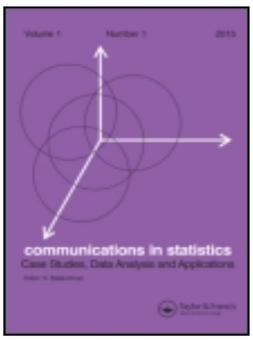

Taylor & Francis
Taylor & Francis Group

Communications in Statistics: Case Studies, Data Analysis and Applications

ISSN: (Print) 2373-7484 (Online) Journal homepage: http://www.tandfonline.com/loi/ucas20

Using arc length to cluster financial time series according to risk

Tharanga Wickramarachchi & Ferebee Tunno

To cite this article: Tharanga Wickramarachchi & Ferebee Tunno (2015) Using arc length to cluster financial time series according to risk, Communications in Statistics: Case Studies, Data Analysis and Applications, 1:4, 217-225, DOI: 10.1080/23737484.2016.1206456

To link to this article: http://dx.doi.org/10.1080/23737484.2016.1206456

Published online: 27 Jul 2016.

COMMUNICATIONS IN STATISTICS: CASE STUDIES, DATA ANALYSIS AND APPLICATIONS
2015, VOL. 1, NO. 4, 217–225
http://dx.doi.org/10.1080/23737484.2016.1206456

Taylor & Francis
Taylor & Francis Group

DATA ANALYSIS

Using arc length to cluster financial time series according to risk

Tharanga Wickramarachchi[a] and Ferebee Tunno[b]

[a]Department of Mathematical Sciences, Georgia Southern University, Statesboro, Georgia, USA; [b]Department of Mathematics and Statistics, Arkansas State University, Jonesboro, Arkansas, USA

ABSTRACT
This article investigates how arc length can be used to partition financial time series according to variability (risk). This technique is predicated on the idea that arc length is an index of volatility, and thus the end result is that safer stocks can be sorted from more risky ones. Performance of arc length is compared with squared returns and absolute returns, two commonly used measures for quantifying the variability of prices. An application involving 30 popular stocks is presented using Maharaj, *k*-means ++, and correlation-based clustering techniques.

ARTICLE HISTORY
Received 11 May 2016
Accepted 23 June 2016

KEYWORDS
Arc length; *k*-means++; Maharaj; volatility

1. Introduction

In finance, there has always been an interest in finding meaningful ways to cluster stocks. One approach is to partition a collection into subsets according to their volatility in order to sort the safer stocks from the more risky ones. This article explores a new way to implement such clustering by utilizing the measure of arc length.

In general, if $\{X_t\}$ is a time series observed at times $t = 1, 2, \ldots, n$, then the sample arc length of this series is the sum of the lengths of the line segments connecting the adjacent sample points $(t, X_t)_{t=1}^n$:

$$\sum_{t=2}^{n} \sqrt{1 + (X_t - X_{t-1})^2}. \tag{1}$$

Similarly, if $\{P_t\}$ is a stock price series with corresponding log price series $\{\ln P_t\}$ observed at times $t = 1, 2, \ldots, n$, then the sample arc length of the log price series is the sum of the lengths of the line segments connecting the adjacent sample points $(t, \ln P_t)_{t=1}^n$:

$$\sum_{t=2}^{n} \sqrt{1 + (\ln P_t - \ln P_{t-1})^2}.$$

It is common practice in finance to look at log prices, as opposed to raw prices, for mathematical convenience. Observe that the log return

$$Y_t = \ln P_t - \ln P_{t-1} = \ln\left(1 + \frac{P_t - P_{t-1}}{P_{t-1}}\right)$$

can often be approximated by $(P_t - P_{t-1})/P_{t-1}$, which is the percent change in price at time t, since $\ln(1 + x) \approx x$ for small x. Returns in general exhibit commonly occurring properties associated with volatility, such as leptokurtosis and persistence.[1] Returns are leptokurtic because their densities tend to have fatter tails than a normal density and are persistent because returns of large/small magnitude tend to be followed by more returns of large/small magnitude. These properties are typically referred to as "stylized facts" and are pervasive in the literature (see Chap. 4 of Taylor (2005)).

Figure 1 shows the log returns of Coca-Cola, Johnson & Johnson, Proctor & Gamble, Alcoa, Eastman Kodak, and Google which correspond to the daily closing prices recorded from January 2005 through December 2007. The picture clearly reveals that the top three series are much less volatile than the bottom three. This article will show that arc length can distinguish between stocks of differing volatility when that difference cannot be discerned upon inspection alone. As a result, investors can make their decisions wisely so that the risk they have to bear is low.

CONTACT Tharanga Wickramarachchi twickramarachchi@georgiasouthern.edu Department of Mathematical Sciences, Georgia Southern University, P.O. Box 8093, Statesboro, GA 30460, USA.

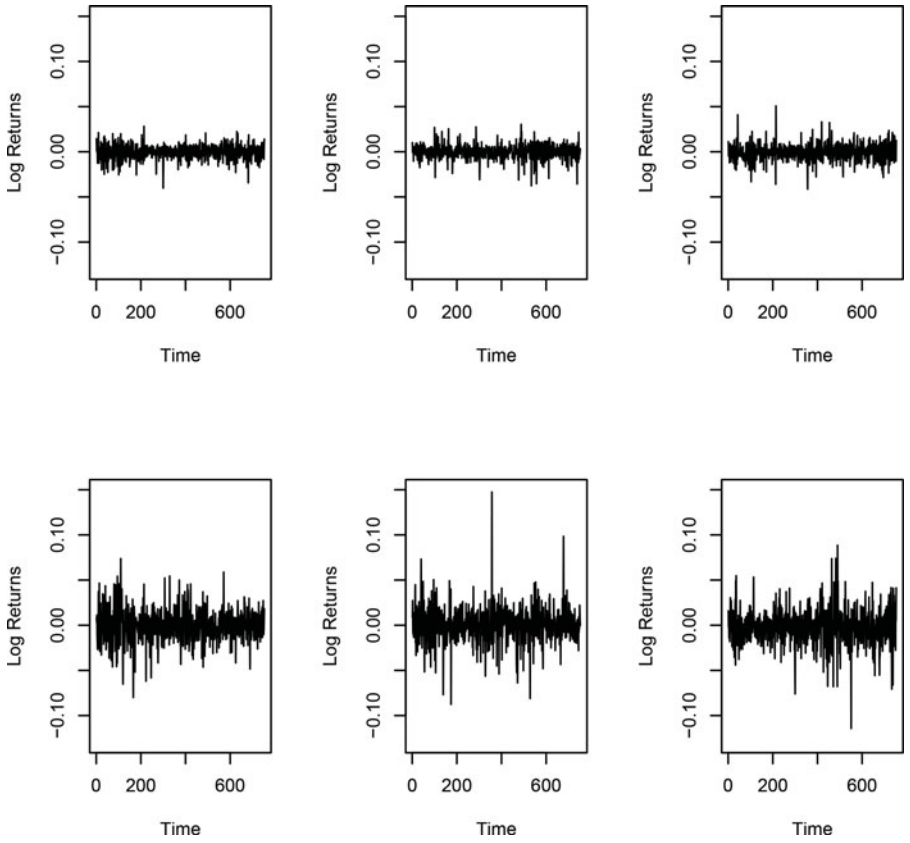

Figure 1. Log returns (clockwise starting at top left): Coca-cola, Johnson & Johnson, Proctor & Gamble, Google, Eastman Kodak, and Alcoa.

Tunno et al. (2012) showed that a standardized arc length test statistic can be used to test for equivalent autocovariances between two independent stationary series. That is, a significant difference in sample arc lengths implies a significant difference in autocovariance structures. If one series has a larger variance, then it should vary more in time and its arc length should be bigger. It was also shown that both arc length and squared differences behave the same when testing for equivalent autocovariances. That is,

$$\sum_{t=2}^{n}(X_t - X_{t-1})^2$$

can be used as a surrogate for Eq. (1) with no effect on the test.

Wickramarachchi et al. (2015), however, have shown that arc length can be used to quantify volatility and in fact outperforms squared returns in this capacity. They prove the functional central limit theorem for arc length under finite second-moment conditions. This was motivated by the fact that most financial time series have finite second moments, but infinite fourth moments, and squared returns require a finite fourth moment for a functional central limit theorem to hold.

It follows from Thm. 2.3 of Wickramarachchi et al. (2015) that the difference between two sample arc lengths has an asymptotic normal distribution. The complete corollary is given below:

Corollary 1.1. *Let* $\{\mathbf{X}_t\} = (X_{1,t}, X_{2,t})^T$ *and* $\{\mathbf{Y}_t\} = (Y_{1,t}, Y_{2,t})^T$ *be bivariate time series such that* $\{\mathbf{Y}_t\}$ *is stationary with* $\mathbf{Y}_t = \mathbf{X}_t - \mathbf{X}_{t-1}$ *and* $\mathrm{E}(\|\mathbf{Y}_t\|^2) < \infty$, *where* $\|\cdot\|$ *is the usual Euclidean norm. Suppose further that* $\{\mathbf{Y}_t\}$ *has component-wise* ϕ*-mixing innovations* $\{\mathbf{Z}_t\}$ *with*

$$\sum_{m=0}^{\infty}\sqrt{\mathrm{E}\left|Y_{i,t}^2 - \left(Y_{i,t}^{(m)}\right)^2\right|} < \infty \qquad i = 1, 2$$

where $\mathbf{Y}_t^{(m)} = g(\mathbf{Z}_t, \mathbf{Z}_{t-1}, \ldots, \mathbf{Z}_{t-m}, \mathbf{0}, \mathbf{0}, \ldots)$ *for some function g. Then, we have*

$$S_n^{(1)} - S_n^{(2)} \xrightarrow{D} \mathrm{N}(0, \tau^2),$$

where $\tau^2 = \mathrm{Var}(\eta_t) + 2\sum_{k=1}^{\infty}\mathrm{Cov}(\eta_0, \eta_k)$, $\eta_t = \sqrt{1+Y_{1,t}^2} - \sqrt{1+Y_{2,t}^2}$,
$\mathrm{S}_{\mathrm{n}}^{(\mathrm{i})} = n^{-1/2}\sum_{1\le t\le n}(\zeta_{i,t} - \mathrm{E}(\zeta_{i,0}))$, *and* $\zeta_{i,t} = \sqrt{1+Y_{i,t}^2}$ *for* $i = 1, 2$.

This result can be used to compare two series statistically in terms of volatility using arc length as the measure. Whichever measure is used, volatility is usually non-constant and could even be a random variable (see Chap. 10 of Dineen (2013)). Certainly, this is the case with a process following, for example, an ARCH or stochastic volatility model.

In this article, we indeed use arc length as the specific measure of volatility to accomplish the goal of clustering stocks according to risk. This approach is an example of what is called feature-based time series clustering (see Liao (2005)), whereby some feature is extracted from the raw data and used in lieu of the data itself. The next section discusses the details of how to use a number of such clustering algorithms, including the Maharaj algorithm which can be driven by a hypothesis test based on Cor. 1.1 to discriminate between two time series. Sec. 3 presents an application along with a comparison of performance between arc length and both squared returns and absolute returns, and Sec. 4 closes the article with some remarks.

2. Clustering algorithms

There are a wide variety of ways to cluster time series data. For an excellent survey of all such techniques, see Liao (2005). In this section, we describe three specific clustering algorithms that will later be used in conjunction with arc length, squared returns, and absolute returns to quantify volatility among a collection of stocks.

2.1. Maharaj algorithm

Maharaj (2000) proposes a hierarchical clustering algorithm based on the p-value of a hypothesis test that assesses whether or not two time series are generated by the same process. According to the algorithm, two time series will go in the same cluster only if the corresponding p-value is greater than a pre-determined level of significance α. More precisely, a new time series will be placed in a given cluster C_k only if all pairwise p-values obtained from tests between the series of interest and the other series in C_k are greater than the selected α. In the next section, we will apply this algorithm using p-values obtained from a hypothesis test based upon the natural dissimilarity measure that follows from Cor. 1.1.

2.2. k-means++ algorithm

The original k-means algorithm for partitioning a numerical data set into k disjoint subsets/clusters was first created by MacQueen (1967) and goes as follows:

1. Choose the number of clusters k for your set S.
2. Randomly partition S into k clusters and determine their centers (averages) or directly generate k random points as cluster centers.
3. Assign each member from S to the nearest cluster, using some pre-chosen distance norm.
4. Recompute the new cluster centers.
5. Repeat Steps 3 and 4 until things stabilize.

The k-means++ algorithm, proposed independently by Ostrovsky et al. (2012) and Arthur and Vassilvitskii (2007), improves upon the regular k-means algorithm by more carefully selecting the initial centers. k-means++ greatly reduces the possibility of suboptimal clustering by substituting the following algorithm in for initial random partition of data points:

1. Choose one center uniformly at random from among the data points.
2. For each data point x, compute the distance $D(x)$ between x and the nearest center that has already been chosen.
3. Add one new data point at random as a new center, using a weighted probability distribution, where point x is chosen with probability proportional to $(D(x))^2$.
4. Repeat Steps 2 and 3 until k distinct centers have been chosen.

2.3. Correlation algorithm

The third clustering technique we consider is a simple hierarchical clustering algorithm that uses a correlation-based distance as the dissimilarity measure. Specifically, this is Pearson's correlation coefficient between two time series $\{U_t\}$ and $\{V_t\}$ given by

$$\rho_{UV} = \frac{\sum_{t=1}^{n} \left(U_t - \overline{U}\right)\left(V_t - \overline{V}\right)}{\sqrt{\sum_{t=1}^{n} \left(U_t - \overline{U}\right)^2}\sqrt{\sum_{t=1}^{n} \left(V_t - \overline{V}\right)^2}},$$

where $\overline{U} = n^{-1}\sum_{t=1}^{n} U_t$ and $\overline{V} = n^{-1}\sum_{t=1}^{n} V_t$.

Golay et al. (1998) propose a cross-correlation-based distance given by $d_{COR}(U_t, V_t) = \sqrt{2\left(1 - COR\left(U_t, V_t\right)\right)}$.

Ticker	Stock	Arc Length	Squared Returns
KO	Coca Cola	753.0226	0.04517905
JNJ	Johnson & Johnson	753.0229	0.04583879
PG	Proctor & Gamble	753.0296	0.05918613
GE	General Electric	753.0340	0.06809023
WMT	Wal-Mart	753.0458	0.09159796
IBM	International Business Machine	753.0469	0.09387548
T	AT&T	753.0490	0.09795123
DD	DuPont	753.0540	0.10803387
DIS	Walt Disney	753.0547	0.10946782
MSFT	Microsoft	753.0577	0.11550568
BP	British Petroleum	753.0589	0.11782078
JPM	JPMorgan Chase	753.0591	0.11826054
MCD	McDonald's	753.0604	0.12073430
HON	Honeywell International	753.0605	0.12100579
C	Citigroup	753.0607	0.12142629
BA	Boeing	753.0692	0.13847253
HD	Home Depot	753.0718	0.14372234
IP	International Paper	753.0726	0.14529586
XOM	Exxon Mobil	753.0738	0.14771585
CVX	Chevron	753.0740	0.14803902
AXP	American Express	753.0773	0.15460847
MRK	Merck	753.0787	0.15750195
INTC	Intel	753.0914	0.18283007
HPQ	Hewlett-Packard	753.0936	0.18741171
AA	Alcoa	753.1150	0.23005622
EK	Eastman Kodak	753.1310	0.26227984
GOOGL	Google	753.1390	0.27821433
UTX	United Technology	753.2607	0.56725511
CAT	Caterpillar	753.3046	0.65125114
AAPL	Apple	753.4317	0.90860994

Figure 2. 30 stocks ordered least to greatest by their log price arc length and squared returns for the period January 2005 through December 2007.

3. Application

Figure 2 provides a list of 30 stocks whose daily closing prices were observed from January 3, 2005 to December 31, 2007 ($n = 754$). Each stock is accompanied by its ticker symbol[2] along with both the log price arc length and log price squared returns for this time period.[3] Note that the order is ascending top-to-bottom for both measures of volatility.

Figure 3 provides a list of the same stocks over the same period, but this time the ticker symbol is accompanied by the log price absolute returns. The order is ascending top-to-bottom in terms of absolute returns but note that this order is slightly different from that of Fig. 2.

3.1. Maharaj algorithm

We applied the Maharaj algorithm twice to these stocks, once setting the level of significance at $\alpha = 0.05$ and then setting it at $\alpha = 0.01$. For simplicity of representation, we write clusters in the form $n_1/n_2/\cdots/n_r$, where cluster 1 contains the first n_1 stocks, cluster 2 contains

Ticker	Stock	Absolute Returns
JNJ	Johnson & Johnson	4.300444
KO	Coca Cola	4.400471
PG	Proctor & Gamble	4.860838
GE	General Electric	5.429393
IBM	International Business Machine	6.062759
WMT	Wal-Mart	6.129222
C	Citigroup	6.302441
MSFT	Microsoft	6.341091
T	AT&T	6.389259
JPM	JPMorgan Chase	6.664406
DIS	Walt Disney	6.796550
DD	DuPont	6.837510
UTX	United Technology	7.050513
AXP	American Express	7.253250
MCD	McDonald's	7.286728
MRK	Merck	7.300584
BP	British Petroleum	7.306878
HON	Honeywell International	7.325252
HD	Home Depot	7.834422
BA	Boeing	7.879072
XOM	Exxon Mobil	8.199893
IP	International Paper	8.201770
HPQ	Hewlett-Packard	8.437870
CVX	Chevron	8.546923
INTC	Intel	8.793151
CAT	Caterpillar	9.457485
AA	Alcoa	9.873937
EK	Eastman Kodak	10.011004
GOOGL	Google	10.396134
AAPL	Apple	14.209773

Figure 3. 30 stocks ordered least to greatest by their log price absolute returns for the period January 2005 through December 2007.

the next n_2 stocks, and so on until finally cluster r contains the last n_r stocks.

At $\alpha = 0.05$, the algorithm identified the same seven clusters for both arc length and squared returns in the form of 2/2/3/8/7/2/6 displayed in Fig. 4. At $\alpha = 0.01$, both arc length and squared returns identified six clusters, which are exactly the same and in the form of 2/2/3/8/9/6 displayed in Fig. 5.

At $\alpha = 0.05$, the algorithm identified nine clusters for absolute returns in the form of 2/1/1/6/8/2/5/4/1

Cluster 1	KO	JNJ						
Cluster 2	PG	GE						
Cluster 3	WMT	IBM	T					
Cluster 4	DD	DIS	MSFT	BP	JPM	MCD	HON	C
Cluster 5	BA	HD	IP	XOM	CVX	AXP	MRK	
Cluster 6	INTC	HPQ						
Cluster 7	AA	EK	GOOGL	UTX	CAT	AAPL		

Figure 4. Clusters for arc length and squared returns based on Maharaj algorithm at $\alpha = 0.05$.

Cluster 1	KO	JNJ						
Cluster 2	PG	GE						
Cluster 3	WMT	IBM	T					
Cluster 4	DD	DIS	MSFT	BP	JPM	MCD	HON	C
Cluster 5	BA	HD	IP	XOM	CVX	AXP	MRK	INTC	HPQ
Cluster 6	AA	EK	GOOGL	UTX	CAT	AAPL			

Figure 5. Clusters for arc length and squared returns based on Maharaj algorithm at $\alpha = 0.01$.

Cluster 1	JNJ	KO						
Cluster 2	PG							
Cluster 3	GE							
Cluster 4	IBM	WMT	C	MSFT	T	JPM		
Cluster 5	DIS	DD	UTX	AXP	MCD	MRK	BP	HON
Cluster 6	HD	BA						
Cluster 7	XOM	IP	HPQ	CVX	INTC			
Cluster 8	CAT	AA	EK	GOOGL				
Cluster 9	AAPL							

Figure 6. Clusters for absolute returns based on Maharaj algorithm at $\alpha = 0.05$.

displayed in Fig. 6. At $\alpha = 0.01$, eight clusters were identified for absolute returns in the form of 3/1/5/9/2/5/4/1 displayed in Fig. 7.

Recalling that most financial series have finite second moments but infinite fourth moments, and also that squared returns require a finite fourth moment for a functional central limit theorem to hold, we now look at two separate simulation studies to assess how the Maharaj algorithm handles clustering when a fourth moment either nearly exists or does not exist at all.

Study 3.1. Define process $\{X_n\}_{n\geq 1}$ such that $X_n = \sigma_n \epsilon_n$, where $\{\epsilon_n\}$ is i.i.d white noise and $\{\sigma_n\}$ is a

Cluster 1	JNJ	KO	PG					
Cluster 2	GE							
Cluster 3	IBM	WMT	C	MSFT	T			
Cluster 4	JPM	DIS	DD	UTX	AXP	MCD	MRK	BP	HON
Cluster 5	HD	BA						
Cluster 6	XOM	IP	HPQ	CVX	INTC			
Cluster 7	CAT	AA	EK	GOOGL				
Cluster 8	AAPL							

Figure 7. Clusters for absolute returns based on Maharaj algorithm at $\alpha = 0.01$.

non-negative volatility process with $\sigma_n^2 = 10^{-6} + \alpha X_{n-1}^2 + \beta\sigma_{n-1}^2$. We then see that $\{X_n\} \sim$ GARCH(1, 1). Four such series, each of length $n = 750$, were then generated from each of the following four cases: (1) $\alpha = 0.1$, $\beta = 0.8$, (2) $\alpha = 0.17$, $\beta = 0.795$, (3) $\alpha = 0.16$, $\beta = 0.81$, and (4) $\alpha = 0.15$, $\beta = 0.82$. (Recalling that finite fourth moments for GARCH(1,1) processes only exist when $\beta^2 + 2\alpha\beta + 3\alpha^2 < 1$, we note that the last three cases yield processes that have nearly infinite fourth moments.)

The Maharaj algorithm was then applied to these sixteen series at the 5% significance level with the correct number of clusters being four. The observed clusters generated from the algorithm were evaluated for accuracy based on the *known ground-truth criteria method* described in Sec. 2.3.1 of Liao (2005). Results show that both arc length and squared returns have an accuracy rate of 64.76% while absolute returns has a rate of 55.06%. This provides further evidence that both arc length and squared returns perform in a similar manner when it comes to clustering based on volatility.

Study 3.2. Define i.i.d process $\{X_n\}_{n\geq 1}$ such that each X_n follows a Pareto distribution with a shape parameter of 3 and a scale parameter of 1. The density function of each X_n then takes the form $f(x_n) = 3/x_n^4$, where $x_n > 1$ for each $n \geq 1$. (Recalling that the rth moment for this particular Pareto process only exists when $r < 3$, we observe that we are dealing here with random variables that have a finite second moment but an infinite fourth moment.) Sixteen such series, each of length $n = 750$, were then generated from processes of the form $\{kX_n\}$, with four each coming the following four cases: (1) $k = 1$, (2) $k = 1.3$, (3) $k = 1.5$, and (4) $k = 1.7$.

The Maharaj algorithm was then applied to these sixteen series at the 5% significance level with the correct number of clusters being four. Once again, the observed clusters generated were evaluated for accuracy based on the aforementioned known ground-truth criteria method. Results show that arc length, squared returns, and absolute returns have accuracy rates of 81.85%, 60.42%, and 80.26%, respectively. As expected, the accuracy rate for squared returns is lower than that of arc length and absolute returns since the hypothesis test does not work as well when the fourth moment is infinite.

Volatility Measure	Scheme	Frequency
Arc Length	16/11/3	613
	22/5/3	295
	27/2/1	92
Squared Returns	16/11/3	540
	22/5/3	360
	27/2/1	100
Absolute Returns	9/16/5	476
	13/16/1	396
	18/11/1	128

Figure 8. Clusters based on the *k*-means++ algorithm with $k = 3$.

3.2. k -means++ algorithm

We next applied the *k*-means++ algorithm to the 30 stocks from earlier using Euclidean distance to partition them into three clusters. The rationale behind the number three is that we wish to identify stocks as having low volatility, medium volatility, or high volatility.

We carried out the algorithm 1, 000 times for each measure of volatility in order to separate the more stable clustering schemes from the less stable ones. The results are given in Fig. 8, where scheme $n_1/n_2/n_3$ means that the first n_1 stocks are low volatility, the next n_2 stocks are medium volatility, and the final n_3 stocks are high volatility. The order of the stocks for arc length and squared returns comes from Fig. 2 while the order of the stocks for absolute returns comes from Fig. 3.

As can be seen, the *k*-means++ algorithm identifies the exact same clusters for both arc length and squared returns with 16/11/3 being the most stable scheme for each of the two volatility measures (it is slightly more stable for arc length since it is correct more often compared to squared returns). On the other hand, the clustering for absolute returns was different since the stock ordering was different, but it is worth noting that none of the three schemes in this case had a stability rate larger than 50%.

3.3. Correlation-based clustering

To begin the correlation-based clustering for the 30 stocks, we first created 753 "pieces" for each stock and for each of the three volatility measures. The series of

	Cluster 1 (low)	Cluster 2 (medium)	Cluster 3 (high)
Arc Length	KO – GOOGL	UTX	CAT, AAPL
Squared Returns	KO – GOOGL	UTX	CAT, AAPL
Absolute Returns	KO – EK	UTX	GOOGL, CAT, AAPL

Figure 9. Clusters based on correlation distance.

arc length pieces took the form $\{\sqrt{1+Y_t^2}\}$ while the series of squared and absolute returns pieces took the forms $\{Y_t^2\}$ and $\{|Y_t|\}$, respectively. All three series go from $t = 2$ to $t = 754$.

These new series were created in order to illustrate how volatility, not price, changes over time. The clustering applied to these series utilizes the correlation distance coupled with average linkage described in Sec. 2.3. As with k-means++, we once again consider three clusters: low volatility, medium volatility, and high volatility.

Figure 9 gives the results, where the stock ordering from Fig. 2 is used for all three volatility measures in order to have the comparisons involved with the correlation process have meaning. As can be seen, both arc length and squared returns end up with the same clusters, while absolute returns break Google from cluster 1 and put it in cluster 3.

As was done at the end of Sec. 3.1 with the Maharaj algorithm, we again supplement the correlation-based clustering results in this section with further simulation studies using those exact same 32 series (i.e., the same sixteen GARCH(1, 1) series and sixteen Pareto series). The accuracy rates under the GARCH(1,1) models for arc length, squared returns, and absolute returns stand at 51.67%, 51.07%, and 51.39%, respectively. All three accuracy rates are closer to each other than before, but arc length still retains a slightly higher value than the other two. Results also show accuracy rates of 49.72%, 40.48%, and 51.51% for arc length, squared returns, and absolute returns, respectively, under the Pareto distributions. As expected, squared returns once again show lower accuracy rates as expected since these series have no finite fourth moment.

4. Concluding comments

The results in this article reveal that arc length can successfully separate highly volatile stocks from less volatile ones. Arc length works just as well as squared returns when a finite fourth moment exists and outperforms it otherwise. Arc length can also quantify risk for a wide range of time series models, including multivariate models (see Wickramarachchi et al. (2015)). It is not clear how squared returns or absolute returns assess volatility in this multivariate setting.

Arc length can be further generalized to include the case of arbitrarily spaced time observations. Specifically, if series $\{X_t\}$ is observed at times $t_1, t_2, \ldots, t_n$, the arc length formula in Eq. (1) becomes

$$\sum_{i=2}^{n} \sqrt{(t_i - t_{i-1})^2 + (X_{t_i} - X_{t_{i-1}})^2}.$$

We note, however, that the functional central limit theorem for unequally spaced time series has not been proved yet and thus it is unknown if the Maharaj algorithm will work in this case. The authors are currently looking into this.

Notes

1. Persistence is sometimes referred to as volatility clustering, which is not to be confused with this article's goal of clustering via the feature of volatility.
2. Eastman Kodak has since changed its ticker symbol to KODK.
3. If arc length takes the form $\sum_{t=2}^{n} \sqrt{1 + Y_t^2}$, then squared and absolute returns take the forms $\sum_{t=2}^{n} Y_t^2$ and $\sum_{t=2}^{n} |Y_t|$, respectively.

References

Arthur, D., and S. Vassilvitskii. 2007. "K-Means++: The Advantages of Careful Seeding." Paper presented at the 18th Annual ACM SIAM Symposium on Discrete Algorithms, New Orleans, LA, January 7–9.

Dineen, S. 2013. *Probability Theory in Finance: A Mathematical Guide to the Black-Scholes Formula*, 2nd ed. United States: American Mathematical Society.

Golay, X., S. Kollias, G. Stoll, D. Meier, A. Valavanis, and P. Boesiger. 1998. "A New Correlation Based Fuzzy Logic Clustering Algorithm for FMRI." *Magnetic Resonance in Medicine* 40:249–260.

Liao, T.W. 2005. "Clustering of Time Series Data - A Survey." *Pattern Recognition* 38:1857–1874.

MacQueen, J. 1967. "Some Methods for Classification and Analysis of Multivariate Observations." In *Proceedings of the 5th*

Berkeley Symposium on Mathematical Statistics and Probability, edited by L. M. Le Cam and J. Neyman, 281–297. Berkeley, CA: University of California Press.

Maharaj, E.A. 2000. "Clusters of Time Series." *Journal of Classification* 17:297–314.

Ostrovsky, R., Y. Rabani, L.J. Schulman, and C. Swamy. 2012. "The Effectiveness of Lloyd-Type Methods for the *k*-Means Problem." *Journal of the ACM* 59 (article 28).

Taylor, S. 2005. *Asset Price Dynamics, Volatility, and Prediction*. Princeton, NJ: Princeton University Press.

Tunno, F., C. Gallagher, and R. Lund. 2012. "Arc Length Tests for Equivalent Autocovariances." *Journal of Statistical Computation and Simulation* 82:1799–1812.

Wickramarachchi, T., C. Gallagher, and R. Lund. 2015. "Arc Length Asymptotics for Multivariate Time Series." *Applied Stochastic Models in Business and Industry* 31:264–281.