Graduate Theses, Dissertations, and Problem Reports

2017

AFIS Based Likelihood Ratios for Latent Fingerprint Comparisons

Shreya Sateesh Kamath

Follow this and additional works at: https://researchrepository.wvu.edu/etd

AFIS based likelihood ratios for latent fingerprint comparisons

Shreya Sateesh Kamath

Thesis submitted
to the Eberly College of Arts and Sciences
at West Virginia University

in partial fulfillment of the requirements for the degree of

Master of Science in
Forensic & Investigative Science

Keith Morris, Ph.D., Chair

Jacqueline Speir, Ph.D.

Afzel Noore, Ph.D.

Department of Forensic & Investigative Science

Morgantown, West Virginia
2016

Keywords: AFIS

ABSTRACT

AFIS based likelihood ratios for latent fingerprint comparisons

Shreya Sateesh Kamath

Latent fingerprints are one of the most common pieces of evidence found on a crime scene and represent accidental or unintentional prints collected as part of a criminal investigation. They are caused when the friction ridge skin comes in contact with a surface, and thus requires the use of chemical processing to be visualized with the naked eye. The comparison and identification of fingerprints depends on various factors such as the substrate quality, surface, duration, environmental factors and examiner experience. These factors can result in reduced clarity or content, and can even cause distortions as compared to a fingerprint taken under controlled conditions. Since the release of the National Academy of Sciences (NAS) report in 2009, the field of fingerprint analysis has come under much scrutiny. Specifically, the need for more research into the determination of the accuracy and reliability of the identifications made by fingerprint examiners has been raised.

One such method used for the comparison of latent fingerprints to known prints is through an Automated Fingerprint Identification System (AFIS). The AFIS used in this research was the AFIX Tracker® where five variables were assessed: match score, match minutiae, match status, delta match score and marked minutiae, to determine which variable(s) was a better indicator of a true match. Bayesian networks were then constructed to compute the likelihood ratios to evaluate the dependency of the variables on one another, where the performance of the likelihood ratios in determining the identity of the unknown latent was assessed using Tippett and ECE plots. Receiver Operating Characteristic (ROC) curves and Bayesian networks were constructed to perform statistical analysis of the matches obtained while comparing a latent print to a ten-print card. A combination of Tippett and Empirical Cross Entropy (ECE) plots were used to assess the performance of the AFIX Tracker® in classifying unknown prints. It was observed that a match minutiae of 15 or higher resulted in a 100% true match result whereas for the non-matches, no more than 13 match minutiae were found. Moreover, the delta match scores difference between the matches and non-matches were notable (delta score of 0.1-153 for matches compared to a score of 0-0.1 for the non-matches). Overall, it was determined that approximately 87% of the time a randomly selected known match would have a higher number of match minutiae as compared to a non-match.

Contents

List of Figures

1. Introduction

1.1   Latent Fingerprints

One of the most common pieces at evidence found on crime scenes are fingerprints. They are often found in two forms: latent and patent. Latent prints are prints that are not easily visible and need to be processed whereas patent prints are prints that are visible to the naked eye. These prints are collected by investigators and then entered into an Automated Fingerprint Identification System (AFIS) or are manually compared to a ten-print card. Both the AFIS and manual comparison processes use various characteristics (*Figure 1.1*) to determine the similarity between two prints.
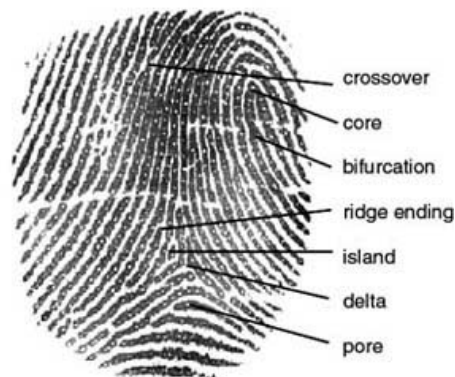
Figure 1.1: Various types of minutiae marked on a fingerprint [1]

Latent fingerprints represent accidental or unintentional prints and are created by the friction ridge skin deposition on a surface. The prints require physical, chemical, digital or optical processing to enable visualization. The quality and quantity of minutiae are highly variable depending on various factors such as method of lift, substrate, surface, environmental factors and duration between deposition and collection. These factors can result in reduced clarity, content, and distortions compared to a fingerprint collected under controlled conditions. This reduction in detail in the latent print compared to the actual patterns of ridges and grooves of a finger, may result in high error rates during the

identification of the print. [5] Conversly, known prints typically have larger, cleaner and richer information (*Figure 1.2*, right). [6]

Figure 1.2: Image on the left is a latent print. It contains large areas where the print is smudged or unclear compared to the known print (on the right) which contains clear ridge details

Latent print examiners make comparisons between distorted and smudged prints against prints of better quality. In the past, fingerprint identifications, if made by an experienced examiner, were claimed to be infallible. [7] The infamous Madrid bombing case shed light on the likelihood of a misidentification of latent prints by examiners. The issue of fingerprint identification being infallible was addressed in the 2009 National Academy of Sciences (NAS) report which states:

> "... other forensic disciplines – such as fingerprints ... no studies have been conducted of large populations to determine how many sources might share the same or similar feature" and "studies should accumulate data on how much a person's fingerprints vary from impression to impression, as well as the degree to which fingerprints vary across a population. With this kind of research, examiners could begin to attach confidence limits to conclusions about whether a print is linked to a particular person." [8]

The report further stated:

> "... one study found that fingerprint examiners did not always agree even with their own past conclusions when the same evidence was presented in a different context." [8]

Currently, latent fingerprint examiners utilize the Analysis, Comparison, Evaluation and Verification (ACE-V) approach to make one of three following decisions [9]:

i) Individualization / Identification - the two friction ridge impressions contain sufficient quality (clarity) and quantity of agreement in friction ridge detail. The latent print examiner determines that two friction ridge impressions originated from the same source, to the exclusion of all others.

ii) Exclusion (Non-Identification) - the two friction ridge impressions contain sufficient quality (clarity) and quantity of friction ridge detail which is not in agreement where the latent print examiner determines that two friction ridge impressions originated from different sources.

iii) Inconclusive - a latent print examiner, trained to competency, is unable to individualize or exclude the source of an impression.

In the ACE-V process, the examiner first analyzes the print to determine if the print is suitable for comparison. In the comparison step, a side-by-side observation of the friction ride impression is made to determine if the two impressions are in agreement. The evaluation step is the formulation of a conclusion (identification, exclusion or inconclusive) based on the analysis and comparison steps. Finally, the verification step includes an independent examination of the latent prints by a second qualified examiner. The NAS report, along with the errors made in the Madrid bombing case has brought much criticism to the forensic field pertaining to fingerprint examination leading to numerous Daubert challenges in court regarding the accuracy of the decision made. [10]

1.2   Automated Fingerprint Identification System

The Integrated Automated Fingerprint Identification System (IAFIS) - now superceeded by the Next Generation Identification System (NGI) - is a national automated system maintained by the FBI that contains a database of all registered fingerprints (latent and ten-prints). It houses the fingerprints and criminal histories on more than 70 million subjects in the criminal master file, along with more than 34 million civil prints. [11]

Generically, an automated fingerprint identification system (AFIS) is used by many police and other agencies for the identification of individuals. There are numerous vendors that supply such systems to law enforcement.

The latent prints collected by investigators are manually entered into the AFIS and minutiae are marked. The prints are then run against the entire database and a candidate list is generated with a corresponding match score. The higher the score, the better the probability that the candidate matches the latent. However, it is often observed that the highest match score does not always belong to the true match. This could be a result of the size of the database or the quality of the latent. A larger database has a higher chance of containing the suspect than a smaller database. Furthermore, if the quality of

the latent is poor, an examiner would be unable to mark a large number of minutiae on the print, which could result in the a lower match score.

The AFIS follows a two class minutiae classification system: terminations and bifurcations which are used to determine the match score of a fingerprint. In the two class model, for each minutia *(P)*, a score value *(s)* along with its membership class *(t)*, angle *(θ)* and corresponding co-ordinates are determined and stored. [12] For each minutia (on the latent and known print), the score ranges between 0-100 indicating the reliability of the minutia. The reliability is assigned based on ridge flow, breaks between ridges and valleys and noise. A partial Point Set Pattern Matching (PSPM) algorithm is used to determine significance of a minutia. The algorithm assigns a score (weight) to each minutia, based on the resemblance between the latent and the known print. The significance of the minutia is directly proportional to its score. For example, if two minutiae P_1 and P_2 have a score of s_1 and s_2 respectively, where if $s_1 < s_2$, then the value of P_1 is less reliable than that of P_2. [12]

For the purpose of this study, a match is defined as *a latent and a known fingerprint having the same characteristic minutiae without any unexplained differences* and non-match is defined as *two fingerprints that do not share the same characteristics which leads to the conclusion of the prints belonging to different sources*. The AFIS only provides a list of candidate exemplars; comparison decisions must be made by a latent print examiner. Due to its automated process, the AFIS has a capability to search through millions of prints at a time. An examiner, performing a manual comparison on millions of prints will not be as efficient as the AFIS system. [13,14] AFIS is a cognitive technological tool that has the ability to store a large amount of information. It can perform more comparisons than its human counterpart, hence acting as an offloading tool. [15] The examiner's cognitive load decreases as he/she is able to transfer some of the tasks to the system, increasing efficiency and decreasing error. While AFIS is a convenient tool for fingerprint comparisons, a complete lights out approach (where there is no examiner involvement) cannot be adopted. The presence of highly distorted or noisy prints renders AFIS's ability to mark minutia ineffective, requiring an expert to manually mark the minutia. [13] As the database size increases, the error rates (particularity the false positive rates) increase. [6] The reason for this is as a database grows, an AFIS searching that database is increasingly likely to find more close non-matches, (prints that are highly similar to the latent, but are in fact from a different individual; also termed "look-alikes"). [6] Furthermore, as the database size increases, the chances of finding a true match also increases, but this can also make the task of the human examiner more demanding and, potentially, more error-prone as the known prints are more similar to one another. [6,13] This research tackled the issue of determining a true match / non-match and the accuracy of the AFIS system.

1.3 Purpose

The purpose of this study was to perform a statistical analysis on the matches obtained while comparing a latent print to a ten-print sample using the AFIS. In this research

study, data generated by the AFIS system, such as the match score, match minutiae, match status, and delta match score were assessed by varying the number of minutiae marked (3-50) on the latent to determine which factor(s) was a better indicator of a true match. Bayesian networks were then constructed to compute the likelihood ratios, where the performance of the likelihood ratios in determining the identity of the unknown latent was assessed using Tippett and ECE plots.

2. Previous Research

2.1 Bayesian network

A Bayesian network (a probabilistic directed acyclic graph), is a powerful probabilistic representations of data. [16] By defining a finite set of variables as nodes, and the relationship between the nodes as edges, a Bayesian network is constructed and thus the relationship between the nodes can be assessed.
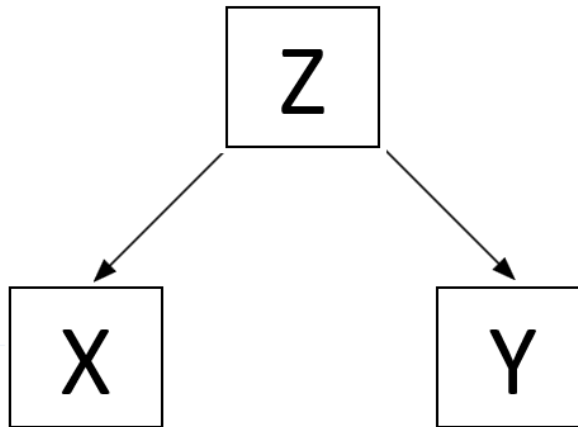
Figure 2.1: A simple Bayesian network showing interaction between nodes X, Y, and Z

The nodes represent variables and the edges determine the influence between those variables. An example of the dependency between the nodes is shown in *Figure 2.1* where Z influences whether or not X and Y take place. Both X and Y are independent of each other due to an absence of an edge linking them. The edges show a conditional dependency between nodes, X, Y, and Z. [17] Each node can assume a value of a certain number of states. When a state is selected, the node is instantiated.

The network was designed based on Bayes' theorem, which states:

$$\underbrace{\frac{p(H_p|E)}{p(H_d|E)}}_{\text{posterior odds}} = \underbrace{\frac{p(E|H_p)}{p(E|H_d)}}_{\text{likelihood ratio}} \cdot \underbrace{\frac{p(H_p)}{p(H_d)}}_{\text{prior odds}} \tag{2.1}$$

From *Equation 2.1*, the evidential value of evidence E, as given in the likelihood ratio, is quantified by the ratio of the probabilities of observing E given two hypotheses. In the case of latent fingerprint comparison, the evidence will consist of some version of the similarity between the latent and known fingerprint, and the hypotheses are, for example, of the form: [18]

> H_p: the prosecutorial hypothesis that states the latent print from the scene and the known fingerprint originate from the same donor (the suspect).

> H_d: the defense hypothesis which states that the latent fingerprint and the known fingerprint originate from different donors.

The strength of the evidence in relation to one of the propositions is the probability of the evidence given one hypothesis, divided by the probability of the evidence given the alternative hypothesis. The prior odds and the posterior odds are assessed by attorneys, police and the jury, whereas the likelihood ratio (LR) is assessed by a criminalist. From *Equation 2.1*, an LR is defined as as the probability of occurrence of the evidence given the suspect is the donor over the probability of occurrence of the evidence given someone other than the suspect is the donor. The LR value (as shown in *Figure 2.2*) ranges from zero to infinity.

1,000,000	Extremely strong	
100,000	Very strong	
10,000	Strong	
1000	Moderately strong	support for H_p
100	Moderate	
10	Limited	
1	Inconclusive	
0.1	Limited	
0.01	Moderate	
0.001	Moderately strong	
0.0001	Strong	support for H_d
0.00001	Very strong	
0.000001	Extremely strong	
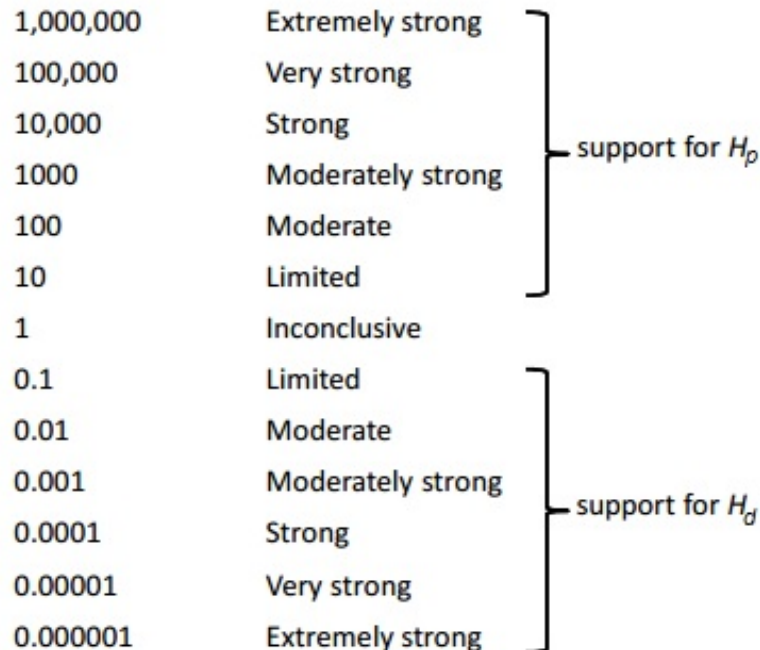
Figure 2.2: Likelihood ratio scale depicting the strength of the evidence [2]

The LR provides a means of assessing the posterior probabilities from the prior odds, since it is the posterior odds in which the court is interested in. An LR > 1 supports the prosecutorial hypothesis whereas an LR < 1 supports the defense hypothesis.

2.2 Fingerprints and the Bayesian network

The AFIS system is widely used to obtain fingerprint comparison results by producing a candidate list. However, it is not often used for a statistical analysis of latent print and known print evidence. [19] In a study conducted by Champod *et al.* (2014), an AFIS was used to study and evaluate fingerprint evidence and test the variability in scores obtained from a known donor versus fingerprints from a different source. [20] An LR based approach was used to assess between-finger and within-finger variability, enabling the addition of transparency in determining assessments made by fingerprint examiners while marking characteristics in fingerprints. The LR was calculated based on the scores obtained from the AFIS system. The numerator, $f(s|H)$, was the probability density of the observed score *(s)* if the latent and known print originated from the same source *(H)*, where the source was considered as the suspect's finger and the origin of the within-finger variability. The within-fingerprint variability was determined by comparing the latent and known prints with one another and the AFIS scores for these comparisons were obtained. The denominator, $f(s|\bar{H})$, represented the likelihood of between-finger variability by observing the evidential score if the latent and known prints did not come from the same source. Furthermore, determination of a minimal database size was also considered to be crucial as it ensured stability so that the score estimator does not differ by more than one order of magnitude in regions where observations were made (regions where the minutiae from the latent matched the minutiae marked on the ten-print card). Two latent prints: one marked with six minutiae and the other yielding ten minutiae were subjected to the database to extract scores based on sample size and the percentiles for each of the scores were measured. A weighted distribution was established, where fingers having a certain classification were assigned a value (in percentage) based on how commonly they occur. The weighted distribution of the identification based on the finger number showed a tendency to associate the latent with thumbs, index or middle finger rather than the little or ring fingers. The highest probability was assigned to fingers that most possibly left the latent and lowest probability to the finger that was thought to be the most improbable. The results from Champod *et al.* (2014) show that reliable between-finger variability is feasible with as few as 10,000 scores.

While reliable between-finger variability can be obtained when analyzing a small set of samples, differences were observed when analyzing between-finger variability. While adding a minutiae did lead to an increase in the match score, it does not cause a notable increase. This was because between-finger variability scores were dependent on the configuration of the minutiae marked: closer the minutiae were to one another, better was the score. [21] The number of minutiae that can be marked on the latent depends on the quality of the latent as that plays an important role in how many minutiae an examiner

can find. The more the number of minutiae marked and the closer they are to one another, better are the chances of obtaining a high match score as a higher number of matching minutiae can be found. Before being analyzed, latents have to undergo a screening process by examiners who often disregard prints of low quality or those that are considered to not have enough details to make a definite examination. As a low quality latent is disregarded, the evidentiary value of the print is unknown due to no analysis being performed on it. A study conducted by Neumann *et al.* (2011) analyzed the evidentiary value of a low quality latent. In the study, a latent, belonging to an individual whose identity was known, was quantified using a statistical model where the corresponding minutiae between the latent and known prints were labelled using the Universal Latent Workstation (ULW). [22] The ULW, a single encoding software improves the quality of latents that are classified as a low quality print by examiners, by translating it into a compatible format recognized by the IAFIS. [23] The weight of the evidence was computed using a likelihood ratio, based on two hypothesis [22]:

i) H_p: the latent belonged to the person (suspect) who made the known print.

ii) H_d: the latent belonged to someone other than the suspect.

In the latents identified by the examiners, the LRs reported were consistently high, thus supporting H_p. It was also found that upon using ULW and the IAFIS, there was a 20% increase in the number of items of evidence that could be reported with an identification conclusion that would ordinarily have been discarded during the analysis phase of the examination.

While improving the quality of the latent can potentially provide better results regarding the identity of the print and result in a higher identification rate, Biedermann *et al.* (2011) [24] put forward two propositions concerning the evidentiary value of results obtained through database searches.

i) Evaluation of DNA evidence based on results obtained from database pointing towards the suspect.

ii) Effect of excluding individuals in a database.

Fimmers *et al.* (2011) [25] stated that a hypothesis must be formulated without using any information from the data to be analyzed. This was then disputed by Biedermann *et al.* (2011) where they raise the question asking if the evidence E confirms hypothesis H. By proposing the Bayes theorem (*Equation 2.2*) it is clearly stated that if E is known at the time H is proposed, E is part of the background information such that *p(E)-p(E|H)=1* (Biedermann *et al.* (2011)). Furthermore, *p(H|E) - p(H)* shows that E does not affect the probability of H. This clearly explains that as evidence E is independent of H, the proposition that the suspect is the source of the crime stain is not a result of the DNA match as this proposition existed even before the match was found.

$$p(H|E) = \frac{p(E|H) * p(H)}{p(E)} \tag{2.2}$$

In the second proposition, Fimmers *et al.*(2011) stated:

> "...recommend in a case in which, after finding exactly one match in a database of size 630.000 ..., the RMP (Random Match Probability) with 5 analyzed STR loci is in the magnitude of 1 in 600.000? It is in our view inconceivable to assume, in such a situation, that the evidential value due to the database match is higher than without a database search. Rather, it seems that the contrary is of significance." [26]

Biedermann *et al.* (2011) furthermore disputed this statement made by Fimmers *et al.* (2011). Since the database size is huge, it is highly likely to find a match within the database compared to the inverse of the rarity of the compared characteristic. Biedermann *et al.* (2011) also argue that as there can only be one match to a particular fingerprint, once this match is determined, there should be less observed correspondence between other prints in the database. The statement by Fimmers *et al.* (2011) also stresses the probability of obtaining a match by chance. This statement is disregard by Biedermann *et al.* (2011) who claim this to be highly unlikely, stating that there are other parameters that are considered before making a match between the prints. It is irrelevant to know whether or not the suspect matches. The other donors in the database are automatically excluded if no notable match characteristics are found. The presence or absence of the suspect in the database does not affect the non-match criteria of the database itself. This research study will assess the theory put forward by Biedermann *et al.* (2011) by assessing the data extracted from the AFIX Tracker® to determine which factors influence a match or a non-match result.

Since the release of the NAS report, the field of fingerprint analysis has faced numerous challenges when trying to determine the evidentiary value of a latent print. [27] Neumann *et al.* (2011) used the LR approach to explore the configurations of three minutiae on latent and known prints to gather knowledge of within- and between-source variability. [28] Latent prints often contain partial and distorted characteristics and have a poor S/N (Signal to Noise) ratio. By utilizing the spatial configuration of three minutiae, the position with respect to surface and pattern can be determined. The fingerprints were initially processed using a Gabor filtering process - an edge detecting filter that assists in feature extraction [14]. The minutiae were then marked on the latent and known prints and the Delaunay triangulation method was applied where each fingerprint was viewed as a collection of triangles. The Euclidian distance *(d)* was used to measure the distance between the the minutiae. The triangles were then grouped into family sets (triangles originating from the same finger and same set of minutiae). Depending on the level of distortion obtained on each finger, the number of triangles drawn between each set of three minutiae per family set varied between 1 and 54. The LR computation was based both on the numerator (within-source) and denominator (between-source) under the propositions of s (the print belonged to the suspect) and $\bar{s}$ (print belonged to someone else). The results show that the R^2 value for the comparison of triangles under defense

proposition $(\bar{s})$ was equal to 0.974 while under prosecutorial proposition (s) was equal to 0.945, showing the robust nature of the approach. The magnitude of the LRs determined under the prosecutorial and defense hypothesis show that configurations of as little as three minutiae can offer major evidentiary value. [22]

3. Experiment

3.1 Methods

The prints used in this study were acquired from the Biometric Collection of People 2008 (BioCoP) fingerprint database developed at West Virginia University. For the purpose of this study, only ten-print cards were used and no palm prints were analyzed. Nine hundred and seventy ten-print cards that were collected from participants were entered into the AFIS (for the purpose of this study, AFIX Tracker® will be used) as known prints. The BioCoP database also consisted of latent prints, collected from the same ten-print card donors, that were developed using ninhydrin (on paper), cyanoacrylate fuming (on a compact disk) and black powder (on glass).

The BioCoP database was combined with another database "CSI1", also developed at West Virginia University, increasing the AFIX Tracker® database to a total of 1861 ten-print cards. The BioCoP contained a total of 7431 latent fingerprints out of which 200 latent prints were selected. A 70-30 approach was followed where 70% (140 prints) were entered into the AFIX Tracker® as known prints (i.e. the identity of the print was known (also referred to as the training set)). The remaining 30% (60 prints) were entered as unknown prints (i.e. the identity of these prints were not known). All prints were obtained through random selection (using R and RStudio®). The random selection conducted on the unknown prints was performed by an individual not part of this study and were re-named to ensure anonymity of the prints prior to analysis. The 970 ten-print cards from the BioCoP database, along with the 200 latent prints collected through random selection were entered into AFIX Tracker® as .bmp files at 500 pixels per inch resolution. The remaining 891 ten-prints from the CSI1 database were entered at 600 pixels per inch resolution. As the ten-prints were not individually scanned into the AFIX Tracker®, their resolution could not be altered.
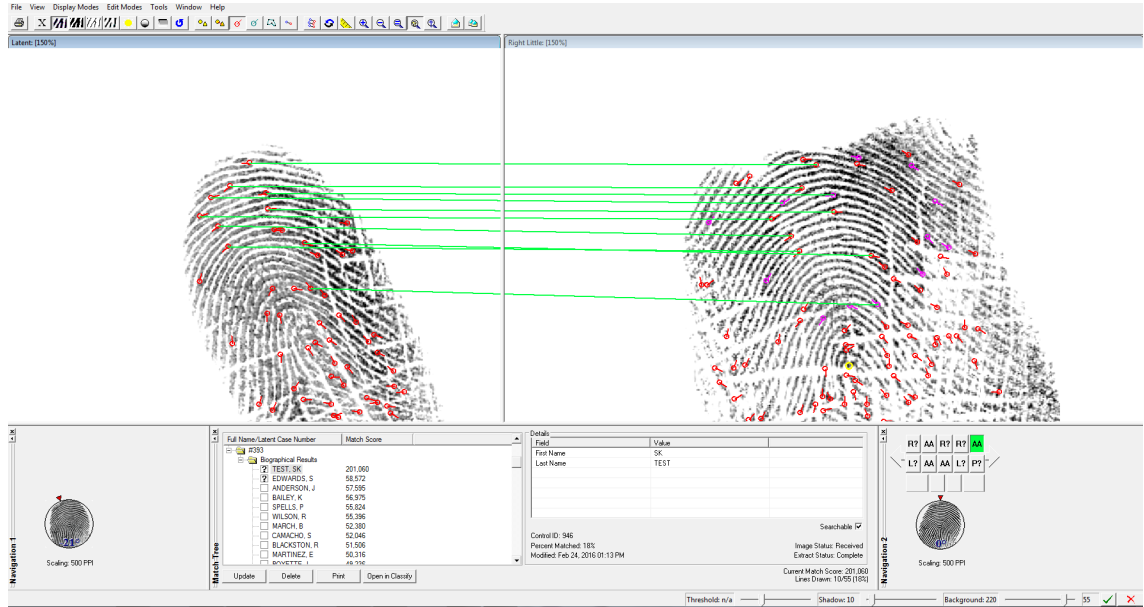
Figure 3.1: Example of results shown by the AFIX Tracker®. The match score, candidate list, control ID and the match minutiae are displayed

Figure 3.1 depicts the search results shown by the AFIX Tracker®. The match score and number of matched minutiae are given by the system. After entry of the latents and known ten-prints, the minutiae on each were marked. For the ten-print cards, the minutiae were manually marked and auto extraction was not utilized. Quality assessment of the prints was not a factor in this research study and due to random selection, there was a variation observed in the number of minutiae marked (for example some 10 print cards contained over 70 minutiae on each print, while some had as low as 10 minutiae marked). Minutiae on each latent were marked ranging from 3 to 50 (3, 5, 7, 10, 15, ... 50). For every range of minutiae marked, the latent was run against the database and the top 100 candidate list was generated. For example, latent A was marked with three minutiae. The latent was then run against the AFIX Tracker® database and its candidate list was generated. Next, the same latent A was marked with the two more minutiae, bringing its count to five and run against the database. This process was repeated until a maximum of 50 minutiae on each latent were marked. For each candidate list generated, control ID, finger number, match minutiae, percent match, match score, match status and total number of minutiae marked (marked minutiae) were obtained. The match minutiae is the total number of minutiae that were matched between the latent and the ten-print card and was calculated using *Equation 3.1* where the match minutiae was rounded up to the

next highest integer.

$$Match\ Minutiae = \frac{percent\ match * minutiae\ marked}{100} \tag{3.1}$$

3.2 Data Analysis

Bayesian networks, along with ROC curves, Equal Error Rate, Tippett and ECE plots were constructed based on variables such as match score, match minutiae, match status, delta match score and marked minutiae. R and RStudio® was used to construct ROC curves and the plots.

3.2.1 ROC Curve

The receiver operating characteristic (ROC) curve is a plot of test sensitivity as the y-coordinate or the true positive rate (TPR) against the specificity as the x-coordinate or the false positive rate (FPR). The sensitivity measures the probability that the latent and known prints are characterized as being a match, given it is a true match. Converesly, the specificity measures the probability that the latent and known prints are characterized as being a match, given they are a non-match. TPR and FPR are widely used in determining the efficiency of a test. [29] The ROC curve assists in providing an evaluation of the diagnostic ability of a test to be able to discriminate between the true states of the test. [5] One of the most common qauntitative measures is to calculate the area under the curve (AUC) which is a combination of sensitivity and specificity. The AUC measures the overall performance of the test and can take any value between 0-1. The closer the AUC is to 1, the more accurate is the test's diagnostics and the better the method's ability to distinguish between the states. An AUC of 0.5 signifies an inconclusive result while AUC's less than 0.5 indicates incorrect classification. [29]

By utilizing the ROC curve, an analysis of the AFIS algorithm can be derived to observe the degree of separation between the true positive and false positive rates.

3.2.2 Tippett Plots

Tippett plots are a valuable tool for assessing the performance of an LR-based method for evidence evaluation that empirically represents the cumulative propotions of LR in a dataset depending on which hypotheis is true (H_p or H_d). [30] These plots provide a graphical representation of the magnitude increments of the LR along with the number of minutiae marked and match score configuration.
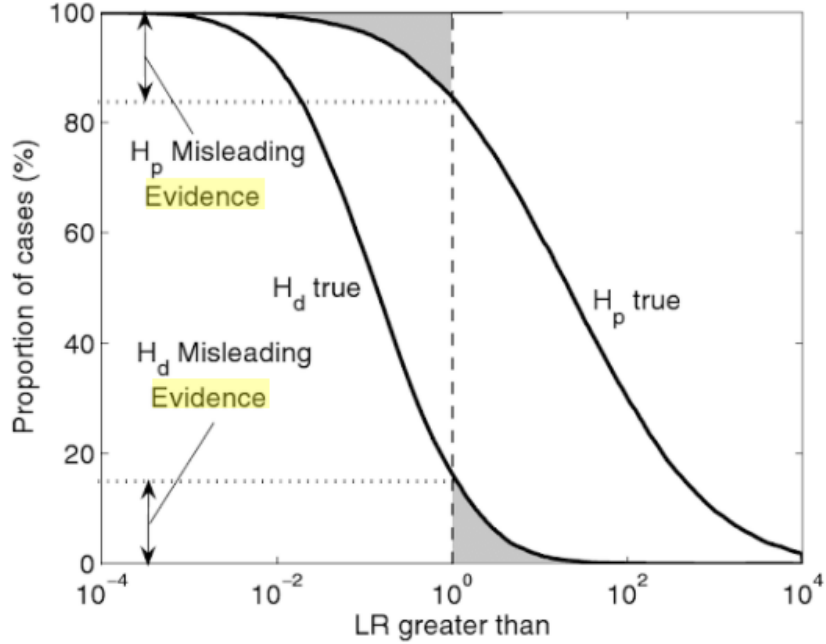
Figure 3.2: A Tippett plot showing the LR distributions with the rates of misleading evidence [3]

An accuracte system should be able to differentiate the true matches (where LR > 1) from the non-matches (LR < 1). Tippett plots also assist in studying the accuracy of a system by comparing the proportion of misleading evidence/mis-classification (as seen in *Figure 3.2*), where the LR values support the wrong hypothesis (LR > 1 when H_d is true and LR < 1 when H_p is true). [3] For the purpose of this project, the threshold is fixed at LR = 1.

3.2.3 Empirical Cross-Entropy

An Emperical Cross-Entropy (ECE) plot provides a measure of accuracy and calibration of a system. It represents the loss of information due to uncertainty in the proposition at each prior. [31]
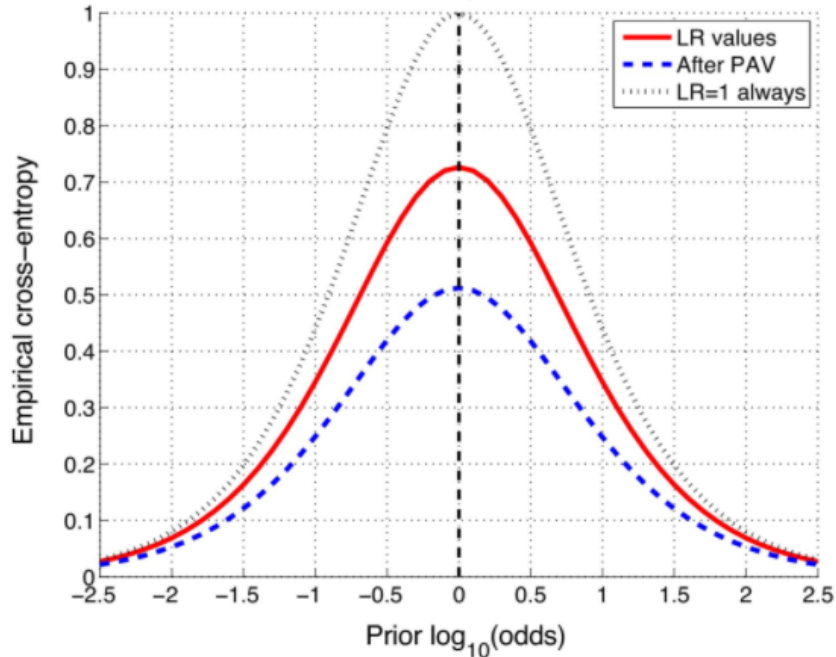
Figure 3.3: A reference ECE plot [4]

ECE plots provide a measure of accuracy and calibration of a system and is weighted by the prior odds (on the x-axis). *Figure 3.3* shows the average uncertainity (measured in entropy) when applying the LR reported by a system. [32] Using R and Rnetica (refer to *Appendix A*), an ECE plot was generated when the LR at a given prior was calculated. The grey curve represents the LR at 1 which shows the system is unable to provide any information regarding the identity of the unknown latent print. The red curve represents the LR of the test sample. The higher this curve, the more information is needed to correctly classify the print. The dotted blue line is the calibration accuracy for when the LR are perfectly calibrated. This calibration is done using Pool Adjacent Violators or PAV. The PAV determines the best convex function for the LR values when all the violators are removed (where all the misleading evidence is removed). This curve represents the system's optimal performance. The lower the dotted blue line, the better the discriminating power. Ideally, there should be little to no difference between the dotted blue line and the red line. The lower the red and blue curves and the smaller the difference between the two shows that the system is calibrated well and can accurately determine how a given unknown is being classified. Furthermore, the closer the experimental LR is to 0 (lower is the ECE value), better is the accuracy.

4. Results

4.1 Preliminary Study on Orientation of the Print

A preliminary analysis on a ten-print card was performed. In order to provide a match, AFIX Tracker® changes the orientation of the print to the most suitable position. A ten-print card was entered into the AFIX Tracker® containing 891 ten-print cards and was marked as a known print. A plain impression from the same ten-print card was taken where only left thumb impression was used. The left thumb impression was entered as a latent into AFIX Tracker® thirty six times where print was rotated in increments of 5° up to ±180°. Forty two minutiae at the same positions were marked on each of the latent. The latents were then run twice: with the orientation locked and without locked orientation. The orientation lock determines whether or not AFIX Tracker® rotates the latent print during the search to provide the best possible match result.
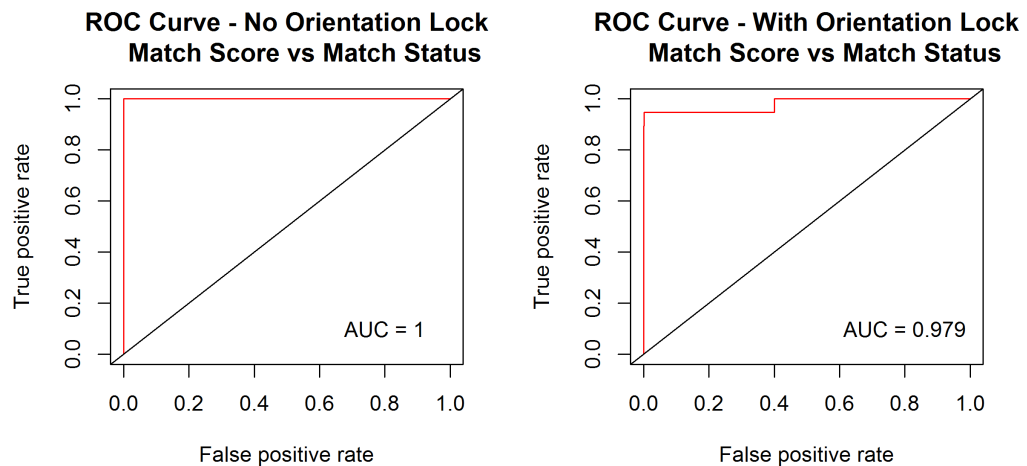
(a) Match Score vs Match Status without orientation lock

(b) Match Score vs Match Status with orientation lock

Figure 4.1: ROC curve of latents searched against database

Figure 4.1 shows the ROC curve constructed based on the data obtained from AFIX

Tracker® where the latents were run against the entire database. The ROC curve of match score vs match status was constructed for both with and without orientation lock searches. Match score signifies the score determined by AFIX Tracker® to assign a value to the fingerprint match. Match status signifies whether or not the latent print matched the rolled impressions and match minutiae signifies the number of minutiae matched between the known and latent prints.

With the orientation lock in place, AFIX Tracker® rotated the print to a maximum of $\pm 15°$. An AUC of 0.979 was determined, which signifies an excellent separation between the true matches and non-matches. It was observed that the match score for the true matches were all higher than the match score for the non-matches, yielding a perfect separation between the two groups. With the orientation lock removed, AFIX Tracker® rotated the print to the best determined fit and an AUC of 1 was determined. While the no orientation lock did have a higher AUC than the with orientation lock method, there was no significant differences observed(DeLong [33], p=0.32 at 95% confidence). Using this preliminary study, it can be determined that using either method does not lead to a significant difference in the match score results. For the purpose of this study, all 200 latents were marked without utilizing the orientation lock feature in AFIX Tracker®.

4.2 Preliminary Study on Clustering vs Unclustering of Marked Minutiae

As mentioned in the methods section, each latent was marked with minutiae ranging from 3-50. A preliminary study was conducted to assess whether marking the minutiae closer to one another has an effect on the match score and match minutiae as compared to when the minutiae are marked further apart. A high quality latent was obtained (latent containing more than 60 minutiae). The latent was first marked with 3-50 minutiae where all the minutiae were close to one another (clustered). The procedure was repeated a second time on the same latent, where the minutiae were marked further apart (unclustered). AFIX Tracker® searches were then conducted on both sets (clustered and unclustered) and it was determined that when the minutiae were marked closer to one another, a higher match score and higher match minutiae was obtained as compared to unclustered prints (*Figure 4.2*, only true match results are shown).

Table 4.1: Clustered vs. unclustered data for true matches. Shown are the match score results for the unclustered prints ranging from 30-50 minutiae marked as no results were obtained when less than 30 minutiae were marked

Status	Min Marked	Match Score	Match Min
Clustered	50	1222572	34
Clustered	45	1259230	35
Clustered	40	1271872	34
Clustered	35	1194699	33
Clustered	30	857493	27
Clustered	25	576366	21
Clustered	20	211176	18
Clustered	15	157220	14
Clustered	10	92772	9
Clustered	7	4124	4
Unclustered	30	1167	3
Unclustered	35	33108	7
Unclustered	40	48008	7
Unclustered	45	52372	9
Unclustered	50	174488	17

Table 4.1 displays the true match results obtained when the latent was marked using the clustered vs. the unclustered marking method, where Min Marked referes to the number of minutiae marked on the latent andMatch Min referes to the number of minutiae matched between the latent and the known print. No true match results were obtained for the unclustered prints when less than 30 minutiae are marked. The clustered prints, however, show true match results when as low as 7 minutiae are marked. Based on this data, minutiae on all 200 latent prints in this study were marked using the clustering method. As crime scene prints often only have small areas that are suitable for analysis, the clustering method was used to mirror the smaller areas obtained from these prints.

Clustered vs. Unclustered Minutiae
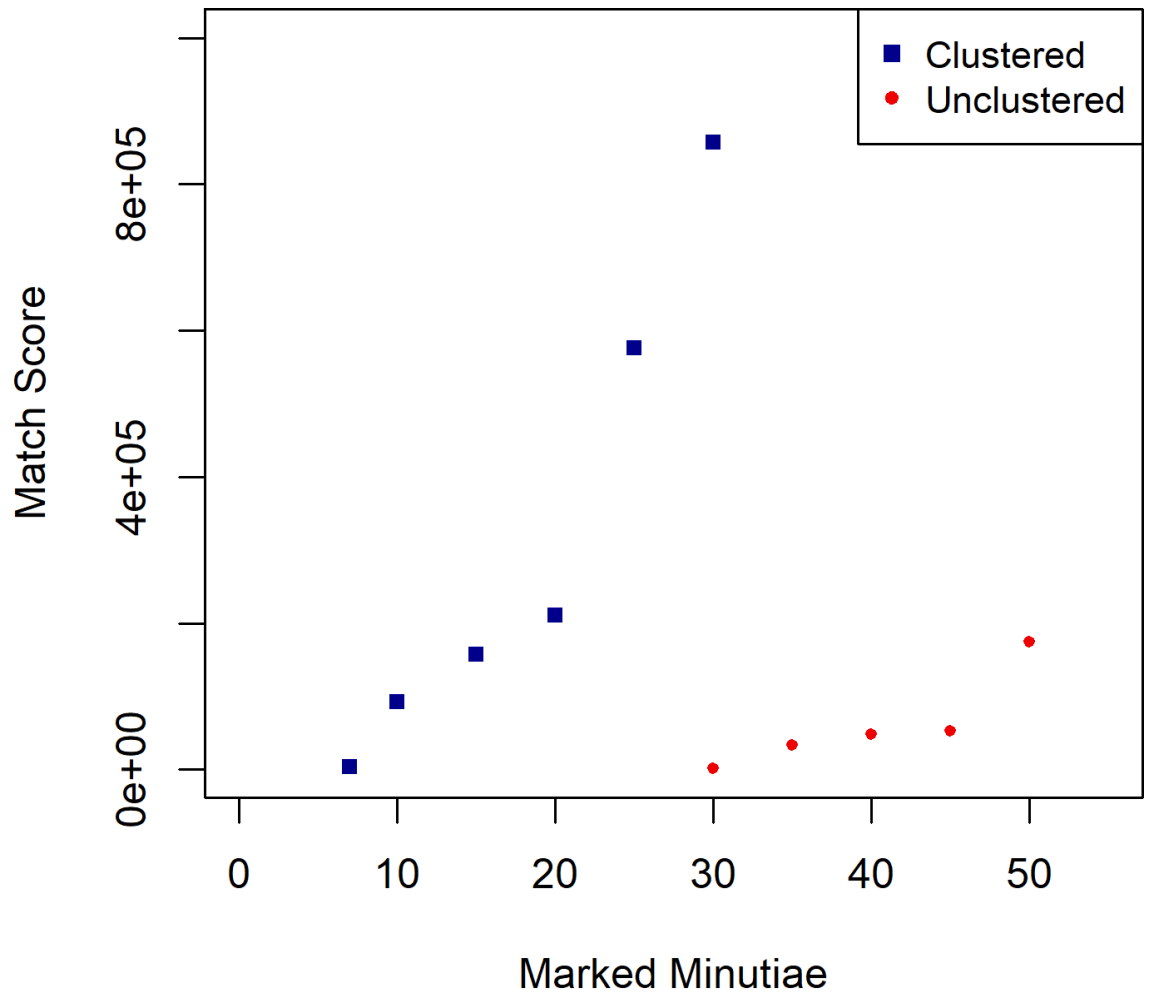
Figure 4.2: Scatter plot clustering vs unclustering

4.3 Auto Extract and Manual Extract

This research study utilized two different databases: BioCoP and CSI1. All the ten-print cards obtained from the BioCop database were at 500ppi where the minutiae were manually marked. Whereas the CSI1 database consisted of prints scanned in at 600ppi where all the minutiae were auto extracted using the auto extract feature provided by AFIX Tracker®. In order to assess if the resolution/extraction method had any significant impact on the results obtained, a study was conducted where a latent, marked with minutiae ranging from 3-50, was subjected to two different database comparisons: one that consisted of only the BioCoP prints and the second that contained both the BioCoP and CSI1 prints (which was a combination of two different resolutions and extraction methods).
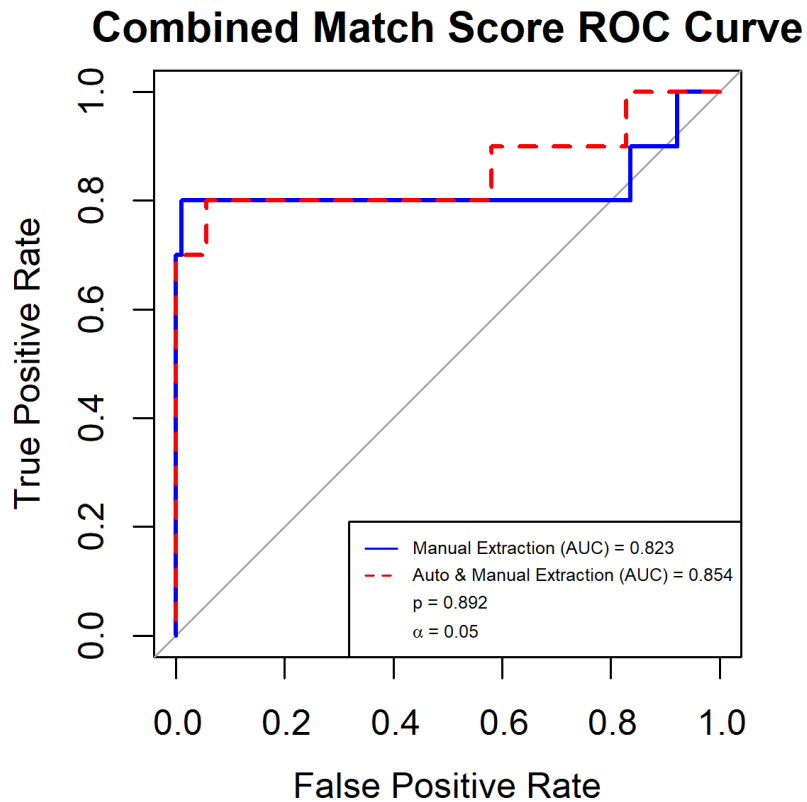
Figure 4.3: ROC curve of a latent marked using manual extraction and a combination of manual and auto extraction

Upon running the AFIX Tracker®, ROC curves were constructed using match score

to determine if significant differences were obtained. As seen in *Figure 4.3*, no significant differences were observed between the manual extraction and the combined extraction method while assessing the match score (DeLong [33], p=0.84 at 95% CI). As the p-value determined using the DeLong [33] method is higher than 0.05, it can be concluded that the auto extract and combined extraction method are comparable to one another. Hence, for this research, both the BioCoP and CSI1 databases were combined. However, a drawback of this study was that the latents were marked by one examiner only. Multiple examiners conducting the procedure could obtain different results.

4.3.1 AFIX Tracker® Search Results for the Known Latents

The search results from AFIX Tracker® were evaluated where a total of 58724 comparisons were obtained, with 624 results returned as matching prints or true matches and 58100 non-match results, where for each latent run against the database, the top 100 candidate list was generated.

Figure 4.4 shows the matches made by the system based on the marked and match minutiae. By observing the trend for the true match status (matches), it can be seen that the number of match minutiae increases as the number of marked minutiae increases. This positive correlation is not evident for the false match status (non-matches). Based on the matches and non-matches, a frequency of occurrence for a true vs a false match can be determined at a particular input value. For instance, for the true matches, at 50 marked minutiae, the corresponding marked minutiae goes as high as 32 whereas for the non-matches, at 50 marked minutiae, the corresponding match minutiae does not reach beyond 13 minutiae.

Figure 4.4: Scatter Plot of Marked Minutiae vs Match Minutiae, where the number of marked minutiae ranges from 3 to 50 (3, 5, 7, 10, 15, ... 50)

Matched Minutiae vs Log of Match Score
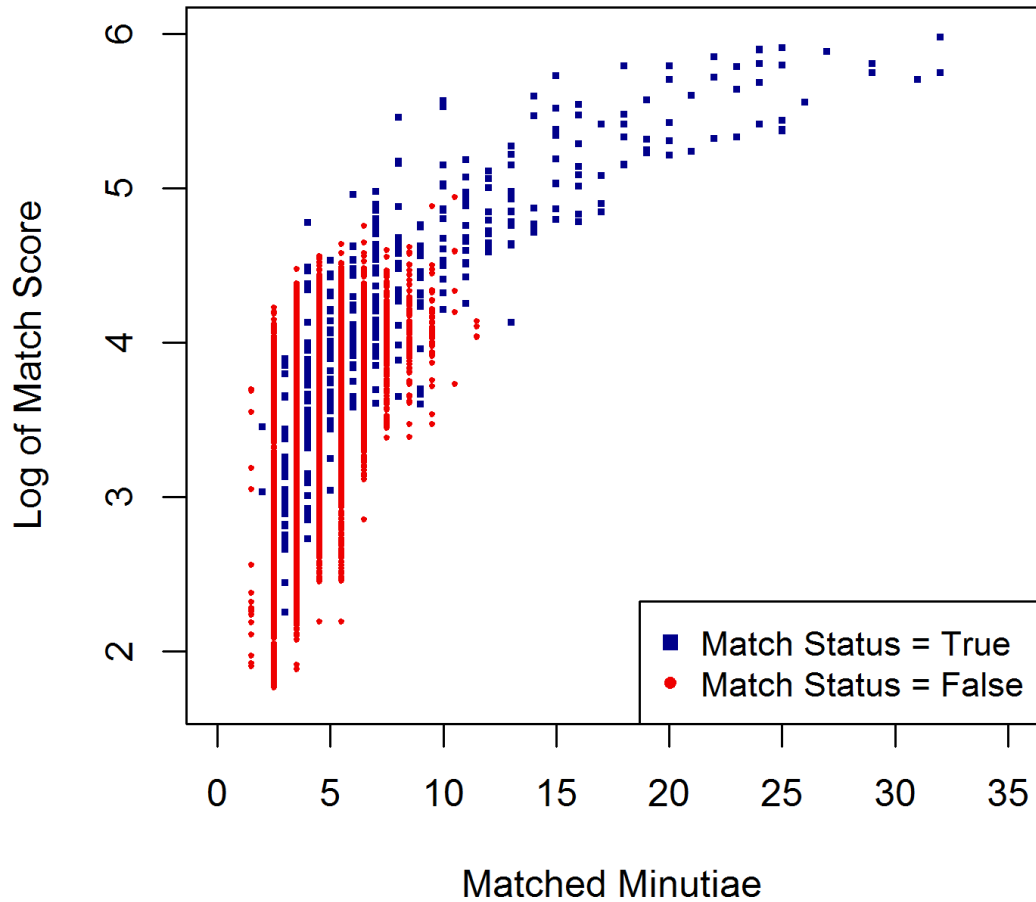
Figure 4.5: Scatter Plot of Match Minutiae vs Match Score

Illustrated in *Figure 4.5* are the matches made by the system, based on the match minutiae vs the $\log_{10}$ of the match score (log match score). The data points for the match minutiae were shifted by 0.5 on the x-axis for the match status of false (non-matches). This was done to prevent the data points for the matches and non-matches from being plotted directly over each other and to enable visualization. The results show an upward trend,

where an increase in the matched minutiae showed an increase in the log match score. A log match score of 6 (match score = 1000000) and match minutiae of 32 were obtained for the matches whereas the highest log match score observed for the non-matches was 5 (match score = 100000) with 13 matching minutiae.
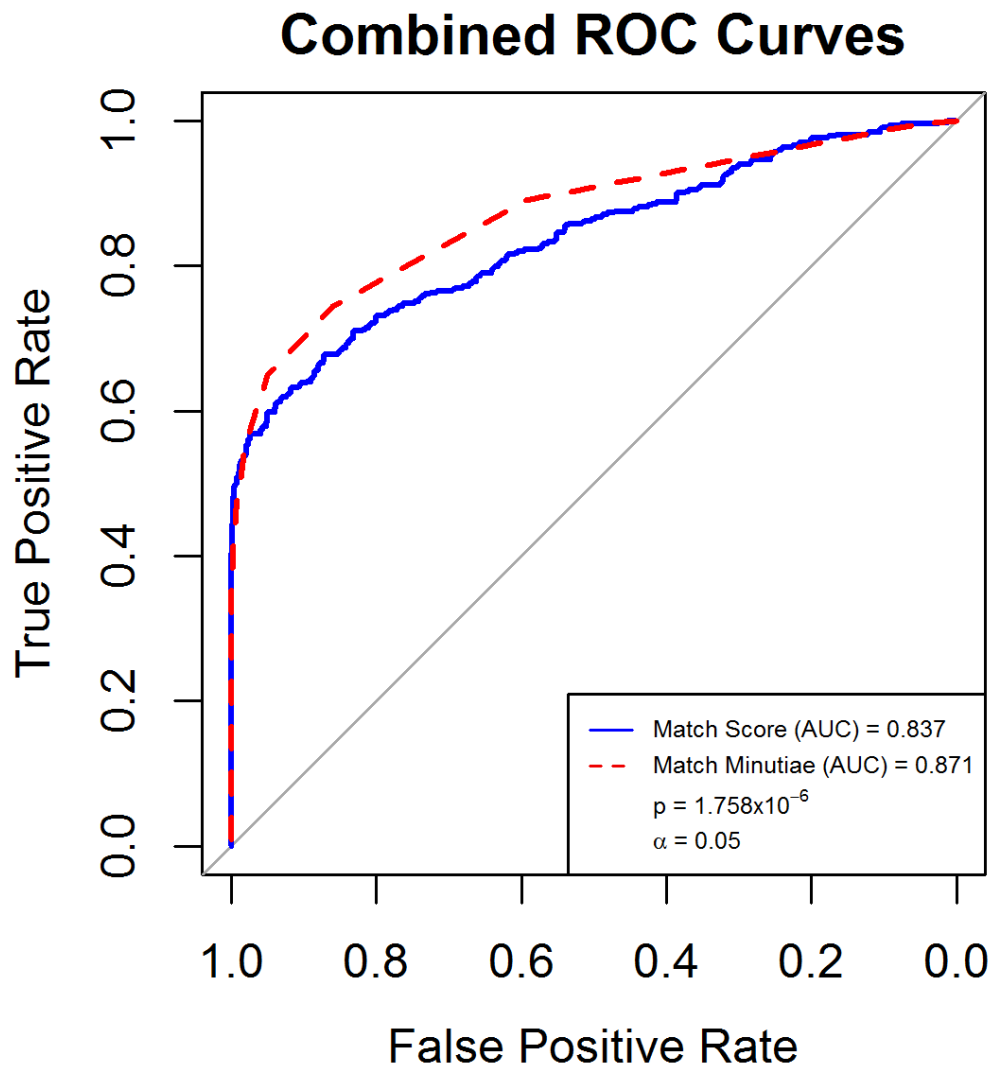
Figure 4.6: Combined ROC Curve of Match Minutiae and Match Score

ROC curves (as seen in *Figure 4.6*) were employed to assess the results obtained from

the AFIX Tracker® with respect to the match score and match minutiae. An AUC of 0.837 was determined for the match status vs match score curve, while AUC of 0.871 was determined for the match status vs match minutiae curve. Hence, it can be concluded that the match minutiae performed significantly better than the match score and was better in discriminating the true matches from the false matches (DeLong [33], p=1.76x10 $^{-6}$ at 95% confidence).
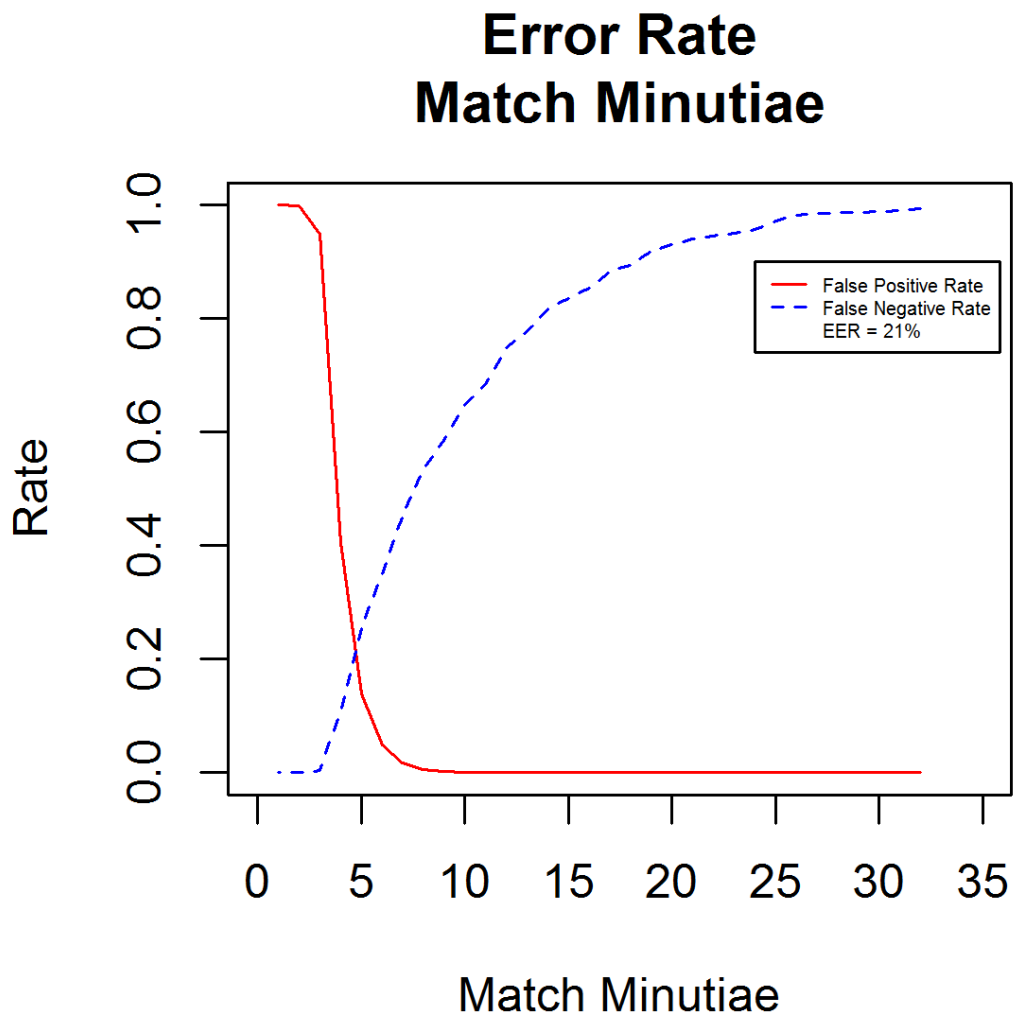
Figure 4.7: EER of Match Minutiae vs Match Status

Figure 4.8: EER of Match Score vs Match Status

Equal error rate or EER plots were constructed for both the match minutiae and the match score. An equal error rate is a location on the curve where the false positive rate (FPR) and the false negative rate (FNR) are equal. The lower the EER, the more accurate is the system in determining a true match as there is less chance of an error. As it assess the potential error rate in the performance of the system, ideally the error rate should

be zero. As this research study deals with variability encountered in a biometric dataset, deviation from the ideal is expected. [34] Displayed in *Figure 4.7* is the error rate plot with respect to match minutiae. An EER of 21% was determined at approximately 5 match minutiae indicating that when 5 or less minutiae were matched between a latent and a ten-print card, the latent print was more likely to be classified as a match despite being non-match. Concluding that, at 5 match minutiae or above, the false positive rate decreases and the false negative rate increases.

As compared to match minutiae, the EER for the match score was found to be at 25% where at a log match score value of approximately 4 or above, the false negative rate increases and the false positive rate decreases (*Figure 4.8*). This plot shows that a latent with a log match score of approximately 4 (match score = 10000) or less, was more likely to be classified as a match instead of a non-match.
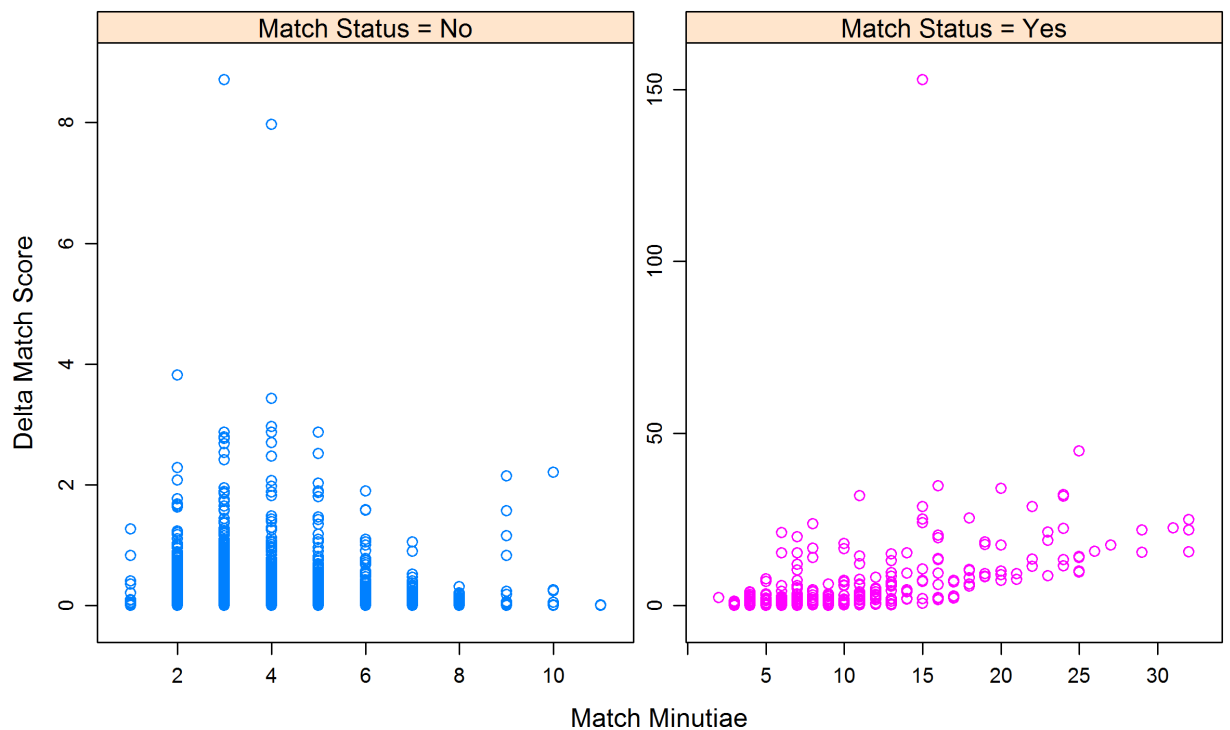
Figure 4.9: Match Minutiae vs Delta Match Score
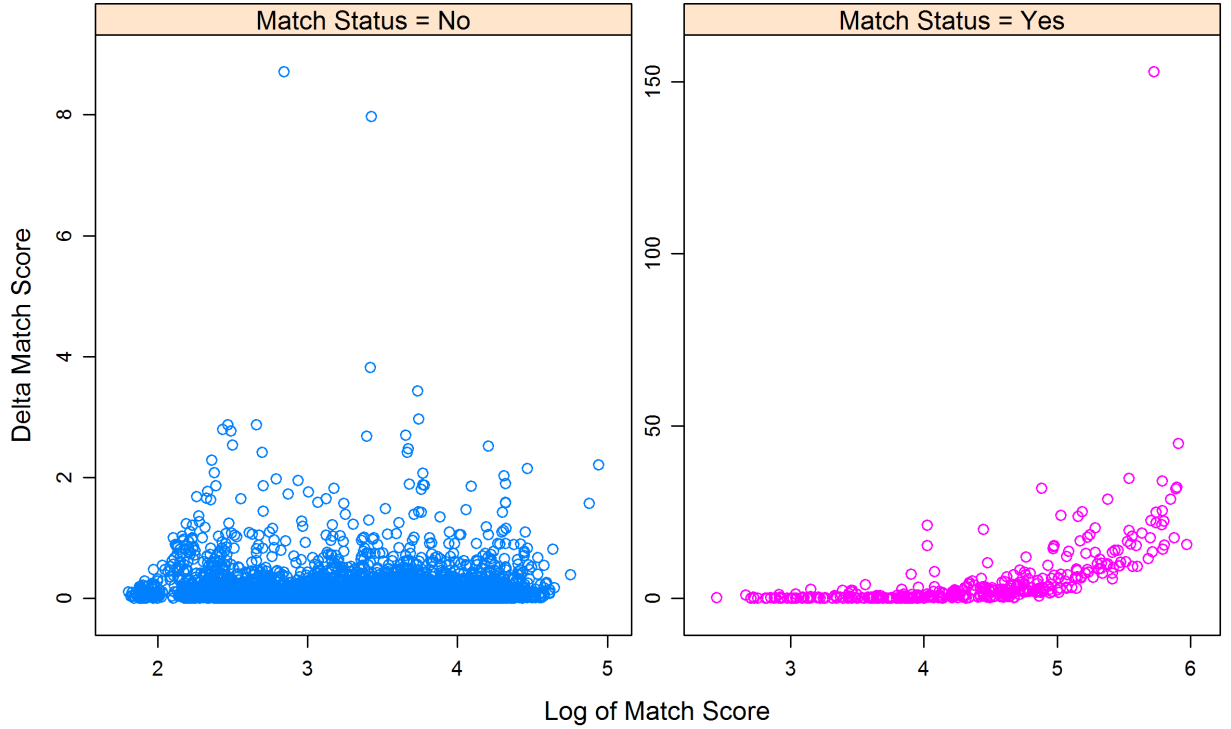
Figure 4.10: Match Score vs Delta Match Score

Figure 4.9 and *Figure 4.10* illustrate the match minutiae vs delta score, and match score vs the delta match score, respectively. The delta match score signifies the difference between the match scores obtained from each individual run. The delta match score was calculated for each individual search performed using *Equation 4.1* where A is the highest match score in that particular search and B is the second highest match score

$$\frac{A - B}{B} \tag{4.1}$$

The similar process was performed for the next set of match scores using *Equation 4.2* where B is the second highest match score and C is the third highest match score.

$$\frac{B - C}{C} \tag{4.2}$$

This was repeated for all subsequent match scores in every search result, where the last match score in every individual run was disregarded. The delta match score was determined for every individual run conducted by the AFIX Tracker®.

Plots were constructed, where the log of match score and match minutiae were plotted against the delta match score (note: the y-axis for match status of "no" only ranges from 0 to approximately 8 whereas the y axis for the match status of "yes" reaches 150). As the match score for match status "no" increased, "no" notable increase in the delta match score is observed, with the highest delta score at approximately 8. Whereas an upward trend was observed for the match status of "yes", with the delta match score going as high as 150 (refer to *Appendix D* for examples of trends of the individual search results).

A similar trend was observed for the match minutiae. As shown in *Figure 4.10*, as the match minutiae increased for the match status of "yes", the delta match score showed an increase (exponential increase), whereas for the match status of "no", a linear trend was observed. Concluding that even though match minutiae was a better indicator of a true match than match score, there was a notable difference in the delta match scores for the true vs the false matches when analyzing match minutiae and match score.

Match_Status

Yes	0
No	100

Match_Score

0 to 1000	27.6
1000 to 2000	16.9
2000 to 5000	23.7
5000 to 10000	21.6
10000 to 25000	9.84
25000 to 50000	0.37
50000 to 1e5	.009
1e5 to 1.5e5	0 +
1.5e5 to 2.5e5	0 +
2.5e5 to 5e5	0 +
5e5 to 7.5e5	0 +
7.5e5 to 1e6	0 +
1e6 to 1.25e6	0 +
4710 ± 5700	

Match_Min

1 to 3	3.74
3 to 5	81.5
5 to 7	12.8
7 to 10	1.97
10 to 15	.033
15 to 20	0 +
20 to 25	0 +
25 to 30	0 +
30 to 35	0 +
4.27 ± 1.2	

Min_Marked

3	0.98
5	6.03
7	8.00
10	10.8
13	0.24
14	0.46
15	13.1
17	0.21
18	0.42
19	0.43
20	12.9
24	0.21
25	11.5
27	0.21
28	0.21
30	10.1
32	0.21
35	8.32
37	0.21
39	0.21
40	7.14
42	0.21
43	0.21
44	0.20
45	4.10
47	0.21
50	3.14
22.7 ± 13	

Delta

0 to 7e-4	8.35
7e-4 to 0.0012	2.48
0.0012 to 0.0016	2.05
0.0016 to 0.002	2.00
0.002 to 0.003	5.08
0.003 to 0.004	5.20
0.004 to 0.005	5.54
0.005 to 0.006	4.70
0.006 to 0.007	3.68
0.007 to 0.008	3.46
0.008 to 0.009	2.98
0.009 to 0.01	3.18
0.01 to 0.011	2.85
0.011 to 0.012	2.50
0.012 to 0.013	2.35
0.013 to 0.014	1.96
0.014 to 0.015	1.85
0.015 to 0.017	3.39
0.017 to 0.019	2.84
0.019 to 0.02	1.26
0.02 to 0.03	8.87
0.03 to 0.04	5.53
0.04 to 0.05	3.32
0.05 to 0.06	2.56
0.06 to 0.07	1.75
0.07 to 0.1	3.43
0.1 to 1	6.54
1 to 10	0.29
10 to 152.81	0 +
0.0677 ± 0.36	

Figure 4.11: Baysian network with the Match_Status = No instantiated

Match_Status

Yes	100
No	0

Match_Score

0 to 1000	5.88
1000 to 2000	6.76
2000 to 5000	11.9
5000 to 10000	12.6
10000 to 25000	13.9
25000 to 50000	17.6
50000 to 1e5	11.6
1e5 to 1.5e5	4.36
1.5e5 to 2.5e5	4.76
2.5e5 to 5e5	5.59
5e5 to 7.5e5	3.92
7.5e5 to 1e6	1.03
1e6 to 1.25e6	0 +
88700 ± 170000	

Match_Min

1 to 3	0.98
3 to 5	25.5
5 to 7	19.7
7 to 10	19.1
10 to 15	18.5
15 to 20	9.33
20 to 25	4.13
25 to 30	1.85
30 to 35	0.83
9.5 ± 6.1	

Min_Marked

3	0.98
5	6.03
7	8.00
10	10.8
13	0.24
14	0.46
15	13.1
17	0.21
18	0.42
19	0.43
20	12.9
24	0.21
25	11.5
27	0.21
28	0.21
30	10.1
32	0.21
35	8.32
37	0.21
39	0.21
40	7.14
42	0.21
43	0.21
44	0.20
45	4.10
47	0.21
50	3.14
22.7 ± 13	

Delta

0 to 7e-4	0.67
7e-4 to 0.0012	0.23
0.0012 to 0.0016	0 +
0.0016 to 0.002	0 +
0.002 to 0.003	0.45
0.003 to 0.004	0.89
0.004 to 0.005	0 +
0.005 to 0.006	0.90
0.006 to 0.007	0.68
0.007 to 0.008	0.67
0.008 to 0.009	0.23
0.009 to 0.01	0.45
0.01 to 0.011	1.36
0.011 to 0.012	0 +
0.012 to 0.013	0.21
0.013 to 0.014	0.21
0.014 to 0.015	0.65
0.015 to 0.017	0.44
0.017 to 0.019	0.67
0.019 to 0.02	0 +
0.02 to 0.03	2.42
0.03 to 0.04	3.11
0.04 to 0.05	0.87
0.05 to 0.06	0.89
0.06 to 0.07	1.31
0.07 to 0.1	2.21
0.1 to 1	25.3
1 to 10	40.7
10 to 152.81	14.5
14.2 ± 32	

Figure 4.12: Baysian network with the Match_Status = Yes instantiated

Figures 4.12 - Figure 4.14 display the Bayesian networks constructed to demonstrate the relationship between the match status (Match_Status), match score (Match_Score), match minutiae (Match_Min), minutiae marked (Min_Marked) and delta match score (Delta). Match status of "yes" indicates a true match result while a match status of no indicates a non-match result. When "no" from the match status node was instantiated, the distribution of the data was analyzed. As shown in *Figure 4.11*, when 3-5 minutiae were matched, 81.5% of the prints were non-matches, where the highest match score observed was $1\text{x}10^5$ and had a delta match score below 10. Whereas when "yes" from the match status node was instantiated (*Figure 4.12*), it was observed that about 72.44%

of the data laid above 5-7 matched minutiae, with the match score as high as 1×10^6 and 55.2% of delta match scores were between 1 and 152.
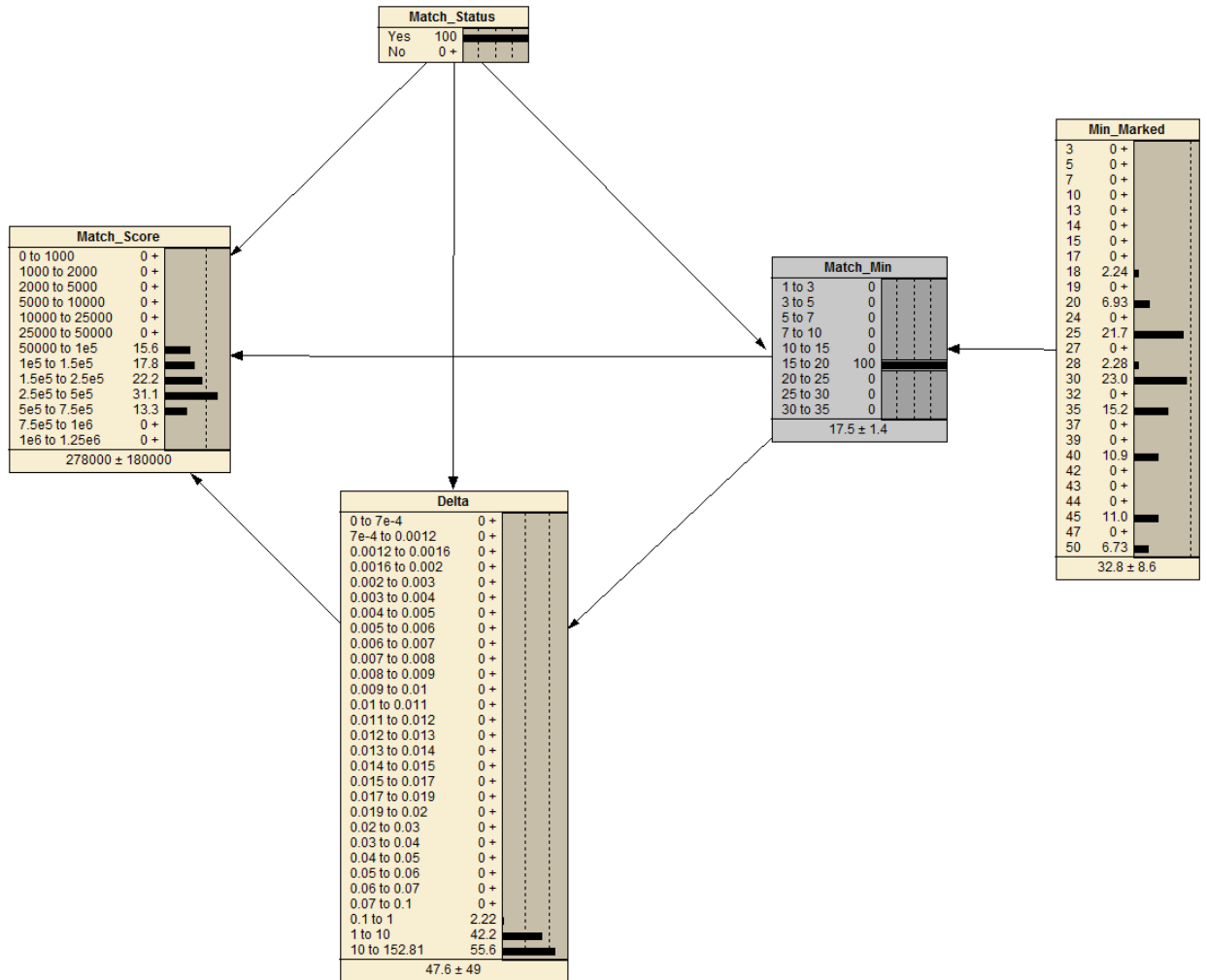
Figure 4.13: Bayesian network illustrating the node Match_Min instantiated at 15-20 and where Match_Status = Yes

Match_Status

Yes	0
No	100

Min_Marked

3	0 +
5	0 +
7	0 +
10	0 +
13	0 +
14	0 +
15	0 +
17	0 +
18	0 +
19	0 +
20	6.66
24	0 +
25	20.0
27	0 +
28	0 +
30	13.3
32	0 +
35	13.3
37	0 +
39	0 +
40	26.7
42	0 +
43	0 +
44	0 +
45	6.67
47	0 +
50	13.3
	35.3 ± 9

Match_Score

0 to 1000	0 +
1000 to 2000	0 +
2000 to 5000	0 +
5000 to 10000	6.67
10000 to 25000	53.3
25000 to 50000	26.7
50000 to 1e5	13.3
1e5 to 1.5e5	0 +
1.5e5 to 2.5e5	0 +
2.5e5 to 5e5	0 +
5e5 to 7.5e5	0 +
7.5e5 to 1e6	0 +
1e6 to 1.25e6	0 +
	29900 ± 22000

Match_Min

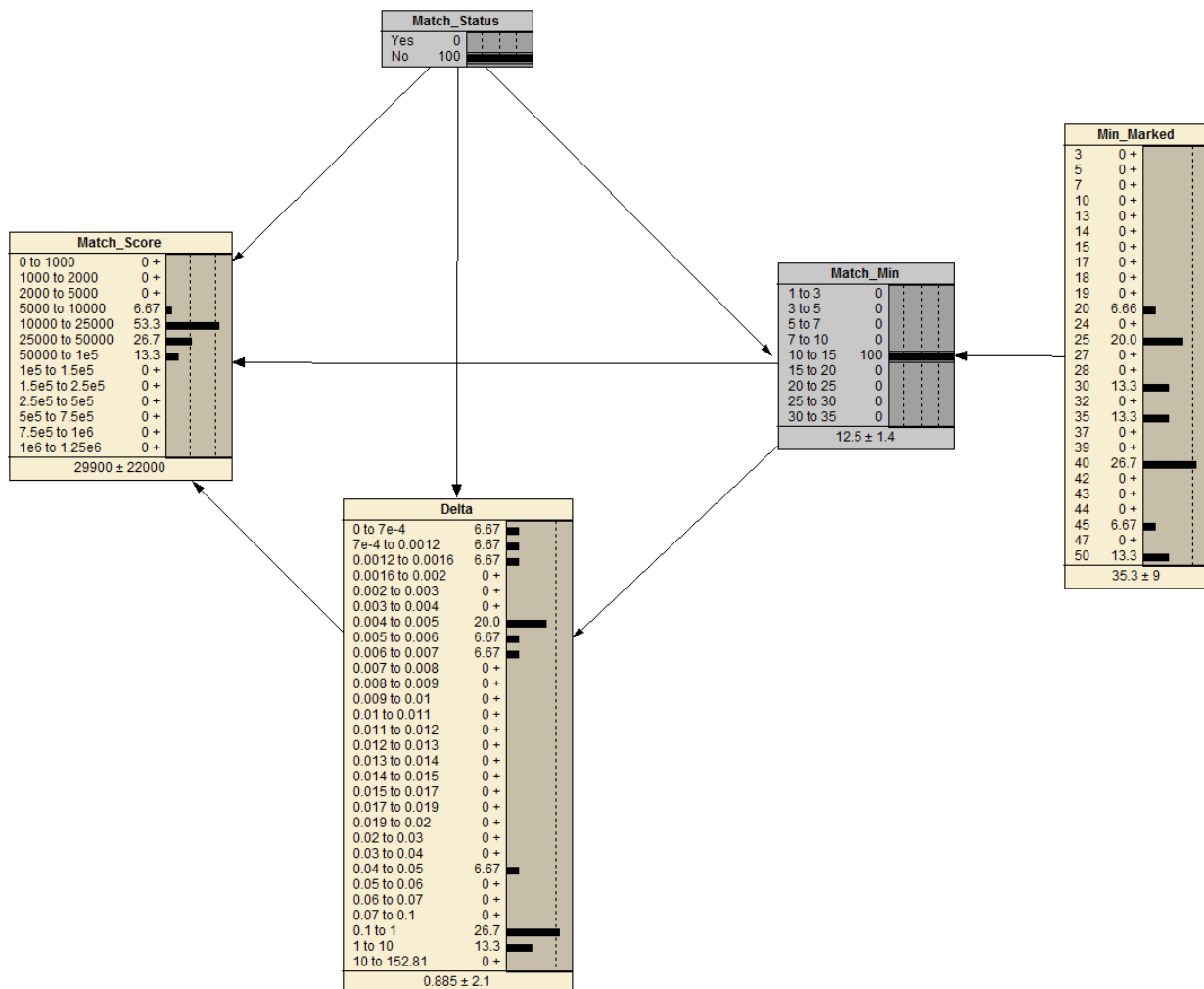|	
1 to 3	0
3 to 5	0
5 to 7	0
7 to 10	0
10 to 15	100
15 to 20	0
20 to 25	0
25 to 30	0
30 to 35	0
	12.5 ± 1.4

Delta

0 to 7e-4	6.67
7e-4 to 0.0012	6.67
0.0012 to 0.0016	6.67
0.0016 to 0.002	0 +
0.002 to 0.003	0 +
0.003 to 0.004	0 +
0.004 to 0.005	20.0
0.005 to 0.006	6.67
0.006 to 0.007	6.67
0.007 to 0.008	0 +
0.008 to 0.009	0 +
0.009 to 0.01	0 +
0.01 to 0.011	0 +
0.011 to 0.012	0 +
0.012 to 0.013	0 +
0.013 to 0.014	0 +
0.014 to 0.015	0 +
0.015 to 0.017	0 +
0.017 to 0.019	0 +
0.019 to 0.02	0 +
0.02 to 0.03	0 +
0.03 to 0.04	0 +
0.04 to 0.05	6.67
0.05 to 0.06	0 +
0.06 to 0.07	0 +
0.07 to 0.1	0 +
0.1 to 1	26.7
1 to 10	13.3
10 to 152.81	0 +
	0.885 ± 2.1

Figure 4.14: Bayesian network illustrating the node Match_Min instantiated at 10-15 and where Match_Status = No

Figure 4.13 demonstrates that with match status "yes" instantiated, when 15-20 or more minutiae were matched between the latent and the ten-print card, a 100% true match was determined with a match score of 50000 or higher and a delta match score between 0.1 and 153. However, *Figure 4.14* shows that when state "no" from node match status was instantiated, along with state 10-15 from node match minutiae, a very low delta match score was determined (between 0 and 10). This clearly demonstrates that there is a greater difference between the delta match scores of a true match and a non-match vs between
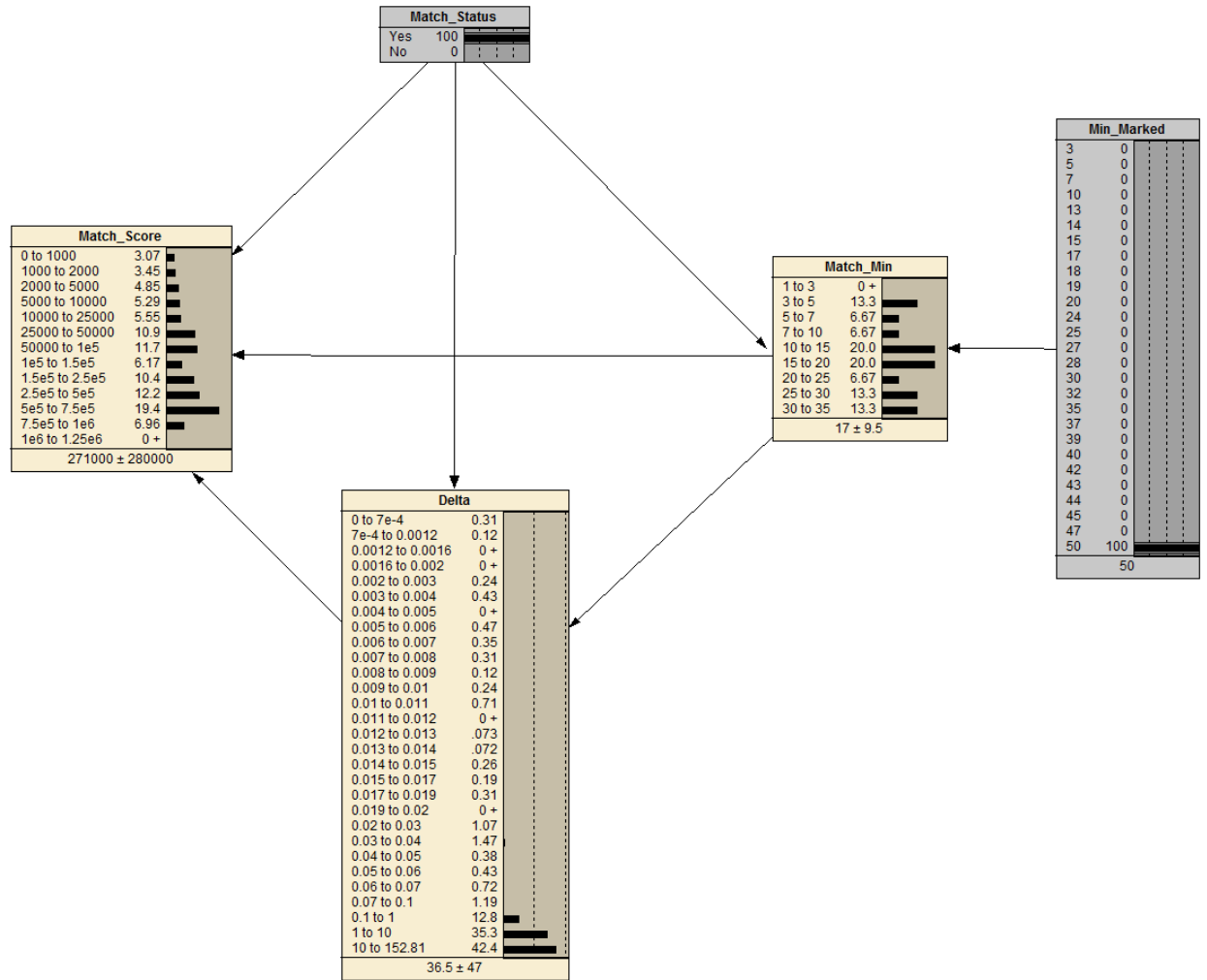
two non-matches.

Match_Status

Yes	100
No	0

Match_Score

0 to 1000	3.07
1000 to 2000	3.45
2000 to 5000	4.85
5000 to 10000	5.29
10000 to 25000	5.55
25000 to 50000	10.9
50000 to 1e5	11.7
1e5 to 1.5e5	6.17
1.5e5 to 2.5e5	10.4
2.5e5 to 5e5	12.2
5e5 to 7.5e5	19.4
7.5e5 to 1e6	6.96
1e6 to 1.25e6	0 +
271000 ± 280000	

Match_Min

1 to 3	0 +
3 to 5	13.3
5 to 7	6.67
7 to 10	6.67
10 to 15	20.0
15 to 20	20.0
20 to 25	6.67
25 to 30	13.3
30 to 35	13.3
17 ± 9.5	

Min_Marked

3	0
5	0
7	0
10	0
13	0
14	0
15	0
17	0
18	0
19	0
20	0
24	0
25	0
27	0
28	0
30	0
32	0
35	0
37	0
39	0
40	0
42	0
43	0
44	0
45	0
47	0
50	100
50	

Delta

0 to 7e-4	0.31
7e-4 to 0.0012	0.12
0.0012 to 0.0016	0 +
0.0016 to 0.002	0 +
0.002 to 0.003	0.24
0.003 to 0.004	0.43
0.004 to 0.005	0 +
0.005 to 0.006	0.47
0.006 to 0.007	0.35
0.007 to 0.008	0.31
0.008 to 0.009	0.12
0.009 to 0.01	0.24
0.01 to 0.011	0.71
0.011 to 0.012	0 +
0.012 to 0.013	.073
0.013 to 0.014	.072
0.014 to 0.015	0.26
0.015 to 0.017	0.19
0.017 to 0.019	0.31
0.019 to 0.02	0 +
0.02 to 0.03	1.07
0.03 to 0.04	1.47
0.04 to 0.05	0.38
0.05 to 0.06	0.43
0.06 to 0.07	0.72
0.07 to 0.1	1.19
0.1 to 1	12.8
1 to 10	35.3
10 to 152.81	42.4
36.5 ± 47	

Figure 4.15: Bayesian network illustrating the node Min_Marked instantiated at 50 and where Match_Status = Yes

Figures 4.15 and *4.16* demonstrates the AFIX Tracker® search results for latents marked with 50 minutiae. With the match status as "yes" (*Figures 4.15*), it can be observed that along with a high match and delta match score, the number of match minutiae was higher (3-35) whereas this was not observed when match status equals "no" (*Figure 4.16*). No more than 15-20 match minutiae were found and the corresponding match scores and delta match scores were lower compared to match status yes. This

suggests a clear distinction between the true matches and the non-matches.
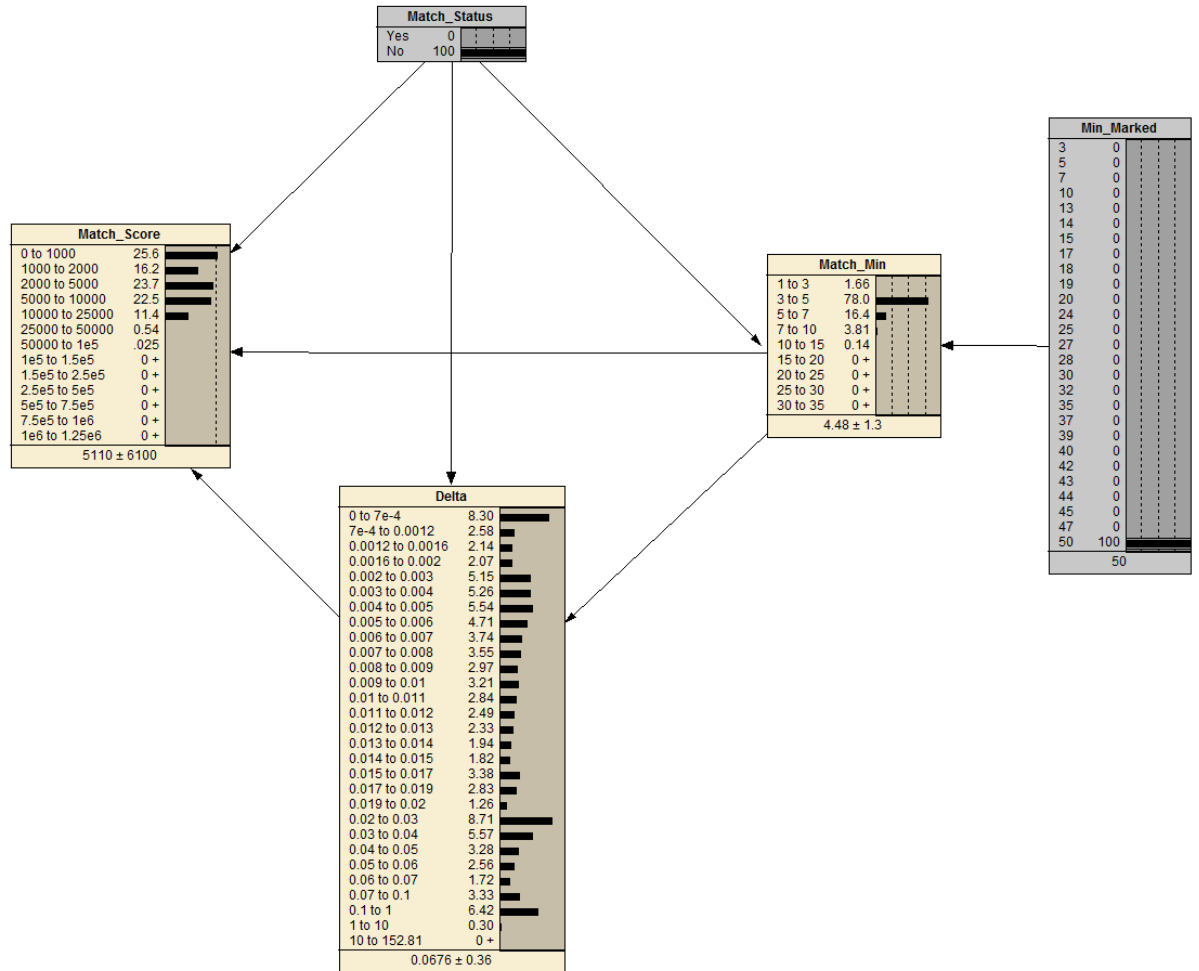

Figure 4.16: Bayesian network illustrating the node Min_Marked instantiated at 50 and where Match_Status = No

4.3.2 Performance Assessment of the Bayesian Network

The 70-30 approach, as mentioned in the methods section, was used to test the performance of the constructed Bayesian network. Seventy percent of the prints were used to establish a Bayesian network and were used as a training set and the remaining 30% were used to test how well the network was performing (referred to as test prints). In order to replicate a realistic crime scene scenario, the test prints that were obtained from the BioCoP database,

similar to the training set, were developed using three different methods: ninydrin, black powder and cyanoacrylate fuming. A subjective quality assessment was conducted on the prints and each latent was classified as good, bad or ugly. A print was classified as good if more than 25 minutiae were marked, as bad if minutiae between 10-24 were marked and ugly if less than 10 minutiae were marked. Fifty seven percent of the latent prints from the test prints were classified as good, 28% were classified as bad and 15% were classified as ugly. In comparison, 56% of the prints in the training set were classified as good, 26% were classified as bad and 18% were classified as ugly.
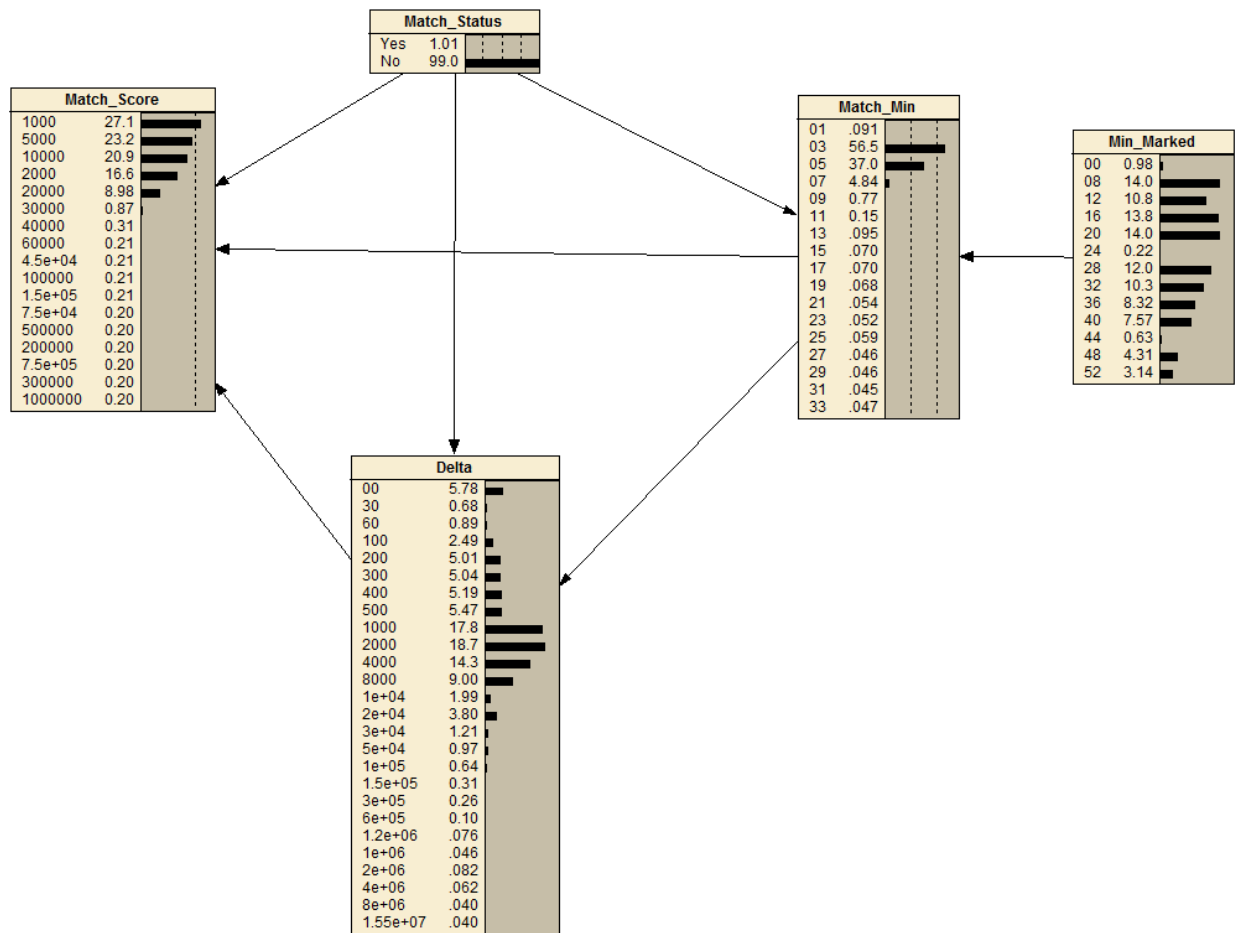
Figure 4.17: Bayesian network constructed using the training set of latent prints

As the Bayesian network was the evidence evaluation method used to analyze the results obtained from the AFIX Tracker®, it is essential to measure how accurately the

network itself performs. The network was built using the training set of latent prints (140 latent prints) where these prints were used to establish a validation database as their identity was known (ground truth is established). The test prints (60 latent prints) were considered to be a representation of future latent prints that would be entered into AFIX Tracker® and were hence used as simulation cases to assess the performance of the network (that was built based on the training set) prior to being used in casework. RNetica was used to construct the Bayesian network and using this constructed network, LR values for the test prints were generated (see *Appendix A* for the R script).

Figure 4.17 used the same training set prints that were used to construct the networks in *Figure 4.11* - *Figure 4.16*, with the only difference in the value of nodes. *Figure 4.17* uses discrete node values whereas *Figure 4.11* - *Figure 4.16* displays the nodes as continuous.

The performance of the network was represented using probability distribution plots such as Tippett and ECE plots. As mentioned in *Section 3.2.2*, Tippett plots utilize calculated LRs to determine how many cases above a certain likelihood ratio support the prosecutorial or defense hypothesis, where the LRs were calculated using the match score and match minutiae. The LRs were computed based on the totality of the test prints data, irrespective of a specific match minutiae or match score.
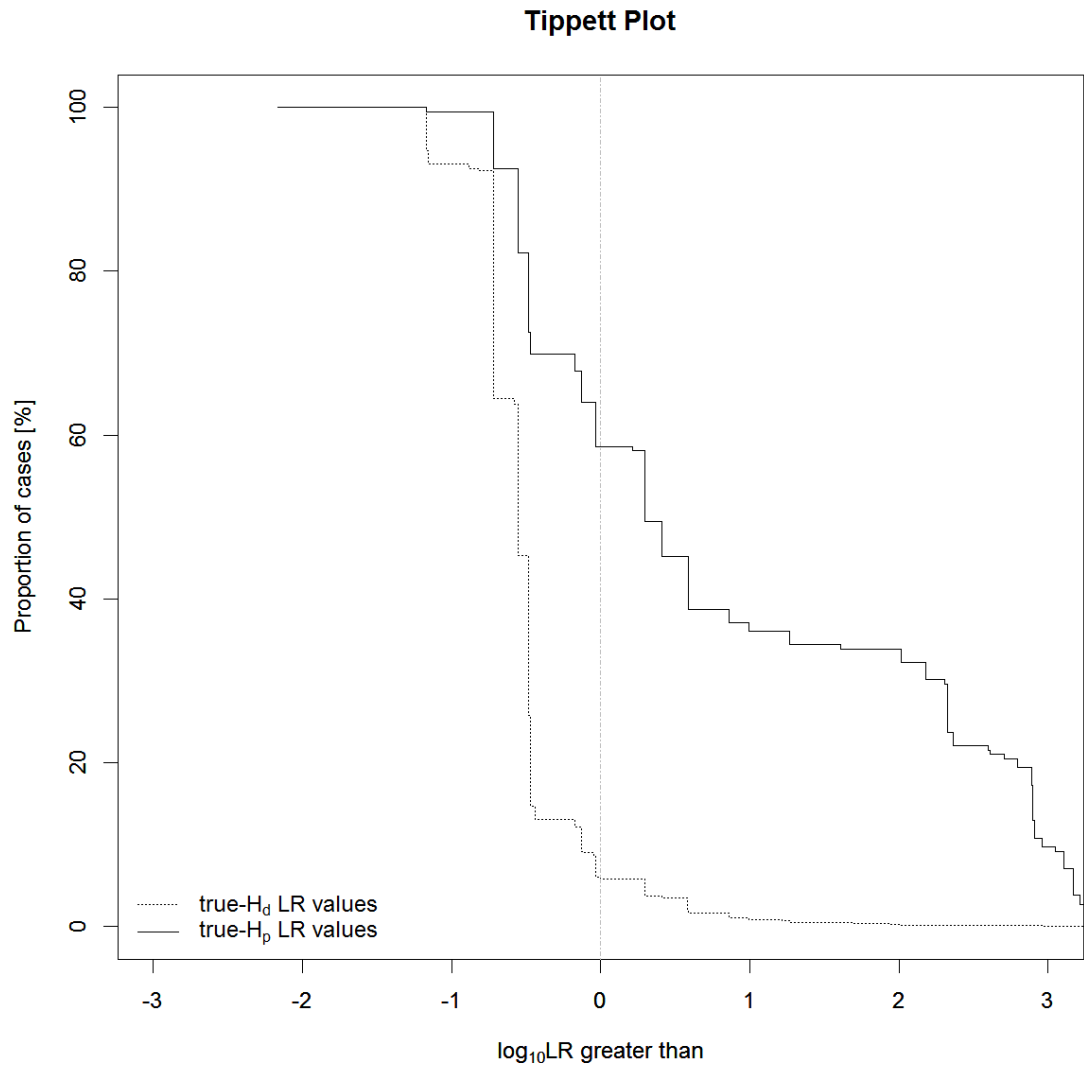
Tippett Plot

Figure 4.18: Tippett plot

As seen in the Tippet plot in *Figure 4.18*, it can be observed that there was an overlap between the two competing hypothesis where 41% of the cases that were true matches had a LLR of 0 or lower, hence incorrectly supporting the defense hypothesis, classifying the prints as non-matches. Whereas 5.7% of the cases that were non-matches and had a LLR of 0 or higher were being classified as matches, therefore incorrectly supporting the

prosecutorial hypothesis. Upon further inspection, it was determined that the percentage of the cases that gave false positives/false negative result were high quality prints and were developed on glass and dusted using black powder (refer to *Appendix E* for results obtained using different development methods). This may be due to minutiae being marked in common areas such as the core and delta regions. The LLRs for the non-matches were interesting: the highest LLR that was obtained had a match score of about 50000 where 50 minutiae were marked and only 7 minutiae was matched. The same data set had a match score of 74000 with 8 matching minutiae and 30 marked minutiae but still had a lower LLR. This indicated that apart from match score and matching minutiae, the number of marked minutiae also has an affect in determining the match score, which in turn influenced the LLR. This may be due to more minutiae being matched in the latent having 30 minutiae marked vs. the latent that had 50 minutiae marked where only 7 matched.
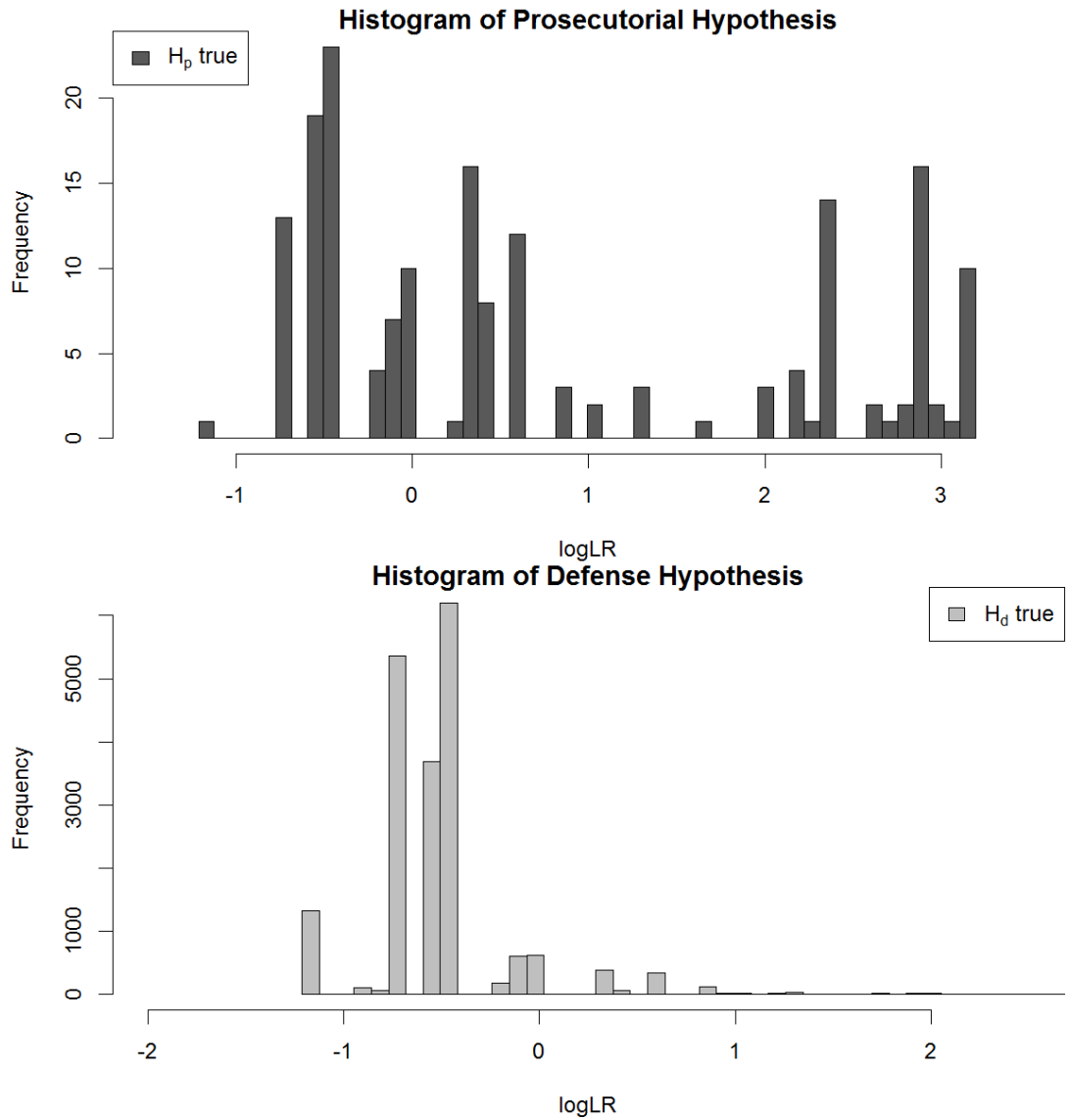
Figure 4.19: LLR histogram displaying frequency of prosecutorial and defence hypothesis

Figure 4.19 shows the frequency distribution of the LLRs for each of the given hypothesis. Fifty nine perfect of the LLRs for H_p lie over 0 while 95% of the LLRs for the H_d lie below 0, which indicates that the network was not able to distinguish between the matches and non matches accurately.
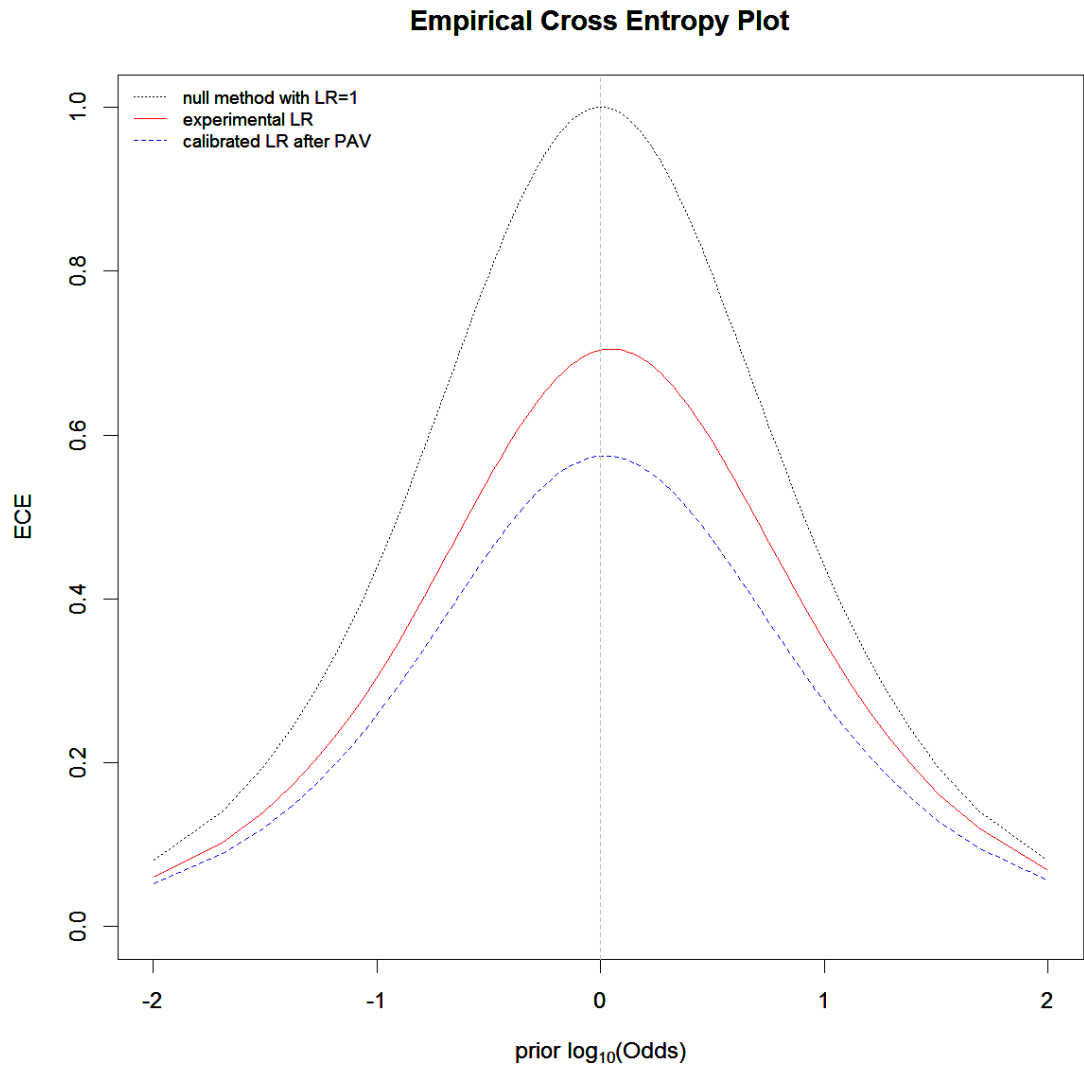
Figure 4.20: Emperical cross entropy plot

Empirical cross entropy (ECE) plots were constructed to further assess the overall performance of the LR values. ECE plots take into account calibration along with the discriminatory power of the LR values. Upon examining the ECE plot in *Figure 4.20*, the calculated LR values were used in assessing the performance of the system. The established model does not have a perfect accuracy (where the experimental LR curve is

closer to the null method than 0) but does provides a better accuracy assessment of the data than the null method (LR = 1).

While the plots constructed above displays the performance of the system with respect to the totality of the test print data, this study further investigated performance of the network based on the match minutiae. *Table 4.2* displays the misleading evidence rates.

Table 4.2: Misleading evidence based on match minutiae

Match Min	Percentage of misleading evidence given H_p (LLR < 1)	Percentage of misleading evidence given H_d (LLR > 1)
1-3	100%	1%
4-7	12%	58%
8-14	0%	100%
15-40	0%	0%

It can be seen that when 1-3 minutiae were matched between the latent and the ten-print card, all latents that were a true match were being classified as a non-match (false negative) whereas only 1% of the prints that were a non-match were being classified as a match (false positive). The highest mis-classification for the matches were observed when minutiae between 1-3 and 4-7 were matched whereas the majority of the mis-classification for the non-matches was found when minutiae between 4-7 and 8-14 were matched. When more than 15 minutiae were marked, none of the prints were mis-classified (refer to *Appendix F* for the Tippet, ECE and Histogram plots associated with different match minutiae).

While this network can be utilized in assessing the identity of an unknown latent, the discrimination power and accuracy of the system was not optimum, hence there is a high chance that a unknown print is classified incorrectly, based on the number of minutiae matched.

4.4   Conclusion

Often, fingerprint examiners depend on the AFIS system to provide a potential candidate list which they then utilize to make a manual comparison. Currently in the United States, there is no fixed number of minutiae that needs to be marked on a latent to make an identification, hence this is done at the discretion of each individual examiner. This study was intended to determine the accuracy of the constructed Bayesian network, using 200 latent prints, in determining a true match vs a non-match when a range of minutiae were marked on the latent print. This research study also assessed the match minutiae, match score, marked minutiae, delta match score and match status to determine which variable or combination of variables was a better indicator of a true match. As expected, it was found that as the number of marked minutiae on the latent increased, the total number of matching minutiae between the latent and ten-print card also increased. Based on match

minutiae vs the match score scatter plot, as the number of match minutiae increased, there was an increase in the match score. A higher match score was determined for the true matches as compared to the non-matches.

From looking at the combined ROC curve of the match minutiae and match score, match minutiae, with an AUC of 0.87 was significantly higher than the match score, with an AUC of 0.84 (DeLong [33], p=1.6x10 $^{-6}$ at 95% confidence), concluding that it was a better indicator of a true match compared to the match score. Furthermore, the EER for the match minutiae (at 21%) was lower than that for the match score (at 25%).

The Bayesian networks constructed provided an assessment of the interdependence of the AFIX Tracker® variables in determining the match outcome. Interestingly, it was observed that a matching minutiae of 15 or higher resulted in a 100% true match result whereas for the non-matches, no more than 13 matching minutiae were observed. Moreover, the delta match scores difference between the matches and non-matches were very notable (delta score of 0.1-153 for matches compared to a score of 0-0.1 for the non-matches). Through this research, it can be concluded that a matching minutiae of 15 or higher produces a high match score along with a notable difference in the delta score between a true match and a non-match compared to two non-matches.

Lastly, while assessing the performance of the Bayesian network, the known prints were used to establish the baseline for the Bayesian network and the unknown prints were run against the network to determine how accurately the prints were classified by the constructed network. Based on the Tippett and histogram plots, some errors were determined. There was a 41% mis-classification rate where the true matches were being classified as non-matches and a 5% mis-classification rate where the non-matches were being classified as matches. Furthermore, as seen in the ECE plots, the discrimination power of the system was poor and the accuracy of the system was not optimum, hence there is a higher chance that the unknown print is classified incorrectly. This low accuracy and poor discrimination might be due to the fact that insufficient latent prints were utilized. Only 60 unknown prints were used to assess the accuracy of the system, while only 140 known prints provided a baseline for the network. Due to this small sample size, it is highly likely that a large amount of variability was observed in the data, resulting in decrease in discrimination power.

4.5   Future Research

Based on the results found in this research, there is future work that can be conducted using this method to evaluate the likelihood to accurately identify an unknown latent. Increasing the number of latent prints in the training set would allow for a better and more extensive representation of the variability observed in the test set.

Another variable that can be assessed is the quality of the latent and ten-print cards. Providing a quality assessment (such as using NFIQ by NIST) would help explain the variability in the data. For example, the NFIQ algorithm can be used to assess the low and high quality areas of the print. The high quality areas can be selected and run against

the AFIX Tracker® to observe any difference in the results. Bayesian networks can be constructed and the performance of the system can be assessed using these results.

This research project combined two databases that contained ten-print cards at different resolutions (500ppi and 600ppi). No significant differences were observed between the two resolutions, which could be due to a small difference between the two. Future studies should consider keeping the resolution of the latent and the ten-print cards consistent to avoid any potential discrepancies. Furthermore, as the known prints were randomly selected, there was no control over the quality of the ten-print cards. Ideally, a ten-print card should be of good quality but this was not always the case in this study. Some ten-print cards (approximately 10%) were of very poor quality (less than 10 marked minutiae per finger) while the latent was of a higher quality (contained more than 40 marked minutiae). This difference could have resulted in lower match scores between the two prints or a higher rate of non-match.

Apart from the AFIX Tracker®, AFIS systems such as MorphoTrak and Cogent are also being utilized by fingerprint examiners. It would be interesting to see how these databases perform as compared to AFIX Tracker®.

Overall, this research provided a quantitative method to assesses the minimum number of minutiae needed to determine a true match. Such studies, combined with a Bayesian analysis could provide valuable information while assessing the likelihood of a latent print belonging to a particular donor. While there were some errors in mis-classification observed, this project sets to serve as a preliminary study upon which extensive research can be done to provide promising results.

Bibliography

[1] Biometrics;. Available from: `http://www.dynotech.com/articles/biometrics.shtml`.

[2] Triggs CM, Buckleton JS, Walsh SJ. Forensic DNA Evidence Interpretation. Boca Raton, Florida: CRC Press; 2004.

[3] Müler C. Speaker Classification I: Fundamentals, Features, and Methods, vol. 4343/2007. Springer: Lecture Notes in Computer Science, Berlin; 2007.

[4] Ramos D, Gonzalez-Rodriguez J. Reliable support: measuring calibration of likelihood ratios. Forensic science international. 2013;230(1):156–169.

[5] Hanley JA, Mcneil BJ. The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology. 1982;143(1):29–36.

[6] Kellman PJ, Mnookin JL, Erlikhman G, Garrigan P, Ghose T, Mettler E, et al. Forensic comparison and matching of fingerprints: using quantitative image measures for estimating error rates through understanding and predicting difficulty. PloS one. 2014;9(5):e94617.

[7] Cole SA. Is fingerprint identification valid? Rhetorics of reliability in fingerprint proponents discourse. Law & Policy. 2006;28(1):109–135.

[8] Council NR. Strengthening Forensic Science in the United States: A Path Forward. The National Academies Press; 2009.

[9] Scientific Working Group on Friction Ridge Analysis ST. Quality Assurance Guidelines for Latent Print Examiners; 2002. Available from: `http://www.swgfast.org/documents/quality-assurance/020822_Quality-Assurance_Latent_2.1.pdf`.

[10] Haber L, Haber RN. Scientific validation of fingerprint evidence under Daubert. Law, Probability and Risk. 2008;7(2):87–109.

[11] of Investigation FB. Integrated Automated Fingerprint Identification System; 2010. Available from: `https://www.fbi.gov/about-us/cjis/fingerprints_biometrics/iafis`.

[12] Acharya T, Bhattacharya BB, Biswas A, Bhowmick P, Bishnu A, Das S, et al.. Fingerprint minutiae matching using scoring techniques. Google Patents; 2008. US Patent 7,359,532.

[13] Dror I, Mnookin J. The use of technology in human expert domains: Challenges and risks arising from the use of automated fingerprint identification systems in forensic science. Law Probability and Risk. 2010;9:47–67.

[14] Beluco A, Engel PM, Beluco A. Classification of textures in satellite image with Gabor filters and a multi layer perceptron with back propagation algorithm obtaining high accuracy. International Journal of Energy and Environment. 2015;6(5):437.

[15] Dror IE, Harnad S. Cognition distributed: How cognitive technology extends our minds. vol. 16. John Benjamins Publishing; 2008.

[16] Grossman D, Domingos P. Learning Bayesian network classifiers by maximizing conditional likelihood. In: Proceedings of the twenty-first international conference on Machine learning. ACM; 2004. p. 46.

[17] Norsys. Introduction to Bayes Nets;. Available from: `https://www.norsys.com/tutorials/netica/secA/tut_A1.htm`.

[18] Alberink I, Jongh A, Rodriguez C. Fingermark evidence evaluation based on automated fingerprint identification system matching scores: the effect of different types of conditioning on likelihood ratios. Journal of forensic sciences. 2014;59(1):70–81.

[19] Haraksim R, Meuwly D. Influence of the Datasets Size on the Stability of the Lr in the Lower Region of the within Source Distribution; 2013. .

[20] Anthonioz NE, Champod C. Evidence evaluation in fingerprint comparison and automated fingerprint identification systems - Modeling between finger variability. Forensic science international. 2014;235:86–101.

[21] Egli NM, Champod C, Margot P. Evidence evaluation in fingerprint comparison and automated fingerprint identification systemsmodelling within finger variability. Forensic science international. 2007;167(2):189–195.

[22] Neumann C, Mateos-Garcia I, Langenburg G, Kostroski J, Skerrett JE, Koolen M. Operational benefits and challenges of the use of fingerprint statistical models: a field study. Forensic science international. 2011;212(1):32–46.

[23] DOJ. The universal latent workstation / version 6.4;. Available from: `https://www.fbibiospecs.cjis.gov/`.

[24] Biedermann A, Gittelson S, Taroni F. Recent misconceptions about the database search problem: A probabilistic analysis using Bayesian networks. Forensic Science International. 2011;212:51–60.

[25] Fimmers R, H Schneider MP, M BP, Schneider. Reply to the letter of Taroni et al. to the Editor with reference to Schneider et al. Recommendations of the German Stain Commission regarding the statistical evaluation of matches following searches in the national DNA database. Rechtsmedizin. 2011;21:57–60.

[26] Taroni F, Biedermann A, Coquoz R, Hicks T, Champod C. Letter to the Editor with reference to Schneider et al. Recommendations of the German Stain Commission regarding the statistical evaluation of matches following searches in the national DNA database. Rechtsmedizin. 2011;21:55–57.

[27] Lawless C, Shaw I, Mennell J. of Applied Sciences Northumbria University S, editor. The Current Position of Fingerprint Evidence - a Literature Review; 2009.

[28] Neumann C, Champod C, Puch-Solis R, Egli N, Anthonioz A, Meuwly D, et al. Computation of likelihood ratios in fingerprint identification for configurations of three minutiae. Journal of forensic sciences. 2006;51(6):1255–1266.

[29] Park SH, Goo JM, Jo CH. Receiver operating characteristic (ROC) curve: practical review for radiologists. Korean Journal of Radiology. 2004;5(1):11–18.

[30] Lucena-Molina JJ, Ramos-Castro D, Gonzalez-Rodriguez J. Performance of likelihood ratios considering bounds on the probability of observing misleading evidence. Law, probability and risk. 2015;14(3):175–192.

[31] Drygajlo A, Jessen M, Gfroerer S, Wagner I, Vermeulen J, Niemi T. Methodological guidelines for best practice in forensic semiautomatic and automatic speaker recognition. Verlag für Polizeiwissenschaft; 2016.

[32] Robertson B, Vignaux GA, Berger CE. Interpreting evidence: evaluating forensic science in the courtroom. John Wiley & Sons; 2016.

[33] DeLong ER, DeLong DM, Clarke-Pearson DL. Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach. Biometrics. 1988;p. 837–845.

[34] Chin R. Computer vision-ACCV'98: third Asian conference on computer vision, Hong Kong, China, January 8-10, 1998, proceedings. vol. 1. Springer Science & Business Media; 1998.

A. R Script to Calculate Likelihood Ratios and Construct Tippett and ECE Plots

```
library(RNetica)
library(isotone)
library(png)

set.seed(123)


# Load data
    ↪ ===============================================================
    ↪
#Fingerprints <- read.csv("C:/Users/Shreya/Desktop/TEST BN (
    ↪ delta)/Unknown Latents Data.csv")
Fingerprints <- read.csv("C:/Users/Shreya/Desktop/TEST_BN_(delta
    ↪ )/Unknown_with_References.csv")


# Discretize FP, BF, Rank and Rank_BF
    ↪ =====================================================
FingerprintsnewBF<-c()
FingerprintsnewFP<-c()
FingerprintsnewRank<-c()
FingerprintsnewRank_BF<-c()
FingerprintsnewRank_D<-c()


qt1<-subset(Fingerprints, select=c(Match_Score, Match_Min, Min_
    ↪ Marked))
```

```r
options(scipen=999)

FingerprintsnewMatch_Score <- sapply(qt1$Match_Score, function(x
    ↪ )
  if (x<50) {"A_0"}else{
    if (x<2000){paste0("A_",as.character((ceiling(x/1000))*1000)
        ↪ )}else{
      if (x<10000){paste0("A_",as.character((ceiling(x/5000))*
          ↪ 5000))}else{
        if (x<40000){paste0("A_",as.character((ceiling(x/10000))
            ↪ *10000))}else{
          if (x<75000){paste0("A_",as.character((ceiling(x/
              ↪ 15000))*15000))}else{
            if (x<150000){paste0("A_",as.character((ceiling(x/
                ↪ 50000))*50000))}else{
              if (x<250000){paste0("A_",as.character((ceiling(x/
                  ↪ 100000))*100000))}else{
                if (x<1000000){paste0("A_",as.character((ceiling
                    ↪ (x/250000))*250000))}else{
                  "A_1250000"}}}}}}}}
)
options(scipen=0)

FingerprintsnewMatch_Min <- sapply(qt1$Match_Min, function(x)
  if (x<1) {"A_0"}else{
    if (x<7){paste("A",as.character((floor(x/2))*2), sep="_")}
        ↪ else{
      if (x<35){paste("A",as.character((floor(x/5))*5), sep="_")
          ↪ }
    }
  }
)

FingerprintsnewMin_Marked <- sapply(qt1$Min_Marked, function(x)
  if (x<4) {"A_0"}else{
    if (x<55){paste("A",as.character((floor(x/4))*4), sep="_")}
  }
)

Data<-subset(Fingerprints, select=c(Match_Status)) # add IDs
```

```r
Match_Min <- FingerprintsnewMatch_Min
Match_Score <- FingerprintsnewMatch_Score
Min_Marked <- FingerprintsnewMin_Marked

Data<-cbind(Data, Match_Min, Match_Score, Min_Marked)

write.csv(Data, "C:/Users/Shreya/Desktop/TEST_BN_(delta)/Unknown
    ↪ _Latents_Data_BNValid.csv", row.names = FALSE)


#——————————————————————————KNOWN DATA
    ↪ ═══════════════════════════════════════
Fingerprints <- read.csv("C:/Users/Shreya/Desktop/TEST_BN_(delta
    ↪ )/Known_Fixed_deleted_NM.csv")


# Discretize FP, BF, Rank and Rank_BF
    ↪ ═══════════════════════════════════════
FingerprintsnewBF<-c()
FingerprintsnewFP<-c()
FingerprintsnewRank<-c()
FingerprintsnewRank_BF<-c()
FingerprintsnewRank_D<-c()


qt1<-subset(Fingerprints, select=c(Match_Score, Match_Min, Min_
    ↪ Marked, Delta))

options(scipen=999)

FingerprintsnewMatch_Score <- sapply(qt1$Match_Score, function(x
    ↪ )
  if (x<50) {"A_0"}else{
    if (x<2000){paste0("A_",as.character((ceiling(x/1000))*1000)
        ↪ )}else{
      if (x<10000){paste0("A_",as.character((ceiling(x/5000))*
          ↪ 5000))}else{
        if (x<40000){paste0("A_",as.character((ceiling(x/10000))
            ↪ *10000))}else{
          if (x<75000){paste0("A_",as.character((ceiling(x/
```

```r
                               ↪ 15000))*15000))}else{
                  if (x<150000){paste0("A_",as.character((ceiling(x/
                      ↪ 50000))*50000))}else{
                     if (x<250000){paste0("A_",as.character((ceiling(x/
                         ↪ 100000))*100000))}else{
                        if (x<1000000){paste0("A_",as.character((ceiling
                           ↪ (x/250000))*250000))}else{
                        "A_1250000"}}}}}}}}
)
options(scipen=0)

# xx<- 999999
# if (xx<1000000){out <- paste0("A_",as.character((floor(xx/
    ↪ 250000))*250000))}


FingerprintsnewMin_Marked <- sapply(qt1$Min_Marked, function(x)
  if (x<4) {"A_0"}else{
    if (x<55){paste("A",as.character((ceiling(x/4))*4), sep="_")
       ↪ }
  }
)


FingerprintsnewMatch_Min <- sapply(qt1$Match_Min, function(x)
  if (x<1) {"A_0"}else{
    if (x<7){paste("A",as.character((floor(x/2))*2), sep="_")}
       ↪ else{
      if (x<35){paste("A",as.character((floor(x/5))*5), sep="_")
         ↪ }}}
)

Delta <- sapply(qt1$Delta, function(x)
if (x<10) {"A_0"}else{
if (x<50){paste0("A_",as.character((ceiling(x/30))*30))}else{
if (x<500){paste0("A_",as.character((ceiling(x/100))*100))}else{
if (x<2000){paste0("A_",as.character((ceiling(x/1000))*1000))}
    ↪ else{
if (x<8000){paste0("A_",as.character((ceiling(x/4000))*4000))}
    ↪ else{
if (x<30000){paste0("A_",as.character((ceiling(x/10000))*10000))
    ↪ }else{
```

```r
if (x<90000){paste0("A_",as.character((ceiling(x/50000))*50000))
    ↪ }else{
if (x<300000){paste0("A_",as.character((ceiling(x/150000))*
    ↪ 150000))}else{
if (x<900000){paste0("A_",as.character((ceiling(x/600000))*
    ↪ 600000))}else{
if (x<2000000){paste0("A_",as.character((ceiling(x/1000000))*
    ↪ 1000000))}else{
if (x<5000000){paste0("A_",as.character((ceiling(x/4000000))*
    ↪ 4000000))}else{
if (x<15000000){paste0("A_",as.character((ceiling(x/10000000))*
    ↪ 10000000))}else{
if (x<20000000){paste0("A_",as.character((ceiling(x/1550000))*
    ↪ 1550000))}else{
"A_25000000"}}}}}}}}}}}}

)

Data<-subset(Fingerprints, select=c(Match_Status)) # add IDs


Match_Min <- FingerprintsnewMatch_Min
Match_Score <- FingerprintsnewMatch_Score
Min_Marked <- FingerprintsnewMin_Marked
Delta <- Delta

Data<-cbind(Data, Match_Min, Match_Score, Min_Marked, Delta)

write.csv(Data, "C:/Users/Shreya/Desktop/TEST_BN_(delta)/Known_
    ↪ Latents_Data_BNValid.csv", row.names = FALSE)
Data <- read.csv("C:/Users/Shreya/Desktop/TEST_BN_(delta)/Known_
    ↪ Latents_Data_BNValid.csv")



## Build the Network================================
# Define States
    ↪ ====================================================
    ↪
st.Match_Status <- toupper(unique(Data$Match_Status))
st.Match_Score <- toupper(unique(Data$Match_Score))
st.Min_Marked <- toupper(unique(Data$Min_Marked))
```

```
st.Match_Min <- toupper(unique(Data$Match_Min))
st.Delta <- toupper(unique(Data$Delta))

# Create new Network
    ↪ ═══════════════════════════════════════════════════════════
    ↪
BN9MM <- CreateNetwork("BN9MM")

# help("RNetica-package")
NetworkTitle(BN9MM) <- "BN_for_the_interpretation_of_AFIS_Data"
NetworkComment(BN9MM) <- "SS_Kamath_&_KB_Morris"

# Create nodes
    ↪ ═══════════════════════════════════════════════════════════
    ↪
Match_Status <- NewDiscreteNode(BN9MM, "Match_Status", states=st
    ↪ .Match_Status)
Match_Score <- NewDiscreteNode(BN9MM, "Match_Score", states=st.
    ↪ Match_Score)
Min_Marked <- NewDiscreteNode(BN9MM, "Min_Marked", states=st.Min
    ↪ _Marked)
Match_Min <- NewDiscreteNode(BN9MM, "Match_Min", states=st.Match
    ↪ _Min)
Delta <- NewDiscreteNode(BN9MM, "Delta", states=st.Delta)

# Change State Titles
    ↪ ═══════════════════════════════════════════════════════════
    ↪
NodeStateTitles(Match_Score) <- formatC(as.numeric(gsub("A_","",
    ↪ unique(Data$Match_Score))), width=7, flag="0")
NodeStateTitles(Min_Marked) <- formatC(as.numeric(gsub("A_","",
    ↪ unique(Data$Min_Marked))), width=2, flag="0")
NodeStateTitles(Match_Min) <- formatC(as.numeric(gsub("A_","",
    ↪ unique(Data$Match_Min))), width=2, flag="0")
NodeStateTitles(Delta) <- formatC(as.numeric(gsub("A_","",unique
    ↪ (Data$Delta))), width=2, flag="0")


NodeStateTitles(Match_Status) <- unique(Data$Match_Status)
#NodeStateTitles(Min_Marked) <- unique(Fingerprints$Min_Marked)

# Add links
```

```
    ↪ ══════════════════════════════════════════════════════════
    ↪
AddLink ( Match _ Status , Match _ Score )
AddLink ( Match _ Status , Match _ Min)
AddLink ( Min _ Marked , Match _ Min )
AddLink ( Match _ Min , Match _ Score )
AddLink ( Match _ Status , Delta )
AddLink ( Delta , Match _ Score )
AddLink ( Match _ Min , Delta )




# Add CPTs
    ↪ ══════════════════════════════════════════════════════════
    ↪

train <- Data
outfile <- tempfile (" train " , fileext =". cas ")
write . CaseFile ( train , outfile )
LearnCases ( outfile , list ( Match _ Score , Match _ Min , Min _ Marked ,
    ↪ Match _ Status , Delta ))

# Manipulate network
    ↪ ══════════════════════════════════════════════════════════
    ↪
CompileNetwork (BN9MM)

# Save the Network
    ↪ ══════════════════════════════════════════════════════════
    ↪
SetNetworkAutoUpdate (BN9MM, TRUE)
WriteNetworks (BN9MM, "C: / Users / Shreya / Desktop/TEST _BN _ ( delta )/BN
    ↪ _Train . dne ")

TesterSK <- read . csv ("C: / Users / Shreya / Desktop/TEST _BN _ ( delta )/
    ↪ Unknown _ Latents _ Data _BNValid . csv ")

#j <- 1

PosteriorYes <- c ()
PosteriorNo <- c ()
PriorYes <- c ()
```

55

```r
PriorNo <- c()


for (j in 1:length(TesterSK$Match_Status)){

  Priors <-NodeBeliefs(Match_Status)
  PriorYes[j] <- as.numeric(Priors[1])
  PriorNo[j] <- as.numeric(Priors[2])

  MinMatch <- as.character(TesterSK$Match_Min[j])
  MatchScore <- as.character(TesterSK$Normalized_Match_Score[j])

  NodeFinding(Match_Min) <- MinMatch
  NodeFinding(Normalized_Match_Score) <- MatchScore

  Posteriors<-NodeBeliefs(Match_Status)
  PosteriorYes[j] <- as.numeric(Posteriors[1])
  PosteriorNo [j] <-as.numeric(Posteriors[2])

  RetractNodeFinding(Match_Min)
  RetractNodeFinding(Normalized_Match_Score)
}




LR <- (PosteriorYes*PriorNo)/(PosteriorNo*PriorYes)
LLR <- log10(LR)
# LR
# LLR

TesterSK <- cbind(TesterSK, PosteriorYes, PosteriorNo, PriorYes,
    ↪   PriorNo, LR, LLR)

write.csv(TesterSK,"C:/Users/Shreya/Desktop/TEST_BN_(delta)/
    ↪ TesterSK.csv", row.names=FALSE)

#========================

RetractNetFindings(BN9MM)

DeleteNetwork(BN9MM)
```

```
rm(BN9MM)
is.active(BN9MM)
is.active(Match_Status)

# Plots═══════════════════════════

TesterSK <- read.csv("C:/Users/Shreya/Desktop/TEST_BN_(delta)/
    ↪ TesterSK.csv")


mainDir <- "C:/Users/Shreya/Desktop/TEST_BN_(delta)/Plots/"

LR.KM <- subset(TesterSK, Match_Status =="Yes", select = "LR")
LR.KM <- LR.KM[['LR']]
LR.KNM <- subset(TesterSK, Match_Status =="No", select = "LR")
LR.KNM <- LR.KNM[['LR']]

PriorYes <- TesterSK$PriorYes[1]
PriorNo <- TesterSK$PriorNo[1]

LR.H1.exp = LR.KM
LR.H2.exp = LR.KNM

log.LR.H1.exp = log10(LR.H1.exp)
log.LR.H2.exp = log10(LR.H2.exp)

min = min(log.LR.H1.exp, log.LR.H2.exp)
max = max(log.LR.H1.exp, log.LR.H2.exp)

x.range.data = data.frame(c(min-1,log.LR.H1.exp, log.LR.H2.exp,
    ↪ max+1)) #rbind(min-1,log.LR.H1.exp, log.LR.H2.exp, max+1)
x.range = x.range.data[order(x.range.data),1]

log.LR.H1.exp <- data.frame(log.LR.H1.exp)
log.LR.H2.exp <- data.frame(log.LR.H2.exp)


title <- "Title_goes_here"

# Tippett Plots═══════════════════════════
Tippett.2 = matrix(0, nrow = length(x.range), ncol = 1)
```

```r
Tippett.1 = matrix(0, nrow = length(x.range), ncol = 1)

for (bb in 1:length(x.range)){
  Tippett.2[bb] = length(which(log.LR.H2.exp > x.range[bb]))/
      ↪ nrow(log.LR.H2.exp)*100
  Tippett.1[bb] = length(which(log.LR.H1.exp > x.range[bb]))/
      ↪ nrow(log.LR.H1.exp)*100
}

false.positives = round(length(which(log.LR.H2.exp > 0))/nrow(
    ↪ log.LR.H2.exp)*100,2)
false.negatives = round(length(which(log.LR.H1.exp < 0))/nrow(
    ↪ log.LR.H1.exp)*100,2)

par(mfrow=c(1,1))
# Save Tippett plot
fname <- paste0(mainDir,"Tippett.png")
png(fname,width=1200,height=1200,res=144)
plot(x.range, Tippett.2, type="s", xlab=expression(paste(log
    ↪ [10],"LR␣greater␣than")), ylab="Proportion␣of␣cases␣[%]",
    ↪ xlim=c(-3,3), ylim=c(0,100), lty=3, main = title)
par(new=TRUE)
plot(x.range, Tippett.1, type="s", xlab=expression(paste(log
    ↪ [10],"LR␣greater␣than")), ylab="Proportion␣of␣cases␣[%]",
    ↪ xlim=c(-3,3), ylim=c(0,100))
legend("bottomleft", c(expression(paste("true-",H[d],"␣LR␣values
    ↪ ")), expression(paste("true-",H[p],"␣LR␣values"))), lty=c
    ↪ (3,1), bty="n")
abline(v=0, col="gray", lty=4)
dev.off()

# Histogram===========================================
breaks = 50
stepbins = (max-min)/breaks
xbars = seq(min, max, by=stepbins)

log.LR.H1.exp = log.LR.H1.exp[(log.LR.H1.exp < (xbars[length(
    ↪ xbars)]-stepbins/2)) & log.LR.H1.exp > (xbars[1]-stepbins/
    ↪ 2)]
log.LR.H2.exp = log.LR.H2.exp[(log.LR.H2.exp < (xbars[length(
    ↪ xbars)]-stepbins/2)) & log.LR.H2.exp > (xbars[1]-stepbins/
    ↪ 2)]
```

```
# Save Histogram
fname <- paste0(mainDir,"Histogram.png")
png(fname,width=1200,height=1200,res=144)
set = par(mfrow=c(2,1), mar=c(4,4,1,2))
hist(log.LR.H1.exp, breaks=xbars-stepbins/2, col="gray35", main=
    ↪ "", xlab="logLR")#, main = title)
legend("topleft", expression(paste(H[p]," true")), fill="gray35"
    ↪ )

hist(log.LR.H2.exp, breaks=xbars-stepbins/2, col="gray75", main=
    ↪ "", xlab="logLR")
legend("topright", expression(paste(H[d]," true")), fill="gray75
    ↪ ")
par(set)
dev.off()

par(mfrow=c(1,1))

# DET Plot==========================================
log.LR.H1.exp = log10(LR.H1.exp)
log.LR.H2.exp = log10(LR.H2.exp)

min = min(log.LR.H1.exp,log.LR.H2.exp)
max = max(log.LR.H1.exp,log.LR.H2.exp)

threshold.range = data.frame(c(min-1,log.LR.H1.exp,log.LR.H2.exp
    ↪ , max+1))
threshold = threshold.range[order(threshold.range),1]

false.positives = matrix(0, nrow = length(threshold), ncol = 1)
false.negatives = matrix(0, nrow = length(threshold), ncol = 1)

log.LR.H1.exp <- data.frame(log.LR.H1.exp)
log.LR.H2.exp <- data.frame(log.LR.H2.exp)

for (cc in 1:length(threshold)){
  tmp.threshold = threshold[cc]
  false.positives[cc] = length(which(log.LR.H2.exp > tmp.
      ↪ threshold))/nrow(log.LR.H2.exp)*100
  false.negatives[cc] = length(which(log.LR.H1.exp <= tmp.
      ↪ threshold))/nrow(log.LR.H1.exp)*100
```

59

```r
}

x = qnorm(false.positives/100)
y = qnorm(false.negatives/100)

x[which(x == -Inf)] = qnorm(0.000001)
y[which(y == -Inf)] = qnorm(0.000001)
x[which(x == Inf)] = qnorm(0.999999)
y[which(y == Inf)] = qnorm(0.999999)


fname <- paste0(mainDir,"DETplot.png")
png(fname,width=1200,height=1200,res=144)
plot(x, y, type="S", xlab="false_positives_[%]", ylab="false_
    ↪ negatives_[%]", xaxt="n", yaxt="n", xlim=c(qnorm(0.0001),
    ↪ qnorm(0.5)), ylim=c(qnorm(0.0001),qnorm(0.5)),  main =
    ↪ title)
axis.range = c(0.0001, 0.001, 0.01, 0.02, 0.05, 0.1, 0.2, 0.5,
    ↪ 1, 2, 5, 10, 20, 40, 50)
axis.gauss = qnorm(axis.range/100)
abline(h=axis.gauss, lty=3, col="gray")
abline(v=axis.gauss, lty=3, col="gray")
abline(a=0, b=1, lty="dashed", col="darkgray")
axis(side=1,at=axis.gauss,labels=axis.range)
axis(side=2,at=axis.gauss,labels=axis.range)
dev.off()

# ECE Function════════════════════════════════════
P.H1 = seq(from=0.01, to=0.99, by=0.01)
P.H2 = 1 - P.H1
a.priori.odds = P.H1/P.H2

LR.H1.exp <- data.frame(LR.H1.exp)
LR.H2.exp <- data.frame(LR.H2.exp)

N.H1 = nrow(LR.H1.exp)
N.H2 = nrow(LR.H2.exp)

set = c("null","exp","cal")


fname <- paste0(mainDir,"ECEplot.png")
```

```
png(fname, width=1200, height=1200, res=144)


for (dd in 1:length(set)){
  if(dd == 1){ ## LR for null method
    LR.H1 = rep(1, times=N.H1)
    LR.H2 = rep(1, times=N.H2)
  }

  if(dd == 2){ ## experimental LR
    LR.H1 = as.matrix(LR.H1.exp)
    LR.H2 = as.matrix(LR.H2.exp)
    LR.H1[which(LR.H1 < 10^-20)] = 10^-20
    LR.H2[which(LR.H2 < 10^-20)] = 10^-20
    LR.H1[which(LR.H1 > 10^20)] = 10^20
    LR.H2[which(LR.H2 > 10^20)] = 10^20
  }

  if(dd == 3){  ## calibrated LR according to PAV
    require("isotone")

    LR.H1.H2.exp = c(LR.H1,LR.H2)
    indices = order(LR.H1.H2.exp)
    LR.H1.H2.exp.sorted = sort(LR.H1.H2.exp)

    posterior.prob = c(rep(1, times=N.H1), rep(0, times=N.H2))
    posterior.prob.sorted = posterior.prob[indices]

    posterior.H1.H2.cal = data.frame(gpava(LR.H1.H2.exp.sorted,
        ↪ posterior.prob.sorted)[1])
    LR.H1.H2.cal = posterior.H1.H2.cal/(1-posterior.H1.H2.cal)/(
        ↪ N.H1/N.H2)
    LR.H1 = LR.H1.H2.cal[which(indices %in% c(1:N.H1)),]
    LR.H2 = LR.H1.H2.cal[which(indices %in% (N.H1+1):(N.H1+N.H2)
        ↪ ),]
    LR.H1[which(LR.H1 < 10^-20)] = 10^-20
    LR.H2[which(LR.H2 < 10^-20)] = 10^-20
    LR.H1[which(LR.H1 > 10^20)] = 10^20
    LR.H2[which(LR.H2 > 10^20)] = 10^20
  }

  penalty.H1 = 0
```

```r
  penalty.H2 = 0

  for (ff in 1:N.H1){
    a = -log2(LR.H1[ff]*a.priori.odds/(1+LR.H1[ff]*a.priori.odds
      ↪ ))
    penalty.H1 = penalty.H1 + a
  }

  for (gg in 1:N.H2){
    b = log2(1 + LR.H2[gg]*a.priori.odds)
    penalty.H2 = penalty.H2 + b
  }

  ECE= P.H1/N.H1*penalty.H1 + P.H2/N.H2*penalty.H2
  col = c("black","red","blue")
  lty = c(3,1,2)
  if (dd %in% c(2,3)) {par(new=TRUE)}

  plot(log10(a.priori.odds), ECE, xlim=c(-2,2), ylim=c(0,1),
    ↪ type="l", col=col[dd], lty=lty[dd], xlab = expression(
    ↪ paste("prior log"[10],"(Odds)")), main = title)
  abline(v=0, lty=2, col="gray")
  legend("topleft", c("null method with LR=1", "experimental LR"
    ↪ , "calibrated LR after PAV"), col=col, lty=lty, bty="n",
    ↪ cex=0.8)
}
dev.off()
```

B. R Script for the Combined ROC Curve

```r
install.packages("pROC")
library(pROC)
library(ROCR)

setwd("C:/Users/Shreya/Desktop/Thesis/Final_Data/ROC_Curves")
getwd()
Data=read.csv("Known_Search_Fingers_Fixed.csv")

png("ROC_Combined_Fixed.png", width = 4, height = 4, units = 'in
    ↪ ', res = 300)
roc1 = plot(roc(Data$Match_Status,Data$Match_Score),  main='
    ↪ Combined_ROC_Curves', col = "blue",xlab='False_Positive_
    ↪ Rate', ylab='True_Positive_Rate')
roc2 = plot(roc(Data$Match_Status, Data$Match_Min), col = "red",
    ↪  add = TRUE,lty=2)
legend("bottomright", legend = c("Match_Score_(AUC)_=_0.837",
                                 "Match_Minutiae_(AUC)_=_0.871",
                                 expression(paste(p,"_","=","_"
                                     ↪ ,1.758,x,10^-6)),
                              expression(paste( alpha, "_=_",
                                  ↪ 0.05))), col=c("blue", "red2"
                                  ↪ ,"white","white"),box.col="
                                  ↪ black", cex=0.5, lty = 1:2)
dev.off()
roc.test(roc1, roc2)
roc.test(roc1, roc2, method="delong")
```

C. R script for Error Rate Plots

```
setwd("C:/Users/Shreya/Desktop/Thesis/Final_Data/Error_Rates")
getwd()
Finger=read.csv("Known_Search_Fingers_Fixed.csv")
names(Finger)

#========================Match Min Plot===============
#Rates vs. cutoff
pred=prediction(Finger$Match_Min, Finger$Match_Status)
perf.fnr=performance(pred,"fnr","fpr")
perf.fpr=performance(pred,"fpr","fnr")
perf.acc=performance(pred, "acc")

# now converting S4 class to vector
fnr = unlist(slot(perf.fnr, "y.values"))
fpr = unlist(slot(perf.fpr, "y.values"))
acc = unlist(slot(perf.acc, "y.values"))

cutoff = unlist(slot(perf.fnr, "alpha.values"))

png("ROC_Match_Minutiae_Fixed.png",width = 4, height = 4, units
    ↪ = 'in', res = 300)
plot(fpr~cutoff,type="l", xlab="Match_Minutiae", ylab="Rate",
    ↪ xlim=c(0,35), main="Error_Rate\nMatch_Minutiae", col="red")
lines(fnr~cutoff,col="blue",lty=8,type="l")
legend(24,0.9, legend=c("False_Positive_Rate", "False_Negative_
    ↪ Rate", "EER_=_21%"), col=c("red", "blue", "white"), box.col
    ↪ ="black",lty=c(1,2), cex = 0.4)

#x axis: 4.710857
#y axis: 0.2132065
locator()
dev.off()
```

```
#after using locator, click on exit to get value of EER


#========================Match Score Plot==============
pred=prediction(Finger$Match_Score_log, Finger$Match_Status)
perf.fnr=performance(pred,"fnr","fpr")
perf.fpr=performance(pred,"fpr","fnr")
perf.acc=performance(pred, "acc")

# now converting S4 class to vector
fnr = unlist(slot(perf.fnr, "y.values"))
fpr = unlist(slot(perf.fpr, "y.values"))
acc = unlist(slot(perf.acc, "y.values"))

cutoff = unlist(slot(perf.fnr, "alpha.values"))

png("ROC_Match_Score_Fixed.png",width = 4, height = 4, units = '
    ↪ in', res = 300)
plot(fpr~cutoff,type="l", xlab="Log_of_Match_Score", ylab="Rate"
    ↪ ,xlim=c(1.70, 6.0), main="Error_Rate\nMatch_Score", col="
    ↪ red")
lines(fnr~cutoff,col="blue",lty=8,type="l")
legend(4.6,0.55, legend=c("False_Positive_Rate", "False_Negative
    ↪ _Rate", "EER_=_25%"), col=c("red", "blue", "white"), box.
    ↪ col="black",lty=c(1,2), cex = 0.4)
#x axis: 3.768671
#y axis: 0.2477137
locator()
dev.off()
```

D. Match Minutiae vs Delta Score for Individual Runs
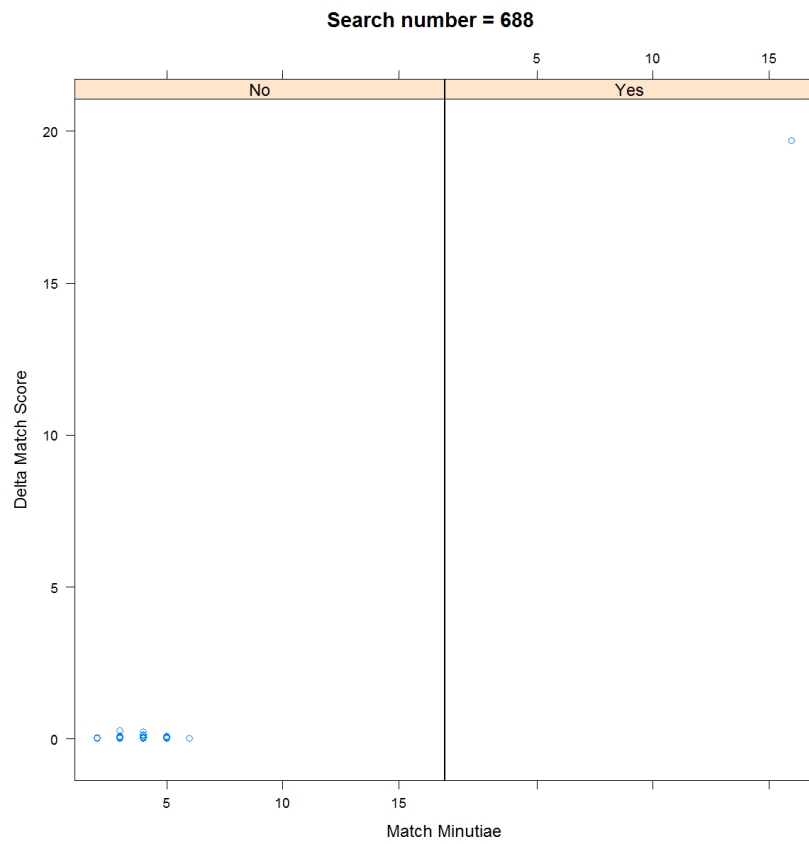
Figure D.1: Match Minutiae vs Delta Match Score for Search Result 688
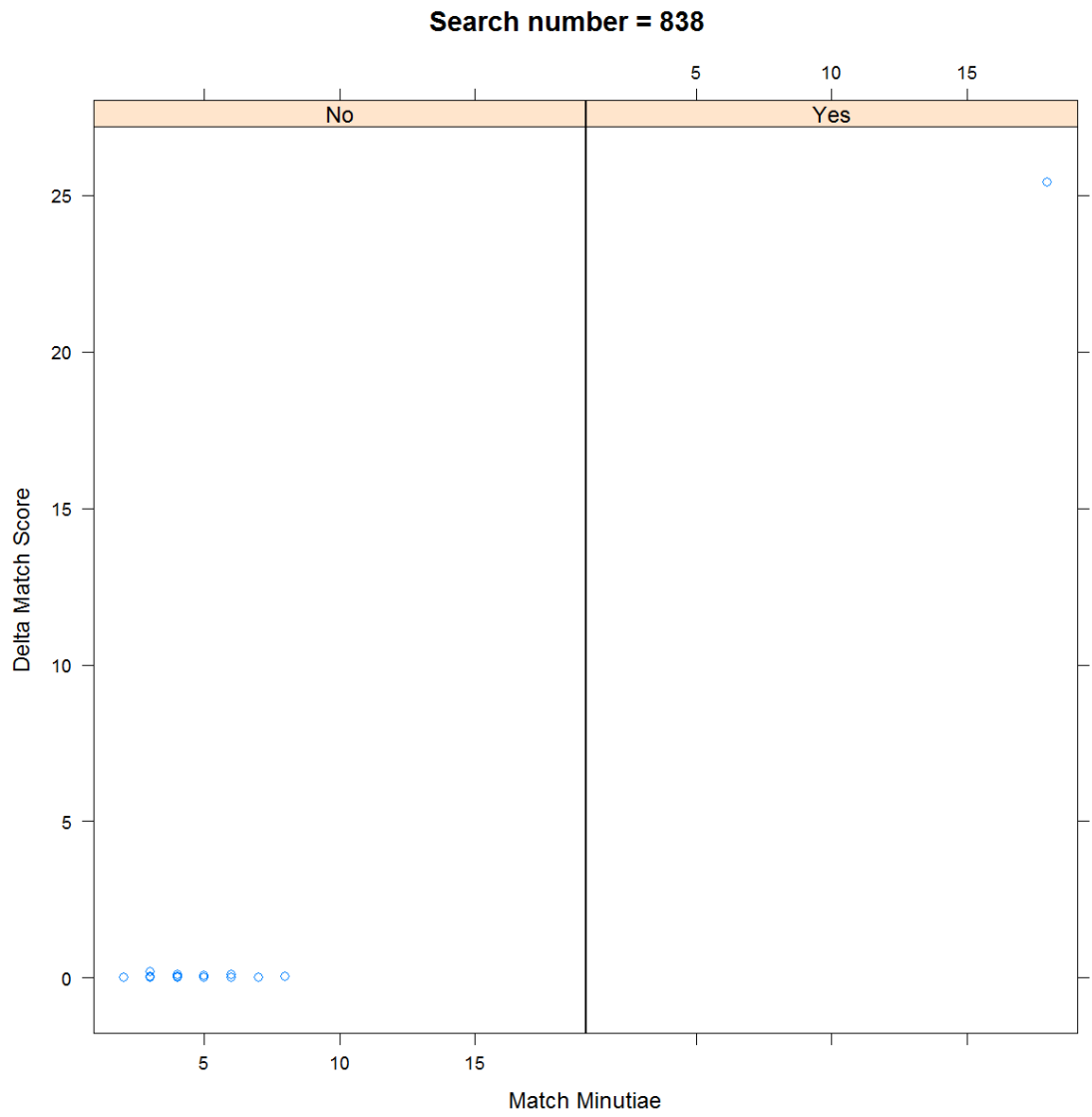
Figure D.2: Match Minutiae vs Delta Match Score for Search Result 838
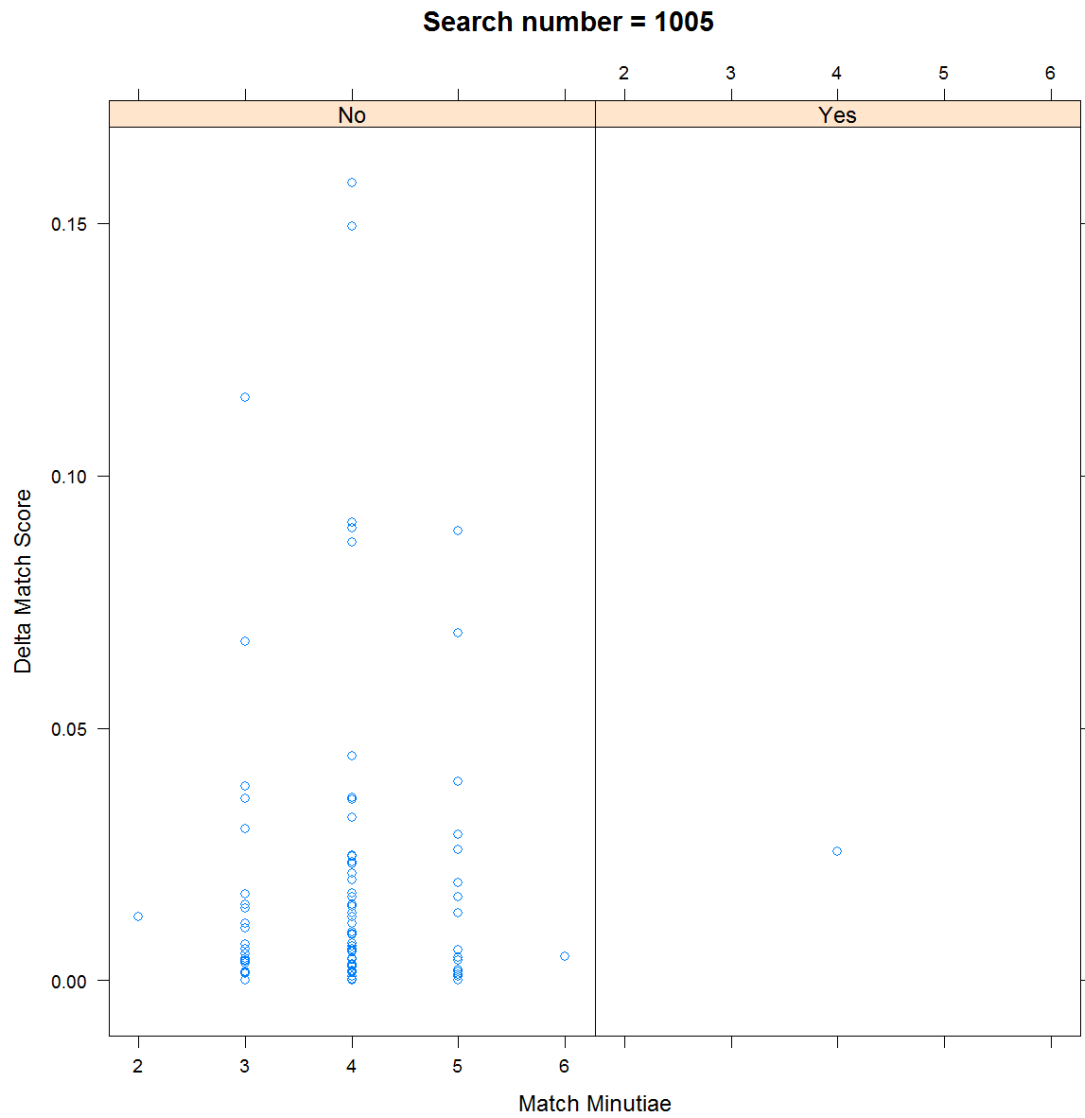
Figure D.3: Match Minutiae vs Delta Match Score for Search Result 1005
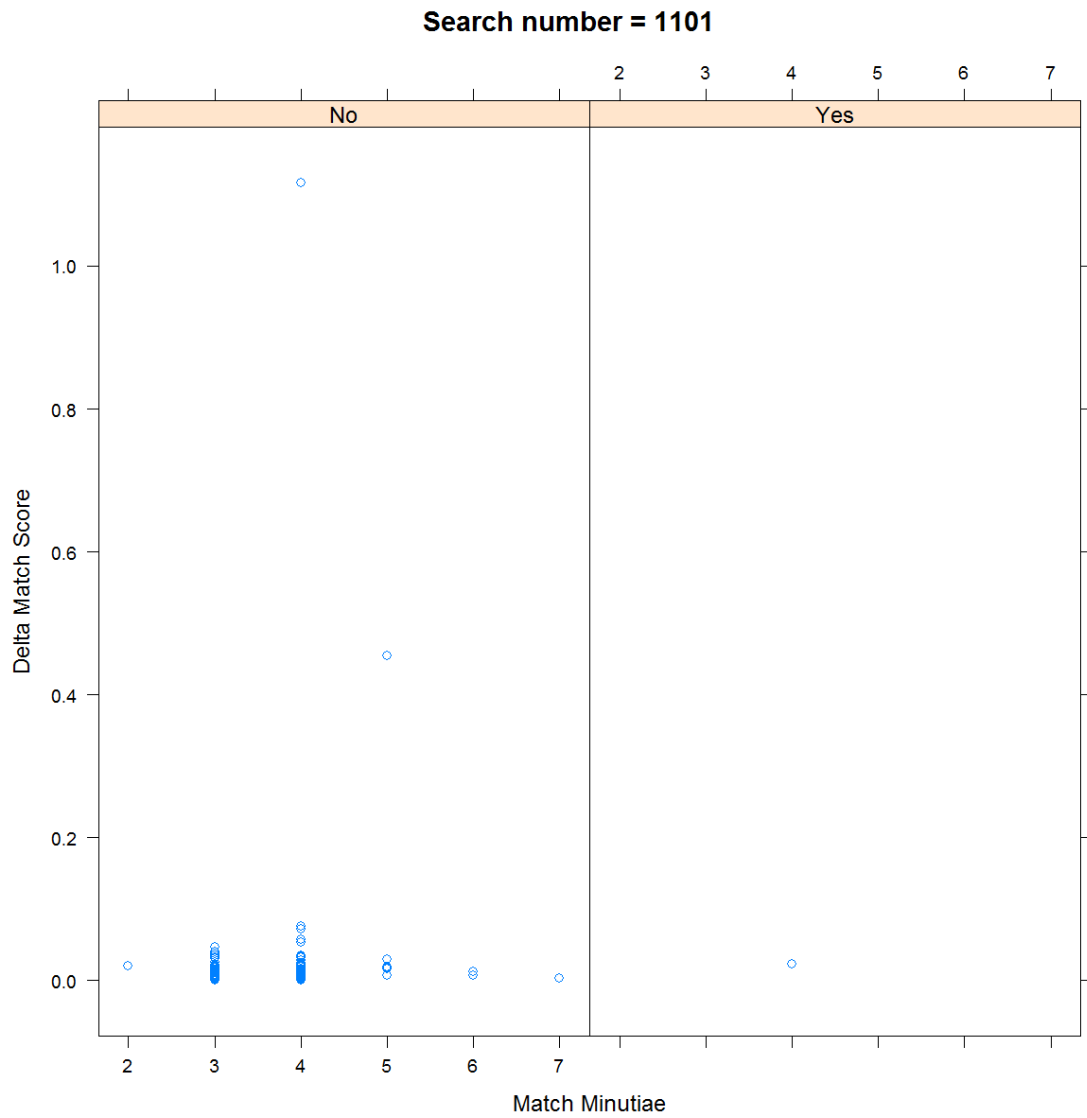
Figure D.4: Match Minutiae vs Delta Match Score for Search Result 1101

E. Test Print Results for Different Development Methods

Table E.1: Test Prints Development Methods

True Match	Highest Match Score	Highest Min Marked	Highest Match Min
Black	3327600	50	40
Nin	59690	25	10
Cyano	17160	40	8
Non-Match	**Highest Match Score**	**Highest Min Marked**	**Highest Match Min**
Black	73944	50	13
Nin	11074	50	7
Cyano	15708	40	8

F. Tippett, Histogram and ECE

Plots Related to Match Minutiae
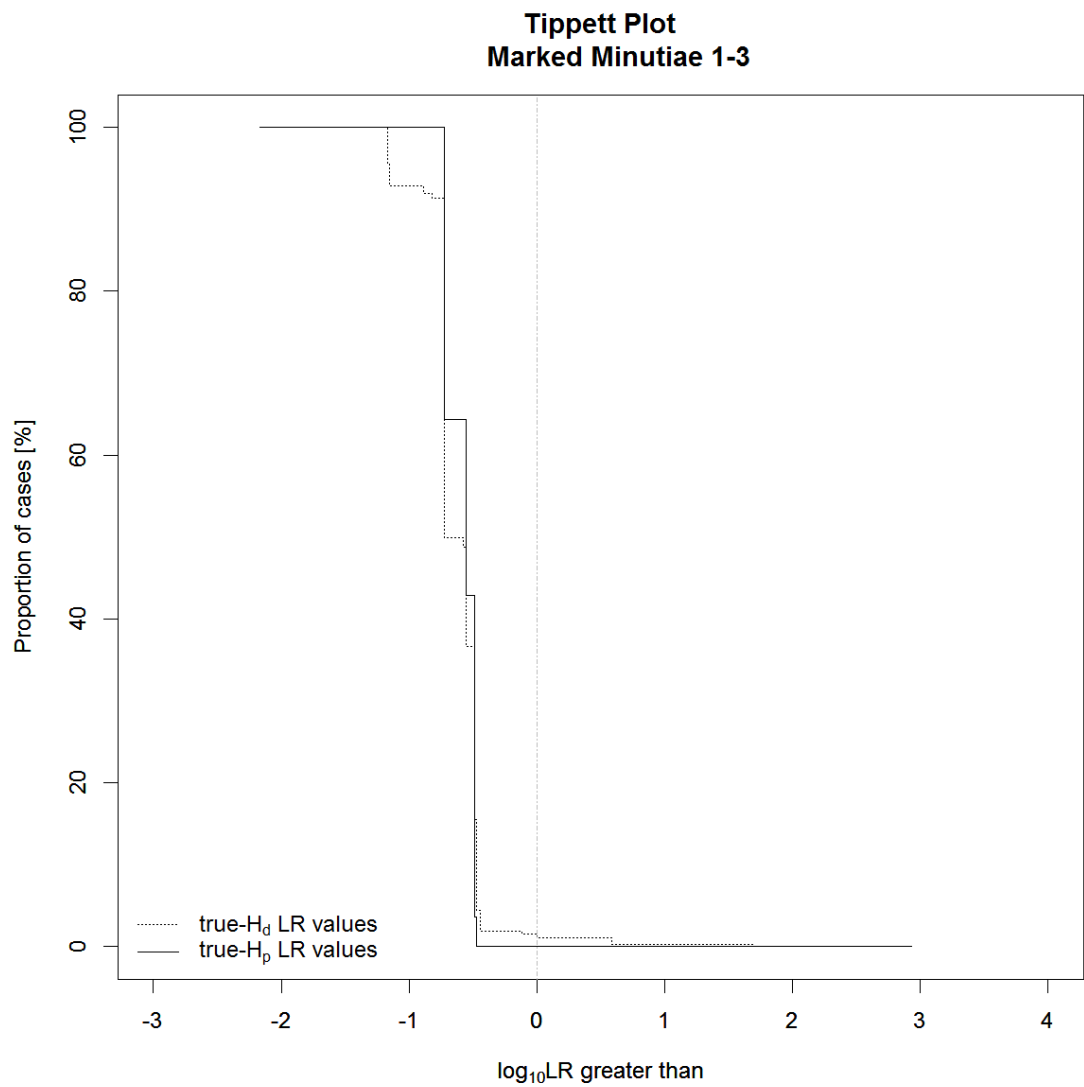
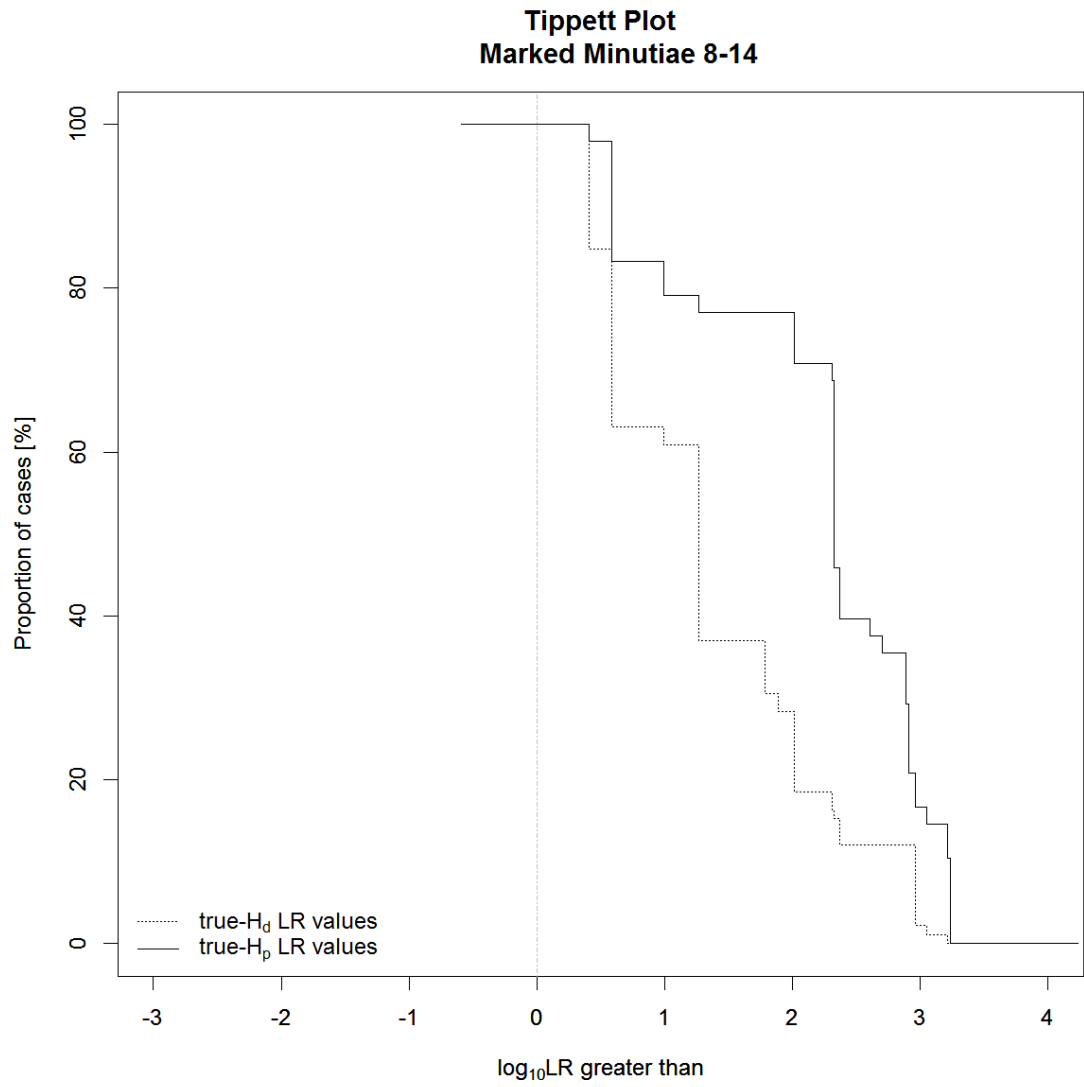Figure F.1: Tippett plot with Match Minutiae 1-3
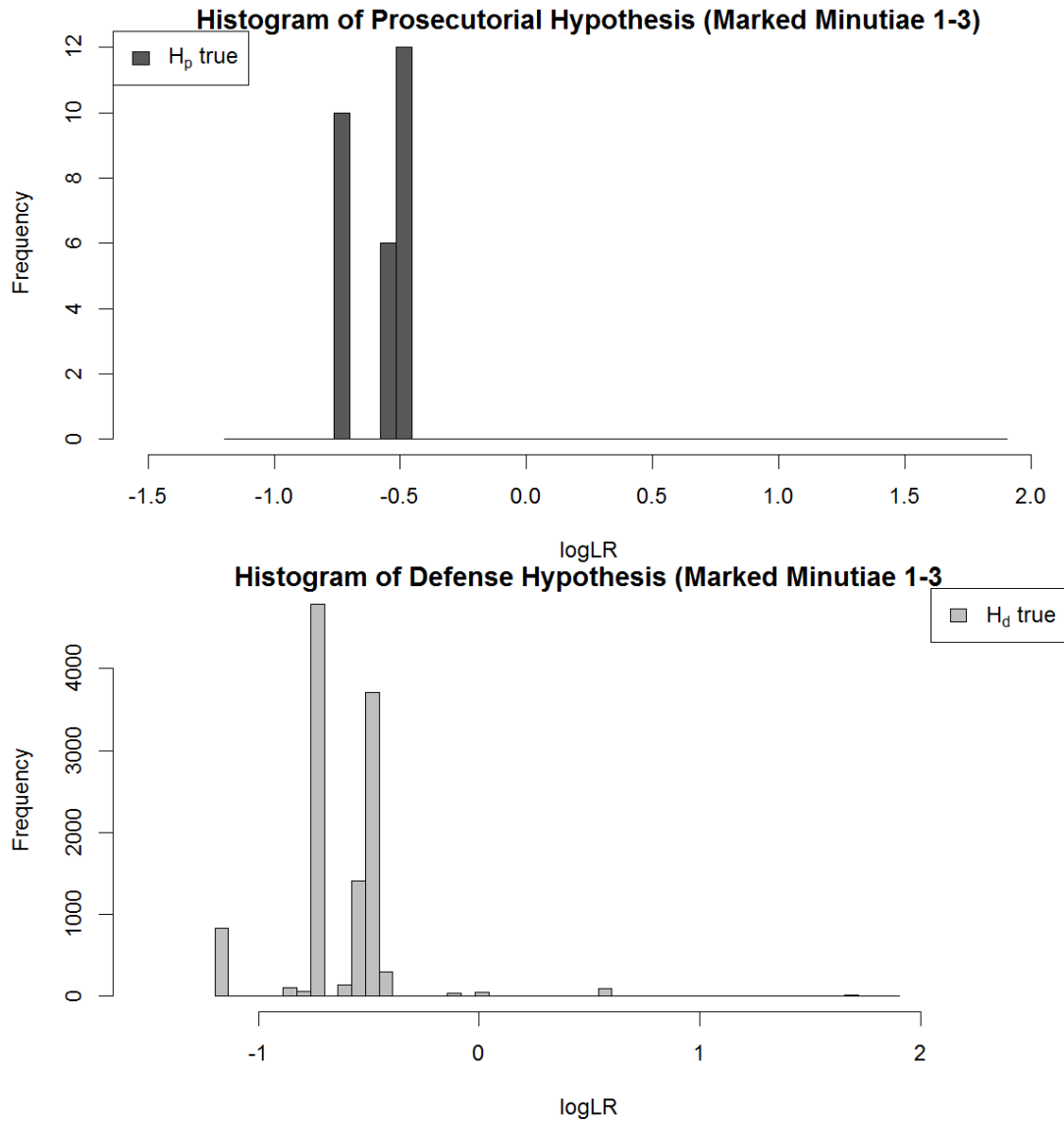
Figure F.2: Histogram with Match Minutiae 1-3
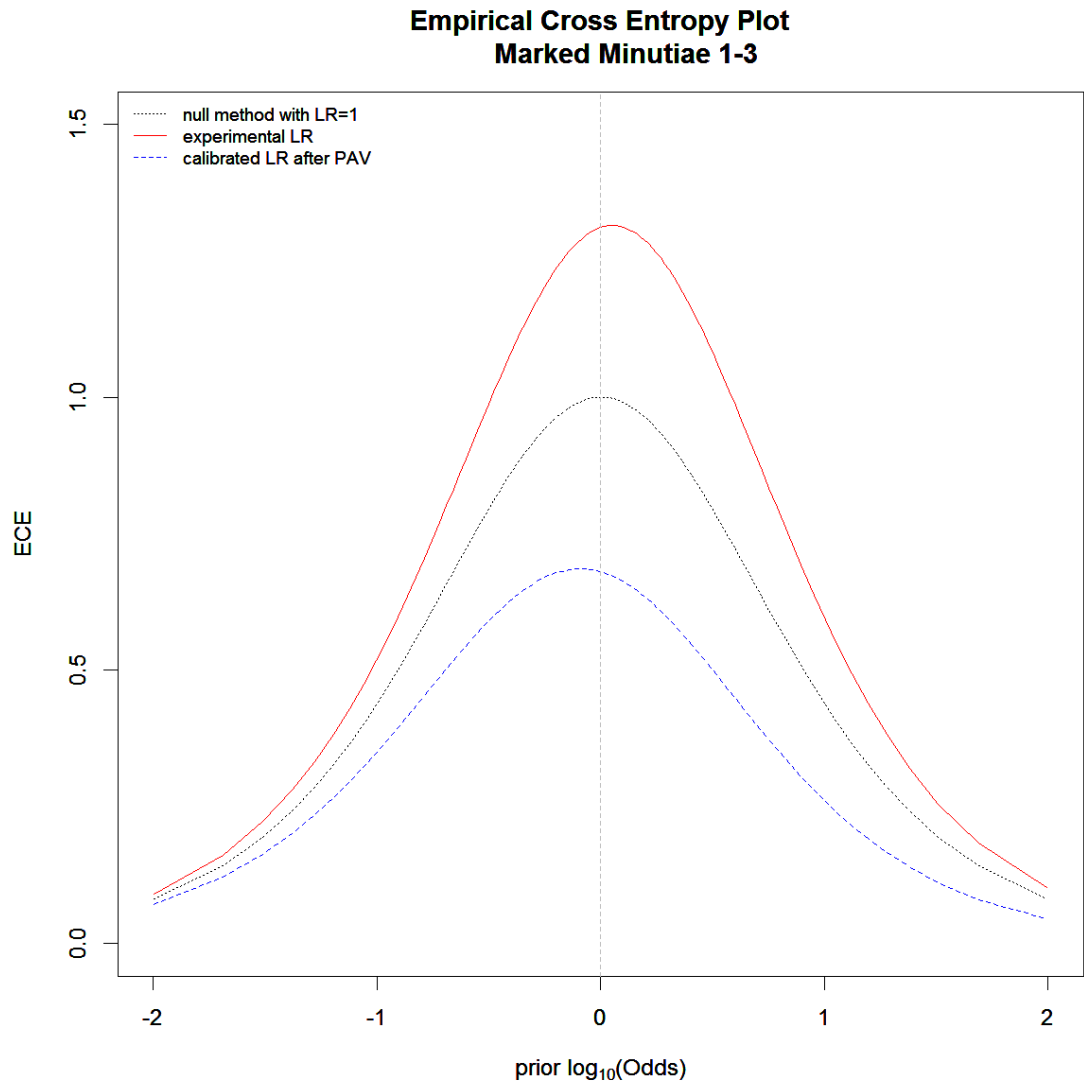
Figure F.3: ECE plot with Match Minutiae 1-3
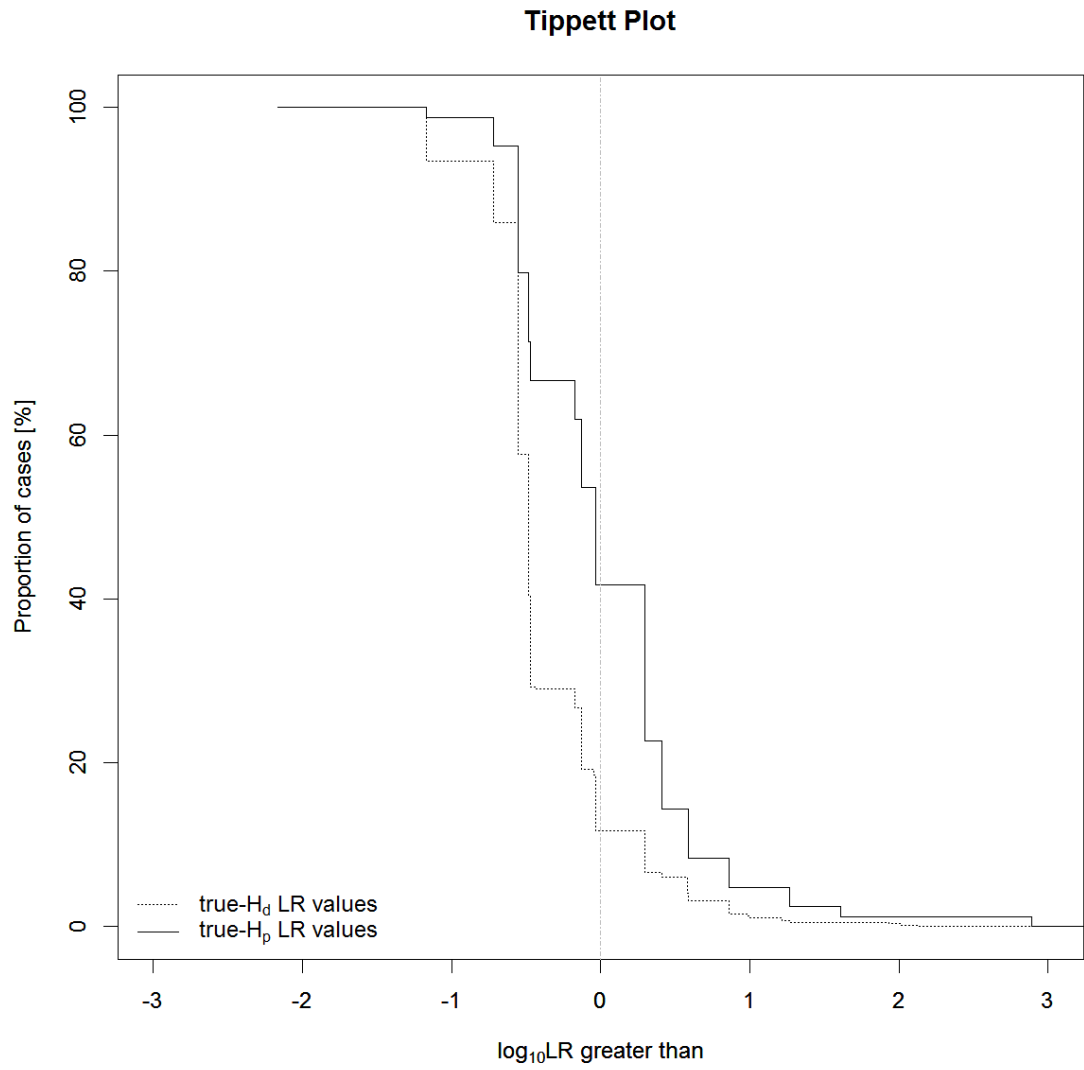
Tippett Plot

Figure F.4: with Match Minutiae 4-7

75
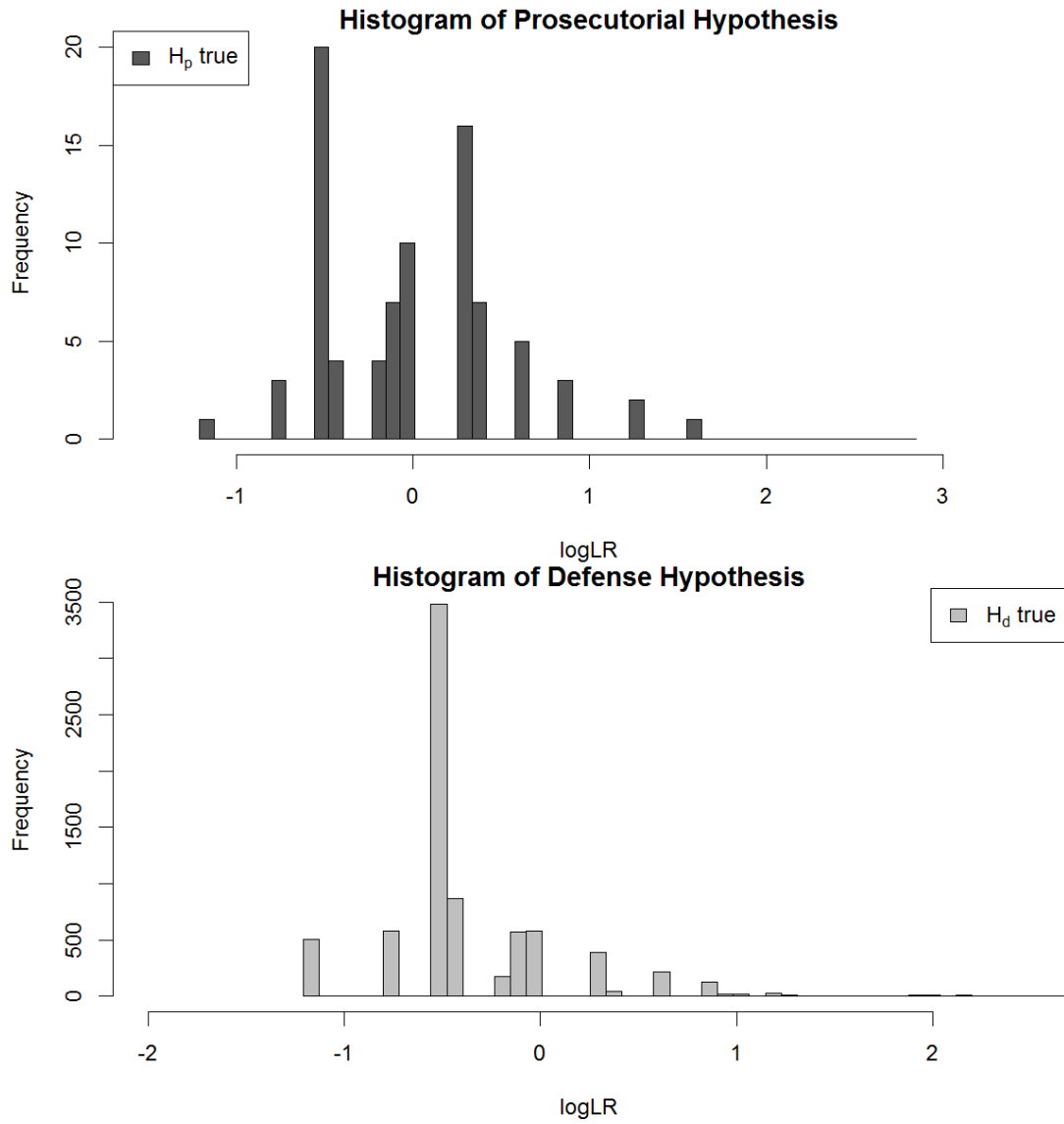
Figure F.5: Histogram with Match Minutiae 4-7
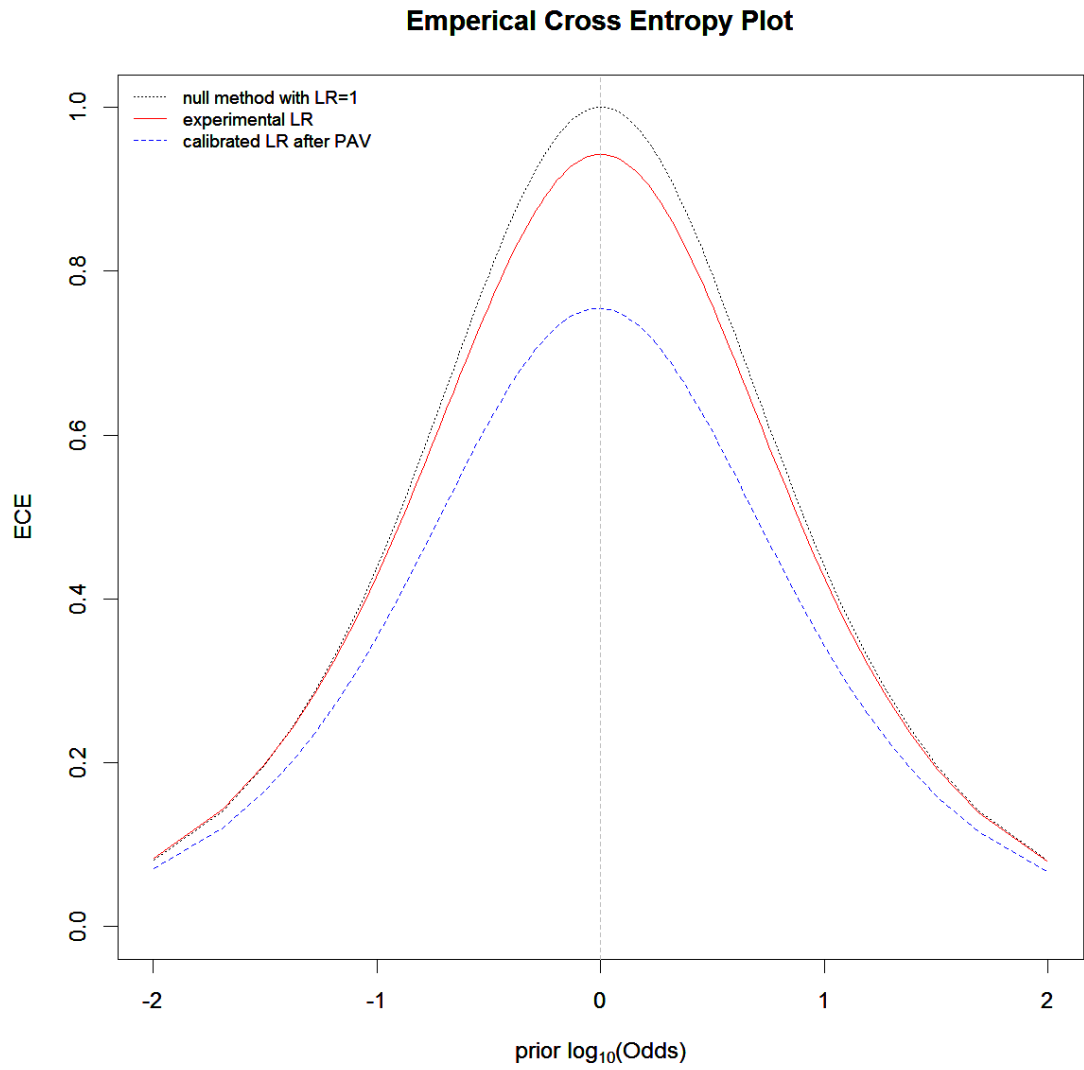
Figure F.6: ECE plot with Match Minutiae 4-7
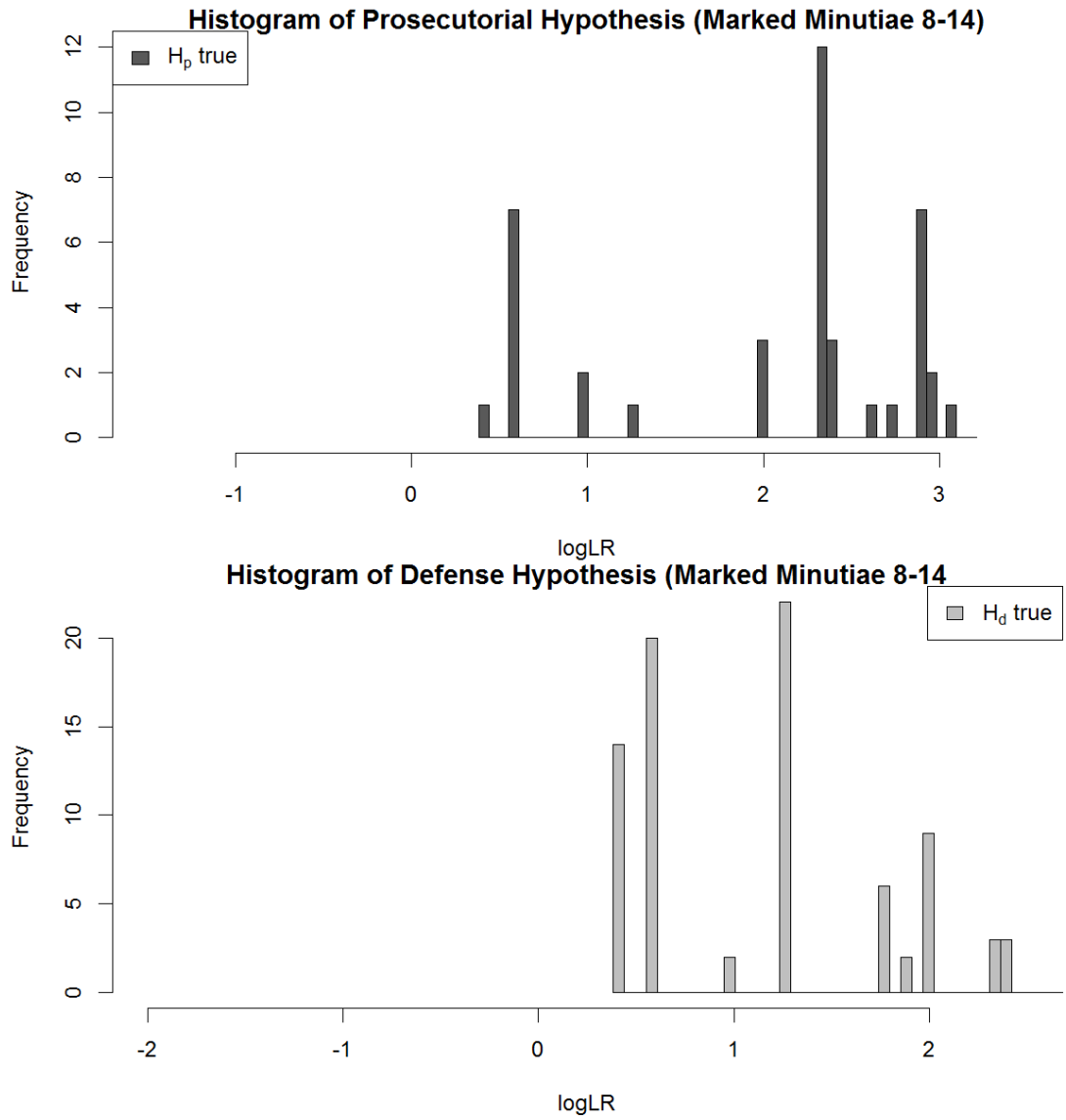
Figure F.7: Tippett plot with Match Minutiae 8-14
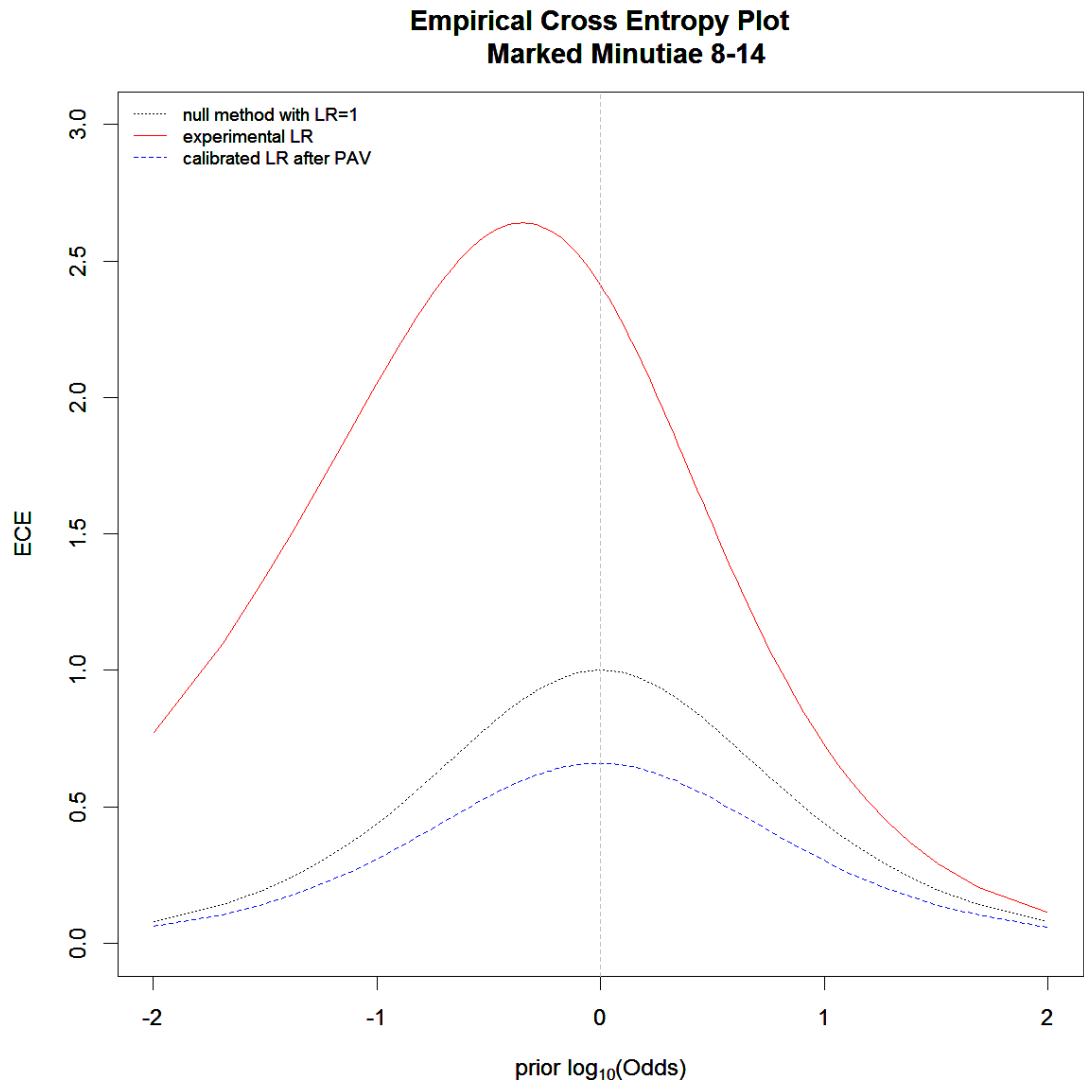
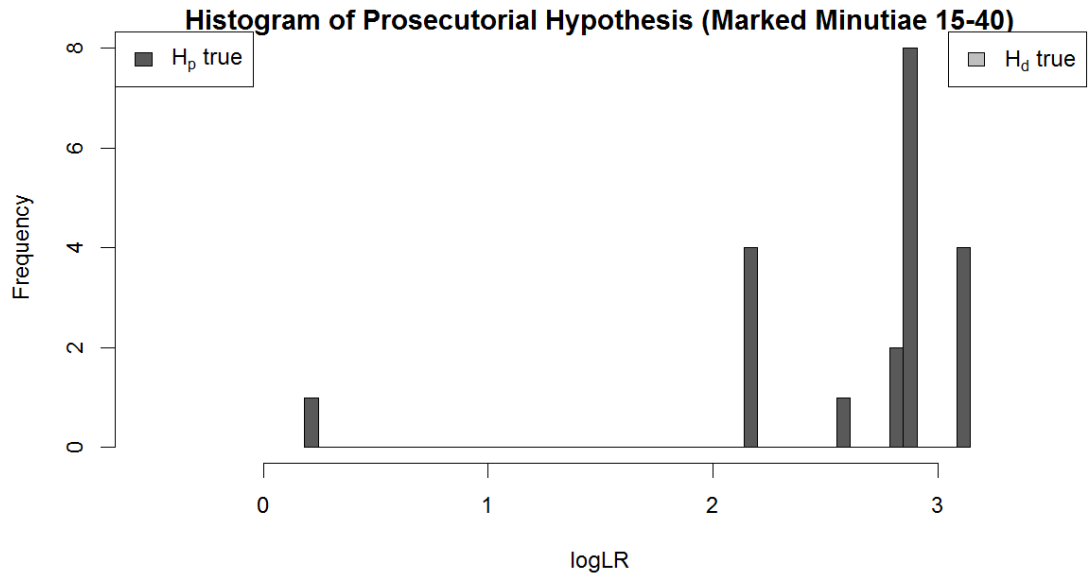Figure F.8: Histogram with Match Minutiae 8-14

Figure F.9: ECE plot with Match Minutiae 8-14

Figure F.10: Histogram with Match Minutiae 15-40