DOI 10.33111/nfmte.2021.046

MODELING RELATION BETWEEN AT-THE-MONEY LOCAL VOLATILITY AND REALIZED VOLATILITY OF STOCKS

Maksym Bondarenko

Consultant Developer C++
CapFi Groupe
25-27 Place de la Madeleine, Paris, 75008, France
ORCID: 0000-0002-4792-6420, E-mail: maksym_bondarenko@outlook.com

In this work we apply univariate and multivariate linear regressions to model the relation between at-the-money local volatility and realized volatility of stocks on the example of Microsoft shares.
Local volatility is extracted from the set of Vanilla option prices on Microsoft stocks by assuming that Microsoft stock price follows Dupire local volatility process. At-the-money local volatility at different maturities is then used in linear regression predictor while realized volatility is a resulting variable.
To handle the ill-posed character of Dupire calibration problem we use genetic algorithm of optimization. To obtain two local volatility datasets (regression inputs) two runs of the calibration are executed as we want to reflect the random nature of the genetic algorithm that can give slightly different values of local volatility for different runs.
The model validation is performed by predicting out-of-sample realized volatility using local volatility and comparing it to real world values of the realized volatility. The statistical significance of local volatility is measured as a predictor of realized volatility at different maturities in the article.
It is concluded that in all models the local volatility at longer maturities proves to be significant predictor of realized volatility (whether we predict realized volatility in a short time interval or in a longer one). Therefore it makes sense to predict the volatility on the market by calibrating local volatility from the options with longer maturities.

Keywords: *genetic algorithm, evolutionary optimization, local volatility, implied volatility, linear regression, multivariate regression, option pricing, Black–Scholes model, financial markets forecasting*

JEL Classification: C15, C61, G12

Modeling relation between at-the-money local volatility and realized volatility of stocks

In the context of growing uncertainty of financial markets the predicting of market volatility becomes increasingly important.

Banks, trading institutions and investment funds use volatility to price financial products and hedge risks.

In his study Burtnyak describes the importance of the valuation of the real options in making investment decisions [6]. Volatility plays the important role in such valuation and is an important parameter to predict.

Recently financial markets suffered a few shocks. Kaminsky and Nehrey in their paper study the impact of COVID-19 crisis on stock returns and the measures of stock prices recovery [13]. Realized market volatility is represented as the return of the stock during some period. By studying changes these returns and risk measures authors conclude that high levels of turmoil and uncertainty were generated during COVID crisis. Hence in this work we are interested specifically in data during COVID crisis as it generates jumps in volatility and allows us to make some predictions.

These studies inspired us to use implied volatility measure to predict realized volatility during the period of COVID crisis when multiple jumps are present. However, as Dupire study shows [11], local volatility model better represents volatility skew than Black-Scholes option pricing model [3]. So, we will use local volatility instead of the implied volatility to predict realized market volatility.

Multiple researches were conducted to predict the future returns of the stocks. In their paper Derbentsev, Matviychuk and Soloviev [9] shown efficiency of using machine learning approaches to predicting financial time series. By designing models, they explored different sets of features and lags. The final dataset contained only past values of target variable with 14 and 15 lag depth. In this case larger dataset provided better training for all models and given more efficient results. In their further work [10] was demonstrated the effectiveness of using models of neural networks, regression (autoregressive) trees and their ensembles for short-term cryptocurrency forecasting tasks. They have applied a simple model of neural network — the multilayer perceptron with one hidden layer. Based on their results, these models allow making short-term forecast with sufficient accuracy: within 3–4%. Results of the binary classification of the direction of price changes showed, that binary autoregressive tree and perceptron models had an average accuracy of about 63% for the daily time series observations, which was higher than for the “naive” model.

The abovementioned and other similar models are mainly concerned with the prediction of future stock prices based on time series of the past prices with certain time lag. However very few researches were conducted to predict the future returns expressed in terms of future volatility. Some studies aimed to predict future volatility using time series of past volatilities.

Christensen and Prabhala study the relation between the time series of volatility implied from historical option prices and realized market volatility. Their studies suggest that implied volatility is better predictor of realized volatility than past realized volatility [8]. This conclusion is important as it indicates that it makes sense to use implicit parameters in predicting realized volatility.

Some studies were dedicated to multivariate prediction of the realized volatility using implied volatility. Hence Luo and Zhang use 6- and 9-month forward variances in a multivariate model to predict the return of the underlying stock over 1-month period [14]. Implied volatilities are used as measures of forward variance.

Despite these achievements no studies were conducted to predict realized volatility from the time series of the past local volatility.

The purpose of this research is to model the relation between at-the-money (ATM) local volatility and realized volatility.

As only few models were built based on local volatility as independent variable to predict the future realized volatility, we firstly want to discover which maturities are more significant for the prediction of the realized volatility and whether such models are able to describe the dynamics of the realized volatility at all. To solve this task we use simple univariate and multivariate linear regressions as done in [8, 14].

In our previous research [5] we have proven that calibration date is neither a good predictor of the local volatility nor there are any autocorrelation tendencies observable for current time-series so we don’t take calibration date t as second factor to our regression model.

For this study local volatility is calibrated from prices of Vanilla options (Call and Put) available on the market. We want to calibrate such matrix at different calibration dates in the past to build the time-series. In order to calibrate local volatility we suppose that underlying stock price follows Dupire model dynamics because local volatility model better represents volatility skew than Black-Scholes option pricing model [11]. We use genetic algorithm of optimization of Cerf [7] to handle the ill-posed character of Dupire calibration problem in the same way as we do in [4].

We used 1^{st} order univariate and multivariate linear regressions to see if at-the-money local volatility at different maturities can predict the realized volatility. We validate the results on out-of-the-sample (OOS) validation points within two different datasets. To obtain two local volatility datasets (regression inputs) two runs of the calibration are executed as we want to reflect the random nature of the genetic algorithm that can give slightly different values of local volatility for different runs.

As suggested in [1] the modeling is carried out in several stages: the selection of variables, analysis of descriptive statistics, estimation of model parameters, selection of an adequate model. We provide the selection of variables later in this introduction, and thereafter three other stages are done for univariate and multivariate models.

To be able to calibrate time series of local volatility we need to choose a liquid asset (stock) whose historical prices are available on the market and were affected by COVID crisis. We have chosen Microsoft shares as an asset to do conduct the experiment. Therefore we use a set of prices of Vanilla options on MSFT stocks from NASDAQ [12] as input. Option prices are chosen for multiple trading dates between January and April 2020. This period contains the peaks of volatility caused by COVID crisis.

Calibration dates are historical trading dates at which option prices are available and at which we can calibrate local volatility. We use two different datasets as arrays of calibration dates to double check the model. The cross-validation and predictive sets of calibration dates are described in Table 1 and Table 2 respectively. In cross-validation dataset the dates of validation are chosen randomly. In predictive dataset the subset of calibration dates consists of first chronological part of the dataset while validation dates – all the latest dates. In both cases the proportion of calibration (sample) dates is about 2/3 of the whole dataset.

As input data is sparse and maturities/strikes differ between calibration dates we use linear interpolation on maturity and strike axis to obtain the values for the same normalized grid for each of calibration dates. Let K be the number of calibration dates. We see from Tables 1, 2 that $K = 10$.

The normalized grid by maturity axis is represented by these fractions of the year because they usually correspond to maturities for which market data is available:

$$Maturities\ grid = T = [\ 0.05, 0.1, 0.15, 0.2, 0.25, 0.4\ ], \qquad (1)$$

where $N = |Maturities\ grid|$ – number of maturities (here $N = 6$).

The normalized grid by strike axis is represented as following. The values chosen in a way that all the available strikes during the period lay inside the grid (no need to apply the extrapolation in this case):

$$Strikes\ grid = [\ 100, 105, 110, \dots, 220, 225, 230\ ], \qquad (2)$$

where $M = |Strikes\ grid|$ – number of strikes (here $M = 26$).

Our goal is to calibrate $N \times M$ matrix of local volatilities from the set of options for each of K calibration dates using evolutionary optimization.

Table 1

CROSS-VALIDATION DATASET

Calibration Dates	Validation dates (OOS)	Spot MSFT, $
1/3/2020		158.62
1/10/2020		161.34
	1/17/2020	167.10
1/24/2020		165.04
1/31/2020		170.23
2/7/2020		183.89
	2/14/2020	185.35
	2/21/2020	178.59
2/28/2020		162.01
3/6/2020		161.57
	3/13/2020	158.83
3/20/2020		137.35
3/27/2020		149.70
4/3/2020		153.83
	4/9/2020	165.14
	4/17/2020	178.60

Table 2

PREDICTIVE VALIDATION DATASET

Calibration Dates	Validation dates (OOS)	Spot MSFT, $
1/3/2020		158.62
1/10/2020		161.34
1/17/2020		167.10
1/24/2020		165.04
1/31/2020		170.23
2/7/2020		183.89
2/14/2020		185.35
2/21/2020		178.59
2/28/2020		162.01
3/6/2020		161.57
	3/13/2020	158.83
	3/20/2020	137.35
	3/27/2020	149.70
	4/3/2020	153.83
	4/9/2020	165.14
	4/17/2020	178.60

Firstly, we parametrize volatility by strike and maturity axis to reduce the dimensionality of the calibration problem and assure the continuity of strike axis [4]. For strike axis B-spline parametrization is used while for time axis we use linear interpolation. Spline parametrization is achieved by determining the values of volatility in control points $\theta = \{\theta_x^i, i = \overline{1,N}, x = \overline{1,X}\}$. Thus, the volatility by strike axis is represented by the following vector (spline) for each maturity T_i:

$$\sigma(T_i,\cdot) = \sum_{x=1}^{X} \theta_x^i \, \varphi_x(\cdot) \, \sigma(T_i,\cdot) = \sum_{x=1}^{X} \theta_x^i \, \varphi_x(\cdot), \qquad (3)$$

where θ_x^i – x^{th} control point for strike axis at maturity T_i, X <= U-j-1 – number of control points, U – number of uniform knots (U <= M),

j – degree of the polynomial, $\varphi_x(\cdot)$ – basis functions. Volatility is a function of strike so "·" designates vector of volatilities w.r.t. all of the strikes.

Secondly, we parametrize volatility on t-axis where it's represented a bit differently – linear interpolation is used now between points T_i and T_{i+1}. So, for any strike $\gamma \in \mathit{Strikes\ grid}$ the volatility is represented as [2]:

$$\sigma(t,\gamma) = \frac{T_{i+1}-t}{T_{i+1}-T_i}\sigma(T_i,\gamma) + \frac{t-T_i}{T_{i+1}-T_i}\sigma(T_{i+1},\gamma), \quad (4)$$

where T_i, T_{i+1} – respectively start and end of the interval in which the point t falls.

To handle the ill-posed character of Dupir calibration problem, we use genetic algorithm of optimization of Cerf [7]. At the first stage of the algorithm it is necessary to generate the initial population (p = 0) of control points $\theta_x^i, i = \overline{1,N}, x = \overline{1,X}$. Then for p^{th} population of candidate solutions of volatility $\theta(p)$, B-spline is built with help of such control points and $\sigma\big(t,\gamma,\theta(p)\big)$ is obtained for all $t \in \mathit{Maturities\ grid}$ and $\gamma \in \mathit{Strikes\ grid}$.

Then $\sigma\big(t,\gamma,\theta(p)\big)$ is inserted to Dupire equation solver and solution (prices) $C^{\theta(p)}(t,\gamma)$ is obtained for each t, γ on discretization grid for each p.

Prices are compared to real market prices with fitness function for each t, γ and for each p. The sum of squared differences is considered to measure the fitness of each p^{th} candidate [4]:

$$\inf_{\theta(p)} G\big(\theta(p)\big), \quad G\big(\theta(p)\big) =$$

$$= \sum_{t \in \mathit{Maturities\ grid}} \sum_{\gamma \in \mathit{Strikes\ grid}} \left(C^{\theta(p)}(t,\gamma) - C^{market}(t,\gamma)\right)^2 w_{t,\gamma}, \quad (5)$$

where $\inf_{\theta(p)} G\big(\theta(p)\big)$ designates the minimum of fitness function $G\big(\theta(p)\big)$ obtained using $\theta(p)$ – the matrix of control points forming candidate solutions $\sigma\big(t,\gamma,\theta(p)\big)$ of p^{th} population.

For numerical results we chosen desired value of the measure of fitness function (5) as 0.3, which was obtained experimentally after many runs of the algorithm, below that algorithm can hardly converge.

Genetic algorithm that is applied to calibrate $\theta(p)$ and by consequence $\sigma\big(t,\gamma,\theta(p)\big)$ for each calibration date τ has following steps.

Firstly, new population is formed. Candidate matrices of control points θ_x^i are then subject to genetic transformations (selection and mutation) based on the fitness of corresponding solutions for each *p*:

$$\theta(p) = \big(\theta(p,1),\dots,\theta(p,L)\big) \xrightarrow[selection]{} \\ \xrightarrow[selection]{} \widehat{\theta(p)} = \big(\theta\widehat{(p,1)},\dots,\theta\widehat{(p,L)}\big) \xrightarrow[mutation]{} \theta(p+1), \quad (6)$$

where L – size of population and the initial system $\theta(0) = (\theta(0,1),\dots,\theta(0,L))$ includes *L* random particles (θ candidates).

Secondly the selection procedure is applied: particles $\widehat{\theta(p)} = \big(\theta\widehat{(p,1)},\dots,\theta\widehat{(p,L)}\big)$ are chosen independently according to Gibbs measure $Gibb\big(\theta(p,a)\big)$, where the probability of selection of a^{th} individual equals $P\left(\theta\widehat{(p,a)} = \theta(p,a)\middle|\theta(p)\right) = Gibb\big(\theta(p,a)\big)$, and the probability of its replacement by another individual equals:

$$P\left(\theta\widehat{(p,a)} = \theta(p,j)\middle|\theta(p)\right) = \frac{Gibb\big(\theta(p,j)\big)}{\sum_{i=1}^{L}\theta(p,i)}, \quad (7)$$

where Gibbs measure: $Gibb\big(\theta(p,a)\big) = e^{\frac{-G(\theta(p,a))}{Tmp}}$, $G\big(\theta(p,a)\big)$ – fitness function, and Tmp – temperature parameter, that decreases progressively with every iteration.

Then the mutation procedure is applied:

$$\theta(p+1,a) = \theta(p,a) + s(2rand() - 1), \quad (8)$$

where $rand()$ – random number between 0 and 1, s – mutation coefficient.

Finally stopping criteria is verified: one of the individuals $\theta(p,a)$ if put into fitness function leads to its satisfying value. If not, we repeat all the steps.

We repeat the evolutionary procedure for all X calibration dates and obtain optimal $\theta(p,a)$ and by consequence from (4) local volatility matrix $\sigma\big(t,\gamma,\theta(p,a)\big)_x$ for each calibration date *x*.

Each local volatility value $\sigma\big(t,\gamma,\theta(p,a)\big)_x$ corresponds to a coordinate on calibration date, maturity and strike grids x, t, γ respectively. Hence we obtain X local volatility matrices of $N \times M$ dimensionality. Because at-the-money values correspond to the most traded options and therefore are the most representative and also to reduce the dimensionality of the problem as we do in [4] we extract only ATM local volatilities. Thus, we get rid of strike axis by taking the only value (ATM) and obtain $N \times X$ matrix of ATM local volatilities. Thus our term structure corresponds to calibration dates from Tables 1 and 2.

We also extract $N \times X$ realized volatility matrix as relative deviation of the value of Microsoft stock S between each calibration date *x* and maturity T_i calculated from this calibration date as done in [14]:

$$RealizedVol_{T_i}(x) = \frac{S_{x+T_i} - S_x}{S_x}, \quad i = \overline{1,N}, x = \overline{1,X}. \tag{9}$$

We firstly build univariate model of relation between realized and local volatilities:

$$\widehat{RealizedVol}_{T_i}(x) = a_{T_i} + b_{T_i} * LocalVolATM_{T_i}(x), \tag{10}$$
$$\mathrm{i} = \overline{1,N}, \quad x = \overline{1,X},$$

where T_i corresponds to the maturities grid (1), and *x* – to one of X calibration dates.

Then we build multivariate model basing on the research of Luo and Zhang who used 6- and 9-month forward variances in a multivariate model to predict the return of the underlying stock over 1-month period [14]. This research inspired us to model realized volatility at extreme maturities T_1 = 0.05 and T_6 = 0.4 (as fractions of the year) with multivariate linear regression where the predictors are local volatility values at all other given maturities:

$$\widehat{RealizedVol}_{T_i}(x) = a_{T_i} + \sum_{j=1}^{N} b_{T_{ij}} * LocalVolATM_{T_{ij}}(x), \quad (11)$$

$$i = 1, N, \ \ x = \overline{1, X}.$$

The models are analyzed according to the following steps.

Firstly the assumptions of the regression model are verified using graphical analysis of distribution plots of residuals.

Secondly R^2, *t*- and *F*-statistics, and out-of-sample *MAE* (mean absolute error) are used to evaluate the quality of regression models. Durbin-Watson test is conducted to determine the presence of autocorrelation (may be caused by overlapping) in the residuals.

$$MAE_{T_i} = \frac{1}{V}\sum_{v=1}^{V} \frac{\left|RealizedVol_{T_i}(v) - \widehat{RealizedVol}_{T_i}(v)\right|}{RealizedVol_{T_i}(v)}, \quad (12)$$

where V – number of validation points (out-of-sample) on maturity axis.

Finally the models are validated on two different sets of calibration/validation dates. We use *MAE* measure (12) and analysis of plots to measure the quality of predictions and validate obtained results. Cross-validation shows if the model can successfully predict the realized volatility at intermediary OOS dates (Table 1) while predictive validation uses the realized volatility at extrapolated OOS calibration dates (Table 2).

Numerical results

ANOVA (analysis of variance) is conducted to verify the assumptions necessary to use the linear regression (Fig. 1). The example of ANOVA at $T_6 = 0.4$ from Fig. 1 demonstrates that the residuals lay close to the normal distribution which means that linear regression model can be used for both univariate and multivariate cases for both cross-validation and predictive datasets. The same assumption is successfully verified for other maturities (not illustrated here for the sake of compactness).

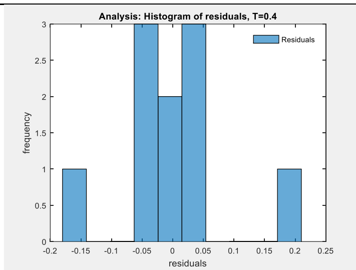
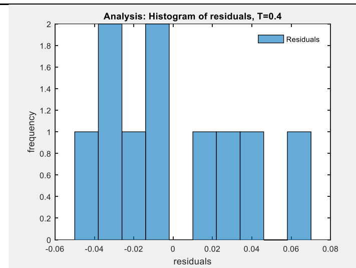
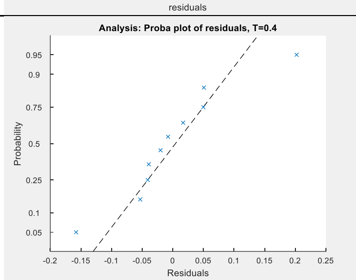
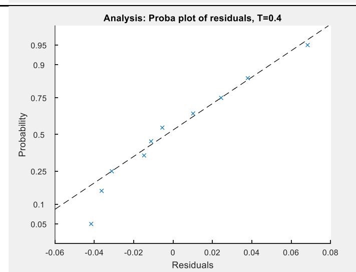
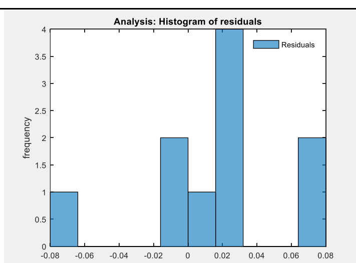
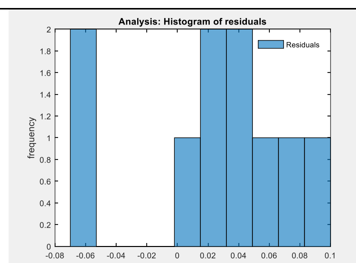
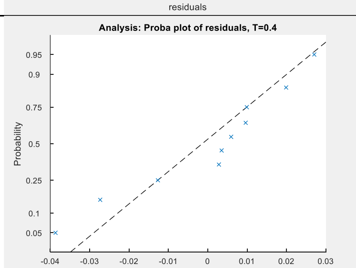
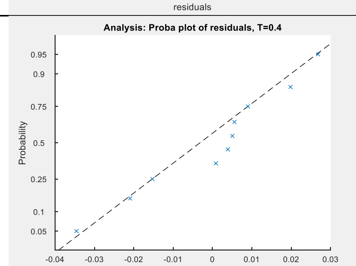
<table>
<tr><th>Cross-validation set</th><th>Predictive validation set</th></tr>
<tr><td colspan="2">UNIVARIATE MODEL</td></tr>
<tr><td>
</td><td>
</td></tr>
<tr><td>
</td><td>
</td></tr>
<tr><td colspan="2">MULTIVARIATE MODEL</td></tr>
<tr><td>
</td><td>
</td></tr>
<tr><td>
</td><td>
</td></tr>
</table>

Fig. 1. ANOVA for cross-validation set and predicitve set at $T_6 = 0.4$

Univariate model.

Let's now take a look at the graphical results of univariate modeling on the example of maturity $T_1 = 0.05$. For cross-validation set Fig. 2 illustrates the dynamics of calibrated realized volatility for the maturitiy $T_1 = 0.05$ as well as the regression plots (modeled realized volatility w.r.t. local volatility) that we have built. From Fig. 2 we can also stipulate that 1st order univariate regression fit the data quite well at all cross-validation points. It does not fit well only the outliers that occur during the crisis (realized volatility peaks). We observe quite the same situation for other maturities but we don't illustrate it in this article for the sake of compactness.

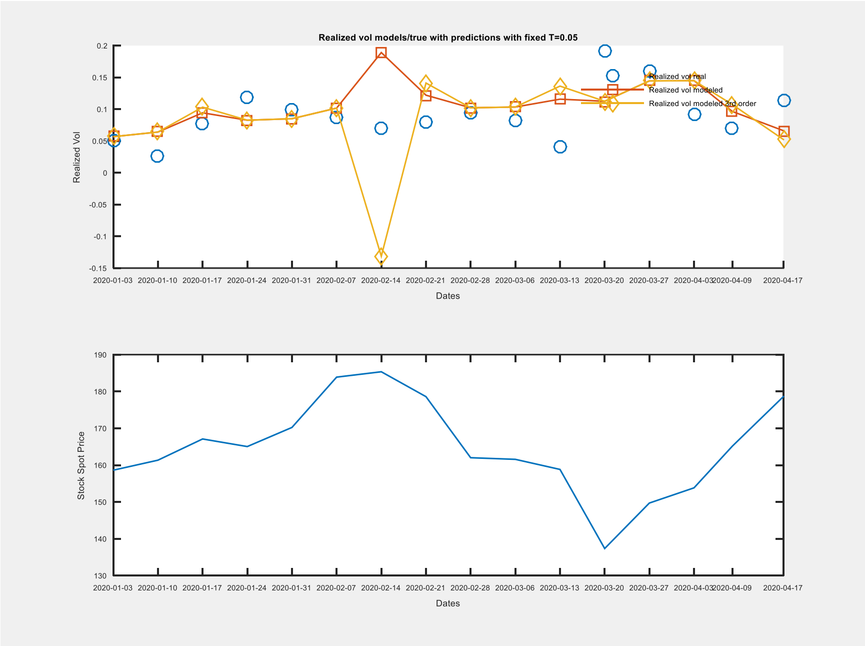


Fig. 2. Real and modeled realized volatility with univariate model at $T_1 = 0.05$ time period for cross-validation dataset

Subplots below main graphs in Fig. 2 illustrate the dymamics of underlying MSFT stock price.

Now let's study the numerical results of 1st order univariate regression modeling given in Table 3. It illustrates the statistics for both cross-validation and predictive sets. The analysis is done for both runs of calibration to reflect the random nature of the genetic algorithm used for the calibration.

Table 3

STATISTICS OF THE 1ST ORDER UNIVARIATE LINEAR REGRESSION FOR CROSS-VALIDATION AND PREDICTIVE SETS

T_i	R^2	*t*-statistic	*p*-value	*F*-statistic	*DW*	*MAE* (OOS)
CROSS-VALIDATION SET						
0.05	0.176	1.306	0.227	1.710	1.858	0.550
0.1	0.149	1.186	0.270	1.410	1.892	2.370
0.15	0.396	2.290	0.051	5.240	2.112	26.712
0.2	0.221	1.506	0.171	2.270	1.155	2.560
0.25	0.532	3.016	0.017	9.090	1.461	0.388
0.4	0.389	2.255	0.054	5.080	1.432	3.501
PREDICTIVE SET						
0.05	0.355	2.096	0.069	4.390	1.363	2.993
0.1	0.176	-1.308	0.227	1.710	1.005	0.767
0.15	0.001	0.094	0.928	0.008	1.061	0.569
0.2	0.030	0.495	0.634	0.245	1.085	0.555
0.25	0.625	3.649	0.007	13.300	0.898	0.596
0.4	0.537	3.043	0.016	9.260	2.494	3.419

Let's firstly analyze auto-correlation Durbin-Watson criteria. From Table 3 we can see that no autocorrelation tendencies at cross-validation dataset are present as *p*-value of Durbin-Watson coefficient is mostly much larger than 0.1 significance level except a few maturities. For the predictive set values are far more alarming. So, we can stand that autocorrelation is not a problematic factor if we base our model on cross-validation dataset but can play its role on predictive dataset. However, getting rid of such autocorrelation is not in the scope of this work.

Let's analyze R^2, *t*- and *F*-statistics, *MAE* from the Table 3. Here and below, *MAE* is calculated on the validation data (out-of-sample), and all other statistical characteristics are calculated on the calibration sample.

Firstly we take a look at the cross-validation set (Table 1). The value of R^2 falls approximately between 0.15 and 0.53. We notice that the fit is better for longer maturities. These values are not impressive however R^2 cannot be used alone to make any meaningful conclusions. In terms of *t*- and *F*-statistics the predictor proves to be insignificant for maturities $T_1 = 0.05$, $T_2 = 0.1$ and $T_4 = 0.2$, and its *p*-value is around 0.2 in average which is superior than 0.1 two-sided significance level. However, some longer maturities, i.e. $T_3 = 0.15$, $T_5 = 0.25$ and $T_6 = 0.4$, have *p*-value lower than 0.1 that tell us that local volatility at these maturities is a significant factor in the model.

From *MAE* we see that the fit of the model on validation points is quite bad as *MAE* values vary between 0.39 and 26,7. This can be explained by the fact that cross-validation set includes validation points that match deliberately high volatility periods. Nevertheless we can state that this model is useless for crisis predictions.

Secondly we take a look at the predictive set (Table 2). The value of R^2 falls approximately between 0.001 and 0.63. The fit is better for longer maturities again. In terms of *t*- and *F*-statistics the predictor proves to be insignificant for maturities $T_2 = 0.1$, $T_3 = 0.15$ and $T_4 = 0.2$, and its *p*-value is around 0.2 in average which is superior than 0.1 two-sided significance level. On the other hand, some other maturities, i.e. $T_1 = 0.05$, $T_5 = 0.25$ and $T_6 = 0.4$, have *p*-value lower than 0.1 that tell us that local volatility at these maturities is a significant factor in the model. *MAE* demonstrates that the fit of the model on validation points is quite bad as *MAE* values vary between 0.56 and 3.42.

We conclude that only local volatility at some longer maturities proves to be a significant predictor of the realized volatility at the corresponding time interval. However, for both cross-validation and predictive datasets the 1st order linear regression does not prove to predict well the out-of-the-sample values of the realized volatility.

Let's now see if multivariate linear model can improve the predictions for some maturities.

Multivariate model.

Let's now take a look at the graphical results of multivariate modeling. For cross-validation set Figs. 3, 4 illustrate the dynamics of calibrated realized volatility for the maturities $T_1 = 0.05$ and $T_6 = 0.4$ respectively as well as the regression plots (modeled realized volatility w.r.t. local volatility) that we have built (1st order). Figs. 3, 4 prove that multivariate model can fit the data quite well at cross-validation points (Table 1).

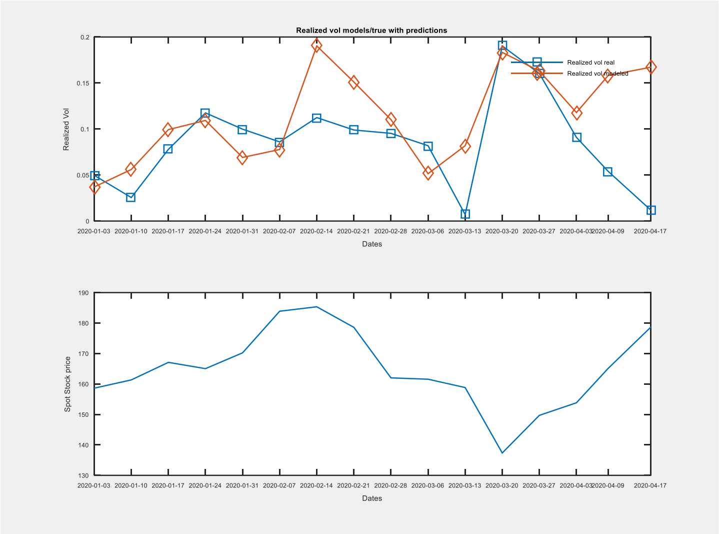


Fig. 3. Real and modeled realized volatility with multivariate model at T_1 = 0.05 time period for cross-validation dataset

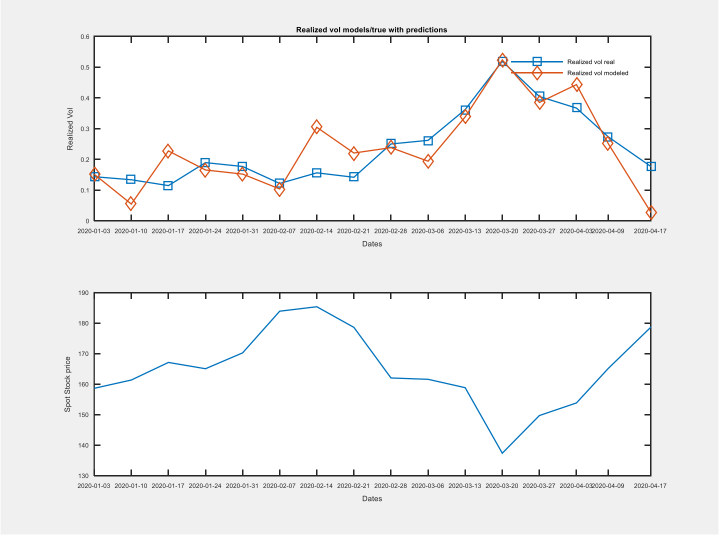


Fig. 4. Real and modeled realized volatility with multivariate model at T_6 = 0.4 time period for cross-validation dataset

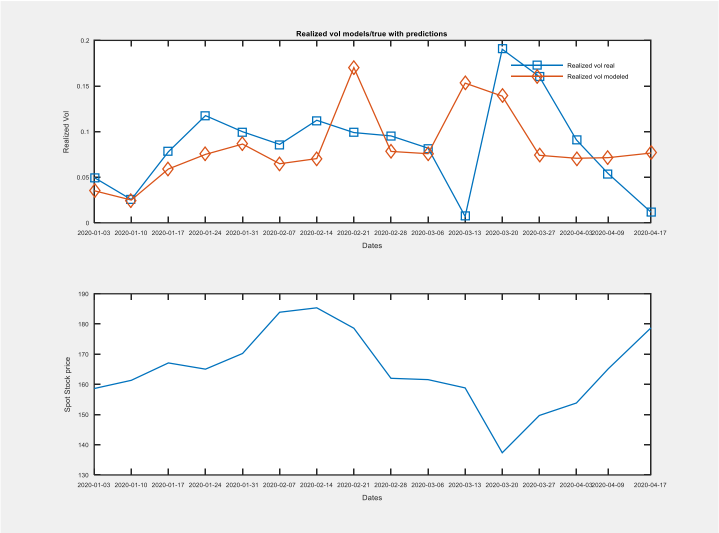


Fig. 5. Real and modeled realized volatility with multivariate model at T_1 = 0.05 time period for predictive dataset

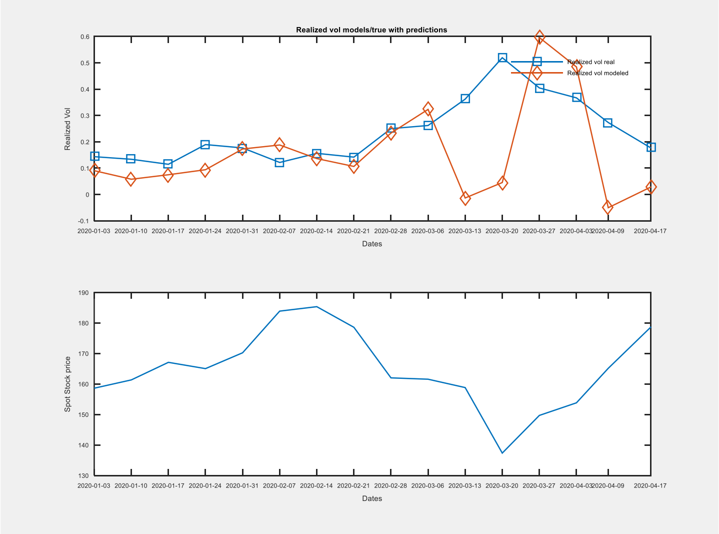


Fig. 6. Real and modeled realized volatility with multivariate model at T_6 = 0.4 time period for predictive dataset

Table 4

STATISTICS OF THE 1ST ORDER MULTIVARIATE LINEAR REGRESSION FOR CROSS-VALIDATION AND PREDICTIVE SETS

Ti	**Factor**	***R*²**	***t*-statistic**	***p*-value**	***MAE* (OOS)**
CROSS-VALIDATION SET					
0.05	$LocalVolATM_{T_1=0,05}$	0.971	-1.336	0.274	2.904
	$LocalVolATM_{T_2=0,1}$		0.055	0.960	
	$LocalVolATM_{T_3=0,15}$		1.747	0.179	
	$LocalVolATM_{T_4=0,2}$		0.058	0.958	
	$LocalVolATM_{T_5=0,25}$		-4.260	0.024	
	$LocalVolATM_{T_6=0,4}$		5.460	0.012	
0.4	$LocalVolATM_{T_1=0,05}$	0.980	1.051	0.371	0.993
	$LocalVolATM_{T_2=0,1}$		-0.276	0.800	
	$LocalVolATM_{T_3=0,15}$		-3.590	0.037	
	$LocalVolATM_{T_4=0,2}$		1.608	0.206	
	$LocalVolATM_{T_5=0,25}$		3.788	0.032	
	$LocalVolATM_{T_6=0,4}$		-2.715	0.073	
PREDICTIVE SET					
0.05	$LocalVolATM_{T_1=0,05}$	0.809	-1.477	0.236	4.054
	$LocalVolATM_{T_2=0,1}$		1.217	0.311	
	$LocalVolATM_{T_3=0,15}$		-0.404	0.713	
	$LocalVolATM_{T_4=0,2}$		-2.339	0.101	
	$LocalVolATM_{T_5=0,25}$		0.408	0.711	
	$LocalVolATM_{T_6=0,4}$		1.851	0.161	
0.4	$LocalVolATM_{T_1=0,05}$	0.810	1.016	0.385	1.260
	$LocalVolATM_{T_2=0,1}$		-1.164	0.329	
	$LocalVolATM_{T_3=0,15}$		1.380	0.261	
	$LocalVolATM_{T_4=0,2}$		1.595	0.209	
	$LocalVolATM_{T_5=0,25}$		-1.999	0.139	
	$LocalVolATM_{T_6=0,4}$		1.512	0.228	

On predictive dataset (Table 2), we see from Figs. 5, 6 that the multivariate regression line falls quite close to the data points. Therefore, there is an apparent potential of the model to predict well further volatility.

Table 4 illustrates the statistics of the 1st order multivariate regression for both cross-validation and predictive sets for maturities $T_1 = 0.05$ and $T_6 = 0.4$. The analysis is done for both runs of calibration to reflect the random nature of the genetic algorithm used for the calibration.

Let's analyze R^2, *t*-statistics and *MAE* from the Table 4. Here, as for the univariate model, the *MAE* is calculated on the validation data, and all other statistical characteristics are calculated on the set of calibration points.

Hereafter we fit the values of realized volatility over the periods $T_1 = 0.05$ and $T_6 = 0.4$ with linear model containing the local volatility at all the maturities including 0.05 and 0.4.

First, we take a look at the cross-validation set statistics (Table 1).

Firstly we analyze the run at $T_1 = 0.05$. From Table 4 we see that R^2 equals to 0.971, which means that the model fits the calibration data perfectly and it is supported by our graphical representations (Fig. 4). This can also be explained by the fact that we have 6 factors and only 10 data points. Only local volatility at $T_5 = 0.25$ and $T_6 = 0.4$ (i.e. longer maturities) prove to be significant factors impacting the value of realized volatility as *p*-value is lower than 0.1 two-sided significance level. In terms of *MAE* such model unfortunately gives quite bad prediction as $MAE = 2.904$.

Let's now investigate the run at $T_6 = 0.4$. From Table 4 we see that R^2 equals to 0.980, which means that the model fits the calibration data perfectly which is also proved by our graphical representations (Fig. 3). Only local volatility at $T_3 = 0.15$, $T_5 = 0.25$ and $T_6 = 0.4$ (i.e. longer maturities) prove to be significant factors impacting the value of realized volatility as *p*-value is lower than 0.1 significance level. In terms of *MAE* such model unfortunately gives quite bad prediction as MAE=0.993.

Secondly we investigate the results for the predictive set (Table 2).

We look at the run at $T_1 = 0.05$. From Table 4 we see that R^2 equals to 0.809, which means very good match to the calibration date points (also seen in the Fig. 6). Once again this can be explained by low number of data points. Only local volatility at $T_4 = 0.2$ and

$T_6 = 0.4$ (i.e. longer maturities) are among more significant factors impacting the value of realized volatility as *p*-value is closer but still superior to 0.1 significance level. In terms of *MAE* such model unfortunately gives quite bad prediction as *MAE* = 4.054.

Let's see the run at $T_6 = 0.4$. From Table 4 we see that R^2 equals to 0.810, which means very good fit to the calibration points (Figure 5). The local volatility at $T_4 = 0.2$, $T_5 = 0.25$ and $T_6 = 0.4$ (i.e. longer maturities) are among more significant factors impacting the value of realized volatility as *p*-value is closer to 0.1 significance level but still superior to 0.1. In terms of *MAE* such model unfortunately gives quite bad prediction as *MAE* = 1.260.

Insufficient significance can be explained by the fact that for the predictive dataset the points of high volatility that were previously used as validation points now we used in fitting dataset. It apparently caused worse fit and less significance. But the local volatility at longer maturities is still much more significant than the one at shorter ones.

We conclude that in all models the local volatility at longer maturities (from 0.15 to 0.4) proves to be significant predictor of realized volatility (whether we predict realized volatility in a short time interval or in a longer one). Therefore it makes sense to predict the volatility on the market by calibrating local volatility from the options with longer maturities. Also neither univariate nor multivariate model proves to perform well the prediction of realized volatility on any of two datasets. Apparently using more data with longer non-overlapping periods would possibly improve the fit and even predictive ability of such models.

Conclusions

In this work we apply univariate and multivariate linear regressions to model the relation between at-the-money local volatility and realized volatility. The experimental calibration of local volatility datasets was conducted using the data of option prices of Microsoft stocks.

To handle the ill-posed character of Dupire calibration problem we use genetic algorithm of optimization of Cerf. To obtain two local volatility datasets (regression inputs) two runs of the calibration are executed as we want to reflect the random nature of the genetic algorithm that can give slightly different values of local volatility for different runs.

The models were validated by predicting out-of-sample realized volatility based on local volatility. We also measure the statistical significance of local volatility as a predictor of realized volatility at different maturities.

Firstly, we observed that residuals of uni- and multivariate linear regression models seem to be normally distributed for cross-validation and prediction sets so the assumption to use the uni- and multivariate linear regression are held.

Secondly, we analyzed auto-correlation using Durbin-Watson criteria and concluded that the autocorrelation is not problematic if we base our model on cross-validation dataset but can play its role on predictive dataset for our univariate model. However getting rid of such autocorrelation is not in the scope of this work.

Thirdly, we conclude that in all models the local volatility at longer maturities proves to be significant predictor of realized volatility (whether we predict realized volatility in a short time interval or in a longer one). Therefore, it makes sense to predict the volatility on the market by calibrating local volatility from the options with longer maturities.

Fourth, neither uniuvariate nor multivariate model proves to perform well the prediction of realized volatility on any of two datasets.

Potential improvement could be using more data with longer non-overlapping periods. It would possibly improve the fit and even predictive ability of such models.

References

1. Babenko, V., Panchyshyn, A., Zomchak, L., Nehrey, M., Artym-Drohomyretska, Z., & Lahotskyi, T. (2021). Classical Machine Learning Methods in Economics Research: Macro and Micro Level Examples. *WSEAS Transactions on Business and Economics, 18*, 209-217. https://doi.org/10.37394/23207.2021.18.22

2. Ben Hamida, S., & Cont, R. (2005). Recovering volatility from option prices by evolutionary optimization. *Journal of Computational Finance, 8*(4), 43-76. https://doi.org/10.21314/JCF.2005.130

3. Black, F. S., & Scholes, M. S. (1973). The Pricing of Options and Corporate Liabilities. *Journal of Political Economy, 81*(3), 637-654.

4. Bondarenko, M., & Bondarenko, V. (2018). Calibration of Dupire local volatility model using genetic algorithm of optimization. *Neiro-Nechitki Tekhnolohii Modelyuvannya v Ekonomitsi (Neuro-Fuzzy Modeling Techniques in Economics)*, *7*, 3-33. https://doi.org/10.33111/nfmte.2018.003

5. Bondarenko, M., & Bondarenko, V. (2021). Modeling time series of ATM local volatility for Microsoft stocks. *SWorldJournal*, *7*(2), 105-116. https://doi.org/10.30888/2663-5712.2021-07-02-084

6. Burtnyak, I. (2021). Models of using real options to make investment decisions. In V. Ponomarenko, T. Klebanova, & L. Guryanova (Eds.), *Systems analysis models in the economic processes management* (pp. 185-193). VSEM – Simon Kuznets KhNEU. http://repository.hneu.edu.ua/handle/123456789/25931

7. Cerf, R. (1998). Asymptotic convergence of genetic algorithms. *Advances in Applied Probability*, *30*(2), 521–550. https://doi.org/10.1239/aap/1035228082

8. Christensen, B.J., & Prabhala, N.R. (1998). The relation between implied and realized volatility. *Journal of Financial Economics*, *50*(2), 125—150. http://doi.org/10.1016/S0304-405X(98)00034-8

9. Derbentsev, V., Matviychuk, A., & Soloviev, V.N. (2020). Forecasting of Cryptocurrency Prices Using Machine Learning. In L. Pichl, C. Eom, E. Scalas, & T. Kaizoji (Eds.), *Advanced Studies of Financial Technologies and Cryptocurrency Markets* (pp. 211-231). Springer. https://doi.org/10.1007/978-981-15-4498-9_12

10. Derbentsev, V., Matviychuk, A., Datsenko, N., Bezkorovainyi, V., & Azaryan, A. (2020). Machine learning approaches for financial time series forecasting. *CEUR Workshop Proceedings*, *2713*, 434–450. http://ceur-ws.org/Vol-2713/paper47.pdf

11. Dupire, B. (1994). *Pricing with a smile*. *Risk*, *7*(1), 18-20.

12. IVolatility. (2020). *Historical Options Data* [Data set]. Retrieved May 1, 2020, from https://www.ivolatility.com/data/historical_data2.html

13. Kaminskyi, A., & Nehrey, M. (2021). Passing through COVID-19 financial shock by Artificial Intelligence ETFs: changes in risk-return correspondence. In V. Ponomarenko, T. Klebanova, & L. Guryanova (Eds.), *Systems analysis models in the economic processes management* (pp. 276-289). VSEM – Simon Kuznets KhNEU. http://repository.hneu.edu.ua/handle/123456789/25931

14. Luo, X., & Zhang, J. E. (2017). Expected Stock Returns and Forward Variance. *Journal of Financial Markets*, *34*, 95-117. https://doi.org/10.1016/j.finmar.2016.06.001

The article was submitted on 2021, May 1