Big Data and Machine Learning in Finance Conference

June, the 10th, 2021

Learning Bermudans

Riccardo AIOLFI

Joint work with: Nicola MORENI, Marco BIANCHETTI, Marco SCARINGI, Filippo FOGLIANI

https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3837499

Contents

Optimal Stopping Time Problems

- Stochastic control problems *(H.Föllmer and A.Schied, 2008)*

- **Optimal stopping time**: find the best strategy about when to **stop a "game"** in order to **maximize** an **expected reward** or **minimize** an **expected cost** related to a stochastic process. The strategy can be any but, to be meaningful, it must be based only on **past and present information**

- Let $\mathcal{H}_t$ be a stochastic process representing the environment,

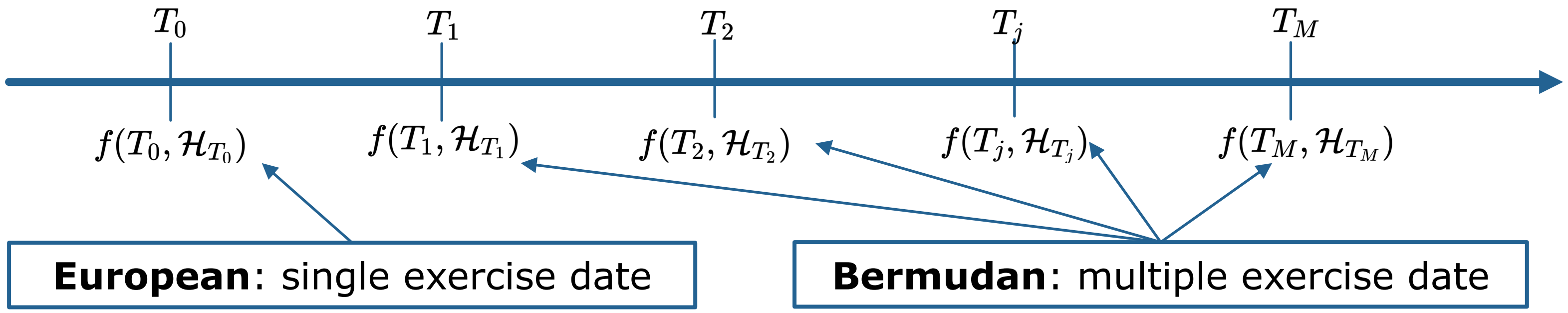
  let $\tau \in \mathbb{T}$ be an admissible strategy (represented by a **random variable**)

  and let $f(\tau, \mathcal{H}_\tau)$ be the **gain** if the **game** is **stopped** according to the chosen strategy

- The **optimal stopping time problem** consists in finding, whenever it exists, a strategy $\tau^* \in \mathbb{T}$ maximizing the expected gain

$$\sup_{\tau \in \mathbb{T}} \mathbb{E}[f(\tau, \mathcal{H}_\tau)] = \mathbb{E}[f(\tau^*, \mathcal{H}_{\tau^*})]$$

Optimal Stopping Times in Finance

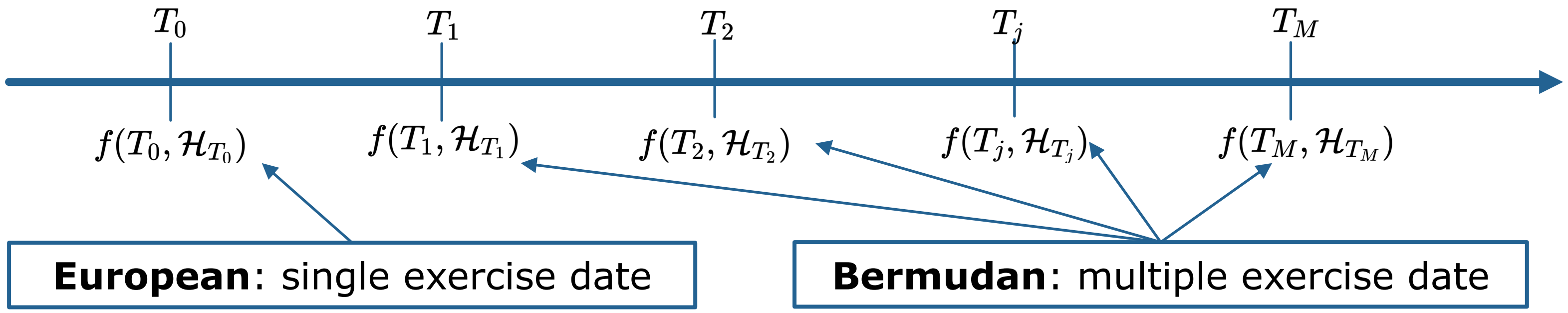


- In finance there are many problems of this type arising for instance when pricing products known as **Bermudan** options; these contracts gives the buyer the right to enter a financial transaction (like buying a security) at a mutually agreed price and at the time which is the best among a predetermined set $\{T_0, T_1, \ldots, T_M\}$

- Simple **strategies** are those in which we always (or we can only) exercise at one precise time T_j

- The **expected gain** $\mathbb{E}[f(\tau_i, \mathcal{H}_{\tau_i})]$ with each of these **trivial stopping times** $\tau_i \equiv T_i$ represents the **price** of what is called a **European option**. Often European option prices **are available as market quotations**

Bermudan *vs* European Optionality



- **European options** corresponding to **Bermudan exercise dates** give **less optionality** and hence their prices are **minorants** of the price of the **Bermudan** option

$$\mathbb{E}[f(\tau_i, \mathcal{H}_{\tau_i})] \leq \mathbb{E}[f(\tau^*, \mathcal{H}_{\tau^*})] \quad \forall \tau_i$$

- In particular the **price of a Bermudan** option **is greater** than **the maximum** of the **corresponding European ones**
- In some sense, the **more** the **European** option **prices** are "**uncorrelated**", the **higher** the **price** of the **Bermudan** option. For example, even if at the time T_1 it may not be convenient to exercise, nonetheless the probability that it will be profitable to exercise the right at T_j could be non negligible. This type of "independence" gives value to Bermudan optionality
- Moreover the **European options** are the **natural hedges** of the Bermudan option *(Hagan 2002)*

Numerical Solutions

- Typically Bermudan options are priced solving a recursive algorithm called ***dynamic programming,*** which involves computing conditional expectations at all times
- Dynamic programming in finance is often tackled by means of Trees, or Monte Carlo techniques. However the **computational burden** required to obtain the solution can be **non-negligible,** especially for high-dimensional systems
- Many Monte Carlo algorithms *(J. Barraquand and D. Martineau,1995, M.Brodie and P.Glasserman,1997, J. N.Tsitsiklis and B. van Roy,2001)* have been proposed which, by introducing some approximations, reduce this complexity, but they are still **tied** to the **efficiency** of the **Monte Carlo methods** and above all to the computational power available
- Recently authors *(Gaspar et al. 2020, Becker et al.2020, Lapeyre et al.2020)* employ **Artificial Neural Networks** to **solve dynamic programming** or to **approximate** the **optimal exercise boundary.** These approaches are nonetheless still based on Monte Carlo sampling.

⇓

- We want to exploit **SL algorithms** in a **new fashion** to price Bermudan options **overcoming** the **computational bottleneck** of dynamic programming techniques based on **numerical simulations**

Formulation of the problem

- **Our idea** is to use Supervised Learning algorithms to **catch** the functional **relationship** between the **price of Bermudan option**, the relevant **European option prices** and a measure of **their correlations,** thus avoiding long and complex Monte Carlo simulations
- As a by-product, these algorithms could help us to **understand** the most important **driving factors** behind the market price

Research questions

Information about Bermudan option (**independent variable**)

Bermudan option price (**dependent variable**)

Supervised Learning algorithms

Interest Rate Swap (IRS) options

- We focus on **interest rates derivatives** and wish to apply our idea to the pricing of Interest Rate Swap Bermudan options (aka **Bermudan Swaptions**)
- These instruments are relevant as they are **embedded in callable debt instruments** or traded in the OTC market for **speculation** purposes
- A **Bermudan swaptions** gives the right to chose the time to **enter an Interest Rate Swap** (IRS) where a series of **fixed rate** interests are exchanged against **floating rate** interests up to a given maturity date
- An IRS is said to be of type **payer (receiver)** if the fixed rate interests are **paid (received)**
- The dates at which the IRS can be entered are called **option expiries**
- We consider products in which, regardless of the date at which the option is exercised, the maturity of the IRS is kept the same (**co-terminal** IRS). The length between the first admissible expiry date and the maturity is called **tenor**
- The time interval from today to the first expiry is the **non-call period**
- The **strike** of the option is expressed as the **fixed rate** of the underlying IRS

Creation of the dataset

- As we want to calibrate SL regression algorithms, we wish to use a dataset where the **independent variables** span a sufficiently wide range of values
- In particular we want to ensure scenarios with high and low variance/covariances between the forward prices of **co-terminal** IRS
- We then built an *in-vitro* dataset based on a simple **short rate model**
- The **two** fundamental **pillars** for creating our **coherent dataset** are

Linear Gaussian One Factor Model (G1++)
(J.Hull and A.White,1990)

Least Square Monte Carlo (LSMC)
(F.A. Longstaff and E. S. Schwartz,2015)

Linear Gaussian One Factor Model

$$r(t) = x(t) + \varphi(t) \qquad \begin{cases} dx(t) = -ax(t) + \sigma dW^{\mathbb{Q}}(t) \\ x(0) = 0 \end{cases}$$

- Market interest rate curve is recovered by design thanks to displacement function $\varphi(t)$
- Single stochastic factor, used to **sample** the interest rate curve **stochastic dynamics**

⇩

We exploit the **two G1++ parameters**, i.e. **speed of mean reversion** a and **volatility** σ to create many different market scenarios that **differ** in the global **level** of **variances/covariances** between co-terminal IRS

Dataset I

Our *in vitro* dataset contains a total of **4340 Bermudan Swaption prices**

- **10 different pairs** of G1++ **volatility** and **speed of mean reversion** to ensure adequate coverage of interest rate curve variance/covariance scenarios (t=0 curve is kept the same).
- **434 Swaptions** for each pair of G1++ parameters $\{a, \sigma\}$. Swaptions differ in contractual specs (non-call period, tenor, strike)

80% training set (3472 samples)

20% test set (868 samples)

Dataset II

FEATURES

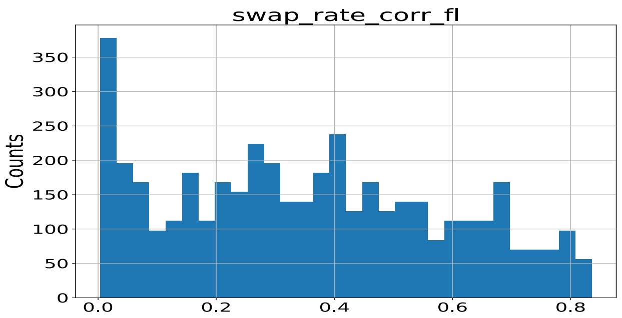
1. **Tenor** $\in \{2Y, 5Y, 10Y, 15Y, 20Y\}$

2. **Strike** $\in \{-100, -75, -60, -50, -40, -30, -25, -20, -15, -10, -7, -5, -2, 0, 20, 25, 30, 50, 100, 200, 300, 400\}$
   (shift w.r.t. IRS par-rate in bps)

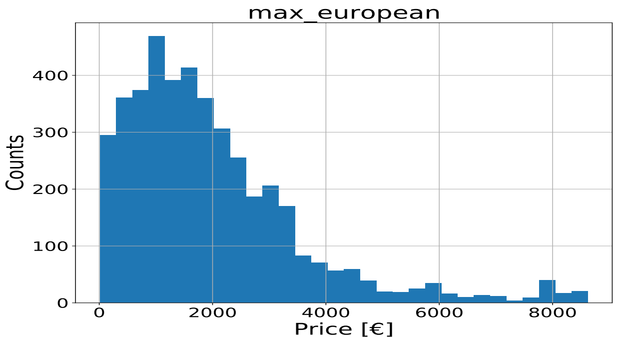
3. **Side** $\in \{\text{payer}, \text{receiver}\}$

4. **Non-call period** $\in \{1Y, 2Y, 3Y, 4Y, 5Y, 7Y, 10Y, 15Y, 20Y\}$

5. **Correlation (between swap rates)**
   $\in [0.0035, 0.8356]$

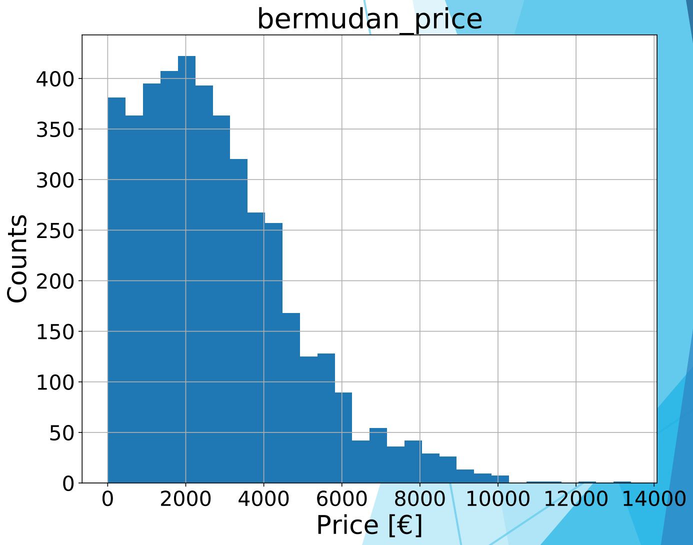


6. **Maximum underlying European Swaption price**
   $\in [11.53€, 8621.53€]$



TARGET

1. **Bermudan price**
   $\in [14.85€, 13405.17€]$



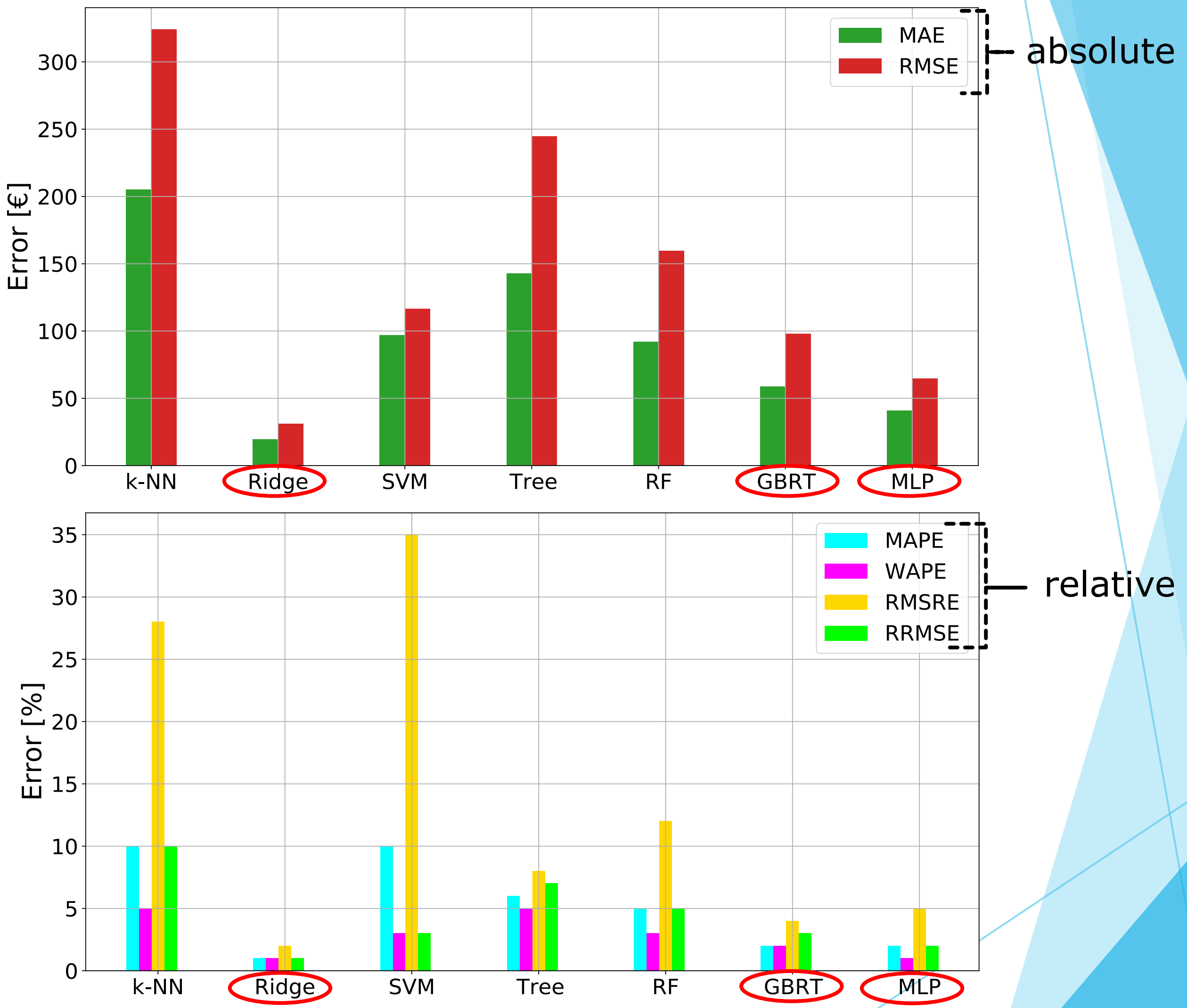
Supervised Learning algorithms

We have implemented with **scikit-learn** and **TensorFlow**:

- K-Nearest Neighbours (k-NN)
- Linear Models:
  - Linear Regression
  - Ridge Regression
  - Lasso Regression
- Support Vector Machine (SVM)
- Decision Tree
- Ensemble of Decision Trees:
  - Random Forest (RF)
  - Gradient Boosted Regression Tree (GBRT)
- Artificial Neural Networks (or Multilayer Perceptron MLP)

Results I

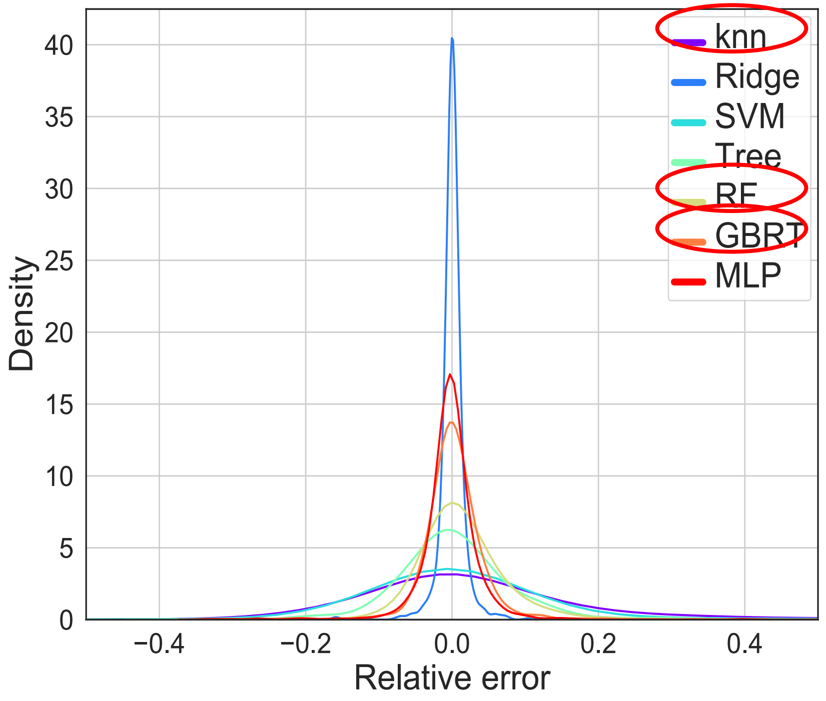
- The best performers are **Ridge**, **MLP** and **GBRT**
- Apart from a few exceptions, the result of the comparison between two models does not change in a relevant way if we observe different metrics: "***best algorithms are always the best***"
- Since in our case the goal was to predict **prices** over an **extended range with different scales**, we believe a **"relative" metric** is **more suitable** than an "absolute" one as it makes the **data more homogeneous**
- We can say that the average price error of the **Ridge**, equal to **1%**, is an **excellent result** if compared to some consensus price services where the average standard deviation is roughly **2%** of the price



Results II

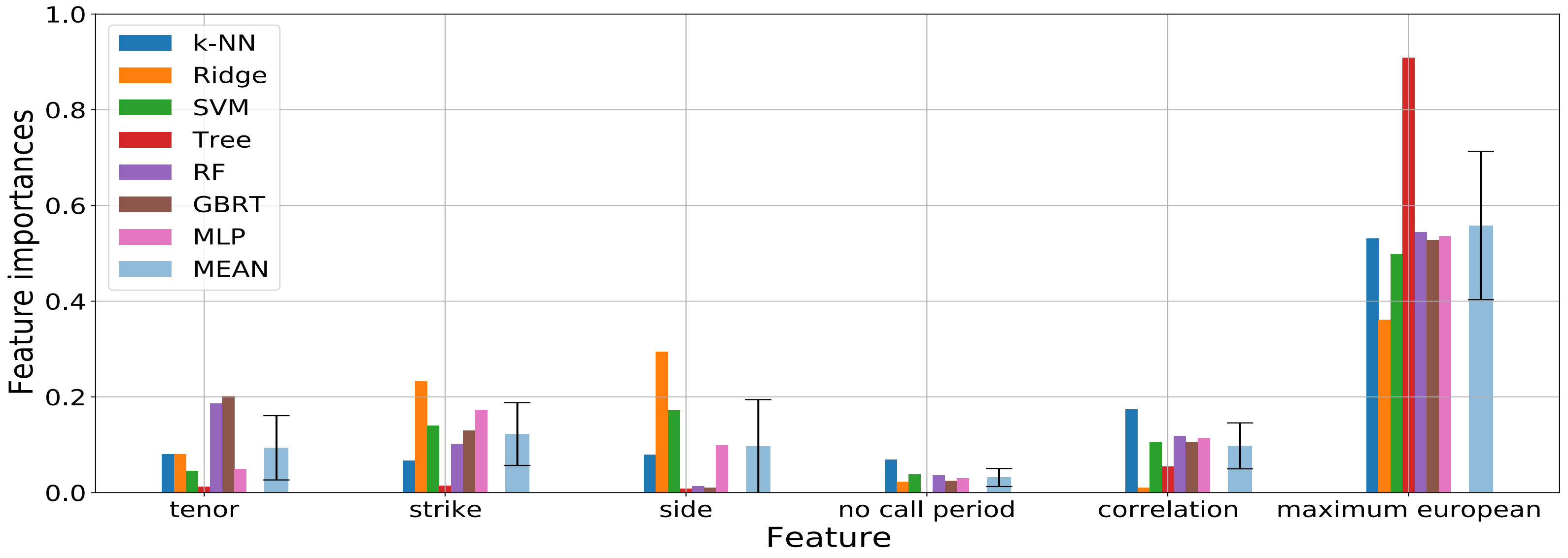
Statistics of relative error for each algorithms

	k-NN	Ridge	SVM	Tree	RF	GBRT	MLP
mean	0.0524	0.0006	0.0475	0.0329	0.0243	0.0036	0.0002
std	0.2771	0.0182	0.3422	0.8381	0.1153	0.0444	0.0041
skew	6.6101	-0.2800	9.6765	0.7100	7.2053	0.3922	1.2675
kurtosis	66.9179	24.2352	126.0146	5.6378	71.2143	17.3850	58.2962
min	-0.4514	-0.1608	-0.8059	-0.3820	-0.2713	-0.4172	-0.4506
25%	-0.0517	-0.0049	-0.0344	-0.0402	-0.0154	-0.0157	-0.0128
50%	-0.0053	-0.0004	-0.0029	-0.0014	0.0049	0.0001	-0.0013
75%	0.0747	0.0061	0.0353	0.0412	0.0363	0.0191	0.0119
max	3.9007	0.1549	5.2834	0.5939	1.5604	0.3683	0.5659



- **Ridge**, **GBRT** and **MLP** have the average values **of relative error closest to zero** with the **lowest standard deviation**
- From values of skewness, kurtosis and quantiles these models possess the most **symmetrical error distributions without one-side bias**

Results III



- All the **tree-based** models have an **endogenous** feature importance estimator, while for all the **other models** an **indirect** method has been used (**permutation importance**)
- Even though with different weights, **all models** point out the **price** of the **maximum European swaption** as the **most relevant** feature (lower bound)
- Furthermore, except for the non-call period, the other features have comparable average values, with the only difference that the **correlation** has the **lowest standard deviation**, a sign that the weights returned by the individual algorithms are very similar to each other

Conclusions

- **Supervised Learning techniques** for the pricing of Bermudan instruments is **promising**. They **overcome** the **computational bottleneck** of numerical simulations and help us to **understand** the most important **driving factors** behind the options market price

- The best models are **Ridge**, **Artificial Neural Network** and **Gradient Boosted Regression Tree**. The **Ridge** has the advantage of having one of the **fastest** training phase, while the **MLP** is the **slowest** and requires a more accurate tuning of its hyperparameters. The **GBRT** require **no data preparation** phase and have an **internal method** for evaluating the importance of features

- Thanks to **feature importance** techniques we have confirmed that the **most determining factor** for the price of a Bermudan option is the value of the **maximum** underlying **European swaption,** which constitutes its lower bound

Comparison for 434 Bermudan swaption*

Algorithm	Training Time**	Pricing time
LSMC***	/	1086.6 s
KNN	12.9×10^{-3} s	9.4×10^{-3} s
Ridge	316×10^{-3} s	7.7×10^{-3} s
SVM	736×10^{-3} s	161×10^{-3} s
Tree	15.9×10^{-3} s	4×10^{-3} s
RF	2.1 s	71.3×10^{-3} s
GBRT	1.6 s	10.9×10^{-3} s
MLP	28 s	98.7×10^{-3} s

* On a MacBook Pro (MacOS version 10.15.7) with an Intel Quad-Core i5 2.3 GHz processor with a memory of 2133 MHz and 8GB of RAM

** With 3472 Bermudan swaption

*** 5 x 10^4 path Monte Carlo for each Bermudan swaption

Future developments

I. The **dataset** could be **extended/refined**
  - → Scenarios + termsheet specifics
  - → Variance coherence

II. Assess model extrapolating capabilities on new scenarios with extreme variance/covariance values

III. Since it is **very challenging** to **infer** the **correlations** from the market, our approach could also prove helpful the **other way round**: it is possible to **estimate correlations** from recent **market** Bermudan swaption **prices** and then use them as the input of our algorithms to get today prices

Bibliography

- H.Föllmer and A. Schied*. "Stochastic Finance"*. Berlin, Boston: De Gruyter, 2008.
- Paul Glasserman. *"Monte Carlo Methods in Financial Engineering".* Springer, 2003.
- Francis Longstaff and Eduardo Schwartz. *"Valuing american options by simulation: A simple least-squares approach"*. Working paper, The Anderson School, UCLA, 1998
- J. Barraquand and D. Martineau. *"Numerical Valuation of High Dimensional Multivariate American Securities"*. In: The Journal of Financial and Quantitative Analysis 30.3 (1995), pp. 383–405
- M. Broadie and P. Glasserman. *"Pricing American-style securities using simulation"*. In: Journal of Economic Dynamics and Control 21.8 (1997). Computational financial modelling, pp. 1323–1352
- J. N. Tsitsiklis and B. van Roy. *"Regression Methods for Pricing Complex American Style Options"*. In: IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council 12 (Feb. 2001), pp. 694–703.
- Sebastian Becker, Patrick Cheridito, and Arnulf Jentzen. *"Deep optimal stopping"*, 2020.
- Raquel M. Gaspar, Sara D. Lopes, and Bernardo Sequeira. *"Neural network pricing of american put options"*. Risks, 8(3), 2020.
- Sebastian Becker, Patrick Cheridito, and Arnulf Jentzen. *"Pricing and hedging american-style options with deep learning"*. Journal of Risk and Financial Management, 13:158, 07 2020.
- Bernard Lapeyre and Jerôme Lelong. *"Neural network regression for bermudan option pricing",* 2020.
- Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *"The Elements of Statistical Learn- ing. Springer Series in Statistics"*. Springer New York Inc., New York, NY, USA, 2001.
- Patrick Hagan. *"In the trenches adjusters: Turning good prices into great prices".* Wilmott, 2002:56–59, 12 2002.

Disclaimer

The views and the opinions expressed in this document are those of the authors and do not represent those of their employers. They can not be held responsible for any use that may be made of these contents. This document is written for informative purposes only, it is not intended to influence any investment decisions or promote any product or service.