# ROBUST MODELS AND EVALUATION FOR SYSTEMS SECURITY RESEARCH

**BY SHRIDATT SUGRIM**

**A dissertation submitted to the**

**School of Graduate Studies**

**Rutgers, The State University of New Jersey**

**in partial fulfillment of the requirements**

**for the degree of**

**Doctor of Philosophy**

**Graduate Program in Electrical and Computer Engineering**

**Written under the direction of**

**Janne Lindqvist**

**and approved by**

________________________________________

________________________________________

________________________________________

________________________________________

________________________________________

**New Brunswick, New Jersey**

**October, 2020**

## ABSTRACT OF THE DISSERTATION

# Robust models and evaluation for systems security research

**by Shridatt Sugrim**

**Dissertation Director: Janne Lindqvist**

Machine learning in modern systems security research is common. Researchers regularly use machine learning based models for tasks such as authentication and user identification. Often the practices followed for developing and evaluating a machine learning model that forms the decision logic of these systems are misleading. For example, the maximum accuracy (ACC) can be inflated when the data used to train a model is class skewed. Additionally, models built on data from small user groups may achieve high performance values but fail to generalize in a larger population. These inflated performance values will lead to unexpected system-level failures.

There are several metrics that are used to evaluate how well a system performs at the task of distinguishing users. Existing metrics are often inadequate because they fail to capture the range of possible contingencies that arise when the measurements that decisions are based on have inherent ambiguities. These ambiguities can result in mistaking one user for another. For authentication or user identification, the consequences for such mistakes is dictated by the target application. Mistakenly granting access to a bank account has significantly different consequences than loading the wrong set of user preferences. Many of the common metrics hide underlying problems within the machine learning models. Models that are not tested with an adequate number of users can fail in surprising ways.

In this PhD thesis, we explore the underlying reasons why the metrics are misleading, and models fail to generalize. We identify the flaws with the metrics and show how some metrics can degrade in performance when assumptions about the number of users are violated. We present surveys of proposals for new authentication or user identification systems from top-tier publication venues. We found that 94% (33/35) the authentication systems surveyed had reporting flaws and 77% of user identification systems used less than 20 participants to validate their system. Finally, we present solutions to these issues in the form of metrics that can be visually checked for flaws and testing methods that can be used to determine when assumptions about population size break down.

# Acknowledgements

No man is an island entire of itself [1], and it takes a small suburb to grow a Phd. This journey has been so long and there are so many to thank, it's not possible to fit them all in these pages. To everyone who has ever taught me something, you have my deepest thanks.

Folks at Rutgers that were a great influence include: Ivan Seskar, Rich Howard, Melike and Kemal Gursoy, Predrag Spasojevic, Richard Gundy, John Kolassa, and many others.

I am grateful for the support of my family: Gomatie Sugrim, Dowlat Sugrim, Chandrakia Sugrim and Larry Averick. I was also supported by many friends and colleagues including: Ben Firner, Jenny Shane, Dragoslav Stojadinovic, Teajaswhy Hari, Haris Kremo, Sanjit Kaul, Bin Zan, K. C. Huang, Kahn Le, Francesco Branzino, John Austen, Roy Montalvo, Rick Anderson, Robert Miller, and countless more.

I am indebted to all my co-authors: Can Liu, Meghan McLean, Xianyi Gao, Victor Kaiser-Pendergrast, Yulong Yang, Michael Sherman, Gradeigh Clark, Arttu Modig, Antti Oulasvirta, Teemu Roos, Hyeong-Sook Park, and Youn-Ok Park for helping make the publications happen. This material is based upon work supported by the National Science Foundation under Grant Number 1750987.

A special thank you to all the ECE folks that helped me travel along my path including: Wade Trappe, Narayan Mandayam, Yanyong Zhang, Athina Petropul, Zoran Gajic, Chris Rose, and Christy Lafferty. I'd also like to remember Zoran Miljanic, David Daut, and Etian Fenson, all of whom we lost too soon.

Finally, without the guidance and encouragement of my advisor, Janne Lindqvist, a dsylexic kid from the Bronx would have never come this far. This is not the end of our collaboration, it's the beginning.

# Dedication

I dedicate this work to my mother and father, Gomatie and Sugrim Sugrim. A seed of curoisty planted decades ago, still continues to grow.

# Table of Contents

# Chapter 1

# Introduction

## 1.1 Overview

All systems that make decisions have a possibility of making the wrong decision [2, 3]. Unfortunately, the correctness of a decision is dependent on the context the decision is made [4]. When the model a system is based on captures the context correctly, then the decision made will align with the expectations of the systems behavior. This contextual knowledge is built into the model a system uses to make decisions by the choices that are made during design and evaluation [5, 6]. For example, decision performance can be measured via a multitude of metrics, each capturing some aspect of how well decisions are made. Since all metrics are a summary of performance, they only give a limited view of the system performance. The context in which the system is evaluated will make some metrics more applicable than others. Thus, the conditions on which a system was designed and tested shape the system's model. If the test conditions do not align with the conditions the system are used under, then the performance metric values acquired in testing will not hold in the realized system. This disparity between testing and usage can lead to unexpected system behavior and ultimately impacts the systems security because the system is unpredictable when used outside it's expected environment.

It is rarely the case that proposed systems are tested beyond the initial evaluation done in the literature. Between 1994 and 2010 Silvia et al. [7], identified only 96 out of $16,055$ papers in the software engineering community that replicated an existing study. It is well accepted that other communities have similar study replication numbers since replication of results is usually considered unpublishable [8]. Therefore, it is important that publications be upfront about the limitations of the system they propose. While much of the literature is focused on developing the model, very little is focused on identifying the limitations of the model. The limitations are just as important because they are often needed to explain the failures of the system. Often consumers of the research are not in a position to make judgments about the possibility of system failures in their context because the publications present only the most interesting / novel (i.e publishable) aspects of the research. The published literature is often lacks the details required to enable replication or implementation.

Because the decision systems are often used as a black box, the literature tends to focus more novel inputs to or usages of, the classification algorithms. Performance reporting often follows defacto standards inherited from the literature body which the publications draw from. There is however no consensus on what metrics should be reported (see Section 2.2). It is rarely the case that systems are tested to determine the at what point do the assumptions of the model no longer hold (see Section 3.2).

Since the task of testing for robustness falls to the proposer of the system, in this PhD proposal we consider two simple questions that are often overlooked when reporting the details of a system. These questions are:

- Given several proposed systems, which one performs the best in my target context?
- What is the limit on the number of users that my system can handle?

To answer the first question we consider the problem of how to measure performance of an authentication system, i.e., a system that performs binary classification of users into authorized and unauthorized classes. Different performance metrics attempt to summarize different aspects of the system performance. Not all metrics are applicable in all contexts and each reported metric must strike a balance between the amount of paper space used, the ability to compare with existing literature, and the amount of information conveyed. Metrics should be chosen not only by what they convey, but also what they leave out. Some metrics convey more information but are more difficult to interpret. Single number summaries like the equal error rate (EER), are easy to compare but hide many details of system performance. These summaries should not be the only performance reporting, instead they should be part of a complete story of performance, that conveys the possible trades offs in error rates one can make by adjusting the parameters of the system. In Section 2, we argue for the reporting of the frequency count of scores (FCS) to augment an receiver operator characteristic (ROC) curve. We show that these metrics together enable side by side comparison of the range of system performance under the breadth of possible conditions (instead of the idealized ones that maximize a specific summary).

To answer the second question we consider the problem of identifying the limits on the number of users that can be identified with a bound on error for an identification system, i.e., a system that uses multi-class classification to identify distinct users. Thus the relevant performance metrics are different for these types of systems. However, the development process for these systems is very similar to those used for authentication systems. An initial step in the process is to identify features to extract from the measurements of observable properties of users.

The features extracted from the measurements are chosen to be representative of the physical phenomenon the system is trying to model. Features are validated with user studies that ultimate produce performance metric values. If the performance results don't meet the desired goals, the design process is iterated potentially adding more preprocessing, or choosing new features (e.g different functions of the measurements). This iteration repeats until the desired performance targets are met [9]. What is often constant in these studies is the user data sources. These source of data often have low user counts.

Since recruitment and collection can be expensive, it is often done only once. This is not ideal because the iteration of the model development process causes the model to becomes more specialized to the current set of user data, i.e., the model becomes overfit. We argue that systems should be tested with a reasonably sized number of users to identify the models sensitivity to the number of users in the system. Since it is often difficult to determine what an appropriate limit on the number of users is, we recommend an iterative approach to testing which uses subsets of the user base with increasing size. We show that this approach determines how sensitive the performance metrics are to the number of users in the system.

Because cheap computation is omnipresent, it is of great importance that our statistical tools evolve to take advantage of this resource. This process is already occurring in some areas of machine learning. As an example, consider how model selection in done today as compared to approaches of the past. Model selection was traditionally done by selecting a distribution family and then finding the parameter value that maximizes a divergence based metric (e.g. Akaike information criterion) [10]. This method required expertise in distribution forms and asymptotic behavior of functions to be employed correctly. However such methods have been replaced with more computationally expensive but easier to interpret tools such as cross validation [11]. The metrics we propose follow this trend of leveraging modern computational availability to lower the amount of expertise required to use statistical learning tools.

## 1.2 Contributions

In this body of work we determined what were the underlying causes for the observed failures. Beyond examining the underlying details of the developed models our work had the follow contributions:

- We surveyed literature from appropriate venues to demonstrate that the issues we raise are present in the literature.
- We show how commonly used metrics in authentication systems, including TPR, FPR, EER, AUROC, ACC, and GC are inherently flawed metrics for understanding authentication system performance.
- We show how any single-number summary provides incomplete information for the evaluation of authentication systems.
- We propose unnormalized Frequency Count Scores (FCS) as an augmentation to current authentication metrics that enable visual comparison and identification of some errors.
- We show how using the FCS with the ROC can further solve the limitations associated with current authentication system metrics.

- We demonstrate flawed comparisons with several existing proposed authentication systems by reimplementing those proposed systems.

- We examined the difficulty of obtaining high recognition performance in identification systems with low participant counts.

- We mimicked the evaluation procedure of an identification system by constructing five distinct systems based on publicly available datasets, each with 20 participants. We used support vector machines, random forests, and neural networks on these datasets and achieved greater than 90% accuracy in three cases.

- We explored the reasons why the low participant count are not adequate to evaluate identification systems.

- We examine the impact of participant count on performance results and observed that performance may degrade as participant count increases. In addition to participant count, we also considered the impact of measurement count, number of measurements per participant, and feature dimension on performance.

- We propose measuring how the performance metrics degrade when the participant count increases as a gauge of the robustness of an identification system.

- We examine how to anticipate this degradation by using randomized participant subsets and note that this is a crucial criteria to demonstrate the stability/security of a novel identification system.

# Chapter 2

# Robust Performance Metrics for Authentication Systems

## 2.1 Chapter Overview

Many authentication systems utilizing machine learning have been proposed (see Table 2.2). However, there is no clear agreement in the community about how these systems should be evaluated or which performance metrics should be reported. Specifically, publications often report misleading single-number summaries which include true positive rate (TPR), false positive rate (FPR), equal error rate (EER), area under ROC curve (AUROC), and maximum accuracy (ACC). Figure 2.1 enumerates the reporting rates of common metrics for thirty-five recent publications.

Improving the metrics and reporting methods can resolve two primary obstacles to the evaluation of authentication systems. These obstacles are (1) skew in the distributions used to train and evaluate the systems, and (2) misleading comparisons that arise from the reported metrics. For example, skew within the population of study participants can artificially inflate the maximum accuracy. Additionally, misleading comparisons can result from commonly reported metrics. For example, it is inappropriate to conclude that one system performs better than another by comparing an EER of 0.05 to 0.10. Similarly, an accuracy of 80% versus 90% does not allow clear inferences about the system performance.

We show the following three primary flaws with existing metrics: 1) It is incomplete to report performance using solely single-number metrics, like the ACC, FPR, TPR, FAR, and FRR. Single-number summaries hide the details of how and what errors occurred. For example, because if a system was trained on mostly unauthorized users' data, it will learn to recognize unauthorized users very well and may not recognize authorized users. 2) Reporting performance results without the parameters of the

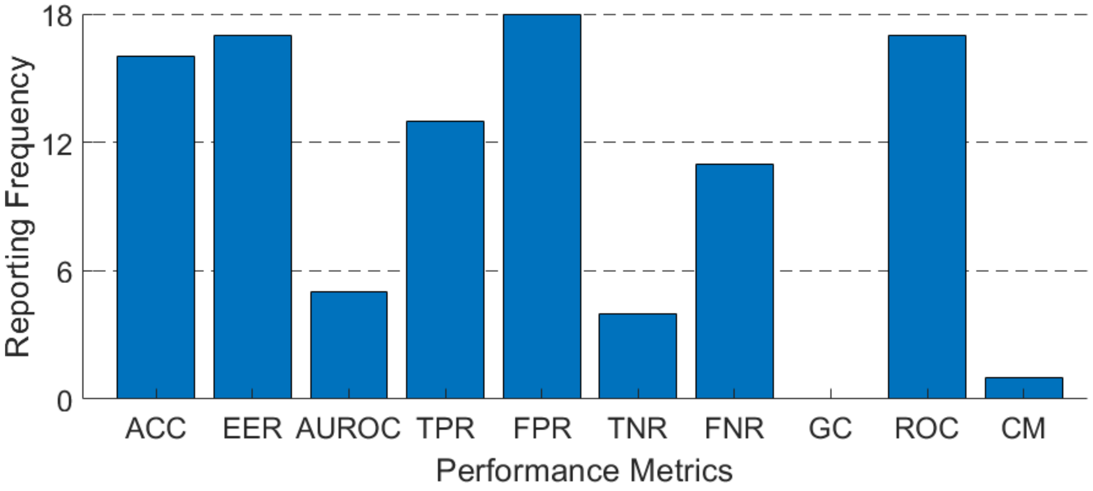


Figure 2.1: The frequency of reported metrics from Table 2.1 for thirty-five recent publications surveyed from venues listed in Table 2.2. Classical biometric and detection summaries such as EER and TPR are often reported because they are widely used in the literature. The FPR is the most reported because it is the common element for two frequently reported metric pairs, the (TPR,FPR) and (FPR,FRR).

model hinders the implementation of the system. The system cannot be faithfully replicated when only the performance is reported. 3) Performance comparisons cannot be made when using single-number summaries derived from the ROC. One cannot conclude that one system will perform better than another in a target application by direct comparison of the EER and other ROC-derived summaries.

In this work, we uniquely propose and demonstrate how the ROC, combined with the unnormalized Frequency Count Scores (FCS) (shown in Figure 2.2), aids in the ability to understand the trade-offs for authentication performance and adequately evaluate the proposed approach.

## 2.2 Review of recent authentication systems

To determine the current state of performance metric reporting, we surveyed recent research published in top venues. The selection criteria for including papers in the review was the following: (1) The article was published in a top venue for systems security, mobile computing, human-computer interaction, or pattern recognition for authentication. These venues included NDSS, CCS, CHI, IMWUT/UbiComp, INFOCOM, MobiCom,

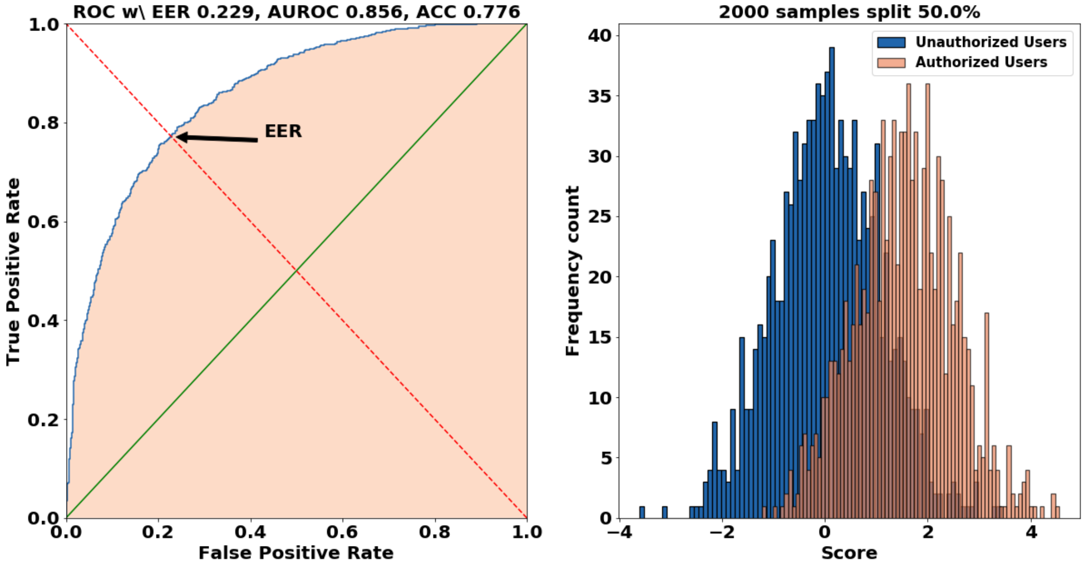


Figure 2.2: The left figure is the ROC curve of an authentication system. The right figure is its corresponding FCS that displays score distributions for measurements of authorized and unauthorized users. The ROC curve is computed by varying the value of the threshold required to grant access and computing the true positive rate (TPR) and the false positive rate (FPR). The FCS is a histogram of measurement scores separated by ground truth. In this FCS figure, the blue histogram represents unauthorized users' scores, determined by the ground truth of the measurement. The red histogram in the FCS figure represents authorized users' scores.

MobiSys, SOUPS, SS&P (Oakland), USENIX Security and Pattern Recognition journal. Machine learning venues (e.g. NIPS) were not included due to their primary focus on algorithms and lack of attention to system applications. (2) In order to evaluate current practices, the paper had to be published within the last 2 years (2016 to 2018). (3) The paper had to propose an authentication scheme. Specifically, the paper had to use machine learning to label users as authorized and unauthorized (as opposed to identifying users from a group). Although we identified many related papers ($n = 58$), only 35 proposed an authentication scheme and were included in the review. We note that we did not find any publications matching our criteria from USENIX Security.

In order to find these articles, one researcher used the Google Scholar search engine to limit results to these venues and included the following search terms: authentication, behavioral, biometric, machine learning, password, recognition, and access. A second researcher separately reviewed the venue proceedings using the search terms in order to generate a complete list of related work.

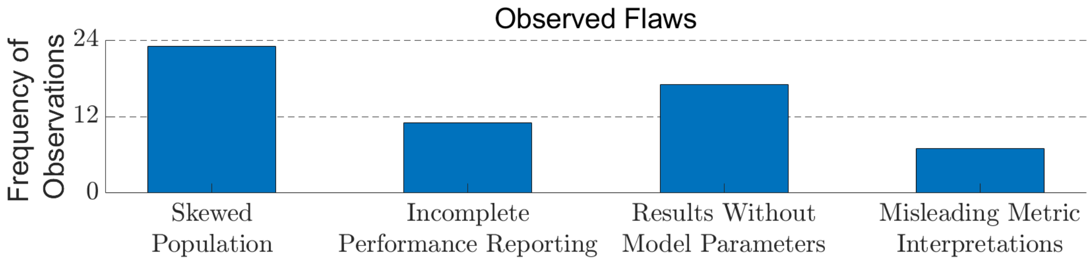


Figure 2.3: The flaws noted in Section 2.2.1 occur several times in many of the publications. We note that the common practice of recruiting $N$ participants and then electing one as the authorized user often leads to skewed measurement populations. Twenty-three of the thirty-five papers surveyed had skewed measurement populations.

In the thirty-five publications that were surveyed, we discovered no uniform approach for reporting metrics. However, there were several recurring themes. Figure 2.1 shows that the most common metric reported is the FPR since the FPR is the common element in two different but related metric pairs: the (TPR,FPR) and (FPR,FRR) pairs. These pairs are often reported when one value is held constant and the other is minimized (e.g. fixing the FPR and adjusting the system parameters until the TPR is optimized). There is often no justification for the value that is chosen to be held constant. Another frequently reported metric is the EER. It is often reported for comparison with existing systems in the literature. Unfortunately, without a uniform approach, we cannot make comparative quantitative conclusions about the performance across all the proposed systems.

Sixteen (less than half) of the publications reported the ROC. Eleven of the publications that reported the ROC had measurement populations that were skewed (here measurement population is defined as the set of measurements from users). In some of these cases we concluded that the effects of the skew did not impact the validity of the claims because the performance claims were based on the ROC.

### 2.2.1 Common Flaws

We observed three flaws that were common to many publications. The three flaws are as follows:

### Flaw 1

*Incomplete performance reporting.* Reporting based solely on single-summary metrics is incomplete. For example, the maximum accuracy (ACC) metric does not identify the type of user (authorized or unauthorized) an error was made on. For example, an accuracy of 90% does not mean that the system makes errors 10% of the time. A system may have been tested on data it would almost always get correct. Specifically, if a system was trained on mostly unauthorized users' data, it will learn to recognize unauthorized users very well. If this system is only tested on unauthorized users, the ACC metric will be very high. However, because this system was built on only unauthorized users' data, it may not recognize authorized users very well. The system will not be trained to identify authorized users' data because it has no model for authorized users. The model is thus incomplete when trained on data that is either mostly authorized or unauthorized users.

Comparisons made solely based on TPR, FPR, or FRR are also incomplete. It is not possible to tell if the optimized value is the result of better discrimination between users or simply an adjustment of system parameters for the purpose of inflating a metric. Since the (TPR,FPR) and (FPR,FRR) are the result of a specific compromise between the two kinds of errors. One can trade one error type for another by adjusting the parameters (e.g. threshold), without improving overall performance.

It is imperative that authors incorporate this knowledge into the interpretation of their metrics. If the authors do not report the frequency of authorized and unauthorized users' data, then the ACC metric provides little information about the system. Of the thirty-five publications surveyed, eleven directly exhibited this flaw. Several others made performance claims based only on one of these metrics but also reported other metrics for comparison.

### Flaw 2

*Results without model parameters.* Performance results reported without the parameters of the model hinders the implementation of the system. The confusion matrix (CM), and all metrics derived from it (e.g. the ACC or TPR), depend on the threshold used to obtain the results. For metrics derived from the confusion matrix, the system cannot be implemented when only the performance is reported. Researchers tend to determine their own thresholds in isolation when they design a system (they cannot know what an implementer would need). If the threshold is unknown, the conditions under which the original research was completed cannot be replicated. We recommend that the ROC and FCS are reported to surpass this limitation. The ROC and FCS show how the system responds to changes in the threshold and covers a range of possible thresholds enabling implementers to choose the threshold that is right for their application.

Every machine learning algorithm has parameters that, when adjusted, significantly change their behavior (e.g. slack factor, size and number of hidden layers, K). Since machine learning usage is often off-the-shelf, it is easy to overlook the necessary parameters required to use the system even though these parameters control the behavior of one of the most critical parts of the system. In our survey, seventeen of the thirty-five publications left out the parameters of the machine learning algorithms used. Without these parameters, the task of replicating the system becomes much more difficult.

In one case, a publication studied the effects of environment on the system performance by comparing the accuracy of the system in several environments. Based on the described methods, the test data had a positive bias as the authors elected a very small subset of the participants to be attackers. The publication did not disclose whether the threshold remained the same in all environments tested. The authors concluded that since the accuracies were less than 5% from each other, that the environmental effects were negligible. However, because the population was skewed, that less than 5% difference could also be accounted for by shifting the threshold in each of the environments. Since the data was mostly comprised of positive samples from authorized users, moving the threshold up or down could yield more or less correct decisions purely due to sampling effects.

### Flaw 3

*Misleading interpretations.* Performance comparisons cannot be made when using single-number summaries derived from the ROC. Direct comparison of the EER (and other ROC-derived summary metrics) does not show whether one system performs better than another. Two systems with similar values for these metrics can have very different ROCs. Because the ROCs are different, the performance will be different when implemented. When a system is implemented, it must be implemented with a target application in mind (e.g. banking or loyalty discounts). These different target applications come with requirements for the amount of false positives (wrong people allowed in) or the amount of false negatives (correct people kept out) that the application can tolerate. In our survey, seven of the thirty-five papers drew direct comparisons about system performance by comparing one of these summary values. Twenty-two of the thirty-five papers reported one of the metrics with the expectation that similar systems could be compared using these summaries.

Systems with different ROCs behave differently when implemented. These differences can lead to unexpected behavior because the error rates of the implemented system may differ from the error rates the implementer expected. As a toy example, consider a case in which an implementer desired to improve the performance of an application that used Touchalytics [47] and required the false positive rate to be 0.1. In this scenario, the implementer may consider switching to SVC2004 [48]. The implementer may look at the EER of SVC2004 which is 0.185 and compare this to the Touchalytics EER of 0.198. From this comparison, an implementer would conclude that the system's performance will improve. He or she would be unpleasantly surprised when their true positive rate dropped from 0.8 to 0.6 at the false positive rate their application required. Therefore, comparing systems on these metrics can lead to detrimental, real-world security consequences.

In several of the surveyed publications the EER and AUROC were used in two different ways. In the first use case, authors made direct claims about the relative behavior of two or more systems either by comparing the proposed system to an existing system which uses the same types of measurements or by adjusting the parameters for their own measurements, to determine the impact on the metric. In some cases, authors concluded that a change in parameter had no impact on performance because the metric was unchanged. However, because they did not include the ROC, we do not know what effect the changes had on the ends of the curve. In contrast, some authors reported the ROC for multiple parameterizations of their system demonstrating that their systems behavior was predictable over a wide range of parameterizations, even though they did not make this claim.

In the second use case, the EER or AUROC is reported with the expectation that these metrics will be used to compare the proposed system to other competing systems. This use case only allows for naïve comparisons such as those detailed in Section 2.2.2. Since the second use case requires knowing what the authors intended the reported metric to be used for, we did not count these cases as evidence of flaw 3.

### 2.2.2 A naïve comparison

Table 2.3 enumerates the top five systems based on the reported performance metrics. We consider the three most common metrics: the EER, ACC, and FPR. For any single metric, this comparison fails to produce a meaningful result for several reasons.

- Not all publications report the same metric. Comparison across different metrics does not have a meaningful interpretation.
- Individually using any of the three above-mentioned metrics can lead to flawed conclusions because no individual metric captures the complete performance.
- Just because a system optimizes a metric, does not mean that it can be utilized in the target application (e.g. an approach implemented for keyboards may not work on touchscreens directly).

It is clear that such a naïve comparison cannot lead to an informed comparison of the proposed systems. It is even difficult to identify if a system is suitable because some of the metrics fail to provide information that is relevant to the context of the target application (e.g. an FPR may not be achieved at a target TPR).

Of importance, system evaluation requires the ability to evaluate the potential security trades-offs of a system. Instead of answering the question, “Has the system produced a single metric that has surpassed a seemingly adequate threshold?” implementers need to answer the question, “Can this system be tuned to meet the needs of my application such that reported metrics show possible security trade-offs?” Many of the current metrics that are reported fail to answer the latter question.

*These flaws occur in many of the publications surveyed.* Figure 2.3 enumerates the observation frequency of each of the described flaws. We also note that almost two-thirds of publications have skewed measurement populations. It is common practice to recruit $N$ participants, take $M$ number of measurements from each of them, and then elect one participant as the authorized user. When this is done, the measurement populations are skewed because there are $N - 1 \times M$ measurements from unauthorized users, and only $M$ measurements from authorized users. In some cases, we were unable to assess whether the measurement population was skewed because the publication did not report the sources of measurements. This was the case in *23 of the 35 publications* we reviewed. However a few reported balanced accuracy in an attempt to compensate for the skew. As we will see, had the FCS been reported, we would have been able to visually assess if there was measurement population skew. One of the publications actually reported a normalized FCS but did not use it for analysis.

## 2.3 Related Work

We discovered only one publication in the systems security community that has studied how performance metrics are reported [49]. They studied flaws that occur in reporting for continuous authentication systems. They note that the EER is among the most popular metrics and observe the misleading characteristics of only reporting the EER and false negative rate (FNR). They also note that data sets are rarely made available, which creates a barrier to follow-up analyses. They additionally advocate for the use of the Gini Coefficient (GC) which is functionally related to the AUROC. We show that the GC and AUROC are also flawed metrics that hinder the comparison and implementation of a system and we instead advocate for the combination of FCS and the ROC.

Bonneau et al. [50] compared different types of authentication systems to text passwords with qualitative metrics in usability, deployability, security. They provide a wide range of real-world constraints but they did not provide quantitative approaches to evaluate the metrics. In contrast, we focus on quantitative metrics in this paper.

The efficacy of the EER in communicating performance has been questioned in other fields [51]: the EER has the significant disadvantage of representing only a single point in the performance continuum and that this misrepresents the capabilities of a system. The paper [51]: argues for the ROC as the main performance metric but does not consider how measurements are separated, nor the utility of looking at score range overlap (we will address how these factors limit the utility of the ROC). Others [52, 53] have argued for using the ROC curve over the accuracy as a performance metric. Papers from several fields, including clinical medicine [54], chemistry [55] and psychiatry [56], have been arguing for the use of the ROC. Although many disciplines call for the usage of the ROC, the interpretation and consequences of a classification error are distinct to each discipline. In our work, we focus on classification error in the context of authentication and show how the ROC alone is an inadequate metric.

Prior research has used normalized histograms to estimate score distributions [5]. This approach is fundamentally different from what we propose. We propose that an unnormalized metric - the Frequency Counts of Scores (FCS) can be used to diagnose security flaws for authentication systems. This approach is not widely known or applied.

Some of the flaws we discuss may be known in the machine learning community. For example, previous research in machine learning has discussed population skew [53]. However, our work clearly shows that our suggestions are unknown in this context and novel to the authentication systems community. Thus, it is imperative that the flaws and our proposed recommendations are discussed.

In summary, using EER as a performance metric has been questioned in continuous authentication systems and other fields. However, there is no work that propose a convincing alternative metrics to EER. We are the first to propose the FCS in addition to the ROC to augment the comparability of authentication systems. Although prior texts discussed the normalized histogram for estimating score distributions, we are the first to use the unnormalized FCS for diagnosing security flaws of authentication systems. The FCS addresses the deficiencies in the availability of data for analysis by enabling analyses to be done on the distribution of scores. This is true even in cases where the data may be sensitive and cannot be made available to the public. The FCS can be used to directly identify thresholds that fit the application criteria. Analysis of the scores can give insight into the modifications a score function might need to achieve better separation of users. With FCS, we can identify two types of flaws in the surveyed publications in top venues: incomplete performance reporting and misleading metric interpretation.

## 2.4 Machine Learning in Authentication Systems

Authentication systems that utilize machine learning can use a variety of methods to distinguish users (e.g. fingerprints, visited locations, and keystroke dynamics). Regardless of the authentication method, the machine learning methods used to classify the measurements are the same. Figure 2.4 shows how machine learning in most authentication systems includes three major operations: preprocessing, scoring and thresholding.

In the preprocessing operation, the measurements are filtered, re-centered and scaled [4]. This operation may discard measurements that fail to meet any admissibility criteria the authentication system may have (e.g. measurements that are too short). Scoring applies a mathematical function ($f : M \to \mathbb{R}$) to the measurements to generate a numerical summary of the measurements (by numerical summary, we mean a number that is used to describe a characteristic of a dataset). Scores of measurements from authorized users by convention score higher (to the right of) than those of unauthorized users [57].

The scoring operation is the most critical part of the authentication process. Scoring measurements well enables unambiguous classifications by separating measurements. The better scoring is at separating measurements, the fewer errors will be made. The scores between different authentication systems are rarely comparable and bare no

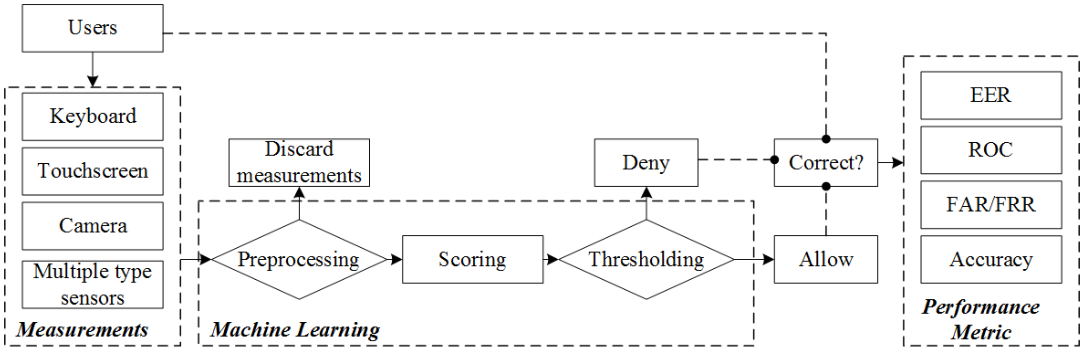


Figure 2.4: The use of machine learning as a classifier is common to many authentication systems. The preprocessing phase prepares measurements by filtering, re-centering and scaling. The scoring phase applies a mathematical function to the measurements to map them to a number. The thresholding phase compares the number to a fixed threshold to make a decision. The full authentication system feeds measurements taken from the user to the machine learning classifier and then evaluates the performance based on the decisions that come out of it.

direct relevance to each other, even if the systems measure the same thing.

Thresholding uses the score (a numerical summary) as evidence for a decision. The choice threshold establishes the minimum required score to be deemed authorized. For any user's measurements, if the score is below the threshold, the user will be denied access. Similarly, if the user's score is above the threshold, the user will be granted access. The further away the score is from the threshold, the more confident we can be in our classification. Thus, user measurements that score significantly higher than the threshold are considered strong evidence for a decision to grant access. User measurements that score significantly lower than the threshold yield a confident decision to deny access. In this sense, the choice of threshold dictates the strength of the decision.

### 2.4.1 How authentication systems research is consumed

While there is no formula for implementing a proposed system from its publication description, there is a common theme that many publications follow. Many publications start with a description of what is measured and why it is important to be measured. A case then is made for why the measured quantities will produce good performance or have some additional benefit (e.g. easy to remember, resistant to some types of attacks, and require fewer resources). A classification algorithm is typically chosen based on criteria such as ease of implementation or good performance with available data for a chosen metric. Finally, a user study is performed to validate the design choices made, demonstrate the claims of utility or defensibility, and potentially compare to existing systems.

An implementer of these systems will have to determine what was measured from the description, and then collect those measurements. The implementer will then need to use the classification and compare the performance metrics achieved by their implementation against those reported in the publications. The implementer will have to pick a system based on a comparison of the reported performance values, the ability to recreate the measurement apparatus (e.g. collecting heart rhythms or breath sounds), and applicability of the system's benefits to their specific case (e.g. the need for resistance to shoulder surfing). As we will see, comparing performance values between publications is often challenging for a variety of reasons, complicating the process of choosing a system. Implementing the classification algorithm can also prove daunting because the descriptions are often inadequate (e.g. often lacking critical parameter values).

### 2.4.2 How classifiers work

The performance of a classifier is influenced by two major factors. The first is how well the scoring function separates the measurements from different users and the second is how well the threshold is chosen.

The scoring operation plays the most important role in the authentication system performance. The score function's ability to separate measurements via their score values reflects the underlying capability of the measurements collected to separate users. The score function can be seen as extracting information, in the form of score separation, from the measurements. If all the measurements from authorized users are distinct from the measurements of unauthorized users, an optimal score function was achieved. If the distinction between authorized and unauthorized users is inadequate, the score function will be inadequate.

The thresholding operation comes immediately after the scoring operation in importance. The selection of a threshold represents the choice of a compromise between error types that a classification system can make [3]. It cannot eliminate error; it can only trade one type of error for another (e.g. decrease the error of authorizing illegitimate users by increasing the error of denying legitimate users). If the scoring function provides good separation, there will be many choices of threshold that yield a good compromise between the error types. Several of the metrics (e.g. EER and ACC) fix a specific threshold and derive a performance metric value from this fixed point. The threshold is often chosen to optimize this metric and it is only this optimized value that is reported.

### 2.4.3 Why authentication systems make mistakes

From a security standpoint, what distinguishes one system from another is not the measurements they collect, but how well they tell authorized users apart from unauthorized users. For example, the problem of granting access to the wrong person has severe consequences if the target application is banking. On the other hand, the problem of denying access to the correct person is not a significant infraction if the target application is loyalty discounts.

Measurements collected from users are often random with an unknown distribution. When these random measurements are fed into a scoring function, the resulting scores will also be randomly distributed. These variable scores will be compared against a fixed threshold and a decision will be made to grant or deny access based on this comparison. If the randomness of the measurements causes the score to incorrectly cross the threshold, an error is made.

The scores of measurements from authorized users score higher than measurements from unauthorized users; therefore, if a score falls above the threshold, it is assumed to have come from an authorized user. If a score is below the threshold, it is assumed to have come from an unauthorized user. In the ideal case, all authorized users' measurements will score much higher than those of unauthorized users. This, however, is rarely the case. Often the scores from both types of users overlap (see Figure 2.2) because of the randomness in the measurements. The greater the overlap between scores, the more likely it is that the system will make a mistake and thus make more errors [2].

## 2.5 Common performance metrics used in authentication

For every decision a classifier must make, there are four possible contingencies: (1) authorize a legitimate user (true positive or TP), (2) authorize an illegitimate user (false positive or FP), (3) deny an illegitimate user (true negative or TN), and (4) deny a legitimate user (false negative or FN). The decision counts (TP, FP, TN, FN) are the fundamental components of all performance metrics. To compute these counts, the authentication system is used on a set of measurements where the ground truth is known. Once the scoring and thresholding is complete, the authentication system will produce a set of decisions based on those measurements. The counts are then computed by comparing the decisions with the ground truth.

There are two families of metrics which differ in the metric from which they are derived (shown in Figure 2.5). The families are: (1) confusion matrix (CM) derived metrics which depend on the threshold and (2) ROC curve derived metrics which may not depend on the threshold. The CM is a count of all possible contingencies arranged in a grid. This contingency table is computed for a fixed value of the threshold and thus depends on it. All related metrics inherit this dependence. Many of the other performance metrics are ratios of the counts enumerated in the confusion matrix (e.g. ACC).

The ROC represents many confusion matrices under varying values of the threshold and thus does not depend on it. Metrics derived from the ROC may be specific points on the ROC, such as the EER, or functions of the ROC (e.g. AUROC). Since the EER is a specific point on the ROC, it corresponds to a specific value of the threshold. Figure 2.5 shows the relationships between the CM related metrics and ROC curve related metrics.

### 2.5.1 Confusion Matrix (CM) related metrics

Table 2.4 shows a confusion matrix and the related metrics derived from it. The true positive rate (TPR) and false positive rate (FPR) are two key metrics that are computed from a confusion matrix to evaluate authentication system performance. TPR is interpreted as the probability that an authorized user will successfully authenticate and FPR is the probability that an unauthorized user will successfully authenticate. FPR is sometimes called the false accept rate (FAR). Other ratios often reported include the false negative rate (FNR) which is alternatively called the false reject rate (FRR), and the true negative rate (TNR).

The maximum accuracy (ACC) is another key metric for authentication system performance. It is interpreted as the relative frequency of a correct classification of a measurement source, regardless of its origin. *Since the accuracy is a function of the threshold, often the value of accuracy that is reported is the maximum across all thresholds.* The maximum accuracy represents the best performance the classifier can offer, however, solely reporting accuracy can be misleading. Because only a single threshold is represented in this performance metric, consumers of the research cannot know how the accuracy will change if the threshold changes. This may lead to the conclusion that a system is unfit for an application because the accuracy achieved is below an error requirement even though a judicious choice of threshold would satisfy an FPR requirement (at the cost of some TPR).

#### Other accuracy metrics

The maximum accuracy is not the only accuracy metric reported. There are several other metrics that are functions of the values across the columns of the CM, such as the balanced accuracy (BAC), $F_1$ score and half total error rate (HTER). Some of these metrics, such as BAC and HTER, attempt to weight the ratios to adjust for skews within the measurement populations. We note that these metrics are also functions of the confusion matrix, and thus still dependent on the value of the threshold. These metrics are reported less frequently than others considered in this paper, and they share

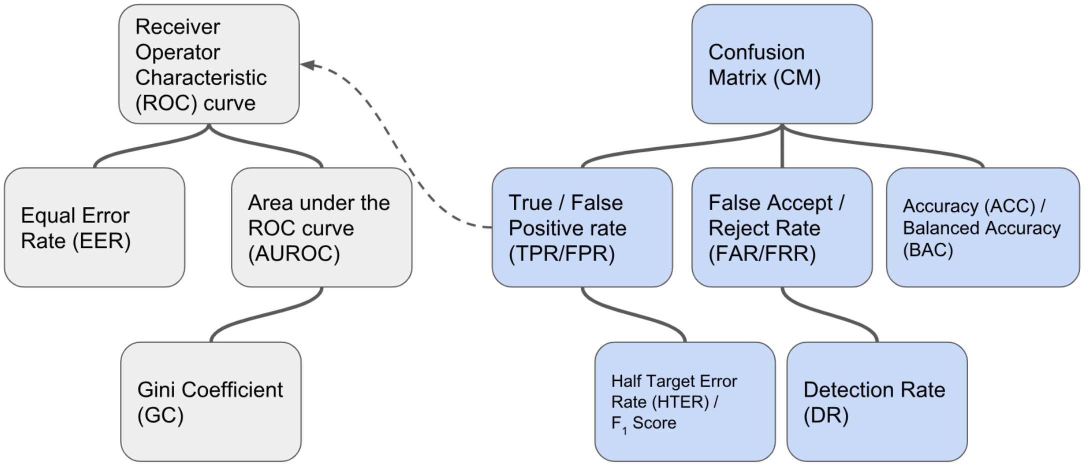

Receiver Operator Characteristic (ROC) curve
Equal Error Rate (EER)
Area under the ROC curve (AUROC)
Gini Coefficient (GC)
Confusion Matrix (CM)
True / False Positive rate (TPR/FPR)
False Accept / Reject Rate (FAR/FRR)
Accuracy (ACC) / Balanced Accuracy (BAC)
Half Target Error Rate (HTER) / F1 Score
Detection Rate (DR)

Figure 2.5: Many of the commonly reported metrics are derived from the CM or the ROC. The ROC represents multiple CMs under varying thresholds. The connection between the ROC and the CM is realized through the (TPR, FPR) pairs. Each point on the ROC is one (TPR, FPR) pair for a fixed value of the threshold.

many issues with the metrics that we consider.

### 2.5.2 ROC curve related metrics

Despite the overwhelming reliance on single threshold metrics generated by a confusion matrix, they have limited utility. Single threshold metrics present an incomplete picture of the system's performance. All of these metrics give no indication of how changes to the threshold affect the behavior of the metric. If the thresholds that were used to derive the metric are not reported, it is not possible to repeat the experiment to determine if the achieved metric values can be obtained in subsequent trials. To implement a system, some insight into the relationship between the performance metrics and the threshold is needed.

The ROC is computed by varying the authentication threshold from the maximum to the minimum possible values of the score and calculating the TPR and FPR for each threshold value. As the threshold lowers, scores that were not initially high enough to grant access will eventually rise above the threshold. Formally, we consider a binary hypothesis of the form:

$$\begin{aligned} X &\overset{h_0}{\sim} f_0 \\ X &\overset{h_1}{\sim} f_1 \end{aligned} \tag{2.1}$$

Where $X$ is the measurement score we are trying to classify and $f_i, i \in \{1, 0\}$ are hypothesized score distributions. With these hypothesis, we can compute the false positive rate (FPR) as

$$FPR(T) = \alpha(T) = \int_T^\infty f_0(x)dx$$

Where T is the threshold parameter. Similarly we can calculate the true positive rate (TPR) as

$$TPR(T) = 1 - \beta(T) = \int_T^\infty f_1(x)dx$$

The ROC curve is then the parametric graph of $(\alpha(T), 1 - \beta(T))$. Using the visualization tool at [58] we can visualize how a single point on the curve is computed. The tool assumes that we have assumed that the $f_i$ are normally distributed, however, in practice we would have to estimate these distributions over the scores computed from the collected measurements. In Figure 2.6, we have a generated ROC curve and a marked specific point corresponding to a single threshold value. In Figure 2.7, we show the two normal distributions assumed to generate the ROC curve, and the black threshold line, $T$ specified in the integrals.

An unauthorized user is erroneously granted access because the random variation caused their measurements to score above the threshold. However, if the random variation caused an authorized user's measurements to score lower than is typical, the user would still be granted access because the threshold is lower. Eventually, as the threshold is lowered, both the number of TPs and the number of FPs will increase. Each value of threshold represents a specific trade-off between the TPR and FPR. In Figure 2.2, we show an estimated ROC curve from a sample distribution. Also shown is the line of indifference, $y = x$ (green line of Figure 2.2). If the ROC is close to this line, the system performance is comparable to blind guessing.

There are three common single-number performance metrics for summarizing the ROC curve: the EER, the AUROC (or AUC in some texts), and the GC.

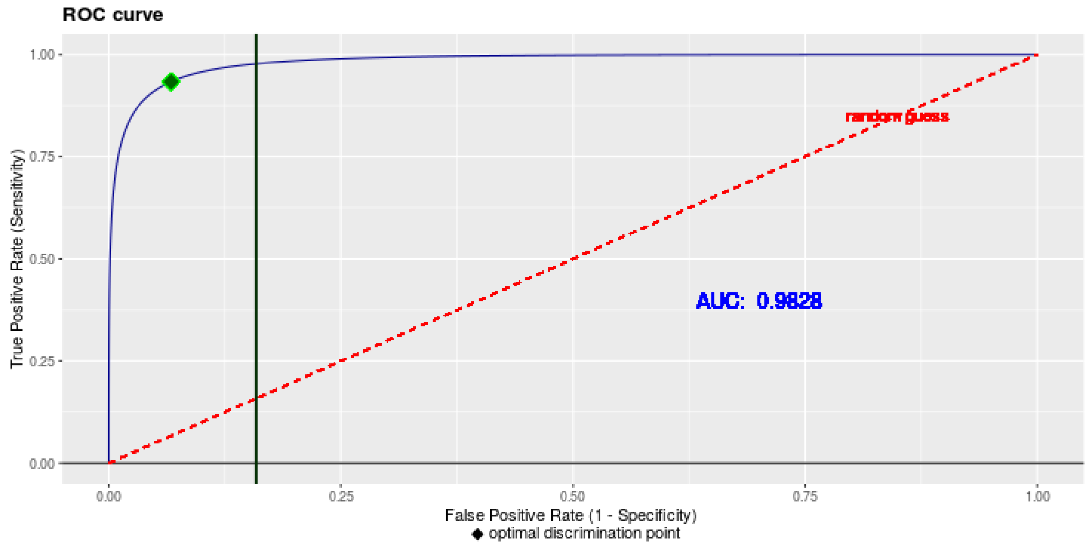


Figure 2.6: An example ROC curve with a specific point marked. This ROC curve assumes the hypothesized distributions are normal.

### Equal Error Rate (EER)

As Figure 2.2 shows, it is the point on the ROC where the $FPR = 1 - TPR$. It is easily identified as the intersection of the line $y = 1 - x$ (red dashed line of Figure 2.2) and the ROC curve. It represents the probability of making an incorrect positive or negative decision in equal probability. Since the ROC is a parametric curve, there is a specific value of the threshold that corresponds to the EER.

### Area Under the ROC Curve (AUROC)

The AUROC is defined as the area below the ROC curve and is depicted as the shaded region in Figure 2.2. It reflects the probability that a random unauthorized user's measurement is scored lower than a random authorized user's measurement. It can be interpreted as a measure of how well a classifier can separate measurements of an authorized user from their unauthorized counterparts. Hanely et al. [57] show that the AUROC is equivalent to a Wilcoxon test of rank, while this proof is involved, we will describe the intuition here. If we consider the computation of the single point shown in Figure 2.6, the $TPR$ integral (red area in Figure 2.7) can be viewed as the probability of observing a score above the threshold $T$. The value of this integral is the y coordinate

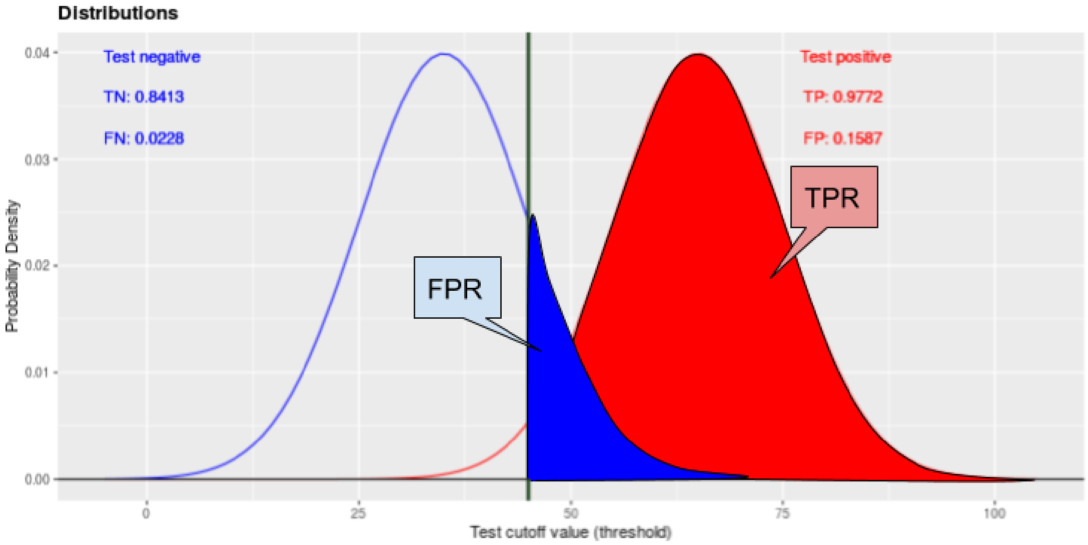


Figure 2.7: The black line in the center of this figure is the specific threshold value used to compute the $TPR$ and $FPR$ of the point marked in Figure 2.6.

which lies above the value of the $FPR$ integral (blue area in Figure 2.7). By integrating this function we use the $FPR$ as our differential element. This means we are summing the probability of observing a positive score above the threshold in units of probability that a negative score is above the threshold. As we move along the $FPR$ axis, the threshold sweeps through the range of possible values, integrating out dependence in the intermediate variable $T$. Thus we are left with the probability that a positive score lies above a negative score.

### Gini Coefficient (GC)

The Gini Coefficient (GC) is functionally related to the AUROC as follows [59]: GC = 2 × AUROC − 1. It also tries to quantify how much separation there will be in the measurements.

## 2.6 Proposed Metric: The Frequency Count of Scores (FCS)

We propose the Frequency Count of Scores (FCS) as an additional performance metric to be reported with the ROC curve. Figure 2.2 shows examples of the FCS coupled with the ROC. The FCS provides the ability to visually diagnose and explain the achieved performance reported in the ROC because the ROC can be constructed from the FCS. By examining the distribution skew and overlap of the score frequencies, we can determine if the proposed systems exhibit any biases towards a positive or negative decision. We can also justify the reported performance observed in the ROC by examining how well the score distributions are separated. Good score separation will yield good system performance which will be reflected in an ROC with a low EER. Sensitivity to changes in the threshold can be assessed by looking at how the scores are spread relative to each class. The two metrics complement each other.

The FCS is a fundamental metric that is different from the ROC and the confusion matrix. It is considered fundamental in this context because it is not derived from the CM or the ROC. It can be used to diagnose model problems, compare systems, and validate implementations. The FCS is constructed by identifying the maximum and minimum scores across all measurements and then choosing a common bin width over this range. Scores are separated by the ground truth and then plotted as separate histograms which are binned using the common bin width. The bin width is a free parameter that can be chosen to reflect the amount of data available and the observed score variability.

The FCS should *not* be normalized to make it look like a distribution. The unnormalized version makes the population skews, score distribution imbalances and score overlap regions visually apparent. The FCS is a useful addition to the reported metrics because it allows a research consumer to visually perform additional analyses which would not be possible with the ROC or CM metrics alone.

How the scores are distributed plays a central role in the performance of a system. A large majority of the decision errors are made because the random variation in measurements causes the scores to erroneously cross a chosen threshold. Because the measurements are not deterministic, the scores are variable even if they are a deterministic function of the measurements. How well the score function separates measurements in the presence of this variability dictates the range of possible error trade-offs between TPR and FPR for a system. If the separation of scores is large, then it is possible to achieve high TPR while keeping the FPR low. Since each choice of threshold represents a compromise between TPR and FPR, larger score separation implies better choices of threshold.

### The FCS can be used to gain performance insights beyond what the existing metrics show

Many of the existing metrics can actually be derived from the FCS. For example, the TPR can be computed as the relative frequency of positive scores that lie beyond the threshold.The proportion of the score range from authorized and unauthorized users that overlaps is important because many of the performance metrics, such as the EER and ACC, attempt to summarize system performance by quantifying how often a measurement from either user will get a score in this overlapping range. The ACC and EER both depend on the width of the overlapping region as well as the relative frequency of the scores that fall within this overlap. Neither metric considers the portion of the scores that lie outside the overlapping score region for either score distribution (authorized and unauthorized). It is this lack of consideration for these other aspects of the scoring that cause these metrics to be incomplete. By reporting the FCS, difficult concepts can be easily visualized. Consider the AUROC: its definition is very technical, and is thus difficult to interpret. However, if we look at two different FCS and note the score overlaps are smaller in one vs. the other, we have captured the essence of what the AUROC is trying to measure.

Some insights into system performance that only the FCS can provide are gained by considering scores that lie outside the overlapping region. Performance metrics, such as the TPR, are directly impacted by the portion of these types of scores. For example, authorized users' scores that lie outside the overlap can only contribute to the TPR. If this portion is not empty, then the TPR may never practically reach zero (e.g. there is a threshold for (A) of Figure 2.9 that achieves non-zero TPR at zero FPR because scores above this threshold could not have come from unauthorized users). Although this can be visually confirmed on the ROC, it would be difficult to identify why it happens from the ROC.

The differing slopes of the ROCs in Figure 2.9 do not indicate how sensitive the classifier is to changes in the threshold. A research consumer cannot ascertain how far along the ROC a change in the threshold will move them purely by looking at the ROC. However, this information can be gathered by looking at the spread of the score distributions in the FCS. If the scores are spread wide relative to the width of the overlapping region, then the classifier is not particularly sensitive to the threshold. If the width of the overlapping region is small compared to the score distributions, small changes in the threshold will cause significant movement along the ROC.

If scores from authorized users that are above the overlap occur with higher relative frequency than scores within the overlap, then the authentication system will produce more positive declarations. A similar result holds for the TNR and unauthorized users' scores which are below the overlap. When examining the FCS of a proposed system, if only one user has scores that lie outside the overlap (e.g. FCS (F) in Figure 2.10), the system may be biased towards decisions in favor of that type of user (e.g denying access since most of the scores range comes from unauthorized users). Without the FCS, it is difficult to determine if a proposed system has this kind of flaw, even if the ROC is reported.

## 2.7 Flaws with Existing Metrics

In this section, we discuss in detail the implications of the observed flaws we summarized in Section 2.2. We note the cases where the FCS aids in diagnosing whether a flaw is present or explains why the flaws occur.

### 2.7.1 Incomplete performance reporting

Skews within the measurement population can artificially inflate some CM derived metrics. Measurements are often split into training and testing data. Training data is used to build a model and testing data is used to compute performance metrics. If the measurement population is skewed, both data sets will exhibit this skew. If this approach is coupled with a report that uses only a single performance metric, misconceptions arise. For example, in one of the papers we reviewed, the authors only report the FNR. Unfortunately, their measurement population was skewed. From their reporting, we cannot know whether the low FNR is due to their system's ability to discern users or due to a skew in the measurement population.

If we only have a single metric available, such as the ACC for two systems we are trying to compare, the system with the better value can be deemed superior. On the surface this seems like a perfectly fine criteria. For example, given that the interpretation of the ACC as an approximation of the probability of a correct classification, a higher ACC would seem to indicate superior performance. Unfortunately, relying on the ACC as the sole criteria can be very misleading. It is possible that the ACC value was inflated by skew in the measurement population.

When a classifier has poor ACC, the scores from authorized and unauthorized measurements will have significant overlap (as seen in the left side of Figure 2.8). These overlapping scores are ambiguous and thus difficult to classify. If we skew the measurement population to have mostly unauthorized users, the scores from the unauthorized users overshadow the scores from the authorized users' measurements (right side of Figure 2.8). In this instance, possible values of the threshold that cause the classifier to make mostly negative decisions (denials of access) will be favored because most classifiers are optimized by minimizing the error over the data on which they are trained [6]. Since the test data is skewed in the same way, a classifier that returns mostly negative decisions will be correct most of the time. This skew in the test data will make the classifier appear to be more accurate because it is being tested on data it would always get correct.

While the detrimental effects of skew are evident for the ACC, any metric that depends on both $N$ and $P$ at the same time (cross column in the confusion matrix of Table 2.4) will be affected by population skew [53]. For reference, $N = TN + FN$ and $P = TP + FP$. Figure 2.8 also demonstrates how the ROC is mostly unaffected by population skew. Population skew can mask the poor score separation by providing performance numbers that are artificially high. However, these flaws are easily identified by the frequency count of scores. The unnormalized counts in FCS show that the total volume of scores from unauthorized users vastly outnumber those of authorized users. This visualization helps designers easily examine the skewed measurement population.

### 2.7.2 Results without model parameters

In the ideal case, the code used to derive the results of a study would be published along with the proposed system. This may not be feasible in all situations. However, most of the systems that use machine learning do not implement the algorithms from scratch. Instead, they often apply to existing implementations in libraries such ask Weka [60] or libSVM [61]. The novelty of these proposed systems often lies in the complete end-to-end performance, not the algorithm used to make decisions. Implementations of the proposed systems can be simplified by having the model parameters (e.g. SVM slack factor, number of hidden layers in the NN, and learning rate) that were used to derive the reported results. These parameters control the behavior of the algorithm and reflect a value judgment made by the researcher based on their understanding of the how the algorithm interacts with the measurements.

By providing the parameters of the algorithms used, authors enable replication of the research, benefiting the community in two ways. First, the data analysis can be replicated exactly to determine if other factors contributed to the reported results. If the data is also available, follow-up analyses can be more easily performed. Updated versions of the libraries that may have fixes for vulnerabilities or performance enhancements can be validated against existing results. Second, any potential implementers of the system only need to replicate the measurement collection and data processing portions of the proposed systems. The properly parameterized software library can essentially be treated as a black box.

### 2.7.3 Misleading performance metric interpretations

A key issue with reporting only a single performance metric as a summary of the system is that the value of the metric does not uniquely identify the classifier from which it came. This information is lost, and with it all knowledge of how the system performs when the parameters are adjusted. In Figure 2.9, we can directly observe this issue in the EER and AUROC performance metrics. The graph shows several ROC curves from different score functions that all have very similar EERs and AUROCs. Each of the ROC's linear portions have different slopes. These differing slopes reflect different sensitivities to the threshold, due to the difference in how the scores are distributed. A change in the threshold has much more impact on the ROC (F) than on the ROC (A) curve, thus an implementation can fail in unexpected ways because the implementers were unaware of this difference.

If we are only given the EER to evaluate a system and we have a specific target for our TPR or FPR, we are unable to determine from the EER if our target will be met. This information is not knowable because many ROCs (and thus many classifiers) have the same EER. Because we do not know which classifiers were used, there are many possibilities for how the performance can vary with the threshold. As we note in the Introduction, many applications have specific requirements for the TPR or FPR, which are attained by controlling the threshold.

By definition, all ROCs must connect the point $(0, 0)$ to the point $(1, 1)$ (i.e. setting the threshold of a classifier higher than the maximum achieved score will result in 0 TPs and 0 FPs, whereas setting it below the minimum will have the opposite effect). The EER fixes a third point that the curve must pass through. However, as depicted in Figure 2.9, these three points do not uniquely determine the curve. There are many ROCs that correspond to a small range of EERs due to what the EER is measuring.

The EER is the point on the ROC in which the probability of an incorrect denial of access is equal to the probability of an incorrect granting of access. Both of these probabilities are proportional to the width of the region of overlap in the scores. In Figure 2.10, the corresponding FCS for each ROC in Figure 2.9 is depicted. Each score distribution has the same width of overlap region, however, the overlapping region moves to the right as the figures are read left to right, top to bottom. As the overlapping region moves right, it consumes more of the score range for the authorized users' measurements.

As the authorized users' score range shrinks, the unauthorized users' score range grows to maintain the width of the overlap. Because all of the overlapping regions are the same width in all cases, a threshold can be identified for each score set that strikes the same balance between the two error types (FP and FN). This threshold will be in a different place for each of the different score distributions. However, each classifier can be tuned to achieve the same EER by picking the correct threshold, even though their individual tolerances to threshold shifts vary greatly. Since there are fewer authorized users' scores, the sensitivity of the classifier to changes in the threshold goes up because each change has a greater impact on the classifications that are made causing the distinct differences in slope in Figure 2.9.

A key shortcoming of the EER is that it focuses only on the overlap between the score ranges. It fails to consider the proportion of the score range that lies outside the overlapping region for either measurement source or any asymmetries in the distribution. Only scores that are within the overlap contribute to classification errors because they can be confused with scores from the alternate class. The proportion of scores outside the overlap is as important as the width of the overlap itself because it governs the probability that an easily confused score will be observed. The EER fails to account for asymmetries in the score distribution. The graph of FCS depicted in Figure 2.10 makes the proportion and the asymmetry visually apparent.

There is no measurement population skew in any of the graphs in Figure 2.10; instead, the authorized range is shrinking. The unnormalized frequency count shows that the probability mass across authorized measurement scores is being redistributed over a smaller range. Thus, the score distributions are becoming more asymmetric. The classifier is becoming more biased because the probability of observing a score that could only have come from an authorized user's measurements is getting smaller. Similar to the skew accuracy problem of Section 2.7.1, observing the weight and range of the scores from both measurement sources allows for the identification of problems, with the authentication system making the overlapping region width and distribution asymmetry apparent.

### AUROC and GC are also flawed

The AUROC is interpreted as the probability that scores of differing measurement sources separate well [57]. This probability is proportional to the width of the overlap in score range. As the width of the overlap gets smaller, the probability increases. Thus bigger values of the AUROC are desirable. If the AUROC → 1, then all authorized users' measurements will score higher than unauthorized users' measurements. The probability of separation is maximum, thus there may be a threshold that achieves perfect classification. Since the Gini Coefficient (GC) is functionally related to the AUROC, it is also functionally tied to the width of the overlap in score range.

Unfortunately, the AUROC and GC also exhibit interpretation flaws because they are summary metrics. They also mask the complexity of the classifier performance. In Figure 2.9, the AUROC was computed for each of the curves (the GC can be computed from the formula). As expected, the range of the AUROC does not vary significantly even though the resulting ROCs are very different. The AUROC is within the range $(0.84 - 0.88)$ across the different classifiers. The AUROC does not vary due to the width of the overlapping region which is held constant, as seen in Figure 2.10. Like the EER, the AUROC focuses heavily on the overlap of scores.

## 2.8 Recommendations for reporting: Solutions to the Pitfalls of Current Reported Metrics

In the ideal case, authors would make all source material available, including data and code. This approach would yield the best results for system evaluation because evaluators and implementers could verify their implementations against the reference provided by the researcher.

Although there is no one-size-fits-all strategy for analysis, we propose guidelines that can be followed to simplify the task of evaluating the proposed system. The following three suggestions may aid consumers of research, including an implementer who needs to choose the best authentication system for their target application.

*First suggestion: Report as many metrics as possible including both the ROC and the FCS.* These two graphs enable comparisons across many parameterizations and serve as a visual check for biases. The FCS enables both researcher and reader to diagnose issues with population skew and score distribution via immediate visual analysis. It can also serve as a diagnostic tool for implementations to verify that the scores produced by the implemented system are within the range the researcher originally reported. Other metrics such as the EER, AUC and ACC should also be reported for comparison with existing literature. These metrics can be added to abstracts and introductions for glanceability but are not a substitute for a complete analysis that includes an ROC and FCS.

*Second suggestion: If the FCS cannot be reported, report the ROC curve to enable an implementer to decide if the system has a threshold that meets the error performance requirements of their target application.* Implementers can find specific points on the ROC curve that satisfy their requirements and be assured that if the implementation is faithful to the proposed system, the can find a value of the threshold that yields the chosen error rates.

*Third suggestion: If the ROC cannot be reported (e.g. for space constraints), report multiple summary metrics that are not functionally dependent.* Since each of the summary metrics, EER, ACC, and AUROC only represent a single aspect of the system's performance, the reader can obtain a more thorough evaluation of the performance if all three are reported. Reporting all three gives readers the ability to compare the proposed systems from the existing body of literature that often only report one or two of these metrics.

### 2.8.1 Case Study

To demonstrate how to use the ROC and FCS to compare systems, we evaluated the authentication performance of three existing datasets via the ROC and FCS. We will first describe the datasets and classifiers that were built. Each publication provides a dataset and a system model to test their dataset. When the classifier is used on the dataset, an FCS and ROC will be computed. Because each publication's dataset and classifier has its own population distribution and score function, we expect the FCS and ROC from each publication's proposed system to be very different. We will show how to use the ROC and FCS to decide between these systems.

### Datasets and classifiers used to create ROCs and FCSs

The SVC2004 dataset [62] is a public signature dataset with 40 types of signatures and 20 genuine samples for each signature. We implemented the linear classifier of Principal Component Analysis, proposed by Kholmatov [48]. The Touchalytics [47] is a public touching behavioral biometric dataset with 41 participants' continuous touching behavior data. We built the authentication system with $k$-nearest-neighbors as described in their paper. We selected $k$=100 and each user contributed 150 periods of touching behavior as templates. The dynamic keystroke dataset [63] is a set of keystroke features that was collected while users input a password. Typing behavior was observed for 51 users, and each user contributed 400 typing samples. The proposed system was built with a one-vs.-all-classifier for each user. In our study, we randomly chose a user and built the authentication system with Manhattan (scaled) similarity that was described in the paper.

### Analysis of these three systems with the FCS and ROC

In Figure 2.11, we display the ROC curves for all three systems. We assumed that the implementer has a fixed requirement on the FPR of 0.1. To choose a system that meets our requirements, we drew a solid black vertical line at our FPR limit. Thus, we can visually identify the system that has the highest TPR for our FPR limit. In this case, Keystroke is the clear winner, even though it does not have the lowest EER. Thus, potential implementers would not able to assess a proposed system when given only the EER.

We also see the slope of the ROC near the fixed FPR target. Observe how quickly the TPR degrades if we need to make the FPR tolerance lower. For example, the segment of the ROC from SVC2004 classifiers around the FPR target is very steep, indicating that a small change in threshold will lead to a significant change of the system's TPR and FPR rates. In contrast, the ROC of the Keystroke dataset is very stable because the TPR changes slightly with the change of FPR. If we have an upper bound on the FPR and want a system that gracefully degrades when the FPR target is lowered, the

Keystroke classifier is the clear winner. It should be noted that if we could tolerate an FPR of 0.3 or higher, the Keystroke system would be inferior to both SVC2004 and Touchalytics when considering both the the TPR value for a fixed FPR and the slope around a fixed FPR.

The FCS can be used to show the asymmetry of the score distributions in detail. Figure 2.12 displays the ROC curves of the three systems along with their corresponding FCS. For example, the ROCs of SVC2004 and Touchalytics are similar, but their FCS shows that the SVC2004 classifier has an advantage because the authorized and unauthorized scores are more separable than the Touchalytics classifier. Additionally, the EERs of SVC2004 and Keystroke classifier are similar, but their FCS shows that the SVC2004 classifier is superior because the unauthorized and authorized users' scores overlap significantly in the Keystroke dataset.

From the FCS of Figure 2.12, we can observe the asymmetry in the scores. The unauthorized users' scores in the Touchalytics classifiers almost cover the entire score range, indicating that the classifiers can never be certain about granting access. Every authorized user's score could have come from an unauthorized user, thus this system may be biased to deny access more often. The Keystroke classifiers are biased in the other direction: all unauthorized users' scores could have come from an authorized user. The SVC2004 classifiers have some score range which does not overlap, and thus can make some decisions with certainty. If our application needs to be balanced, SVC2004 is our best choice. If the application needs to be biased towards denials, then we should choose Touchalytics. If we want more positive decisions, Keystroke has a higher probability of delivering them.

We have showed how relying on the EER can produce erroneous conclusions about authentication systems. We have provided evidence for the limitations of reporting the EER in three systems: Keystroke, SVC2004, and Touchalytics. We have also showed how the ROC and FCS should be implemented as a solution to the limitations of single number summaries. In Figure 2.12, we show how the FCS compliments ROC to improve reporting authentication system metrics.

## 2.9 Discussion and Conclusions

We have proposed robust metrics for evaluating machine learning-based authentication systems: ROC curves and their corresponding FCS. We argue for the ROC as a method of reporting classification performance because it is able to provide an overview of the authentication performance across all thresholds. However, the ROC misses some scoring details, such as the difference in the width of score ranges and the asymmetry of score distributions. This scoring detail can indicate whether a classification bias is present in the scoring function and how sensitive the error rates are to changes in the threshold. Therefore, we introduce the FCS as an augmentation to the ROC curve. We believe reporting the ROC and FCS together provides a robust metric for evaluating the performance of authentication systems.

The commonly used authentication performance metrics, such as EER, AUROC, GC and ACC, are inherently flawed. EER only focuses on the overlap between the score ranges and does not consider the proportion of the score range that lies outside of the overlapping region. Since the scores inside the overlapping region are the reason errors are made in the authentication system, there always needs to be balance between the two types of errors.

The two types of error rates of the system depend heavily on the thresholds in the overlapping region of scores. If the proportion of scores inside the overlap is large, one will likely encounter a score that is difficult to classify. We show in Figure 2.10 that with a similar EER, system (A) is much better than system (F) because system (A) is able to completely separate some of the measurements from different user types. Therefore, EER, AUROC, and ACC hide important information that could be used for comparison between authentication systems. Even the ROC by itself provides a limited analysis. While the differences between (A) and (F) visually manifest in the ROC as a higher TPR at 0.0 FPR and a different slope, it is not visually obvious why this happens or how the TPR changes as the threshold changes. Reporting practices that focus on a single metric limit the ability to compare systems by ignoring these factors.

We introduce the FCS to augment the ROC in order to evaluate and compare the performance of authentication systems. The FCS is fundamentally different from the ROC curve and the CM because it is not derived from either and thus brings additional information into the analysis. We can use FCS to detect the measurement population skew, asymmetries in the scoring distribution and assess sensitivity to threshold changes. Since the scores in FCS are not normalized, the population skews are visually apparent. Usage of the FCS is not limited to authentication systems. The ability to identify distribution imbalance and threshold sensitivity is relevant to any applications that use machine learning to decide where their measurements come from.

We have illuminated the problems with current reporting practices in authentication system research. Reporting these single-number summaries alone is a barrier to comparison between systems and can misrepresent a system's potential. For example, some metrics do not show the performance trade-offs or whether performance degrades outside the conditions for which the system was designed. We proposed a solution to the limitations of current metrics: reporting a full set of metrics that includes the FCS and the ROC. We argue that performance reporting should be as comprehensive as possible and that the the FCS and ROC can help in this regard by provides additional information to evaluate authentication systems. We believe it is crucial for our community to adopt more transparent reporting of metrics and performance.

| | Definition | Description |
|---|---|---|
| TP | True Positive | Authorized legitimate users count. |
| FP | False Positive | Authorized illegitimate users count. |
| TN | True Negative | Denied illegitimate users count. |
| FN | False Negative | Denied legitimate users count. |
| CM | Confusion Matrix | Table of contingency counts. |
| ACC | Maximum Accuracy | Probability of a correct declaration. |
| TPR | True Positive Rate | How often a legitimate users is authorized. |
| TNR | True Negative Rate | How often an illegitimate user is denied. |
| FPR (FAR) | False Positive Rate (False Accept Rate) | How often an illegitimate user is authorized. |
| FNR (FRR) | False Negative Rate (False Reject Rate) | How often a legitimate user is denied. |
| ROC | Receiver operator characteristic curve | Curve of (TPR, FPR) by varying threshold |
| ERR | Equal Error Rate | The point that TPR equals 1-FPR |
| AUROC | Area under the ROC curve | Probability of scores of random legitimate users are higher than illegitimate user. |
| GC | Gini-coefficient | Calculated from the AUROC |
| FCS | Unnormalized frequency count of scores | Histogram of scores separated by ground truth |

Table 2.1: Performance metric name abbreviations

| Venue | References |
|---|---|
| CCS | [12, 13, 14] |
| CHI | [15, 16, 17, 18] |
| IMWUT/UbiComp | [19, 20, 21, 22, 23] |
| INFOCOM | [24, 25, 26, 27, 28] |
| MobiCom | [29, 30] |
| MobiSys | [31, 32] |
| NDSS | [33, 34, 35, 36] |
| Pattern Recognition | [37, 38, 39, 40, 41, 42, 43] |
| SS&P (Oakland) | [44] |
| SOUPS | [45, 46] |

Table 2.2: Publications surveyed grouped by venues

| EER % | ACC % | FPR % |
|---|---|---|
| 0.00 [38] | 99.30 [14] | 0.00 [19] |
| 0.34 [40] | 98.61 [30] | 0.01 [38] |
| 0.59 [37] | 98.47 [27] | 0.10 [22] |
| 0.95 [42] | 98.00 [28] | 0.10 [33] |
| 1.26 [14] | 97.00 [18] | 0.10 [42] |

Table 2.3: The top five authentication systems according to a naïve comparison of their best reported values for EER, ACC, and FPR metrics. These metrics are reported the most often but rarely yield a meaningful comparison. There is no clear winner as each of the top five performers in each category varies significantly.

|  | Measurement Source |  |
|---|---|---|
|  | Authorized<br>(Positive) | Unauthorized<br>(Negative) |
| Grant Access<br>(Positive) | TP | FP |
| Deny Access<br>(Negative) | FN | TN |

$$TPR = \frac{TP}{TP+FN} \quad FPR = \frac{FP}{FP+TN}$$

$$ACC = \frac{TP+TN}{TP+TN+FN+FP}$$

Table 2.4: For a single value of the threshold, the confusion matrix (CM) arranges the counts of all possible contingencies in a grid.

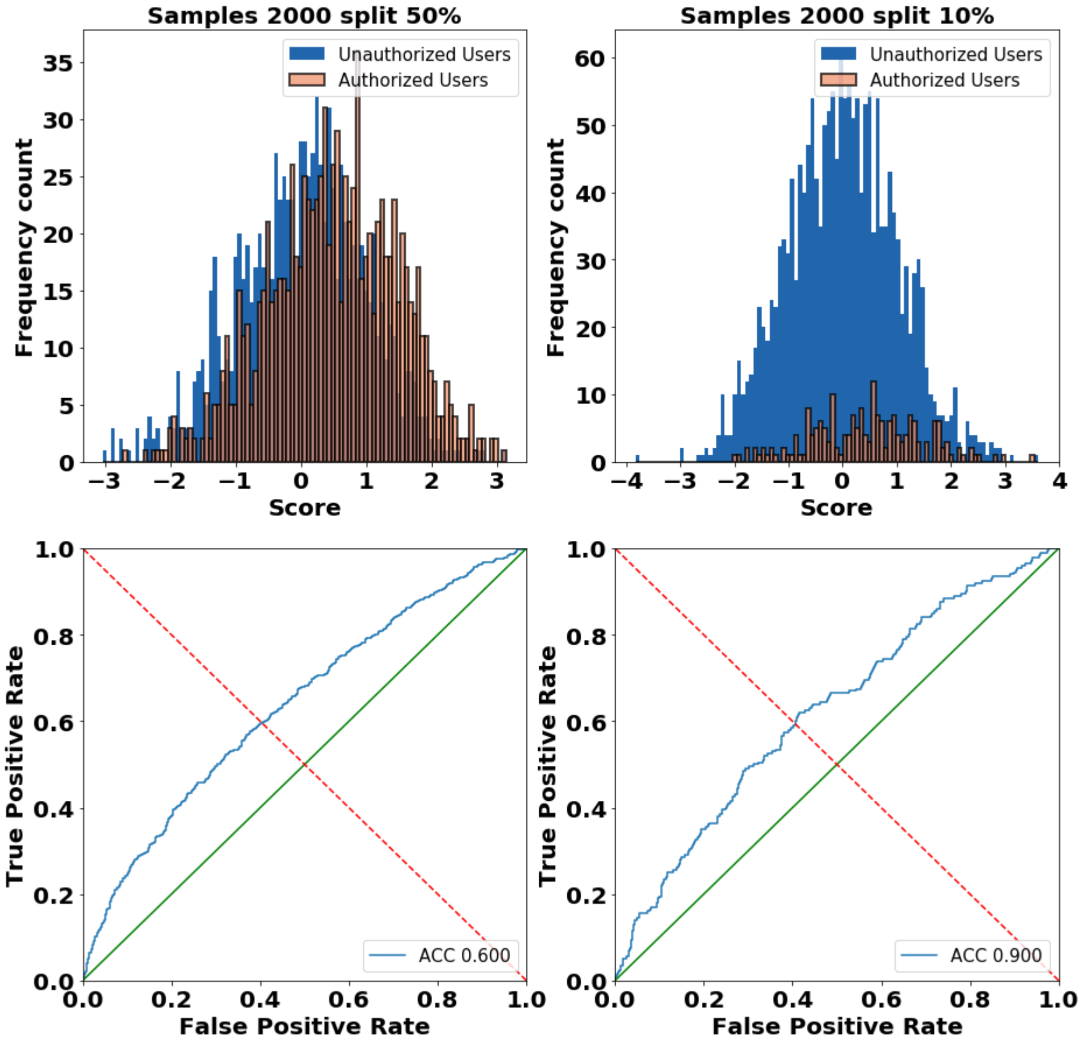


Figure 2.8: Measurement population skew can cause low accuracy classifiers to have artificially high accuracy values. On the left side, the set of 2000 measurements is evenly split between authorized and unauthorized. We drew the FCS and computed maximum accuracy $ACC \approx 60\%$ and the ROC under this measurement split. On the right side, the set of 2000 measurements is skewed to include 10% authorized and 90% unauthorized measurements. Because of this skew, the FCS shows that the positive scores are effectively buried in the negative scores. The maximum accuracy achieved is $ACC \approx 90\%$ which is reached by choosing a threshold that results in mostly negative declarations. This reported accuracy is misleading because the scoring function was the same.

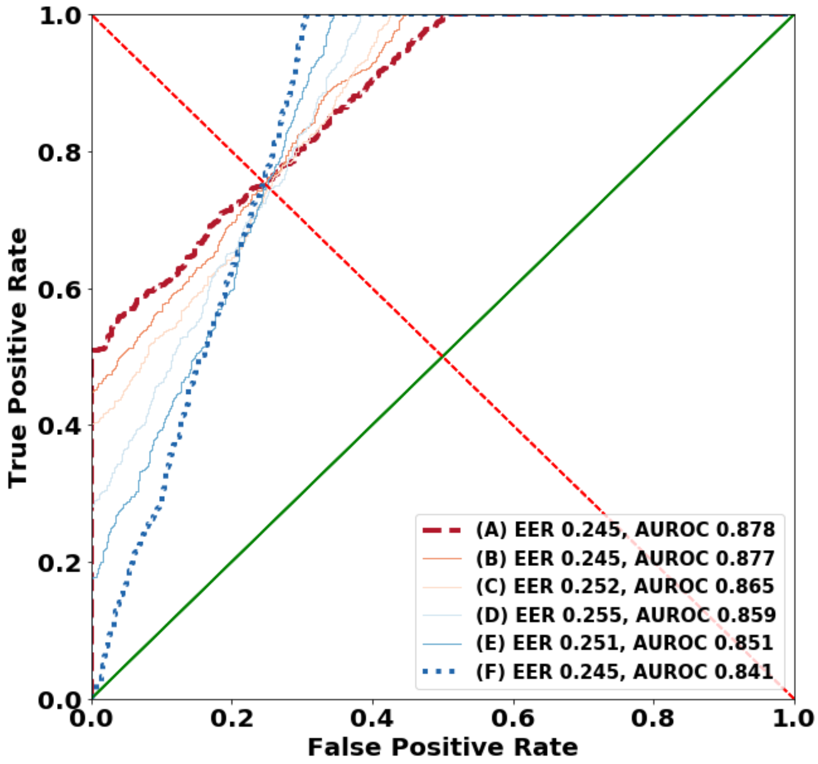


Figure 2.9: The EER does not represent a single ROC curve. Instead, it represents a family of ROC curves. While each member of this family have similar EER, their performance varies significantly across the range of possible thresholds. Unexpected sensitivity to changes in the threshold can lead to surprises in system behavior when the thresholds used in the implementation deviate from the published values. In this case, ROC (F) is mostly inferior to ROC (A) because (A) achieves a higher TPR at 0.0 FPR. However, according to the EER, they are essentially the same. If an implementer has a specific TPR or FPR target, the EER may be of little value to them as they cannot determine how the TPR/FPR may vary between the EER and their target. It should be noted that if the target application can tolerate an FPR > EER then (F) is the superior choice, however this tolerance cannot be known to the researcher. There is no skew in these examples.

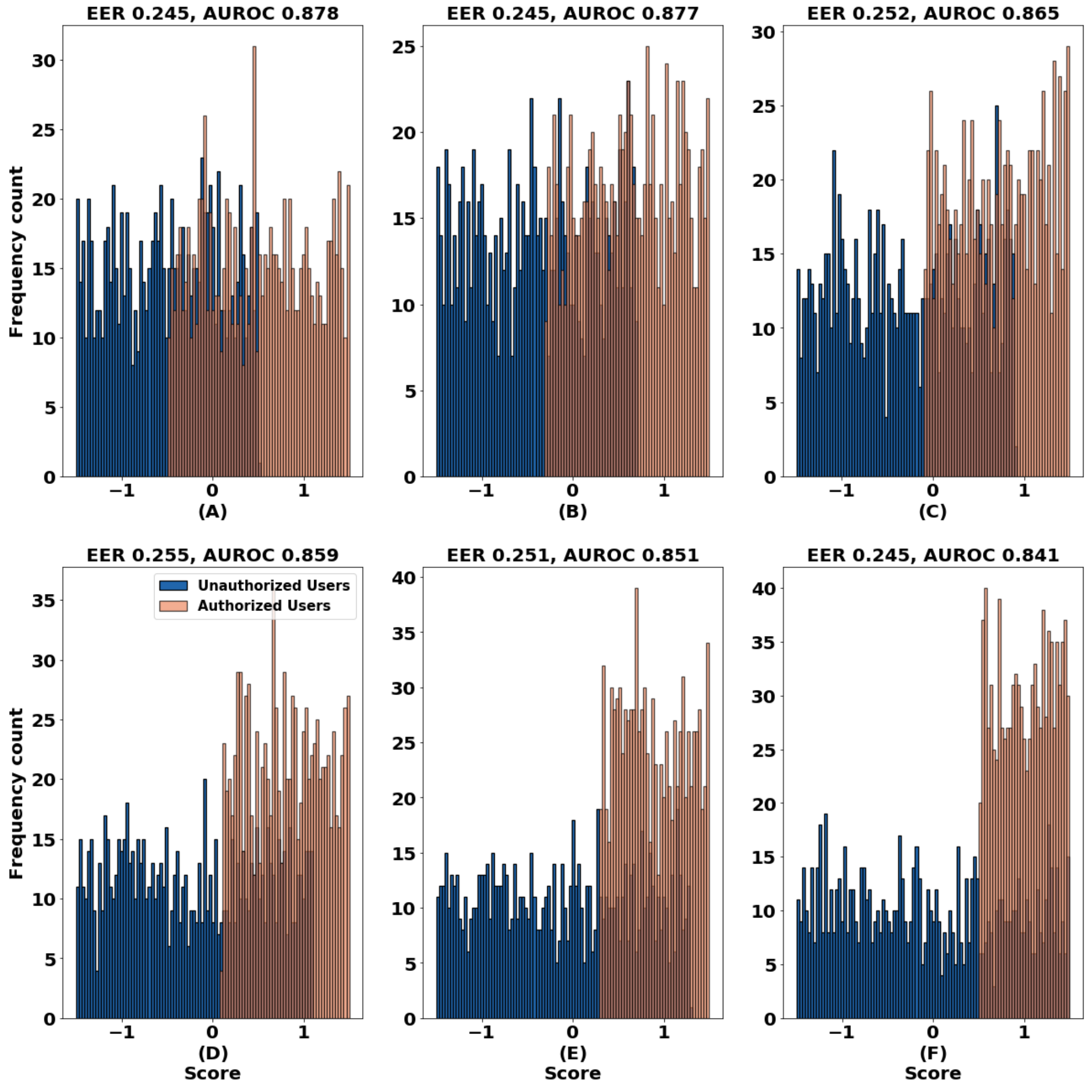


Figure 2.10: These FCS graphs show where the shapes in Figure 2.9 come from. They were constructed so that as we move from left to right and top to bottom, the region of scores that an authorized source has to definitively identify as authorized is shrinking, however the width of the overlap stays the same. This explains why (A) has non-zero TPR at FPR(0.0) but (F) does not. As the authorized score region is consumed by the overlap in scores, there are fewer distinct scores from the authorized user and thus fewer ways to get a purely true declaration. This difference explains why the range of possible trade-offs is worse for (F) than (A), as reflected in the slope of the ROC. The ROCs are linear because these score distributions are uniform (purely for example purposes). If the distribution shape changes but the overlap region remains the same, the EER behavior will be unchanged, however the ROC will be more curved. There is no skew in these examples, only the score distributions change.

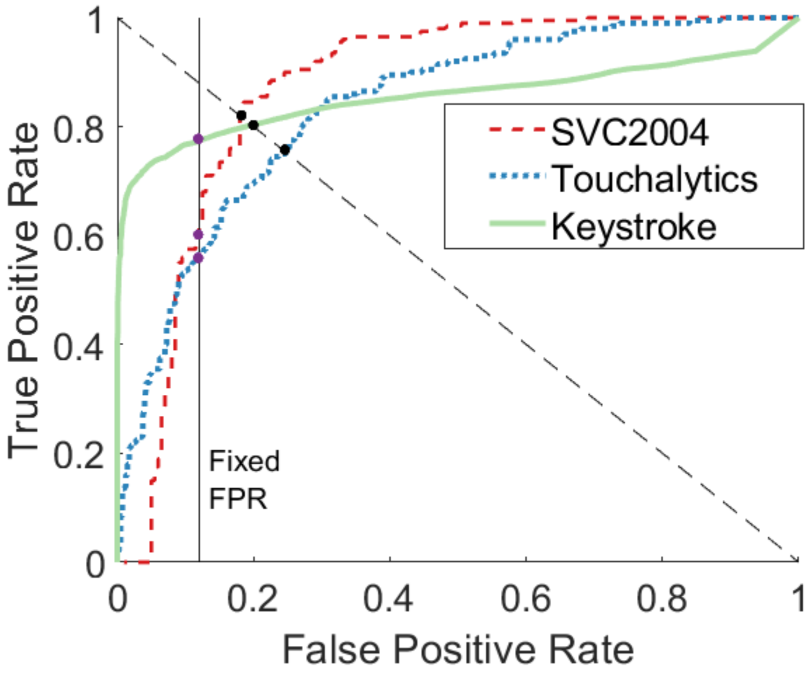
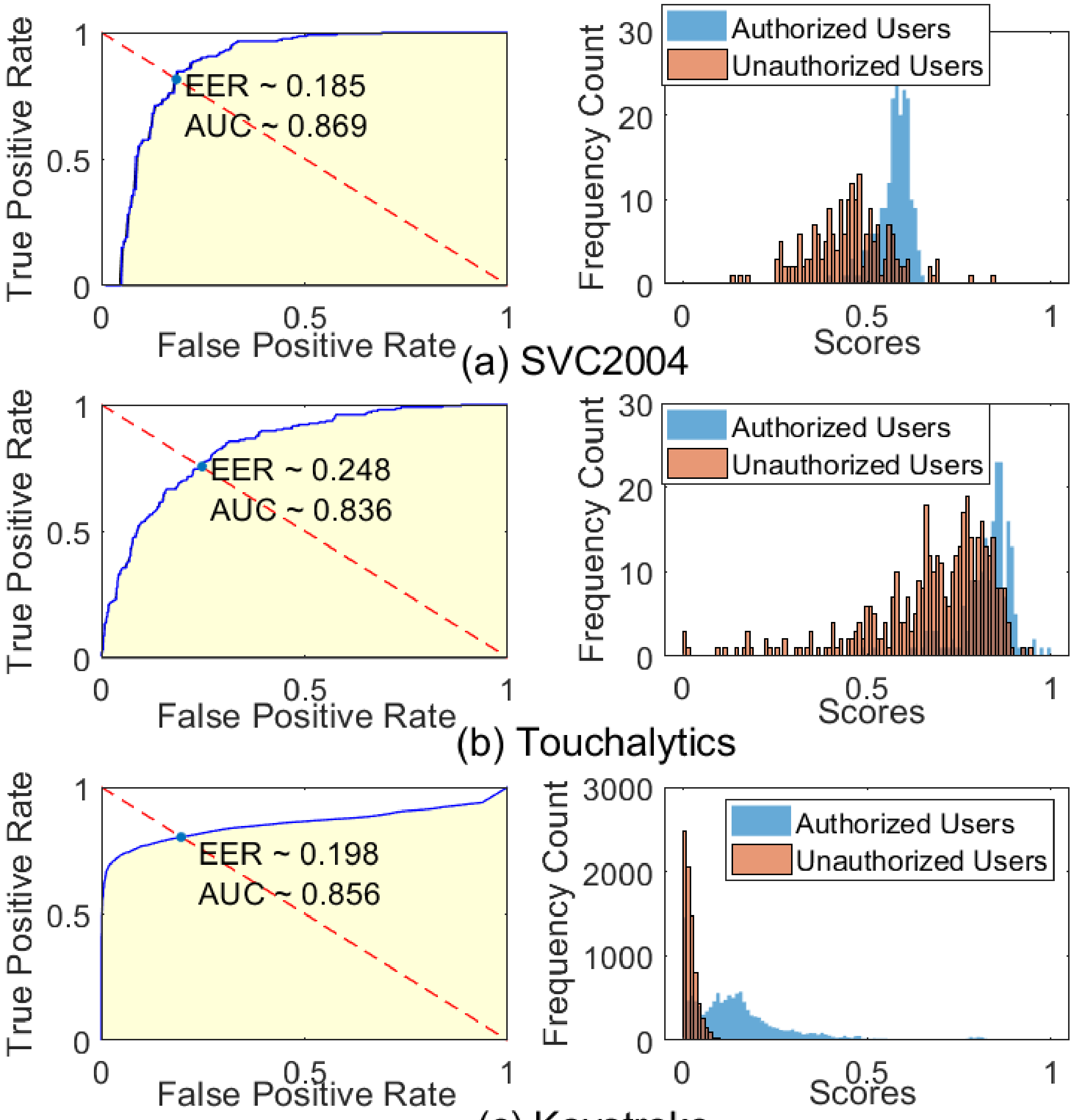


Figure 2.11: Comparing systems with a specific FPR target in mind is done by finding the highest TPR for that FPR. To find the highest, TPR draw a vertical line at that FPR and then identify the ROC that crosses the line at the highest point. A similar procedure works for specific TPR targets with horizontal lines.



(a) SVC2004

(b) Touchalytics

(c) Keystroke

Figure 2.12: The FCS can be used to decide between different systems that have similar EERs and ROCs.

# Chapter 3

# Exploring Performance Limits For User Identification Systems

## 3.1 Chapter Overview

Identifying individuals is a key component in many systems like automated grocery (e.g. Amazon Go [64], Alibaba Taocafe, DeepBlue Takego), personalized recommendation systems [65] (e.g. ads [66, 67], movies [68], products [69], music [70]), or multi-user interfaces [71]. The typical identification system measures observable features of a user and then feeds these measurements to a decision mechanism. The decision mechanism learns the distributions of measurements from the dataset and makes predictions by partitioning the space of measurement values.

Many recently proposed identification systems use machine learning classification algorithms as their decision mechanism [72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97]. An identification system's performance is measured by how well it performs classification. These systems are often validated with a user study where participants are recruited, observables are measured multiple times, and then the measurements are used as a dataset for a classification algorithm.

For a system to be robust, it is critical to know what conditions cause it to fail. The user studies used to evaluate these proposed systems should provide some insights about the limits of the proposed systems. Even the most ideal systems which are capable of identifying large numbers of participants with minimal error will fail in unexpected ways when used beyond its upper bounds (an upper limit on number that can be identified with minimal error) as described in Figure 3.1. Often these bounds are not included in the analysis of a proposed system nor are they reported in the publications if known.

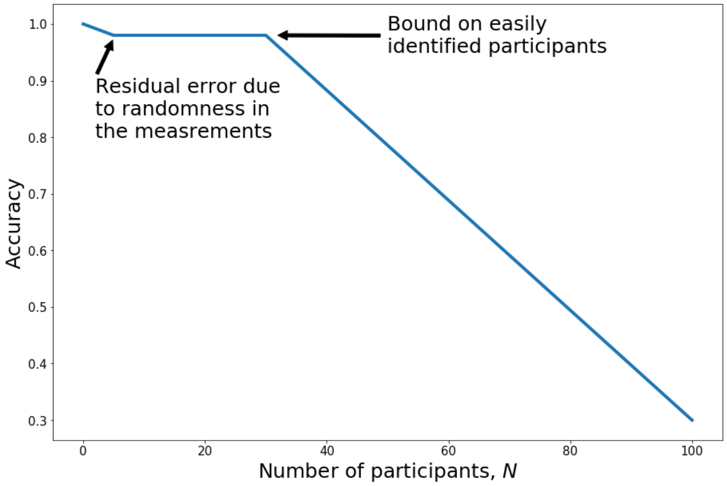


Figure 3.1: Even an ideal system that is capable of identifying many users with minimal error will eventually reach an upper bound, $\mathcal{N}$, beyond which the performance will decrease with out recovery. Because the measurement value range is finite, the ability to distinguish participants based on these measurements begins to degrade as more participants are added. Consequently, users that were easily identified will be confused with new users that have similar measurement values. We can choose a threshold (e.g. accuracy greater than 80%) below which we declare that the systems error rate makes it unsuitable. Beyond this point, we declare that the system has failed. Because the measurements can vary randomly, it is often very difficult to identify the bound, $\mathcal{N}$, beyond which the performance decreases monotonically.

User studies reported in the literature are often inadequate to consider a system well tested. Although researchers may collect large amounts of measurement data per participant, the participant count is often low. Collecting a large number of measurements from a small group of participants does not test the limits of the decision mechanism. To verify that small participant counts is an issue for identification systems, we surveyed 30 recently proposed identification systems and noted that the median number of participants for their user studies was 12. Of the systems surveyed no system reported a limit on the number of users that can be handled by the system or identified the conditions which drive the system to failure. Several of these systems use classification algorithms such as support vector machines, neural networks or random forests as multiclass classifiers. Their performance is often measured using two common metrics, accuracy and the confusion matrix [98]. Accuracy is the relative frequency of a correct classification. The confusion matrix is a contingency table that enumerates how often any one class gets confused for any other class. Together, both metrics quantify different aspects of how often the decision mechanism fails.

In this paper, we aim to make a strong generalizable claim: **no one should be surprised that classification algorithms are able to distinguish classes derived from small numbers of participants.** We demonstrate this by building five different identification systems based on five different publicly available datasets that measured humans. We tasked these systems with identifying the humans who were measured. We chose human-generated data to ensure the probability distributions of the data would be similar those encountered when conducting a user study to evaluate a proposed system. The datasets were used only as a source of human-generated measurements: we are not concerned with the datasets' original purposes. We purposely minimized the effort to make the classifiers perform better than guessing. We examine the reasons why this is easy to achieve. We consider the impact of *measurement count* (the total number of measurements taken across all participants), *feature dimension* (a function of number of distinct measured observables), *sample diversity* (how distinct the measurements of each participant are from each other), and *participant count* (number of participants in the user study). We use the insights from our dataset analysis to reason about the degradation of performance as the participant count increases. The major contributions are as follows:

**We examined the recognition performance of identification systems with low participant counts.** We mimicked the evaluation procedure of an identification system by constructing five distinct systems based on publicly available datasets, each with 20 participants. We used support vector machines, random forests, and neural networks on these datasets and achieved greater than 90% accuracy in three cases.

**We explored the reasons why the low participant count is not adequate to evaluate identification systems.** We examined the impact of participant count on performance results and observed that performance may degrade as participant count increases. In addition to the participant count, we also analyzed the impact of measurement count, number of measurements per participant, and feature dimension on performance.

**We outlined a method for more rigorous testing of an identification system.** We proposed measuring how the performance metrics degrade when the participant count increases as a gauge of the robustness of an identification system. We examined how to anticipate this degradation by using randomized participant subsets, and noted it as a crucial criteria to demonstrate the performance of a novel identification system.

It is fairly well accepted that having a small participant count in a user study is inadequate to asses that system [99, 100]. However, the literature does not document the impact of small participant counts on the performance metrics of identification systems proposed in literature. To determine the scope of this problem, we describe our survey of the literature on recently proposed systems in Section 3.2. We then construct five systems from human generated datasets in Section 3.3, which explore what performance can be achieved with minimal data processing effort. We then examine how the properties of a dataset impact the performance metrics in Section 3.4. From this analysis, we identify probability distributions that play a central role in system performance. We discussed how these probability distributions create bounds on the number of easily identified participants in Section 3.5. We argue that the bounds are

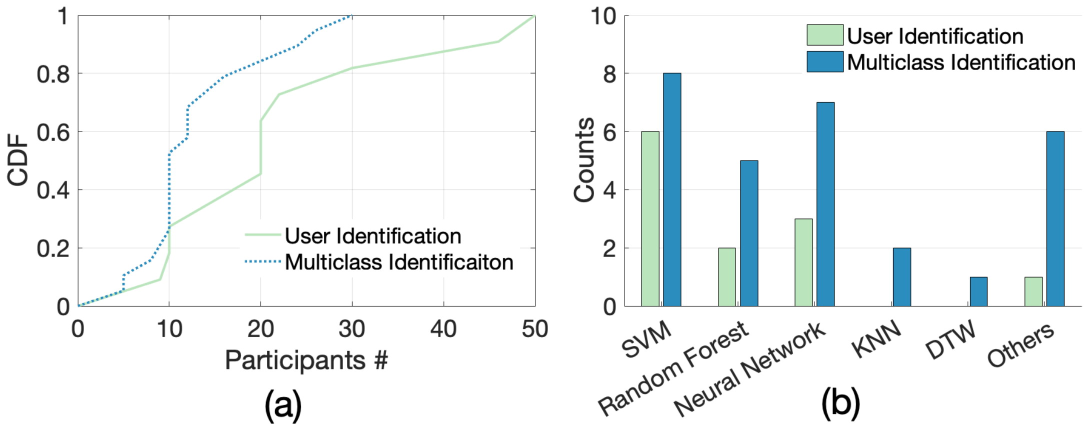


Figure 3.2: Summary of the participant counts and classification approaches used in the surveyed two types of identification systems: user identification and multiclass identification. (a) shows the cumulative distribution function of the two types of identification systems. We found more than 77% of the publications in these two types of systems recruited 20 or less participants in their user studies. (b) summarizes the classification techniques used. We found support vector machines (SVM), random forests and neural networks are among the most common. Others category includes Hidden Markov Model [101, 103], Jaccard similarity coefficient [104], k-nearest neighbors [87, 83].

a realization of limits on measurement precision. Because of the stochastic nature of these systems, it is hard to definitively identify when the bound has been reached. We suggest strategies for testing the robustness of a system without concrete knowledge of its bounds in Section 3.6. Finally, we give related work in Section 3.7 and summarize the findings in Section 3.8.

## 3.2 Survey of Recently Proposed Identification Systems

To understand how many participants were recruited in recent publications, we surveyed the papers published in top venues during the last four years (2016-2019). We focused on systems literature where machine learning is often used as a black box. We considered systems papers from top-tier conferences in mobile and ubiquitous computing, human-computer interaction and networking. These conferences included CHI, IMWUT/UBICOMP, Infocom, MobileHCI, MobiSys, MobiCom (no papers discovered), and UIST (see Table 3.1).

The systems proposed in these articles can be separated in to two cases: multiclass identification systems and user identification systems. A multiclass identification system measures a user and attempts to predict one of several classes (e.g. standing posture, handwritten digits, or a hand gesture in free space). The user identification system is a subset of the multiclass identification system where the classes are in one-to-one correspondence with the users. That is, each class uniquely identifies an individual user. For multiclass identification, the number of classes could be less than the number of participants (e.g. a fixed set of gestures). In this case the probability distributions of measurements become concentrated into a smaller set of classes. The underlying decision mechanism is still the same in both cases because the distribution of classes depends on the distribution of measurements. These systems measure users, compute distributions of classes based on those measurements, and then predict on new inputs based on those distributions. We argue that the reliability and performance of identification systems cannot be fully evaluated with the user studies of low participant counts.

We did not consider machine learning and data mining conferences, such as NeurIPS and KDD, because they focus on algorithms instead of system applications and often use curated datasets instead of generating their own data by recruiting participants. Figure 3.2 summarizes the participant counts and techniques that were used in the 30 surveyed publications. Our goal is to bring attention to the misleading results that arise from recruitment practices and to advocate for testing for failures.

## 3.3 Datasets and Construction Of The Identification Systems

To analyze the potential issues of the identification systems with small participant counts, we constructed five user identification systems using publicly available datasets. We chose the user identification task as it was easy to implement, had readily available data, and has a simple interpretation of the measurement distributions. We used common classification techniques with minimal tuning. We discovered that high classification performance is achievable when the participant count is low.

In all cases, the data was used only as a source of human-generated measurements. We did not assess the datasets usefulness for its collected purposes (e.g recognizing walking activity). We were solely interested in the discerability of the measurements from distinct participants. Our study was approved by the Institutional Review Board (IRB) of our institution.

### 3.3.1 Choosing Datasets

To identify datasets that we could use, we examined datasets from several public repositories including, UCI Machine Learning Repository [105], Kaggle.com [106], Data.gov [107], and other public data repositories. For a dataset to be included in our study, the dataset had to meet the following simple criteria:

- Unique identifier for each participant
- More than 20 participants
- More than one measurement per participant

We did not restrict datasets based on measurement type, number of features, or other dataset properties to maintain generalizability of the results. Table 3.2 lists the datasets.

### 3.3.2 Choosing Identification Methods

To construct a user identification system, we applied the three most common algorithms used in our survey reported in Section 2 - Figure 3.2(b)): random forest, support vector machines and neural networks. Each identification system was constructed with 20 users with unique identifiers. The unique identifiers served as the class labels that would be learned by the machine learning algorithms. The systems were evaluated on how well they predicted the identifier when given an unlabeled measurement. In the case of a multiclass identification system the evaluation would be the same, however, the number of classes may not be equal to the number of participants.

All classifiers were implemented using the sci-kit learn [108] library. We aimed to minimize the amount of machine learning knowledge required to implement an identification system with high performance. We performed very little optimization on each of the algorithms. In the cases where we did not use the default parameters, the selected parameters were chosen strictly to prevent infinite loops and minimize run time in order to treat the machine learning as a black box. Our principal effort was to build identification systems from human-generated data. Our goal was to demonstrate how misleading results will arise when a selection of participants produced a dataset that was artificially easy to classify.

### 3.3.3 Identification Systems Performance

The performance metrics we computed were 1) the accuracy score (ACC), 2) the confusion matrix (CM), and 3) the number of easily identified users. The first two metrics are widely reported in our survey of identification systems and machine learning literature when multi-class classification is evaluated [82, 86, 77, 76, 87, 75, 89, 90, 88, 91, 103, 80, 95, 109, 104, 110, 97, 81, 92, 93, 94, 111, 96]. The number of easily identified users is a simple metric derived from the confusion matrix (see table 3.4).

These metrics were computed using a standard validation technique where the data is split into two subsets, a training set which consists of 80% of the measurements and a testing set with the remaining 20%. For datasets where there were more than 20 participants, we also ran the analysis over multiple randomly chosen subsets of 20 participants. This was done to eliminate the possibility that a specific chosen subset would inflate the performance metrics purely by chance. A large variation in performance across these randomized subsets would indicate bias in the identification system.

No preference was given to any particular algorithm. In the case of support vector machines and neural networks, the default parameters were used. For random forest we set the n_estimators (number of decision trees to test) to 1000 to ensure a large breadth search and the max_depth to 20 to prevent infinite loops (more details in Appendix 5.2).

#### Accuracy Score

The accuracy score provides a simple summary of performance by computing the relative frequency of a correct decision. This summary, however, is incomplete as it looses details of the systems performance on each individual participant. Additionally, the

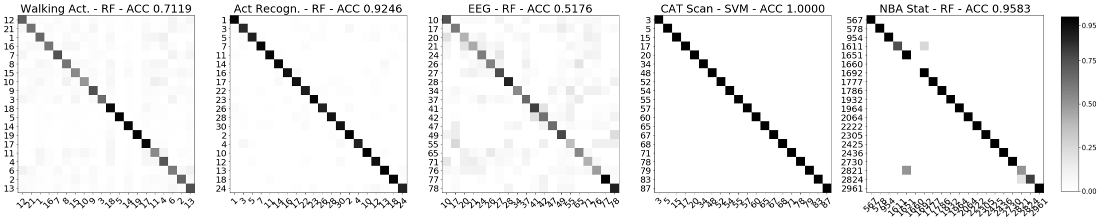


Figure 3.3: Each confusion matrix and accuracy reported in this figure represents the best achieved performance across 10 iterations. A solid black square on the main diagonal means that the participant represented by this identifier is easy to identify. In all cases there were at least a few participants that were easy to identify which implies that their measurement values were distinct. The EEG dataset was the only case where the off-the-shelf algorithms were unable to achieve the accuracy goal across all participants. The axes contain the numeric identifiers used in the most favorable run, the identifiers have no ordering.

accuracy score can be misleading when the number of measurements per participant is unbalanced [98].

To calculate the accuracy score (ACC), we ran ten iterations with randomized participant subsets and reported the best accuracy achieved across all sets. Table 3.3 shows the best results achieved across ten different re-samplings. *This method of model selection demonstrates a scenario where the performance is misleading because of the serendipitous favorability of the dataset.* In one case the reported accuracy was 100%.

The classifiers performances were similar when the measurement values from different participants was very distinct. In all cases, the accuracy metric varied by at most $\approx 8\%$. The values presented in Figure 3.3 are among the highest observed to highlight what is achievable with favorable subsets. Table 3.3 shows that three of the data sets had at least one algorithm that was able to achieve 80% accuracy. *Thus, an algorithm was discovered that would achieve reasonable performance with minimal tuning.*

All algorithms performed poorly on the EEG readings dataset regardless of parametrization of the algorithms. The best achieved accuracies were $\approx 52\%$ for random forest, $\approx 51\%$ for neural networks and $\approx 33\%$ for support vector machines. Parameter optimization techniques such as grid / random parameter search did not improve the results. This observation informs the strategy we propose and we will discuss why the reasons for this failure lie with the measurements in Section 3.4.

### Confusion Matrix (CM)

To illuminate the identification performance we computed the confusion matrix. It is a contingency table that tabulates how often one participant identifier is confused for another. The confusion matrix allows us to identify when participants fail to be distinct. Figure 3.3 shows both of the accuracy score and confusion matrix results.

### Number Of Easily Identified Users

A user is easily identified if their measurements are classified correctly most of the time. This can be computed as a count of the main diagonal probabilities which are above a threshold (e.g. 80%). Table 3.4 enumerates the number of participants that can be easily identified by the classifier.

The strength of this metric lies in its dependence on how the measurement values separate participants. Because this metric is more sensitive to measurement separation than accuracy, it gives a more meaningful summary of the identification systems performance. Using this second metric we can see that the accuracy does not always give the full picture. For example in the NBA dataset, random forest has a slightly higher accuracy than neural networks, however, neural networks easily identifies more participants. This would indicate that neural networks are better at finding the structural separations in this type of measurement data. In contrast, random forests do significantly better at separating users in the walking activity data. Finally, support vector machines have a slight advantage in the CT scan data, because of the accuracy metric, even though all algorithms can easily classify all participants. *Multiple metrics help to identify cases where a single metric is artificially high because of artifacts in the data.*

In summary, we noticed the performance of identification systems varies across the different data sets and classification algorithms. We will analyze the reasons that cause the differences in performance.

## 3.4 The impact of dataset properties on classification performance

The reliability of an identification system depends on the generalizability of classification. Generalizability here means how the classifier performs with participants which it was not trained on. To achieve such generalization, the dataset will need to be both representative of the intended user base, and contain enough information to ensure that the distributions governing the observations of measurements are approximated well. Predicting the generalizability of a system is often a difficult task because the properties of the reported datasets can only be used to gauge expected performance under certain constraints.

The representativeness of a dataset is related to the diversity of participants and dimensionality of the feature space. The expectation is that measuring more observables from a diverse array of participants would yield a model that has better coverage of the intended users. We show in this section that it is very difficult to define the diversity of a dataset. For example, we often lack ground truth about its intended user base. We also demonstrate that feature dimensionality rarely predicts performance because the size of the feature space is not a predictor of discernibility.

The size of the dataset is often used to gauge whether there are enough measurements to deem the approximation of a distribution sufficient. Generally, more datapoints yields a better approximation of a distribution [3]. However, it is often unclear which distribution is being approximated. We argue that sample size is poorly defined and neither of the potential definitions is sufficient to predict the error in approximation of the distributions that the system is attempting to learn. There are two types of distributions that impact performance and each definition of sample size is only related to one of them.

We compare impact of variation in the dataset properties, including measurement count, participant diversity and feature dimension (Table 3.2) on system performance when the participant count was fixed at 20. All of these properties of the data sets may affect performance because all algorithms attempt to optimize their fit of the data [6]. These differences between datasets can be lensed through the distributions that the identifier is trying to learn.

### 3.4.1 What is Sample Size: Participant Count or Measurement Count?

Sample size is a term used often, but unfortunately ambigious. Different disciplines do not agree on the definition of a sample [112]. Different disciplines tend to focus on different aspects of the analysis, the term sample size get used in two ways.

Often in the HCI literature, sample size means the number of participants. However, in the machine learning literature, sample size is usually used to refer to the count of measurements taken across all participants.

To avoid confusion, we will explicitly identify either the participant count, $N$, or the measurement count, $T$. One critical observation is that a large measurement count does not imply large participant count. The number of participants within a study and how the participants were selected impacts sample diversity [113, 114], which is a gauge of how different the measurements from distinct participants are.

#### Why Do the Differences between Participant Count, $N$, and Measurement Count, $T$, Matter?

The classification performance is controlled by two types of distributions: 1) the distribution of measurements taken from all participants, $P(V)$ (also known as the population distribution), and 2) the distributions of measurements from a single participant, $P_h(V)$ where $h$ is a unique index for the participants. We will call this the individual distribution. Both of these distributions are approximated by machine learning algorithms when it fits a curve to the points in the dataset. The errors in approximation are directly related to the number of points within the dataset, but each of the different counts is only related to a specific distribution. We argue that participant count, $N$, can be used to gauge the error in approximation of the distributions among participants if there is sufficient participant diversity. We also argue that measurement count, $T$, alone is insufficient to gauge whether either distribution is well-approximated because it does not represent how many measurement points exist per individual. Thus, a critical difference between $N$ and $T$ is that, under specific conditions, $N$, can be used to compare the evaluations done between two systems, while $T$ cannot.

The population distribution, $P(V)$, is illustrated in Figure 3.4 (B) and (D). It captures how a specific measured observable (or sets of measured observables) varies across the intended user base. It is approximated by the relative frequency of a measurement values collected across a sampling of that user base. The individual distributions, $P_h(V)$, are illustrated in Figure 3.4 (A) and (C). Since the measurements from each participant may have different variation characteristics, each participant, $h$, has their own distribution $P_h(V)$. How these $P_h(V)$ distributions overlap will impact how well the measurements separate, and ultimately dictate the system performance.

If there is sufficient participant diversity, then larger participant count, $N$, may imply a better approximation of the population distribution, $P(V)$. In the absence of a systematic selection biases, as $N$ increases, so do the chances of observing distinct measurement values. Thus, when $N$ is sufficiently large, we can consider $P(V)$ well approximated as more of the range of measurement values has been explored.

Total measurement count, $T$, is insufficient to ensure that $P(V)$ is well-approximated. The number of measurement per participant, $M$, also impacts the approximation. If we assume that each individual distribution, $P_h(V)$, is well-approximated and that the sampling has no systematic bias, then a larger $N$ may also imply that the set of distributions has better coverage over the range of possible measurement values.

Both distributions are necessary to estimate the details of data. The population distribution, $P(V)$, can inform the sampling procedure. If we knew the population distribution, we could build our recruiting policy using standard techniques [115, 116, 100, 99, 117, 118], allowing us to answer questions such as "For a given measurement value (or values), $V$, how likely, $P(V)$, is it to observe this $V$ from the population?" However, the population distribution cannot discern whether the measurement type effectively separates individuals because the distribution lacks that granularity. By looking at the overlap in individual distributions, $P_h(V)$, can be used to gauge if the measurements are adequate for distinguishing individuals in the sample of the intended user base. However, the $P_h(V)$ distributions can not discern if there are additional distinct individuals among the intended users that were not accounted for. To answer this question we could use the population distribution, $P(V)$, to identify probable measurement values that are not present in our data (e.g. values beyond 5 and $-5$ in Figure 3.4). The individual distributions, $P_h(V)$, also cannot assess the likelihood that an unseen user is easily confused with the members of the sample. To solve this issue, we would need to compute the conditional probabilities of overlap which uses the $P(V)$ distribution as a prior. Thus, a purely mathematical analysis would require both distributions in order to determine how many individuals a system must be tested with before the performance degrades below tolerable levels.

Given that distributions of users are rarely known, user studies help to estimate these distributions. In most systems, there is an enrollment phase [4] in which several measurements are taken from all the participants, $h$, that are available and an estimate of $P_h(V)$ is made. This process is applied to each $h$ which may use the system. When the system encounters new, unlabeled measurements, it makes a decision using these estimated distributions. The quality of the distribution estimates is a function of the number of measurements taken from each individual. Since each individual has a different level of variability for any type of measurement, the number of measurements required to get a good estimate will generally not be the same across individuals. Even though these differences are present, in practice a large number, $M$, is selected which accommodates some range of variability across all $h$ and ensure that all $P_h(V)$ are well-approximated. Under this assumption $M * N = T$.

For any given measurement value, $V$, if only a small number of participants have a high probability of producing that value, the performance will be high. This is a property of what is being measured, and we consider the measurement discerning when this happens (case (A) in Figure 3.4). When this is true, we may be able to identify individuals in the sample population easily. In contrast, if many individuals produce the same $V$ when measured, then the probability, $P_{h'}(V)$, of observing a $V$ from a randomly chosen individual, $h'$, is high, thus this measurement type will be less useful for identifying individuals (case (C) in Figure 3.4)). The multiclass classification problem tries to identify $h$ for an arbitrary $V$ by considering all the probabilities, $P_h(V)$, across all $h$ for which we have an approximation of the individual distribution. In the simplest case we can ask the question “for any $V$ which $h$ was the most likely to produce this $V$?” (however the decision logic is often more complex, e.g. taking into account correlations between individuals).

### The Impact of $M$ and $N$ on Performance

It is difficult to gauge how effective a system will be at identifying individuals just by looking at the number $N$ of users used to test it. Consider the examples of identifiers built for the EEG dataset from section 5.1.4 and the CT scan dataset of section 5.1.3. We observe significantly different performance (CT scan $ACC \approx 100\%$ vs. EEG $ACC \approx 52\%$) for two datasets where the number of participants used to build the identifiers was kept the same, $N = 20$. For each dataset the average number of measurements per participant, $M$, was comparable (see Table 3.2). Therefore, the total number of measurements, $T \approx M * N$, was comparable. The significant difference in performance can only be explained by the discernibility of the feature space, which is a function of the individual distributions, $P_h(V)$. The performance of the random forest classifier on the CT scan dataset only dropped by $\approx 1\%$ when the participant count was increased to $N = 80$ (See Figure 3.5). In contrast, using the random forest classifier on a subset of 10 participants for the EEG dataset achieved an increased accuracy of $\approx 65\%$.

A system with more samples per participant (higher $M$) will not necessarily yield better results. For example, the NBA data set of Section 5.1.5 has an order of magnitude fewer measurements per participants than the activity recognition data set of Section 5.1.2, yet it achieves $\approx 3\%$ higher maximum accuracy. If the feature space is highly discerning, then under-sampling the participants may not cause significant degradation in performance because the individual distributions are spread apart. On the other hand, if a feature space is not discerning, sampling each participant further will not produce any improvement. The approximations of the individual distributions will become tighter, but distribution overlaps will remain the same.

### 3.4.2 How Participant Diversity Affects Performance

The two types of distributions from Section 4.1 highlight a key challenge when formulating a participant recruiting policy: ensuring that you have covered the breadth of measurement variation within a population. To be sufficiently representative, a dataset must collect measurements from a wide range of distinct individuals in order to determine if we have adequately covered the range of measurement values that have non-trivial probabilities in the population distribution, $P(V)$. If the value range is not covered, the generalization beyond the participants recruited will suffer because a system built from this dataset will encounter measurement values in the intended users that are significantly different than the values with which it has been trained. These circumstances render the behavior of the system indeterminate.

For any population, diversity refers to the degree of difference between members of that population. Other disciplines, such as ecology, attempt to categorize the variation within in a population by computing the Shannon entropy of the probability distribution on specimen observation [113]. Such metrics often assume that all specimens are readily distinguishable from each other, that is, low variation within an individual. It is expected that when a rare specimen is observed, the observer would be able to easily recognize that the observed specimen is distinct from the previously observed samples.

In comparison, it is more difficult to determine population diversity within the context of identification systems, because we are viewing each participant through the lens of limited precision measurements. In Figure 3.4 we observed cases where the participant count, $N$, was inadequate to cover the entire range of possible measurement values, $V$. A fully characterized feature space requires the recruitment of more participants. If the measurement values not observed are rare (e.g. 7.5 in case (B) of Figure 3.4), then we may need to recruit significantly more participants before we observe these rare values. If we could sample until we covered the full range of values, we would ensure that we have a reasonable participant diversity. Unfortunately in most cases, the range of values is not known apriori.

The number of participants in a study, $N$, cannot always be used to directly determine the if the study was sufficiently diverse because of the break down in distributions covered in Section 3.4.1. Consider the participant counts of the CT scan and the EEG datasets in Table 3.2). The two datasets have similar number of participants, but the best achieved performance of an identification system built from EEG dataset was 65% accuracy when the participant count was 10 (see Figure 3.5). As the participant count increased, the system performance degraded. We argue that the system is well tested because we can identify a participant count beyond which the performance guarantees no longer hold. In contrast, the performance of the identification system built with the CT scan data does not degrade significantly as the participant count increases. For this identifier, the user limit is unknown because our tests could not produce a reduction in performance even with 80 users. This difference in performance as the participant count increases will play a large role in the robustness of systems built upon these identification systems.

### 3.4.3 How Feature Dimension Affects Performance

Feature dimension is a count of the number of distinct measurement types and functions of the measurements values that make up feature vector. It does not consider redundancy among features in the feature space nor does it contain any information about the distribution of measurement values. For example, if we use a length as a feature, this length could be measured in meters or kilometers. Since one value is simply a scalar multiple of the other, good algorithms will treat them as the same feature. Techniques such principal component analysis can be used to reduce the feature dimension by considering the minimal number of vector components required to represent the information within the feature space.

*Feature dimension is rarely useful for predicting how well a system will preform.* Despite this, it is often reported. It cannot be used to perform relative comparisons of identification systems. For example, the identifier built from the walking activity data set uses only four features (see Table 3.2), but still achieves $\approx 71\%$ accuracy (see Table 3.3). In contrast, the EEG data set identifier has 39 features to work with but performs significantly worse ($\approx 51\%$ at best) than the walking activity data set identifier. Further still, the NBA data set identifier uses 51 features and achieves $\approx 95\%$ accuracy. The CT scan dataset identifier has a lower feature dimension to work with than the activity recognition dataset identifier, but performs better. The ability of a system to distinguish individuals is only as good as the discernibility of measurements will allow.

## 3.5 Bounds on the number of easily identified participants

All identification systems degrade in performance as the number of users increases. This is because the system reaches the upper bound (or the upper limit) of easily identifiable users. In this section, we show that the upper bound arises because each measurement obeys a natural distribution across the intended users, $\mathcal{P}(V)$.

The natural distribution, $\mathcal{P}(V)$, can only assign non-zero probabilities to a finite range of values. All participant measurements values are drawn from this range. As the participant count, $N$, increases, it becomes increasingly probable that any measurement value will be observed from multiple participants. To be able to predict when the system will fail, we need to know at what participant count, $N$, performance begins to degrade beyond tolerable levels. The upper bounds in all systems differ in how large that bound is, and how quickly it can be found in a practical setting. In the example of the identifier built from CT scan dataset of Appendix 5.1.3, the upper bound is very large (see Figure 3.5). Thus, any system based on these measured observables would be able to distinguish a large number of participants. It would not be practical to build an identification system that requires a full body scan of an individual to perform the identification. However, the system demonstrates a case where machine learning algorithms do most of the work with minimal configuration.

We can notice trends in performance with a scatter plot of the principal components as $N$ increases. Figure 3.6 shows four scatter plots of the first two principal components for the NBA stats dataset. Each scatter plot is labeled with the achieved accuracy and participant count, $N$. When participant count is small ($N = 20$), the measurement separation is very good, and thus, the classifier will easily discern participants. As $N$ increases, we can see crowding within the center of the graph. This crowding is a form of concept drift [119], where the conditional distribution of a participants identifier given the measurement, $P(h|V)$, changes as the number of participants grows. In our case, an increased participant count is the source of the distribution shift as opposed to temporal drifts which are normally observed. This comparison demonstrates that performance analysis with a small number of participants (compared to the bound on participants that can be represented) is incomplete at best.

### 3.5.1 Why Are the Metric Values of Low Participant Count Studies Misleading?

If variability of measurements within an individual is low enough to produce distinct participant measurements, such as those shown in Figure 3.4 (A), then the system will only make mistakes if the recruiting procedure produces two individuals with the similar individual distributions, $P_h(V)$ (e.g. similar location and scale). In a Bayesian formulation, the equivalent condition is $P(V) \propto P(h)$.Thus, the performance of either class of identification system with small number of distinct participants, $N$, is largely dictated by the population distribution $P(V)$. The distinctiveness of participants implies that the probability of observing a specific measurement value is very dependent on recruiting a specific individual.

The identification system can be thought of as partitioning the space of measurement values into bins which correspond to the classes (as in Figure 3.7). In the case of user identification, each distinct participant's measurements might neatly fall into a bin designated for them. As long as the participant count is low, the values in each bin will mostly have come from distinct participants with overlaps being rare. Such a condition would artificially inflate summary metrics that try to count mistakes, for example, accuracy or the confusion matrix, because mistakes are artificially rare. As the number of participants increases, $N \nearrow$, it becomes more likely that a bin will have measurements from more than one participant. If there are naturally $\mathcal{N}$ distinct bins, then the pigeonhole principle [121] guarantees that there will eventually be at least one bin with multiple measurements from different participants and thus the system will start to accumulate errors which will drive accuracy down.

For multiclass identification systems, the number of bins may not coincide with the number of participants. In many cases the raw measurements are mapped into a different representation via a deterministic function to achieve the same bin separation. When the number of bins is smaller than $N$, a single participant might get mapped into multiple bins. Although this mapping may lower the effective number of participants required by concentrating them into fewer classes, it does not eliminate the problem that too few participants underexplores the space of possible inputs. Thus, it is still necessary to test with increased participant count to ensure that the space of possible measurement values does not contain values that fail to be mapped.

Machine learning algorithms are designed to optimize the amount of information that is extracted from measurement data. This optimization tries to bin measurements to have a maximal separation between classes. This separation translates to classification performance, for example, accuracy. The problem of misleading metrics arises when the participant count is so small that the ability to draw these boundaries is artificially easy. When this occurs, *the performance of a system is less influenced by the effectiveness of the measurements at separating participants and more influenced by the how diverse a sample population the recruiting process produces.*

## 3.6 An Iterative Approach to Testing Systems

In this section, we describe an iterative approach for testing identification systems. Instead of setting a goal number for participants $N$ in the beginning of the study, we can keep increasing $N$ while studying the system and its performance. This is in contrast to statistical group comparisons in experimental designs using null hypothesis statistical testing (NHST) where you must set the target $N$ in advance.

We cannot make performance guarantees on identifications system by only knowing the number of participants $N$. This is because we do not know the upper bounds as described in Section 3.5. Although it may not be possible to know how far we are from the bounds, we can gauge how the performance degrades as $N$ increases. This can serve as a method for relative comparison of systems. We can take an iterative approach where we assess how the performance metrics of the system react to increasing $N$ without starting a study with a large set of participants. Instead, we can iteratively add participants to the study until we have identified an $N$ that causes the performance to degrade below a tolerable level. We describe this in Algorithm 1.

The analysis in Section 3.4 demonstrates no single property of the dataset is a good indicator that the resulting system will perform well. Individually, none of these values can guarantee that the experiments conducted truly tested the generalization limits of an identification system. Even when these values are optimal, such systems might still be susceptible to unexpected identification errors if participants were chosen with some systematic bias which causes the dataset fail to be representative. It is usually not possible to know the variability of the measured quantities apriori. Thus, it is difficult to construct a practical recruiting policy that eliminates all possible bias before collecting some measurements.

By establishing trends in how the performance varies when different subgroups of participants are selected and degrades when the sample diversity increases, we can determine if a system requires further testing. Cycling the participants into randomized subsets compliments the approach of increasing participants. It can identify subsets of the population that are artificially distinctive, which would produce higher than normal accuracies. It can also identify subsets which are very similar, which would lead to lower than normal accuracies. This cycling provides another check on how brittle the models learned from a specific size subset are. Sample diversity itself is difficult to measure directly. Instead, we can use participant count, $N$, as a proxy for sample diversity with some considerations. We need to ensure a reasonably sized number of samples per participant, $M$. This $M$, will can be chosen as the largest number of samples required to ensure good approximation of $P_h(V)$ for all $h$. It has to be determined empirically after the initial set of measurements is taken by looking at the variance of the distribution estimations. We would also need to eliminate systematic participant selection bias in our recruiting process by identifying factors in the process that might limit the range of measurement values.

**ALGORITHM 1:** Iterative approach to testing

```
Data: Initial $n$
Result: Plot of performance metric vs $n$
Collect $m$ labeled measurements from all $n$ participants (where $m$ is sufficiently
  large as in Section 3.4.1);
Choose randomized subsets from $n$ participants;
foreach Randomized subset do
    Build model on data of subset;
    Compute performance metric for each model;
end
Compute interquartile range (IQR) as a measure variability of the performance
  metric across subsets;
while Performance metric above tolerable level and performance metric
  unstable do
    Increase number of participants to $n'$ ;
    Collect $m$ labeled measurements from new participants ;
    Choose randomized subsets from $n'$ participants;
    foreach Randomized subset do
        Build model on data of subset;
        Compute performance metric for each model;
    end
    Compute IQR;
    $n \longleftarrow n'$;
end
Plot metric against $n$ with interquartile range error bars;
```

### 3.6.1 Test with Increasing Participant Counts, $N$

The rate of degradation of the performance metrics is both a function of the dataset and the machine learning algorithms. We argued in Section 3.5 that performance degradation as participant count, $N$, increases will occur regardless of algorithm or sampled dataset. However, each system will differ in the rate of degradation because each algorithm has a different efficiency for extracting information from the dataset and each dataset's representation of the natural phenomenon begin measured is of varying quality. The differences in the rate of degradation can be used to compare both algorithms (see Figure 3.8) and measurement types (see Figure 3.5) as each exhibits differing rates of degradation as $N$ increases.

The rates of degradation may differ for each algorithm and dataset pair. The bound on easily identified users is largely a function of the measurement type. Specifically, it depends on the amount of information that can be extracted from a measurement type. We will always observe a steady degradation in performance as new participants are added, even in the ideal case. The identification system will only experience gradual drops in performance. For example, it not possible that the accuracy remains constant at 100% up to $\mathcal{N}$ participants and then goes to 0% with the $\mathcal{N}+1$ participants, unless all $\mathcal{N}$ participants are replaced with a new set of $\mathcal{N}$ participants that are distributed differently. Knowing the rate at which new participants degrade the performance gives us an idea of where the bound on users that can be identified while satisfying a constraint on error occurs. The perfect identifier scenario would lead to a degradation rate that is proportional to the probability of observing each individual in the population (see Figure 3.1). However, most identifiers will not be perfect, and the rate of degradation will often be faster than this ideal scenario. In Figure 3.5, we observe that a single algorithm (parameterized in the same way) has very different performance degradations as the number of participants, $N$, increases up to 80. The various sensitivities to increasing participant count gives us an idea of how the system will fail when the participant count grows larger than anticipated.

### 3.6.2 Test with Randomizing Subsets

When the number of participants is limited, an additional check for unstable identification performance is to select multiple randomized subsets of participants and evaluate models derived from these subsets independently. Similar to leaving-out-k cross validation [120], we obtain multiple values for the performance of the system. Randomizing the subsets differs from standard cross validation in several ways. First, when a subset of participants is chosen, the data is partitioned into a training and testing portions. The model is trained and tested with the respective portions. This includes computing the performance metric. Participants not chosen for this subset will not be evaluated against this model since no training data was present for them. Additional analysis could be done by testing the system with these left out members, though this analysis would produce a different performance evaluation (a test of out-experiment generalization). Secondly, the distributions being trained on changes with every subset, instead of being drawn from a common pool as is the case in cross validation procedures. Each subset iteration builds a new model which reflects the current subset. Thus, the process of randomizing subsets is not testing a specific model, but instead the ability to construct discerning models across a breadth of user sub populations.

Figure 3.5 shows how different measurement types exhibit significant variation in scale. This variation is a function of the discernibility of measurement type. If a measurement type has high discernibility, then the performance metrics will be stable across a wide range of participants.

By cycling different participant subsets, we increase the likelihood that our system will encounter subsets of participants that may artificially increase performance. When the algorithm is fixed, each subset will yield different overlap behavior for natural distributions, $\mathcal{P}_h(V)$, for all $h$ in the subset. These distributions will result in differing boundary choices (see Figure 3.7). The differences between subsets will ultimately dictate the variation in performance. If the variation in performance across subsets is significant, then the performance may degrade quickly as the participant count, $N$, increases because participants that are hard to classify will be added to the subset. On the other hand, if the variation across subsets is not large, we will need to test with more participants to determine the limit on easily identifiable participants.

We can measure the spread of performance metric values (e.g. IQR or variance) as the subsets are varied to quantify how much a metric value can change when a subset is favorable. If the subsets are very different, the spread will be large. In Figure 3.5, for each $N$, ten participant subsets of size $N$ were chosen from the larger dataset. Then, a classifier was trained and tested with the data from these ten subsets. The CT scan identifier barely degrades as $N$ increases, and the error bars for the metric IQR across all subsets are so small that it is not visible on the plot. In contrast, the identifier built on the EEG dataset starts with varying performance, but this variation stabilizes as the performance degrades. The performance stabilizes because the subset distributions become more stable (and overlap significantly more) as the participant counts increases.

Thus, after some intermediate $N$, it becomes more difficult to find a subset of the population which will separate well by chance. Since the performance metric variability on subsets is only a weak gauge of the potential performance degradation, the technique of cycling subsets of participants should be used to augment the analysis with increasing participant counts.

## 3.7 Related work

The subject of sample size has been discussed for decades [99, 100]. Studies have determined sample size with power analysis where the shape of the distribution is assumed to belong to a specific family [117, 122]. In these cases samples size means participant count and the number of measurements per participant was one. This class of analysis is not applicable to studies where the goal is to build a decision system based on multiple measurements from many diverse participants. The assumed distribution families are too simplistic and the number of measurements from a single participant is often greater than one.

The issue of low participant count in user studies is common across different research communities. Caine [115] analyzed the sample sizes of all 465 manuscripts in the proceedings of CHI 2014 and found the common sample size is only 12. In the HCI community, researchers may try to mitigate the issue by collecting more data from the same participants. However, we showed that this approach is not adequate because each participant can only provide a limited amount of variation for a given measured observable.

Many other fields have raised concerns when the participant count is low. Raudys and Jain [123] discussed the influences of sample sizes on feature selection and error estimation for different types of simple classifiers such as Euclidean distance classifier and Fisher's linear discriminant. Button et al. [118] showed the average statistical power of studies in neurosciences is very low. They emphasize that this situation leads to overestimates of effect size and low reproducibility of results. Anderson and Vingrys [116] proposed three situations that need to be considered while conducting research with small sample sizes in psychophysical and neurophysiological studies. Hackshaw [124] overviews the strengths and limitations of small sample size in clinical studies. The main issue for small participant count studies is that the outcomes have large standard error and no firm conclusions.

Our survey shows that small participant counts are an issue in user and multi-class identification studies. The breadth of previous work focused on trying to learn population distribution parameters of an assumed distribution shape. These shape assumptions were used as a guiding principle to select participant count. Bounds on error were established only as an afterthought by considering measurement variation after the procedure was done.

In our approach, we do not make strong assumptions about the shape of the population distribution to determine when our statistics have converged. Instead, we apply an iterative approach that uses the error metrics to identify the limits of the system's ability to discern individuals. We update our model as new data is made available and estimate distributions we cannot know a priori. Our approach provides a method for reasoning about a problem that is often poorly defined without making assumptions that limit generalizability. Our approach is adaptive and can react to shifts in the population and unexpected experimental conditions.

## 3.8 Discussion and Conclusions

We have shown that testing an identification system with a small number of users is rarely adequate and often misleading. To demonstrate this, we constructed five user identification systems from publicly available datasets. Three of these systems yielded $\geq 90\%$ accuracy when the participant count was small.

To explain why such misleading results can arise from low participant count user studies, we delved into the properties of the measurements that would impact these metrics. We demonstrated that as the participant count increases, the system performance must decrease. We reasoned that because all measurements can only be made with finite precision, an upper bound on the number of easily identifiable individuals must exist. As the participant count approaches this bound, the performance of the system must degrade.

We showed the issue of low participant count is common in the user and multiclass identification studies by surveying the recently published papers in top-tier venues. Seventy-seven percent of the 30 surveyed papers were supported by user studies with 20 or less participants. Although some of the work collected thousands of measurements from the small participant sets, we argued that these measurements do not compensate for a lack of sample diversity, which is mainly affected by the variation between participants.

We showed that the participant count can be used as a proxy for sample diversity, given that the user study factors are controlled. We can establish an estimate that gauges how the system performance will degrade when the participant count increases. We demonstrated that performance metric variation on randomized participant subsets can be a useful approach to diagnose performance degradation when the participant count increases. Knowing these factors will enable us to reason about the likelihood of failure for a system in a target application.

*To conclude, we argue that limit on easily identified participants can and should be experimentally determined by increasing the participant count iteratively.* There is no single participant count that will be sufficient for every experiment. As such, we cannot prescribe a fixed value or gauge what a large value would be. To learn how a system performance degrades when the number of participants increases, it is critical that we recruit until it fails.

Table 3.1: Publications surveyed grouped by publication venues. Others category includes Infocom, MobileHCI, and MobiSys

| Venue | Case 1: User Identification | Case 2: Multiclass Identification |
|---|---|---|
| IMWUT/UBICOMP | [101], [72], [73], [74], [75], [76], [77] | [82], [83], [84], [85], [86], [87], [102] |
| CHI | [78], [79], [80] | [88], [103], [89], [90], [91] |
| UIST | | [92], [93], [94] |
| OTHERS | [81] | [95], [104], [96], [97] |

Table 3.2: We chose datasets that had at least 20 participants. The total measurement count is often misinterpreted as the sample size. The average number of measurements per participant can indicate how well-characterized the statistics of an individual participant is. The feature dimension is an indication of the complexity of what is being measured.

| | Act. Recogn. | Walking Act. | CAT Scan | EEG | NBA Stat. |
|---|---|---|---|---|---|
| Participant Count | 30 | 22 | 97 | 81 | 296 |
| Total Measurements | 10299 | 149332 | 53500 | 23986 | 5444 |
| Average Measurements | 343 | 6788 | 552 | 296 | 18 |
| Feature Dimensions | 562 | 4 | 385 | 39 | 51 |

Table 3.3: Maximum Accuracy - We achieved greater than 50% accuracy for all datasets with at least one algorithm. Each algorithm was tested on randomized subsets of each dataset for 10 iterations. In all cases where there were more than 20 participants in the dataset, each iteration was done with a random choice of 20. Thus we can achieve very high accuracies if we carefully select our participants and algorithms.

| | EEG | NBA Stat. | Act. Recogn. | Walking Act. | CAT Scan |
|---|---|---|---|---|---|
| Neural Network | 0.51 | 0.95 | 0.82 | 0.5901 | 1.00 |
| Random Forest | 0.52 | 0.96 | 0.92 | 0.7119 | 1.00 |
| SVM | 0.3297 | 0.79 | 0.79 | 0.57 | 1.00 |

Table 3.4: Easy Identification - We define participant easily identified when they are identified correctly at least 80% of the time. Note that the accuracy of random forest is higher than neural network in the NBA case (see Table 3.3), though neural networks easily identifies more participants. This is because accuracy only considers correct decisions without concern for whom they occur. In these datasets, random forests make more correct decisions overall, but neural networks have more certainty per individual.

| | EEG | NBA Stat. | Act. Recogn. | Walking Act. | CAT Scan |
|---|---|---|---|---|---|
| Neural Network | 1 | 18 | 12 | 2 | 20 |
| Random Forest | 1 | 16 | 20 | 5 | 20 |
| SVM | 1 | 12 | 12 | 1 | 20 |

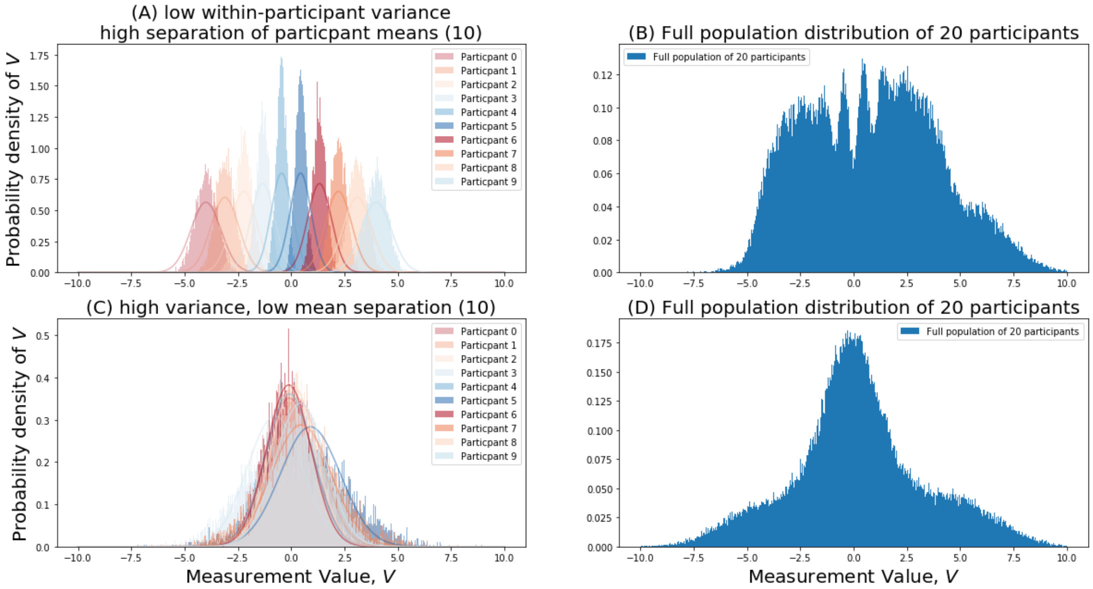


Figure 3.4: The variation of a measurement's value has two possible sources. The first is random fluctuations in the value when repeated measurement are taken from an individual (i.e variation within an individual). The second is random fluctuations in the value when measurements are taken from many individuals through out the population. If the variation within an individual is small compared to the variation among individuals then these measurements may separate participants well. For example, the ten participants in (A) separate well. When the variation condition holds, any measurement value, $V_h$ from a particular participant $h$, has a low probability, $P_{h'}(V_h)$, of coming from another participant $h'$. For example, repeated CT scans within a small time window of a single individual should have very little variation, however, these scans should be very different between individuals because they measure the entire body. In contrast, when the variation among participants is high compared to the variation among users (C), all measurement values have comparable probabilities of coming from each user (e.g. EEG data has so much noise in the measurement that any individual value can easily have come from many individuals). The population distributions shown in (B) and (D) show the measurement range of the full population of 20 participants. This distribution can tell us if a measurement value is reasonable for a population, that is, has a non-zero probability of being observed. It cannot be used to determine whether the measurements are useful for distinguishing individuals.

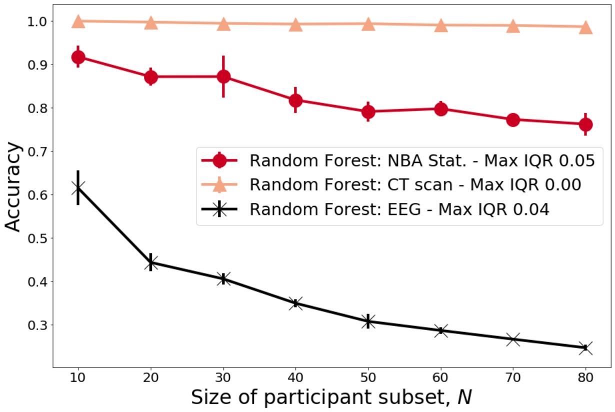


Figure 3.5: We can compare the accuracy of the random forest classifier when run on three of the data sets with enough participants to increase $N$ significantly. For each $N$, we perform 10 runs with randomized subsets of size $N$. The CT scan dataset has significantly less variation in the performance metric. We measured this by looking at the interquartile range of performance metric values when the given randomized participant subsets. This variation in the CT scan dataset was an order of magnitude lower that the other datasets. The range is not visible in this figure because of scale. Because the CT scan dataset is resilient to variations across humans, its performance degrades much more slowly as the number of participants increases. The variability between randomized subsets may give an indication of how performance will degrade as the participant count increases.

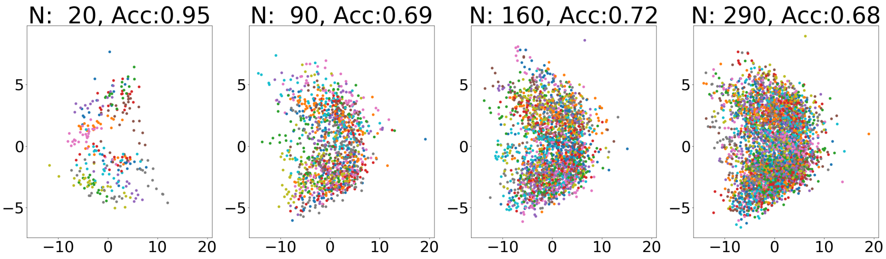


Figure 3.6: As we increase the size of the participant count, $N$, the first two principal components become more crowded. The feature space in often multi-dimensional and these two principal components are the most variable linear combinations of the components in the feature space [120]. The clustering of points in this figure is a two dimensional representation of the per-individual distributions, $P_h(V)$ discussed in Figure 3.4. This crowding explains why the performance of the identifier will degrade with increased $N$. When $N$ is large, the various point distributions overlap into an incoherent mass.

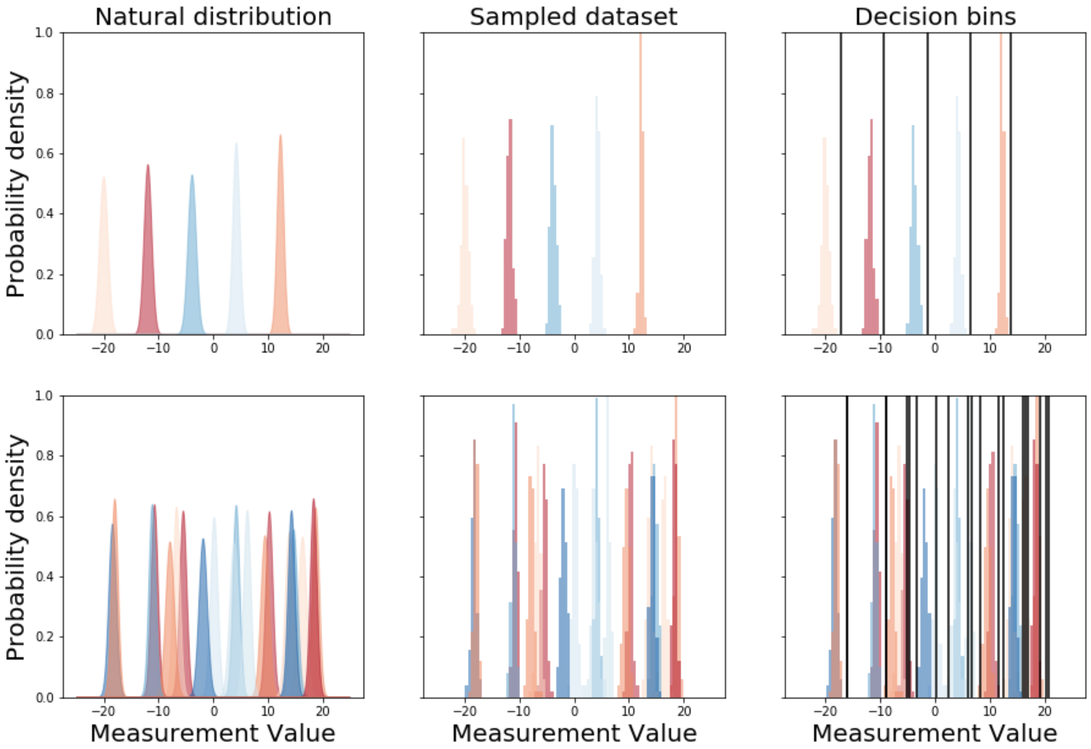


Figure 3.7: When the variation within participants is low (low variance in $P_h(V)$) and the participant count, $N$, is small, the probability of observing a measurement value is proportional to the probability of picking that participant, $P(V) \propto P(h)$. At low $N$, samples drawn from the natural individual distribution, $\mathcal{P}_h(V)$, separate well. Thus, it is very easy to draw the boundaries of the bins (e.g. the top row where $N = 5$). As the number of participants increases, the $\mathcal{P}_h(V)$ over lap, thus the samples will overlap, and the bins become more difficult to draw (e.g. the bottom row where $N = 20$). The identification system tries to optimize the placement of bin boundaries by using the sampled dataset as an approximation for the natural distribution.

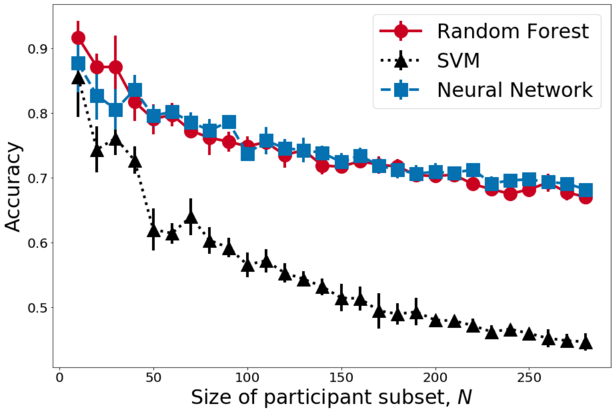


Figure 3.8: In the NBA dataset, we can observe a drop in accuracy as the participant count, $N$, increases regardless of which algorithm was used. For each algorithm, we trained a model with $N$ participants taken from the full population of 290+ players. We then calculated the accuracy for each model and repeated the process for ten iterations. The graph shows the median value and the interquartile range used as error bars. As $N$ increases, we can see that all models' performance degrades. However, some are more impacted than others. As we noted in Section 3.3.2, each algorithms performs differently when given different types of measurement data. Although these differences indicate that the degradation rates will not be the same, it should be noted that eventually all algorithms degrade when participant count is large enough.

# Chapter 4

# Conclusion

Our work in this thesis is a step towards shoring up the science of security research. The goal is to shift from obscure qualitative statements which cannot be verified, to more quantitative analysis. While this is not always possible, it should be a design principle that we do as much quantitative analysis as is possible.

We cannot expect that most modern researchers will also become experts in statistical methods. This is equivalent to attempts at solving computer security problems by making all the users security experts. Since we want the systems we build to be useful to the widest audience possible, there is a clear need to have modern tools that can leverage the ubiquity of cheap computation while still maintaining a level of understand-ability that doesn't require years of statistical training.

Because measurements come in varying degrees of quality, decision problems have varying degrees of difficulty. However, at the beginning of a study it may not be apparent if the decision problem that is being solved is difficult or easy. There are many factors that can contribute to a problems difficulty. We have explored some issues that may lead to unexpected system behaviors such as poor scoring, or under sampled populations. However, there are many more potential problems that could be improved by leveraging computation power to perform exploratory analysis in an explainable way. Modern researchers need methods and tools that will give them answer to questions like:

- Is there enough information in the measurements to make the decision I've built my system around with some degree of certainty?

- Does the system I've built use all the information available in the measurements to ensure the most informed decision?

  - Going beyond model selection, am I loosing valuable information in other parts of my pipeline (e.g pre-processing that erroneously drops data as outliers, or feature encodings that are not faithful representations of the data)

- Are the reported metric values relevant to the problem being solved and trustable?

- Does my collected data cover the population I expect to use the proposed system?

- Under what conditions do the assumptions of my system break?

Security is, by it's very nature, a competition. Given the proliferation of computing devices, it is incorrect to expect that the users should shoulder any significant portion of the burden of ensuring that their devices take the most secure action in questionable situations. Systems should be built to protect the user from unintended consequences of the systems behavior. This can be achieved by considering the failure modes of a system during the design phase, and having sane failure behavior when the system is used outside the conditions it was designed for. To make these kinds of analysis possible, future researchers need to be armed with tools that make the failure modes of a system easily identifiable.

# Chapter 5

# Appendix

## 5.1 DataSet details

The following URLs were used to retrieve each dataset:

- Walking Activity - `https://archive.ics.uci.edu/ml/datasets/User+Identification+From+Walking+Activity`
- NBA Player Statistics - `https://www.kaggle.com/drgilermo/nba-players-stats`
- CAT Scan Localization - `https://www.kaggle.com/uciml/ct-slice-localization`
- Activity Recognition - `https://archive.ics.uci.edu/ml/datasets/Human+Activity+Recognition+Using+Smartphones`
- EEG Readings - `https://www.kaggle.com/broach/button-tone-sz`

### 5.1.1 Activity Recognition - A Common Mobile Platform Example

The initial purpose of this dataset was to determine the posture of a participant [125]. The authors recruited participants between the ages of 19 and 48 to provide an inertial measurement unit (IMU) data. The feature space consisted of time and frequency domain components read from a mobile phone based accelerometer and gyroscope. The full feature space has 561 features. This dataset is a typical example of the kinds of measurements that would be used to build a mobile platform identification system (e.g. gait recognition [39])

To process the dataset for use as an identification dataset, we merged the train and test sets that were provided with the ground truth labels, which we used as an additional feature. We then separated the user field from the measurements and used this as our class label. We then randomly selected 20 random labels and ran the multi-class classifiers.

The best confusion matrix for this dataset (see Figure 3.3) shows that all participants were easily distinguished by the random forest algorithm (black squares along the main diagonal of the matrix). This metric might lead us to believe that a system constructed from this data might be a viable identification system. Unfortunately, because the participant count is small, we do not have any idea how this system would perform when more participants were tested.

The feature list for this data is stored in a separate file names features.txt, there is a total of 561 features, all of which were used as the data. The user ID for each record was also stored in a separate file named subject_train.txt and subject_test.txt.

### 5.1.2 Walking Activity - Having a Small Number of Features Does Not Imply All Is Lost

The Walking Activity dataset was initially collected to perform gait authentication using two staged approaches. The dataset authors first inferred the posture from the data (e.g. walking or standing) and then perform a one class classification to determine if the posture readings correspond to the authorized user [126]. This dataset has only 4 features: time-step, x acceleration, y acceleration, z acceleration. There were a total of 22 participants who were all recruited by convenience methods.

To use this dataset to build an identification system, each of the individual participants readings was separated into individual files. To process this dataset we merged all of the files while applying a label corresponding to the file name/participant identifier. Since this data was time series, it would normally be the case that multiple samples would be analyzed as a group (e.g. within a time window of 5 seconds) to determine if they belonged to a particular participant. To see how far we could get with a naive approach, we treated each sample as a distinct measurement labeled by userID.

The random forest classifier had the best performance on this dataset, achieving $\approx 71\%$ accuracy (shown in Figure 3.3). While the identification performance of this dataset was not as good as the activity recognition dataset, it should be noted that this dataset is using significantly fewer features (561 $\rightarrow$ 4). Since this is time series data, by treating each sample as distinct, we are not taking advantage of the temporal correlations that exist between samples. We could potentially capture some of these correlations if we folded samples that span a fixed interval into higher dimension samples. This would lower the total sample size but increase the feature space dimension, which could capture some of the time information. Still, the naive approach does perform better than guessing, even though the temporal information is ignored. The results of this dataset demonstrate that even with a very low dimension feature space high performance can be achieved.

For the walking activity dataset all features were used. The id's served as labels.The feature list was:

"time-step", "x acceleration", "y acceleration", "z acceleration".

### 5.1.3 CAT Scan Localization - When Discernibility Is High, More Testing Is Required to Identify the Limits of the System

This dataset consisted of 384 features extracted from full body CT scan images which were used to localize CT slices [127]. Data from 97 participants was analyzed and histograms of the physiological features (bone structures and air inclusions) were extracted. This dataset did not have an existing label, however, each record was labeled with a patientID. To process this dataset, we separated the patientID for each measurement and used it as the label. Because this dataset had such a large participant count we ran the analysis with several different 20 participants subsets. The support vector machine performance was 100% for most subset chosen (see Figure 3.3) and the variation between subsets was $\leq 1\%$.

The discernibility of this feature space is incredibly high. We ran the random forest algorithm with a 90 participant subset and only saw a $\approx 1\%$ drop in accuracy. To ensure that the features were not leaking label information into the classifier (e.g. one of the features may have been equal to the patientID), we identified the maximum feature importance (as reported by the random forest algorithm) and then removed all features that had an importance within 50% of that maximum (8 features total). With the most important features removed, the random forest classifier still achieved $\approx$ 99% classification accuracy with a participant subset of 20, and $\approx$ 98% accuracy with a subset of 90. The high discernibility of this feature space is not surprising since the measurements are the result of an entire body scan in a room sized instrument. Because the discernibility of this feature space is very good, determining an upper-bound on number of distinct individuals this type of measurement could distinguish would require testing with a significantly larger number of participants.

All features were used from this dataset. The patient_id served as the label, and all other columns were used as the feature vector. To ensure that one of the values was not highly correlated with the patient_id, we took the top 5 features from the high accuracy random forest and removed them from the dataset. When this was done the performance values did not change much.

### 5.1.4 EEG Readings - Poorly Discernible Features Will Easily Fail But PCA Might Give You Insight into Why

In this dataset measurements from a head mount EEG instrument were collected. There was a total of 81 participants, several of these who were suffering from Schizophrenia, a chronic illness. The original authors used the measurements to determine if there was a correlation between the illness and certain patterns in the measurements [128].

The EEG dataset represents measurement data where all users look very similar. In Figure 3.3 we show the best achieved classification results. The best algorithm was random forest however, the difference between random forest and neural networks was not significant (see Table 3.3). As in the CT scan case, we selected several random subsets of the participant count, and re-ran the analysis several times. Each sampling produced approximately the same results (no greater than $\approx$ 3% variation).

Since the procedure and algorithms used for this dataset mirror the procedure and algorithms used for the CT scan dataset, why was the performance so different? One easy observation is that the size of the feature space is significantly smaller ($384 \rightarrow 40$). However, in the walking activity dataset, the feature space was significantly smaller

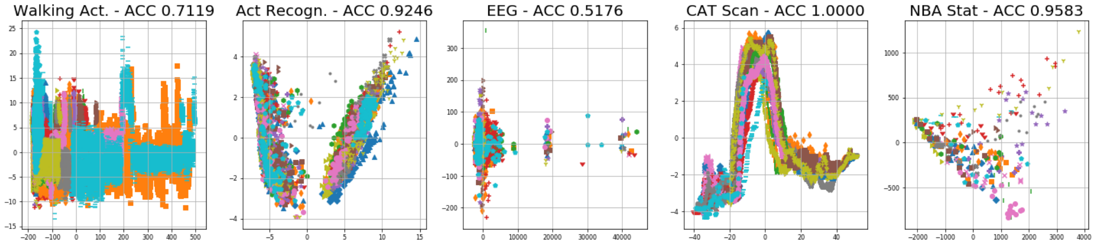


Figure 5.1: The first two principal components of the measurements for each dataset is shown. These components can be used to explain the differences in performance observed for each dataset. For the CT scan, activity recognition and NBA stats datasets, the high accuracy is coupled with significant separation in the measurements from each participant. Distinct clusters can be observed indicating that even the limited information captured in these two components is enough to distinguish may participants. The walking activity dataset begins to hint at why performance would degrade as there are clusters, but they fail to be distinct and have some significant overlap. The EEG dataset demonstrates the worst case where there is almost no separation. Most of the participants measurements are overlapping in one large area with only a small portion lying outside this giant component. All test set measurements from each of the 20 participants that were classified in Figure 3.3 are plotted. The measurements form each participant have a distinct colors and marker shapes.

than the activity recognition dataset, yet this system achieved ≈ 70% accuracy.

We can gain some insight into why the performance is so different by looking at a scatter plot of the first two principal components. In Figure 5.1, we show the first two principal components for each of the 20 participant subset that corresponds to the confusion matrix of Figure 3.3. As we can see, in the CT scan case, measurements from each participant form clusters and the overall shape are not uniformly distributed about the origin. In contrast, consider the EEG components where most of the measurements form one giant component sitting on top of the origin with other highly overlapping components to the right. This measurement dataset represents the worst case scenario for discernibility of the feature space. All participants produce measurements that vary significantly and the ranges over which they vary completely overlap. In this situation, it becomes very difficult to identify which measurement came from which participant.

For the EEG dataset we filtered out the following features:

u’trial’, u’condition’, u’ITI’, u’rejected’, u’Fz_N100’, u’FCz_N100’, u’Cz_N100’, u’FC3_N100’, u’FC4_N100’, u’C3_N100’, u’C4_N100’, u’CP3_N100’, u’CP4_N100’, u’Fz_P200’, u’FCz_P200’, u’Cz_P200’, u’FC3_P200’, u’FC4_P200’, u’C3_P200’, u’C4_P200’, u’CP3_P200’, u’CP4_P200’, u'Fz_B0', u'FCz_B0', u'Cz_B0', u'FC3_B0', u'FC4_B0', u'C3_B0', u'C4_B0', u'CP3_B0', u'CP4_B0', u'Fz_B1', u'FCz_B1', u'Cz_B1', u'FC3_B1', u'FC4_B1', u'C3_B1', u'C4_B1', u'CP3_B1', u'CP4_B1'

to eliminate some empty columns. The subject column was used as the label.

### 5.1.5 NBA Player Statistics - Large Numbers of Samples Per Participant Are Not Necessary If the Features Are Highly Discernible

The NBA Player statistics dataset is an artificial dataset constructed from player statistics spanning the years 1950 to 2017 [106]. Since the player careers tend to be shorter than 25 years, this dataset was explicitly constructed to have only a small number of samples per players on average. The stats of a player is a function of the players' capability and environmental circumstances of that year. Thus each measurement's features should be centered around a mean, but have random variation due to environmental factors (e.g. health or number of home games).

The dataset was processed to have a small number of partially distinct samples from each player, but a large number of players overall. The player statistics served as a measurement source with enough variation such that there should be some overlap between players. Each measurement of a player has the potential to overlap with another player's measurement. When this happens the classifier may confuse one player for another.

All players were given a unique numerical identifier, and each time-stamped playing statistic was treated as a single measurement. Several categorical values (E.g. team or position) were also encoded as a single integer value. Only players with at least 15 measurements were counted, thus the final dataset had measurements from 290+ players (there are very few datasets with such a large participant size).

We conducted the same analysis on this dataset as on all other previous datasets. Given the small measurement size and low density of measurements per participant, as compare to all other datasets, we might expect much worse performance. The classification performance, however, is high because the each player has values that are very distinct. The team and position alone provide significant clustering into distinct groups. These groups are then refined further by playing characteristics. In Figure 5.1 we see the first two principal components of all the samples in the test set for the 20 players shown in the CM of Figure 3.3. Even though the point density is very low ($\approx 15$ per player), the clustering of each players samples is very tight. These two features alone provide significant discernibility between players.

For the NBA player statics, all names were encoded as numeric identifiers including the player name and team name. The full feature list was:

u'Year', u'Player', u'Pos', u'Age', u'Tm', u'G', u'GS',u'MP', u'PER', u'TS%', u'3PAr', u'FTr', u'ORB%', u'DRB%', u'TRB%',u'AST%', u'STL%', u'BLK%', u'TOV%', u'USG%', u'blanl', u'OWS', u'DWS',u'WS', u'WS/48', u'blank2', u'OBPM', u'DBPM', u'BPM', u'VORP', u'FG',u'FGA', u'FG%', u'3P', u'3PA', u'3P%', u'2P', u'2PA', u'2P%', u'eFG%',u'FT', u'FTA', u'FT%', u'ORB', u'DRB', u'TRB', u'AST', u'STL', u'BLK',u'TOV', u'PF', u'PTS'

Where player was the player name which served as the label to be predicted.

## 5.2 Parameters used for each algorithm on each dataset

Table 5.1 enumerates all the parameters used for each dataset. In most cases the defaults were sufficient. For the case of EEG we used a random parameter search to try to improve results for the Random Forest and Support Vector Machine algorithms, but there was no significant gain in performance. The default neural network has a single hidden layer with 100 neurons. The default kernel for support vector machines was the RBF kernel and it makes multi-class decisions via one-vs-one run offs.

Table 5.1: All parameter arguments used for every algorithm, dataset pair

| | Act. Recogn. | Walking Act. | CT Scan | EEG | NBA Stat. |
|---|---|---|---|---|---|
| RandomForestClassifier | n_estimators =1000<br>max_depth =20 | n_estimators =1000<br>max_depth =20 | n_estimators =1000<br>max_depth =20 | n_estimators =1000<br>max_depth =20 | n_estimators =1000<br>max_depth =20 |
| svm.SVC | default | default | default | default | default |
| MLPClassifier | default | default | default | default | default |

# References

[1] John Donne. Meditation xvii devotions upon emergent occasions, 1624.

[2] Robert Gray Gallager. *Stochastic Processes, Theory for Applications*, chapter 8, pages 1–1000. Cambridge University Press, Cambridge, UK, 1st edition, 2013.

[3] Steven M Kay. *Fundamentals of statistical signal processing: Detection theory, vol. 2*, chapter 3, pages 61–65. Prentice Hall Upper Saddle River, NJ, USA:, 1998.

[4] Michael E. Schuckers. *Computational Methods in Biometric Authentication.* Springer London, 2010.

[5] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An introduction to statistical learning*, volume 112. Springer, 2013.

[6] Yaser Said Abu-Mostafa. *Learning From Data*, chapter Training versus Testing, pages 39–69. AMLbook.com, Pasadena, CA, 1st edition, 2012.

[7] Fabio Silva, Marcos Suassuna, Csar Frana, Alicia M. Grubb, Tatiana Gouveia, Cleviton Monteiro, and Igor Ebrahim dos Santos. Replication of empirical studies in software engineering research: A systematic mapping study. *Empirical Software Engineering*, 19, 09 2012.

[8] Fiona Fidler and John Wilcox. Reproducibility of scientific results. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy.* Metaphysics Research Lab, Stanford University, winter 2018 edition, 2018.

[9] Aurlien Gron. *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems.* O'Reilly Media, Inc., 1st edition, 2017.

[10] D Anderson and K Burnham. Model selection and multi-model inference. *Second. NY: Springer-Verlag*, 63(2020):10, 2004.

[11] Bradley Efron and Trevor Hastie. *Computer age statistical inference*, volume 5. Cambridge University Press, 2016.

[12] Jian Liu, Chen Wang, Yingying Chen, and Nitesh Saxena. Vibwrite: Towards finger-input authentication on ubiquitous surfaces via physical vibration. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 73–87. ACM, 2017.

[13] Ivo Sluganovic, Marc Roeschlin, Kasper B. Rasmussen, and Ivan Martinovic. Using reflexive eye movements for fast challenge-response authentication. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, pages 1056–1067, New York, NY, USA, 2016. ACM.

[14] Linghan Zhang, Sheng Tan, and Jie Yang. Hearing your voice is not enough: An articulatory gesture based liveness detection for voice authentication. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, CCS '17, pages 57–71, New York, NY, USA, 2017. ACM.

[15] Sauvik Das, Gierad Laput, Chris Harrison, and Jason I. Hong. Thumprint: Socially-inclusive local group authentication through shared secret knocks. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 3764–3774, New York, NY, USA, 2017. ACM.

[16] Can Liu, Gradeigh D. Clark, and Janne Lindqvist. Where usability and security go hand-in-hand: Robust gesture-based authentication for mobile systems. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 374–386, New York, NY, USA, 2017. ACM.

[17] Munehiko Sato, Rohan S. Puri, Alex Olwal, Yosuke Ushigome, Lukas Franciszkiewicz, Deepak Chandra, Ivan Poupyrev, and Ramesh Raskar. Zensei: Embedded, multi-electrode bioimpedance sensing for implicit, ubiquitous user recognition. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 3972–3985, New York, NY, USA, 2017. ACM.

[18] Stefan Schneegass, Youssef Oualil, and Andreas Bulling. Skullconduct: Biometric user identification on eyewear computers using bone conduction through the skull. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 1379–1384, New York, NY, USA, 2016. ACM.

[19] Max T. Curran, Nick Merrill, John Chuang, and Swapan Gandhi. One-step, three-factor authentication in a single earpiece. In *Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers*, UbiComp '17, pages 21–24, New York, NY, USA, 2017. ACM.

[20] Hidehito Gomi, Shuji Yamaguchi, Kota Tsubouchi, and Naomi Sasaya. Towards authentication using multi-modal online activities. In *Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers*, UbiComp '17, pages 37–40, New York, NY, USA, 2017. ACM.

[21] Chenyu Huang, Huangxun Chen, Lin Yang, and Qian Zhang. Breathlive: Liveness detection for heart sound authentication with deep breathing. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 2(1):12:1–12:25, March 2018.

[22] Rui Liu, Cory Cornelius, Reza Rawassizadeh, Ronald Peterson, and David Kotz. Vocal resonance: Using internal body voice for wearable authentication. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 2(1):19:1–19:23, March 2018.

[23] Jiayao Tan, Xiaoliang Wang, Cam-Tu Nguyen, and Yu Shi. Silentkey: A new authentication framework through ultrasonic-based lip reading. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 2(1):36:1–36:18, March 2018.

[24] Y. Chen, J. Sun, X. Jin, T. Li, R. Zhang, and Y. Zhang. Your face your heart: Secure mobile face authentication with photoplethysmograms. In *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, pages 1–9, May 2017.

[25] Li Lu, Jiadi Yu, Yingying Chen, Hongbo Liu, Yanmin Zhu, Yunfei Liu, and Minglu Li. Lippass: Lip reading-based user authentication on smartphones leveraging acoustic signals. In *IEEE INFOCOM 2018 - The 37th Annual IEEE International Conference on Computer Communications*, April 2018.

[26] C. Song, A. Wang, K. Ren, and W. Xu. Eyeveri: A secure and usable approach for smartphone user authentication. In *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, pages 1–9, April 2016.

[27] Y. Yang and J. Sun. Energy-efficient w-layer for behavior-based implicit authentication on mobile devices. In *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, pages 1–9, May 2017.

[28] S. Yi, Z. Qin, E. Novak, Y. Yin, and Q. Li. Glassgesture: Exploring head gesture interface of smart glasses. In *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, pages 1–9, April 2016.

[29] Huan Feng, Kassem Fawaz, and Kang G. Shin. Continuous authentication for voice assistants. In *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking*, MobiCom '17, pages 343–355, New York, NY, USA, 2017. ACM.

[30] Feng Lin, Chen Song, Yan Zhuang, Wenyao Xu, Changzhi Li, and Kui Ren. Cardiac scan: A non-contact and continuous heart-based user authentication system. In *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking*, MobiCom '17, pages 315–328, New York, NY, USA, 2017. ACM.

[31] Jagmohan Chauhan, Yining Hu, Suranga Seneviratne, Archan Misra, Aruna Seneviratne, and Youngki Lee. Breathprint: Breathing acoustics-based user authentication. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*, pages 278–291. ACM, 2017.

[32] Feng Lin, Kun Woo Cho, Chen Song, Wenyao Xu, and Zhanpeng Jin. Brain password: A secure and truly cancelable brain biometrics for smart headwear. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '18, pages 296–309, New York, NY, USA, 2018. ACM.

[33] Claude Castelluccia, Markus Dürmuth, Maximilian Golla, and Fatma Deniz. Towards implicit visual memory-based authentication. In *Network and Distributed System Security Symposium (NDSS 2017)*, 2017.

[34] David Freeman, Sakshi Jain, Markus Dürmuth, Battista Biggio, and Giorgio Giacinto. Who are you? a statistical approach to measuring user authenticity. In *Network and Distributed Systems Security (NDSS) Symposium 2016*, pages 1–15, 2016.

[35] Erkam Uzun, Simon Pak Ho Chung, Irfan Essa, and Wenke Lee. rtcaptcha: A real-time captcha based liveness detection system. In *Network and Distributed Systems Security (NDSS) Symposium 2018*, 2018.

[36] Weitao Xu, Guohao Lan, Qi Lin, Sara Khalifa, Neil Bergmann, Mahbub Hassan, and Wen Hu. Keh-gait: Towards a mobile healthcare user authentication system by kinetic energy harvesting. In *Network and Distributed Systems Security (NDSS) Symposium 2017*, 2017.

[37] Ishan Bhardwaj, Narendra D. Londhe, and Sunil K. Kopparapu. A spoof resistant multibiometric system based on the physiological and behavioral characteristics of fingerprint. *Pattern Recognition*, 62:214 – 224, 2017.

[38] Feng Cheng, Shi-Lin Wang, and Alan Wee-Chung Liew. Visual speaker authentication with random prompt texts by a dual-task cnn framework. *Pattern Recognition*, 83:340 – 352, 2018.

[39] Matteo Gadaleta and Michele Rossi. Idnet: Smartphone-based gait recognition with convolutional neural networks. *Pattern Recognition*, 74:25 – 37, 2018.

[40] Jiacang Ho and Dae-Ki Kang. Mini-batch bagging and attribute ranking for accurate user authentication in keystroke dynamics. *Pattern Recognition*, 70:139 – 151, 2017.

[41] Manabu Okawa. Synergy of foregroundbackground images for feature extraction: Offline signature verification using fisher vector with fused kaze features. *Pattern Recognition*, 79:480 – 489, 2018.

[42] R. Raghavendra, Kiran B. Raja, Sushma Venkatesh, and Christoph Busch. Improved ear verification after surgery - an approach based on collaborative representation of locally competitive features. *Pattern Recognition*, 83:416 – 429, 2018.

[43] Lin Zhang, Lida Li, Anqi Yang, Ying Shen, and Meng Yang. Towards contactless palmprint recognition: A novel device, a new benchmark, and a collaborative representation based identification approach. *Pattern Recognition*, 69:199 – 212, 2017.

[44] Yunpeng Song, Zhongmin Cai, and Zhi-Li Zhang. Multi-touch authentication using hand geometry and behavioral information. In *Security and Privacy (SS&P), 2017 IEEE Symposium on*, pages 357–372. IEEE, 2017.

[45] Heather Crawford and Ebad Ahmadzadeh. Authentication on the go: Assessing the effect of movement on mobile device keystroke dynamics. In *Thirteenth Symposium on Usable Privacy and Security (SOUPS 2017)*, pages 163–173, Santa Clara, CA, 2017. USENIX Association.

[46] Katharina Krombholz, Thomas Hupperich, and Thorsten Holz. Use the force: Evaluating force-sensitive authentication for mobile devices. In *Twelfth Symposium on Usable Privacy and Security (SOUPS 2016)*, pages 207–219, Denver, CO, 2016. USENIX Association.

[47] M. Frank, R. Biedert, E. Ma, I. Martinovic, and D. Song. Touchalytics: On the applicability of touchscreen input as a behavioral biometric for continuous authentication. *IEEE Transactions on Information Forensics and Security*, 8(1):136–148, Jan 2013.

[48] Alisher Kholmatov and Berrin Yanikoglu. Identity authentication using improved online signature verification method. *Pattern Recogn. Lett.*, 26(15):2400–2408, November 2005.

[49] Simon Eberz, Kasper B. Rasmussen, Vincent Lenders, and Ivan Martinovic. Evaluating behavioral biometrics for continuous authentication: Challenges and metrics. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, ASIA CCS '17, pages 386–399, New York, NY, USA, 2017. ACM.

[50] Joseph Bonneau, Cormac Herley, Paul C. van Oorschot, and Frank Stajano. The quest to replace passwords: A framework for comparative evaluation of web authentication schemes. In *2012 IEEE Symposium on Security and Privacy*, pages 553–567, May 2012.

[51] John Oglesby. What's in a number? moving beyond the equal error rate. *Speech Communication*, 17(1):193–208, 1995.

[52] Foster J. Provost, Tom Fawcett, and Ron Kohavi. The case against accuracy estimation for comparing induction algorithms. In *Proceedings of the Fifteenth International Conference on Machine Learning*, ICML '98, pages 445–453, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc.

[53] Tom Fawcett. An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874, 2006.

[54] Mark H. Zweig and Gregory Campbell. Receiver-operating characteristic (roc) plots: a fundamental evaluation tool in clinical medicine. *Clinical Chemistry*, 39(4):561–577, 1993.

[55] Christopher D. Brown and Herbert T. Davis. Receiver operating characteristics curves and related decision measures: A tutorial. *Chemometrics and Intelligent Laboratory Systems*, 80(1):24 – 38, 2006.

[56] David L Streiner and John Cairney. What's under the roc? an introduction to receiver operating characteristics curves. *The Canadian Journal of Psychiatry*, 52(2):121–128, 2007. PMID: 17375868.

[57] James A Hanley and Barbara J McNeil. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143(1):29–36, 1982.

[58] Luis D. Berrizbeitia. Receiver operating characteristic (roc) curves - https://kennis-research.shinyapps.io/roc-curves/. applet, Aug 2016.

[59] David J Hand and Robert J Till. A simple generalisation of the area under the roc curve for multiple class classification problems. *Machine learning*, 45(2):171–186, 2001.

[60] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.

[61] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[62] Dit-Yan Yeung, Hong Chang, Yimin Xiong, Susan George, Ramanujan Kashi, Takashi Matsumoto, and Gerhard Rigoll. Svc2004: First international signature verification competition. In David Zhang and Anil K. Jain, editors, *Biometric Authentication*, pages 16–22, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

[63] K. S. Killourhy and R. A. Maxion. Comparing anomaly-detection algorithms for keystroke dynamics. In *2009 IEEE/IFIP International Conference on Dependable Systems Networks*, pages 125–134, June 2009.

[64] Amazon. http://amazon.com/go, May 2019.

[65] Paul Resnick and Hal R Varian. Recommender systems. *Communications of the ACM*, 40(3):56–59, 1997.

[66] FaceBook. https://www.facebook.com/business/ads, May 2019.

[67] Twitter. https://ads.twitter.com/, May 2019.

[68] Netflix. https://help.netflix.com/en/node/100639, May 2019.

[69] Amazon. https://www.amazon.com/, May 2019.

[70] Spotify. https://www.spotify.com/, May 2019.

[71] Sauvik Das, Gierad Laput, Chris Harrison, and Jason I. Hong. Thumprint: Socially-inclusive local group authentication through shared secret knocks. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 3764–3774, New York, NY, USA, 2017. ACM.

[72] Muhammad Shahzad and Shaohu Zhang. Augmenting user identification with wifi based gesture recognition. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 2(3):134:1–134:27, September 2018.

[73] Xiang Zhang, Lina Yao, Salil S. Kanhere, Yunhao Liu, Tao Gu, and Kaixuan Chen. Mindid: Person identification from brain waves through attention-based recurrent neural network. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 2(3):149:1–149:23, September 2018.

[74] Anna Huang, Dong Wang, Run Zhao, and Qian Zhang. Au-id: Automatic user identification and authentication through the motions captured from sequential human activities using rfid. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 3(2):48:1–48:26, June 2019.

[75] Wei Wang, Alex X. Liu, and Muhammad Shahzad. Gait recognition using wifi signals. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '16, pages 363–373, New York, NY, USA, 2016. ACM.

[76] Xiao Wang, Tong Yu, Ming Zeng, and Patrick Tague. Xrec: Behavior-based user recognition across mobile devices. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 1(3):111:1–111:26, September 2017.

[77] Yuanying Chen, Wei Dong, Yi Gao, Xue Liu, and Tao Gu. Rapid: A multimodal and device-free approach using noise estimation for robust person identification. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 1(3):41:1–41:27, September 2017.

[78] Stefan Schneegass, Youssef Oualil, and Andreas Bulling. Skullconduct: Biometric user identification on eyewear computers using bone conduction through the skull. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 1379–1384, New York, NY, USA, 2016. ACM.

[79] Ken Pfeuffer, Matthias J. Geiger, Sarah Prange, Lukas Mecke, Daniel Buschek, and Florian Alt. Behavioural biometrics in vr: Identifying people from body motion and relations in virtual reality. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, pages 110:1–110:12, New York, NY, USA, 2019. ACM.

[80] Munehiko Sato, Rohan S. Puri, Alex Olwal, Yosuke Ushigome, Lukas Franciszkiewicz, Deepak Chandra, Ivan Poupyrev, and Ramesh Raskar. Zensei: Embedded, multi-electrode bioimpedance sensing for implicit, ubiquitous user recognition. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 3972–3985, New York, NY, USA, 2017. ACM.

[81] Tobias Grosse-Puppendahl, Xavier Dellangnol, Christian Hatzfeld, Biying Fu, Mario Kupnik, Arjan Kuijper, Matthias R. Hastall, James Scott, and Marco Gruteser. Platypus: Indoor localization and identification through sensing of electric potential changes in human bodies. In *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '16, pages 17–30, New York, NY, USA, 2016. ACM.

[82] Yongsen Ma, Gang Zhou, Shuangquan Wang, Hongyang Zhao, and Woosub Jung. Signfi: Sign language recognition using wifi. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 2(1):23:1–23:21, March 2018.

[83] Namrata Srivastava, Joshua Newn, and Eduardo Velloso. Combining low and mid-level gaze features for desktop activity recognition. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 2(4):189:1–189:27, December 2018.

[84] Yanwen Wang and Yuanqing Zheng. Modeling rfid signal reflection for contact-free activity recognition. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 2(4):193:1–193:22, December 2018.

[85] Zijie Zhu, Xuewei Wang, Aakaash Kapoor, Zhichao Zhang, Tingrui Pan, and Zhou Yu. Eis: A wearable device for epidermal american sign language recognition. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 2(4):202:1–202:22, December 2018.

[86] Raghav H. Venkatnarayan and Muhammad Shahzad. Gesture recognition using ambient light. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 2(1):40:1–40:28, March 2018.

[87] Cheng Zhang, Anandghan Waghmare, Pranav Kundra, Yiming Pu, Scott Gilliland, Thomas Ploetz, Thad E. Starner, Omer T. Inan, and Gregory D. Abowd. Fingersound: Recognizing unistroke thumb gestures using a ring. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 1(3):120:1–120:19, September 2017.

[88] Cheng Zhang, Qiuyue Xue, Anandghan Waghmare, Ruichen Meng, Sumeet Jain, Yizeng Han, Xinyu Li, Kenneth Cunefare, Thomas Ploetz, Thad Starner, Omer Inan, and Gregory D. Abowd. Fingerping: Recognizing fine-grained hand poses using active acoustic on-body sensing. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, pages 437:1–437:10, New York, NY, USA, 2018. ACM.

[89] Hongyi Wen, Julian Ramos Rojas, and Anind K. Dey. Serendipity: Finger gesture recognition using an off-the-shelf smartwatch. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 3847–3851, New York, NY, USA, 2016. ACM.

[90] Maximilian Schrapel, Max-Ludwig Stadler, and Michael Rohs. Pentelligence: Combining pen tip motion and writing sounds for handwritten digit recognition. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, pages 131:1–131:11, New York, NY, USA, 2018. ACM.

[91] Jess McIntosh, Asier Marzo, Mike Fraser, and Carol Phillips. Echoflex: Hand gesture recognition using ultrasound imaging. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 1923–1934, New York, NY, USA, 2017. ACM.

[92] Toshiyuki Ando, Yuki Kubo, Buntarou Shizuki, and Shin Takahashi. Canalsense: Face-related movement recognition system based on sensing air pressure in ear canals. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, UIST '17, pages 679–689, New York, NY, USA, 2017. ACM.

[93] Jun Gong, Yang Zhang, Xia Zhou, and Xing-Dong Yang. Pyro: Thumb-tip gesture recognition using pyroelectric infrared sensing. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, UIST '17, pages 553–563, New York, NY, USA, 2017. ACM.

[94] Jess McIntosh, Asier Marzo, and Mike Fraser. Sensir: Detecting hand gestures with a wearable bracelet using infrared transmission and reflection. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, UIST '17, pages 593–597, New York, NY, USA, 2017. ACM.

[95] Frederic Kerber, Michael Puhl, and Antonio Krüger. User-independent real-time hand gesture recognition based on surface electromyography. In *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '17, pages 36:1–36:7, New York, NY, USA, 2017. ACM.

[96] Saiwen Wang, Jie Song, Jaime Lien, Ivan Poupyrev, and Otmar Hilliges. Interacting with soli: Exploring fine-grained dynamic gesture recognition in the radio-frequency spectrum. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, UIST '16, pages 851–860, New York, NY, USA, 2016. ACM.

[97] Chuyu Wang, Jian Liu, Yingying Chen, Hongbo Liu, Lei Xie, Wei Wang, Bingbing He, and Sanglu Lu. Multi-touch in the air: Device-free finger tracking and gesture recognition via cots rfid. In *Proc. of IEEE INFOCOM*, 2018.

[98] Shridatt Sugrim, Can Liu, Meghan McLean, and Janne Lindqvist. Robust performance metrics for authentication systems. In *Network and Distributed Systems Security (NDSS) Symposium 2019*, 2019.

[99] Robert V Krejcie and Daryle W Morgan. Determining sample size for research activities. *Educational and psychological measurement*, 30(3):607–610, 1970.

[100] Margarete Sandelowski. Sample size in qualitative research. *Research in nursing & health*, 18(2):179–183, 1995.

[101] Deepak Vasisht, Anubhav Jain, Chen-Yu Hsu, Zachary Kabelac, and Dina Katabi. Duet: Estimating user position and identity in smart homes using intermittent and incomplete rf-data. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 2(2):84:1–84:21, July 2018.

[102] Wenjie Ruan, Quan Z. Sheng, Lei Yang, Tao Gu, Peipei Xu, and Longfei Shangguan. Audiogest: Enabling fine-grained hand gesture detection by decoding echo signal. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '16, pages 474–485, New York, NY, USA, 2016. ACM.

[103] Kun Qian, Chenshu Wu, Zimu Zhou, Yue Zheng, Zheng Yang, and Yunhao Liu. Inferring motion direction using commodity wi-fi for interactive exergames. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 1961–1972, New York, NY, USA, 2017. ACM.

[104] Raghav H. Venkatnarayan, Griffin Page, and Muhammad Shahzad. Multi-user gesture recognition using wifi. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '18, pages 401–413, New York, NY, USA, 2018. ACM.

[105] Dua Dheeru and Efi Karra Taniskidou. UCI machine learning repository, 2017.

[106] Omri Goldstein. Nba players stats since 1950, 2017.

[107] N/A. Data.gov, 2018.

[108] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[109] Aditya Virmani and Muhammad Shahzad. Position and orientation agnostic gesture recognition using wifi. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '17, pages 252–264, New York, NY, USA, 2017. ACM.

[110] Tianming Zhao, Jian Liu, Yan Wang, Hongbo Liu, and Yingying Chen. Ppg-based finger-level gesture recognition leveraging wearables. In *Proc. of IEEE INFOCOM*, 2018.

[111] Xiang 'Anthony' Chen and Yang Li. Bootstrapping user-defined body tapping recognition with offline-learned probabilistic representation. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, UIST '16, pages 359–364, New York, NY, USA, 2016. ACM.

[112] Peter Bruce and Andrew Bruce. *Practical Statistics for Data Scientists: 50 Essential Concepts.* " O'Reilly Media, Inc.", 2017.

[113] Anne Chao and Tsung-Jen Shen. Nonparametric estimation of shannons index of diversity when there are unseen species in sample. *Environmental and ecological statistics*, 10(4):429–443, 2003.

[114] Edward H Simpson. Measurement of diversity. *Nature*, 163(4148):688, 1949.

[115] Kelly Caine. Local standards for sample size at chi. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 981–992, New York, NY, USA, 2016. ACM.

[116] Andrew John Anderson and Algis Jonas Vingrys. Small samples: Does size matter? *Investigative Ophthalmology and Visual Science*, 42(7):1411, 2001.

[117] Robert C MacCallum, Michael W Browne, and Hazuki M Sugawara. Power analysis and determination of sample size for covariance structure modeling. *Psychological methods*, 1(2):130, 1996.

[118] Katherine S Button, John PA Ioannidis, Claire Mokrysz, Brian A Nosek, Jonathan Flint, Emma SJ Robinson, and Marcus R Munafò. Power failure: why small sample size undermines the reliability of neuroscience. *Nature Reviews Neuroscience*, 14(5):365, 2013.

[119] João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. A survey on concept drift adaptation. *ACM Comput. Surv.*, 46(4):44:1–44:37, March 2014.

[120] Robert Tibshirani Trevor Hastie and Jerome H. Friedman. The elements of statistical learning new york. *NY: Springer*, pages 115–163, 2001.

[121] Benoît Rittaud and Albrecht Heeffer. The pigeonhole principle, two centuries before dirichlet. *The Mathematical Intelligencer*, 36(2):27–29, 2014.

[122] John M Lachin. Introduction to sample size determination and power analysis for clinical trials. *Controlled clinical trials*, 2(2):93–113, 1981.

[123] S. J. Raudys and A. K. Jain. Small sample size effects in statistical pattern recognition: recommendations for practitioners. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(3):252–264, March 1991.

[124] A. Hackshaw. Small studies: strengths and limitations. *European Respiratory Journal*, 32(5):1141–1143, 2008.

[125] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, and Jorge Luis Reyes-Ortiz. A public domain dataset for human activity recognition using smartphones. In *ESANN*, 2013.

[126] Pierluigi Casale, Oriol Pujol, and Petia Radeva. Personalization and user verification in wearable systems using biometric walking patterns. *Personal and Ubiquitous Computing*, 16(5):563–580, 2012.

[127] Franz Graf, Hans-Peter Kriegel, Matthias Schubert, Sebastian Pölsterl, and Alexander Cavallaro. 2d image registration in ct images using radial image descriptors. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 607–614. Springer, 2011.

[128] Judith M Ford, Vanessa A Palzes, Brian J Roach, and Daniel H Mathalon. Did i do that? abnormal predictive processes in schizophrenia when button pressing to deliver a tone. *Schizophrenia bulletin*, 40(4):804–812, 2013.