Classificazione generi musicali mediante reti evolute con algoritmo NEAT

Introduzione

Lo scopo di questo lavoro è studiare i problemi (e possibilmente le relative soluzioni) che si incontrano nel classificare generi musicali mediante reti neurali.

Setup

Tutti gli esperimenti sono stati condotti su file formato MIDI da cui sono state estratte tutte le 156 feature "scalari" (ossia quelle il cui valore fosse un solo valore reale, non un vettore) offerte dall'applicazione *jSymboli*c (http://jmir.sourceforge.net/jSymbolic.html).

Per l'evoluzione della rete, è stato realizzato un progetto in linguaggio Python, che sfrutta la libreria *neat-python* (https://github.com/CodeReclaimers/neat-python) come implementazione dell'algoritmo NEAT.

Per l'allenamento (ossia per l'evoluzione) della rete, sono stati realizzati diversi training set.

L'analisi, inizialmente, è stata limitata a solo due generi, arbitrari, ("classica" e "jazz"), scelti solamente per la grande quantità di file MIDI liberamente disponibile in rete; in un secondo momento, è stato aggiunto un terzo genere (il "rock") scelto arbitrariamente con gli stessi criteri. In totale, ogni training set è costituito da percentuali pressoché uguali di brani dei generi interessati, e da non meno di 200 brani per genere. I tre generi sono stati quindi raggruppati in dataset corrispondenti a tutte le quattro combinazioni possibili (le tre coppie e il dataset comprensivo di tutti). Questa decisione è stata presa per studiare le interazioni tra i vari generi e in particolare per analizzare come le single feature estratte possano essere "caratteristiche" o meno di un singolo genere. È stato inoltre costruito un ulteriore set di controllo (anche questo in 4 varianti), composto da brani che in nessun caso la rete vedrà mai durante la sua evoluzione e che serve come unico scopo a valutare il comportamento della rete su brani mai visti (essenzialmente, a valutare la presenza di *overfitting*). La dimensione del set di controllo è di circa 50 brani per ogni genere.

Bisogna specificare che ad ogni brano è sempre stato assegnato uno e un solo genere, e che la classificazione (benché controllata a mano da chi scrive) è pur sempre arbitraria e pertanto può contenere errori. La dimensione relativamente elevata dei vari training set dovrebbe servire proprio ad attenuare questi fenomeni.

La fitness di ogni individuo è stata valutata come 1 – l'errore medio di classificazione sul training set, ulteriori dettagli saranno forniti nel seguito.

I dati proposti nel seguito fanno riferimento a varie run dell'algoritmo, con vari parametri specificati di volta in volta, ma sempre per un totale di 1000 generazioni (o fino al raggiungimento di almeno 99.5% di correttezza nella classificazione). Può sembrare limitante e sicuramente arbitrario, ma, come sarà più chiaro nel seguito, non tutti i generi sono ugualmente "facili" da distinguere e un limite

fissato di generazioni aiuta a valutare questa "difficoltà" in termini di "bontà" della soluzione dopo un numero fissato di generazioni (in alternativa si sarebbero potute valutare le generazioni necessarie per ottenere una fissata soglia di fitness, cosa che però porta facilmente a lunghissimi tempi d'esecuzione in caso di problemi difficili).

Tutte le run sono state eseguite da 5 processi concorrenti su processore Intel i7-6700k a 4.0GHz e 16 GB di RAM. Sono stati anche fatti tentativi di computazione sulla GPU, ma con scarsi risultati (oltre alla complessità aggiunta, il problema in esame non si presta ottimamente al calcolo su GPU in quanto l'overhead di trasmissione risulta maggiore del guadagno, probabilmente a causa del numero relativamente basso dei dati e delle scarse doti di programmazione su GPU di chi scrive).

A ogni generazione è mantenuta una popolazione di circa 2000 individui.

Le decisioni e i risultati

Fin dai primissimi esperimenti è emersa la necessità di stabilire come mappare ogni genere su un valore reale, esprimibile dalla rete. Dovendo inizialmente classificare due generi, la scelta più naturale è stata mapparli sui valori 0 e 1. La rete avrebbe quindi prodotto solo ed esclusivamente valori in quell'intervallo (mediante opportune funzioni d'attivazione) e la fitness di ogni rete si sarebbe potuta valutare come

$$1 - \frac{\sum |output\ atteso - output\ ottenuto|}{grandezza\ del\ dataset}$$

Ossia come 1 – l'errore medio di classificazione. Essendo la massima differenza tra output atteso e output ottenuto pari a 1, anche la fitness avrebbe assunto solo valori compresi tra 0 e 1. Si noti, che valori di output casuali, produrrebbero una fitness media del 50%.

Questo modello, nella sua semplicità, ha fin da subito dato risultati sorprendentemente buoni.

La tabella sottostante, riassume i punteggi medi sui dati di controllo (mai visti dalla rete durante l'evoluzione) degli individui migliori dopo 1000 generazioni.

Dataset	Score medio
Classica / Rock	98.67 %
Classica / Jazz	97.36 %
Jazz / Rock	86.49 %

Tabella 1 Fitness media sul set di controllo

Da questi dati si evince chiaramente che, come è facilmente intuibile, distinguere il Jazz dal Rock è più arduo rispetto agli altri. Le cose si complicano ulteriormente, se si cerca di distinguere i tre generi insieme. Per prima cosa, occorre trovare una codifica per il terzo genere. Tuttavia, data la natura

essenzialmente lineare della rete, un qualsiasi valore reale intermedio tra gli altri vorrebbe dire vincolare un legame tra i generi. Di fatti, come si evince più chiaramente dai dati nel seguito, la qualità della soluzione è sempre più vicina a quella di valori casuali. Per questo motivo, si è cercata un'altra codifica dell'output. La cosa più ragionevole pensata, è stata di rappresentare ogni genere con un distinto nodo di output. In questo modo la rete avrebbe potuto evolvere indipendentemente ciò che ritiene significativo per un genere, producendo come uscita un vettore di "gradi di confidenza" per cui un singolo brano sia di un dato genere. I vari generi, quindi sono codificati come "versori" in un ideale spazio vettoriale.

Analogamente a prima, la fitness viene calcolata come:

$$1 - \frac{\sum ||output\ atteso - output\ ottenuto||}{grandezza\ del\ dataset}$$

Dove anziché un semplice valore assoluto, ora si calcola una norma vettoriale. Si noti che benché concettualmente analogo, questo metodo è ora più "severo" nei confronti degli individui, perché il massimo errore possibile è maggiore di 1 (in caso di vettori "diametralmente opposti" la distanza massima è infatti pari alla radice quadrata del numero di generi) e di conseguenza anche l'errore medio prodotto da valori casuali. Infatti, mentre aggiungendo valori "accettati" in un intervallo si aumenta la probabilità che un valore casuale sia accettabile, aggiungendo una dimensione all'output si introducono infinite combinazioni di valori non accettabili, di fatto rendendo molto meno probabile che un valore casuale sia accettabile.

Dataset	Dimensione output	Score medio	Score medio atteso da valori casuali
Classica / Jazz / Rock	1	85.92%	66.66%
Classic / Jazz / Rock	3	75.27%	3.01%

Tabella 2 Fitness media sul set di controllo (caso 3 generi)

A questo punto si è cercato di vedere se l'aumento di dimensione possa giovare anche agli altri problemi, ottenendo i risultati riportati di seguito.

Dataset	Score medio a dimensione 1	Score medio a dimensione 2	Score medio atteso da valori casuali
Classica / Rock	98.67 %	98.50 %	23.48 %
Classic / Jazz	97.36 %	96.28 %	23.48 %
Jazz / Rock	86.49 %	78.54 %	23.48 %

Tabella 3 Fitness media sul set di controllo con valore atteso da output casuali per dimensione 2

Come si evince, le differenze appaiono minime e poco.

Come si accennava prima, però, la metrica dell'errore medio penalizza le configurazioni con output di dimensione superiore. Per valutare le configurazioni in maniera più equa, per tanto, si riportano di seguito le percentuali medie dei file classificati correttamente nel control set. I dati riportati, sono ottenuti considerando un file correttamente classificato se la sua distanza dall'output atteso risulta minore di una soglia (fissata arbitrariamente a 0.2). Poiché anche in questo caso si misura una distanza, teoricamente, ancora si penalizzano le esecuzioni con output a dimensione maggiore, ma in pratica, analizzando i risultati ottenuti dalle varie reti, nella quasi totalità dei casi l'output prodotto corrisponde esattamente a un genere (e quindi, se errato, con distanza molto maggiore di 0.2). In altre parole, il numero di "falsi negativi" ossia di campioni per cui la rete abbia prodotto un output "nella direzione giusta" ma "non abbastanza pronunciato" da rientrare nella soglia sono una minoranza assolutamente trascurabile, che quindi non pregiudica l'imparzialità del test.

		Dataset			
		Classic/Rock	**Classic/Jazz**	**Jazz/Rock**	**Classic/Jazz/Rock**
Dimensione output	**1**	98.68%	97.29%	85.10%	72.55%
	2	98.73%	97.22%	85.07%	-
	3	-	-	-	79.11%

Tabella 4 Percentuale di file classificati correttamente nel set di controllo

Come si evince dalla tabella riportata, in caso di distinzione tra due soli generi non si notano differenze apprezzabili. Nel caso invece di distinzione di 3 generi, si guadagna precisione (anche se la fitness tenderebbe a fornire valori più pessimistici). Probabilmente, questo guadagno è dovuto al non aver vincolato a priori nessun legame tra i generi, cosa che invece accade dovendo esprimere più di due possibili categorie con un unico valore reale.

Al fine di scongiurare il pericolo di overfitting, visti i risultati iniziali apparentemente troppo promettenti, è stata realizzata una piccola variante all'algoritmo NEAT standard, che prevede di sostituire una percentuale del training set con dati nuovi mai visti ogni N generazioni. Ovviamente, i dati sostitutivi immessi sono stati presi da un terzo insieme di dati costruito appositamente, in modo da mantenere l'imparzialità dei dati di controllo (che quindi rimangono ignoti in fase d'evoluzione).

Per motivi di tempo, non sono stati effettuati test con più configurazioni possibili e si è scelto di fissare il numero di generazioni tra una modifica dei dati e la successiva a 50 e la percentuale di modifica al 10%. Questo significa che, in questa variante, NEAT gira normalmente per 50 generazioni, dopo di che il 10% del training set (completamente a caso) viene sostituito da altrettanti dati presi da un altro set; i dati rimossi dal training set vengono immessi nell'insieme di dati a disposizione per le prossime modifiche (e quindi potrebbero essere reinseriti). Si noti che per motivi di tempo, non è stato effettuato nessun controllo su come avvenga la scelta di brani da sostituire e sulla scelta dei brani con cui sostituire. Un approccio più evoluto potrebbe impiegare delle euristiche più raffinate per selezionare i brani più "significativi" o per garantire che rimanga costante la proporzione tra i generi contenuti nel dataset.

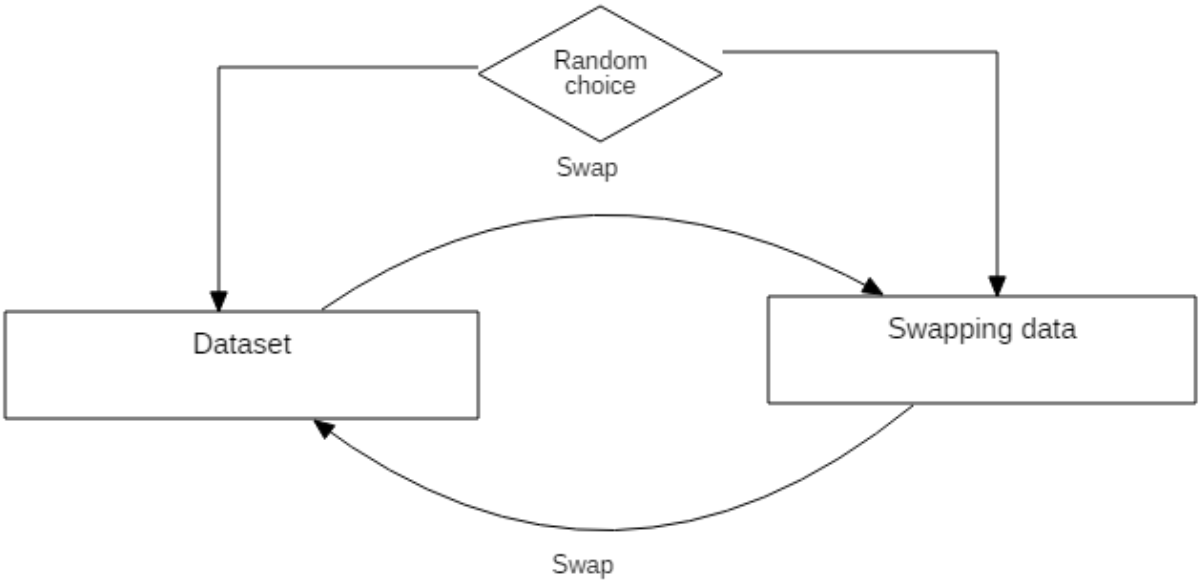

Figura 1 Schema di principio della sostituzione

Questo algoritmo, nel suo essere intrinsecamente più severo sugli individui, ha prodotto i seguenti valori di fitness media sul control set.

Dataset	**Score medio a dimensione 1**	**Score medio a dimensione N**
Classica / Jazz / Rock	85.75 %	75.86 % (N = 3)
Classica / Rock	98.95 %	98.23 % (N =2)
Classica / Jazz	97.71 %	96.53 % (N =2)
Jazz / Rock	90.49 %	87.14 % (N =2)

Tabella 5 Fitness media sul control set (MTS)

Come si evince confrontando con la tabella precedente, i risultati sono leggermente migliorati. Questo probabilmente in virtù del fatto che quest'algoritmo promuove la variabilità delle soluzioni e non premia soluzioni che pur non essendo propriamente overfitting, siano dovute a ridondanze del training set (come ad esempio brani troppo simili tra di loro, tutti di uno stesso autore e via dicendo).

Come nel caso precedente, si riportano anche le percentuali medie di file correttamente classificati, al fine di mostrare un più equo metro di valutazione.

Dataset					
Dimensione output		**Classic/Rock**	**Classic/Jazz**	**Jazz/Rock**	**Classic/Jazz/Rock**
	1	98.95%	97.71%	89.56%	69.99%
	2	98.60%	97.40%	90.91%	-
	3	-	-	-	81.43%

Tabella 6 Percentuale di brani classificati correttamente nel control set

Come si può osservare, anche in questo caso, per la distinzione tra due soli generi non si apprezzano miglioramenti significativi. Invece, nel caso di distinzione tra tre generi, la percentuale di brani classificati correttamente aumenta del 2%, a dimensione 3 e cala del 3% circa a dimensione 1.

In ultima analisi, si è cercato di stabilire quali feature siano più significative per ogni genere e pertanto si è realizzato un meccanismo di "ranking" di ogni feature.

In prima approssimazione, lo score di ogni feature è dato dalla media pesata (in base allo score ottenuto dagli individui migliori sui dati di controllo, al quadrato) del peso che quell'individuo attribuiva alla data feature. Il peso attribuito dall'individuo alla feature, in caso in cui la feature sia collegata direttamente all'output, è dato dal peso della connessione, altrimenti (o più in generale) dal prodotto dei pesi delle connessioni lungo il cammino fino al neurone di output. Questi valori sono stati calcolati per ogni combinazione di dataset, algoritmo e codifica dell'output e riportati, scalati per motivi "grafici", qui di seguito.

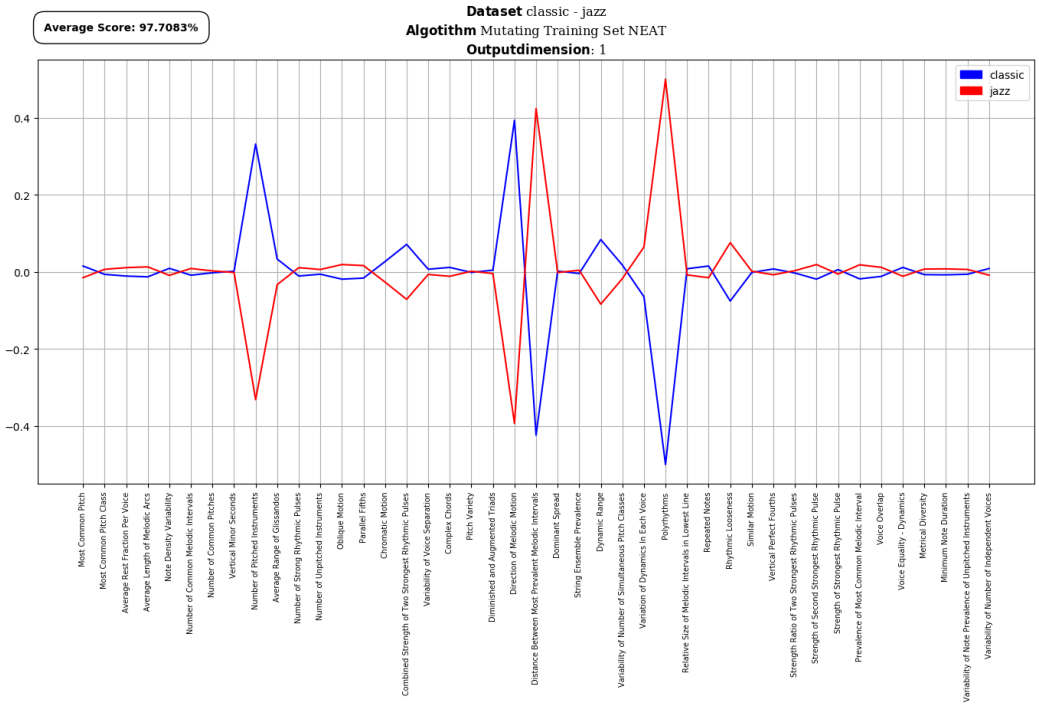

Grafico 1 MTS Classic/Jazz 1

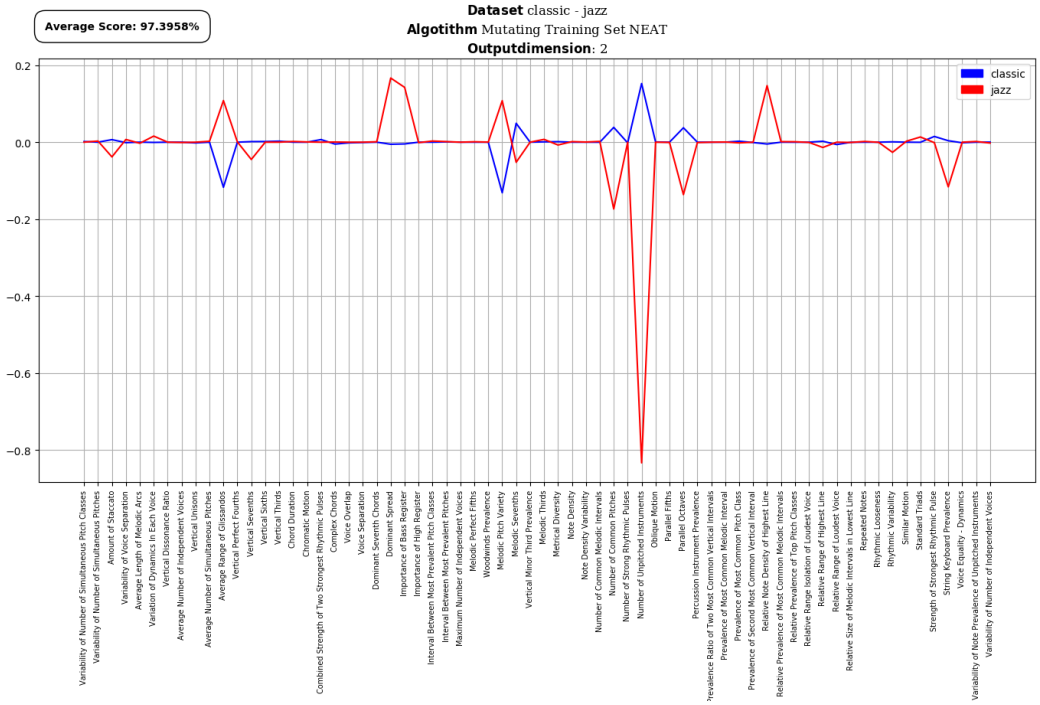

Grafico 2 MTS Classic/Jazz 2

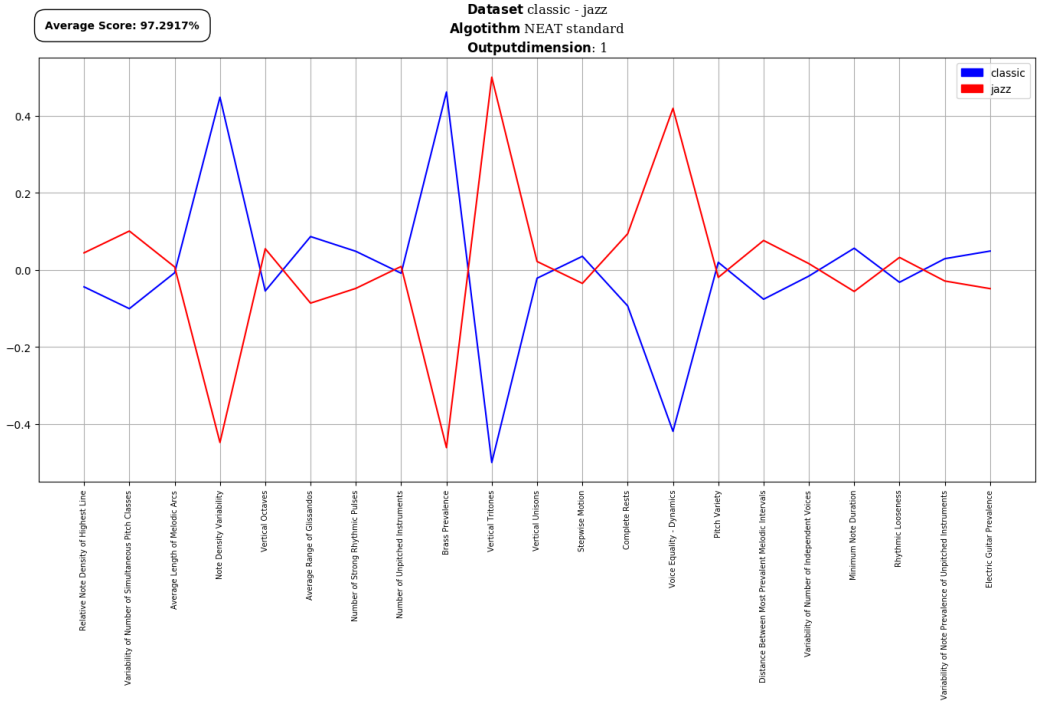

Grafico 3 Classic Jazz 1

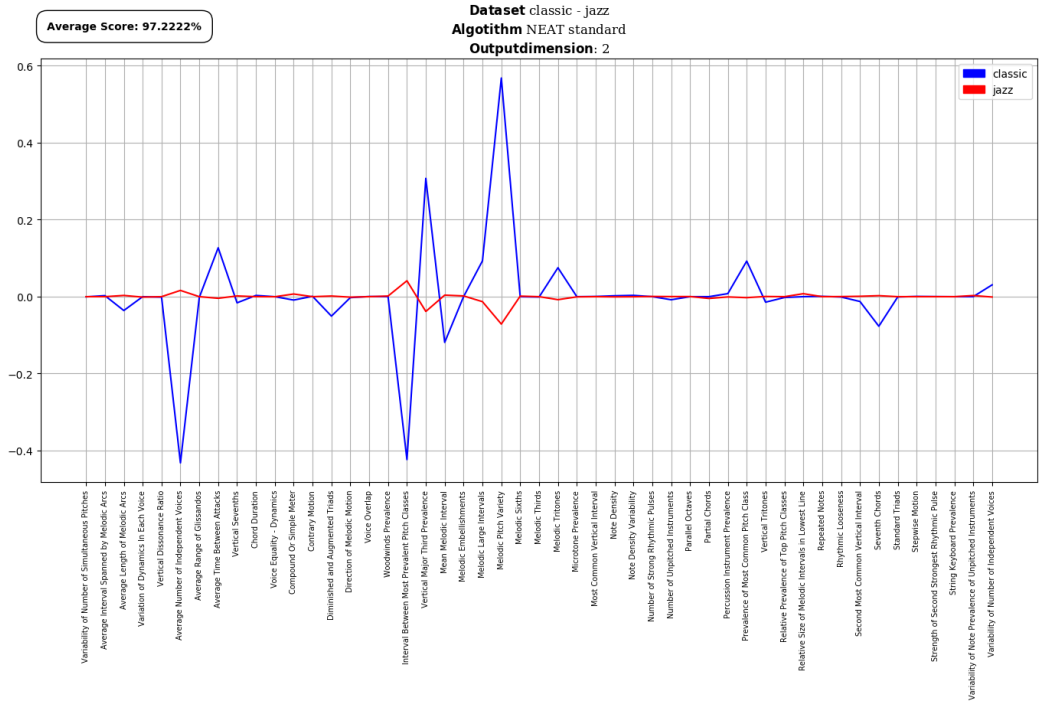

Grafico 4 Classic Jazz 2

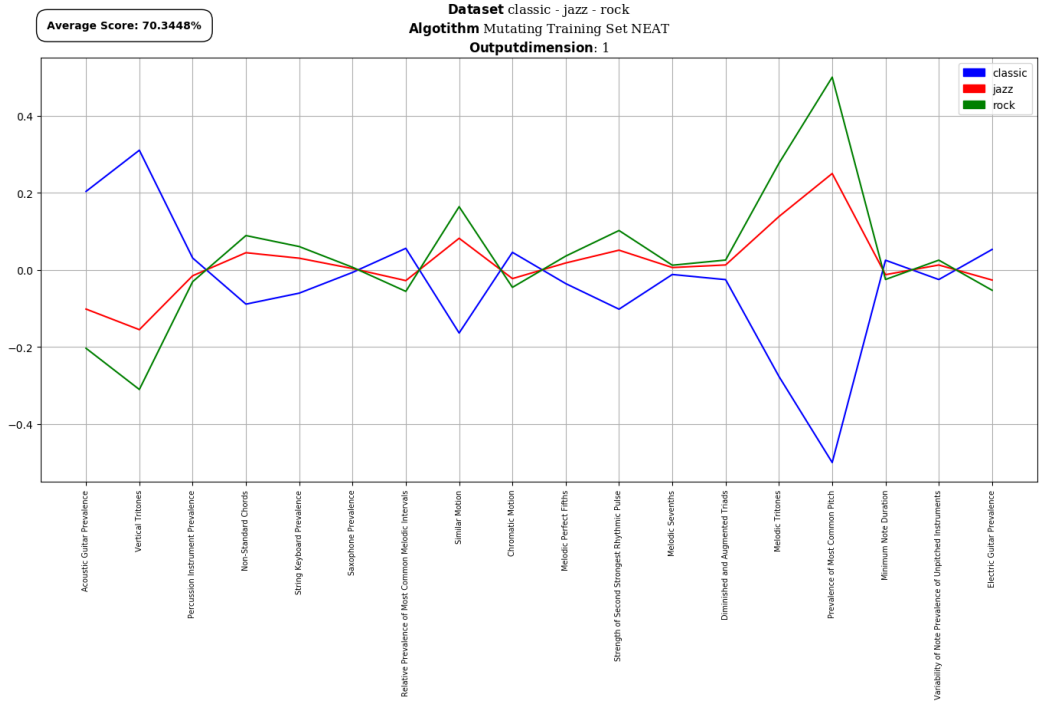

Grafico 5 MTS Classic/Jazz/Rock 1

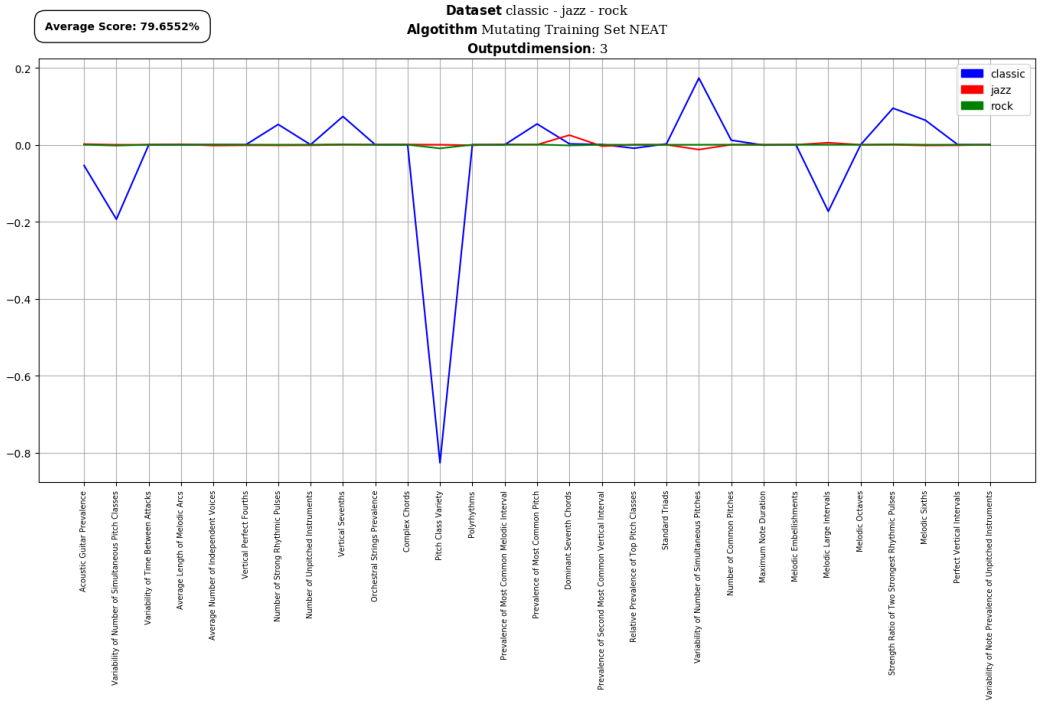

Grafico 6 MTS Classic/Jazz/Rock 3

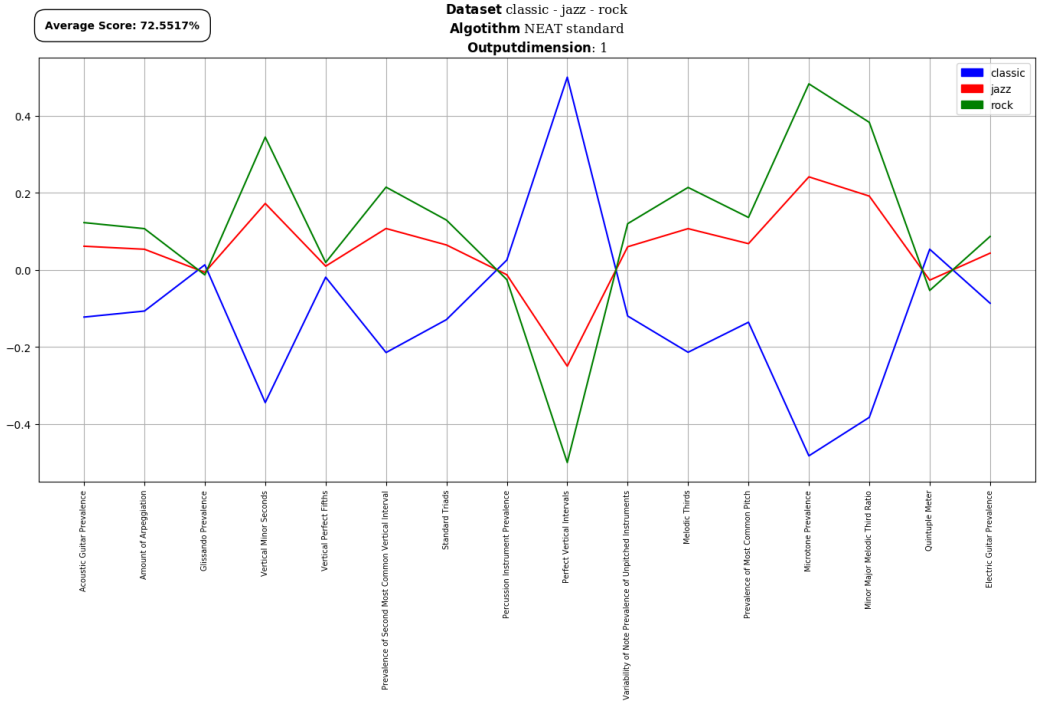

Grafico 7 Classic/Jazz/Rock 1

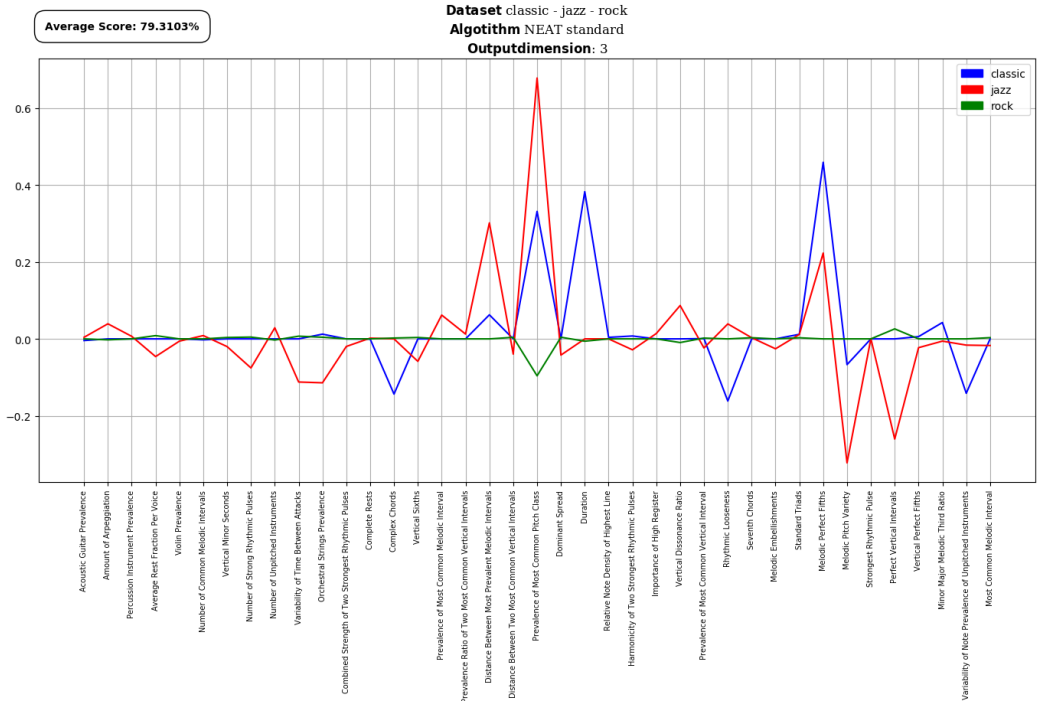

Grafico 8 Classic/Jazz/Rock 3

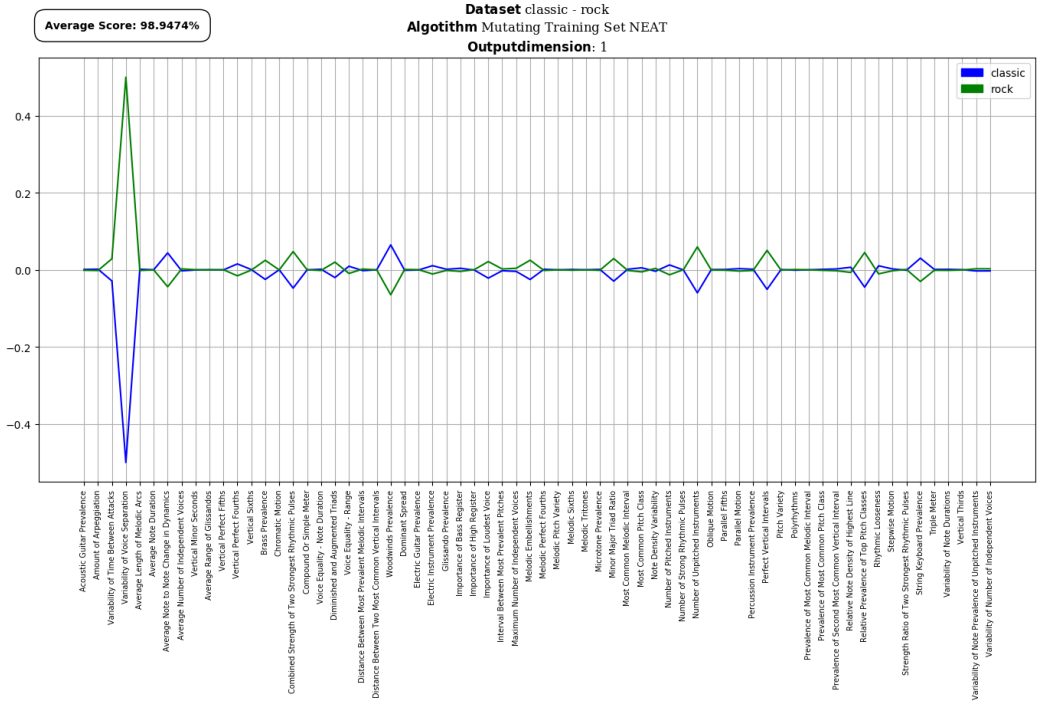

Grafico 9 MTS Classic/Rock 1

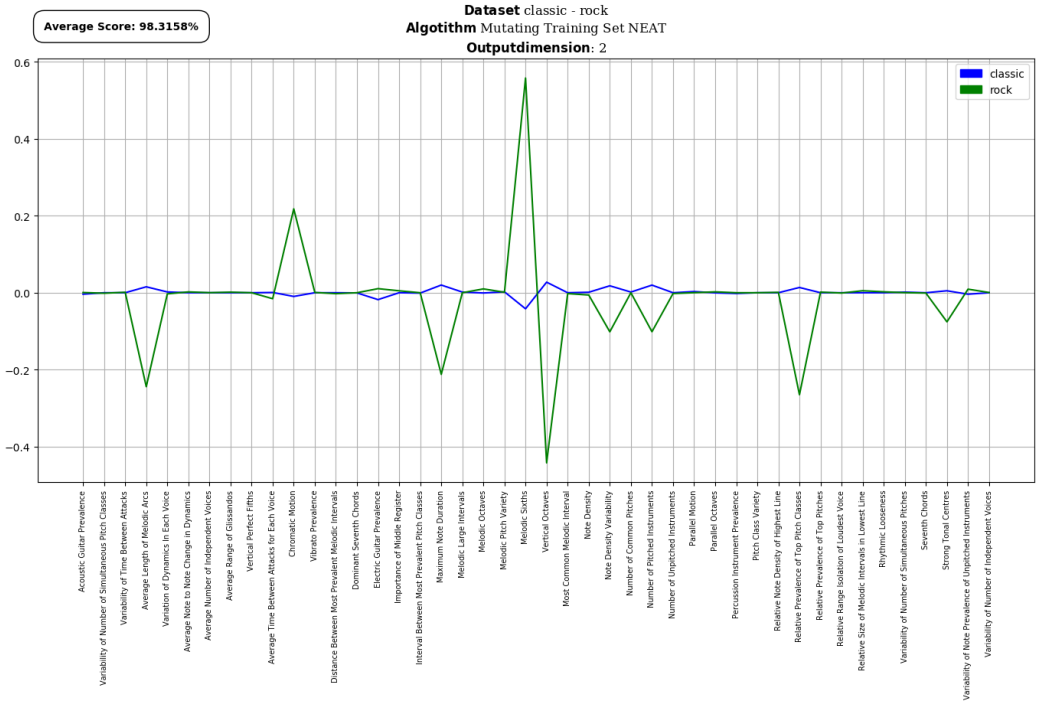

Grafico 10 MTS Classic/Rock 2

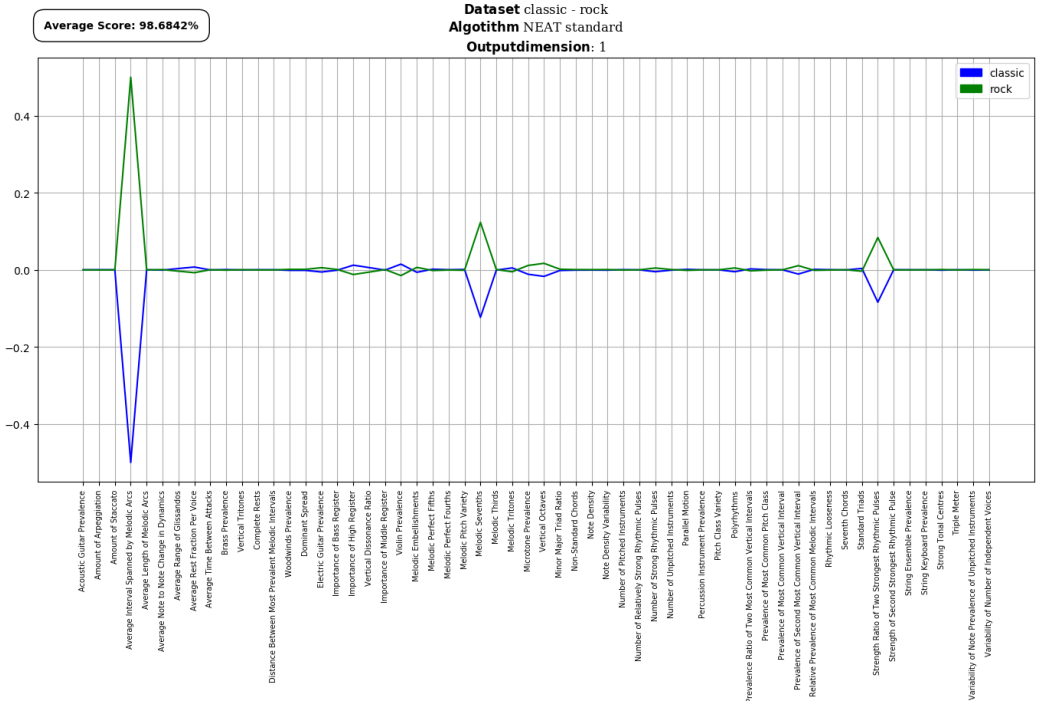

Grafico 11 Classic/Rock 1

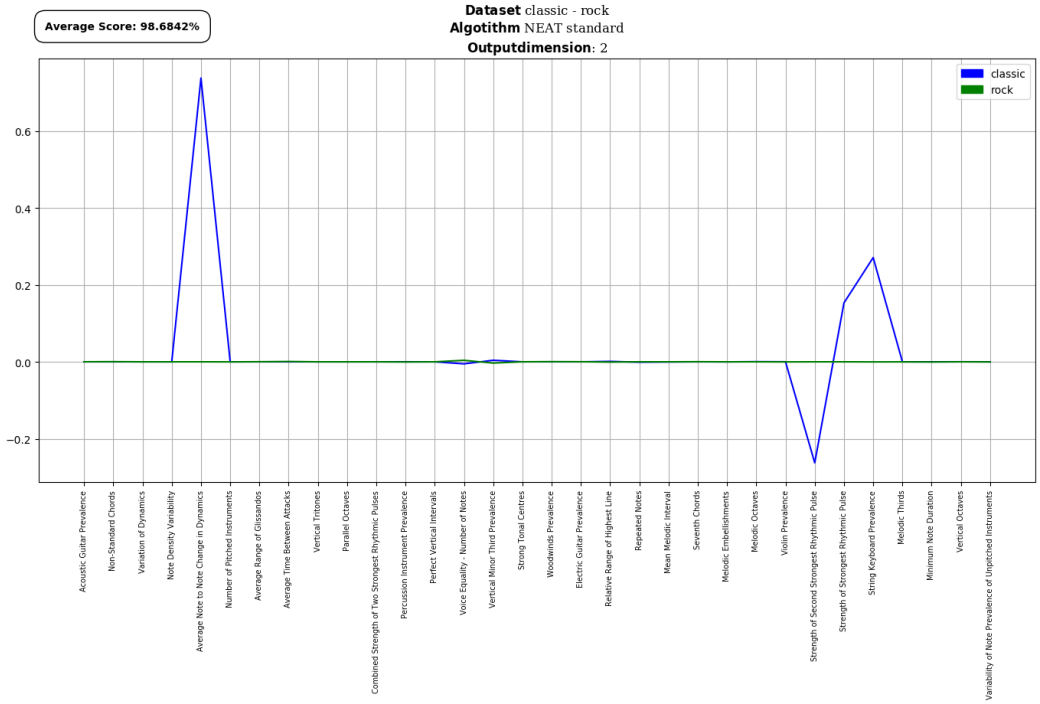

Grafico 12 Classic/ Rock 2

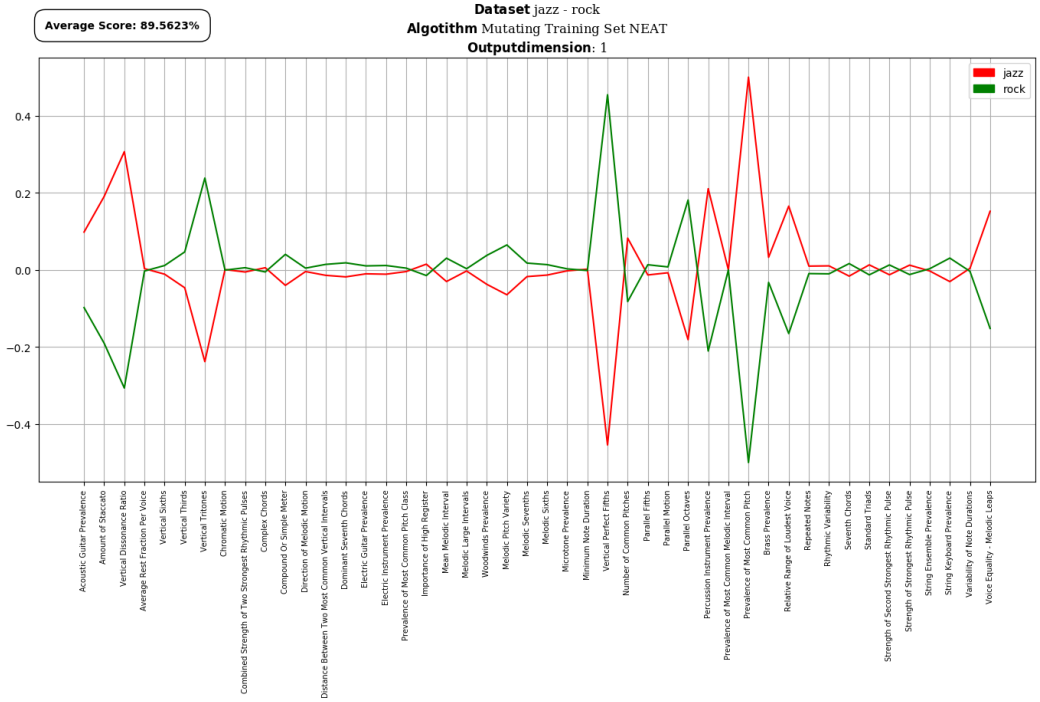

Grafico 13 MTS Jazz/Rock 1

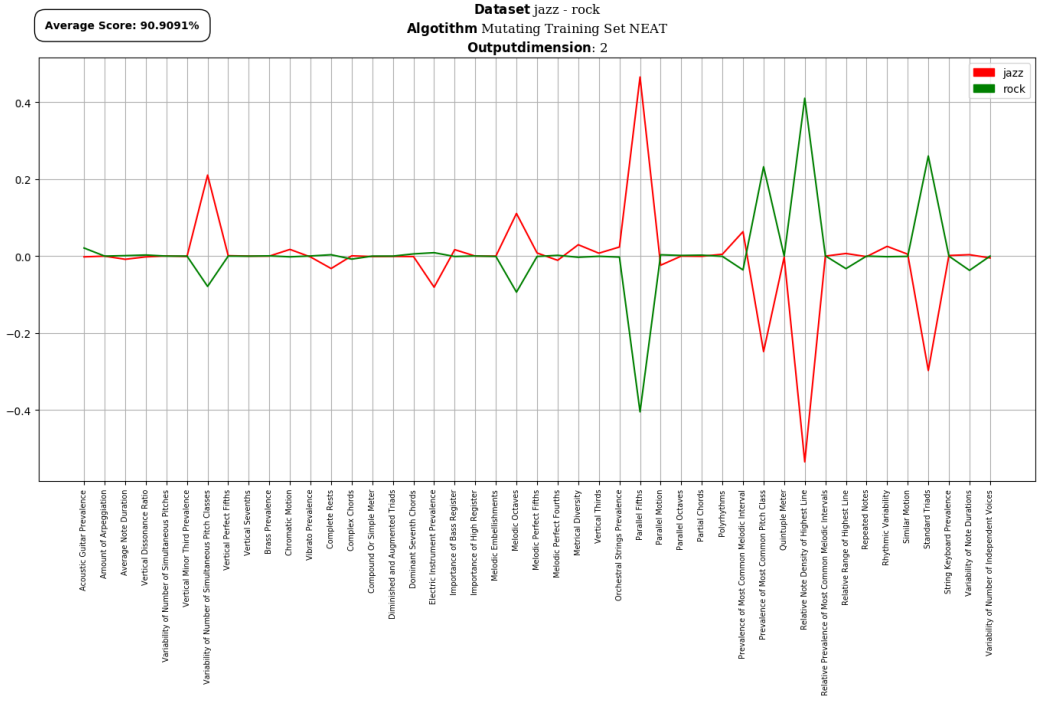

Grafico 14 MTS Jazz/Rock 2

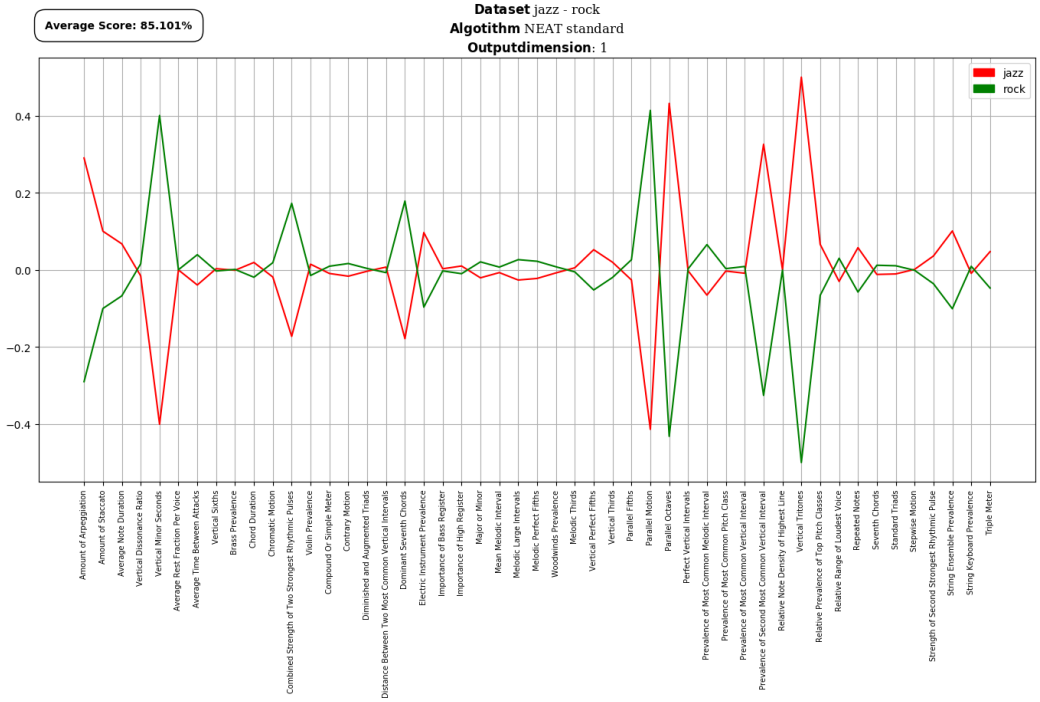

Grafico 15 Jazz/Rock 1

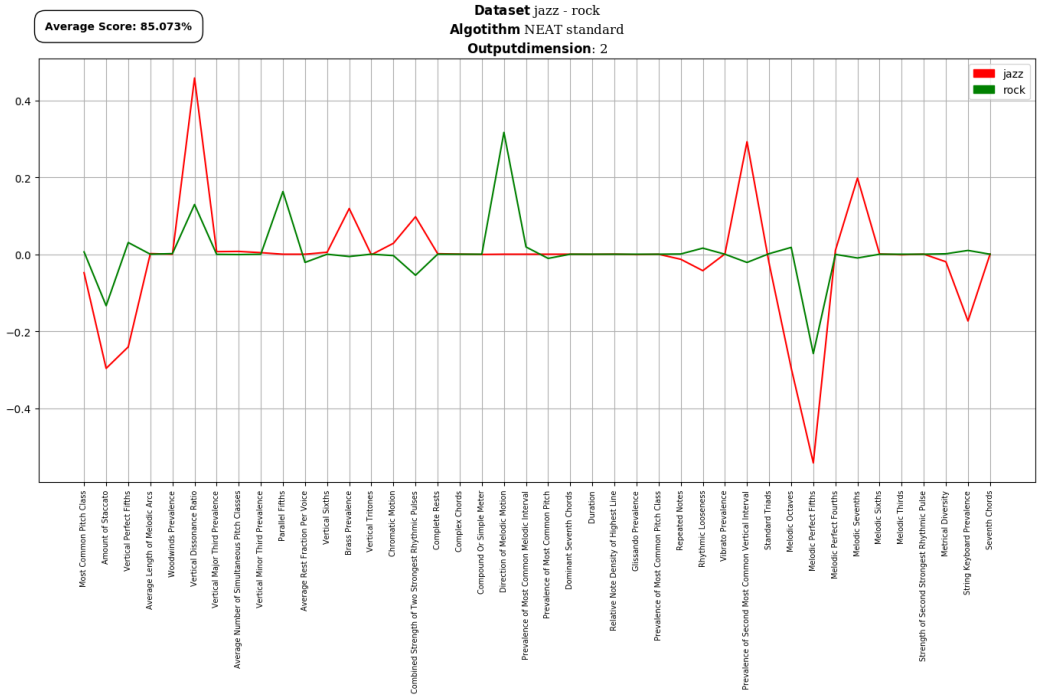

Grafico 16 Jazz/Rock 2

Un aspetto significativo che si evince da alcuni grafici è la presenza di simmetria rispetto allo zero.

Mentre questo è scontato nel caso in cui l'output sia rappresentato con dimensione 1 (ciò che spinge verso un genere allontana e contribuisce negativamente per l'altro), non è banale nel caso a dimensione superiore, che sono liberi di evolvere ogni output indipendentemente. Difatti, gli individui migliori, già dopo poche generazioni avevano colto la natura "binaria" del training set (se è un genere non è l'altro) sviluppando forti correlazioni negative tra i neuroni di output. Queste correlazioni negative, si riflettono quindi in score opposti per la stessa feature e quindi in una simmetria. Per un esempio di ciò, si guardino i grafici 14 e 10.

Si osserva inoltre come in generale un solo neurone di output comporti un minor numero di feature usate. Mentre aver completa carta bianca spinge la rete a provare più feature e a tararle individualmente, il dover tener conto dell'interazione su un singolo output di tutte spinge verso individui con meno feature, ma più nettamente pronunciate. Un effetto simile è anche prodotto dalla variante sviluppata dell'algoritmo NEAT. Sostituire il training set, porta gli individui migliori verso soluzioni più generali e che quindi comportano tipicamente più feature, se non altro come "retaggio" del training set precedente.

In conclusione, il dover distinguere due generi specifici è un problema che si risolve più velocemente con un solo neurone di output (data la natura binaria dell'input). Mentre per distinguere N generi, appare più conveniente la rappresentazione con N neuroni di output in quanto permette di non "cablare" legami di nessun tipo.

Sviluppi futuri

Uno sviluppo interessante sarebbe l'allargamento ad altri generi, probabilmente mantenendo uno schema con output a dimensione 1 solo per la distinzione tra 2 generi specifici, e uno schema a dimensione N per la distinzione di N generi contemporaneamente. Sarebbe anche interessante sviluppare un dataset non binario, in cui cioè ai brani non corrisponda un'etichetta sola, ma dei valori "fuzzy logic". Al mondo esistono migliaia di generi musicali, sottogeneri e generi affini, per cui una classificazione netta, benché fattibile su grandi categorie, diventa sempre più fallace man mano che si introducono generi affini. Un dataset in cui i brani siano classificati anche come "intermedi", nella mia personale opinione, modellerebbe molto meglio un mondo variegato. Per ottenere tale dataset, si potrebbe richiedere la collaborazione di un gruppo vario di persone, alle quali si chieda di classificare dei brani ciascuno con un solo genere. A ogni brano poi si assocerebbe un genere composto dalla distribuzione dei generi ricevuti. Ad esempio: dato un brano in cui il 20% delle persone lo ha ritenuto "jazz", 30% "blues" e 50% "rock", si etichetterebbe (0.2, 0.3, 0.5, 0, 0, 0…..).