTest2<- predict(model, newdata=testset, type = "response")

  #Threshold
  thresh  <- 0.8769559

  predFac <- cut(predTest2, breaks=c(-
Inf, thresh, Inf), labels=c("anulades", "vigents"))
  cTab    <- prop.table(table(predFac,testset$renova, dnn=c("predicted",
"actual")))

  #Calculs addicionals
  correctos=(cTab[1,1]+cTab[2,2])/sum(cTab)
  correctos

  fn<-cTab[2,1]/sum(cTab[1,1]+cTab[2,1])#Falsos negatius
  fn

  sensitivity<- cTab[2,2]/sum(cTab[2,1]+cTab[2,2]) # sensitivity
  onemSpecificity<- 1 - cTab[1,1]/sum(cTab[1,1]+cTab[1,2]) #1-Specificity

  list("Tables" = tabs, "Summary_results" = summary(model), "Quality_tabl
e"=qualitat,"Threshold"= thresh,"Classification_table_testset"=addmargins
(cTab), "Correctos"=correctos,"False_negatives"=fn, "Sensitivity"= sensit
ivity, "1-specificity"=onemSpecificity)
  })

model_train_test(data)

#write.table(cbind(testset$renova,predTest2), file = "Probit_test.csv", s
ep = ",",col.names=c("testset_renova","predTest2"),row.names = FALSE)

#### Corba ROC del conjunt de dades test ####

prob <- prediction(predTest2,  testset$renova, label.ordering = c('0', '1
'))
tprfpr <- performance(prob, "tpr", "fpr")
tpr <- unlist(slot(tprfpr, "y.values")) # TP
fpr <- unlist(slot(tprfpr, "x.values")) # FP
roc <- data.frame(tpr, fpr)
#X11()
ggplot(roc) + geom_line(aes(x = fpr, y = tpr)) +geom_abline(intercept = 0
, slope = 1, colour = "gray") +   ylab("Sensibilitat") +   xlab("1 - Espe
cificitat") +  geom_hline(yintercept=sensitivity,  colour = "blue") +  ge
om_vline(xintercept=onemSpecificity,  colour = "blue")
```

```
#Mesures de rendiment

#(a) AUC

auc(train$renova,as.numeric(predTest))
auc(testset$renova,as.numeric(predTest2))

#(b) Error quadratic mig (MSE)

mse_train<-sum((train$renova-as.numeric(predTest))^2)/nrow(train)
mse_train
mse_test<-sum((testset$renova-as.numeric(predTest2))^2)/nrow(testset)
mse_test

#(c) Biaix

bias_train<-mean(as.numeric(predTest))-mean(train$renova)
bias_train
bias_test<-mean(as.numeric(predTest2))-mean(testset$renova)
bias_test

#### Bondat de l'ajust ####

# Anova

#Model redu?t (model nomes amb l'intercept com a variable explicativa)
formula_rm=renova ~1
model_r<- glm(formula_rm,data=train,family=binomial(link=probit))

# Test per comparar el model complet amb el redu?t

#Ho: [Beta_1=Beta_2=...=Beta_10=0 / Beta_0]

anova(model_r, model, test="Chisq")
#Pr(>Chi) < 2.2e-
16 *** es rebutja Ho i es conclou que  LR_Model es estadisticament signif
icatiu respecte al LR_Model_red


### ARBRE DE DECISIo CONDICIONAL

model_train_test<-defmacro(x,expr={
  train<-subset(x,mostra==1)
  attach(train)
  cat('Resultados del ajuste del modelo simple \n')
  tabs<-
rbind(round(table(train$renova),digits=0),round(prop.table(table(train$re
nova)),digits = 3))
```

```
  rownames(tabs) <- c("Freq","Propr")

  ## Formula ##
  formula=renova ~ sex_customer+Age_client+HomeType1+nunpol_sum+Insuredca
pital_content_H+Insuredcapital_continent_H+Policy_seniority+Client_Senior
ity+Policy_numSupplements+Policy_PaymentMethod1+nclaims+nclaims_A+nclaims
_C

  #Model
  model <-
ctree(formula, data=train, controls =ctree_control(teststat = c("max"),te
sttype = c("Teststatistic"),mincriterion = 0.99,minbucket=300))

  ####   Evaluacio del model  ####

  #prediccio
  predTest<- predict(model, newdata=train, type = "response")

  #Taules de classificacio

  qualitat<-matrix(0,20,8)
  colnames(qualitat) <-  paste(c("encerts","Falsos positius","Falsos nega
tius","Threshold","Sensitivity","1-Specificity","Max(TP-FN)","Max(TN-
FP)"), sep = "")

  for(i in 1:20) {

    thresh  <- (0.7+i/100)
    predFac <- cut(predTest, breaks=c(-
Inf, thresh, Inf), labels=c("anulades", "vigents"))
    cTab    <- table(train$renova, predFac, dnn=c("actual", "predicted"))
    addmargins(cTab)

    qualitat[i,1]=(cTab[1,1]+cTab[2,2])/sum(cTab)# encerts
    qualitat[i,2]<-cTab[1,2]/sum(cTab[1,2]+cTab[2,2])#Falsos positius
    qualitat[i,3]<-cTab[2,1]/sum(cTab[1,1]+cTab[2,1])#Falsos negatius
    qualitat[i,4]<-thresh
    qualitat[i,5]<-cTab[2,2]/sum(cTab[2,1]+cTab[2,2]) # sensitivity
    qualitat[i,6]<-1 - cTab[1,1]/sum(cTab[1,1]+cTab[1,2]) #1-Specificity
    qualitat[i,7]<-max(cTab[2,2]-cTab[2,1]) #Max(TP-FN)
    qualitat[i,8]<-max(cTab[1,1]-cTab[1,2])#Max(TN-FP)
  }
  qualitat

  #### Avaluacio del model amb el testset ###

  testset<- subset(x,mostra==0)
```

```
  #Prediccions del model
  predTest2<- predict(model, newdata=testset, type = "response")

  #Threshold
  thresh <- 0.8769559  #CT

  predFac <- cut(predTest2, breaks=c(-
Inf, thresh, Inf), labels=c("anulades", "vigents"))
  cTab    <- prop.table(table(predFac,testset$renova, dnn=c("predicted",
"actual")))
  #addmargins(cTab)

  #Calculs addicionals
  correctos=(cTab[1,1]+cTab[2,2])/sum(cTab)
  correctos

  fn<-cTab[2,1]/sum(cTab[1,1]+cTab[2,1])#Falsos negatius
  fn

  sensitivity<- cTab[2,2]/sum(cTab[2,1]+cTab[2,2]) # sensitivity
  onemSpecificity<- 1 - cTab[1,1]/sum(cTab[1,1]+cTab[1,2]) #1-Specificity

  list("Tables" = tabs, "Summary_results" = summary(model), "Quality_tabl
e"=qualitat,"Threshold"= thresh,"Classification_table_testset"=addmargins
(cTab), "Correctos"=correctos,"False_negatives"=fn, "Sensitivity"= sensit
ivity, "1-specificity"=onemSpecificity)
})

model_train_test(data)

#write.table(cbind(testset$renova,predTest2), file = "CT_test.csv", sep =
 ",",col.names=c("testset_renova","predTest2"),row.names = FALSE)

#### Corba ROC del conjunt de dades test ####

prob <- prediction(predTest2,  testset$renova, label.ordering = c('0', '1
'))
tprfpr <- performance(prob, "tpr", "fpr")
tpr <- unlist(slot(tprfpr, "y.values")) # TP
fpr <- unlist(slot(tprfpr, "x.values")) # FP
roc <- data.frame(tpr, fpr)
#x11()
ggplot(roc) + geom_line(aes(x = fpr, y = tpr)) +geom_abline(intercept = 0
, slope = 1, colour = "gray") +   ylab("Sensibilitat") +   xlab("1 - Espe
cificitat") +  geom_hline(yintercept=sensitivity,  colour = "blue") +  ge
om_vline(xintercept=onemSpecificity,  colour = "blue")

#Mesures de rendiment
```

```
#(a) AUC

auc(train$renova,as.numeric(predTest))
auc(testset$renova,as.numeric(predTest2))

#(b) Error quadratic mig (MSE)

mse_train<-sum((train$renova-as.numeric(predTest))^2)/nrow(train)
mse_train
mse_test<-sum((testset$renova-as.numeric(predTest2))^2)/nrow(testset)
mse_test

#(c) Biaix

bias_train<-mean(as.numeric(predTest))-mean(train$renova)
bias_train
bias_test<-mean(as.numeric(predTest2))-mean(testset$renova)
bias_test

#Grafic: Arbre

png("tree.png", res=80, height=1900, width=3000)
plot(model, gp = gpar(fontsize = 7),inner_panel=node_inner, type="simple"
)# Figure
dev.off()


### XARXA NEURONAL

model_train_test<-defmacro(x,expr={
  train<-subset(x,mostra==1)
  attach(train)
  cat('Resultados del ajuste del modelo simple \n')
  tabs<-
rbind(round(table(train$renova),digits=0),round(prop.table(table(train$re
nova)),digits = 3))
  rownames(tabs) <- c("Freq","Propr")

  ## Formula ##
  formula=renova ~ sex_customer+Age_client+HomeType+nunpol_sum+Insuredcap
ital_content_H+Insuredcapital_continent_H+Policy_seniority+Client_Seniori
ty+Policy_numSupplements+Policy_PaymentMethod+nclaims+nclaims_A+nclaims_C
  #afegir el prune per podar l'arbre

  #Model
  model <- nnet(formula, data=train, size=10,maxit=10000,decay=5e-
6, linout=T)
```

```
  ## Evaluacio del model  ##

  #prediccio
  predTest<- predict(model, newdata=train)

  #Taules de classificacio

  qualitat<-matrix(0,20,8)
  colnames(qualitat) <-  paste(c("encerts","Falsos positius","Falsos nega
tius","Threshold","Sensitivity","1-Specificity","Max(TP-FN)","Max(TN-
FP)"), sep = "")

  for(i in 1:20) {

    thresh  <- (0.7+i/100)
    predFac <- cut(predTest, breaks=c(-
Inf, thresh, Inf), labels=c("anulades", "vigents"))
    cTab    <- table(train$renova, predFac, dnn=c("actual", "predicted"))
    addmargins(cTab)

    qualitat[i,1]=(cTab[1,1]+cTab[2,2])/sum(cTab)# encerts
    qualitat[i,2]<-cTab[1,2]/sum(cTab[1,2]+cTab[2,2])#Falsos positius
    qualitat[i,3]<-cTab[2,1]/sum(cTab[1,1]+cTab[2,1])#Falsos negatius
    qualitat[i,4]<-thresh
    qualitat[i,5]<-cTab[2,2]/sum(cTab[2,1]+cTab[2,2]) # sensitivity
    qualitat[i,6]<-1 - cTab[1,1]/sum(cTab[1,1]+cTab[1,2]) #1-Specificity
    qualitat[i,7]<-max(cTab[2,2]-cTab[2,1]) #Max(TP-FN)
    qualitat[i,8]<-max(cTab[1,1]-cTab[1,2])#Max(TN-FP)
  }
  qualitat

  #### Avaluacio del model amb el testset ###

  testset<- subset(x,mostra==0)

  #Prediccions del model

  predTest2<- predict(model, newdata=testset)

  #Threshold
  thresh <- 0.8769559  #NN

  predFac <- cut(predTest2, breaks=c(-
Inf, thresh, Inf), labels=c("anulades", "vigents"))
  cTab    <- prop.table(table(predFac,testset$renova, dnn=c("predicted",
"actual")))
  #addmargins(cTab)
```

```
  #Calculs addicionals
  correctos=(cTab[1,1]+cTab[2,2])/sum(cTab)
  correctos

  fn<-cTab[2,1]/sum(cTab[1,1]+cTab[2,1])#Falsos negatius
  fn

  sensitivity<- cTab[2,2]/sum(cTab[2,1]+cTab[2,2]) # sensitivity
  onemSpecificity<- 1 - cTab[1,1]/sum(cTab[1,1]+cTab[1,2]) #1-Specificity

  list("Tables" = tabs, "Summary_results" = summary(model), "Quality_tabl
e"=qualitat,"Threshold"= thresh,"Classification_table_testset"=addmargins
(cTab), "Correctos"=correctos,"False_negatives"=fn, "Sensitivity"= sensit
ivity, "1-specificity"=onemSpecificity)
  })

model_train_test(data)

#write.table(cbind(testset$renova,predTest2), file = "NN_test.csv", sep =
 ",",col.names=c("testset_renova","predTest2"),row.names = FALSE)

#### Corba ROC del conjunt de dades test ####

prob <- prediction(predTest2,  testset$renova, label.ordering = c('0', '1
'))
tprfpr <- performance(prob, "tpr", "fpr")
tpr <- unlist(slot(tprfpr, "y.values")) # TP
fpr <- unlist(slot(tprfpr, "x.values")) # FP
roc <- data.frame(tpr, fpr)
#X11()
ggplot(roc) + geom_line(aes(x = fpr, y = tpr)) +geom_abline(intercept = 0
, slope = 1, colour = "gray") +   ylab("Sensibilitat") +   xlab("1 - Espe
cificitat") +  geom_hline(yintercept=sensitivity,  colour = "blue") +  ge
om_vline(xintercept=onemSpecificity,  colour = "blue")

#Mesures de rendiment

#(a) AUC

auc(train$renova,as.numeric(predTest))
auc(testset$renova,as.numeric(predTest2))

#(b) Error quadratic mig (MSE)

mse_train<-sum((train$renova-as.numeric(predTest))^2)/nrow(train)
mse_train
mse_test<-sum((testset$renova-as.numeric(predTest2))^2)/nrow(testset)
```

```
mse_test

#(c) Biaix

bias_train<-mean(as.numeric(predTest))-mean(train$renova)
bias_train
bias_test<-mean(as.numeric(predTest2))-mean(testset$renova)
bias_test

# Grafic

#install.packages("devtools")
require(devtools)
#source_url('https://gist.githubusercontent.com/fawda123/7471137/raw/466c
1474d0a505ff044412703516c34f1a4684a5/nnet_plot_update.r')

plot.nnet(model)




### SVM

model_train_test<-defmacro(x,expr={ #x= bbdd
  train<-subset(x,mostra==1)# conjunt d'entrenament
  attach(train)
  cat('Resultados del ajuste del modelo simple \n')
  tabs<-
rbind(round(table(train$renova),digits=0),round(prop.table(table(train$re
nova)),digits = 3))
  rownames(tabs) <- c("Freq","Propr")

  ## Formula ##
  formula=renova ~ sex_customer+Age_client+HomeType1+nunpol_sum+Insuredca
pital_content_H+Insuredcapital_continent_H+Policy_seniority+Client_Senior
ity+Policy_numSupplements+Policy_PaymentMethod1+nclaims+nclaims_A+nclaims
_C

  #Model
  model <- svm(formula, data=train)

  ## Avaluacio del model  ##

  predTest<- predict(model, newdata=train, type = "response")

  #Taules de classificacio
```

```
  #### Es contruira la taula "qualitat" on visualizar indicadors calculat
s a partir de las taules de classificacio ####
  #Aqui es visualitzen els falsos positius (TP), falsos negatius (FN), th
reshold, sensitividad y 1-
especificidad considerant diferents taules de clasificacio

  qualitat<-matrix(0,20,8)
  colnames(qualitat) <-  paste(c("encerts","Falsos positius","Falsos nega
tius","Threshold","Sensitivity","1-Specificity","Max(TP-FN)","Max(TN-
FP)"), sep = "")

  for(i in 1:20) {

    thresh  <- (0.7+i/100)            # threshold per categoritzar les pr
obabilitats predites
    predFac <- cut(predTest, breaks=c(-
Inf, thresh, Inf), labels=c("anulades", "vigents"))
    cTab    <- table(train$renova, predFac, dnn=c("actual", "predicted"))
    addmargins(cTab)

    qualitat[i,1]=(cTab[1,1]+cTab[2,2])/sum(cTab)# encerts
    qualitat[i,2]<-cTab[1,2]/sum(cTab[1,2]+cTab[2,2])#Falsos positius
    qualitat[i,3]<-cTab[2,1]/sum(cTab[1,1]+cTab[2,1])#Falsos negatius
    qualitat[i,4]<-thresh
    qualitat[i,5]<-cTab[2,2]/sum(cTab[2,1]+cTab[2,2]) # sensitivity
    qualitat[i,6]<-1 - cTab[1,1]/sum(cTab[1,1]+cTab[1,2]) #1-Specificity
    qualitat[i,7]<-max(cTab[2,2]-cTab[2,1]) #Max(TP-FN)
    qualitat[i,8]<-max(cTab[1,1]-cTab[1,2])#Max(TN-FP)
  }
  qualitat

  #### Avaluacio del model amb el testset ###

  testset<- subset(x,mostra==0) #Creacio del conjunt de dades test, extra
ient les polisses que representen el 30% de la mostra

  #Prediccions del model
  predTest2<- predict(model, newdata=testset, type = "response")

  #Threshold
  thresh <- 0.8769559   #svm

  predFac <- cut(predTest2, breaks=c(-
Inf, thresh, Inf), labels=c("anulades", "vigents"))
  cTab    <- prop.table(table(predFac,testset$renova, dnn=c("predicted",
"actual")))
  #addmargins(cTab)
```

```
  #Calculs addicionals
  correctos=(cTab[1,1]+cTab[2,2])/sum(cTab)
  correctos

  fn<-cTab[2,1]/sum(cTab[1,1]+cTab[2,1])#Falsos negatius
  fn

  sensitivity<- cTab[2,2]/sum(cTab[2,1]+cTab[2,2]) # sensitivity
  onemSpecificity<- 1 - cTab[1,1]/sum(cTab[1,1]+cTab[1,2]) #1-Specificity

  list("Tables" = tabs, "Summary_results" = summary(model), "Quality_tabl
e"=qualitat,"Threshold"= thresh,"Classification_table_testset"=addmargins
(cTab), "Correctos"=correctos,"False_negatives"=fn, "Sensitivity"= sensit
ivity, "1-specificity"=onemSpecificity)
  })

model_train_test(data)

#write.table(cbind(testset$renova,predTest2), file ="SVM_test.csv", sep =
 ",",col.names=c("testset_renova","predTest2"),row.names = FALSE)

#### Corba ROC del conjunt de dades test ####

prob <- prediction(predTest2,  testset$renova, label.ordering = c('0', '1
'))# Creacio d'objectes predictius, transforma les dades d'entrada en un
format estandaritzat
tprfpr <- performance(prob, "tpr", "fpr")
tpr <- unlist(slot(tprfpr, "y.values")) # TP
fpr <- unlist(slot(tprfpr, "x.values")) # FP (Error tipus I) -
roc <- data.frame(tpr, fpr)
#X11()
ggplot(roc) + geom_line(aes(x = fpr, y = tpr)) +geom_abline(intercept = 0
, slope = 1, colour = "gray") +   ylab("Sensibilitat") +   xlab("1 - Espe
cificitat") +  geom_hline(yintercept=sensitivity,  colour = "blue") +  ge
om_vline(xintercept=onemSpecificity,  colour = "blue")

#Mesures de rendiment

#(a) AUC

auc(train$renova,as.numeric(predTest))
auc(testset$renova,as.numeric(predTest2))

#(b) Error quadratic mig (MSE)

mse_train<-sum((train$renova-as.numeric(predTest))^2)/nrow(train)
mse_train
```

```
mse_test<-sum((testset$renova-as.numeric(predTest2))^2)/nrow(testset)
mse_test

#(c) Biaix

bias_train<-mean(as.numeric(predTest))-mean(train$renova)
bias_train
bias_test<-mean(as.numeric(predTest2))-mean(testset$renova)
bias_test



### ENSEMBLE DE MODELS

# importacio de la bbdd, divisio d'aquestes en train i testset, definicio
 de la formula
data <- read_excel("dades_hogar2.xlsx")
train<-subset(data,mostra==1)
attach(train)
testset<- subset(data,mostra==0)
formula=renova ~ sex_customer+Age_client+HomeType+nunpol_sum+Insuredcapit
al_content_H+Insuredcapital_continent_H+Policy_seniority+Client_Seniority
+Policy_numSupplements+Policy_PaymentMethod+nclaims+nclaims_A+nclaims_C

# es tornen a exectutar els cinc models i es calcula la prediccio del tes
tset de cadascun d'ells
modelLR <- glm(formula,data=train, family=binomial(link=logit))
pred.LR<- predict(modelLR, newdata=testset, type = "response") #prediccio
 Logit

modelP <- glm(formula,data=train, family=binomial(link=probit))
pred.P<- predict(modelP, newdata=testset, type = "response") #prediccio P
robit

modelNN <- nnet(formula, data=train, size=10,maxit=10000,decay=5e-
6, linout=T)
pred.NN<- predict(modelNN, newdata=testset) #prediccio xarxa neuronal

# s'utilitza el model amb les variables HomeType1 i Policy_PaymentMethod1
 que son les recodificades com a dicotomiques
formula1=renova ~ sex_customer+Age_client+HomeType1+nunpol_sum+Insuredcap
ital_content_H+Insuredcapital_continent_H+Policy_seniority+Client_Seniori
ty+Policy_numSupplements+Policy_PaymentMethod1+nclaims+nclaims_A+nclaims_
C

modelCT <-
ctree(formula1, data=train, controls =ctree_control(teststat = c("max"),t
esttype = c("Teststatistic"),mincriterion = 0.99,minbucket=300))
```

```
pred.CT<- predict(modelCT, newdata=testset, type = "response") #prediccio
 arbre de decisio condicional

model <- svm(formula1, data=train)
pred.SVM<- predict(model,testset, type = "response") #prediccio svm


# ensemble 3: mitjana de les probabilitats
ensemble3          <- data.frame(pred.LR,pred.P,pred.CT,pred.NN,pred.SVM)
ensemble3$Mitjana <- rowMeans(ensemble3[, 1:5])
ensemble3$predict  <- ifelse(ensemble3$Mitjana>0.8769559,1,0)  # 0.876955
9 es la freq??ncia d'1 en la variable renova

table(ensemble3$predict,testset$renova)
prop.table(table(ensemble3$predict,testset$renova))

# ensemble 1: maxim
ensemble1          <- data.frame(pred.LR,pred.P,pred.CT,pred.NN,pred.SVM)
ensemble1$pred.LR<- ifelse(ensemble1$pred.LR < 0.8769559,0,1)
ensemble1$pred.P<- ifelse(ensemble1$pred.P < 0.8769559,0,1)
ensemble1$pred.CT<- ifelse(ensemble1$pred.CT < 0.8769559,0,1)
ensemble1$pred.SVM<- ifelse(ensemble1$pred.SVM < 0.8769559,0,1)
ensemble1$pred.NN<- ifelse(ensemble1$pred.NN < 0.8769559,0,1)

ensemble1$Renova.1 <- rowSums(ensemble1[, 1:5] == 1)
ensemble1$Renova.0  <- rowSums(ensemble1[, 1:5] == 0)
ensemble1$predict  <- ifelse((ensemble1$Renova.1>ensemble1$Renova.0),1,0)

table(ensemble1$predict,testset$renova)
prop.table(table(ensemble1$predict,testset$renova))

# ensemble 21: prioritzacio dels 1
ensemble21<-ensemble1
ensemble21$predict  <- ifelse(ensemble21$Renova.1>0,1,0)
table(ensemble21$predict,testset$renova)
prop.table(table(ensemble21$predict,testset$renova))

# ensemble 22: prioritzacio dels 0
ensemble22<-ensemble1
ensemble22$predict  <- ifelse(ensemble22$Renova.0>0,0,1)
table(ensemble22$predict,testset$renova)
prop.table(table(ensemble22$predict,testset$renova))
```