



UNIVERSITÀ DI PISA

Corso di Laurea Magistrale in Data Science and Business Informatics

RAVDESS

Andreazzoli Camilla
Fabbri Lucia
Ferri Diletta

ANNO ACCADEMICO 2022-2023

Indice

1	Data Understanding & Preparation	1
1.1	Data Semantics	1
1.2	Distribuzione delle variabili e statistiche	2
1.3	Assessing data quality	2
2	Clustering	5
2.1	k-means	5
2.1.1	Scelta del miglior k	5
2.1.2	Descrizione e sintesi dei risultati	6
2.1.3	k-means limitato ad alcune features	7
2.1.4	k-means con trasformazione delle features categoriche	7
2.2	Density Based Clustering	8
2.2.1	Scelta dei parametri DBSCAN	8
2.2.2	Descrizione e sintesi dei risultati DBSCAN	9
2.2.3	OPTICS	9
2.3	Hierarchical clustering	9
3	Classification	11
3.1	Decision Trees	11
3.1.1	Variabile target <i>emotion</i>	11
3.1.2	Variabile target <i>sex</i>	12
3.2	K-NN	13
3.2.1	Variabile target <i>emotion</i>	13
3.2.2	Variabile target <i>sex</i>	14
3.3	Naive Bayes	15
3.3.1	Variabile target <i>emotion</i>	15
3.3.2	Variabile target <i>sex</i>	16
3.4	Confronto finale	17
4	Pattern Mining	18
4.1	Estrazione di Frequent Pattern	18
4.2	Estrazione di Association Rules	19
4.2.1	Association rules per predire la variabile target	20

Capitolo 1

Data Understanding & Preparation

Il dataset che andremo ad analizzare è *Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS)*: esso contiene 2452 audio registrati da 24 attori professionisti, equamente suddivisi per sesso, con un accento nordamericano neutrale. A ciascun attore viene richiesto di pronunciare una frase oppure una canzone, in modo tale da esprimere una data emozione. Lo scopo della nostra indagine è quello di scoprire ed analizzare le relazioni presenti tra i vari record del dataset e proseguire con la classificazione di una delle features presenti. Il lavoro svolto sarà quindi descritto all'interno dei seguenti capitoli: *Data Understanding & Preparation, Clustering, Classification e Pattern Mining*.

1.1 Data Semantics

In questo primo capitolo andremo ad eseguire un'analisi esplorativa dei dati a nostra disposizione per capirne la natura e la distribuzione; successivamente verranno eseguite tutte le operazioni di pre-processing necessarie per ottenere un dataset più pulito e agevole. Per comodità abbrevieremo Mel-Frequency Cepstral Coefficients con l'acronimo MFCC, spectral centroid con SC, infine stft chromagram con STFT. Riportiamo la descrizione delle features considerate nella tabella seguente:

Tipologia	Nome	Dominio
Categorica Nominale	modality vocal_channel emotion statement sex actor channels	audio-only speech, song neutral, calm, happy, fearful, disgust, surprised, angry, sad kids are talking by the door, dogs are sitting by the door M,F da 1 a 24 1(mono), 2 (stereo)
Categorica Ordinale	emotional_intensity repetition sample_width	normal, strong 1st, 2nd 1 (means 8-bit), 2 (means 16-bit)
Numerica	frame_rate frame_width length_ms frame_count intensity zero_crossings_sum Statistics of the original audio signal Statistics of the MFCC Statistics of the SC Statistics of the STFT	frequenza dei campioni registrati (Hz) memoria occupata da ciascun frame (Byte) lunghezza file audio (ms) numero di frame per campione volume del campione registrato (dBFS) somma dello zero-crossing-rate mean, std, min, max, kur, skew mfcc_mean, mfcc_std, mfcc_min, mfcc_max sc_mean, sc_std, sc_min, sc_max, sc_kur, sc_skew stft_mean, stft_std, stft_min, stft_max, stft_kur, stft_skew

Tabella 1.1: Descrizione generale delle features presenti nel dataset

1.2 Distribuzione delle variabili e statistiche

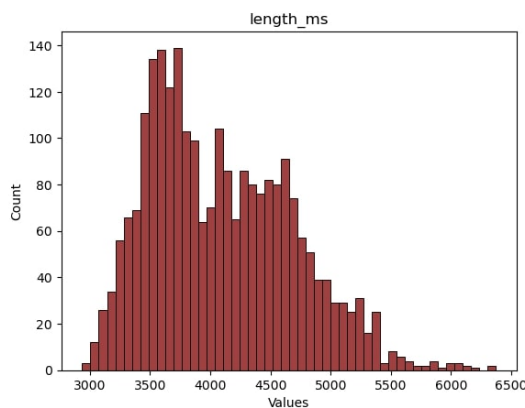
Andremo ad analizzare a livello quantitativo le features del dataset, soffermandoci in particolare sulla distribuzione e le statistiche di alcune di esse.

Iniziamo con una descrizione quantitativa delle features categoriche:

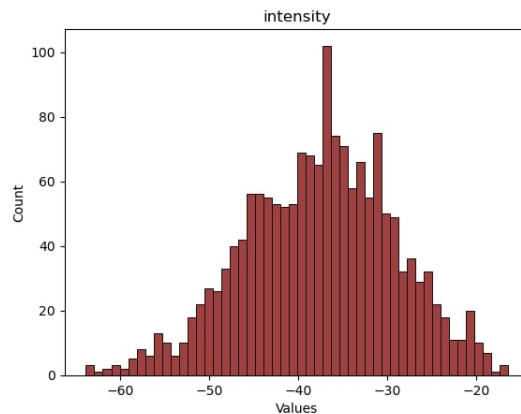
Features	Valori assunti
modality	audio-only
vocal_channel	1335 speech, 921 song
emotion	376 fearful, angry, happy, calm e sad, 192 surprised e disgust, 188 neutral
emotional_intensity	1320 normal, 1332 strong
statement	1226 valori sia per il primo che per per il secondo statement
repetition	1226 sia per la prima che per la seconda ripetizione
actor	ogni attore identificato con un numero da 1 a 24. Gli indici pari vengono utilizzati per donne, indici dispari per uomini. Il numero di record assegnati agli attori non è omogeneo.
sex	1248 audio maschili, 1204 audio femminili
channels	2246 mono, 6 stereo

Tabella 1.2: Distribuzione dei valori assunti da ciascuna feature categorica

Riportiamo nelle figure (a) e (b) sottostanti la distribuzione di due tra le variabili continue che riteniamo tra quelle visivamente più interessanti.



(a) Distribuzione della feature *length_ms*



(b) Distribuzione della feature *intensity*

È possibile osservare che la maggior parte degli audio ha una lunghezza compresa tra 3,4 secondi e 3,7 secondi, e tra 4,1 secondi e 4,6 secondi; questo è sicuramente influenzato dal fatto che un audio può essere stato registrato in modalità *speech* o *song*, ed è anche influenzato dall'emozione che esprime. Per quanto riguarda la distribuzione dell'intensità, notiamo che la maggior parte degli audio assume un valore compreso tra -45 dBFS e -30 dBFS. Riteniamo quindi che l'intensità sia fortemente legata al timbro vocale, al sesso dell'attore e all'emozione da esso espressa.

Abbiamo poi analizzato la matrice di correlazione, ma il risultato non è stato soddisfacente dal momento che quest'ultima conteneva diverse righe e colonne con valore NaN, in quanto diversi attributi presentavano valori mancanti, la cui analisi viene riportata nella sezione successiva.

1.3 Assessing data quality

La qualità del dataset utilizzato incide profondamente con l'accuratezza e l'affidabilità dei risultati ottenuti. In questa sezione andremo quindi ad individuare la presenza di possibili valori mancanti, errori ed outliers presenti all'interno del dataset. Il lavoro svolto viene descritto nelle seguenti sottosezioni: *Errori, valori mancanti ed eliminazione di variabili* ed *Outliers*.

Errori, valori mancanti ed eliminazione di variabili

Analizzando tutte le varie features ci siamo accorte che la variabile *frame_count* contava 35 audio con valore -1, ragione per cui inizialmente nella matrice di correlazione gli attributi *frame_count* e *length_ms* risultavano avere una correlazione di circa 0.76 che è abbastanza alta, ma non tanto quanto ci saremmo aspettate; abbiamo ritenuto che questi valori fossero dei veri e propri errori, visto il significato della feature *frame_count*. Abbiamo dunque deciso di sostituire questi errori con la moda. Per quanto riguarda la presenza di valori mancanti, abbiamo trovato che *vocal_channel* presentava 196 valori nulli, *intensity* 816 ed infine *actor* il più numeroso, con ben 1126 valori mancanti. Abbiamo deciso quindi di operare nel modo seguente:

- *vocal_channel*: abbiamo analizzato la distribuzione della feature *length_ms* separando i dati rispetto a *vocal_channel* stesso. Osservando il grafico, abbiamo scelto come valore soglia per riempire i valori mancanti l'intersezione delle due curve, che avviene all'incirca per un valore pari a 4.2s, come mostrato in figura 1.1: gli audio con durata inferiore a 4.2s sono stati registrati come *speech*, tutti gli altri come *song*. Riteniamo che una tale sostituzione sia piuttosto intuitiva, ma contemporaneamente accurata;

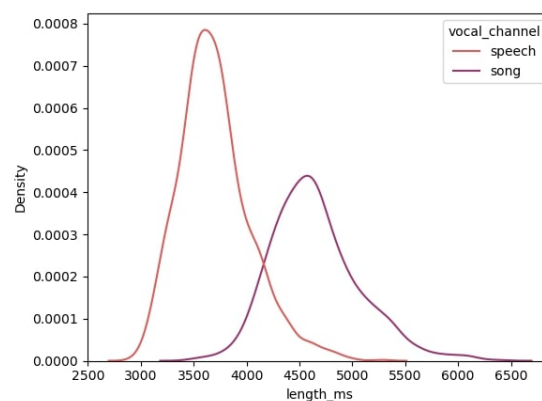


Figura 1.1: Distribuzione della lunghezza in millisecondi rispetto a *vocal_channel*

- *intensity*: abbiamo deciso di sostituire i valori mancanti con la mediana calcolata sui record che presentano la stessa emozione e stesso sesso dell'attore. Ad esempio, se un audio aveva un valore nullo per l'*intensity* e figurava *sex=F* ed *emotion=happy*, allora lo abbiamo sostituito con la mediana dell'*intensity* di tutti gli audio di attrici femminili con emozione happy.

Abbiamo inizialmente deciso di eliminare dal dataframe le seguenti features, perchè non significative dal nostro punto di vista ai fini dell'analisi:

- *modality, stft_max, sample_width, frame_rate* in quanto assumevano lo stesso valore in ciascun record;
- *actor* poichè abbiamo ritenuto che il modo di esprimersi di ciascun attore dovrebbe essere oggettivo e non dovrebbe influenzare la registrazione degli audio; inoltre presentava un numero molto alto di valori nulli;
- *channels, frame_width* perchè i record associati hanno sempre la stessa combinazione: 2446 audio hanno mono e 1, e i restanti 6 stereo e 2.

Abbiamo verificato inoltre che il dataframe non presentasse dei duplicati.

Una volta sistemati i vari errori e valori mancanti abbiamo controllato nuovamente la matrice di correlazione, riuscendo ora a vedere tutte le correlazioni possibili, come riportato in figura 1.2. Abbiamo quindi deciso di eliminare le variabili altamente correlate, ovvero quelle con una correlazione maggiore di ± 0.95 . Dalla matrice di correlazione sono quindi emerse le seguenti coppie di features fortemente correlate: *frame_count* e *length_ms*, *mfcc_min* e *mfcc_std*, *stft_mean* e *stft_skew*, *std* e *min*, *std* e *max*, *min* e *max*. Abbiamo quindi deciso di eliminare ulteriormente le features *frame_count*, *stft_min*, *mfcc_min*, *min*, *max*.

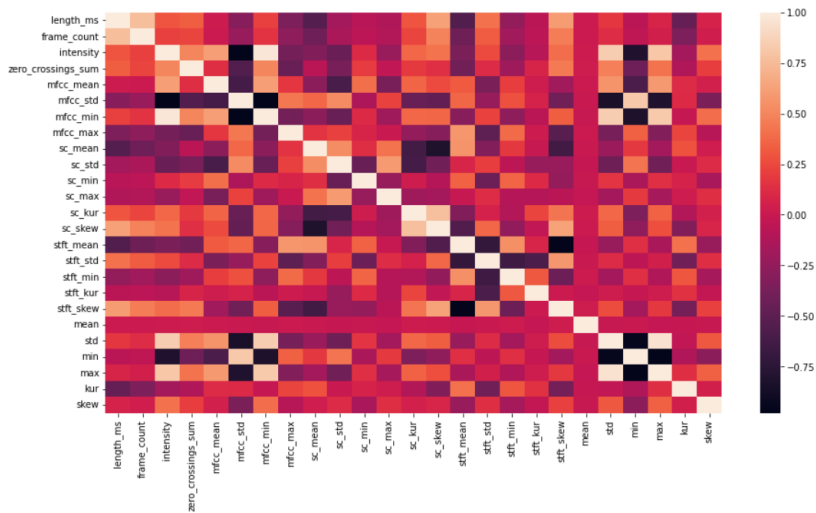


Figura 1.2: Heatmap dopo aver eliminato le variabili non interessanti

Outliers

Per analizzare la presenza di possibili outliers abbiamo tracciato i boxplot delle diverse variabili, e ne abbiamo notato una presenza piuttosto significativa. Dunque, dopo aver calcolato lo scarto interquartile IQR ed aver individuato whisker superiore ed inferiore per ciascuna feature, abbiamo individuato tutti gli outliers presenti, e siamo passate ad analizzarli sia in modo quantitativo (numero totale di outliers per ciascun attributo) che in modo qualitativo (caratteristiche comuni e non comuni per ciascuno di essi). Abbiamo osservato che alcune features riportavano complessivamente un numero relativamente basso di outliers (meno di 35), mentre per altre variabili il numero rilevato è risultato decisamente molto più alto, anche di diverse centinaia.

Riportiamo nella tabella sottostante il numero di outliers per ciascun attributo:

<i>length_ms</i>	<i>intensity</i>	<i>zero_c_s</i>	<i>mfcc_mean</i>	<i>mfcc_std</i>
11	26	52	12	1
<i>mfcc_max</i>	<i>sc_mean</i>	<i>sc_std</i>	<i>sc_min</i>	<i>sc_max</i>
1	5	30	0	439
<i>sc_kur</i>	<i>sc_skew</i>	<i>stft_std</i>	<i>stft_min</i>	<i>stft_kur</i>
152	35	33	336	69
<i>stft_skew</i>	<i>mean</i>	<i>std</i>	<i>kur</i>	<i>skew</i>
14	536	172	85	32

Per quanto riguarda le features che presentano un numero consistente di outlier, abbiamo deciso di non eliminarle, in quanto potrebbero fare riferimento a qualche caratteristica propria dell'audio. Procediamo quindi con un'analisi qualitativa di questi ultimi:

- *zero_crossings_sum*: presenta solo 52 outliers superiori, e quasi tutti sono caratterizzati da *emotion* angry, *emotional_intensity* strong e *sex* F;
- *sc_max*: 189 femmine e 250 maschi, senza una netta distinzione delle *emotion*;
- *sc_kur*: 101 maschi e 51 femmine, per entrambi abbiamo una prevalenza di *emotion* fearful e la quasi assenza di disgust, surprised e neutral;
- *mean* e *std*: diversa distribuzione della variabile *sex*, ma entrambi presentano una prevalenza di *emotion* happy, fearful ed angry;
- *kur*: 25 femmine e 60 maschi, per le femmine abbiamo una prevalenza di *emotion* sad, mentre per i maschi una leggera prevalenza di *emotion* disgust.

Abbiamo perciò deciso di eliminare gli outliers che si presentavano in un numero ragionevole, scegliendo 30 come valore soglia. Per questo motivo abbiamo eliminato gli outliers di *length_ms*, *intensity*, *mfcc_mean*, *mfcc_std*, *mfcc_max*, *sc_mean*, *sc_min*, *stft_skew* e sono risultati quindi 55 records da eliminare, circa il 2.24%. Successivamente, nella sezione riguardante il clustering, abbiamo implementato gli algoritmi sia considerando l'intero dataframe da 2452 audio che quello modificato con 2397 audio, e dal momento che i risultati a cui siamo pervenuti sono simili, abbiamo riportato l'analisi del dataframe privo di outliers.

Capitolo 2

Clustering

All'interno di questo capitolo andremo ad utilizzare diverse tecniche di clustering per analizzare più a fondo i nostri dati, con l'obiettivo di trovare gruppi di oggetti con caratteristiche simili. L'analisi è stata svolta sfruttando quattro algoritmi di clustering: K-means, DBSCAN, OPTICS e Hierarchical Clustering.

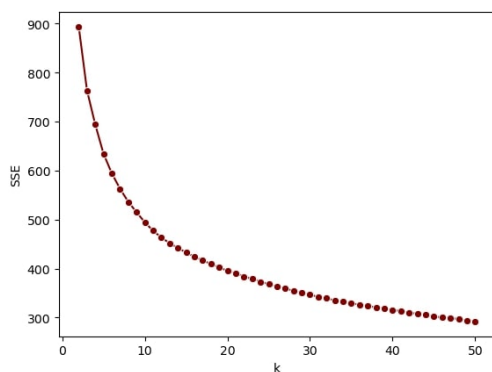
Visto che questi algoritmi non operano sugli attributi categorici, abbiamo svolto la nostra analisi eliminando dal dataset le prime sei colonne corrispondenti alle variabili categoriche (*vocal_channel*, *emotional_intensity*, *sex*, *repetition*, *statement*). Per l'implementazione degli algoritmi abbiamo normalizzato i nostri dati utilizzando sia *MinMaxScaler* che *StandardScaler*, ed abbiamo infine deciso di utilizzare la prima normalizzazione poichè risulta più omogenea, in quanto i valori di ciascun attributo risultano compresi in un range da zero ad uno.

2.1 k-means

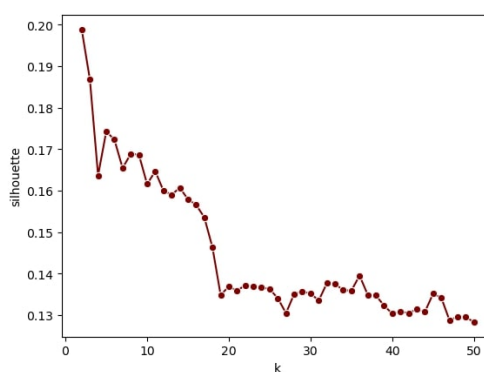
In questa sezione approfondiremo l'uso dell'algoritmo k-means e i risultati dell'analisi.

2.1.1 Scelta del miglior k

Procediamo con la determinazione del miglior valore di k, ovvero il numero di cluster in cui vogliamo suddividere il dataset. Per decidere il numero di cluster è stato calcolato l'*SSE* e la *Silhouette Score*. Ciascun grafico è stato ottenuto facendo variare il numero di possibili cluster k da 2 a 50, perché riteniamo che non sia significativo considerare un numero di cluster maggiore di 50.



(a) Curva dell'*SSE* in funzione del numero di cluster *k*



(b) Curva dell'indice di *Silhouette* in funzione del numero di cluster *k*

Possiamo osservare che la curva dell' *SSE*, mostrata in figura (a), risulta monotona decrescente e tende a raggiungere il valore 300, inoltre è priva di un vero e proprio punto di gomito; il grafico dell'indice di *Silhouette*, mostrato in figura (b), tende ad assestarsi intorno al valore 0.13 in modo non monotono e presenta dei picchi isolati. Dal momento che il nostro obiettivo è quello di minimizzare l'indice *SSE* e contemporaneamente massimizzare la *Silhouette*, riteniamo che il miglior compromesso sia un numero di cluster non superiore a 14, in quanto per $k > 14$ l'indice di *Silhouette* tende solo a decrescere e ad assestarsi ad un valore soglia vicino a 0.13, mentre

l'*SSE*, superato il valore $k = 14$, non presenta un miglioramento così significativo. Osservando poi l'andamento della *Silhouette* riteniamo che i valori più interessanti per k siano quelli in corrispondenza dei picchi, ovvero $k = 5, 8, 11, 14$. Abbiamo quindi deciso di riportare il caso $k=5$ in quanto crediamo che sia il più efficace da visualizzare e descrivere, mentre a nostro avviso il miglior compromesso tra miglior *SSE* ed indice di *Silhouette* si ha per $k = 11$.

k	<i>SSE</i>	<i>Silhouette Score</i>
5	634.394	0.174
8	544.264	0.168
11	477.118	0.164
14	441.887	0.159

2.1.2 Descrizione e sintesi dei risultati

Riportiamo nella Tabella 2.1 seguente il numero di dati presenti e la composizione delle emozioni in ogni cluster.

ID	#Elementi	Angry	Calm	Disgust	Fearful	Happy	Neutral	Sad	Surprised
A	489	40	94	63	37	48	51	78	78
B	337	159	11	7	86	54	5	12	3
C	646	77	87	45	121	123	46	113	34
D	487	70	79	18	85	97	32	96	10
E	448	27	84	58	37	54	54	68	66

Tabella 2.1: Composizione dei cluster ottenuti tramite kmeans

Andiamo di seguito ad analizzare i diversi cluster, in particolare rispetto alle diverse emozioni presenti e anche alla distribuzione tra maschi e femmine, che riteniamo essere interessante.

Il Cluster A presenta prevalentemente le emozioni calm, disgust, sad e surprised. La maggioranza dei punti appartiene ad attori di sesso femminile (326).

Il Cluster B ha una prevalenza di dati con emozione angry, fearful ed happy, mentre una scarsissima presenza delle altre emozioni (sono circa ciascuna presente solo per il 2, 3, 4%). Per quanto riguarda il sesso invece non è possibile osservare una distinzione netta, e il numero di dati all'incirca si equivale (146 femmine e 191 maschi)

Il Cluster C è il più numeroso, e le emozioni si dispongono in modo piuttosto omogeneo. Al contrario, la divisione del sesso è assai evidente: gli audio con attori di sesso maschile (439) sono più del doppio di quelli femminili (207).

Il Cluster D vede in prevalenza audio con emozioni fearful, happy e sad, mentre vi è una bassissima affluenza di audio con emozioni disgust e surprised (9% e 5%). La divisione tra audio maschili (68) e femminili (419) è di nuovo molto evidente.

Il Cluster E presenta prevalentemente emozioni come calm, disgust, neutral e surprised, con una bassa presenza di emozioni angry (circa il 7%). È molto chiara la divisione tra gli audio femminili (89) e maschili (359).

Riportiamo nella figura 2.1 sottostante, a scopo esemplificativo, lo scatter plot riguardante le features *mfcc_max* e *stft_std*.

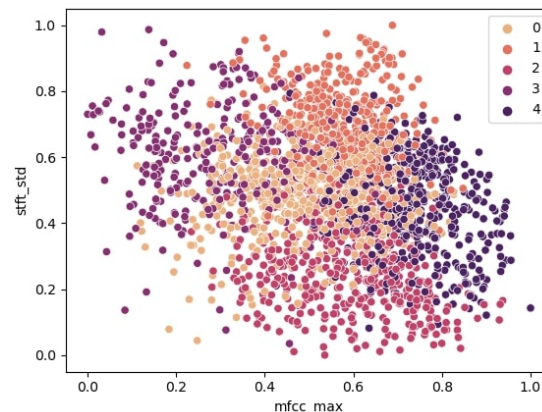


Figura 2.1: Scatter plot sulle variabili *mfcc_max* e *stft_std*

Riteniamo che i risultati del k-means possano essere stati negativamente influenzati dalla forte presenza degli outliers ancora presenti, dal momento che il calcolo dei centroidi è molto sensibile a questi ultimi.

2.1.3 k-means limitato ad alcune features

Dal momento che non è stato possibile raggiungere risultati pienamente soddisfacenti dall'applicazione dell'algoritmo k-means applicato a tutte le features, abbiamo deciso di eseguire l'algoritmo anche su alcuni gruppi di caratteristiche selezionate. Abbiamo quindi copiato in nuovi dataframe più piccoli i dati, selezionando gruppi di features scarsamente correlati, nel tentativo di portare alla luce pattern meno evidenti (gruppi 1,2,3), e features relative a statistiche della stessa tipologia (gruppi 4,5,6,7):

1. *length_ms*, *zero_crossings_sum*, *mean*, *std*, *skew*;
2. *length_ms*, *intensity*, *sc_min*, *sc_max*, *kur*;
3. *zero_crossings_sum*, *mfcc_mean*, *sc_mean*, *stft_kur*, *mean*;
4. *mfcc_mean*, *mfcc_std*, *mfcc_max*;
5. *sc_mean*, *sc_std*, *sc_min*, *sc_max*, *sc_kur*, *sc_skew*;
6. *stft_std*, *stft_min*, *stft_kur*, *stft_skew*;
7. *mean*, *std*, *kur*, *skew*.

I risultati ottenuti sono stati sufficientemente soddisfacenti per alcuni dei gruppi di features, per quanto riguarda i valori di *SSE* e *Silhouette*, ma nessuno di questi si presenta con una divisione precisa dei dati rispetto alle variabili a cui siamo interessati.

Riportiamo i risultati ottenuti dal primo gruppo, che riteniamo essere il più informativo. Tramite l'analisi dell'*SSE* e dell'indice di *Silhouette* abbiamo concluso che il numero ideale di cluster fosse tre. I tre cluster individuati evidenziano alcune caratteristiche, visibili in figura 2.2: uno dei tre è piuttosto omogeneo rispetto alle emozioni, mentre negli altri due si possono notare delle differenze. In uno di questi è praticamente assente l'emozione *surprised*, con invece una forte presenza delle emozioni *calm*, *sad* e *happy*; nell'ultimo la quasi totalità appartiene all'emozione *angry*, con una presenza considerevole di *happy* e *fearful*, mentre tutte le altre emozioni sono molto poco presenti.

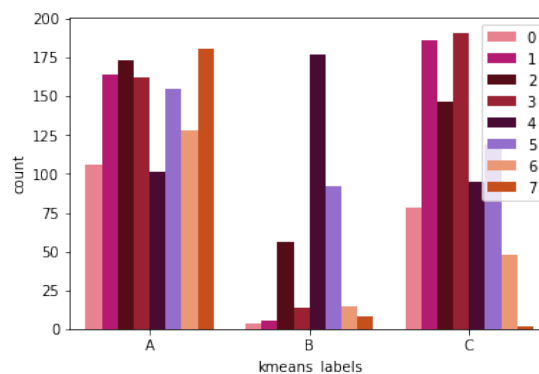


Figura 2.2: Distribuzione delle emozioni nei cluster con solo alcune features

Altri gruppi di features hanno portato a valori simili di *SSE* e *Silhouette*, che però sono stati scartati o per la presenza di cluster molto più numerosi degli altri o perché era stato scelto di analizzare un numero minore di features, che si concludeva in distanze minori.

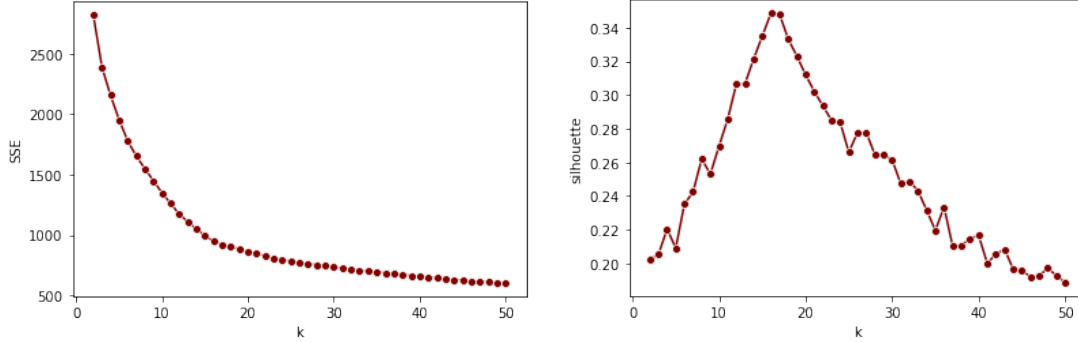
2.1.4 k-means con trasformazione delle features categoriche

Per effettuare un'analisi più completa abbiamo deciso di applicare una trasformazione binaria alle features categoriche, le quali in precedenza erano state eliminate perché non utilizzabili dall'algoritmo di clustering.

Per prima cosa abbiamo visualizzato tutte le features diverse dal tipo numerico, risultando quindi *vocal_channel*, *emotion*, *emotional_intensity*, *sex*, *repetition*, *statement*. Abbiamo quindi effettuato una trasformazione binaria in quanto per ogni feature i possibili valori assunti erano sempre due; unica eccezione è stata la feature *emotion* per la quale non è stato possibile mapparla in

binario senza poter aumentare la dimensionalità del dataframe, dunque la colonna del dataframe riguardante la variabile *emotion* è stata eliminata.

A partire da queste trasformazioni abbiamo quindi creato un nuovo dataframe e successivamente abbiamo applicato tutte le operazioni di clustering come in precedenza, cercando poi di visualizzare quelli che fossero i miglioramenti utilizzando la feature *emotion* per la distinzione grafica dei cluster. Riportiamo in figura (a) e (b) l'andamento di *SSE* e *Silhouette*:



(a) Valori *SSE* per clustering con trasformazione features categoriche

(b) Valori *Silhouette* per clustering con trasformazione features categoriche

I grafici riportati mostrano un andamento molto favorevole: fino a $k=16$ *SSE* e *Silhouette* sono inversamente proporzionali, superato questo valore entrambi tendono a decrescere. Analizzando i dati con $k=16$, valore che massimizza la *Silhouette* e che presenta un gomito nel grafico dell'*SSE*, abbiamo notato che la distribuzione dei record nei diversi cluster era omogenea ed i risultati per i valori di *SSE* e *Silhouette*, rispettivamente 901.946 e 0.335, fossero migliori e più interessanti rispetto alle analisi precedenti.

2.2 Density Based Clustering

Il secondo tipo di algoritmo che abbiamo applicato al nostro dataset è il *DBSCAN*.

2.2.1 Scelta dei parametri DBSCAN

Per poter utilizzare questo algoritmo è necessario scegliere due parametri: *epsilon* e *MinPts*. Per scegliere i parametri input del DBSCAN abbiamo calcolato e rappresentato attraverso dei grafici le distanze di ciascun punto del dataset dal proprio k^{th} vicino più vicino, e per ciascun k^{th} abbiamo scelto l' ϵ migliore guardando il punto di gomito del grafico. Fissate le coppie (k^{th}, ϵ) abbiamo calcolato per ciascuna di esse il relativo indice di *Silhouette* per poter scegliere il caso migliore. La scelta finale è stata per $k^{th} = 25$ ed $\epsilon = 0.58$. La funzione distanza dal 25-esimo vicino è riportata nel grafico 2.3.

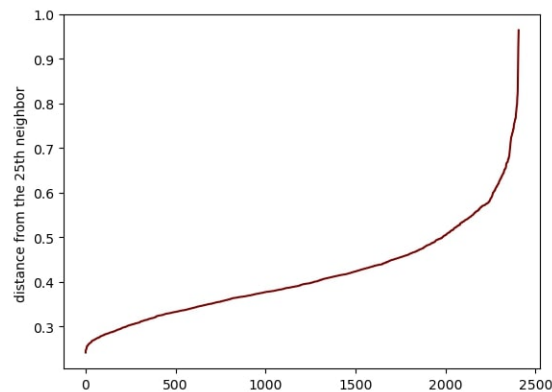


Figura 2.3: Distanza di ciascun punto dal k-esimo vicino più vicino, per $k=25$

2.2.2 Descrizione e sintesi dei risultati DBSCAN

Osservando il risultato ottenuto dall'applicazione del DBSCAN sui nostri dati possiamo concludere che esso non è molto soddisfacente, infatti il numero di cluster individuati dal DBSCAN è pari a due. Nel primo sono presenti 18 dati corrispondenti ai noise points del dataset, mentre tutti gli altri punti vengono raggruppati all'interno dell'altro cluster, dunque questo non ci permette di ottenere alcun tipo di informazione. In conclusione per questo algoritmo quasi tutti i dati sono molto simili.

Considerando che, a livello puramente teorico, le performance del DBSCAN sono fortemente influenzate da una distribuzione densa dei dati, crediamo che proprio questo possa essere uno dei motivi che porta ad essere insoddisfacenti i nostri risultati. Per appurare ciò, abbiamo visualizzato alcuni scatterplot in due dimensioni ed effettivamente i dati da noi utilizzati risultano essere disposti in maniera assai densa.

2.2.3 OPTICS

Per cercare di migliorare i risultati ottenuti attraverso il DBSCAN, abbiamo provato ad eseguire le operazioni di clustering anche utilizzando l'algoritmo OPTICS. Anche in questo caso, abbiamo provveduto all'eliminazione delle variabili categoriche *vocal_channel*, *emotion*, *emotional_intensity*, *statement*, *repetition*, *sex* e successivamente abbiamo applicato le operazioni di scaling e fit come effettuato in precedenza.

L'algoritmo OPTICS richiede due parametri: *Min_Pts* ed ϵ . Abbiamo fissato il valore di ϵ pari al valore di default *np.inf*, mentre per individuare il numero ottimale di *Min_Pts* abbiamo calcolato l'indice di *Silhouette* per valori compresi tra 3 e 11. I risultati ottenuti non sono stati soddisfacenti, in quanto in ognuno di questi casi alla quasi totalità dei dati viene assegnata la label -1, che raggruppa i noise points.

2.3 Hierarchical clustering

L'ultimo algoritmo utilizzato è quello del clustering gerarchico. Abbiamo implementato l'algoritmo utilizzando varie distanze e metodi di linkage, in particolare le distanze considerate sono state euclidea, precomputed e manhattan, mentre i metodi di linkage sono stati min-link, max-link, ward ed average. Dalle nostre analisi, confrontando i coefficienti di *Silhouette* dei vari casi e la distribuzione dei dati all'interno di ciascun cluster, risulta che il metodo migliore sia quello che considera la distanza euclidea ed il metodo ward: emergono tre cluster, composti rispettivamente da 998, 917 e 492 dati, con un coefficiente di *Silhouette* all'incirca di 0.175.

La figura 2.4 sottostante rappresenta il dendrogramma ottenuto dall'algoritmo.

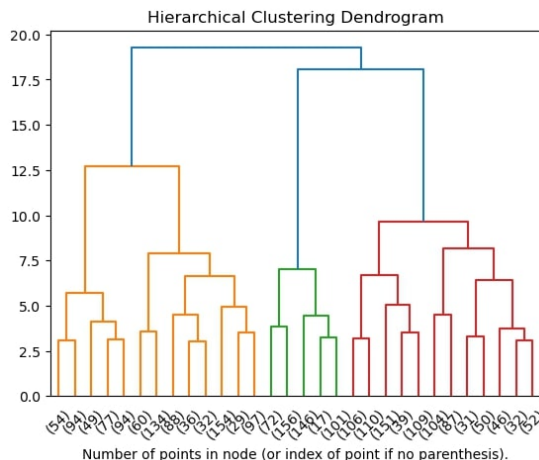


Figura 2.4: Dendrogramma ottenuto dall'implementazione dell'algoritmo gerarchico

Attraverso le crosstab sottostanti è possibile vedere la distribuzione delle emozioni e del sesso all'interno dei clusters:

cluster	A	B	C
angry	132	215	26
calm	166	105	84
disgust	91	38	62
fearful	131	187	48
happy	145	163	68
neutral	84	54	50
sad	158	125	84
surprised	91	30	70

cluster	A	B	C
F	733	319	135
M	265	598	357

Per quanto riguarda la distribuzione delle emozioni all'interno dei clusters: calm, disgust, neutral, sad e surprised si dispongono quasi per metà nel primo cluster, mentre angry, fearful e happy prevalentemente nel secondo cluster. È possibile inoltre osservare che solo il 7% dei dati che prevedono l'emozione angry si dispongono nel cluster C; un simile comportamento si riscontra negli audio caratterizzati dall'emozione fearful, solo il 13% si dispone in C. Infine osserviamo che una percentuale piuttosto bassa (16%) di audio caratterizzati dall'emozione surprised si dispone in B.

Per quanto riguarda la caratterizzazione degli audio in base al sesso, osserviamo che all'interno del primo cluster A vi è una grande prevalenza di record femminili, circa il 73%, contro un 27% di audio maschili; all'interno del secondo e terzo cluster troviamo una situazione ribaltata con una prevalenza di audio maschili: in B abbiamo circa il 65% di record maschili ed il 35% di audio femminili, infine in C abbiamo il 73% di audio maschili e il 27% femminili.

Per visualizzare quanto ottenuto dall'algorithm riportiamo nella figura 2.5 sottostante a titolo esemplificativo lo scatter plot relativo agli attributi *zero_crossings_sum* e *mfcc_mean*.

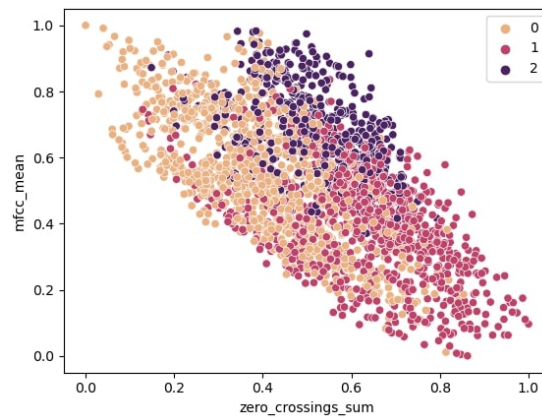


Figura 2.5: Scatter plot relativo agli attributi *zero_crossings_sum* e *mfcc_mean*

Capitolo 3

Classification

In questo capitolo andremo ad eseguire la classificazione del dataset per le variabili *emotion* e *sex*, le stesse che sono state analizzate nella parte di clustering (capitolo 2). I classificatori utilizzati sono rispettivamente Decision Trees, KNN e Naive Bayes, la cui analisi specifica è riportata nella sezioni successive.

Abbiamo quindi proceduto con la suddivisione del dataset in training set e test set con un holdout iniziale di 70% e 30% con stratificazione della variabile target, in modo tale da avere una distribuzione omogenea rispetto alla variabile target scelta.

3.1 Decision Trees

In questa sezione utilizzeremo gli alberi decisionali per classificare le variabili *emotion* e *sex*. Per la scelta dei parametri ottimali abbiamo utilizzato le metodologie grid e randomized search, le combinazioni per i parametri sono state calcolate utilizzando i seguenti insiemi, rispettivamente:

```
{min_samples_split=[0.002, 0.01, 0.05, 0.1, 0.2], min_samples_leaf= [0.001, 0.01, 0.05, 0.1, 0.2],  
max_depth=[2, 4, 6, 8, 10, 12, 16, None]};  
{min_samples_split=np.linspace(0.002,0.2,7), min_samples_leaf=np.linspace(0.001,0.2,7),  
max_depth=[2, 4, 6, 8, 10, 12, 16, None], criterion=[gini, entropy]};  
{min_samples_split=loguniform(1e-2, 1e0), min_samples_leaf=uniform(0.001, 0.2),  
max_depth=randint(2, 200), criterion=[gini, entropy]}
```

In particolare, per quanto riguarda la grid search, per ciascuna possibile combinazione è stata implementata una repeated stratified k-fold con n_split=10 e n_repeats=10, mentre per la randomized search è stata utilizzato lo stesso procedimento, scegliendo inoltre come parametro n_iter=200. Per ciascuna ricerca abbiamo salvato il classificatore migliore, e comparando gli score di questi ultimi sul test set fissato precedentemente abbiamo visto quale fosse in grado di generalizzare al meglio sul test set. Infine, scelto il classificatore migliore, abbiamo deciso di fare una RepeatedStratifiedKFold per valutarne ulteriormente la performance.

3.1.1 Variabile target *emotion*

Il decision tree migliore è risultato dalla grid search, con parametri dati da max_depth=10, min_samples_leaf=0.001, min_samples_split=0.002, criterion=gini, mostrato in figura 3.1

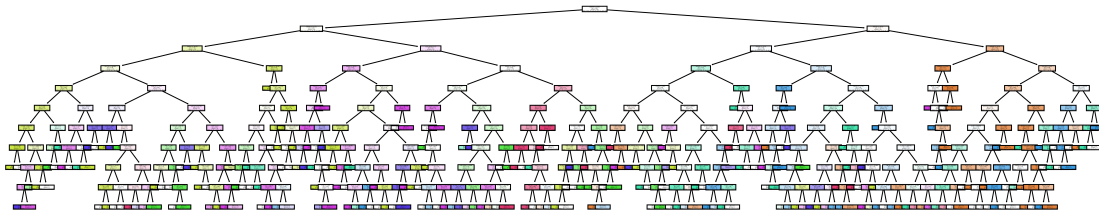


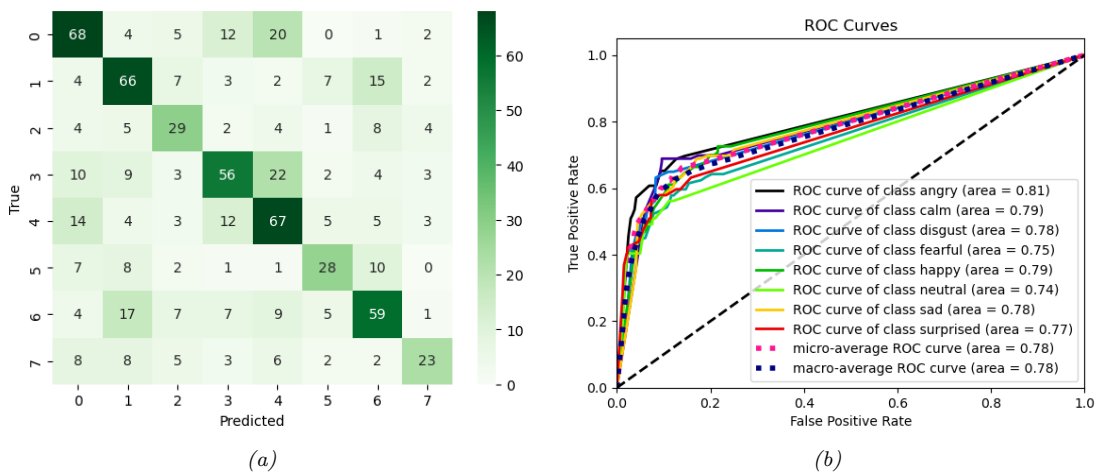
Figura 3.1: Rappresentazione grafica del decision tree ottimale

Riportiamo qui di seguito il classification report che mostra i parametri *accuracy*, *precision*, *recall*, *f-1 score*:

	precision	recall	f1-score	support
angry	0.57	0.61	0.59	112
calm	0.55	0.62	0.58	106
disgust	0.48	0.51	0.49	57
fearful	0.58	0.51	0.55	109
happy	0.51	0.59	0.55	113
neutral	0.56	0.49	0.52	57
sad	0.57	0.54	0.55	109
surprised	0.61	0.40	0.48	57
accuracy			0.55	720
macro avg	0.55	0.54	0.54	720
weighted avg	0.55	0.55	0.55	720

Come mostra la tabella sopra, il classificatore scelto possiede un'accuratezza del 55% circa, l'emozione con cui si dimostra più preciso è *surprised*, mentre la peggiore è data da *disgust* (circa 48%). Entrambe le medie pesate e non pesate delle precisioni delle singole emozioni sono pari al 55%. Per quanto riguarda la recall, l'emozione che assume il valore più alto è dato dall'emozione *calm* mentre rispetto all'emozione *surprised* risulta solo del 40%. Le medie complessive risultano del 54% (non pesata) e 55% (pesata). Infine, come potevamo aspettarci visti gli scarsi valori di precision e recall per le emozioni *disgust* e *surprised*, l'indice f1-score è il più basso per queste due emozioni, rispettivamente 49% e 48%.

Riportiamo, inoltre, nelle figure (a) e (b) la confusion matrix e ROC curve:



Possiamo osservare che l'area sottesa a ciascuna delle curve ROC è simile, infatti il valore più basso è quello dell'emozione *neutral* (0.74), mentre il più alto per *angry* con 0.81.

Infine, abbiamo implementato una `RepeatedStratifiedKFold` con parametri `n_splits=10` e `n_repeats=10` per avere una stima maggiormente precisa per valutare la performance del classificatore scelto, ottenendo un'accuratezza dello 0.5168 ± 0.04 .

La classificazione portata avanti per la feature *sex* porta, invece, a dei risultati molto più significativi. Anche in questo caso i parametri ottimali del decision tree, riportati in figura 3.2, sono derivati dalla grid search e sono dati da `max_depth=8`, `min_samples_leaf=0.001`, `min_samples_split=0.002`, `criterion=gini`.

3.1.2 Variabile target *sex*

Riportiamo il classification report per questo classificatore:

	precision	recall	f1-score	support
F	0.89	0.95	0.92	355
M	0.94	0.88	0.91	365
accuracy			0.91	720
macro avg	0.92	0.91	0.91	720
weighted avg	0.92	0.91	0.91	720

Possiamo osservare che tendenzialmente il decision tree scelto si comporta in modo pressochè equivalente per la classificazione di entrambi i generi, risultando complessivamente poco più

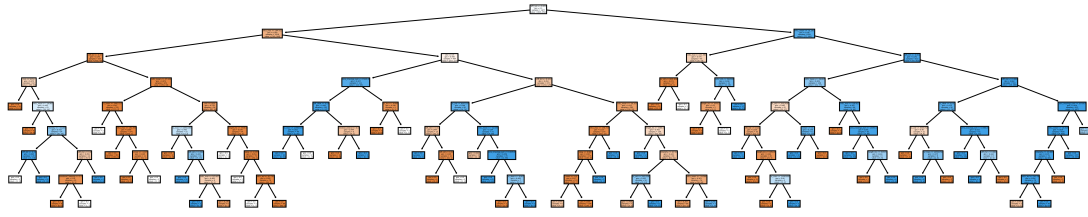
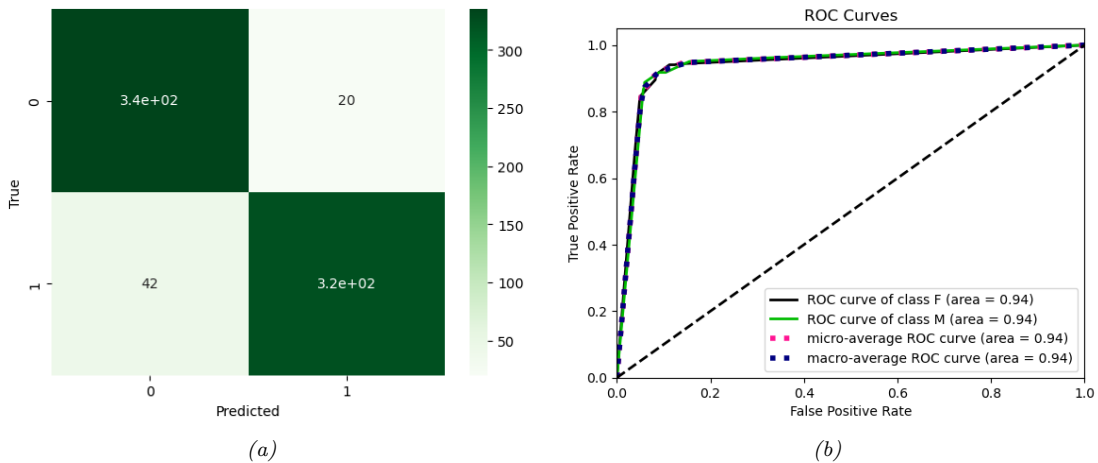


Figura 3.2: Rappresentazione grafica del decision tree ottimale per la classificazione della variabile sex

efficace per la classificazione del genere femminile. Riportiamo, inoltre, la confusion matrix e la ROC curve nelle figure (a) e (b) sottostanti.



3.2 K-NN

In questa seconda sezione utilizzeremo l'algoritmo del K-NN (Nearest Neighbor Classifier) per analizzare le variabili target. In entrambi i casi, è necessario scegliere il valore di k , cioè il numero di vicini da considerare, le metriche con cui calcolare le distanze e i pesi da applicare per la scelta della classe da assegnare al record in esame.

Sia considerando come variabile target le emozioni che il sesso, sono stati trovati i migliori parametri effettuando una Grid Search, confrontando cioè tutte le possibili combinazioni tra i seguenti parametri:

Parametro	Possibili valori
k	Numeri dispari tra 1 e 101
Metrica	Euclidea, Manhattan
Peso	Uniforme, dipendente dalla distanza

3.2.1 Variabile target *emotion*

Come è possibile vedere dalla figura 3.3, i parametri che ottengono la migliore performance sono: distanza con metrica Manhattan e pesi proporzionali alla distanza, con un numero k di vicini considerati uguale a 53. Questo numero di vicini è all'incirca quanto normalmente consigliato, cioè la radice quadrata del numero di istanze nel training set. L'accuratezza dell'algoritmo con i migliori parametri è del 43.47%.

Riportiamo il classification report relativo al migliore classificatore:

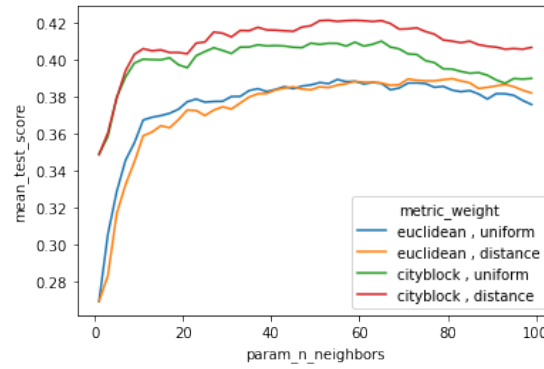
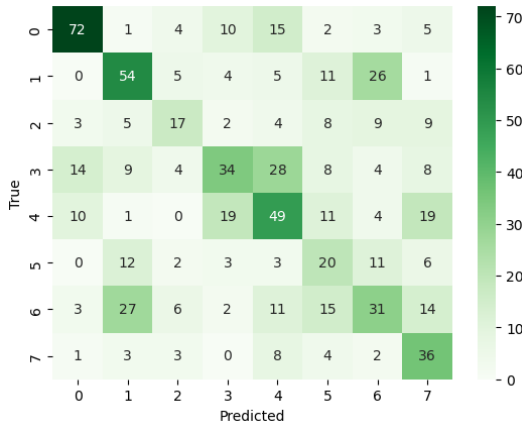


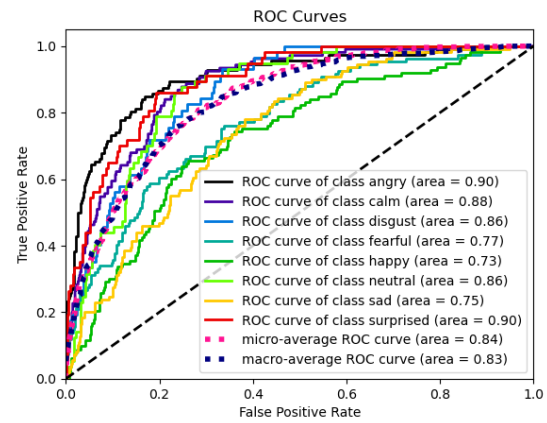
Figura 3.3: Performance delle diverse combinazioni di metriche e pesi con variabile target *Emotion*

	precision	recall	f1-score	support
angry	0.70	0.64	0.67	112
calm	0.48	0.51	0.50	106
disgust	0.41	0.30	0.35	57
fearful	0.46	0.31	0.37	109
happy	0.40	0.43	0.42	113
neutral	0.25	0.35	0.29	57
sad	0.34	0.28	0.31	109
surprised	0.37	0.63	0.46	57
accuracy			0.43	720
macro avg	0.43	0.43	0.42	720
weighted avg	0.45	0.43	0.43	720

Come mostra la tabella sopra, è possibile osservare che il K-NN non è molto efficace per classificare le emozioni, ad eccezione dell'emozione *angry* e *calm* che risultano avere i parametri migliori. In particolare risulta molto poco efficace nel classificare l'emozione *neutral*. Riportiamo poi nelle figure (a) e (b) seguenti la confusion matrix e la relativa ROC curve:



(a)



(b)

3.2.2 Variabile target *sex*

Tracciando di nuovo lo stesso grafico presentato nel caso precedente, ed osservabile in figura 3.4, troviamo che la migliore combinazione di parametri è uguale alla precedente, cambiando solo il numero k di vicini considerati, che è in questo caso 9. La quantità di dati da considerare per classificare i record è molto minore, e come è possibile notare dal grafico, la performance peggiora molto velocemente con l'aumentare di k : il considerare troppi dati potrebbe portare all'overfitting del modello, peggiorandone le prestazioni. Nel caso ottimale l'accuratezza dell'algorithmo è del 93%, è quindi molto efficiente nel classificare il sesso dell'attore.

Riportiamo nella seguente tabella le metriche relative al migliore K-NN:

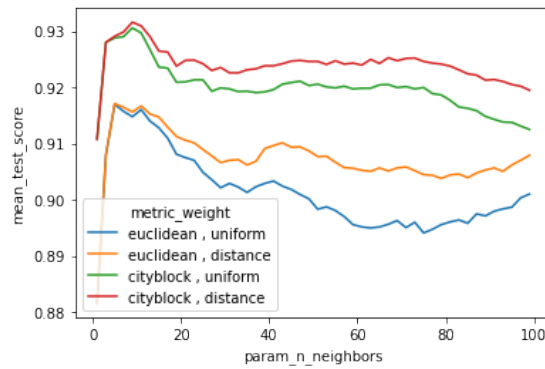
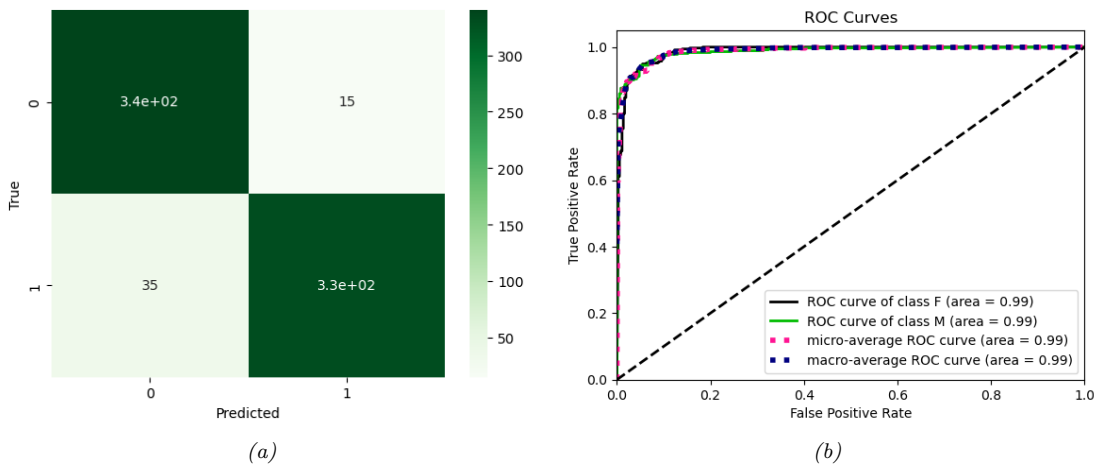


Figura 3.4: Performance delle diverse combinazioni di metriche e pesi con variabile target *Sex*

	precision	recall	f1-score	support
F	0.91	0.96	0.93	355
M	0.96	0.90	0.93	365
accuracy			0.93	720
macro avg	0.93	0.93	0.93	720
weighted avg	0.93	0.93	0.93	720

Riportiamo, inoltre, la confusion matrix e la ROC curve:



Come per il decision tree, anche in questo caso la classificazione della variabile *sex* risulta avere delle performance molto più alte e le metriche risultano pressochè equivalenti per entrambi i sessi.

3.3 Naive Bayes

In quest'ultima sezione andiamo ad analizzare le nostre variabili target utilizzando il classificatore Naive Bayes. Come prima cosa, abbiamo utilizzato l'algoritmo di classificazione *GaussianNB()*, il quale si basa sull'applicazione del teorema di Bayes, assumendo quindi una forte indipendenza tra i dati che gli vengono forniti. Successivamente, abbiamo applicato l'algoritmo di classificazione *CategoricalNB()*, il quale è particolarmente adatto per tutti quei casi in cui sono presenti variabili discrete distribuite in modo categorico.

Vediamo ora i risultati ottenuti in entrambi i casi di classificazione per le nostre variabili target *emotion* e *sex*, utilizzando gli algoritmi appena presentati.

3.3.1 Variabile target *emotion*

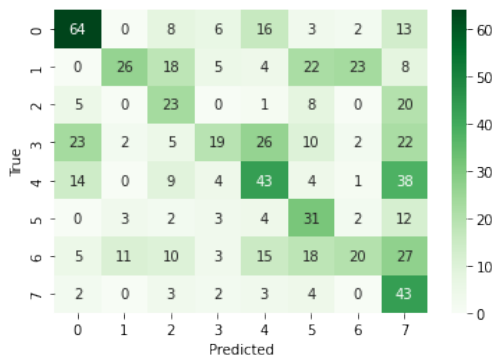
Come prima operazione, abbiamo eseguito la classificazione usando il classificatore *GaussianNB()*: riportiamo a seguito il classification report, mostrando la qualità dei risultati ottenuti tramite i seguenti parametri *accuracy*, *precision*, *recall*, *f-1 score*:

	precision	recall	f1-score	support
angry	0.57	0.57	0.57	112
calm	0.62	0.25	0.35	106
disgust	0.29	0.40	0.34	57
fearful	0.45	0.17	0.25	109
happy	0.38	0.38	0.38	113
neutral	0.31	0.54	0.39	57
sad	0.40	0.18	0.25	109
surprised	0.23	0.75	0.36	57
accuracy			0.37	720
macro avg	0.41	0.41	0.36	720
weighted avg	0.44	0.37	0.36	720

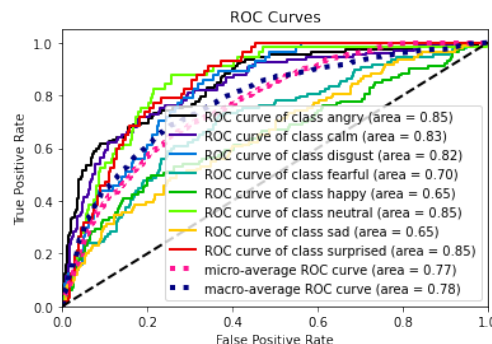
In linea generale, è possibile osservare che l'algoritmo non è molto efficiente nella classificazione delle emozioni in quanto la media degli score dei risultati è molto bassa; così come in precedenza, le emozioni *angry* e *calm* risultano essere le quelle con parametri migliori. I risultati peggiori sono stati ottenuti invece con le emozioni *surprised*, *disgust* e *neutral*

Per cercare di ottenere risultati migliori, abbiamo provato le operazioni di classificazione utilizzando il classificatore *CategoricalNB* ma i risultati sono stati simili ai precedenti: le emozioni classificate come migliori oppure come peggiori sono sempre le stesse, ma in questo caso si abbassano ulteriormente gli score per ciascuna di esse così come l'accuratezza generale e le prestazioni dell'algoritmo.

Per una maggiore completezza di analisi, decidiamo di riportare nelle figure (a) e (b) di seguito rispettivamente la Confusion Matrix e la ROC Curve in riferimento al miglior classificatore tra i due, ovvero il *GaussianNB*. A conferma dei nostri risultati, nella figura (b) possiamo osservare che l'area sottesa a ciascuna delle ROC Curve è diversa a seconda dell'emozione che viene presa in considerazione: le emozioni che corrispondono a ROC Curve superiori portano risultati pressoché accettabili, mentre osservando le emozioni corrispondenti alle ROC Curve inferiori, i risultati sono molto scarsi ed inefficienti.



(a)



(b)

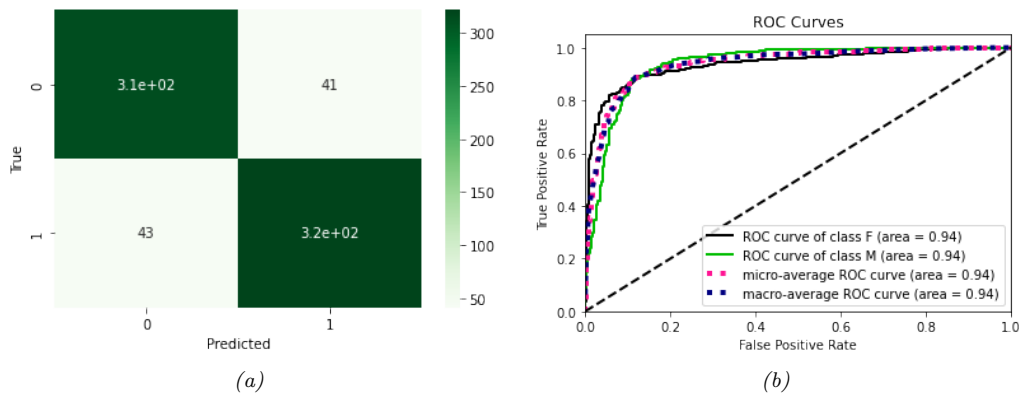
3.3.2 Variabile target *sex*

Anche per quanto riguarda la variabile target *sex*, abbiamo eseguito le operazioni di classificazione usando entrambi i classificatori *GaussianNB* e *CategoricalNB*. Dall'analisi dei classification report, possiamo dire che i risultati ottenuti dai classificatori sono pressoché identici, con un leggero miglioramento in termini di prestazioni ed accuratezza utilizzando il classificatore *GaussianNB*. Per questo motivo, e per una maggiore chiarezza dei risultati, riportiamo il relativo classification report:

	precision	recall	f1-score	support
F	0.88	0.88	0.88	355
M	0.89	0.88	0.88	365
accuracy			0.88	720
macro avg	0.88	0.88	0.88	720
weighted avg	0.88	0.88	0.88	720

Il classificatore si comporta complessivamente nello stesso modo entrambi i sessi, anche se notiamo un leggero miglioramento per quanto riguarda gli score del genere femminile.

A conferma dei risultati ottenuti, riportiamo nelle figure seguenti (a) e (b) seguenti rispettivamente la Confusion Matrix e la ROC Curve della variabile target *sex*:



3.4 Confronto finale

Dall'analisi fatta, possiamo riassumere le migliori performance ottenute dai tre classificatori sulle variabili *emotion* e *sex* per capire quale possa essere il classificatore che generalizza nel modo più efficace.

	accuracy	macro_avg (p)	macro_avg (r)	macro_avg (f1)	support
decision tree	0.55	0.55	0.54	0.54	720
K-NN	0.43	0.43	0.43	0.42	720
gaussian-NB	0.37	0.41	0.41	0.36	720

Dalla tabella riportata sopra possiamo osservare che, per quanto riguarda la variabile *emotion*, in linea di massima il decision tree risulta essere il classificatore migliore tra i tre, mentre il peggiore è il Gaussian-NB. Per quanto riguarda le singole emozioni, osservando le varie performance dei tre classificatori, possiamo affermare che il K-NN permette di classificare al meglio l'emozione *angry*.

Per quanto riguarda la variabile *sex*:

	accuracy	macro_avg (p)	macro_avg (r)	macro_avg (f1)	support
decision tree	0.91	0.92	0.91	0.91	720
K-NN	0.93	0.93	0.93	0.93	720
gaussian-NB	0.88	0.88	0.88	0.88	720

possiamo osservare che tutti i classificatori sono pressochè equivalenti, probabilmente a nostro avviso il migliore per questa variabile risulta essere il K-NN.

Capitolo 4

Pattern Mining

In questo capitolo cercheremo di fare emergere i vari pattern che caratterizzano il nostro dataset, soffermandoci in particolare su alcune association rules che riteniamo essere maggiormente interessanti. Nel nostro studio ci focalizzeremo, in particolare, su pattern riguardanti le variabili *emotion* e *sex*. Per il nostro studio abbiamo utilizzato l'algoritmo Apriori.

4.1 Estrazione di Frequent Pattern

Abbiamo iniziato la nostra analisi discretizzando tutte le variabili continue tramite la funzione `pandas.qcut`. Successivamente, abbiamo visto come cambia il numero di itemset frequenti in funzione del supporto, fissando il parametro `zmin=2`; riportiamo in figura 4.1 il grafico appena descritto con supporto variabile nell'intervallo `[2%,30%]`. Per facilitare la visualizzazione nel grafico, abbiamo effettuato una trasformazione logaritmica in base 10 sul numero di itemset frequenti del tipo $T(x) = \log_{10}(x)$, dal momento che questi avevano ordini di grandezza molto diversi tra loro.

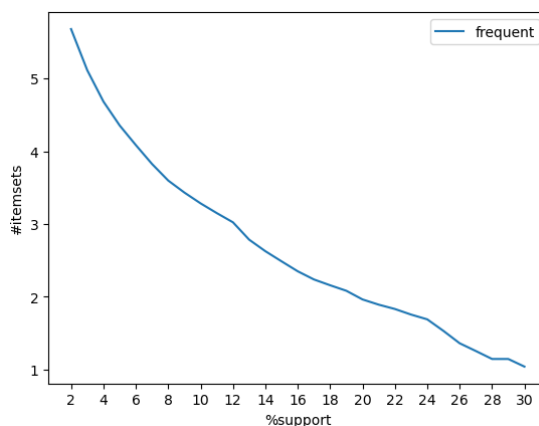
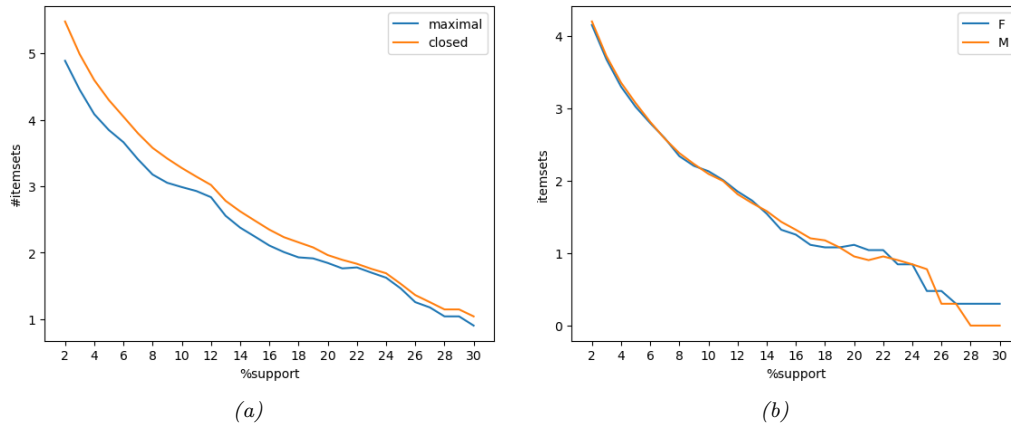


Figura 4.1: Numero di itemset frequenti in funzione del supporto

Dal grafico sopra è possibile osservare come il numero di itemset frequenti diminuisca velocemente all'aumentare del supporto, in particolare si passa ad avere un numero di itemset frequenti molto alto per supporti bassi, con un ordine di grandezza compreso tra 10^4 e 10^6 . Successivamente, per valori del supporto superiori a circa il 20%, il numero di itemset frequenti rimane ben al di sotto del centinaio, fino ad essere solo otto itemset frequenti per il supporto pari al 30%. Sempre utilizzando lo stesso criterio, riportiamo in figura (a) il numero di itemset frequenti massimali e chiusi per `%supporto∈[2%,30%]`: per ciascun valore del supporto il numero di itemset massimali risulta essere sempre strettamente inferiore di quello degli itemset chiusi. Riportiamo, infine, in figura (b) il numero di itemset massimali distinguendo rispetto al sesso, con trasformazione logaritmica analoga alla precedente.



4.2 Estrazione di Association Rules

Siamo poi passate a studiare le association rules relative al dataset analizzato, focalizzandoci inizialmente su quelle che presentano come conseguente un valore appartenente alle variabili *emotion* oppure *sex*. Per quanto riguarda la variabile *emotion*, ci siamo soffermate su association rules con confidenza=50, 55, 60 % e supporto variabile tra il 10% e il 30%. Riportiamo nella tabella seguente il numero di regole che hanno come conseguente un'emozione con confidenza=55% e supporto variabile tra il 2% e il 15%.

	2	3	4	5	6	7	8	9	10	11	12	13	14	15
angry	5666	1138	323	131	65	29	16	9	6	4	2	1	0	0
calm	923	149	27	14	7	2	1	1	0	0	0	0	0	0
disgust	1	0	0	0	0	0	0	0	0	0	0	0	0	0
fearful	42	4	1	0	0	0	0	0	0	0	0	0	0	0
happy	2	0	0	0	0	0	0	0	0	0	0	0	0	0
neutral	0	0	0	0	0	0	0	0	0	0	0	0	0	0
sad	7	0	0	0	0	0	0	0	0	0	0	0	0	0
surprised	4	0	0	0	0	0	0	0	0	0	0	0	0	0

Come mostra la tabella, non è presente alcuna regola di associazione che abbia come conseguente l'emozione *neutral*, e solo l'emozione *angry* presenta alcune regole che la hanno come conseguente con un supporto significativo. Per questo motivo abbiamo deciso di concentrarci prevalentemente sull'emozione *angry*, come approfondiremo nella sezione successiva.

La nostra analisi ha portato anche alla scoperta di altre regole, che presentano altre variabili come conseguente della regola di associazione, che riteniamo essere piuttosto interessanti, visti i valori elevati di confidenza, supporto e lift. Ne riportiamo di seguito una, che ha come conseguente il valore di *emotional_intensity strong*. È possibile visualizzare le caratteristiche della regola nella tabella sottostante in figura 4.2. Riportiamo accanto il grafico della variabile *mfcc_std*, presente nell'antecedente.

consequent	strong
antecedent	((83.61999999999999, 122.54]_mfcc_std,)
abs_support	495
%_support	20.650814
confidence	0.825
lift	1.789615

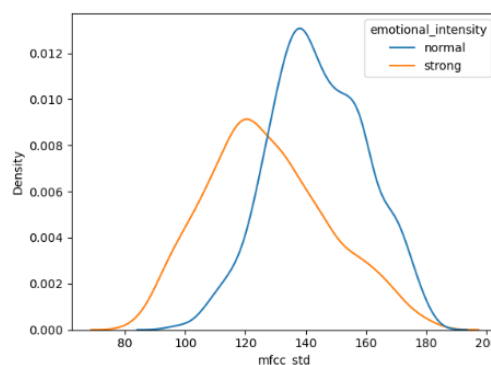


Figura 4.2: Caratteristiche della regola e distribuzione della variabile *mfcc_std*

4.2.1 Association rules per predire la variabile target

In questa sezione andiamo ad approfondire alcune regole di associazione che possono essere utili nell'identificazione delle variabili target. Questo perchè presentano nel conseguente della regola uno specifico valore della variabile target presa in esame. Prenderemo in esame, come fatto precedentemente, *sex* e *emotion*.

Variabile target *sex*

Presenteremo due regole che riteniamo essere particolarmente significative, visto l'alto valore della confidence e un valore significativo del lift, che predicano una la classe *M* e l'altra *F*. Possiamo vedere le caratteristiche della prima regola nella tabella sottostante 4.3. Riportiamo accanto alla tabella, la distribuzione della variabile presente nell'antecedente (*stft_std*).

consequent	M
antecedent	((0.209, 0.318]_stft_std,)
abs_support	547
%_support	22.820192
confidence	0.911667
lift	1.797093

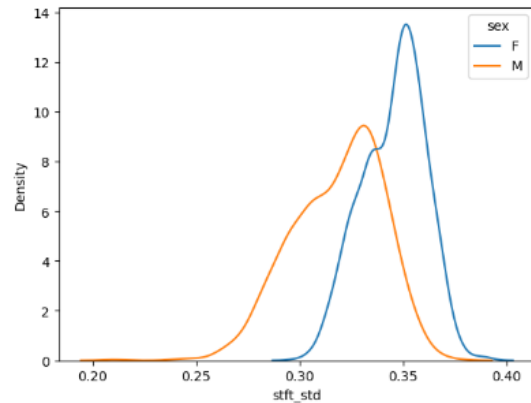


Figura 4.3: Caratteristiche della regola e distribuzione della variabile *stft_std*

La seconda ha come conseguente *F*. Come nel caso precedente, troviamo tutte le caratteristiche della association rule nella tabella in figura 4.4. Accanto è riportato uno scatterplot che mappa i dati rispetto alle due variabili presenti nell'antecedente della regola di associazione, e li distingue rispetto al sesso. È possibile notare come nel rettangolo specificati dalla regola i dati presenti sono prevalentemente di sesso femminile.

consequent	F
antecedent	((0.349, 0.392]_stft_std, (-0.001, 0.000202]_stft_min)
abs_support	515
%_support	21.48519
confidence	0.924596
lift	1.876593

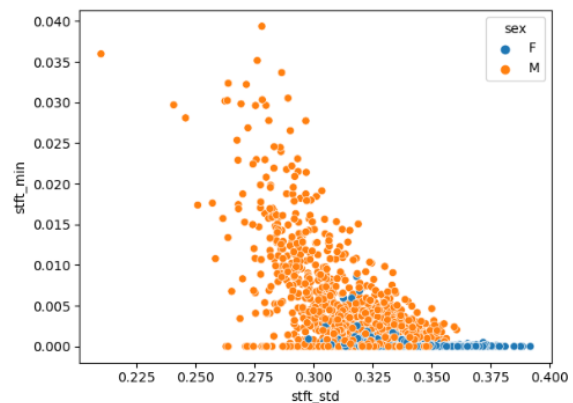


Figura 4.4: Caratteristiche della regola e scatter plot delle variabili *stft_std* e *stft_min*

Variabile target *emotion*

Riportiamo in questa sezione una sola regola, che abbiamo trovato essere particolarmente significativa, che ha come conseguente l'emozione *angry*. Le caratteristiche della regola, con l'antecedente, il supporto, la confidence e il lift, sono riportate nella tabella sottostante. Come è possibile notare, la confidence di questa regola non è altrettanto alta come quelle illustrate in precedenza, come era possibile aspettarsi, visto che le emozioni, al contrario di *sex* o di *emotional_intensity*, assumono otto valori diversi, invece che due. Abbiamo ritenuto questa regola particolarmente interessante perché abbiamo ritrovato in molte altre regole, con supporto e confidence simili, se non peggiori, le due variabili presenti nell'antecedente.

consequent	angry
antecedent	((-25.528, -16.819]_mfcc_mean, (-32.716, -16.354)_intensity
abs_support	192
%_support	8.010013
confidence	0.578313
lift	3.716399