Backward Stochastic Differential Equation : Numerical Results

Majdi Rabia

February 20, 2017

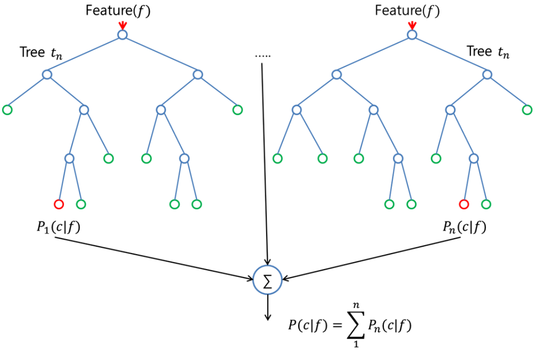
Sommaire

Exmple of a Non-linear driver : bid-ask model

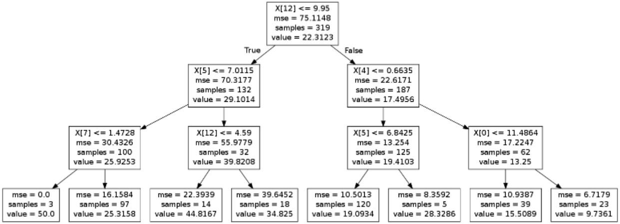
$$\begin{cases} f(t, Y_t, Z_t) = -Z_t\theta - rY + (R-r)(Y - \dfrac{Z_t}{\sigma})^- \\ \theta = \dfrac{\mu - r}{\sigma} \\ \xi(X_T) = (X_T - K_1) - 2(X_T - K_2) \end{cases}$$

Sommaire

Overview


Feature(f)
Tree t_n
.....
Feature(f)
Tree t_n
$P_1(c|f)$
$P_n(c|f)$
$\sum$
$P(c|f) = \sum_1^n P_n(c|f)$

Max leaf nodes



This parameter helps avoid an overfitting by taking bigger values, but reaching a point, this will lead to some inaccuracies(cf following analysis on a specific example)

Accuracy

- **n_estimators** : Represents the number of trees randomly generated. We try to take the biggest value possible in our simulations, as the MSE decreases with this parameter.
- **max_features** : The number of features to consider when looking for the best split:
- **max_depth** : The maximum depth of the tree
- **min_samples_split** : The minimum number of samples required to split an internal node
- **min_samples_leaf** : The minimum number of samples required to be at a leaf node

Performance

- **n_jobs** : The number of jobs to run in parallel for both fit and predict
- **warm_start** : When set to True, reuse the solution of the previous call to fit and add more estimators to the ensemble, otherwise, just fit a whole new forest.

- **Accuracy** : We focused especially on the maximum number of leafs allowed in the Random Forest simulation, which seemed to be the most sensitive parameter in one dimension. However, in higher dimension, results are not bad, but could be improved by constrain on other parameters.
- **Performance** Warm_start parameter could be helpful. From what I understand, it seems to keep in memory the last generated trees, and adding useful other estimators. This leads to better time performance, but I am going to analyse this on simple data to understand exactly how it works. Indeed, it could be used for $\mathbb{E}[Y_{t+dt}|\mathcal{F}_t]$, as it is stable, but might be dangerous for $\mathbb{E}[Y_{t+dt}\Delta B_t|\mathcal{F}_t]$

$N = 1000$

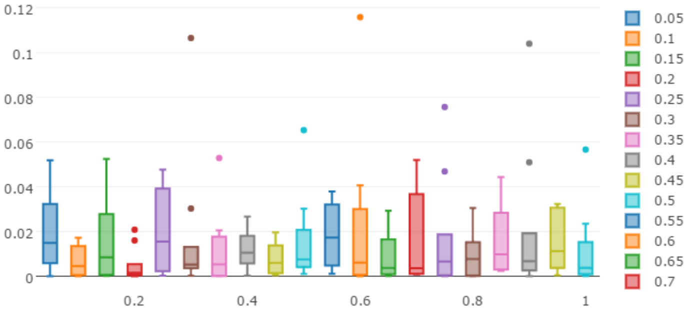


Figure: MSE against $\frac{RF\_max\_leaf\_nodes}{N}$ with N = 1000

$N = 10000$

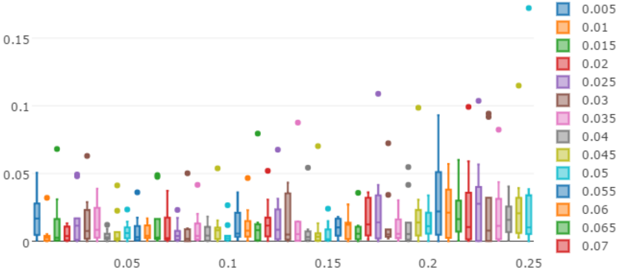


Figure: MSE against $\frac{RF\_max\_leaf\_nodes}{N}$ with N = 10000

$N = 10000$

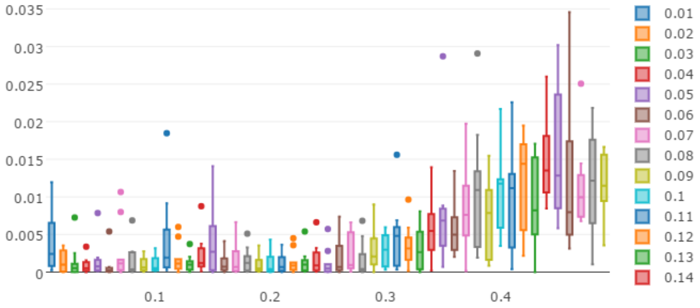


Figure: MSE against $\frac{RF\_max\_leaf\_nodes}{N}$ with N = 100000

Remarks

- Getting a clear analysis from the previous three figures is not easy. It seems like there exists a stable region (with high variance) around 10 to 20%.
- We will take 10% in the following simulations.

Sommaire

Overview

		MESH	RF	GRADBOOST	LSM	DERIVATIVE	SOURCE	driver	correlation
1D	bid_ask_european_call						Gobet	non_linear	NO
	bid_ask_call_combination						Gobet	non_linear	NO
	cva						Guyon	non_linear	NO
	american_option						Glasserman	non_linear	NO
HD	bid_ask_european_call_7						theory	non_linear	NO
	bid_ask_european_call_25						theory	non_linear	NO
	bid_ask_european_call_100						theory	non_linear	NO
	bid_ask_call_combination_20							non_linear	NO
	spread_correlated						labart, lelong	non_linear	YES
	max_five_assets_european						glasserman	linear	NO

European Call One Dimension : expected $= 7.15$

EXPECTED	7.15

	stat parameter	values	time average	N_particles	n_picard
Derivative	mean	7.1779	4.01s	1000	3
Derivative	std	0.3088			
Derivative	95% confidence interval	[7.1476, 7.2082]			
Derivative	min	6.7418			
Derivative	max	7.7752			
RandomForest	mean	7.1417	70s	10^4	5
RandomForest	std	0.0815			
RandomForest	95% confidence interval	[7.1337, 7.1497]			
RandomForest	min	7.0288			
RandomForest	max	7.2926			
Mesh	mean	7.1875	100s	4000	10
Mesh	std	0.1387			
Mesh	95% confidence interval	[7.1739, 7.2011]			
Mesh	min	6.9085			
Mesh	max	7.4492			
LSM	mean	7.1553	15.9s	10^5	20
LSM	std	0.0375			
LSM	95% confidence interval	[7.1516, 7.159]			
LSM	min	7.0907			
LSM	max	7.2242			

Spread Call One Dimension : expected = 2.95 (Gobet)

Expected	2.96

	stat parameter	values	time average	N_particles	n_picard
RandomForest	mean	2.9694	3s	10^4	10
RandomForest	std	0.0436			
RandomForest	95% confidence interval	[2.9651, 2.9737]			
RandomForest	min	2.8896			
RandomForest	max	3.0422			
Derivative	mean	2.9403	2.5s	10^3	3
Derivative	std	0.1441			
Derivative	95% confidence interval	[2.9121, 2.9685]			
Derivative	min	2.7534			
Derivative	max	3.2832			
LSM	mean	2.9381	10s	10^5	10
LSM	std	0.0154			
LSM	95% confidence interval	[2.9366, 2.9396]			
LSM	min	2.9159			
LSM	max	2.9787			
Mesh	mean	2.841	35s	4.10^3	10
Mesh	std	0.0711			
Mesh	95% confidence interval	[2.834, 2.848]			
Mesh	min	2.6986			
Mesh	max	2.9682			
GradBoosting	mean	2.9687	10s	10^4	3
GradBoosting	std	0.0465			
GradBoosting	95% confidence interval	[2.9641, 2.9733]			
GradBoosting	min	2.886			
GradBoosting	max	3.0544			

European bid-ask Basket Option on 7 assets : expected = 3.30 (Black-Scholes)

Expected	3.308

	stat parameter	values	time average	N_particles	n_picard
RandomForest	mean	3.3135	70s	10^4	3
RandomForest	std	0.0455			
RandomForest	95% confidence interval	[3.309, 3.318]			
RandomForest	min	3.24			
RandomForest	max	3.415			
Mesh	mean	3.3186	140s	3000	3
Mesh	std	0.0831			
Mesh	95% confidence interval	[3.3105, 3.3267]			
Mesh	min	3.1293			
Mesh	max	3.4872			
GradBoosting	mean	3.3409	41s	10^4	3
GradBoosting	std	0.0434			
GradBoosting	95% confidence interval	[3.3366, 3.3452]			
GradBoosting	min	3.2336			
GradBoosting	max	3.3868			
Derivative	mean	3.3306	40s	10^3	3
Derivative	std	0.1796			
Derivative	95% confidence interval	[3.2602, 3.401]			
Derivative	min	3.0466			
Derivative	max	3.5348			

Spread Call on 20 assets : expected = 5.71 (simulation in 1dim)

Expected	5.71

	stat parameter	values	time average	N_particles	n_picard
RandomForest	mean	5.6999	8min24s	10 000	3
RandomForest	std	0.0188			
RandomForest	95% confidence interval	[5.6981, 5.7017]			
RandomForest	min	5.6574			
RandomForest	max	5.7359			
GradBoosting	mean	5.7064	8min20s	10 000	3
GradBoosting	std	0.0186			
GradBoosting	95% confidence interval	[5.7046, 5.7082]			
GradBoosting	min	5.67			
GradBoosting	max	5.7388			
Derivative	mean	5.6941	41min	4 000	3
Derivative	std	0.0281			
Derivative	95% confidence interval	[5.6913, 5.6969]			
Derivative	min	5.6357			
Derivative	max	5.7342			
Mesh	mean	5.6885	5min48s	3 000	3
Mesh	std	0.0438			
Mesh	95% confidence interval	[5.6842, 5.6928]			
Mesh	min	5.6116			
Mesh	max	5.7929			

Bid-ask Call on 25 and 100 assets using only RandomForest

Dimension : 25

Expected	2.382
mean	2.3879
std	0.0185
95% confidence i	[2.3861, 2.3897]
min	2.3626
max	2.4327
time average	7min12s

Dimension : 100

Expected	2.019
mean	2.0341
std	0.0126
95% confidence i	[2.0329, 2.0353]
min	2.0194
max	2.0531
time average	7min12s

Max Call on 5 assets (linear driver)

EXPECTED		23.052	GLASSERMAN	

	stat parameter	values	time average	N_particles
GradBoosting	mean	23.0663	4s	10000
GradBoosting	std	0.2093		
GradBoosting	95% confidence interval	[23.0458, 23.0868]		
GradBoosting	min	22.6828		
GradBoosting	max	23.445		
Mesh	mean	22.9485	29s	3000
Mesh	std	0.4441		
Mesh	95% confidence interval	[22.905, 22.992]		
Mesh	min	22.0012		
Mesh	max	23.6536		
Derivative	mean	22.9131	14s	1000
Derivative	std	0.6985		
Derivative	95% confidence interval	[22.8446, 22.9816]		
Derivative	min	21.6485		
Derivative	max	23.9269		
RandomForest	mean	23.0165	7s	10000
RandomForest	std	0.286		
RandomForest	95% confidence interval	[22.9885, 23.0445]		
RandomForest	min	22.5737		
RandomForest	max	23.5364		

Payoff ($(S_1 - S_2)^+$) (linear driver but with Z)

Dimension : 2	Expected	4.134
	mean	4.1612
	std	0.0658
	95% confi	[4.1548, 4.1676]
	min	4.0351
	max	4.2798
	time aver	50s

Remarks

- The one dimensional examples look good after the calibration of Max_leaf_nodes parameter
- The high dimensional examples are quite accurate, but the tree regression is costly in time.
- Thinking about using warm_start parameter ...

Sommaire

- 2-BSDE
- writing
- hyperbole implementation maybe ...