McMaster University

Integrated Biomedical Engineering & Health Sciences Program

IBEHS - 4QZ3
Modelling of Biological Systems

Lecture 10

TAYLOR DEVET MASC.

PHD. CANDIDATE BIOLOGICAL AND BIOMEDICAL ENGINEERING

MCGILL UNIVERSITY

SHRINERS HOSPITAL FOR CHILDREN

Todays Aims…

Phase Space

Attractors

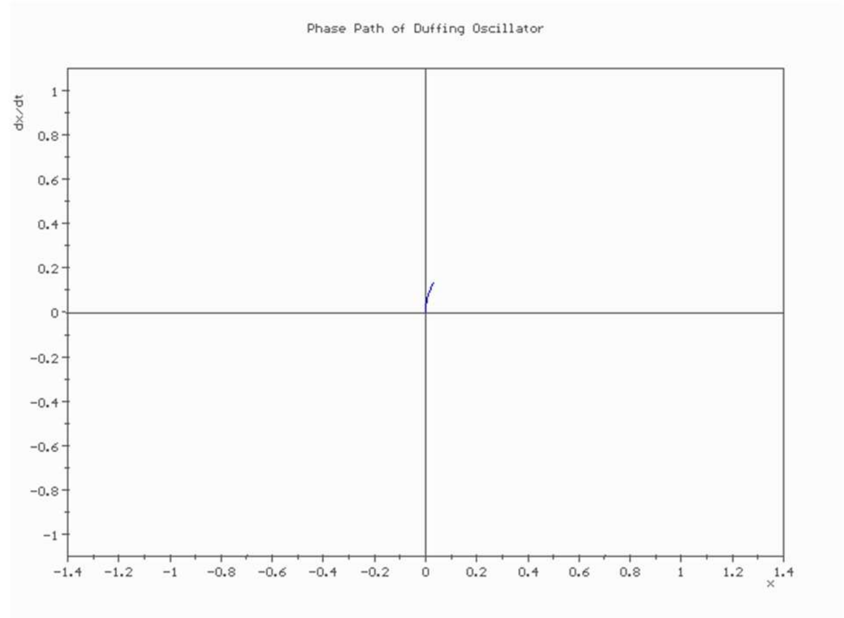
Machine Learning

Other Features of Chaos – Phase Space

- If there are n-variables, then the state of the system can be described as a point in an n-dimensional space whose coordinates are the values of the dynamical variables.

Representations of all possible values a system can take on

- values of variables evolve in time (coordinates of the point in phase space move). Thus can analyze time behavior of the system by analyzing the motion of this point that represents the values of the variables.



Phase Space

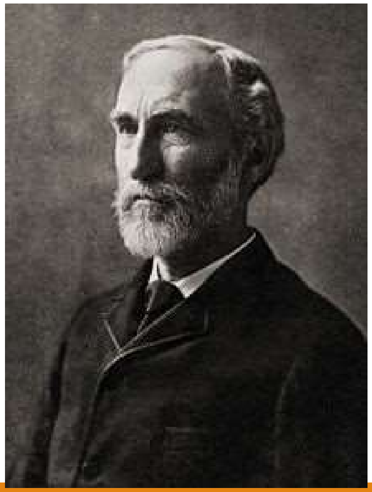
- analyze properties of a dynamical system by determining the topological properties of the phase space trajectory

Late 1800s

Henri Poincaré
- Discovered deterministic chaotic systems

Ludwig Boltzmann
- Studied thermodynamics
- Boltzmann constant

Josiah Willard Gibbs
- Studied thermodynamics
- Gibbs free energy

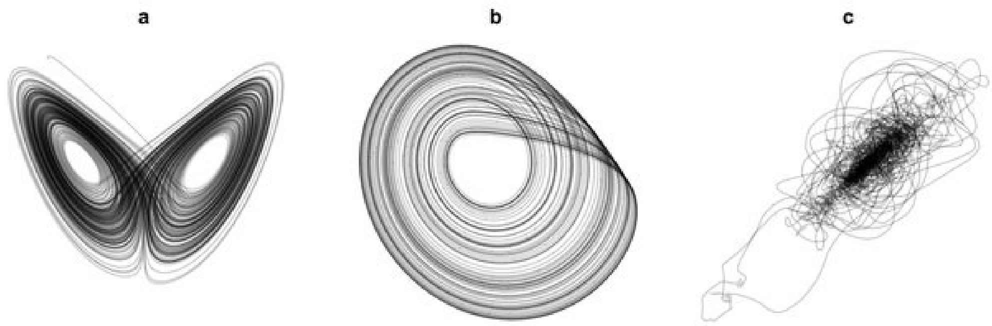


Other Features of Chaos – Attractors

- The point in phase space that represents how the system moves as the values of the variables evolve in time.

If the initial state of the system is not on the attractor, then the phase space point moves exponentially rapidly toward the attractor as time goes by.

Due to sensitivity to initial conditions, two points on the attractor diverge exponentially fast from each other as time goes by, even though they both remain on the attractor.



Other Features of Chaos – Strange attractors

- An attractor is typically finite in phase space.

- Sensitivity to initial conditions

- however, cannot diverge forever

Thus, trajectories from nearby initial points on the attractor diverge and are folded back onto the attractor, diverge and are folded back, etc.

- attractor structure consists of many fine layers.

- Thus, the attractor is fractal.

Other Features of Chaos – Strange and chaotic

"chaotic" = sensitivity to initial conditions

"strange" = fractal attractor

The typical chaotic system is therefore chaotic and strange

- there are also chaotic systems that are not strange - there are non-chaotic systems that are strange

Other Features of Chaos – Dimension of the attractor

The fractal dimension (FD) of the attractor is related to the number of independent variables needed to generate the time series of the values of the variables.

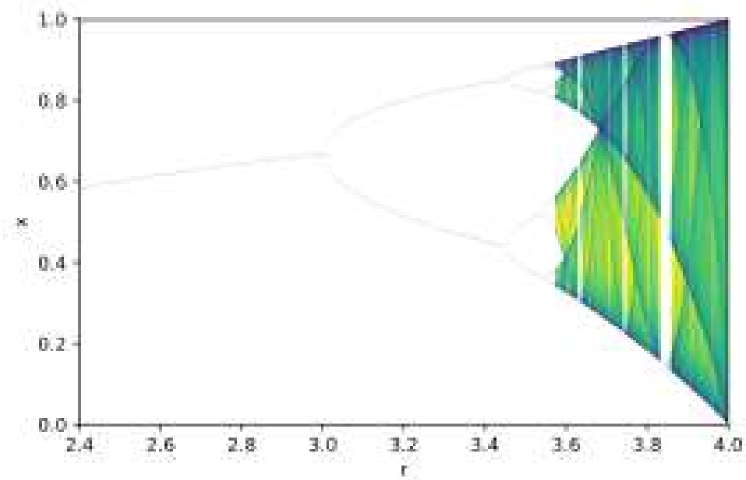
d is the smallest integer greater than the fractal dimension of the attractor

- the time series can be generated by a set of d differential equations with d independent variables.
- e.g. $FD_{attractor} = 2.03$
- the time series of the values of the variables can be generated by 3 independent variables in 3 coupled nonlinear differential equations.

Other Features of Chaos - Bifurcations

A bifurcation occurs when the form of the trajectory in phase space changes because a parameter passes through a certain threshold value.

- often useful to plot how the form of the dynamics depends on the value of a parameter.



Other Features of Chaos - Control

Linear systems

- small changes in the parameters added to the values of the variables produce small changes in subsequent values of the variables.

Nonlinear systems

- small changes in the parameters added to the values of the variables can produce enormous changes in subsequent values of the variables because of the sensitivity to initial conditions.

Thus, in principle, a chaotic system can be controlled faster and finer and requires smaller amounts of energy for such control than a linear system.

How to analyze real data for Chaotic Dynamics

1. Calculate the FD of the temporal domain (i.e. time data), or frequency data

2. Transform the time series into a geometric object in phase space and then analyze the topological properties of this set
   - i.e. instead of using fractals to analyze the values of the time series itself, use fractals to analyze a representation of the time series in a phase space.

FD of the phase space set is different than the fractal dimension of the values of the time series itself, and it conveys different information about the nature of the process that produced the data.

Chaos vs Noise?

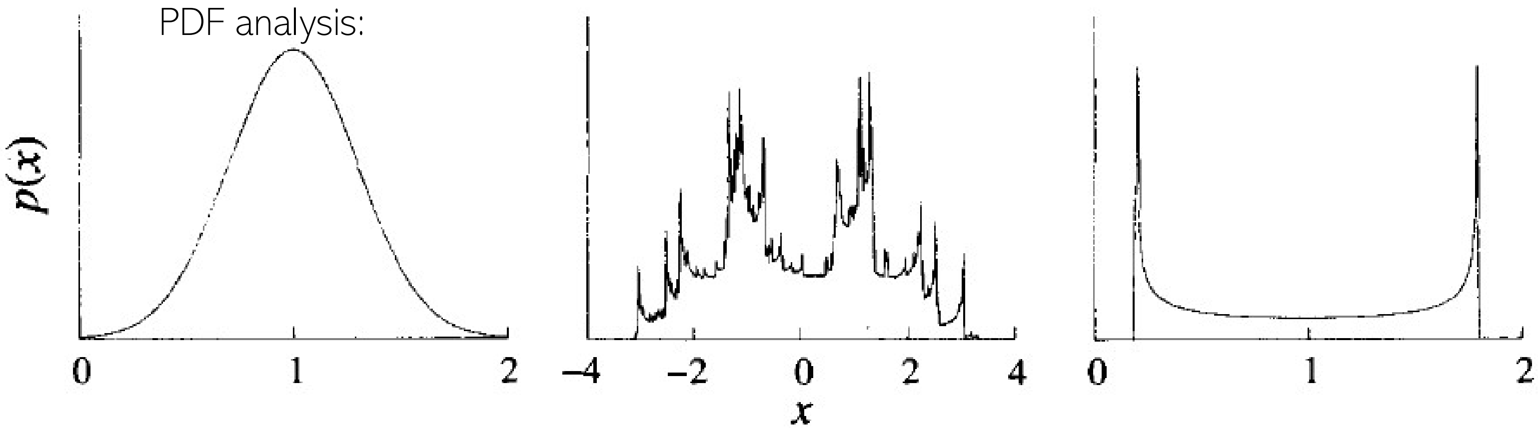
A general principle to keep in mind when trying to make the distinction between chaos and noise:

- A very high order system will not be distinguishable from noise unless the system is so dominated by a few variables that it behaves as a low dimensional attractor

Methods for Determining If Signals Are Fractal

1. Power spectral analysis.
2. Autocorrelation function.
3. Dispersion analysis (i.e. RD).
4. Standard statistical measures (e.g. PDF analysis).

PDF analysis:



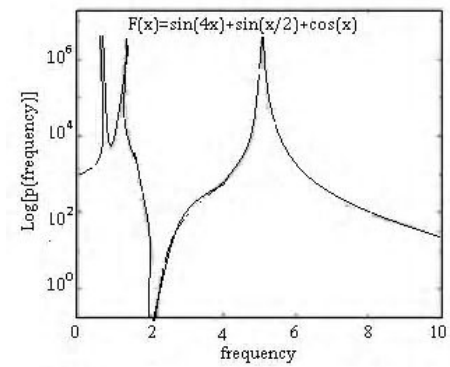
Methods suggestive of underlying chaos:

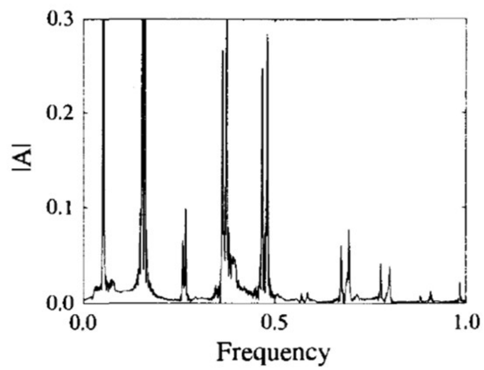
1. Visual inspection for irregularity and bifurcations in periodicities
2. The power spectral density
3. Autocorrelation function

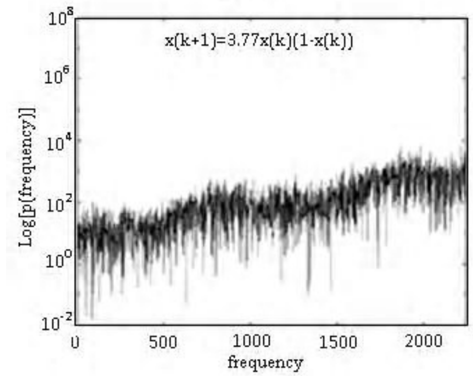
Methods suggestive of underlying chaos: Visual and Power

1. "Visual" methods depend on the appearance of the signal to give:
   - an impression of irregularity in rhythm or amplitude,
   - the absence of periodic or repeating segments of the same form

2. Power spectral analysis with FFT
   - a chaotic signal commonly shows broadly spread peaks in the power spectrum.







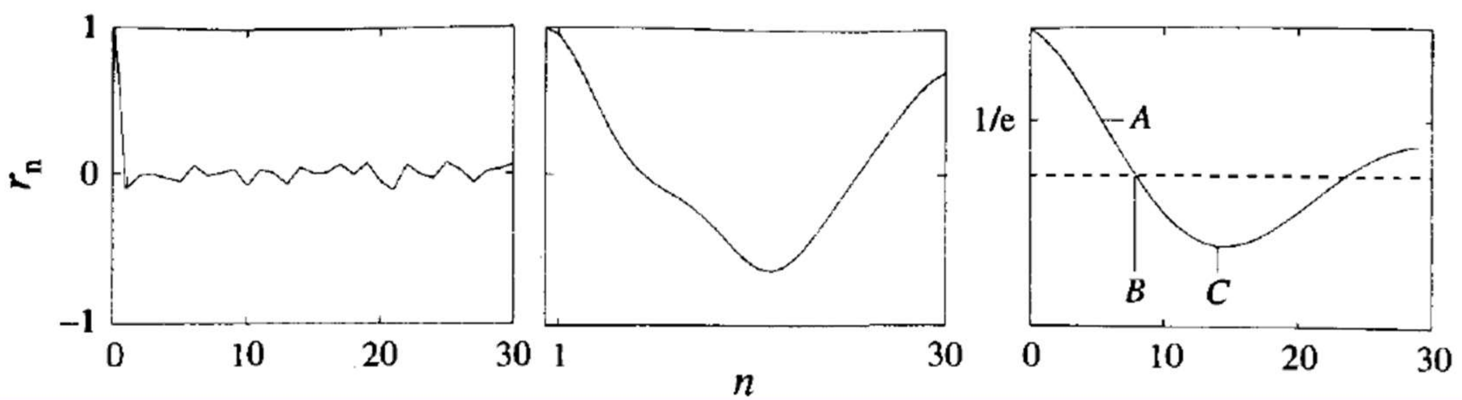
Methods suggestive of underlying chaos: Autocorrelation

- positive correlation over even a short range distinguishes the chaotic signal from random noise even though it will not necessarily distinguish chaos from filtered or smoothed noise.

- noise function shows no correlation even over short times (i.e. rn=0).

- spring function shows only short range correlation, and then some periodicity in the correlation function.

- the fractal correlation also provides an estimate of the time lag to be used in the embedding to create a pseudo-phase space reconstruction of the signal to see if it has the features of a chaotic attractor (right).



Methods suggestive of underlying chaos: Autocorrelation

Typical correlation coefficient:

$$r = \frac{Cov(Y_i, Y_j)}{Var(Y)}$$

Yi and Yj are subsets of observations of Y

Autocorrelation Function:

$$r_n = \frac{N}{N-n} \frac{\sum_{i=1}^{i=N-n} Y_i Y_{i+n} - \frac{\left(\sum_{i=1}^{i=N} Y_i\right)^2}{N}}{\sum_{i=1}^{i=N} Y_i^2 - \frac{\left(\sum_{i=1}^{i=N} Y_i\right)^2}{N}}$$

(i.e. autocovariance / variance)

Special methods for characterizing low dimensional chaotic signals:

1. Visual analysis of phase plane plots
2. Correaltion, and information dimension
3. Estimation of Lyapunov exponents
4. Calculations of entropy

Special situations

1. Observations of changing states exhibiting bifurcations or changes of periodicities

2. Interventions resulting in period doublings or changes in apparent chaotic state

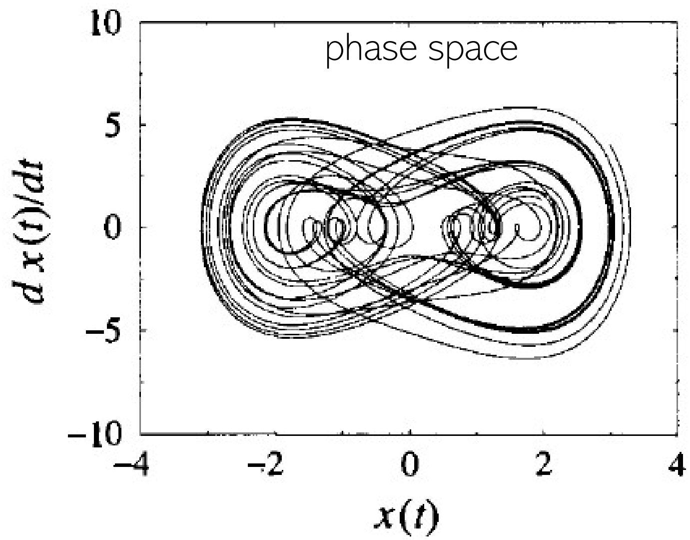
Pseudo-phase space plots

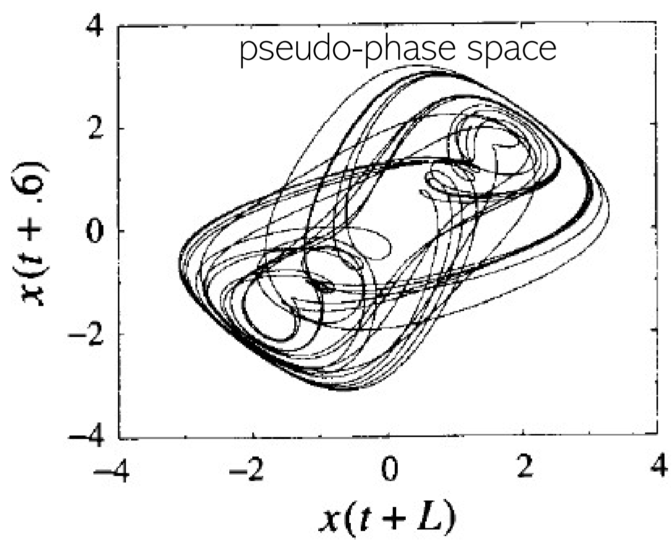
Embedding: Turning the time series into a set in pseudo-phase space.

Phase Space: consider 3 independent variables can plot variable X vs. Y OR plot of derivative of a variable vs. variable (left).

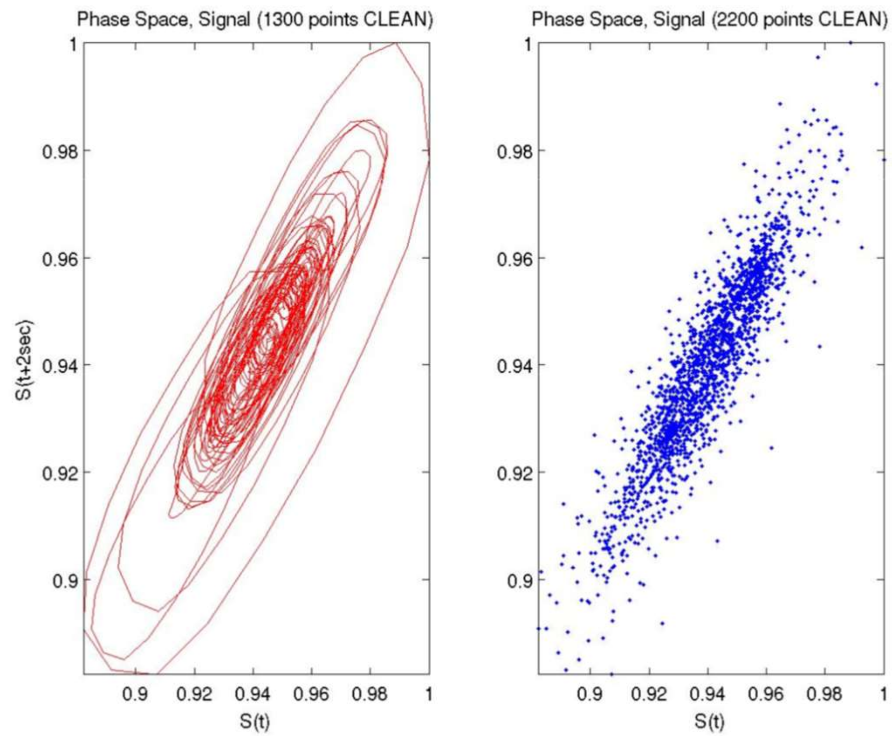
Pseudo-phase space:

plot variable+lag vs. variable

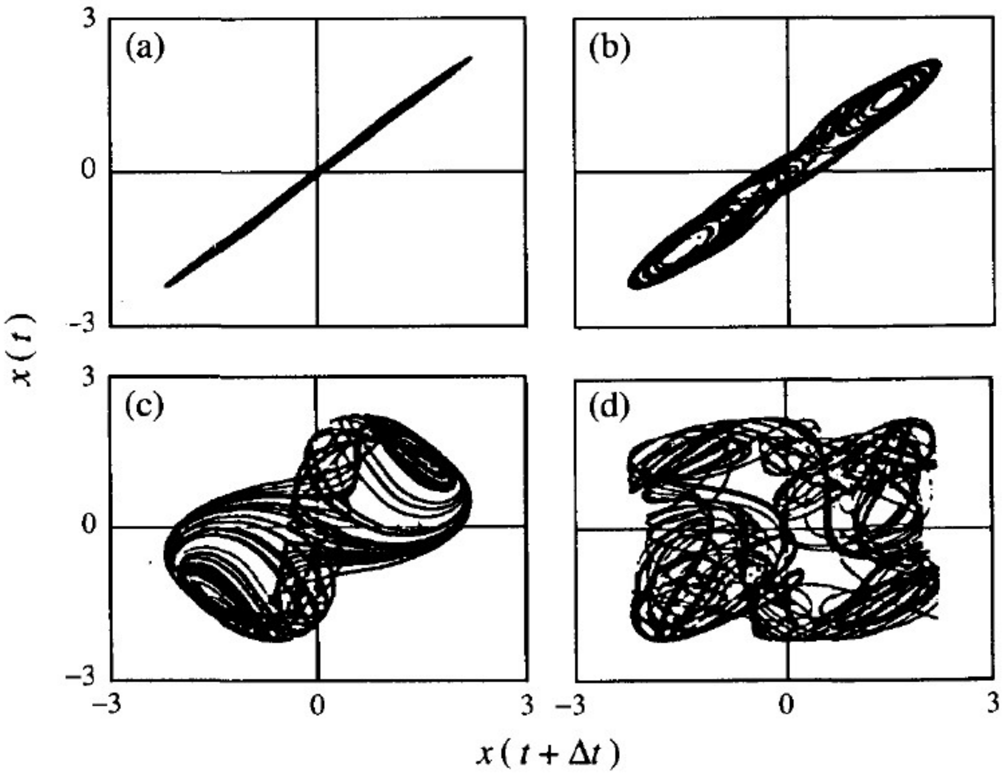




Pseudo-phase space plots



Reconstructions with lags

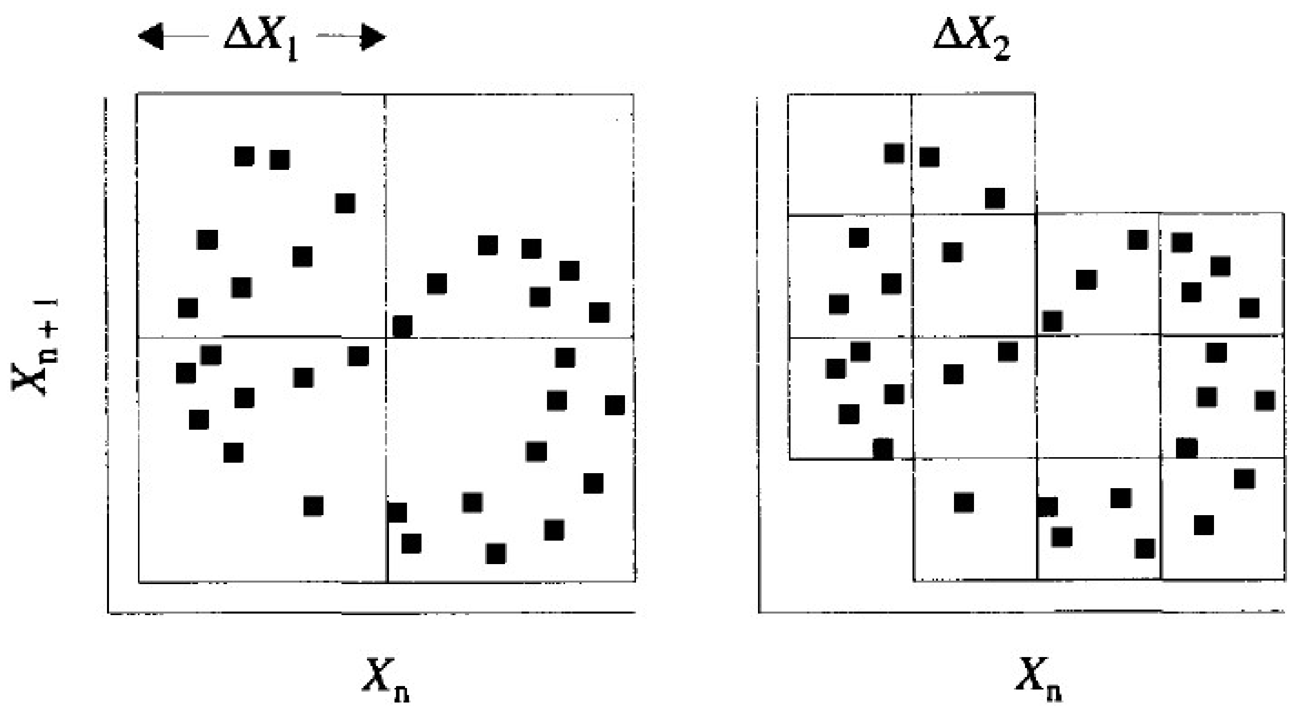


(a) 0.05
(b) 0.25
(c) 1.25
(d) 6.25

Box counting:

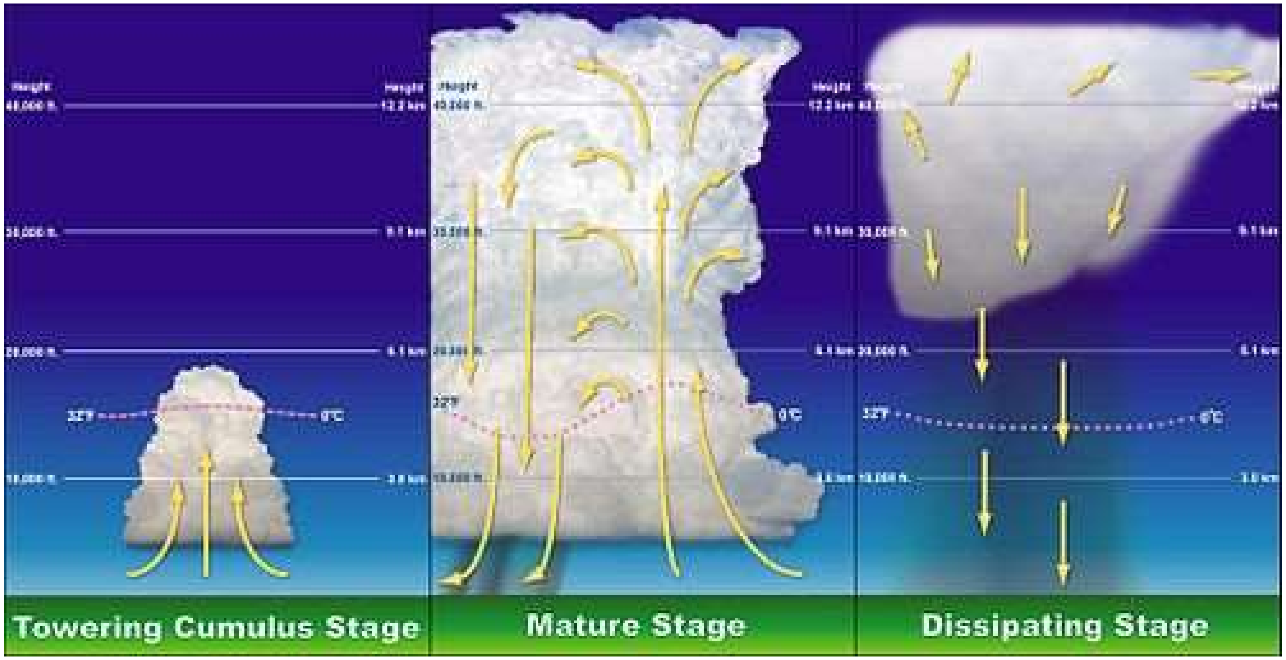
Topology of determining fractal dimension of phase space

Box counting to determine FD of 2D space. The # of boxes N(r) occupied by at least one point of the set is determined for boxes of ever smaller size r. The fractal dimension is the slope of logN(r) versus log(l/r).



Lorenz Model

A famous model exhibiting chaotic behavior was first developed in 1963 by Edward Lorenz to describe complex atmospheric convection.



Lorenz Model

The Lorenz equations represent an extreme simplification of the Navier-Stokes equations describing fluid flow between two boundaries held at different temperatures representing the Earth's surface and the upper atmosphere.

The Lorenz model consists of the set of first-order, autonomous, nonlinear differential equations:

$$\frac{dx}{dt} = -\sigma(x + y)$$

$$\frac{dy}{dt} = -xz + rx - y$$

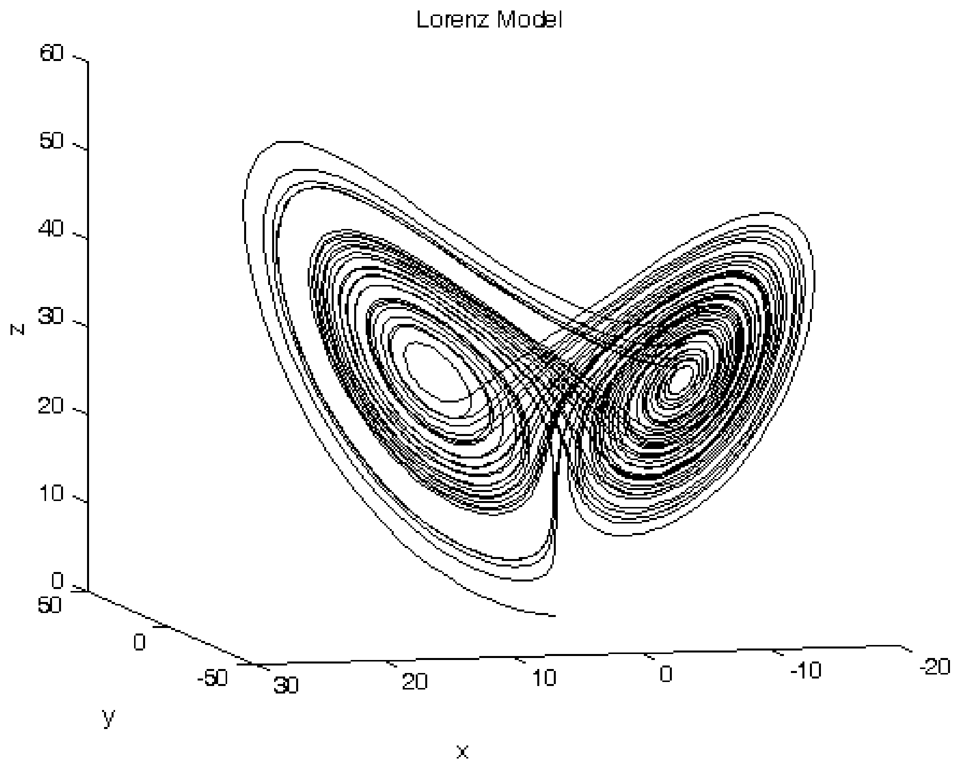
$$\frac{dz}{dt} = xy - bz$$

σ includes fluid velocity and thermal conductivity (Prandl number).

r = Rayleigh number

b = geometry dependent factor

Lorenz Model



The solution of this system generates a three-dimensional phase-space orbit for

σ=10

r=28

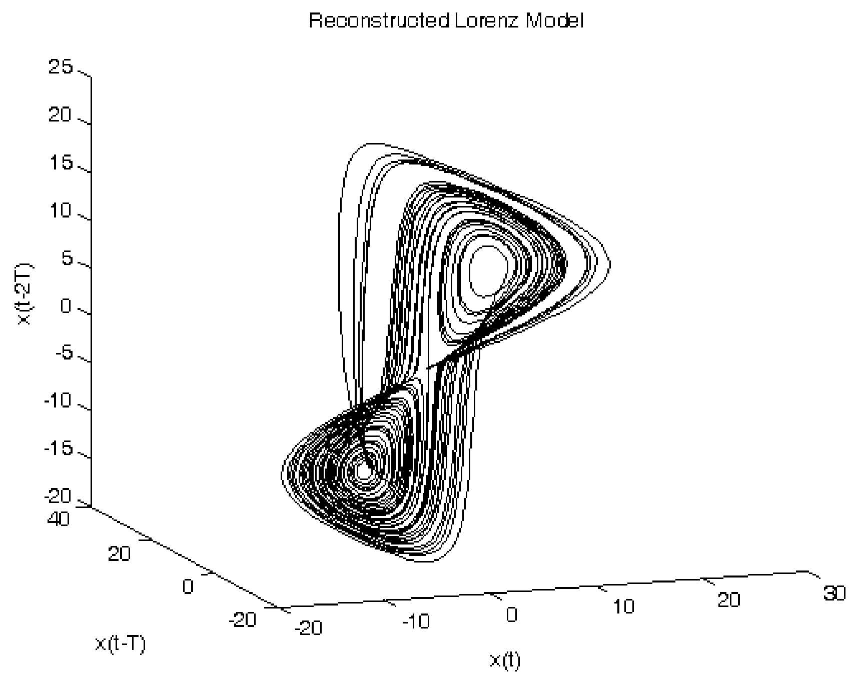
b=8/3

Try Matlab code: Lorenz_reconstruct.m

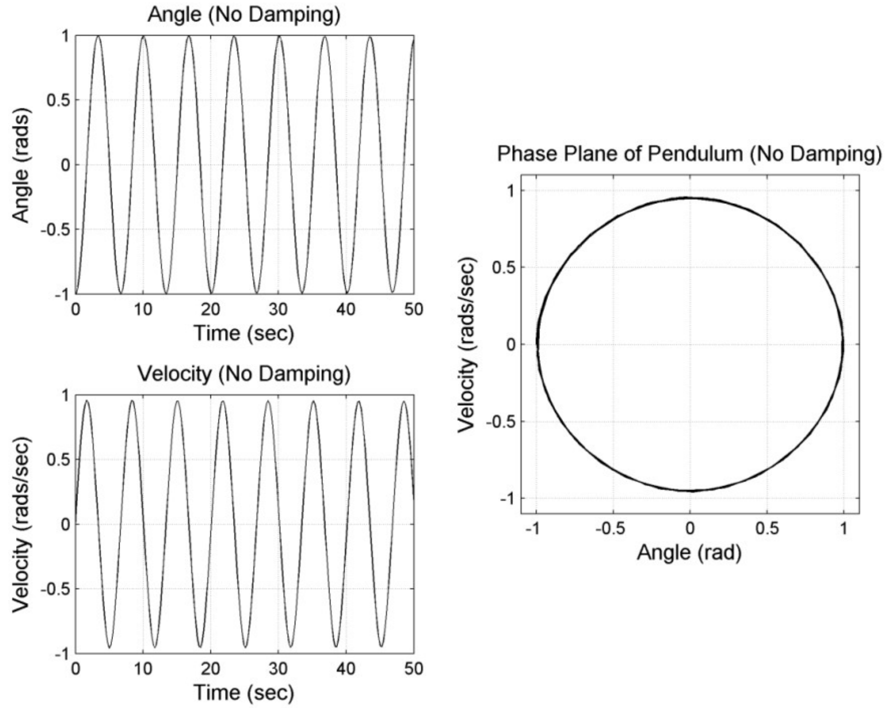
Lorenz Attractor

- The reconstruction is performed in two dimensions by plotting x(t) vs. x(t-τ) where τ is a suitably chosen time delay.

Lorenz attractor reconstructed from the x time series using time delay coordinates.

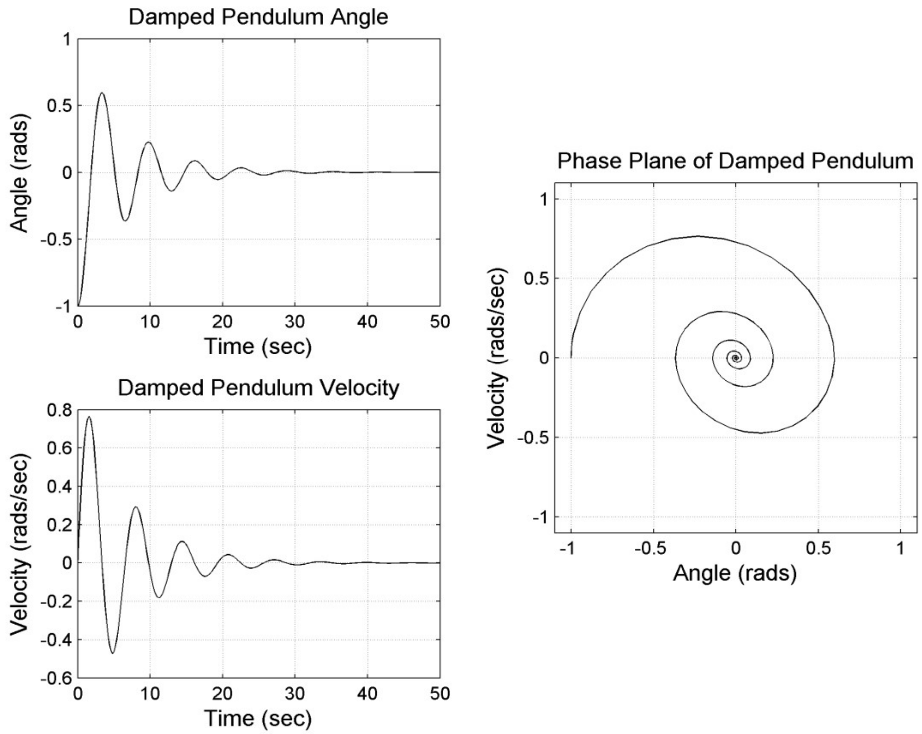


Visualizing the state-space with a phase-plane plot



- A plot of the state variables against each other
- If the phase-space has 2 dimensions, a plot is known as the phase-plane
- Depicts the attractor of the system
- The attractor is the tendency of the system at equilibrium. In this example it is a limit cycle

Visualizing the state-space with a phase-plane plot



- Pendulum with damping
- Here the attractor is a stable node

Phase-space plots of measured data

- The phase-space can give us important information about a system (from its signals)
- Nonlinear systems can have complicated phase-space plots
- However, we often do not have access to all the variables.
  - For example, body temperature is one variable of the inflammation response but others are invasive
- Is there a way to recover, or at least estimate the phase-space of a multidimensional system from a phase-space plot?

Answer: Yes, With Delay Embedding

Floris Takens showed that the method of delayed embedding sufficiently reconstructs a time series

Reconstruction is just an estimate, not perfect

Method

- Use a delayed version of the signal as a "new" dimensional measurement time series. Each additional series is delayed by a multiple of the chosen delay τ

Formally we would say time series $x[n]$ of length N can be reconstructed into multidimensional time series $y[n_d, k]$ of k dimensions from 1 to m, where each delayed vector n_d comes from $x[n]$ delayed by τ

$$y[n_d, k] = x[n + (k-1)\tau, k] \ldots x[N - (m-1)\tau, m],$$

With MATLAB

- The matlab routine **`delay_emb`** is provided to perform the operation of delay embedding. The function has a format:

```
y=delay_emb(x,m,tau)
```

- Inputs
  - **`X:`** the original one-dimensional time series
  - **`m:`** the embedding dimension of new *m*-dimensional vector **`y`**
  - **`tau:`** the delay
- Outputs
  - y: a matrix of vectors representing the system in embedded space

Getting a good embedding

Takens showed that

- *m* should be twice D where D is the "true" dimension of the system
  - True meaning if we had access to each dimension
- Theoretically **τ** could be any number

However in practice

- *m* just needs to be "sufficiently" large
- **τ** can't be too large or too small
  - Each delayed vector should have some correlation to the previous vector, but not too much

Lorenz Equations

Before continuing, we revisit the Lorenz system

- useful for demonstrating concepts in the rest of the lecture.

Lorenz system

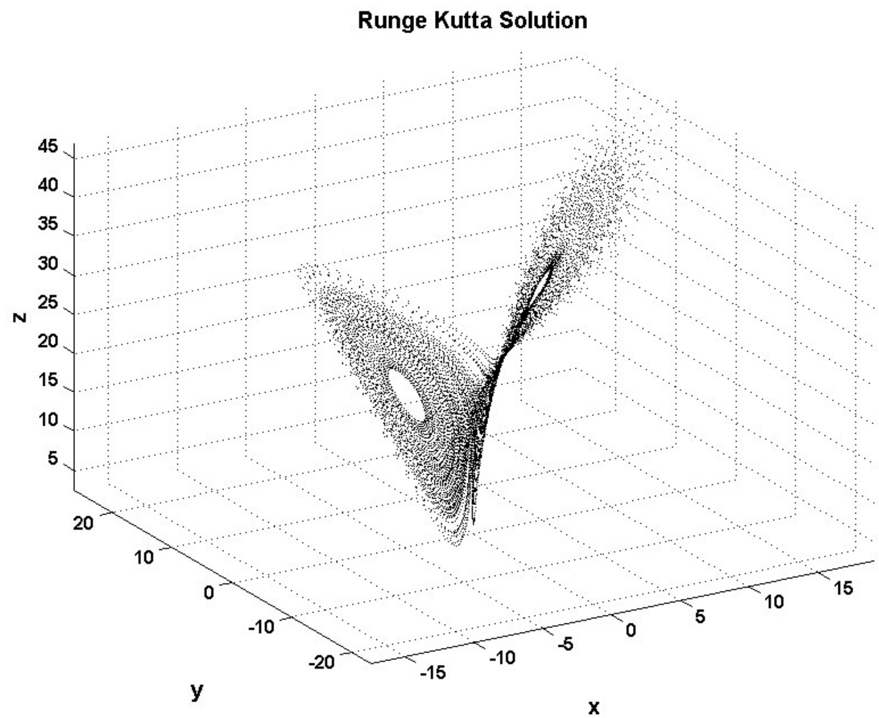
- Simplified fluid undergoing convection (the movement of fluid due to a heat source)
  - Here x, y and z are the spatial dimensions
- The constants
  - σ, the Prandtl number, the ratio of the viscosity of the fluid to the thermal conductivity
  - β, the Rayleigh number, a dimensionless number which represents a ratio of buoyant forces to viscous forces within the flow of a fluid
  - ρ, the density of the fluid.

$$\frac{dx}{dt} = \sigma(y - x)$$

$$\frac{dy}{dt} = x(\rho - z) - y$$

$$\frac{dz}{dt} = xy - z\beta$$

Lorenz System Phase Space



Methods of finding the delay

There is more leeway in choosing the best delay compared to choosing the best value for m, a range of delay values may be fine.

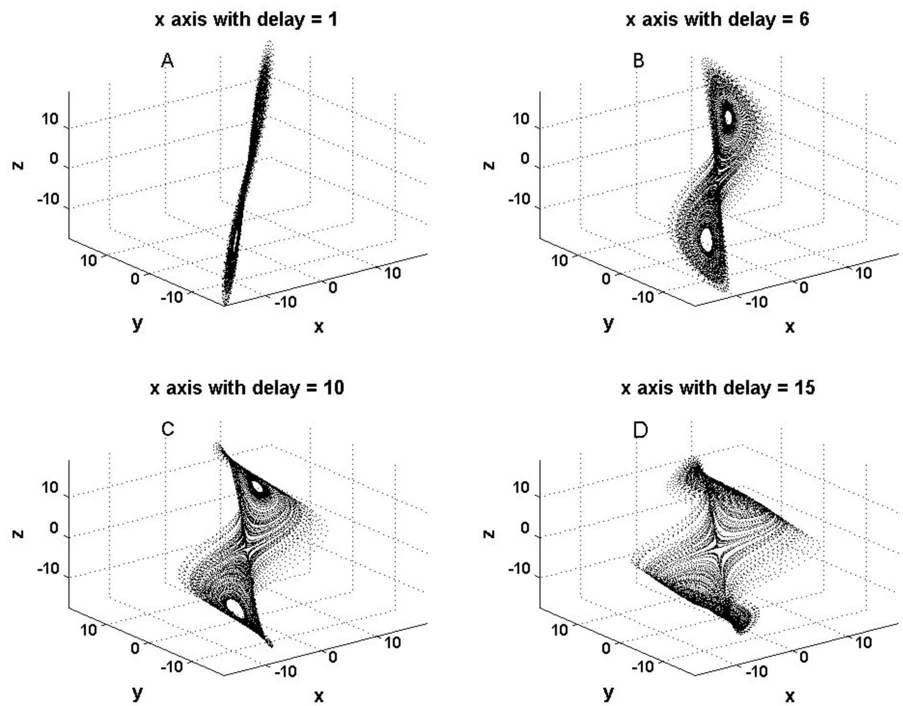
Trial and Error

- Simple but it works

Autocorrelation

- Minimum or 0

Estimating the Delay



Methods of finding embedding dimension

- Trial and error
- False Nearest neighbors
- Principle component analysis

False Nearest Neighbors

- Points along the trajectory of a properly unfolded phase-space have specific neighbors.
- If unfolded in too few dimensions, the neighbors may be false
  - For example points separated in the Z dimension might appear close together projected to the XY plane
- By examining the ratio of the change in false nearest neighbors, we can estimate a reasonable embedding dimension.

Distance of vectors

- Here we define distance of two vectors using the Euclidian no

$$d = \sqrt{\left(\vec{x}_a - \vec{x}_b\right)^2} \quad \text{or} \quad d = \left\| \overline{x_a} - \overline{x_b} \right\|$$

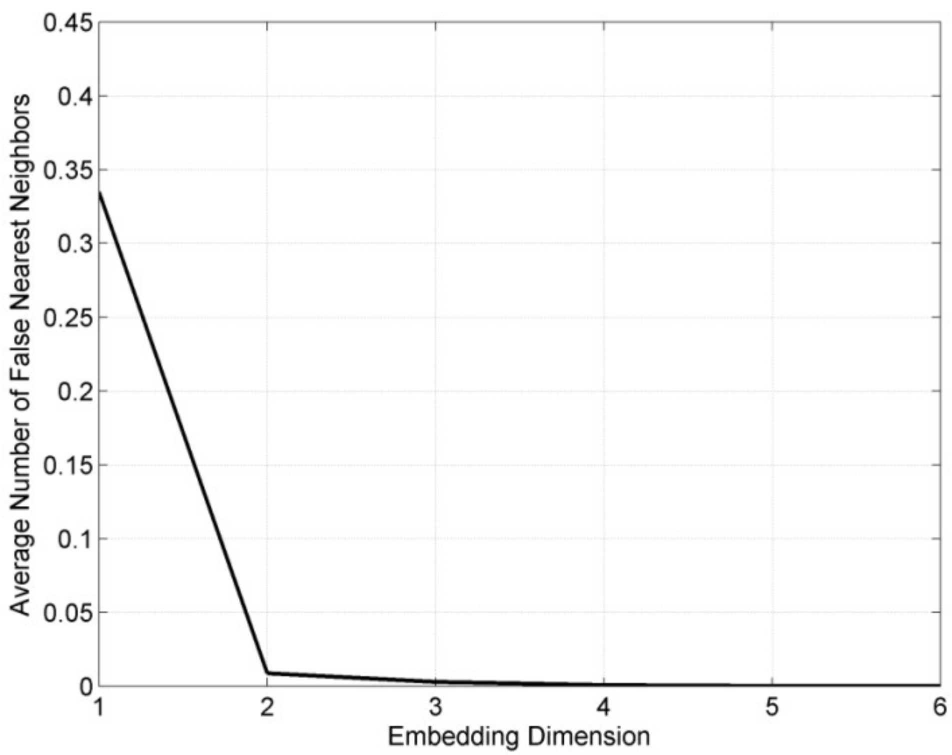
- Other definitions, such as the max norm, can also be used
  - Max norm is more computationally efficient

With MATLAB

numnear = **fnumnear(x,tau,em,r)**

- Inputs
  - **x:** input vector (i.e. the signal)
  - **tau:** is the delay used for embedding
  - **em:** maximum number of dimensions to test
  - **r**: the distance that defines nearest neighbors.
    - A range of **r** values should be tested,
    - Good starting point is 0.1 *std(**x**)
- Outputs
  - **numnear:** a series of ratio values of the number of neighbors in dimension m to dimension **m**+1

Nearest Neighbors for Lorenz System



We see a large drop off from dimension 1 to 2, and then a small drop off from dimension 2 to 3. After that there is negligible change.

Data Reduction using PCA

The eigenvalues describe how much of the variance is accounted for by the associated principal component and if a component is really necessary.

Eigenvalues are ordered by size; that is:

$$\lambda_1 > \lambda_2 > \lambda_3 \ldots > \lambda_M.$$

If an eigenvalue is zero or 'close to' zero, then its associated principal component contributes little to the data and can be eliminated. This component accounts for only a small amount of the variance in the data.

This tells us the effective dimension of the data set.

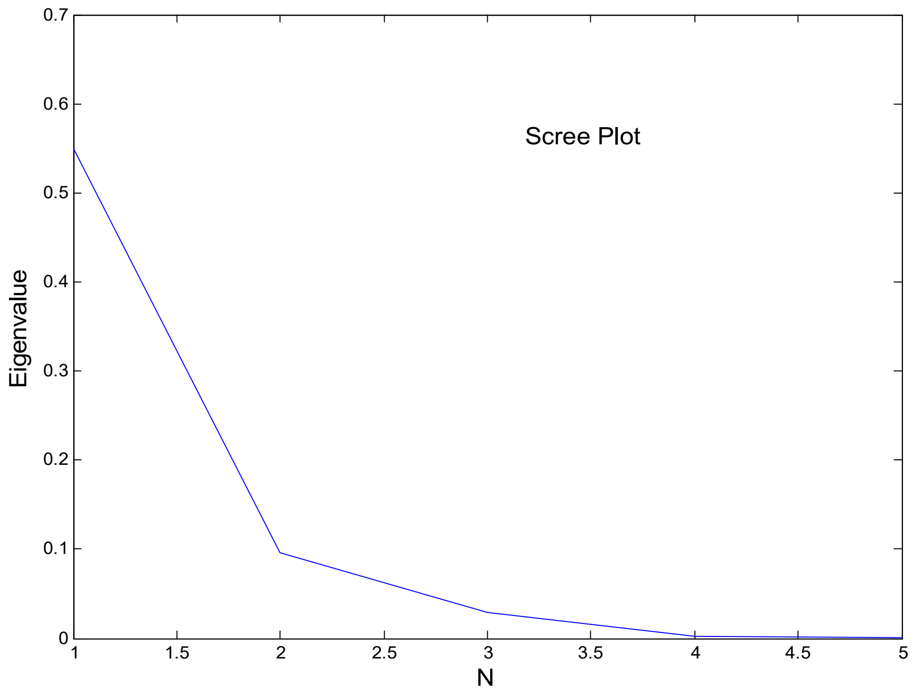
How do you decide if an eigenvalue is small enough so that its associated component can be removed from the data set?

The Scree Plot

The eigenvalues are in order of large to small. Plotting them in order will show their relative value.

Such a plot is called the Scree plot

The actual dimension of the data set is taken where the Scree plot becomes more-or-less flat.

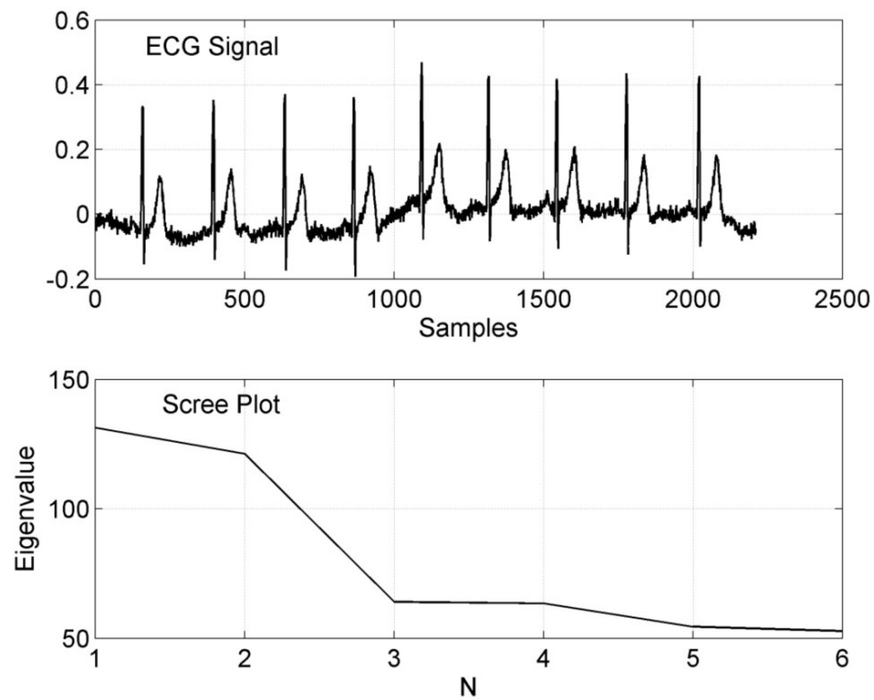


Finding *m* : PCA

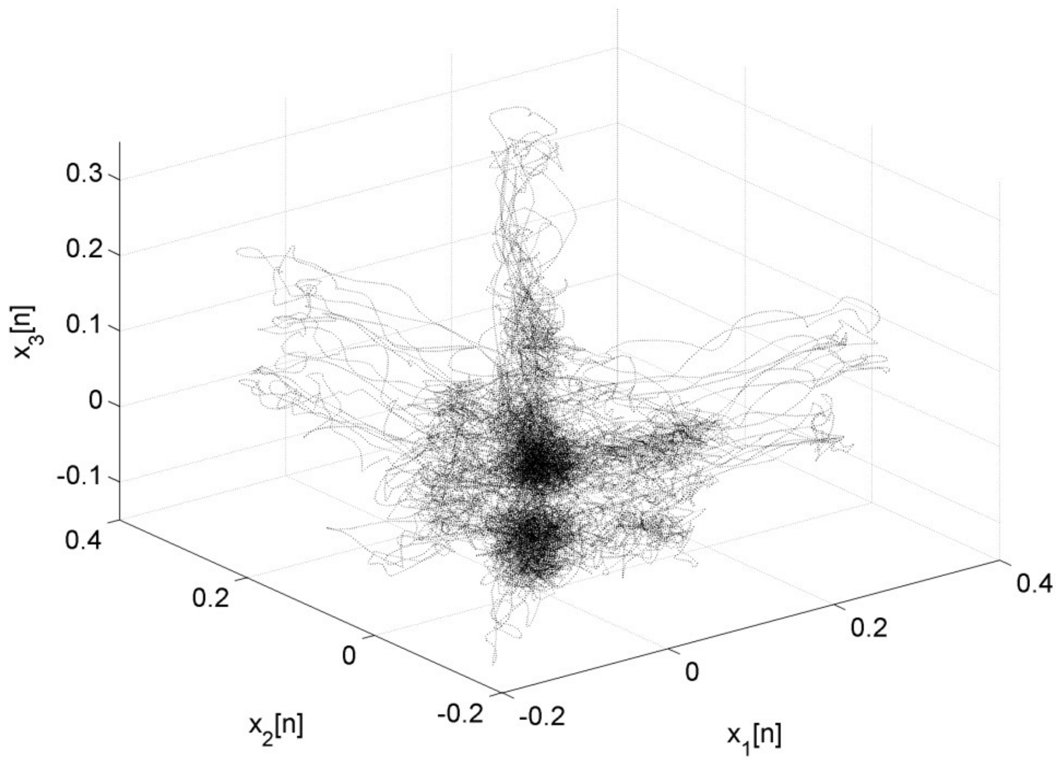
Use principle component analysis to determine the number of principle components and use that as your estimate for *m*

```
% Determination of embedding dimension using PCA
%
x=load('ECGtest.csv');          % Load the ECG file.
fs = 2100;                      % Sampling frequency
      m = 6;                          % Embedding dimension
%
X = delay_emb(x,m,tau);          % Generate delayed signals
   [U,S,pc]= svd(X' ,'econ');       % Perform the decomposition
   eigen = diag(S).^2;             % Get the eigenvalues
```

Scree plot showing 3 Dimensions



ECG embedded in 3 dimensions



The delay was found using the autocorrelation method and taken as 381 samples

Another nonlinear system: Logistic Map

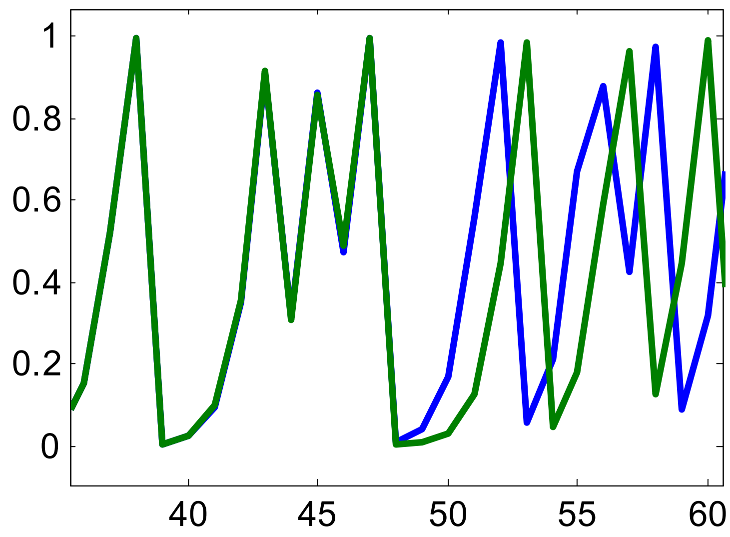
- A simple nonlinear system is known as the Logistic mapping function, a simplification of a Predator Prey model
- $x[n+1] = r*x[n]*(1-x[n])$
- Exhibits chaotic behavior if $r > 3.58$
- If $0 < x_0 < 1$, the output is always bounded by 0 and 1
- Phase-plane can be visualized by plotting $x[n+1]$ against $x[n]$

Properties of Chaos

- Exponential Divergence
  - Divergence is the phenomenon of trajectories of a system that begin with similar initial conditions ending up with very different trajectories.
  - This is the opposite of convergence, in which systems tend towards the same value over long periods of time
  - While nonchaotic systems may show divergence, only chaotic systems have trajectories that diverge exponentially.

Sensitivity to Initial Conditions

- Since the divergence is exponential, a plot of the log divergence should give a straight line
- The slope of the line gives an estimate of the Lyapunov exponent, a measurement of how quickly the divergence happens



Properties of the Lyapunov Exponent

For a multidimensional system, we measure divergence along the phase-space (or the estimated phase space)

The Lyapunov exponent (λ) is a measure of the divergence

- $\lambda > 0$ signifies chaotic behavior
- $\lambda = 0$ signifies a limit cycle
- $\lambda < 0$ converging trajectories.

A chaotic system does not have a single Lyapunov exponent

- There is a λ for each direction in phase-space: The *Lyapunov spectrum*
- Typically we are interested in the largest exponent or the sum of all exponents, as these give a general description of the system

When we refer to the Lyapunov exponent (symbol λ) we are referring not to the Lyapunov spectrum, but the single largest Lyapunov exponent.

Finding λ with MATLAB

Not always possible to take many measurements at similar initial conditions

Instead use nearest neighbors as substitutes for initial conditions of different trajectories

- provided they are sufficiently separated in time, why?

Since **λ** is not constant across the phase-space, we need to sample many areas of the phase-space

- The average of all the areas should approach the largest **λ**

After averaging the log divergence curves we examine the mean curve and look for a linear region

MATLAB Implementation

```
[lambda, S_mean,linear_end] = max_lyp(x,m,tau,fs,radius);
```

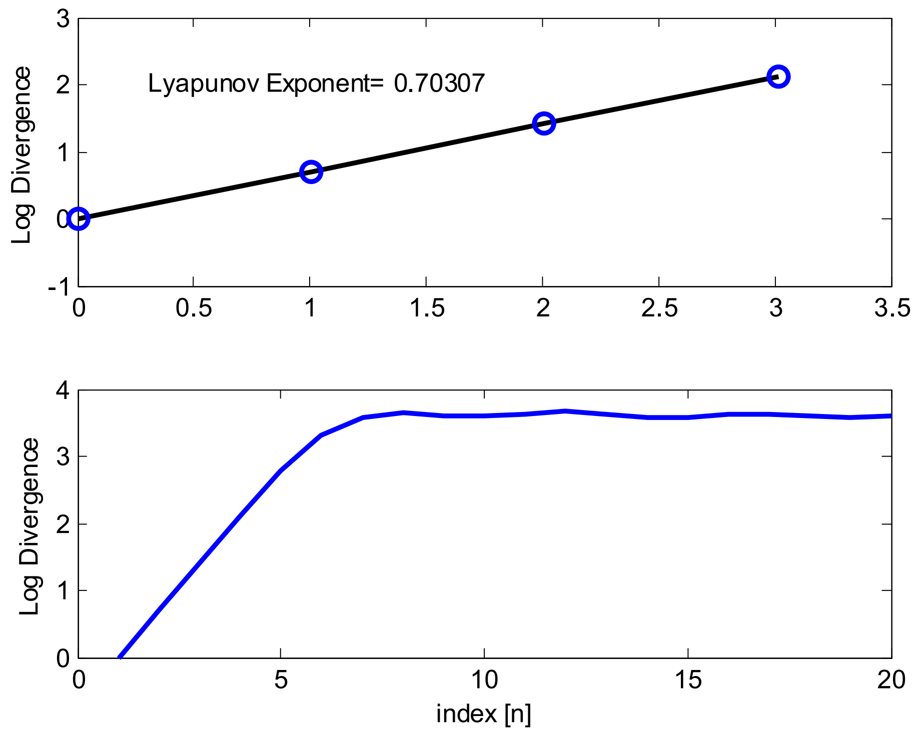
Inputs

- `x`: the time series to be analyzed the embedding
- `m`: and `tau`: the embedding parameters
- `fs`: the sampling frequency
- `radius`: the nearest neighbors cutoff.

Outputs

- `lambda`: the estimate of λ
- `linear_end`: the index of the end of the linear region

`max_lyp` output for Logistic map

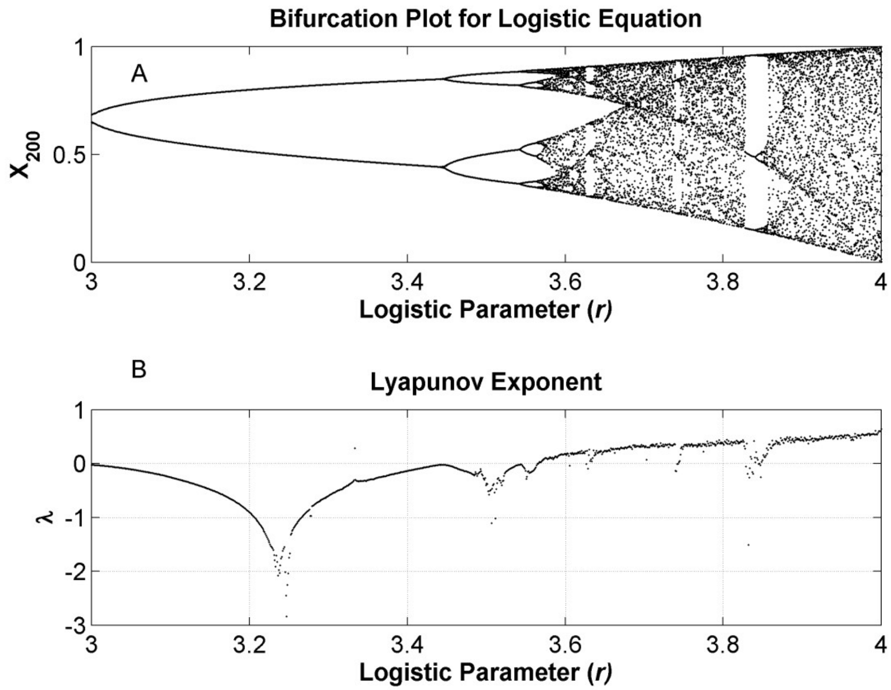


Important

- max_lyp will give you an estimate for **λ** whether it is appropriate or not

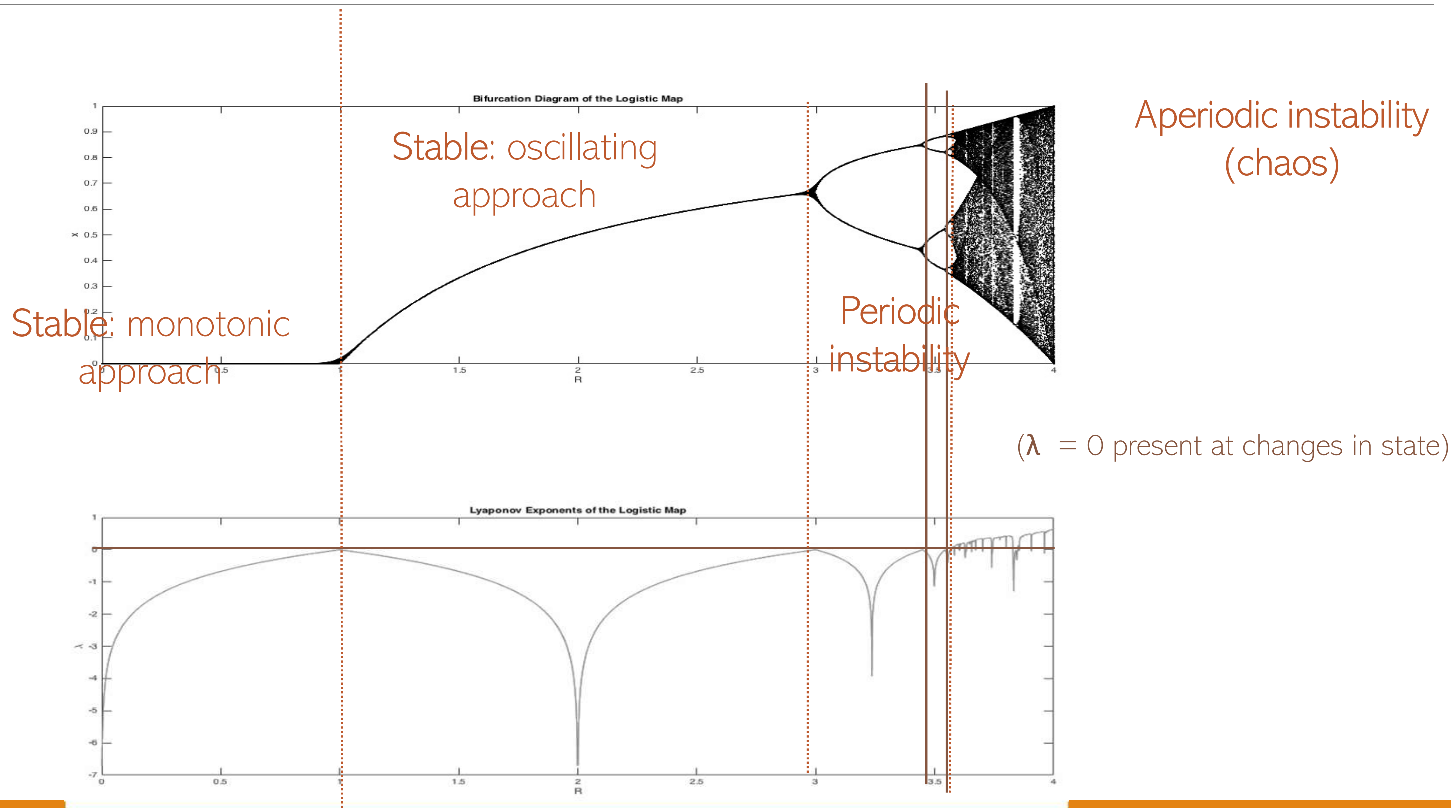
You must double check that the divergence is indeed linear

Bifurcation plot of the Logistic Map



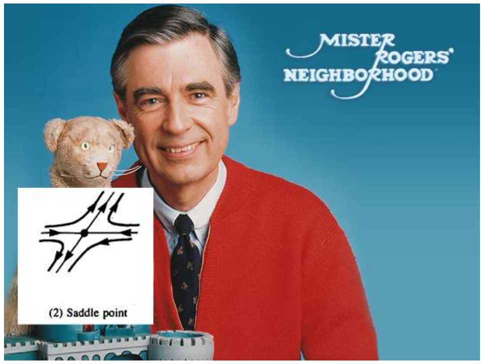
Lyapunov exponent

λ > 0 signifies chaotic behavior
λ = 0 signifies a limit cycle
λ < 0 converging trajectories



Node, Focus, or Saddle point

If the pair of ODEs are NOT LINEAR… you can approximate them using linear equations in the neighbourhood of the fixed point (x*,y*)



$$A = \left.\frac{\partial f}{\partial x}\right|_{x^*,y^*} \qquad B = \left.\frac{\partial f}{\partial y}\right|_{x^*,y^*}$$

$$C = \left.\frac{\partial g}{\partial x}\right|_{x^*,y^*} \qquad D = \left.\frac{\partial g}{\partial y}\right|_{x^*,y^*}.$$

f(x,y) and g(x,y) are the non-linear ODEs

$$\frac{dX}{dt} = AX + BY,$$

$$\frac{dY}{dt} = CX + DY.$$

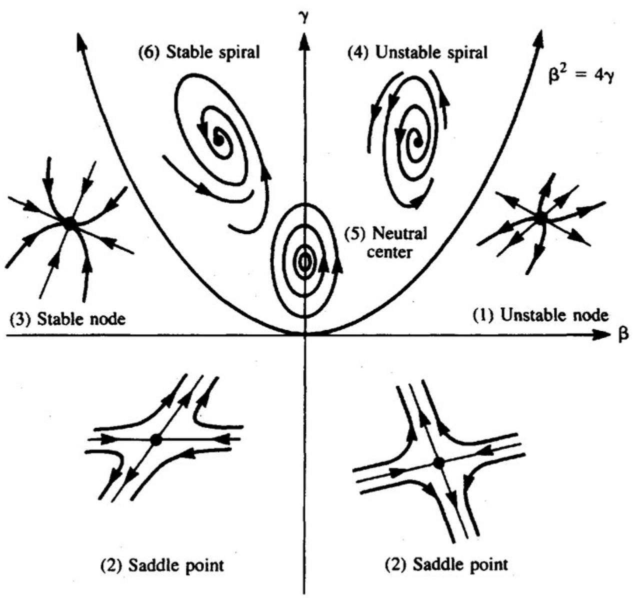
Approximated linear equations

Node, Focus, or Saddle point

Table 5.1 ***Linear Systems of two ODEs***

	Full algebraic notation	*Equivalent Vector–Matrix Notation*
Equations	$\frac{dx}{dt} = a_{11}x + a_{12}y$ $\frac{dy}{dt} = a_{21}x + a_{22}y$ $\quad y^{I}y^{S} = 2$	$\frac{d\mathbf{x}}{dt} = \mathbf{A}\mathbf{x}, \qquad \mathbf{A} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$
Significant quantities	$\beta = a_{11} + a_{22}$, $\gamma = a_{11}a_{22} - a_{12}a_{21}$, $\delta = \beta^2 - 4\gamma$	Tr $\mathbf{A}$, det $\mathbf{A}$, disc $\mathbf{A}$
Characteristic equation	$\lambda^2 - \beta\lambda + \gamma = 0$	$\det(\mathbf{A} - \lambda\mathbf{I}) = 0$
Eigenvalues	$\lambda_{1,2} = \frac{\beta \pm \sqrt{\delta}}{2}$	$\lambda_{1,2} = \frac{\mathrm{Tr}\,\mathbf{A} \pm \sqrt{\mathrm{disc}\,\mathbf{A}}}{2}$
Identities	$\lambda_1 + \lambda_2 = \beta$,	$\lambda_1 + \lambda_2 = \mathrm{Tr}\,\mathbf{A}, \qquad \lambda_1\lambda_2 = \det\mathbf{A}$
Eigenvectors	$\begin{pmatrix} a_{12} \\ \lambda_1 - a_{11} \end{pmatrix}, \begin{pmatrix} a_{12} \\ \lambda_2 - a_{11} \end{pmatrix}$	$\mathbf{v}_1, \mathbf{v}_2$ such that $(\mathbf{A} - \lambda\mathbf{I})\mathbf{v}_i = 0$
Solutions	$x = c_1 a_{12} e^{\lambda_1 t} + c_2 a_{12} e^{\lambda_2 t}$, $y = d_1 e^{\lambda_1 t} + d_2 e^{\lambda_2 t}$, where $d_1 = c_1(\lambda_1 - a_{11})$, $d_2 = c_2(\lambda_2 - a_{11})$.	$\mathbf{x} = c_1\mathbf{v}_1 e^{\lambda_1 t} + c_2\mathbf{v}_2 e^{\lambda_2 t}$.

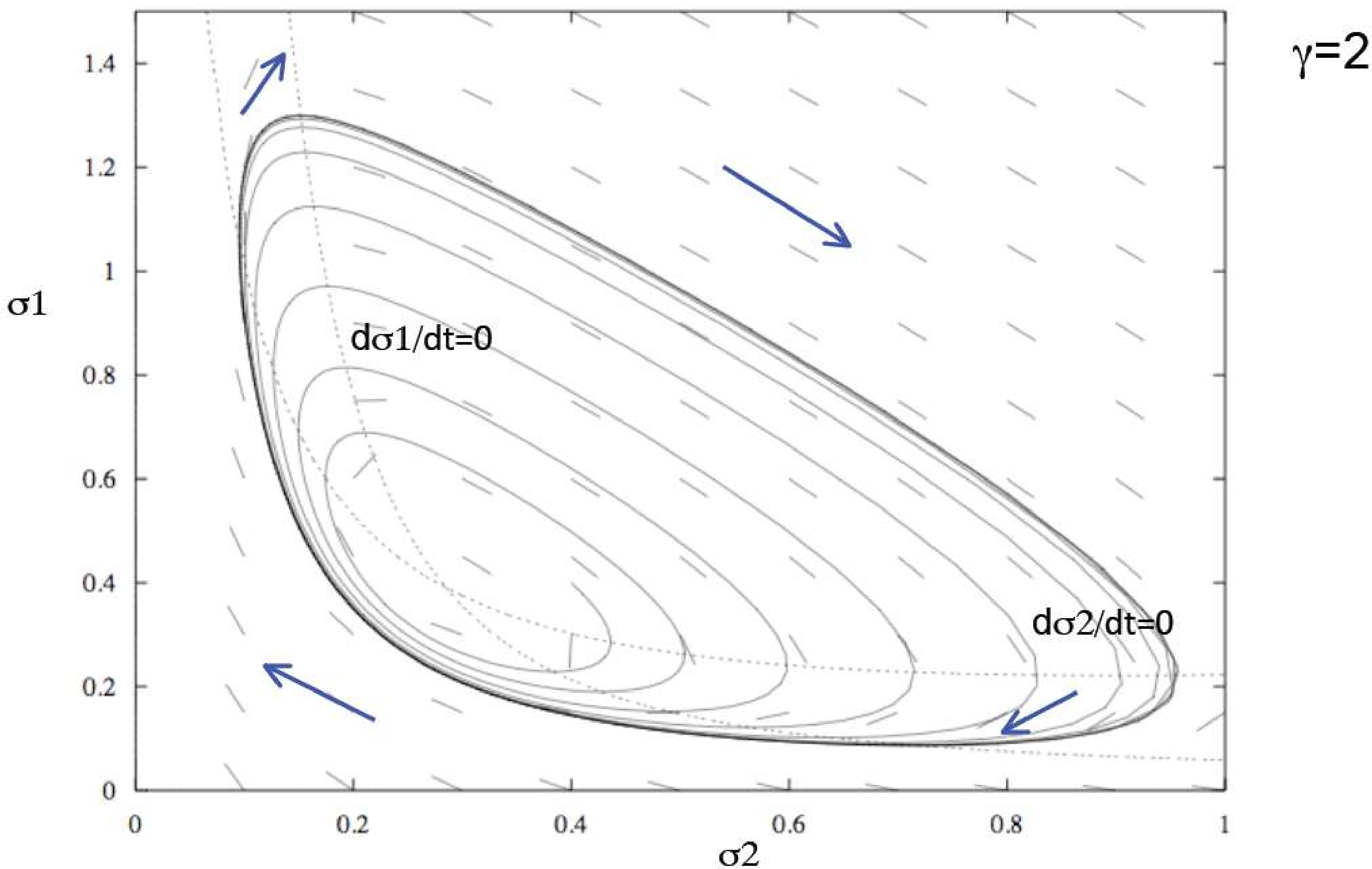
Node, Focus, or Saddle point



*focus = spiral

To summarize, the steady state can be classified into six cases as follows:

1. Unstable node: $\beta > 0$ and $\gamma > 0$.
2. Saddle point: $\gamma < 0$.
3. Stable node: $\beta < 0$ and $\gamma > 0$.
4. Unstable spiral: $\beta^2 < 4\gamma$ and $\beta > 0$.
5. Neutral center: $\beta^2 < 4\gamma$ and $\beta = 0$.
6. Stable spiral: $\beta^2 < 4\gamma$ and $\beta < 0$.

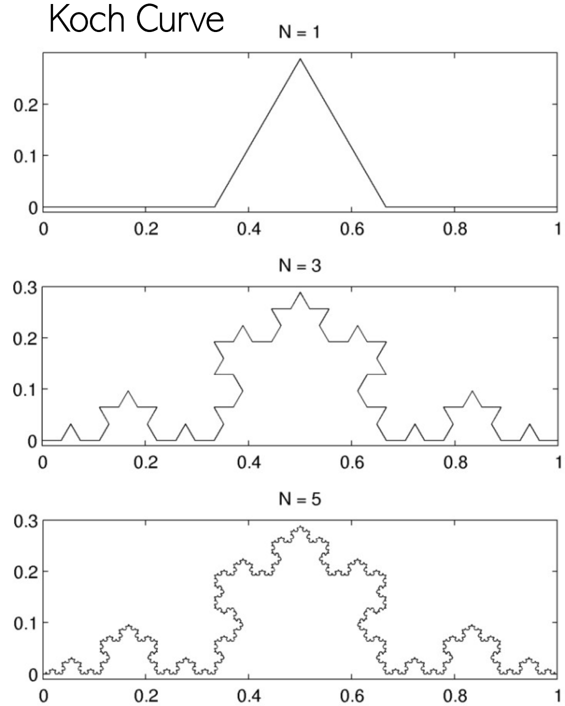

γ=2
σ1
σ2
dσ1/dt=0
dσ2/dt=0
0
0.2
0.4
0.6
0.8
1
1.2
1.4

More phase-space parameters: Dimensional Analysis

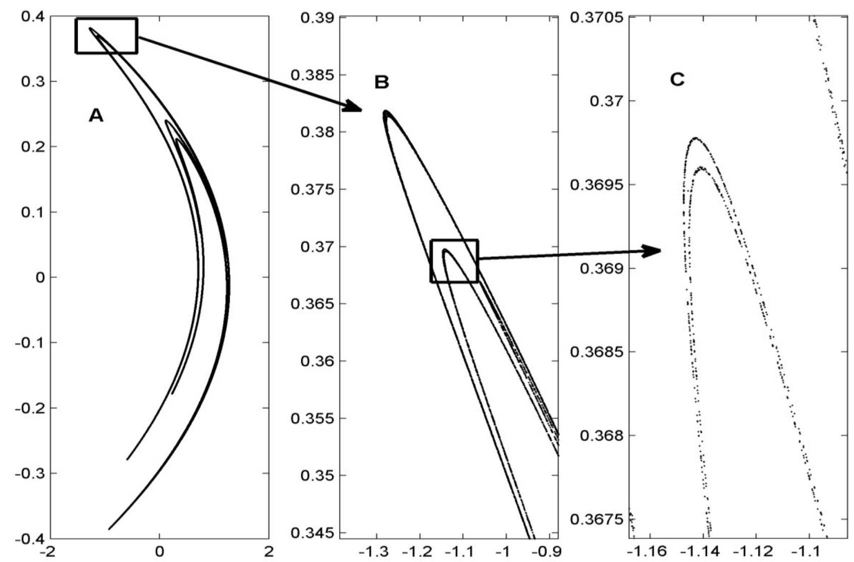
- Embedding dimension can only be integer values
  - For example, from PCA or False Nearest Neighbor analysis
- Chaotic attractors however can take on dimensions of fractional values
  - For example 2.05, 3.5
- Such objects are called fractals
  - Fractal objects have the property of self affinity
    - Their shapes are made of repeated patterns

Some Fractal Objects

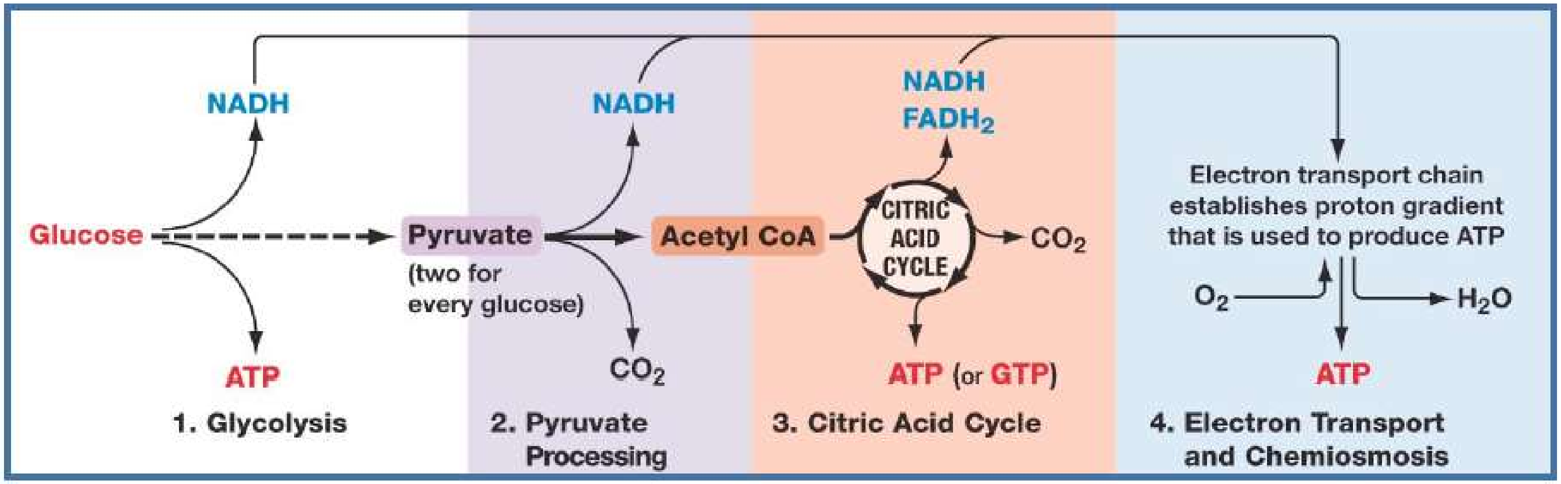
Koch Curve



Hénon Map

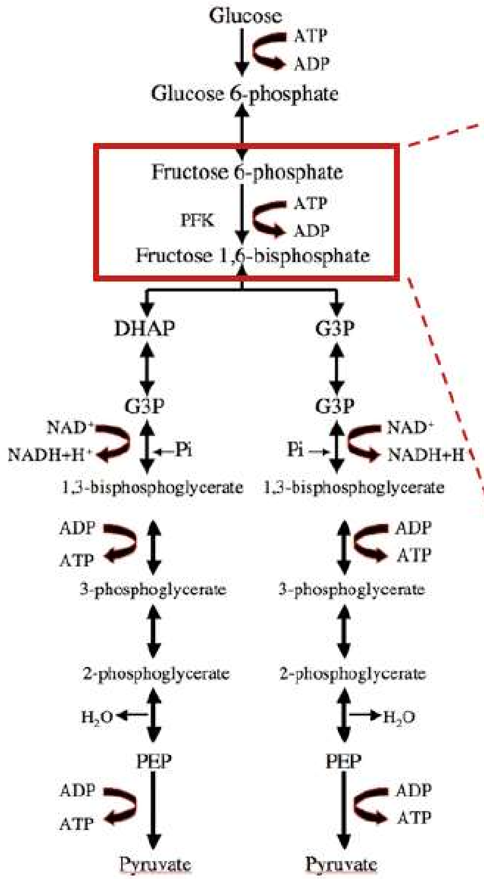


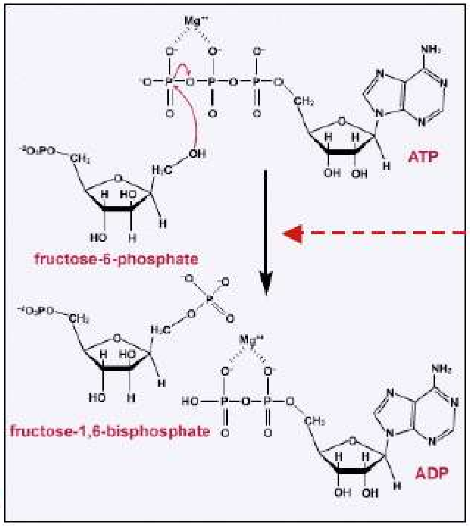
Glycolysis



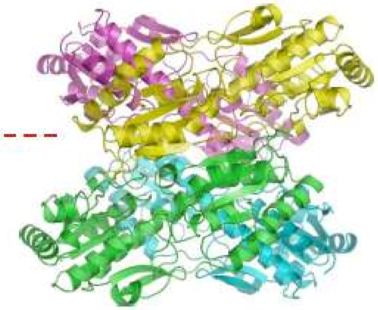
- Glycolysis literally means "splitting sugars."
- Glycolysis is the metabolic pathway that converts glucose into pyruvate.
- During glycolysis, two molecules of pyruvate are formed for every molecule of glucose.
- Pyruvate is then used in the Kreb cycle.
- Glycolysis also yields to 2 molecules of ATP and 2 molecules of NADH.
- Glycolysis takes place in the cytoplasm.

Glycolysis & PFK

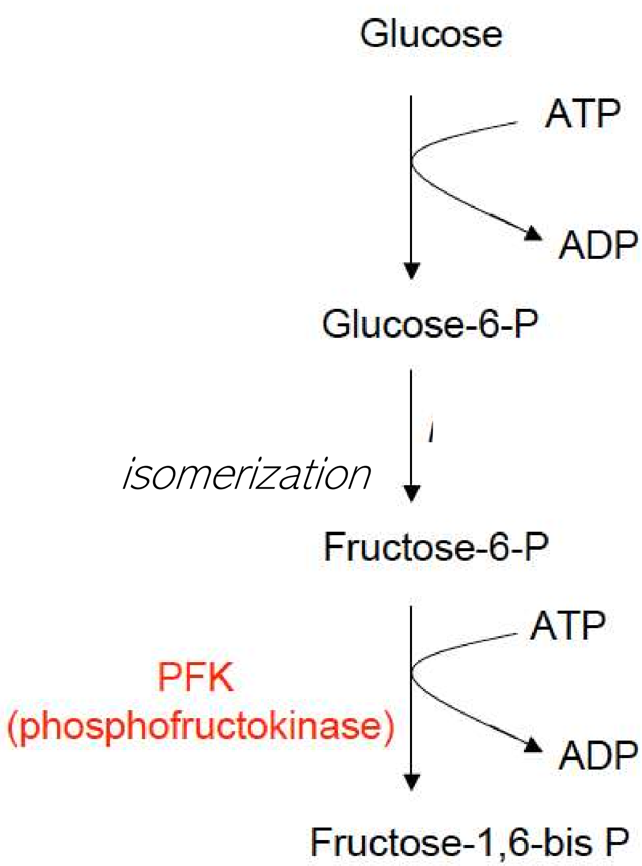


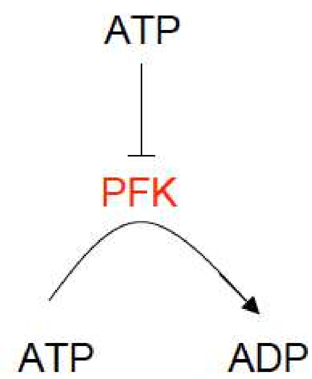


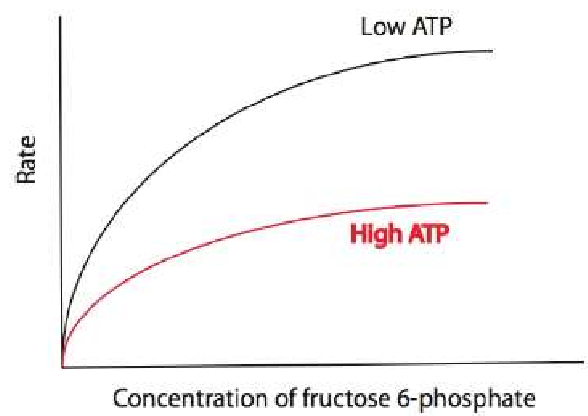
phosphofructokinase (PFK)



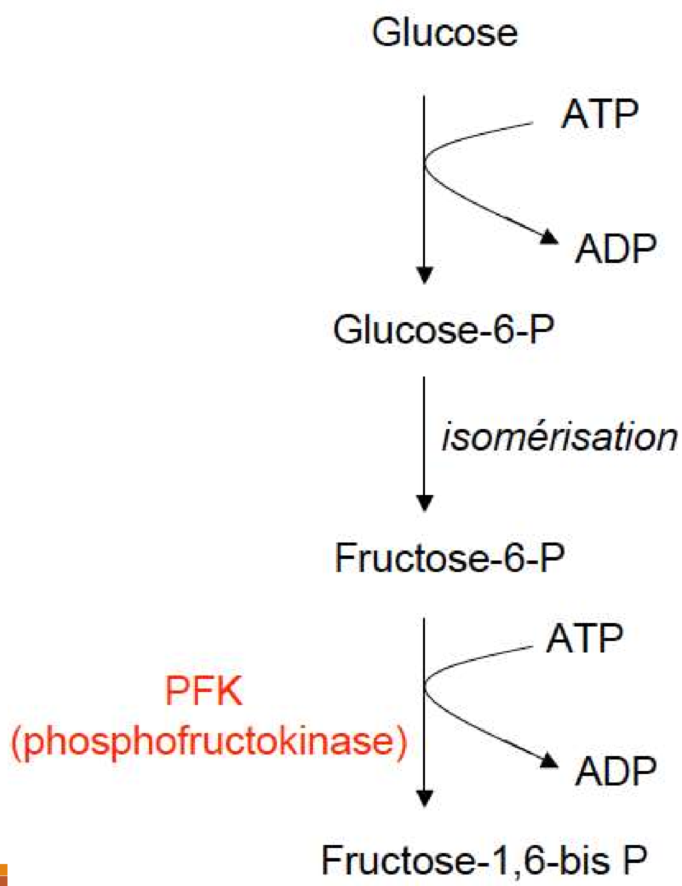
Glycolysis & PFK

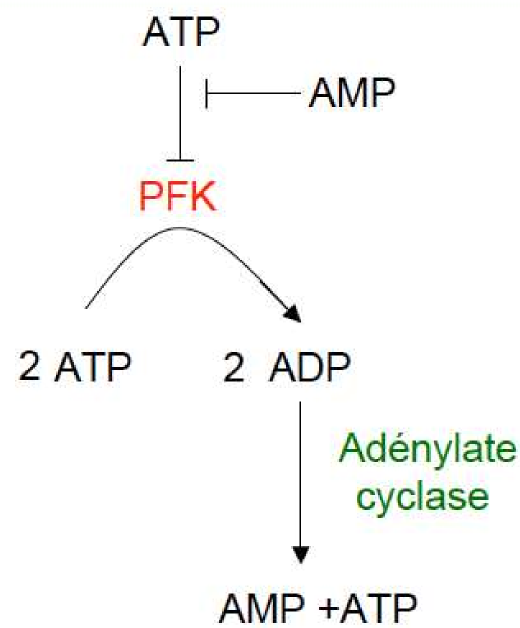






Glycolysis & PFK





Allosterically inhibited by ATP and allosterically activated by AMP (indicating cell's energetic needs). So, if PFK is active the ratio of [ATP]/[AMP] decreases.

Slide courtesy Dr. Didier Gonze. Université Libre de Bruxelles, Belgium

Glycolysis Example

Consider: the nonlinear, coupled system proposed by Sel'kov describing glycolysis at the rate limiting phosphofructokinase (PFK) step:

$$\frac{\partial x}{\partial t} = -x + ay + x^2 y$$

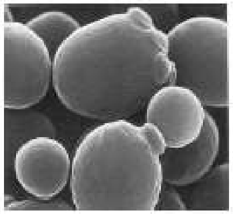
$$\frac{\partial y}{\partial t} = b - ay - x^2 y$$

- a and b are rate constants, x=[ADP] and y=[F6P]

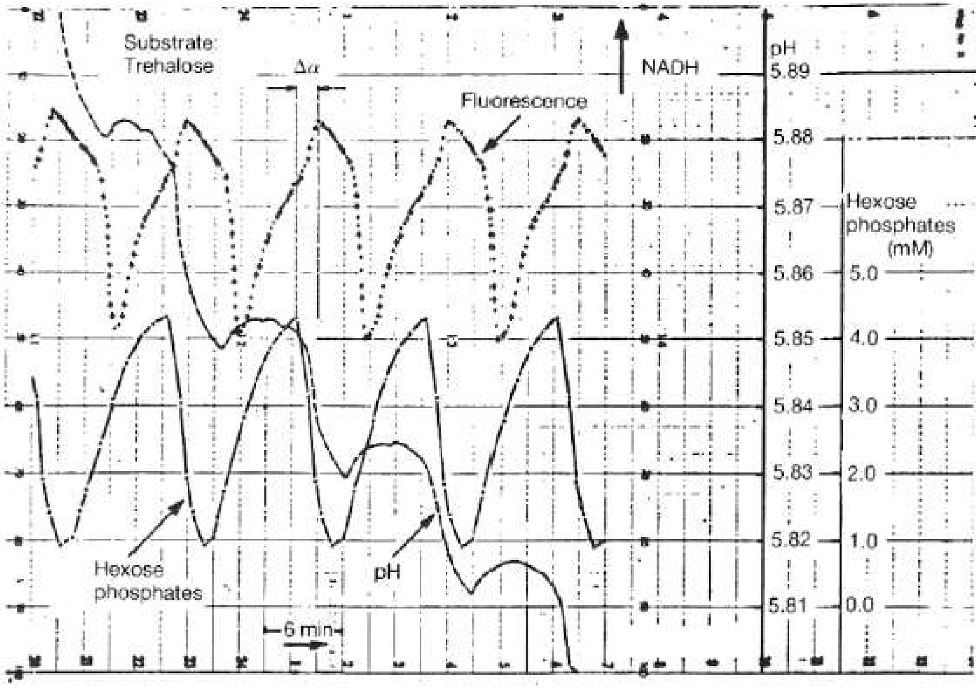
- there is a non-linear term (x^2y) but this is not chaotic (2 degrees of freedom- ADP and F6P). Also time is implicit in these differential equations.

If explicit time dependence was included in this system (e.g. change b such that $b=b_0\sin(\omega t)$ then there would be 3 degrees of freedom including time, and chaos would be possible

Glycolytic oscillations in *Saccharomyces cerevisiae*



S. cerevisiae
(yeast)



Glycolytic oscillations in a yeast extract subjected to constant injection of the substrate (trehalose). Chemical analyses show that the various hexoses oscillate with the same frequency as NADH.

Hess & Boiteux (1968) In Regulatory Functions of Biological Membranes. Ed. J. Jarnefelt, Elsevier.
Hess B, Boiteux A, Krüger J (1969) Cooperation of glycolytic enzymes. Adv Enzyme Regul. 7:149-67

Slide courtesy Dr. Didier Gonze. Université Libre de Bruxelles, Belgium

Ranges of Glycolytic Oscillation in Yeast Extract

Input rate* mM/hr	Period min	Amplitude in mM NADH	Damping	Waveform
< 20	—	steady high level of NADH	—	—
20	8.6	0.2–0.4	—	double periodicities, nonsinusoidal
40	6.5	0.6	—	nonsinus-sinus
60–80	5.0	0.3	—	stabile sinus
120	3.5	0.2	—	stabile sinus
> 160	—	steady low level of NADH	+++	

* Fructose or glucose serve as substrates. Cell-free extract of ~60 mg/ml.

Hess B, Boiteux A, Krüger J (1969) Cooperation of glycolytic enzymes. Adv Enzyme Regul. 7:149-67

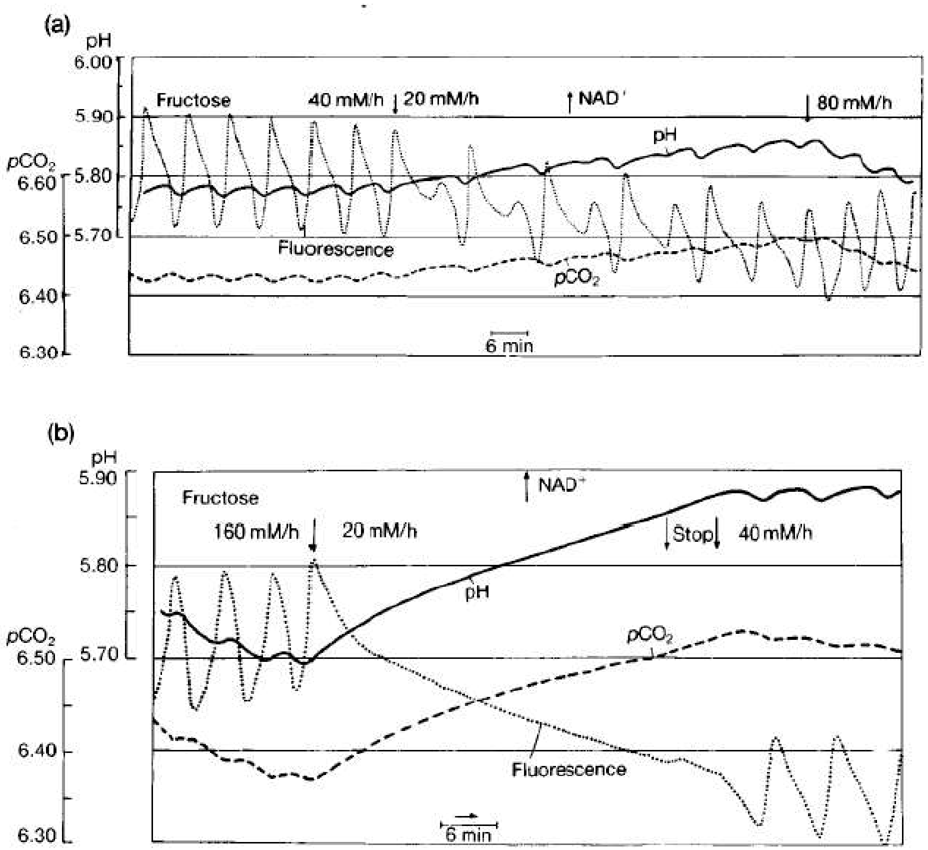


Fig. 2.4. Control of glycolytic oscillations in yeast extracts by the substrate injection rate. (a) The diminution of the rate of injection of fructose from 40 to 20 mM/h causes a lengthening of the period as well as a change in the waveform of oscillations; this change is reversible. (b) Decreasing the injection rate below 20 mM/h causes the reversible suppression of the oscillations (Hess & Boiteux, 1968b).

Control of glycolytic oscillations by the substrate injection rate:

(a) The diminution of the injection rate causes a lengthening of the period

(b) Decreasing the injection rate below a certain threshold causes the reversible suppression of the oscillations

Hess B, Boiteux A (1968) Hoppe Seylers Z Physiol Chem. 349:1567-74.

PFK is responsible for the glycolytic oscillations

1) If glucose-6-phosphate or fructose-6-phosphate is taken as substrate, oscillations are still observed.

2) If fructose 1,6-bis-phosphate is used as substrate, there are no oscillations.

3) NH_4+ (actives PFK) inhibits oscillations

4) Citrate (inhibits PFK) inhibits oscillations

5) Amplitude and frequency of oscillations can be varied by adding purified PFK to the cultured cells.

Hypothesis: activity of <u>PFK</u> is stimulated by <u>1 or several</u> <u>ADP</u> molecules

$E \qquad \gamma \qquad S_2$

$$\gamma S_2 + E \underset{k_{-3}}{\overset{k_3}{\rightleftharpoons}} ES_2^{\gamma}$$

Inactive

Active

Sel'kov (1968) Self-oscillations in Glycolysis, *Eur J Biochem* 4: 79-86.

Evolution equations (s_1 = [ATP] ; s_2 = [ADP] ; e = [E] ; x_1 = [$ES_2{}^\gamma$] ; x_2 = [$S_1ES_2{}^\gamma$])

$$\begin{cases} \dfrac{ds_1}{dt} = v_1 - k_1 s_1 x_1 + k_{-1} x_2 \\ \dfrac{ds_2}{dt} = k_2 x_2 - k_3 s_2^\gamma e + k_{-3} x_1 - v_2 s_2 \\ \dfrac{dx_1}{dt} = -k_1 s_1 x_1 + (k_{-1} + k_2) x_2 + k_3 s_2^\gamma e - k_{-3} x_1 \\ \dfrac{dx_2}{dt} = k_1 s_1 x_1 - (k_{-1} + k_2) x_2 \end{cases}$$

with:

$$e_0 = e + x_1 + x_2$$

(total concentration in PFK)

$$u_1 = \frac{x_1}{e_0} \qquad u_2 = \frac{x_2}{e_0} \qquad \sigma_1 = \frac{k_1 s_1}{k_{-1} + k_2} \qquad \sigma_2 = \left(\frac{k_3}{k_{-3}}\right)^{1/\gamma} s_2$$

$$\tau = \frac{e_0 k_1 k_2}{k_{-1} + k_2} t \qquad u_1 + u_2 + \frac{e}{e_0} = 1$$ (enzyme conservation)

$$\left\{ \begin{array}{rcl} \dfrac{d\sigma_1}{dt} & = & v - \dfrac{k_{-1} + k_2}{k_2} u_1 \sigma_1 + \dfrac{k_{-1}}{k_2} u_2 \\ \dfrac{d\sigma_2}{dt} & = & \alpha \left(u_2 - \dfrac{k_{-3}}{k_2} \sigma_2^{\gamma} (1 - u_1 - u_2) + \dfrac{k_{-3}}{k_2} u_1 \right) - \eta \sigma_2 \\ \epsilon \dfrac{du_1}{dt} & = & u_2 - \sigma_1 u_1 + \dfrac{k_{-3}}{k_{-1} + k_2} \left(\sigma_2^{\gamma} (1 - u_1 - u_2) - u_1 \right) \\ \epsilon \dfrac{du_2}{dt} & = & \sigma_1 u_1 - u_2 \end{array} \right.$$

where $\epsilon = \dfrac{e_0 k_1 k_2}{(k_2 + k_{-1})^2} \qquad v = \dfrac{v_1}{k_2 e_0} \qquad \eta = \dfrac{v_2 (k_{-1} + k_2)}{e_0 k_1 k_2} \qquad \alpha = \dfrac{k_{-1} + k_2}{k_1} \left(\dfrac{k_3}{k_{-3}} \right)^{1/\gamma}$

Analysis of the 2-equation system (for σ_1 et σ_2) in the phase space

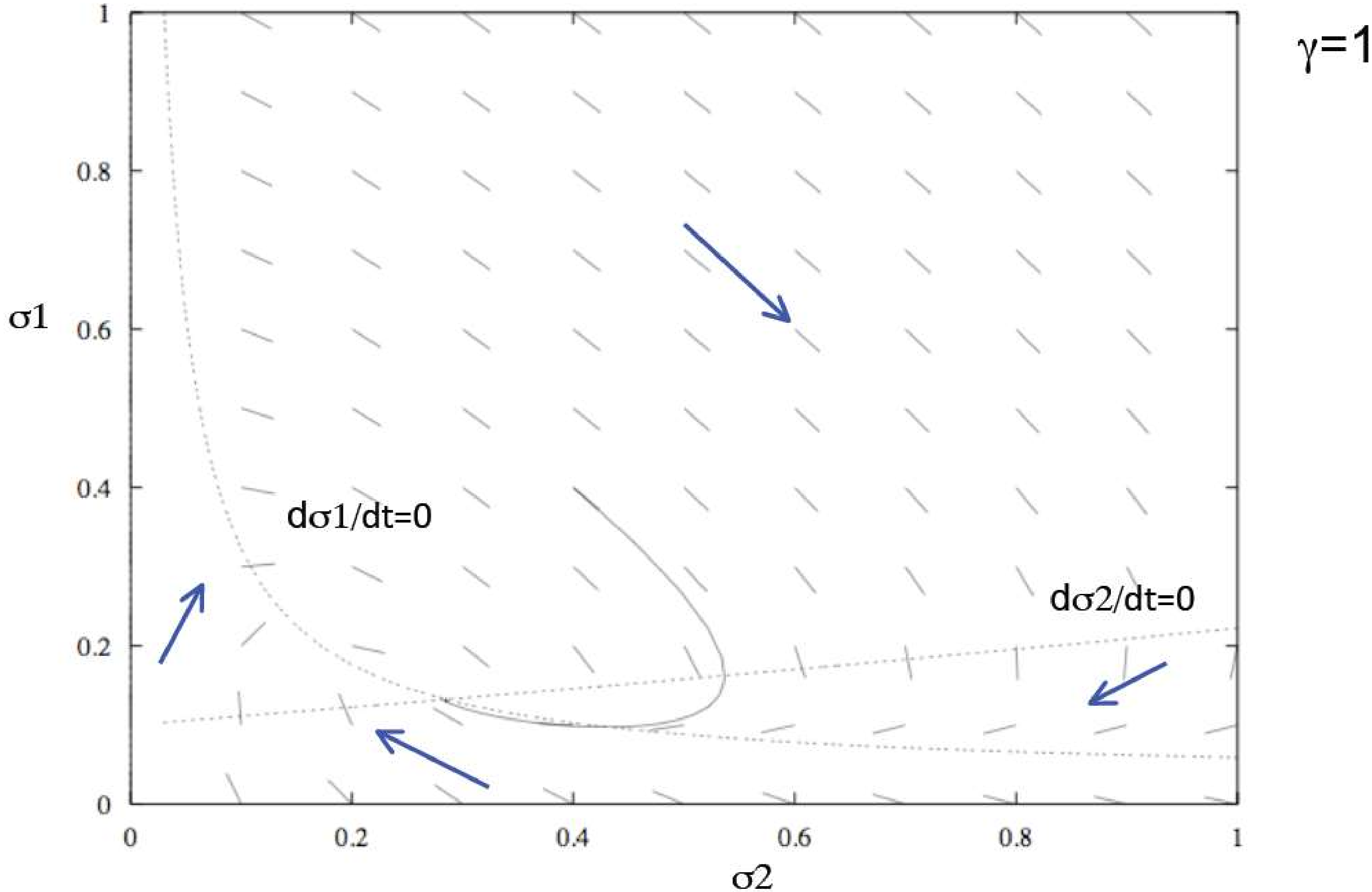
Nullclines:

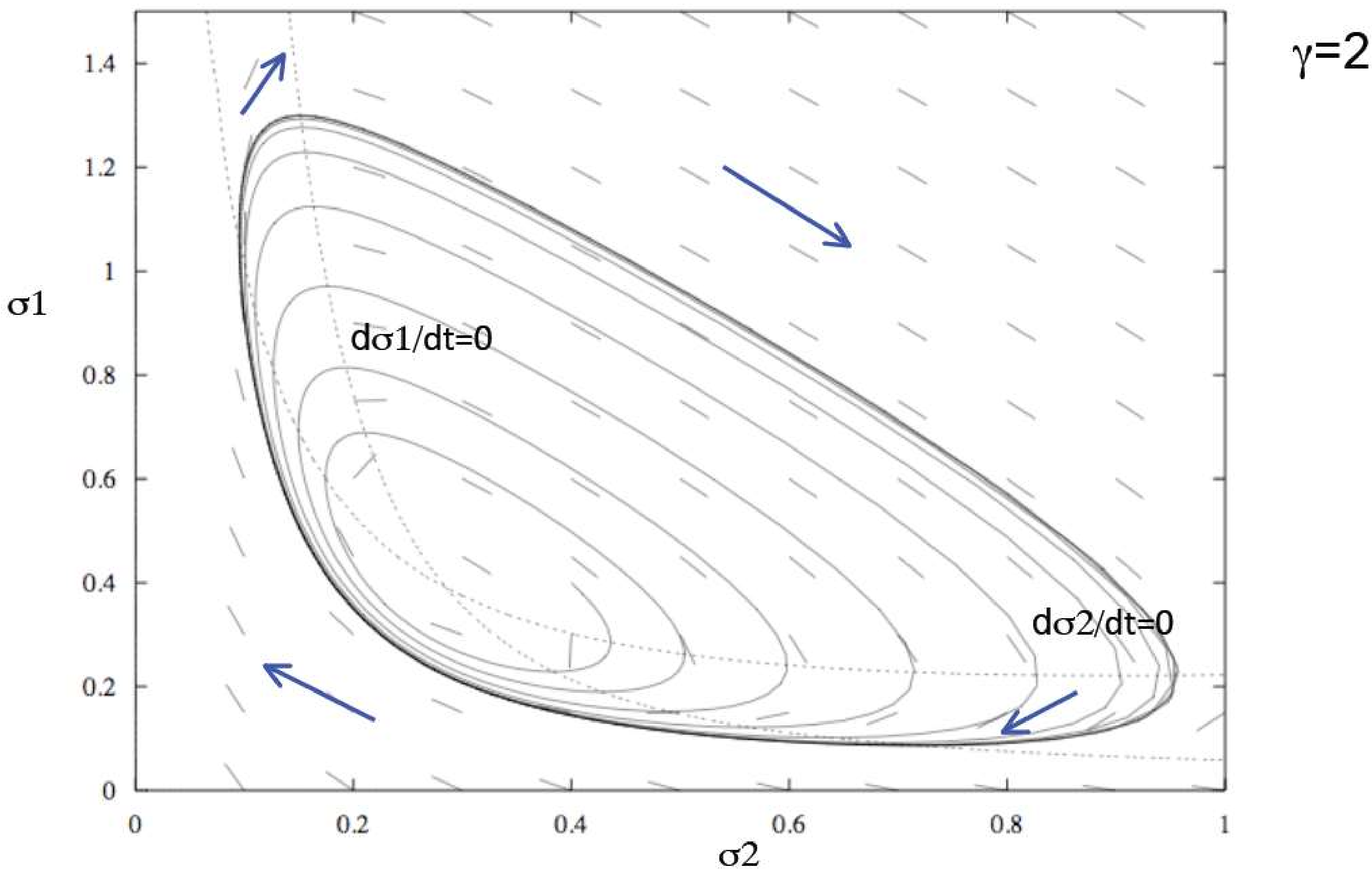
(1) $v = f(\sigma_1, \sigma_2)$ Hence: $\sigma_1 = \dfrac{v}{1-v}\dfrac{1+\sigma_2^\gamma}{\sigma_2^\gamma}$

(2) $\alpha f(\sigma_1, \sigma_2) = \eta\sigma_2$ Hence: $\sigma_1 = \dfrac{1+\sigma_2^\gamma}{\sigma_2^{\gamma-1}(p-\sigma_2)}$ where $p = \dfrac{\alpha}{\eta}$

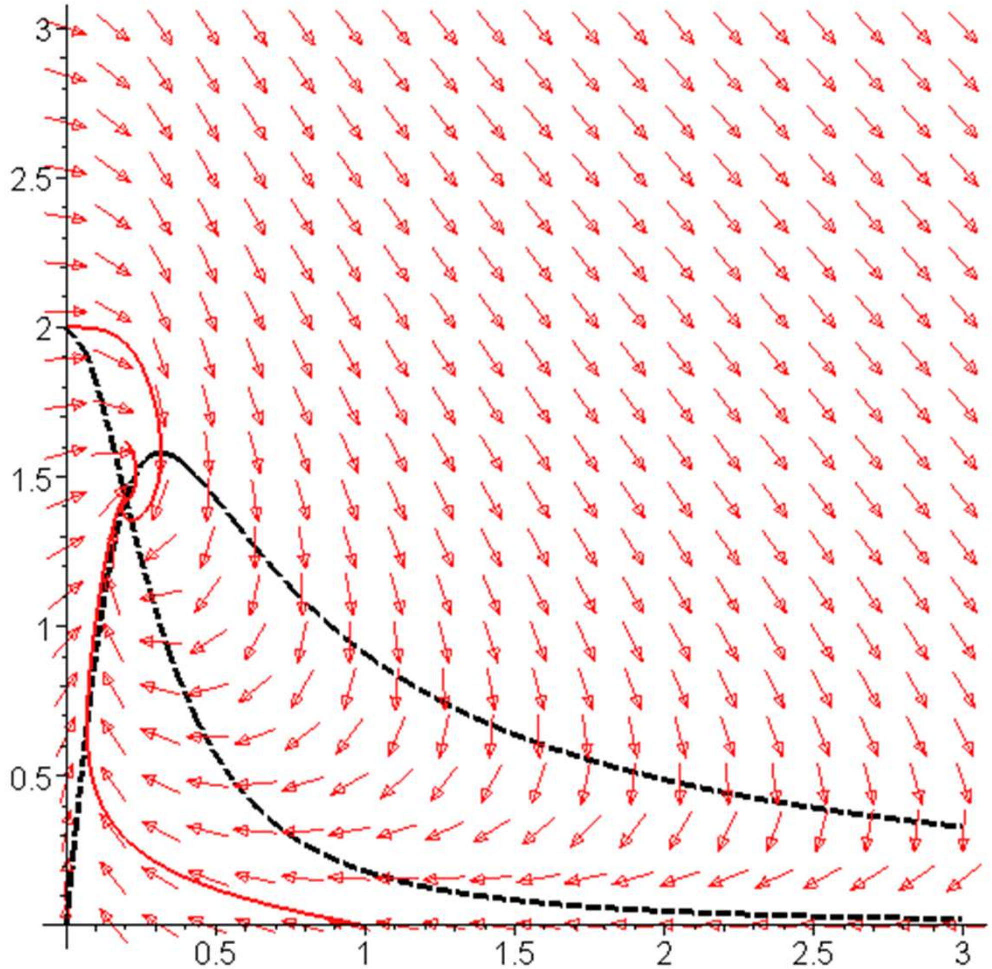
The intersection of the nullclines defines the steady state: $\sigma_1^{SS}, \sigma_2^{SS}$

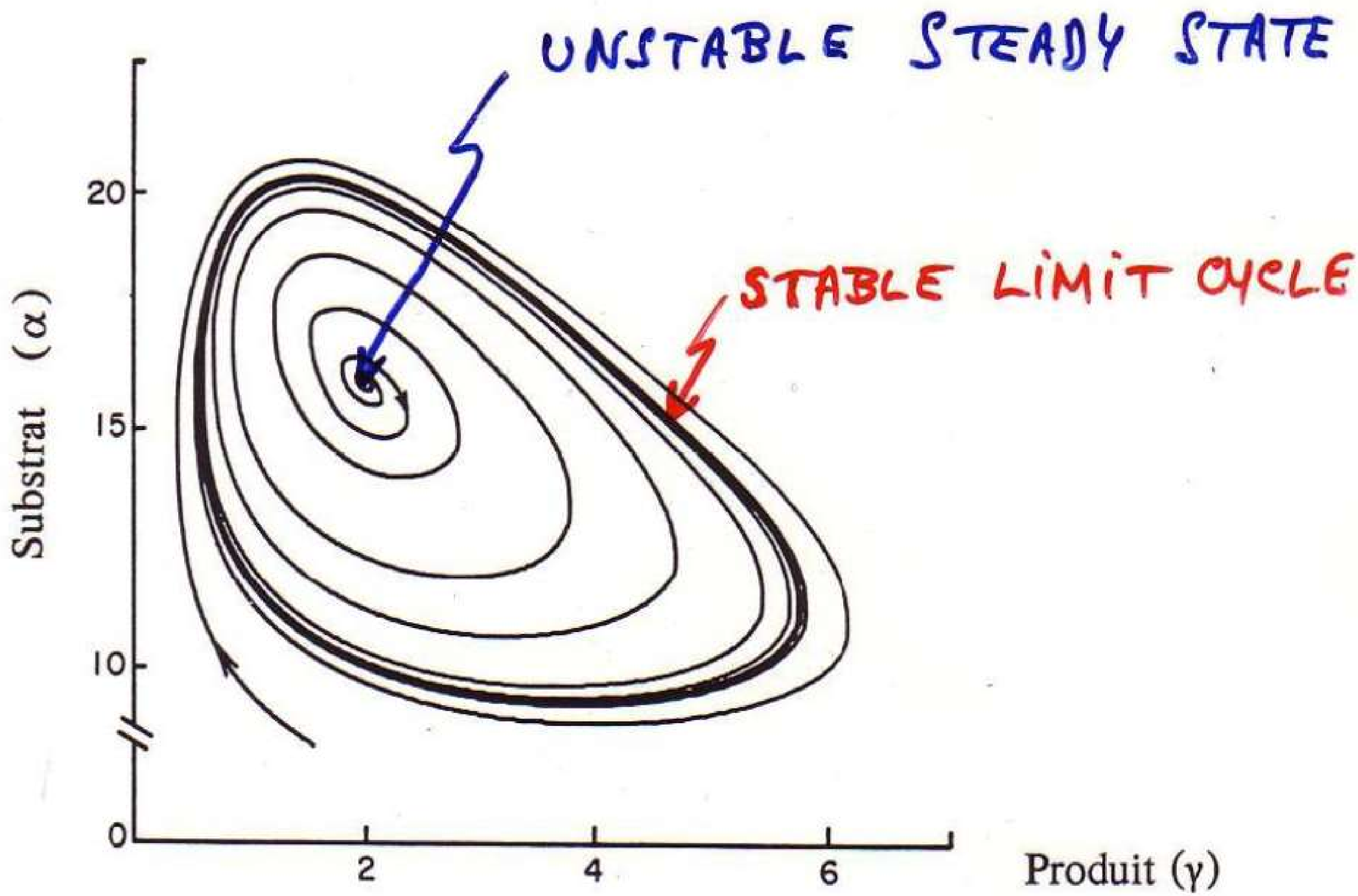
recall γ is number of ADP molecules over the reaction

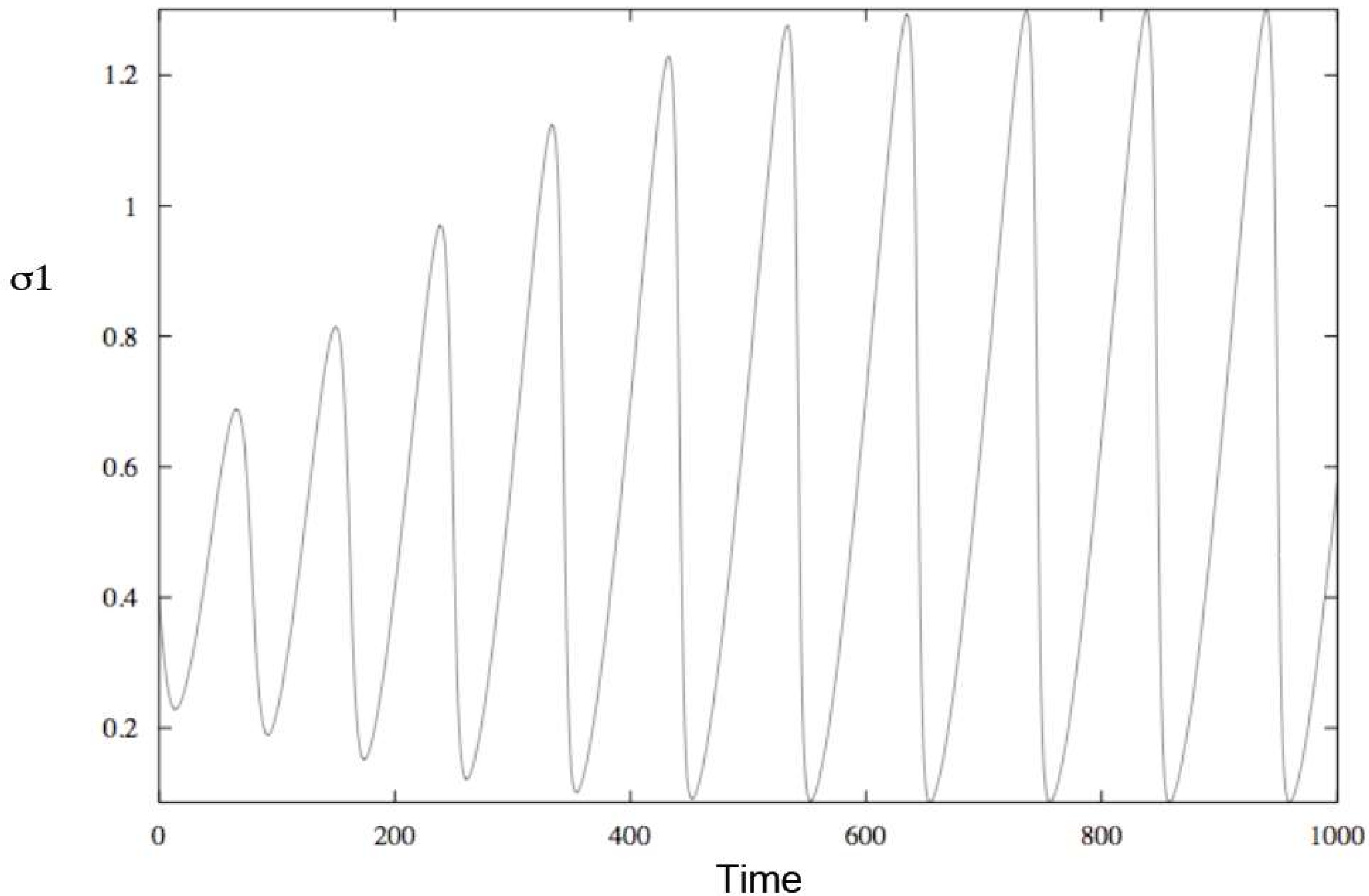


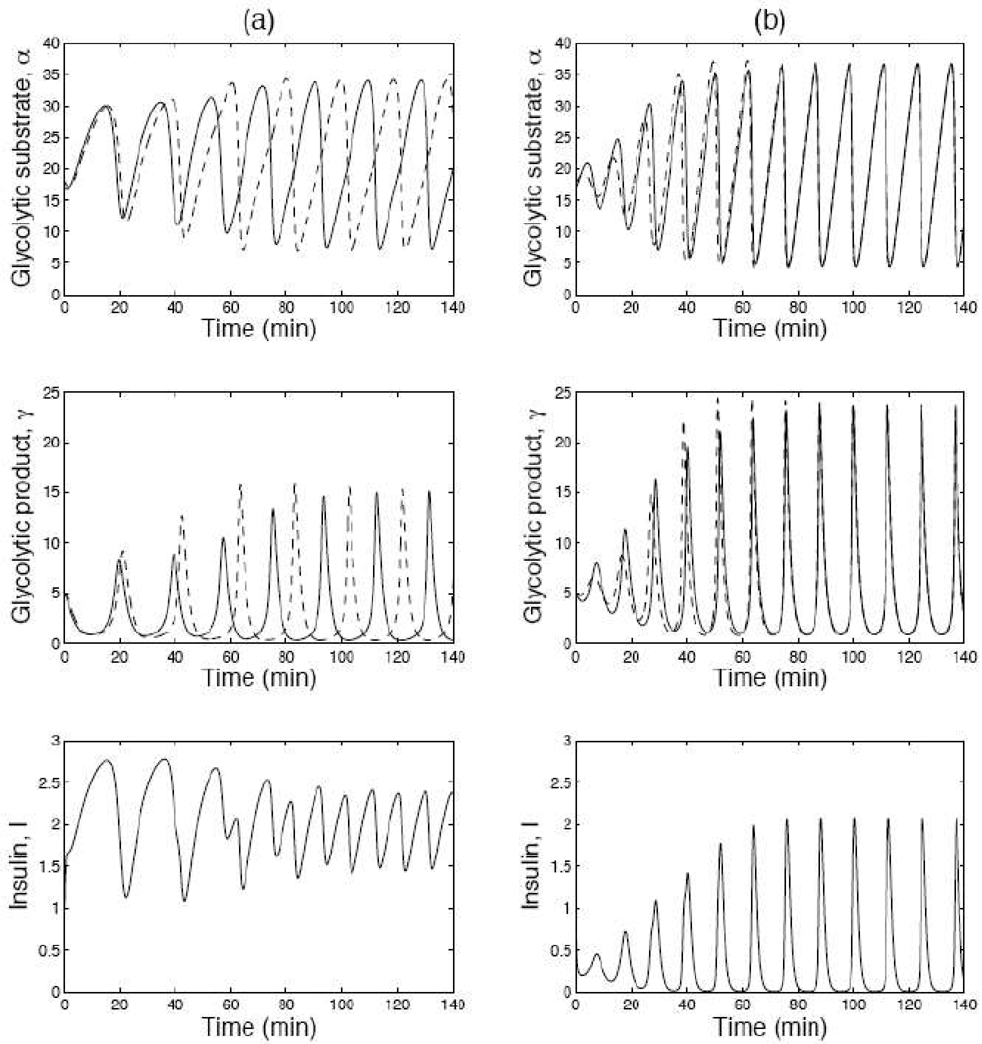

γ=2
σ1
σ2
dσ1/dt=0
dσ2/dt=0
1.4
1.2
1
0.8
0.6
0.4
0.2
0
0.2
0.4
0.6
0.8
1

a = .1;   b = .2




UNSTABLE STEADY STATE
STABLE LIMIT CYCLE
20
15
10
0
2
4
6
Substrat (α)
Produit (γ)


σ1
1.2
1
0.8
0.6
0.4
0.2
0
200
400
600
800
1000
Time

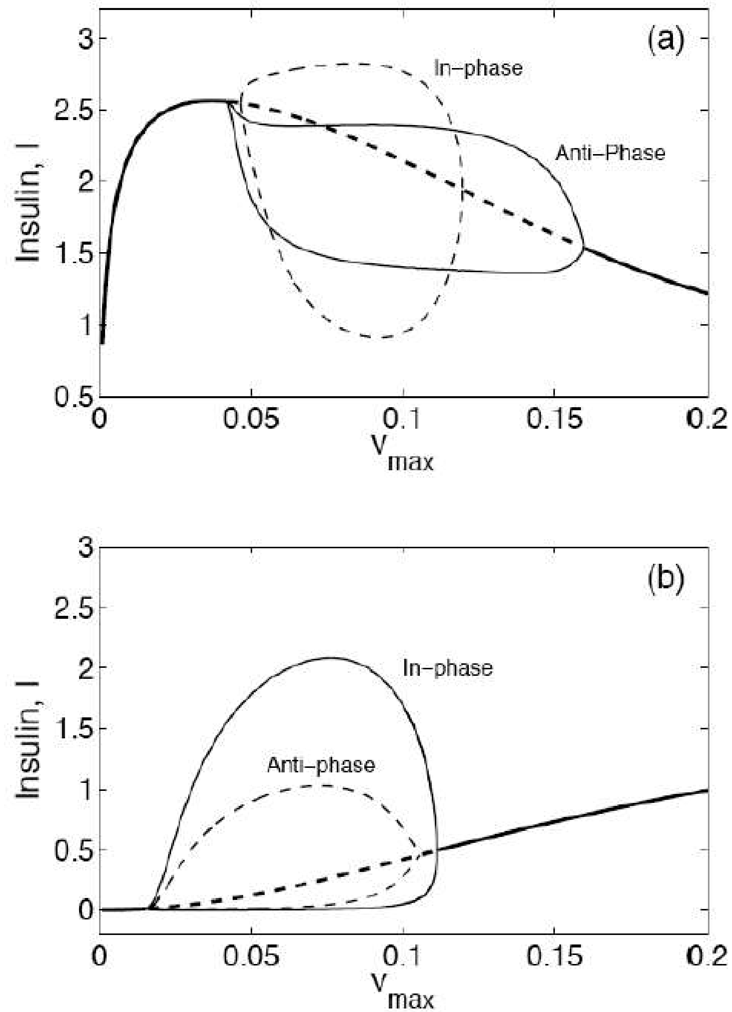


The mode of synchronization depends on the way the two oscillators are coupled:

(a) When insulin release is controlled by the glycolytic **substrate** (α), the oscillations are in anti-phase

(b) When insulin release is controlled by the glycolytic **product** (γ), the oscillations are in phase.

Slide courtesy Dr. Didier Gonze. Université Libre de Bruxelles, Belgium



Bifurcation diagram as a function of the maximum rate of glucose input (v_{max}) into the cell.

(a) When insulin release is controlled by the glycolytic **substrate**, the stable limit cycle regime corresponds to anti-phase synchronization, while the unstable limit cycle regime corresponds to in-phase oscillations.

(b) When insulin release is controlled by the glycolytic **product**, the stable limit cycle regime corresponds to in-phase synchronization and the unstable limit cycle regime corresponds to antiphase oscillations.

Slide courtesy Dr. Didier Gonze. Université Libre de Bruxelles, Belgium

Other Information:

Check out the Glycolysis Simulator by Dr. Dr. Bernhard Palsson
http://gcrg.ucsd.edu/sites/default/files/Attachments/Images/publications/books/systemsBiology1/MetlabNotebooks/glycolysis.zip

https://www.youtube.com/watch?v=QgPvBLFQ_xU

Conclusions

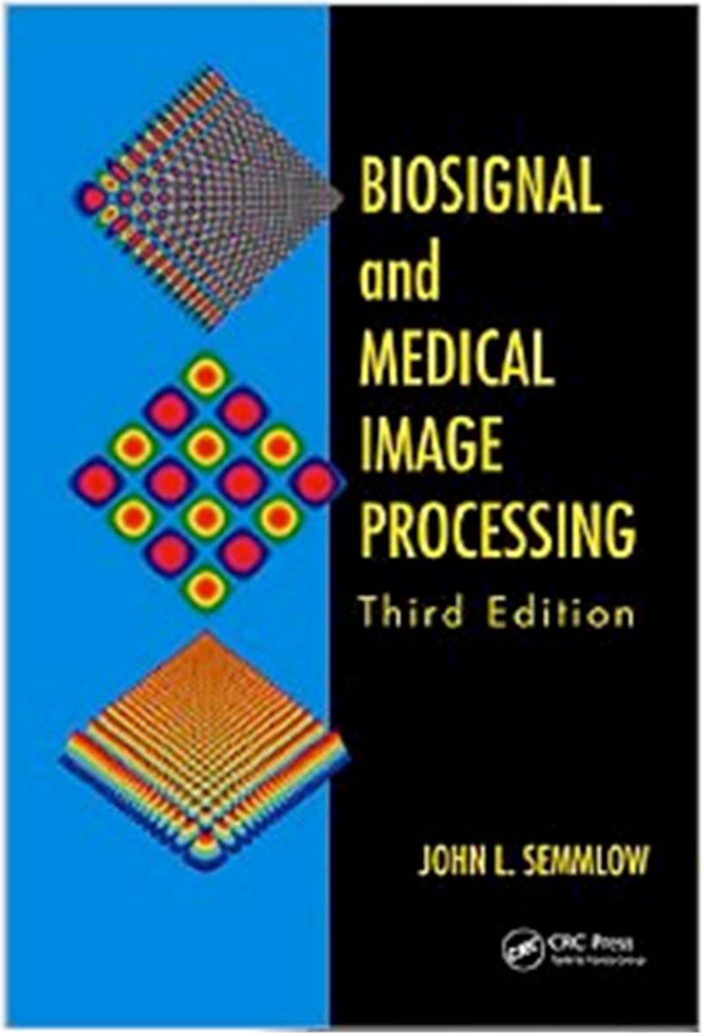
Nonlinear analysis

- Gives more information about nonlinear signals than linear analysis alone
- Has to be used carefully because it can take a long time and give misleading results

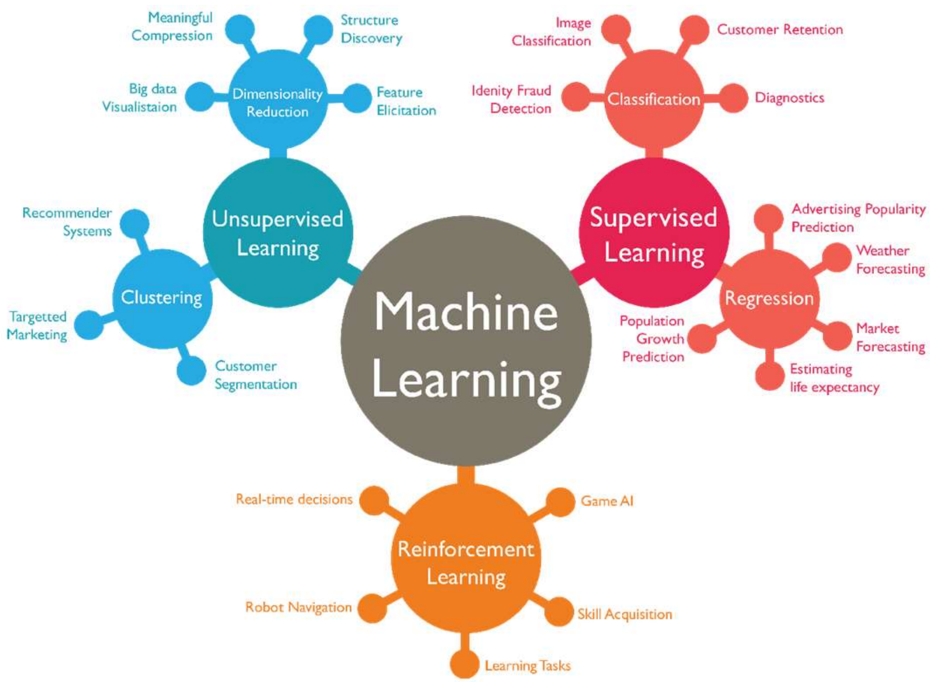
Surrogate data testing is needed to establish nonlinearity

- Otherwise nonlinear testing could be useful for distinguishing between signal types, but won't be advantageous

Note: All materials, code and data for the following materials are from Biosignal and Medical Image processing 3rd Edition



Machine Learning



https://www.wordstream.com/blog/ws/2017/07/28/machine-learning-applications

Statistical Learning

- Machine learning is the general term but if a model uses statistical methods to achieve its goal it is statistical learning

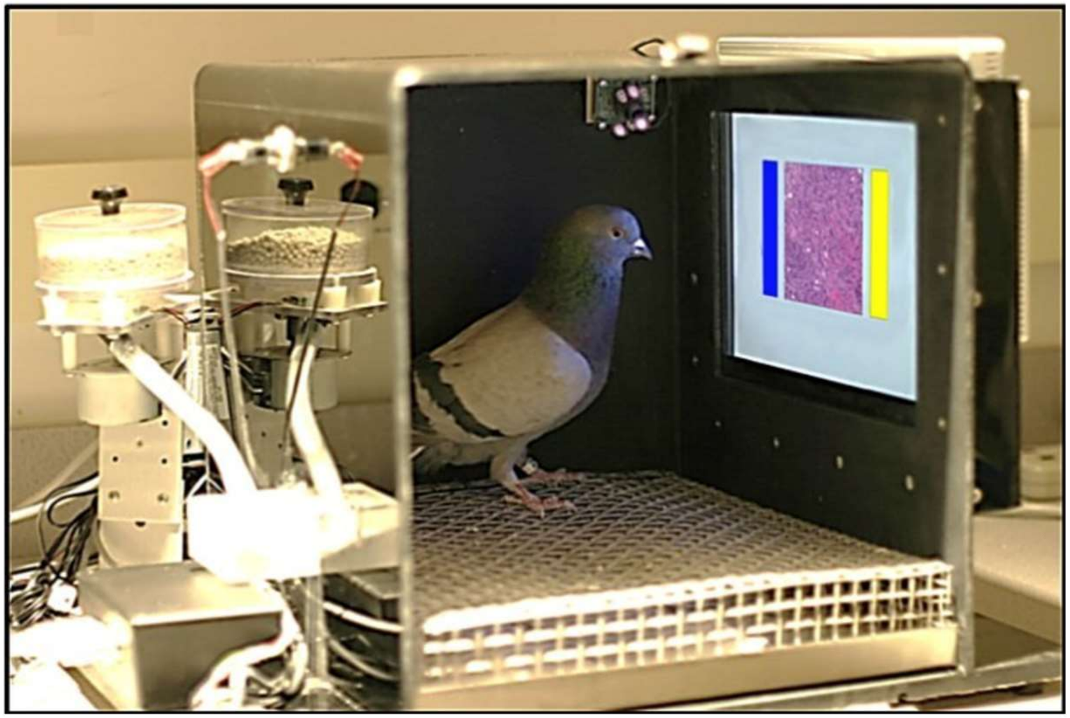
Goal

- build a model that makes predictions based on evidence in the presence of uncertainty.

Paging Dr. Pigeon; You're Needed in Radiology

By **NICHOLAS BAKALAR** NOV. 24, 2015

The New York Times



The pigeons' training environment at the University of Iowa included a food pellet dispenser, a touch-sensitive screen that projected medical images, and blue and yellow choice buttons on either side.
University of Iowa/Wassermann Lab

Examples

Consider machine learning for a complex task or problem involving a large amount of data and/or lots of variables, but no existing formula or equation.

For example:

1) face recognition and speech recognition.

2) When rules of a task are constantly changing—as in fraud detection from transaction records.

3) If the nature of the data keeps changing, and the program needs to adapt such as for predicting shopping trends.

Real World Applications

- Finance
  - credit scoring
  - Predicting stock trends
  - Targeted ads & suggestions
  - Fraud Detection
- Image processing and computer vision
  - facial recognition
  - motion detection
  - object detection
- Computational biology
  - tumor detection
  - drug discovery
  - DNA sequencing
- Energy production
  - price
  - load forecasting
- Automotive, aerospace, and manufacturing
  - predictive maintenance
  - Autonomy
- Natural language processing
  - Google Home/Amazon Alexa
  - Automated phone systems

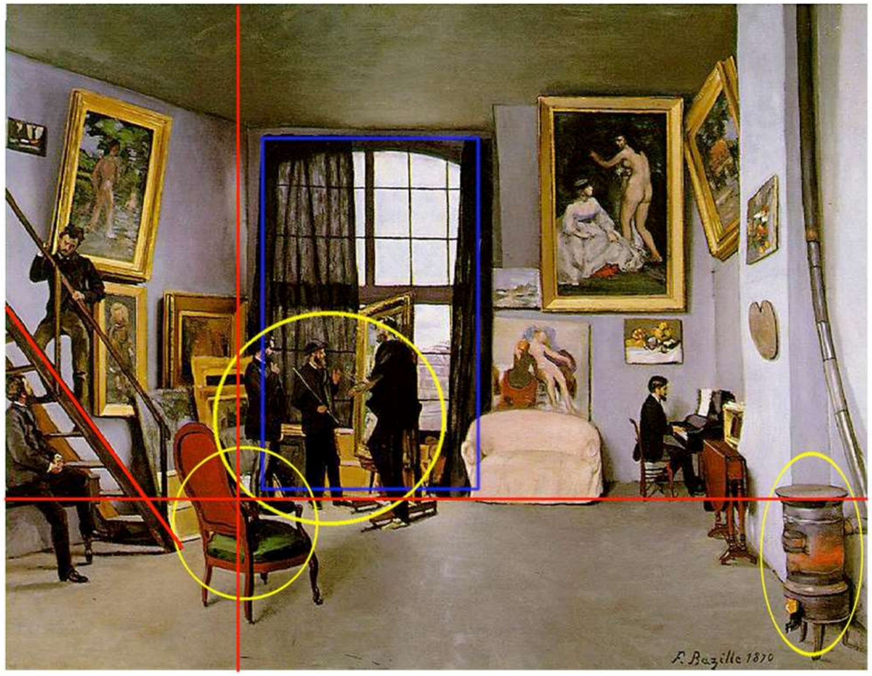
Creating Algorithms that Can Analyze Works of Art

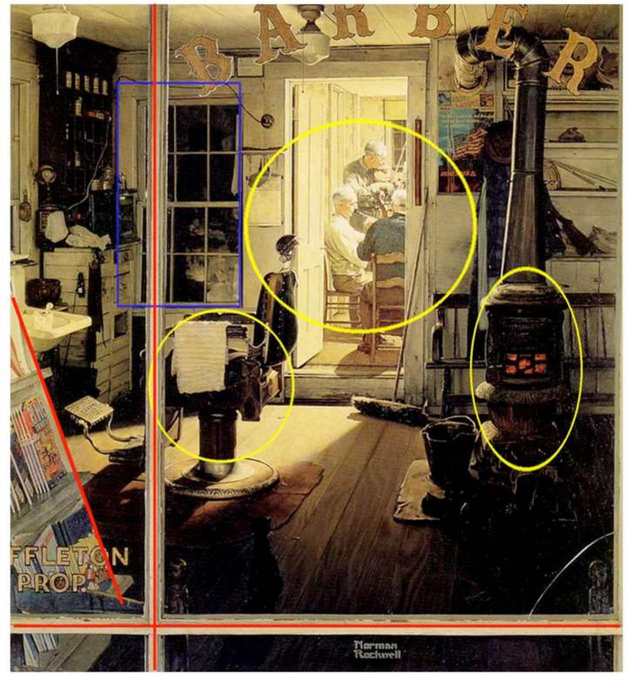
- Art and Artificial Intelligence Laboratory (Rutgers University)
- used machine learning to classify paintings by style, genre, and artist.
- Algorithms they developed classified the styles of paintings in large database with 60% accuracy, outperforming typical non-expert humans.
- tested >1,700 paintings from 66 different artists working over a span of 550 years. - machine learning readily identified connected works

Diego Velazquez's
Portrait of Pope Innocent X"

Francis Bacon's
"Study After
Velazquez's Portrait of Pope Innocent X.

https://sites.google.com/site/digihumanlab/research/artistic-influence

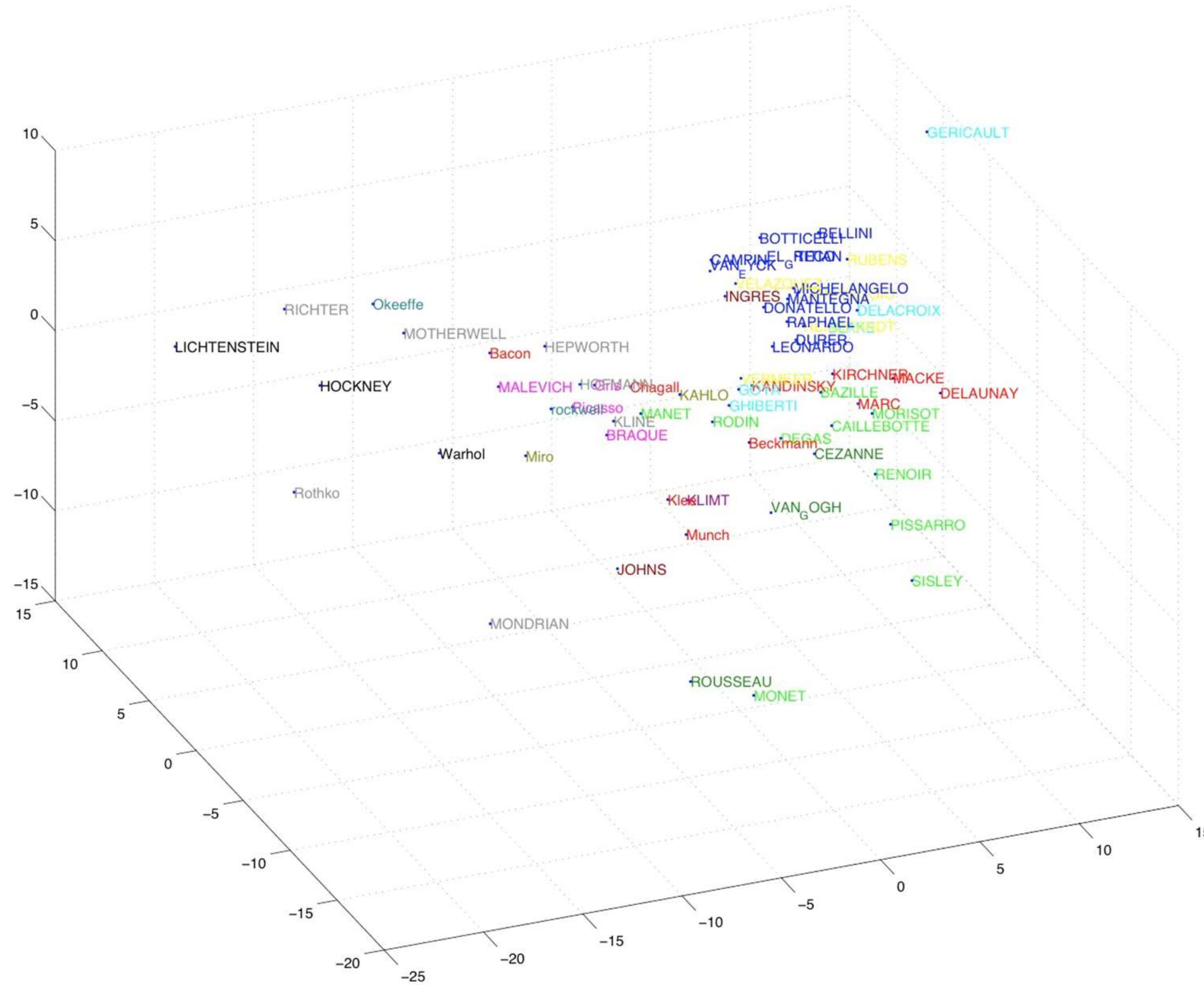




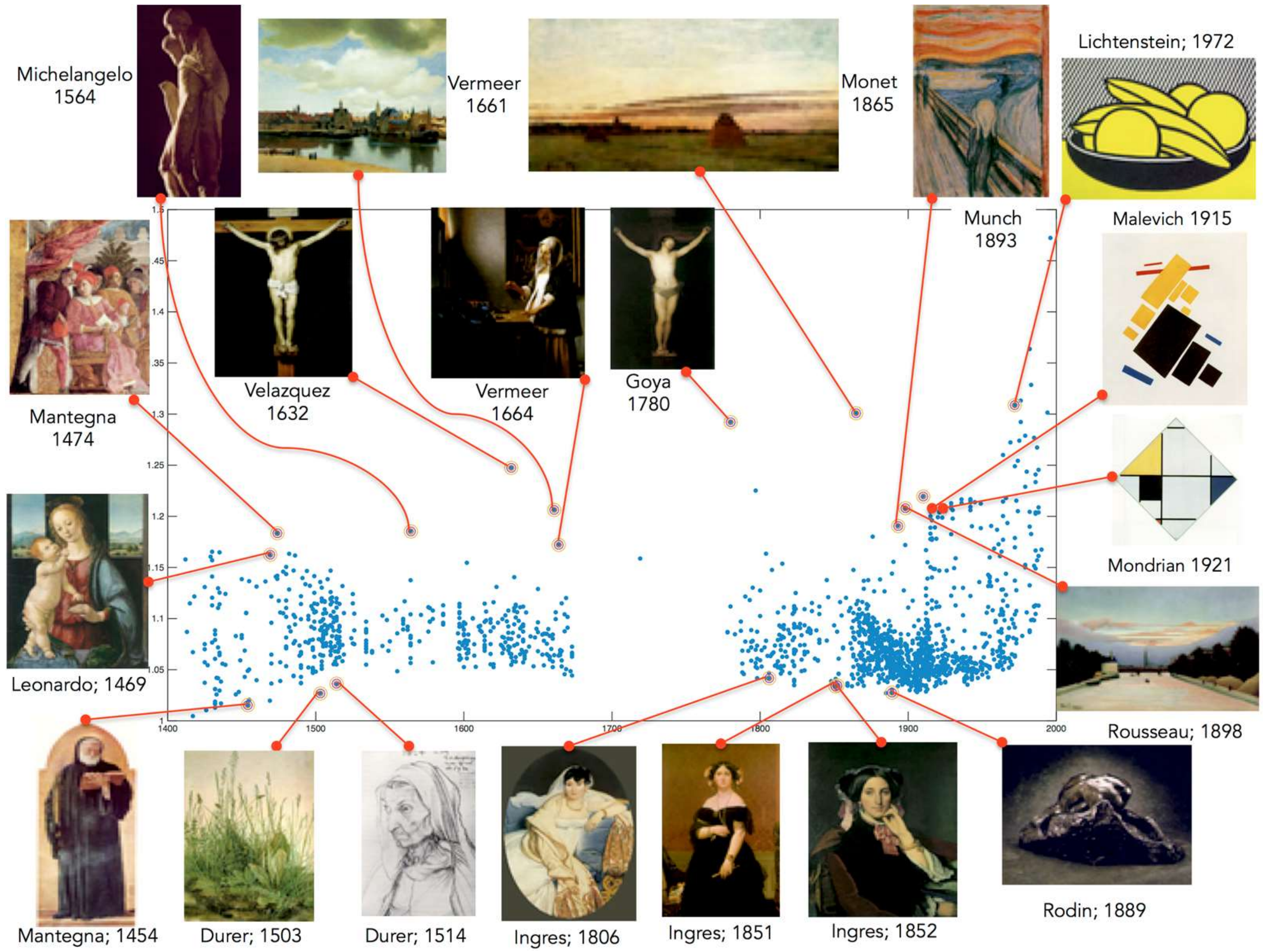
Frederic Bazille's Studio 9 Rue de la Condamine (left) and Norman Rockwell's Shuffleton's Barber Shop (right).

→ The composition of both paintings is divided in a similar way.

→ Yellow circles indicate similar objects, red lines indicate composition, and the blue square represents similar structural element.

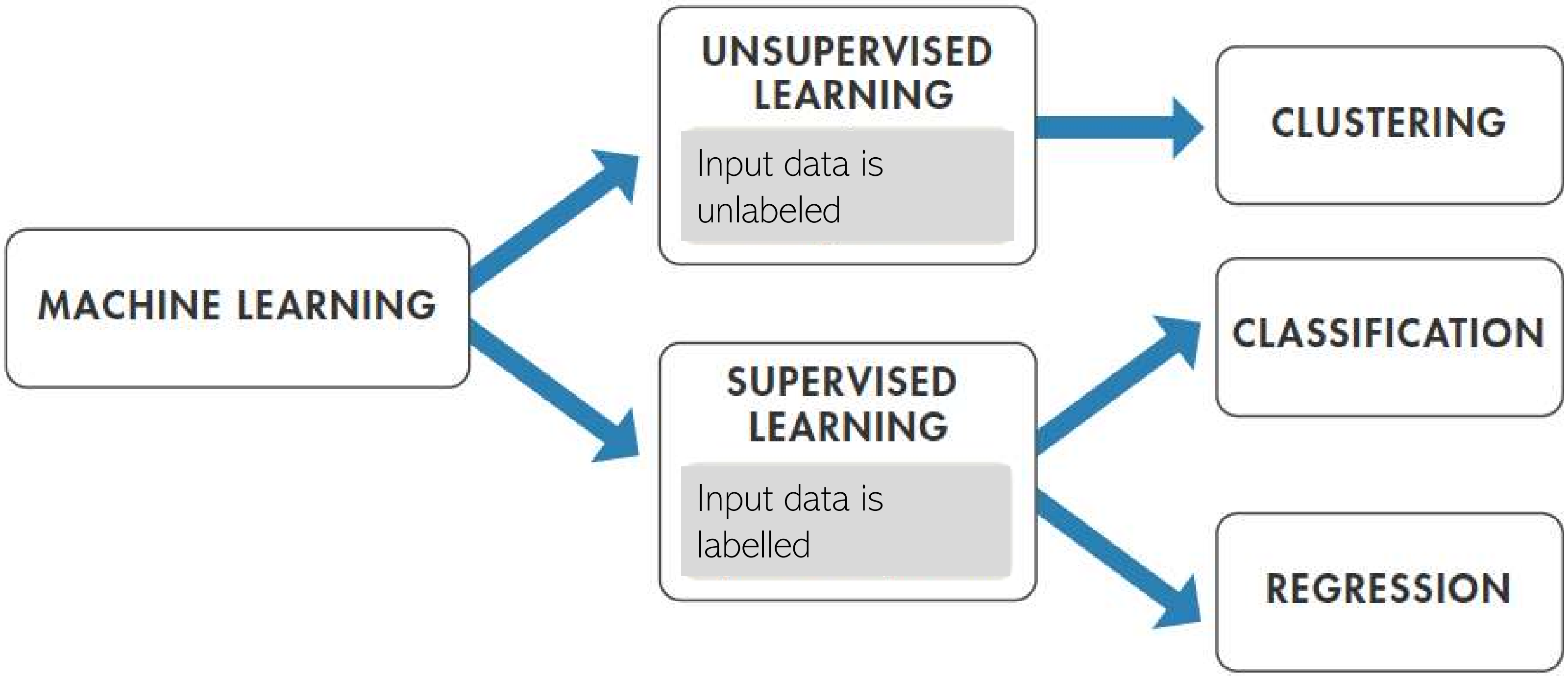


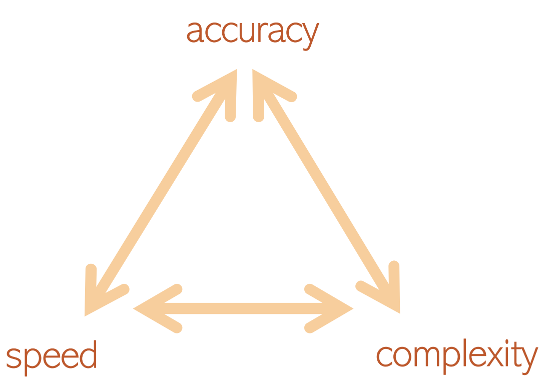
GERICAULT
BOTTICELLI
BELLINI
CAMPIN
VAN EYCK
TITIAN
RUBENS
MICHELANGELO
MANTEGNA
INGRES
DONATELLO
DELACROIX
RAPHAEL
DURER
LEONARDO
RICHTER
Okeeffe
MOTHERWELL
LICHTENSTEIN
HEPWORTH
Bacon
HOCKNEY
MALEVICH
Chagall
KAHLO
KANDINSKY
KIRCHNER
MACKE
DELAUNAY
BAZILLE
GHIBERTI
MARC
MORISOT
rockwell
Picasso
MANET
KLINE
RODIN
CAILLEBOTTE
BRAQUE
DEGAS
Beckmann
CEZANNE
Warhol
Miro
RENOIR
Rothko
Klee
KLIMT
VAN GOGH
PISSARRO
Munch
JOHNS
SISLEY
MONDRIAN
ROUSSEAU
MONET

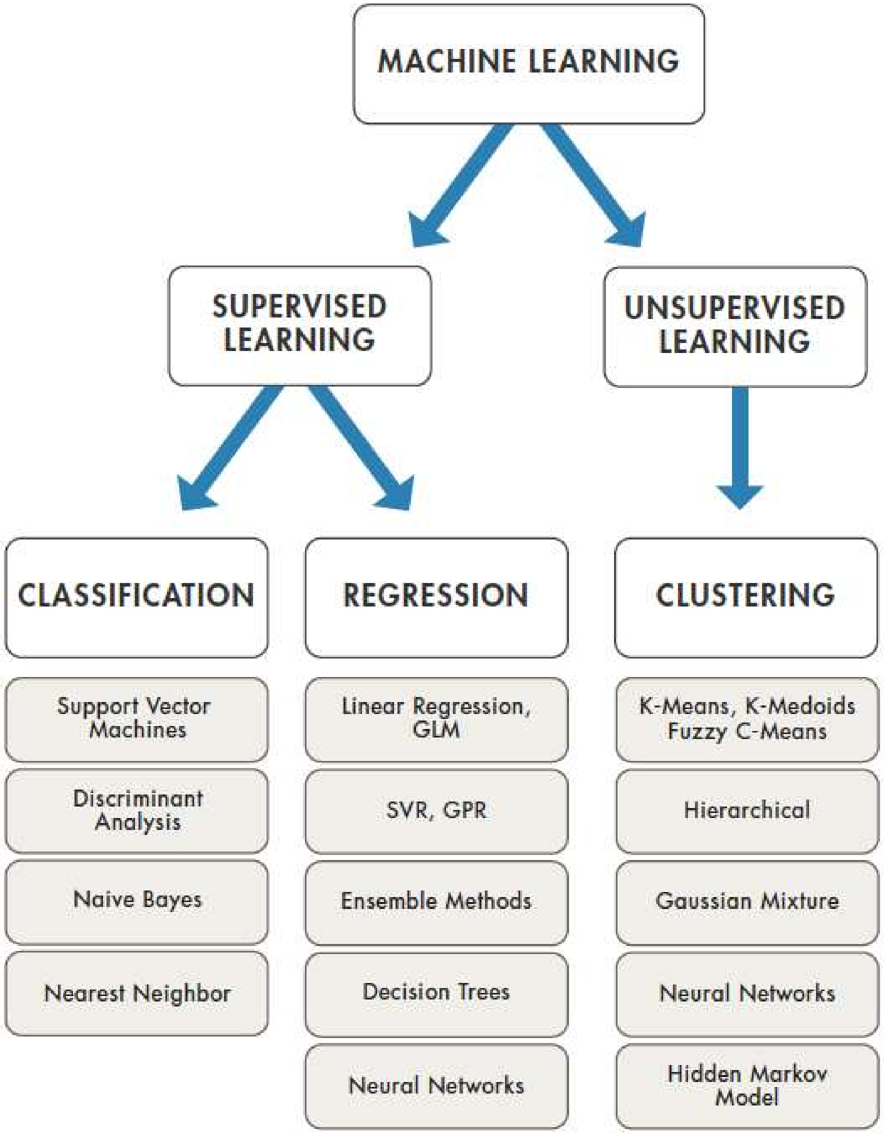


https://sites.google.com/site/digihumanlab/research/artistic-creativity

Types of Learning




accuracy
speed
complexity


MACHINE LEARNING
SUPERVISED LEARNING
UNSUPERVISED LEARNING
CLASSIFICATION
REGRESSION
CLUSTERING
Support Vector Machines
Discriminant Analysis
Naive Bayes
Nearest Neighbor
Linear Regression, GLM
SVR, GPR
Ensemble Methods
Decision Trees
Neural Networks
K-Means, K-Medoids Fuzzy C-Means
Hierarchical
Gaussian Mixture
Neural Networks
Hidden Markov Model

Supervised Learning

Supervised learning is the most common approach in Biomedical Engineering applications.

- uses data that is labelled into classes to train an algorithm on how to sort data
- Has a data set for training and one for validation

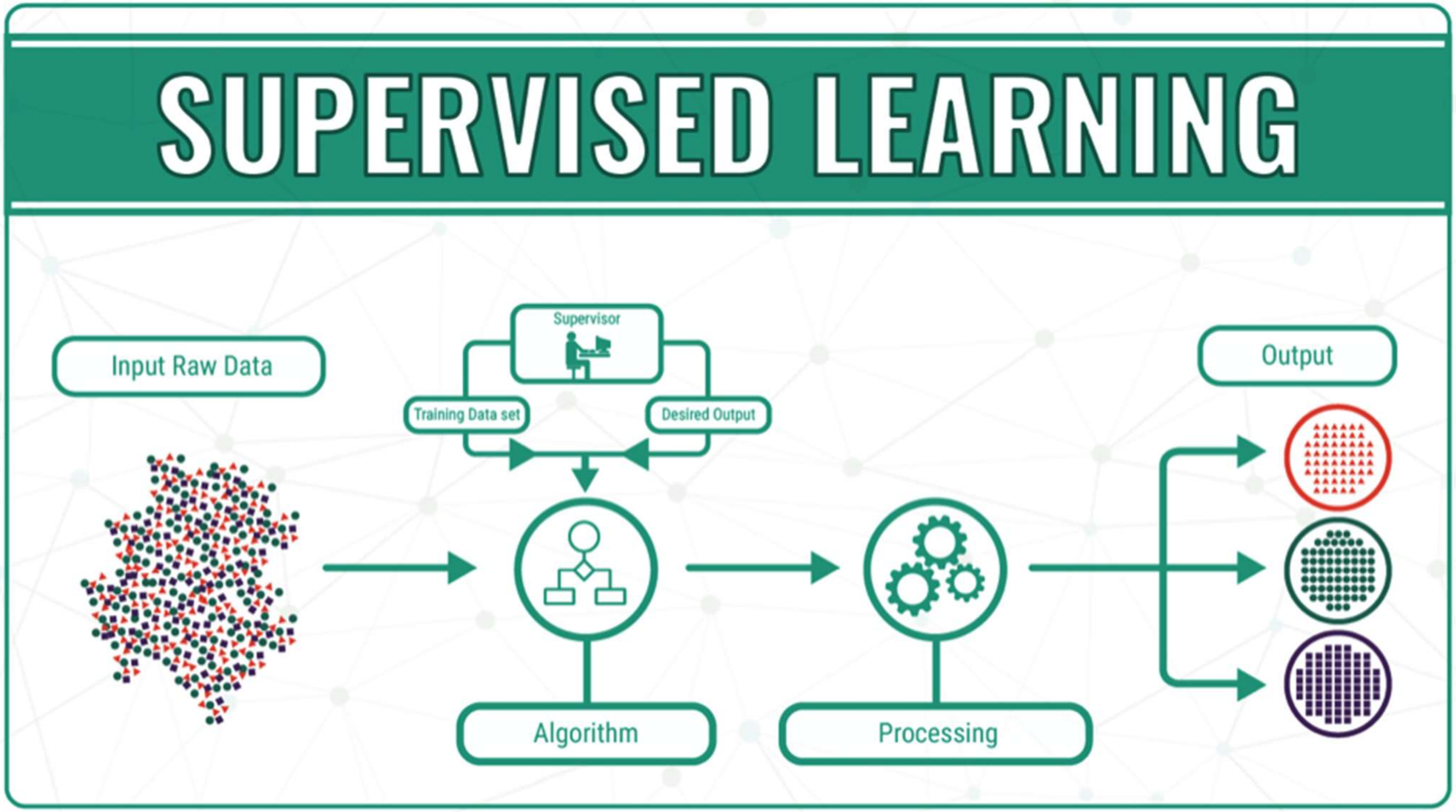
2 Steps:

Training

- Parameters start out classifier free and are adjusted to minimize classification errors
- Use a subset of the data called the training set to let algorithm know which group the data belongs in

Validation

- A validation subset is used that the algorithm has never seen
- correct classification is also known but is not used to modify classifier parameters, just check accuracy
- Classifier should perform with minimum error on data that it has never seen.



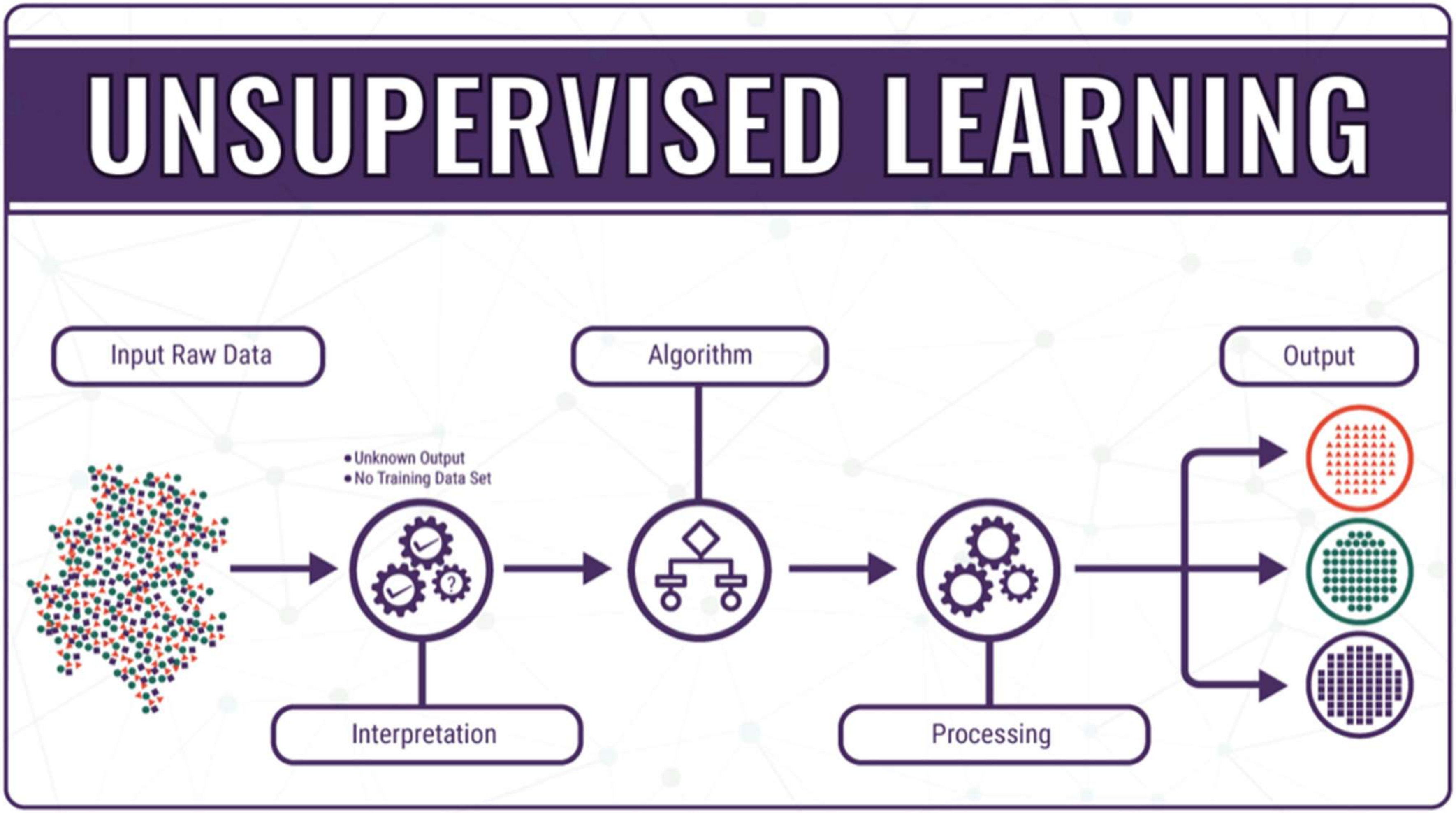
https://medium.com/@himanshuit3036/supervised-learning-methods-using-python-bb85b8c4e0b7

Unsupervised Learning

- Data is not labelled
- the classifier attempts to find patterns within the data with no a priori knowledge of the data patterns and, perhaps, not even the number of classes that exist.
- used to draw inferences from datasets consisting of input data without labeled responses.

Clustering

- most common unsupervised learning technique.
- used for exploratory data analysis to find hidden patterns or groupings in data.
- Applications include gene sequence analysis, market research, and object recognition



https://technative.io/why-unsupervised-machine-learning-is-the-future-of-cybersecurity/

Supervised Learning

Pros

Easier to understand the sorting

Know how many classes there are before sorting

Can be very picky with class definition to tailor the decision boundary accuracy

Once it is trained, you can stop changing the algorithm

Cons

Expensive to label

Don't always have knowledge to create label

Fairly simple systems, can't handle complex sorting as well

Doesn't give you unknown information

- You have to understand which features and groups you are giving it

Possible to over-train

Unsupervised Learning

Pros

Useful for very large sets

Useful for unlabeled data

- Labelling is time consuming and expensive
- Labels can be confirmed after they have been sorted

Finds patterns that may be difficult to find otherwise

Good at identifying outliers

Cons

Time consuming to actually complete the algorithm

- Analyzes and calculates all possibilities for all data

The more features added, the more complicated and time consuming

Algorithm is always changing with the addition of new data

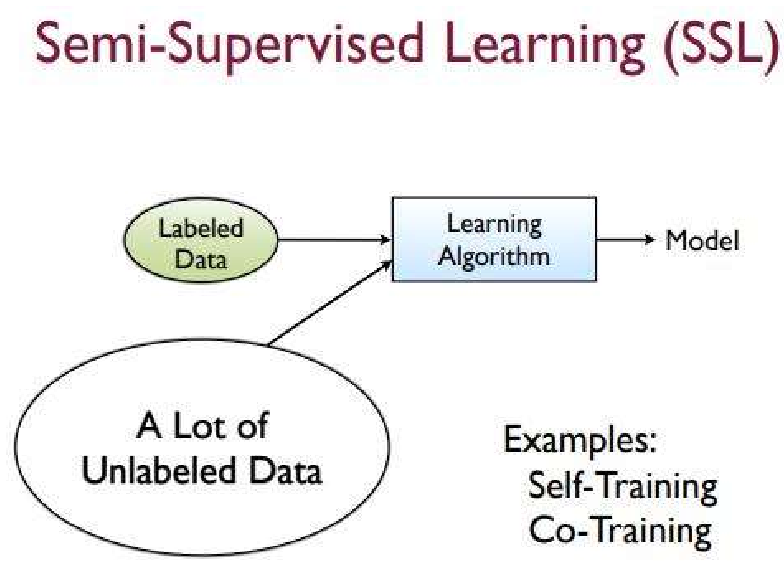
Semi Supervised

Unsupervised algorithms generate labels

These labels are used for a supervised algorithm

Humans may label some data

Can help with high cost of labelling



https://www.programmersought.com/article/19703919020/

Loss Function

Maps variable vs its “cost”

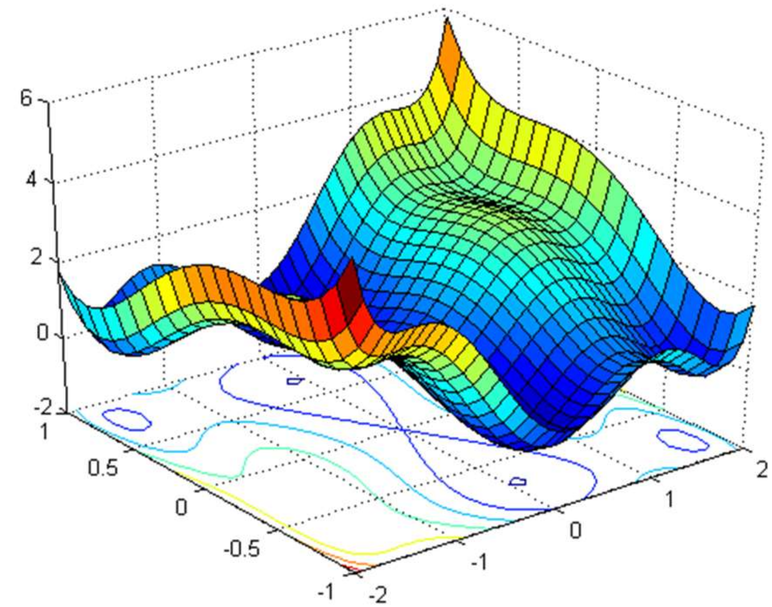
Plots the difference between the estimated and true value of a variable

Also known as error / cost function

In general the goal of any optimization algorithm is to minimize the loss function

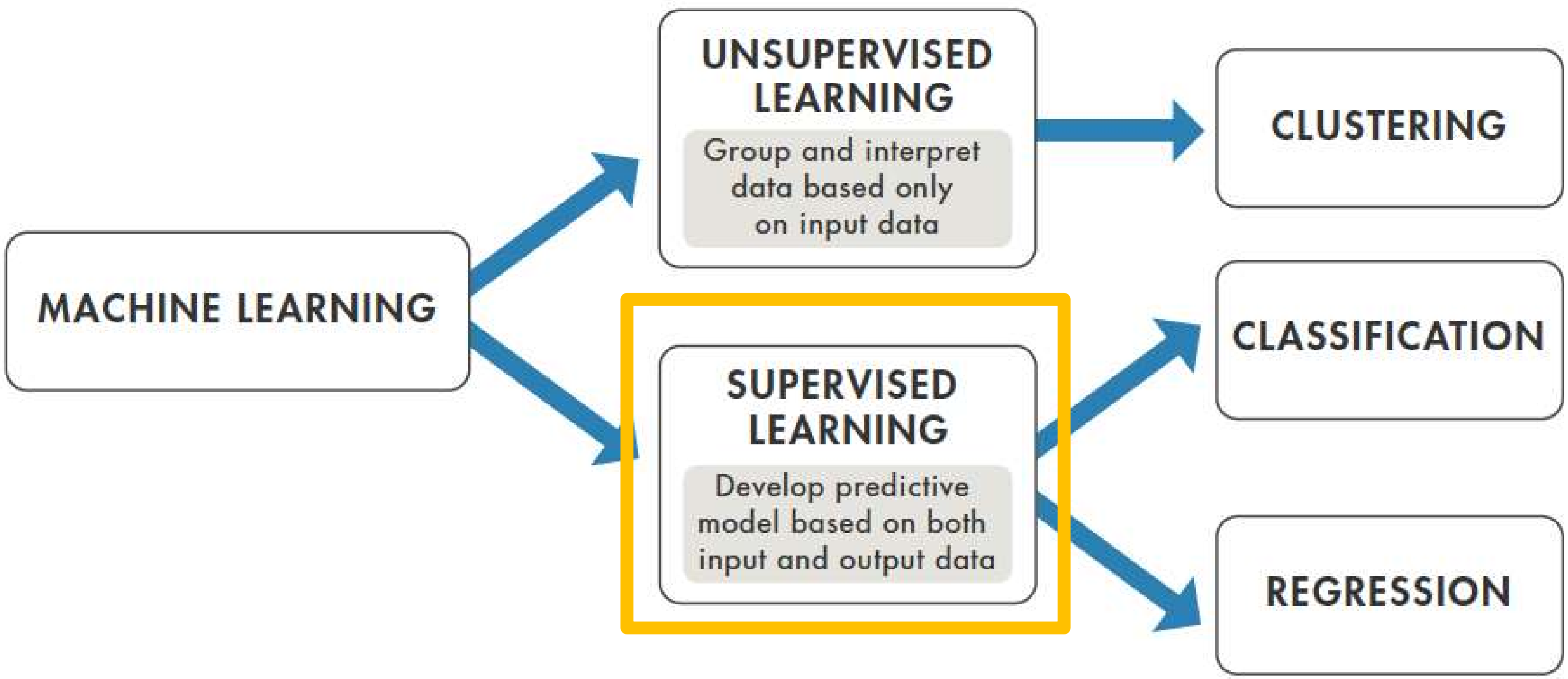
Different ways to calculate Loss:

- Mean squared error
- Log-loss
- Likelihood loss



https://algorithmia.com/blog/introduction-to-loss-functions

Types of Learning



Supervised Learning

Trains a model on known input and output data so that it can predict future outputs

Classification techniques

- Predict discrete responses (e.g. is a tumor is cancerous or benign).
- classify input data into categories.
- typical applications include medical imaging and speech recognition.

Regression techniques

predict continuous responses (e.g. changes in metabolic demand with exercise)

- engineering application example is electricity load forecasting.

Supervised Learning (cont.)

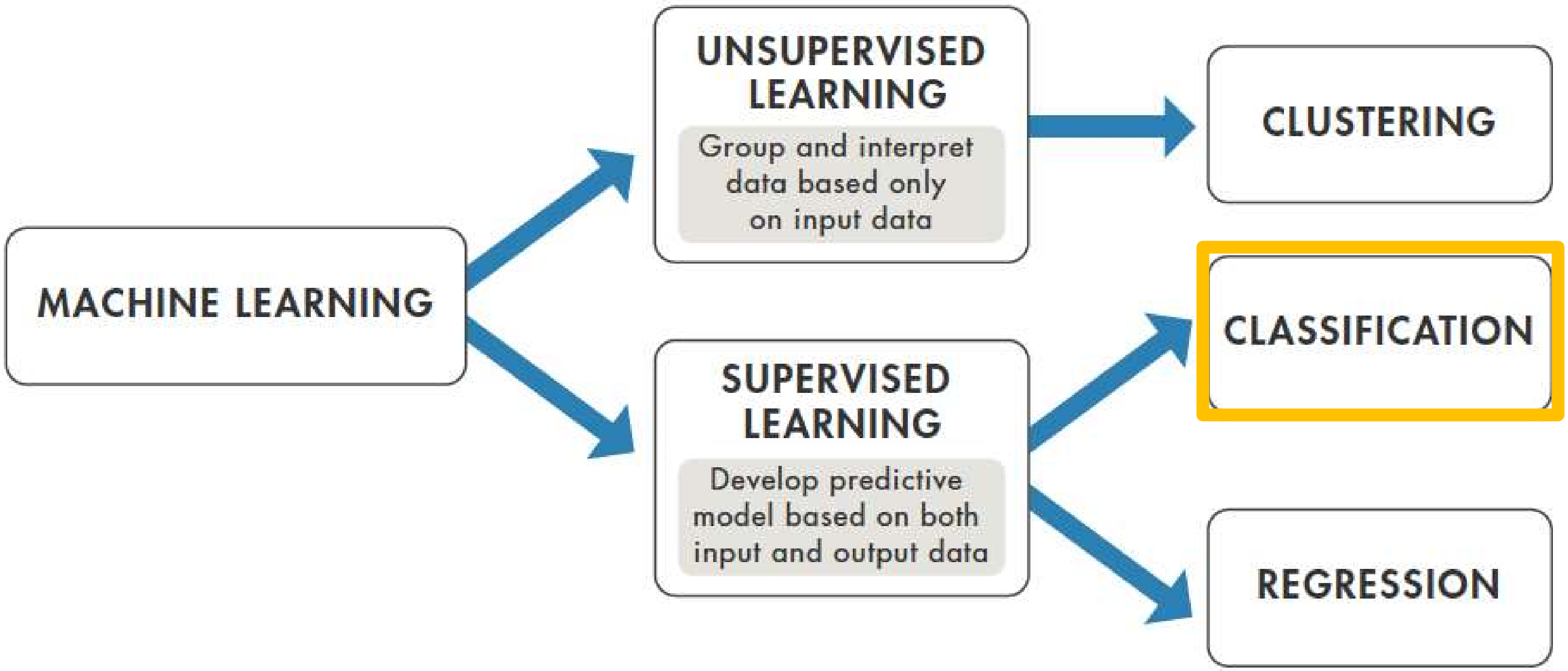
When training is complete, the classifier is applied to a test set where it performs its designed function to determine the most likely condition based on a given data pattern.

It is only the classification error that occurs during the testing phase that really matters.

A common failure of classifiers is that they perform well on the training set (after training)… but then do poorly on the test data aka their real world application.

Such classifiers are said to generalize poorly and/or be over fit

Types of Learning



Classification

- Determining a disease or condition from a range of measurements or other diagnostic data is an application of classification.
- Classification is applied to a pattern of descriptors. It finds a class which best fits each pattern.
- Classification attempts to associate a pattern of input variables with either a specific class or another variable.
  - If the output is a variable, the analysis is referred to as "regression"
  - If the output is a discrete number identifying a specific class, the analysis is referred to as "classification."

Classifiers

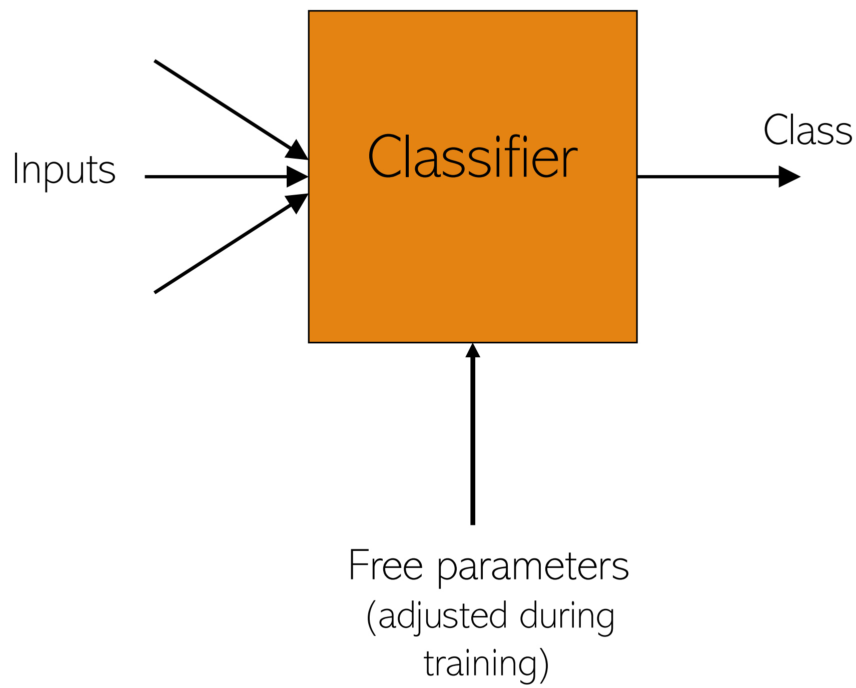
Classifiers establish a relationship between an input pattern and a discrete output

they can be viewed as mathematical functions and classifier development can thought of as function approximation

For the pattern of a number of inputs, it determines the mostly likely condition associated with that pattern.

The inputs can have any value, but typical output values are ± 1 or 0

Classification is done using two basic strategies: supervised and unsupervised learning.



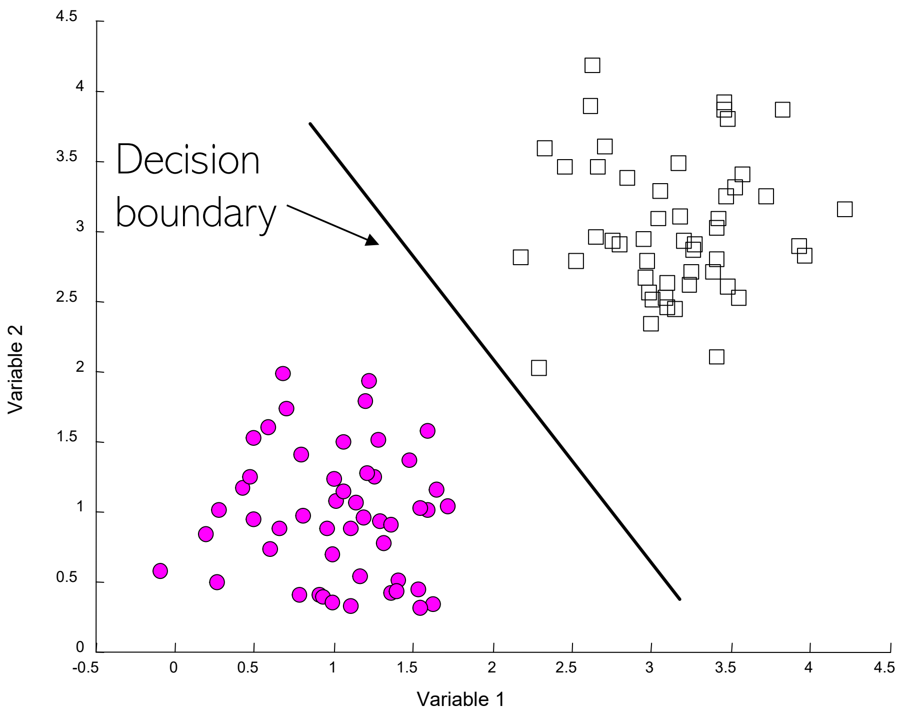
The Classification Problem

This figure shows a graphical representation of a typical classification problem.

There are 2 classes and 2 descriptive variables.

The 2 classes can be easily identified from the 2-variable scattergram.

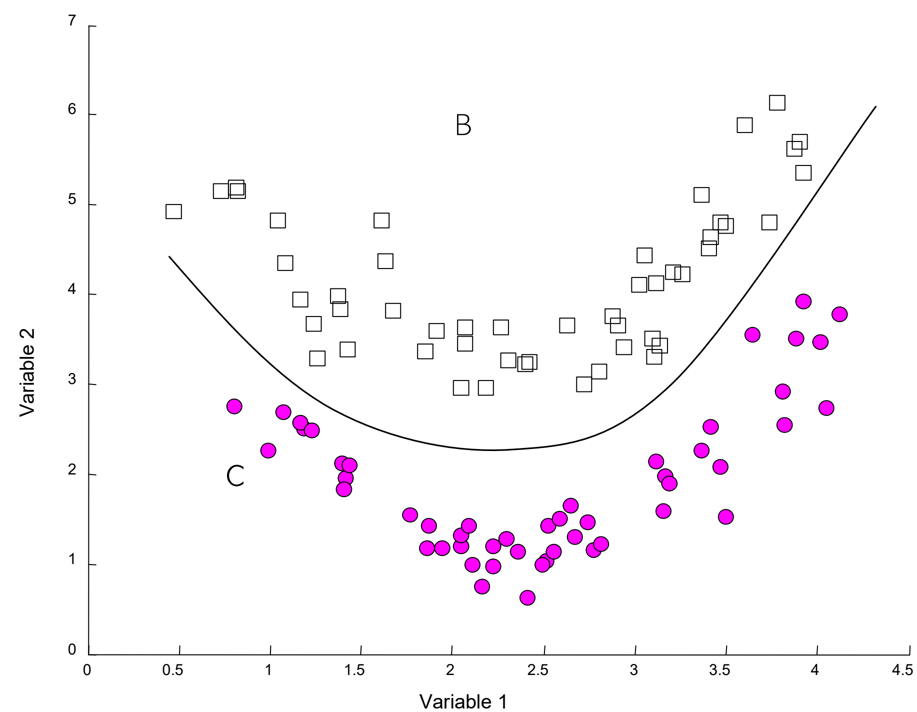
A decision boundary separates measurement patterns associated with the two classes.



Nonlinearly Separable

Data is still separable but requires a more complex shape to do so

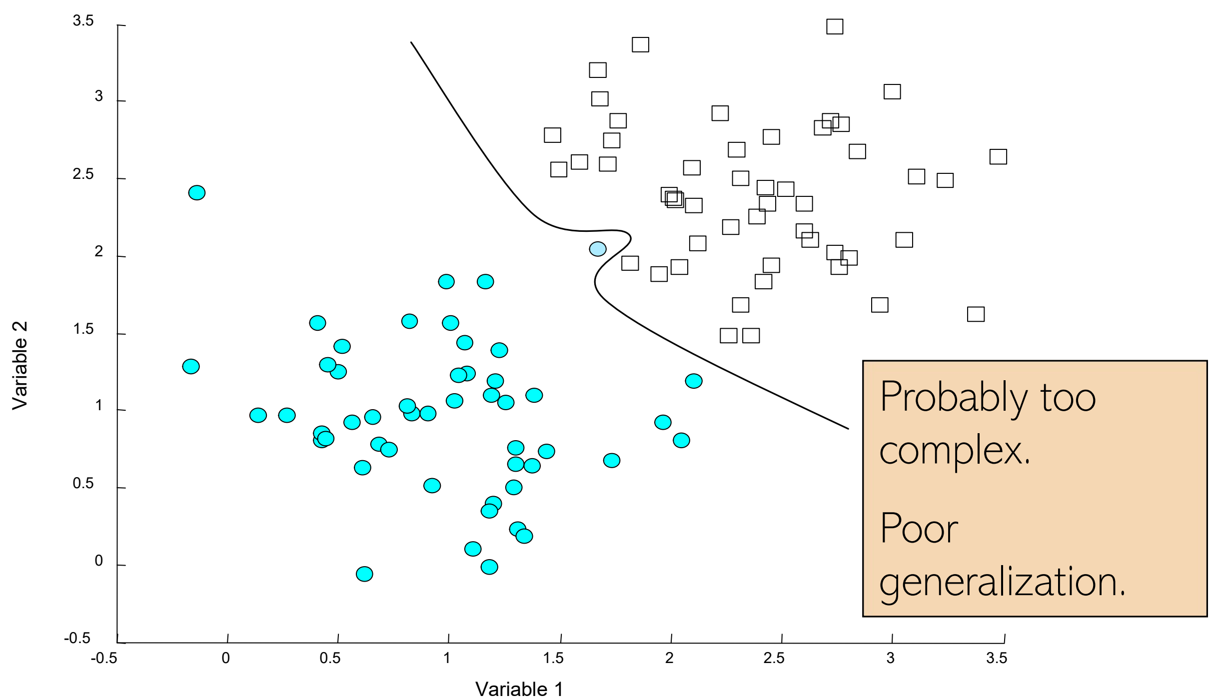
These two classes can still be correctly identified, but a curve is needed to separate them.



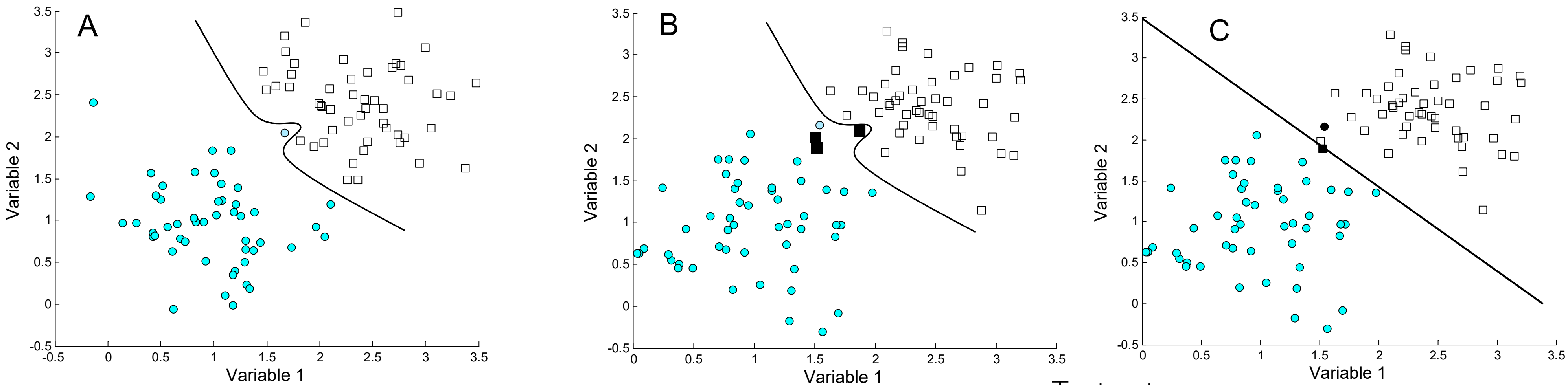
More complicated…

The two classes overlap somewhat, and a very complicated boundary is required to separate them.

If this is a training set, the boundary shown is unlikely to generalize well.



Overtraining



Training set

Test set

Three errors occur when this complicated boundary is applied to a test set.

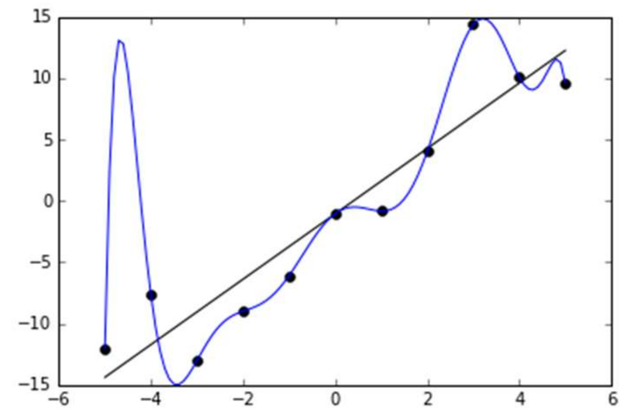
A simple straight line produces only two errors.

Over training

Algorithm works very well with training set but doesn't work on other data

Occurs when there is too little data or the data is inhomogeneous

Over fitting
- Model contains more variables than can be explained by the data
- Model follows data too closely and cannot be generalized and used for other data sets



https://en.wikipedia.org/wiki/Overfitting#/media/File:Overfitted_Data.png

Machine Capacity

The complexity of the boundary is determined by the classification algorithm.

Since classification is a form of machine learning, classification algorithm complexity is termed machine capacity.

If capacity is too large for the data, the classifier will overtrain. It will perform well on the training set, but will not generalize well and perform poorly on the test set.

A machine with too little capacity will show excessive errors in training and sub-par performance in classifying the test set.