# System Modeling, part 2

Marc Claesen

July 9, 2015

1. Nonlinear systems & linearization

2. System identification (cont)
    - Grey box identification
    - Black box identification

# Outline

1. Nonlinear systems & linearization

2. System identification (cont)
   - Grey box identification
   - Black box identification

# Nonlinear systems

In this course we focus on the linear state-space representation:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), \\ y(t) = Cx(t) + Du(t). \end{cases} \qquad \begin{cases} x[k+1] & = Ax[k] + Bu[k], \\ \quad y[k] & = Cx[k] + Du[k]. \end{cases}$$

# Nonlinear systems

In this course we focus on the linear state-space representation:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), \\ y(t) = Cx(t) + Du(t). \end{cases} \qquad \begin{cases} x[k+1] & = Ax[k] + Bu[k], \\ \quad y[k] & = Cx[k] + Du[k]. \end{cases}$$

Most real life systems involve nonlinearity:

$$\begin{cases} \dot{x}(t) = f\big(x(t), u(t)\big), \\ y(t) = g\big(x(t), u(t)\big), \end{cases}$$

# Nonlinear systems

In this course we focus on the linear state-space representation:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), \\ y(t) = Cx(t) + Du(t). \end{cases} \qquad \begin{cases} x[k+1] & = Ax[k] + Bu[k], \\ \quad y[k] & = Cx[k] + Du[k]. \end{cases}$$

Most real life systems involve nonlinearity:

$$\begin{cases} \dot{x}(t) = f\big(x(t), u(t)\big), \\ y(t) = g\big(x(t), u(t)\big), \end{cases}$$

where $f$ and/or $g$ contain some nonlinearity, such as:

# Nonlinear systems

In this course we focus on the linear state-space representation:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), \\ y(t) = Cx(t) + Du(t). \end{cases} \qquad \begin{cases} x[k+1] & = Ax[k] + Bu[k], \\ \quad y[k] & = Cx[k] + Du[k]. \end{cases}$$

Most real life systems involve nonlinearity:

$$\begin{cases} \dot{x}(t) = f\big(x(t), u(t)\big), \\ y(t) = g\big(x(t), u(t)\big), \end{cases}$$

where $f$ and/or $g$ contain some nonlinearity, such as:

- *powers*: e.g. $\dot{x}(t) = Ax(t) + Bu(t) + \gamma u(t)^2$,

# Nonlinear systems

In this course we focus on the linear state-space representation:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), \\ y(t) = Cx(t) + Du(t). \end{cases} \qquad \begin{cases} x[k+1] & = Ax[k] + Bu[k], \\ \quad y[k] & = Cx[k] + Du[k]. \end{cases}$$

Most real life systems involve nonlinearity:

$$\begin{cases} \dot{x}(t) = f\big(x(t), u(t)\big), \\ y(t) = g\big(x(t), u(t)\big), \end{cases}$$

where $f$ and/or $g$ contain some nonlinearity, such as:

- *powers*: e.g. $\dot{x}(t) = Ax(t) + Bu(t) + \gamma u(t)^2$,
- *interactions*: e.g. $\dot{x}(t) = Ax(t) + Bu(t) + \gamma x(t)u(t)$,

# Nonlinear systems

In this course we focus on the linear state-space representation:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), \\ y(t) = Cx(t) + Du(t). \end{cases} \qquad \begin{cases} x[k+1] & = Ax[k] + Bu[k], \\ \quad y[k] & = Cx[k] + Du[k]. \end{cases}$$

Most real life systems involve nonlinearity:

$$\begin{cases} \dot{x}(t) = f\big(x(t), u(t)\big), \\ y(t) = g\big(x(t), u(t)\big), \end{cases}$$

where $f$ and/or $g$ contain some nonlinearity, such as:

- *powers*: e.g. $\dot{x}(t) = Ax(t) + Bu(t) + \gamma u(t)^2$,
- *interactions*: e.g. $\dot{x}(t) = Ax(t) + Bu(t) + \gamma x(t)u(t)$,
- *clipping*: e.g. $\alpha \leq x(t) \leq \beta$.

# Linearization around equilibrium point

# Linearization around equilibrium point

Nonlinear systems have (several) equilibrium points $x_e$, $u_e$, $y_e$:

$$\begin{cases} \dot{x}_e = f\left(x_e, u_e\right) = 0, \\ y_e = g\left(x_e, u_e\right). \end{cases}$$

# Linearization around equilibrium point

Nonlinear systems have (several) equilibrium points $x_e$, $u_e$, $y_e$:

$$\begin{cases} \dot{x}_e = f\left(x_e, u_e\right) = 0, \\ y_e = g\left(x_e, u_e\right). \end{cases}$$

Linearizing in the region of $(x_e, u_e, y_e)$:

$$x = x_e + \Delta x, \quad u = u_e + \Delta u, \quad y = y_e + \Delta y,$$

# Linearization around equilibrium point

Nonlinear systems have (several) equilibrium points $x_e$, $u_e$, $y_e$:

$$\begin{cases} \dot{x}_e = f\big(x_e, u_e\big) = 0, \\ y_e = g\big(x_e, u_e\big). \end{cases}$$

Linearizing in the region of $(x_e, u_e, y_e)$:

$$x = x_e + \Delta x, \quad u = u_e + \Delta u, \quad y = y_e + \Delta y,$$

with $\Delta x$, $\Delta u$ and $\Delta y$ *sufficiently* small.

# Linearization around equilibrium point

Nonlinear systems have (several) equilibrium points $x_e$, $u_e$, $y_e$:

$$\begin{cases} \dot{x}_e = f\left(x_e, u_e\right) = 0, \\ y_e = g\left(x_e, u_e\right). \end{cases}$$

Linearizing in the region of $(x_e, u_e, y_e)$:

$$x = x_e + \Delta x, \quad u = u_e + \Delta u, \quad y = y_e + \Delta y,$$

with $\Delta x$, $\Delta u$ and $\Delta y$ *sufficiently* small.

Linearizing is done via first order Taylor expansions:

$$\begin{cases} \frac{dx}{dt} = \frac{d\Delta x}{dt} = f(x, u) = f(x_e + \Delta x, u_e + \Delta u), \\ y_e + \Delta y = g(x, u) = g(x_e + \Delta x, u_e + \Delta u). \end{cases}$$

# Example: decalcification plant

Used to reduce concentration of calcium hydroxide in water:

- chemical reaction: $Ca(OH)_2 + CO_2 \to CaCO_3 + H_2O$
- reaction speed: $r = c[Ca(OH)_2][CO_2]$
- rate of change of concentration:

$$\frac{d[Ca(OH)_2]}{dt} = \frac{k}{V} - \frac{r}{V},$$
$$\frac{d[CO_2]}{dt} = \frac{u}{V} - \frac{r}{V},$$

  with inflow rates $k$ and $u$ in mol/s and tank volume $V$ in L.
- input $u$: inflow of $CO_2$, output: $[Ca(OH)_2]$

# Nonlinear model and equilibrium point

Nonlinear model for the given reactor:

# Nonlinear model and equilibrium point

Nonlinear model for the given reactor:

$$\begin{aligned}
\frac{d[Ca(OH)_2]}{dt} &= \frac{k}{V} - \frac{c}{V}[Ca(OH)_2][CO_2], \\
\frac{d[CO_2]}{dt} &= \frac{u}{V} - \frac{c}{V}[Ca(OH)_2][CO_2], \\
y &= [Ca(OH)_2],
\end{aligned}$$

with two state variables: $x_1 = [Ca(OH)_2]$ and $x_2 = [CO_2]$.

# Nonlinear model and equilibrium point

Nonlinear model for the given reactor:

$$\frac{d[Ca(OH)_2]}{dt} = \frac{k}{V} - \frac{c}{V}[Ca(OH)_2][CO_2],$$
$$\frac{d[CO_2]}{dt} = \frac{u}{V} - \frac{c}{V}[Ca(OH)_2][CO_2],$$
$$y = [Ca(OH)_2],$$

with two state variables: $x_1 = [Ca(OH)_2]$ and $x_2 = [CO_2]$.

The equilibrium point $(k_{eq}, u_{eq}, x_{1,eq}, x_{2,eq}, y_{eq})$ of this system is:

# Nonlinear model and equilibrium point

Nonlinear model for the given reactor:

$$\frac{d[Ca(OH)_2]}{dt} = \frac{k}{V} - \frac{c}{V}[Ca(OH)_2][CO_2],$$
$$\frac{d[CO_2]}{dt} = \frac{u}{V} - \frac{c}{V}[Ca(OH)_2][CO_2],$$
$$y = [Ca(OH)_2],$$

with two state variables: $x_1 = [Ca(OH)_2]$ and $x_2 = [CO_2]$.

The equilibrium point $(k_{eq}, u_{eq}, x_{1,eq}, x_{2,eq}, y_{eq})$ of this system is:

$$\frac{k_{eq}}{V} - \frac{c}{V}[Ca(OH)_2]_{eq}[CO_2]_{eq} = 0,$$
$$\frac{u_{eq}}{V} - \frac{c}{V}[Ca(OH)_2]_{eq}[CO_2]_{eq} = 0.$$

# Linearization of the decalcification plant

For small deviations near the equilibrium:

$$\begin{aligned}
\frac{d\Delta x_1}{dt} &= -\frac{c}{V}[CO_2]_{eq}\Delta x_1 - \frac{c}{V}[Ca(OH)_2]_{eq}\Delta x_2,\\
\frac{d\Delta x_2}{dt} &= -\frac{c}{V}[CO_2]_{eq}\Delta x_1 - \frac{c}{V}[Ca(OH)_2]_{eq}\Delta x_2 + \frac{\Delta u}{V},\\
\Delta y &= \Delta x_1.
\end{aligned}$$

# Linearization of the decalcification plant

For small deviations near the equilibrium:

$$\begin{aligned}\frac{d\Delta x_1}{dt} &= -\frac{c}{V}[CO_2]_{eq}\Delta x_1 - \frac{c}{V}[Ca(OH)_2]_{eq}\Delta x_2,\\ \frac{d\Delta x_2}{dt} &= -\frac{c}{V}[CO_2]_{eq}\Delta x_1 - \frac{c}{V}[Ca(OH)_2]_{eq}\Delta x_2 + \frac{\Delta u}{V},\\ \Delta y &= \Delta x_1.\end{aligned}$$

The resulting linear state-space model is $\begin{cases}\dot{x}(t) = Ax(t) + Bu(t)\\ y(t) = Cx(t) + Du(t)\end{cases}$:

# Linearization of the decalcification plant

For small deviations near the equilibrium:

$$\begin{aligned}
\frac{d\Delta x_1}{dt} &= -\frac{c}{V}[CO_2]_{eq}\Delta x_1 - \frac{c}{V}[Ca(OH)_2]_{eq}\Delta x_2,\\
\frac{d\Delta x_2}{dt} &= -\frac{c}{V}[CO_2]_{eq}\Delta x_1 - \frac{c}{V}[Ca(OH)_2]_{eq}\Delta x_2 + \frac{\Delta u}{V},\\
\Delta y &= \Delta x_1.
\end{aligned}$$

The resulting linear state-space model is $\begin{cases}\dot{x}(t) = Ax(t) + Bu(t)\\ y(t) = Cx(t) + Du(t)\end{cases}$:

$$\begin{bmatrix}\frac{d[Ca(OH)_2]}{dt}\\ \frac{d[CO_2]}{dt}\end{bmatrix} = -\begin{bmatrix}\frac{c}{V}[CO_2]_{eq} & \frac{c}{V}[Ca(OH)_2]_{eq}\\ \frac{c}{V}[CO_2]_{eq} & \frac{c}{V}[Ca(OH)_2]_{eq}\end{bmatrix}\begin{bmatrix}[Ca(OH)_2]\\ [CO_2]\end{bmatrix} + \begin{bmatrix}0\\ \frac{1}{V}\end{bmatrix}u(t)$$

$$y(t) = [Ca(OH)_2]$$

# Outline

1. Nonlinear systems & linearization

2. System identification (cont)
    - Grey box identification
    - Black box identification

# Main classes of identification methods

**White box modeling**: based on first principles.

# Main classes of identification methods

**White box modeling**: based on first principles.
→ known equations (structure) & parameters (coefficients).

# Main classes of identification methods

**White box modeling**: based on first principles.
→ known equations (structure) & parameters (coefficients).

**Grey box identification**: first principles & experimentation.

# Main classes of identification methods

**White box modeling**: based on first principles.
→ known equations (structure) & parameters (coefficients).

**Grey box identification**: first principles & experimentation.
→ known equations, unknown/uncertain parameters.

# Main classes of identification methods

**White box modeling**: based on first principles.
→ known equations (structure) & parameters (coefficients).

**Grey box identification**: first principles & experimentation.
→ known equations, unknown/uncertain parameters.

**Black box identification**: based on experimentation.

# Main classes of identification methods

**White box modeling**: based on first principles.
→ known equations (structure) & parameters (coefficients).

**Grey box identification**: first principles & experimentation.
→ known equations, unknown/uncertain parameters.

**Black box identification**: based on experimentation.
→ unknown equations & unknown parameters.

# Main classes of identification methods

**White box modeling**: based on first principles.
→ known equations (structure) & parameters (coefficients).

**Grey box identification**: first principles & experimentation.
→ known equations, unknown/uncertain parameters.

**Black box identification**: based on experimentation.
→ unknown equations & unknown parameters.

Most popular approaches are forms of black box identification.

# Outline

1. Nonlinear systems & linearization

2. System identification (cont)
   - Grey box identification
   - Black box identification

# Grey box identification: conceptual

# Grey box identification: conceptual

Grey box identification starts from a known model structure but with unknown/uncertain parameters ↔ **parametric statistics**.

# Grey box identification: conceptual

Grey box identification starts from a known model structure but with unknown/uncertain parameters ↔ **parametric statistics**.

We assume linear, continuous time state space representation:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), \\ y(t) = Cx(t) + Du(t). \end{cases}$$

# Grey box identification: conceptual

Grey box identification starts from a known model structure but with unknown/uncertain parameters ↔ **parametric statistics**.

We assume linear, continuous time state space representation:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), \\ y(t) = Cx(t) + Du(t). \end{cases}$$

**Given**: states, inputs, outputs and guesstimates of $\tilde{A}$, $\tilde{B}$, $\tilde{C}$ & $\tilde{D}$.

# Grey box identification: conceptual

Grey box identification starts from a known model structure but with unknown/uncertain parameters ↔ **parametric statistics**.

We assume linear, continuous time state space representation:

$$\begin{cases} \dot{x}(t) = A x(t) + B u(t), \\ y(t) = C x(t) + D u(t). \end{cases}$$

**Given**: states, inputs, outputs and guesstimates of $\tilde{A}$, $\tilde{B}$, $\tilde{C}$ & $\tilde{D}$.
**Task**: estimate $\hat{A}$, $\hat{B}$, $\hat{C}$ and $\hat{D}$ adequately via experiments.

# Grey box identification: conceptual

Grey box identification starts from a known model structure but with unknown/uncertain parameters $\leftrightarrow$ **parametric statistics**.

We assume linear, continuous time state space representation:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), \\ y(t) = Cx(t) + Du(t). \end{cases}$$

**Given**: states, inputs, outputs and guesstimates of $\tilde{A}$, $\tilde{B}$, $\tilde{C}$ & $\tilde{D}$.
**Task**: estimate $\hat{A}$, $\hat{B}$, $\hat{C}$ and $\hat{D}$ adequately via experiments.

"*All models are wrong, but some are useful.*" - George E. P. Box

# Linear regression

Consider input matrix $\mathbf{X}$, output vector $\mathbf{y}$ and residuals $\epsilon$:

$$\mathbf{X}\theta = \mathbf{y} + \epsilon.$$

The parameter vector $\theta$ must be estimated, given the observations.

# Linear regression

Consider input matrix $\mathbf{X}$, output vector $\mathbf{y}$ and residuals $\epsilon$:

$$\mathbf{X}\theta = \mathbf{y} + \epsilon.$$

The parameter vector $\theta$ must be estimated, given the observations.

A common estimation approach is ordinary least squares (OLS):

$$\begin{aligned}(\mathbf{X}^T\mathbf{X})\hat{\theta}_{OLS} &= \mathbf{X}^T\mathbf{y},\\ \hat{\theta}_{OLS} &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}.\end{aligned}$$

# Linear regression

Consider input matrix $\mathbf{X}$, output vector $\mathbf{y}$ and residuals $\epsilon$:

$$\mathbf{X}\theta = \mathbf{y} + \epsilon.$$

The parameter vector $\theta$ must be estimated, given the observations.

A common estimation approach is ordinary least squares (OLS):

$$\begin{aligned}(\mathbf{X}^T\mathbf{X})\hat{\theta}_{OLS} &= \mathbf{X}^T\mathbf{y},\\ \hat{\theta}_{OLS} &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}.\end{aligned}$$

The OLS estimate minimizes the sum-of-squares of errors, i.e.:

$$\hat{\theta}_{OLS} = \arg\min_{\theta} \sum_{i=1}^{N} \Big(y(i) - \sum_{j=1}^{d} X(i,j)\theta(j)\Big)^2$$

# Linear regression with ordinary least squares

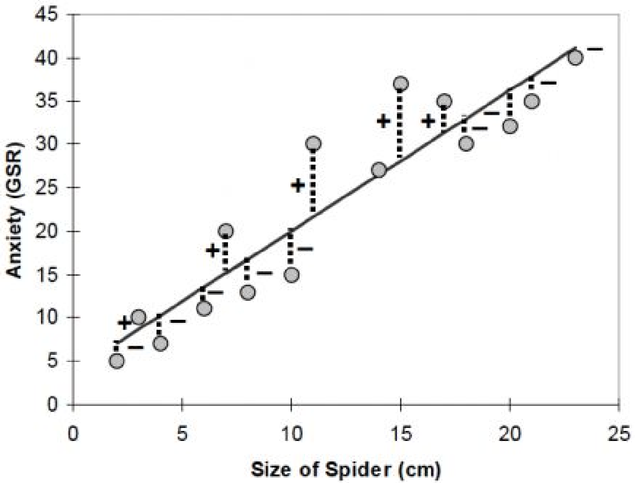


Image taken from `http://freakonometrics.hypotheses.org/2348`.

# Maximum likelihood estimation

The maximum likelihood estimate $\hat{\theta}_{ML}$ is the parameter vector that maximizes the likelihood $\mathcal{L}(\cdot)$ of observing the (known) outputs $\mathbf{y}$, given the (known) inputs $\mathbf{X}$:

$$\hat{\theta}_{ML} = \arg\max_{\theta} \mathcal{L}\big(\mathbf{y}, \mathbf{X} \mid \theta\big)$$

# Maximum likelihood estimation

The maximum likelihood estimate $\hat{\theta}_{ML}$ is the parameter vector that maximizes the likelihood $\mathcal{L}(\cdot)$ of observing the (known) outputs $\mathbf{y}$, given the (known) inputs $\mathbf{X}$:

$$\hat{\theta}_{ML} = \arg\max_{\theta} \mathcal{L}\big(\mathbf{y}, \mathbf{X} \mid \theta\big)$$

For some structures, ML estimate can be obtained in closed form.

# Maximum likelihood estimation

The maximum likelihood estimate $\hat{\theta}_{ML}$ is the parameter vector that maximizes the likelihood $\mathcal{L}(\cdot)$ of observing the (known) outputs $\mathbf{y}$, given the (known) inputs $\mathbf{X}$:

$$\hat{\theta}_{ML} = \arg\max_{\theta} \mathcal{L}\big(\mathbf{y}, \mathbf{X} \mid \theta\big)$$

For some structures, ML estimate can be obtained in closed form.

**Example**: least squares estimators are the maximum likelihood estimators if the associated residuals $\epsilon$ are normally distributed.

# Maximum a posteriori (MAP) estimation

Bayesian: maximum likelihood estimation with a *prior* $p(\theta)$.
$\rightarrow$ MAP estimation is a regularization of ML estimation

# Maximum a posteriori (MAP) estimation

Bayesian: maximum likelihood estimation with a *prior* $p(\theta)$.
$\rightarrow$ MAP estimation is a regularization of ML estimation

Bayes' theorem: $P(A \mid B) = P(B \mid A) \cdot P(A) \; / \; P(B)$.

# Maximum a posteriori (MAP) estimation

Bayesian: maximum likelihood estimation with a *prior* $p(\theta)$.
$\rightarrow$ MAP estimation is a regularization of ML estimation

Bayes' theorem: $P(A \mid B) = P(B \mid A) \cdot P(A) \ / \ P(B)$.

If a prior distribution $p(\cdot)$ is available for $\theta$, then the posterior distribution for $\theta$ becomes:

$$\theta \mapsto \mathcal{L}(\theta \mid \mathbf{y}, \mathbf{X}) = \frac{\mathcal{L}(\mathbf{y}, \mathbf{X} \mid \theta) p(\theta)}{\int_{\vartheta} \mathcal{L}(\mathbf{y}, \mathbf{X} \mid \vartheta) p(\vartheta) d\vartheta}.$$

# Maximum a posteriori (MAP) estimation

Bayesian: maximum likelihood estimation with a *prior* $p(\theta)$.
→ MAP estimation is a regularization of ML estimation

Bayes' theorem: $P(A \mid B) = P(B \mid A) \cdot P(A) \;/\; P(B)$.

If a prior distribution $p(\cdot)$ is available for $\theta$, then the posterior distribution for $\theta$ becomes:

$$\theta \mapsto \mathcal{L}(\theta \mid \mathbf{y}, \mathbf{X}) = \frac{\mathcal{L}(\mathbf{y}, \mathbf{X} \mid \theta) p(\theta)}{\int_{\vartheta} \mathcal{L}(\mathbf{y}, \mathbf{X} \mid \vartheta) p(\vartheta) d\vartheta}.$$

The MAP estimate is the mode of the posterior distribution of $\theta$:

$$\hat{\theta}_{MAP} = \arg\max_{\theta} \mathcal{L}(\mathbf{y}, \mathbf{X} \mid \theta) p(\theta).$$

# Errors-in-variables approach

Additionally accounts for measurement errors in inputs.
$\leftrightarrow$ standard regression only accounts for errors in *outputs*

# Errors-in-variables approach

Additionally accounts for measurement errors in inputs.
↔ standard regression only accounts for errors in *outputs*

Typically described via *latent variables*:

$$\begin{cases} x = x^\star + \eta, \\ y = y^\star + \epsilon, \\ y^\star = g(x^\star \mid \theta), \end{cases}$$

with $x$, $y$ the observed inputs, outputs and latent variables $x^\star$, $y^\star$.

# Errors-in-variables approach

Additionally accounts for measurement errors in inputs.
↔ standard regression only accounts for errors in *outputs*

Typically described via *latent variables*:

$$\begin{cases} x = x^\star + \eta, \\ y = y^\star + \epsilon, \\ y^\star = g(x^\star \mid \theta), \end{cases}$$

with $x$, $y$ the observed inputs, outputs and latent variables $x^\star$, $y^\star$.
**Assumption**: latent variables $x^\star$ and $y^\star$ exist which follow the true functional relationship $g(\cdot)$.

# Errors-in-variables approach

Additionally accounts for measurement errors in inputs.
↔ standard regression only accounts for errors in *outputs*

Typically described via *latent variables*:

$$\begin{cases} x = x^\star + \eta, \\ y = y^\star + \epsilon, \\ y^\star = g(x^\star \mid \theta), \end{cases}$$

with $x$, $y$ the observed inputs, outputs and latent variables $x^\star$, $y^\star$.
**Assumption**: latent variables $x^\star$ and $y^\star$ exist which follow the true functional relationship $g(\cdot)$.

**Task**: estimate $\theta$.

# Outline

1. Nonlinear systems & linearization
2. System identification (cont)
   - Grey box identification
   - Black box identification

# Black box identification

Start from unknown equations & unknown parameters.
→ related to **machine learning** and **nonparametric statistics**.

# Black box identification

Start from unknown equations & unknown parameters.
→ related to **machine learning** and **nonparametric statistics**.

If we assume a linear state space system:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), \\ y(t) = Cx(t) + Du(t). \end{cases} \qquad \begin{cases} x[k+1] & = Ax[k] + Bu[k], \\ y[k] & = Cx[k] + Du[k]. \end{cases}$$

# Black box identification

Start from unknown equations & unknown parameters.
→ related to **machine learning** and **nonparametric statistics**.

If we assume a linear state space system:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), \\ y(t) = Cx(t) + Du(t). \end{cases} \qquad \begin{cases} x[k+1] & = Ax[k] + Bu[k], \\ y[k] & = Cx[k] + Du[k]. \end{cases}$$

Black box identification deals with:

# Black box identification

Start from unknown equations & unknown parameters.
→ related to **machine learning** and **nonparametric statistics**.

If we assume a linear state space system:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), \\ y(t) = Cx(t) + Du(t). \end{cases} \qquad \begin{cases} x[k+1] & = Ax[k] + Bu[k], \\ y[k] & = Cx[k] + Du[k]. \end{cases}$$

Black box identification deals with:

- unknown states, both in number & physical interpretation

# Black box identification

Start from unknown equations & unknown parameters.
→ related to **machine learning** and **nonparametric statistics**.

If we assume a linear state space system:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), \\ y(t) = Cx(t) + Du(t). \end{cases} \qquad \begin{cases} x[k+1] & = Ax[k] + Bu[k], \\ y[k] & = Cx[k] + Du[k]. \end{cases}$$

Black box identification deals with:

- unknown states, both in number & physical interpretation
  → dimensions of $A$, $B$ & $C$ unknown

# Black box identification

Start from unknown equations & unknown parameters.
→ related to **machine learning** and **nonparametric statistics**.

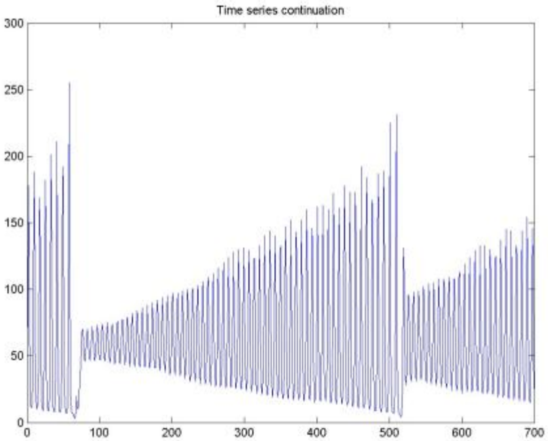
If we assume a linear state space system:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), \\ y(t) = Cx(t) + Du(t). \end{cases} \qquad \begin{cases} x[k+1] & = Ax[k] + Bu[k], \\ y[k] & = Cx[k] + Du[k]. \end{cases}$$

Black box identification deals with:

- unknown states, both in number & physical interpretation
  → dimensions of $A$, $B$ & $C$ unknown
- unknown parameters (values in $A$, $B$, $C$, $D$)

# Time series: Santa Fe laser



# Modelling the Santa Fe laster

This laser can be treated as an autonomous discrete time system:

$$\begin{cases} x[k+1] = f\big(x[k-N+1], \ldots, x[k]\big), \\ y[k] = x[k]. \end{cases}$$

The output depends on the past $N$ states & no inputs.

# Modelling the Santa Fe laster

This laser can be treated as an autonomous discrete time system:

$$\begin{cases} x[k+1] = f\left(x[k-N+1], \ldots, x[k]\right), \\ y[k] = x[k]. \end{cases}$$

The output depends on the past $N$ states & no inputs.
→ how large is $N$? → **unknown structure**

# Modelling the Santa Fe laster

This laser can be treated as an autonomous discrete time system:

$$\begin{cases} x[k+1] = f\big(x[k-N+1], \dots, x[k]\big), \\ y[k] = x[k]. \end{cases}$$

The output depends on the past $N$ states & no inputs.
→ how large is $N$? → **unknown structure**

Treat it as a regression problem with $N$ inputs: $y = f\big(X_1, \dots, X_N\big)$.

# Modelling the Santa Fe laster

This laser can be treated as an autonomous discrete time system:

$$\begin{cases} x[k+1] = f\big(x[k-N+1], \dots, x[k]\big), \\ y[k] = x[k]. \end{cases}$$

The output depends on the past $N$ states & no inputs.
→ how large is $N$? → **unknown structure**

Treat it as a regression problem with $N$ inputs: $y = f\big(X_1, \dots, X_N\big)$.
→ lets say linear, i.e. $y = \mathbf{X}\theta$ → **unknown parameters** $\theta \in \mathbb{R}^N$.

# Modelling the Santa Fe laster

This laser can be treated as an autonomous discrete time system:

$$\begin{cases} x[k+1] = f\big(x[k-N+1], \ldots, x[k]\big), \\ y[k] = x[k]. \end{cases}$$

The output depends on the past $N$ states & no inputs.
→ how large is $N$? → **unknown structure**

Treat it as a regression problem with $N$ inputs: $y = f\big(X_1, \ldots, X_N\big)$.
→ lets say linear, i.e. $y = \mathbf{X}\theta$ → **unknown parameters** $\theta \in \mathbb{R}^N$.
→ for given $N$, we can estimate $\theta$ via grey box methods.

# Modelling the Santa Fe laster

This laser can be treated as an autonomous discrete time system:

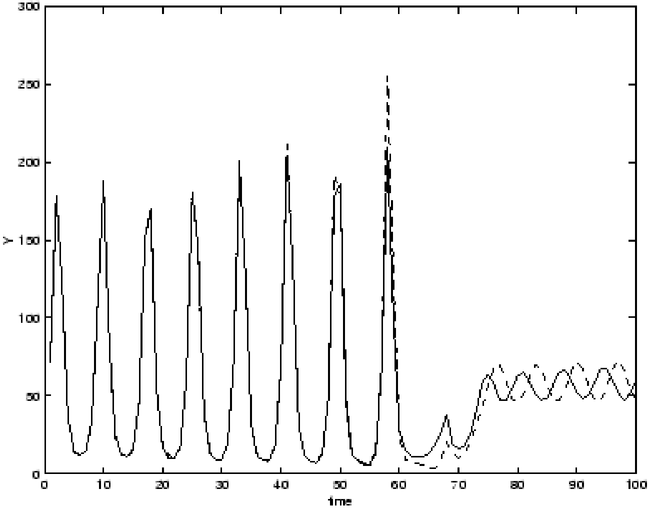
$$\begin{cases} x[k+1] = f\big(x[k-N+1], \ldots, x[k]\big), \\ y[k] = x[k]. \end{cases}$$

The output depends on the past $N$ states & no inputs.
→ how large is $N$? → **unknown structure**

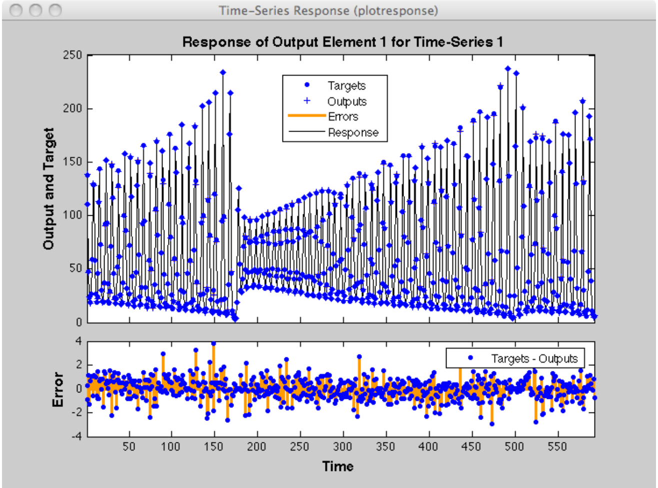
Treat it as a regression problem with $N$ inputs: $y = f\big(X_1, \ldots, X_N\big)$.
→ lets say linear, i.e. $y = \mathbf{X}\theta$ → **unknown parameters** $\theta \in \mathbb{R}^N$.
→ for given $N$, we can estimate $\theta$ via grey box methods.

Nonlinear models can be obtained via machine learning methods.
→ neural networks, support vector machine, random forest, ...

# Predictions of a least-squares support vector machine



# Predictions of an artificial neural network



# Neural network: biological

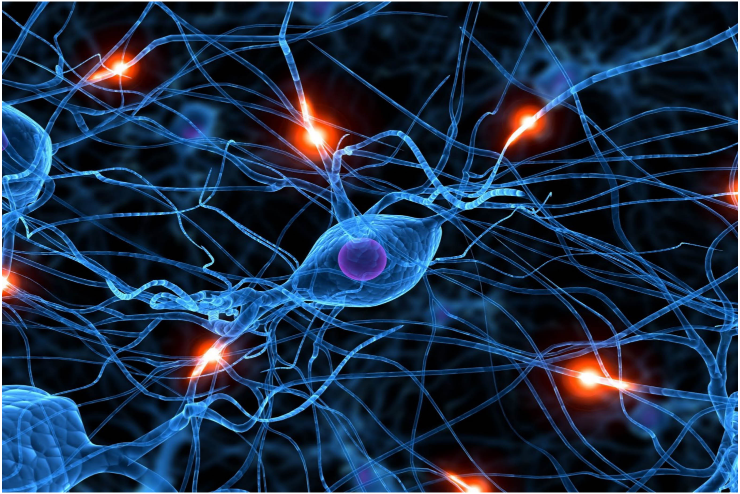


Image taken from http://www.extremetech.com/wp-content/uploads/2013/09/340.jpg.

# Structure of a single neuron

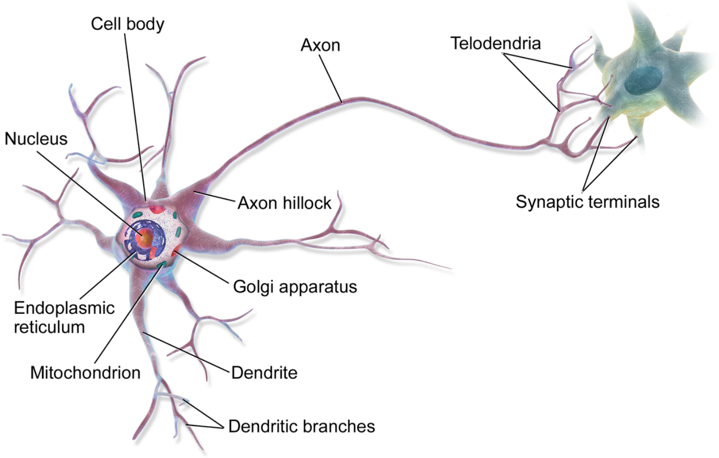
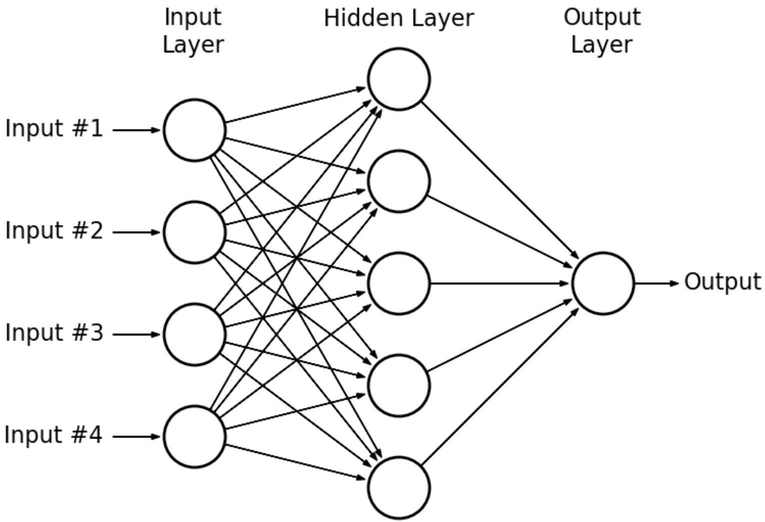


Image taken from `http://en.wikipedia.org/wiki/File:Blausen_0657_MultipolarNeuron.png`.

# Neural network: artificial


Input
Layer
Hidden Layer
Output
Layer
Input #1
Input #2
Input #3
Input #4
Output