Deep learning of implied volatility surface in time

April 22, 2022

Goal

Find an arbitrage-free interpolation of the market implied volatility surface. Arbitrage-free in our context means:

1. Positivity: For $m \in \mathbb{R},\ T \in \mathbb{R}_{>0}\ \sigma_{\mathrm{BS}}(m, T) > 0$
2. Differentiability: For $T > 0,\ m \mapsto \sigma_{\mathrm{BS}}(m, T)$ is twice differentiable in $\mathbb{R}$
3. Monotonicity: For $m \in \mathbb{R},\ T \mapsto \sqrt{T}\sigma_{\mathrm{BS}}(m, T)$ is increasing on $\mathbb{R}_{>0}$ and

$$\sigma_{\mathrm{BS}}(m, T) + 2T\frac{\partial \sigma_{\mathrm{BS}}(m, T)}{\partial T} \geq 0$$

4. Butterfly arbitrage-free: For $m \in \mathbb{R},\ T \in \mathbb{R}_{>0}$

$$\left[1 - \frac{m\frac{\partial \sigma_{\mathrm{BS}}(m,T)}{\partial m}}{\sigma_{\mathrm{BS}}(m, T)}\right]^2 - \frac{1}{4}\left[\sigma_{\mathrm{BS}}(m, T)\, T\frac{\partial \sigma_{\mathrm{BS}}(m, T)}{\partial m}\right]^2 + T\sigma_{\mathrm{BS}}(m, T)\frac{\partial^2 \sigma_{\mathrm{BS}}(m, T)}{\partial m^2} \geq 0$$

Goal

Find an arbitrage-free interpolation of the market implied volatility surface. Arbitrage-free in our context means:

5. Limit condition: If $T > 0$, then $\lim_{m \to \infty} d_+(m, T) = -\infty$
6. Right boundary: If $m \geq 0$, then

$$N\left(d_-(m, T)\right) - \sqrt{T}\frac{\partial \sigma_{\mathrm{BS}}(m, T)}{\partial m} n\left(d_-(m, T)\right) \geq 0$$

   where N denotes the standard normal cdf and n its pdf
7. Left boundary:

$$N\left(-d_-(m, T)\right) + \sqrt{T}\frac{\partial \sigma_{\mathrm{BS}}(m, T)}{\partial m} n\left(d_-(m, T)\right) \geq 0$$

8. Asymptotic slope: For $T > 0$, then $2|m| - \sigma_{\mathrm{BS}}^2(m, T)\, T > 0$

What we have

quote_date	root	expiration	strike	option_type	trade_volume	bid_size_1545	bid_1545	ask_size_1545	ask_1545	implied_volatility_1545
05.10.2020	SPXW	07.10.2020	3075	C	0	14	329.8	29	330.9	0.5305
05.10.2020	SPXW	07.10.2020	3075	P	269	150	0.1	634	0.15	0.5045
05.10.2020	SPXW	07.10.2020	3080	C	0	14	324.8	29	325.9	0.5229
05.10.2020	SPXW	07.10.2020	3080	P	53	986	0.05	743	0.15	0.4863
05.10.2020	SPXW	07.10.2020	3085	C	0	14	319.8	29	320.9	0.5153
05.10.2020	SPXW	07.10.2020	3085	P	0	989	0.05	574	0.15	0.4792
05.10.2020	SPXW	07.10.2020	3090	C	42	14	314.8	29	315.9	0.5077
05.10.2020	SPXW	07.10.2020	3090	P	13	892	0.05	532	0.15	0.472
05.10.2020	SPXW	07.10.2020	3095	C	0	14	309.8	29	310.9	0.5002
05.10.2020	SPXW	07.10.2020	3095	P	2	946	0.05	489	0.15	0.4649
05.10.2020	SPXW	07.10.2020	3100	C	16	14	304.8	29	305.9	0.4926
05.10.2020	SPXW	07.10.2020	3100	P	650	968	0.05	427	0.15	0.4578
05.10.2020	SPXW	07.10.2020	3105	C	0	14	299.8	29	300.9	0.485
05.10.2020	SPXW	07.10.2020	3105	P	13	1009	0.05	70	0.15	0.4506
05.10.2020	SPXW	07.10.2020	3110	C	32	14	294.8	29	295.9	0.4774
05.10.2020	SPXW	07.10.2020	3110	P	5	1263	0.05	51	0.15	0.4435
05.10.2020	SPXW	07.10.2020	3115	C	0	4	289.8	19	290.9	0.4699
05.10.2020	SPXW	07.10.2020	3115	P	12	10	0.1	33	0.15	0.4464
05.10.2020	SPXW	07.10.2020	3120	C	0	14	284.8	29	285.9	0.4623
05.10.2020	SPXW	07.10.2020	3120	P	23	218	0.1	20	0.15	0.4392

Pre-processing

We removed data if

1. the bid-price was less or equal 0.375 to reduce noise
2. the implied volatility is less than 0.05 % to reduce noise

Put-Call Parity

By the Put-Call parity we have

$$V^C - V^P = DF - DK,$$

where V^C, V^P are the option prices, $D = e^{-rT}$ is the discount factor and K the strike.
Now, we can do a linear regression on the set

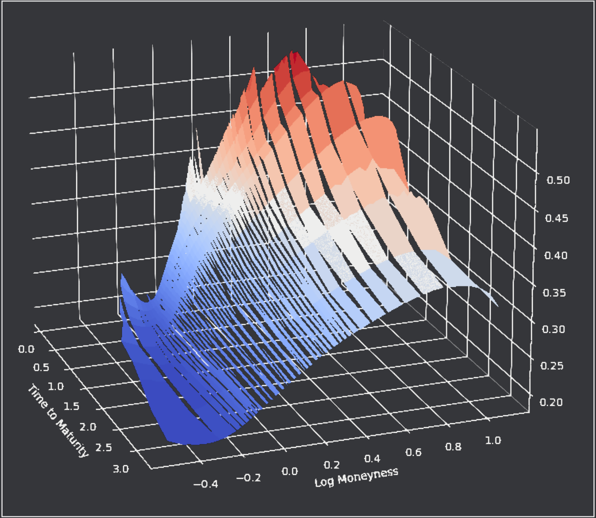
$$\left\{ \left(K_i, \left(V_i^C - V_i^P \right) \right) : i = 1, \ldots, n \right\}$$

and $n \in \mathbb{N}$ is the number of data points for time to maturity T.
The linear regression is defined as

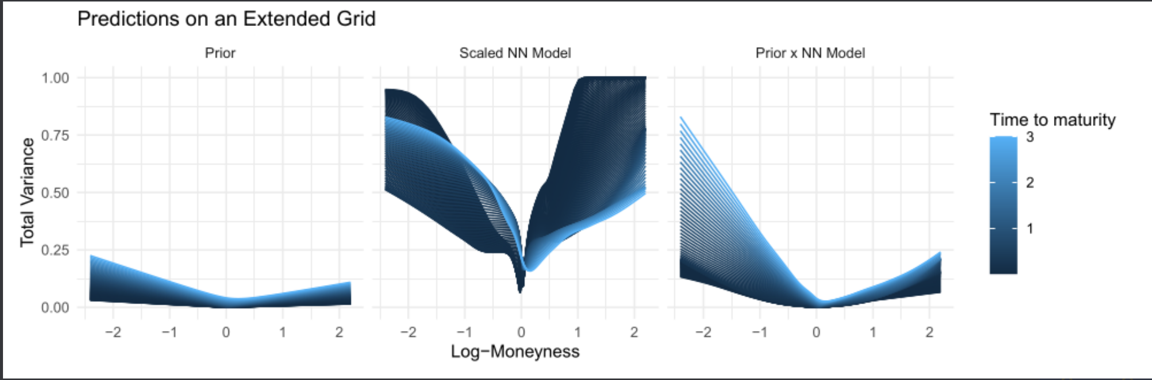
$$V_i^C - V_i^P = \alpha + \beta K_i + \epsilon_i$$

and as a result we get the discounted underlying price $S_0 = \alpha$ to compute the log-Moneyness.

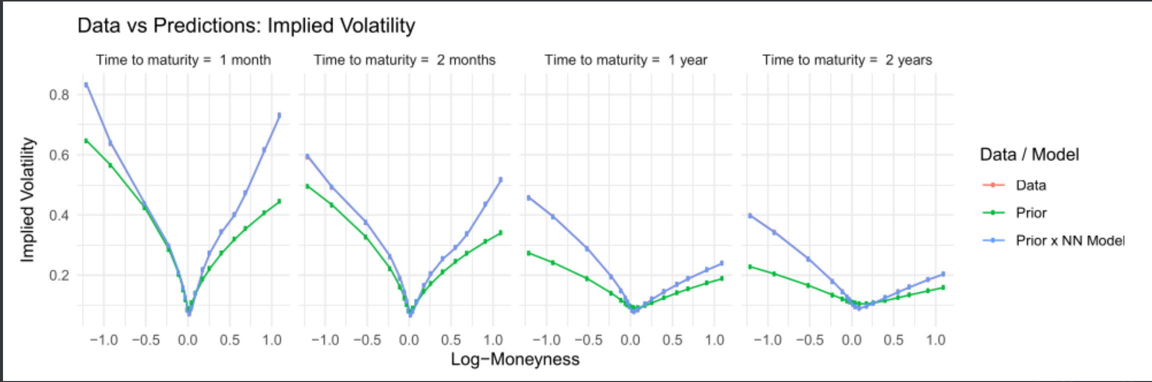
Market implied volatility surface



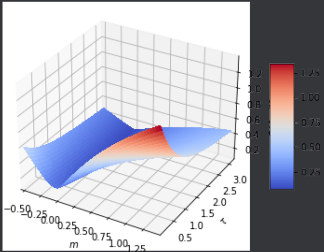
Our approach



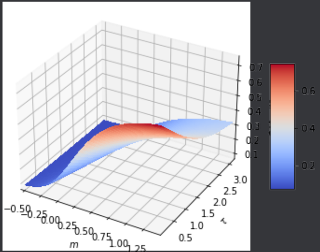
Our approach



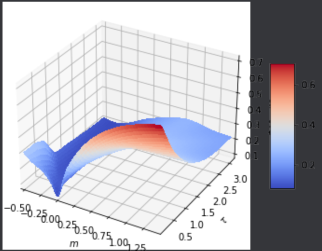
Zheng Model



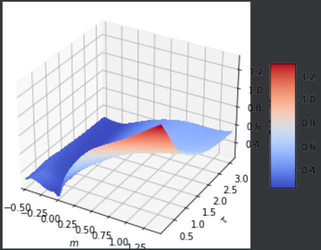
Zheng DNN

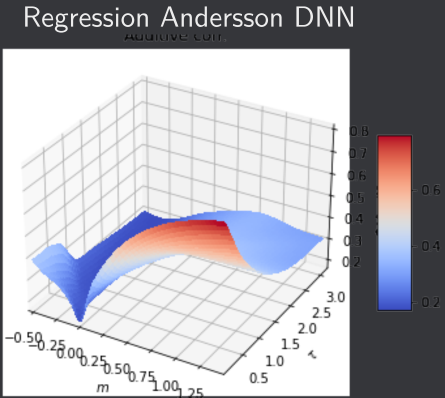


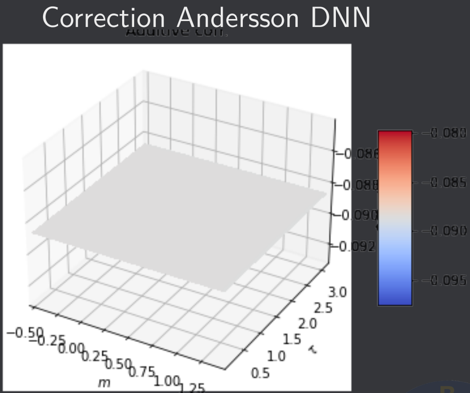
Andersson DNN



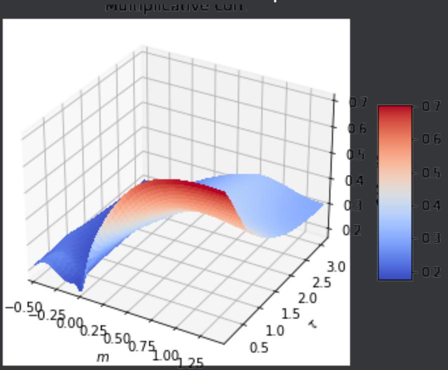
Andersson Multiplicative DNN



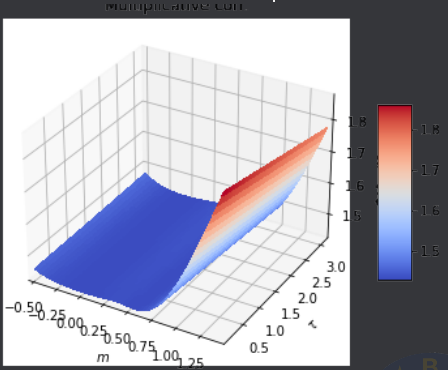

Regression Andersson DNN
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.6
0.4
0.2
3.0
2.5
2.0
1.5
1.0
0.5
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
1.25
m


Correction Andersson DNN
−0.080
−0.085
−0.090
−0.095
−0.086
−0.088
−0.090
−0.092
3.0
2.5
2.0
1.5
1.0
0.5
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
1.25
m

Regression Andersson Multiplicative DNN



Correction Andersson Multiplicative DNN



Comparison

	Zheng et al.	Zheng et al. -- DNN	Additive correction	Multiplicative correction
ATM	7.0996e-03	4.7290e-03	7.9634e-03	1.0675e-02
constraint 3	0.0000e+00	0.0000e+00	1.9906e-03	0.0000e+00
constraint 4	0.0000e+00	4.6294e-05	0.0000e+00	2.1758e-03
constraint 6	0.0000e+00	0.0000e+00	0.0000e+00	0.0000e+00
constraint 7	0.0000e+00	0.0000e+00	5.0924e-04	0.0000e+00
constraint 8	1.9027e-02	2.4906e-02	2.4351e-02	1.3768e-01

Possibilities for improvement

1. Cleaning up the market data even more/ partition into different regions
2. Take imbalances of the data set into account
3. Hard constraints
4. Condition for the perfect fit at-the-money
5. Hyperparameter-tuning/ architecture

Thank you for your attention!

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 813261 and is part of the ABC-EU-XVA project.