



Module #2

Needs-based recommendation systems and Next Best Actions

Raffaele Zenti (raffaele.zenti@virtualb.it)

Co-Founder, Head of Financial Data Science, Virtual B SpA



The key to financial personalization



Needs/goals are key to targeting clients

- After all, we are still monkeys: most of us don't really understand financial and insurance products (so we don't buy...)
- But we all have real needs – and we understand them, eg:
 - protect ourself, our family, our things → insurance
 - buy things → payment tools & services
 - save for future consumption → savings & investments
 - anticipate future consumption or investments → borrowing

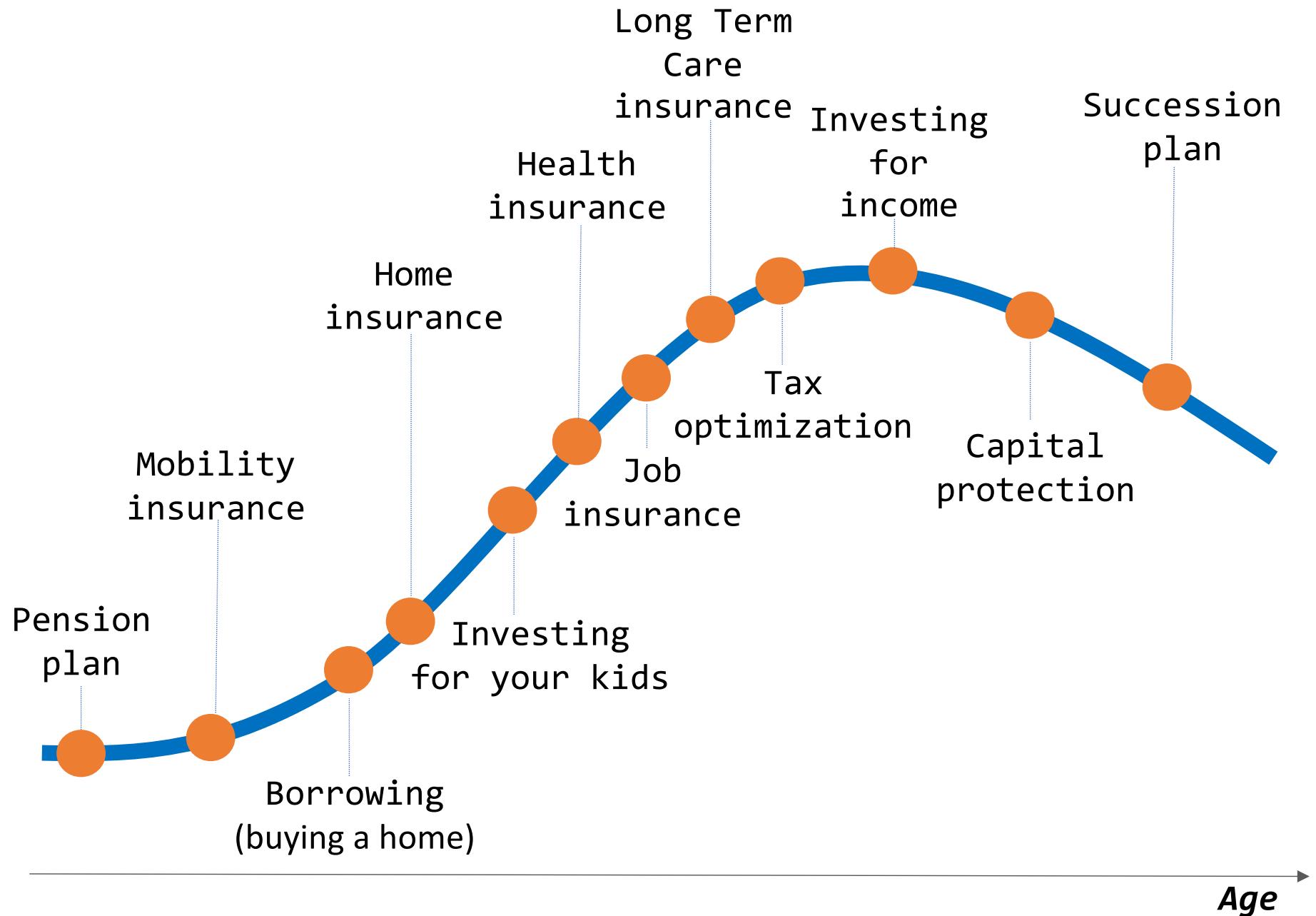
We tend buy what we need: thus, needs are a good starting point for recommending financial products and services in a personalized way

Another good reason to estimate client's needs: the law

- MIFID/IDD: coherence between needs/goals and financial/insurance products sold to clients
- Key information about clients must be collected by the use of a MIFID/IDD questionnaire → **Data**
- **Data** → a lot of detailed information collected through MIFID/IDD questionnaires can be crunched by algorithms
- Basically you get a broad survey for free – that's the reason why MIFID/IDD questionnaires should be properly prepared

**“Make a virtue
out of necessity”**

Financial needs: “the theory”



The reality of financial & insurance needs

- Not everybody will start a family at 30...
- Maybe at 75 not everybody is willing to plan her inheritance process
- What about if at 50 you have 2 divorces and 2 maintenance allowances?
- Maybe at 35 someone faces a big recession, is fired, and cannot buy a home
- ...

**Financial needs change over time following our random life
And our random lifes are not all equal**

Financial needs change overtime

How? Ask data!

$$\text{Need}(i, t) = f(\text{client situation}(t), \text{context}(t))$$

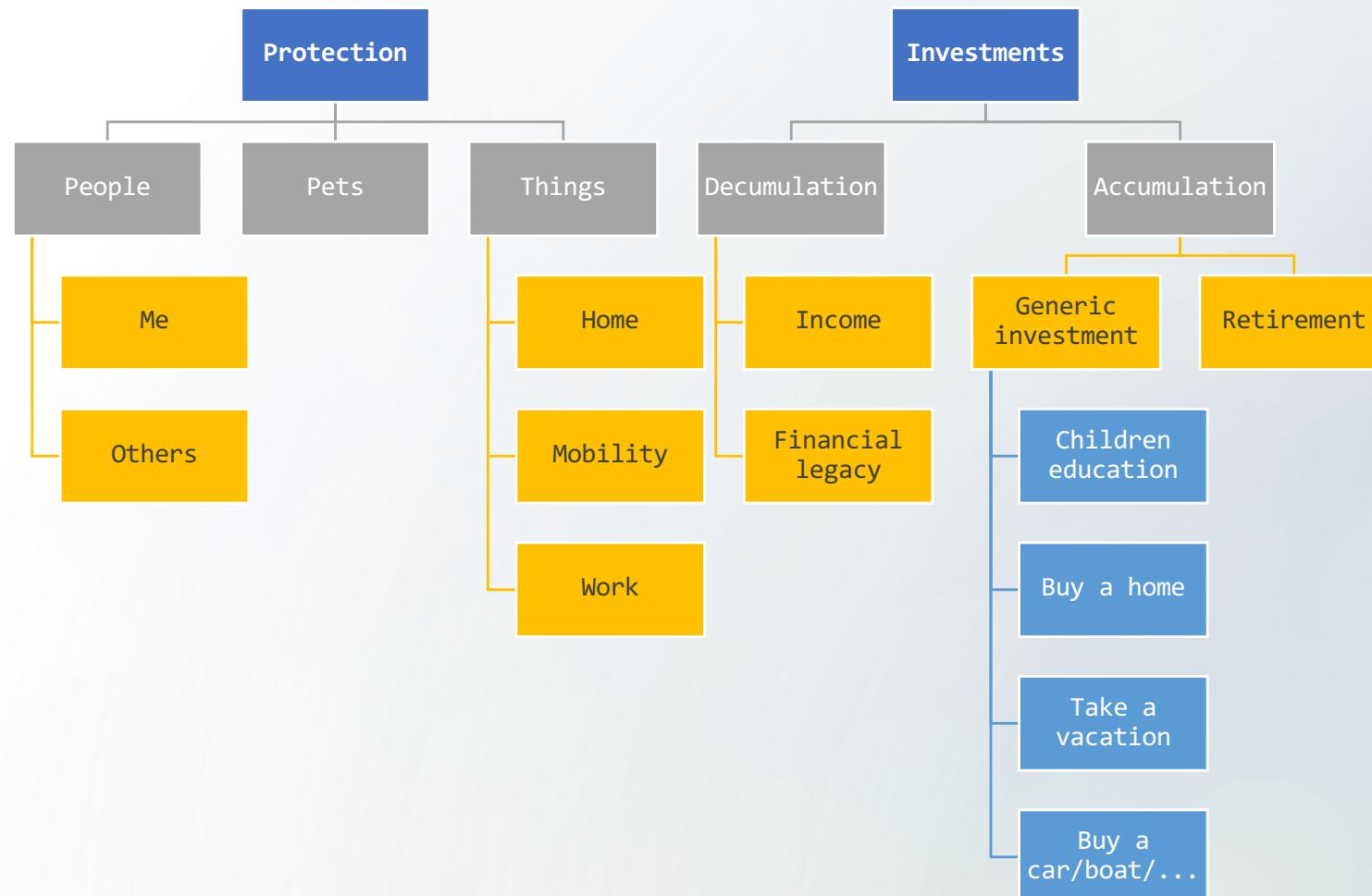
That is (...smell of ML):

$$Y = f(x_1, x_2, x_3, \dots)$$

Y = responses = Need

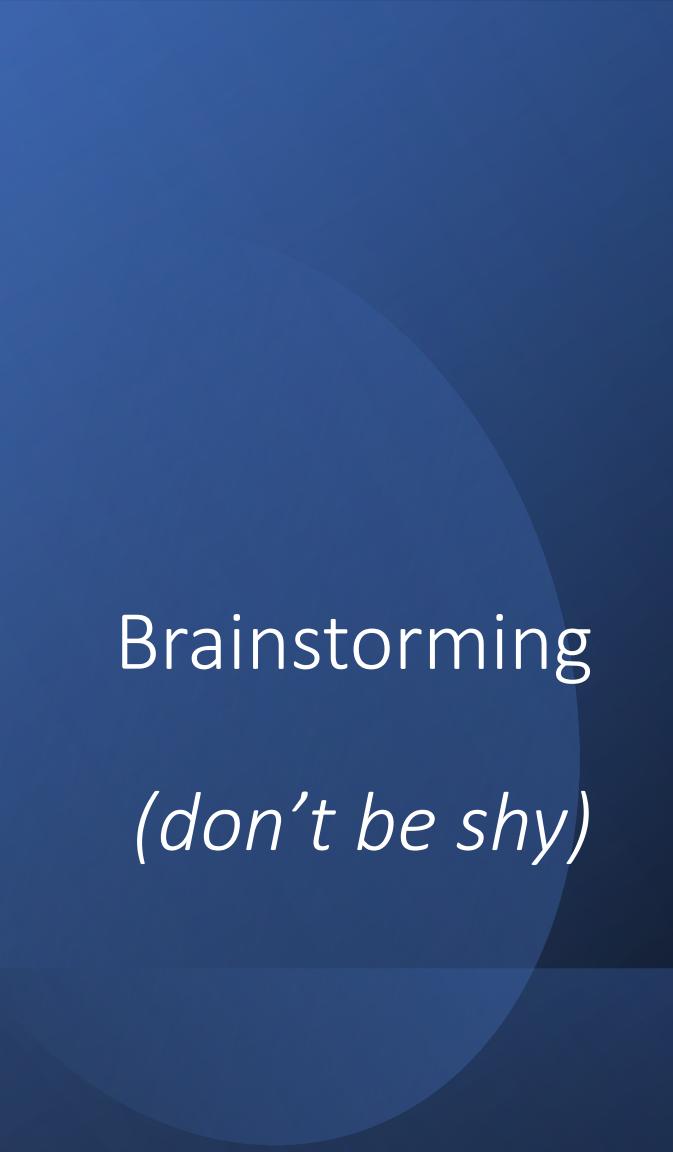
$x_1, x_2, x_3, \dots = X$ = features = client situation, context

Needs & goals, an example for an insurance company with a wealth management business



Let's look at
the data





Brainstorming *(don't be shy)*

- What would you do?
- What are the key steps to start from the data and arrive at customer needs or goals?
- And then, how to recommend product/services based on needs and goals
- What kind of algorithms would you use? Why?
- ...

Estimating needs: a Supervised ML problem

- A client might have/not have a given financial need (or goal)
- Each client might have more needs
- Needs can be satisfied by financial products

I could teach a ML algorithm
to recognize presence/absence
of needs

It's a typical classification
problem

And what
about the
Ys?
Welcome to
the real
world!

- In a classification problem we teach an algorithm to put labels
- But... Where are the labels?
- Who is able to say «Client A has need Z»?
- **Needs are not observable!**
- Hence, we have our Xs but we are not sure about the Ys → We have a problem...



«I'm Mr Wolf, I Solve Problems»

- Basically, two ways:
 - Straight - A human (an expert) puts the explicit labels
 - «*Client A has/has not need Z*»
- Less straight - Use implicit labels:
 - we try to learn from those who should know if a given client has a given need/goal
 - these guys are financial advisors, insurance brokers, private bankers – aka «relationship managers»

Case 1: explicit labels = = a human being creates the Ys

- She puts:
 - $Y = 1$ if client(i) has need(j)
 - $Y = 0$ otherwise
- Quite common in image recognition
- If the human-labeler is reliable → very good
- But: financial needs are not easy to spot – they are not cats/dogs/pedestrians/etc
- So: you need investment/insurance/banking experts → expensive

Case 2: Implicit labels = = Y_s inferred from expert behaviour

- If an advisor sells a financial product that satisfies a given need/goal, then, probably it was in order to satisfy that need
- Thus:
 - $Y = 1$ if client(i) owns a product that satisfies need(j)
 - $Y = 0$ otherwise
- If the human is reliable \rightarrow very good
- If the human sells products that maximize HER OWN profits \rightarrow the algorithm will learn exactly that process (do you remember all these fancy talks about AI and ethics?)
- Solutions:
 - filtering experts and their behaviors, doing «expert-picking»
 - using a priori information (Bayesian models)
 - combining a number of different models (Bayesian Model Averaging)

One-vs-All models

Vs

Multiclass models

- **One-vs-All (Binomial):**
- 1 need \leftrightarrow 1 model
- As many models as there are needs (10 needs \leftrightarrow 10 models, 30 \leftrightarrow needs, etc)
- **True multiclass (Multinomial):**
- N needs \leftrightarrow 1 model
- One single big model (10 needs \leftrightarrow 1 model, 100 needs \leftrightarrow 1 model, etc)
- → often more complex = might be less robust (see Occam's Razor)



From needs to recommendation (NBA): finding the best matching



- **Content-based filtering** - knowledge-based methods that rely mostly on the domain-knowledge
- **Case-based recommender systems** - apply case-based reasoning (CBR) that solves the recommendation problem based on (old) similar cases



Matlab session starts |



Take home on estimating needs and find the NBA

- Typical application of Supervised ML (but other methods can be applied...)
- Point of advantage = Business know-how during feature selection/engineering
- Several business applications:
 - Next Best Action (recommendations)
 - Product targeting of needs' estimation
 - Cross-selling/Up-selling
 - Precision marketing online/offline
 - ...

Now YOU

- **It's your turn: use your favorite techniques to estimate customers' needs and recommend products, write the code (use my code, or start from scratch, or whatever...as you like), and we'll talk about it next time**



Next time

- Each group will present ideas, results, doubts, code snippet, etc
- Be short and concise (you are 20 groups):
 - 3' each group
 - Skip the details, go straight to the key methods and findings
 - Use selected plots, charts, tables (commented code snippets are good if concise)
 - If you have doubts and the minutes of the presentation are not enough, no problem, you can write to me (several of you have done so)
- Get ready to share your screen and your work - be it little or a lot – don't be shy
- My email address: raffaele.zenti@virtualb.it