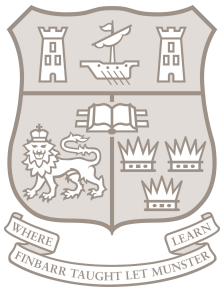
Profiling Side-Channel Attacks on Cryptographic Algorithms

Neil Hanley

Department of Electrical and Electronic Engineering,
University College Cork,
Ireland.

neilh@eleceng.ucc.ie

9th June 2014

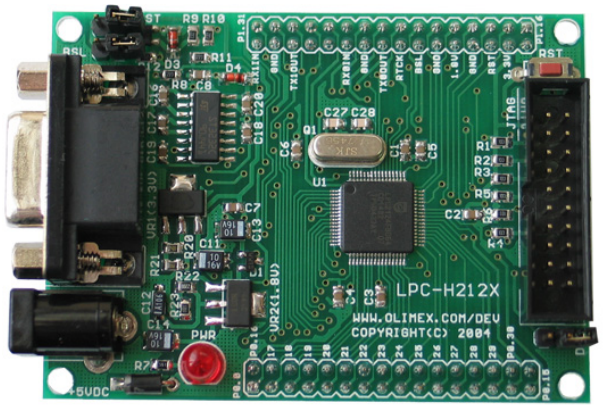


Overview

Power based side-channel attacks can be broadly split into two catergories:

- Non-profiling:
  - DPA, CPA, MIA, ...
  - Collision based attacks generally fall here too.
  - Often works with minimal assumptions.
  - Moderate to large number of traces required for successful key recovery.
- Profiling:
  - TA, Stochastic, SVMs, LR, NN, ...
  - Stronger adversarial as profiling device required.
  - Can recover key with minimal or only a single trace.

Acquisition Setup



Template Attacks

- Attack Outline
- Practical Attack Considerations
- Model Validity

Trace Only Template Attacks - AES

- Templates on Multiple Intermediate Values
- Templates on S-Box Only
- Templates on the Key

Trace Only Template Attacks - Multiplication

- Leakage
- Templates on Multiplication & Squaring Operations
- Attacking ECDSA

Machine Learning Methods for SCA

- Stochastic Methods
- Logistic Regression
- Support Vector Machines
- Neural Networks
- Comparison

Template Attacks

- Training (learning) stage.

$$\hat{\mu}^{(i)} = \frac{1}{m^{(i)}} \sum_{j=1}^{m^{(i)}} x^{(j,i)}$$

$$\hat{\Sigma}^{(i)} = \frac{1}{m^{(i)}} \sum_{j=1}^{m^{(i)}} \left(x^{(j,i)} - \hat{\mu}^{(i)} \right) \left(x^{(j,i)} - \hat{\mu}^{(i)} \right)^{\top}$$

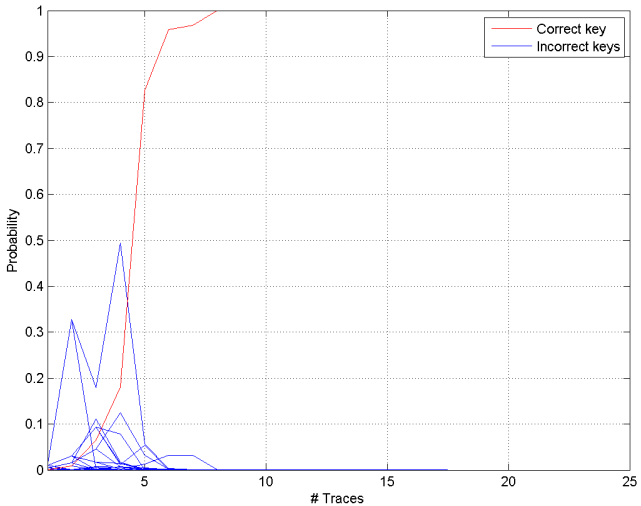
- Testing (classification) stage.

$$\mathcal{N}\left(x \mid \mu^{(i)}, \Sigma^{(i)} \right) = \frac{1}{\sqrt{(2\pi)^n \left| \Sigma^{(i)} \right|}} e^{-\frac{1}{2}\left(x-\mu^{(i)}\right)\left(\Sigma^{(i)}\right)^{-1}\left(x-\mu^{(i)}\right)^{\top}}$$

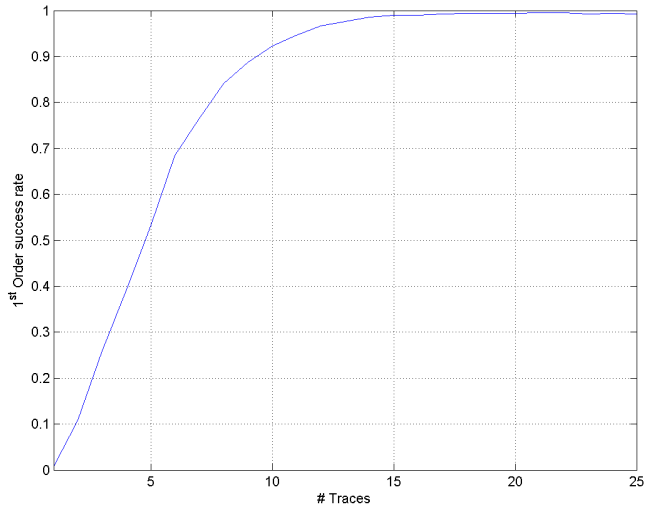
$$\Pr\left(o^{(i)} \mid x \right) = \frac{p\left(x \mid o^{(i)} \right) \Pr\left(o^{(i)} \right)}{\sum_{j=1}^{|\mathcal{K}|} p\left(x \mid o^{(j)} \right) \Pr\left(o^{(j)} \right)}$$

Example Template Attack against AES

- $10k$ training traces.
- S-Box output of round 1.
- Hamming weight power model.
- 20 features selected via SOSD.



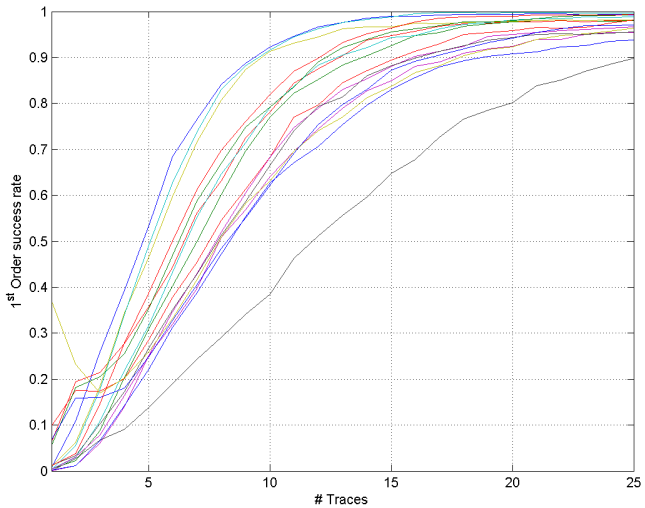
Amplified template attack.



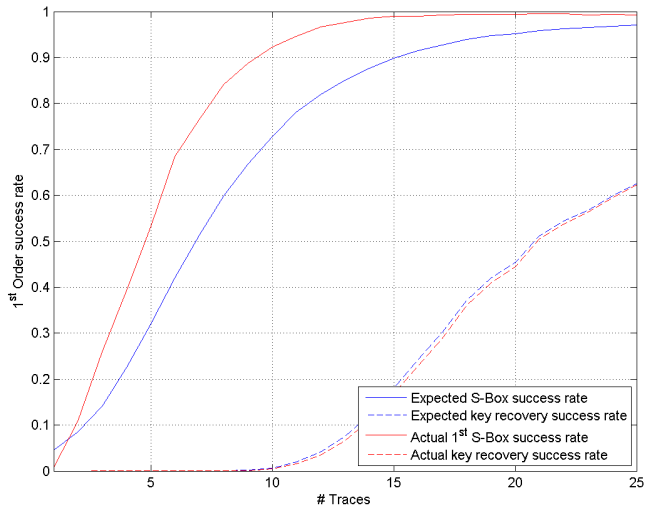
Success rate.

Full vs. Partial Key Recovery

- Key bytes not classified equally.
- Cannot arbitarily extrapolate results for a single key byte to key as a whole.



Byte success rates.



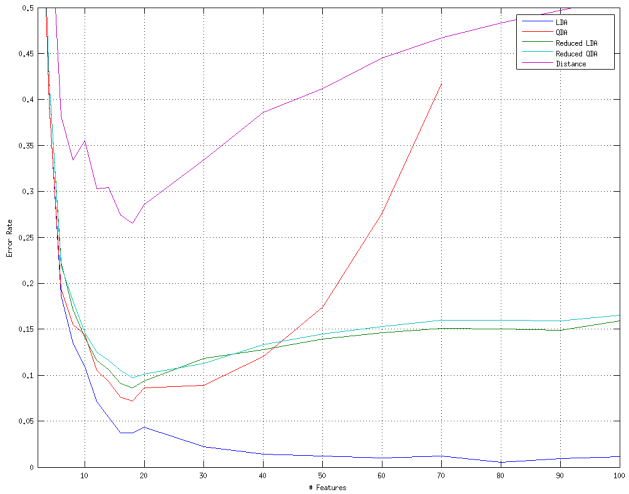
Key success rate.

Practical Attack Considerations

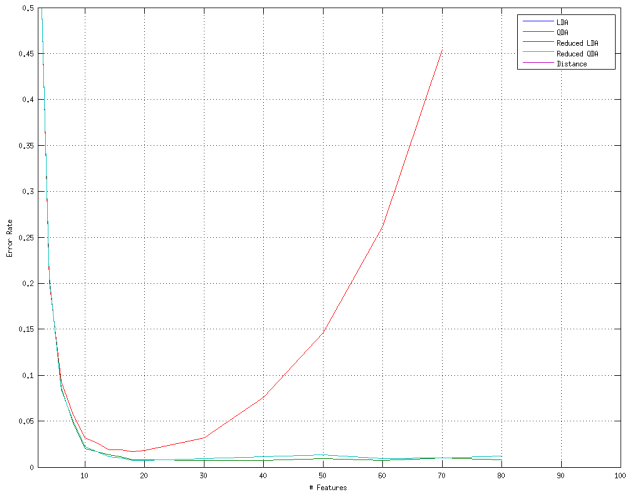
- Power Model.
  - Bit, multi-bit, Hamming weight, identity.
- Normalisation
  - Norm, range, scale, z-score.
- Feature selection.
  - SOSD, SOST, Pearson's correlation, PCA, Fisher's linear discriminant, (NICV), (SNR).
- Training set size.
- Target intermediate value.
- Classification algorithm.
  - Euclidean distance, reduced templates, LDA, QDA.

Classification Algorithms

- The **Identity model** allows key recovery in a single trace, and has an increased number of relevant features.
- Taking the **z-score** can help to prevent a single feature dominating the classification.
- Increasing the training set beyound $\gtrsim 15k$ traces, doesn't decrease the expected error (QDA).
- Targeting the **S-Box** is the most efficent intermediate value for key recovery.



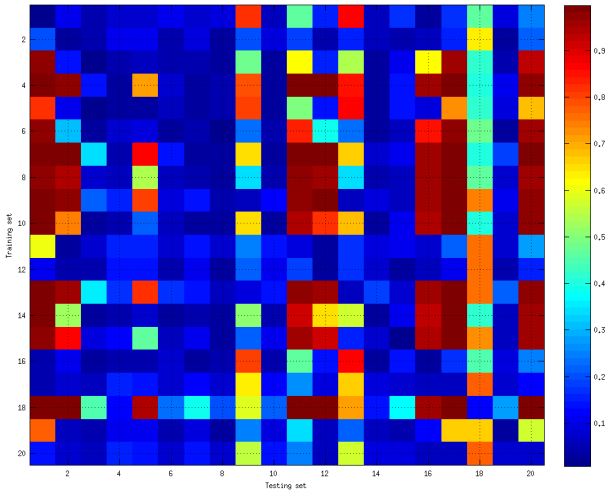
SOST.



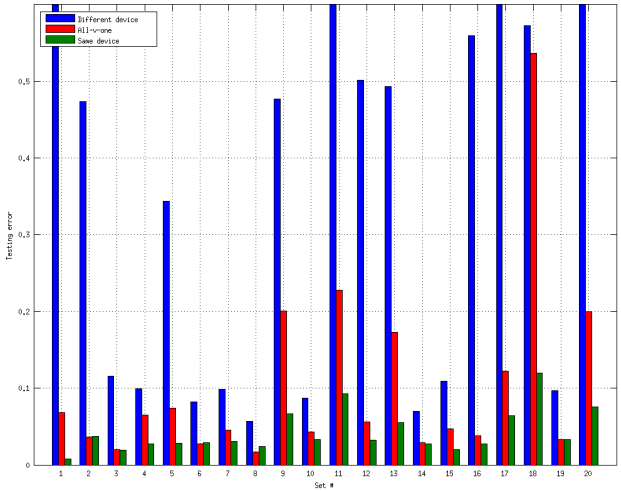
Fisher's linear discriminant.

Model Validity

- 20 PIC Microcontrollers (cheap!!!)
- Basic devices $\rightarrow$ advantageous for experiment.
- Improved classification when using multiple devices.



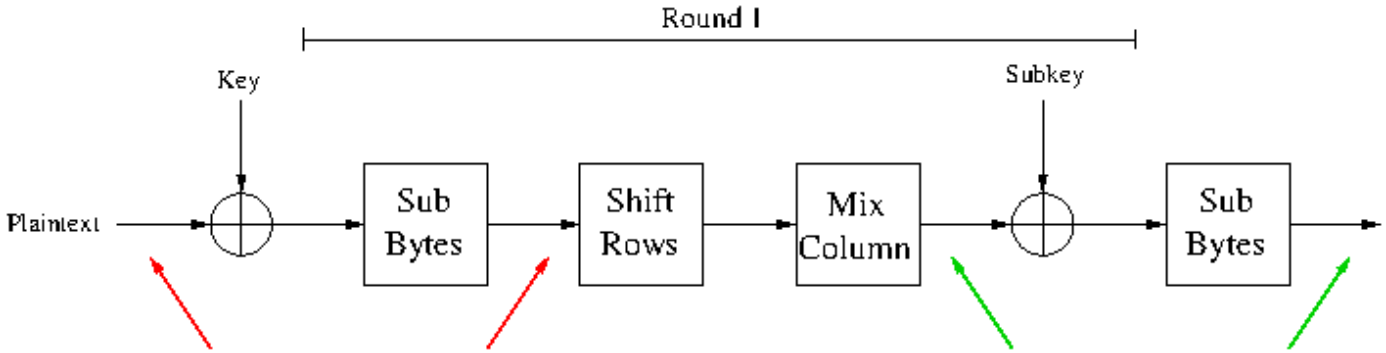
Single device templates.



Multi device templates.

Templates on Multiple Intermediate Values

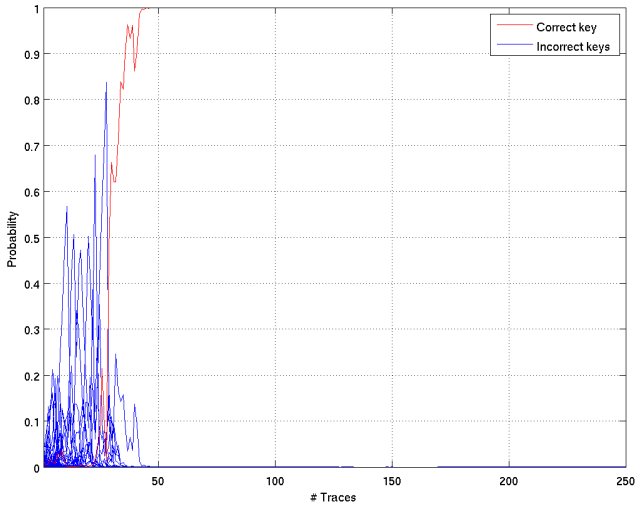
- Target values *must* be seperated by a non-linear operation.



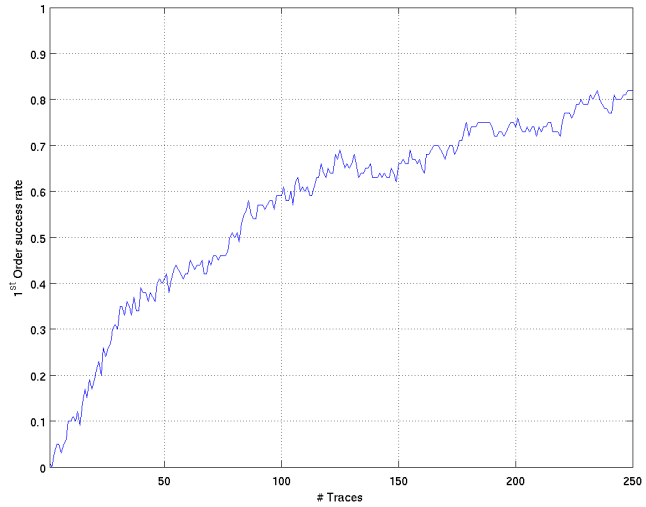
AES unknown plaintext attack target.

Templates on Multiple Intermediate Values

- Use of the Hamming weight model has much lower success rate.
- Poor classification of the Plaintext/MixColumn bytes at fault.

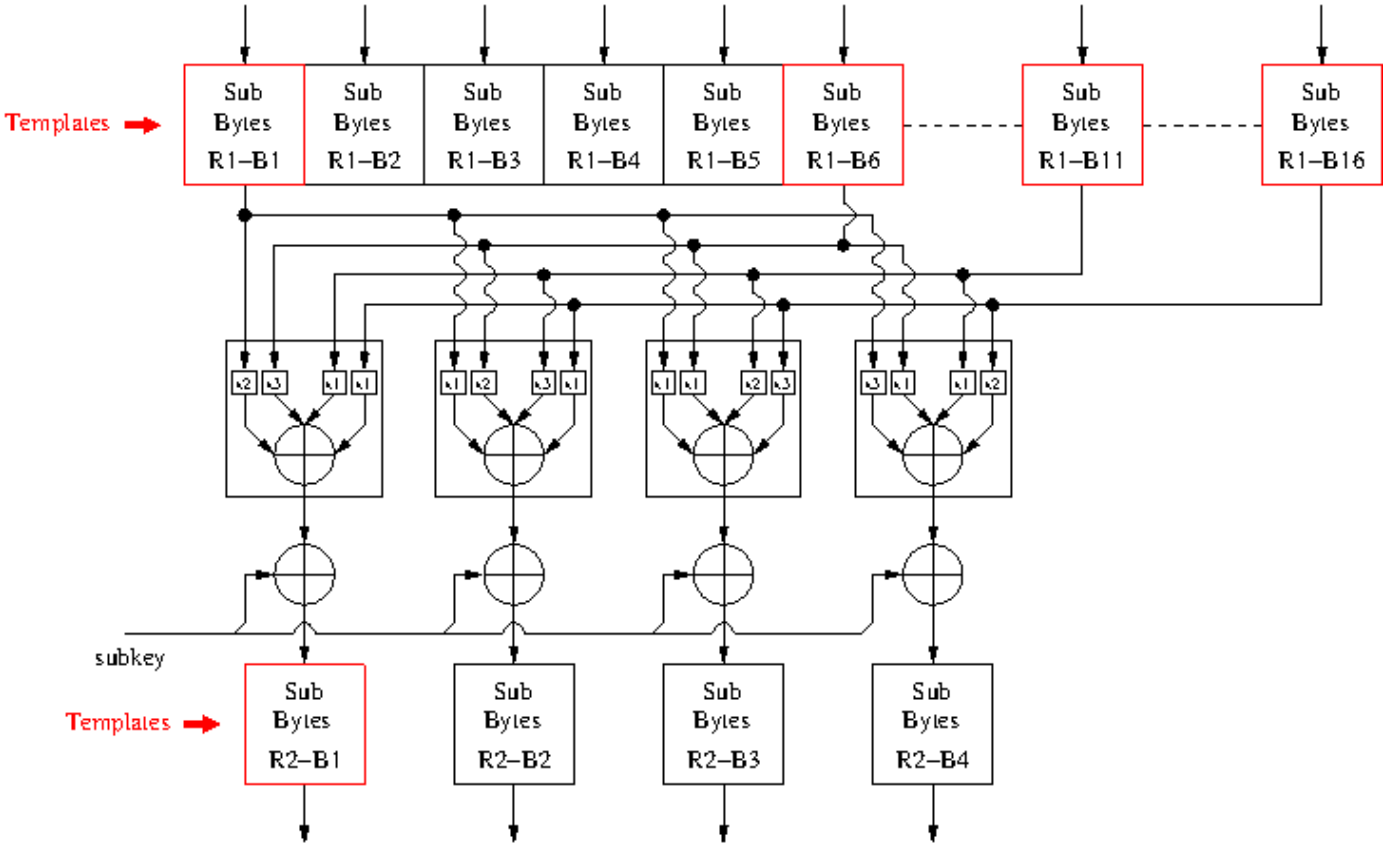


Single attack instance.



Success rate.

Templates on S-Box Only



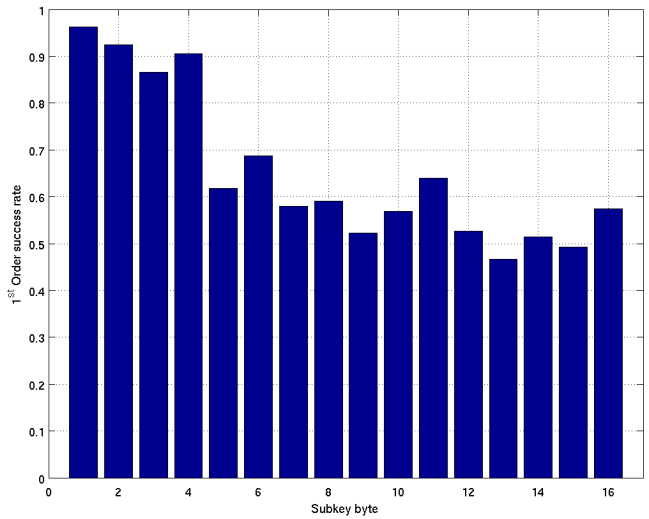
Target S-Boxes to extract the first sub-key byte.

Templates on S-Box Only

	S-Box							
Round	1	2	3	4	5	6	7	8
1	0.0087	0.1088	0.0753	0.0752	0.1271	0.0037	0.1555	0.1020
2	0.0104	0.0490	0.1103	0.0704	0.1068	0.0075	0.1632	0.1491

	S-Box							
Round	9	10	11	12	13	14	15	16
1	0.1170	0.0731	0.0050	0.1209	0.1279	0.1194	0.0797	0.0108
2	0.1911	0.1232	0.0124	0.1941	0.1931	0.1190	0.1467	0.0098

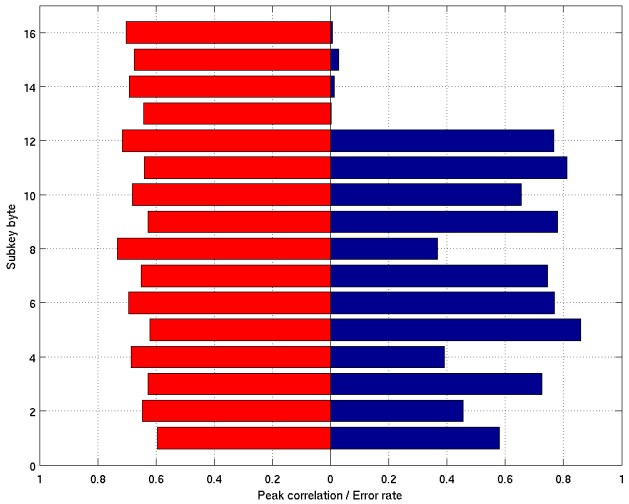
Comparison of S-Box classification errors.



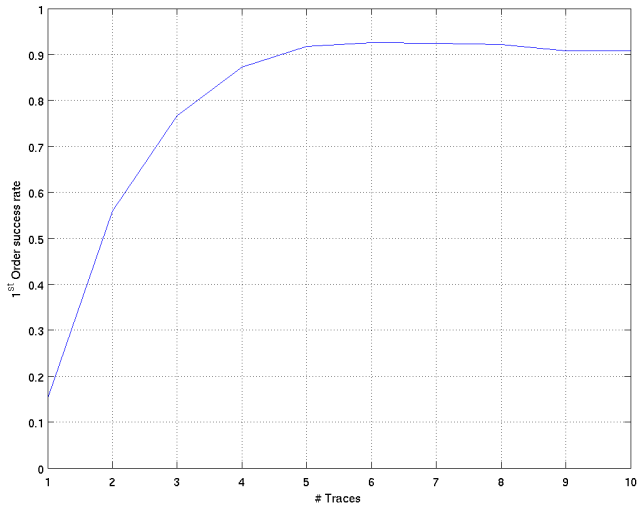
Success rate for each key byte.

Templates on the Key

- Attacking key bytes directly is unreliable.
- Better approach is to attack S-Boxes in different rounds of the key expansion.



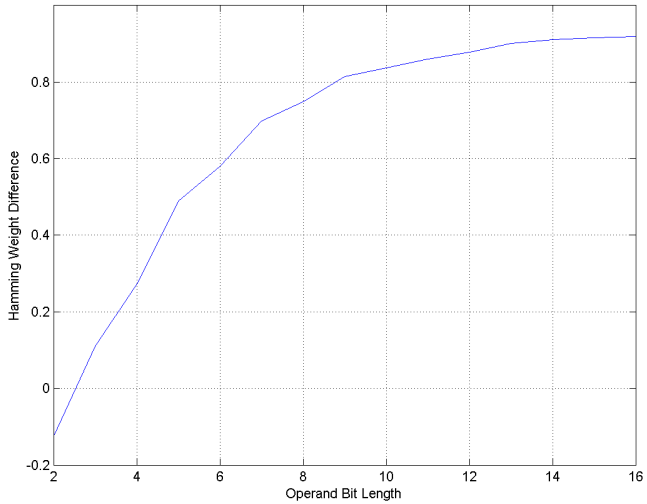
Error for each key byte.



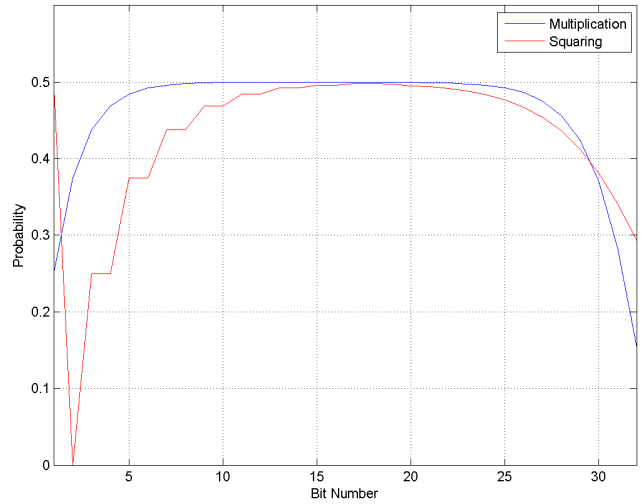
Success rate for *entire* key.

Power Difference Between Multiplication & Squaring Operations

- Expected Hamming weight of the *result* of a multiplication operation different to that of a squaring.
- RSA and ECC operations consist of many single precision multiplications giving many potential points of leakage.

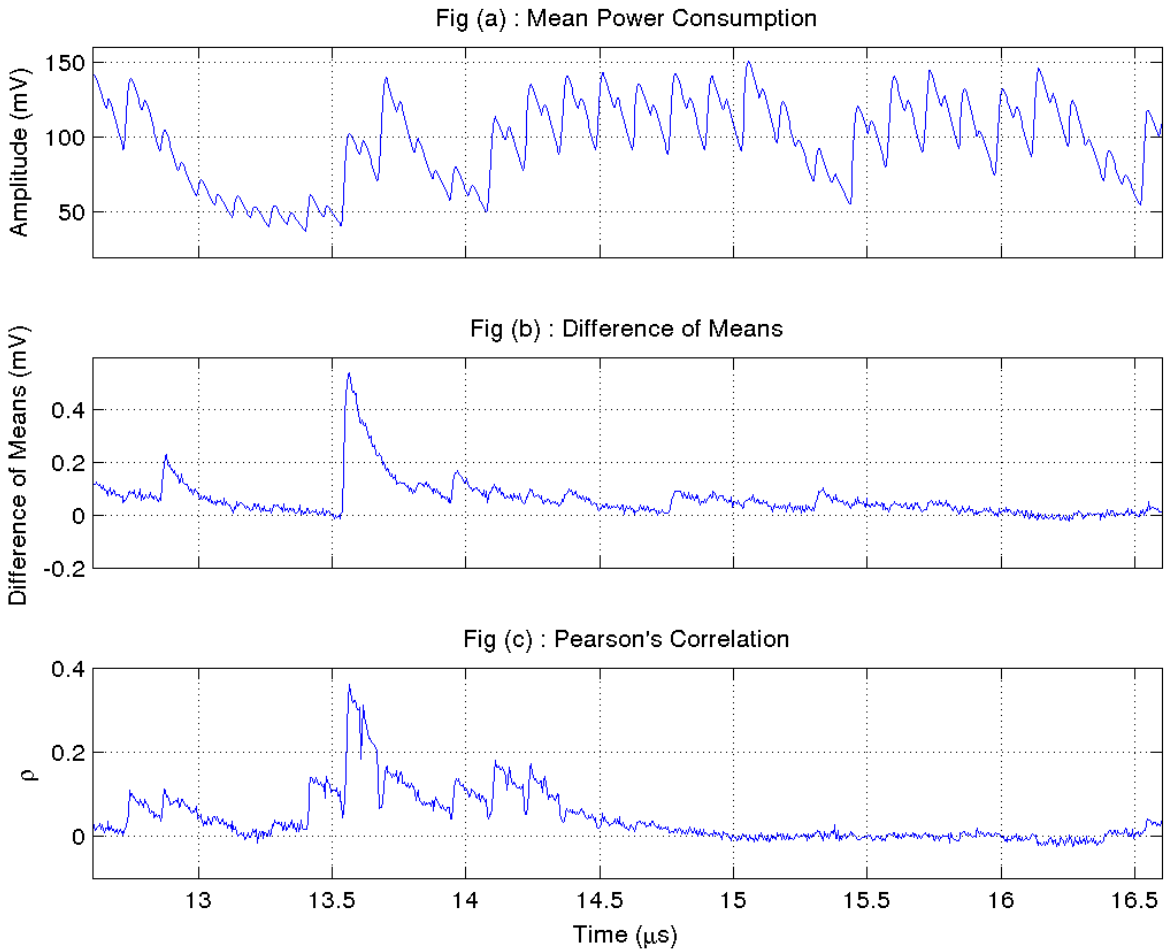


Hamming weight difference.



Probability of an output bit.

Multiplier Leakage



Multiplier target leakage.

Montgomery Product

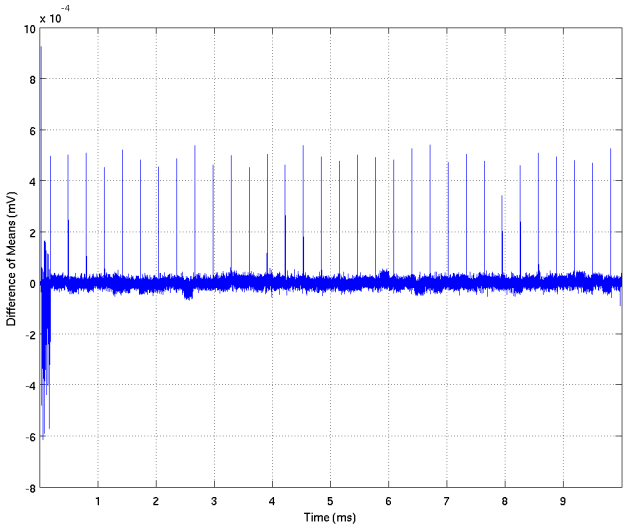
Data: $N = (N_{w-1}, \ldots, N_1, N_0)_b$, $x = (x_{w-1}, \ldots, x_1, x_0)_b$,
$y = (y_{w-1}, \ldots, y_1, y_0)_b$ with $0 \leq x, y < N$, $R = b^w$,
$\gcd(N, b) = 1$ and $N' = -N^{-1} \bmod b$

Result: $A \leftarrow x\, y\, R^{-1} \bmod N$

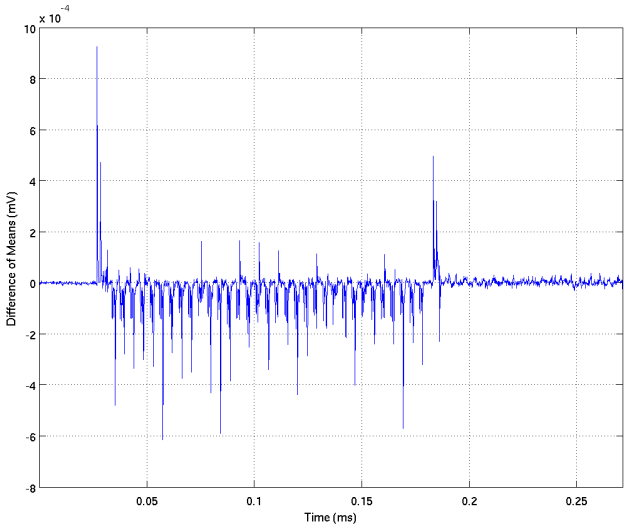
$A \leftarrow 0$;
for $i = 0$ **to** $w - 1$ **do**
    $u_i \leftarrow (a_0 + x_i\, y_0) N' \bmod b$;
    $A \leftarrow (A + x_i\, y + u_i\, N)/b$;
end
if $A \geq N$ **then** $A \leftarrow A - N$;
return A ;

Difference of Means

- Large peak for each round.
- Double peak for initial round.
- Smaller peaks where initial word is re-used.



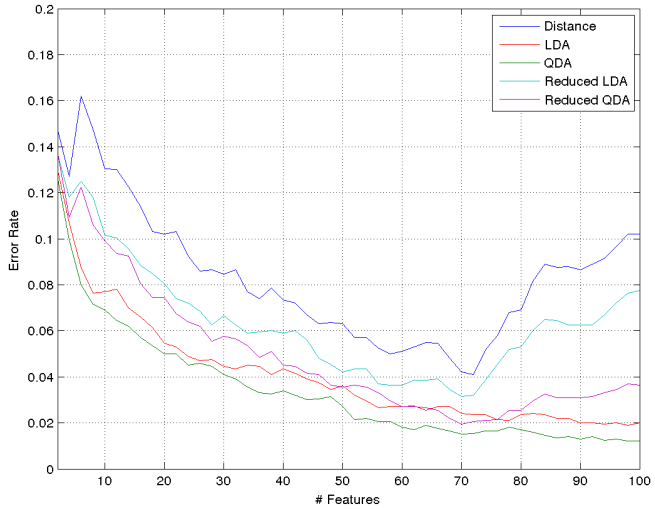
Difference of means trace.



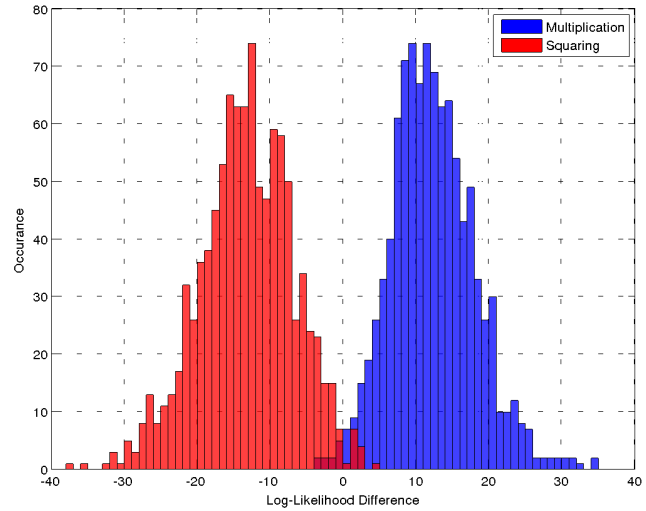
Difference of means initial loop.

Tempate Attack on 1024-bit Montgomery Multiplication

- Equivalent to recovery of a single key-bit in an RSA exponentiation.
- Single trace recovery overcomes many countermeasures.
- Longer keys lead to more efficient attacks.
- Error equally distributed between multiplication & squaring.



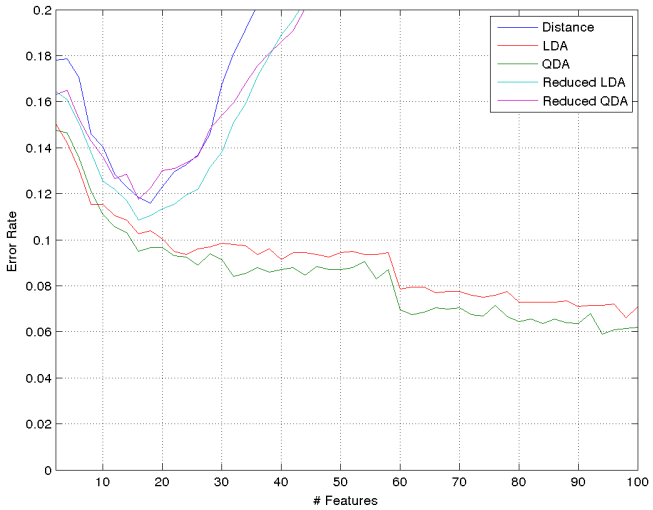
Error rate.



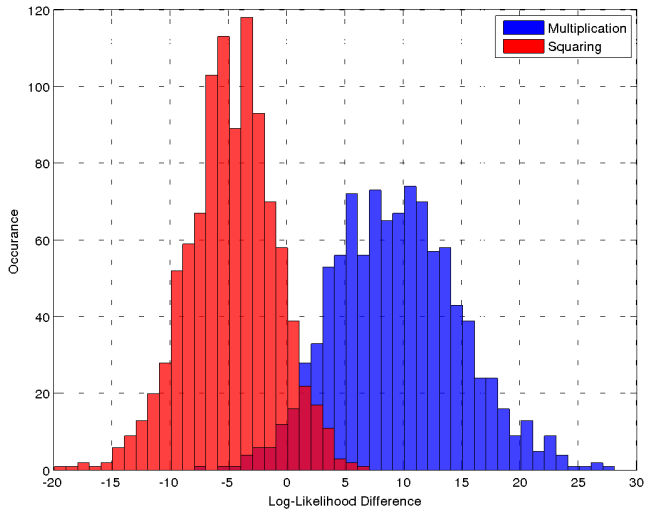
Log likelihood.

Tempate Attack on 160-bit Montgomery Multiplication

- Errors no longer equally distributed.
- For ECC, many multiplication operations per group doubling/addition - compensate for poorer classification.



Error rate.



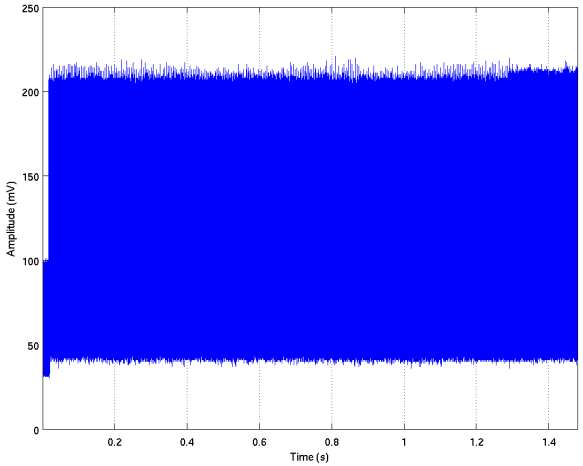
Log likelihood.

Tempate Attack on ECDSA

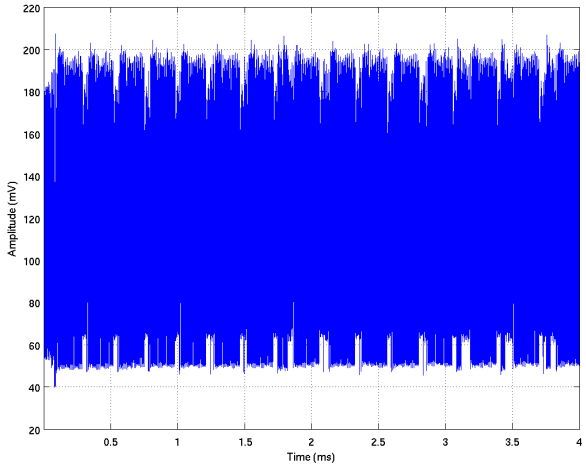
Single-trace attack on ECC scalar multiplication of ECDSA signature.

- 192-bit Scalar multiplicand k *must* be random for each signature.
- Unified group operations for doubling & addition.
- Projective coordinates with random Z-coordinates used.
- Blinding of the base point P is not implemented, but knowledge of P is not utilised in the attack.
- Templates built for group operation.
- Determining the empheral secret k allows the recovery of the private key d which allows the forging of signatures.

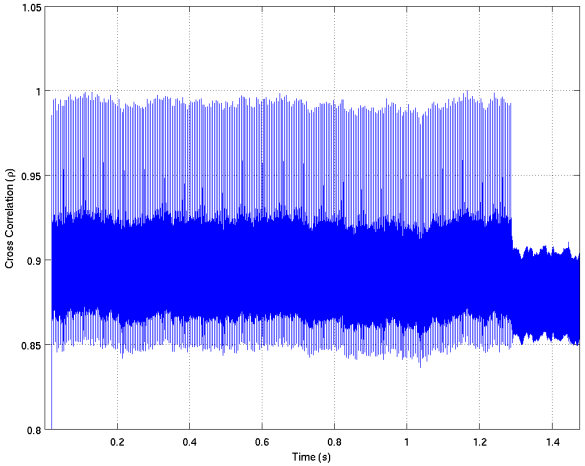
Tempate Attack on ECDSA



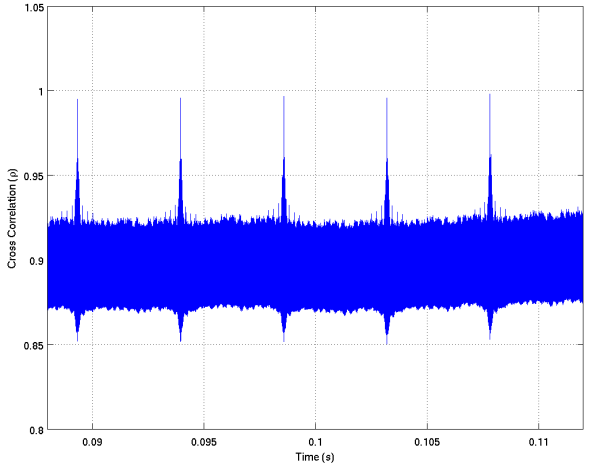
Full ECDSA power trace.



Mean group operation.



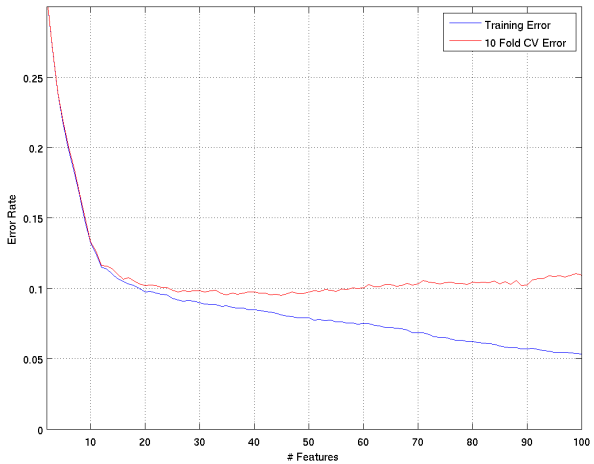
Cross correlation trace.



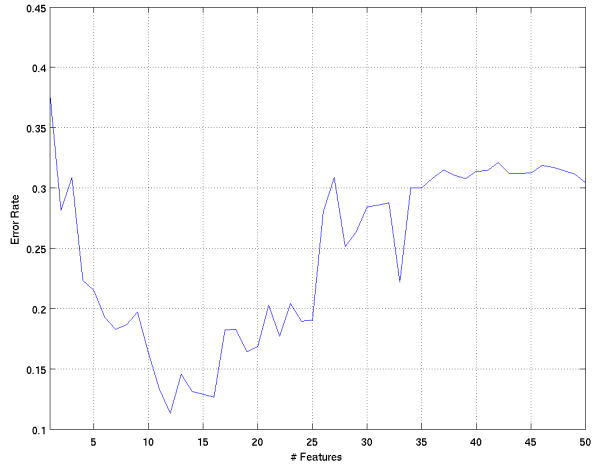
Zoomed x-correlation trace.

Tempate Attack on ECDSA

- Attacker needs to estimate the number of features via cross validation.
  - Optimal number of features here differs.
- Minimum error of 0.113 achieved on classification of group operations:
  - Expected that ≈ 32 operations will be incorrectly classified for 192-bit key.



Cross validation error.



Testing error.

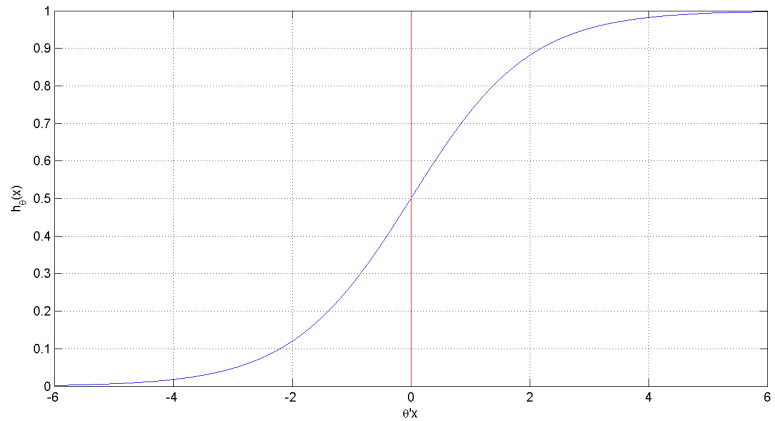
Stochastic Methods

Introduced by Schindler *et al.* [SLP05].

- Linear regression based classification.
- Looks to model power consumption with less traces than TA.
- Deterministic & random trace parts modelled seperately.
  - Modelling of random part optional.
- Required to select basis function.
  - Bit-wise, identity, Hamming weight.
  - One-hot encoding.
  - Polynomial expansion.

Logistic Regression

- Binary classification algorithm.
  - Bit-wise, one-v-all, one-v-one, binary tree.
- Tunable regularisation parameter to prevent overfitting to training data.
- Learned weighting parameters applied directly to trace to determine if class label is 0 or 1 based on sign of output.
- Sigmoid function can be used to convert distances to probabilities.

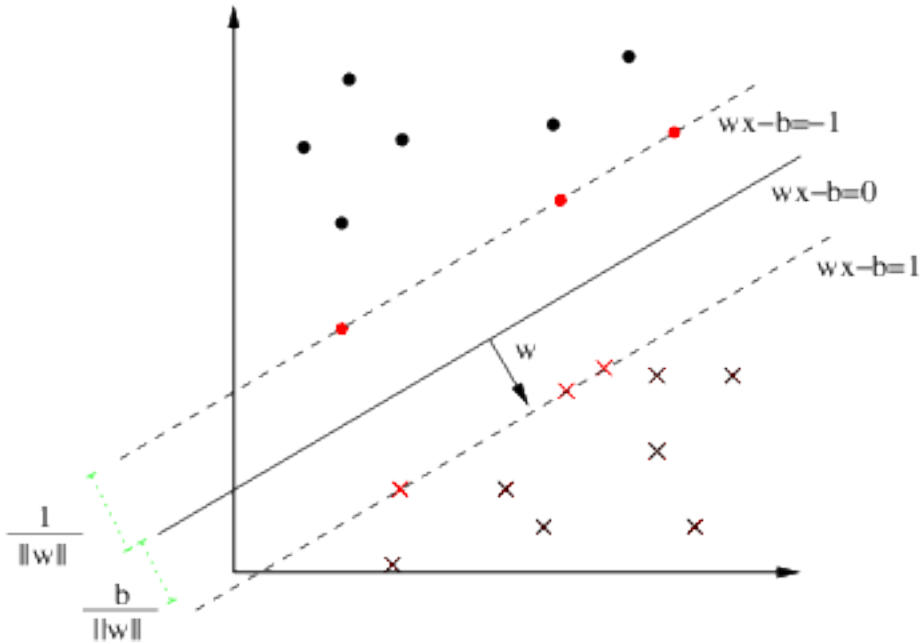


Sigmoid function.

Support Vector Machines

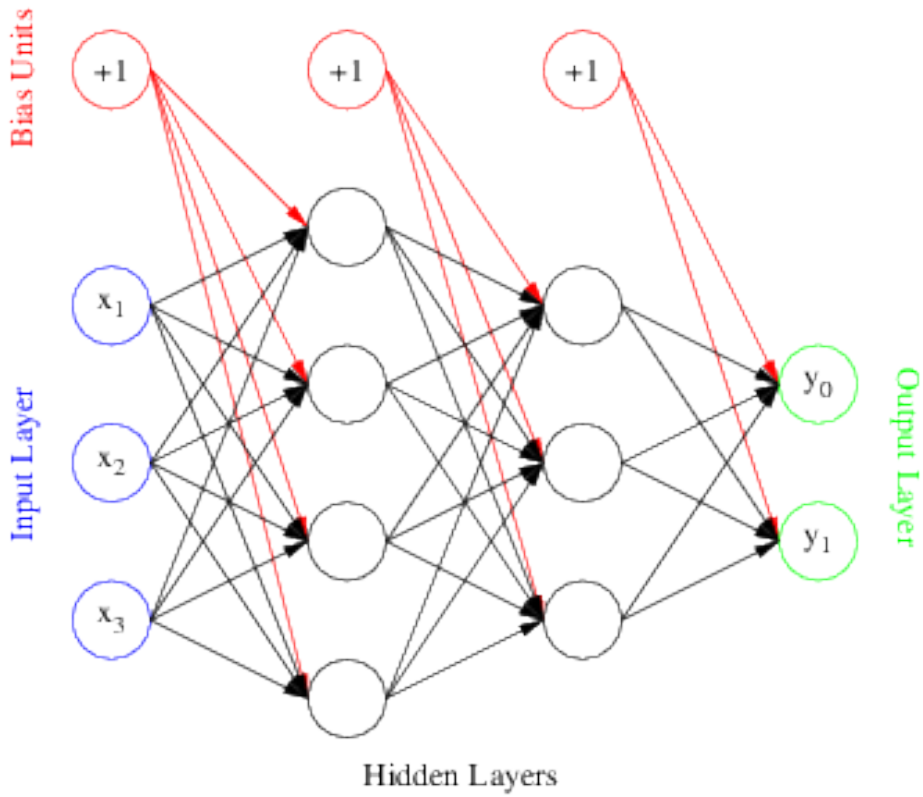
Seperately suggested by Hospodar *et al.* [HM+11], and Lerman *et al.* [LBM11].

- Binary classification algorithm.
- Non-parametic approach.
  - Greater flexibilty in feature construction & selection
- Kernel selection, regularisation parameter.
- Probabilistic output via Platt's method [Pla99].



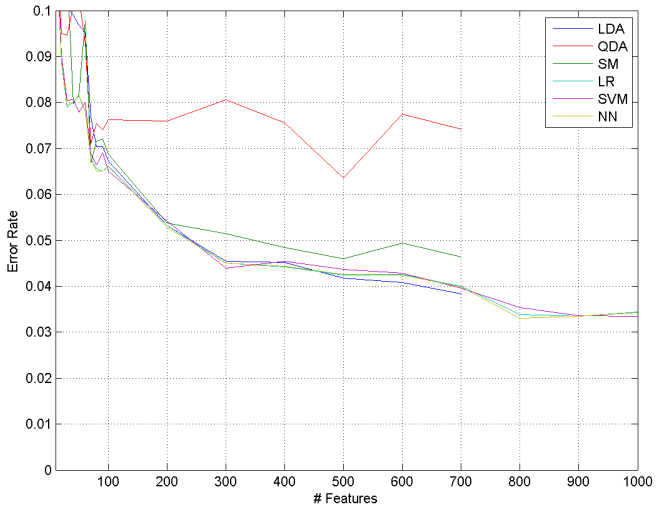
Neural Networks

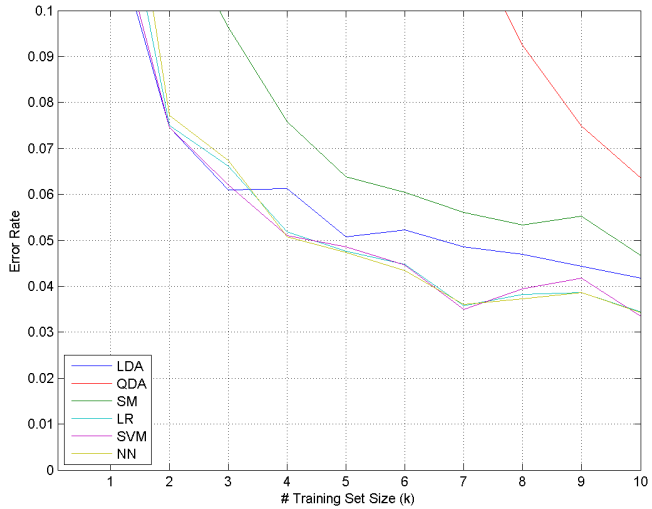
- Originally meant to mimic the operation of the brain,
- Backpropogation algorithm used to learn weighting parameters.
- Select the number of hidden layers/units, regularisation parameter.
- Sigmoid function used as activation unit.



Comparison - Multiplier (binary)

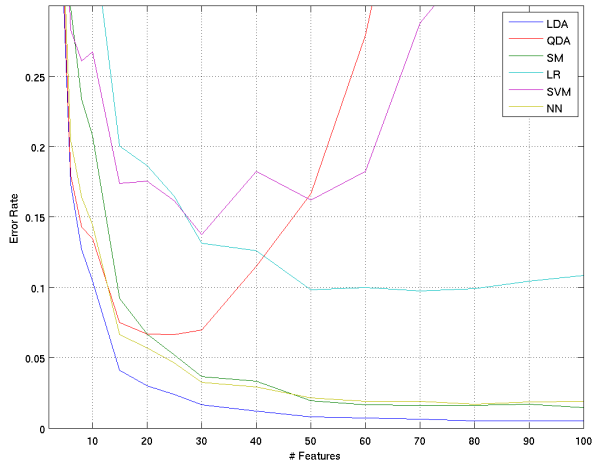
Classifier	Parameters
SM	*Max* method with half the trace for estimating the noise, 2^{nd} order polynomial expansion performed on class labels
LR	regularisation parameter $\lambda = 100$
SVM	linear kernel with 10-fold cross validation used to estimate cost C
NN	single layer with 100 hidden units and $\lambda = 100$

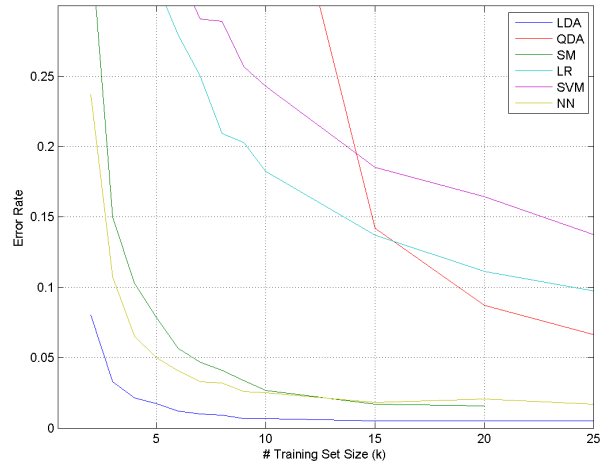




Comparison - AES (multiclass)

Classifier	Parameters
SM	*Max* method with half the trace for estimating the noise, 6^{th} order polynomial expansion performed on bit decomposition of class values
LR	*One-v-All* multi-class, regularisation parameter $\lambda = 1$
SVM	*One-v-All* multi-class with the number of "All" samples restricted to $\times 10$ of the "One", Gaussian kernel with 10-fold cross validation used to estimate cost C and σ
NN	single layer with 100 hidden units and $\lambda = 1$





Conclusions

- In geneal, LDA can be expected to perform at least as well as QDA where the noise has a Gaussian distribution, for a lower computational cost due to the pooling of covariance matrices.
- Single-trace attacks are feasible against both symmetric and asymmetric algorithms hence countermeasures based on restricting the number of available attack traces must be supplemented with other approaches.
- Neural networks can have comparable performance to LDA, and are relatively robust when *reasonable* parameters are chosen.
- Optimal feature selection is non-trivial, with subsequent impact on classifier performance varying.
  - *Junk in = junk out!!!*

Questions

neilh@eleceng.ucc.ie
n.hanley@qub.ac.uk