Convex Optimization Term Paper

Vighnesh N - EE17B119

May 2020

1

Name: Vighnesh Natarajan

Roll No: EE17B119

Title: Optimizing the amount of medicines produced and distributed to states during a pandemic like covid, formulated and analysed using online convex optimization methods.

Reference: Introduction to Online Convex Optimization by Elad Hazan
- https://arxiv.org/pdf/1909.05207.pdf

Contribution: I have formulated the problem myself, in a couple of different styles, with analysis and simulations performed based on methods in the book cited above, and have used the actual data from the covid19india website.

2 Introduction and motivating the problem

In the present scenario of our covid world, I was motivated to try and choose a problem that is present around us, and atleast try and do some analysis of it based on techniques learnt in course and in the reference.
The problem:
With the cases in our country, the problem aims at accuratly supplying the required number of medicines to minimize the loss of life. The assumption is that only a certain fraction of the cases are critical, and the medicines supply needs to be able to cater to that demand, without shortages, or over production. The Data being used is from the official covid19india.org website.
The Cases can be fit to any model and the formulation can be appropriately adjusted. Here, I have fit three states states to exponentials, as the confirmed cases for most states with large number of cases follows this to a reasonable extent.

I have focused mainly on Maharashtra, Karnataka and Andhra Pradesh for the simulations. Have added a simulation at the end, considering more states.

3 Data and Curve fitting

Plots for data from the covid19india.org

NOTE: In the actual implementation of the problem, on day i, only data until day has been used, to decide on day i+1. Over here, I am just showing a fit curve for three states, for report purpose. The parameters from this curve involve all data and are only used for regret analysis.

```
%termpaper data, example of how it was preprocessed for MH and KA
tbl = readtable('data.csv');
mh = [];   % Cumulative
mh_d = []; % Daily
x1 = linspace(1,size(mh,1),size(mh,1));

for i=1:size(tbl.MH)
    if(mod(i,3)==1)
        if(i==1)
            mh = [mh; tbl.MH(i)];
        else
            mh = [mh; mh(int16(i/3))+tbl.MH(i)];
        end
        mh_d = [mh_d; tbl.MH(i)];
    end
end
f_mh = fit(x1.',mh,'exp1');

ka = [];   %Cumulative
ka_d = []; %Daily
for i=1:size(tbl.KA)
    if(mod(i,3)==1)
        if(i==1)
            ka = [ka; tbl.KA(i)];
        else
            ka = [ka; ka(int16(i/3))+tbl.KA(i)];
        end
        ka_d = [ka_d; tbl.KA(i)];
    end
end
f_ka = fit(x1.',ka,'exp1');
```

Link to code attached below

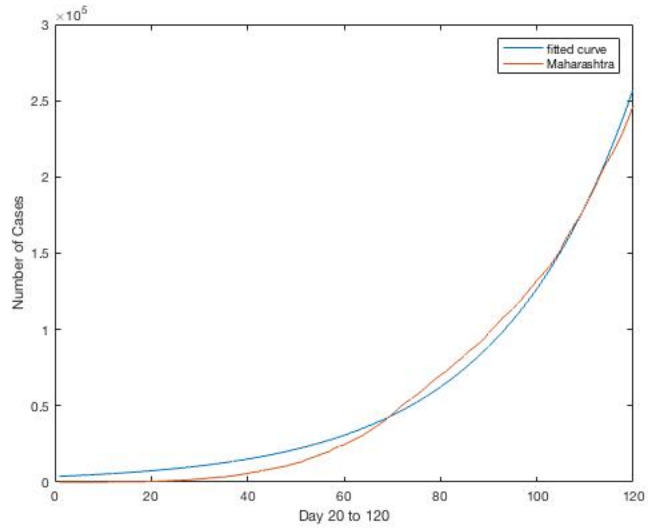

Figure 1: Plots for Maharashtra

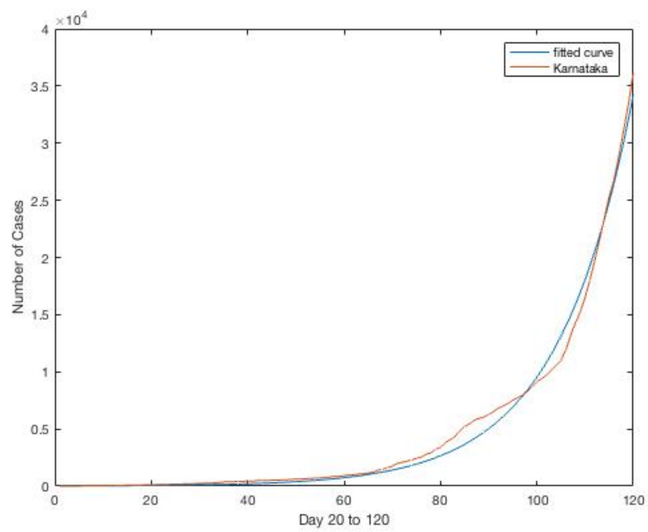

Figure 2: Plots for Karnataka

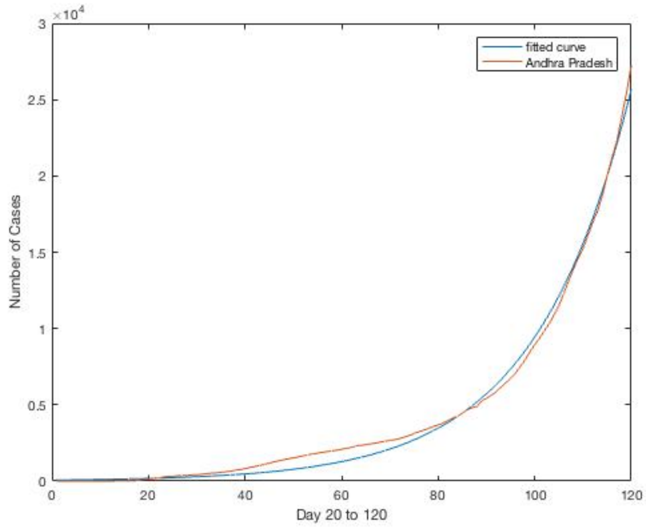

Figure 3: Plots for Andhra Pradesh

The different states have been fit to exponentials. Specifically:

$$n(t) = be^{at}$$

Code to extract data

Estimation of next alpha from current alpha for more accurate prediction (Important for weekly production).

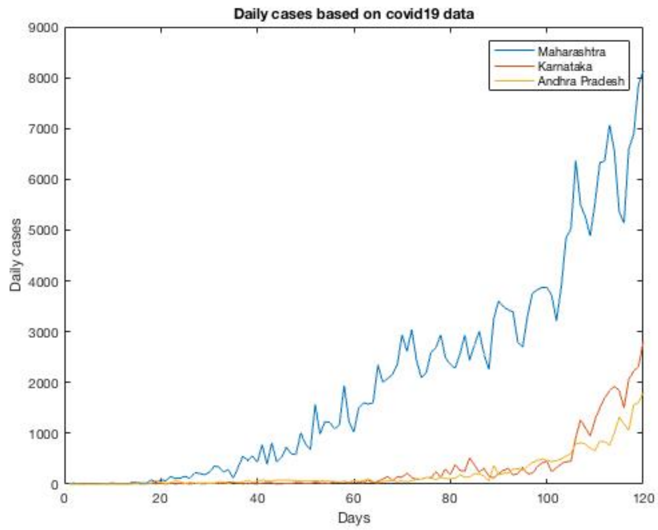

Figure 4: Daily cases in all three states

4 Problem formulation for daily production

Our goal is to minimize the costs and supply sufficient to the critical patients.

Defining the parameters, variables and constants used:
m_i = Medicines supplied to state i
M = Total number of medicines
$delM$ = Total increment in production required
γ = Minimum Critical population fraction (Can be updated every day by taking ratio of deaths and recoveries)
α_i and β_i = a and b of the exponentials
k_i = Number of critical population in state i
p, b = Objective Cost and the cost factor
μ = Weight factor for serious cases
N = Number of states

In all expressions, the exponent will have (t+1) as we need to produce for the next day.

Formulation:

$$Minimizep - \mu\Sigma_{i=1}^{N} log(k_i)$$
Subject To: $\Sigma_{i=1}^{N} m_i < M + delM$
$$k_i > \gamma * \beta_i \text{ i = 1...N}$$
$$m_i > k_i e^{\alpha_i(t+1)} \text{ i = 1...N}$$

$$p > b * delM$$
$$delM > 0$$

Explanation:
The motivation for the objective: We want to minimize cost as well as try and cater to as many people as possible. Minimize cost is indicated by p, and trying to cater to as many people by -log(k_i). The log function is used because the cases grow as an exponential, and just adding the log as an appropriate equalizer.
We want to count the critical population of a state to have a minimum value, based on the ratio γ
The cost depends on the increases in requirement, with a factor b.
The first constraint $\Sigma_{i=1}^{N} m_i < M + delM$ tries to minimize the increase in delM, along with the price equation.

5 Dual analysis

Finding the dual, to do a thorough analysis and attempt online dual ascent, which has easier projections (Much easier).
Calculating Lagrangian

$$L(m, k, delM, p, \lambda) = (p - \Sigma_{i=1}^{N} log(k_i)) + \lambda_1(\Sigma_{i=1}^{N} m_i - M - delM) + \Sigma_{i=1}^{N}(\lambda_{2i}(\gamma\beta_i - k_i)) + \Sigma_{i=1}^{N}(\lambda_{3i}(k_i e^{\alpha_i(t+1)} - m_i)) + \lambda_4(0 - delM) + \lambda_5(b * delM - p)$$

$$\phi(\lambda) = \underset{m,k,delM,p}{argmin} \; L(m, k, delM, p, \lambda)$$

So,

$$1)\nabla_m L = 0$$
$$2)\nabla_k L = 0$$
$$3)\frac{\partial L}{\partial delM} = 0$$
$$4)\frac{\partial L}{\partial p} = 0$$

From this, we get
$1 - \lambda_5 = 0$ (From (4))
$\lambda_1 - \lambda_{3i} = 0$
So, $\lambda_1 = \lambda_{3i} = \lambda_3 = \lambda$ (From (1))
$b\lambda_5 - \lambda_4 - \lambda_1 = 0$ (From (3))
$-\frac{\mu}{k_i} - \lambda_{2i} + \lambda_{3i} e^{\alpha_i(t+1)} = 0$ (from (2))

To simplify,

$$\lambda_1 = \lambda_{3i} = \lambda$$
$$\lambda_4 = b - \lambda$$
$$\lambda_5 = 1$$
$$\lambda_{2i} = -\frac{\mu}{k_i} + \lambda e^{\alpha_i(t+1)} \text{ (Can substitute for } k_i \text{ based on this)}$$

Substituting and cancelling out terms, we finally get

$$\phi(\lambda) = \mu(N - \Sigma_{i=1}^{N} log(k_i)) - \lambda M + \lambda \Sigma_{i=1}^{N}(\beta_i \gamma e^{\alpha_i(t+1)}) - \Sigma_{i=1}^{N}(\gamma \beta_i \mu / k_i)$$

So, the dual optimization problem is

$$\text{maximize: } \phi(lambda)$$
$$\text{subject to: } \lambda \geq 0$$
$$\lambda \leq b$$
$$\lambda \geq \frac{\mu}{k_i} e^{-\alpha_i(t+1)}, \text{ i = 1...N}$$

6 Short Readings and understanding on Online convex optimization

Before going deeper, I shall discuss on what I learnt about this topic online.
In online convex optimization, an agent iteratively takes decisions on the problem, without any knowledge of what the outcome of each decision will be, and aptly matches the description of the problem I am challenging. Every time a decision, the agent suffers from a "loss" or "regret", which in this case is a the amount of medicines produced and disproportionately distributed. These regrets are unknown to the agent before hand.
For every action taken by the agent x_t, there is an effect/outcome $f_t(x_t)$, which influences the next decision to be taken.
The text book offers proof that the maximum regret faced can be bounded in the explanations of the algorithms, which indicates that with time, the agent will arrive at the optimal solution.

7 Solving using Online convex optimization methods

Motivation for solving as online problem: The parameters of the problem keep changing with time. Hence, I have attempted to demonstrate the algorithms, online gradient descent and online dual ascent.
Explanation of online algorithms:
In each iteration, the algorithm takes a step towards achieveing the objective. In the case where the step is outside the range of the function, a projection is

done to bring it back to the feasible set.

The most important fact to note is that the fact that the future cost functions observed may be very different to the one observed until the present, due to the fact that the parameters constantly change, in an online problem. In addition, the feasible set also changes with changes in the constraints and as time passes, and this leads to changes in the optimal point.
Per my analysis, this is where attempting the same problem using a dual ascent proved to be much simpler. As shown above, i was able to exactly calculate the dual, and the feasibility set has minimal changes, with much simpler projections, and was much simpler to get working. Simple useage of KKT conditions at the end give us the corresponding primal optimal value

The final result of the algorithm was not much different from stochastic gradient descent.
Checking Slater's condition:
Objective function is clearly convex. Because, -log(x) is convex, and p is affine. Sum of convex and affine is convex, and we are trying to minimize a convex function.

$$\nabla^2(-log(x)) = \frac{1}{x^2} \geq 0, \, \forall x$$

Hence, Satisfies second order test for convexity and -log(x) is convex.

All the constraints are clearly affine. For affine constraints, the slater's condition can be relaxed to allow equality also, which is usually strict inequality, how ever, for the chosen problem, strict inequality points also are present in the feasible set.
Hence clearly, slater's condition is satisfied, for all values(practical/realistic values) of the parameters of the formulation, and $p^* = d^*$!!. Hence, solving the problem using dual ascent is justified.

The primal optimal point is then obtained, and the medicines to be distributed are obtained by complimentary slackness property.

The online gradient descent/dual ascent algorithm
(The text book only described the descent. However, i myself tried out the dual ascent algorithm, based on what was learnt in class.)
1:Input: convex set K, T, $x_1 \epsilon$ K, η
2:for t = 1 to T do
3: Play x_t and observe cost $f_t(x_t)$.
4: Update and project:$y_{t+1} = x_t - \eta \nabla f_t(x_t)$
5: $x_{t+1} = \Pi_K(y_{t+1})$
6:end for

Couple of challenges in the problem in hand is that the convex set K keeps changing. The input point for a cycle may not be in the feasible set K. I also tried a method where in each day, update step is still done, and projected onto the new feasible set K.
To implement the projection in the primal, I tried a couple of methods. A simple if-else based system for each of the constraints, which was tedious. In addition, the most simple to do the projection was to use fmincon, and cvx which i used for testing. Note: Since the optimal point will keep changing, in the dual problem, on each day, I iterated to that feasible point, and let that be the final step of the day. Even otherwise, the lambdas can be used to get the primal solution through a simple argmin, tho that wont be optimal for the day. To code out this problem, i mostly used symbolic variables in matlab.

Other methods giving unstable convergence is primal dual optimization.
I have attached different versions of the code in my github repository -
`https://github.com/vighneshn/ConvexOptimEE5121`

8 Plots, values and results

Here, I shall be running simulations for a few parameters, and slight variations on formulation.
A simple variation on the formulation will be changing $\log(k_i)$ to $\log(k_i/n_i)$ where n_i is the population of the state at that time.This will affect the λ_{3i} values, which will all be different.

Note: These plot considers the vulnerable population to be 20% of the total population.

The first three plots are for the daily medicine production for each of the states, along with the daily new cases for each of the states.

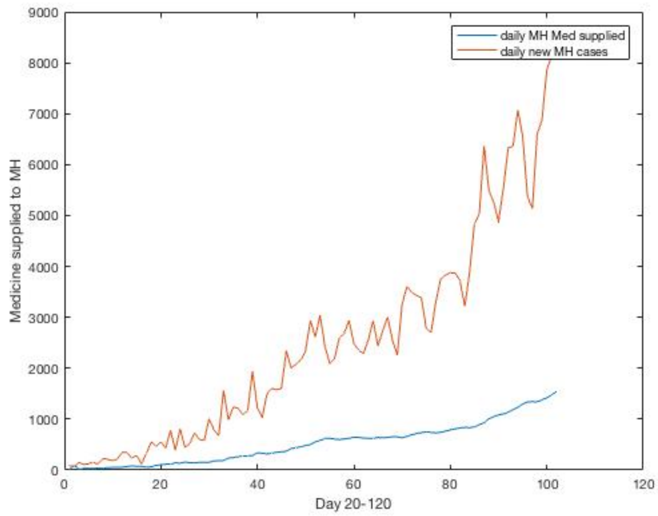

Figure 5: Daily cases and medicines supplied to Maharashtra

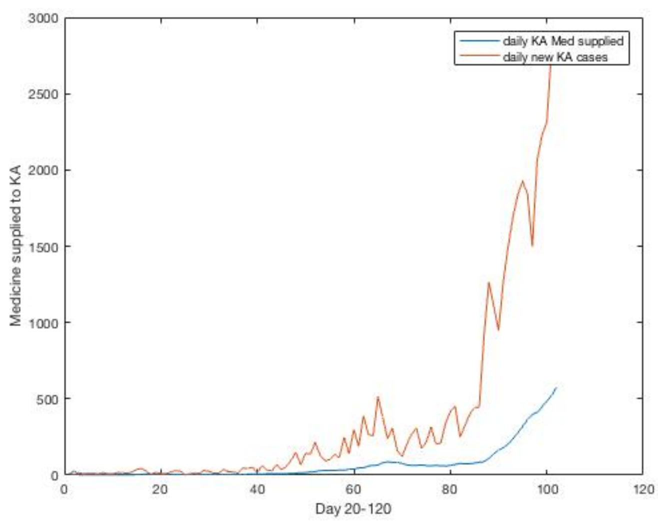

Figure 6: Daily cases and medicines supplied to Maharashtra

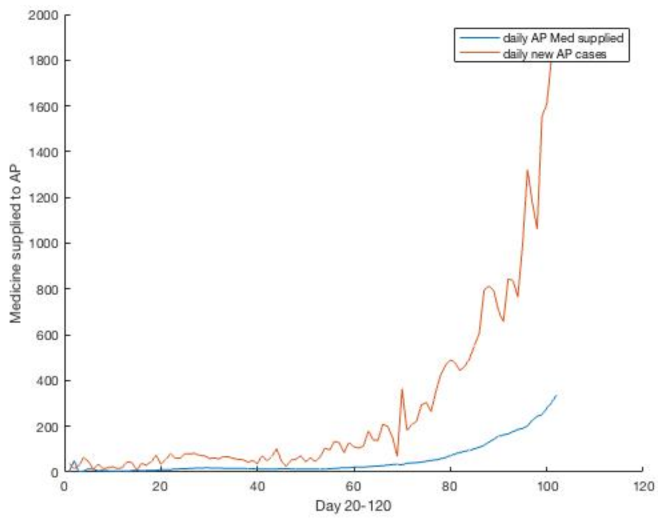

Figure 7: Daily cases and medicines supplied to Maharashtra

The next plot shows the daily medicine supplied to all the states, along with the total supplied to all the states.

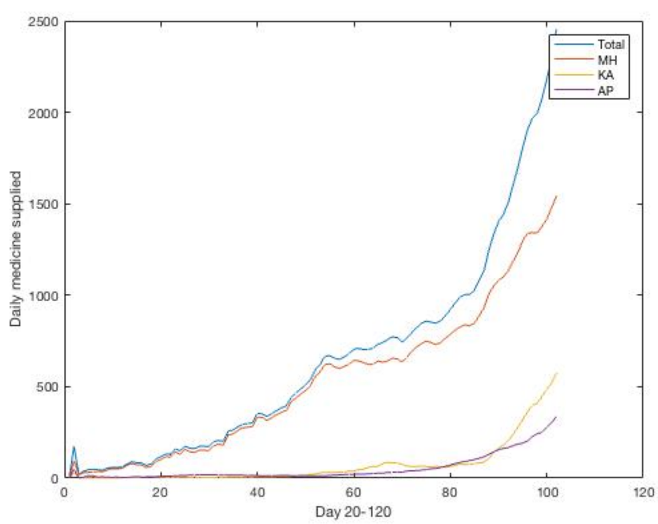

Figure 8: Daily cases and medicines supplied to all the states

The next few plots are cumulative plots. The first one is culumative supplied to each state, the second is the cumulative total medicines supplied

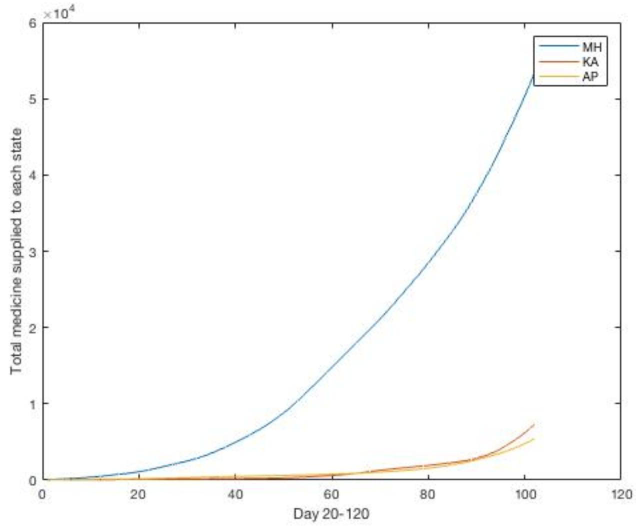

Figure 9: Daily cases and medicines supplied to all the states

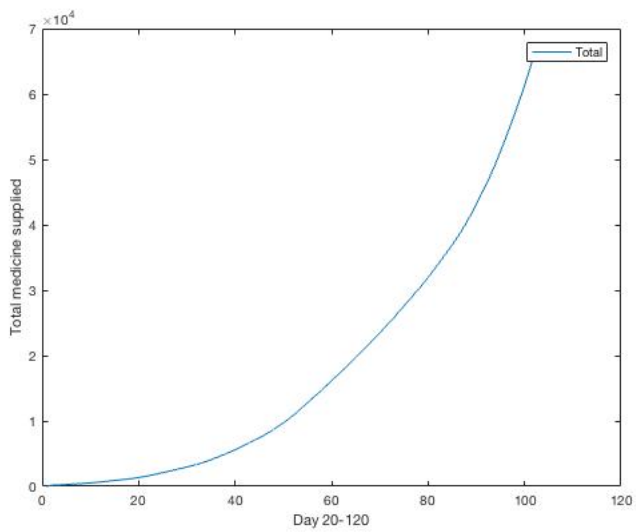

Figure 10: Daily cases and medicines supplied to all the states

9 Regret analysis

The definition of regret (As per the reference Book cited) is

$$\text{regret} = \Sigma_{i=1}^{N} f_t(x_t) \text{ - } \Sigma_{i=1}^{N} f_t(x^*)$$

In my problem, since the optimal point keeps changing, i shall scale down the final split among the states to calculate the appropriate regret at each cycle. In other words, the way I am calculating the regret is:

$$\text{regret} = \Sigma_{i=1}^{N}(M_t z_t) - \Sigma_{i=1}^{N} M_t z^*$$

Where, z_t is the fractional split among all the states, and M_t is the total medicines present at day t. z^* is the final medicine split that has been settled to after all these days, and shall be calculated with knowledge of all the days. This way of framing the regret directly handles the scaling down, and involves all the observations and actions taken.

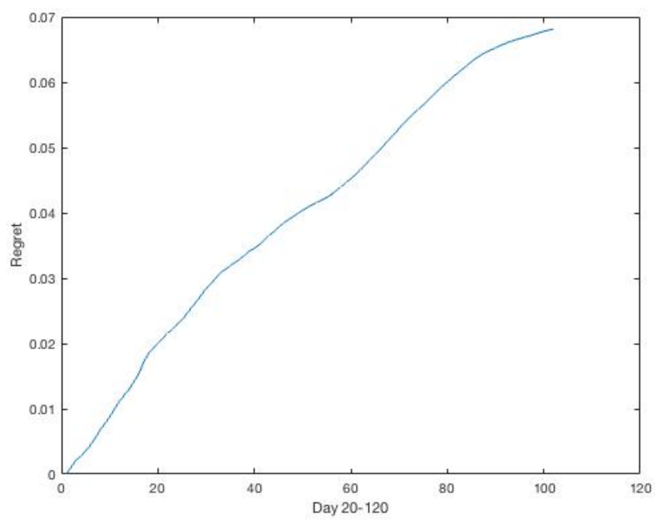

Figure 11: Plot of regret vs days

As we can see from the regret graph, the curve starts to reduce in slope, and with more days will be closer to flattening. This is as expected, with proofs from the textbook showing the regret starts to saturate.

10 Formulation as LP and finding exact dual solution

The exact dual solution can be found by formulating as an LP, and then using the KKT conditions to find the primal optimal solution.

$$Minimizep - \mu\Sigma_{i=1}^{N}(k_i)$$
$$\text{Subject To: } \Sigma_{i=1}^{N} m_i < M + delM$$
$$k_i > \gamma * \beta_i \text{ i = 1...N}$$
$$m_i > k_i e^{\alpha_i(t+1)} \text{ i = 1...N}$$
$$p > b * delM$$
$$delM > 0$$

$$L(m, k, delM, p, \lambda) = (p - \Sigma_{i=1}^{N}(k_i)) + \lambda_1(\Sigma_{i=1}^{N} m_i - M - delM) + \Sigma_{i=1}^{N}(\lambda_{2i}(\gamma\beta_i - k_i)) + \Sigma_{i=1}^{N}(\lambda_{3i}(k_i e^{\alpha_i(t+1)} - m_i)) + \lambda_4(0 - delM) + \lambda_5(b * delM - p)$$

$\lambda_1 = \lambda_{3i} = \lambda$
$\lambda_4 = b - \lambda$
$\lambda_5 = 1$
$\lambda_{2i} = -\mu + \lambda e^{\alpha_i(t+1)}$

$$\phi(\lambda) = -\lambda M + \lambda\Sigma_{i=1}^{N}(\beta_i\gamma e^{\alpha_i(t+1)}) - \Sigma_{i=1}^{N}(\gamma\beta_i\mu)$$

So, the dual optimization problem is

$$\text{maximize: } \phi(\lambda)$$
$$\text{subject to: } \lambda \geq 0$$
$$\lambda \leq b$$
$$\lambda \geq \mu e^{-\alpha_i(t+1)}, \text{ i = 1...N}$$

Clearly here, depending on the constants, the maxima is either at $\lambda = 0$ or at $\lambda = b$ The primal optimal point can then obtained, and the medicines to be distributed are obtained by complimentary slackness property.

11 Simulations for More states

Solution here was taking time to converge through the algorithms described above, and were much more wavering, and not settling, and hence i had to rely on cvx for this part.

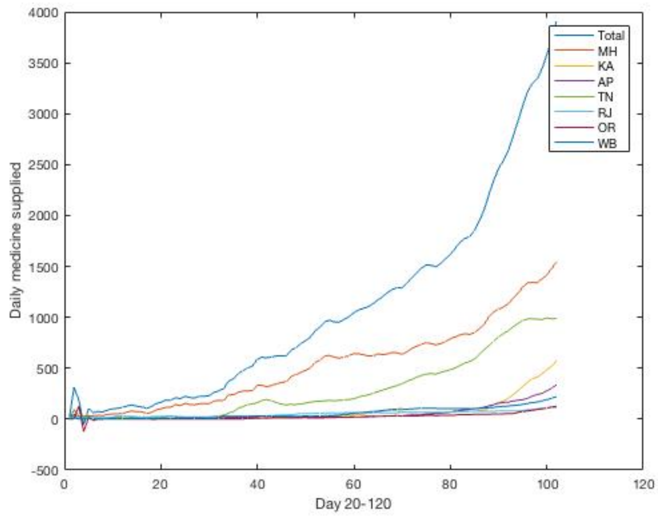

Figure 12: Daily medicine supplied to many states

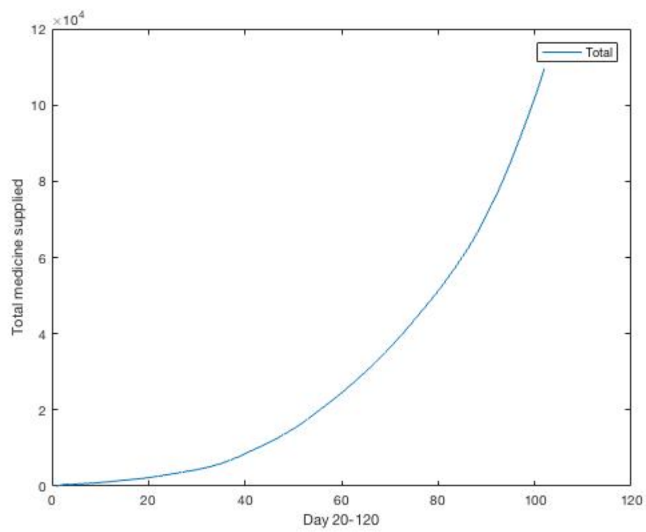

Figure 13: Total medicine supplied to all states

The Initial jitter is due to the initial condition and the time it takes to converge.

12 Simulations for Weekly production

Another possible case scenario is when the medicines are delivered only once a week, and you need to produce for the entire next week, and hence need to be accurate. The regret is expected to be larger.

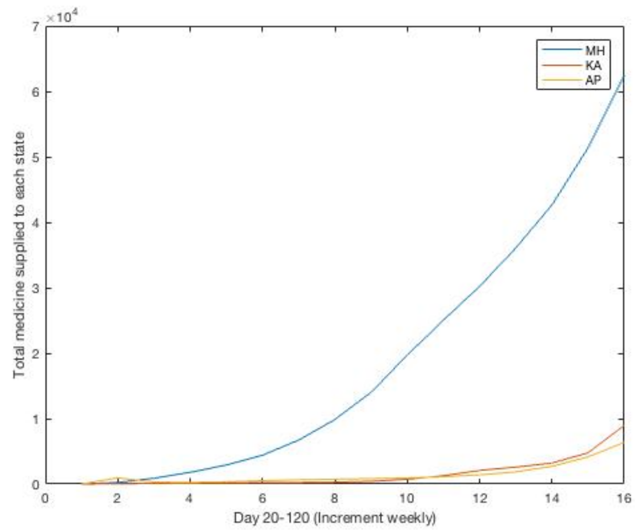

Figure 14: Cumulative statewise medicines supplied weekly

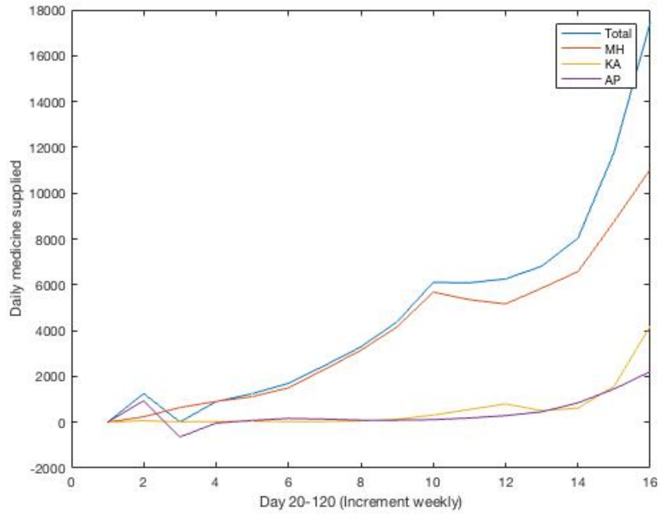

Figure 15: weekly production for each state

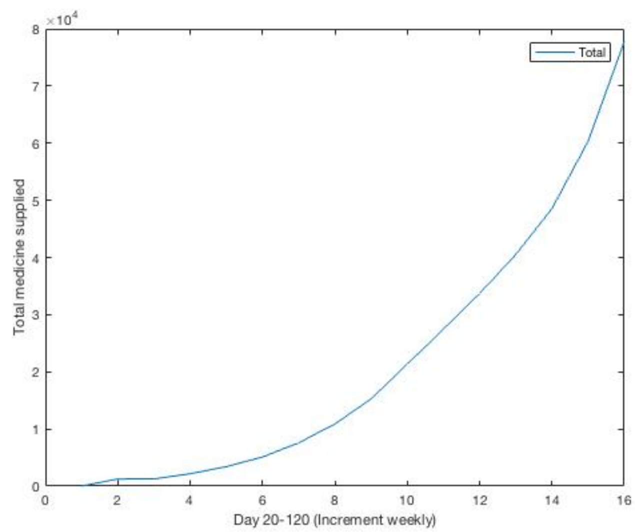

Figure 16: Total medicine supplied to all states

We can see the regret value is much higher, though it saturates quickly.

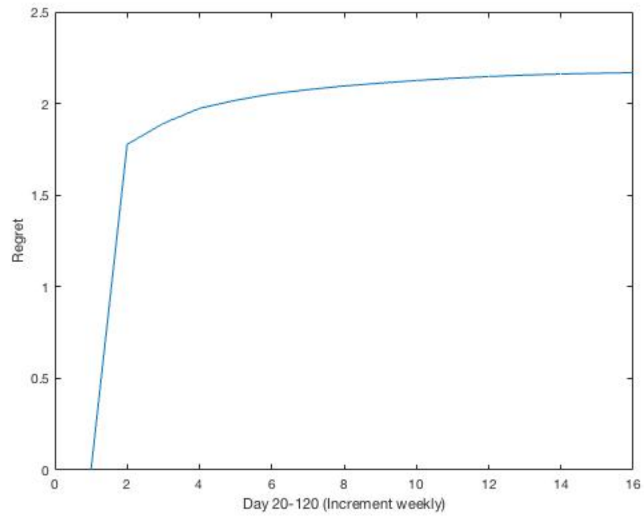

Figure 17: Weekly regret

The overall observation here is since the anticipation is one week ahead, the medicines produced are slightly greater, as all the cases for states start of exponential and then start to slow down. This explains the excess production.

13 Conclusion

The chosen problem has been thoroughly analysed, in both the primal and the dual, with various analysis methods from the book used to tackle the problem.