Adversarial biometric attacks on mobile devices

Valerie Ding
Dept. of Computer Science
Stanford University
dingv@stanford.edu

Stephanie Dong
Dept. of Computer Science
Stanford University
sxdong11@stanford.edu

Jonathan Li
Dept. of Computer Science
Stanford University
johnnyli@stanford.edu

Abstract

We develop clustering algorithms and Bayesian graphical models to spoof human biometric patterns in mobile inputs. We frame this as a targeted adversarial attack generation problem, and the applications are twofold. First, by developing methods to generate a mass of adversarial examples, we can develop more robust discriminatory classifiers for enhanced security in sensitive environments such as biometric identification on smartphones. Second, our vector mapping heuristics transform generated examples to time-sequence data, which can be employed to augment sparse datasets and develop emulators of fine-grained human behavior.

1 Introduction

Keystroke pattern and dynamics classification is an important application of machine learning to computer security and authentication. Much of existing literature focuses on traditional computer keyboard dynamics analysis, but the massive increase in popularity and computing power of mobile devices in the last ten years has spurred significant interest in biometric-focused authentication models for mobile devices.

Existing literature emphasizes need for more nuanced security protocols in personal devices. As mobile devices store increasingly valuable and confidential information, learning classifiers to detect fraud is becoming ever more applicable and important. However, it still remains to be seen how robust these user verification classifiers are against general attacks by malicious agents. To this end, we generate attacks against user verification classifiers using mobile biometrics data, performing white-hat analysis of various keyboard dynamics user verification schemes.

The 2016 Teh *et al.* survey of touch dynamics authentication on mobile devices [8] shows that probabilistic modeling, cluster analysis, decision trees, SVMs, and neural nets are the top most widely used in the decision making process (Figure 1).

2 Data and Features

2.1 Dataset and featurization

We used the MEU-Mobile KSD (Keystroke Dynamics) Data Set from the UCI Machine Learning Repository [1], containing 51 records for each of 56 subjects - 2856 records total - of haptic, momentum, and timing features measured of a common sequence (`.tie5Roanl`) typed on a Nexus 7 mobile device. There are 71 features monitored, characterized by the attributes `Hold`, `Up-Down`, `Down-Down`, `Pressure`, `Finger-Area`, `Average Hold`, `Average Pressure`, and `Average Area`.

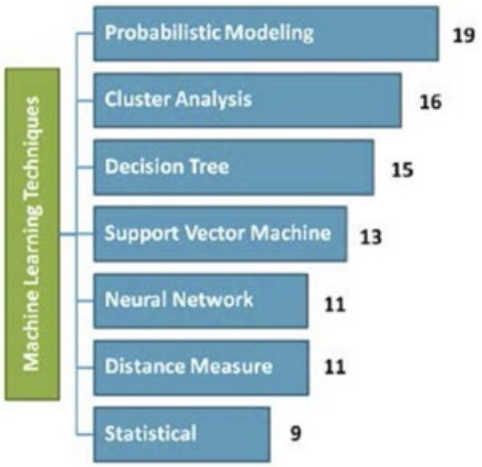


Figure 1: Machine learning techniques vs. the number of papers that employed them.

Hold .	Hold t	Hold i	Hold e	Hold Shift	Hold 5	Hold Shift
89	92	64	85	123	82	70
90	88	99	83	123	101	81
87	90	83	65	79	73	96
71	81	62	72	83	94	89

Table 1. Cross section of features from first four user typing sequences.

2.2 Data preprocessing

The user verification classifiers will be trained on the same dataset from which we generate attacks, with one key difference being we do not remove Subject labels from the training data.

To prepare data for training our comparison models, we concatenate each record with every other record to form a new concatenated comparison vector where the binary label is whether or not the record comes from a different user. This generates on the order of 8 million data points to then feed into our learning algorithms.

We implemented a flexible resampling framework that can utilize a variety of undersampling and oversampling methods to undersample the majority class and oversample the minority class. This ensures parity between labels of different user and same user in the training data. In our work, we undersampled the majority class with random undersampling, which randomly selects examples from the majority class. Our implementation does so without replacement.

3 Methods and models

3.1 Spoof attack task

A spoof is successful if it manages to trick the discriminatory classifier that it is an authentic user. We employ a combination of differential privacy policy and model emphasis on generalisability across users to develop targeted attacks without specific knowledge of a particular user's patterns, but insight into structure and different data patterns of human touch dynamics behavior on mobile devices.

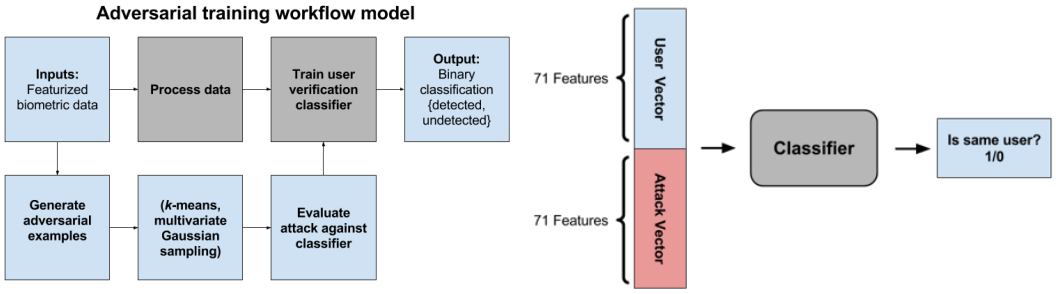


Figure 2: Framework to evaluate attack success and generating spoofs to trick discriminatory classifiers

3.2 *k*-means clustering

We implemented a variational *k*-means clustering framework initialized with random seed, varying on *k*. The *k* centroids determined upon convergence become the basis for our adversarial example generation.

3.3 Multivariate Gaussian distribution models

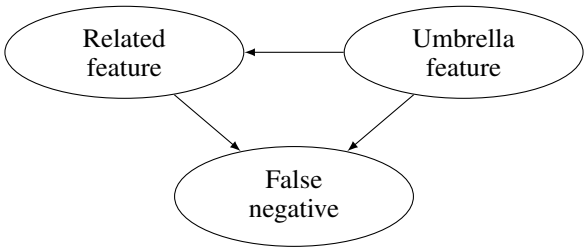
Due to the 71-element featurization of our dataset, we are working with high-dimensional data. Thus, we model the clusters generated by *k*-means as multivariate Gaussian distributions. Modeling the clusters as multivariate Gaussian distributions over the random variable set $x = [X_1, ..., X_d]$ allows us not to make independence assumptions about joint variables, as the density function is defined as:

$$f(x, \mu, \Sigma) = \frac{1}{\sqrt{|\Sigma|(2\pi)^d}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\right) \tag{1}$$

We thus were able to generate a set of representative multivariate Gaussian distributions from the set of centroids learned through data clustering. Using the mean and covariance of the centroids, we can form the density distribution of cluster character distributions. We can then generate scored adversarial attacks by sampling from each centroid's characteristic multivariate Gaussian distribution, with thresholds in place to de-weight outliers. In this way, the feature data we generate is similar to the original feature data at frequencies proportionate to the density function probabilities.

3.4 Bayesian network models

We developed probabilistic graphical models, namely Bayesian networks, to simulate the user classification process from biometric feature vectors. The motivation behind employing Bayesian networks was the existence of exact probabilistic inference techniques such as variable elimination which would allow for forward and backward examination of feature and classification probabilistic dependencies. Our insight was that by examining the contributing features to classification decisions, we could score their principality to the decisions, generating numerical saliency maps that could inform as well as speed up the process of generating successful adversarial attacks.



Pragmatically, we construct Bayesian models that are not naïve, hence do not make an independence assumption between feature variables. Bae et al. describe a method to learn Bayesian networks from correlated data [9]; to this end, we generated Pearson correlation matrices across 71 features to better understand the correlated nature of the dataset.

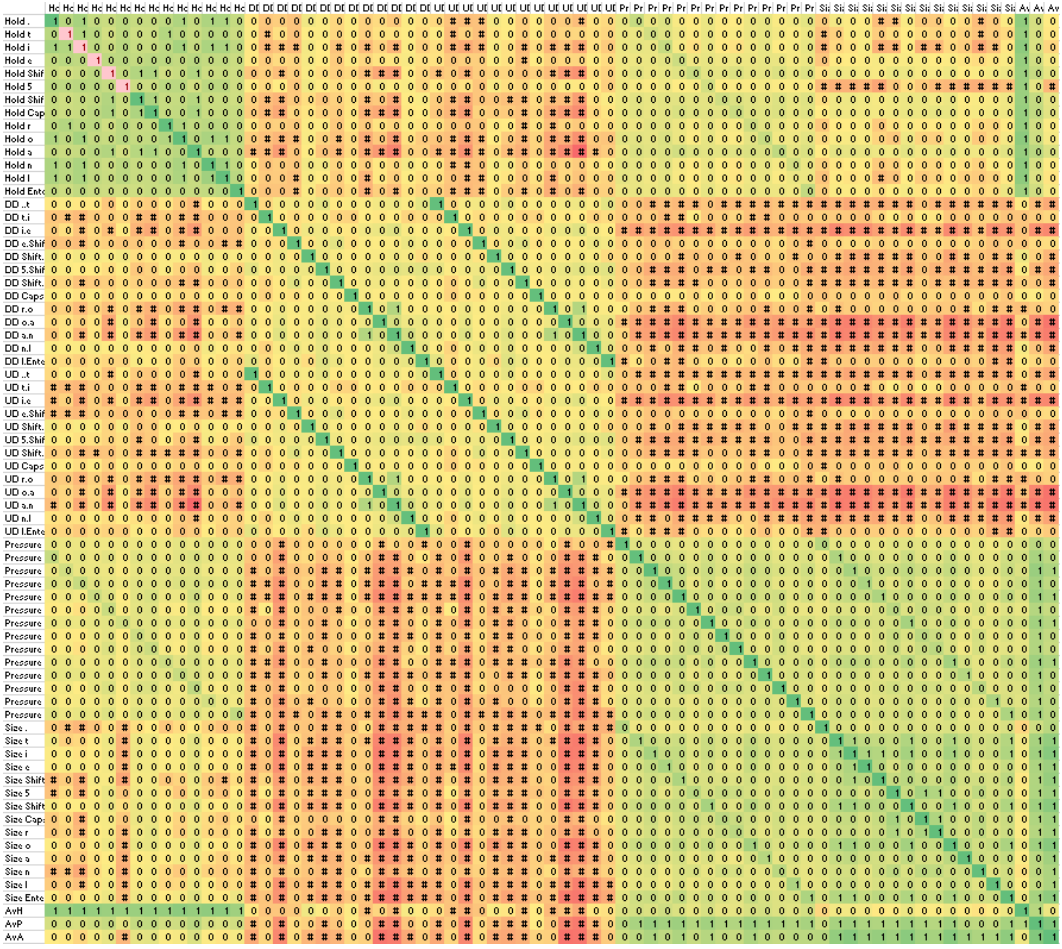


Figure 3: 71 by 71 correlation matrix (all user data). Green tint is higher correlation, and red tint is lower correlation. Even visually, we begin to parse correlated and uncorrelated features.

4 Results and Discussion

Matching our intuition, the k-means attacks performed best with higher k (until 32-64 range with cluster method), reaching 77% success rate (undetected spoof) against a 5-layer deep neural net. Likewise, using our multivariate Gaussian distribution model to introduce humanistic perturbations and augment the number of generated adversarial examples, we achieve a lower maximum 67% success rate.

Based on the adversarial success rate metric of model robustness, we find that logistic regression is most resilient to our attacks (maximum success rate 20%), compared to the artificial neural net and deep neural nets. However, comparing attack success rate against false negative rate of each classifier reveals that the Logistic classifier is most resilient to attack because it had the highest false negative rate. The Logistic classifier is more likely to predict "false" on average.

From the plot of maximum attack success rate against classifier false negative rate, there is a roughly linear reciprocal relationship between attack success rate and false negative rate of the classifier. However, the 10-DNN classifier shows a reversal of this trend. It has both lower false negative rate and a lower attack success rate compared to the previous 5-DNN classifier. Further work will

determine if this relationship still holds for fully connected neural network classifiers beyond 10 hidden layers.

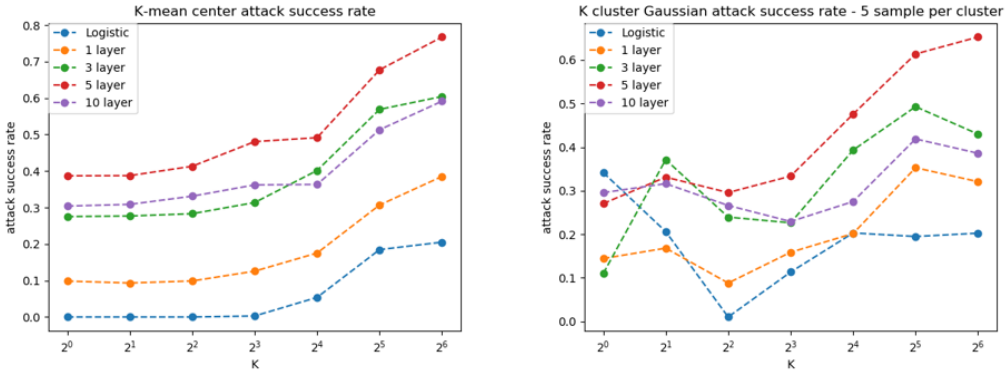


Figure 4: Adversarial success rate charts for different k and across five discriminatory classifier models (logistic regression, 1-layer DNN, 3-layer DNN, 5-layer DNN, and 10-layer DNN).

Attack	v Logistic	v 1-DNN	v 3-DNN	v 5-DNN	v 10-DNN
k-means center	20%	39%	60%	77%	59%
k-cluster Gaussian samples	20%	35%	49%	67%	42%

Table 1: Table of maximum success rates, across all attack methods and k values.

Model	Loss	Accuracy	Recall	Precision	False Negative Rate
Logistic	6.661	58.12%	23.88%	78.18%	76.12%
1-DNN	0.690	69.10%	49.05%	83.31%	50.95%
3-DNN	0.536	73.73%	70.37%	76.04%	29.63%
5-DNN	0.531	73.45%	72.52%	74.30%	27.48%
10-DNN	0.450	79.78%	77.46%	81.37%	22.54%

Table 2. Classification models evaluated on on validation set. False Negative Rate is calculated as 100% - Recall.

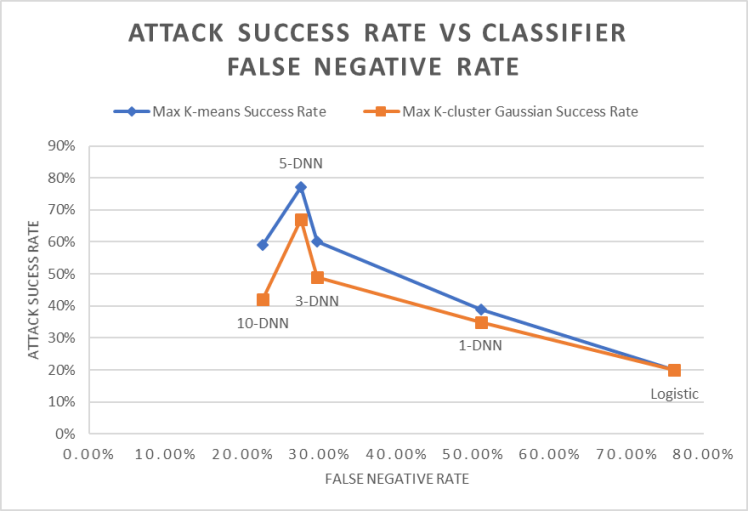


Figure 5: Maximum attack success rate plotted against false negative rate for each classifier.

5 Conclusions and future work

Our multivariate Gaussian cluster characterization and sampling algorithm, coupled with our Bayesian network heuristics, achieved 77% successful spoof rate against state-of-the-art mobile biometric fraud classifiers (neural nets, as described by Teh et al.). This means that 77% of the time, our adversarial biometric examples successfully trick the best discriminatory classifiers that they are from the genuine user.

We described a framework that allows for infinite generation of adversarial examples according to a probability distribution generated from our multivariate Gaussian models and Bayesian networks. The flexibility of the framework is perhaps the most simultaneously promising and disturbing result; we explored targeted adversarial attacks on individual users, but demonstrated that our Bayesian network heuristics can be learned on any set of users. It is not difficult to plant a keylogger and use our methods to compromise modern biometric security systems; we have exposed a major security vulnerability that motivates further research in developing more resilient discriminatory classification algorithms for mobile biometric authentication.

Future work could include developing more extensive evaluation frameworks for measuring spoof success, and applying our proposed adversarial example generation methods to develop more robust discriminatory classifiers; for example, in the context of generating adversarial examples that challenge and provide areas of insight to developing more discriminatory authentication classifiers, the evaluation metrics might prioritize maximizing false positives (incorrectly authenticating a false user) in the classifier.

Another potential project involves exploring other types of attacks on user verification classifiers. For example, a common industry standard user verification system measuring keystroke dynamics will retrain its classifier as a user continues to use the device. A targeted attack against this retraining could inject enough malicious data to compromise the system. An attack could inject enough malicious data to shut the original user out of his own device? The structure of this attack, a stream of input data, would build upon the developments of this paper, as one methodology for developing this classifier involves classifying unlabeled data in order to retrain the model.

Acknowledgments

We would like to thank our mentor Steve Mussmann for extensive discussion and feedback. We would also like to thank Christopher Sauer, Alisha Rege, and Prof. Dan Boneh for advice on data and methods. Finally, we would like to thank Prof. Percy Liang and Prof. Stefano Ermon for valuable insight on paradigms of artificial intelligence, especially adversarial systems and Bayesian networks.

CodaLab

Our model training and evaluation code can be found at:
https://worksheets.codalab.org/worksheets/0x32e063d86fa04b358280f98f5922a437/

References

[1] N. Al-Obaidi. MEU-Mobile KSD Data Set. UCI Machine Learning Repository, 2016.

[2] I. de Mendizabal-Vazquez, D. de Santos-Sierra, J. Guerra-Casanova, and C. Sanchez-Avila. Supervised classification methods applied to Keystroke Dynamics through Mobile Devices. *ICCST*, 2014.

[3] T. Cho. Pattern Classification Methods for Keystroke Analysis. *SICE-ICASE*, 2006.

[4] L.J.P. van der Maaten. Accelerating t-SNE using Tree-Based Algorithms. *Journal of Machine Learning Research*, 2014.

[5] A. Fawzi, S. Moosavi-Dezfooli, P. Frossard. Robustness of classifiers: from adversarial to random noise. *NIPS*, 2016.

[6] C. Dwork, A. Roth. Differential privacy. *Foundations and Trends in Computer Science*, 2014.

[7] Y. Gal, Z. Ghahramani. Bayesian Convolutional Neural Networks with Bernoulli Approximate Variational Inference. *arXiv:1506.02158*, 2016.

[8] P.S. Teh, N. Zhang, A.B.J. Teoh, K. Chen. A survey on touch dynamics authentication in mobile devices. *Computers & Security*, 2016.

[9] H. Bae, S. Monti, M. Montano, M.H. Steinberg, T.T. Perls, P. Sebastiani. Learning Bayesian Networks from Correlated Data. *Nature Scientific Reports*, 2016.

[10] S. Sen, K. Muralidharan. Putting "pressure" on mobile authentication. *ICMU*, 2014.

[11] N. Jeanjaitrong, P. Bhattarakosol. Feasibility study on authentication based keystroke dynamic over touch-screen devices. *ISCIT*, 2013.