Solving a regression problem by linear programming

A very common and important problem in statistics is linear regression, the problem of fitting a straight line to statistical data. The most commonly employed technique is the method of least squares, but there are other interesting criteria where linear programming can be used to solve for the optimal values of the regression parameters.

Let $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ be data points and a_1 and a_0 be the parameters of the regression line $y=a_1x+a_0$.

(a) Formulate a linear program whose optimal solution minimizes the sum of the absolute deviations of the data from the line, i.e., formulate

$$\min_a \sum_{i=1}^{n} |y_i - (a_1 x_i + a_0)|$$

as an LP.

(b) Formulate the minimization of the maximum absolute deviation as an LP, i.e., formulate

$$\min_a \max_i |y_i - (a_1 x_i + a_0)|$$

as an LP.

(c) Generalize the model to allow fitting to general polynomials $y = a_k x^k + a_{k-1} x^{k-1} + \ldots + a_1 x + a_0$.

Solution

The difficulty here lies in the fact that he optimization problem as it is stated in the problem set is not linear: the absolute value or the maximum functions are not linear. So we need to reformulate these somehow using simple tricks that make the problems linear.

a) Note that our goal is to find values for a_1 and a_0 which minimize $\sum_{i=1}^{n} |y_i - (a_1 x_i + a_0)|$.

Thus, a_1 and a_0 are variables, and x_i's and y_i's are given data. However, the above function is not linear. To make it linear, we need to introduce new variables. For i=1,…,n, let $z_i = |y_i - (a_1 x_i + a_0)|$. Then the new model is:

Minimize $\sum_{i=1}^{n} z_i$

subject to $z_i = |y_i - (a_1 x_i + a_0)|$, for each i=1,…,n

However, now we have non-linear functions in the constraints.

Suppose for each i=1,…n, we substitute $z_i = |y_i - (a_1 x_i + a_0)|$ by a pair of related constraints:

$z_i \geq y_i - (a_1 x_i + a_0)$ (1)

and $z_i \geq -y_i + (a_1 x_i + a_0)$ (2)

Note that (1) and (2) provide that $z_i \geq |y_i - (a_1 x_i + a_0)|$. But since our model is trying to minimize z_i's, in the optimal solution the value of each z_i will be taken all the way down to $|y_i - (a_1 x_i + a_0)|$. Summarizing, the linear program is:

Minimize $\sum_{i=1}^{n} z_i$

subject to $z_i \geq y_i - (a_1 x_i + a_0)$, for each i=1,…,n (1)

$z_i \geq -y_i + (a_1 x_i + a_0)$, for each i=1,…,n (2)

b) We want to $\min_a \max_i |y_i - (a_1 x_i + a_0)|$. a_1 and a_0 are variables, and x_i's and y_i's are given data. But the maximum of absolute values is not a linear function. To make it linear, we need to introduce a new variable. Let $z = \max_i |y_i - (a_1 x_i + a_0)|$. Then the new model is:

Minimize z

subject to $z = \max_i |y_i - (a_1 x_i + a_0)|$

Now we have a non-linear function in the constraint. However, the following equivalent formulation takes care of that problem.

Minimize z

subject to $z \geq y_i - (a_1 x_i + a_0)$, for each i=1,…,n (1)

$z \geq -y_i + (a_1 x_i + a_0)$, for each i=1,…,n (2)

Note that (1) and (2) provide that $z \geq \max_i |y_i - (a_1 x_i + a_0)|$. But since our model is trying to minimize z, in the optimal solution the value of each z will be taken all the way down to $\max_i |y_i - (a_1 x_i + a_0)|$.

c) Just replace ($a_1 x_i + a_0$) above with ($a_k x_i^k + a_{k-1} x^{k-1}_i + \ldots + a_1 x_i + a_0$).

AMPL (software for solving linear programs) model for part (a)

```
set Points;

param x{Points};

param y{Points};
```

```
var a1;

var a0;

var z{Points};


minimize obj: sum{i in Points} z[i];


s.t. c1{i in Points}: z[i] >= y[i]-(a1*x[i]+a0);


s.t. c2{i in Points}: z[i] >= -y[i]+(a1*x[i]+a0);


data;


set Points := 1 2 3 4;


param: x        y   :=

1       1       4

2       2       6

3       3       6

4       4       8 ;
```

AMPL model for part (b)

```
set Points;

param x{Points};

param y{Points};


var a1;

var a0;

var z;


minimize obj: z;


s.t. c1{i in Points}: z >= y[i]-(a1*x[i]+a0);


s.t. c2{i in Points}: z >= -y[i]+(a1*x[i]+a0);


data;


set Points := 1 2 3 4;


param: x        y   :=

1       1       4

2       2       6
```

3 3 6

4 4 8 ;