Original Research Article

Educational and Psychological Measurement
2025, Vol. 85(3) 429–457

DOI: 10.1177/00131644241299834
journals.sagepub.com/home/epm
Sage

Item Classification by Difficulty Using Functional Principal Component Clustering and Neural Networks

James Zoucha[1], Igor Himelfarb[2] and Nai-En Tang[2]

Abstract

Maintaining consistent item difficulty across test forms is crucial for accurately and fairly classifying examinees into pass or fail categories. This article presents a practical procedure for classifying items based on difficulty levels using functional data analysis (FDA). Methodologically, we clustered item characteristic curves (ICCs) into difficulty groups by analyzing their functional principal components (FPCs) and then employed a neural network to predict difficulty for ICCs. Given the degree of similarity between many ICCs, categorizing items by difficulty can be challenging. The strength of this method lies in its ability to provide an empirical and consistent process for item classification, as opposed to relying solely on visual inspection. The findings reveal that most discrepancies between visual classification and FDA results differed by only one adjacent difficulty level. Approximately 67% of these discrepancies involved items in the medium to hard range being categorized into higher difficulty levels by FDA, while the remaining third involved *very easy* to *easy* items being classified into lower levels. The neural network, trained on these data, achieved an accuracy of 79.6%, with misclassifications also differing by only one adjacent difficulty level compared to FDA clustering. The method demonstrates an efficient and practical procedure for classifying test items, especially beneficial in testing programs where smaller volumes of examinees tested at various times throughout the year.

[1]University of Northern Colorado, Greeley, CO, USA
[2]National Board of Chiropractic Examiners, Greeley, CO, USA

Corresponding Author:
Igor Himelfarb, National Board of Chiropractic Examiners, 901 54th Avenue, Greeley, CO 80634, USA.
Email: ihimelfarb@nbce.org

Keywords

item classification, functional principal component clustering, neural networks

Introduction

Testing in the professions has a long history, originating in Imperial China where the selection of civil servants was based on standardized examinations (Zwick, 2002). Today, credential testing serves as a crucial source of evidence for demonstrating professional competencies and qualifications (Buckendahl, 2017). Assessments are also prevalent in educational settings as they serve to measure student learning and ability (Bennett, 2011; Suskie, 2018). A widely used method for estimating ability levels, and consequently determining assessment-based outcomes, is item response theory (IRT; Lord & Novick, 1968). Given that student advancement depends on the precision of these evaluations, their accuracy is of critical importance. To effectively determine mastery of a subject matter on a test, the exam must include a wide range of test items that represent various ability levels (Kline, 2015; Mislevy, 1992).

The region of greatest importance along the test continuum would be those areas closest to the passing thresholds (Muijtjens et al., 2003). Furthermore, to assure score comparability, structural constraints are established such that each test form produced has similar psychometric properties (Kolen & Brennan, 2014).

In addition, to ensure that a test remains both accurate and fair across different test-taking groups, one important factor is the consistency of its difficulty level (Embredson & Reise, 2000; Kolen & Brennan, 2014). In operational testing programs, maintaining difficulty consistency involves calibrating the items and classifying them across the difficulty spectrum so that the test has a balanced mix of easy, medium, and hard questions. If the ratio of these misclassified items fluctuates, it can skew the results, undermining the reliability of the test as an accurate measure of ability. Variability in classification accuracy across different item clusters may raise concerns about the validity and fairness of the test. In particular, if certain difficulty levels or item characteristics consistently lead to misclassification, the model may fail to provide a fair assessment of examinee performance across the ability spectrum. This inconsistency could compromise both the test's validity—its ability to measure what it is intended to measure—and its fairness, particularly if specific groups of examinees are more likely to encounter misclassified items, leading to biased outcomes.

Test Construction and Item Characteristic Curves

Item Characteristic Curves (ICCs; Lord, 1975; Rosenbaum, 1987) play a fundamental role in test construction, particularly within the framework of IRT. ICCs describe the relationship between an individual's estimated ability level and the probability of correctly answering a specific test item. Depending on a specific IRT model, each curve may be modeled using three key parameters: the difficulty of the item, the item's

discrimination, and the guessing factor (Hambleton et al., 1991). The difficulty parameter (the equivalent of an intercept) shifts the curve along the ability axis, indicating the level of ability at which the item has a 50% chance of being answered correctly. The discrimination parameter (an equivalent of slope) defines how steeply the probability increases as ability increases, with steeper curves indicating that the item is more effective at differentiating between test-takers of varying abilities. Finally, the guessing parameter accounts for the likelihood of low-ability individuals guessing the correct answer (Baker, 2001). Easier questions will tend to come earlier on the x-axes, whereas more difficult questions will have corresponding ICCs more to the right on the axes (Lord, 1977).

In the process of test construction, ICCs are employed to ensure that test items collectively provide appropriate measures across a wide range of ability levels. By examining the ICCs of individual items, test developers select or modify items to match the intended difficulty and discrimination characteristics of the test. For instance, items with high discrimination values are often preferred because they help better distinguish between test-takers with similar but slightly different ability levels.

At the National Board of Chiropractic Examiners (NBCE), characterization of items into one of the difficulty sub-groups is typically based on items' historical performance (item-level statistics) and visual inspection of the ICCs. Specifically, the items' difficulty subgroup is decided by their maximum information theta point (MIT; Birnbaum, 1968, p. 464). Items' MIT between –3 and –2 are assigned to the *very easy* subgroup; items' MIT between –2 and –1 are assigned to the *easy* subgroup; items' MIT between –1 and 0 are assigned to the *moderately easy* subgroup; items' MIT between 0 and 1 are assigned to the *moderately hard* subgroup; items' MIT between 1 and 2 are assigned to the *hard* subgroup; and items' MIT between 2 and 3 are assigned to the *very hard* subgroup.

Classifying test items into difficulty levels based on a single set of parameters, especially for the operational programs with a relatively small number of test-takers, may be problematic due to the issues related to statistical reliability and stability. The lack of ability-level diversity in small samples may pose another challenge. A limited number of test-takers may not represent the full range of abilities expected in the broader testing population. If the sample is skewed toward a particular ability level (e.g., mostly high-ability or low-ability test-takers), the difficulty estimates will be biased, potentially causing the test to be over-calibrated or under-calibrated for that specific group, rather than reflective of a more diverse, general population. This can impair the test's ability to assess all examinees fairly, ultimately reducing the test's effectiveness in measuring the intended construct. However, by reviewing ICCs, which are functions of all three item-level parameters revealing the probability of answering an item correctly across the entire ability spectrum, we are able to make more informed decisions about item behavior.

A similar approach was developed by Belov (2021) proposing the use of item difficulty modeling to predict statistical parameters of an item. The author proposed predicting a discrete ICC based on *softmax* classification. This method leverages the

one-to-one mapping between a monotonically non-decreasing ICC and a probability mass function (PMF). A neural network was trained using soft labels for each item, achieved by mapping the ICCs to PMFs.

A different study introduced penalized splines for estimating growth curves using data collected over time, explaining the flexibility of splines and how penalized spline models balance model fit and smoothness by incorporating a penalty term. The authors described piecewise linear models, higher-order splines, and the use of linear mixed-effects models for estimating penalized splines, addressing technical aspects such as hypothesis testing and confidence intervals (Suk et al., 2019).

Functional Data Analysis

Assuming continuity and smoothness in ICCs allows for application of functional data analysis (FDA; Kokoszka & Reimherr, 2017; Ramsay & Silverman, 2002) for the purpose of item classification based on difficulty estimates. FDA is particularly useful for analyzing data collected over continuous domains, such as time, space, or frequency, where observations are not isolated points but entire processes. FDA focuses on identifying patterns, trends, and relationships within these continuous datasets, often revealing insights that are not apparent in traditional data analysis (Wang et al., 2016). Through FDA, traditional scalar statistical methods such as linear models and principal component analysis can be generalized to data represented in infinite dimensional Hilbert space. In practice, a sample of curves is observed along a finite range of a domain where a semiparametric or nonparametric method is used to reconstruct a functional form from a set of discrete data points (Jacques & Preda, 2014). Here, densely collected repeated measures across a continuum such as time (t) are viewed as a single set (i) making up one observation $X_i(t)$ (Hall et al., 2006). Functional data are often represented using basis expansions, approximating functions through a linear combination of a suitable basis such as B-splines, Fourier, or wavelets (Ramsay & Silverman, 2005; Jacques & Preda, 2013). Partitioning functional data by basis expansion may reduce measurement error while maintaining functional structures (Abraham et al., 2003).

FDA offers two key advantages: it imposes fewer restrictive statistical assumptions and leverages richer information when drawing inferences. By focusing on continuous data structures such as curves or functions, FDA can model complex phenomena more flexibly and comprehensively, allowing for more accurate insights than traditional methods that rely on discrete data points and often stricter assumptions (Fortuna & Maturo, 2018; Ramsay & Silverman, 2005). Advances of modern computing facilitate this expansion of application of FDA to high-dimensional data (Hall et al., 2006; Shang, 2014). Functional principal components (FPCs; Benko et al., 2009) are essential tools in FDA, as they capture the primary modes of variation within functional data, serving a critical function in dimensionality reduction and feature extraction (Dai & Müller, 2018). The primary goal of functional principal component analysis (FPCA) is to provide an optimal representation of functional

data, making it one of the most widely used techniques for clustering and analyzing functional datasets.

Recently, Engelhard (2023) introduced a functional approach for modeling unfolding response data. In this study, FDA has been used for examining cumulative item response data. Seven decision parameters that can provide a guide to conducting FDA were described. These decision parameters were illustrated with real data using two scales that were designed to measure attitude toward capital punishment and attitude toward censorship. The analyses suggested that FDA offers a useful set of tools for examining unfolding response processes. Another research used functional classification to conduct morphological analysis of electrocardiographic (ECG) curves. The authors employed clustering techniques to group patients based on the shape of their ECGs, independent of clinical diagnosis. The study demonstrates that analyzing both the ECG curves and their first derivatives improves classification accuracy, and it proposes this clustering approach as a potential semi-automatic diagnostic tool for distinguishing between normal and pathological ECG patterns (Ieva et al., 2013).

Clustering

Clustering could be seen as an unsupervised learning process aiming to partition data into homogeneous sub-groups (Bolleddu, 2022). After clustering, observations within each cluster are similar to each other while being dissimilar to out-of-group clusters (Piernik & Morzy, 2021; Wu et al., 2021). Like centroids, principal components can be utilized to form distinct clusters of examples due to their orthogonality, which helps in separating the data into independent and non-overlapping groups (Hall & Hosseini-Nasab, 2006).

When applied to functional data, clustering algorithms are able to find representative curves corresponding to different modes of variation (Tarpey & Kinateder, 2003). Functional clustering methods can generally be classified into four main categories: raw data clustering, filtering methods, adaptive methods, and distance-based methods. Raw data clustering operates by directly grouping curves based on their observed values. In contrast, filtering methods first approximate the curves using basis functions or eigenfunctions, after which clustering is performed on the basis of expansion coefficients or principal component scores, as these scores reflect the degree to which observations align with each FPC. Adaptive methods employ probabilistic models to cluster the basis expansion coefficients, FPC scores, or the curves themselves. Finally, distance-based methods extend traditional multivariate clustering techniques to functional data by grouping curves according to a chosen distance metric (Jacques & Preda, 2014).

Clustering is often used as a preprocessing step to classification and is trusted as a mechanism for improving classification quality (Khan, Baseer, & Javed, 2017; Piernik & Morzy, 2021; Trivedi et al., 2015; Tsai et al., 2011). Piernik and Morzy (2021) examined this hypothesis by applying a clustering algorithm to segment training data into groups, subsequently utilizing this information in various classification

algorithms to predict test data labels. The classification algorithms employed, both linear and nonlinear, included methods such as penalized multinomial regression, Bayesian generalized linear models, and random forests, among others. Their findings revealed that certain combinations led to performance improvements with no cases of performance degradation overall.

Liao (2005) reviewed the application of clustering techniques to time series data across various domains. In discussing model-based clustering methods, the author highlighted neural network-based clustering as a prominent approach for future research and development. Furthermore, in searching for novel approaches to clustering, Marcoulides and Trinchera (2024) introduced an algorithmic method for detecting unobserved heterogeneity in longitudinal growth data. The authors suggested using natural cubic smoothing splines to cluster individuals based on their growth trajectories, avoiding restrictive assumptions often imposed by traditional models. The study highlights the method's utility in accurately capturing individual differences in growth trajectories without relying on predefined class structures, offering a valuable tool for behavioral and social science research.

Neural Networks

Neural networks are a class of machine learning (ML) models inspired by the structure and function of the human brain, designed to recognize patterns and make predictions based on data (Abdi et al., 1999; Gurney, 2018; Haykin, 1994). Comprising interconnected layers of artificial neurons, neural networks learn to map inputs to outputs through a process of adjusting weights based on errors in predictions (Müller et al., 2012).

In neural networks, the architecture typically consists of three main types of layers: input layers, hidden layers, and output layers, each serving a distinct purpose in the overall functioning of the model (Goodfellow et al., 2016). The input layer is the first layer in a neural network and is responsible for receiving the raw data that will be processed. Each neuron (or node) in the input layer corresponds to one feature of the input data. For example, in an image classification problem, the input layer might have one neuron for each pixel in the image. The input layer passes these data to the subsequent layers without performing any computations (LeCun et al., 2015).

Hidden layers sit between the input and output layers and are where most of the computation in a neural network occurs. These layers consist of neurons that apply transformations to the input data using a set of weights and biases, followed by an activation function that introduces non-linearity into the model. This non-linearity allows the network to learn and model more complex patterns in the data. A network can have one or more hidden layers, and deep neural networks typically have many such layers, enabling them to capture intricate relationships in the data (Goodfellow et al., 2016; Nielsen, 2015).

The output layer is the final layer of a neural network and produces the prediction or classification result. In a classification task, the output layer's neurons represent the possible classes, and their values correspond to the predicted probabilities for each class. For regression tasks, the output layer may contain one or more neurons representing the predicted continuous values. The type of activation function used in the output layer depends on the task—*softmax* is commonly used for multi-class classification, while linear functions are used for regression (Hastie et al., 2009).

Key developments, such as backpropagation and gradient descent, have enabled the training of large-scale neural networks, making them one of the most powerful tools in modern AI research (LeCun et al., 2015). As data move from the input layer through to the output layer, each neuron the data passes through applies a weighted sum of its inputs and an activation function to introduce non-linearity and produce an output. During model training, the final output is compared to true target values, and loss is computed through a loss function. Backpropagation calculates the gradient of the loss function with respect to the network's parameters, and gradient descent uses these gradients to update parameters with the goal of reducing loss and improving model performance. These neural networks have found widespread application across diverse fields, including image recognition, natural language processing, and predictive analytics, owing to their capacity to generalize from data and effectively model nonlinear relationships (Goodfellow et al., 2016).

Neural networks are powerful tools for classification tasks, capable of learning intricate decision boundaries and handling complex relationships in data. They offer advantages over other methods by automatically learning relevant features and capturing sophisticated patterns, eliminating the need for manual feature engineering (Swingler, 1996).

Current Study

The objective of this study was to introduce a robust and systematic approach for classifying test items into predetermined difficulty levels, ensuring the structural integrity and fairness of assessments used to categorize examinees into pass or fail categories. The study addresses the challenge of item categorization, particularly given the similarity in shape of ICCs. By applying FDA to group items according to their FPCs, the study aims to offer an empirical alternative to traditional, more subjective visualization-based methods of categorization. Furthermore, by basing decisions on item behavior across the entire ability spectrum, rather than solely on the parameters obtained from the 3-parameter logistic (3PL) model, the clustering algorithm gains a more comprehensive understanding of how each item functions. This approach allows the algorithm to consider the interplay between the discrimination, guessing, and difficulty parameters, providing a more detailed view of item performance. As a result, the algorithm is better equipped to identify and categorize items with similar overall behavior patterns, leading to more accurate clustering.

In addition, 3PL IRT models (explained in the next section of this article) require larger sample sizes for stable parameter estimation. One potential solution is to use a less-complex IRT model, such as the 2PL or Rasch models, but this comes at the cost of relying solely on the difficulty and discrimination parameters, or just the difficulty parameter in the case of the Rasch model. Our proposed approach offers an alternative that avoids this tradeoff.

Furthermore, the study assessed the effectiveness of this FDA-based clustering method by integrating a neural network trained to predict ICCs. The secondary objective was to evaluate the accuracy and reliability of the neural network in replicating the FDA-driven classification process. The results of the analysis will provide insights into how closely the neural network aligns with the FDA method, offering a practical solution for large-scale testing programs, particularly in today's contexts when the rapid and accurate assessment is essential for timely and defensible results.

Method

In this study, we outlined the functional classification process which consists of three main parts. A clustering method based on FPCs, which was applied on the set of ICCs to identify distinctive patterns among the curves (Fortuna & Maturo, 2018). Clustering provided an effective approach for distinguishing our sample functions, allowing for empirical classification even when the ICCs exhibited similar shapes or locations. Next, clustered groups were given a difficulty level labeled matching to those established by the operational testing program. Once established, a functional regression was fit to evaluate the accuracy of the functional representation of the ICCs and how well the assigned categorical membership through clustering represented the data.

Data

The data for this study were drawn from the three most recent operational administrations of the NBCE basic science (Part I) exam (National Board of Chiropractic Examiners [NBCE], 2024). The dataset included 240 operational, multiple-choice items, covering a range of topics across various clinical areas.

Discretizing ICCs and Finding Mean Curves of Original Groups. A 3PL IRT model was fitted to the data and used to estimate the discrimination (a), difficulty (b), and guessing (c) parameter values for each item in the dataset. A sequence of length 1,000 for ability (θ) was produced to define ability levels between -3 to 3. The 3PL function is denoted as follows:

$$P(X_i|\theta_i) = c_j + (1 - c_j)\frac{1}{1 + e^{-a_j(\theta_i - b_j)}}$$

Inputting all values into the function, discretized ICC(θ)s were produced for each item and saved. Each ICC was a sequence of realized probabilities of getting an item correctly conditioned on values of θ along the ability domain. All 240 ICCs had been assigned difficulty labels, which were determined based on the item parameters, visual inspection of the curves, and discussions among the test development team. There were six coded item difficulty levels in total: *very easy*, *easy*, *moderately easy*, *moderately hard*, *hard*, and *very hard*. The mean curves for each visually determined group were established and later compared to the groups generated through functional clustering.

Creating a Functional Data Object. Functions from the *fda* package in *R* (Febrero-Bande & De La Fuente, 2012) were used to facilitate the FPCA and regression. To use these functions, the matrix of ICCs needed to be converted into functional data objects through the basis expansion and smoothing techniques. Basis expansion allows for representing the discretized functional data matrix in the form of a linear combination,

$$\mathrm{Y}(\theta) = \sum_{v=1}^{V} \varphi_v \gamma_v(\theta)$$

where $\gamma_v(\theta)$ are the v basis functions of θ and have corresponding coefficients φ_v that project the discretized curves into Hilbert space, representing them in a reduced form that best maintains information of the true curves. For scalar data, basis functions are fixed, whereas basis functions used for functional data vary across the entire domain. The possible number of basis functions that can be used to represent a set of functional observations ranges from 1 to U where U = 1,000, or the length of each ICC in this study. Choosing a number which is too small may result in a loss of information while picking V = U could lead to the inclusion of noise.

As a possible remedy or counterbalance, a smoothing parameter λ was applied. In FDA, smoothing often involves adding a penalty term to the least square minimization problem used to estimate the coefficients φ_k:

$$\min_{\varphi_v} \left[\int_{-3}^{3} (\mathrm{Y}(\theta) - \sum_{v=1}^{V} \varphi_v \gamma_v(\theta))^2 d\theta + \lambda \Sigma_{v=1}^{V} \left(\int_{-3}^{3} \left(\gamma_v''(\theta) \right)^2 d\theta \right) \right]$$

The first part of the equation calculated the sum of squared residual evaluated across the domain of θ. Given these are functional observations, integrating results in a scalar metric can be useful for representing the difference between observed and fitted curves. In the second half of the equation, $\gamma_v''(\theta)$ represents the second derivative of the v basis functions evaluated over θ. This penalty term sums up the squared curvature across the domain to give a single scalar value representing the overall roughness of the function. Overall, the equation above represents the first step of a two-part optimization problem. First, for a given λ, coefficients φ_v must be found such that

they minimize the sum of squared residuals plus the penalty term. Second, the results of each minimization facilitated by a set of lambdas are compared by calculating the generalized cross-validation measure (GCV). The GCV, developed by Craven and Wahba (1978), was applied through the *lambda2gvc()* function in the *fda* package. The main benefit of GCV over traditional cross-validation is when applied to complex models or large datasets, it reduces computational burdens by computing its metric through a single pass. A single pass refers to fitting a model without leaving out any subsets of data. The GCV score of the fitted model corresponding to λ is calculated using the following formula:

$$GCV(\lambda) = \frac{n}{(n - df(\lambda))^2} SSE(\lambda)$$

In the formula, n is the number of observations, $df(\lambda)$ is the effective degrees of freedom depending on λ, and $SSE(\lambda)$ is the sum of squares error (*SSE*) for the model fit with λ. Degrees of freedom are the number of independent pieces of information that go into the estimation of parameters. A larger $df(\lambda)$ indicates a more complex model while smaller values indicate simpler models. By including effective degrees of freedom, GCV penalizes more complex models to help prevent overfitting. The value of λ that minimizes the GCV measure is selected as the optimal smoothing parameter and applied to create the smoothed functional data object.

Finding FPCAs and Applying Fuzzy Clustering. Upon smoothing the functional ICC object, six FPCs were extracted. These FPCs were then used in fuzzy clustering to categorize ICCs into one of six groups by following the filtering functional classification method. The choice of six FPCs aligns with the number of difficulty levels previously established by the testing program (difficulty levels). Fuzzy clustering was chosen over deterministic algorithms as it clusters based on degrees of membership, rather than categorizing with absoluteness. Thus, each ICC has a membership value associated with an FPC-based cluster, indicating the degree of belongingness to each cluster. Given that adjacent difficulty levels often have operational overlap, fuzzy clustering fits the context of these data (Srinivsa et al., 2005). The probabilistic assignment introduces a degree of variation, serving as a proxy for the influence of an item's classification history across different administrations on its categorization. While an item might perform unusually in a single administration due to external factors, it is rare for a test construction team to drastically change an item's difficulty level if its previous designation was consistent. Therefore, incorporating a controlled level of variation through probabilistic clustering simulates the influence of potential factors that could engender class movement among items. This was facilitated through the *Fclust()* function in R's *fclust* package (Ferraro et al., 2019).

Comparing Group Assignments. After identifying the six clusters and assigning each ICC to one of them, the clusters can then be labeled according to their respective difficulty levels. The designation of a difficulty was decided by comparing the shape of

the mean curves produced by grouping through the original visualization method versus mean curves of the ICCs clustered according to the FPCs. For example, if the mean curve of cluster 1 best resembled the mean curve of the previously labeled *very easy* items, cluster 1 was designated as the *very easy* item cluster for the rest of the analysis. This process was repeated until all clusters were labeled. Discrepancy ratios were then calculated to assess the level of agreement between the two methods on the difficulty of an item based on its ICC.

Fitting a Functional Regression. Next, a functional regression was fit. With a functional regression estimated, the *SSE* can be reviewed, and a permutation *F*-test, along with post hoc *t*-tests, was applied. The *SSE* served as an indicator of the information loss incurred when fitting a model to the observed ICCs. The permutation *F*-test illustrated whether and where statistical differences existed between mean curves representing the clustered difficulty groups of ICCs.

The *F*-observed function and the two critical *F*-functions were plotted together. Significant differences in functional behavior among the six ICC cluster groups were observed at points where the *F*-observed values exceeded the critical *F*-thresholds. This comparison between difficulty groups assesses the impact of cluster group membership and identifies the intervals where the shape of ICCs within the same cluster differs from those in other clusters. If *F*-observed function falls below the critical region at any point, then there is no statistically discernible difference between the functional form of each difficulty group at said point. If significance was found for the permutation *F*-test, post hoc *t*-tests were performed for each pairwise cluster group comparison. For these permutation *t*-tests, the alpha level was adjusted using the Bonferroni method as such:

$$\alpha_z = \alpha / p$$

(Bonferroni, 1936).

Through basis expansion, functional data can be represented such that traditional statistical models can be realized as following:

$$Y_r(\theta) = \beta(\theta) + \sum_{s=1}^{S} X_{r,s}(\theta)\beta_s(\theta) + \varepsilon_r(\theta)$$

What should be noted is the equation above matches that of a multiple linear regression with the addition of (θ) indicating the effect of β_s changes at different points of θ. Here, *Yr(θ)* represents the r^{th} ICC the fitted model is predicting. The intercept $\beta_0(\theta)$ is the baseline functional form for all ICCs. The design matrix $X_{r,s}$ holds the values 0 or 1 at the (*r, s*) indices. The binary indicator determines the group membership of each ICC by controlling which functional coefficients are included in the model when predicting an ICC. If $X_{r,s} = 1$, the s^{th} functional coefficient $\beta_s(\theta)$ is included in the model for the r^{th} ICC, otherwise for $X_{r,s} = 0$, the corresponding $\beta_s(\theta)$ is not included. The functional coefficients $\beta_s(\theta)$ represent the effect of each group on the ICC. Unlike scalar regression, where the effects are fixed and changes to the

predicted outcome are linear, in functional regression, the coefficients themselves are functions. Consequently, the effect of an item with a specified difficulty—represented on the ICC by the probability of answering the item correctly—changes, depending on the value of θ. The ε_r is the r^{th} error term corresponding to the r^{th} curve.

Grid Search. This process requires the selection of hyperparameter values, which are the number of basis functions used to represent the ICCs and the β functions denoting the effect of cluster membership, as well as the smoothing parameter λ. To determine the best value of hyperparameters for representation and model fitting, a grid search was conducted. This grid search explored a range of initial values for the basis functions and lambda, iterating through each combination to identify the one that yielded the functional linear model with the lowest *SSE*. B-spline basis functions were used for basis expansion, and values in the grid search ranged from 50 to 800 by 50. For λ, a sequence of values from 0 to 10 by 1 were tested in conjunction with the choices for number of basis functions. Restricting the hyperparameter values within these trial ranges is considered reasonable, as they align with the current understanding of the ICCs. Given the inherent smoothness of these functions, the curves may be adequately represented with fewer basis functions, thereby reducing computational costs. Furthermore, relatively low values of λ were tested despite the fact that λ theoretically has no upper bound. Values closer to 1 prioritize a closer fit to the data with minimal smoothing. The *ICC-*β*-*λ combination that resulted in the lowest *SSE* was subsequently utilized to carry out the aforementioned steps: representing the ICCs as functional data objects, extracting FPCs, and fitting the final functional regression model.

Training and Applying a Neural Network Classifier. The final step was feeding a subset of about 80% ICCs and their difficulty cluster classification values into a neural network for training. The ratio of difficulties was maintained as best as possible in the training process such that the neural network receives a similar proportion of *very easy* to *very difficult* questions as was present on the original exam. During model training, the learning process was monitored by visualizing changes in the categorical cross-entropy loss function, commonly used in multi-class classification tasks where labels are one-hot encoded, as is the case in this study, and was employed alongside accuracy metrics. The equation for the categorical cross-entropy loss function is as follows:

$$L = -\sum_{r=1}^{R}\sum_{c=1}^{C} y_{r,c}\log(\widehat{y_{r,c}})$$

The total loss L is computed as a summation over all items r and all categories c. The values of $y_{r,c}$ are binary, taking the value of 0 or 1 to indicate whether the true class label for item r corresponds to category c, while $\widehat{y_{r,c}}$ represents the predicted probability that item r belongs to category c. For each item, only the logarithm of the

predicted probability corresponding to the true category of that item contributes to the total loss. The loss for a single item can be calculated by taking the dot product of two vectors, as follows:

$$\textit{If item } r \textit{ is in cluster } 1 \rightarrow y_r = [\,1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0\,]$$

$$\textit{Hypothetical predicted probabilities} \rightarrow \hat{y}_r = [\,.5 \quad .1 \quad .1 \quad .1 \quad .1 \quad .1\,]$$

$$L = -(1 * \log(.5)) + (0 * \log(.1)) + \ldots + (0 * \log(.1)) = -\log(.5)$$

The total loss is an extended formulation, summing over all items and iterating through the specified number of epochs. This process allows for the identification of potential learning issues, such as stagnation at local minima or maxima.

The neural network employed in this study consisted of four layers: an input layer, two hidden layers with 256 and 128 neurons, respectively, and an output layer. Given the relatively small dataset and the smooth nature of the ICCs, a highly dense neural network was deemed unnecessary. The decreasing number of neurons from the first to the second hidden layer reflects the belief that the most complex structures of the curves could be learned efficiently. Consequently, using a more complex neural network was not considered a worthwhile investment of limited computational resources. The ReLU activation function was applied to both hidden layers, while the *softmax* activation function was used in the output layer to constrain cluster assignment probabilities between 0 and 1.

To mitigate potential overfitting, dropout regularization was applied to the hidden layers during training. This technique randomly deactivates subsets of neurons during each training iteration, preventing the network from becoming overly reliant on specific neuron groups. The trained network was subsequently tested on the remaining 20% of the dataset without dropout, and accuracy metrics were evaluated to assess the network's effectiveness in classifying ICCs into the predefined functional cluster groups.

Results

Figure 1 presents the ICCs, while Figure 2 displays the mean curves for each difficulty group categorized through visualization. As anticipated, the mean curves for each difficulty group followed the expected patterns: for the easiest items, test-takers with lower ability demonstrated a relatively high probability of answering correctly, whereas for the harder questions, probabilities did not begin to increase until higher ability levels were attained.

Upon running the grid search loop, the triplet producing the lowest *SSE* included 800 b-spline basis functions for representing the ICCs, 100 b-spline basis functions for representing the beta functions, and $\lambda = 1$. By tuning these specific hyperparameter values to best represent the data, we ensured that the basis-expanded form of the functions was optimized to preserve the maximum amount of information compared to all other triplet combinations tested, as well as any default values automatically assigned by the R functions used during the creation of functional objects.

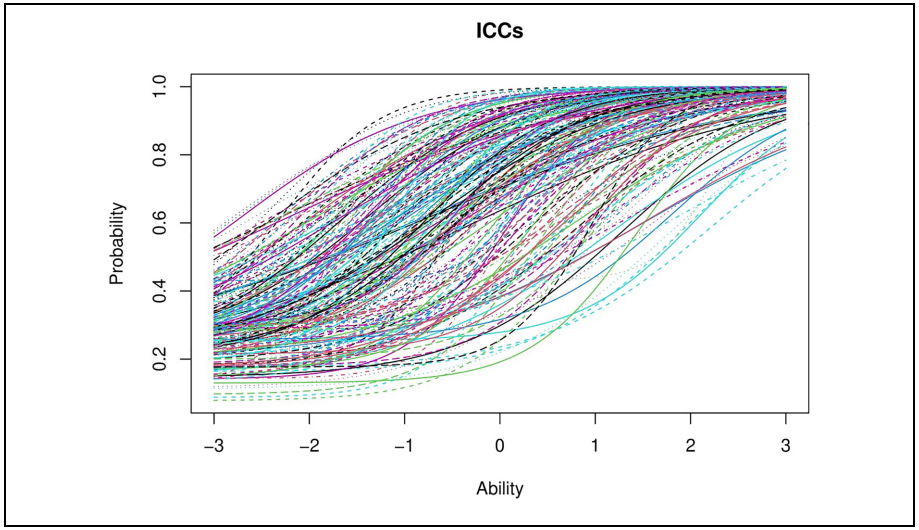

Figure 1. Two hundred forty Item Characteristic Curves.

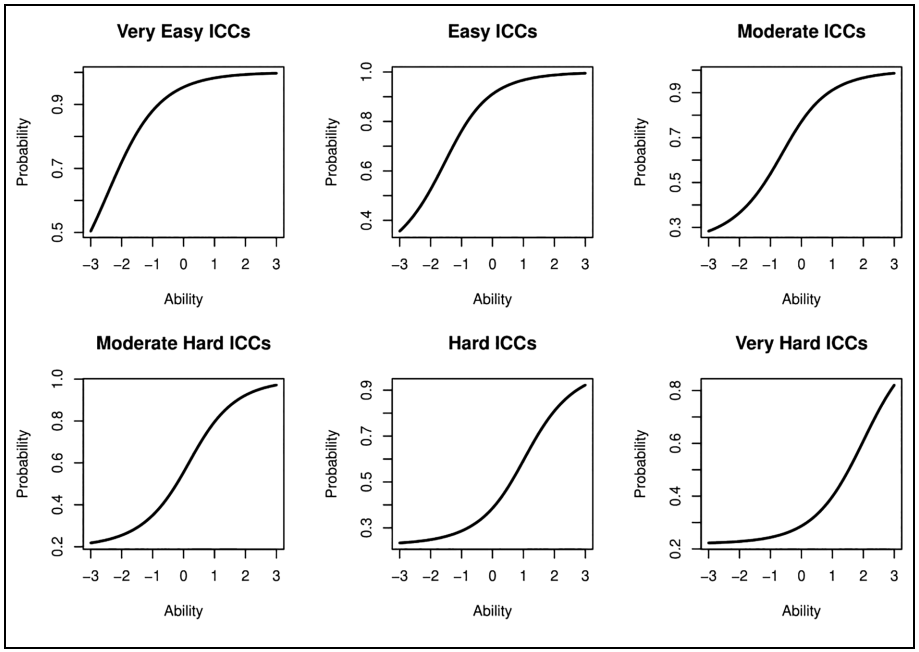

Figure 2. Mean ICCs for Each Difficulty Based on Initial Classification Through Visualization.

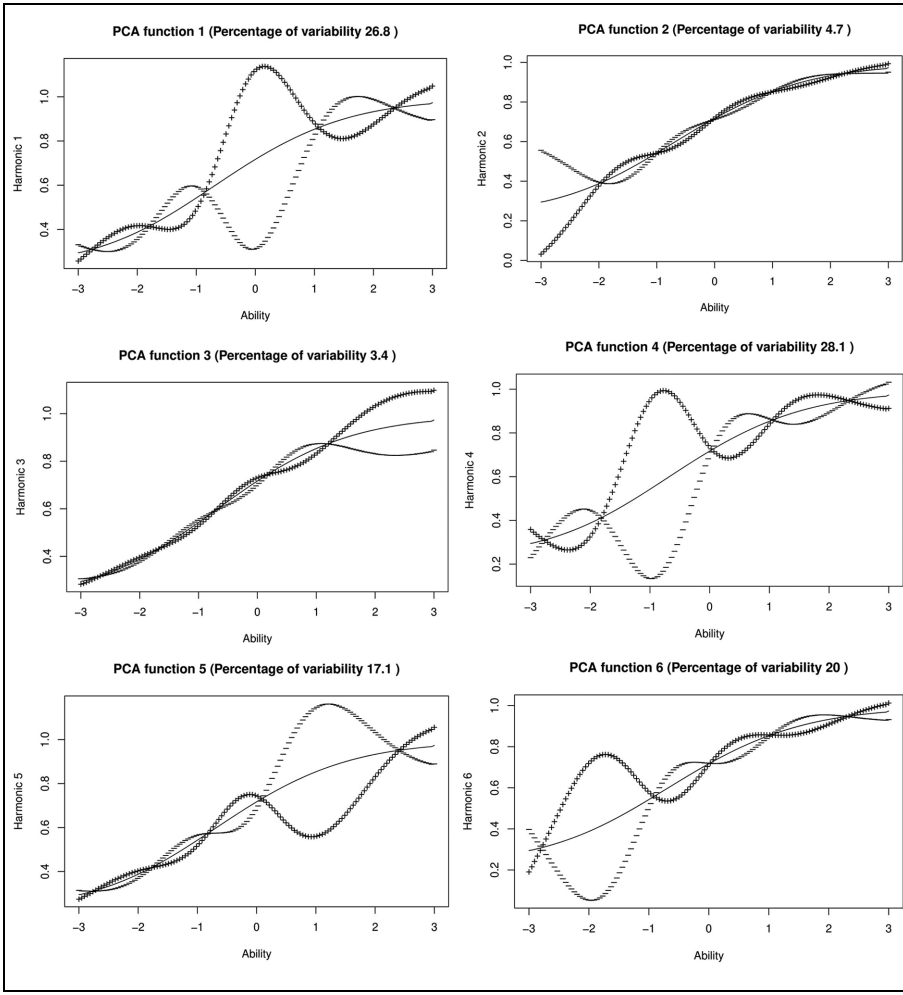

Figure 3. The Solid Center Lines Represent the Eigenfunction Corresponding to the 6th Rotated FPC.

Note. The line of "–" and line "+" indicate the eigenfunction's contribution to the variability of the data at specified points/intervals.

FPCA was then applied to the smoothed ICCs, $Y_r(\theta)$, where six FPCs were found.

Individual plots for each rotated FPC were generated, as shown in Figure 3. The solid line represents one eigenfunction, while the lines made of "+" or "—" oscillating around the solid line represent the variability around that eigenfunction at each point along the x-axis θ. On the y-axis, a large oscillation over a specific interval of θ indicates that the corresponding eigenfunction has a greater contribution to the

sample of ICCs within that interval. If the oscillation is small, the corresponding eigenfunction contributes less to the variability at a specific point or interval. The fourth FPC had the most variation which accounted for at 28.1%, with the largest contribution being from $\theta = [-2, 0]$, while the third FPC had the least variation at 3.4%.

Once the FPCs were identified, the corresponding FPC scores for all ICCs were utilized for fuzzy clustering. After clustering the ICCs and assigning membership categorizations, centroids for each cluster were determined by calculating six mean functions for each cluster group. Figure 4 displays the plotted centroids. These centroids were then used to label the difficulty levels of each cluster by comparing their shapes to one another and to the mean curves from the initial groupings created through visualization. Upon review, it was determined that Cluster 4 best represents the *very easy* items, Cluster 2 corresponds to *easy* items, Cluster 3 represents *moderate* items, Cluster 5 includes *moderately hard* items, Cluster 6 represents *hard* items, and Cluster 1 corresponds to *very hard* items. Of the 240 ICCs, 129 difficulty classifications were consistent between the visual categorization and the clustering-based categorization, as shown in Table 1.

All but three items with differing classifications had a discrepancy of ± 1 adjacent difficulty level. The clustering method typically classified items that were visually categorized as *moderate* to *hard* into one adjacent higher difficulty level. Additional discrepancies were found between the *very easy* and *easy* categories, where the clustering method typically placed items into a lower difficulty class than originally assigned. These discrepancies are detailed in Table 2, which lists the total counts for each difficulty level based on both the visualization-based categorization and the clustering-based categorization. Items 122, 147, and 231, as shown in Figure 5, were notable outliers, each exhibiting a classification discrepancy of two difficulty levels. Item 122, initially categorized as *moderately hard*, was reclassified by the clustering method into the *very hard* group. Item 147, originally coded as *moderate*, was reassigned to the *hard* category by clustering. Similarly, Item 231, also categorized as *moderately hard*, was reclassified by clustering as a *very hard* item.

Given the new group assignments, six β functions were created for fitting the final regression model:

$$Y_r(\theta) = \beta_0(\theta) + \sum_{s=1}^{6} X_{r,s}\beta_s(\theta) + \varepsilon_r(\theta)$$

The lowest *SSE* from the final regression model was 649.22. The plot in Figure 6 shows that the calculated F statistic consistently exceeded both the pointwise and maximal F critical values across the entire range of θ. This indicates a statistically significant overall difference in the functional forms of the clustered ICC groups throughout the range of θ. Following this, pairwise permutation t-tests were conducted to identify which groups contributed to the differences observed in the permutation F-test. Acknowledging that multiple comparisons increase the Type I error

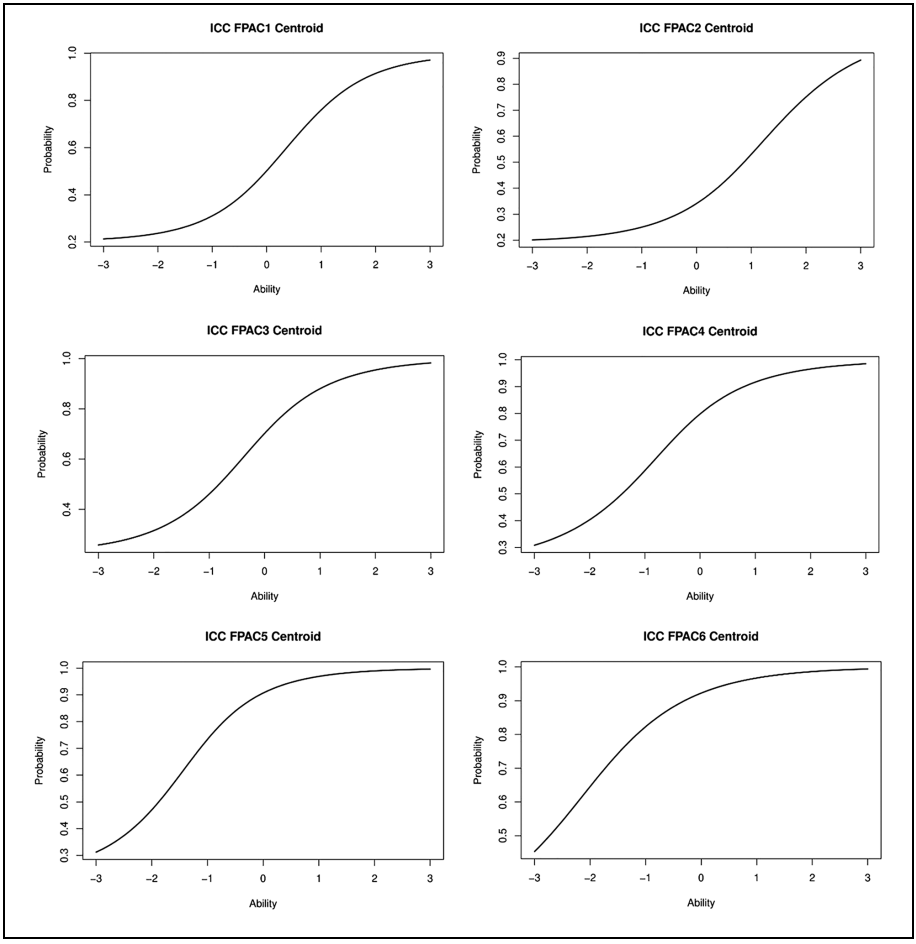

Figure 4. Mean Curves of Each Group of ICCs Clustered Together Labeled as Centroids.
Note. Centroids are ordered in by ascending difficulty once labels are given from the top left down to the bottom right. Centroid 4—very easy, Centroid 2—easy, Centroid 3—moderate, Centroid 5—moderately hard, Centroid 6—hard, Centroid 1—very hard.

Table 1. The Number of Times FPC Clustering Placed Items in Lower or Higher Difficulties Compared to Their Original Categorization by Visualization, as well as the Ratio of Total Agreement Between the Two Methods.

Total items categorized in lower difficulties by FPC clustering	Total items categorized in higher difficulties by FPC clustering	Total items with matching categorizations between methods
30	81	129/240 (53.75%)

Table 2. The Comparison Between the Visualization and FPC Clustering Based on Items' ICCs.

Difficulty level	Categorization by visualization	Categorization by FPC clustering
Very easy	12	38
Easy	66	43
Moderately easy	84	49
Moderately hard	42	46
Hard	31	36
Very hard	5	28

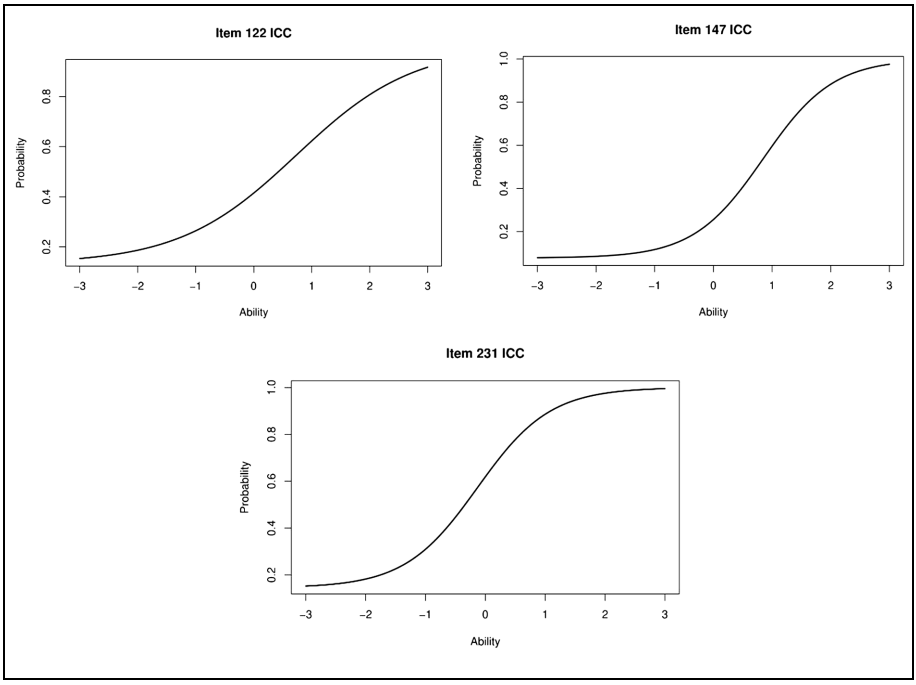

Figure 5. ICCs for Items That Were Clustered Into Difficulty Classes More Than One Adjacent Class Away From Their Initial Classification Through Visualization.

rate, the alpha level was adjusted using the Bonferroni correction, resulting in an adjusted $\alpha_z = .003$, as the original α was .05, and a total of 15 pairwise comparisons were made between adjacent and non-adjacent difficulty clusters.

Figure 7 shows all pairwise comparisons for adjacent difficulty classes found through clustering. That is, the *very easy* cluster was compared to the *easy* cluster, *easy* was compared to *moderate*, *moderate* was compared to *moderately hard*,

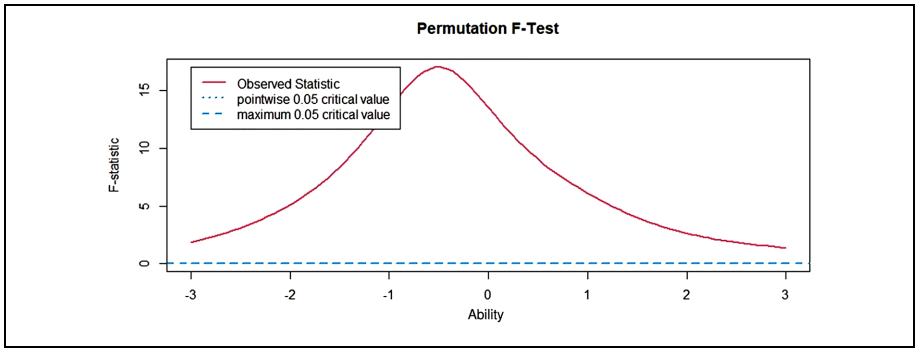

Figure 6. Permutation F-Test.

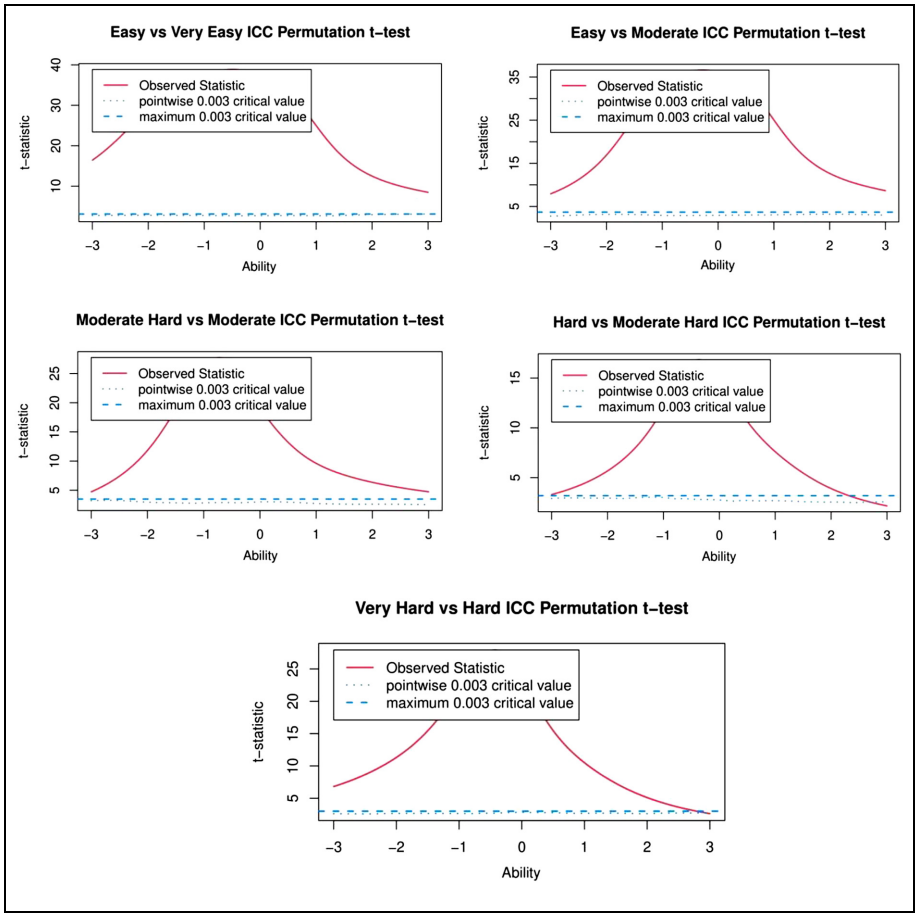

Figure 7. Adjacent Cluster Group Permutation *t*-Test Comparisons.

moderately hard was compared to *hard*, and *hard* was compared to *very hard*. For the *very easy* and *easy* clusters, there was a statistically significant difference between the mean curves of the cluster ranging from about $\theta = [-3, -0.5]$, with the largest degree of difference being around $\theta = -3$. This suggests that the primary distinction between *very easy* and *easy* items lies in the first half of their ICC curves. The absence of significant differences over much of the interval between these adjacent categories may explain why Table 2 shows more ICCs being clustered into the *very easy* category than the original visualization-based classification. Items initially categorized as *easy* by visualization were reclassified as *very easy* through the clustering process. When comparing *easy* and *moderate* ICCs, a significant difference was found around $\theta = [-1.9, 3]$, indicating that these two clusters can be distinguished across nearly the entire curve, except for the very beginning, with the largest difference occurring around $\theta = -1.15$.

In contrast, *moderate* and *moderately hard* ICCs show significant differences over the interval $\theta = [-3, 1]$, meaning these curves can be differentiated across most of the curve, except where $\theta > 1$, with the greatest divergence occurring around $\theta = -1.25$. Comparing *moderately hard* and *hard* ICCs, the two clusters can be distinguished across all but the tail end at $\theta = [2.25, 3]$, with the largest difference near $\theta = -0.2$. Finally, for the *hard* and *very hard* ICCs, significant differences appear from $\theta = [-0.8, 3]$, suggesting that these groups can be differentiated in the latter two-thirds of their curves, with the largest difference at $\theta = 0.9$. Since ICCs ranging from *moderate* to *hard* were frequently classified one level higher when the clustering method was applied, the non-significant ranges in the latter comparisons likely contributed to the observed shift in Table 2, where fewer items were classified as *moderate* and more as *very hard* compared to the initial visualization-based categorization.

For the non-adjacent comparisons seen in Figure 8, almost all the pairwise permutation *t*-tests illustrated significance across the entire range of θ, aside from *moderate* versus *hard* and *very easy* versus *moderate* comparisons showing non-significance right before $\theta = -3$. The point at which the greatest difference between the curves of the clustered groups varies based on which specific non-adjacent comparison is being reviewed, but all seem to agree that this ranges from about $\theta = [-1, 0]$. Referring back to the items clustered into a difficulty group more than one adjacent class away, it may be that the smaller observed F-values near the end range of θ lead to such a result. Altogether, these comparisons do indicate the differences in functional form between adjacent ICC classes can be more difficult to discern in some cases. All aforementioned depictions also illustrate where the clustering method most likely made their decisions regarding ICC assignment. Overall, the clustering method resulted in counts of each cluster being much closer to being even. Although the general trend remained where *easy* to *moderate hard* questions made up the largest proportion of the entire set, *very hard* items had the lowest total.

Given the clustering classifications, the neural network was trained on a subset of the ICCs and their new difficulty codes. Figure 9 shows a plot tracking the loss and accuracy for both the training (blue) and validation (green) sets through each of the

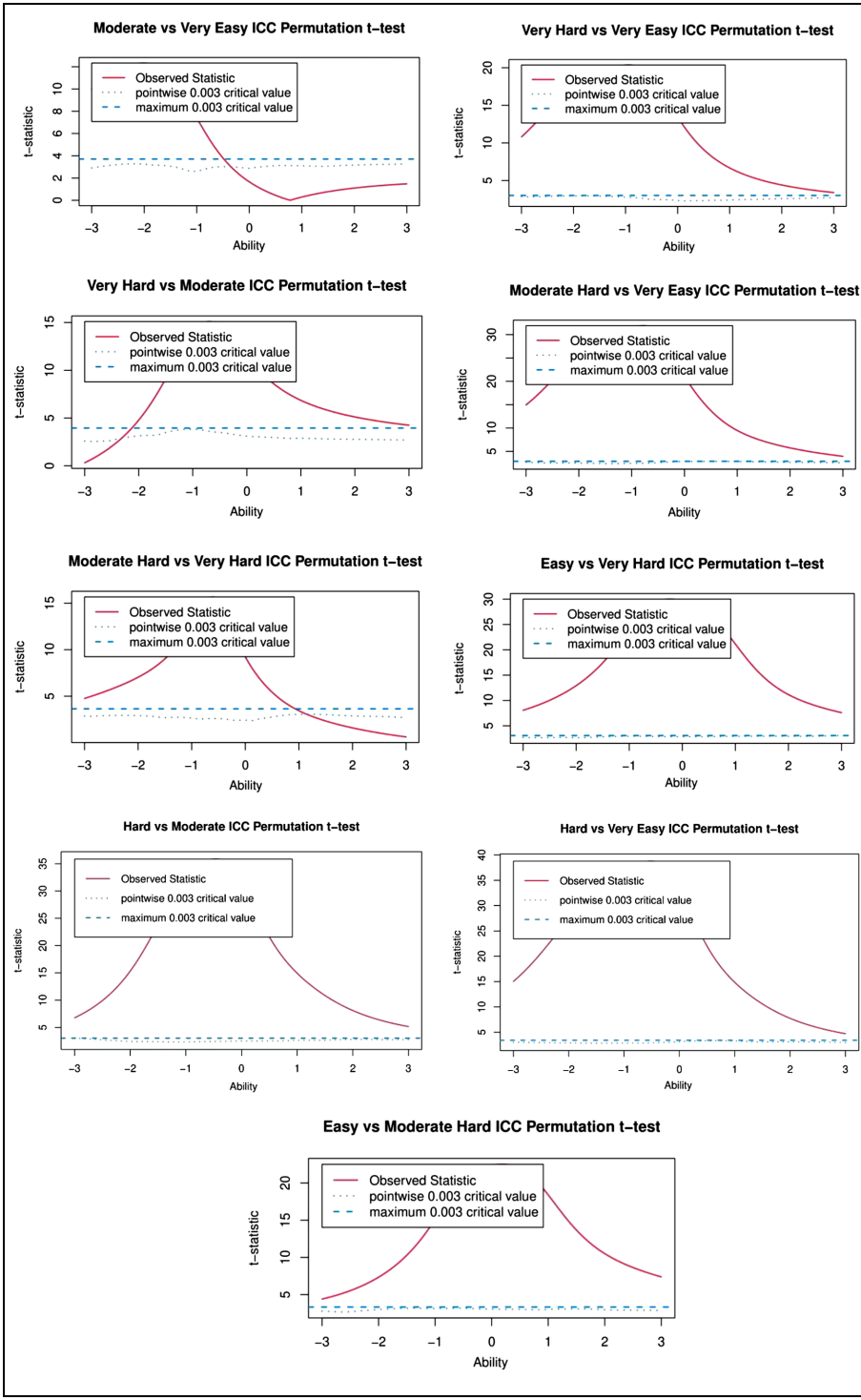

Figure 8. Non-Adjacent Cluster Group Permutation *t*-Test Comparisons.

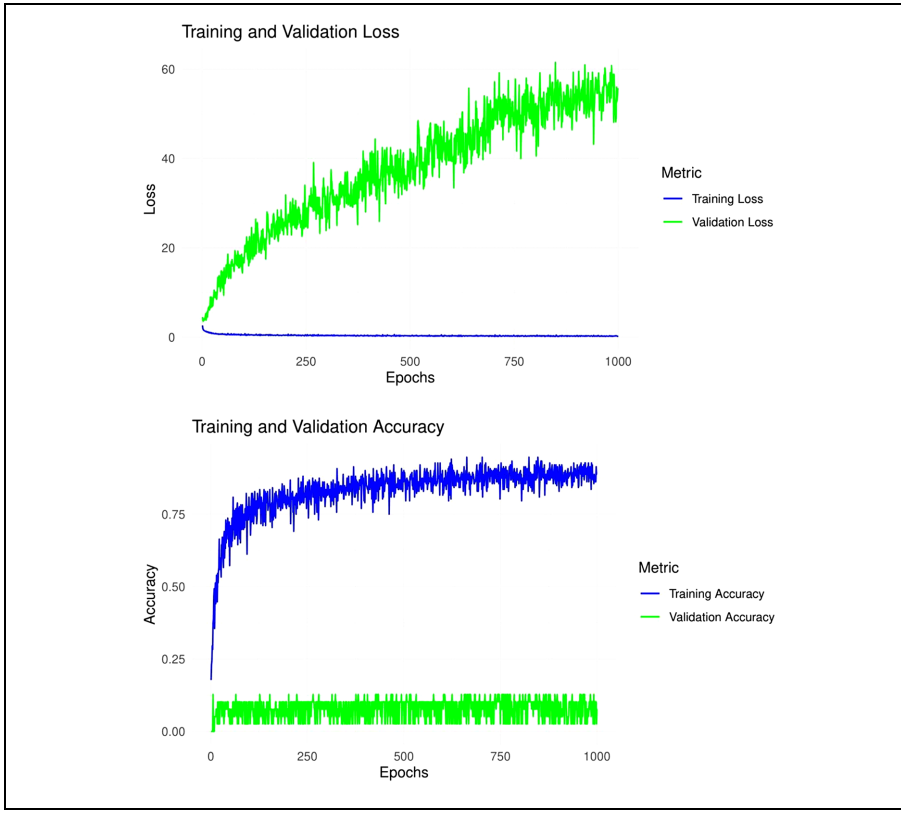

Figure 9. Training (Blue/Top graphs) and Validation (Green/Bottom graphs) Set Loss (Top figure) and Accuracy (Bottom figure) Metrics Over the 1,000 Epochs Used to Train the Neural Network.

1,000 epochs used to train the neural network. Ideally, as the neural network learns, we expect to see a gradual decrease in loss and a corresponding increase in accuracy over the epochs. Examining the trends of the loss function for both the training and validation sets, there appears to be signs of overfitting, as the training loss keeps decreasing slightly while the validation loss begins to rise. This notion is corroborated by the trends seen for accuracy. The accuracy for training increases while the validation accuracy oscillates around a point. When this trained neural network was given the remaining ICCs as a test set, it accurately predicted the difficulty classifications 79.6% of the time during its first application to test data.

A confusion matrix was generated and is presented in Table 3 to identify where misclassifications exist when applying the neural network to the test set. This matrix compares the predicted difficulty level of an item from the test set against its actual difficulty level. Upon reviewing the confusion matrix, we observed that in the row

Table 3. Confusion Matrix.

Prediction classification	Actual (FPC cluster) classification						
	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Cluster 6	Total
Cluster 1	10	0	0	0	0	6	16
Cluster 2	0	6	1	0	0	0	7
Cluster 3	0	0	6	0	0	0	6
Cluster 4	0	0	0	9	0	0	9
Cluster 5	0	0	0	0	8	3	11
Cluster 6	0	0	0	0	0	0	0
Total	10	6	7	9	8	9	49

for Cluster 1 (the *very hard* cluster), 10 items were correctly classified, while six were misclassified. These six items were classified into Cluster 6 (the *hard* cluster), but the neural network incorrectly placed them to be in Cluster 1. For Cluster 2, six ICCs were correctly classified, with one misclassification. This misclassified ICC was predicted to belong to Cluster 2 (the *easy* cluster) but was grouped in Cluster 3 (the *moderate* cluster). In addition, in the row for Cluster 5 (the *moderately hard* cluster), we find three misclassified ICCs. The neural network predicted these three ICCs to be part of Cluster 5, although they were grouped in Cluster 6. Overall, 10 out of 49 ICCs were misclassified. All other entries in the confusion matrix show agreement between the model's predictions and the actual clustering, resulting in the reported accuracy rate of 79.6%.

To assess the consistency of the results, the neural network was re-trained and tested on different subsets two additional times. While classification accuracy remained around 80%, some misclassifications involved items being predicted as belonging to clusters two or more difficulty levels away from their true clusters in both re-trials. Specifically, three items with a true cluster of 3 (*moderate*) were misclassified as belonging to Cluster 6 (*hard*), and two items with a true cluster of 3 (*moderate*) were misclassified as Cluster 1 (*very hard*).

Among the consistently misclassified items, the discrimination parameter ranged from 1.133 to 1.804, indicating that these items should effectively differentiate between examinees of varying ability levels. The difficulty parameter for these items ranged from 0.172 to 2.211, suggesting that classification challenges were not confined to specific difficulty levels. In addition, the guessing parameter ranged from 0.113 to 0.19 among the misclassified items, further indicating no clear pattern linked to guessing.

Based on these item-level parameters, it appears that the neural network may have been particularly sensitive to small variations in ability near the threshold defined by the difficulty parameter. The network might have struggled to fully capture the variations in slope (discrimination) across the ICCs, especially within the ability range of [–1, 1], where the largest differences between *moderate* and *very hard* clustered items

occur. Across all tests, items falling between the moderate and very hard clusters were consistently the most problematic during classification.

Discussion

Test items are building blocks of any assessment (Haladyna & Rodriguez, 2013). The quality and validity of test forms largely depend on the quality of the items it contains. Item difficulty is a crucial factor in determining the overall effectiveness of both individual items and the test as a whole. As a result, accurately predicting item difficulty is vital in any educational setting (AlKhuzaey et al., 2021). Traditional methods for estimating item difficulty involved either placing items on test forms and pre-testing them as field test items or relying on expert judgment when assessing item difficulty qualitatively (Wauters et al., 2012). Pre-testing items involves embedding them into operational test forms and administering them as if they were live items; however, the data collected from these items do not contribute to the total score. The items are then scored based on the assumptions of Classical or Item Response Theories. While this method is commonly used by large testing programs, smaller assessment programs may require multiple field test administrations of new items to reliably establish their statistical characteristics. The second approach, relying on expert judgment, presents even more challenges. Subject matter experts may introduce bias based on their own knowledge or experience, leading to difficulty estimations that may not align with actual results once the items are operational. This can result in a discrepancy between the expected and observed difficulty levels during live administrations.

Several studies examined data-driven approaches to item difficulty estimation. For example, Hsu, Lee, Chang & Sung (2018) presented a novel method for automating item difficulty estimation in social studies tests, specifically for multiple-choice questions. Using word-embedding techniques, they constructed a semantic space from learning materials and projected item texts into this space to generate vectors. By calculating cosine similarity between the vectors of item elements, they extracted semantic features, which were then used to train and test a classifier. Another study compared various ML methods for predicting item difficulty in English reading comprehension tests using text features from item wordings. The analysis covered a range of ML algorithms, including regularization methods, support vector machines, decision trees, random forests, neural networks, and Naïve Bayes, applied to both regression and classification tasks. The results showed that ML algorithms using text features can rival expert predictions, making them useful when item pre-testing is limited or unavailable (Štěpánek et al., 2023). However, neither of these algorithms has been implemented in an operational testing program nor evaluated using data from such a program.

This study outlined an empirical process for classifying items in difficulty categories through FDA and neural networks. The process involved clustering according to FPC scores extracted from the ICCs. Comparing this functional difficulty clustering method to categorization through visualization, there were discrepancies, but

almost all were only differing by one adjacent class. The discrepancies do not necessarily mean one method is wrong, but they do show differing points of emphasis when categorizing can lead to different results, and therefore, consistency in decision making is important. Empirical data-driven methods of classification may provide better consistency if they are structured correctly. Future studies may want to compare the results of differing cluster algorithms to see which has the highest rate of agreeance with what a test creation team believes. With this in mind, it is important to recall fuzzy clustering allows for observations to have overlapping membership in available clustering which introduces some assignment variation. This means replications may not result in exactly the same outcomes, although they may be very similar. If an operational testing program were to use fuzzy clustering, they would need to save the results of the agreed-upon clustering algorithm. This would allow the extracted FPCs and centroids to be reused, ensuring consistent cluster membership when new items are introduced into the algorithm. In addition, it must be considered that an item's history influences how they are categorized. Adding a weighting mechanism to slightly bias the results of clustering and/or classification algorithm based on an item's history may also lead to greater rates of agreement.

Regarding the neural network's performance, the accuracy on the test data was approximately 80%. The item cluster classes that posed the greatest classification challenges ranged from *moderate* to *very hard*. With a larger set of ICCs for training, the model could potentially achieve higher accuracy on the test set, as the training data accuracies approached 90%, as indicated by the plot. Increasing the number of epochs during training might also have improved the network's ability to learn subtle differences in the functional forms between adjacent classes. Furthermore, it was mentioned that the plots for the neural network may have indicated overfitting to the training set. Some control measures for future neural networks could be integrated to adjust for this. Options include setting a maximum norm value to cap the weights' norms in the neural network at a certain threshold, adjusting the weights during training if necessary, experimenting with different learning rates, modifying the batch size, and other similar adjustments. Considering the relatively small sample of items in this study, a simpler psychometric model might have resulted in more accurate classification by the neural network. The Rasch model, which estimates only the difficulty parameter, provides more precise difficulty estimates than the 2PL or 3PL models in smaller sample scenarios (Thissen & Wainer, 1982). Consequently, a neural network trained on ICC structures derived from a Rasch model may capture more precise patterns, leading to higher classification accuracy.

Aside from some of the limitations of this pilot design, its utility would be seen in testing programs where results from administration come in at various times of the year and in smaller quantities. Connecting a pipeline categorization algorithm to a program's database can streamline ICC generation, simplify item labeling, and offer an empirical basis for item categorization. This may even be integrated into a fully automated system, which takes test results, adjusts items classifications, flags anomalies, and then generates various forms of tests to increase test production while

saving on labor resources. Altogether this process presents a data-driven approach for item classification meant to uphold the structural integrity of a test that can be tailored to the evolving needs of testing programs.

Declaration of Conflicting Interests

The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

ORCID iD

Igor Himelfarb https://orcid.org/0000-0002-2622-6062

References

Abdi, H., Valentin, D., & Edelman, B. (1999). *Neural networks* (No. 124). Sage.

Abraham, C., Cornillon, P. A., Matzner-Løber, E., & Molinari, N. (2003). Unsupervised curve clustering using B-splines. *Scandinavian Journal of Statistics, 30*(3), 581–595.

AlKhuzaey, S., Grasso, F., Payne, T. R., & Tamma, V. (2021). A systematic review of data-driven approaches to item difficulty prediction. In I. Roll, D. McNamara, S. Sosnovsky, R. Luckin & V. Dimitrova (Eds.), *International conference on artificial intelligence in education* (pp. 29–41). Springer.

Baker, F. B. (2001). *The basics of item response theory*. ERIC Clearinghouse on Assessment and Evaluation.

Belov, D. I. (2021, July). Predicting item characteristic curve (ICC) using a softmax classifier. In M. Wiberg, D. Molenaar, J. González, J.-S. Kim, & H. Hwang (Eds.), *Quantitative psychology: The 86th annual meeting of the psychometric society* (pp. 171–184). Springer.

Benko, M., Härdle, W., & Kneip, A. (2009). Common functional principal components. *Annals of Statistics*, *37*(1), 1–34.

Bennett, R. E. (2011). Formative assessment: A critical review. *Assessment in Education: Principles, Policy & Practice*, *18*(1), 5–25.

Birnbaum, A. (1968). Some latent trait models and their use in inferring an examinee's ability. In F. M. Lord & M. R. Novick (Eds.), *Statistical theories of mental test scores* (pp. 397–479). Addison-Wesley.

Bolleddu, P. (2022). An analysis of clustering techniques. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, *8*(2), 52–57.

Bonferroni, C. (1936). Teoria statistica delle classi e calcolo delle probabilita. *Pubblicazioni del istituto superiore di scienze economiche e commericiali di firenze*, *8*, 3–62.

Buckendahl, C. W. (2017). Credentialing: A continuum of measurement theories, policies, and practices. In S. Davis-Becker & C. W. Buckendahl (Eds.), *Testing in the professions* (pp. 1–20). Routledge.

Craven, P., & Wahba, G. (1978). Smoothing noisy data with spline functions. *Numerische Mathematik*, *31*(4), 377–403.

Dai, X., & Müller, H. G. (2018). Principal component analysis for functional data on Riemannian manifolds and spheres. *The Annals of Statistics*, *46*(6), 3334–3361.

Embredson, S. E., & Reise, S. P. (2000). *Item response theory for psychologists*. Lawrence Erlbaum.

Engelhard, G. (2023). Functional approaches for modeling unfolding data. *Educational and Psychological Measurement*, *83*(6), 1139–1159.

Febrero-Bande, M., & De La Fuente, M. O. (2012). Statistical computing in functional data analysis: The R package fda. usc. *Journal of Statistical Software*, *51*, 1–28.

Ferraro, M. B., Giordani, P., & Serafini, A. (2019). *fclust: Fuzzy clustering* (R package version 2.2.1). https://CRAN.R-project.org/package=fclust

Fortuna, F., & Maturo, F. (2018). K-means clustering of item characteristic curves and item information curves via functional principal component analysis. *Quality & Quantity*, *53*(5), 2291–2304.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. The MIT Press.

Gurney, K. (2018). *An introduction to neural networks*. CRC Press.

Haladyna, T. M., & Rodriguez, M. C. (2013). *Developing and validating test items*. Routledge.

Hall, P., & Hosseini-Nasab, M. (2006). On properties of functional principal components analysis. *Journal of the Royal Statistical Society Series B*, *68*(1), 109–126.

Hall, P., Müller, H.-G., & Wang, J.-L. (2006). Properties of principal component methods for functional and longitudinal data analysis. *The Annals of Statistics*, *34*(3), 1493–1517.

Hambleton, R. K., Swaminathan, H., & Rogers, H. J. (1991). *Fundamentals of item response theory*. Sage Publications.

Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (*Vol. 2*, pp. 1–758). Springer.

Haykin, S. (1994). *Neural networks: A comprehensive foundation*. Prentice Hall.

Hsu, F. Y., Lee, H. M., Chang, T. H., & Sung, Y. T. (2018). Automated estimation of item difficulty for multiple-choice tests: An application of word embedding techniques. *Information Processing & Management*, *54*(6), 969–984.

Ieva, F., Paganoni, A. N. N. A., Pigoli, D., & Vitelli, V. (2013). Multivariate functional clustering for the analysis of ECG curves morphology. *Applied Statistics*, *62*(3), 401–418.

Jacques, J., & Preda, C. (2013). Funclust: A curves clustering method using functional random variables density approximation. *Neurocomputing*, *112*, 164–171.

Jacques, J., & Preda, C. (2014). Functional data clustering: A survey. *Advances in Data Analysis and Classification*, *8*(3), 231–255.

Khan, A., Baseer, S., & Javed, S. (2017). Perception of students on usage of mobile data by K-mean clustering algorithm. *International Journal of Advanced and Applied Sciences, 4*(2), 17–21.

Kline, P. (2015). *A handbook of test construction (psychology revivals): Introduction to psychometric design*. Routledge.

Kokoszka, P., & Reimherr, M. (2017). *Introduction to functional data analysis*. Chapman and Hall/CRC.

Kolen, M. J., & Brennan, R. L. (2014). *Test equating: Methods and practices*. Springer.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444.

Liao, T. W. (2005). Clustering of time series data—A survey. *Pattern Recognition*, *38*(11), 1857–1874.

Lord, F. M. (1975). The ‘‘ability’’ scale in item characteristic curve theory. *Psychometrika*, *40*(2), 205–217.

Lord, F. M. (1977). Practical applications of item characteristic curve theory. *Journal of Educational Measurement*, *14*(2), 117–138.

Lord, F. M., & Novick, M. R. (1968). *Statistical theories of mental test scores*. Addison-Wesley.

Marcoulides, K. M., & Trinchera, L. (2024). A novel approach for identifying unobserved heterogeneity in longitudinal growth trajectories using natural cubic smoothing splines. *Journal of Behavioral Data Science*, *4*(1), 1–18.

Mislevy, R. J. (1992). *Linking educational assessments: Concepts, issues, methods, and prospects*. ETS Policy Information Center Report.

Muijtjens, A. M., Kramer, A. W., Kaufman, D. M., & Van der Vleuten, C. P. (2003). Using resampling to estimate the precision of an empirical standard-setting method. *Applied Measurement in Education*, *16*(3), 245–256.

Müller, B., Reinhardt, J., & Strickland, M. T. (2012). *Neural networks: An introduction*. Springer.

National Board of Chiropractic Examiners. (2024). Part III. *NBCE.org*. https://www.mynbce.org/pt-iii/

Nielsen, M. A. (2015). *Neural networks and deep learning*. Determination Press.

Piernik, M., & Morzy, T. (2021). A study on using data clustering for feature extraction to improve the quality of classification. *Knowledge and Information Systems*, *63*(7), 1771–1805.

Ramsay, J. O., & Silverman, B. W. (2005). *Functional data analysis* (2nd ed.). Springer.

Ramsay, J. O., & Silverman, B. W. (Eds.). (2002). *Applied functional data analysis: Methods and case studies*. Springer.

Rosenbaum, P. R. (1987). Comparing item characteristic curves. *Psychometrika*, *52*, 217–233.

Shang, H. L. (2014). A survey of functional principal component analysis [Review of A survey of functional principal component analysis]. *AStA Advances in Statistical Analysis*, *98*, 121–142.

Štěpánek, L., Dlouhá, J., & Martinková, P. (2023). Item difficulty prediction using item text features: Comparison of predictive performance across machine-learning algorithms. *Mathematics*, *11*(19), 4104.

Suk, H. W., West, S. G., Fine, K. L., & Grimm, K. J. (2019). Nonlinear growth curve modeling using penalized spline models: A gentle introduction. *Psychological Methods*, *24*(3), 269–290.

Suskie, L. (2018). *Assessing student learning: A common sense guide*. Wiley.

Swingler, K. (1996). *Applying neural networks: A practical guide*. Morgan Kaufmann.

Tarpey, T., & Kinateder, K. (2003). Clustering functional data. *Journal of Classification*, *20*(5), 93–114.

Thissen, D., & Wainer, H. (1982). Some standard item response theory applications. In H. Wainer & D. Thissen (Eds.), *New directions for testing and measurement* (pp. 397–412). Jossey-Bass.

Trivedi, S., Pardos, Z. A., & Heffernan, N. T. (2015). The utility of clustering in prediction tasks. *ArXiv* (Cornell University).

Tsai, C.-F., Lin, W.-Y., Hong, Z.-F., & Hsieh, C.-Y. (2011). Distance-based features in pattern classification. *EURASIP Journal on Advances in Signal Processing*, *2011*(1), 62.

Wang, J. L., Chiou, J. M., & Müller, H. G. (2016). Functional data analysis. *Annual Review of Statistics and Its Application*, *3*(1), 257–295.

Wauters, K., Desmet, P., & Van Den Noortgate, W. (2012). Item difficulty estimation: An auspicious collaboration between data and judgment. *Computers & Education*, *58*(4), 1183–1193.

Wu, R., Wang, B., & Xu, A. (2021). Functional data clustering using principal curve methods. *Communications in Statistics—Theory and Methods*, *51*(20), 7264–7283.

Zwick, R. (2002). *Fair game? The use of standardized admissions tests in higher education*. Routledge.