Usable Security & Authentication
– Implicit mobile authentication

Mike Just, Heriot-Watt University
COINS Summer School on Auth Ecosystems
31 July 2016

LEADERS IN IDEAS AND SOLUTIONS

Some preparation

In preparation for the afternoon session, download and read the following paper:

- "Data Driven Authentication: On the Effectiveness of User Behaviour Modelling with Mobile Device Sensors", in *MoST 2014*.
  - http://arxiv.org/abs/1410.7743
- Can also find this on my webpage at
  - www.justmikejust.co.uk/publications

Behavioural authentication

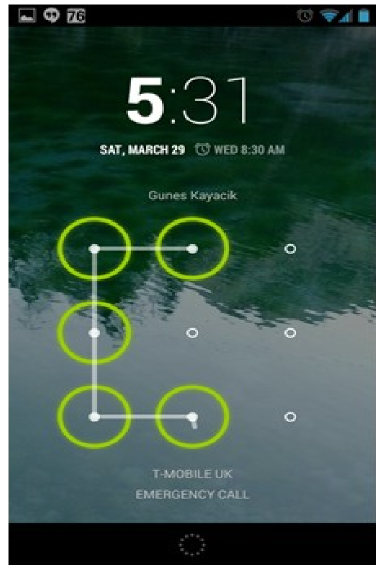
- Challenges with explicit forms of authentication
  - Knowledge: Creation and recall of information
  - Possession: Issuance and retention
  - Physiological: Can be explicit or implicit (behaviour)
- Let's focus on implicit
  - Capturing and verifying natural user actions
  - Discussed for decades, and today's “big data” helps

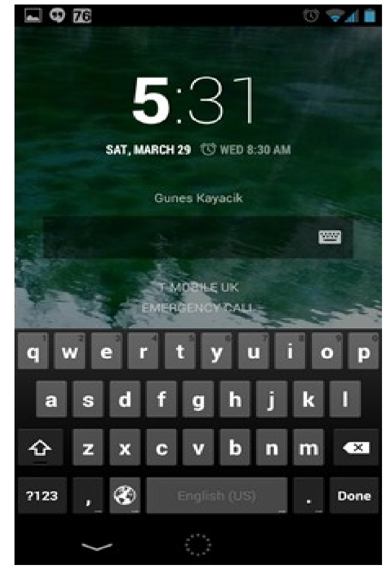
Behavioural authentication

- Several interesting forms of implicit behaviour
  - Talking, handwriting, walking (gait), etc.
  - Online behaviour, such as location, IP [NDSS'16]
  - All of which are interesting to study
- But let's follow a different approach
  - What's a good source of data to use?
  - What's a resource that needs protection

HERIOT WATT UNIVERSITY

Mobile device security





- Payment functions, sensitive data
- BYOD, enterprise security
- PINs, patterns, passwords de-facto methods

Insecurity & unusability

- Most people don't lock their smartphones
  - 64% (Consumer Reports, '13)
- Many who do lock, find it annoying
  - 40-47% (Harbach et al., '13; Egleman et al., '14)
- Current smartphone protections are a failure
  - Security: No protection for most users
  - Usability: Annoying experience for many users

Implicit authentication for mobile devices?

- Current authentication designed for fixed PCs
- Modern mobile devices offer rich new services
  - Many applications, several sensors
- People have strong connection with devices
  - Much data is collected
- Many interactions: sensor-based & data-driven
- Why not implicit device authentication?

Sensor-based authentication

- Basic idea
  - Use sensor data to train device
  - Result is a user profile for the device
  - Subsequent sensor input is compared to profile
  - If match: no PIN/pattern/password required
- Ideal result
  - “No lock” users: security better, usability “ok”
  - “Lock” users: security “ok”, usability better

Many questions (1)

- Will fewer explicit authentications be less annoying to users who currently lock their devices?
- Will fewer explicit authentications encourage non-adopters to lock their devices?
- Can devices be trained to recognize users? If so, how long does training take? Re-training?
- What is the impact on security? Would devices become more vulnerable? Would users feel more/less secure?

Many questions (2)

- Collecting sensor data consumes resources. Can today's devices do this effectively, without a noticeable impact on resources (e.g., battery)?
- Are some sensors more effective than others? If so, how effective is it to profile user behaviour?
- How often must sensors be sampled? How does the sampling rate impact battery consumption, and security?

Many ~~problem~~ interesting areas

- Modeling behaviour from sensors
- Security
- Resource consumption
- Usability, adoption

Lecture outline

- **Modeling behaviour from sensors**
- Security
- Resource consumption
- Usability, adoption

Multiple models

- NB: The following slides present a variety of research that I've tried to assimilate
- Pubs: MoST '14, PerCom '15, MobileHCI '15
- Varying factors
  - Data: Cell vs. WiFi for location
  - Participant sizes varied for each experiment
  - Models: Decision trees vs. simple histograms

Multiple models

- Following slides will focus mostly on
  - Histogram based model
  - Larger sensor dataset
- Data collection app
  - Same one used across all cases
  - Built our own

Sensors & datasets

- Sensors
  - Location: Cellular, WiFi
  - Ambient: Light, magnetometer, microphone
  - Behavioural: Accelerometer, app usage, rotation
- Datasets
  - Collected from real-world behaviour
  - Approximately 30 participants, several weeks

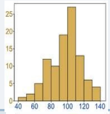
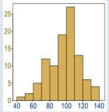
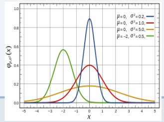
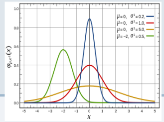
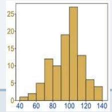
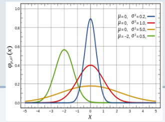
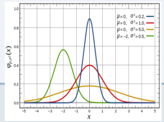
Data representation

Time	Location	Probe	Values
1396184023	Cell1	Wifi	Wifi1, Wifi2
1396184077	Cell1	App	App1, App3, App4
1396184192	Cell1	Light	15 lux
1396184201	Cell2	Noise	57 dB
1396184227	Cell3	Magnetic	[+0.1, +0.5, +0.3]
1396184301	Cell3	Rotation	[+0.2, +0.7, -0.1]

- Cell tower ID observed at time t
- Sensors provide single or multiple samples
- Discrete or continuous data

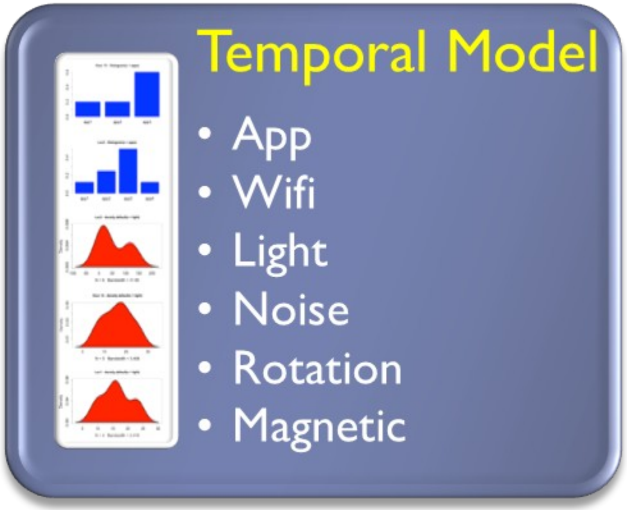
Modeling behaviour (1)

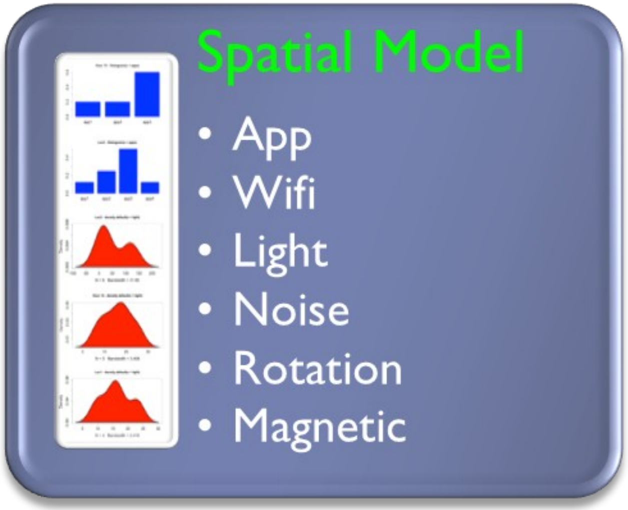
- Sensor readings for two “anchors”
  - Location (spatial)
  - Time (temporal)
- **User profile** consists of sensor readings for different locations and times

Cell=112561	App	
	Wi-Fi	
	Light	
	Noise	
	Cpu	
	Rotation	
	Magnetic	
Cell=1122134	…	

Modeling behaviour (2)

- When building a profile, inputs are collected for each sensor in both the temporal and spatial models and represented as probability distribution functions (pdfs)
- When validating a score, sensor data is compared to each profile pdf, for both location and time



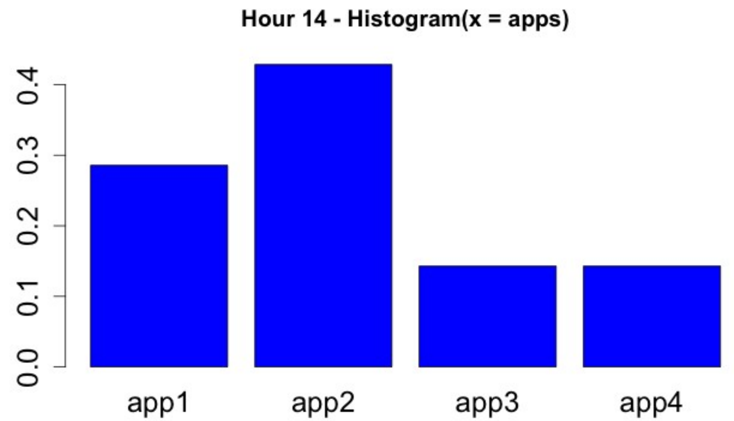


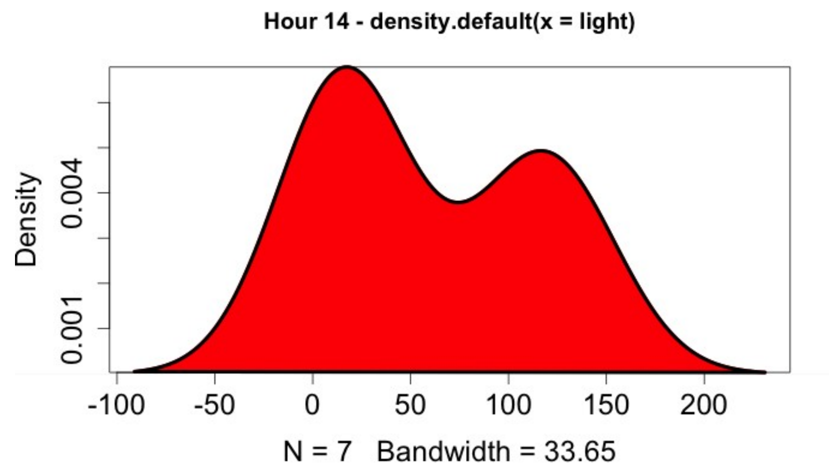
Building profiles

	App	Wifi	Light	Noise	Rot	Mag	
	app1	wifi1	15	55	[.1, .3, .5]	[.1, .3, .5]	Loc1
14:15	app2	wifi1	17	89	[.6, .2, .9]	[.0, .2, .2]	
	app2	wifi3	23	85	[.7, .3, .1]	[.1, .0, .3]	Loc1
14:30	app1	wifi4	10	79	[.9, .5, .6]	[.2, .1, .8]	
	app3	wifi5	22	66	[.2, .6, .2]	[.1, .0, .9]	Loc2
14:45	app4	wifi2	29	50	[.9, .7, .9]	[.0, .0, .1]	
	app2	wifi2	30	54	[.0, .1, .8]	[.4, .3, .2]	Loc2
15:00	app2	wifi2	17	59	[.1, .8, .3]	[.2, .2, .4]	
	app1	wifi6	7	65	[.4, .9, .4]	[.3, .1, .7]	
15:15	app3	wifi2	12	77	[.5, .0, .5]	[.1, .0, .3]	Loc2
	app3	wifi7	19	89	[.6, .2, .1]	[.0, .4, .2]	
	app3	wifi2	25	90	[.3, .4, .9]	[.0, .1, .1]	

Building profiles (temporal)

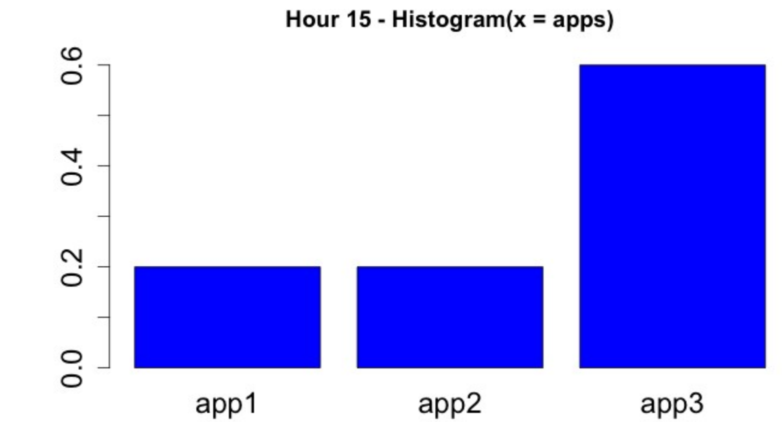
Time	App	Light	Location
	app1	15	Loc1
14:15	app2	17	
	app2	23	Loc1
14:30	app1	10	
	app3	22	Loc2
14:45	app4	29	
	app2	30	
15:00	app2	17	Loc2
	app1	7	
15:15	app3	12	Loc2
	app3	19	
	app3	25	

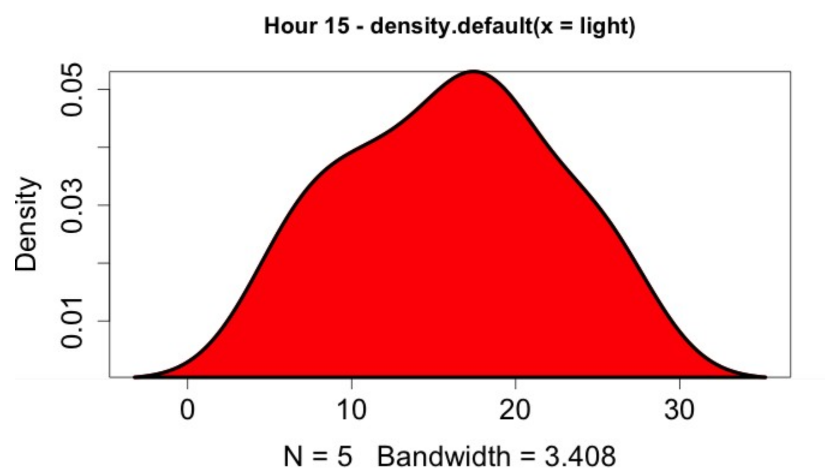




Building profiles (temporal)

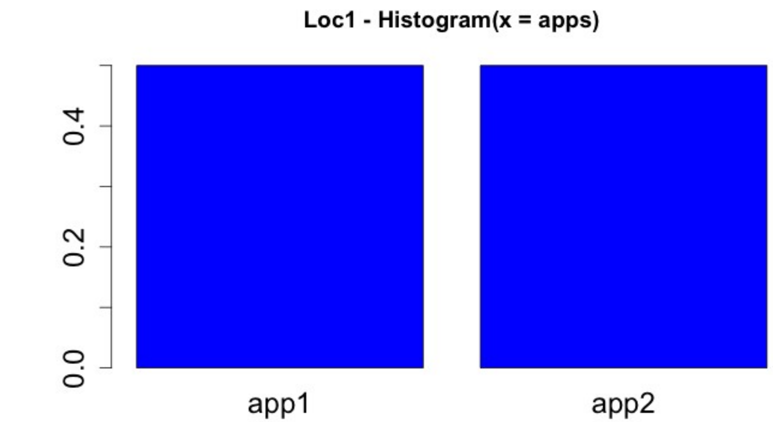
	App	Light	
14:15	app1	15	Loc1
	app2	17	
14:30	app2	23	Loc1
	app1	10	
14:45	app3	22	Loc2
	app4	29	
	app2	30	Loc2
15:00	app2	17	
	app1	7	
15:15	app3	12	Loc2
	app3	19	
	app3	25	

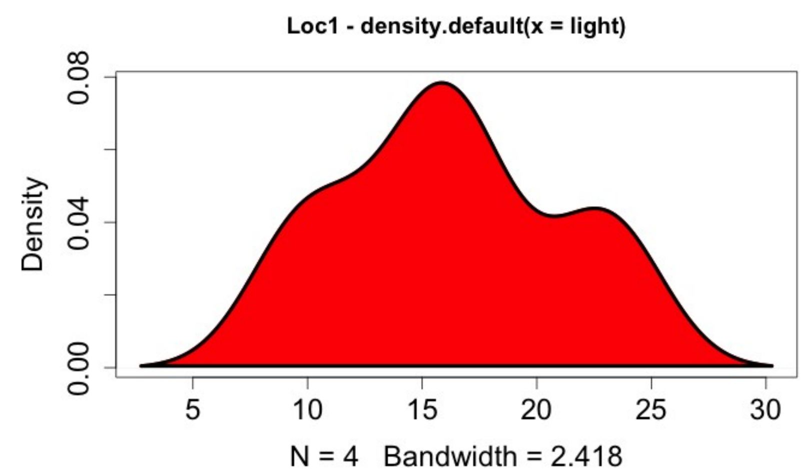




Building profiles (spatial)

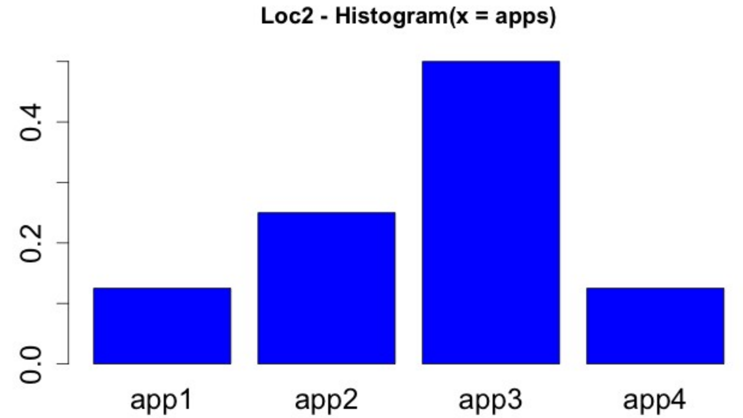
Time	App	Light	Location
14:15	app1	15	Loc1
	app2	17	
14:30	app2	23	Loc1
	app1	10	
14:45	app3	22	Loc2
	app4	29	
	app2	30	Loc2
15:00	app2	17	
	app1	7	Loc2
15:15	app3	12	
	app3	19	
	app3	25	

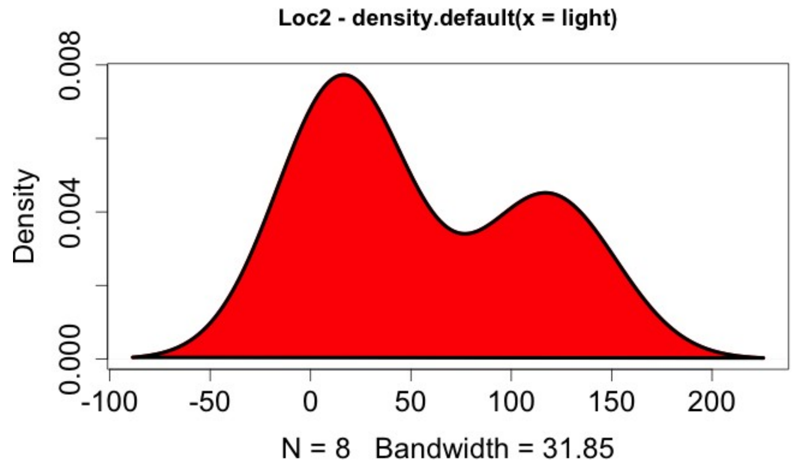




Building profiles (spatial)

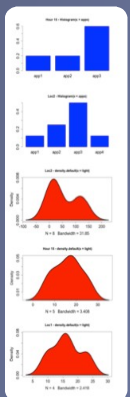
Time	App	Light	Location
14:15	app1	15	Loc1
	app2	17	
	app2	23	Loc1
14:30	app1	10	
	app3	22	Loc2
14:45	app4	29	
	app2	30	Loc2
15:00	app2	17	
	app1	7	Loc2
15:15	app3	12	
	app3	19	
	app3	25	





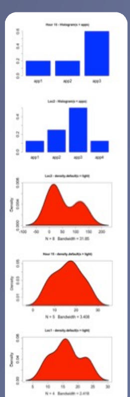
HERIOT WATT UNIVERSITY

Computing comfort



Temporal

- App
- Wifi
- Light
- Noise
- Rotation
- Magnetic



Spatial

- App
- Wifi
- Light
- Noise
- Rotation
- Magnetic

App	Light
app1	15
app2	17
app2	23
app1	10
app3	22
app4	29
app2	30
app2	17
app1	7
app3	12
app3	19
app3	25

14:15 Loc1

14:30 Loc1

14:45 Loc2

15:00 Loc2

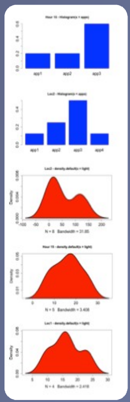
15:15 Loc2

```
p( light = 17 |
   location = Loc1)
```

```
p( light = 17 |
   hour = 14)
```

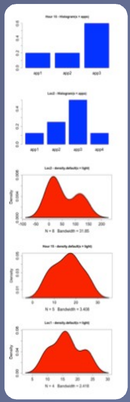
Each input creates a temporal and a spatial conditional probability (or comfort).

Computing comfort



Temporal

- App
- Wifi
- Light
- Noise
- Rotation
- Magnetic



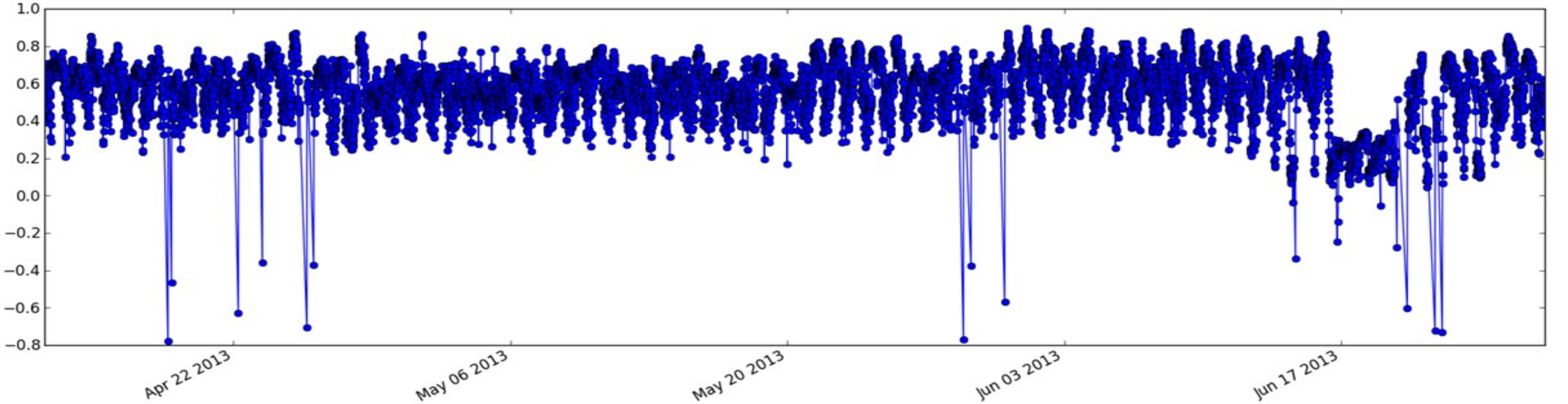
Spatial

- App
- Wifi
- Light
- Noise
- Rotation
- Magnetic

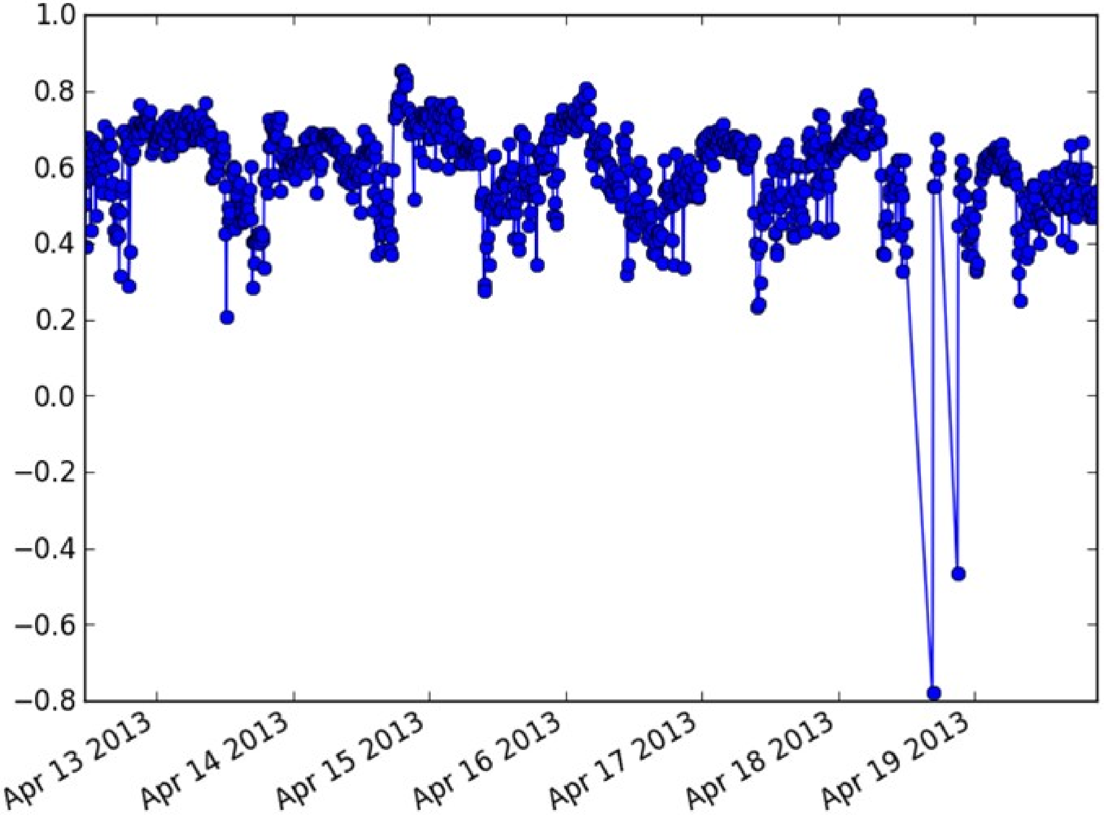
App	Wifi	Light	Noise	Rot	Mag
app1	wifi1	15	55	[.1, .3, .5]	[.1, .3, .5]
app2	wifi1	17	89	[.6, .2, .9]	[.0, .2, .2]
	wifi3		85	[.7, .3, .1]	
	wifi4			[.9, .5, .6]	

- Data from sensors compared to models
- Each event produces two comfort scores
  1. Score from each sensor is aggregated into a sensor score first
  2. Scores from sensors are aggregated into temporal and spatial scores
  3. Overall comfort score, is computed by aggregating temporal & spatial scores

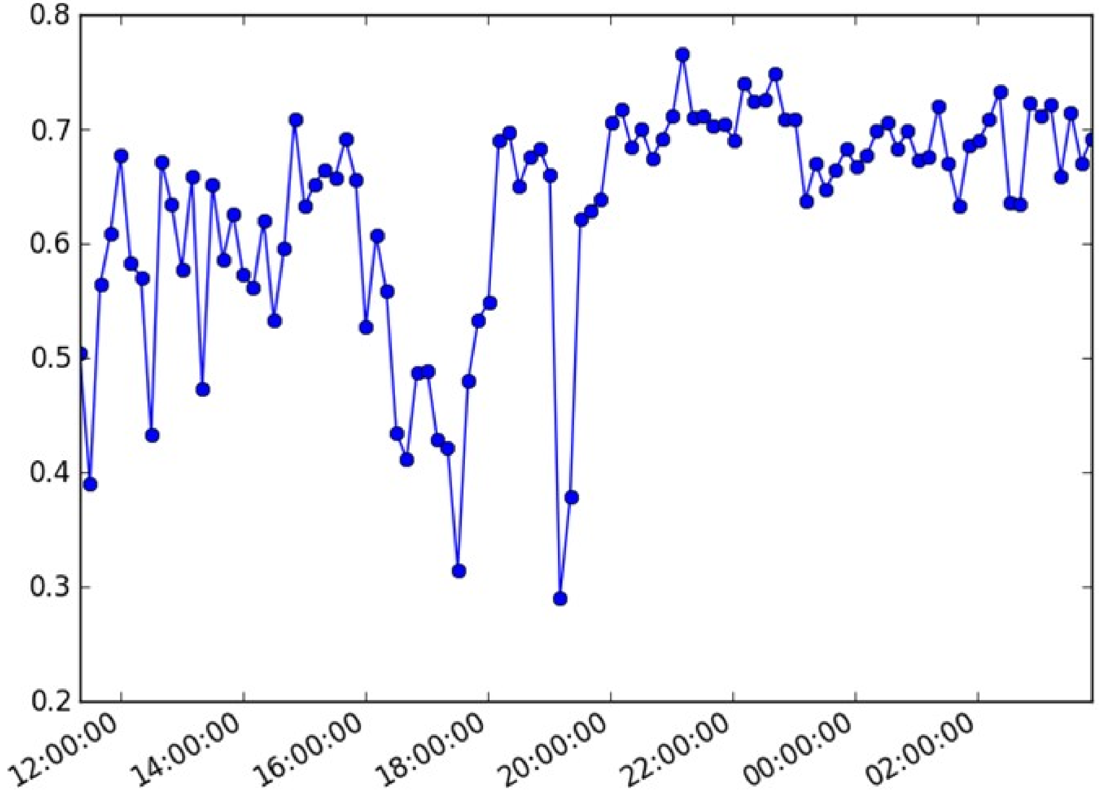
Computing comfort (5 months)



Computing comfort (1 week)

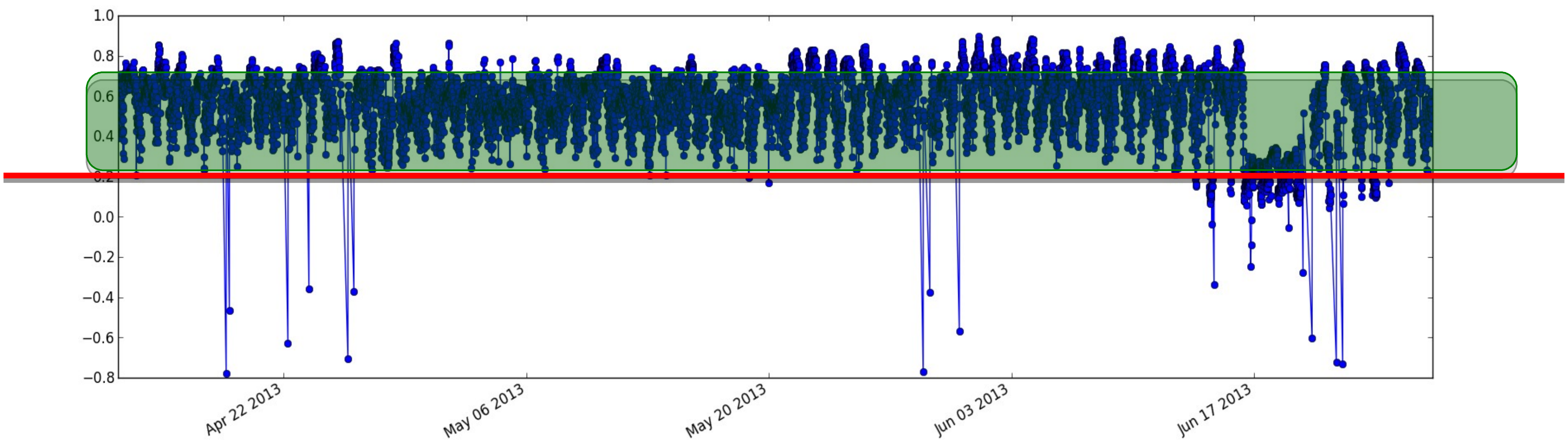


Computing comfort (1 day)



Setting detection threshold

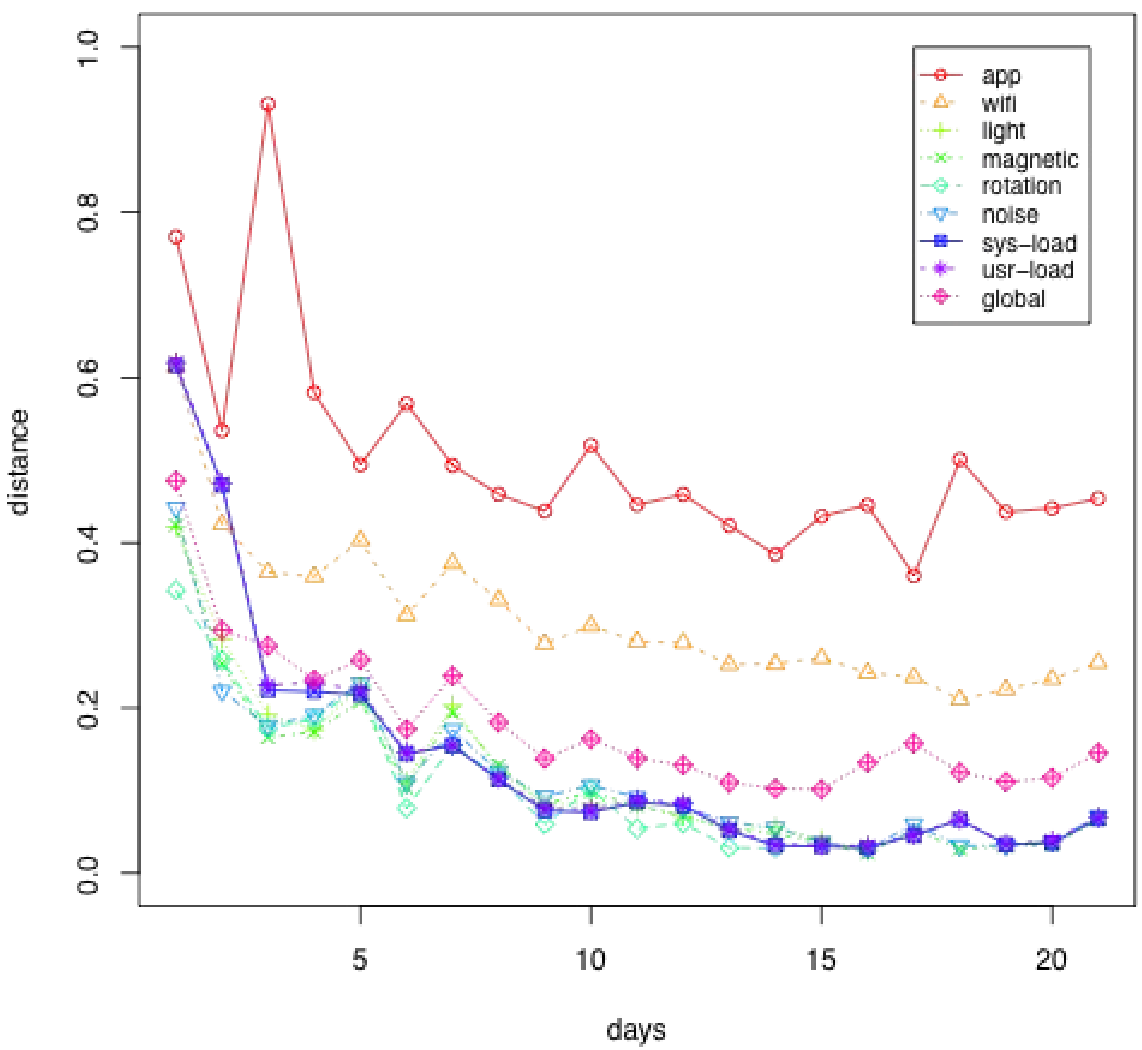
- Set automatically based upon past observations and performance to balance security and usability
  - Balance of FAR and FRR



Training duration & convergence

- How long does it take to train a device?
- Can measure comfort score changes between days
  - Following graph compares between day N and N-1 using Levenshtein distance

Convergence results



Convergence results

	Convergence (Global)	Convergence (Temporal)	Convergence (Spatial)
User 1	9 days	9 days	9 days
User 2	10 days	8 days	10 days
User 3	3 days	9 days	1 days
User 4	9 days	7 days	9 days
User 5	9 days	8 days	14 days
User 6	9 days	5 days	11 days
User 7	6 days	6 days	8 days

Convergence results

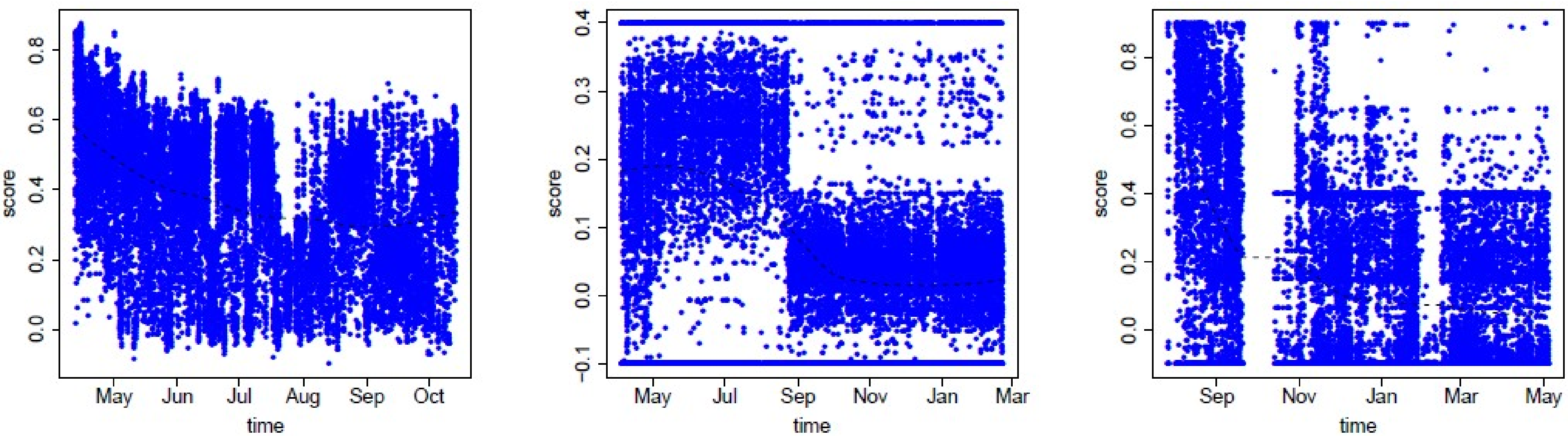
Profile convergence	GCU	Rice	MIT
Global	9.00 days	10.07 days	12.35 days
Temporal	7.00 days	10.00 days	12.18 days
Spatial	9.50 days	5.40 days	10.58 days

Convergence results

- Typically 3-5 days to establish rough estimate of user model
  - Familiar locations, available networks, favourite apps
- 1-2 weeks to establish a finer model
  - Ordering of locations, ordering of WiFi, etc.
- Retraining
  - Some degradation after about 6 months

Behaviour drift

- Drift in scores (and hence, behaviour) in examples users from all three datasets over 6 months



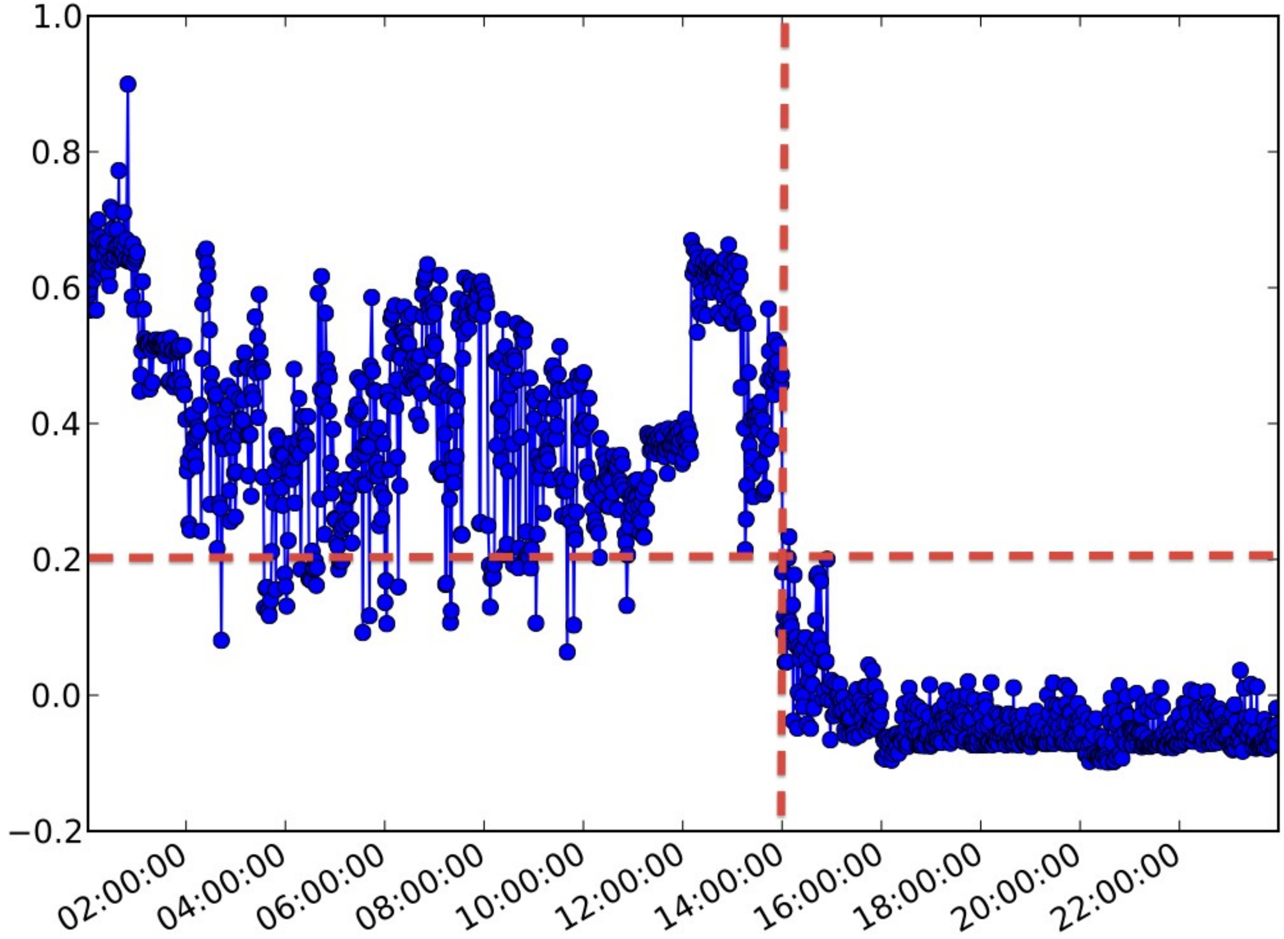
Lecture outline

- Modeling behaviour from sensors
- **<u>Security</u>**
- Resource consumption
- Usability, adoption

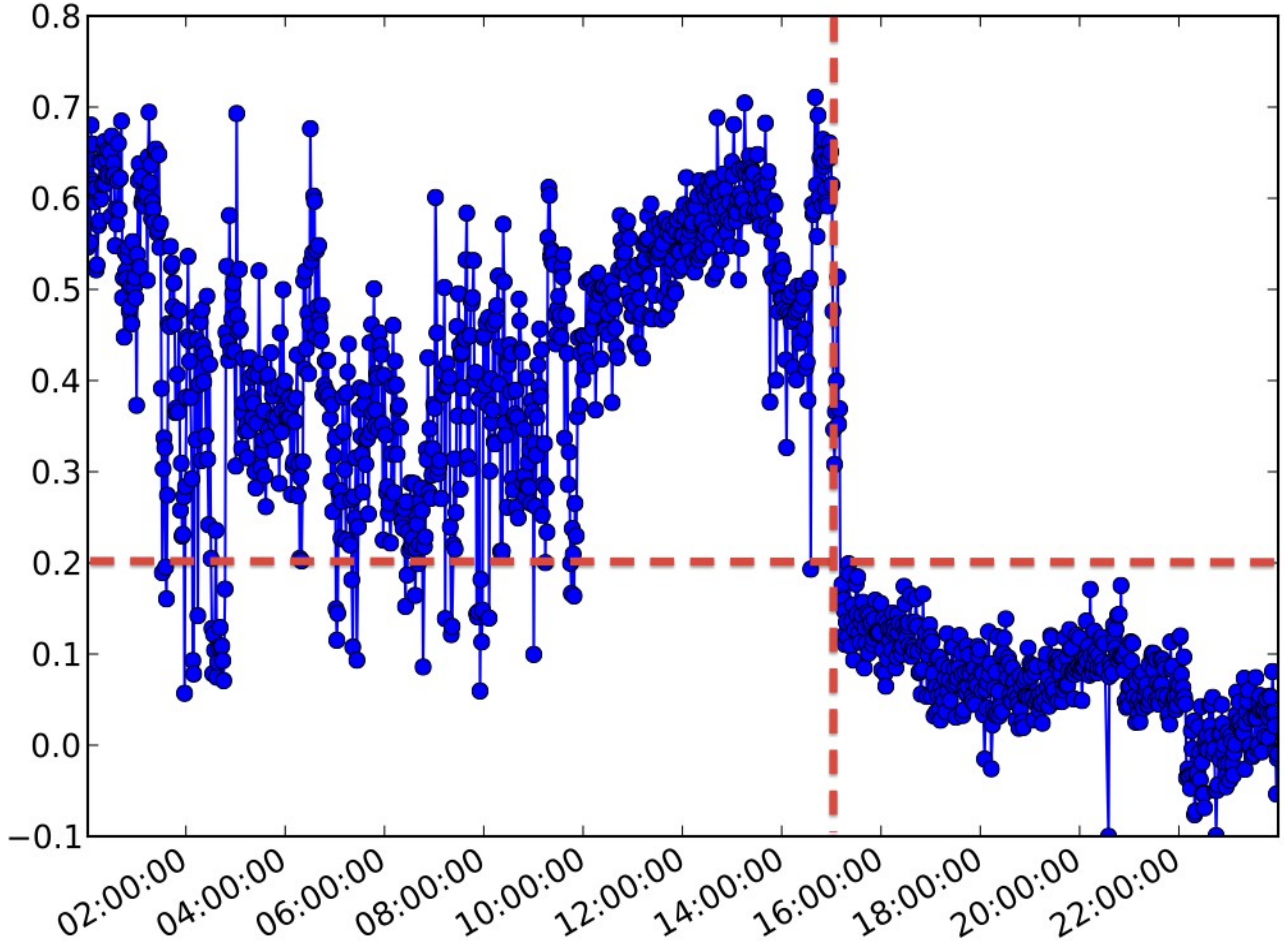
Security model

- Four attack profiles
  - Uninformed. Low knowledge.
  - Informed. Some knowledge.
  - Outsider. Low access.
  - Insider. Some access
- Owner uses device for a few weeks
  - Models are built
  - Threshold is determined
- Attacker behaviour simulated by an individual who assumes each of the attack profiles

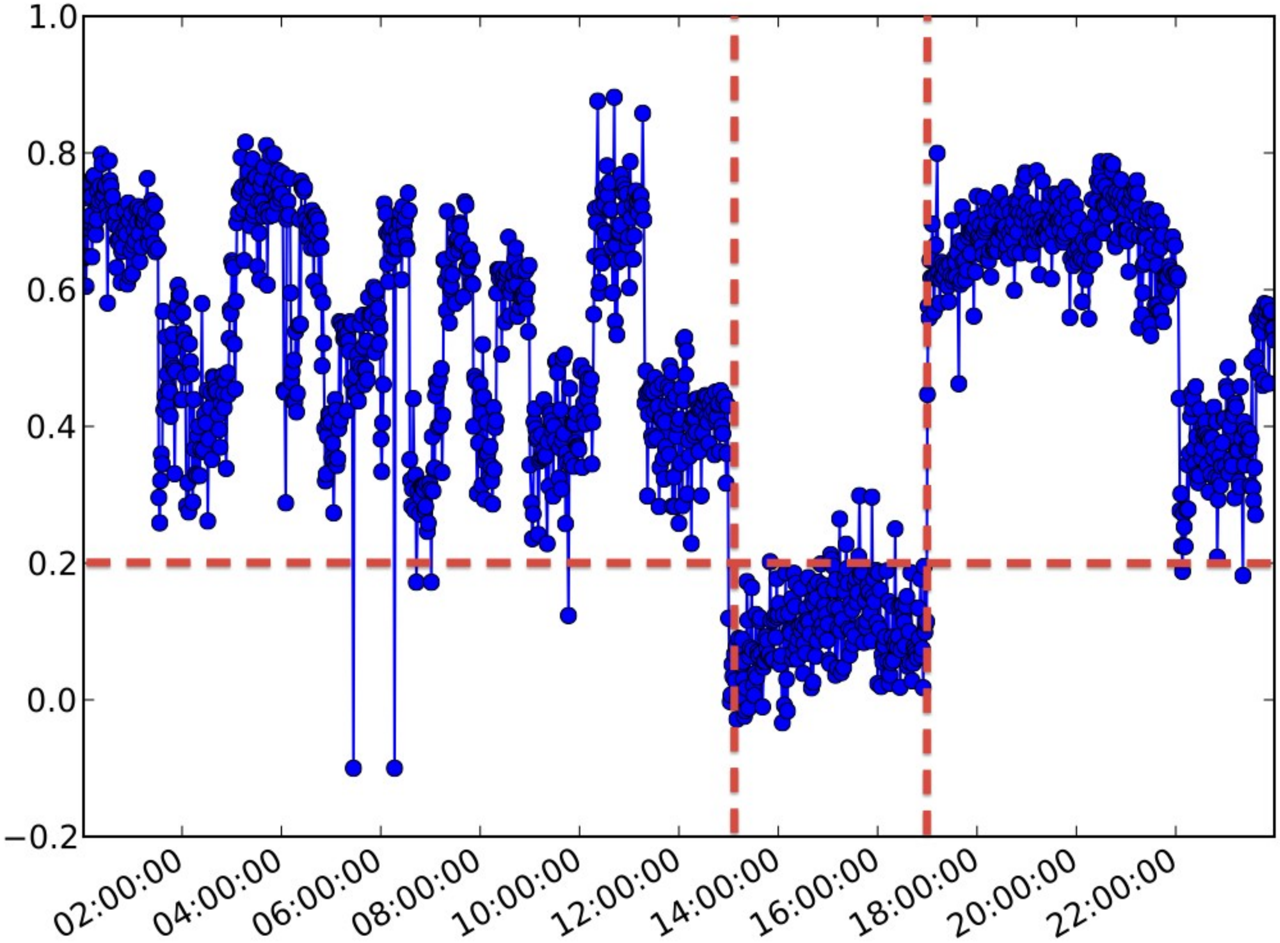
1: Uninformed outsider



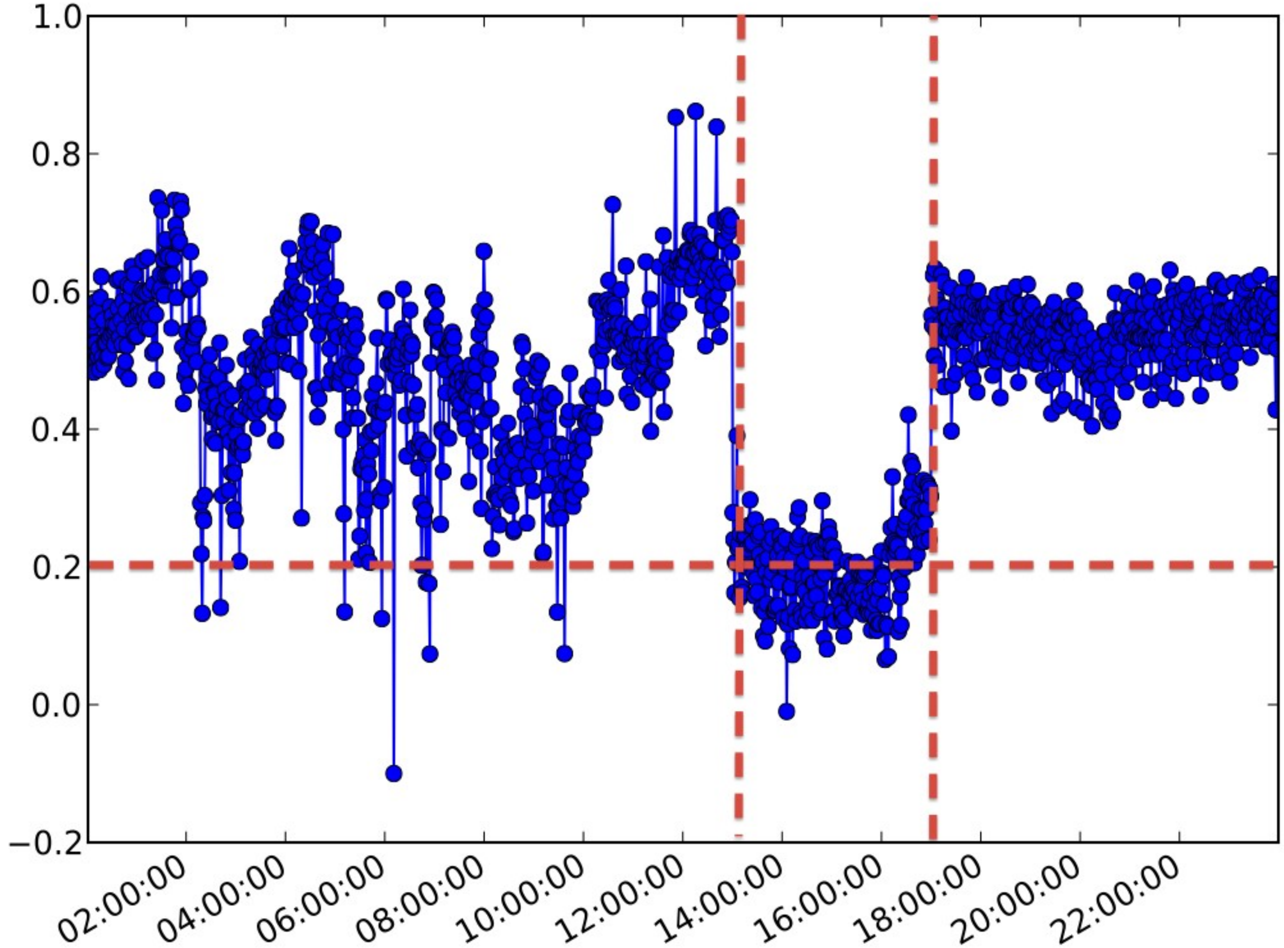
2: Informed outsider



3: Uninformed insider



4: Informed insider



Lecture outline

- Modeling behaviour from sensors
- Security
- **Resource consumption**
- Usability, adoption

Mobile device consumption

- Sensor use offers more than PC
  - Rich interactions: user, device, environment
- Sensors also consume resources (battery)
- Battery capacity increases, but demand is high
  - Samsung Galaxy S3-S5: 2100-2600-2800mAh
- Some users charge devices multiple times a day

Related work (resource consumption)

- Minimise use of “high drain” sensors
  - (Wu et al., 2013; Paek et al., 2010; Zhuang et al., 2010; Wang et al., 2009)
- Innovative solutions
  - Shared caching (Hopfner et al., 2003)
  - Speculative sensing (Nath et al., 2012)
  - Selective sampling (Krause et al., 2005)
  - Adaptive sampling (Rachuri et al., 2012)
- Optimising for security not considered

Related work (sensor authentication)

- Learning user behaviour from sensor data
  - (Kayacik et al., 2014; Gupta et al., 2012; Shi et al., 2011)
- Detect anomalies when user behaviour doesn't match profile
- Typically assumes fixed sampling rate
- No consideration of battery consumption

Battery consumption - Method

- Hardware
  - 2 Samsung Galaxy S4
- Method
  - Both devices carried through “daily routine” for four full days
- Tools
  - Our sensor data collector
  - PowerTutor: measure mW consumed by collector

Battery consumption – Results (1)

Rate	Battery Consumption (mAh)
1 min	10.83
5 min	2.72
10 min	1.04
15 min	0.71
20 min	0.45

- Proportional drop in consumption as sampling frequency decreases

Battery consumption – Results (2)

Active Sensor	Battery Consumption (mAh)
Accelerometer	2.08
Apps Usage	1.46
GPS	2.31
Light	0.86
Magnetic Field	0.49
Microphone	1.71
Gyro	2.01
Wi-Fi + Cell	1.62

- Sampling rate = 1 min
- Some high consumers

Battery consumption – User impact

Rate	Light Drain	Medium Drain	High Drain
baseline	260.00h	28.89h	10.40h
1 min	124.80h (52.0%)	25.79h (10.7%)	9.97h (4.1%)
5 min	204.39h (21.4%)	28.04h (2.9%)	10.29h (1.1%)
10 min	235.30h (9.5%)	28.55h (1.2%)	10.36h (0.4%)
15 min	242.79h (6.6%)	28.66h (0.8%)	10.37h (0.3%)
20 min	248.85h (4.3%)	28.74h (0.5%)	10.38h (0.2%)

- Impact to light, medium and high users
- Significant impact for light and medium

Attack detection - Method

- Attacks
  - Uninformed adversary
  - Informed adversary
  - Varying knowledge (e.g., app usage) and access (e.g., locations)
- Data sets
  - Normal usage (3 weeks) for 4 users
  - Attack scenarios from 1 user

Attack detection results - All sensors, Uninformed

Rate	Uninformed Attack		Normal
	Detection Time	Detection Rate	False Positives Rate
1 min	183s (~3min)	92.07%	1.39%
5 min	3591s (~1hr)	92.10%	0.72%
10 min	4790s (~1.3hr)	92.98%	1.45%
15 min	5406s (~1.5hr)	96.42%	3.26%
20 min	5987s (~1.6hr)	95.65%	1.47%

- Detection time unacceptable for >= 5 minute sampling
- Detection rate not affected

Attack detection results - All sensors, Informed

Rate	**Informed Attack**		**Normal**
	Detection Time	Detection Rate	False Positives Rate
1 min	1657s (~27min)	28.82%	1.39%
5 min	6012s (~2.6 hr)	20.00%	0.72%
10 min	Undetected		1.45%
15 min	Undetected		3.26%
20 min	Undetected		1.47%

- Attacks undetected for >=10 min sampling rate
- Detection rate is very low

Attack detection results - Ambient/All sensors

Sensor	Uninformed		Informed		Normal
	Detection Time	Detection Rate	Detection Time	Detection Rate	False Positives
Wi-Fi	183s (~3min)	100%	1825s (~30min)	9.03%	28.10%
Noise	1020s (~17min)	59.64%	Undetected		0.66%
Magnetic	Undetected		Undetected		1.83%
Light	Undetected		3686s (~1 hr)	6.02%	40.98%
Ambience	183s (~3min)	97.36%	Undetected		1.10%
All	183s (~3min)	92.07%	1657s (~27min)	28.82%	1.47%

- 1 minute sampling rate
- No sensor sub-set does as well as all sensors

Attack detection results - Behavioural/All sensors

Sensor	Uninformed		Informed		Normal
	Detection Time	Detection Rate	Detection Time	Detection Rate	False Positives
App	183s (~3min)	100%	1290s (~21min)	80.72%	40.98%
Accel	Undetected		Undetected		0.58%
Gyro	593s (~10min)	94.73%	Undetected		5.88%
Behavioral	6233s (~1.7hr)	13.15%	1825s (~30min)	3.61%	1.03%
All	183s (~3min)	92.07%	1657s (~27min)	28.82%	1.47%

- 1 minute sampling rate
- No sensor sub-set does as well as all sensors

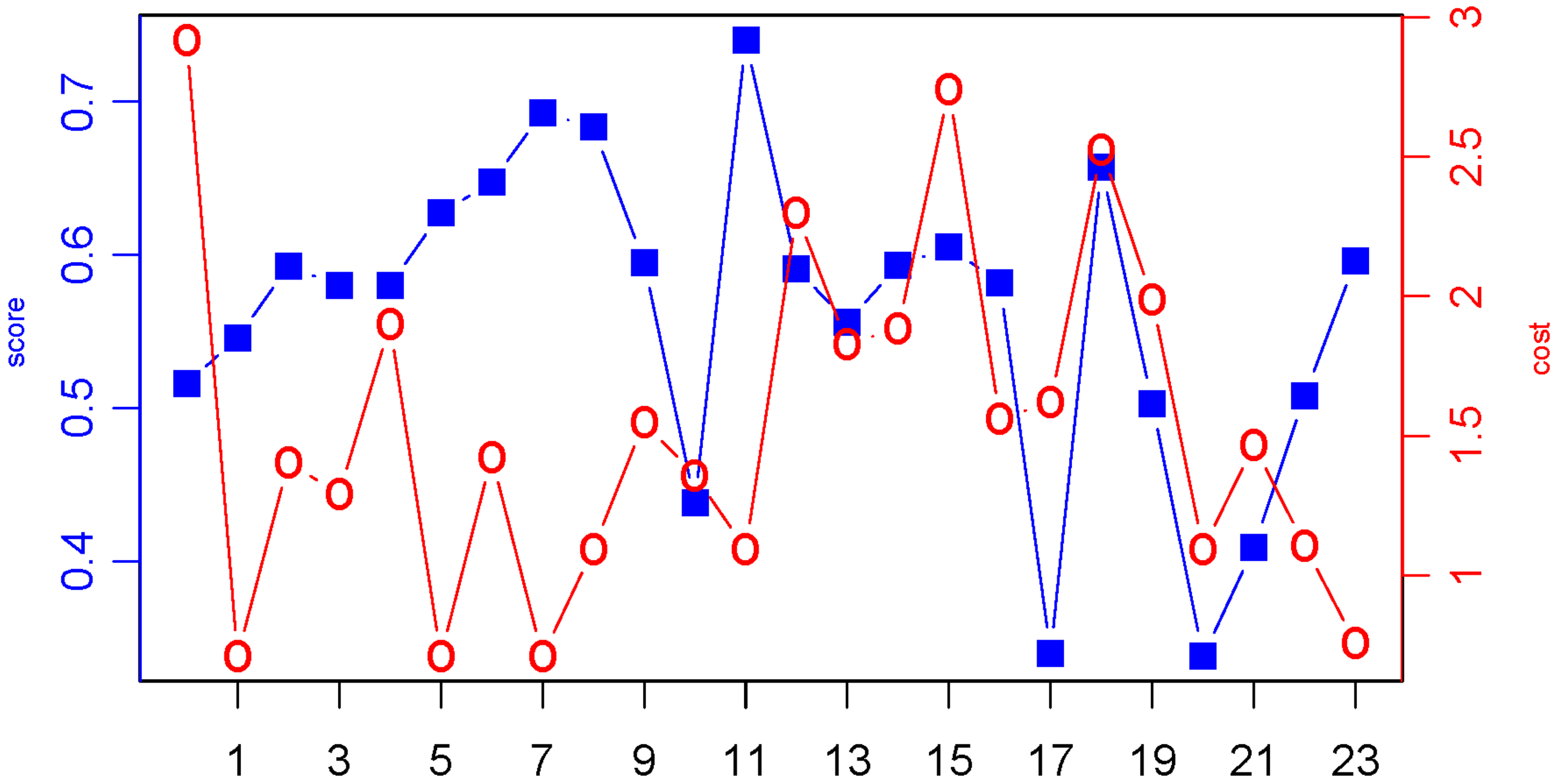
Attack detection results - Summary

- Less than 1 min sampling rate leaves device vulnerable
- No one-size-fits-all combination of sensors is satisfactory
- Possible improvement
  - Adaptively change the sampling rate
  - Only use 1 min sampling “when necessary”

Adaptive sampling - Description

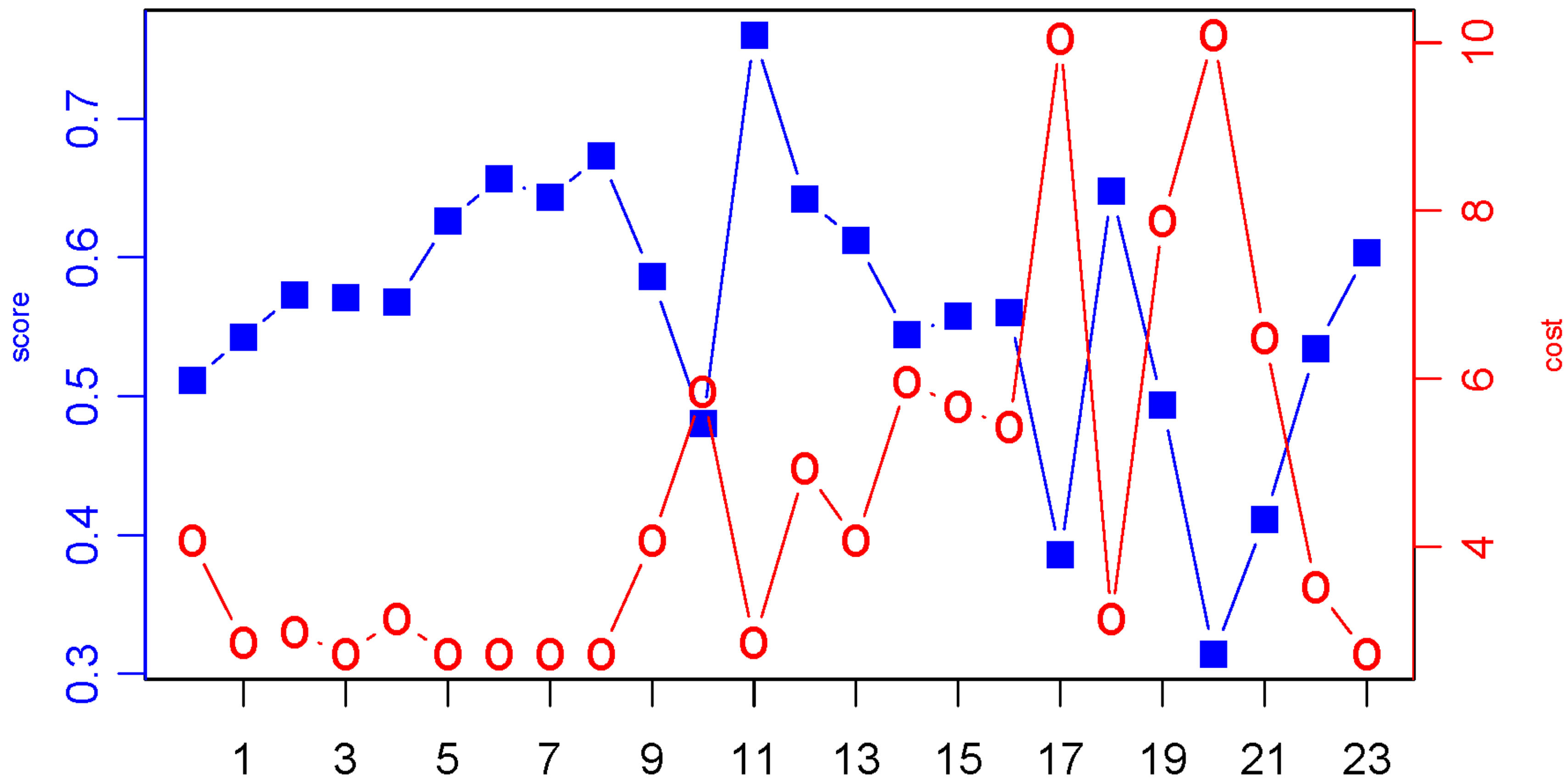
- Alter sampling rate based on triggers
- Investigated 4 adaptive sampling techniques
  - Relative change in detection score
  - Absolute detection score level
  - Context-based: Based on device location
  - Time-based: Based on hour-of-day
- Will mostly focus on first technique (above)
  - Increase ($d>0.5$)
  - Maintain ($0.1<d<0.5$)
  - Decrease ($d<0.1$)

Adaptive sampling – Normal use (1)



- Relative comfort changes trigger sampling rate

Adaptive sampling – Normal use (2)



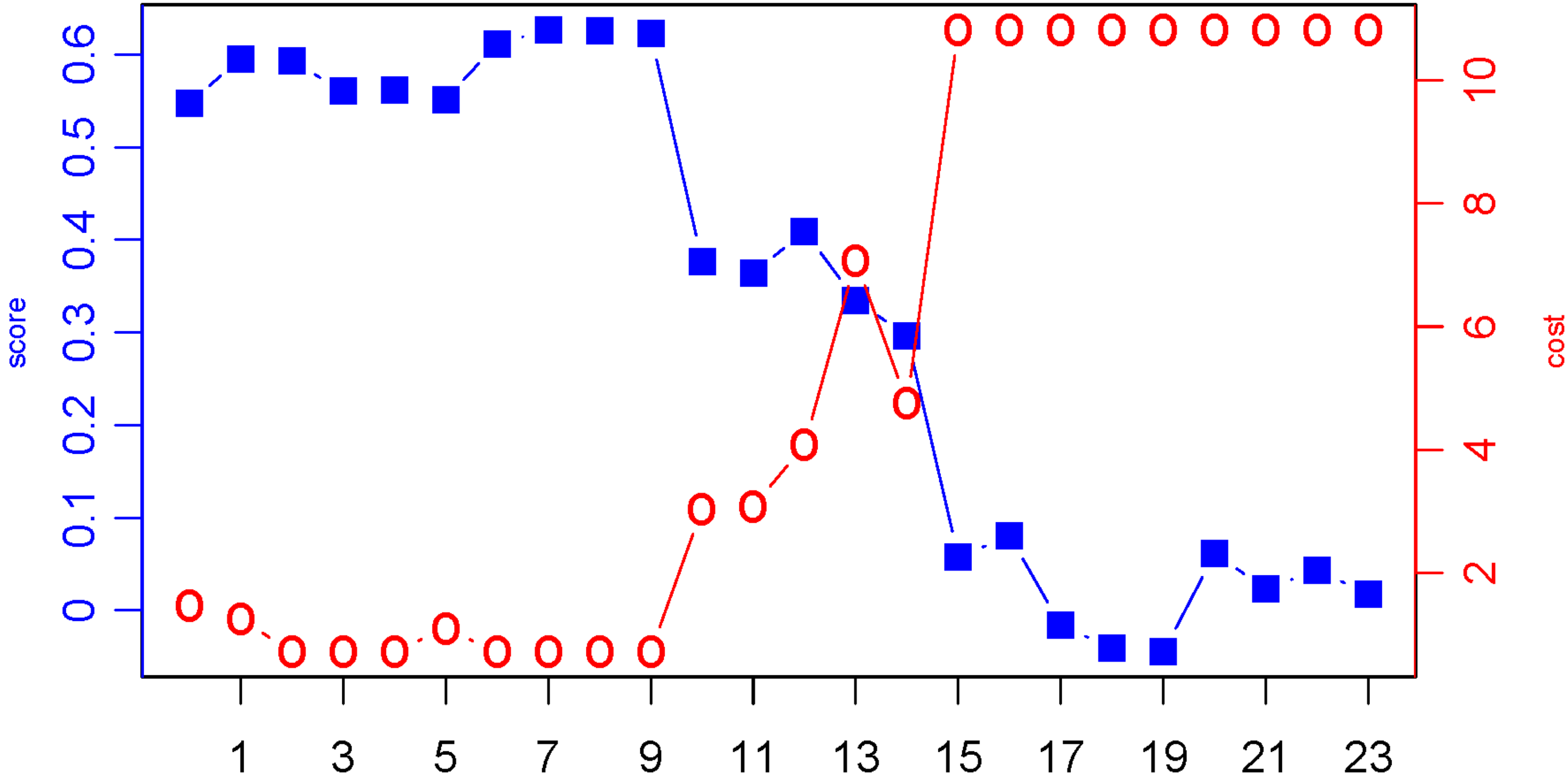
- Absolute comfort triggers sampling rate

Adaptive sampling results – Uninformed attack

Technique	Uninformed			Normal	
	Detection Time (s)	Detection Rate (%)	Battery Cost (mAh)	False Positives Rate (%)	Battery Cost (mAh)
Baseline (1 min)	183 (~3min)	92.07	10.83	1.39	10.83
Change in Detection Score	183 (~3min)	97.37	5.34	3.15	1.54

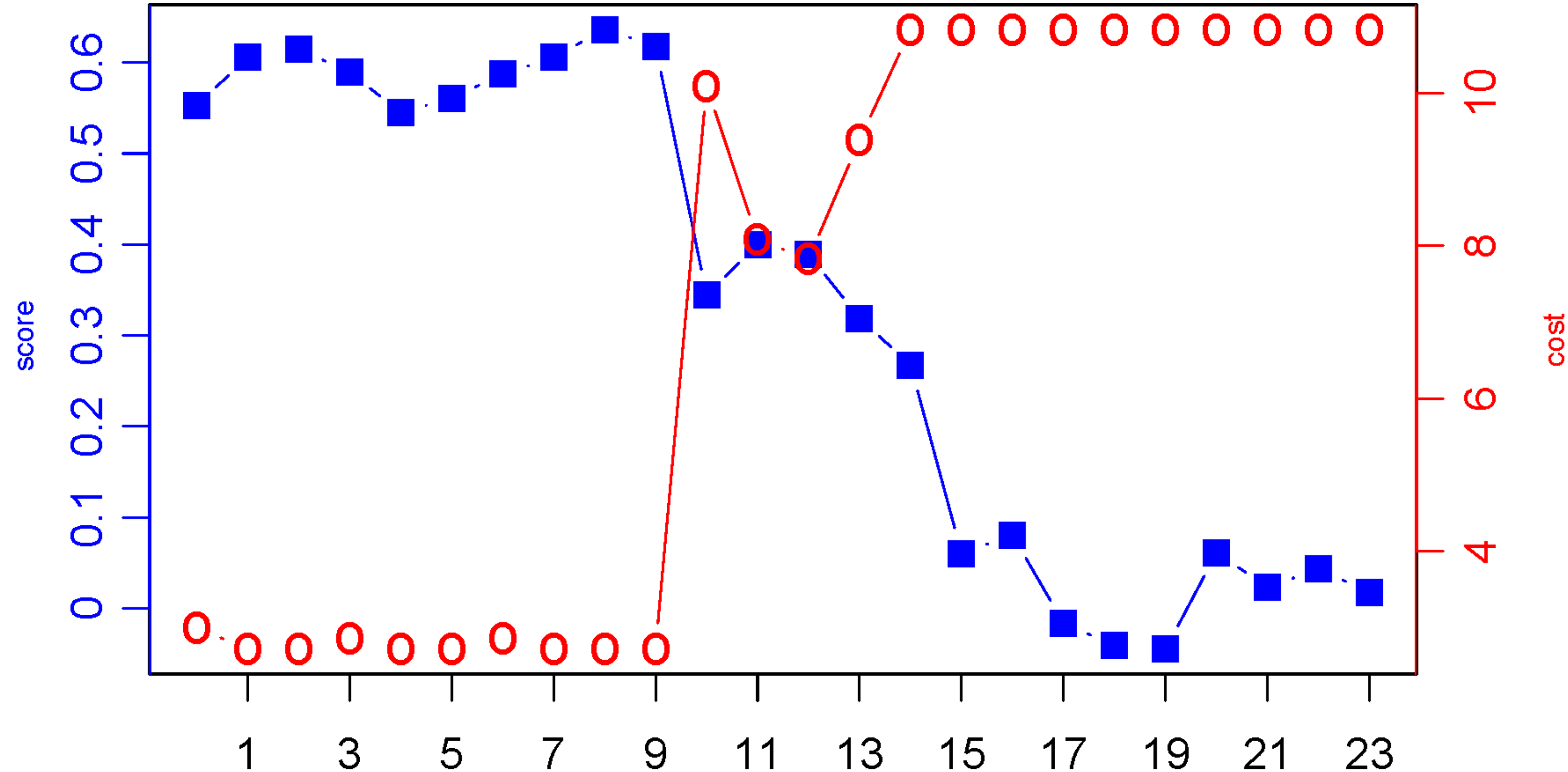
- Similar DT and DR
- Battery consumption halved during attack
- Battery consumption reduced 7-fold during normal use

Adaptive sampling results – Uninformed attack



- Relative comfort changes trigger sampling rate

Adaptive sampling results – Uninformed attack



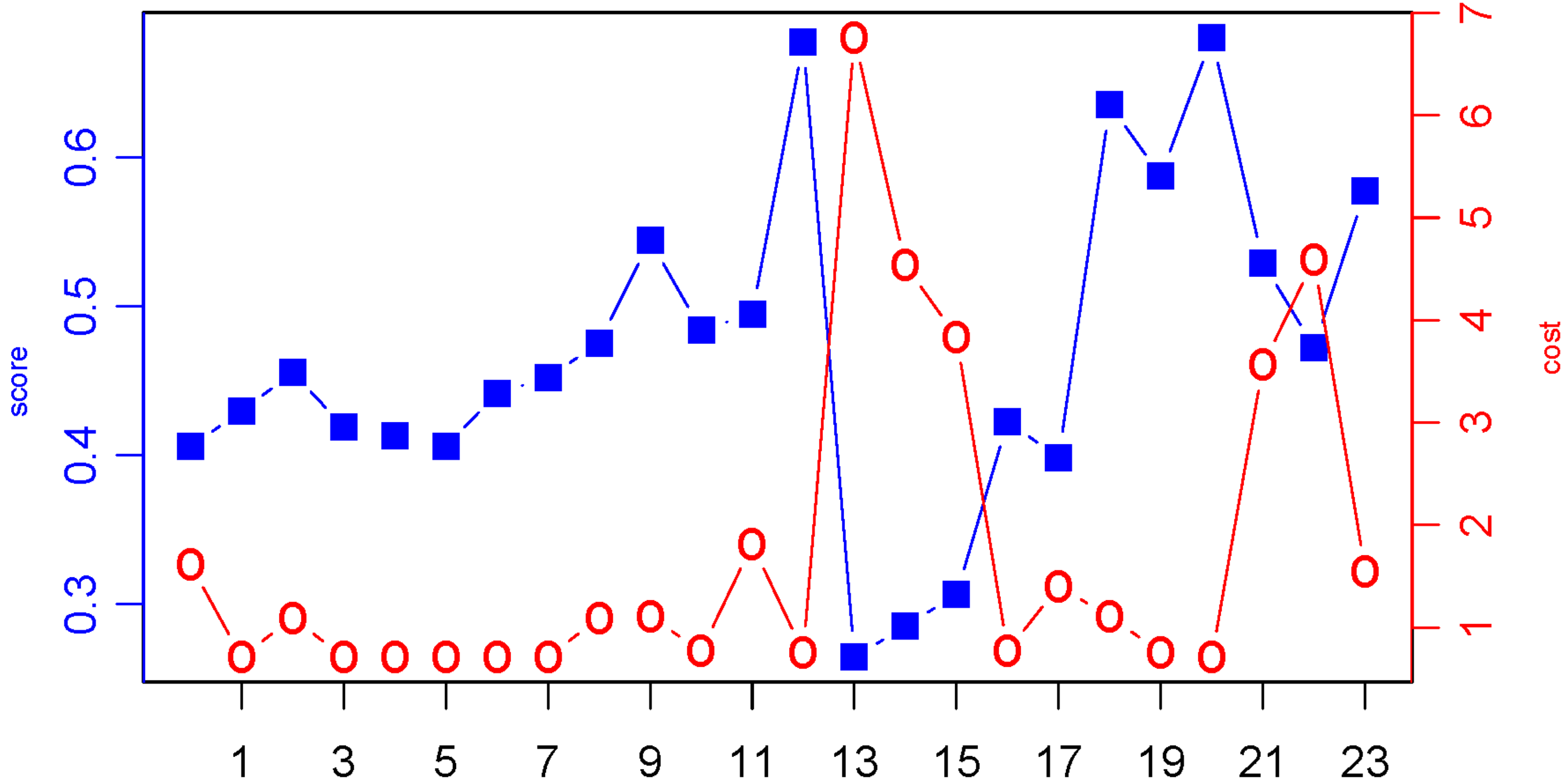
- Absolute comfort triggers sampling rate

Adaptive sampling results – Informed attack

Technique	Informed			Normal	
	Detection time (s)	Detection Rate (%)	Battery Cost (mAh)	False Positives Rate (%)	Battery Cost (mAh)
Baseline (1min)	1657 (~27min)	28.82	10.83	1.39	10.83
Change in Detection score	1206 (~20min)	36.48	1.75	3.15	1.54

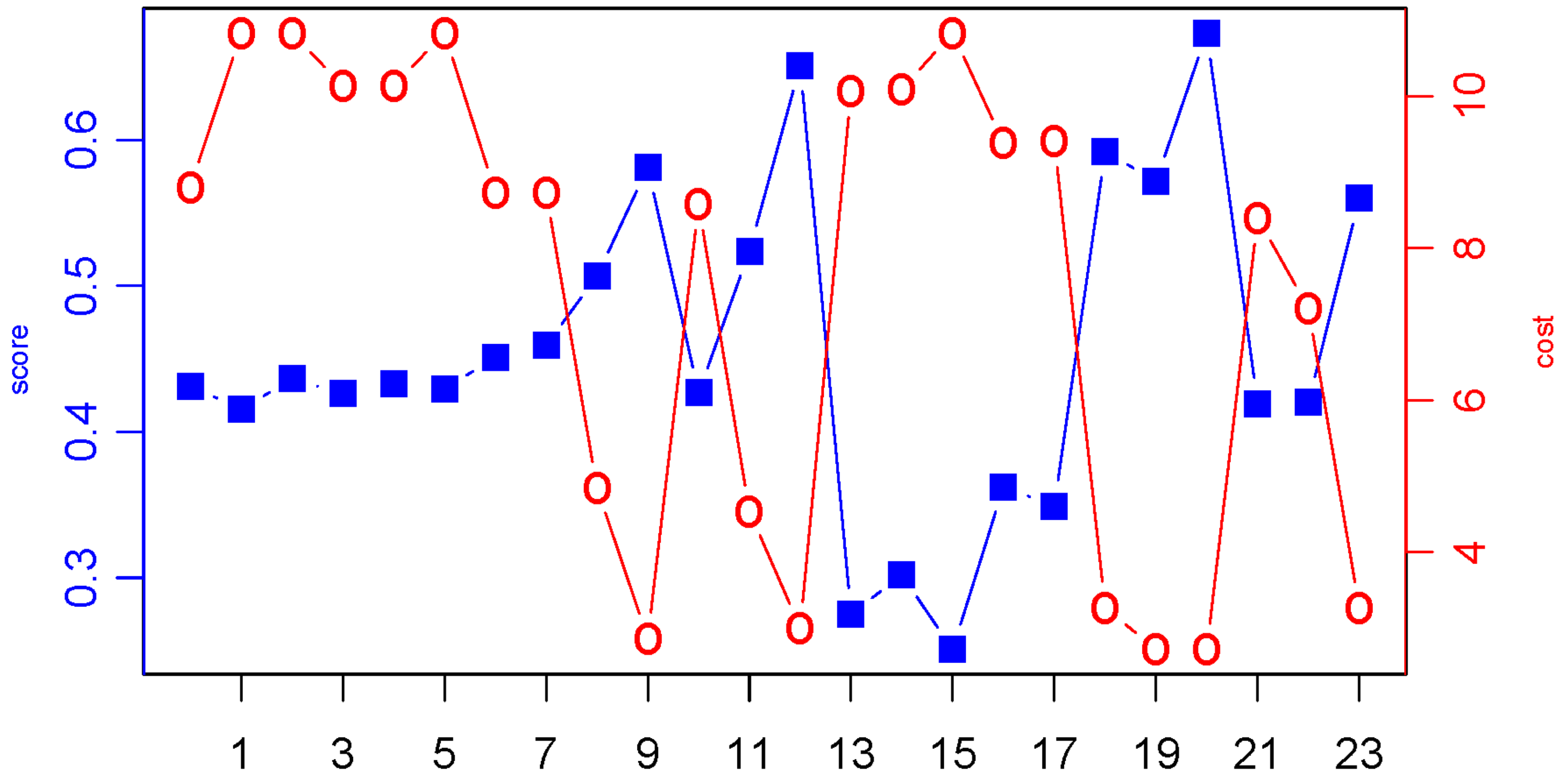
- DT and DR improved (statistical anomaly)
- Battery consumption reduced 6-fold during attack
- Battery consumption reduced 7-fold during normal use

Adaptive sampling results – Informed attack



- Relative comfort changes trigger sampling rate

Adaptive sampling results – Informed attack



- Absolute comfort triggers sampling rate

Lecture outline

- Modeling behaviour from sensors
- Security
- Resource consumption
- **Usability, adoption**

Usability study objectives

- Objective 1: How do users **perceive** proposal re: **annoyance, convenience, security**?
- Objective 2: Will users **adopt** our proposal?
- Objective 3: Which of “**No Lock**” or “**Lock**” users chose to adopt our proposal?
- Objective 4: In which **locations** was our proposal used?

Usability study method

- Phase 1: Build a profile based on user-device behaviour with 1 week of sensor data
- Phase 2: Deploy our proposal: PIN/pattern requested if sensors readings don't match
- Phase 3: Give users the option to continue with our proposal or not. And by location.

Usability evaluation

- System Usability Scale (SUS) questionnaire
- User perception questionnaire
  - Annoyance, convenience, security
- Ranking of mechanisms
  - Annoyance, convenience, security
- Efficiency results (empirical)
  - Number of logins
  - Time taken to login

Efficiency results (1)

Group	Phase I	Phase II	Phase III
“No Lock”	0 of 62 (0%)	23 of 68 (34%)	14 of 59 (24%)
“Lock”	45 of 45 (100%)	16 of 56 (29%)	12 of 46 (26%)

Table 1. Average number of times that participants entered a PIN/pattern per day to unlock their phone.

- Moderate increase of unlocks for users who currently do not lock their phone. (green)
- Considerable decrease of unlocks for users who currently lock their phone. (yellow)

Efficiency results (2)

Group	Phase I	Phase II	Phase III
“No Lock”	0 seconds	131 seconds	86 seconds
“Lock”	240 seconds	105 seconds	90 seconds

Table 2. Average time taken per day (sec) to enter PIN/Pattern.

- Considerable decrease in time spent unlocking the phone for users who currently lock their phone.

Perception – Annoyance (by phase)

The number of times in which I had to unlock my phone today was annoying.

Group	Phase I	Phase II	Phase III
"No Lock"	2	2.62	2.02
"Lock"	3.13	1.89	1.79

Table 3. Average ratings across each of the 3 phases.
(1. Strongly disagree, 5. Strongly agree).

- No lock group feels more annoyed in Phase II (yellow) but this annoyance level decreases in Phase III (green).
- Lock group feels less annoyed when using the proposed mechanism in Phase II & III (orange).

Perception – Annoyance (ranking)

Mechanism	Annoyance
Password	1.84
Pattern	2.31
PIN	2.47
Proposed mechanism	3.95
No lock	4.42

Table 4. Perception of annoyance (1=most annoying, 5=least annoying).

- Proposed mechanism ranked 2nd least annoying and significantly better than password, PIN and pattern.

Perception – Convenience (by phase)

Overall, the number of times in which I unlocked the phone today was convenient.

Group	Phase I	Phase II	Phase III
“No Lock”	3.77	3.26	3.81
“Lock”	3	4.02	3.82

Table 5. Average ratings across each of the 3 phases.

- (1. Strongly disagree, 5. Strongly agree).

- No lock group feels less convenient in Phase II (yellow) but the convenience level increases in Phase III (green).
- Lock group feels the proposed mechanism is more convenient both in Phase II & III (orange).

Perception – Convenience (ranking)

Mechanism	Convenience
No lock	1.55
Proposed mechanism	2.42
Pattern	3.58
PIN	3.84
Password	4.47

Table 6. Perception of convenience (1=most convenient, 5=least convenient).

- Proposed mechanism ranked 2nd most convenient and significantly better than PIN and password.

Perception – Security (by phase)

I felt secure with today's phone protection mechanism.

Group	Phase I	Phase II	Phase III
"No Lock"	3.22	3.6	4
"Lock"	3.63	3.68	3.86

Table 7. Average ratings across each of the 3 phases.
(1. Strongly disagree, 5. Strongly agree).

- Both groups feel secure when using the proposed mechanism in Phase II & III (yellow).

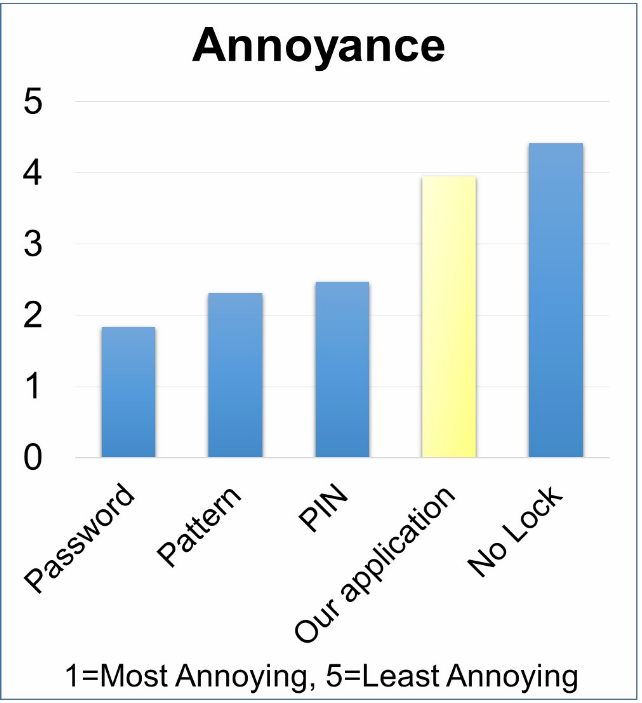
Perception – Security (ranking)

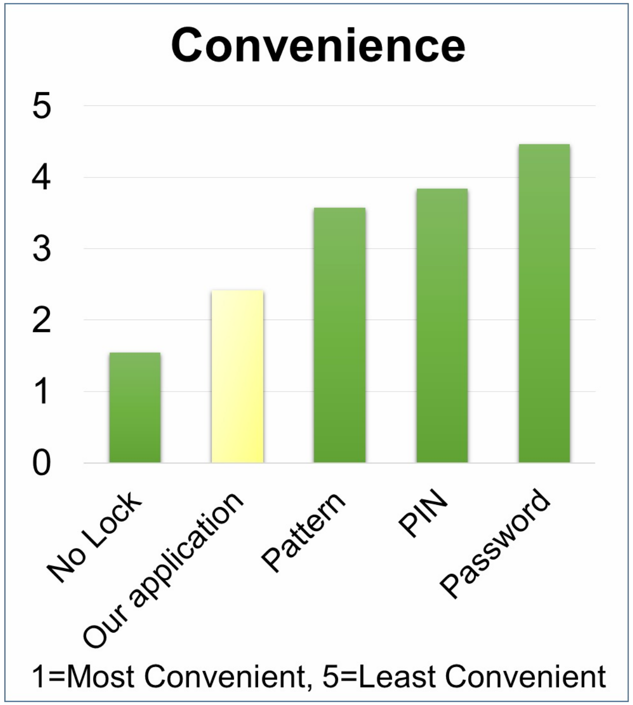
Mechanism	Security
Password	2
PIN	2.17
Proposed mechanism	2.79
Pattern	2.9
No lock	5

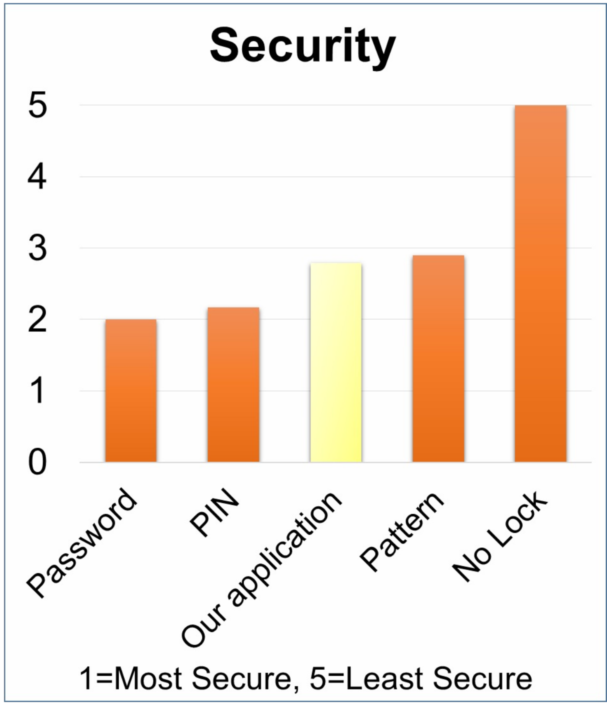
Table 8. Perception of security (1=most secure, 5=least secure).

- Proposed mechanism ranked 3rd most secure and significantly better than the No lock.

Perception Summary







Adoption results (1)

Location	“No Lock”	“Lock”
Home	1	9
Work	4	5
Other Places	7	3
On the move	5	4
New Places	8	2
Overall	8	9

Table 9. Final adoption results distributed by location.

- (8+9)/20 = 85% adoption rate

Adoption results (2)

- Adoption patterns tended towards increased usability, e.g., “at home”
  - Only 1 of 10 “no lock” users adopted our solutions (preferring to use no lock at home)
  - 9 of 10 “lock” users adopted our solutions (preferring reduced # of unlocks at home)

Some reading

- N. Micallef, M. Just, L. Baillie, M. Halvey, G. Kayacik, “Why aren’t users using protection? Investigating the usability of smartphone locking”, in *MobileHCI 2015*.
- N. Micallef, G. Kayacik, M. Just, L. Baillie, D. Aspinall, “Sensor use and usefulness: Trade-offs for data-driven authentication on mobile devices”, in *PerCom 2015*.
- G. Kayacik, M. Just, L. Baillie, D. Aspinall, N. Micallef, “Data Driven Authentication: On the Effectiveness of User Behaviour Modelling with Mobile Device Sensors”, in *MoST 2014*.

Behaviour drift

- Detection by periodically comparing scores
- Dramatic behaviour change (move to a new city) with incremental or complete retraining
- Similar, though complete adapts quicker