A financial cure for the pandemic? Dissecting the World Bank's emergency financing facility

by

Xiyuan Zheng

Graduate Program in Statistical and Actuarial Sciences

School of Graduate and Postdoctoral Studies
The University of Western Ontario
London, Ontario, Canada

Acknowledgements

I would like to express my deepest appreciation to Dr. Rogemar Mamon for the kind guidance and extraordinary support he provided throughout this project.

Abstract

The pandemic bond issued by the World Bank (WB) in 2017 is a financial innovation, enabling the transfer of pandemic risk in the underdeveloped countries to the financial market. It covers various perils of diseases that could overwhelm the regional or global public-health systems and adversely impact the economy. If all of the six parametric triggers are activated - some of them are dependent on each other - the bond's principal and coupons will be used to finance coordinated, swift and resilient medical response to safeguard the well-being of the populace and to provide monetary support for affected businesses and households especially to those who are most vulnerable. However, this product is criticised for its onerous trigger requirements. In this research, the WB's pandemic bond-pricing framework, which requires inputs that are only partially available, is examined. From a rather unstructured COVID-19 data set, we underscore a major contribution of organising an information database customised to facilitate the pandemic-bond valuation. The pricing is analogous to that of a non-defaultable bond but with trigger-risk adjustments and recovery rate. A multivariate time series model in conjunction with the Monte-Carlo simulation is utilised to capture the triggers' underlying variables along with the trigger risk's quantification. We use a LIBOR-based discount factor and bootstrapping to address parameter estimation based on data that may have some reliability issue. Our findings show that the bond value is particularly sensitive to both the conditional probability of a pandemic happening as well as the unknown and random total number of deaths in covered areas.

Keywords: Vector autoregressive moving average model, catastrophe bond, trigger events, floating-interest rate

Contents

List of Figures

List of Tables

List of Appendices

Chapter 1

Introduction

According to the World Economic Forum (WEF), global risks includes the high likelihood of natural disasters; many of these disasters are catastrophic in magnitude (Harrison & Williams, 2016). The WEF highlighted the top 10 major risks: inter-state conflict, extreme weather events, failure of national governance, state collapse or crisis, unemployment or underemployment, natural catastrophes, failure of climate change adaptation, water crises, data fraud or theft and cyber attacks. Those directly affected by almost all of these risks live in extremely poor regions and countries. Without global assistance, it is foreseen that the costs of natural disaster mitigation are unsupportable.

At the 2015 Paris conference on climate change - tackling extreme weather events, failure of climate-change adaptation, natural catastrophes, and water crises - an agreement was struck that allocates an annual budget of USD 100 billion to fund interventions related to climate change. Such fund allotment must be spent wisely and reliance on the capability of science and technology to produce fact-based risk assessment that will be critical in coming up with the most effective intervention. As noted by Swiss Re, however, some regions may become uninsurable in the near future. Thus, the creation of new financial innovations that can address and mitigate immediately the harmful effects of disasters is necessary. To this end, catastrophe (CAT) bonds came into being as in the aftermath of Northridge earthquake and Hurricane Andrew.

CAT bonds are financial securities that transfer a particular set of risks from a sponsor to investors. The term “sponsor" here is synonymous to risk carriers, typically an insurance company; but in some cases, this may be a country or regional government. The investors in the global financial markets then absorb the risk of losses from a catastrophe via the mechanism provided by a CAT bond. If specified conditions (e.g., linked to attributes of natural disaster or other catastrophes) are met, the bond’s cash flow are used to cover the cost of the sponsor, and the obligation to repay the interest and principal is waived. Otherwise, it generates steady interest flows and repays the principal to investors just like other debt instruments. To avoid conflict of interest, the process of distributing payments to investors or payouts to sponsors is executed through a special purpose vehicle.

A couple of advantages that CAT bonds could bring are: (*i*) transfer of an insurer’s insolvency

risk to capital market when a catastrophe occurs (Hofer, Gardoni, & Zanini, 2019) and (*ii*) provision of an effective diversification and potential for generation of stable returns in an investors' portfolio (Mariani & Amoruso, 2016).

In the wake of the Ebola crisis from 2014-2016, the World Bank (WB) launched the Pandemic Emergency Financing Facility (PEFF) whose primary goal is to pay out a sponsor (i.e., country's government) as soon as a threshold or a set of thresholds is triggered. In this sense, the catastrophe underlying the bond is a pandemic. Countries can then use the payout money to finance anything that will alleviate the situation for a disease outbreak from spreading. In other words, if a pandemic were to happen before the bond matures, the invested cash will go towards initiatives in developing countries to fighting off a pandemic. If there no pandemic on or before the bond's maturity, the investors will receive the returns on their investment, which is higher than the regular bonds and other financial instruments. Through the PEFF, $320 million worth of bonds was raised going into a collective pool. Although the designs of CAT bond-payout conditions are coached in stringent parametric constraints or indices' triggers as in the case of the WB's pandemic bond, the trigger designs also have a positive effect of countering the moral hazard risk; see (Doherty, 1997) and (Niehaus & Mann, 1992).

There are two components that constitute the World Bank's PEFF: an insurance window and a cash window. The insurance window, namely "the pandemic bond", is a catastrophe bond for which the financial-market participants invested in. It pays monthly floating-rate coupons to investors for covering the pandemic risk in those developing countries. Once a pandemic happens and the pre-defined parametric triggers are satisfied, then the bond's principal is used as funding sources distributed to all eligible countries. The cash window, on the other hand, complements the insurance window. The former possesses a fast financial injection mechanism that is unavailable in the latter. For example, when there is a epidemic outbreak which is not covered by the the pandemic bond, the PEFF would use the funds donated by the steering countries (Germany and Japan) with their approval of course.

In 2020, the coronavirus swept around the globe, and COVID-19 became the first ever outbreak that activated the pandemic bond's payout mechanism since its issuance in 2017. On 17 April 2020, WB declared that the criteria for the payout were satisfied on 31 March 2020. A maximum payout amount for coronavirus of 195.84 million US dollars will be distributed to 64 countries that has the lowest income in the world.

Despite the good intentions of WB's pandemic bond, it was heavily criticised at the height of the COVID-19 pandemic. The disapprobation stems from the fact that it provides generous returns to investors whilst access to funding is too difficult, especially during the early stages of the outbreak when immediate countermeasures are critical. The payout timing of the WB's pandemic bond happened almost three and half months since the start of the COVID-19 global health crisis. This is too long for the populace in dire straits and the bond's funding capability became useless for such an extremely infectious disease. Some news or feature articles even described it as a scheme "waiting for people to die" (The Guardian, 2020) and "financialization run amok" (The Conversation, 2020).

Our objectives in this project are: (*i*) construct an extensive database for all inputs organised primarily for pandemic-bond analysis using COVID-19 and financial market data; (*ii*) dissect the World Bank's PEFF by looking at details of its processing framework; (*iii*) model and simulate the dynamics of the trigger drivers of the pandemic bond; (*iv*) value a pandemic bond from the perspective of coronavirus using the COVID-19 data. By accomplishing these goals, we hope to verify if those criticisms are supported by quantitative evidence and to make contribution in the further studies of the pandemic-risk-linked CAT bonds.

Various research investigations have been conducted in analysing the CAT-security market and their pricing frameworks; see, for example, (Young, 2004), (Jarrow, 2010), (Nowak & Romaniuk, 2013), and (Burnecki & Giuricich, 2017). In this research, we shall treat the payout event of the pandemic bond analogous to the default event of a defaultable bond, and price it as an non-defaultable bond with trigger-risk adjusted discount rate. The general pricing framework of WB's PEFF and the characteristics of the pandemic bond will also be considered.

The mathematical model for the payout conditions and the pricing set up for the pandemic bond are introduced in Chapter 2. Details of the data processing and implementation methodology for modelling trigger drivers and interest rates are discussed in Chapter 3. Chapter 4 presents all the modelling and the pricing results with sensitive analyses involving the payout probability and bond valuation. Chapter 5 concludes with the summary of our research findings and pinpointing possible research directions in the future.

Chapter 2

Dissecting the pandemic bond structure

2.1 Description

2.1.1 Characteristics of the pandemic bond

Both the International Bank for Reconstruction and Development (IBRD) and International Development Association (IDA) make up the WB. IBRD was the WB's organisation that issued the pandemic bond. There are two classes of notes in the structure of the pandemic bond. Each covers two different sets of disease perils; but both include coronavirus. The basic information for these two classes of notes are summarised in Tables 2.2 and 2.3. The monthly floating-rate coupon payments of the pandemic bond are supported by the donor countries: Australia, Japan, and Germany.

According to the Prospectus Supplement of the pandemic bond (The International Bank for Reconstruction and Development, 2017), for a covered, non-flu disease peril, the pandemic bond's payout mechanism is activated when the following six trigger conditions are met:

1. Eighty-four days has passed since the start date of the WHO-recognised disease outbreak.

2. Such outbreak must have happened in at least 2 territories worldwide[1], with each territory has at least 20 confirmed death cases.

3. The calculated total number of confirmed death (TCD) in the entire covered area[2] has to be greater than a pre-defined threshold[3].

4. The number of rolling total cases (RTC) in the entire covered area has to reach a predefined threshold.

[1] Refer to Table B.1 in Appendix B; all of the worldwide territories in that list constitute the "worldwide area".

[2] Refer to Table B.2 in Appendix B; all of the covered territories in that list constitute the "covered area". The "covered area" means figures in those territories are included the trigger calculation, but not all of those territories will be funded if payout happens.

[3] All "pre-defined thresholds" are related to class of notes and type of diseases; see (The International Bank for Reconstruction and Development, 2017).

5. The calculated confirmation ratio (CR) has to reach a predefined threshold.

6. The calculated new case growth rate (GR) has to be greater than 0.

On the other hand, if the disease outbreak is related to flu (covered by class-A note), the payout mechanism is activated when the following trigger conditions are met:

1. Forty-two days have passed since the start date of the WHO-recognised flu outbreak.

2. The rolling number of confirmed cases has to be greater than or equal to 5000.

3. The calculated new case growth rate has to be greater than 0.

4. The growth rate mean (a component to calculate the new case growth rate) has to be at least 0.265.

The statistics of the triggers stated above (e.g., RTC, CR, etc) could be calculated from the variables listed in Table 2.1. Specific formulae with pre-defined threshold for coronavirus will be introduced in the next subsection. Once all those conditions are satisfied, a portion of or the entire principal of the pandemic bond will be utilised to fund those countries which are eligible to receive the payments. Specifically, those countries are the same countries that are eligible to receive lending from the IDA, and we will call them "IDA countries" in the succeeding discussion.

Full name	Short hand	Formula	Definition
Calculated number of confirmed cases	CC	N/A	The calculated number of cumulative confirmed case based on the reported number of confirmed case in one covered territory C.
Calculated number of probable cases	PC	N/A	The calculated number of cumulative probable cases based on the reported number probable case in one covered territory C.
Calculated number of total cases	TC	CC + PC	The calculated number of cumulative total cases based on the calculated number of confirmed cases and the calculated number of probable cases in one covered territory C.
Total number of cases in the covered area	TCA	$\sum_c TC^c$	The sum of the calculated number of total cases in all of the covered territories.
Calculated number of confirmed deaths	CD	N/A	The calculated number of cumulative confirmed deaths based on the reported number of confirmed deaths in one covered territory C.
Total confirmed deaths in the covered area	TCD	$\sum_c CD^c$	The sum of the calculated number of the confirmed deaths in all of the covered territories.
Number of infected countries (or territories) in the worldwide area	IC	N/A	The number of territories listed in the Appendix B.1 affected by the pandemic, with condition that each country has at least 20 confirmed deaths.

Table 2.1: List of abbreviations and terminology

Properties	Class A	Class B
Issuer	IBRD	IBRD
Issue date	7 July 2017	7 July 2017
Issue price	100% at par	100 % at par
Scheduled maturaty date	15 July 2020 (partially or entirely extendable monthly until 15 July 2021)	15 July 2020 (partially or entirely extendable monthly until 15 July 2021)
Specified currency unit	USD	USD
Aggregate nominal value	225 million	95 million
Interest rate	6-month USD LIBOR + 6.50%	6-month USD LIBOR + 11.10%
Interest payment date	15th day of each month	15th day of each month
Interest rate reset date	15 January & 15 July	15 January & 15 July
Covered virus	Flu and coronavirus	Filovirus, coronavirus, lassa fever(LF), rift valley fever (RVF) and Crimean Congo Hemorrhagic Fever(CCHF)
Payout percentage	Flu: 100% coronavirus 16.7%	Please see Table 2.3: Class B payout percentage

Table 2.2: Basic information about the pandemic bond

Situation	$250 \leqslant$ TCD $\leqslant 750$ (%)	$750 \leqslant$ TCD $\leqslant 2500$ (%)	TCD $\geqslant 2500$ (%)
Regional [4] (corona)	37.50	75.00	100.00
Regional (filo)	30.00	60.00	100.00
Regional (LF, RVF, CCHF)	15.00	30.00	50.00
Global[5] (corona)	43.75	87.50	100.00
Global (filo)	35.00	70.00	100.00
Global (LF, RVF, CCHF)	17.50	35.00	50.00

Table 2.3: Class-B payout percentage

[4] "Regional" means the number of the infected countries worldwide ranging from 2 to 7; an infected country means it has at least 20 confirmed deaths.

[5] "Global" means the number of infected countries worldwide is greater than 7.

2.1.2 Data-interlink processing for the pandemic bond

The processing framework of the pandemic bond relies on the cooperation of three major parties: The World Bank (WB), The World Health Organization (WHO), and AIR Worldwide (AIR) - a third-party agency who is responsible for calculating all trigger statistics before a payout decision is made. When a covered disease or peril occurs, AIR would use the numbers reported from WHO as primary source data and calculate the trigger statistics for periods after the outbreak. In an ideal situation, WHO's report should contain all the information that AIR needs to carry out the calculation. In particular, the following variables:

(*i*) The reported number of confirmed cases in each country.

(*ii*) The reported number of confirmed death in each country.

(*iii*) The reported number of probable cases in each country.

In cases where any one of those variables are not reported by WHO, AIR would use its own data collection, judgement, and models to estimate the missing variables.

The process of determining if the payout is activated after an outbreak are made in a 16-day period. When a pandemic outbreak occurs, AIR would take the figures reported by WHO and calculate trigger statistics on a weekly basis. At the same time, in order to perform the proper calculation procedure (linear average method discussed in the next paragraph), it also needs the data collected from the week immediately following the week under examination. For example, assume that there is a non-flu disease outbreak, and the 84-day rolling period has passed. To determine whether a payout is triggered on the 85th day (earliest time to pay) or not, AIR needs the data from the week where the 85th day falls in, and the data from the week immediately after. The two consecutive 14-day period for collecting the data is called the "reporting window". Moreover, AIR also needs a 2-day "calculation window", which is immediately following the end of the reporting window, to calculate the trigger statistics. Because of this 16-day gap, the earliest day possible for WB to announce the payout event will be 101 days after the start of a non-flu pandemic outbreak. The same framework applies to the flu situations as well.

To convert the "reported" figures in the WHO's reports to "calculated" figures stated in Table 2.1 and to use them as the input parameters for determining the status of triggers, AIR uses the following two smoothing and averaging techniques to process the raw data in each country (territory) in the covered area:

1. A Hyman cubic spline interpolation (Hyman, 1983) is applied to the reported numbers (e.g., total reported confirmed cases, total reported confirmed deaths, total reported probable cases) to fill the missing data points in that country. The period being applied by this method starts from the date when the first case being detected in a country to the end of the current reporting window.

2. A linear (i.e., arithmetic) average in a 14-day rolling window centered at the evaluation date t is used to remove the "lumpiness" of the reported figures. For example, if x_i's

are the reported numbers in the WHO's reports, AIR would use

$$\frac{\sum_{i=t-6}^{t+7} x_i}{14} \tag{2.1}$$

in getting the final result for "calculated" figures on day t.

2.2 The underlying valuation assumptions

We first lay out the assumptions surrounding the modelling framework that leads to the pandemic bond price representation. All processes in our setting are defined and supported by a filtered probability space $(\Omega, \mathscr{F}, \{\mathscr{F}_t\}, \mathbb{Q})$, where $\{\mathscr{F}_t\}$ is a filtration generated jointly by all the processes (interest rate and evolving triggers).

- The pandemic bond price is modelled only from the perspective of coronavirus. This means trigger requirements, payout conditions and data inputs were solely based on coronavirus data.
- The issuer of the pandemic bond, IBRD (or WB) is not defaultable. In this way, we focus on modelling the payout process only.

2.2.1 Interest-rate process

We assume that the interest rates have dynamics as per the the Vasiček model's stochastic differential equation (Vasiček, 1977)

$$\mathrm{d}r_t = a(x_t)(b(x_t) - r_t)\mathrm{d}_t + \sigma(x_t)\mathrm{d}W_t,$$

where W_t is a standard Brownian motion. The parameters a, b, and σ are constants and they refer to the speed of mean reversion, mean-reverting level, and volatility, respectively.

The issuance date of the bond is t_0 and the maturity date is T. We define t_i as the ith payment date in $[t_0, T]$ at which a coupon is due or a principal matures. In this project, we assume that $T = 5$ (years).

The LIBOR rates will be simulated using the Euler discretisation, and leads to the recursion

$$\begin{aligned} r_{t+\Delta t} &= r_t + a(x_t)(b(x_t) - r_t)\Delta t + \sigma(x_t)\Delta W_t \\ &= r_t + a(x_t)(b(x_t) - r_t)\Delta t + \sigma(x_t)\sqrt{\Delta t}\epsilon_t. \end{aligned} \tag{2.2}$$

We use the approximation (in distribution sense) $\Delta W_t = \sqrt{\Delta t}\epsilon_t$, where $\epsilon_t \sim N(0,1)$.

The risk-free rate (one-month LIBOR) or the floating-coupon rate (six-month LIBOR) are independent from the pandemic events. This means that the occurrence of pandemic events will not have major impact on interest-rate movement.

2.2.2 Payout triggers for coronavirus

Description of the triggers

As briefly introduced above, in order to decide whether a payout should be made at time t, a set of triggers has to be satisfied. Specific to the coronavirus situation, the following conditions have to be met:

- Trigger 1: A coronavirus outbreak occurred, recognized by the WHO.
- Trigger 2: Eighty-four days have passed since the outbreak start time recognised in the WHO reports.
- Trigger 3: Such outbreak has spread to two or more worldwide territories, and each worldwide territory has at least 20 confirmed death cases.
- Trigger 4: The total confirmed deaths (TCD) at time t in entire covered area is greater than threshold 1 (T1), where

$$\text{T1} = \begin{cases} 2500 & \text{for Class A Note} \\ 250 & \text{for Class B Note.} \end{cases}$$

  .

- Trigger 5: The rolling total cases (RTC) at time t is at least 250 in the entire covered area, where RTC at day t is calculated from the total case amount (TCA), where

$$\text{RTC(t)} = \text{Total Case Amount(t)} - \text{Total Case Amount(t} - 84).$$

- Trigger 6: Confirmation ratio (CR) at time t exceeds the threshold 2 (T2) in the entire covered area. The confirmation ratio at day t is calculated from the number of rolling total cases (RTC) defined above and the number of rolling confirmed cases (RCC) defined as

$$\text{RCC(t)} = \text{Total Confirmed Case(t)} - \text{Total Confirmed Case (t} - 84)$$

  and

$$\text{CR(t)} = \frac{\text{RCC(t)}}{\min(\text{RTC(t)}, 750)}$$

  with T2 as 20% for having coronavirus as the cause of the pandemic.

- Trigger 7: Case growth rate (GR) in the entire covered area at time t is greater than 0, where GR is calculated through the following steps:

$$\text{NCRC}_i = \ln\left[\frac{\text{TCA}(t-14(i-1)) - \text{TCA}(t-14i)}{\text{TCA}(t-14i) - \text{TCA}(t-14(i+1))}\right]$$

for $i = 1,\ldots,5$,

$$\mu(t) = \frac{1}{5}\sum_{i=1}^{5} \text{NCRC}_i,$$

$$S(t) = \sqrt{\frac{\sum_{i=1}^{5}(\text{NCRC}_i - \mu(t))^2}{4}},$$

$$\text{se}(t) = \frac{S(t)}{\sqrt{5}},$$

$$\text{GR(t)} = \mu(t) - 1.533 \cdot \text{se}(t).$$

Notice that triggers 2 to 7 are the original triggers embedded in the pandemic bond's framework when a covered, non-flu disease outbreak happens. Trigger 1 is added on top of those original triggers since they materialised only if a covered disease outbreak occurs.

2.2.3 Mathematical models for the triggers

We define, in probabilistic terms, the above activated trigger events as follows:

Trigger-1 event as set A, and let the indicator function $X_A(t) : \Omega \to \{0,1\}$ give a binary outcome of whether or not a pandemic happens in the worldwide territories at time t. That is,

$$X_A(t) = \begin{cases} 1 & \text{if a (coronavirus) outbreak has occurred at time } t \\ 0 & \text{otherwise.} \end{cases}$$

Trigger-2 event as set B, and let the indicator function $X_B(t) : \Omega \to \{0,1\}$ produce a binary outcome of whether or not an 84-day rolling period has passed since the outbreak-start date, given that trigger 1 has been activated. This means

$$X_B(t)|(X_A(t) = 1) = \begin{cases} 1 & \text{if } X_A(t-84) = 1 \\ 0 & \text{otherwise.} \end{cases}$$

Trigger-3 event as set C, and let the indicator function $X_C(t) : \Omega \to \{0,1\}$ specify a binary outcome of whether or not 2 or more worldwide countries have at least 20 confirmed deaths caused by a pandemic by time t, given that Trigger 1 is activated. This tells us that

$$X_C(t)|(X_A(t) = 1) = \begin{cases} 1 & \text{if 2 or more countries worldwide are affected, and each affected country has at least 20 deaths caused by the relevant disease at } t \\ 0 & \text{otherwise.} \end{cases}$$

Trigger-4 event as set D, and let the indicator function $X_D(t) : \Omega \to \{0,1\}$ provide a binary outcome of whether or not the total confirmed deaths (TCD) in the entire covered area exceeds T1 at t, given that trigger 1 has been activated. Thus,

$$X_D(t)|(X_A(t)=1) = \begin{cases} 1 & \text{if TCD(t) in covered area exceeds T1} \\ 0 & \text{otherwise.} \end{cases}$$

Trigger-5 event as set E, and let the indicator function $X_E(t) : \Omega \to \{0,1\}$ give rise to a binary outcome of whether or not RTC(t) in the covered area exceeds 250, given that trigger 1 has been activated so that

$$X_E(t)|(X_A(t)=1) = \begin{cases} 1 & \text{if RTC}(t) \geqslant 250 \\ 0 & \text{otherwise.} \end{cases}$$

Trigger-6 event as set F, and let the indicator function $X_F(t) : \Omega \to \{0,1\}$ expresses a binary outcome of whether or not CR(t) in the covered area exceeds T2, given that trigger 1 has been activated. Therefore,

$$X_F(t)|(X_A(t)=1) = \begin{cases} 1 & \text{if CR}(t) \geqslant \text{T2} \\ 0 & \text{otherwise.} \end{cases}$$

Trigger-7 event as set G, and let the indicator function $X_G(t) : \Omega \to \{0,1\}$ represent a binary outcome of whether or not GR(t) in the covered area exceeds 0, given that trigger 1 has been activated. This yields

$$X_G(t)|(X_A(t)=1) = \begin{cases} 1 & \text{if GR}(t) \geqslant 0 \\ 0 & \text{otherwise.} \end{cases}$$

Let O be the intersection of the above sets (events), i.e., all triggers of the pandemic bond are activated. Specifically,

$$O = A \cap B \cap C \cap D \cap E \cap F \cap G.$$

The pandemic bond's payout time is denoted by τ. In particular, τ is $\mathscr{F}_t$- measurable, that is,

$$\tau^{-1}(O) = \{\omega \in \Omega : \tau(\omega) \in O\}$$

such that

$$\tau^{-1}(O) \in \mathscr{F}, O \in \mathscr{B}([0,\infty]).$$

Ergo, the overall payout trigger depends on an indicator function $X_O(t) : \Omega \to \{0,1\}$ implying that the conditions for the pandemic bond to payout are all met, and so

$$X_O(t) := \begin{cases} 1 & \text{if } \prod_{i \in \boldsymbol{K}} X_i(t) = 1 \\ 0 & \text{otherwise.} \end{cases} \tag{2.3}$$

where $i \in \boldsymbol{K} = \{A, B, \ldots, G\}$.

2.2.4 Payout & recovery ratios

If all the triggers are activated at t, a fraction or the entire principal of the notes will be distributed to those IDA countries, and the distributed funds will be used to combat the infectious disease. As a result, the floating-coupon payments, which are calculated based on the principal amount, will be reduced as well. The fraction of the principal to be deducted for each class of note are called the payout ratio. The payout ratio for coronavirus varies based on the characteristic of Class A and Class B notes and the severity of the (coronavirus) outbreak. More notably, we have:

- For "Class A", the payout ratio is 16.7 % once all triggers are activated.
- For "Class B", the payout ratio is related to the TCD and IC, i.e.,

$$\text{Payout ratio (regional)} = \begin{cases} 37.5\% & \text{if } 250 \leqslant \text{TCD}(t) \leqslant 750 \\ 75\% & \text{if } 750 \leqslant \text{TCD}(t) \leqslant 2500 \\ 100\% & \text{if TCD}(t) \geqslant 2500. \end{cases}$$

$$\text{Payout ratio (global)} = \begin{cases} 43.75\% & \text{if } 250 \leqslant \text{TCD}(t) \leqslant 750 \\ 87.5\% & \text{if } 750 \leqslant \text{TCD}(t) \leqslant 2500 \\ 100\% & \text{if TCD}(t) \geqslant 2500. \end{cases}$$

From the payout ratios, the recovery ratio can be deduced as (1 - payout ratio) with respect to each class of the notes.

2.2.5 Pricing the fixed coupon payment

When considering the pricing framework, we start with a simple case where the bond pays fixed coupons. Assuming the pandemic bond pays fixed amount of coupon c at each coupon payment date, the outstanding nominal note amount p will be paid at maturity T. There are n coupon payments scheduled between the issuance date and maturity date. For each coupon-payment time at t_i, $i = 1, \ldots, n$, consider the payment as a zero-coupon bond which pays c or p at times t_i, and nothing before time t_i.

A defaultable claim with an exogenous default processes can be modelled as a default-free claim with default-adjusted short rate, under the assumption of recovery of market value

(Duffie & Singleton, 1999). Assuming as well that the market is arbitrage-free, the bond will be treated as a new pandemic bond with reduced coupon payment and principal. Therefore, the loss process L_t is viewed as the payout process explicitly expressed in Subsection 2.2.4. At the same time, we consider the payout event analogous to a default event (Duffie & Singleton, 1999). For all time $s_i \leqslant t_i$, suppose

- h_{s_i} is the conditional probability at time s_i, under the risk-neutral measure $\mathbb{Q}$, of payout happening between s_i and the next day $s_i + 1$, given the information available s_i at which payout did not happen; here,

$$h_{si} = E^{\mathbb{Q}}\left[X_O(t_i) = 1 \middle| \mathscr{F}_{s_i}\right],$$

- L_{s_i} is the payout ratio or the loss process given the information at s_i, which was defined in Subsection 2.2.4,
- r_{s_i} represents the default-free short rate at time s_i and follows a Vasiček model,
- $V(s_i+1)$ is the value of the bond at time s_i+1 without the payout event being triggered,
- $V'(s_i+1)$ denotes the value of the bond at time s_i+1 with the payout event triggered,
- $Z(s_i)$ stands for the fair value of the bond at time s_i, and
- Y_{t_i} is the scheduled payment at t_i (c or p).

Then,

$$\begin{aligned} Z(s_i) &= h_{s_i} e^{-r_{s_i}(s_i+1-s_i)} E^{\mathbb{Q}}[V'(s_i+1)|\mathscr{F}_{s_i}] + (1-h_{s_i}) e^{-r_{s_i}(s_i+1-s_i)} E^{\mathbb{Q}}[V(s_i+1)|\mathscr{F}_{s_i}] \\ &= h_{s_i} e^{-r_{s_i}} (1-L_{s_i}) E^{\mathbb{Q}}[V(s_i+1)|\mathscr{F}_{s_i}] + (1-h_{si}) e^{-r_{si}} E^{\mathbb{Q}}[V(s_i+1)|\mathscr{F}_{s_i}]. \end{aligned} \tag{2.4}$$

Equation (2.4) can be rewritten as (Duffie & Singleton, 1999)

$$Z(s_i) = E^{\mathbb{Q}}\left[e^{-\sum_{j=0}^{t_i-1} R_{s_i+j}} Y_{t_i} \middle| \mathscr{F}_{s_i} \right], \tag{2.5}$$

where the trigger-risk adjusted short rate is R_{s_i}, and the discount factor $e^{-R_{s_i}} = h_{s_i} e^{-r_{s_i}}(1 - L_{s_i}) + (1-h_{s_i}) e^{-r_{s_i}}$. Using the approximation of e^c when c is small, we get $R_{s_i} \approx r_{s_i} + h_{s_i} L_{s_i}$.

Combining and incorporating all virtual "zero coupon" bond into original pandemic bond, the price of a pandemic bond with fixed coupon payments at s_i is

$$\sum_{i=1}^{n} Z(s_i) = \sum_{i=1}^{n} E^{\mathbb{Q}}\left[e^{-\sum_{j=0}^{t_i-1} R_{s_i+j}} Y_{t_i} \middle| \mathscr{F}_{s_i} \right]. \tag{2.6}$$

The equivalent expression in the continuous case of equation (2.6) is

$$\sum_{i=1}^{n} Z(s_i) = \sum_{i=1}^{n} E^{\mathbb{Q}}\left[e^{-\int_{s_i}^{t_i} R_j \mathrm{d}j} Y_{t_i} \middle| \mathscr{F}_{s_i} \right]. \tag{2.7}$$

2.2.6 Pricing floating-coupon payment

The only difference between the fixed and floating coupon is that the floating-coupon payment depends on a floating-benchmark rate. Therefore, the coupon payment is a random variable until the benchmark rate is known. As given in Table 2.2, the benchmark rate of the pandemic bond is the 6-month US dollar LIBOR rate, which resets in January and July of each year. Standing at time $s_i \leqslant t_i$, we could estimate the payment $C(t_i)$ with the information available at s_i. Define

- u as the additional spread added based on the benchmark rate,
- p as the nominal note amount which the coupon payment depends on,
- $r_{t_i}^{\text{6-LB}}$ as the benchmark rate applied at ith payment date t_i, and
- q as the actual number of days covered in each interest period.

Therefore,

$$\begin{aligned} E^{\mathbb{Q}}\left[C(t_i)\middle|\mathscr{F}_{s_i}\right] &= E^{\mathbb{Q}}\left[\left(r_{t_i}^{\text{6-LB}}+u\right)\cdot\left(\frac{\text{q}}{360}\right)\cdot\text{p}\middle|\mathscr{F}_{s_i}\right] \\ &= \frac{\text{q}}{360}\cdot\text{p}\cdot\left(E^{\mathbb{Q}}\left[r_{t_i}^{\text{6-LB}}\middle|\mathscr{F}_{s_i}\right]+u\right), \end{aligned} \tag{2.8}$$

where $E^{\mathbb{Q}}[r_{t_i}^{\text{6-LB}}|\mathscr{F}_{s_i}]$ can be interpreted as the forward rate of the 6-month LIBOR given the information at s_i and could be deduced from a zero-coupon bond's yield curve. Under the Vasiček model, the zero-coupon bond's yield at time t and maturity T (Zhou & Mamon, 2012) is

$$Y(t,\text{T}) = -\frac{\log Z(t,\text{T})}{\text{T}-t}, \tag{2.9}$$

where $Z(t,\text{T})$ is the zero-coupon bond's price at time t.

For each coupon payment, standing at time s_i, we could construct the yield curve using formula (2.9) for each payment date, that is, $Y(s_i,t_i)$ for $i=0,\ldots,n$. From equation (2.8), we obtain the value of the floating-rate coupon payments with basis of information available at s_i.

2.2.7 The main pricing result

Subsections 2.2.5 and 2.2.6 provides the preliminary derivations supporting the risk-neutral pricing methodology for the pandemic bond.

At time $s\in[0,T]$, the pandemic bond price $Z(s)$ is

$$\begin{aligned} Z(s) &= \sum_{i=1}^{n} E^{\mathbb{Q}}\left[e^{-\int_s^{t_i} R_j \text{d}j} Y_{t_i}|\mathscr{F}_s\right] \\ &= \sum_{i=1}^{n} E^{\mathbb{Q}}\left[e^{-\int_s^{t_i} R_j \text{d}j} C(t_i)|\mathscr{F}_s\right] + E^{\mathbb{Q}}\left[e^{-\int_s^{t_n} R_j \text{d}j}\cdot\text{p}|\mathscr{F}_s\right]. \end{aligned} \tag{2.10}$$

Let the short rate (1-month LIBOR) be denoted by $r_t^{\text{1-LB}}$, and the benchmark rate (6-month LIBOR) be $r_t^{\text{6-LB}}$; these LIBOR rates are almost surely correlated. Considering the covariance $\text{Cov}(r_t^{\text{1-LB}}, r_t^{\text{6-LB}})$, equation (2.10) becomes

$$\begin{aligned} Z(s) &= \sum_{i=1}^{n} \left\{ E^{\mathbb{Q}}\left[e^{-\int_s^{t_i} R_j \mathrm{dj}} \middle| \mathscr{F}_s \right] \cdot E^{\mathbb{Q}}\left[C(t_i) \middle| \mathscr{F}_s \right] + \text{Cov}(r_t^{\text{1-LB}}, r_t^{\text{6-LB}})\Big|_{t=s} \right\} + E^{\mathbb{Q}}\left[e^{-\int_s^{t_n} R_j \mathrm{dj}} \cdot \mathrm{p} \middle| \mathscr{F}_s \right] \\ &= \sum_{i=1}^{n} \left\{ E^{\mathbb{Q}}\left[e^{-\int_s^{t_i} R_j \mathrm{dj}} \middle| \mathscr{F}_s \right] \cdot \left[\frac{\mathrm{q}}{360} \cdot \mathrm{p} \cdot (E^{\mathbb{Q}}\left[r_{t_i}^{\text{6-LB}} \middle| \mathscr{F}_s \right] + u) \right] + \text{Cov}(r_t^{\text{1-LB}}, r_t^{\text{6-LB}})\Big|_{t=s} \right\} \\ &\quad + E^{\mathbb{Q}}\left[e^{-\int_s^{t_n} R_j \mathrm{dj}} \cdot \mathrm{p} \middle| \mathscr{F}_s \right]. \end{aligned} \tag{2.11}$$

2.2.8 Risk-neutral and real-world pricing

Considering that the underlying variables of the pandemic bond are not tradeable (e.g. number of confirmed cases, death, etc), it is impossible to replicate every payoff of the bond in the market. Hence, we rely on actuarial valuation under $\mathbb{P}$ and choose it to coincide with the financial pricing measure $\mathbb{Q}$ under this incomplete market setting (i.e., $(E^{\mathbb{Q}}[\cdot] = E^{\mathbb{P}}[\cdot])$. This is often the case for valuing medium to long-term insurance products. Although the risk-neutral approach for determining the fair value of such (re)insurance product may not be practically plausible to achieve, it is theoretically viewed as a benchmark price when it is introduced to the market. The issuer of the bond may adjust the price (relative to the benchmark) to set the desired gross premium (Liu, Mamon, & Gao, 2013).

Chapter 3

Statistical data modelling and implementation

3.1 Data collection and processing

The pertinent COVID-19 numbers used in this project was based on the data sets constructed by the scientific online publication *Our World in Data* (Roser, Ritchie, Ortiz-Ospina, & Hasell, 2020), where they collect the daily statistics of the pandemic from the European Center for Disease Prevention and Control(ECDC). These data sets integrate statistics for several variables of interest that have little discrepancy with WHO's reported numbers. Our data collection period spans the start of COVID-19 outbreak date recognised by the WHO, which is 31 December 2019 up to 29 May 2020.

Further to the description given in Chapter 2, AIR would use interpolation to fill the missing numbers since the time when the first case was confirmed in each country or territory. We implemented the same methodology to 'find' missing values before the data are fed into our models.

There are four fundamental variables – constructed from the three variables stated in Subsection 2.1.2 – needed in order to generate the realisations of triggers 3 to 7. We recall them as follows:(*i*) number of countries affected by the COVID-19 worldwide (IC), (*ii*) reported number of confirmed cases in covered area, (*iii*) reported number of confirmed deaths in covered area, and (*iv*) reported number of total cases in covered area.

The first three variables can be obtained from the data set directly. However, an essential component of the (*iv*) *reported number of total cases in covered area* is the total number of reported probable cases. Such a total number is not explicitly stated by many countries' health departments during the COVID-19 crisis, especially in many covered developing countries. For those countries that did have this item reported (e.g., Canada and New Zealand), the reported numbers of probable cases have different meanings defined by each health department. Additionally, the figure for the probable cases was not consolidated at a global level (i.e., country reports submitted to international organisations such as the WHO and

ECDC). Therefore, the (*iv*) *reported number of total cases in covered area* must be estimated, and we employ a heuristic method to do this.

Although it has been shown in some studies that the fatality rate of COVID-19 is closely related to the infected patient's age and the condition on the local health-care system; see `https://covid-19-canada.uwo.ca/en/analysis_regression.html`, accessed 2020-06-17. For the sake of simplicity, we assume that the death rate (DR) of the COVID-19 in the covered area is a fixed number. Let the d_i be the fatality rate in the confirmed cases calculated on the i^{th} day in our data. The DR then is estimated by taking the average of the reported daily confirmed death rates in the entire covered area, i.e.,

$$\text{DR} = \text{E}[d_i] = \frac{1}{n}\sum_{i=1}^{n} d_i \tag{3.1}$$

Let $\text{TCD}'(t)$ denote the (*ii*) *reported confirmed cases in covered area* at time t, and $\text{TCA}'(t)$ denote (*iv*) *reported total number of cases in covered area* at time t [1]. With the estimated death rate of COVID-19 in the covered area in equation(3.1), we can further estimate the TCA'(t) as

$$\text{TCA'(t)} \approx \frac{\text{TCD'(t)}}{\text{DR}} \tag{3.2}$$

The rationale for equation (3.2) is as follows. Instead of estimating the number of probable cases in the entire covered area, which is very difficult and unlikely to be reliable due to the missing data, we directly estimate the number of cases "deemed confirmed"; of course, this means "not necessarily confirmed" considering various reasons (e.g., lack of tests). The concept of this approximated number from the left-hand side of the equation (3.2) is analogous to that of TCA', which is the sum of the reported number of confirmed cases and the reported number of probable cases.

The above-mentioned heuristics enable us to obtain fully the estimates of all four fundamental variables[2]. The VARMA models are introduced in Subsection 3.2.2 to model the variables' dynamics.

3.2 Modelling the trigger risk

3.2.1 Breaking down the trigger risk

As stated in Chapter 2, the trigger risk, which is a similar in concept to "hazard rate", is defined as the conditional probability

[1] The difference between TCA'(t) and TCA(t) is that TCA'(t) has to be processed with the smoothing and averaging techniques described in Subsection 2.1.2 to obtain $\text{TCA}(t)$. The same logic applies to $\text{TCD}'(t)$ and $\text{TCD}(t)$.

[2] See Appendix A for the constructed data set.

$$h_s = E^{\mathbb{P}}[X_O(t) = 1|\mathscr{F}_s]. \tag{3.3}$$

Since X_O is an indicator function, its expectation is the conditional probability $\mathbb{P}[X_O(t) = 1|\mathscr{F}_s]$. As the statistics for triggers 2 through 7 only materialise if trigger 1 is activated, we can further expand equation (3.3) to get

$$\begin{aligned} h_s &= E^{\mathbb{P}}[X_O(t) = 1|\mathscr{F}_s] = \mathbb{P}[X_O(t) = 1|\mathscr{F}_s] \\ &= \mathbb{P}[X_B(t) = \ldots = X_G(t) = 1|X_A(t) = 1]\mathbb{P}[X_A(t) = 1|\mathscr{F}_s]. \end{aligned} \tag{3.4}$$

The first component of the right-hand side of equation (3.4), $\mathbb{P}[X_B(t) = \ldots = X_G(t) = 1|X_A(t) = 1]$, can be interpreted as the conditional probability that the triggers altogether are activated given the occurrence of a pandemic. This joint conditional probability is driven by the four fundamental variables listed in Section 3.1. Therefore, determining such a joint conditional probability is essentially modelling and evaluating those four variables accurately. Time series models could be used as efficient mathematical tools in infectious-disease modelling and surveillance because of their straightforward implementation (Allard, 1998). More importantly, there is a strong interconnection amongst the four variables, a multivariate time series model (e.g.,Vector Autoregressive Moving Average (VARMA)) would be more appropriate. The VARMA model is presented in Subsection 3.2.2 to model the dynamic behaviour of the four fundamental variables.

Using the Monte-Carlo simulation method with the aid of the VARMA-estimated parameters, various paths of the four underlying variables could be generated. From the simulated variables, the resulting trigger statistics are recovered. Then, we could estimate the proportion of the total number of simulated paths for which all triggers are activated at the same time. This proportion is the estimate of the conditional probability $\mathbb{P}[X_B(t) = \ldots = X_G(t) = 1|X_A(t) = 1]$.

The second factor in the right-hand side of equation (3.4), $\mathbb{P}[X_A(t) = 1|\mathscr{F}_s]$, is interpreted as the probability (on a per-day basis) that at least one pandemic happens at future time t given that no pandemic occurs currently (time s). We assume that the pandemic occurrence follows a simple Poisson distribution, and the frequency parameter can be estimated using the historical record of major pandemics. Since the 1700s, there have been approximately 14 major non-flu pandemics around the globe[3]. On average, there are approximately 5 major pandemics per 100 years or 35600 days. So, standing at the issuance day where $s = 0$, our conditional-probability estimate is

$$\mathbb{P}[X_A(t) = 1|\mathscr{F}_0] \sim \text{Poisson}(\lambda) \tag{3.5}$$

where $\lambda = \frac{5}{36500}$.

[3]Refer to Appendix C chronicling the major non-flu pandemics from the 1700s onwards.

Moreover, given greater mobility nowadays, the likelihood of having a large-scale outbreak would be larger than those in the 18th, 19th and 20th century. Therefore, a sensitivity analysis will be conducted on the bond price with respect to a varying λ.

3.2.2 Multivariate time-series models

Suppose z_t is a stationary vector time series. The vector autoregressive model of lag order p, or VAR(p) for short, is given by

$$z_t = \phi_0 + \phi_1 z_{t-1} + \ldots + \phi_p z_{t-p} + a_t, \tag{3.6}$$

where ϕ_0 is the intercept vector; $z_{t-1}, \ldots, z_{t-p}$ are the p lags used as predictors; ϕ_i are the coefficient matrices for the p lags; and a_t is the noise vector which is usually assumed to follow a multivariate normal distribution.

A VAR(p) model with k variables can be written in matrix form

$$\begin{bmatrix} z_{1t} \\ z_{2t} \\ \vdots \\ z_{kt} \end{bmatrix} = \begin{bmatrix} \phi_{10} \\ \phi_{20} \\ \vdots \\ \phi_{k0} \end{bmatrix} + \begin{bmatrix} \phi_{1,11} & \cdots & \phi_{1,1k} \\ \phi_{1,21} & \cdots & \phi_{1,2k} \\ \vdots & \ddots & \vdots \\ \phi_{1,k1} & \cdots & \phi_{1,kk} \end{bmatrix} \begin{bmatrix} z_{1,t-1} \\ z_{2,t-1} \\ \vdots \\ z_{k,t-1} \end{bmatrix} + \ldots + \begin{bmatrix} \phi_{p,11} & \cdots & \phi_{p,1k} \\ \phi_{p,21} & \cdots & \phi_{p,2k} \\ \vdots & \ddots & \vdots \\ \phi_{p,k1} & \cdots & \phi_{p,kk} \end{bmatrix} \begin{bmatrix} z_{1,t-p} \\ z_{2,t-p} \\ \vdots \\ z_{k,t-p} \end{bmatrix} + \begin{bmatrix} a_{1t} \\ a_{2t} \\ \vdots \\ a_{kt} \end{bmatrix}$$

Similarly, a VMA(q) model, which stands for vector moving average model of lag order q, has the representation

$$z_t = \mu + a_t - \theta_1 a_{t-1} - \ldots - \theta_q a_{t-q},$$

where μ is the mean vector for the vector time series; $a_t, \ldots, a_{t-q}$ are the vector errors from lag 0 to lag q; and $\theta_1, \ldots, \theta_q$ are the coefficients of the lagged error vectors in positions 1, …, q.

A VMA(q) model with k variables can be written in matrix form

$$\begin{bmatrix} z_{1t} \\ z_{2t} \\ \vdots \\ z_{kt} \end{bmatrix} = \begin{bmatrix} \mu_{10} \\ \mu_{20} \\ \vdots \\ \mu_{k0} \end{bmatrix} + \begin{bmatrix} a_{1t} \\ a_{2t} \\ \vdots \\ a_{kt} \end{bmatrix} - \begin{bmatrix} \theta_{1,11} & \cdots & \theta_{1,1k} \\ \theta_{1,21} & \cdots & \theta_{1,2k} \\ \vdots & \ddots & \vdots \\ \theta_{1,k1} & \cdots & \theta_{1,kk} \end{bmatrix} \begin{bmatrix} a_{1,t-1} \\ a_{2,t-1} \\ \vdots \\ a_{k,t-1} \end{bmatrix} - \ldots - \begin{bmatrix} \theta_{q,11} & \cdots & \theta_{q,1k} \\ \theta_{q,21} & \cdots & \theta_{q,2k} \\ \vdots & \ddots & \vdots \\ \theta_{q,k1} & \cdots & \theta_{q,kk} \end{bmatrix} \begin{bmatrix} a_{1,t-q} \\ a_{2,t-q} \\ \vdots \\ a_{k,t-q} \end{bmatrix}$$

A VARMA(p, q) model combines the VAR(p) and VMA(q) models yielding

$$z_t = \phi_0 + \phi_1 z_{t-1} + \ldots \phi_p z_{t-p} + a_t - \theta_1 a_{t-1} - \ldots \theta_q a_{t-q}.$$

The best VAR(p) model is selected by minimising the Akaike Information Criterion (AIC) (Akaike, 1974), whilst the best VARMA(p, q) model is selected via the extended cross-correlation matrix (Tsay, 2013).

In a VARMA model (including VAR and VMA), it is always assumed that the input time series is stationary. Therefore, before we fit a VARMA model to a vector time series data of the four fundamental variables, the original vector of data must be transformed into a stationary form. In this project, we employ the log transformation and differencing[4]. The augmented Dickey-Fuller (ADF) test (Dickey & Fuller, 1981) and the Kwiatkowski–Phillips–Schmidt–Shin (KPSS) test (Kwiatkowski, Phillips, Schmidt, & Shin, 1992) to confirm that the transformed time series are indeed stationary.

3.2.3 Model validation

In order to test whether the multivariate time-series model's goodness of fit to the dataset, the Ljung-Box (LB) test (Ljung & Box, 1978) is applied on the residuals and squared residuals for serial correlation and auto-regressive conditional heterroscedastic (ARCH) effect.

The LB test involves the following:
Null hypothesis H_0: The data are randomly distributed.
versus
Alternative hypothesis H_a: The data are not randomly distributed; that is, they exhibit serial correlation.

The LB statistics is calculated as

$$Q = n(n+2)\sum_{k=1}^{h}\frac{\widehat{\rho_k}^2}{n-k}, \tag{3.7}$$

where n is the sample size, h is the number of lags being tested, and $\hat{\rho}_k$ is the sample auto-correlation at lag k.

For a model to be considered a good fit to the vector of time-series data, the LB-test result on serial-correlation and ARCH effects should not reject H_0. In other words, the LB test's p-value should not be smaller than the significance level α. Here, we choose the conventional value for $\alpha = 0.05$.

3.2.4 Asymptotic approach using the bootstrap

In certain situations of modelling the non-linear vector time series using VARMA model, the best fitted model may not comply with some of the modelling assumptions. For example, the distribution of the residuals of the best fitted models may not be multivariate normal. This can lead to inaccurate statistical inference which is based on the normality assumption of the residuals. An asymptotic approach such as bootstrapping provides reliable estimates of model parameters and their corresponding statistical inference without

[4]Refer to Appendix D.1 for details.

making any assumptions. As long as the resulting model has insignificant serial autocorrelation and structural changes on the residuals, the model can be viewed as a good fit to the multivariate times-series data.

The steps for the bootstrap method are as follows:

1. Obtain the stationary vector time series by applying the log transformation and differencing on the original vector of time-series data.

2. Select the best fitted VAR(p) or VARMA(p, q) model on the transformed vector time series.

3. Partition the stationary time series into n consecutive time blocks to construct a sample pool. In this way, the autocorrelation structure of the time series is preserved.

4. Randomly sample blocks from the sample pool with replacement until the original length of the vector time series is obtained.

5. Fit the selected VAR(p) or VARMA(p, q) model in the step 2 to the reconstructed data set.

6. Obtain all coefficient estimates, their standard errors, residuals, LB-test statistics on the residuals for serial correlation, and the squared residuals for ARCH effects.

7. Repeat steps 4 - 7 above m times, taking the average of the parameter coefficients as the best estimates. At the same time, compute the standard error for those coefficient estimates and the distribution of p-values of LB-test statistics as well.

In this project, we chose n = 7 and m = 1,000, and the implementation details are included in Appendix D.1.

With the parameters obtained from the validated VAR(p) or VARMA(p, q) model, we simulate 10,000 sample paths for each of the four underlying variables given the random amount of time (in days) which follows an exponential distribution

$$\text{Exp}(\lambda)$$

The rationale behind this assumption is that we previously suggested that the pandemic's occurrence follows a Poisson distribution. This implies that the time between two pandemics has an exponential distribution. Both Poisson and exponential distributions share the same rate parameter λ.

3.2.5 Estimation of trigger probability

With the 10,000 Monte-Carlo simulations for each underlying variable generated from the best fitted model, the following steps are implemented to estimate the conditional trigger probability $\mathbb{P}[X_O(t) = 1|\mathscr{F}_0]$:

1. Conduct a linear average on each underlying variables with a 14-day rolling windows as stated in equation (2.1) to obtain the "calculated" figures defined in Table 2.1.

2. Evaluate triggers 3 to 7 from the "calculated" figures based on the formula listed in Subsection 2.2.2.

3. Start with the 85th observation, and check if all triggers 3 through 7 are activated at the same time. If the data series has a length less than 85, it will be treated as trigger 2 "eighty-four days have passed since the outbreak start" is not activated.

4. Record the TCD and IC which are used to calculate the expected payout ratio for Class B notes.

5. Determine how many sample paths out of 10,000 have all 5 triggers simultaneously activated during its development time. The proportion is the estimated conditional probability $\mathbb{P}[X_B(t) = \ldots = X_G(t) = 1 | X_A(t) = 1]$.

6. Multiply $\mathbb{P}[X_A(t) = 1|\mathscr{F}_0]$ obtained from the Poisson distribution with above estimated probability. Then, we could estimate the trigger probability $\mathbb{P}[X_O(t) = 1|\mathscr{F}_0]$.

7. Calculate the expected payout ratio L for Class B notes as it is related to IC & TCD.

Details of the R-code implementation for the above estimation are demonstrated in Appendix D.2.

3.3 Modelling interest rate

As stated in Chapter 2, we suppose that the interest rate follows the Vasiček model with dynamics

$$\mathrm{d}r_t = a(b - r_t)\mathrm{d}t + \sigma \mathrm{d}W_t.$$

Using the maximum-likelihood estimation (MLE) technique put forward in Zhou and Mamon (Zhou & Mamon, 2012), the parameters a, b, and σ are calibrated as

$$\hat{a} = -\frac{\sum_{i=1}^{N-1} r_{t_i}\left(\sum_{i=1}^{N-1} r_{t_i} - \sum_{i=1}^{N-1} r_{t_{i+1}}\right) + (N-1)\left(\sum_{i=1}^{N-1} r_{t_i} r_{t_{i+1}} - \sum_{i=1}^{N-1} r_{t_i}^2\right)}{\Delta t\left[(N-1)\sum_{i=1}^{N-1} r_{t_i}^2 - \left(\sum_{i=1}^{N-1} r_{t_i}\right)^2\right]},$$

$$\hat{b} = -\frac{-\sum_{i=1}^{N-1} r_{t_i} r_{t_{i+1}} \sum_{i=1}^{N-1} r_{t_i} + \sum_{i=1}^{N-1} r_{t_{i+1}} \sum_{i=1}^{N-1} r_{t_i}^2}{\sum_{i=1}^{N-1} r_{t_i}\left(\sum_{i=1}^{N-1} r_{t_i} - \sum_{i=1}^{N-1} r_{t_{i+1}}\right) + (N-1)\left(\sum_{i=1}^{N-1} r_{t_i} r_{t_{i+1}} - \sum_{i=1}^{N-1} r_{t_i}^2\right)},$$

and

$$\hat{\sigma} = \sqrt{\frac{1}{(N-1)\Delta t}\sum_{i=1}^{N-1}\left(r_{t_{i+1}} - r_{t_i} - \hat{a}(\hat{b} - r_{t_i})\Delta t\right)^2},$$

where

N is the total number of observation in the sample,
r_{t_i} is the observed interest rate at time point t_i, and
Δt is the time increment between each interest observation.

With the aid of the MLE technique to calculate the Vasiček parameters, we model the evolution of the one-month and six-month LIBOR rates as the discounting rates and floating-coupon payments are pegged on these LIBOR rates.

As the discount rate 1-month US LIBOR and the base floating coupon rate 6-month US LIBOR are highly likely to be correlated, we will capture this relation.

Let W_{t1} be the Brownian motion driving the 1-month US LIBOR rate, and W_{t2} be the Brownian motion governing the 6-month US LIBOR rate. It is imposed on these Brownian motions to have the correlation through

$$\mathrm{d}W_{t1}\mathrm{d}W_{t2} = \rho \mathrm{d}t. \tag{3.8}$$

where ρ can be estimated from the interest-rate data. Such relationship in equation (3.8) requires the construction of W_{t2} as follows:

$$W_{t2} = \rho W_{t1} + \sqrt{1-\rho^2} W_{t3} \tag{3.9}$$

where W_{t3} and W_{t1} are independent Brownian motions.

In this project, we price the pandemic bond as if it was issued at its original issuance date of 07 July 2017, and matures 5 years later which will be on 15 July 2022. In addition, we assume that the par value of one note is \$1,000. As indicated in Table 2.2, the floating-coupon rate is updated every 15 January and 15 July in each year. Therefore, the following steps are taken to price the pandemic bond:

1. Calibrate the model parameters (a, b, σ) using the one-month and six-month LIBOR data.

2. With the estimated model parameters, simulate the one-month LIBOR rate using the Euler discretisation covering the life of the pandemic bond.

3. Simulate the six-month LIBOR rate using the Euler discretisation spanning the pandemic bond's life.

4. Record each six-month LIBOR rate, as they are updated on 15 January and 15 July in each year during the life of the pandemic bond.

5. Calculate the floating payment at each interest-payment date employing the formula (2.8) in Subsection 2.2.6.

6. Apply the trigger-risk adjustment to the one-month LIBOR in equation (2.5).

7. Discount each floating payment using the trigger-risk adjusted discount rate.

8. Sum all the discounted floating payments to obtain the valuation in one path.

9. Repeat steps 2 to 8 for 10,000 times to obtain the expected value representing the value of Class A and Class B notes and their associated standard errors.

10. Vary λ using the values $\frac{5}{36500}$, $\frac{10}{36500}$, $\frac{25}{36500}$, and $\frac{50}{36500}$. Repeat step 9 to conduct the sensitivity analysis of the bond price with respect to λ.

Appendix D.3 contains the algorithm details in modelling the interest rates and pricing the pandemic bond.

Chapter 4

Empirical results

4.1 The four underlying variables and their trends

Figures 4.1 – 4.3 display the trend of the four fundamental variables we discussed in Section 3.1 since the start of COVID-19.

From Figure 4.3, we can see that our heuristic approach to obtain the TCA'(t) in equation (3.2) is reasonable. That is, the estimate for the reported number of total cases or 'amount' (TCA') is always greater than or equal to the actual reported number of confirmed cases. The gap between the blue and red lines is the estimated number of probable cases in the covered area. This gap widens, and it shows that the COVID-19 severity in some covered developing countries (e.g., Brazil) started to magnify towards the end of May 2020.

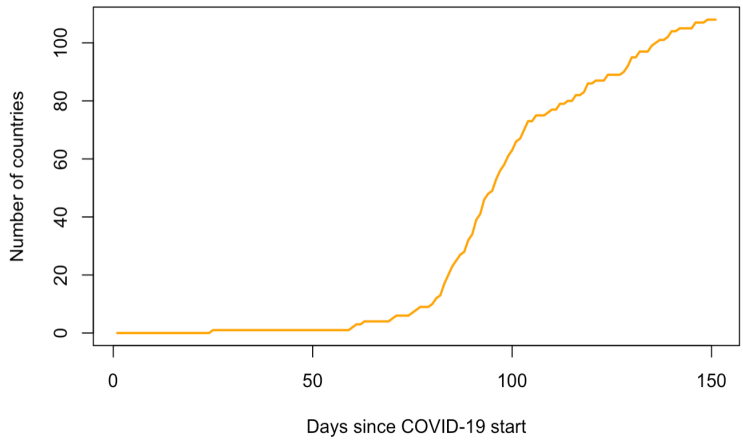

Figure 4.1: Number of infected countries worldwide

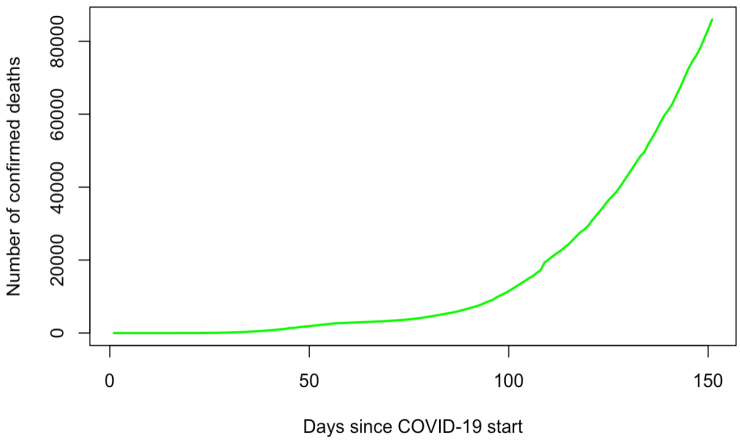

Figure 4.2: Reported total number of confirmed deaths in the covered area

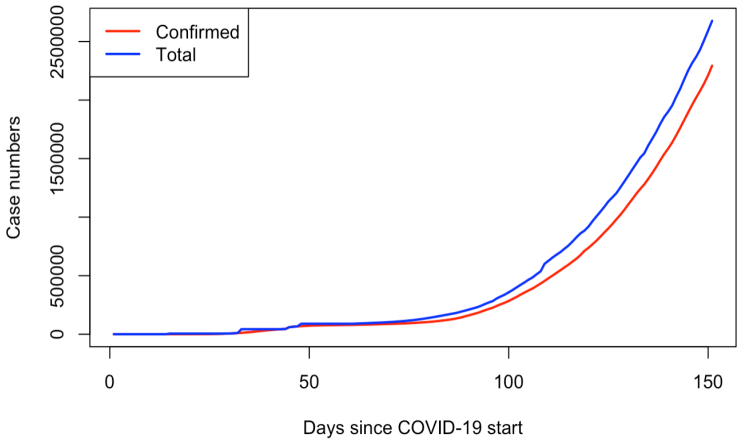

Figure 4.3: Reported total number of confirmed case versus estimated total case

4.2 Multivariate time-series modelling

Using the AIC and the extended cross-correlation matrices as selection criteria, the VAR(9) and VARMA(2, 1) were chosen in the modelling of the four fundamental variables. In the end, the VAR(9) model was picked as our final model because VARMA(2, 1) failed in both the serial-correlation and ARCH-effect tests.

Tables 4.1 and 4.2 display the p-values arising from the Ljung-Box test of autocorrelation and ARCH effect on model residuals for the chosen VAR(9) model. The significance level α is set to 5%. Those p-values all stay above the 5% significance level that validate the VAR(9) model as it passes both tests asymptotically.

Residual lags	Average p-value	Standard error of p-value
Lag 5	0.989	0.002460
Lag 10	0.985	0.002904
Lag 15	0.926	0.005830
Lag 20	0.839	0.008854
Lag 25	0.733	0.010969
Lag 30	0.751	0.010552

Table 4.1: p-values of Ljung-Box test for serial correlation of residuals

Squared residual lags	Average p-value	Standard error of p-value
Lag 5	0.063	0.005746
Lag 10	0.158	0.009270
Lag 15	0.190	0.010422
Lag 20	0.239	0.011523
Lag 25	0.120	0.008498
Lag 30	0.156	0.009625

Table 4.2: p-values of Ljung-Box test for the ARCH effect

4.3 Simulation of trigger processes

All trigger-simulated results generated under a 150-day horizon are shown in Figures 4.4 – 4.8. The red dashed lines are the thresholds of the triggers, and the cyan lines are the averaged paths.

Notice that in Figure 4.8, the starting point is the 85th day since the beginning of the simulation. This is because the growth rate can be calculated only if the number of observations is greater than 84.

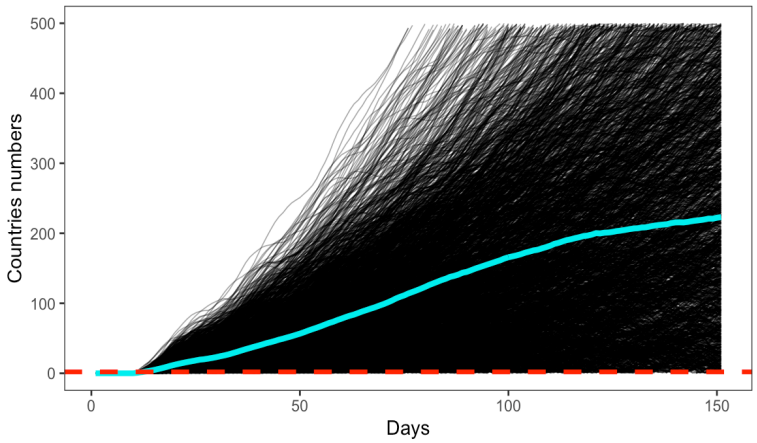

Figure 4.4: **[Trigger 3]** Simulated number of infected territories worldwide

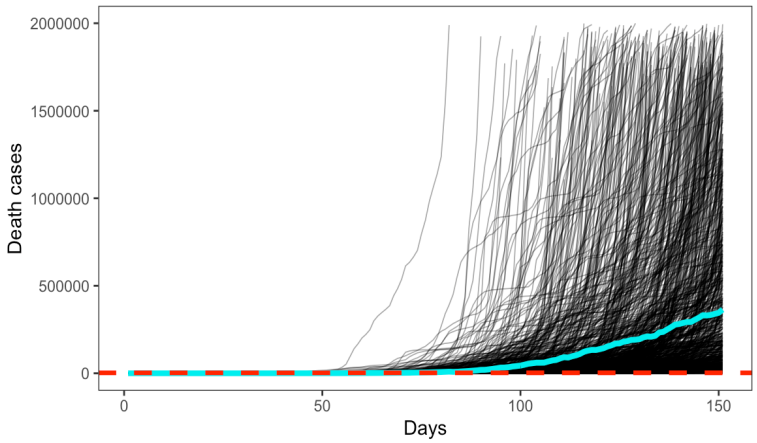

Figure 4.5: **[Trigger 4]**Simulated total number of deaths in the covered area

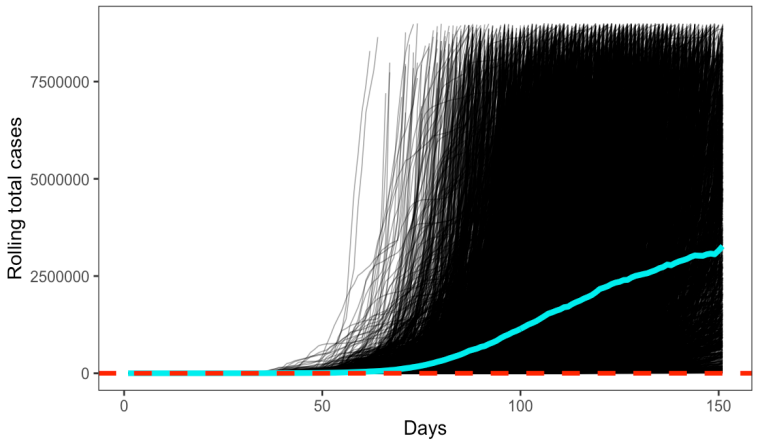

Figure 4.6: **[Trigger 5]** Simulated rolling total number of cases in the covered area

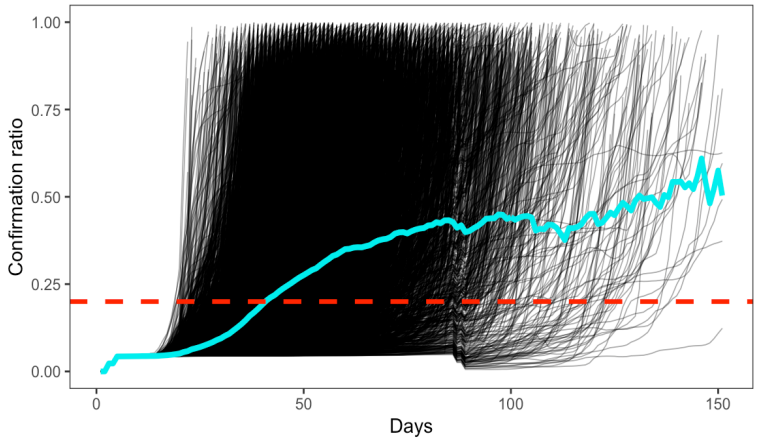

Figure 4.7: **[Trigger 6]** Simulated confirmation ratio in the covered area

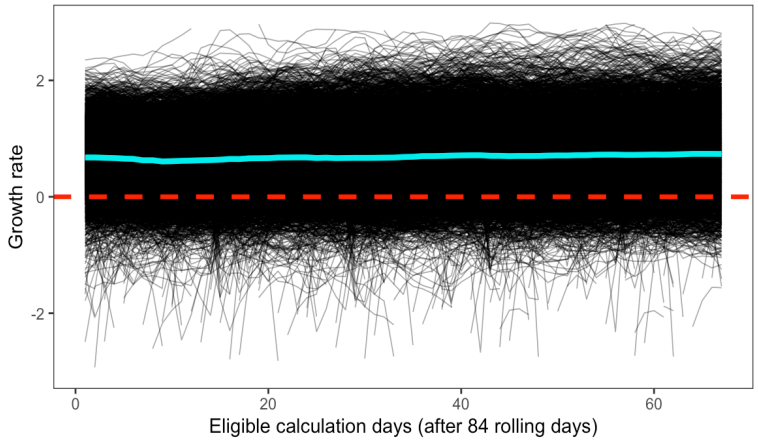

Figure 4.8: **[Trigger 7]** Simulated growth rate in the covered area

Fixed pandemic development time (days)	Proportion of all triggers activated (%)	Expected TCD	Expected IC	Expected payout for Class B(%)
100	56.44	82,722	184.34	100
120	80.10	688,172	191.37	100
150	95.20	19,175,134	192.58	100
180	98.94	701,691,669	195.65	100
200	100.00	5,722,654,940	195.91	100

Table 4.3: Evolution of trigger-activation probabilities versus number of days

From Table 4.3, we can see that given 10,000 simulations of for each trigger from trigger 3 through trigger 7[1], the proportion of all triggers that were activated, which is the estimated conditional probability $\mathbb{P}[X_B(t) = \ldots = X_G(t) = 1 | X_A(t) = 1]$, approaches to 100% as the development time for COVID-19 increases to 200 days.

At the same time, the expected payout ratio for Class B stays consistently at 100%. This result actually fits reality, where the confirmed number of death cases and number of infected countries exceeded the threshold of the 100% payout ratio of Class B at the very early stage of COVID-19 pandemic. Table 4.3 also demonstrates that the pandemic bond's payout timing is not as ideal as we hoped for. In a catastrophic pandemic such as the COVID-19, there is still an approximately 43.56% chance that payout gets delayed 15 days after the first available payout opportunity .

Table 4.4 displays estimated trigger risks utilising 10,000 Monte-Carlo simulations with the COVID-19 development time modelled using the exponential distribution. It shows that the the payout probability $\mathbb{P}[X_A(t) = 1 | \mathscr{F}_0]$, or equivalently the trigger risk h_0, is dominated by the conditional pandemic occurrence probability $\mathbb{P}[X_A(t) = 1 | \mathscr{F}_0]$. Even though $\mathbb{P}[X_B(t) = \ldots = X_G(t) = 1 | X_A(t) = 1]$ ranges from 88% to 100%, their changes are barely reflected in the final payout probability. This is because the payout probability is largely depend on the the pandemic occurrence probability.

Multiplying these estimated trigger risks (h_0) with the expected payout ratios L for each class of note, we obtain the trigger risk adjustments on the discount rate in Table 4.5.

4.4 Estimation results for the interest-rate model

4.4.1 Estimated parameters for the Vasiček model

Table 4.6 displays the estimated parameters using the MLE method stated in Section 3.3 for the Vasiček model. The numbers in parenthesis are the corresponding 95% confidence limits.

[1] Trigger 2 is automatically activated for development days longer than 84 days.

λ	$\mathbb{P}[X_A(t)=1\|\mathscr{F}_0]$ (%)	$\mathbb{P}[X_B(t)=1\ldots=X_G(t)=1\|X_A(t)=1]$ (%)	$\mathbb{P}[X_O(t)=1\|\mathscr{F}_0]$ (%)
$\frac{5}{36500}$	0.01389	98.78	0.01372
$\frac{10}{36500}$	0.02777	97.69	0.02713
$\frac{25}{36500}$	0.06942	93.82	0.06513
$\frac{50}{36500}$	0.13885	87.70	0.12177

Table 4.4: Estimated trigger risk

λ	Class of notes	h_0 (%)	L (%)	$h_0 L$ (%)
$\frac{5}{36500}$	A	0.01372	16.7	0.00229
	B	0.01372	100	0.01372
$\frac{10}{36500}$	A	0.02713	16.7	0.00453
	B	0.02713	100	0.02713
$\frac{25}{36500}$	A	0.06513	16.7	0.01088
	B	0.06513	100	0.06513
$\frac{50}{36500}$	A	0.12177	16.7	0.02121
	B	0.12177	100	0.12177

Table 4.5: Trigger-risk adjustment on the discount rate

4.4.2 Simulated interest-rate paths

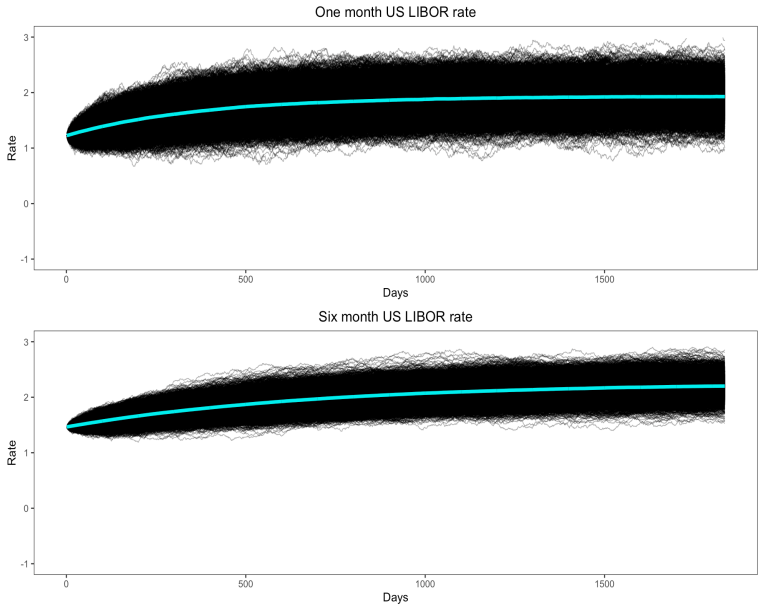

Figure 4.9: Simulated LIBOR paths

Parameters	One-month USD LIBOR (%)	Six-month USD LIBOR (%)
$\widehat{a}$	0.9719 (0.9133, 1.0304)	1.6969 (1.6493 ,1.7444)
$\widehat{b}$	1.9305 (1.5516, 2.3094)	2.2557 (2.0550, 2.4564)
$\widehat{\sigma}$	0.3453 (0.3288, 0.3619)	0.3194 (0.3042 ,0.3345)
r_0	1.2263	1.4654

Table 4.6: Estimated interest-rate model parameters.

λ	Class A	Class B
$\frac{5}{36500}$	1333.17 (1333.01, 1333.34)	1557.50 (1557.32, 1557.67)
$\frac{10}{36500}$	1333.17 (1333.01, 1333.34)	1557.50 (1557.31, 1557.66)
$\frac{25}{36500}$	1333.17 (1333.01, 1333.33)	1557.46 (1557.29, 1557.64)
$\frac{50}{36500}$	1333.16 (1333.00, 1333.32)	1557.42 (1557.24, 1557.59)

Table 4.7: Bond prices for a $1,000 par value

4.5 Computing the pandemic bond price

Table 4.7 depicts the bond pricing results. As λ increases, the price of the pandemic bond based on our model in Chapter 2 generally decreases. When λ increases by 10, the price of the Class-A note only decreases by about $0.01 whilst Class B decreases by around $0.08. This means that Class-A note is less sensitive to the frequency of the pandemic related to coronavirus, as expected. The change in λ only has impact on h_0 and does not affect the another key component in the trigger adjustment L. We note that L is primarily influenced by the total number of deaths (TCD), which is a fundamental variable and it is fixed in Class A notes. So, a change in λ can hardly affect the price of the Class-A note.

Chapter 5

Concluding remarks

5.1 Research implications

With respect to the COVID-19 early-stage data used in this project, both classes of notes in the pandemic bond should be priced at a premium on the issuance date. The bond price, especially the price of the Class-B note, is quite sensitive to two key variables: the frequency of the pandemic occurrences and the random total of deaths in the covered area.

From the modelling process, the WB's pandemic bond has the following limitations:

- *Additional modelling risk.*

  As discussed in Section 3.1, since the number of probable cases was not reported to the WHO as a regulated and well-defined figure, AIR would estimate this number for each country based on their own data and model as stated in (The International Bank for Reconstruction and Development, 2017). This is an additional modelling risk when it comes to calculating the trigger statistics.

- *Flawed assumption regarding the quality of reported data in some countries.*

  In the calculation of the trigger statistics , the numbers reported by the WHO are used as primary-source data. However, as an international coordinator who consolidates numbers reported from each country, WHO cannot validate the numbers reported from each country individually. In fact, due to various reasons such as political influence and lack of testing, the numbers reported during the COVID-19 pandemic may not reflect reality at all; and there is high likelihood that these numbers are under-reported or may be in some cases over-estimated. Certainly, this is a very complicated obstacle to overcome as it is linked to different country systems' resilience ability to deal with the pandemic. Needless to say, this is a real issue that needs to be addressed in order to make the bond more useful and efficient in transferring the pandemic risk.

- *Lack of transparency by the calculation agency.*

  The determination process of the triggers completely relies on the calculation agency AIR, but they never disclosed the calculated figures to the public as they had the con-

fidentiality agreement with WB. This information asymmetry makes it more difficult for outsiders to price this product when they want to trade it.

- *Payout criteria are too onerous.*

  The purpose of the pandemic bond is to provide funds to underdeveloped countries for the purpose of reducing damage caused by the pandemics, and obviously the best timing to do so is the early stage of the outbreak. From our modelling results for COVID-19 in Table 4.3, although the payout will be activated eventually given enough time to develop (200 days), there is a 43.56% chance that the triggers were not activated at the early stage (100 days). If the payout mechanism is activated too late (e.g., after 100 days), it may fail to achieve its goal. Without early intervention, the pandemic would probably be spreading in those underdeveloped/developing countries and compromising the populations' health that could have been mitigated if the payout were made possible early. Moreover, there is an additional 16-day gap (reporting window + calculation window) between the triggers' activation and payout declaration. This can further delay the payment and diminish the usability of the funding.

We elaborate on the imperfections of our approach to pandemic-bond valuation and the associated statistical modelling:

- *Bias in the COVID-19 data.*

  As mentioned in Chapter 2, this project focuses only on pricing the pandemic bond from the standpoint of the coronavirus. However, the COVID-19 is only a specific type of coronavirus. For a more accurate pricing, data on other coronaviruses such as SARS or MERS should be included too.

- *Reliability of data input.*

  As previously discussed, the numbers reported in the WHO's reports can not be validated. In particular, during the COVID-19 pandemic, the numbers reported from China have been doubtfully trustworthy; but this data constitute the majority (at least 80%) of the reported figures in the covered area from January to mid March in 2020.

  Of special note is the fact that the countries included in the covered area are all developing or underdeveloped nations. Thus, they may not have efficient and robust mechanisms to detect and report COVID-19 cases compared to Japan, New Zealand, Australia, and developed countries in Europe or North America. This uncertainty in data accuracy could have biased our data-driven modelling results.

5.2 Future directions

- *Utility of epidemiological models to capture the evolution of the four fundamental variables.*

  We recognise that the multivariate time-series models may not provide the best solution for modelling the long-term disease development. Time series models lose their

forecasting power as the modelling range gets longer (e.g., 6 month or 1 year). Using epidemiological stochastic models such as SIR (Suspected-Infected-Recovered) or SEIR (Suspected-Exposed-Infected-Recovered) models may allow for a more accurate behaviour modelling of the entire disease transmission. Nonetheless, more information must be collected for this modelling approach. For example, the SIR model requires the number of susceptible cases; yet, this number is not consolidated at the international level. As a result, it will take tremendous amount of effort to collect such information in each country and territory. Meanwhile, people can transmit the disease more easily through flights or public transport by land travels nowadays; so, it is even more difficult to track the susceptible cases on a global scale.

- *Constrain a VAR model to match the reality.*

  The pandemic bond's payout triggers were activated on 31 March 2020, which was 91 days after the start of the pandemic. We can constrain our model parameters such that they match this observation in which the payout is triggered with 100% certainty at the 91st day in the case of the COVID-19 pandemic.

- *Incorporate more data of the covered perils.*

  For an accurate pricing of the pandemic bond in a comprehensive perspective, more data of covered disease perils in both Class A and Class B notes should be taken into account.

- *Enrich the modelling structure with explicit correlation between the interest rate and pandemic encapsulated in the R_s process.*

  In our current model setting, we assumed that the pandemic events will have no direct impact on interest-rate movements. But, in fact, these two variables are related. The monetary policy used to stimulate the economy in each country during the COVID-19 pandemic has a direct nexus to the interest rate (Lilley & Rogoff, 2020). A decline in mortality or an increase in the life expectancy is also one of the drivers in the possible decline in the interest rate as stated in (Liu et al., 2013). In our bond-pricing equation (2.7), some effects of pandemic and mortality have been embedded in the trigger-risk adjusted discount rate R_s, and this adjustment term can be further expanded based on the explicit impact of pandemic and mortality on interest rates.

- *Use more flexible interest-rate models.*

  In this project, we assumed that the interest rate (LIBOR) dynamics is governed by the Vasiček model. However, the movements of LIBOR do not lead to constant "parameters" (a, b , σ). We can utilise flexible interest-rate models with dynamic parameter estimation (Zhou & Mamon, 2012) to improve the bond valuation 's accuracy.

References

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, *19*(6), 716–723.

Allard, R. (1998). Use of time-series analysis in infectious disease surveillance. *Bulletin of the World Health Organization*, *76*(4), 327.

Burnecki, K., & Giuricich, M. (2017). Stable weak approximation at work in index-linked catastrophe bond pricing. *Risks*, *5*(4), 64.

Dickey, D., & Fuller, W. (1981). Likelihood ratio statistics for autoregressive time series with a unit root. *Econometrica: Journal of the Econometric Society*, *49*(2), 1057–1072.

Doherty, N. (1997). Financial innovation in the management of catastrophe risk. *Journal of Applied Corporate Finance*, *10*(3), 84–95.

Duffie, D., & Singleton, K. (1999). Modeling term structures of defaultable bonds. *The Review of Financial Studies*, *12*(4), 687–720.

Harrison, C., & Williams, P. (2016). A systems approach to natural disaster resilience. *Simulation Modelling Practice and Theory*, *65*, 11–31.

Hofer, L., Gardoni, P., & Zanini, M. (2019). Risk-based cat bond pricing considering parameter uncertainties. *Sustainable and Resilient Infrastructure*, 1–15. `https://www.tandfonline.com/doi/full/10.1080/23789689.2019.1667116`, accessed 2020-06-23.

Hyman, J. (1983). Accurate monotonicity preserving cubic interpolation. *SIAM Journal on Scientific and Statistical Computing*, *4*(4), 645–654.

Jarrow, R. (2010). A simple robust model for cat bond valuation. *Finance Research Letters*, *7*(2), 72–79.

Kwiatkowski, D., Phillips, P. C., Schmidt, P., & Shin, Y. (1992). Testing the null hypothesis of stationarity against the alternative of a unit root. *Journal of Econometrics*, *54*(1-3), 159–178.

Lilley, A., & Rogoff, K. (2020). Negative interest rate policy in the post COVID-19 world. *VOX, CEPR Policy Portal*. `https://voxeu.org/article/negative-interest-rate-policy-post-covid-19-world?qt-quicktabs_cepr_policy_research=0`, accessed 2020-06-23.

Liu, X., Mamon, R., & Gao, H. (2013). A comonotonicity-based valuation method for guaranteed annuity options. *Journal of Computational and Applied Mathematics*, *250*, 58–69.

Ljung, G., & Box, G. (1978). On a measure of lack of fit in time series models. *Biometrika*, *65*(2), 297–303.

Mariani, M., & Amoruso, P. (2016). The effectiveness of catastrophe bonds in portfolio diversification. *International Journal of Economics and Financial Issues*, *6*(4), 1760–1767.

Niehaus, G., & Mann, S. (1992). The trading of underwriting risk: An analysis of insurance futures contracts and reinsurance. *Journal of Risk and Insurance*, *59*(4), 601–627.

Nowak, P., & Romaniuk, M. (2013). Pricing and simulations of catastrophe bonds. *Insurance: Mathematics and Economics*, *52*(1), 18–28.

Roser, M., Ritchie, H., Ortiz-Ospina, E., & Hasell, J. (2020). Coronavirus pandemic (COVID-19). *Our World in Data*. https://ourworldindata.org/coronavirus, accessed 2020-05-30.

The Conversation. (2020). *Pandemic bonds: The financial cure we need for COVID-19?* https://theconversation.com/pandemic-bonds-the-financial-cure-we-need-for-covid-19-134503, accessed 2020-05-30.

The Guardian. (2020). *World bank's $500 m pandemic scheme accused of 'waiting for people to die'*. https://www.theguardian.com/global-development/2020/feb/28/world-banks-500m-coronavirus-push-too-late-for-poor-countries-experts-say, accessed 2020-05-30.

The International Bank for Reconstruction and Development. (2017). *Prospectus Supplement*. http://pubdocs.worldbank.org/en/882831509568634367/PEF-Final-Prospectus-PEF.pdf, accessed 2020-05-06.

Tsay, R. (2013). *Multivariate Time Series Analysis: With R and Financial Applications*. Hoboken, New Jersey: John Wiley & Sons.

Vasiček, O. (1977). An equilibrium characterization of the term structure. *Journal of Financial Economics*, *5*(2), 177–188.

Young, V. (2004). Pricing in an incomplete market with an affine term structure. *Mathematical Finance: An International Journal of Mathematics, Statistics and Financial Economics*, *14*(3), 359–381.

Zhou, N., & Mamon, R. (2012). An accessible implementation of interest rate models with Markov-switching. *Expert Systems with Applications*, *39*(5), 4679–4689.

Appendix A

Constructed data set

This Appendix displays the four underlying variables we constructed based on the source data of COVID-19 from the Our World in Data in Section 3.1. The fourth column "Estimated number of total cases in the covered area" was estimated using our heuristic method mentioned in equations (3.1) and (3.2). Their plots are displayed in Section 4.1 on page 27.

Table A.1: COVID-19 data constructed for bond pricing

Date	Infected worldwide country number	Total confirmed deaths in the covered area	Total confirmed cases in the covered area	Estimated total cases in the covered area
2019-12-31	0	0	27	63
2020-01-01	0	0	27	63
2020-01-02	0	0	27	63
2020-01-03	0	0	44	63
2020-01-04	0	0	44	63
2020-01-05	0	0	59	63
2020-01-06	0	0	59	63
2020-01-07	0	0	59	63
2020-01-08	0	0	59	63
2020-01-09	0	0	59	63
2020-01-10	0	0	59	63
2020-01-11	0	1	59	63
2020-01-12	0	1	59	63
2020-01-13	0	1	60	63
2020-01-14	0	1	60	5296
2020-01-15	0	2	60	5296
2020-01-16	0	2	60	5296
2020-01-17	0	2	65	5296
2020-01-18	0	2	82	5296
2020-01-19	0	3	218	5296

Date	Infected worldwide country number	Total confirmed deaths in the covered area	Total confirmed cases in the covered area	Estimated total cases in the covered area
2020-01-20	0	3	237	5296
2020-01-21	0	6	389	5296
2020-01-22	0	17	531	5296
2020-01-23	0	17	628	5296
2020-01-24	1	26	889	5296
2020-01-25	1	41	1337	5296
2020-01-26	1	56	2003	5296
2020-01-27	1	81	2795	5296
2020-01-28	1	106	4558	6028
2020-01-29	1	132	6028	6028
2020-01-30	1	170	7770	7770
2020-01-31	1	213	9755	9755
2020-02-01	1	259	11858	43052
2020-02-02	1	305	14452	43052
2020-02-03	1	362	17265	43052
2020-02-04	1	427	20504	43052
2020-02-05	1	493	24386	43052
2020-02-06	1	565	28116	43052
2020-02-07	1	638	31285	43052
2020-02-08	1	724	34713	43052
2020-02-09	1	813	37323	43052
2020-02-10	1	910	40299	43052
2020-02-11	1	1018	42790	43052
2020-02-12	1	1115	44819	44819
2020-02-13	1	1369	59961	59961
2020-02-14	1	1382	64118	64118
2020-02-15	1	1525	66660	66660
2020-02-16	1	1667	68668	89499
2020-02-17	1	1773	70722	89499
2020-02-18	1	1871	72615	89499
2020-02-19	1	2010	74365	89499
2020-02-20	1	2124	74763	89499
2020-02-21	1	2242	75657	89499
2020-02-22	1	2353	76499	89499
2020-02-23	1	2452	77156	89499
2020-02-24	1	2605	77391	89499
2020-02-25	1	2679	77930	89499
2020-02-26	1	2734	78385	89499
2020-02-27	1	2767	78876	89499

Date	Infected worldwide country number	Total confirmed deaths in the covered area	Total confirmed cases in the covered area	Estimated total cases in the covered area
2020-02-28	2	2818	79319	89499
2020-02-29	3	2873	79908	89499
2020-03-01	3	2918	80705	90900
2020-03-02	4	2971	81326	92551
2020-03-03	4	3013	81866	93860
2020-03-04	4	3061	82863	95355
2020-03-05	4	3107	83619	96788
2020-03-06	4	3154	84449	98252
2020-03-07	4	3199	85872	99654
2020-03-08	4	3249	86980	101212
2020-03-09	5	3326	87924	103610
2020-03-10	6	3378	88537	105230
2020-03-11	6	3465	89785	107940
2020-03-12	6	3546	91034	110464
2020-03-13	6	3632	92300	113143
2020-03-14	7	3744	94033	116632
2020-03-15	8	3857	95964	120152
2020-03-16	9	3994	98023	124419
2020-03-17	9	4134	100194	128781
2020-03-18	9	4297	102388	133858
2020-03-19	10	4471	104929	139279
2020-03-20	12	4652	107747	144917
2020-03-21	13	4848	111274	151023
2020-03-22	17	5042	115239	157066
2020-03-23	20	5245	119338	163390
2020-03-24	23	5446	123746	169651
2020-03-25	25	5647	128969	175913
2020-03-26	27	5870	135218	182859
2020-03-27	28	6163	142648	191987
2020-03-28	32	6443	151868	200709
2020-03-29	34	6751	161161	210304
2020-03-30	39	7050	170404	219618
2020-03-31	41	7366	179942	229462
2020-04-01	46	7764	191550	241860
2020-04-02	48	8246	203777	256875
2020-04-03	49	8689	215725	270675
2020-04-04	53	9081	226175	282887
2020-04-05	56	9803	241653	305378
2020-04-06	58	10343	255753	322200

Date	Infected worldwide country number	Total confirmed deaths in the covered area	Total confirmed cases in the covered area	Estimated total cases in the covered area
2020-04-07	61	10850	268734	337994
2020-04-08	63	11484	283946	357744
2020-04-09	66	12138	301597	378117
2020-04-10	67	12849	318735	400265
2020-04-11	70	13547	338533	422009
2020-04-12	73	14225	356479	443130
2020-04-13	73	14964	374142	466151
2020-04-14	75	15575	390771	485184
2020-04-15	75	16439	411209	512099
2020-04-16	75	17229	432914	536709
2020-04-17	76	19308	454956	601473
2020-04-18	77	20162	478642	628076
2020-04-19	77	21029	500775	655084
2020-04-20	79	21813	524336	679507
2020-04-21	79	22541	547259	702185
2020-04-22	80	23445	571456	730346
2020-04-23	80	24329	594238	757884
2020-04-24	82	25417	620271	791777
2020-04-25	82	26585	645797	828162
2020-04-26	83	27655	675838	861494
2020-04-27	86	28404	712497	884826
2020-04-28	86	29501	737466	918999
2020-04-29	87	30975	767793	964917
2020-04-30	87	32304	798654	1006317
2020-05-01	87	33602	833571	1046751
2020-05-02	89	34961	869697	1089086
2020-05-03	89	36415	904316	1134380
2020-05-04	89	37518	941408	1168740
2020-05-05	89	38684	981266	1205063
2020-05-06	90	40244	1020056	1253659
2020-05-07	92	41868	1062749	1304249
2020-05-08	95	43493	1109407	1354870
2020-05-09	95	45173	1153981	1407205
2020-05-10	97	46812	1200087	1458262
2020-05-11	97	48462	1241710	1509662
2020-05-12	97	49610	1280032	1545424
2020-05-13	99	51775	1327219	1612867
2020-05-14	100	53606	1376830	1669905
2020-05-15	101	55485	1429117	1728438

Date	Infected worldwide country number	Total confirmed deaths in the covered area	Total confirmed cases in the covered area	Estimated total cases in the covered area
2020-05-16	101	57669	1485525	1796473
2020-05-17	102	59600	1538962	1856627
2020-05-18	104	61054	1586478	1901921
2020-05-19	104	62748	1638326	1954691
2020-05-20	105	65128	1698546	2028832
2020-05-21	105	67294	1762656	2096306
2020-05-22	105	69859	1828978	2176209
2020-05-23	105	72324	1895081	2252998
2020-05-24	107	74370	1959966	2316733
2020-05-25	107	76066	2021300	2369566
2020-05-26	107	78077	2079448	2432212
2020-05-27	108	80641	2142367	2512084
2020-05-28	108	83264	2212127	2593794
2020-05-29	108	85930	2293464	2676844

Appendix B

Lists of Covered Area and Worldwide Area

This Appendix outlines the detailed list of "worldwide" and "covered" area introduced in Subsection 2.1.1. The first columns in Tables B.1 and B.2 are the original area (territories) defined in the Prospectus Supplement of the pandemic bond, which are categorised by political entities. The second columns are territories stated in the WHO report format, which further divide those political entities into smaller territories (i.e., splitted territories) by geography and autonomy. The third columns include the country labels of the second columns for tracking the nationality of those splited territories. Column 1's should be viewed independently as they are not necessarily row matching with column 2's and 3's.

Table B.1: List of the worldwide area

Defined worldwide areas (territories)	**Splitted worldwide areas (territories) [formats in WHO Reports]**	**Splitted World areas (territories) country label**
Afghanistan	Afghanistan	Afghanistan
Albania	Albania	Albania
Algeria	Algeria	Algeria
Andorra	Andorra	Andorra
Angola	Angola	Angola
Antigua and Barbuda	Antigua and Barbuda	Antigua and Barbuda
Argentina	Argentina	Argentina
Armenia	Armenia	Armenia
Australia (including, but not limited to, Norfolk Island,Cocos (Keeling Islands), and Christmas Island)	Australia	Australia

Defined worldwide areas (territories)	**Splitted worldwide areas (territories) [formats in WHO Reports]**	**Splitted World areas (territories) country label**
Austria	Norfolk Island	Australia
Azerbaijan	Cocos (Keeling Islands)	Australia
Bahamas	Christmas Island	Australia
Bahrain	Austria	Austria
Bangladesh	Azerbaijan	Azerbaijan
Barbados	Bahamas	Bahamas
Belarus	Bahrain	Bahrain
Belgium	Bangladesh	Bangladesh
Belize	Barbados	Barbados
Benin	Belarus	Belarus
Bhutan	Belgium	Belgium
Bolivia	Belize	Belize
Bosnia and Herzegovina	Benin	Benin
Botswana	Bhutan	Bhutan
Brazil	Bolivia	Bolivia
Brunei	Bosnia and Herzegovina	Bosnia and Herzegovina
Bulgaria	Botswana	Botswana
Burkina Faso	Brazil	Brazil
Burma (Myanmar)	Brunei	Brunei
Burundi	Bulgaria	Bulgaria
Cambodia	Burkina Faso	Burkina Faso
Cameroon	Myanmar	Myanmar
Canada	Burundi	Burundi
Cape Verde	Cambodia	Cambodia
Central African Republic	Cameroon	Cameroon
Chad	Canada	Canada
Chile	Cape Verde	Cape Verde

Defined worldwide areas (territories)	**Splitted worldwide areas (territories) [formats in WHO Reports]**	**Splitted World areas (territories) country label**
China (including Hong Kong SAR, Macau SAR, and Taiwan)	Central African Republic	Central African Republic
Colombia	Chad	Chad
Comoros	Chile	Chile
Congo, Dem. Rep. of	China	China
Congo, Rep. of	Hong Kong	China
Cook Islands	Macau	China
Costa Rica	Taiwan	China
Cte d'Ivoire	Colombia	Colombia
Croatia	Comoros	Comoros
Cuba	Congo, Dem. Rep. of	Congo, Dem. Rep. of
Cyprus	Congo, Rep. of	Congo, Rep. of
Czech Republic	Cook Islands	Cook Islands
Denmark (including but not limited to Faroe Islands and Greenland)	Costa Rica	Costa Rica
Djibouti	Cte d'Ivoire	Cte d'Ivoire
Dominica	Croatia	Croatia
Dominican Republic	Cuba	Cuba
East Timor	Cyprus	Cyprus
Ecuador	Czech Republic	Czech Republic
Egypt	Denmark	Denmark
El Salvador	Faroe Islands	Denmark
Equatorial Guinea	Greenland	Denmark
Eritrea	Djibouti	Djibouti
Estonia	Dominica	Dominica
Ethiopia	Dominican Republic	Dominican Republic
Fiji	East Timor	East Timor

Defined worldwide areas (territories)	**Splitted worldwide areas (territories) [formats in WHO Reports]**	**Splitted World areas (territories) country label**
Finland	Ecuador	Ecuador
France (including, but not limited to, French Polynesia, French Guiana, Wallis and Futuna, St. Martin, Reunion, Mayotte, Martinique, New Caledonia, Guadeloupe, and Saint Pierre and Miquelon)	Egypt	Egypt
Gabon	El Salvador	El Salvador
Gambia	Equatorial Guinea	Equatorial Guinea
Georgia	Eritrea	Eritrea
Germany	Estonia	Estonia
Ghana	Ethiopia	Ethiopia
Greece	Fiji	Fiji
Grenada	Finland	Finland
Guatemala	France	France
Guinea	French Polynesia	France
Guinea-Bissau	French Guiana	France
Guyana	Wallis and Futuna	France
Haiti	St. Martin	France
Honduras	Reunion	France
Hungary	Mayotte	France
Iceland	Martinique	France
India	New Caledonia	France
Indonesia	Guadeloupe	France
Iran	Saint Pierre and Miquelon	France
Iraq	Gabon	Gabon
Ireland	Gambia	Gambia

Defined worldwide areas (territories)	**Splitted worldwide areas (territories) [formats in WHO Reports]**	**Splitted World areas (territories) country label**
Israel	Georgia	Georgia
Italy	Germany	Germany
Jamaica	Ghana	Ghana
Japan	Greece	Greece
Jordan	Grenada	Grenada
Kazakhstan	Guatemala	Guatemala
Kenya	Guinea	Guinea
Kiribati	Guinea-Bissau	Guinea-Bissau
Kosovo	Guyana	Guyana
Kuwait	Haiti	Haiti
Kyrgyzstan	Honduras	Honduras
Laos	Hungary	Hungary
Latvia	Iceland	Iceland
Lebanon	India	India
Lesotho	Indonesia	Indonesia
Liberia	Iran	Iran
Libya	Iraq	Iraq
Liechtenstein	Ireland	Ireland
Lithuania	Israel	Israel
Luxembourg	Italy	Italy
Macedonia	Jamaica	Jamaica
Madagascar	Japan	Japan
Malawi	Jordan	Jordan
Malaysia	Kazakhstan	Kazakhstan
Maldives	Kenya	Kenya
Mali	Kiribati	Kiribati
Malta	Kosovo	Kosovo
Marshall Islands	Kuwait	Kuwait
Mauritania	Kyrgyzstan	Kyrgyzstan

Defined worldwide areas (territories)	**Splitted worldwide areas (territories) [formats in WHO Reports]**	**Splitted World areas (territories) country label**
Mauritius	Laos	Laos
Mexico	Latvia	Latvia
Micronesia	Lebanon	Lebanon
Moldova	Lesotho	Lesotho
Monaco	Liberia	Liberia
Mongolia	Libya	Libya
Montenegro	Liechtenstein	Liechtenstein
Morocco	Lithuania	Lithuania
Mozambique	Luxembourg	Luxembourg
Namibia	Macedonia	Macedonia
Nauru	Madagascar	Madagascar
Nepal	Malawi	Malawi
Netherlands (including, but not limited to, Aruba, Sint Maarten, Curacao, and Netherlands Antilles)	Malaysia	Malaysia
New Zealand (including, but not limited to, Tokelau)	Maldives	Maldives
Nicaragua	Mali	Mali
Niger	Malta	Malta
Nigeria	Marshall Islands	Marshall Islands
Niue	Mauritania	Mauritania
North Korea	Mauritius	Mauritius
Norway (including, but not limited to, Svalbard)	Mexico	Mexico
Oman	Micronesia	Micronesia
Pakistan	Moldova	Moldova
Palau	Monaco	Monaco
Panama	Mongolia	Mongolia
Papua New Guinea	Montenegro	Montenegro

Defined worldwide areas (territories)	**Splitted worldwide areas (territories) [formats in WHO Reports]**	**Splitted World areas (territories) country label**
Paraguay	Morocco	Morocco
Peru	Mozambique	Mozambique
Philippines	Namibia	Namibia
Poland	Nauru	Nauru
Portugal	Nepal	Nepal
Qatar	Netherlands	Netherlands
Romania	Aruba	Netherlands
Russia	Sint Maarten	Netherlands
Rwanda	Curacao	Netherlands
Samoa	Netherlands Antilles	Netherlands
San Marino	New Zealand	New Zealand
So Tom and Prncipe	Tokelau	New Zealand
Saudi Arabia	Nicaragua	Nicaragua
Senegal	Niger	Niger
Serbia	Nigeria	Nigeria
Seychelles	Niue	Niue
Sierra Leone	North Korea	North Korea
Singapore	Norway	Norway
Slovakia	Svalbard	Norway
Slovenia	Oman	Oman
Solomon Islands	Pakistan	Pakistan
Somalia	Palau	Palau
South Africa	Panama	Panama
South Korea	Papua New Guinea	Papua New Guinea
South Sudan	Paraguay	Paraguay
Spain	Peru	Peru
Sri Lanka	Philippines	Philippines
St. Kitts and Nevis	Poland	Poland

Defined worldwide areas (territories)	**Splitted worldwide areas (territories) [formats in WHO Reports]**	**Splitted World areas (territories) country label**
St. Lucia	Portugal	Portugal
St. Vincent and the Grenadines	Qatar	Qatar
Sudan	Romania	Romania
Suriname	Russia	Russia
Swaziland	Rwanda	Rwanda
Sweden	Samoa	Samoa
Switzerland	San Marino	San Marino
Syria	So Tom and Prncipe	So Tom and Prncipe
Tajikistan	Saudi Arabia	Saudi Arabia
Tanzania	Senegal	Senegal
Thailand	Serbia	Serbia
Togo	Seychelles	Seychelles
Tonga	Sierra Leone	Sierra Leone
Trinidad and Tobago	Singapore	Singapore
Tunisia	Slovakia	Slovakia
Turkey	Slovenia	Slovenia
Turkmenistan	Solomon Islands	Solomon Islands
Tuvalu	Somalia	Somalia
Uganda	South Africa	South Africa
Ukraine	South Korea	South Korea
United Arab Emirates	South Sudan	South Sudan
United Kingdom (including, but not, limited to, Anguilla, Bermuda, British Virgin Islands, Cayman Islands, Gibraltar, Malvinas, Montserrat, Pitcairn, Saint Helena, Turks and, Caicos Islands, Jersey, Guernsey, and, Isle of Man)	Spain	Spain

Defined worldwide areas (territories)	**Splitted worldwide areas (territories) [formats in WHO Reports]**	**Splitted World areas (territories) country label**
United States (including, but not limited to, American Samoa, Guam, Puerto Rico, Northern Mariana Islands, and United States Virgin Islands)	Sri Lanka	Sri Lanka
Uruguay	St. Kitts and Nevis	St. Kitts and Nevis
Uzbekistan	St. Lucia	St. Lucia
Vanuatu	St. Vincent and the Grenadines	St. Vincent and the Grenadines
Venezuela	Sudan	Sudan
Vietnam	Suriname	Suriname
Yemen	Swaziland	Swaziland
Zambia	Sweden	Sweden
Zimbabwe	Switzerland	Switzerland
	Syria	Syria
	Tajikistan	Tajikistan
	Tanzania	Tanzania
	Thailand	Thailand
	Togo	Togo
	Tonga	Tonga
	Trinidad and Tobago	Trinidad and Tobago
	Tunisia	Tunisia
	Turkey	Turkey
	Turkmenistan	Turkmenistan
	Tuvalu	Tuvalu
	Uganda	Uganda
	Ukraine	Ukraine
	United Arab Emirates	United Arab Emirates

Defined worldwide areas (territories)	**Splitted worldwide areas (territories) [formats in WHO Reports]**	**Splitted World areas (territories) country label**
	United Kingdom	United Kingdom
	Anguilla	United Kingdom
	Bermuda	United Kingdom
	British Virgin Islands	United Kingdom
	Cayman Islands	United Kingdom
	Gibraltar	United Kingdom
	Malvinas	United Kingdom
	Montserrat	United Kingdom
	Pitcairn	United Kingdom
	Saint Helena	United Kingdom
	Turks and Caicos Islands	United Kingdom
	Jersey	United Kingdom
	Guernsey	United Kingdom
	Isle of Man	United Kingdom
	United States	United States
	American Samoa	United States
	Guam	United States
	Puerto Rico	United States
	Northern Mariana Islands	United States
	United States Virgin Islands	United States
	Uruguay	Uruguay
	Uzbekistan	Uzbekistan
	Vanuatu	Vanuatu
	Venezuela	Venezuela
	Vietnam	Vietnam
	Yemen	Yemen

Defined worldwide areas (territories)	**Splitted worldwide areas (territories) [formats in WHO Reports]**	**Splitted World areas (territories) country label**
	Zambia	Zambia
	Zimbabwe	Zimbabwe

Table B.2: List of the covered area

Defined covered area (territories)	**Covered area (territories) splitted** [**format in WHO reports**]	**Country labels of splitted covered area**
Afghanistan	Afghanistan	Afghanistan
Albania	Albania	Albania
Algeria	Algeria	Algeria
Angola	Angola	Angola
Antigua and Barbuda	Antigua and Barbuda	Antigua and Barbuda
Argentina	Argentina	Argentina
Armenia	Armenia	Armenia
Azerbaijan	Azerbaijan	Azerbaijan
Bangladesh	Bangladesh	Bangladesh
Belarus	Belarus	Belarus
Belize	Belize	Belize
Benin	Benin	Benin
Bhutan	Bhutan	Bhutan
Bolivia	Bolivia	Bolivia
Bosnia and Herzegovina	Bosnia and Herzegovina	Bosnia and Herzegovina
Botswana	Botswana	Botswana
Brazil	Brazil	Brazil
Bulgaria	Bulgaria	Bulgaria
Burkina Faso	Burkina Faso	Burkina Faso
Burma (Myanmar)	Myanmar	Myanmar
Burundi	Burundi	Burundi
Cambodia	Cambodia	Cambodia
Cameroon	Cameroon	Cameroon
Cape Verde	Cape Verde	Cape Verde
Central African Republic	Central African Republic	Central African Republic
Chad	Chad	Chad
Chile	Chile	Chile

Defined covered area (territories)	**Covered area (territories) splitted [format in WHO reports]**	**Country labels of splitted covered area**
China (including Hong Kong SAR, Macau SAR, and Taiwan)	China	China
Colombia	Hong Kong	China
Comoros	Macau	China
Congo, Dem. Rep. of	Taiwan	China
Congo, Rep. of	Colombia	Colombia
Costa Rica	Comoros	Comoros
Cte d'Ivoire	Congo, Dem. Rep. of	Congo, Dem. Rep. of
Croatia	Congo, Rep. of	Congo, Rep. of
Djibouti	Costa Rica	Costa Rica
Dominica	Cte d'Ivoire	Cte d'Ivoire
Dominican Republic	Croatia	Croatia
East Timor	Djibouti	Djibouti
Ecuador	Dominica	Dominica
Egypt	Dominican Republic	Dominican Republic
El Salvador	East Timor	East Timor
Equatorial Guinea	Ecuador	Ecuador
Eritrea	Egypt	Egypt
Ethiopia	El Salvador	El Salvador
Fiji	Equatorial Guinea	Equatorial Guinea
Gabon	Eritrea	Eritrea
Gambia	Ethiopia	Ethiopia
Georgia	Fiji	Fiji
Ghana	Gabon	Gabon
Grenada	Gambia	Gambia
Guatemala	Georgia	Georgia
Guinea	Ghana	Ghana
Guinea-Bissau	Grenada	Grenada

Defined covered area (territories)	**Covered area (territories) splitted [format in WHO reports]**	**Country labels of splitted covered area**
Guyana	Guatemala	Guatemala
Haiti	Guinea	Guinea
Honduras	Guinea-Bissau	Guinea-Bissau
India	Guyana	Guyana
Indonesia	Haiti	Haiti
Iran	Honduras	Honduras
Iraq	India	India
Jamaica	Indonesia	Indonesia
Jordan	Iran	Iran
Kazakhstan	Iraq	Iraq
Kenya	Jamaica	Jamaica
Kiribati	Jordan	Jordan
Kosovo	Kazakhstan	Kazakhstan
Kyrgyzstan	Kenya	Kenya
Laos	Kiribati	Kiribati
Lebanon	Kosovo	Kosovo
Lesotho	Kyrgyzstan	Kyrgyzstan
Liberia	Laos	Laos
Libya	Lebanon	Lebanon
Macedonia	Lesotho	Lesotho
Madagascar	Liberia	Liberia
Malawi	Libya	Libya
Malaysia	Macedonia	Macedonia
Maldives	Madagascar	Madagascar
Mali	Malawi	Malawi
Marshall Islands	Malaysia	Malaysia
Mauritania	Maldives	Maldives
Mauritius	Mali	Mali
Mexico	Marshall Islands	Marshall Islands

Defined covered area (territories)	**Covered area (territories) splitted [format in WHO reports]**	**Country labels of splitted covered area**
Micronesia	Mauritania	Mauritania
Moldova	Mauritius	Mauritius
Mongolia	Mexico	Mexico
Montenegro	Micronesia	Micronesia
Morocco	Moldova	Moldova
Mozambique	Mongolia	Mongolia
Namibia	Montenegro	Montenegro
Nauru	Morocco	Morocco
Nepal	Mozambique	Mozambique
Nicaragua	Namibia	Namibia
Niger	Nauru	Nauru
Nigeria	Nepal	Nepal
Pakistan	Nicaragua	Nicaragua
Palau	Niger	Niger
Panama	Nigeria	Nigeria
Papua New Guinea	Pakistan	Pakistan
Paraguay	Palau	Palau
Peru	Panama	Panama
Philippines	Papua New Guinea	Papua New Guinea
Poland	Paraguay	Paraguay
Romania	Peru	Peru
Russia	Philippines	Philippines
Rwanda	Poland	Poland
Samoa	Romania	Romania
So Tom and Prncipe	Russia	Russia
Senegal	Rwanda	Rwanda
Serbia	Samoa	Samoa
Seychelles	So Tom and Prncipe	So Tom and Prncipe
Sierra Leone	Senegal	Senegal

Defined covered area (territories)	**Covered area (territories) splitted [format in WHO reports]**	**Country labels of splitted covered area**
Solomon Islands	Serbia	Serbia
Somalia	Seychelles	Seychelles
South Africa	Sierra Leone	Sierra Leone
South Sudan	Solomon Islands	Solomon Islands
Sri Lanka	Somalia	Somalia
St. Kitts and Nevis	South Africa	South Africa
St. Lucia	South Sudan	South Sudan
St. Vincent and the Grenadines	Sri Lanka	Sri Lanka
Sudan	St. Kitts and Nevis	St. Kitts and Nevis
Suriname	St. Lucia	St. Lucia
Swaziland	St. Vincent and the Grenadines	St. Vincent and the Grenadines
Syria	Sudan	Sudan
Tajikistan	Suriname	Suriname
Tanzania	Swaziland	Swaziland
Thailand	Syria	Syria
Togo	Tajikistan	Tajikistan
Tonga	Tanzania	Tanzania
Trinidad and Tobago	Thailand	Thailand
Tunisia	Togo	Togo
Turkey	Tonga	Tonga
Turkmenistan	Trinidad and Tobago	Trinidad and Tobago
Tuvalu	Tunisia	Tunisia
Uganda	Turkey	Turkey
Ukraine	Turkmenistan	Turkmenistan
Uruguay	Tuvalu	Tuvalu
Uzbekistan	Uganda	Uganda
Vanuatu	Ukraine	Ukraine
Venezuela	Uruguay	Uruguay
Vietnam	Uzbekistan	Uzbekistan

Defined covered area (territories)	**Covered area (territories) splitted [format in WHO reports]**	**Country labels of splitted covered area**
Yemen	Vanuatu	Vanuatu
Zambia	Venezuela	Venezuela
Zimbabwe	Vietnam	Vietnam
	Yemen	Yemen
	Zambia	Zambia
	Zimbabwe	Zimbabwe

Appendix C

List of major pandemics since 1700s

In this Appendix, we display the historical record of major non-flu pandemics around the globe since the 1700s. This record was used to estimate the pandemic's occurrence frequency λ in Subsection 3.2.1 on page 19.

Table C.1: Historical record of major non-flu pandemics

Name of the non-flu pandemic	**Start & end**	**Duration (no. of years)**
Cholera pandemic 1	1817-1824	7
Cholera pandemic 2	1829-1837	8
Cholera pandemic 3	1846-1860	14
Cholera pandemic 4	1863-1875	12
Cholera pandemic 5	1881 - 1896	15
Cholera pandemic 6	1899 - 1923	24
Cholera pandemic 7	1961 - 1975	14
Third plague	1885	1
Yellow fever	Late 1800s	1 to 2
HIV	1981 - present	39
SARS	2002 - 2003	1
EBOLA	2014-2016	2
MERS	2012 - present	8
COVID-19	2020 - present	1

Appendix D

Implementation codes

This Appendix includes all the implementation codes of Chapter 3.

D.1 Multivariate time-series modelling

The following R code imports the raw data presented in Appendix A, and model the four fundamental variables utilising the multivariate time series with bootstrap techniques described in Subsections 3.2.1 – 3.2.4.

```
## Modelling the underlying variables using VARMA
##Import the Basic data & pacakge

library(zoo)
library(xts)
library(forecast)
library(tseries)
library(aTSA)
library(MTS)
library(mvnTest)
library(normwhn.test)
library(fUnitRoots)
require(data.table)
require(VARtests)
require(portes)
require(mlVAR)
library(ggplot2)
library(reshape2)

##Import the data
raw.data <- read.csv(file = "~/Downloads/New.Triggers (2).csv")
raw.data <- ts(raw.data, frequency = 365, start = c(2020,1))
colnames(raw.data) <- c("counrty", "total.death",
```

```
                        "total.confirm", "total.case")
plot(raw.data)


##VAR and VARMA Model for modelling the probability of
##trigger2 to trigger 7 being activated


## Load the data and plot the graph
raw.data <- read.csv(file = "~/Downloads/New.Triggers (2).csv")
raw.data <- ts(raw.data, frequency = 365, start = c(2020,1))
colnames(raw.data) <- c("counrty", "total.death",
                        "total.confirm", "total.case")
plot(raw.data)


## Build Model Based on Original Data

par(mfrow=c(2,2))
plot(country, main = "Number of Affected Countries Worldwide",
     xlab = "Days since COVID-19 Start",
     ylab = "Number of Countries")

plot(death, main = "Number of Confirmed Deaths in Covered Area",
     xlab = "Days since COVID-19 Start",
     ylab = "Number of Confirmed Deaths")

plot(confirm, main = "Number of Confirmed Cases
     & Total Cases in Covered Area",
     xlab = "Days since COVID-19 Start",
     ylab = "Number of Confirmed Cases", col = "red")

plot(total, main = "Number of Total Cases in Covered Area",
     xlab = "Days since COVID-19 Start",
     ylab = "Number of Total Cases", col = "blue")


## Covert to time series structure
country <- ts(raw.data[,1])
death <- ts(raw.data[,2])
confirm <- ts(raw.data[,3])
total <- ts(raw.data[,4])


## Tansform the origianl time series to stable time series
```

```
matrix <- as.matrix(cbind(country, death, confirm))
plot(matrix)

log.xi1 = t(matrix[1,])

## Transform to increments, 1st difference
country <- diff(country)
death <- diff(death)
confirm <- diff(confirm)
total <- diff(total)

##Subsitute 0 values with 0.1 to avoid -Inf or NaN
death[death == 0] <- 0.1
confirm[confirm == 0] <- 0.1


s.matrix2 <- as.matrix(cbind(country,
                             log(confirm), log(death)))

##Use ADF test for test unit root
adf.test(log(confirm))
adf.test(log(death))
adf.test(country)

plot(s.matrix2)

##Log transformation to remove exponential trend
s.matrix2 <- as.matrix(cbind(country,
                             log(confirm),log(death)))
log.xi2 <- t(s.matrix2[1,])

# 2nd difference on country and log death & confirm cases
country <- diff(country)
death <- diff(log(death))
confirm <-diff(log(confirm))

## ADF test
adf.test(confirm)
adf.test(death)
adf.test(country)


s.matrix2 <- as.matrix(cbind(country, confirm, death))
plot(s.matrix2)
```

```
## Fit VAR model and VARMA model to the original time series
## VAR Model
VARorder(s.matrix2, maxp = 20)
model2<- VAR(s.matrix2, p = 9)
MTSdiag(model2)
MarchTest(model2$residuals)
DH.test(b.model2$residuals)

## Try another package to confirm
vars::VARselect(s.matrix2,lag.max = 20)
alter.model2 <- vars::VAR(s.matrix2, p =9)


## VARMA Model
Eccm(s.matrix2)
varma.model <- VARMA(s.matrix2, p = 1, q = 1)
b.varma.model <- refVARMA(varma.model)
MTSdiag(b.varma.model)
MarchTest(b.varma.model$residuals)
DH.test(b.varma.model$residuals)


## VAR model outperforms the VARMA

## Implementing the bootstraping to estimate the parameters in VAR model
##break the stationary time series vector into  subsamples
break.sample <- function(input, tot.length, sample.n){
  list = seq(1, tot.length, sample.n)
  l = tot.length/sample.n
  pool = list()
  index = 1
  for (num in list){
    temp1 = num + sample.n - 1
    value <- input[num:temp1,]
    pool[[index]] <- value
    index <- index + 1
  }
  return(pool)
}

## Divide the original time series into 7 blocks
broken.sample<-break.sample(s.matrix2, 147, 21)

sample.index <- seq(1, 7, 1)
```

```
## function apply to each indices lst
assamble <- function(vector, raw.lst) {
  int <- raw.lst[[1]][1,]
  N <- length(vector)
  for (each in vector){
    int <- rbind(int, raw.lst[[each]])
  }
  int <- int[-1,]
  return(ts(int))
}


## function apply to each lst
sample.ts <- lapply(resample,
                    function(i) assamble(i, broken.sample))

## Using VAR to model
models <- lapply(sample.ts, function(i) VAR(i, p = 9))


## function to calculate the residual covariance matrix
cov.res <- function(residuals){
  result <- matrix(c(cov(model2$residuals)),
                   nrow = ncol(residuals), ncol = ncol(residuals))
  return (result)
}

## find stand error
se <- function(x) sqrt(var(x)/length(x))


### Now the Bootstrapping
Bootstrap <- function(broken.sample, broken.size, resample.size){
  set.seed(39)
  sample.index <- seq(1, broken.size, 1)
  resample <- lapply(1:resample.size, function(i)
    sample(sample.index, size = 7, replace = T))
  ## reconstruct the time series
  sample.ts <- lapply(resample, function(i) assamble(i, broken.sample))
  ## Apply the VAR model to the resampled ts
  models <- lapply(sample.ts, function(i) VAR(i, p = 9))
  ## Extract the parameters
  ## parameters for lags phi1
  parameters <- lapply(models, '[', c('Phi'))
```

```
avg.parameters <- round(apply(array(unlist(parameters),
                        c(3, 27, resample.size)), c(1,2), mean),8)
## Extract the constants
ph0 <- lapply(models, '[', c('Ph0'))
avg.ph0 <- round(apply(array(unlist(ph0),
                        c(3, 1, resample.size)), c(1,2), mean),8)
## Apply variance for the lag parameters
variance <- round(apply(array(unlist(parameters),
                        c(3, 27, resample.size)), c(1,2), var),8)
## Calculate the Standard Errors
se <- sqrt(variance) / sqrt(resample.size)
## Apply variance for the constant parameters
ph0.var <- round(apply(array(unlist(ph0),
                        c(3, 1, resample.size)), c(1,2), var),8)
ph0.se <- sqrt(ph0.var) / sqrt(resample.size)
## Extract the residuals
residuals <- lapply(models, '[', c('residuals'))
## Extract the avg.residuals
avg.residuals <- round(apply(array(unlist(residuals),
                       c(140, 3, resample.size)), c(1,2), mean),8)
## Calcualte the Ljung-Box Statistics lag 14
LB.Stats <- lapply(residuals, function(i)
  LjungBox(i,lags=seq(5,30,5),
           order=81,season=1,squared.residuals=FALSE))
## Extract the avergaed Ljung-Box stats
avg.LB <- round(apply(array(unlist(LB.Stats),
                            c(6, 4, resample.size)),
                      c(1,2), mean),8)
## Calculate the Heterdasity statistics
Hetero.Stats <- lapply(residuals,
                       function(i) LjungBox(i,lags=seq(5,30,5),
                       order=81,season=1,squared.residuals=TRUE))
## Extract the averaged Hetero stats
avg.Hetero <- round(apply(array(unlist(Hetero.Stats),
                          c(6, 4, resample.size)),
                          c(1,2), mean),8)
## Calculate the residual covraince martix
mannual.res <- cov.res(avg.residuals)
## residual.covariance matrices
res.cov <- lapply(models, '[', c('Sigma'))
## Extract the averaged residual covariance
avg.res.cov<- round(apply(array(unlist(res.cov),
                          c(3, 3, resample.size)), c(1,2), mean),8)
## Return the results
result <- list(avg.parameters, se, avg.ph0, ph0.se,
```

```
                avg.res.cov, mannual.res, avg.LB,
                avg.Hetero, LB.Stats, Hetero.Stats)
  names(result) <- c("avg.phi", 'phi.se', 'ph0', 'ph0.se',
                     'avg.residal.cov', 'manual.avg.res.cov', 'avg.LB',
                     'avg.Hetero', 'LB.Stats', "Hetero.Stats")
  return(result)
}



##Record the bootstrap results
set.seed(251121253)
asy.model <- Bootstrap(broken.sample, 7, 1000)
AC.P.value <- asy.model$LB.Stats
ARCH.P.value <- asy.model$Hetero.Stats


## Extract P-values for model validation
plot.p.values <- function (LB.statistics.lst){
  lag5 <- lapply(LB.statistics.lst, function(i) return (i[1,4]))
  lag5.mean <- mean(unlist(lag5))
  lag5.se <- se(unlist(lag5))
  lag10 <- lapply(LB.statistics.lst, function(i) return (i[2,4]))
  lag10.mean <- mean(unlist(lag10))
  lag10.se <- se(unlist(lag10))
  lag15 <- lapply(LB.statistics.lst, function(i) return (i[3,4]))
  lag15.mean <- mean(unlist(lag15))
  lag15.se <- se(unlist(lag15))
  lag20 <- lapply(LB.statistics.lst, function(i) return (i[4,4]))
  lag20.mean <- mean(unlist(lag20))
  lag20.se <- se(unlist(lag20))
  lag25 <- lapply(LB.statistics.lst, function(i) return (i[5,4]))
  lag25.mean <- mean(unlist(lag25))
  lag25.se <- se(unlist(lag25))
  lag30 <- lapply(LB.statistics.lst, function(i) return (i[6,4]))
  lag30.mean <- mean(unlist(lag30))
  lag30.se <- se(unlist(lag30))
  y1 <- unlist(lag5)
  y2 <- unlist(lag10)
  y3 <- unlist(lag15)
  y4 <- unlist(lag20)
  y5 <- unlist(lag25)
  y6 <- unlist(lag30)
  x = seq(0,1,0.01)
  plot(lag5.kernel <- density(y1, kernel = "gaussian"),
```

```
      main = "Distribution of P-Value for Ljung-Box of Lag 5")
  plot(lag10.kernel <- density(y2, kernel = "gaussian"),
      main = "Distribution of P-Value for Ljung-Box of Lag 10")
  plot(lag15.kernel <- density(y3, kernel = "gaussian"),
      main = "Distribution of P-Value for Ljung-Box of Lag 15")
  plot(lag20.kernel <- density(y3, kernel = "gaussian"),
      main = "Distribution of P-Value for Ljung-Box of Lag 20")
  plot(lag25.kernel <- density(y3, kernel = "gaussian"),
      main = "Distribution of P-Value for Ljung-Box of Lag 25")
  plot(lag30.kernel <- density(y3, kernel = "gaussian"),
      main = "Distribution of P-Value for Ljung-Box of Lag 30")
  result <- matrix(c(lag5.mean, lag10.mean,
                     lag15.mean, lag20.mean,
                     lag25.mean, lag30.mean,
                     lag5.se, lag10.se,
                     lag15.se, lag20.se,
                     lag25.se, lag30.se),
                   nrow = 6, ncol = 2)
  rownames(result) <- c("lag5", "lag10", "lag15",
                        "lag20", "lag25", "lag30")
  colnames(result) <- c("LB P-Value mean", "LB P-Value SE")
  return(result)
}


##Plot the P-value distribution in serial correlation
##in residuals and the ARCH effect in residuals
plot.p.values(AC.P.value)
plot.p.values(ARCH.P.value)
```

D.2 Trigger-risk estimation

The following R code uses the parameters estimated from VAR(9) model and bootstrap technique to simulate the four underlying variables and estimate the trigger-risk model parameters as described in Subsection 3.2.5.

```
## Simulate data from the estimated parameters
## of the bootstrap results
Extract_para <- function(input){
  recorder = list()
  N = length(input)/9
  i = 1
  j = 1
  while (i <= N) {
```

```
    temp1 <- j
    temp2 <- j + 2
    recorder[[i]] <- input[,temp1:temp2]
    i <- i + 1
    j <- j + 3
  }
  return(recorder)
}


para<- Extract_para(asy.model$avg.phi)
means <- as.vector(t(asy.model$ph0))
lg <- 9
Nt <- 5000
inits <- s.matrix2[1:9,]
resi.cov = asy.model$manual.avg.res.cov


##Generate data and recover to original numbers
Data.Generate <- function(para, mean, lag,
                          N, init, residual.cov, xi1, xi2){
  stable.data <- simulateVAR(para, mean = mean,
                             lag = lag, N = N,
                             init = init,
                             residuals = residual.cov,
                             burnin = 0)
  stable.data <- as.matrix(stable.data)
  ## Reverse 2nd difference
  stable.data <- diffinv(stable.data, xi = xi2)
  ## Reverse log
  stable.data <- cbind(stable.data[,1],
                       exp(stable.data[,2]),
                       exp(stable.data[,3]))
  ## Reverse 0s
  stable.data[stable.data == 0.1] <- 0
  ## Reverse the 1st difference
  stable.data <- diffinv(stable.data, xi = xi1)
  return(stable.data)
}

# Test
b <- Data.Generate(para, mean = means, lag = lg, N = Nt,
                   init = inits, residual.cov = resi.cov,
                   xi1 = log.xi1, xi2 = log.xi2)
```

```
## Define the function to recover the total
##number of cases from the total confirmed
## death
Total.recover <- function(data, death.rate){
  confirm <- data[,2]
  death <- data[,3]
  total <- death / death.rate
  total <- pmax(confirm, total)
  result <- cbind(data, total)
  return(result)
}


## Apply 14-day averaging windows
linear.smooth <-function(data){
  confirm <- data[,2]
  death <- data[,3]
  total <- data[,4]
  N = length(confirm)
  smoothed.confirm <- frollmean(confirm, 14, fill=NA,
                                algo="exact",
                                align="center",
                                na.rm=FALSE, hasNA=NA,
                                adaptive=FALSE)
  smoothed.confirm[1:6] <- confirm[1:6]
  smoothed.confirm <- smoothed.confirm[1:(N - 7)]
  smoothed.death <- frollmean(death, 14, fill=NA,
                              algo="exact",
                              align="center",
                              na.rm=FALSE, hasNA=NA,
                              adaptive=FALSE)
  smoothed.death[1:6] <- death[1:6]
  smoothed.death <- smoothed.death[1:(N-7) ]
  smoothed.tot <- frollmean(total, 14, fill=NA,
                            algo="exact",
                            align="center",
                            na.rm=FALSE, hasNA=NA,
                            adaptive=FALSE)
  smoothed.tot[1:6] <- total[1:6]
  smoothed.tot <- smoothed.tot[1:(N-7)]
  result <- cbind(data[,1], smoothed.confirm,
                  smoothed.death, smoothed.tot)
  return(result)
}
```

```
##Function to calculate the triggers

## Function to calculate the rolling cases
rolling <- function (input_vector){
  indicator = 1
  recorder = vector()
  while (indicator <= length(input_vector)){
    if (indicator - 84 <= 0){
      recorder <- c(recorder, input_vector[indicator])
      indicator = indicator + 1
    } else {
      temp1 = input_vector[indicator] -
        input_vector[indicator - 84]
      recorder <- c(recorder, temp1)
      indicator = indicator + 1
    }
  }
  return(recorder)
}

##Function to calculate the confirmation ratio
confirmation.ratio <- function(rolling.confirm,
                               rolling.total) {
  CR = rolling.confirm / pmin(rolling.total, 750)
  return (CR)
}

##Function to calculate the growth rate
growth.rate <- function(rolling.total){
  N = length(rolling.total)
  NCRC1 = vector()
  NCRC2 = vector()
  NCRC3 = vector()
  NCRC4 = vector()
  NCRC5 = vector()
  start = 85
  while (start <= N){
    temp1 = rolling.total[start] - rolling.total[start - 14]
    temp2 = rolling.total[start-14] - rolling.total[start-28]
    temp3 = rolling.total[start-28] - rolling.total[start-42]
    temp4 = rolling.total[start-42] - rolling.total[start-56]
    temp5 = rolling.total[start-56] - rolling.total[start-70]
```

```
    temp6 = rolling.total[start-70] - rolling.total[start-84]
    NCRC1 <- c(NCRC1, log(temp1/temp2))
    NCRC2 <- c(NCRC2, log(temp2/temp3))
    NCRC3 <- c(NCRC3, log(temp3/temp4))
    NCRC4 <- c(NCRC4, log(temp4/temp5))
    NCRC5 <- c(NCRC5, log(temp5/temp6))
    start = start + 1
  }
  matrix = as.matrix(cbind(NCRC1, NCRC2, NCRC3, NCRC4, NCRC5))
  mu = apply(matrix, 1, mean)
  s = sqrt(apply(matrix, 1, var))
  se = s/sqrt(5)
  GR = mu - 1.533 * se
  return(GR)
}



Evaluate.Triggers <- function(row, T1, T2){
  trigger3 <- row[1]
  trigger4 <- row[2]
  trigger5 <- row[3]
  trigger6 <- row[4]
  trigger7 <- row[5]
  if (((trigger3 >= 2) & !is.na(trigger3)) &&
      ((trigger4 >= T1) & !is.na(trigger4)) &&
      ((trigger5 >= 250) &  !is.na(trigger5)) &&
      ((trigger6 > T2) & !is.na(trigger6)) &&
      ((trigger7 > 0) & !is.na(trigger7))) {
    return (TRUE)
  } else {
    return (FALSE)
  }
}

## Function to determine if all the triggers being activated
## at 85th day
Trigger.determine2 <- function(country.ts, confirm.ts,
                               death.ts, total.ts, T1, T2){
  rolling.c = rolling(confirm.ts)
  rolling.t = rolling(total.ts)
  CR = confirmation.ratio(rolling.confirm = rolling.c,
                          rolling.total = rolling.t)
  GR = growth.rate(total.ts)
  N = length(GR)
```

```
  N <- N + 84
  trigger3 <- country.ts[85:N]
  trigger4 <- death.ts[85:N]
  trigger5 <- rolling.t[85:N]
  trigger6 <- CR[85:N]
  trigger7 <- GR
  trigger.matrix <- as.matrix(cbind(trigger3, trigger4,
                                    trigger5, trigger6,
                                    trigger7))
  print(trigger.matrix)
  R <- apply(trigger.matrix, 1,
             function(x) Evaluate.Triggers(x, T1, T2))
  print(R)
  result <- sum(R)
  if (result > 1) {
    return (TRUE)
  }else {
    return (FALSE)
  }

}


## Function to plot the data
Plot.data <- function (data, Title, x.label, y.label,
                       y.lim.low = 0, y.lim.high = 2000000,
                       y.int = 0){
  rownames(data) = paste("time", seq(length(data[,1])), sep="")
  colnames(data) = paste("trial", seq(length(data[1,])), sep="")
  dat = as.data.frame(t(data))
  dat$trial = rownames(dat)
  mdat = melt(dat, id.vars="trial")
  mdat$time = as.numeric(gsub("time", "", mdat$variable))
  mdat$grand <- 1
  datplot <- ggplot(mdat, aes(x=time, y=value, group=trial)) +
    theme_bw() +
    theme(plot.title = element_text(hjust = 0.5)) +
    theme(panel.grid=element_blank()) +
    geom_line(size=0.3, alpha=0.4) +
    ylim (y.lim.low, y.lim.high) +
    stat_summary(aes(group = grand), fun="mean", colour="cyan2",
                 geom="line", size = 1.5) +
    geom_hline(yintercept=y.int, linetype="dashed",
               color = "red", size = 1.2)
```

```
  print(datplot + labs(title = Title, y=y.label, x = x.label))
}


## Output the raw triigers
set.seed(251121253)
Raw.Trigger <- function(para, mean = means,
                        lag = lg, N = 200,
                        init = inits, residual.cov = resi.cov,
                        xi1 = log.xi1, xi2 = log.xi2,
                        col) {
  result <- Data.Generate(para, mean = means, lag = lg, N ,
                          init = inits, residual.cov = resi.cov,
                          xi1 = log.xi1, xi2 = log.xi2)[,col]
  return(result)
}

##Plot the actual triggers
Trigger.result <- function(para, mean = means,
                           lag = lg, N = 150,
                           init = inits,
                           residual.cov = resi.cov,
                           xi1 = log.xi1, xi2 = log.xi2,
                           trigger, death.rate){
  output <- Data.Generate(para, mean = means, lag = lg, N ,
                          init = inits, residual.cov = resi.cov,
                          xi1 = log.xi1, xi2 = log.xi2)
  if (trigger == "4") {
    death.ts <- output[,3]
    result <- death.ts
  } else if (trigger == "5") {
    total.ts <- Total.recover(output, death.rate)
    total.ts <- total.ts[,4]
    rolling.t <- rolling(total.ts)
    result = rolling.t
  } else if (trigger == "6") {
    total.ts <- Total.recover(output, death.rate)[,4]
    confirm.ts <- output[,2]
    rolling.t <- rolling(total.ts)
    rolling.c <- rolling(confirm.ts)
    CR <- confirmation.ratio(rolling.c, rolling.t)
    result <- CR
  } else if (trigger == "7") {
    total.ts <- Total.recover(output, death.rate)[,4]
    rolling.t <- rolling(total.ts)
```

```
    GR <- growth.rate(rolling.t)
    result <- GR
  }
  return(result)
}


##Plot the triggers results in the simulation

## country number simulation results
set.seed(251121253)
country.nums <-replicate(10000, Raw.Trigger(para = para,
                                            N = 200,col = 1),
                         simplify = TRUE)
Plot.data(country.nums, Title = "Infected Country Numbers",
          x.label = "Days",y.label = "Countries Numbers",
          y.lim.low = 0, y.lim.high = 500, y.int = 2)


## Obtain the expected number of countries being infected
expected.country <- function(data, day.mark){
  return(mean(data[,data[day.mark,] >= 0]))
}

expected.country(country.nums, 100)
expected.country(country.nums, 120)
expected.country(country.nums, 150)
expected.country(country.nums, 180)
expected.country(country.nums, 200)


## Plot the confirmed case numbers
set.seed(251121253)
confirmed.nums <-replicate(10000, Raw.Trigger(para = para, col =2),
                           simplify = TRUE)
Plot.data(confirmed.nums, Title = "Simulated Confirmed Numbers",
          x.label = "Days",y.label = "Confirmed Cases",
          y.lim.low = 0, y.lim.high = 90000000)

## plot the death case numbers
set.seed(251121253)
death.nums <-replicate(10000, Raw.Trigger(para = para,
                                          N = 150, col =3),
                       simplify = TRUE)
Plot.data(death.nums, Title = "Simulated Death Numbers",
```

```
          x.label = "Days",y.label = "Death Cases",
          y.lim.low = 0, y.lim.high = 90000000)

##Calculate the expected death numbers
expected.payout <- function(data, day.mark){
  return(mean(data[day.mark,]))
}

expected.payout(death.nums, 100)
expected.payout(death.nums, 120)
expected.payout(death.nums, 150)
expected.payout(death.nums, 180)
expected.payout(death.nums, 200)


## Simulation results for trigger 4 - trigger 7
set.seed(251121253)
trigger4 <-replicate(10000, Trigger.result(para = para,
                            N = 150, trigger = "4",
                            death.rate = 0.03),
                     simplify = TRUE)
Plot.data(trigger4, Title = "Simulated Death Numbers",
          x.label = "Days",y.label = "Death Cases",
          y.int = 2500)

set.seed(251121253)
trigger5 <-replicate(10000, Trigger.result(para = para,
                            N = 150, trigger = "5",
                            death.rate = 0.03),
                     simplify = TRUE)

Plot.data(trigger5, Title = "Simulated Rolling Total Cases",
          x.label = "Days",
          y.label = "Rolling Total Cases",
          y.lim.low = 0, y.lim.high = 9000000,
          y.int = 250)

set.seed(251121253)
trigger6 <-replicate(10000,
                     Trigger.result(para = para,
                     N = 150, trigger = "6",
                     death.rate = 0.03),
                     simplify = TRUE)
Plot.data(trigger6, Title = "Simulated Confirmation Ratio",
          x.label = "Days",
```

```
          y.label = "Confirmation Ratio",
          y.lim.low = 0,
          y.lim.high = 1,
          y.int = 0.2)

set.seed(251121253)
trigger7 <-replicate(10000,
                     Trigger.result(para = para,
                     trigger = "7", death.rate = 0.03),
                     simplify = TRUE)
Plot.data(trigger7, Title = "Simulated Growth Rate",
          x.label = "Eligible Calculation Days
          (After 84 rolling days)",
          y.label = "Growth.rate",
          y.lim.low = -3, y.lim.high = 3)




## Function to determine if payout is
#  activated in one simulation path [1] or not [0]
Triggers2 <- function(para, mean, lag, N, init,
                      residual.cov, xi1,  xi2,
                      D.rate, T1 =250, T2 =0.2){
  b <- Data.Generate(para, mean, lag, N,
                     init, residual.cov, xi1, xi2)
  result <- Total.recover(b, D.rate)
  result <- linear.smooth(result)
  country <- abs(result[,1])
  confirm <- result[,2]
  death <- result[,3]
  total <- result[,4]
  final <- Trigger.determine2(country, confirm,
                              death, total, T1, T2)
  return(final)
}

choose.N <- function(freq.per.cen){
  Nt = rexp(1, rate = freq.per.cen / 36500)
  Nt <- as.integer(Nt)
  Nt <- pmax(91, Nt)
  Nt <- pmin(365, Nt)
  return (Nt)
}
```

```
##Calculte the payout probability
set.seed(251121253)
##Frequency lambda = 5 / 100 years
result.10000<-replicate(10000,
                        Triggers2(para, mean = means,
                        lag = lg, N = choose.N(freq.per.cen),
                        init = inits, residual.cov = resi.cov,
                        xi1 = log.xi1, xi2 = log.xi2,
                        0.03, 2500, 0.2),
                        simplify = TRUE)

sum(result.10000) / length(result.10000)
## 98.78%


set.seed(251121253)
##Frequency lambda = 10/ 100 years
result.10000<-replicate(10000,
                        Triggers2(para, mean = means,
                                  lag = lg, N = choose.N(10),
                                  init = inits, residual.cov = resi.cov,
                                  xi1 = log.xi1, xi2 = log.xi2,
                                  0.03, 250, 0.2),
                        simplify = TRUE)

sum(result.10000) / length(result.10000)
## 97.69%


set.seed(251121253)
##Frequency lambda = 25 / 100 years
result.10000<-replicate(10000,
                        Triggers2(para, mean = means,
                        lag = lg, N = choose.N(25),
                        init = inits, residual.cov = resi.cov,
                        xi1 = log.xi1, xi2 = log.xi2,
                        0.03, 250, 0.2),
                        simplify = TRUE)
sum(result.10000) / length(result.10000)
## 93.82%



set.seed(251121253)
```

```
##Frequency lambda = 50 / 100 years
result.10000<-replicate(10000,
                        Triggers2(para, mean = means,
                        lag = lg, N = choose.N(50),
                        init = inits, residual.cov = resi.cov,
                        xi1 = log.xi1, xi2 = log.xi2,
                        0.03, 250, 0.2),
                        simplify = TRUE)
sum(result.10000) / length(result.10000)
##87.7%
```

D.3 Interest-rate modelling and bond pricing

The following is the implementation code based on the algorithm described in Section 3.3. It is implemented using both Python and R codes.

The first part is the Python code to count the number of days q in each interest period and extract the 6-month LIBOR reset date assuming the pandemic bond was issued at 2017-07-07 and matures at 2022-07-15.

The second part is the R code to simulate the interest-rate paths and discount the cash flows with trigger-risk adjustments.

```
# -*- coding: utf-8 -*-
"""LIBOR calibration using Vaceik Model.ipynb

Automatically generated by Colaboratory.

# General Import

Package Import
"""

# General imports
import string
import numpy as np
import pandas as pd
import math as math
import random
from datetime import datetime
from datetime import timedelta
from datetime import date

## read the LIBOR data
```

```
!gdown https://drive.google.com/uc?id=1BxyWqIq2K_A8sXZvH9yEmCVLs66dQf4o
! unzip '/content/Uploaded LIBOR Rate.zip'

## convert the time stamp into readable date format
def parser(s):
    return datetime.strptime(s, '%d.%m.%Y')

## read the data with converted date stamp
LIBOR_data  = pd.read_csv('Uploaded LIBOR Rate.csv',
                          parse_dates=[0], index_col=0,
                          squeeze=True, date_parser=parser)
LIBOR_data.head()

One_month_LIBOR = pd.DataFrame(LIBOR_data['1M'])
## Just test what is the monthly mean for the 1-month LIBOR data
LIBOR_monthly_mean = One_month_LIBOR.resample("M").mean()
Six_month_LIBOR = pd.DataFrame(LIBOR_data['6M'])

## Extract the correlation
LIBOR_data['1M'].corr(LIBOR_data['6M'])


## Define a function rolling at each payment date, count the day number
def duration(payment_date, start_year, start_month, start_date,
             end_year, end_month, end_date):
  #Do monthly rolling only
  start = date(start_year, start_month, start_date)
  end = date(end_year, end_month, end_date)
  day_count_record = []
  payment = date(start_year, start_month, payment_date)
  while payment < end:
    day_count = daycount(start, payment)
    day_count_record.append(day_count)
    ## update start & and payment
    # current year
    if start_month != 12 :
      start = payment
      start_month += 1
      payment = date(start_year, start_month, payment_date)
    # next year
    elif start_month == 12:
      start = payment
      start_month = 1
      start_year += 1
      payment = date(start_year, start_month, payment_date)
```

```
  #Do situation when start == end
  if payment == end:
    day_count = daycount(start, payment)
    day_count_record.append(day_count)
  elif payment > end:
    day_count = daycount(start, end)
    day_count_record.append(day_count)

  return day_count_record

## define a function to count days number between two dates
def daycount(start, end):
  delta = end - start
  return delta.days

## Define a function generate time schedule
## for next interest reset date (6-month)
def Time_to_reset (start_year, start_month, start_date,
            end_year, end_month, end_date):
  start = date(start_year, start_month, start_date)
  end = date(end_year, end_month, end_date)
  day_count_record = []
  day_count = 0
  while start <= end:
    ## reset date is before January
    if start < date(start_year, 1, 15):
      reset = date(start_year, 1, 15)
      duration = daycount(start, reset)
      day_count = day_count + duration
      day_count_record.append(day_count)
      start = reset
    ## reset date is before July
    elif start < date(start_year, 7, 15):
      reset = date(start_year, 7, 15)
      duration = daycount(start, reset)
      day_count = day_count + duration
      day_count_record.append(day_count)
      start_year = start_year + 1
      start = reset

  return day_count_record

duration_list = duration(15, 2017, 7, 7, 2022, 7, 15)
```

```
reset_list = Time_to_reset(2017, 7, 7, 2022, 7, 15)

df = pd.DataFrame(duration_list, columns=["interest.period"])
duration.csv = df.to_csv('duration_list.csv', index=False)

df2 = pd.DataFrame(reset_list, columns = ["interest.reset.duration"])
reset = df2.to_csv('reset_list.csv', index=False)



from google.colab import files
#files.download('duration_list.csv')

#files.download("reset_list.csv")

### Interest Rate Modelling
library(dplyr)
## Pandemic Modelling
## Assume the pandemic follows a poisson ditribution:
freq.per.cen = 5
pandemic.p <- ppois(0,
                    lambda = freq.per.cen
                    / (360 * 100),
                    lower.tail = FALSE)
pandemic.p2 <- ppois(0,
                     lambda = freq.per.cen *
                       2 / (360 * 100),
                     lower.tail = FALSE)
pandemic.p3 <- ppois(0, lambda = freq.per.cen *
                       5/ (360 * 100),
                     lower.tail = FALSE)
pandemic.p4 <- ppois(0, lambda = freq.per.cen *
                       10 / (360 * 100),
                     lower.tail = FALSE)

## Interest Movement Modelling
LIBOR <- read.csv(file = "~/Downloads/summer project data
                  /Uploaded LIBOR Rate.csv")
six.month.libor <- LIBOR[,8]
one.month.libor <- LIBOR[,5]


## Extract a in Vasicek model
extract_a <- function(interest.ts, dt){
  lag1 <- na.omit(lag(interest.ts))
```

```
  lead1 <- na.omit(lead(interest.ts))
  temp1 <- sum(lag1)
  temp2 <- sum(lead1)
  temp3 = temp1 - temp2
  left <- sum(lag1 * temp3)

  N = length(interest.ts)
  temp1 <- sum(lag1 * lead1)
  temp2<- sum(lag1 * lag1)
  right <- (N-1) * (temp1 - temp2)

  bottom <- dt * (((N-1) * temp2) - (sum(lag1))^2)

  result = - (left - right)/bottom
  return(result)
}

## Extract b in Vasicek model
extract_b <- function(interest.ts){
  lag1 <- na.omit(lag(interest.ts))
  lead1 <- na.omit(lead(interest.ts))
  temp1 <- sum(lag1)
  temp2 <- sum(lag1 * lead1)
  left = -(temp1 * temp2)

  temp1 <- sum(lead1)
  temp2 <- sum(lag1 * lag1)
  right <- temp1 * temp2

  top <- left + right

  N = length(interest.ts)
  left <- sum(lag1 * (sum(lag1) -
                        sum(lead1)))
  right <-  (N - 1) * (sum(lag1 * lead1)
                       - sum(lag1 * lag1))
  bottom <- left + right

  result = - (top/bottom)
  return(result)
}


## Extract sigma in Vasicek model
extract_sigma <- function(interest.ts, dt){
```

```
  lag1 <- na.omit(lag(interest.ts))
  lead1 <- na.omit(lead(interest.ts))
  N = length(interest.ts)
  a = extract_a(interest.ts, dt)
  b = extract_b(interest.ts)
  temp = sum((lead1 - lag1 - a * (b - lag1) * dt)^2)
  result = sqrt(temp/ ((N - 1) * dt))
  return(result)
}



likelihood_fun <- function(interest.ts, dt){
  N = length(interest.ts)
  a = extract_a(interest.ts, dt)
  b = extract_b(interest.ts)
  sigma = extract_sigma(interest.ts, dt)
  lead1 <- na.omit(lead(interest.ts))
  lag1 <- na.omit(lag(interest.ts))
  temp = sum((lead1 - lag1 - a * (b - lag1) * dt)^2)
  likelihood = - ((N - 1)/2 * log(2 * pi)) -
    (N - 1)*log(sigma) - 0.5*(N-1) * log(dt) -
    - 1 / (2 * sigma^2 * dt) * temp
  return(likelihood)
}

## Estimate the standard error for a, b, sigma.
SEs <- function(a, b, sigma, dt, interest.ts){
  N <- length(interest.ts)
  temp1 <- 2 * a * (dt)^2 * exp(-2 * a * dt) *
    sum(interest.ts - b)^2
  temp2 <- sigma^2 * (1- exp(-2*a*dt))
  temp3 <- N * exp(-4 * a * dt) *
    (exp(2*a*dt) - 2*a*dt -1)^2
  temp4 <- 2 * a * (1 - exp(-2*a*dt))^2
  temp5 <- (temp1/temp2) + (temp3/temp4)
  SEa <- sqrt(1/temp5)

  temp6 <- (2*N*a*(1-exp(-a*dt)))
  / (sigma^2*(1 + exp(-a *dt)))
  SEb <- sqrt(1/temp6)

  temp7 <- 2*N/(sigma^2)
  SEc <- sqrt(1/temp7)
  return(list(SEa, SEb, SEc))
```

```
}

##Vescek Model for Euler discretization
interest_path <- function(a, b, sigma, N, rt){
  dt = 5 / N
  std = sqrt(dt)
  normal = rnorm(N, mean = 0, sd = std)
  interest = vector()
  interest[1] <- rt
  time_point = 1
  while (time_point <= N){
    next_int = interest[time_point] +
      a*(b - interest[time_point])* dt
    + sigma * normal[time_point]
    interest[time_point + 1] <- next_int
    time_point = time_point + 1
  }
  return(interest)
}


##Import the duration of the float coupon payment &
##days to next reset date to update the new LIBOR Rate
##Assuming the issue date and end date is 2017-7-7 and 2022-7-15

##Following are the files incoporated with Python code
interest.period <- read.csv(file = "~/Downloads/duration_list.csv")
reset.period <- read.csv(file = "~/Downloads/reset_list.csv")

interest.period <- as.vector(interest.period[,1])
reset.period <- as.vector(reset.period[,1])

N = tail(reset.period, n = 1)

##Build the Monte Carlo simulation for calculating the
#float coupon bond price
##Assume the face value is 1000
##Extract the parameters form six_month_LIBOR
six.a = extract_a(six.month.libor, 1/360)
six.b = extract_b(six.month.libor)
six.sig = extract_sigma(six.month.libor, 1/360)
SEs(six.a, six.b, six.sig, 1/360, six.month.libor)
```

```
## Define a function to find the coupon rate on the reset date:
reset_rate_list <- function(reset.period, interets.lst){
  applied_rate = vector()
  applied_rate <- interets.lst[1]
  for (day in reset.period){
    applied_rate <- c(applied_rate, interets.lst[day])
  }
  return(applied_rate)
}

## Simulate the six-month LIBOR
six.month = interest_path(a = six.a, b = six.b,
                          sigma = six.sig,
                          N = 1834, rt = 1.46544)

applied_rate = reset_rate_list(reset.period, six.month)

##Calculate the float interest payment with the simulated
##interest path
float.pay <- function(applied_rate, reset.pd,
                      interest.pd, spread, principle){
  indicator = 1
  past.days = 0
  payment = vector()
  for (days in interest.pd){
    if (past.days < reset.pd[indicator]) {
      pay <- (days / 360)  * ((applied_rate[indicator]
                               + spread)/100)
      * principle
      ##check
      print(days)
      print(applied_rate[indicator])
      payment <- c(payment, pay)
      past.days <- past.days + days

    } else if (past.days >=  reset.pd[indicator]){
      indicator <- indicator + 1
      pay <- (days / 360)  * ((applied_rate[indicator]
                               + spread)/100)
      * principle
      ##check
      print(days)
      print(applied_rate[indicator])
      payment <- c(payment, pay)
      past.days <- past.days + days
```

```
      print(pay)
    }
  }
  last_index = length(payment)
  payment[last_index] <- payment[last_index] + principle
  return(payment)
}

## paramteres for the one-month US LIBOR
one.a = extract_a(one.month.libor, 1/360)
one.b = extract_b(one.month.libor)
one.sig = extract_sigma(one.month.libor, 1/360)
SEs(one.a, one.b, one.sig, 1/360, one.month.libor)

## Simulate the one-month LIBOR
one.month = interest_path(one.a, one.b, one.sig,
                          N = 1834, rt = 1.46544)


## Plot the paths of simulated interest rate
one.month.rate <-replicate(10000,
                           adjusted.one.month.discount.rate
                           (1, pandemic.p, 1,
                             one.month.libor, 1834,
                             rt = 1.22633,
                             interest.period, 6.0, 1000),
                           simplify = TRUE)
Plot.data(one.month.rate, Title = "One Month LIBOR Rate",
          x.label = "Days",
          y.label = "One-Month-LIBOR Rate", y.lim.low = -1,
          y.lim.high = 3, y.int = -3)


six.month.rate <-replicate(10000,
                           interest_path(six.a, six.b, six.sig,
                           N = 1834, rt = 1.46544),
                           simplify = TRUE)
Plot.data(six.month.rate, Title = "Six Month LIBOR Rate",
          x.label = "Days",
          y.label = "LIBOR Rate",
          y.lim.low = -1, y.lim.high = 3,
          y.int = -3)
```

```
## calculate the discount rate
discount_rates <- function(interest.lst, interest.pd, dt){
  due_time = cumsum(interest.pd)
  discount = vector()
  for (due.day in due_time){
    temp1 = sum(interest.lst[1:due.day] * dt) / 100
    dis_rate = exp(-temp1)
    discount <- c(discount, dis_rate)
  }
  return(discount)
}



hazard.rate <- function(trigger.risk, pandemic.risk,
                        expected.loss){
  hazard  = trigger.risk * pandemic.risk
  final = hazard * expected.loss
  return(final)
}

##Function to calculate the bond price
Price <- function(trigger.risk, pandemic.risk,
                  expected.loss,
                  one.month.LIBOR, N, one.rt,
                  six.rt, six.month.LIBOR,
                  reset.pd, interest.pd, spread,
                  principle =1000) {
  one.a = extract_a(one.month.LIBOR, 1/360)
  one.b = extract_b(one.month.LIBOR)
  one.sig = extract_sigma(one.month.LIBOR, 1/360)
  six.a = extract_a(six.month.LIBOR, 1/360)
  six.b = extract_b(six.month.LIBOR)
  six.sig = extract_sigma(six.month.LIBOR, 1/360)
  six.month = interest_path(a = six.a, b = six.b,
                            sigma = six.sig,
                            N = N, rt = six.rt)
  one.month = interest_path(a = one.a, b = one.b,
                            sigma = one.sig,
                            N = N, rt = one.rt)
  applied_rate = reset_rate_list(reset.pd, six.month)
  float_payment = float.pay(applied_rate, reset.pd,
                            interest.pd, spread, principle)
  risk.adjust = hazard.rate(trigger.risk,
                            pandemic.risk, expected.loss)
```

```
  one.month = one.month + risk.adjust
  discount.r <- discount_rates(one.month, interest.pd, 1/360)
  payment.schedule = discount.r * float_payment
  value = sum(payment.schedule)
  return(value)
}

## Function to apply trigger risk adjsutment
adjusted.one.month.discount.
rate <- function(trigger.risk, pandemic.risk,
                 expected.loss, one.month.LIBOR,
                 N, rt, six.month.LIBOR, reset.pd,
                interest.pd, spread, principle =1000,
                Sim.N) {
  one.a = extract_a(one.month.LIBOR, 1/360)
  one.b = extract_b(one.month.LIBOR)
  one.sig = extract_sigma(one.month.LIBOR, 1/360)
  one.month = interest_path(a = one.a,
                            b = one.b, sigma = one.sig,
                            N = N, rt = rt)
  risk.adjust = hazard.rate(trigger.risk,
                            pandemic.risk, expected.loss)
  one.month = one.month + risk.adjust
  return(one.month)
}




## Price of the class A at lambda = 5 / 36500
set.seed(251121253)
Averaged.Price.A1 <- replicate(10000,
                     Price(.9878, pandemic.p, 0.167,
                           one.month.libor, 1834,
                           one.rt = 1.2263, six.rt = 1.4654,
                           six.month.libor, reset.period,
                           interest.period, 6.5, 1000),
                              simplify = T)
averaged.price <- mean(Averaged.Price.A1)
sqrt(var(Averaged.Price)/10000)
averaged.price + 1.96 * sqrt(var(Averaged.Price)/10000)
averaged.price - 1.96 * sqrt(var(Averaged.Price)/10000)


## Price of the class B at lambda = 5 / 36500
```

```
set.seed(251121253)
Averaged.Price.B1 <- replicate(10000,
                     Price(.9878, pandemic.p, 1,
                            one.month.libor, 1834,
                            one.rt = 1.2263,
                            six.rt = 1.4654,
                            six.month.libor,
                            reset.period,
                            interest.period, 11.1, 1000),
                               simplify = T)
averaged.price.B1 <- mean(Averaged.Price.B2)
sqrt(var(Averaged.Price.B1)/10000)
averaged.price.B1  + 1.96 * sqrt(var(Averaged.Price.B)/10000)
averaged.price.B1  - 1.96 * sqrt(var(Averaged.Price.B)/10000)


## Price of the class A pandemic with lambda = 10 / 36500
set.seed(251121253)
Averaged.Price.A2 <- replicate(10000,
                     Price(.9769, pandemic.p2,
                     0.167, one.month.libor,
                     1834, one.rt = 1.2263,
                     six.rt = 1.4654, six.month.libor,
                     reset.period,
                     interest.period, 6.5, 1000),
                               simplify = T)
averaged.price.A2 <- mean(Averaged.Price.A2)
sqrt(var(Averaged.Price.A2)/10000)
averaged.price.A2 + 1.96 * sqrt(var(Averaged.Price.A2)/10000)
averaged.price.A2 - 1.96 * sqrt(var(Averaged.Price.A2)/10000)


## Price of the class B with lambda = 10 /36500
set.seed(251121253)
Averaged.Price.B2 <- replicate(10000,
                     Price(.9769, pandemic.p2, 1,
                     one.month.libor, 1834,
                     one.rt = 1.2263,
                     six.rt = 1.4654, six.month.libor,
                     reset.period, interest.period,
                     11.1, 1000), simplify = T)
averaged.price.B2 <- mean(Averaged.Price.B2)
sqrt(var(Averaged.Price.B2)/10000)
averaged.price.B2  + 1.96 * sqrt(var(Averaged.Price.B2)/10000)
averaged.price.B2  - 1.96 * sqrt(var(Averaged.Price.B2)/10000)
```

```
## Price of the class A
## pandemic with lambda = 25 /36500
set.seed(251121253)
Averaged.Price.A3 <- replicate(10000,
                     Price(.9382, pandemic.p3, 0.167,
                           one.month.libor, 1834,
                           one.rt = 1.2263, six.rt = 1.4654,
                           six.month.libor, reset.period,
                           interest.period, 6.5, 1000),
                               simplify = T)
averaged.price.A3 <- mean(Averaged.Price.A3)
sqrt(var(Averaged.Price.A3)/10000)
averaged.price.A3 + 1.96 * sqrt(var(Averaged.Price.A3)/10000)
averaged.price.A3 - 1.96 * sqrt(var(Averaged.Price.A3)/10000)


## Price of the class B
## pandemic with lambda = 25 /36500
set.seed(251121253)
Averaged.Price.B3 <- replicate(10000,
                     Price(.9382, pandemic.p3,
                           1, one.month.libor,
                           1834, one.rt = 1.2263,
                           six.rt = 1.4654, six.month.libor,
                           eset.period, interest.period,
                           11.1, 1000), simplify = T)
averaged.price.B3 <- mean(Averaged.Price.B3)
sqrt(var(Averaged.Price.B3)/10000)
averaged.price.B3 + 1.96 * sqrt(var(Averaged.Price.B3)/10000)
averaged.price.B3 - 1.96 * sqrt(var(Averaged.Price.B3)/10000)


## Price of the class A
## pandemic with lambda = 50 /36500
set.seed(251121253)
Averaged.Price.A4 <- replicate(10000,
                     Price(.877, pandemic.p4,
                           0.167, one.month.libor,
                           1834, one.rt = 1.2263,
                           six.rt = 1.4654, six.month.libor,
                           reset.period, interest.period,
                           6.5, 1000), simplify = T)
```

```
averaged.price.A4 <- mean(Averaged.Price.A5)
sqrt(var(Averaged.Price.A5)/10000)
averaged.price.A4 + 1.96 * sqrt(var(Averaged.Price.A5)/10000)
averaged.price.A4 - 1.96 * sqrt(var(Averaged.Price.A5)/10000)



set.seed(251121253)
Averaged.Price.B4 <- replicate(10000,
                     Price(.877, pandemic.p4,
                             1, one.month.libor,
                            1834, one.rt = 1.2263,
                            six.rt = 1.4654, six.month.libor,
                            reset.period,
                           interest.period, 11.1, 1000),
                               simplify = T)
averaged.price.B4 <- mean(Averaged.Price.B4)
sqrt(var(Averaged.Price.B4)/10000)
averaged.price.B4 + 1.96 * sqrt(var(Averaged.Price.B4)/10000)
averaged.price.B4 - 1.96 * sqrt(var(Averaged.Price.B4)/10000)
```