ABSTRACT

Title of Document: MY MOBILE MUSIC:
AN ADAPTIVE PERSONALIZATION
SYSTEM FOR DIGITAL AUDIO PLAYERS

Tuck Siong Chung, Ph.D., 2007

Directed By: Professor Roland T. Rust,
Department of Marketing

Professor Michel Wedel,
Department of Marketing

This paper develops a music recommendation system that automates the downloading of songs into a mobile digital audio device. The system tailors the compositions of the songs to the preferences of individuals based on past behaviors. We describe and predict individual listening behaviors using a lognormal hazard function. Our recommendation system is the first to accomplish this and there is as of this moment no existing alternative. Our proposed approach provides an improvement over alternative methods that could be used for product recommendations. Our system has a number of distinct features. First, we use a Sequential Monte Carlo algorithm that enables the system to deal with massive historical datasets containing listening behavior of individuals. Second, we apply a variable selection procedure that helps to reduce the dimensionality of the problem, because in many applications the collection of songs needs to be described by a very large number of explanatory variables. Third, our system recommends a batch of products rather than a single product, taking into account the predicted utility and the uncertainty in the parameter estimates, and applying experimental design methods.

MY MOBILE MUSIC: AN ADAPTIVE PERSONALIZATION SYSTEM FOR DIGITAL AUDIO PLAYERS

By

Tuck Siong Chung

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park, in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2007

Advisory Committee:
Professor Roland T. Rust, Co-Chair
Professor Michel Wedel, Co-Chair
Professor John P. Rust
Associate Professor P.K. Kannan
Associate Professor Wendy Moe

Dedication

To my wife Angie who has been both supportive and accommodating during the pursuit of my doctoral degree.

Acknowledgements

I am grateful to many individuals who contributed to this dissertation, especially to Dr. Roland T. Rust and Dr. Michel Wedel for providing many hours of guidance and stimulating discussions, and for all their patience and encouragement.

Committee members, Dr. John P. Rust, Dr. P.K. Kannan and Dr. Wendy Moe provided valuable comments and suggestions that helped me to improve the quality of this dissertation.

I would like to thank Dr. Howard Frank, the Dean of Robert H. Smith School of Business, Maryland, College Park for his generous financial and moral support that made it possible to run the experiment for this dissertation.

Table of Contents

Dedication ii
Acknowledgements iii
Table of Contents iv
List of Tables v
List of Figures vi
Chapter 1: Introduction 1
Chapter 2: The Model 7
  Section 1: The Lognormal Hazard Function 9
  Section 2: Bayesian Variable Selection 14
  Section 3: Sequential Monte Carlo Estimation 17
  Section 4: Model Averaging 20
  Section 5: Bayesian Experimental Design 23
Chapter 3: Simulation Study 28
  Section 1: Simulated Data 30
  Section 2: Estimation 31
  Section 3: Simulation results 34
Chapter 4: Experimental Study 41
  Section 1: Data collection 41
  Section 2: Estimation and customization procedures 45
  Section 3: Results of the experiment 46
Chapter 5: Discussion and conclusion 59
Appendix: Survey on Music Preference Study 63
References 64

List of Tables

Table 1 *Parameter used to generate simulated data* 30

Table 2 *Parameter estimates in the first simulation study (with variable selection)* 34

Table 3 *Parameter estimates in the first simulation study (without variable selection)* 34

Table 4 *Credible intervals of parameter estimates in the first simulation study* . 35

Table 5 *Comparison of Mean Squared Errors (MSE) of predictions with and without model averaging* 36

Table 6 *Comparison of design performances* 37

Table 7 *Distribution of the song-specific constants.* 51

Table 8 *Distribution of subject weights.* 52

Table 9 Comparison of play-lists' performance in terms of how well they recommend songs 54

Table 10 Subjects' ratings of the play-lists' performance obtained from post experiment 56

List of Figures

Figure 1 *Plot of the lognormal hazard function* 12

Figure 2 *Transition in actual utilities for the optimal designs during sequential design generation* 39

Figure 3 *Change in song compositions during the sequential generation of the optimal design* 40

Figure 4 *Screen shot of PDA software* 42

Figure 5 *Sigma values of the estimated experiment subject's parameters* 47

Figure 6 *Percentage distribution of songs heard by the duration heard in milliseconds* 48

Figure 7 *Percentage distribution of songs by the ratio of listening duration to song length.* 48

Figure 8 *Distribution of experimental subjects by the number of non-zero song characteristic coefficients.* 49

Figure 9 *The distribution of experimental subjects by the number of non-zero coefficients for describing song genres.* 50

Figure 10 *Distribution plot of the song-specific constants.* 52

Figure 11 Distribution of subject weights. 53

Chapter 1: Introduction

Mobile digital audio devices have become so common that in 2005 about eleven percent of the American adult population have an iPod or MP3 player (Pew Internet & American Life survey). Concurrent with the increasing penetration of these devices is the growing adoption of digital music in 2006 (US Music Consumer Survey, 2006). The use of music play-lists has steadily increased as consumers look for a more personalized digital music collection to suit their individual tastes. Despite the growing importance of the issue of music recommendation for mobile devices, as far as we know there is yet to be published any article that develops a music recommendation system for mobile music devices.

The objective of this paper is to fill this gap and to develop such a music recommendation system. This paper will also provide a validation of the proposed recommendation system through an actual implementation of the system on Personal Digital Assistants (PDAs), and data collected through an recommendation experiment. The target audiences for this paper are researchers who are involved in the research in recommendation systems, and practitioners who are looking for a better recommendation system for their music products. Our recommendation system is directly implementable and provides useful new features that will benefit recommendation practice as it resolves a number of the challenges of a real life recommendation system.

A choice of music is highly dynamic. An individual music choice may change due to the individual's emotions, listening context and contacts with the other media. This calls for a system that is able to make real time recommendations. One

of the challenges of making music recommendations arises from the heterogeneity of individual music preferences. Music preferences are highly personal as they relate to specific personality characteristics, cognitive ability and emotions (Rentfrow and Gosling, 2003). An effective music recommendation system therefore is one that provides individual customization.

In order to carry out the customization, information on individual music preferences is necessary. The problem of explicitly asking individuals for their music preferences is that they are reluctant to actively provide personal information due to the efforts involved, and the fear of the invasion of privacy. Explicit inputs of individual preferences in recommendation systems are sparse because only a small proportion of them are willing to provide the inputs (see Konstan et al. 1997) and as a consequence, most of the information in the systems based on these data is missing (Ying, Feinberg and Wedel, 2006). In addition, when asked about their product preferences, individuals may not be able to fully or accurately express them. For example, one problem with the use of genre to classify music type is that a system's categorization of genres may not map to an individual's mental model of music. Yet all studies published in the marketing literature on recommendation systems depend on such input elicited from respondents. The use of explicit preference data that comes in the form of product ratings creates another problem. Individuals don't always response to rating scale in accordance to their preferences. They commonly indicate their preferences by choosing the middle, or the extreme of the rating scales, (Rossi, Gilula and Allenby, 2001).

An alternative to explicit inputs from the individuals is to infer their preferences from their demographic profiles. However, due to the less than perfect match between demographic profiles and personality traits, recommendation systems that utilize only demographic profiles for recommendations are inherently inaccurate. A more effective music recommendation system will be one that infers music preferences based on the individuals' past choice behaviors, as well as the choice behaviors of other similar individuals. Recommendation systems generally fall into two categories -- content filtering and collaborative filtering. Simply stated, content filtering makes recommendations based on an individual's past preferences for product attributes. Other the other hand, collaborative filtering predicts an individual's preferences using a weighted sum of other individual's preferences. The weights are reflection of how closely one individual's preferences are to the others. Our recommendation system falls into the category of a hybrid system since it utilizes both content and collaborative filtering. Recommendation systems that also use a hybrid approach can be found in the marketing literature (e.g. Ansari, Essegaier and Kohli, 2000).

An additional challenge for music recommendation arises from the problem of massive datasets created as a result of the number of songs and individuals involved, and the potentially large number of explanatory variables. A massive dataset is an issue when the algorithm is computationally intensive, as with the Markov Chain Monte Carlo (MCMC) procedure that involves more than a single pass through the data. A large number of explanatory variables can thus, results in an unreasonable time lag for product customization. The last challenge results from the demands on

the system for individualized customization. This means that the system developed needs to sequentially update individual level estimates to refine the coefficient estimates and to adapt to the changes in individual music preferences.

There are a number features in our recommendation system that resolve the above mentioned challenges of a recommendation system. First, the Sequential Monte Carlo algorithm that we use updates the parameter estimates as new data come in. The Sequential Monte Carlo algorithm is ideal for the implementation of real-time customization not only because of the sequential way in which it updates parameter estimates, but also because of the speed with which the estimation is done.

Second, our music recommendation system removes the need for individuals to explicitly provide inputs on their song preferences. Music preferences are learned based on the history of how long a song is listened to, with the assumption that individuals will listen to songs that provide higher utilities.

Third, our recommendation system addresses the problem of massive data. The ability to handle large datasets comes partly from the processing of data in blocks, and partly from the incorporation of a variable selection step into our algorithm. The variable selection step removes redundant or irrelevant variables that unnecessarily complicate the analysis, therefore reducing the system's computational burden.

Fourth, the use of Bayesian methods in our system ties in with the need for sequential updating of individual level estimates. As more and more blocks of data come in, our estimates of individual preferences improve. Our system also adapts the customization process to the changes in tastes and preferences. This is because, as

new information is obtained on the individuals, this information is used to dynamically update the system's estimation of preferences.

Fifth, our recommendation system automates the downloading of songs to a mobile audio device, and as a result removes the need for individual intervention. When the system removes the need for individual inputs, and also adapts to individual changes in preferences, it enables the music downloading process to be fully automated.

We reviewed some of the existing music recommendation system in practice and the systems that have been developed in the academic literature. In practice no mobile audio device recommendation system based on past behavior is available. Some of the existing music recommendation systems, such as CDNow, MediaUnbound and MoodLogic just to name a few, ultimately allow the website users to download the recommended music into mobile music audio devices. However, they based their recommendations on music data groupings and the website users' interests. These systems take inputs from the users on their music preferences using general questions on their music preferences and tastes, ratings on a sample of music objects, and open ended questions on favorite types of music. Other music recommendation systems like the Pandora.com and last.fm broadcast music tracks to their internet radio station listeners, and modify their future broadcasts based on the user indication of which artists are and are not acceptable. However, such systems are not specifically design to help the website users download music into their mobile audio devices, and are based on simple similarity measures of user mentioned preferences to general song characteristics (music genomes).

In the academic literature recommendation systems are static, and their performances are validated through the use of secondary data collected for other purposes (e.g. Ansari and Mela, 2003; Montgomery et al., 2004) or through simulated data (e.g. Raghu et al., 2001; Ariely, Lynch and Aparicio, 2004). Some of these papers (e.g. Ansari, Essegaier and Kohli, 2000) use rating data and therefore encounter the problem of rating scale response biases mentioned in Rossi, Gilula and Allenby (2001). In addition, unlike our recommendation system that recommends a play-list (i.e. a set of products) these recommendation systems make product prediction to an individual one product at a time.

This paper is divided into five chapters. Chapter one provides an introduction, background and motivations for the paper. Chapter two provides a description of the model used to estimate subject preferences and a description of the play-list customization procedures. Chapter three describes the simulation study conducted as an initial test of our model. The experimental study is discussed in chapter four, along with results. Finally, our paper ends with chapter five that provides a discussion and conclusions.

Chapter 2: The Model

This section describes the model used to estimate subject preferences and the customization of individualized music play-lists to suit these preferences. The model has the following features: (1) it uses a hazard model to estimate subject's utility from the time a subject spent listening to a song, (2) it uses a Bayesian variable selection procedure to select relevant song characteristic in the hazard model for each subject, (3) it uses a particle filter for estimating the models in real time, allowing for sequential updating of the parameters as new data come in, and accommodating the massive data accumulated on song listening behavior, and (4) it uses a model averaging procedure for collaborative filtering to generate recommendations. To achieve this, models of different subjects are averaged based on their similarity. The model averaging also allows us to deal with idiosyncratic song effects not captured by the song descriptor variables, through a pooled model with song-specific constants. Finally, (5) it uses optimal sequential experimental design methods to generate the utility maximizing play-lists across a sequence of batch recommendations, based on the model-averaged estimates.

This section starts by describing the hazard function. The hazard function is used to link a subject's utility for a song to how long this subject listens to the song. A hazard model predicts duration and the probability of termination for an event. In our case, the event corresponds to the act of listening.

The Bayesian Variable Selection procedure is described next. This procedure is used to deal with the situation in which the number of explanatory variables is large. It reduces the number of variables to a more manageable size and removes the

variables which are redundant from the model, at the individual level. The variable selection simplifies and improves the parameter estimations and model predictions. We illustrate the Bayesian Variable Selection, and demonstrate that for some subjects some of the variables can be removed from the model.

The next subsection describes the Sequential Monte Carlo procedure, based on particle filtering. This procedure is used to deal with a massive dataset and for sequential updating of the parameters estimates as new data are obtained. First, we describe how the particles are generated. Second, we describe the updating of particle weights, which dictates how much of the parameters in each particle are to be used for prediction. Third, we describe the particle rejuvenation or re-sampling, an important step in particle filtering. This step is used when the effectiveness of the particles drops below a tolerance level.

We move on to the description of the Model Averaging next. Model averaging is used because we believe that the prediction of a subject's song preferences can be improved by borrowing the information from other similar subjects who have listened to the same song. In addition, model averaging also allows us to introduce song-specific constants into the prediction. The song-specific constants are important because they are used to incorporate characteristics of songs that are not otherwise captured by the prediction variables.

Lastly, we discussed how the individualize play-list is designed using an Experimental Design approach. In our model, we derive the optimal play-list by maximizing a function consisting of the proxy of song's utility (i.e. the predicted listening duration), and an information criterion that accounts for parameter

uncertainty, enabling the listening data to be used to update the individual hazard model parameters efficiently.

Section 1: The Lognormal Hazard Function

We make the assumption that we can infer song preferences based on the time a subject spends listening to a particular song. The assumption that a subject will listen to songs they like longer than to songs they don't like is not an unreasonable one. It is a commonly observed phenomenon that an individual will prolong pleasurable experiences and shorten painful ones. Specifically, Holbrook and Gardner (1993) shows that pleasure has a positive effect on music listening duration. We model listening behavior using a commonly-used parametric survival model - the lognormal hazard model. The basic input for our model is the amount of time a subject has listened to a particular song. Individual models and an aggregate model are estimated using the available data. Our models are fully individualized and estimated separately for each subject, which enables real-time processing of the data, individual level variable selection and individual level recommendations. The estimates are obtained using song attributes and genre as predictors. The aggregate model on the other hand, predicts all subjects' listening durations based purely on song-specific constants. This is done by using a dummy variable in the log-normal hazard model for each song across all subjects. The aggregate model is used to capture the impact of an individual song's unique characteristics, which are not otherwise reflected in the prediction by the individual level models.

Equation 1, 2 and 3 below describes the aggregate model, where y_k represents an observation describing the log listening duration of a subject for the song *k*. For

simplicity in notation, we have omitted the individual level subscript. If y_k has a lognormal distribution with a mean of μ and variance of σ^2, then the probability density function is

$$f(y_k \mid \mu, \sigma) = (2\pi)^{-1/2}(y_k\sigma)^{-1}\exp\left\{-1/2\sigma^2(\log(y_k)-\mu)^2\right\} \quad (1)$$

The survival function is given by

$$S(y_k \mid \mu, \sigma) = 1 - \Phi[\{\log(y_k)-\mu\}/\sigma] \quad (2)$$

We can thus write the likelihood function of (μ, σ) as

$$L(\mu, \sigma) = \prod_{k=1}^{N} f(y_k \mid \mu, \sigma)^{\delta_k} S(y_k \mid u, \sigma)^{(1-\delta_k)} \quad (3)$$

There are N observations in the aggregate dataset, which is formed by stacking the data of every subject's listening duration to every song. The number of observations per subject need not be the same. In the aggregate model, we introduce covariates through μ, and write $\mu_k = \boldsymbol{z}'\boldsymbol{\alpha}$ for observation *k*. $\boldsymbol{z}'$ is vector consisting of a binary dummy variable for every song, and $\boldsymbol{\alpha}$ is the vector of coefficients for the aggregate model. The data are censored as we are only able to observe the listening duration up to the length of a particular song. Whether there is censoring in a particular observation is indicated using a censoring indicator δ_k.

In the individual model, the probability density, survival and likelihood function have a similar form as the aggregate model. However, model estimation is done subject by subject using only the subject's data, for reasons that will become apparent below. The likelihood of the individual model takes the form

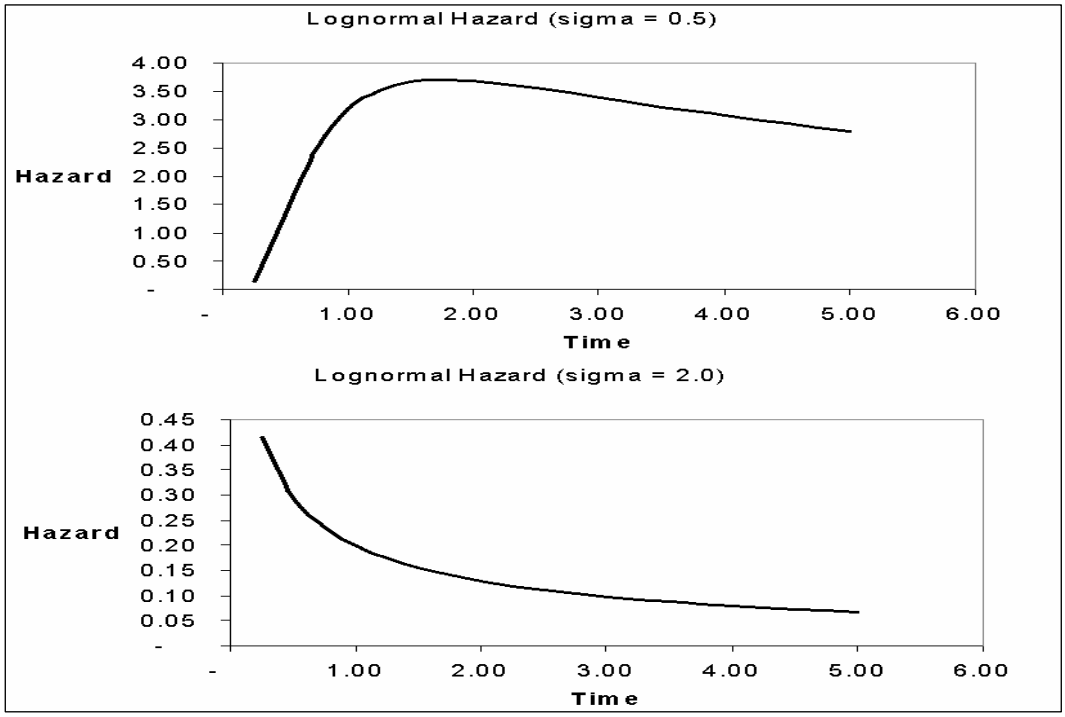
$$L(\mu_i, \sigma_i) = \prod_{k=1}^{n_i} f(y_{ik} \mid \mu_i, \sigma_i)^{\delta_k} S(y_{ik} \mid u_i, \sigma_i)^{(1-\delta_k)} \quad (4)$$

The term n_i above represents the number of observations for subject i. The covariates in the individual model are introduced through μ_i, and we write $\mu_{ik} = \mathbf{x}'\boldsymbol{\beta}_i$ for observation k. $\mathbf{x}'$ represents song attributes and genre variables and $\boldsymbol{\beta}_i$ is the subject specific coefficients. In both the aggregate and the individual-level models we assume that the error terms are independent and identically distributed (iid) random variables. In addition, the variance term σ_i in the models act as the shape parameter that will determine the shape of the log normal hazard functions.

We utilize the lognormal hazard function because it is flexible and represents the type of non-monotone hazard that we expect for songs, that is, unimodal. Other hazard functions like the log-logistic or the expo-power hazards (Saha and Hilton, 1997) represent these shapes as well, but the lognormal has the additional, and for our study crucial, characteristic that the mean is available in closed form, and that it facilitates computation for massive data set in real time.

The hazard function is defined as the ratio of the probability that subject i will stop listening to song j to the probability that subject i is still listening to song j at a particular time. The shape of this hazard function is plotted in figure 1 using different values of σ.

Figure 1 Plot of the lognormal hazard function



As shown in the figure, the hazard function is downward sloping when σ is at a higher value (e.g. a value of 2). This describes a listening behavior in which a subject has a higher tendency to stop listening to a song (i.e. a higher hazard rate) at the beginning. The tendency to continue listening increases (i.e. the hazard rate drops) as the subject listens longer. This reflects a case of increasing interest/attractiveness of the song. When σ is at a lower value such as 0.5, the hazard function describes a different listening behavior. The hazard rate increases from zero to a maximum and then tapers off when the duration increases. This would imply the case when a subject needs certain duration of listening before s/he can decide whether the song is preferable. At the beginning of the curve, the hazard rate increases with the listening duration, possible due to the subject's better understanding of the song's attributes or appeal as time passes. Tendency to make a decision on whether to reject a song increases with a better understanding. If the subject decides to listen to a song longer

than the duration in which the hazard curve inflects, the tendency to stop listening to the song reduces with the listening duration.

The closed form for the posterior distribution of our model parameters is not available. To resolve this issue, we use Markov Chain Monte Carlo (MCMC) and Metropolis Hasting algorithm to sample from the posterior distribution. For the individual models, the first block of data on a subject that comes in is used to generate the initial particles used for the Sequential Monte Carlo procedure. Each particle represents a Metropolis-within-Gibbs sample of the subject's coefficients after a burn-in period. The procedure updates subject's parameters using importance sampling. (The details of the Sequential Monte Carlo procedure are given in a subsequent section of this paper.) The MCMC sampler requires one complete pass through the first block of data. We use a variable selection step into our MCMC procedure. If the variable selection step is not applied, the MCMC sampling will involve simply the successive simulation of $p(\boldsymbol{\beta}_i \mid \sigma_i^2, \boldsymbol{\Omega}_{\boldsymbol{\beta}_i}, \boldsymbol{y}_i)$, $p(\boldsymbol{\Omega}_{\boldsymbol{\beta}_i} \mid \boldsymbol{\beta}_i)$ and $p(\sigma_i^2 \mid \boldsymbol{\beta}_i, \boldsymbol{y}_i)$, where $\boldsymbol{\Omega}_{\boldsymbol{\beta}_i}$ represents the variance-covariance matrix of subject i's coefficients. (The details of the MCMC sampling with the variable selection step are explained in the next section.) For the individual models, estimation of subjects' parameters is done with the Sequential Monte Carlo procedure on the second block of data onwards.

For the aggregate model, we do not include a variable selection step since we need the coefficient for each variable. Each variable in the aggregate model is a song-specific dummy, and the parameters estimated using this model provide a song-specific constant for every song. We will only be able to generate a constant for a

particular song when at least one subject has listened to the song. This means that we need to generate new song-specific constants as new data come in, because some subjects have now listened to new songs. Consequently, the Sequential Monte Carlo procedure is not used for the aggregate model because we will always require new MCMC estimations to generate new song-specific constants. To keep the size of the dataset manageable for the MCMC estimation, only the most recent listening duration of a subject for a particular song is used. In other words, if subject i has listened to a song j in the most recent block of data and also in a different block of data in the past, we will only use the information from the last data block. An additional benefit of this approach is that the song-specific constants are estimated using only the most current data. When there is a change in music tastes, this change is reflected in the song constants.

Section 2: Bayesian Variable Selection

Typical applications of our music recommendation system have a huge number of explanatory variables. This is especially the case when the system includes variables describing the song attributes and granularly defined genres. In a situation where the number of explanatory variables is large, the vector of explanatory variables could contain many redundant or irrelevant variables that unnecessarily complicate the analysis. In our recommendation system, variable selection is applied to the individual models but not to the aggregate model, because we desire estimates of all song-specific constants.

The goal of variable selection is to ignore a variable x_{jp} (i.e. the p-th attribute variable for song j) if β_{ip} (i.e. subject i's p-th coefficient) is equal to zero (Chipman, George and McCulloch, 2001). This involves selecting a sub-model, the likelihood of which has the form

$$p(\boldsymbol{y}_i \mid \boldsymbol{\beta}_{i\boldsymbol{\psi}_i}, \sigma_i^2, \psi_i) = N(\boldsymbol{x}_{i\boldsymbol{\psi}_i} \boldsymbol{\beta}_{i\boldsymbol{\psi}_i}, \sigma_i^2 \boldsymbol{I}) \tag{5}$$

where $p(\boldsymbol{y}_i \mid \boldsymbol{\beta}_{i\boldsymbol{\psi}_i}, \sigma_i^2, \boldsymbol{\psi}_i)$ is the likelihood of $\boldsymbol{y}_i$ given the parameters. This is similar to the likelihood shown in equation 4, but with some of the coefficients set to a value of zero. We indicate if the value of β_{ip} is set to zero using $\boldsymbol{\psi}_i = (\psi_{i1}, \cdots, \psi_{iP})$. When β_{ip} is not equal to zero ψ_{ip} has the value of one, otherwise ψ_{ip} has a value of zero. The vector $\boldsymbol{\psi}_i$ is updated via a Metropolis search during the estimation process. A different set of values of $\boldsymbol{\psi}_i$ represents a different set of coefficients remaining in the model after the variable selection. $\boldsymbol{\beta}_{i\boldsymbol{\psi}_i}$ and $\boldsymbol{x}_{i\boldsymbol{\psi}_i}$ are the vector of regression coefficients and the x matrix corresponding to a particular $\boldsymbol{\psi}_i$.

We apply a Metropolis-Hastings Algorithm for variable selection that involves the successive simulation of

$$p(\boldsymbol{\psi}_i^{\boldsymbol{new}} \mid \boldsymbol{\beta}_{i\boldsymbol{\psi}_i}, \sigma_i^2, \boldsymbol{\psi}_i^{\boldsymbol{old}}, \boldsymbol{y}_i) \tag{6}$$

$$p(\boldsymbol{\beta}_{i\boldsymbol{\psi}_i} \mid \sigma_i^2, \boldsymbol{\Omega}_{\boldsymbol{\beta}_{i\boldsymbol{\psi}_i}}, \boldsymbol{\psi}_i, \boldsymbol{y}_i)$$

$$p(\boldsymbol{\Omega}_{\boldsymbol{\beta}_{i\boldsymbol{\psi}_i}} \mid \boldsymbol{\beta}_{i\boldsymbol{\psi}_i}, \boldsymbol{\psi}_i)$$

$$p(\sigma_i^2 \mid \boldsymbol{\beta}_{i\boldsymbol{\psi}_i}, \boldsymbol{\psi}_i, \boldsymbol{y}_i)$$

After a burn-in period, the Metropolis-within-Gibbs sample draws for the values of $\boldsymbol{\beta}_{i\boldsymbol{\psi}_i}, \sigma_i$ *and* $\boldsymbol{\psi}_i$ are saved as initial particles used for the Sequential Monte Carlo procedure described in the next section.

We use the following priors for $\boldsymbol{\beta}_{i\boldsymbol{\psi}_i}$, $\boldsymbol{\Omega}_{\boldsymbol{\beta}_{i\boldsymbol{\psi}_i}}$ and σ_i^2

$$p(\boldsymbol{\beta}_{i\boldsymbol{\psi}_i} \mid \sigma_i^2, \boldsymbol{\Omega}_{\boldsymbol{\beta}_{i\boldsymbol{\psi}_i}}, \boldsymbol{\psi}_i, \boldsymbol{y}_i) = N\left(\mathbf{0}, \boldsymbol{\Omega}_{\boldsymbol{\beta}_{i\boldsymbol{\psi}_i}}\right) \tag{7}$$

$$p(\boldsymbol{\Omega}_{\boldsymbol{\beta}_{i\boldsymbol{\psi}_i}} \mid \boldsymbol{\beta}_{i\boldsymbol{\psi}_i}, \boldsymbol{\psi}_i) = IG(v/2, v\boldsymbol{\lambda}^x/2)$$

$$p(\sigma_i^2 \mid \boldsymbol{\beta}_{i\boldsymbol{\psi}_i}, \boldsymbol{\psi}_i, \boldsymbol{y}_i) = IG(v/2, v\lambda^y/2)$$

The use of a prior mean of **0** for $\boldsymbol{\beta}_{i\boldsymbol{\psi}_i}$ is a neutral choice reflecting the indifference between positive and negative values for the coefficients. These coefficients are normally distributed with an diagonal variance-covariance matrix $\boldsymbol{\Omega}_{\boldsymbol{\beta}_{i\boldsymbol{\psi}_i}}$. We draw the values of the variance vector $\boldsymbol{\Omega}_{\boldsymbol{\beta}_{i\boldsymbol{\psi}_i}}$ using a prior of $IG(v/2, v\boldsymbol{\lambda}^x/2)$. We chose a small value $v = 5$ for a diffuse prior, and equate $\boldsymbol{\lambda}^{\boldsymbol{x}}$ to the sample variance of $\boldsymbol{x}_i$. The value of σ_i^2 is drawn similarly with a prior of $IG(v/2, v\lambda^y/2)$, while equating λ^y to the sample variance of $\boldsymbol{y}_i$.

The proposed $\boldsymbol{\psi}_i^{new}$ is accepted with the probability:

$$\min\left\{1, \frac{f(\boldsymbol{\psi}_i^{new} \mid \boldsymbol{\beta}_{i\boldsymbol{\psi}_i^{new}}, \sigma_i^2, \boldsymbol{y}_i)}{f(\boldsymbol{\psi}_i^{old} \mid \boldsymbol{\beta}_{i\boldsymbol{\psi}_i^{old}}, \sigma_i^2, \boldsymbol{y}_i)}\right\} \tag{8}$$

Where $f(\boldsymbol{\psi}_i \mid \boldsymbol{\beta}_{i\boldsymbol{\psi}_i}, \sigma_i^2, \boldsymbol{y}_i) \propto f(\boldsymbol{y}_i \mid \boldsymbol{\beta}_{i\boldsymbol{\psi}_i}, \sigma_i^2, \boldsymbol{\psi}_i)\, p(\boldsymbol{\psi}_i^{\boldsymbol{new}} \mid \boldsymbol{\beta}_{i\boldsymbol{\psi}_i}, \sigma_i^2, \boldsymbol{\psi}_i^{\boldsymbol{old}}, \boldsymbol{y}_i)$

A symmetric transition kernel is used for our Metropolis algorithm. This is based on the assumption that every variable has the same probability of being in the model. It is also assumed that all variables have the same probability of switching

from a non-zero coefficient to a zero coefficient as it is vice versa. In other words, we assume that

$$p(\boldsymbol{\gamma}_i^{new} \mid \boldsymbol{\beta}_{i\boldsymbol{\gamma}_i}, \sigma_i^2, \boldsymbol{\gamma}_i^{old}, \boldsymbol{y}_i) = 1/P \text{ and } \sum_{p=1}^{P} \mid \psi_{ip}^{new} - \psi_{ip}^{old} \mid = 1 \tag{9}$$

These are realistic assumptions as they reflect a belief that all variables have the same prior probability of being in the model. They are also convenient assumptions because they simplify the probability of acceptance for $\boldsymbol{\psi}_i^{new}$ to

$$\min\left\{1, \frac{f(\boldsymbol{y}_i \mid \boldsymbol{\beta}_{i\boldsymbol{\psi}_i^{new}}, \sigma_i^2, \boldsymbol{\psi}_i^{new}}{f(\boldsymbol{y}_i \mid \boldsymbol{\beta}_{i\boldsymbol{\psi}_i^{old}}, \sigma_i^2, \boldsymbol{\psi}_i^{old}}\right\} \tag{10}$$

The variable selection step is implemented as a part of the MCMC sampling for the individual models on the first block of data. A set of initial particles is generated and is used for the Sequential Monte Carlo procedure.

Section 3: Sequential Monte Carlo Estimation

One of the challenges of analyzing online consumer behavior is the sheer mass of the data available. A standard Markov Chain Monte Carlo method generally requires a complete scan of the dataset and also a large number of iterations to estimate the model parameters. This makes Bayesian analysis of massive datasets using Markov Chain Monte Carlo methods infeasible.

In this paper, we use the Sequential Monte Carlo Method to estimate the model parameters in the individual models (Ridgeway and Madigan, 2003). Not only does this procedure allows us to analyze large datasets, it also enables us to analyze the data in blocks. In other words, we could estimate a subject's parameters using the data available in one time period, and improve the estimate as new blocks of data

becomes available later. This type of sequential updating of individual level parameters as new data come in is crucial for mobile recommendation systems.

A typical Monte Carlo method samples from the posterior, $f(\boldsymbol{\Theta}_i \mid y_i)$ for subject i, where $\boldsymbol{\Theta}_i$ refers to subject i's parameters (i.e $\boldsymbol{\beta}_{i\boldsymbol{\psi}_i}, \sigma_i \; and \; \boldsymbol{\psi}_i$) and y_i refers to the data. The method then estimates $E(h(\boldsymbol{\Theta}_i) \mid y_i)$ as $(1/M)\sum_{m=1}^{M} h(\boldsymbol{\Theta}_i^m)$. Here M refers to the number of draws from the posterior used to estimate $\boldsymbol{\Theta}_i$ and h is a transformation. In our procedure, we estimate subject parameters using Markov Chain Monte Carlo Methods on the first block of data. We retain the estimates obtained in the different iterations as "particles" after a burn-in period. Each particle is a vector containing the current estimated values of the parameters $\boldsymbol{\beta}_{i\boldsymbol{\psi}_i}, \sigma_i \; and \; \boldsymbol{\psi}_i$ obtained from the Metropolis-within-Gibbs sample draw procedure described in the previous section.

When a second block of data arrives, our estimates of the subjects' parameters are updated using importance sampling. We represent the first block of data as ${}^{1}y_i$ and the second block of data as ${}^{2}y_i$. If $\boldsymbol{\Theta}_i^1, \cdots, \boldsymbol{\Theta}_i^M$ are drawn from $f(\boldsymbol{\Theta}_i \mid {}^{1}y_i)$ we can estimate the posterior expectation of any function $h(\boldsymbol{\Theta}_i)$ as

$$\hat{E}(h(\boldsymbol{\Theta}_i) \mid {}^{1}y_i, {}^{2}y_i) = \left(\sum_{m=1}^{M} w_{im} h(\boldsymbol{\Theta}_i^m) / \sum_{m=1}^{M} w_{im}\right) \qquad (11)$$

where w_i's are the importance sampling weights for subject i. The value of w_{im} for particle m is given by:

$$w_{im} = f((\boldsymbol{\Theta}_i^m) \mid {}^{1}y_i, {}^{2}y_i) / f((\boldsymbol{\Theta}_i^m) \mid {}^{1}y_i) \qquad (12)$$

This greatly simplifies (Ridgeway and Madigan, 2003) to:

$$\propto \prod_{k=1}^{K} f(y_{ik} \mid \boldsymbol{\Theta}_i^m),\ \{y_1, \cdots, y_K\} \in {}^2y_i \qquad (13)$$

where $y_{i1}, \cdots, y_{iK}$ are the observations in the ${}^2y_i$ data block.

The posterior variance of the parameter estimates conditioned on ${}^1y_i$ and ${}^2y_i$ will be smaller than those conditioned on ${}^1y_i$ alone. This means that $f(\boldsymbol{\Theta}_i \mid {}^1y_i, {}^2y_i)$ has a narrower distribution than $f(\boldsymbol{\Theta}_i \mid {}^1y_i)$. The narrower $f(\boldsymbol{\Theta}_i \mid {}^1y_i, {}^2y_i)$ is, as compared to $f(\boldsymbol{\Theta}_i \mid {}^1y_i)$, the larger is the proportion of draws from $f(\boldsymbol{\Theta}_i \mid {}^1y_i)$ that has zero importance weights. This reduces the efficiency of the importance step.

We mitigate the loss of efficiency by the incorporation of a rejuvenation step. The rejuvenation step is used when the effective sample size (ESS) of the importance sampling fall below a tolerance level. We use the tolerance level of 0.1, which means that the rejuvenation step is invoked when the ESS falls below 10% of the sample size used for the Sequential Monte Carlo procedure (Ridgeway and Madigan, 2003). The ESS is the number of observations from a simple random sample needed to obtain an estimate with Monte Carlo variation equal to that obtained with a weighted draw of M particles. Kong, Liu and Wong (1994) shows that the ESS can be approximated by:

$$ESS = (\sum_{m=1}^{M} w_{im})^2 / \sum_{m=1}^{M} (w_{im})^2 \qquad (14)$$

We carry out the rejuvenation step using a systematic re-sampling procedure (Arulampalam et al., 2002). The basic idea is to reduce the number of particles with small sampling weights and increase the number of particles with large weights. The

procedure involves uniform sampling of the particles with replacement using the probabilities derived from the particle weights; particles with higher weights will be selected more often. The sampling is done M times if we need to generate a new set of M particles. After the re-sampling is done, the new particles are each given the new weights of 1/M.

The particles and the importance sampling weights generated with the Sequential Monte Carlo procedure are used to predict the listening duration. Without model averaging, the predicted mean listening duration of subject i for song j, that is in using model i, is given by the weighted average:

$$\hat{E}(y_{ij}) = \left(\sum_{m=1}^{M} w_{im} (\boldsymbol{X}'_j \boldsymbol{\beta}_{i\psi_i}^m + (\sigma_i^m)^2 / 2) / \sum_{m=1}^{M} w_{im} \right) \quad (15)$$

Where w_{im} is the m-th particle sampling weight for subject i, $\boldsymbol{X}'_j$ represents a p-vector of attributes for song j, $\boldsymbol{\beta}_{i\psi_i}^m$ is the m-th particle's values for subject i, and $(\sigma_i^m)^2$ is the m-th particle's residual variance for subject i. Model averaging is used in our system because averaging the prediction over different models will improve our predictions of a subject's listening duration, in particular for songs that the subject has not listened to. The model averaging is similar in spirit to collaborative filtering and is described in the next section.

Section 4: Model Averaging

Typically Model Averaging is used to deal with the problem of model uncertainty. For example, a researcher may have several believable models, each with its own sets of parameters, but is not sure which one is correct. Alternatively,

the researcher may not believe that any of the models is actually correct, but uses them as proxies for some unknown underlying model.

Our motivation for using Model Averaging is closer to the latter since we believe that averaging over the different models will improve our predictions of a subject's listening duration, or stated differently, we use model averaging as our procedure for collaborative filtering. This is especially useful in the case where we have to predict the listening duration of a song that a subject has not listened to before, in generating the play-lists. In such instances, we have to rely on the listening behavior of the other subjects who have similar preferences to make a prediction. It is the essence of collaborative filtering applied to subjects' models rather than the conventional method of applying on subjects' data. Whereas existing recommendation systems define recommendations based on how close the data of person j is to that of person i, we combine individuals' models based on how well they predict the target individuals' data. That is, we borrow strengths across models of different individuals, by combining the predictions/recommendations based on them. In our Model Averaging, similarity between subjects is based on how well one subject's parameters predicts the listening duration of another subject. Even for the case of predicting the listening duration of a song that a subject has listened before, the averaging will give a more accurate prediction because we incorporate the prediction from an aggregate model. The impact of individual song constants is incorporated in the prediction by using the aggregate model as a part of the Model Averaging. Note that the song-specific constants would not be estimable in the individual models, due to the severe lack of data, even using Bayesian shrinkage.

Let $Model_1, Model_2, \ldots, Model_I$ be the models estimated for subjects 1,…,I, where I represents the total number of subjects. We also define $Model_{I+1}$ to be the model that contains the particles of the aggregate model's parameters. In addition, we define $\boldsymbol{\xi}$ to be a I+1 by J matrix of indicator variables. $\boldsymbol{\xi_j} = (\xi_{ij}, \cdots, \xi_{I+1,j})$ and $\xi_{ij} = 1$ if subject i has ever listened to song j, $\xi_{ij} = 0$ otherwise. In addition, $\xi_{I+1,j} = 1$ if at least one subject has ever listened to song j, $\xi_{I+1,j} = 0$ otherwise.

The subject weights are calculated using the posterior odds as in a typical model averaging approach. For example, to calculate the weights that should be applied to subject 2's (i.e. $Model_2$'s) prediction of expected listening duration for subject 1 (i.e. ϕ_{21}), we use (Hoeting et al. 1999):

$$\phi_{21} = \frac{p(Model_2 \mid \boldsymbol{y}_1)}{p(Model_1 \mid \boldsymbol{y}_1)} = \frac{p(\boldsymbol{y}_1 \mid Model_2)}{p(\boldsymbol{y}_1 \mid Model_l)} * \frac{p(Model_2)}{p(Model_1)} \qquad (16)$$

Here, $\boldsymbol{y}_i = (y_{ij})$ represents the vector of log observed listening duration for subject 1. $P(Model_1)$ … $P(Model_{I+1})$ represent the prior probabilities for model 1 to I+1. We use a uniform prior for the models due to the absence of any reason favoring one model to another, therefore the probabilities for all the models are the same and are all equal to

$$p(Model_i) = 1/(I+1) \qquad (17)$$

This simplifies the posterior odds above to the form of a Bayes factor:

$$\phi_{21} = \frac{p(Model_2 \mid \boldsymbol{y}_1)}{p(Model_1 \mid \boldsymbol{y}_1)} = \frac{p(\boldsymbol{y}_1 \mid Model_2)}{p(\boldsymbol{y}_1 \mid Model_1)} \qquad (18)$$

Thus the weight that we should place on a model in the prediction of another subject's data is based on the ratio of the likelihood of that model over the likelihood

of the subject's own model. The ratio is based on the likelihood calculated from one model's particles divided by the likelihood calculated using the particles form another model. The same observations are used for the calculation of both likelihoods. We calculate the between-subject weights for all combinations of subjects, while treating the aggregate model as one of the candidate models. We thus obtain values ϕ_{11} to $\phi_{I+1,I+1}$.

The model-averaged prediction of the expected log-listening duration of subject 1 for song 1 is given by the expression:

$$\sum_{i=1}^{I+1} \xi_{i1} * \phi_{i1} * \hat{E}(y_{i1}) / \sum_{i=1}^{I+1} \xi_{i1} * \phi_{i1} \qquad (19)$$

In this equation ξ_{i1} has the value of one, that is, the target individuals' indicator is always set to one so that the targets individuals' data are always used in the prediction. The predictions of the expected log listening duration for the subjects are used in the experimental design stage to generate the play-list as described in the next section. In the experimental design stage customized individualized music play-list are created for all the subjects.

Section 5: Bayesian Experimental Design

The optimal music play-list that we create determines the songs that are downloaded into the mobile audio devices. The term optimal needs some clarification. Fully optimal designs for nonlinear models with unknown parameters are not obtainable in practice. We define a design to be optimal in the Bayesian sense and given a specific prior, if there is no better design to be found under our multi-criteria objective unless the parameters are known exactly in advance.

When deriving the optimal music play-list for our subjects, we face a sequential decision problem that seeks a balance between immediate payoff on one hand and immediate information that might lead to increased future payoff on the other hand. Immediate payoff is maximized when we maximize a subject's predicted utilities, while immediate information is maximized when we maximize the efficiency in estimating a subject's parameters. Maximizing immediate payoff, that is expected utility, has the disadvantage that the design converges to the same set of songs on all future recommendations. That, in turn makes some parameters inestimable, because subsets of songs that enable the identification of such parameters (i.e. genres) no longer appear in the design. For example, a subject may show a strong preference for Jazz music from the past estimates. We maximize immediate payoff by recommending only Jazz songs to this subject. However, this makes it impossible to estimate the other parameters because songs of the other genres are no longer in the play-list. Instead of using the immediate payoff maximizing approach, we generate a sequence of play-lists that dynamically optimizes both expected utility and parameter efficiency over a finite time horizon.

A multi-criteria experimental design problem that optimizes a combination of an outcome and an information criterion is discussed in Verdinelli and Kadane (1992). The objective function takes the form of

$$E\left[\boldsymbol{y}_d'\boldsymbol{1} + \varpi \log|(\boldsymbol{x}_d'\boldsymbol{x}_d + R)/\sigma^2|\right] \tag{20}$$

Where $\boldsymbol{y}_d'\boldsymbol{1}$ represents the predicted outcome as result of the experiment design, $\boldsymbol{x}_d'$ is the design matrix, R/σ^2 is the prior precision matrix, and ϖ is a weight reflecting the relative emphasis that are placed on the outcome versus the Shannon

Information, or Bayes D-optimality, criterion $(\boldsymbol{x}_d'\boldsymbol{x}_d + R)/\sigma^2$. What is different between the design of Verdinelli and Kadane (1992) and our design is that unlike ours, their design is not sequential. Even in a case of static multi-criteria design, balancing between the gains in information against current yield is difficult. Often, good performance with respect to one criterion works against the performance of others. The challenge then is to understand how the performance of the design changes when we trade off one criterion for another, as reflected in the weight ϖ. There is no hard and fast rule on what the value of ϖ should be, and in many cases it is determined subjectively, by the end user of the objective function or through some experimentation (Verdinelli, 1992; Verdinelli and Kadane, 1992). We derive it for a batch sequential design problem.

We extend the work of Verdinelli and Kadane (1992) to a batch sequential experimental design, because of the way our data are collected and the way we customize the play-list. Each time we customize the play-list for a subject, we obtain new data on the subjects' music preferences, and based on the data we have received so far we customize a play-list for the next round of listening. It is batch sequential design rather than a fully sequential design because our play-list involves designing a set of songs (a batch of design points) rather than just a single song (a single design point). Whether the use of a sequential design gives a better estimation than a static design depends on how good the initial estimate of the parameters is. When the initial estimates are poor, sequential design provides an advantage over static designs, however when the initial design yields estimates that are very close to the real

parameter values sequential design may lead to a lower estimation efficiency (see Ford, Titterington and Kitsos, 1989 for a discussion).

In our model, we derive the optimal play-list for subject i by maximizing the function

$$\sum_{q=1}^{Q} E(y_{iq}) + [\log\{(N_d - k_d) * Q + 1\}]^2 \log | \boldsymbol{x}'_{d_{\psi_i}} \boldsymbol{x}_{d_{\psi_i}} + R_{\psi_i} / \sigma_i^2 | \quad (21)$$

In the equation above, Q is the number of songs in our optimal play-list, which is defined a-priori. $E(y_{iq})$ is the prediction of the expected log listening duration for song q based on the model averaging we described in the previous section, given that song q is one of the songs in the optimal play-list. The ψ_i subscript in the design matrix $\boldsymbol{x}_{d_{\psi_i}}$ and the prior precision matrix R_{ψ_i} / σ_i^2 indicates that the variables that are taken out of the model during the variable selection process are not used for designing the optimal play-list. The static weight ϖ in equation 20 is replaced by the expression $[\log\{(N_d - k_d) * Q + 1\}]^2$, here k_d is a count variable that keeps track of how many play-lists we have generated for the subject so far, and N_d is the number of designs that will be created before the weight drops to zero. In other words, N_d is the finite time horizon over which the design is optimized. The "+1" is added to the expression so that when $k_d = N_d$, the weight ϖ has a value of zero, and the optimal play-list is derived by solely maximizing $E(y_{iq})$. Pronzato and Thierry (2003) prove that designs generated based on the criteria in the form of equation (21) yields both approximately unbiased estimates of the coefficients and optimal designs.

In deriving the optimal play-list, we use a design procedure that maximizes the outcome and the design efficiency criterion simultaneously. The outcome component of the criterion $E(y_{iq})$ maximizes the subject's expected utility. The design criterion improves our future inference of the subject's model parameters. That is, it increases the "spread" of the design to continue to provide information on the individuals attribute coefficients, even as the play-lists are successively zoomed in on the subjects' preferences as data accumulate. The objective of formulating the weight as we do above is to ensure that a greater importance is placed on getting a more accurate estimate of the parameters in the initial designs. As the estimates improve in subsequent designs, we place greater emphasis on increasing the utility that the songs provide. The value of N_d, although any reasonable time horizon may be chosen, is calibrated in the simulation done in the next section. It is desirable for the value of N_d to be somewhat large so that the songs in the play-list do not converge quickly to a very similar type of songs. However, a larger value of N_d means a lower overall utility of the songs in the initial design and if the subjects' parameters change the later designs may not be attuned to the individual estimates well. We choose a reasonable value of N_d through our simulation study such that the initial design will not perform worse than an ad-hoc design, while we ensure that not an excessive number of play-lists are generated based on the same parameter values. Providing utilities that are too low in the initial design will greatly discourage the subject from using our recommendation system, and we would like the expected utilities of the play-lists to gradually increase over the finite time horizon N_d.

Using the expression in equation 21, the optimal design is derived using a modified Fedorov method (Cook and Nachtsheim, 1980) and is implemented in a program written in R. The method we implement involves choosing an initial design, which we derive using a "Greedy" criterion of maximum design utility. At each iteration, the algorithm will exchange each song in the design with a song chosen from the song database, so as to optimize the design according to the multi-criteria. If an improvement is achieved, the new design is adopted. This iterative replacement method is carried out until there is no further improvement even after a complete sweep of all the songs available.

We evaluate how well our model and recommendation system perform using both simulation studies and an experiment. The details of the simulation studies are described in the next section. After the section on simulation, the experimental study is detailed next.

Chapter 3: Simulation Study

As an initial test of our model, we ran two simple simulation studies using five artificial subjects. The first simulation study is used to test how well the sequential Monte Carlo procedure is able to recover the parameters from a simulated dataset. In addition, the study tests the effectiveness of the variable selection step. This is to see if the variable selection really does remove redundant variables from the model.

In the second simulation study, song-specific constants were used to simulate the data. The objective of the study is to find out if the model averaging will improve the prediction of a subject's listening duration for the songs. In addition, we compare

the performance of the play-list design of four different design approaches. These four approaches are: (1) a maximum utility design (2) an optimal design (3) a minimum dissimilarity design (4) a random design. The weight used for balancing the outcome and information criterion was also calibrated from the simulation results of the sequential effect of the weight on the designs.

Section 1: Simulated Data

The data for the first simulation study were created using the parameter values shown in table 1 below.

Table 1 Parameter used to generate simulated data

Variables	Subject 1	Subject 2	Subject 3	Subject 4	Subject 5
Constant	0.00	0.00	0.00	0.00	0.00
Volume	0.00	0.00	**0.35**	**0.25**	**0.35**
Tempo	0.00	**0.35**	**0.35**	0.00	0.00
Voice	**0.50**	**0.45**	0.00	**0.35**	**0.35**
Size	0.00	0.00	0.00	**0.55**	**0.05**
Purpose	0.00	0.00	**0.35**	0.00	0.00
Mood	**0.50**	0.00	0.00	0.00	**0.30**
Alternate	**1.00**	0.00	0.00	0.00	0.00
Dance	0.00	**0.35**	0.00	0.00	0.00
Jazz	0.00	0.00	**1.00**	0.00	0.00
Latin	0.00	0.00	0.00	0.00	0.00
Metal	0.00	0.00	0.00	0.00	0.00
Pop	0.00	0.00	0.00	0.00	0.00
Hip_Hop	0.00	0.00	0.00	0.00	0.00
Reggae	0.00	0.00	0.00	0.00	0.00
Rock	0.00	0.00	0.00	**0.25**	0.00
Soul	0.00	0.00	0.00	0.00	0.00
Vocals	0.00	0.00	0.00	0.00	**0.20**
Std Dev	**0.50**	**0.50**	**0.50**	**0.50**	**0.50**

The values of the standard deviation (i.e. σ) for all the subjects were set at 0.5. This gives a hazard functional curve in which the subjects need some listening duration before making a decision on whether to accept or reject the songs. The song attributes used in the simulated datasets consists of variables that describe the genre that a song belongs to in addition to the characteristics of the songs. The genre used for our model indicates if a song is: Alternative/Indies, Electronic/Dance, Jazz, Metal, Pop, Rap/Hip Hop, Reggae/Ska, Rock , Soul/R&B or Vocal. The variables that describes a songs musical context include: the perceived loudness of the song, the song's tempo, whether the voice of the song is more instrumental or vocal, whether the song is more of a solo or a more orchestra performance, the purpose of the song be it for plain listening or dancing, and finally whether the mood of the song is happy

or angry. We obtained these song attributes for 400 of the actual songs that will be used for an experiment. The attributes of each of the 400 actual songs were obtained using the information extracted from the www.musiclens.de music recommendation site.

Two blocks of data were generated for each subject in both of the two simulation studies. Each block of data was generated by randomly assigning 50 songs to each subject. To ensure that we have a dataset that is of a reasonable size, we assumed that the subject listening to the each assigned song for ten times and therefore giving 500 observations in each block of data. The censoring of the data is kept at a level roughly between 25-35%. For the first study the data were simulated using the parameter values in table 1. For the second study the data were simulated with an additional song-specific constant for each song at a value between 0.00 and 0.25.

Section 2: Estimation

In both studies, the first block of data was used to generate the initial particles using a Metropolis-within-Gibbs sample draw for the individual model. 2500 iterations were used for the MCMC procedure and a burn-in of 2000 iterations was used. This provides us with 500 particles. The variable selection step was applied on the individual models. The variable selection indicator vector $\boldsymbol{\psi}_i$ was updated via a Metropolis search by randomly changing one of the P indices in $\boldsymbol{\psi}_i^{old}$ (e.g. ψ_{ip}^{old}). When the existing value of the index ψ_{ip}^{old} was one, we set the new value ψ_{ip}^{new} to zero. This is equivalent to setting the corresponding value β_{ip}^{new} to zero if β_{ip}^{old} is not zero. If

the existing value of the index ψ_{ip}^{old} was zero, we set the new value ψ_{ip}^{new} to one by assigning the value of β_{ip}^{new} to a non- zero coefficient taken from the β_{i}^{old} vector. A variable which had a non-zero coefficient was then chosen randomly and was given a coefficient of zero. This procedure is similar to the one used in Sha, Tadesse and Vannucci (2006).

In the aggregate model, a Gibbs sample draw was done only for the coefficients of the songs heard by at least one of the subjects. This keeps the coefficients of the unheard songs, and therefore the song-specific constants, at zero. No variable selection and Sequential Monte Carlo procedure was used on the aggregate model. The Metropolis-within-Gibbs sampling was applied only after the second block of data and, only the most recent listening duration of a subject for a particular song was used in the aggregate model. The composition of the songs heard in the two data blocks are different, which means that running the MCMC sampling after the second data blocks would result in a higher number of song-specific constants. 2500 iterations were used for the MCMC procedure and a burn-in of 2000 iterations was used generating 500 particles. With no Sequential Monte Carlo procedure applied, we assigned an equal importance sampling weight to the particles in the aggregate model.

In the first study, the simulation ends at the Sequential Monte Carlo step. The predicted listening durations for the subjects for all 400 songs using the estimated parameters are compared to the predicted value using the initial simulated parameters. In the second study, the model averaging and experimental design stage are

implemented after the Sequential Monte Carlo procedure. The results of the simulations are shown in the next section.

Section 3: Simulation results

The estimation of the subjects' parameters in the first study is shown in table 2 below. An additional table (i.e. table 3) shows the estimates when there is no variable selection applied. The values are obtained after using two blocks of data and after applying the Sequential Monte Carlo procedure.

Table 2 Parameter estimates in the first simulation study (with variable selection)

Variables	Subject 1	Subject 2	Subject 3	Subject 4	Subject 5
Constant	0.00	0.00	0.00	0.00	0.00
Volume	0.00	0.00	**0.34**	**0.19**	**0.44**
Tempo	0.00	**0.42**	**0.42**	0.00	0.00
Voice	**0.52**	**0.47**	0.00	**0.52**	**0.46**
Size	0.00	0.00	0.00	**0.81**	0.00
Purpose	0.00	0.00	**0.38**	0.00	0.00
Mood	**0.57**	0.00	0.00	0.00	**0.38**
Alternate	**0.88**	0.00	0.00	0.00	0.00
Dance	0.00	**0.50**	0.00	0.00	0.00
Jazz	0.00	0.00	**0.48**	0.00	0.00
Latin	0.00	0.00	0.00	0.00	0.00
Metal	0.00	0.00	0.00	0.00	0.00
Pop	0.00	0.00	0.00	0.00	0.00
Hip_Hop	0.00	0.00	0.00	0.00	0.00
Reggae	0.00	0.00	0.00	0.00	0.00
Rock	0.00	0.00	0.00	0.00	0.00
Soul	0.00	0.00	0.00	0.00	0.00
Vocals	0.00	0.00	0.00	0.00	**0.03**
Std Dev	**0.69**	**0.67**	**0.51**	**0.83**	**0.63**

Table 3 Parameter estimates in the first simulation study (without variable selection)

Variables	Subject 1	Subject 2	Subject 3	Subject 4	Subject 5
Constant	0.00	0.00	0.00	0.00	0.00
Volume	**0.04**	**-0.04**	**0.38**	**0.23**	**0.44**
Tempo	**-0.04**	**0.51**	**0.40**	**-0.04**	**0.06**
Voice	**0.59**	**0.52**	**0.01**	**0.57**	**0.52**
Size	**-0.01**	**-0.01**	**0.03**	**0.79**	**0.04**
Purpose	**-0.06**	**-0.01**	**0.41**	**-0.01**	**-0.03**
Mood	**0.60**	**0.03**	**-0.01**	**-0.02**	**0.35**
Alternate	**0.98**	**-0.11**	**-0.03**	**-0.08**	**-0.16**
Dance	**-0.07**	**0.41**	0.00	**-0.09**	**-0.31**
Jazz	**-0.05**	**-0.18**	**0.63**	**-0.23**	**-0.13**
Latin	**-0.02**	**-0.05**	**-0.05**	**-0.03**	**-0.01**
Metal	**-0.01**	**-0.07**	**0.02**	**-0.09**	**-0.03**
Pop	**-0.01**	**-0.01**	**-0.07**	**-0.05**	**-0.13**
Hip_Hop	**-0.03**	**0.06**	**-0.08**	**-0.06**	**0.01**
Reggae	0.00	**-0.01**	0.00	0.00	0.00
Rock	**-0.05**	**-0.15**	**-0.11**	**0.24**	**-0.17**
Soul	**-0.11**	**-0.04**	**-0.07**	**0.02**	**-0.03**
Vocals	**-0.13**	**0.01**	**0.01**	0.00	**-0.02**
Std Dev	**0.75**	**0.67**	**0.44**	**0.78**	**0.57**

Comparing the values in table 1, 2 and 3, the variable selection seems to work well. It effectively forced the non significant parameters to a value of zero. In the

case of subject 4 the genre parameter is set to 0 even when it was given a positive value during the data simulation. We attribute this to the fact that the song characteristics have captured enough of the variability to make the genre variable redundant. This is consistent with our argument that variable selection helps to deal with non essential and potentially numerous genre variables. Without using the variable selection step in our MCMC procedure, the parameters are more spread out and non essential variables are given non zero coefficients. The parameter recovery is also reasonable, with the credible intervals that are tightly bounded. Table 4 shows the statistics of the posterior distribution of the parameter. At the interest of space, only the minimum, maximum, mean and median values of subject 1 are shown. The posterior distributions of the other four subjects have the same pattern as that of subject 1.

Table 4 Credible intervals of parameter estimates in the first simulation study

Variables	Min	Mean	Median	Max
Constant	0.00	0.00	0.00	0.00
Volume	0.00	0.00	0.00	0.00
Tempo	0.00	0.00	0.00	0.00
Voice	**0.47**	**0.52**	**0.51**	**0.57**
Size	0.00	0.00	0.00	0.00
Purpose	0.00	0.00	0.00	0.00
Mood	**0.49**	**0.57**	**0.58**	**0.64**
Alternate	**0.74**	**0.88**	**0.84**	**1.06**
Dance	0.00	0.00	0.00	0.00
Jazz	0.00	0.00	0.00	0.00
Latin	0.00	0.00	0.00	0.00
Metal	0.00	0.00	0.00	0.00
Pop	0.00	0.00	0.00	0.00
Hip_Hop	0.00	0.00	0.00	0.00
Reggae	0.00	0.00	0.00	0.00
Rock	0.00	0.00	0.00	0.00
Soul	0.00	0.00	0.00	0.00
Vocals	0.00	0.00	0.00	0.00
Std Dev	**0.69**	**0.69**	**0.69**	**0.69**

In the second simulation study, we look at the performance of the model averaging in our algorithm. The Mean Squared Errors (MSE) of the predicted listening duration for all the 400 songs are used to compare the prediction based on model average against the prediction using only subject's own particles. The Mean

Squared Errors are calculated using the predicted listening duration minus the "actual" listening duration of the 400 songs. The listening durations before there is any censoring are compared. The "actual" duration is calculated using the original parameter values that we have used to simulate the data. The calculation is deterministic. As shown in table 5, compared to the model calibrated on the data itself, the improvement from the model averaging is substantial. The MSEs under model averaging are close to half of those without model averaging.

Table 5 Comparison of Mean Squared Errors (MSE) of predictions with and without model averaging

	Subject 1	**Subject 2**	**Subject 3**	**Subject 4**	**Subject 5**
MSE (with Model Averging)	3.30	2.78	6.60	16.76	3.66
MSE (Without Model Averging)	6.34	3.46	11.10	43.29	4.71
t-value for difference in means	-3.420***	-1.824*	-4.79***	-7.36***	-2.145**

Note: * reflects < 0.1 significance, ** reflects < 0.05 significance, *** reflects < 0.01 significance,

We compare the performance of the optimal design, a design generated using our model, against three other designs generated using: (1) maximum predicted utility (2) minimum dissimilarity and (3) random approach. The maximum utility design creates the play-list that gives the highest predicted listening duration derived from the model averaging. This design maximizes immediate payoff without any consideration for immediate information. Comparing this design to the optimal design will give an indication of how our model tradeoff immediate payoff for immediate information.

The minimum dissimilarity design is derived by minimizing the average Gower distances of the songs in the play-list from the songs that a subject prefer in the subject last run of listening. We infer that a subject prefers a song when the subject finishes listening to the song - in other words the duration is censored. The

minimum dissimilarity design uses a clustering algorithm similar to the one used by Pandora.com. Pandora.com, an existing commercial music website, recommends new songs to an individual by looking for songs that have attributes closest to the songs the individual indicated as desirable. The minimum dissimilarity design enables us to compare our model with a commercially available recommendation system.

The random design creates a play-list by choosing songs randomly for each subject. This is the design we use when we do not have any preference data, and the design is used to generate our initial play-lists in our experimental study. With the incorporation of preference data, the three other designs should provide higher utilities than the random design. In other words, the random design helps to detect any error in the algorithms of the three other designs.

The performances of the four designs in terms of total utilities and posterior precisions are shown in table 6.

Table 6 Comparison of design performances

	Subj 1	Subj 2	Subj 3	Subj 4	Subj 5
Maximum utility design					
Total utility	458.33	318.22	400.09	484.31	361.87
Posterior Precision	31.01	24.81	26.02	25.51	27.67
Optimal design					
Total utility	416.29	318.22	400.09	465.89	357.98
Posterior Precision	30.98	24.81	26.02	25.45	27.60
Minimum dissimilarity design					
Total utility	306.10	281.49	267.63	289.73	271.20
Posterior Precision	31.26	24.88	26.24	25.62	27.58
Random design					
Total utility	229.09	153.25	173.62	214.88	187.22
Posterior Precision	30.96	24.54	26.04	25.29	27.37

Note: The optimal design refers to the design generated using our model.

Table 6 shows the performance of the optimal design when we set the value of N_d in equation 21 to ten. In other words, ten designs are created before the weight

applied on the Shannon information drops to zero. Operationally this implies that we re-estimate a subject's preferences after we have created ten music play-lists for the subject. Over the duration in which we create the ten music play-lists, we assume that the subjects' preference parameters are constant. After the tenth design, the estimation process starts from the MCMC procedure again which generates a new set of initial particles to be used in the Sequential Monte Carlo estimations that follow.

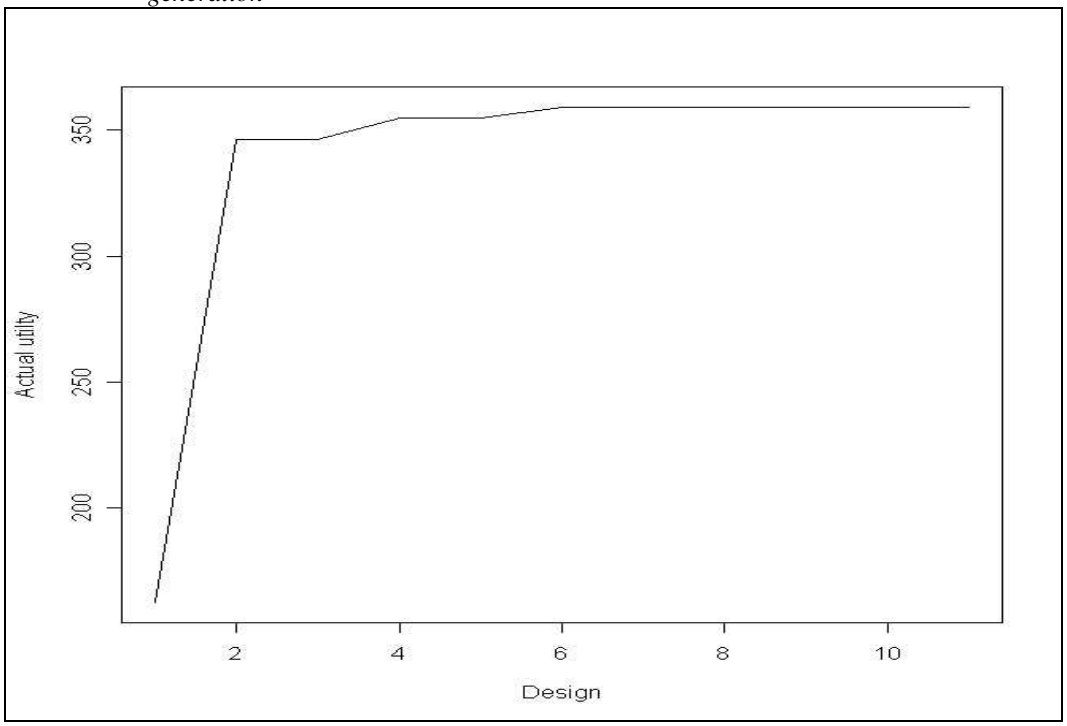
Looking at table 6, the random design always gives the lowest utility. This is reasonable since the random design does not incorporate any preference data. Since it is a random design, the posterior precisions of the parameters estimates may or may not be higher than the maximum utility design depending on the random design composition. The optimal design, created using our model, is obtained from the maximum utility design by trading off utility for D-optimality. This means that in most cases, the utility of the optimal design is lower than the maximum utility design.

The minimum dissimilarity design gives a lower utility but higher posterior precision than the optimal design. On the other hand, the minimum dissimilarity design does not predict subjects' preferences as well as the optimal design.

We simulate the changes in the actual utility of the optimal design from design one to ten for subject five to see if design utilities do improve through sequential design generation. The transition of the actual utilities for the optimal design for this subject is shown in figure 2. The first design is created randomly rather than using the multi-criteria objective, the ten other designs that follows are the ones created optimally. Actual utility of the design is calculated from the true parameters of this subject. Notice that the actual utility increases with each subsequent design. This is a

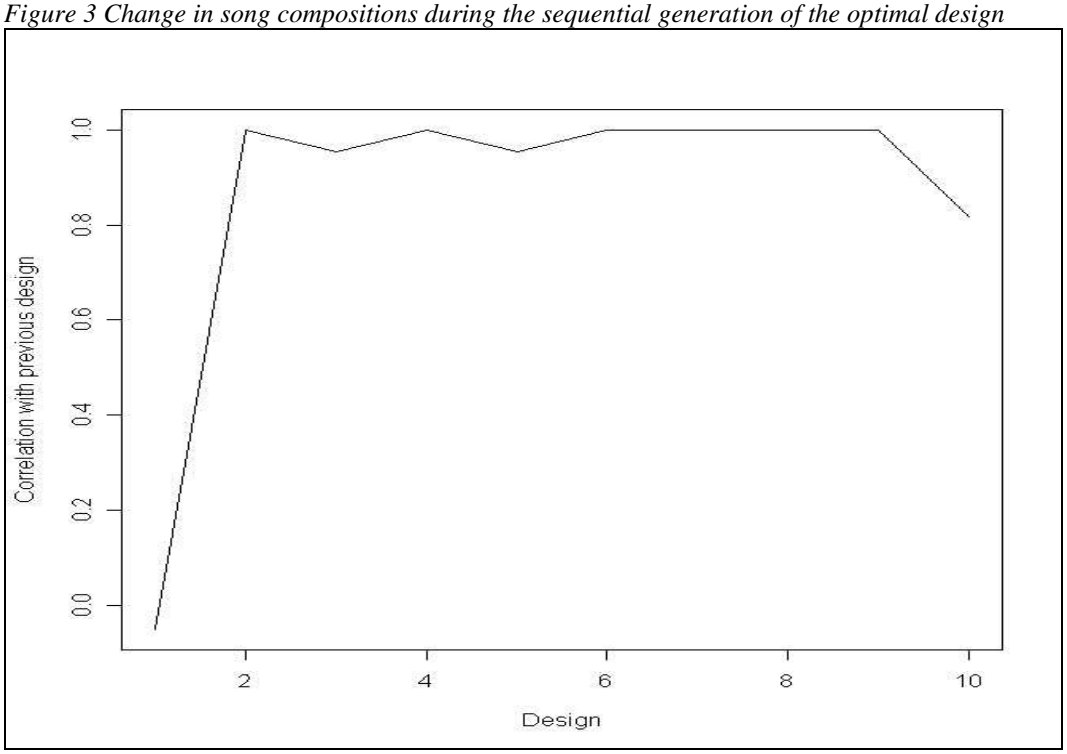
result of an improvement in the estimation of the parameters with more data, and a result of a lower emphasis placed on the Shannon information criterion.

Figure 2 *Transition in actual utilities for the optimal designs during sequential design generation*



The changes in song compositions in each design created sequentially are shown in figure 3. Using the value of one to indicate that a song is chosen to be in the design and a zero otherwise, we can calculate a correlation value between one design and the design that precedes it. A correlation value of one indicates that a design has not changed since the previous design, while a correlation value of zero indicates that the composition of songs has changed completely. As it is shown in figure 3, the correlation value varies between 0.80 and 1.00. The figure shows that the improvement in utilities from the sequential generation of the optimal design is a result of a change in song compositions over and above the change in the weight applied on the Shannon information.

Figure 3 Change in song compositions during the sequential generation of the optimal design



Our recommendation system appears to perform the way that it should based on the results of the simulation study. The optimal designs created based on our model provide higher utilities compared to the random and the minimum dissimilarity design. The minimum dissimilarity design, a clustering algorithm used in Pandora.com, will be used as the benchmark to compare our model in the experimental study. In addition, the utilities of the optimal design converge towards the maximum over the duration in which the 10 different sequential designs are created.

As a further proof of our concept we ran an experiment involving real subjects and a working music recommendation system. The details of the experiment are described in the next section.

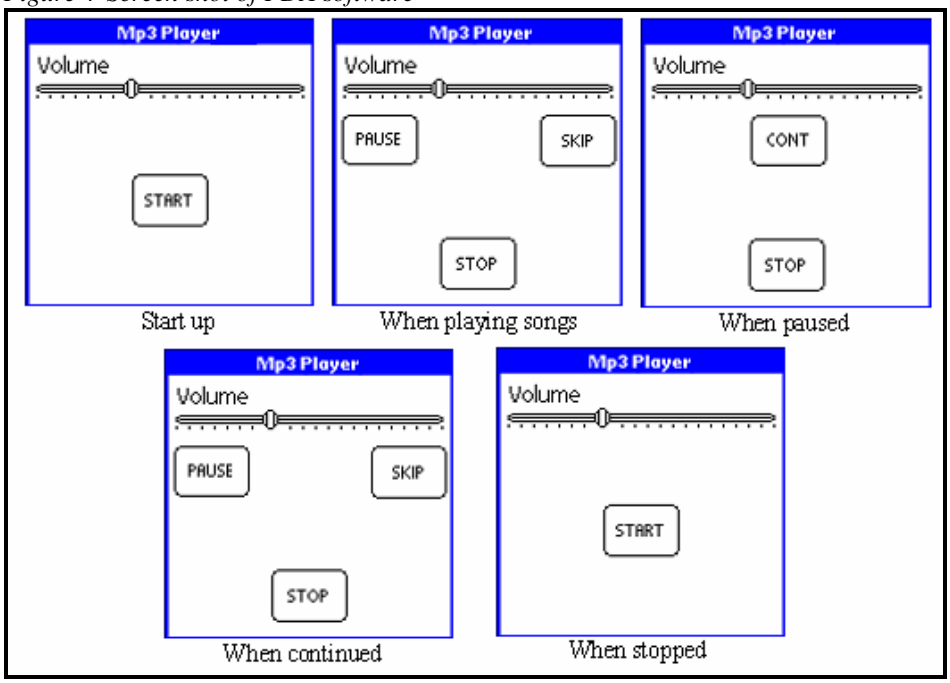
Chapter 4: Experimental Study

Section 1: Data collection

The experiment was run using an actual implementation of the recommendation system. The experiment required the subjects to listen to the songs played from a Palm TX PDA for two waves of listening. In each wave songs were played by the PDAs from the individual play-lists that were downloaded. The play-lists used for the first wave were generated randomly, while the play-lists in the second wave were generated using either the customization procedure described in our model, or a heuristic customization procedure we used as a benchmark.

We developed our recommendation system from scratch and two programs were written for this experiment. The first program was written for the PDAs to allow them to play Mp3 songs according to a play-list, and also to act as data collection instruments. The programming of the PDAs was done using the Handheld Basic Software developed by Peter Holmes Consulting. The screen shots of the PDAs program is shown in figure 4.

Figure 4 Screen shot of PDA software



The second program was written in R for a desktop computer. The R program implements the estimation and customization procedures using ours, and the benchmark customization procedures. Estimations of subjects' parameters were made using the data downloaded from the PDAs to the desktop computer. Finally, the individually customized play-lists generated in the desktop computer were downloaded into the PDAs to determine the actual songs the PDAs play.

Subjects in the experiment are undergraduates and graduate students from an eastern United States university who volunteered to participate in the experiment in the period of April and May 2007. Subjects who finished the whole experiments were given monetary rewards in addition to the chance of winning a Palm TX PDA. The 86 subjects who participated in the experiment were solicited using emails without any screening based on demographic criteria. Of the subjects 77% were aged between 18 to 21 years old, 20% were aged 22 to 29 years old, while the rest were 30

years old and above. In addition, 37% of the subjects are male, while 63% of the subjects are female.

Subjects who signed up for the experiment knew that the experiment involved understanding music preferences. They were informed that the experiment involved two waves of listening, and that their listening behaviors were recorded. About one-third of the subjects were randomly chosen to be in the control group. The subjects in the control group were subjected to the same procedures as the experimental group except that their play-lists are customized using a clustering procedure. For the subjects in the experimental group, their play-lists were customized using the procedures described in our model.

The data captured in the experiment are records of subjects' listening duration for the songs played from the PDAs. 16,835 data points are collected, where each data point represents an instance in which a particular subject had listened to a particular song. To collect the data while the subjects were in their "natural" situations, the PDAs were issued out to the subjects during the duration of the experiment. After the subjects had collected the PDAs, they had the liberty to decide when and where to listen to the songs. There were two waves of listening for each subject, with each wave lasting five days each. There was a gap of two days in between the two waves so as to allow the researchers to retrieve the data from the PDA from the first wave, analyze the first wave data, and finally customized the play-lists for the second wave. Due to the limitation on the number of PDAs available, the subjects were broken up into three different batches. The running of the experiment for these three batches was completed in one and a half months. To supplement the

data collected through the PDAs, subjects were asked to complete up a short survey, shown in the appendix, after the whole experiment was over to rate the quality of songs in the play-lists both in wave one and two.

Play-lists were used to determine which songs to play in the PDAs. Each play-list holds the name of 50 different Mp3 songs out of the universal set of 400 unique Mp3 songs used for the experiment. At the first wave of listening, randomly generated initial play-lists were used to determine which songs to play to the subjects. These initial play-lists were generated randomly and were not customized to the preference of the subjects due to the lack of prior preference data.

The songs were played sequentially in the PDAs, and the subjects had the option of skipping over the song if they disliked them. In addition, the participants also had the options of pausing the current song played, resuming the playing of the paused song, and stopping the song altogether. The subjects however, were not given the option of deciding which song should be played next, to prevent order effects. The duration data captured indicate how long a song is listened to until the subject consciously chose to skip or stop the song that is presently playing. Five days after the PDAs were issued to the subjects for the first wave of listening, the PDAs were returned for data download. The duration data from the mobile music devices was downloaded into the desktop computer for preference estimation and play-list customization. Customized play-lists were generated for the second wave. These play-lists were generated using either our customization procedure or a benchmark customization procedure. The benchmark procedure used a clustering algorithm and is described in more details in the next sub-section. Two days after the PDA were

returned from the first wave, the subjects collect the PDAs again for the second wave of listening.

Section 2: Estimation and customization procedures

The estimation and customization procedures based on our model were used to create the second wave play-lists for the subjects in the experimental group. For those subjects in the control group a clustering procedure was used. The estimation and customization procedures based on our model were carried out in the following steps: (1) 500 particles using the first wave's data for each subject were generated, where each particle was a vector containing the estimated values of a subject's parameters. These particles were generated using the data on listening duration and song attributes similar to the ones we used in the simulation section. These song attributes consist of variables that described the genre that a song belongs to in addition to the characteristics of the songs. 2500 iterations were used for the MCMC procedure with a burn-in of 2000. The variable selection step was also applied to the MCMC procedure. (2) An aggregate model was estimated to generate song-specific constants. 500 particles were generated for the aggregate model. In the context of the aggregate model, each particle corresponds to a vector containing an estimate of the different song constants. (3) After the individual and aggregate models' particles were estimated, the model averaging step was used to predict the listening duration of each subject for the 400 different songs. In the process, subject weights were generated. These weights were used to determine how much of one subject's parameter should be used to predict another target subject's listening duration. (4) The Bayesian Experimental Design procedure was used to customize the play-list for

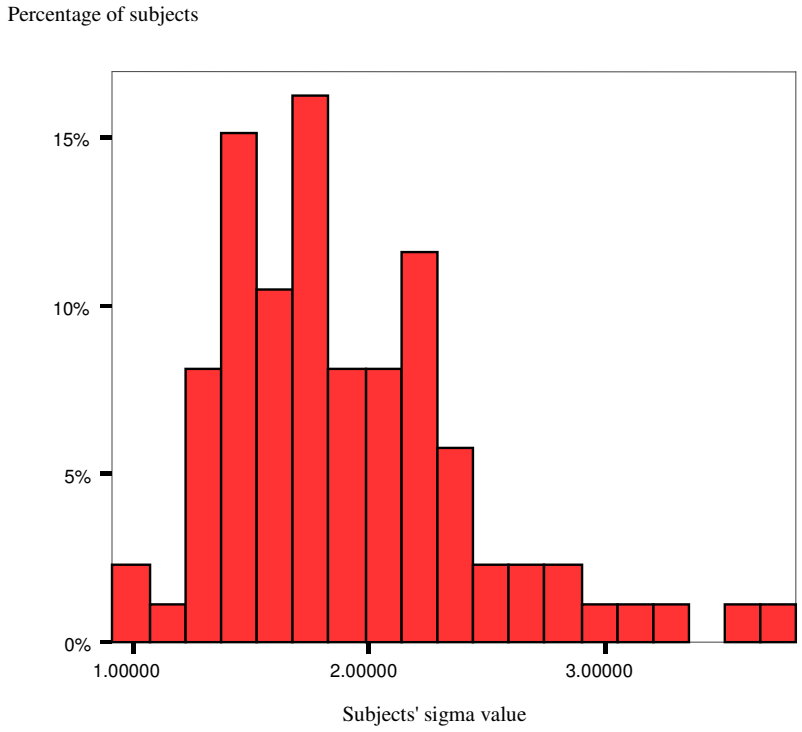
the individual subjects. Similar to the simulation, the weights used in the optimization criteria were such that the system converges to a maximum utility design after ten customizations.

We used a heuristic clustering algorithm as the benchmark to compare our model against. Such a heuristic procedure is conceptually similar to the one used by for example Pandora.com, an existing internet radio recommendation system. For the purpose of the benchmark algorithm we inferred that a subject likes a particular song if the subject finished listening to it during the first wave of listening. The clustering procedure generated a play-list for this subject in the second wave by choosing the songs that were most similar to the preferred songs. Similarity was measured using the Gower distances between one song from the next in the song attribute space. The clustering procedure minimizes the total Gower distances of the 50 songs in the play-lists.

Section 3: Results of the experiment

The lognormal hazard function is used in this paper to model subjects' listening behavior. For the lognormal hazard function, the standard deviation of the distribution determines the function's shape. A small sigma value (e.g. 0.5) describes a non-monotone hazard function, and as the value of sigma increases (e.g. around 2.0) the hazard function adopts a shape closer to a downward sloping curve (Figure 1). The distribution of 86 subjects' sigmas based on the first wave (i.e. standard deviations of the estimated parameters) is shown in figure 5.

Figure 5 Sigma values of the estimated experiment subject's parameters



Given that the sigma values of in figure 5 is closer to 2.0 then 0.5, a downward sloping hazard function generally describes the subject's listening behavior. This implies that the subjects' do not need to listen to the songs for long before they could decide whether a song is preferable. In addition, the tendency to continue listening to a song increases as the listening duration increases. Figure 6 and 7 gives an indication on how the subjects chose which songs to listen to. Figure 6 shows how long in milliseconds that the songs are listened to by the subjects. Figure 7 show the ratio between listening duration and the length of the songs. Both figures show that apart from the songs which the subjects chose to finish, the portion of songs that was skipped early was higher than the portion of songs that are skipped later into the song. These two figures show further that the subjects make a quick decision on which songs to finish listening to.

Figure 6 Percentage distribution of songs heard by the duration heard in milliseconds

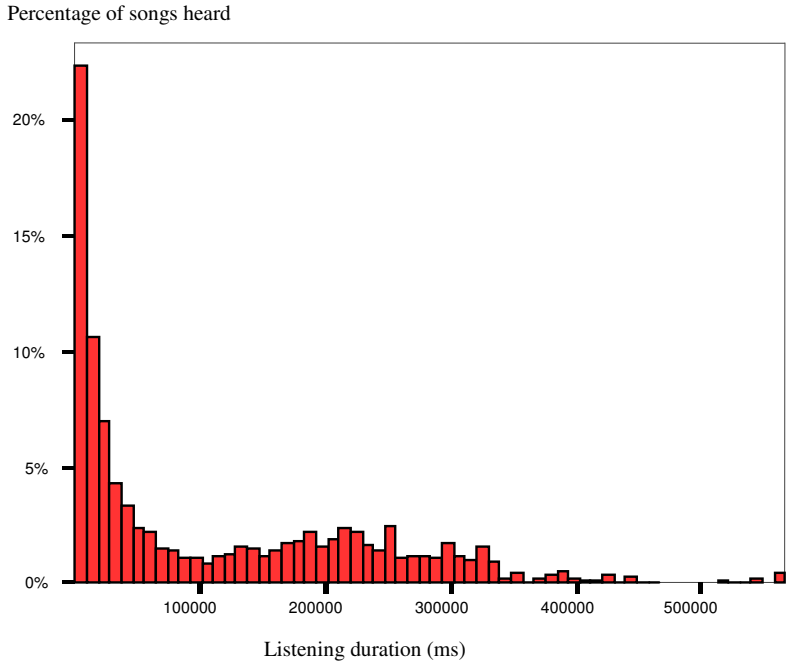


Figure 7 Percentage distribution of songs by the ratio of listening duration to song length..

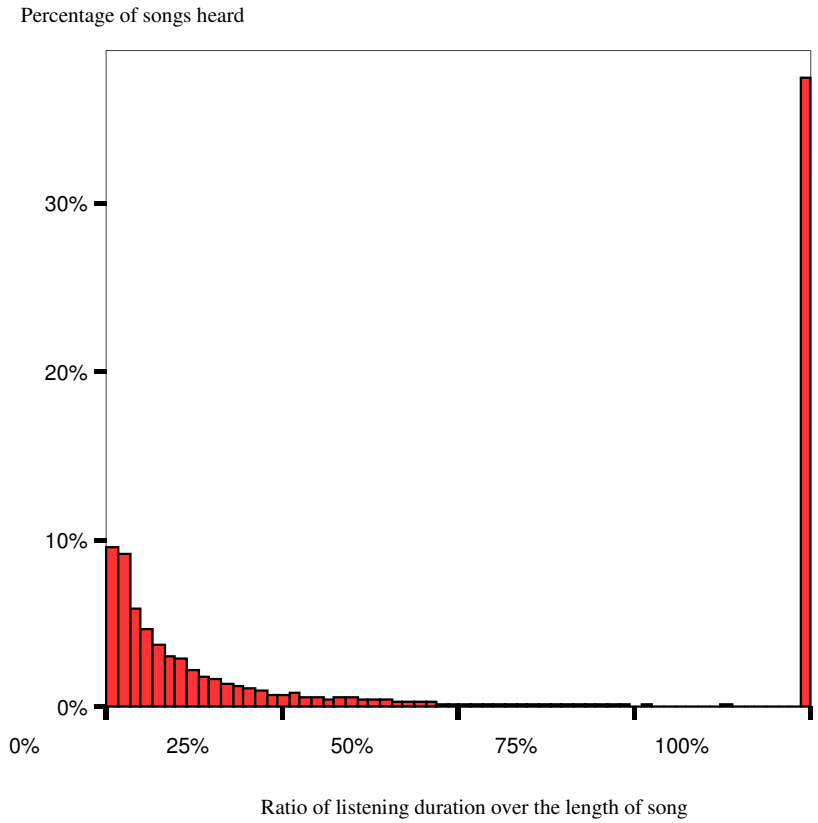


Figure 8 and 9 demonstrate the effectiveness of the variable selection procedure. They show the distribution of the subjects by the number of parameter estimates which are set to zero by the variable selection procedure. Figure 8 shows the distribution for the six variables that describe the song characteristics (e.g. perceived loudness of the song, song tempo, etc). Figure 9 shows the distribution for the eleven variables that describe the song genres (e.g. Jazz, pop, etc.)

Figure 8 Distribution of experimental subjects by the number of non-zero song characteristic coefficients.

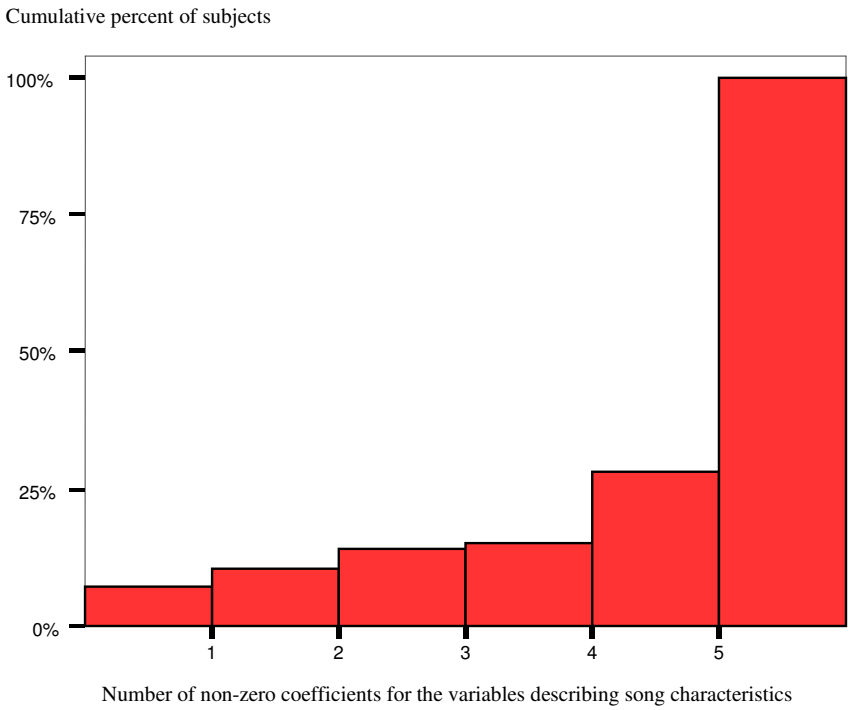
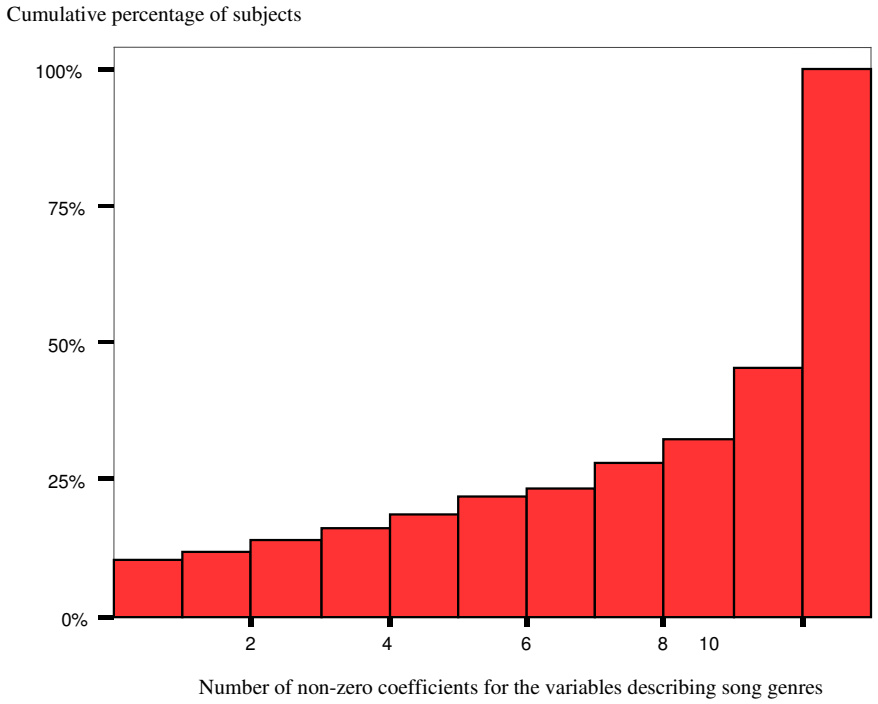


Figure 9 The distribution of experimental subjects by the number of non-zero coefficients for describing song genres.



From figure 8 and 9 we can see that the preferences of about 14% of the subjects can be described using only half of the song characteristics variables (figure 8) , and the preferences of about 20% of the subjects can be described using only half of the variables describing genres (figure 9). For these subjects, the variable selection eliminates the variables that are redundant and therefore simplifies the parameter estimation and predictions.

The Sequential Monte Carlo estimation procedure is used to deal with the challenges of analyzing big dataset. In our experiment, the computation time using Sequential Monte Carlo methods is about 15% of the MCMC methods when the same dataset is used. A standard Markov Chain Monte Carlo method requires a complete scan of the dataset and also a large number of iterations to estimate the model parameters. Computational time for the MCMC method increases greatly with the

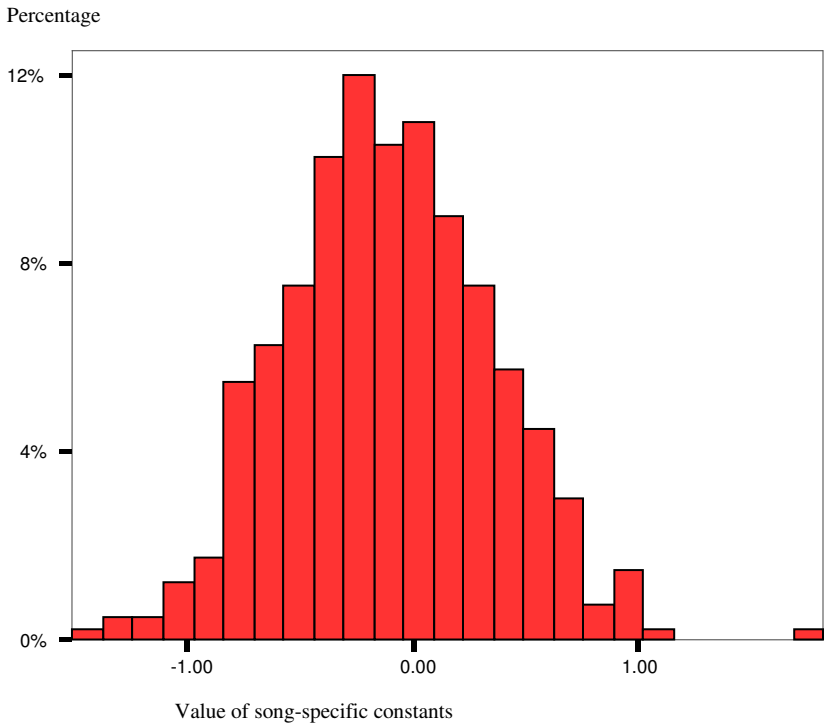
size of the dataset. The ability to break datasets into blocks helps to keep the computational time of the Sequential Monte Carlo method short. Therefore, the benefit of the Sequential Monte Carlo Method is more pronounced when the size of the dataset is greater than the dataset we have in our experiment.

We used the aggregate model to capture the impact of individual song's unique characteristics which are not otherwise reflected in the prediction by the individual level models. This aggregate model uses a dummy variable for each song across all subjects. The coefficients of the aggregate model therefore represent the song-specific constants. The distribution of the song-specific constants is shown in table 7 and in figure 10. The song-specific constants follow a normal distribution with a high variance. The normal distribution is a result of the use of a normal prior when we estimated the constants. We infer that the 400 songs that we used for our experiment represents a diverse collection of songs with very different characteristics. We have deliberately chosen the songs to be greatly different in order to better cater to the diverse taste in music among the experiment subjects.

Table 7 Distribution of the song-specific constants..

Mean	Median	Min	Max	25th Percentile	50th Percentile	75th Percentile	Standard Deviation
-0.11	-0.12	-1.50	1.82	-0.42	-0.12	0.20	0.46

Figure 10 Distribution plot of the song-specific constants.



Our motivation for using model averaging is to improve our predictions of a subject's listening duration. We combine individual models (i.e. sets of parameter estimates) based on how well they predict the target individual's data. The weight that we place on a model in the prediction of another subject's data is based on the ratio of the likelihood of that model over the subjects' own model. We calculate the between-subject weights for all combination of subject while treating the aggregate model as one of the candidate models. Table 8 below shows the distribution of subject weights.

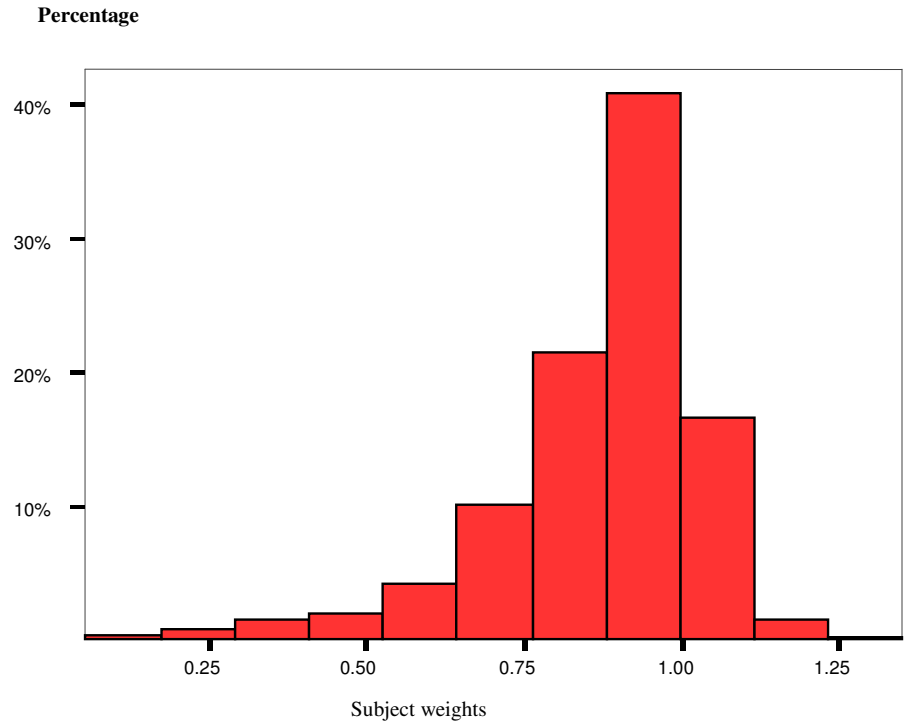
Table 8 Distribution of subject weights.

Mean	Median	Minimum	Maximum	Standard Deviation
0.87	0.91	0.05	1.35	0.19

On average a weight of about 0.87 is applied to other subject's parameters to predict the target subject listening duration. There are instances where one subject's

parameters are poor predictor of another subject's listening duration (i.e. when the weight is at 0.05), but in other instances another's subject's parameters maybe a better predictor for the target's subject preferences. It is in the instances when the weights are above 1.00 that the model averaging approach helps to improve the predictions of the subjects' listening duration substantially. This happens in about 16% of the cases. The distribution of the subject weights is shown in figure 11.

Figure 11 Distribution of subject weights.



In table 9, we compare the performance of the play-lists in terms of how well they recommend songs to subjects. The performances of the random play-lists, and also the play-lists generated using of our model and the benchmark model are shown. The comparisons are done using two criteria: (1) the percentage in which the subjects listened to each of the songs presented to them. (2) The proportion of all the songs presented to the subjects which they finished listening.

Table 9 Comparison of play-lists' performance in terms of how well they recommend songs.

	Mean percentage of each song that the subjects have listened to			Mean proportion of the songs that the subjects have finished Listening to		
	(1)	(2)	(3)	(4)	(5)	(6)
Play-lists generated by our model	55.00%	55.00%		0.46	0.46	
Play-lists generated by benchmark model	44.79%		44.79%	0.36		0.36
Random play-lists		44.09%	40.57%		0.34	0.30
Difference in mean	10.21%*	10.91%*	4.22%*	0.10*	0.12*	0.06*
Percent improvement	22.78%	24.74%	10.41%	27.78%	35.29%	20.00%

Note: * reflects < 0.01 significance

Based on the results shown, our model performs 23% better than the benchmark model in terms of percentage of each song listened to (Column 1, table 9) ,and 28% better for the proportion of songs that the subjects have finished listening (Column 4, table 9). This comparison is done using the data in the second wave of listening because it is the data based on customized play-lists.

When we compare the listening behavior of the subjects before and after we have customized the play-lists (i.e. comparison of wave one and wave two data), the play-lists customized using our model results in a 25% (column 2, table 9) improvement in the percentage of each song listened to, and a 35% improvement in the proportion of songs that the subjects finished (column 5, table 9).

Comparatively, the play-list customized using the benchmark model results in a 10% improvement in the percentage of each song listened to (column 3, table 9), and a 20% improvement in proportion of songs that the subjects finished (column 6, table 9).

Looking at the relative ratios of improvement, our model is 2.59 (i.e. 10.91% / 4.22%) times better than the benchmark model in increasing the percentage of each song listened to, and is 2.00 times (i.e. 0.12 / 0.06) better than the benchmark model

in increasing the number of songs that the subjects finished. These results show that our model is indeed better than the benchmark model.

To provide an indication of the best case and worst scenarios for the performance of our and the benchmark recommendation system we analyze the recommendation performances on an individual basis. For our model the changes in the percentage of each song listened to before and after the play-lists are customized range between -42.41% to 184.14% and have a median of 8.85%. Comparatively, the range for the benchmark model is between -64.40% and 124.32% with a median of 3.29%. In terms of the changes in the proportions of songs that the subjects finished our model has a range between -50.98% and 373.12% with a median at 13.08%. The respective percent changes for the benchmark model are -66.94% and 239.85% respectively. The median for the changes in proportion for the benchmark model is 8.32%. This means that our model has a less severe worst case scenario and a better best case scenario compared to the benchmark model both for the percentages of each song listened to and the proportions of songs that the subjects finished. By using the median and as a result reducing the effect of outliers, our model still performs better than the benchmark model.

When we asked the subjects to indicate how well the play-lists perform using a post experiment survey, we are not able to see any significant difference in the way the subjects rate our model and the benchmark model, and in the way the subject rate the play-lists before and after they are customized. The actual survey used is shown in the appendix, and the results of the survey are shown in table 10. For the rating on subjects' song satisfaction, a value of 1 reflects that the subjects are very satisfied,

and a value of 7 reflects that they are very unsatisfied. The values in table 10 are around 3.0, which corresponds to the average rating of "somewhat satisfying". For the rating on the proportion of songs the subjects liked, a rating of 1 reflects that subjects like all the songs, and a rating of 5 reflects that subjects like none of the songs. A value close to 3.0 as shown in table 10 indicates that the subjects like only some of the songs in the play-lists. None of the differences in ratings are significant. The problem of using rating data in making recommendations is discussed in Rossi, Gilula and Allenby (2001). In this paper the authors show that rating data do not always indicate the true preferences. In our case, subjects reported ratings are not useful in making any preference predictions.

Table 10 Subjects' ratings of the play-lists' performance obtained from post experiment survey

	Satisfaction with the songs			Proportion of songs liked		
	(1)	(2)	(3)	(4)	(5)	(6)
Play-lists generated by our model	3.63	3.63		3.11	3.11	
Play-lists generated by benchmark model	3.59		3.59	3.14		3.14
Random play-lists		3.86	3.55		3.26	3.17
difference in mean	0.04	-0.23	0.04	-0.03	-0.15	-0.03

Note: For satisfaction, a rating of 1 reflects very satisfied and a rating of 7 reflects very unsatisfied. For the proportion of songs liked, a rating of 1 reflects that subjects like all the songs, and a rating of 5 reflects that subjects like none of the songs. None of the differences in ratings is significant.

We investigate further into the results of our satisfaction measure by correlating the actual listening behaviors of the subjects with their responses in the post experiment survey. The satisfaction levels with the songs generated by the customized play-lists indicated in the survey correlate significantly with the actual

listening behavior when we combine the survey results of the subjects in the control and the experiment group. The correlation between the satisfaction levels and the percentages of each song listened to in the second wave is at -0.31, and the correlation between satisfaction levels and the proportions of songs that the subjects finished in the second wave is at -0.30. The correlations are negative because smaller values for the satisfaction measure indicate higher levels of satisfaction. Both of the correlations are significant at the 0.01 level. The results are different when we correlate the satisfaction levels with the songs in the second wave and the actual listening behaviors separately for the play-lists generated using our method and the play-lists generated by the benchmark model. For the benchmark model the correlations are significant at the 0.01 level. The correlations are -0.40 between satisfaction level and the percentages of each song listened to, and -0.39 between satisfaction level and the proportions of songs that the subjects finished in the second wave. The corresponding correlations for our model is at -0.11 and -0.15 respectively. Not only are the correlations smaller for the play-lists generated using our model, the correlations are also not significant. The performance of our model is shown to be better than the benchmark model. The fact that correlation between satisfaction level and actual listening behavior is significant for the benchmark model and not our model indicates that the subjects are stricter in indicating their satisfaction level when a recommendation system is performing better in contrast to a recommendation system that is performing poorer. This provides some explanations as to why our model does not result in a higher satisfaction level when compared to the benchmark model. This is inline with Gilula and Allenby (2001) observation that

rating data do not always indicate the true preferences. We run into the same problem when we look at the correlation between the proportions of songs that subjects like indicated in the survey and the actual listening behaviors. The correlations are significant at the 0.01 level for the play-lists generated using our model and are non significant for the play-lists generated using the benchmark model.

The subjects indicated that they are wiling to pay about $6.65 on average for the recommendation system. This is lower compared to the $9.99 that Rhapsody offers for the song download. The lower willingness to pay for our system is explained partly by the fact that subjects do not always indicate their true preferences in a survey. For example, we did not find any significant results between willingness to pay and satisfaction level, and between willingness to pay and actual listening behaviors. A spontaneous remark on the willingness to pay was given by a subject in the survey form. She remarked: “Free as a first time product trial. Maybe later if I like the service, I would consider upgrading my subscription for $5.00 extra a month”. This provide a possible explanation that apart from the problem of the subjects not indicating their true preferences in the survey, another possible reason for the lower willingness to pay is that the subjects may not see the recommendation system as a separate product but as a possible value add to existing system. In addition, subjects may want a trial period to further evaluate the performance of our recommendation system before they can provide a more accurate indication of their willingness to pay.

We move on to discuss our results in the next section and also provide a conclusion to this paper.

Chapter 5: Discussion and conclusion

The main objective of this paper is to develop a music recommendation system that automates the downloading of songs into a mobile digital audio device. We have taken the approach that involves as little effort as possible from the consumers' end. This provides some challenges to our recommendation system. First, we cannot use consumer demographic profiles or prior music preference questions to help in the generation of the initial play-list. Our use of a randomly generated initial play-list is due to the belief that asking for personal information may be considered an invasion of privacy, and may discourage the adoption of the recommendation system. However, we could in principle use such information to initialize our recommendation system. Second, we choose not to involve individuals when they visit the music downloading website, for example by asking them to indicate which songs they prefer after listening to a collection of music snippets. We have not incorporated this feature due to our desire to minimize the effort required from individuals during the process of music recommendation and downloading. Nevertheless, this could in principle be incorporated in an extension of our system. Third, we have also kept our Mp3 software interface simple to reduce the effort needed to operate the mobile music playing device. On the reverse side, subjects do not have the option to choose which songs they want to listen to.

The simulation section of this paper demonstrated that our model does achieve it objectives in handling massive data and improving predictions through model

averaging. The speed comparison between MCMC and Sequential Monte Carlo was shown to be substantial. By using simulated data in the simulation, and thus knowing the true parameters, the Sequential Monte Carlo and variable selection procedures were shown to provide good estimates of an individual's preferences. Experimental results show that variable selection does simplify estimation and prediction as different individuals differ in the number of variables need to definite their listening behaviors. The results also show that for some individuals, model averaging does in fact help to improve predictions.

Looking at the results of the experiment, our model provides 23 – 35% improvement in recommendations. This improvement is achieved in a single wave and in a natural experimental setting in which the subjects have a choice or when, where and how they want to listen to the songs. Running the experiment in a natural setting brings with it a set of new challenges. First, unlike recommendation systems based on secondary data we did not have the options of fine tuning our model in the midst of our experiment. Second, the listening context in which the subjects are in can be very different when they listening to the songs in the first wave versus the context they are in for the second wave. For example, a subject could be listening to the songs mostly when they are working out in a gym in the first wave, but chose to listen to the songs mostly when the subject is driving in the second wave. Preferences may change with the listening context and thus reducing the effectiveness of our recommendations. We expect such changes to diminish over time as recommendations are made in multiple waves. Future extension for our model could potentially address the issue of context effect on listening preferences. Our

experiment results show that individuals make their choices on which songs they prefer quickly. In addition, given the great speed in which the Sequential Monte Carlo procedure updates parameter estimates, it can be programmed into the mobile music playing device itself. The system then becomes a distributed computing system incorporating parallel computation on the PDAs for individual subjects. After an individual's particles is downloaded from the website into the music device, the device can effectively chose which of the potential multiple play-lists to use after observing several of the individual choices. Third, we have less data points per subject per wave of listening compared to the 500 data points per subjects used in our simulation. Instead of the 500 data points, we have on average only 98 data points per subject per wave of listening, which translates to about 3 hours of listening on average. This resulted from the fact that we did not impose any total listening duration requirement on our subjects. The lower number of data points affects the accuracy of our recommendation system. We have also not utilized a maximum utility design when we recommend our songs, but one in which utility and precision are traded off in a single recommendation wave. We have deliberately incorporated precision into the criteria used to recommend a composition of songs because this is how the system operates in real-life conditions. Even in the simulation, the use of precision in this way reduces our design utility by about 3%.

In conclusion, we believe that have achieve our objectives for developing a full working and directly implementable recommendation system. The system achieves dramatic improvements over a realistic and heuristic one, that itself has not yet been implemented as far as we are aware of. This system is one that automates

the downloading of songs into a mobile digital audio device based just on past listening behavior. One immediate refinement that can be done to our model is to improve the attributes used to differentiate one song from the next. The music genome project which characterizes different music with a different music "DNA" is an natural direction to take in achieving this goal. Possible immediate application of our recommendation system is to incorporate it into the itune website, in which ipods are used as the music playing and data collection devices, but we envision many other useful applications.

Appendix: Survey on Music Preference Study

To help us better understand your listening experience during the study please take a few minutes to complete this survey.

Question 1 and 2 applies to your listening experience in the last five days. Please place a check in the circle that best describes your listening experience.

Question 1:

What proportion of the sings do you like?

O All O Most O Some O Not many O None

Question 2:

How satisfied are you with the songs?

O Very Satisfied O Satisfied O Somewhat satisfied

O Neither Satisfied or disatisfied O Somewhat dissatified

O Dissatified O Very dissatified

Question 3 and 4 applies to your listening experience in the first five days. Please place a check in the circle that best describes your listening experience.

Question 3:

What proportion of the sings do you like?

O All O Most O Some O Not many O None

Question 4:

How satisfied are you with the songs?

O Very Satisfied O Satisfied O Somewhat satisfied

O Neither Satisfied or disatisfied O Somewhat dissatified

O Dissatified O Very dissatified

Question 5:

If the online music company decides to launch the website, it plans to provide its services through a monthly subscription plan. Consumers who subscribe to the service will have unlimited song downloads. As a comparision, Rhapsody (an existing music website) offers unlimited song download at a fee of $9.99 a month.

How much would you be willing to pay for the monthly subscription to the website that offers automatic download of music?

US$ _____

References

Ansari, Asim, Carl F. Mela. 2003. E-Customization. *Journal of Marketing Research*. **40**(2), 131-145.

Ansari, Asim, Skander Essegaier, Rajeev Kohli. 2000. Internet Recommendation Systems. *Journal of Marketing Research*. **37**(3), 363-375.

Ariely, Dan, John G. Lynch. Jr., Manuel Aparicio IV. 2004. Learning by Collaborative and Individual-Based Recommendation Agents. *Journal of Consumer Psychology*. **14**(1&2), 81-95.

Arulampalam, Sanjeev, Simon Maskell, Neil Gordon, Tim Clapp. 2002. A Tutorial on Particle Filters for On-Line Non-linear/Non-Gaussian Bayesian Tracking. *IEEE Transactions on Signal Processing*. **50**(2), 174-188.

Chaloner, Kathryn, Isabella Verdinelli. 1995. Bayesian Experimental Design: A Review. *Statistical* Science. **10**(3), 273-304.

Chipman, Hugh, Edward I. George, Robert E. McCulloch. 2001. The Practical Implementation of Bayesian Model Selection. *Model Selection.* Institute of Mathematical Statistics, Beachwood, Ohio. **38,** 65-116.

Cook, R. Dennis, Christopher J. Nachtsheim. 1980. A Comparision of Algorithms For Constructing Exact D-Optimal Design. *Technometrics*. **22**, 315-324.

Ford, Ian, D. M. Titterington, Christos P. Kitsos. 1989. Recent Advances in Nonlinear Experimental Design. *Techometrics*. **31**(1), 49-60.

Hoeting, Jennifer A., David Madigan, Adrian E. Raftery, Chris T. Volinsky. 1999. Bayesian Model Averaging: A Tutorial. *Statistical Science*. **14**(4), 382-417.

Holbrook, Morris B., Meryl P. Gardner. 1993. An Approach to Investigating the Emotional Determinants of Consumption Durations: Why Do People Consumer What They Consumer For As Long As They Consumer It? *Journal of Consumer Psychology*. **2**(2), 123-142.

Kiefer, J., J. Wolfowitz. 1959. Optimum Designs in Regression Problems. *The Annals of Mathematical Statistics.* **30**(2), 271-294.

Kong, Augustine, Jun S. Liu, Wing Hung Wong. 1994. Sequential Imputations and Bayesian Missing Data Problems. *Journal of the American Statistical Association*. **89**(425), 278-288.`

Konstan, Joseph, Bradley N. Miller, David Maltz, Jonathan Herlocker, et al. 1997. Grouplens: Applying Collaborative Filtering to USENET news. *Communications of the ACM*. **40**(3), 77-87.

Montgomery, Alan L., Kartik Hosanagar, Ramayya Krishnan, Karen B. Clay. 2004. Designing a Better Shopbot. *Management Science*. **50**(2), 189-206.

Pronzato, Luc, Éric Thierry. 2003. Sequential Experimental Design and Response Optimisation. *Statistical Methods & Applications*. **11**, 277-292.

Raghu, T.S., P.K. Kannan, H.R. Rao, Andrew B. Whinston. 2001. Dynamic Profiling of Consumers for Customized Offerings Over the Internet: A Model And Analysis. *Decision Support System*. **32**(2001), 117-134.

Rentfrow, Peter J., Samuel D. Gosling. 2003. The Do Re Mi's of Everyday Life: The Structure and Personality Correlates of Music Preferences. *Journal of Personality and Social Psychology*. **84**(6), 1236-1256.

Ridgeway, Greg, David Madigan. 2003. A Sequential Monte Carlo Method for Bayesian Analysis of Massive Datasets. *Data Mining and Knowledge Discovery*. **7**(3), 301-319.

Rossi, Peter E, Zvi Gilula, Greg M. Allenby. 2001. Overcoming Scale Usage Heterogeneity: A Bayesian Hierarchical Approach. *Journal of the American Statistical Association*. **96**(453), 20-31.

Saha, Atanu, Lynette Hilton. 1997. Expo-power: A Flexible Hazard Function for Duration Data Models. Economic Letters. **54**,227 – 233.

Sha, Naijun, Mahlet G. Tadesse, Marina Vannucci. 2006. Bayesian Variable Selection for the Analysis of Microarray Data with Censored Outcomes. BioInformatics. 22(18), 2262 – 2268.

Verdinelli, Isabella. 1992. Advances in Bayesian Experimental Design. *Bayesian Statistics 4*. Oxford University Press. 467-481.

Verdinelli, Isabella, Joseph B. Kadane. 1992. Bayesian Designs for Maximizing Information and Outcome. *Journal of American Statistical Association*. **87**(418), 510-515.

Ying, Yuan Ping, Fred Feinberg, Michel Wedel. 2006. Leveraging Missing Ratings to Improve Online Recommendation Systems. *Journal of Marketing Research*. **43**(3), 355-365.