PROJECTE APRENENTATGE AUTOMÀTIC I

PREDICCIÓ DEL TIPUS DE VIDRE EN FUNCIÓ DE LA SEVA COMPOSICIÓ QUÍMICA

SERGIO CÁRDENAS GRACIA & JAN SALLENT BAYÀ

Primavera 2022

ÍNDEX

DESCRIPCIÓ DEL PROJECTE

Per dur a terme aquest treball, utilitzem el dataset "glass". Aquest dataset recull 9 paràmetres. La variable resposta és categòrica de 7 classes (tot i que al dataset només hi ha representades 6), on cada una és un tipus de vidre diferent. Les altres 8 variables són totes numèriques (7 representen els percentatges que representen diferents metalls a la composició de la mostra, i la vuitena és l'índex de refracció).

Nosaltres treballem amb una versió binaritzada del problema, on la variable resposta ens permet identificar només si el tipus de vidre és el building windows non float processed o no.

Hem escollit aquest dataset per ser un problema relativament petit i tractable. Hi ha 214 instàncies, de manera que a nivell computacional serà un problema senzill. A més a més, cap de les instàncies té missing values, fent així el pre-processat més assequible.

El problema d'aquest dataset és predir el tipus de vidre en funció del seu índex de refracció i el seu contingut d'òxid (per exemple Na, Fe, K…).

Així, l'objectiu del nostre treball és fer l'anàlisi i el tractament de les dades per tal de plantejar i validar un model que s'ajusti al dataset de la millor manera i ens permeti fer bones prediccions.

El procediment d'aquest treball es divideix fonamentalment en 3 parts. La primera és el preprocessat de les dades per tal que sigui més fàcil treballar amb elles i no cometem errors que portin a conclusions equivocades. La segona és el plantejament i anàlisi de 4 models: anàlisi de discriminant, SVM amb kernel lineal, SVM amb kernel polinòmic i SVM amb kernel gaussià. Finalment escollim el model més adient i n'extraiem l'error generalitzat.

Per aquest treball intentem senzillament maximitzar l'accuracy independentment de com estigui repartida entre les classes ja que no hi ha cap motiu que ens inclini a prioritzar classificar sempre bé una classe o una altre. És a dir, és indiferent si classifiquem una mostra negativa com a positiva o una positiva com a negativa, el que volem minimitzar és el nombre total de mostres mal classificades.

DESCRIPCIÓ DE LES DADES

Abans d'entrar en el propi estudi de les dades, hem volgut posar una mica de context a aquestes i al treball tot plegat. D'entrada, sabem que cadascuna de les mostres presenta 9 atributs a més de la classe, però quins són aquests i què representen?

El primer, l'índex de refracció (RI), és una mesura que descriu numèricament com es propaga la llum a través d'un medi, en aquest cas, de la mostra que volem classificar. Es calcula dividint la velocitat de la llum en el buit entre la velocitat de la llum en el medi.

D'altra banda, tenim 8 òxids de diferents elements químics: Na, Mg, Al, Si, K, Ca, Ba i Fe. La presència de cadascun d'aquests a la nostra mostra ve descrita de forma percentual, és a dir, ens indica quin percentatge de la composició total del material que estem estudiant representa cadascun d'aquests elements, de manera que la suma dels 8 atributs hauria de sumar el 100%.

Per acabar aquest treball previ, ens hem informat sobre les aplicacions que té estudiar això. Hem vist que aquest tipus de classificació és d'extrema importància en el camp de la criminologia, on ser capaç de decidir si una mostra és d'un tipus de vidre o altre pot ser la clau per resoldre un crim. En concret, com nosaltres treballarem amb la versió binaritzada del dataset, serem capaços de determinar si una mostra pertany a un tipus de vidre específic o no, confirmant o descartant possibles sospites en escenes de crims.

PREPROCESSAT

Passem ara a l'exploració de les dades, començant pel preprocessat d'aquestes. Declarem la variable "Class" com a factor i fem un "summary" de les dades per obtenir el següent:

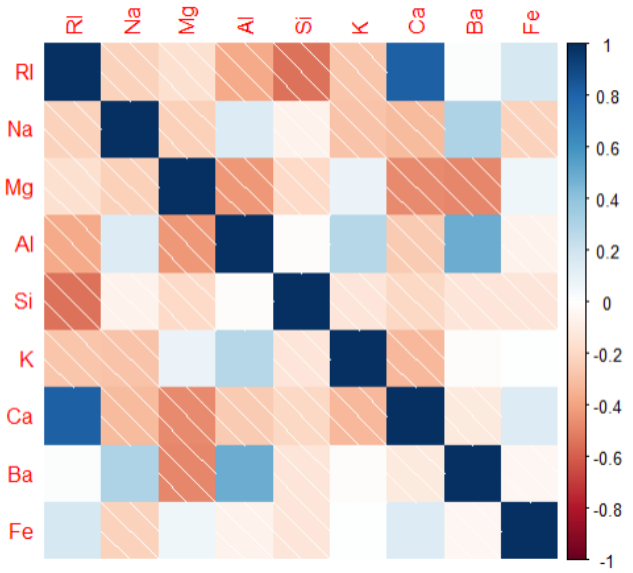
RI	Na	Mg	Al	Si
Min. :1.511	Min. :10.73	Min. :0.000	Min. :0.290	Min. :69.81
1st Qu.:1.517	1st Qu.:12.91	1st Qu.:2.115	1st Qu.:1.190	1st Qu.:72.28
Median :1.518	Median :13.30	Median :3.480	Median :1.360	Median :72.79
Mean :1.518	Mean :13.41	Mean :2.685	Mean :1.445	Mean :72.65
3rd Qu.:1.519	3rd Qu.:13.82	3rd Qu.:3.600	3rd Qu.:1.630	3rd Qu.:73.09
Max. :1.534	Max. :17.38	Max. :4.490	Max. :3.500	Max. :75.41

K	Ca	Ba	Fe	Class
Min. :0.0000	Min. : 5.430	Min. :0.000	Min. :0.00000	N:138
1st Qu.:0.1225	1st Qu.: 8.240	1st Qu.:0.000	1st Qu.:0.00000	P: 76
Median :0.5550	Median : 8.600	Median :0.000	Median :0.00000	
Mean :0.4971	Mean : 8.957	Mean :0.175	Mean :0.05701	
3rd Qu.:0.6100	3rd Qu.: 9.172	3rd Qu.:0.000	3rd Qu.:0.10000	
Max. :6.2100	Max. :16.190	Max. :3.150	Max. :0.51000	

D'aquesta taula, podem veure que no hi ha cap valor destacable a les nostres variables numèriques i, per tant, no hi ha "outliers" a les nostres dades. Pel que fa a la variable resposta, veiem que aproximadament ⅔ de les observacions tenen resposta negativa i l'altre ⅓ restant la positiva. Aquestes proporcions poden tenir sentit perquè estem treballant amb una versió binaritzada d'un problema que originalment té més de dos possibles respostes, de manera que la classe negativa agrupa tots els altres tipus de vidre diferents al que volem diferenciar. Si comparem les variables entre elles veiem que n'hi ha algunes amb valors més grans que altres. Per exemple, el Silici (Si) té una mitjana de 72.65 mentre que el Ferro (Fe) en té poc més d'un 0.05. Aquesta diferència de magnitud en els valors de les diferents variables pot donar peu a models poc precisos degut a la sobredimensió que presenten algunes variables respecte d'altres. Per tal d'evita-ho, una bona solució és escalar les dades. Escalant les dades modifiquem les variables per tal de que totes tinguin la mateixa mitjana i comprimim així els valors dins d'un rang més limitat.

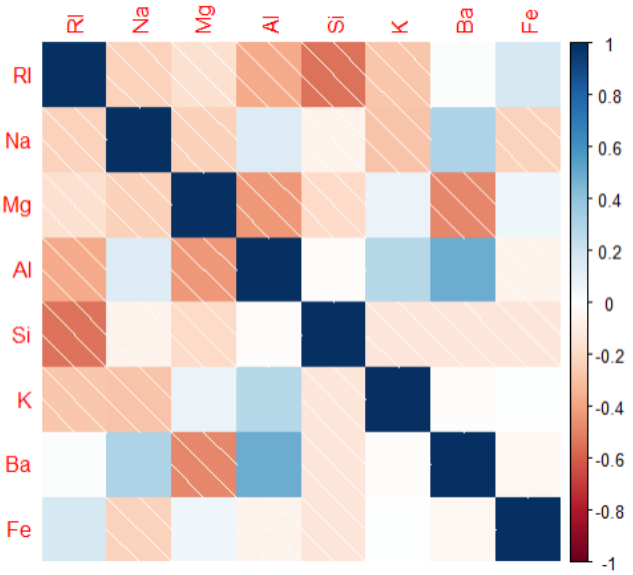
Amb això vist, passem ara a separar el nostre conjunt de dades en training i test, utilitzant el 90% de les dades pel set de training i el 10% restant pel de test. La divisió s'ha fet de forma pseudoaleatoria, però assegurant-nos de que la proporció de les respostes de les dues classes s'ha mantingut intacta, és a dir, que tant el conjunt de training com el de test tindran ⅓ de respostes positives i ⅔ de negatives. No mantenir la proporció de la variable resposta és una mala pràctica i pot conduir a resultats i conclusions errònies.

Amb els dos conjunts de dades ja separats, passem a estudiar les correlacions de les variables dins el datatrain. Per fer-ho, utilitzem la comanda "cor()" de R, que calcula la correlació de Pearson, utilitzada per comparar variables numèriques contínues, com és el nostre cas. Fem un plot dels resultats i obtenim la següent taula:



D'aquesta taula observem un valor prou alt de correlació entre l'Índex de Refracció (RI) i el Calci (Ca), en concret del 0.81, de manera que eliminem la variable Calci (Ca) del datatrain. Hem decidit treure el Calci (Ca) enlloc de l'Índex de Refracció (RI) perquè aquest últim és l'única variable que no és un metall i, per tant, creiem que és un factor més rellevant.

Si tornem ara a fer el plot de les correlacions veurem que ara no hi ha gairebé cap valor a destacar i, dins d'aquests, no considerem que cap sigui prou significatiu com per eliminar-lo.

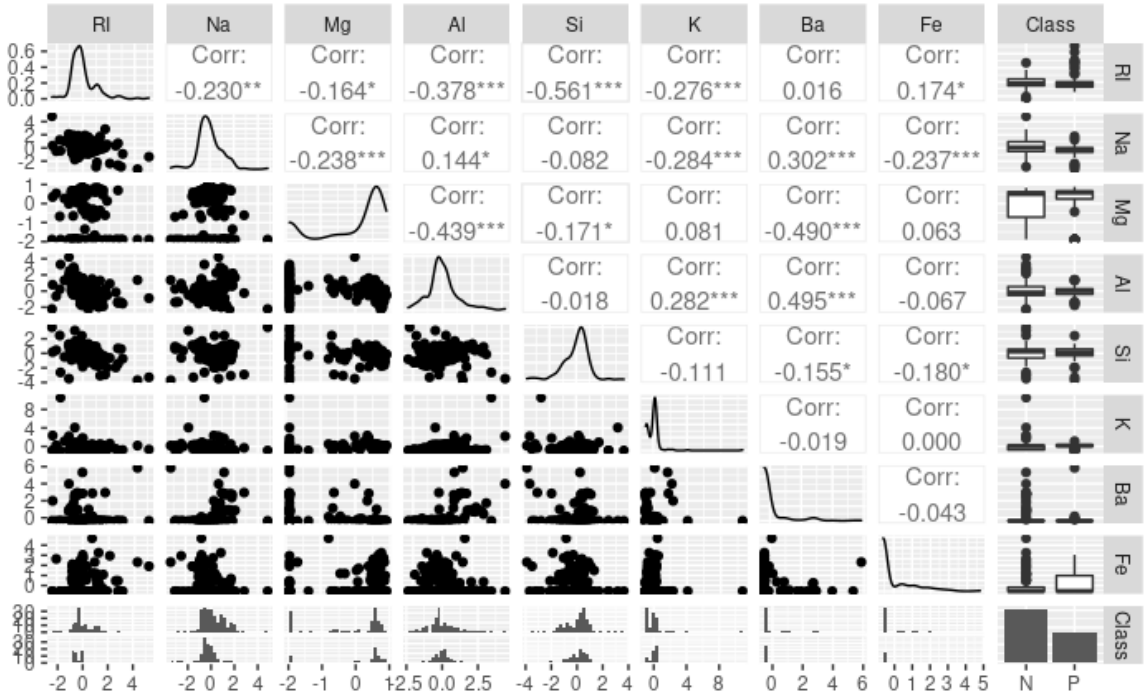


MODELS PROPOSATS

Un cop realitzat el pre-processat i tota l'exploració de les dades pertinent, procedim a plantejar i valorar diferents models per després triar el que considerem millor.

LDA/QDA

Tal com hem dit abans, el primer model escollit és el d'anàlisi de discriminant. Aquest model assumeix que les dades de cada classe segueixen una distribució normal. Observant la funció de densitat de les variables veiem que, tot i que algunes poden tenir alguna semblança amb la distribució normal, hi ha algunes que clarament no són normals. Malgrat no tenir clar la normalitat provem com funciona el model, sabedors de que segurament no sigui el millor.



Coneixem dos tipus de models d'anàlisi de discriminant: lineal (LDA) i quadràtic (QDA). LDA suposa que les dades són normals amb mateixa covariància mentre que QDA suposa que no són iguals en aquest darrer aspecte. Per comprovar si la covariància de les distribucions de les variables és igual fem el Box's M-Test. Aquest test parteix de la hipòtesi nul·la de que les covariàncies són iguals i té com a hipòtesi alternativa que no ho siguin, de manera que si obtenim un p-valor inferior al 0.05 podrem rebutjar la hipòtesi nul·la i assumirem que les covariàncies són diferents.

En el nostre cas, el p-valor del test és pràcticament 0 (2.2^{-16}), per tant, queda molt clar que les covariàncies són diferents i per tant el model més apropiat és el QDA. Per fer-lo, fem servir la tècnica de Cross-Validation. Aquesta tècnica es basa en dividir les nostres dades d'entrenament en dos grups. Amb un grup (normalment més gran) s'intenta construir el model que fa la millor predicció de l'altre grup, és a dir, que obté menor error d'entrenament. Aquest procés es va repetint iterativament (canviant cada cop com repartim les dades d'entrenament) fins que obtenim el model "promig" que millor prediu les nostres pròpies dades. En el nostre cas farem servir LOOCV (Leave-One-Out-Cross-Validation). LOOCV és un tipus de Cross-Validation i la seva particularitat és que es fan tants grups com dades tinguem. Per tant, s'intenta predir una única mostra utilitzant tota la resta per formular el model. Aquesta subtècnica de Cross-Validation és la que obté una major precisió, ja que requereix fer tantes iteracions com dades tinguis. Això té un cost computacional bastant alt però, com que en el nostre cas comptem amb molt poques dades, ens ho podem permetre sense cap problema.

Plantejant el model QDA a partir del nostre training set i validant-lo mitjançant LOOCV obtenim un valor d'accuracy relativament petit, lleugerament superior al 60% (61.46%). És a dir, el nostre model prediu la classe de manera equivocada en 2 de cada 5 observacions. Aquesta imprecisió del nostre model pot ser conseqüència de l'assumpció de normalitat de les dades, quan realment no totes ho eren.

Per analitzar més en detall de quina manera s'equivoca el nostre model, analitzarem la matriu de confusió. Aquesta matriu ens indica la quantitat de dades que hem predit correctament o incorrectament per a cadascuna de les dues classes.

	Reference	
Prediction	N	P
N	59	9
P	65	59

Els resultats de la matriu són bastant sorprenents, ja que veiem que el model sembla tenir més facilitat per encertar una de les dues classes. Concretament, el nostre model prediu bé un 86.76% dels positius mentre que només ho fa amb un 47.58% dels negatius. Això vol dir que hi ha una gran quantitat d'observacions negatives que es classifiquen de forma errònia com a positives. Per tant, malgrat la nostra intuïció i degut al comportament del model, si aquest ens indica que un vidre és positiu no podrem estar gens segurs de que això sigui cert, mentres que si classifica una observació com a negativa és molt probable que realment ho sigui.

Una possible explicació a aquest fet que hem de tenir en compte és que el nostre dataset és una adaptació binària d'un problema multiclasse amb 7 classes (tot i que només 6 es veien representades al dataset). Com ja hem comentat abans, els positius es corresponen a un sol tipus de vidre, mentre que els negatius representen els 6 restants. Dins d'aquests últims, poden haver-hi alguns amb propietats molt diferents al que volem diferenciar, sent fàcils de classificar, però si hi ha algun altre tipus de vidre amb propietats similars al que estem identificant és bastant probable que el nostre model els classifiqui malament. És a dir, la variabilitat de les dades pels diferents tipus de vidre pot generar aquests errors de predicció.

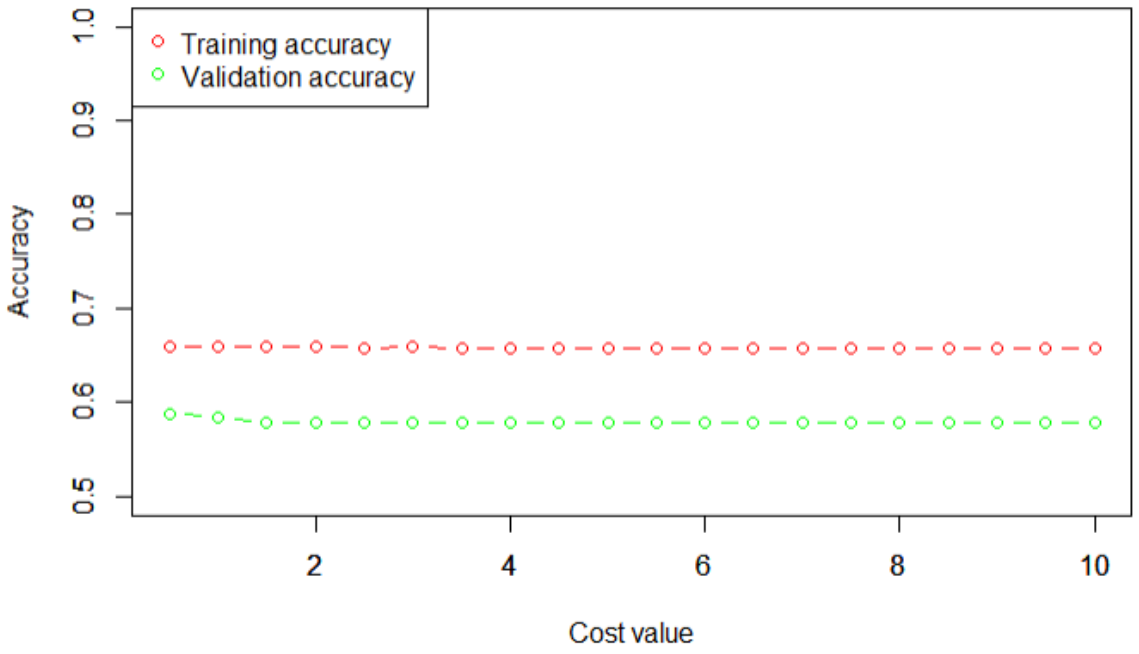
Una altra possible explicació podria ser que les dades no siguin linealment separables. És raonable pensar que un conjunt de dades linealment separables serà molt més fàcil de modelar i, per tant, de predir. Així, podem pensar que els resultats pobres del nostre model QDA són fruit també de la no separabilitat lineal de les dades, però no som capaços d'afirmar-ho. Per fer-ho, haurem de plantejar algun altre model, com per exemple un SVM amb un kernel lineal, i, si els seus resultats són també poc precisos, podrem afirmar amb prou seguretat que les dades no són linealment separables.

SVM

El següents models proposats són SVM (Support Vector Machine). Aquest model intenta buscar un hiperplà separador entre les dues classes. Els models tenen un hiperparàmetre de regularització, que a partir d'ara anomenarem cost. El cost penalitza la quantitat de dades d'entrenament mal classificades. Com més gran sigui el cost, més penalitzarem l'error de classificació i, per tant, el models tendiran a cometre'n menys, a canvi d'augmentar la probabilitat d'acabar provocant models sobreajustats.

SVM - KERNEL LINEAL

En el cas del kernel lineal, no estem aplicant cap transformació a les dades, de manera que només s'obté un bon rendiment si les dades són linealment separables. En el mètode anterior de QDA hem obtingut uns resultats relativament pobres, però desconeixem si el motiu ha estat la falta de gaussianitat de les variables o, pel contrari, la manca de separabilitat lineal de les dades. Si ara tornem a obtenir un model tant pobre quedarà clar que les dades no són separables i, per tant, haurem d'aplicar algun mètode kernel diferent del lineal. Per tal de trobar l'hiperparàmetre òptim hem utilitzat altre cop el mètode de LOOCV. La gràfica següent mostra l'error de validació i de training en funció del valor de cost.



Com veiem, l'hiperparàmetre cost no té cap impacte en la classificació, ja que obtenim els mateixos valors d'accuracy sempre. Això indica que, per molt que castiguem els errors, el kernel lineal no és capaç de trobar un pla millor que separi les dades. Per tant, sembla ser que aproximadament ⅓ de les dades no són separables linealment. Tal com hem dit abans, ens veiem obligats doncs a fer servir altres kernels que transformin les dades. L'accuracy de la validació és força semblant entre el QDA i el SVM lineal ja que ronda el 60%, però si observem la matriu de confusió veiem diferències:

```
          Reference
Prediction   N   P
         N 117  58
         P   7  10
```

Ara, les probabilitats s'han invertit, de manera que classifiquem bé la classe negativa (94%) i, en canvi, classifiquem molt malament la classe positiva (14%). El que passa és que la classe negativa és majoritària (suposa un 64% de les dades totals) i el nostre model senzillament prediu gairebé sempre la classe majoritària ja que té més possibilitats d'encertar la classe. Aquesta idea pot donar un accuracy relativament raonable, però sabem que és un model dolentíssim, ja que no ens aporta cap mena d'informació.

SVM - KERNEL POLINÒMIC

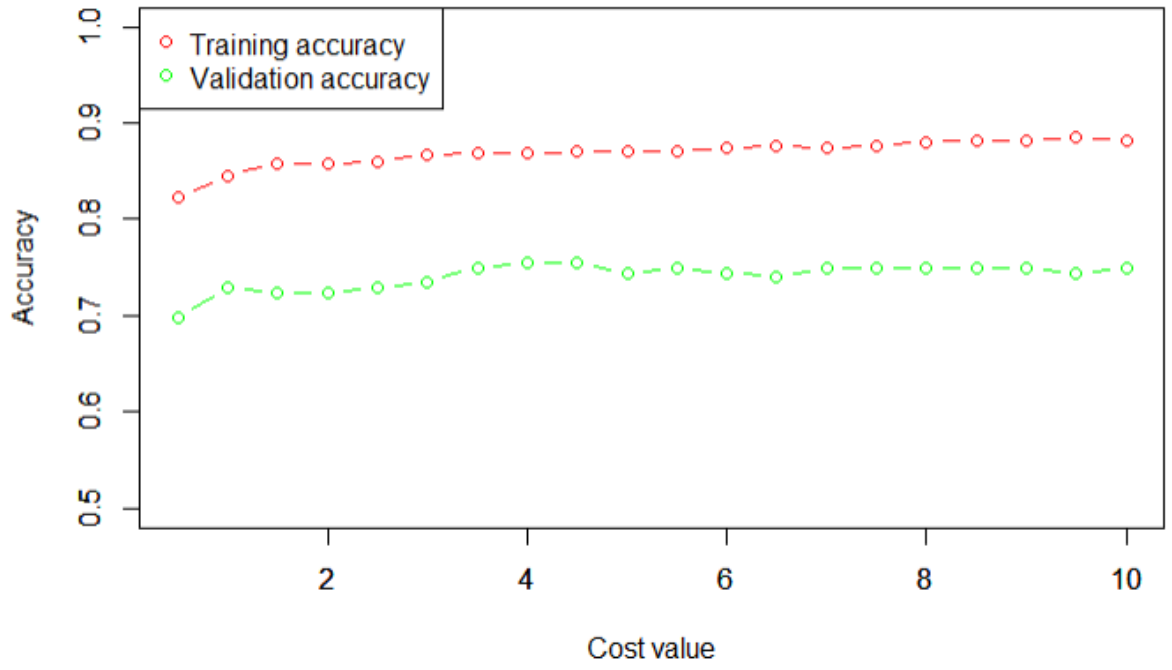
Per tal d'aconseguir una major accuracy i uns models que siguin més profitosos que no pas predir sempre la classe majoritària, i, tenint en compte que estem treballant amb unes dades no separables linealment, fem servir el kernel polinòmic. El kernel polinòmic té la fórmula següent:

$$k\,(x, y)\ =\ (x^{T} * y\ +\ s)^{Q}$$

El paràmetre Q d'aquesta fórmula no representa res més que el grau del polinomi. Si Q fos 1 obtindríem el mateix model que pel kernel lineal, de manera que no tindria sentit agafar aquest valor. Per tant, el dubte principal és si triar una funció de kernel quadrada o cúbica. Simplement comparant a base de prova i error hem vist que els millors resultats s'obtenien per a Q = 3, de manera que treballarem amb aquest valor.

D'altra banda, la s representa una mena de biaix que s'aplica a les dades per tal d'obtenir valors del mateix ordre. Aquest paràmetre per defecte acostuma a ser 1, tot i que a R per defecte pren valor 0. En el nostre cas, l'hem declarat com a s = 1, seguint el mateix criteri de prova i error que hem utilitzat per a Q.

Amb els valors d'aquests hiperparàmetres decidits passem ara a buscar el cost òptim, utilitzant un altre cop el mètode de LOOCV. Els resultats són els següents:



A diferència del kernel lineal, ara el cost té un impacte important en el valor de l'accuracy tant pel cas de training com per al de la validació. Per trobar el cost òptim hem de trobar un valor per al qual l'accuracy de validació del model sigui prou bona, de manera que les prediccions siguin prou encertades, i alhora la diferència entre l'accuracy de training i la de validació no sigui gaire significativa, perquè del contrari estaríem triant un model sobreajustat. En el nostre cas, si observem el gràfic, el primer que ens crida l'atenció és que els valors d'accuracy de training i de validació semblen diferents (amb valors de diferència propers al 10%), però al no ser una diferència gaire exagerada procedirem d'igual manera. En quant a l'accuracy del set de validació, veiem que és bastant similar per a tots els valors de l'hiperparàmetre cost. Així, triem 4 com a valor per a l'hiperparàmetre cost, ja que sembla ser el que maximitza l'accuracy de validació (al voltant del 75%) i minimitza la diferència amb l'accuracy de training (al voltant del 10%) de la millor manera possible.

Per comprovar com es reparteixen les prediccions en funció de cadascuna de les classes tornem a fer un print de la matriu de confusió:

```
          Reference
Prediction   N   P
         N 112  13
         P  12  55
```

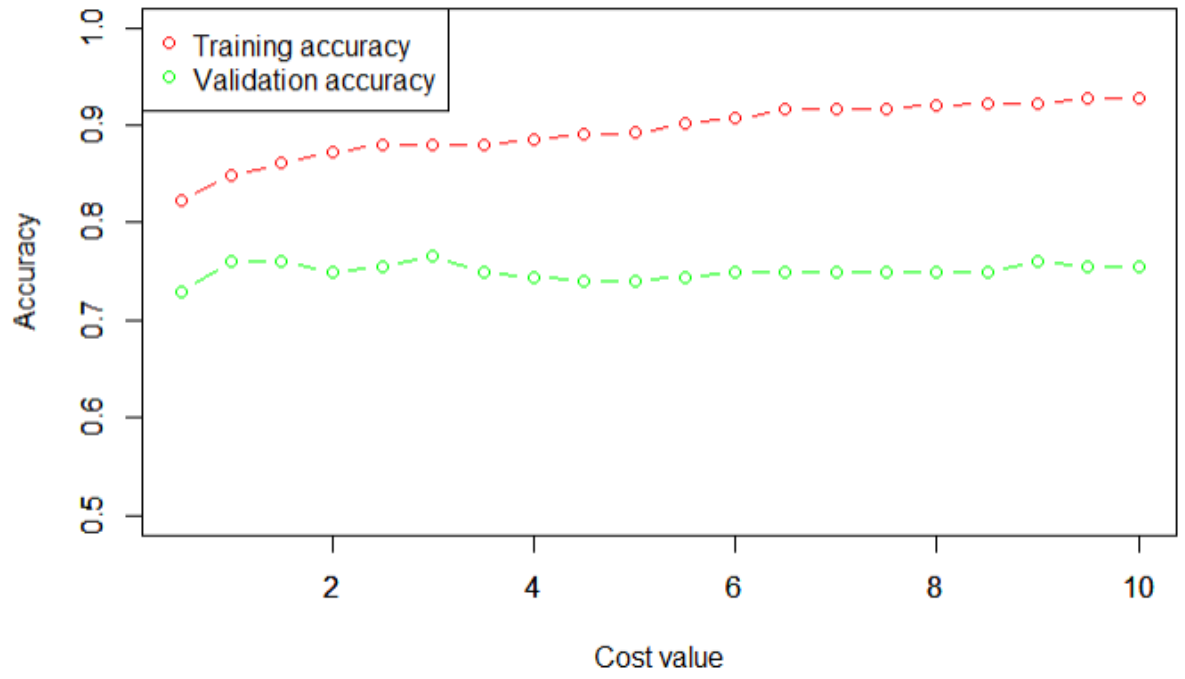
En aquest cas, observem que la matriu obtinguda és molt més bona, amb un valor d'accuracy alt per a les prediccions de les dues classes i bastant més homogeni que abans (90% per la negativa i 80% per la positiva).

SVM - KERNEL GAUSSIÀ

Finalment, fem servir el kernel gaussià. Aquest kernel presenta un millor rendiment si les dades tenen una distribució gaussiana, però, com hem vist amb el QDA, no sembla del tot clar que en el nostre cas sigui així. La fórmula del kernel gaussià és:

$$k(x, y) = exp\left(\frac{\gamma \cdot || x - y ||^2}{2}\right)$$

L'hiperparàmetre γ controla la variància, augmentant-la a mesura que aquest es fa petit. Tal com hem fet en el cas del kernel polinòmic, hem buscat el valor òptim a base de prova i error, obtenint γ = ¼ . Amb aquest valor, busquem el cost òptim utilitzant un altre cop el mètode de LOOCV. Obtenim els resultats següents:



Del gràfic observem que el cost segueix tenint un fort impacte en el valors d'accuracy de training i validació, i que aquests continuent sent lleugerament diferents. Per trobar el cost òptim fem servir el mateix procediment que hem explicat a l'apartat anterior. D'aquesta manera, escollim cost = 1, amb el qual obtenim un valor d'accuracy de validació prou alt (proper a un 77%) i un de training prou similar (inferior al 10%).

Podem veure que aquests valors són pràcticament idèntics als obtinguts prèviament pel kernel polinòmic. Per veure si els resultats també coincideixen per a les prediccions de cada classe, mirem la matriu de confusió.

```
          Reference
Prediction   N   P
         N 115  20
         P   9  48
```

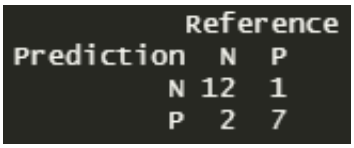
La matriu obtinguda és similar però presenta algunes diferències. Per una banda, els vidres amb classe negativa es classifiquen bé amb una precisió del 92%, mentre que en el kernel polinòmic obteníem un 90%. Per contra, ara només classifiquem bé un 70% de les observacions positives, mentre que abans ho fèiem per a un 80%. Per tant, aquest model millora lleugerament les prediccions negatives però empitjora significativament les positives.

MODEL ESCOLLIT I ERROR GENERALITZAT

Un cop analitzats els quatre models que hem plantejat, hem d'escollir quin és el millor de tots i calcular el seu error generalitzat. Per calcular l'error generalitzat senzillament fem les prediccions per a les nostres dades de test amb el model escollit.

Tant el model QDA com el SVM amb kernel lineal queden descartats sense cap mena de dubte, ja que obtenim un accuracy molt més petit que amb els altres dos models. D'aquesta manera, el dubte és si triar el SVM amb kernel polinòmic o gaussià. Tal com hem vist abans, obtenim un accuracy molt semblant pels dos kernels, però el polinòmic destaca en les prediccions de les observacions de classe positiva. Per tant, concluïm que el SVM amb kernel polinòmic i amb els paràmetres Q = 3, s = 1 i cost = 4 és el millor model per a aquestes dades.

L'error generalitzat obtingut és 0.136 o, dit d'una altre forma, l'accuracy del model és d'un 86,36%. Fent la matriu de confusió amb les dades de test obtenim que l'accuracy de la classe positiva és d'un 87% mentre que la de la negativa és d'un 85%. Així, comprovem que el model triat ens permet fer bones prediccions, ja que presenta un accuracy alt i molt homogeni entre les dues classes.



	Reference	
Prediction	N	P
N	12	1
P	2	7

CONCLUSIONS

En quant als models, hem comentat prèviament els resultats obtinguts per a cadascun i els hem comparat entre ells per tal de triar el més adient pel nostre dataset. En general, hem vist que la no separabilitat lineal de les dades limitava bastant l'eficiència dels models, en especial aquells menys complexos. També, en quant a la gaussianitat de les dades veiem que els models amb més dependència d'aquesta assumpció obtenien pitjors resultats. Així, podem concloure que les nostres dades són linealment no separables i no segueixen una distribució normal.

D'altra banda, ens hem trobat amb la limitació del nombre de dades, que era bastant petit i ha pogut condicionar bastant l'anàlisi general del dataset i, en conseqüència, els resultats obtinguts. Amb poques dades potser no es compleix massa bé la hipòtesi de variables independents idènticament distribuïdes (iid). Segurament, amb una millor mostra (amb un nombre d'observacions superior) podríem haver plantejat models més representatius que ens donessin resultats més precisos.

A més, com hem comentat abans, al ser una adaptació binaritzada d'un problema que originalment tenia 7 classes, hem unit 6 classes (amb les seves característiques corresponents) com a una de sola, possiblement amb valors bastant dispersos per a les diferents variables, fent més complicat que el model trobi els paràmetres que la diferencien de l'altre.

Finalment, malgrat les esmentades complicacions, hem obtingut un model capaç de fer prediccions amb una precisió superior al 85% que, tot i que podria millorar amb el que hem comentat (augmentant el número d'observacions, per exemple), és un resultat amb el que podem estar bastant satisfets.

REFERÈNCIES

https://www.openml.org/search?type=data&status=active&id=41 - Dades

https://datascienceplus.com/how-to-perform-logistic-regression-lda-qda-in-r/ - Diferència entre LDA i QDA

https://ca.wikipedia.org/wiki/%C3%8Dndex_de_refracci%C3%B3 - Índex de refracció

https://scikit-learn.org/stable/common_pitfalls.html - Inconsistent preprocessing