

# Scientific Computing Final Project

## AI CUP 2020 - 歌聲轉譜競賽

組別：我係快樂欵甘蔗 man

資工二 B07902142 許庭維

資工二 B07902144 彭約博

資工三 B06902136 賴冠毓

/\*我們並未使用 ML 相關的方法，所以 report 中並無提及 model 相關資料，而是按照老師在社團中說明的 report 寫法\*/

### 1. 想法和方法

我們的想法很簡單，基本上就是把一首歌曲跑完，每次根據前一個音與當前音符之間音高(pitch)的差距，來判斷是否為兩個不同的聲音。

而當疑似出現破音(爆音)或雜訊的時候，我們再透過當前音符與前後一個音符之間 pitch 的比較，來決定是否該視此音為爆音，而將它排除掉。

最後如果當前的音和上個音 pitch 一樣，則檢查這個音的開始時間，減去上個音的結束時間，如果是 0 或是負值，代表這兩個應該要是同個音，就把這個音刪除，並且把上一個音的結束時間設成這個音的結束時間。

### 2. 如何逐步驗證方法的適用性

這很顯而易見。要判斷兩個不同的音，在不考慮響度(振幅)、音色的情況下，能作為依據的就是音高(頻率)了。

而破音的部分，就算是歌手，也可能不小心失誤，更何況雜訊這種外界的不可抗力因素了，所以這在現實生活中是的確可能發生的情況。

舉例來說：有一段歌聲是 Re So So Si Do Si La

如果在 Si 中途產生了一段雜訊 Mi，不做特殊處理的話會變成 Si Mi Si，但這段歌聲並沒有 Mi，而我們的方法便是將 Mi 跟前後兩個音符 Si 去比較，發現差異極大，而順利將 Mi 這個雜訊剔除。

最後檢查如果當前的音和上個音 pitch 一樣，而當前音的開始時間與上個音的結束時間的差是 0 或是負值，代表這兩個應該要是同個音(0 表示從後面被截斷，負值表示從中間被抽離)，只是不小心被我們分割了，要把兩個音重新併起來。

### 3. 如何根據你的發現來逐步調整方法

最早我們只完全根據前一個「音符」與當前音符之間 pitch 的差距，來判斷是

否為兩個不同的聲音，但我們隨後就發現了這樣會導致比對的次數過高，只要 pitch 差不超過 1，可以視為同一段(個)音，所以我們額外紀錄了前一個音的 pitch，若當前音符與前一個「音」的 pitch 不超過 1，即可視為同一個音，直接進入下個音符的比較。

而不久後我們也發覺疑似有破音、雜訊的可能，而例子在驗證方法的適用性中提過了，我們便根據這個例子，將我們的 code 升級成可以處理雜訊的情況。最後我們觀察 json 檔時，有時候明明應該要是同一個音的，卻不小心被我們分割成兩個音，所以我們才又去檢查這個音的開始時間與上個音的結束時間的差，來判斷這兩個音是否為同個音。

#### 4. Improvement over the baseline method

- (1)判斷時間加快(不需每個音符皆比較、比較前一個音高即可)
- (2)有能力處理雜訊這種常見的惱人情況

#### 5. Insight and conclusions

在與其他組討論的時候，我們發現到雜訊是個相當關鍵的因素。有些組別使用效率比我們更好的 DP，但卻沒有考慮雜訊的情況，因此排名落後我們，所以做 project 還是需要考慮真實可能發生的所有情況會比較好。不過我們的 code 似乎也還有優化的空間，也許可以試著改用去跟整首歌平均 pitch 做比較。

#### 6. Division of labor (請列出組員之間如何分工)

系級	學號	姓名	工作
資工二	B07902142	許庭維	提出演算法、優化算法
資工二	B07902144	彭約博	樂理知識、實作 code 並測試
資工三	B06902136	賴冠毓	撰寫 report、統整與總結