Advertising budget allocation

Outline

Managerial motivation

Guided construction of the model

Worked example by hand

Case study and implementation

Results and verification of the KKT conditions

Sensitivity analysis

Exercises

Learning objectives (1/2)

At the end of this chapter you will be able to:

- model diminishing returns with concave functions;
- solve convex NLPs with the function constraints of Gurobi;

Learning objectives (2/2)

At the end of this chapter you will be able to:

- verify the KKT conditions numerically;
- build the budget value curve and use λ to negotiate the budget.

Prerequisites

- From the module *Convex Optimization*, Chapter 3 «Convex and concave functions»: concave functions and diminishing returns
- From this module, Chapter 2 «Background: convexity, quadratic programming and KKT conditions»: convex quadratic programs, the KKT conditions and the multipliers as shadow prices
- From this module, Chapter 3 «The solver for nonlinear models»: functional constraints for nonlinear functions in `gurobipy`

Managerial motivation

Introduction: Managerial motivation

- How should a campaign of 100.000 € be split among social, search, TV and influencers?
- Every channel shows **diminishing marginal returns**: the first euros yield a lot, the following ones less and less
- The response is therefore a **concave** function of the spend
- The optimum follows an elegant economic rule, which we prove here and verify numerically

A universal scheme

- Allocating a budget among uses with diminishing returns is a scheme that occurs everywhere:
  - marketing over several channels
  - funds among projects
  - time among activities
  - fertilizer among fields
- Theory says something strong and **verifiable**: at the optimum the marginal returns are equal to each other
- And this is exactly what the solver will return, digit by digit

Summary

- The problem: split a budget among channels with diminishing returns
- **Concave** response in the spend $\Rightarrow$ convex problem, certifiable global optimum
- Key rule: at the optimum the **marginal returns are equal**

Guided construction of the model

Introduction: Guided construction of the model

- We translate the allocation problem into a mathematical model
- We fix data, decision variables, objective function and constraints
- We then derive from the **KKT conditions** the economic rule that governs the optimum

The problem in words

- **We decide** how much to spend on each advertising channel
- **The objective** is to maximize the total expected response of the campaign
- **The constraints**:
  - the overall spend cannot exceed the budget
  - each channel has a maximum useful investment

Data (input of the model)

Symbol	Type	Meaning
n	$\in \mathbb{Z}_{\geq 1}$	number of channels, indexed by $i \in \{1, \ldots, n\}$
b	$\in \mathbb{Q}_{>0}$	total budget (thousands of €)
$r_i(\cdot)$	increasing concave function	expected response of channel i as a function of the spend
u_i	$\in \mathbb{Q}_{>0}$	maximum useful investment in channel i

In the case study $r_i(x) = a_i \log(1 + k_i\, x)$, with given $a_i, k_i \in \mathbb{Q}_{>0}$

Decision variables

We introduce the following n non-negative variables:

$$x_i = \text{spend on channel } i \text{ (thousands of €)}, \qquad \forall i \in \{1, 2, \ldots, n\}$$

Using these variables we build a convex NLP model for the problem

The model

Allocation model with concave response

$$\max \quad \sum_{i=1}^{n} r_i(x_i) \tag{8.1a}$$

$$\text{subject to} \quad \sum_{i=1}^{n} x_i \leq b, \tag{8.1b}$$

$$x_i \leq u_i, \quad \forall i \in \{1, 2, \ldots, n\}, \tag{8.1c}$$

$$x_i \geq 0, \quad \forall i \in \{1, 2, \ldots, n\}. \tag{8.1d}$$

The objective function

$$\max \sum_{i=1}^{n} r_i(x_i)$$

- It maximizes the **total response** of the campaign
- It is **separable** by channel: a sum of functions of a single variable
- It is a sum of **concave** functions, hence concave: the problem has a certifiable global optimum

The constraints

- $\sum_i x_i \leq b$: the overall spend does not exceed the budget; it is the **only constraint linking the channels** to each other — without it every channel would go to its own cap (one linear constraint)
- $x_i \leq u_i$: the caps per channel, beyond which the channel absorbs no further useful spend (n linear constraints)
- $x_i \geq 0$: they define the variables of the model

The economic condition at the optimum (1/2)

By the KKT conditions, at the optimum there exists $\lambda \geq 0$ (**shadow price of the budget**) such that, for every channel not stuck at the bounds 0 or u_i:

$$r_i'(x_i^*) = \lambda$$

In words: **the last euro invested yields the same in all the active channels**

The economic condition at the optimum (2/2)

- If this were not the case, moving one euro from the channel with the low marginal return to the one with the high marginal return would improve the response: we would not be at the optimum
- Channels **at the cap** u_i may have a marginal return $> \lambda$: we would like to invest more in them but we cannot
- Channels **at zero** have an initial marginal return $< \lambda$: they are not worth even the first euro

Summary

- Variables: spend x_i per channel; objective: $\max \sum_i r_i(x_i)$, concave and separable
- Constraints: budget $\sum_i x_i \leq b$ (the only one linking the channels) and caps $x_i \leq u_i$
- KKT condition: $r_i'(x_i^*) = \lambda$ for all the active channels
- λ is the **shadow price** of the budget

Worked example by hand

Introduction: Worked example by hand

- We solve by hand a case with only two logarithmic channels
- We use the condition of equal marginal returns and the budget constraint
- We finally compute the **shadow price** λ and give its managerial reading

Example: two logarithmic channels (1/3)

Data: $r_1(x) = 100\log(1 + 0.2x)$, $r_2(x) = 60\log(1 + 0.1x)$, budget $b = 50$ (thousands of €), without caps

Step 1 — condition of equal marginal returns:

$$\frac{100 \cdot 0.2}{1 + 0.2x_1} = \frac{60 \cdot 0.1}{1 + 0.1x_2} \quad \Longrightarrow \quad \frac{20}{1 + 0.2x_1} = \frac{6}{1 + 0.1x_2}$$

Example: two logarithmic channels (2/3)

Step 2 — budget. The budget is saturated, hence $x_2 = 50 - x_1$. Substituting:

$$20\,(1 + 0.1(50 - x_1)) = 6\,(1 + 0.2x_1)$$

$$120 - 2x_1 = 6 + 1.2x_1$$

from which

$$x_1 = \frac{114}{3.2} = 35.6, \qquad x_2 = 14.4$$

Example: two logarithmic channels (3/3)

Step 3 — shadow price.

$$\lambda = \frac{20}{1 + 0.2 \cdot 35.6} = 2.46$$

- One more euro of budget would yield ≈ 2.46 units of response
- And it yields the same whichever channel it is fed into: this is the meaning of the KKT condition

Summary

- Two equations, two unknowns: equal marginal returns + saturated budget
- Solution: $x_1 = 35.6$ and $x_2 = 14.4$ (thousands of €)
- Shadow price $\lambda = 2.46$: value of one more unit of budget
- The more "reactive" channel ($a_1 k_1 = 20 > a_2 k_2 = 6$) receives more spend

Case study and implementation

Introduction: Case study and implementation

- We move to the realistic case: four channels and a budget of 100 (thousands of €)
- We see how to handle the logarithm with the **function constraints** of Gurobi
- The problem stays convex: the optimum found is **global and certified**

Data of the case study

Four channels with $r_i(x) = a_i \log(1 + k_i x)$ and budget $b = 100$:

channel	a_i	k_i	cap u_i
social	260	0.14	60
search	380	0.09	80
TV	520	0.025	120
influencer	190	0.20	35

a_i measures the scale of the channel, k_i the **speed at which it saturates**

How the logarithm is handled

- For each channel we introduce two auxiliary variables:

$$g_i = 1 + k_i x_i, \qquad z_i = \log(g_i)$$

- The link $z_i = \log(g_i)$ is a **function constraint** of Gurobi (`addGenConstrLog`)
- The objective becomes **linear** in the z_i: $\max \sum_i a_i z_i$
- With `FuncNonlinear = 1` the logarithm is handled exactly, not by a piecewise linear approximation

Implementation in Gurobi

```
m.Params.FuncNonlinear = 1    # log handled exactly
x = m.addVars(4, ub=u)
g = m.addVars(4, lb=1.0)      # g_i = 1 + k_i x_i
z = m.addVars(4, lb=-GRB.INFINITY)
for i in range(4):
    m.addConstr(g[i] == 1 + k[i] * x[i])
    m.addGenConstrLog(g[i], z[i])
m.addConstr(x.sum() <= B)
m.setObjective(gp.quicksum(a[i] * z[i] for i in range(4)),
               GRB.MAXIMIZE)
```

Summary

- Case study: 4 channels, logarithmic responses, budget $b = 100$
- Reformulation: $g_i = 1 + k_i x_i$, $z_i = \log(g_i)$, objective linear in the z_i
- Gurobi handles the logarithm with `addGenConstrLog` and `FuncNonlinear = 1`
- Convex problem $\Rightarrow$ **certified global optimum**

Results and verification of the KKT conditions

Introduction: Results and KKT conditions

- We read the optimal solution channel by channel
- We verify **on the numbers** that the marginal returns coincide
- We check that λ is really the value of one more unit of budget

Results of the case study

```
Total response: 1.449.8 (thousands of useful contacts)
    channel |  spend |   cap | response |  marginal
     social |   24.3 |    60 |    385.3 |   8.2693
     search |   34.8 |    80 |    539.5 |   8.2693
         TV |   22.9 |   120 |    235.2 |   8.2693
 influencer |   18.0 |    35 |    289.8 |   8.2693

Check: +1000 EUR of budget -> response +8.244
```

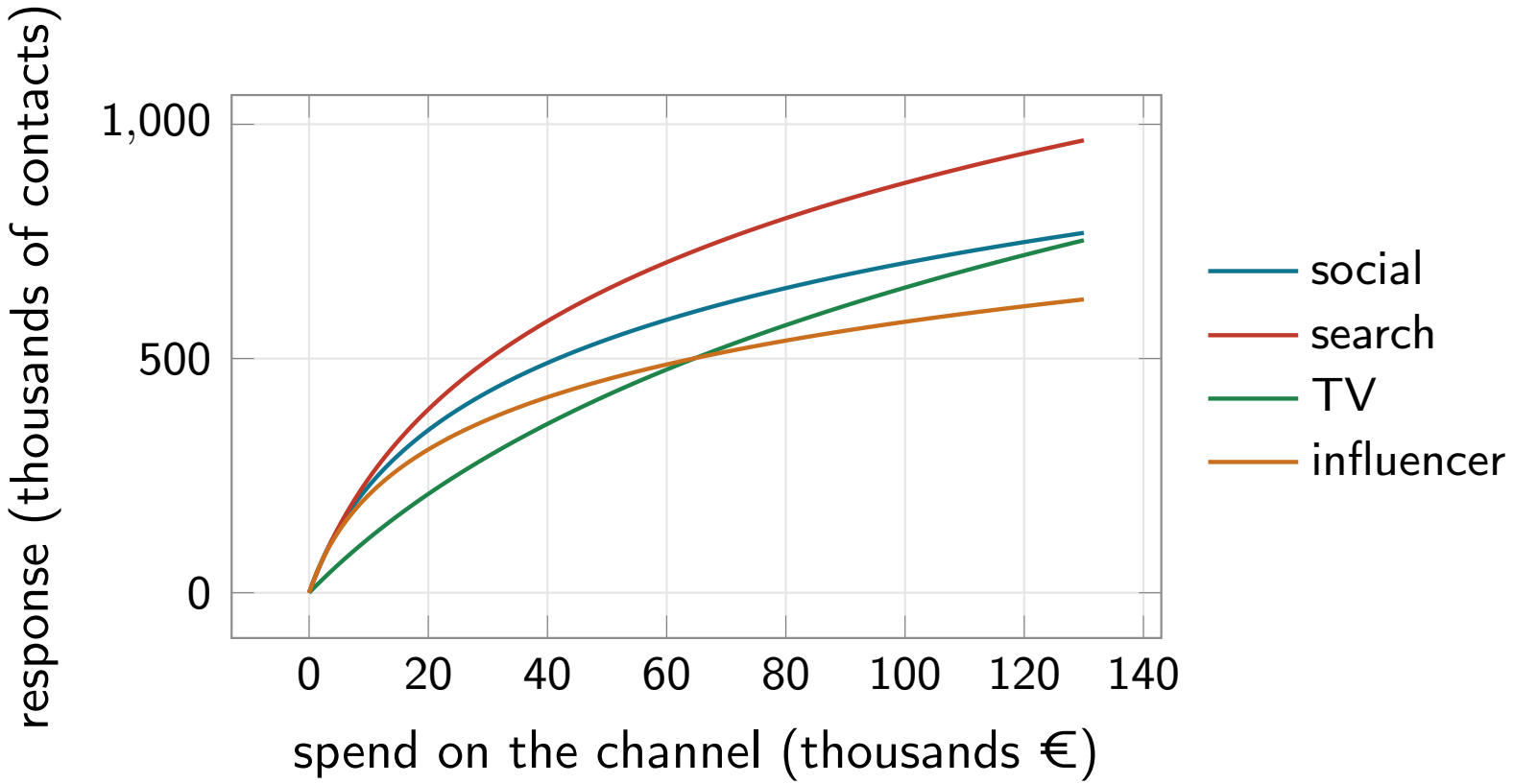
The theory can be seen in the numbers

- The four marginal returns coincide to the fourth digit: $\lambda = 8.2693$
- The *actual* increase in response obtained with one thousand euros more (8.244) confirms the reading of the multiplier
- No channel is at its cap: with $b = 100$ the budget is the only scarce resource

A managerial note

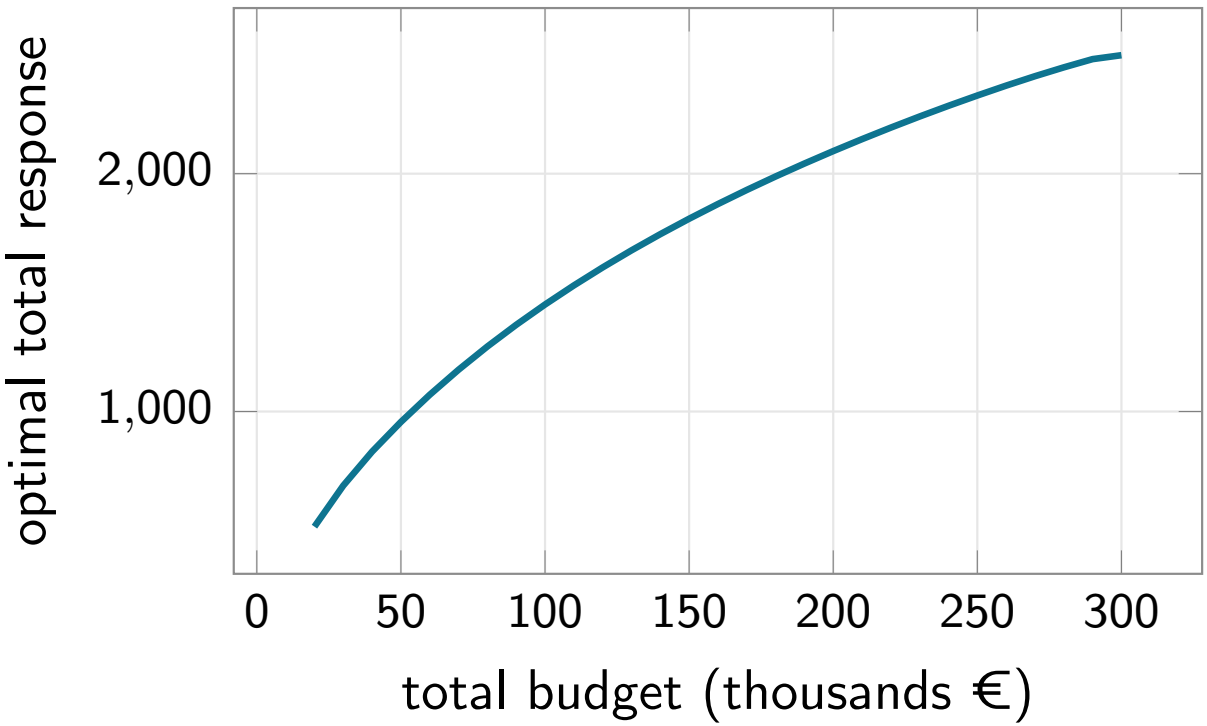
- TV receives *less* than social (22.9 against 24.3) despite its much higher cap
- What matters is not the **size** of the channel but the **speed at which it saturates** (k_i)
- TV has $k = 0.025$: its curve grows slowly, so its marginal return decreases slowly but starts lower relative to the spend invested

Concave response curves



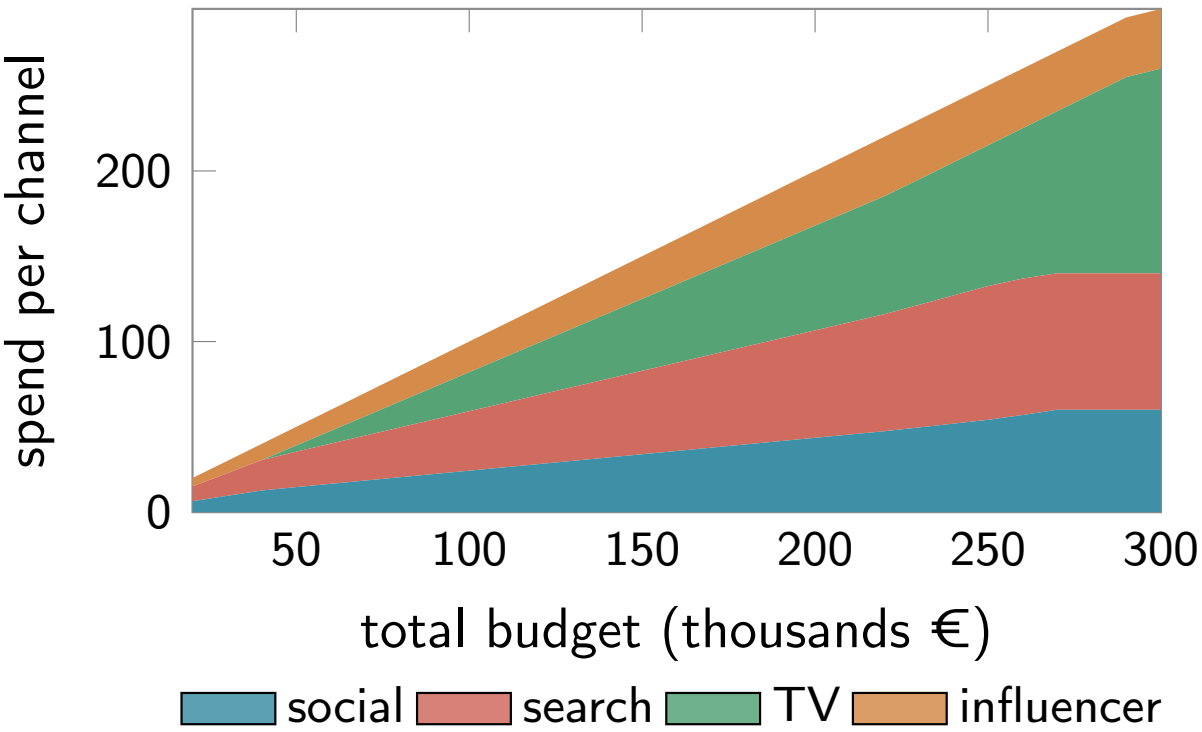
All concave, but with very different saturation speeds

Value curve of the budget



The value curve is concave: every tranche of budget yields less than the previous one, and its slope is exactly λ

Optimal mix as the budget grows



The channels grow together keeping their marginal returns equal; the influencer channel saturates first at its cap (35)

Summary

- Optimal response 1.449.8 with $b = 100$; no channel at its cap
- The four marginal returns coincide: $\lambda = 8.2693$, as the theory predicts
- The check by perturbation (+1000 €) gives +8.244: consistent with λ
- What determines the mix is not the size of the channel but k_i

Sensitivity analysis

Introduction: Sensitivity analysis

- We let the budget vary and observe how the optimal response and λ change
- We will see that λ **decreases** as the budget grows, until it vanishes
- It is the curve needed to **negotiate** the budget with the finance department

Response and λ as the budget varies

```
b =  20: response   515.9   lambda = 18.971
b =  60: response 1.070.4   lambda = 10.909
b = 100: response 1.449.8   lambda =  8.244
b = 180: response 1.988.2   lambda =  5.538
b = 260: response 2.370.2   lambda =  4.049
b = 300: response 2.498.1   lambda =  0.000
```

At $b = 300$ all the channels are at their caps

Reading the table

- The budget value curve is **concave**: λ falls from 19 to 4 as the budget grows
- At $b = 300$ all the channels are at their caps ($60 + 80 + 120 + 35 = 295$)
- The budget stops being the scarce resource and λ **collapses to zero**: the last euros no longer find a useful use

How λ is used in a company

- For a CFO the *curve* of λ is the quantitative argument for deciding how much budget to grant to marketing
- Rule: the campaign is funded as long as λ exceeds the value of one euro invested elsewhere
- No need for a negotiation by gut feeling: the model produces the **price** of every additional euro

Summary

- The optimal response grows with the budget, but in a **concave** way
- λ decreases: from 18.97 with $b = 20$ to 8.24 with $b = 100$
- When all the channels are at their caps, $\lambda = 0$
- The curve of λ is the tool for negotiating the budget

Exercises

Introduction: Exercises

- We test what we have seen with four exercises
- They revisit the example by hand, the alternative functional forms and the role of λ
- For each one: first the statement, then the solution

Exercise: check

- Solve the example by hand with $b = 80$
- Where does the additional budget end up?
- How does λ change?

Solution (1/2)

From the condition $r_i'(x_i) = \lambda$ we obtain

$$x_1 = \frac{100}{\lambda} - 5, \qquad x_2 = \frac{60}{\lambda} - 10$$

and imposing $x_1 + x_2 = b$:

$$\lambda = \frac{160}{b + 15}$$

With $b = 80$: $\lambda = 160/95 = 1.68$ (against 2.46 with $b = 50$)

Solution (2/2)

- The optimal spends become

$$x_1 = 54.4, \qquad x_2 = 25.6$$

- The 30 additional units are split in the proportion 100 : 60, that is 18.75 to channel 1 and 11.25 to channel 2
- No channel is abandoned: both grow keeping their marginal returns equal, and λ decreases

Exercise: exponential response

- Replace the response of TV with

$$r(x) = 520\left(1 - e^{-0.025x}\right)$$

- Repeat the optimization: does the mix change?
- The function has a natural ceiling: is the bound u_i still needed?

Solution (1/2)

- Exponential marginal return: $r'(x) = 13\, e^{-0.025x}$, against $13/(1 + 0.025x)$ of the logarithm
- Same initial value (13), but the exponential **decays much faster**
- From $r'(x) = \lambda$ it follows that $x_{\mathrm{TV}} = 40 \log(13/\lambda)$

Solution (2/2)

- Imposing the budget $b = 100$ we obtain $\lambda \approx 8.00$ and the mix

$$(25.4,\ 36.4,\ 19.4,\ 18.8)$$

- TV drops from 22.9 to 19.4: the spend moves to the other channels, total response $\approx 1.442.7$
- The bound is no longer needed for TV: the function saturates by itself at 520. It remains useful only if u_i represents a *contractual* or inventory limit

Exercise: excluded channel

- Add a fifth channel with initial marginal return $a_5 k_5 = 5 < \lambda$
- Verify that the optimum leaves it at zero
- Interpret its "reduced cost" $a_5 k_5 - \lambda$

Solution

- The marginal return of a logarithmic channel is maximum at $x = 0$ and equals $a_5 k_5 = 5$
- Already the first euro yields $5 < \lambda = 8.27$: it is better to leave it to the other channels, hence $x_5^* = 0$
- The reduced cost $a_5 k_5 - \lambda = 5 - 8.27 = -3.27$ measures how much is **lost** for every unit forced on that channel
- The channel would enter only if the budget grew enough to push λ below 5

Exercise: fair coverage

- The budget serves three customer segments, with different coverage per channel
- Maximize the coverage of the **worst segment**:

$$\max z \quad \text{with} \quad z \leq \text{coverage}_g \quad \forall g$$

- How much does fairness cost in terms of total response?

Solution (1/2)

- The max-min model is written with one variable z and one constraint per segment:

$$\begin{aligned} \max \quad & z \\ \text{subject to} \quad & z \leq \sum_i c_{gi}\, r_i(x_i), \qquad \forall g, \\ & \textstyle\sum_i x_i \leq b, \quad 0 \leq x_i \leq u_i \end{aligned}$$

- It remains a **convex** problem: the minimum of concave functions is concave

Solution (2/2)

- The optimum moves spend towards the channels that reach the weakest segment, even if they are less efficient in aggregate
- The **price of fairness** is measured by comparing the two solutions:

$$\text{total response}^{\max \sum} - \text{total response}^{\max \min}$$

- This is the correct way to present the trade-off to the management: not "fair or efficient", but *how much* fairness costs

Final summary (1/2)

- Diminishing returns $\Rightarrow$ **concave** response $\Rightarrow$ convex NLP with a global optimum
- The model: $\max \sum_i r_i(x_i)$ with budget $\sum_i x_i \leq b$ and caps $x_i \leq u_i$
- Golden rule (KKT): at the optimum $r_i'(x_i^*) = \lambda$ for all the active channels

Final summary (2/2)

- Gurobi solves it with the function constraints (`addGenConstrLog`, `FuncNonlinear = 1`)
- In the numbers: four marginal returns equal to 8.2693 and a check by perturbation of $+8.244$
- λ decreases with the budget and vanishes when all the channels are at their caps
- What counts is not the size of the channel but its saturation speed k_i