

# CS 325: Project 3

---

Colin Bradford

Charles Jenkins

Albert Le

May 25, 2015

## 1 PROBLEM 1

### **Part A:**

#### **i. Linear Problem Formulation**

Let each route be represented as variables of the form "SourceDestination." For example a route from Plant 1 to Warehouse 1 would be "p1w1."

Objective function:

Minimize:

$$10 \text{ p1w1} + 15 \text{ p1w2} + 11 \text{ p2w1} + 8 \text{ p2w2} + 13 \text{ p3w1} + 8 \text{ p3w2} + 9 \text{ p3w3} + 14 \text{ p4w2} + 8 \text{ p4w3} + 5 \text{ w1r1} + 6 \text{ w1r2} + 7 \text{ w1r3} + 10 \text{ w1r4} + 12 \text{ w2r3} + 8 \text{ w2r4} + 10 \text{ w2r5} + 14 \text{ w2r6} + 14 \text{ w3r4} + 12 \text{ w3r5} + 12 \text{ w3r6} + 6 \text{ w3r7}$$

Subject To:

Supply Constraints:

$$\text{p1w1} + \text{p1w2} = 150$$

$$\text{p2w1} + \text{p2w2} = 450$$

$$\text{p3w1} + \text{p3w2} + \text{p3w3} = 250$$

$$\text{p4w2} + \text{p4w3} = 150$$

Demand Constraints:

$$w1r1 = 100$$

$$w1r2 = 150$$

$$w1r3 + w2r3 = 100$$

$$w1r4 + w2r4 + w3r4 = 200$$

$$w2r5 + w3r5 = 200$$

$$w2r6 + w3r6 = 150$$

$$w3r7 = 100$$

Balancing Constraints:

$$p1w1 + p2w1 + p3w1 - w1r1 - w1r2 - w1r3 - w1r4 \leq 0$$

$$p1w2 + p2w2 + p3w2 + p4w2 - w2r3 - w2r4 - w2r5 - w2r6 \leq 0$$

$$p3w3 + p4w3 - w3r4 - w3r5 - w3r6 - w3r7 \leq 0$$

Non-negativity Constraints:

$$p1w1, p1w2, p2w1, p2w2, p3w1, p3w2, p3w3, p4w2, p4w3, w1r1, w1r2, w1r3, w1r4, w2r3, w2r4, w2r5, w2r6, w3r4, w3r5, w3r6, w3r7 \geq 0$$

## ii. LINDO Code and Output

```
MIN 10 p1w1 + 15 p1w2 + 11 p2w1 + 8 p2w2 + 13 p3w1 + 8 p3w2 + 9 p3w3
    + 14 p4w2 + 8 p4w3 + 5 w1r1 + 6 w1r2 + 7 w1r3 + 10 w1r4 + 12 w2r3
    + 8 w2r4 + 10 w2r5 + 14 w2r6 + 14 w3r4 + 12 w3r5 + 12 w3r6 + 6 w3r7
ST
    p1w1 + p1w2 = 150
    p2w1 + p2w2 = 450
    p3w1 + p3w2 + p3w3 = 250
    p4w2 + p4w3 = 150
    w1r1 = 100
    w1r2 = 150
    w1r3 + w2r3 = 100
    w1r4 + w2r4 + w3r4 = 200
    w2r5 + w3r5 = 200
    w2r6 + w3r6 = 150
    w3r7 = 100
    p1w1 + p2w1 + p3w1 - w1r1 - w1r2 - w1r3 - w1r4 <= 0
    p1w2 + p2w2 + p3w2 + p4w2 - w2r3 - w2r4 - w2r5 - w2r6 <= 0
    p3w3 + p4w3 - w3r4 - w3r5 - w3r6 - w3r7 <= 0
    p1w1 > 0
    p1w2 > 0
    p2w1 > 0
    p2w2 > 0
    p3w1 > 0
    p3w2 > 0
```

```

p3w3 > 0
p4w2 > 0
p4w3 > 0
w1r1 > 0
w1r2 > 0
w1r3 > 0
w1r4 > 0
w2r3 > 0
w2r4 > 0
w2r5 > 0
w2r6 > 0
w3r4 > 0
w3r5 > 0
w3r6 > 0
w3r7 > 0
END

```

LP OPTIMUM FOUND AT STEP 13

OBJECTIVE FUNCTION VALUE

1) 17100.00

VARIABLE	VALUE	REDUCED COST
P1W1	150.000000	0.000000
P1W2	0.000000	8.000000
P2W1	200.000000	0.000000
P2W2	250.000000	0.000000
P3W1	0.000000	2.000000
P3W2	150.000000	0.000000
P3W3	100.000000	0.000000
P4W2	0.000000	7.000000
P4W3	150.000000	0.000000
W1R1	100.000000	0.000000
W1R2	150.000000	0.000000
W1R3	100.000000	0.000000
W1R4	0.000000	5.000000
W2R3	0.000000	2.000000
W2R4	200.000000	0.000000
W2R5	200.000000	0.000000
W2R6	0.000000	1.000000
W3R4	0.000000	7.000000
W3R5	0.000000	3.000000

W3R6	150.000000	0.000000
W3R7	100.000000	0.000000

ROW	SLACK OR SURPLUS	DUAL PRICES
2)	0.000000	-10.000000
3)	0.000000	-11.000000
4)	0.000000	-11.000000
5)	0.000000	-10.000000
6)	0.000000	-5.000000
7)	0.000000	-6.000000
8)	0.000000	-7.000000
9)	0.000000	-5.000000
10)	0.000000	-7.000000
11)	0.000000	-10.000000
12)	0.000000	-4.000000
13)	0.000000	0.000000
14)	0.000000	3.000000
15)	0.000000	2.000000
16)	150.000000	0.000000
17)	0.000000	0.000000
18)	200.000000	0.000000
19)	250.000000	0.000000
20)	0.000000	0.000000
21)	150.000000	0.000000
22)	100.000000	0.000000
23)	0.000000	0.000000
24)	150.000000	0.000000
25)	100.000000	0.000000
26)	150.000000	0.000000
27)	100.000000	0.000000
28)	0.000000	0.000000
29)	0.000000	0.000000
30)	200.000000	0.000000
31)	200.000000	0.000000
32)	0.000000	0.000000
33)	0.000000	0.000000
34)	0.000000	0.000000
35)	150.000000	0.000000
36)	100.000000	0.000000

NO. ITERATIONS= 13

### iii. Optimal Shipping Routes and Minimum Cost

The optimal shipping routes and quantity of refrigerators per route are:

Route	Refrigerators
P1W1	150
P2W1	200
P2W2	250
P3W2	150
P3W3	100
P4W3	150
W1R1	100
W1R2	150
W1R3	100
W2R4	200
W2R5	200
W3R6	150
W3R7	100

The optimal minimum cost is \$17,100

#### Part B:

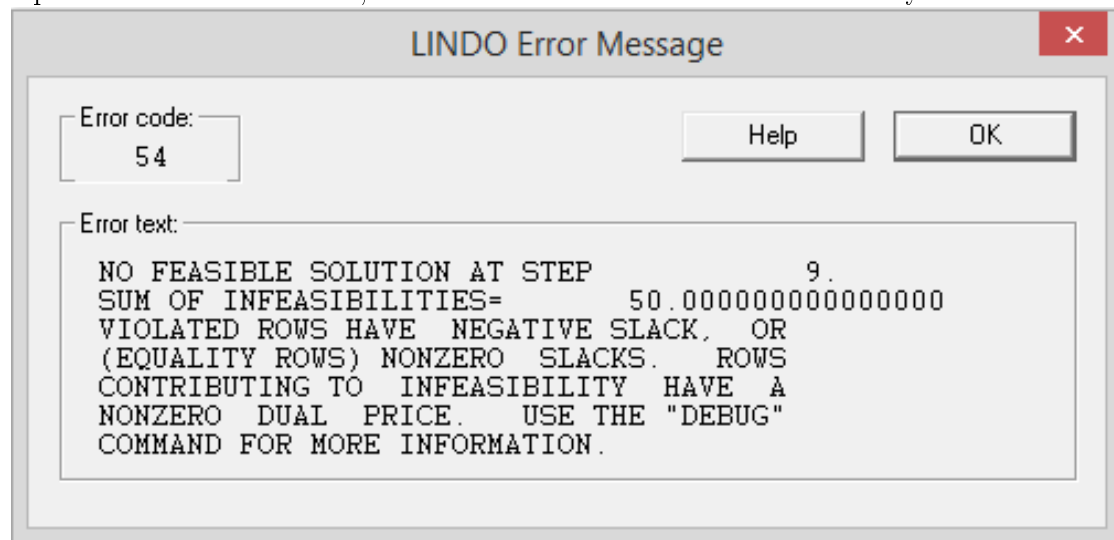
To remove Warehouse 2 and all its associated routes, we remove all variables in our equation that contain "w2" in it. The resulting LINDO code is as follows:

```
MIN 10 p1w1 + 11 p2w1 + 13 p3w1 + 9 p3w3 + 8 p4w3 + 5 w1r1 + 6 w1r2
    + 7 w1r3 + 10 w1r4 + 14 w3r4 + 12 w3r5 + 12 w3r6 + 6 w3r7
ST
p1w1 = 150
p2w1 = 450
p3w1 + p3w3 = 250
p4w3 = 150
w1r1 = 100
w1r2 = 150
w1r3 = 100
w1r4 + w3r4 = 200
w3r5 = 200
w3r6 = 150
w3r7 = 100
p1w1 + p2w1 + p3w1 - w1r1 - w1r2 - w1r3 - w1r4 <= 0
p3w3 + p4w3 - w3r4 - w3r5 - w3r6 - w3r7 <= 0
p1w1 > 0
p2w1 > 0
p3w1 > 0
```

```
p3w3 > 0
p4w3 > 0
w1r1 > 0
w1r2 > 0
w1r3 > 0
w1r4 > 0
w3r4 > 0
w3r5 > 0
w3r6 > 0
w3r7 > 0
```

END

Upon execution of the code, we are met with these solution infeasibility notices:



LINDO Solver Status

×

Optimizer Status

Status:Infeasible

Iterations:4

Infeasibility:50

Objective:18800

Best IP:N/A

IP Bound:N/A

Branches:N/A

Elapsed Time:00:00:18

Update Interval:

1

Interrupt Solver

Close

There is no feasible solution here because Warehouse 3's total possible supply (400) is not capable of supporting the combined demand of R4, R5, R6, and R7 (650). On the other hand, Warehouse 1 is in fact capable of supporting R1, R2, R3, and R4 with its 850 supply to 550 demand ratio, but that still does not solve the larger supply chain deficit and feasibility issue. Overall, we need Warehouse 2 to supplement the supply to

R3, R4, R5, and R6 and sufficiently "spread the load" on the warehouses.

### Part C:

To simulate the limitation of 100 shipments (inbound and outbound) on Warehouse 2, we add two new constraints to our code for Part A. The constraints limit the inbound and outbound shipments to being less than or equal to 100.

LINDO Code and Output:

```
MIN 10 p1w1 + 15 p1w2 + 11 p2w1 + 8 p2w2 + 13 p3w1 + 8 p3w2 + 9 p3w3
    + 14 p4w2 + 8 p4w3 + 5 w1r1 + 6 w1r2 + 7 w1r3 + 10 w1r4 + 12 w2r3
    + 8 w2r4 + 10 w2r5 + 14 w2r6 + 14 w3r4 + 12 w3r5 + 12 w3r6 + 6 w3r7
ST
    p1w1 + p1w2 = 150
    p2w1 + p2w2 = 450
    p3w1 + p3w2 + p3w3 = 250
    p4w2 + p4w3 = 150
    w1r1 = 100
    w1r2 = 150
    w1r3 + w2r3 = 100
    w1r4 + w2r4 + w3r4 = 200
    w2r5 + w3r5 = 200
    w2r6 + w3r6 = 150
    w3r7 = 100
    p1w1 + p2w1 + p3w1 - w1r1 - w1r2 - w1r3 - w1r4 <= 0
    p1w2 + p2w2 + p3w2 + p4w2 - w2r3 - w2r4 - w2r5 - w2r6 <= 0
    p3w3 + p4w3 - w3r4 - w3r5 - w3r6 - w3r7 <= 0
    p1w1 > 0
    p1w2 > 0
    p2w1 > 0
    p2w2 > 0
    p3w1 > 0
    p3w2 > 0
    p3w3 > 0
    p4w2 > 0
    p4w3 > 0
    w1r1 > 0
    w1r2 > 0
    w1r3 > 0
    w1r4 > 0
    w2r3 > 0
    w2r4 > 0
    w2r5 > 0
```



```

w2r6 > 0
w3r4 > 0
w3r5 > 0
w3r6 > 0
w3r7 > 0
p1w2 + p2w2 + p3w2 + p4w2 <= 100
w2r3 + w2r4 + w2r5 + w2r6 <= 100
END

```

LP OPTIMUM FOUND AT STEP 7

OBJECTIVE FUNCTION VALUE

1) 18300.00

VARIABLE	VALUE	REDUCED COST
P1W1	150.000000	0.000000
P1W2	0.000000	8.000000
P2W1	350.000000	0.000000
P2W2	100.000000	0.000000
P3W1	0.000000	4.000000
P3W2	0.000000	2.000000
P3W3	250.000000	0.000000
P4W2	0.000000	9.000000
P4W3	150.000000	0.000000
W1R1	100.000000	0.000000
W1R2	150.000000	0.000000
W1R3	100.000000	0.000000
W1R4	150.000000	0.000000
W2R3	0.000000	7.000000
W2R4	50.000000	0.000000
W2R5	50.000000	0.000000
W2R6	0.000000	4.000000
W3R4	0.000000	4.000000
W3R5	150.000000	0.000000
W3R6	150.000000	0.000000
W3R7	100.000000	0.000000

ROW	SLACK OR SURPLUS	DUAL PRICES
2)	0.000000	-20.000000
3)	0.000000	-21.000000
4)	0.000000	-19.000000

5)	0.000000	-18.000000
6)	0.000000	5.000000
7)	0.000000	4.000000
8)	0.000000	3.000000
9)	0.000000	0.000000
10)	0.000000	-2.000000
11)	0.000000	-2.000000
12)	0.000000	4.000000
13)	0.000000	10.000000
14)	0.000000	13.000000
15)	0.000000	10.000000
16)	150.000000	0.000000
17)	0.000000	0.000000
18)	350.000000	0.000000
19)	100.000000	0.000000
20)	0.000000	0.000000
21)	0.000000	0.000000
22)	250.000000	0.000000
23)	0.000000	0.000000
24)	150.000000	0.000000
25)	100.000000	0.000000
26)	150.000000	0.000000
27)	100.000000	0.000000
28)	150.000000	0.000000
29)	0.000000	0.000000
30)	50.000000	0.000000
31)	50.000000	0.000000
32)	0.000000	0.000000
33)	0.000000	0.000000
34)	150.000000	0.000000
35)	150.000000	0.000000
36)	100.000000	0.000000
37)	0.000000	0.000000
38)	0.000000	5.000000

NO. ITERATIONS= 7

We see that this is a feasible solution, though it raises our optimal cost from \$17,100 to \$18,300.

The optimal shipping routes and quantity of refrigerators per route are:

Route	Refrigerators
P1W1	150
P2W1	350
P2W2	100

P3W3	250
P4W3	150
W1R1	100
W1R2	150
W1R3	100
W1R4	150
W2R4	50
W2R5	50
W3R5	150
W3R6	150
W3R7	100

**Part D:**

For some cost  $c_{ab}$  and some route  $x_{ab}$  where their subscripts represent the route from a to b, the objective function is:

$$\text{Min}(\sum_p \sum_w c_{pw} x_{pw} + \sum_w \sum_r c_{wr} x_{wr})$$

*s.t.*

(Supply Constraint) For each source node p, where s is supply:  $\sum_w x_{pw} = s_p$

(Demand Constraint) For each destination node r, where d is demand:  $\sum_w x_{wr} = d_r$

(Balancing Constraint) For each intermediate node w:  $\sum_p x_{pw} - \sum_r x_{wr} \leq 0$

(Non-negativity Constraint) For all nodes p and r:  $x_{pw}, x_{wr} \geq 0$

## 2 PROBLEM 2

**Part A:**

- i.
- ii.
- iii.

**Part B:**

- i.
- ii.
- iii.

**Part C:**

- i.
- ii.
- iii.

## 3 PROBLEM 3

**Part A:**

## i. General Problem Statement

Objective:

$\text{Min } \sum_{i=1}^n b_i$ , for some  $b$  representing  $\sum_{i=1}^n |y_i - (a_1 x_i + a_0)|$

s.t.

For each  $i = 1 \dots n$ ,  $b_i - (y_i - (a_1 x_i + a_0)) \geq 0$

For each  $i = 1 \dots n$ ,  $b_i + (y_i - (a_1 x_i + a_0)) \geq 0$

## ii. LINDO Code and Output

```
MIN b1 + b2 + b3 + b4 + b5 + b6 + b7
```

```
ST
```

```
b1 + 1 a1 + a0 >= 5
b1 - 1 a1 - a0 >= -5
b2 + 1 a1 + a0 >= 3
b2 - 1 a1 - a0 >= -3
b3 + 2 a1 + a0 >= 13
b3 - 2 a1 - a0 >= -13
b4 + 3 a1 + a0 >= 8
b4 - 3 a1 - a0 >= -8
b5 + 4 a1 + a0 >= 10
b5 - 4 a1 - a0 >= -10
b6 + 5 a1 + a0 >= 14
b6 - 5 a1 - a0 >= -14
b7 + 6 a1 + a0 >= 18
b7 - 6 a1 - a0 >= -18
b1 > 0
b2 > 0
b3 > 0
b4 > 0
b5 > 0
b6 > 0
b7 > 0
```

```
END
```

```
LP OPTIMUM FOUND AT STEP 7
```

```
OBJECTIVE FUNCTION VALUE
```

```
1) 12.25000
```

VARIABLE	VALUE	REDUCED COST
B1	2.000000	0.000000
B2	0.000000	0.250000

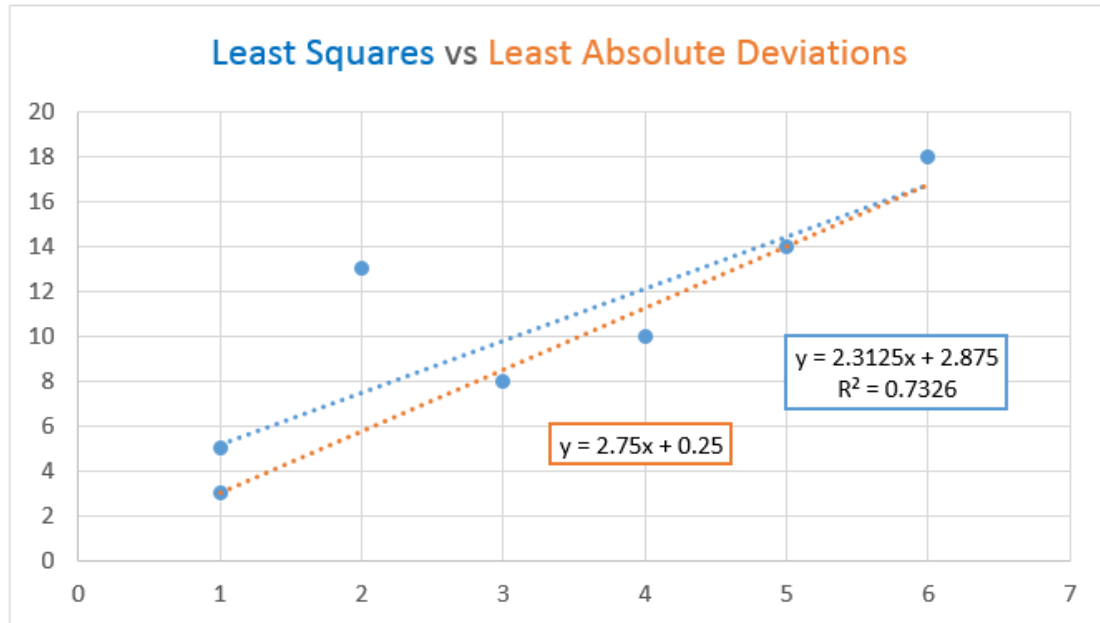
B3	7.250000	0.000000
B4	0.500000	0.000000
B5	1.250000	0.000000
B6	0.000000	0.750000
B7	1.250000	0.000000
A1	2.750000	0.000000
A0	0.250000	0.000000

ROW	SLACK OR SURPLUS	DUAL PRICES
2)	0.000000	-1.000000
3)	4.000000	0.000000
4)	0.000000	0.000000
5)	0.000000	-0.750000
6)	0.000000	-1.000000
7)	14.500000	0.000000
8)	1.000000	0.000000
9)	0.000000	-1.000000
10)	2.500000	0.000000
11)	0.000000	-1.000000
12)	0.000000	0.000000
13)	0.000000	-0.250000
14)	0.000000	-1.000000
15)	2.500000	0.000000
16)	2.000000	0.000000
17)	0.000000	0.000000
18)	7.250000	0.000000
19)	0.500000	0.000000
20)	1.250000	0.000000
21)	0.000000	0.000000
22)	1.250000	0.000000

NO. ITERATIONS= 7

The sum of the absolute deviations was 12.25.

### iii. Graph



The LAD approach creates a line that has a steeper slope and lesser y-intercept than the least squares line. The line more tightly follows the path of the majority of the data points and seems to be less affected by the outlier at  $x = 2$ .

## Part B:

### i. General Problem Statement

Objective:

$\text{Min } b$ , for some  $b$  representing  $\text{Max}|y_i - (a_1x_i + a_0)|$

s.t.

For each  $i = 1 \dots n$ ,  $b + a_1x_i + a_0 \geq y_i$

For each  $i = 1 \dots n$ ,  $b - a_1x_i - a_0 \geq -y_i$

### ii. LINDO Code and Output

MIN  $b$

ST

$b + 1 a_1 + a_0 \geq 5$

$b - 1 a_1 - a_0 \geq -5$

$b + 1 a_1 + a_0 \geq 3$

$b - 1 a_1 - a_0 \geq -3$

$b + 2 a_1 + a_0 \geq 13$

$b - 2 a_1 - a_0 \geq -13$

$b + 3 a_1 + a_0 \geq 8$

$b - 3 a_1 - a_0 \geq -8$

$b + 4 a_1 + a_0 \geq 10$

```

b - 4 a1 - a0 >= -10
b + 5 a1 + a0 >= 14
b - 5 a1 - a0 >= -14
b + 6 a1 + a0 >= 18
b - 6 a1 - a0 >= -18
b > 0
END

LP OPTIMUM FOUND AT STEP      3

      OBJECTIVE FUNCTION VALUE

    1)      3.833333

VARIABLE           VALUE          REDUCED COST
    B              3.833333           0.000000
    A1              2.333333           0.000000
    A0              4.500000           0.000000

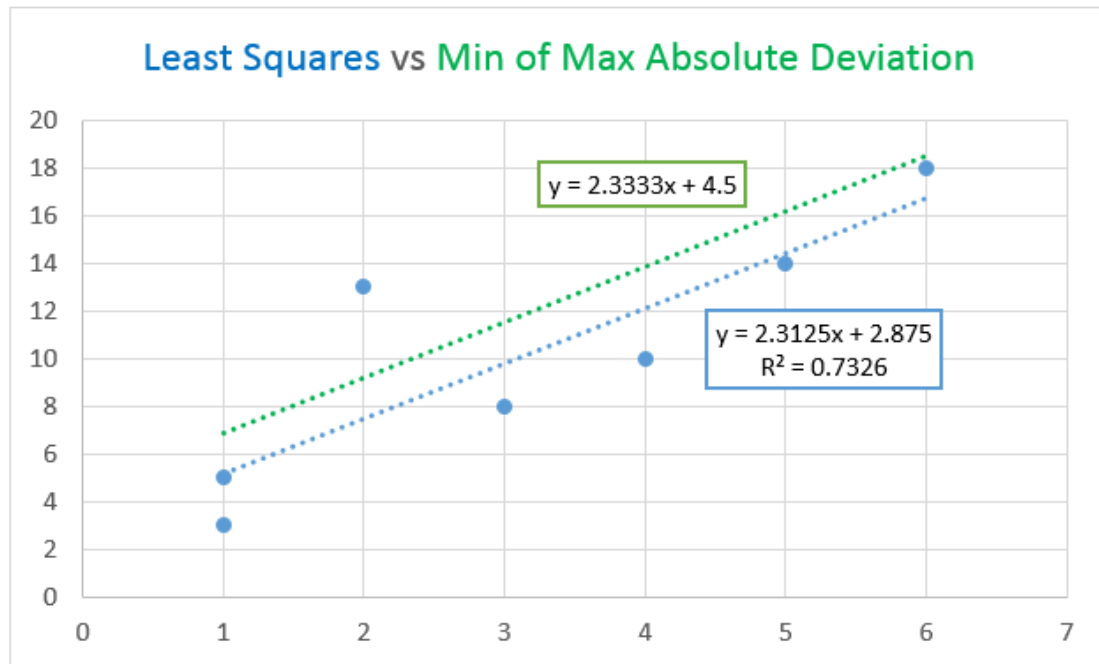
      ROW    SLACK OR SURPLUS      DUAL PRICES
    2)           5.666667           0.000000
    3)           2.000000           0.000000
    4)           7.666667           0.000000
    5)           0.000000          -0.333333
    6)           0.000000          -0.500000
    7)           7.666667           0.000000
    8)           7.333333           0.000000
    9)           0.333333           0.000000
   10)           7.666667           0.000000
   11)           0.000000          -0.166667
   12)           6.000000           0.000000
   13)           1.666667           0.000000
   14)           4.333333           0.000000
   15)           3.333333           0.000000
   16)           3.833333           0.000000

NO. ITERATIONS=           3

```

The min of the max absolute deviations was 3.833333.

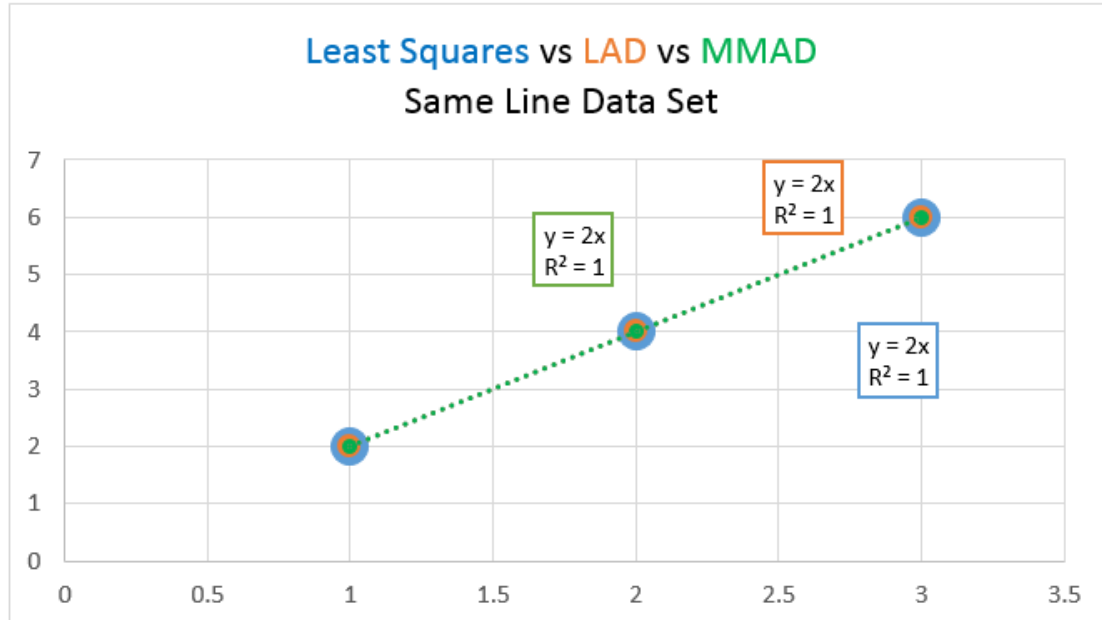
### iii. Graph



The MMAD approach gives a line that appears to be more affected by the outlier at  $x = 2$  with a much higher y-intercept than the LAD and least squares approaches. It's slope however, is only slightly steeper than the least squares line.

#### iv. Same Line Data Set

Yes, the data set  $(x,y) = [(1,2), (2,4), (3,6)]$  computes the same line for all three methods.





LAD Method:

MIN b1 + b2 + b3

ST

b1 + 1 a1 + a0 >= 2  
b1 - 1 a1 - a0 >= -2  
b2 + 2 a1 + a0 >= 4  
b2 - 2 a1 - a0 >= -4  
b3 + 3 a1 + a0 >= 6  
b3 - 3 a1 - a0 >= -6  
b1 > 0  
b2 > 0  
b3 > 0

END

LP OPTIMUM FOUND AT STEP 1

OBJECTIVE FUNCTION VALUE

1) 0.0000000E+00

VARIABLE	VALUE	REDUCED COST
B1	0.000000	1.000000
B2	0.000000	1.000000
B3	0.000000	1.000000
A1	2.000000	0.000000
A0	0.000000	0.000000

ROW	SLACK OR SURPLUS	DUAL PRICES
2)	0.000000	0.000000
3)	0.000000	0.000000
4)	0.000000	0.000000
5)	0.000000	0.000000
6)	0.000000	0.000000
7)	0.000000	0.000000
8)	0.000000	0.000000
9)	0.000000	0.000000
10)	0.000000	0.000000

NO. ITERATIONS= 1

MMAD Method:

```

MIN b
ST
  b + 1 a1 + a0 >= 2
  b - 1 a1 - a0 >= -2
  b + 2 a1 + a0 >= 4
  b - 2 a1 - a0 >= -4
  b + 3 a1 + a0 >= 6
  b - 3 a1 - a0 >= -6
  b > 0
END

LP OPTIMUM FOUND AT STEP      1

```

#### OBJECTIVE FUNCTION VALUE

1)           0.0000000E+00

VARIABLE	VALUE	REDUCED COST
B	0.000000	1.000000
A1	2.000000	0.000000
A0	0.000000	0.000000

ROW	SLACK OR SURPLUS	DUAL PRICES
2)	0.000000	0.000000
3)	0.000000	0.000000
4)	0.000000	0.000000
5)	0.000000	0.000000
6)	0.000000	0.000000
7)	0.000000	0.000000
8)	0.000000	0.000000

NO. ITERATIONS=           1

Least Squares Line was produced in Excel (see graph above) rather than LINDO.

#### Part C: