

GIM | RELEVANCE COUNTS.

Pricing models for very saturated tech markets

AI vs. classical approaches

Neli Dilkova-Gnoyke
Senior Expert - Advanced Analytics



GIM | RELEVANCE COUNTS.



Background and Objectives



GIM | RELEVANCE COUNTS.

? BACKGROUND

Pricing studies for technical consumer goods.

- Technical goods markets are vast.
- Many brands and models, sometimes 300+
- Many brand-specific features = prohibitions
- Intense alternative-specific pricing necessary
- Clients need actionable insights fast

We researchers:

...usually use evoked set questions to strongly reduce the brand/model number in the tasks.

...need many respondents to get enough observations on brand/model.

...end up with large and sparse choice datasets (“Swiss cheese pattern”).

🎯 OBJECTIVES

PREDICTION is in focus! –

need for generalization to real world pricing scenarios

Find a model which is optimal in respect to

- External predictive power (main criterion)
- Internal predictive power
- Speed

In the best case all of these at once 😊



EVOKED SET DEFINITION

An evoked set is a set of products which a consumer considers in their purchase process.

I.e., it is a subset of products which is individually specific.



EXAMPLE OF AN EVOKED SET ELICITATION



Choice of relevant brands

E.g.: "Please choose between 1 and 4 brands which you would consider buying."



Choice of relevant models

E.g.: "For each brand you selected, please choose between 1 and 5 models which you would consider buying."

The decision space is reduced to max. 20 individually relevant products per respondent.

PRICING SURVEY FOR A TECHNICAL GOOD

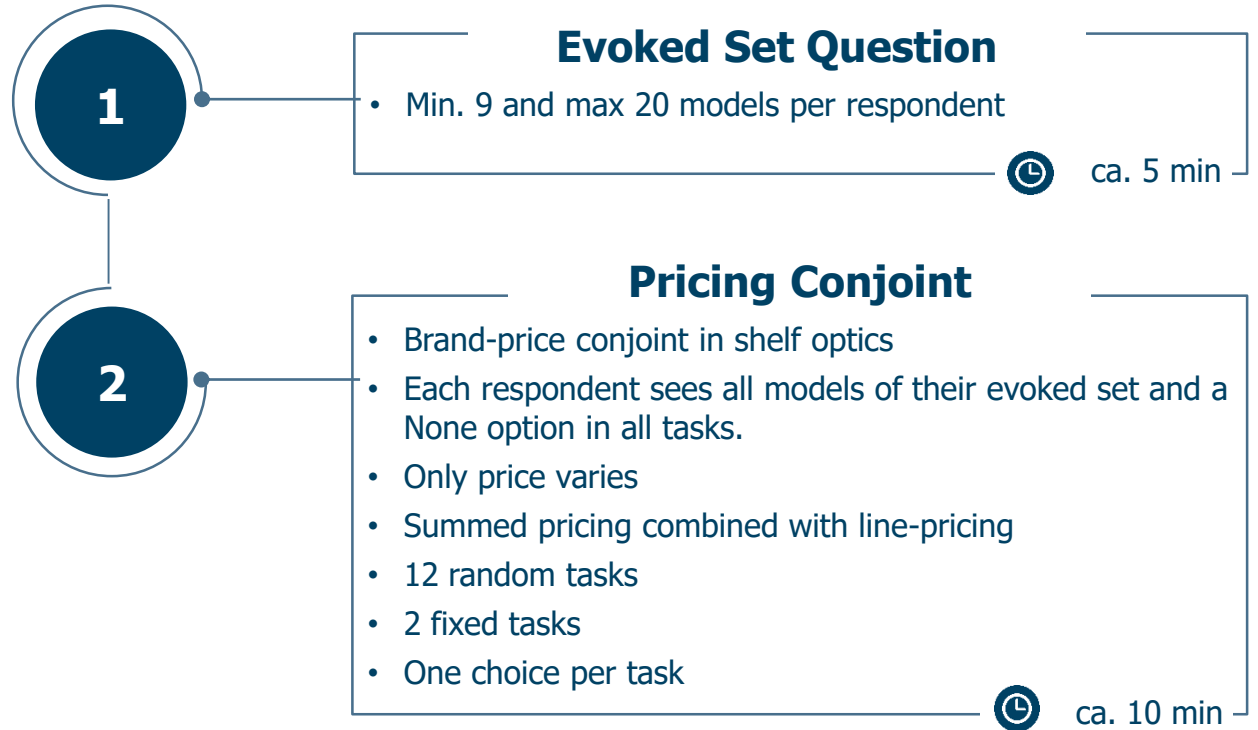
321 brand/model combinations (65% of total market sales)

Model-specific line pricing



- **Methodology:** CAPI
- **Duration:** ca. 25 minutes overall
- **Markets:** one country
- **Sample:** n=339 cleaned cases, purchase intenders

STUDY SETUP



Some details about the data used in this research

- 321 brand/model combinations
- 65% of total market sales coverage
- Model-specific line pricing
- Average evoked set size 11 models, i.e. 3.4% of all models in the study
- Model chosen most often 6% of all possible choice situations
- Choice of NONE option 2% of all possible choice situations
- Models never chosen 18 models

-> Many variables and few observations per variable!

Dimensions of the full dataset are 44820 x 326.

Each row includes min. 94% zeroes!

Extract from the full dataset, concepts belonging to the evoked set are marked in blue

1	ID	TaskID	Choice_ID	Price	1_1	2_1	3_1	4_1	5_1	6_1	7_1	8_1	9_1	10_1	11_1
1981	11121	1	0	3,4619	1	0	0	0	0	0	0	0	0	0	0
1982	11121	1	2	3,5346	0	1	0	0	0	0	0	0	0	0	0
1983	11121	1	0	3,778	0	0	1	0	0	0	0	0	0	0	0
1984	11121	1	0	4,179	0	0	0	0	0	0	0	0	0	0	0
1985	11121	1	0	4,3081	0	0	0	0	0	0	0	0	0	0	0
1986	11121	1	0	4,4323	0	0	0	0	0	0	0	0	0	0	0
1987	11121	1	0	3,6077	0	0	0	0	0	0	0	0	0	0	0
1988	11121	1	0	3,8917	0	0	0	0	0	0	0	0	0	0	0
1989	11121	1	0	4,2329	0	0	0	0	0	0	0	0	0	0	0
1990	11121	1	0	0	0	0	0	0	0	0	0	0	0	0	0
1991	11121	2	0	4,3199	1	0	0	0	0	0	0	0	0	0	0
1992	11121	2	0	4,3254	0	1	0	0	0	0	0	0	0	0	0
1993	11121	2	0	4,6238	0	0	1	0	0	0	0	0	0	0	0
1994	11121	2	0	4,0334	0	0	0	0	0	0	0	0	0	0	0
1995	11121	2	39	4,0186	0	0	0	0	0	0	0	0	0	0	0
1996	11121	2	0	4,1747	0	0	0	0	0	0	0	0	0	0	0
1997	11121	2	0	4,2434	0	0	0	0	0	0	0	0	0	0	0
1998	11121	2	0	4,5578	0	0	0	0	0	0	0	0	0	0	0
1999	11121	2	0	4,974	0	0	0	0	0	0	0	0	0	0	0
2000	11121	2	0	0	0	0	0	0	0	0	0	0	0	0	0
2001	11121	3	0	3,8972	1	0	0	0	0	0	0	0	0	0	0
2002	11121	3	0	3,9198	0	1	0	0	0	0	0	0	0	0	0
2003	11121	3	0	4,1377	0	0	1	0	0	0	0	0	0	0	0
2004	11121	3	0	4,1372	0	0	0	0	0	0	0	0	0	0	0
2005	11121	3	0	4,3586	0	0	0	0	0	0	0	0	0	0	0
2006	11121	3	0	4,5019	0	0	0	0	0	0	0	0	0	0	0
2007	11121	3	0	3,6826	0	0	0	0	0	0	0	0	0	0	0
2008	11121	3	217	3,887	0	0	0	0	0	0	0	0	0	0	0
2009	11121	3	0	4,3091	0	0	0	0	0	0	0	0	0	0	0
2010	11121	3	0	0	0	0	0	0	0	0	0	0	0	0	0
2011	11121	4	0	3,6582	1	0	0	0	0	0	0	0	0	0	0
2012	11121	4	2	3,7186	0	1	0	0	0	0	0	0	0	0	0
2013	11121	4	0	3,9965	0	0	1	0	0	0	0	0	0	0	0
2014	11121	4	0	4,455	0	0	0	0	0	0	0	0	0	0	0
2015	11121	4	0	4,4389	0	0	0	0	0	0	0	0	0	0	0
2016	11121	4	0	4,7817	0	0	0	0	0	0	0	0	0	0	0
2017	11121	4	0	3,884	0	0	0	0	0	0	0	0	0	0	0
2018	11121	4	0	4,1339	0	0	0	0	0	0	0	0	0	0	0
2019	11121	4	0	4,4121	0	0	0	0	0	0	0	0	0	0	0
2020	11121	4	0	0	0	0	0	0	0	0	0	0	0	0	0
2021	11121	5	0	3,4502	1	0	0	0	0	0	0	0	0	0	0



How to model such a Swiss cheese-type dataset? – two classical approaches

- Option 1: individual logistic regression for each respondent

Easy to build up and fast to calculate. But prone to overfitting - each model fitted to a specific respondent.

How would it predict on unseen respondents? – My hypothesis: poorer than any other model.

- Option 2: Hierarchical Bayes estimation:

Main-effects HB model for each brand/model combination with linear price parameters -> 304 individual models to be built overall

And what about Machine Learning Algorithms? Let us try out the most promising one suggested by the literature.



Research Design



GIM | RELEVANCE COUNTS.



Motivation for this research

Presentation by Dimitri Belyakov at the SKIM/Sawtooth European Conference in 2019:

“Applying machine learning to CBC data”

- Application of XGBoost and Neural Networks to “regular” sized CBC problems.
- Explanation of various approaches for data handling of choice data so that they can be modelled using XGBoost.
- Performance comparison based on paired comparisons.

Dimitri came to the conclusion that HB is brilliant and ML cannot beat it.

Presentation by Keith Chrzan and Joseph Retzer at the Sawtooth Software Conference in San Diego in 2019:

“Trees, Forests and Situational Choice Experiment”

- Findings on the application of decision trees on unconditional choice data. RF predicted choices better than MNL on various occasions
- In a 10-fold cross validation study the Gradient Boosting algorithm catboost outperformed MNL out-of-sample

But what about sparse choice data?

XGBoost is a Gradient Boosting algorithm optimized for sparse data. Let’s try it on sparse choice data and challenge the classical approaches.



Short introduction to XGBoost

"XGBoost is an optimized distributed gradient boosting library designed to be highly efficient, flexible and portable. It implements machine learning algorithms under the Gradient Boosting framework. XGBoost provides a parallel tree boosting (also known as GBDT, GBM) that solve many data science problems in a fast and accurate way. The same code runs on major distributed environment (Hadoop, SGE, MPI) and can solve problems beyond billions of examples."

- Basically a very clever and very fast ML technique for regression and classification problems - a highly efficient implementation of gradient boosting. It produces a prediction model which is an ensemble of weak prediction models, typically decision trees (i.e., simple model in the core of it: chains of if-then-conditions).
- Open source library available for multiple programming languages and operating systems
- Was the algorithm of choice for many winning teams of machine learning competitions
- Needs wide format of the dataset and one hot encoding (i.e. binarization) of the categorical variables.

For choice data, this means bringing each task for each respondent in one row.

Example of a xgb-format for brand-price conjoint experiment data

Choice_ID	197_9	198_9	199_9	200_9	201_9	202_9	203_10	204_10	205_10	206_10
242	0	0	0	0	0	0	0	0	0	0
242	0	0	0	0	0	0	0	0	0	0
242	0	0	0	0	0	0	0	0	0	0
242	0	0	0	0	0	0	0	0	0	0
242	0	0	0	0	0	0	0	0	0	0
242	0	0	0	0	0	0	0	0	0	0
241	0	0	0	0	0	0	0	0	0	0
235	0	0	0	0	0	0	0	0	0	0
258	0	0	0	0	0	0	0	0	0	0
258	0	0	0	0	0	0	0	0	0	0
241	0	0	0	0	0	0	0	0	0	0
235	0	0	0	0	0	0	0	0	0	0
229	3,8067	4,0743	4,3106	0	0	0	0	0	0	0
229	4,264	4,7196	5,0014	0	0	0	0	0	0	0
229	3,7051	4,1184	4,1856	0	0	0	0	0	0	0
197	3,823	4,0163	4,3281	0	0	0	0	0	0	0
231	3,5885	3,8408	4,0664	0	0	0	0	0	0	0
197	4,2023	4,5587	4,7599	0	0	0	0	0	0	0
197	3,8329	4,0741	4,2005	0	0	0	0	0	0	0
197	3,8158	4,1616	4,3265	0	0	0	0	0	0	0
197	3,5448	3,9044	4,0422	0	0	0	0	0	0	0
197	3,3538	3,6149	3,8336	0	0	0	0	0	0	0
197	3,531	3,8535	4,092	0	0	0	0	0	0	0
197	3,6508	3,9117	4,1166	0	0	0	0	0	0	0
235	0	0	0	0	0	0	0	0	0	0
235	0	0	0	0	0	0	0	0	0	0
234	0	0	0	0	0	0	0	0	0	0
234	0	0	0	0	0	0	0	0	0	0
304	0	0	0	0	0	0	0	0	0	0
304	0	0	0	0	0	0	0	0	0	0
299	0	0	0	0	0	0	0	0	0	0
235	0	0	0	0	0	0	0	0	0	0

METHODS IN THE TEST



Individual logistic regressions based on individual evoked set



Hierarchical Bayes, main effects only



XGBoost



COMPARISON BASED ON

- Individual Hit Rates (within sample)
- Mean Absolute Error (within and out-of sample)

5-fold cross-validation.

Training on random 75% of total respondents.

Prediction of two holdout tasks.







Results and Learnings



GIM | RELEVANCE COUNTS.

INDIVIDUAL HIT RATES*

ACROSS BOTH HOLDOUT TASKS

	NONE included	NONE excluded
	40%	38%
	38%	39%
	19%	22%
	0.3%	0.3%

* Hit rates can only be computed within sample.



▶ INTERPRETATION




Individual logistic regressions and HB provide similar individual hit-rates both with and without considering the NONE.

XGBoost performs worse on hit rates.

All models perform significantly better than chance.

MEAN ABSOLUTE ERROR

ACROSS BOTH HOLDOUT TASKS, including NONE

	Within sample	Out of sample
	0.24%	0.38%
	0.32%	0.45%
	0.30%	0.25%

INTERPRETATION




Individual logistic regressions deliver lowest in-sample MAE, but out-of sample XGBoost performs much better.

Out of sample individual logistic regressions and HB came out quite close.

Possibly overfitting of the logistic regression model.

MEAN ABSOLUTE ERROR

ACROSS BOTH HOLDOUT TASKS, without NONE

	Within sample	Out of sample
	0.24%	0.37%
	0.25%	0.38%
	0.29%	0.24%

INTERPRETATION

XGBoost performs best out of sample when the NONE option is not considered.

Again, individual logistic regressions and HB are at par.

Could be an effect specific to this dataset. Possibly due to price being the only variable for each model and because interaction effects were almost never significant in this dataset.

Possibly an indicator for homogeneity of respondents with similar evoked sets for this type of product.

COMPUTATION SPEED

DURATION OF COMPUTATIONS FOR ONE RUN ON AN AVERAGE PC FOR THIS DATASET

(on one core only, for the sake of a fair comparison)



IMPLICATIONS

All else being equal, XGBoost is a time-saving option compared to HB.

It would become the more time saving, the more respondents we use, because of the parallelization possibility which comes with it.

I only used one core on my PC with the XGboost algorithm. It can run much faster if distributed across cores. This can be done rather easily.



LEARNINGS

XGBoost can be used with conditional choice data, given you prepare your dataset accordingly.

XGBoost would be my method of choice over classical approaches if we:

- need to provide results for a highly complex brand-price choice modeling study
- with high external validity
- fast.

It works very well with sparse datasets.

The larger the dataset, the higher the benefit in terms of speed.



References



GIM | RELEVANCE COUNTS.

Burnham, K. P.; Anderson, D. R. (2002), *"Model Selection and Multimodel Inference"* (2nd ed.), Springer-Verlag

Belyakov, D. (2019) SKIM/Sawtooth European Conference: *"Applying machine learning to CBC data"*

Chen, Tianqi; Guestrin, Carlos (2016). *"XGBoost: A Scalable Tree Boosting System"*. In Krishnapuram, Balaji; Shah, Mohak; Smola, Alexander J.; Aggarwal, Charu C.; Shen, Dou; Rastogi, Rajeev (eds.). *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*. ACM. pp. 785–794. *arXiv:1603.02754*

Brathwaite, T., A. Vij and J.L. Walker (2017) *"Machine Learning Meets Microeconomics: The Case of Decision Trees and Discrete Choice," arXiv preprint arXiv:1711.04826*.

Chrzan, K., Retzer, J. (2019) Sawtooth Software Conference San Diego, USA: *"Trees, Forests and Situational Choice Experiments"*

Sekhar, C.R., Minal, and E. Madh (2016) *"Mode Choice Analysis Using Random Forrest Decision Trees," Transportation Research Procedia, 17: 644 – 652*

<https://xgboost.readthedocs.io/en/latest/>, retrieved July 30th 2020

CONTACT



Neli Dilkova-Gnoyke

Senior Expert – Advanced Analytics

+49 15151853957

n.dilkova@g-i-m.com



GIM | RELEVANCE COUNTS.

**GIM | Gesellschaft für
Innovative Marktforschung mbH**

Goldschmidtstraße 4 - 6
69115 Heidelberg

www.g-i-m.com