

# User Authentication through Keystroke Dynamics

Devashish Singh (163059001)

Prateek Patidar (163059006)

Shubham Singh (163059008)

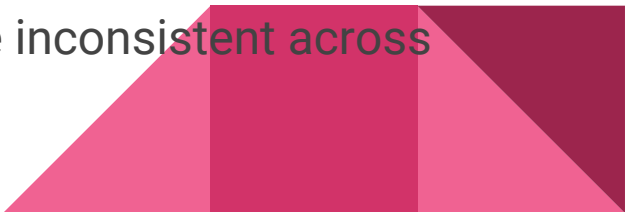
Hareesh Kumar (16305R013)

# Problem Description

Keystroke dynamics—the analysis of typing rhythms to discriminate among users—has been proposed for detecting impostors (i.e., both insiders and external attackers).

Since many anomaly-detection algorithms have been proposed for this task, it is natural to ask which are the top performers (e.g., to identify promising research directions).

Unfortunately, we cannot conduct a sound comparison of detectors using the results in the literature because evaluation conditions are inconsistent across studies.

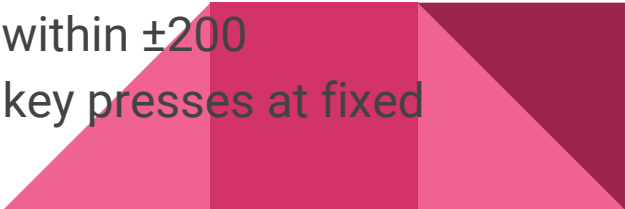


# Dataset Collection

As the subject types the password, it is checked for correctness. If the subject makes a typographical error, the application prompts the subject to retype the password.

Whenever the subject presses or releases a key, the event (i.e., keydown or keyup), along the key involved, and a timestamp for the moment at which the keystroke event occurred.

An external reference clock was used to generate highly accurate timestamps. The reference clock was demonstrated to be accurate to within  $\pm 200$  microseconds (by using a function generator to simulate key presses at fixed intervals).



51 subjects (typists) typed the same password, and each subject typed the password 400 times over 8 sessions (50 repetitions per session).

They waited at least one day between sessions, to capture some of the day-to-day variation of each subject's typing.

The password (.tie5Roanl) was chosen to be representative of a strong 10-character password.



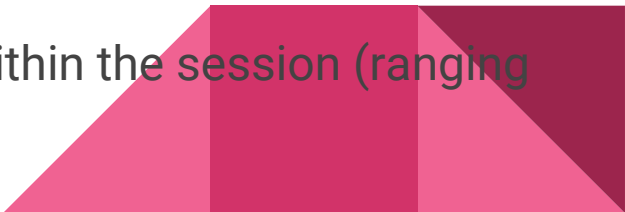
# Dataset Information

The data are arranged as a table with 34 columns. Each row of data corresponds to the timing information for a single repetition of the password by a single subject.

The first column, `subject`, is a unique identifier for each subject (e.g., `s002` or `s057`).

The second column, `sessionIndex`, is the session in which the password was typed (ranging from 1 to 8).

The third column, `rep`, is the repetition of the password within the session (ranging from 1 to 50).

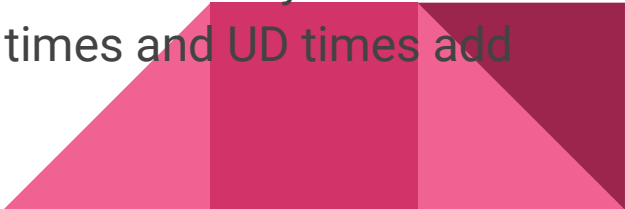


The remaining 31 columns present the timing information for the password. The name of the column encodes the type of timing information.

Column names of the form H.key designate a hold time for the named key (i.e., the time from when key was pressed to when it was released).

Column names of the form DD.key1.key2 designate a keydown-keydown time for the named digraph (i.e., the time from when key1 was pressed to when key2 was pressed).

Column names of the form UD.key1.key2 designate a keyup-keydown time for the named digraph (i.e., the time from when key1 was released to when key2 was pressed). Note that UD times can be negative, and that H times and UD times add up to DD times.



# Approach

We did some feature engineering on the dataset i.e. added square and square root of features.

We then used some classification methods available in scikit-learn library on the data like Logistic Regression, Support Vector Machines, Random Forests, K NN Classification, MLP and Gaussian Naive Bayes.

Each of the anomaly detectors in our comparison was comprised of two functions, a training function and a scoring function.

The training function takes a matrix of password-timing information as input, and it outputs a detection model. Each row of the input matrix encodes password-timing information from one repetition of the genuine user typing the password.

The function uses this set of timing information to build a model of that user's typing. The details of the model are detector specific, and they need not take a particular form for our evaluation.

The scoring function takes the detection model produced by the training function and training as input.

It outputs a set of scores, indicating the degree to which that new sample is similar to the typing model.





# Results of experimentation (BEST results)

Logistic Regression : 0.76862745098

Support Vector Machines : 0.752941176471

Random Forests : 0.90245098039

K NN Classification : 0.157679738562

Gaussian Naive Bayes : 0.57091503268



# Predicting user based on Keystrokes

We have applied some correlation techniques to match the user-keystroke pattern with the already learnt pattern to test identity of the user.

Currently, we take 2 inputs for storing and test it against the target.



# References

- [1] Kevin S. Killourhy and Roy A. Maxion. "Comparing Anomaly Detectors for Keystroke Dynamics," in Proceedings of the 39th Annual International Conference on Dependable Systems and Networks (DSN-2009), pages 125-134, Estoril, Lisbon, Portugal, June 29-July 2, 2009. IEEE Computer Society Press, Los Alamitos, California, 2009.
- [2] T. Sing, O. Sander, N. Beerenwinkel, T. Lengauer. "ROCR: visualizing classifier performance in R," *Bioinformatics* 21(20):3940-3941 (2005)
- [3] Shambhu Upadhyaya Hayreddin C, eker. Enhanced recognition of keystroke dynamics using gaussian mixture models. Military Communications Conference, MILCOM 2015 IEEE, 2015