



Evolució de la durada de les cançons al llarg del temps

Joan Manel Garcia-Reyero Sais

Tutor:
Bernat Gascón Cabestany

Abstract

In this research we are going to study the duration of songs over the last sixty years. We start with an exploratory graphic analysis. The distribution of the songs' durations is almost Gaussian, with 4min being the mode. We then create a graphic with the averages of the durations per year. In it we see that before 1970 the average song duration stayed between 2.5 min and 3.5 min. But in the seventies it completely changed, and after the seventies we get averages between 3.5 min and 4.5 min. The graphic of standard deviations shows us that an increase in the dispersion of song durations precedes the increase of the averages: in the sixties the deviations increase from values between 0 min and 1 min to values between 1.5 min and 2.5 min.

Finally we introduce a hypothesis of what could have happened. A rule in radio stations, which said that songs could only last 2.5 min or less, kept the duration of songs short. During the sixties, with the creation of symphonic rock, the dispersion increases. Once in the seventies, the *2.5 minutes rule* gets canceled and bands are able to produce longer songs. Adding this to a popularization of symphonic rock, we see the averages grow.

This is, in a synthetic way, our theory of the evolution of songs duration. In further sections of the paper we defend it with graphics that help to see in a clearer way what was happening and why our theory could be true.

Índex

1	Hipòtesi	3
2	Extraient la informació	3
3	Mètode de treball	5
3.1	Gràfic utilitzant Matplotlib	5
4	Gràfic de distribució de les durades	6
5	Mitjana i desviació tipus	7
5.1	Mitjana	9
5.2	Desviació tipus	10
6	Anàlisi dels resultats	12
6.1	Primers anys	12
6.2	La transició	14
6.2.1	Anàlisi numèrica	15
6.2.2	Anàlisi històrica	15
6.3	Darrers anys	17
7	Conclusions	19
A	Codi	21

1 Hipòtesi

Green Day és una banda formada als anys vuitanta als Estats Units. Un dia, mentre estava escoltant música, vaig veure que les cançons més recents eren més llargues que les més antigues. En aquell moment se'm van ocórrer diferents hipòtesis per les quals això passava: la primera era que les cançons de *Green Day* s'havien fet cada cop més comercials a mesura que el grup s'anava fent més popular. És a dir, que van començar tocant punk, que és un gènere en el qual les cançons solen ser curtes, i després van passar a tocar música més comercial que podríem classificar com a rock, amb cançons amb una durada més gran. L'altre raó que se'm va passar pel cap va ser que podia ser que la música en general hagués augmentat la durada de les seves cançons. Tot seguit, vaig anar a comprovar si això passava amb la resta de grups que tenia en el meu reproductor. Vaig mirar primer un grup més antic, *Ramones*, (1976-1996), i després vaig mirar grups més moderns com *Pierce the Veil* o *My Chemical Romance*, els dos fundats al segle XXI, i vaig veure que la meua hipòtesi es complia, les cançons de *Ramones* eren molt més curtes que les dels grups més actuals.

Per tant, vaig decidir fer l'estudi per veure si això que es complia en les meves escasses 150 cançons també passava amb la resta de cançons relativament modernes (entendem com a modernes les cançons dels últims cent anys).

El present estudi farà una anàlisi estadística basada en la base de dades de *Million Song Dataset*, analitzant la hipòtesi que la durada de les cançons ha augmentat al llarg del temps. Un cop fet el gràfic que ens donava la mitjana aritmètica de les durades de cada any vaig veure que des dels anys seixanta als setanta hi havia un increment important de les mitjanes de les durades, i vaig decidir centrar-me sobretot en aquell període de temps.

2 Extraient la informació

La informació està extreta d'una pàgina web anomenada *Million Song Dataset* (labrosa.ee.columbia.edu/millionsong), que pertany a una fundació anomenada *Labrosa*.

Segons la seva pàgina principal, *Million Song Dataset* és: “Una col·lecció gratuïta de metadada d'un milió de cançons contemporànies i populars”. El que volem per fer un estudi sobre si la durada de les cançons ha crescut al

llarg del temps és una base de dades amb moltes cançons, com més millor, i que inclogui cançons tan antigues com sigui possible. També ha d'incloure, per a cada cançó l'any en què la van treure i la seva durada. La base de dades trobada compleix tots els requisits: té un nombre elevat de cançons (un milió, encara que un cop eliminades les que no contenen la durada o la data en queden 515394), i les cançons contenen: el títol, la durada, l'any en que la van treure i el compositor. Per tant, compleix tots els requisits.

Només faltaven un parell de condicions per comprovar: si era un projecte fiable i si les cançons eren de gèneres heterogenis, ja que si no no estariem estudiant l'evolució de la durada de la música, estariem analitzant la d'uns gèneres concrets i la mostra no seria vàlida pel que jo volia estudiar.

Sempre seguint el que posa a la seva pàgina web, han creat la base de dades per diversos motius: el primer és per encoratjar la investigació d'algoritmes; per proveir una base de dades de referència per a treballs de recerca; com una drecera alternativa per crear una base de dades més gran, i per ajudar a començar els nous investigadors. Això ja ens dóna una primera idea que és un projecte que està fet seriosament. Encara que hi ha dos fets més que suporten aquesta teoria:

El primer és que hi ha més d'una pàgina a la xarxa que recomana *Million Song Dataset*.

El segon és que són clients de *The Echo Nest* (<http://the.echonest.com/>), que és una xarxa que acumula dades de cançons per poder fer aplicacions i recerca. Aquests ja tenen més de 35 millions de cançons, de més de 2 milions d'artistes diferents, i tenen com a clients companyies com Twitter, Spotify, MTV, la BBC o VEVO. Per tant, si *Million Song Dataset* ha extret bona part de la informació que ens ofereixen d'aquesta web segurament serà un projecte fiable.

Respecte a la nostra segona condició, si hi ha varietat de gèneres, a la secció de *FAQ (Frequently Asked Questions)* de *Million Song Dataset*, ens diu exactament d'on treuen les cançons de la base de dades. El procés de selecció és complex, però ens diuen que un 48% de les cançons les van extreure mirant quins eren el 100 artistes més populars segons *The Echo Nest*, seguint un camí aleatori entre els artistes enllaçats a partir d'aquests 100 primers, i descarregant les cançons. L'element aleatori de l'algorisme, i el fet que el nombre total d'artistes sigui molt més gran que els 100 primers, ens fa pensar que la representació dels diferents gèneres no és esbiaixada.

Million Song Dataset compleix tots els requisits per ser la base de dades ideal. Només cal descarregar la informació i eliminar les cançons que no

contenien les dades que ens interessaven.

3 Mètode de treball

El contingut de la base de dades, un cop eliminades totes les cançons que no contenien les dades que ens interessaven, que per al projecte inicial és bàsicament la data en què està feta i les durades de les cançons, és d'un total de 515576 cançons. Per tant, queda descartada l'opció de manipular totes aquestes dades amb un Excel, ja que el màxim de files que hi podem posar són 65536, així que vaig optar per l'opció de fer-ho en Python, un llenguatge per programar l'ordinador. Per tant, el primer que vaig haver de fer va ser assignar totes les durades a una llista de Python, i les dates de les cançons a una altra, de manera que la data i quedés a la posició $i - 1$ (a Python es comença nomenant per 0), i que la durada i , corresponent a la cançó amb la data i , a la posició $i - 1$ de la llista de durades. Per tant, quedaria quelcom semblant a això:

```
durades = [durada_1, durada_2, ... , durada_n]
dates = [data_1, data_2, ... , data_n]
```

3.1 Gràfic utilitzant Matplotlib

Quan pensem avui en dia en un telèfon, poca gent se l'imagina sense aplicacions que el complementin. Passa igual amb un llenguatge de programació. Existeixen milers de llibreries, que són programes que estan a Internet que fan alguna funció, i l'usuari només ha de descarregar-los i "cridar-los" en el seu propi programa perquè facin la funció desitjada. Doncs bé, hi ha una llibreria de Python anomenada *Matplotlib* per a graficar dades: només li has d'entrar les dades que vols que surtin en el gràfic, assenyalant quina és la variable dependent i quina la independent, i *Matplotlib* et farà el gràfic. A continuació tenim un exemple del codi necessari per fer un gràfic simple:

```
import matplotlib as plt

llista_x = [1, 2]
llista_y = [4, 8]
```

```
plt.hist(llista_x, llista_y)
plt.show()
```

El que estem fent és: primer importem *Matplotlib*; el `as plt` ens permet de cridar-ho com a `plt` en comptes de com a `matplotlib`, qüestió de comoditat. A continuació definim les llistes que seran els diferents punts que posarem al gràfic. A la `llista_x` els de les x , i a la `llista_y` els de les y . Per tant, els punts que ens quedaran seran $(1, 2)$ i $(4, 8)$. Després cridem *Matplotlib*, però com a `plt`, i li diem que ens faci un histograma amb els integrants d'aquelles dues llistes, i finalment li diem que ens ho ensenyi. Encara ens quedarien ajustar els eixos i les dimensions de mesura, però això ja es qüestió d'estètica i visibilitat, no de funcionalitat.

4 Gràfic de distribució de les durades

El primer que hem de fer és un gràfic de distribució de les durades, per fer-nos una primera idea de com estaran repartides aquestes. Ho farem utilitzant *Matplotlib*, i amb un codi semblant al de l'exemple anterior, encara que amb petites modificacions.

Només li hem posat una llista amb dades, ja que només ens interessen les durades, i el nombre de cançons amb la mateixa durada, que és com ens ho deixarà *Matplotlib* si només li donem una entrada. També li hem de dir com volem que pinti les barres, encara que no hi ha un criteri de decisió per això altre que anar provant. Al final vaig optar per 300 bins.

```
plt.hist(durades, bins = 300)
```

A la segona línia el que fem és posar-li un títol al gràfic. Encara que volem que ens digui el nombre de cançons amb que està fet en el títol. Per fer-ho fem això: `(%d cançons)'' % len(durades)`, el que ens diu aquest primer percentatge és que allà hi va escrita una paraula que li marcarem després, i aquesta, que ens introdueix el segon percentatge, és el `len(durades)`. El `len` ve de la paraula anglesa *length*, és a dir la longitud, per tant voldrà dir que el que hi va allà és la longitud de la llista de cançons, que serà igual al nombre total de cançons.

```
plt.title("Distribucio global (%d cancons)" % len(durades))
```

A la següent línia, al `plt.axis` estem decidint la longitud dels eixos, l'abast d'unitats que li volem donar, els dos primers números són per l'eix x , que anirà de 0 min a 15 min, i els dos últims per l'eix y , de 0 a 35000 cançons. Com decidim aquests valors és qüestió d'anar provant fins que aconseguim un nombre en el qual el gràfic encaixi bé.

```
plt.axis([0, 15, 0 , 35000])
```

A continuació, a la part on posa `plt.xlabel` i `plt.ylabel`, estem posant el títol de l'eix x i l'eix y .

```
plt.xlabel("Durades (minuts)")
plt.ylabel("Nombre de cançons")
```

Finalment li diem que ens ho ensenyi, i que s'aturi a la següent línia. El codi complet i el gràfic queden tal com mostra la figura 1.

```
plt.hist(durades, bins = 300)
plt.title("Distribucio global (%d cançons)" % len(durades))
plt.axis([0, 15, 0 , 35000])
plt.xlabel("Durades (minuts)")
plt.ylabel("Nombre de cançons")
plt.show()
plt.close()
```

La figura 1 ens mostra que hi ha molt poques cançons que durin poc, i a la vegada poques que duren molt. El dibuix que ens queda té una certa semblança a una distribució normal o *gaussiana* encara que no ho és, ja que no acaba de ser simètrica del tot a causa que les cançons de llarga durada se'ns disparen una mica. De fet, el rang és: $[0.083, 50]$, per tant, va de 0.083 min, és a dir 5 s, a 50 min. La mitjana de la durada de les cançons és de 4.11 min.

5 Mitjana i desviació tipus

Ara ja hem vist com estan distribuïdes les durades, i ja tenim una idea general de quines són les més i menys duradores. Ara els següents passos seran fer un gràfic amb la mitjana de les durades de cada any, per tenir una idea general

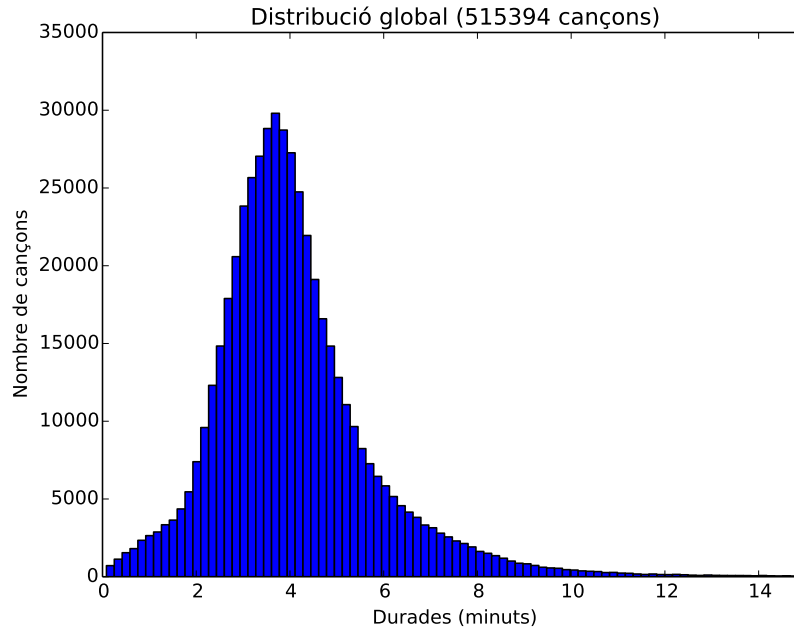


Figura 1: Distribució global

de com han evolucionat en el temps, però aquest s'ha de complementar amb un gràfic de les desviacions tipus de cada any, ja que podria ser que un any donés una mitjana molt alta a causa que hi hagués unes poques cançons que duressin molt. Però abans de fer els gràfics, hem de fer un petit codi en Python que ens faci la mitjana i la desviació tipus.

El que tenim ara mateix és una llista amb totes les durades i una llista amb tots els anys, que són correlatius. Però per fer la mitjana i la desviació tipus necessitem fer un diccionari. Els diccionaris ordenen les seves entrades mitjançant claus, en el nostre cas les claus seran els anys. Cada clau té un contingut que la acompanya, que en el nostre codi serà una llista de totes les durades de les cançons d'aquell any. Per tant, hem de fer un codi que ens ho agrupi tot:

```
def add_to_dic(dic, year, duration):
    if year in dic:
        dic[year].append(duration)
```

```

else:
    dic[year] = [duration]

def build_dic(years, durations):
    dic = {}
    i = 0
    while i < len(years):
        add_to_dic(dic, years[i], durations[i])
        i = i + 1
    return dic

dic = build_dic(anys, durades)

```

Aquí hem fet dues funcions, que és el que s'utilitza en python per fer un algoritme perquè dugui a terme una acció. La primera el que fa és mirar al diccionari, anomenat `dic` si l'any amb el qual ens trobem és ja una entrada del diccionari. Si ho és, afegirà a la llista que estigui relacionada amb aquell any la durada corresponent a l'any que hem anomenat primer. Si l'any no és al diccionari, crearà una entrada amb aquell any, i li relacionarà una llista amb la durada corresponent a aquest.

5.1 Mitjana

Ara ja tenim la informació com ens interessa per a calcular la mitjana,

$$\bar{x} = \frac{\sum x_i n_i}{n}.$$

Ja podem procedir a calcular la mitjana de les durades de cada any. Primer cal fer una funció que ens faci la mitjana dels diferents nombres d'una llista.

```

def average(array):
    total_sum = sum(array)
    if total_sum == 0:
        return None
    return total_sum / float(len(array))

```

El que fa és sumar tots els components de la llista, i dividir-ho pel número de components de la llista.

Ara ens cal fer que faci la mitjana de cada una de les llistes relacionades a cada any del diccionari, i ho farem així:

```
def final_average(years, durations):
    a_years = []
    averages = []
    for e in range(min(years), max(years)):
        if e in dic:
            a_years.append(e)
            averages.append(average(dic[e]))
    return a_years, averages
```

El que fa aquest programa és recórrer totes les entrades del diccionari, per cada entrada crida la funció que ens feia la mitjana, i fa la mitjana de tots els components de la llista relacionada amb aquella entrada. Finalment, agafa tots els anys, i els fica en una llista, `a_years`, i fica les mitjanes de les durades en una llista, també en aquest cas correlativa amb la dels anys.

Finalment ja li podem demanar a `Matplotlib` que ens faci el gràfic, que ens quedarà així:

A la figura 2 veiem que les mitjanes de tots els anys varien entre 2.5 i 3.5 fins el 1970, encara que just en aquell any passa un fet més que curiós, la mitjana de les durades puja de manera notable per situar-se a entre 4 min i 4.5 min i estabilitzar-se allà, arribant a un màxim de 4.3 min. Això podria ser degut a diferents fets de tipus social: podria ser que de sobte hi hagués hagut un canvi de tendències degudes a fets que de moment desconeixem, i que hagin fet augmentar la durada de les cançons; o podria ser la creació d'un nou gènere musical que tingués les cançons molt llargues, tant que ens fes pujar la mitjana d'aquesta manera, i per tant aquest gràfic no seria tan representatiu com sembla a primer cop d'ull.

5.2 Desviació tipus

Per tant, abans de posant-nos a investigar sobre els canvis que va patir la música als anys setanta, cal fer el gràfic de la desviació típica de cada any,

$$\sigma = \frac{\sum (x_i - \bar{x})^2}{n},$$

per veure si podem ja descartar alguna de les teories anteriors.

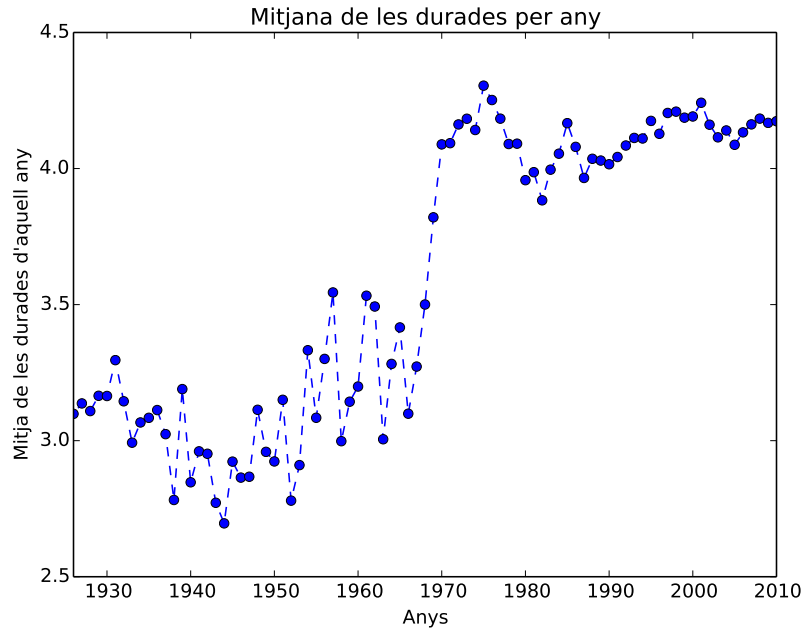


Figura 2: Mitjanes de les durades de cada any

Per fer-lo, podem aprofitar el codi que hem creat anteriorment per aconseguir tenir totes les durades d'un mateix any en una entrada d'un diccionari. Per tant, un cop ja tenim això només hem de crear un programa que ens faci la desviació típica en totes aquestes entrades.

```
def stdev(array):
    media = average(array)
    calc = []
    for e in array:
        c = e - media
        y = pow(c, 2)
        calc.append(y)
    return math.sqrt(sum(calc) / len(array))
```

Aquest programa, per totes les entrades d'una llista, els resta la mitjana aritmètica de la llista, que ja ha estat calculada anteriorment i després les eleva al quadrat. A continuació fa el sumatori de tots els valors que ha

aconseguit en el pas anterior i el resultat d'aquest el divideix pel nombre d'entrades de la llista. Finalment fa l'arrel quadrada del resultat de la divisió, i ja tenim la desviació típica. Encara que pel que volem fer nosaltres, encara no estem.

```
def final_deviation(years, durations):
    s_years = []
    deviations = []
    for e in range(min(years), max(years)):
        if e in dic:
            s_years.append(e)
            deviations.append(stdev(dic[e]))
    return s_years, deviations
```

El que fa aquesta funció és, per cada una de les llistes associades a les entrades al diccionari, fer la desviació típica, i ficar-la a una llista, correlativament a l'any al qual pertany la desviació, que va en una altra llista.

Ara ja sí que tenim el programa fet, i ja li podem dir a `Matplotlib` que ens faci el gràfic. Figura 3

6 Anàlisi dels resultats

Com hem comentat abans, observar cada gràfic de manera individual no té gaire sentit, ja que l'un sense l'altre no està complet. Però si observem la relació entre tots dos podem veure diferents aspectes interessants.

6.1 Primers anys

Considerarem les cançons anteriors als anys anys 60 com les més antigues. El que trobem és que les mitjanes varien entre 2.5 min i 3.5 min, i que les desviacions típiques entre 0 min i 1 min. Aquesta primera dada ens mostra que totes les cançons antigues tenen una mitjana molt semblant, i que a més totes tenen

Una de les hipòtesis per les quals passa això és que totes les cançons d'aquests escassos vint anys tenien una durada semblant, d'aquí les desviacions típiques petites, però que hi havia algunes cançons llargues en alguns dels anys que feien que aquesta mitjana canviés, i que fessin pujar una mica la desviació tipus. Si aquesta hipòtesi fos certa implicaria que si la mitjana és

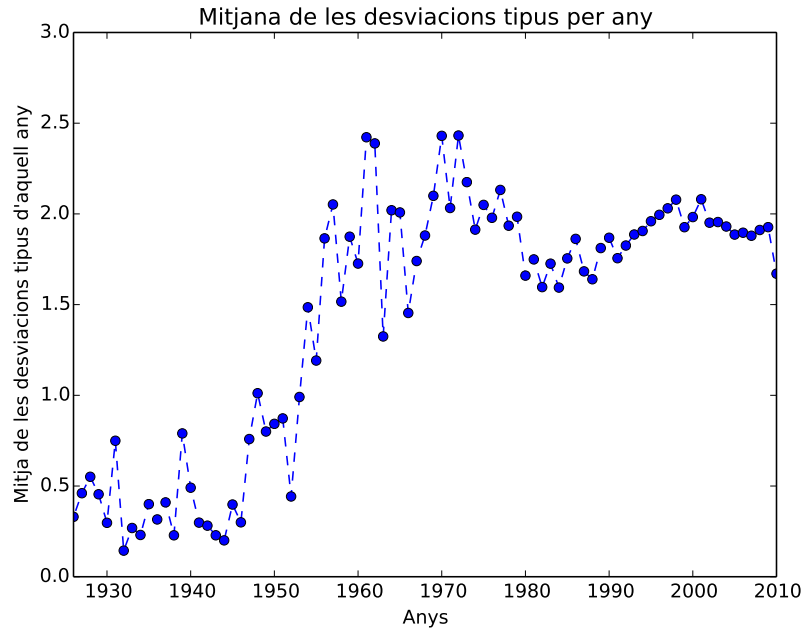


Figura 3: Mitjana de les desviacions tipus de cada any

més gran del que és habitual en aquest període, la desviació tipus també ho hauria de ser, i per tant si pintéssim un gràfic de les desviacions típiques respecte les mitjanes per ordre creixent, el resultat hauria de ser un gràfic creixent.

Un cop feta la figura 4 veiem que no ens queda cap figura definida, i per tant no hi ha una relació directa, almenys en els primers anys, entre la mitjana aritmètica i la desviació tipus.

Un cop descartada aquesta hipòtesi, sobre els primers anys només ens queda remarcar el que és obvi: les cançons són, en general, curtes, i no hi ha molta diferència entre diferents durades de cançons. Les raons per les que les cançons eren curtes són diverses: primer de tot hi havia pressió dels promotors musicals, ja que només deixaven posar cançons de 2.5min o menys. Una altra raó és que els suports on anaven gravades les cançons tenien poca capacitat, per tant no podien fer cançons gaire llargues si volien ficar al disc més d'un parell de cançons.

El que podem fer per acabar de completar aquesta part és agafar un any a l'atzar i fer-ne la distribució general. Escollirem l'any 1955.

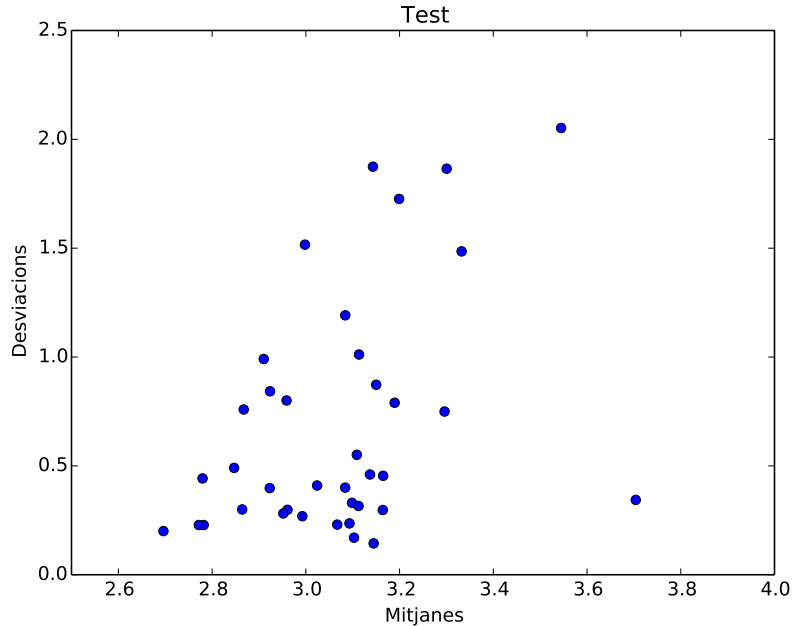


Figura 4: Relació entre les mitjanes i les desviacions dels primers anys

A la figura 5 veiem d'una manera més representativa el que ens diuen els gràfics de les mitjanes aritmètiques (*Figura 2*) i de les desviacions tipus (*Figura 3*). Gariebé totes les cançons es troben entre els 2 min i els 04 min, encara que en tenim moltes més en els dos que en els quatre.

6.2 La transició

Als anys seixanta les desviacions típiques es disparen, fent pujar només una mica les mitjanes. En canvi als setanta la durada general de les cançons augmenta.

Als anys seixanta les desviacions típiques augmenten molt (*Figura 3*), però no ho fan així les mitjanes (*Figura 2*), que puja el seu valor encara que d'una manera no notable. Als setanta, en canvi, les desviacions tipus segueixen més o menys igual que als seixanta, però les mitjanes de les cançons incrementen molt. En aquest punt analitzarem aquest increment, tant de les mitjanes aritmètiques com de les desviacions, primer de manera numèrica, i complementant-la a posteriori amb una hipòtesi històrica.

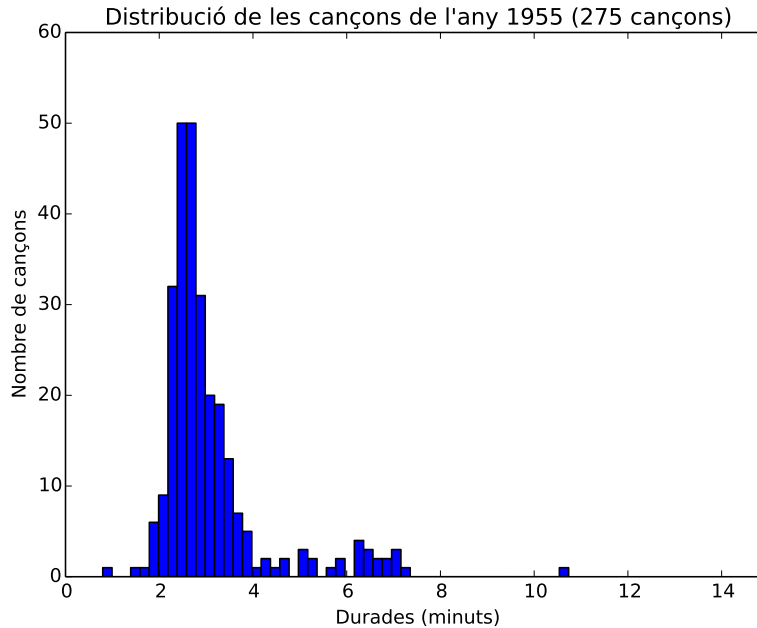


Figura 5: Distribució de les durades al 1955

6.2.1 Anàlisi numèrica

Farem primer de tot un gràfic per ajudar a fer aquesta anàlisi numèrica, que ens servirà també per a la històrica.

A la figura 6 veiem com a l'any 1965 la gran majoria de valors es troben entre els 2 min i els 3 min, cosa que compleix el que veiem al gràfic de les mitjanes aritmètiques (*Figura 2*). Veiem també que a diferència del gràfic de 1955, mostrat anteriorment, hi ha una sèrie de valors extrems que fan variar la desviació tipus, sense fer pujar de manera tan notable la mitjana. En canvi al 1970, any en què comencen a pujar les durades, veiem que la majoria de cançons es concentren entre els 3 min i els 4 min, fet que ens fa pujar les mitjanes.

6.2.2 Anàlisi històrica

El rock simfònic comença a sorgir en aquestes dates, però al principi la producció era mínima ja que no es fa popular fins a la dècada següent. Les

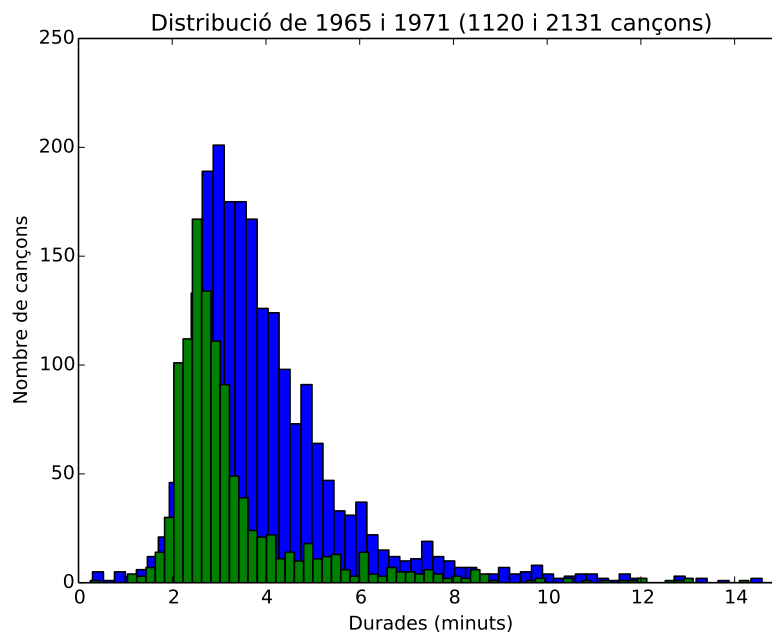


Figura 6: Comparació de les distribucions del 1965 (verd) i 1971 (blau)

cançons de rock simfònic eren molt llargues. Aquesta petita producció de rock simfònic, que pot venir també acompanyat d'altres gèneres amb cançons de llarga durada, ens pot fer variar molt la desviació típica que fins als seixanta era tan petita. A més ens pot canviar la mitjana, que es el que passa a partir dels seixanta. A partir d'això podem treure una hipòtesi del que ens passa.

Abans dels anys setanta hi havia una norma a les ràdios musicals que no podien posar música que durés més de 2.5min. Moltes cançons més duradores tenien una versió més curta per emetre-la a la ràdio, i és que convertir una cançó de 3 min a 2.5 min no és difícil. En canvi convertir una cançó de 7 min, que és el que solien durar com a mínim les cançons en gèneres de tan llarga durada, a 2.5 min és gairebé impossible. Tampoc hi havia cap altra manera de fer la teva música popular que fos senzilla i gratuïta, ja que la manca d'Internet i televisió ho feia impossible. El rock simfònic es va començar a fer popular a la dècada dels setanta, coincidint amb la suspensió de la norma.

La derogació de *la norma dels dos minuts i mig* va comportar que no hi hagués una durada màxima de les cançons que hi sonaven. Això ens porta a

una sèrie de fets.

Primer de tot la popularització del rock simfònic, amb grups com *Pink Floyd* que es van acabar guanyant una gran popularitat. Això ajuda a pujar el valor de les mitjanes aritmètiques (*Figura 2*) de cada any, ja que com més cançons llargues més gran serà la mitjana.

Aquesta teoria desemboca en una pregunta: si el nombre de cançons duradores puja, perquè no hi ha un canvi radical del valor de les desviacions típiques consecutivament? Com que es va derogar la *norma de la ràdio*, els artistes tenien molta més llibertat per poder-se expressar i alhora fer-se coneguts mitjançant la ràdio. Aquest fet comporta una pujada general de la durada de les cançons. Això ens explica l'augment de les mitjanes aritmètiques, i que no pugi la desviació típica (*Figura 3*) de la manera que ho fan aquestes, ja que la pujada general de les durades contrasta amb les dels estils de música amb durades extremadament grans que en aquest període estan en auge.

Si donem un cop d'ull al gràfic de distribucions que hem fet anteriorment (*Figura 3*), veiem que totes les dates històriques que hem donat aquí en aquesta secció coincideixen amb el gràfic. Per tant no podem refutar aquesta hipòtesi.

6.3 Darrers anys

A partir del 1980 les mitjanes de les durades s'estabilitzen i varien entre 4 min i 4.5 min la majoria. Les desviacions típiques baixen una mica respecte a la dècada anterior, encara que segueixen més altes que als seixanta. Això és degut a una estabilització de les cançons de llarga durada.

Si mirem el gràfic de les mitjanes i de les desviacions tipus dels últims deu anys veiem que gairebé no varien. Això implica que gairebé totes les cançons duraven entre 6 min i 2 min, ja que la mitjana és 4 i la desviació tipus és 2. Per tant, que totes les cançons estiguin en una franja tan marcada en els últims deu anys és un fet curiós.

Si fem un gràfic de la distribució de les cançons dels últims deu anys, per exemple del 2005, veurem que el que ens diu el gràfic ens compleix.

A la figura 7 veiem que la majoria de valors es concentren entre els 2 min i els 6 min, mentre que la moda és 4. Per tant, els gràfics de les desviacions típiques i de les mitjanes ens demostren que tots els gràfics d'aquests deu anys tindran una estructura semblant a la d'aquest.

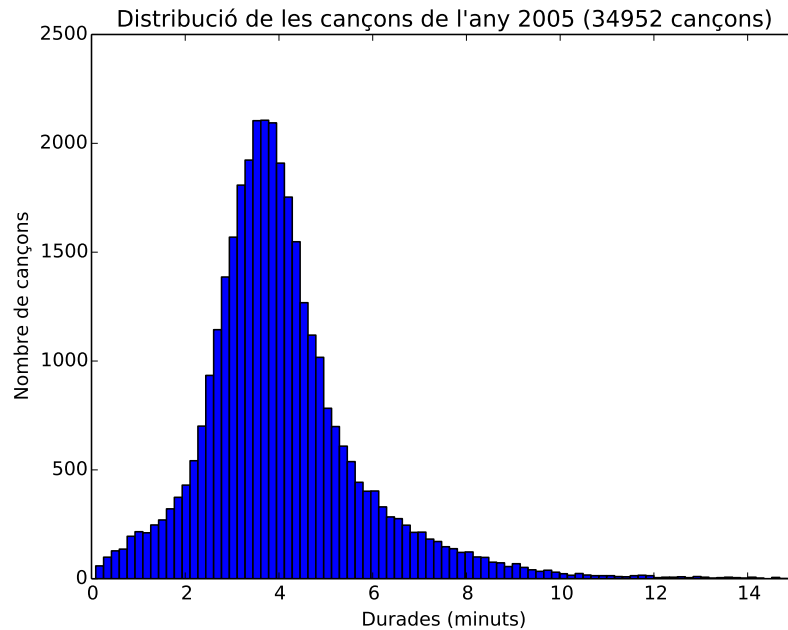


Figura 7: Distribució de les durades al 2005

Per tant, en aquest punt hem vist que les cançons modernes, considerant modern des del 1980 cap endavant, es poden dividir en dos grups, segons la seva desviació tipus: abans i després de l'any 2000. Hem vist un gràfic posterior al 2000, per tant ara el compararem amb un anterior al 2000, per exemple 1990.

Veiem a la figura 8 que gairebé no canvia respecte a la figura 7, hi ha petites variacions, però la forma que formen les barres del gràfic és molt semblant. Veiem que al gràfic de les desviacions tipus (*Figura 3*) hi ha una petita variació entre el 1990 i el 2005, però que quan ho passem a una distribució, els valors gairebé no canvien. Per tant, podem afirmar que no només les cançons dels últims deu anys tenen unes distribucions molt semblants. Sinó que ho fan les dels últims trenta, és a dir, des dels vuitanta fins a l'actualitat.

Aquí podríem trobar una altra hipòtesi històrica: als anys vuitanta es crea la primera cadena de televisió dedicada exclusivament a la música. A partir d'aquest moment es comença a crear una xarxa, tant a la televisió com a internet uns anys més tard, que permetrà a l'artista ser conegut de

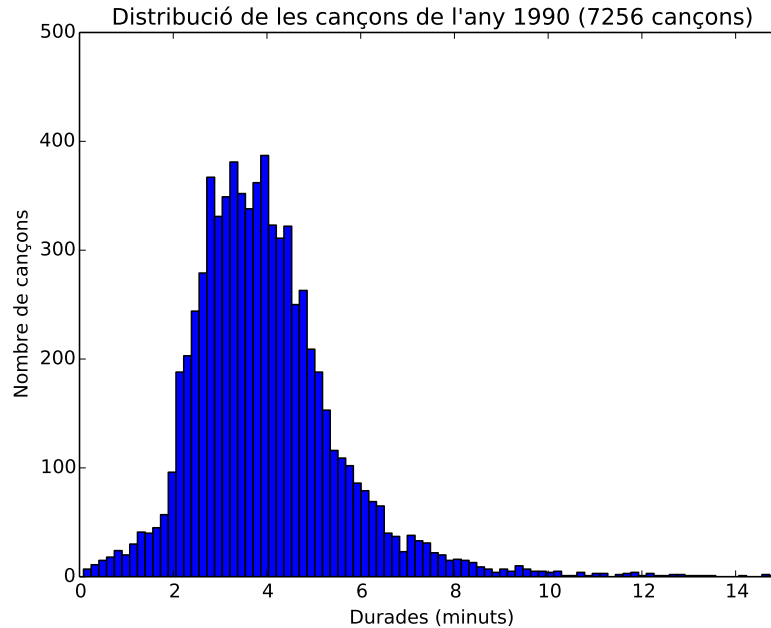


Figura 8: Distribució de les durades al 1990

moltes més maneres. Això ja donarà total llibertat als artistes de fer música de diferents durades i que totes siguin acceptades pels mitjans. Seguirant tenint el lideratge les cançons d'entre 3.5 min i 4 min, encara que hi haurà moltes cançons de valors més extrems, tan curtes com llargues. Finalment, aquesta estructura es mantindrà fins a l'actualitat.

7 Conclusions

La hipòtesi que hem presentat a l'inici del treball era certa: les durades de les cançons augmenten de manera general a mida que avancen els anys. Hem vist primer un gràfic de les distribucions que ja ens deia que hi havia moltes més cançons de 3min i 4min que de 2min, per tant ja ens donava una primera idea del que podia acabar passant.

A continuació el gràfic de les mitjanes de les durades de cada any ens ho deia ven clar: les durades de les cançons puja a mida que ens acostem a l'actualitat. No ho fa, però, de la manera que pensàvem que ho faria. No

ens esperàvem unes durades tant igualades a les dates més antigues i més modernes, amb una pujada tant significativa entre les dues, la anomenada transició. Després el gràfic de les desviacions tipus complementava al de les mitjanes. Aquest també té una pujada molt gran, però al contrari del de les mitjanes, que augmenta als setanta, aquest ho feia als seixanta. Per tant, aquí s'intuïa que una sèrie de moviments es començaven a gestar.

Després de fer una recerca important vam aconseguir trobar una explicació històrica per la qual passaven tots aquests fets. L'aparició del rock simfònic al 1960 i la seva popularització al 1970, i la derogació de *la norma dels dos minuts i mig*, també al 1970, en són la clau.

A partir d'aquí dividim les nostres dates en tres parts: les més antigues, que van des de l'inici fins als seixanta; dels seixanta als vuitanta, a les que anomenem la transició a causa del canvi, i finalment les modernes, dels vuitanta cap endavant. Comencem a fer una sèrie de gràfics centrant-nos en diferents anys, tant com per veure d'una manera més visual que està passant a cada un dels períodes, com per defensar la nostra teoria. Tota la informació que ens donen els gràfics li aporten arguments a favor, de manera que l'acabem donant per bona.

Així doncs, en aquestes escasses vint pàgines hem vist com evoluciona la música, hem vist que la nostra hipòtesi inicial es complia, i hem aconseguit extreure diferents gràfics i dades per treure unes conclusions sòlides.

A Codi

```
# -*- coding: utf-8 -*-
import math
import trackdb
import matplotlib.pyplot as plt
meta = trackdb.read('track-meta.db', ['year', 'duration'])
nl_years = list(meta['year'])
nl_durations = list(meta['duration'])

# Popping duratins < 5'' #

years = []
durations = []

def pop_durations():
    c = 0
    while c < len(nl_durations):
        if nl_durations[c] >= 5:
            durations.append(nl_durations[c])
            years.append(nl_years[c])
        c = c + 1
    return durations

pop_durations()

# CONVERTING DURATIONS FROM SECONDS TO MINUTES #

def minutes(durations):
    i = 0
    while i < len(durations):
        durations[i] = durations[i] / 60
        i = i + 1
    return durations
```

```

durations = minutes(durations)

# GENERAL GRAPHIC #

# graphic of the distribution of all the durations

plt.hist(durations, bins = 300)
plt.title(u"Distribucio global (%d cancons)" % len(durations))
plt.axis([0, 15, 0, 35000])
plt.xlabel(u"Durades (minuts)")
plt.ylabel(u"Nombre de cancons")
plt.show()
plt.savefig('general.pdf')
plt.close()

# AVERAGE #

# calculating the average of...

def average(array):
    total_sum = sum(array)
    if total_sum == 0:
        return None
    return total_sum / float(len(array))

# STANDARD DEVIATION #

# function for the standart deviation of...

def stdev(array):
    media = average(array)
    calc = []
    for e in array:

```

```

        c = e - media
        y = pow(c, 2)
        calc.append(y)
    almostfinished = sum(calc) / len(array)
    return math.sqrt(almostfinished)

```

#AVERAGE OF THE DURATIONS PER YEARS #

*# looking to the general list, and creating
#a dictionary of all the years, and the durations
#related to them*

```

def add_to_dic(dic, year, duration):
    if year in dic:
        dic[year].append(duration)
    else:
        dic[year] = [duration]

def build_dic(years, durations):
    dic = {}
    i = 0
    while i < len(years):
        add_to_dic(dic, years[i], durations[i])
        i = i + 1
    return dic

```

```

build_dic(years, durations)

```

```

dic = build_dic(years, durations)

```

calculating the average of all the duratons related to its year

```

def final_average(years, durations):

```



```

a_years = []
averages = []
for e in range(min(years), max(years)):
    if e in dic:
        a_years.append(e)
        averages.append(average(dic[e]))
return a_years, averages

#print final_average(years, durations)

# GRAPHIC OF THE AVERAGE OF THE DURATIONS PER YEARS #

a_years, averages = final_average(years, durations)

plt.plot(a_years, averages, linestyle='--', marker='o', color='b')
plt.axis([1926, 2010, 2.5, 4.5])
plt.title(u"Mitjana de les durades per any")
plt.xlabel(u"Anys")
plt.ylabel(u"Mitja de les durades d'aquell any")
plt.show()
plt.savefig('averages.pdf')
plt.close()

# GRAPHIC OF THE STANDART DEVIATION PER YEARS #

def final_deviation(years, durations):
    s_years = []
    deviations = []
    for e in range(min(years), max(years)):
        if e in dic:
            s_years.append(e)

```

```

        deviations.append(stdev(dic[e]))
    return s_years, deviations

#print final_average(years, durations)

s_years, deviations = final_deviation(years, durations)

plt.plot(s_years, deviations, linestyle='--', marker='o', color='b')
plt.axis([1926, 2010, 0, 3])
plt.title(u"Mitjana de les desviacions tipus per any")
plt.xlabel(u"Anys")
plt.ylabel(u"Mitja de les desviacions tipus d'aquell any")
plt.savefig('deviation.pdf')
plt.show()

# GRAPHIC OF THE DISTRIBUTION OF THE DURATION OF THE SONGS PER YEAR #

def search(year, dic):
    return dic[year]

fiveties = search(1955, dic)
#print fiveties

sixties = search(1965, dic)
#print sixties

seventies = search(1971, dic)
#print seventies

nineties = search(1990, dic)
#print nineties

```

```
twothousandfive = search(2005, dic)
```

```
# 1955's distribution
```

```
plt.hist(fiveties, bins = 50)
plt.axis([0, 15, 0 , 60])
plt.title(u"Distribucio de les
cancons de l'any %d (%d cancons)" % (1955, len(fiveties)))
plt.xlabel(u"Durades (minuts)")
plt.ylabel(u"Nombre de cancons")
plt.show()
plt.savefig('1955.pdf')
plt.close()
```

```
# 1965 vs 1971 distribution
```

```
plt.hist(seventies, bins = 100)
plt.hist(sixties, bins = 100)
plt.axis([0, 15, 0 , 250])
plt.title(u"Distribucio de 1965 i 1971
(%d i %d cancons)" % (len(sixties), len(seventies)))
plt.xlabel(u"Durades (minuts)")
plt.ylabel(u"Nombre de cancons")
plt.show()
plt.savefig('1965-71.pdf')
plt.close()
```

```
# 2005 distribution
```

```
plt.hist(twothousandfive, bins = 300)
plt.axis([0, 15, 0 , 2500])
plt.title(u"Distribucio de les cancons de l'any %d (%d cancons)"
% (2005, len(twothousandfive)))
plt.xlabel(u"Durades (minuts)")
plt.ylabel(u"Nombre de cancons")
```

```

plt.show()
plt.savefig('2005.pdf')
plt.close()

# 1990 distribution

plt.hist(nineties, bins = 100)
plt.axis([0, 15, 0, 3000])
plt.title(u"Distribucio de les cançons de l'any %d
(%d cançons)" % (1990, len(nineties)))
plt.xlabel(u"Durades (minuts)")
plt.ylabel(u"Nombre de cançons")
plt.show()
plt.savefig('1990.pdf')
plt.close()

```

```

# RANGE #

```

```

#print min(durations)
#print max(durations)

```

```

#----- TEST -----

```

```

#a_years
#averages

```

```

def test():
    a = 0
    test_years = []
    test_averages = []

```

```

test_deviations = []
for e in range(a_years[0], a_years[37]):
    test_years.append(e)
    test_averages.append(averages[a])
    test_deviations.append(deviations[a])
    a = a + 1
return test_deviations, test_averages

td, ta = test()

plt.plot(ta, td, 'ro', marker='o', color='b')
plt.axis([2.5, 4, 0, 2.5])
plt.title(u"Test")
plt.xlabel(u"Mitjanes")
plt.ylabel(u"Desviacions")
#plt.show()
plt.savefig('test.pdf')
plt.close()

```