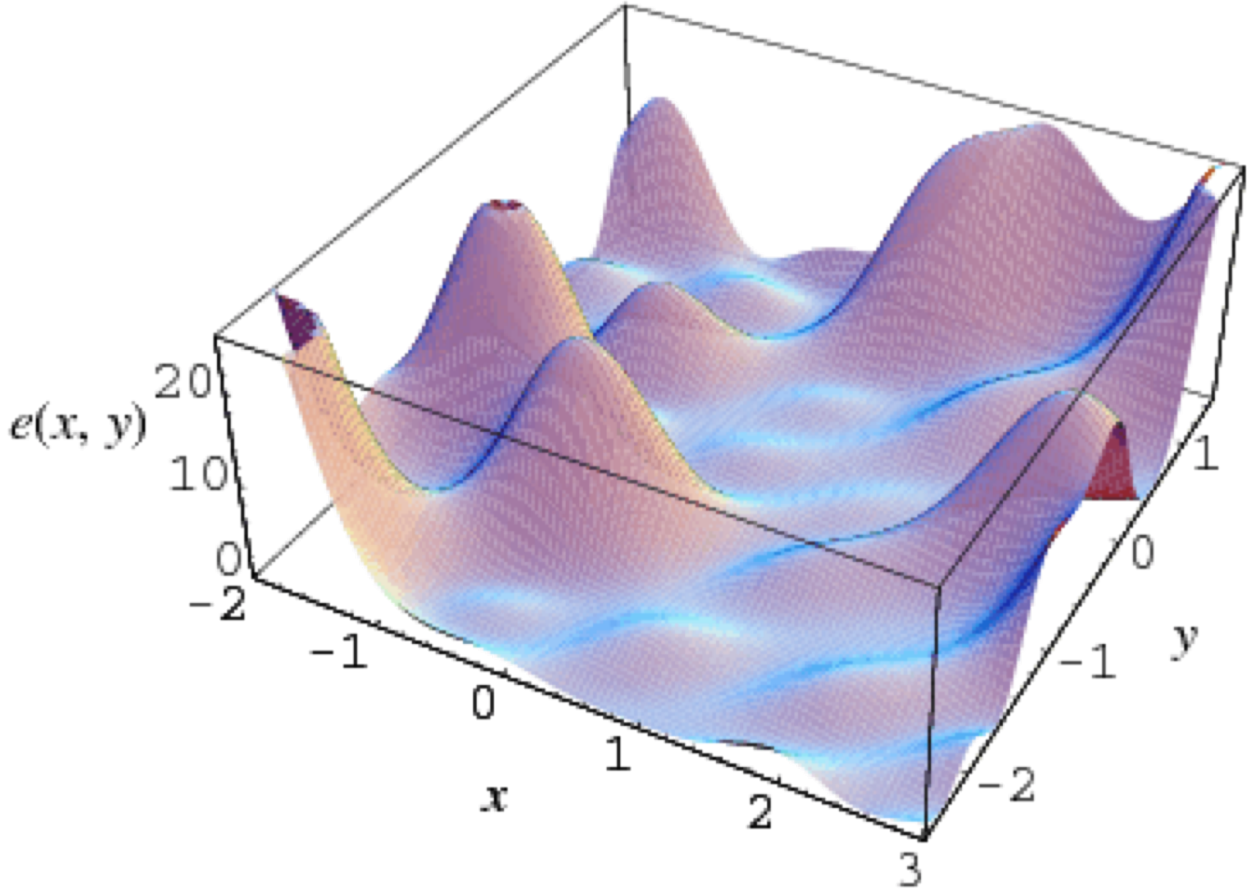
Building Economic Models of Human-Computer Interaction Part II

by @leifos and @guidozuc

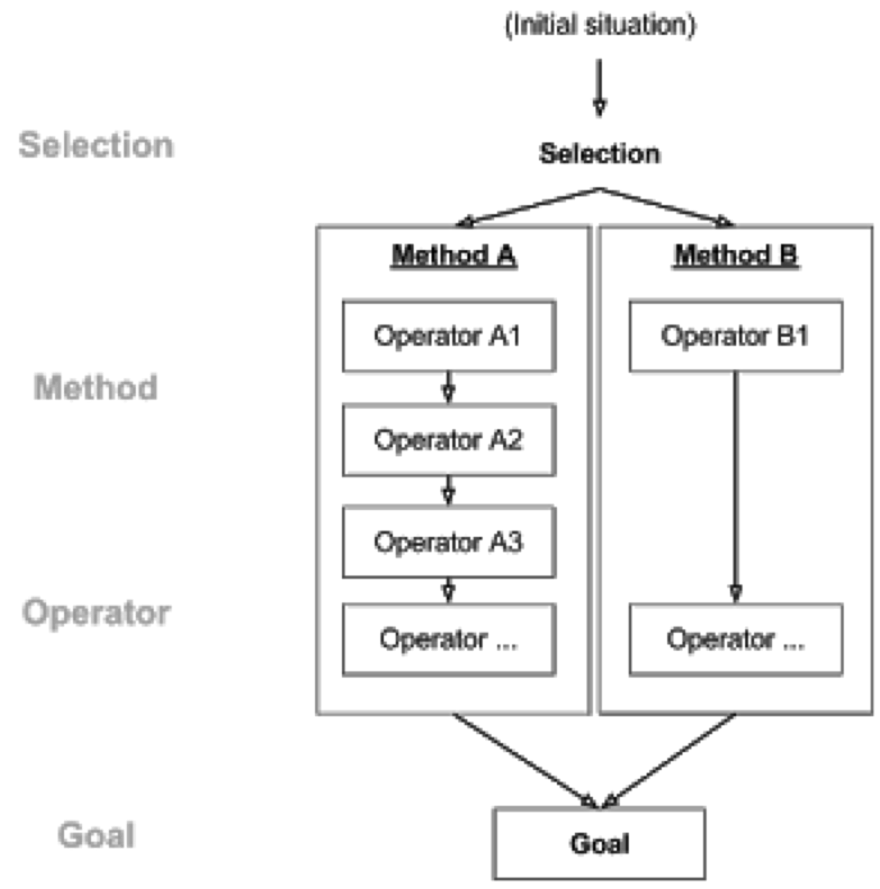


REDUCING FRICTION

Making interfaces more usable

GOMS

- **G**oals,
- **O**perators,
- **M**ethods and
- **S**election Rules
- One of the first approaches to modelling the time taken to perform a task.
  - Lots of variants on GOMS



Card, Moran & Newell (1983)

GOMS-KLM

- **Keystroke Level Model**
  - Aims to model the low level costs of interaction
  - Helps to find ways that are more efficient or better ways to complete a task
    - Analyze the steps involved in a given process
    - Re-arrange or remove unneeded steps
  - Well suited to **very goal directed tasks** that are completed in a short period of time
  - A **method** is **modelled** as a **sequence** of **operations**

Card et al (1980)

GOMS-KLM

Operator		Description	Time (s)
K	Press a Key or button (including shift, control), time varies with user skill	Best typist (135 wpm)	0.08
		Good typist (90 wpm)	0.12
		Avg skilled typist (40 wpm)	0.20
		Avg no-secretary typist (40 wpm)	0.28 or...
		Typing random letters	0.50
		Typing complex codes	0.75
		Worst typist (unfamiliar with keyboard)	1.20
P	Point with a mouse	Point with a mouse (range is 0.8 to 1.6 sec, not including button press)	1.10
H	Home to/from keyboard or other device		0.40
D (nd, ld)	Draw nd straight lines of total length ld		9 nd + 0.16 ld
M	Mentally prepare		1.35 Olson & Nilsen say 1.62
S	Scan	e.g., find coordinates of spreadsheet, not in original KLM	2.29
R(t)	Response by system	Varies with command, including wait if required	t

GOMS-KLM

- Prediction tend to only be valid for expert users who never make errors
- Only provides one view for evaluating the design (cost)
  - Other aspects usefulness, enjoyment, etc. are not considered.
- Model is very low level, and cumbersome
  - Though a variant Quick & Dirty GOMS presents models a simple tree consisting of subtasks
  - And the parent node predicts the time.

Cost Models

- Here we will be taking a more practical approach and defining cost models at an appropriate level of abstraction
  - Depends on the interface, interaction, scenario, etc.
  - For example a keystroke level is appropriate for text entry
  - But, cost of entering a query, viewing snippets, selecting items, could be modelled as a higher level

COST BENEFIT ANALYSIS

A super efficient overview

Cost Benefit Analysis (CBA)

- Aims to **estimate and total up** the **value** of the **benefits** and the **costs** associated with a particular **decision/choice**.
- Provides the **basis** for the **comparison** of **decision/choices**.
- **Assumption**: benefits and costs must be formulated in the same unit of measurement
  – However, we can perform a cost-effectiveness analysis if costs and benefits are in different measurements.

Applying CBA in CHI

- List the alternative decisions/choices
- List the stakeholders
  - For simplicity we will consider only one stakeholder e.g. the user
  - But we could consider other users/collaborators, advertisers, vendors, etc. too.
- Select a measurement and measure all the cost/benefit elements
- Apply discount rate (if appropriate)
- Calculate the net present value.

Speak or Type

- Alternatives:
  - (a) ask Siri,
  - (b) type in the question yourself
- Unit of Measurement
  - Time Spent
  - Quality of Response (given the input)
- Discount Rate:
  - Users prefer to receive a good response sooner (so each additional interaction means the benefit is discounted, by say 1/#interactions)

Speak or Type

- Given a ranking of documents

Ask	**Error**	**Ask**	**Error**	**Ask**	**Ans**

- We assume that the user receives some gain for each document (gain = benefit - cost)

Ask	**Ans**	1	0	0	2	0	4	0	0

- Discount the value the further down the ranked list (say 1/rank)

Type	**Ans**	0.33	0.25	0.2	0.16	0.14	0.12	0.11	0.1

- Compute NPV e.g (3*1 + 1*.33 + 2*.16 + 4*.12) = 4.16

Type

- Given the sequence of interactions:

Type	**Ans**

- We assume that the user receives some payoff for each document (payoff = benefit - cost)
  - Let's say: Payoff(type) = -20, Payoff(Ans) =100

-20	**100**

- Discount the value

1	**0.5**

- Compute NPV
  - (-20*1 +100*0.5) *= 30*

Speak

- Given the sequence of interactions when speaking:

Ask	**Error**	**Ask**	**Ans**

- We assume that the user receives some payoff for each document (payoff = benefit - cost)
  – Let's say: Payoff(Ask) = -10, Payoff(Error)=-5, Payout(Ans)=100

-10	**-5**	-10	100

- Discount the value

1	**0.5**	0.33	0.25

- Compute NPV

  (-10*1 -5*0.5 -10*.33 + 100*.25) *= 9.17*

- *But if there is no error then the the NPV(speak) = 40*

Cost Benefit Analysis

- Compute the NPV of different methods to determine which alternative is preferable
- But, there is no notion of uncertainty
  - What if there is some probability of an error
  - How good does the system need to be, before you would always speak to the agent?
- We need to consider the uncertainty associated with payoffs.

DECISION THEORY

A super quick introduction

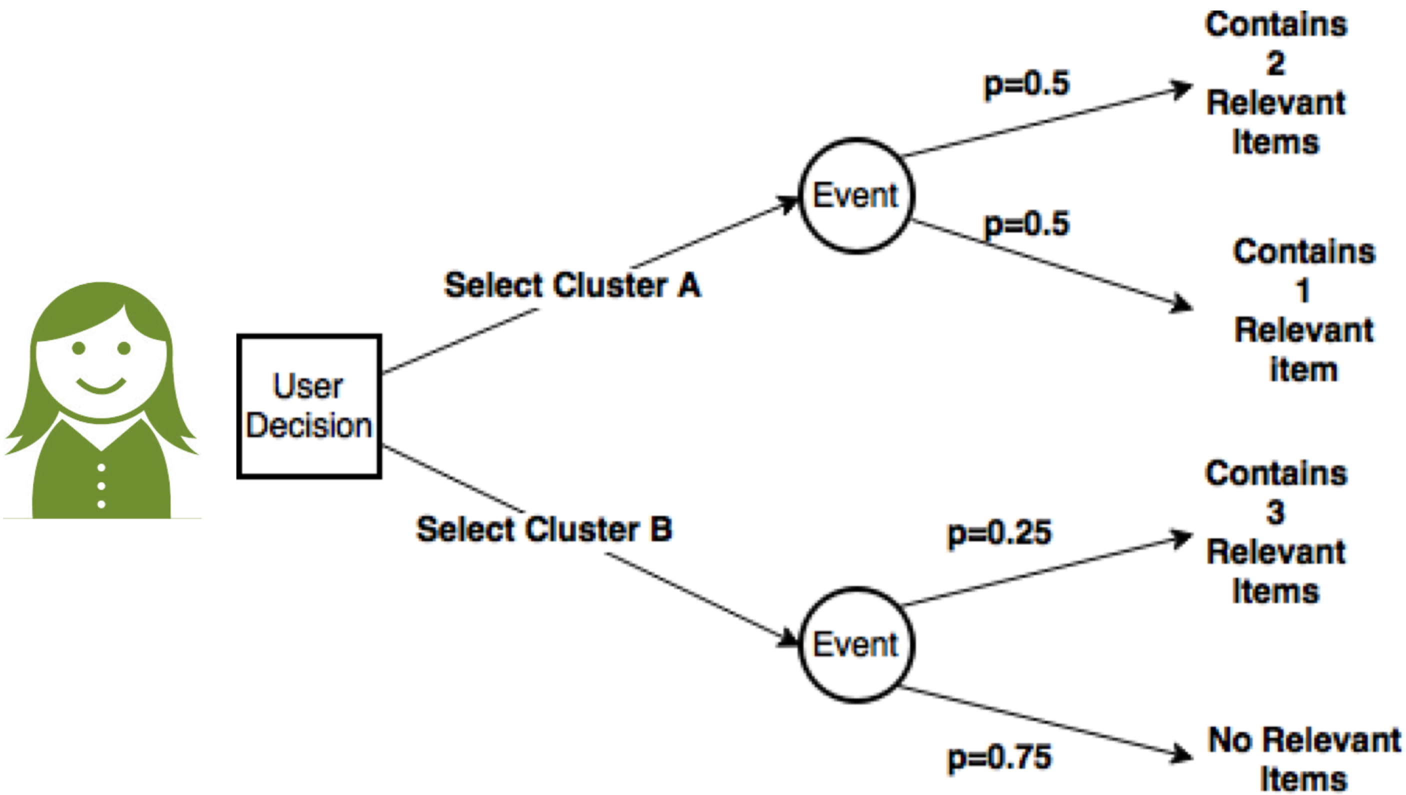
Decision Theory

- Extending the idea of CBA
- **Decision Theory** considers decision problems
  - where the goal is to select the best available/known alternative.
  - Often under uncertainty
- **Example:** You have been given the choice between using *Google* or *Yahoo!*
  - Which one would you use to search the web?
  - Which one would you use to read the news?
  - Which one would you use for a joke?

Decision Theory

- There are **four basic elements:**
  - **Acts**: the choices/decisions considered by the user
  - **Events:** occurrences taking place outside the control of the user
  - **Outcomes:** the result of the occurrence of acts and events
    - Usually have some probability of occurring
    - i.e. Uncertainty in the outcome
  - **Payoff**: the value the user places on the occurrences
    - Payoff = Benefit - Cost
- It is often useful to represent the decision problem as a tree.

Browsing Clusters Example



- **Actions**: User can select cluster A or B
- **Events**: System responds with documents
- **Outcomes**: With some probability different amounts of relevant items are returned
- **Payoffs**: The benefit minus the cost for each outcome.

Expected Value

- Expected Value of an Event is:

$$E[event] = \sum_{outcomes} p(outcome) \times g(outcome)$$

  – Where ***p*** is the probability and ***g*** is the gain.

- Decisions:
  – Select Cluster A: Expected Payoff is: 0.5 * 2 + 0.5 * 1 =1.5
  – Select Cluster B: Expected Payoff is: 0.25 * 3 + 0.0 = 0.75
- Since the expected payoff of **A** is greater the **B**, then the user should select **A**.

EXAMPLE

Page Finding

Page Finding

- Your friend has recently completed a marathon, and you have found the page of times for runners.
  - They are ordered by time.
  - You would like to know how fast your friend completed the marathon.
  - However, there are thousands of runners in the list.
- Actions:
  - (a) Scroll through list until you find friends name, or
  - (b) using the Find Command, type friends name

Example adapted from Russell

Scrolling

- **Action (a) – Scrolling**
  - Outcomes:
    - (1) Finds correct name (*p=1.0*)
  - Payoffs in terms of costs only
    - (1) on average examine about ***n/2*** runners
  - Notes and Assumptions
    - To examine **1** runner takes **2** seconds
  - Total Cost = ***n*** seconds.

Finding

- **Action (b) – Find Command**
  - Outcomes:
    - (1) Finds correct name (*p=1.0*)
  - Payoffs in terms of costs only
    - (1) on average examine about ***m / 2*** runners
  - Notes and Assumptions
    - To reduce down to **m** runners the user enters **k** letters
      - Where $m = n/(k+1)^2$
    - To examine 1 runner takes 2 seconds.
    - To enter 1 letter takes 2 seconds.
    - To switch to Finding takes 5 seconds.

Scroll vs Find

- **Expected Cost for Finding** = m + 2k + 5
  - Where $m = n/(k+1)^2$
- **Expected Cost for Scroll** = n
- Which action should the user take?
  - Compare the costs
  - Scroll, if Total Cost of Scrolling is less than Total Cost of Finding e.g.,
  - Scroll, if $n < m + 2k + 5$
- Homework – do the math ☺

Scrolling with Uncertainty

- Action (a) – Scrolling
  - Outcomes:
    - (1) Finds correct name (p=0.9)
    - (2) Finds incorrect name (p=0.05)
    - (3) Misses name (p=0.05)
  - Payoffs in terms of costs only
    - (1) on average examine about ***n/2*** runners
    - (2) on average examine about ***n/2*** runners
    - (3) examines all ***n*** runners
    - To examine 1 runner takes 2 seconds.

Finding with Uncertainty

- Action (b) – Find Command
  - Outcomes:
    - (1) Finds correct name (p=0.98)
    - (2) Finds incorrect name (p=0.01)
    - (3) Misses name (p=0.01)
  - Payoffs in terms of costs only
    - (1) on average examine about ***m / 2*** runners
    - (2) on average examine about ***m / 2*** runners
    - (3) examines all ***m*** runners
    - To reduce down to **m** runners the user enters **k** letters
      - Where $m = n/(k+1)^2$ (assumption)
    - To examine 1 runner takes 2 seconds.
    - To enter 1 letter takes 2 seconds.

Scroll vs Find

$$C_s = \underbrace{\frac{2n}{2} * 0.9}_{\text{correct}} + \underbrace{\frac{2n}{2} * 0.05}_{\text{incorrect}} + \underbrace{2n * 0.05}_{\text{missed}} = 1.05n$$

$$\begin{aligned} C_f &= \overbrace{\frac{2m}{2} * 0.98}^{\text{correct}} + \overbrace{\frac{2m}{2} * 0.01}^{\text{incorrect}} + \overbrace{2m * 0.01}^{\text{missed}} + \overbrace{2k}^{\text{letters}} + \overbrace{5}^{\text{CTRL+F}} = \\ &= 1.01m + 2k + 5 \end{aligned}$$

If ***n*** is large, then finding is cheaper than scrolling.

And the user is more likely to get to the correct name & time via the find command.

Cost-Effectiveness Ratio

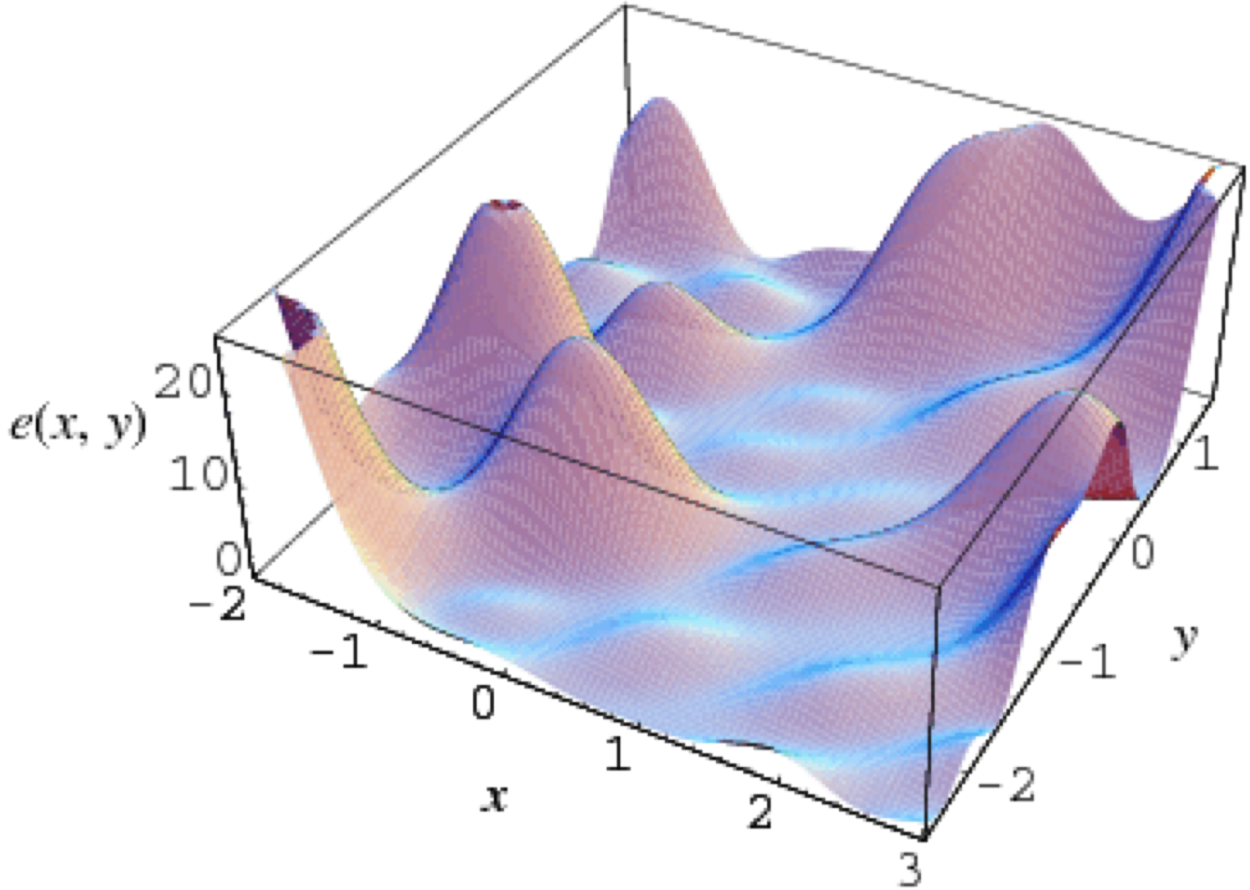
- Since our costs are in time, and our gain is based on whether we find the answer or not, then we need to consider the ratio, e.g.

$$\frac{B_a}{C_a}\ vs. \frac{B_b}{C_b}$$

- The ratio that is higher, is therefore, the better decision – assumes that people want to maximize their rate of gain
- We will see that **Information Foraging Theory** uses a similar ratio (e.g. effectiveness-cost ratio) where the gain is divided by the cost.

Defining Costs and Benefits

- **Costs** and **benefits** have been referred to in a variety of ways
- **Benefit**: happiness, enjoyment, satisfaction, gain utility, expected utility, usefulness
- **Cost**: mental/cognitive, physical, financial, temporal
  - But often time is used as a proxy for cost
- Generally the costs and benefits are considered to be **common but abstracted unit.**
- **Estimating costs and benefits is a major challenge.**



ECONOMIC / OPTIMIZATION MODELS

Optimization Models

- Provide a powerful tool for analyzing the designs of organisms, artifacts and systems.
- Key to an optimization model is:
  - **an objective function**,
  - **Profit/utility/benefit function**
  - **Cost function** and
  - any **constraints/requirements** that need to be satisfied.

Hillier & Lieberman (2001)

Murty (2003)

Optimization Problem

- For example
  - Imagine that you are studying for a test, and you have summaries, lectures, and papers.
  - How much time should you spend reading through each resource type?
  - It is the day before the exam, so you have about 10 hours to revise.
  - Your objective is to maximize how much you know about the course.

Objective Function: what to optimize

- The objective function is a mathematical model that we want to maximize or minimize
  - i.e. Maximize the profit, for a fixed cost
  - Or minimize the cost for a given level of output
  - Thus they generally take the form of min/max some function subject to some constraints
  - The task determines the objective function that is used.

Hillier & Lieberman (2001)

Murty (2003)

Optimality and Rationality

- **Optimization models** often **assume** that **human behavior** is **rational**
  - Perfect information, Infinite computational power
- However, models can be developed with more **realistic assumptions** of human behavior:
  - **Bounded rationality**
  - **Satisficing** which can be considered as local optimization
  - Imperfect information and constraints

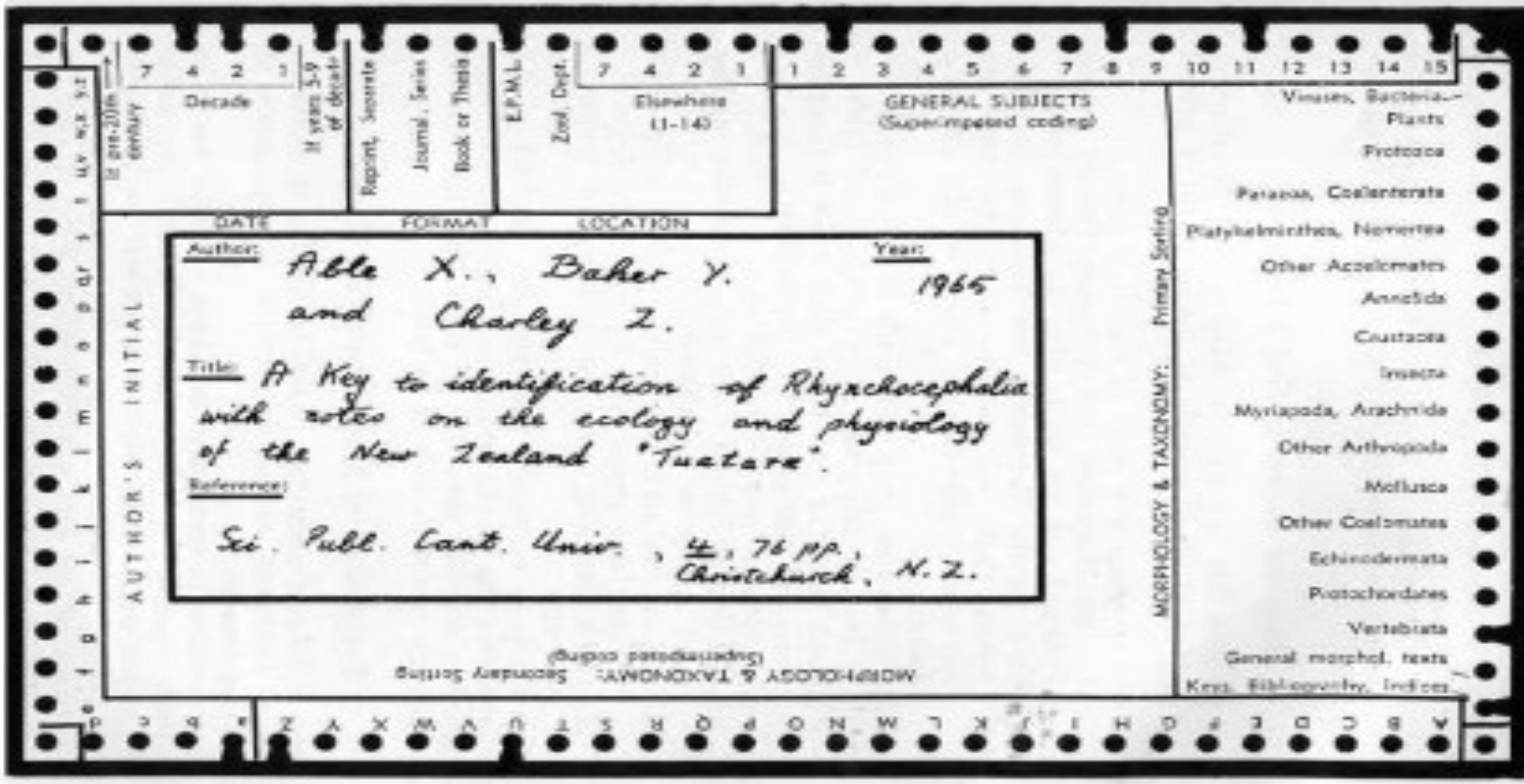
Simon (1972)

Simon (1955)

Stigler (1961)

Optimization Models

- They **shouldn’t be applied naively**
  - But can be used to expand our understanding of the interactions
- They **do not imply users are** able achieve the **optimal** in a particular scenario/task
- Can be used to determine how well a person could perform
  - i.e. how much they deviate from the optimal



EXAMPLE

User versus System

Time to spend searching?

- Cooper wondered:
  - how much time a **user** should spend searching, and
  - how much time the **system** should spend searching?
- (library) systems at the time were mechanized, also employed librarians, etc.
- What is the most **economic division** of **effort** b/w **user** and **system**?

User-System Interaction

- A **user** can choose from a range of **information seeking strategies**
- The **user's time** is an **economic quantity**
  - i.e. cost
- The **user** pursues a **particular strategy** until the **cost** incurred **exceeds** the **utility received**,
  - At this point the user may choose another strategy
  - Or they stop

Cooper (1972)

The trade-off

- A system needs to consider more than just matching, it also needs to consider:
  - The **cost** of the **search** to the **system**
  - The **cost** of the **search** to the **user**
  - The **benefit** to the **user**
  - The **most economic division** of **effort** between the **user** and the **system** to accomplish the user's search goals and objectives

Cooper (1972)

Variables that influence cost

- **Time Spent**
  - Time spent at the console
  - Time required to map request into query language
  - Time waiting for system response
- **System Design**
  - The design of the console, its flexibility, responsiveness, features, etc
- **Results Quality**
  - The quality of the results, their presentation, etc
- **Total Cost** is a combination of these factors
  - What system (or method of accessing the relevant information) is determined by a cost-benefit function.

User-System Time Trade-off Model

- Total Cost: $C_t = t_u.c_u + t_s.c_s$

    tu – time user spent in seconds

    cu – cost per second to the user

    ts – time system spent in seconds

    cs – cost per second to the system

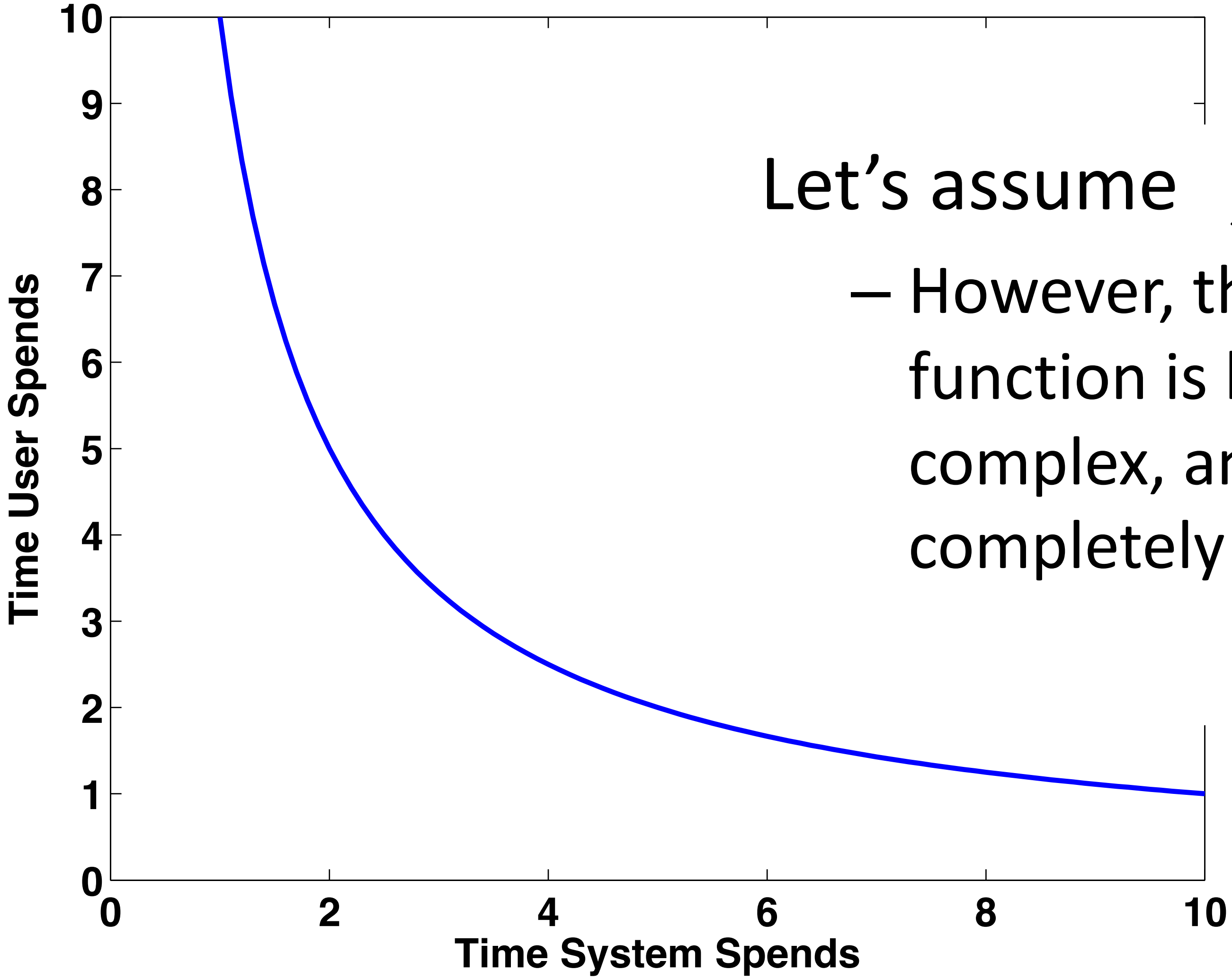
- ***P*** – performance resulting from the user-system interaction.

    where

$$P = f(t_u, t_s)$$

Cooper (1972)

User-System Time Trade-off Model



Let's assume $P = t_u . t_s$

- However, the actual function is likely to be more complex, and/or a completely different shape.

Cooper (1972)

Optimal Division of Time

- Given the model:

$$P = t_u . t_s$$

- it is possible to determine optimal level of $\mathbf{t_s}$ and $\mathbf{t_u}$ that **minimizes** the **total cost** for a given level of performance.

- i.e. This is the objective function!

Cooper (1972)

Optimal Division of Time

- Given the model: $P = t_u . t_s$

it is possible to determine optimal level of $\mathbf{t_s}$ and $\mathbf{t_u}$ that **minimizes** the **total cost** for a given level of performance.

$$T_u{}^* = \sqrt{\frac{P.c_s}{c_u}}$$

$$T_s{}^* = \sqrt{\frac{P.c_u}{c_s}}$$

Cooper (1972)

Insights from Model

- Given: $$T_u{}^* = \sqrt{\frac{P.c_s}{c_u}}$$
- If $\boldsymbol{c_s}$ goes up, then $\boldsymbol{t_u}$ goes up, while $\boldsymbol{t_s}$ goes down.
  – The user needs to invest more in issuing a good query
- If ***P*** goes up, then $\boldsymbol{t_u}$ and $\boldsymbol{t_s}$ goes up.
  – i.e. to get more relevant documents you need to search more.

Cooper (1972)

All models are wrong

but some are useful

George E.P. Box

Summary

- **Formal models** provide a way to **think** and **reason** about **interaction**
- Users make many choices when interacting with a system
- Designers make many choices when designing a system
  - Such **choices** often involve **trade-offs**
- Using **Economic** / **Optimization Models** focuses our attention on **salient variables** to **draw insights** about the interaction.

MORE EXAMPLE

Some search related models

Interactions & Decisions

- **Explain** why user behavior
- **Compare** different sequences of interactions
- **Reason** when certain functionality will be better than other functionality
- **Determine** how valuable the functionality needs to be for it to be used

Interactions & Decisions

Python

Search

Resuts for: Python

Showing page 1

Python - Official Site
www.python.org
Official site for the interpreted, interactive, object-oriented, extensible, hydromatic, systematic programming language
[ ] Mark - Find Similar - Find more like these

Python (programming language) - Wikipedia
en.wikipedia.org/Python
Python is a widely used high-level, general-purpose, interpreted, dynamic programming language. Its design philosophy emphasises..
[ ] Mark - Find Similar - Find more like these

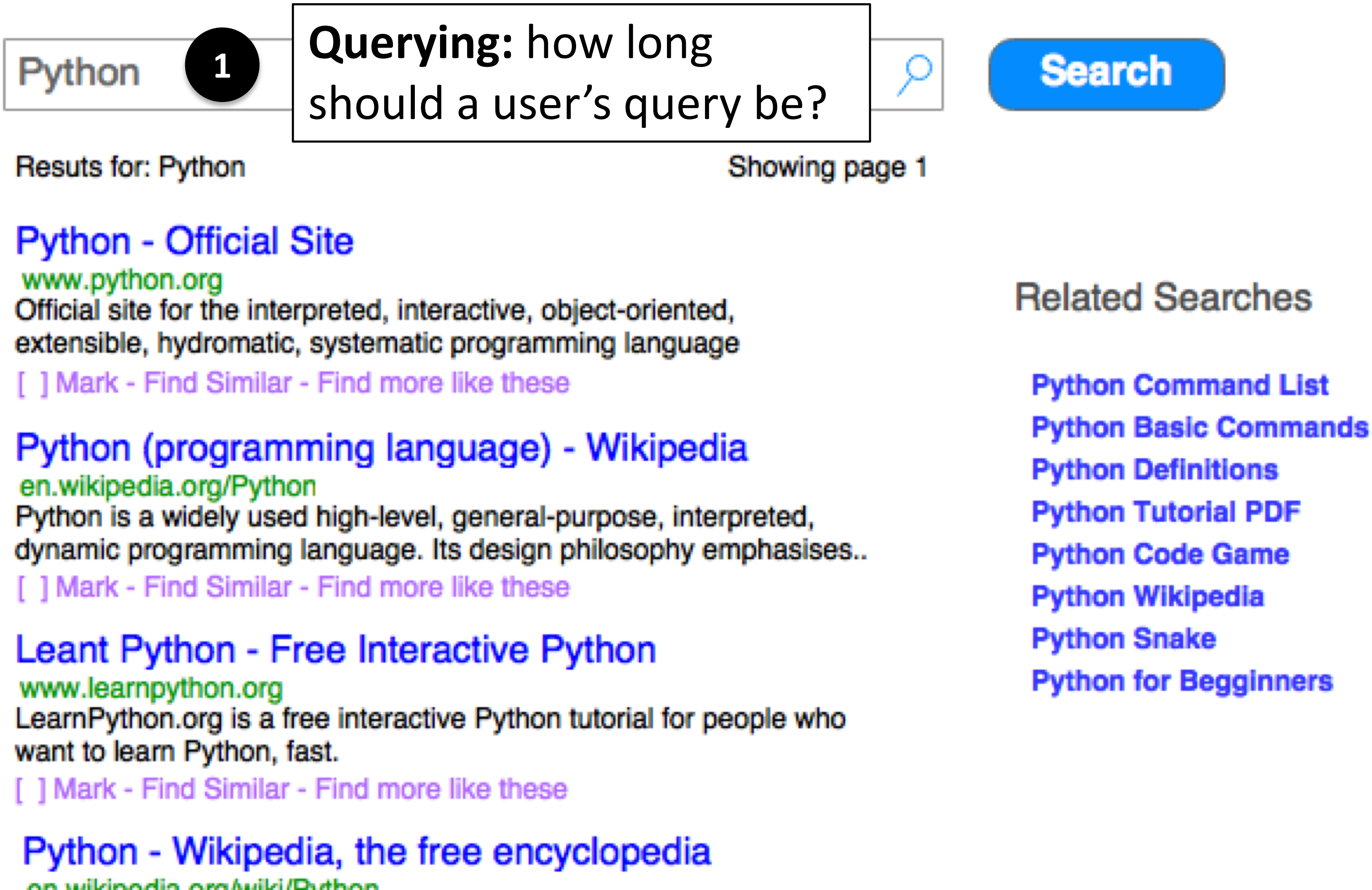
Leant Python - Free Interactive Python
www.learnpython.org
LearnPython.org is a free interactive Python tutorial for people who want to learn Python, fast.
[ ] Mark - Find Similar - Find more like these

Python - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/Python

Related Searches

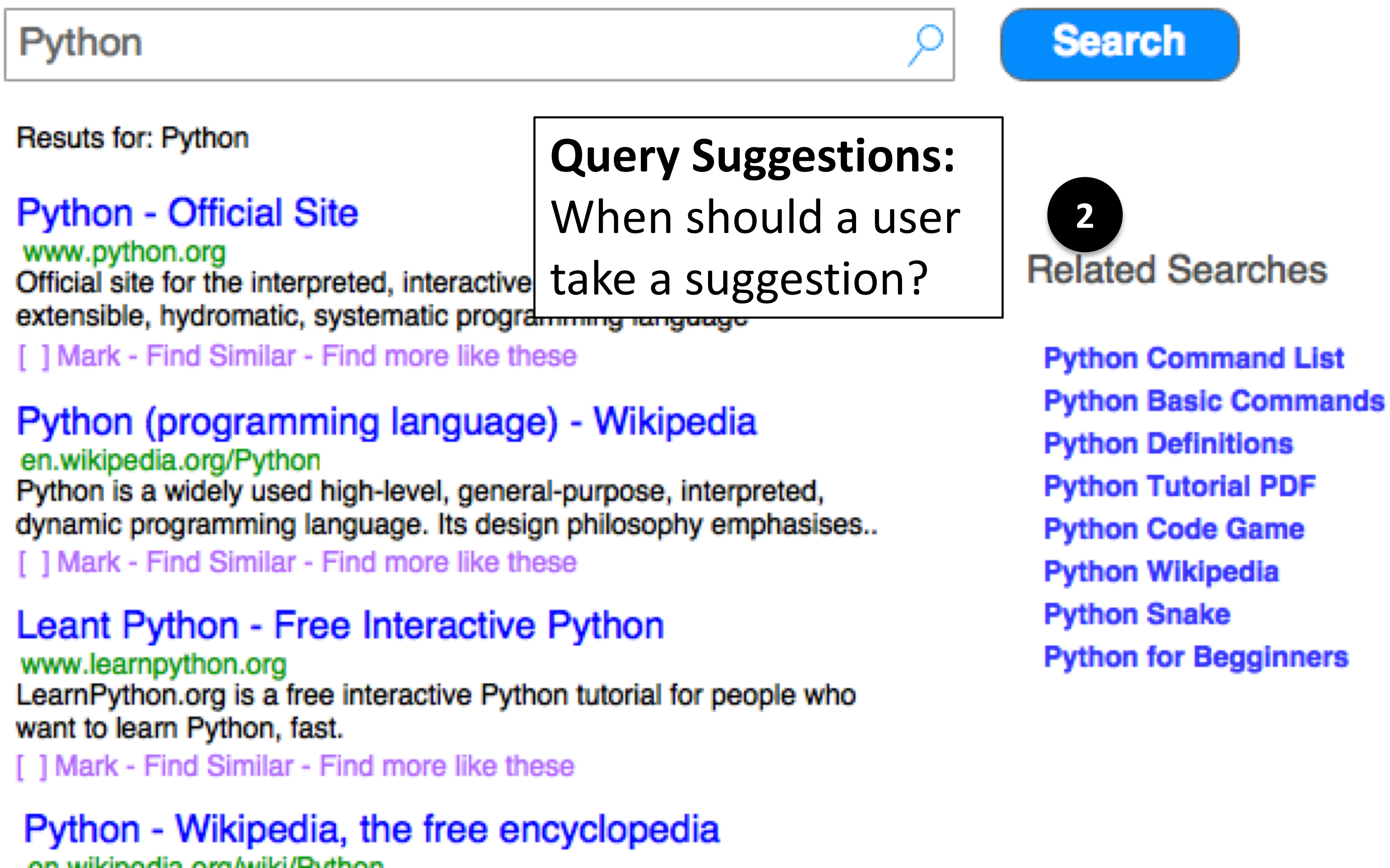
- Python Command List
- Python Basic Commands
- Python Definitions
- Python Tutorial PDF
- Python Code Game
- Python Wikipedia
- Python Snake
- Python for Begginners

Interactions & Decisions

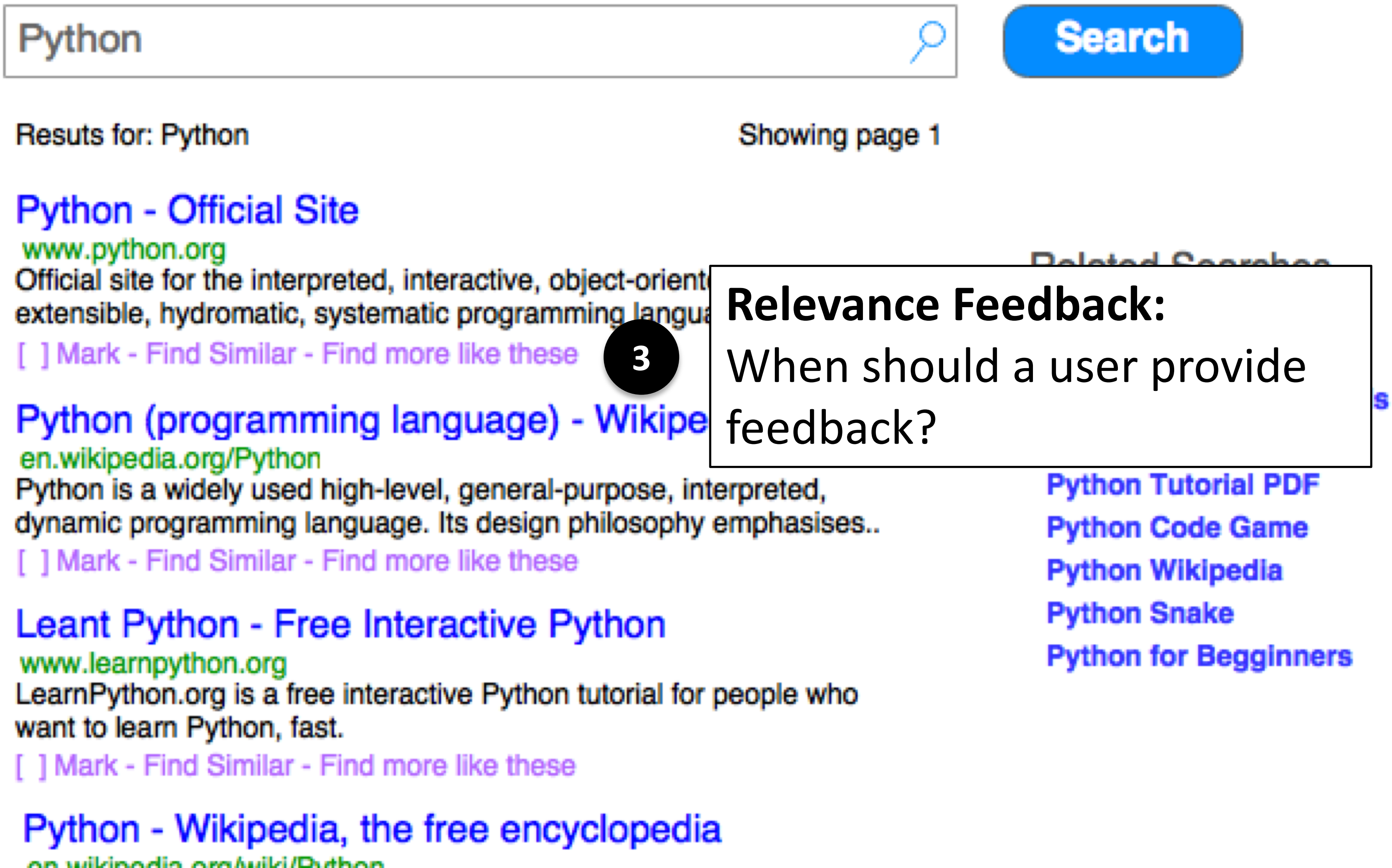


Interactions & Decisions

Query Suggestions: When should a user take a suggestion?



Interactions & Decisions



HOW LONG SHOULD A QUERY BE?

Or why do users pose short queries?

A Model of Query Length

- Users tend to pose short queries
- **But longer queries perform better**
- Many attempts to illicit longer queries
  - Instructing users
  - Longer query boxes
  - Glow boxes
- **Inline query autocomplete** and **voice queries** have meant that queries are getting longer.
- But, why?

A Model of Query Length

- What is the relationship between the benefit of a query and the length of a query?
  - ***W*** is the number of words in the query
  - The benefit ***b(W)*** from a query with ***W*** words:

$$b(W) = k.\log_a(W + 1)$$

  - ***α*** represents how quickly benefits drops off
  - ***k*** is a scaling factor

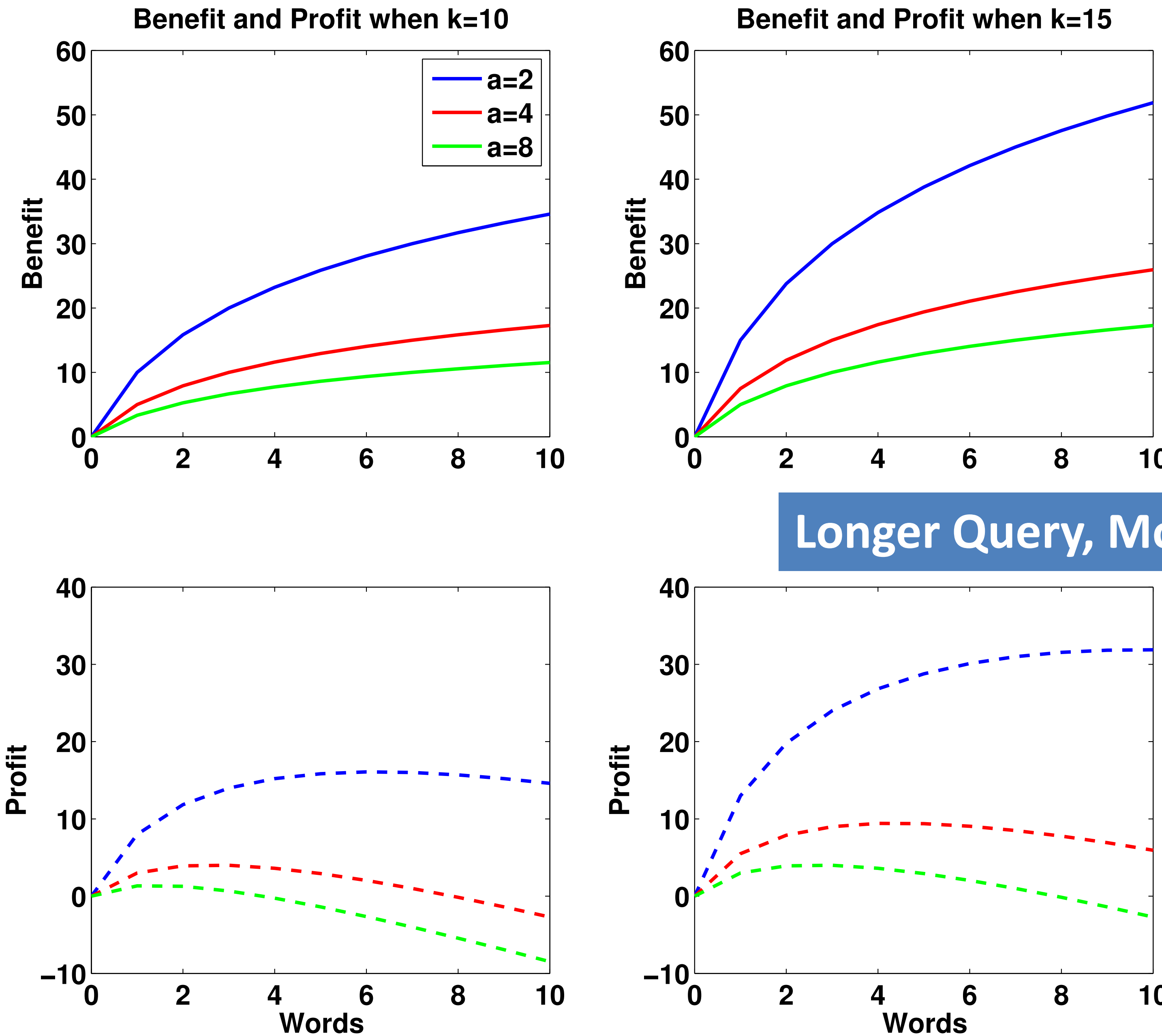
A Model of Query Length

- The cost of entering a query with ***W*** words

$$c(w) = W.c_w$$

- The profit function

$$\begin{aligned} \pi &= b(W) - c(W) \\ &= k.\log_a(W+1) - W.c_w \end{aligned}$$



Longer Query, More Benefit

Trade-off between Length and Profit

Optimal Query Length

- By differentiating the profit function with respect to ***W***, and solve we arrive at:

$$W^\star = \frac{k}{c_w . \log a} - 1$$

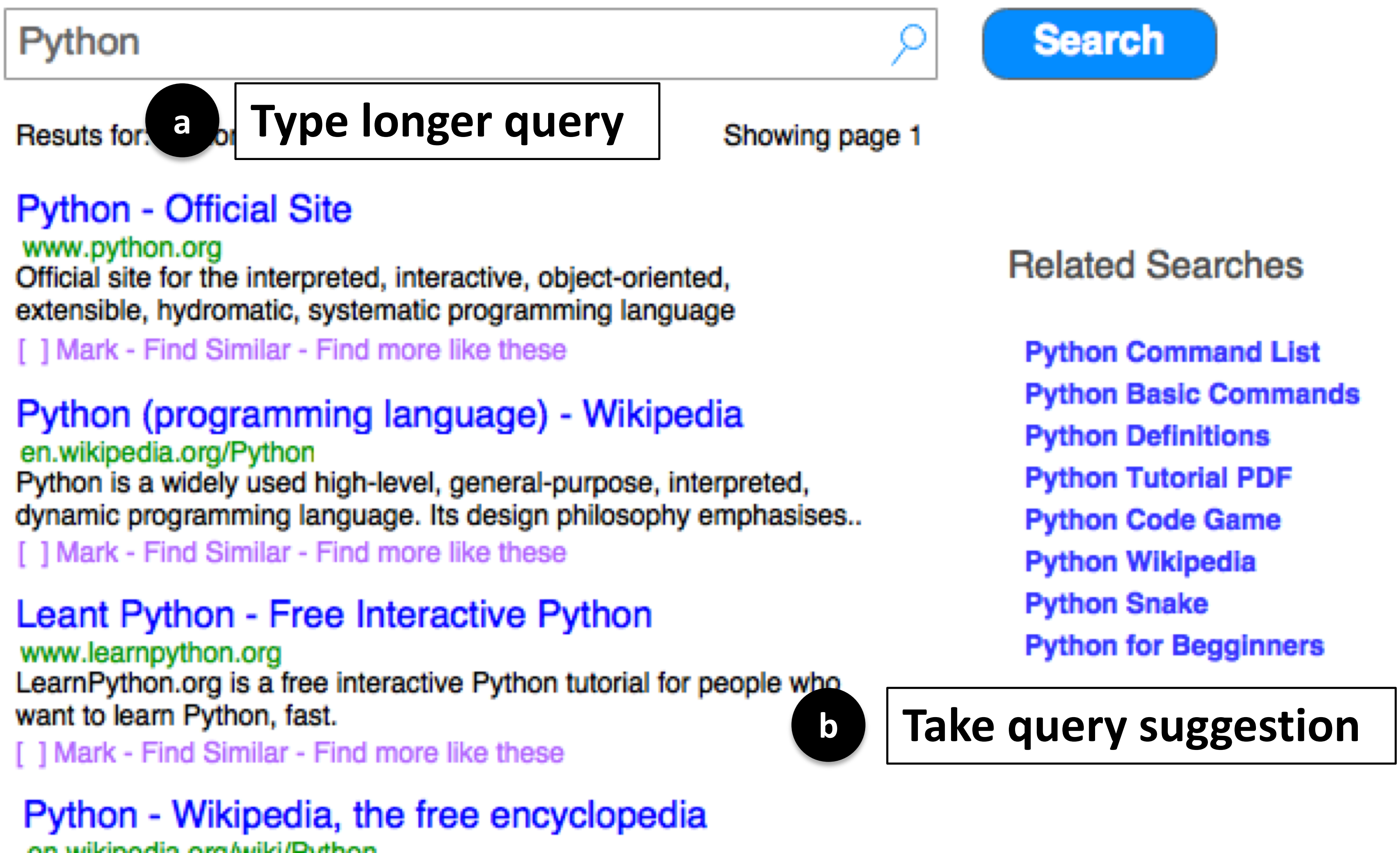
- ***W**** increases, i.e. queries get longer, when:
  - C_w decreases
  - ***α*** increases i.e. diminishing returns kicks in later
  - ***k*** increases

REFORMULATE OR TAKE THE SUGGESTION?

A Model of Querying Choices

- A user enters a query into the system and the system doesn't retrieve any relevant documents ☹
- Let's assume that this is because the query is **underspecified** or **impoverished** in some way
- Choices
  - (a) reformulate and make it more specific, or
  - (b) take a query suggestion

A Model of Querying Choices



A Model of Querying Choices

- ***C_{q2}* -** cost of modifying the original query
  - cost of choice (a)
- ***C_{es}(.)* -** cost of examining the suggestions, which is proportional to the number of suggestions ***Q_s***
- ***P_s* -** probability that a suggestion exists that the user takes, otherwise they need to reformulate, anyway.
- ***C_c* -** cost of taking the suggestion, cost of choice (b):

$$c_{es}(Q_s) + p_s.c_c + (1 - p_s).c_{q2}$$

Reformulate or Take Suggestion?

- If the LHS is less than the RHS, then it is better to reformulate,
- Else it is better to examine the suggestions, first.

$$
\begin{aligned}
c_{q2} &< c_{es}(Q_s) + p_s.c_c + (1 - p_s).c_{q2} \\
p_s.c_{q2} - p_s.c_c &< c_{es}(Q_s) \\
c_{q2} - c_c &< \frac{c_{es}(Q_s)}{p_s}
\end{aligned}
$$

- We can see that $\mathbf{p_s}$ has a big impact, and magnifies the cost of examining suggestions.
- An obvious trade-off is b/w $\mathbf{c_{es}}$ and $\mathbf{p_s}$:
  – more suggestions, greater $\mathbf{p_s}$, but higher $\mathbf{c_{es}}$

GIVE RELEVANCE FEEDBACK?

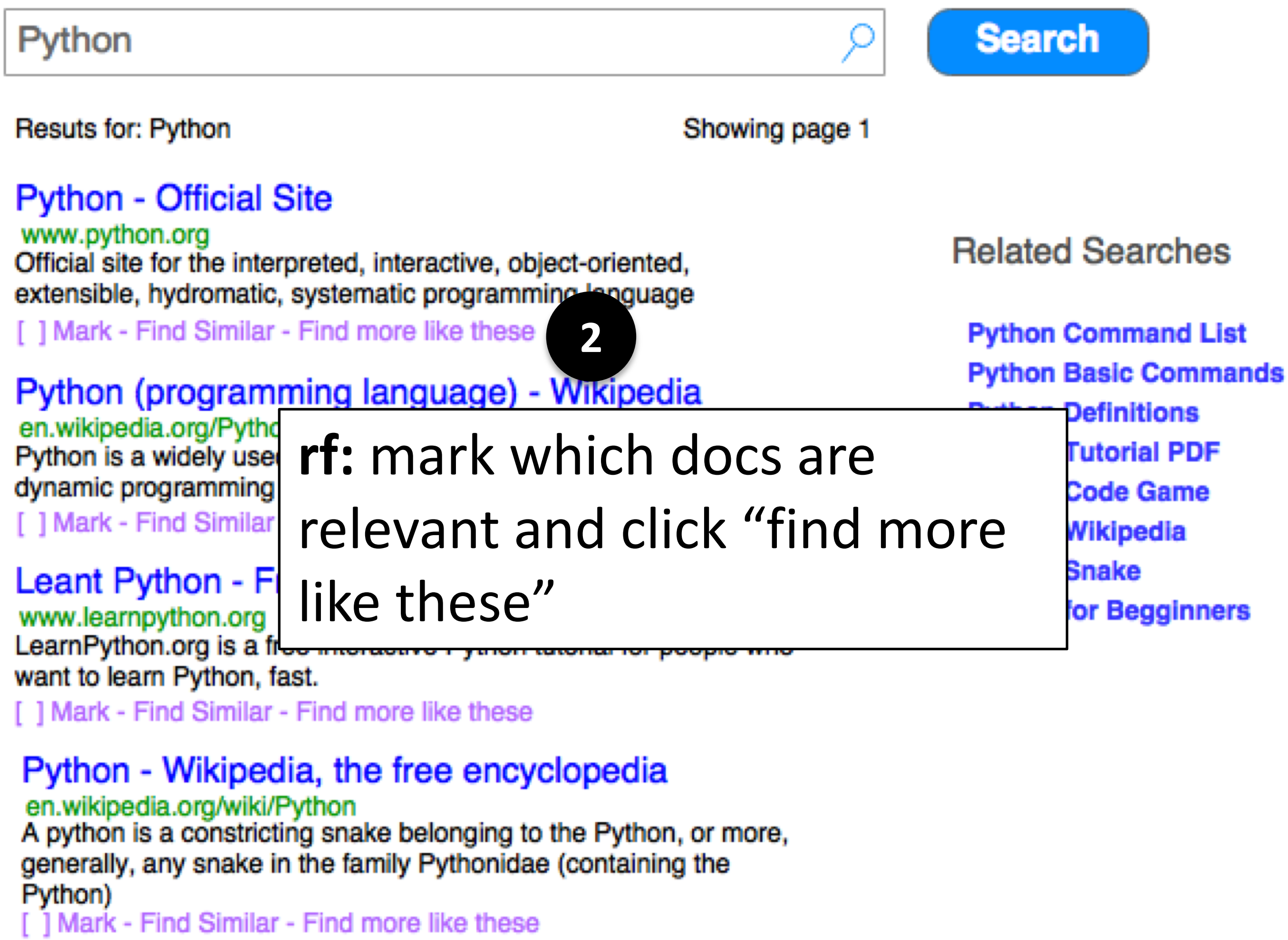
Issue a new query

A Cost Model of Issuing a New Query

$$c_{nq} = c_q + N.c_a + c_q + N.c_a$$

- **c_q**: cost of issuing a query
- **c_a**: avg cost of examining a document

Relevance Feedback



rf: mark which docs are relevant and click “find more like these”

A Cost Model of Relevance Feedback

$$c_{rf} = c_q + N.c_a + c_d + M.c_m + c_c + N.c_a$$

- $\mathbf{c_q}$: cost of issuing a query
- $\mathbf{c_a}$: avg cost of examining a document
- $\mathbf{c_c}$: cost of click on find more like this button
- $\mathbf{c_d}$: cost of deciding to mark and do RF
- $\mathbf{c_m}$: cost of marking a document as relevant; **M** number of marked docs

When to issue a new query?

- Let's assume for now that benefits from each choice are equivalent
- Opt to issue a new query when:

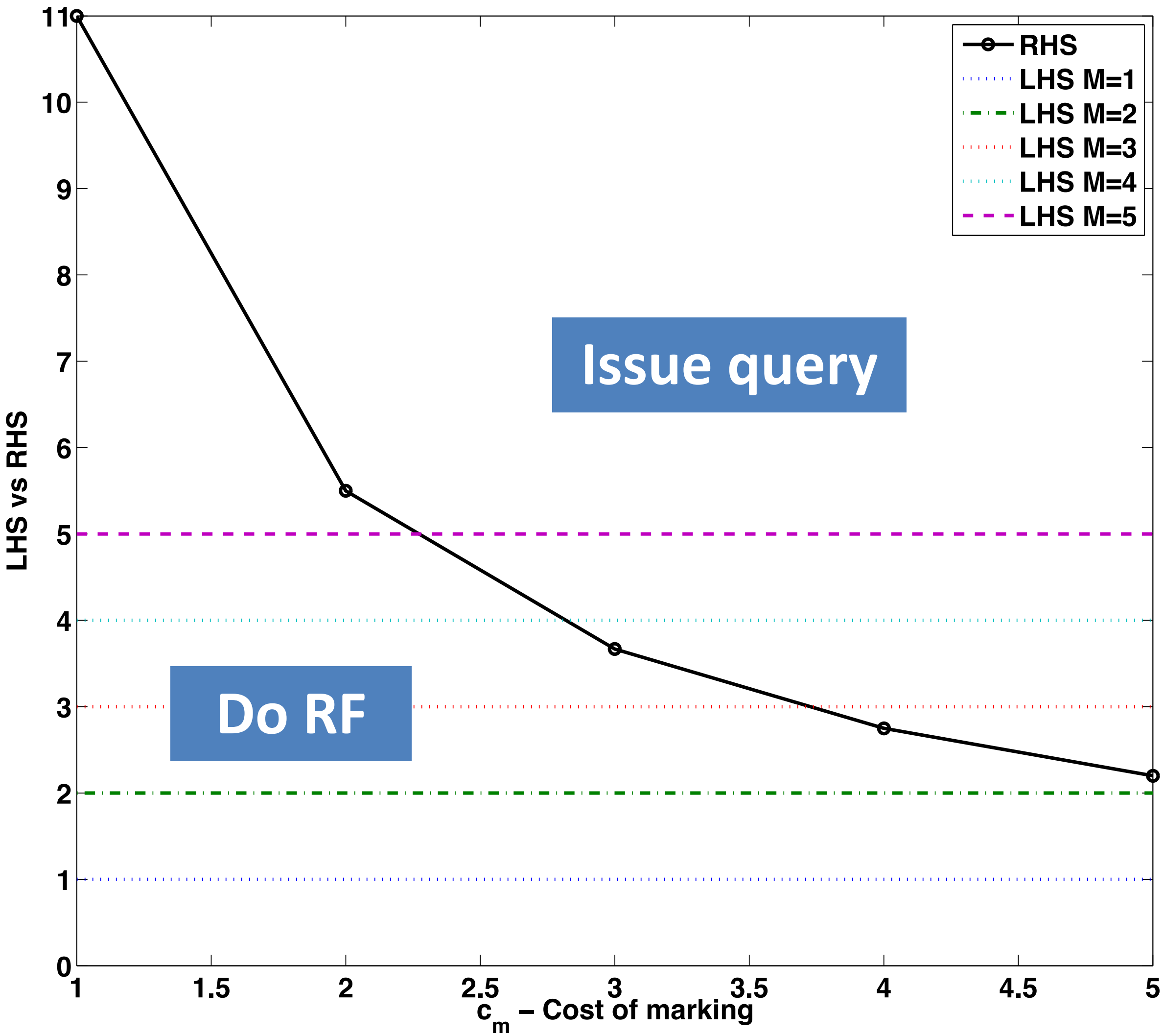
$$
\begin{aligned}
c_{nq} &< c_{rf} \\
2.c_q + 2.N.c_a &< c_q + 2.N.c_a + c_d + M.c_m + c_c \\
c_q &< c_d + M.c_m + c_c
\end{aligned}
$$

Marking documents for RF

- We can derive the relationship with respect to **M**: the number of documents marked for RF

$$M > \frac{c_q - c_d - c_c}{c_m}$$

When to RF and when to query?

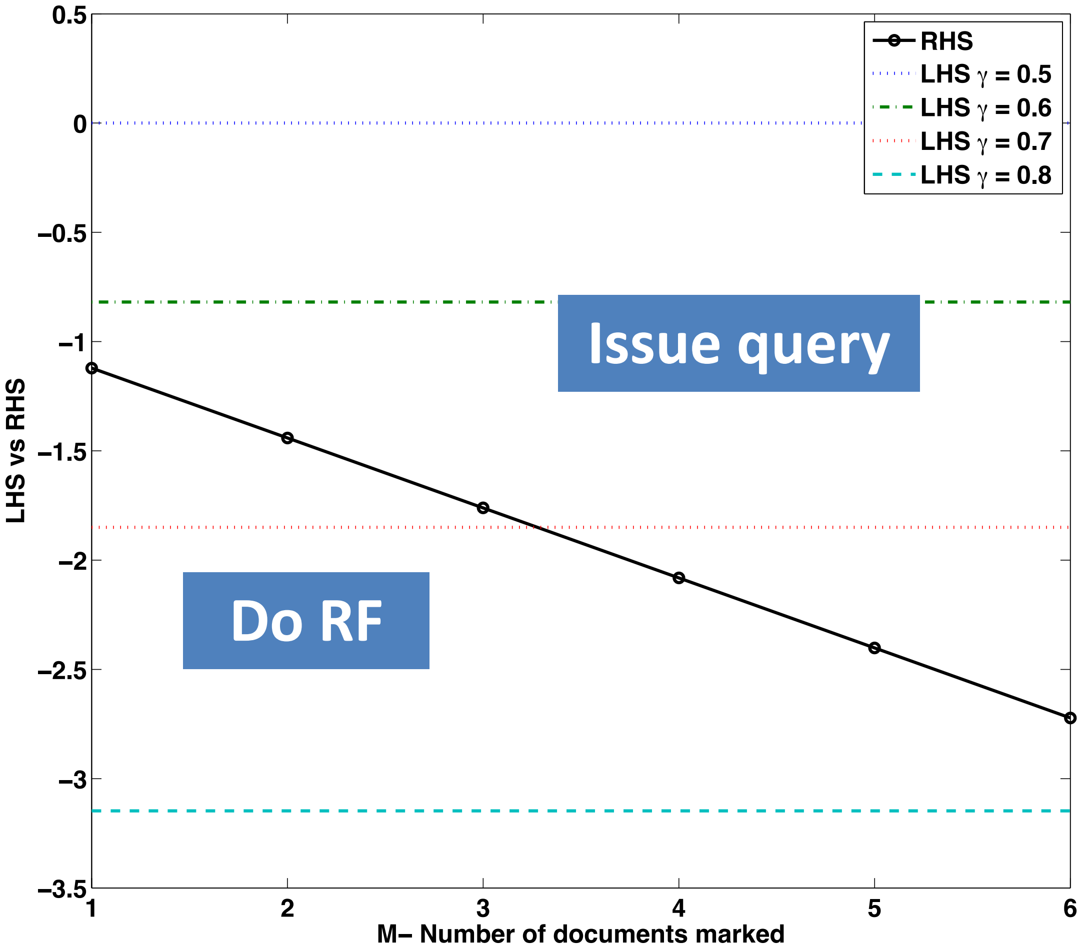


$$M > \frac{c_q - c_d - c_c}{c_m}$$

- If the cost of marking is low, then it is better to do RF
- If M has to be high to have the same benefit as querying, then querying is more desirable

Different benefit functions

$$N^{\beta} - N^{\gamma} \quad > \quad \frac{c_q - c_d - M.c_m - c_c}{k.d(q)}$$



- **ϒ**: performance of RF, **β** =0.5 for Q
- Issue query if LHS > RHS
- If benefit(Q) > benefit(RF), then LHS is positive
- Benefit of RF needs to be substantially greater than benefit of next query for RF to be useful i.e. **ϒ >> β**

SUMMARY

Summary

- We have explored how we can create a variety of models based on **costs** and **benefits**.
- We've created some **really simple abstracted models**
  - They are the starting point for **more complex & realistic models**
- Each model, however, **highlights salient costs** & **benefits** that are likely to effect the choices users make
  - They make **predictions** about **user behavior**
  - And **suggest** what we need to **improve** in our **system** for an **option** to be **profitable**

Summary

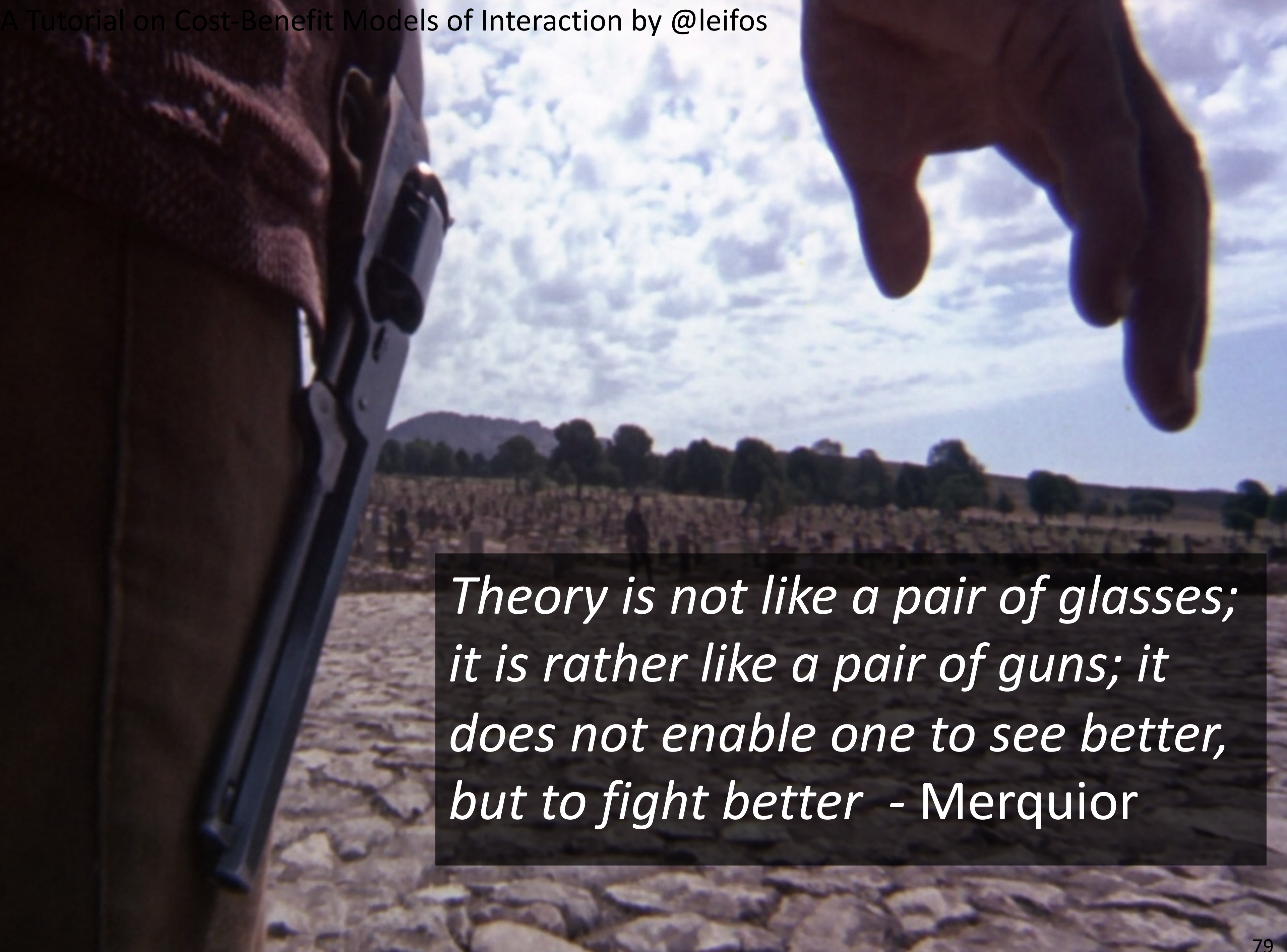
- Before any **experimentation**, such models provide a **formal guide on how to proceed**

- **The models** provide **hypothesizes** about **behavior**

- These can be used to inform design and be tested in practice

Challenges

- How do we measure the **costs and benefits**?
- How do we measure the **uncertainty**?
- And, how do we construct experiments that enable us to test the hypothesizes generated from the models?



Theory is not like a pair of glasses; it is rather like a pair of guns; it does not enable one to see better, but to fight better - Merquior

END OF SESSION TWO