Predicting Stock Market Crashes using Urns and the Quantum Lorenz Ordering of Correlations Matrices

by

J Rhet Montana

A thesis submitted in partial fulfillment

of the requirements for the degree of

Master of Science

Department of Applied Mathematics

Delft University of Technology

June, 2021

Professor Dr. Kees Oosterlee

Professor Dr. Pasquale Cirillo

"Americans only learn from catastrophe and not experience."
~ Theodore 'Teddy' Roosevelt ~

Abstract

By sampling financial correlation matrices over sliding windows, it has been shown in recent work that the quantum majorization induced partial ordering on this space of correlation matrices known as the "quantum Lorenz ordering" (QLO) can be used to characterize systemic risk by clustering correlation matrices according to their degree centrality on the associated directed graph called the "quantum majorization graph" (QMG). In this work, clusterings of the QMG are used to construct an online Bayesian nonparametric alarm system for the prediction of stock market crashes via the so-called "reinforced urn process" (RUP). To test the efficacy of this modelling methodology we exploit extreme value theory to systematically define stock market crashes by studying the tail of an appropriately fitted generalized pareto distribution (GPD) for stock market drawdowns. This approach identified 13 extreme drawdowns between 1985-2020, for which the RUP was trained from 1986-2005 to predict the 8 extreme drawdowns from 2005-2020. Of the three correlation metrics used to test this approach, the QLO corresponding to the set of upper Tail-Dependence λ^{U} matrices was shown to outperform the others: Pearson's ρ and the Gini correlation γ^{G}. Tail-Dependence was able to predict all 8 crashes with just 5 false alarms over a 12 month time horizon, all 8 with 7 false alarms over an 8 month time horizon, 7 out of 8 with 9 false alarms over a 4 time horizon, and 7 out of 8 with 17 false alarms over a 2 month time horizon. This approach was then tested against the usage of the Log-Periodic Power Law Singularity (LPPLS) model's confidence indicators with promising results. The quantum Lorenz ordering is meant to rank a set of correlation matrices by the amount of dispersion reflected in their spectra: a true heterogeneity. We consider this dispersion from the standpoint of measurement error as has been in the application of random matrix theory (RMT) to correlation matrices in portfolio risk theory. We provide analytical relations between quantum majorization and random matrix cleaning for a few RMT filtering schemes posing quantum majorization as a desirable condition for RMT filtering. The RUP is tested using these RMT cleaned correlation matrices as well.

Contents

List of Figures

List of Tables

1 Introduction

1

Introduction

1.1 | Brief

In this thesis, we develop an Urn-Based Alarm System for Predicting Stock Market Crashes via the usage of a recently developed methodology for the characterization of systemic risk based on correlation matrices: quantum majorization. We explain how quantum majorization characterizes systemic market risk, from the standpoint of Markowitz portfolio optimization schemes. We then show that, by sampling Correlation Matrices for the constituents of major Stock Market Indices over sliding windows, we are able to predict the "Black Monday" Cash of 1987, the "Great Recession" of 2008, the "COVID-19" Crash of 2020, along with most other major stock market events, by analyzing a Correlation Matrices position on the so-called Quantum Majorization Graph (QMG), which associates to each correlation matrix a specifically coloured marble in an Urn.

By representing the quantum lorenz ordering with the so-called quantum majorization Graph, we show that we are able to develop an Alarm System for stock market crashes using the Reinforced Urn Process (RUP), with good results. In its own right, this application of the RUP extends previous modelling methodologies utilized in Credit Risk and Epidemiology, into an entirely new domain: the prediction of Stock Market Crashes. Using Extreme Value Theory (EVT), we identify 13 notable stock market crashes between the years of 1980-2020 including infamous crashes such as "Black Monday", the "Tech Bubble", the "Sub-Prime Mortgage Crisis", "Corona", among other

smaller events. We train and test our RUP from Jan 1, 1985 to Dec 31, 2020 in an online fashion and show that our model is able to predict most crashes with relatively few "false alarms" on 3, 6, and 12 month time horizons. We permute these results across a wide variety of correlation metrics including Pearson's ρ^{P}, Gini's γ^{G}, and even the Upper ,and Lower, Tail-Dependencies λ^{U} and λ^{L}, respectively. We show that the RUP based on λ^{U} outperforms other correlation metrics.

Furthermore, we believe that unifying the theory of portfolio risk, Random Matrix Theory (RMT) "cleaning recipes", and quantum majorization will help make our Urn-Based Alarm System that much more intuitive, interpret-able, and robust in practice. Indeed, any alarm system, that is based on the sequential sampling of correlation matrices, may be sensitive to the particular type of correlation matrix the practitioner uses, which could easily be one of these RMT inspired "cleaning recipes". Since RMT inspired "cleaning recipes" aim to reduce portfolio risk by narrowing the dispersion of a correlation matrices eigenvalues, and quantum majorization characterizes portfolio risk by ranking two different correlation matrices based on how dispersed their eiganvalues are, their relationship needs to be explored, and accounted for, in the development, and assessment of, our Urn-Based Alarm System.

Hence, the goal of this thesis is threefold: First, in Chapters 2 and 8, we discuss the relationship between the dispersion of eigenvalues of a correlation matrix and portfolio risk through the lens of RMT inspired correlation matrix "cleaning recipes". Second, in Chapter 6, we develop the mathematical framework for our urn based alarm system. Third, in Chapter 8 we analyze the results of our model in proper historical back-tests and permute said results over several of the aforementioned "cleaning recipes" to see how they affect our results.

We hope to encourage the reader that correlation matrices are both an interesting and powerful tool for measuring, characterizing, and modelling systemic market risk. Additionally, as deep-learning based methodologies for tackling such problems become more and more prevalent, we hope to convince the reader that there is still a place for the development of pure mathematical models in the modern world. And that the RUP is, for example, one of such approaches.

1.2 | Setting the Stage

The contents of this work are built on top of the so-called Quantum Majorization Graph (QMG), developed by Fontanari et al. [2019], which represents the Quantum Lorenz Ordering (QLO) on the space of $d \times d$ correlation matrices in the form of a directed acyclic graph. By sequentially sampling Correlation Matrices of the daily returns for the constituents of major Stock Market

Indices over sliding windows, Fontanari et al. [2019] showed we can characterize the temporal evolution of market risk, over said windows, by analyzing a correlation matrices position on the QMG via spectral clustering, as is common practice in machine learning and the study of complex networks, in general. Using this clustering, we can choose an arbitrary number of k-Clusters and associate to each node in the QMG (correlation matrix) a level of risk $l \in \{1, \dots, k\}$ based on which "cluster" a correlation matrix (node) belongs to according to the spectral clustering.

In this work, we exploit the QMG formalism from Fontanari et al. [2019] to develop an Urn-Based Alarm System for Stock Market Crashes by associating to each cluster, mentioned earlier, a specifically coloured marble in a Polya urn. In this way, we are able to follow the approach of Cirillo and Hüsler [2011] and develop an Alarm System as a so-called Reinforced Urn Process (RUP) in true "Bayesian" and "Online" fashion along the lines of the framework developed by Antunes M and FK [2003].

By "correlation matrix", we consider a Universe $\mathcal{S}$ of d Financial Assets $\mathcal{S} = \{\mathcal{S}_1, \dots, \mathcal{S}_d\}$ that we observe at (daily) frequencies, emphasizing that we could also consider weekly, or even monthly frequencies. We sample a vector of (log) Daily-Returns $\mathcal{R}_\iota(\mathcal{S}_i)$ for Asset $\mathcal{S}_i$ over the time interval $\iota = [t_m, t_n]$

$$\mathcal{R}_\iota(\mathcal{S}_i) = \begin{pmatrix} \mathcal{R}_{t_m}(\mathcal{S}_i) \\ \vdots \\ \mathcal{R}_{t_n}(\mathcal{S}_i) \end{pmatrix}, \qquad \mathcal{R}_t(\mathcal{S}_i) = \log\left(\frac{\mathcal{S}_{i,t}}{\mathcal{S}_{i,t-1}}\right) \qquad \forall t \in \iota = [t_m, t_n] \tag{1.1}$$

where $\iota = [t_m, t_n]$ can be thought of as a "Sliding-Window" within a (much) larger Time-Interval $I = [0, T]$ where $0 \leqslant t_n < t_m \leqslant T$. Here, we define a $d \times d$ correlation matrix $\rho = \rho_\iota$ whose elements are sampled over the interval ι

$$\rho_\iota(i, j) = \begin{cases} \rho(\mathcal{R}_\iota(\mathcal{S}_i), \mathcal{R}_\iota(\mathcal{S}_j)), & \text{if } i \neq j \\ 1, & \text{otherwise} \end{cases} \tag{1.2}$$

for some Correlation or Dependence Metric $\rho(X, Y) \in [-1, 1]$.

1.2.1 | Portfolio Risk & the Quantum Lorenz Ordering

More generally, by "correlation matrix" we mean a $d \times d$ positive-semidefnite (PSD) hermitian operator ρ defined on a Hilbert space $\mathcal{H}_d$ of dimension $d = \dim(\mathcal{H}_d)$ with $\mathrm{diag}(\rho) = |1\rangle$

ones down the diagonal, exclusively. That is to say, $\rho \in \mathrm{PSD}(\mathcal{H}_d)$, for the set of PSD Matrices $\mathrm{PSD}(\mathcal{H}_d)$, where $\mathrm{Tr}[\rho] = d$. As such, any correlation matrix $\rho \in \mathrm{PSD}(\mathcal{H})$ can be decomposed into

$$\rho = \sum_{i=1}^{d} \lambda_i(\rho)|\psi_i\rangle\langle\psi_i| \tag{1.3}$$

with eigenvalues $\lambda(\rho) \coloneqq \{\lambda_i(\rho) : i = 1, \ldots, d\}$ and eigenbasis $\{|\psi_i\rangle\}_{i=1}^{d}$, by the spectral theorem of hermitian operators. Hence, since $\rho \in \mathrm{PSD}(\mathcal{H}_d)$, and $\mathrm{Tr}[\rho] = d$, we know $\{\lambda_i(\rho) \geqslant 0\}_{i=1}^{d}$ defines a (discrete) distribution on $\lambda(\rho) \in \mathbb{R}_+^d$. Thus, a correlation matrix ρ can be referred to as a "density operator". This "eigen"-distribution, if you will, is the central object of this work.

The so-called "Lorenz Ordering" is a partial-ordering on the space of probability distribution functions (PDF). Here, one PDF p is "higher up", on the Lorenz Ordering, than another PDF, say p^*, if p "majorizes" p^*, written as $p \succeq p^*$. The "majorization condition" $p \succeq p^*$ implies

$$\begin{aligned} \sum_{i=1}^{\alpha} p_i^{\downarrow} &\geqslant \sum_{i=1}^{\alpha} p_i^{*\downarrow} \qquad \forall \alpha = 1, \ldots, d \\ \int_{-\infty}^{\alpha} x p(x) dx &\geqslant \int_{-\infty}^{\alpha} x p^*(x) dx \qquad \forall \alpha \leqslant \infty \end{aligned} \tag{1.4}$$

for discrete and continuous PDF's, respectively. The reason we call this the "Lorenz" Ordering is because 1.4 happens to correspond to the "Lorenz Curve" of a PDF

$$\begin{aligned} L_\alpha(p) &= \sum_{i=1}^{\alpha} p_i^{\downarrow} \\ L_\alpha(p(\alpha)) &= \mu^{-1} \int_{-\infty}^{\alpha} x p(x) dx \end{aligned} \tag{1.5}$$

for the discrete and continuous case with corresponding "majorization condition(s)"

$$p \succeq p^* \Leftrightarrow \begin{cases} L_\alpha(p) \geqslant L_\alpha(p^*) \qquad \forall \alpha \in \{1, \ldots, d\} \\ L_\alpha(p(\alpha)) \geqslant L_\alpha(p^*(\alpha)) \qquad \forall \alpha \in \{1, \ldots, d\} \end{cases} \tag{1.6}$$

for discrete and continuous PDF's, respectively. Aptly so, applying this "majorization condition" to the eigenvalues $\lambda(\rho)$ and $\lambda(\rho^*)$ of density operators (or quantum states), ρ and ρ^*, gives us the

definition of "quantum majorization"

$$\rho \succeq \rho^* \Leftrightarrow \sum_{i=1}^{\alpha} \lambda_i^{\downarrow}(\rho) \geqslant \sum_{i=1}^{\alpha} \lambda_i^{\downarrow}(\rho^*) \qquad \forall \alpha \in \{1, \ldots, d\} \tag{1.7}$$

and the subsequent "quantum lorenz ordering" (QLO) which can be equivalently expressed according to the definition of the so-called "quantum lorenz curve" (QLC)

$$\begin{aligned} L_\alpha(\rho) &\coloneqq \frac{1}{d} \sum_{i=1}^{d} \lambda_i^{\downarrow}(\rho) \\ \rho \succeq \rho^* &\Leftrightarrow \sum_{i=1}^{\alpha} L_\alpha(\rho) \geqslant \sum_{i=1}^{\alpha} L_\alpha(\rho^*) \qquad \forall \alpha \in \{1, \ldots, d\} \end{aligned} \tag{1.8}$$

defined on the space of density operators, and thus, correlation m matrices.

1.2.2 | Quantum Majorization & Random Matrix Theory

Max Otto Lorenz [1905] first invented the lorenz curve to study the distribution of wealth among the members of a society. Thus, the Lorenz curve $L_\alpha(\cdot)$ has become synonymous with the study of statistical variability, in general contexts. Furthermore, in their seminal work, Fontanari et al. [2019] alluded to the distinction, originally made by Gini [1912], between the two different types of statistical variability;

1. Measurement Error
2. Socioeconomic Variability

Socioeconomic variability reflects a true heterogeneity (distribution of wealth within a society) whereas measurement error reflects a sort of ignorance for the true underlying value (precise height of a mountain). Gini [1912] argued that the amount of Statistical Variability ought to be measured differently when dealing with the two different types of statistical variability. Specifically, Fontanari et al. [2019] considered the Lorenz Curve $L_\alpha(\rho)$, of a correlation matrix ρ, within the context of the 2^{nd} variety: socioeconomic variability. As fruitful and sound as that line of reasoning may be, we believe that studying the Lorenz Curve $L_\alpha(\rho)$ in the context of the 1^{st} variety, measurement error, reveals a novel connection between quantum majorization and the "Random Matrix Theory" (RMT) inspired correlation matrix "cleaning recipes" as extensively analyzed by

the likes of, for instance, Bouchaud and Potters [2009] and Bun et al. [2017].

As noted by Bouchaud and Potters [2009], the "Modern Portfolio Theory" (MPT) of Markowitz [1952] shows us how Measurement Error can (potentially) effect the "optimally" obtained weights of a Portfolio $\Omega^* = \Omega^*(\mathcal{S})$ for a set of Financial Assets $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_d\}$. Indeed, the whole point of MPT 1952 is to minimize the risk $\sigma^2 = \sigma^2(\Omega_{\mathcal{S}})$ of a portfolio $\Omega = \Omega_{\mathcal{S}}$

$$\sigma^2(\Omega_{\mathcal{S}}) = \sum_{i=1}^{d} \sum_{j=1}^{d} \omega_i \sigma_i \rho_{i,j} \sigma_i \omega_i \tag{1.9}$$

to find a set of "optimal" weights

$$\Omega^*_{\mathcal{S}} = \{\omega^*_1, \ldots, \omega^*_d\} \tag{1.10}$$

noting that the $\rho_{i,j}$'s of equation 1.9 denote the elements of the $d \times d$ correlation matrix ρ for the (daily) returns on the set of financial assets $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_d\}$. Here, we see that the "optimal" Weights $\Omega^*_{\mathcal{S}} = \{\omega^*_1, \ldots, \omega^*_d\}$ take the form

$$\omega^*_i \propto \mu_i + \sum_{j=1}^{d} \sum_{k=1}^{d} (\lambda_j^{-1}(\rho) - 1)\psi_{j,i}\psi_{j,k}\mu_j \tag{1.11}$$

for the eigenvalues $\lambda_i(\rho)$ of the correlation matrix ρ, with $\psi_{i,j}$ denoting the j^{th} element of the i^{th} eigenvector $|\psi_i\rangle$ and expected return $\mu_i = \mu(\mathcal{S}_i)$ for the financial asset $\mathcal{S}_i$. Since each ω^*_i is a function of the inverse $\lambda_i^{-1}(\rho)$ of the eigenvalues of ρ, such a strategy tends to allocate large ω^*_i to financial assets $\mathcal{S}_i$ associated with ρ's smallest eigenvalues. However, the measurement of these small eigenvalues may entirely be dominated by measurement error, meaning, that our $\Omega^*_{\mathcal{S}} = \{\omega^*_1, \ldots, \omega^*_d\}$ may not be optimal at all. Hence, such a portfolio's "realized" (future) risk, is directly related to how "dispersed" our eigenvalues $\lambda(\rho)$ are. Which is exactly what the Lorenz curve $Ł_\alpha(\rho)$ is meant to study.

In fact, the RMT "cleaning recipes" of Bouchaud and Potters [2009] and Bun et al. [2016] intend to "distort" the distribution of $\lambda(\rho)$ in a manner that "lowers" ("raises") the large (small) eigenvalues. Hence, these RMT "cleaning recipes" can be seen as an attempt to filter out the measurement error, and get to the "true" Spectrum $\lambda(\rho)$ that honestly reflects the heterogeneity in the relationships between the financial assets $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_d\}$: socioeconomic variability. For this reason, we find it important to build the upcoming Alarm System with a perspective that respects this intermediate attempt to "clean" correlation matrices that accounts for measurement

error and socioeconomic variability.

1.2.3 | Quantum Majorization Graph

By considering the constituent financial assets $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_d\}$ of major stock market indices, we can track the temporal evolution of market risk by treating the index itself as a portfolio we wish to hedge. From the standpoint of correlation matrices, we can measure said market risk by sequentially sampling correlation matrices over sliding-windows with windows and shifts of arbitrary size, forming a collection of correlation matrices $\rho = \{\rho_1, \ldots, \rho_n\}$ over the Time-Interval $I := [t_0, T]$ of interest. Hence, each $\rho_i \in \rho$ corresponds to its own time-interval $\iota_i \in I$. In this way, any risk-measure, defined on a particular correlation matrix $\rho_i = \rho_{\iota_i} \in \rho$, characterizes the risk of the market over the time interval ι_i spanned by the samples used to construct $\rho_i = \rho_{\iota_i}$.

Using this collection $\rho = \{\rho_1, \ldots, \rho_n\}$, we can represent the corresponding QLO on the so-called quantum majorization Graph (QMG), as per the work of Fontanari et al. [2019]. In short, the adjacency matrix $A \in \{0,1\}^{n \times n}$, associated to the QMG $G = (V, E)$, is defined according to

$$A = \Psi - I_n \qquad \text{such that} \qquad \Psi_{i,j} = \begin{cases} 1 \text{ if } \rho_i \succeq \rho_j \\ 0 \text{ otherwise} \end{cases} \tag{1.12}$$

with "Quantum Majorization Matrix" (QMM) Ψ and edge-set $E \sim A$, corresponding to A, to remove self-directed loops, along with vertex-set $V \sim \rho$ corresponding to the set of correlation matrices $\rho = \{\rho_1, \ldots, \rho_n\}$ of cardinality $|V| = n$. Trivially, such a QMG $G = (V, E)$ is a special kind of directed acyclic graph (DAG). "Directed" because the quantum lorenz ordering "ranks" two different correlation matrices ρ_i and ρ_j and "acyclic" because the QLO is, well, an ordering.

As is common practice in Machine Learning and the study of complex networks, in general, Fontanari et al. [2019] clustered the Vertex-Set $V \sim \rho$ using the spectral clustering algorithm. By choosing $k = \{2, 3\}$-Clusters, Fontanari et al. [2019] showed that these clusters corresponded to appropriately chosen discriminating thresholds for the in-out degree centrality $\theta(v_i)$ of a Vertex $v_i \in V$

$$\theta(v_i) = \frac{1}{2} + \frac{1}{2n}(deg^+(v_i) - deg^-(v_i)) \in [0, 1] \tag{1.13}$$

with $deg^+(v_i)$ and $deg^-(v_i)$ denoting the "out", and "in", "degrees" of the Vertex $v_i \in V$ where $\theta = 1$ corresponds to the "riskiest" correlation matrix (time-interval), and $\theta = 0$ the "safest". Intu-

itively enough, for $k = 2$-clusters, the appropriate discriminating threshold d^*, for distinguishing between "risky" and "safe" time periods, turns out to be $d^* = 0.54$. Whereas for $k = 3$-Clusters, we have discriminating thresholds $\{d_1, d_2\} = \{0.38, 0.64\}$, distinguishing between "safe", "moderate", and "risky" time periods in the market.

1.2.4 | Reinforced Urn Alarm System

Interestingly, we can use the QMG to construct a so-called "Reinforced Urn Process" (RUP). Using the spectral clustering, discussed earlier, we can associate a specific correlation matrix $\rho_i \in \rho$ to one of the k-Clusters $\rho_i \in \{C_1, \ldots, C_k\}$. With a little imagination, we can associate a Colour C_i to each of these k-Clusters $\{C_1, \ldots, C_k\}$ and model the temporal evolution of risk, as characterized by quantum majorization, using an Urn model, such as the RUP. In this way, we think of observing a new ρ_i the same way as drawing a coloured marble out of a Polya urn. For instance, by choosing $k = 3$-Clusters, we can associate any $\rho_i \in \rho$ to $\rho_i \in \{C_1, C_2, C_3\}$ belonging to "Safe", "Moderate", and "Risky" regions for colours $C = \{C_1, C_2, C_3\}$, respectively, depending on ρ_i's position on the $G = (V, E)$ QMG as per the spectral clustering.

It has been shown by Cirillo and Hüsler [2011], and Peluso et al. [2015], that the RUP may be of practical interest for developing alarm systems and catastrophe models. The idea is simple: Define a State-Space S with elements $s = (n, l) \in S$ that denote the "level" of risk l at time-instant n. The process evolves over time in such a way that

$$X_0 = (0, 0) \to (1, l_1) \to (2, l_2) \to (3, l_3) \to \cdots \to X_{n^*} = (n^*, \mathcal{L}) \to X_0 = (0, 0) \tag{1.14}$$

the process returns to the initial $(0, 0)$-State once the "catastrophic" Level $\mathcal{L}$ is reached at the critical time-instant n^*. Here, State-Transitions between $X_n = s = (n, l) \to X_{n+1} = (n + 1, l_{n+1})$ are entirely characterized by the sampling of colours $C = \{C_1, \ldots, C_k\}$ out of the "composition" of the Urn $U(s)$ according to the Rule-of-Motion (RoM) $d : S \times C \to S$

$$d(s, C_i) = (n + 1, l(C_i)), \qquad \forall C_i \in \{C_1, \ldots, C_k\} \tag{1.15}$$

where the new Risk-Level $l = l(C)$ depends on which category of risk the newly sample correlation matrix ρ_i belongs to. This construction creates a sequence of r stopping-times, which are

denoted by $\tau \coloneqq \{\tau_1, \ldots, \tau_r\}$,

$$\tau_s = \inf\{n \geqslant \tau_{s-1} : X_n = (0,0)\} \tag{1.16}$$

that signify when the process X suffers a catastrophic failure.

Along the lines of Antunes M and FK [2003], an alarm system can be constructed using this Process X. In general, we "cast" an "alarm" $\mathcal{A}_t$, at time t, for a "catastrophe" $\mathcal{C}_{t+j}$, in j time-steps, when we have

$$\mathbb{P}[\mathcal{C}_{t+j}|\vec{X}_1] \geqslant \gamma, \qquad \gamma \in (0,1) \tag{1.17}$$

where $\vec{X}_1$ denotes the "Past" $\vec{X}_1 = \{X_1, \ldots, X_{t-k}\}$ information for some arbitrarily chosen $k \geqslant 0$ that distinguishes $\vec{X}_1$ from the "present" $\vec{X}_2 = \{X_{t-k+1}, \ldots, X_t\}$. Here, it is said that the "Alarm"is of "size" γ. Clearly, according to the RUP, a "catastrophe" $\mathcal{C}_{t+j}$ is nothing more than the event $X_{n+j} = (n+j, \mathcal{L})$, or, in terms of our Stopping-Times

$$\mathbb{P}[\mathcal{C}_{t+j}|\vec{X}_1] \geqslant \gamma \rightarrow \mathbb{P}[\tau_{r+1} = n+j|\{\tau_1, \ldots, \tau_r\}] \geqslant \gamma \tag{1.18}$$

which can be easily computed thanks to the underlying urn $U(s)$ construction of the evolution of the Process X. Indeed, all that needs to be computed are the total number of possible sampling sequences out of the Urns $U(s)$ that result in $X_{n+j} = (n+j, \mathcal{L})$.

1.3 | A Reader's Guide to this Thesis

The contents of this Thesis are collected into 2 Parts: Part II, and Part III. Part II State-of-the-Art offers more detailed introductions to the core mathematical objects that we will be using to develop our Urn-Based Alarm System for Stock Market Crashes. Part III Implementation discusses the implementation details of the Urn-Based Alarm System. In particular, Part II contains Chapters 2, 3, and 4. Whereas Part III consists of Chapters 6, 7, and 8.

In Chapter 2, we touch on the general theory of Portfolio Risk and build a stronger intuition for as to why we should be interested in the statistical dispersion of a correlation matrices Eigenvalues from the lens of RMT inspired "cleaning recipes". In Chapter 3, we cover the key elements from Fontanari et al. [2019] and build the quantum majorization graph (QMG), whose spectral clustering essentially defines our urn model. Then, in Chapter 4, we cover the details of the Reinforced Urn Process (RUP), and bayesian alarm system of Antunes M and FK [2003], and outline

how they fit together using the Urn-Based Alarm System model of Cirillo and Hüsler [2011].

Then, in Part III, we begin with Chapter 6 where we discuss the Data we'll be using, the different measures of correlation we'll be considering, and quickly explain the methodology we'll be using to precisely define a Stock Market "Crash" using a novel approach based on Extreme Value Theory (EVT). In Chapter **??** we rigorously define the RUP, and corresponding alarm system, we'll be implementing along with new analytical results on the relationship between the quantum majorization and several prominent "cleaning recipes" inspired from RMT which we'll be permuting our urn model over. Lastly, in Chapter 8, we analyze said results for the different correlation measures, and RMT "cleaning recipes".

II

State of the Art

2

Portfolio Risk & Random Matrix Theory

In this section, we outline the intimate relationship that the eigenvalues of an empirical correlation Matrix ρ share with the "optimality" of financial portfolios, under the Modern Portfolio Theory (MPT) framework of Markowitz [1952]. In particular, many authors including Bouchaud and Potters [2009], comment on how the optimality criteria of MPT optimized portfolios are often compromised by measurement errors in the empirical correlation matrix itself. Interestingly enough, Gini [1912] characterize statistical variability in two varieties:

1. Measurement Error
2. Socioeconomic Variability

Bouchaud and Potters [2009] show how measurement errors in the emprical correlation matrix distort the "optimal" portfolio Weights towards small eigenvalues, which are often affected by measurement error. Hence, several authors including Bouchaud and Potters [2009], Bun et al. [2016], and Bun et al. [2017], have introduced "cleaning" techniques, inspired by Random Matrix Theory (RMT) (see Marčenko and Pastur [1967]), to narrow the distribution of a correlation matrices spectrum. Later, in chapter **??**, we will see how the cleaning techniques of Bouchaud and Potters [2009], Bun et al. [2016], and Bun et al. [2017] relate to the Quantum Lorenz Ordering (QLO), but, for now, this chapter will be organized as follows: In section 2.1, we will briefly recap MPT and discuss the empirical correlation matrices role in portfolio optimization. Then, in

section 2.2 we discuss how the so-called "Out-of-Sample" risk is related to how dispersed a correlation matrices eigenvalues are. Lastly, in section 2.3, we will define a handful of RMT "filtering" schemes for empirical correlation matrices that have intriguing relationships to the QLO, which will be discussed later.

2.1 | Correlation Matrices & Portfolio Risk

Correlation matrices have always been used in the problem of minimizing the risk of a portfolio. Indeed, the correlation matrix ρ for a set of d financial assets $\mathcal{S} = \{\mathcal{S}_1, \dots, \mathcal{S}_d\}$ is of central importance in the "Modern Portfolio Theory" (MPT) of Markowitz [1952]. Under the MPT framework, we consider the returns $\mathcal{R}_\iota(\mathcal{S}_i)$ for each of the d financial assets $\mathcal{S}_i \in \{\mathcal{S}_1, \dots, \mathcal{S}_d\}$

$$\mathcal{R}_\iota(\mathcal{S}_i) = \begin{pmatrix} \mathcal{R}_{t_m}(\mathcal{S}_i) \\ \vdots \\ \mathcal{R}_{t_n}(\mathcal{S}_i) \end{pmatrix}, \qquad \mathcal{R}_t(\mathcal{S}_i) = \log\left(\frac{\mathcal{S}_{i,t}}{\mathcal{S}_{i,t-1}}\right) \qquad \forall t \in \iota = [t_m, t_n] \tag{2.1}$$

whose samples are collected over the time-interval $\iota = [t_m, t_n]$. Using these d vectors of returns $\mathcal{R} = \{\mathcal{R}_\iota(\mathcal{S}_1), \dots, \mathcal{R}_\iota(\mathcal{S}_d)\}$, we define the "empirical" correlation Matrix ρ using equation

$$\rho_\iota(i,j) = \begin{cases} \rho(\mathcal{R}_\iota(\mathcal{S}_i), \mathcal{R}_\iota(\mathcal{S}_j)), & \text{if } i \neq j \\ 1, & \text{otherwise} \end{cases} \tag{2.2}$$

for an arbitrary correlation metric $\rho(X, Y) \in [-1, 1]$. Here, the aim of MPT is to find the set of optimal weights $\Omega^*(\mathcal{S}) = \{\omega_1^*, \dots, \omega_d^*\}$ to assign to each financial asset by minimizing the portfolio's $\Pi(\mathcal{S}) = \{\omega_1 \mathcal{S}_1, \dots, \omega_d \mathcal{S}_d\}$ risk $\mathcal{R}^2 = \mathcal{R}^2(\Pi(\mathcal{S}))$ according to the program

$$\begin{aligned} \min \quad & \mathcal{R}^2 = \sum_{i,j=1}^{d} \omega_i \sigma_i \rho_{i,j} \sigma_i \omega_i \\ \text{s.t.} \quad & \mu^* = \sum_{i=1}^{d} \omega_i \mu_i, \qquad \mu_i = \mathbb{E}[\mathcal{R}_\iota(\mathcal{S}_i)] \end{aligned} \tag{2.3}$$

subject to obtaining a desired return μ^*. Solving the quadratic program of equation(s) 2.3 provides the optimal weights $\Omega^*(\mathcal{S}) = \{\omega_1^*, \dots, \omega_d^*\}$ corresponding to the time period used to sample individual asset volatilities σ_i and correlations $\rho_{i,j}$. However, the performance of the resulting

"optimal" portfolio $\Pi^*(\mathcal{S}) = \{\omega_1^* \mathcal{S}_1, \ldots, \omega_d^* \mathcal{S}_d\}$ will be subject to the future time period for which the samples σ_i and $\rho_{i,j}$ were not used in the optimization program (we cannot predict the future). To formalize this, we introduce the following three notions for the empirical correlation matrix ρ, the "True" correlation matrix P and $\langle \tilde{\omega}_\rho | = \{\omega_1^* \sigma_1, \ldots, \omega_1^* \sigma_1\}$ denotes the optimal weights for ρ:

- "In-Sample" risk $\mathcal{R}^2_{in}$ is the Risk for the optimal portfolio over the time period used to build it.

$$\mathcal{R}^2_{in} = \langle \tilde{\omega}_\rho | \rho | \tilde{\omega}_\rho \rangle \tag{2.4}$$

- "True" risk $\mathcal{R}^2_{true}$ is the Risk in the ideal world where the true correlation matrix P is used.

$$\mathcal{R}^2_{true} = \langle \tilde{\omega}_P | P | \tilde{\omega}_P \rangle \tag{2.5}$$

- "Out-of-Sample" risk $\mathcal{R}^2_{out}$ is the risk of the portfolio constructed using the empirical correlation matrix ρ, but is observed in the next time period.

$$\mathcal{R}^2_{out} = \langle \tilde{\omega}_\rho | \rho^{-1} P \rho^{-1} | \tilde{\omega}_\rho \rangle \tag{2.6}$$

Interestingly, Bouchaud and Potters [2009] show that one can use a convexity argument to say

$$\overline{\langle \tilde{\omega}_\rho | \rho | \tilde{\omega}_\rho \rangle} \leqslant \langle \tilde{\omega}_P | P | \tilde{\omega}_P \rangle \tag{2.7}$$

where $\overline{A}$ denotes the long-run average for the matrix A, from which we notice

$$\mathcal{R}^2_{in} \leqslant \mathcal{R}^2_{true} \tag{2.8}$$

as one might expect. By optimality, we can say something similar with respect to the "True" and "Out-of-Sample" Risks

$$\mathcal{R}^2_{true} \leqslant \mathcal{R}^2_{out} \tag{2.9}$$

and thus we have the full relation

$$\mathcal{R}^2_{in} \leqslant \mathcal{R}^2_{true} \leqslant \mathcal{R}^2_{out} \tag{2.10}$$

for "In", "True" and "Out-of-Sample" Risks $\mathcal{R}^2_{in}$, $\mathcal{R}^2_{true}$, and $\mathcal{R}^2_{out}$, respectively. Hence, the risk that an "optimized" portfolio Π^* will be exposed to in the future will always be greater than, or equal to, the risk suggested by the quadratic program used to construct Π^* in the first place. Since this procedure relies on the empirical correlation matrix ρ, with spectral decomposition

$$\rho = \sum_{i=1}^{d} \lambda_i(\rho) v_i v_i^T \tag{2.11}$$

it would be interesting to see how the "optimal" weights $\Omega^*(\mathcal{S}) = \{\omega_1^*, \ldots, \omega_d^*\}$ are related to the correlation matrices eigenvalues $\lambda_i(\rho)$. Reason being, the eigenvalues of a correlation matrix turn out to have strong financial interpretations that allow us to characterize portfolio risk in an interesting way. We explain this relationship next.

2.2 | Spectral Dispersion & Portfolio Risk

From the previous section we saw how Markowitz "optimized" portfolios are exposed to substantial "out-of-sample" risk. In this Section, we want to see if we can use the Empirical Correlation Matrix ρ to characterize the amount of "out-of-sample" risk $\mathcal{R}^2_{out}$ a portfolio $\Pi^*(\mathcal{S})$ might be exposed to. We know that any correlation matrix ρ will have a spectral decomposition of the form

$$\rho = \sum_{i=1}^{d} \lambda_i(\rho) v_i v_i^T \tag{2.12}$$

since correlation matrices are hermitian (symmetric). Hence, it might be interesting to see if we relate the "optimality" of a portfolio $\Pi^*(\mathcal{S})$ to the eigenvalues $\lambda(\rho)$ of ρ.

We begin by solving the quadratic program of equation(s) 2.3. Using lagrange-multipliers, we find that the optimal weights $\Omega^*(\mathcal{S}) = \{\omega_1^*, \ldots, \omega_d^*\}$ obey the relation

$$\omega_i^* \sigma_i = \mu^* \frac{\sum_j \rho_{i,j}^{-1} \frac{\mu_j}{\sigma_j}}{\sum_{i,j} \frac{\mu_i}{\sigma_i} \sum_j \rho_{i,j}^{-1} \frac{\mu_j}{\sigma_j}} \tag{2.13}$$

using individual asset volatility's $\sigma_i = \sigma(\mathcal{S}_i)$ and correlation matrix elements $\rho_{i,j}$. We can simplify the above expression for $\omega_i^* \sigma_i$

$$\omega_i^* \propto \mu_i + \sum_{j=1}^{d} \sum_{k=1}^{d} (\lambda_j^{-1}(\rho) - 1) v_{j,i} v_{j,k} \mu_j \tag{2.14}$$

for the eigenvalues $\lambda(\rho) = \{\lambda_1(\rho), \ldots, \lambda_d(\rho)\}$ of ρ and Eigenbasis $\{v_i\}_{i=1}^{d}$ where $v_{i,j}$ denotes the j^{th} element of the i^{th} eigenvector v_i. Equation 2.14 reveals something interesting:

Markowitz optimization schemes tend to allocate large weights to small eigenvalues.

However, as alluded to in Bouchaud and Potters [2009], the measurement of small eigenvalues may be entirely dominated by measurement error: the 1^{st} variety of statistical variability. Hence, since $\Omega^*(\mathcal{S}) = \{\omega_1^*, \ldots, \omega_d^*\}$ is meant to minimize the risk of our portfolio $\mathcal{R}^2 = \mathcal{R}^2(\Pi_S)$, our entire notion of portfolio risk is directly related to the amount of dispersion in the distribution of the eigenvalues $\lambda(\rho)$ of the correlation matrix ρ. Thus, the amount of "out-of-sample" risk a portfolio will be exposed to is potentially compromised by the shape of the Spectrum $\lambda(\rho)$. For this reason, the spectrum $\lambda(\rho)$ of the empirical correlation matrix ρ, a core element of this thesis, becomes an object of interest for the characterization of systemic market risk. For instance, we can consider correlation matrices for the 30 constituents of the Dow Jones Industrial Average (DJIA) and use $\lambda(\rho)$ as means to measure systemic market risk.

To address all of this, several "Cleaning Recipes", inspired by Random Matrix Theory (RMT), have been developed and studied in Bouchaud and Potters [2009] among others. We introduce a few of these next in hopes they bear an interesting relationship with the quantum lorenz ordering (QLO), later.

2.3 | Some Filtering Procedures

We will briefly define, and discuss, a few of the most prominent random matrix cleaning techniques Bun et al. [2017]: The Basic Linear Shrinkage Estimator (BLS), Eigenvalue Clipping (CLP), and the Rotationally Invariant Estimator (RIE). We denote with $\Xi^{BLS}[\rho]$, $\Xi^{CLP}[\rho]$, and $\Xi^{RIE}[\rho]$ their respective cleaning operations $\Xi^*[\cdot]$. For the remainder of this section, we will refer to the fol-

lowing Spectral Decomposition for a cleaned financial correlation matrix $\Xi^*[\rho]$

$$\Xi^*[\rho] = \sum_{k=1}^{d} \xi_k^* v_i v_i^T \tag{2.15}$$

with "cleaned" eigenvalues $\{\xi_k^*\}_{k=1}^d \in \lambda(\Xi^*[\rho])$ while preserving the original eigen-basis $\{|\psi\rangle_k\}_{k=1}^d$ from the "raw" Correlation Matrix ρ with eigenvalues $\{\lambda_k(\rho)\}_{k=1}^d \in \lambda(\rho)$. We begin with BLS.

Definition 2.1 (Basic Linear Shrinkage)**.** The Basic Linear Shrinkage, or Linear Shrinkage, estimator for the true correlation matrix $\Xi^{BLS}[\rho]$ is defined as,

$$\Xi^{BLS}[\rho] \coloneqq \sum_{i=1}^{d} \xi_i^{BLS} v_i v_i^T, \qquad \xi_i^{BLS} = \beta\lambda_i(\rho) + (1-\beta) \tag{2.16}$$

that is, $\Xi^{BLS}[\rho] = \beta\rho + (1-\beta)I_d$, for some constant $\beta \in [0,1]$.

Though quite simple, this method proves to be difficult to outperform in practice Bun et al. [2016]. It is not unreasonable to think that the effectiveness of this procedure may be related to the effectiveness of the Talmudic "$\frac{1}{N}$" allocation Duchin and Levy [2009]. Indeed, $\Xi^{BLS}[\rho]$ forms a convex combination between ρ and I_d, the identity matrix which corresponds to the Talmudic "$\frac{1}{N}$" allocation.

Remark 2.2. It has been shown in Bun et al. [2017] that an optimal choice for β is $\beta \coloneqq (1 + 2q\kappa)^{-1}$ where $q = d^{-1}T$, $d = \dim(\rho)$, and κ a hyper-parameter for an Inverse-Wishart prior.

Definition 2.3 (Eigenvalues Clipping)**.** The Eigenvalues Clipping, Clipping, or Clipped, estimator for the true correlation matrix $\Xi^{CLP}[\rho]$ is defined as,

$$\Xi^{CLP}[\rho] \coloneqq \sum_{i=1}^{d} \xi_i^{CLP} v_i v_i^T, \qquad \xi_i^{CLP} \coloneqq \begin{cases} \lambda_i(\rho), & \text{if } i \leqslant \lceil d\beta \rceil \\ \gamma, & \text{otherwise} \end{cases} \tag{2.17}$$

where the top $\lceil d\beta \rceil$ eigenvalues are untouched and γ is an appropriately chosen trace-preserving constant such that $\mathrm{Tr}[\Xi^{CLP}[\rho]] = \mathrm{Tr}[\rho] = 1$.

Remark 2.4. A typical procedure is to choose β such that eigenvalues beyond the Marcenko-Pastur upper edge ($\lambda_i(\rho) \geqslant (1 + \sqrt{d/T})^2$, see Bouchaud and Potters [2009]) are kept (signals), and normalize the rest (noise).

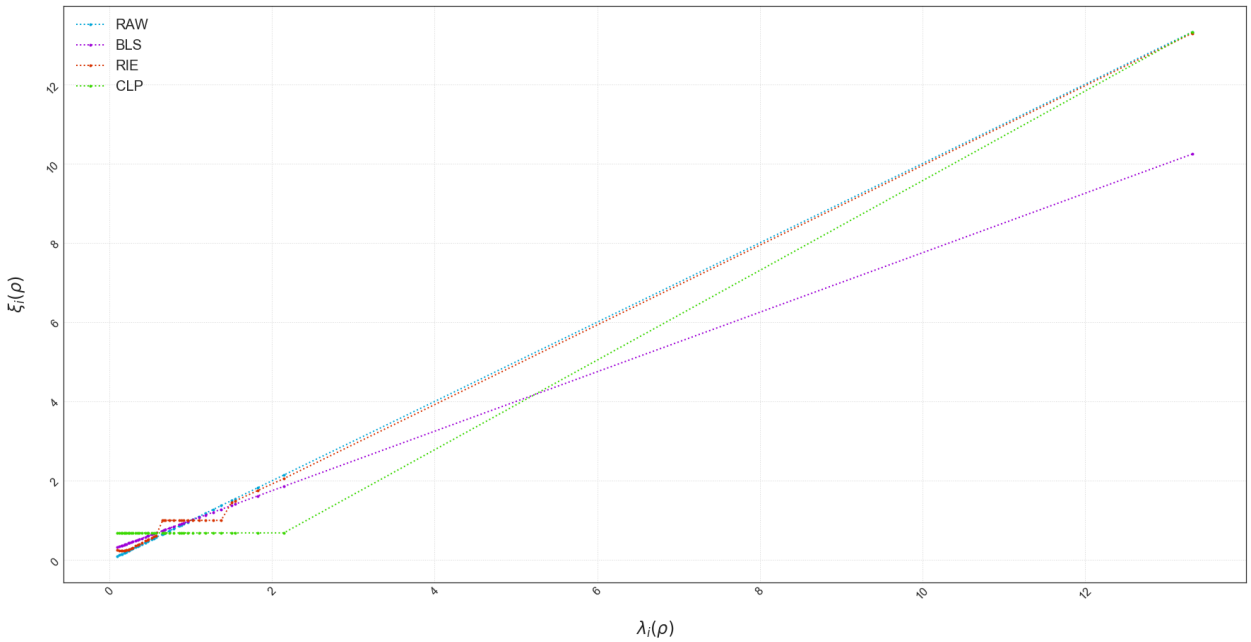


Figure 2.1: RMT Cleaning recipes as functions of the Raw Spectrum

Plotting $\xi_i(\rho)$ against $\lambda(\rho)$ for the RMT filtering schemes RIE, BLS, and CLP..

The idea with eigenvalue clipping is that, perhaps, the top $\lceil d\beta \rceil$ eigenvalues represent a sufficient rank-$\lceil d\beta \rceil$ approximation to the true correlation matrix. Thus, normalizing the bottom $(d - \lceil d\beta \rceil)$-eigenvalues, with (say) a trace-preserving constant γ, will suppress noisy fluctuations in the spectra, while retaining the strong signals indicated with the top $\lceil d\beta \rceil$-eigenvalues.

Definition 2.5 (Rotationally Invariant Estimator)**.** The Rotationally Invariant, Nonlinear Shrinkage, or Oracle Estimator for the true correlation matrix $\Xi^{\text{RIE}}[\rho]$ is defined as,

$$\Xi^{\text{RIE}}[\rho] \coloneqq \sum_{i=1}^{d} \xi_i^{\text{RIE}} v_i v_i^T, \qquad \xi_i^{\text{RIE}} \coloneqq \frac{\lambda_i(\rho)}{|1 - q + q z_i g_\rho(z_i)|^2} \tag{2.18}$$

for $q = d^{-1}T$, $|\cdot|$ the complex modulus of a complex number, and $g_\rho(z_i)$ the Stieljes transform, the normalized trace of the Resolvent $G_\rho(z_i)$, that is $g_\rho(z_i) = \frac{1}{d}\text{Tr}[G_\rho(z)]$, of ρ, defined below

$$G_\rho(z_k) = \sum_{i=1}^{d} \frac{\lambda_k(\rho) + i\eta}{(\lambda_k(\rho) - \lambda_i(\rho))^2 + \eta^2} v_i v_i^T \tag{2.19}$$

where $z_k = \lambda_k(\rho) - i\eta$ and $\lambda_k(\rho)$ an eigenvalue $\lambda_k(\rho) \in \lambda(\rho)$.

Similar to the Basic Linear Shrinkage estimator in 2.1, the RIE estimator in 2.5 "shrinks" the top eigenvalues while "growing" the bottom ones. However, unlike the BLS estimator, it does so in "nonlinear" fashion, shrinking and growing eigenvalues at a rate proportional to their relative size. In Bun et al. [2017] it is shown how this induces a "systemic bias" towards small eigenvalues $\lambda_i(\rho) \to 0$, and thus, one needs to be careful when implementing the RIE estimator, especially for small d and T (as is usually the case with Financial Portfolios).

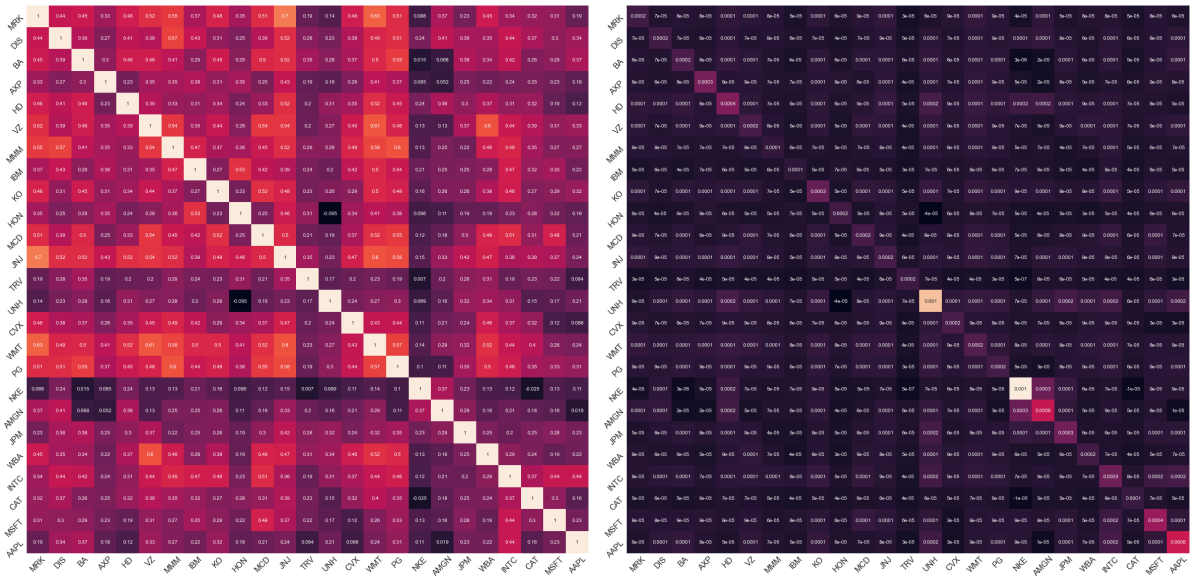
3

A Naive Alarm-System using the Quantum Lorenz Order

Consider a Market consisting of d financial assets $\mathcal{S} \coloneqq \{\mathcal{S}_1, \ldots, \mathcal{S}_d\}$. In Fontanari et al. [2019], a novel approach to characterizing market risk was introduced: the Quantum Lorenz Order. The Quantum Lorenz Order is a Partial Ordering between Hermitian Matrices of equal trace that was originally developed for Quantum Statistical Mechanics to study the dynamics of Quantum States (see Nielsen and Vidal [2001], and Sagawa [2020]). In a financial context, this ordering can be used to rank Correlation Matrices based on the amount of risk embedded in the assets they represent Fontanari et al. [2019]. Risk managers often use determinant-based, and physically inspired risk measures like the Frobenius-Norm, and Entropy, to compare correlation matrices. In Fontanari et al. [2019], it is shown that these measures are monotonic with respect to this Ordering and define a special sub-class of risk-measures: the $\mathcal{M}_\lambda$-Class Fontanari et al. [2019]. Thus, if we sequentially sample Correlation Matrices over sliding windows, we can track the temporal evolution of $\mathcal{M}_\lambda$ risk-functionals with respect to $\mathcal{S}$. If we take $\mathcal{S}$ to be the (say) DOW Jones Industrial Average, we are now, perhaps, in a position to build an alarm system for market crashes using the Quantum Lorenz Ordering (QLO) of financial correlation matrices Cirillo and Hüsler [2011]. In section 3.1, we introduce the central object used to define the Quantum Lorenz Ordering: the Quantum Lorenz Curve (QLC). and relate its interpretation to Markowitz' Modern Portfolio Theory (MPT) Markowitz [1952], and modern Random Matrix cleaning techniques Bun et al. [2017].

3.1 | The Quantum Lorenz Curve

Changing gears, we take a more general view of Financial Correlation Matrices. This perspective reveals the appealing mathematical properties of Correlation Matrices that make the Quantum Lorenz Curve such an interesting object to study. Here, we refer to a Correlation Matrix, in the most general sense, according to Axiom 3.1 and display sample Correlation and Covariance matrices, in Figure 3.1, for 25 constituents of the Dow Jones Industrial Average.



(a) Correlation Matrix ρ **(b)** Covariance Matrix Σ

Figure 3.1: DOW Jones Industrial Average Correlations

Correlation 3.1(a) and Covariance 3.1(b) Matrices for 27 of the 30 DJIA constituents.

Axiom 3.1 (General Properties of Correlation Matrices)**.** We refer to the set of $d \times d$ Correlation Correlation Matrices $\rho \in P(\mathcal{S})$ on a market $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_d\}$ with the following axioms:

- ρ is a Positive Semi-Definite Matrix on a Hilbert Space $\mathcal{H} = \mathbb{R}^d$, that is $\rho \in \mathrm{PSD}(\mathcal{H})$.
- ρ is a Hermitian Matrix on $\mathcal{H}$, that is $\rho = \rho^\dagger$.
- ρ neceassrily has trace d, with d necessarily $d = \dim(\mathcal{H})$, that is $\mathrm{Tr}[\rho] = d$,
- ρ has entries between -1 and 1 with 1's on the diagonal, that is $\rho_{i,j} \in [-1, 1]$ with $\rho_{i,i} = 1$.

By the Spectral Theorem of Hermitian Matrices, we know any Correlation Matrix $\rho \in P(\mathcal{S})$

has the following spectral decomposition

$$\rho = \sum_{k=1}^{d} \lambda_k(\rho) v_k v_k^T \tag{3.1}$$

with $\lambda(\rho) \coloneqq \{\lambda_k\}_{k=1}^d$ denoting the spectrum of ρ with eigenbasis $\{v_k\}_{k=1}^d$. Additionally, since ρ has trace $\mathrm{Tr}[\rho] = d$, we know that $\sum_{k=1}^d \lambda_k(\rho) = d$. Furthermore, since $\rho \in \mathrm{PSD}(\mathcal{H})$ is a Positive Semi-Definite Matrix, we also know $p_k \geqslant 0, \forall \lambda_k \in \lambda(\rho)$. Thus, the spectrum $\lambda(\rho)$ of a (Financial) Correlation Matrix $\rho \in P(\mathcal{S})$ defines a distribution on $\mathbb{R}^d$! However, once normalized, $\lambda(\rho)$ defines a probability distribution. This is why we choose the letter p to define the spectrum $\lambda(\rho) = \{\lambda_k\}_{k=1}^d$.

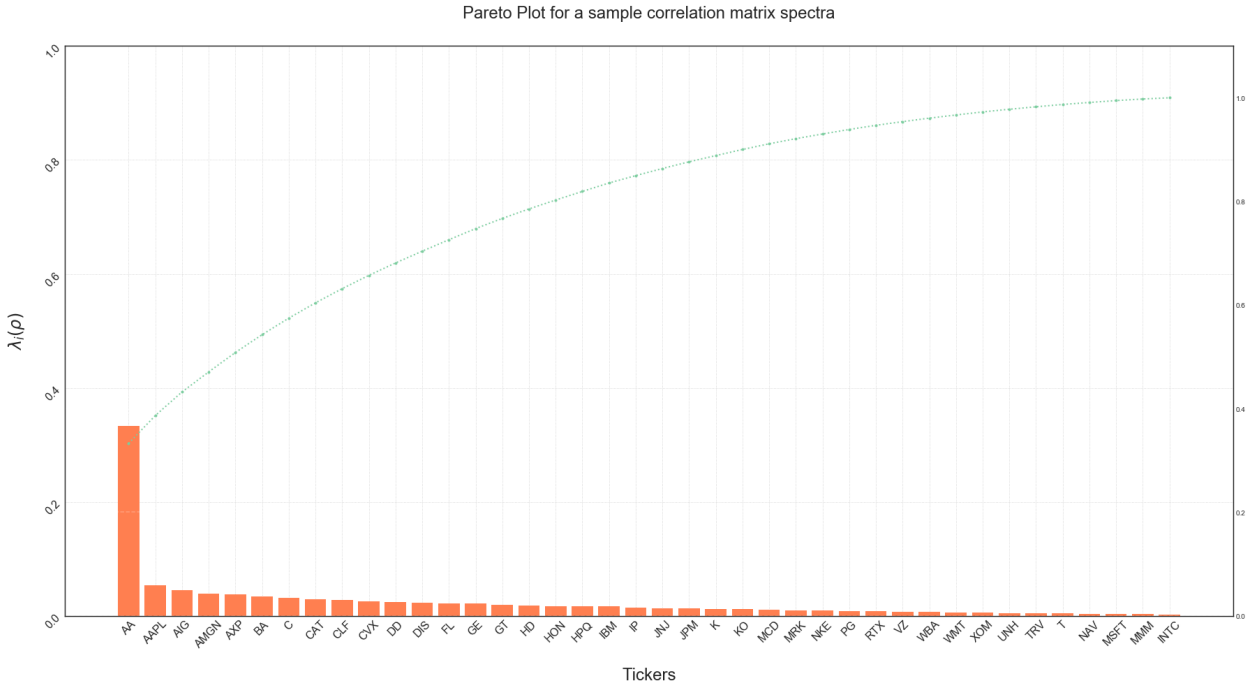


Figure 3.2: Pareto Chart for a Financial Correlation Matrix

Sorted Eigenvalues $\lambda_i^{\downarrow}(\rho)$ from 3.1(a) with the corresponding Quantum Lorenz Curve $L_\alpha(\rho)$.

The extent to which the dispersion of the distribution $\lambda(\rho)$ does indeed characterize a true heterogeneity , and underlying Systemic Risk in the Market $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_d\}$, a natural object to consider is the Cumulative Distribution Function (CDF) of $\lambda(\rho)$. We plot the example Spectrum $\lambda(\rho)$ from Figure 3.1 in Figure 3.2. In fact, this heterogeneity precisely corresponds to the so-called "Lorenz-Curve" of Lorenz [1905], which Max Otto Lorenz invented to study wealth distributions

of societies. When applied to the Spectra of Hermitian Matrices, and thus Financial Correlation Matrices, we get the Quantum Lorenz Curve in Defintion 3.2.

Definition 3.2 (Quantum Lorenz Curve)**.** Let A be a Hermitian Matrix on a Hilbert Space $\mathcal{H}$. The Quantum Lorenze Curve $L_\alpha(A)$, of order α, for A is defined as

$$L_\alpha(A) \coloneqq \frac{1}{\mathrm{Tr}[A]} \sum_{k=1}^{\alpha} \lambda_k^{\downarrow}(A), \qquad \forall \alpha \in \{1, \dots, d\} \tag{3.2}$$

with $\lambda_k^{\downarrow}(A) \in \lambda(A)$ denoting the sorted eigenvalues in descending order.

Definition 3.2 precisely corresponds to the red -line depicted in Figure 3.2. Furthermore, the Quantum Lorenz Curve, has an incredibly intuitive interpretation of risk: $L_\alpha(\rho)$ represents the total percentage portfolio variance explained by the top α influential assets (top α eigenvalues). This interpretation hints at the possibility of using the Quantum Lorenz Curve to order Correlation Matrices. Indeed, suppose $L_\alpha(\rho_2) \geqslant L_\alpha(\rho_1)$, $\forall \alpha \in \{1, \dots, d\}$. This would suggest that, $\forall \alpha \in \{1, \dots, d\}$, every rank-$\alpha$ approximation of ρ_2 is better than the corresponding rank-α approximation of ρ_1. In fact, this is exactly the definition of Quantum Majorization!

Definition 3.3 (Quantum Majorization)**.** Let A, B be Hermitian Matrices on a Hilbert Space $\mathcal{H}$. We say B Quantum Majorizes A, written as $B \succ A$, if, and only if,

$$L_\alpha(B) \geqslant L_\alpha(A), \qquad \forall \alpha \in \{1, \dots, d\} \tag{3.3}$$

with $L_\alpha(\cdot)$ denoting the Quantum Lorenz Curve.

Before calling Quantum Majorization an Ordering, it is important to verify a few properties. The reader is encouraged to see Fontanari et al. [2019] for all of the details, but we will glance over the main points with Axiom 3.4.

Axiom 3.4 (Order Conditions)**.** To guarantee financial interpret-ability, an Order on Financial Correlation Matrices should obey the following properties:

P1 *Minimal Element*: The least risky element is the Identity Matrix $\rho = I_d$

P2 *Maximal Element*: The most risky element is the set of rank-1 matrices $\rho = |\psi\rangle\langle\psi|$

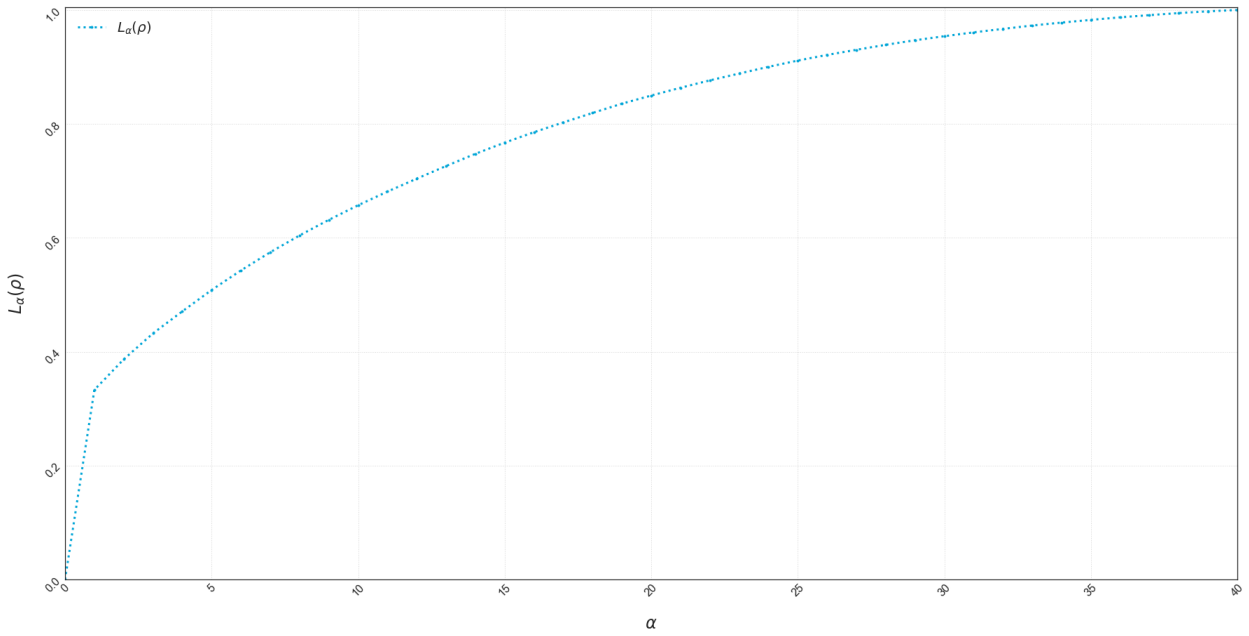


Figure 3.3: Pareto Chart for a Financial Correlation Matrix

Sorted eigenvalues $\lambda_i^{\downarrow}(\rho)$ from 3.1(a) with the corresponding Quantum Lorenz Curve $L_\alpha(\rho)$.

P3 *Monotonic*: The order should not increase in the rank of the Correlation Matrices.

P4 *Quasi-Convexity*: A convex combination of two Correlation Matrices should not be less risky than the riskiest of the originals.

Proposition 3.5. *Definition 3.3 satisfies properties* P1 — P4 *stated in Axiom 3.4.*

We state Proposition 3.5 without proof and refer the reader to Fontanari et al. [2019] for further details, but note that the proof is immediate from the definition of the Quantum Lorenz Curve 3.2.

Though it is not obvious why Definition 3.3 works, we refer the reader to Fontanari et al. [2019] for the proof. Additionally, we refer the reader to Fontanari et al. [2019], Arnold and Sarabia [2018], and Sagawa [2020] for complete discussions on the general theory of Majorization, and Quantum Majorization, respectively, including proofs and alternative definitions. For our purposes, this definition is clear, concise, and provides a nice geometric interpretation that is easily exploitable for characterizing risk, especially from the standpoint of Bouchaud and Potters [2009]. However, it is easy to see how the definition of Quantum Majorization simply applies the notion of (Classical) Majorization Arnold and Sarabia [2018] to the spectrum of Hermitian

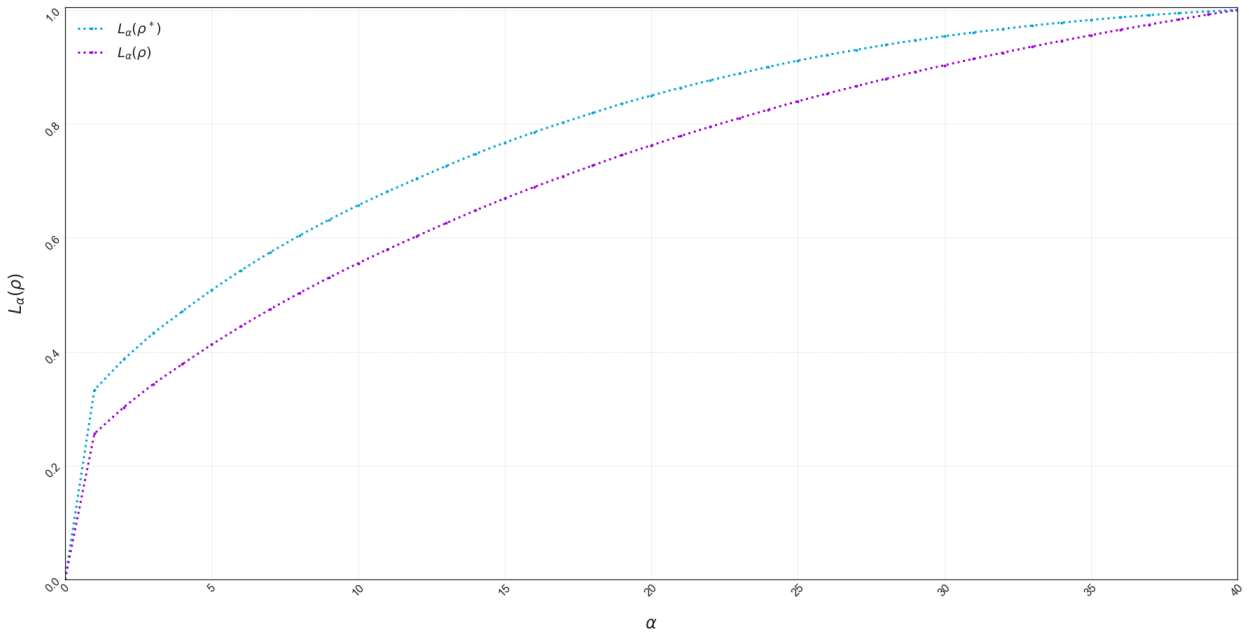


Figure 3.4: Quantum Lorenz Curve for a Quantum Majorization Relation

Quantum Lorenz Curves depicting a $\rho_l \succ \rho_k$ relation between $L_\alpha(\rho^*)$ and $L_\alpha(\rho)$.

Matrices, and thus Financial Correlation Matrices.

Indeed, since we consider Financial Correlation Matrices $\rho \in P(\mathcal{S})$ of the same dimension $d = \dim(\rho)$, the Quantum Lorenz Curve 3.2 of a Financial Correlation Matrix ρ is

$$L_\alpha(\rho) \coloneqq \frac{1}{d} \sum_{k=1}^{\alpha} \lambda_k^{\downarrow}(\rho) \tag{3.4}$$

with eigenvalues $\{\lambda_k^{\downarrow}\}$. Thus, we say a Financial Correlation Matrix ρ majorizes another $\tilde{\rho}$, that is $\rho \succ \tilde{\rho}$, if, and only if,

$$\frac{1}{d} \sum_{k=1}^{\alpha} \lambda_k^{\downarrow}(\rho_l) \geqslant \frac{1}{d} \sum_{k=1}^{\alpha} \lambda_k^{\downarrow}(\rho_k) \tag{3.5}$$

which precisely corresponds to the definition of the Classical Majorization of vectors $\vec{p}, \vec{q} \in \mathbb{R}^d$ Arnold and Sarabia [2018] since $\lambda(\rho_k)$, and $\lambda(\rho_l)$ are also vectors in $\mathbb{R}^d$, that is, $\lambda(\rho_k), \lambda(\rho_l) \in \mathbb{R}^d$. Thus, the Quantum Majorization of Hermitian Matrices (of equal trace) corresponds to the Classical Majorization of their respective eigenvalues.

3.2 | The Quantum Majorization Graph

The question becomes, how can we use this characterization of Risk, the Quantum Lorenz Ordering, to measure Systemic Risk. Quantum Majorization allows us to compare the Risk embedded in two different Correlation Matrices, of the same size, but how can this characterize Systemic Risk in the Market. To take advantage of the Quantum Lorenz Ordering, Fontanari et al. [2019] sequentially sampled Correlation Matrices, for the constituents of major Financial Indices, over sliding windows. For convenience, we capture this with Definition 3.6 below.

Definition 3.6 (Temporal State Space of Financial Correlations)**.** For a Market of d financial assets $\mathcal{S} \coloneqq \{\mathcal{S}_1, \ldots, \mathcal{S}_d\}$, and a sliding-window $\Delta = (h, m, I)$, corresponding to the time-interval $I = [t_0, T]$, we define the Temporal State Space of Financial Correlations $\mathcal{P}_\Delta(\mathcal{S}; I)$ as

$$\mathcal{P}_\Delta(\mathcal{S}; I) \coloneqq \{\rho_k = \rho_{\iota_k}(\mathcal{S}) : \iota_k = [kh, kh + m] \in I, \forall k \in \{0, \ldots, n-1\}\} \tag{3.6}$$

where $n = |\mathcal{P}_\Delta(\mathcal{S})|$ is the cardinality of $\mathcal{P}_\Delta(\mathcal{S})$ and is explicitly calculable: $n = \left\lfloor \frac{T}{h} \right\rfloor - \left\lfloor \frac{m-h}{h} \right\rfloor$.

Example 3.7. Suppose we want to sample correlation matrices for the DOW Jones Industrial Average constituents over the $T = 252$-day trading period for the year 2019. We can define a Sliding-Window $\Delta = (10, 100, [0, 252])$ with $m = 100$-day windows and $h = 10$-day shifts. Then ρ_0 would correspond to the time-interval $\iota_0 = [0, 100]$, and ρ_1 to $\iota_1 = [10, 110]$, so on so forth, until we have a collection of $n = 16$ Financial Correlation Matrices $\{\rho_{\iota_k}\}_{k=0}^{n-1} = \mathcal{P}_\Delta(\mathcal{S}) \in \mathcal{P}(\mathcal{S})$.

The Temporal State-Space $\mathcal{P}_\Delta(\mathcal{S}; I)$ creates a temporal sequence of Correlation Matrices that can be used to track Systemic Risk when those Correlation Matrices represent the constituents $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_d\}$ of a major Financial Index, such as the Dow Jones. Using this collection, we can build the Quantum Lorenz Ordering by checking which Correlation Matrices $\rho_i, \rho_j \in \mathcal{P}_\Delta(\mathcal{S}; I)$ Quantum Majorize each other. Fontanari et al. [2019] embedded the Quantum Lorenz Ordering into a $\{0, 1\}^{n \times n}$ Matrix for $n = |\mathcal{P}_\Delta(\mathcal{S}; I)|$ the number of Correlation Matrices in $\mathcal{P}_\Delta(\mathcal{S}; I)$. We introduce the so-called Quantum Majorization Matrix next with Definition 3.8.

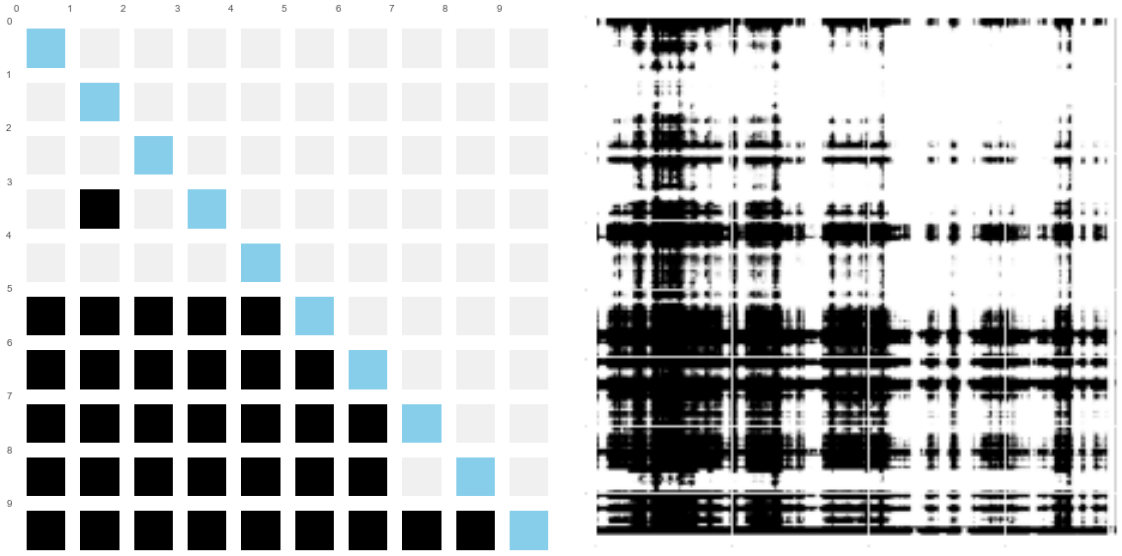
Definition 3.8 (Quantum Majorization Matrix)**.** Given a Temporal State Space $\mathcal{P}_\Delta(\mathcal{S}; I)$, the so-

called Quantum Majorization Matrix $\Psi = \Psi(P_\Delta(S))$ has entries

$$\Psi_{k,l} \coloneqq \begin{cases} 1, & \text{if } \rho_l \succ \rho_k \quad \forall \rho_k, \rho_l \in P_\Delta(\mathcal{S}) \\ 0, & \text{otherwise} \end{cases} \tag{3.7}$$

We display the Quantum Majorization Matrix, for the Quantum Lorenz Ordering, built using the 30 constituents of the Dow Jones sampled between January 1, 1990 until December 31, 2020 using 100-day Windows, and 10-day Shifts in Figure 3.5(b). The thick horizontal bands in the Quantum Majorization Matrix of Figure 3.5(b) correspond to major Stock Market Events. For an in depth analysis on this object, we direct the reader to Fontanari et al. [2019] for all the details. In short, the reader can check, by definition 3.8, that Correlation Matrices in these bands Quantum Majorize more Correlation Matrices than other Correlation Matrices Quantum Majorize them. Hence, they are "Riskier". Referring to the Quantum Majorization Matrix on the left, in Figure 3.5(b), the black cell in position $\{3, 1\}$ states that Correlation Matrix $\rho_3 \succ \rho_1$.

Fontanari et al. [2019] then used a directed acyclic graph (DAG) to represent the QLO aptly named, the Quantum Majorization Graph (QMG). Representing this information in a graph has many advantages as graphs have become very popular Data-Structures in the study of complex networks. We introduce the QMG next.



(a) Jan 1, 1990 - Nov 1, 1990 **(b)** Jan 1, 1990 - Dec 6, 2020

Figure 3.5: Sample Quantum Majorization Graphs

Quantum Majorization Graphs with **black**-entries representing some majorization relation $\rho_l \succ \rho_k$.

Definition 3.9 (Quantum Majorization Graph)**.** Given a Temporal State Space $\mathrm{P}_{\Delta}(\mathcal{S};\mathrm{I})$, the so-called Quantum Majorization Graph is the ordered-pair $\mathrm{G} = (V, A)$ where

- V is the set of nodes $\{v_i\}_{i=1}^{d} \in V$ corresponding to $\{\rho_i\}_{i=1}^{d} \in \mathrm{P}_{\Delta}(\mathcal{S};\mathrm{I})$
- A is the Adjacency Matrix $A = \Psi - \mathrm{I}_n$ corresponding to the QMM Ψ.

It is easy to see why the QMG $\mathrm{G} = (V, A)$ is a DAG. Indeed, since each edge represents an ordering between two correlation Matrices $\rho_i \succ \rho_j$, each edge will be directed. Secondly, since the QLO is transitive, cycles cannot exist. Hence, $\mathrm{G} = (V, A)$ is directed and acyclic. For this reason, using a DAG to represent the QLO feels natural. We display a sample QMG over the same time period as that of the QMM of figure 3.5(a) with figure 3.6. For a more detailed analysis, we encourage the reader to see Fontanari et al. [2019]. In short, the reader can verify in the two Figures that thick horizontal bands in the QMM Ψ correspond to correlation matrices with many directed-edges coming out of them in the QMG $\mathrm{G} = (V, A)$.

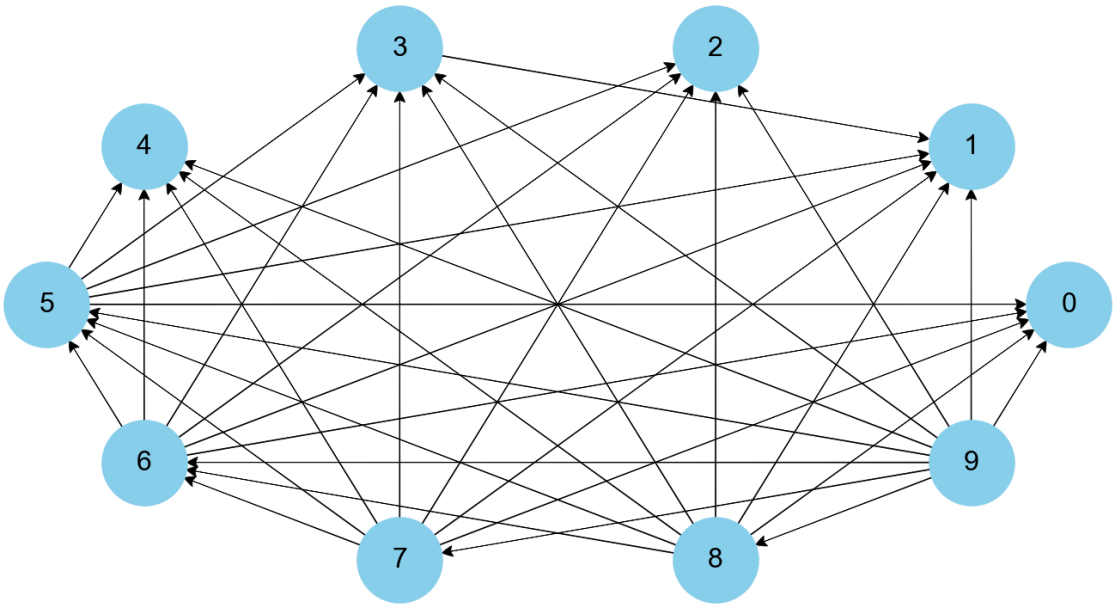


Figure 3.6: Pareto Chart for a Financial Correlation Matrix

Sorted Eigenvalues $\lambda_i^{\downarrow}(\rho)$ from 3.1(a) against the induced CDF $\sum \lambda_i^{\downarrow}(\rho)$.

3.2.1 | Two Simple Risk-Measures on $G = (V, A)$

Given these two mathematical objects representing the QLO, we want to see if we can measure systemic risk using these two objects: the QMM Ψ and QMG $G = (V, A)$. Fontanari et al. [2019] introduced two such measures. For all the details, we refer the reader to Fontanari et al. [2019]. For our purposes, however, it suffices to Define these two measures and briefly discuss their relative interpretations. Here, we introduce the "Degree Centrality" of a correlation matrix $\rho_i \in P_\Delta(\mathcal{S}; I)$, and the "$\Psi$-Density" of the QMM.

Definition 3.10 (Degree Difference: $\theta(\rho_i)$)**.** Given a QMG $G = (V, A)$, the Degree Centrality of a correlation matrix $\rho_i \in P_\Delta(\mathcal{S}; I)$ is defined as

$$\theta(\rho_i) = \frac{1}{2} + \frac{1}{2n}(deg^+(v_i) - deg^-(v_i)) \in [0, 1] \tag{3.8}$$

where $deg^+(v_i)$ and $deg^-(v_i)$ denote the number of outgoing, and incoming, edges for the node $v_i \in V$ corresponding to the correlation matrix ρ_i, respectively.

For instance, we refer to the $G = (V, A)$ depicted in figure 3.6. Here, we can easily see that $\theta(\rho_9) = 1$ while $\theta(\rho_0) = \frac{2}{10}$. As such, ρ_9 represents a time-interval $\iota_9 = [90, 190]$ that is much riskier in the market than that of ρ_0 over $\iota_0 = [0, 100]$. Hence, we'd expect the market to be more more turbulent for times $\forall t \in [90, 190]$ than $\forall t \in [0, 100]$.

It's easy to see when $\theta(\rho_i)$ takes the two extreme values: 0 and 1. Indeed, if ρ_i quantum majorizes every other correlation matrix $\forall \rho_j \in P_\Delta(\mathcal{S}; I)$, then $\theta(\rho_i) = 1$. On the other hand, if ρ_i is quantum majorized by every other correlation matrix $\forall \rho_j \in P_\Delta(\mathcal{S}; I)$, then $\theta(\rho_i) = 0$. Hence, the $\frac{1}{2} + \frac{1}{2n}$ term is simply a normalization constant. If one correlation Matrix were to be riskier than another, we'd expect the degree centrality to be greater than the other. Furthermore, it turns out that the degree centrality belongs to a much broader class of centrality measures in the study of complex networks. Thus, any centrality measure found in the literature can (in theory) be used on the Graph $G = (V, A)$ and be given an interpretation based on risk. Lastly, we have the Ψ-Density $\Upsilon(\Psi)$ of the quantum majorization matrix Ψ. We introduce this below with definition 3.11.

Definition 3.11 (Ψ-Density: $\Upsilon(\Psi)$)**.** Given a Quantum Majorization Matrix (QMM) $\Psi = \Psi(P_\Delta(\mathcal{S}; \iota))$,

we define the Ψ-Density $\Upsilon(\Psi)$ with

$$\Upsilon(\Psi) = \frac{2\sum_{i=1}^{m}\sum_{i\neq j}^{m}\Psi_{i,j}}{m(m-1)} \in [0,1] \tag{3.9}$$

on any (sub)interval $\iota \in I$ where $\forall m \leqslant n$ denotes the Cardinality $m = |P_{\Delta}(\mathcal{S};\iota)|$ of the $P_{\Delta}(\mathcal{S};\iota) \in P_{\Delta}(\mathcal{S};I)$.

We introduce the notation $P_{\Delta}(\mathcal{S};\iota)$ in the above definition to stress that the Ψ-Density can be used to track systemic risk in a temporal fashion, over sliding windows, just like the correlation matrices themselves. Furthermore, the Ψ-density is normalized to the $[0,1]$ range via the constant $2/m(m-1)$. For instance, the Ψ-Density $\Upsilon(\Psi)$ for the QMM depicted in Figure 3.5(a) is equal to $30/40 = 0.75$. Indeed, higher levels of $\Upsilon(\Psi)$ indicate time-periods where more quantum majorization is present. We'd expect this to mean that the market is riskier over those time-intervals. We encourage the reader to see Fontanari et al. [2019] for more details.

3.3 | An Alarm-System using the Quantum Lorenz Order

Fontanari et al. [2019] applied spectral clustering to the QMG $G = (V, A)$ in an attempt to cluster correlation matrices according to their relative positions on $G = (V, A)$. The idea is that correlation matrices, with similar risk profiles, should belong to similar "communities", in analogue with social networks. In fact, spectral clustering is a very popular algorithm for community detection in the study of complex networks, in general. Spectral clustering relies on the spectral decomposition of the graph Laplacian $L(G)$ associated the graph $G = (V, A)$ of interest, the QMG in our case. Before moving forward, let us define the graph Laplacian for a directed graph.

Definition 3.12 (Directed Graph Laplacian)**.** The Laplacian matrix of a directed graph $G = (V, A)$, of cardinality $n = |V|$, is defined as

$$L \coloneqq I_n - \frac{1}{2}(\Phi^{\frac{1}{2}} P \Phi^{-\frac{1}{2}} + \Phi^{\frac{1}{2}} P^{\intercal} \Phi^{-\frac{1}{2}}) \tag{3.10}$$

where I_n is the $n \times n$ Identity Matrix, P is the "Transition Matrix" of the graph, and Φ is a matrix such that $diag(\Phi) = \phi(P)$, $0^{'s}$ elsewhere, where $\phi(P)$ denotes the "Perron Vector" of P.

For our purposes, the precise definitions of the transition matrix P and perron vector $\phi(P)$ are unnecessary. The curious reader can easily find details in the literature. The important thing

to understand is that we can use the spectral decomposition of the Graph Laplacian $L = L(G)$

$$L = \sum_{i=1}^{n} \lambda_i(L) v_i v_i^T \tag{3.11}$$

to cluster each node $v_i \in V$ into an arbitrary number of k-Clusters, the principle advantage to other clustering algorithms that don't enforce a specific number of clusters. The spectral clustering algorithm follows a relatively simple procedure, which we outline next.

Definition 3.13. The spectral clustering algorithm, given a Graph $G = (V, A)$ of Cardinality $n = |V|$, with a specified number of k-Clusters, follows these steps:

1. Compute the graph Laplacian L=L(G) and its associated Spectral Decomposition

$$L = \sum_{i=1}^{n} \lambda_i^{\downarrow}(L) v_i^{\downarrow} v_i^{\downarrow T}. \tag{3.12}$$

   with eigenvalues $\lambda^{\downarrow}(\rho)$ and eigenbasis $\{v_i^{\downarrow}\}_{i=1}^{n}$ sorted in descending order.

2. Identify the k eigenvectors $\{v_{(n-k+1)}^{\downarrow}, v_{(n-k+2)}^{\downarrow}, \ldots, v_i\}$ associated to the k-smallest Eigenvalues $\lambda_{n-k+1}^{\downarrow}(L), \ldots, \lambda_n^{\downarrow}(L)$.

3. Stack the k eigenvectors into a $n \times k$ Matrix, with each column being one of the $n \times 1$ Eigenvectors.

4. Run the k-means clustering algorithm on this $n \times k$ column matrix.

This procedure will cluster each node $v \in V$ into one of k-clusters based on their position in the k-(lower)dimensional spectral projection.

Fontanari et al. [2019] showed that this approach can cleanly cluster each correlation matrix $\rho_i \in P_{\Delta}(8; I)$ into $k = \{2, 3\}$-Clusters corresponding to the degree centrality $\theta(\rho_i)$. The author(s) argued that more clusters are unnecessary. Hence, by sampling correlation matrices, in a temporal fashion, over sliding windows, we can assign each new correlation matrix to a cluster using the $\theta(\rho_i)$ risk measure at each iteration. We depict such a clustering corresponding to Tail-Dependence matrices in figure 3.7.

Surprisingly, this procedure seems to assign cluster membership quite accurately, according to periods of sever market turbulence. For instance, notice the time-period 2007-2008 where red

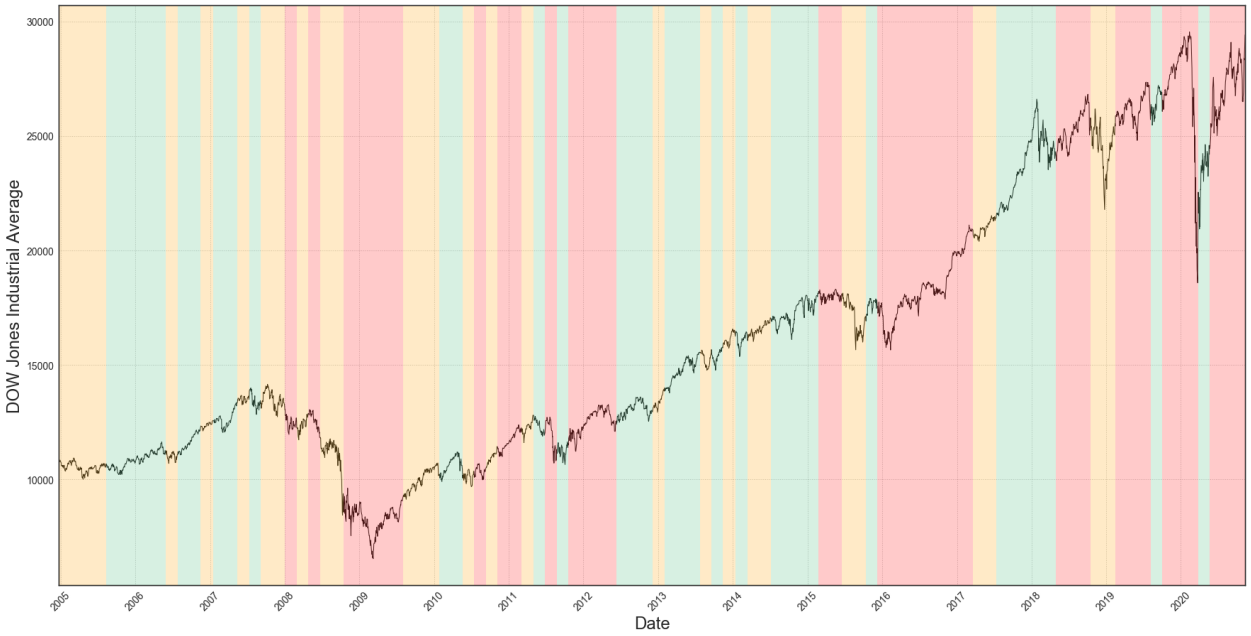


Figure 3.7: Naive Alarm System

$\theta(\rho_i)$ clustered according to ρ_i, ρ_i, and ρ_i plotted with the Dow Jones Industrial Avaerage Index from Jan 1 2005, until Dec 31 2020.

clusters are very dense. On the other hand, the subsequent rally is very green, and hence "safe(r)". Something similar can be said about the late 2000's, 2018, 2019, and 2020 where the Market was known to be turbulent. Furthermore, periods of dense red clusters are all preceded by periods of yellow clusters, a crucial feature for any alarm system. The question thus becomes, can we use this naive alarm system as the basis for a more advanced methodology. A natural candidate is to utilize the so-called "Reinforced Urn Process" (RUP) of Cirillo and Hüsler [2011] by modelling these cluster memberships as the sampling of coloured marbled out of a state space of Polya Urns. We explore this topic in the next section.

4

Urn-Based Alarm Systems

When considering some natural, or even man-made (for that matter), phenomenon, it is often vital to predict whether the variable of interest will surpass a critical threshold sometime in the (near) future. Examples that have (potentially) catastrophic impacts on human life include, black-outs, floods, epidemics, and stock market crashes. In Antunes M and FK [2003] and de Mare [1980], rigorous and generalized bayesian alarm systems have been developed with interesting applications to infectious disease outbreaks and gaussian processes, respectively. In Section 4.1, we will briefly introduce the bayesian framework that was developed in Antunes M and FK [2003] and de Mare [1980]. In section 4.4.1 will go over a novel approach, taken in Cirillo and Hüsler [2011], to building Bayesian Alarm Systems using a Polya-like urn model for solar sunspot data.

4.1 | Alarm Systems

Many times, the temporal evolution of the variable of interest can be represented by a discrete parameter stochastic process (or Time Series), such as $\vec{X} = \{X_0, \ldots, X_T\}$. To formalize this idea, we can break the time series $\vec{X}$ up into 3 different sections: the "past" $\rightarrow \vec{X}_1$, "present" $\rightarrow \vec{X}_2$ and "future" $\rightarrow \vec{X}_3$. After choosing some $s \geqslant 0$, we can break the time sequence $\{0, \ldots, t-1, t, t+1, \ldots, T\}$ up, and define $\vec{X}_1 = \{X_0, \ldots, X_{t-s}\}$, $\vec{X}_2 = \{X_{t-s+1}, \ldots, X_t\}$, and $\vec{X}_3 = \{X_{t+1}, \ldots, X_T\}$: the "past", "present", and "future", respectively. Intuitively, a "Catastrophe" is considered to be an event (albeit small and rare in probability) in the σ-algebra generated by the "future", that is $\sigma(\vec{X}_3)$. More formally, we define the notion of a "Catastrophe" with Definition 4.1.

Definition 4.1 (Catastrophe). An event $\mathcal{C}_{t+r} \in \sigma(\vec{X}_3)$ is a "catastrophe" if $\mathbb{P}[\mathcal{C}_{t+r}|\vec{X}_1, \vec{X}_2] \leqslant \eta$ for a small enough η and $r \geqslant 1$.

Remark 4.2 (What is a Catastrophe?). In practice, η is simply an abstraction that helps characterize the sort of phenomenon we are dealing with.

When developing an alarm system, the objective is to construct an "event predictor" (detection region) so that, whenever the process ($\vec{X}$) enters the region, an "Alarm" is cast. In Antunes M and FK [2003], the so-called "Operating Characteristics" of an alarm system were constructed. We begin introducing these characteristics with definitions 4.3 and 4.4 below.

Definition 4.3 (Alarm). An event $\mathcal{A}_t \in \sigma(\vec{X}_2)$ is an "alarm" for a "catastrophe" $\mathcal{C}_{t+j} \in \sigma(\vec{X}_3)$ if $\mathbb{P}[\mathcal{C}_{t+j}|\mathcal{A}_t] \geqslant \varepsilon$ for some threshold $\varepsilon \in [0, 1]$ in j time-steps into the future.

Definition 4.4 (Alarm Size). The "Alarm" $\mathcal{A}_t \in \sigma(\vec{X}_2)$ has "Size" $\mathbb{P}[\mathcal{A}_t|\vec{X}_1] = \alpha_t$ and "Detection Probability" $\mathbb{P}[\mathcal{C}_{t+j}|\mathcal{A}_t, \vec{X}_1] = \chi_t$.

Intuitively, an Alarm System should be "Optimal" if it outperforms all other possible Alarm Systems of equal "size" $\mathbb{P}[\tilde{A}|\vec{X}_2] = \alpha_t$. Indeed, we borrow from Antunes M and FK [2003] and define this "Optimality" condition with Definition 4.5 below.

Definition 4.5 (Alarm Optimality). The "Alarm" $\mathcal{A}_t \in \sigma(\vec{X}_2)$ is said to be "optimal" of "size" α_t if,

$$\mathbb{P}[\mathcal{C}_{t+j}|\mathcal{A}_t, \vec{X}_1] = \sup_{\tilde{\mathcal{A}}_t \in \sigma(\vec{X}_2)} \mathbb{P}[\mathcal{C}_{t+j}|\tilde{\mathcal{A}}_t, \vec{X}_1] \tag{4.1}$$

where the supremum is taken over all subsets $\tilde{\mathcal{A}}_t \in \sigma(\vec{X}_2)$ such that $\mathbb{P}[\tilde{\mathcal{A}}_t|\vec{X}_2] = \alpha_t$.

There are several other quantities of interest associated to an Alarm System. We will borrow the so-called "Operating Characteristics" from Antunes M and FK [2003], further elaborating on Definitions 4.3 and 4.4 with definition 4.6.

Definition 4.6 (Operating Characteristics of an Alarm System). Operating Characterstics

O_C1 *Alarm Size*: $\mathbb{P}[\mathcal{A}_t|\vec{X}_1]$

O_C2 *Probability of Detecting a Catastrophe*: $\mathbb{P}[\mathcal{C}_{t+j}|\mathcal{A}_t, \vec{X}_1]$

O_C3 *Probability of a Correct Alarm*: $\mathbb{P}[\mathcal{A}_t|\mathcal{C}_{t+j}]$

O_C4 *Probability of a False Alarm*: $\mathbb{P}[\mathcal{A}_t|\mathcal{C}^c_{t+j}]$

O_C5 *Probability of an Undetected Catastrophe*: $\mathbb{P}[\mathcal{C}_{t+j}|\mathcal{A}^c_t, \vec{X}_1]$

Thinking in terms of a confusion matrix, Definition 4.6 contains all of the information one could possibly want to know about an alarm system. Indeed, O_C1 corresponds to a "true negative", O_C2 corresponds to a "true positive", O_C3 a "False-Positive (Type-I Error)", and O_C4 a "False-Negative (Type-II Error)". Intuitively, we want an alarm system, among other things, to have a $\mathbb{P}[\mathcal{A}_t|\mathcal{C}_{t+j}] \to 1$ as close to 1 as possible. On the other hand, we want to minimize the number of "false alarms", i.e. $\mathbb{P}[\mathcal{A}_t|\mathcal{C}^c_{t+j}] \to 0$. To address this, we state the following Theorem 4.7, without proof. For the proof, we direct the reader to de Mare [1980].

Theorem 4.7 (Optimal Alarm System)**.** *An Alarm System* $\vec{\mathcal{A}} := \{\mathcal{A}_1, \ldots, \mathcal{A}_r\}$ *is optimal of "Size"* $\vec{\alpha} := \{\alpha_1, \ldots, \alpha_r\}$ *if,*

$$\mathcal{A}_t = \{X_2 \in \vec{X}_2 : \frac{\mathbb{P}[\mathcal{C}_t|X_2, \vec{X}_1]}{\mathbb{P}[\mathcal{C}_t|\vec{X}_1]} \leqslant \zeta_t\} \tag{4.2}$$

where ζ_t *is taken such that* $\mathbb{P}[\mathcal{A}_t|\vec{X}_1] = \alpha_t$.

This rather simple bayesian framework for catastrophe (event) prediction is quite robust, and allows us to build alarm systems for a wide variety of statistical models. Again, for an application of this construction to the detection of Infectious Disease Outbreaks, see Antunes M and FK [2003]. For an application to Gaussian Processes, see de Mare [1980]. In fact, in Section 4.4.1, we will see how we can exploit Polya-like Urn Models for Event Detection using the composition of Urns to build Bayesian priors to construct an Alarm System and evaluate its accuracy using the Operating Characteristics from 4.6.

4.2 | Reinforced Random Walk

Now that we have a Bayesian framework for Event Prediction, we must ask: What kind of process do we want $\vec{X}$ to follow? A natural candidate is to consider the so-called "Reinforced Random Walk", or RRW, for short. The reason for using this class of Stochastic Processes for Event Prediction is simple: A RRW has the tendency to "circle back" to familiar territory. To see how this works consider an Integer-Valued Stochastic Process $\vec{X}$ defined on a State-Space S with elements

$$\vec{X} \coloneqq \{X_0 = (0, l_0), X_1 = (1, l_1), \dots, X_n = (n, l_n)\}, \qquad n \in \mathbb{Z}^+, l \in \mathbb{Z} \tag{4.3}$$

such that each tuple $s = (n, l)$ is an element of the State-Space $s \in S$. If we associate to each "Interval" $\{(n, l), (n, l+1)\}$ a "Weight" $w_{(n,l)}$, we can recursively update $w_{(n,l)} = w_{(0,l)} + \sum_{k=1}^{m} a_k$ when $\vec{X}$ has "crossed" $\{(n, l), (n, l+1)\}$ some number of m-times where $w_{(0,l)}$, and each a_k, are an arbitrary initialization, and "reinforcement" parameter, respectively. In this way, whenever $X_n = (n, l)$ ends up "crossing" into $X_{n+1} = (n+1, l+1)$ it becomes more and more likely that $\vec{X}$ will do so again in the future since the "Weight" $w_{(n,l)}$ associated to the "Interval" $\{(n, l), (n, l+1)\}$ keeps getting updated. In other words, the Probability $\mathbb{P}[X_{n+1} = (n+1, l \pm 1)|\vec{X}_n]$ that X_{n+1} will be in state $s = (n+1, l \pm 1)$ will somehow be proportional to the "Weights" $w_{(n,l-1)}$, and $w_{(n,l)}$, associated to the "Intervals" $\{(n, l-1), (n, l)\}$, and $\{(n, l), (n, l+1)\}$, respectively. Up until now, we have omitted a few technical details that fully characterize the RRW. Before moving forward, let us go ahead and formally define the RRW with definition 4.8.

Definition 4.8 (Reinforced Random Walk)**.** A RRW, defined on a State-Space S, is a sequence of integer-valued Random Variables $\vec{X} = \{X_n = s : s = (n, l) \in S\}$, with $n \in \mathbb{Z}^+$ and $l \in \mathbb{Z}$, whose transition probabilities $p(l, l+1)$ are equal to

$$\begin{aligned} p(l, l+1) &= \mathbb{P}[X_{n+1} = (n+1, l+1)|\vec{X}_n] \\ &= 1 - \mathbb{P}[X_{n+1} = (n+1, l-1)|\vec{X}_n] \\ &= \frac{w_{(n,l)}}{w_{(n,l)} + w_{(n,l-1)}} \end{aligned} \tag{4.4}$$

where $\vec{X}_n$ denotes the history of the process $\vec{X}$, up until time n, such that

$$w_{(n,l)} = w_{(0,l)} + \sum_{k=1}^{m} \alpha_k \qquad \& \qquad w_{(n+1,l)} - w_{(n,l)} \geqslant 0 \tag{4.5}$$

with equality when (X_n, X_{n+1}) are neither $\{(n, l), (n+1, l+1)\}$ or $\{(n, l+1), (n+1, l)\}$ while m denotes the number of times $\vec{X}$ "crossed" from $(n, l) \to (n, l+1)$ with Hyper-Parameters $w_{(0,l)}$ and α_k.

It is easy to verify that such a Process $\vec{X}$, as defined above in definition 4.8, is a markovian since $p(l, l \pm 1) \geqslant 0$ and $p(l, l+1) + p(l, l-1) = 1$. More interestingly, however, Diaconis Diaconis P [1980] showed that, if we take $\alpha_k = 1$, such a Process $\vec{X}$ is equivalent to equipping every State

in the State-Space $s \in S$ with an independent Polya urn $U(s)$ where each $w_{(n,l)}$ and $w_{(n,l-1)}$ correspond to the number of, say, Black and Red marbles $\mathcal{B}_0$ and $\mathcal{R}_0$ in $U(s)$. Thus, the probability that $X_n = (n, l)$ moves to $X_{n+1} = (n+1, l \pm 1)$ can be thought of as the probability of sampling a certain colored marble from the Urn $U(s)$. We'll go more into the details of the Polya urn construction next, in Section 4.3, before moving onto the more general Reinforced Urn Process (RUP) P. Muliere [2000] in Section 4.4.

4.3 | Polya Urn Models

Polya urn models have a long-standing history in the study of Probability and Statistics. The construction proposed by Polya HM [2009] is simple, yet ingenious. Consider an Urn with an initial composition containing $\mathcal{B}_0$ and $\mathcal{R}_0$ number of Black, and Red, marbles, respectively. For every time $t = 1, 2, 3, \ldots$ we draw a marble from the Urn and update the composition of the Urn according to some rule that characterizes the Urn Model. This setup generates a sequence of Random Variables $\{X_t\}$, either 0 or 1, depending on whether a Black or Red marble was drawn, respectively, at time t. If we let $P_{\mathcal{B}_t} = \frac{\mathcal{B}_t}{\mathcal{B}_t + \mathcal{R}_t}$ be the proportion of Black marbles in the Urn U at time t, then X_{t+1} has distribution,

$$X_{t+1} \sim \text{Bern}(P_{\mathcal{B}_t}) \tag{4.6}$$

for all $t \geqslant 0$ where $\text{Bern}(P_{\mathcal{B}_t})$ denotes the Bernoulli Distribution. In Polya's original construction HM [2009], the Urn was "reinforced" with $\delta \geqslant 0$ number of Black, or Red, marbles whenever a marble of that color is drawn. Mathematically, if we define the Composition $C_t = C_t(U)$ of the Urn U at time t with $C_t = \{\mathcal{B}_t, \mathcal{R}_t\}$, containing $\mathcal{B}_t$ Black marbles and $\mathcal{R}_t$ Red marbles, we have

$$C_{t+1} = \begin{cases} \{\mathcal{B}_{t+1} + \delta, \mathcal{R}_{t+1}\}, \text{with Probability } P_{\mathcal{B}_t} \\ \{\mathcal{B}_{t+1}, \mathcal{R}_{t+1} + \delta\}, \text{with Probability } 1 - P_{\mathcal{B}_t} \end{cases} \tag{4.7}$$

since $X_{t+1} \sim \text{Bern}(P_{\mathcal{B}_t}) = P_{\mathcal{B}_t}^{X_t}(1 - P_{\mathcal{B}_t})^{X_t}$ and $X_t \in \{0, 1\}$. Hence, in keeping with the analogy to RRW's alluded to in Section 4.3, Polya's Urn model can replace the Weight-Interval updating

mechanism used to construct the RRW. Here, we can replace the weight updating mechanism

$$\begin{aligned} w_{(n,l)} = w_{(0,l)} + \sum_{k=1}^{m} a_k \quad &\rightarrow \quad \mathcal{B}_n = \mathcal{B}_0 + \sum_{k=1}^{m} \delta \\ w_{(n,l-1)} = w_{(0,l-1)} + \sum_{k=1}^{m} a_k \quad &\rightarrow \quad \mathcal{R}_n = \mathcal{R}_0 + \sum_{k=1}^{m} \delta \end{aligned} \tag{4.8}$$

meaning the Transition Probability $p(l, l+1)$, in defintion 4.8 of the RRW, can be equivalently expressed as

$$\begin{aligned} p(l, l+1) &= \mathbb{P}[X_{n+1} = (n+1, l+1) | \vec{X}_n] \\ &= 1 - P_{\mathcal{R}_n} \\ &= \frac{\mathcal{B}_n}{\mathcal{B}_n + \mathcal{R}_n} \end{aligned} \tag{4.9}$$

completing the analogy. The advantage of the approach using Urns to characterize Transition Probabilities between States of a RRW are two-fold:

1. *Inter-pretability*: In practice these models are easier to visualize and understand.
2. *Generalized Reinforcement Schemes*: This allows us to use a plethora of Urn reinforcement schemes that would complicate the formulation of the traditional RRW.

This concludes this Section on Polya's urn Model. We move onto the so-called "Reinforced Urn Process" next.

4.4 | Reinforced Urn Process

Thanks to P. Muliere [2000], we can define a Reinforced Random Walk on a countable State-Space of Urns: A "Reinforced Urn Process", or RUP for short. Reinforced Random Walks were introduced earlier to model the tendency of a Random Walker to return to past locations. The precise definition of a RRW relies on the transition probabilities between States in the State-Space $s \in S$. The ingenuity of Polya's Urn construction allows for a simple, yet robust, mechanism for calculating transition probabilities between two states $s, \bar{s} \in S$ in the State-Space S by viewing the Weights associated to any two states $w_s, w_{\bar{s}}$ as the proportion of colored marbles in an Urn $U(s)$. Hence, by endowing each state $s \in S$ with an Urn, say $U(s)$, we can construct a RRW and use Urns to compute transition probabilities "nonparametrically" as we observe the evolution of $\vec{X}$ in an "online" fashion. The RUP specification relies on four elements.

Definition 4.9 (RUP Specification)**.** The RUP specification relies on the four following elements.

E1: A countable State-Space S with elements $s \in S$.

E2: A finite set of Colours C with colors. $c = (c_1, \ldots, c_\kappa) \in C$ of cardinality $\kappa \geqslant 1$.

E3: An Urn "Composition Function" $U(s)$ that maps S to the set of κ-tuples of non-negative real numbers whose sum is a strictly positive number.

E4: A "Rule of Motion" RoM $d : S \times C \to S$ that specifies the state transitions for the RRW.

Each $s \in S$ comes equipped with its own Urn $U(s)$ containing $m_s(c) \geqslant 0$ marbles for each colour $c \in C$. Without loss of generality, the RoM satisfies the additional condition that there exists no more than one colour $c = c(s, \bar{s})$ such that $d(s, c(s.\bar{s})) = \bar{s}$ for any two $s, \bar{s} \in S$. We are now ready to full specify the behavior of the RUP with Definition 4.10.

Definition 4.10 (Reinforced Urn Process)**.** A RRW $\vec{X}$ on S starting at state $s_0 \in S$ is defined recursively:

1. Initialize $\vec{X}$ with $X_0 = s_0$.
2. For $X_{n-1} = s \in S$, when $\forall n \geqslant 1$ a marble of colour c is drawn from $U(s)$ and is then "reinforced" with $r(s)$ marbles of that colour c, i.e. $m_s(c) \to m_s(c) + r(s)$.
3. If the colour of the marble drawn was $c \in C$, we set $X_n = d(s, c)$.

We say that $\vec{X} \in \text{RUP}(S, C, U, d)$.

In P. Muliere [2000], a closed-form expression for the finite-dimensional laws of $\vec{X}$ was derived. From this expression, it is shown that the Process $\vec{X}$ is partially-exchangeable in the sense of Diaconis & Freedman Diaconis P [1980]. We state the partial-exchangeability condition with Axiom 4.11.

Axiom 4.11 (Partial-Exchangeability)**.** According to Diaconis P [1980], $\vec{X}$ is partially-exchangeable

if, for two "equivalent" sequences $s = \{s_0, \dots, s_n\}$ and $\bar{s} = \{\bar{s}, \dots, \bar{s}_n\}$, we have

$$\mathbb{P}[X_0 = s_0, \dots, X_n = s_n] = \mathbb{P}[X_0 = \bar{s}_0, \dots, X_n = \bar{s}_n] \tag{4.10}$$

where "equivalence" is defined as having the property that the number of transitions between any two states $s_i, s_j \in S$ are the same for both processes.

Before stating the Theorem that defines the finite-dimensional law of $\vec{X}$, as alluded to a moment ago, we need to introduce some additional notation and terminology:

- "Admissibility": A sequence $s = \{s_0, \dots, s_n\}$ is said to be "admissible" if there exists a finite sequence of sampled coloured marbles $\vec{c} = \{c_0, \dots, c_n\}$ such that $d(s_i, c_i) = s_{i+1}$.
- $h_s(c)$: We count the total number of states $t(s, s^* = d(s, c))$ that are reachable from a state $s \in S$, after sampling a marble of colour c, with the quantity $h_s(c) = t(s, s^*)$.
- $H(s)$: We count the total number of possible transitions between s, and any other state $s^* \in S$, with the function $H(s) = \sum_{s^* \in s} t(s, s^*)$.

We now have all the pieces in place to introduce theorem 4.12, which we state without proof. For the proof, we direct the reader to P. Muliere [2000], or Cirillo and Hüsler [2011].

Theorem 4.12 (Finite-Dimensional Law). *For all $n \geqslant 1$, and any finite-sequence $s = \{s_0, \dots, s_n\}$ with elements $s_i \in S$, we have*

$$\mathbb{P}[\vec{X} = \{X_0 = s_0, \dots, X_n = s_n\}] = 0, \qquad \textit{if } s \textit{ is } \textbf{\textit{not}} \textit{ "admissible"} \tag{4.11}$$

otherwise we have

$$\mathbb{P}[\vec{X} = \{X_0 = s_0, \dots, X_n = s_n\}] = \prod_{s \in S} \frac{\prod_{c \in C} \prod_{j=0}^{h_s(c)} (m_s(c) + j r(s))}{\prod_{k=0}^{H(s)-1} (k r(s) + \sum_{c \in C} m_s(c))} \tag{4.12}$$

with "reinforcement" $r(s)$, number of c-coloured marbles in state s, $m_s(c)$, and the convention that $\Pi_0^{-1} = 1$.

The reader can find an interesting application (in Peluso et al. [2015]) of the above RUP construction to credit risk modelling where the process $\vec{X}$ models the credit-rating l of a company at time t with a state-space S and corresponding elements $s = (t, l) \in S$. The author(s) considered

$l = 0$ to be the "safest" credit-rating and L to be the "riskiest". The RoM is specified to transition between credit-ratings as per the rule

$$\begin{aligned} d(s, c_1) &= (t+1, l) \\ d(s, c_2) &= (t+1, l+1) \\ d(s, c_3) &= (t+1, l-1) \\ d(s, c_4) &= (t+1, L) \end{aligned} \tag{4.13}$$

with colours $\{c_1, \ldots, c_4\} \in C$ where sampling a c_1-coloured marble leaves the credit-rating unchanged while sampling a c_4-coloured marble immediately sends the company into default. Further details are omitted and are left for the reader to explore in Peluso et al. [2015]. For now, we move on and wish to explore the possibility of using the RUP construction to build an Alarm-System along the lines of Antunes M and FK [2003] and that discussed earlier in Section 4.1.

4.4.1 | An Urn-Based Alarm System

The authors of Cirillo and Hüsler [2011] built a Bayesian Alarm System $\vec{\mathcal{A}}$ for Catastrophe Prediction following the framework of Antunes M and FK [2003] (Section 4.1) using the Reinforced Urn Process (RUP). Generally speaking, Urns have become a popular modelling methodology in "Bayesian Nonparametrics" (see, for example, HM [2009]). To see how we can translate the RUP into the language of an Alarm System $\vec{\mathcal{A}}$, first consider a Time-Series $\vec{Y} = \{Y_0, \ldots, Y_T\}$ denoting the variable of interest with $\vec{Y}_1$, $\vec{Y}_2$, and $\vec{Y}_3$ the "Past", "Present", and "Future", respectively, as in Section 4.1. We can discretize the process $\vec{Y}$ into a new process $\vec{Y}^*$ using k different regions so that when $Y_t \in [d_0, d_1) \Rightarrow Y_t^* = 0$, $Y_t \in [d_1, d_2) \Rightarrow Y_t^* = 1$ and so on up until $Y_t^* \in [d_{k-1}, d_k] \Rightarrow Y_t^* = k-1$. We now have a Process $\vec{Y}^* = \{0, 1, \ldots, k-1\}$ with individual elements $\{0, 1, \ldots, k-1\}$ that can each be associated to specific colours $\{c_1, c_2, \ldots, c_k\} \in C$ of an Urn. We now have a fully specified RUP, $\vec{Z} \in \text{RUP}(S, C, U, d)$.

In the Solar Flare Event Prediction problem of Cirillo and Hüsler [2011], the author(s) showed that the RUP $\vec{X} \in \text{RUP}(S, C, U, d)$ is capable of learning the periodicity of extreme Solar Flare outbursts. The raw data is shown in Figure 4.1, along with a stylized idea of the discretization procedure described earlier, represented by the colours green, yellow, orange, and red. In this setup, $S = \mathbb{Z}^+ \times \{0, 1, \ldots, L\}$ with $s = (n, l)$ denoting the "level" of risk l at instant n. Here, $l = 0$ corresponds to the "safest" state and $L = 3$ represents a "Catastrophe" $\mathcal{C}_{n^*}$. As before, each

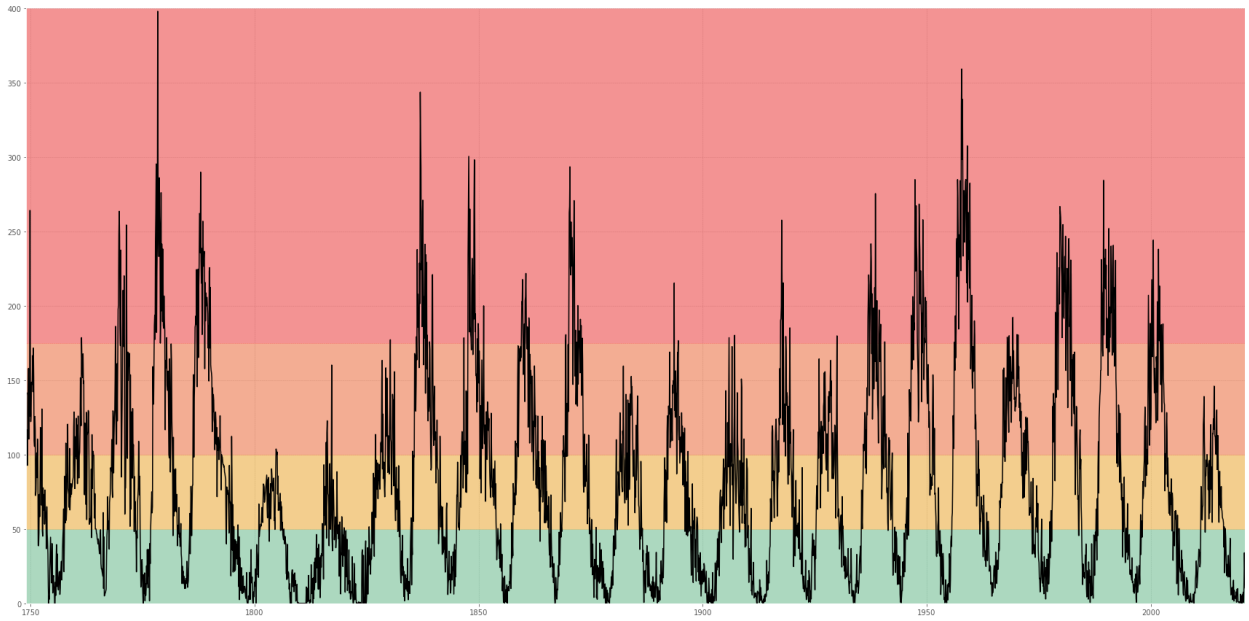


Figure 4.1: Naive Alarm System

$\vec{Y}$ clustered according to $Y_i \in [0, 0.25]$, $Y_i \in [0.25, 0.85]$, and $Y_i \in [0.85, 1.0]$.

State $s \in S$ is endowed with an Urn $U(s)$ on a set of four colors $C = \{c_1, c_2, c_3, c_4\}$ containing $m_s(c)$ number of c-coloured marbles for each color $c \in C$ with reinforcement mechanism $r(s)$. The $\mathrm{RUP}(S, C, U, d)$ is then characterized by its RoM $d : S \times C \to S$. The authors defined the function d according to Definition 4.13 below.

Definition 4.13 (Rule of Motion)**.** The RoM is a $d : S \times C \to S$ mapping of the form

$$d(s, c) := \begin{cases} (n+1, l-1), & \text{if } c = c_1 \\ (n+1, l), & \text{if } c = c_2 \\ (n+1, l+1), & \text{if } c = c_3 \\ (n+1, 3), & \text{if } c = c_4 \end{cases} \tag{4.14}$$

Whenever a "Catastrophe" $\mathcal{C}_{n^*}$ strikes in state $s = (n^*, L)$, we reset the process $\vec{X}$ to the initial state $\vec{X}_{n^*} \to X_0 = (0, 0)$ and start the process over until Catastrophe strikes again. Now, we define $\tau_0 = (0, 0)$ and, for $r \geqslant 1$, we set $\tau_r = \inf\{n \geqslant \tau_{r-1} : X_n = (0, 0)\}$. This process generates a sequence $\vec{\tau} = \{\tau_0, \tau_1, \ldots, \tau_r\}$ between two consecutive visits of the initial state, $X_0 = (0, 0)$ by the process $\vec{X}$. Restarting the process after each Catastrophe allows the process to learn from the

past to make predictions about the future. In particular, if a state s has already been visited by $\vec{X}$, then the composition of the Urn $U(s)$ has been updated via reinforcement. Otherwise, if a state s^* has not yet been visited, its corresponding Urn $U(s^*)$ simply contains the prior information related to the Urns initial composition. According to this model, $\vec{X}$ can only reach a "Catastrophic" risk-level if either:

1. we happen to repeatedly sample c_3-coloured marbles
2. we happen to extract a c_4-coloured marble.

We say that the Process $\vec{X}$ is recurrent if

$$\mathbb{P}[\bigcap_{r=0}^{\infty}\{\tau_r < +\infty\}] = 1 \tag{4.15}$$

or, alternatively, if the following condition

$$\lim_{n\to+\infty}\prod_{k=1}^{n}\frac{m_{(k,l)}(c_1)+m_{(k,l)}(c_3)}{\sum_{i=1}^{4}m_{(k,l)}(c_1)} = 0 \tag{4.16}$$

is satisfied. It is easy to visualize the evolution of $\vec{X}$ in terms of the sequence of stopping-times $\vec{\tau} = \{\tau_0, \tau_1, \dots, \tau_r\}$. If $\vec{X}$ is recurrent, given we reset $\vec{X} \to (0,0)$ every time l reaches L, meaning a "Catastrophe" occurs, then we generate a sequence of "0-Blocks" associated to the sequence $\vec{\tau} = \{\tau_0, \tau_1, \dots, \tau_r\}$. A "0-Block" can be thought of as a "Block" of states in S that start in $s_0 = (0,0)$ and up in $s_n = (n, L)$, where $n \geqslant 1$. In the model of Cirillo and Hüsler [2011], $L = 3$. In this way, an example of $\vec{X}$ could be

$$\begin{aligned}\vec{X} = \{&(0,0),(1,0),(2,1),(3,1),(4,2),(5,3),\\ &(0,0),(1,1),(2,2),(3,3),\\ &(0,0),(1,0),(2,0),(3,3),\\ &(0,0),(1,1),(2,2),(3,2),(4,2),(5,3),\dots\}\end{aligned} \tag{4.17}$$

which would generate the following set of "0-Blocks" Figure 7.1.

$$\underbrace{0,0,1,1,2,3}_{\text{Block-1}}\,|\,\overbrace{0,1,2,3}^{\text{Block-2}}\,|\,\underbrace{0,0,0,3}_{\text{Block-3}}\,|\,\overbrace{0,1,2,2,2,3}^{\text{Block-4}} \tag{4.18}$$

An interesting point made in Cirillo and Hüsler [2011] is that, given $\vec{X}$ is recurrent and "Exchangeable", then, by Diaconis & Freedman Diaconis P [1980], each "0-Block" is a Markov Chain. Therefor, the process $\vec{X}$ is a mixture of Markov Chains. As such, Muliere P. Muliere [2000] showed that, given the construction of the RUP $\vec{X} \in \text{RUP}(S, C, U, d)$, Reincorced Urn Processes are generally conjugated which turns out to be connected to the conjugacy of the Product Dirichlet Process, of which our RUP construction is related to. The details outlining this relation are left for the reader to explore in P. Muliere [2000] and Cirillo and Hüsler [2011]. For our purposes, all this means is that we can use the RUP for Nonparametric Bayesian Prediction, i.e. Antunes Antunes M and FK [2003] and Cirillo Cirillo and Hüsler [2011]. All that is left is for us to define an "Alarm-Region", along the lines of Antunes M and FK [2003] and Section 4.1, for which to associate our $\vec{X} \in \text{RUP}(S, C, U, d)$ to. The author(s) of Cirillo and Hüsler [2011] showed that an appropriate Alarm-Region is the probability of our Process $\vec{X}$ reaching the risk threshold L exceeding some predetermined threshold, γ, in $n + k$ time-steps, following the "0-Block" terminology. It turns out, thanks to Theorem 4.12, that this probability is relatively easy to compute.

Definition 4.14 (RUP Alarm Region)**.** We cast an "Alarm" $\mathcal{A}_n$ at time n if

$$\mathbb{P}[\tau_r = n + k | \vec{\tau}_{r-1}] \geqslant \gamma, \qquad \gamma \in [0, 1] \tag{4.19}$$

where $\vec{\tau}_{r-1}$ denotes the sequence of stopping-times $\{\tau_1, \ldots, \tau_{r-1}\}$ and γ is a predetermined threshold, which can be explicitly calculated via Theorem 4.12 using

$$\mathbb{P}[\tau_r = n + k | \vec{\tau}_{r-1}] = \sum_{j=1}^{h} \mathbb{P}_j[(n, l), (n + k, L)] \tag{4.20}$$

where $\mathbb{P}_j[(n, l), (n + k, L)]$ denotes the probability of the j^{th}-feasible path, out of h, taking $(n, l) \to (n + k, L)$, occurring.

All of the details, including the specific implementation and application details to the Solar Flare data set can be explored in Cirillo Cirillo and Hüsler [2011]. In short, every RUP application is going to be a little different based on the application. Detailing the entire implementation in Cirillo and Hüsler [2011] would be redundant for our purposes. As a teaser for the curious reader, the hyper-parameters γ, k-future time steps, and reinforcement mechanism $r(s)$ were all optimized so as to minimize the Alarm Systems $\vec{A}$ number of "False-Alarms". We will come back to this modelling approach later in Chapter 6 where we'll make a few adaptations for the problem

of predicting stock market crashes. For now, we move onto the next section.

5

Bubbles & Crash Theory

In order to test the efficacy of an alarm system, or any sort of predictive model for that matter, we need to define what we want our model to predict, precisely. In the case of this work, we are trying to predict stock market crashes. Thus, we need to define what a stock market crash is, exactly. There is no general concensus on how to (rigorously) define a stock market crash. It is easy to quantify what a crash does, but not what it is. In this chapter, we review one of the more well-known approaches to defining stock market crashes: that of Didier Sornette [2006]. In Section 5.1, we formally define a stock market drawdown, and show what they look like. Second, in Section 5.2, we show how we can use the fitting, of the distribution of drawdowns, to the "Stretched-Exponential" as a means to defining stock market crashes as "outliers" to this fitting. Lastly, in Section 5.3 we cover an (sort of) alarm system for stock market crashes developed by the same author, Didier Sornette [2006], based on the so-called "Log-Periodic Power Law Singularity" (LPPLS) model. In this way, not only do we lay the foundation for a new methodology for defining stock market crashes, which we will introduce later, but also provide a State-of-the-Art stock market crash model, or "bubble detector", as the author(s) Didier Sornette [2006] refer to it, that we can test our Urn-Based alarm system against.

5.1 | Drawdowns

Stock Markets are in a constant state of fluctuation. Some days the market rises. On other days, it falls. However, every once in awhile, the Stock Market will not only fall, but "Crash".

Counter-intuitively, it's hard to pinpoint what this means, precisely, in a systematic way. Typically, "Crashes" are associated to time periods in which the Stock Market falls by a large amount with "large" being (say) $\sim 25\%$. This begs the question: When does the time-period start? When does it end? What if a 6-day long "Crash" is interrupted by a 1-day positive return on day three? Do we calculate the Loss from the top of that interruption? Or the original descent? It's easy to see how tricky of a question it becomes. Thankfully, we can rely on the Definition of a "Drawdown" to define Stock Market Crashes, which we do next.

A Stock Market "Crash" is nothing more than an extreme Stock Market "Drawdown". Simply put, a "Drawdown", according to Didier Sornette [2006], is the cumulative loss (negative return) associated to a time-period in which a Stock Market Index sees consecutively negative (daily) returns. Think, for example, of the DOW Jones Industrial Average (DJIA). More precisely, we capture this notion with Definition 5.1.

Definition 5.1 (Drawdown)**.** Suppose $\{\mathcal{P}_t\}_{t=1}^{T}$ is a time series representing the value of some Financial Index at closure. Consider $\mathcal{P}_{max} = \mathcal{P}_{t^\star}$ a local maximum, and $\mathcal{P}_{min} = \mathcal{P}_{t^\star+\delta}$ to be the next corresponding minimum. The "Drawdown" associated to $\mathcal{P}_{max}$ and $\mathcal{P}_{min}$ is defined as the percentage

$$\mathcal{D}_{t^\star} \coloneqq \frac{\mathcal{P}_{min} - \mathcal{P}_{max}}{\mathcal{P}_{min}} \tag{5.1}$$

where $\mathcal{P}_{max}$ and $\mathcal{P}_{max}$ both satisfy the condition $\mathcal{P}_{max} = \mathcal{P}_{t^\star} > \mathcal{P}_{t^\star+1} > \cdots > \mathcal{P}_{t^\star+\delta} = \mathcal{P}_{min}$ where $t^\star \geqslant 2$ and $\delta \geqslant 1$. Each Drawdown has an associated $(\mathcal{D}, t^*, \delta)$ tuple.

One of the most infamous stock market crashes in history was the "Black Monday" Crash of 1987. For the readers convenience, we display the 1987 "Black Monday" crash in figure 5.1.

On October 14, 1987, the DJIA began a 4-day period of negative consecutive daily returns culminating in a 22% loss on "Black Monday" October 19, 1987. During this 4-day "drawdown", the DJIA lost a little over 30% of its value.

Thanks to definition 5.1, we now have a precise way of measuring the timing, duration, and intensity, of Stock Market drawdowns: $t^\star$, δ, and $\mathcal{D}$, respectively. Naturally, we now must determine which of these drawdowns $\mathcal{D} \coloneqq \{\mathcal{D}_{t^\star}\}$ correspond to a "Crash". Following the approach developed in Didier Sornette [2006], we can define a "Crash" (loosely speaking) as a drawdown $\mathcal{C}_{t^\star} \in \mathcal{D}$ that is an outlier to an appropriately fitted "Stretched Exponential" distribution of the Rank-Ordered drawdowns, $\mathcal{D}_{[1]} < \mathcal{D}_{[2]}, \ldots, < \mathcal{D}_{[N]}$, where $\mathcal{D}_{[1]}$ is the smallest (most negative) drawdown. Here, $n \in \{i\}_{i=1}^{N}$ is the rank of the Order Statistics alluded to above. Now, $\mathcal{D}_{[n]}$ can

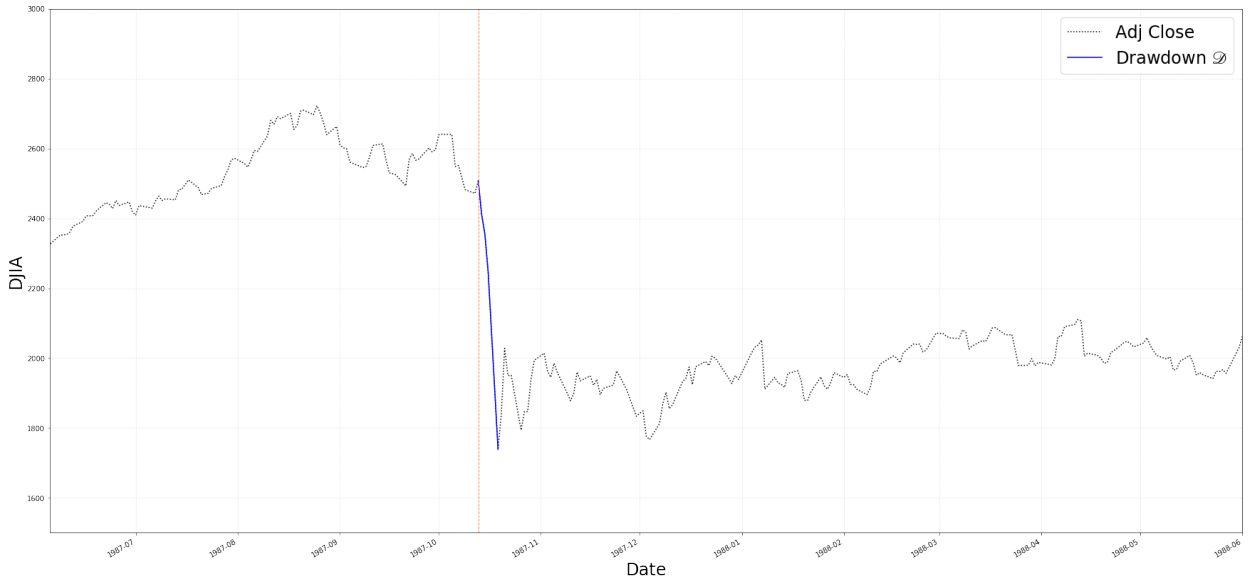


Figure 5.1: Log-Periodic Power Law Singularity (LPPLS) DOW Jones Industrial Average (1985-2020)

be plotted against the rank number $N(\mathcal{D}_{[i]}) \in \{i\}_{i=1}^{N}$ allowing us to fit the rank-distributed drawdowns with pretty good results. For all the details, we encourage the reader to see Didier Sornette [2006]. For now, we move ahead and define the "Stretched Exponential" with Definition 5.2 below.

Definition 5.2 (Stretched Exponential Distribution of Ranked Drawdowns)**.**

$$N(\mathcal{D}; A, b, z) = Ae^{-b^{-z}|\mathcal{D}|^z} \tag{5.2}$$

As mentioned in Didier Sornette [2006], the "Stretched-Exponential" is not particularly interesting, it just so happens to fit the bulk of the distribution (well enough) to allow us to single out "extreme" drawdowns. The author(s) of Didier Sornette [2006] took the log-transform of definition 5.2, as follows,

$$\log N(\mathcal{D}) = \log A - b|\mathcal{D}|^z \tag{5.3}$$

in order to guarantee the robustness of the fitting procedure. We fitted Equation 5.3 using the (Log) Daily-Returns of the DOW Jones Industrial Average (DJIA) from Jan 1,1985 to Dec 31, 2020. We plot the results in Figure 5.2 highlighting drawdowns $\mathcal{D} \geqslant 0.20$. Precisely distinguishing Crashes from the bulk of the distribution is complicated. In Didier Sornette [2006], the authors define a "Crash" as an "outlier" to the distribution of $N(\mathcal{D})$ that does not fit in the continuation of the statistics accountable for at least 95% of the distribution. We highlight what we believe to be

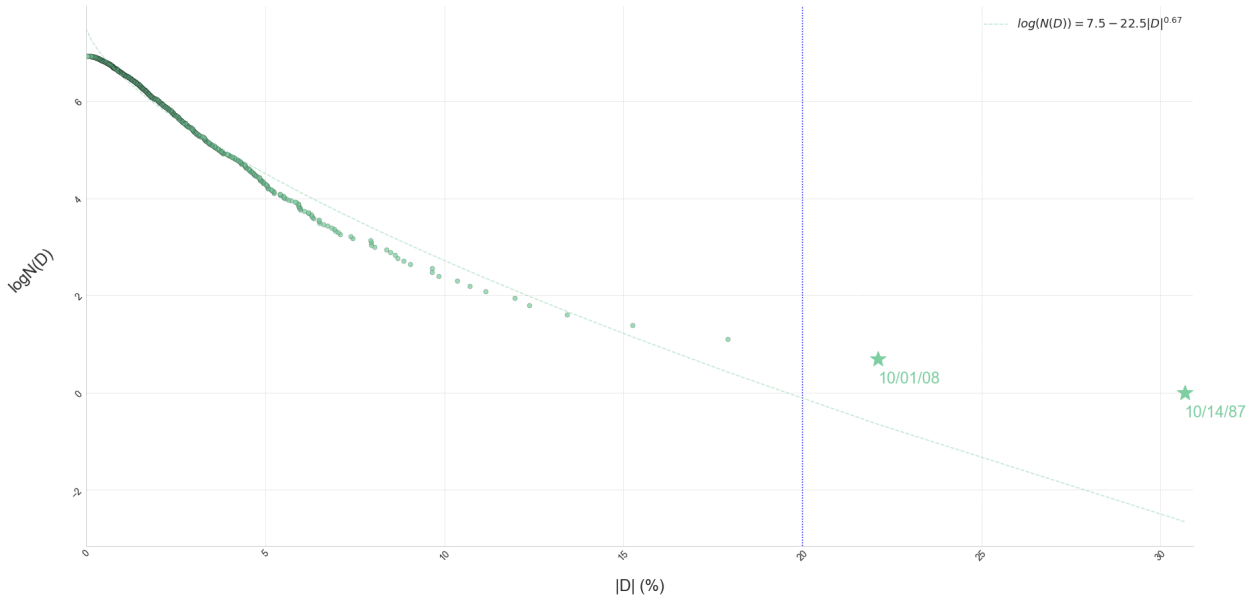


Figure 5.2: Strecthed Exponential Distribution of Ranked DJIA Drawdowns: 1985-2020

these points in figure 5.2 which, coincidentally, correspond to drawdowns $\{\mathcal{D}_{t_i^*}\} \leqslant -20\%$ below the -20% threshold. These "crashes" coincide with the 2008 Financial Crisis (-23%) and Black Monday (-30%).

5.2 | Price Coarse-Graining Drawdowns

In practice, the drawdown of definition 5.1 might be insufficient. To see why, a drawdown $\mathcal{D}_{t^\star}$, according to definition 5.1, is the cumulative loss from the local maximum $\mathcal{P}_{t^\star}$ to the next local minimum $\mathcal{P}_{t^\star+\delta}$ regardless of how small each positive (daily) log-return $r_{t^\star+\delta+1} = \log\frac{\mathcal{P}_{t^\star+\delta+1}}{\mathcal{P}_{t^\star+\delta}} \geqslant 0$ is. However, since financial data is very noisy, we can introduce a threshold ε that ignores any $r_{t^\star+\delta+i} \leqslant \varepsilon$. To this end, Didier Sornette [2006] introduces a "Price Coarse-Graining" algorithm, and associated "ε-drawdown", that interrupts consecutive daily losses only when the reverse positive daily return exceeds the threshold ε. Since every Financial Asset behaves differently, Didier Sornette [2006] propose the usage of an individual assets volatility, say σ_{DJIA} for the DJIA, to determine such a threshold.

We display an example of when the ordinary drawdown doesn't capture interesting financial behavior in Figure 5.4. On August 27, 2001, the Dow Jones Industrial Average (DJIA) began a -22% drawdown. However, this drawdown was interrupted by a series of positive daily returns, displayed in red. The ordinary definition 5.1 of a drawdown would exit the algorithm early and

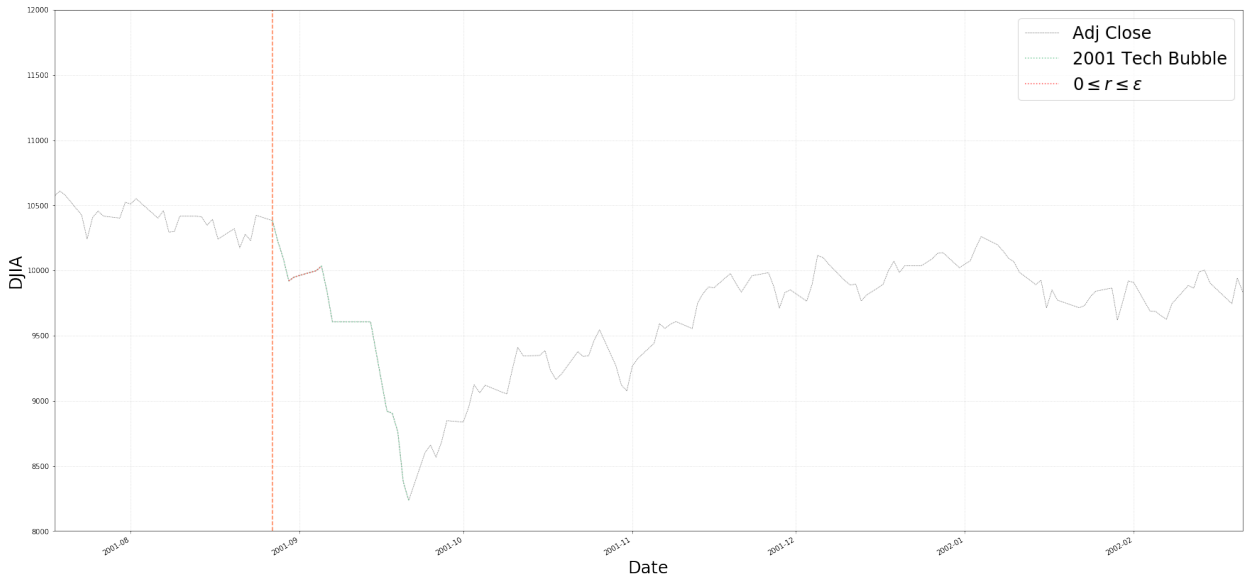


Figure 5.3: Stock Market Crashes: DOW Jones Industrial Average (1985-2020)

not reflect the true severity of the drawdown. To address this, we introduce the "Price Coarse-Graining Algorithm" of Didier Sornette [2006] with definition 6.21.

Definition 5.3 (Price Coarse-Graining Algorithm & ε-Drawdown)**.** The Price Coarse-Graining algorithm, and associated ε-drawdown, works as follows:

- Identify a local maximum $\mathcal{P}_{\varepsilon,\max} = \mathcal{P}_{t^\star}$ as before
- Identify a local minimum $\mathcal{P}_{\varepsilon,\min} = \mathcal{P}_{t^\star+\delta}$ such that $r_{t^\star+\delta+1} \geqslant \varepsilon$.
- Calculate $\mathcal{D}_\varepsilon$ as $\frac{\mathcal{P}_{\varepsilon,\min} - \mathcal{P}_{\varepsilon,\max}}{\mathcal{P}_{\varepsilon,\min}}$.
- Ignore any resulting drawdowns such that $\mathcal{D}_\varepsilon \geqslant 0$.

Definition 6.21 allows us to define the Stretched Exponential Distribution for ε-Drawdowns, just as before.

Definition 5.4 (Coarse-Grained "Stretched" Exponential Distribution of Drawdowns)**.**

$$N(\mathcal{D}_\varepsilon; A, b, z) = Ae^{-b|\mathcal{D}_\varepsilon|^z} \tag{5.4}$$

We follow the same fitting procedure discussed in Section 5.2 and fit the log ranks of the drawdowns in Figure 5.5. Following the approach of Didier Sornette [2006], we consider $\varepsilon = \{0, \frac{\sigma}{2}, \frac{\sigma}{4}\}$

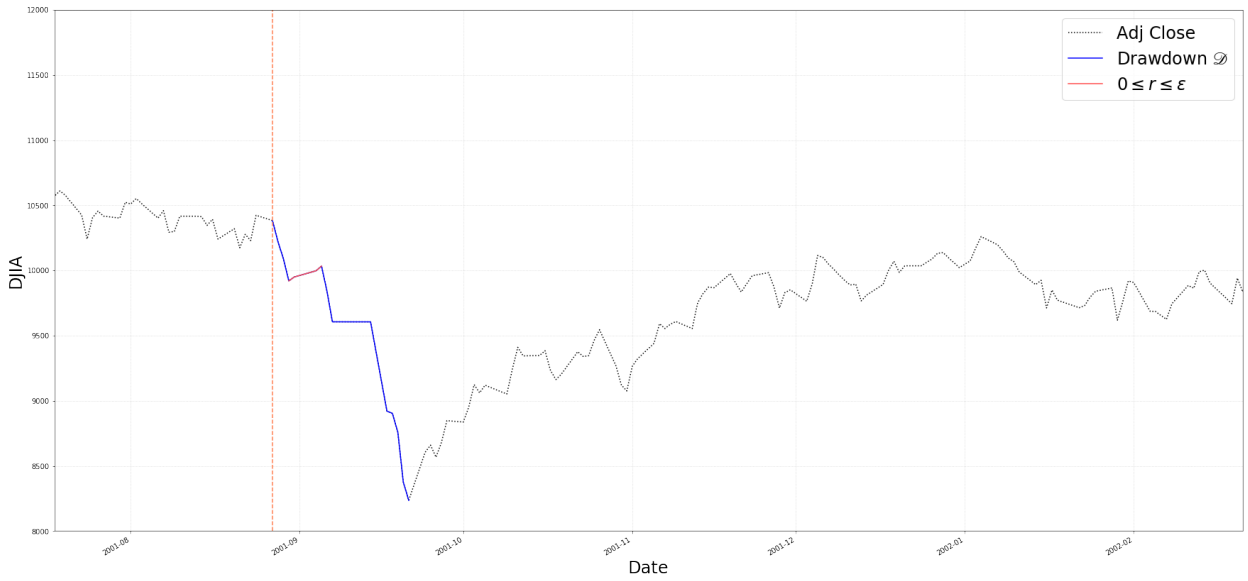


Figure 5.4: Stock Market Crashes: DOW Jones Industrial Average (1985-2020)

where σ is nothing but the volatility (standard-deviation) of the Financial Index of interest calculated as

$$\sigma^2 = N^{-1} \sum_{i=1}^{N} (r_{i+1} - \mathbb{E}[r])^2 \tag{5.5}$$

where $r_{i+1} = \log(\frac{\mathcal{P}_{t+1}}{\mathcal{P}_t})$ and $\mathbb{E}[r]$ is the long run average. Expressing the ε-thresholds in units of volatility immunizes our Price Coarse-Graining algorithm from noisy return data.

Looking at figure 5.5 we notice that the three thresholds for $\varepsilon = \{0, \frac{\sigma}{2}, \frac{\sigma}{4}\}$ yielded similar results in terms of identifying outlier drawdowns. However, the ε-threshold $\varepsilon = \frac{\sigma}{2}$ identified a drawdown during the so-called "Tech Bubble" beginning on August 27, 2001. Referring to Figure 6.6, we plot the oulier drawdowns from the (Price Coarse-Grained) drawdown algorithm(s) against the DJIA over the time period 1985-2020.

5.3 | LPPLS

In this section, the derivation of the LPPLS model is outlined based on the original work of Anders Johansen and Sornette [2006]. The LPPLS Model is a nonlinear function that aims to regress the Expected Log-Price $\mathbb{E}[\log(p(t))]$ of a Financial Asset $p(t)$ at time t. Before moving forward,

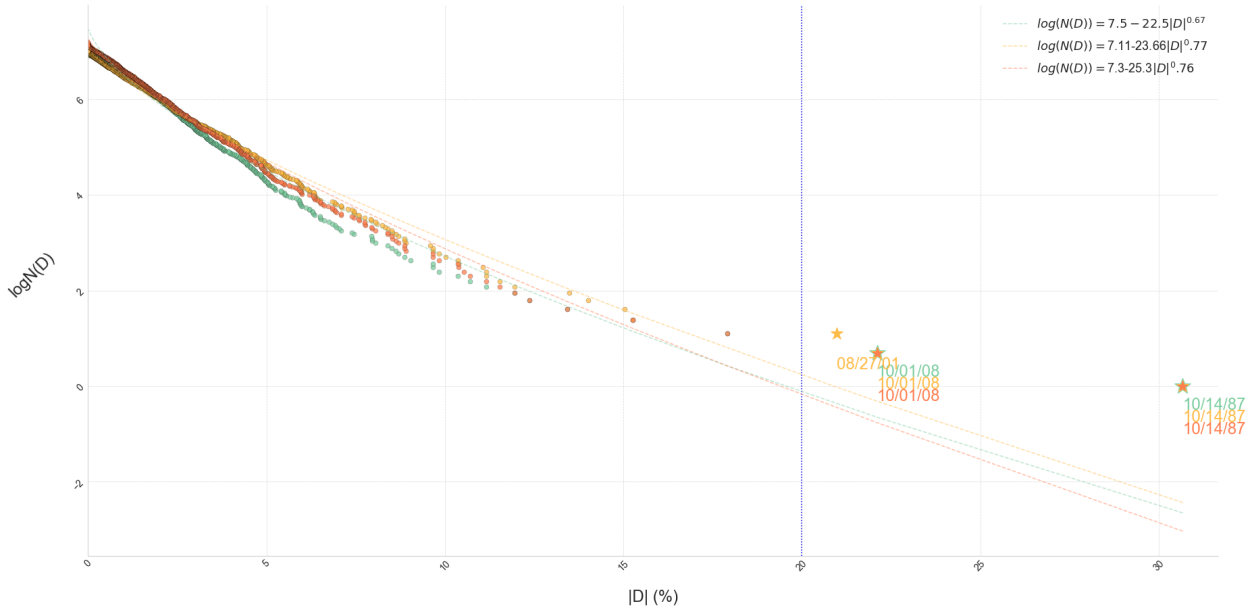


Figure 5.5: Strecthed Exponential Distribution of Ranked DJIA Drawdowns for $\varepsilon = \{0, \frac{\sigma}{2}, \frac{\sigma}{4}\}$: 1985-2020

we provide the LPPLS Model, for the readers convenience, below with Equation 5.13.

$$\mathbb{E}[\log(p(t))] = A + B(t_c - t)^m + C(t_c - t)^m \cos(\omega \log(t_c - t) - \phi) \tag{5.6}$$

The LPPLS model assumes that agents are risk-neutral, have rational expectations and ignore arbitrage, interests-rates, dividends, risk aversion, information asymmetry, along with the market-clearing condition. As such, the rise of the expected asset price must compensate for the expected risk, implying that the asset price is necessarily a martingale

$$\mathbb{E}[p(t)|\mathcal{F}_s] = p(s), \qquad \forall s \leqslant t \tag{5.7}$$

for the conditional expectation $\mathbb{E}[\cdot|\mathcal{F}_s]$ on the filtration $\mathcal{F}_s$. We use a discontinuous jump-diffusion process, J to model the occurence of a crash taking place, taking a value of 0 before, and 1 after, the critical time t_c. We can use a CDF $F(t)$ to model t_c with density $f(t) = \frac{dF(t)}{dt}$ and crash hazard rate $h(t) = \frac{f(t)}{(1-F(t))}$ representing the probability of a crash occurring between t and $t + dt$, provided that the crash hasn't happened yet. We can calculate the expectation of an increment for the discontinuous jump-diffusion process dJ as $\mathbb{E}[dJ] = h(t)dt$. If the price of the asset falls

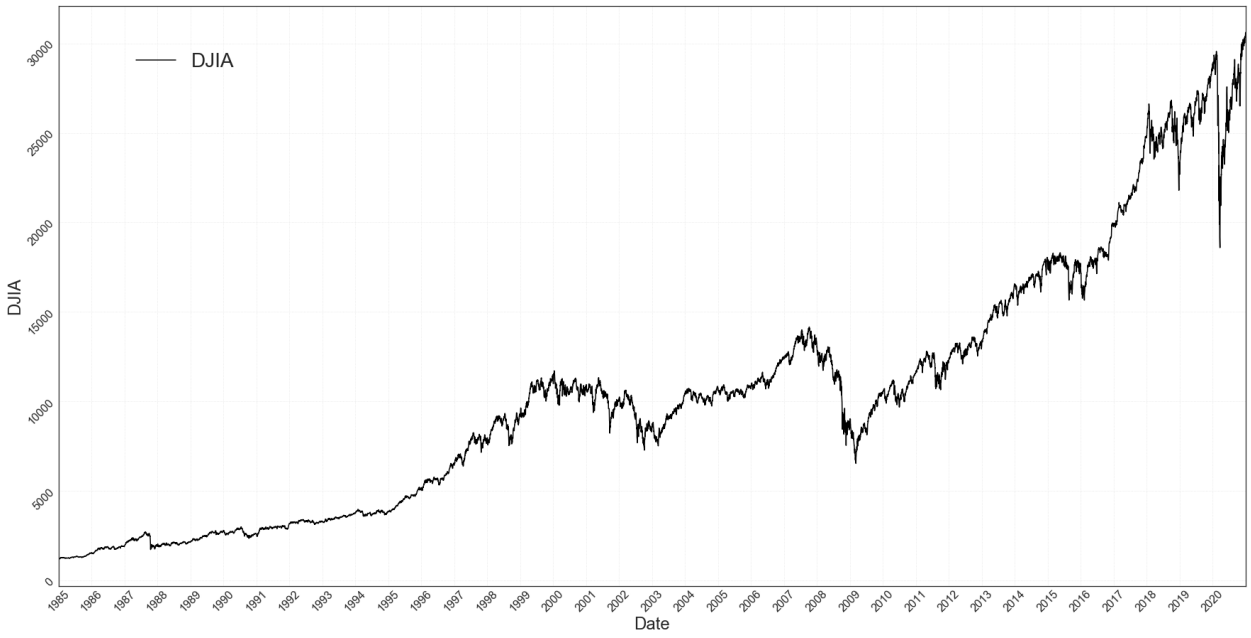


Figure 5.6: Stock Market Crashes: DOW Jones Industrial Average (1985-2020)

at a fixed percentage $\alpha \in [0, 1]$ then we have

$$\begin{aligned} dp &= \mu(t)p(t)dt - (t)dJ + \sigma(t)p(t)dW(t) \\ \frac{dp}{p} &= \mu(t)dt - kdJ + \sigma(t)dW(t) \end{aligned} \tag{5.8}$$

for the increment dp of $p(t)$, time-dependent return $\mu(t)$, time-dependent volatility $\sigma(t)$, and brownian motion $W(t) \sim \mathcal{N}(0, 1)$. However, since $p(t)$ defines a martingale, we have $\mu(t) = kh(t)$ which implies that the risk of the crash $h(t)$ is proportional to the time-dependent return $\mu(t)$. Conditioning on no Crash occurring, we get

$$\begin{aligned} \frac{dp}{p} &= \mu(t)dt - kp(t) \cdot 0 + \sigma(t)dW(t) \\ &= \mu(t)dt + \sigma(t)dW(t) \end{aligned} \tag{5.9}$$

which, upon Integrating and taking Expectations, yields

$$\mathbb{E}[\log(\frac{p(t)}{p(t_0)})] = k\int_{t_0}^{t} h(s)ds, \qquad t_0 \leqslant s \leqslant t \tag{5.10}$$

for initial condition t_0. The final step is the specification of the crash hazard rate $h(t)$, which aims

to quantify the probability that a large number of agents will all take the same sell position at the same time. Anders Johansen and Sornette [2006] propose a model that captures the "imitative local micro-interactions" of the agents, which takes the form of

$$s_i = sgn(K \sum_{i=1}^{N} s_j + \sigma\varepsilon_i) \tag{5.11}$$

where $s_i \in \{-1, +1\}$ when Agent i is "selling", or "buying", respectively. Furthermore, Kis simply a coupling constant between the set of Agents and $sgn(\cdot)$ is simply the "sign"-Function. Omitting details, which we encourage the reader to see Anders Johansen and Sornette [2006] for, we get

$$h(t) \approx a(t_c - t)^{m-1}(1 + bcos(\omega log(t_c - t) + \phi)) \tag{5.12}$$

which, after plugging back into the expression for $\mathbb{E}[log(p(t))]$ as per Equation 5.10, yields

$$\mathbb{E}[log(p(t))] = A + B(t_c - t)^m + C(t_c - t)^m cos(\omega log(t_c - t) - \phi) \tag{5.13}$$

where several (arbitrary) simplifications are made to get back to the LPPLS Model of Equation 5.13. We plot a sample LPPLS fit to the Dow Jones Industrial Average (DJIA) from January 1, 1900 to December 31, 2020 in Figure 5.7.

This bring us to the LPPLS confidence indicator (CI) which provides a measure of confidence to the observation of a true Log-Periodic Power Law Singularity pattern. As such, a high level for the LPPLS "Positive" CI indicates a future drop or higher levels of volatility. The opposite for the LPPLS "Negative" CI. Loosely speaking, the LPPLS CI is defined as the number of Fitting Windows, each of size $dt = (t_2 - t_1)$, in which the fitted LPPLS Model satisfy the specified Filter Conditions. Hence, each LPPLS CI takes a value $CI \in [0, 1]$. We compute the CI with the following procedure:

1. Creating Fitting (Time) Windows by shrinking t_1 toward the endpoint t_2.
2. Define the search space for the calibration procedure.
3. Calibrate the LPPLS Model for each Fitting Window.
4. Specify the filter conditions and catalog the number of Fitting Windows that satisfied the filter conditions.

Figure 5.7: LPPLS Model for the DOW Jones Industrial Average (Jan 1, 1900-Dec 31, 2020)

5. Divide the number of fitting Windows from Step: 4 by the total number of fitting windows, yielding the "Positive" CI.

Following this approach, we calculated the positive and negative CI's for the LPPLS model on the entirety of the DJIA price dataset from January 1, 1900 until December 31, 2020. We display the resulting CI's in Figure 5.8. For visualization purposes, we only plot the results from 2010 onward, to avoid cluttering.

This model is interesting for our purposes since these CI's can be used in direct one-to-one correspondence with the probability of witnessing a catastrophic event

$$\mathbb{P}[\tau_r = n + k | \{\tau_1, \ldots, \tau_r\}] \tag{5.14}$$

from the bayesian alarm system discussed in chapter 4. Hence, we'll be able to compare the CI's, from the LPPLS model, with that of the stopping-time probabilities from Equation 5.14 in direct correspondence with one another under the guise of the formal online bayesian alarm system. Thus, the Reinforced Urn Process, which we will develop later, can be tested against a state-of-the-art stock market crash prediction model. Furthermore, understanding how the LPPLS model works will help us analyze the two methodologies and discuss, what we believe to be, their relative strengths and weaknesses.

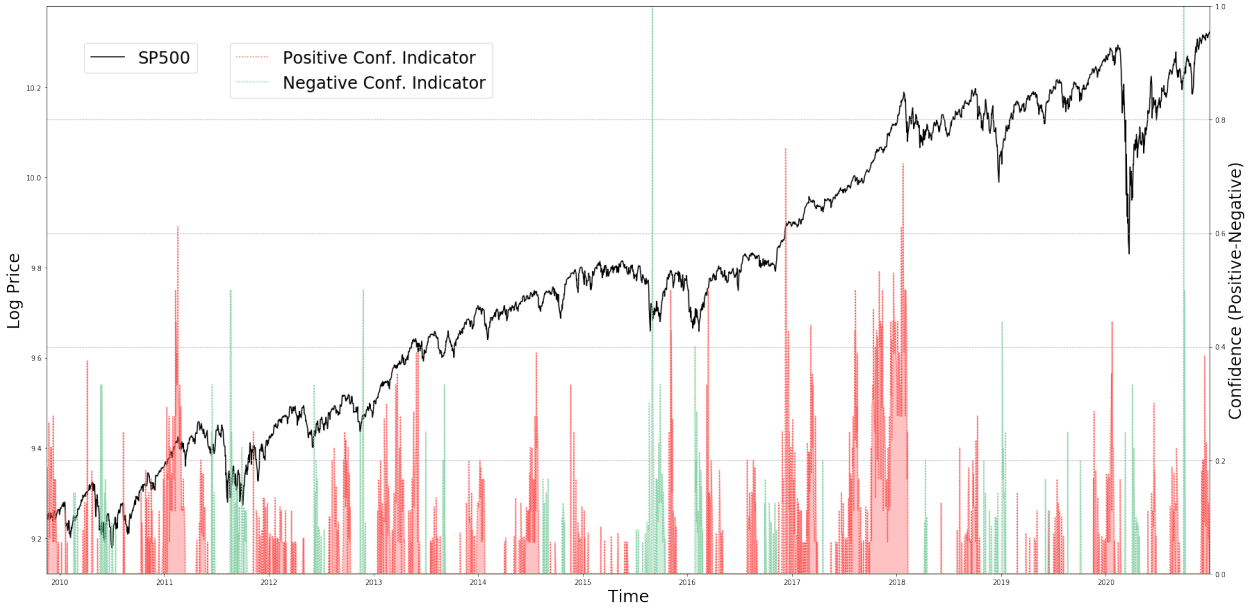


Figure 5.8: LPPLS Positive and Negative COnfidence Indicators (2010-2020

III

Implementation

6

Methodology

In this chapter, we will focus on three things: 1. Gathering the data that we will need to build our alarm system. 2. Familiarizing ourselves with different forms of correlation and dependence. 3. Implementing Extreme Value Theory (EVT) as a means to systematically define stock market crashes. As such, this chapter is broken down into thre different sections. In section 6.1 we will take a closer look at the Dow Jones Industrial Average (DJIA) and discuss the historical constituents that we will be considering to calculate our correlation matrices so as to avoid the so-called "survivorship bias". Additionally, we will specify the sliding window structure we will be using to build the QLO (Quantum Lorenz Ordering). Then, in section 6.2, we will cover three of the main correlation metrics commonly used in the financial literature: Pearson's ρ, Spearman's ρ^S, and Kendall's τ^K. Additionally, we will cover two other correlation and dependence metrics, that are more exotic and less frequently used: Gini's γ^G and Upper (Lower) Tail Dependence λ^U. Lastly, in section 6.3, we will show how we can use the so-called "Generalized Pareto Distribution" (GPD) to not only fit the distribution of drawdowns, but also estimate the threshold where the GPD's tail begins. Hence, any drawdown beyond this threshold will be consider a crash. We identify 13 such drawdowns between 1985-2020 on the DJIA index.

6.1 | Data

We kick things off by considering the Dow Jones Industrial Average (DJIA) as the index of interest to track stock market crashes. Others candidates included the S&P 500, Russel 2000, and Nasdaq.

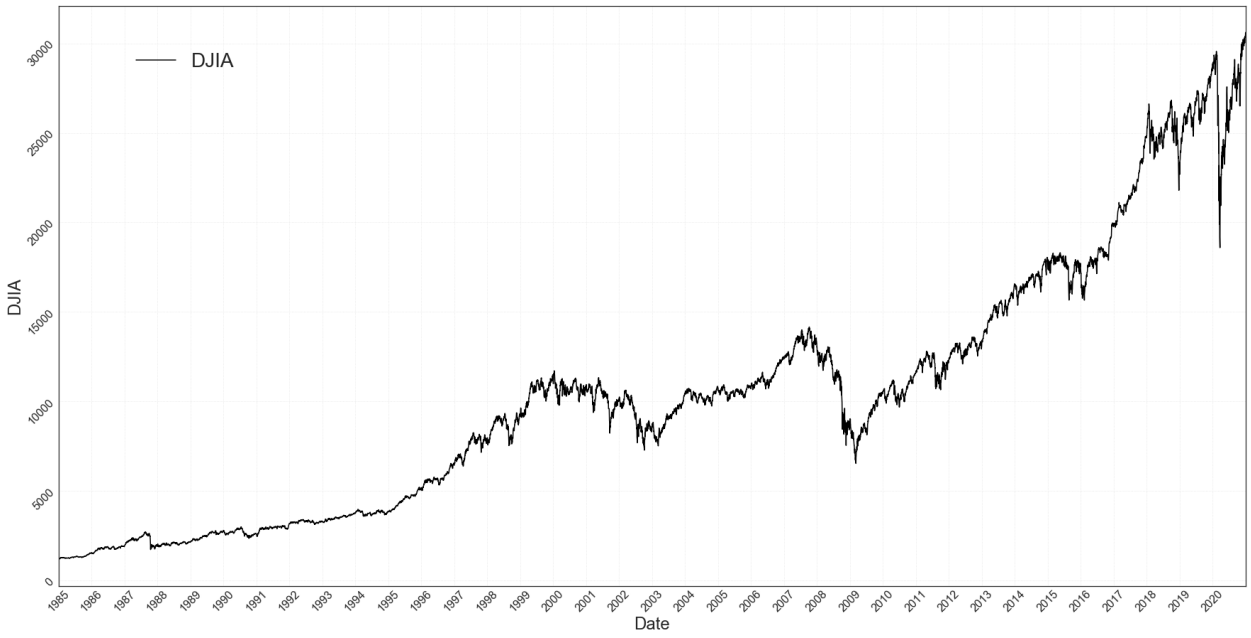


Figure 6.1: DOW Jones Industrial Average (1985-2020)

However, given the nature of this work, we choose to work with the DJIA for two reasons.

1. Historical DJIA data extending back to 1900 (even 1896) can be gathered via freely available data sets, and hence, we can infer better statistics for our EVT analysis.

2. More importantly, the DJIA is composed of 30 constituents (12 originally in 1896), and thus, our correlation matrices will be of a more manageable size 30×30 as opposed to the S&P 500 which has 500 components.

We will see in a moment how we lied (a little bit) in the second point above. For now, the point is that the DJIA will allow for faster calculations and better EVT inference. For the readers convenience we plot the DJIA index from January 1, 1985 to December 31, 2020. The DJIA, as mentioned before, is a weighted price index for 30 of the largest US market cap stocks. Some of these companies include Apple Inc. (AAPL), Caterpillar (CAT), AT&T (T), and the Minnesota Mining and Manufacturing Co. (MMM). As the economy changes, however, companies enter, and exit, the index so as to reflect the state of the economy at the time, which is what stock market indices, like the DJIA, are intended to reflect for investment and even public policy efforts. Hence, the DJIA is a standard index to track if one is interested in studying stock market crashes, which we are.

Now comes the question of building our correlation matrices for the 30 DJIA constituents.

Since we are trying to predict stock market crashes via the QLO, a partial-ordering defined on the eigenvalues for the set of symmetric PSD hermitian operators, a set of which any (symmetric) correlation matrix belongs to, we will collect the correlation matrices for DJIA constituents by computing the correlation coefficient $\rho_{i,j} \in [-1, 1]$ between the log returns $\mathcal{R}_i(\mathcal{S}_i) = \{\mathcal{R}_{i,hn}, \ldots, \mathcal{R}_{i,hn+m}\}$ and $\mathcal{R}_j(\mathcal{S}_j) = \{\mathcal{R}_{j,hn}, \ldots, \mathcal{R}_{j,hn+m}\}$ collected over time intervals $\iota_n = [hn, hn+m]$ representing our sliding windows, as per the language of chapter 3, corresponding to each pair of assets $(\mathcal{S}_i, \mathcal{S}_j) \in \mathcal{S} = \{Ss_1, \ldots, \mathcal{S}_{30}\}$. In so doing we form a collection of correlation matrices $P_\Delta(\mathcal{S}; I)$, for $\Delta = (h, m)$, $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_{30}\}$, and time interval $I = [0, T]$, that we called the temporal state space of correlation matrices. or just temporal state space for short.

At this point, we need to specify what we want our sliding windows, as characterized by the $\Delta = (h, m)$ tuple, to look like. For reasons we will get into in the next chaper, chapter 7, we choose to sample our correlation matrices $\rho_n = \rho_{\iota_n} \in P_\Delta(\mathcal{S}; I)$ over sliding windows of 40 days, each of length 100 days, that is $\Delta = (40, 100)$. If the reader needs to be refreshed on what the temporal state space looks like, in more detail, we refer them back to definition 3.6 of chapter 3. The only thing left to discuss is the actual set of constituents we will be gathering our correlation matrices over, which we do next in section 6.1.1.

6.1.1 | A Note on "Survivorship" Bias

At first glance, ones first instinct, when trying to build the model we are considering in this work, might be to look at the 30 constituents in the DJIA today (present time) and see how many of those constituents were on the market dating back to some arbitrary time in the past. For instance, we found that, of the 30 DJIA components on December 31, 2020, 25 were on the market (being publicly traded) back in May 1986. Hence, the practitioner might build the QLO, using the set of 25×25 correlation matrices corresponding to those 25 assets. However, consider the example of AAPL, which went public in March of 1986. Apple Inc. wasn't added to the DJIA until June 2015. Hence, the model trying to predict stock market crashes on the DJIA, by tracking correlation matrices consisting of assets that weren't present in the DJIA at the time (as would be the case for AAPL) can potentially take away from the models historical performance, and add a bias to the time period closer to the date of its addition to the index (June 2015 for AAPL). This is known in the industry as "survivorship bias" and is infamous among modelers trying to back-test trading strategies based on stock market indices such as the DJIA. Since the RUP alarm system, which we are considering in this work, has to be trained in the past, to make predictions about the future,

this is something we should consider.

Our first attempt was to track the 30 constituents present in the DJIA at every moment in time. In fact, since 1985, roughly 50 different companies (excluding mergers & acquisitions) have been in the DJIA at some point in time. However, using freely available data, the greatest number of assets we were able to include in a dynamically updated set of DJIA constituents was 21 since many companies have gone out of business since. We found that using just 21 assets effected the informativeness of quantum majorization. Thus, we changed gears and grabbed every single asset that ever existed in the DJIA index (dating back to 1985) that is still being traded today, but isn't necessarily still in the index today. Such companies include. but are not limited to, Alcoa Inc. (AA), International Paper (IP), and Navistar (NAV) which left the DJIA in 2013, 2004, and 1991, respectively. We believe this simple ad hoc modification helped our model better track, and reflect, systemic risk in the market over time to the full extent that quantum majorization does. Hence, even though the DJIA consists of 30 companies, our temporal state space consists of correlation matrices $\rho_n = \rho_{\iota_n} \in P_\Delta(\mathcal{S}; I)$, each of dimension $\dim(\rho_n) = 43$, sampled over 40 day sliding windows $\iota_n = [40n, 40n + 100]$ of length 100 days over the time interval I corresponding to May 6, 1986 until December 31, 2020 for the set of assets (variables) $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_{43}\}$.

6.2 | Correlations & Dependence

In this section we familiarize ourselves with a few correlation metrics to test our upcoming alarm system with. These metrics include the well-known Pearson's ρ, Gini's γ^G correlation and Upper (Lower) Tail Dependence λ^U (λ^L). All in all, this section will leave us with three measures of correlation and dependence with corresponding temporal state spaces $P_\Delta(\mathcal{S}; I)$, $\Gamma^G_\Delta(\mathcal{S}; I)$, and $\Lambda^U_\Delta(\mathcal{S}; I)$, respectively, which we build the QLO with.

The most commonly used measure of Correlation is the so-called "Pearson Product-Moment Correlation", or just "Pearson's ρ", for short. Pearson's ρ measures the amount of linear correlation between two sets of data. In words, it is essentially the standardized Covariance between X and Y (see Definition 6.1).

Definition 6.1 (Pearson's Product-Moment Correlation: ρ)**.** Pearson's rho is defined as

$$\rho(X, Y) = \frac{\mathbb{C}\mathrm{ov}(X, Y)}{\sigma_X \sigma_Y} \tag{6.1}$$

where σ_X and σ_Y are the respective Standard Deviations between X and Y.
Pearson's ρ is considered to be a reliable measurement of correlation when the variables of interest are known to follow Normal and Elliptical distributions. Thus, Pearson's ρ should be used with caution when considering variables known to exhibit nonlinear behavior.

As we will see in a couple sections, we will need the so-called Kendall's τ^K correlation metric to define tail-dependence so, even though we won't be testing our alarm system with it, we will go ahead and define it here, for convenience.

Definition 6.2 (Kendall's Tau Rank Correlation: τ^K)**.** Given two random variables X and Y, both of length n, the Kendall's τ^K correlation is defined as

$$\tau^K(X,Y) = \frac{2}{n(n-1)} \sum_{i<j} sgn(X_i - X_j)(Y_i - Y_j) \tag{6.2}$$

for the sgn function $sgn(x) = x^{-1}|x|$.

Kendall's τ^K belongs to the class of "rank" correlation metrics with some interesting properties that can be found in the literature including the seminal work of KENDALL [1938]. For now, we move onto the so-called Gini Correlation, which deserves special attention, next.

6.2.1 | Gini Correlation

The standard Gini Correlation plays an important rule in describing the correlation structure between random variables under the presence of heavy-tailed phenomenon. Loosely speaking, the Gini Correlation between two Random Variables X and Y is based on the Covariance between X and the rank of the other $G(Y)$, that is $\mathbb{C}ov[X, G(Y)]$. Therefor, for a pair of Random Variables (X, Y), there will be two Gini Correlations: one based on $\mathbb{C}ov[X, G(Y)]$, and the other based on $\mathbb{C}ov[Y, F(X)]$.

Definition 6.3 (Gini Correlation)**.** The standard Gini Correlation(s) $\gamma_1(X,Y)$ and $\gamma_2(X,Y)$ are defined as

$$\gamma_1(X,Y) \coloneqq \frac{\mathbb{C}ov(X, G(Y))}{\mathbb{C}ov(X, F(X))} \quad \& \quad \gamma_2(X,Y) \coloneqq \frac{\mathbb{C}ov(Y, F(X))}{\mathbb{C}ov(Y, G(Y))} \tag{6.3}$$

where $F(X)$ and $G(Y)$ denote the marginal Distribution Functions for X and Y, respectively.

In the literature, see for example Vanderford et al. [2020], it is common to simply denote the standard Gini Correlation between X and Y using $\gamma(X,Y)$ where $\gamma(X,Y) \neq \gamma(Y,X)$ is an Asymmetric

measure of dependence. However, one can easily "symmetrize" the standard Gini Correlation by taking either the Arithemtic, or Geometric, mean between $\gamma_1(X,Y)$ and $\gamma_2(X,Y)$ as per Defintion 6.4. We borrow from Vanderford et al. [2020] the following defintion.

Definition 6.4 (Symmetrized Gini Correlation: γ)**.** We define the "Symmetrized" Gini Correlation $\gamma(X,Y)$, between two Random Variables X, Y as,

$$\gamma(X,Y) \coloneqq \frac{1}{2}(\gamma_1(X,Y) + \gamma_2(X,Y)) \quad \text{or} \quad \gamma(X,Y) \coloneqq \sqrt{\gamma_1(X,Y)\gamma_2(X,Y)} \tag{6.4}$$

corresponding to the Arithmetic and Geometric means, respectively.

"Symmetrizing" the standard Gini Correlation enables us to construct a symmetric (Gini) Correlation Matrix γ that retains the Spectral properties allowing us to build the Quantum Lorenz Order of the corresponding Temporal State-Space of (Gini) Correlation Matrices $\Gamma_\Delta(\mathcal{S})$.

Definition 6.5 (Gini Correlation Matrix)**.**

$$\gamma_{i,j} \coloneqq \begin{cases} \gamma(X_i, X_j), & \text{if } i \neq j \\ 1, & \text{otherwise} \end{cases} \tag{6.5}$$

where $\gamma(X,Y)$ is as Definition 6.4.

As for efficiently computing $\gamma(X,Y)$, that is a seperate problem. However, a clever solution is proposed in Sang et al. [2016]. Following this approach, we can estimate the standard Gini Correlations $\gamma_1(X,Y)$, and $\gamma_2(X,Y)$ by using a ratio of U-statistics.

Definition 6.6 (Gini Correlation Estimation)**.** We express $\gamma_1(X,Y)$, and $\gamma_2(X,Y)$ as per Defintion 6.3 using the following ratio of U-statistics

$$\gamma_1(X,Y) = \frac{U_1}{U_2} \quad \& \quad \gamma_2(X,Y) = \frac{U_3}{U_4} \tag{6.6}$$

where U_1, U_2, U_3 and U_4 are computed using the following formulae

$$\begin{aligned} U_1 &\coloneqq \frac{1}{4\binom{n}{2}} \sum_{i=1}^{n} (2i - 1 - n) X_{Y_{(i)}} \quad \& \quad U_2 \coloneqq \frac{1}{4\binom{n}{2}} \sum_{i=1}^{n} (2i - 1 - n) X_{(i)} \\ U_3 &\coloneqq \frac{1}{4\binom{n}{2}} \sum_{i=1}^{n} (2i - 1 - n) Y_{X_{(i)}} \quad \& \quad U_4 \coloneqq \frac{1}{4\binom{n}{2}} \sum_{i=1}^{n} (2i - 1 - n) Y_{(i)} \end{aligned} \tag{6.7}$$

where $X_{(i)}$ and $Y_{(i)}$ denote the i^{th} (ascending) Order-Statistic for X and Y while $X_{Y_{(i)}}$ and $Y_{X_{(i)}}$ denote the X and Y corresponding to the $X_{(i)}$ and $Y_{(i)}$ Order-Statistics, respectively.

With Definitions 6.3, 6.4, 6.5, and 6.6, we now have all of the machinery in place to add yet another Correlation Metric to our arsenal with corresponding Temporal State-Space $\Gamma_\Delta(\mathcal{S})$, as per Definition 3.6.

6.2.2 | Tail Dependence

Steering away from the notion of Correlation, we can also consider the concept of "Tail-Dependence". In Probability, Tail-Dependence aims to provide a measure of how much a pair of Random Variables X and Y move with each other in the tails of their respective distributions. Tail-Dependence can be measured in the Upper, and Lower, tails of the distribution according to Definitions 6.7 and 6.8 below.

Definition 6.7 (Upper Tail Dependence: λ_U)**.**

$$\lambda_U \coloneqq \lim_{p \to 1} \mathbb{P}[X > F^{\leftarrow}(p) | Y > G^{\leftarrow}(p)] \tag{6.8}$$

Definition 6.8 (Lower Tail Dependence: λ_L)**.**

$$\lambda_L \coloneqq \lim_{p \to 0} \mathbb{P}[X \leqslant F^{\leftarrow}(p) | Y \leqslant G^{\leftarrow}(p)] \tag{6.9}$$

It is relatively straightforward to construct nonparametric estimators for λ_U and λ_L using their respective Order-Statistics, $X_{(i)}$ and $Y_{(i)}$, by considering the Joint-Indicator function of the co-appearance of observations X_i and Y_i above (or below) a certain threshold k in $X_{(k)}$ and $Y_{(k)}$. To this end, we use the following Nonparametric estimator described in Definition 6.9.

Definition 6.9 (Nonparametric Estimator for Tail-Dependence)**.** We estimate λ_U and λ_L with $\tilde{\lambda}_U = \tilde{\lambda}_U(k^*)$ and $\tilde{\lambda}_L = \tilde{\lambda}_L(k^*)$ where $\tilde{\lambda}_U(k)$ and $\tilde{\lambda}_L(k)$ are computed using

$$\begin{aligned} \tilde{\lambda}_U(k) &= \frac{1}{k}\sum_{i=1}^{n} \mathbb{I}\{X_i > X_{(n-k)}, Y_i > Y_{(n-k)}\} \\ \tilde{\lambda}_L(k) &= \frac{1}{k}\sum_{i=1}^{n} \mathbb{I}\{X_i \leqslant X_{(n-k)}, Y_i \leqslant Y_{(n-k)}\} \end{aligned} \tag{6.10}$$

where k^* is chosen such that $\forall\tilde{\lambda}_{[.]}(k) \in [\tilde{\lambda}_{[.]}(k^* - \varepsilon), \tilde{\lambda}_{[.]}(k^* + \varepsilon)]$ is a relatively stable interval.

The approach in the above definition 6.9 works fine if the number of variables we are considering are few. In the context of this work, however, we are considering the dependence between a number of variables $d \sim \mathcal{O}(100)$. This makes the above approach impractical. However, we can exploit Copulas that are known to exhibit tail-dependence to compute the tail-dependence between the returns of financial assets $(\mathcal{S}_i, \mathcal{S}_j) \in \mathcal{S}$. Simply put, a copula $C(u, v)$ is nothing more than a bi-variate cumulative distribution function of the form

$$C(u, v) \coloneqq \mathbb{P}[X \leqslant F^{\leftarrow}(u), Y \leqslant G^{\leftarrow}(v)] \tag{6.11}$$

with (left) inverse cumulative distribution functions $F^{\leftarrow}(X), G^{\leftarrow}(Y)$ for X and Y, respectively. Two well-known copulas that exhibit Upper, and Lower, tail-dependence are the Gumbel and Clayton copulas, respectively.

Definition 6.10 (Tail-Dependence of the Gumbel and Clayton Copulas)**.** Gumbel and Clayton copulas that "join" two random variables X and Y, denoted by $C^G_\theta(u, v)$ and $C^C_\theta(u, v)$, respectively, take the following form

$$\begin{aligned} C^G_\theta(u, v) &\coloneqq e^{-[(-\ln(u))^\theta + ((-\ln(v))^\theta]^{\frac{1}{\theta}}}, \qquad \theta \in [-1, \infty) \\ C^C_\theta(u, v) &\coloneqq \max\{[u^{-\theta} + v^{-\theta} - 1]^{-\frac{1}{\theta}}, 0\}, \qquad \theta \in [1, \infty) \end{aligned} \tag{6.12}$$

where, by taking the appropriate limits below, we obtain the Upper and Lower tail-dependencies

$\lambda^L[C_\theta^G(u,v)]$ and $\lambda^U[C_\theta^C(u,v)]$, for the Gumbel and Clayton copulas, respectively

$$\begin{aligned}\lim_{p\to 1}\frac{C_\theta^G(p,p)}{1-p} &\to \lambda^U[C_\theta^G(u,v)] \coloneqq 2-2^{\frac{1}{\theta}}\\ \lim_{p\to 0}\frac{C_\theta^C(p,p)}{p} &\to \lambda^L[C_\theta^C(u,v)] \coloneqq \frac{1}{2^{\frac{1}{\theta}}}\end{aligned} \tag{6.13}$$

since $\lambda^L[C_\theta^G(u,v)] = 0$ and $\lambda^U[C_\theta^C(u,v)] = 0$.

Naturally, we need to find an appropriate parameter θ to exploit equation(s) 6.13. Interestingly enough, there is an explicitly calculable relationship between θ and the Kendall's tau correlation $\tau_K(X,Y)$. We state this (without proof) with proposition 6.11 below.

Proposition 6.11. *The Gumbel and Clayton copulas can be parameterized using the relation(s)*

$$\theta_G = \frac{1}{1-\tau_K(X,Y)} \quad \& \quad \theta_C = \frac{2\tau_K(X,Y)}{1-\tau_K(Y,Y)} \tag{6.14}$$

where $\lambda^L[C_G(u,v)]$ *and* $\lambda^U[C_C(u,v)]$ *become*

$$\begin{aligned}\lambda^U[C_G(u,v)] &\coloneqq 2-2^{\frac{1}{\theta}}, \qquad \theta_G = \frac{1}{1-\tau_K(X,Y)}\\ \lambda^L[C_C(u,v)] &\coloneqq \frac{1}{2^{\frac{1}{\theta}}}, \qquad \theta_C = \frac{2\tau_K(X,Y)}{1-\tau_K(X,Y)}\end{aligned} \tag{6.15}$$

with θ_G *and* θ_C *corresponding to the Gumbel and Clayton copulas, respectively.*

Hence, we can build a upper (lower) tail-dependence matrix λ^U (λ^L) by simply using the correlation matrix associated to the Kendall's τ metric τ^K. For completeness, we formally define the upper (lower) tail-dependence matrix λ^U (λ^U) below, with definition.

Definition 6.12 (Tail-Dependence Matrix)**.**

$$\lambda_{i,j}^U = \begin{cases} 2-2^{\frac{1}{\theta}}, \text{ if } i \neq j \\ 0, \text{ otherwise}\end{cases} \tag{6.16}$$

Without looking too far ahead, we suspect this parameterization of the Gumbel (Clayton) copluas C_G (C_C) to provide interesting behavior. Reason being, λ^U is a nonlinear function of Kendall's τ.

Furthermore, λ^{U} retains the same spectral properties as any other correlation metric. We illustrate why with corollary

Corollary 6.13. *The diagonal entries of the upper (lower) tail-dependence matrix* $\mathtt{diag}(\lambda^{U}) = \vec{1}$ *(*$\mathtt{diag}(\lambda^{L}) = \vec{1}$ *are equal to unity.*

Proof. Consider $\lambda^{U}_{i,j} = 2 - 2^{\frac{1}{\theta}}$ with $\theta = (1 - \tau^{K}_{i,j})^{-1}$. If $i = j$, then $\tau^{K}_{i,j} = 1$. In which case we have,

$$\lim_{\tau^{K}_{i,j} \to 1} \theta \to \infty. \tag{6.17}$$

and thus $2 - 2^{\frac{1}{\theta}} \to 2 - 1 = 1$. A similar argument goes for λ^{L}. □

Hence, we have $\mathrm{Tr}[\lambda^{U}] = d$ for d the number of assets in a portfolio. Trivially, since λ^{U} is symmetric and $\lambda^{U}_{i,j} \geqslant 0$, we know that λ^{U} is a PSD matrix with eigenvalues $\lambda(\lambda^{U}) \geqslant 0$. Thus, the spectrum of λ^{U} too forms a distribution from which a proper lorenz curve is defined.

Note that the above argumentation carries over to the lower tail-dependence matrix λ^{L} as well. We encourage the reader to verify for themselves. Lastly, and again for completeness, we define the temporal state space for λ^{U} and λ^{L} below

Definition 6.14 (Temporal State-Space (Tail-Dependence))**.** We define the temporal state-space(s)

$$\begin{aligned} \Lambda^{U}_{\Delta}(\mathcal{S}; I) &:= \{\lambda^{U}_{k} = \lambda^{U}_{\iota_k}(\mathcal{S}) : \iota_k = [kh, kh + m] \in I, \forall k \in \{0, \dots, n-1\} \\ \Lambda^{L}_{\Delta}(\mathcal{S}; I) &:= \{\lambda^{L}_{k} = \lambda^{L}_{\iota_k}(\mathcal{S}) : \iota_k = [kh, kh + m] \in I, \forall k \in \{0, \dots, n-1\} \end{aligned} \tag{6.18}$$

associated to the upper and lower tail dependence matrices λ^{U}, and λ^{L}.

6.3 | Defining Crashes using Extreme Value Theory

In this section, we apply a slightly different methodology to distinguishing Stock Market Drawdowns from the Price Coarse-Graining algorithm of Didier Sornette [2006]: Extreme Value Theory (EVT). We keep the definition 5.1 of drawdowns, however, we try to fit our drawdowns $\mathcal{D}$ to the so-called "Generalized Pareto Distribution" (GPD) of definition 6.15

$$\mathcal{D} \sim G_{\theta} \tag{6.19}$$

instead of the "Stretched Exponential" of definition 5.2. The GPD $G_\theta(x;\theta)$ is parameterized by the $\theta = (\gamma;\mu,\sigma)$-tuple with "shape", "location", and "scale" parameters γ, μ, and σ, respectively. We stress that the shape $\gamma \in (-\infty,\infty)$ is the parameter of interest in the present analysis.

Definition 6.15 (Generalized Pareto Distribution)**.**

$$G(x;\gamma,\mu,\sigma) = \begin{cases} 1-(1+\gamma\frac{x-\mu}{\sigma})^{-\frac{1}{\gamma}}, & \gamma \neq 0 \\ 1-e^{-\frac{x-\mu}{\sigma}}, & \gamma = 0 \end{cases} \tag{6.20}$$

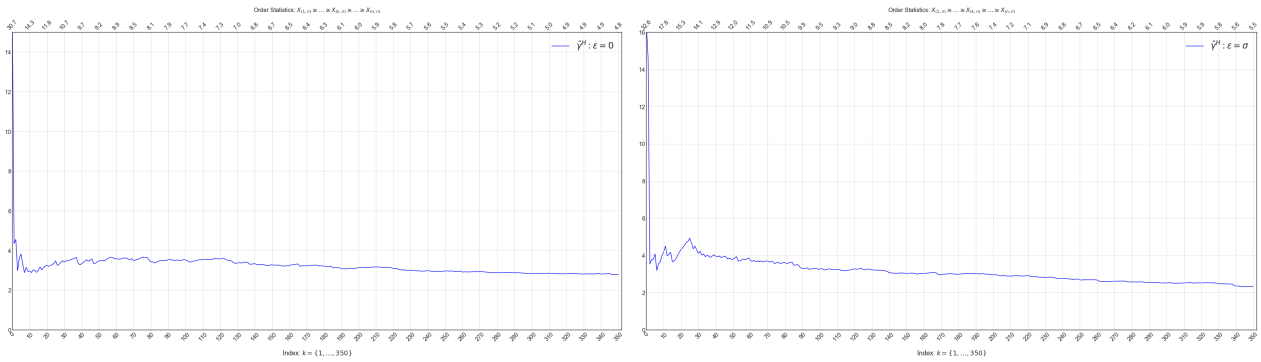
The GPD is widely used in the study of fat-tailed phenomenon. From the standpoint of quantitative risk management, the GPD is often preferred to other parametric families of distributions when computing risk-measures such as the Value-at-Risk (VaR), Expected Shortfall (ES) or any other quantile $Q(p) = F^{-1}(p)$ on the distribution $F(L)$ of losses L. Reason being, the GPD can model as fat (or thin) of tails as the data wants to suggest, unlike other distributions, such as the normal $\mathcal{N}(\mu,\sigma^2)$, which often underestimate (or even overestimate) the severity of the tail. Thus, for fitting purposes, if one wants to study the distribution of drawdowns $\mathcal{D} \sim F$, the GPD $F = G_\theta$ would be a natural candidate. In which case, our first step would be to find γ which can be achieved with Maximum Likelihood Estimation (MLE)

$$\mathcal{L}(\theta;X) = \prod_{i=1}^{n} g_\theta(x_i)\mathbb{1}_{\{\gamma(x_i-\mu)\geqslant-\sigma\}}, \qquad \theta = \{\gamma,\mu,\sigma\} \tag{6.21}$$

for the likelihood function $\mathcal{L}(\theta;X)$, density $g_\theta(x)$, and parameter set $\theta = (\gamma,\mu,\sigma)$. The reason for the indicator function $\mathbb{1}_{\{\gamma(x_i-\mu)\geqslant-\sigma\}}$ is easily deduced from the definition of the GPD G_θ. No analytical solution to equation 6.21 exists, however, several estimators based on the so-called Order Statistics $\{X_{(i,n)}\}_{i=1}^n$ for the raw data $X = \{X_1,\ldots,X_n\}$ do exist

$$X_{(n,n)} \leqslant \cdots \leqslant X_{(k,n)} \leqslant \cdots \leqslant X_{(1,n)} \tag{6.22}$$

with $X_{(n,n)}, X_{(k,n)}, X_{(n,n)}$ denoting the minimum, k^{th}-maximum, and maximum, respectively. We introduce one of the more well-known estimator, the Hill Estimator $\hat{\gamma}^H$, with definition 6.16.



(a) Hill estimator $\hat{\gamma}_0^H$ **(b)** Hill estimator $\hat{\gamma}_\sigma^H$

Figure 6.2: Hill estimators for the drawdowns $\mathcal{D}_0$ and $\mathcal{D}_\sigma$ using $\varepsilon = \{0, \sigma\}$ truncated after $k = 350$

Definition 6.16 (Hill Estimator γ^H)**.**

$$\hat{\gamma}^H(k; n) = \frac{1}{k}\sum_{i=1}^{k} \log\left(\frac{X_{(i,n)}}{X_{(k,n)}}\right) \tag{6.23}$$

As suggested by Didier Sornette [2006], we computed the Hill estimator $\hat{\gamma}^H$ for the distribution(s) of drawdowns $\mathcal{D}_\varepsilon$ for $\varepsilon = \{0, \frac{\sigma}{4}, \frac{\sigma}{2}, \sigma\}$. We denote with $\hat{\gamma}_\varepsilon^H$ the Hill estimator associated to each respective choice for ε. The estimators for the extreme cases $\varepsilon = 0$ and $\varepsilon = \sigma$, that is $\hat{\gamma}_0^H$ and $\hat{\gamma}_\sigma^H$, are plotted for the readers convenience in figure 6.2. Referring to this figure, we notice different behavior for the choices $\varepsilon = \{0, \sigma\}$. The key to inferring $\hat{\gamma}^H$ from the Hill plot is to look for stability. Looking at these two plots, we're tempted to say that $\hat{\gamma}_0^H$ stabilizes somewhere between the 30^{th} and 40^{th} order statistics $\mathcal{D}_{0,(30,n)} = -.09$ and $\mathcal{D}_{0,(40,n)} = -.11$. Likewise, for $\hat{\gamma}_\sigma^H$ we notice stabilization somewhere between $\mathcal{D}_{\sigma,(50,n)} = -.11$ and $\mathcal{D}_{\sigma,(60,n)} = -.13$. If this were true, we'd be in a position to classify drawdowns $\mathcal{D}_0 \leqslant -.10$ and $\mathcal{D}_\sigma \leqslant -.12$ as "crashes" since these are the drawdowns that characterize the tails of their respective distributions. The question becomes, can we verify this.

To address this, we can leverage the following relationship between the shape γ of the GPD G_θ: The shape γ of a GPD $G_\theta(x; \theta = (\gamma, \mu, \sigma))$ fully specifies the finiteness of the moments (mean, variance, skewness, kurtosis) for a Random Variable $X \sim G_\theta(x; \theta)$. We state this fact with proposition 6.17, without proof. The proof is simple enough using basic integration.

Proposition 6.17. *If* $X \sim G(x; \gamma, \mu, \sigma)$*, for an I.I.D. Sequence* $X = \{X_1, \dots, X_n\}$*, then*

$$\mathbb{E}[|X|^p] < \infty \Leftrightarrow \gamma > p \tag{6.24}$$

for any $\forall p = \{1, 2, \dots, k\}$.

That is to say, if the GPD $G_\theta(x; \theta)$ is parameterized by a (say) $\gamma = 1.7$, the 1^{st}-Moment (the mean) might be finite, but all the others (Variance, Skewness, Kurtosis)

$$\mathbb{E}[|X|^2], \mathbb{E}[|X|^3], \mathbb{E}[|X|^4], \cdots \not< \infty \tag{6.25}$$

would all be infinite. If $\gamma = 0.9$, on the other hand, even the mean would be infinite $\mathbb{E}[|X|] \to \infty$. Luckily for us, we can look at the so-called Maximum to Sum (MS) ratio for the drawdowns $\mathcal{D}_\varepsilon$ to understand the behavior of the moments. Looking ahead, if the MS ratio, which we will define next, tells us which moments for the distribution of drawdowns are finite, then we'd be in a position to pinpoint the region of stabilization in the Hill plot from before.

Definition 6.18 (MS Ratio)**.** For an I.I.D. Sequence of N Random Variables $X = \{X_1, \dots, X_N\}$ we denote $|X|^p = \{|X_1|^p, \dots, |X_N|^p\}$ and define the Maximum-to-Sum (MS) Ratio $R_n(p)$ with

$$R_n(p) = \frac{M_n(p)}{S_n(p)}, \qquad S_n(p) = \sum_{i=1}^{n} |X_i|^p \tag{6.26}$$

for $n \leqslant N$ and $M_n(p) = \max(|X|^p)$.

To take advantage of the MS ratio, we can use the following proposition 6.19 which we state, without proof. It is a relatively straightforward application of the Law of Large Numbers (LLN).

Proposition 6.19. *By the Law of Large Numbers (LLN), we have for the MS-Ratio* $\frac{M_n(p)}{S_n(p)}$

$$\lim_{n \to \infty} \frac{M_n(p)}{S_n(p)} = 0 \qquad \Leftrightarrow \qquad \mathbb{E}[|X|^p] < \infty \tag{6.27}$$

for any moment $p = \{1, 2, \dots, k\}$.

That is to say, in a similar fashion to the Hill estimator, we can plot the MS ratio $M_n(p)/S_n(p)$ as a function of n and check for convergence. If the ratio $M_n(p)/S_n(p)$ seems to be converging to 0, for large n, then we can say that moment p is finite. We visualize this with what we call

the MS plot. As before, we plot the two extreme cases: the MS ratio for the drawdowns $\mathcal{D}_\varepsilon$ for $\varepsilon = \{0, \sigma\}$. We begin by plotting the MS ratios for the first 4 moments $p = \{1, 2, 3, 4\}$ for the ε-drawdowns, for $\varepsilon = 0$, in figure 6.3.

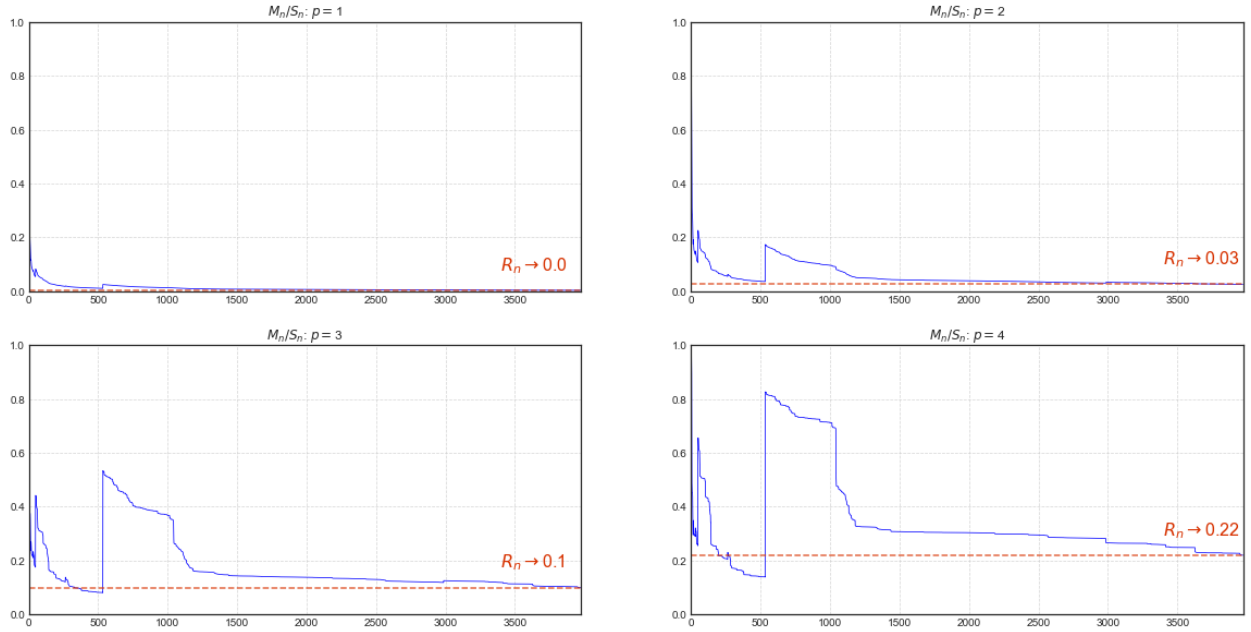


Figure 6.3: MS-Plot for DJIA Drawdowns $\mathcal{D}_0$ (1900-2020) for $p = 1$ (top left) up to $p = 4$ (bottom right)

Referring to figure 6.3 we notice strong convergence for the first 3 moments $p = \{1, 2, 3\}$ of the distribution of ε-drawdowns $\mathcal{D}_0$ for $\varepsilon = 0$. However, as for the 4^{th} moment, we do not see convergence at all. From this we conclude that the 4^{th} moment $p = 4$ for the ε-drawdowns $\mathcal{D}_0$ ($\varepsilon = 0$) is infinite. We use this information to look for stabilization in the Hill plot for $\hat{\gamma}_0^{\text{H}}$ from before in figure 6.2(a). Indeed, since the moment $p = 3$ is finite, but $p = 4$ is not, the Hill plot for $\hat{\gamma}_0^{\text{H}}$ should stabilize somewhere between $\hat{\gamma}_0^{\text{H}} \in [3, 4]$. Now we define any drawdown below the threshold, for which stabilization first occurs, as a crash. We will see what this looks like in a moment. For now, we take a look at the MS plot for ε-drawdowns for $\varepsilon = \sigma$ in figure 6.4.

Referring to figure 6.4 we notice strong convergence for all 4 moments $p = \{1, 2, 3, 4\}$ of the distribution of ε-drawdowns for $\varepsilon = \sigma$. Unlike for the drawdowns associated to $\varepsilon = 0$, we conclude that the 4^{th} moment $p = 4$ is finite. Thus, we will look for stabilization in the Hill plot for $\hat{\gamma}_0^{\text{H}}$ somewhere above the value for $\hat{\gamma}_\sigma^{\text{H}} = 4$. Now we define any drawdown below the associated threshold as a crash. Again, this will become clearer in a moment when we go back to the Hill plots, which we do next.

We bring the Hill plots from figure 6.2 back in figure 6.5. In figure 6.5(a) we highlight area for

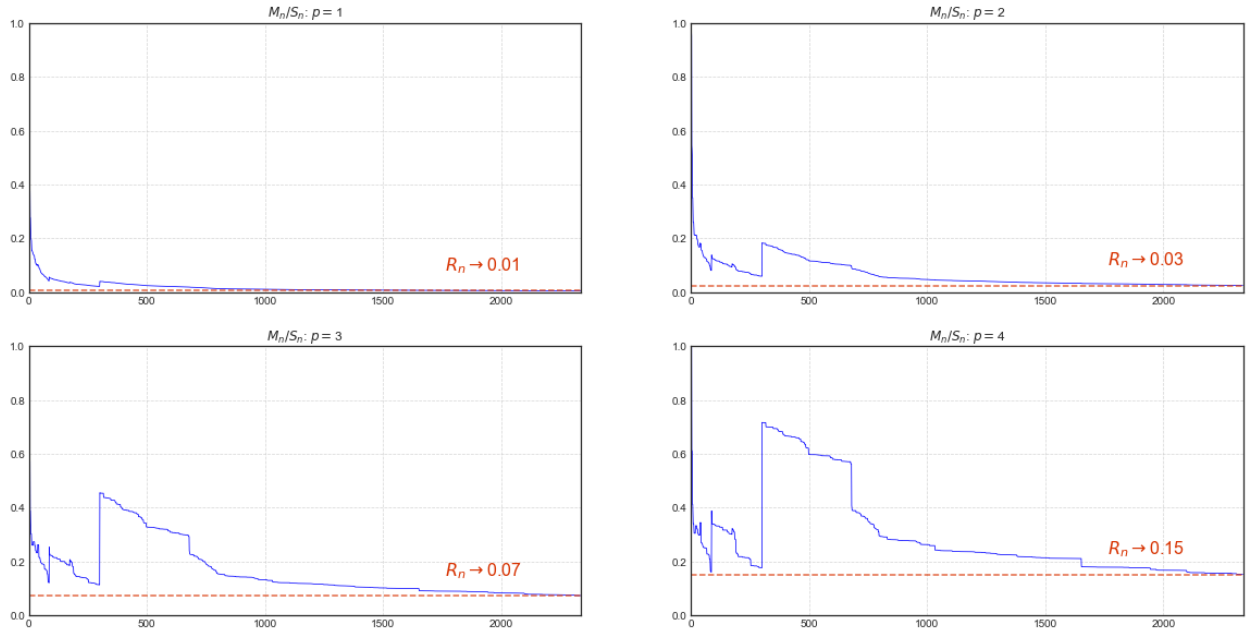
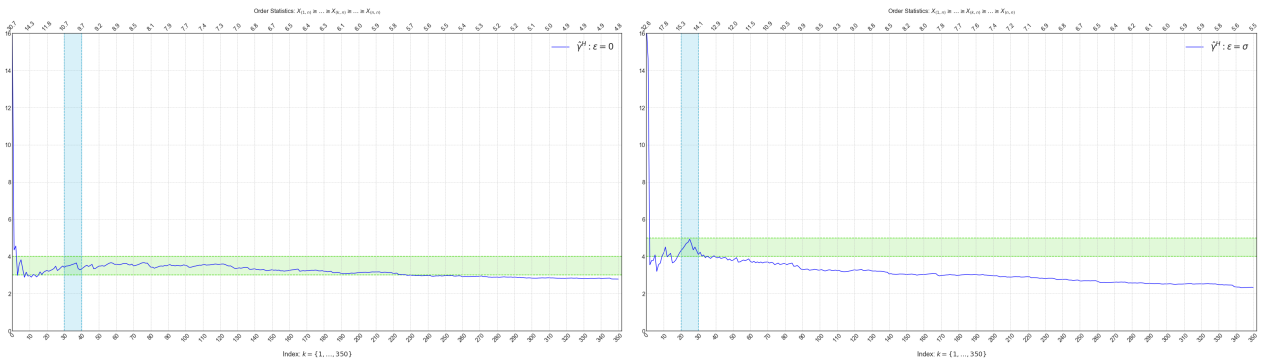


Figure 6.4: MS-Plot for DJIA Drawdowns $\mathcal{D}_\sigma$ (1900-2020) for $p = 1$ (top left) up to $p = 4$ (bottom right)



(a) Hill estimator $\hat{\gamma}_0^H$ **(b)** Hill estimator $\hat{\gamma}_\sigma^H$

Figure 6.5: Hill estimators for the drawdowns $\mathcal{D}_0$ and $\mathcal{D}_\sigma$ using $\varepsilon = \{0, \sigma\}$ truncated after $k = 350$

$\hat{\gamma}_0^H \in [3, 4]$ in green. Furthermore, we notice that the plot seems to stabilize somewhere between the 30^{th} and 40^{th} order statistics indeed. Drawdowns above the threshold of $k = 30$ correspond to ε-drawdowns below -0.09. Hence, for the distribution of ε-drawdowns $\mathcal{D}_0$, we classify any drawdown $\mathcal{D}_0 \leqslant -0.09$ as a crash. Similarly, for the ε-drawdowns for $\varepsilon = \sigma$, we highlight $\hat{\gamma}_\sigma^H \in [4, 5]$ in green since the 4^{th} moment here was shown to be finite. The associated Hill plot $\hat{\gamma}_\sigma^H$ seems to stabilize somewhere between the 20^{th} and 30^{th} order statistics. As we can see along the secondary (top) axis, this threshold corresponds to ε-drawdowns $\mathcal{D}_\sigma \leqslant -0.14$. Hence, any ε-drawdown, when $\varepsilon = \sigma$, below -0.14 will be classified as a crash.

We repeated the same analysis for ε-drawdowns, when $\varepsilon = \{\frac{\sigma}{4}, \frac{\sigma}{2}\}$, as well. In these two cases we found the appropriate thresholds to be $\mathcal{D}_{\frac{\sigma}{4}} \leqslant -0.10$ and $\mathcal{D}_{\frac{\sigma}{2}} \leqslant -0.11$ for $\varepsilon = \frac{\sigma}{4}$ and $\varepsilon = \frac{\sigma}{2}$, respectively. Indeed it makes sense for the threshold associated to the larger $\varepsilon = \frac{\sigma}{2}$ to be greater than the threshold for $\varepsilon = \frac{\sigma}{4}$ since any ε-drawdown obeys the relation $\mathcal{D}_{\frac{\sigma}{2}} \geqslant \mathcal{D}_{\frac{\sigma}{4}}$, necessarily, as per the definition of an ε-drawdown. It is relatively straight forward for the reader to convince themselves why this is the case. We give this a formal definition.

Definition 6.20 (Crash)**.** Let $\mathcal{D}_{(n,n),\varepsilon} \leqslant \mathcal{D}_{(k,n),\varepsilon} \leqslant \mathcal{D}_{(1,n),\varepsilon}$ denote the order-statistics for the set of ε-drawdowns $\mathcal{D}_\varepsilon$, associated to some financial asset. We define the set of all crashes $\mathcal{C}_\varepsilon$ as

$$\mathcal{C}_\varepsilon = \{\mathcal{D}^*_\varepsilon \in \mathcal{D}_\varepsilon : \mathcal{D}^*_\varepsilon \leqslant \mathcal{D}_{(k^*,n),\varepsilon}\} \tag{6.28}$$

where k^* is the index in the order statistics for which the Hill plot begins to stabilize.

Hopefully the above definition clarifies to the reader what we mean, exactly, by a crash. For convenience, we package the findings from the analysis performed on the set of ε-drawdowns for $\varepsilon = \{0, \sigma/4, \sigma/2, \sigma\}$ in the following table. These values for $\mathcal{D}_{(k^*,n),\varepsilon}$ correspond to the events we want our alarm system to predict.

Crashes and the Finiteness of Moments					
ε	1^{st}	2^{nd}	3^{rd}	4^{th}	$\mathcal{D}_{(k^*,n),\varepsilon}$
$\varepsilon = 0$	Finite	Finite	Finite	Finite	$-.09$
$\varepsilon = \frac{\sigma}{4}$	Finite	Finite	Finite	Finite	$-.10$
$\varepsilon = \frac{\sigma}{2}$	Finite	Finite	Finite	Finite	$-.11$
$\varepsilon = \sigma$	Finite	Finite	Finite	Infinite	$-.14$

Table 6.1: ε-drawdown moments

Given the time period for which we are building our model (1985-2020), our analysis also showed that the different choices for ε all identified the same crashes. On the other hand, they did not agree on the relative magnitudes of said crashes. This is good news for us since this suggests that the particular choice for ε, in defining ε-drawdowns, is arbitrary for the purposes of identifying and predicting crashes. We display the crashes associated to the ε-drawdowns $\mathcal{D}_\varepsilon$ for the particular choice of $\varepsilon = \sigma$, arbitrarily, in figure 6.6.

In this figure we notice some clustering of crashes around the infamous 1987 "Black Monday", 2008 "Financial Crisis", and 2020 "COVID-19" crashes. Recall the special case, from chapter 5 (figure 5.2), that we could no clearly distinguish the COVID-19 crash as an outlier to the stretched

Figure 6.6: Stock Market Crashes: DOW Jones Industrial Average (1980-2020)

exponential $N(\mathcal{D})$. This is because COVID-19 consisted of a sequence of large drawdowns all of which were interrupted by short rallies larger than any reasonable choice for ε. In fact, any ε large enough to identify the entirety of the COVID-19 crash would define the entire Dow Jones time series, itself, as a crash. We deal with this case in the next section and propose the (ϵ, δ)-drawdown, δ signifying some temporal parameter.

6.3.1 | Time-Price Coarse-Graining Algorithm

In this short section we analyze the special case of the COVID-19 crash from 2020. As mentioned before, defining the entirety of this crash is a nontrivial task as several extreme drawdowns were witnessed over a very short period of time. We display the 4 crashes identified by the EVT analysis from the previous section, for $\varepsilon = \sigma$, just as in figure 6.6, with figure 6.7.

Simply allowing these crashes to stand, as they are seen in figure 6.7, will introduce a potential bias into our model. Indeed, if we're sampling correlation matrices over 10-day sliding windows, the sampling of a single correlation matrix may predict all three of the latter four crashes observed between March 3rd to March 18th. Since the classification of crashes is a severely imbalanced data set (very few crashes), allowing this phenomenon to stand might make our model look a lot better than it actually is. Thus, to make as strong of an argument as possible, and to make sure our model

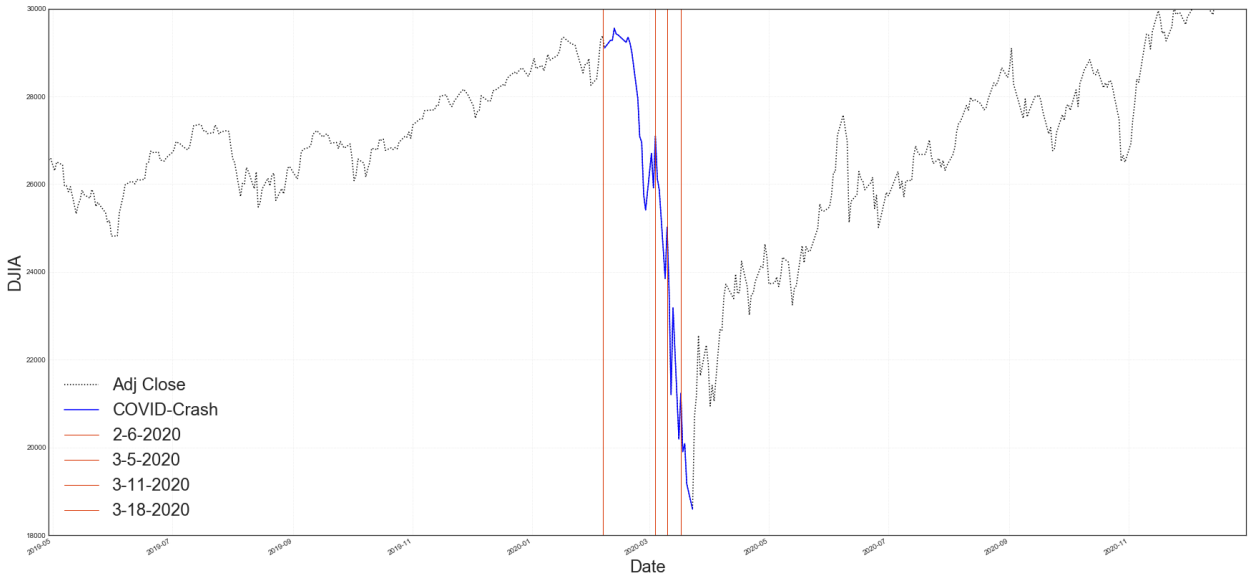


Figure 6.7: CODIV-19 phenomenon according to the ε-drawdown for $\varepsilon = \sigma/2$

reflects the true psycho-temporal context of financial market behavior, we introduce the (ε, δ)-drawdown to cluster these three crashes, or potentially all four of these crashes, together into just one which is exactly how people characterize the COVID-19 crash.

Definition 6.21 ((ε, δ)-Drawdown)**.** The (ε, δ)-Drawdown, works as follows:

- Working backwards recursively, consider each ordered pair $(\mathcal{C}_{t_{i-1},\varepsilon}, \mathcal{C}_{t_i,\varepsilon}) \in \mathcal{C}_\varepsilon$ and identify any $\mathcal{C}_{t_i,\varepsilon}$ such that $t_i - t_{i-1} \leqslant \delta$.
- Define $\overline{\mathcal{P}}_{t_{i-1}}$ as the local maximum for the financial asset associated to $\mathcal{C}_{t_{i-1},\varepsilon}$, along with $\underline{\mathcal{P}}_{t_i}$ the local minimum associated to $\mathcal{C}_{t_i,\varepsilon}$, and set $\mathcal{C}_{t_{i-1},\varepsilon} = \frac{\overline{\mathcal{P}}_{t_{i-1}} - \underline{\mathcal{P}}_{t_i}}{\overline{\mathcal{P}}_{t_{i-1}}}$ and discard $\mathcal{C}_{t_i,\varepsilon}$ from $\mathcal{C}_\varepsilon$.

We order ε in the (ε, δ) tuple first to stress that the ε-drawdown algorithm needs to be implemented first. Note that the (ε, δ)-drawdown can be implemented on the drawdowns $\mathcal{D}_\varepsilon$ as well. We just choose to do so on the already identified crashes $\mathcal{C}_\varepsilon$ given the nature of this work. We display the resulting (ε, δ)-drawdown, corresponding to the COVID-19 crash, in figure 6.8.

In the above figure, we happened to choose a $\delta = 10$, or roughly two weeks, since trading weeks are typically 5 days long. As a result, the latter three crashes between March 3$^{\text{rd}}$ to March 18$^{\text{th}}$ were grouped into one beginning on March 3$^{\text{rd}}$. We ignore the magnitude of this newly defined crash since its not important to us. We chose $\delta = 10$ so as to immunize our model

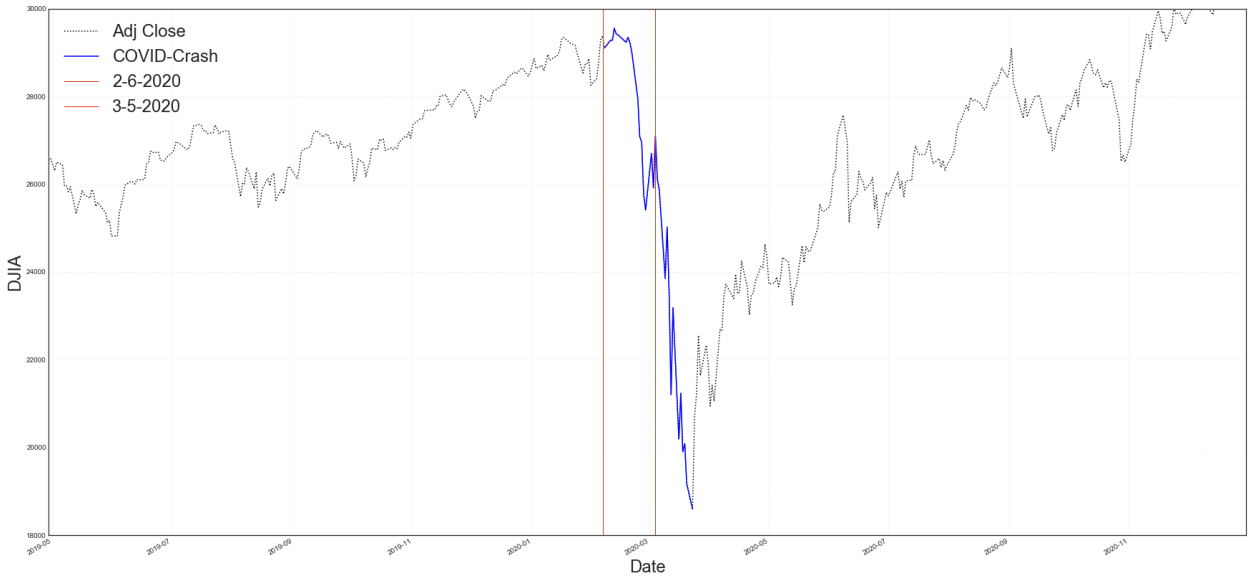


Figure 6.8: COVID-19 coarse grained according to the (ε, δ) algorithm

from the implicit (potentially) bias mentioned earlier. The time interval associated to the crashes between February 6^{th} and March 3^{rd} is big enough to consider the two crashes separately. Indeed, we believe most investors would want an alarm system to be able to predict major market events separated by a month, as is the case here. Hence, if our model predicts both, the alarm system will be accurately characterizing the risk present in the market over this time period. This concludes chapter 6, and we will move on to chapter 7 where we introduce our model.

7

The Model

In this chapter, we take everything we've done so far and build the Reinforced Urn Process (RUP) that we will be using for our alarm system. To build our RUP we make a couple generalizations to the framework developed in chapter 4. Additionally, we want our alarm system to run in an online fashion, only using historical information to make predictions about the future. Doing this requires attention to detail. Lastly, we revisit the RMT filtering schemes, which we introduced in chapter 2, within the context of quantum majorization. Hence, in section 7.1, we generalize the detection regions, stopping rule, in addition to generalizing the reinforcement scheme of the Urns themselves. Second, in section 7.2 we explain how we sample, and assign, correlation matrices $\rho_k \in P_\Delta(\mathcal{S}, I)$ to clusters in an online and temporal fashion. Lastly, in section 7.4, calculate the respective effects of each RMT filtering scheme on the quantum lorenz ordering to get a feel for how these RMT filtering schemes may (or may not) help us improve our alarm system. Before getting started, we encourage the reader to revisit chapter 4, or consult Cirillo and Hüsler [2011], if they need to be refreshed on some of the above details.

7.1 | A Few Good Generalizations

Recall from chapter 4 how the stopping rule was defined: The process $\vec{X}$, which evolved according to the RoM (Rule of Motion) $d : S \times C \to S$, resets to the initial state $s = (0, 0)$ every time $X_{n^*} = (n^*, \mathcal{L})$ reached the critical risk level $\mathcal{L}$ at time instant n^*. As mentioned in Cirillo and

Hüsler [2011], this stopping rule may yield an alarm system that casts alarms in clusters, as opposed to in continuity, if the process $\vec{X}$ undergoes the consecutive sampling of the riskiest colored marbles (red in our case). To remedy this, we can generalize this stopping rule so as to only reset to the initial state $s = (0,0)$ if, in addition to $X_{n^*} = (n^*, \mathcal{L})$ reaching $\mathcal{L}$, the next subsequent q marbles are the riskiest (red) as well. In this way, an example set of 0-blocks may look like

$$\underbrace{0,0,1,1,2,3,\overbrace{3,\ldots,3}^{\text{q-times}}}_{\text{Block-1}} \,|\, \overbrace{0,1,2,3,\underbrace{3,\ldots,3}_{\text{q-times}}}^{\text{Block-2}} \,|\, \underbrace{0,0,0,3,\overbrace{3,\ldots,3}^{\text{q-times}}}_{\text{Block-3}} \tag{7.1}$$

stressing that, as in the model described in chapter 4, the sequence of q marbles would necessarily have to be either c_3 or c_4 coloured marbles. Though this introduces another hyper parameter into the RUP, this new stopping rule has powerful consequences. Indeed, the QMG has the tendency to form dense regions of red around known time periods of high volatility. Thus, we believe this simple generalization will allow us to build a more robust, and realistic, model. Mathematically, however, this generalization takes us away from the mixture of Markov chains and moves us into the class of semi-Markov chains, instead. We don't address this formalism but rather note that it would be interesting to consider in future work.

The second generalization involves the reinforcement scheme of the Urn. Up until now, we have been reinforcing each urn $U(s)$ with an $r(s)$ number of c_i coloured marbles whenever a c_i coloured marble is drawn out of the urn $U(s)$. This $m_s(c_i) \to m_s(c_i) + r(s)$ updating mechanism can be interpreted as a matrix operation via a "Reinforcement Matrix" (RM) with diagonal entries

$$RM = \begin{array}{c} \\ c_1 \\ c_2 \\ c_3 \end{array} \begin{array}{c} \begin{array}{ccc} c_1 & c_2 & c_3 \end{array} \\ \begin{pmatrix} r(s) & 0 & 0 \\ 0 & r(s) & 0 \\ 0 & 0 & r(s) \end{pmatrix} \end{array} \tag{7.2}$$

such that $m_s(c_i) \to RM\vec{e_i}$ where $\vec{e_i}$ is a standard unit vector with entry 1 in the i^{th} place and $0's$ elsewhere for c_i. It's easy to see how we can use the RM to define a more general reinforcement scheme.

Definition 7.1 (Balanced (Triangular) Reinforcement Matrix)**.** Given a set C of 3 different coloured

marbles $C = \{c_1, c_2, c_3\}$, we define the Balanced (Triangular) Reinforcement Matrix (RM) with

$$\begin{array}{c} \\ c_1 \\ c_2 \\ c_3 \end{array} \begin{array}{c} \begin{array}{ccc} c_1 & c_2 & c_3 \end{array} \\ \begin{pmatrix} \delta_{1,1} & \delta_{1,2} & \delta_{1,3} \\ 0 & \delta_{2,2} & \delta_{2,3} \\ 0 & 0 & \delta_{3,3} \end{pmatrix} \end{array}, \qquad s_i = \sum_{j=1}^{3} \delta_{i,j} = \delta \qquad \forall i \in \{1, 2, 3\} \tag{7.3}$$

where $\delta_{i,j}$ denotes the number of C_j coloured marbles to reinforce the Urn with when a C_i coloured marble is drawn.

The RM allows us to model dependence in the sampling of marbles. For instance, if we believe that risky states are typically preceded by moderate states, we can set $\delta_{2,3} = \delta - \delta_{2,2}$ so that every time a c_2 coloured marble is sampled from $U(s)$, the urn $U(s)$ is reinforced with $\delta_{2,2}$ and $\delta_{2,3}$ number of c_2 and c_3 coloured marbles. Considering this generalization in our model may allow the RUP to transition into, and out of, periods of extreme volatility more easily. The author(s) Cirillo and Husler [2009] posit that the RM allows the RUP to mimic the so-called "risky threshold mechanism" considered by likes of, for example Allan Gut [1999], in "Extreme Shock" models.

The last generalization we consider concerns the definition of the detection region. Recall from chapter 4 that the detection region took the form

$$\mathbb{P}[\tau_r = n + k | \tau_1, \ldots, \tau_{r-1}] \geqslant \gamma \in [0, 1] \tag{7.4}$$

for some predetermined threshold γ. That is, if the probability that the process will reset to $X_0 = (0, 0)$ exceeds the threshold γ, we cast an "alarm". However, what if we want to compute the detection region of an interval. For example, we may be interested in the probability that a crash will occur in the next 3, 6, or 12 months. In this case, we'd have to consider the detection region for the interval

$$\mathbb{P}[\tau_r \in [n, n + k) | \tau_1, \ldots, \tau_{r-1}] \geqslant \gamma \in [0, 1] \tag{7.5}$$

instead. Computing this interval is straightforward enough. Indeed, we show how to do this with the following theorem.

Theorem 7.2 (Detection Interval). *We calculate the Probability that the* r^{th} *Catastrophe will take*

place within the next $n + k$ *steps with*

$$\begin{aligned}\mathbb{P}[\tau_r \in (n, n+k]|\vec{\tau}_{r-1}] &= \sum_{i=1}^{k} \mathbb{P}[\tau_r = n + i|\vec{\tau}_{r-1}] \\ &= \sum_{i=1}^{k}\sum_{j=1}^{h} \mathbb{P}_j[(n, l), \ldots, (n+i, \mathcal{L})|\vec{X}_{n-1}]\end{aligned} \tag{7.6}$$

where $\mathbb{P}_j[(n, l), \ldots, (n+k, L)|\vec{X}_{n-1}]$ *denotes the Probability of the* j^{th}*-feasible path, out of* h*, from state* $s_n = (n, l) \rightarrow s_{n+i} = (n+i, \mathcal{L})$.

Proof. The proof immediately follows from the construction of the Process X and theorem 4.12 of Section 4.4.1. □

For instance, if we sample correlation matrices over 10-day sliding windows, choosing a $k = 6$ would roughly correspond to predicting whether a crash will occur in the next 3 months since trading weeks are typically 5 days long. For 6 months, we would need to choose $k = 12$ and for 12 months we'd need $k = 24$. However, a simple combinatorial argument shows that there are a total of $(|C| - 1)^k$ possible sampling sequences for $((n, l), \ldots, (n + k))$, as in equation 7.6, for $|C|$ the number of colours in the urn. Hence, and together with the generalized stopping rule we introduced earlier, the actual number of possible sampling sequences will be even larger. This makes computing the detection region for $k \geqslant \frac{\mathcal{L}}{2}$ computationally expensive. We can easily work around this by sampling correlation matrices over 20, or even 40, day sliding windows to compute the detection region(s) corresponding to 6 and 12 month intervals, instead for $k = 6$ in both cases.

In summary, the detection region, reinforcement scheme, and stopping rule of chapter 4 were all generalized in the following ways:

1. Detection region generalized to account for an interval: $\mathbb{P}[\tau_r \in (n, n+k]|\vec{\tau}_{r-1}]$
2. Reinforcement mechanism generalized to account for dependence:

$$RM = \begin{array}{c} \\ c_1 \\ c_2 \\ c_3 \end{array} \begin{array}{c} \begin{array}{ccc} c_1 & c_2 & c_3 \end{array} \\ \begin{pmatrix} \delta_{1,1} & \delta_{1,2} & \delta_{1,3} \\ 0 & \delta_{2,2} & \delta_{2,3} \\ 0 & 0 & \delta_{3,3} \end{pmatrix} \end{array}, s_i = \sum_{j=1}^{3} \delta_{i,j} = \delta, \forall i \in \{1, 2, 3\} \tag{7.7}$$

3. Stopping rule generalized to account for clustering such that X_n resets to the initial state $(0,0)$ only when $X_{n^*} = (n^*, \mathcal{L})$ and the next q marbles are all c_3.

First, the generalized detection region will allow us to build a more (financially) interpret-able alarm system. Second, the generalized reinforcement mechanism will allow our model to transition through periods of high volatility. Third, the generalized stopping rule will immunize our alarm system from the side effect of the clustering of alarms.

7.2 | A RUP based on Quantum Majorization

At this point, we are probably ready to specify the components of the RUP itself. First things first, so we reiterate that our RUP is defined using the set of 3 colours $C = \{c_1, c_2, c_3\}$ corresponding to green, yellow, and red coloured marbles, respectively. Needless to say, these colours correspond to "safe", "moderate", and "risky" time periods of 100 days provided that our temporal state space $P_\Delta(\mathcal{S}; I)$ is defined on $\Delta = (\cdot, 100)$. Second, we take advantage of the RM of definition 7.1 and establish a dependence between the sampling of yellow and red marbles according to

$$\text{RM} = \begin{array}{c} \\ c_1 \\ c_2 \\ c_3 \end{array} \begin{array}{c} \begin{array}{ccc} c_1 & c_2 & c_3 \end{array} \\ \begin{pmatrix} \delta & 0 & 0 \\ 0 & \delta_{2,2} & \delta_{2,3} \\ 0 & 0 & \delta \end{pmatrix} \end{array} \qquad \delta_{2,3} = \delta - \delta_{2,2} \tag{7.8}$$

where $\delta_{2,3} = \delta - \delta_{2,2}$ enforces the balanced nature of the reinforcement scheme. We go ahead and fully specify the RUP, and associated alarm system, borrowing from the framework of chapter 4. For instance, referring to definition 9.1, choosing a $\delta = 8$, and (say) $\delta_{2,3} = 5$ would mean that every time we sample a c_2 coloured marble, we reinforce the urn $U(s)$ with $\delta_{2,2} = 3$ and $\delta_{2,3} = 5$ c_2 and c_3 coloured marbles, respectively. Whereas in drawing either a c_1 or c_3 coloured marble we retain the $\delta = 8$ original reinforcement mechanism. As a brief refresher, we review the basic tenants of the RUP specification below. A RUP is chacterized by the following elements

- State space $S = \mathbb{N}_0^+ \times \{0, 1, \ldots, \mathcal{L}\}$ where the last risk level $\mathcal{L}$ corresponds to a catastrophe.
- Colours $C = \{c_1, \ldots, c_k\}$ of cardinality $|C| = k$.
- Rule of motion RoM $d : S \times C \to S$

- State space of urns $U(s)$, number $m_s(c)$ of coloured marbles c in $U(s)$ with reinforcement scheme according to the RM of equation 9.1.

where $\mathcal{Z}_0 = (0,0)$ is recursively updated according to the RoM $\mathcal{Z}_{n+1} = d((n,l),c_i)$ which resets to $\mathcal{Z}_0 = (0,0)$ the instant $\mathcal{Z}_{n^*} = (n^*, \mathcal{L})$ reaches $\mathcal{L}$ at critical time instant n^*. We say $\mathcal{Z} \in \text{RUP}(S, C, U, d)$. We have reiterated the RUP construction and detailed the handful of generalizations that make our model unique from previously utilized applications.

Before moving forward, it would be useful to formally define the evolution, and full characterization of, the process $\mathcal{Z}$ itself. Recall the Rule-of-Motion (RoM) we will be using in this construction:

$$d((n,l),c) = \begin{cases} (n+1, l-1), & \text{if } c = c_1 \\ (n+1, l), & \text{if } c = c_2 \\ (n+1, l+1), & \text{if } c = c_3 \end{cases} \tag{7.9}$$

Definition 7.3. Given a set of colours $C = \{c_1, c_2, c_3\}$, we define an underlying process $\vec{Y} = \{Y_1, \ldots, Y_N\}$ taking elements $Y_n \in C$. The process $\mathcal{Z} \in \text{RUP}(S, C, U, d)$ is updated via a RoM $\mathcal{Z}_{n+1} = d((n,l),c_i)$ for $Y_n = c_i$ where $U((n,l))$ is reinforced according to the RM

$$\text{RM} = \begin{array}{c} \\ C_1 \\ C_2 \\ C_3 \end{array} \begin{array}{c} \begin{array}{ccc} C_1 & C_2 & C_3 \end{array} \\ \begin{pmatrix} \delta & 0 & 0 \\ 0 & \delta_{2,2} & \delta_{2,3} \\ 0 & 0 & \delta \end{pmatrix} \end{array} \qquad \delta_{2,3} = \delta - \delta_{2,2} \tag{7.10}$$

characterized by the $(\delta_{2,3}, \delta)$ to model dependence. The process $\mathcal{Z}_n$ resets to the initial state $\mathcal{Z}_0 = (0,0)$ only when the stopping rule

$$\mathcal{Z}_{n^*} = (n^*, \mathcal{L}) \;\&\; Y_{n^*+1} = \cdots = Y_{n^*+q} = c_3 \tag{7.11}$$

is satisfied, implicitly depending on the $(q, \mathcal{L})$ tuple.

As mentioned in the definition above, the process $\mathcal{Z} \in \text{RUP}(S, C, U, d)$ is now implicitly dependent on the two tuples $(\delta_{2,3}, \delta)$ and $(q, \mathcal{L})$ Now we put all the pieces together and formalize the RUP for the prediction of stock market crashes. We build the temporal state space $P_\Delta(\mathcal{S}; I)$ of

definition 3.6 using 40 of the most prominent historical DJIA constituents

$$\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_{40}\} \tag{7.12}$$

so as to avoid the so-called "survivorship bias" discussed earlier in chapter 6. In $\mathrm{P}_{\Delta}(\mathcal{S}; \mathrm{I})$, our time interval of interest corresponds to the time period ranging from May 6, 1986 up to Dec 31, 2020. We sample correlation matrices over sliding windows corresponding to a $\Delta = (40, 100)$. For instance, the first correlation matrix $\rho_0 = [0, m] \in \mathrm{P}_{\Delta}(\mathcal{S}; \mathrm{I})$ corresponds to the correlation matrix sampled using the log returns of the constituents $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_{40}\}$ for the m days following January 7. Hence, letting the model begin on May 6, 1986 ensures that the RUP will be using purely historical information to make predictions about the future.

The last piece is generating the underlying process $\vec{Y} = \{Y_1, \ldots, Y_n\}$ to the $\mathcal{Z} \in \mathrm{RUP}(S, C, U, d)$. Each element Y_n corresponds to the clustering of the n^{th} node in the QMG $G = (V, A)$ of size $|V| = n$ according to the spectral clustering algorithm. The QMG is built in an online fashion as more and more correlation matrices $\rho_n \in \mathrm{P}_{\Delta}(\mathcal{S}; \mathrm{I})$ are sampled. This is perhaps the most important component of our model since the QMG fully specifies the nature of the composition of the state space of urns $U(s)$ that our RUP is defined on. For clarity, we outline the algorithm for building the QLO, the corresponding adjacency matrix A, and the clustering of the node set generating the sequence $\vec{Y} \in C^N$ of drawn marbles for the directed acyclic QMG $G = (V, A)$ of size $|V| = N$.

Definition 7.4 (QMG clustering algorithm)**.** We start with a set of colours $C = \{c_1, c_2, c_3\}$.

1. Suppose we are in some time $t = hn + m$.
2. For every $0 \leqslant n \leqslant N = |\mathrm{P}_{\Delta}(\mathcal{S}; \mathrm{I})|$, we sample $\rho_n = \rho_{\iota_n}$ where $\iota_n = [hn, hn + m]$.
3. For every $0 \leqslant m \leqslant n$ we check whether $\rho_m \succ \rho_n$ or $\rho_n \succ \rho_m$ and set $A_{m,n} = \{0, 1\}$ and $A_{n,m} = \{0, 1\}$ accordingly.
4. Form the QMG $G = (V, A)$ with node set $v_n \in V$ corresponding to the set of correlation matrices $\rho_n \in \mathrm{P}_{\Delta}(\mathcal{S}; \mathrm{I})$.
5. Assign the n^{th} node of $G = (V, A)$ to a cluster according to the risk measure $\theta(\rho_n) \in [0, 1]$ with associated discriminating thresholds $0 \leqslant d_1 \leqslant d_2 \leqslant 1$ by setting $Y_n = c_i$.

6. Update the RUP according to $\mathcal{Z}_{n+1} = d((n, l), c_i)$.

7. If $\mathbb{P}[\tau_r \in [n, n+k] | \tau_1, \ldots, \tau_{r-1}] \geqslant \gamma$ we cast an alarm. Now repeat.

This algorithm generates a sequence $\vec{Y} \in \{c_1, c_2, c_3\}^N$ that recursively updates $\mathcal{Z} \in \mathrm{RUP}(S, C, U, d)$ according to $\mathcal{Z}_{n+1} = d((n, l), c_i)$

The detection region is defined using the threshold γ. All in all, this means that our RUP is defined using a hyper parameter set consisting of 5 values: γ, $(q, \mathcal{L})$, and $(\delta_{2,3}, \delta)$.

Lastly, since each correlation matrix ρ_n is sampled over the time interval $\iota_n = [t-m, t]$, when we are in time $t = hn + m$, the model is built using only historical information. Furthermore, since the next correlation matrix ρ_{n+1} isn't sampled for another h trading days after that of ρ_n, the cluster assignment of $\rho_n \leftarrow c_i$ characterizes the risk level the QMG is suggesting the market will endure for the next (future) h days until the next ρ_{n+1} is sampled, h days later.

7.3 | Predicting Stock Market Crashes with a RUP

We build our RUP for the prediction of stock market crashes starting from the temporal state space: a collection of correlation matrices sampled over sliding windows. These sliding windows are characterized by the $\Delta = (h, m)$ tuple such that each correlation matrix in the temporal state space $\rho_n = \rho_{\iota_n} \in P_\Delta(\mathcal{S}; I)$ is associated to a time period $\iota_n = [hn, hn+m] \in I$ for which the log returns for each asset $\mathcal{S}_i \in \mathcal{S}$ are sampled over. By temporally sampling these correlation matrices $\rho_n \in P_\Delta(\mathcal{S}; I)$, as per the 7 step algorithm provided in the previous section 7.2, we generate a process $\vec{Y} = \{Y_1, \ldots, Y_N\}$, for $N = |P_\Delta(\mathcal{S}; I)|$ the number of correlation matrices in $P_\Delta(\mathcal{S}; I)$, taking elements $Y_n \in \{c_1, c_2, c_3\}$ in the set of colours $C = \{c_1, c_2, c_3\}$.

Furthermore, thanks to the three generalizations made in section 7.1, the process $\mathcal{Z} \in \mathrm{RUP}(S, C, U, d)$ is implicitly characterized by the two tuples $(q, \mathcal{L})$ and $(\delta_{2,3}, \delta)$ defining the RUP's stopping rule and reinforcement mechanism, respectively, and the detection region threshold γ. We denote with $\Theta = (\gamma; (q, \mathcal{L}), (\delta_{2,3}, \delta))$ this parameter set. We neglect the parameter k in the parameter set Θ since k is being fixed to give the model financial interpret-ability. Now, given a RUP $\mathcal{Z} \in \mathrm{RUP}(S, C, U, d)$, and detection interval $\mathbb{P}[\tau_r \in [n, n+k] | \tau_1, \ldots, \tau_{r-1}]$, we can now formally define the alarm system.

Definition 7.5 (Alarm System)**.** Given a process $\mathcal{Z} \in \mathrm{RUP}(S, C, U, d)$, along with the parameter set $\Theta = (\gamma; (q, \mathcal{L}), (\delta_{2,3}, \delta))$, we denote with $\mathcal{A}_\Theta(\mathcal{Z}) = \mathcal{A}(\mathcal{Z}; \Theta)$ an alarm system whose elements

$\mathcal{A}_n \in \mathcal{A}_\Theta(\mathcal{Z})$ take the form

$$\mathcal{A}_n = \mathbb{1}[p(\tau_r) \geqslant \gamma],\ \tau_r = \inf\{n \geqslant \tau_{r-1} : \mathcal{Z}_n = (n, \mathcal{L}), \{Y_{n+1+i}\}_{i=0}^{q} = c_3\} \tag{7.13}$$

where $p(\tau_r) = \mathbb{P}[\tau_r \in [n, n+k) | \tau_1, \ldots, \tau_{r-1}]$ is the detection interval.
As alluded to earlier, our choice of k is crucial to the efficacy of $\mathcal{A}_\Theta(\mathcal{Z})$. Indeed, k determines which time horizon our alarm system $\mathcal{A}_\Theta(\mathcal{Z})$ is making predictions over. Furthermore, the choice of k is made in correspondence with the sliding windows we use to sample correlation matrices. Since we want $\mathcal{A}_\Theta(\mathcal{Z})$ to be able to make predictions over long and short time horizons, we find that we need to sample correlation matrices such that our longest time horizon corresponds to a k no larger than 8 in order to speed up calculations when individuating sampling paths out of the urns $U(s)$ for the state space $s \in S$. In which case, this would mean we could sample correlation matrices over sliding windows of $\Delta = (40, 100)$. Indeed, by choosing $\Delta = (40, 100)$ we could make predictions on 4, 8, and 12 month time horizons corresponding to $k = 2, 4, 6$, respectively since a 40 day sliding window corresponds to 2 months of trading.

For instance, if we have a temporal state space of correlation matrices $P_\Delta(\mathcal{S}; I)$, with $\Delta = (40, 100)$, the clustering $Y_0 = c_i$ for correlation matrix $\rho_0 \in P_\Delta(\mathcal{S}; I)$ would determine the level of risk the market will be in for the next 40 days (2 months) until ρ_1 is sampled 40 days later. Hence, when $\mathcal{Z}_0 = (0, 0)$ we can compute $\mathbb{P}[\tau_1 \in [0, k = 2)]$ the probability of there being a crash in 4 months, so on and so forth, for $k = 4, 6$. In this way, the detection region $\mathbb{P}[\tau_1 \in [n, k = 2)$ continually gets updated alongside the evolution of the process $\mathcal{Z}_n = d((n-1, l), c_i)$. For this reason, we stick with the choice of sampling correlation matrices over 40 day sliding windows of length 100 and fix $k = \{2, 4, 6\}$ so as to make predictions over 4, 8, and 12 month time horizons. Furthermore, we refer to figure 7.1 to revisit the crashes we want our alarm system $\mathcal{A}_\Theta(\mathcal{Z})$ to predict, as identified by the EVT analysis of chapter 6.

Thus, the task of building an optimal alarm system for the prediction of stock market crashes is simple enough. We use the temporal state space to train a RUP $\mathcal{Z} \in \mathsf{RUP}(S, C, U, d)$ to cast alarms according to the detection region and find an optimal parameter set $\Theta^* = (\gamma; (q, \mathcal{L}), (\delta_{2,3}, \delta))$ such that

$$\Theta^* = \underset{\Theta}{\operatorname{argmax}}\, u[\mathcal{A}_\Theta(\mathcal{Z})] \tag{7.14}$$

for some optimality criterion $u[\mathcal{A}_\Theta(\mathcal{Z})] \in \mathbb{R}$. As with any predictive model we need to separate our model $\mathcal{Z} \in \mathsf{RUP}(S, C, U, d)$ into training and testing sets $\mathcal{Z}_{\text{train}}$ and $\mathcal{Z}_{\text{test}}$, respectively. Concretely, we train our process $\mathcal{Z} \in \mathsf{RUP}(S, C, U, d)$ so as to find an optimal parameter set Θ^*_{train}

such that

$$\Theta^*_{\text{train}} = \underset{\Theta}{\text{argmax}}\, \mathfrak{u}[\mathcal{A}_\Theta(\mathcal{Z}_{\text{train}})] \tag{7.15}$$

and then evaluate our alarm systems $\mathcal{A}_{\Theta^*_{\text{train}}}(\mathcal{Z})$ performance using the optimality criterion

$$\mathfrak{u}[\mathcal{A}_{\Theta^*_{\text{train}}}(\mathcal{Z}_{\text{test}})] \tag{7.16}$$

for some evaluation metric $\mathfrak{u}[\cdot]$. All that is left is to specify the RUP's training and testing periods, which we do next in subsection 7.3.1.

7.3.1 | Testing & Training

We reiterate that our model samples correlation matrices over 40 day sliding windows of length 100 days. That is, every 40 (trading) days, we sample a correlation matrix using the log returns of the DJIA historical constituents spanning the previous 100 days. This defines a temporal state space $\mathrm{P}_\Delta(\mathcal{S};\mathrm{I})$ with $\Delta = (40, 100)$, $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_40\}$ DJIA historical constituents, over the time interval I spanning May 6, 1986 until January 1, 2020, of cardinality $|\mathrm{P}_\Delta(\mathcal{S};\mathrm{I})| = 218$ number of correlation matrices $\{\rho_n\}_{n=1}^{218} = \mathrm{P}_\Delta(\mathcal{S};\mathrm{I})$. These correlation matrices will be used to generate a process $\vec{Y} = \{Y_1, \ldots, Y_{218}\} \in \{c_1, c_2, c_3\}^{218}$, via the algorithm of definition 7.4 using the spectral clustering of the Quantum Majorization Graph (QMG), which recursively updates the $\mathcal{Z} \in \mathrm{RUP}(S, C, U, d)$ according to $\mathcal{Z}_{n+1} = d((n, l), c_i)$ and the general RUP specification. At each iteration n, the detection region $p(\tau_r) = \mathbb{P}[\tau_r \in [n, n+k)|\tau_1, \ldots, \tau_{r-1}]$ for the r^{th} stopping time $\tau_r = \inf\{n \geqslant \tau_{r-1} : \mathcal{Z}_n = (n, \mathcal{L}), \{Y_{n+1+i}\}_{i=0}^{q} = c_3\}$ is computed $\forall k = \{2, 4, 6\}$ corresponding to 4, 8, and 12 month time horizons. An alarm $\mathcal{A}_n = 1$ is cast if $p(\tau_r) \geqslant \gamma \in [0, 1]$ for a predetermined threshold γ, where $\mathcal{A}_n = 0$ otherwise. This generates the alarm system $\mathcal{A}_\Theta(\mathcal{Z}) = \{\mathcal{A}_1, \ldots, \mathcal{A}_{218}\}$ for which we use a function $\mathfrak{u}[\mathcal{A}_\Theta(\mathcal{Z})] \in \mathbb{R}$ to evaluate its performance. All that is left to do is separate the temporal state space into training and testing sets $\mathrm{P}_\Delta(\mathcal{S};\mathrm{I}_{\text{train}})$ and $\mathrm{P}_\Delta(\mathcal{S};\mathrm{I}_{\text{test}})$, respectively. We use the correlation matrices $\rho_n \in \mathrm{P}_\Delta(\mathcal{S};\mathrm{I}_{\text{train}})$ for bayesian learning and find an optimal paramater set Θ^* for the alarm system $\mathcal{A}_{\Theta^*}(\mathcal{Z}_{\text{train}})$. We use this set Θ^* to evaluate $\mathcal{A}_{\Theta^*}(\mathcal{Z}_{\text{test}})$. We now define the training and testing time intervals $\mathrm{I}_{\text{train}}$ and I_{test}, respectively.

To separate the training and testing time intervals $\mathrm{I}_{\text{train}}$ and I_{test}, we refer back to figure 7.1 to look at the 13 crashes $\mathcal{C} = \{\mathcal{C}_1, \ldots, \mathcal{C}_{13}\}$ as identified by the EVT analysis of chapter 6. We want the training and testing sets to be as balanced as possible. Considering the "Black Monday" crash

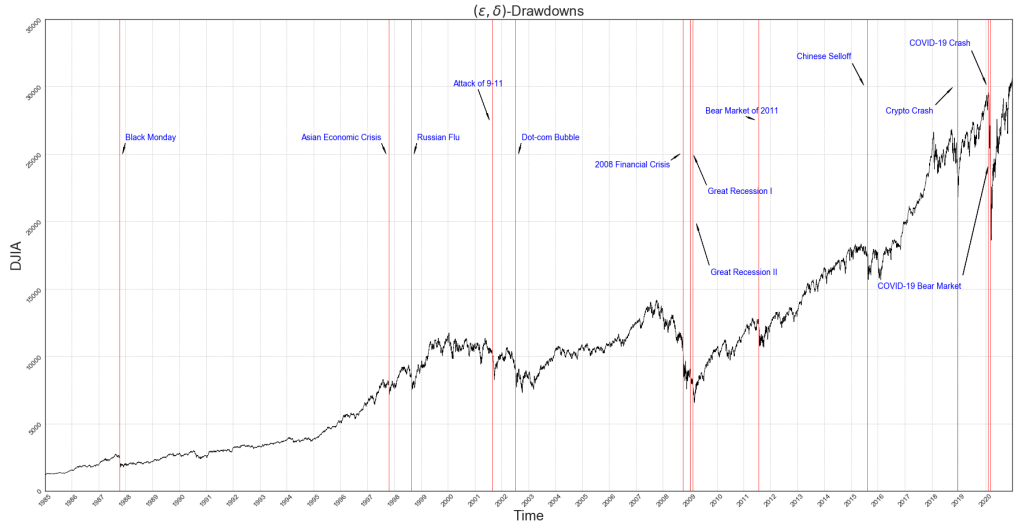


Figure 7.1: Stock Market Crashes: DOW Jones Industrial Average (2000-2020)

of 1987, our choice of sliding windows $\Delta = (40, 100)$ generates a QMG $G = (V, A)$ of size just $|V| = 9$ between our start date May 6, 1986 and October 1987. Hence, our model simply hasn't had enough time to say something significant about this "Black Monday" crash. This means we essentially only have 12 crashes to predict, 8 of which occur after 2005. Thus, we will train our model from May 6, 1986 until January 1, 2005 and thus test our model from January 1, 2005 until December 31, 2020. At first glance, the relative size of the time intervals I_{train} and I_{test} seems to be imbalanced. However, we believe this splits the training and testing crashes optimally. Indeed, if we shorten the training period, the number of training and testing crashes would be even more imbalanced. If we train even longer, our model won't have the chance to predict one of the most infamous global economic recessions in history, the 2008 "Great Recession". Hence, we have the following training and testing time intervals I_{train} and I_{test}.

Train $\rightarrow$ May 6, 1986 until January 1, 2005

Test: $\rightarrow$ January 1, 2005 until December 31, 2020

We visualize this with figure 7.2. Hence, we run the algorithm as per definition 7.4 from May 6, 1986 until January 1, 2005. This generates a process $\vec{Y} = \{Y_1, \ldots, Y_{n_{train}}\}$ each element taking values $Y_n \in \{c_1, c_2, c_3\}$. Needless to say, the temporal state space $P_\Delta(\mathcal{S}; I)$ is also split up into training and testing periods $P_\Delta(\mathcal{S}; I_{train})$ and $P_\Delta(\mathcal{S}; I_{test})$, respectively. Here, I_{train} corresponds to the time

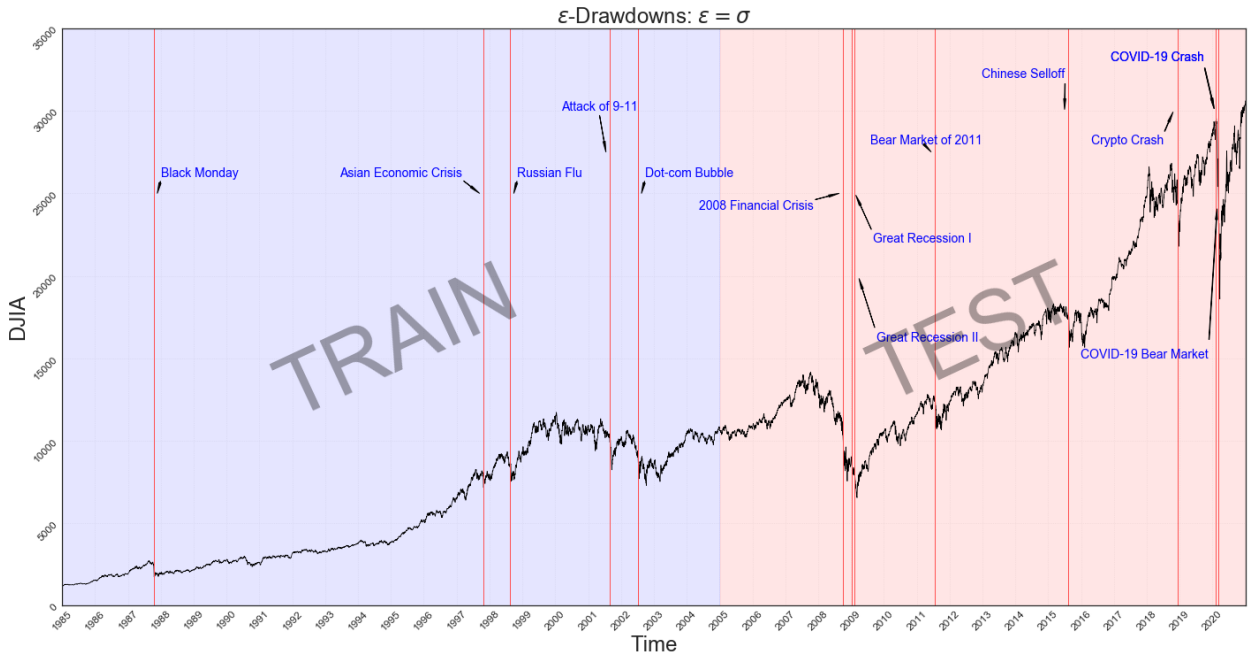


Figure 7.2: Training and Testing sets

period May 6, 1986 to January 1, 2005 of which each correlation matrix $\rho_n = \rho_{\iota_n} \in \mathrm{P}_\Delta(\mathcal{S}; \mathrm{I}_{\text{train}})$ is sampled over the time interval $\iota_n = [40n, 40n + 100]$ corresponding to sliding windows of length 100 over 40 day (2 month) shifts. Likewise for the testing period I_{test}, we have a temporal state space $\mathrm{P}_\Delta(\mathcal{S}; \mathrm{I}_{\text{test}})$ with I_{test} corresponding to the time interval January 1, 2005 until December 31, 2020. All in all, the temporal state spaces $\mathrm{P}_\Delta(\mathcal{S}; \mathrm{I}_{\text{train}})$ and $\mathrm{P}_\Delta(\mathcal{S}; \mathrm{I}_{\text{test}})$ contain 116 and 102 correlation matrices, respectively.

In figure 7.3 we display an example process $\vec{Y} = \{Y_1, \ldots, Y_{n_{\text{train}}}\}$ for the training period. In this plot, notice that each individual (vertical) band represents a single $Y_n \in \{c_1, c_2, c_3\}$ corresponding to a correlation matrix $\rho_n = \rho_{\iota_n} \in \mathrm{P}_\Delta(\mathcal{S}; \mathrm{I}_{\text{train}})$ for the log returns of DJIA constituents sampled over the time interval $\iota_n = [40n, 40n + 100]$. Thus, each vertical band in the figure has a width of 40 days. Using figures 7.1 and 7.2 as references, the reader can pinpoint where each crash in the training set falls in this plot. Indeed, each crash in the training set is either surrounded by red, preceded by yellow, or even both as in the case of the 1997 "Asian Economic Crisis", and the early 2000's "Attack of 9-11" and "Dot-com Bubble". The crash corresponding to the 1998 "Russian Flu" is mildly preceded by yellow whereas our intuition for the 1987 "Black Monday" crash is perhaps verified since the clustering oscillates between green and red prior to, and after, the crash.

Figure 7.3: Clustering $\vec{Y} \in C^N$: Training period using $\Lambda_{\Delta}^{U}(\mathcal{S}; I_{\text{train}})$

7.3.2 | Final Notes

Up until now we have glanced over a few details concerning the parametrization of the $\mathcal{Z} \in \text{RUP}(S, C, U, d)$. For instance, we haven't discussed the initial urn composition $U_0(s)$, its relationship with the reinforcement mechanism, or the interpretation of the $(q, \mathcal{L})$ tuple and the anti-clustering parameter q itself. In addition, we make a note of what of what happens when we choose shorter sliding windows of size (say) 20, or even 10, days.

We begin by briefly discussing the $(q, \mathcal{L})$ tuple. We refer to q as the "anti-clustering" parameter. As we will see later in chapter 8, stock market crashes tend to be buried in dense regions of c_3 marbles. Hence, during these time periods, the process $\mathcal{Z} \in \text{RUP}(S, C, U, d)$ will be sampling several $\{Y_n\} = c_3$ consecutively, and possibly, in addition to reaching $\mathcal{L}$. Without considering this parameter q, $\mathcal{Z}$ will keep resetting to $(0, 0)$ which will keep reinforcing the urn $U((0, 0))$ with more and more c_3 marbles. In turn, this will teach the urn that $(0, 0)$ is not safe which will teach $\mathcal{Z}$ to signal more and more false alarms in "clusters". We'd rather want $U((0, 0))$ to be reinforced with c_1 (or c_2) marbles, instead. Because of this, we'd expect the alarm system to take advantage of the parameter q for larger time horizons (larger k). More on this later.

Furthermore, in an effort to aid our grid search for finding an optimal Θ^*, we state corollary 7.6 whose proof is immediate.

Corollary 7.6. *Consider the* $(q, \mathcal{L})$ *tuple. If* $q + \mathcal{L} \geqslant k$*, then*

$$\mathbb{P}[\tau_r \in [n, n+k) | \tau_1, \ldots, \tau_{r-1}] = 0 \tag{7.17}$$

for the detection interval of the r^{th} *stopping time* τ_r*.*

The proof follows follows from the construction of the stopping rule. For our purposes, we can leverage this corollary to simplify our grid search for finding an optimal Θ^*. For example, suppose we have $k = 6$. We know we don't have to consider any $q \geqslant k$ since $\mathcal{L} > 0$ necessarily. Also, we found that any $\mathcal{L} \geqslant 6$ to be sub-optimal in any case. Hence, our grid search need only consider $(q, \mathcal{L})$ on the $[0, k] \times [0, 6]$ interval. This is the convention we adopt in our grid search.

Second, let us discuss the initial urn composition, which we denote with $U_0(s)$. We consider an (almost) uniformly distributed initial urn containing 35 c_1, 35 c_2, and 30 c_3 coloured marbles. Thus, every urn $U(s)$ will be Dirichlet distributed $U(s) \sim Dir(35, 35, 30)$. We choose this distribution in an attempt to give our model more (financial) interpret-ability. A crucial feature of our alarm system would be for the alarm system to be more and more accurate over longer and longer time horizons. Indeed, an oracle could predict there will be a crash in the next 100 years, in which case, they'd surely be right every time. A true test would be if the oracle can predict when a crash is going to happen 2 months in advance. Initializing the urn with fewer c_3 marbles (significantly more biased towards c_1 and c_2 marbles) would make terminal sampling paths over longer time horizons less likely by the law of compounding small probabilities and thus (potentially) compromising this crucial feature of the alarm system. This is especially true if the process $\mathcal{Z} \in RUP(S, C, U, d)$ finds itself in unfamiliar territory where it's having to sample out of the initial urns frequently. Furthermore, as is the case in figure 7.3, we don't see a strong bias towards any particular c_i anyways.

Third, and somewhat related to the above discussion on the initial urn composition, we need to consider the reinforcement mechanism as per the RM of 9.1. Since our initial urn contains 100 marbles, our reinforcement mechanism, and associated dependence structure, should reflect the size of $U_0(s)$. The author(s) Cirillo and Hüsler [2011] also considered an initial urn containing 100 marbles where they argued a reinforcement parameter greater than 10 to be unrealistic. We follow the same convention and allow δ to range over the $[1, 10]$ interval. Furthermore, the dependence parameter $\delta_{2,3}$ can be no larger than δ itself. Thus, the grid search for $(\delta_{2,3}, \delta)$ will range over the $(\delta_{2,3}, \delta) \in [0, 10] \times [1, 10]$ interval.

Lastly, choosing shorter sliding windows like (say) 10, does not change the visualization of

the clustering. Regions of red, green, and yellow correspond to certain periods of time regardless of how frequently the correlation matrices are sampled. We found that, since a temporal state space $P_{\Delta}(\mathcal{S}; I)$ sample over 40 day sliding windows is just a subset of the temporal state space corresponding to 10 day sliding windows, the intermittent correlation matrices will belong to similar communities on the QMG. However, choosing sliding windows of size 10 instead, as the author(s) Fontanari et al. [2019] did, would require us to consider $k = 6, 12, 24, 36$ to make predictions for 2, 4, 8, and 12 month time horizons. Thus, our choice of 40 is simply an effort to speed up calculations.

7.4 | RMT Filtering Interlude

In this section we revisit the RMT filtering schemes introduced in chapter 2. In particular, we want to see if they can be used to improve the results of our RUP alarm system $\mathcal{A}_\Theta(\mathcal{Z})$. Indeed, RMT filtering schemes were developed to filter out measurement error: the 1st variety of statistical variability as per Gini [1912]. Hence, if the correlation matrices used in the development of our RUP alarm system are inflicted by measurement error, maybe these RMT filtering schemes can improve our results. First we want to see how RMT filtering affects the dispersion of a correlation matrices spectrum. In particular, we want to see if $\rho \succ \Xi^*[\rho]$ relations can be built for an RMT filtering scheme $\Xi^*[\rho]$.

We provide analytical relationships for quantum majorization relations between cleaned correlation matrices for the basic linear shrinkage $\Xi^{BLS}[\rho]$ and clipped $\Xi^{CLP}[\rho]$ RMT filtering schemes. In addition we provide a heuristic intuition for how an equivalent relation can be drawn for the rotationally invariant estimator $\Xi^{RIE}[\rho]$, but omit a full investigation. As such, we pay particular attention to basic linear shrinkage and eigenvalue clipping. We begin with basic linear shrinkage $\Xi^{BLS}[\rho]$.

Lemma 7.7. *We have* $\rho \succ \Xi^{BLS}[\rho]$, $\forall \rho \in P(S)$.

Proof. Let us consider the Quantum Lorenz Curve $L_\alpha(\Xi^{BLS}[\rho])$, that is

$$\begin{aligned} L_\alpha(\Xi^{BLS}[\rho]) &= (1-\beta)\lambda_1^{\downarrow} + \beta + \cdots + (1-\beta)\lambda_\alpha^{\downarrow} + \beta \\ &= (1-\beta)(\lambda_1^{\downarrow} + \cdots + \lambda_\alpha^{\downarrow}) + \alpha\beta \\ &= L_\alpha(\rho) + \beta(\alpha - L_\alpha(\rho)) \\ &\leqslant L_\alpha(\rho), \qquad \forall \beta \in [0,1] \end{aligned} \tag{7.18}$$

since $\alpha - L_\alpha(\rho) \leqslant 0$, by construction of $L_\alpha(\rho)$. Thus, $\rho \succ \Xi^{BLS}[\rho]$, completing the proof. □

The astute reader may have recognized that the above proposition can be read as a direct consequence of axiom 3.4. Indeed, the basic linear shrinkage scheme is simply a convex combination of the correlation matrix ρ, and the least risky element in the ordering, the identity matrix I_d. Thus, the fact that $\Xi^{BLS}[\rho]$ can be read as being inherently less risky than ρ itself makes sense. Let's see if we can verify something similar for eigenvalue clipping $\Xi^{CLP}[\rho]$ next.

Lemma 7.8. *We have* $\rho \succ \Xi^{\mathrm{CLP}}[\rho], \forall \rho \in \mathrm{P}(S)$.

Proof. By construction of $\Xi^{\mathrm{CLP}}[\rho]$ we know $\mathrm{L}_{\lceil d\beta \rceil}(\Xi^{\mathrm{CLP}}[\rho]) = \mathrm{L}_{\lceil d\beta \rceil}(\rho)$ and that the remaining $d - \lceil d\beta \rceil$ eigenvalues of $\Xi^{\mathrm{CLP}}[\rho]$ are equal to a trace preserving constant γ equal to

$$\gamma = \frac{1}{d - \lceil d\beta \rceil}(1 - \mathrm{L}_{\lceil d\beta \rceil}(\rho)) \tag{7.19}$$

which means that $\mathrm{L}_{\alpha \geqslant \lceil d\beta \rceil}(\Xi^{\mathrm{CLP}}[\rho]) \leqslant \mathrm{L}_{\alpha}(\rho)$, by construction of $\mathrm{L}_{\alpha}(\rho)$, completing the proof. □

So far, for the eigenvalue clipping and basic linear shrinkage RMT filtering schemes $\Xi^{\mathrm{CLP}}[\rho]$ and $\Xi^{\mathrm{BLS}}[\rho]$, both are majorized by the original correlation matrix ρ. As for the third RMT filtering scheme, the rotationally invariant estimator RIE, establishing such a relationship is a little bit more difficult, for our purposes.

The same relationship for $\rho \succ \Xi^{\mathrm{RIE}}[\rho]$, as is the case for the two schemes above, should hold in the large N limit. However, for finite N, which is of the dimension we are working with in this work, the RIE scheme is not necessarily trace preserving. Hence, a we believe subtle normalization argument is needed to establish the above relation for the RIE scheme. We omit this exploration in this work and leave it for future work. As a teaser, Bun et al. [2017] show that $\mathrm{Tr}[\Xi^{\mathrm{CLP}}[\rho^2]] \leqslant \mathrm{Tr}[\rho^2]$ which essentially means that the spectrum of $\Xi^{\mathrm{CLP}}[\rho]$ is narrower than the spectrum of ρ. However, we provide, what we believe to be, evidence for this in figure 7.4.

We can visualize Lemmas 7.7 and 7.8 by analyzing Figure 7.4. Indeed, we see, for the Correlation Matrix depicted in Figure 3.1(a), the majorization relations $\rho \succ \Xi^{\mathrm{BLS}}[\rho]$, $\rho \succ \Xi^{\mathrm{CLP}}[\rho]$, and $\rho \succ \Xi^{\mathrm{RIE}}[\rho]$ (after normalization). However, notice we cannot infer a QLO between the cleaning recipes $\Xi[\rho]$ themselves. However, for this specific choice of β, it appears we have $\Xi^{\mathrm{BLS}}[\rho] \succ \Xi^{\mathrm{RIE}}[\rho]$, but this clearly does not hold in general and depends on β. Furthermore, it is very difficult to majorize $\Xi^{\mathrm{CLP}}[\rho]$, especially for smaller and smaller choices of $\lceil d\beta \rceil$, since such a $\Xi^{\mathrm{CLP}}[\rho]$ quickly approaches the Quantum Lorenz Curve for the uniform spectrum $\mathrm{L}_{\alpha}(\lambda(\mathrm{I}_d))$.

7.4.1 | Basic Linear Shrinkage

We just saw how basic linear shrinkage generates a new spectrum $\Xi^{\mathrm{BLS}}[\rho]$ that is majorized by the original correlation matrix ρ. However, it would be more interesting to see if we can establish a stronger relationship on the QLO iteslf. For instance what if we have two different correlation

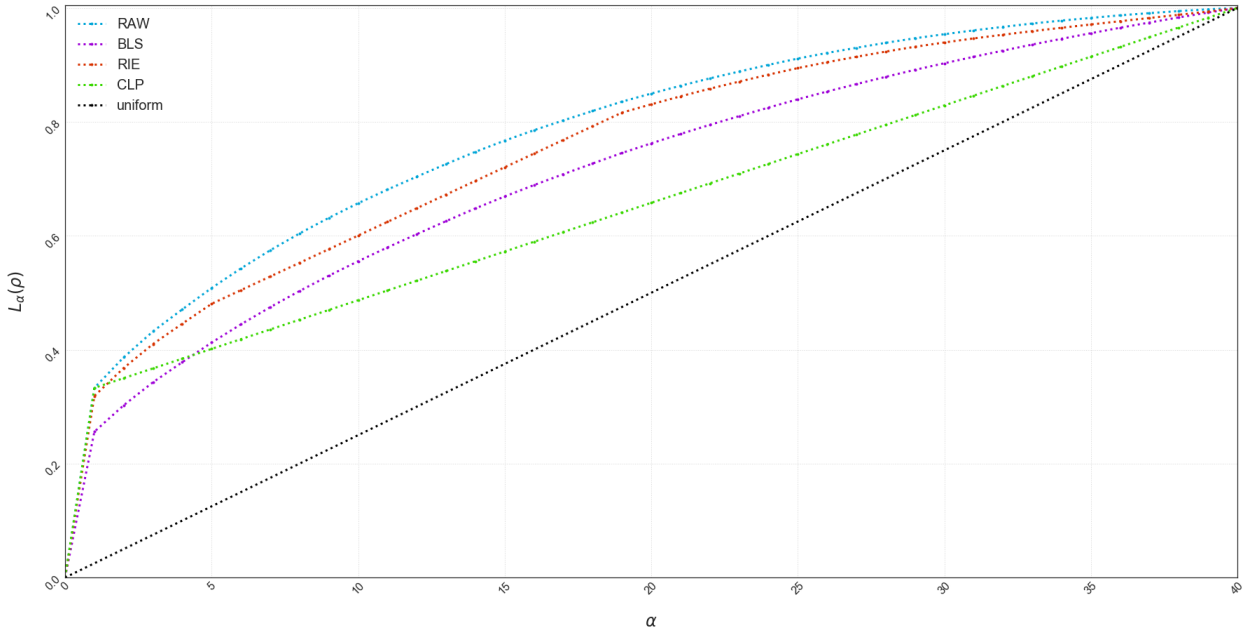


Figure 7.4: Lorenz Curve of RMT Cleaning recipes

Plotting $L_\alpha(\Xi[\rho])$ against α for the RMT filtering schemes RIE, BLS, and CLP.

matrices ρ_l and ρ_k such that $\rho_l \succ \rho_k$. What can we say about $\Xi^{\text{BLS}}[\rho_l]$ and $\Xi^{\text{BLS}}[\rho_k]$. Indeed, understanding this relationship will allow us to understand how basic linear shrinkage effects the QMG and thus, in turn, the RUP alarm system $\mathcal{A}_\Theta(\mathcal{Z})$ itself. We address this with the following proposition.

Proposition 7.9. *Let* $\rho_k, \rho_l \in P(S)$ *and fix* $\beta \in \mathbb{R}$. *We have that* $\rho_l \succ \rho_k$ *if, and only if,* $\Xi^{BLS}[\rho_l] \succ \Xi^{BLS}[\rho_k]$.

Proof. ($\Rightarrow$) First, suppose $\rho_l \succ \rho_k$, that is $L_\alpha(\rho_l) \geqslant L_\alpha(\rho_k)$ for all α, or, in other words,

$$\lambda_1^{\downarrow}(\rho_l) + \cdots + \lambda_\alpha^{\downarrow}(\rho_l) \geqslant \lambda_1^{\downarrow}(\rho_k) + \cdots + \lambda_\alpha^{\downarrow}(\rho_k), \qquad \forall \alpha \in \{1, \ldots, d\}$$

then, after applying the $\Xi^{\text{BLS}}[\cdot]$ mapping, we just need to verify that

$$\begin{aligned}
(1-\beta)\lambda_1^{\downarrow}(\rho_l) + \cdots + (1-\beta)\lambda_\alpha^{\downarrow}(\rho_l) + \alpha\beta &\geqslant (1-\beta)\lambda_1^{\downarrow}(\rho_k) + \cdots + (1-\beta)\lambda_\alpha^{\downarrow}(\rho_k) + \alpha\beta \\
(1-\beta)\lambda_1^{\downarrow}(\rho_l) + \cdots + (1-\beta)\lambda_\alpha^{\downarrow}(\rho_l) &\geqslant (1-\beta)\lambda_1^{\downarrow}(\rho_k) + \cdots + (1-\beta)\lambda_\alpha^{\downarrow}(\rho_k) \\
(1-\beta)(\lambda_1^{\downarrow}(\rho_l) + \cdots + \lambda_\alpha^{\downarrow}(\rho_l)) &\geqslant (1-\beta)(\lambda_1^{\downarrow}(\rho_k) + \cdots + \lambda_\alpha^{\downarrow}(\rho_k)) \\
\lambda_1^{\downarrow}(\rho_l) + \cdots + \lambda_\alpha^{\downarrow}(\rho_l) &\geqslant \lambda_1^{\downarrow}(\rho_k) + \cdots + \lambda_\alpha^{\downarrow}(\rho_k) \\
L_\alpha(\rho_l) &\geqslant L_\alpha(\rho_k)
\end{aligned} \tag{7.20}$$

and thus, $\rho_l \succ \rho_k \Rightarrow \Xi^{\text{BLS}}[\rho_l] \succ \Xi^{\text{BLS}}[\rho_k]$. Proving the reverse implication is the same. This completes the proof. □

The immediate implication from Proposition 7.9 is clear and interesting enough to give it its own corollary: Basic Linear Shrinkage preserves the QLO. Thus, when it comes to the Alarm-System we are building, applying the Basic Linear Shrinkage Estimator $\Xi^{\text{BLS}}[\rho]$ will not effect our model. For this reason, we ignore the basic linear shrinkage RMT filtering scheme and move onto eigenvalue clipping in the next section.

7.4.2 | Clipping

We now turn our attention to the eigenvalue clipping RMT filtering scheme $\Xi^{\text{CLP}}[\rho]$. This section will be organized as follows: We will state, and prove, a proposition that describes how the eigenvalue clipping RMT filtering scheme preserves the quantum lorenz ordering (QLO). Then we will revisit the alarm systems constructed with using the Pearson's ρ and upper tail dependence λ^{U} temporal state spaces $P_\Delta(\mathcal{S}; I)$ and $\Lambda_\Delta^{U}(\mathcal{S}; I)$, respectively. We choose these two as they performed well, but for different reasons. Upper tail dependence predicted every crash (for the most part), but with a few false alarms. Whereas Pearson's ρ missed a few crashes (about half), but signalled the fewest false alarms. Thus, we believe that studying these two metrics will offer insight into any intuitions we may end up drawing after studying eigenvalue clipping.

We would like to determine whether the filtering scheme $\Xi^{\text{CLP}}[\rho]$ preserves, or destroys, the quantum lorenz ordering (QLO). Indeed, if we want to see how clipping correlation matrices effects our alarm system, it would be useful to understand how so, exactly. To this end, we state proposition 7.10 showing the relationship between the raw, and clipped, QLO.

Proposition 7.10. *Let* $\rho_l, \rho_l \in P(S)$ *and fix* $\lceil d\beta \rceil$. *Then* $\forall \rho \in P(S)$, *we have* $\Xi^{CLP}[\rho_l] \succ \Xi^{CLP}[\rho_k]$ *if* $\rho_l \succ \rho_k$.

Proof. If $\rho_l \succ \rho_k$ we have $L_\alpha(\rho_l) \geqslant L_\alpha(\rho_k), \forall \alpha \in \{1, \ldots, d\}$. By choosing $\alpha \geqslant \lceil d\beta \rceil$ we have

$$\lambda_1^\downarrow(\rho_l) + \cdots + \lambda_\alpha^\downarrow(\rho_l) \geqslant \lambda_1^\downarrow(\rho_k) + \cdots + \lambda_\alpha^\downarrow(\rho_k)$$
$$\lambda_1^\downarrow(\rho_l) + \cdots + \lambda_{\lceil d\beta \rceil}^\downarrow(\rho_l) + \cdots + \lambda_\alpha^\downarrow(\rho_k) \geqslant \lambda_1^\downarrow(\rho_k) + \cdots + \lambda_{\lceil d\beta \rceil}^\downarrow(\rho_k) + \cdots + \lambda_\alpha^\downarrow(\rho_k)$$

in which case, we need only verify the inequality

$$\lambda_1^\downarrow(\rho_l) + \cdots + \lambda_{\lceil d\beta \rceil}^\downarrow(\rho_l) + \gamma_l(\alpha - \lceil d\beta \rceil) \geqslant \lambda_1^\downarrow(\rho_k) + \cdots + \lambda_{\lceil d\beta \rceil}^\downarrow(\rho_k) + \gamma_k(\alpha - \lceil d\beta \rceil)$$

where if we let $s_l = \lambda_1^\downarrow(\rho_l) + \cdots + \lambda_{\lceil d\beta \rceil}^\downarrow(\rho_l)$, and $s_k = \lambda_1^\downarrow(\rho_k) + \cdots + \lambda_{\lceil d\beta \rceil}^\downarrow(\rho_k)$ we can define $\gamma_l = \frac{(1-s_l)}{d-\lceil d\beta \rceil}$, with γ_k defined accordingly, and get

$$\begin{aligned}
s_l + \frac{(1-s_l)}{d-\lceil d\beta \rceil}(\alpha - \lceil d\beta \rceil) &\geqslant s_k + \frac{(1-s_k)}{d-\lceil d\beta \rceil}(\alpha - \lceil d\beta \rceil) \\
s_l - s_k &\geqslant \frac{(1-s_k)}{d-\lceil d\beta \rceil}(\alpha - \lceil d\beta \rceil) - \frac{(1-s_l)}{d-\lceil d\beta \rceil}(\alpha - \lceil d\beta \rceil) \\
s_l - s_k &\geqslant \frac{\alpha - \lceil d\beta \rceil}{d-\lceil d\beta \rceil}(s_l - s_k) \\
1 &\geqslant \frac{\alpha - \lceil d\beta \rceil}{d-\lceil d\beta \rceil}
\end{aligned}$$

where the last inequality holds since α is clearly less than d. This completes the proof. □

Proving that $\Xi^{CLP}[\rho_l] \succ \Xi^{CLP}[\rho_k]$ does not imply $\rho_l \succ \rho_k$ is a straightforward procedure, which we omit. It's easy for the reader to verify this for themselves. Furthermore, notice the limiting situations for $\Xi^{CLP}[\rho]$. If β is chosen such that every eigenvalue is "clipped", $\lambda(\Xi^{CLP}[\rho])$ simply converges to the uniform spectrum $\lambda(I_d)$. On the other hand, when β is chosen such that no eigenvalues is "clipped", $\lambda(\Xi^{CLP}[\rho])$ converges to the original spectrum $\lambda(\rho)$.

More importantly, the above proposition tells us something useful. Eigenvalue clipping generates, but does not destroy, quantum majorization relations. Indeed, since $\rho_l \succ \rho_k$ implies $\Xi^{CLP}[\rho_l] \succ \Xi^{CLP}[\rho_k]$, but not the other way around, a quantum majorization graph (QMG) can only become more dense as a result of eigenvalue clipping. Hence, we'd expect eigenvalue clipping to run the risk of making the QMG, and thus our alarm system, in a sense, "noisier". We

can try to verify this intuition by analysing the respective alarm systems for the Pearson's ρ and upper tail dependence λ^U correlation and dependence metrics.

8

Results

In this chapter, we will finally cover the results of our RUP. We want to train the RUP's detection regions to predict crashes on short, medium, and long time horizons corresponding to the sampling of correlations matrices over 2, 6, and 12 month (1 year) time-intervals. We believe these intervals represent realistic investment scenarios. As different correlation and dependence metrics are intended to capture and characterize different phenomenon, each of the metrics covered in chapter 6 were treated separately in our analysis. Indeed, it would be unreasonable to expect each QMG to yield the same number of optimal clusters for the different metrics. For instance, the QMG associated to Gini's γ^G might have 4 optimal clusters whereas Spearman's ρ^S might only have 2. One more thing to consider is the sheer computational complexity in the individuation of the urns sampling paths, as alluded to by Cirillo and Hüsler [2011]. Thus, for longer time horizons (3, and 6 months), we increase the size of our sliding windows from 10 to 20 and 40 so as to extend our prediction horizon and increase computational efficiency. We explain what we mean by this, exactly, and build individually optimized alarm systems for each correlation metric in section 8.2. We then take each of these optimized alarm systems and apply the various RMT filtering schemes from section 7.4 to their respective temporal state spaces and compare the effectiveness of the RMT schemes. Lastly, we compare the detection regions $\mathbb{P}[\tau_r \in [n, n+k]$ of our RUP, and corresponding alarm systems, to that of the confidence intervals associated to the LPPLS of section 8.4 and discuss the relative strengths and weaknesses of the two models.

8.1 | Evaluation Metrics

Now comes the question of optimizing our alarm system $\mathcal{A}_\Theta(\mathcal{Z})$. Following a similar approach to Cirillo and Hüsler [2011], we can perform a grid search over the parameter set $\Theta = (\gamma; (q, \mathcal{L}), (\delta_{2,3}, \delta))$ so as to maximize (or minimize) some optimality criterion for the accuracy of $\mathcal{A}_\Theta(\mathcal{Z})$. Since $\mathcal{A}_\Theta(\mathcal{Z}) \in \{0,1\}^N$ is a binary variable, potential criterion span the set of all evaluation metrics defined on the 2×2 contingency table (or confusion matrix) corresponding to the operating characteristics discussed earlier in chapter 4. Such a confusion matrix (CM) would take the form of table ?? with TP, FP, FN, TN denoting the number of "True Positives", "False Positives", "False

Confusion Matrix	$\mathcal{C}_{n+k} = 1$	$\mathcal{C}_{n+k} = 0$
$\mathcal{A}_n = 1$	TP	FP
$\mathcal{A}_n = 0$	FN	TN

Table 8.1: Confusion matrix

Negatives", and "True Negatives". We can easily frame the above CM in correspondence with the alarm systems $\mathcal{A}_\Theta(\mathcal{Z})$ operating characteristics (OC) according to table 8.2 with $\mathbb{P}[\mathcal{C}_{t+k}|\mathcal{A}_t]$

Diagnostics	$\mathcal{C}$	$\mathcal{C}^c$
$\mathcal{A}$	$\mathbb{P}[\mathcal{C}_{t+k}\|\mathcal{A}_t]$	$\mathbb{P}[\mathcal{C}^c_{t+k}\|\mathcal{A}_t]$
$\mathcal{A}^c$	$\mathbb{P}[\mathcal{C}_{t+k}\|\mathcal{A}^c_t]$	$\mathbb{P}[\mathcal{C}^c_{t+k}\|\mathcal{A}^c_t]$

Table 8.2: Operating characteristics

corresponding to the probability of $\mathcal{A}_\Theta(\mathcal{Z})$ signaling a true positive TP, and so on for the other elements.

Though the optimality of an alarm system, as per chapter 4 and the work of Antunes M and FK [2003], is defined according minimizing the probability of casting a false alarm $\mathbb{P}[\mathcal{C}^c_{n+k}|\mathcal{A}_n]$, or maximizing the probability of casting a correct alarm $\mathbb{P}[\mathcal{C}_{n+k}|\mathcal{A}_n]$, it would be hard for us to compare our $\mathcal{Z} \in \mathsf{RUP}(S, C, U, d)$ based alarm system $\mathcal{A}_\Theta(\mathcal{Z})$ to that of the (say) LPPLS confidence indicators of chapter 5 since our $\mathcal{A}_\Theta(\mathcal{Z})$ casts alarms every 40 days whereas the LPPLS confidence indicators casts alarms every day. Hence, the OC for $\mathcal{A}_\Theta(\mathcal{Z})$ would look very different than the OC for the LPPLS confidence indicators. Thus, we should try to use a more balanced metric that allows for the direct comparison of the two models.

Furthermore, for the purposes of our $\mathcal{A}_\Theta(\mathcal{Z})$ alarm system, we found that analyzing, and optimizing with respect to, the CM yields the same results as the optimality criterion defined on the OC as per chapter 4. That is to say, for the sake of simplicity, we will find an optimal parameter set Θ^* with respect to a metric defined on the CM.

We introduce the metric we will be using and explain why we use this metric in a moment. The Mathews Correlation Coefficient MCC, or phi coefficient ϕ, is defined below.

Definition 8.1 (Mathews Correlation Coefficient)**.** The phi coefficient ϕ is defined as

$$\phi[\cdot] = \frac{\mathsf{TP} \times \mathsf{TN} - \mathsf{FP} \times \mathsf{FN}}{\sqrt{(\mathsf{TP}+\mathsf{FP})(\mathsf{TP}+\mathsf{FN})(\mathsf{TN}+\mathsf{FP})(\mathsf{TN}+\mathsf{FN})}} \in [-1, 1] \tag{8.1}$$

where TP, FP, FN, and TN are as before and the argument $\cdot$ in $\phi[\cdot]$ denotes some binary classifier.

Since we are trying to predict just 8 crashes $\mathcal{C} = \{\mathcal{C}_1, \ldots, \mathcal{C}_8\}$ with over a 100 correlation matrices, we have a severely imbalanced data set. Furthermore, as with any alarm system, false alarms are a serious issue. Because of this, we should use a balanced metric that understands the trade off between signaling correct predictions (TP's) and accumulating false alarms (FP's). For instance, we can use easily optimize $\mathcal{A}_\Theta(\mathcal{Z})$ with respect to any desirable CM element, such as the number of TP's and FP's, by setting the detection region threshold γ to a relatively extreme level (close to 1 or 0). By setting $\gamma = 0$, or close to 0, we would maximize the number of TP's (predict every crash) disregarding the number of false alarms. Likewise, we could minimize the number of FP's (signal no false alarms) by setting $\gamma = 1$, or close to 1 disregarding the number of correct predictions.

Hence, unlike the raw FP's and TP's, the phi coefficient ϕ will naturally try to maximize the number of TP's using the number of FP's as a constraint. This is precisely the sort of balance we want our alarm system to have. Indeed, we want the alarm system to understand the trade off between being correct (maximizing TP) and being believable (minimizing FP). In fact, the ϕ coefficient is precisely the Pearson correlation coefficient of two binary variables, that is $\phi = \rho(\{0,1\}^N, \{0,1\}^N)$. Thus maximizing our alarm system $\mathcal{A}_\Theta(\mathcal{Z})$ with respect to ϕ will tell us how closely the our alarms are tracking with reality. In this way, when $\phi = -1$, our alarms predict the exact opposite of reality. When $\phi = 1$, our alarms predict reality, exactly. When $\phi = 0$, our alarms are no better than random guessing.

8.2 | Correlation Metrics

As promised, we present the results of our RUP Alarm System $\mathcal{A}_\Theta(\mathcal{Z})$ for each correlation metric, independently. To keep this section as simple and readable as possible, we only present the results for the "optimal" alarm systems for each $k = \{2, 4, 6\}$ corresponding to the different 4, 8, and 12 month time horizons according to the detection region $\mathbb{P}[\tau_r \in [n, n+k)|\tau_1, \ldots, \tau_{r-1}]$. Each alarm system $\mathcal{A}_{\Theta^*}(\mathcal{Z})$ is optimized by finding the optimal parameter set $\Theta^* = (\gamma^*, (q, \mathcal{L})^*, (\delta_{2,3}, \delta))$ as per the argument

$$\Theta^* = \underset{\Theta}{\operatorname{argmax}}\, \phi[\mathcal{A}_\Theta(\mathcal{Z})] \tag{8.2}$$

via a simple grid search using the ϕ correlation coefficient. Furthermore, we only display diagnostic plots for one selected alarm system per correlation metric. In each subsection, however, we display each temporal state space $P_\Delta(\mathcal{S}; I)$ and corresponding clustering for the associated correlation metirc. Unless stated otherwise, each temporal state space $P_\Delta(\mathcal{S}; I)$ is characterized by the $\Delta = (40, 100)$ sliding window structure, as stated before.

8.2.1 | Pearsons ρ

We start by plotting the clustering associated to Pearson's ρ in figure 8.1. This clustering $\vec{Y} \in C^N$ happens to correspond to the discriminating thresholds equalling $(d_1, d_2) = (0.48, 0.64)$ for the $\theta_n = \theta(\rho_n)$ risk measure of chapter 3. That is to say

$$Y_n = \begin{cases} c_1, & \text{if } \theta(\rho_n) \in [0, 0.48) \\ c_2, & \text{if } \theta(\rho_n) \in [0.48, 0.64) \\ c_3, & \text{if } \theta(\rho_n) \in [0.64, 1] \end{cases} \tag{8.3}$$

whenever the n^{th} correlation matrix ρ_n is assigned to a cluster as per the algorithm described in definition 7.4. We notice this clustering is good. Indeed, all the major crashes including the 2008 "Financial Crisis" and 2020 "COVID-19" crash are surrounded by c_2 and c_3 coloured marbles. Furthermore, the 2015 "Chinese Selloff" and 2018 "Crypto Crash" are preceded by c_3 coloured marbles. On the other hand, the 2011 "Bear Market" does not seem to be well encapsulated by this clustering. Having said that, there seems to be a lot of c_3 coloured marbles as compared to c_2 marbles meaning our $\mathcal{Z} \in \text{RUP}(S, C, U, d)$ may be susceptible to a high number of false alarms. Let us see the results for ourselves.

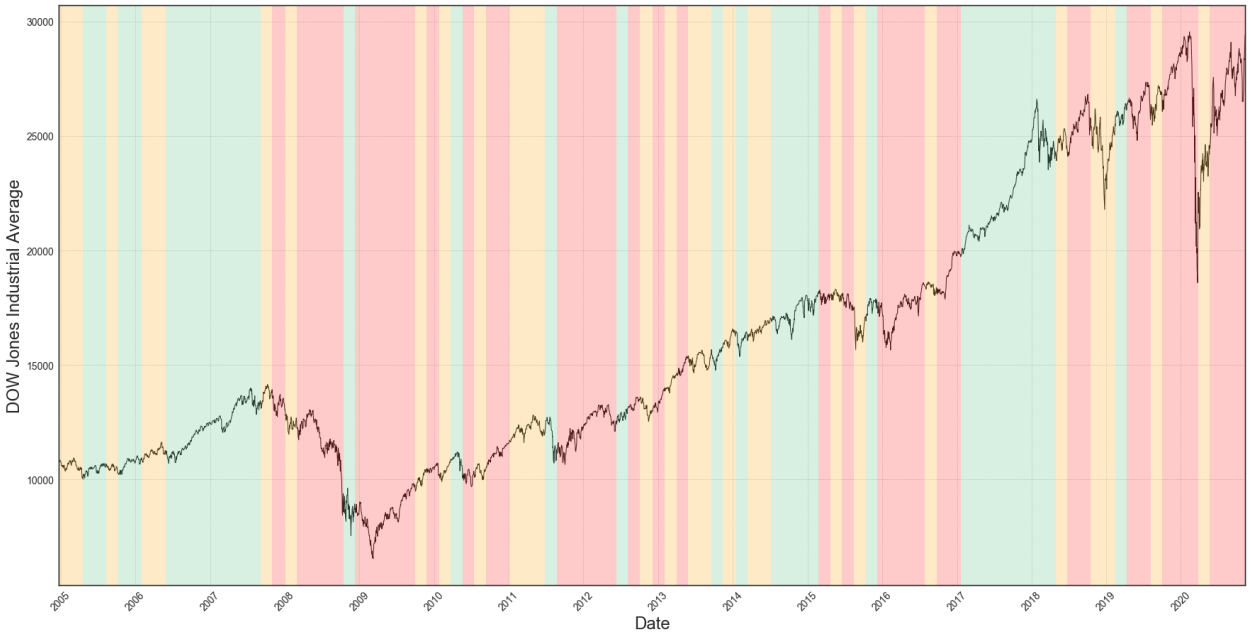


Figure 8.1: Pearson ρ $P_{\Delta}(\mathcal{S}, I)$: Clustering

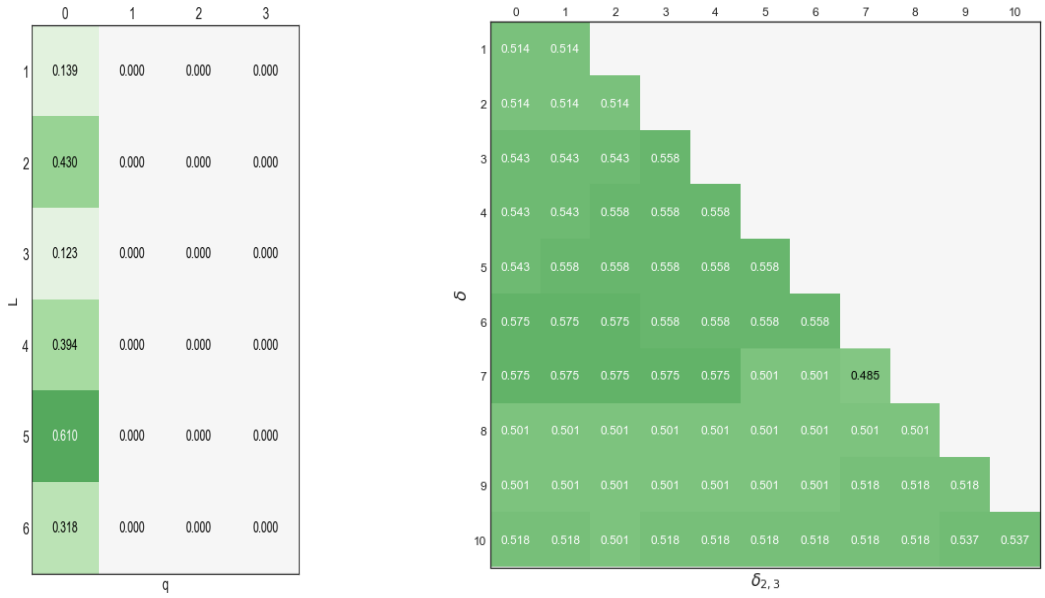
RUP Performance: Pearson Correlation $P_{\Delta}(\mathcal{S}, I)$			
Operating Characteristics	4-month	8-month	12-month
ϕ coefficient	0.61	0.61	0.64
Number of correct Non-alarms	97	97	96
Number of correct Alarms	4	4	5
Number of False Alarms	1	1	2
Number of undetected Crashes	4	4	3

Table 8.3: Pearson ρ $P_{\Delta}(\mathcal{S}, I)$: Operating characteristics

In any case $k = 2, 4, 6$ for 4, 8, and 12 month time horizons, Pearsons ρ is yielding a ϕ coefficient $\phi \geqslant 0.61$. This is quite a strong (positive) binary correlation with the actual occurrence of stock market crashes. However, as we noted at the onset, there are several undetected crashes (false negatives) in every case. In order for Pearsons ρ to achieve a high ϕ coefficient, it must sacrifice detecting a few crashes. This can be read as, any parameter set $\Theta = (\gamma; (q, \mathcal{L}), (\delta_{2,3}, \delta)$ capable of predicting most crashes would also (necessarily) have to generate a lot of false alarms. This can be interpreted as Pearson's ρ inducing a QMG that is noisier than we would like.

Our grid search honed in on a parameter set Θ^* that limited the number of false alarms. This corresponds with the notion of optimality discussed in chapter 4 and by Antunes M and FK [2003].

Interestingly, Pearsons ρ was no better at predicting crashes on a 8 month time horizon as it was on a 4 month time horizon. The 12 month time horizon prediction was slightly better. Let us now turn our attention to the grid search used to obtain the 8 month model.



(a) ϕ varying $(q, \mathcal{L})$ **(b)** ϕ varying $(\delta_{2,3}, s)$

Figure 8.2: Pearsons ρ optimal stopping rule and reinforcement hyper paramaters
In 8.2(a) we compute ϕ varying $(q, \mathcal{L})$. In 8.2(b) we fix $(q, \mathcal{L}) = (2, 3)$ varying $(\delta_{2,3}, \delta)$

Referring to figure 8.2 we notice that the 8 month pearson model does not seem to be benefiting much from the q anti-clustering parameter, but rather wants just increasing $\mathcal{L}$ instead. Additionally we see some interesting activity in figure 8.2(b). In this snapshot, adding the dependence parameter $\delta_{2,3}$ to the reinforcement mechanism seems to enhance the model. For instance, starting from $\delta = 6$, adding more and more $\delta_{2,3}$ seems to be improving the model. Let us take a look at γ.

Referring to figure 8.3 we plot both the number of false alarms and the ϕ coefficient as a function of the detection region threshold γ. In this particular plot, γ ranged from 0 to 0.60. Clearly, increasing γ decreased the number of false alarms and increased the ϕ coefficient. With the exception of $\gamma \in [0.27, 0.28]$, the number of false alarms and ϕ coefficient evolved in direct dis-correspondence with one another as γ increased towards 0.60. This illustrates how the ϕ coefficient naturally tries to reduce the alarm systems false alarms. Hence, although it missed several alarms, the model is very believable since a crash actually occurs $4/5 = 80\%$ of the time whenever it casts an alarm. We display the optimal hyper parameterizations Θ^* for the three

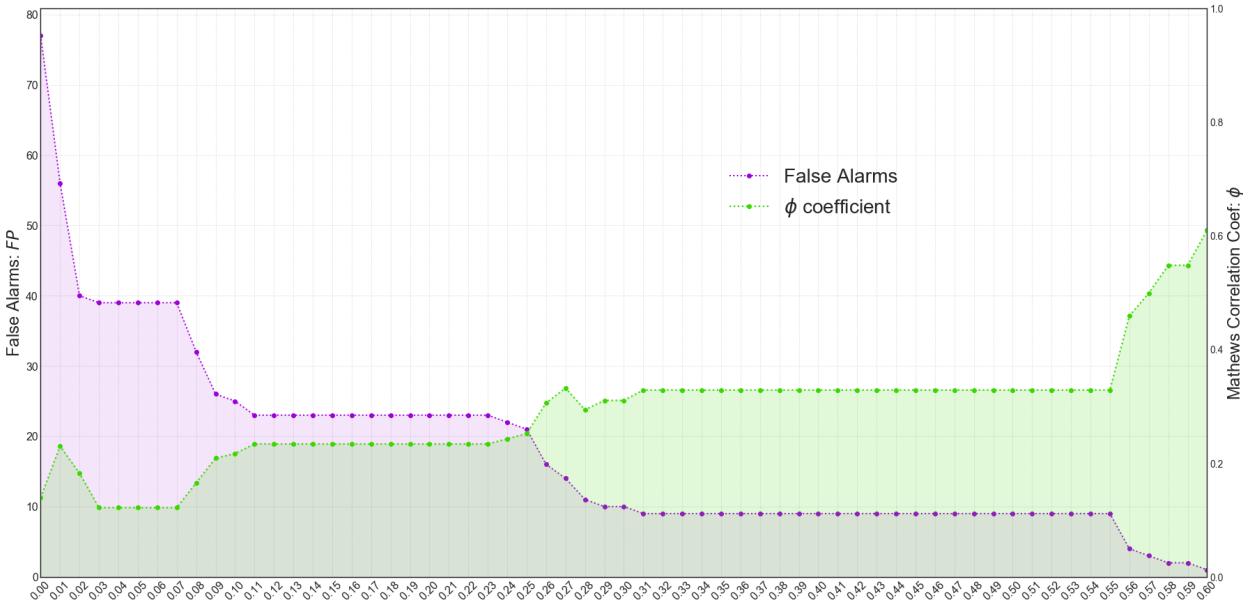


Figure 8.3: $\mathrm{P}_{\Delta}(\mathcal{S};\mathrm{I})$ 8 month ($\mathrm{k}=4$): ϕ and # of False Alarms varying $\gamma \in [0, 0.6]$

models next.

Referring to the table above, we repeated the above analysis for all three 4, 8, and 12 month

Optimal Hyper-parameters: Pearson Correlation $\mathrm{P}_{\Delta}(\mathcal{S},\mathrm{I})$			
Θ^*	4-month	8-month	12-month
γ	0.48	0.60	0.13
$(q, \mathcal{L})$	$(0, 5)$	$(0, 5)$	$(1, 2)$
$(\delta_{2,3}, \delta)$	$(2, 6)$	$(3, 7)$	$(2, 7)$

Table 8.4: Pearson ρ $\mathrm{P}_{\Delta}(\mathcal{S},\mathrm{I})$: Hyper-parameters

models. We see that the 4 and 8 month models are essentially identical with exception to the choice of γ. Indeed, we can even change the reinforcement mechanism $(3, 7)$ for the 8 month model to the $(2, 6)$ mechanism without changing the ϕ coefficient. On the other hand, the 12 month model seems to like the anti-clustering parameter q. Naturally, adding this parameter will reduce the number of viable sampling sequences, and hence, the detection region decreased to 0.13 as a result. Nevertheless, the 12 month model was able to make one more correct prediction, at the expense of an additional false alarm. This illustrates what the ϕ coefficient is doing. Indeed, ϕ recognizes the imbalanced nature of the positive class (crashes). In other words, adding a correct prediction, according to ϕ, is worth accumulating a certain number of extra false alarms

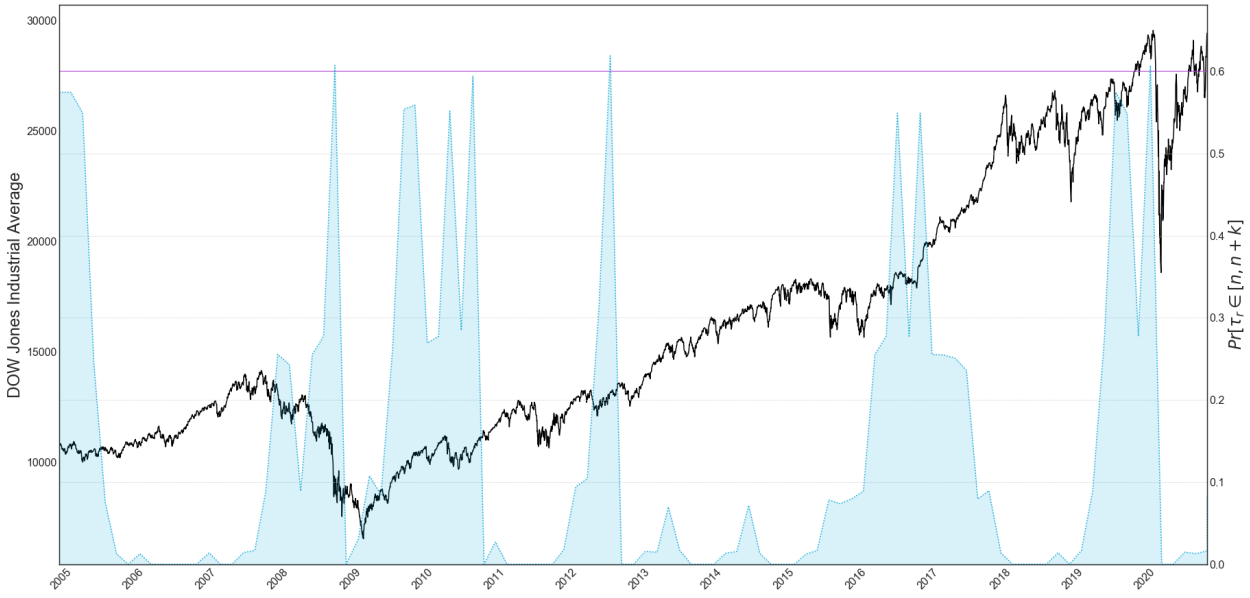


Figure 8.4: Pearson ρ $P_{\Delta}(\mathcal{S}, I)$: Detection region: 8-month time horizon

(1 in this case).

We conclude this section by plotting the detection region for the 8 month model along with the threshold γ. We refer to figure 8.4. Analyzing this model allows to see which crashes the Pearson correlation model(s) missed. Here we see that the 2011 "Bear Market", 2015 "Chinese Selloff", 2018 "Crypto Crash", and the last of the waves associated to the 2008 great recession, i.e. the 2008 "Great Recession II" in January of 2009. However, it correctly predicted the initial 2008 "Financial Crisis", the first 2008 "Great Recession I", and the two extreme drawdowns associated to the 2020 "COVID-19" crashes.

8.2.2 | Ginis γ^{G}

We move to the clustering associated to Gini's γ^{G} in figure 8.5. This clustering $\vec{Y} \in C^{N}$ happens to correspond to the discriminating thresholds equalling $(d_1, d_2) = (0.48, 0.64)$ for the $\theta_n = \theta(\rho_n)$ risk measure of chapter 3. That is to say

$$Y_n = \begin{cases} c_1, & \text{if } \theta(\rho_n) \in [0, 0.48) \\ c_2, & \text{if } \theta(\rho_n) \in [0.48, 0.54) \\ c_3, & \text{if } \theta(\rho_n) \in [0.54, 1] \end{cases} \tag{8.4}$$

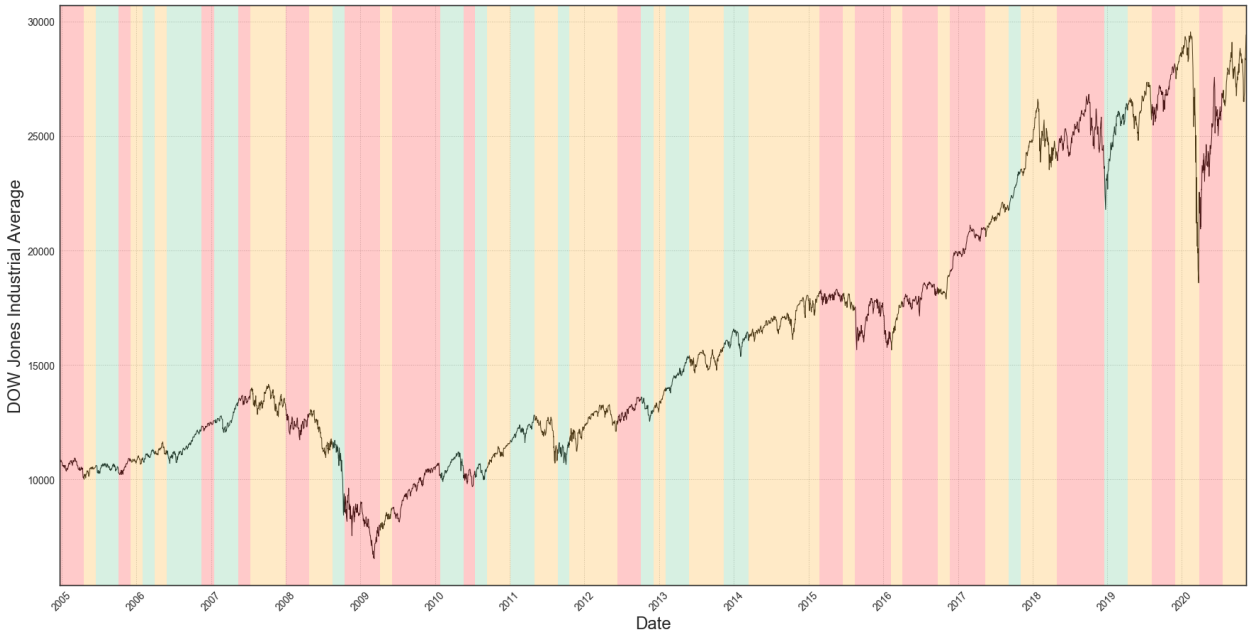
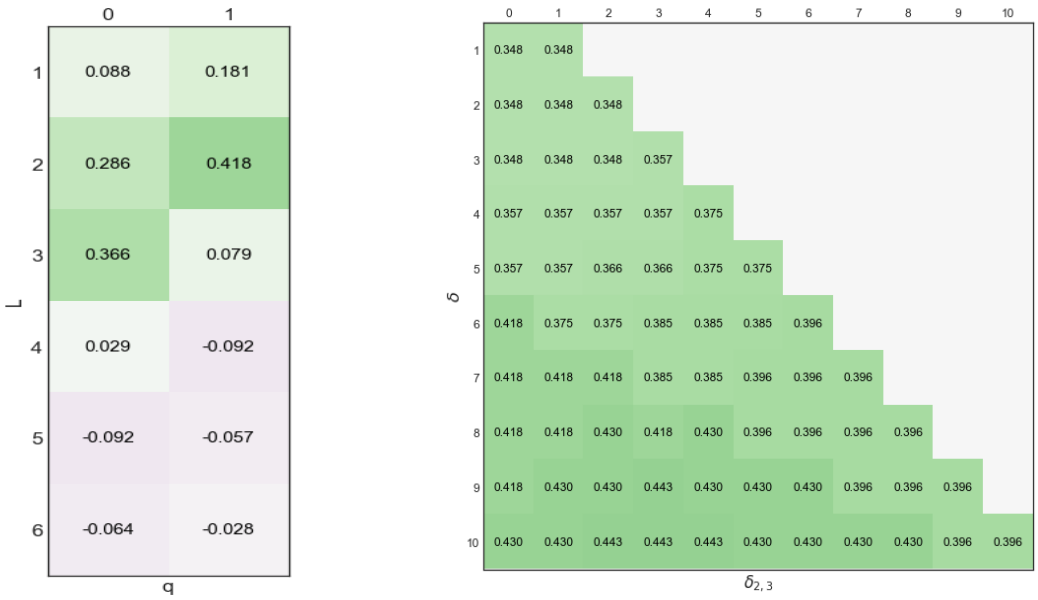


Figure 8.5: Gini correlation $\Gamma\Delta(\mathcal{S}, \mathrm{I})$: Clustering

whenever the n^{th} correlation matrix ρ_n is assigned to a cluster as per the algorithm described in definition 7.4. We notice this clustering is much more yellow and red than that of Pearson's ρ clustering. However, all the major crashes including the 2008 "Financial Crisis" and 2020 "COVID-19" crash are surrounded by c_2 and c_3 coloured marbles along with the smaller events such as the 2015 "Chinese Selloff" and 2018 "Crypto Crash" as well. The 2011 "Bear Market" is briefly preceded by yellow. Furthermore, there seems to be a lot of c_3 coloured marbles, especially between 2014-2016, meaning our $\mathcal{Z} \in \mathrm{RUP}(S, C, U, d)$ may be susceptible to signalling a high number of false alarms over this time period.

Referring to figure 8.6(a), we see our grid search didn't honed in on levels $\mathcal{L} \leqslant 3$. Furthermore, it even exploited the anti-clustering parameter q, which we didn't expect on the shorter time horizon. However, looking back at the clustering in figure 8.5, we see that red regions are particularly dense. Hence, we believe the RUP was trained to detect the probability of sampling multiple c_3 marbles consecutively. As a result, even over the shorter time horizon, the anti-clustering parameter q helped improve the model.

We turn to figure 8.6(b) to analyze the $(\delta_{2,3}, \delta)$ grid search. We see the model wanted to increase, not only the strength of reinforcement δ, but also the dependence $\delta_{2,3}$ parameter as well. Referring to the clustering in figure 8.5, we see several periods of red that were preceded by dense regions of yellow. Hence, by increasing dependence between the sampling of c_2 and c_3 colored



(a) ϕ varying $(q, \mathcal{L})$ **(b)** ϕ varying $(\delta_{2,3}, s)$

Figure 8.6: Gini Correlation $\Gamma^{G}_{\Delta}(\mathcal{S}, I)$: Grid search
In 8.6(a) we compute ϕ varying $(q, \mathcal{L})$. In 8.6(b) we fix $(q, \mathcal{L}) = (2, 3)$ varying $(\delta_{2,3}, \delta)$

marbles, the RUP enabled itself to anticipate entering dense periods of red.

Since we see the anti-clustering parameter $q > 0$, especially for the relatively low $k = 2$, we'd expect to see the detection region threshold to be low as well. Indeed, referring to figure 8.7 we see that our optimal γ^* is just 0.11 as compared to something on the order of 0.48, as was the case for Pearson's ρ 4 month model. This plot also illustrates what the ϕ coefficient is doing. The instant γ reaches 0.11, our FP's decreased and ϕ rose. However, as γ continued, ϕ dropped even though FP dropped as well, suggesting that the increase from $\gamma = 0.11$ to $\gamma = 0.12$ cost our model at least one TP.

In table 8.5 we see the Gini correlation models operating characteristics. Even though the ϕ coefficient for the 4 month is lower than that of Pearson's ρ 4-month model, Gini was able to predict 7/8 crashes, with 17 false alarms. This is interesting suggesting that Pearson's ρ, by comparison, simply had a blind spot to those 4 crashes it couldn't see. So instead of searching for a parameter set to detect them, it rose γ to 0.48 instead to reduce the number of false alarms. On the other hand Gini correlation sees those crashes and tries to optimize ϕ by capturing as many crashes as possible. Over longer time horizons, however, Gini was able to reduce the number of FP's by enough to generate a $\phi = 0.63$, as compared to the 0.61 of the corresponding Pearson's ρ 8 month model. We see something similar in the 12 month case.

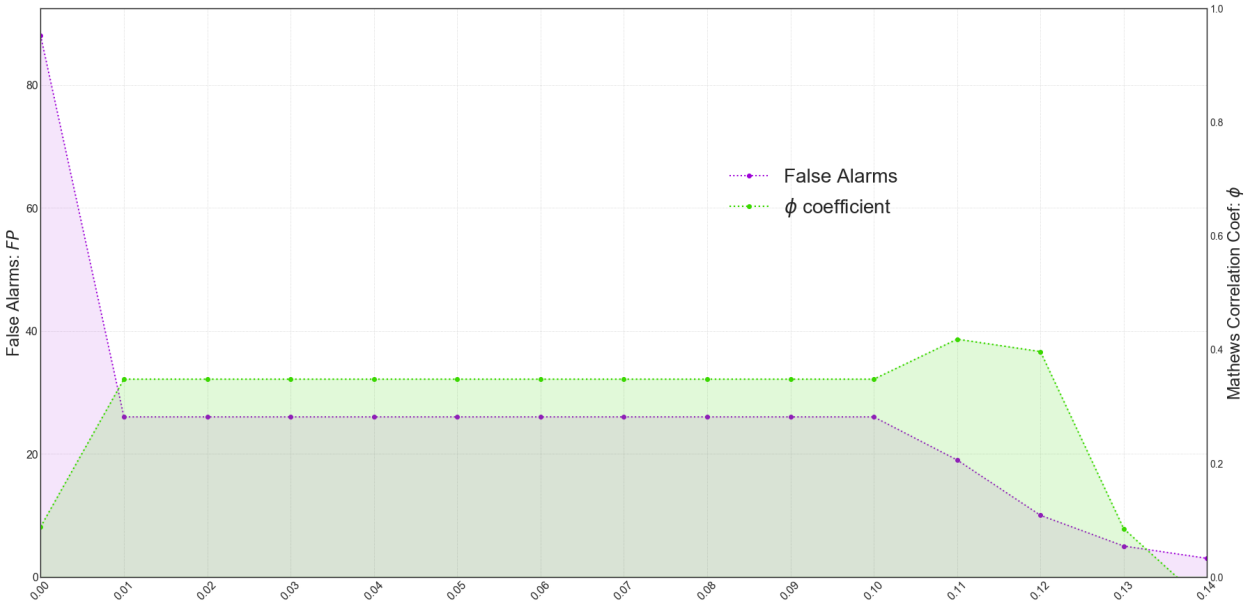


Figure 8.7: Gini Correlation $\Gamma^{G}_{\Delta}(\mathcal{S}, I)$ 4 month ($k = 2$): ϕ and # of False Alarms varying $\gamma \in [0, 0.14]$

RUP Performance: Gini Correlation $\Gamma^{G}_{\Delta}(\mathcal{S}, I)$			
Operating Characteristics	4-month	8-month	12-month
ϕ coefficient	0.50	0.63	0.68
Number of correct Non-alarms	81	91	93
Number of correct Alarms	7	7	7
Number of False Alarms	17	7	5
Number of undetected Crashes	1	3	1

Table 8.5: Gini correlations $\Gamma_{\Delta}(\mathcal{S}, I)$: Operating characteristics

In table 8.6, we see Gini correlation took advantage of the anti-clustering parameter in all

Optimal Hyper-parameters: Gini Correlation $\Gamma^{G}_{\Delta}(\mathcal{S}, I)$			
Θ^*	4-month	8-month	12-month
γ	0.11	0.22	0.06
$(q, \mathcal{L})$	$(1, 2)$	$(2, 3)$	$(2, 3)$
$(\delta_{2,3}, \delta)$	$(2, 8)$	$(6, 9)$	$(4, 7)$

Table 8.6: Gini Correlation $\Gamma^{G}_{\Delta}(\mathcal{S}, I)$: Hyper-parameters

three models. For reasons mentioned earlier, related to the associated clustering in figure 8.5,

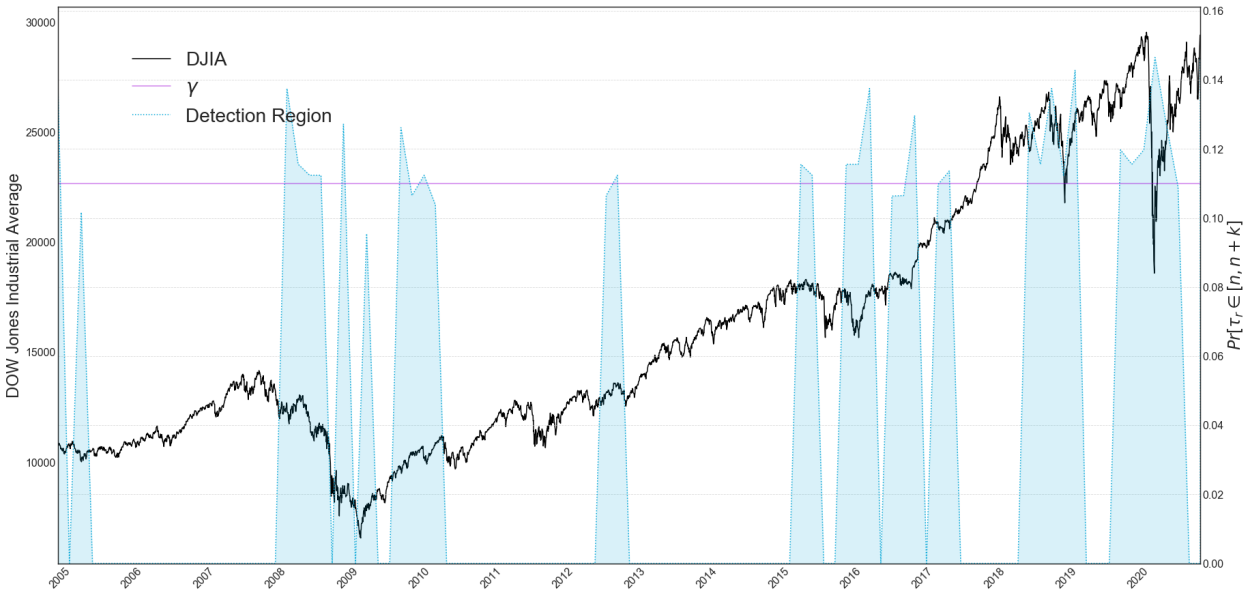


Figure 8.8: 4 month Gini correlation detection region with $\gamma = 0.11$

this makes sense. Interestingly, the γ for the 8 month model is much larger than the γ for the 12 month model even though they have the same $(q, \mathcal{L})$ tuple. Lastly, we refer to figure 8.8 for the associated detection region. We can infer from this plot that the only crash Gini was missing was the 2011 "Bear Market". Other than that, Gini correlation seemed to hone in on the others.

8.2.3 | Tail-Dependence λ^U

We now turn our attention to the upper Tail-Dependence λ^U metric. We plot the process $\vec{Y} = \{Y_1, \ldots, Y_N\} \in C^N$ corresponding to the set of correlation matrices $\Lambda^U_\Delta(\mathcal{S}; I)$ of cardinality $|\Lambda^U_\Delta(\mathcal{S}; I)| = 106$ in figure 8.9, which we now turn our attention to. This clustering $\vec{Y} \in C^N$ happens to correspond to the discriminating thresholds equalling $(d_1, d_2) = (0.54, 0.66)$ for the $\theta_n = \theta(\lambda^U_n)$ risk measure of chapter 3. That is to say

$$Y_n = \begin{cases} c_1, & \text{if } \theta(\lambda^U_n) \in [0, 0.54) \\ c_2, & \text{if } \theta(\lambda^U_n) \in [0.54, 0.66) \\ c_3, & \text{if } \theta(\lambda^U_n) \in [0.66, 1] \end{cases} \tag{8.5}$$

whenever the n^{th} tail dependence matrix λ^U_n is assigned to a cluster as per the algorithm described in definition 7.4. We notice this clustering is pretty good. Again, every crash including the 2008

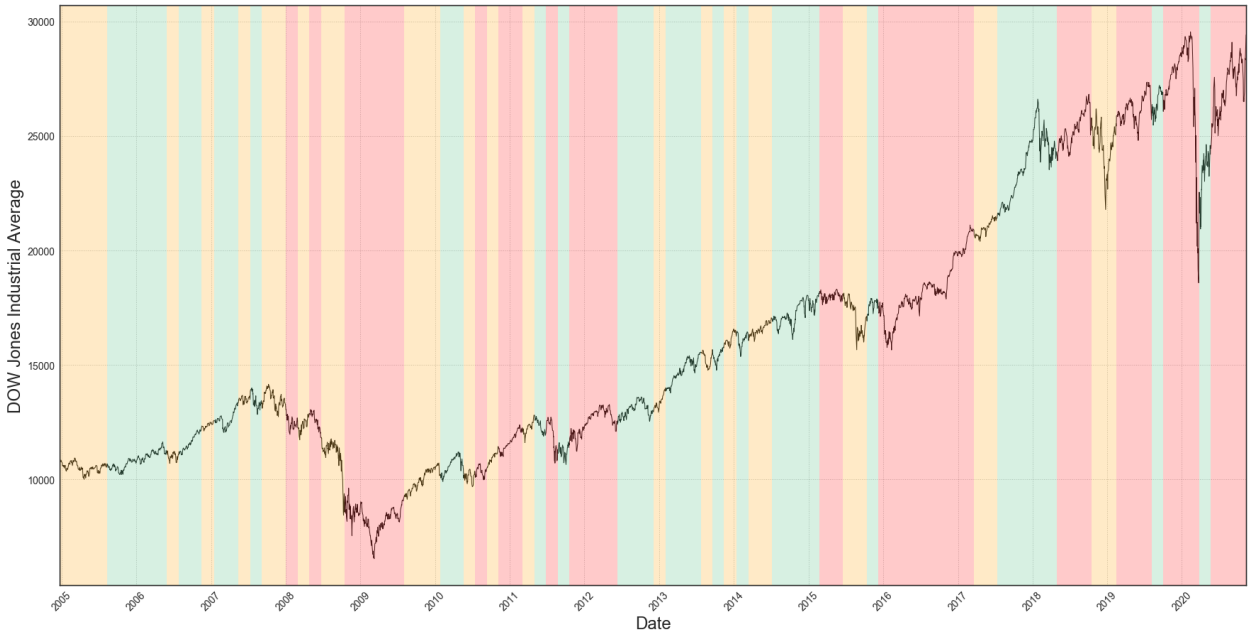


Figure 8.9: Tail-Dependence $\Lambda_{\Delta}(\mathcal{S}, \mathrm{I})$: Clustering

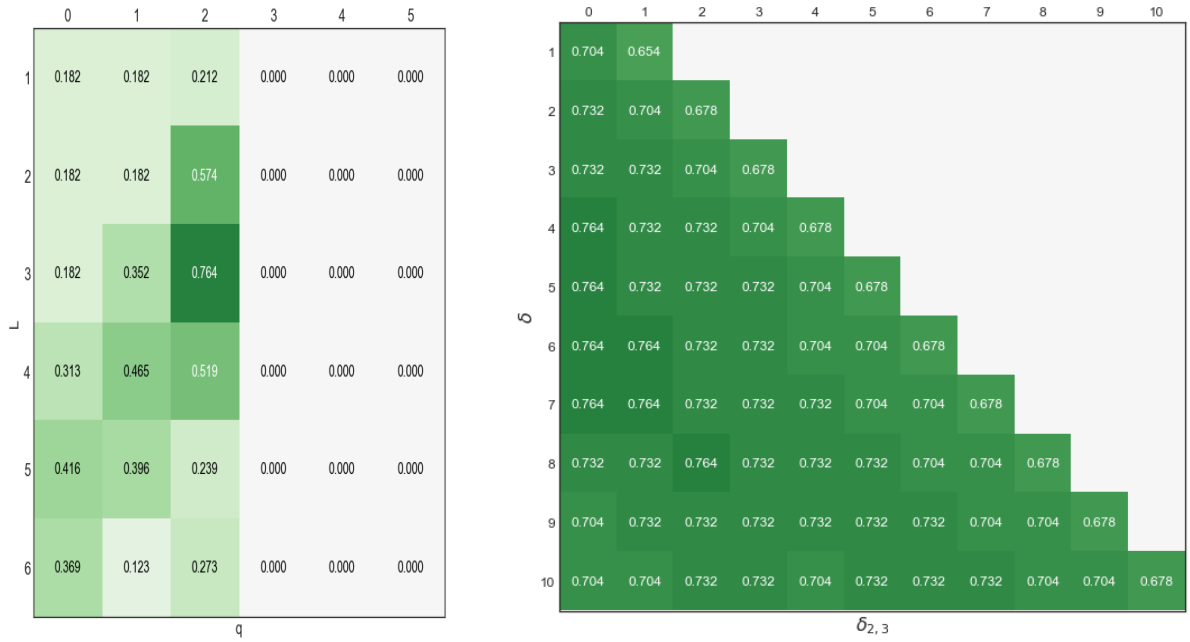
"Financial Crisis" and 2020 "COVID-19" crash are surrounded by c_2 and c_3 coloured marbles in addition to the 2011 "Bear Market", 2015 "Chinese Selloff" and even the 2018 "Crypto Crash". Additionally, there seems to be a (roughly) equal amount of c_1, c_2, and c_3 coloured marbles in the plot. Which, together with the dense regions of red surrounding known crashes, suggests that the process $\mathcal{Z} \in \mathrm{RUP}(S, C, U, d)$, corresponding to tail dependence, should be able to predict most (if not all) crashes with relatively few false alarms.

RUP Performance: Tail-Dependence $\Lambda^{U}_{\Delta}(\mathcal{S}, \mathrm{I})$			
Operating Characteristics	4-month	8-month	12-month
ϕ coefficient	0.57	0.70	0.76
Number of correct Non-alarms	89	91	93
Number of correct Alarms	7	8	8
Number of False Alarms	9	7	5
Number of undetected Crashes	1	0	0

Table 8.7: Upper Tail $\Lambda^{U}_{\Delta}(\mathcal{S}, \mathrm{I})$: Operating characteristics

In any case $k = 2, 4, 6$ for 4, 8, and 12 month time horizons, tail dependence λ^{U} is yielding a ϕ coefficient $\phi \geqslant 0.57$. This is quite a strong (positive) binary correlation with the actual

occurrence of stock market crashes. Additionally, as we noted at the onset, every crash on 8 and 12 month time horizons is being detected with relatively few false alarms. The 4 month model missed just one crash. Tail dependence λ^{u} achieves a high ϕ coefficient, without having to sacrifice signalling a lot of false alarms. As expected, tail dependence λ^{u} became better and better at predicting crashes over longer time horizons. The 12 month time horizon prediction was clearly the best with a ϕ coefficient of 0.76 and just 5 false alarms. Let us now turn our attention to the grid search used to obtain the 12 month model.



(a) ϕ varying $(q, \mathcal{L})$ **(b)** ϕ varying $(\delta_{2,3}, s)$

Figure 8.10: Upper Tail $\Lambda^{u}_{\Delta}(\mathcal{S}, I)$: Grid search
In 8.6(a) we compute ϕ varying $(q, \mathcal{L})$. In 8.10(b) we fix $(q, \mathcal{L}) = (2, 3)$ varying $(\delta_{2,3}, \delta)$

Referring to figure 8.10 we notice that the 12 month tail dependence model benefits quite a bit from the q anti-clustering parameter. Indeed, instead of increasing $\mathcal{L}$ closer to (say) 6, the grid search honed in on a larger q. We suspected this might happen over long time horizons, and we see it here. Additionally, we see some interesting activity in figure 8.10(b). In this snapshot, adding the dependence parameter $\delta_{2,3}$ to the reinforcement mechanism seems to enhance the model. For instance, starting from $\delta = 8$, adding a dependence factor from $\delta_{2,3} = 0$ to $\delta_{2,3} = 3$ increases the ϕ coefficient from 0.732 to 0.764.

Referring to figure 8.3 we plot both the number of false alarms and the ϕ coefficient as a function of the detection region threshold γ. In this particular plot, γ ranged from 0 to 0.08.

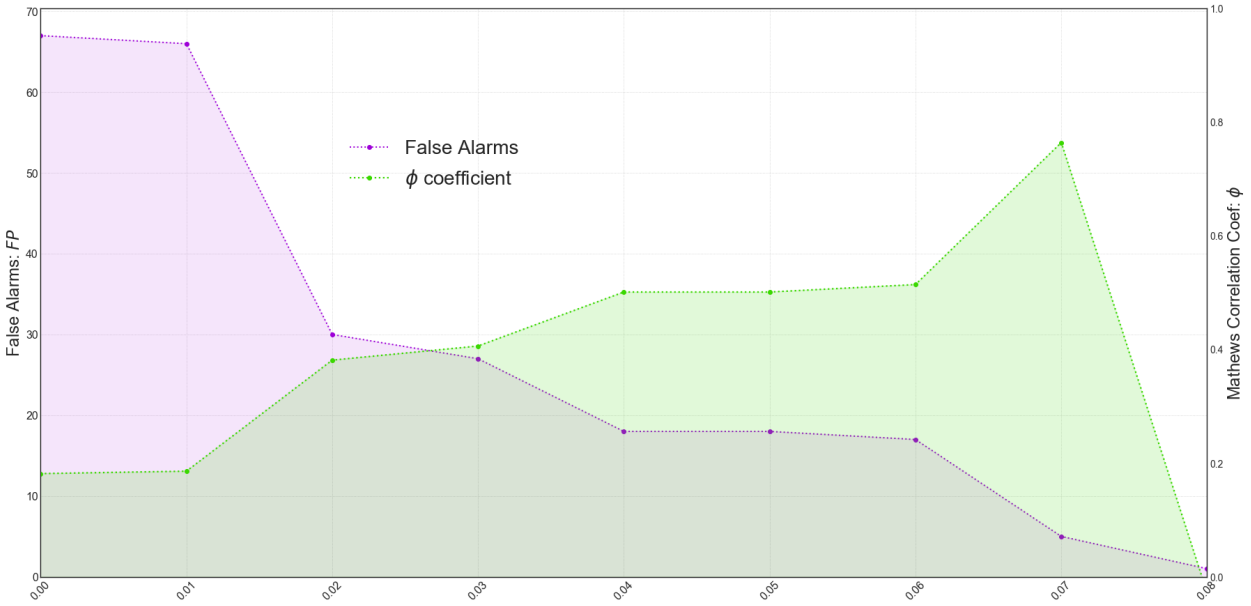


Figure 8.11: Upper Tail $\Lambda^{U}_{\Delta}(\mathcal{S}, I)$: γ

Clearly, increasing γ decreased the number of false alarms and increased the ϕ coefficient up until $\gamma = 0.07$. Beyond $\gamma > 0.07$, the number of false alarms may have decreased even further, however, this came at the expense of losing several correct predictions. We checked and any γ can be increased up to 0.0705 before losing several correct predictions, at which point, ϕ coefficient plummets to a $\phi = -0.12$ once $\gamma = 0.08$. This illustrates how the ϕ coefficient naturally tries to reduce the alarm systems false alarms. Hence, although it signalled a few false alarms (5), the model predicted every crash with a high very degree accuracy which we believe is impressive. We display the optimal hyper parameterizations Θ^* for the three models next.

Optimal Hyper-parameters: Upper Tail $\Lambda^{U}_{\Delta}(\mathcal{S}, I)$			
Θ^*	4-month	8-month	12-month
γ	0.43	0.55	0.07
$(q, \mathcal{L})$	$(0, 3)$	$(0, 3)$	$(2, 3)$
$(\delta_{2,3}, \delta)$	$(0, 8)$	$(0, 8)$	$(2, 7)$

Table 8.8: Upper Tail-Dependence hyper-parameters

Referring to table 8.8 above, we repeated the above analysis for all three 4, 8, and 12 month models. We see that the 4 and 8 month models are essentially identical with exception to the

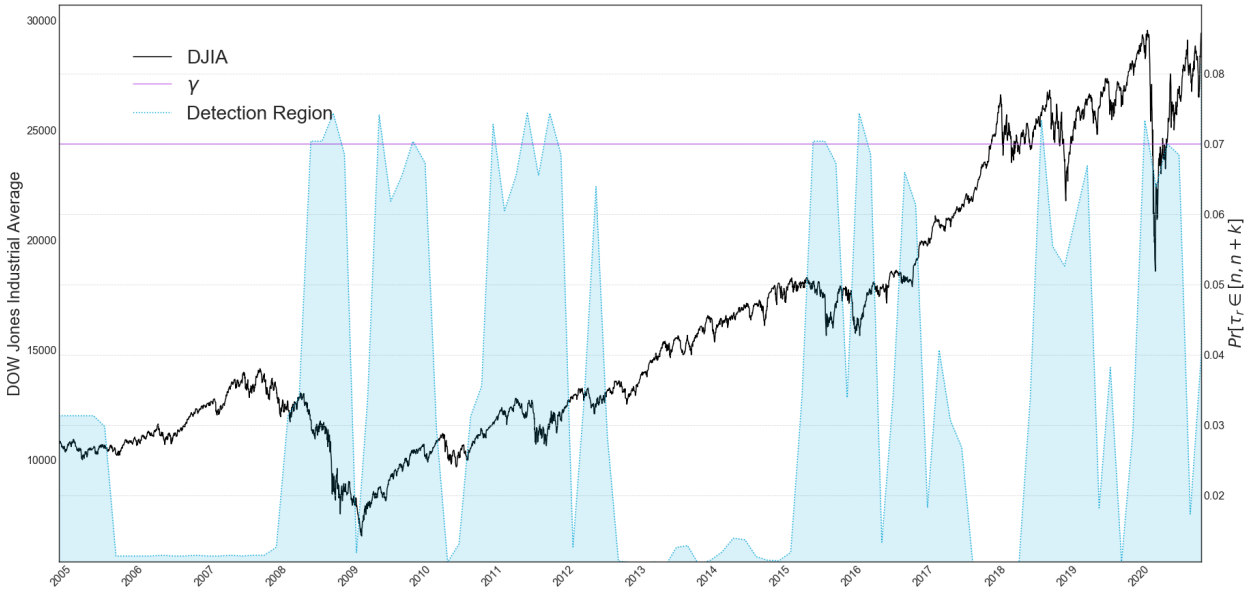


Figure 8.12: Tail-Dependence detection region with $\gamma = 0.07$

choice of γ. Indeed, neither returned an anti-clustering coefficient and both found a reinforcement mechanism without modelling dependence between c_2 and c_3 coloured marbles. Retroactively, perhaps we could have anticipated this since, with exception to the 2007-2008 range in figure 8.9, we don't see too many dense regions of yellow preceding periods of red. However, we believe introducing dependence with $(\delta_{2,3}, \delta)$, as per 8.10(b) mitigated a few false alarms for the 12 month model. Additionally, introducing the anti-clustering parameter $q = 2$ helped eliminate the two false alarms between the 8 month and 12 month models. Naturally, adding this parameter $q = 2$ will reduce the number of viable sampling sequences, and hence, the detection region decreased to 0.07 as a result. All in all, tail dependence was able predict every crash for both the 8 month and 12 month models: the only metric we've seen be able to do this since Spearman ρ^S. However, tail dependence λ^U did so with much fewer false alarms. This illustrates what the ϕ coefficient is doing. Indeed, ϕ recognizes the imbalanced nature of the positive class (crashes). In other words, adding a correct prediction, according to ϕ, is worth accumulating a certain number of extra false alarms (1 in this case).

We conclude this section by plotting the detection region for the 8 month model along with the threshold γ. We refer to figure 8.4. Analyzing this model allows to see which crashes the Pearson correlation model(s) missed. Here we see that the 2011 "Bear Market", 2015 "Chinese Selloff", 2018 "Crypto Crash", and the last of the waves associated to the 2008 great recession, i.e. the 2008 "Great Recession II" in January of 2009. However, it correctly predicted the initial 2008

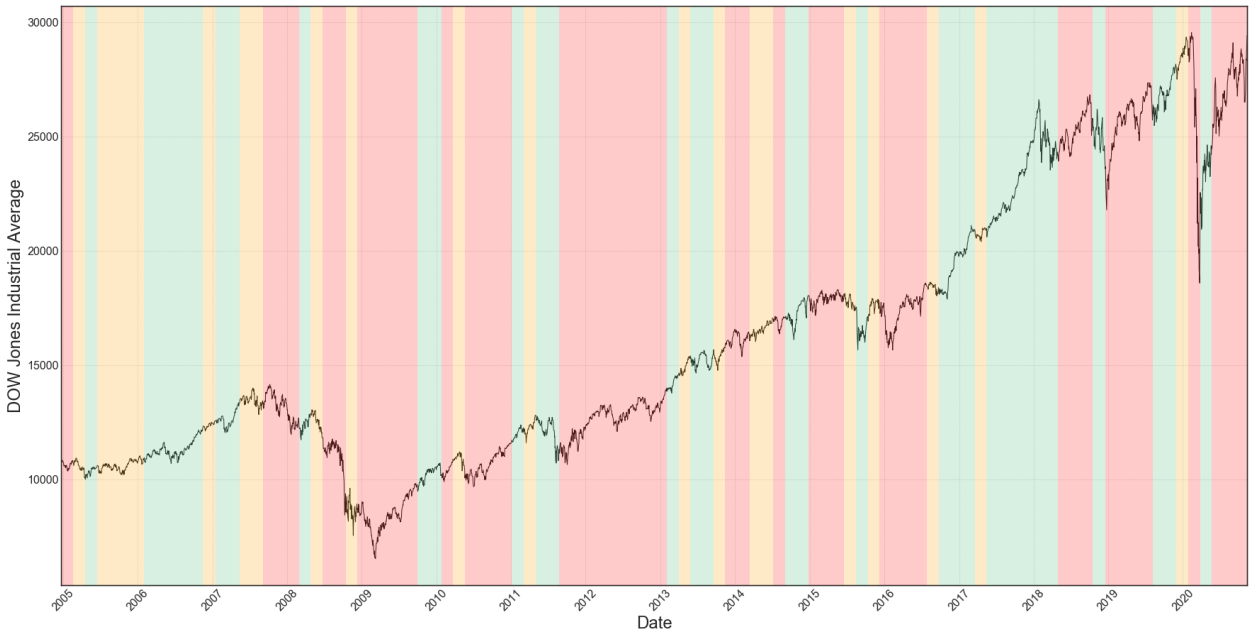


Figure 8.13: Clipped Pearson's ρ: $\vec{Y} \in C^N$

"Financial Crisis", the first 2008 "Great Recession I", and the two extreme drawdowns associated to the 2020 "COVID-19" crashes.

8.3 | RMT Filtering

In this section, we explore the RMT filtering schemes introduced in section 7.4. Remember we omit the basic linear shrinkage estimator $\Xi^{BLS}[\rho]$ in this analysis since this filtering scheme preserves the QLO. In particular, we consider the eigenvalue clipping $\Xi^{CLP}[\rho]$ and $\Xi^{RIE}[\rho]$ schemes. To keep this analysis simple, we only consider the Tail-Dependence and Pearson's ρ models of the previous section.

8.3.1 | Eigenvalue Clipping

Using figure 8.1 from earlier as a reference, we can see that clipping the Pearson's ρ correlation matrices seemed to narrow c_3 coloured bands in the clustering in figure 8.13. Additionally, we see more green coloured marbles c_1 than we did before. The main crashes, the 2008 "Financial Crisis" and 2020 "COVID-19" crashes seem to be covered by the clustering along with the 2015 "Chinese Selloff" and 2018 "Crypto Crash". However, long red bands persist over periods withour

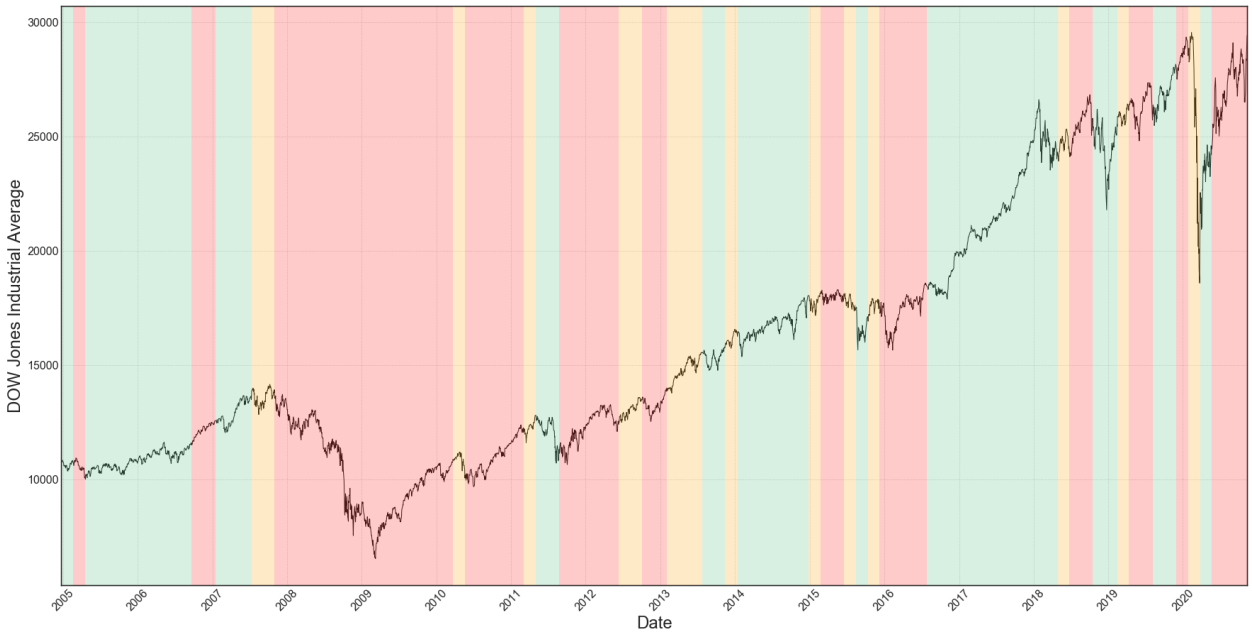


Figure 8.14: Clipped Tail-Dependence λ^{U}: $\vec{Y} \in C^{N}$

crashes, which may be an indication of a few false alarms. We will cover the results in a moment.

Let us turn our attention to figure 8.14. Using figure 8.9 from before as a reference, we notice that clipping the tail dependence matrices created very narrow bands around the 2015 "Chinese Selloff", the 2018 "Crypto Crash", and the 2020 "COVID-19" crashes. which seems to be a good sign. However, we also notice the red band surrounding the 2008 "Financial Crisis" got widened by quite a bit, which may be indication of a lot of false alarms. We see a few transitions between red and yellow, which is also a good sign. All in all, we don't expect the clipped tail dependence matrices to perform better than the raw case. However, Pearson's ρ correlation matrices may actually benefit from the clipping procedure. More on this later. For now, let us look at the results of the two models.

Interestingly enough, our intuition for the performance of the clipped upper tail dependence matrices seems to be on point. Indeed, in any case for $k = 2, 4, 6$, corresponding to 4, 8, and 12 month time horizons, tail dependence is predicting the same number of crashes as before: 7 over 4 months, 8 over 8 months, and 8 over 12 months. Additionally, however, it signalled more false alarms in all three cases, which is what we anticipated. For completeness, We provide the following table, for the curious reader, displaying the optimal parameter set found for Θ^{*}, as before.

On the other hand, clipping Pearson's ρ correlation matrices seemed to significantly improve

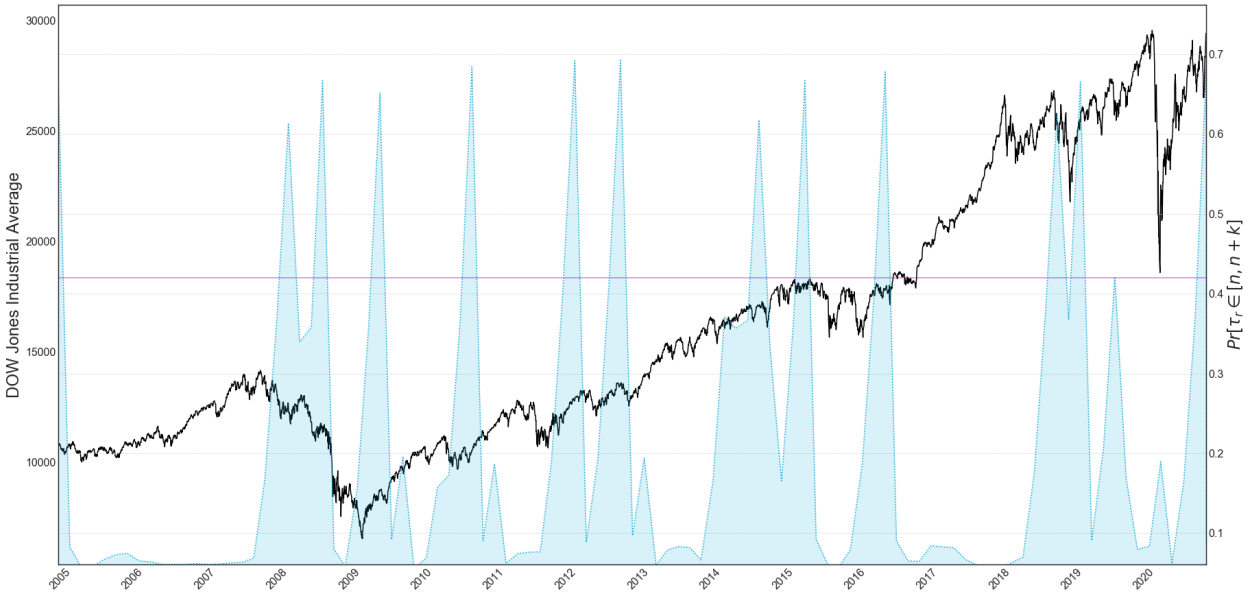


Figure 8.15: Clipped Pearsons ρ Alarm System: Detection Region

Clipped Ξ^{CLP} RUP Performance: Short time horizon						
Operating Characteristics	4-mo. Horizon		8-mo. Horizon		12-mo. Horizon	
	λ^u	ρ	λ^u	ρ	λ^u	ρ
ϕ coefficient	0.40	0.35	0.57	0.57	0.45	0.76
Number of correct Non-alarms	78	94	79	89	82	93
Number of correct alarms	7	3	8	7	8	8
Number of False alarms	20	4	9	1	16	5
Number of undetected Crashes	1	5	0	1	0	0

Table 8.9: RUP Operating characteristics: Clipped

our results on the 12 month time horizon. In fact, this would rival the best model we have trained up to date. However, we notice a sharp drop off in the performance over the 4 and 8 month time horizons. To investigate why this might be the case, let us analyze the detection region corresponding to this 12 month model in figure 8.13. Indeed, it appears as though the clipped Pearson's ρ model may just be getting lucky over this time period. Perhaps the model is simply taking advantage of the relatively longer 12 month time horizon and benefiting from a higher margin of error as it simply signals alarms periodically. This would explain why the models

RUP Performance: Short time horizon						
Θ^*	4-mo. Horizon		8-mo. Horizon		12-mo. Horizon	
	λ^u	ρ	λ^u	ρ	λ^u	ρ
γ	0.47	0.46	0.39	0.31	0.47	0.42
$(q, \mathcal{L})$	$(0, 2)$	$(0, 5)$	$(0, 3)$	$(0, 4)$	$(0, 3)$	$(0, 4)$
$(\delta_{2,3}, \delta)$	$(2, 9)$	$(2, 7)$	$(2, 9)$	$(2, 7)$	$(2, 9)$	$(4, 7)$

Table 8.10: Eigenvalue clipping: Hyper-parameters

performance drops off significantly over shorter time horizons.

8.3.2 | Rotationally Invariant Estimator

As before, we plot the resulting clusterings for the rotationally invariant estimator acted upon $\mathrm{P}_{\Delta}(\mathcal{S}; \mathrm{I})$ and $\Lambda^{u}_{\Delta}(\mathcal{S}; \mathrm{I})$. In figure 8.16 we see a dramatic change in the associated clustering for Pearson's ρ. Indeed, it appears that the rotationally invariant estimator corrupted the QMG corresponding to the $\mathrm{P}_{\Delta}(\mathcal{S}; \mathrm{I})$ temporal state space of correlation matrices. In this plot, we notice some red around known crashes, but this correspondence seems to be completely random. Thus, we'd be tempted to say that the associated alarm system will have a low ϕ coefficient, even if it correctly predicts several crashes.

Turning our attention to Tail-Dependence, under the rotationally invariant estimator, we see some familiar behavior. Comparing figure 8.17 to that of the clustering for the raw Tail-Dependence matrices of figure 8.9, we notice that the corresponding QMG seems to have been corrupted as well. Not only have the dense regions of red, surrounding the 2008 financial crisis, been widened, but it seems the 2015, 2018, and 2020 COVID-19 crashes have been disregarded.

Out of curiosity, we trained and tested the associated alarm systems and display the results in the following table. For the sake of simplicity, we ignore the hyper-parameterization analysis. We also only display the results over 4, 8, and 12 month time horizons. We immediately notice that the corresponding ϕ coefficient is much lower in any case as compared to the raw instances from the prior section. With the exception of the 12 month time horizon, the majority of crashes are going undetected in addition to the signalling of false alarms.

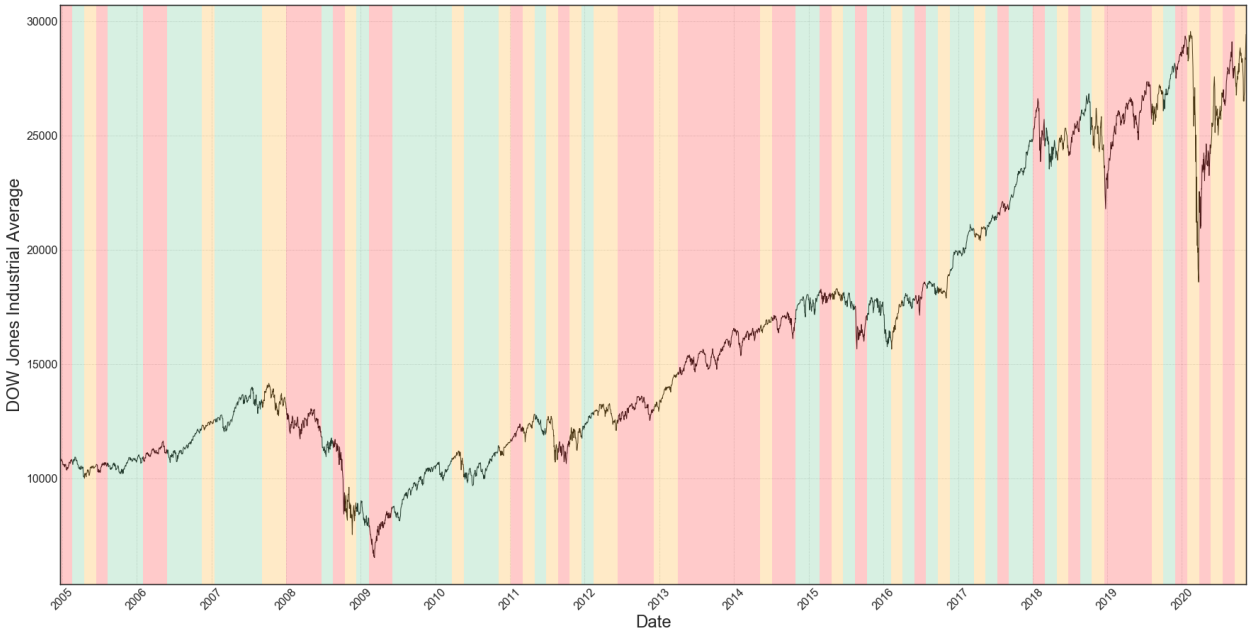


Figure 8.16: RIE Pearson's ρ: $\vec{Y} \in C^N$

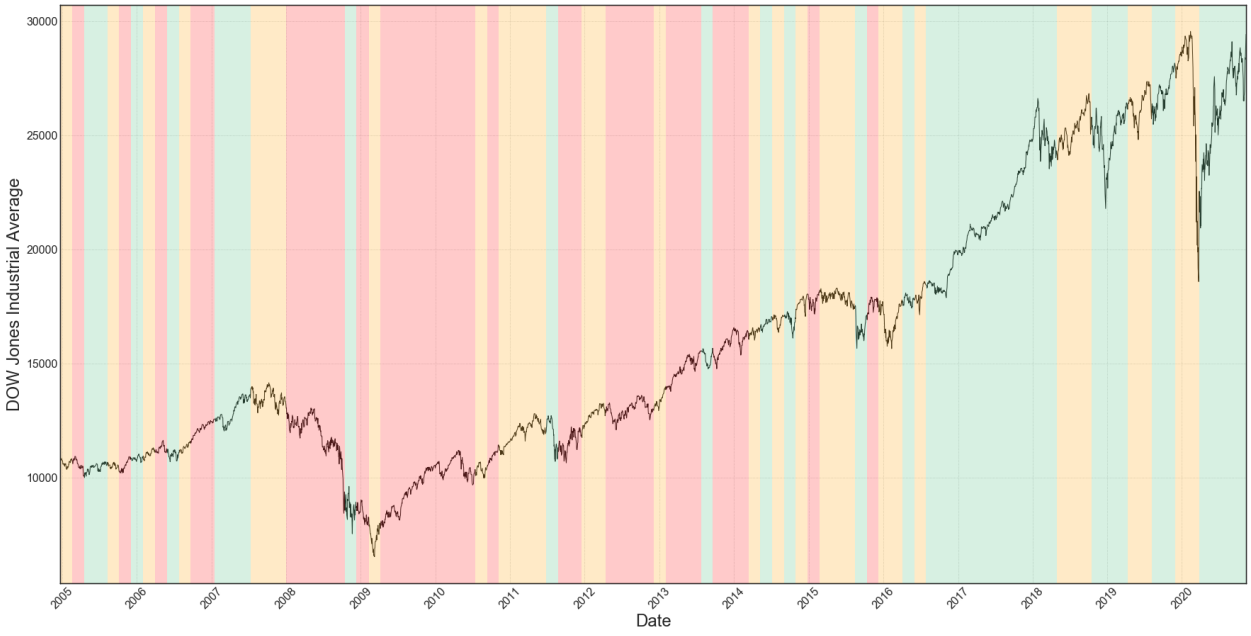


Figure 8.17: RIE Tail-Dependence λ^U: $\vec{Y} \in C^N$

Nonlinear Shrinkage Ξ^{RIE} RUP Performance						
Operating Characteristics	4-mo. Horizon		8-mo. Horizon		12-mo. Horizon	
	λ^u	ρ	λ^u	ρ	λ^u	ρ
ϕ coefficient	0.12	0.11	0.20	0.25	0.23	0.34
Number of correct Non-alarms	81	79	72	77	75	79
Number of correct alarms	4	3	5	5	5	6
Number of False alarms	17	19	26	21	23	19
Number of undetected Crashes	4	5	5	3	3	2

Table 8.11: Operating characteristics for the RIE cleaned correlation matrices

8.3.3 | RMT Takeaways

Perhaps the results of this section should not have come at a surprise. Since RMT filtering schemes filter out measurement error by distorting the (quantum) Lorenz curve closer to the uniform distribution, correlation matrices become less risky. From a portfolio optimization standpoint, we want this behavior to make the optimal weightings more conservative. From the perspective of the QMG (quantum majorization graph), however, cleaning the correlation matrices (in a sense) seems to "hide" the true exposure to systemic risk from the QLO (quantum Lorenz ordering). Hence, the QLO gets confused and makes the QMG noisier, corrupting the clustering, and, in turn, harming our alarm system.

Additionally, we saw that eigenvalue clipping performed better than the rotationally invariant estimator, which makes sense. Indeed, the majorization of eigenvalue clipping essentially corresponds to the majorization of a sub full rank spectrum. Hence, in eigenvalue clipping we're only losing partial information in the least informative region of the spectrum. On the other hand, the rotationally invariant estimator acts on the entire spectrum distorting the entire spectrum along with the QLO's perception of systemic risk.

All in all, we believe that these RMT results can be thought of as, perhaps ironically, a verification. Suppose for a moment that these RMT schemes made the alarm system better. This might mean that their corresponding portfolios would be reflecting the amount of risk in the market at the time, instead of dispersing that risk over future time horizons which would make those portfolios that much more volatile. Hence, the fact that RMT cleaned QLO's do not know how risky the market actually is only verifies their methodology in studying portfolio risk. This is just

a theory and needs to be explored further.

8.4 | RUP vs. LPPLS

In this section we revisit the confidence indicators (CI) of the LPPLS model introduced in chapter 5. Recall that a (positive) CI is a measure of confidence that the LPPLS fitting is signalling an LPPLS pattern, corresponding to a "bubble" in the price of, in our case, the DJIA. These CI's are a measure $CI \in [0, 1]$ that can be thought of as a probability that a crash is on the horizon. Hence, we can consider these CI's in analogy to the stopping-time probabilities

$$\mathbb{P}[\tau_r \in [n, n + k]|\tau_1, \ldots, \tau_{r-1}] \tag{8.6}$$

produced by the process $\mathcal{Z} \in \mathcal{RUP}(S, C, U, d)$ at each iteration of the RUP's evolution.

To this end, we plot the stopping time probabilities for our RUP, that we will henceforth refer to as simply the RUP's detection region, against the CI's of the LLPLS model in figure 8.18. This plot corresponds to the testing period 2005-2020 that we've been considering thus far. Furthermore, the RUP detection region depicted in this figure corresponds to that of the alarm system associated to the upper Tail-Dependence temporal state space $\Lambda^{U}_{\Delta}(\mathcal{S}; I)$, from before, scaled up by a factor of 10 to make the plot more readable. Interestingly, the RUP detection region seems to be accumulating around known stock market crashes more accurately than the CI's of the LPPLS model. For instance, between 2006-2008 the CI's are sending strong signals, as is the case for the 2013-2015 and 2016-2017 time periods. Over these same time periods, meanwhile, the RUP detection region subsides to its relative minimum.

We trained a similar alarm system for stock market crashes using the CI's over the same time period as the RUP (1986-2005) in a similar fashion as the RUP. Clearly the CI's don't depend on the $(q, \mathcal{L})$ and $(\delta_{2,3}, \delta)$ parameters and only depends on the detection region threshold γ. Thus, we vary γ over the same range as we did the RUP, computing the ϕ coefficient over the same 4, 8, and 12 month time horizons. We depict this optimization scheme in figure 8.19, corresponding to the 12 month time horizon, where ϕ, and the nuber of false alarms FP, are plotted as a function of γ over the $\gamma \in [0, 1.0]$ range.

On the right vertical axis, notice the scale of FP for the LPPLS model. The sheer number of FP's is simply due to the fact that the LPPLS model is signalling alarms every day, whereas our RUP is signalling alarms every 2 months. As γ increases, however, the number of FP's quickly

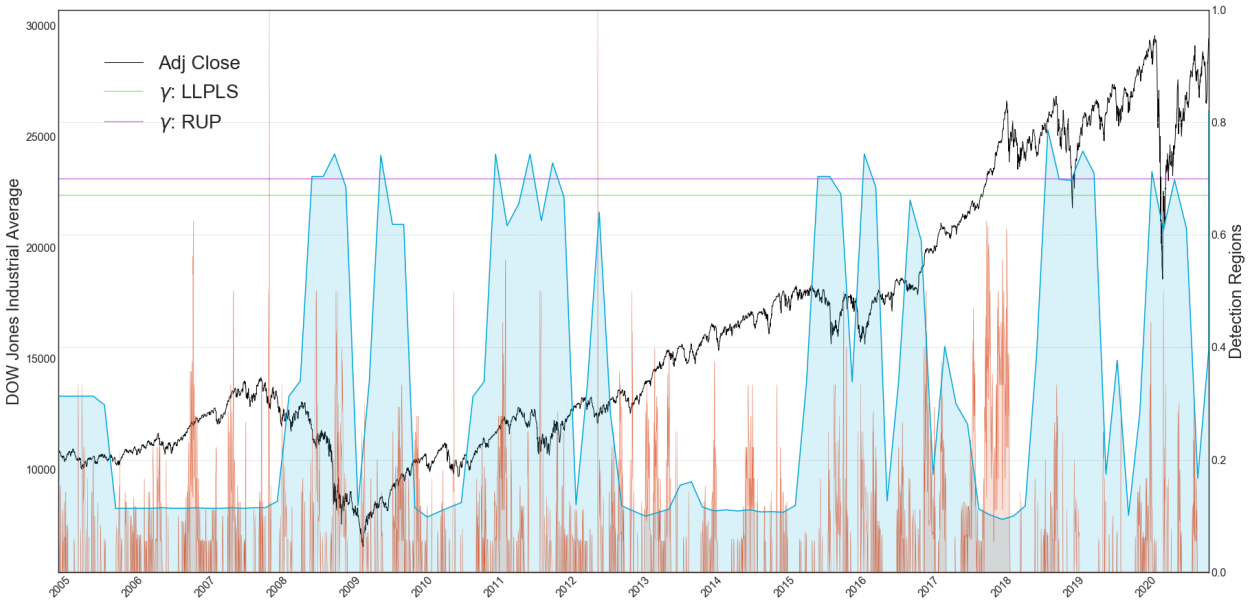


Figure 8.18: LLPLS vs. RUP Detection Region

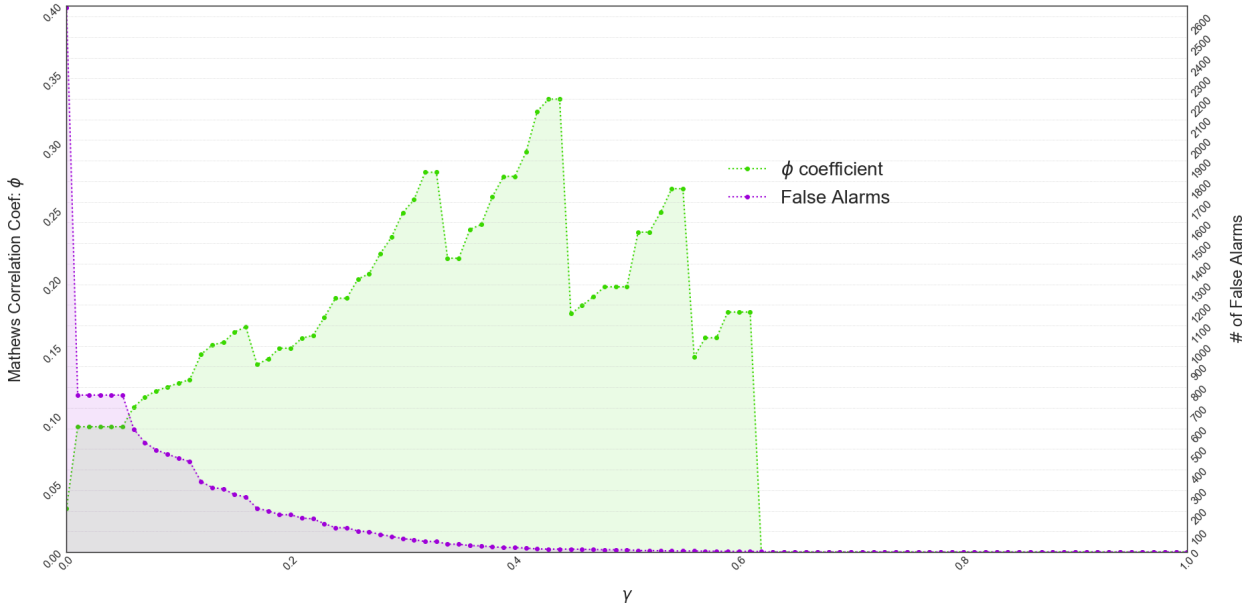


Figure 8.19: LLPLS ϕ coefficient

drops to a number FP $= 14$ where ϕ achieved its maximum at $\phi^* = 0.33$. The corresponding γ is plotted in 8.18 as the green horizontal line. Referring to table 8.12, we display the resulting oper-

RUP (Tail-Dependence) vs. LPPLS						
Operating Characteristics	4-mo. Horizon		8-mo. Horizon		12-mo. Horizon	
	λ^u	LPPLS	λ^u	LPPLS	λ^u	LPPLS
ϕ coefficient	0.40	0.11	0.57	0.18	0.76	0.33
Number of correct Non-alarms	89	3976	91	3970	93	3984
Number of correct alarms	7	2	8	3	8	4
Number of False alarms	9	31	7	28	5	14
Number of undetected Crashes	1	6	0	5	0	4

Table 8.12: LPPLS vs. RUP: Operating characteristics

ating characteristics of the alarm systems corresponding to the Tail-Depndence RUP and the CI's of the LPPLS model. We performed the same optimization scheme over the 4, 8, and 12 month time horizons. The operating characteristics, and ϕ coefficient, seem to favor the RUP model. Indeed, for every time horizon, the RUP is not only accurately predicting more crashes, but also signalling fewer FP's (false alarms). However, the fact that the RUP is signalling fewer FP's can be read as a consequence of the relative frequencies of the two alarm systems. On the other hand, the fact that the RUP is accurately predicting more crashes is a promising sign. Drawing a stronger conclusion would be unfair to the LPPLS model.

IV
Conclusion

9 Conclusion

9.1 That's a Wrap

In this thesis, we applied the quantum majorization of financial correlation matrices to the prediction of stock market crashes via the reinforced urn process (RUP). This was accomplished by using a directed acyclic graph to represent the partial ordering induced by quantum majorization, which was aptly named the quantum majorization graph, to cluster each time period associated to the correlation matrices according to the in-out degree centrality if its corresponding node in the graph. These clusters were used to represent coloured marbles in an urn to build a reinforced urn process to predict stock market crashes. Naturally, several sub-problems came into view including the precise definition of a stock market crash, which correlation metric to use.

9.1.1 Portfolio Risk and Random Matrix Theory

In prior work, quantum majorization was considered from standpoint of socioeconomic variability. In chapter 7 we considered quantum majorization from the standpoint of measurement error. In this vein, several random matrix theory (RMT) "filtering" schemes have been devised to rid correlation matrices of measurement noise. We showed that these filtering schemes bore interesting relationships with the quantum majorization of said correlation matrices and, in some cases, even the quantum majorization graph. The RUP stock market crash prediction model was permuted over these cleaned correlation matrices.

9.1.2 | Stock Market Crashes

In chapter 6 we exploited a novel approach to defining stock market crashes using extreme value theory. This was accomplished by fitting the distribution of the so-called ε-drawdowns of the Dow Jones Industrial average (DJIA) to the generalized pareto distribution (GPD). We used the shape γ of the GPD in tandem with the maximum-to-sum (MS) plot to identify the threshold that characterizes the tail of the GPD. Any ε-drawdown beyond this threshold was identified as a crash. Such ε-drawdowns included, but were not limited to, the 1987 "Black Monday", 2000 "Tech Bubble", 2008 "Financial Crisis", and 2020 "COVID-19" crashes. Additionally, we employed an ad hoc procedure to cluster crashes that occurred very close to eachother, as was the case for the COVID-19 crash that would otherwise not be identified by the currently utilized approach of fitting the stretched exponential distribution. In total, 13 crashes were identified for which the RUP was tasked to predict.

9.1.3 | RUP Model and Alarm System

In chapter 7 we designed an algorithm for the sampling and clustering of financial correlation matrices that represented the drawing of marbles out of a state space of polya urns. This algorithm defined an online learning algorithm that belongs to the class of Bayesian non-parametric models. At each time step the probability of a crash occurring within (several) time horizons was computed where an "alarm" was cast when this probability exceeded a specified threshold defining the detection region. In the RUP specification, a crash corresponded to when the RUP reached a specific terminal state called the stopping rule with associated stopping time. We generalized this stopping rule so as to place the stricter criteria of sampling a q number of risky correlation matrices consecutively. This was one of three generalizations that we considered including accounting for a dependence structure between the sampling, and reinforcement, of yellow and red marbles along with the calculation of the stopping rule's stopping times over a time interval. This generalization helped not only improve the model but also provide increased financial interpret-ability.

9.2 | Results and Takeaways

In chapter 8 we showed that the RUP was able to predict stock market crashes with good accuracy. Of the three different correlation metrics we considered, the so-called measure of Tail-

Dependence λ^{U} performed the best predicting 100% of crashes over the testing period whilst signalling just 5 false alarms over a 12 month time horizon and 7 false alarms over an 8 month time horizon. On the shorter time horizons of 2 and 4 months Tail-Dependence predicted 7/8 crashes with 17 and 9 false alarms, respectively.

The stronger performance over longer time horizons showed that the RUP is performing in correspondence with human intuition, which is a promising sign for the efficacy of the proposed RUP methodology. Additionally, though Tail-Dependence performed best, the other two metrics Gini's γ^{G} and Pearson's ρ performed well also, showing that the RUP is robust to the choice of correlation metric which is another promising sign. Furthermore, the two generalizations we made, concerning the reinforcement mechanism and stopping rule, were shown to dramatically improve the performance of the model.

We also applied the aforementioned RMT filtering schemes to our correlation matrices to see if they would improve our model. Surprisingly, we didn't see much improvement which might suggest that RMT filtering was actually (ironically) doing its job by reducing the RUP's perception of risk exposure. In particular, we showed that basic linear shrinkage preserves the QLO, whereas clipping makes the ordering denser. We did not conclude how nonlinear shrinkage affects the ordering but provided some intuition as to how it might.

Lastly, we compared our RUP methodology to a well known stock market crash (bubble) detector: the Log-Periodic Power Law Singularity (LPPLS) model. We computed the LPPLS's associated so-called "confidence indicators" (CI) as an analog to the RUP's detection regions. We showed that the RUP was able to predict more crashes than the LLPLS model and overall performed better with respect to several performance metrics defined on the confusion matrix including the ϕ correlation coefficient, the risk-ration (RR), and F1 score. We failed to draw a more conclusive finding as the LPPLS model and RUP make predictions with different frequencies.

9.3 | Future Work

9.3.1 | Data and Markets

An immediate improvement can be made by accessing paid financial data repositories and gathering complete data of historical financial index constituent memberships. For instance, one can get all 500 components of the S&P500 so as to avoid the "survivorship" bias of chapter 6. One can

then repeat these results over any financial index including the FTSE, CAC40, DAX30, or even the NIKKEI225, HSI, and STI in the asian markets. Avoiding survivorship bias will make the RUP more reflective of the index its trying to make predictions over.

9.3.2 | RUP Modelling and Alarm System

As for the RUP itself, several additional considerations can be made. For instance, the reinforcement matrix, which is generalized in this work, can be generalized further to account for time-dependence and even randomness itself. For instance, one can incorporate seasonality into the strength, dependence, and even randomness into the reinforcement matrix (RM)

$$\mathrm{RM} = \begin{array}{c} \\ C_1 \\ C_2 \\ C_3 \end{array} \begin{array}{c} \begin{array}{ccc} C_1 & C_2 & C_3 \end{array} \\ \begin{pmatrix} \delta_{1,1}(t;\omega) & \delta_{1,2}(t;\omega) & \delta_{1,3}(t;\omega) \\ 0 & \delta_{2,2}(t;\omega) & \delta_{2,3}(t;\omega) \\ 0 & 0 & \delta_{3,3}(t;\omega) \end{pmatrix} \end{array}, \qquad s_i = \sum_{j=1}^{3} \delta_{i,j}(t;\omega) = \delta, \qquad \forall i \in \{1,2,3\} \tag{9.1}$$

where the random variable ω can be taken such that the $\delta_{i,j}$'s are Dirichlet distributed so as to maintain the balance nature of the RM.

As for predicting stock market crashes, we saw that our RUP in this work was signalling more false alarms over shorter time horizons, which was to be expected. However, in practice, this may not be a drawback. For instance, one can incorporate rewards drawn from the payoffs of american (or european) options as part of the learning procedure. Indeed, out of the money put options over shorter time horizons are much cheaper. In which case, because of convexity and the non-linearity of options payoffs, the notion of an alarm systems optimality in a financial context might have a higher tolerance for false alarm signalling over shorter time horizons. In fact, an alarm system that signals a lot of false alarms over the 2 month interval might produce a more profitable trading strategy provided they detect a sufficient number of those crashes. It would be interesting to see if buying out of the money put options (american or european) beyond the EVT threshold would yield a profitable trading strategy.

This provides a natural lead into Reinforcement Learning (RL). Indeed, the RUP is essentially a very simple (yet ingenious) rienforcement learning algorithm. The topic of Q-learning is extensively discussed in, for example, Jang et al. [2019]. The urn compositions of the RUP can be analogized to the state transition probabilities of Q-learning and RL algorithms in general. The last piece is introducing some reward for which Q-learning can learn from. One can incorporate

the put option strategy just discussed to train a RL model to compare against the RUP.

Following the RL line of reasoning, one can move into the framework of geometric deep learning, or graph neural networks, to learn the dynamics of the evolution of the quantum majorization graph. A study of graph neural networks can be found in Zhou et al. [2020]. To date, models for dynamically growing graph neural networks are seldom but shouldn't be too difficult to generalize into. In which case, one can combine the RUP, RL, and the deep geometric learning of the quantum majorization graph to predict the dynamics of systemic market risk.

9.3.3 | Compositional Data Analysis

One can also consider the so-called compositional data analysis of Aitchison [1982]. The central object of interest of this work is the (probability) distribution of a correlation matrices eigenvalues. The temporal evolution of the quantum majorization graph is simply a consequence of the stochastic behavior of the correlation matrices eigenvalues. Since the spectrum is a distribution, if one wanted to study its statistical properties, we'd be studying a distribution of distributions for the correlation matrices spectrum. Following Aitchison [1982], one can use the inverse mapping of the softmax function to map the eigenvalues onto $\mathbb{R}^d$ for which any multivariate distribution can be fitted to define a distribution on the simplex. Learning the statistical properties of the correlation matrices eigenvalues would certainly be interesting. For instance, Karrer and Newman [2009] propose a random graph model for directed acyclic networks. Understanding the statistics of our correlation matrices spectra might allow us to predict the dynamics of a correlation matrix since the probability of majorization is obviously related to how dispersed one spectrum is.

9.3.4 | Quantum Thermo-Majorization and D-Majorization

The last suggestion we give for future work concerns the generalization of quantum majorizaton into the so-called thermo-mjorization and D-majorization. Given two probability distributions $p = \{p_1, \dots, p_d\}$ and $q = \{q_1, \dots, q_d\}$ one can define the so-called relative Lorenz curve $L_\alpha(p, q)$ by permuting the elements of p and q where $p_\pi = \{p_{\pi_{(1)}}, \dots, p_{\pi_{(d)}}\}$ and $q_\pi = \{q_{\pi_{(1)}}, \dots, q_{\pi_{(d)}}\}$ such that

$$\frac{p_{\pi_{(1)}}}{q_{\pi_{(1)}}} \geqslant \frac{p_{\pi_{(2)}}}{q_{\pi_{(2)}}} \geqslant \cdots \geqslant \frac{p_{\pi_{(d)}}}{q_{\pi_{(d)}}}.$$

where $L_\alpha(p, q)$ defines the CDF of the uniform distribution , the diagonal line on the $[0, 1] \times [0, 1]$ unit square, when $p = q$. In view of the RMT filtering schemes, we propose one can consider

$p = \lambda(\rho)$ the eigenvalues of a correlation matrix, and $q = \Omega$ the portfolio weights found as a result of MPT optimization. As discussed in Bouchaud and Potters [2009], if MPT allocates large weights to small eigenvalues, then the relative Lorenz curve $L_\alpha(\lambda(\rho), \Omega)$ would be highly distorted as opposed to being uniform if the weights are proportional to the eigenvalues $\lambda(\rho)$. Perhaps the Gini index $\mathcal{G}[L_\alpha(\lambda(\rho), \Omega)]$ of Gini [1912], and discussed extensively in Fontanari et al. [2019], can be used as a measure of out-of-sample risk, to whatever extent one believes their correlation matrix has been corrupted by meaurement noise.

In view of the connection between quantum majorization and RMT filtering, it might make sense for the relative Lorenz curve $L_\alpha(\lambda(\rho), \Omega)$ to majorize the relative Lorenz curve $L_\alpha(\xi(\rho), \Omega_\xi)$ between the "cleaned" eigenvalues $\xi(\rho)$ and resulting portfolio weights Ω_ξ since RMT filtering is meant to increase the size of small eigenvalues and decrease the weights assigned to them. Such a majorization condition $L_\alpha(\lambda(\rho), \Omega) \succ L_\alpha(\xi(\rho), \Omega_\xi)$ is known as "D-majorization" and is extensively analyzed by Sagawa [2020]. Such a D-majorization condition could be posed as a necessary condition for the efficacy of any RMT filtering scheme. This might be an interesting line of research.

Additionally, the quantum majorization condition considered in this work $L_\alpha(\rho) \succ L_\alpha(\rho^*)$ can be generalized to the so-called quantum thermo-majorization condition via the quantum relative Lorenz curves $L_\alpha(\rho, I) \succ L_\alpha(\rho^*, I)$ where the eigenvalues of ρ, ρ^*, and I are permuted as per the equation of the relative Lorenz curve above where $L_\alpha(\rho) = L_\alpha(\rho, I)$. This has an interesting connection with the second law of thermodynamics extensively discussed in Sagawa [2020] where quantum thermo-majorization is taken with respect to a "reference" distribution, or Gibbs state, ρ^G via the majorization condition $L_\alpha(\rho, \rho^G) \succ L_\alpha(\rho^*, \rho^G)$. In future work, one can repeat the work conducted in this thesis by using an appropriate Gibbs state ρ^G. This approach amounts to specifying a system Hamiltonian which would be an interesting line of future research.

Bibliography

J. Aitchison. The statistical analysis of compositional data. *Journal of the Royal Statistical Society. Series B (Methodological)*, 44(2):139–177, 1982. ISSN 00359246.

J. H. Allan Gut. Extreme shock models. *Extremes*, 2:295–307, 1999. doi: 10.1023/A:1009959004020.

O. L. Anders Johansen and D. Sornette. Crashes as critical points. *International Journal of Theoretical and Applied Finance*, 3, 2006.

A.-T. M. Antunes M and T. FK. A bayesian approach to event prediction. *Time Series Annals*, 24: 631–646, 2003. doi: 10.1007/s11009-012-9281-z.

B. C. Arnold and J. M. Sarabia. *Majorization and the Lorenz Order with Applications in Applied Mathematics and Economics*. Statistics for Social and Behavioral Sciences. Springer, 2018. ISBN 978-3-319-93772-4.

J. P. Bouchaud and M. Potters. Financial applications of random matrix theory: a short review. 2009.

J. Bun, R. Allez, J. Bouchaud, and M. Potters. Rotational invariant estimator for general noisy matrices. *IEEE Transactions on Information Theory*, 62(12):7475–7490, 2016. doi: 10.1109/TIT.2016.2616132.

J. Bun, R. Allez, J. Bouchaud, and M. Potters. Cleaning large correlation matrices: Tools from random matrix theory. *Physics Reports*, 666:1–109, 2017. ISSN 0370-1573. doi: 10.1016/j.physrep.2016.10.005.

P. Cirillo and J. Husler. An urn-based approach to generalized extreme shock models. *Statistics and Probability Letters*, 79:969–976, 2009. doi: 10.1007/s11009-012-9281-z.

P. Cirillo and J. Hüsler. Extreme shock models: An alternative perspective. *Statistics Probability Letters*, 81(1):25–30, Jan 2011. ISSN 0167-7152. doi: 10.1016/j.spl.2010.09.014. URL http://dx.doi.org/10.1016/j.spl.2010.09.014.

J. de Mare. Optimal prediction of catastrophes with applications to gaussian processes. *Annals of Probability*, 8:841–850, 1980. doi: 10.1007/s11009-012-9281-z.

F. D. Diaconis P. de finetti's theorem for markov chains. *Annals of Probability*, 8:115–130, 1980.

J. J. Didier Sornette. Endogenous versus exogenous crashes in financial markets. *Brussels Economic Review*, 49, 2006.

R. Duchin and H. Levy. Markowitz versus the talmudic portfolio diversification strategies. *The Journal of Portfolio Management*, 35:71–74, 12 2009. doi: 10.3905/JPM.2009.35.2.071.

A. Fontanari, P. Cirillo, K. Oosterlee, and I. Eliazar. Portfolio risk and the quantum majorization of financial correlation matrices. *SSRN*, 2019. doi: 10.2139/ssrn.3309585.

C. Gini. Variabilità e mutabilità. *Memorie di metodologica statistica(Ed. Pizetti E, Salvemini, T)*, 1912.

M. HM. *Polya Urn Models*. 2009.

B. Jang, M. Kim, G. Harerimana, and J. W. Kim. Q-learning algorithms: A comprehensive classification and applications. *IEEE Access*, 7:133653–133667, 2019. doi: 10.1109/ACCESS.2019.2941229.

B. Karrer and M. E. J. Newman. Random graph models for directed acyclic networks. *Physical Review E*, 80(4), Oct 2009. ISSN 1550-2376. doi: 10.1103/physreve.80.046110. URL http://dx.doi.org/10.1103/PhysRevE.80.046110.

M. G. KENDALL. A NEW MEASURE OF RANK CORRELATION. *Biometrika*, 30(1-2):81–93, 06 1938. ISSN 0006-3444. doi: 10.1093/biomet/30.1-2.81. URL https://doi.org/10.1093/biomet/30.1-2.81.

M. O. Lorenz. Methods of measuring the concentration of wealth. *Publications of the American Statistical Association*, 9(70):209–219, 1905. ISSN 15225437. doi: 10.2307/2276207.

V. A. Marčenko and L. A. Pastur. DISTRIBUTION OF EIGENVALUES FOR SOME SETS OF RANDOM MATRICES. *Mathematics of the USSR-Sbornik*, 1(4):457–483, apr 1967. doi: 10.1070/sm1967v001n04abeh001994. URL https://doi.org/10.1070%2Fsm1967v001n04abeh001994.

H. Markowitz. Portfolio selection. *The Journal of Finance*, 7(1):77–91, 1952. ISSN 00221082, 15406261. doi: 10.2307/2975974,. URL http://www.jstor.org/stable/2975974.

M. A. Nielsen and G. Vidal. Majorization and the interconversion of bipartite states. *Quantum Info. Comput.*, 1(1):76–93, Jan. 2001. ISSN 1533-7146.

S. W. P. Muliere, P. Secchi. Urn schemes and reinforced random walks. *Applications of Stochastic Processes*, 28:59–78, 2000.

S. Peluso, A. Mira, and P. Muliere. Reinforced urn processes for credit risk models. *Journal of Econometrics*, 184(1):1–12, 2015. ISSN 0304-4076. doi: https://doi.org/10.1016/j.jeconom.2014.08.003. URL https://www.sciencedirect.com/science/article/pii/S0304407614001791.

T. Sagawa. Entropy, divergence, and majorization in classical and quantum thermodynamics, 2020.

Y. Sang, X. Dang, and H. Sang. Symmetric gini covariance and correlation, 2016.

C. Vanderford, Y. Sang, and X. Dang. Two symmetric and computationally efficient gini correlations. *Dependence Modeling*, 8(1):373–395, 2020. doi: doi:10.1515/demo-2020-0020. URL https://doi.org/10.1515/demo-2020-0020.

J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun. Graph neural networks: A review of methods and applications. *AI Open*, 1:57–81, 2020. ISSN 2666-6510. doi: https://doi.org/10.1016/j.aiopen.2021.01.001. URL https://www.sciencedirect.com/science/article/pii/S2666651021000012.