Pràctica APA:
Spotify All Time Top 2000s Mega Dataset

Júlia Alice Amenós Dien
Èric Ryhr Mateu

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat d'Informàtica de Barcelona

Índex

1. Introducció

El *dataset* que explorarem s'anomena *Spotify All Time Top 2000s Mega Dataset* i conté les cançons més populars a Spotify des de l'any 1956 fins l'any 2019. De cada cançó, conté informació textual com el títol, l'artista, l'any i el gènere i, a més a més, paràmetres arbitraris definits per Spotify com l'energia, la dançabilitat, la sonoritat, etc. També conté informació de la longitud de la cançó, la velocitat (BPM) i la popularitat. Es pot descarregar el conjunt de dades original desde la plataforma OpenML:
https://www.openml.org/search?type=data&status=active&id=43386&sort=runs

A partir d'aquesta informació, en un primer moment, vàrem plantejar la hipòtesi de si és possible ajustar un model de regressió per predir l'any de sortida d'una cançó (almenys dins d'un cert marge). Vam estudiar la idea de si incloure el gènere dins la nostra informació per realitzar la predicció o ignorar-lo. Si bé a priori sembla molt important, dins el *dataset* hi ha quasi 150 gèneres diferents. Això comporta que hi hagi molt poques cançons per gènere (alguns gèneres només tenen una cançó), i no sembla que ens aporti gaire informació. Un aspecte a discutir és si és possible agrupar els gèneres similars entre sí o si només tenim en compte els gèneres més populars.

Al veure que el problema de regressió resultava inviable (pels mals resultats obtinguts, veure Annex i codi al *notebook*), vam decidir canviar el nostre objectiu a un problema de classificació sobre determinar si una cançó és pre-2000 o post-2000. D'aquesta manera, vam poder aprofitar gran part del codi ja fet sobre la descripció i el preprocessament de dades.

2. Estudis previs

El nostre dataset ha estat utilitzat en altres treballs, on el focus de l'estudi es centra en la variable popularitat. El principal objectiu és classificar les cançons segons si són populars o no. Ara bé, no hem trobat cap estudi amb un problema semblant al nostre.

Hem fet una recerca i existeixen diversos estudis sobre la predicció de l'any de sortida d'una cançó. Si més no, a diferència del nostre *dataset*, disposen d'un major nombre de mostres de cançons, així com més variables que descriuen les peces musicals. Per exemple, un treball comptava amb més de 500.000 exemples i 90 atributs per cadascun d'ells[1]. En canvi, el nostre problema es centra en característiques més 'intuïtives' d'una cançó, fent-lo menys tècnic i més comprensible a l'hora d'extreure conclusions a nivell general.

3. Exploració de les Dades

El dataset conté 1994 cançons diferents i 14 atributs, 13 sense considerar la variable objectiu. Com s'ha mencionat anteriorment, la nostra finalitat és predir si la cançó és prèvia o posterior a l'any 2000.

3.1. Selecció d'atributs

A continuació es descriuen els 15 atributs del conjunt de dades:

- **Title**: Títol de la cançó
- **Artist**: Nom de l'artista
- **Top Genre**: Gènere
- **Year**: Any de sortida
- **Beats per Minute (BPM)**: Tempo
- **Energy**: Nivell d'energia
- **Danceability**: Nivell de ballabilitat, com més alt més fàcil de ballar.
- **Loudness**: Nivell d'intensitat sonora
- **Liveness**: Nivell de percepció d'una actuació en viu. Un valor alt, indica la presència d'aplaudiments o soroll d'audiència que dona la sensació de ser una actuació en directe.
- **Valence**: Nivell de valència emocional. Un valor alt indica que la peça musical es percep de manera positiva i alegre.
- **Length**: Durada (en segons)
- **Acoustic**: Nivell d'acústica.
- **Speechiness**: Quantitat de vocals presents en la peça musical.
- **Popularity**: Nivell de popularitat.

Tal i com hem mencionat anteriorment, la variable objectiu s'ha obtingut a partir de la columna 'Year' de la següent forma:

- **Is_recent:** Indica si la cançó va sortir abans de l'any 2000 (0) o després (1).

Els títols de la cançó son pràcticament valors únics i no aporten informació rellevant pel que es vol predir; per tant, els hem descartat directament. Pel que fa als artistes, tenir-los ens aporta moltíssima informació sobre els anys de sortida de les cançons. Ara bé, nosaltres volem veure si és possible diferenciar les cançons pre i post 2000, solament a partir de les seves propietats musicals. Per tant, els hem descartat també.

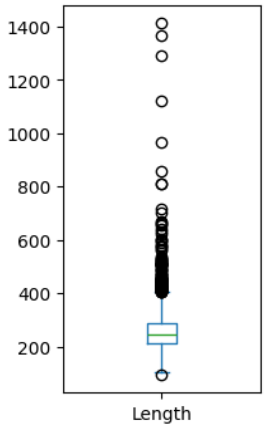
D'aquestes 13 variables, només 11 les tindrem en compte a l'hora d'entrenar els models: hem descartat l'índex, el títol, l'artista i l'any. És a dir, només inclourem les característiques que fan referència a la musicalitat de les cançons.

3.2. Missing values

El conjunt de dades no disposava de ningun valor perdut en cap atribut, així que no hem tingut problemes en aquesta part.

3.3. Outliers

Hem observat que disposem de diversos outliers per la majoria de les variables. Al principi no sabíem si es tractaven d'errors, però al analitzar i verificar aquests casos extrems hem vist que els valors son correctes. La que més variabilitat té és la durada, amb cançons de menys de 2 min fins a cançons de més de 23 min. Ara bé, atès que hi ha bastantes instàncies atípiques i nosaltres volem ser capaços de classificar qualsevol tipus de cançó, hem considerat que era informació rellevant i no hem descartat cap exemple.



3.4. Valors erronis

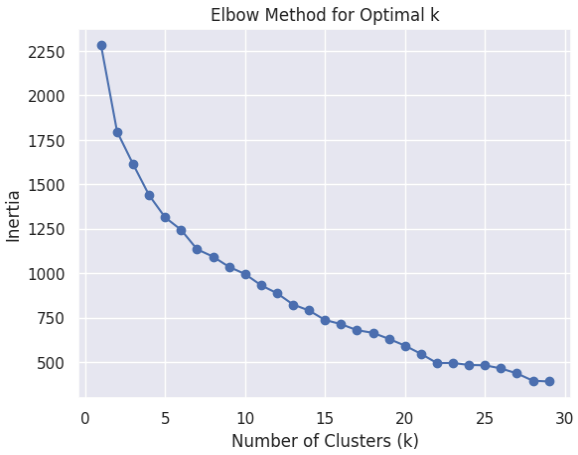
En el dataset inicial, teníem 4 instàncies que superaven els 1000 segons i els valors estaven codificats amb una coma en les mil·lèsimes. Això ens donava problemes a l'hora de llegir el dataset, així que vam cercar els valors i vam fer el canvi manualment.

3.5. Codificació de variables categòriques i extracció d'atributs

Totes les variables eren numèriques, exceptuant el Gènere. En total tenim 149 gèneres diferents, però alguns d'ells només inclouen una cançó. No voliem descartar la variable perquè realment aporta informació, però incloure els 149 gèneres amb OHE generaria una matriu molt dispersa i un augment significatiu de la dimensionalitat de les dades. Donat que el nostre conjunt de dades no és massa gran (2000 valors aprox), podria resultar problematic.

Una possible solució era reagrupar els valors poc freqüents en una única categoria. Ara bé, gèneres molt diferents estarien en el mateix grup: 'irish singer-songwriter', 'electronica', 'acid jazz', 'australian indie folk', etc. Si tinguéssim menys valors podríem haver reagrupat manualment les dades en gèneres més generals, com per exemple canviar '*uk pop'* a '*pop'*. Ara bé, també es donava algun cas on un subgènere podria pertànyer a més d'un gènere. En definitiva, l'última proposta que vam considerar va ser fer la reagrupació de gèneres mitjançant *clustering*.

Vam utilitzar l'algorisme *KMeans*, juntament amb l'*Elbow Method* per tal de seleccionar el número de clústers (*k*) òptim. En la gràfica podem veure que el punt d'inflexió no és massa clar, així que vam visualitzar els resultats per 5, 7, 9 o 10 clústers.

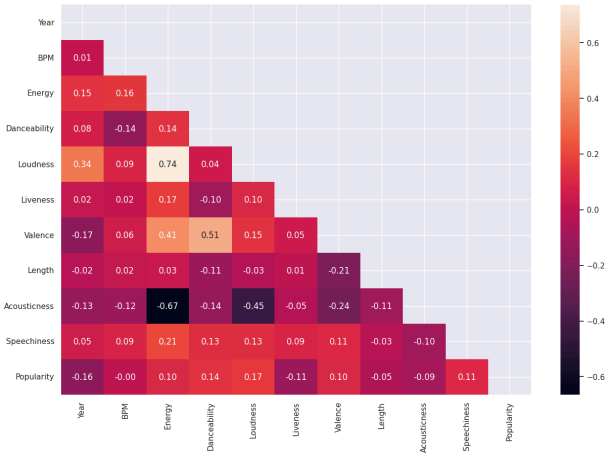


El resultat més coherent va ser l'agrupació en 7 clústers. Per conseqüent, vam eliminar la variable 'Genre' i vam afegir 'GenreCluster', que pren valors de l'1 al 7. Tot i ser números, aquests valors son categòrics, no nominals, per lo que posteriorment vam haver de transformar la variable amb *One Hot Encoding*. En definitiva, la codificació dels gèneres va afegir 7 noves variables al dataset, sumant un total de 17 atributs a tenir en compte en les prediccions.

Cluster 0: alternative rock, alternative dance, alternative metal, alternative pop rock, alternative country, german alternative rock, latin alternative, alternative hip hop
Cluster 1: album rock, art rock, dance rock, irish rock, glam rock, classic rock, modern rock, blues rock, modern folk rock, belgian rock, celtic rock, australian rock, rock-and-roll, garage rock, canadian rock, classical rock, classic canadian rock, hard rock
Cluster 2: cyberpunk, carnaval limburg, disco, neo mellow, electro, blues, reggae, eurodance, gabba, contemporary vocal jazz, classic soul, permanent wave, australian psych, downtempo, folk, europop, electropop, celtic, boy band, glam metal, reggae fusion, funk, britpop, latin, detroit hip hop, mellow gold, east coast hip hop, big beat, christelijk, canadian folk, metropopolis, contemporary country, classic schlager, g funk, big room, neo soul, arkansas country, scottish singer-songwriter, gangster rap, edm, atl hip hop, happy hardcore, electronica, alaska indie, bebop, australian americana, levenslied, irish singer-songwriter, nederpop, punk, indie anthem-folk, streektaal, stomp and holler, basshall, classic soundtrack, trance, finnish metal, laboratorio, australian indie folk
Cluster 3: dutch hip hop, dutch metal, dutch cabaret, dutch indie, dutch rock, dutch americana, dutch prog
Cluster 4: adult standards
Cluster 5: dutch pop, folk-pop, italian pop, dance pop, classic uk pop, pop, classic country pop, brill building pop, art pop, barbadian pop, german pop, belgian pop, chamber pop, baroque pop, danish pop rock, australian pop, la pop, acoustic pop, bow pop, bubblegum pop, candy pop, irish pop, classic italian pop, canadian pop, uk pop, hip pop, indie pop
Cluster 6: british invasion, british soul, british folk

3.6. Correlació entre variables

Podem veure que no sembla que hi hagi una gran correlació entre les característiques musicals i l'any de sortida de la cançó. De fet, les relacions no superen el 0.2, exceptuant l'atribut *Loudness*, amb un 0.34.



3.7. Escalat variables

L'últim pas ha estat normalitzar les dades abans d'entrenar els models, a fi d'evitar que siguin influenciats de manera desigual i dominats per les característiques de major magnitud. En alguns models també hem provat a estandaritzar les dades, però els canvis en el rendiment eren negligibles.

4. Protocol de remostreig

A fi d'obtenir un avaluació robusta del model, hem dividit el nostre dataset en dos conjunts *train* / *test*, agafant un 70% de les dades per l'entrenament i un 30% per la validació. Donat que tenim 2000 exemples aproximadament, utilitzar 1200 per entrenar i 600 per validar es una bona proporció. És important notar que estem treballant amb un conjunt desbalancejat: hi ha 1195 cançons pre-2000 i només 799 post-2000. Per tal de garantir que la proporció de classes es mantingui desbalancejada al fer la partició, hem utilitzat l'*split* estratificat. A l'hora de realitzar els *permutation importance* per veure els atributs més importants per cada model, els realitzem a partir de 500 mostres escollides aleatòriament del conjunt de test.

A l'hora d'ajustar els models, hem fet servir el conjunt de train per entrenar-los i trobar els millors hiperparàmetres. En general, el procediment utilitzat ha estat fer servir *BayesSearchCV* (o *GridSearchCV* si el temps d'entrenament era reduït) amb rangs de valors amplis i anar-los reduint gradualment, aconseguint així els hiperparàmetres òptims. Per mitigar l'*overfitting* i obtenir puntuacions que reflecteixin realment la generalització del model, hem utilitzat *K-fold cross validation*. Segons el temps d'entrenament per cada model, hem anat variant entre 5 i 10 subgrups *(K)*. Finalment, mencionar que degut al desbalanceig enter classes, hem fet servir la puntuació f1-score, que mesura el balanceig entre precisió i recall per les classes. Utilitzar l'*accuracy* no seria correcte, ja que podria ser que el model classifiqués tot cap a la classe majoritària i obtenir bons resultats. Finalment, per validar el resultat final de cada model amb els seus millors hiperparàmetres, hem utilitzat la partició de test.

Convé notar que hem especificat un estat random a l'hora de crear la partició, per poder reproduir els resultats i fer comparacions justes.

5. Models lineals/polinòmics

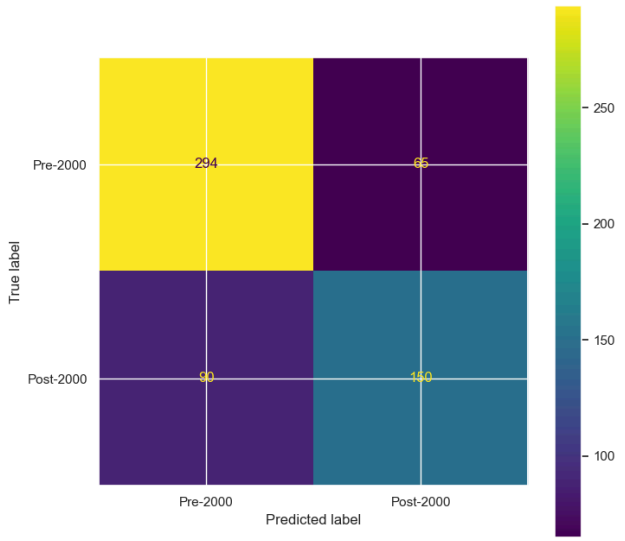
Com ja hem explicat abans, farem servir el f1-score sobre el conjunt de test per validar i puntuar els nostres models.

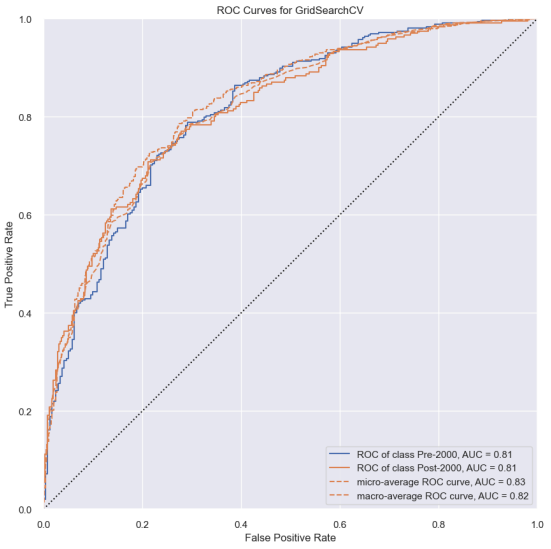
5.1. *Linear Discrimination Analysis* (LDA)

Començarem aplicant un mètode de discriminació lineal a les nostres dades. Es tracta d'un model poc costós de recursos i temps computacional que ens ha semblat adient com a punt de partida. Hem realitzat un entrenament de cerca en graella sobre els següents paràmetres per trobar el millor model:

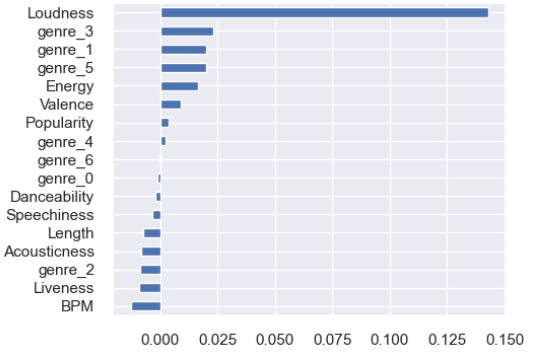
- **solver**: svd, lsqr, eigen
- **shrinkage**: Null, auto, 0.1, 0.2, 0.5

Com a millors paràmetres hem obtingut *solver: svd* i *shrinkage: None* amb una f1-score de 0.725 sobre el conjunt test, el qual sembla un bon resultat com a punt de partida. Veiem com distribueix les nostres dades:





Veiem com aconsegueix classificar amb més èxit les cançons pre-2000 que les posteriors, segurament degut al fet de que tinguem un dataset desbalancejat com ja hem explicat abans. Comprovarem si amb un altre model aconseguim solucionar aquest problema. A la corba *Receiver Operating Characteristic* (ROC) podem veure que les àrees sota la corba (AUC) de les dues classes són similars i per les dues podem obtenir al voltant de 75% de veritables positius a canvi de 25% de falsos positius. Per últim, podem veure quin pes dona aquest model als diferents atributs del conjunt de dades. Podem veure que amb diferència l'atribut predominant és la sonoritat seguit del clúster de gèneres 3 (*dutch music*).

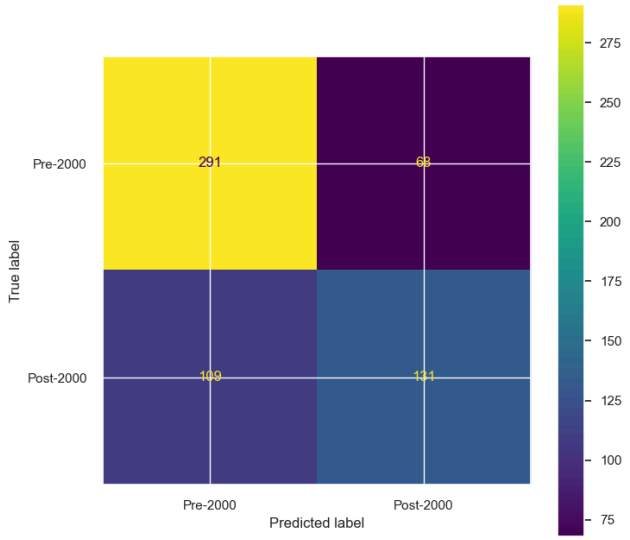


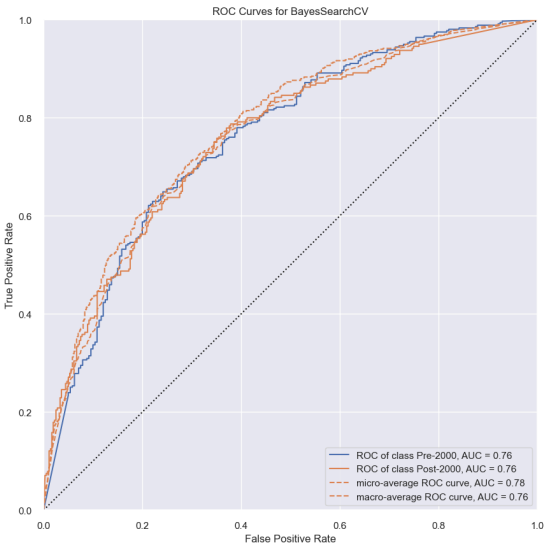
5.2. *K-Nearest Neighbors* (KNN)

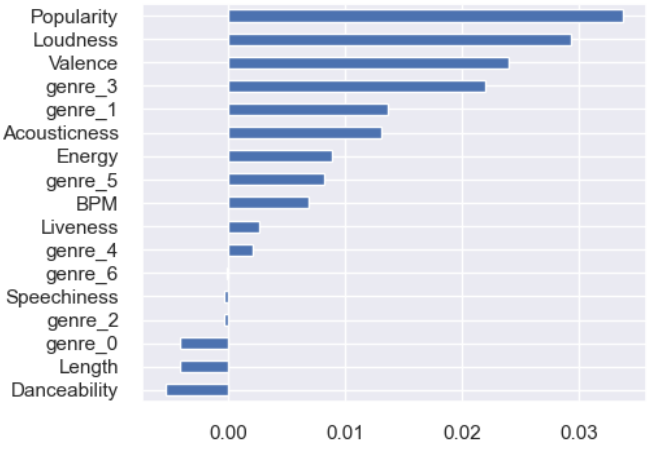
A continuació, provarem un model *K-Nearest Neighbors*. És un model més costós que l'anterior computacionalment i veurem si ens ajuda a millorar les prediccions sobre la classe post-2000. Provarem els següents paràmetres fent una cerca bayesiana amb 15 iteracions (si féssim una cerca en graella trigaríem massa i probablement no milloraria massa el resultat):

- **n_neighbors**: 1, 3, 5, 7, 11, 15
- **weights**: distance, uniform
- **leaf_size**: 1, 5, 10, 20, 30
- **metric**: l2, l1, cosine

Com a millors paràmetres hem obtingut *leaf_size: 20*, *metric: l1*, *n_neighbors: 5* i *weights: distance*, amb una puntuació f1 de 0.680 sobre el conjunt test. El resultat és bastant decepcionant, i més observant com distribueix les nostres mostres.







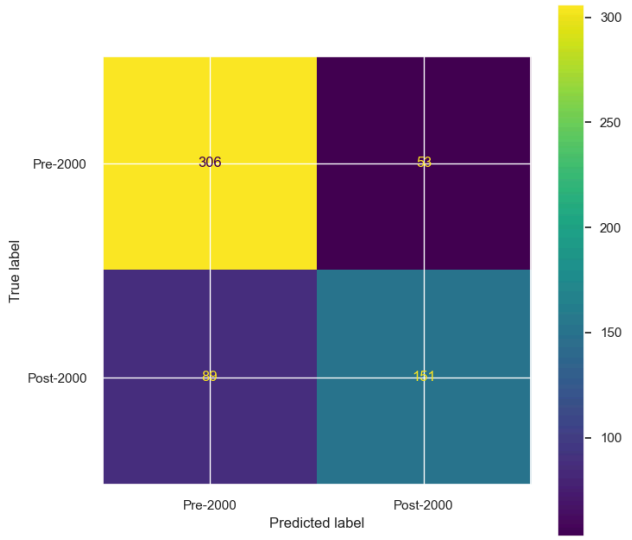
Veiem a la matriu de distribució que per la classe pre-2000 tenim una distribució similar al model anterior però per la classe post-2000 tenim un resultat pitjor, amb el model equivocant-se per quasi la meitat de les mostres. Pel que fa a la corba ROC, com era d'esperar tenim un resultat pitjor que el model anterior amb una AUC més baixa de només 0.75. Si veiem la importància que dona als atributs, veiem que són valors insignificants que no semblen aportar massa. En conclusió, KNN no sembla ser un model adient pel nostre problema.

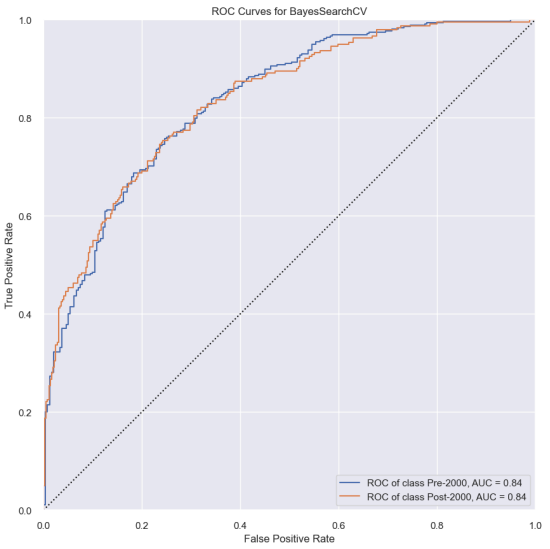
5.3. SVM polinòmic

Per últim, per acabar els models lineals, provarem el model *Support Vector Machine* (SVM) en el seu format polinòmic. Els SVM solen funcionar prou bé quan el nombre d'atributs es escàs, atès que s'augmenta la dimensionalitat de l'espai per poder obtenir hiperplans que separen millor les dades. Com l'anterior, és un model costós d'entrenar. Provarem els següents paràmetres en una cerca bayesiana idèntica a la d'abans:

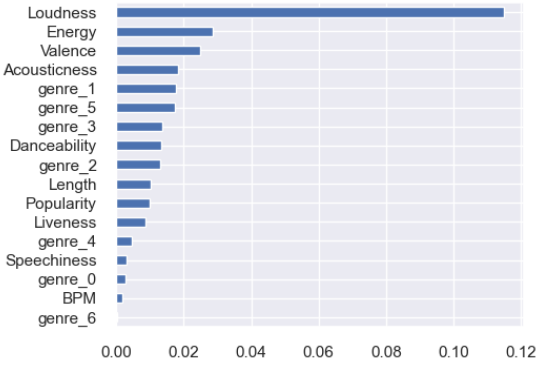
- **C**: 101 valors linealment espaiats entre 10^{-3} i 10^{3} inclosos
- **degree**: 2, 3

Els millors paràmetres obtinguts han estat *C: 1.148* i *degree: 3* amb una puntuació f1 de 0.746 sobre el conjunt test, el millor resultat fins ara. Podem veure que tenim una clara millora sobre el model anterior KNN i una petita millora sobre el primer LDA. Observem si hem aconseguit també millores a l'hora de classificar la classe post-2000.





Veiem uns resultats molt semblants als del model LDA, amb facilitat per classificar la classe pre-2000 i dificultats per classificar la classe minoritària. Pel que fa a la corba ROC, veiem un resultat similar amb un AUC de 0.84 per les dues classes. Pel que fa a la importància dels atributs, la sonoritat segueix sent el més important però seguida ara de l'energia i la valència o positivitat. De totes maneres, sembla que els models lineals tenen dificultats amb aquest conjunt de dades, veurem si els models no lineals tenen més facilitats.



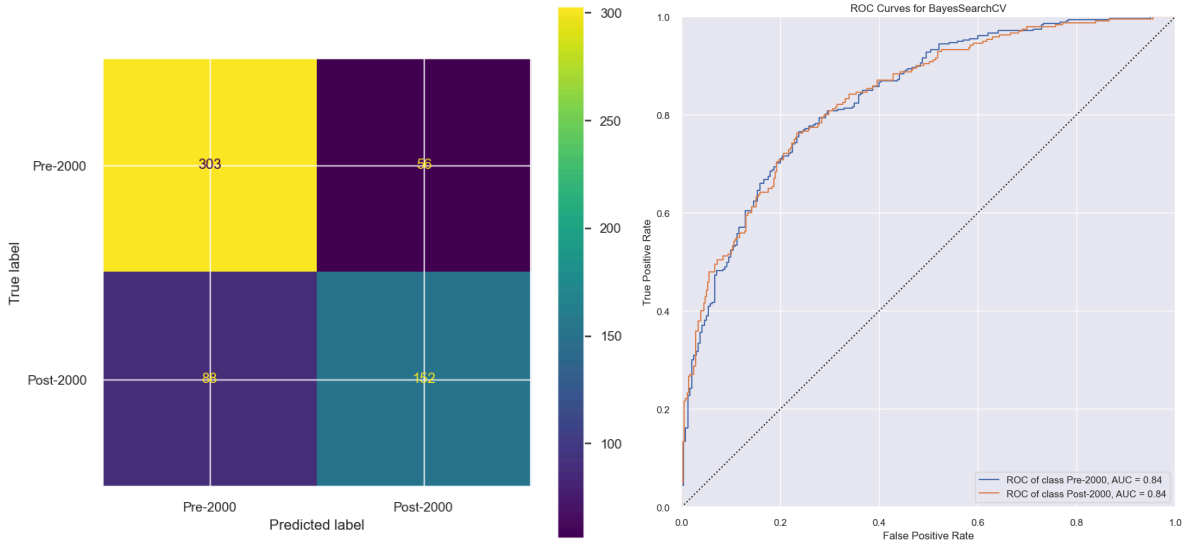
6. Models no lineals

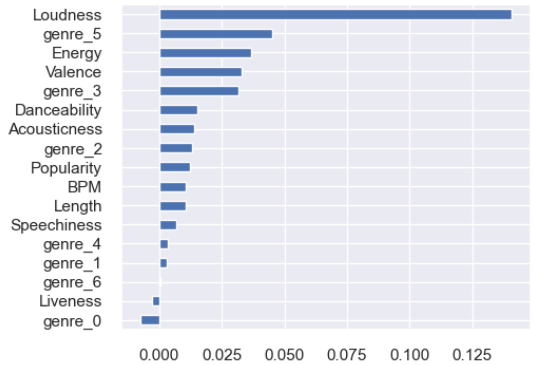
6.1. SVM RBF

Entrenarem de nou un SVM, però utilitzant un kernel RBF. En aquest cas tenim dos paràmetres: C i Gamma. La Gamma controla la influència que té cada un dels exemples d'entrenament a l'hora de classificar.

- **C**: 101 valors linealment espaiats entre 10^{-3} i 10^{3} inclosos. Després 50 entre 10^{-1} i 10^{1}.
- **Gamma:** 'auto' o 'scale'.

Els model que millor s'ajusta a les dades posseeix els paràmetres *C: 125.9* i *gamma: auto* i obté un 0.743 de f1-score pel conjunt test, una mica pitjor que l'anterior SVM polinòmic. A continuació es mostra la matriu de confusió per conjunt de validació:





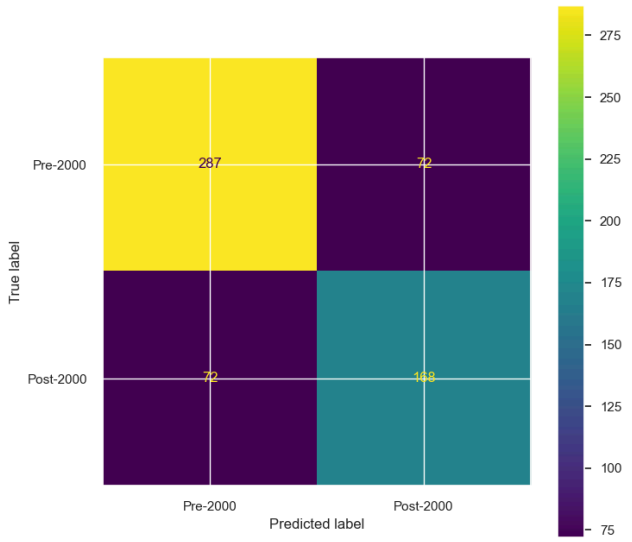
Comparant-lo amb el SVM de kernel polinòmic, podem veure que aconseguim un cert augment de la precisió del post-2000, en compensació amb una petita disminució de la precisió de pre-2000. Per les corbes ROC, tenim bons resultats amb un 0.84 d'àrea. En ambdós casos, aconseguim un 75% dels True Positives amb una mica més del 20% de falsos positius. Finalment, observant la importància dels atributs, podem veure que la categoria *Loudness* torna a destacar notablement per sobre de les altres: és a dir, el model depèn moltíssim d'aquesta característica a l'hora de classificar les instàncies.

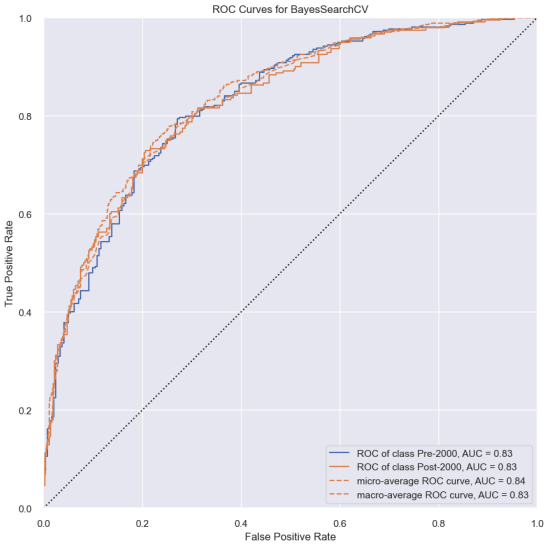
6.2. MLP

El següent model a entrenar és un perceptró multicapa. Hem normalitzat les dades utilitzant l'escalat MinMax. Aquest model posseeix diversos hiperparàmetres a explorar:

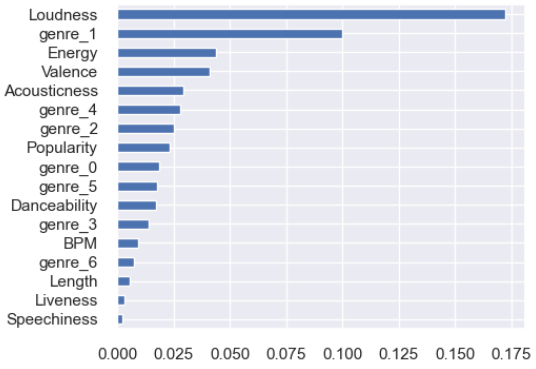
- **Hidden_layer_sizes:** Hem explorat 9 arquitectures diferents, utilitzant una capa des de 2 fins a 200 neurones.
- **Activation:** relu, logistic, identity i tanh.
- **Learning_rate_init:** 10 valors linealment espaiats entre 10^{-5} i 10.

Hem utilitzat una cerca bayesiana per estimar els millors resultats: *activation: logistic, hidden_layer_sizes: 10* i *learning_rate_init: 0.1*. Hem aconseguit una puntuació f1-score de 0.749 amb el conjunt test, la millor puntuació fins ara. Observem la matriu de confusió per veure el perquè:





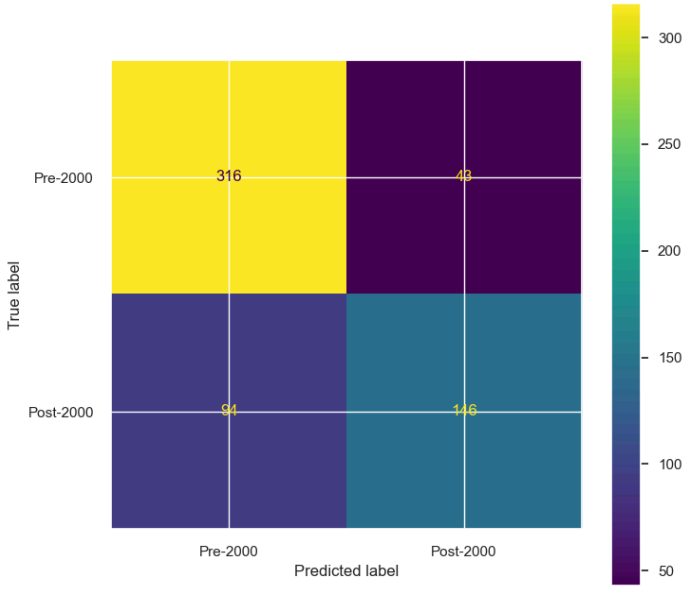
Mirant la matriu de confusió podem veure que hem aconseguit el model que millor prediu la classe post-2000 a canvi de ser el que pitjor prediu la classe pre-2000. A la curva ROC tenim una àrea de 0.83, un xic pitjor que els anteriors però pot sortir a compte si de veritat volem aquesta millora a la classe minoritària. Per últim, veiem que l'atribut *Loudness* segueix sent el més important però aquest cop tenim molta importància al *genre 1*, que es tracta del clúster de rock. Sembla que té molta importància per predir la classe post-2000.



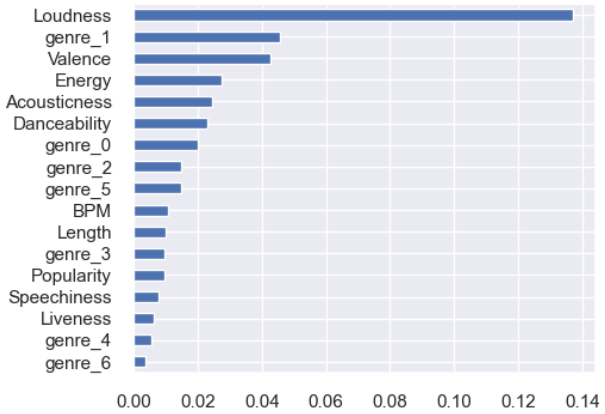
6.3. Voting Classifier (SVM POLINÒMIC + MLP)

Per últim, en aquest apartat utilitzarem una combinació dels dos models que han obtingut els millors resultats per les nostres dades. Vam provar tant un *Stacking Classifier* com un *Voting Classifier*, que combinava el SVM de kernel Polinòmic i el MLP. El millor resultat el va obtenir el combinador per majoria de vot, on el resultat f1-score sobre el conjunt de test fou de 0.751, l'únic model que ha aconseguit superar el 75%.

Mencionar que donat que hem utilitzat un SVM, retornem la classe predita directament, sense probabilitats. És per aquest motiu que no hem pogut utilitzar una votació suau. La següent imatge mostra la matriu de confusió obtinguda:



El recall de les dues classes son de 0.77, però per la classe majoritària obtenim major precisió que la minoritària. D'aquest model és interessant analitzar la rellevància dels atributs. Com tots els models, la variable *Loudness* té una importància significativa en comparació amb la resta. Ara bé, els següents atributs amb major rellevància son el clúster de gèneres 1 (música rock), valència o positivitat i energia, que corresponen a la combinació de les importàncies dels dos models usats, SVM polinòmic i MLP.



7. Model final

A fi de decidir quin model s'adequa millor al nostre problema, farem una recapitulació dels resultats obtinguts. En el plot apareixen més models dels 6 que hem explicat en aquest informe. Teniem l'ambició d'intentar aconseguir un model que arribés com a mínim a una puntuació del 80%, així que vam anar provant diversos models, cosa que ha suposat tot un repte. També voldríem mencionar que, tenint en compte la poca correlació que hi havia entre els atributs i la variable objectiu, estem molt satisfets amb els resultats assolits.

	train XV acc	test acc	test f1 score (0)	test f1 score (1)	test f1 score (W)
Voting (QSVM + MLP)	NaN	0.771285	0.821847	0.680653	0.751250
MLP	0.756882	0.759599	0.799443	0.700000	0.749721
SVM Polinomic	0.768006	0.762938	0.811671	0.680180	0.745926
Stacking (QSVM + MLP)	NaN	0.761269	0.810596	0.677201	0.743898
SVM RBF	0.767354	0.759599	0.808000	0.678571	0.743286
SVM Lineal	0.747882	0.746244	0.794595	0.668122	0.731358
Stacking (RBF + LDA)	NaN	0.749583	0.801587	0.660633	0.731110
Random Forest	0.767240	0.749583	0.805699	0.647887	0.726793
LDA	0.749255	0.741235	0.791386	0.659341	0.725363
Voting (RBF + LDA)	NaN	0.741235	0.801027	0.630072	0.715549
Gradient Boosting	0.757650	0.722871	0.783854	0.613953	0.698904
GNB	NaN	0.694491	0.726457	0.654064	0.690261
QDA	0.719291	0.692821	0.728614	0.646154	0.687384
KNN	0.726731	0.704508	0.766798	0.596811	0.681805

Els f1-scores mostrats pertanyen als obtinguts pels models en el conjunt de validació. Podem veure que els pitjors resultats (entre el 68% i 69% d'f1-score) van ser obtinguts pel KNN i el QDA. Donat que ambdós models son lineals, considerem que les nostres dades no son del tot linealment separables i per tant, es produeixen més errors de predicció. Ara bé, el SVM de kernel polinòmic ha aconseguit resultats molt satisfactoris, amb un 74.6% d'f1-score, fent-lo el tercer millor model entrenat. Això es deu a que ens permet capturar relacions polinòmiques, cosa que el fa més flexible que els dos models lineals mencionats prèviament.

Centrant-nos en els models no lineals descrits en l'informe, podem veure que tots obtenen entre el 74% i el 75% d'f1-score. Ara bé, voldríem comentar que la puntuació del Gradient Boosting ens ha sorprès bastant. Hem fet una cerca prou amplia per trobar els millors paràmetres, però tot i així no s'ha aconseguit superar el 70%.

Finalment, com hem comentat anteriorment, el model que ha obtingut major puntuació és el Voting Classifier entre el SVM de kernel polinòmic i el MLP. Si ens fixem, un model és lineal i l'altre no, i probablement per aquest motiu la combinació dels dos funciona tan bé.

8. Interpretabilitat

Per realitzar l'anàlisi d'interpretabilitat hem escollit els següents models. D'entre els lineals hem agafat el LDA per ser un model molt poc costós computacionalment i d'entre els no lineals el MLP, per ser el que millor aconseguia classificar la classe post-2000.

8.1. Anàlisi de valors atípics

Per l'anàlisi de valors atípics primer provarem amb la cançó més llarga que tenim al dataset, la qual es tracta de *Echoes* de *Pink Floyd* i té les següents característiques:

	BPM	Energy	Danceability	Loudness	Liveness	Valence	Length	Acousticness	Speechiness	Popularity	is_recent	genre_0	genre_1	genre_2	genre_3	genre_4	genre_5	genre_6
225	134	32	28	-17	11	14	1412	37	4	58	0	0	1	0	0	0	0	0

A més a més, donat que l'atribut *Loudness* sembla ser el que més ha influenciat els nostres models, provarem les cançons amb la sonoritat més baixa i amb la més alta, que resulten ser *The First Time Ever I Saw Your Face* de *Roberta Flack* i *I'm going home* de *Ten Years After* respectivament. Tenen els següents atributs:

	BPM	Energy	Danceability	Loudness	Liveness	Valence	Length	Acousticness	Speechiness	Popularity	is_recent	genre_0	genre_1	genre_2	genre_3	genre_4	genre_5	genre_6
144	121	3	31	-22	16	14	261	73	4	64	0	0	0	0	0	1	0	0

	BPM	Energy	Danceability	Loudness	Liveness	Valence	Length	Acousticness	Speechiness	Popularity	is_recent	genre_0	genre_1	genre_2	genre_3	genre_4	genre_5	genre_6
241	117	93	38	-2	81	40	639	18	10	26	1	0	1	0	0	0	0	0

Per últim, escollirem dues cançons aleatòriament amb el mateix valor de sonoritat (per així no poder influenciar la decisió dels models per aquest atribut) i que pertanyin a classes diferents. Per exemple, *Human Nature* de *Michael Jackson* (pre-2000) i *Love Yourself* de *Justin Bieber* (post-2000). Tenen les següents característiques:

	BPM	Energy	Danceability	Loudness	Liveness	Valence	Length	Acousticness	Speechiness	Popularity	is_recent	genre_0	genre_1	genre_2	genre_3	genre_4	genre_5	genre_6
23	93	51	62	-10	26	69	246	54	3	65	0	0	0	0	0	0	1	0
26	100	38	61	-10	28	52	234	84	44	83	1	0	0	0	0	0	1	0

A continuació veurem com han classificat els models aquestes cançons:

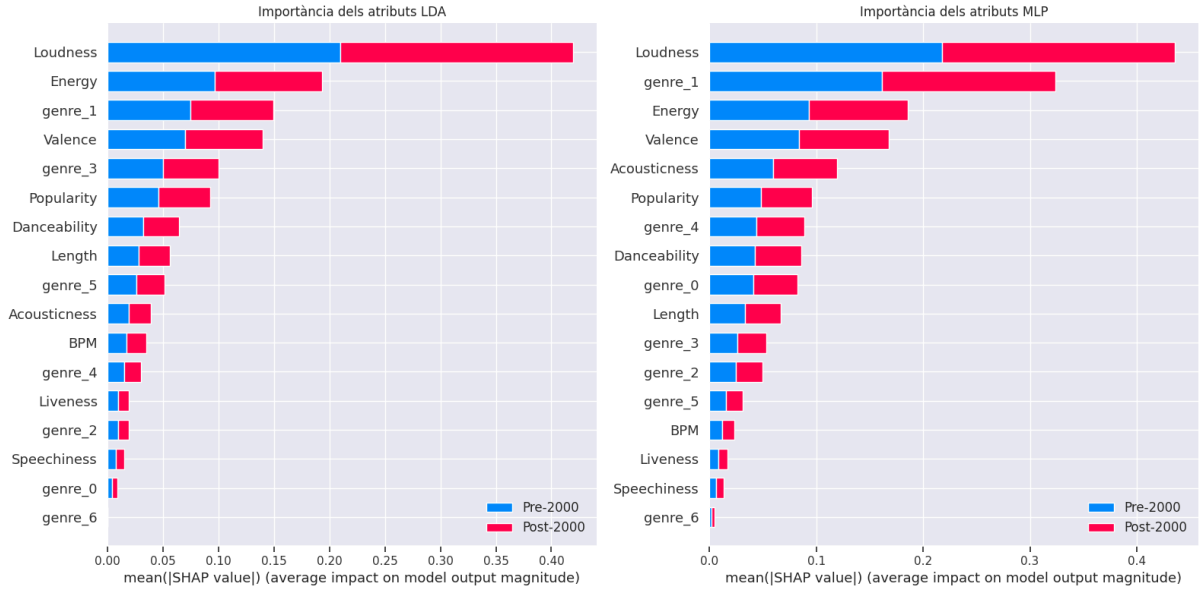
Cançó	Real	LDA	MLP
Echoes	0	0	0
The First Time Ever I Saw Your Face	0	0	0
I'm going home	1	1	1
Human Nature	0	0	0
Love Yourself	1	1	1

Veiem que cap dels dos models ha tingut dificultats per classificar aquestes cançons correctament. Així doncs, on no coincideixen?

8.2. Anàlisi de divergència

Realitzarem un anàlisi on divergeixen els dos models per intentar comprendre en què es fixa cadascun a l'hora de classificar les cançons i veure quines conclusions en podem extreure. Primer de tot, veurem en quins casos divergeixen. De tot el conjunt test de 599 mostres, els models no es posen d'acord en 65. D'aquests 65, 27 són bons per part del model LDA i 38 per part del model MLP.

A continuació veurem quina importància donen els models als atributs per aquests casos divergents. Com és d'esperar, al haver només dues classes, els models donen la mateixa proporció d'importància a cadascuna.



Veiem que, primer de tot i com era d'esperar pels *permutation importance*, l'atribut *Loudness* té molt d'impacte en ambdós models. Veiem que la resta dels atributs de l'LDA canvien molt respecte a quan l'hem analitzat anteriorment, mentre que el MLP manté si fa no fa el mateix ordre d'atributs, amb l'única diferència que dona molta més importància al clúster de gèneres 1 (els de rock), donant-li quasi tanta importància com a la sonoritat. En canvi, l'LDA prioritza més l'energia de la cançó, almenys per aquests 65 exemples.

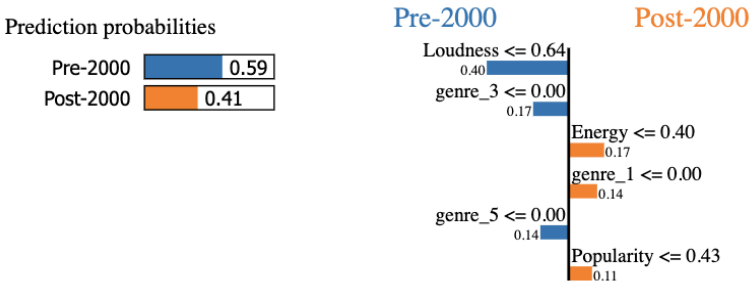
8.3. Anàlisi d'un exemple concret

D'aquests 65 exemples, analitzarem un exemple escollit aleatòriament, mitjançant LIME, una tècnica que ens permet interpretar les decisions preses pels models. La cançó que analitzarem és Post-2000, i el MLP el classifica correctament, mentre que el LDA no.

	BPM	Energy	Danceability	Loudness	Liveness	Valence	Length	Acousticness	Speechiness	Popularity	genre_0	genre_1	genre_2	genre_3	genre_4	genre_5	genre_6
265	0.955414	0.333333	0.209302	0.56	0.123711	0.3125	0.087912	0.622449	0.037736	0.247191	0.0	0.0	1.0	0.0	0.0	0.0	0.0

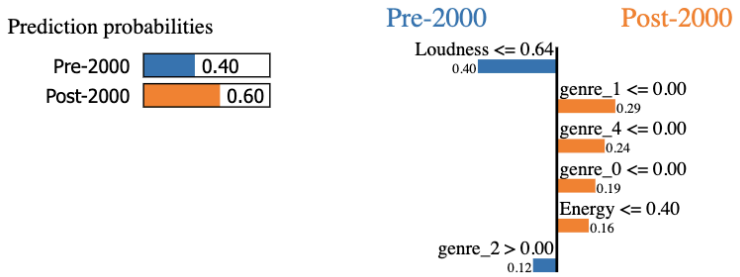
Per LDA, podem veure que tot i estar equivocat, el model indica amb una probabilitat del 0.59 que la mostra és Pre-2000. L'atribut que té major influència és la sonoritat, seguida d'altres atributs com els clústers de gèneres 3, 5 i 1, l'energia i la popularitat. És interessant notar que, com hem mostrat en la *permutation importance*, son variables que tenen un major impacte en el model. D'aquesta mostra podem extreure les següents conclusions sobre l'aprenentatge de LDA:

- Una Loudness menor que el 0.64 indica que son cançons Pre-2000.
- Els gèneres dels clústers 1 solen ser de cançons Pre-2000. Veiem que el fet que la mostra no pertanyi al grup 1, incita el model a dir que l'exemple no és Pre-2000, sino Post-2000.
- Els gèneres dels clústers 3 i 5 solen ser cançons Post-2000.
- Una energia i popularitat petita indica cançons Post-2000.



Recordar que les variables estan normalitzades entre 0 i 1. També convé remarcar que això es informació que el model ha après durant el seu entrenament, i que pot no correspondre amb la realitat. Per l'MLP, el model indica amb una probabilitat del 0.6 que la peça musical és Post-2000. En comparació amb el model anterior, la predicció no està influenciada per la popularitat i els clústers dels gèneres tinguts en compte son diferents. Durant l'entrenament, el MLP ha tret les següents conclusions:

- Una Loudness menor que el 0.64 indica que son cançons Pre-2000.
- Els gèneres del clúster 2 solen ser cançons Pre-2000.
- Els gèneres dels clústers 1, 4 i 0 solen indicar cançons Pre-2000. El fet que la cançó no ho sigui, confon el model indicant que podria ser Post-2000.
- Una energia petita indica cançons Post-2000.



Veiem que els dos models han après atributs força semblants, de fet, la informació extreta a partir de la sonoritat, l'energia i el clúster 1 son exactament iguals pels dos. Tanmateix, podem veure l'efecte de la importància dels atributs dels dos models en la predicció: el fet que l'MLP doni més pes als gèneres 4 i 0 permeten classificar correctament la mostra, mentre que l'LDA s'equivoca perquè dona més pes als gèneres 3 i 5.

9. Conclusions

Autoavaluació d'èxits, fracasos i dubtes.

La realització d'aquest estudi ha comportat diversos reptes que comentarem a continuació. En un primer moment, la nostra idea era intentar predir l'any de sortida de la cançó a partir de les seves característiques musicals. Érem ben conscients de que era un problema bastant difícil, sobretot tenint en compte la limitació en el nombre d'atributs del nostre dataset. No érem capaços d'assolir molts bons resultats (puntuacions màximes del 40%), produint errors mitjans de predicció de 12 anys. Per consegüent, vam buscar una alternativa al nostre objectiu, mantenint l'essència del que volem predir, però modificant el problema a un de classificació: determinar si la cançó és pre-2000 o post-2000.

A més a més, el nostre dataset comptava amb altres inconvenients, com la variable Gènere, que disposava d'un gran nombre de categories, però era informació rellevant que no voliem eliminar. Un altre punt important fou el desbalanceig entre classes. Veient que la majoria de models es centraven significativament en una única variable, vam provar de fer *undersampling* o *oversampling* de les dades per a l'entrenament dels models, però no s'observaven canvis significatius.

Conclusions científiques i personals

Com bé hem explicat abans, no hem pogut derivar amb èxit l'any d'una cançó a partir de les característiques de les que disposàvem, proporcionades per Spotify. Una possible conclusió que es pot treure d'això és que al llarg del temps apareixen cançons populars si fa o no fa de tots els gèneres quasi anualment, cosa que fa difícil saber si una cançó de, per exemple, *funk* es tracta d'una dels anys 70 als seus orígens o es tracta de *funk* modern. Tot i això, sí que hem pogut veure una variable significativa, la sonoritat (*loudness*). Cada cop més en els anys recents tenim música més comprimida, amb el volum percebut més elevat per provocar una sensació d'eufòria immediata a l'oient de seguida que comencen a escoltar una cançó[4]. Personalment, creiem que la primera conclusió és vàlida a mitges. Si bé es treuen cançons de tots els gèneres anualment, seran populars només les dels gèneres de moda o populars o bé d'artistes famosos. Pel que fa a la anomenada *Loudness War*, hem pogut percebre els efectes personalment escoltant sobretot les cançons més populars al llarg dels anys.

Possibles extensions i limitacions conegudes

A partir del nostre estudi, hem pogut extreure que un dels trets que diferencia significativament la música d'abans i després del 2000 és la intensitat sonora de la cançó. Podria ser que per altres llindars d'anys, la variable significativa fos una altra. El problema es podria extendre a veure a quina dècada pertany la cançó, fent d'això un problema de classificació multiclasse. A més a més, podríem veure tendències musicals entre anys i predir per exemple, com evolucionarà la música i determinar quines característiques han de tenir les cançons en un futur per a ser populars.

10. Referències

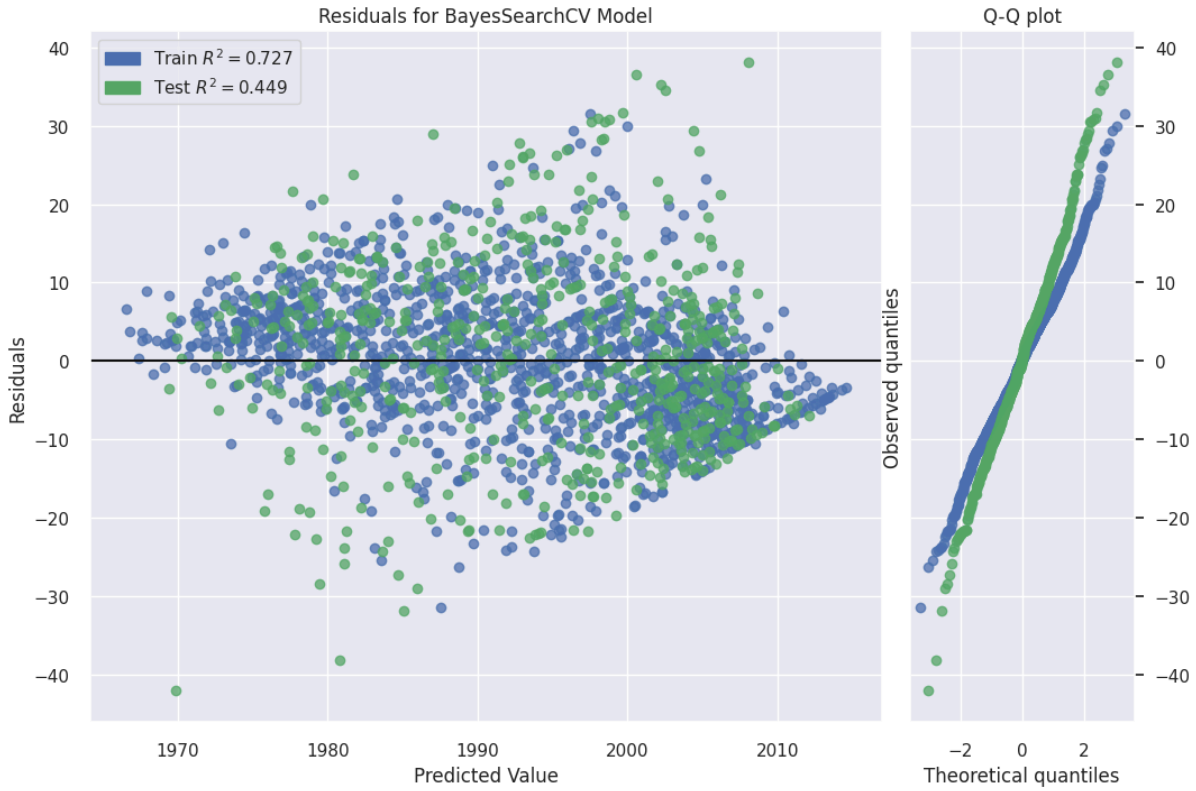
- [1]: Rasmus Salmon & Haider al Saadi. (2020). Predicting release year of songs. https://www.nbi.dk/~petersen/Teaching/ML2020/FinalProject/FinalProject7_HaiderRasmusMS_PredictingMusicPublicationYear.pdf
- [2]: Predicting songs release year using linear regression - Databricks. (n.d.). https://databricks-prod-cloudfront.cloud.databricks.com/public/4027ec902e239c93eaaa8714f173bcfc/3175648861028866/3427136292952836/657465297935335/latest.html
- [3]: Year Prediction - Million Songs Database. (n.d.). T. Bertin-Mahieux. https://samyzaf.com/ML/song_year/song_year.html
- [4]: Wikipedia contributors. (2023c, November 9). Loudness war. Wikipedia. https://en.wikipedia.org/wiki/Loudness_war

11. Annex

El problema de regressió que vam plantejar inicialment intentava predir l'any exacte de sortida de la peça musical. Ara bé, predir l'any exacte era força complicat, aconseguint un error mitjà de 12 anys respecte la predicció real. El millor model obtingut, el qual es tractava d'un *Random Forest* tenia els següents resultats:

- Puntuació de validació creuada:
  Train R2 score: 0.4371
  Test R2 score: 0.4098
  Mitjana d'anys d'error: 12.05

- Puntuació de residus:



Ara bé, tot i buscar els millors hiperparàmetres, veiem que el model obté una puntuació prou alta, de 0.72, pel conjunt de d'entrenament (sense cross validation), però de 0.45 pel conjunt de test. Això indica que el model està sobreajustant-se i no és gaire robust a l'hora de predir dades no vistes.