Monte Carlo Simulation using the Benson-Zangari Approach   March 14, 2013

Contents

1   Introduction

The Benson-Zangari approach [1] is a method for Monte Carlo simulation of risk factors that has a number of practical advantages over standard approaches. In particular, it does not require that the simulated factors are linearly independent, it does not depend on the ordering of factors, and is computationally simple.

This document describes how Benson-Zangari is used in RiskServer. Section 2 describes the Benson-Zangari approach. Section 3 discusses different assumptions on the drift associated with risk factors. Section 4 discusses how to apply Benson-Zangari in such a way as to minimize day-to-day variation in Monte Carlo simulation.[1]

---

[1]This approach is available with RiskServer 5.4 Phase II.

2 Benson-Zangari

To describe the original algorithm, let $f_1, \ldots, f_k$ be factors (financial returns[2]) and assume that we have n historical observations $f_j(1), \ldots, f_j(n)$ of the latter; the indices $i = 1, \ldots, n$ stand for increasing time points, with $f_j(n)$ the most recent observation. We suppose that the historical covariance matrix $\mathbf{C}$ is estimated through EWMA with a decay parameter $0 < \lambda \leq 1$.

To simulate normal factors with zero mean and covariance $\mathbf{C}$, one first forms a matrix of historical data as follows:

$$\mathbf{R} = \sqrt{\frac{1-\lambda}{1-\lambda^n}} \begin{pmatrix} f_1(n) & f_2(n) & \cdots & f_k(n) \\ \lambda^{1/2} f_1(n-1) & \lambda^{1/2} f_2(n-1) & \cdots & \lambda^{1/2} f_k(n-1) \\ \vdots & \vdots & \cdots & \vdots \\ \lambda^{(n-1)/2} f_1(1) & \lambda^{(n-1)/2} f_2(1) & \cdots & \lambda^{(n-1)/2} f_k(1) \end{pmatrix}. \tag{1}$$

The case $\lambda = 1$ corresponds to usual sampling with equal weights, and it can recovered by taking the limit $\lim_{\lambda \to 1} (1-\lambda)/(1-\lambda^n) = 1/n$. It is easy to verify that $\mathbf{C} = \mathbf{R}'\mathbf{R}$ (where a prime in this document denotes the transpose of a matrix or vector). Indeed,

$$(\mathbf{R}'\mathbf{R})_{ij} = \frac{1-\lambda}{1-\lambda^n} \sum_{k=0}^{n-1} \lambda^k f_i(n-k) f_j(n-k), \tag{2}$$

which is the EWMA estimator of the covariance between factors f_i and f_j.

Let $\mathbf{Z} = (Z_1, \ldots, Z_n)'$ be a vector consisting of iid (independent identically distributed) standard normal random variables. Then

$$\mathbf{Y} = \mathbf{R}'\mathbf{Z} \tag{3}$$

yields a vector of k returns such that

$$\mathbb{E}(\mathbf{Y}) = \mathbf{0} \tag{4}$$

$$\mathbb{E}(\mathbf{Y}\mathbf{Y}') = \mathrm{Cov}(\mathbf{Y}) = \mathbf{R}'\mathrm{Cov}(\mathbf{Z})\mathbf{R} = \mathbf{R}'\mathbf{I}_{n \times n}\mathbf{R} = \mathbf{C}, \tag{5}$$

i.e. $\mathbf{Y}$ is a simulated vector of returns with desired distributional properties. Summarizing, to create one joint factor scenario, we

1. Determine the data matrix $\mathbf{R}$,

2. Draw one scenario for the random vector $\mathbf{Z}$,

---

[2]In RiskServer the return type depends on the nature of the factor; the default choice for stocks or volatilities is log-return, i.e. $f_n = \log(p_n/p_{n-1})$, with p the price or volatility level; for interest rates or spreads is difference return, i.e. $f_n = r_n - r_{n-1}$, with r the interest rate level. The raw returns are demeaned if the *demeanReturn* input is specified in the *valuationSpec*, i.e. $f_n = \log(p_n/p_{n-1}) - \bar{f}_n$, with $\bar{f}_n = 1/n \sum_{i=1}^{n} \log(p_n/p_{n-1})$. The default behavior assumes that raw returns are not demeaned.

3. Calculate $\mathbf{Y}$ by the linear transform (3).

If several independent joint factor scenarios have to be generated, one just draws independent scenarios for $\mathbf{Z}$ and applies the latter algorithm.

The Benson-Zangari algorithm has two great advantages. First, it does not require a Cholesky decomposition and works well, in contrast to Cholesky, in the case of a singular covariance matrix $\mathbf{C}$.[3] Second, computing a scenario for a certain factor only requires historical data of that factor. The different historical factor time-series are not mixed since

$$Y_j = \sum_{i=1}^{n} \mathbf{R}_{ij} Z_i = \sqrt{\frac{1-\lambda}{1-\lambda^n}} \sum_{i=1}^{n} \lambda^{(i-1)/2} f_j(n-i+1) Z_i. \tag{6}$$

For the same reason, the factor scenario does not depend on the ordering in the vector $\mathbf{Y}$. In practice it is not necessary to store the data matrix $\mathbf{R}$. A scenario for an additional factor is generated by just performing the calculation (6). Also, caching of factor scenarios is straightforward because earlier generated factor scenarios do not change when new factors are included, provided that always the same $\mathbf{Z}$-scenario was used in (6). Note that the dimension of $\mathbf{Z}$ depends on the length of the data window only and that it is therefore not very costly to store $\mathbf{Z}$ permanently.

In contrast, the Cholesky method requires to rerun a Cholesky decomposition on the covariance matrix each time that a new factor is included in the analysis.

3 Understanding Drifts in Risk Server

The distributional properties of the simulated factor returns defined by Eq. (3), embodied in Eqs. (4,5), are consistent with zero expected returns, and a EWMA estimate of the covariance obtained using n historical factor returns. The number n depends on the lookback period over which historical returns are defined, the sampling return horizon, and the overlapping properties of the historical returns, which are inputs controlled by the user as part of the *risk setting* portion of the input (the *valuationSpec* section of the RML input query).

In this document we denote by h_s the sampling return horizon, and by h_a the analysis horizon corresponding to the desired forecast period. As a concrete example, we might be interested in forecasting risk over an analysis horizon of one month ($h_a = 1M$). We might do that by using historical daily returns ($h_s = 1D$), by using weekly returns ($h_s = 5D$) defined with or without overlap, or returns compounded over a different user specified return horizon. The simulated returns defined by Eq. (3) correspond to return period defined by h_s.

---

[3]The matrix will be singular if the number of factors is greater than n, the number of historical observations available. But even if the matrix is not singular but the number of factors is large, using the Cholesky decomposition can be challenging due to almost degenerate nature of the resulting matrix: with numerical errors the matrix might well be singular for all practical purposes.

The choice of whether to incorporate drifts, when simulating factor returns, is left to the user via the specification of a *drift formula*, currently specified in the *volatility setting* portion of the input (the *monteCarloSpec* section of the RML input query). Current available choices are *noDrift*, *noDriftPandL*, and *inSampleDrift*.

Since historical estimates of factor drifts are notoriously noisy, the Risk Server default choice is to use zero drift (the *noDrift* case), i.e. to apply Eq. (3) to generate a sample return corresponding to h_s. In the more general case when we have a vector of drifts d and we want $\mathbb{E}(\mathbf{Y}) = d$, the equation becomes:

$$\mathbf{Y} = \mathbf{R}'\mathbf{Z} + d \tag{7}$$

If the choice *inSampleDrift* is specified by the user, the k-dimensional vector d is estimated as[4]

$$d = \frac{1}{n}\sum_{i=1}^{n} f(i). \tag{8}$$

If the analysis horizon and the sampling horizon do not match, the simulated returns corresponding to h_a are derived from Eq. (7) by simple scaling, i.e.

$$\mathbf{Y}_a = \sqrt{\frac{h_a}{h_s}}\mathbf{R}'\mathbf{Z} + \frac{h_a}{h_s}d \tag{9}$$

Let's now focus on some consequences of Eq. (9) in the case of zero drift, i.e. let's consider what happens if the user chooses to generate returns with zero drift in Monte Carlo simulation, i.e. if we set $d = 0$ in Eq. (9). It is important to realize that, in this case, we are enforcing a zero drift condition on the specific return defined by the return type associated to the factor f_j. If f^E represents the log-normal return of an equity with price level S, i.e. $f^E(i) = \log(S_i/S_{i-1})$, the equation guarantees that the expected value of the log-return is zero ($\mathbb{E}(f^E) = 0$). This does not however guarantee that the expected value of the P&L (or of the simple return) of the underlying equity is zero. The reason is that

$$\mathbb{E}(\text{P\&L}) = \mathbb{E}(S_{i+1} - S_n) = S_n\mathbb{E}(\exp(f^E) - 1) = S_n[\exp(\sigma_E^2/2) - 1] \neq 0, \tag{10}$$

where σ_E is the estimate of the volatility of the equity factor.

Similar considerations apply to rate factors. If f^r represents the difference return associated to a zero coupon rate factor corresponding to tenor T, $\mathbb{E}(f^r) = 0$. However, the expected return of the corresponding discount bond is given by:

$$\mathbb{E}(\text{P\&L}) = \mathbb{E}(D_{n+1}^T - D_n^T) = D_n^T\mathbb{E}(\exp(f^r) - 1) = D_n^T[\exp(\sigma_{r,d}^2/2) - 1], \tag{11}$$

---

[4]If the *demeanReturn* choice is made, raw (i.e. not demeaned) returns are used in Eq. (8).

where $\sigma_{r,d}$ is the estimate of the volatility of the rate factor; in the case factor returns are defined as

$$f^r(i) = \log(D_i/D_{i-1}) = -T(r_i - r_{i-1}).\tag{12}$$

Rate factors can also be defined to be of type log-return, i.e. $f^r_{ln}(i) = \log(r_i/r_{i-1})$. In this case expectation of the P&L of the corresponding discount factor is given by

$$
\begin{aligned}
\mathbb{E}(\text{P\&L}) = \mathbb{E}(D^T_{n+1} - D^T_n) &= D^T_n \left\{ \mathbb{E}(\exp[-T(r_{n+1} - r_n)]) - 1 \right\} \\
&= D^T_n \left\{ \mathbb{E}(\exp[-\frac{T}{r_n}(\exp(f^r_{ln}(n)) - 1)]) - 1 \right\} \\
&\approx D^T_n \left\{ \mathbb{E}(\exp[-\frac{T}{r_n} f^r_{ln}(n)]) - 1 \right\} \\
&= D^T_n \left(\exp[(\frac{T\sigma_{r,ln}}{r_n})^2/2] - 1 \right)
\end{aligned}\tag{13}
$$

where $\sigma_{r,ln}$ is the volatility of the factor and we assume that $f^r_{ln} \ll 1$.

The value of σ in the above equations are computed according to the specification (decay factor, sampling frequency, overlap, and lookback period) specified in the (volatility) risk settings. Following the notation used above, for asset j:

$$\sigma^2_j = \frac{1-\lambda}{1-\lambda^n} \sum_{i=1}^n \lambda^{(i-1)} f^2_j(n-i+1)\tag{14}$$

The three equations above and their derivation point to what needs to be done to have $\mathbb{E}(\text{P\&L}) = 0$ in the case of log-normal price and difference returns rate factors, and $\mathbb{E}(\text{P\&L}) \approx 0$ in the case of log-normal rate factors. This can be obtained by selecting the *noDriftPandL* case, as the value in the driftFormula.

We apply a drift term, identical to the Ito's drift term commonly used in stochastic calculus, that cancels the drift caused by the RM definition of the factor. This is achieved by generating Monte Carlo factor returns as

$$\mathbf{Y}_a = \sqrt{\frac{h_a}{h_s}} \mathbf{R}'\mathbf{Z} + \frac{h_a}{h_s} \mathbf{e}\tag{15}$$

with $\mathbf{e}_j = -\sigma^2_j/2$ for any asset j corresponding to log-normal price return or to a difference-return for rates, and $\mathbf{e}_j = \sigma^2_j/2$ for any asset j corresponding to log-normal rate factors.

4 Minimizing Day-to-Day Changes in Monte Carlo VaR

Every Monte Carlo VaR (MCVaR) estimate has an associated random error, because we can only generate a finite number of simulations. If a new set of random numbers is drawn, the sign and magnitude of this error can change. This effect can lead to random changes to MCVaR estimates from one day to the next. In

addition, MCVaR can change due to changing market conditions, which lead to changing risk estimates. In order to meaningfully compare MCVaR from one day to the next, it is important to minimize the changes due to random fluctuations. Although this random change would disappear as we add more and more simulations, this is not practical as pricing in each scenario can be computationally demanding.[5]

Here we describe an approach to minimizing random changes in MCVaR that involves re-using the same random draws in a consistent way from one day to the next.[6] Recall that the Benson-Zangari procedure involves multiplying random normal draws with weighted historical returns as in Eq. (3). The way to stabilize MCVaR is to uniquely associate a given normal draw with a given return date, so that the same random draw always multiplies the same historical return.

More concretely, consider a simulated risk factor return $Y_{t,k}$ on day t for factor k:

$$Y_{t,k} = R_{1,k}Z_1 + R_{2,k}Z_2 + \cdots + R_{n,k}Z_n \tag{16}$$

where the indexing is the same as in Eq. (1). If the random draws Z are used in the same way on the next day, we have

$$Y_{t+1,k} = R_{2,k}Z_1 + R_{3,k}Z_2 + \cdots + R_{n+1,k}Z_n \tag{17}$$

In other words, a different draw multiplies each historical return on day $t+1$ compared to t.[7]

Across analysis dates, we associate a given random draw with a given return date as follows. As the estimation window moves forward by a day, one historical return drops out at the beginning, and a new return is added at the end. As this happens, we re-use the first random draw (associated with the return that drops out, $R_{1,k}$) to multiply the new return that has been added ($R_{n+1,k}$). This way, the same random draw continues to multiply all returns that are common to the two MCVaR estimates ($R_{2,k}$ through $R_{n,k}$). That is, we replace Eq. (17) with the following:

$$Y_{t+1,k} = R_{2,k}Z_2 + R_{3,k}Z_3 + \cdots + R_{n+1,k}Z_1 \tag{18}$$

The important point is that for a particular choice of risk settings, draw Z_i corresponds to historical return $R_{j,k}$ regardless of the analysis date; the indexing is arbitrary. This is achieved by indexing all dates in a particular way depending on the *statisticsTerm*. Suppose the *statisticsTerm* is 100 business days. We start by taking a reference date (set arbitrarily to May 6, 1997), and assigning this date the index $i = 1$. Then we move forward, assigning May 7, 1997 the index 2, and so on until we get to 100. Then the indexing loops back to 1, to cover the entire history.

The effect of this stabilization on a sample linear bond/equity portfolio is illustrated in figure 1. Because

---

[5]RiskServer allows a maximum of 5000 simulations.

[6]This is a new approach that is available with RiskServer 5.4 Phase II.

[7]This is the approach currently used in RiskServer. Note that although we are re-using the same random draws, they are not applied in a consistent way. The net effect is as though we were drawing new random numbers on each day.

this is a linear portfolio (the PV depends linearly on the risk factor levels), the limit of MCVaR as random error goes to zero is equal to the Parametric VaR. Parametric VaR therefore provides a lower bound on the day-to-day variation in MCVaR forecasts for linear portfolios, as changes in Parametric VaR are due purely to changing risk estimates. In figure 2 we compare the variation of daily MCVaR changes in the two approaches with that of Parametric VaR. We see that the stable simulation approach leads to substantially lower daily variability. In fact, the effect of stabilization on 1000 simulations is greater than increasing the number of simulations to 5000 without the stabilization.
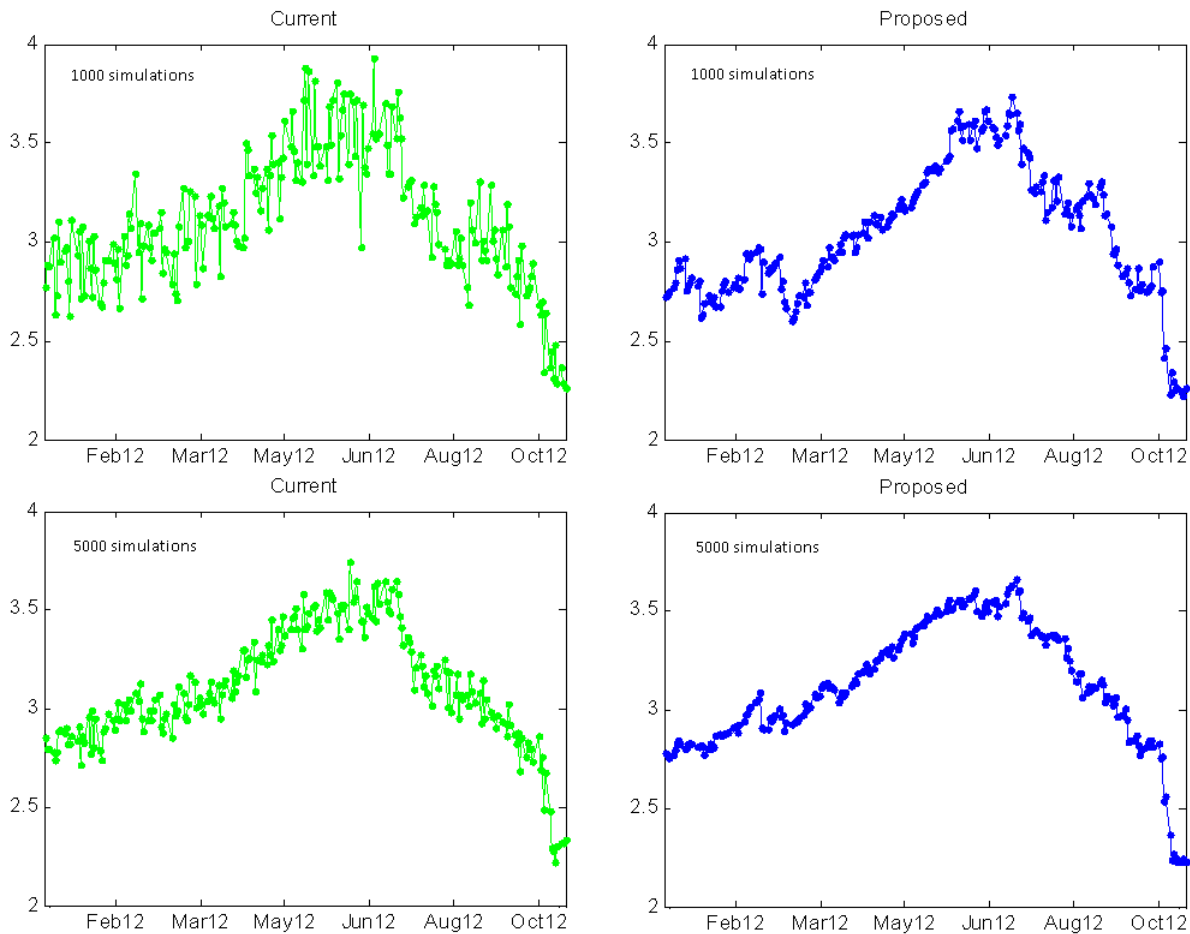

Figure 1: Impact of error stabilization (sample bond/equity portfolio, 1000 and 5000 scenarios)

4.1   Using the Stabilization Procedure in RiskServer

The stabilization procedure described above is made available in RiskServer for historical estimation windows (sometimes referred to as lookback periods) that are specified as a fixed time period (e.g. 1 year). In addition, for the procedure to be effective a fixed number of simulations and consistent historical windows
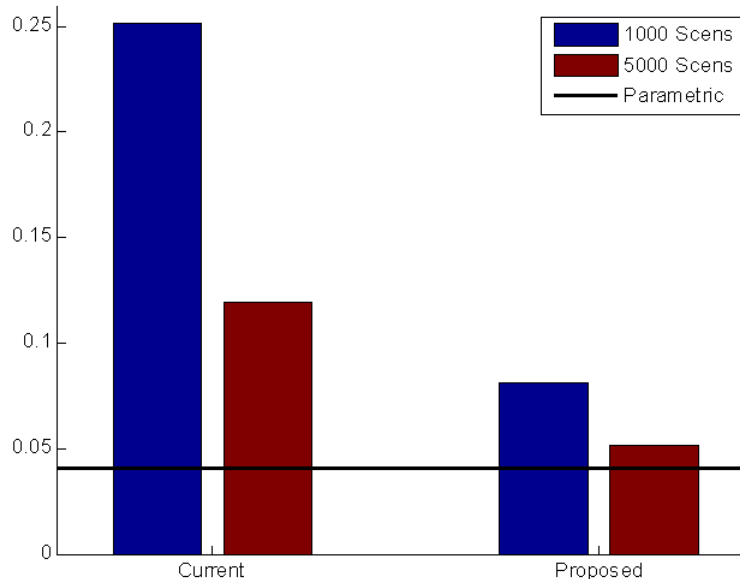
Figure 2: Standard deviation of daily VaR changes (sample bond/equity portfolio, October 2007–October 2012)

must be used across different analysis dates. The new methodology does not apply for arbitrary start and end dates[8]. In other words, the stabilization will not be effective if on one day a 1 year window is used, and on the next a 2 year window is used. Similarly, it will not apply if on one day 1000 simulations are used, and on the next 2000 simulations are used.

Furthermore, it is important to note that the use of multi-day returns along with non-overlapping or partially overlapping returns leads to another, unrelated source of daily variation. Therefore it is important to use the stabilization procedure along with maximally overlapping returns, e.g. for 5-day returns, choose 4 days of overlap.

To enable the new methodology the *monteCarloSeedingMethod→stable* input should be specified in the *valuationSpec* section of the RML input. If *monteCarloSeedingMethod* is not specified, the methodology defaults to the original methodology; this is equivalent to specifying the *monteCarloSeedingMethod→original* input.

---

[8]In the RM4 interface the lookback period input is specified under RiskSettings. The stabilization described here applies if *TimeSeriesDates* is of type *Historical Term*. If no selection is made for this input, stabilization applies because the default choice is one year of trailing lookback period, corresponding to a historical term (or statisticsTerm in a direct RML query) of one year.

References

[1] P. Benson and P. Zangari. A general approach to calculating VaR without volatilities and correlations. *RiskMetrics Monitor*, pages 19–24, 1997.

Client Service Information is Available 24 Hours a Day

clientservice@msci.com

Americas		Europe, Middle East & Africa		Asia Pacific	
Americas	+ 1.888.588.4567 (toll free)	Amsterdam	+ 31.20.462.1382	China North	10800.852.1032 (toll free)
Atlanta	+ 1.404.551.3212	Cape Town	+ 27.21.673.0100	China South	10800.152.1032 (toll free)
Boston	+ 1.617.532.0920	Frankfurt	+ 49.69.133.859.00	Hong Kong	+ 852.2844.9333
Chicago	+ 1.312.675.0545	Geneva	+ 41.22.817.9777	Seoul	+827.0768.88984
Montreal	+ 1.514.847.7506	London	+ 44.20.7618.2222	Singapore	800.852.3749 (toll free)
Monterrey	+ 52.81.1253.4020	Madrid	+ 34.91.700.7275	Sydney	+ 61.2.9033.9333
New York	+ 1.212.804.3901	Milan	+ 39.02.5849.0415	Tokyo	+ 81.3.5226.8222
San Francisco	+ 1.415.836.8800	Paris	0800.91.59.17 (toll free)		
Sao Paulo	+ 55.11.3706.1360	Zurich	+ 41.44.220.9300		
Stamford	+1.203.325.5630				
Toronto	+ 1.416.628.1007				

Notice and Disclaimer

- This document and all of the information contained in it, including without limitation all text, data, graphs, charts (collectively, the "Information") is the property of MSCl Inc., its subsidiaries (including without limitation Barra, Inc. and the RiskMetrics Group, Inc.) and/or their subsidiaries (including without limitation the FEA, ISS, and CFRA companies) (alone or with one or more of them, "MSCI"), or their direct or indirect suppliers or any third party involved in the making or compiling of the Information (collectively (including MSCI), the "MSCI Parties" or individually, an "MSCI Party"), as applicable, and is provided for informational purposes only. The Information may not be reproduced or redisseminated in whole or in part without prior written permission from the applicable MSCI Party.

- The Information may not be used to verify or correct other data, to create indices, risk models or analytics, or in connection with issuing, offering, sponsoring, managing or marketing any securities, portfolios, financial products or other investment vehicles based on, linked to, tracking or otherwise derived from any MSCI products or data.

- Historical data and analysis should not be taken as an indication or guarantee of any future performance, analysis, forecast or prediction.

- None of the Information constitutes an offer to sell (or a solicitation of an offer to buy), or a promotion or recommendation of, any security, financial product or other investment vehicle or any trading strategy, and none of the MSCI Parties endorses, approves or otherwise expresses any opinion regarding any issuer, securities, financial products or instruments or trading strategies. None of the Information, MSCI indices, models or other products or services is intended to constitute investment advice or a recommendation to make (or refrain from making) any kind of investment decision and may not be relied on as such.

- The user of the Information assumes the entire risk of any use it may make or permit to be made of the Information.

- NONE OF THE MSCI PARTIES MAKES ANY EXPRESS OR IMPLIED WARRANTIES OR REPRESENTATIONS WITH RESPECT TO THE INFORMATION (OR THE RESULTS TO BE OBTAINED BY THE USE THEREOF), AND TO THE MAXIMUM EXTENT PERMITTED BY LAW, MSCI, ON ITS BEHALF AND ON THE BEHALF OF EACH MSCI PARTY, HEREBY EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES (INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF ORIGINALITY, ACCURACY, TIMELINESS, NON-INFRINGEMENT, COMPLETENESS, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE) WITH RESPECT TO ANY OF THE INFORMATION.

- Without limiting any of the foregoing and to the maximum extent permitted by law, in no event shall any of the MSCI Parties have any liability regarding any of the Information for any direct, indirect, special, punitive, consequential (including lost profits) or any other damages even if notified of the possibility of such damages. The foregoing shall not exclude or limit any liability that may not by applicable law be excluded or limited, including without limitation (as applicable), any liability for death or personal injury to the extent that such injury results from the negligence or willful default of itself, its servants, agents or sub-contractors.

- Any use of or access to products, services or information of MSCI requires a license from MSCI. MSCI, Barra, RiskMetrics, ISS, CFRA, FEA, EAFE, Aegis, Cosmos, BarraOne, and all other MSCI product names are the trademarks, registered trademarks, or service marks of MSCI in the United States and other jurisdictions. The Global Industry Classification Standard (GICS) was developed by and is the exclusive property of MSCI and Standard & Poor's. "Global Industry Classification Standard (GICS)" is a service mark of MSCI and Standard & Poor's.