From Code to Composition: Personalized Recommendation and Product Innovation*

David Argente Yale University
Maria Trejos Yale University
Diana Van Patten Yale University
Kai Hao Yang Yale University

March 2026

Abstract

This paper studies how personalized recommendation algorithms (PRAs), a central application of artificial intelligence, shape product innovation and differentiation. We develop an equilibrium model in which consumers have heterogeneous tastes, producers choose product characteristics subject to costs, and a platform recommends products while learning preferences from noisy signals. The model highlights a key mechanism: while personalization narrows the set of products each consumer is exposed to, it induces producers to differentiate across consumer segments. As a result, PRAs increase the dispersion of product characteristics in the market as a whole and foster innovation by making new products more distinct from the existing stock. We test these predictions in the music industry, where PRAs have been widely adopted since the mid-2010s. Using tools from music information retrieval applied to raw audio files, we construct measures of similarity across songs and track their evolution over time. We find that songs have become less similar, both within a given year and relative to past releases, indicating an increase in cross-sectional differentiation and intertemporal innovation. Our results suggest that recommendation algorithms expand market-wide variety and increase innovation by reshaping producers' incentives.

JEL Classification Numbers: O33, D83
Keywords: Artificial Intelligence, Recommendation Algorithms, Innovation, Music

*We thank Fernando Alvarez, Pete Klenow, Oleg Itskhoki, Alex Monge, and Chris Tonetti for useful comments.

1 Introduction

Artificial intelligence increasingly shapes how consumers encounter products. Personalized recommendation algorithms (PRAs) are now central to consumption in digital markets, including music, e-commerce, news, social media, finance, and healthcare. By filtering vast product spaces and tailoring exposure to individual preferences, these systems play a growing role not only in matching consumers to existing goods, but also in shaping incentives for what gets produced.

This raises a fundamental question: do recommendation algorithms expand or compress the space of products that are created? On the one hand, by amplifying content that performs well with targeted audiences, algorithms may encourage imitation and convergence. On the other hand, by segmenting consumers and revealing heterogeneity in demand, they may induce producers to differentiate more strongly across market niches.

This paper studies how PRAs affect product innovation and differentiation. We argue that understanding this question requires an equilibrium perspective. Recommendation systems do not operate in a fixed product space. By changing which consumers see which products, they alter the demand environment faced by producers, who then adjust product characteristics in response. We therefore develop a model of consumers, producers, and a recommendation platform. Consumers have heterogeneous preferences over product characteristics. Producers earn payoffs from consumer engagement and choose product characteristics optimally, given their ability, their beliefs about preferences, and the recommendation policy of the platform. The platform determines which products are recommended to which consumers and gradually learns about consumer preferences from noisy signals generated by past choices.

A central feature of the platform's problem is the trade-off between exploitation and exploration. In practice, PRAs are designed to maximize consumer engagement by learning individual preferences and tailoring exposure accordingly. To do so, platforms balance two distinct objectives. On the one hand, they exploit existing knowledge by recommending products a user is already likely to enjoy. On the other hand, they explore by occasionally recommending products that reveal more about consumer tastes. In this environment, recommendation policies shape not only the matching between consumers and products, but also the endogenous distribution of product characteristics supplied to the market.

We find that PRAs widen the dispersion of product characteristics in the market as a whole, even as they reduce the variety that each consumer is exposed to when supply is fixed. The key insight is that equilibrium supply responses reverse the partial equilibrium intuition. Holding supply fixed, personalization narrows exposure, so one might expect recommenda-

tion algorithms to reduce variety. Once producers respond, however, the logic changes. In equilibrium, producers must be indifferent across the consumers they target. If targeting one consumer segment were strictly more profitable, producers would crowd into that segment until congestion lowered its attractiveness. This no-arbitrage condition implies that recommendation policies expand the range of characteristics supplied in the market, generating more diversity across products even as individual exposure narrows. PRAs can therefore be welfare enhancing because they improve the matching between products and consumers and reduce misallocation. By contrast, the opposite result can arise in a partial equilibrium model with no endogenous supply response.

We formalize this mechanism and characterize equilibrium outcomes under three benchmark regimes: no personalization, full exploitation, and full exploration. The model predicts that recommendation algorithms increase the cross-sectional dispersion of product characteristics. It also predicts an increase in intertemporal originality, meaning that new products become more distinct from what has been produced in the past. In our setting, PRAs do not simply make products more different from one another within a given period. They also shift the direction of innovation over time.

The intuition is straightforward. Intertemporal originality is high when new products are both more diverse among themselves and different from the existing stock. Personalized recommendation contributes to both margins. By segmenting consumers and revealing differences in tastes, it encourages producers to spread out across a wider range of characteristics. At the same time, it makes it profitable to move away from previously common designs and target underserved niches. As a result, recommendation algorithms increase not only within-period differentiation but also the extent to which new products explore new parts of the song characteristic space.

To test our theory, we focus on the music industry, where PRAs have been adopted on a large scale since the mid-2010s, most notably through platforms such as Spotify. Music provides a direct record of creative output, and recent advances in Music Information Retrieval (MIR) and music engineering make it possible to analyze musical content using high-dimensional song features and summarize that information with measures of originality.

Using state-of-the-art MIR methods applied to songs across several genres before and after the adoption of PRAs, we find that songs have become less similar to one another. The distribution of originality shifts rightward, indicating greater dispersion across the market. This empirical pattern is consistent with the model's prediction that recommendation algorithms encourage producers to differentiate their output, with heterogeneity in consumer preferences generating greater creative variation across genres such as pop, rock, and country.

Our empirical results show that songs have become significantly more differentiated over

time. Cross-sectional originality increases sharply after the mid-2010s, coinciding with the widespread adoption of recommendation algorithms. Pairwise originality rises, indicating that songs are less similar to their contemporaries. We also document an increase in intertemporal originality: recent songs are more distinct from the stock of previously released music. These patterns are robust to controls for genre, song structure, and duration, and they are not explained by changes in concentration, prices, or demand trends.

Taken together, the theoretical and empirical results point to a common conclusion. Recommendation algorithms expand the range of products supplied in the market. By revealing heterogeneity in demand and segmenting consumers, they strengthen incentives for differentiation and innovation.

Literature Review This paper is related to several strands of work on recommendation systems, innovation, and music platforms. A large literature studies how recommender systems affect consumption diversity, meaning the range of products consumers discover or consume, or the concentration of sales across a fixed catalog. Fleder and Hosanagar (2009) show that recommender systems can reduce aggregate sales diversity even when they help individual consumers discover products they would not otherwise find. Using a randomized field experiment for a large online retailer, Lee and Hosanagar (2019) show that recommendation can increase individual-level exploration while reducing aggregate sales diversity. Closer to our setting, Anderson et al. (2020) show that algorithmically driven listening on Spotify is less diverse than user-driven listening, and Holtz et al. (2020) show in a Spotify field experiment that personalization raises engagement, lowers individual-level diversity, and increases aggregate diversity. This literature focuses on consumption outcomes and takes the supply of products as fixed. Our model captures features consistent with these findings when the set of songs is held fixed. However, we allow the supply of songs to be endogenous. In our setting, personalized recommendation changes what producers choose to create and how feedback from demand to supply affects differentiation and innovation in equilibrium.

The closest paper to ours is Schnidman (2025), which develops a structural model of the recorded music industry with consumer choice, platform recommendations, and rightsholder release decisions. That paper studies how recommendation affects entry, song characteristics such as length, margins, and welfare. Our paper is complementary but distinct. In that framework, the recommender maps observed song and consumer characteristics into recommendation probabilities, but the platform does not learn consumer tastes over time. As a result, the model does not feature an explicit exploration-exploitation trade-off, which is central to how PRAs operate in practice. Our model builds these forces directly into the theory. The platform and producers learn from noisy feedback, beliefs evolve over time through a

Kalman filter, and equilibrium can be compared across no personalization, full exploitation, and full exploration. This structure yields tractable definitions of cross-sectional and intertemporal originality that can be taken to the data. Empirically, we test these predictions using raw audio files across years and genres, rather than relying on a limited set of observed song attributes.

Our paper is also related to work on directed innovation. Acemoglu (2002) shows that technical change is directed toward more profitable uses, with the direction of innovation shaped by prices and market size, while Acemoglu and Linn (2004) show empirically that larger potential markets induce more innovation. More recently, Agrawal et al. (2018) and Gans (2019) emphasize that advances in artificial intelligence operate primarily through improvements in prediction, which reshape economic decision-making. In our setting, personalized recommendation algorithms improve predictions of consumer preferences and determine which products are exposed to which consumers. This changes the effective profitability of different consumer segments and, in turn, the return to designing products for different parts of the characteristic space. As a result, PRAs affect not only matching but also the direction of product innovation.

The paper is also related to the economics of music, digitization, and platforms. Waldfogel (2017) argues that digitization expanded the supply of cultural goods by lowering the cost of bringing new products to market. Aguiar and Waldfogel (2018) develop a structural model of demand and entry in recorded music in which consumers learn about product quality through consumption. They show that when quality is uncertain at the time of release, experimentation increases the welfare gains from new products. Benner and Waldfogel (2016) show that technological change altered release strategies in recorded music, with major labels relying more heavily on proven artists while independent labels released more music. Datta et al. (2018) show that adoption of streaming increases music consumption and discovery, and Aguiar and Waldfogel (2021) show that Spotify playlist placement has large causal effects on song success. These papers establish why streaming platforms matter for discovery, promotion, and entry. Taking this environment as given, our paper studies how personalized recommendation affects innovation and differentiation in the songs that are produced. To the best of our knowledge, this paper is the first to study how personalized recommendation algorithms shape the direction of product innovation and the extent to which products become more differentiated and more novel both within and across time.

Finally, our empirical strategy builds on the music information retrieval literature. Bello (2011) develops a measure of structural similarity in music based on recurrence structure. Cilibrasi and Vitányi (2005) provide the compression-based distance that underlies our pairwise similarity measure. Goto (2006) develops a method for chorus detection, and Stöter

et al. (2019) develop Open-Unmix for music source separation. We intentionally do not innovate on the measurement side. Instead, we rely on well-established MIR methods developed by experts in the field. Our contribution is to combine these tools at scale, across years and genres, to construct measures of originality from the audio itself, and to complement them with controls for musical structure such as chorus timing and vocal onset. This yields measures that are comparable across songs and suitable for econometric analysis.

2 Model

Overview Time $t \in \mathbb{N} \cup \{0\}$ is discrete. There are $N \in \mathbb{N}$ consumers in every period. In each period t, a unit mass of differentiated and short-lived producers arrive and each produces a song. A platform recommends songs to consumers, who then randomly select songs to listen to.

Consumers Consumers have differentiated preferences over songs. Each song is described by a vector $s \in \mathbb{R}^K$ of attributes, where $K \in \mathbb{N}$ is the number of possible attributes. Consumer i's utility for consuming song $s \in \mathbb{R}^K$ is

$$\langle \theta_i, s \rangle = \sum_{k=1}^{K} \theta_{i,k} s_k \,,$$

where $\theta_i \in \mathbb{R}^K$ is consumer i's type. Suppose that $\{\theta_i\}_{i=1}^N$ is i.i.d. drawn from a multivariate normal distribution with mean μ_i and variance-covariance matrix Σ_i.

Producers In each period, a unit mass of short-lived producers arrives. Producers are differentiated in their ability. The distribution of producers' abilities is given by a CDF F, which admits a density f and is supported on $\mathbb{R}_+$. A producer with ability $z \in \mathbb{R}_+$ can produce any song $s \in \mathbb{R}^K$ at a cost $z \cdot C(s)$, where $C(s)$ is a convex function. In most of the analysis, we assume that $C(s) = \|s\|^2/2$. For each time a song s is listened to by a consumer with type θ_i, the producer of that song obtains a payoff of

$$\phi(\langle \theta_i, s \rangle) \,,$$

where $\phi : \mathbb{R} \to \mathbb{R}$ is an increasing function. For most of the analysis, we will assume that $\phi(z) = z$, or $\phi(z) = \mathbf{1}\{z \geq 0\}$.

Recommendation Mechanism A recommendation mechanism is a mapping $R : \mathbb{R}^K \to 2^{\mathcal{N}}$, which specifies, for any song $s \in \mathbb{R}^K$, the set of consumers $R(s) \subseteq \mathcal{N}$ to whom song s is recommended. Given a recommendation mechanism R, let

$$S_i^R := \{s \in \mathbb{R}^K | i \in R(s)\}$$

be the set of possible songs that will be recommended to consumer i. Given the set of recommended songs, each consumer randomly listen to them. Specifically, if the mass of songs a consumer is recommended to is m, they listen to each song $Q(m)$ times where $Q : \mathbb{R}_+ \to \mathbb{R}_+$ is a decreasing function $\lambda > 0$. That is, if a consumer is recommended to many songs (i.e., m is large), the listen to each recommended song very few times. In contrast, if a consumer is recommended to only a few songs (i.e., m is small), they listen to each song many times. For most of the analysis, we assume that $Q(m) = \lambda/m$ for some constant λ.[1] In our analyses, we consider three main recommendation mechanisms. The *no recommendation* mechanism R^{random} assigns all songs to everyone: $R^{\text{random}}(s) = \mathcal{N}$. The *full exploitation mechanism* R^{exploit} recommends each consumer the best songs given the platform's belief (see more details below). The *full exploration mechanism* R^{explore} recommends each consumer the songs that maximizes learning (see details below).

Equilibrium In each period, given a recommendation mechanism R, and $Q = (Q_i)_{i=1}^N \subseteq \mathbb{R}^N$, where Q_i denotes the quantity consumer i listens to a particular song they are recommended to. Song producer with ability z solves

$$\max_{s \in \mathbb{R}^K} \sum_{i=1}^{N} Q_i \mathbf{1}\{s \in S_i^R\} \mathbb{E}[\phi(\langle s, \theta_i \rangle) \mid \mathcal{F}] - zC(s)\,, \tag{1}$$

where $\mathcal{F}$ is the information set in that given period.

The (mixed) strategy of a producer (randomly) selects a song given (i) their type z; (ii) the vector Q of quantities that each consumer listens to a given song they are recommend to; and (iii) the information set. That is, a (mixed) strategy of a producer is a transition probability $\sigma : \mathbb{R} \times [0,1]^N \to \Delta(\mathbb{R}^K)$, where $\sigma(\cdot|z, Q)$ is the distribution of songs a producer chooses to produce, when their type is z, given Q.

An equilibrium is given by a vector $Q^* = (Q_i^*)_{i=1}^N \in \mathbb{R}^N$ and a strategy σ^* such that producers choose songs optimally according to σ^*, and that the resulting songs, together with the recommendation rule, are consistent with Q^*. That is, given the information set

[1] This is equivalent to assuming that consumers uses the platform for a fixed amount of time (depicted by λ), and randomly selects songs to listen to while using the platform.

and the recommendation mechanism, for all z and for all $s \in \text{supp}(\sigma^*(z, Q^*))$, s solves equation (1), and that

$$Q_i^* = Q\left(\int_0^\infty \sigma^*(S_i^R|z, Q^*)F(\mathrm{d}z)\right).$$

for all i.

Learning and Dynamics Given the set of songs a consumer θ_i listened to, let $\overline{s}$ be its average. The platform and the producer observe a signal

$$x_i = \langle \theta_i, \overline{s} \rangle + \varepsilon_i \,,$$

where $\varepsilon_i \sim N(0, \sigma_\varepsilon^2)$ is an independent noise. Therefore, after a set of songs is chosen by consumer i, the belief about θ_i evolves and becomes a multivariate normal with mean and variance:

$$\begin{aligned} \mu_i^{\text{plat}}(x_i; \overline{s}) &:= \mathbb{E}[\theta_i | x_i, \overline{s}] = \mu_i + M_i(\overline{s})(x_i - \overline{s}^\intercal \mu_i) \\ \Sigma_i^{\text{plat}}(x_i; \overline{s}) &:= \text{Var}(\theta_i | x_i, \overline{s}) = \Sigma_i - M_i(\overline{s})\overline{s}^\intercal \Sigma_i \,, \end{aligned} \tag{2}$$

where $M_i(\overline{s}) := \Sigma_i \overline{s}(\overline{s}^\intercal \Sigma_i \overline{s} + \sigma_\varepsilon^2)^{-1}$ is the variance correction term from the Kalman filter.

Given the platform's signals $\{x_i\}_{i=1}^N$. The producers then observe a noisy signal y_i of x_i for each i, where

$$y_i = x_i + \eta_i \,,$$

and $\eta_i \sim N(0, \sigma_\eta^2)$ is an independent noise. The producer's belief after observing signal realizations $\{y_i\}_{i=1}^N$ and average song $\overline{s}$ is then given by:

$$\begin{aligned} \mu_i^{\text{prod}}(y_i; \overline{s}) :=& \mathbb{E}[\theta_i | y_i; \overline{s}] = \mu_i + N_i(\overline{s})(y_i - \overline{s}^\intercal \mu_i) \\ \Sigma_i^{\text{prod}}(y_i; \overline{s}) :=& \text{Var}(\theta_i | y_i; \overline{s}) = \Sigma_i - N_i(\overline{s})\overline{s}^\intercal \Sigma_i \,, \end{aligned} \tag{3}$$

where $N_i(\overline{s}) := \Sigma_i \overline{s}(\overline{s}^\intercal \Sigma_i \overline{s} + \sigma_\varepsilon^2 + \sigma_\eta^2)^{-1}$ is the variance correction term from the Kalman filter.

Thus, given that a consumer listened to a set of songs that average to $\overline{s}$ and the signal for the producers is y_i, $\theta_i | y_i, \overline{s} \sim N(\mu_i^{\text{prod}}(y_i; \overline{s}), \Sigma_i^{\text{prod}}(y_i; \overline{s}))$, and hence for any song s, consumer i's payoff $\langle \theta_i, s \rangle$ from a producer's perspective is a normal random variable, with

$$\langle \theta_i, s \rangle | y_i, \overline{s} \sim N(s^\intercal \mu_i^{\text{prod}}(y_i; \overline{s}), s^\intercal \Sigma_i^{\text{prod}}(y_i; \overline{s}) s) \,.$$

The above updating rule determines filtrations $(\mathcal{F}_t, \mathcal{G}_t)_{t \geq 0}$, where $\mathcal{F}_t$ denotes the producer's information set in period t and $\mathcal{G}_t$ denotes the platform's information set in period t: In period 0, the initial information sets $(\mathcal{F}_0.\mathcal{G}_0)$ are the null set. Recursively, in period $t \geq 0$, given the information sets $(\mathcal{F}_{t-1}, \mathcal{G}_{t-1})$ in period $t-1$, signals $x_{i,t}$ about consumers

types $(\theta_i)_{i=1}^N$ are generated based the average songs each consumer listened to in period $t-1$, which updates the platform's information set to $\mathcal{G}_t$ via equation (2). Likewise, signal $y_{i,t}$, which is a garbling of $x_{i,t}$ is generated, and the producers update their information sets to $\mathcal{F}_t \subseteq \mathcal{G}_t$ via equation (3). Given information sets $(\mathcal{F}_t, \mathcal{G}_t)$, an equilibrium (Q_t^*, σ_t^*) is determined. Each producer z produces a song $s \sim \sigma_t^*(z, Q_t^*)$. The recommendation mechanism then recommend songs to consumers, based on information $\mathcal{G}_t$. Consumers then randomly select songs to listen to. Since all signals are linear Gaussian, the information sets $(\mathcal{F}_t, \mathcal{G}_t)$ can be summarized by the producer's posterior mean and variances, denoted by $(\mu_{i,t}, \Sigma_{i,t})_{i=1}^N$ and the platform's posterior mean and variances, denoted by $(\nu_{i,t}, \Lambda_{i,t})_{i=1}^N$.

Song Distributions and Similarity Given histories and the filtration $(\mathcal{F}_t, \mathcal{G}_t)_{t\geq 0}$. Let $(s_t^*)_{t\geq 0}$ be the stochastic process that describes the equilibrium songs. Specifically, for each t, conditional on $\mathcal{F}_t$, $s_t^* \sim \sigma_t^*(z, Q_t^*)$. Let Γ^* be the distribution of the process $(s_t^*)_{t\geq 0}$, and let γ_t^* be the marginal distribution of s_t^*. For each $t \geq 0$. Given a song distribution Γ^*, we may define the *originality* of songs by how different the songs are from each other. We consider two notions of originality: *cross-sectional originality* and *intertemporal originality*. Cross-section originality measure how original songs are within each period, whereas intertemporal originality measures how original songs in each period are compared to songs in the past.

Specifically, for any two songs $s, s' \in \mathbb{R}^K$, we measure the distance between the two songs by the squared distance:

$$\mathcal{K}(s, s') := \sum_{k=1}^{K} (s_k - s'_k)^2 .$$

For each t, we let

$$\mathcal{K}_t^{\text{cross}} := \frac{1}{2} \int \int \mathcal{K}(s, s') \gamma^*(\mathrm{d}s) \gamma^*(\mathrm{d}s') = \mathrm{tr}(\mathrm{Var}_{s\sim\gamma_t^*}(s))$$

be the *cross-sectional originality* of songs. The index $\mathcal{K}_t^{\text{cross}}$ measures how original the songs are within a given period t. Likewise, we let

$$\begin{aligned}\mathcal{K}_t^{\text{inter}} :=& \frac{1}{2(t-1)} \sum_{j=1}^{t-1} \int_{\mathbb{R}^K} \int_{\mathbb{R}^{K\times(t-1)}} \mathcal{K}^t(s, s') \gamma_j^*(\mathrm{d}s') \gamma_t(\mathrm{d}s) \\ =& \frac{1}{2(t-1)} \left((t-1)\mathrm{tr}(\mathrm{Var}_{s\sim\gamma_t^*}(s)) + \sum_{j=1}^{t-1} \mathrm{tr}(\mathrm{Var}_{s\sim\gamma_j^*}(s)) + \sum_{j=1}^{t-1} |\mathbb{E}_{s\sim\gamma_t^*}[s] - \mathbb{E}_{s\sim\gamma_j^*}[s]|^2 \right)\end{aligned}$$

be the *intertemporal originality* of songs. The index $\mathcal{K}_t^{\text{inter}}$ measure how original the songs in period t are compared to songs in the past.

2.1 Analysis

In this section, we consider three different recommendation mechanisms: no personalized recommendation, full exploitation mechanism, and full exploration mechanism. In practice, recommendation mechanisms are typically a mixed of these three. Our result will characterize equilibrium song distributions under these three bases separately, which can then be used to triangulate the equilibrium outcomes when the mechanisms are mixed. For simplicity, we assume that $\phi(y) = y$, $C(s) = \|s\|^2/2$, and $Q(m) = \lambda/m$ for some $\lambda > 0$. The equilibrium behavior is qualatively similar under different specifications of functional forms.

No Personalized Recommendation

We consider first the benchmark of no personalized recommendation. That is, $R^{\text{random}}(s) = \mathcal{N}$ for all s. In this case, $S_i^{R^{\text{random}}} = \mathbb{R}^K$ for all i, and hence in equilibrium $Q_i^* = \lambda$ for all i. That is, for each consumer i, every song will have λ chance of being listened to. Thus, given information set $\mathcal{F}$, producer z solves

$$\max_{s \in \mathbb{R}^K} \lambda \left[\sum_{i=1}^{N} \mathbb{E}[\langle \theta_i, s \rangle | \mathcal{F}] - z \frac{\|s\|^2}{2} \right] .$$

first-order condition implies that the equilibrium song given information set $\mathcal{F}$ is given by:

$$s = \frac{1}{z} \lambda \sum_{i=1}^{N} \mu_i(\mathcal{F}) \,,$$

where

$$\mu_i(\mathcal{F}) := \mathbb{E}[\theta_i | \mathcal{F}]$$

is the expectation of θ_i given information set $\mathcal{F}$.

In this equilibrium, each consumer θ_i faces the same set of songs

$$\left\{ \frac{\lambda}{z} \sum_{i=1}^{N} \mu_i(\mathcal{F}) \middle| z \in \mathbb{R}_+ \right\} ,$$

indexed by producer's productivity z, which are distributed according to a distribution γ^*.[2]

[2]More precisely,

$$\gamma^*(s_1 \leq x_1, \ldots, s_K \leq x_K) = \mathbb{P}\left[\frac{\lambda}{z} \sum_{i=1}^{N} \mu_{i1} \leq x_1, \ldots, \frac{\lambda}{z} \sum_{i=1}^{N} \mu_{iK} \leq x_K \right] = 1 - F\left(\max_k \left\{ \frac{\lambda \sum_{i=1}^{N} \mu_{ik}}{x_k} \right\} \right) ,$$

for all $x \in \mathbb{R}_+^K$

and each will randomly select a λ fraction of them. The average song selected is thus

$$\overline{s} := \lambda\beta\overline{\mu}(\mathcal{F})\,, \tag{4}$$

where $\beta := \mathbb{E}[1/z]$ and $\overline{\mu}(\mathcal{F}) = \sum_{i=1}^{N} \mu_i(\mathcal{F})$

To describe the evolution of information sets $(\mathcal{F}_t, \mathcal{G}_t)_{t\geq 0}$ throughout time, first note that since the platform does not actively recommend different songs to different consumers, it is without loss to keep track of the producer's beliefs $(\mu_{i,t}, \Sigma_{i,t})_{i=1}^{N}$.

Using equation (3) and equation (4), the evolution of $(\mu_{i,t}, \Sigma_{i,t})_{i=1}^{N}$ can be derived explicitly. The initial values are given by the prior: $\mu_{i,0} := \mu_i$, $\Sigma_{i,0} := \Sigma_i$. For each $t \in \mathbb{N}\cup\{0\}$ and for each i, the producer observes all the past histories in periods up until period $t-1$, the average song $\overline{s}_{i,t}$ a consumer i listens to in period t, and signal $y_{i,t} = \langle \theta_i, \overline{s}_{i,t}\rangle + \varepsilon_{i,t} + \eta_{i,t}$ in period t. With no recommendation, as equation (4) shows, the average song $\overline{s}_{i,t}$ listened to by consumer i in that period, is proportional to the producer's posterior belief of the aggregate type $\sum_i \theta_i$. That is, $\overline{s}_{i,t} = \lambda\beta\bar{\mu}_t$, for all i and for all t, where

$$\overline{\mu}_t := \sum_{i=1}^{N} \mu_{i,t}\,.$$

As a result, by equation (3), producer's posterior mean $(\mu_{i,t+1})i = 1^N$ and variance $(\Sigma_{i,t+1})_{i=1}^{N}$ in period $t+1$ are given by:

$$\begin{aligned}
\mu_{i,t+1} &= \mu_{i,t} + \lambda\beta \cdot \overline{\mu}_t^{\intercal} N_i(\lambda\beta\overline{\mu}_t)(\theta_i - \mu_{i,t}) + N_i(\lambda\beta\overline{\mu}_t)(\varepsilon_{i,t} + \eta_{i,t}) \\
\Sigma_{i,t+1} &= \Sigma_{i,t} - \lambda\beta \cdot N_i(\lambda\beta\overline{\mu}_t)\overline{\mu}_t^{\intercal}\Sigma_{i,t}\,.
\end{aligned}$$

This defines, recursively, a stochastic processes $(\mu_{i,t}, \Sigma_{i,t})$. The information sets $(\mathcal{F}_t)$ are then given by the natural filtration generated by all histories. Note that for all i and for all t, $\mu_{i,t+1} \mid \mathcal{F}_t$ is normally distributed, with mean

$$\mathbb{E}[\mu_{i,t+1} \mid \mathcal{F}_t] = \mu_{i,t}$$

and variance

$$\mathrm{Var}(\mu_{i,t+1} \mid \mathcal{F}_t) = \lambda\beta N_i(\lambda\beta\overline{\mu}_t)\overline{\mu}_t^{\intercal}\Sigma_{i,t}\,.$$

and thus $(\mu_{i,t})_{t\geq 0}$ is a martingale for all i. By the martingale convergence theorem, $\{\mu_{i,t}\} \to$

$\mu_{i,\infty}$ almost surely for some i, and $\{\Sigma_{i,t}\} \to \Sigma_{i,\infty}$ for all i. The limit must satisfy

$$\Sigma_{i,\infty}\overline{\mu}_{\infty} = \Sigma_{i,\infty}\sum_{i=1}^{N}\mu_{i,\infty} = 0\,.$$

Therefore, the long-run average song consumer i listens to is given by

$$\tilde{s}^{\text{random}} := \overline{s}_{i,\infty} = \lambda\beta\overline{\mu}_{\infty}\,.$$

Let $\mathcal{K}^{\text{random}} := \mathcal{K}^*(\tilde{s}^{\text{random}})$ denote the cross-sectional originality of songs in the long run under the recommendation mechanism.

It is noteworthy that in the long run, the posterior variance $\Sigma_{i,\infty}$ might not be zero, since it could be that $\bar{\mu}_{\infty}$ lies with probability one in the kernel of $\Sigma_{i,\infty}$ for all i. This means that full learning is not guaranteed, and the producer's belief updates—despite getting independent signals every period—might get "trapped" in certain directions. This is because song productions are endogenous and the producer only seeks to maximize their profit rather than internalizing learning efficiency. As a result, even in the long-run, equilibrium songs $\overline{s}_{i,\infty}$ might not be able to target precisely consumers' types θ_i.

In the meantime, the long-run average song $\overline{s}_{i,\infty}$, is the same for all consumer i. This is because producers only care about their profit, which depends only on the aggregate consumer surplus $\sum_i\langle\theta_i, s\rangle$, and every consumer has equal chance of listening to each songs.

Full Exploitation

Now, we consider the full exploitation recommendation mechanism R^{exploit}, which aims to maximize consumer surplus (given the platform's information set $\mathcal{G}$) in each period, taking into account the producers' costs. To formalize this idea, note that for a given $z \in \mathbb{R}_+$, given an information set $\mathcal{G}$, the the song that maximizes consumer i's expected surplus solves

$$\max_{s\in\mathbb{R}^K}\left[\langle\mu_i(\mathcal{G}), s\rangle - z\frac{\|s\|^2}{2}\right],$$

which is given by

$$\frac{1}{z}\mu_i(\mathcal{G})\,.$$

The *full exploitation* mechanism recommends to each consumer, all songs that maximizes consumer surplus net of cost for some producer. That is,

$$S_i^{R^{\text{exploit}}} = \{s | s = \alpha\cdot\mu_i(\mathcal{G}),\, \alpha \geq 0\}\,.$$

That is, for each consumer i, the mechanism recommends a song to this consumer if and only if it is proportional to their expected type given the platform's information set.

Proposition 1 below characterizes the equilibrium under the full exploitation mechanism given arbitrary information sets.

Proposition 1. *Suppose that the recommendation mechanism is R^{exploit}, the producers' information set is $\mathcal{F}$, and the platform's information set is $\mathcal{G}$. For each i, let*

$$P_i^* := \frac{\|\mu_i(\mathcal{G})\|}{\sum_{j=1}^N \|\mu_i(\mathcal{G})\|},$$

let

$$Q_i^* := \frac{\lambda}{P_i^*},$$

and let

$$\sigma^*(z, Q) := \sum_{i=1}^N P_i^* \delta_{\left\{\frac{Q_i}{z}\mu_i(\mathcal{G})\right\}},$$

for all $z \geq 0$ and for all $Q \in [0,1]^N$. Then (Q^*, σ^*) is an equilibrium given information sets $(\mathcal{F}, \mathcal{G})$.

Given information sets $(\mathcal{F}, \mathcal{G})$, in the equilibrium (Q^*, σ^*), each producer randomizes over songs that are proportional to the projection of the producer's expected type onto the platform's expected type, with the weights being proportional to the lengths of these projections.

The equilibrium song distribution is then given by

$$\gamma^*(s_1 \leq x_1, \ldots, s_K \leq x_K) = \sum_{i=1}^N \frac{\lambda}{Q_i^*}\left[1 - F\left(\max_k \left\{\frac{Q_i^*}{x_k}\mu_k(\mathcal{G})\right\}\right)\right]$$

and θ_i consumer randomly selects λ fraction of songs they are recommended to, which gives an average of

$$\overline{s}_i = \mathbb{E}_{s\sim\gamma^*}[s_i] = \mathbb{E}\left[\frac{1}{z}\right] P_i^* Q_i^* \mu_i(\mathcal{G}) = \lambda\beta\mu_i(\mathcal{G}).$$

To see how beliefs $(\mu_{i,t}, \Sigma_{i,t})$ and $(\nu_{i,t}, \Lambda_{i,t})$ evolve under full exploitation, Let the initial values be $\mu_{i,0} := \mu_i =: \nu_{i,0}$, $\Sigma_{i,0} := \Sigma_i =: \Lambda_{i,0}$. In each period t, by equation (3), the producers'

beliefs $(\mu_{i,t}, \Sigma_{i,t})$ evolves according to:

$$\mu_{i,t+1} = \mu_{i,t} + \lambda\beta \cdot \nu_{i,t}^{\intercal} N_i(\lambda\beta\nu_{i,t})(\theta_i - \mu_{i,t}) + N_i(\lambda\beta\nu_{i,t})(\varepsilon_{i,t} + \eta_{i,t})$$
$$\Sigma_{i,t+1} = \Sigma_{i,t} - N_i(\lambda\beta\nu_{i,t})\nu_{i,t}^{\intercal}\Sigma_{i,t}$$

and

$$\nu_{i,t+1} = \nu_{i,t} + \lambda\beta \cdot \nu_{i,t}^{\intercal} M_i(\lambda\beta\nu_{i,t})(\theta_i - \nu_{i,t}) + M_i(\lambda\beta\nu_{i,t})\varepsilon_{i,t}$$
$$\Lambda_{i,t+1} = \Lambda_{i,t} - M_i(\lambda\beta\nu_{i,t})\nu_{i,t}^{\intercal}\Lambda_{i,t}$$

Note that the process $\{\nu_{i,t}\}$ is a martingale with respect to the filtration generated by $\{\nu_{i,t}, \Lambda_{i,t}\}$. Thus,

$$\mathbb{E}[\nu_{i,t+1} \mid \nu_{i,t}, \Lambda_{i,t}] = \nu_{i,t}\,,$$

and

$$\mathrm{Var}(\nu_{i,t+1} \mid \nu_{i,t}, \Lambda_{i,t}) = M_i(\lambda\beta\nu_{i,t})\nu_{i,t}^{\intercal}\Lambda_{i,t}\,.$$

By the martingale convergence theorem, it then follows that $\{\nu_{i,t}\} \to \nu_{i,\infty}$ almost surely, where $\nu_{i,\infty}$ satisfy $\Lambda_{i,\infty}\nu_{i,\infty} = 0$ for all i almost surely. The long-run average song consumer i listens to is thus

$$\bar{s}_{i,\infty} = \lambda\beta\nu_{i,\infty}\,,$$

and the long-run songs in the market is given by

$$\tilde{s}^{\text{exploit}} = \bar{s}_{i,\infty},\ \text{w.p.}\ \frac{1}{N}\,.$$

Let $K^{\text{exploit}} := K^*(\tilde{s}^{\text{exploit}})$ denote the originality of songs in the long run under the recommendation mechanism.

Full exploration

A recommendation mechanism that fully aims for exploration maximizes the variance reduction. To formally define this mechanism, first recall that for a symmetric matrix A, the *spectrum norm* of A, $\|A\|$ is the square root of the largest eigen value of A. Given a variance covariance matrix Σ_i, equation (2) states that the platform's posterior variance of θ_i updates to

$$\Sigma_i - M_i(s)s^{\intercal}\Sigma_i\,.$$

when consumer i listens to s on average. Thus, the symmetric matrix

$$M_i(s)s^\intercal\Sigma_i = \Sigma_i s(s^\intercal\Sigma_i s + \sigma_\varepsilon^2)^{-1}s^\intercal\Sigma_i$$

can be interpreted as the *variance reduction* from listening to s. Let

$$L(s) := \|\Sigma_i s(s^\intercal\Sigma_i s + \sigma_\varepsilon^2)^{-1}s^\intercal\Sigma_i\|\,.$$

We say that the *full exploration mechanism* seeks to maximize the variance reduction $L(s)$, while taking into account the cost $C(s)$. Specifically, for a fixed z, consider the maximization problem

$$\max_{s\in\mathbb{R}^K}\left[L(s) - z\frac{\|s\|^2}{2}\right]\,.$$

It can be shown that the solution s^* is given by

$$s^* = \max\left\{\kappa(z)v^*, 0\right\}\,,$$

where v^* is the eigenvector (normalized to $\|v^*\| = 1$) of Σ_i with the largest eigenvalue, and

$$\kappa(z) := \sqrt{\frac{\sigma(\sqrt{2}\lambda^* - \sqrt{z}\sigma)}{\lambda^*\sqrt{z}}}\,,$$

which is decreasing in z, and $\lim_{z\to 0}\kappa(z) = \infty$. As a result,

$$\bigcup_{z\geq 0}\operatorname*{argmax}_{s\in\mathbb{R}^K}\left[L(s) - \frac{z\|s\|^2}{2}\right] = \{\alpha v^*|\alpha\geq 0\}\,.$$

That is, the set of songs that maximize the variance reduction net of cost for all producers are exactly vectors proportional to v^*.

We now define the *full exploration* mechanism R^{explore} as

$$S_i^{R^{\text{explore}}} := \{s|s = \alpha v(\mathcal{G})\,,\alpha\geq 0\}\,,$$

where $v_i(\mathcal{G})$ is the eigenvector corresponding to the largest eigenvalue of $\text{Var}(\theta_i\mid\mathcal{G})$.

The equilibrium analysis, under this definition, is then similar to the full exploitation mechanism, as summarized by Proposition 2 below. To describe the equilibrium outcome, we first establish some notation. For any vectors $x, y\in\mathbb{R}^K$, let

$$\text{Proj}\left[x\mid y\right] := \frac{\langle x, y\rangle}{\|y\|^2}y$$

be the projection of x onto $\text{span}\{y\}$.

Proposition 2. *Suppose that the recommendation mechanism is R^{explore}, the producers' information set is $\mathcal{F}$, and the platform's information set is $\mathcal{G}$. For each i, let*

$$P_i^* := \frac{\|\text{Proj}\left[\mu_i(\mathcal{F}) \mid v_i(\mathcal{G})\right]\|}{\sum_{j=1}^N \|\text{Proj}\left[\mu_i(\mathcal{F}) \mid v_i(\mathcal{G})\right]\|},$$

let

$$Q_i^* := \frac{\lambda}{P_i^*},$$

and let

$$\sigma^*(z, Q) := \sum_{i=1}^N P_i^* \delta_{\left\{\frac{Q_i}{z}\text{Proj}\left[\mu_i(\mathcal{F})|v_i(\mathcal{G})\right]\right\}},$$

for all $z \geq 0$ and for all $Q \in [0,1]^N$. Then (Q^*, σ^*) is an equilibrium given information sets $(\mathcal{F}, \mathcal{G})$.

In this equilibrium, since the platform only recommends songs proportional to $v_i(\mathcal{G})$ to consumer i, each producer essentially chooses among targeting which consumers. When targeting consumer i, the optimal song for a producer with z is given by $\frac{Q_i^*}{z}\text{Proj}\left[\mu_i(\mathcal{F}) \mid v_i(\mathcal{G})\right]$ In the equilibrium (Q^*, σ^*), each producer randomizes among targeting different types of consumers according to weights $\{P_i^*\}_{i=1}^N$ and chooses $\frac{Q_i^*}{z}\|\text{Proj}\left[\mu_i(\mathcal{F}) \mid v_i(\mathcal{G})\right]\|$ with probability P_i^*.

In this equilibrium, producer with productivity z randomizes among songs

$$\left\{\frac{Q_1^*}{z}\text{Proj}\left[\mu_1(\mathcal{F}) \mid v_1(\mathcal{G})\right], \ldots, \frac{Q_N^*}{z}\text{Proj}\left[\mu_N(\mathcal{F}) \mid v_N(\mathcal{G})\right]\right\}$$

and chooses $Q_i^*/z\text{Proj}\left[\mu_i(\mathcal{F}) \mid v_i(\mathcal{G})\right]$ with probability $P_i^* = \lambda/Q_i^*$.

The equilibrium song distribution is then given by

$$\gamma^*(s_1 \leq x_1, \ldots, s_K \leq x_K) = \sum_{i=1}^N \frac{\lambda}{Q_i^*}\left[1 - F\left(\max_k\left\{\frac{Q_i^*}{x_k}[\text{Proj}\left[\mu_i(\mathcal{F}) \mid v_i(\mathcal{G})\right]]_k\right\}\right)\right]$$

and θ_i consumer randomly selects λ fraction of songs they are recommended to, which gives an average of

$$\begin{aligned}\overline{s}_i :=& \mathbb{E}\left[\frac{1}{z}\right] Q_i^* P_i^* \text{Proj}\left[\mu_i(\mathcal{F}) \mid v_i(\mathcal{G})\right] \\ =& \lambda\beta\text{Proj}\left[\mu_i(\mathcal{F}) \mid v_i(\mathcal{G})\right].\end{aligned}$$

To see how the learning process behaves under full exploration, let the initial values be $\mu_{i,0} := \mu_i =: \nu_{i,0}$, $\Sigma_{i,0} := \Sigma_i =: \Lambda_{i,0}$. In each period t, let $\mu_{i,t}$ and $\nu_{i,t}$ be the posterior expected value of θ_i of the producers and the platform, respectively. Also, let $v_{i,t}$ be the eigenvector corresponding to the largest eigenvalue of $\Lambda_{i,t}$ for all i and for all t. The equilibrium average song listened to by consumer i, as shown above, is given by

$$\bar{s}_{i,t} = \lambda\beta \text{Proj}\left[\mu_{i,t} \mid v_{i,t}\right],$$

and the platform observes a signal $x_{i,t} = \langle \theta_i, \bar{s}_{i,t} \rangle + \varepsilon_{i,t}$, while the producers observe a signal $y_{i,t} = x_{i,t} + \eta_{i,t}$, where $\{\varepsilon_{i,t}, \eta_{i,t}\}$ are iid normal random variables. The beliefs of the platform $\{(\nu_{i,t}, \Lambda_{i,t})\}_{i=1}^N$ and producers $\{(\mu_{i,t}, \Sigma_{i,t})\}_{i=1}^N$, by equation (2), thus evolve over time according to:

$$\mu_{i,t+1} = \mu_{i,t} + \lambda\beta \cdot [\text{Proj}\left[\mu_{i,t} \mid v_{i,t}\right]]^\intercal N_i(\lambda\beta \text{Proj}\left[\mu_{i,t} \mid v_{i,t}\right])(\theta_i - \mu_{i,t}) + N_i(\lambda\beta \text{Proj}\left[\mu_{i,t} \mid v_{i,t}\right])(\varepsilon_{i,t} + \eta_{i,t})$$
$$\Sigma_{i,t+1} = \Sigma_{i,t} - N_i(\lambda\beta \text{Proj}\left[\mu_{i,t} \mid v_{i,t}\right])[\text{Proj}\left[\mu_{i,t} \mid v_{i,t}\right]]^\intercal \Sigma_{i,t}$$

and

$$\nu_{i,t+1} = \nu_{i,t} + \lambda\beta \cdot [\text{Proj}\left[\mu_{i,t} \mid v_{i,t}\right]]^\intercal M_i(\lambda\beta \text{Proj}\left[\mu_{i,t} \mid v_{i,t}\right])(\theta_i - \mu_{i,t}) + M_i(\lambda\beta \text{Proj}\left[\mu_{i,t} \mid v_{i,t}\right])\varepsilon_{i,t}$$
$$\Lambda_{i,t+1} = \Lambda_{i,t} - M_i(\lambda\beta \text{Proj}\left[\mu_{i,t} \mid v_{i,t}\right])[\text{Proj}\left[\mu_{i,t} \mid v_{i,t}\right]]^\intercal \Lambda_{i,t}$$

where $v_{i,t+1}$ is the (unit) eigenvector of

$$M_i(\lambda\beta \text{Proj}\left[\mu_{i,t} \mid v_{i,t}\right])[\text{Proj}\left[\mu_{i,t}\right] \mid v_{i,t}]^\intercal \Lambda_{i,t}$$

with the largest eigenvalue for all i, t. Let $\{\mathcal{H}_t\}_{t\geq 0}$ be the natural filtration generated by this process. Then both $\{\mu_{i,t}\}$ and $\{\nu_{i,t}\}$ are martingales with respect to $\{\mathcal{F}_t\}$. That is, $\mathbb{E}[\mu_{i,t+1} \mid \mathcal{F}_t] = \mu_{i,t}$ and $\mathbb{E}[\nu_{i,t+1} \mid \mathcal{F}_t] = \nu_{i,t}$. Moreover,

$$\text{Var}(\mu_{i,t+1} \mid \mathcal{H}_t) = N_i(\lambda\beta \text{Proj}\left[\mu_{i,t} \mid v_{i,t}\right])[\text{Proj}\left[\mu_{i,t} \mid v_{i,t}\right]]^\intercal \Sigma_{i,t}$$

and

$$\text{Var}(\nu_{i,t+1} \mid \mathcal{H}_t) == M_i(\lambda\beta \text{Proj}\left[\mu_{i,t} \mid v_{i,t}\right])[\text{Proj}\left[\mu_{i,t} \mid v_{i,t}\right]]^\intercal \Lambda_{i,t}\,.$$

In fact

$$\mu_{i,t+1} \mid \mathcal{H}_t \sim N(\mu_{i,t}, \Sigma_{i,t+1} - \Sigma_{i,t})$$

and

$$\nu_{i,t+1} \mid \mathcal{H}_t \sim N(\nu_{i,t}, \Lambda_{i,t+1} - \Lambda_{i,t})\,.$$

By the martingale convergence theorem, it then follows that $\{\mu_{i,t}\} \to \mu_{i,\infty}$ and $\{\nu_{i,t}\} \to \nu_{i,\infty}$ almost surely, where $\mu_{i,\infty}$ and $\nu_{i,\infty}$ satisfy $\Sigma_{i,\infty}\mu_{i\cdot\infty} = 0 = \Lambda_{i,\infty}\nu_{i,\infty}$ for all i almost surely. The long-run average song consumer i listens to is thus

$$\bar{s}_{i,\infty} = \lambda\beta \text{Proj}\left[\mu_{i,\infty} \mid v_{i,\infty}\right],$$

and the long-run songs in the market is given by

$$\tilde{s}^{\text{exploit}} = \bar{s}_{i,\infty}, \text{ w.p. } \frac{1}{N}.$$

Let $\mathcal{K}^{\text{explore}} := \mathcal{K}^*(\tilde{s}^{\text{explore}})$ denote the originality of songs in the long run under the recommendation mechanism.

Comparison of Originality

Combining the analyses above, we have the following comparison of originality of songs in the long run under three different mechanisms.

PROPOSITION 3. $\mathcal{K}^{\text{explore}} > \mathcal{K}^{\text{exploit}} > \mathcal{K}^{\text{random}}$.

To illustrate Proposition 3, Figure 1 presents the long-run distribution of songs under different recommendation regimes in an economy with $N = 15$ consumers, each characterized by a true preference vector $\theta_i \in \mathbb{R}^2$. [3] The black points denote the true preferences θ_i, while gray rays indicate the direction of each taste vector. The red star marks the aggregate preference, $\sum_i \theta_i$. The colored markers represent songs: yellow circles for the no-PRA regime, orange crosses for exploitation, blue triangles for exploration, and green asterisks for the optimal regime. Each cloud of markers comes from repeating the learning process many times with different randomized priors and noise realizations, so that the plot shows the distribution of asymptotic song positions across simulations.[4]

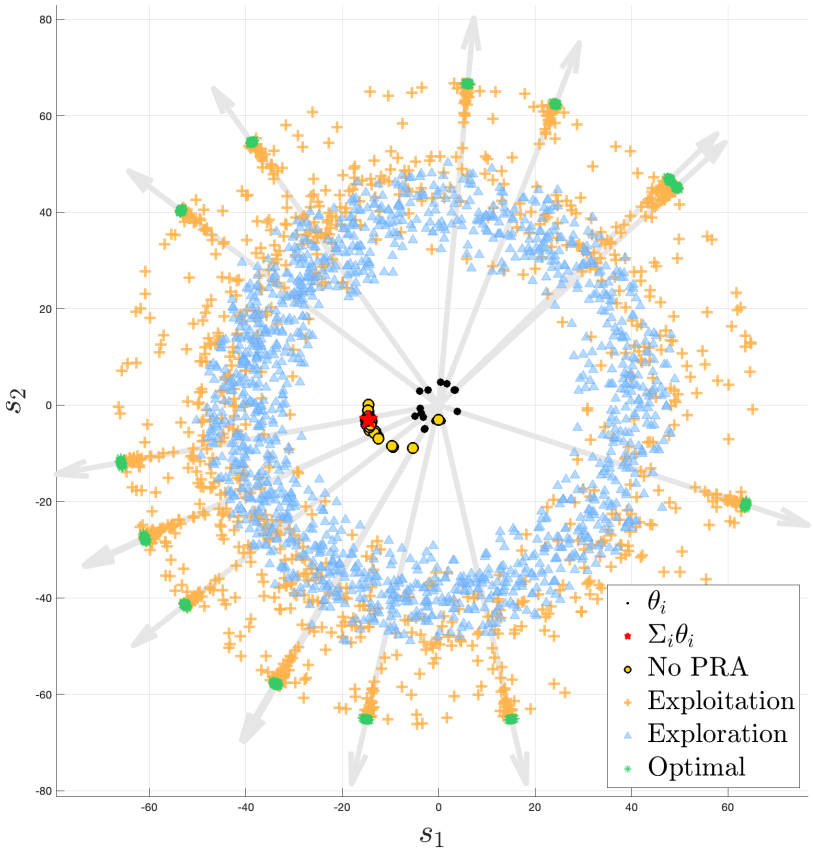
Without personalized recommendation, songs collapse toward the aggregate of posterior beliefs. Specifically, the no-PRA rule generates $\tilde{s}^{\text{random}} = \sum_i \mu_{i,\infty}$. This long-run song points toward the average taste, but because it remains close to the origin, the implied revenue is low—recall that revenue grows with the magnitude of the long-run song. The lack of differentiation means the producer effectively averages across all consumers, ignoring preference heterogeneity.

Under exploitation, producers align songs with predicted tastes, generating outcomes of the form $\tilde{s}^{\text{exploit}} \propto \nu_{i,\infty}$. These songs lie in the direction of the current posterior mean for each

[3] For simplicity, we assume $\lambda = z = 1$, $\sigma_\eta = 0$, and we performe 100 independent simulations.

[4] The circular symmetry of the clouds across regimes reflects the isotropy of Gaussian priors and shocks.

Figure 1: Long-Run Distribution of Songs



Notes: The figure shows the long-run distribution of songs under four regimes: no PRA, exploitation, exploration, and optimal. It is based on $N = 15$ consumers with preference vectors $\theta_i \in \mathbb{R}^2$. Priors for posterior means are drawn from standard normal distributions, and covariance matrices are initialized as perturbed identities. Signals are updated for $T = 100$ periods under each regime. The procedure is repeated 100 times with independent draws of priors and noise. We assume $\lambda = z = 1$ and $\sigma_\eta = 0$. Colored markers represent the limiting song locations under each regime: yellow circles (no PRA), orange crosses (exploitation), blue triangles (exploration), and green stars (optimal). Black points are true preferences θ_i, gray rays show their directions, and the red "X" denotes the aggregate preference $\sum_i \theta_i$.

consumer. By concentrating along these directions, exploitation produces higher revenue than no recommendation, since songs move further away from the origin. However, the regime fails to deliver full learning because signals merely reinforce prior beliefs, preventing producers from uncovering the complete structure of preferences and leaving residual uncertainty about the true θ_i.

In exploration, signals are aligned with the directions that minimizes uncertainty reduction, leading to songs of the form $\tilde{s}^{\text{explore}} \propto v_{i,\infty}$, where $v_{i,\infty}$ is the eigenvector associated with

the largest eigenvalue of the posterior covariance. This pushes songs along the direction that, when listened to, gives the largest marginal information regarding consumers' tastes. The benefit is richer information acquisition: producers learn along dimensions that exploitation ignores. Yet full learning may not be guaranteed, since the process can still leave residual uncertainty depending on the geometry of the priors and the noise. The blue triangles in the figure trace out the wide distribution of such exploratory songs, which tend to lie further out in the plane, but are not always perfectly aligned with consumer preferences.

Lastly, the optimal regime blends the logic of exploitation and exploration. It rescales the exploratory posterior means, yielding outcomes of the form $\tilde{s}^{\text{optimal}} \propto \nu_{i,\infty}$, with $\nu_{i,\infty}$ shaped by exploration. Songs are thus aligned with the best predictions obtained through active information acquisition. This regime delivers the highest revenue, as songs extend farthest from the origin while remaining tied to consumer preference directions. Learning may remain partial, but the green stars in the figure cluster along true tastes while pushing outward in magnitude, illustrating their higher revenue.

3 Application: Music Industry

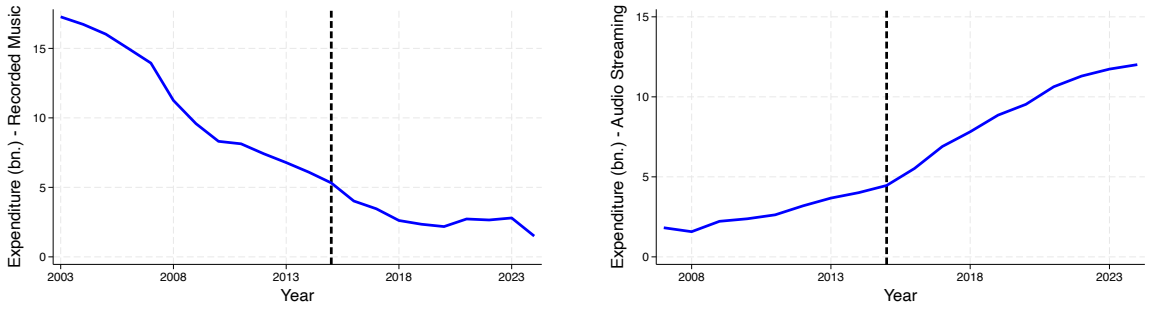
3.1 Streaming Takes Over

The adoption of personalized recommendation algorithms in music followed a fundamental shift in how consumers access content. Until the early 2010s, most expenditure in the industry was concentrated on recorded music in the form of discs, tapes, vinyl, and digital downloads. As shown in panel (a) of Figure 2, spending on recorded music fell sharply after 2008, declining from nearly $15 billion to less than $3 billion by the early 2020s.

In contrast, audio streaming and radio services expanded rapidly over the same period. Panel (b) of Figure 2 shows that expenditure on streaming rose steadily after 2010, overtaking recorded music by the mid-2010s and continuing to grow thereafter. Today, streaming dominates the industry, with expenditures above $12 billion annually.[5] Streaming platforms have become the primary channel through which consumers access music, creating an environment in which recommendation algorithms play a central role.

[5]The timeline of platform entry illustrates how quickly this transformation occurred: Rhapsody (2001), Last.fm (2002), Pandora (2005), Spotify (2008 in Sweden and 2011 in the United States), Apple Music (2015), Amazon Music (2016), and YouTube Music (2015).

Figure 2: Recorded Music versus Audio Streaming: Expenditures



(a) Recorded Music (b) Audio Streaming

Notes: Expenditures are in billions of U.S. dollars. Recorded music includes audio discs, tapes, vinyl, and digital downloads. Audio streaming includes streaming and radio services such as satellite radio. Data on expenditures for audio streaming began to be recorded in 2007. The dashed line marks 2014, which coincides with the large-scale adoption of personalized recommendation algorithms. Source: Personal Consumption Expenditures (PCE), Bureau of Economic Analysis (BEA).

3.2 Personalized Recommendation Algorithms: Timeline

The transition from recorded music to streaming was followed by the widespread adoption of personalized recommendation algorithms. A key milestone occurred in 2014, when Spotify acquired *The Echo Nest*, a music intelligence company specializing in artificial intelligence and machine learning techniques for analyzing audio content. This acquisition marked the platform's strategic turn toward large-scale algorithmic personalization.[6]

In 2015 Spotify launched *Discover Weekly*, the first large-scale AI-powered playlist, which quickly became a hallmark of personalized streaming. Other platforms soon followed. In 2016 Apple Music introduced *My New Music Mix* and *My Favorites Mix*, and in 2019 YouTube Music launched *Discover Mix*, *Your Mix*, and *New Release Mix*.

By the late 2010s, nearly 90 percent of the global streaming market was operating with personalized recommendation algorithms as a central feature of how users accessed and discovered music.

[6]This built on smaller acquisitions such as Tuningo (2013) and was followed by others including Seed Scientific (2015), Cord Project and Soundwave (2016), CrowdAlbum (2016), Preact (2016), Sonalytic (2017), MightyTV (2017), and Mediachain (2017).

4 Data

To evaluate how PRAs affect originality, we measure the cross-sectional similarity of songs before and after their widespread adoption. Following guidance from the Yale School of Music, we rely on tools from Music Information Retrieval (MIR), a field that develops computational methods for analyzing and retrieving information from audio recordings. MIR techniques are widely used in practice for recommendation and playlisting, genre classification, music search services such as Shazam, and detecting cover songs or plagiarism. In our setting, these methods allow us to quantify how similar or distinct songs are from one another, enabling a systematic comparison of musical originality before and after the introduction of PRAs.

4.1 Audio Files

We acquire song files from Amazon Music to conduct audio content analysis.Our primary sample consists of the Top 100 songs in the pop, rock, and country genres for the years 2003, 2008, 2013, 2018, and 2023.[7] Songs are assigned to years based on their release dates, and all within-year and cross-year comparisons preserve this release chronology. We focus on these genres because of their broad and persistent popularity over time. Genre assignment is defined using Spotify's genre taxonomy. To provide a more representative view of the music landscape beyond the very top of the charts, we also construct a secondary sample. For each genre and year, we draw a random sample of 100 songs from the Top 1000 songs in that genre-year. In total, this yields: 100 songs $\times$ 5 years $\times$ 3 genres $\times$ 2 sample = 3,000 songs.

To measure musical similarity, we compute pairwise similarity matrices within each genre and sample. For a given genre-sample combination, pooling the 100 songs from each of the five years yields 500 songs. We compute all pairwise similarities within this set, resulting in a 500 $\times$ 500 matrix, or 250,000 similarity scores per genre–sample combination. Across 3 genres and 2 samples, this amounts to 1.5 million pairwise similarity scores in total.[8]

4.2 Spotify API Data

We complement the raw audio files with metadata from the public Spotify API, which provides detailed artist and track information such as genre, duration, and popularity. Spotify

[7] Total streams are only available for the top 200 songs through Spotify Charts from 2017 onward. Most streaming activity is concentrated among the most popular songs. Figure A7 shows that the top 100 songs account for approximately 73% of all streams among the top 200 songs.

[8] As will become clear below, computing this number of pairwise similarities is computationally intensive. Generating the full set of similarity scores requires approximately four weeks on a Mac Studio equipped with an Apple M2 Ultra chip (24 cores: 16 performance and 8 efficiency cores) and 192 GB of memory.

is the leading global streaming platform, with over 600 million users.[9]

Spotify structures its data into distinct objects, including Albums, Artists, and Tracks. Each object is uniquely identified by a Spotify ID, a long alphanumeric string that serves as the primary key across multiple dataset components. The query endpoint is the principal method for retrieving these identifiers, allowing searches by track names, album names, genres, or release years and returning lists containing partial metadata along with Spotify IDs. We obtain this information for each raw audio file in our dataset, as well as for the top 1000 songs in each of the selected genres on Spotify for every year from 2003 to 2023.[10] Each track entry includes metadata on all associated artists and albums, along with duration, release date, name, and popularity score.

In what follows, we develop several metrics using Music Information Retrieval (MIR) techniques that analyze raw audio files directly. This approach has several advantages: it is reproducible with open-source tools, transparent in its definitions, flexible enough to incorporate methodological improvements over time, and ensures comparability across datasets and years.[11] We emphasize that these metrics are already well established in the music information retrieval (MIR) literature. We deliberately follow the existing literature, as our contribution lies in the scale and application of these techniques rather than in their construction. In what follows, we seek to adhere closely to established practices in the field.

5 Audio Content Analysis

5.1 Similarity Across Songs

We measure similarity across songs by following the methodology of Bello (2011). The goal is to assess whether songs are becoming more similar over time by measuring how songs

[9] Figure A8 uses credit card data from over 180 million individuals to estimate that Spotify's mean zip code has 0.28 accounts per capita (median: 0.21). Figure A9 shows that Spotify is more prevalent among individuals under 35 years of age.

[10] We us the following API endpoint in July 2025: https://developer.spotify.com/documentation/web-api/reference/get-recommendation-genres.

[11] Spotify has long provided proprietary audio features through its Web API. These metrics, originally developed by The Echo Nest, which was acquired by Spotify in 2014 and quantify musical attributes such as acousticness, danceability, energy, instrumentalness, liveness, loudness, speechiness, and valence on a 0–1 scale. Additional information included tempo, key, and mode, although chord progressions were not explicitly available. Despite their widespread use, the exact definitions and update frequency of these measures have never been disclosed, creating uncertainty about whether they remain static, are periodically recalculated, or are redefined. In November 2024, Spotify further restricted access by removing many of these features from its Web API for all new applications, as part of an effort to limit the ability of third-party developers to compete with its own recommendation technology. For these reasons, we do not rely on Spotify's metrics for our main analysis.

released in different years compress when compared to each other.[12] To construct a measure of similarity, we follow these steps.[13]

Step 1 - Identifying Notes at Any Given Moment: To analyze a song's structure, we first convert the raw audio signal, a long sequence of sound wave amplitudes in the time domain, into its frequency components using a Fourier Transform, which reveals which pitches are present at each moment in time. Since the same musical note occurs across multiple octaves (e.g., A at 440 Hz, 880 Hz, 1760 Hz, etc.), we map the detected frequencies to their corresponding note names. This enables consistent identification of notes across pitch ranges and facilitates the analysis of melodies and harmonies. In the Western 12-note octave system, each note lies within a specific frequency band, which can be isolated through this transformation.

To capture how pitch content evolves over time, we divide the signal into overlapping short segments, each a few milliseconds in duration, and analyze them individually. This produces a time series of note intensities, indicating the relative strength of each note in each segment. Instead of using raw amplitude or loudness, we compute the energy in each frequency, which more accurately reflects the contribution of each note to the sound and is more robust to variations in instrumentation and recording conditions.

The resulting representation is a chromagram: a 12-row-by-time matrix in which each row corresponds to a musical note (e.g., C, C#, D...) and each column corresponds to a specific moment in time. This representation functions as a simplified, time-evolving form of sheet music, capturing the harmonic and melodic content of the audio signal while remaining resilient to changes in timbre, tuning, and volume.

Figure 3 shows the chromagram for the first 20 seconds of *Let It Be* by The Beatles. The underlying $12 \times M$ chromagram matrix is defined as $\hat{C} = \left[\hat{\mathbf{C}}_0,\ \hat{\mathbf{C}}_1,\ \ldots,\ \hat{\mathbf{C}}_{M-1}\right]$, where rows $p = 0, \ldots, 11$ correspond to pitch classes and columns $m = 0, \ldots, M-1$ correspond to successive time frames. The chromagram represents the distribution of pitch content over time by projecting frequency components onto twelve chroma bands: C, C#, D, D#, E, F, F#, G, G#, A, A#, and B. Each time window aggregates the intensity of these chroma values, resulting in a time-series representation of the song's harmonic content. As time progresses, the chromagram captures changes in the song's structure, highlighting the most prominent notes at different moments and key tonal events.

[12] Here, "compress" refers to the process of reducing the amount of data required to represent a song by eliminating redundancies, such as repeated patterns or structures. For example, if two songs share similar musical structures and repetitions, a compression algorithm will reduce their data size in a similar manner when compressing them individually and when compressing them together.

[13] We provide a technical description of all the steps taken in Appendix B.

Figure 3: Chromagram



Notes: The figure shows the chromagram for the first 20 seconds of *Let It Be* by The Beatles and depicts the distribution of pitch content over time by mapping musical notes onto twelve chroma bands. C, C#, D, D#, E, F, F#, G, G#, A, A#, and B. The x-axis represents time in seconds, while the y-axis corresponds to the chroma values. Each point in the chromagram indicates the intensity of a specific pitch class at a given moment.

Step 2 - Identifying Repeated Sections: Given the chromagram $\{\hat{\mathbf{C}}_m\}_{m=0}^{M-1}$, we identify repeated patterns by analyzing the time-time similarity matrix. Similarity between frames is computed as:

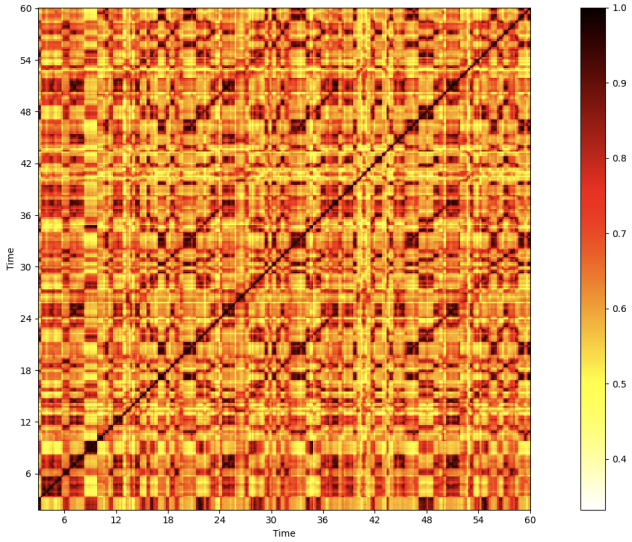
$$S_{n,m} = 1 \; - \; \frac{\left\|\hat{\mathbf{C}}_n - \hat{\mathbf{C}}_m\right\|_2}{\sqrt{12}}, \qquad n, m = 0, \ldots, M-1,$$

so that $S_{n,m} = 1$ for identical chroma frames and decreases toward 0 as they diverge. In this $M \times M$ matrix, repeated musical sections appear as diagonal stripes. Figure 4 presents the time-time similarity matrix for *Let It Be*. This matrix is symmetric along its main diagonal; thus, it always contains a 45-degree line corresponding to self-comparison. Additional parallel diagonal lines indicate repeated sections, such as the chorus, which reoccurs at different times in the song. The spacing of these diagonal patterns corresponds to the time intervals between repetitions.

Step 3 – Delay Embedding for Temporal Structure: To capture short-term melodic and rhythmic context, we construct a delay-embedded version of the chromagram. Instead of comparing individual chroma frames, which can be sensitive to local tempo shifts or misalignment, we stack a sequence of consecutive frames into a higher-dimensional vector

$$\mathbf{y}_n = \left[\hat{\mathbf{C}}_n,\ \hat{\mathbf{C}}_{n-1},\ \ldots,\ \hat{\mathbf{C}}_{n-10}\right] \in \mathbb{R}^{12\times 11}.$$

Figure 4: Similarity Matrix: *Let It be* by The Beatles



Notes: The figure shows the time-time similarity matrix of *Let It Be* by the Beatles, which represents the self-similarity of chroma features over time. The x-axis and y-axis both correspond to time in seconds, covering the entire duration of the song. Each point (n, m) in the matrix represents the computed similarity between the chroma features at time frames n and m. The color intensity in the matrix indicates the degree of similarity.

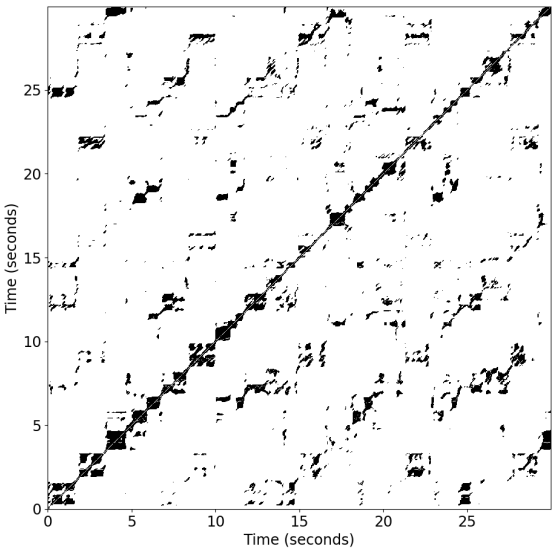
This transformation encodes local musical patterns across time into a single feature vector.[14] The resulting representation improves robustness to timing variation and enhances the detection of structural similarity between songs.

Step 4 – Recurrence Plot Generation: Using the delay-embedded sequence, we compute a pairwise Euclidean distance matrix between all vectors and generate a recurrence plot by thresholding distances: entries are set to 1 when the distance falls below the median value, and 0 otherwise. This binarized recurrence plot highlights repeating musical structures while suppressing noise and reducing dimensionality. Figure 5 shows the recurrence plot for *Let It Be*. Darker regions indicate pairs of time points with greater similarity, corresponding to smaller distances in the delay-embedded space. These areas highlight repeated musical patterns within the song.

Step 5 – Normalized Compression Distance (NCD) Calculation: The final step involves computing the Normalized Compression Distance (NCD) between the recurrence

[14]Following Bello (2011), we set the embedding dimension to 10.

Figure 5: Recurrence Plot: *Let It be* by The Beatles



Notes: The figure shows the recurrence plot of the Beatles song *Let It Be*. The recurrence plot is symmetric because the distance metric used in the delay-embedded space is symmetric– i.e., the distance from point A to point B is identical to the distance from point B to point A. This symmetry ensures that the recurrence plot mirrors itself along the diagonal.

plots of two songs:

$$NCD(s, s') = \frac{K(s, s') - \min(K(s), K(s'))}{\max(K(s), K(s'))}$$

where s and s' represent the recurrence plots of the two songs, and $K(\cdot)$ denotes the Kolmogorov complexity, defined as the minimum length of a computer program in a fixed universal language required to generate the sequence.[15] For instance, a song like *Twinkle, Twinkle, Little Star* has a low $K(s)$ since a short program can generate its repetitive structure, whereas *Beethoven's Symphony No. 9* has a high $K(s)$ due to its many variations and transitions. Similarly, a randomly generated song would exhibit a high $K(s)$ because it lacks repeating patterns. The NCD quantifies how well the songs compress together relative to compressing them individually. This metric is advantageous because it does not require explicit synchronization of feature sequences (e.g., tempo or pitch alignment).

Because compression algorithms are sequential, the finite-sample approximation to Kolmogorov complexity is not perfectly symmetric: compressing the concatenation (s, s') need

[15]Following Cilibrasi and Vitányi (2005), in our application, we approximate this measure using `bzlib`, a widely used compression algorithm based on the Burrows–Wheeler transform.

not yield the same length as compressing (s', s). Following common practice in empirical applications of compression-based distances, we remove this ordering dependence by defining the symmetrized pairwise measure

$$\mathcal{K}(s, s') \equiv \frac{1}{2}\left[NCD(s, s') + NCD(s', s)\right], \tag{5}$$

which yields a symmetric measure of dissimilarity between songs.

We interpret $\mathcal{K}(s, s')$ as an originality measure: it equals zero when two songs are identical, indicating no new information relative to one another. Cover versions typically generate values around 0.96–0.99, reflecting a high degree of similarity while still capturing subtle structural differences.

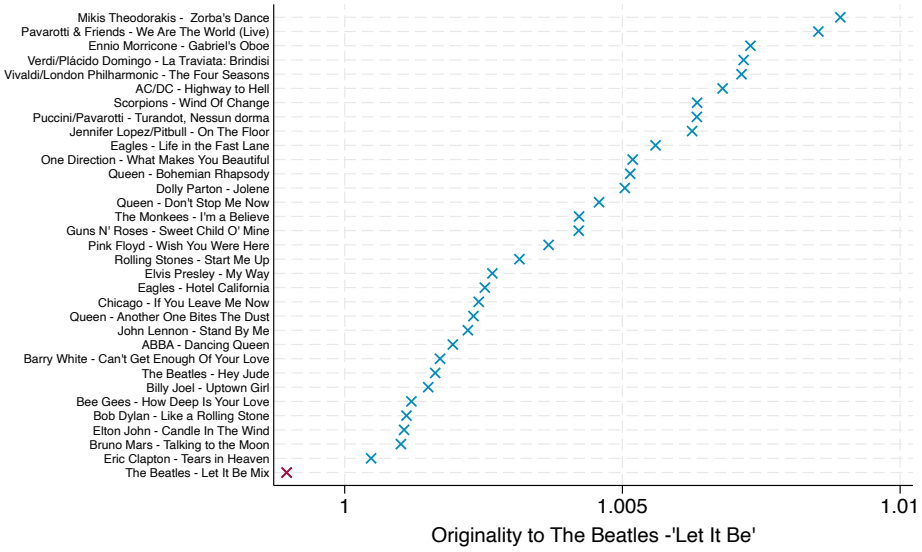
Add something about how the model connects to the data

For each song s, we define its overall originality as the average pairwise value

$$\mathcal{K}(s) \equiv \frac{1}{|\mathcal{S}| - 1} \sum_{s' \in \mathcal{S},\, s' \neq s} \mathcal{K}(s, s'), \tag{6}$$

where $\mathcal{S}$ denotes the set of songs in the sample. This produces a song-level originality score summarizing how distinct s is relative to the rest of the dataset.

Figure 6: Originality Relative to *Let It be* by The Beatles



Notes: The figure plots originality scores relative to The Beatles' *Let It Be*. The red marker denotes the 2021 remix of *Let It Be*.

Figure 6 illustrates the distribution of originality scores relative to The Beatles' *Let It*

Be. The red marker at the baseline corresponds to the 2021 remix of *Let It Be*, which lies below 1 since it shares structural and harmonic features with the original track. Songs that are stylistically similar, such as Elton John's *Candle in the Wind* or John Lennon's *Imagine*, also appear close to the benchmark, reflecting low originality since their harmonic progressions and overall structure compress efficiently together with *Let It Be*. By contrast, live versions of songs—for example, performances with audience noise, improvisation, or added instrumentation—tend to lie farther away. This occurs because live recordings contain more heterogeneous acoustic signals and noise. Finally, stylistically distinct songs such as Vivaldi's *Four Seasons* or AC/DC's *Highway to Hell* appear at the far end of the distribution with the highest originality scores, consistent with their divergence in both genre and musical structure relative to *Let It Be*.

6 Evolution of Song Characteristics Over Two Decades

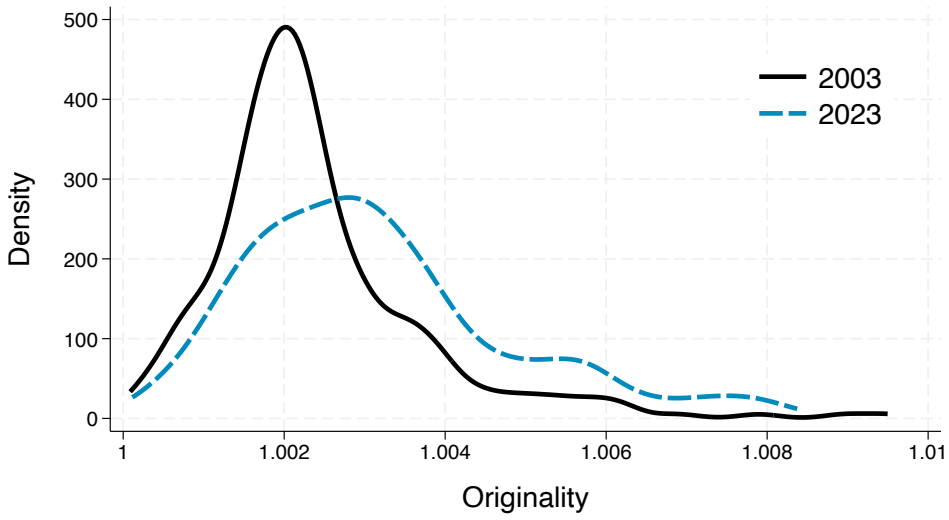
6.1 Cross-Sectional Originality Over Time

We begin by computing song-level originality separately for each year and genre. For each year–genre combination, we construct a 100 × 100 matrix of pairwise similarities. Because the diagonal entries correspond to a song's similarity with itself, they are equal to zero by construction and are excluded from the aggregation. We symmetrize each off-diagonal pair using Equation 5 and then compute song-level originality according to Equation 6.

Figure 7 presents the distribution of song-level originality, $\mathcal{K}(s)$, in 2003 and 2023, after controlling for differences in genre. The distribution shifts to the right over time, indicating that songs in 2023 are more differentiated from other songs released in the same year. Consistent with this pattern, the mean originality increases from 1.002383 (std. dev. 0.001332) in 2003 to 1.003136 (std. dev. 0.001652) in 2023. A paired t-test confirms that the increase is statistically significant (t=-5.85, p<0.001). Replicating the analysis with songs selected from the Top 1000 in each genre-year yields qualitatively and quantitatively similar results (Figure A10).

To put these magnitudes into perspective, if we take The Beatles' *Let It Be* as a benchmark, as shown in Figure 6, average originality in 2003 is comparable to John Lennon's *Stand by Me*, whereas by 2023 it resembles the Rolling Stones' *Start Me Up*. This shift indicates that songs released in 2023 are more differentiated from their contemporaries than songs released in 2003, a pattern that coincides with the period following the introduction of personalized recommendation algorithms and is consistent with our model's prediction that

Figure 7: Distribution of Originality Across Songs: 2003-2023



Notes: The figure shows the distribution of originality for 2003 and 2023, after controlling for song genre. We compute the pairwise similarity among the songs and then average these values to obtain a song-specific originality score. The sample consists of the top 100 songs in the pop, rock, and country genres.

personalization increases cross-sectional dispersion in song attributes.[16]

6.2 Regression Evidence and Song-Structure Controls

Although the distributional evidence is suggestive, we now assess the robustness of the increase in cross-sectional originality in a regression framework. We estimate

$$\mathcal{K}_{igt} = \beta \mathbb{1}\{\text{PostPRA}_t\} + \theta_g + \mathbf{X}'_{igt}\Gamma + \varepsilon_{igt}, \tag{7}$$

where $\mathcal{K}_{igt}$ is song-level originality for song i in genre g and year t, and θ_g denotes genre fixed effects. The indicator $\mathbb{1}\{\text{PostPRA}_t\}$ equals one for years in which personalized recommendation algorithms (PRAs) were widely deployed (2015, 2018, and 2023), and zero for earlier years (2003, 2008, and 2013).

It is important to incorporate controls beyond genre. Over the streaming era, songs have become progressively shorter and key musical elements have moved earlier in the track

[16] Figure A1 in the Appendix shows the patterns for Taylor Swift– the only artist in our sample with top 100 songs in both 2008 and 2023. It displays the pairwise similarity of two songs—"You Belong With Me" (2008) and "You're Losing Me" (2023)—relative to the rest of the top 100 pop songs of those years. It reveals patterns similar to those observed in the overall song distribution. Even for Taylor Swift's songs, the figure indicates that her overall song originality has increased.

(e.g. Figure A5). These shifts are plausibly related to streaming incentives, since a stream is typically counted after a short listening threshold and platform ranking mechanisms reward early engagement. Such changes in duration and structure can mechanically affect compression-based similarity measures, and therefore may confound time-series comparisons of cross-sectional originality. For that reason, we augment equation (7) with controls for song length and for two well-established Music Information Retrieval (MIR) measures that capture the timing of salient song components.

Chorus onset time. The chorus is the section of a song that is typically repeated the most times and serves as a central, most memorable part of the composition. We construct a measure of the *main chorus onset time*, defined as the second at which the song's primary repeated section begins. The algorithm follows standard MIR practice for structural repetition detection, which follows the work by Goto (2006). We first transform each audio file into a normalized chromagram. We then compute a self-similarity matrix from these chroma features to detect repeated musical sections and retain only repeated segments that last at least 10 seconds, which helps filter out false positives, such as brief melodic phrases or recurring instrumental motifs that do not function as a full chorus. We identify the main chorus as the most consistently recurring segment across the track. The onset (in seconds) of this segment is our *best chorus onset time*; we use its logarithm as a control to capture systematic shifts in song structure over time. Appendix B provides full implementation details.

Vocal onset time. Vocal detection is a well-established field in audio analysis.[17] We use voice detection to determine if the *vocal onset time* is changing over time, defined as the second at which vocal activity first appears in the track. We begin by isolating the vocal stem using Open-Unmix, a neural network-based framework for music source separation Stöter et al. (2019), which applies a learned time–frequency mask to the song's spectrogram to recover the vocal component. We convert the resulting vocal spectrogram to decibels and apply temporal smoothing to attenuate short-lived transients and background noise. We then track increases in vocal energy over time, smooth the resulting series, and define vocal onset as the first prominent peak (above the 90^{th} percentile) subject to a two-second separation rule. The resulting onset time is measured in seconds, and we use its logarithm as a control. Appendix B provides full details.

Table 1 reports the results. We pool both samples—the Top 100 songs and the random sample of songs from the Top 1000 for each genre and year—and include an indicator for the

[17] It plays an important role in everyday applications such as voice calls, audio processing, and command recognition for electronic devices and smartphones.

Table 1: Cross-sectional Originality Before and After PRAs

Note: The dependent variable is song-level originality, $\mathcal{K}(s)$, multiplied by 1000. The regressions pool two samples: (i) the Top 100 songs in each genre-year and (ii) 100 songs randomly drawn from the Top 1000 in the same genre-year; all specifications include a dummy for the sampling frame. Post-PRA$_t$ is an indicator for years after PRAs were introduced at scale. Columns (1)–(3) report unweighted estimates, while columns (4)–(6) weight observations by Spotify's popularity metric, an integer index between 0 and 100 from the Spotify Web API that reflects the relative popularity of a track and is based primarily on total streams with greater weight placed on recent listening activity. Controls include the logarithm of song duration; columns (3) and (6) additionally control for the logarithm of the best chorus onset time and the logarithm of the vocal onset time. Standard errors are clustered at genre × year.

	Unweighted			Popularity-weighted		
	(1)	(2)	(3)	(4)	(5)	(6)
	Originality	Originality	Originality	Originality	Originality	Originality
Post-PRA	0.5519***	0.3981**	0.3930**	0.5990***	0.4269**	0.4226**
	(0.176)	(0.155)	(0.149)	(0.185)	(0.163)	(0.159)
Log Duration		-1.4212***	-1.0735***		-1.4682***	-1.1465***
		(0.190)	(0.209)		(0.220)	(0.228)
Observations	3,000	2,991	2,801	2,991	2,991	2,801
R-squared	0.043	0.083	0.076	0.052	0.091	0.084
Song Controls	N	N	Y	N	N	Y
Genre	Y	Y	Y	Y	Y	Y

sampling frame in all specifications. Columns (1)–(3) present unweighted estimates. Column (1) includes genre fixed effects and the sample indicator, column (2) adds log duration, and column (3) further adds the MIR structure controls (log chorus onset and log vocal onset). The coefficient on Post-PRA$_t$ is positive and statistically significant across specifications, indicating higher cross-sectional originality in the post-PRA period even after controlling for changes in song length and structure. Columns (4)–(6) replicate the same specifications weighting observations by Spotify's popularity index; the estimates are very similar when weighting by popularity, indicating that the increase in originality is not driven exclusively by less popular tracks.

Figure A11 replicates the baseline specification in Table 1 while estimating the model separately by year and by sampling frame (Top 100 versus a random draw from the Top 1000), including the full set of controls. The estimates show no evidence of a pre-trend in the pre-PRA years, but originality increases thereafter, with a clear upward shift by 2018 and especially by 2023. Importantly, the increase is visible in both samples.

Because the increase in cross-sectional originality is concentrated after the mid-2010s and coincides with the large-scale adoption of PRAs, a natural concern is that some other change occurring around the same time could be driving the pattern. For such an alternative

explanation to account for our results, it would need to involve a meaningful shift in demand, industry fundamentals, or listening concentration that also breaks around the mid-2010s. The available evidence does not support these coincident shifts. Figure A2 indicates that relative demand is stable: within the 2003 cohort, songs that were more popular at release tend to remain more popular today. Figure A3 shows that the decline in music prices began well before PRAs were introduced at scale and therefore does not line up with the post-2015 change in originality documented in Figure A11. Finally, Figure A6 shows no upward trend in stream concentration within Spotify's Top 200 chart (as measured by the Gini coefficient and the HHI), which alleviates concerns that a contemporaneous rise in winner-take-all listening mechanically reshapes chart composition.

6.3 Pairwise Cross-Sectional Originality

Our preferred approach uses the pairwise measure $\mathcal{K}(s, s')$ directly, rather than first aggregating it into the song-level average $\mathcal{K}(s)$. Working at the pair level has two advantages. First, it exploits the full set of within-year, within-genre pairwise comparisons (a 100×100 similarity matrix for each year, genre, and sampling frame), thereby substantially increasing precision relative to regressions based on song-level averages. Second, it allows us to control flexibly for song-specific heterogeneity by including fixed effects for *each* song appearing in a pair.

With these song fixed effects, the specification absorbs any song-level attribute, observable or unobservable, that affects its average similarity to other songs. This includes differences in duration, instrumentation, recording style, production quality, and whether the track is a live version, remix, or remaster. It also captures factors such as algorithmic playlist exposure or a higher share of streams coming from personalized recommendations. As a result, the identifying variation comes from systematic shifts in pairwise similarity across years, rather than from changes in observable song characteristics or other song-level confounds.

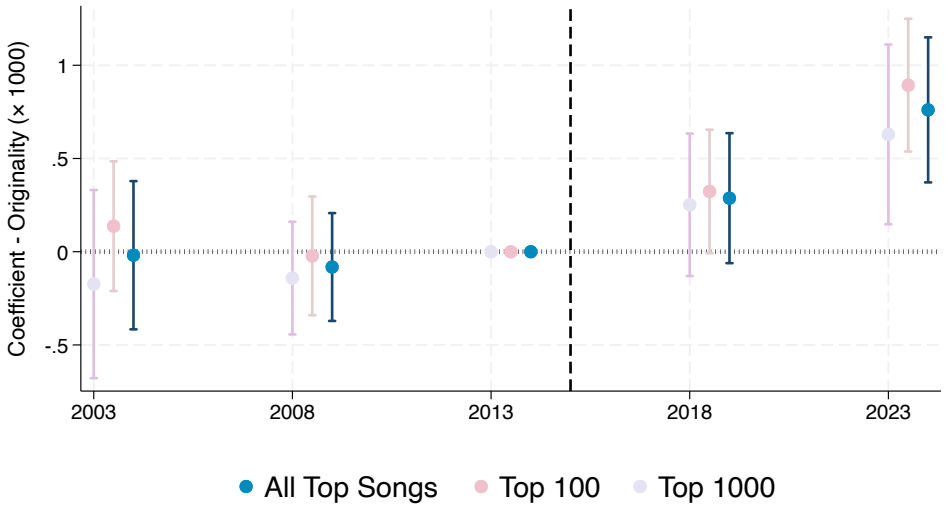
Let $\mathcal{K}_{ijgt}$ denote the pairwise originality between songs i and j in genre g and year t, computed as in Equation 5. We estimate the dyadic two-way fixed-effects regression

$$\mathcal{K}_{ijgt} = \sum_{\tau \in \{2003,2008,2018,2023\}} \beta_\tau \, \mathbb{1}\{t = \tau\} \; + \; \alpha_{ig} \; + \; \alpha_{jg} \; + \; \varepsilon_{ijgt}, \tag{8}$$

where the omitted category is $t = 2013$, and α_{ig} and α_{jg} are fixed effects for each song in the pair within genre g. We cluster standard errors at the genre-year level to allow for arbitrary dependence across all song pairs within a genre-year. This accommodates common genre-specific shocks and platform-level changes, including the introduction of PRAs, that

may jointly affect similarity among songs in the same genre and year.

Figure 8: Cross-Sectional Originality Over Time - Pairwise Estimation



Notes: The figure plots the estimated coefficients on year indicators from the pairwise specification for the Top 100 and Top 1000 samples in pop, rock, and country between 2008 and 2023. The regression includes song fixed effects and clusters standard errors at the genre-year level. Vertical bars denote 90% confidence intervals. The dashed vertical line marks 2015, corresponding to the introduction of large-scale personalized recommendation algorithms in the music industry.

Figure 8 plots the estimated coefficients $\{\beta_\tau\}$ from equation (8), separately for the Top 100 sample, the Top 1000 random sample, and the pooled sample.[18] The estimates are close to zero in the pre-PRA years and rise thereafter, with a clear increase by 2018 and a larger increase by 2023. Taken together, the pairwise specification reinforces the earlier results. The increase in cross-sectional originality is not driven by changes in observable song characteristics or by compositional shifts across samples. Instead, it reflects a systematic rise in differentiation across contemporaneous songs. The timing of the effect is consistent with the introduction of personalized recommendation algorithms and with the theory predictions.

6.4 Intertemporal Originality

The previous section documents that songs have become more differentiated from their contemporaries. Cross-sectional differentiation, however, does not by itself imply innovation: songs could become more dispersed within a year even if the musical frontier is not moving

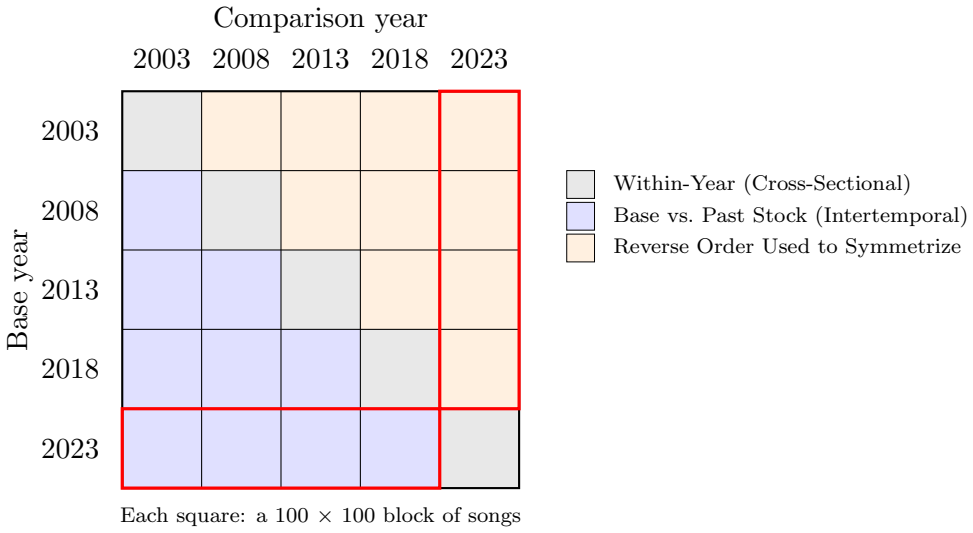
[18] In the pooled specification, we include a fixed effect for the sampling frame (Top 100 versus Top 1000) to account for systematic differences between the two samples.

relative to what came before. In this section, we study *intertemporal originality*, which asks whether new releases are becoming more distinct from the stock of previously released songs. This is the relevant object for the dynamics of musical innovation.

Constructing originality relative to the stock of past songs. For each genre and sampling frame, we pool the 100 songs observed in each of the five years in our audio sample (2003, 2008, 2013, 2018, and 2023), yielding 500 songs per genre-sample. We then compute a 500×500 matrix of pairwise originality measures $\mathcal{K}(s, s')$ (defined in equation (5)) across all songs in that genre-sample. This matrix contains the full set of within-year and cross-year comparisons. Intertemporal originality uses the cross-year elements that compare a *base* song to the *stock* of songs released earlier. For example, a song released in 2023 is compared to the other 400 songs from 2018, 2013, 2008, and 2003 (the stock available in our sample), while a song released in 2008 is compared only to the 100 songs from 2003.

Figure 9 illustrates how the intertemporal analysis uses cross-year blocks of the full 500×500 pairwise matrix, whereas the cross-sectional analysis relied only on within-year blocks.

Figure 9: Intertemporal Originality: Pairwise Matrix



Notes: The diagram is schematic and shown for a single genre and sampling frame. Pooling 100 songs from each of the five years (2003, 2008, 2013, 2018, and 2023) yields 500 songs, so the pairwise object is a 500×500 similarity matrix. Each square represents a 100×100 block of pairwise similarities between songs from two years, so the full matrix is composed of 25 year-by-year blocks. The diagonal blocks contain within-year comparisons, which are the objects used in the cross-sectional analysis. Intertemporal originality is constructed from the below-diagonal blocks that compare songs in a base year to the stock of songs released in earlier years. Because compression distances are order-dependent in finite samples, we compute similarities in both concatenation orders and symmetrize them; the corresponding reverse-order blocks appear above the diagonal. The red outlines highlight the comparison set for songs released in 2023 relative to the stock of songs from 2003–2018, in both directions.

Figure A12 compares the distribution of intertemporal originality for songs released in 2008 and 2023, where each song is evaluated relative to the stock of previously released songs. In particular, songs released in 2008 are compared to the 2003 stock, while songs released in 2023 are compared to the stock of songs from 2003, 2008, 2013, and 2018. The distribution for 2023 shifts to the right relative to 2008, indicating that more recent releases are more distinct from the existing catalog. A paired t-test confirms that the difference is statistically significant ($t = -3.13$, $p = 0.0019$). These results suggest that songs released in 2023 are more differentiated from the historical stock than songs released in 2008.

6.5 Reduced-Form Evidence

For each base song s released in year t and genre g, define its originality relative to the stock of past songs as

$$\mathcal{K}_s^{\text{stock}} \equiv \frac{1}{|\mathcal{S}_{g,<t}|} \sum_{s' \in \mathcal{S}_{g,<t}} \mathcal{K}(s, s'), \tag{9}$$

where $\mathcal{S}_{g,<t}$ denotes all songs in genre g released prior to t in our sample. We then estimate

$$\mathcal{K}_{igt}^{\text{stock}} = \beta \, \mathbb{1}\{\text{PostPRA}_t\} + \theta_g + \mathbf{X}'_{igt}\Gamma + \varepsilon_{igt}, \tag{10}$$

where θ_g are genre fixed effects. The indicator $\mathbb{1}\{\text{PostPRA}_t\}$ equals one for the post-PRA years in our audio sample (2018 and 2023) and zero for the earlier years (2008 and 2013).[19] As before, we pool both samples— Top 100 songs and Top 1000 songs—and include an indicator for the sampling frame in all specifications.

Table 2 reports the results. Columns (1)–(3) present unweighted estimates and add controls sequentially. Column (1) includes only genre fixed effects and the sampling-frame indicator. Column (2) adds the logarithm of song duration, and column (3) adds the full set of song controls (i.e. chorus onset time and vocal onset time). Columns (4)–(6) repeat the same specifications weighting observations by popularity. Across all specifications, the coefficient on $\mathbb{1}\{\text{PostPRA}_t\}$ is positive and statistically significant, indicating that songs released after the diffusion of PRAs are, on average, more distinct from the stock of earlier songs.

Appendix Figure A13 complements Table 2 by estimating equation (10) separately by year and sampling frame using the full set of controls. The year-by-year estimates mirror the table: little change before the mid-2010s and a clear increase in originality in 2018 and 2023 in both samples.

[19] The 2003 cohort has no prior stock within our sample, so it is excluded from the intertemporal regressions.

Table 2: Intertemporal Originality Relative to the Stock of Past Songs

Note: The dependent variable is intertemporal originality relative to the stock of earlier songs, $1000 \cdot \mathcal{K}^{\text{stock}}(s)$, where $\mathcal{K}^{\text{stock}}(s)$ is defined in equation (9). The sample pools the Top 100 songs and a random draw from the Top 1000 in each genre-year; all specifications include an indicator for the sampling frame. $\mathbb{1}\{\text{PostPRA}_t\}$ equals one for 2018 and 2023 and zero for 2008 and 2013. Columns (1)–(3) report unweighted estimates, while columns (4)–(6) weight observations by Spotify's popularity index. Song controls in columns (3) and (6) include the full set of covariates used in Section 6.2. Standard errors are clustered at genre × year.

	(1)	(2)	(3)	(4)	(5)	(6)
Originality						
Post-PRA	0.2048**	0.2300**	0.2142**	0.2109***	0.2423**	0.2219**
	(0.078)	(0.086)	(0.085)	(0.067)	(0.079)	(0.077)
Log Duration		0.2610	0.3346**		0.3030	0.4234**
		(0.155)	(0.131)		(0.177)	(0.145)
Observations	2,401	2,392	2,208	2,392	2,392	2,208
R-squared	0.015	0.016	0.015	0.017	0.019	0.019
Song Controls	N	N	Y	N	N	Y
Genre	Y	Y	Y	Y	Y	Y

6.6 Pairwise Intertemporal Originality

As in Section 6.6, we also estimate a pairwise specification that uses the full set of cross-year comparisons. This approach leverages the underlying 500 × 500 similarity matrix directly, and it controls flexibly for song-specific heterogeneity on both sides of each comparison. In particular, it absorbs any song-level attributes, observed or unobserved, that affect pairwise originality additively through song fixed effects for base songs and stock songs.

Let $\mathcal{K}_{ijgtt'}$ denote pairwise originality between a base song i released in year t and a stock song j released in year $t' < t$, within genre g. We estimate

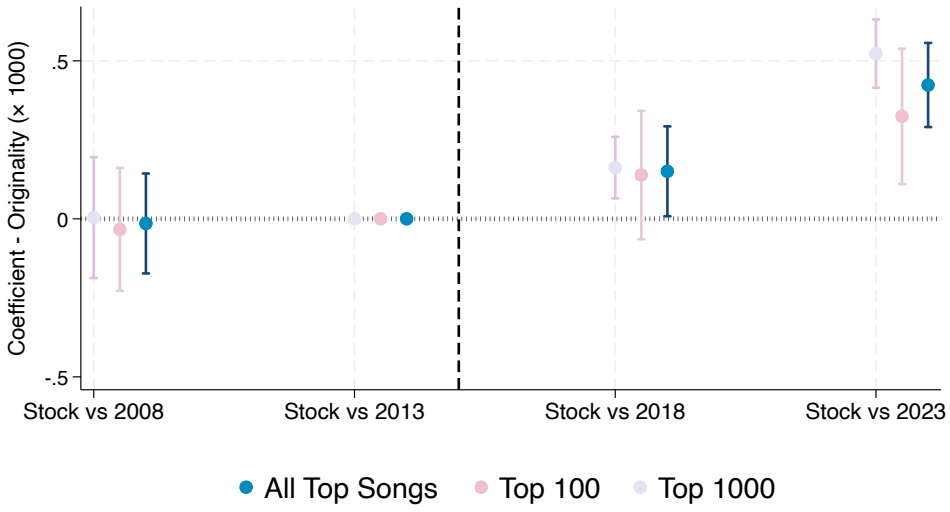
$$\mathcal{K}_{ijgtt'} = \sum_{\tau \in \{2008,2018,2023\}} \beta_\tau \, \mathbb{1}\{t = \tau\} + \alpha_{ig} + \lambda_{jgt'} + \varepsilon_{ijgtt'}, \tag{11}$$

where 2013 is the omitted base year. The fixed effects α_{ig} absorb all time-invariant characteristics of the base song within genre, while $\lambda_{jgt'}$ absorb all stock-song characteristics interacted with the stock year. Standard errors are clustered at the genre-by-base-year level.

Figure 10 summarizes the estimates from equation (11). Relative to the omitted base year (2013), the coefficients are close to zero in 2008, but rise in 2018 and increase further in 2023. The pattern is present in both sampling frames and the pooled sample.[20] This evidence

[20] In the pooled specification, sampling-frame differences are absorbed by the fixed effects. We define base-song fixed effects at the level of track (base) × genre × sampling frame, and stock-song fixed effects at the

Figure 10: Intertemporal Originality Relative to the Stock of Past Songs: Pairwise



Notes: The figure plots coefficients on base-year indicators from equation (11), separately for the Top 100 sample, the Top 1000 sample, and the pooled sample. For each base year, base songs are compared to the stock of songs released in earlier years within the same genre; for example, a song released in 2023 is compared to the songs from 2018, 2013, 2008, and 2003 in the same genre-sample. Vertical bars denote 90% confidence intervals based on standard errors clustered at genre × base year. The vertical line marks 2015, corresponding to the period in which PRAs were introduced at scale.

complements the cross-sectional results by showing that the increased differentiation among contemporaneous songs is accompanied by greater distinctiveness relative to the historical catalog. Taken together, the results point to an upward shift in musical innovation in the post-PRA period.

7 Discussion on Extensions

Peer Effects Individuals' choices can be shaped by the consumption of their peers. Given the paper's focus, however, our main interest is not in peer effects per se, but in whether the platform's recommendation algorithm amplifies—or mitigates—them, and how this interaction affects the endogenous production of songs. Thus, we can incorporate peer effects straightforwardly by introducing peer influence directly into consumers' utility functions and modifying the platform's problem, so that it internalizes the presence of peer effects. In this extended framework, the platform would tend to recommend songs that appeal to the average

level of track (stock) × genre × sampling frame. Because these fixed effects include the sampling-frame indicator, they allow all song fixed effects to differ between the Top 100 and Top 1000 samples.

listener more strongly than in the absence of peer effects, but the consumers' problem would remain unchanged except through their utility function. Therefore, we focus on peer effects only insofar as they interact with the platform, abstracting from broader peer dynamics that would complicate the analysis without altering its main insights.

Stock of Previous Songs Songs listened to in a given period may include both new releases and, plausibly, songs produced in earlier periods. A simple way to acknowledge this coexistence in our model is to reinterpret λ_i as the fraction of time devoted to listening to *new* songs through the algorithm. Under this interpretation, the remaining fraction of time—spent on older songs or on music not mediated by the algorithm—is abstracted away so that we can focus on how PRAs change the production of new songs.

Repeated Consumption Repeated consumption can be informative in our setting insofar as it serves as a proxy for intensity: How much an individual enjoys a song may be reflected in how often they listen to it. Our model, however, already embeds this notion of intensity through the payoff structure: a higher payoff aligns with higher enjoyment and intensity. Moreover, in the theory, these payoffs also determine producers' compensation, just as in reality platforms reward artists depending on the number of times their songs are streamed.

8 Conclusion

Understanding the impact of artificial intelligence on creative output requires a general equilibrium framework. Personalized recommendation algorithms (PRAs) do not simply match consumers with products: by shaping the set of songs or goods that consumers face, they also influence how producers choose what to create. Our model shows that these forces interact in equilibrium. Each consumer is exposed to a narrower set of products tailored to her predicted tastes, but producers expand the range of characteristics they are willing to supply in order to reach different segments of the market. This highlights the importance of accounting for producer responses when evaluating the effects of algorithmic recommendation systems.

We validate the model with evidence from the music industry, where PRAs have been widely adopted since the mid-2010s. Using tools from Music Information Retrieval applied directly to raw audio files to summarize their high-dimensional attributes, we document that songs have become less similar to one another over time following the introduction of PRAs. The increase in originality after the adoption of PRAs is consistent with the model's prediction that recommendation algorithms generate broader creative dispersion at the market level.

Together, the theoretical and empirical results suggest that PRAs, while narrowing exposure for individual consumers, have the aggregate effect of expanding creative output. Far from homogenizing cultural production, recommendation algorithms can foster diversity in equilibrium by reshaping the incentives of producers. Moreover, PRAs are welfare enhancing insofar as they improve the matching between songs and consumers, and therefore reduce misallocation.

Appendix

Omitted Proofs

Proof of Proposition 1

First, note that from the producers perspective, given the recommendation mechanism and the platform's information set $\mathcal{G}$, only songs that take form of $\alpha\mu_i(\mathcal{G})$ for some i will give a positive profit. Therefore, optimal songs must take the form of $\alpha\mu_i(\mathcal{G})$ for some $\alpha \geq 0$ and for some i. For each i, given information set $\mathcal{F}$ and Q_i, if producer z wants to target i, the optimal song must be $\alpha_i^*\mu_i(\mathcal{G})$, where α_i^* solves

$$\max_{\alpha\geq 0}\left[Q_i\mathbb{E}[\langle\theta_i,\alpha\{\mu_i(\mathcal{G})\}_{i=1}^N\rangle \mid \mathcal{F},\mu_i(\mathcal{G})] - z\frac{\|\alpha\mu_i(\mathcal{G})\|^2}{2}\right],$$

since $\mathcal{F}\subseteq\mathcal{G}$ and since $\{\theta_i\}_{i=1}^N$ is independent, $\mathbb{E}[\theta_i \mid \mathcal{F},\mu_i(\mathcal{G})] = \mu_i(\mathcal{G})$. Thus, the producer's problem becomes

$$\max_{\alpha\geq 0}\left[Q_i\langle\mu_i(\mathcal{G}),\alpha\mu_i(\mathcal{G})\rangle - z\frac{\|\alpha\mu_i(\mathcal{G})\|^2}{2}\right].$$

First order condition implies that

$$\alpha_i^* = \frac{Q_i}{z},$$

and thus the optimal song tailored to consumer i is

$$\frac{Q_i}{z}\mu_i(\mathcal{G}),$$

which in turn gives an expected payoff of

$$\frac{Q_i^2}{z}\|\mu_i(\mathcal{G})\|^2.$$

In equilibrium, all producers must be indifferent among targeting all consumers, since otherwise some consumers won't be targeted and there would be profitable deviation for a producer to arbitrage and target them. Therefore, in equilibrium, it must be that

$$\frac{Q_i^{*2}}{z}\|\text{Proj}\left[\mu_i(\mathcal{F}) \mid \mu_i(\mathcal{G})\right]\|^2 = \frac{Q_j^{*2}}{z}\|\text{Proj}\left[\mu_j(\mathcal{F}) \mid \mu_j(\mathcal{G})\right]\|^2$$

for all i,j. Multiplying z on both sides, and take the nonnegative square root, we have

$$Q_i^*\|\mu_i(\mathcal{G})\| = Q_j^*\|\mu_j(\mathcal{G})\|,$$

which determines Q^* given the posterior beliefs of both the platform and the producers, up to a constant. Given these Q^*, each producer is indifferent among targeting each consumer. That is, each producer z is indifferent among producing

$$\{Q_1^*\mu_1(\mathcal{G}), \ldots, Q_N^*\mu_N(\mathcal{G})\} .$$

Thus, the equilibrium strategy must be supported on

$$\{Q_1^*\mu_1(\mathcal{G}), \ldots, Q_N^*\mu_N(\mathcal{G})\} .$$

Meanwhile, by the definition of P_i^* and σ^*, the producers' randomize among these songs in a way that is consistent with Q_i^*. That is,

$$\text{supp}(\sigma^*(z, Q^*) \subseteq \{Q_1^*\mu_1(\mathcal{G}), \ldots, Q_N^*\mu_N(\mathcal{G})\}$$

for all $z \in \mathbb{R}_+$, and, moreover,

$$\int_0^\infty \sigma^*(S_i^{R^{\text{exploit}}}|z, Q^*)F(\mathrm{d}z) = P_i^* ,$$

and hence

$$\frac{\lambda}{\int_0^\infty \sigma^*(S_i^{R^{\text{exploit}}}|z, Q^*)F(\mathrm{d}z)} = Q_i^* ,$$

as desired. □

Proof of Proposition 2

Given information set $\mathcal{F}$ and Q_i, if producer z wants to target i, since $v_i(\mathcal{G})$ does not contain additional information about θ_i relative to $\mathcal{F}$, the optimal song is given by solving

$$\max_{\alpha \geq 0} \left[Q_i \langle \mu_i(\mathcal{F}), \alpha v_i(\mathcal{G}) \rangle - z \frac{\|\alpha v_i(\mathcal{G})\|^2}{2} \right] .$$

First order condition implies that the optimal α is

$$\frac{Q_i}{z} \frac{\langle \mu_i(\mathcal{F}), v_i(\mathcal{G}) \rangle}{\|v_i(\mathcal{G})\|^2} ,$$

which gives maximized expected payoff of

$$Q_i \langle \mu_i(\mathcal{F}), \alpha v_i(\mathcal{G}) \rangle - \alpha^2 \frac{\|v_i(\mathcal{G})\|^2}{2} .$$

In equilibrium, all producers must be indifferent among targeting all consumers, since otherwise some consumers won't be targeted and there would be profitable deviation for a producer to arbitrage and target them. Therefore, in equilibrium, it must be that

$$Q_i^{*2}\frac{\langle \mu_i(\mathcal{F}), v_i(\mathcal{G})\rangle^2}{z\|\mu_i(\mathcal{G})\|^2} = Q_j^{*2}\frac{\langle \mu_j(\mathcal{F}), v_i(\mathcal{G})\rangle^2}{z\|\mu_j(\mathcal{G})\|^2}$$

for all i, j. Multiplying z on both sides, and take the nonnegative square roots, we have

$$Q_i^*\|\mathrm{Proj}\left[\mu_i(\mathcal{F}) \mid v_i(\mathcal{G})\right]\| = Q_j^*\|\mathrm{Proj}\left[\mu_j(\mathcal{F}) \mid v_j(\mathcal{G})\right]\|\,,$$

which determines Q^* given the posterior beliefs of both the platform and the producers, up to a constant. Given these Q^*, each producer is indifferent among targeting each consumer.

Therefore, let

$$P_i^* := \frac{\|\mathrm{Proj}\left[\mu_i(\mathcal{F}) \mid \right.}{\sum_{i=1}^N \|\mathrm{Proj}\left[\mu_j(\mathcal{F}) \mid v_j(\mathcal{G})\right]\|}\,,$$

and let

$$Q_i^* := \frac{\lambda}{P_i^*}\,,$$

and

$$\sigma^*(z, Q^*; \mathcal{F}) := \sum_{i=1}^N P_i^*\delta_{\left\{\frac{Q_i^{*2}}{z}\mathrm{Proj}\left[\mu_i(\mathcal{F})|v_i(\mathcal{G})\right]\right\}}\,.$$

It then follows that, given $(Q_i^*)_{i=1}^N$, producers are indifferent among producing

$$\{Q_1^*\mathrm{Proj}\left[\mu_1(\mathcal{F}) \mid v_1(\mathcal{G})\right], \ldots, Q_N^*\mathrm{Proj}\left[\mu_N(\mathcal{F}) \mid v_N(\mathcal{G})\right]\}\,,$$

and assigns probability P_i^* to $Q_i^*\mathrm{Proj}\left[\mu_i(\mathcal{F}) \mid v_i(\mathcal{G})\right]$ for all i, which in turn gives

$$\int_0^\infty \sigma^*(S_i^{R^{\text{explore}}}|z, Q^*; \mathcal{F})F(\mathrm{d}z) = P_i^*\,,$$

and hence

$$\frac{\lambda}{\int_0^\infty \sigma^*(S_i^{R^{\text{explore}}}|z, Q^*; \mathcal{F})F(\mathrm{d}z)} = Q_i^*\,.$$

As a result, (Q^*, σ^*) is an equilibrium. □

References

Acemoglu, D., 2002. Directed technical change. The Review of Economic Studies 69, 781–809. doi:10.1111/1467-937X.00226.

Acemoglu, D., Linn, J., 2004. Market size in innovation: Theory and evidence from the pharmaceutical industry. The Quarterly Journal of Economics 119, 1049–1090. doi:10.1162/0033553041502144.

Agrawal, A., Gans, J., Goldfarb, A., 2018. Prediction Machines: The Simple Economics of Artificial Intelligence. Harvard Business School Press.

Aguiar, L., Waldfogel, J., 2018. Quality predictability and the welfare benefits from new products: Evidence from the digitization of recorded music. Journal of Political Economy 126, 492–524. doi:10.1086/696229.

Aguiar, L., Waldfogel, J., 2021. Platforms, power, and promotion: Evidence from spotify playlists. The Journal of Industrial Economics 69, 653–691. doi:10.1111/joie.12263.

Anderson, A., Maystre, L., Mehrotra, R., Anderson, I., Lalmas, M., 2020. Algorithmic effects on the diversity of consumption on spotify, in: Proceedings of The Web Conference 2020, pp. 2155–2165. doi:10.1145/3366423.3380281.

Bello, J.P., 2011. Measuring structural similarity in music. IEEE Transactions on Audio, Speech, and Language Processing 19, 2013–2025.

Benner, M.J., Waldfogel, J., 2016. The song remains the same? technological change and positioning in the recorded music industry. Strategy Science 1, 129–147. doi:10.1287/stsc.2016.0012.

Cilibrasi, R., Vitányi, P.M., 2005. Clustering by compression. IEEE Transactions on Information theory 51, 1523–1545.

Datta, H., Knox, G., Bronnenberg, B.J., 2018. Changing their tune: How consumers' adoption of online streaming affects music consumption and discovery. Marketing Science 37, 5–21. doi:10.1287/mksc.2017.1051.

Fleder, D., Hosanagar, K., 2009. Blockbuster culture's next rise or fall: The impact of recommender systems on sales diversity. Management Science 55, 697–712. doi:10.1287/mnsc.1080.0974.

Gans, J., 2019. The Economics of Artificial Intelligence: An Agenda. University of Chicago Press.

Goto, M., 2006. A chorus section detection method for musical audio signals and its application to a music listening station. IEEE Transactions on Audio, Speech, and Language Processing 14, 1783–1794.

Holtz, D., Carterette, B., Chandar, P., Nazari, Z., Cramer, H., Aral, S., 2020. The engagement-diversity connection: Evidence from a field experiment on spotify. doi:10.48550/arXiv.2003.08203, arXiv:2003.08203.

Lee, D., Hosanagar, K., 2019. How do recommender systems affect sales diversity? a cross-category investigation via randomized field experiment. Information Systems Research 30, 239–259. doi:10.1287/isre.2018.0800.

McFee, B., Raffel, C., Liang, D., Ellis, D.P., McVicar, M., Battenberg, E., Nieto, E., 2015. librosa: Audio and music signal analysis in python, in: Proceedings of the 14th Python in Science Conference (SciPy 2015), pp. 18–25.

Schnidman, M.E., 2025. Playing to the algorithm: How spotify's recommendations shape music production. URL: https://mschnidman.github.io/files/Schnidman_PTA_20250825.pdf. unpublished manuscript, August 25, 2025.

Stöter, F.R., Uhlich, S., Liutkus, A., Mitsufuji, Y., 2019. Open-unmix-a reference implementation for music source separation. Journal of Open Source Software 4, 1667.

Waldfogel, J., 2017. How digitization has created a golden age of music, movies, books, and television. Journal of Economic Perspectives 31, 195–214. doi:10.1257/jep.31.3.195.

Online Appendix for

From Code to Composition:

Personalized Recommendation and its Impact on Creativity

March 2026

David Argente
Yale University

Maria Trejos
Yale University

Diana Van Patten
Yale Univeristy

Kai Hao Yang
Yale Univeristy

Table of Contents

Figures and Tables **1**

A.1 Spotify API Data: Validation 9

Audio Content Analysis: Technical Details **9**

B.1 Chromagram 9

B.2 Similarity Across Songs 11

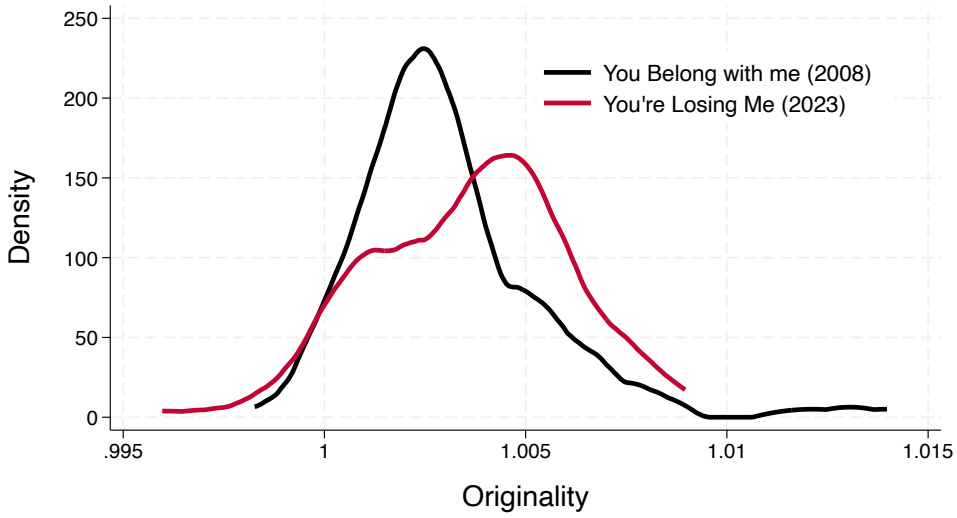
B.3 Chorus Detection 13

B.4 Vocal Onset Detection 14

Appendix for Online Publication

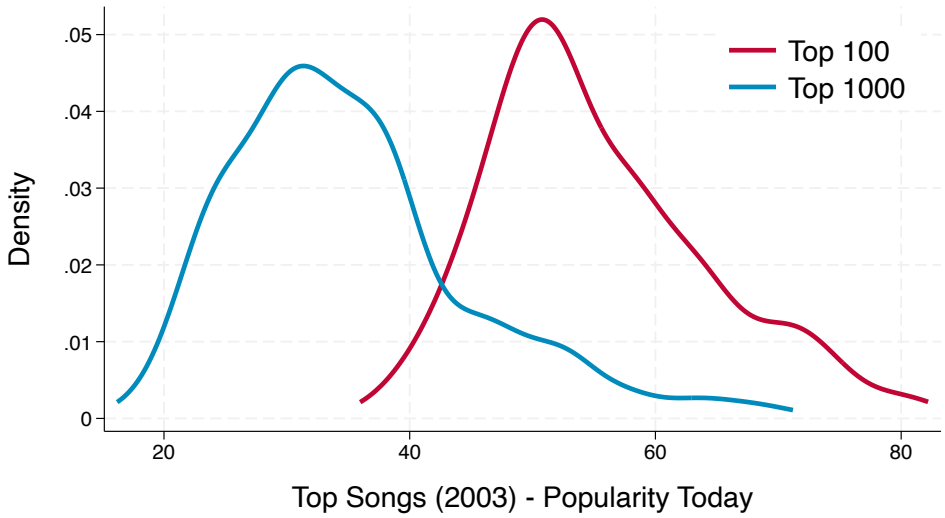
A Figures and Tables

Figure A1: Distribution of Originality Across Songs: Taylor Swift 2008 and 2023



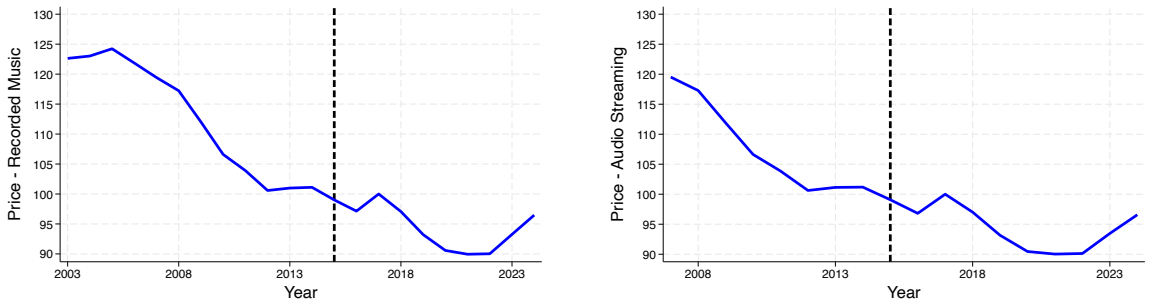
Notes: The figure shows the distribution of pairwise originality for two Taylor Swift songs– "You Belong With Me" (2008) and "You're Losing Me" (2023)—relative to the top 100 pop songs of those years.

Figure A2: Popularity Over Time



Notes: The figure shows the distribution of 2023 popularity for the top 100 and top 1000 songs released in 2003. Spotify's popularity metric, an integer index between 0 and 100 from the Spotify Web API that reflects the relative popularity of a track and is based primarily on total streams with greater weight placed on recent listening activity.

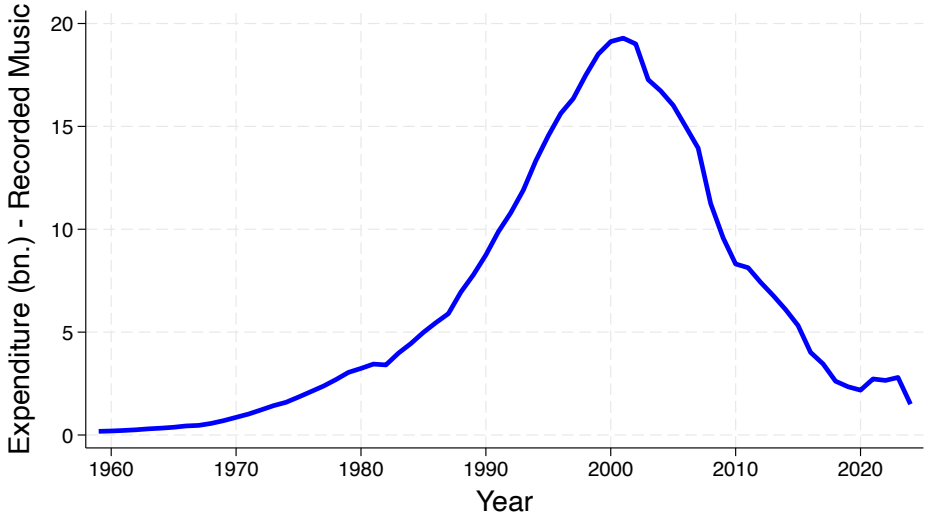
Figure A3: Recorded Music versus Audio Streaming: Prices



(a) Recorded Music (b) Audio Streaming

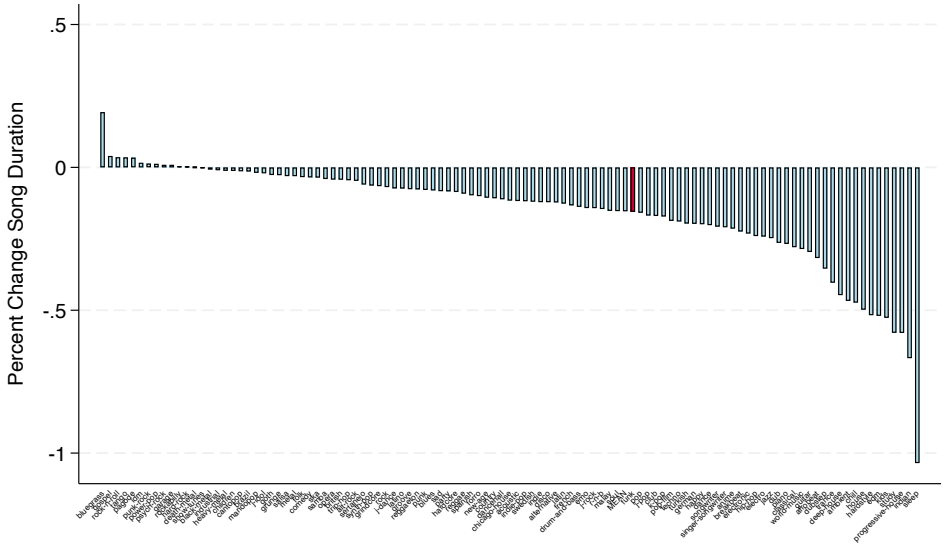
Notes: The figure shows prices for both recorded music and audio streaming. Recorded music includes audio discs, tapes, vinyl, and digital downloads. Audio streaming includes streaming and radio services such as satellite radio. The dashed line marks 2014, which coincides with the large-scale adoption of personalized recommendation algorithms. Source: Personal Consumption Expenditures (PCE), Bureau of Economic Analysis (BEA).

Figure A4: Music Industry Revenues



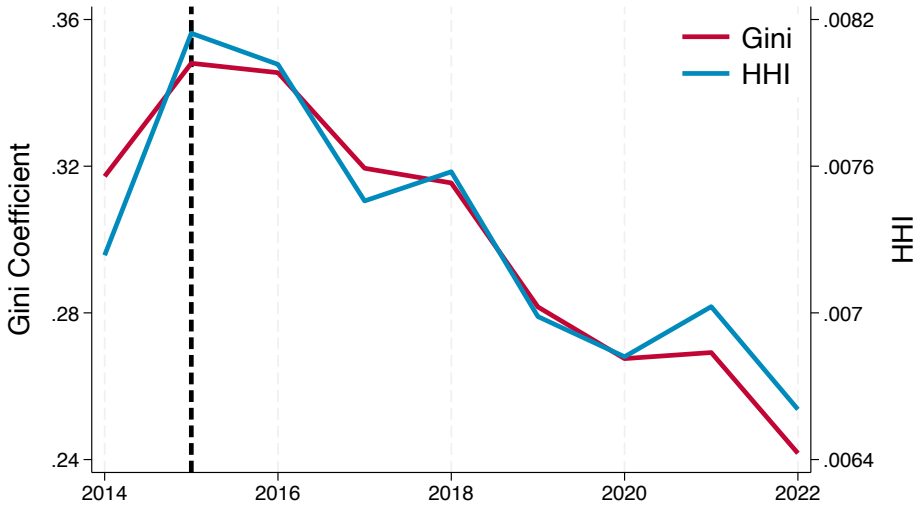
Notes: The figure shows total expenditures in recorded music from 1959 to 2024. Recorded music includes audio discs, tapes, vinyl, and digital downloads. Source: Personal Consumption Expenditures (PCE), Bureau of Economic Analysis (BEA).

Figure A5: Percent Change in Song Duration: All Genres



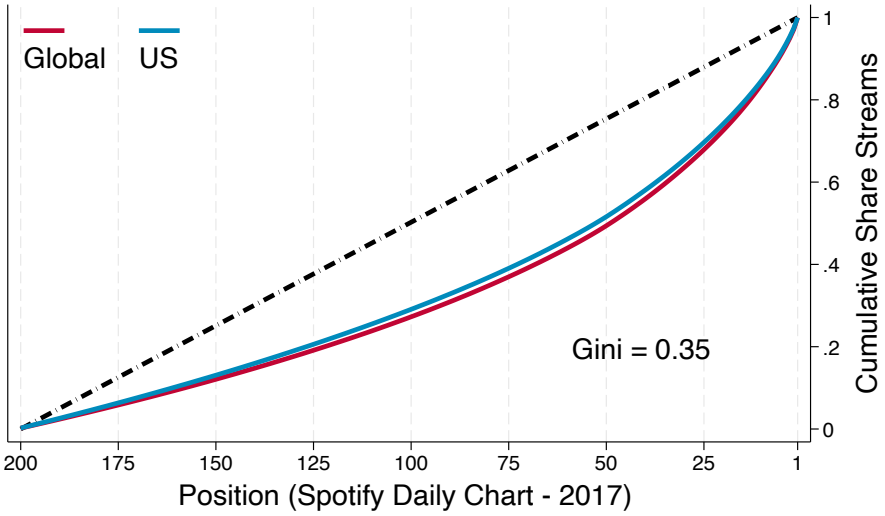
Note: The figure shows the percent change in the average duration of songs from 2008 to 2023 across each of Spotify's music genres. Data were collected from the top 1000 songs per genre for each year, using the songs' release dates.

Figure A6: Gini and HHI: Top 200 Songs



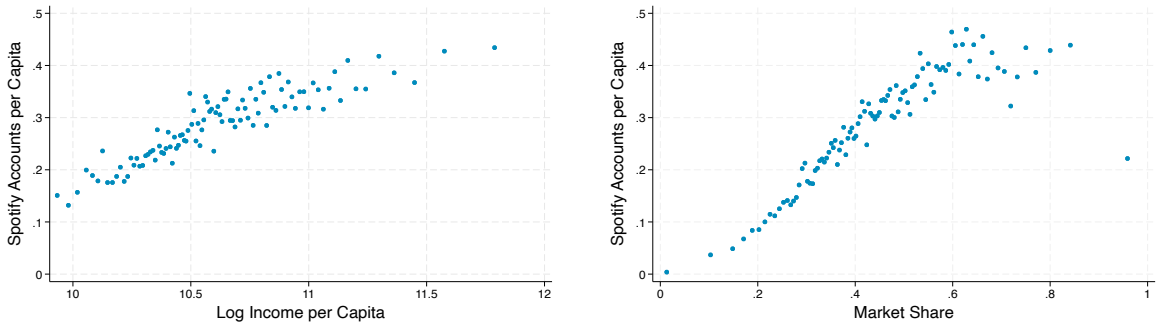
Notes: The figure plots annual concentration measures for streams within Spotify's Daily Top 200 Chart. For each year, we aggregate daily stream counts across the year and compute stream shares within the Top 200; the Herfindahl–Hirschman Index (HHI) is the sum of squared shares, and the Gini coefficient is computed from the implied Lorenz curve. The dashed vertical line marks 2015.

Figure A7: Gini of Streaming Distribution: Top 200 Songs



Notes: The figure uses Spotify Daily Chart data for 2017. For each region (Global and United States), we aggregate daily streams across all days in 2017 for each chart position (1–200), and then compute the Lorenz curve as the cumulative share of total streams accounted for by the top-k positions. The dashed 45-degree line is the equality benchmark. The x-axis is shown in reverse order so that position 1 (most-streamed) appears on the right.

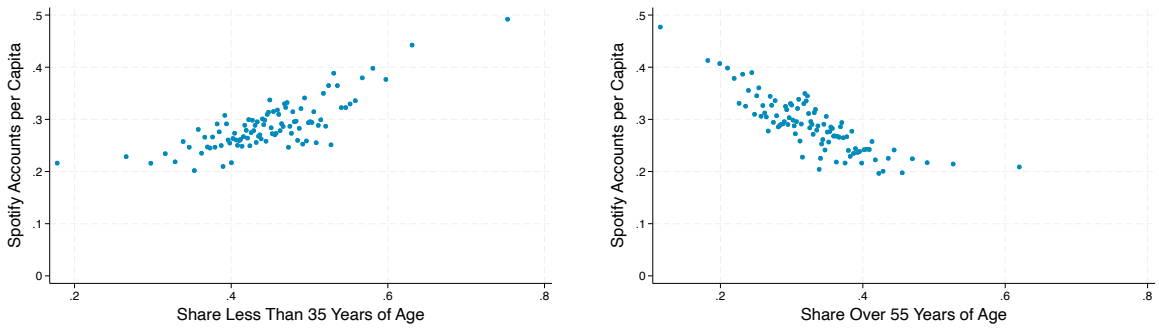
Figure A8: Spotify's Penetration and Market Share



(a) Income (b) Market Share

Notes: The figures use zipcode-level transaction data to construct measures of Spotify penetration and market share. Spotify accounts per capita are imputed from debit card transactions scaled by an assumed annual subscription cost of $99 and adjusted for the share of premium users and debit card coverage. Market share is defined as Spotify transaction value divided by total streaming-related transaction value within the zipcode. Observations are weighted by zipcode population and binned into 100 equal-sized bins. The source is the MBHS3 Data, which includes de-identified credit and debit card transaction data covering over 180 million consumers in the U.S.

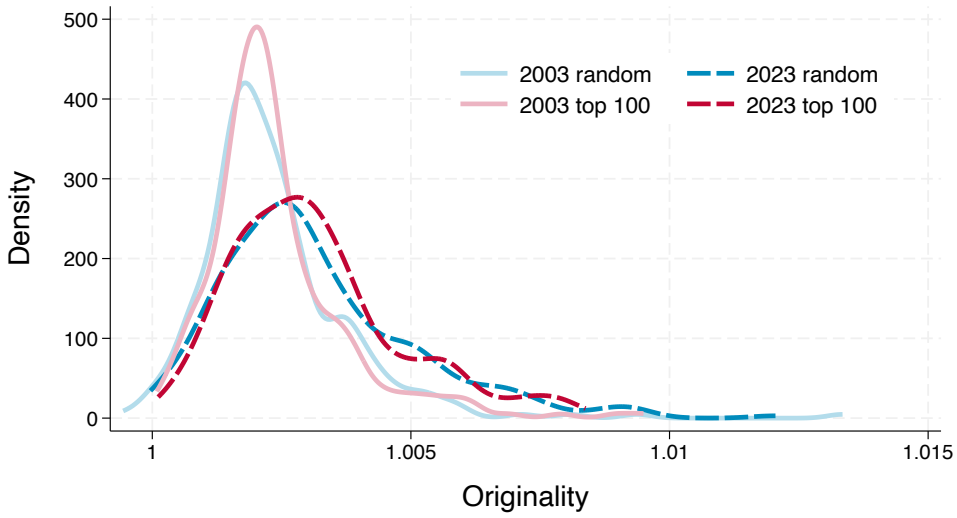
Figure A9: Spotify's Penetration by Age



(a) Under 35 (b) Over 55

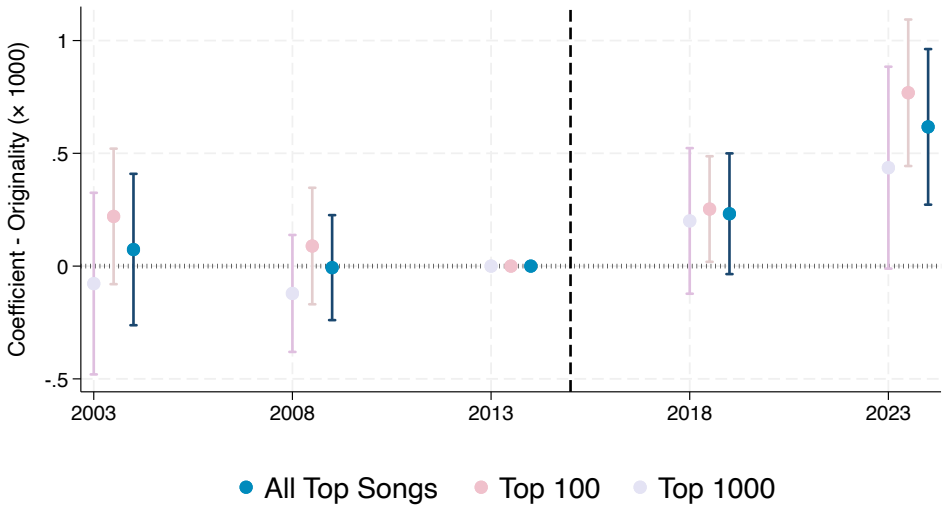
Notes: The figures plot zipcode-level Spotify accounts per capita against demographic composition. Spotify penetration is constructed from debit card transaction data, scaled by assumed annual subscription costs and adjusted for premium user share and debit card coverage. Age shares are computed using zipcode demographic data. Observations are weighted by population and displayed using 100 equal-sized bins. The source is the MBHS3 Data, which includes de-identified credit and debit card transaction data covering over 180 million consumers in the U.S.

Figure A10: Distribution of Originality Across Songs: 2003-2023 (Top 100 and Top 1000)



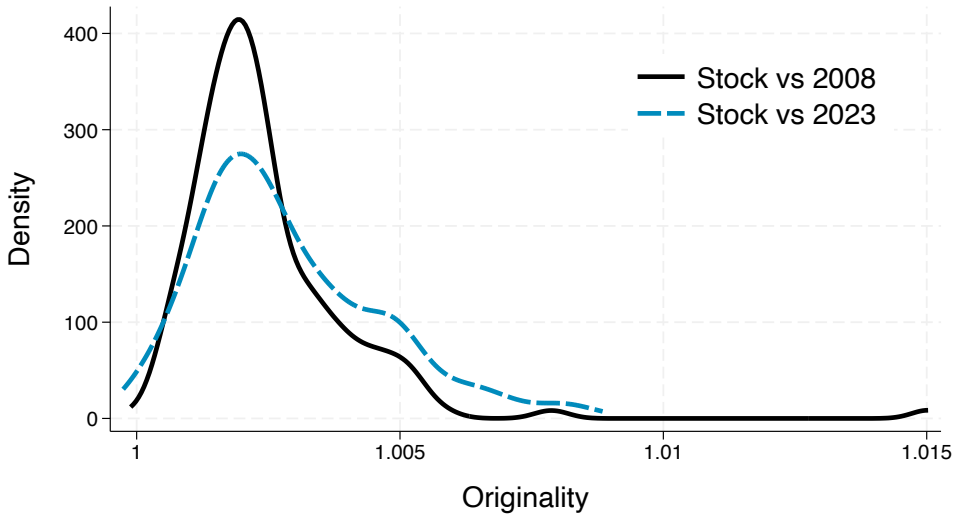
Notes: The figure shows the distribution of originality for 2003 and 2023, after controlling for song genre. We compute the pairwise similarity among the songs and then average these values to obtain a song-specific originality score. The sampls consists of the top 100 songs in the pop, rock, and country genres and a random sample of 100 songs from the top 1000 songs in the same genres.

Figure A11: Cross-Sectional Originality Over Time



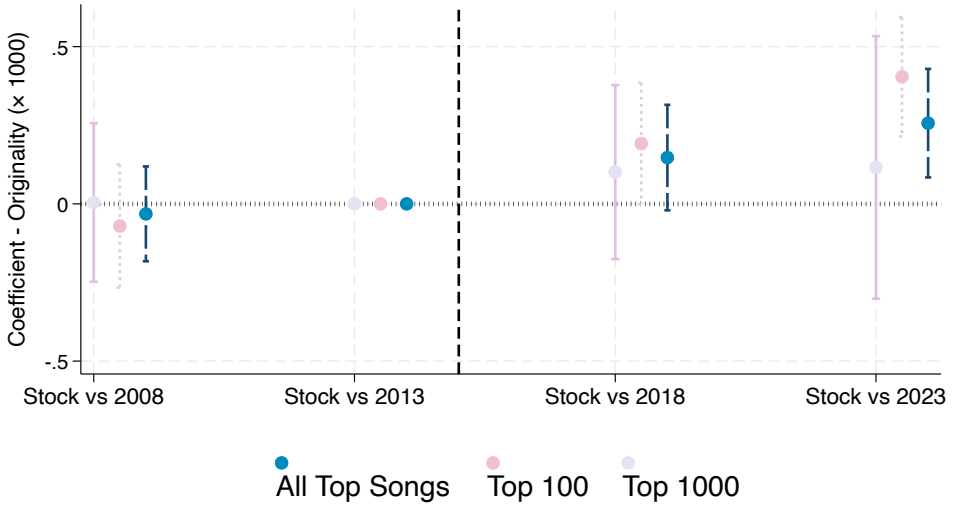
Notes: The figure plots the estimated coefficients on year indicators from the cross-sectional originality regression for the Top 100 (red) and Top 1000 (blue) songs in pop, rock, and country between 2008 and 2023. Estimates control for both genre and track duration. Estimates are shown for the Top 100 sample, the Top 1000 sample, and the pooled sample. Vertical bars denote 90% confidence intervals based on standard errors clustered at genre × base year. The dashed vertical line marks 2015, corresponding to the introduction of large-scale personalized recommendation algorithms.

Figure A12: Distribution of Intertemporal Originality: 2008 vs. 2023



Notes: The figure plots the distribution of intertemporal originality for songs released in 2008 and 2023. For each base year, intertemporal originality is defined as the average pairwise originality relative to the stock of previously released songs within the same genre. Thus, songs released in 2008 are compared to the 2003 stock, while songs released in 2023 are compared to the stock from 2003, 2008, 2013, and 2018. The sample consists of the top 100 songs in the pop, rock, and country genres.

Figure A13: Intertemporal Originality Relative to the Stock of Past Songs



Notes: The figure plots coefficients on base-year indicators from the intertemporal specification in which each base-year song is compared to the stock of songs released in earlier years within the same genre. For each base year, base songs are compared to the stock of songs released in earlier years within the same genre; for example, a song released in 2023 is compared to the songs from 2018, 2013, 2008, and 2003 in the same genre-sample. Coefficients are normalized to zero for the 2013 base year. Estimates are shown for the Top 100 sample, the Top 1000 sample, and the pooled sample. Vertical bars denote 90% confidence intervals based on standard errors clustered at genre × base year. The dashed vertical line marks 2015, corresponding to the introduction of large-scale personalized recommendation algorithms.

A.1 Spotify API Data: Validation

While the stream count for individual tracks is not directly accessible from the Spotify Web API, the platform provides a popularity score that reflects a track's relative engagement. This measure, however, does not by itself guarantee that our dataset includes only the most widely streamed and recognized tracks. To assess the completeness of our dataset, we compare it against Spotify's official charting data, which records the 200 most-streamed songs each week.[21]

We compiled weekly chart data for two representative years, 2019 and 2024, and constructed an independent dataset of the most widely streamed tracks. Using Spotify ID as a unique identifier, we merged the chart dataset with our track dataset to assess coverage. After accounting for identical artist and track names, the match rate reached 96.5 percent, and incorporating album-based matching further improved coverage to 97.2 percent. These results confirm that our dataset effectively captures the vast majority of highly popular songs on the platform.

B Audio Content Analysis: Technical Details

B.1 Chromagram

The chromagram captures, at each moment, the relative intensity of each of the 12 notes–similar to a simplified form of sheet music–while remaining robust to variations in timbre, loudness, and tuning. To generate this representation from an audio signal, we adapt a pipeline based on PyChorus's `create_chroma` function, which in turn wraps Librosa's `librosa.feature.chroma_stft`.[22] Below we describe in detail each step.

Let $y[n]$ be a discrete-time audio signal of length L with $n = 0, 1, \ldots, L-1$, sampled at f_s (Hz) samples per second. Let N be the frame length (e.g. 2048) and H be the hop length (hop length, e.g. 512). We slice $y[n]$ into $M \approx \lfloor (L-N)/H \rfloor + 1$ overlapping frames.

To reduce *spectral leakage*–spurious energy in neighboring frequency bins caused by abrupt frame edges–each frame is multiplied by the Hann window

$$w[n] = 0.5\Big(1 - \cos\big(\tfrac{2\pi n}{N-1}\big)\Big), \quad n = 0, \ldots, N-1.$$

This window satisfies $w[0] = w[N-1] = 0$, rising smoothly to 1 at the center. The windowed

[21] Spotify's official charts are available at https://charts.spotify.com.

[22] PyChorus is an open-source library for automatically detecting choruses and other salient sections in music; its core algorithm builds on the work of Goto (2006) on chorus detection.

frame for frame index m is

$$x_m[n] = y[n + mH]\; w[n], \quad n = 0, \ldots, N-1.$$

Then, a discrete Fourier transform of each windowed frame is computed:

$$X_m[k] = \sum_{n=0}^{N-1} x_m[n]\; e^{-j\,2\pi\,\frac{k}{N}\,n}, \quad k = 0, 1, \ldots, N-1.$$

Here k is the frequency-bin index, corresponding to nominal frequency $f_k = \frac{k}{N}\, f_s$. Because $x_m[n]$ is real, only $k = 0, \ldots, \frac{N}{2}$ are needed.

The power spectrogram is formed by squaring the magnitude of each STFT coefficient:

$$S_m[k] = |X_m[k]|^2, \quad k = 0, \ldots, \tfrac{N}{2}.$$

The quantity $S_m[k]$ measures the *energy* (intensity) at frequency f_k. Unlike raw amplitude $|X_m[k]|$, energy adds linearly across independent components, aligns better with perceptual loudness, and is always nonnegative.

Since recordings may be slightly sharp or flat relative to standard pitch (A4 = 440 Hz). Librosa estimates a global tuning offset Δ (in semitones) and corrects each bin frequency:

$$\tilde{f}_k = 2^{\Delta/12}\; f_k.$$

A semitone, the smallest interval used in classical Western music which is equal to a twelfth of an octave, is the frequency ratio $2^{1/12}$. There are 12 semitones per octave, indexed $p = 0, 1, \ldots, 11$. These indices are assign in chromatic order as

$$\mathrm{C} = 0,\ \mathrm{C\#} = 1,\ \ldots,\ \mathrm{A} = 9,\ \ldots,\ \mathrm{B} = 11,$$

so that A4 (the A above middle C tuned to 440 Hz) has index $p_{\mathrm{A4}} = 9$. The center frequency of pitch class p is then defined by

$$c_p \;=\; 440\ \mathrm{Hz} \;\times\; 2^{\frac{p - p_{\mathrm{A4}}}{12}},$$

which ensures $c_{p_{\mathrm{A4}}} = 440$ Hz and places the other semitone centers equally spaced on a logarithmic (pitch) scale. Then, a soft triangular filterbank $W \in \mathbb{R}^{12 \times (1+N/2)}$ is constructed to map each STFT bin into the 12 semitone classes. The weight $W_{p,k}$ assigned to bin k for

pitch class p is defined by

$$W_{p,k} = \max\Big\{1 - \frac{\big|\log_2(\tilde{f}_k/c_p)\big|}{1/12},\, 0\Big\},$$

where $\tilde{f}_k$ is the tuning-corrected frequency of bin k and c_p is the center frequency of semitone p. This choice ensures that $W_{p,k} = 1$ when $\tilde{f}_k = c_p$, that the weight decays linearly in the logarithmic frequency domain as $\tilde{f}_k$ moves away from c_p, and that $W_{p,k} = 0$ once $\big|\log_2(\tilde{f}_k/c_p)\big| \geq 1/12$ (i.e. beyond one semitone). In this way, only frequency bins within ± 1 semitone of each semitone center contribute to the corresponding chroma bin, with closer bins contributing more strongly and more distant bins gradually dropping out.

Next, for each frame m, weighted energies are summed into the 12 pitch classes:

$$C_m[p] = \sum_{k=0}^{N/2} W_{p,k}\, S_m[k], \quad p = 0, \ldots, 11.$$

Frames with very low total chroma energy are treated as silent. The total energy in frame m:

$$E_m = \sum_{p=0}^{11} C_m[p].$$

If E_m falls below the small threshold $\varepsilon = 10^{-6}$, the entire chroma vector is zeroed out. Each remaining chroma vector is normalized to unit Euclidean length:

$$\hat{\mathbf{C}}_m = \frac{\mathbf{C}_m}{\|\mathbf{C}_m\|_2 + \varepsilon}, \quad \|\mathbf{C}_m\|_2 = \sqrt{\sum_{p=0}^{11} C_m[p]^2}.$$

After this normalization, each component $\hat{C}_m[p]$ lies in $[0, 1]$ and satisfies $\sum_{p=0}^{11} \hat{C}_m[p]^2 = 1$.

Lastly, all M normalized vectors are assembled into the $12 \times M$ chromagram matrix:

$$\hat{C} = \big[\hat{\mathbf{C}}_0,\ \hat{\mathbf{C}}_1,\ \ldots,\ \hat{\mathbf{C}}_{M-1}\big],$$

where rows $p = 0, \ldots, 11$ correspond to pitch classes and columns $m = 0, \ldots, M-1$ correspond to successive time frames.

B.2 Similarity Across Songs

To assess how musical similarity evolves over time, we compare pairs of songs by measuring how well their internal repeating structures "compress" together. For each song we begin with

its normalized chromagram $\{\hat{\mathbf{C}}_m\}_{m=0}^{M-1}$ and form a *delay-embedded* sequence using $n_{\text{steps}} = 10$:

$$\mathbf{y}_n = \left[\hat{\mathbf{C}}_n,\ \hat{\mathbf{C}}_{n-1},\ \ldots,\ \hat{\mathbf{C}}_{n-10}\right] \in \mathbb{R}^{12\times 11}.$$

This construction follows the approach implemented in `librosa.sequence.stack_memory`, which generates time-delay embeddings to incorporate short-term temporal structure. The resulting vectors capture melodic and rhythmic context spanning approximately ten frames.

From the delay-embedded sequence $\{\mathbf{y}_n\}_{n=0}^{T-1}$, define Euclidean distances

$$D_{i,j} \;=\; \|\mathbf{y}_i - \mathbf{y}_j\|_2.$$

The binary recurrence matrix $R \in \{0,1\}^{T\times T}$ is constructed by a k-nearest-neighbors rule on these embedded vectors. Fix an exclusion width $w = 1$ to ignore trivial temporal neighbors near the main diagonal. For each time index j, let $\mathcal{N}_k(j)$ be the set of indices of the k nearest neighbors of $\mathbf{y}_j$ among all i satisfying $|i - j| \geq w$. Then

$$R_{i,j} \;=\; \begin{cases} 1, & \text{if } i \in \mathcal{N}_k(j), \\ 0, & \text{otherwise.} \end{cases}$$

where k adapts to the sequence length according to

$$k \;=\; 2\left\lceil \sqrt{T - 2w + 1}\,\right\rceil.$$

Because neighbor relations need not be mutual, R is not symmetric.

We then quantify similarity between two songs x and y by computing the *Normalized Compression Distance* (NCD) between their recurrence plots:

$$\text{NCD}(x,y) = \frac{K(x,y) \;-\; \min\{K(x), K(y)\}}{\max\{K(x), K(y)\}},$$

where $K(\cdot)$ is the Kolmogorov complexity–approximated here by the size of the `bzlib`-compressed representation of the plot–and $K(x,y)$ is the complexity of the concatenated pair.[23] If two songs share extensive structural redundancy, $K(x,y)$ will be close to $\min\{K(x), K(y)\}$ and NCD will be near zero; if they are dissimilar, NCD approaches one. This compression-based measure requires no explicit alignment of tempo or pitch: it purely reflects how much one song's repeating patterns help compress another's. In practice, identical recordings yield NCD = 0, while cover versions (which preserve most structure but introduce small variations)

[23] Compression-based distances are computed using CompLearn: https://complearn.org/ncd.html.

produce values around 0.96, indicating high—but not perfect—similarity.

B.3 Chorus Detection

The chorus is the section of a song that is typically repeated the most times and serves as a central, most memorable part of the composition. Identifying the chorus in a song is essential for various applications in Music Information Retrieval (MIR), including song summarization, music recommendation, and cover song identification. In this section, we describe the methodology used for chorus detection, which follows the work by Goto (2006).

Given the normalized chromagram $\{\hat{\mathbf{C}}_m\}_{m=0}^{M-1}$, the *self-similarity matrix* is

$$S_{n,m} = 1 \; - \; \frac{\left\|\hat{\mathbf{C}}_n - \hat{\mathbf{C}}_m\right\|_2}{\sqrt{12}}, \qquad n, m = 0, \ldots, M-1,$$

so that $S_{n,m} = 1$ for identical chroma frames and decreases toward 0 as they diverge. In this $M \times M$ matrix, repeated musical sections appear as diagonal stripes.

To better highlight these repeating patterns, especially across varying offsets, it is useful to re-index the matrix by the *lag* $\ell = n - m$, defining

$$\widetilde{S}(n, \ell) = S_{n,n-\ell}, \quad \ell = -(M-1), \ldots, 0, \ldots, (M-1).$$

This transformation maps a diagonal at offset ℓ in S to a roughly horizontal band at row ℓ in $\widetilde{S}$, making structural repetitions more visually and algorithmically accessible.

Audio often produces isolated high-similarity pixels–*speckle*–that do not correspond to sustained repeats. To suppress speckle, each fixed-lag row $\widetilde{S}(\cdot, \ell)$ is convolved with a one-second rectangular window. Concretely, let

$$L_{\text{smooth}} = \left\lceil 1.0 \text{ second} \times \frac{M}{T_{\text{song}}} \right\rceil$$

be the number of chroma frames in one second (where T_{song} is the song's duration in seconds). Then each entry of $\widetilde{S}(\cdot, \ell)$ is replaced by the average over its L_{smooth}-frame neighborhood, reinforcing continuous bands and eliminating brief spikes.

Next, for each lag ℓ, all contiguous intervals of frames are identified

$$n_{\text{start}} \leq n \leq n_{\text{end}}$$

for which the smoothed similarity satisfies $\widetilde{S}(n, \ell) \geq 0.8$. Intervals shorter than ten seconds

are discarded; that is, only those with a length satisfying

$$n_{\text{end}} - n_{\text{start}} \;\geq\; \left\lceil 10 \text{ seconds} \times \frac{M}{T_{\text{song}}} \right\rceil$$

frames are retained. This threshold helps eliminate false positives such as short melodic phrases or recurring instrumental motifs that are not full chorus segments. Each surviving interval $[n_{\text{start}}, n_{\text{end}}]$ at lag ℓ is called a *chorus line*, representing one occurrence of a repeated section whose duration is $(n_{\text{end}} - n_{\text{start}}) \times (T_{\text{song}}/M)$ seconds. Next, we rank the repeated sections based on how frequently they occur with consistent timing and spacing. The spacing between repetitions, referred to as the offset, indicates how far apart in time each repeat appears relative to its original occurrence. Repeats with similar offsets reflect a regular pattern of repetition, which is characteristic of structured musical sections such as choruses.

To choose the main chorus, we score each chorus line by counting how many other lines overlap it. Two lines $[n^i_{\text{start}}, n^i_{\text{end}}, \ell_i]$ and $[n^j_{\text{start}}, n^j_{\text{end}}, \ell_j]$ are said to overlap if their time intervals intersect and their lag difference satisfies

$$\left|\ell_i - \ell_j\right| \;\leq\; \delta, \quad \delta = \left\lceil 0.2\,(n^i_{\text{end}} - n^i_{\text{start}}) \right\rceil$$

(i.e. within 20% of the first line's length). This overlap count serves as a proxy for how prominently and consistently the section recurs throughout the song. Summing these overlap counts gives each line a *score*. The line with the highest score is selected as the main chorus, and its start frame n^* is converted back to seconds by

$$t_{\text{chorus}} = n^* \times \frac{T_{\text{song}}}{M},$$

yielding the estimated chorus onset time.

B.4 Vocal Onset Detection

Vocal detection is a well-established field in audio analysis, playing an important role in everyday applications such as voice calls, audio processing, and command recognition for electronic devices and smartphones. In this paper, voice detection is used to determine if the onset of vocal activity in songs is changing over time (e.g., vocal onsets occurring earlier). To do so, we leverage techniques in signal processing and deep learning. We describe each step used to isolate the vocal content from complex audio.

To isolate the vocal component from the original raw audio files, this analysis uses Open-Unmix–a deep neural network model for music source separation Stöter et al. (2019). Open-

Unmix provides pre-trained weights capable of decomposing a full-track audio signal into four stems: vocals, drums, bass, and other instruments.

The process begins with the full-mix waveform $x[n]$, $n = 0, 1, \ldots, L-1$, sampled at f_s Hz. Its short-time Fourier transform is computed in overlapping frames of length N and hop size H, using the Hann window

$$w[n] = 0.5\Big(1 - \cos\Big(\tfrac{2\pi\, n}{N-1}\Big)\Big), \quad n = 0, \ldots, N-1,$$

which smoothly tapers each frame to zero and reduces spectral leakage. The resulting complex spectrogram is

$$X_m[k] = \sum_{n=0}^{N-1} x[n + mH]\, w[n]\, e^{-j\,2\pi\, k\, n/N}, \quad m = 0, \ldots, M-1,\ k = 0, \ldots, \tfrac{N}{2}.$$

To isolate vocals, Open-Unmix uses pretrained weights to derive a learned *soft mask* $M_m[k] \in [0, 1]$ at each time–frequency bin. A soft mask attenuates non-vocal elements by scaling the original spectrogram with values between zero (full suppression) and one (full retention), enabling partial separation rather than binary filtering. The mask $M_m[k]$ is predicted using stacked bidirectional LSTM layers, which integrate past and future spectral context to identify vocal features such as vibrato, sustained notes, and phrasing.[24] The model is trained on the MUSDB18 dataset—consisting of paired mixtures and ground-truth vocal and instrumental stems–by minimizing the mean-squared error between its masked output and the true vocal spectrogram, thereby learning to produce masks that best recover the vocal component.[25]

The estimated vocal spectrogram is then

$$\widetilde{X}_m[k] = M_m[k]\; X_m[k].$$

We convert this to a decibel-scale amplitude representation:

$$D_m[k] = 20\, \log_{10}\big(|\widetilde{X}_m[k]| + \varepsilon\big),$$

using $\varepsilon = 10^{-6}$ to avoid taking the logarithm of zero. This logarithmic mapping compresses large peaks and enhances quieter vocal nuances.

Next, we smooth $D_m[k]$ over time with a centered three-second moving average, thereby

[24] Details on LSTM can be found in **?**.

[25] The dataset is available here: https://www.kaggle.com/datasets/raycarterrete/musdb18-stem-division-vocals-and-instrumentals.

emphasizing sustained vocal segments over brief fluctuations. If $L_3 = \lceil 3\,\text{sec} \times (M/T_{\text{song}}) \rceil$ is the number of frames in three seconds, then

$$\widetilde{D}_m[k] = \frac{1}{L_3} \sum_{i=m-\lfloor L_3/2 \rfloor}^{m+\lfloor L_3/2 \rfloor} D_i[k].$$

This averaging emphasizes vocal phrases lasting multiple seconds while suppressing brief, non-vocal transients. The conversion to decibels and subsequent smoothing are designed to reduce false positives: because the goal is to detect vocal onsets, even a single-frame error can lead to a misleading result. As such, the algorithm is calibrated to favor high-confidence detections. This emphasis makes the resulting onsets more reliable, which is crucial given that even one incorrect detection can significantly distort our results.

To detect onsets, we use Librosa's onset detection framework, described in McFee et al. (2015), which collapses the smoothed spectrogram across frequency into an energy-difference envelope:[26]

$$E[m] = \sum_{k=0}^{N/2} \max\big(\widetilde{D}_{m+1}[k] - \widetilde{D}_m[k],\, 0\big),$$

capturing the total positive rise in vocal energy at each frame. To suppress spurious peaks, the envelope $E[m]$ is further smoothed using a two-second moving average over $L_2 = \lceil 2\,\text{sec} \times (M/T_{\text{song}}) \rceil$ frames:

$$\overline{E}[m] = \frac{1}{L_2} \sum_{i=m-\lfloor L_2/2 \rfloor}^{m+\lfloor L_2/2 \rfloor} E[i].$$

Finally, vocal onsets are identified as those local maxima of $\overline{E}[m]$ that exceed the 90th percentile of its values and are separated by at least two seconds. Converting the selected frame indices m back into time via

$$t = \frac{m}{M}\, T_{\text{song}}$$

yields the estimated vocal-phrase onset times in seconds. The algorithm detects points of significant energy increase, which typically correspond to the beginnings of vocal phrases, while aiming to minimize false positives caused by instrumental transients or ambient noise.

[26] `librosa.onset.onset_strength` adapted here for the vocal stem.