Norges miljø- og biovitenskapelige universitet

Institutt for datavitenskap

Digitalisering på Ås

DAT121
Introduction to data science

5 Multidimensionality (multicriteria optimization)

5.1 Rational choice
5.2 Multiple objectives
5.3 Computing the Pareto front

Schedule for DAT121 parts 4 and 5

Norwegian University of Life Sciences

Friday, 25th August 2023

10.15 – 11.00 lecture on good practice
11.15 – 12.00 **interest group sessions**
13.15 – 15.00 project work and tutorial

Monday, 28th August 2023

9.15 – 10.00 first multidimensionality lecture
10.15 – 10.?? Pandasia presentation in TF1-115
11.15 – 11.?? **linjeforeningen's presentation** (back in TF1-205)
13.15 – 15.00 project work and tutorial

Tuesday, 29th August 2023

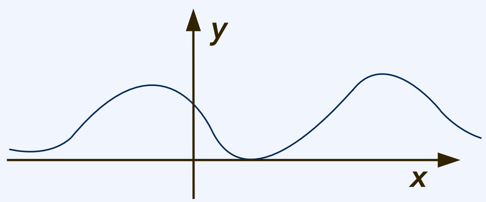
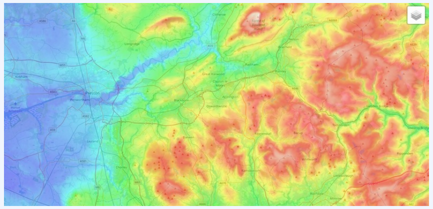
9.15 – 10.00 Q&A session and discussion
10.15 – 11.00 second multidimensionality lecture
11.15 – 12.00 **interest group sessions**
13.15 – 15.00 project work and tutorial

Dimensionality and optimization

Norwegian University of Life Sciences

optimization objective

Is there one parameter, or are there multiple parameters?
Is there one objective, or are there multiple objectives?

	utility or cost	multicriteria (MCO)
single parameter		 §
multivariate optimization		

Parameters: Quantites that are part of the solution and **directly in your control**.

Objectives: Quantities that should become as small (or as large) as possible, but can be influenced only **indirectly, through a good choice of parameters**.

Noregs miljø- og biovitskaplege universitet

Institutt for datavitskap

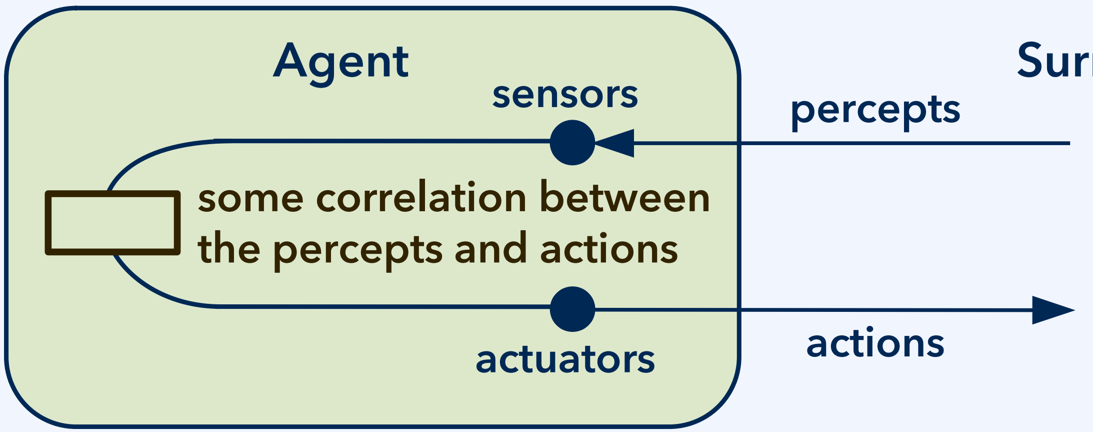
Digitalisering på Ås

5 Multidimensionality

5.1 Rational choice

Agents and decisions

Agents are systems that can **interact** with their surroundings. For interaction, two directions are needed. **Percepts** (input signals) are received through **sensors**, and actions (motion, speech, *etc.*) are done through actuators.



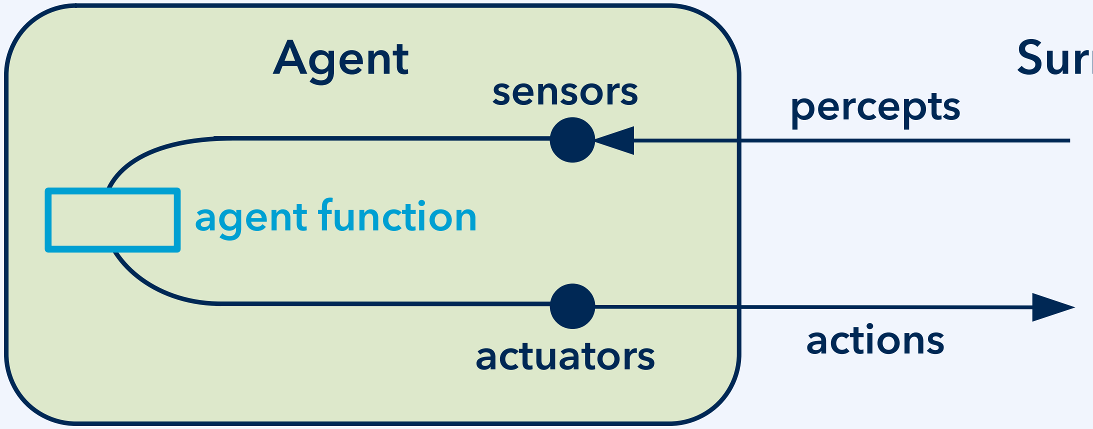
Almost any system can be analysed as an agent. Humans, cleaning robots, and seagulls can be understood as agents. But so can a pocket calculator, the stock exchange, the earth's climate system, or a single neuron in the brain.

This is not so much a category of systems, it is rather a way of analysing them.

Agents and decisions

agent

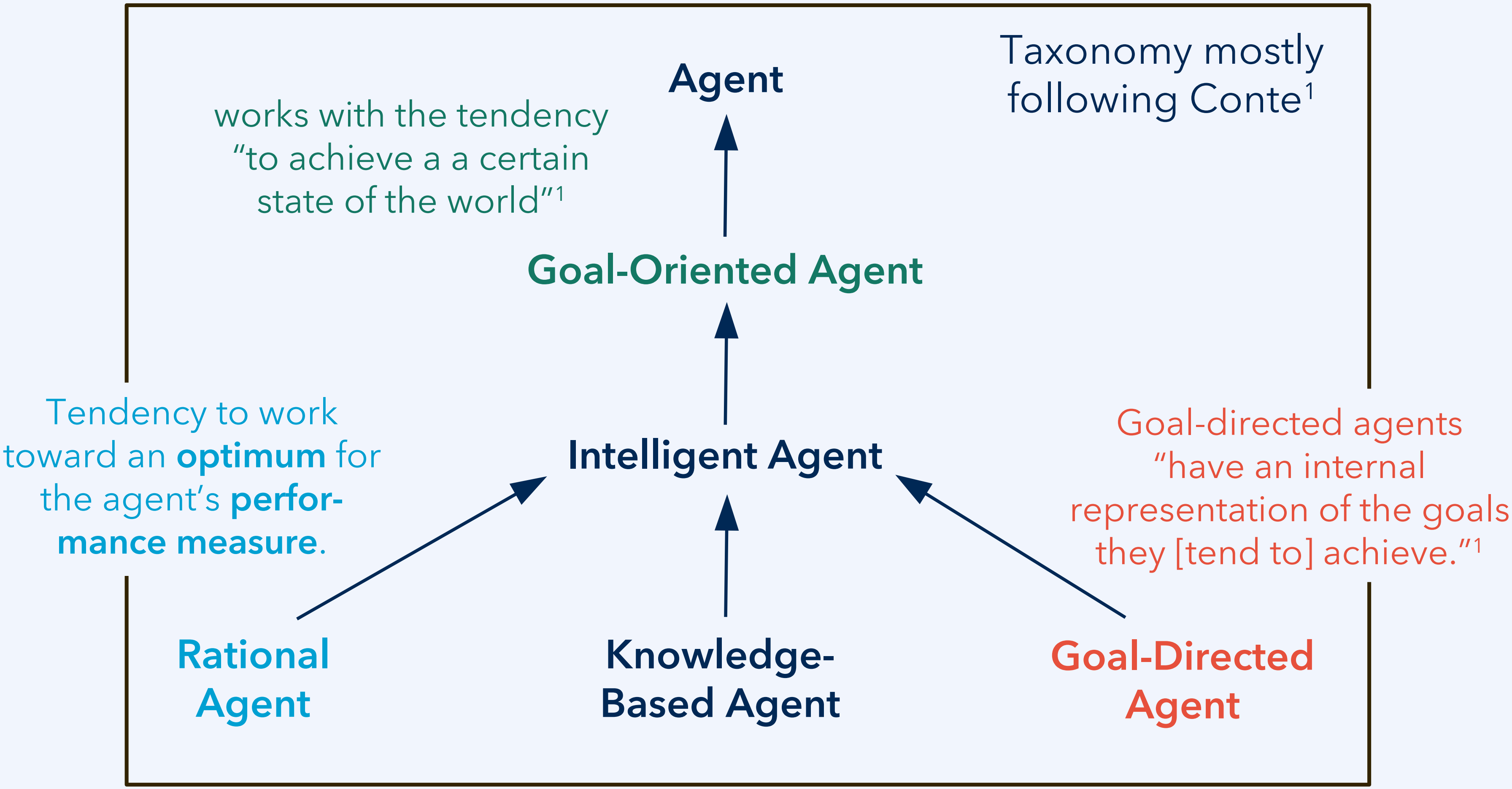
Agents are systems that can **interact** with their surroundings. For interaction, two directions are needed. **Percepts** (input signals) are received through **sensors**, and actions (motion, speech, *etc.*) are done through actuators.



This characterization is sufficiently general to make **any program with an input and an output** an "agent"; *e.g.*, a program may read from standard input, files, and devices, and act upon standard output, files, and devices.

We may want to become more specific about what we expect from AI agents.

Agency and rationality



[1]R. Conte, "Rational, goal-oriented agents," in R. A. Meyers (ed.), *Encyclopedia of Complexity and Systems Science*, Springer, **2009**.

doi:10.1007/978-0-387-30440-3_445

Rational decision making

Necessary criteria for rational decision making:

Assume A, B, C are possible states of affairs. X and Y are probability distributions (lotteries) over states of affairs, *e.g.*, "50% chance A, 50% chance B".

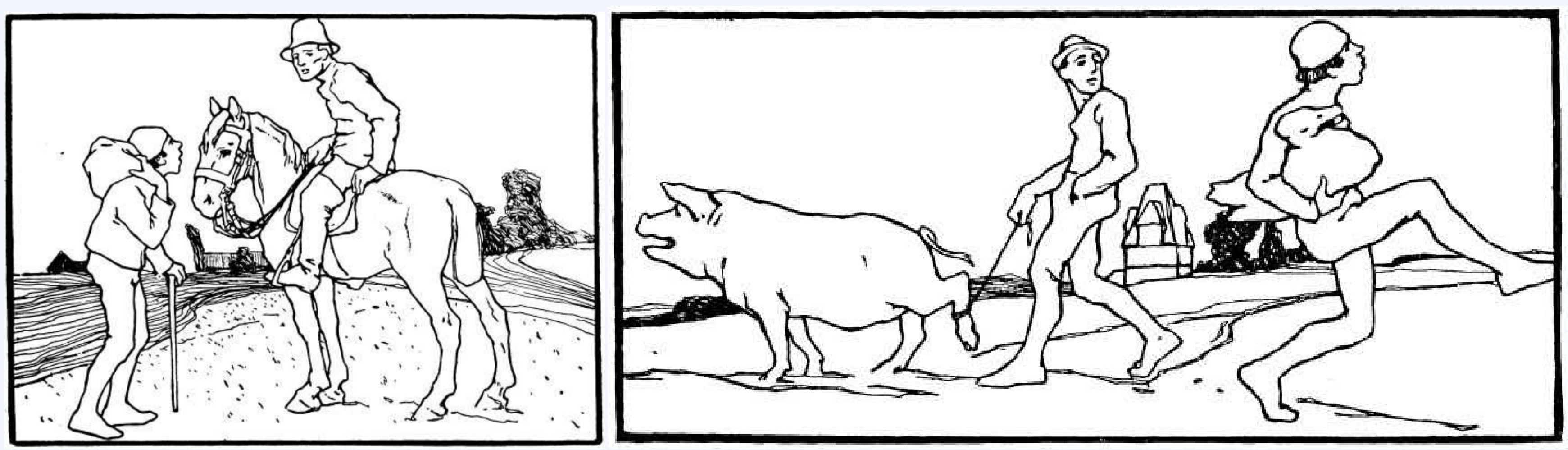
- **Transitivity:** If the agent prefers A over B, and B over C, then the agent also prefers A over C, whenever given the choice.
- **Monotonicity:** If the agent prefers A over B, and both X and Y have only A and B as their possible outcomes, where the probability of A is greater in the case of X, then the agent prefers X over Y.
- **Continuity:** If the agent prefers A over B, and B over C, then there is exactly one lottery X, with A and C as its only possible outcomes, such that the agent is indifferent between B and X. For any other lottery Y between A and C, the agent prefers Y over B if the probability of A is greater than in case of X; otherwise, the agent prefers B over Y.

Rational decision making

Necessary criteria for rational decision making:

Assume A, B, C are possible states of affairs. X and Y are probability distributions (lotteries) over states of affairs, *e.g.*, “50% chance A, 50% chance B”.

- **Transitivity:** If the agent prefers A over B, and B over C, then the agent also prefers A over C, whenever given the choice.



“Hans im Glück” (“Lyykehans”) makes irrational choices, violating transitivity; artist: O. Ubbelohde (1909).

Multiple objectives

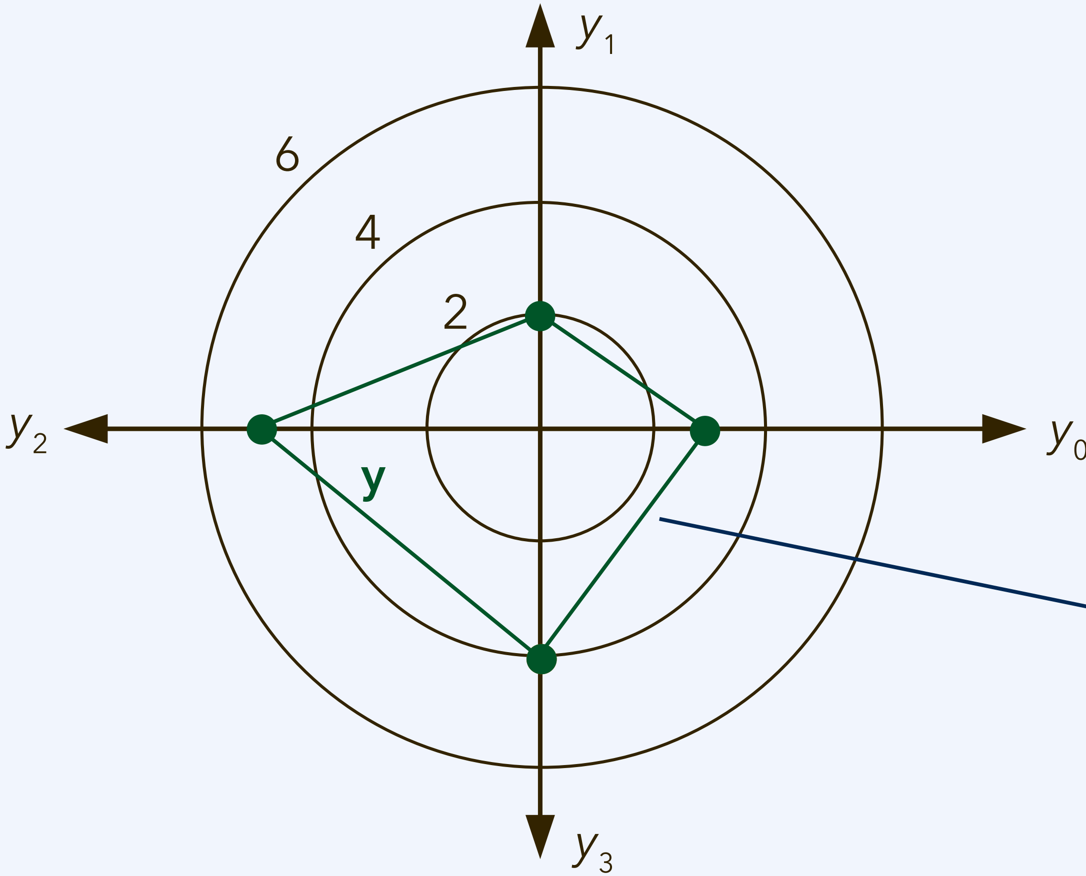
Song by Walther von der Vogelweide (~ 1200)

Diu wolte ich gerne in einen schrîn.
Jâ leider des enmac niht sîn,
daz **guot** *und* **weltlich êre**
und **gotes hulde** *mêre*
zesamene in ein herze komen.

These I would like to have in one box.
But sadly that may not be,
that **goods** and **worldly honour**,
and **God's grace** additionally,
come together in a single heart.

Dominated choices and irrationality

Spider diagrams are often used to visualize points in objective space.



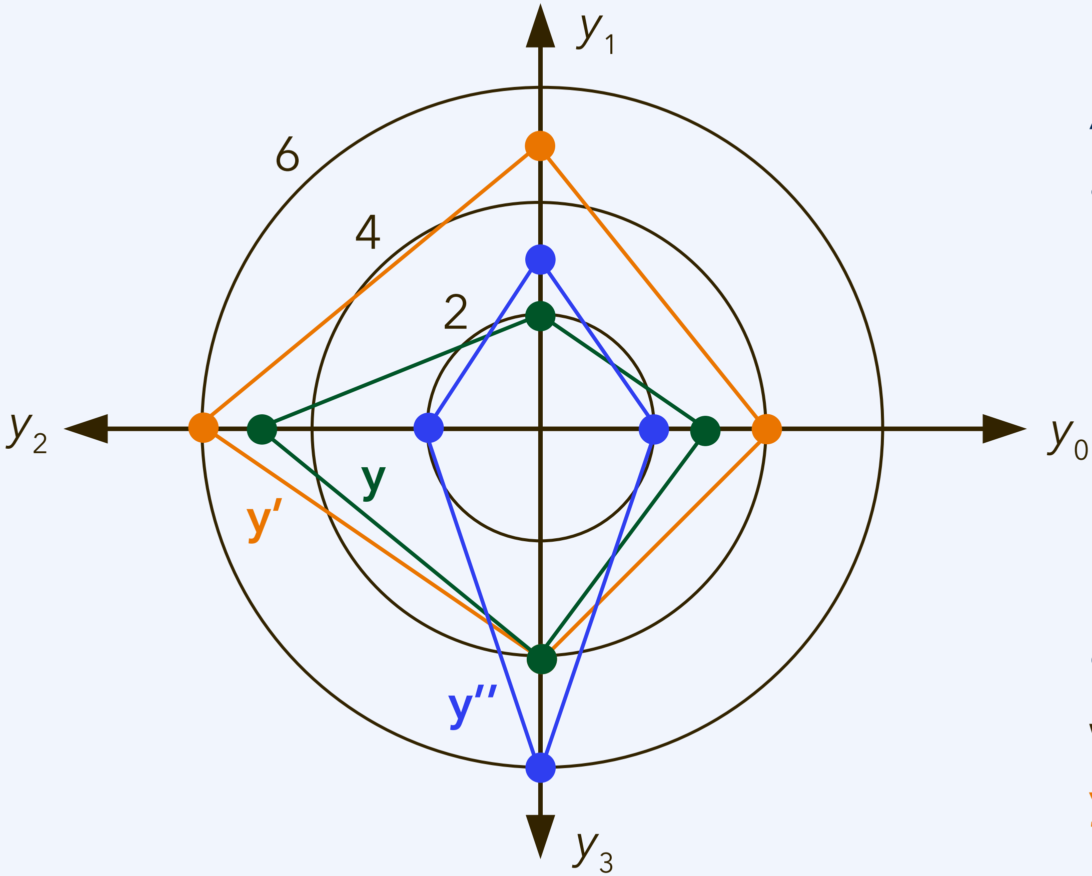
Assume that y_0, y_1, y_2, and y_3 are all minimization objectives.

$\mathbf{y} = [3, 2, 5, 4]$

Each polygon in a spider diagram represents **one point** in objective space, in this case **y**.

Dominated choices and irrationality

Spider diagrams are often used to visualize points in objective space.



Assume that y_0, y_1, y_2, and y_3 are all minimization objectives.

y = [3, 2, 5, 4]

y′ = [4, 5, 6, 4]

y″ = [2, 3, 2, 6]

Note: **y** and **y′** perform equally in criterion y_3, and in the three other criteria, **y** outperforms **y′**.

We say: y dominates y′. Since **y′** is dominated, it cannot be a rational choice to select **y′**.

Decision support and decision making

As a **decision support** system, *e.g.*, based on multicriteria optimization, artificial intelligence can assist *another entity* (such as a human decision maker).

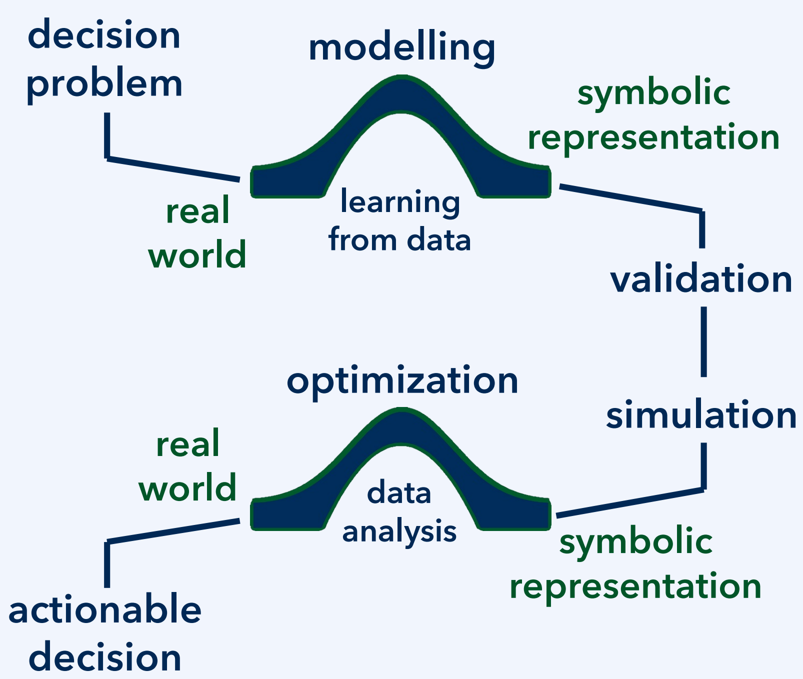
However, if we employ a single objective ("overall cost" or "overall utility"),

$$y = f(\mathbf{x}), \text{ where } \mathbf{x} = [x_0, x_1, \ldots, x_{m-1}],$$

where y is a *scalar*, a single numerical value, **decision making** can be automated.

The best decision is then given by a choice of parameters x_0 to x_{m-1} such that y becomes minimal (if it is "cost") or maximal (if it is "utility").

Typical workflow in data and modelling based decision support and decision making



Noregs miljø- og biovitskaplege universitet

Institutt for datavitskap

Digitalisering på Ås

5 Multidimensionality

5.1 Rational choice

5.2 Multiple objectives

Specification of an MCO problem

Three elements need to be specified to facilitate multicriteria decision support:

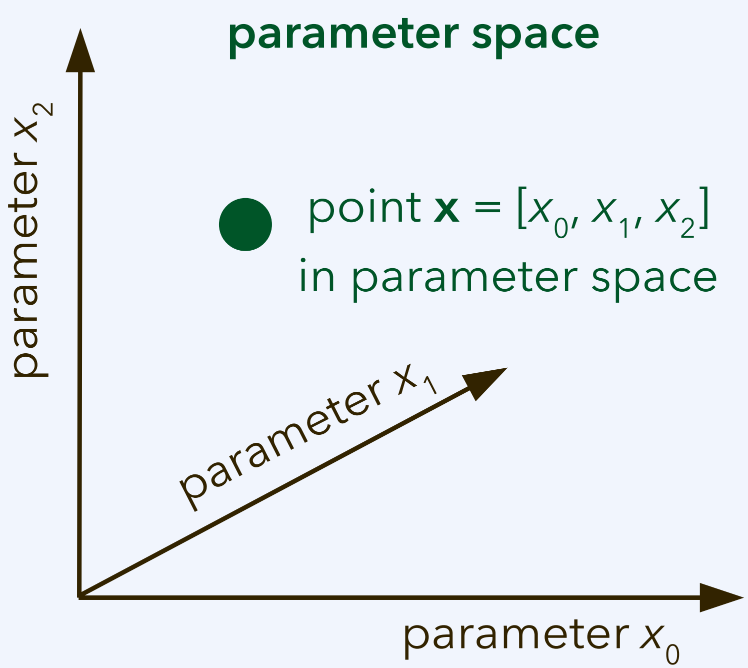
1) The **parameter space**: What is it that can be varied and is in direct control (or to be assumed as being under direct control) of the decision maker? What quantities define that which is possible in the scenario?

   The permitted range (+ any applicable constraints) need to be stated.

2) The **objective space**: What are our criteria? If numerical optimization with scipy is to be used, best expressed as mnimization objectives.

3) The **objective function** – if all criteria are minimization objectives, this can be called a multicriteria **cost function**: This must *actually be implemented* ***as a function***, in the sense that the term has in programming.

Specification of an MCO problem

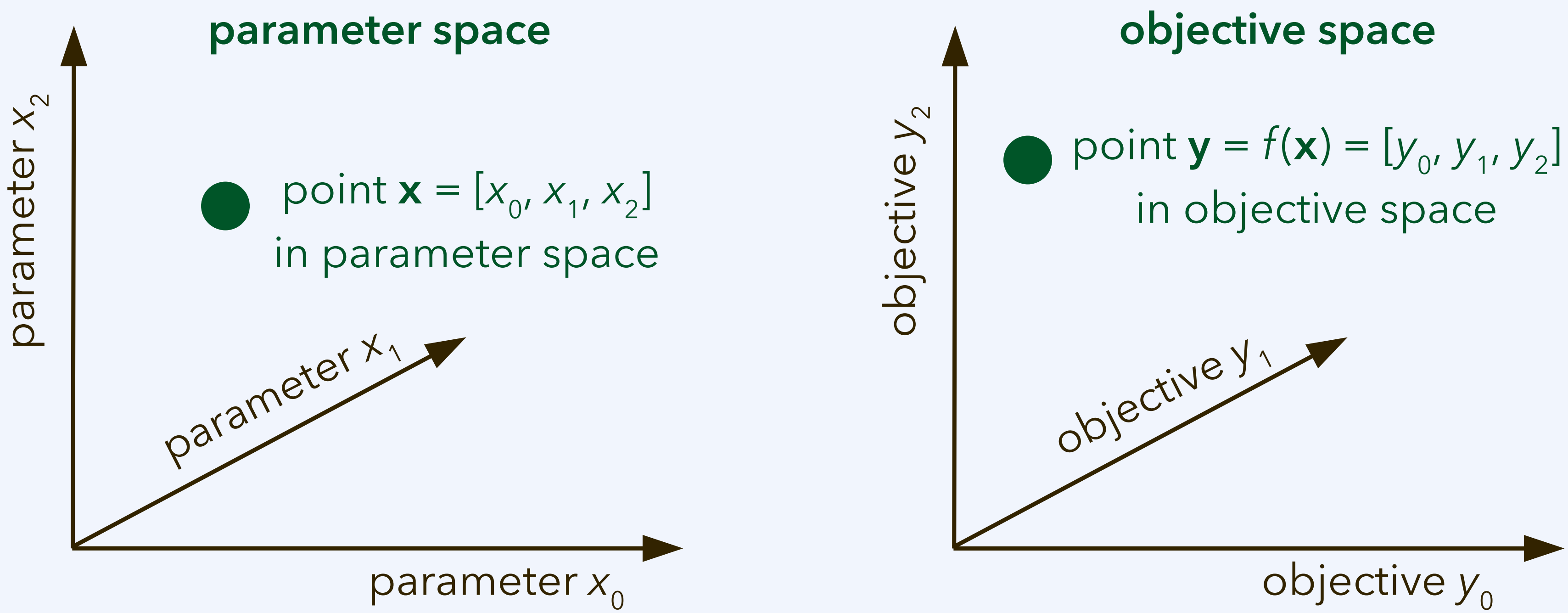


optimization parameter

In general, there can be any number *m* of parameters (*m*-dimensional parameter space); the parameter space is defined by parameters and a range of values that is **accessible**, *i.e.*, within which decision making is actually possible.

This requires that multiple parameter values can be selected independently.

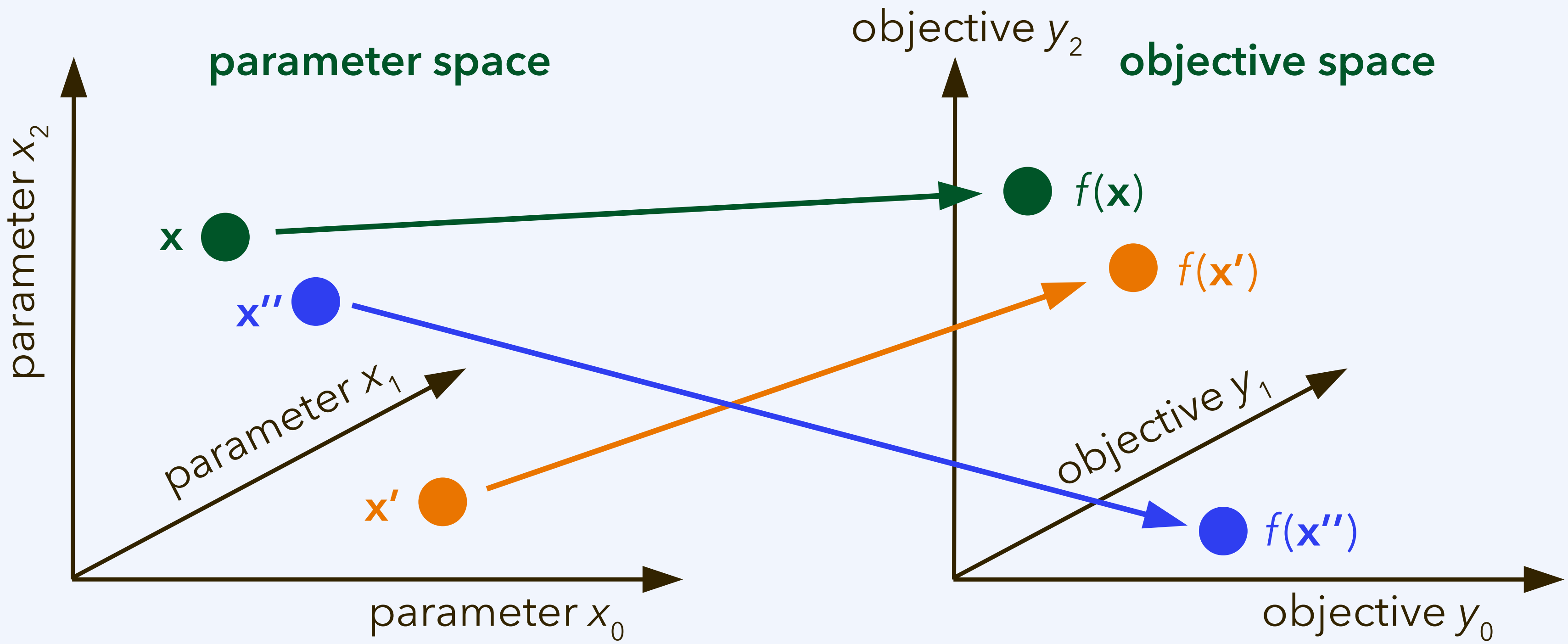
Specification of an MCO problem



In general, there can be any number *m* of parameters (*m*-dimensional parameter space) and any number *n* of objectives (*n*-dimensional objective space).

The optimization problem is defined by a function *f* that maps a list of parameters $x = [x_0, \ldots, x_{m-1}]$ to the outcome for the objectives $\mathbf{y} = f(\mathbf{x}) = [y_0, \ldots, y_{n-1}]$.

Specification of an MCO problem

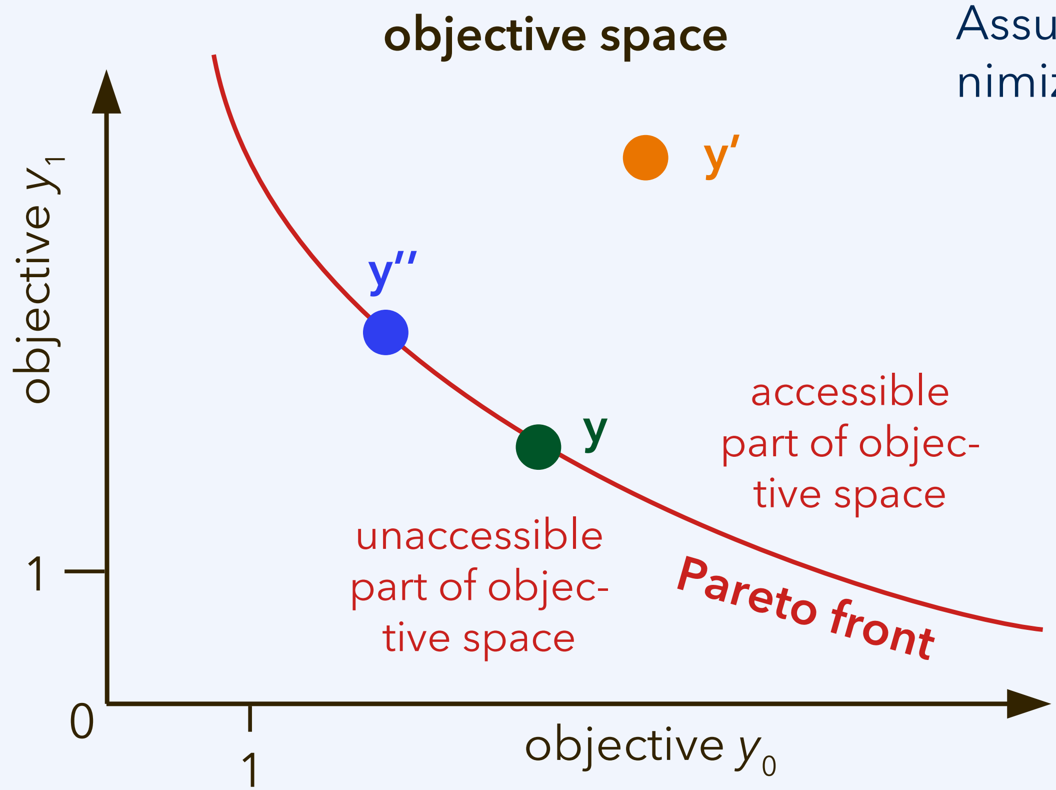


Design choices are made by selecting a point in parameter space, but they are **evaluated in objective space**, by comparing the outcomes for $y_0, \ldots, y_{n-1}$.

The optimization problem is defined by a function f that maps a list of parameters $\mathrm{x} = [x_0, \ldots, x_{m-1}]$ to the outcome for the objectives $\mathbf{y} = f(\mathbf{x}) = [y_0, \ldots, y_{n-1}]$.

Pareto front (also called Pareto set)

An accessible point in objective space belongs to the **Pareto front** whenever it is **not dominated** by any other accessible point in objective space.



Assume that there are two minimization objectives, y_0 and y_1.

y = [3, 2]

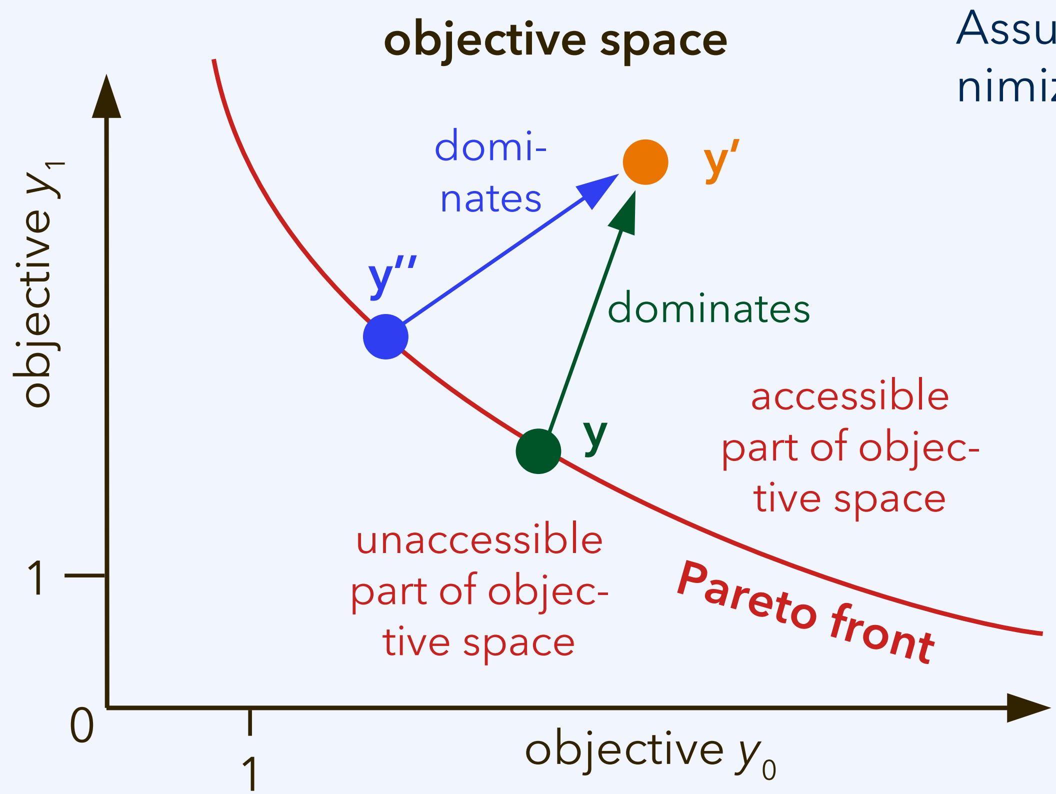
y' = [4, 5]

y'' = [2, 3]

Pareto optimality

Pareto front (also called Pareto set)

An accessible point in objective space belongs to the **Pareto front** whenever it is **not dominated** by any other accessible point in objective space.



Assume that there are two minimization objectives, y_0 and y_1.

y = [3, 2]
y' = [4, 5]
y'' = [2, 3]

y and **y''** are rational compromises between the two objectives.

y' is not, because it is dominated by other accessible points.

Dimensionality in multicriteria optimization

- The dimension of the **parameter space**, defined by m independently variable parameters x_0, …, x_{m-1}, is exactly m, by construction.
- The **objective space**, defined over n criteria y_0, …, y_{n-1}, has dimension n.
- The **accessible part of objective space** (*i.e.*, image of the objective function) cannot be higher-dimensional than the objective space as such. Therefore, its dimension q must satisfy $q \leq n$. But the image of a continuous function cannot be of a dimension greater than that of its domain. Therefore, $q \leq m$.
- The **Pareto front** in objective space must be lower-dimensional than objective space itself, due to the domination criterion; therefore, $p \leq n-1$ for its dimension p. But it must also all be accessible; therefore, $p \leq q \leq m$.
- The dimension $p' \leq m$ of the **Pareto-optimal region in parameter space** must be at least the same as that of the Pareto front; hence, $p \leq p' \leq m$.

Example (molecular model parameterization)

The considered problem was from model optimization. The task was to parameterize models that accurately reflect physical behaviour.

Four model parameters, $m = 4$.

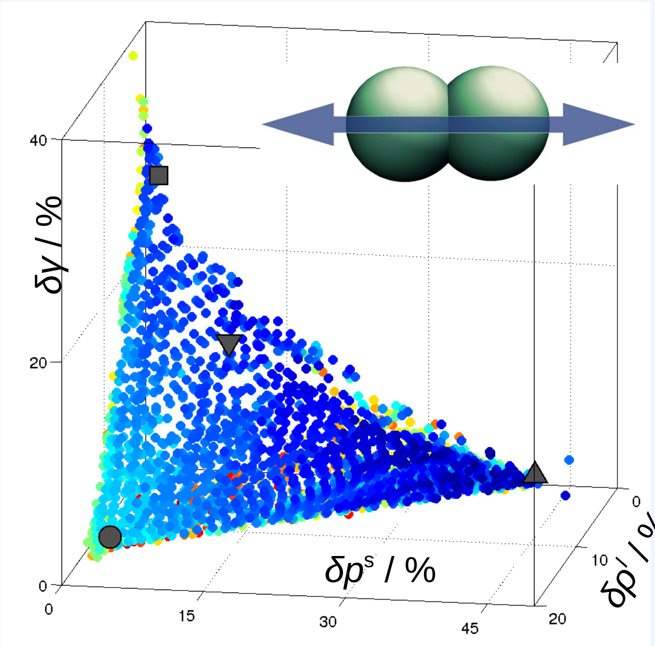
Three criteria (quantifying accuracy of predictions for three kinds of properties), $n = 3$.

Dimension of image of the objective function (accessible part of objective space), $q = 3$.

Dimension of the Pareto front, $p = 2$.

Dimension of Pareto-optimal part of parameter space: $p' = 2$.

2CLJQ molecular models of low-molecular fluids: Objective space and Pareto front



K. Stöbener, P. Klein, M. Horsch, K. Küfer, H. Hasse, *Fluid Phase Equilib.* 411, 33 – 42, **2016**.

Noregs miljø- og biovitskaplege universitet

Institutt for datavitskap

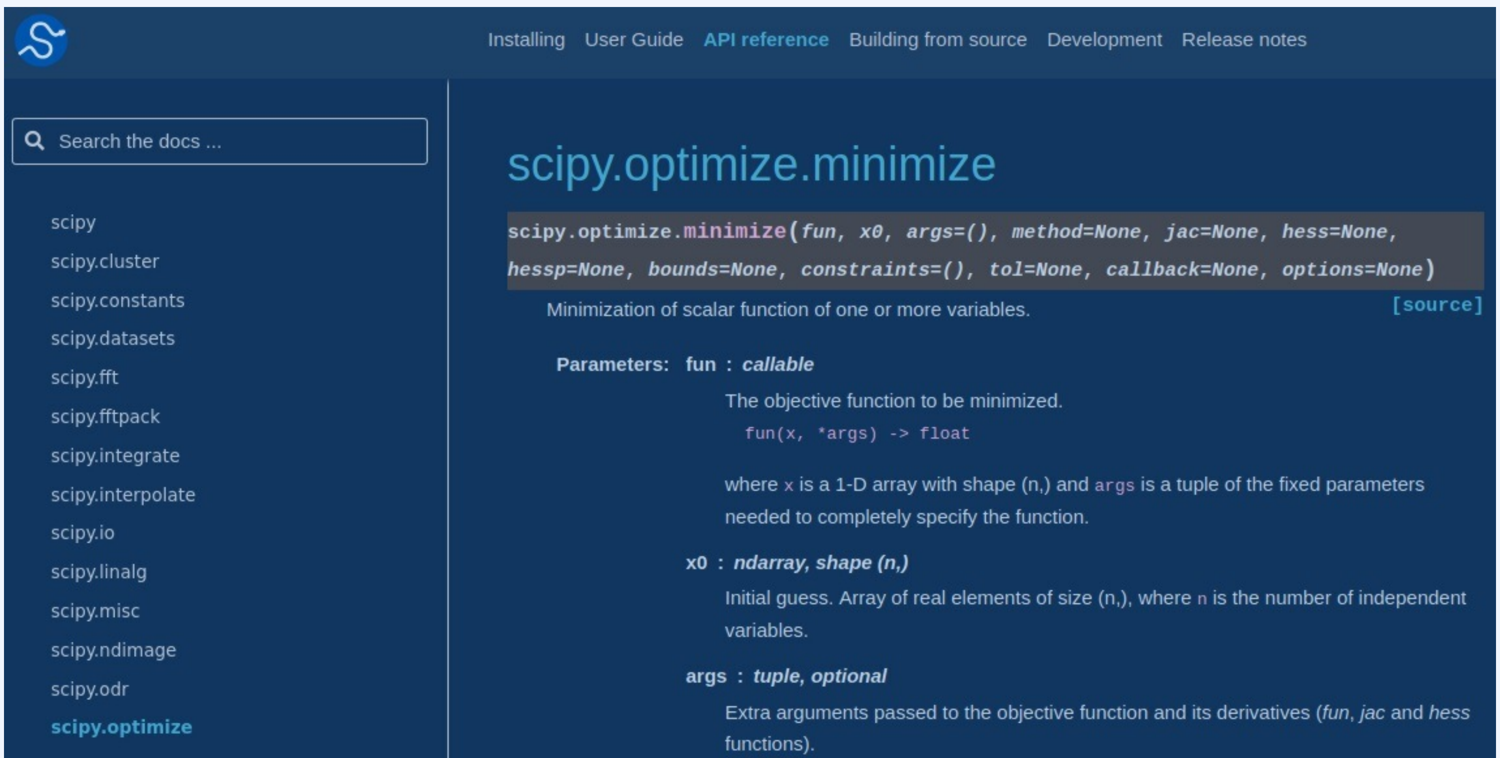
Digitalisering på Ås

5 Multidimensionality

Single-objective optimization using scipy

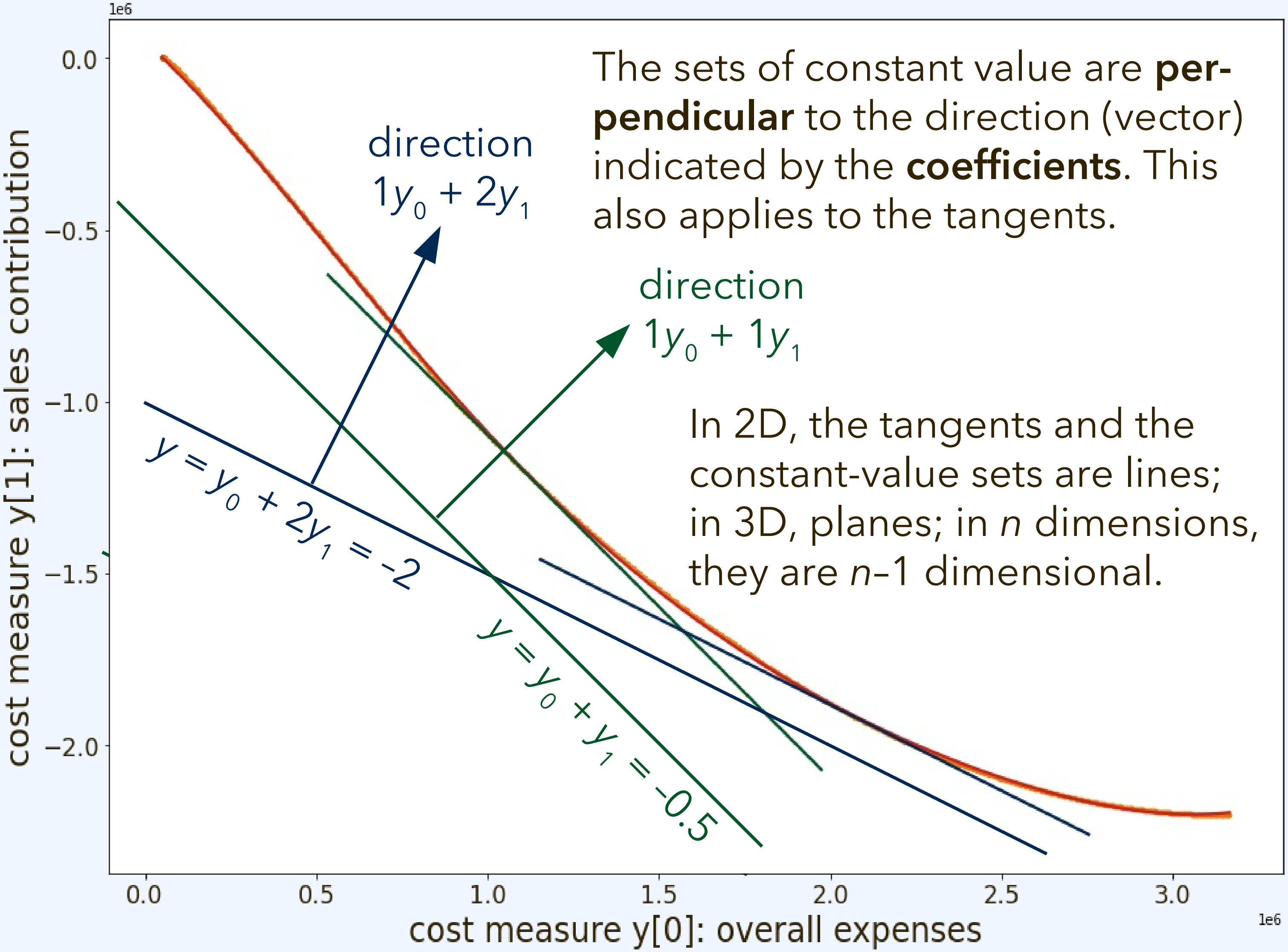
In Python it is possible to assign an *object reference to a function* to a variable. This means that functions can also be passed as arguments to other functions.

One common use of this is during optimization, using scipy.opt.minimize:



https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.minimize.html

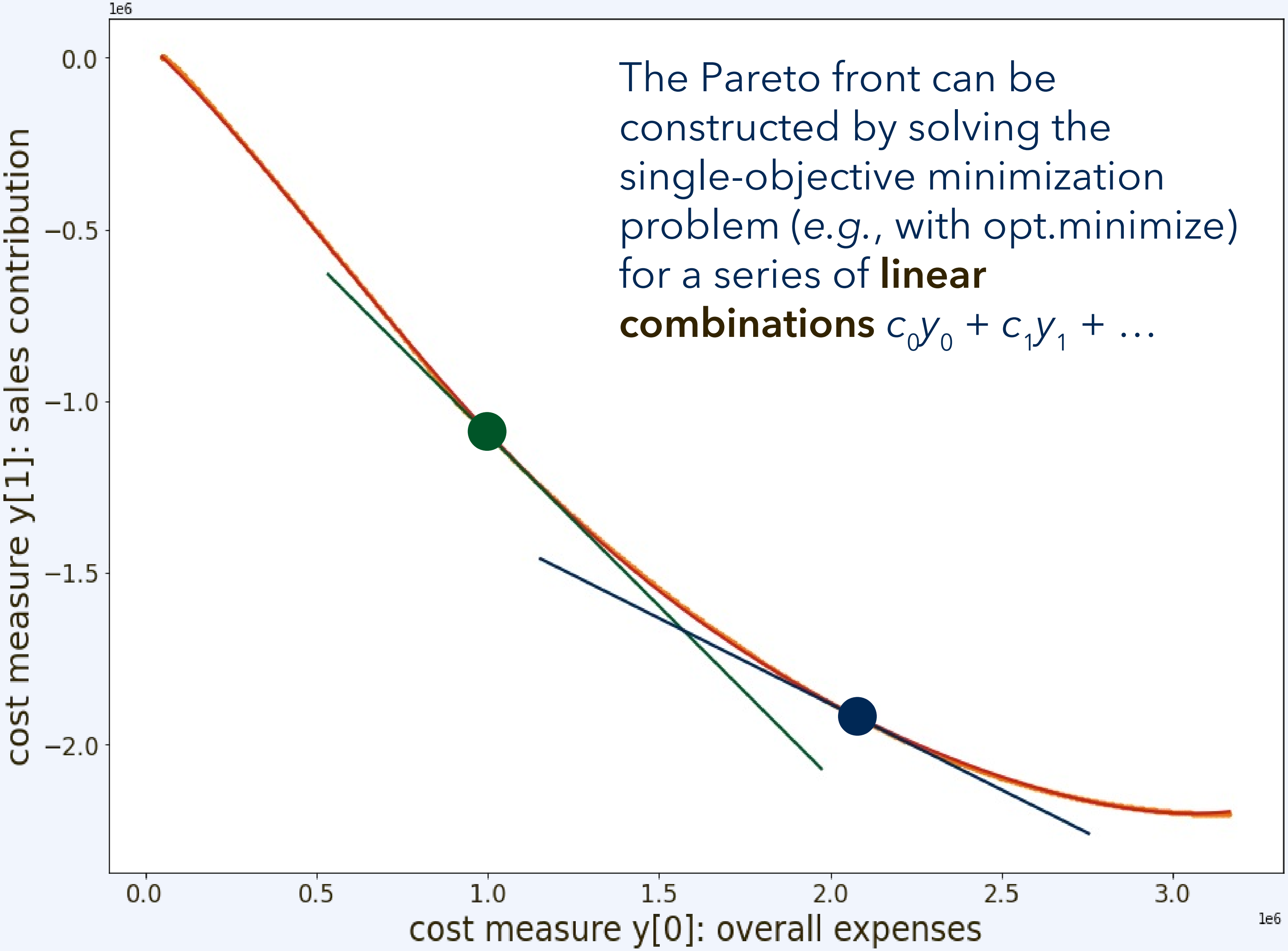
Linear combinations of objectives



The sets of constant value are **perpendicular** to the direction (vector) indicated by the **coefficients**. This also applies to the tangents.

In 2D, the tangents and the constant-value sets are lines; in 3D, planes; in *n* dimensions, they are *n*–1 dimensional.

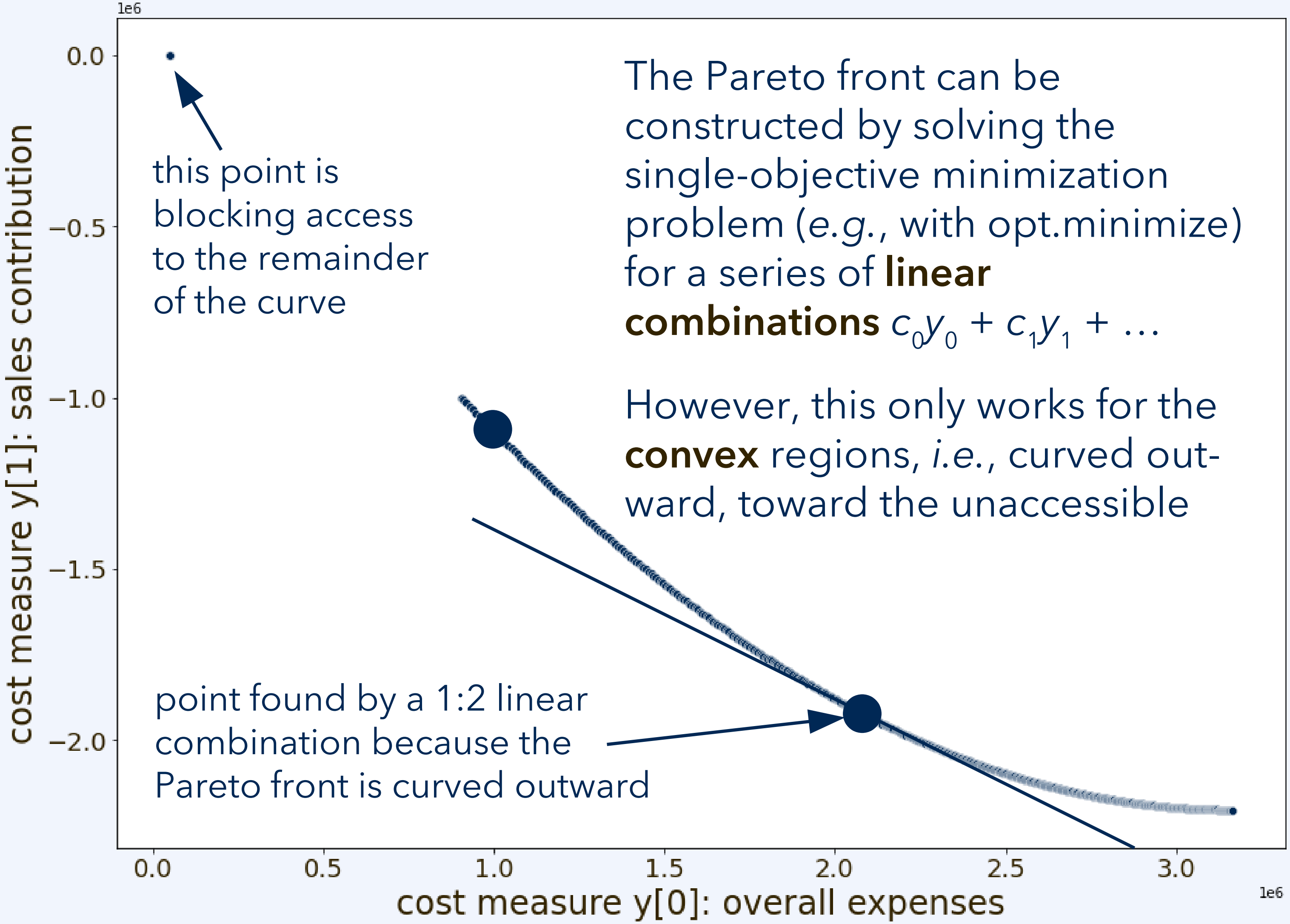
Linear combinations of objectives



The Pareto front can be constructed by solving the single-objective minimization problem (*e.g.*, with opt.minimize) for a series of **linear combinations** $c_0 y_0 + c_1 y_1 + \ldots$

“Sandwiching” by linear combinations

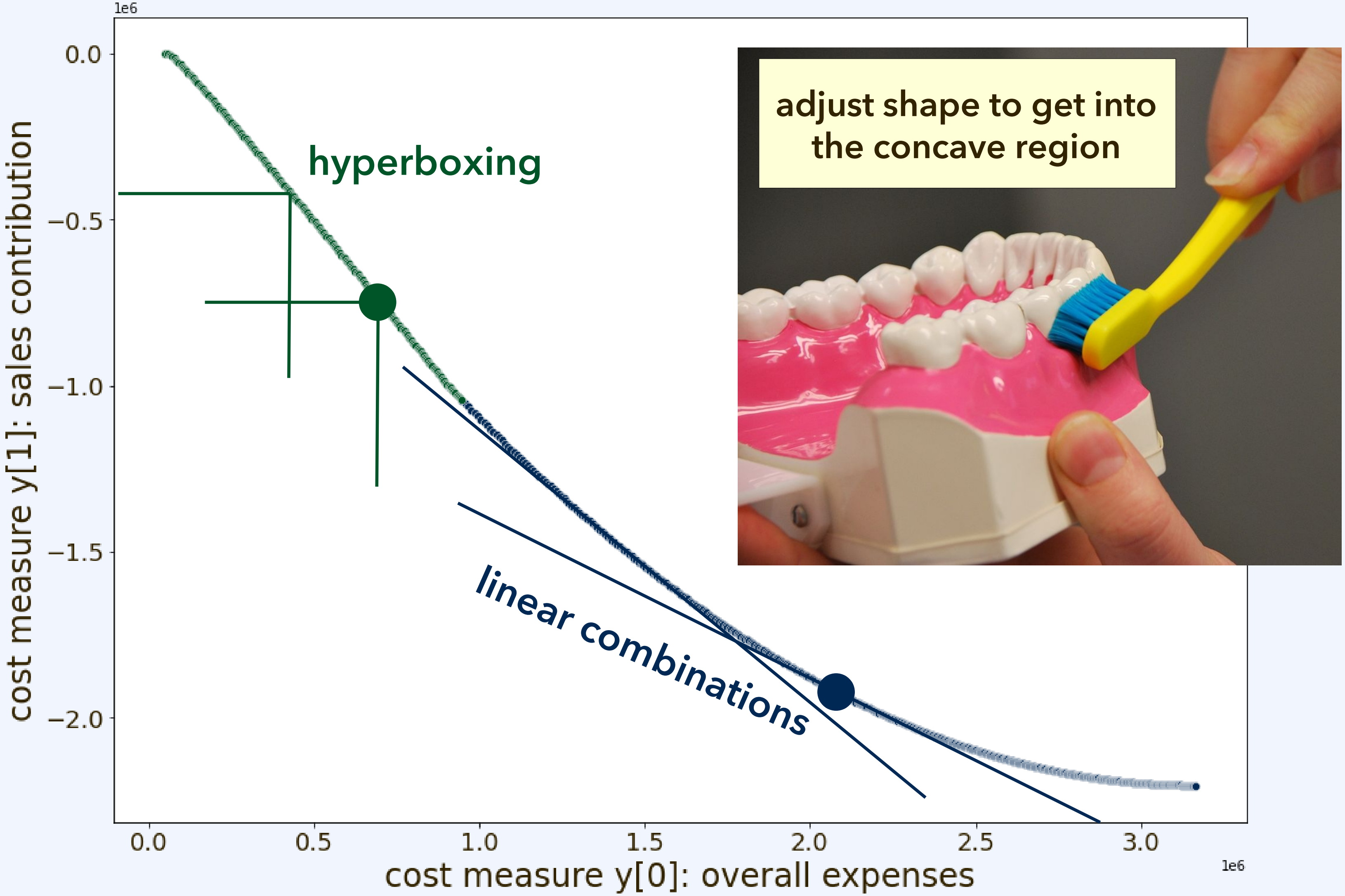
Norwegian University
of Life Sciences



The Pareto front can be constructed by solving the single-objective minimization problem (*e.g.*, with opt.minimize) for a series of **linear combinations** $c_0y_0 + c_1y_1 + \ldots$

However, this only works for the **convex** regions, *i.e.*, curved outward, toward the unaccessible

“Hyperboxing” for concave regions

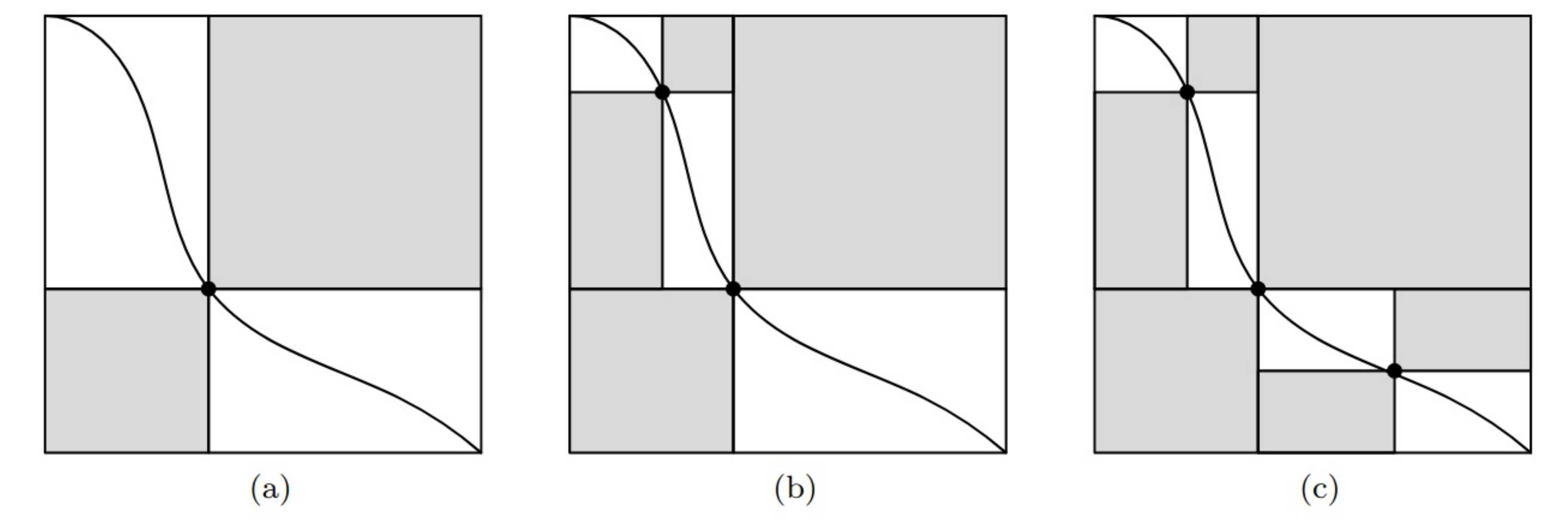


Sandwiching + hyperboxing method

Norwegian University of Life Sciences

Convex regions are detected by minimizing linear combinations $\sum_k c_k y_k$.

Concave regions are detected by hyperboxing.[1] Both tasks reduce to single-objective minimization, which can be handled using scipy.optimize.



[1]K. Dächert, K. Teichert, *An improved hyperboxing algorithm for calculating a Pareto front representation*, preprint, arXiv:2003.14249 [math.OC], **2020**.

Norges miljø- og biovitenskapelige universitet

Institutt for datavitenskap

Digitalisering på Ås

DAT121
Introduction to data science

5 Multidimensionality (multicriteria optimization)

5.1 Rational choice
5.2 Multiple objectives
5.3 Computing the Pareto front