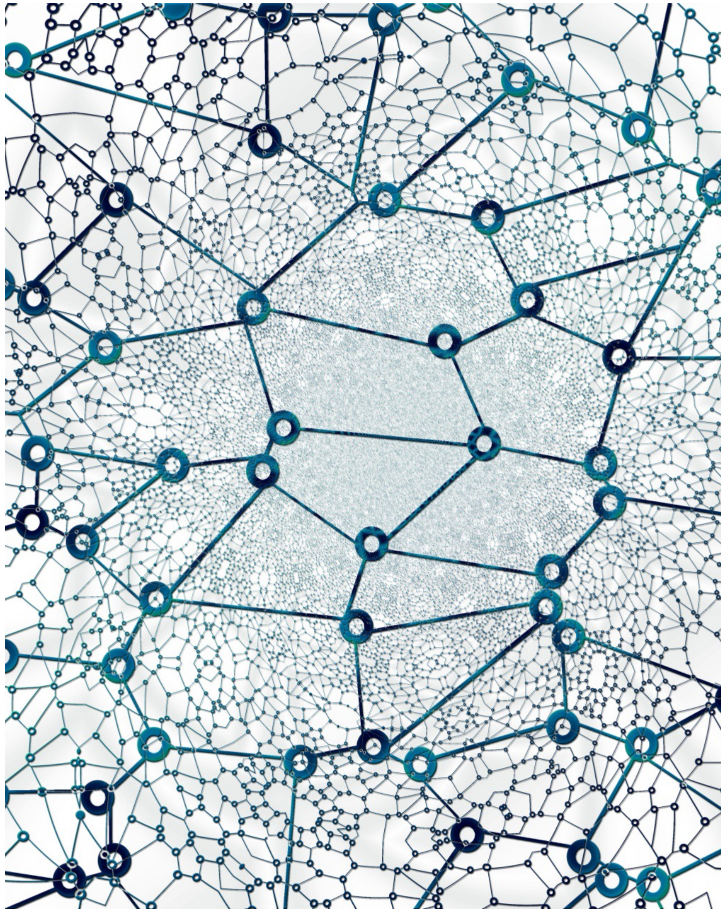
Bad Apples and Good Labels: Learning in Real-time Fault Detection

James A Grant
Lancaster University

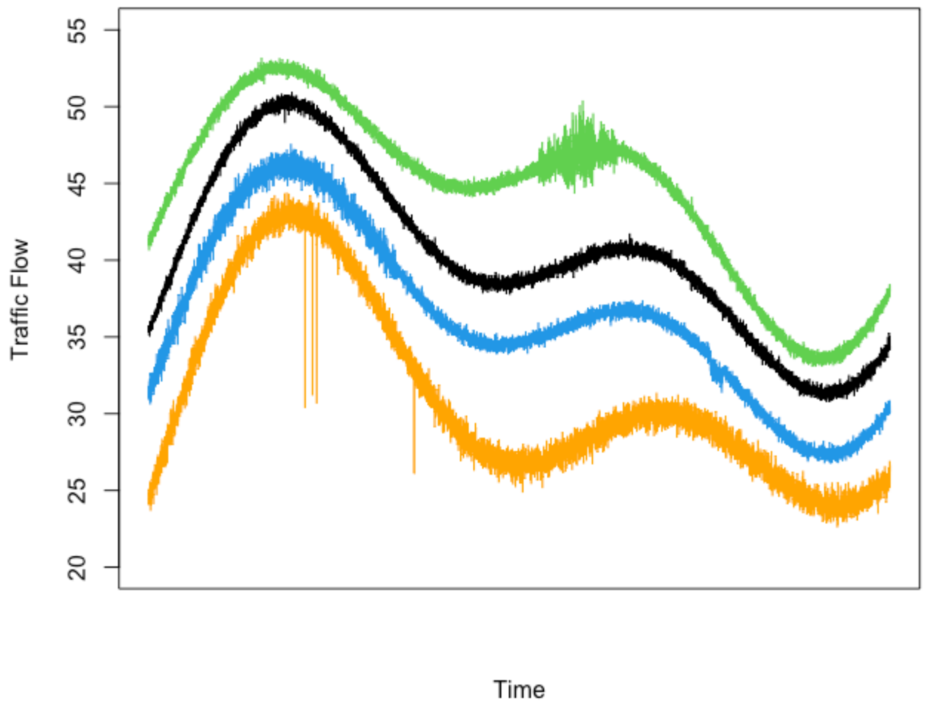
NG-CDI Tech Talks - 18th January 2022

j.grant@lancaster.ac.uk - @james_a_grant

Example: Telecoms Network Control



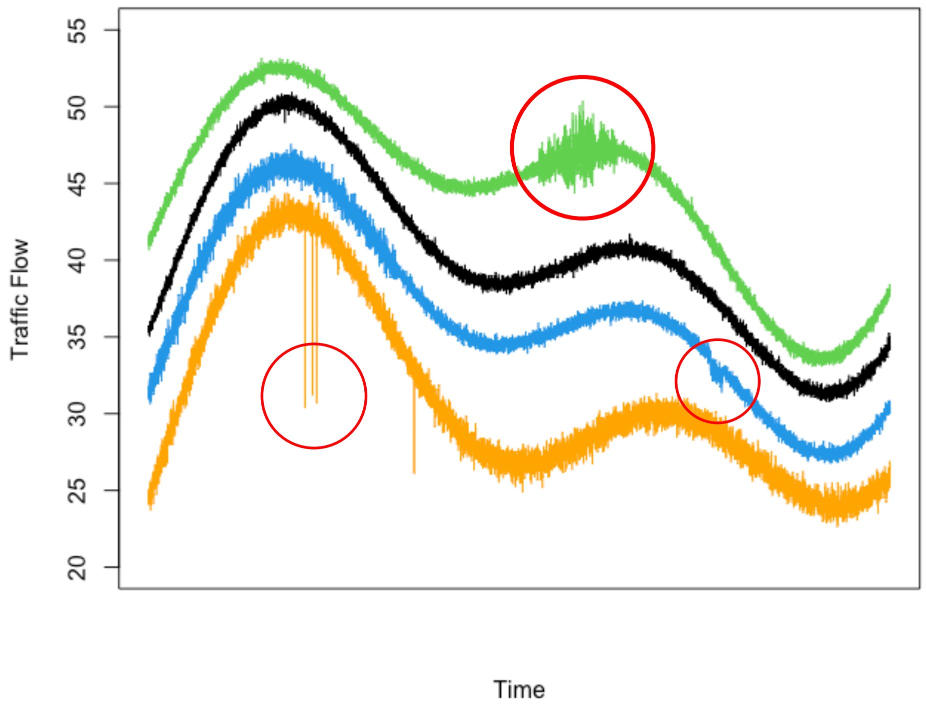
Example: Telecoms Network Control



Network traffic data takes the form of multivariate time series.

Engineers monitor these series for outages, faults, etc. and reroute traffic or schedule maintenance accordingly.

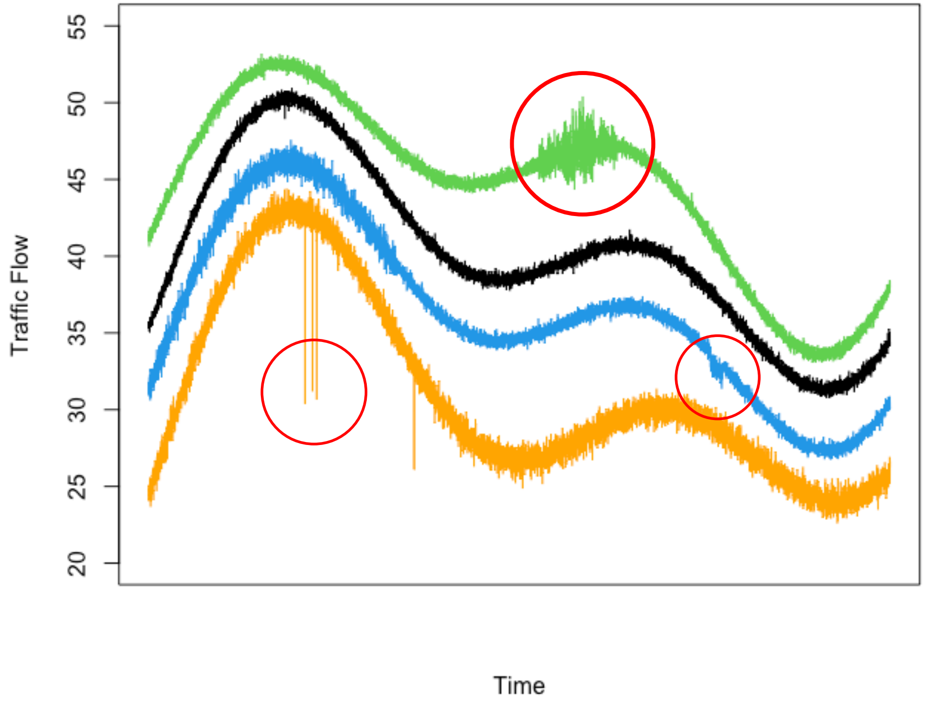
Example: Telecoms Network Control



Network traffic data takes the form of multivariate time series.

Engineers monitor these series for outages, faults, etc. and reroute traffic or schedule maintenance accordingly.

Example: Telecoms Network Control

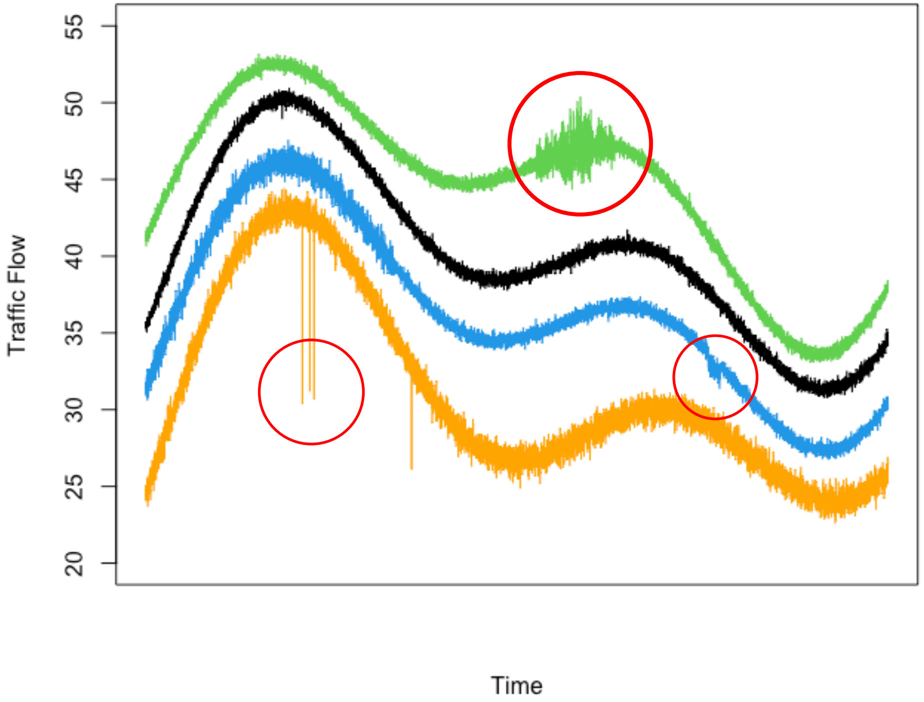


Network traffic data takes the form of multivariate time series.

Engineers monitor these series for outages, faults, etc. and reroute traffic or schedule maintenance accordingly.

Can machine learning replicate this?

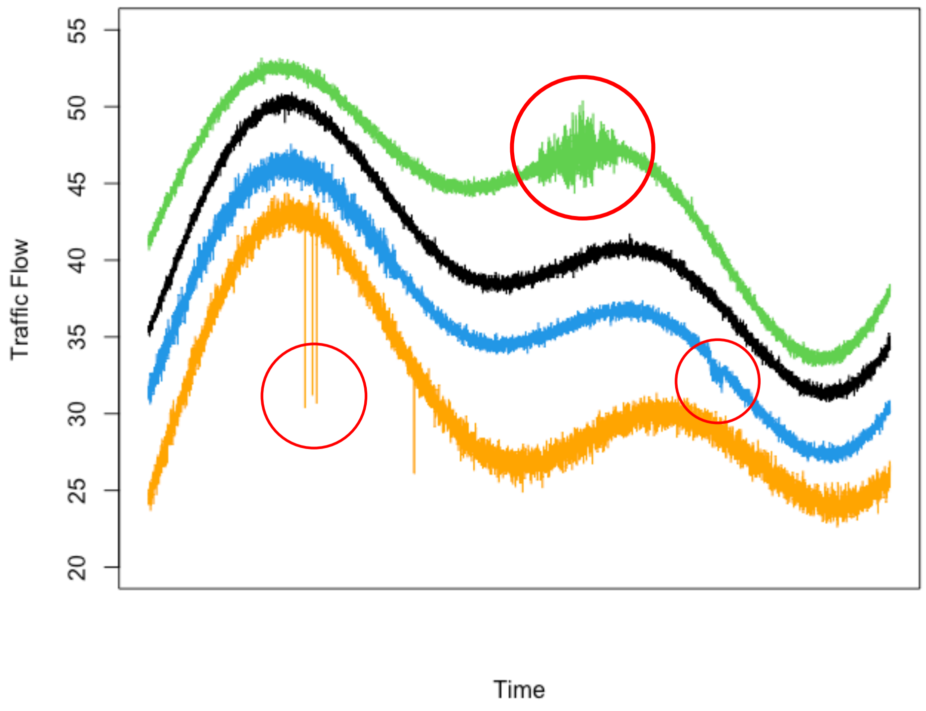
Example: Telecoms Network Control



Automating this process is **hard**

- Combining different knowledge
- Domain expertise
- Actions taken are complex
- Unseen examples and changing 'normal' behaviour

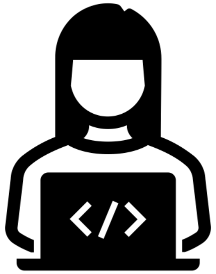
Example: Telecoms Network Control



A complete replacement with autonomous decision-making is unrealistic.

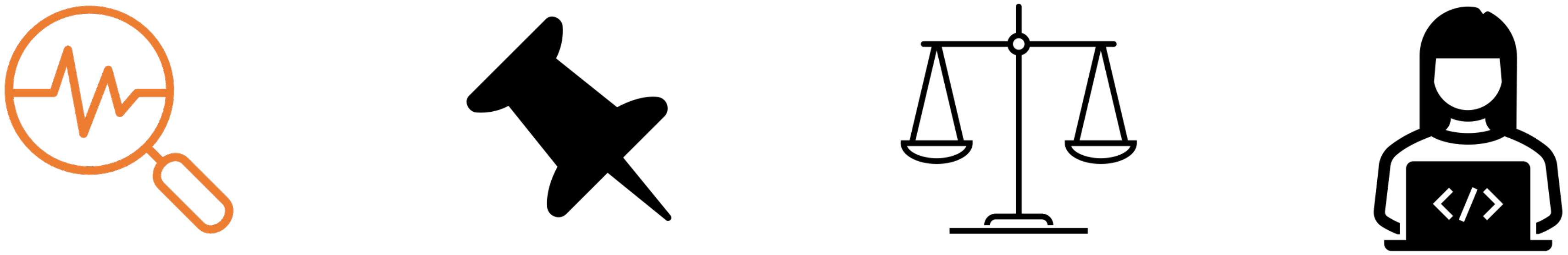
A semi-autonomous approach

We instead consider not trying to **make** decisions (per se), but flagging **when** a non-trivial decision needs to be made.

A semi-autonomous approach

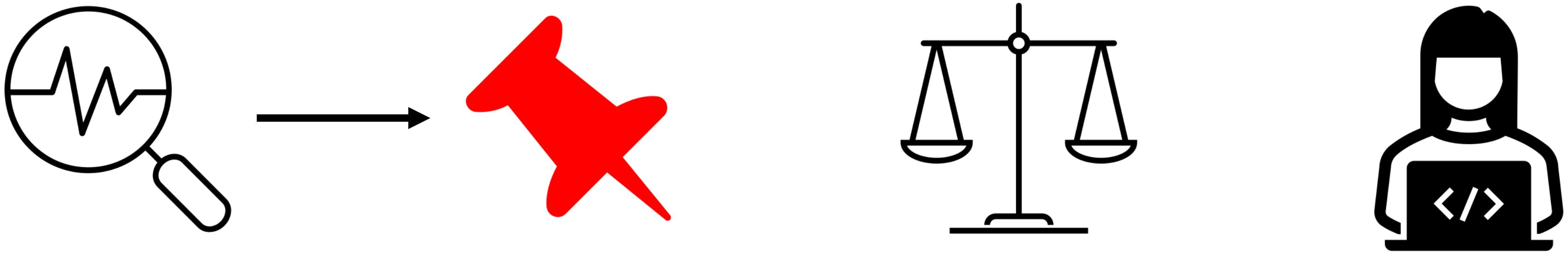
We instead consider not trying to **make** decisions (per se), but flagging **when** a non-trivial decision needs to be made.



Monitor the data

A semi-autonomous approach

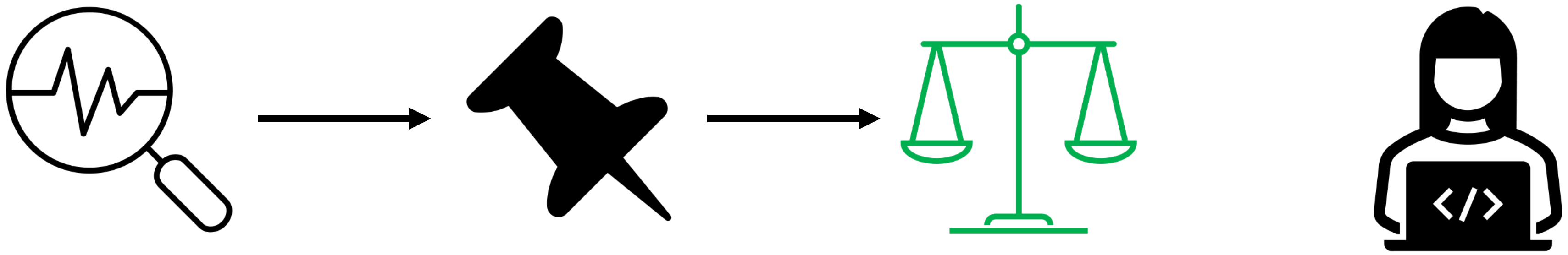
We instead consider not trying to **make** decisions (per se), but flagging **when** a non-trivial decision needs to be made.



Pin-point interesting regions

A semi-autonomous approach

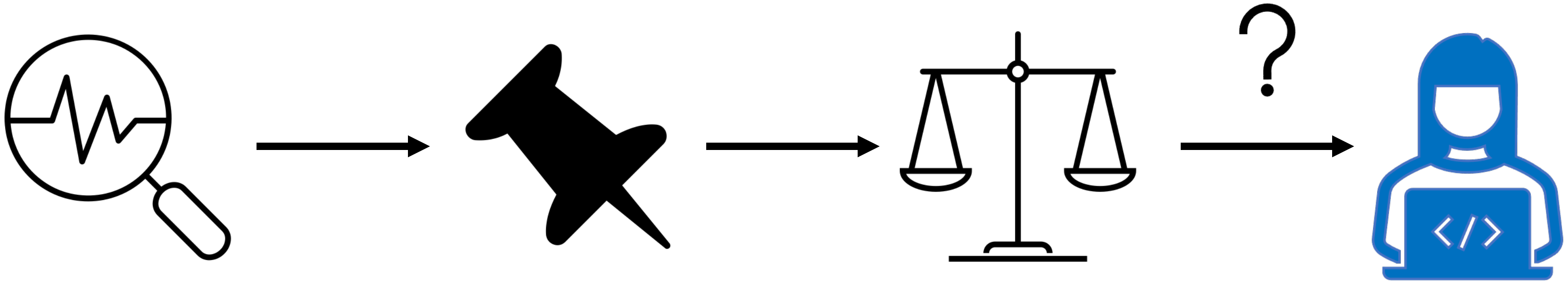
We instead consider not trying to **make** decisions (per se), but flagging **when** a non-trivial decision needs to be made.



Weigh up whether they are important

A semi-autonomous approach

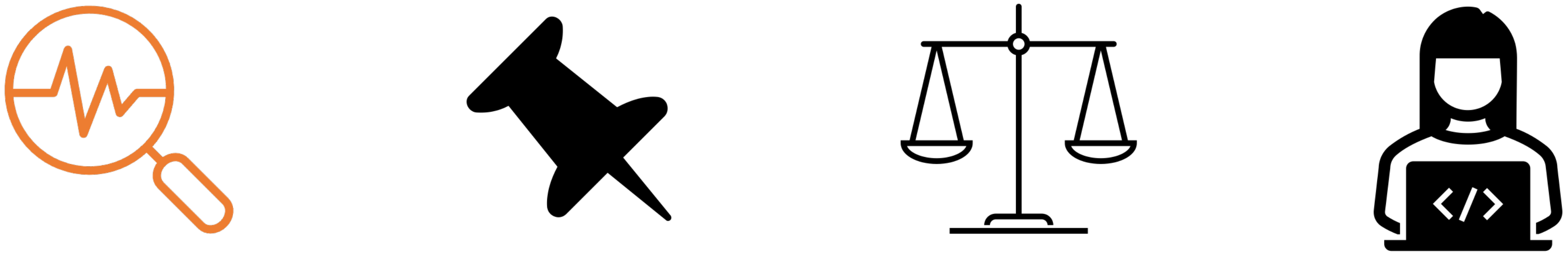
We instead consider not trying to **make** decisions (per se), but flagging **when** a non-trivial decision needs to be made.



Potentially pass to a human

A semi-autonomous approach

We instead consider not trying to **make** decisions (per se), but flagging **when** a non-trivial decision needs to be made.



Return to the monitoring phase

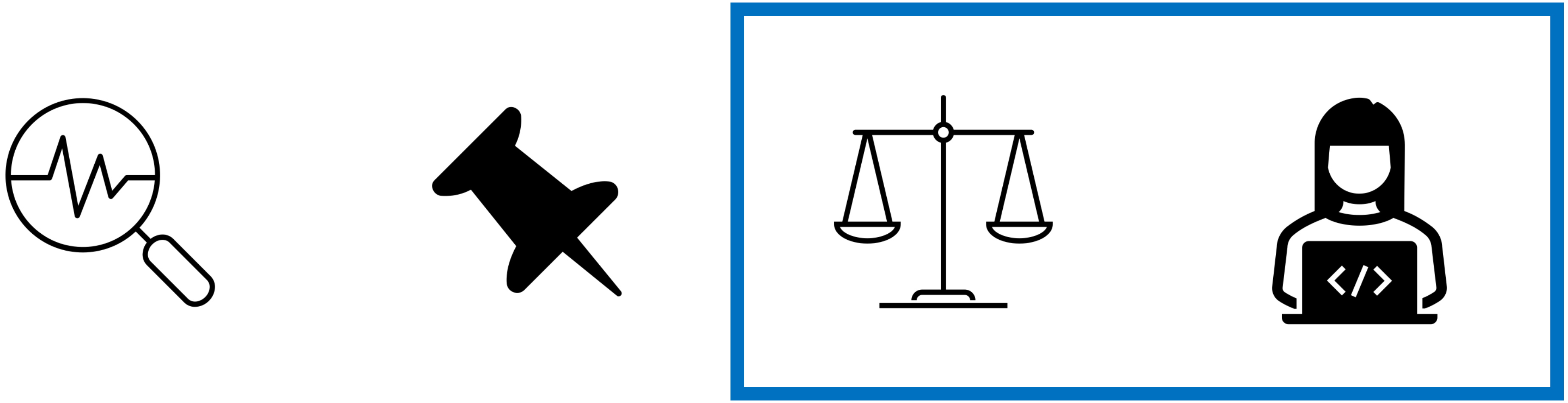
A semi-autonomous approach

We instead consider not trying to **make** decisions (per se), but flagging **when** a non-trivial decision needs to be made.

Part 1 – Anomaly Detection

A semi-autonomous approach

We instead consider not trying to **make** decisions (per se), but flagging **when** a non-trivial decision needs to be made.



Part 2 – Classification

Learning to Classify

We pose the decision to flag or not as a binary classification task.

Each potentially interesting anomaly ($t = 1,2, \ldots$) has

- Associated feature vector $x_t \in \mathbb{R}^d$ - size of deviation/extraneous variables/baseline deviated from/etc.
- True (latent) class $C_t \in \{0,1\}$ – not interesting/interesting

To some extent x_t's can predict C_t's – e.g. logistic regression-like relationship mediated by parameter $\theta \in \mathbb{R}^d$.

Learning to Classify

Binary classification/logistic regression is really well studied.

Learning to Classify

Binary classification/logistic regression is really well studied.

Offline Binary Classification: Have a history of $x_1, \ldots x_n$ and $C_1, \ldots, C_n$ and produce estimate $\hat{\theta}_n$. Predict any future $\hat{C}_t$ based on x_t and $\hat{\theta}_n$.

Learning to Classify

Binary classification/logistic regression is really well studied.

Offline Binary Classification: Have a history of $x_1, \dots x_n$ and $C_1, \dots, C_n$ and produce estimate $\hat{\theta}_n$. Predict any future $\hat{C}_t$ based on x_t and $\hat{\theta}_n$.

Online Binary Classification: Little or no historic data. Iteratively observe x_t, predict $\hat{C}_t$, observe **true** C_t, and update estimate $\hat{\theta}_t$.

Learning to Classify

Binary classification/logistic regression is really well studied.

Offline Binary Classification: Have a history of $x_1, \dots x_n$ and $C_1, \dots, C_n$ and produce estimate $\hat{\theta}_n$. Predict any future $\hat{C}_t$ based on x_t and $\hat{\theta}_n$.

Online Binary Classification: Little or no historic data. Iteratively observe x_t, predict $\hat{C}_t$, observe **true** C_t, and update estimate $\hat{\theta}_t$.

Online Binary Classification with Partial Feedback: Same setting as online – but only observe true C_t if $\hat{C}_t = 1$.

Online Binary Classification with Partial Feedback, or ‘**Apple Tasting**’.

Apple Tasting

- Learning to identify good and bad apples *(Helmbold et al. 1992, 2000)*.
- **Aim**: let all good apples through, remove all bad apples.
- Class only revealed by taste – which destroys the apple:
  - Desirable for bad apples. Wasteful for good apples.

Apple Tasting

- Learning to identify good and bad apples *(Helmbold et al. 1992, 2000)*.
- **Aim**: let all good apples through, remove all bad apples.
- Class only revealed by taste – which destroys the apple:
  - Desirable for bad apples. Wasteful for good apples.

- Challenge is that to maximise accuracy, some good apples must be removed for sake of learning – **but which ones and how many**?

Balancing Exploration and Exploitation

- Repeatedly face the following question:
  - Given observed features x_t, and a guess of the class $\mathrm{P}(C_t = 1)$ (based on a $\hat{\theta}_t$) do we choose treat as a good or bad apple?
  - **NB:** doesn't have to be treat as bad if $P(C_t = 1) > 0.5$ – can have more conservative view of trade-off.
  - For ease in what follows: assume parity between false positive and false negative.

Balancing Exploration and Exploitation

- Repeatedly face the following question:
    - Given observed features x_t, and a guess of the class $P(C_t = 1)$ (based on a $\hat{\theta}_t$) do we choose treat as a good or bad apple?

- Why not just use best guess all the time?
    - Could work brilliantly - if x_i sequence is sufficiently variable, if you start with good data
    - Could also fail catastrophically – initialise $\hat{\theta}$ poorly and only observe data which confirms bias.

Balancing Exploration and Exploitation

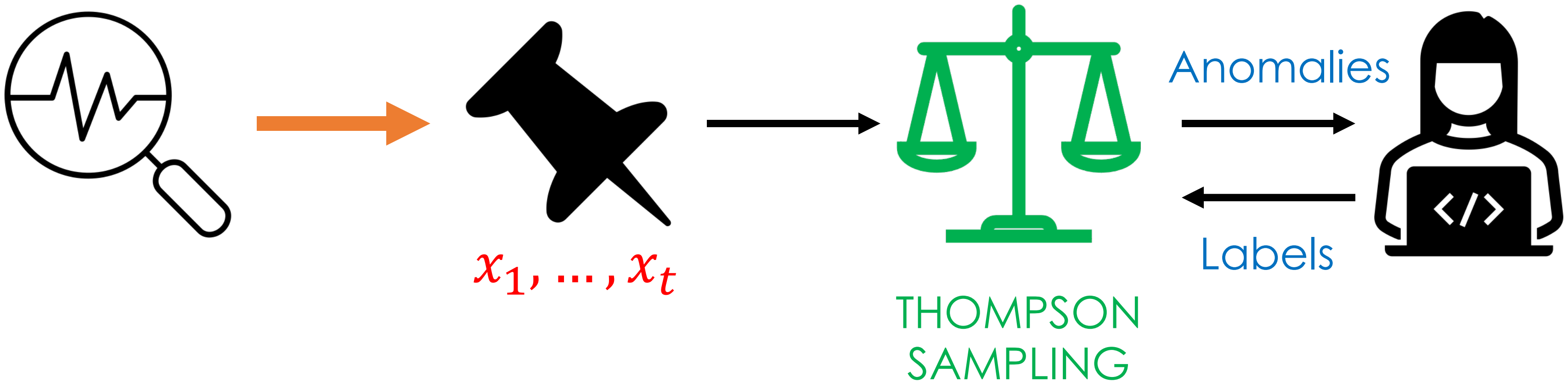
- Superior methods ensure we have enough data to maintain a good estimate of $\hat{\theta}_t$.
- Two main techniques:
  - **Confidence bounds** - only treat as a good apple if we're very certain it's good (effectively shift $\hat{\theta}_t$ to the limit of some region Θ_t such that $P(\theta \in \Theta_t) > 1 - \delta$)
  - **Randomisation** – add (appropriate) noise to $\hat{\theta}_t$, so that sometimes an estimated label $\hat{C}_t$ will be flipped (encouraging exploration)
- Both converge to using $\hat{C}_t$ once $\hat{\theta}_t$ is well estimated.

Randomised Decision Making via Thompson Sampling

- Initialise with a prior distribution $\pi_0(\theta)$
- At time $t = 1, 2, \ldots$
  - Draw a sample $\tilde{\theta}_t$ from the current posterior $\pi_{t-1}(\theta)$
  - Treat $\tilde{\theta}_t$ as the true parameter and estimate $\hat{C}(\tilde{\theta}_t)$ based on x_t.
  - If $\hat{C}(\tilde{\theta}_t) = 1$
    - Remove the apple/show anomaly to human
    - Observe C_t and update the belief distribution to $\pi_t(\theta)$.
  - If $\hat{C}(\tilde{\theta}_t) = 0$
    - Let apple/anomaly pass
    - Observe nothing and set $\pi_t(\theta) = \pi_{t-1}(\theta)$.

Summing up

We've put anomaly detection and online classification (Apple Tasting via Thompson Sampling) together to produce a semi-autonomous algorithm.



Summing up

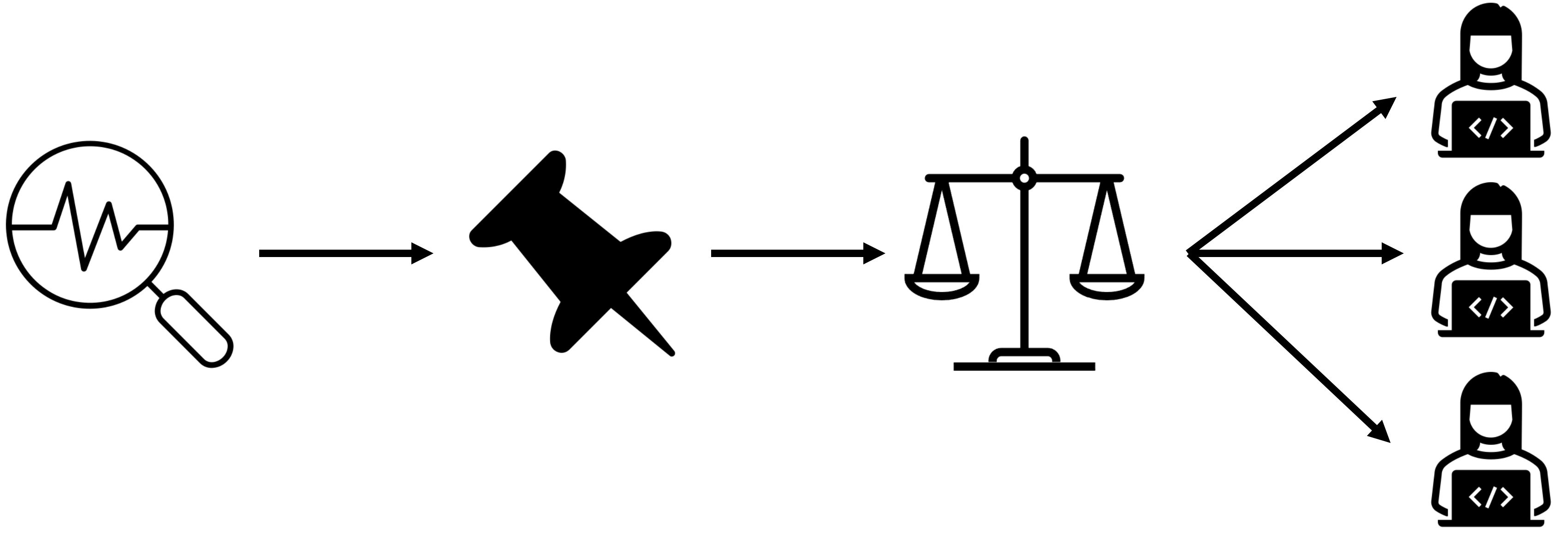
We’ve put anomaly detection and online classification (Apple Tasting via Thompson Sampling) together to produce a semi-autonomous algorithm.

The approach allows us to **automate where possible**, without large amounts of initial labelled data, and continues to **learn as it proceeds**.

The principle is simple but widely applicable/extendable.

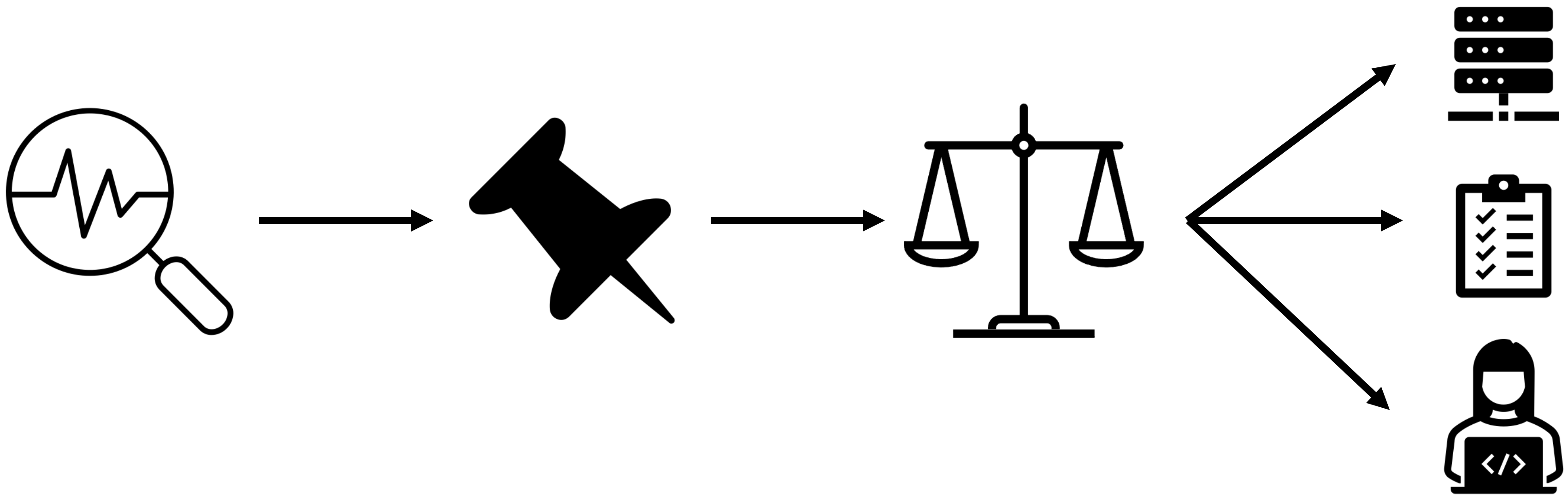
Summing up

Going forward in this space we want to explore more complex decision-making setups:



Summing up

Going forward in this space we want to explore more complex decision-making setups:



References & Contact

- Grant, J.A., Leslie D.S. (2021). *Apple Tasting Revisited: Partially Monitored Online Binary Classification*. In Submission, arXiv:2109.14412.

- Helmbold, D.P., Littlestone, M., and Long, P.M. (2000). *Apple Tasting*. Information and Computation.

j.grant@lancaster.ac.uk @james_a_grant