A note on Stochastic Modeling of Corona Virus

In this note we develop a stochastic differential equation model for CORONA VIRUS and its impacts on economy and well-being in terms of a tri-state (X_t, Y_t, Z_t) model as follows.

$$dX_t = X_t(c_t - X_t)^\alpha \, dt + \sigma_1 X_t \, dB_t^{(1)}$$

$$dY_t = \mu(X_t, Z_t) Y_t \, dt + \sigma(X_t, Z_t) \, Y_t dB_t^{(2)}$$

$$dZ_t = \kappa(X_t, Y_t, Z_t)(\theta(X_t, Y_t) - Z_t) + \sigma_2 \sqrt{Z_t} \, dB_t^{(3)}$$

Here, X_t represents daily new cases, Y_t is a financial index and Z_t is a virtual index for well-being and confidence level of life and community. We employ a logistic model of spread of epidemics where α is for social distancing measure and c_t is an estimated upper bound. Y_t obeys a standard finance model with return rate μ and volatility σ of markets. We assume Z_t follows a Cox-Ingersoll-Ross (CIR) model in which κ is a mean return rate to a dynamic mean θ. Based on the proposed model we formulate on-line estimation and identification of model and optimal reopening of makes with safety margin. We report our preliminary simulation and testing and our findings.

For the first model of X_t one can use a different model than $f(X_t) = X_t(c - X_t)^\alpha$, e.g., spline model is given by

$$f(x) = x \sum_{k=1}^{m} f_k \, B_k(x)$$

Then, we identify model function f by determining coefficient $\{f_k\}$.

A functional map (μ, σ) for process Y_t and (κ, θ) for process Z_t are needed to be introduced. One can perform an on-line estimate and online model identification as follows:

online estimate of (c, α) Compute return rate.

$$R_t = \frac{X_{t+\Delta t} - X_t}{X_t}$$

We formulate the least squares problem

$$\sum_{t=1}^{T} |R_t - (c - X_t)^\alpha|^2$$

over (c, α). One can solve it by the Gauss-Newton method.

online model identification of (μ, κ) Compute return rate.

$$\tilde{R}_t = \frac{Y_{t+\Delta t} - Y_t}{Y_t}$$

We formulate the least squares problem

$$\sum_{t=1}^{T} |\tilde{R}_t - \mu(X_t, Z_t)|^2 \text{ over maps } \mu, \theta, \kappa$$

subject to a model dynamics for Z_t.

For example we propose to use a model for (κ, μ) as

$$\kappa\,((c_t - X_t)^{-\alpha} - z)z,$$

and

$$\mu = \mu(z-1), \quad \sigma(x,z) = \sigma\sqrt{z}$$

In Figure 1 we show simulation results of our model and one can observe recovery of (Y_t, Z_t) as of now (April 22, 2020).

It is important to determine when one can reopen markets. We formulate it as an optimal stopping time $\tau \geq 0$ (i.e., optimal timing of reopening):

Optimal stopping Based on the model we consider the optimal stopping problem:

$$\max_{\tau\geq 0} E[\psi(X_\tau, Y_\tau, Z_\tau) + \int_0^\tau f(X_t, Y_t, Z_t)\,dt]$$

subject to (recovery model for X_t)

$$dX_t = -X_t(c_t - X_t)^\alpha\,dt + \sigma_1 X_t\,dB_t^{(1)}.$$

For example, we let $\psi(x) = c - x$ (index for cases of X_t when stop) and $f = f_0$-(a constant for safety margin of stopping and larger f_0 means safer). Value function defined by

$$v(x) = E^{0,x}[E[\psi(X_\tau) - \int_9^\tau f_0\,dt]$$

satisfies a dynamical (Bellman-Dynkin) principle

$$\max\{\mathcal{L}v + f, \psi - v\} = 0$$

where generator $\mathcal{L}$ of X_t is defined as

$$\mathcal{L}v = -x(c-z)^\alpha v_x + \frac{\sigma x^2}{2} v_{xx}.$$

The optimal stopping policy is determined by a feedback form

$$\tau = \inf_{t\geq 0}\{X_t < v(X_t)\}.$$

where v is a value function (see, Figure 2).

The following is a matlab implementation of our model.

```
x=1; xx=x; si=.1; c=10; al=.7; while x<c;
x=exp(.01*(c-min(c,x))^al+.1*si*randn-si^2*.5*.01)*x; xx=[xx x]; end
for k=1:200; x=exp(-.01*(c-min(c,x))^al+.1*si*randn-si^2*.5*.01)*x; xx=[xx x]; end
```

```
z=1; th=1;ka=2; for k=1:100;
z=exp(.01*ka*((c-xx(k))^(-al)-z)+.1*si*sqrt(abs(z))*z*randn-abs(z)*si^2*.5*.01)*z;zz(k)=


y=1; mu=.03;  for k=1:100; y=exp(.01*mu*(zz(k)-1)
+.1*sqrt(abs(zz(k)))*randn-abs(zz(k))*.5*.01)*z;yy(k)=y;end

%%%%% OPTIMAL STOPPING
n=50; dx=10/n; e=ones(n,1); h=spdiags([-e 2*e -e],-1:1,n,n); h(n,n)=1; h=h/dx/dx; h0=h;
n=50; e=ones(n,1); dc=spdiags([-e e],[-1 1],n,n); dc(n,n)=1;dc=dc/2/dx;
h=-spdiags(x.*(10-x).^.5,0,n,n)*dc-.5*si^2*spdiags(x.^2,0,n,n)*h0;
x=[dx:dx:10]'; psi=10-x;  v=(-h+.01*speye(n))\psi;
k=find(v-psi(:)<0); dd=0*v; dd(k)=1e8; dd=spdiags(dd(:),0,n,n);
vv=v; v=(-h+dd)\(-ones(n,1)+dd*psi(:)); norm(vv-v)
```

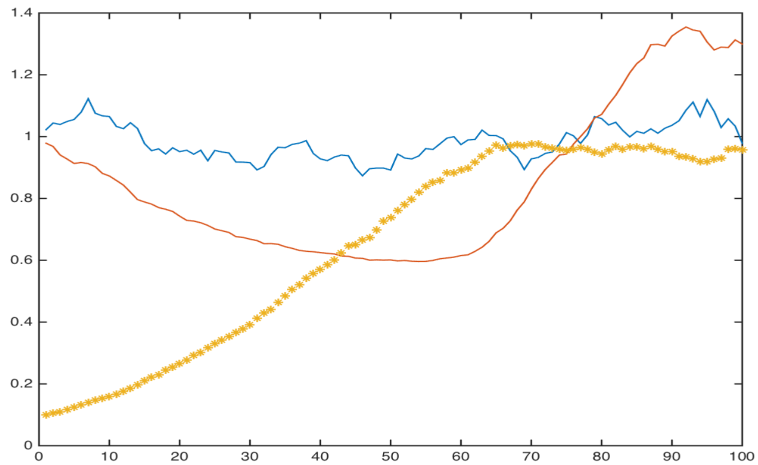

Figure 1: Y_t:Blue, Z_t: Red and X_t: star Lines

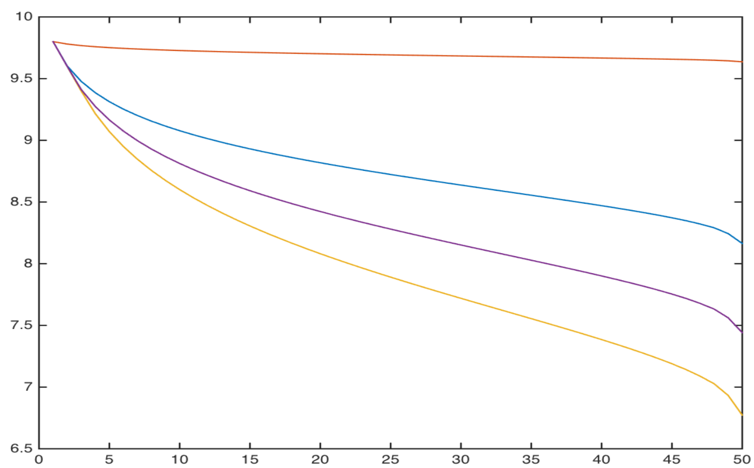

Figure 2: Value function for $f_0 = .1,\ 1,\ 1.5$ and 2