So Happy Together?
Combining Rasch and Item Response Theory Model
Estimates with Support Vector Machines to Detect Test Fraud

Sarah L. Thomas
Shelby, NC

B.S. University of North Carolina, 2010
M.A. Wake Forest University, 2012

A Dissertation presented to the Graduate Faculty of the University of Virginia
in Candidacy for the Degree of Doctor of Philosophy

Department of Psychology

University of Virginia
May, 2016

Karen M. Schmidt

Steven M. Boker

Timo von Oertzen

J. Patrick Meyer

Acknowledgements

I am incredibly grateful to my advisor, Dr. Karen Schmidt, whose continual guidance, feedback, support, and patience have helped me develop important skills and achieve my goals. Thank you for all the time and effort you have invested in me.

Dr. Steve Boker, Dr. Timo von Oertzen, and Dr. Patrick Meyer generously agreed to serve as my committee members and have guided, challenged, and aided me throughout this process.

My mother, Ann Thomas, has been an unfailing source of encouragement and delightful “momisms” that have brightened my days and given me the motivation to keep pushing towards my goals.

My thanks, and probably a large quantity of wine, are undoubtedly due to my dear friend and roommate, Erin Westgate, who has comforted me on bad days, celebrated with me on good days, and has made this journey more fun at every step of the way.

I feel an overwhelming sense of gratitude towards my beloved boyfriend, Daniel Toton, who has loved me unconditionally throughout this process and who has exhibited an incredible level of patience and understanding towards me I’ve gone through the long process of obtaining my PhD. You are truly the best.

I would also like to thank the rest of the Toton family, Andrea Worsham, Rachel Howard, Myka Morgan, and my colleagues at the University of Virginia for encouraging me to keep pushing and helping me maintain my sanity.

This is by no means a complete list of those who deserve acknowledgement so to all of the rest of you, I’d like to say thank you from the bottom of my heart.

Table of Contents

List of Tables

Table Page

List of Figures

Figure Page

Abstract

Each year, thousands of people must pass standardized exams to be certified in medical, legal, clinical, and technological fields. Unfortunately, the increasing number of examinees taking such tests seems to have been accompanied by an increase in the instances of reported cheating and the invention of more sophisticated cheating techniques. The stealing and sharing of proprietary test content, such as when items are recorded and then compromised via sharing, is associated with legal consequences for the culprit but can also have negative consequences for the integrity, reputation, and budget of a testing program. Compromised items represent a significant threat to the integrity of testing. The purpose of the present study was to detect items that were suspected to be compromised using a combination of Rasch and Item Response Theory model estimates, along with other item properties such as average response times and local dependence estimates, with Support Vector Machines (SVMs). In this study, we used this combination of methods to detect items of an international healthcare certification exam (*N* = 13,584) that were suspected to be compromised in screenshots or notes. The results showed that this method appeared to be somewhat accurate at classifying suspected compromised and suspected uncompromised items, but that the main factor driving these results appeared to be the relative size of the classes. The SVMs showed apparent bias towards predicting the items into the category of whichever item class was larger. Thus, the accuracy of the SVMs, balanced for class size, was much lower than desired. We hypothesized that the most important item features would be Rasch item infit and outfit, but the item feature results showed that the two most important features were the Rasch model standard error and the Rasch item difficulty. We also hypothesized that Rasch

estimates would outperform 3PL estimates, but the evidence for this hypothesis was mixed, with some Rasch estimates outperforming the 3PL estimates and others performing worse than the 3PL estimates. Our final hypothesis was that the 3PL discrimination would outperform the 3PL lower limit. The evidence for this final hypothesis was somewhat mixed, but discrimination outperformed the lower limit in the majority of the models. However, the results of the feature weights in the current study should be interpreted cautiously for several reasons. The assessment of feature weights provides information on the importance of features in the data for deriving the classifications made by the models. In the current study, the classifications made by the models seemed to be biased by class size and were often inaccurate. Thus, the feature weights may contribute very little to the understanding of what item properties actually distinguish between suspected uncompromised and suspected compromised items. The current study combined Rasch and Item Response Theory model estimates with Support Vector Machines, bringing the disparate fields of psychometrics and machine learning together, and showing a promising future for the resulting hybrid method.

So Happy Together? Combining Rasch and Item Response Theory Model Estimates with Support Vector Machines to Detect Test Fraud

Each year thousands of people must pass standardized certification exams to be certified in medical, legal, clinical, and technological fields. In 2014 alone, over 100,000 U.S. medical licensing examinations were taken by M.D. and D.O. degree holders (United States Medical Licensing Examination Performance Data, 2014) and the number of examinees who take high-stakes standardized tests increases each year. Unfortunately, this increase in the pervasiveness of standardized testing seems to have been accompanied by an increase in the instances of reported cheating (Bliss, 2012; FairTest, 2007; Simha, Armstrong, & Albert, 2012) and the invention of more sophisticated cheating techniques (Sorenson, 2010).

Cizek (2003) broadly defines cheating in this context as any act that breaks an established rule about the administration or completion of a test or assignment, provides an unfair advantage to some students, or makes the interpretations of a score less accurate. Although cheating could consist of a variety of different behaviors, cheating specifically on tests is sometimes referred to as *test fraud* (Kingston & Clark, 2014). Test fraud is any prohibited or dishonorable action deliberately taken by an examinee in order to attain undeserved high scores on tests, or aid others in doing so (Thomas et al., 2016). Test fraud not only penalizes honorable test takers by allowing unscrupulous examinees to appear relatively more proficient (Fremer & Addicott, 2011), it can also damage the integrity and reputation of the testing program itself (Cizek, 1999; Dickison & Maynes, 2014; Fremer & Addicott, 2011).

Although data on the prevalence of test fraud is understandably difficult to obtain, a meta-analysis of cheating in U.S. and Canadian college students found that reported cheating rates ranged from 9% to 95%, with an average of 70.4% of students admitting to some form of cheating while in college (Whitley, 1998). One might expect that the desire to cheat on standardized certification exams would be as high as the desire to cheat in college, but the rate of cheating on standardized exams to be lower due to stricter exam security measures and the potential legal consequences of being caught.

Some of the more advanced test fraud tactics involve examinees recording test items, response options, and/or presumed test answers and sharing them with future test-takers (Smith, 2004; Thomas et al., 2016). The result is examinees with pre-knowledge, so-called because the examinees have accessed exam content before they begin the exam. Like data on the prevalence of test fraud in general, data on the frequency of examinee pre-knowledge is challenging to obtain. Many testing companies who may have insights into the prevalence of examinee pre-knowledge do not provide this information, possibly due to their desire for confidentiality. One source suggests that answer keys may be accessed by more than 20% of the test-taking population of information technology certification exams, depending on the desirability of the certification (Maynes, 2008). However, it is unclear what evidence underlies this assertion, and if this result generalizes to certification exams in other career areas.

The purpose of the present study was to detect items that were compromised, or leaked, using a combination of Rasch and Item Response Theory (IRT) models, common in education and psychology with Support Vector Machines (SVMs) from the field of machine learning. The investigated certification exam was compromised by screenshots and notes that contained proprietary test items and answers. These screenshots and notes were discovered by the test publisher, creating a situation in which there were suspected item categories. These item categories consisted of *suspected compromised items,* which the test publisher discovered in the screenshots and/or notes, and *suspected uncompromised items,* which the test publisher did not discover in the screenshots and/or notes. The objective of the current study was to classify items as *predicted compromised items* or *predicted uncompromised items* according to their psychometric characteristics. Predicted compromised items were those that are suspected of being compromised in the screenshots and/or notes based on their properties and predicted uncompromised items are those that were not suspected of being compromised in the screenshots and/or notes. The predicted item categories were then compared to the suspected item categories.

The Role of Statistics in Detecting Test Fraud

Using statistical tools to detect test fraud is particularly important in cases of examinee pre-knowledge, as there may be no detectable, external signs of cheating because the proprietary test information may have been memorized (Bliss, 2012). Previous investigations into the detection of pre-knowledge (e.g. a student obtaining answers from a student who previously took the test; Hui, 2010; Shu, 2011) have focused largely on comparisons of person-fit indices (Clark et al., 2014; Karabatsos, 2003; Marianti, Fox, Avetisyan, Veldkamp, & Tijmstra, 2014; Sinharay, 2015). Additional comprehensive reviews of statistical methods used to detect various types of cheating are found in Kingston and Clark (2014), Bliss (2012), or Haney and Clarke (2007).

Karabatsos (2003) assessed the performance of 36 person-fit indices in detecting aberrant response patterns, i.e., those that do not align well with expectations; for example, a low ability individual answering difficult items correctly, but easy items incorrectly. In this study, Karabatsos (2003) used a Rasch model on simulated data that included five aberrant response patterns (cheating, creatively and uniquely interpreting item meanings and choosing answers accordingly, guessing, careless responding, and random responding). The top five best person fit statistics were identified; the best one, H^t (Sijtsma, 1986; Sijtsma & Mejer, 1992), compared conformity between an examinee's response pattern and the response patterns of all other examinees (Karabatsos, 2003). However, the results of a similar study in which various IRT models were applied to simulated data to detect simulated cheaters' data showed that the best method, *lco* difference (Ferrando, 2007), a sum of squared, standardized residuals across items, was better than other methods, including H^t (Clark et al., 2014). However, all of the tested person-fit methods performed poorly on data with low rates of cheating.

In the present study, we chose to focus on classifying items rather than focusing on analyses of person fit for several reasons. First, person fit statistics do not necessarily identify compromised items, given their person-centered focus, whereas the direct focus on items would likely play a vital role in addressing instances of test fraud and preventing future test fraud. If test publishers have information on which items are compromised, they can alter or remove those items. Some test publishers have even turned compromised items into Trojan Horse items, which are relatively easy, compromised items that have been slightly altered on the test so that examinees with pre-knowledge will respond incorrectly to these items (Caveon, 2008). For example, if the original compromised item reads "What is $x^4 + x^3 – x$ if $x^2 = 16$?", the test publisher could alter the item to read "What is $x^4 + x^2 – x$ if $x^2 = 16$?" where the second term is now squared instead of cubed, in hopes that examinees with pre-knowledge will overlook the small change in the item and select the correct answer for the original, compromised item, thus responding incorrectly to the altered item. Secondly, information on item classifications may be useful in accurately classifying examinees. This is because we expect examinees with pre-knowledge who are responding outside of their true trait level to show differential functioning on suspected compromised and suspected uncompromised items. Thus, if the predicted item categories are relatively accurate, differences in person performance that reflect the suspected item categories may be detected. For example, holding all other factors (e.g. item difficulty, trait level, etc.) constant, examinees with pre-knowledge may respond relatively faster on suspected compromised items than on suspected uncompromised items, but examinees without pre-knowledge may show no difference or a smaller difference. Additionally, examinees with pre-knowledge may respond more correctly on suspected compromised items than on suspected uncompromised items, but examinees without pre-knowledge may show no difference or a smaller difference, again holding all other factors (e.g. item difficulty, trait level, etc.) constant. These differences in person responses are most likely to be detectable when examinees with pre-knowledge are responding to items far from their trait level. For example, suppose that two examinees with average latent trait levels respond to a very difficult item. We expect both examinees to respond incorrectly, but suppose one answers the item correctly, and does so very quickly. This response pattern is unexpected, but could represent a number of circumstances besides pre-knowledge. The person could have studied a similar problem right before the test or been quickly guessing answers and happened to select the correct response. Thus, this unexpected response pattern on one item may not indicate pre-knowledge, but over a number of items, these unexpected response patterns could provide evidence of pre-knowledge. Bear in mind that the estimated trait level of an examinee with pre-knowledge depends on many factors, including the examinee’s ability to memorize and recall answers to compromised items and the proportion of compromised items. Our philosophy is that item classifications are important for addressing instances of test fraud and could be useful for the classification of examinees. Additionally, when assessing person fit for the purpose of detecting test fraud in real data, there is rarely a way to check the accuracy of one’s results because information regarding which examinees have pre-knowledge and which do not is not typically available, as in the case with the current set of data. Thus, we will focus on predicting the suspected item categories.

In the current study, we chose to focus on a combination of Rasch and Item Response Theory (IRT) models with Support Vector Machines (SVMs) for a number of reasons. First, Rasch and IRT model estimates provide valuable psychometric information about items and persons in a very detailed manner. For example, Rasch item infit and outfit estimates assess mismatch between a person’s expected response to an item and their actual response. This type of information is more complex than the information contained in many item statistics, such as item-total correlations, which simply indicate the direction and strength of the relationship between a response on an item and the total exam scores. Infit and outfit give more nuanced information by accounting for a person’s estimated ability level at the level of the item responses. These estimates could be valuable in detecting compromised items. Secondly, until now, many statistical analyses of test fraud have not been able to harness the power of data mining techniques like SVMs because category information on the status, or suspected status, of items was not available, but is required in order to use an SVM. Thirdly, this combination of analyses offers advantages to the detection of test fraud that are not achieved by either analysis alone. Rasch and IRT model estimates provide detailed information about items, but the effect of item compromise on the interpretation of Rasch and IRT model estimates is often unclear. For example, high item infit estimates, which indicate a larger than expected difference between expected and actual responses, are often interpreted as indicating problems with the quality of the item (Bohlig, Fisher, Masters, & Bond, 1998). However, in the case of item compromise the interpretation is unclear, i.e., high item infit could indicate poor item quality or item compromise. Thus, using Rasch and IRT model estimates gives us detailed information on the functioning of the items in the sample and the performance of persons in the sample, but does not allow us to use that information to predict instances of item compromise. Through the combination of Rasch and IRT models with SVMs, it becomes possible to investigate the differences between suspected compromised and suspected uncompromised items on Rasch and IRT model estimates. The SVMs are an unsupervised process, which allows the data to drive the predictions. SVMs also have the advantage of not being limited to the investigation of linear relationships. The combination of these methods marries detailed information on items and persons with the advantages of a data-driven classifier. Previous research has shown that using a combination of cluster, correlational, and Rasch model analyses to classify items as predicted compromised or predicted uncompromised yielded accurate results. Thomas et al. (2016) found that 64 of the 65 items were correctly classified using a combination of statistical flags from cluster, correlational, and Rasch model analyses.

They also showed that correlational and Rasch model analyses generally performed better than cluster analyses. Using Rasch model estimates in the current study will allow us to replicate and expand their work. Additionally, Thomas et al. (2016) used statistical cutoffs to create flags, but the method used in the current study allow the estimates to serve as predictors of item compromise in their continuous state, without imposing the limitation of using cutoff scores. In other words, the estimates are free to predict item compromise without the researcher imposing constraints on the direction of prediction. Thus, the current study expands and improves the work of Thomas et al. (2016) in addition to bringing two fairly disparate fields together. In the current study, we believe that a combination of the Rasch and IRT models with Support Vector Machines may yield accurate item classifications, creating a happy union of the methods.

If this combination method performs well, test publishers interested in detecting item compromise may be able to predict the status of their items even in the absence of suspected item categories. This would be a huge advantage for test publishers, who often pay for continual internet monitoring to detect proprietary test content online (Fremer & Addicott, 2011). Not only does internet monitoring represent a significant cost in terms of time and money, it is also an inefficient method for detecting compromised items as the amount of information that must be searched is potentially infinite. Moreover, proprietary content may be hidden behind paywalls, encrypted, or hidden from search functions in other ways. Once possible proprietary content has been found, the testing company must investigate to verify that the content in question is proprietary content and if it is, may begin an official investigation. However, if compromised items could be detected using the method proposed in this paper, then this method could be periodically applied to sets of testing data for detection of compromised items. The test publishers could then alter or remove items that are predicted to be compromised. Furthermore, the method itself is flexible and could be used not only in the field of test fraud study, but also in any classification problem that could use Rasch and IRT model estimates as properties of units. This method could be used with persons, rather than items as the unit of interest. The features for the persons could be Rasch or IRT model person estimates (e.g. trait level, person infit, person outfit, etc.). For example, this method could be used to train a system to classify students who are in an English honors program or not, based on their performance on previous English tests. Next, the method could be used to classify other students into those who resemble the students in the honors program and those who do not. This type of analysis could be used to identify students who should be included in honors programs but are currently not.

In the following sections, techniques that yield item properties that may be useful in detecting item compromise will be introduced and described including: Rasch model estimates, three-parameter logistic (3PL) IRT model estimates, item response times, Yen's (1984) Q_3 for identifying local dependence of items, and linear logistic test model (LLTM) estimates. Then the methods for using these item properties and the suspected item categories to create functions that can separate compromised items from uncompromised items, which, in the current study, are Support Vector Machines (SVMs) will be discussed. Item properties that will be used in the SVMs will be called features, in keeping with the literature on SVMs (Guyon, Weston, Barnhill, & Vapnik, 2002).

The Rasch model

Broadly, the Rasch model (Rasch, 1960) estimates the probability of a person responding to an item in a certain category (here, with a correct response), given their latent trait level (θ_{Rasch}) and the difficulty of the item (R Diff; See Equation 1). The Rasch model estimates item and person parameters on the same logit scale, which allows them to be easily compared (Rasch, 1960). The difficulty of the item (R Diff) is the location of the item on the logit scale and can be seen on the Item Characteristic Curve (ICC; see Figure 1). The Rasch model is superior to other models, such as the two-parameter logistic (2PL; Lord & Novick, 1968), and the three-parameter logistic (3PL, Birnbaum, 1968) models, because it provides interval-scale measurement (Wright & Linacre, 1989) and double cancellation (Embretson & Reise, 2000). Double cancellation requires that the probability of answering an item correctly increases as person ability increases and decreases as item difficulty increases across ICCs (Embretson & Reise, 2000). When the property of double cancellation is achieved, the ICCs do not cross one another. The 2PL and 3PL models do not ensure double cancellation and are therefore considered lacking in important measurement properties by some psychometricians (e.g. Embretson & Reise, 2000). The Rasch model was used in the current study because it is well-suited to analyzing scored test data and is flexible (Rasch, 1960). Additionally, previous research shows that item measures commonly obtained through Rasch model estimation are useful for the detection of test fraud (Thomas et al., 2016).

The Rasch model can be represented as:

Equation 1

$$P(X_{ij} = 1 | \theta_j, \beta_i) = \frac{\exp(\theta_j - \beta_i)}{1 + \exp(\theta_j - \beta_i)} \text{ ,}$$

where θ_j represents person *j*'s latent trait score, β_i represents the difficulty of item *i*, and *P(X$_{ij}$ = 1*) represents the probability that person *j* choses category *X* for item *i*, given β_i and θ_j.

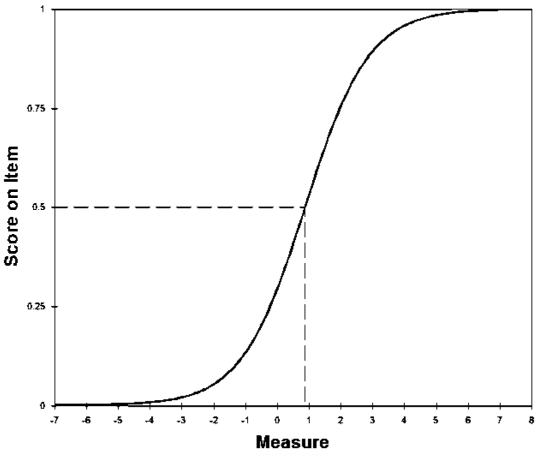

Figure 1. An expected Rasch model Item Characteristic Curve (ICC). Note that the difficulty (R Diff = 0.87, in this case) is equal to the location at which the probability of getting the item correct is 0.50.

Rasch model assumptions. The Rasch model assumes that items are unrelated to each other except through the underlying trait (local independence) and that the Item Characteristic Curves (ICCs; see Figure 1) are the same except for their location on the latent trait scale (Schmidt & Embretson, 2012). Other assumptions are unidimensionality, equal discrimination across items, and the absence of successful guessing. Some argue that the Rasch model also assumes that the latent ability measured by the study is quantitative in nature (Michell, 2008), although this is debated (Borsboom & Mellenbergh, 2004).

Rasch model estimates. The Rasch model item features used in the SVMs of the current study were item difficulty, standard error, standardized infit, and standardized outfit, which are described below. These estimates are commonly used, easily obtained, and may provide important information in detecting item compromise. It is important to note that the interpretation of many of these estimates may be influenced by the presence of compromised items and examinees with pre-knowledge. For more information on how these estimates may perform in cases of compromised items and examinees with pre-knowledge, see Thomas et al. (2016).

Item difficulty. Item difficulty (β_{Rasch}) represents the location of an item on the logit scale. More positive numbers represent relatively more difficult items and more negative numbers represent relatively less difficult items. The total number of correct items is a sufficient statistic for estimating item difficulty in the Rasch model because no other information is necessary to obtain item difficulty estimates. The scale of measurement for the items is often scaled such that the mean item difficulty is zero.

Item standard error. The standard error of an item indicates the precision level associated with the item difficulty estimate for that item. Higher standard errors indicate lower precision and lower standard errors indicate higher precision. Item standard error can be expressed as:

Equation 2

$$SE_i = \frac{1}{\sqrt{\sum(P_{ni}\,(1 - P_{ni}))}}$$

where SE_i represents the standard error of item *i* and P_{ni} represents the probability of a correct response on item *i* by persons with trait level *n*, summed over all trait levels.

Item infit. Item infit indicates the extent to which an individual's responses deviate from their expected responses when the item measure and the individual's estimated trait level are in similar locations on the logit scale (Wright, 1980; Wright & Masters, 1982). In the current study, *z*-standardized (Z-STD) item infit statistics were used. Underfit, indicated by more positive numbers, represents a larger than expected difference between expected and actual responses and overfit, indicated by more negative numbers, represents a smaller than expected difference between expected and actual responses. Standardized infit can be expressed as:

Equation 3

$$ZStdIn_i = \left(v_i^{\frac{1}{3}} - 1\right)\left(\frac{3}{q_i}\right) + \left(\frac{q_i}{3}\right)$$

where *ZStdIn*$_i$ represents the *z*-standardized infit for item *i*, v_i represents the weighted mean square for item *i*, and q_i represents the standard deviation of the mean squared infit for item *i* (Wright & Masters, 1982). Mean squared item infit can be expressed as:

Equation 4

$$v_i = \sum_{n}^{N} w_{ni}\, z_{ni}^2 / \sum_{n}^{N} w_{ni}$$

where v_i is the weighted mean square for item *i*, the persons range from *n* to *N*, z_{ni}^2 represents the standardized residual squared for person *n* on item *i*, and w_{ni} represents the variance of the score residuals for all persons *n* on item *i*.

We hypothesize that item infit will outperform most of the other item features in the Support Vector Machines, because it contains more nuanced information than many of the item features. Infit gives more sophisticated information by accounting for a person's estimated ability level in the individual responses, which allows the unexpectedness of the response to be measured. The mismatch of the expected versus actual responses measured by item infit may be particularly useful in detecting test fraud. For example, suppose that an examinee with pre-knowledge has a high estimated trait level. The examinee's estimated ability may be high largely because of their previous exposure to compromised items in situations in which the majority of items are compromised, the examinee studied the compromised items thoroughly, and the examinee used their knowledge of the compromised items on the test to answer most of the compromised items correctly. Accordingly, if a compromised item with a slightly higher difficulty level than the examinee's trait level is administered to them, they may respond correctly, even though the expected probability of correct response suggests that the examinee should respond incorrectly. Item infit captures such patterns of unexpected responses across the entire sample of examinees and may therefore be valuable in identifying items with suspicious patterns of responses. However, the literature on properties that distinguish uncompromised from compromised items is minimal, so nine item features were tested to allow empirical testing of the relative importance of these features.

Item outfit. Item outfit indicates the extent to which an individual's responses deviate from their expected responses when an individual answers items that are far from the individual's trait level on the latent dimension measured (Wright, 1980; Wright & Masters, 1982). In the current study, *z*-standardized (Z-STD) item outfit statistics were used. As in the case of item infit, underfit is indicated by more positive numbers and represents a larger than expected difference between expected and actual responses and overfit is indicated by more negative numbers and represents a smaller than expected difference between expected and actual responses. Standardized outfit can be expressed as:

Equation 5

$$ZStdOut_i = \left(u_i^{\frac{1}{3}} - 1\right)\left(\frac{3}{q_i}\right) + (\frac{q_i}{3})$$

where *ZStdOut$_i$* represents the *z*-standardized outfit for item *i*, *u$_i$* represents the unweighted mean square for item *i*, and *q$_i$* represents the standard deviation of the mean squared outfit for item *i* (Wright & Masters, 1982). The unweighted mean square can be expressed as:

Equation 6

$$u_i = \sum_{n=1}^{N} z_{ni}^2/N$$

where u_i is the unweighted mean square for item *i*, the persons range from *n* = 1 to *N*, and z_{ni}^2 represents the standardized residual squared for person *n* on item *i*.

We hypothesize that item outfit, like item infit, will outperform most of the other item features in the Support Vector Machines. Item outfit captures complex information regarding the mismatch of the expected versus actual responses when items and persons are far apart on the logit scale. Suppose an examinee with pre-knowledge who has a high estimated trait level is administered a difficult compromised item that is located far above their estimated trait level. They may respond correctly even though the expected probability of correct response suggests that the examinee should respond incorrectly. Item outfit captures such patterns of unexpected responses across the entire sample of examinees and may prove valuable in identifying items with suspicious patterns of responses.

The Three-Parameter Logistic Model

The three-parameter logistic model (3PL, Birnbaum, 1968) is an IRT model that estimates the probability of a person responding to an item in a certain category (here, with a correct response), given their latent trait level (θ_{3PL}), the difficulty of the item (3PL Diff), the discrimination of the item (3PL Discr), and the lower limit of the item (3PL Lower) (see Equation 7). The 3PL, like the Rasch model, estimates item and person parameters on the same logit scale. The difficulty of the item (3PL Diff) is the location of the item on the logit scale. However, the Item Characteristic Curves (ICCs) (see Figure 2) in the 3PL framework may have varying slopes (discrimination) and lower asymptotes other than zero (lower limits), unlike the Rasch model ICCs. The 3PL can be represented as:

Equation 7

$$P\left(X_{ij} = 1 \middle| \theta_j, \beta_i,\ \alpha_i,\ \gamma_i\right) = \gamma_i + (1 - \gamma_i) \frac{\exp[\alpha_i(\theta_j - \beta_i)]}{1 + \exp[\alpha_i(\theta_j - \beta_i)]}$$

where X_{ij} is the response to item *i* by person j, θ_j represents the latent trait score for person j, β_i represents the difficulty of item i, α_i represents the discrimination for item i, and γ_i represents the lower limit for item *i*.

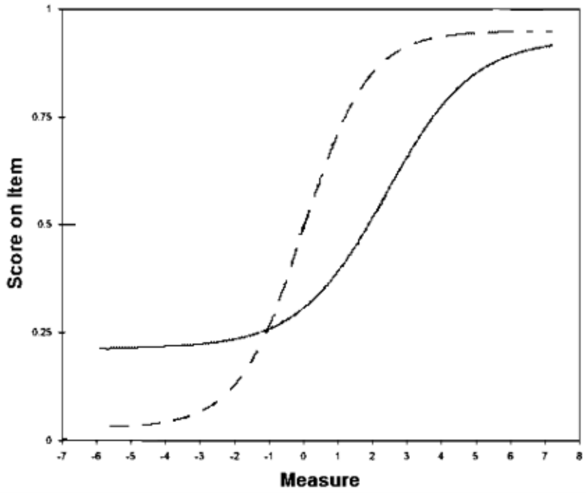

Figure 2. 3PL model Item Characteristic Curves (ICC). Note that the two items (represented by dashed and solid lines) have different slopes (the item represented by the dashed line has a higher discrimination parameter) and that the item represented by the solid line appears to have a non-zero lower limit.

Three-Parameter Logistic Model assumptions. The 3PL model assumes that items are unrelated to each other except through the underlying trait (local independence) and has the assumption of unidimensionality. Unlike the Rasch model, the 3PL makes no assumptions about equal discrimination or the absence of successful guessing, as indicated in the parameters of the 3PL model.

Three-Parameter Logistic Model estimates. The 3PL model item features used in the SVMs of the current study were item difficulty, discrimination, and lower limit, which are described below. It is important to note that the interpretation of many of these estimates may be influenced by the presence of compromised items and examinees with pre-knowledge; see Thomas et al. (2016) for additional information.

Item difficulty. The difficulty of the item (β) is the location of the item on the logit scale. More positive numbers represent relatively more difficult items and less positive numbers represent relatively less difficult items. The 3PL model item difficulties are not expected to equal the Rasch item difficulties because of the inclusion of discrimination and the lower limit parameters in the 3PL model. Thus, the number of correct responses is not a sufficient statistic for estimating item difficulty in the 3PL model.

Item discrimination. Item discrimination (*α*) indicates the slope of an ICC (see Figure 2; Lord & Novick, 1968; Embretson & Hershberger, 1999) and represents how well an item distinguishes between low and high ability examinees. Highly discriminating items are represented by steeper ICC curves indicating that they are good at distinguishing low ability examinees from high ability examinees. Items with poor discrimination (very low, or negative) should be revised or deleted, according to Embretson and Hershberger (1999).

Item lower limit. Item lower limits (*c*; Birnbaum, 1968) represent the lower bounds of the ICC for a single item (see Figure 2; Embretson & Hershberger, 1999). For example, if the lower part of an ICC curve asymptotes above 0.00, the probability of getting that item correct remains higher than 0.00, regardless of the estimated ability level. The lower limit parameter is sometimes called a lower asymptote or the pseudo-guessing parameter. It is called the pseudo-guessing parameter because the lower limit is not necessarily the probability of a single answer choice being correct (e.g. .25 for a multiple choice item with four possible answers). For example, examinees may be able to rule out distractor response options before guessing, leading to a lower limit parameter of greater than .25.

The Rasch Model versus the Three-Parameter Logistic Model. The primary theoretical difference between the Rasch model and the 3PL is that the Rasch model does not allow the latent trait score of an examinee to be influenced by the characteristics of the items in the sample beyond item difficulty, but the 3PL does (as does the 2PL by Lord & Novick, 1968). For instance, in the Rasch model framework, two people with identical raw scores get identical latent trait estimates. If Examinee A and Examinee B both get 45 of 50 items correct, they will receive the same latent trait score. In the 3PL framework, the latent trait scores of Examinee A and Examinee B depend on which of the 45 items they answered correctly and the characteristics of those items (difficulty, discrimination, and lower limit). If the 45 items that Examinee B answered correctly were more highly discriminating items than the 45 items Examinee A answered correctly, Examinee B's latent trait score will be higher, holding item difficulty and lower limit constant. Remember that the 2PL and the 3PL do not achieve the property of double cancellation and therefore do not meet the criteria for objective measurement according to some psychometricians (Embretson & Reise, 2000). Hence, in the 2PL and 3PL models, the probability of a correct response for a particular pair of items ordered by difficulty but varying by discrimination could switch positions based on the order of trait level, which is not logical. In the current study, both Rasch and 3PL model estimates were used as item features because both may provide useful information about item compromise.

Item Response Times

In addition to investigating how Rasch and 3PL model estimates perform as features in SVMs predicting item compromise, item response time will be investigated by including average response time per item it as an additional feature. Thomas et al. (2016)

suggest that examinees with pre-knowledge may have response patterns that are identifiable by their item response times and correctness of item responses. Examinees with pre-knowledge may respond more quickly to suspected compromised items than to other items, because they were previously exposed to these items, and thus may spend less time reading and responding to them (Marianti, Fox, Avetisyan, Veldkamp, & Tijmstra, 2014). In contrast, examinees without pre-knowledge may respond more slowly to suspected compromised items than those with pre-knowledge. Thus, item response times may be instrumental for detecting item compromise. The benefits of including item response time will be assessed by comparing the relative weight, or contribution, of each item feature in the SVMs. If item response time does not perform as well as the other item features in the SVMs, that information is important as well, as it implies that the other item features are better predictors of item compromise than item response times.

Yen's Q_3 Estimate of Local Dependence

Local independence is an assumption of the Rasch model, the 3PL model, and Support Vector Machines. However, in the case of item compromise, this assumption may be violated because examinees who viewed compromised content may have more responses in common with each other than can be explained by the latent trait alone. Therefore, in the current study, Yen's Q_3 statistic (Yen, 1984) for local dependence (LD) was calculated to use as an item feature that quantifies violations of this assumption. Yen's Q_3 provides a pairwise estimate of the relationship between the residuals of two items across all persons, controlling for the latent trait, and can be expressed as:

Equation 8

$$Q_{3_{jk}} = rd_j d_k$$

where ${Q_3}_{jk}$ represents the Q_3 coefficient between item j and item k, d represents the residual of item j or item k depending on the subscript, and r represents the correlation between them. Q_3 estimates range from -1 to 1.

Once calculated, Q_3 estimates can be interpreted in a number of ways. Yen (1993) states that Q_3 values of +/- .20 or greater indicate local dependence. However, Chen and Thissen (1997) found that using a cutoff of .20 for Q_3 estimates leads to lower power than other methods. De Ayala (2008) suggests that because the Q_3 is a correlation coefficient, it can be squared to produce values similar to R^2 that indicate the extent of local dependence. Squared Q_3 values of .05 or greater, representing five percent or more shared variability between items, may indicate local dependence (De Ayala, 2008). In the current study, however, the SVMs require that each item feature contains one value per item (e.g. one item difficulty estimate per item), so the pairwise Q_3 estimates needed to be condensed in order to use the Q_3 as an item feature. Therefore, the Q_3 estimates for each item were represented as a single summary statistic. The range (Max - Min) of Q_3 estimates was used as an item feature because the Q_3 is an estimate of local dependence between items and more extreme Q_3 estimates in either direction indicate strong relationships between items and therefore increased local dependence. The mean or standard deviation were not used because they are less useful for looking at extremes, as they would take into account the entire distribution of Q_3 estimates. Additionally, the range of Q_3 estimates for each item is unique and non-redundant, whereas other summary statistics that capture extremes, such as the minimum and maximum, are more likely to be redundant between items because of the pairwise nature of Q_3 estimates. For example, the maximum Q_3 for Item A might also be the maximum Q_3 for Item B if the highest Q_3

for both items is the *Q₃* for Item A with Item B. Note that the maximum for Item A and Item B is not necessarily redundant, as either item might have a higher Q_3 estimate with another item.

Linear Logistic Test Model

The linear logistic test model (LLTM; Fischer, 1973) incorporates item stimulus features into the estimation of item difficulty. For example, suppose there are two items (Item A and Item B) on an English test. Suppose that the vocabulary in Item A is more challenging than the vocabulary in Item B, but Item B contains more complex syntax than Item A. The LLTM incorporates the knowledge of the item stimulus features into the estimates of item difficulty. The LLTM can be represented as:

Equation 9

$$P(X_{is} = 1|\theta_s, \tau_k) = \frac{\exp(\theta_s - \sum_k \tau_k q_{ik})}{1 + \exp(\theta_s - \sum_k \tau_k q_{ik})}$$

where X_{is} represents a response by person *s* to item *i* (here, a correct answer), θ_s represents the trait level estimate for person *s*, τ_k represents the weight of stimulus factor *k* in item difficulty, q_{ik} represents the value of stimulus factor *k* in item *i*, and $\sum_k$ represents the sum of τ_k multiplied by q_{ik} over all of the stimulus factors (Embretson & Reise, 2000).

In the current study, the item designs were unknown due to the proprietary nature of the data. However, the status of items as suspected compromised or suspected uncompromised was an unplanned item stimulus feature that may have impacted the item difficulty estimates. Thus, the LLTM was used to assess the effect of suspected compromised status on item difficulty. The outcome measures for this analysis were the LLTM difficulty estimates, because they represent the effect of the item status (suspected compromised or suspected uncompromised) on the item parameter, i.e. how item compromise influences item difficulty. However, the LLTM assumes that the researcher has defined in the model a set of features that contribute to item difficulty for the items. The difficulty of the items probably cannot be explained entirely by the status of the items as compromised or uncompromised and no additional information about the nature of the items was provided, so the results of the LLTM should be interpreted cautiously.

Item Features Summary

The item features calculated to classify suspected compromised and suspected uncompromised items into predicted compromised and predicted uncompromised items were: Rasch item difficulty, standard error, infit, and outfit, 3PL difficulty, discrimination, and lower limit, average item response times, the range of Yen’s (1984) Q_3 pairwise estimates for each item, and LLTM difficulty estimates. The relative importance of these item features in the SVMs was assessed. Throughout this paper, the item features will be referred to by the abbreviations listed in Table 1.

Table 1

Item Feature Abbreviations

Item Feature	Abbreviation
Rasch Difficulty	R Diff
Rasch Standard Error	R SE
Rasch Infit	R Infit
Rasch Outfit	R Outfit
3PL Difficulty	3PL Diff
3PL Discrimination	3PL Discr
3PL Lower Limit	3PL Lower
Q_3 Range	Q_3 Range
Average Item Response Time	Avg Time
LLTM Difficulty	LLTM Diff

Note. This table contains the abbreviations for each item feature that will be used in tables and plots.

Support Vector Machines

Support vector machines (SVMs) are analyses that classify distinct classes of scores in such a way that the classes are separated by the maximum possible margin of space (Cortes & Vapnik, 1995). This is accomplished by using a decision boundary, a line, plane, or hyperplane (depending on the dimensionality of the data) that divides the classes, and a margin, the space between the decision boundary and the nearest points (see Figure 3A). The data points that are nearest to the decision boundary and are on the margin are called *support vectors*. SVMs typically operate on subsets of the data; for instance, there is generally a subset of the data designated as a *training set* and the remainder of the data is designated as a *test set*. The optimal locations of the decision boundary and margin are selected based on a training set of data, which consists of feature scores and class membership. Then, the decision boundary and margin are used to classify a test set of data and the accuracy of the classifications made by the SVM on the test set is assessed. Linear SVMs are used for linearly separable data, which are data with a clear, linear division between data points belonging to one class and another (see Figure 3A), and non-linear SVMs are used for non-linearly separable data, data with no clear, linear division between data points belonging to one class and another (see Figure 3B). Note that non-linearly separable data does not indicate a lack of separability in all situations, as the data may be separable using non-linear SVMs. SVMs are often utilized in research paradigms that involve classification, regression, and novelty detection (Bishop, 2006, p. 325). They have the desirable property that any local solution for parameters is the global optimum for convex functions, i.e. any solution of parameters is equivalent to the best solution of parameters in a particular dataset (Bishop, 2006, p. 325). For additional information on SVMs see Cortes & Vapnik (1995), Burges (1998), Cristianini and Shawe-Taylor (2000), and Müller, Mika, Rätsch, Tsuda, and Schölkopf (2001).

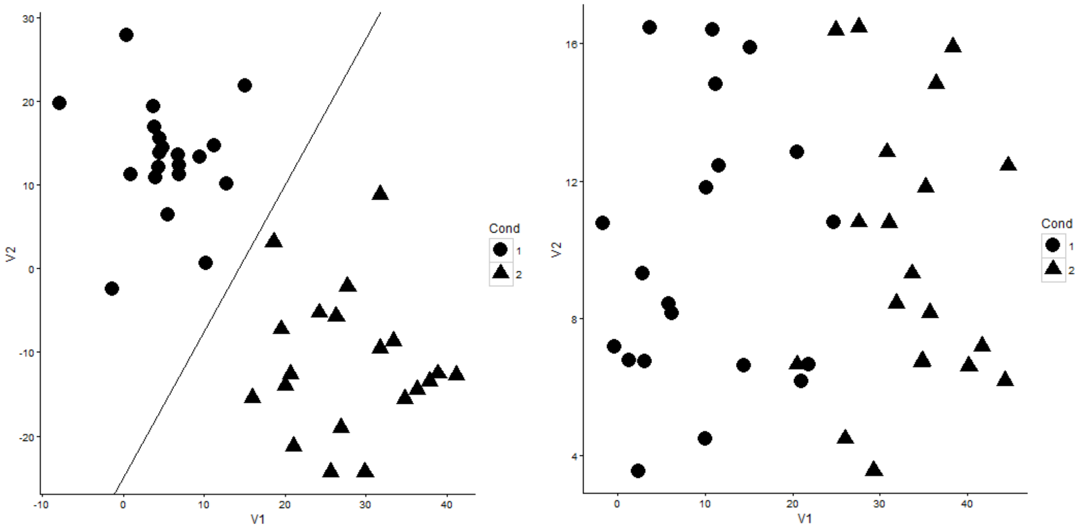

Figure 3. Linear separability of data. A case of linearly separable (left) and non-linearly separable data (right).

The steps to running an SVM analysis can be represented as follows:

1. Separate the data into a training set and a test set
2. Scale the data
3. Choose a kernel (similarity function that operates on pairs of data points)
4. Choose parameters
5. Train the SVM on the training set
6. Test the SVM on the test set and assess the performance of the SVM

Specific information on how these steps were accomplished in the current study is to be found in the SVM Analyses part of the Data Analyses section. Many of the steps listed above can be combined in modern software implementations of SVM analysis. For example, some functions are able to combine steps one, five, and six using cross-validation. An example of cross-validation is *K*-fold cross validation, which splits the data into *k* equal subsamples and then trains the SVM on *k*-1 of the *k* subsamples, using the last subsample as a test set. This is repeated *k* times until every subsample has been used as a test set once (Mosteller & Tukey, 1968; Stone, 1974). The purpose of cross-validation is to avoid a situation in which an SVM performs very well on the training set, but does not generalize well to other sets of data drawn from the same population, such as a test set or validation set. This problem is called overfitting and it can decrease the accuracy of the SVM. The opposite problem, underfitting, occurs when the SVM does not perform very well on the training data or on the test or validation set. Both overfitting and underfitting increase classification errors and decrease the accuracy of the classifier in the test set. Cross-validation is used to approximate the performance of an SVM outside of the training set in the case where there is not a separate validation set. In the current study, 10-fold cross validation was used as described in the SVM Analyses part of the Data Analyses section.

SVM assumptions. The assumptions of SVM are independent and identically distributed data, which are sometimes collectively called the i.i.d. or IID assumption (Dundar, Krishnapuram, Bi, & Rao, 2007). The assumption of independent data requires that the responses of one participant are unrelated to the responses of another. Identically distributed data are typically obtained in SVM by standardizing or scaling of the variables in the training set then using those same values to standardize or scale the variables in the test set (Hsu, Chang, & Lin, 2003).

Choosing a kernel. A kernel is a similarity function that operates on pairs of data points (Hofmann, Schölkopf, & Smola, 2008). Kernel methods take the features of the data, sometimes called the input space, and map them into higher dimensional feature space. The dimensionality of the feature space depends on the number of support vectors. In other words, kernels are a mapping function that transforms the input data into a higher dimensional feature space. Kernel methods are computationally efficient because they do not require actual computation of the data coordinates in the feature space, which can be very high in dimensionality, unlike other methods (Burges, 1998). Kernel functions compute the inner products between pairs of data in the feature space, which allows for much faster computation (Burges, 1998). There are many types of kernels; for example, linear, radial basis function (RBF), polynomial (POLY), and sigmoid kernels.

When conducting an SVM analysis, it is important to select a kernel that is appropriate for the data. For example, if the data are linearly separable, such as the data in Figure 3A, a linear kernel would probably be most appropriate. If the data are not linearly separable, the most appropriate kernel would depend on the characteristics of the data. Generally, a good kernel to begin with is the radial basis function (RBF) kernel (Hsu, Chang & Lin, 2003), but other kernels may be used, depending on the data. The RBF is a good first kernel choice because it can handle non-linearly separable data but the linear kernel is actually a special case of the RBF kernel (Keerthi and Lin, 2003). Thus, separation in linearly separable or non-linearly separable data can be attained using the RBF kernel. The polynomial (POLY) kernel maps the input data into a polynomial feature space. Thus, data that were non-linearly separable in the input space may become linearly separable in the feature space. For example, suppose one class of data is surrounded by another in two-dimensional input space. If a two-degree polynomial kernel were used (i.e. a quadratic polynomial), then the classes could be linearly separable in the feature space as the classes would be mapped onto a higher-dimensional quadratic shape. The class of data points previously surrounded by the other class could now be higher or lower than the other class of data points, allowing the classes to be separated. In the current study, the RBF kernel and the POLY kernel were utilized to attempt to classify suspected compromised and suspected uncompromised items.

SVM parameters. SVMs require that the researcher set various parameters, depending on the type of kernel that is used. For both the radial basis function (RBF) kernel and the polynomial kernel (POLY), researchers must set the C and γ (gamma) parameters, which are defined below (Meyer, Dimitriadou, Hornik, Weingessel, & Leisch, 2015). The POLY kernel also requires that a degree parameter be set. The degree parameter indicates if the mapping of the data into higher dimensional space is linear, quadratic, cubic, quartic, etc. To choose the best value for these parameters, researchers generally vary each parameter through a range of values and assess which value of the parameter performs best. This process is called parameter tuning and is completed using a separate data set for validation or using cross-validation on the training set. (Shawe-Taylor & Cristianini, 2004, p. 220).

C parameter. *C,* sometimes referred to as cost or a slack variable, is a penalizing parameter that balances complexity of the resulting model with the frequency of error (Cortes & Vapnik, 1995). *C* parameters that are too large can cause overfitting to the training set but *C* parameters that are too small can cause underfitting to the training set (Alpaydin, 2004, p. 224). The best value for the C parameter depends on the data (Cherkassy & Ma, 2004).

ɤ ***parameter.*** The ɤ, or gamma, parameter controls the influence of a single data point in the training set (Pedregosa et al., 2011). Low ɤ parameters indicate that a data point can have a large range of influence and high ɤ parameters indicate that a data point can have a small range of influence. Thus, *C* and ɤ both affect model complexity and accuracy.

SVM Outcomes. First, a feature selection algorithm was used to assess the relative contributions of each feature to the SVMs and to investigate which item features have the highest weights in the SVM model variants and are therefore considered the most important item features for classification. Second, the accuracy of the resulting classifications was assessed in terms of overall accuracy, sensitivity, specificity, balanced accuracy, and positive and negative predictive values, which are described in detail below.

The overall accuracy (ACC) will be calculated as:

$$ACC = \frac{True\ Positives + True\ Negatives}{N}.$$

where *N* represents the total number of cases (True Positives + True Negatives + False Positives + False Negatives).

Sensitivity is the proportion of actual positive cases that are classified as positive cases (e.g. suspected compromised items that are classified as predicted compromised) (see Table 2). Sensitivity (Altman & Bland, 1994a) can be expressed as:

$$Sensitivity = \frac{True\ Positives}{True\ Positives + False\ Negatives}.$$

High sensitivity is desirable because it represents a low false negative (type II error) rate.

Specificity is the proportion of actual negative cases that are classified as negative cases (e.g. suspected uncompromised items that are classified as predicted uncompromised) (see Table 2) (Altman & Bland, 1994a). Specificity can be expressed as:

$$Specificity = \frac{True\ Negatives}{True\ Negatives + False\ Positives}.$$

High specificity is desirable because it represents a low false positive (type I error) rate.

Table 2

Possible Outcomes of a Two Group Classification Problem

	True Condition: Positive	True Condition: Negative
Test Result: Positive	True Positive	False Positive
Test Result: Negative	False Negative	True Negative

Note. This table shows a matrix of possible outcomes in classification analyses for a two class problem.

Balanced accuracy was developed to serve as a measure of accuracy when the class sizes are unequal (Brodersen, Ong, Stephan, & Buhmann, 2010). Unlike overall accuracy, it accounts for imbalanced data sets that have disparate base rates, or prevalence, for each class. For example, overall accuracy may show that a classifier is performing at high levels of accuracy, but the classifier may simply select the most prevalent class for every classification (Brodersen, Ong, Stephan, & Buhmann, 2010). Balanced accuracy accounts for this and can be expressed as:

$$Balanced\ Accuracy = \frac{Sensitivity + Specificity}{2}\ .$$

Positive predictive value (PPV) (Altman & Bland, 1994b) represents the proportion of positive test results that are actually true positive cases. That is, given a positive test result, the ratio answers the question, how likely is it that the case is a true positive? PPV can be expressed as:

$$PPV = \frac{True\ Positives}{True\ Positives + False\ Positives}\ .$$

The PPV represents the probability that a positive test result actually represents a positive case, so a high PPV is desirable.

Negative predictive value (NPV) (Altman & Bland, 1994b) is the counterpart of PPV and represents the proportion of negative test results that are actually true negative cases. If a given test result is negative, the ratio answers the question, what is the probability that the case is a true negative? NPV can be represented as:

$$NPV = \frac{True\ Negatives}{True\ Negatives + False\ Negatives}\ .$$

The NPV represents the probability that a negative test result actually represents a negative case, so a high NPV is desirable.

SVM Model Variants

In addition to investigating the performance of SVM models attempting to classify suspected compromised and suspected uncompromised items using the radial basis function (RBF) kernel and the polynomial (POLY) kernel, the effects of two other variables were investigated, sample size and uncompromised to compromised item split. Sample size (*N*) was varied over five values; starting from the original sample size of 13,584 down to a sample size of 200 (see Table 3). Random selection of participants continued until the desired sample size was reached for each sample size. Then the item features were calculated for each set of randomly selected participants, yielding item feature estimates for each sample size. The item features for any given *N* were calculated from the same participants as every other model variants with that *N*. In other words, the 200 participants used to calculate the item features in the RBF, *N* = 200, .25/.75 split model variant were the same as the 200 participants used to calculate the item features in the POLY, *N* = 200, .75/.25 split model variant and in every other model variant where *N*

= 200. The split of suspected uncompromised to suspected compromised items in the data was also varied. The original data had 61 suspected uncompromised items and 86 suspected compromised items, yielding a split of .41 to .59. However, we also wanted to investigate cases in which each class of items was much more prevalent than the other, so we data sets were created with suspected uncompromised to suspected compromised item splits of .25/.75 and .75/.25. These splits are labelled Split 1, Split 2, and Split 3, as shown in Table 3. Split 1 was created by solving the following set of equations:

Equation 10

$$\frac{x}{N} = .25 \text{ where } x \leq 61 \text{ and } \frac{y}{N} = .75 \text{ where } y \leq 86.$$

In Equation 10, *N* represents the total number of items (147, in this case), *x* represents the number of suspected uncompromised items, and *y* represents the number of suspected compromised items. The stated conditions of $x \leq 61$ and $y \leq 86$ were used because there were only 61 suspected uncompromised items and 86 suspected compromised items in the original data that could possibly be selected. This set of equations was solved in such a way that allowed retention of the maximum possible number of items. The solution for Split 1 was to create a dataset with 29 suspected uncompromised items and all 86 suspected compromised items (115 total items).

Split 3 was created by solving the following set of equations:

Equation 11

$$\frac{x}{N} = .75 \text{ where } x \leq 61 \text{ and } \frac{y}{N} = .25 \text{ where } y \leq 86.$$

In Equation 11, *N* represents the total number of items (147, in this case), *x* represents the number of suspected uncompromised items, and *y* represents the number of suspected compromised items. The solution for Split 3 was to create a dataset with all 61 suspected uncompromised items and 20 suspected compromised items (81 total items). These solutions gave us the desired item splits while retaining the maximum possible number of items. Items were randomly selected items until the desired number of suspected compromised items was reached. Similar to the consistency of the sample size variants within level, the items selected for any given split were the same for all model variants with that split. In other words, the 29 suspected uncompromised items in the RBF, *N* = 200, Split 1 model variant were the same as the 29 suspected uncompromised items in the POLY, *N* = 500, Split 1 model variant and in every other model variant with Split 1. All model variants are crossed with the others leading to a total of 30 models (2 x 5 x 3) (see Table 3). The item features were the same for every model variant: Rasch item difficulty, standard error, infit and outfit, 3PL difficulty, discrimination, and lower limit, average item response time, the range of Q_3 estimates, and LLTM difficulty estimates.

Table 3

SVM Model Variants

Kernels (2)	*N* (5)	Uncompromised to Compromised Split (3)
	200	
RBF	500	Split 1: .25 / .75
POLY	1,000	Split 2: .41 / .59 (ORIG)
	5,000	Split 3: .75 / .25
	13,584 (ORIG)	

Note. This table shows all of the SVM model variants investigated in the current study. Properties of the original data are denoted as “ORIG.”

Hypotheses

Of the item features in this study, we hypothesize that item infit and item outfit will have the highest feature weights in the SVMs. Item infit and outfit capture complex information regarding the mismatch of the expected versus actual responses, as a function of the distance between an item’s difficulty location and a person’s expected response. Item infit and outfit statistics capture patterns of unexpected responses across the entire sample of examinees and may prove valuable in identifying items with suspicious patterns of responses.

We also hypothesize that the Rasch model estimates will have higher feature weights than the 3PL estimates. Thus, we expect Rasch item difficulty, standard error, infit, and outfit to be more important (i.e. have higher feature weights) in the classifications than 3PL item difficulty, discrimination, and lower limit. This hypothesis is based on the results of Thomas et al. (2016) who found that Rasch item difficulty, mean-squared infit, and mean-squared outfit outperformed indices of discrimination and lower limit.

Our final hypothesis is that the 3PL item discrimination will have a higher feature weight than the 3PL lower limit in distinguishing between suspected compromised and suspected uncompromised items. This hypothesis is also based on the results of Thomas et al. (2016), who found that items classified using traditional cutoffs for discrimination were more accurate than items classified using traditional cutoffs for the lower limit.

Summary of the Introduction

The primary purpose of this study is to pioneer a new exploratory data analysis procedure that uses a combination of Rasch and Item Response Theory models with Support Vector Machines. In the current study, this procedure was used to classify items into predicted compromised items (items predicted to be compromised in notes and/or screenshots) and predicted uncompromised items (items not predicted to be compromised in notes and/or screenshots) based on their item properties (Rasch model: difficulty, standard error, infit, and outfit, 3PL: difficulty, discrimination, and lower limits, average item response times, the range of Q_3 estimates of local dependence, and LLTM difficulty estimates). The weights of the item features in the SVMs will be examined to determine the features that are the most important predictors of item compromise. We hypothesize that item infit and outfit will have the highest feature weights, that Rasch model estimates will have higher feature weights than the 3PL estimates, and that the 3PL item discrimination will have a higher feature weight than the 3PL lower limit. The accuracy of the SVMs was investigated in terms of overall accuracy, sensitivity, specificity, balanced accuracy, and positive and negative predictive values to assess the performance of the SVMs in classifying predicted compromised and predicted uncompromised items. Additionally, the models were varied in terms of which kernel was used (RBF or POLY), the sample size, and the split of suspected uncompromised to suspected compromised items in the data. Although this procedure was used for detecting suspected compromised items in the current study, the hybrid method combining Rasch and Item Response Theory models with Support Vector Machines is flexible and could be used in a broad array of contexts.

Method

Participants

The participants were 13,584 examinees from 423 schools who took an international healthcare certification exam. The participants were degree holders representing five different degrees. There were 13,508 examinees who took the exam once, 74 examinees who took the exam twice, and two examinees who took the exam three times, resulting in a total of 13,662 exams. Data from exam re-takes were excluded to meet the assumption of local independence, leaving only data from examinees' first exam sittings (*N* = 13,584 exams).

Data

Minimal information on the data provided because of the proprietary nature of the test. Item level data consisted of scored item responses, item order, and item response times in seconds. There were 160 items of various types, but data from all items that were not multiple choice, single answer items were excluded, leaving 147 items for analysis. Other item types were excluded to eliminate noise in the item features. For example, the average item response time might be different for multiple choice, multiple answer items than it would be for multiple choice, single answer items, which could cause the results to be influenced by unimportant differences in item types rather than showing meaningful differences in items reflecting their suspected compromised or suspected uncompromised status. Person level data (*N* = 13,584) consisted of a participant ID, testing session ID, school ID, and degree type. See Table 4 for a small example of the data including many of the key item features for this study and Appendix A for the full dataset of item features (*N* = 13,584, Split 2).

Table 4

Example Data Set

Item	R Diff	R SE	R Infit	R Outfit	3PL Diff	3PL Discr	3PL Lower	Q_3 Range	Avg Time (secs)	Suspected Item Category 1= Comp
1	0.80	0.02	0.48	0.92	1.96	0.45	0.14	0.09	50.00	1
102	0.87	0.02	-0.70	0.09	2.51	0.68	0.25	0.16	75.59	0

Note. This table shows two example rows of data for the current study.

The publisher of this certification exam discovered screenshots and notes containing confidential test content. This created a situation in which there were two sets of items; items the test publisher discovered in screenshots and/or notes, called *suspected compromised* items, and items that the test publisher did not discover in screenshots and/or notes, called *suspected uncompromised* items. Of the 147 multiple choice, single answer items, 86 were suspected compromised items and 61were suspected uncompromised items. The researchers were given this information regarding the frequencies of item classes at the beginning of the study.

Data Analyses

Data Cleaning and Preparation

As mentioned in the data section above, items that were not multiple choice, single answer items were excluded, leaving 147 items in the data, and exam re-takes were excluded to meet the assumption of independent data, leaving N = 13,584 participants and exams.

Rasch Model Analyses

The software program, Winsteps (Linacre, 2016), was used to obtain Rasch model estimates on the dichotomously scored items. Estimation in Winsteps (Linacre, 2016) is completed using the joint maximum likelihood estimation (JMLE) procedure. JMLE allows for missing data and uses all of the available information to produce estimates (An & Yung, 2014). The mean item difficulty was scaled to zero.

3PL Model Analyses

The 3PL model was estimated using the “rasch.mml2” function in the “sirt” package of R (Robitzsch, 2015). The 3PL was estimated using the marginal maximum likelihood (MML) estimation procedure. The argument “center.b=TRUE” was used to scale the difficulty parameters such that the mean item difficulty was zero. Due to estimation challenges, the starting values of the difficulty, discrimination, and lower limit parameters were set to the values of the Winsteps estimate or index for each parameter to achieve convergence. Additionally, the minimum and maximum difficulty were set to negative eight and positive eight, respectively.

Item Response Times

Average item response times were calculated using the “colMeans” function in the “base” package in R (R Core Team, 2016) on the columns of item response times (in seconds). No data was excluded when calculating the average item response times, which means that any outliers could influence the results. It was believed that these outliers might contain meaningful information that could help distinguish suspected compromised items from suspected uncompromised items. However, if average item response time does not perform well as an item feature in the current study, that could indicate a high level of noise in the item response time data, which could be addressed in future studies by removing extreme item response times before predicting suspected item compromise.

Yen's *Q_3* Analyses

Yen's (1984) Q_3 estimates of local dependence were obtained using the "yen.q3" function in the "sirt" package of R (Robitzsch, 2015). The "yen.q3" function uses parameter estimates from the "rasch.mml2" function, also in the "sirt" package, which yields estimates that are identical to Winsteps when the argument "center.b=TRUE" (mean item difficulty equal to zero) is used (Robitzsch, 2015).

LLTM Analysis

The linear logistic test model was run using the "LLTM" function in the "eRm" package of R (Mair, Hatzinger, & Maier, 2015). The *Q*-matrix of stimulus factors was a vector of 1s and 0s indicating the specified and known levels of distinction among these items, that is, the status of each item as suspected compromised or suspected uncompromised. No other information was known on the design specifying the complexity of the items, due to their proprietary nature.

SVM Analyses

All SVM analyses were conducted in R (R Core Team, 2015), using the "svm" function in the "e1071" package (Meyer et al., 2015). The type of classification used was *C*-classification. *C*-classification differs from the other types of SVM models (e.g. nu-classification, epsilon-SVM regression, or nu-SVM regression) in the formulation of the error term that is minimized by the SVM (StatSoft, Inc., 2013). For more information on other types of SVM models, see StatSoft, Inc. (2013). The argument "cross=10" was used in the "svm" function of the "e1071" package to achieve 10-fold cross validation.

Scaling. Hsu, Chang, and Lin (2003) showed that scaling the data prevents variables with larger ranges from dominating variables with smaller ranges. The variables were standardized using the default “scale=TRUE” command in the “svm” function (Meyer et al., 2015). This option scales the variables to a mean of zero and unit variance.

Kernels. The radial basis function (RBF) kernel was investigated, as recommended by Hsu, Chang, and Lin (2003), as was the polynomial (POLY) kernel. The kernels were specified using the ‘kernel = “radial”’ and ‘kernel = “polynomial”’ arguments of the “svm” function in the “e1071” package of R, respectively. For more information on the parameters needed for each kernel see the Parameter Tuning section below.

Parameter tuning. Cross-validation was used to select the best values for the SVM parameters. The argument “cross=10” was used in the “tune.svm” function of the “e1071” package to achieve 10-fold cross validation during the parameter tuning process. The parameters that are used in an SVM depend on the type of kernel. The RBF kernel and the POLY kernel both require C and γ (gamma) parameters. The POLY kernel also requires a degree parameter. The “tune.svm” function in the “e1071” package was used for parameter tuning (Meyer et al., 2015). This function uses a grid search over ranges of parameters provided by the researcher. There is minimal literature regarding what ranges should be searched to obtain SVM parameters, so fairly wide parameter ranges were used in the parameter tuning. The “tune.svm” function in the “e1071” package uses overall accuracy to calculate the classification errors (Meyer et al., 2015). Parameters with the smallest errors are considered the best parameters. Multiple parameter tunings were used in an attempt to improve the consistency and quality of the chosen parameters. The first parameter tuning process tuned over a range of 2^{-4} to 2^{4} for γ and a range of 2^{-4} to 2^{4} for C (and a range of one to four for the degree parameter in the POLY kernel). Then the best 30 unique sets of parameters were selected and tuning occurred over those parameters again. This tunes over a maximum of 900 parameter combinations for the RBF kernel (30 γ x 30 C). Once again the best 30 unique sets of parameters were selected and tuning occurred over those parameters again. Then, the best combination of parameters identified by the final tuning were used as the parameters in the SVMs.

Tuning after the first tune cycle was limited to the best 30 unique sets of parameters for several reasons. The first is that choosing the number of best unique parameters sets using a constant value, rather than selecting a proportion of the results (e.g. selecting the best half of results), prevents the tuning process from growing exponentially in the number of parameter sets to tune over and helps to keep the time necessary for the tuning process feasible. Suppose, for example, that the best half of unique sets of parameters were selected, and the original tune provided 300 parameter sets. The best half should be about 150 parameter sets. The second tuning process would then cross every value of the 150 gamma parameters with every value of the 150 cost parameters (and with every value of the degree parameters for the POLY kernel), resulting in 22,500 parameter sets (150 x 150) to tune over for the second tune for the RBF kernel. If the best half of the results from that tuning were selected, there would be over 11,250 sets of parameters for the third tune. This exponential growth of the parameter tuning process when selecting a proportion of the results is problematic in terms of the time and computing power necessary to complete the tuning process. The number 30 was chosen somewhat arbitrarily as the constant value of parameter sets to select because it seemed large enough to capture a variety of the best performing parameter sets, but limited the number of parameter sets to tune over, and the time required for the tuning process to reasonable amounts. However, other values (e.g. 40, 50, 60, etc.) could have been chosen as the number of parameter sets to select. These constraints on the selection of tuning parameters were utilized to make the tuning process more computationally and temporally manageable.

SVM Outcomes. An implementation of the Support Vector Machine Recursive Feature Extraction (SVM-RFE) feature selection algorithm (Guyon, Weston, Barnhill, & Vapnik, 2002) in the “e1071” package in R (Meyer et al., 2015) was used to assess the feature weights across the model variants**.** The calculation of the SVM outcomes was conducted in R, using true positive, false positive, true negative, and false negative values needed for the calculation of overall accuracy, sensitivity, specificity, balanced accuracy, and positive and negative predictive values from the prediction table obtained with the function “table” (“base” package; R Core Team, 2016) applied to the results of the “predict” function (“e1071” package; Meyer et al., 2015) and the suspected item categories.

Results

Item Feature Estimates

The item feature estimates for the original data of N = 13,584 with a suspected uncompromised to suspected compromised split of .41/.59 are presented below. The item features will be referred to by the abbreviations listed in Table 1.

Rasch model. The Rasch model was fit to the data to obtain estimates of item difficulty, standard error, standardized infit, and standardized outfit. Descriptive statistics on the item features, including the Rasch model estimates, for the original data of *N* = 13,584 with a suspected uncompromised to suspected compromised split of .41/.59, can be seen in in Table 5. In general, the suspected uncompromised items were easier than the suspected compromised items, had a slightly larger range of standard errors, had lower infit estimates, and slightly lower outfit estimates. The standardized infit and outfit estimates for both suspected item categories had large ranges, indicating misfit between the actual and expected responses on the items.

Table 5

Item Features for Suspected Item Categories

	Suspected Uncompromised				Suspected Compromised			
	M	*SD*	Min	Max	*M*	*SD*	Min	Max
R Diff	-.19	.86	-2.45	2.85	.14	.86	-1.83	1.80
R SE	.02	.01	.02	.05	.02	.00	.02	.03
R Infit	.23	2.79	-7.41	8.81	.46	4.11	-9.90	9.90
R Outfit	.20	3.62	-9.28	8.87	.18	4.53	-9.90	9.90
3PL Diff	-.77	2.70	-8.00	8.00	.55	2.47	-4.22	8.00
3PL Discr	.50	.24	.16	1.31	.52	.21	.05	1.05
3PL Lower	.10	.14	.00	.48	.13	.15	.00	.52
Q_3 Range	.10	.03	.06	.16	.10	.05	.05	.38
Avg Time	56.31	22.88	24.12	169.50	54.49	10.92	26.68	79.82

Note. Descriptive statistics for the item features separated by suspected item category.

3PL model estimates. The 3PL model was fit to the data to obtain estimates of item difficulty, discrimination, and the lower limit. Model level fit statistics indicated that the 3PL model fit the data better than the Rasch model, indicating that the items in the data exhibited varying discriminations and lower limits (see Table 6). However, the results of the Rasch model should not be discounted because of this, as the Rasch model is a measurement model for the development of scales that allow stable inferences to be drawn (Linacre, 1996). The failure of the data to fit the Rasch model implies that the data displays undesirable qualities for objective measurement.

The results of the 3PL model indicated that the suspected uncompromised items were, on average, easier than the suspected compromised items and had a larger range of item difficulty estimates (see Table 5). The average and variability of discrimination parameters was comparable across suspected item categories, but the suspected uncompromised items had a slightly larger range of discrimination parameters than the suspected compromised items. The suspected uncompromised items were similar to the suspected compromised items in terms of the average, variability, and range of lower limit parameters.

Table 6

Model Fit for Rasch and 3PL Models

	Rasch	3PL
AIC	2,242,588	2,233,936
BIC	2,243,700	2,237,251

Note. Model level fit statistics for the Rasch (left) and 3PL (right) models.

Yen's Q_3. The pairwise Q_3 estimates of local dependence in the original data (N = 13584, .41/.59 split) ranged from Q_3 = - .09 to Q_3 = .32 (M = -.01, SD = .02) with 31.51% positive Q_3 values and 68.49% negative Q_3 values. Only one of the 10,731 calculated pairwise Q_3 estimates was more extreme than Yen's (1993) recommended cutoff of .20 or had a squared value above .05, indicating that local dependence was not problematic in this data set as assessed by traditional cutoff scores. The Q_3 ranges for the suspected uncompromised items was largely equivalent to the Q_3 ranges for the suspected compromised items, but the suspected uncompromised items did exhibit slightly lower variability in Q_3 range (see Table 5).

Item response times. The raw, un-averaged response times in the original data (N = 13,584, Split 2) ranged from 0 seconds to 360,006 seconds (M = 55.24, SD = 422.61). This wide range could suggest suspicious test taking behaviors. The average item response times ranged from about 24 seconds to almost three minutes (see Table 5). The suspected uncompromised items exhibited a larger variability in average response times and a higher maximum average response time than the suspected compromised items.

Linear logistic test model. The LLTM was fit to the data to obtain estimates of the difference in item difficulty between suspected uncompromised and suspected compromised items. The results of the LLTM indicated that the suspected compromised items were about 0.31 logits more difficult than the suspected uncompromised items, η = -.31, SE_η = .003. However, recall that the LLTM assumes that the researcher has defined every important feature that contributes to item difficulty for the items, but the difficulty of the items probably cannot be explained entirely by the status of the items as suspected compromised or suspected uncompromised. Otherwise, all 86 suspected compromised items would have the same predicted item difficulty value, and all 61 suspected uncompromised items would have the same predicted item difficulty value. Thus, the results of the LLTM should be interpreted cautiously. As such, the results of the LLTM were assessed as a separate analysis and LLTM estimates were not used as item features in the SVMs.

Relationships between item features. The correlations between the item features for original data (N = 13,584; Split 2) can be seen in Table 7. Many of the item features were significantly related to one another. For example, standardized infit and outfit were strongly, positively and significantly related to one another in an almost perfect correlation. It is important to note that Table 5 and Table 7 show the descriptive statistics and correlations for item features in the original data (*N* = 13,584; Split 2), but not for the other sample sizes or splits. Thus, the results displayed in Table 5 and Table 7 may not be true for all of the data sets in this study.

Table 7

Correlations between Item Features in Original Data

	R Diff	R SE	R Infit	R Outfit	3PL Diff	3PL Discr	3PL Lower	Q_3 Range
R SE	-.64***							
R Infit	.28***	-.10						
R Outfit	.38***	-.17*	.97***					
3PL Diff	.82***	-.42***	.23**	.22**				
3PL Discr	-.06	.25**	-.59***	-.65***	.29***			
3PL Lower	.28***	-.13	.09	.05	.56***	.44***		
Q_3 Range	.19*	-.10	-.05	-.07	.22**	.16	.07	
Avg Time	.26**	-.26**	.05	.11	.12	-.16	-.08	-.07

Note. *** $p < .001$; ** $p < .01$; * $p < .05$

Support Vector Machine Outcomes

In the following sections, the results of the SVM analyses will be described in terms of the feature weights and the accuracy measures of the models. Recall that the features that have the highest weights in the SVMs represent the most important features in the prediction of item compromise in the current data. The accuracy measure results described below are the overall accuracy, sensitivity, specificity, balanced accuracy, and positive and negative predictive values. The models were varied in terms of which kernel was used (RBF or POLY), the sample size, and the split of suspected uncompromised to suspected compromised items in the data (see Table 3).

All of the SVMs were run under two additional conditions, one with class weights, representing the relative proportion of suspected compromised and suspected uncompromised items in the data, and one without class weights. The class weights were passed to the parameter tuning function and to the SVM function. The results presented below were obtained from the SVMs without class weights, as there appeared to be minimal differences (an average of about six percent difference for accuracy measures) between the SVMs with and without class weights.

Feature weights. We hypothesized that item infit and outfit would be the most important item features (i.e. have higher feature weights than the other features), that Rasch model estimates would outperform the 3PL estimates, and that the 3PL item discrimination would outperform the 3PL lower limit. We investigated these hypotheses by using a feature selection algorithm, assessing the average absolute feature weights, and evaluating the ranks of the features across the models. See Appendix B for the feature weights for all model variants.

The result of the Support Vector Machine Recursive Feature Extraction (SVM-RFE) is a ranked list of the features, with lower numbers meaning higher relative importance (e.g. a feature ranked number one is considered the most important feature in the data). This ranking is determined by the feature weights. According to the SVM-RFE feature selection algorithm, the most important features in this data were the Rasch item difficulty, Rasch item outfit, and Rasch standard error, in that order, as hypothesized. The least important features in the data, according to the SVM-RFE, were the Q_3 range, 3PL item difficulty, and the average response time (see Table 8). However, it is important to note that the implementation of the SVM-RFE algorithm assessed feature importance using a linear kernel. Thus, the results of the feature selection algorithm were not used to limit the item features of the model variants for several reasons. First, the feature selection algorithm used a linear kernel, but the performance of the RBF and POLY kernels were assessed in the current study. The features that are considered best for one kernel may not be the best for another. Thus, there is no guarantee that the best features for a linear kernel would be the best features for either of the kernels investigated. Secondly, if the feature weights for each type of kernel used in the current study (RBF and POLY) were assessed and the features with the highest weights for each were selected, the resulting models would not be comparable as they would probably contain different item features for the RBF kernel than they would for the POLY kernel. Third, the SVMs do not appear to necessitate a reduction in features, so rather than limiting the features prior to analyses, all of the features were retained and their performance was compared.

Table 8

SVM-RFE Results

1	R Diff
2	R Outfit
3	R SE
4	R Infit
5	3PL Discr
6	3PL Lower
7	Avg Time
8	3PL Diff
9	Q_3 Range

Note. This table show the ranks of all the features as calculated by the SVM-RFE feature selection algorithm.

The average absolute value of the feature weights for the RBF kernel model variants (across all item splits and sample sizes) showed that the Rasch standard error, Rasch difficulty, and 3PL lower limit were the three most important features in the data, and that the Rasch outfit, Rasch infit, and Q_3 range were the three least important features in the data (see Table 9). The average absolute value of feature weights for the POLY kernel model variants (across all item splits and sample sizes), replicated the results of the RBF kernel, by also showing that the Rasch standard error, Rasch difficulty, and 3PL lower limit were the most important item features (see Table 9).

Table 9

Average Absolute Feature Weights

	R Diff	R SE	R Infit	R Outfit	3PL Diff	3PL Discr	3PL Lower	Avg Time	*Q*₃ Range
RBF	5.24	7.44	2.12	1.90	3.63	3.59	4.60	2.92	2.27
	(4.99)	(5.93)	(2.14)	(2.65)	(4.47)	(3.59)	(5.45)	(3.17)	(2.80)
POLY	.20	.21	.18	.13	.18	.13	.20	.16	.11
	(.23)	(.27)	(.26)	(.19)	(.23)	(.14)	(.20)	(.16)	(.15)

Note. Higher means indicate higher feature importance. The numbers in parentheses are the standard deviations for the distributions of absolute feature weights.

However, it is also possible that high feature weights in a few model variants could have artificially inflated the average absolute feature weights. To account for this, the rank of the absolute feature weights was computed, from one to nine for every model variant, and then a sum of the ranks was calculated (see Figure 4). In this analysis, lower rank sums indicate higher importance.

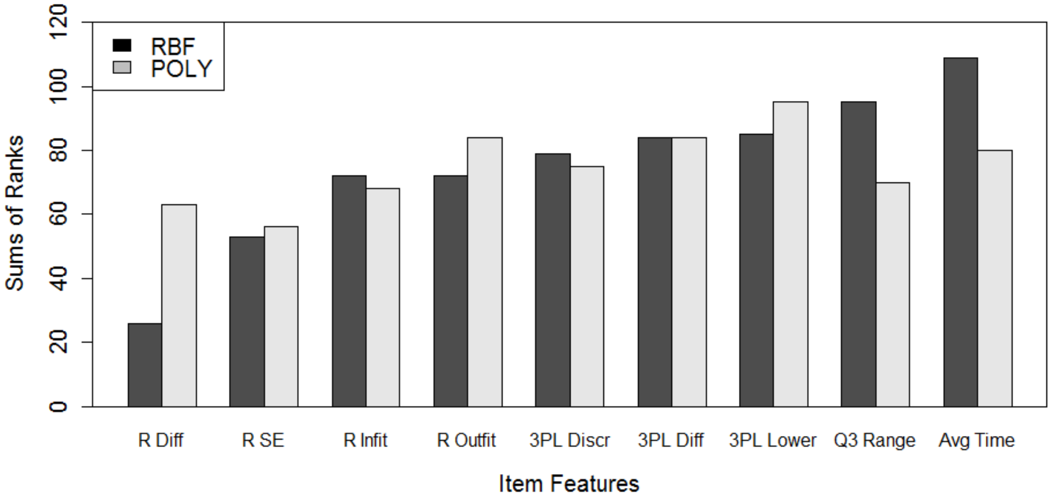

Figure 4. Rank sums for item features. These ranks sums are sorted from smallest to largest for the RBF kernel.

As can be seen in Table 9 and Figure 4, both the average absolute feature weights and the sum of the ranked feature weights suggest that the Rasch item difficulty and the Rasch standard error of the item difficulty were the two most important features for detecting item compromise in the present study in both RBF and POLY kernels, contrary to the hypothesis that Rasch infit and outfit would be the most important item features. The relative importance of the other feature weights did not appear to be as consistent. For example, the average absolute feature weights would suggest that the 3PL lower limit is almost as important as Rasch difficulty and standard error for RBF and POLY kernels (see Table 9), but the rank sums would suggest that 3PL discrimination is the feature that is closest to the importance of the Rasch difficulty and Rasch standard error in RBF and POLY models.

Further inspection revealed that the order of the least important features in the RBF model variants were consistent; Rasch outfit was the least important, then Rasch infit, then Q_3 range, and then average response time (in order of least important to more important). The least important feature in the POLY model variants was consistently the Q_3 range, but the order of the other least important features varied. However, Rasch outfit and the average response time were consistently among the four least important features of the POLY model variants. These findings suggest that Rasch outfit, Q_3 range, and average response times were generally the least important features for classifying suspected compromised and suspected uncompromised items in this data.

The evidence for the support of the hypothesis that the Rasch estimates would outperform the 3PL estimates was mixed. The results of the SVM-RFE showed that the Rasch estimates did outperform all of the 3PL estimates, with Rasch estimates ranked first through forth and 3PL estimates ranked fifth, sixth, and eighth (see Table 8). The results of the average absolute feature weights for the RBF kernel models showed that the Rasch item difficulty and standard error had higher feature weights than the 3PL estimates, but that Rasch infit and outfit had lower feature weights than the 3PL estimates (see Table 9). The results of the average absolute feature weights for POLY kernel models showed that the Rasch standard error had a higher feature weight than the 3PL estimates, but that the feature weights of the other Rasch estimates were tied with the 3PL estimates (see Table 9). Rasch difficulty had the same average absolute feature weight as the lower limit, Rasch infit had the same average absolute feature weight as the 3PL difficulty, and Rasch outfit had the same average absolute feature weight as discrimination. The rank sums for RBF kernel models showed that Rasch item difficulty and standard error had lower rank sums than the 3PL estimates, but Rasch infit and outfit had higher rank sums than the 3PL estimates (see Figure 4). The rank sums for POLY kernel models showed that Rasch item difficulty and standard error had lower rank sums than the 3PL estimates, but that discrimination had a lower rank sum than Rasch infit and outfit and that lower limit had a lower rank sum than Rasch outfit. Thus, there is not enough evidence to conclude that the Rasch estimates outperformed the 3PL estimates.

The evidence for the hypothesis that the 3PL discrimination would outperform the 3PL lower limit was somewhat mixed. The results of the SVM-RFE showed that discrimination outperformed lower limit, with feature ranks of five and six, respectively (see Table 8). The rank sums also showed that discrimination outperformed lower limit, with the sum of ranks for discrimination and lower limit at 72 and 79, respectively, for RBF kernel models, and 68 and 75, respectively, for POLY kernel models (see Figure 4).

However, the average absolute feature weights showed that discrimination was a less important feature than the lower limit, with average absolute feature weights of 3.59 and 4.60, respectively, for the RBF kernel, and average absolute feature weights of .13 and .20, respectively, for the POLY kernel (see Table 9). Further investigation showed that in 19 of the 30 models, the discrimination had a higher feature weight and was ranked lower than the lower limit, indicating that discrimination was a more important feature than the lower limit in those models. Thus, most of the models indicated that discrimination outperformed the lower limit, but not all.

Overall Accuracy. Recall that overall accuracy is simply the number of correctly classified cases divided by the total number of cases. In general, the model variants with the RBF kernel slightly outperformed those with the POLY kernel in terms of overall accuracy, collapsing across sample size and suspected uncompromised to suspected compromised item splits (see Table 10). However, the variability of overall accuracy for models with POLY kernels was smaller than the variability of overall accuracy for models with RBF kernels.

Table 10

SVM Outcomes by Kernel Type

	RBF	POLY
Overall Accuracy	.77 (.12)	.72 (.07)
Sensitivity	.76 (.39)	.68 (.47)
Specificity	.51 (.45)	.40 (.44)
Balanced Accuracy	.64 (.16)	.54 (.02)
PPV	.81 (.17)	.76 (.16)
NPV	.92 (.10)	.92 (.12)

Note. The average outcomes are shown by kernel, with standard deviations of those outcomes in parentheses. Positive predictive value (PPV) and negative predictive value (NPV) are defined in the PPV and NPV sections below.

The effect of sample size was investigated by aggregating the results for each sample size over the kernel type and the item split. In general, there did not appear to be consistent increases across the SVM accuracy outcomes based on sample size (see Table 11). This is surprising, as one might think that accuracy would decrease with lower sample size. This decrease could indicate minimal change among the item features calculated under each sample size. Overall accuracy and balanced accuracy were slightly lower and less variable for sample sizes of 500 and 1,000 compared to the other sample sizes.

Table 11

SVM Outcomes by N

	N = 200	*N* = 500	*N* = 1,000	*N* = 5,000	*N* = 13,584
Overall Accuracy	.76 (.14)	.72 (.08)	.74 (.06)	.78 (.14)	.73 (.11)
Sensitivity	.68 (.50)	.68 (.49)	.67 (.50)	.81 (.38)	.77 (.39)
Specificity	.56 (.48)	.38 (.48)	.45 (.44)	.51 (.43)	.38 (.48)
Balanced Accuracy	.62 (.19)	.53 (.04)	.56 (.07)	.66 (.17)	.57 (.10)
PPV	.80 (.19)	.75 (.16)	.77 (.15)	.82 (.18)	.79 (.18)
NPV	.92 (.13)	.88 (.14)	.91 (.12)	.94 (.09)	.94 (.10)

Note. The average outcomes by sample size, with standard deviations of those outcomes in parentheses. Positive predictive value (PPV) and negative predictive value (NPV) are defined in the PPV and NPV sections below.

The effect of sample size was investigated by aggregating the results for each sample size over the kernel type and the sample size. There did appear to be differences in the accuracy outcomes based on item split (see Table 12), as overall accuracy and balanced accuracy were higher for Split 1 and Split 3 than for Split 2. Additionally, Split 1 and Split 2 had perfect sensitivity on average, with little variance, and Split 3 had perfect specificity on average, with no variance. The nuances of these results will be investigated in the sections below.

Table 12

SVM Outcomes by Item Split

	Split 1	Split 2	Split 3
Overall Accuracy	.81 (.08)	.63 (.04)	.79 (.07)
Sensitivity	1.00 (.00)	1.00 (.01)	.16 (.28)
Specificity	.25 (.32)	.12 (.12)	1.00 (.00)
Balanced Accuracy	.63 (.16)	.56 (.06)	.58 (.14)
PPV	.80 (.08)	.62 (.03)	1.00 (.00)
NPV	1.00 (.00)	.98 (.03)	.79 (.06)

Note. The average outcomes are shown by item split, with standard deviations of those outcomes in parentheses. Positive predictive value (PPV) and negative predictive value (NPV) are defined in the PPV and NPV sections below.

As can be seen from Figure 5, the results of the SVMs with both the RBF and POLY kernels indicate that Split 2 models generally exhibit lower overall accuracy than Split 1 or Split 3 models. This suggests that the SVMs may be more accurate when the classes are more imbalanced in the data, which will be investigated further in the sections below.

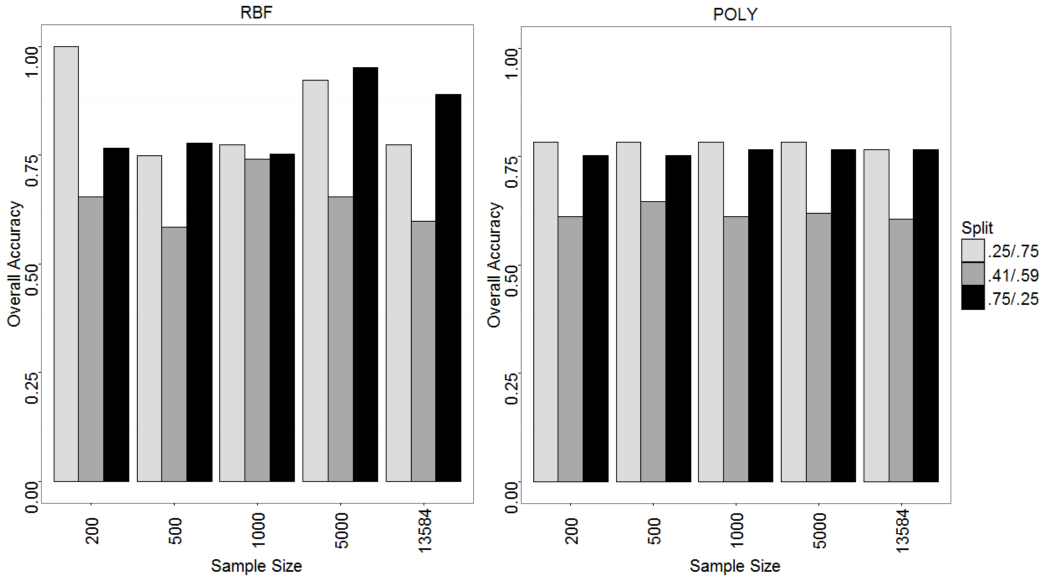

Figure 5. Overall accuracy of SVM models. The results for the RBF kernel are on the left and the results for the POLY kernel are on the right.

Sensitivity and Specificity. Sensitivity is the proportion of actual positive cases (suspected compromised) that are classified as positive cases (predicted compromised). Specificity is the proportion of actual negative cases (suspected uncompromised) that are classified as negative cases (predicted uncompromised). Sensitivity and specificity were generally higher for models with RBF kernels ($Sensitivity_{RBF}$ = .76 and $Specificity_{RBF}$ = .51) than models with POLY kernels ($Sensitivity_{POLY}$ = .68 and $Specificity_{POLY}$ = .40) (see Table 10). The results of the SVMs indicate that in Split 1 and Split 2 models, which are those with a majority of suspected compromised items (75% and 59%, respectively), sensitivity is generally much higher than specificity (see Figure 6). This indicates that these SVMs are classifying a large proportion of the suspected compromised items as predicted compromised, but a very small proportion of the suspected uncompromised items are being classified as predicted uncompromised. Thus, these SVMs seemed to be biased towards classifying items as predicted compromised.

The results of the SVMs for Split 3 models, which are those with a majority of suspected uncompromised items (75%), show the opposite results, indicating that specificity is much higher than sensitivity (see Figure 6). Thus, these SVMs appear to be biased as well, but in the opposite direction of the Split 1 and Split 2 models. The Split 3 models showed apparent bias towards classifying items as predicted uncompromised. These results seem to indicate that the classifications of the SVMs were biased by the relative size of the classes.

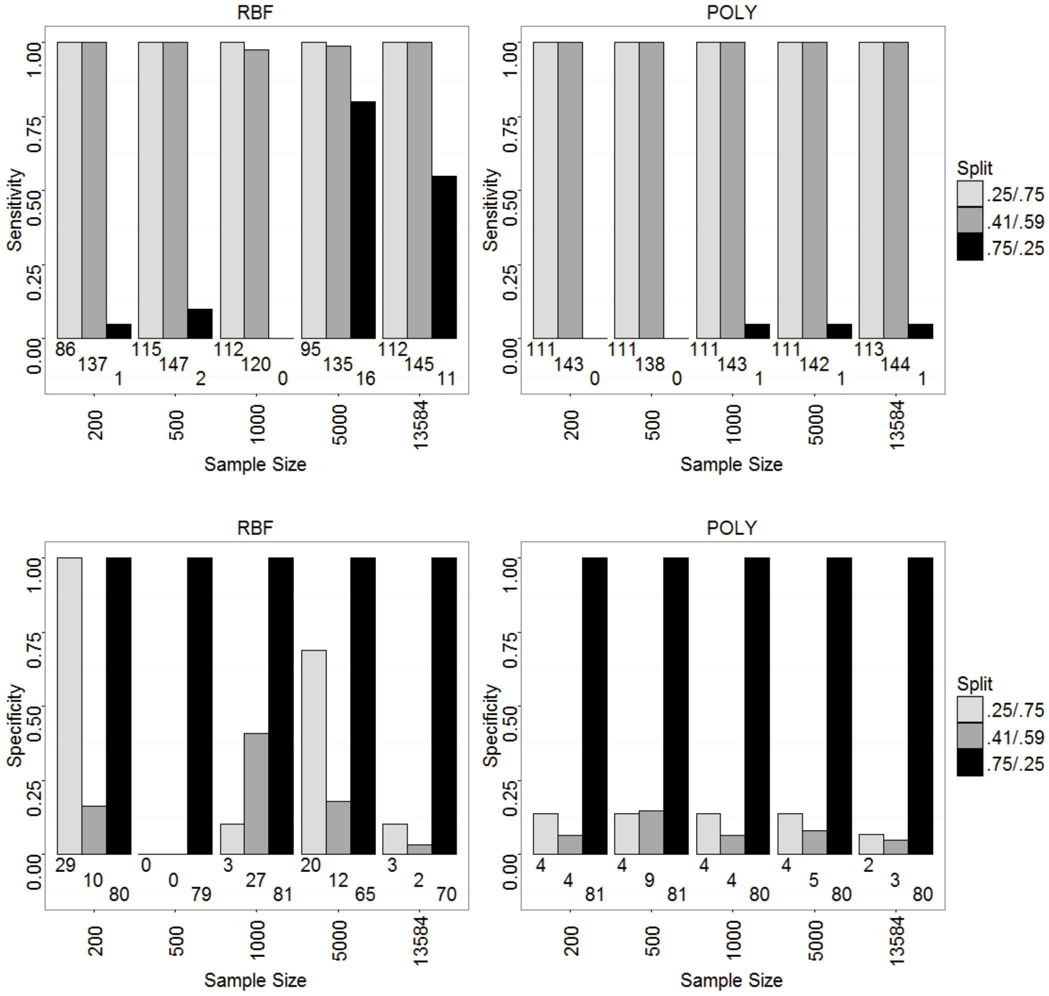

Figure 6. Sensitivity and specificity of SVM models. Sensitivity (top) and specificity (bottom) for the RBF (left) and POLY (right) kernels. Missing bars indicate a value of zero. The numbers under each bar represent the number of items in each model variant that were predicted compromised for sensitivity and predicted uncompromised for specificity.

The average number of items predicted into each predicted item category for each split can be viewed in Table 13. These findings indicated that Split 1 and Split 2 models, on average, classified the majority of the items as predicted compromised, but Split 3 models, on average, classified the majority of the items as predicted uncompromised.

Table 13

Number of Items Classified into Predicted Item Categories

	Average Number Predicted Compromised	Average Number Predicted Uncompromised
Split 1	107.70	7.30
Split 2	139.40	7.60
Split 3	3.30	77.70

Note. For Split 1, there were 29 suspected uncompromised items and 86 suspected compromised items, for Split 2 there were 61 suspected uncompromised items and 86 suspected compromised items and for Split 3 there were 61 suspected uncompromised items and 20 suspected compromised items.

Balanced accuracy. The balanced accuracy is the average of sensitivity and specificity and shows the accuracy of the classifier after accounting for class size. The results show that the balanced accuracies of the models are much lower than the overall accuracy would suggest (see Figure 7). The balanced accuracies were generally higher in models with RBF kernels (64%) than in models with POLY kernels (54%), but the balanced accuracies were also more variable for models with RBF kernels (SD_{RBF} = .16) than for models with POLY kernels (SD_{POLY} = .02) (see Table 10).

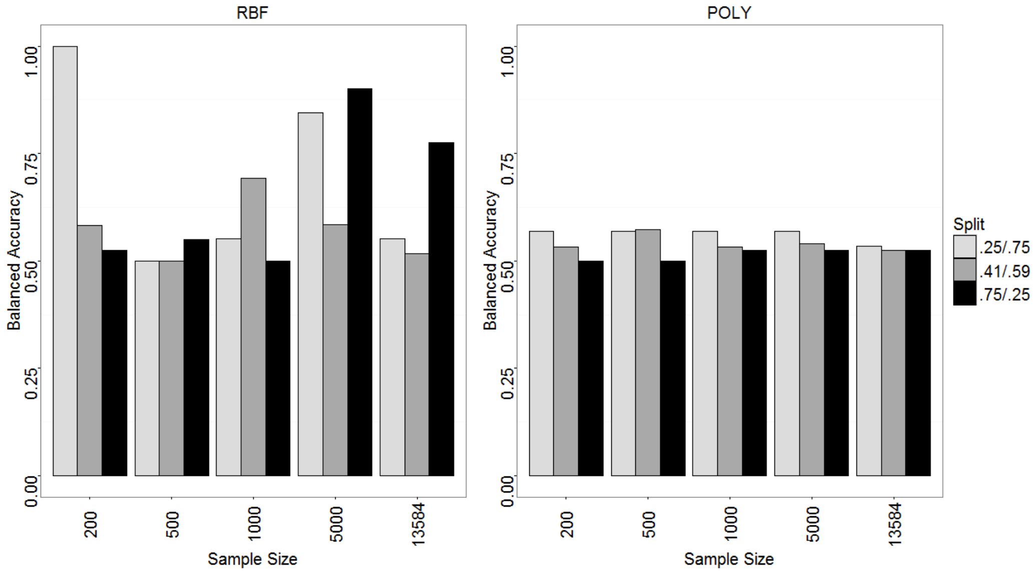

Figure 7. Balanced accuracy of SVM models. The results for the RBF kernel are on the left, the results for the POLY kernel are on the right.

PPV. Positive predictive value (PPV) represents the proportion of positive test results (predicted compromised) that are actually true positive cases (suspected compromised). A PPV of one indicates that every item that was predicted compromised was a suspected compromised item. The average PPV was slightly higher for models with RBF kernels (PPV_{RBF} = .81) than for models with POLY kernels (PPV_{POLY} = .76), but the variability of PPV was equivalent across kernels (see Table 10). PPV was generally higher for the Split 3 model variants than for the Split 1 and Split 2 model variants, with seven of the 10 Split 3 model variants exhibiting a PPV of one (see Figure 8). These PPV results indicate that items that were predicted compromised in the Split 3 models, which were previously found to show apparent bias towards predicting the majority of items as suspected uncompromised, were more likely to actually represent suspected compromised items than the items that were predicted compromised in other models (Split 1 and Split 2). This may be because the items that were classified as predicted compromised in the Split 3 models were those that were most different from the predicted uncompromised items and therefore the predicted compromised items were more likely to represent suspected compromised items.

The observant reader might notice that three Split 3 models (RBF, *N* = 1,000; POLY, *N* = 200; POLY, *N* = 500) exhibited a PPV of zero, which seems to contradict the previously stated idea that when models are biased in one direction, cases classified in the opposite direction are more likely to represent actual differences. However, in these three models, zero items were classified items as predicted compromised, resulting in a PPV of zero. Thus, it is important to note that PPV does not take into account the number of items in each predicted category. For example, the RBF, *N* = 200, Split 3 model classified only one item as predicted compromised and that item was indeed a suspected compromised item, resulting in a PPV of one.

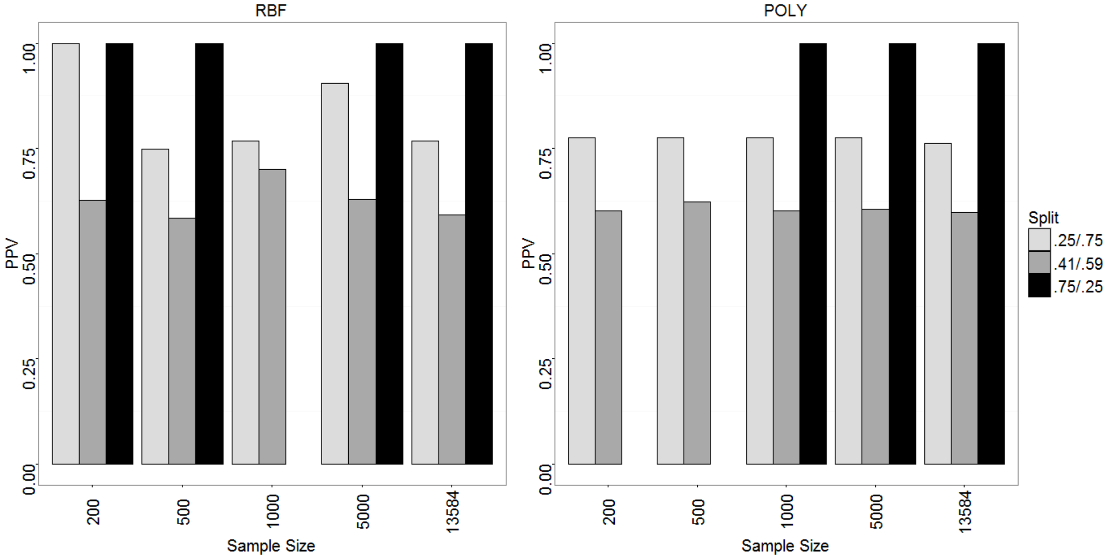

Figure 8. PPV of SVM models. The results for the RBF kernel are on the left, the results for the POLY kernel are on the right. Missing bars indicate a value of zero.

NPV. Negative predictive value (NPV) represents the proportion of negative test results (predicted uncompromised) that are actually true negative cases (suspected uncompromised). The average NPV and the spread of the NPVs was generally equivalent for models with RBF and POLY kernels (see Table 10). NPV was generally higher for Split 1 and Split 2 model variants than for Split 3 model variants, with 16 of the 20 Split 1 and Split 2 models exhibiting a perfect NPV of one (see Figure 9). An NPV of one indicates that every item that was predicted uncompromised was a suspected uncompromised item. The NPV results indicate that items classified as predicted uncompromised in Split 1 and Split 2 models, which were previously found to show apparent bias towards predicting the majority of items as suspected compromised, were more likely to actually represent suspected uncompromised items than items that were predicted compromised in other models (e.g. Split 3). In the two Split 1 or Split 2 models exhibiting an NPV of zero (RBF, *N* = 500, Split 1 and Split 2), no items were classified as predicted compromised. The results of NPV, combined with the results of PPV, seem to suggest that in models that are biased towards a larger class, cases that are classified in the minority class are more likely to represent actual minority cases.

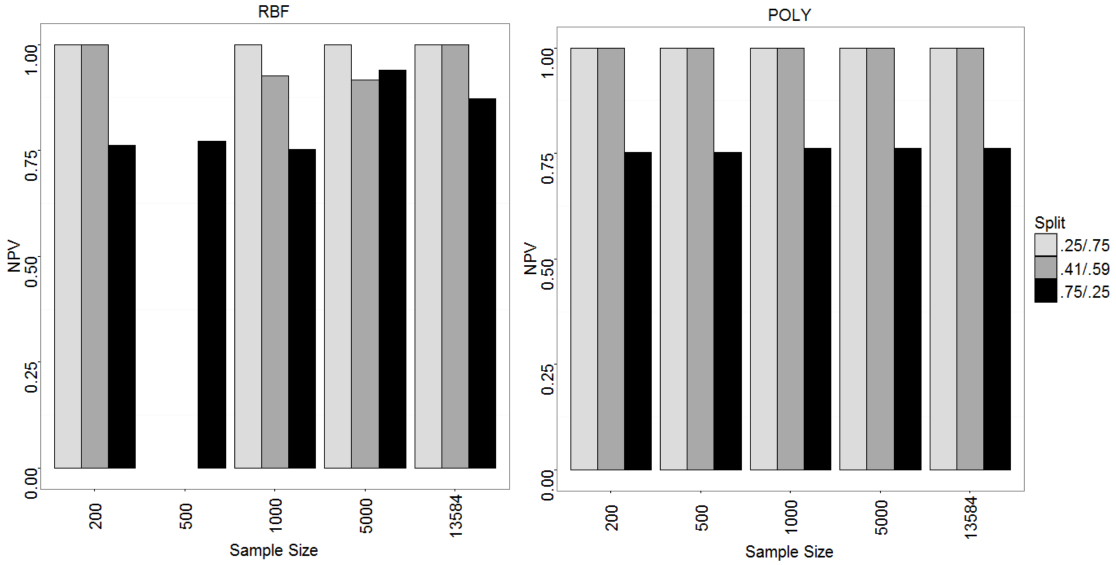

Figure 9. NPV of SVM models. The results for the RBF kernel are on the left, the results for the POLY kernel are on the right. Missing bars indicate a value of zero.

Consistency of Correct Classifications

The accuracy of the SVM models was further investigated in such a way that accounted for the number of items in each predicted category. This was done by calculating the number of times each item was classified correctly across the models. This analysis was conducted for each item split individually as the number of items in the data varied by split. For example, Split 1 had all 86 suspected compromised items and 29 suspected uncompromised items, but Split 3 had 20 suspected compromised items and all 61 suspected uncompromised items. The results for each split were consistent across models with RBF and POLY kernels, so the results in the following section are collapsed over kernel. There were 10 models per split.

The results for Split 1 indicated that all 86 suspected compromised items were correctly classified in all 10 models (see Figure 10). The majority of suspected uncompromised items were classified incorrectly in most of the Split 1 models, but three of the suspected uncompromised items were correctly classified in eight or nine of the models.

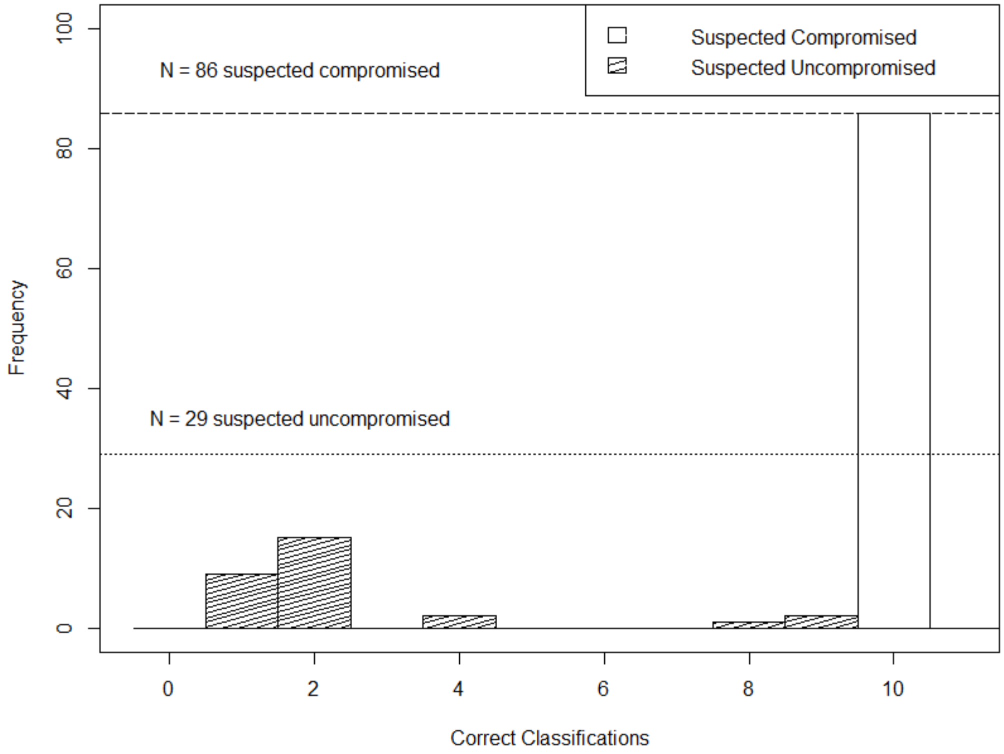

Figure 10. Frequency of correct classifications for Split 1. The *x*-axis represents the number of correct classifications for any given item for the Split 1 models (ranging from zero to 10) and the *y*-axis represents the number of items that fall into those categories.

The results for Split 2 are similar to the results for Split 1, with the suspected compromised items correctly classified most of the time (see Figure 11). However, an increased proportion of suspected uncompromised items were incorrectly classified in all 10 Split 2 models, as compared to Split 1 models. Additionally, the distribution of correct classification of suspected uncompromised items was more spread out, with more items spread across more levels of correct classifications. These findings indicate that in the Split 1 and Split 2 models, the suspected compromised items were generally classified correctly but the suspected uncompromised items were not.

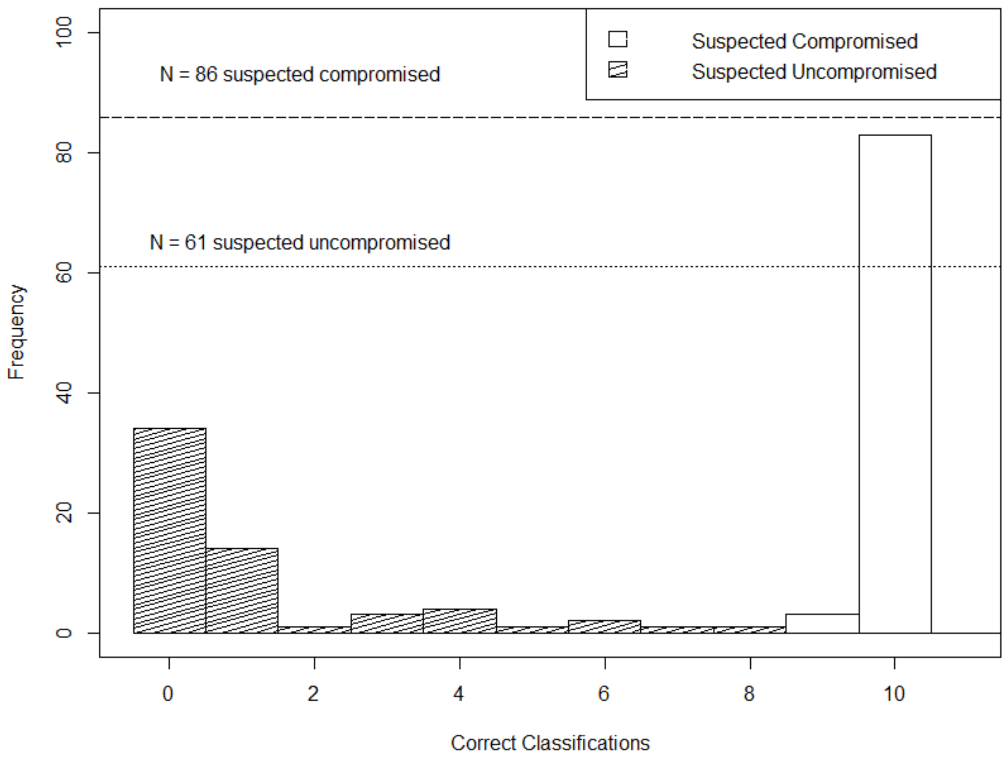

Figure 11. Frequency of correct classifications for Split 2. The *x*-axis represents the number of correct classifications for any given item for the Split 2 models (ranging from zero to 10) and the *y*-axis represents the number of items that fall into those categories.

The results for the Split 3 models show results in the opposite direction of the Split 1 and Split 2 models, with all of the suspected uncompromised items correctly classified in all the models and most of the suspected compromised items incorrectly classified (see Figure 12).

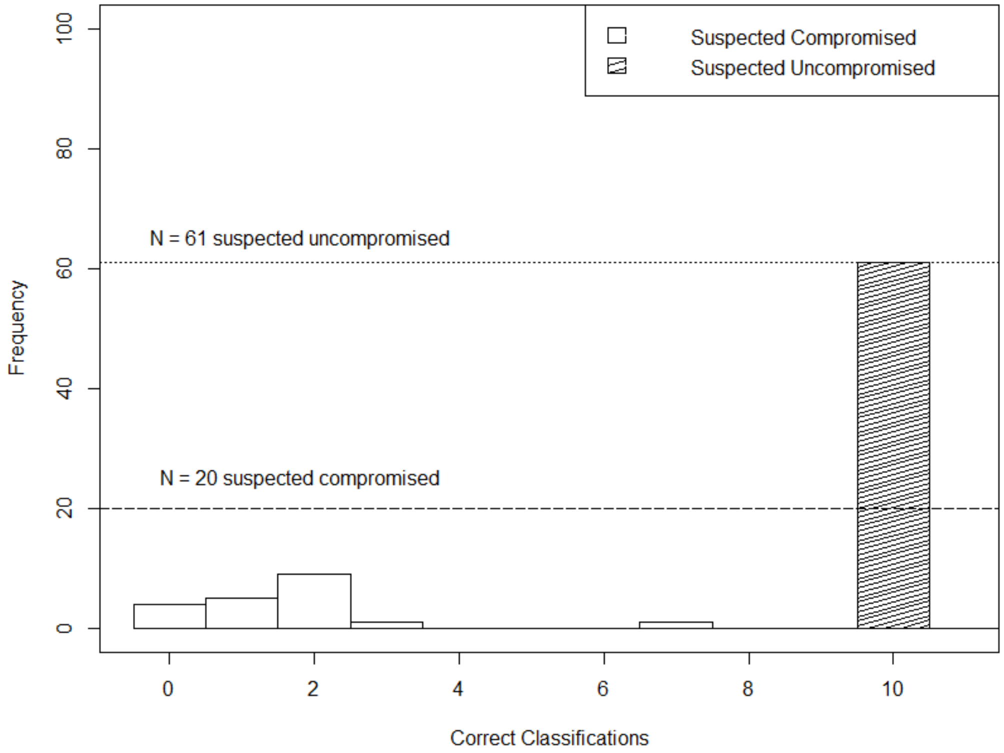

Figure 12. Frequency of correct classifications for Split 3. The *x*-axis represents the number of correct classifications for any given item for the Split 3 models (ranging from zero to 10) and the *y*-axis represents the number of items that fall into those categories.

Table 14 contains the percentage of correct classifications for suspected compromised and suspected uncompromised items for each model variant.

Table 14

Correct Classifications by Suspected Item Category

			Percentage of Correct Classifications	
	N	Split	Suspected Compromised	Suspected Uncompromised
RBF	*N* = 200	1	100%	100%
	N = 500	1	100%	0%
	N = 1000	1	100%	10%
	N = 5000	1	100%	69%
	N = 13584	1	100%	10%
	N = 200	2	100%	16%
	N = 500	2	100%	0%
	N = 1000	2	98%	41%
	N = 5000	2	99%	18%
	N = 13584	2	100%	3%
	N = 200	3	5%	100%
	N = 500	3	10%	100%
	N = 1000	3	0%	100%
	N = 5000	3	80%	100%
	N = 13584	3	55%	100%
	N = 200	1	100%	14%
POLY				
	N = 500	1	100%	14%
	N = 1000	1	100%	14%
	N = 5000	1	100%	14%
	N = 13584	1	100%	7%
	N = 200	2	100%	7%
	N = 500	2	100%	15%
	N = 1000	2	100%	7%
	N = 5000	2	100%	8%
	N = 13584	2	100%	5%
	N = 200	3	0%	100%
	N = 500	3	0%	100%
	N = 1000	3	5%	100%
	N = 5000	3	5%	100%
	N = 13584	3	5%	100%

Note. The percentages are calculated based on the number of items in each suspected category. For Split 1, there were 29 suspected uncompromised items and 86 suspected compromised items, for Split 2 there were 61 suspected uncompromised items and 86 suspected compromised items and for Split 3 there were 61 suspected uncompromised items and 20 suspected compromised items.

Separability of Item Features

To investigate the separability of the item features in the input space, the distributions of item features were plotted for suspected compromised and suspected uncompromised items in the original data (*N* = 13,584; Split 2). The distributions of these features for the suspected item categories were largely overlapping, indicating that these features were not separable in the raw data (see Figures 13-21).

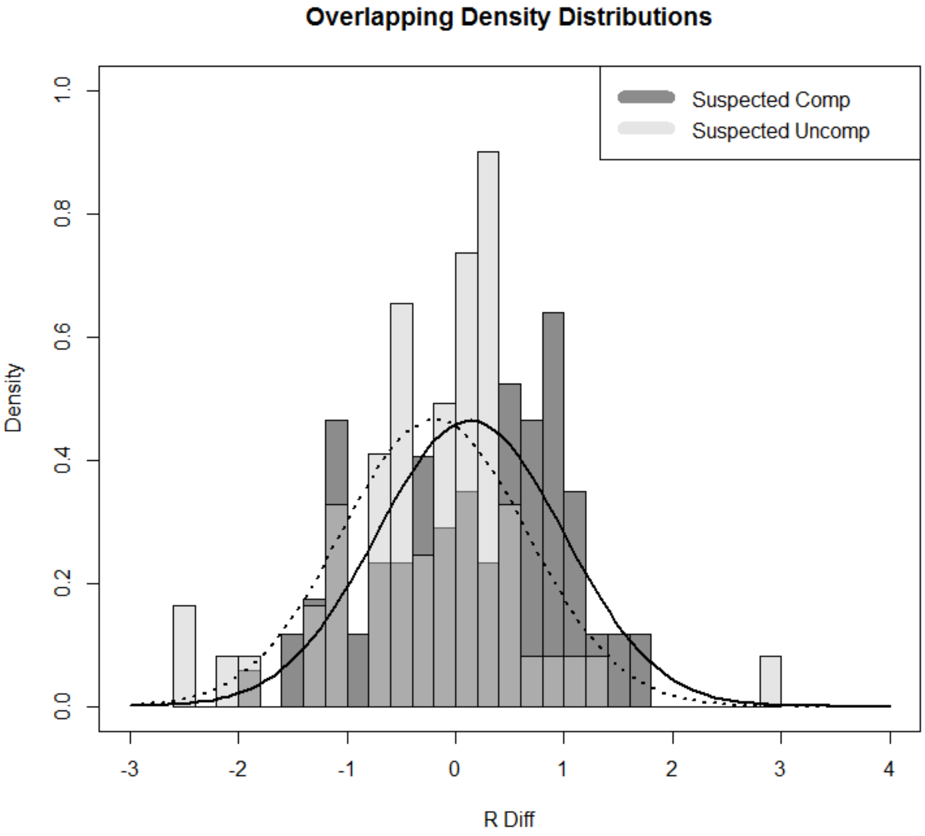

Figure 13. Rasch difficulty density distributions. The density distributions for suspected compromised (dark grey) and suspected uncompromised items (light grey). Medium grey hues indicate overlap of the distributions. The lines represent a normal distribution around the suspected compromised items (solid) and the suspected uncompromised (dashed) items.

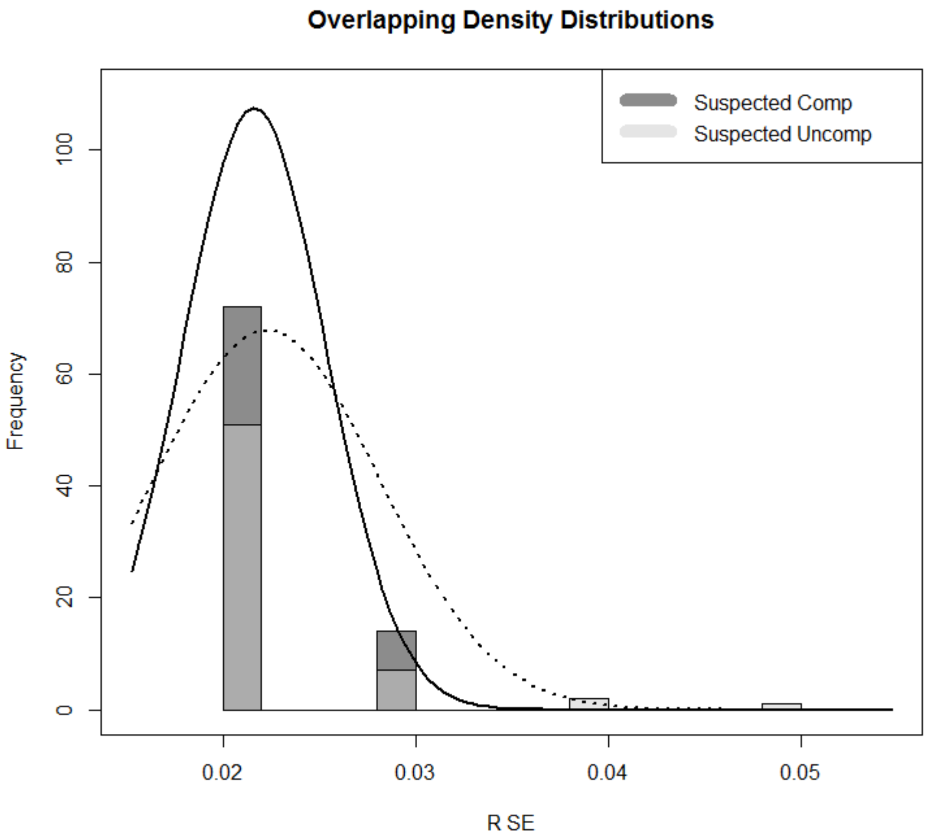

Figure 14. Rasch standard error density distributions. The density distributions for suspected compromised (dark grey) and suspected uncompromised items (light grey). Medium grey hues indicate overlap of the distributions. The lines represent a normal distribution around the suspected compromised items (solid) and the suspected uncompromised (dashed) items.

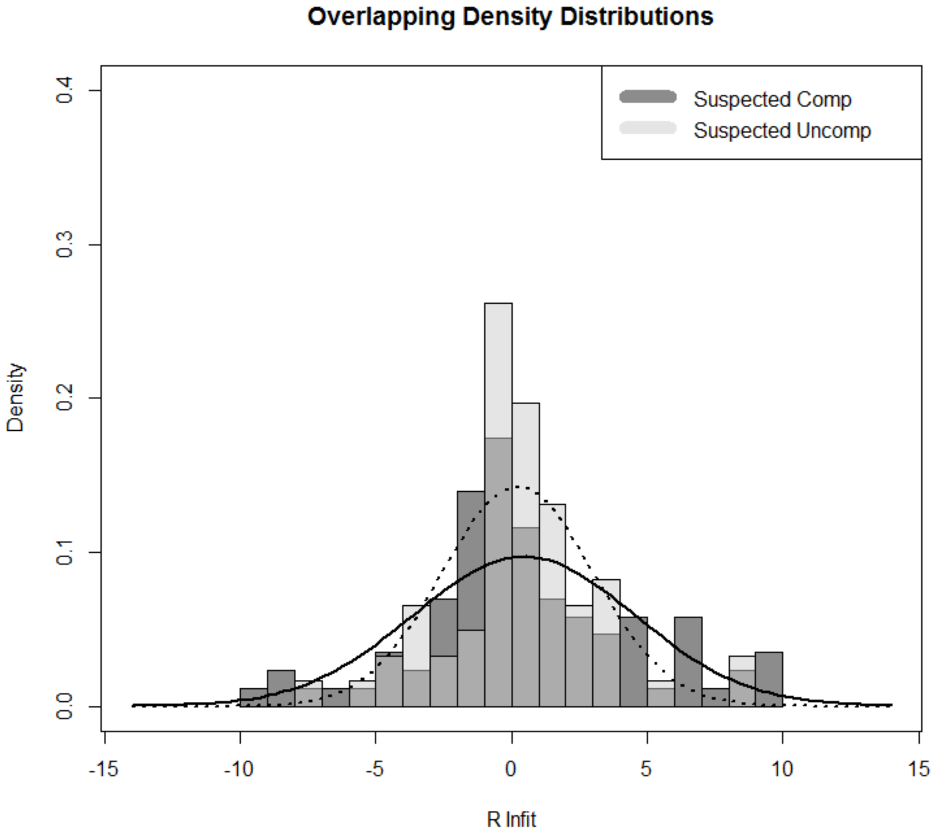

Figure 15. Rasch infit density distributions. The density distributions for suspected compromised (dark grey) and suspected uncompromised items (light grey). Medium grey hues indicate overlap of the distributions. The lines represent a normal distribution around the suspected compromised items (solid) and the suspected uncompromised (dashed) items.

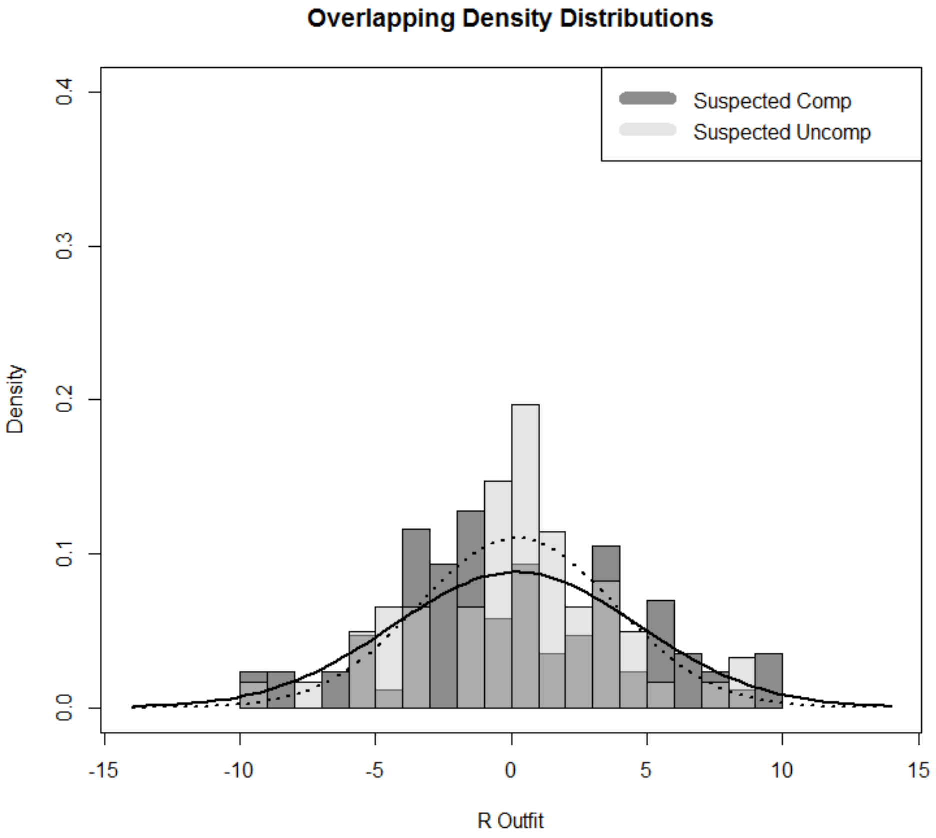

Figure 16. Rasch outfit density distributions. The density distributions for suspected compromised (dark grey) and suspected uncompromised items (light grey). Medium grey hues indicate overlap of the distributions. The lines represent a normal distribution around the suspected compromised items (solid) and the suspected uncompromised (dashed) items.

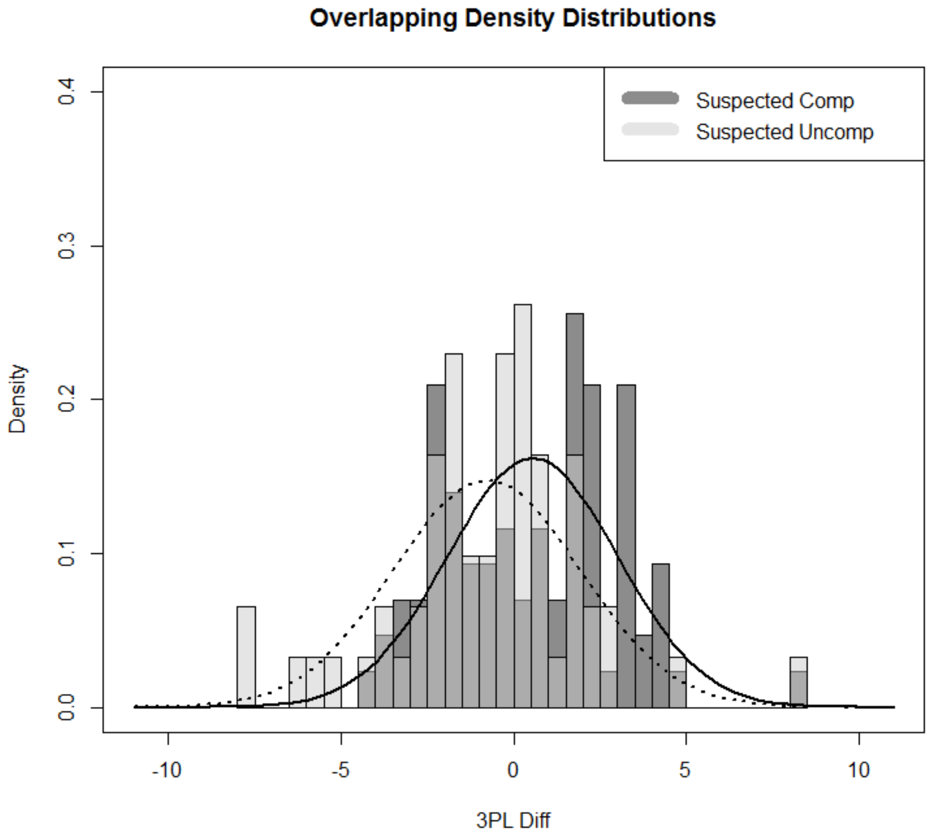

Figure 17. Three-parameter logistic difficulty density distributions. The density distributions for suspected compromised (dark grey) and suspected uncompromised items (light grey). Medium grey hues indicate overlap of the distributions. The lines represent a normal distribution around the suspected compromised items (solid) and the suspected uncompromised (dashed) items.

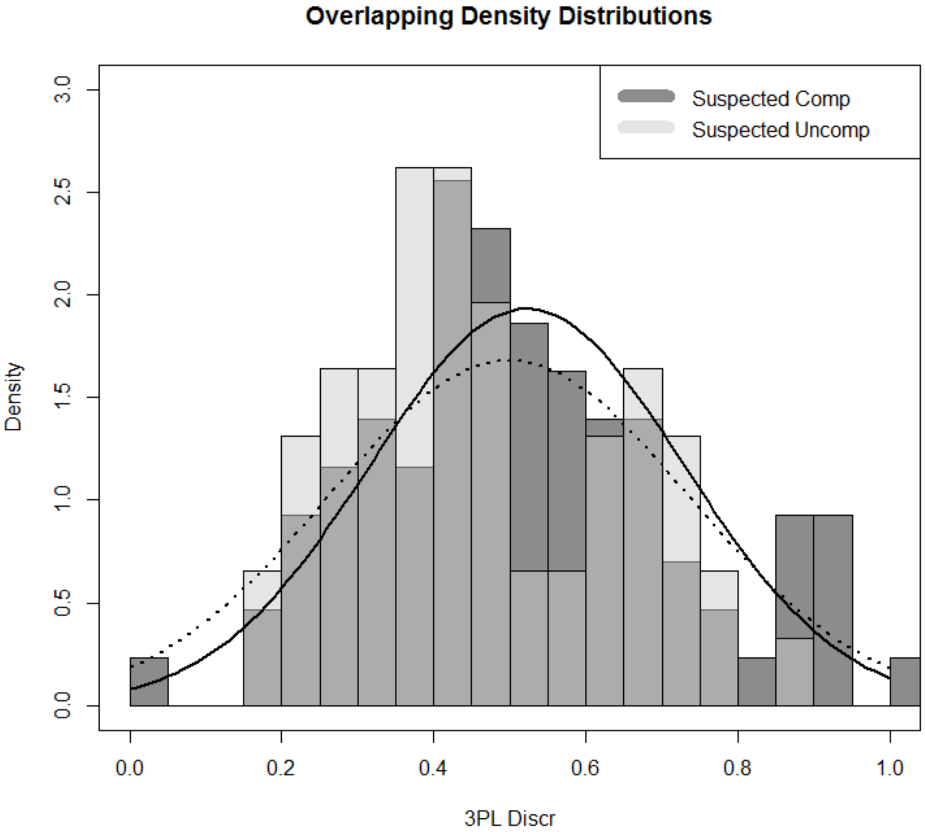

Figure 18. Three-parameter logistic discrimination density distributions. The density distributions for suspected compromised (dark grey) and suspected uncompromised items (light grey). Medium grey hues indicate overlap of the distributions. The lines represent a normal distribution around the suspected compromised items (solid) and the suspected uncompromised (dashed) items.

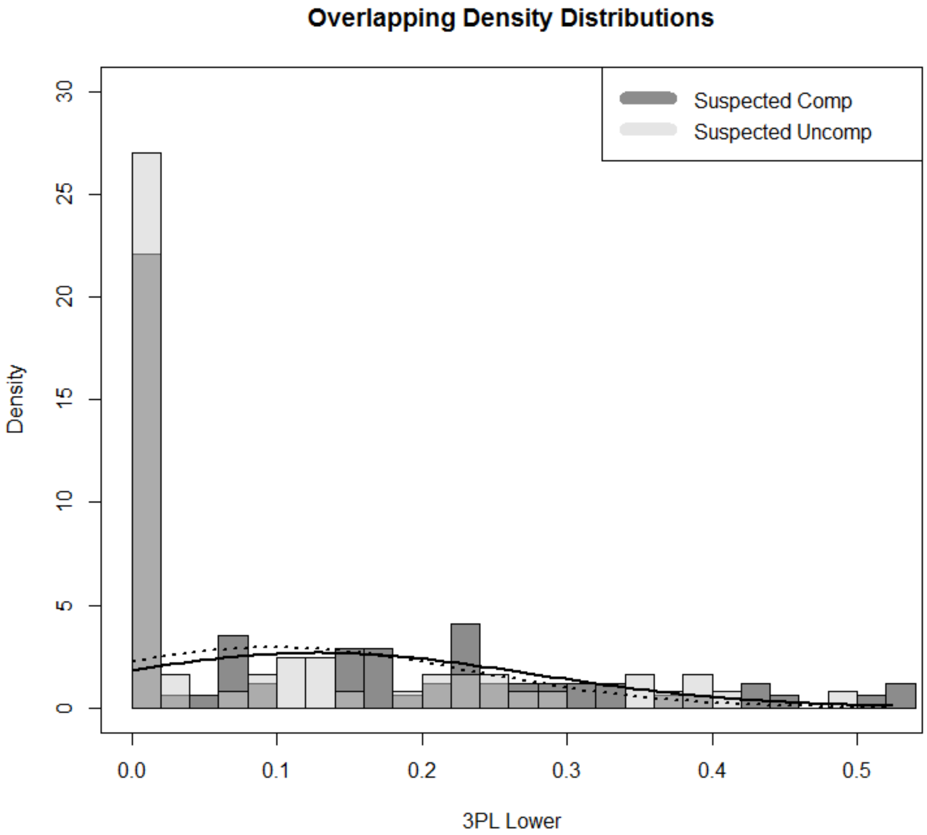

Figure 19. Three-parameter logistic lower limit density distributions. The density distributions for suspected compromised (dark grey) and suspected uncompromised items (light grey). Medium grey hues indicate overlap of the distributions. The lines represent a normal distribution around the suspected compromised items (solid) and the suspected uncompromised (dashed) items.

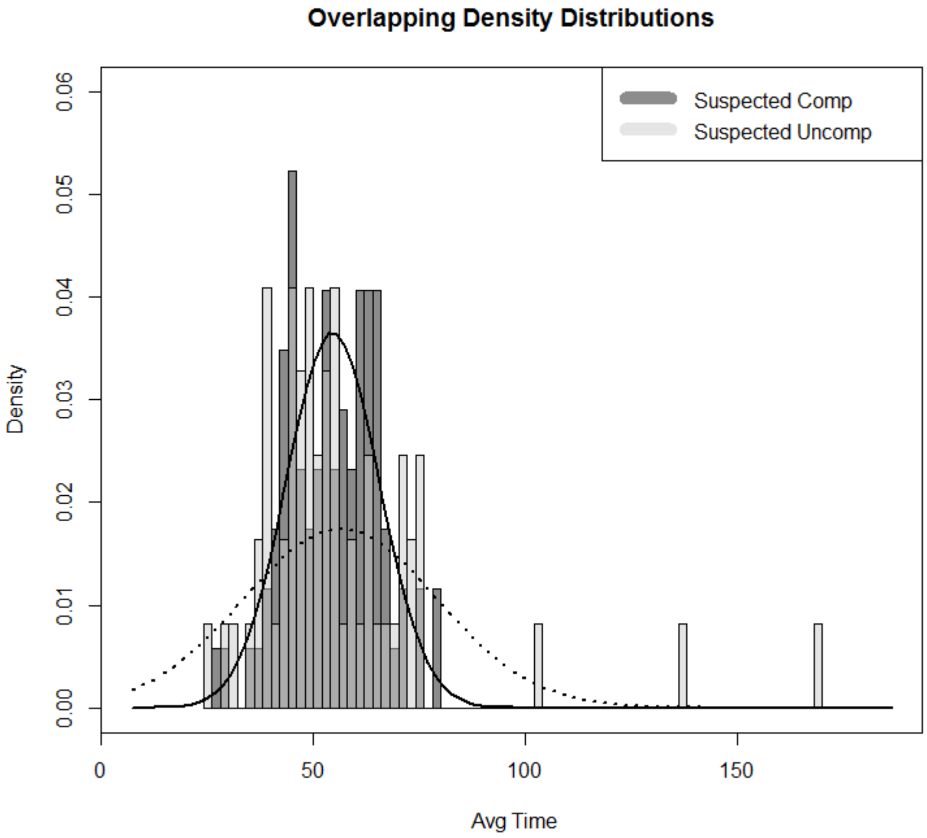

Figure 20. Average response time density distributions. The density distributions for suspected compromised (dark grey) and suspected uncompromised items (light grey). Medium grey hues indicate overlap of the distributions. The lines represent a normal distribution around the suspected compromised items (solid) and the suspected uncompromised (dashed) items.

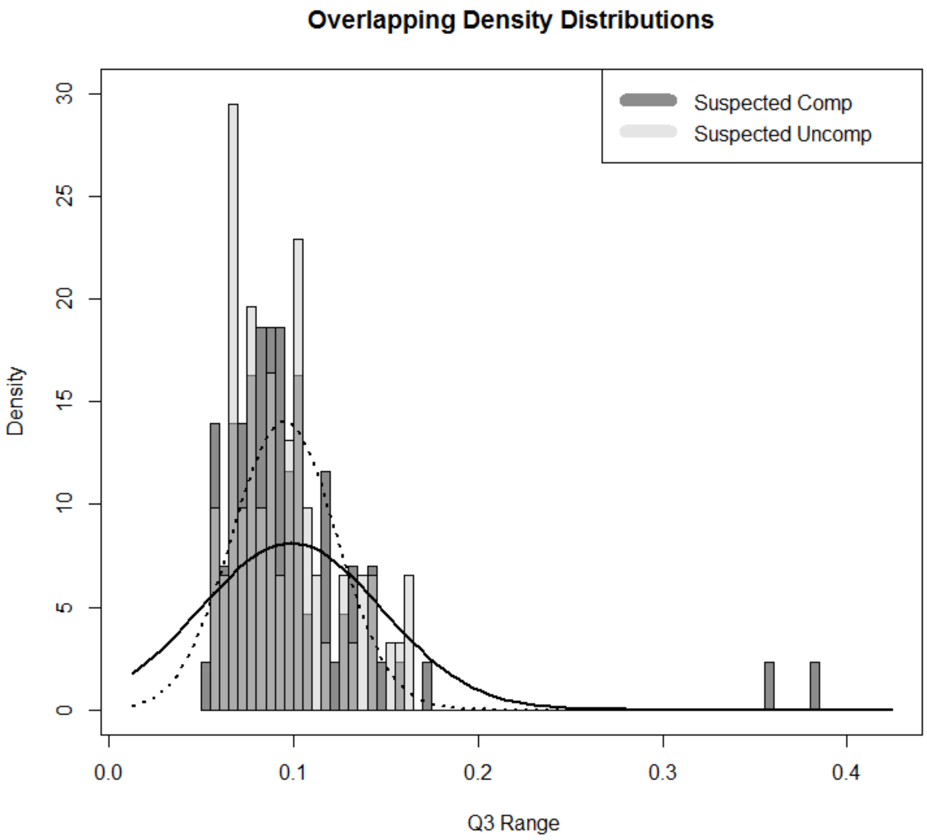

Figure 21. Range of Q_3 estimates density distributions. The density distributions for suspected compromised (dark grey) and suspected uncompromised items (light grey). Medium grey hues indicate overlap of the distributions. The lines represent a normal distribution around the suspected compromised items (solid) and the suspected uncompromised (dashed) items.

SVM Parameters

The gamma, cost, and, for POLY kernels, degree parameters that were selected for each of the thirty models by the multi-step tuning process for the SVM can be viewed in Table 15. Recall that cost is a penalizing parameter that balances complexity of the resulting model with the frequency of error (Cortes & Vapnik, 1995). Cost parameters that are too large can cause overfitting to the training set but cost parameters that are too small can cause underfitting to the training set (Alpaydin, 2004, p. 224). Additionally, the gamma parameter controls the influence of a single data point in the training set (Pedregosa et al., 2011.). Low γ parameters indicate that a data point can have a large range of influence and high γ parameters indicate that a data point can have a small range of influence. The degree parameter in the POLY kernels indicates if the mapping of the data into higher dimensional space is linear (degree = 1), quadratic (degree = 2), cubic (degree = 3), or quartic (degree = 4).

The relationships between parameters were assessed to investigate if the parameter tuning process selected independent or related parameters. The correlation between gamma and cost was not statistically significant for the models collapsed across kernel, *r* (28) = .10, *p* = .599, or within either kernel, RBF *r* (13) = .22, *p* = .439 and POLY *r* (13) = .15, *p* = .588. In the POLY kernel, degree was not statistically significantly related to cost, *r* (13) = .17, *p* = .554, or gamma, *r* (13) = -.05, *p* = .869.

The relationships between overall accuracy and balanced accuracy with the selected parameters were assessed to investigate the effects of the parameters on the accuracy of the models. In the RBF kernel, neither gamma nor cost was a statistically significant predictor of overall accuracy (see Table 16). However, cost was a statistically significant, strong, positive predictor of balanced accuracy in RBF kernel models (see Table 17). In the POLY kernel, gamma was a statistically significant, strong, negative predictor of overall accuracy (see Table 17), but none of the parameters were statistically significant predictors of balanced accuracy (Table 18).

Table 15

Parameters for SVM Models

Kernel	*N*	Split	Gamma	Cost	Degree
RBF	200	1	4	8	NA
RBF	200	2	0.0625	4	NA
RBF	200	3	0.0625	0.0625	NA
RBF	500	1	4	4	NA
RBF	500	2	1	2	NA
RBF	500	3	0.0625	0.0625	NA
RBF	1000	1	0.125	2	NA
RBF	1000	2	0.125	4	NA
RBF	1000	3	0.0625	0.0625	NA
RBF	5000	1	1	2	NA
RBF	5000	2	0.0625	0.0625	NA
RBF	5000	3	0.5	16	NA
RBF	13584	1	0.0625	2	NA
RBF	13584	2	0.125	8	NA
RBF	13584	3	0.25	8	NA
POLY	200	1	0.125	0.5	3
POLY	200	2	4	0.125	3
POLY	200	3	0.25	0.125	1
POLY	500	1	1	0.0625	2
POLY	500	2	0.25	0.25	2
POLY	500	3	0.0625	0.0625	1
POLY	1000	1	2	0.0625	2
POLY	1000	2	4	0.125	2
POLY	1000	3	0.25	0.5	4
POLY	5000	1	0.25	0.0625	3
POLY	5000	2	1	1	4
POLY	5000	3	0.5	0.5	4
POLY	13584	1	1	0.125	2
POLY	13584	2	2	8	3
POLY	13584	3	0.5	0.0625	4

Notes. The parameters for every model variant are given in the table above. NA is listed under RBF Models since degree parameters were not part of the model.

Table 16

Regression of RBF Parameters on Overall Accuracy

	β	*t*	*p*
Gamma	.28	1.11	.289
Cost	.39	1.56	.145

Note. DV is overall accuracy. *** *p* < .001; ** *p* < .01; * *p* < .05

Table 17

Regression of RBF Parameters on Balanced Accuracy

	β	*t*	*p*
Gamma	.22	1.00	.339
Cost	.59	2.67	.020*

Note. DV is balanced accuracy. *** *p* < .001; ** *p* < .01; * *p* < .05

Table 18

Regression of POLY Parameters on Overall Accuracy

	β	*t*	*p*
Gamma	-.59	-2.85	.016*
Cost	-.35	-1.69	.119
Degree	-.05	-.24	.817

Note. DV is overall accuracy. *** *p* < .001; ** *p* < .01; * *p* < .05

Table 19

Regression of POLY Parameters on Balanced Accuracy

	β	*t*	*p*
Gamma	.02	.06	.952
Cost	-.18	-.60	.560
Degree	.12	.41	.689

Note. DV is balanced accuracy. *** *p* < .001; ** *p* < .01; * *p* < .05

Discussion

This study combined Rasch and Item Response Theory estimates, along with other item features such as average response times and local dependence estimates, with Support Vector Machines (SVMs) to classify suspected compromised and suspected uncompromised items. The overall accuracy seemed to show that the models were somewhat accurate at classifying suspected compromised and suspected uncompromised items, but the relative size of the classes seemed to be strongly related to the classifications of the models. Specifically, the results indicated that models with a majority of suspected compromised items (Split 1 and Split 2) showed apparent strong bias towards classifying items as predicted compromised and that models with a majority of suspected uncompromised items (Split 3) showed apparent strong bias towards classifying items as predicted uncompromised. Thus, the imbalance of the classes appeared to be strongly related to the classifications of the models and the accuracy of the SVMs, balanced for class size, was much lower than desired. However, cases that were classified in the opposite direction of the apparent bias were more likely to be correct classifications. For example, items that were predicted compromised items in Split 3 models, which appeared to be biased towards predicting items as predicted uncompromised, were more likely to represent suspected compromised items. This result may indicate that when models are biased in a particular direction, only the items that are the most different from the predominant class are classified into the other predicted category.

The SVM model variants with radial basis function (RBF) kernels slightly outperformed the models with polynomial (POLY) kernels on average and both kernels exhibited similar variability in the results, with the exception of POLY kernels exhibiting less variability in overall accuracy and balanced accuracy. There did not appear to be consistent differences in the accuracy outcomes between models with differing sample sizes. This may be because the results of the SVMs seemed to be mostly driven by the class sizes, rather than separations in the item features. There were consistent differences in the outcomes of the SVMs based on the item split, representing the previously discussed opposite directions of apparent bias for models with a majority of suspected compromised items (Split 1 and Split 2) and models with a majority of suspected uncompromised items (Split 3).

The relationships between the SVM parameters (cost, gamma, and degree for POLY kernels) were assessed to gain insight into the parameter selections of the tuning process. The results showed that the parameters were unrelated to one another for both the RBF and POLY kernel models. In the RBF kernel models, cost (a penalizing parameter that balances complexity of the resulting model with the frequency of error) was positively related to balanced accuracy, but unrelated to overall accuracy. Thus, models with higher cost parameters generally had higher balanced accuracy. However, large cost parameters can cause overfitting to the training set (Alpaydin, 2004, p. 224), so it is unclear if the models with high cost parameters in the current study would exhibit more accurate results in a separation validation set than models with low cost parameters or if the link between cost parameters and balanced accuracy is caused by overfitting in the current data. In the POLY kernel models, gamma (a parameter that controls the influence of a single data point) was negatively related to overall accuracy, but unrelated to balanced accuracy. Thus, models with lower gamma parameters, allowing a single data point to have more influence, generally had higher overall accuracy. This finding may also be the result of overfitting to the current data, and would be best addressed by testing the models in a separate validation set.

We hypothesized that the item features with the highest weights would be Rasch item infit and outfit, but the item feature results showed that the two most important features for distinguishing between suspected compromised and suspected uncompromised items were the Rasch model standard error and the Rasch item difficulty. The three least important features in the SVMs were the Rasch outfit, Q_3 range, and average response times. The relatively high importance of Rasch difficulty estimates in predicting compromise is consistent with the findings of Thomas et al. (2016), who found that item difficulty was the most important Rasch model predictor of item compromise. We also hypothesized that Rasch estimates would outperform 3PL estimates, but the evidence for this hypothesis was mixed. Consistent with this hypothesis, the results of the SVM-RFE feature selection algorithm showed that Rasch estimates had lower rank than the 3PL estimates. However, analyses of average absolute feature weights and rank sums of features showed that Rasch estimates of item difficulty and standard error had higher feature weights than the 3PL estimates, but Rasch estimates of infit and outfit had lower feature weights than the 3PL estimates, with the exception of the average absolute feature weights results for the POLY kernel models. Thus, there was not enough evidence to conclude that the Rasch estimates outperformed the 3PL estimates. Finally, we hypothesized that 3PL discrimination would outperform the 3PL lower limit. The results of a feature selection algorithm and analyses of rank sums for the features showed that discrimination did have a higher feature weight than the lower limit, but the average absolute feature weights showed that discrimination had a lower feature weight than the lower limit. Further investigation showed that in 19 of the 30 models, the discrimination had a higher feature weight and was ranked lower than the lower limit, indicating that discrimination was a more important feature than the lower limit in those models. Thus, the majority of the models indicated that discrimination outperformed the lower limit, but not all. The results of the current study appeared to replicate the finding of Thomas et al. (2016) that discrimination is more accurate at distinguishing suspected compromised items from suspected uncompromised items than the lower limit. However, the results of the feature weights in the current study should be interpreted cautiously for several reasons. The assessment of feature weights provides information on the importance of features in the data for deriving the classifications made by the models. In the current study, the classifications made by the models showed apparent bias depending on the relative class sizes and were often inaccurate. Thus, the feature weights may contribute very little to the understanding of what item properties actually distinguish between suspected uncompromised and suspected compromised items.

The current study successfully combined Rasch and Item Response Theory model estimates with Support Vector Machines to address a classification problem. This method marries measurement models from the field of psychometrics with unsupervised techniques from the field of machine learning. Additionally, this method could be used for any situation in which the researcher believes that Rasch or Item Response Theory model estimates may be important in the classifications of units (e.g. items or persons). However, there were limitations in the application of this method in the current study, which are discussed in more detail below.

Limitations of the Current Study

There are many possible reasons that the SVMs in the current study exhibited lower than desired accuracy and apparent bias in classifications, but in the paragraphs below, limitations in the following areas will be addressed: the parameter tuning process, the quality of the suspected item categories, and how separable the suspected item categories were based on the item features.

Firstly, the parameter tuning process of the current study had limitations in the number of tunings, the assessment of performance, and the selection of best parameters. Each SVM model variant in the study was analyzed once, rather than a large number of times. This means that only one multi-step parameter tuning process took place per model variant. Future research could address this limitation by running each model variant multiple times. Alternatively, simulations could be used to investigate the best ranges of parameters to tune over as well as what are reasonable and unreasonable parameters for various kernels, given the characteristics of a particular data set. Such research would contribute greatly to the literature on the usage of SVMs. Additionally, the parameter tuning function used the overall accuracy to calculate the performance of the parameters. Thus the best parameters may represent those that were best in terms of overall accuracy, but better parameters may have been obtained by using the balanced accuracy, or other accuracy measures, to assess performance in the tuning process. Finally, the constraints that were utilized to make the parameter tuning process more computationally and temporally manageable, such as selecting only the 30 best results for each parameter to tune over in the next step, may have decreased the quality of the resulting best parameters. Therefore, the parameters selected for the SVMs by the parameter tuning process may not have been the best possible parameters. Other parameter tunings may have led to different parameters being selected for use in the SVMs, which may have improved the accuracy of the models.

Secondly, the quality of the suspected item categories from the testing company cannot be verified by the researchers. It is possible that the suspected item categories do not reflect actual compromised and uncompromised items. For example, suppose that screenshots of an item were found but the item was never shared or posted online. Or the items could have been posted online but very viewed by few examinees. The unknown extent of the item compromise is a confounding factor in the results of this study. Thus, it is possible that this combination of methods is performing as best as it can given the weaknesses of the suspected item categories. This may explain why the analysis of item features did not show the expected results, that Rasch item infit and Rasch item outfit would be the most important features. If the suspected item categories do not represent actual differences in the level of item compromise, the feature weights would not be valid estimates of the importance of the features in predicting compromise. They would instead represent the importance of the item features in distinguishing between items that the testing company deemed suspected compromised and suspected uncompromised, which is not particularly useful in detecting item compromise. Several methods for overcoming this limitation are discussed in the Directions for Future Research section below.

Thirdly, another reason for the low performance of the SVMs could be that the items are not separable in the feature space, either because of the relatively small number of items (N = 147) or because these nine item features are simply not useful in distinguishing suspected compromised from suspected uncompromised items. Future research should attempt classification on a larger pool of items in data that includes suspected categories in addition to addressing the previously discussed limitations of the models. Once these limitations are addressed, the question of whether or not these item features are useful in distinguishing suspected compromised from suspected uncompromised items can and should be reexamined.

Directions for Future Research

The reason that SVMs were able to be used in the current study was because suspected item categories were obtained from the test publisher. However, the quality of the suspected item categories cannot be verified by the researchers. Thus, it is possible that this combination of methods was performing well, given the poor quality of the suspected item categories. One way to address this limitation in future research would be to mimic item compromise in a laboratory setting. All participants would take a computerized test, but participants in an experimental condition would be allowed to study half of the test items, with answers, for some amount of time before beginning the test. The control condition participants would not receive any test items or answers. The participants would then complete the test and a brief survey about psychological and educational factors that may impact their performance on the exam. Not only could the statistically identified compromised items be compared to the actual compromised items at the conclusion of the study, but the statistically identified examinees with pre-knowledge could be compared to those who did indeed have access to test answers. If this experiment were successful, subsequent experiments could use the same paradigm to investigate other aspects of cheating by manipulating the accuracy of the answer key, the amount of time participants had to study the answer key, or the presence of distractor tasks between studying the answer key and taking the test. The greatest benefit of such an experiment would be that the correct status of examinees would be known, unlike in most cases of test fraud, allowing a direct evaluation of the accuracy of various statistical methods for identifying examinees with pre-knowledge to be assessed. Such analyses would greatly contribute to the literature on test fraud.

It is important to the science of the detection of test fraud that future researchers directly compare the performance of a wider variety of item properties for detecting item compromise. This would allow the most important item properties for detecting item compromise in various circumstances to be identified. These item properties could then be used to predict the status of their items even in the absence of suspected item categories. This would be more efficient than continual internet monitoring to detect proprietary test content online, which is currently a common method used to detect item compromise (Fremer & Addicott, 2011). Not only does internet monitoring represent a significant cost in terms of time and money, it is also an inefficient method for detecting compromised items as the amount of information that must be searched is potentially infinite. Moreover, proprietary content may be hidden behind paywalls, encrypted, or hidden from search functions. However, if compromised items could be detected using a predictive method, like the one proposed in this paper, then this method could be periodically applied to sets of testing data for detection of compromised items.

Conclusions

In summary, this study used a combination of Rasch and Item Response Theory estimates, and other item properties such as average response times and local dependence estimates, with Support Vector Machines. The results showed that this method appeared to be somewhat accurate at classifying suspected compromised and suspected uncompromised items, but that the main factor driving these results was the relative size of the classes. The SVMs showed apparent bias towards predicting the items into the category of whichever item class was larger. Thus, the accuracy of the SVMs, balanced for class size, was much lower than desired. However, cases that were classified in the opposite direction of the apparent bias were more likely to be correct classifications.

The item feature results showed that the two most important features were the Rasch model standard error and the Rasch item difficulty. However, the results of the feature weights in the current study should be interpreted cautiously for several reasons. The assessment of feature weights provides information on the importance of features in the data for deriving the classifications made by the models. In the current study, the classifications made by the models appeared to be biased by class size and often inaccurate. Thus, the feature weights obtained in this study may contribute very little to the understanding of what item properties actually distinguish between suspected uncompromised and suspected compromised items.

The current study pioneered a flexible, hybrid method that combines the disparate fields of psychometrics and machine learning, and can be adapted for application in a variety of contexts. This method has a promising future as the age of big data continues and the need for classifications persists.

References

Alpaydin, E. (2004). *Introduction to Machine Learning*. Cambridge, MA: MIT Press.

Altman, D.G. & Bland, J.M. (1994a). Statistics notes: Diagnostic tests 1: sensitivity and specificity. *BMJ*, *308*, 1552.

Altman, D.G. & Bland, J.M. (1994b). Statistics notes: Diagnostic tests 2: predictive values. *BMJ*, *309*, 102.

An, X. and Yung, Y. (2014, March). Item response theory: What it is and how you can use the IRT procedure to apply it. Presented at the annual meeting of the SAS® Global Forum, Washington, D.C.

Birnbaum A. (1968). Some latent trait models and their uses in inferring an examinee's ability. In F.M. Lord & M.R. Novick, *Statistical theories of mental test scores* (pp. 395-479). Reading, MA: Addison-Wesley.

Bishop, C. M. 2006. *Pattern Recognition and Machine Learning*. New York, NY: Springer.

Bliss, T. J. (2012). *Statistical methods to detect cheating on tests: A review of the literature* (Unpublished doctoral dissertation). Brigham Young University, Provo, Utah.

Bohlig, M., Fisher, W.P. Jr., Masters, G.N., & Bond, T. (1998). Content validity and misfitting items. *Rasch Measurement Transactions*, *12*, 607.

Borsboom, D., & Mellenbergh, G. J. (2004). Why psychometrics is not pathological: A comment on Michell. *Theory & Psychology*, *14*, 105–120.

Brodersen, K.H., Ong, C.S., Stephan, K.E., & Buhmann, J.M. (2010). The balanced accuracy and its posterior distribution. Proceedings of the *20th International Conference on Pattern Recognition*.

Burges, C.J.C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery, 2*, 121-167.

Caveon (2008, March 2). Trojan items and answer-key arbitrage. *Caveon Security Insights*. Retrieved from http://caveon.com/df_blog/trojan-items-and-answer-key-arbitrage

Chen, W.H. & Thissen, D. (1997). Local dependence indexes for item pairs using Item Response Theory. *Journal of Educational and Behavioral Statistics, 22*, 265-289.

Cherkassy, V., & Ma, Y. (2004). Practical selection of SVM parameters and noise estimation for SVM regression. *Neural Networks*, *17*, 113-126.

Cristianini, N. and Shawe-Taylor, J. (2000) An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods. Cambridge University Press, Cambridge, MA.

Cizek, G. J. (1999). *Cheating on tests: How to do it, detect it, and prevent it*. Mahwah, NJ: Erlbaum.

Cizek, G. J. (2003). *Detecting and preventing classroom cheating: Promoting integrity in schools*. Thousand Oaks, CA: Corwin Press.

Clark, M., Skorupski, W., Jirka, S., McBride, M., Wang, C., & Murphy, S. (2014, February). *An investigation into statistical methods to identify aberrant response patterns*. Retrieved from http://researchnetwork.pearson.com/wp-content/uploads/Data-Forensics-RD-Research-Report-Person-Fit.pdf

Cortes, C. & Vapnik, V (1995). Support vector networks. *Machine learning*, *20*, 273-297.

De Ayala, R.J. (2008). *The theory and practice of item response theory*. New York, NY: Guilford Press.

Dickison, P., & Maynes, D. (2014, November 19). *Unlocking the mystery of the validity triangle: Is test security required for test validity?* Retrieved March 25, 2015 from http://www.slideshare.net/caveonweb/caveon-webinar-series-unlocking-the-mystery-of-the-validity-triangle-112014

Dundar, M., Krishnapuram, B., Bi, J. & Rao, R. (2007, January). Learning classifiers when the training data is not IID. In *IJCAI* (pp. 756-761). Retrieved October 22, 2015 from http://people.ee.duke.edu/~lcarin/IJCAI07-121.pdf

Embretson, S. E., & Hershberger, S. L. (1999). *The new rules of measurement: What every psychologist and educator should know*. Mahwah, NJ: Lawrence Erlbaum Associates.

Embretson, S. E. and Reise, S. (2000). *Item response theory for psychologists*. Mahwah, NJ: Lawrence Erlbaum.

FairTest. (2007, July). *Cheating cases continue to proliferate.* Retrieved June 1, 2015 from http://www.fairtest.org/cheating-cases-continue-proliferate

Ferrando, P. J. (2007). Factor-analytic procedures for assessing response pattern scalability. *Multivariate Behavioral Research*, *42*, 481-507.

Fischer, G. H. (1973). The linear logistic test model as an instrument in educational research. *Acta Psychologica*, *37*, 359-374.

Fremer, J., & Addicott, S. (2011, February 16). Patrolling the web: Who is selling your exams on the internet? Retrieved from http://www.caveon.com/downloads/Patrolling_the_Web.pdf

Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine Learning, 46*, 389-422.

Haney, W. M., & Clarke, M. J. (2007). Cheating on tests: Prevalence, detection, and implications for on-line testing. In E. M. Anderman & T. B. Murdock (Eds.), *Psychology of academic cheating* (pp. 255-287). San Diego, CA: Elsevier Academic Press.

Hofmann, T. Schölkopf, B. & Smola, A. (2008). Kernel methods in machine learning. *The Annals of Statistics, 36*, 1171–1220

Hsu, C., Chang, C., & Lin, C. (2003). A practical guide to support vector classification. Retrieved from http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf

Hui, H. F. (2010). *Stability and sensitivity of a model-based person-fit index in detecting item pre-knowledge in computerized adaptive test*. (Unpublished doctoral dissertation). University of Hong Kong, Pokfulam, Hong Kong.

Karabatsos, G. (2003). Comparing the aberrant response detection performance of thirty-six person-fit statistics. *Applied Measurement in Education*, *16*, 277-298.

Keerthi, S.S. Lin, & C.J. (2003). Asymptotic behaviors of support vector machines with Gaussian kernel. *Neural Computation*, *15*:1667–1689.

Kingston, N. M., & Clark, A. K. (2014). *Test Fraud: Statistical Detection and Methodology*. New York, NY: Routledge.

Linacre, J. M. (2016). Winsteps® Rasch measurement computer program. Beaverton, Oregon: Winsteps.com

Linacre J. M. (1996). The Rasch model cannot be "disproved”! *Rasch Measurement Transactions*, *10,* 512-514.

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

Mair, P., Hatzinger, R., & Maier M. J. (2015). eRm: Extended Rasch Modeling. 0.15-6. http://erm.r-forge.r-project.org/

Marianti, S., Fox, J., Avetisyan, M., Veldkamp, B.P., & Tijmstra, J. (2014). Testing for aberrant behavior in response time modeling. *Journal of Educational and Behavioral Statistics*, *39,* 426-451.

Maynes, D. (2008). Caveon speaks out on IT exam security: The last five years. Retrieved September 10, 2015 from http://www.caveon.com/articles/it_exam_security.htm

Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., & Leisch, F. (2015). e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien. R package version 1.6-7. http://CRAN.R-project.org/package=e1071

Michell, J. (2008). Is psychometrics pathological science? *Measurement: Interdisciplinary Research and Perspectives*, *6,* 7-24.

Mosteller F. and Tukey J.W. *Data analysis, including statistics.* In Handbook of Social Psychology. Addison-Wesley, Reading, MA, 1968.

Müller, K.R. Mika, S., Rätsch, G., Tsuda, K., & Schölkopf, B. (2001). An introduction to kernel-based learning algorithms. *IEEE transactions on neural networks*, *12*, 181-201.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python, *Journal of Machine Learning Research*, *12*, 2825-2830.

R Core Team (2015). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Rasch, G. (1960). *Probabilistic models for some intelligence and attainment tests*. Copenhagen, Denmark: Danish Institute for Educational Research. (Expanded ed., 1980 Chicago: University of Chicago Press.)

Robitzsch, A. (2015). sirt: Supplementary Item Response Theory Models. R package version 1.9-0. http://CRAN.R-project.org/package=sirt

Schmidt, K. M., & Embretson, S. E. (2012). Item response theory and measuring abilities. In J.A. Schinka and W.F. Velicer (Eds.), *Research Methods in Psychology* (2nd ed.). Volume 2 of Handbook of Psychology (I.B. Weinger, Editor-in-Chief). NY: John Wiley & Sons, Inc.

Shawe-Taylor, J. & Cristianini, N. (2004). Kernel methods for pattern analysis. New York, NY: Cambridge university press.

Shu, Z. (2011). *Detecting test cheating using a deterministic, gated item response theory model* (Unpublished doctoral dissertation). University of North Carolina at Greensboro, Greensboro, NC.

Sijtsma, K. (1986). A coefficient of deviant response patterns. *Kwantitative Methoden, 7,* 131–145.

Sijtsma, K., & Meijer, R. R. (1992). A method for investigating the intersection of item response functions in Mokken's non-parametric IRT model. *Applied Psychological Measurement, 16*, 149–157.

Simha, A., Armstrong, J. P., & Albert, J. F. (2012). Who leads and who lags? A comparison of cheating attitudes and behaviors among leadership and business students. *Journal of Education for Business, 87,* 316–324.

Sinharay, S. (2015). Assessment of person fit for mixed-format tests. *Journal of Educational and Behavioral Statistics, 40,* 343-365.

Smith, R. W. (2004, April 2). *The impact of braindump sites on item exposure and item parameter drift*. Paper presented at the American Education Research Association, San Diego, California.

Sorenson, D. (2010, Dec 15). 50 ways students cheat on tests. Retrieved from http://www.caveon.com/downloads/50_Ways_12-15-10.pdf

Stone, M. Cross-validatory choice and assessment of statistical predictions. (1974). *Journal of the Royal Statistical Society, 36*, 111–147.

StatSoft, Inc. (2013). Electronic Statistics Textbook. Tulsa, OK: StatSoft. WEB: http://www.statsoft.com/textbook/.

Thomas, S. L., Schmidt, K. M., Brunnert, K.A., Bontempo, B., Wilson, D.J., & Maynes, D. (2016). *Do cheaters never prosper? Using psychometric analyses to detect test fraud*. Manuscript submitted for publication.

United States Medical Licensing Examination Performance Data. (2014). *2013 Step 1*. Retrieved March 25, 2015, from http://www.usmle.org/performance-data

Whitley, B. E. (1998). Factors associated with cheating among college students. *Research in Higher Education*, *39*, 235-274

Wright, B. D. (1980). [Afterword]. In G. Rasch (reprint), Probabilistic models for some intelligence and attainment tests. Chicago, IL: University of Chicago Press.

Wright, B. D. & Linacre, J. M. (1989). Observations are always ordinal; measurements, however, must be interval. *Archives of Physical Medicine and Rehabilitation, 70*, 857-860.

Wright B. D. & Masters G. N. (1982). *Rating Scale Analysis*, Chicago: MESA Press.

Yen, W.M. (1984). Effects of local item dependence on the fit and equating performance of the three-parameter logistic model. *Applied Psychological Measurement*, *8*, 125-145.

Yen, W.M. (1993). Scaling performance assessments: Strategies for managing local item dependence. *Journal of Educational Measurement*, *30*, 187-213.

Appendix A

Item	R Diff	R SE	R Infit	R Outfit	3PL Diff	3PL Discr	3PL Lower	Q_3 Range	Avg Time	Suspected Item Category (1 = Comp)
1	0.80	0.02	0.48	0.92	1.96	0.45	0.14	0.09	50.00	1
2	0.81	0.02	-3.45	-3.53	1.25	0.47	0.00	0.06	62.74	1
3	0.45	0.02	6.42	6.08	3.38	0.55	0.46	0.10	50.99	1
4	0.51	0.02	4.49	4.53	2.20	0.40	0.28	0.06	54.56	1
5	0.62	0.02	3.87	3.32	3.03	0.87	0.43	0.08	59.65	1
6	-0.23	0.02	2.01	3.24	-2.64	0.27	0.00	0.09	62.98	1
7	-0.22	0.02	0.22	0.09	-0.99	0.41	0.08	0.08	60.02	1
8	1.20	0.02	9.90	9.90	8.00	0.05	0.00	0.10	36.95	1
9	0.17	0.02	5.09	5.32	3.27	0.50	0.52	0.09	52.47	1
10	0.43	0.02	4.16	4.26	-0.23	0.27	0.00	0.15	68.39	1
11	-1.12	0.03	-0.70	-1.28	-2.61	0.51	0.00	0.06	45.76	1
12	0.87	0.02	-1.96	-2.15	2.23	0.54	0.18	0.13	67.78	1
13	-0.58	0.02	0.68	1.92	-3.22	0.31	0.00	0.09	60.97	1
14	-1.01	0.03	-0.83	-1.64	-2.24	0.53	0.00	0.09	56.17	1
15	0.09	0.02	-4.81	-5.64	0.88	0.78	0.22	0.08	50.19	1
16	-0.04	0.02	4.53	5.59	-4.22	0.16	0.00	0.09	52.85	1
17	0.68	0.02	0.61	0.82	1.77	0.47	0.17	0.10	40.59	1
18	0.68	0.02	6.75	6.79	3.11	0.43	0.33	0.07	62.43	1
19	0.20	0.02	-1.73	-2.47	1.51	0.68	0.34	0.09	67.34	1
20	-1.14	0.03	-0.98	-2.05	-2.11	0.59	0.00	0.07	45.05	1
21	0.10	0.02	0.63	0.68	-0.63	0.37	0.01	0.06	43.73	1
22	-0.03	0.02	0.01	-0.48	0.06	0.45	0.18	0.09	78.80	1
23	0.84	0.02	9.90	9.90	4.67	0.42	0.38	0.08	39.04	1
24	0.92	0.02	6.06	5.88	1.56	0.26	0.00	0.08	75.18	1
25	-0.32	0.02	0.55	0.84	-1.94	0.35	0.00	0.08	79.82	1
26	-0.02	0.02	-5.17	-6.63	0.99	0.93	0.29	0.17	35.09	1
27	-0.31	0.02	-1.08	-1.91	-0.44	0.53	0.15	0.08	47.81	1
28	-0.08	0.02	1.25	2.32	-1.60	0.31	0.00	0.08	45.63	1
29	0.64	0.02	-4.94	-5.25	1.84	0.72	0.22	0.09	64.94	1
30	0.43	0.02	3.31	3.15	2.10	0.43	0.31	0.13	67.40	1
31	0.92	0.02	-9.90	-9.90	1.94	0.92	0.16	0.11	64.85	1
32	1.19	0.02	2.24	3.24	3.26	0.59	0.22	0.07	63.01	1
33	1.32	0.02	1.37	2.45	3.43	0.61	0.21	0.10	43.31	1

34	-1.00	0.03	0.10	0.37	-3.74	0.36	0.00	0.07	53.72	1
35	-1.32	0.03	-0.87	-3.04	-0.49	0.75	0.51	0.07	56.78	1
36	-1.83	0.03	-0.55	-1.67	-3.25	0.60	0.00	0.06	49.98	1
37	1.32	0.02	4.21	5.23	4.03	0.64	0.27	0.09	64.73	1
38	0.75	0.02	-1.62	-1.02	2.19	0.63	0.23	0.07	42.09	1
39	1.52	0.02	1.09	1.46	3.61	0.39	0.07	0.07	61.63	1
40	0.95	0.02	-3.88	-3.39	2.41	0.75	0.22	0.08	53.82	1
41	-1.38	0.03	-1.33	-3.34	-2.02	0.69	0.00	0.08	41.17	1
42	0.59	0.02	2.36	2.24	0.56	0.34	0.00	0.09	43.98	1
43	-0.16	0.02	-7.80	-9.90	0.38	1.05	0.08	0.16	41.64	1
44	0.16	0.02	-0.55	-1.32	1.93	0.81	0.44	0.10	44.88	1
45	-1.42	0.03	-1.89	-5.66	-1.12	0.94	0.14	0.10	28.07	1
46	0.84	0.02	6.16	5.99	3.35	0.50	0.31	0.08	60.73	1
47	1.07	0.02	-2.07	-1.67	2.03	0.45	0.04	0.07	38.30	1
48	-0.81	0.02	-1.44	-2.61	-1.68	0.56	0.00	0.09	57.26	1
49	-1.08	0.03	-0.82	-1.15	-2.61	0.49	0.00	0.07	59.06	1
50	0.19	0.02	-4.45	-4.82	0.16	0.57	0.00	0.07	61.32	1
51	-0.69	0.02	-0.23	-0.55	-2.36	0.41	0.00	0.06	52.96	1
52	1.03	0.02	-2.31	-2.57	1.75	0.43	0.00	0.10	63.45	1
53	0.23	0.02	-0.77	-1.06	0.80	0.51	0.19	0.08	42.02	1
54	-1.07	0.03	-1.53	-3.61	-1.38	0.69	0.09	0.09	46.42	1
55	1.47	0.02	3.01	3.84	4.22	0.41	0.15	0.38	52.31	1
56	-0.67	0.02	-0.35	-0.72	-2.20	0.43	0.00	0.07	47.98	1
57	-1.17	0.03	-0.78	-2.11	-2.46	0.54	0.00	0.08	45.45	1
58	-0.74	0.02	-2.84	-5.64	-0.19	0.88	0.24	0.13	26.68	1
59	0.98	0.02	8.27	7.85	1.83	0.23	0.00	0.10	54.05	1
60	0.31	0.02	-6.43	-6.90	0.68	0.68	0.06	0.14	74.23	1
61	-1.46	0.03	-0.92	-3.09	-2.35	0.65	0.00	0.06	45.37	1
62	0.54	0.02	-8.50	-8.63	1.55	0.87	0.21	0.12	57.19	1
63	-0.81	0.02	-0.59	-0.51	-2.34	0.45	0.00	0.09	64.12	1
64	-0.37	0.02	-0.61	-1.22	-0.61	0.49	0.17	0.08	46.33	1
65	0.21	0.02	3.31	3.89	-1.03	0.27	0.00	0.09	55.17	1
66	1.19	0.02	1.98	2.19	2.74	0.39	0.08	0.10	65.50	1
67	-0.29	0.02	2.15	3.33	-3.56	0.23	0.00	0.12	62.71	1
68	0.83	0.02	2.76	3.08	1.25	0.34	0.00	0.05	63.04	1
69	0.35	0.02	-2.53	-3.30	2.01	0.89	0.39	0.10	71.18	1
70	-0.27	0.02	0.67	0.87	-2.03	0.33	0.00	0.11	43.61	1
71	-0.41	0.02	0.60	0.63	-1.97	0.35	0.08	0.12	45.02	1

72	0.73	0.02	-1.38	-1.91	2.31	0.63	0.26	0.10	44.22	1
73	1.03	0.02	-0.57	-0.37	2.25	0.48	0.10	0.06	60.15	1
74	0.51	0.02	6.42	6.48	-0.14	0.23	0.00	0.08	65.12	1
75	0.84	0.02	7.46	7.29	1.23	0.24	0.00	0.12	59.24	1
76	0.98	0.02	4.83	5.13	3.25	0.44	0.24	0.36	48.71	1
77	-0.60	0.02	-1.44	-2.45	-1.41	0.54	0.00	0.12	44.63	1
78	0.60	0.02	-8.70	-8.91	1.80	0.93	0.25	0.09	65.21	1
79	-1.17	0.03	-1.60	-3.81	-1.64	0.71	0.00	0.14	61.75	1
80	-0.40	0.02	-2.59	-3.77	-0.81	0.60	0.00	0.14	58.98	1
81	-1.32	0.03	-1.50	-3.27	-1.92	0.70	0.00	0.13	49.15	1
82	-0.47	0.02	1.27	1.96	-3.11	0.29	0.00	0.07	53.46	1
83	-0.28	0.02	2.36	3.86	-3.69	0.22	0.00	0.11	54.26	0
84	0.45	0.02	-7.41	-7.65	1.01	0.70	0.09	0.12	46.00	0
85	0.34	0.02	5.18	5.65	-0.81	0.24	0.00	0.06	65.06	0
86	-2.45	0.05	-0.74	-5.28	-1.77	1.12	0.21	0.14	31.22	0
87	0.47	0.02	3.03	3.39	0.86	0.34	0.11	0.06	54.54	0
88	-0.07	0.02	-4.56	-5.64	0.23	0.71	0.10	0.11	58.94	0
89	-0.15	0.02	-3.54	-4.97	0.55	0.76	0.24	0.14	37.69	0
90	-0.44	0.02	1.38	2.06	-3.11	0.29	0.00	0.10	42.13	0
91	0.12	0.02	1.24	0.84	2.25	0.71	0.48	0.08	44.01	0
92	-0.58	0.02	0.25	0.77	-2.58	0.36	0.00	0.07	54.38	0
93	-0.01	0.02	1.09	1.00	-1.31	0.32	0.00	0.13	70.28	0
94	1.14	0.02	8.81	8.76	2.78	0.18	0.00	0.11	50.32	0
95	0.04	0.02	1.13	1.45	-1.05	0.34	0.00	0.15	55.62	0
96	0.21	0.02	-4.51	-5.15	0.64	0.65	0.11	0.10	46.14	0
97	0.76	0.02	3.94	3.94	2.29	0.40	0.19	0.10	39.53	0
98	0.52	0.02	-0.48	-0.66	1.88	0.61	0.27	0.13	72.53	0
99	-0.41	0.02	-2.14	-3.50	-0.13	0.66	0.21	0.11	47.67	0
100	-1.29	0.03	-1.42	-3.66	-1.83	0.71	0.00	0.10	39.96	0
101	0.04	0.02	2.85	3.26	-1.84	0.25	0.00	0.09	48.02	0
102	0.87	0.02	-0.70	0.09	2.51	0.68	0.25	0.16	75.59	0
103	-0.41	0.02	-0.70	-1.80	0.56	0.65	0.40	0.07	41.36	0
104	0.22	0.02	0.74	0.84	0.36	0.42	0.12	0.14	39.29	0
105	0.41	0.02	0.53	1.14	0.13	0.37	0.00	0.16	75.67	0
106	0.22	0.02	-0.12	-0.09	0.55	0.46	0.15	0.08	52.73	0
107	-0.40	0.02	-3.29	-4.64	-0.47	0.70	0.04	0.10	44.38	0
108	0.17	0.02	-0.76	-1.12	0.21	0.46	0.09	0.09	53.98	0
109	-2.41	0.04	-0.62	-3.60	-2.62	0.89	0.00	0.09	29.68	0

110	-0.60	0.02	-0.47	-0.62	-2.02	0.43	0.00	0.09	52.87	0
111	-0.17	0.02	-3.67	-4.27	-0.31	0.64	0.00	0.13	55.26	0
112	0.33	0.02	2.49	2.67	-0.34	0.31	0.00	0.07	51.45	0
113	0.02	0.02	0.57	0.81	-0.85	0.38	0.00	0.07	63.57	0
114	0.33	0.02	-3.25	-3.44	0.30	0.50	0.00	0.10	69.89	0
115	0.22	0.02	3.67	4.60	-1.16	0.25	0.00	0.11	43.47	0
116	-0.56	0.02	-1.22	-1.49	-1.51	0.50	0.00	0.10	51.75	0
117	-0.76	0.02	-1.02	-1.16	-1.82	0.52	0.00	0.07	44.40	0
118	-0.02	0.02	0.50	0.69	0.23	0.43	0.22	0.10	49.49	0
119	0.24	0.02	-0.23	-0.66	1.85	0.66	0.38	0.06	60.98	0
120	-1.07	0.03	0.16	1.17	-4.06	0.35	0.00	0.07	70.95	0
121	1.25	0.02	8.47	8.87	4.79	0.71	0.35	0.09	49.36	0
122	0.16	0.02	0.77	0.67	-0.37	0.38	0.02	0.06	102.53	0
123	0.33	0.02	1.85	2.06	1.61	0.44	0.29	0.08	63.91	0
124	0.18	0.02	1.01	0.70	1.80	0.57	0.39	0.09	45.06	0
125	-1.39	0.03	-0.10	-0.07	-3.93	0.42	0.00	0.07	52.78	0
126	-0.64	0.02	-5.47	-9.28	0.03	1.19	0.12	0.16	38.85	0
127	-0.37	0.02	3.08	4.53	-6.19	0.16	0.00	0.10	48.63	0
128	-0.44	0.02	-2.29	-4.14	0.48	0.80	0.36	0.14	49.49	0
129	0.33	0.02	0.76	0.48	1.92	0.56	0.35	0.07	59.02	0
130	-1.09	0.03	0.86	3.22	-7.69	0.21	0.00	0.07	46.04	0
131	-0.53	0.02	-0.62	0.06	-1.87	0.43	0.00	0.07	74.66	0
132	-1.00	0.02	-0.81	-1.00	-2.44	0.49	0.00	0.10	56.18	0
133	0.26	0.02	1.24	1.40	-0.31	0.36	0.00	0.07	67.23	0
134	-0.73	0.02	-0.30	-0.28	-2.48	0.41	0.00	0.08	46.89	0
135	-0.33	0.02	0.34	0.64	-1.87	0.36	0.00	0.09	70.67	0
136	-1.04	0.03	0.54	2.49	-5.60	0.27	0.00	0.08	36.96	0
137	-0.67	0.02	-0.47	-0.79	-2.03	0.45	0.00	0.08	38.62	0
138	1.70	0.02	-2.59	-2.70	3.06	0.63	0.05	0.12	56.38	1
139	0.74	0.02	9.90	9.90	4.00	0.20	0.25	0.10	71.55	1
140	0.41	0.02	8.69	8.86	4.17	0.56	0.52	0.06	55.77	1
141	1.80	0.02	1.51	3.21	4.31	0.67	0.18	0.13	51.23	1
142	-1.85	0.03	-0.21	0.33	-5.08	0.42	0.00	0.06	35.40	0
143	0.02	0.02	3.56	4.61	-2.27	0.23	0.00	0.08	63.04	0
144	0.07	0.02	1.28	1.78	-0.99	0.33	0.00	0.08	137.62	0
145	-2.05	0.04	0.18	1.63	-8.00	0.28	0.22	0.07	24.12	0
146	-0.08	0.02	-0.85	-0.99	-0.42	0.47	0.07	0.08	169.50	0
147	2.85	0.03	2.21	7.69	8.00	1.31	0.13	0.10	72.98	0

Appendix B

Kernel	N	Split	R Diff	R SE	R Infit	R Outfit	3PL Diff	3PL Discr	3PL Lower	Q_3 Range	Avg Time
RBF	200	1	-10.25	15.07	-5.73	-3.13	1.59	12.09	17.95	-1.57	-5.84
RBF	200	2	7.92	-15.45	3.19	-2.91	1.26	-1.71	-3.78	-1.48	-6.87
RBF	200	3	-0.03	0.04	0	0.01	-0.01	0.02	0.03	-0.05	0.01
RBF	500	1	-6.94	10.23	-3.84	-2.08	1.2	8.26	12.27	-0.88	-9.95
RBF	500	2	15.62	-15.66	4.13	0.47	5.49	-6.39	-6.3	3.54	-4.05
RBF	500	3	-0.04	0.05	-0.01	-0.01	-0.02	0.03	0.02	-0.05	0.01
RBF	1000	1	-0.96	4.32	-0.63	-0.11	-0.84	2.02	-0.03	-0.43	-0.63
RBF	1000	2	3.92	-12.25	4.28	0.84	7.53	-6.18	4.56	2.57	-6.51
RBF	1000	3	-0.03	0.04	-0.05	-0.04	-0.03	0.03	-0.01	-0.1	-0.01
RBF	5000	1	-3.43	5.77	-0.94	-0.5	-2.3	3.33	0.71	-0.63	-2.34
RBF	5000	2	0.19	-0.26	0.08	-0.01	0.25	-0.05	-0.03	0.18	-0.22
RBF	5000	3	-11.53	8.65	5.68	8.22	-12.15	-6.72	-5.36	-6.54	-0.24
RBF	13584	1	-1.26	4.46	-0.54	0.17	-0.39	2.57	-0.43	-0.75	-0.68
RBF	13584	2	7.44	-13.11	0.57	-7.36	11.81	-2.42	9.84	6.9	-4.28
RBF	13584	3	-9.08	6.3	2.13	2.67	-9.53	-2.04	-7.71	-8.42	-2.21
POLY	200	1	0.3	-0.99	-0.41	-0.2	-0.16	-0.31	-0.19	-0.48	0.26
POLY	200	2	0.02	0.02	0.03	0.01	0.01	0.03	0	0.01	0.01
POLY	200	3	0	0	0	0	0	0	0	0	0
POLY	500	1	-0.07	0.07	0.02	0.06	-0.04	-0.01	0.06	0.01	-0.11
POLY	500	2	0.37	-0.28	0.22	-0.12	-0.74	-0.39	-1.52	0.36	-0.25
POLY	500	3	-0.01	0	-0.01	0.01	0	0.01	-0.01	0	0
POLY	1000	1	-0.03	0.06	-0.02	-0.02	0	0.02	0.07	-0.01	-0.18
POLY	1000	2	0.16	-0.22	0.07	0.13	0.06	-0.06	-0.03	-0.17	0.22
POLY	1000	3	-0.73	0.62	0.25	0.11	-0.47	-0.15	0.13	0.09	-0.22
POLY	5000	1	0.03	-0.15	-0.01	0.03	-0.04	-0.12	-0.09	-0.02	0.01
POLY	5000	2	-0.31	-0.19	0.12	-0.03	-0.18	-0.07	0.22	-0.02	-0.63
POLY	5000	3	0.02	-0.03	-0.22	-0.19	0.17	0.29	0.25	0.29	0.07
POLY	13584	1	0.24	-0.21	0.25	0.3	0.18	-0.19	0.08	0.07	-0.16
POLY	13584	2	-0.61	0.33	0.99	0.76	-0.58	-0.36	-0.29	0.03	-0.25
POLY	13584	3	-0.08	0.01	-0.03	-0.03	-0.07	0	-0.04	0.08	-0.02