Introduction to recursive machine learning

Inference

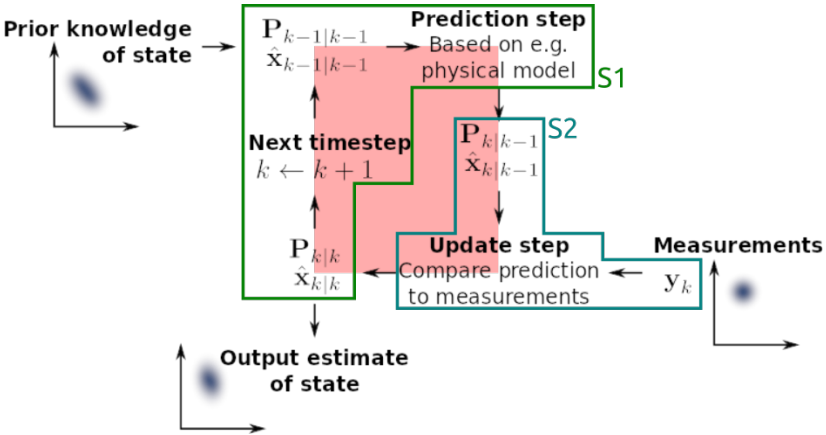
Juan Pablo Carbajal
juanpablo.carbajal@ost.ch

Eastern Switzerland University of Applied Sciences OST
Institute for Energy Technology IET
Scientific Computing and Engineering Group SCE

January 2023 - Rapperswil

Recap and Q&A

Sessions structure



Kalman filter: prediction step

Focus

The iteration represents a dynamical model of a process:

$$\boldsymbol{x}_k = \mathbf{A}\boldsymbol{x}_{k-1} + \boldsymbol{\epsilon}_k \quad \boldsymbol{\epsilon}_k \sim \mathcal{N}(0, \mathbf{Q}_k)$$
$$\boldsymbol{y}_k = \mathbf{H}\boldsymbol{x}_k + \boldsymbol{\rho}_k \quad \boldsymbol{\rho}_k \sim \mathcal{N}(0, \mathbf{R}_k)$$

Mean and variance of forward predictions (prediction step)

$$\boldsymbol{m}_k = \mathbf{A}\boldsymbol{m}_{k-1} \qquad \mathbf{P}_k = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^\top + \mathbf{Q}_{k-1}$$

This slide summarizes the modeling and prediction step of the KF.
How do the predicted measurements look like?

Data driven models

What's a data driven model?

Data-driven as complementary to mechanistic. The model does not intentionally exploit prior knowledge about the **data generating process**.

If prior knowledge exists it's desirable to consider it, but:

- Too much work
- Ignorance of methods

In science, efforts should be taken to exploit the prior knowledge.

There **are not** intrinsic data-driven models.

Linear regression can be mechanistic in some context, e.g. for an ideal gas at constant volume, pressure and temperature are linearly related:

$$PV = nRT$$

What prior knowledge do you have about

- falling objects?
- the motion of planets in their orbits?
- the behavior of bacteria?
- the way people like/dislike movies?

Mention a problem for which you have loads of prior knowledge.

Linear regression

model : $y_k = t_k a + b + \epsilon_k$

Batch

$$\mathbf{D} = \begin{bmatrix} t_1 & 1 \\ \vdots & \\ t_N & 1 \end{bmatrix}, \quad \boldsymbol{y}^\top = \begin{bmatrix} y_1 & \cdots & y_N \end{bmatrix}$$

$$\mathbf{D} \begin{bmatrix} a \\ b \end{bmatrix} = \boldsymbol{y} + \boldsymbol{\epsilon}$$

Recursive (assumes ordered correlates)

$$\boldsymbol{x}_k = \begin{bmatrix} a \\ b \end{bmatrix}$$

$$\mathbf{A} = \mathbf{I}, \quad \mathbf{Q} = \mathbf{0}$$

$$\mathbf{H} = \begin{bmatrix} \Delta t & 1 \end{bmatrix}, \quad \mathbf{R} = \boldsymbol{\Sigma}_\epsilon$$

What would be the batch and recursive form of $y_k = t_k a_k + b + \epsilon_k$?
What would be the measurement model if correlates are not ordered?
How can you get rid of explicit t_k in the model? (t_k is in principle unbounded!)

Linear regression

model : $y_k = t_k a + b + \epsilon_k$

Batch

$$\mathbf{D} = \begin{bmatrix} t_1 & 1 \\ \vdots & \\ t_N & 1 \end{bmatrix}, \quad \boldsymbol{y}^\top = \begin{bmatrix} y_1 & \cdots & y_N \end{bmatrix}$$

$$\mathbf{D} \begin{bmatrix} a \\ b \end{bmatrix} = \boldsymbol{y} + \boldsymbol{\epsilon}$$

Recursive (assumes ordered correlates)

$$\boldsymbol{x}_k = \begin{bmatrix} a \\ b \end{bmatrix}$$

$$\mathbf{A} = \mathbf{I}, \quad \mathbf{Q} \neq \mathbf{0}$$

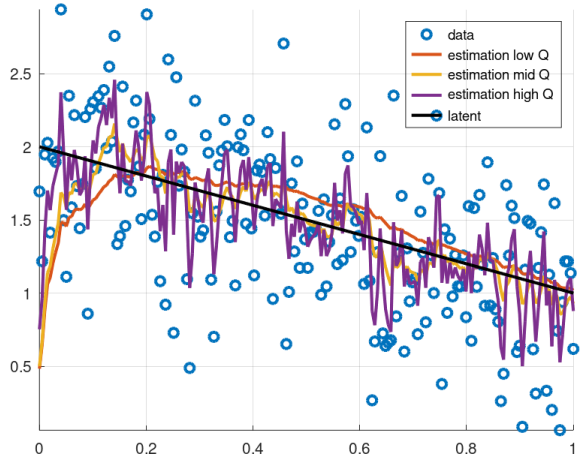
$$\mathbf{H} = \begin{bmatrix} \Delta t & 1 \end{bmatrix}, \quad \mathbf{R} = \boldsymbol{\Sigma}_\epsilon$$

See `s_filtering_interactive.py` (time varying linear regression) and `kalman_linreg.ipynb`

The model without process noise does not update the initial values. By setting a symmetric non-negative process noise covariance the states update.

Linear regression

model : $y_k = t_k a + b + \epsilon_k$



source: *s_recursive_linreg.m*

Polynomial regression

model : $y_k = a_n t_k^n + \cdots + a_1 t_k + a_0 + \epsilon_k$

Batch

$$\mathbf{D} = \begin{bmatrix} t_1^n & \cdots & t_1 & 1 \\ & \vdots & & \\ t_N^2 & \cdots & t_N & 1 \end{bmatrix}$$

$$\boldsymbol{y}^\top = \begin{bmatrix} y_1 & \cdots & y_N \end{bmatrix}$$

$$\mathbf{D} \begin{bmatrix} a \\ b \end{bmatrix} = \boldsymbol{y} + \boldsymbol{\epsilon}$$

Recursive

$$\boldsymbol{x}_k = \boldsymbol{a}$$
$$\mathbf{A} = \mathbf{I}$$
$$\mathbf{Q} = \mathbf{0} \text{ (fixed)}, \quad \mathbf{Q} \neq \mathbf{0} \text{ (adaptive)}$$
$$\mathbf{H}_k = \begin{bmatrix} t_k^n & \cdots & t_k & 1 \end{bmatrix}$$
$$\mathbf{R} = \boldsymbol{\Sigma}_\epsilon$$

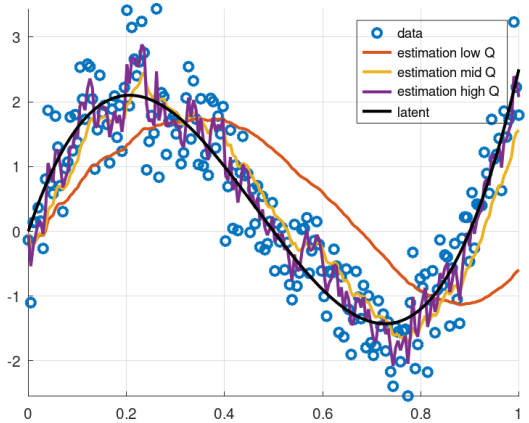
What is the issue with this measurement matrix? How would you avoid this issue?

$$\boldsymbol{x} = \begin{bmatrix} a_0 \\ \vdots \\ a_n \end{bmatrix}$$
$$\mathbf{A} = \mathbf{I} + \Delta t \mathbf{C}, \quad C_{ij} = \delta_{i(j+1)}$$

See `s_polyreg.py`

Polynomial regression

model : $y_k = a_n t_k^n + \cdots + a_1 t_k + a_0 + \epsilon_k$



source: *s_recursive_linreg.m*

Delayed regression

model : $y_k = w_1 y_{k-1} + \cdots + w_d y_{k-d} + \epsilon_k$ (auto-regressive model: AR)

Batch

$$\mathbf{D} = \begin{bmatrix} y_{d+1-1} & \cdots & y_1 \\ & \vdots & \\ y_{N-1} & \cdots & y_{N-d} \end{bmatrix}$$

$$\boldsymbol{y}^\top = \begin{bmatrix} y_0 & \cdots & y_N \end{bmatrix}$$

$$\mathbf{D}\boldsymbol{w} = \boldsymbol{y}_{(d+1):N} + \boldsymbol{\epsilon}$$

Recursive

$$\boldsymbol{x}_k = \boldsymbol{w}_k$$
$$\mathbf{A} = \mathbf{I}$$
$$\mathbf{Q} = \mathbf{0} \text{ (fixed)}, \quad \mathbf{Q} \neq \mathbf{0} \text{ (adaptive)}$$
$$\mathbf{H}_k = \begin{bmatrix} y_{k-1} & \cdots & y_{k-d} \end{bmatrix}$$
$$\mathbf{R} = \boldsymbol{\Sigma}_\epsilon$$

The rows of the design matrix are moving windows.

Extreme learning machines

Basis function decomposition

Any function of the independent variable can be put in the design matrix:

$$\mathbf{D}_{k:} = \begin{bmatrix} \phi_n(t_k) & \cdots & \phi_1(t_k) \end{bmatrix}$$

which corresponds to the measurement matrix, i.e.

$$\mathbf{H}_k = \mathbf{D}_{k:}$$

combined with the drift dynamic model

$$\begin{aligned} \boldsymbol{x}_k &= \boldsymbol{w}_k \\ \mathbf{A} &= \mathbf{I}, \quad \mathbf{Q} \neq \mathbf{0} \end{aligned}$$

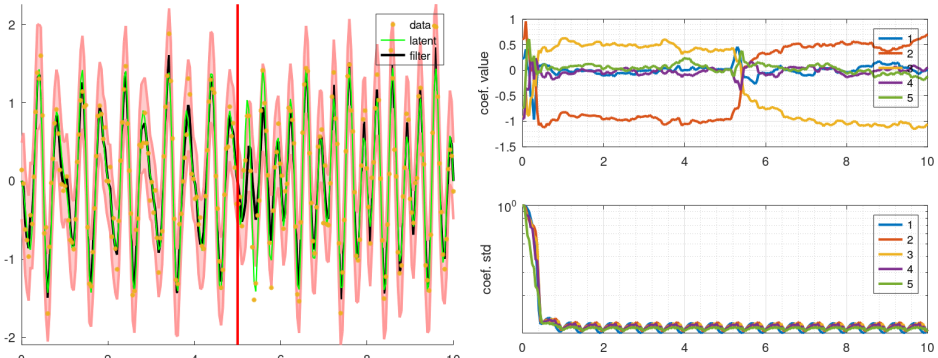
gives the recursive version.
Properties of the function set $\{\phi_i\}$ can be used to write a more stable model, e.g. polynomials.
See `s_fourier.m` and `s_fourier_adaptive.m` for frequency tracking.

In machine learning $\mathbf{H}_k$ is also known as a feature vector.
The only requisite of KF is that the models (dynamic and measurement) are linear in the states.

In the book Särkkä, S. (2013). Bayesian Filtering and Smoothing, this model is called **Linear-in-parameters regression model II**, described in example 3.2, page 41.

Extreme learning machines

Basis function decomposition



source: *s_fourier_adaptive.m*

The process noise allows the states to be adapted online. This means that the amplitude of each basis function is updated at each iteration. A quick change in the coefficients is picked up quickly (depending on process noise level).

Here the amplitudes of frequency 2 and 3 got exchanged.

The linear readout, the only part that is trained in RC, is replaced with a KF with the measurement model:

$$\hat{\boldsymbol{y}}_k = \mathbf{H}\boldsymbol{m}_k + \boldsymbol{\rho}_k \quad \boldsymbol{\rho}_k \sim \mathcal{N}(0, \mathbf{R})$$

The system structure is:

$$\left.\begin{array}{l} \boldsymbol{u}(t) = \mathscr{C}[\boldsymbol{x}(t)] \in \mathbb{R}^{\dim \boldsymbol{q}} \\ \mathscr{D}_{\boldsymbol{\lambda}}\boldsymbol{q}(t) = \boldsymbol{u}(t) \\ \boldsymbol{z}(t) = \mathscr{R}[\boldsymbol{q}(t)] \in \mathbb{R}^{\dim \boldsymbol{z}} \end{array}\right\} \text{Rerservoir}$$

$$\left.\begin{array}{l} \mathbf{H}_k = \boldsymbol{z}(t_k)^\top \\ \mathbf{A} = \mathbf{I} \\ \mathbf{Q} = \mathbf{0}\ (\text{fixed}), \quad \mathbf{Q} \neq \mathbf{0}\ (\text{adaptive}) \\ \mathbf{R} \neq \mathbf{0} \end{array}\right\} \text{Kalman filter}$$

I choose $Q = 0$ and the systems work well for a while, e.g. residuals $\hat{\boldsymbol{y}}_k - \boldsymbol{y}_k$ are acceptable. But then residuals start growing. What can I say?

Mechanistic models

Multiple Object tracking in 1D

Refer to `tracking_example.py`.

- For more than 2 targets: position information
- For 2 targets: position of 1st, relative position of 2nd w.r.t. 1st

The "true" motion of target n (with mass m_n) is given by:

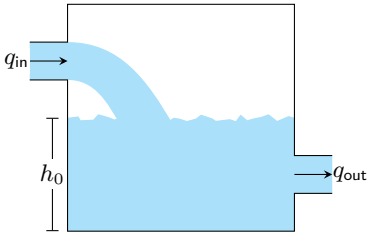
$$\begin{bmatrix} \dot{p}_n \\ \dot{v}_n \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & \frac{1}{m_n} \end{bmatrix} \begin{bmatrix} p \\ \dot{v} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m_n} \end{bmatrix} f(t)$$

where $f(t)$ is a chaotic force. It is modelled with:

$$\begin{bmatrix} \dot{\tilde{p}}_n \\ \dot{\tilde{v}}_n \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & \frac{1}{m_n} \end{bmatrix} \begin{bmatrix} \tilde{p} \\ \tilde{v} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \eta(t), \quad \eta(t) \sim \mathcal{N}(0, \sigma_\eta^2)$$

What would be the model if you weant to track an harmonic oscillator (spring-mass system, RLC-circuit, etc.)?

Water tank



The water level in the tank is given by: $\dot{h} = q_{\text{in}}(t) - q_{\text{out}}(t)$
Discharge is a function of the water level: $q_{\text{out}}(h(t)) = Ch^s$, e.g. $s = \frac{1}{2}$.
For any $s \neq 1$, it is a nonlinear system.
For small variations of the level $h(t) = h_o + \Delta h(t)$:

$$\begin{aligned} q_{\text{out}}(h(t)) &= Ch(t)^s \simeq C\left[h_o^s + sh_o^{s-1}(h(t) - h_o)\right] \\ &= C(1-s)h_o^s + Csh_o^{s-1}h(t) := -C_1(h_o) - C_2(h_o)h \\ \dot{h} &= C_2(h_o)h + q_{\text{in}}(t) + C_1(h_o) \end{aligned}$$

The linearization adds a constant (negative) term to the input.

$$C_1(h_o) = C(s-1)h_o^s$$
$$C_2(h_o) = Csh_o^{s-1}$$

Are they correlated?

$$C_2(h_o)h_o\frac{s-1}{s} = C_1(h_o)$$

Water tank

Input modelling

Linearized dynamics

$$\dot{h} = C_2(h_o)h + q_{\text{in}}(t) + C_1(h_o)$$

The input has a known contribution $q(t)$, and an unknown extra (small) inflow $u(t)$:

$$q_{\text{in}}(t) = q(t) + u(t)$$

We will model it with :

$$\begin{bmatrix} \dot{h} \\ \dot{\delta}_{C_1} \\ \dot{\boldsymbol{u}} \end{bmatrix} = \begin{bmatrix} C_2(h_o) & 1 & \boldsymbol{U}(t) \\ 0 & 0 & \mathbf{0} \end{bmatrix} \begin{bmatrix} h \\ \delta_{C_1} \\ \boldsymbol{u} \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ \mathbf{0} \end{bmatrix} [q(t) + C_1(h_o)] + \mathbf{L}\boldsymbol{\eta}(t)$$

$$\dot{\boldsymbol{x}} = \mathbf{F}(t)\boldsymbol{x} + \mathbf{B}\tilde{q}(t) + \mathbf{L}\boldsymbol{\eta}(t), \quad \boldsymbol{\eta}(t) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{\eta}})$$

where $\boldsymbol{U}(t)$ is a feature vector (a given set of time-dependent functions), and $\boldsymbol{u}$ their coefficients. δ_{C_1} is a correction to the constant $C_1(h_o)$ produced by the linearization.

The unknown inflow is modelled as

$$\tilde{u}(t) = \sum_{n=1}^{N} u_n U_n(t)$$

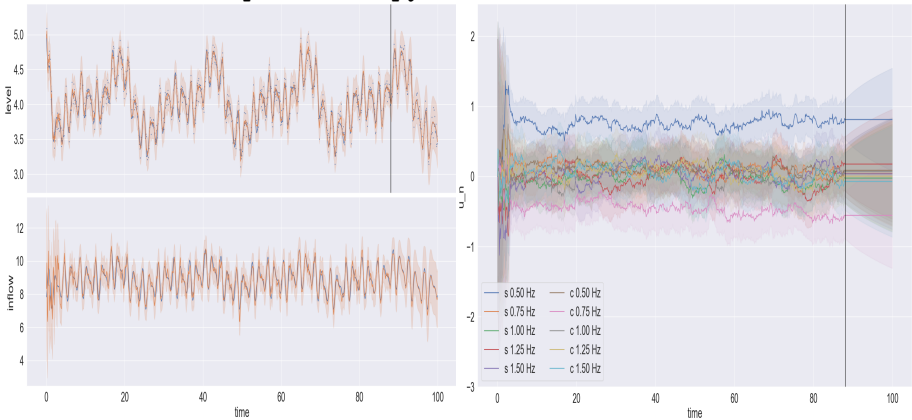
and the known inflow is combined with the constant produced by the linearization

$$\tilde{q}(t) = q(t) + C_1(h_o)$$

Water tank

Example

Refer to `tank_example_basis.py`



Parameter estimation I

Augmented state

Augment your state with the parameters with constant noisy dynamics (drift model)

$$^{\text{extra}}x_k = {}^{\text{extra}}x_{k-1} + \epsilon_k, \quad \epsilon_k \sim \mathcal{N}(0, \sigma^2_{\text{extra}})$$

Works out-of-the-box if resulting model is linear in parameters, i.e. if we have in the dynamics a term of the form ax_{k-1}, adding a to the state will render the model non-linear.
σ_{extra} controls how "reactive" or "nervous" the parameter is. Larger values, quicker adaptation, larger confidence intervals.
See `s_fourier_adaptive.m`

Frequently there is a work-around to the "non-linearization" problem, imagination is the limit.

Batch regression of historical data

Functional data analysis

If you have historical data (not growing, not online) batch analyses could be performed.
Consider data $\{(t_i, \boldsymbol{x}_i)\}_{i=1}^{N}$
The model

$$\boldsymbol{x}_k = \mathbf{A}\boldsymbol{x}_{k-1}$$

can be learned from the data using linear regression.
For an ODE model

$$\dot{\boldsymbol{x}} = \mathbf{B}\boldsymbol{x}$$

Estimate the time derivative $\dot{\boldsymbol{x}}$ from the data, then do linear regression.
Nonlinear models can also be learned this way.
Once the dynamics is identified by a batch method, apply the KF formalism and continue recursively.

There is plenty of methods of learning dynamical systems from data, sometimes called "data-driven dynamical systems" or "system identification", and probably by other names depending on the community.
Some references:

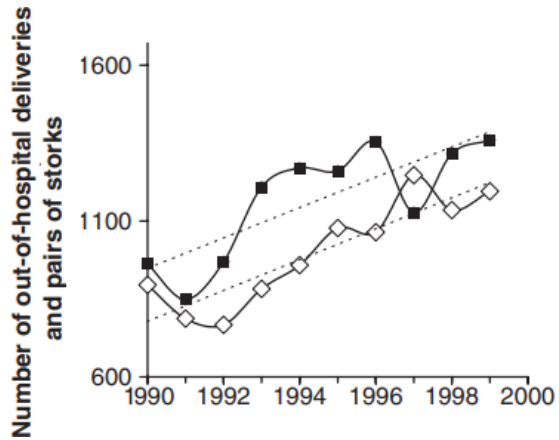
- Functional Data Analysis (2005) by J. O. Ramsay and B. W. Silverman
- Gaussian Processes for Machine Learning (2006) by Carl Edward Rasmussen and Christopher K. I. Williams
- Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control (2019) by Steven L. Brunton and J. Nathan Kutz

Coffee break: 15 minutes

Stochastic modeling

We will now digress to more general concepts related to statistical learning (modern machine learning). We will get used to talk about *conditional probabilities*, and understand what they mean. This will leads us to grasp the idea of *inference* from data. After this zoom out, we will come back to the KF to understand the update step, which is what makes the model reactive to incoming data.

Observed correlations

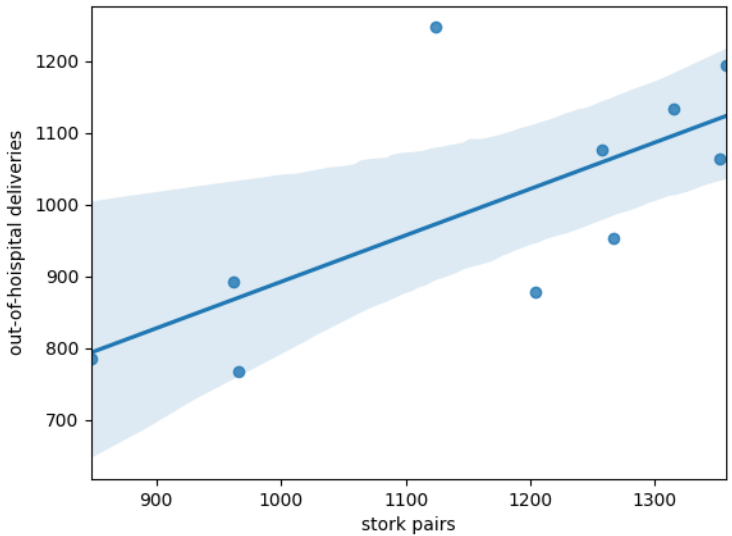


source: *New evidence for the theory of the stork. DOI: 10.1111/j.1365-3016.2003.00534.x*

More at `https://www.tylervigen.com/spurious-correlations`

It is well established that the number of newborns and stork couples is robustly correlated for data from European countries. The plot shows this data for several years. Filled symbols are stork couples and empty symbols baby deliveries out of hospitals.

Observed correlations



The plots shows the two variables plotted jointly (one vs. the other). There seems to be a pattern in the pairs of values: a relation between the two variables. What does it mean?

- Can I guess one variable if I know the other?
- Will one variable change if I forcibly change the other?

Compare this plot to a functional plot like the ones used in Calculus (even if you include noise).

Correlation and Causation

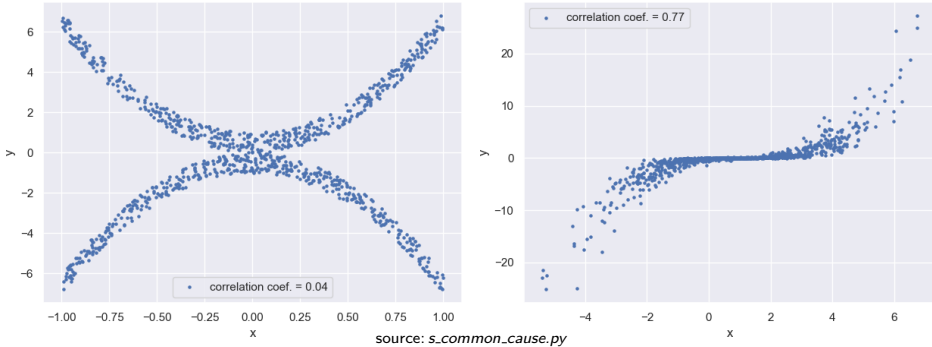
Correlation doesn't imply causation

causal, no correlation

$$x \sim \mathcal{U}(-1, 1)$$
$$y \leftarrow a\frac{\epsilon}{|\epsilon|}x^2 + \epsilon \quad \epsilon \sim \mathcal{U}(-1, 1)$$

not causal, correlated

$$c \sim \mathcal{N}(0, 1)$$
$$x \sim \mathcal{N}(2c - 1, 0.2)$$
$$y \sim \mathcal{N}(c^3, 0.1)$$

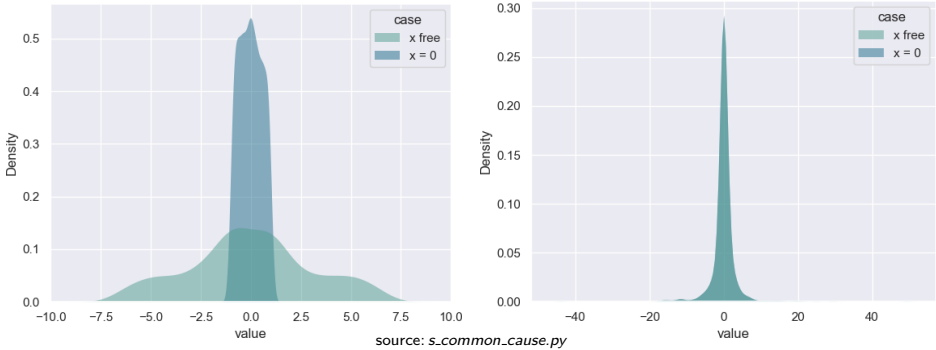


source: *s_common_cause.py*

X vs. Y plots have different interpretations. On the left panel we see a "functional" or "causal" plot, in which changing the values of x will affect the values of y. This is because the variables are connected by a function, y being a function of x. Note that the correlation, however, is negligible. On the right panel we see a plot that would seem to also be "causal", saying that values of x affect y, however if we inspect the process generating the data we see that x does not affect y. The variables are correlated and are statistically dependent, but they are not causally dependent.

$$x \leftarrow 0 \sim \mathcal{U}(-1, 1)$$
$$y \leftarrow a\frac{\epsilon}{|\epsilon|}x^2 + \epsilon \quad \epsilon \sim \mathcal{U}(-1, 1)$$

$$c \sim \mathcal{N}(0, 1)$$
$$x \leftarrow 0 \sim \mathcal{N}(2c - 1, 0.2)$$
$$y \sim \mathcal{N}(c^3, 0.1)$$

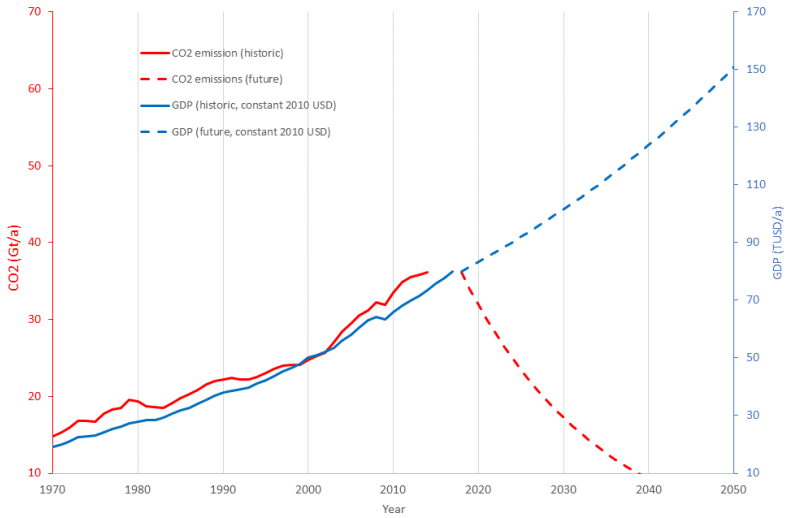


source: *s_common_cause.py*

The situation becomes evident when we intervene the values of x. That is, we make an experiment and set x to a value (in this case $x \leftarrow 0$).
In the right panel, it is evident that the distribution of y after the intervention is the same as before the intervention, x wasn't affecting y, although they were correlated. In the left panel however, the intervention on x radically changed the distribution of y.
These types of distributions are called interventional distributions. We will see that they are conceptually different from conditional distributions.
For more information refer to

- Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. 2017. Elements of Causal Inference: Foundations and Learning Algorithms. The MIT Press.
- Judea Pearl, and Dana Mackenzie. 2019. The Book of Why. Harlow, England: Penguin Books.

Correlation or Causation?



source: *https://nordborg.ch/2018/08/12/sustainable-growth-is-an-oxymoron/*

Historically, CO2 emissions have been strongly correlated with how much money we have. The wealthier we are, the more CO2 we emit. To increase wealth, we use more energy, which often comes from burning fossil fuels and release more CO2. Also, wealthier societies have higher energy requirements (e.g. comfort).

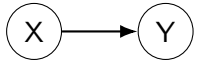
Are these explanations describing causal relations?

Other sources `https://ourworldindata.org/co2-gdp-decoupling`

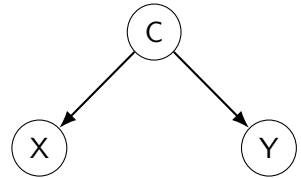
Graphical models

Graphic models are a tool to visualize dependencies between (random) variables. The nodes of a graph represent the variables and arrows indicate direction of influence.

$$x \sim \mathcal{U}(-1, 1)$$
$$y \leftarrow a\frac{\epsilon}{|\epsilon|}x^2 + \epsilon \quad \epsilon \sim \mathcal{U}(-1, 1)$$



$$c \sim \mathcal{N}(0, 1)$$
$$x \sim \mathcal{N}(2c - 1, 0.2)$$
$$y \sim \mathcal{N}(c^3, 0.1)$$

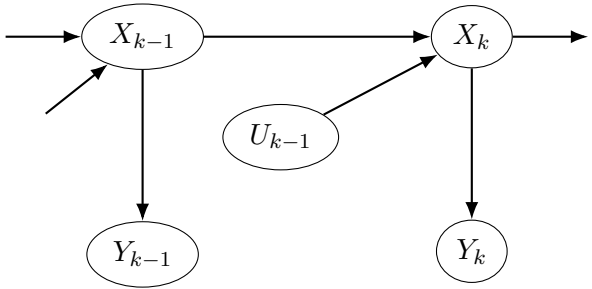


There are also undirected graphical models. See `https://en.wikipedia.org/wiki/Graphical_model` for an overview.

1. Draw a graphical model for the GDP-CO2 relation. Add new variables if you need them.
2. Can you draw a different graphical model?

For an iterated map

$$\boldsymbol{x}_k = \mathbf{A}\boldsymbol{x}_{k-1} + \mathbf{B}\boldsymbol{u}_{k-1} + \boldsymbol{\epsilon}_x$$
$$\boldsymbol{y}_k = \mathbf{H}\boldsymbol{x}_k + \boldsymbol{\epsilon}_y$$



(Conditional) Probability

The night incident

source: *Luis Prade and Gan Khoon Lay at https://thenounproject.com*

A dark night, a policeman walks down an apparently deserted street. Suddenly burglar alarm. Across the street a jewelry store with a broken window. A masked man crawls out of the broken window, carrying a bag which turns out to be full of expensive jewelry. The policeman doesn't hesitate at all in deciding that this man is dishonest.

Is guilt implied by evidence?

Is it a logical deduction? Alternative hypothesis?
Is there validity in the reasoning?

Propositional logic

State several propositions and determine their truth value. Does the value depend on the context?

Works with propositions (A, B, etc.) that can be true or false.

$A \equiv$ It rains at 10:00

$B \equiv$ We see a blue sky

Logic

- negation $\neg A$: true if A is false
- conjunction $A \wedge B$: true if A and B are both true
- disjunction $A \vee B$: true if any of (or both) A or B are true
- implication $A \implies B$: says that $A \wedge \neg B$ is false ($\neg A \vee B$ is true)

Boolean

- $\bar{A}$
- AB
- $A + B$
- $A \implies B$

Propositional logic

Works with propositions (A, B, etc.) that can be true or false.

$A \equiv$ It rains at 10:00

$B \equiv$ We see a blue sky

Boolean	Probability
• $\bar{A}$	• $p(\bar{A}) = 1 - p(A)$
• AB	• $p(AB) = p(A)p(B)$ (independent)
• $A + B$	• $p(A \vee B) = p(A) + p(B) - p(AB)$
• $A \implies B$	• $p(B\|A) = p(AB)/p(A)$

Implication (logical)

$A \implies B$: says that $A\bar{B}$ is false ($\bar{A} + B$ is true).
Means **only that** $A = AB$ (the truth value of A is the same as the truth value of AB).

A	$\implies$	B	AB
T	T	T	T (✓)
T	F	F	F
F	T	T	F (✓)
F	T	F	F (✓)

- All true propositions logically imply all other true propositions.
- It doesn't mean: B deducible from A. This depends on the background information.
- It doesn't mean: B is caused by A. Causal physical consequence can be effective only at a later time. The rain at 10:00 is not the physical cause of the clouds before 10:00. Implication doesn't follow the uncertain causal direction clouds→rain, but rather the certain non-causal rain→clouds.

State several true propositions and state their implication. Verify that $A\bar{B}$ is false in all cases. Try to state weird or funny implications? E.g. the day sky is blue, implies that chickens lay eggs.

A $\equiv$ the day sky is blue →True

B $\equiv$ chickens lay eggs →True

$A\bar{B}$ $\equiv$ the day sky is blue and chickens do not lay eggs →False

The night incident

source: *Luis Prade and Gan Khoon Lay at https://thenounproject.com*

A dark night, a policeman walks down an apparently deserted street. Suddenly burglar alarm. Across the street a jewelry store with a broken window. A masked man crawls out of the broken window, carrying a bag which turns out to be full of expensive jewelry. The policeman doesn't hesitate at all in deciding that this man is dishonest.

Is guilt implied by evidence?

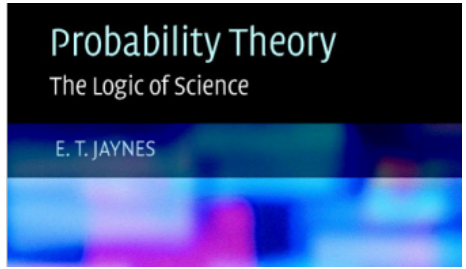
Is it a logical deduction? Alternative hypothesis?
Is there validity in the reasoning?

Probability

Maxwell and Jaynes

The actual science of logic is conversant at present only with things either certain, impossible, or entirely doubtful, none of which (fortunately) we have to reason on. Therefore the true logic for this world is the calculus of Probabilities, which takes account of the magnitude of the probability which is, or ought to be, in a reasonable man's mind.

James Clerk Maxwell (1850)



Probability is a numerical between value assigned to a plausibility. Impossibility is assigned the probability 0 and certainty is assigned the probability 1.

We will see how Bayesian probability includes classical logic as a particular case and extends it to include reasoning on plausible events.

Conditional probability

Definition

The plausibility (probability) of a proposition A considering that proposition B is true:

$$p(A|B) = \frac{p(AB)}{p(B)}$$

The plausibility (probability) of a proposition A depends, in general, on whether we known/assume/believe some other proposition B is true.
Example:

$A \equiv$ My English pronunciation is perfect

$B \equiv$ I grew up in Argentina

$C \equiv$ I grew up in England

which is bigger? $p(A|B)$ or $p(A|C)$

All probabilities are conditional probabilities.

All probabilities are conditional probabilities, because there is always some assumptions we made or knowledge we have that is relevant for the assignation.
Think of "The probability of a 3 in a fair die is $\frac{1}{6}$". What's the background information?

Product rule

Plausibility (probability)

To evaluate $p(AB|C)$ we can proceed as follows:

1. determine the probability of A true given information C.
   $\rightarrow p(A|C)$
2. based on that, determine the probability of B true.
   $\rightarrow p(B|AC)$

1. determine the probability of B true given information C.
   $\rightarrow p(B|C)$
2. based on that, determine the probability of A true.
   $\rightarrow p(A|BC)$

Both paths should give the same

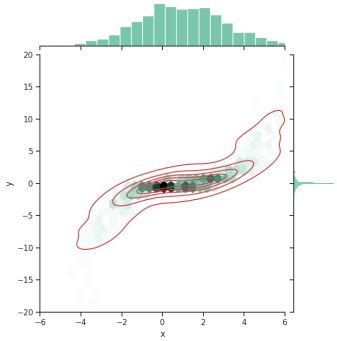
$$p(AB|C) = p(A|BC) \overbrace{p(B|C)}^{\text{"prior" to knowing } A} = p(B|AC) \overbrace{p(A|C)}^{\text{"prior" to knowing } B}$$

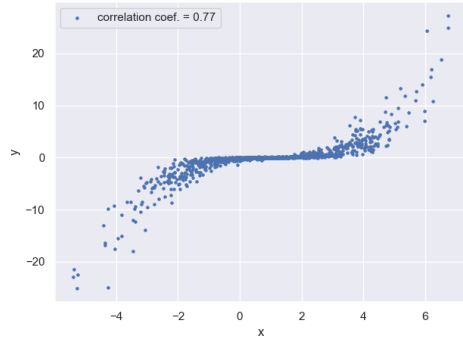
- $p(AB|C)$ is called the joint probability of A and B. The probability that both propositions are true (given background information C).
- There is nothing special in the "prior", it is just the probability we would assign "before" we knew the other piece of information. Here "before" is a temporal expression but it might have nothing to do with time.
- Visually explained
  `https://setosa.io/ev/conditional-probability/`

Distribution of Y for known X

$$Y \simeq f(X) + \epsilon, \quad \epsilon \sim \mathcal{D}$$

where f is a deterministic relation (without randomness)





- The left plot shows how the values of X and Y are jointly distributed. The right plot is an approximation of the joint probability density fuction.
- The mathematical relation stated is not necessarily causal, it could be a model for the data in the plot **inspired** by the observed correlation. That's why I used $\simeq$, just to make this explicit.

If I observed a value of X, say $X = x$, what is the distribution of Y?

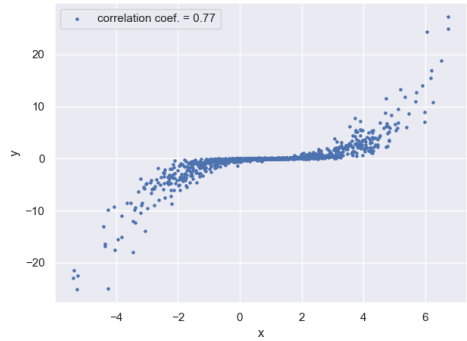
Distribution of Y for known X

$$Y \simeq f(X) + \epsilon, \quad \epsilon \sim \mathcal{D}$$

where f is a deterministic relation (without randomness)
If we observe (or known that) $X = x$, then

$$Y|_{X=x} \simeq f(x) + \epsilon$$

For $X = x$, Y is distributed like ϵ plus a shift given by $f(x)$.



The resulting distribution is called the conditional distribution of Y given $X = x$. If we take x as a variable (instead of a value) then we have the conditional distribution as a function of x.

We use a symbol for this concept, usually, $p(Y|x)$, which means: $p(Y|X = x)$ for different values of x.

The symbol p expresses an idea. It will be generally associated with a function: the probability (density) function of a (continuous) variable. For example, for a Gaussian variable we write $Z \sim \mathcal{N}(\mu, \sigma^2)$ or

$$p(Z = z) \text{ or } p(z) \equiv \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{z-\mu}{\sigma}\right)^2}$$

To actually compute that distribution we need the values of μ and σ, hence more correctly it should be written as $p(Z = z|\mu\,\sigma)$.

Sum rule

Plausibility (probability)

$$p(A|C) + p(\bar{A}|C) = 1 \;\rightarrow\; \sum_i p(A_i|C) = 1 \;\rightarrow\; \int_{-\infty}^{\infty} p(A = a|C)\mathrm{d}a = 1$$

States that the probability distributes totally over the plausible propositions.
It's a particular case of (using $B = \bar{A}$):

$$p(A + B|C) = p(A|C) + p(B|C) - p(AB|C)$$

Syllogisms

Assume that $A \implies B$ is true.

Strong

- if A is true, then B is true
  - $A \equiv$ rain at 10:00, $B \equiv$ clouds before 10:00
  - $A \equiv$ man is burglar, $B \equiv$ robbed store
- if B is false, then A is false
  - $\bar{B} \equiv$ no clouds before 10:00, $\bar{A} \equiv$ no rain at 10:00
  - $\bar{B} \equiv$ didn't rob store, $\bar{A} \equiv$ man isn't burglar

Weak

- if B is true, then A is more plausible Evidence doesn't prove A, verification of its consequence gives more confidence in A
  - $B \equiv$ clouds before 10:00, $A \equiv$ rain at 10:00
  - $B \equiv$ robbed store, $A \equiv$ man is burglar
- if A is false, then B less plausible Evidence doesn't disprove B, one of the reasons eliminated, less confidence in B
  - $\bar{A} \equiv$ no rain at 10:00, $B \equiv$ clouds before 10:00
  - $\bar{A} \equiv$ man isn't burglar, $B \equiv$ robbed store

The implication does not imply causation or deduction. The examples are just for illustration of the syllogisms.
This is the table correspoding to the assumption "$A \implies B$ is true".

A	$\implies$	B	AB
T	T	T	T
F	T	T	F
F	T	F	F

Syllogisms

Assume that $A \implies B$ is true.
$C \equiv A \implies B$

Strong

- if A is true, then B is true

$$p(B|AC) = \frac{p(AB|C)}{p(A|C)} = \frac{p(A|C)}{p(A|C)} = 1$$

- if B is false, then A is false

$$p(A|\bar{B}C) = \frac{\overbrace{p(A\bar{B}|C)}^{=0}}{p(\bar{B}|C)} = 0$$

Weak

- if B is true, then A is more plausible

$$p(A|BC) \geq p(A|C)$$

$$p(A|BC) = \underbrace{\frac{\overbrace{p(B|AC)}^{=1}}{p(B|C)}}_{\geq 1} p(A|C)$$

- if A is false, then B less plausible

$$p(B|\bar{A}C) \leq p(B|C)$$

$$\frac{p(\bar{A}|BC)}{p(\bar{A}|C)} \leq 1$$

The implication does not imply causation or deduction. The examples are just for illustration of the syllogisms.
This is the table correspoding to the assumption "$A \implies B$ is true".

A	$\implies$	B	AB
T	T	T	T
F	T	T	F
F	T	F	F

Note that in the first syllogism

$$p(\bar{A}|BC) \leq p(\bar{A}|C)$$

because

$$p(\bar{A}|BC) = 1 - p(A|BC)$$

Weaker sillogism

Assume that "if A is true, then B is more plausible" is true.
$\rightarrow$if B is true, then A more plausible: $p(A|BC) \geq p(A|C)$

In other words:
Assume that if "a man is robbing a jewelry store" (A) it is more plausible that he will be "found in the suspicious situation" (B).
$\rightarrow$If he is actually found (B true), then it is more plausible that he is robbing the store.

We assume that $p(B|AC) \geq p(B|C)$ is true. The product rule says:

$$p(B|AC) = \frac{p(A|BC)}{p(A|C)} p(B|C)$$

If the assumption is true, then it must be that:

$$\frac{p(A|BC)}{p(A|C)} \geq 1 \rightarrow p(A|BC) \geq p(A|C)$$

Note:
If a proposition is independent of all other propositions, then conditioning on it (knowing whether it is true), doesn't change the plausibility of the other propositions:

$$\begin{aligned} p(AB|C) &= p(A|BC)p(B|C) \\ &= p(B|AC)p(A|C) \\ \text{if } p(B|AC) &= p(B|C) \text{ (independence)} \\ \rightarrow p(A|BC)p(B|C) &= p(B|C)p(A|C) \\ \rightarrow p(A|BC) &= p(A|C) \end{aligned}$$

Lunch: 1 hour

Some probability problems

Two die

Consider the events D_1 and D_2 as the number shown by two fair 3-faces die, with face values 1, 2, and 3.

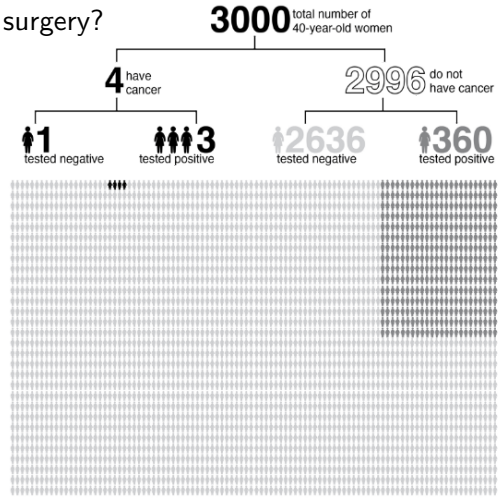
1. List all possible outcomes
2. Write a program to generate data from this process
3. What's the distribution of $D_1 + D_2$?
4. If somebody tell us that they rolled $D_1 = 1$, what's the distribution of $D_1 + D_2$?
5. If somebody tell us that they rolled $D_1 + D_2 = 4$, what's the joint distribution of D_1 and D_2?
6. If somebody tell us that they rolled $D_1 + D_2 \leq 3$, what's the distribution of D_1?

Note:

$$P(D_1 D_2 | D_1 + D_2) P(D_1 + D_2) = P(D_1 + D_2 | D_1 D_2) P(D_1 D_2)$$

Breast cancer

A forty-year-old woman gets a positive mammogram for breast cancer. Should she have surgery?

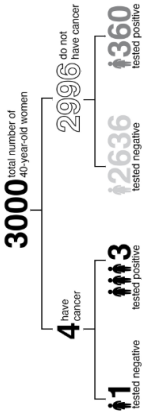


source: *Pearl J. The book of why. Figure 3.3*

$D \equiv$ has cancer ("disease"). $T \equiv$ mammogram result ("test"). $P(D|T)$?

Breast cancer

$D \equiv$ has cancer ("disease"). $T \equiv$ mammogram result ("test"). $P(D|T)$?



$$\overbrace{P(D|T)}^{\text{updated prob. of } D} = \frac{\overbrace{P(T|D)}^{\text{likelihood ratio}}}{P(T)} \overbrace{P(D)}^{\text{prior prob. of } D}$$

$$P(T) = P(T|D)P(D) + P(T|\bar{D})P(\bar{D}) = 0.121$$

$$P(D) = \frac{4}{3000} \quad P(\bar{D}) = \frac{2996}{3000}$$

$$P(T|D) = \frac{3}{4} \quad P(T|\bar{D}) = \frac{360}{2996}$$

$$\frac{P(T|D)}{P(T)} = \frac{0.75}{0.121} \simeq 6.2$$

$$\rightarrow P(D|T) \simeq \frac{24}{3000} = 0.008 \sim 1\%$$

All probabilities are computed from the frequency data in the graph.

$$P(T) = \sum_{x \in \text{possible values of D}} P(T\ (D{=}x)) \quad \text{marginal}$$

$$P(T\ (D{=}x)) = P(T|D{=}x)P(D{=}x)$$

$$P(T) = \sum_{x \in \text{possible values of D}} P(T|D{=}x)P(D{=}x)$$

Breast cancer

$D \equiv$ has cancer ("disease"). $T \equiv$ mammogram result ("test"). $P(D|T)$?
$Z \equiv$ genetic risk for breast cancer: $P(D|Z) = \frac{1}{20}$
Knowing that $P(T|DZ) = P(T|D)$

$$\frac{P(T|DZ)}{P(T|Z)} \simeq 5 \rightarrow P(D|TZ) \simeq \frac{1}{4} \sim 25\%$$

If the prior of having the desease changes, so does the posterior!
This test is insensitive to the genetic of the patient.

King's pardon

The three prisoners problem

You Y are a royal prisoner with two other: A, B. The king executes only one of you: X_Y, X_A, X_B. A honest guard is allowed to tell you if A is pardoned (G_A) or B (G_B), but nothing about you. Do you ask the guard? Is $P(X_Y|G_A)$ different than $P(X_Y) = 1/3$?

```
import numpy as np
import pandas as pd
K = rng.choice(['Xa','Xb','Xy'], size=100000)  # King's choice
G = []  # What the guard says
for x in K:
  if x == 'Xa':  # a is executed, guard says b is pardoned
    G.append('Gb')
  elif x == 'Xb':  # b is executed, guard says a is pardoned
    G.append('Ga')
  elif x == 'Xy':  # you are executed, guard says a or b is pardoned
    G.append(rng.choice(['Ga','Gb']))

G = pd.Series(G, name="Guard"); K = pd.Series(K, name="King")
P = pd.crosstab(G, K, normalize=True, margins=True)
P.rename(index={"All": "P(X)"}, columns={"All":"P(G)"}, inplace=True)
P_Ga = P.loc["Ga", "P(G)"]
P_Xy_and_Ga = P.loc["Ga", "Xy"]
P_Xy_given_Ga = P_Xy_and_Ga / P_Ga
```

source: *s_kings_pardon.py*

King's pardon

Analysis

You Y are a royal prisoner with two other: A, B. The king executes only one of you: X_Y, X_A, X_B. A honest guard is allowed to tell you if A is pardoned (G_A) or B (G_B), but nothing about you. Do you ask the guard? Is $P(X_Y|G_A)$ different than $P(X_Y) = 1/3$?

Frequency analysis of data generated by program:

	X_A	X_B	X_Y	$P(G)$
G_A	0.00	0.33	0.17	0.5
G_B	0.33	0.00	0.17	0.5
$P(X)$	0.33	0.33	0.33	1.0

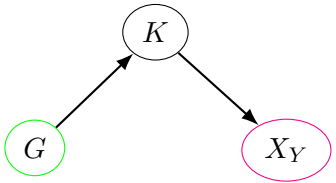
$$P(X_Y|G_A) = \frac{P(X_Y G_A)}{P(G_A)} \approx \frac{1}{3}$$

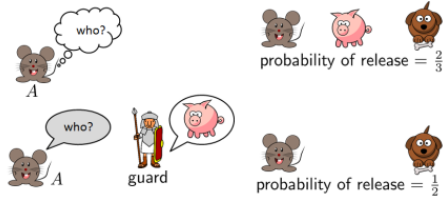
King's pardon

Analysis

You Y are a royal prisoner with two other: A, B. The king executes only one of you: X_Y, X_A, X_B. A honest guard is allowed to tell you if A is pardoned (G_A) or B (G_B), but nothing about you. Do you ask the guard? Is $P(X_Y|G_A)$ different than $P(X_Y) = 1/3$?

Most people reason causally:





For more information on causal graphs refer to

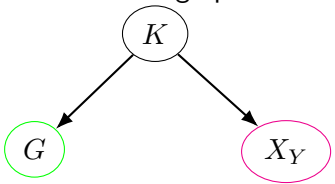
- Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. 2017. Elements of Causal Inference: Foundations and Learning Algorithms. The MIT Press.
- Judea Pearl, and Dana Mackenzie. 2019. The Book of Why. Harlow, England: Penguin Books.

King's pardon

Analysis

You Y are a royal prisoner with two other: A, B. The king executes only one of you: X_Y, X_A, X_B. A honest guard is allowed to tell you if A is pardoned (G_A) or B (G_B), but nothing about you. Do you ask the guard? Is $P(X_Y|G_A)$ different than $P(X_Y) = 1/3$?

Actual causal graph:



For more information on causal graphs refer to

- Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. 2017. Elements of Causal Inference: Foundations and Learning Algorithms. The MIT Press.
- Judea Pearl, and Dana Mackenzie. 2019. The Book of Why. Harlow, England: Penguin Books.

Let's make a deal

Shell game

You're given the choice of three shells. One shell hides a ball B, the others are empty. You pick a shell, say 1 ($Y = 1$), and the dealer, who knows what's below the shells, turns around one of the empty shells, say 3 ($D = 3$). He says to you, "Do you want to pick shell 2?" Is $P(B = 2|D = 3\,Y = 1)$ different than $P(B = 2) = 1/3$?

Is it to your advantage in general to switch your choice of shells?

Write a program to generate data from this process.

shell 1 ($Y = 1$)	shell 2	shell 3	Outcome if you switch
Ball	–	–	Lose
x	Ball	–	Win
x	–	Ball	Win

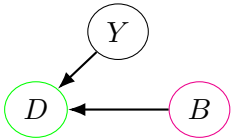
Let's make a deal

Shell game

You're given the choice of three shells. One shell hides a ball B, the others are empty. You pick a shell, say 1 ($Y = 1$), and the dealer, who knows what's below the shells, turns around one of the empty shells, say 3 ($D = 3$). He says to you, "Do you want to pick shell 2?" Is $P(B = 2|D = 3\,Y = 1)$ different than $P(B = 2) = 1/3$?

Is it to your advantage in general to switch your choice of shells?

Causal graph:



$Y \equiv$ your choice.
$D \equiv$ dealer's choice. $B \equiv$ location of the ball.

$$P(B|DY) = P(D|BY)\frac{P(B)}{P(D|Y)} \quad P(D|Y) = \sum_{B=1}^{3} P(DB|Y)$$

$$\begin{aligned} P(D = 3|Y = 1) &= P(D = 3\,B = 1|Y = 1) \quad (= 1/4) \\ &+ P(D = 3\,B = 2|Y = 1) \quad (= 1/4) \\ &+ P(D = 3\,B = 3|Y = 1) \quad (= 0 \text{ dealer doesn't show ball}) \\ &= 1/2 \end{aligned}$$

$Y = 1$						
B	1	1	2	3	3	2
D	2	3	3	2	1	1

$$\rightarrow \frac{P(B = 1, 2, 3)}{P(D = 3|Y = 1)} = \frac{2}{3}$$

$$P(B = 2|D = 3\,Y = 1) = P(D = 3|B = 2\,Y = 1)\frac{2}{3} = 1 \cdot \frac{2}{3}$$

$$P(B = 1|D = 3\,Y = 1) = P(D = 3|B = 1\,Y = 1)\frac{2}{3} = \frac{1}{2} \cdot \frac{2}{3}$$

Let's fake a deal

You're given the choice of three shells. One shell hides a ball B, the others are empty. You pick a shell, say 1 ($Y = 1$), and another player, who doesn't know what's below the shells, turns around another shell, say 3 ($O = 3$). The dealer says to you, "Do you want to pick shell 2?" Is $P(B = 2|O = 3\, Y = 1)$ different than $P(B = 2) = 1/3$

Is it to your advantage in general to switch your choice of shells?

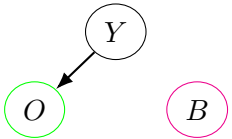
Write a program to generate data from this process.

Let's fake a deal

You're given the choice of three shells. One shell hides a ball B, the others are empty. You pick a shell, say 1 ($Y = 1$), and another player, who doesn't know what's below the shells, turns around another shell, say 3 ($O = 3$). The dealer says to you, "Do you want to pick shell 2?" Is $P(B = 2|O = 3\, Y = 1)$ different than $P(B = 2) = 1/3$

Is it to your advantage in general to switch your choice of shells?

Causal graph:



$Y \equiv$ your choice.
$O \equiv$ other's choice. $B \equiv$ location of the ball.

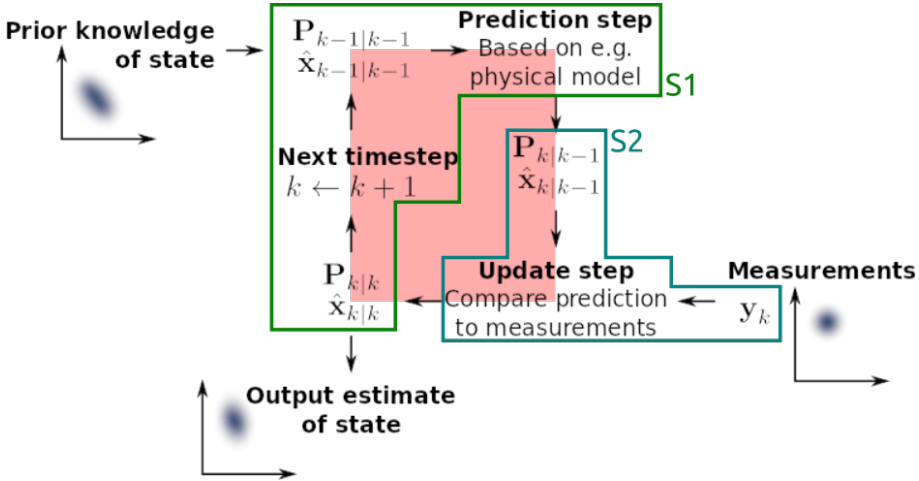
Write a program to generate data from this process.

Coffee break: 15 minutes

Update step

We are now back to the Kalman filter. We will first review what we know and then advance to the full algorithm.

KF overview



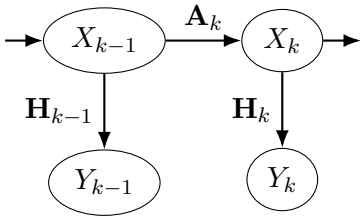
In the first part of the seminar we learned the modeling component that lead to the prediction step of the Kalman filter: how to advance the model starting from a "known" state. You should be able to use the Kalman filter in applications and intuitively understand the effect of the different matrices in the setup.

We will now look at the whole Kalman filter algorithm from a probabilistic perspective. This allows us to introduce the update step in a unified framework that also includes other KF algorithms (nonlinear, non-gaussian).

KF: Probabilistic prediction step

Given the distribution of the previous state conditioned on all past measurements, it provides the distribution of the current state conditioned on all past measurements

$$p(x_{k-1}|y_{:k-1}) \to p(x_k|y_{:k-1})$$
$$p(x_k x_{k-1}|y_{:k-1}) = p(x_k|x_{k-1}y_{:k-1})p(x_{k-1}|y_{:k-1})$$



Dynamic model: $p(x_k|x_{k-1})$ uses only x_{k-1}: it blocks information from previous measurements. $p(x_k|x_{k-1}y_{:k-1}) = p(x_k|x_{k-1})$

x_{k-1} is distributed. We "average" over all possible previous states, compatible with the measurements so far:

$$p(x_k|y_{:k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|y_{1:k-1})\mathrm{d}x_{k-1}$$

- When the proposition $A \equiv$ the variable takes the value v we write

$$p(A|C) \doteq p(v|C)$$

  I have omitted the background information in the slide, but it is always there. For example, the (parameters of the) distribution of the noise.

- $p(x_k|y_{:k-1})$ Is the **predictive distribution**.
- The independence on history, and only on previous state is called Markov property.
- The last equation is called the Chapman-Kolmogorov equation.
- When y_k is given (measured), $p(y_k|x_k)$ is called the likelihood of the measured value (data).
- See Theorem 4.1 from Särkkä, S. (2013). Bayesian Filtering and Smoothing doi:10.1017/CBO9781139344203

KF: Probabilisitc update step

Prediction gives the **predictive distribution**:

$$p(\boldsymbol{x}_k|\boldsymbol{y}_{:k-1}) = \int p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})p(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{:k-1})\mathrm{d}\boldsymbol{x}_{k-1}$$

Update gives the **filtering distribution**: condition the current state also on the current measurement

$$p(\boldsymbol{x}_k|\boldsymbol{y}_{:k}) \propto p(\boldsymbol{y}_k|\boldsymbol{x}_k)p(\boldsymbol{x}_k|\boldsymbol{y}_{:k-1})$$

- $p(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{:k-1})$ previous estimate.
- $p(\boldsymbol{x}_k|\boldsymbol{y}_{:k-1})$ what we believe $\boldsymbol{x}_k$ would be before we observed $\boldsymbol{y}_k$ (predictive distribution).
- $p(\boldsymbol{y}_k|\boldsymbol{x}_k)$ the likelihood of the observed $\boldsymbol{y}_k$ given the model (measurement).

- The update proportionality comes from Baye's rule:

$$p(x_k|y_{:k-1}y_k) = \frac{p(y_k|x_k y_{:k-1})p(x_k|y_{:k-1})}{p(y_{:k-1}y_k)} = \frac{p(y_k|x_k)p(x_k|y_{:k-1})}{p(y_{:k})}$$

$$p(y_{:k}) = \int p(y_k|x_k)p(x_k|y_{:k-1})\mathrm{d}x_k$$

  The first factor doesn't depend on $y_{:k-1}$ due to the measurement model, which gives the distribution of the measurement based only on x_k.

- See Theorem 4.1 from Särkkä, S. (2013). Bayesian Filtering and Smoothing doi:10.1017/CBO9781139344203

Distribution of Gaussian variables

For Gaussian variables:

$$\mathcal{N}(\boldsymbol{x}|\boldsymbol{m},\mathbf{P}) = \frac{1}{\sqrt{(2\pi)^n \det \mathbf{P}}} \exp -\frac{1}{2}(\boldsymbol{x}-\boldsymbol{m})^\top \mathbf{P}^{-1}(\boldsymbol{x}-\boldsymbol{m})$$

There are several exact identities that can be used.
Given:

$$\begin{aligned}\boldsymbol{x} &\sim \mathcal{N}(\boldsymbol{m},\mathbf{P})\\ \boldsymbol{y}|\boldsymbol{x} &\sim \mathcal{N}(\mathbf{H}\boldsymbol{x}+\boldsymbol{u},\mathbf{R})\end{aligned}$$

Then:

$$\begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{y}\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}\boldsymbol{m}\\ \mathbf{H}\boldsymbol{m}+\boldsymbol{u}\end{bmatrix}, \begin{bmatrix}\mathbf{P} & \mathbf{P}\mathbf{H}^\top\\ \mathbf{H}\mathbf{P} & \mathbf{H}\mathbf{P}\mathbf{H}^\top+\mathbf{R}\end{bmatrix}\right)$$
$$\boldsymbol{y} \sim \mathcal{N}(\mathbf{H}\boldsymbol{m}+\boldsymbol{u}, \mathbf{H}\mathbf{P}\mathbf{H}^\top+\mathbf{R})$$

The formulas are given in appendix A (p. 209) of Särkkä, S. (2013). Bayesian Filtering and Smoothing doi:10.1017/CBO9781139344203
Do exercise 3.5 from that book.

The conditional of y on x implies:

$$y = Hx + u + \epsilon, \quad \epsilon \sim \mathcal{N}(0, R)$$

The mean value of y given x is then:

$$\mathrm{E}[y] = \mathrm{E}[Hx+u+\epsilon] = H\,\mathrm{E}[x] + \mathrm{E}[u] + \mathrm{E}[\epsilon] = Hm + u$$

The covariance of y given x is then:

$$\mathrm{cov}[y] = \mathrm{cov}[Hx+u+\epsilon] = H\,\mathrm{cov}[x]H^\top + \mathrm{cov}[u] + \mathrm{cov}[\epsilon] = HPH^\top + R$$

This gives the marginal distribution of y, shown at the bottom. We got this result before by propagation of uncertainty.

Distribution of Gaussian variables

For Gaussian variables:

$$\mathcal{N}(\boldsymbol{x}|\boldsymbol{m},\mathbf{P}) = \frac{1}{\sqrt{(2\pi)^n \det \mathbf{P}}} \exp -\frac{1}{2}(\boldsymbol{x}-\boldsymbol{m})^\top \mathbf{P}^{-1}(\boldsymbol{x}-\boldsymbol{m})$$

There are several exact identities that can be used.
Given:

$$\begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{y}\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}\boldsymbol{a}\\ \boldsymbol{b}\end{bmatrix}, \begin{bmatrix}\mathbf{A} & \mathbf{C}\\ \mathbf{C}^\top & \mathbf{B}\end{bmatrix}\right)$$

Then:

$$\begin{aligned}\boldsymbol{y} &\sim \mathcal{N}(\boldsymbol{b},\mathbf{B})\\ \boldsymbol{x}|\boldsymbol{y} &\sim \mathcal{N}(\boldsymbol{a}+\mathbf{C}\mathbf{B}^{-1}(\boldsymbol{y}-\boldsymbol{b}),\ \mathbf{A}-\mathbf{C}\mathbf{B}^{-1}\mathbf{C}^\top)\end{aligned}$$

Do exercise 3.5 from the book.
Check the Schur complement of a matrix.
See also Rasmussen, C. E., and Williams, C. K. I. (2005). Gaussian processes for machine learning. MIT Press. Appendix A.2

Having the marginal distribution of $\boldsymbol{y}$, we get $\boldsymbol{b}$ and B. We use them in the identity to obtain C:

$$\begin{aligned}&\mathrm{E}[y|x] = b + C^\top A^{-1}(x-a) = Hm + u + C^\top P^{-1}(x-m) \doteq Hx + u\\ &\quad \to C^\top P^{-1}(x-m) = H(x-m) \Rightarrow C^\top = HP\end{aligned}$$

Could also be computed from the cross-covariance.

KF: Prediction and Update step

These identities provide us analytic formulas for the KF.

Prediction

$$p(\boldsymbol{x}_k|\boldsymbol{y}_{:k-1}) = \mathcal{N}(\boldsymbol{x}_k|\hat{\boldsymbol{m}}_k, \hat{\mathbf{P}}_k)$$
$$\hat{\boldsymbol{m}}_k = \mathbf{A}_k \boldsymbol{m}_{k-1}$$
$$\hat{\mathbf{P}}_k = \mathbf{A}_k \mathbf{P}_{k-1} \mathbf{A}_k^\top + \mathbf{Q}_{k-1}$$

Measurement

$$p(\boldsymbol{y}_k|\boldsymbol{x}_k) = \mathcal{N}(\boldsymbol{y}_k|\hat{\boldsymbol{y}}_k, \mathbf{S}_k)$$
$$\hat{\boldsymbol{y}}_k = \mathbf{H}_k \hat{\boldsymbol{m}}_k$$
$$\mathbf{S}_k = \mathbf{H}_k \hat{\mathbf{P}}_k \mathbf{H}_k^\top + \mathbf{R}_k$$

Update

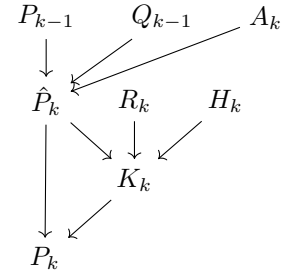
$$p(\boldsymbol{x}_k|\boldsymbol{y}_{:k}) = \mathcal{N}(\boldsymbol{x}_k|\boldsymbol{m}_k, \mathbf{P}_k)$$
$$\mathbf{K}_k = \hat{\mathbf{P}}_k \mathbf{H}_k^\top \mathbf{S}_k^{-1}$$
$$\boldsymbol{m}_k = \hat{\boldsymbol{m}}_k + \mathbf{K}_k (\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k)$$
$$\mathbf{P}_k = \hat{\mathbf{P}}_k - \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^\top$$

After the update one re-estimates the output

$$\mathcal{N}(\boldsymbol{y}_k|H\boldsymbol{m}_k, \mathbf{H}_k \mathbf{P}_k \mathbf{H}_k^\top + \mathbf{R}_k)$$

The proof of these formulas uses the identities seen before. A step by step explanation is given in Särkkä (2013), section 4.3, p. 57.

Note: the evolution of P_k doesn't depend on the current data/state, only on the model.



If A, Q, H, R do not depend on the data or the states history, then P_k only depends on the model specification.

Kalman filter

$$\text{while no measurement: predict} \begin{cases} \hat{\boldsymbol{m}}_k = \mathbf{A}_k \boldsymbol{m}_{k-1} \\ \hat{\mathbf{P}}_k = \mathbf{A}_k \mathbf{P}_{k-1} \mathbf{A}_k^\top + \mathbf{Q}_{k-1} \end{cases}$$

$$\text{predict measurement} \begin{cases} \hat{\boldsymbol{y}}_k = \mathbf{H}_k \hat{\boldsymbol{m}}_k \\ \hat{\mathbf{S}}_k = \mathbf{H}_k \hat{\mathbf{P}}_k \mathbf{H}_k^\top + \mathbf{R}_k \end{cases}$$

$$\text{on measurement: update} \begin{cases} \mathbf{K}_k = \hat{\mathbf{P}}_k \mathbf{H}_k^\top \hat{\mathbf{S}}_k^{-1} \\ \boldsymbol{m}_k = \hat{\boldsymbol{m}}_k + \mathbf{K}_k \left(\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k \right) \\ \mathbf{P}_k = \hat{\mathbf{P}}_k - \mathbf{K}_k \hat{\mathbf{S}}_k \mathbf{K}_k^\top \end{cases}$$

Implement these functions in a programming language:

- `m, P` $\longleftarrow$ `kf_predict(m, P, A, Q)`
- `m, P, K` $\longleftarrow$ `kf_update(m, P, H, R, y)`
- `y, S` $\longleftarrow$ `kf_measure(m, P, H, R)`

1. Propagation of uncertainty to output using best estimate
2. Kalman gain
3. Update mean by projecting error on Kalman gain
4. Update cov by propagating uncertainty on measurement backward

Do not forget to update your measurement prediction after and update.

Which matrices participate in each function?

How would you determine which states are most sensitive to the residuals $\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k$?

Wrap-up

Correlation and Causation

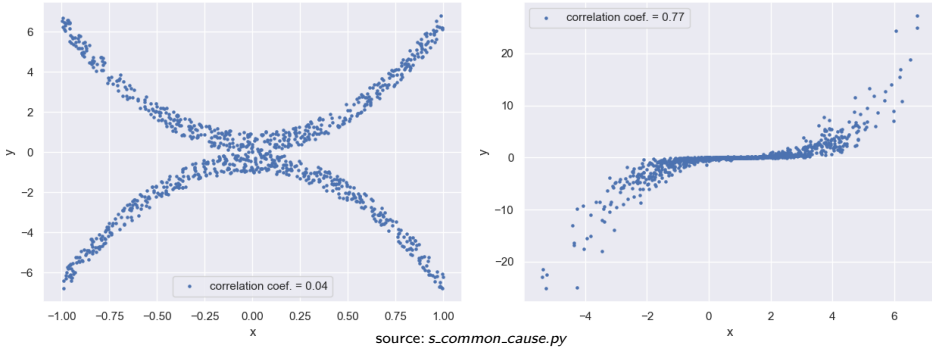
Correlation doesn't imply causation

causal, no correlation

$$x \sim \mathcal{U}(-1, 1)$$
$$y \leftarrow a\frac{\epsilon}{|\epsilon|}x^2 + \epsilon \quad \epsilon \sim \mathcal{U}(-1, 1)$$

not causal, correlated

$$c \sim \mathcal{N}(0, 1)$$
$$x \sim \mathcal{N}(2c - 1, 0.2)$$
$$y \sim \mathcal{N}(c^3, 0.1)$$



source: *s_common_cause.py*

X vs. Y plots have different interpretations. On the left panel we see a "functional" or "causal" plot, in which changing the values of x will affect the values of y. This is because the variables are connected by a function, y being a function of x. Note that the correlation, however, is negligible. On the right panel we see a plot that would seem to also be "causal", saying that values of x affect y, however if we inspect the process generating the data we see that x does not affect y. The variables are correlated and are statistically dependent, but they are not causally dependent.

Extreme learning machines

Basis function decomposition

Any function of the independent variable can be put in the design matrix:

$$\mathbf{D}_{k:} = \begin{bmatrix} \phi_n(t_k) & \cdots & \phi_1(t_k) \end{bmatrix}$$

which corresponds to the measurement matrix, i.e.

$$\mathbf{H}_k = \mathbf{D}_{k:}$$

combined with the drift dynamic model

$$\begin{aligned} \boldsymbol{x}_k &= \boldsymbol{w}_k \\ \mathbf{A} &= \mathbf{I}, \quad \mathbf{Q} \neq \mathbf{0} \end{aligned}$$

gives the recursive version.
Properties of the function set $\{\phi_i\}$ can be used to write a more stable model, e.g. polynomials.
See `s_fourier.m` and `s_fourier_adaptive.m` for frequency tracking.

In machine learning $\mathbf{H}_k$ is also known as a feature vector.
The only requisite of KF is that the models (dynamic and measurement) are linear in the states.

In the book Särkkä, S. (2013). Bayesian Filtering and Smoothing, this model is called **Linear-in-parameters regression model II**, described in example 3.2, page 41.

Kalman filter

$$\text{while no measurement: predict} \begin{cases} \hat{\boldsymbol{m}}_k = \mathbf{A}_k \boldsymbol{m}_{k-1} \\ \hat{\mathbf{P}}_k = \mathbf{A}_k \mathbf{P}_{k-1} \mathbf{A}_k^\top + \mathbf{Q}_{k-1} \end{cases}$$

$$\text{predict measurement} \begin{cases} \hat{\boldsymbol{y}}_k = \mathbf{H}_k \hat{\boldsymbol{m}}_k \\ \hat{\mathbf{S}}_k = \mathbf{H}_k \hat{\mathbf{P}}_k \mathbf{H}_k^\top + \mathbf{R}_k \end{cases}$$

$$\text{on measurement: update} \begin{cases} \mathbf{K}_k = \hat{\mathbf{P}}_k \mathbf{H}_k^\top \hat{\mathbf{S}}_k^{-1} \\ \boldsymbol{m}_k = \hat{\boldsymbol{m}}_k + \mathbf{K}_k \left(\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k \right) \\ \mathbf{P}_k = \hat{\mathbf{P}}_k - \mathbf{K}_k \hat{\mathbf{S}}_k \mathbf{K}_k^\top \end{cases}$$

Implement these functions in a programming language:

- `m, P` ⟵ `kf_predict(m, P, A, Q)`
- `m, P, K` ⟵ `kf_update(m, P, H, R, y)`
- `y, S` ⟵ `kf_measure(m, P, H, R)`

1. Propagation of uncertainty to output using best estimate
2. Kalman gain
3. Update mean by projecting error on Kalman gain
4. Update cov by propagating uncertainty on measurement backward

Do not forget to update your measurement prediction after and update.

Which matrices participate in each function?

How would you determine which states are most sensitive to the residuals $\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k$?