DEGREE PROJECT IN INFORMATION AND COMMUNICATION TECHNOLOGY,
SECOND CYCLE, 30 CREDITS
STOCKHOLM, *SWEDEN 2020*

Analyzing music genre classification using item response theory

A case study of the GTZAN data

BENEDITH MULONGO

KTH ROYAL INSTITUTE OF TECHNOLOGY
SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

Analyzing music genre classification using item response theory

BENEDITH MULONGO

Master in Machine learning
Date: July 16, 2020
Supervisor: Bob sturm
Examiner: Sten Ternström
School of Electrical Engineering and Computer Science
Swedish title: Att analysera klassificering av musikgenre med hjälp av item respons teorin

Abstract

Machine learning models are usually evaluated by metrics such as accuracy, confusion matrix, recall. Those metrics are used in evaluation methods for assessing the models. However, those metrics used for assessing the models give no information regarding the ability of the models or the difficulty of the test data used to evaluate the given models. Without a reliable measure of the difficulty of test data it is harder to obtain an accurate evaluation of the performance of the models.

Some of the limitations previously mentioned regarding the widely used metrics in current evaluation methods in machine learning may be overcome by using *Item response theory*. Item response theory is a psychometric framework used to simultaneously estimate the ability of the examinee and the difficulty of the test. This framework enables the comparison of test items by the mean of their difficulty and the comparison of examinees by the means of their respective ability. This thesis focuses on the applications of item response theory in machine learning and specifically applications of item response theory for evaluating music genre classification.

In particular, a case study analysis of the GTZAN music data is made. The GTZAN dataset is a dataset composed of 1000 audio tracks with a duration of 30 seconds. The GTZAN dataset have been documented to possess many flaws such as the fact that many artists appear frequently within the same genre category. Therefore, a control study of the GTZAN data is made, where one group of classifiers is trained with randomly stratified data and the second group is trained with data where the training and the test data do not have songs from a common artist. The responses patterns from those two group of classifiers are analyzed using item response theory.

Sammanfattning

Maskininlärningsmodeller utvärderas vanligtvis med mätvärden så som noggrannhet, förväxlingsmatris, återkallelse. Dessa mätvärden ger varken information om kvaliteten på modellerna eller svårighetsgraden för de testdata som används för att utvärdera modellerna. Utan denna information kan ingen tillförlitlig jämförelse göras mellan olika test data och/eller olika modeller.

Vissa av de tidigare nämnda begränsningarna avseende de aktuella utvärderingsmetoderna i maskininlärning kan övervinnas genom att applicera *Item respons teori*. Item respons teori är ett psykometriskt ramverk som används för att samtidigt uppskatta förmågan av en student och svårighetsgraden för en test. Detta ramverk möjliggör jämförelse av test data med hjälp av dess svårighetsgrad och jämförelse av studenterna med hjälp av deras respektive förmåga. Denna uppsats fokuserar på möjliga tillämpningar av item response teori i maskininlärning och specifikt dess tillämpningar för klassificering av musikgenren.

Närmare bestämt görs en fallstudieanalys av GTZAN-musikdata. GTZAN har dokumenterats innehålla många brister, till exempel det faktum att många artister ofta förekommer i samma genre kategori. Därför görs en kontrollstudie av GTZAN-data, där en grupp klassificerare tränas med slumpmässigt stratifierade data och den andra gruppen tränas med data där träningsdata och testdata inte har låtar från en gemensam artist. Svarmönstren från dessa två grupper av klassificerare analyseras med hjälp av item response teori.

Contents

Chapter 1

Introduction

1.1 Introduction

An operationalization of a measurement framework is required in order to measure a latent (hidden) trait such as an examinee's reading proficiency, mathematics ability and health. Operationalization works through the identification of reliable and valid indicators. Those indicators can then be combined to constitute a test instrument to be administered to examinees for assessing their ability.

A test instrument can be assessed by the means of the classical test measurement model. In the classical test measurement framework especially in quantitative studies, a person's ability is assessed by computing the total amount of correct answers or correct test items. However, the classical test measurement framework has major drawbacks such as its inability to generalize beyond the given test and its incapacity to estimate the difficulty of the given test instrument. There is indeed a lack of generality because the estimated ability of an examinee varies depending on the given test; moreover, the difficulty of a test varies depending on the group to which it is assigned [1]. Those inherent limitations increase the difficulty to perform reliable comparisons between groups and between tests without a prior test equating.

Item response theory is a psychometric measurement framework for evaluation of tests and examinees. It is used to find group-invariant and item-invariant characteristics of tests and examinees. Group-invariance means that an item's characteristics such as its difficulty is independent of the group's ability from which the characteristics parameters are estimated; on the other hand, item's invariance means that the estimated ability of an examinee is independent of the test from which the ability is estimated [2]. The two invariances

are called the invariance principle of item response theory. The invariance principle enables comparisons between test and examinees. The same principle allows the representation of the examinee's ability and the item's difficulty on the same latent space. Furthermore, there are two assumptions in item response theory, which are unidimensionality and local independence. Unidimensionality means that there is a single latent variable (the ability) that defines the observations on the manifest variable (the item test); contrariwise, local independence means that the examinee's performances on different test items are independent given the ability [3].

IRT's properties such as group-invariance and item's invariance can be advantageous in model evaluation in different fields such as machine learning. For example, in machine learning, classification is one of the widely used methods. In classification there are two main components that is the classifier and the dataset. The dataset is split into three distinct parts: train, test and evaluation data. The train data is used for training the classifier, the evaluation data is used for improving the model and tuning the hyperparameters, finally the test data is used for measuring the performance of the classifier on "unseen" test items.

Machine learning models can be evaluated using metrics such as accuracy, recall, confusion matrix, roc curve. For instance, accuracy is simply a percentage measure of the number of correctly classified test items over the total number of test items. The accuracy metric assumes that all test items have the same difficulty; furthermore, the comparison between different models is based solely on their total score. It should be noted that evaluation methods based on metrics such as accuracy resemble in many aspects to the aforementioned classical test measurement. As for classical test measurement, without a reliable estimation of the test items difficulty, the model performance may not be reliably estimated.

The item response theory framework provides a way to simultaneously estimate the test items' difficulty and the examinees' ability. IRT can be successfully applied in machine learning as an alternative evaluation method by reformulating the trained classifiers as the examinees and the test data as the test items. This reinterpretation gives the possibility of estimating the ability of the classifiers and the difficulty of the test items such as it is possible to compare different models by analyzing their behavior on respectively easier and harder items. Furthemore, the test classifiers ability is independent of the test on which it is estimated on (item-invariance) and the test items difficulty is independent on the classifiers on which it is estimated (group-invariance).

IRT has been applied in machine learning in few cases in the literature.

For instance, the paper by Fernando Martínz-Plumed et al. [4] in which the authors evaluate item response theory in machine learning by analyzing the behaviour of popular classification methods in machine learning using thirteen different datasets. The same authors have also published different articles on the same topic such as the paper [5] and [6]. Another relevant document is the master thesis by Stuart Jones [7]; which is a survey of IRT's applications using a machine learning perspective. Moreover, there are other existing published articles about machine learning and item response theory such as the paper [8] in which an application of neural network as an alternative method in IRT is discussed. Another article [9], in which the author analyse the item response theory framework for assessing deep-learning models.

1.2 Research Question

Although there are a few articles published about the application of item response theory in machine learning, few of them discuss specifically the application of IRT in classification problems. Moreover, the application of IRT in music genre classification and music information retrieval is not widely studied in the literature. However, given the previously mentioned advantages of using IRT instead of classical methods, it may be beneficial from an analytical point of view to investigate music genre classification with the GTZAN dataset using item response theory.

The GTZAN [10] music data is one of such data for which music genre classification is not yet analyzed under the IRT framework. Therefore, the research question is : to which extent item response theory can be applied to music genre classification using the GTZAN dataset in regard to the items difficulty and discriminability? Moreover, which resulting advantages can be observed when applying item response theory to machine learning in general and music genre classification in particular ? Which items are problematic for testing trained classifiers?

1.3 Sustainability

This project does not uses any valuable materials or products that may have negative impacts on the environment. The only materials used are computer resources and immaterial software products. It can therefore be concluded that questions related to sustainability do not directly apply to this project.

In my opinion, there is no detrimental economical impact of this project.

This project does not represent any immediate economical impacts; however, if further researches are done item response theory may be incorporated in a commercial software for analyzing machine learning models and datasets.

Regarding the societal sustainability, its application in this thesis is minimal. As aforementioned this project may influence the adoption of IRT techniques in the machine learning society. However, the applicability of the societal sustainability in this project remains minimal.

1.4 Ethical aspects

There is no conflict of interest in this project. This project is conducted as a part of the master program in machine learning and is considered as an original work of the author based on the idea and guidance from professor Bob Sturm.

However, copyright material such as music data are used for training the models and investigating the research questions. It should, however, be noted that each audio excerpts has only a duration of 30 seconds and it is a public data freely available on the internet [1].

The intellectual property is respected as closely as possible, by referencing to books and articles and mentioning authors and professors that have contributed to the project. Furthermore, the results are analysed in an objective way by mentioning weakness of the study and further works and improvements.

Given the fact that this study does not incorporate human survey or animals, many of the ethical aspects are not directly applicable other than the ones previously mentioned.

[1] The data is freely accessible at : `http://marsyas.info/downloads/datasets.html`

Chapter 2

Background

2.1 Theoretical background

2.1.1 Item response theory

IRT [2] is a psychometric measurement framework for evaluation of tests and examinees. It is used to find group-invariant and item-invariant characteristics of tests and examinees. Item response theory comprises two main groups: uni-dimensional and multidimensional item response theory. Uni-dimensional item response theory assumes unidimensionality which means that there is a single latent variable (the ability) that defines the observations on the manifest variable (the item test), e.g. there is a single latent mathematics proficiency that underlies the examinees' performance on a given item [3]. On the other hand, multi-dimensional item response theory [11] assumes that there is more than one latent variable, e.g. the examinees' performance is determined simultaneously by both the mathematics proficiency ability and the speed reading ability. In the case of multi-dimensionality, all latent variables influence the performance, therefore their combined analysis is required. In this thesis, the focus is on the uni-dimensional item response theory.

Apart from the dimensionality of the latent trait measured, item response theory can be divided into two groups : dichotomous and polytomous item response theory. In dichotomous IRT, there are only two response categories allowed for each items, namely zero and one. Zero represents the absence of the latent trait measured and one represents its presence. In short, one is a correct response and zero an incorrect one. On the other hand, polytomous item response theory has more than two categories.

Item response theory can be applied to a single-group population or to a multiple group population. When a test is administered simultaneously to two or more groups, multiple group item response theory may be applied for analysing the test items and the examinees' ability across groups. One of the advantages of using multiple group IRT is the ability to analyse how the items' difficulty vary across groups, which is called *differential item functioning*.

2.1.1.1 Main models

Uni-dimensional item response theory comprises four main models : Rasch, one-parameter logistic, two-parameter logistic and three-parameter logistic. The name of each model is related to the number of the parameters in the model; furthermore, logistic function is the underlying function for all models. A logistic function has the following mathematical form : $f(\theta) = \frac{1}{1+\exp L(\theta)}$ where $L(\theta)$ is a linear function of θ. The logistic function is used to describe the characteristics of an item, which can be represented graphically using a curve named item characteristic curve (ICC).

Rasch model is the simplest of all the models used in item response theory. Rasch model is a probabilistic method for test analysis, it was developed by the Danish mathematician Georg Rasch [11]. A historical and mathematical development of the Rasch model can be found in the book *Item Response Theory Parameter Estimation Techniques* [11]. Nonetheless, a general introduction of the Rasch model is required.

The equation for the Rasch model is described by the following logistic function:

$$P(u_{ij} = 1|\theta_j, \beta_i) = \frac{1}{1 + e^{-(\theta_j - \beta_i)}} \tag{2.1}$$

where :
i is the index of the $i : th$ item
j is the index of the $j : th$ examinee
θ is the ability level
β is the difficulty parameter

Equation (2.1) is the Rasch mathematical function in item response theory. It should be noted that the property of uni-dimensionality and local independence previously mentioned still holds for the Rasch model.

The relation $\epsilon = (\theta_j - \beta_i)$ from equation (2.1) can be analyzed as a distance measure between the examinee's ability θ_j and the item's difficulty β_i. If the examinee's ability is much higher than the difficulty of the item, then

the probability of a correct answer approaches one; however, if the examinee's ability is much lower than the difficulty of the item then the probability of a correct answer approaches zero. Furthermore, the probability of a correct answer approaches $\frac{1}{2}$ as the examinee's ability matches the item difficulty. The item difficulty is then the point where the probability of a correct answer is $\frac{1}{2}$ [2].

Figure 2.1 shows the item characteristic curve for two Rash models having respectively item difficulties $\beta = 0$ and $\beta = 1$. As it can be observed the item's difficulty parameter move the curve as shown by the vertical dashed lines in figure 2.1.

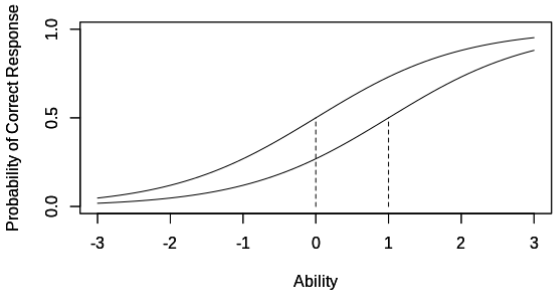

Figure 2.1: Item characteristic curve for two Rasch models $\beta = 0$ and $\beta = 1$

The one-parameter logistic model (1PL) is similar to the Rasch model in many aspects. The probability of a correct answer is almost equal except from the fact that the 1PL model has an additional parameter alpha α. The alpha parameter increases the level of complexity and adds an additional degree of freedom e.g. for fitting the logistic model to the data.

The mathematical expression for the one-parameter logistic model (1PL) is given by :

$$P(u_{ij} = 1|\theta_j, \beta_i) = \frac{1}{1 + e^{-\alpha(\theta_j - \beta_i)}} \tag{2.2}$$

where :
i is the index of the $i : th$ item
j is the index of the $j : th$ examinee
θ is the ability level
β is the difficulty parameter

α is the discriminability parameter

Compared to the Rasch logistic function, the 1PL logistic function has an additional α parameter as shown in equation 2.2. The parameter α is called the discrimination parameter; moreover, it is set to one in the Rasch model and it is completely free in the 1PL model. As it can be seen in equation 2.2, the α parameter does not have any subscript and it is constant for all items while the difficulty parameter β varies for each item i. Furthermore, the parameter α is an additional degree of freedom that can be used to improve the model fit to the data.

The discrimination parameter α defines the slope of the item characteristic curve. When α is high, the slope of the curve is sharp, resulting in a better discrimination between high-ability and low-ability examinees. However, when α is low, the curve is flat which makes the discrimination more ambiguous that is the probability of a correct answer is similar for both high-ability and low-ability examinees.

The same previously mentioned properties are still true for the 1PL model. Figure 2.2 shows the item characteristic curve for a 1PL model having parameters $\beta = 0.5$ and $\alpha = 3$.

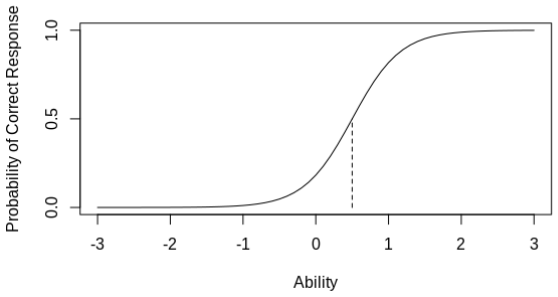

Figure 2.2: Item characteristic curve for a 1PL model

The two-parameter logistic model (2PL) was developed by Birnbaum (1968) inspired by Lord (1952) [1]. The 2PL model is a further extension of the one-parameter logistic model, where the discrimination parameter α is a free parameter. In the 2PL model, the discrimination parameter α does have a subscript as shown in equation 2.3, this means that the parameter varies for every test items. The mathematical function for the probability of a correct answer for the two-parameter logistic model (2PL) is :

$$P(u_{ij} = 1|\theta_j, \beta_i) = \frac{1}{1 + e^{-\alpha_i(\theta_j - \beta_i)}} \tag{2.3}$$

or more simply

$$P(\theta_j) = \frac{1}{1 + e^{-\alpha_i(\theta_j - \beta_i)}} \tag{2.4}$$

for $i = 1, 2, \ldots, n$

where :

i is the index of the $i : th$ item

j is the index of the $j : th$ examinee

θ is the ability level

β is the difficulty parameter

α is the discriminability parameter

The same previously mentioned properties such as local independence and invariance principles are still true for the 2PL model. Figure 2.3 shows the item characteristic curve for two 2PL models.

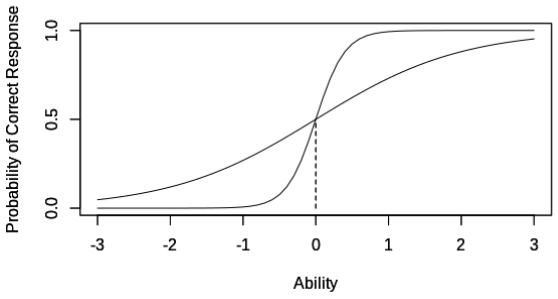

Figure 2.3: Item characteristic curve for two 2PL models ($\beta = 0, \alpha = 5$) and ($\beta = 0, \alpha = 1$)

The three-parameter logistic model (3PL) is a further refinement of the two-parameter logistic model (2PL). The difference is the fact that a new parameter c with an additional degree of freedom is introduced in the mathematical equation (2.4) of the 3PL model

All the previously mentioned models do not consider the fact that an examinee although lacking the required ability may correctly answer an given item. This drawback is considered in the three-parameter logistic model (3PL).

The three-parameter logistic model has an additional parameter c_i called "*the guessing parameter*" or more formally the *the pseudo-chance level* parameter. The pseudo-chance level parameter is the probability of a correct answer from an low-ability examinee to a given item [1].

Considering the pseudo-chance level parameter, the mathematical expression for the two-parameter logistic model in equation (2.4) is rewritten as follows:

$$P(\theta_j) = c_i + (1 - c_i)\frac{1}{1 + e^{-\alpha_i(\theta_j - \beta_i)}} \tag{2.5}$$

for $i = 1, 2, \ldots, n$

where :

i is the index of the $i : th$ item
j is the index of the $j : th$ examinee
θ is the ability level
β is the difficulty parameter
α is the discriminability parameter
c is the pseudo-chance level

Figure 2.4 shows the item characteristic curve for two 3PL models. All the parameters are constant, only the pseudo-chance level parameter varies. The pseudo-chance level moves the curve up and down, that is, it changes the lower bound of the item characteristic curve as shown in figure 2.4.

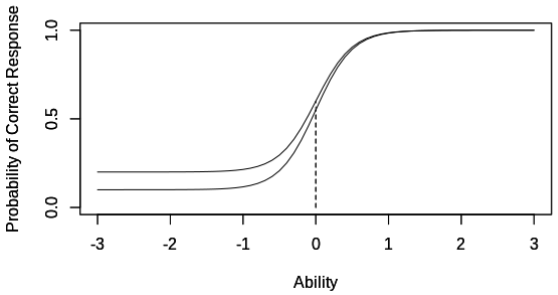

Figure 2.4: Item characteristic curve for a two 3PL models with item's parameter $\alpha = 4, \beta = 0, c = 0.1$ and $\alpha = 4, \beta = 0, c = 0.2$

2.1.1.2 Multiple group Items Response Theory

Multiple group IRT is used for analyzing test administered to multiple groups from distinct population. This enable the possibility of performing differential item functioning, that is the items' variations across groups. The differential item functioning analysis enables the detection of potential bias in the items. If an item's characteristic curve differs completely across different groups when the ability of each group is accounted then it may be concluded that the item is biased. This conclusion follows the assumption that two examinees having the same ability from potentially distinct groups should generally respond similarly to the same test item - a violation of this assumption may therefore indicates a bias [12].

Multiple group IRT is an essential model in item response theory. The theory behind Multiple group IRT has been developed inter alia by Darrell Bock and Michele F. Zimowski (1997) [13]. An extensive discussion on Multiple group IRT and its parameters' estimation can be found in [13] and [11].

Differential Item functioning (DIF) is an essential application of multiple group IRT. A complete dedicated chapter can be found in [3]. When using the item characteristic curve, two forms of DIF can be observed. The first form of DIF is defined as *uniform*, and the item characteristic curves for each groups does not have any intersection and they differ mainly in the difficulty parameters. On the other hand, the second form of DIF is defined *non-uniform* and the item characteristic curves for each groups does intersect in one point [3].

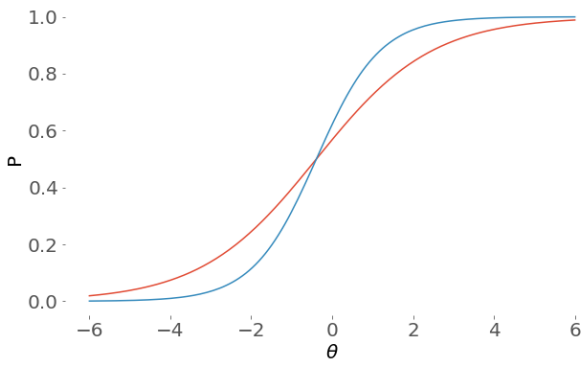

Figure 2.5: Nonuniform differential item functioning

Figures 2.6 and 2.5 shows respectively example of uniform and nonuniform differential item functioning. In nonuniform differential item functioning the

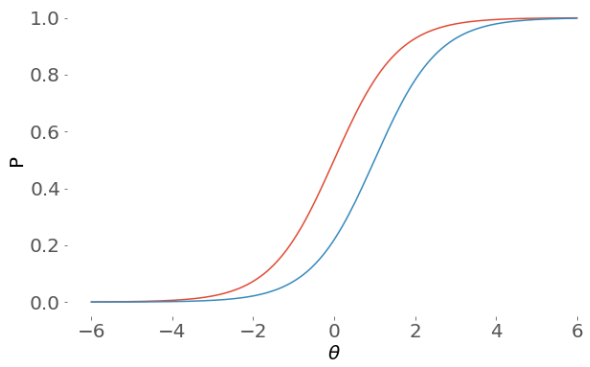

Figure 2.6: Uniform differential item functioning

test item is difficult for one group on one part of the ability level and the test item is easier on the other part. However, in the uniform differential item functioning the test item is more difficult or easier for one particular group independently on the ability levels.

2.1.1.3 Important concepts and assumptions

2.1.1.3.1 Local independence is a fundamental assumption IRT. Local independence means that a given examinee's response to a set of items is independent if the examinee's ability is constant. Thereby, the only factor influencing the response to a given item is the examinee's ability, if this factor is known and constant, the responses to each items become independent [1]. This can be expressed mathematically as follows:

$$P(I_1, I_2, \cdots, I_n|\theta) = P(I_1|\theta)P(I_2|\theta)\cdots P(I_n|\theta) \tag{2.6}$$

$$P(I_1, I_2, \cdots, I_n|\theta) = \prod_{i=1}^{N} P(I_i|\theta) \tag{2.7}$$

Where :
I_i is the $i : th$ item
θ is the ability level

2.1.1.3.2 Invariance principle states that the items parameters are independent on the sub-population on which they are estimated and that the examinees' abilities are independent on the items from which they are estimated. For a given population having members or sub-populations of high and low abilities, if the same test is administered to and estimated from the two sub-populations, the items' parameters (difficulty and discrimination) must be equal [1]. This propriety enables the comparison of different groups and different items measuring the same latent traits.

2.1.1.3.3 Negative Discrimination The basic assumption or intuition from item response theory is that the population having higher ability should respond more correctly than those having lower ability. That is on the characteristic curve, an increase of ability should result to an increase of the probability of correctly answering the given item [2]. However, some items are easier to answer for low-ability examinees and harder for high-ability examinees as show in 2.7. The general assumption in IRT is that those items must be disregarded or the conditions under which the items is correct must be redefined.

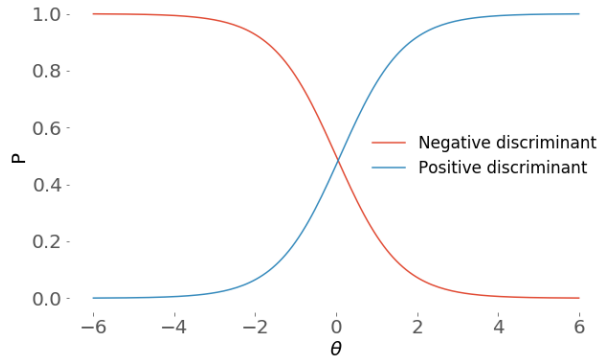

Figure 2.7: Negative and positive discriminant ICC

2.1.1.3.4 Item and test information The parameters estimation procedures need to be carried with high precision, that is with low variability. The higher the precision the lower the uncertainty is in the estimation of the parameters and the more information carried by the estimated parameters on the underlining distribution.

When administering a test to a population having a given ability, the test needs to carry a sufficient amount of information in the interval where popu-

lation's ability lies, then the populations' ability can be estimated with higher precision.

$$I_i(\theta) = \frac{[P_i'(\theta)]^2}{P_i(\theta)(1 - P_i(\theta))}, i = 1, 2, \ldots, n \tag{2.8}$$

$$I(\Theta) = \sum_{i=1}^{n} I_i(\theta) \tag{2.9}$$

where
$P_i(\theta)$: is the probability of correctness for item i for ability θ
$P_i'(\theta)$: is the derivative of $P_i(\theta)$
$I_i(\theta)$: is the information measure for the i:th test item
$I(\Theta)$: is the information measure for the all test

Equation (2.8) is the item information function and (2.9) is for the test information function.

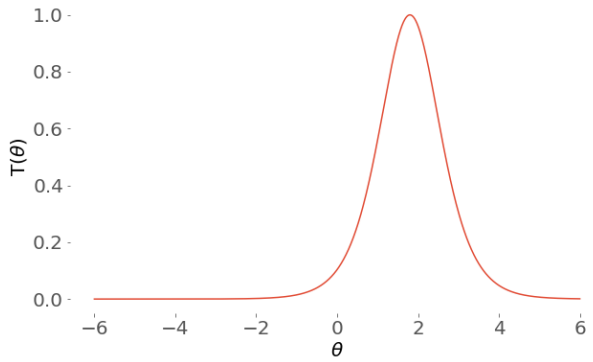

Figure 2.8: An example information curve is shown for each ability level the corresponding information level is presented

Figure 2.8 shows an example of an item information curve. It can be observed that the precision of the test is maximal around 1.8.

2.1.1.3.5 Standard error is defined as the amount of the variability in the estimation of a parameter estimated value compared to the parameter's unknown true value [2]. It can be computed mathematically as :

$$SE(\hat{\theta}) = \frac{1}{\sqrt{I(\Theta)}} \tag{2.10}$$

2.1.1.3.6 Total score is an important transformation of the item characteristic function $P(x_i = 1|\theta, \beta)$, where $x_i = 1$ represents a correct response to the $i : th$ item. In place of having the ability θ spanning on the infinite scale $\left]-\infty, \infty\right[$, instead a transformation can be applied such that for each ability θ the expected number of correctly answered items is reported. The transformation in question is defined as the *total characteristic function* [3] and can be calculated as follows :

$$T(\theta) = \sum_{i=0}^{N} P(x_i = 1|\theta, \beta_i, \alpha_i) \tag{2.11}$$

where :
N is the total number of items in the test.
θ is the ability
β_i is the difficulty parameter for item i
α_i is the discrimination parameter for item i

2.1.1.3.7 Model fit is an important part of the IRT analysis. When a IRT model is fitted to the data, it should be assessed in order to ensure that the assumptions are correct and that the model fits the observations. Different methods for doing model assessment can be found in the literature [2], [3], [1].

An important measure for the model fit is the chi-square goodness-of-fit index [2], defined as follows:

$$\chi^2 = \sum_{g=0}^{G} f_g \frac{\left[p(\theta_g) - P(\theta_g)\right]^2}{P(\theta_g)Q(\theta_g)} \tag{2.12}$$

where :
G is the total number of ability groups
θ_g is the ability for group g
f_g is the number of examinees having ability θ_g

$p(\theta_{g)}$ is the observed proportion of correct response for group g
$P(\theta_g)$ is the probability of correct response for group g
$Q(\theta_g) = 1 - P(\theta_g)$

The same definition can be found in [2].

2.1.1.3.8 Test calibration is the procedures of estimating the items parameters and the examinees' abilities [2]. The commonly methods for estimating the model parameters are Joint Maximum Likelihood and the Marginal Maximum Likelihood. Expectation maximisation and the Newton-Raphson algorithm are important methods used for parameters estimation especially when there are two steps : items parameters estimations and examinees' abilities estimation.

In the book [11], an exhaustive discussion is dedicated to the different parameter estimation techniques and models. It can be found in [11], all the details about different estimation procedures, algorithms and the mathematics behind major item response theory models.

The joint maximum-likelihood estimation function for both items and examinees' responses can be expressed mathematically as follows :

$$P(U|\theta) = \prod_{j=1}^{N} \prod_{i=1}^{n} P_i(\theta_j) Q_i(\theta_j) \tag{2.13}$$

where :
N is number of examinees,
n is the length of the test,
$P(U|\theta)$ is a probability matrix of the examinees' responses u_{ij} for each ability level θ. The matrix is of dimension $n \times N$,
$P_i(\theta_j)$: is the probability of correctness for item i for ability θ_j
$Q_i(\theta_j) = 1 - P_i(\theta_j)$

In order to jointly estimate the examinees' ability and the test items' parameters, the logarithmic of the joint likelihood equation (2.13) is taken and the derivatives are calculated such as each parameters can be updated using the Newton-Raphson algorithm. For details about different estimation and optimization methods, see [11].

Given the fact that this thesis does not focus on the particular implementation of parameters estimation techniques for item response theory, the estimation process is based on widely used packages and libraries in the R programming language . The particular package used in this theses are the ltm R

package for item response theory analysis. The ltm R package [1] is widely used in may IRT research papers and books such as [2] [14] [5] [12].

2.1.2 Machine learning

Machine learning is a multi-disciplinary discipline within computer science in general and particularly within artificial intelligence [15]. Machine learning employs methods from different fields such as algorithmic, information theory, computational statistics, neuro-science. It is applied for solving problems in domain such as speaker recognition, person image recognition, dialog systems, automatic vehicle navigation where it is difficult or impossible to develop conventional algorithms that assumes a clear problem definition and a procedural step-by-step solution. Machine learning is used for making inferences and drawing conclusions from the data, past observations and experiences.

Machine learning attempts to mimic the human ability of learning from experience, therefore the data is a crucial element in modern machine learning [15]. Data is used for training systems to learn a task for instance learning the relation between an given image I and an name describing the content of the image such as $tree$ - the ultimate goal of the system is to learn the function $f(I) = tree$. Once the system is trained and has learned the function $f(I) = tree$, the next step is to test the trained system's ability to recognize different instance of images depicting the same objects $tree$ under different circumstances or environment not present in the training data. The ultimate goal of testing is to test the robustness of the trained systems and check if the system really has learned a trait common to all trees summarized in the function $f(I) = tree$.

Different methods are used for learning the function $f(x) = C_i$ relating an element of interest x and a class C_i. If the data is given in such way that each element of interest x has a class C_i annotated, then the machine learning methods employed is the *supervised learning*. The supervised learning learning methods assume that the data is annotated or labelled. However, annotating or labeling the data is expensive and tedious, therefore new methods such as the *unsupervised learning* are developed to learn from unlabeled data. With unlabeled data, the system has only access to x and not C_i, the classes. The ultimate goal of unsupervised learning is to automatically find the underlying function $f(x) = C_i$ without having access to any labeled data.

Apart from supervised and unsupervised learning, there are other methods

[1] The ltm R packge for IRT can be found at : `https://cran.r-project.org/web/packages/ltm/index.html`

and heuristics for learning from the data, such as semi-supervised learning, genetics and evolutionary learning, deep learning, reinforcement learning, deep reinforcement learning, to name a few.

2.1.2.1 Supervised learning

As aforementioned, supervised learning assumes labeled data. Classification algorithms are used for learning from labeled data. There are in fact many classification algorithms and a complete discussion about all of them is out of the scope of this thesis. However, major classifications algorithms are super vector machine, decision trees, naive Bayes classifier, random forest, artificial neural networks, k-nearest neighbours, Gaussian process classifier, Adaboost classifiers and logistic regression classifiers. Further reading about different classifiers can be found in [16] [17] [18] [19].

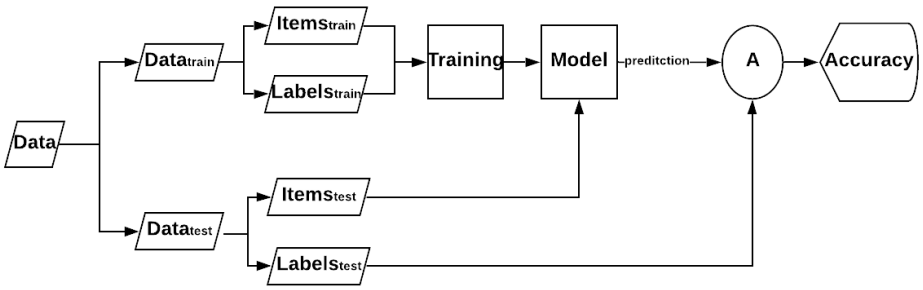

Figure 2.9: Machine learning training and testing

Figure 2.9 shows the general procedure for training and testing machine learning models in supervised learning. First the data is divided into two sets, the training and testing dataset respectively. The training data is used for training and building the model; on the other hand, the testing data without annotations is used for testing the model's predictive power. The model predicts the labels of the test data. Afterwards, the predicted labels are compared against the true labels. The number of correctly labelled items are computed and an accuracy measure is obtained. The accuracy is defined as follows:

$$Accuracy(classifier) = \frac{\sum_{i=1}^{N} \mathbb{I}\big[C_i = classifier(x_i)\big]}{N} \tag{2.14}$$

where :
N is the total number of test items
$\mathbb{I}$ is the indicator function

x_i is the $i : th$ test item
C_i is the true label of of test item x_i

Two important stages among other things in the training process are omitted in figure 2.9, namely the feature extractions and the evaluation procedures. The data is usually presented in such a way that it is impossible to build a model directly from it. A sound file or an image are represented in a way not enabling a direct model construction. Therefore, a transformation is applied in order to transform the raw data, for instance a sound file, to a set of numerical vectors representing a set of particular measurable properties of the file in question. Those transformations are called feature extractions and are performed after data acquisition.

Moreover, before testing the model, an optional evaluation can be performed in order to assess the model and improve it, before testing and an eventual deployment. So the general procedure in machine learning is to divide the data into three distinct part : training, testing and validation data. The evaluation data are used for hyper-parameters tuning, that is finding a set of hyper-parameters that best suits the model and improve its accuracy. Furthermore, the evaluation can also be used in order to prevent that the model completely "copies" the data without building a latent representation of the property the systems is intended to learn, this is called overfitting [17]. Overfitting decreases the performance of the model such as its accuracy. This due to the fact that the model can not generalise and correctly classify items not present in the training data or items different from the training data.

A machine learning model can either be built for a binary-class or a multi-class problem. Many classification algorithms can be adjusted for multi-class problems. In order to analyse how the accuracy varies across different class, a *confusion matrix* can be computed. A confusion matrix shows for each true class how the predicted class is distributed among other classes. This enables the possibility to assess which classes the system is commonly confusing.

2.2 Application of IRT in machine learning

When studying the literature on the application of item response theory in machine learning, three main trends and applications can be observed. The first applications of IRT in machine learning, are the use of IRT for evaluating and comparing models based on their abilities and analysing items based on the difficulty and discriminability parameters [6] [5] [20]; that is, the usage of IRT as an enhanced analytical tool in machine learning. However, the second application is more related with the application of IRT for enhancing machine learning models, by combining IRT and ML models in an unified model. Finally, the third application is more related to the application of IRT in order to solve specific problems in machine learning such as the cold-start problem, dataset for bench-marking.

As previously mentioned, the second applications are more concerned with the enhancement of existing machine learning methods by using IRT. IRT and machine learning models are combined in order to create a hybrid model having the strength of both machine learning and item response theory [21] [22] [23]. One of the aims of the hybrid model is the achievement of explainability and interpretability of decision rules in machine learning. Explainability and interpretability are critical concerns in many machine learning models where black-box heuristics such as deep learning and artificial neural networks are used. In [21], the authors discuss the construction of a theoretical hybrid model called *supervised item response theory*, as an aid for explainable decision making. However, few indications are given regarding the generalisations of the proposed models and the accuracy achieved by making the models more explainable. On the other hand, paper [23] is about integrating IRT in machine learning for ensemble learning, where the weight of each classifiers are computed based on their respective abilities (to classify harder items). The third paper [22] is about making deep knowledge tracing explainable by using IRT.

The third application are more closely related to the first aforementioned application. In this case, IRT is used to solve a specific problem in machine learning and to build benchmark models for comparing the machine learning models' performance and the human performance [24] [25] [9]. Item response theory is used for bench-marking purposes and particularly for problem solving. Paper [24] is about the integration of machine learning for solving the problem of cold-start in adaptive learning systems. The cold-start problem is the lack of prior information regarding an new user in an adaptive learning system. Machine learning systems are used to predict the user's ability and response patterns. The hypothesis is that the user experience will be enhanced by

predicting those missing information. However, the system is assessed based only on its predictive accuracy on the data and it is not further tested with *real users*, for who the system is intended. Paper [25] [9] is about the application of IRT for natural language processing problems such as textual entailment and sentiment analysis.

This thesis is more related to the first application of item response theory in machine learning. Three important papers have been written on the general application of IRT in machine learning as an analytical tool for model assessment and instance analysis [6] [4] [5]. The focus of the papers [6] [4] [5] is more on model evaluation and classifiers analysis than on the particular items or test data. However, this thesis will be focused more on the analysis of the test items, in particular the music data from the GTZAN dataset [10]. Moreover, there are few studies focusing on the application of item response theory in music data analysis and music information retrieval. The aim of this study is therefore to investigate the extent to which item response theory can be applied to music genre classification and music data analysis. The experiment is conducted using the GTZAN dataset [10].

Chapter 3

Methods

3.1 Machine learning

In order to answer the research question regarding the extent to which item response theory can be applied in machine learning in general, and music data analysis in particular, the GTZAN [10] data was chosen for performing the experiments. The GTZAN is an widely used dataset in music genre classification and music information retrieval [26]. The data consist of 1000 audio excerpts with a duration of 30 seconds sampled from 10 different musical genres : blues, classical, country, disco, hiphop, jazz, metal, pop, reggae, rock. There are 100 audio excerpts in each genre. An exhaustive analysis of the GTZAN data can be found in [27].

Figure 3.1 shows the data distribution for all the genres. The figure is obtained by projecting the feature matrix without the labels on a 2D dimensional space using the PCA algorithm for dimensionality reduction and the K-means clustering algorithm for cluster detection. As can be observed in figure 3.1, the spread of the classical songs is much broader and contains many outliers. The spread of classical songs might increase the ability to discriminate classical audio excerpts from other genres. However, many data items from some classes such as rock, disco, country do not form a clear separate cluster, thus increasing the difficulty to correctly classify data items from those classes.

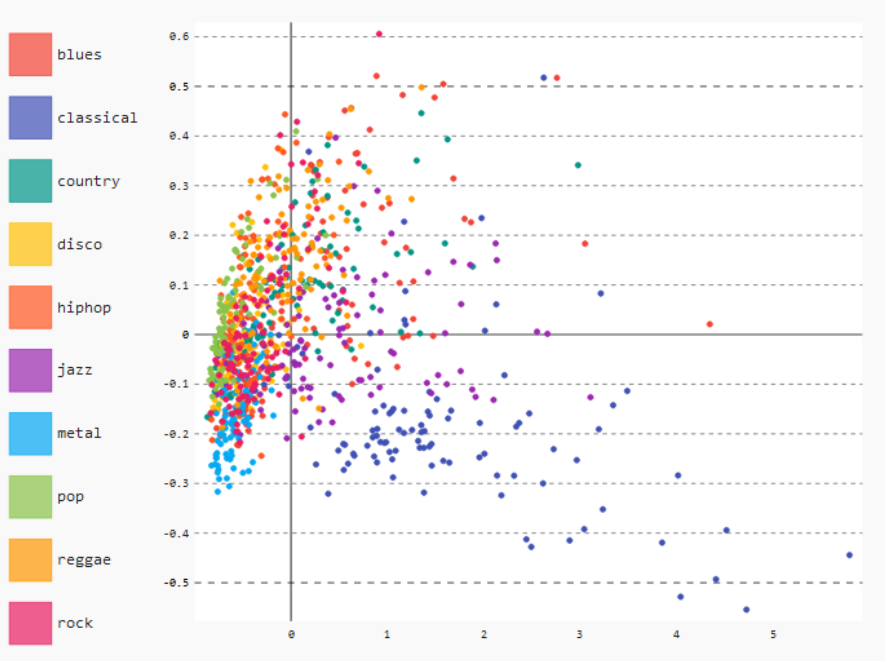

Figure 3.1: Data distribution in 2D using PCA

In paper [28], the authors have shown that an artist filter needs to be applied in music genre classification; otherwise an false conclusion can be drawn regarding performance of the models. Many models perform specifically well when the songs of the same artist occur both in the train and test data, which transforms the classification problem from genres classification to artists identification. The same problem can be observed in the GTZAN data [27], where for instance the songs in the Reggae category are predominantly songs from Bob Marley. The songs in the GTZAN data are heavily imbalanced with respect to the diversity of the artists. Therefore, this thesis investigates the influence of artist filters both using the accuracy metric and item response theory as an analytical tool.

The investigation is performed by training 53 different models [1] or classifiers using randomly stratified data and filtered data which in total gives 106 classifiers. The data division index for chosen which items to insert in the randomly stratified dataset or in the filtered dataset is from [14] [2]. Randomly stratified data are the data where the artist filter is not applied - the audio excerpts from the same artist may be present in both the train and test data. On the other hand, the filtered data are the data where the artist filter is applied such that no audio excerpts from the same artist appear in both the train and test data. The final test data is the intersection of both the filtered test data and the stratified test data. This procedure ensures that the test data is the same and constant for both models trained with filtered and stratified data. Figure 3.2 shows the procedure more graphically.

As previously mentioned, the test is the intersection of the filtered test data and the stratified test data. The final test data consist of 74 test items and the train data consist totally of 640 items. However, no evaluation data are used, because the aim of the study is not create the best model but to have a variety and diversity of models for analysing different models and items. Figures 3.3 and 3.4 show the distribution for each genres in the filtered and stratified train and test data. As can be observed all the genres are represented in the test data, even though there is an imbalance between the different genres.

[1] The complete list for all the models or classifiers used in this thesis can be found in the appendix A.2

[2] The index files can be found in thís repository: `https://code.soundsoftware.ac.uk/projects/confint/repository`

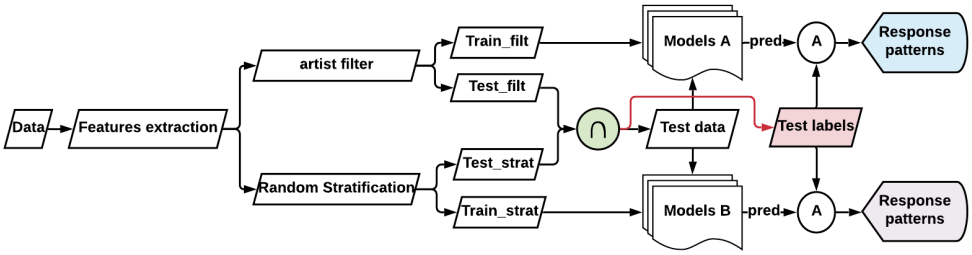

Figure 3.2: Models A represents all the 53 classifiers trained on filtered data. On the other hand, Models B represents all the 53 classifiers trained on stratified data. The response patterns are all the responses of the models to the test data, where 1 represents a correct prediction and 0 an incorrect prediction.

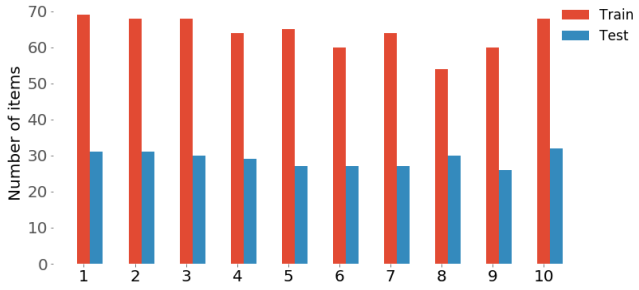

Figure 3.3: Filtered train and test data

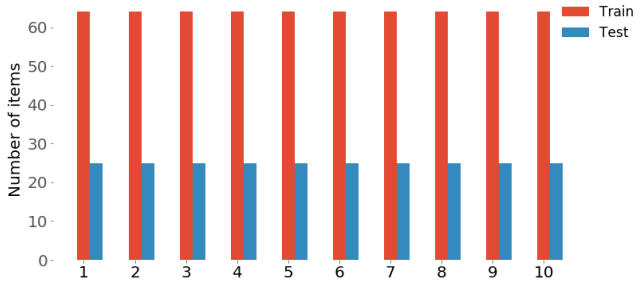

Figure 3.4: stratified train and test data

The figures show the proportions of the number items for filtered and stratified test and train data. On the y-axis, the number of items is presented while on the x-axis the number $\{1, 2, \ldots, 10\}$, which is numerical encoding of the 10 genres { blues, classical, country, disco, hiphop, jazz, metal, pop, reggae, rock }.

Figure 3.5 shows the genres distribution of the final test data which is an intersection of the filtered test data and the stratified test data. See appendix table A.1 for an exhaustive list of all the test items.

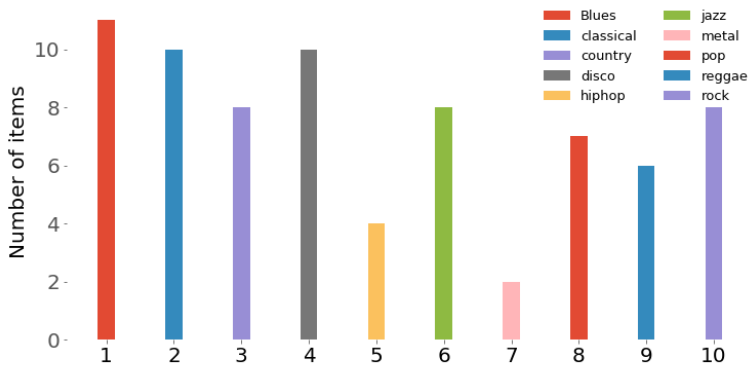

Figure 3.5: Test data distribution. On the y-axis, the number of items is presented while on the x-axis the number $\{1, 2, \ldots, 10\}$, which is numerical encoding of the 10 genres $\{$ blues, classical, country, disco, hiphop, jazz, metal, pop, reggae, rock $\}$.

As aforementioned before the data can be used in classification algorithms, the audio data need to be transformed in numerical format such as the classification algorithms can be applied. The numerical format should reflect the specific characteristic of the audio data, this procedure is called features extraction. The features extraction is an important part of music analysis and classification [29]. A song $\mathcal{S}$ from genre $\mathcal{G}_{blues}$ has extracted features $\mathcal{X}$ such as $\mathcal{F}(\mathcal{S}) = \mathcal{X}$, where $\mathcal{F}(\mathcal{S})$ is the features extraction function transforming the song from $\mathcal{S}$ to numerical format $\mathcal{X}$ and $\mathcal{G}_i$ is the genre i of a song $\mathcal{S}$.

The ultimate aim is that two songs $\mathcal{S}_i \in \mathcal{G}_i$ and $\mathcal{S}_j \in \mathcal{G}_j$, where $\mathcal{G}_i$ and $\mathcal{G}_j$ are from genres i and j such as $i, j \in \{$ blues, classical, country, disco, hiphop, jazz, metal, pop, reggae, rock $\}$. Given the songs have extracted features $\mathcal{F}(\mathcal{S}_i) = \mathcal{X}_i$ and $\mathcal{F}(\mathcal{S}_j) = \mathcal{X}_j$ respectively, they should be significantly different such as their features $\mathcal{F}(\mathcal{S}_i) = \mathcal{X}_i$ and $\mathcal{F}(\mathcal{S}_j) = \mathcal{X}_j$ are sufficient to discriminate them and to differentiate from each other.

The extracted features for the GTZAN data in this thesis, are the conventional commonly used features in music and speech analysis [29]. The features consist of extracted statistics such as the mean and the standard deviation of the zero crossing rate, spectral centroid, spectral bandwidth, rolloff, spectral contrast and the Mfccs of each audio items in the GTZAN data.

Usually in sound and music analysis, the raw features $\mathcal{X}$ are not directly used in the classification algorithms. The so-called short-time processing is used to change the features into different overlapping frames. The analysis is then done on the frame level since analyzing the sound as a whole increases the difficulty to find a trend as the sound may vary quickly. For instance, in a sound file different words may be spoken on different frames, different melodies may be played. Therefore, the frames are overlapping in order to ensure that the no words or sounds are missed. For instance the spoken word "institution", the first phonemes of the word may appear in the first frame and the last phonemes may be present in the last frame, in order to avoid such problems the frames need to be overlapping. In this experiment, the window length (W_L) is 2048 and the window size or hop length (W_S) is 512, reaching a overlapping value of 0.75 as specified in Librosa [3] [30]

Regarding the MFCC feature, only the first 13 coefficients are retained as they are the most significant ones. However, it should be noted that there is theoretically an infinite number of statistics and features which could possibly been used for audio features extraction, thereby a choice sometimes arbitrary needs to be taken. The statistics and features used are therefore commonly used features in the field of music audio analysis [29].

After the features are extracted, the classification algorithms algorithms are used for building two models. One trained with stratified data and the other with filtered data. The classification algorithms are support vector machine, decision trees, naive Bayes classifier, random forest, artificial neural networks, k-nearest neighbours, Gaussian process classifier, Adaboost classifiers and logistic regression classifiers. Those are widely used classifications algorithms for multi-class and binary class problems [16] [17] [18] [19]. A complete list of all the classifiers used and their accuracy on respectively filtered data and stratified data can be found in table A.2.

Strictly speaking, the music genre classification applied to the GTZAN is a multi-class problem, where the aim is to predict the class or genres C_i of a song $\mathcal{S}$ such as $i \in \{$ blues, classical, country, disco, hiphop, jazz, metal, pop, reggae, rock $\}$. The aforementioned classifiers are used to train and learn from the data and the 74 test items are administered to the classifiers for model assessment. 106 different classifiers with different hyper-parameters [4] are con-

[3] Librosa is a widely used python package for audio and music signal processing. It provides a diversity of important functions commonly used in music information retrieval `https://librosa.github.io/librosa/`

[4] The hyper-parameters are chosen by using the evaluation data for some models, but in general the hyper-parameters are arbitrary in order to have a variety of classifiers with different

structed, where half (53) of the classifiers are trained with filtered data and the other half are trained with stratified data. The 74 test items are administered to the 106 classifiers and the performance are recorder in matrix A of size 53×2, where each columns are represent the accuracy measures for classifiers trained respectively with filtered and stratified data as shown in figure 3.2.

Other than the accuracy matrix for each models, the response matrix or score matrix $\mathcal{I}$ are also computed when the test items are administered to each classifiers. The response matrix $\mathcal{I}$ is binary matrix of dimension 106×74. The element $\mathcal{I}_{ij}$ of matrix $\mathcal{I}$ is 1 if classifier i correctly classify item j and 0 otherwise. This matrix is later used for item response analysis.

For multiple group item response theory analysis, the aforementioned $\mathcal{I}$ is slightly modified for counting for the fact that the first 53 rows are for $group_1$ which is classifiers trained with filtered data and the last 53 rows are for $group_2$ which is classifiers trained with stratified data.

3.2 Item response theory

Item response theory is applied to the response matrices previously obtained in the machine learning stage. The principal assumption for the item response theory application is the fact that the 106 classifiers are considered in the IRT sense as the examinees and the 76 test items are interpreted as the test or the test instrument. It should be noted that there is a limitation because the numbers of classifiers are not significantly high compared to the length of the test instrument in order to estimate all the parameters reliably. However, the test construction and the the number of classifiers are sufficient for giving critical insights in order to respond to the research questions.

Three different analysis are performed using the three response matrices computed in the machine learning stage. The first analysis is plain IRT analysis of the response matrix where the responses are dichotomized either correct prediction (1) or incorrect prediction (0). The second applications are the multi-group analysis of test items. Finally, the last application is the analysis of the response matrix through the polytomous item response theory.

The guiding hypothesis is that some items in the test data may have a negative discriminant. Those items are scrutinized and thoroughly analyzed in order to observe the possible reasons for the negative discriminant.

---

abilities. The codes will be published at `https://github.com/benedictmulongo/IRT_FOR_MACHINE_LEARNING`

3.3 Audio manipulation

Compared to other data formats such as text data, numerical data such as the audio and image data are more manipulable. The audio and image can be slight modified without altering the true content of the data. For instance, with an effect such as fading, amplitude modulation can be applied to audio data without modifying the content of the data. A blues song will still sound as a blues after such manipulation. The same is true for an image data that is rotated or slightly blurred, the content of the image will remain the same. Such techniques are widely used in deep learning for data augmentation in order to create models robust to changes and capable of recognizing the same content under different forms and shapes.

Pitch shifting is an audio manipulation capable of changing the pitch of the singer or artist without necessarily affecting the melody. This is particularly important for analyzing the effects a small change in the properties of the audio data may have to parameters such as discriminability and difficulty.

A pitch shifting effect can be implemented by a low frequency oscillator (LFO) and a delay buffer. A low frequency oscillator is low frequency (less than $20Hz$) wave signal such as sine, square, triangle wave and a delay buffer is a buffer or a memory array used to implement delay in audio manipulation. A delay block of with memory capacity of d samples, takes input sample $x[n]$ and return the output sample $x[n-d]$ delayed.

Figure 3.6 shows the block diagram for an example pitch shifting method. The input signal x is put into the blocks and the output is either $y[n] = (1-f)x[n] + fx[1]$ or $y[n] = (1-f)x[n-d] + fx[n-d+1]$ depending of the variable d with is dependent on the output of the LFO function. A complete discussion of different audio effects and their corresponding implementation in Matlab can be found in [31]. The method used to picth shift the audio data is based on [31].

The modification of the audio data is performed to items with a negative discriminant in order to investigate how this modification may affect the sensitivity of the parameters of those items in terms of the discriminability and the difficulty.

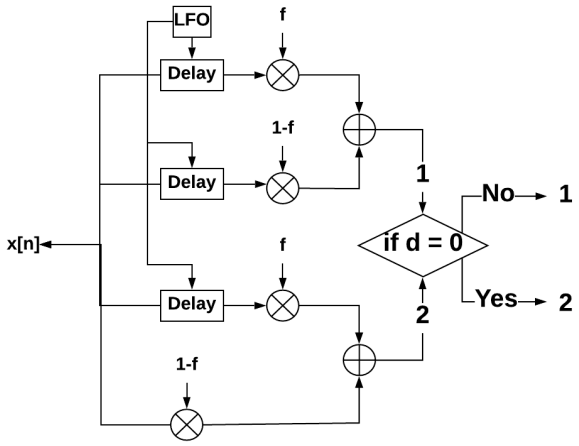

Figure 3.6: Block diagram for pitch shifting where a delay buffer named Delay is controlled by a LFO, x is the input signal. Moreover, the variable f and d is also a function of the LFO signal

3.4 Data analysis

The machine learning models are analysed using the different IRT models such 1PL, 2PL, 3PL, polytomous, multi-group. The items are analysed by their item characteristic curves, the classifiers by their models parameters. Furthermore, the standard error and the item fitness for the parameters are analysed for evaluated the models and comparing different models. Those metrics are used because they are standards procedure in item response theory analysis [1] [3] [2].

3.4.1 Analysis of variance (ANOVA)

ANOVA is a statistical method used for checking the hypothesis that there is no differences in the treatments effects in an experimental study. In an experiment, it is possible to check the effect of a treatment by comparing the difference between a group where the treatment is applied and another group where the treatment is not applied - this called a one-factor ANOVA or ONE-way ANOVA. On the other hand, if the comparison is not just based on the treatment but also on the gender, then the analysis is called a two-factor ANOVA [32].

$$ss^2 = \frac{1}{DF}\sum(y_i - \hat{y})^2 = \frac{SS}{DF} \tag{3.1}$$

where :
DF is the degree of freedom
SS is the sum of squares
$\hat{y}$ is mean

The ANOVA is computed by first computing the sum of squares between treatments (SS_{Error}) and within treatments ($SS_{Treatment}$). Furthermore, the degree of freedom is also computed between treatments (DF_{Error}) and within treatments ($DF_{Treatment}$). Then, the total sum for the sum of squares and degree of freedom is calculated as respectively as $SS_{Total} = SS_{Error} + SS_{Treatment}$ and $DF_{Total} = DF_{Error} + DF_{Treatment}$.

Moreover, the mean square treatment (MST) is computed as the sum of squares divided by the degree of freedom. This gives the possibility of obtaining the F-test or the variance of ratio test as $F = \frac{MST_{Treatment}}{MST_{Error}}$. Comparing the F-test and the value obtained from the F-table, it can be concluded that the null hypothesis is rejected if F-test value is greater than the value at the F-table.

A complete discussion about the ANOVA and the algorithmic procedure for obtaining the ANOVA can be found at [32]. However, given the fact it is tedious to calculate the ANOVA by hand, in this thesis the ANOVA is computed using the python statistical package called *statsmodels* [5].

[5] https://www.statsmodels.org/stable/index.html

Chapter 4

Results

4.1 Machine learning

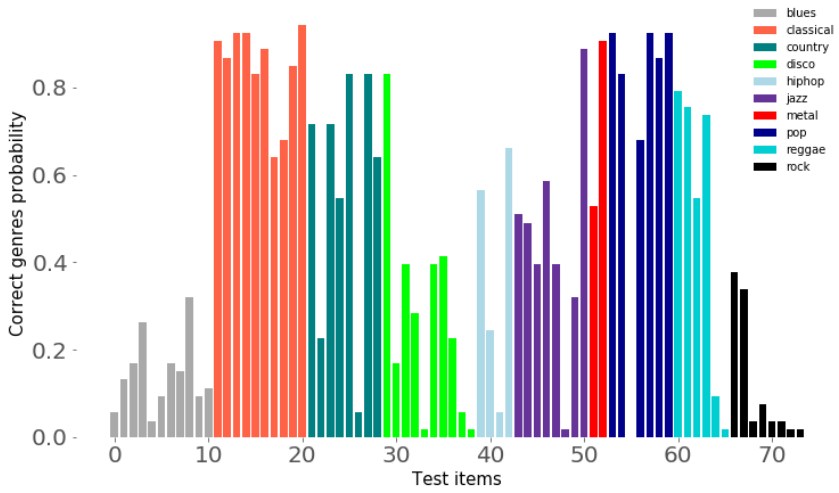

Figure 4.1: Test frequency of correct response by genres for filtered data

As aforementioned the number of test items is in total 74. On the y-axis of figures 4.2 and 4.1, the frequency of correct responses in percentage for each items is shown, on the other hand, on the x-axis the test items is represented by a color corresponding to the genre class the test item belongs to.

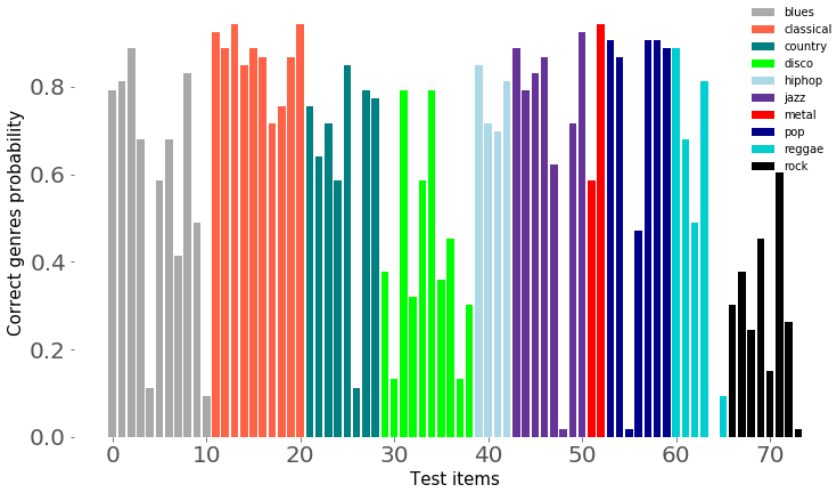

Figure 4.2: Test frequency of correct response by genres for stratified data

Figures 4.2 and 4.1 show the frequencies of correct responses for each test items by genres for classifiers trained with filtered and stratified data. It can be observed that the test items in the classical items are much easier to classify.

Figure 4.3 shows the accuracy of all the classifiers for classifiers trained with filtered and stratified data. As can be seen the classifiers trained with stratified data has generally a higher accuracy than those trained with filtered data.

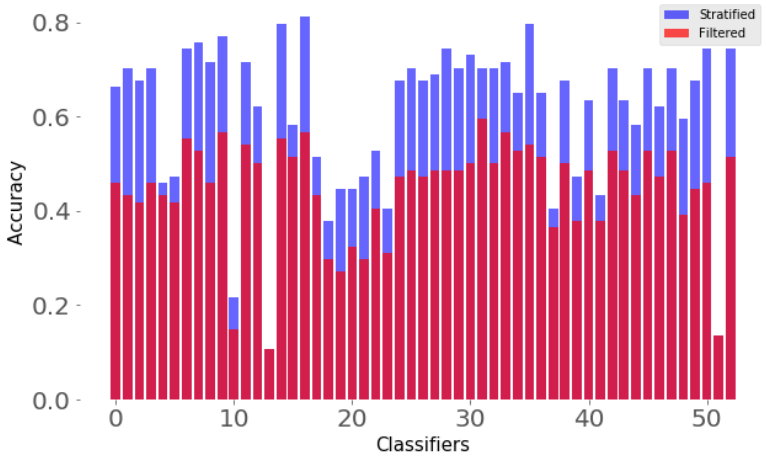

Figure 4.3: The figure shows all the classifiers' accuracy. On the y-axis is shown the accuracy in percentage while on the x-axis all the 53 classifiers' are shown for on both the filtered and stratified data. This figure shows the proportion of accuracy for the same models both for stratified and filtered data.

4.2 Items response theory

4.2.1 Models comparison

Figure 4.4 shows the correlation for the discriminability parameters estimated in different IRT models. The discriminabitity is the measure of how well the high-ability examinees can be distinguished from the low-ability examinees. Figure 4.4 suggests that the discriminability parameters estimated from different IRT models are not correlated.

Figure 4.5 shows the correlation for the difficulty parameters in 4 distinct IRT models. It can be observed that there is no clear correlation between the discriminability and difficulty parameters for the different models. It may be concluded that there are no observable correlation between the different IRT models.

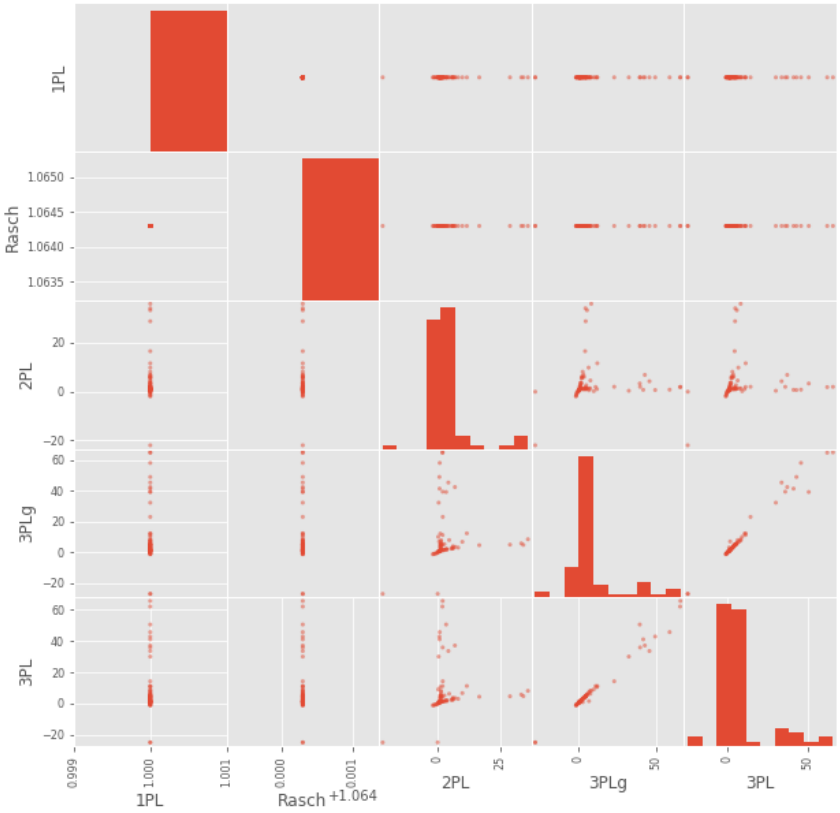

Figure 4.4: Correlation for discriminability parameters

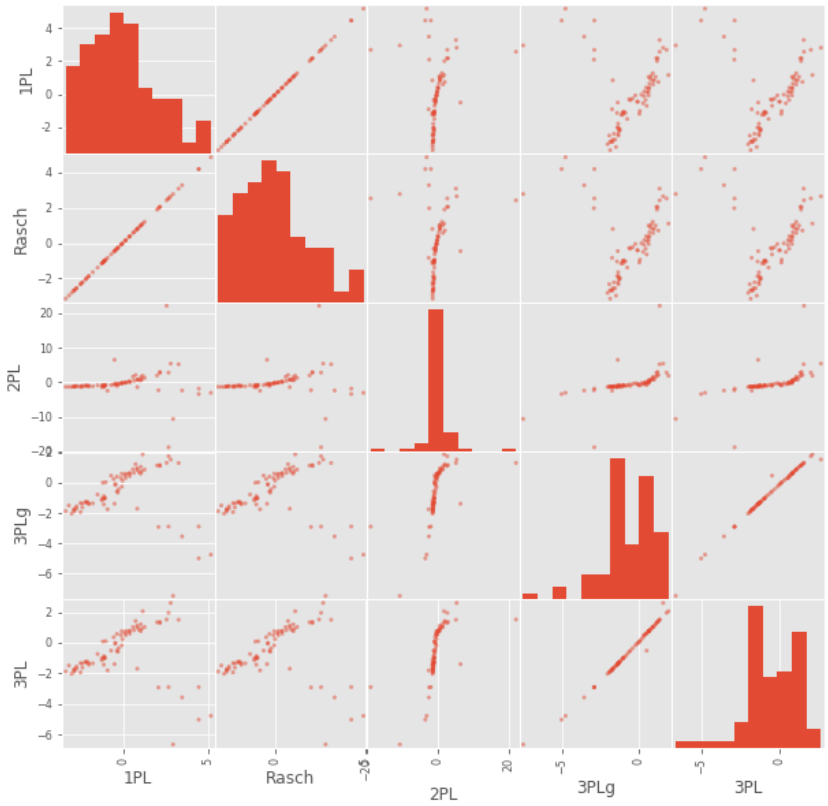

Figure 4.5: Correlation for difficulty parameters

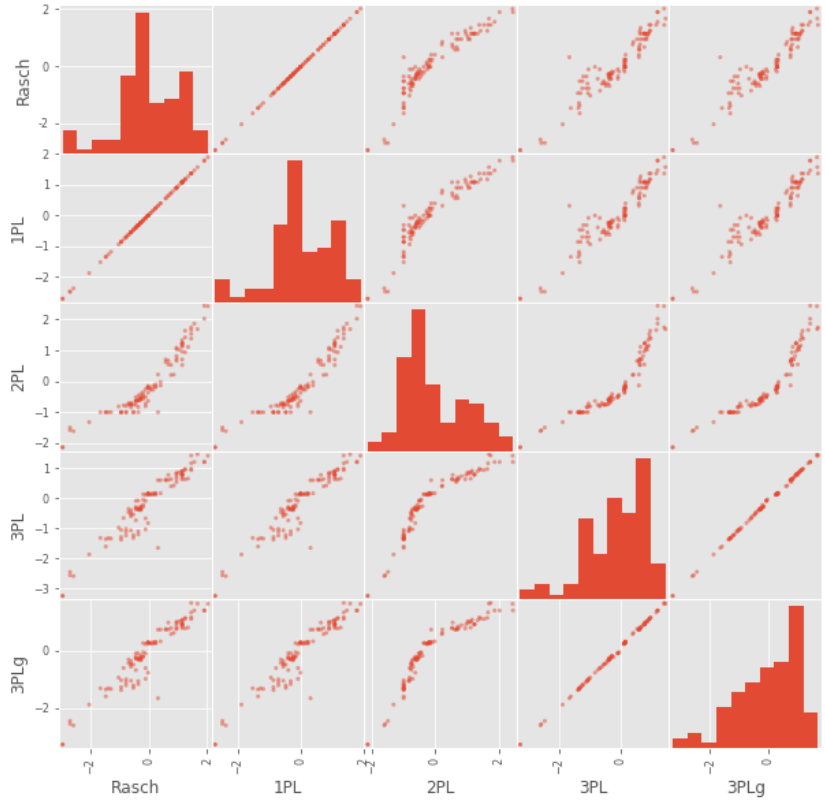

Figure 4.6: Correlation for ability parameters

Figure 4.6 shows the correlation for the estimated ability parameters in different IRT models. The estimated classifiers' ability seem to be correlated in different models. The estimation are therefore consistent throughout all models.

Table 4.1: One-way ANOVA

	df	sum_{sq}	$mean_{sq}$	F	$PR(>F)$
Data group	1.0	32.2	32.2	46.7	5.7e-10
Residual	104.0	71.7	0.69	NaN	NaN

Table 4.1 shows the one-way ANOVA analysis for two groups of classifiers, classifiers trained with filtered data respective stratified data. In table 4.1, df represents the degree of freedom, sum_{sq} the sum of square, $mean_{sq}$ the mean of square, F the variance of ratio test and $PR(>F)$ is the p-value. Furthermore, Data group is the treatment that is the presence or the absence of filtering and residual is the error, that is each value in the columns are computed between groups and within groups. It can be observed that there is in fact a significantly difference between classifiers trained with filtered data and stratified data, this difference may not be caused by simple randomness.

Table 4.2: Two-way ANOVA

	df	sum_{sq}	$mean_{sq}$	F	$PR(>F)$
Data group	1.0	28.4	28.4	59.4	2.99e-11
Classifier	7.0	18.8	2.7	5.6	2.79e-05
Data group:Classifier	7.0	6.5	0.93	1.95	7.2e-02
Residual	80.0	38.3	0.48	NaN	NaN

The aforementioned description for table 4.1 still holds for table 4.2. In table 4.2, a two-way ANOVA analysis is performed there not only the data group is considered but also the classification algorithm such as KNN, SVC, MLP, decision tree. From the figure, it can be observed that there is also a significantly difference between classifiers trained with filtered data and stratified data as previously observed in table 4.1. Moreover, the classification algorithms seem also to be an important factor in the ability differences between different models.

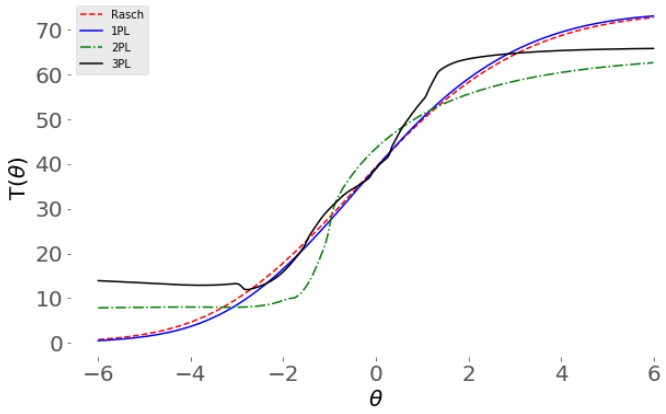

Figure 4.7: Total scores for IRT models

Figure 4.7 shows the total scores for different IRT models. The total score is the expected number of correctly answered or classified test items in a given model for different ability levels. It can be observed that the expected number are highest for higher ability level and lowest for lower ability level in the Rasch and 1PL model. The 2PL appears to be trade-off between the simplest models and the more complex models.

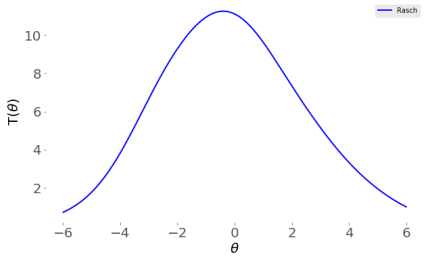

Figure 4.8: Information curve for Rasch

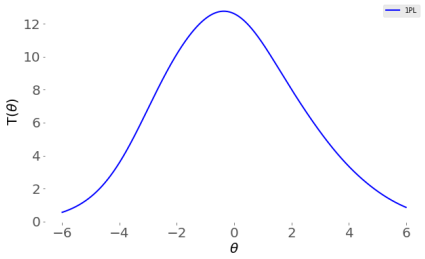

Figure 4.9: Information curve for 1PL

Figure 4.8, 4.9, 4.10 and 4.11 show the information curves for different IRT models. The Rasch and 1PL are the simplest models and have a smooth information curve. The test estimation is precise for classifiers with moderate abilities. The 2PL and 3PL have a more complex information curves, however the 3PL has a even more complex form and no clear maximimun can be spotted. Furthermore, there is evidence from the above figures that the 2PL model better explains the data than any other models as can be seen in table C.1.

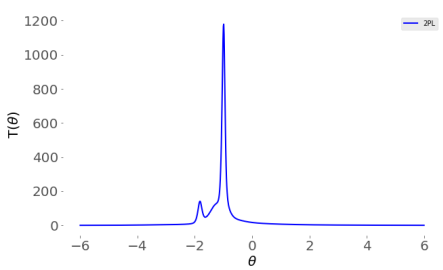

Figure 4.10: Information curve for 2PL

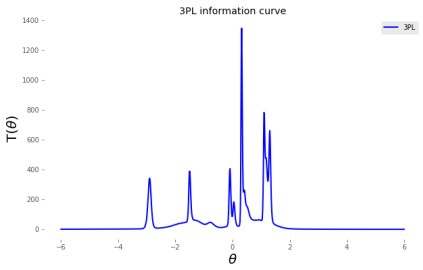

Figure 4.11: Information curve for 3PL

Figure 4.12, 4.13, 4.14 and 4.15 show the error curves for different models and they are the other side of the information curves. It can be seen that the test is generally precise for moderate classifiers.

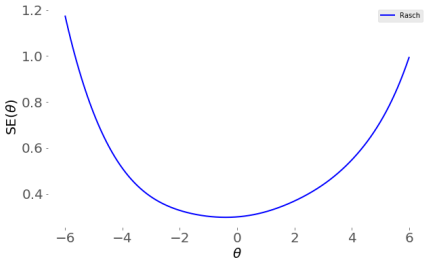

Figure 4.12: Error curve for Rasch

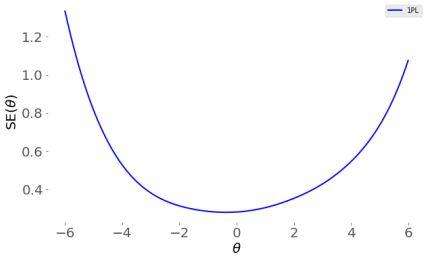

Figure 4.13: Error curve for 1PL

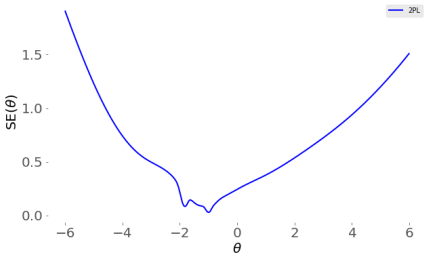

Figure 4.14: Error curve for 2PL

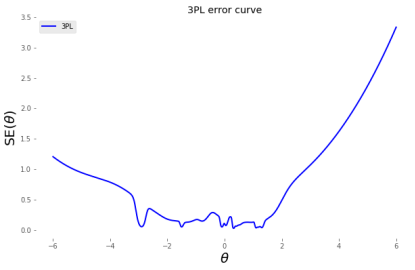

Figure 4.15: Error curve for 3PL

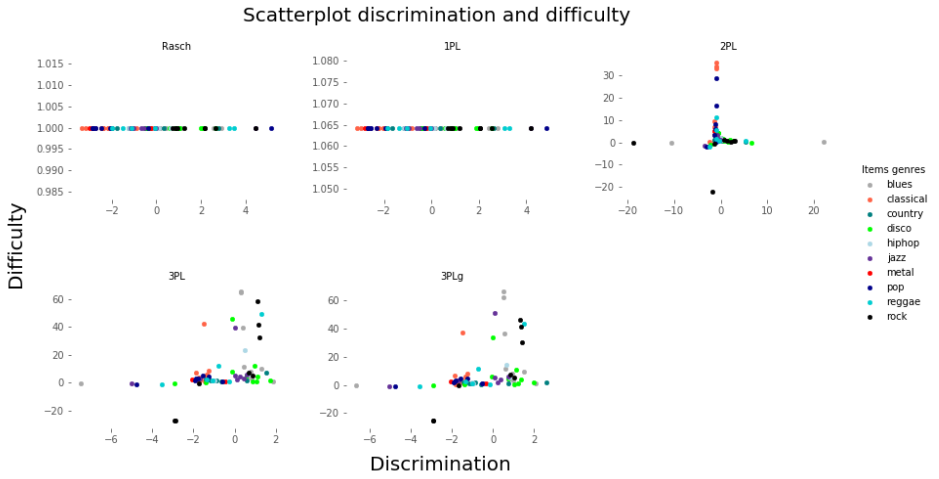

Figure 4.16: Correlation for discriminability parameters

Figure 4.16 shows the correlation of the difficulty and discriminability parameters. As it can be observed there is no clear correlation between the two parameters. It must be said; however, that for the 2PL models, most of the test items have a discriminant parameters around zero and a higher difficulty parameter.

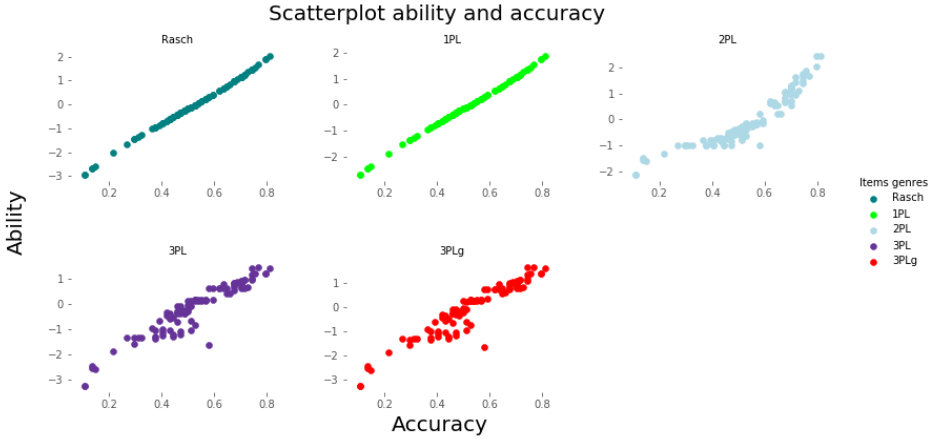

Figure 4.17: Correlation ability vs accuracy

Figure 4.17 shows the correlation of the ability and accuracy for all classifiers. There is a clear correlation between accuracy and ability. It can be observed that when the model fits the data and all the test items have a positive discriminant, then the correlation between the accuracy and the ability

measures is evident. Therefore, in this case a higher accuracy implies a higher ability and vice versa [4].

4.2.2 2PL

4.2.2.1 Items characteristic curves

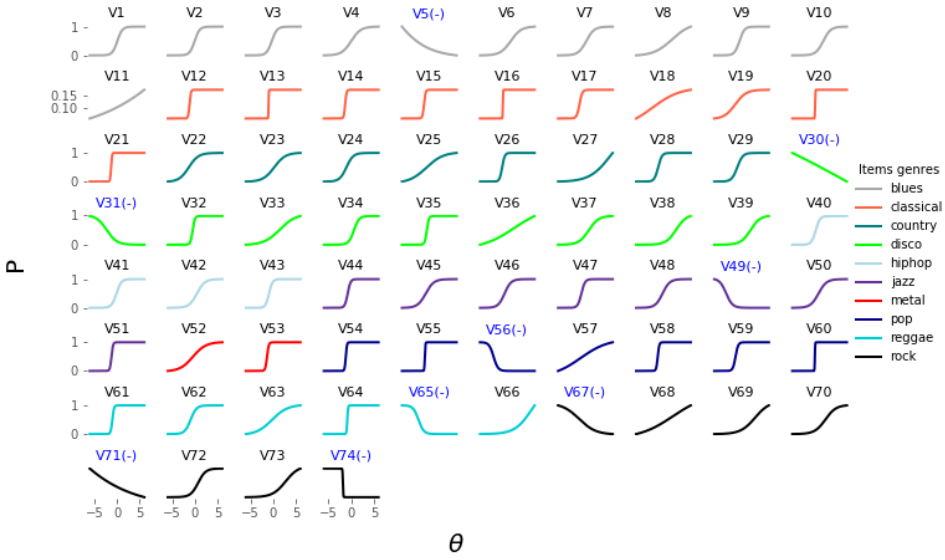

Figure 4.18: Icc for all test items

Figure 4.18 shows the item characteristic curves (ICC) for all 74 test items. In section C.0.0.1, the ICCs are shown for each items by their respective genres. The test items shown in figure 4.18 are presented by a color representing the genre to which the test item belongs. For instance the test item V1 is from the blues genre. On the x-axis of figure 4.18, the probability of a correct response is shown and on the y-axis the ability levels θ are presented.

4.2.2.2 Items information curves (IIC)

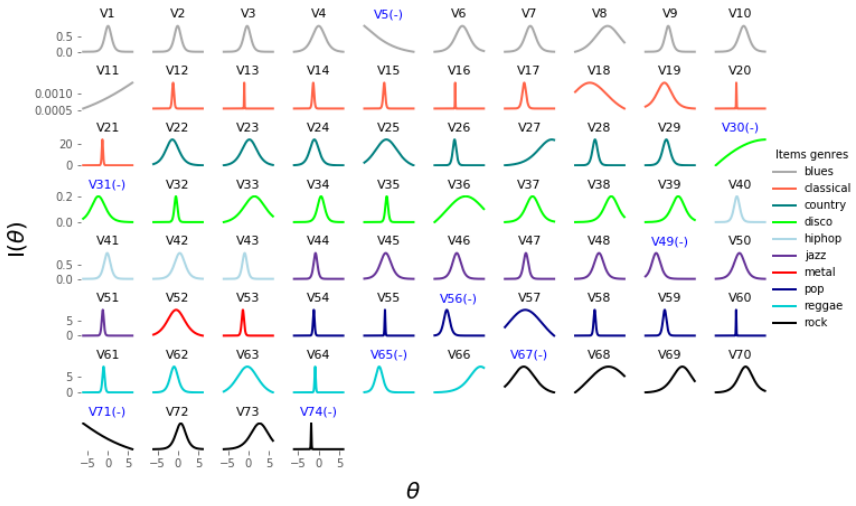

Figure 4.19: IIC for all test items

Figure 4.19 shows the information curves for all 74 test items. In section C.0.0.2, the information curves are shown for each items by their respective genres. Figure 4.19 is similar to figure 4.18 , the difference is that on the y-axis is presented the information level corresponding to different ability levels.

4.2.2.3 Items information and error curves

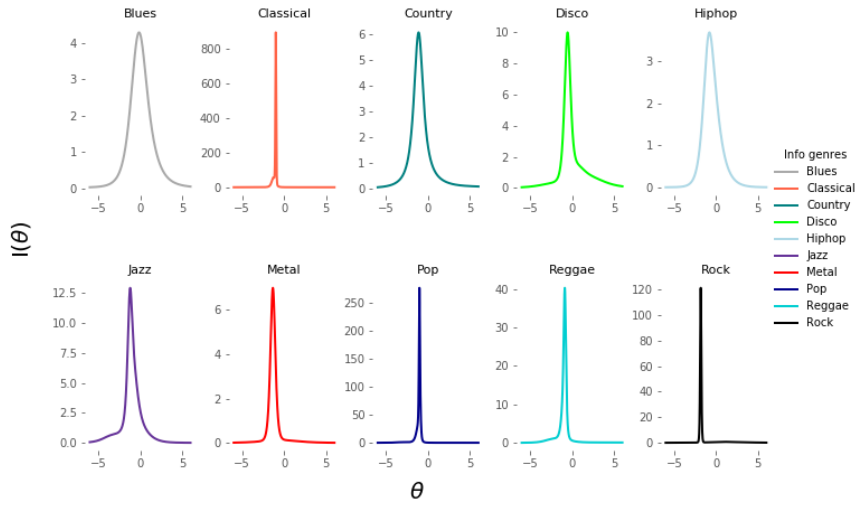

Figure 4.20: Information curves for all test items by genres

Figure 4.20 shows the information curves according to the the genres. The figure suggests that the test items from the classical genre are the most informative.

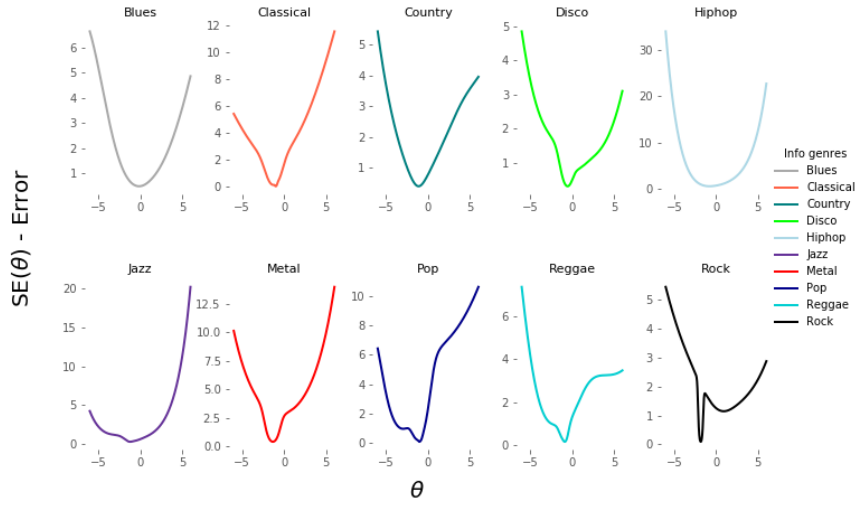

Figure 4.21: Error curves for all test items by genres

Figure 4.21 is similar to figure 4.20, that is instead of having a measure of how much information each genres contain; figure 4.21 shows a measure of the minimum error in determining the ability level of a given examinee for each of the genre classes. It can be seen from the figure that all items have a minimum error around -0.5 and -2, only the blues genre has a minimum error around 0.

4.3 Items with negative discriminability

Table 4.3: items with negative discriminant

Items	Index	Song	Discrimination	Difficulty
V5	23	blues.00023	-0.240784	-10.593972
V30	301	disco.00001	-0.063796	6.522547
V31	321	disco.00021	-0.894630	-2.295188
V49	587	jazz.00087	-1.408842	-3.402118
V56	769	pop.00069	-2.061430	-3.033730
V65	879	reggae.00079	-1.718398	-2.379427
V67	910	rock.00010	-0.542913	-1.408560
V71	933	rock.00033	-0.121095	-18.814481
V74	947	rock.00047	-22.011629	-1.821236

Table 4.3 shows the parameters for items with negative discriminant in the 2PL model. Those items exhibits an deviating behavior. The high-ability classifiers have a higher probability of answering wrong to those items than the The low-ability classifiers.

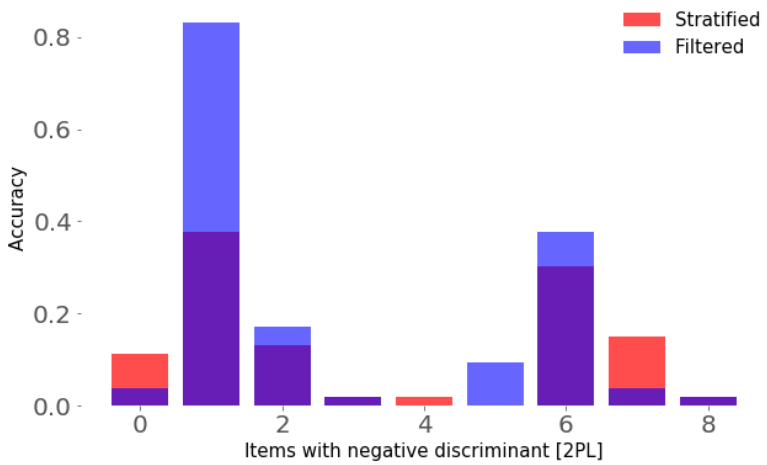

Figure 4.22: Barplot for the frequency of correct response for items with negative discriminant. The order of the items follows exactly the order shown in table 4.3.

Figure 4.22 shows the accuracy for the negative items for classifiers trained with filtered and stratified data.

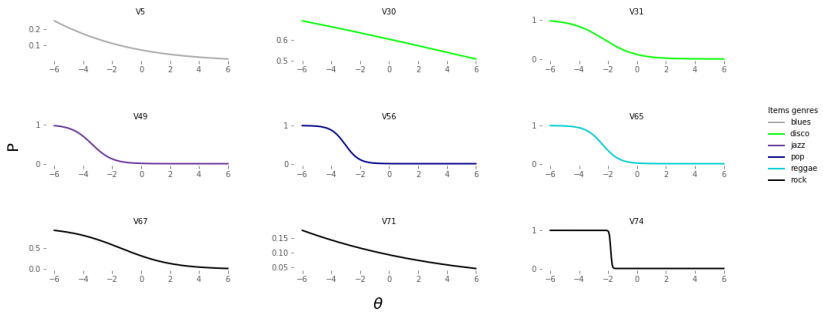

Figure 4.23: ICCs for test items with negative discriminant

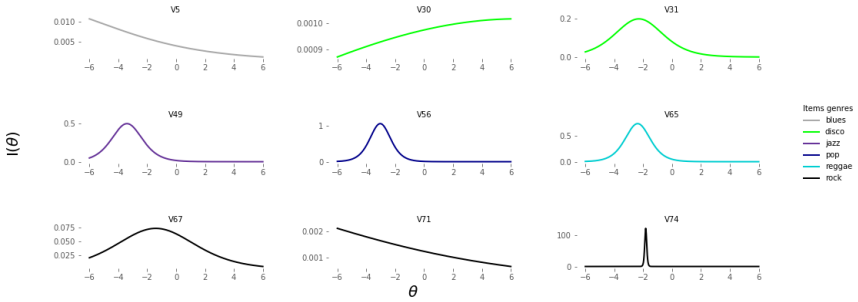

Figure 4.24: Information curves for test items with negative discriminant

Figure 4.23 and 4.24 show respectively the ICCs and the information curves for the test items having negative discriminant.

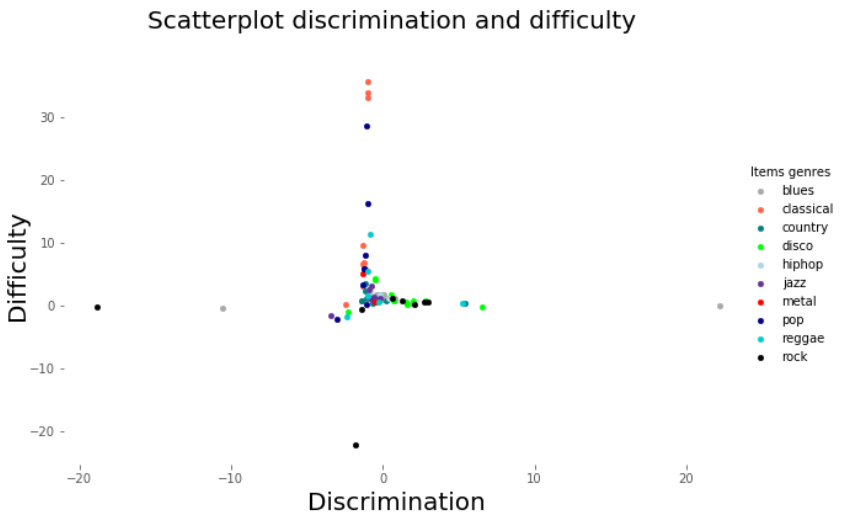

Figure 4.25: 2PL scatter-plot for all items

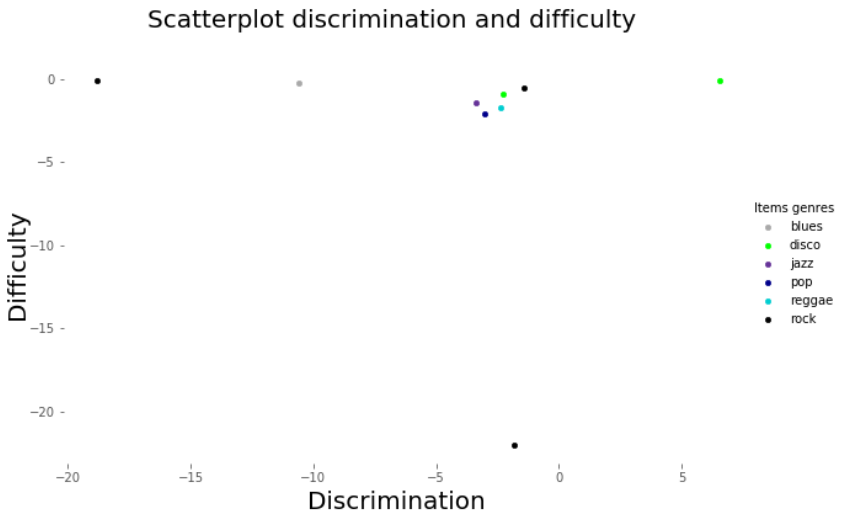

Figure 4.26: 2PL scatter-plot for items with negative discr.

Figure 4.25 and 4.26 show respectively the scatter plot for 2PL model and the test items having negative discriminant. The test items having negative discriminant are moderately difficult, where the test items from the rock genre are the most difficult.

4.4 Pitch shifted song music

Table 4.4: items with negative discriminant

Items	Index	Song	Discrimination	Difficulty
V5	23	blues.00023	-1.252245	-2.927820
V30	301	disco.00001	-0.690728	-0.494615
V31	321	disco.00021	-1.131139	-1.521004
V49	587	jazz.00087	-0.832103	-6.030717
V56	769	pop.00069	-2.318738	-2.795330
V65	879	reggae.00079	-0.341620	-2.362997
V71	933	rock.00033	-2.153143	-2.210614

Table 4.4 shows the parameters for items with negative discriminant in the 2PL model where the test items from the previous models have be modified. The test items are modified in such way that their pitch are shifted.

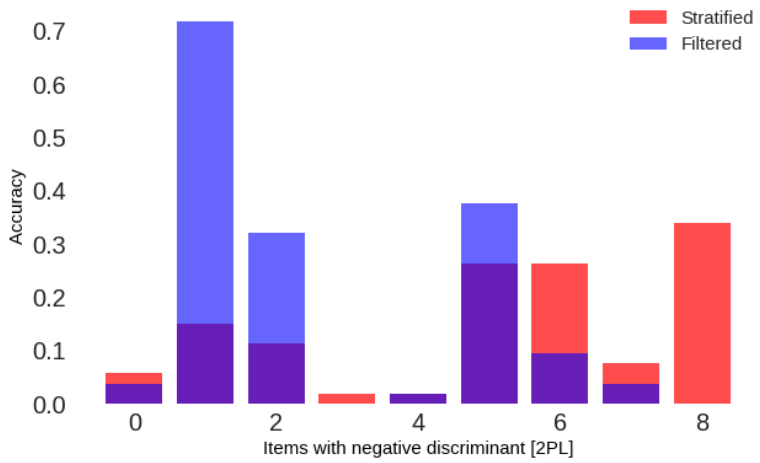

Figure 4.27: Barplot for frequency of correct rresponse neg. items

Compared to figure 4.22, in figure 4.27 the test evaluated in classifiers trained with stratified data are more frequently correctly answered. The number of items with negative discriminant have also reduced.

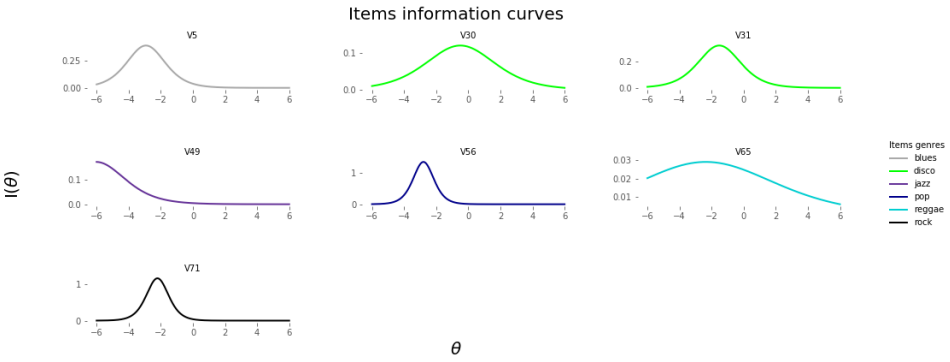

Figure 4.28: 2PL scatter-plot for all items

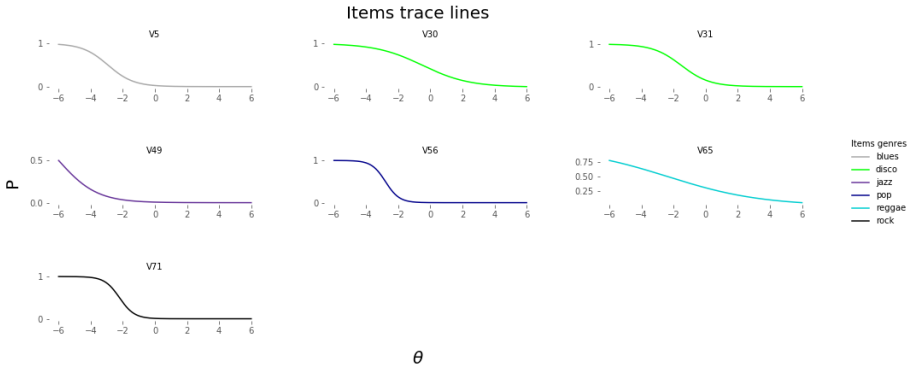

Figure 4.29: 2PL scatter-plot for items with negative discr.

Figure 4.28 and 4.29 show respectively the ICCs and the information curves for the test items having negative discriminant.

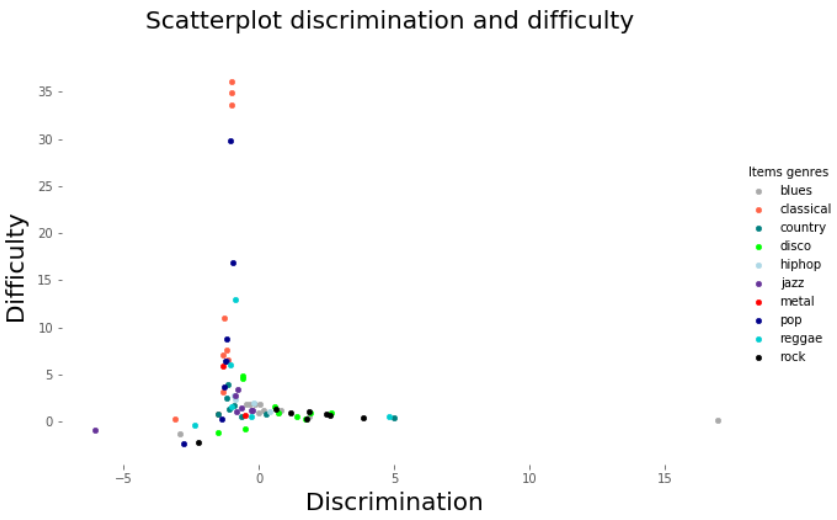

Figure 4.30: 2PL scatter-plot for all items

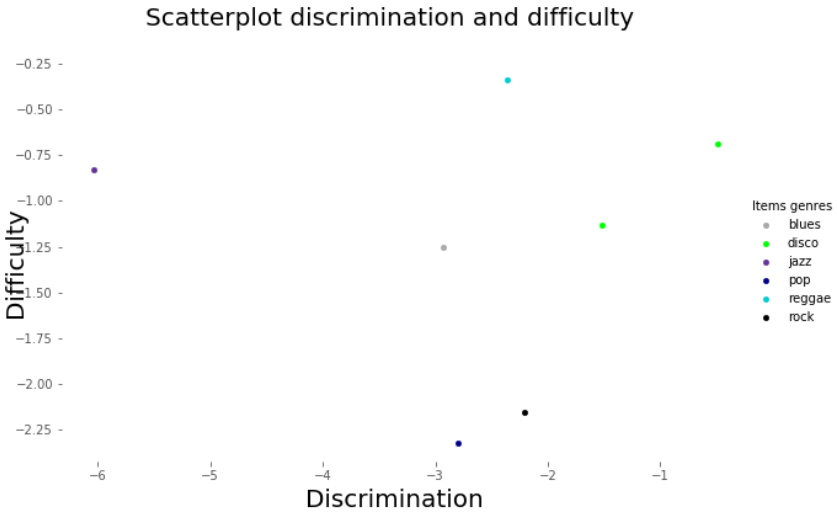

Figure 4.31: 2PL scatter-plot for items with negative discr.

Figure 4.30 and 4.31 show respectively the scatter plot for 2PL model and the test items having negative discriminant.

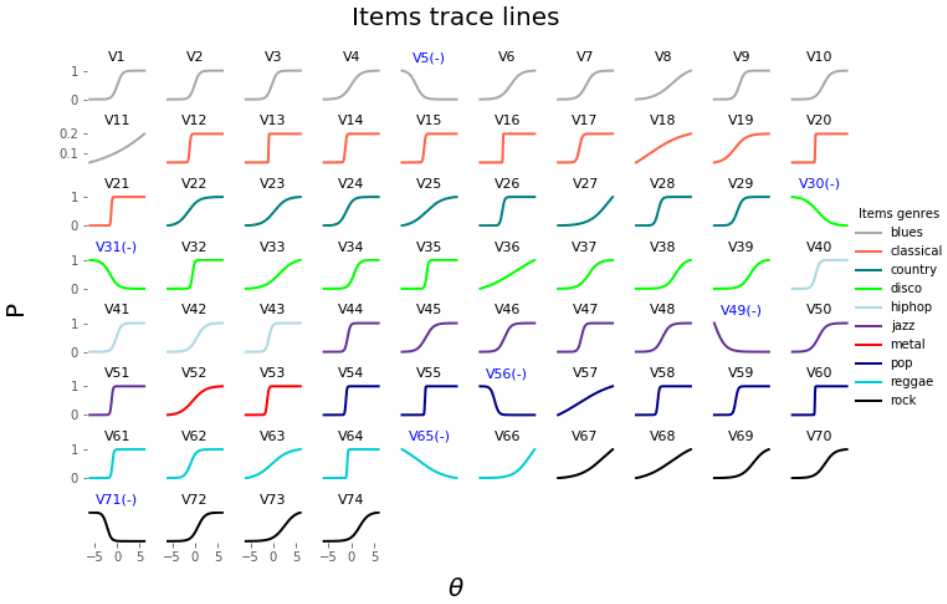

Figure 4.32: ICCs

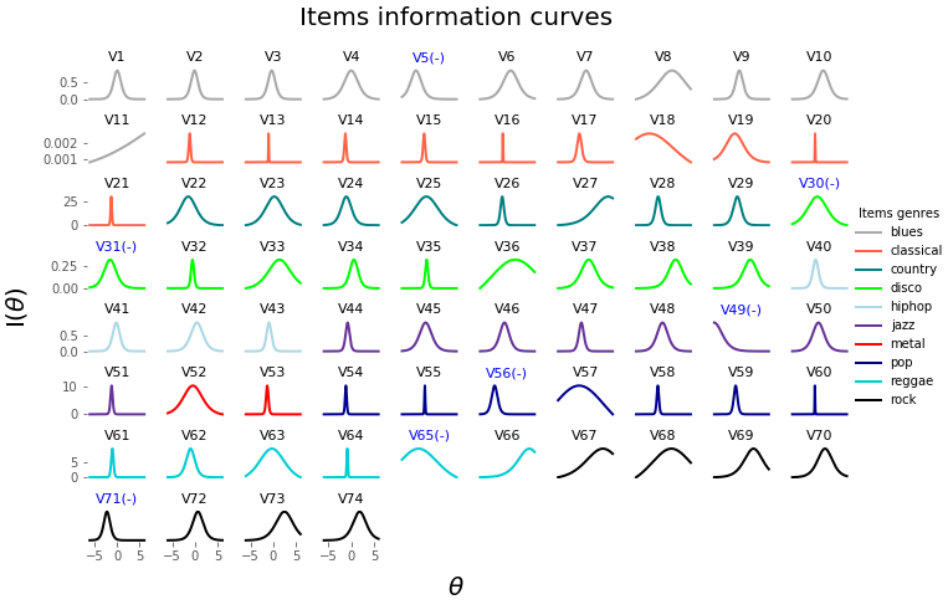

Figure 4.33: Information curves

Figures 4.32 and 4.33 show respectively the item characteristic and information curves for the new model where the test items with negative discriminant are pitch shifted.

4.5 Differential items functioning

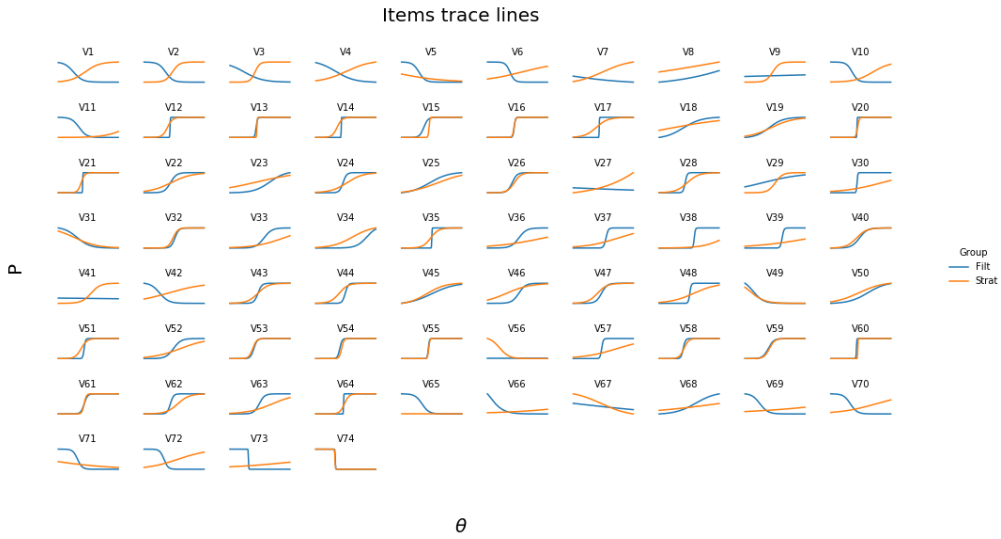

Figure 4.34: Differential item functioning for classifiers trained with filtered and stratified data

Figure 4.34 shows the differential item functioning for each test items. The differential item functioning shows how each test items varies depending on $group_1$ and $group_2$, which are respectively classifiers trained on filtered and stratified data.

Chapter 5

Discussion

Regarding the applications of IRT in music genre classification in particular and machine learning in general, it can be observed from the results of previous chapters that IRT can be successfully applied for analyzing the items' difficulty and the performances of the models and giving more detailed information than simply analyzing metrics such as accuracy, recall, confusion matrix.

Tables 4.1 and 4.2 have shown that there is a significant difference between the models trained with filtered data respective stratified data; moreover, the difference in the data are more important than the difference in the particular algorithms used to build the models. In this case, the data is a more determining factor than the algorithms.

It can be observed from figures 4.2 and 4.1 that the test items from the classical genre are frequently correctly classified independently on the given data on which the classifiers have be trained on. This may correspond to figure 3.1, where all the data points are projected on the 2D figure. In figure 3.1, all the songs data from the classical genres form a clear separate cluster, which may facilitates the classification task. On the other hand, the test items from the rock genres seem to be harder to classify correctly for most classifiers in the experiment. As it have been shown in [33], the rock genre has a broad diversity of artists. This fact is also confirmed when listening to songs from the rock genres. They possess particularly a broad variety of musicality, tones , rhythms and instruments. All those singular aspects may increase the difficulty to correctly classify test items from the rock genre.

A comparison between the accuracy of the classifiers trained with filtered and stratified data is presented figure 4.3. It can be observed that classifiers trained on stratified data have a generally higher accuracy on every item. This difference in the accuracy between the two groups of classifiers follow logi-

cally from the structure of the experimental setup; that is the classifiers trained with stratified data have been trained on items independently of the artist which implies that the same artist may be present both in the test and train data, and as a result the classifiers classify those items much more correctly.

In section 4.2.1, important key differences between different IRT models are presented. Figure 4.4 and 4.5 show respectively the correlation for the discriminability parameters and the correlation for the difficulty parameters for different IRT models. The difficulty and discriminability parameters estimated for test items in different IRT models seem to be uncorrelated. This uncorrelatedness or independence is comprehensible given the fact that the different IRT models are based on nonidentical assumptions and has a distinct number of parameters. For instance the 3PL IRT model has three freely estimated parameters; on the other hand, the 2PL model has only two three freely parameters.

The classifiers' abilities estimated from different IRT models seem to be correlated although the items' parameters estimated from different models seem to be independent from each other. The classifiers' abilities correlation can be observed in figure 4.6. As shown in figure 4.6, the classifiers' abilities are highly correlated.

As aforementioned the total score is the expected number of correctly answered or correctly classified test items in a given model for different ability levels. The total scores for different IRT models are presented in figure 4.7. Generally the IRT models having a higher number of parameters such as the 2PL and 3PL, have a more complex shape. For the 2PL and 3PL models, the expected number of items correctly classified are generally higher for low-ability classifiers and lower for high-ability classifiers compared to the Rash and 1Pl models. This can be explained by the fact that the complex models require more data in order to estimate all the parameters precisely.

Figure 4.8, 4.9, 4.10 and 4.11 show the information curves for different IRT models; on the other hand, figure 4.12, 4.13, 4.14 and 4.15 show the error curves for respectively the Rasch, 1PL, 2PL and 3PL models. The information curve gives a measure of which ability interval the test can reliably estimate the classifiers ability [11]. It can be observed from those aforementioned figures that the given test is suitable for moderate able classifiers. In fact, the maximum information value is obtained when the ability level is around zero and the same is true for the minimum error value. It can be therefore concluded that the given test items can reliably estimate the ability of classifiers with an ability level around zero.

It should be noted; however, that for the 2PL and 3PL models the informa-

tion and error curves are more complex. Especially for the 3PL models there is no clear peak in the information curve and no clear interval can be observed, where the items can reliably estimate the classifiers' ability. This can be due to the fact that the 3PL model has a significantly large number of parameters to be estimated. Furthermore, the number of test items are not significantly large in order to reliable estimate all those parameters. Therefore, without lack for generality the discussion is pursued in terms of the 2PL model, which better explain the data.

Based on figure 4.16 no correlation can be observed between the difficulty and discriminability parameters. However, on figure 4.17, it is clear that a correlation can be observed between the ability parameters and the accuracy of the classifiers. It can be observed that when the model fits the data, the accuracy does correlate with the ability parameters [4].

As previously stated the classifiers trained with stratified data have a generally higher accuracy which is also true for the ability as shown in table C.3. The fact that the accuracy does correlate with the ability is key point for implementing in IRT in machine learning, although there have been results where a high accuracy did not necessary imply an high IRT score [25], whose definition the authors did not clarify. Nonetheless, the observed correlation between the accuracy and the ability is logical from a theoretical and intuitive point of view. For instance, given a reliable test instrument, the fundamental assumption is that an examinee responding correctly to most test items in the test instrument is more able or has more knowledge than an another examinee failing to respond correctly to the same test items. Otherwise, the test instrument would be faulty or the examinees would be cheating.

Figure 4.18 shows the a summary of the item characteristic curves for each items in the test data. Some items have a negative discriminant as shown by their shapes. For instance items V5, V30, V31, V49, V56, V65, V67, V71, V74 have a negative discriminant. A negative discriminant is an indication that the items may have been represented in a faulty form. In fact, a negative discriminant violates the assumption that the high-ability classifiers should respond frequently more correctly compared to the law-ability classifiers. Nonetheless, for solving the problem that items with negative discriminant pose, different approaches have been proposed [4].

Depending on the forms of the items, the items with negative discriminant may either be disregarded from the test items or modified in such way that the high-ability classifiers correctly classify those items when the low-ability does. In music classification, those modifications can be easily performed by modifying the property of the music data. The experiment presented in section

4.4 shows the items with negative discriminant obtained after transforming the negative items from section 4.3 in such way that the pitch is shifted. It can be observed that the number of negative items is reduced and that the ICC for most items is modify in a positive way. For an particular application, if an suitable modifications can be found, it should be implemented, otherwise the negative items should be removed from the test items.

In [4], the authors suggest that the negative items should be removed only if the IRT model is used for model evaluation. On the other hand, when building the machine learning model, the items with negative discriminant should not be disregarded as they may be necessary for increasing the model's generalisation capacity by "seeing" as many examples as possible. It should be noted; however, that even though the quantity of the training data may influence an ML model's accuracy, the quality of the data is an even more critical and valuable property for building a robust model. A model trained on poor data such as items with negative discriminant may not necessarily performs well, especially if the correlation between the ability and the accuracy holds.

In section C.0.0.1, each characteristic curves are shown in details in their corresponding genre category. Figures C.1, C.2, C.3, C.4, C.5, C.6, C.7, C.8, C.9, C.10, and table C.2 suggest that the items in genre categories *Blues*, *Disco*, *Rock* are generally harder and that the items in category *Classical* are generally easier.

Figure 4.19 shows a summary of the information curves for each items in the test data. It can be observed that the items in category *Classical* are more precise in such way the estimation the classifiers' ability is more precise. However, the items in category *Rock* are more diverse and imprecise.

Figures in section C.0.0.2 suggest that the test items is generally suited for classifiers with moderate ability. In section 4.2.2.3, and in figure 4.20 the same result is shown for each genres categories. There is indeed some differences between the genres; nonetheless the peak is obtained generally around zero.

In section 4.3, the data of items with negative discriminant are presented in details. There are in total nine items with negative discriminant in 2PL model. it can be observed from section 4.3 that the items from the category *Rock* are the most represented. This may be comprehensible due to the fact that the data suggest that the *Rock* genre category are hard to correctly classify.

Moreover, a particular property about items with negative discriminant are the fact that the generally able classifiers fail to correctly classify those items and the less able classifiers particularly succeed to classify the same items as shown in figure 4.22. On the other hand, figures 4.23 and 4.24 show respectively the ICCs and the information curves for the test items having negative

discriminant. Furthermore, figure 4.23 show that the items are generally easier and figure 4.24 suggest that test can estimate precisely only the ability of moderate able classifiers.

On the other hand, in section 4.4 similar result as in section 4.3 are presented. The results on section 4.4 is obtained by modifying the pitch of the songs. Furthermore, it can be observed that the number of items are reduced and that the more able classifiers more frequently correctly classify those items as shown in figure 4.27.

In figure 4.28 the ICCs is presented and it can be observed that the discriminability value has increased. Furthermore, it can be observed from figure 4.29 that the information curves and the peak are much more higher than the figures shown in section 4.3.

Figure 4.34 shows the items differential function between the classifiers trained with filtered and stratified data. There are two main observations that can be made from figure 4.34. The first observation is the fact that items V5, V30, V31, V49, V56, V65, V67, V71, V74, whose discriminant are negative are more likely to have a high discriminability for classifiers trained with filtered data than those trained with stratified data. One hypothetical reason for such behaviour may be the fact that classifiers trained with filtered do no rely on artist information thereby able classifiers can be easily discriminate from less able classifiers. The second observation is the fact that the test items have frequently a negative discriminant for the group of classifiers trained with filtered data. This observation is a good indication that there is something wrong either with the items or the models. This indication can be further analysed in order to evaluate and choose the right model.

Chapter 6

Conclusions

Items response theory may be applied as an evaluation tool for the quality of the data during training and as a model evaluation tool for chosen the machine learning models. IRT is a more rigorous method compared to the current evaluation methods in machine learning. It should; however, be noted that item response theory requires a great deal of test items in order to reliably estimates all the required parameters. The sufficient number of test data is not always available especially in machine learning where the test data constitutes a small of amount the whole data which is usually used for model training. Therefore, IRT may not always be applicable and practical in every machine learning experiments and projects. This is one of the biggest challenges for the application of item response theory in machine learning.

The accuracy measure is a fundamental metric for models evaluation; however, IRT may have more advantages. Whenever applicable, IRT can give a measure of the difficulty of different test items such that the models may not just be compared based on the total performance but also on theirs ability to classify harder test items. Furthermore, the item characteristic curves (ICC) and the item information curves (IIC) give more insight for reliably assessing the data quality. The ICC gives information about the discriminiability and difficulty of test items and the IIC gives insights about the amount of information each test items possess, that is the reliability of the test item for determining the ability of the classifiers.

This experiment have shown some of the advantages of using IRT as a analysis tool in music genre classification in general and machine learning in particular. In fact, the experiment have shown that there are a significant difference in the ability and accuracy of models trained with filtered data respective stratified data. Models trained with stratified data have higher accuracy than

those trained with filtered data, which may confirm the assumption that the stratified models rely more on artists information than filtered models. Furthermore, given the fact that the accuracy is correlated or proportional to the ability, the data also suggest that models trained with stratified data are more able. This suggest that the quality of data and the assumptions made in data do influence the ability and the accuracy of the models.

It has been shown also that the negative items can be an good indication of an eventual fault in the definition of correctness particularly those items having a negative discriminant. Those items should be modified if possible or otherwise removed as they may influence the classifiers' ability as a result. Furthermore, it also be shown in section 4.4 that the items can be modified in order to change the sign of the discriminability parameters, whenever possible. However, a critical challenge is how to find an appropriate transformation or modification such as the particular characteristics of the sound is intact and unaltered, whenever the items can not be disregarded from the test set.

Given the fact that a model do generate a great deal of items with negative discriminant, it may be possible that increasing the accuracy and the capability of an classifier or an ensemble of classifiers may not produce the expected outcomes in the presence of test items having negative discriminant. Contrariwise, increasing the ability in those cases may decrease the generalisation ability of the classifier, which may imply a less optimal performance for items having negative discriminant.

Moreover, it should be noted that the study done in this thesis have been less focused on the classifiers' characteristics but more on the test items and data on which the classifiers are trained on. A further investigation can be made regarding the classifiers difference in their ability to correctly classify music data. Moreover, the study is performed under the dichotomous IRT. As aforementioned music genre classification is a multi-class problem, therefore a further investigation about polytomous IRT may be an interesting path. Item response theory may also be applied more directly during the machine learning training process by successively training and evaluating models until a final set of appropriate data and models are retained. Further investigation may be appropriate.

Acknowledgments

Thanks to the examineer **Sten Ternström** and my supervisor **Bob sturm** for all the advice and guides throughout this thesis.

Bibliography

[1] Ronald K Hambleton et al. *Fundamentals of Item Response Theory*. 1st ed. SAGE Publications, 1991, p. 183. ISBN: 080393646X.

[2] Frank B. Baker and Seock-Ho Kim. *The Basics of Item Response Theory Using R*. Springer, 2017, p. 173. ISBN: 9783319542041. DOI: 10.1080/15366367.2018.1462078.

[3] R. J. de Ayala. *The Theory and Practice of Item Response Theory*. 1 edition. The Guilford Press, 2008, p. 466. ISBN: 9781593858698. DOI: 10.1080/15305058.2011.556771.

[4] Fernando Martínez-Plumed et al. "Item response theory in AI: Analysing machine learning classifiers at the instance level". In: *Artificial Intelligence* 271 (2019), pp. 18–42. ISSN: 00043702. DOI: 10.1016/j.artint.2018.09.004. URL: https://doi.org/10.1016/j.artint.2018.09.004.

[5] Ricardo B. C. Prudêncio, J. Hernández-Orallo, and Adolfo Martínez-Usó. "Analysis of Instance Hardness in Machine Learning Using Item Response Theory". In: *Second International Workshop on Learning over Multiple Contexts (ECML 2015)* (2015). URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.716.4340%7B%5C&%7Drep=rep1%7B%5C&%7Dtype=pdf.

[6] Fernando Martínez-Plumed et al. "Making sense of Item response theory in machine learning". In: *Frontiers in Artificial Intelligence and Applications* 285.September (2016), pp. 1140–1148. ISSN: 09226389. DOI: 10.3233/978-1-61499-672-9-1140.

[7] Stuart Jones. "An Analysis of Item Response Theory". In: (2019), p. 62. URL: https://etd.auburn.edu/handle/10415/6773.

[8] R Benitez Rochel et al. “Neural networks applied to Item Response Theory”. In: *Proceeding of the ICSC Symposia on Neural Computation (NC’2000) May 23-26, 2000 in Berlin, Germany* (2000), pp. 23–26. URL: http://www.lcc.uma.es:8080/repository/fileDownloader?rfname=LCC559.pdf.

[9] John Lalor et al. “Understanding Deep Learning Performance through an Examination of Test Set Difficulty: A Psychometric Case Study”. In: (2019), pp. 4711–4716. DOI: 10.18653/v1/d18-1500. arXiv: 1702.04811.

[10] G. Tzanetakis and P. Cook. “Musical genre classification of audio signals”. In: *IEEE Trans. Speech Audio Process.* 10.5 (July 2002), pp. 293–302.

[11] Frank B. Baker and Kim Seock-Ho. *Item Response Theory Parameter Estimation Techniques*. 2nd. CRC Press, 2004, p. 528. ISBN: 9780824758257.

[12] Insu Paek and Ki Cole. *Using R for Item Response Theory model applications*. Routledge, 2019, p. 272. ISBN: 9781138542792.

[13] R. Darrell Bock and Michele F. Zimowski. “Multiple Group IRT”. In: *Handbook of Modern Item Response Theory*. Ed. by Wim J. van der Linden and Ronald K. Hambleton. New York, NY: Springer New York, 1997, pp. 433–448. ISBN: 978-1-4757-2691-6. DOI: 10.1007/978-1-4757-2691-6_25. URL: https://doi.org/10.1007/978-1-4757-2691-6_25.

[14] Francisco Rodríguez-Algarra, Bob L. Sturm, and Simon Dixon. “Characterising Confounding Effects in Music Classification Experiments through Interventions”. In: *Transactions of the International Society for Music Information Retrieval* 2.1 (2019), p. 52. DOI: 10.5334/tismir.24.

[15] Jeremy Watt, Reza Borhani, and Aggelos K. Katsaggelos. *Machine Learning Refined: Foundations, Algorithms, and Applications*. Cambridge University Press, 2016. ISBN: 9781316402276.

[16] Kubat Miroslav. *An Introduction to Machine Learning*. 2nd ed. Springer International Publishing, 2017, p. 348. ISBN: 9783319639123. URL: https://link.springer.com/book/10.1007%7B%5C%%7D2F978-3-319-63913-0.

[17] Simon Rogers and Mark A. Girolami. *A First Course in Machine Learning*. Chapman and Hall / CRC machine learning and pattern recognition series. CRC Press, 2011, pp. I–XX, 1–285. ISBN: 978-1-43-982414-6.

[18] Ethem Alpaydin. *Introduction to Machine Learning*. 2nd ed. The MIT Press, 2014. ISBN: 978-0-262-01243-0.

[19] Stephen Marsland. *Machine Learning - An Algorithmic Perspective*. Chapman and Hall / CRC machine learning and pattern recognition series. CRC Press, 2009, pp. I–XVI, 1–390. ISBN: 978-1-4200-6718-7.

[20] Cristian Zanon et al. "An application of item response theory to psychological test development". In: *Psicologia: Reflexao e Critica* 29.1 (2016). ISSN: 16787153. DOI: 10.1186/s41155-016-0040-x. URL: http://dx.doi.org/10.1186/s41155-016-0040-x.

[21] Tsuyoshi Idé and Amit Dhurandhar. "Supervised item response models for informative prediction". In: *Knowledge and Information Systems* 51.1 (2017), pp. 235–257. ISSN: 02193116. DOI: 10.1007/s10115-016-0976-2.

[22] Chun-Kit Yeung. "Deep-IRT: Make Deep Learning Based Knowledge Tracing Explainable Using Item Response Theory". In: (2019). arXiv: 1904.11738. URL: http://arxiv.org/abs/1904.11738.

[23] Ziheng Chen and Hongshik Ahn. "Item Response Theory based Ensemble in Machine Learning". In: (2019), pp. 1–35. arXiv: 1911.04616. URL: http://arxiv.org/abs/1911.04616.

[24] Konstantinos Pliakos et al. "Integrating machine learning into item response theory for addressing the cold start problem in adaptive learning systems". In: *Computers and Education* 137.April (2019), pp. 91–103. ISSN: 03601315. DOI: 10.1016/j.compedu.2019.04.009. URL: https://doi.org/10.1016/j.compedu.2019.04.009.

[25] John P. Lalor, Hao Wu, and Hong Yu. "Building an evaluation scale using item response theory". In: *EMNLP 2016 - Conference on Empirical Methods in Natural Language Processing, Proceedings* (2016), pp. 648–657. DOI: 10.18653/v1/d16-1062. arXiv: 1605.08889.

[26] B. L. Sturm. "A Survey of Evaluation in Music Genre Recognition". In: *Adaptive Multimedia Retrieval: Semantics, Context, and Adaptation*. Ed. by A. Nürnberger et al. Vol. LNCS 8382. Oct. 2014, pp. 29–66.

[27] Bob L. Sturm. “The State of the Art Ten Years After a State of the Art: Future Research in Music Information Retrieval”. In: *Journal of New Music Research* 43.2 (2014), pp. 147–172. ISSN: 17445027. DOI: 10.1080/09298215.2014.894533. URL: http://dx.doi.org/10.1080/09298215.2014.894533.

[28] Elias Pampalk, Arthur Flexer, and Gerhard Widmer. “Improvements of audio-based music similarity and genre classificaton”. In: *ISMIR 2005 - 6th International Conference on Music Information Retrieval* (2005), pp. 628–633.

[29] Theodoros Giannakopoulos and Aggelos Pikrakis. *Introduction to Audio Analysis: A MATLAB Approach*. 2014, pp. 1–266. ISBN: 9780080993881. DOI: 10.1016/C2012-0-03524-7.

[30] Brian McFee et al. “librosa: Audio and Music Signal Analysis in Python”. In: *Proceedings of the 14th Python in Science Conference* Scipy (2015), pp. 18–24. DOI: 10.25080/majora-7b98e3ed-003.

[31] Eric Tarr. *Hack audio: an introduction to computer programming and digital signal processing in MATLAB*. Routledge, 2018, p. 493. ISBN: 9781138497559. URL: http://capitadiscovery.co.uk/uwl/items/702311.

[32] Kandethody M. Ramachandran and Chris P. Tsokos. *Mathematical Statistics with Applications in R: Second Edition*. 2014, pp. 1–800. ISBN: 9780124171138. DOI: 10.1016/C2012-0-07341-3.

[33] B. L. Sturm. “The State of the Art Ten Years After a State of the Art: Future Research in Music Information Retrieval”. In: *J. New Music Research* 43.2 (2014), pp. 147–172.

Appendix A

Test data

Table A.1: Test data

Item	Index	File name	Title
V1	12	blues.00012.wav	Cross Road Blues
V2	13	blues.00013.wav	Terraplane Blues
V3	15	blues.00015.wav	Walking blues
V4	20	blues.00020.wav	Preachin' Blues
V5	23	blues.00023.wav	Stones In My Passway
V6	24	blues.00024.wav	Traveling Riverside Blues
V7	25	blues.00025.wav	Milkcow's Calf Blues
V8	28	blues.00028.wav	Traveling Riverside Blues
V9	62	blues.00062.wav	The Things I Did for You
V10	64	blues.00064.wav	Clifton's Squeeze-box Boogie
V11	98	blues.00098.wav	Iceman
V12	116	classical.00016.wav	Symphony No.38 in D, K.504, "Prague", Andante
V13	119	classical.00019.wav	Symphony No.39 in E flat, K.543, Andante con moto
V14	121	classical.00021.wav	Symphony No.39 in E flat, K.543, Finale (Allegro)
V15	124	classical.00024.wav	Symphony No.40 in G minor, K.550, Menuetto (Al...
V16	129	classical.00029.wav	Symphony No.41 in C, K.551, "Jupiter", Molto a...
V17	134	classical.00034.wav	Quartet in F Major for Strings, Allegro modera...
V18	135	classical.00035.wav	Quartet in F Major for Strings, Assez vif - Tr...
V19	136	classical.00036.wav	Quartet in F Major for Strings, Tres lent
V20	137	classical.00037.wav	Quartet in F Major for Strings, Vif et agite
V21	141	classical.00041.wav	Ainsi la nuit for String Quartet: VII. Temps s...
V22	231	country.00031.wav	Come On Joe

V23	233	country.00033.wav	Teach me to cheat
V24	247	country.00047.wav	I'm the Girl in the USA
V25	250	country.00050.wav	Never Alone
V26	251	country.00051.wav	Never Knew Lonely
V27	253	country.00053.wav	Liza Jane
V28	257	country.00057.wav	The Heart Won't Lie
V29	264	country.00064.wav	I Will Always Love You
V30	301	disco.00001.wav	I Will Survive
V31	321	disco.00021.wav	Never Can Say Goodbye
V32	358	disco.00058.wav	Boogie Wonderland
V33	376	disco.00076.wav	Thank God It's Friday
V34	377	disco.00077.wav	Heaven must be missing an angel
V35	379	disco.00079.wav	Love Is Just The Game
V36	380	disco.00080.wav	nan
V37	382	disco.00082.wav	Disco Nights (Rock-Freak)
V38	386	disco.00086.wav	I love the night life
V39	394	disco.00094.wav	WHY?
V40	462	hiphop.00062.wav	Keeping It Moving
V41	465	hiphop.00065.wav	Check The Rhime
V42	470	hiphop.00070.wav	Buggin' Out
V43	475	hiphop.00075.wav	Stressed Out
V44	573	jazz.00073.wav	Scrapple From The Apple
V45	576	jazz.00076.wav	Stairway To The Stars
V46	578	jazz.00078.wav	Our Love Is Here to Stay
V47	584	jazz.00084.wav	Juicy Fruit
V48	585	jazz.00085.wav	Crystal Palace
V49	587	jazz.00087.wav	So What
V50	588	jazz.00088.wav	Freddie Freeloader
V51	599	jazz.00099.wav	Teo
V52	630	metal.00030.wav	Trust
V53	678	metal.00078.wav	Prince Charming
V54	700	pop.00000.wav	Saturate Me
V55	766	pop.00066.wav	Could I Have This Kiss Forever
V56	769	pop.00069.wav	If
V57	770	pop.00070.wav	I Get Lonely
V58	788	pop.00088.wav	Candy (Wade Robson Remix)
V59	791	pop.00091.wav	Everything My Heart Desires
V60	796	pop.00096.wav	Quit Breaking My Heart
V61	835	reggae.00035.wav	Dub Old Timer

V62	846	reggae.00046.wav	It's Magic
V63	848	reggae.00048.wav	Westbound Train
V64	868	reggae.00068.wav	Wolf and Leopards
V65	879	reggae.00079.wav	Chicken Halk Dub
V66	883	reggae.00083.wav	Many Rivers To Cross
V67	910	rock.00010.wav	Worthy
V68	911	rock.00011.wav	Cradle and All
V69	931	rock.00031.wav	Honky Tonk Women
V70	932	rock.00032.wav	All my love
V71	933	rock.00033.wav	Gimme Shelter
V72	939	rock.00039.wav	The Song Remains The Same
V73	942	rock.00042.wav	D'yer Mak'er
V74	947	rock.00047.wav	The Wanton Song

Table A.2: classifiers' accuracy

Index	Classifiers	Filtered train data	Stratified train data
1	KNN1	0.459459	0.662162
2	KNN2	0.432432	0.702703
3	KNN3	0.418919	0.675676
4	KNN4	0.459459	0.702703
5	KNN5	0.432432	0.459459
6	KNN6	0.418919	0.472973
7	SVC1	0.554054	0.743243
8	SVC2	0.527027	0.756757
9	SVC3	0.459459	0.716216
10	SVC4	0.567568	0.770270
11	SVC5	0.148649	0.216216
12	SVC6	0.540541	0.716216
13	GaussProc1	0.500000	0.621622
14	GaussProc2	0.108108	0.108108
15	GaussProc3	0.554054	0.797297
16	GaussProc4	0.513514	0.581081
17	GaussProc5	0.567568	0.810811
18	GaussProc6	0.432432	0.513514
19	DecTree1	0.297297	0.378378
20	DecTree2	0.270270	0.445946
21	DecTree3	0.324324	0.445946
22	DecTree4	0.297297	0.472973
23	DecTree5	0.405405	0.527027
24	DecTree6	0.310811	0.405405
25	RandForest1	0.472973	0.675676
26	RandForest2	0.486486	0.702703
27	RandForest3	0.472973	0.675676
28	RandForest4	0.486486	0.689189
29	RandForest5	0.486486	0.743243
30	RandForest6	0.486486	0.702703
31	MLP1	0.500000	0.729730
32	MLP2	0.594595	0.702703
33	MLP3	0.500000	0.702703
34	MLP4	0.567568	0.716216
35	MLP5	0.527027	0.648649

36	MLP6	0.540541	0.797297
37	Adaboost1	0.513514	0.648649
38	Adaboost2	0.364865	0.405405
39	Adaboost3	0.500000	0.675676
40	Adaboost4	0.378378	0.472973
41	Adaboost5	0.486486	0.635135
42	Adaboost6	0.378378	0.432432
43	LogisticRegr1	0.527027	0.702703
44	LogisticRegr2	0.486486	0.635135
45	LogisticRegr3	0.432432	0.581081
46	LogisticRegr4	0.527027	0.702703
47	LogisticRegr5	0.472973	0.621622
48	LogisticRegr6	0.527027	0.702703
49	NaiveBayes	0.391892	0.594595
50	ExtraTree1	0.445946	0.675676
51	ExtraTree2	0.459459	0.743243
52	DummyClassifier	0.135135	0.135135
53	Voting Hard (SVC)	0.513514	0.743243

Appendix B

Negative items data

Table B.1: Items with negative discriminant [2PL]

Items	Index	Song	Discrimination	Difficulty	Guessing
V5	23	blues.00023	-0.240784	-10.593972	0
V30	301	disco.00001	-0.063796	6.522547	0
V31	321	disco.00021	-0.894630	-2.295188	0
V49	587	jazz.00087	-1.408842	-3.402118	0
V56	769	pop.00069	-2.061430	-3.033730	0
V65	879	reggae.00079	-1.718398	-2.379427	0
V67	910	rock.00010	-0.542913	-1.408560	0
V71	933	rock.00033	-0.121095	-18.814481	0
V74	947	rock.00047	-22.011629	-1.821236	0

Table B.2: Items with negative discriminant [3PL]

Unnamed: 0	Index	Song	Discrimination	Difficulty	Guessing
V5	23	blues.00023	-0.408597	-7.482852	2.498894e-02
V31	321	disco.00021	-0.684272	-2.926245	1.210056e-05
V49	587	jazz.00087	-0.917728	-5.007846	1.824013e-06
V56	769	pop.00069	-1.218435	-4.757381	1.230530e-06
V65	879	reggae.00079	-1.055526	-3.554830	1.824410e-06
V67	910	rock.00010	-0.447174	-1.715802	9.451078e-05
V71	933	rock.00033	-26.991799	-2.901290	7.699989e-02
V74	947	rock.00047	-27.021256	-2.897788	8.430718e-07

Appendix C

Items response theory

Table C.1: Models' summary

	Models	AIC	AICc	SABIC	BIC	logLik	df	p
A	Rasch	7111.735	7491.735	7192.698	7311.493	-3480.868	NaN	NaN
B	2PL	6636.231	5610.557	6795.998	7030.420	-3170.116	73.0	0.0
A	2PL	6636.231	5610.557	6795.998	7030.420	-3170.116	NaN	NaN
B	3PL	6498.016	5651.760	6737.666	7089.300	-3027.008	74.0	0.0
A	3PL	6498.016	5651.760	6737.666	7089.300	-3027.008	NaN	NaN
B	3PLg	6494.722	5648.466	6734.372	7086.005	-3025.361	0.0	0.0
A	2PL	6636.231	5610.557	6795.998	7030.420	-3170.116	NaN	NaN
B	3PLg	6494.722	5648.466	6734.372	7086.005	-3025.361	74.0	0.0

Table C.2: Items' parameters 2PL [sorted]

Items	Title	Discrimination	Difficulty	Guessing
V11	blues.00098	0.097076	22.163208	0
V30	disco.00001	-0.063796	6.522547	0
V27	country.00053	0.447461	5.429611	0
V66	reggae.00083	0.558242	5.231013	0
V69	rock.00031	0.634717	2.991001	0
V38	disco.00086	0.905797	2.806965	0
V73	rock.00042	0.700547	2.747219	0
V68	rock.00011	0.269349	2.109375	0
V39	disco.00094	0.944641	1.981166	0
V8	blues.00028	0.494098	1.894428	0
V36	disco.00080	0.281473	1.578860	0
V33	disco.00076	0.566253	1.493394	0
V70	rock.00032	0.856909	1.281789	0
V10	blues.00064	1.229544	0.811994	0
V6	blues.00024	0.876974	0.771300	0
V37	disco.00082	0.923208	0.736042	0
V72	rock.00039	1.219407	0.669168	0
V34	disco.00077	1.756944	0.588379	0
V42	hiphop.00070	1.177062	0.407393	0
V23	country.00033	0.814354	0.255689	0
V7	blues.00025	1.161047	0.191266	0
V1	blues.00012	1.835191	0.069489	0
V4	blues.00020	0.962188	0.009934	0
V2	blues.00013	1.859835	-0.103023	0
V41	hiphop.00065	1.896640	-0.138976	0
V48	jazz.00085	1.274304	-0.195106	0
V50	jazz.00088	1.143364	-0.229453	0
V63	reggae.00048	0.581009	-0.246372	0
V3	blues.00015	1.756419	-0.290678	0
V9	blues.00062	1.871207	-0.450797	0
V52	metal.00030	0.703861	-0.472998	0
V32	disco.00058	4.126773	-0.535170	0
V35	disco.00079	4.447365	-0.539366	0
V46	jazz.00078	1.419049	-0.603713	0
V25	country.00050	0.535114	-0.633506	0

V44	jazz.00073	3.239105	-0.770704	0
V45	jazz.00076	1.127099	-0.781734	0
V40	hiphop.00062	2.158550	-0.853476	0
V64	reggae.00068	11.482652	-0.857994	0
V47	jazz.00084	2.540334	-0.877995	0
V43	hiphop.00075	2.619900	-0.899298	0
V29	country.00064	1.652602	-0.924906	0
V55	pop.00066	16.428653	-0.968942	0
V62	reggae.00046	1.562438	-0.981867	0
V16	classical.00029	33.957433	-0.985960	0
V20	classical.00037	35.786034	-0.986611	0
V13	classical.00019	33.151908	-1.002400	0
V61	reggae.00035	5.648727	-1.029086	0
V60	pop.00096	28.654580	-1.033991	0
V57	pop.00070	0.328957	-1.042742	0
V24	country.00047	1.339772	-1.049825	0
V26	country.00051	3.512438	-1.134237	0
V15	classical.00024	5.951888	-1.148230	0
V28	country.00057	2.414768	-1.165795	0
V54	pop.00000	8.040866	-1.184524	0
V12	classical.00016	6.882160	-1.218064	0
V51	jazz.00099	5.859576	-1.218822	0
V58	pop.00088	6.043644	-1.250232	0
V17	classical.00034	3.302158	-1.291649	0
V21	classical.00041	9.705605	-1.305855	0
V59	pop.00091	3.458312	-1.312491	0
V14	classical.00021	6.735084	-1.318947	0
V53	metal.00078	5.241094	-1.337775	0
V22	country.00031	0.940582	-1.391734	0
V19	classical.00036	0.824725	-1.395206	0
V67	rock.00010	-0.542913	-1.408560	0
V74	rock.00047	-22.011629	-1.821236	0
V31	disco.00021	-0.894630	-2.295188	0
V65	reggae.00079	-1.718398	-2.379427	0
V18	classical.00035	0.327931	-2.438170	0
V56	pop.00069	-2.061430	-3.033730	0
V49	jazz.00087	-1.408842	-3.402118	0
V5	blues.00023	-0.240784	-10.593972	0

V71	rock.00033	-0.121095	-18.814481	0

Table C.3: Classifiers' abilities

Classifiers	Rasch	1PL	2PL	3PL	3PLg
KNN1[filtered]	-0.441138	-0.398306	-0.758941	-0.734745	-0.671524
KNN1[stratified]	0.855633	0.805407	0.216430	0.416692	0.621490
KNN2[filtered]	-0.607908	-0.553269	-0.738678	-0.401064	-0.351775
KNN2[stratified]	1.144205	1.072681	0.526050	0.623964	0.793409
KNN3[filtered]	-0.691565	-0.631001	-0.712286	-0.355949	-0.302098
KNN3[stratified]	0.950026	0.892864	0.731100	0.641851	0.787430
KNN4[filtered]	-0.441138	-0.398306	-0.631747	-0.297032	-0.199636
KNN4[stratified]	1.144205	1.072681	0.962853	0.769788	0.920569
KNN5[filtered]	-0.607908	-0.553269	-0.760556	-0.486128	-0.446767
KNN5[stratified]	-0.441138	-0.398306	-0.517796	-0.085778	-0.085183
KNN6[filtered]	-0.691565	-0.631001	-0.753774	-0.463688	-0.422473
KNN6[stratified]	-0.357855	-0.320922	-0.465280	-0.079132	-0.081423
SVC1[filtered]	0.145878	0.146989	-0.273780	0.121721	0.248118
SVC1[stratified]	1.451726	1.357167	1.743817	1.414298	1.662297
SVC2[filtered]	-0.023398	-0.010211	-0.150360	0.143010	0.278886
SVC2[stratified]	1.559415	1.456692	1.866801	1.199121	1.398670
SVC3[filtered]	-0.441138	-0.398306	-0.640537	-0.323094	-0.261516
SVC3[stratified]	1.244346	1.165362	1.425769	0.942587	1.117442
SVC4[filtered]	0.231358	0.226350	-0.172875	0.117705	0.251503
SVC4[stratified]	1.670104	1.558929	1.690759	1.452555	1.639266
SVC5[filtered]	-2.574783	-2.372647	-1.611749	-2.584961	-2.591248
SVC5[stratified]	-2.037459	-1.878802	-1.320951	-1.868461	-1.871058
SVC6[filtered]	0.060992	0.068165	-0.275490	0.123865	0.249666
SVC6[stratified]	1.244346	1.165362	1.637490	0.961875	1.141627
GaussProc1[filtered]	-0.191067	-0.165960	-0.488560	-0.284529	-0.187093
GaussProc1[stratified]	0.581347	0.551053	0.705704	0.612694	0.756648
GaussProc2[filtered]	-2.945445	-2.710225	-2.136511	-3.241178	-3.254705
GaussProc2[stratified]	-2.945445	-2.710225	-2.136511	-3.241178	-3.254705
GaussProc3[filtered]	0.145878	0.146989	-0.150399	0.157935	0.299389
GaussProc3[stratified]	1.901647	1.772556	2.445677	1.199937	1.400794
GaussProc4[filtered]	-0.107387	-0.088225	-0.807788	-0.669983	-0.600334
GaussProc4[stratified]	0.317536	0.306340	-1.000002	-1.651040	-1.647031
GaussProc5[filtered]	0.231358	0.226350	-0.189368	0.159638	0.302164
GaussProc5[stratified]	2.023165	1.884522	2.434308	1.407876	1.629040
GaussProc6[filtered]	-0.607908	-0.553269	-0.839150	-1.068056	-1.038399
GaussProc6[stratified]	-0.107387	-0.088225	-0.717423	-1.034732	-0.974707

DecTree1[filtered]	-1.468712	-1.352557	-1.000004	-1.359147	-1.344481
DecTree1[stratified]	-0.944657	-0.866119	-0.999999	-1.377434	-1.363969
DecTree2[filtered]	-1.651821	-1.522245	-1.000010	-1.319468	-1.303388
DecTree2[stratified]	-0.524461	-0.475729	-0.999991	-1.227373	-1.205855
DecTree3[filtered]	-1.290444	-1.187192	-1.000022	-1.341860	-1.327512
DecTree3[stratified]	-0.524461	-0.475729	-1.000003	-1.319109	-1.289390
DecTree4[filtered]	-1.468712	-1.352557	-1.000007	-1.603811	-1.584791
DecTree4[stratified]	-0.357855	-0.320922	-0.999993	-1.201655	-1.160116
DecTree5[filtered]	-0.775518	-0.709003	-0.999996	-1.112691	-1.093780
DecTree5[stratified]	-0.023398	-0.010211	-0.634056	-0.825635	-0.758724
DecTree6[filtered]	-1.379037	-1.269390	-1.000003	-1.324438	-1.306095
DecTree6[stratified]	-0.775518	-0.709003	-0.999482	-1.239293	-1.199952
RandForest1[filtered]	-0.357855	-0.320922	-0.619221	-0.384468	-0.312935
RandForest1[stratified]	0.950026	0.892864	1.088113	0.838299	1.005564
RandForest2[filtered]	-0.274527	-0.243499	-0.546550	-0.352188	-0.277433
RandForest2[stratified]	1.144205	1.072681	1.332554	0.892601	1.059742
RandForest3[filtered]	-0.357855	-0.320922	-0.615634	-0.352582	-0.278160
RandForest3[stratified]	0.950026	0.892864	1.038123	0.810274	0.974333
RandForest4[filtered]	-0.274527	-0.243499	-0.594748	-0.377910	-0.306200
RandForest4[stratified]	1.046160	0.981904	1.018831	0.856414	1.023607
RandForest5[filtered]	-0.274527	-0.243499	-0.539263	-0.334977	-0.257309
RandForest5[stratified]	1.451726	1.357167	1.383552	0.975629	1.162593
RandForest6[filtered]	-0.274527	-0.243499	-0.593568	-0.395128	-0.329160
RandForest6[stratified]	1.144205	1.072681	1.153579	0.911394	1.080272
MLP1[filtered]	-0.191067	-0.165960	-0.444294	-0.046242	-0.047948
MLP1[stratified]	1.346781	1.260127	1.113317	0.647662	0.807556
MLP2[filtered]	0.404518	0.387054	-0.018219	0.163396	0.327033
MLP2[stratified]	1.144205	1.072681	0.707128	0.686822	0.845188
MLP3[filtered]	-0.191067	-0.165960	-0.348461	0.132449	0.268906
MLP3[stratified]	1.144205	1.072681	1.130713	0.863058	1.019141
MLP4[filtered]	0.231358	0.226350	-0.095503	0.157307	0.309103
MLP4[stratified]	1.244346	1.165362	0.942135	0.623803	0.778435
MLP5[filtered]	-0.023398	-0.010211	-0.372203	0.125591	0.270131
MLP5[stratified]	0.762820	0.719386	0.204909	0.394508	0.579275
MLP6[filtered]	0.060992	0.068165	-0.230442	0.136948	0.276340
MLP6[stratified]	1.901647	1.772556	2.027169	1.198987	1.397740
Adaboost1[filtered]	-0.107387	-0.088225	-0.509195	-0.090990	-0.088647
Adaboost1[stratified]	0.762820	0.719386	0.530322	0.631605	0.787527
Adaboost2[filtered]	-1.030022	-0.945390	-0.814956	-0.978149	-0.929419

Adaboost2[stratified]	-0.775518	-0.709003	-0.799933	-1.021594	-0.991446
Adaboost3[filtered]	-0.191067	-0.165960	-0.563417	-0.222886	-0.218570
Adaboost3[stratified]	0.950026	0.892864	0.660191	0.533755	0.756340
Adaboost4[filtered]	-0.944657	-0.866119	-1.000001	-1.290267	-1.261044
Adaboost4[stratified]	-0.357855	-0.320922	-0.831764	-1.096645	-1.054197
Adaboost5[filtered]	-0.274527	-0.243499	-0.407123	-0.269964	-0.171725
Adaboost5[stratified]	0.671438	0.634548	0.544570	0.769933	0.978808
Adaboost6[filtered]	-0.944657	-0.866119	-0.998422	-1.050896	-1.022816
Adaboost6[stratified]	-0.607908	-0.553269	-0.699452	-0.601568	-0.555667
LogisticRegr1[filtered]	-0.023398	-0.010211	-0.200203	0.151344	0.287226
LogisticRegr1[stratified]	1.144205	1.072681	1.229844	0.782594	0.921083
LogisticRegr2[filtered]	-0.274527	-0.243499	-0.530146	-0.308869	-0.220047
LogisticRegr2[stratified]	0.671438	0.634548	0.668422	0.600608	0.741310
LogisticRegr3[filtered]	-0.607908	-0.553269	-0.755796	-0.539713	-0.476499
LogisticRegr3[stratified]	0.317536	0.306340	0.220598	0.594287	0.733244
LogisticRegr4[filtered]	-0.023398	-0.010211	-0.200203	0.151344	0.287226
LogisticRegr4[stratified]	1.144205	1.072681	1.229844	0.782594	0.921083
LogisticRegr5[filtered]	-0.357855	-0.320922	-0.607303	-0.358715	-0.280787
LogisticRegr5[stratified]	0.581347	0.551053	0.620706	0.595922	0.734679
LogisticRegr6[filtered]	-0.023398	-0.010211	-0.200203	0.151344	0.287226
LogisticRegr6[stratified]	1.144205	1.072681	1.229844	0.782594	0.921083
NaiveBayes[filtered]	-0.859852	-0.787349	-0.779368	-0.660092	-0.599676
NaiveBayes[stratified]	0.404518	0.387054	-0.132344	0.594424	0.736119
ExtraTree1[filtered]	-0.524461	-0.475729	-0.752499	-0.540827	-0.479812
ExtraTree1[stratified]	0.950026	0.892864	0.819243	0.727231	0.886892
ExtraTree2[filtered]	-0.441138	-0.398306	-0.634073	-0.369733	-0.265470
ExtraTree2[stratified]	1.451726	1.357167	1.531661	0.931467	1.105330
Dummy[filtered]	-2.693150	-2.480777	-1.495516	-2.445566	-2.436498
Dummy[stratified]	-2.693150	-2.480777	-1.584256	-2.556723	-2.541877
Voting(SVC)[filtered]	-0.107387	-0.088225	-0.256978	0.108850	0.229966
Voting(SVC)[stratified]	1.451726	1.357167	1.661643	1.157022	1.334087

C.0.0.1 Items characteristic curves by genres

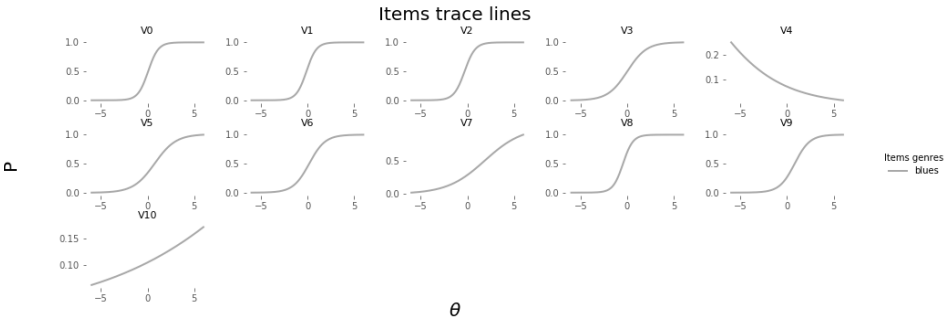

Figure C.1: Icc for test items with negative discriminant

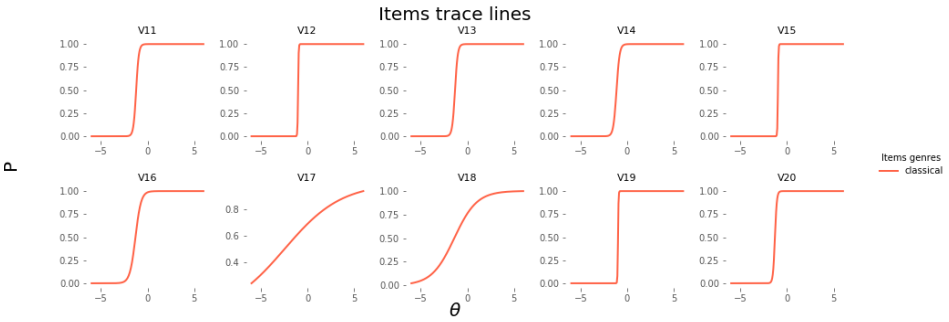

Figure C.2: Icc for test items with negative discriminant

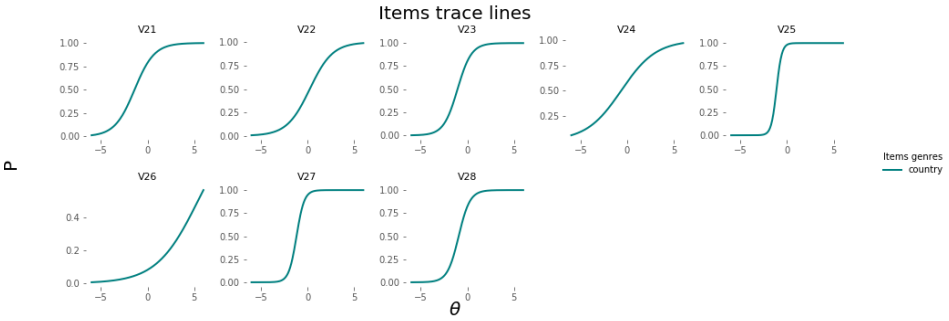

Figure C.3: Icc for test items with negative discriminant

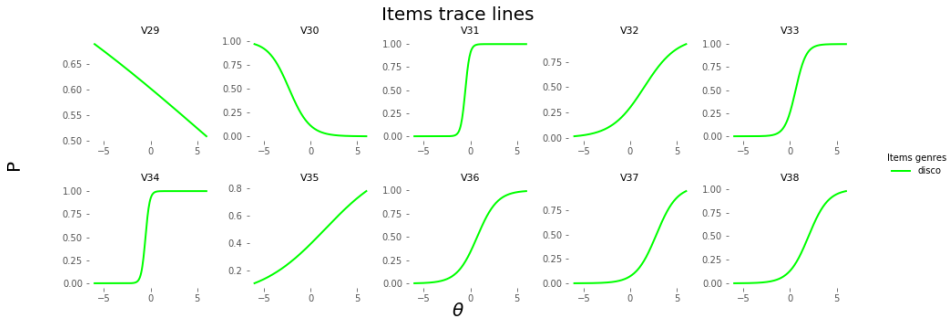

Figure C.4: Icc for test items with negative discriminant

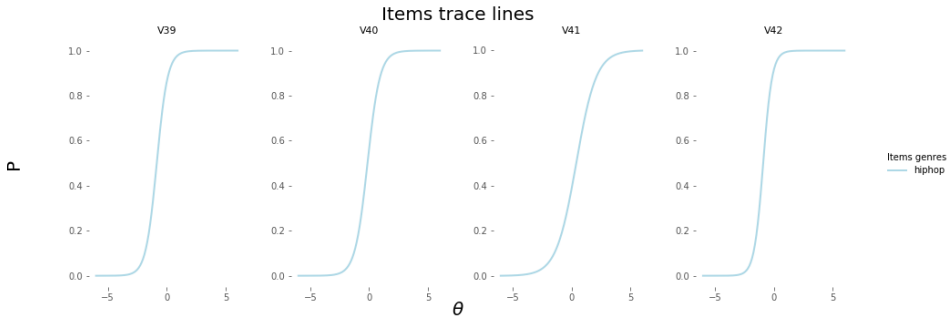

Figure C.5: Icc for test items with negative discriminant

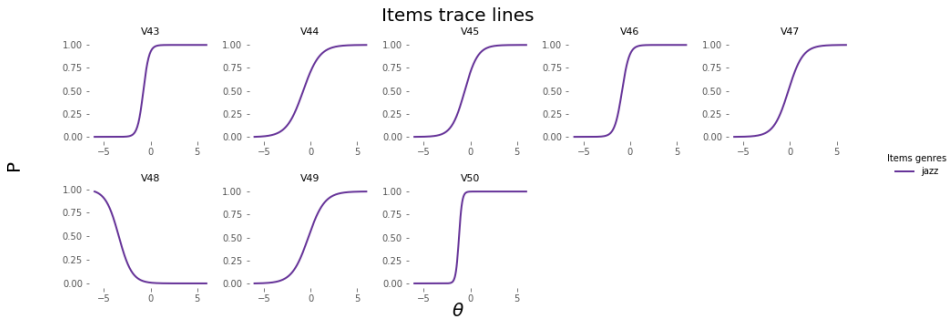

Figure C.6: Icc for test items with negative discriminant

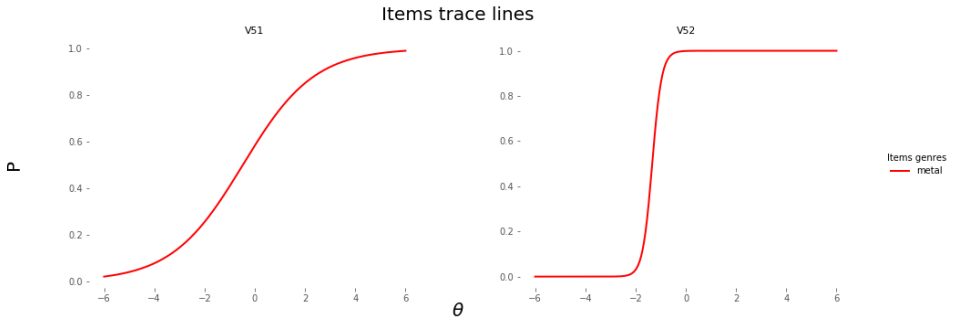

Figure C.7: Icc for test items with negative discriminant

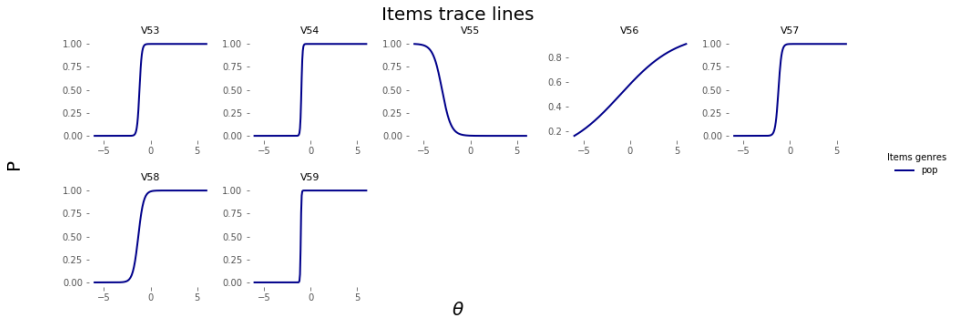

Figure C.8: Icc for test items with negative discriminant

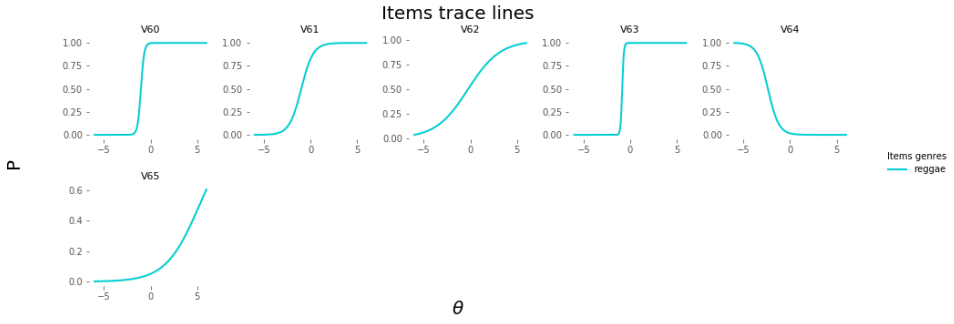

Figure C.9: Icc for test items with negative discriminant

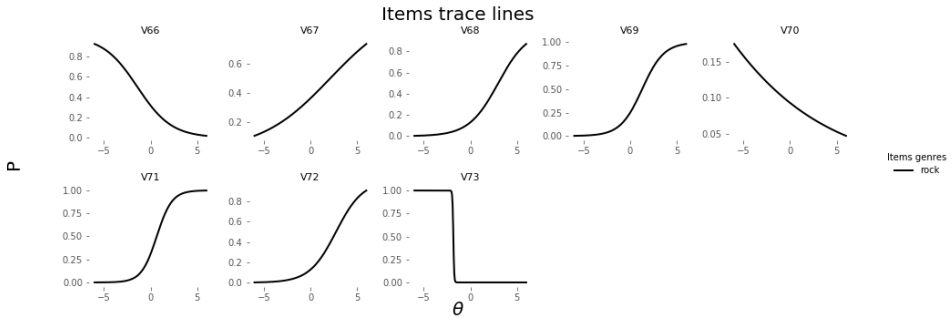

Figure C.10: Icc for test items with negative discriminant

C.0.0.2 Items information curves by genres

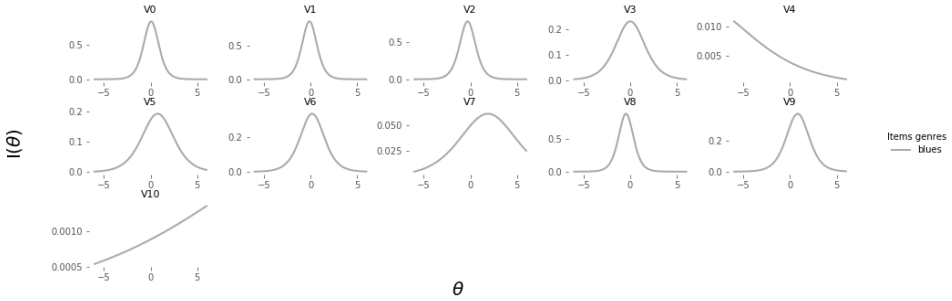

Figure C.11: Icc for all test items

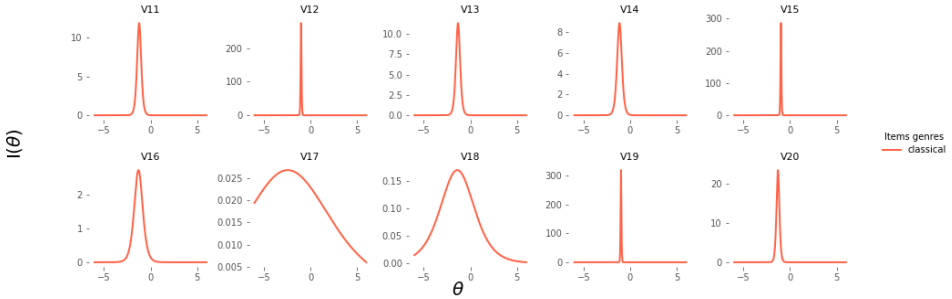

Figure C.12: Icc for all test items

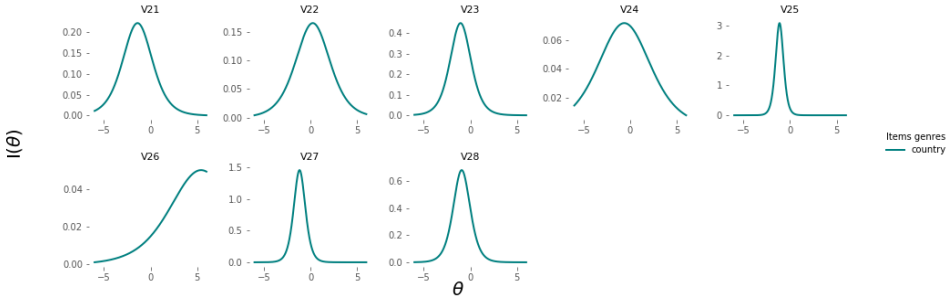

Figure C.13: Icc for all test items

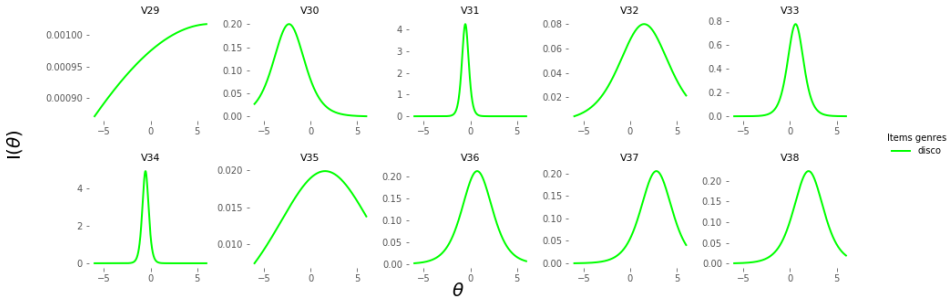

Figure C.14: Icc for all test items

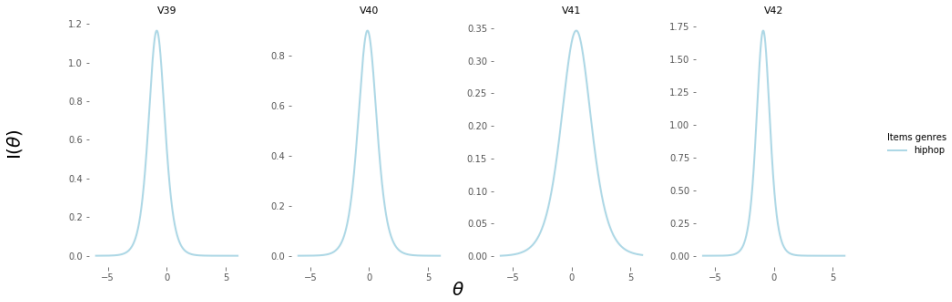

Figure C.15: Icc for all test items

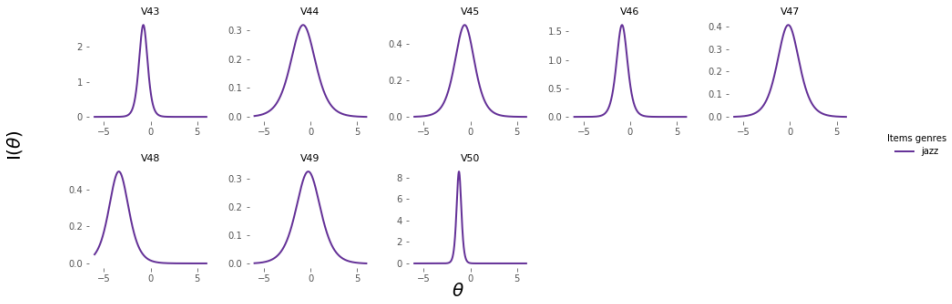

Figure C.16: Icc for all test items

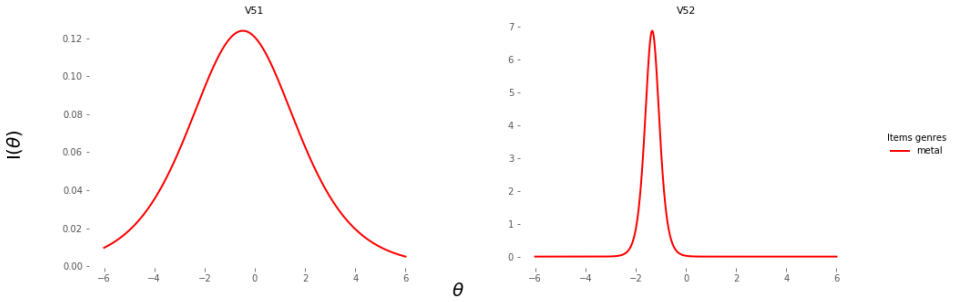

Figure C.17: Icc for all test items

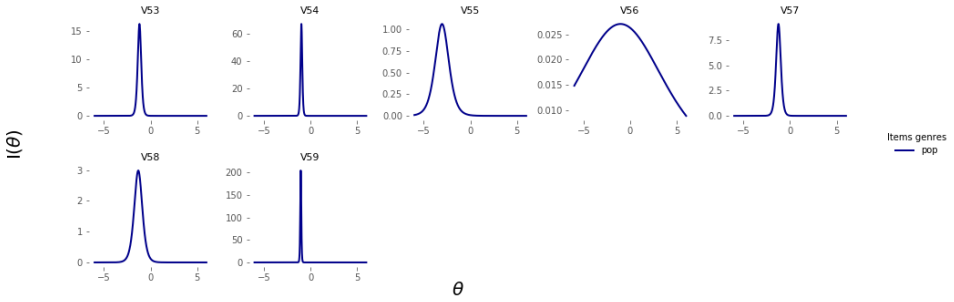

Figure C.18: Icc for all test items

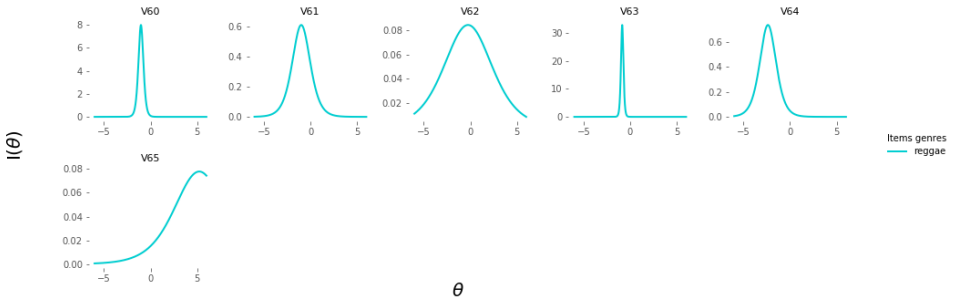

Figure C.19: Icc for all test items

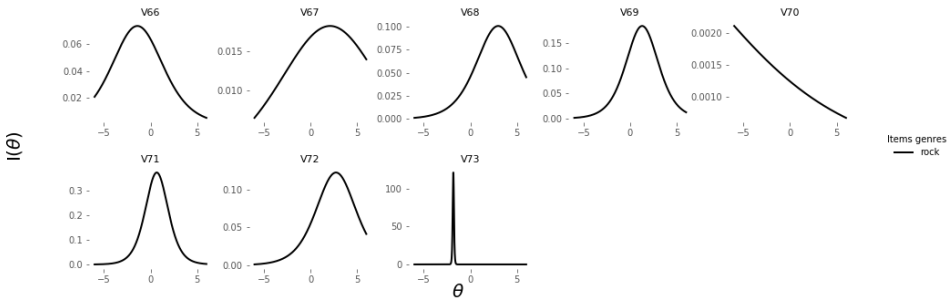

Figure C.20: Icc for all test items