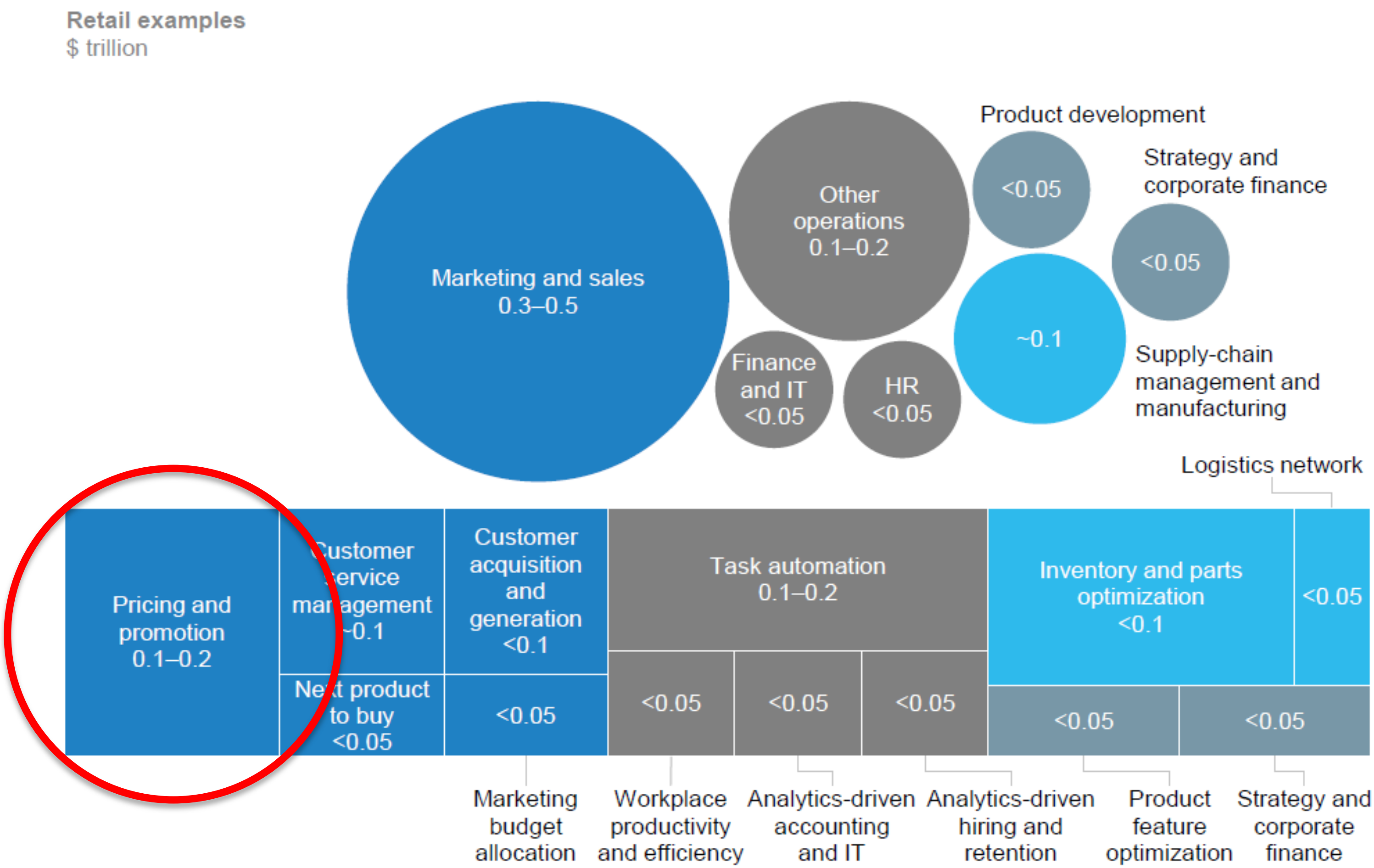
Pricing the Long Tail by Explainable Product Aggregation and Monotonic Bandits

Marco Mussi[1], Gianmarco Genalti[1], Francesco Trovó[1]
Alessandro Nuara[2], Nicola Gatti[1], and Marcello Restelli[1]

[1] Politecnico di Milano [2] ML cube

28th Conference on Knowledge Discovery and Data Mining, Washington D.C.

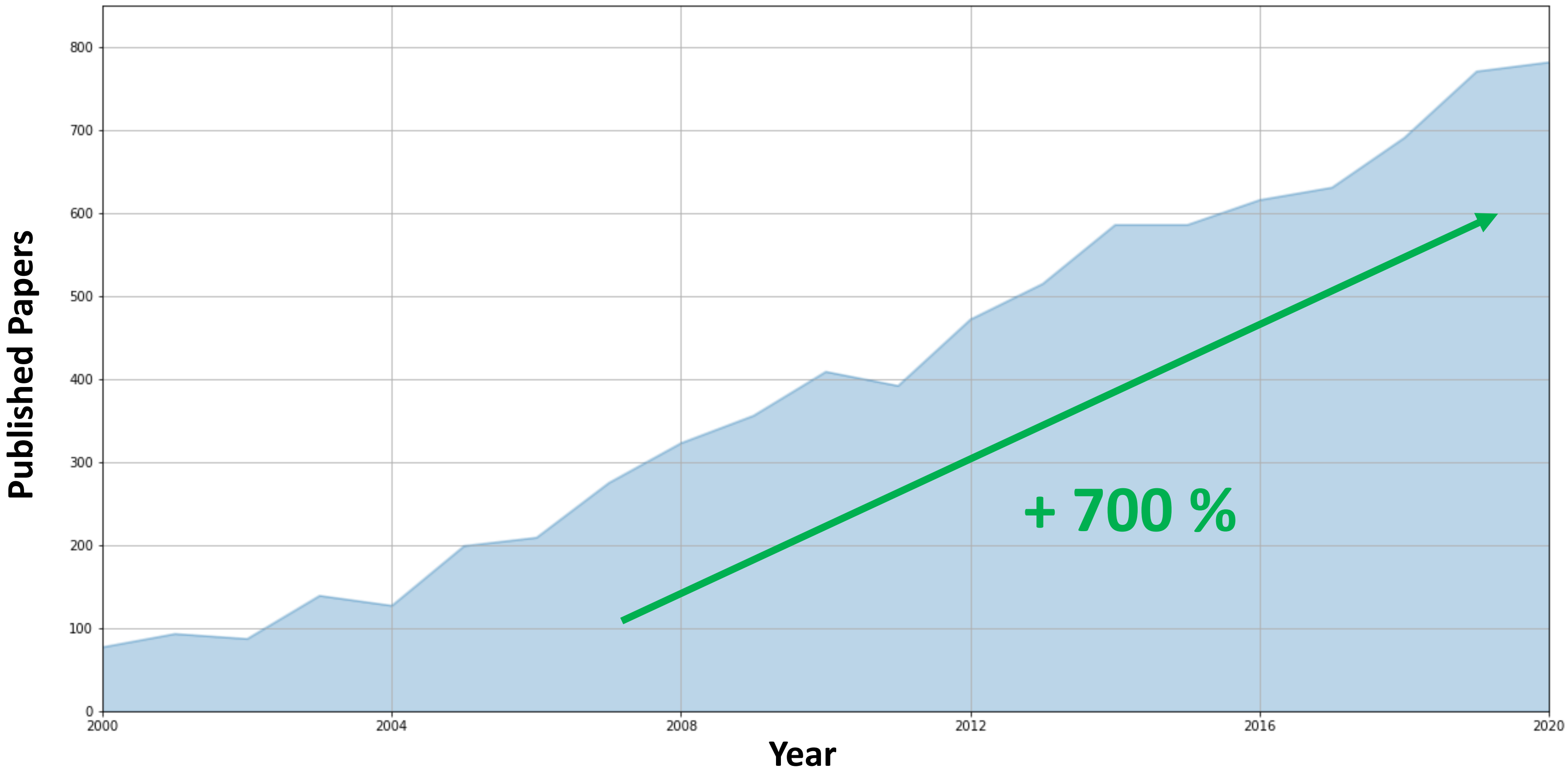
AI in Retail Business



[1] Chui, Michael, et al. "Notes from the AI frontier: Insights from hundreds of use cases." *McKinsey Global Institute* (2018).

Research on Dynamic Pricing

Scientific Production on Dynamic Pricing[2]

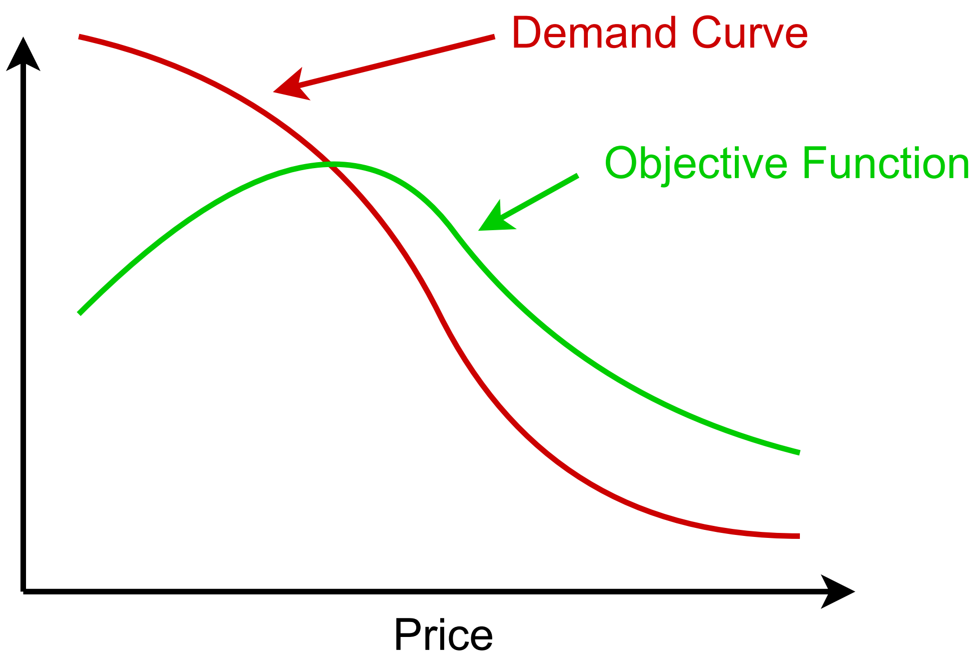


[2] www.scopus.com

Dynamic Pricing

Consider an e-commerce which sells a product

- Customers **visit** the product page and **decide** whether to buy or not
- By aggregating users choices, we are able to build a **demand curve**



[3] Arnoud V Den Boer. Dynamic pricing and learning: historical origins, current research, and new directions. Surveys in operations research and management science, 20(1):1–18, 201

Goal

Provide a dynamic pricing algorithm:

- **Common among all products**
- **Online**
- **Explainable**

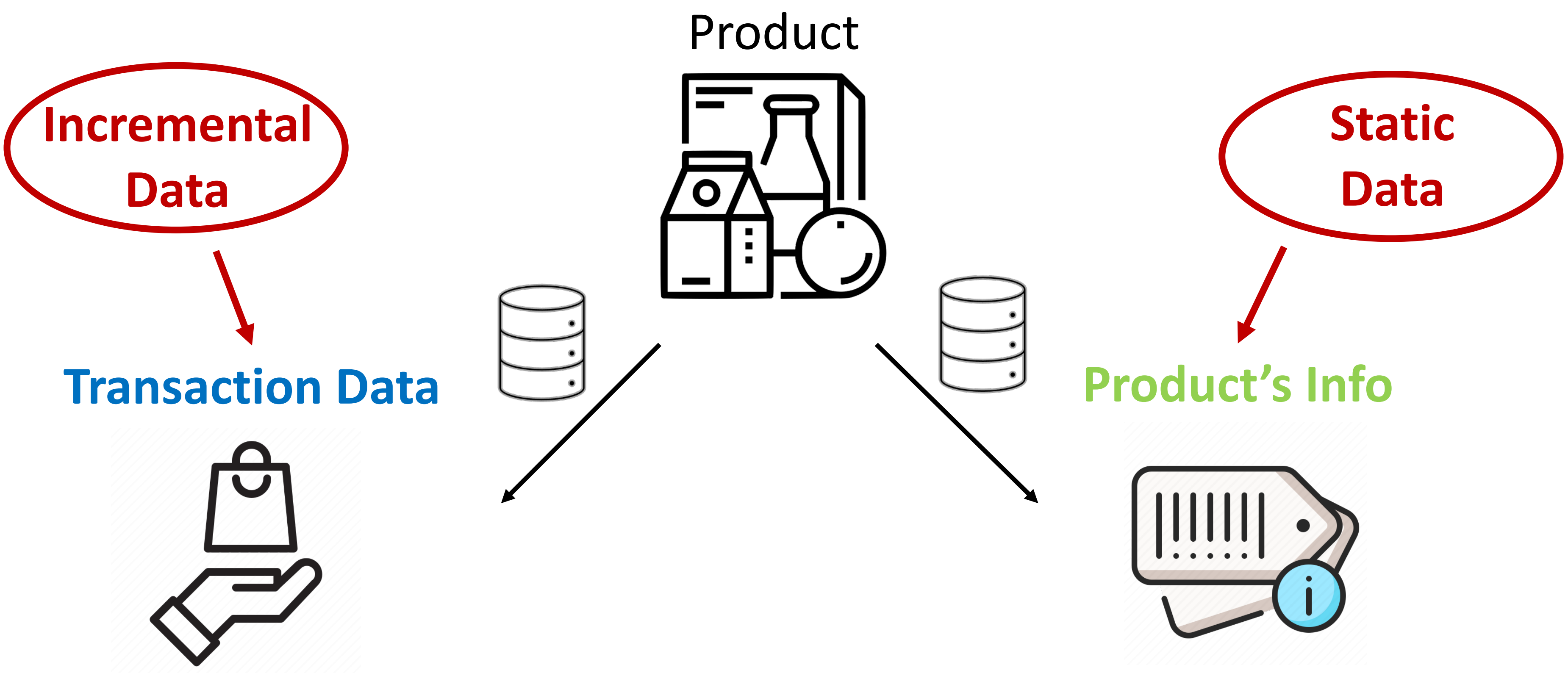
The Long Tail Paradigm

Real-World scenario:
≈ **20000** products in total
- ≈ **1000** best sellers
- ≈ **12000** long tail
- ≈ **7000** never sold before

[4] Chris Anderson. The long tail: Why the future of business is selling less of more. Hachette Books, 2006

Setting and Goal

Available Data



- Weekly Sales
- Price History

- Textual description
- Supplier’s Cost

Problem Formulation

- The e-commerce sells N **non-perishable** products with **unlimited availability**

- At every time t, for every product j, we aim to set a **percentage margin** m_{jt} defined as

$$m_{jt} := \frac{p_{jt} - c_j}{c_j}$$

  - p_{jt} is the selling price of product j at time t
  - c_i is the acquisition cost for product j

Objective Function

Goal: Maximize *total profit*

$$f_{jt}(m_{jt}) \coloneqq m_{jt}\, c_j\, v_{jt}(m_{jt})$$

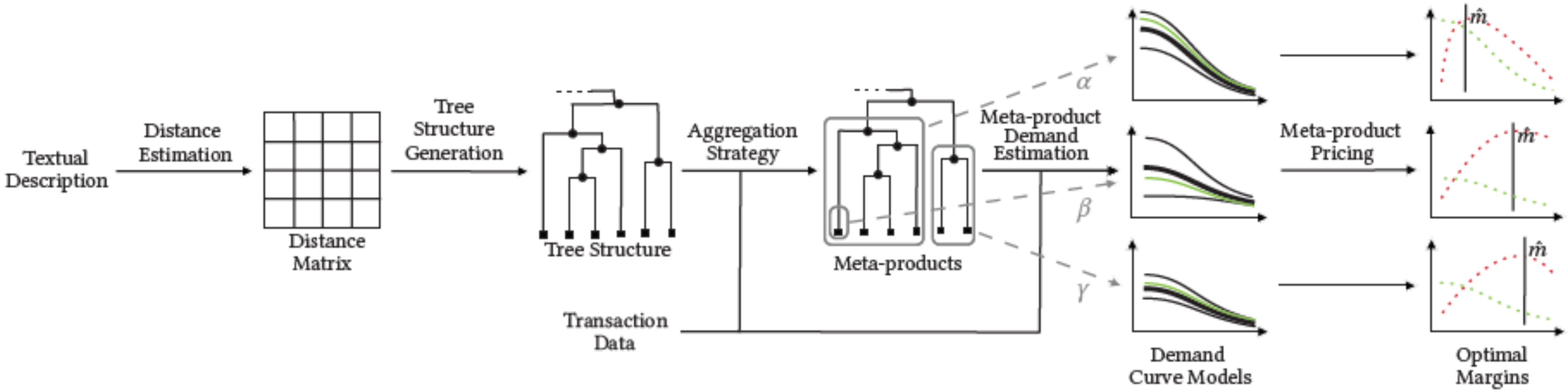
- $v_{jt}(m_{jt})$ is the number of units products j would sold at time t by setting the margin as m_{jt}

This implies find the **optimal pricing strategy**:

$$\boldsymbol{m}_t^* \coloneqq \underset{\{m_{1t}, m_{2t}, \dots\}}{\operatorname{argmax}} \sum_{j=1}^{N} f_{jt}(m_{jt})$$

Solution

Algorithm - Overview



Demand Curve Model

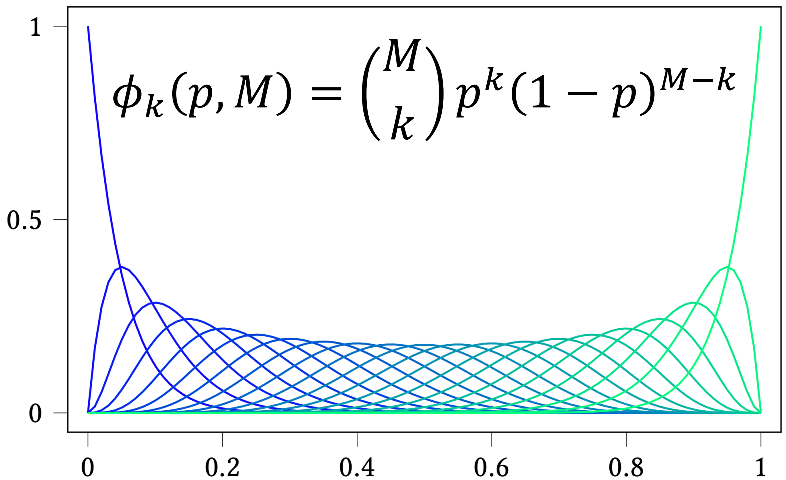
- To estimate the volume curve v_i we resort to *Bayesian Linear Regression* (BLR) with a basis function expansion:

$$\hat{v}_{jt}(m, \boldsymbol{w}_{jt}) = w_{0,\mathrm{jt}} + \sum_{i=1}^{M-1} w_{i,jt}\phi_i(m)$$

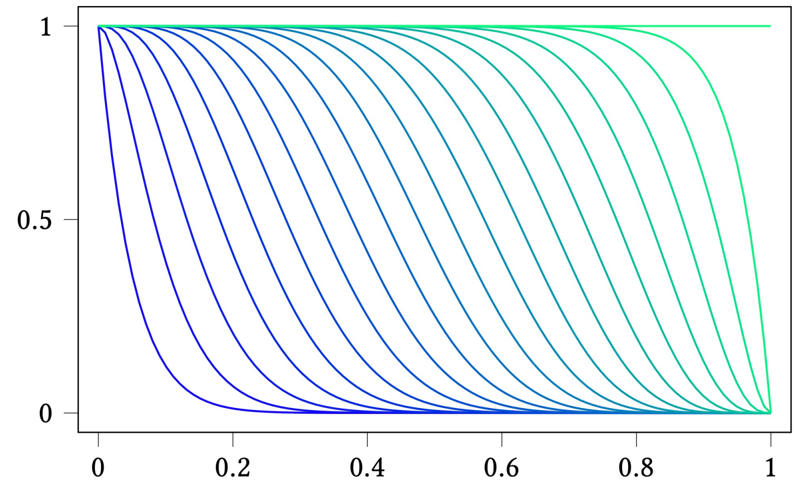
- We assume that the true volume curve is **non-increasing in price**. This assumption:
  - Is realistic in our setting since this goods are **non-luxury**
  - Alleviate **data scarcity** problems

Bernstein Polynomials

Price values are expanded using transformed **Bernstein Polynomial**



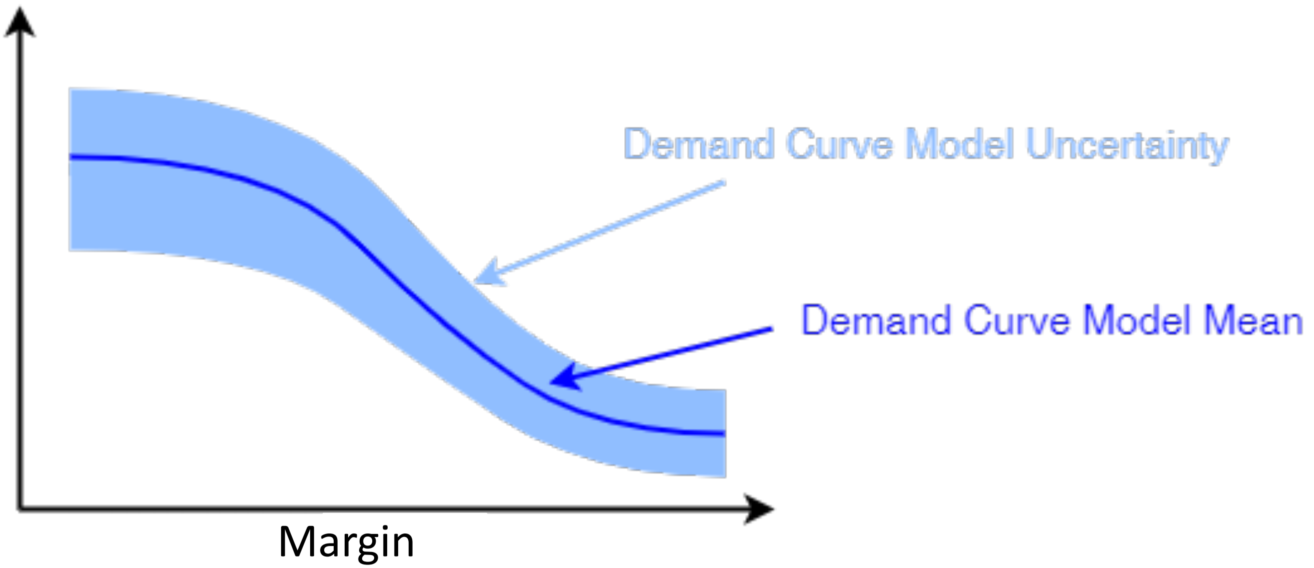
Bernstein Polynomials → Transformed Bernstein Polynomials Expansion



$\boldsymbol{w} \sim \mathcal{LN}(\mu, \sigma^2)$ We select a **positive support prior** distribution of BLR

Dealing with Price Exploration

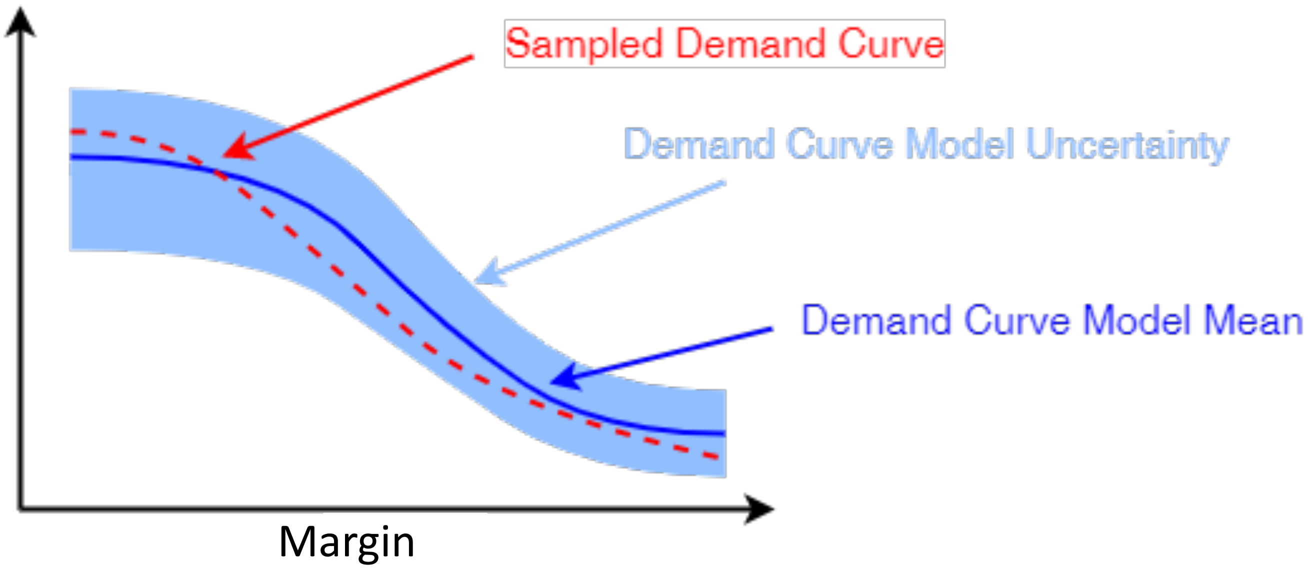
- BLR uncertainty over weights' posterior to sample[3] a single curve $\hat{v}_{jt}(\cdot, \tilde{\boldsymbol{w}}_{jt})$
- Objective function $\hat{v}_{jt}(\cdot, \tilde{\boldsymbol{w}}_{jt})$ is computed over the range of possible margins



[5] Thompson, William "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples." *Biometrika* (1933): 285-294.

Dealing with Price Exploration

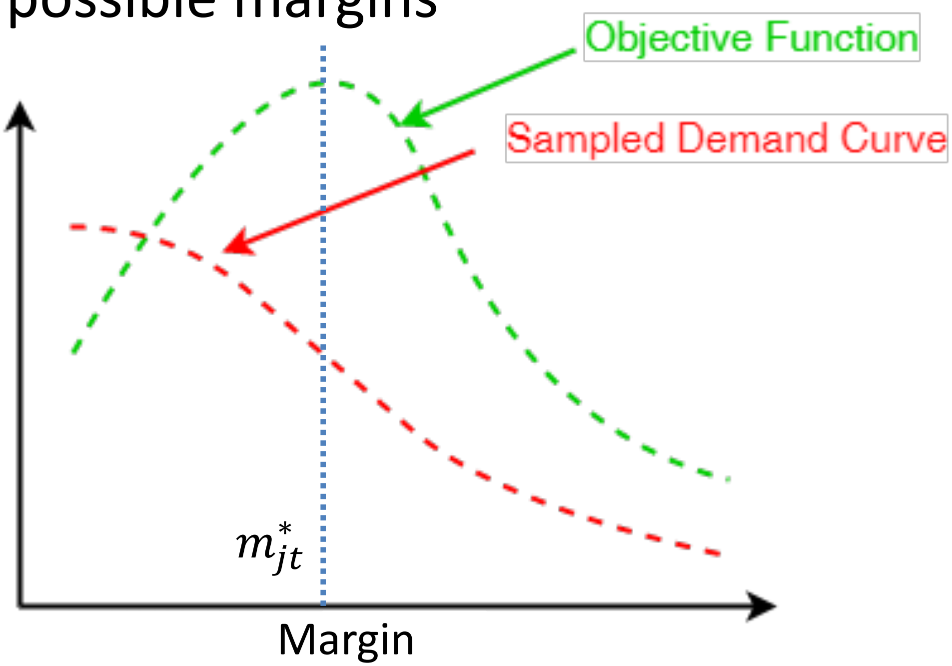
- BLR uncertainty over weights' posterior to sample[3] a single curve $\hat{v}_{jt}(\cdot, \tilde{\boldsymbol{w}}_{jt})$
- Objective function $\hat{v}_{jt}(\cdot, \tilde{\boldsymbol{w}}_{jt})$ is computed over the range of possible margins



[5] Thompson, William "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples." *Biometrika* (1933): 285-294.

Dealing with Price Exploration

- BLR uncertainty over weights' posterior to sample[3] a single curve $\hat{v}_{jt}(\cdot, \tilde{\boldsymbol{w}}_{jt})$
- Objective function $\hat{v}_{jt}(\cdot, \tilde{\boldsymbol{w}}_{jt})$ is computed over the range of possible margins



Chosen margin m^*_{jt} s.t.

$$m^*_{jt} = \underset{m}{\operatorname{argmax}}\ \hat{v}\left(m, \tilde{\boldsymbol{w}}_{jt}\right) m\, c_j$$

[5] Thompson, William "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples." *Biometrika* (1933): 285-294.

Why Product Aggregation

- Most of the products do not register enough sales to fit the BLR model properly

- **Naïve Strategy:** for each long-tail product, apply the same margin as the most similar best-seller product

    - Best-sellers and long-tail products have **significantly different market behaviors**: the Naïve Strategy doesn't control the bias induced by the procedure

- Our goal is to design a way to aggregate data from **similar long-tail products**

Products Similarity

- The only technical data available are **products' textual descriptions**

$$\kappa_{ij} = tf_{ij} \log \frac{N}{df_i} \xrightarrow{\textit{Cosine similarity}} \mathcal{S} = \begin{bmatrix} 1 & \cdots & 0.3 \\ \vdots & \ddots & \vdots \\ 0.7 & \cdots & 1 \end{bmatrix}$$

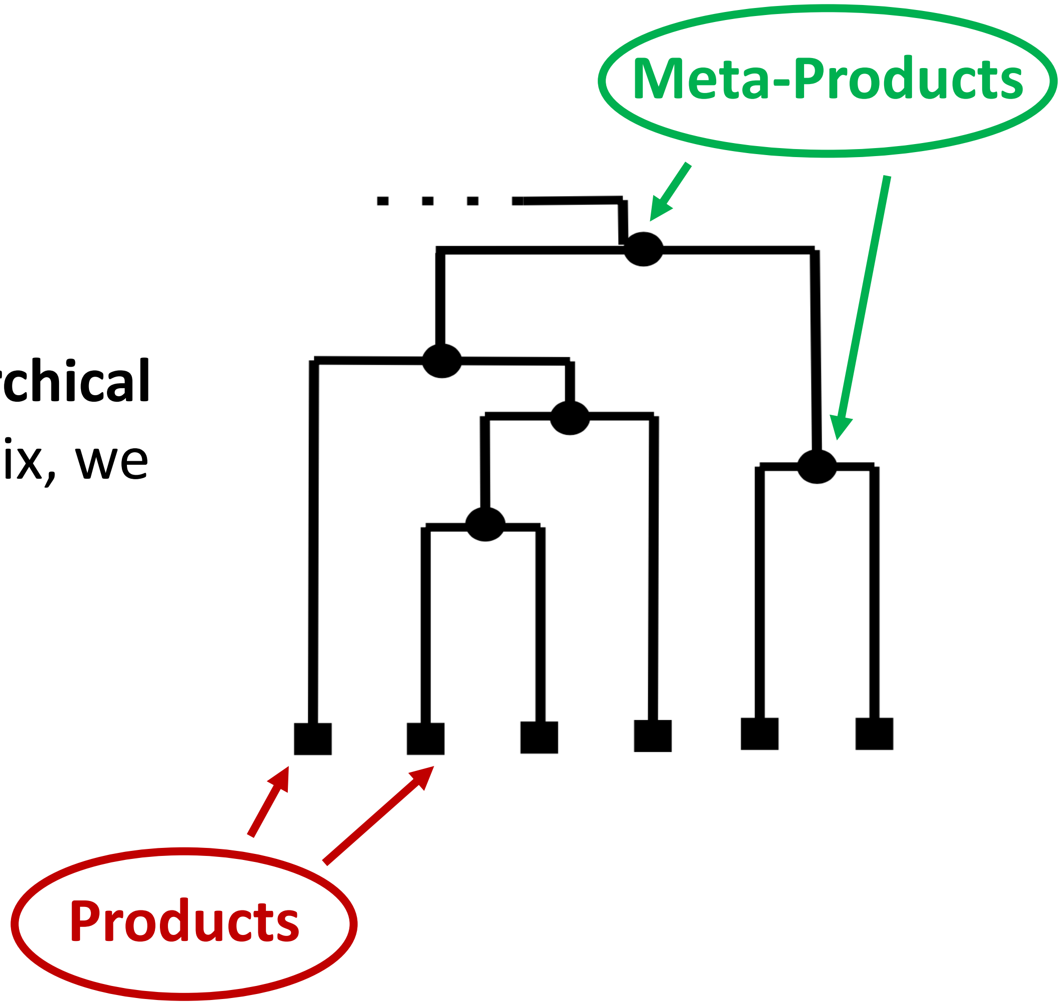
tf_{ij} = number of occurrences of word i in description of j

df_i = number of products containing i in description

- We resort to TF-IDF to **encode each product in a vector $\boldsymbol{\kappa}_j$**, then, **cosine similarity** is used to build a **matrix expressing the similarity** between each couple of products

Aggregation Structure

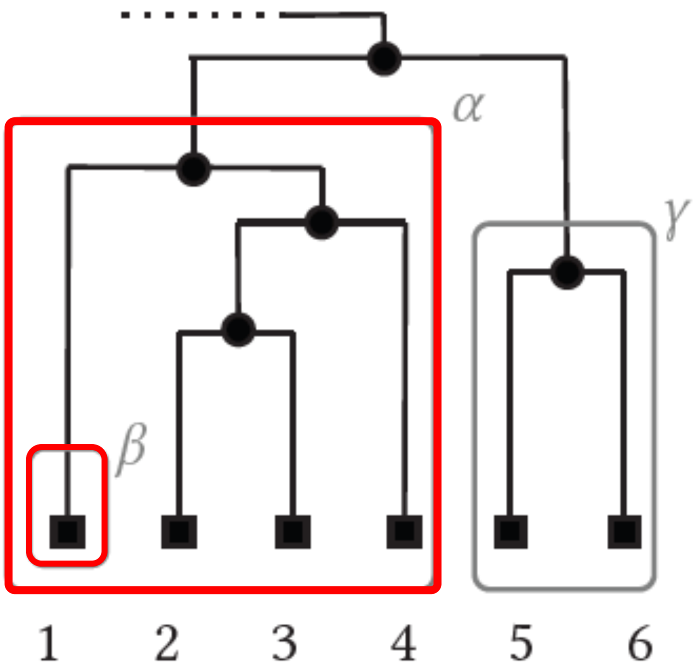
Using a **single linkage hierarchical clustering** on similarity matrix, we construct a **tree structure**



Product Aggregation Strategy

- Threshold τ represents the minimum amount of data samples required to start the single product pricing routine

- For every product j:
  - If its non-zero samples are **at least** τ, the product can be priced alone
  - If not, we **climb up a level** in the tree and **recursively perform this check** on the next meta-product



Data Sample Aggregation

- Once the meta-product α related to product j has been found, the meta-product data are estimated:

$$v_{\alpha t} \coloneqq \sum_{k \in \alpha} v_{kt}$$

$$m_{\alpha t} \coloneqq \sum_{k \in \alpha} m_{kt} \frac{v_{kt}}{v_{\alpha t}}$$

- Treating the meta-product as a single product, compute its **optimal margin,** and apply it **to product** j

Experimental Campaign

Experimental Setting

- The solution is tested in **both synthetic** and **real** environments
- An A/B test is performed on a real e-commerce over 7826 products with an yearly turnover of 2.5MEuros
- The test lasted $\boldsymbol{T}$ **= 8 weeks**
- The **same time period** $\boldsymbol{C}$ of the **previous year** is considered as benchmark

Configuration A

$N_A = 5694$ products

Priced by our algorithm

Configuration B

$N_B = 2132$ products

Priced by human specialist

Evaluation Metric

- The chosen metric is the **total profit** collected during the **period of 8 weeks**

$$M(A,T) := \sum_{t=1}^{T} \sum_{j \in A} v_{jt} m_{jt} c_j$$

- Due to the different overall magnitude in the two sets' weekly profits, we compare the **ratio of profits between the two periods**

$$G := \frac{M(A,T)}{M(A,C)} \frac{M(B,C)}{M(B,T)}$$

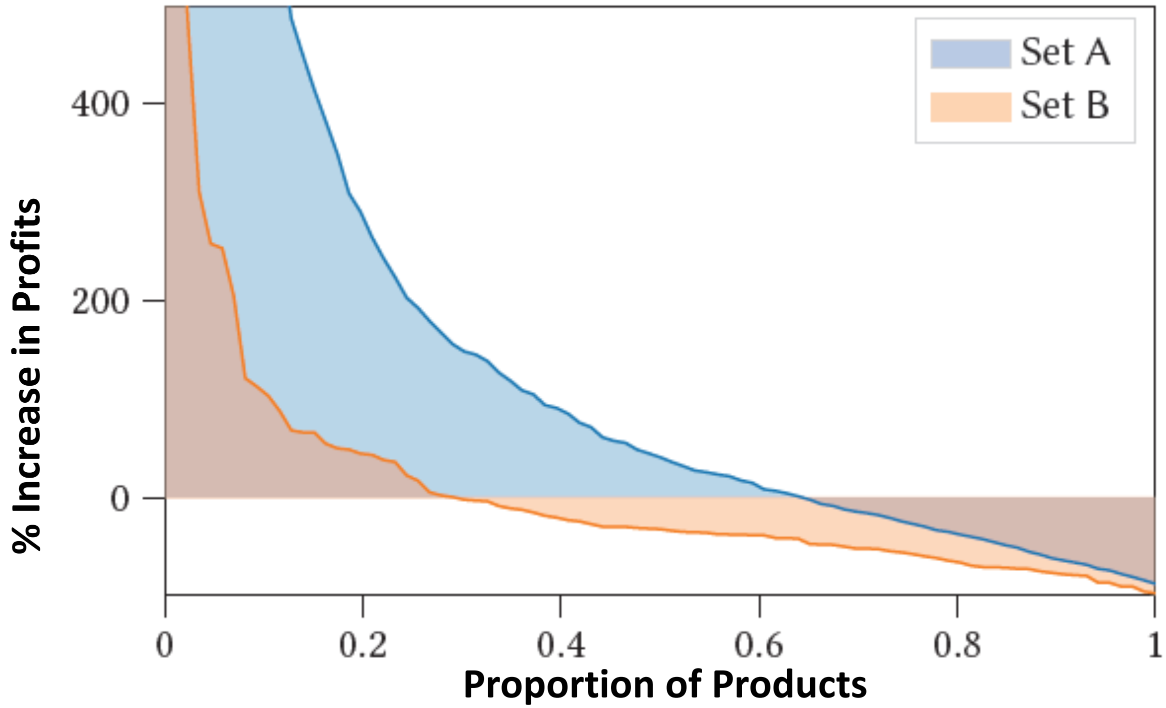
Overall Results

- We **separated the analysis** between bestsellers (or *popular*) products and long-tail ones

- The metric is **in favor of our algorithm**, attesting an increment in profits w.r.t previous year that is 40% **higher in configuration A**

Popular G_P	**Long tail** G_{LT}	**Total** G
1.18	1.91	1.4

Results at Product Level



Configuration A had a positive increase in profits in 65% of the product. Configuration B had only 29% of the products improved from previous year

Conclusions

- We provided a novel dynamic pricing algorithm especially suited for **long-tail** business models
- We proposed an **aggregation strategy** to automatically group products with too **scarce data**
- We evaluated the methodology through a **real-world campaign**, obtaining results in favor of the algorithm

Future Works

- Integration of **advertising** and **recommendation** strategies in the long-tail framework

- Modelling of products' **interactions** in term of units sold

- Analyzing the performances under a more complex **user behavior model**

Thank you for the attention!

Take a look at our work