Know Unknowns from Knowns
- Novelty Detection and its Applications

Yuhua Li

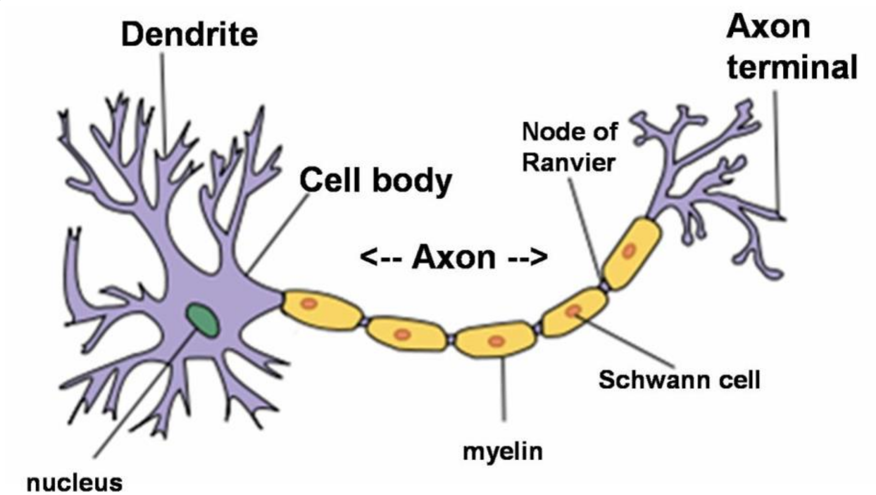
School of Computing, Science and Engineering
University of Salford

29 November 2017
Salford Computer Society

Machine Learning

- It is the ability of machines to learn and solve problems just like humans
- Algorithms that learn models to discover patterns and relationships or uncover "hidden insights" from available data for prediction, description and diagnosis
- Types of machine learning
  - Supervised learning
  - Unsupervised learning
  - Reinforcement learning

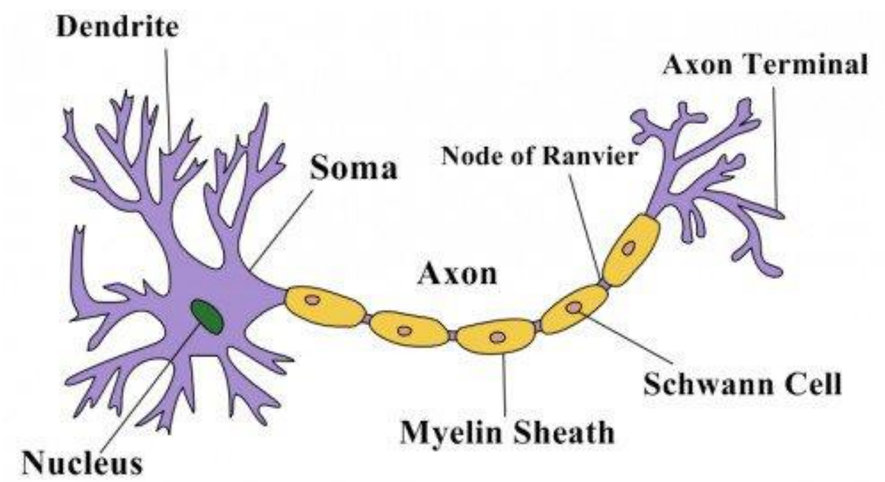
Brain – Neurons



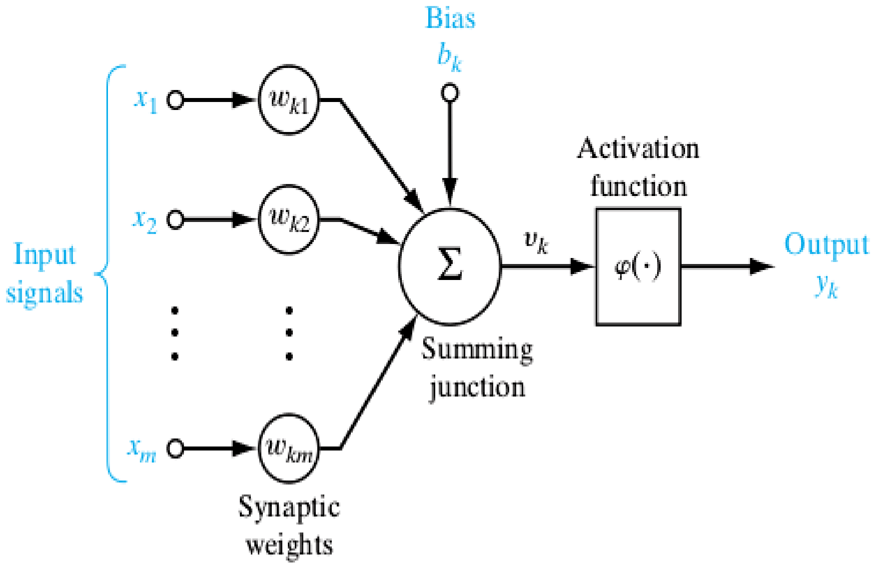
- We are born with about 100 billion neurons
- A neuron may connect to as many as 100,000 other neurons
- Signals "move" via electrochemical signals
- The synapses release a chemical transmitter – the sum of which can cause a threshold to be reached – causing the neuron to "fire"
- Synapses can be inhibitory or excitatory

Neuron Model

Biological

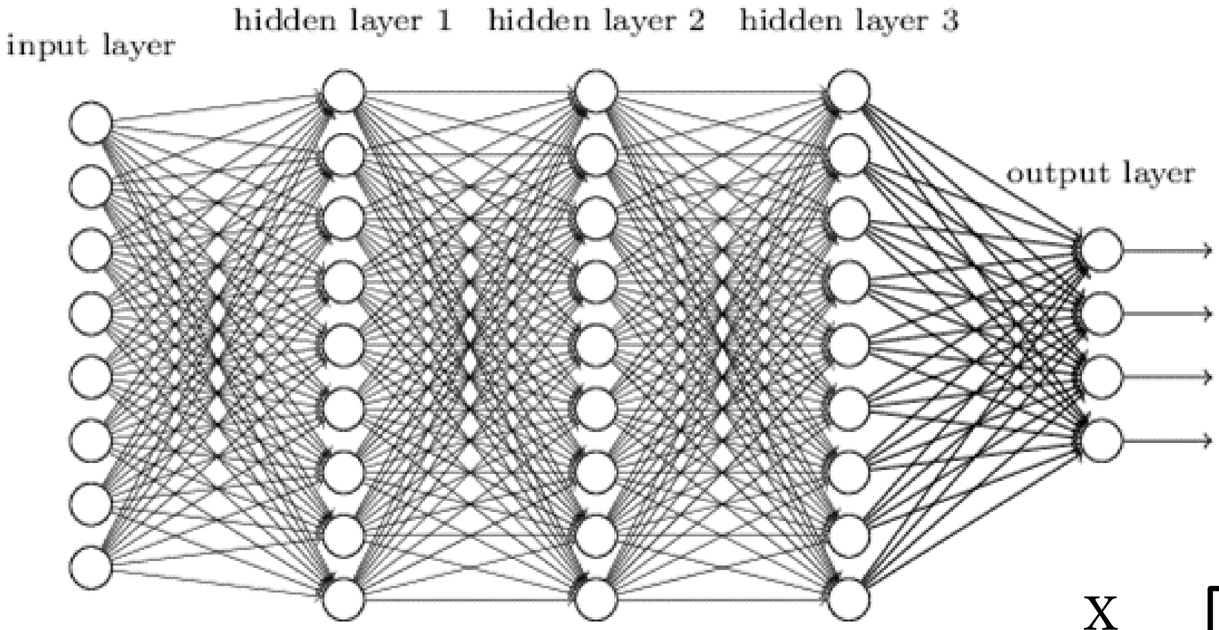


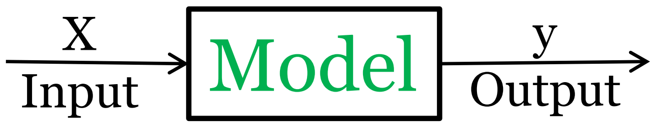
Artificial



Neural Network Layers

- Each layer receives its inputs from the previous layer and forwards its outputs to the next layer





$$y = f(\mathrm{X})$$

AI / Machine Learning Examples

- Deep Blue won its first game against a world champion on 10 Feb1996

- Google AlphaGo beat top human Go master, Lee Sedol (2016).
  - AlphaGo Fan (Oct 2015)
  - AlphaGo Lee (Mar 2016)
  - AlphaGo Zero (Oct 2017)

  - Search space
    - Chess: 35^{80}
    - Go: 250^{150}

- Face authentication

- Driverless cars

There are known knowns. There are things we know that we know. There are known unknowns. That is to say, there are things that we now know we don't know. But there are also unknown unknowns. There are things we do not know we don't know.

There are known knowns. There are things we know that we know. There are **known unknowns**. That is to say, there are things that we now know we don't know. But there are also **unknown unknowns**. There are things we do not know we don't know.

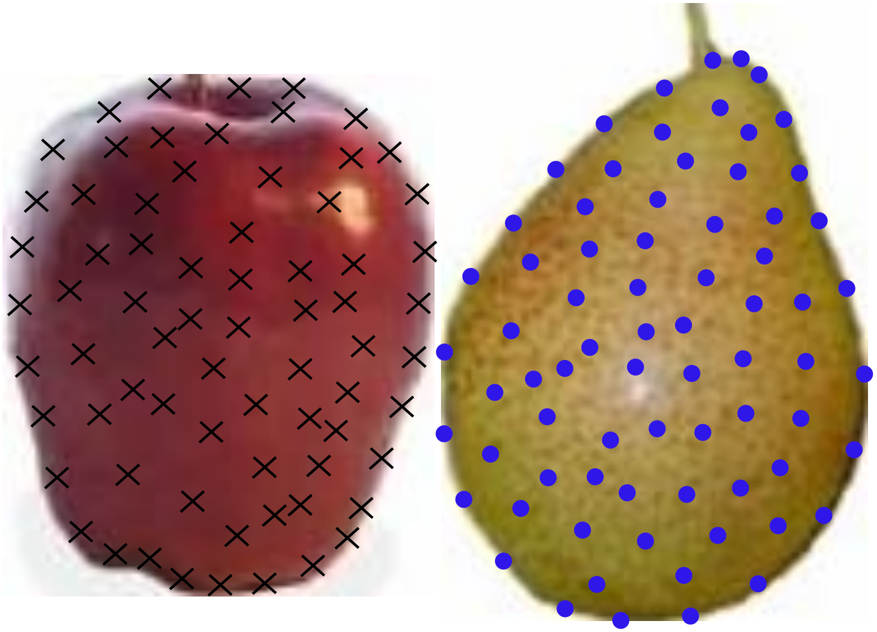
Donald Rumsfeld

US Secretary of State for Defence

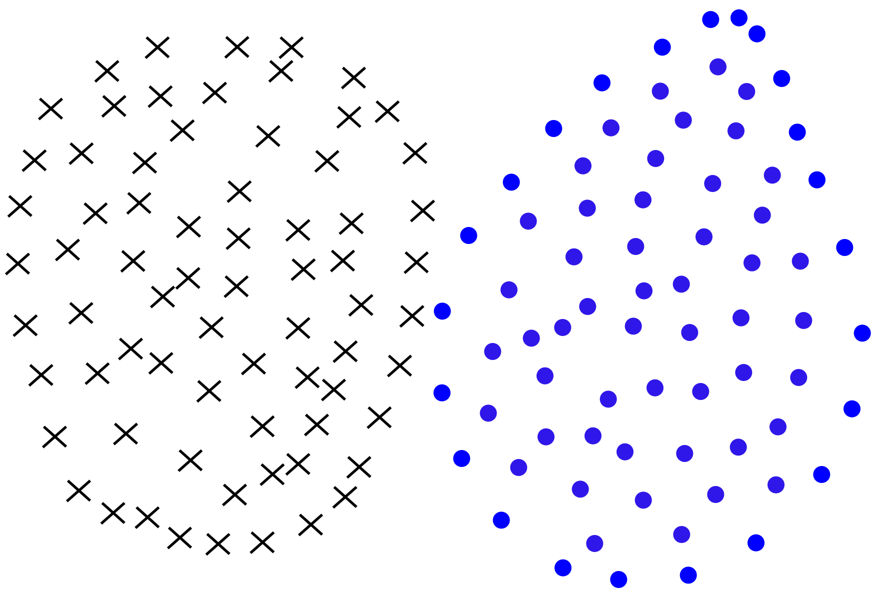
February 2002

Novelty Detection Introduction

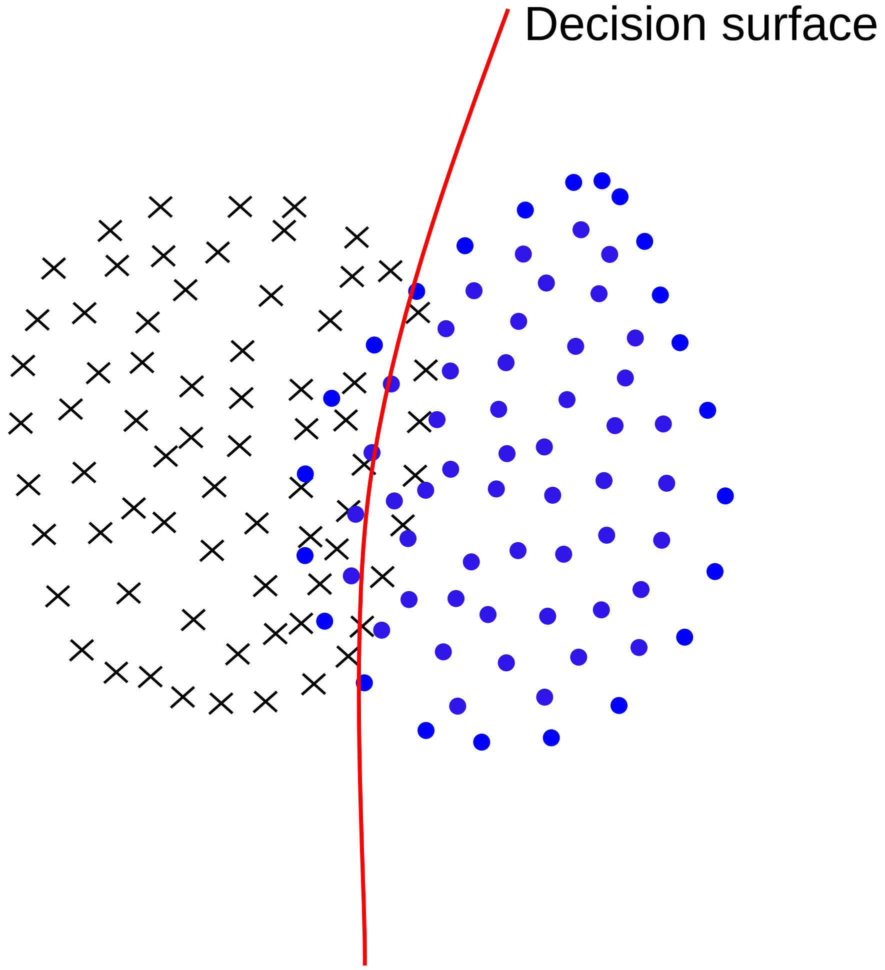
- Object Representation Using Samples/data



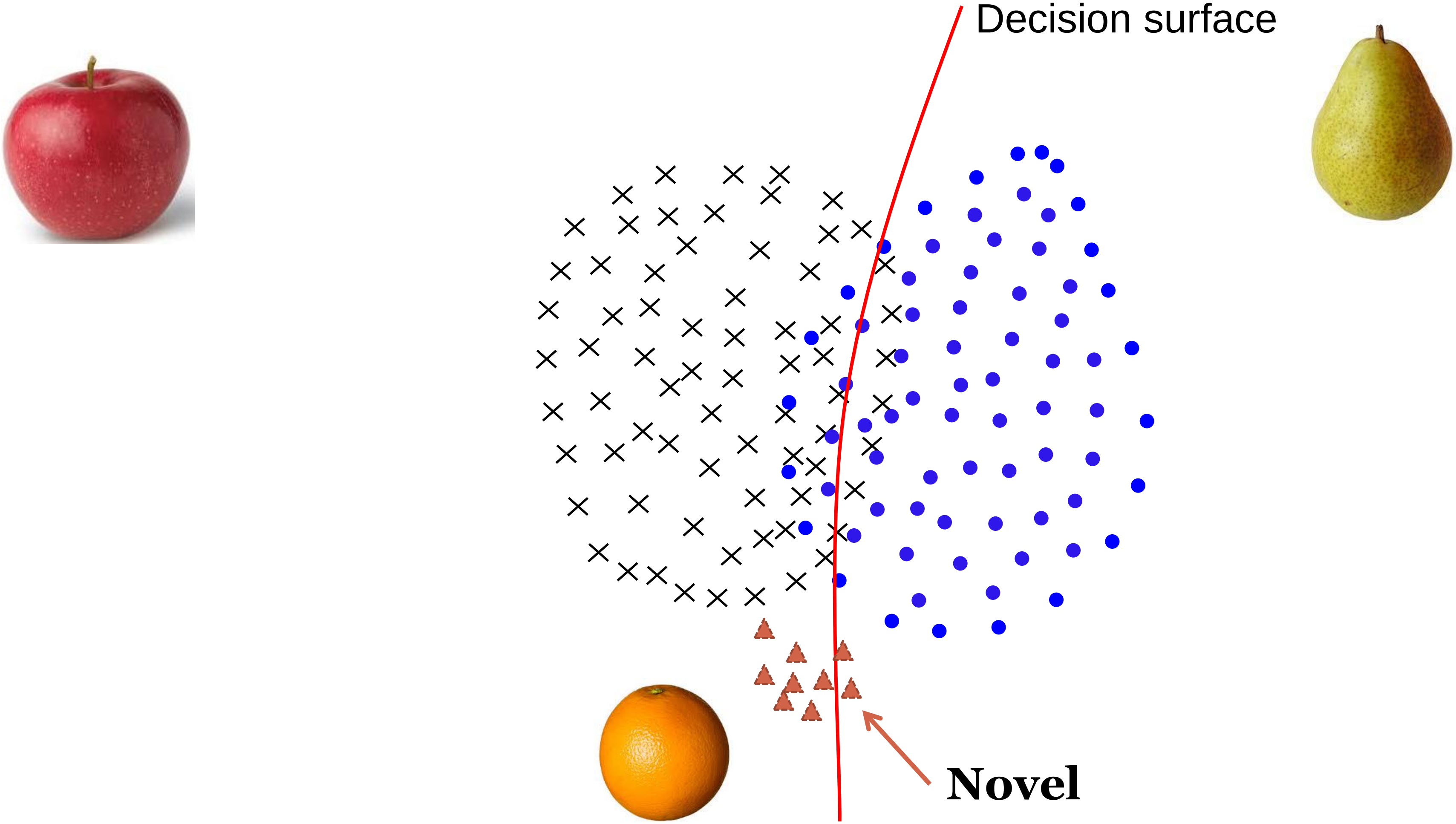
Novelty Detection Introduction



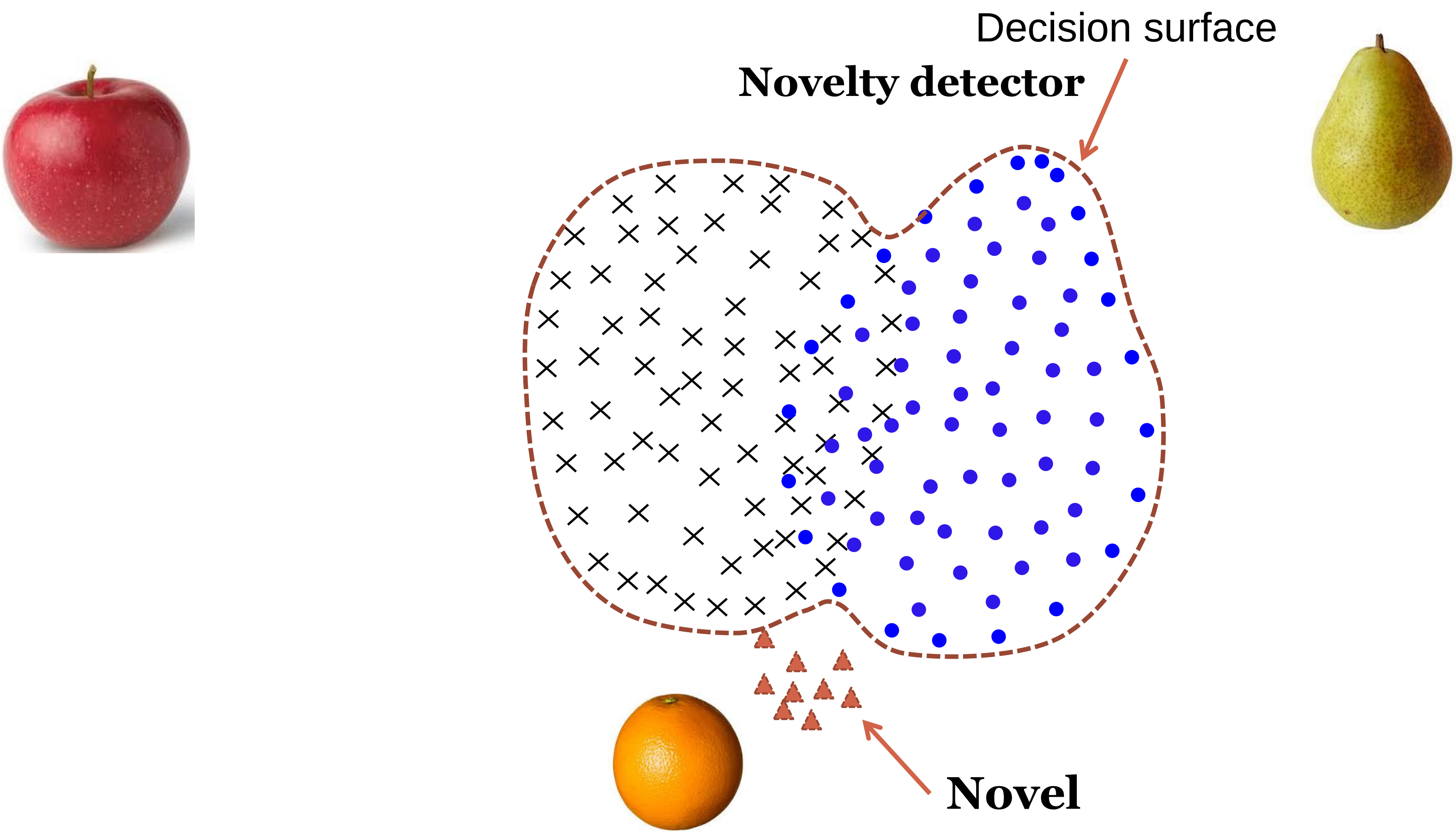
Novelty Detection Introduction



Novelty Detection Introduction



Novelty Detection Introduction



Novelty Detection Introduction

- Novelty is a pattern in the data that does not conform to the expected behaviour
- Novel events occur relatively infrequently or never occurred before
- However, when they do occur, their consequences can be quite dramatic and quite often in a negative sense
- Novelty detection
  - learns a model purely based on data collected from normal or **known** events/condition
  - detects **unknown** events that may occur in the future
- It is a powerful technique in the era of big data where
  - we have plenty of data about a system under normal operations,
  - but very limited or nil data about abnormal events.

Application Domains

- Electronic IT security
  - Malware/ransomware detection
- Healthcare informatics/medical diagnostics and monitoring
  - Early warning of patient deterioration
- Industrial monitoring and damage detection
  - Nuclear power station monitoring
- Image processing/video surveillance
  - Novel objects recognition in images/video streams
  - Novel events in security or surveillance
- Text mining
  - New topic detection
- Sensor networks
  - Sensor faults, malicious attacks
- Financial engineering
  - Insurance / Credit card fraud detection
  - Capital market surveillance

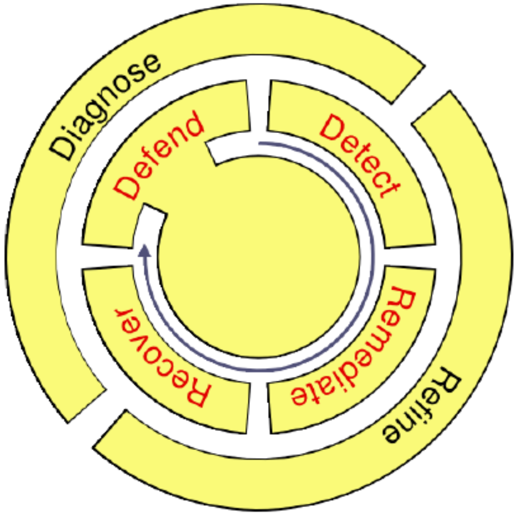
Intrusion Detection

- Intrusion Detection:
  - Intrusions are defined as attempts to bypass the security mechanisms of a computer or network
  - Process of monitoring the events occurring in a computer system or network and analyzing them for intrusions
- Challenges
  - Traditional signature-based intrusion detection systems are based on signatures of known attacks and cannot detect emerging cyber threats
  - Substantial latency in deployment of newly created signatures across the computer system
- Anomaly detection can alleviate these limitations

Malware Detection

- Data collection from packet capturing and network flow summarisation to extract features
  - memory usage (i.e. actual size of the process in memory)
  - peak memory usage (i.e. the requested memory allocation)
  - number of threads
  - number of handles (resources the process has open, e.g. files)

  - packets per address pair
  - bytes per address pair
  - flows per address pair



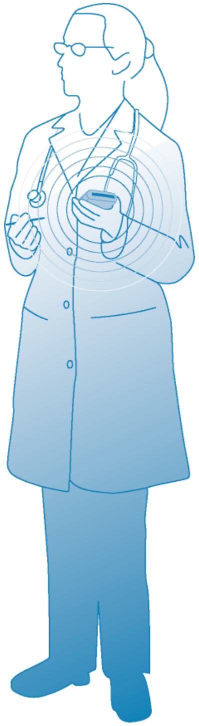
Fraud Detection

- Fraud detection refers to detection of criminal activities occurring in commercial organizations
  - Malicious users might be the actual customers of the organization or might be posing as a customer (also known as identity theft).
- Types of fraud
  - Credit card fraud
  - Insurance claim fraud
  - Mobile / cell phone fraud
  - Insider trading

- Challenges
  - Fast and accurate real-time detection
  - Misclassification cost is very high

Healthcare Informatics

- Detect anomalous patient records
  - Indicate disease outbreaks, instrumentation errors, etc.
- Key Challenges
  - Only normal labels available
  - Misclassification cost is very high
  - Data can be complex: spatio-temporal



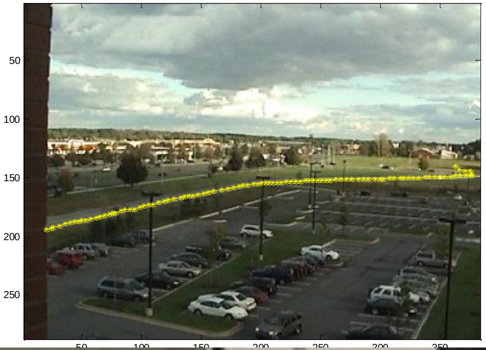
Industrial Damage Detection

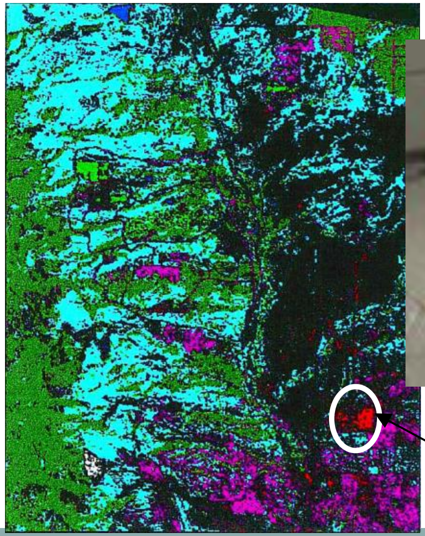
- Industrial damage detection refers to detection of different faults and failures in complex industrial systems, structural damages, intrusions in electronic security systems, suspicious events in video surveillance, abnormal energy consumption, etc.
  - Example: Aircraft Safety
    - Anomalous Aircraft (Engine) / Fleet Usage
    - Anomalies in engine combustion data
    - Total aircraft health and usage management

- Key Challenges
  - Data is extremely huge, noisy and unlabelled
  - Most of applications exhibit temporal behavior
  - Detecting anomalous events typically require immediate intervention

Image Processing

- Detecting outliers in a image monitored over time
- Detecting anomalous regions within an image
- Used in
  - mammography image analysis
  - video surveillance
  - satellite image analysis
- Key Challenges
  - Detecting collective anomalies
  - Data sets are very large





Project:
Capital Market Manipulation Detection

- The Financial Crisis of 2008
  - It almost brought down the world's financial system. It took huge taxpayer-financed bail-outs to shore up the industry.
- 2010 Flash Crash
  - At 2:42 pm of 6 May 2010, Dow Jones index lost nearly 1,000 points (~9%) in 5 minutes
  - tens of billions of dollars in losses in just five minutes. (manipulator Navinder Singh Sarao earn ~$40m)

Price Manipulation

Best Ask:
Lowest price seller can afford.

Best Bid:
Highest price buyer can pay.

Market Price is between and decided by **Bid** and **Ask**;

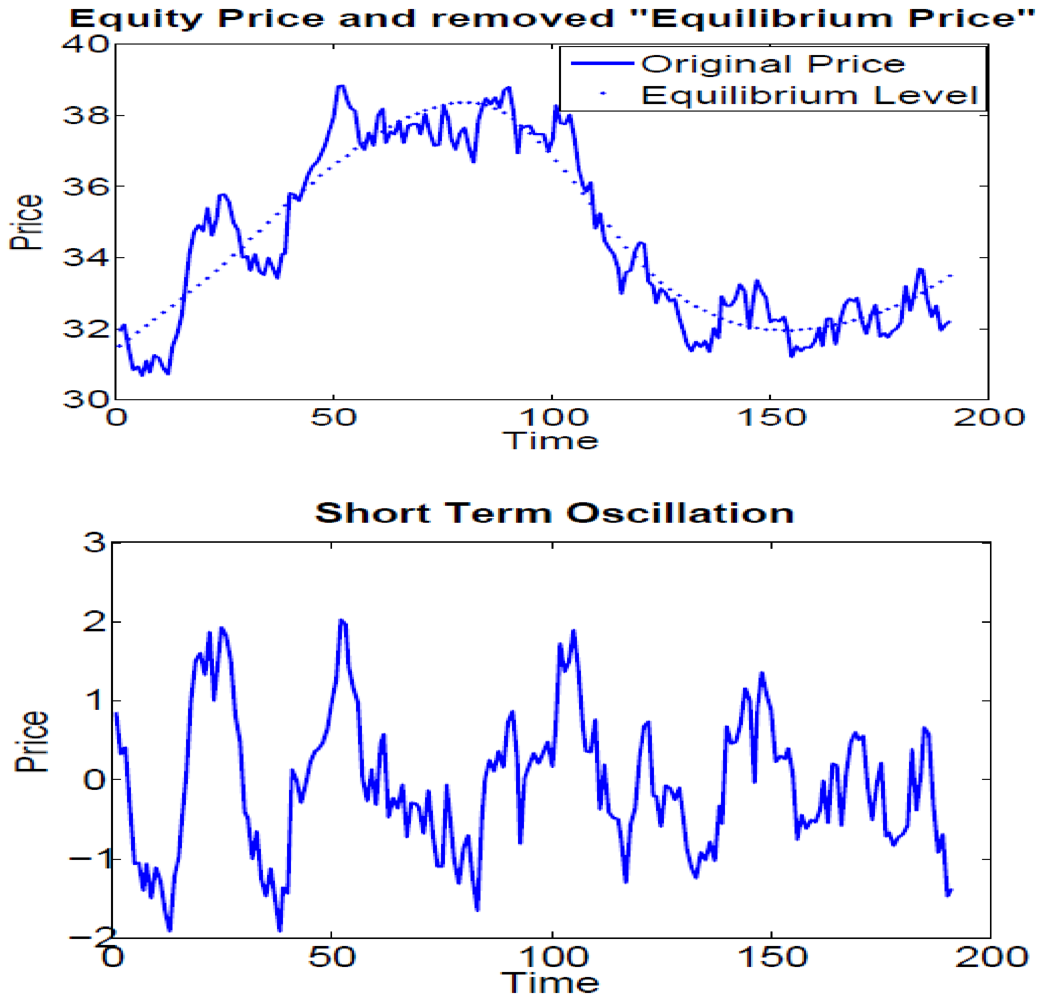
Manipulation on either **Bid** or **Ask** can change the **Market Price**

Price Manipulation Detection

- Feature Extraction

- Wavelet is applied as the feature extraction method. Equilibrium level in the price is removed;

- The short oscillation is remained. The manipulation patterns are hidden somewhere in the small oscillations.

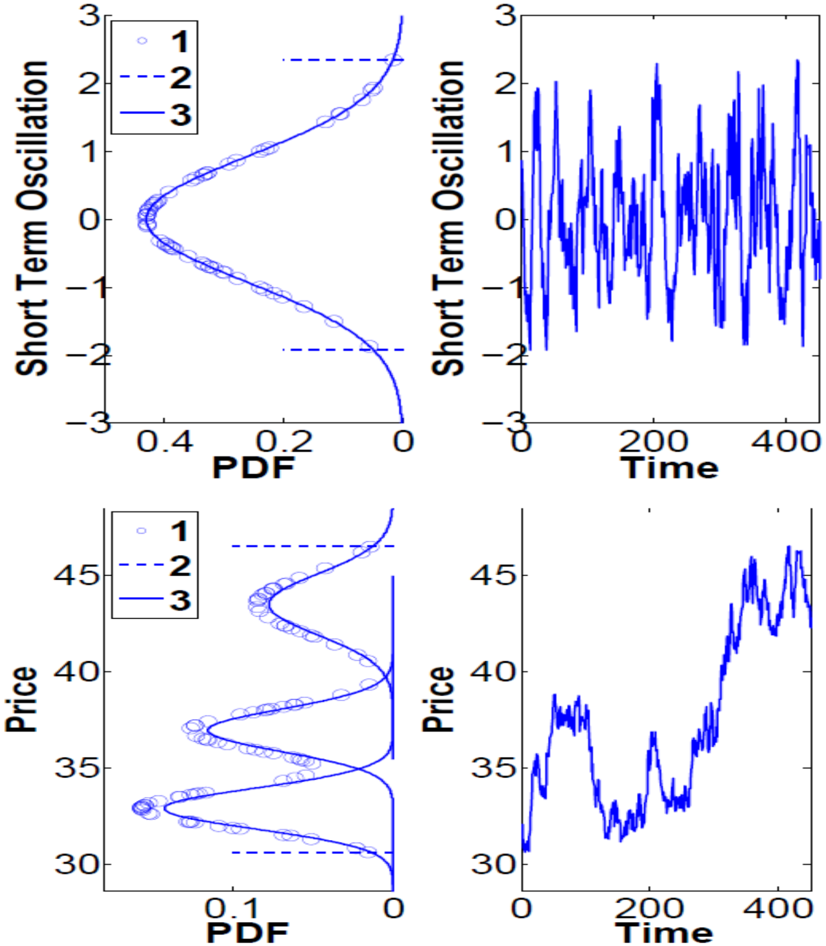


Price Manipulation Detection

- Model the features

- GMM learning the PDF of the oscillation.

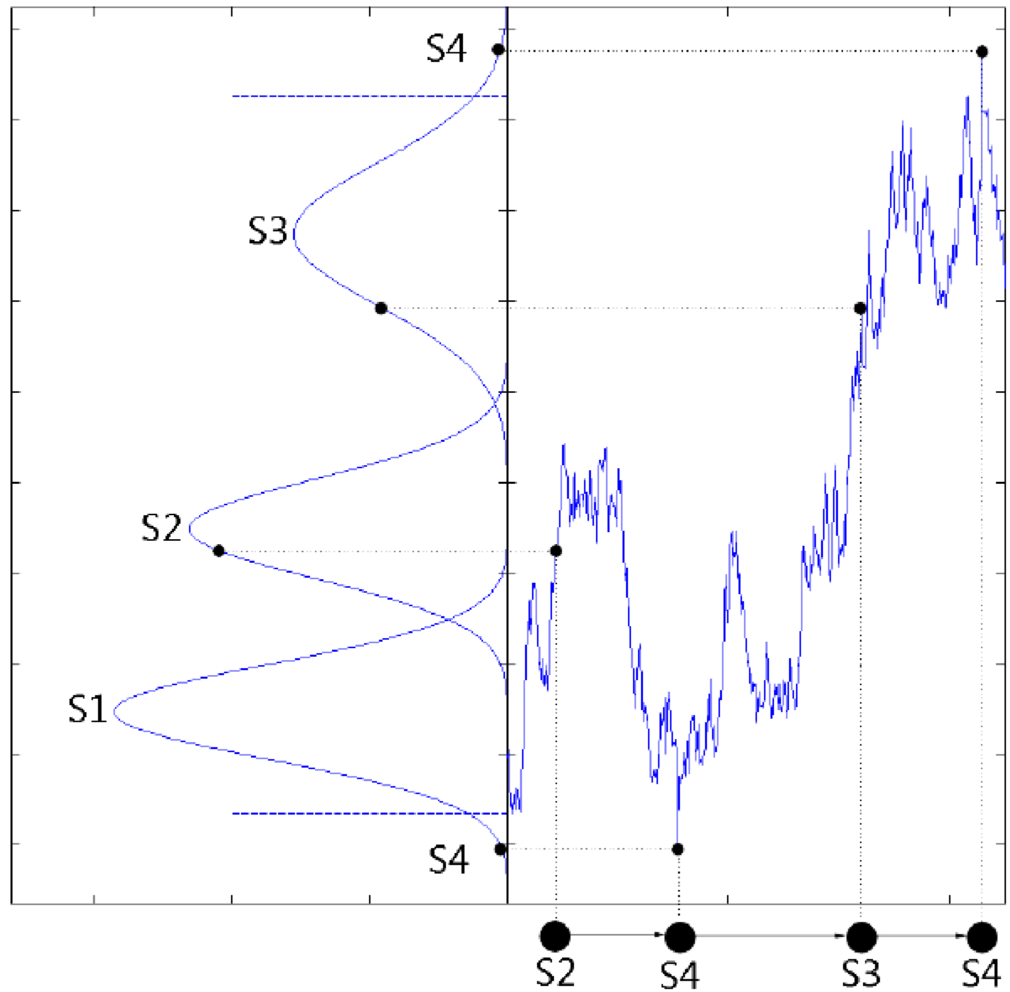
- GMM learning the PDF of the original price as well.



Price Manipulation Detection (4)

❑ Model the features

- HMM modeling the learned PDF.
- Thresholds are set to include 99% of the normal examples.
- Examples outside the thresholds are learned as the “anomaly states” while the examples inside the thresholds are learned as “normal states”.
- HMM with Anomaly States (HMMAS) is then modeled.



Acknowledgement

- Sponsors and Partners

Fidessa Trade/Invest/Inform

citi

First Derivatives plc

NYSE Technologies

KOFAX

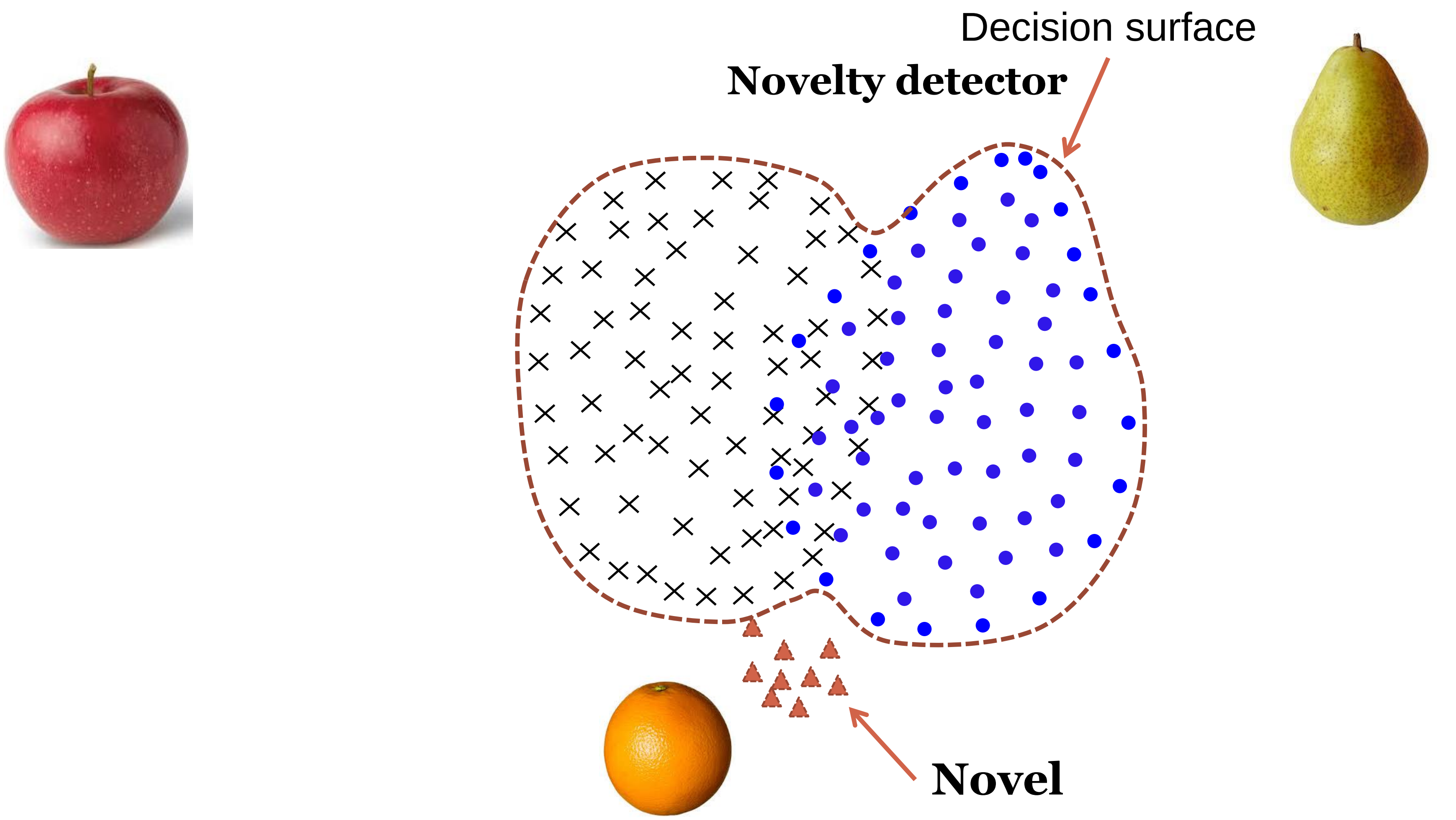
Invest Northern Ireland

- Researchers

  Yi Cao
  Eduardo Gerlein
  Scott McDonald
  Fan Sun
  Yauheniya Shynkevich

Yi Cao, Yuhua Li, et al. (2015)
"Adaptive hidden Markov model with abnormal states for price manipulation detection," IEEE Transactions on Neural Networks and Learning Systems 26(2) 318 – 330.

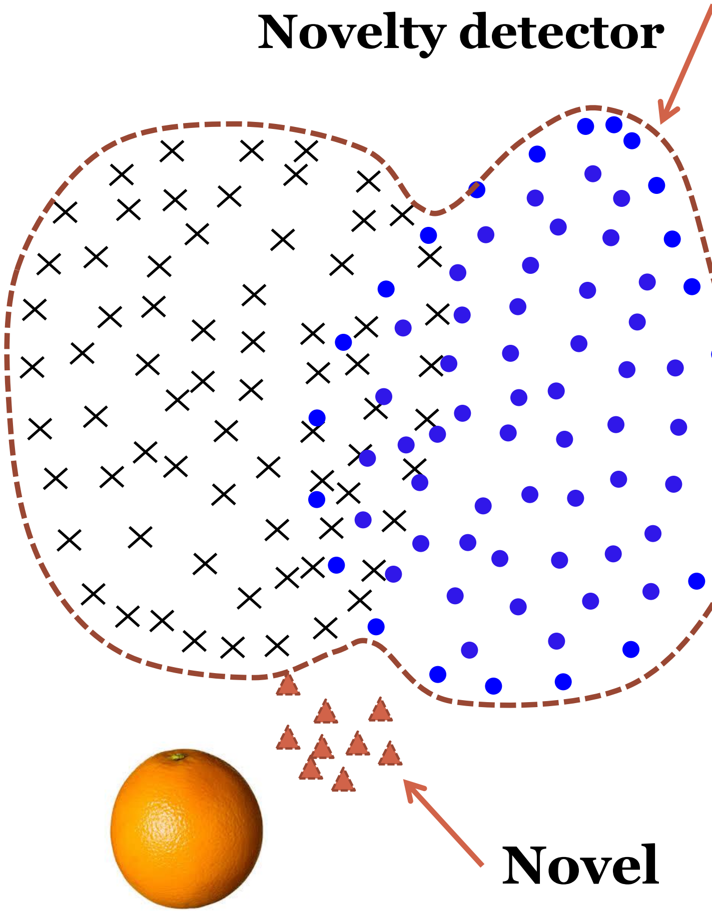
New Method for Novelty Detection

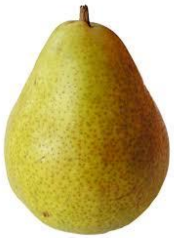


Questions

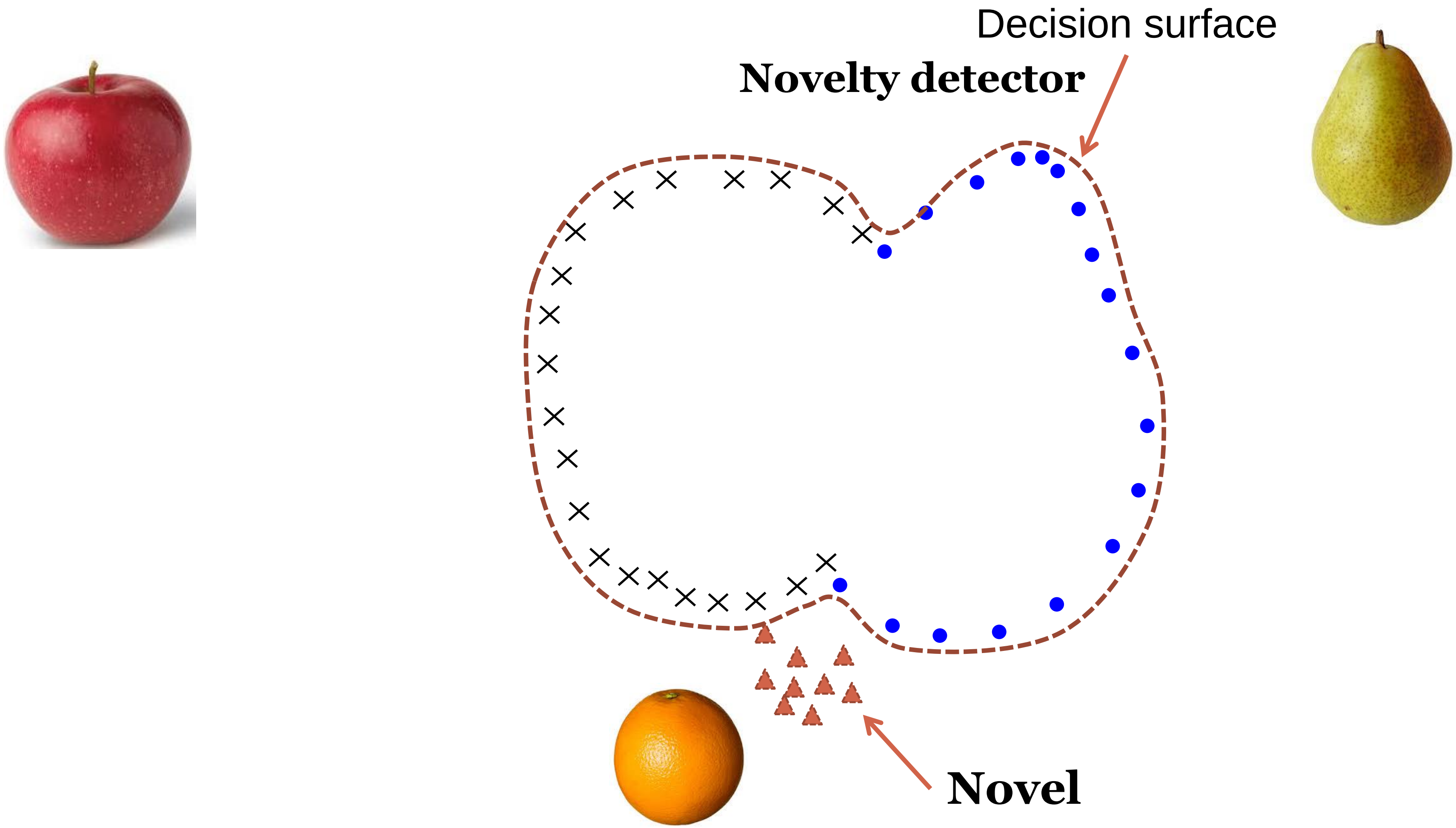
- Do we need all the sampling points (patterns)?
  - If not, which patterns are critical?
- How to identify the critical patterns?

Critical Patterns based Novelty Detection

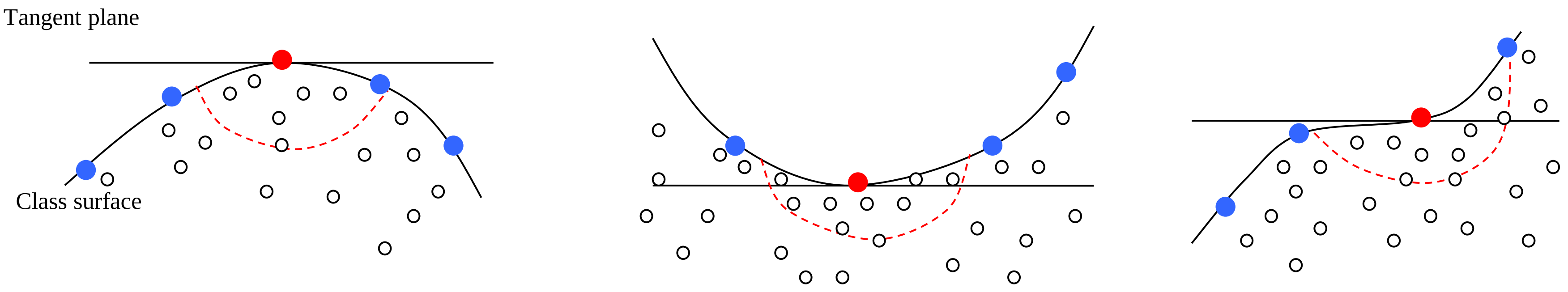




Critical Patterns based Novelty Detection



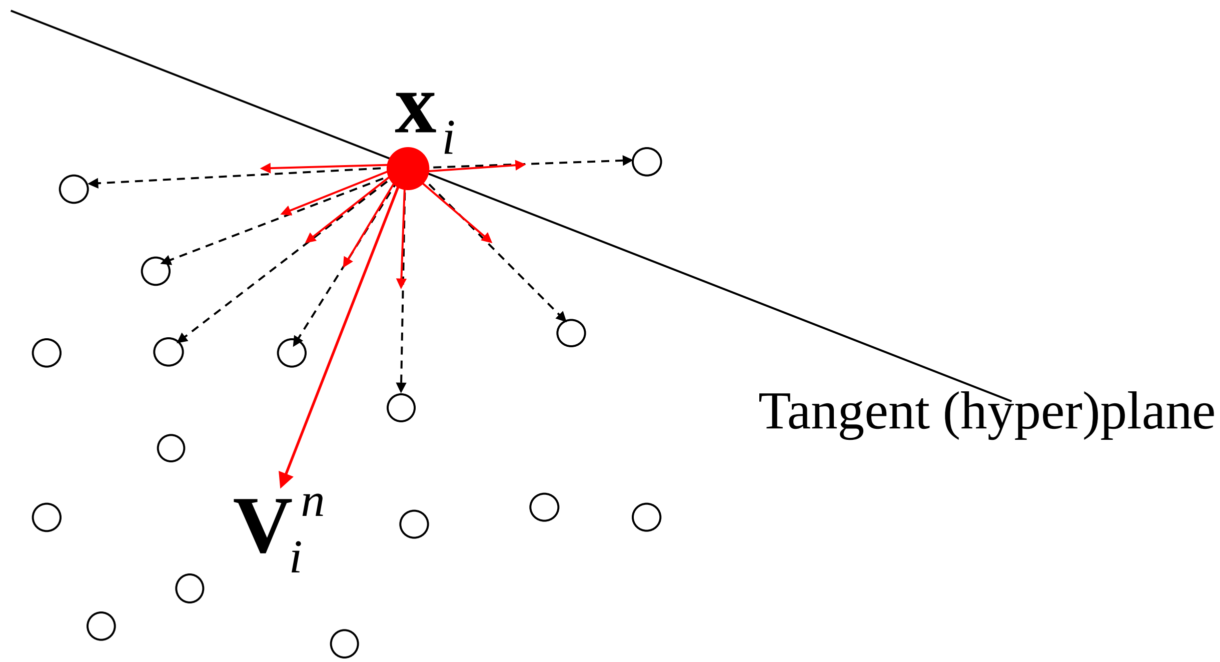
Critical Patterns based Novelty Detection



(a) (b) (c)

Construction of Tangent Plane

- Use of local geometrical and statistical information



Edge Patterns Selection

- Local geometrical information
  - Implicit function surface *S*: $f(x_1, x_2, \cdots, x_d) = 0$

    Let *C* be a curve defined by differentiable parametric functions $x_1(t)$, $x_2(t)$, …, $x_d(t)$ which lies on the surface *S*
  - Then the tangent vector **T** to the curve *C* at the point is given by

$$\mathbf{T} = \left(\frac{d}{dt} x_1(t), \frac{d}{dt} x_2(t), \cdots, \frac{d}{dt} x_d(t) \right)$$

also

$$\frac{df}{dt} = \frac{\partial f}{\partial x_1} \frac{dx_1}{dt} + \frac{\partial f}{\partial x_2} \frac{dx_2}{dt} + \cdots + \frac{\partial f}{\partial x_d} \frac{dx_d}{dt}$$

$$= \nabla f(x_1, x_2, \cdots, x_d) \cdot \mathbf{T} = 0$$

where $\nabla f(x_1, x_2, \cdots, x_d) = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \cdots, \frac{\partial f}{\partial x_d} \right)$ is the gradient of *f* at the point

  - So $\nabla f(x_1, x_2, \cdots, x_d)$ must be normal to the surface *S*

Edge Patterns Selection

- Local statistical information
  - Construct a hyper sphere with radius *r* centred at **x**

$$\Gamma(\mathbf{x}) = \{\mathbf{y} : distance(\mathbf{y}, \mathbf{x}) \le r\}$$

    The sphere has a volume of *v* and contains *k*-nearest neighbours $(\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_k)$

  - Estimate the expected vector of **y** in $\Gamma(\mathbf{x})$

$$E\{\mathbf{y} - \mathbf{x}\} \cong \int_{\Gamma(\mathbf{x})} (\mathbf{y} - \mathbf{x}) \frac{p(\mathbf{y})}{p(\mathbf{x}).v} d\mathbf{y} \cong \frac{r^2}{d+2} \cdot \frac{1}{p(\mathbf{x})} \cdot \nabla p(\mathbf{x})$$

  - For a given point **x**, the local mean $E\{\mathbf{y} - \mathbf{x}\}$ can be estimated by the mean of its *k*-nearest neighbours
  - Thus the gradient vector $\nabla p(\mathbf{x})$ at point **x** is estimated as

$$\nabla p(\mathbf{x}) = \frac{d+2}{r^2} \cdot p(\mathbf{x}) \cdot \left(\frac{1}{k} \sum_{i=1}^{k} (\mathbf{x}_i - \mathbf{x}) \right)$$

Edge Patterns Selection

// i: index for patterns in the dataset
// j: index for patterns in *k*NN

For a given pattern $\mathbf{x}_i$

find *k*NNs for $\mathbf{x}_i$

for *j*=1, 2, …, *k*

draw a vector $\mathbf{v}_{ij}$ from $\mathbf{x}_i$ to its *j*th nearest neighbour

normalise $\mathbf{v}_{ij}$ to unit vector $\mathbf{v}_{ij}^u$

add up all $\mathbf{v}_{ij}^u$ to approximate normal vector: $\mathbf{v}_i^n = \sum_{j=1}^{k} \mathbf{v}_{ij}^u$

for *j*=1, 2, …, *k*

calculate dot product $\theta_{ij} = \mathbf{v}_{ij}^T \cdot \mathbf{v}_i^n$

if $\theta_{ij} \geq 0$

increase counter *l* by one

find the ratio of *k*NNs with $\theta_{ij} \geq 0$: $l_i = \frac{1}{k} l$

if $l_i \geq 1 - \gamma$

select $\mathbf{x}_i$ as an edge pattern

end

Yuhua Li, Liam Maguire (2011) “Selecting critical patterns based on local geometrical and statistical information,” IEEE Trans on Pattern Analysis and Machine Intelligence 33(6), 1189-1201.

Level set methods

- The ice cube melts or freezes as temperature increases or drops, this results in the interface (between ice and water) moving in space over time
- LSM are a well-established and powerful collection of numerical algorithms for tracking the motion of dynamic implicit surfaces/interfaces
- They were pioneered by American mathematicians Osher and Sethian in 1988
- LSM employ an implicit function, called LSF, to represent complicated boundaries, and then advance the boundaries using the time-dependent PDE which govern the dynamics of the boundaries' evolution.
- Hence LSM can be considered as a class of deformable models.

Level set methods

- Example

  $x^2 + y^2 = r^2$

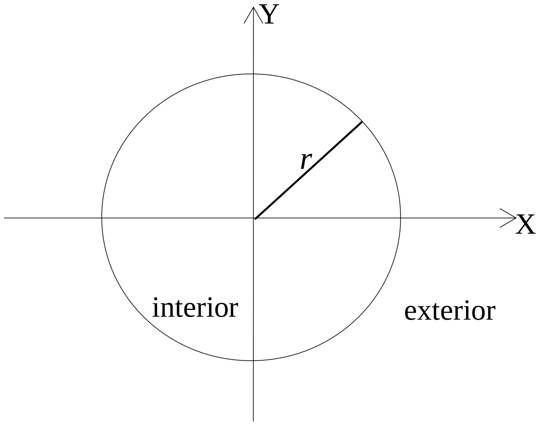
  $\varphi(\vec{p}) = x^2 + y^2 - r^2$

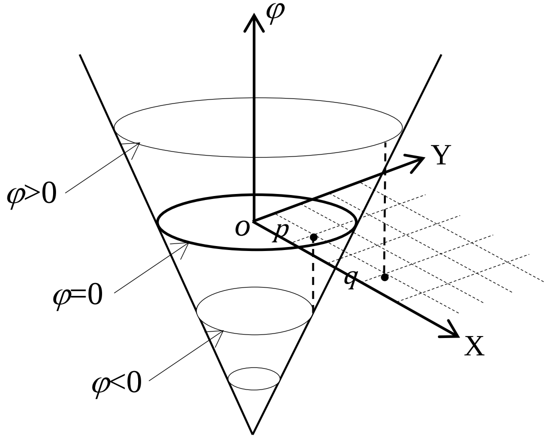
  Circle: $\varphi(\vec{p}) = 0$

  Interior: $\varphi(\vec{p}) < 0$

  Exterior: $\varphi(\vec{q}) > 0$

  2-d closed circle is described using a 3-d function





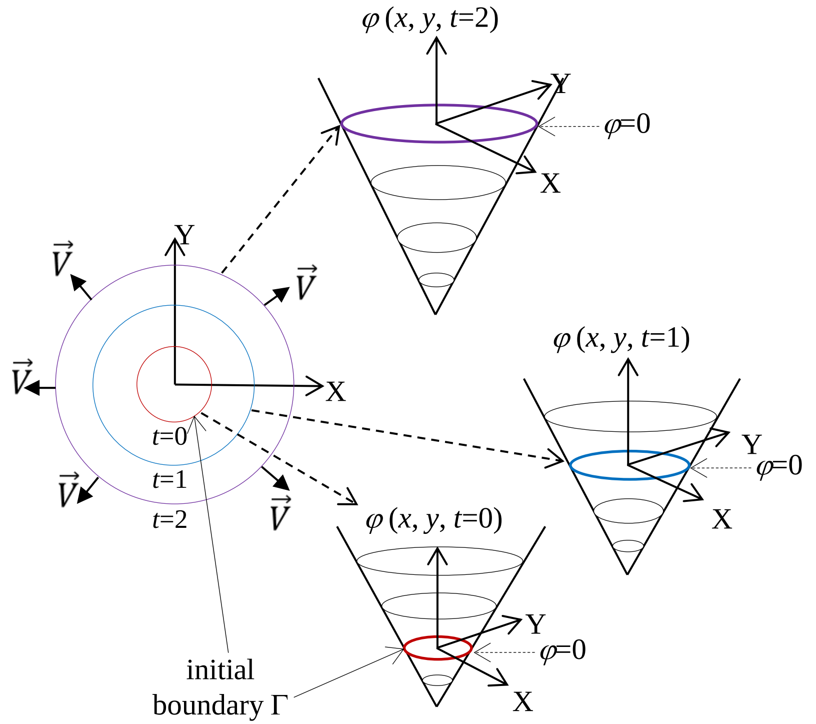
Level set methods

- The original idea behind the LSM is simple:
  - given a boundary Γ in $\mathbb{R}^d$ of co-dimension one, bounding an open region Ω, the boundary subsequent motion can be computed under a self-generated velocity field $\vec{V}$ that can depend on the position, time, and the geometry of the boundary
- The evolution direction and speed is controlled by $\vec{V}$ and the magnitude of $\vec{V}$, respectively.
- The boundary evolution is governed by PDE

$$\frac{\partial \varphi}{\partial t} + \vec{V} \cdot \nabla \varphi = 0$$

- Normally evolution speed, *a* , is a function of the points on the boundary surface ($\vec{x}$) and the time variable *t*., then LSE:

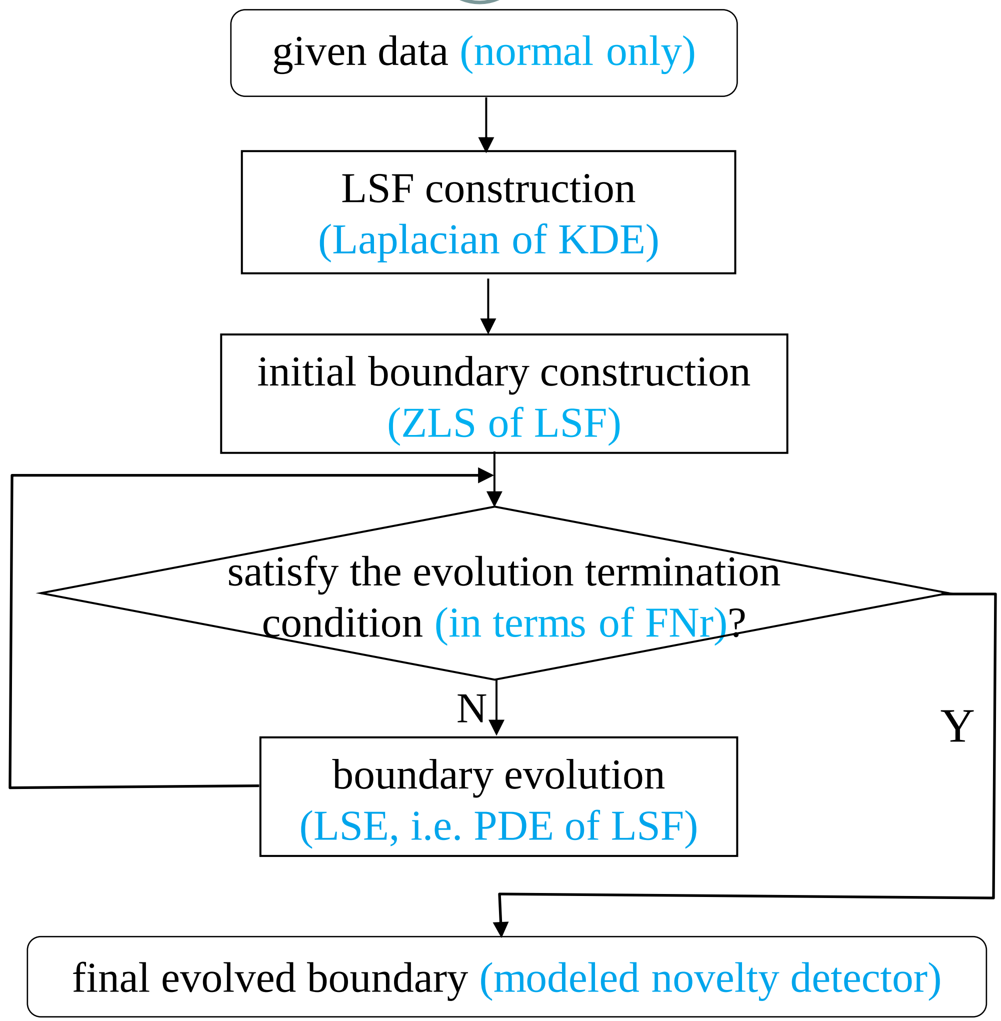
$$\frac{\partial \varphi(\vec{x}, t)}{\partial t} + a(\vec{x}, t)|\nabla \varphi(\vec{x}, t)| = 0$$



Novelty Detection Using LSM

- It constructs decision boundary directly in the input space rather than in a feature space using kernel trick;
- Fully data-driven boundary evolution;
- Nonparametric;
- No any assumption on data distribution.

Novelty Detection Using LSM



Novelty Detection Using LSM

% *training*: given available normal dataset

M1: LSF construction ------------Training Process----------

*dens*1 = apply KDE to *training*

g = construct a grid in the given space occupied by *training*

*dens*2 = evaluate KDE on *g*

φ = approximate the Laplace's differential operator on *dens*2

M2: Current λ_i computation

s = 0 % s: an accumulator for exterior points

for each point $\vec{x_i} \in training$

if $\varphi(\vec{x_i}) > 0$ **then** *s* = *s* +1 **end if**

Calculate $\lambda_i = \frac{s}{|training|}$ **end for**

M3: Boundary evolution

while $(\lambda_i \notin [\lambda - \varepsilon, \lambda + \varepsilon])$

if $\lambda_i < \lambda - \varepsilon$ **then**

φ = shrink the current *φ* applying LSE(14) with *a*<0

λ_i = apply M2 to *training* using *φ*

else if $\lambda_i > \lambda + \varepsilon$ **then**

φ = expand the current *φ* applying LSE (14) with *a*>0

λ_i = apply M2 to *training* using *φ* **end if**

end while

% detection: unseen dataset

------------Detection Process------------

for each point $\vec{x_i}$ ∊ *detection*

if $\varphi(\vec{x_i}) > 0$ **then**

$\vec{x_i}$ is detected abnormal

else

$\vec{x_i}$ is detected normal

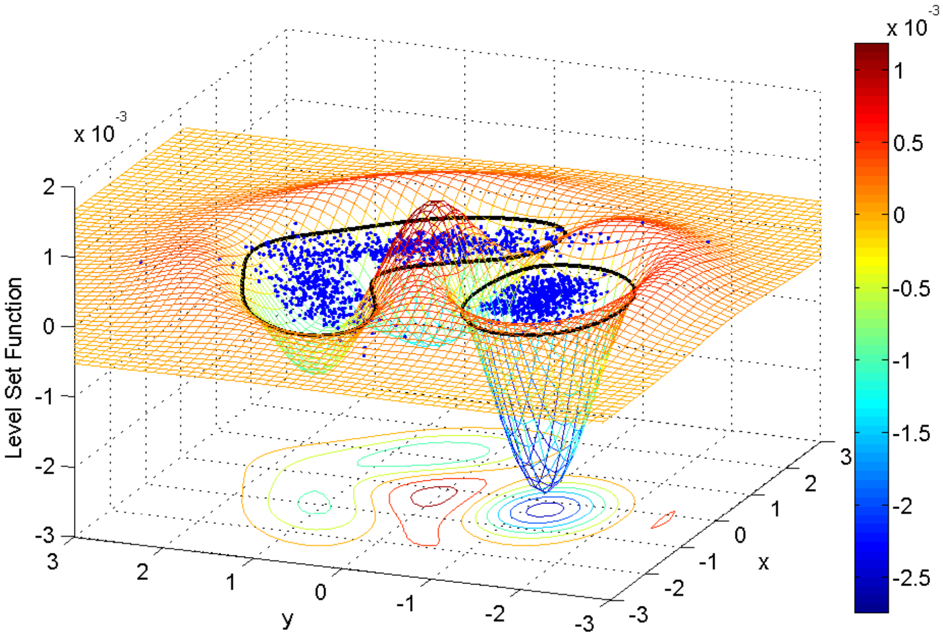
end if

end for

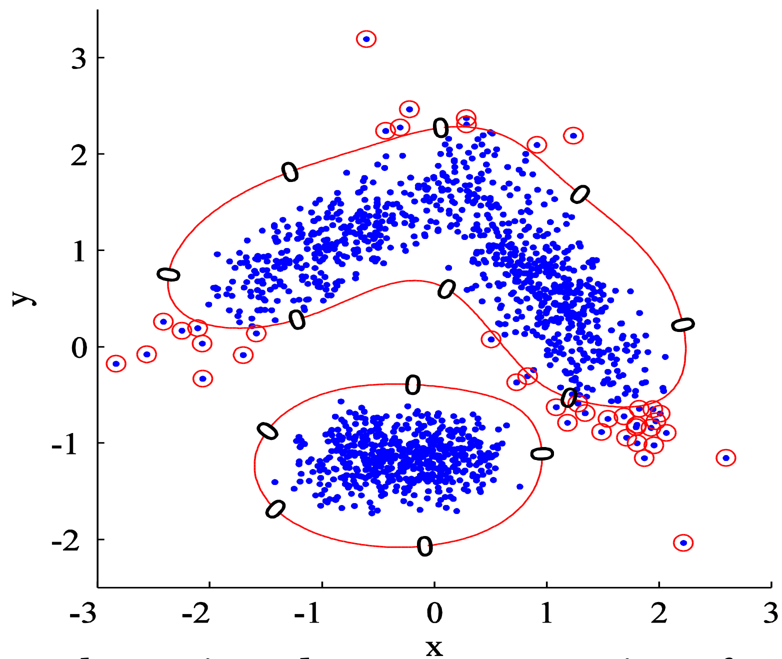
Xuemei Ding, Yuhua Li, et al. 2015, 'Novelty detection using level set methods' , IEEE Transactions on Neural Networks and Learning Systems, 26 (3) , pp. 576-588

Illustration example

- One intermediate visualization during the training process with 3D data



The implicit LSF φ. φ=0 defines the boundary (two black closed curves)



The projected 2-D representation of the boundary $\partial\Omega$ where φ=0