

EE5121

CVX Assignment Report

Varun Sundar
ee16b068

Indian Institute of Technology Madras

1 Implementation Details

- All questions have been done in **python 3.6**
- Run as : **python q-x.py**
- Install packages as: **pip install -r requirements.txt**
- All plots are saved to the **logs** directory.

2 Q1

We formulate the problem as follows:

$$\min_x \quad \min ||y - x||^2 \quad (1a)$$

$$\text{subject to} \quad ||Ax||_1 \leq b. \quad (1b)$$

This is not in the standard SOCP form. In order to recast it appropriately, we modify it as: (mainly in the epigraph form)

$$\min_t \quad t \quad (2a)$$

$$\text{subject to} \quad ||y - x||^2 < t^2, \quad (2b)$$

$$||a_j^T x||_1 \leq t_i \quad \forall i \in \{1, 2, \dots, n-1\}, \quad (2c)$$

$$\sum_{i=1}^{n-1} t_i \leq b. \quad (2d)$$

We tune b such that only 20 jumps are observed in the resultant estimate. (Notice that $b = 20$ is an overestimate of the constraint, since the L1 convex relaxation is a slack approximation to L0).

Here, A is the matrix as indicated in the question's hint. We observe the following plot with $b = 5.3825$. The error is around 15.211 and the L1 value is around b . Note that as b decreases, the error increases. Hence, there is a trade-off between MSE and jumps in the signal (very much similar to the bias-variance trade-off). So we choose the maximum b which preserves 20 jumps. We

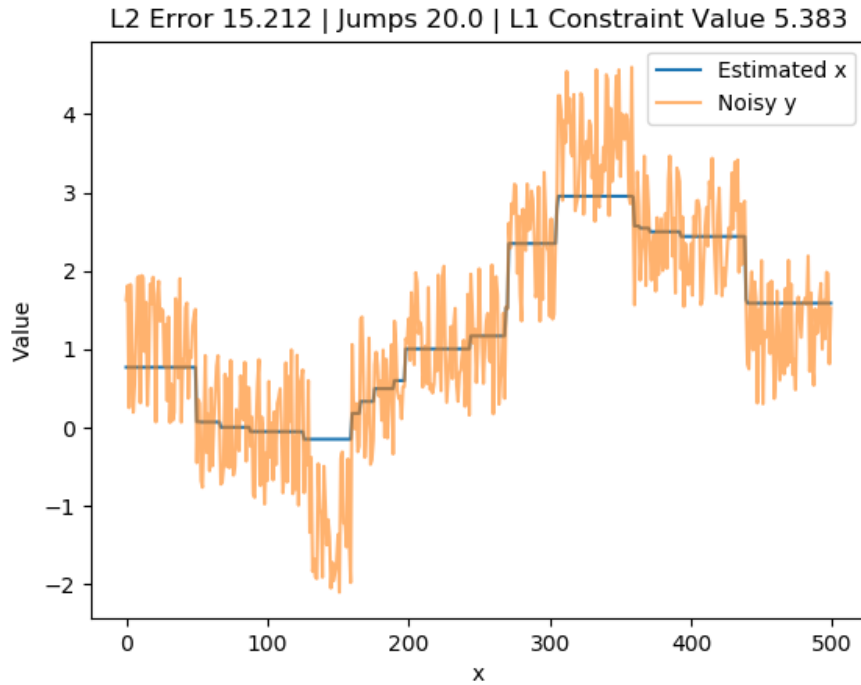


Figure 1: Plot of \hat{x} and noisy y . We have indicated the error and L1 constraint value.

```

Terminal: Local x +
Consumption levels: [ 50.5  94.5 100. 100. 100. ]
INFO - q2-LP - Completed after 0:00:00
(torch) [~/Documents/SEM-6/convex_optim/cvx_assign]$ python q1.py
WARNING - q1-SOCP - No observers have been added to this run
INFO - q1-SOCP - Running command 'main'
INFO - q1-SOCP - Started
(499, 500)
20.0
Status: optimal
L2 Error 15.211570633234606
L1 Constraint Value 5.3825000046806455
L1 Constraint Violation 4.6806452047576386e-09
No of jumps 20.0

```

Figure 2: Screen-shot of question 1

arrive at our chosen value of b via a binary grid-search.

Notice the fact that computing jumps visually is slightly ambiguous, the "jumps" are of order 0.8. So we do this by $\|Ax\|_0$, using numpy, and consider anything > 0.01 as a jump (ie... tolerance is 0.01).

3 Q2

We formulate the following LP:

```

Terminal: Local x +
No of jumps 20.0
INFO - q1-SOCP - Completed after 0:00:30
(torch) [~/Documents/SEM-6/convex_optim/cvx_assign]$ python q2.py
WARNING - q2-LP - No observers have been added to this run
INFO - q2-LP - Running command 'main'
INFO - q2-LP - Started
Status: optimal
Revenue: 192.5
Activity Levels: [ 4. 22.5 31.  1.5]
Revenue per component: [ 12.  32.5 139.  9. ]
Average price per unit for each activity level [3.  1.44444444 4.48387097 6.  ]
Consumption levels: [ 50.5  94.5 100. 100. ]
INFO - q2-LP - Completed after 0:00:00
(torch) [~/Documents/SEM-6/convex_optim/cvx_assign]$

```

Figure 3: Screen-shot of question 2

$$\min_{t,x} \quad 1^T t \quad (3a)$$

$$\text{subject to} \quad t_j \leq p_j * x_j \quad \forall j \in \{1, 2, 3, 4\}, \quad (3b)$$

$$t_j \leq p_j * q_j + p_{disc} * (x_j - q_j) \quad \forall j \in \{1, 2, 3, 4\}, \quad (3c)$$

$$Ax \leq c_{max}, \quad (3d)$$

$$0 \leq x. \quad (3e)$$

We obtain the following optimal values:

- Revenue: 192.5
- Activity Levels: [4, 22.5, 31., 1.5]
- Revenue per component: [12., 32.5, 139., 9.]
- Average price per unit for each activity level: [3., 1.44444444, 4.48387097, 6.]

We make the following observations:

- Consumption levels are met for $c_{3,4,5}$.
- Discounted price is used for $x_{2,3}$. By noting the average price per unit activity.
- Activity levels cross the threshold for $x_{2,3}$.
- Further, revenue per component is max for $x_{2,3}$.

4 Q3

With the convex relaxation of rank and the lemma provided to constrain rank, we obtain the following SDP formulation:

```

Terminal: Local x +
INFO - q2-LP - Completed after 0:00:01
(torch) [~/Documents/SEM-6/convex_optim/cvx_assign]$ python q3.py
WARNING - q3-SDP - No observers have been added to this run
INFO - q3-SDP - Running command 'main'
INFO - q3-SDP - Started
(88, 88)
Status: optimal
Value of r: 337.1089691146928
Actual rank (via np.linalg): 19
INFO - q3-SDP - Completed after 0:00:00
(torch) [~/Documents/SEM-6/convex_optim/cvx_assign]$

```

Figure 4: Screen-shot of question 3

$$\min_{r, \hat{X}, Y, Z} \quad r \quad (4a)$$

$$\text{subject to} \quad \text{tr}(Y) + \text{tr}(Z) \leq 2r \quad (4b)$$

$$\hat{X}_{i,j} = X_{i,j} \quad \forall i, j \in J \quad \text{set of non-zero entries of } X, \quad (4c)$$

$$\begin{pmatrix} Y & X \\ X^T & Z \end{pmatrix} \geq 0, \quad (4d)$$

$$Y \geq 0. \quad (4e)$$

$$Z \geq 0. \quad (4f)$$

We obtain the rank to be 19, by using **np.linalg.matrix_rank** by thresholding to $1e - 6$.

Note that the optimisation formulation is cast in the epigraph form - we may also eliminate r entirely and recast as:

$$\min_{\hat{X}, Y, Z} \quad \text{tr}(Y) + \text{tr}(Z) \quad (5a)$$

$$\hat{X}_{i,j} = X_{i,j} \quad \forall i, j \in J \quad \text{set of non-zero entries of } X, \quad (5b)$$

$$\begin{pmatrix} Y & X \\ X^T & Z \end{pmatrix} \geq 0, \quad (5c)$$

$$Y \geq 0. \quad (5d)$$

$$Z \geq 0. \quad (5e)$$