Math 3660 - Spring 2022
Mathematical Models in Economics

Steven Tschantz

2/20/22

Model Estimation

Assignment 12

Instructions

Save a copy of this notebook, complete the exercises, save and submit on Brightspace your final version by start of class Tues. Apr. 19.

Nikhil Jayswal

Edit the line above that reads Enter your name here, inserting instead your name. You may also want to save to a file name that includes your name. By adding your name into the file and file name, you reduce the chance I will mix-up your submission with someone else's.

Exercises

1. Suppose we want to fit a linear model $y = \beta_1 + \beta_2\, x + \epsilon$ for constants β_1 and β_2 and a random disturbance term ϵ, to samples (x_i, y_i) for $i = 1, \ldots, n$. The analysis above suggests solving the equations (3) with the expectations involving the disturbance term taken to be zero. Show that this gives the same parameters β_1 and β_2 which minimize the sum of the squares of the errors
$\mathrm{SSE} = \sum_{i=1}^{n} (y_i - (\beta_1 + \beta_2\, x_i))^2$
and illustrate that this gives the same as the Fit command, at least for a

particular numerical list of pairs.

You may use *Mathematica* to do some of the calculation for you if you take a particular n, say $n = 5$, and symbolic x_i and y_i such as

In[1]:=
```
ClearAll
```
Out[1]=
```
ClearAll
```

In[2]:=
```
data = {{x1, y1}, {x2, y2}, {x3, y3}, {x4, y4}, {x5, y5}};
```

In[3]:=
```
xvector = data[[All, 1]]
```
Out[3]=
```
{x1, x2, x3, x4, x5}
```

In[4]:=
```
yvector = data[[All, 2]]
```
Out[4]=
```
{y1, y2, y3, y4, y5}
```

We define the sum of squares of errors as follows.

In[5]:=
```
se[beta1_, beta2_] = (yvector - (beta1 + beta2 * xvector)) ^2
```
Out[5]=
```
{(-beta1 - beta2 x1 + y1)^2, (-beta1 - beta2 x2 + y2)^2,
 (-beta1 - beta2 x3 + y3)^2, (-beta1 - beta2 x4 + y4)^2, (-beta1 - beta2 x5 + y5)^2}
```

In[6]:=
```
sse = Total[se[beta1, beta2]]
```
Out[6]=
```
(-beta1 - beta2 x1 + y1)^2 + (-beta1 - beta2 x2 + y2)^2 +
 (-beta1 - beta2 x3 + y3)^2 + (-beta1 - beta2 x4 + y4)^2 + (-beta1 - beta2 x5 + y5)^2
```

To find parameters β_1, β_2 that minimize the error term, we can take the partial derivatives of the error term w.r.t each one of them, and set the derivatives to zero.

In[7]:=
```
minimumErrorConditions = {D[sse, beta1] == 0, D[sse, beta2] == 0};
MatrixForm[Simplify[minimumErrorConditions]]
```
Out[8]//MatrixForm=

$$\begin{pmatrix} 5\,\text{beta1} + \text{beta2}\,(\text{x1} + \text{x2} + \text{x3} + \text{x4} + \text{x5}) == \text{y1} + \text{y2} + \text{y3} + \text{y4} + \text{y5} \\ \text{beta1}\,(\text{x1} + \text{x2} + \text{x3} + \text{x4} + \text{x5}) + \text{beta2}\left(\text{x1}^2 + \text{x2}^2 + \text{x3}^2 + \text{x4}^2 + \text{x5}^2\right) == \text{x1 y1} + \text{x2 y2} + \text{x3 y3} + \text{x4 y4} + \text{x5 y5} \end{pmatrix}$$

These equations must be the same for a fit of the linear model $y = \beta_1 + \beta_2\, x + \epsilon$. We do the analysis as shown above. Since the first input variable is set to 1, we have the first expectation equation as shown below. We define ϵ to be a normal random variable with a mean of 0.

In[9]:=
```
epsilonDistribution = NormalDistribution[0, 1];
```

In[10]:=
```
eqn1 = Mean[1 * yvector] ==
  beta1 * Mean[{1, 1, 1, 1, 1}] + beta2 * Mean[1 * xvector] + Mean[epsilonDistribution]
```
Out[10]=

$$\frac{1}{5}\,(\text{y1} + \text{y2} + \text{y3} + \text{y4} + \text{y5}) == \text{beta1} + \frac{1}{5}\,\text{beta2}\,(\text{x1} + \text{x2} + \text{x3} + \text{x4} + \text{x5})$$

In[11]:=
```
1/5 (y1 + y2 + y3 + y4 + y5) == beta1 + 1/5 beta2 (x1 + x2 + x3 + x4 + x5)
```

Out[11]=

$$\frac{1}{5}(y1+y2+y3+y4+y5) == beta1 + \frac{1}{5} beta2\,(x1+x2+x3+x4+x5)$$

The second equation involves the second input variable, x. We assume independence of x and ϵ so that the expectation $E[x\,\epsilon] = E[x]\,E[\epsilon]$

In[12]:=
```
eqn2 = Mean[xvector * yvector] == beta1 * Mean[xvector] +
   beta2 * Mean[xvector * xvector] + Mean[xvector] * Mean[epsilonDistribution]
```

Out[12]=

$$\frac{1}{5}(x1\,y1+x2\,y2+x3\,y3+x4\,y4+x5\,y5) ==$$
$$\frac{1}{5} beta1\,(x1+x2+x3+x4+x5) + \frac{1}{5} beta2\,(x1^2+x2^2+x3^2+x4^2+x5^2)$$

The equations are written together below in a simplified form.

In[13]:=
```
MatrixForm[Simplify[{eqn1, eqn2}]]
```

Out[13]//MatrixForm=

$$\begin{pmatrix} 5\,beta1 + beta2\,(x1+x2+x3+x4+x5) == y1+y2+y3+y4+y5 \\ beta1\,(x1+x2+x3+x4+x5) + beta2\,(x1^2+x2^2+x3^2+x4^2+x5^2) == x1\,y1+x2\,y2+x3\,y3+x4\,y4+x5\,y5 \end{pmatrix}$$

We can clearly see that these are the same equations as obtained above for minimizing the sum of squares of errors.

Unfortunately, the Fit command doesn't work with symbolic values, so for the last step we will generate our own table of random points, starting from a linear relationship between x and y with random noise added, and then check that the formulas for β_1 and β_2 agree with the result of Fit.

We start off with a linear function $y = 8 + 5\,x$ and add to it a random noise ϵ which is a normal random variable with mean 0 and variance 1.

In[14]:=
```
yfunc[x_] = 8 + 5 * x
```

Out[14]=

```
8 + 5 x
```

We use the above function to generate 5 pairs of values (x_i, y_i) and then add random noise to it.

In[15]:=
```
xi = Table[Random[UniformDistribution[{100, 200}]], 5]
```

Out[15]=

```
{119.402, 186.968, 115.272, 169.904, 122.543}
```

In[16]:=
```
yi = yfunc[xi] + Table[Random[epsilonDistribution], 5]
```

Out[16]=

```
{606.321, 941.534, 584.466, 857.767, 619.865}
```

Let us now computer β_1 and β_2 using the equations obtained above.

In[17]:=
```
eqn1 = eqn1 /. {x1 → xi⟦1⟧, x2 → xi⟦2⟧, x3 → xi⟦3⟧, x4 → xi⟦4⟧,
   x5 → xi⟦5⟧, y1 → yi⟦1⟧, y2 → yi⟦2⟧, y3 → yi⟦3⟧, y4 → yi⟦4⟧, y5 → yi⟦5⟧}
```
Out[17]=
```
721.991 == beta1 + 142.818 beta2
```

In[18]:=
```
eqn2 = eqn2 /. {x1 → xi⟦1⟧, x2 → xi⟦2⟧, x3 → xi⟦3⟧, x4 → xi⟦4⟧,
   x5 → xi⟦5⟧, y1 → yi⟦1⟧, y2 → yi⟦2⟧, y3 → yi⟦3⟧, y4 → yi⟦4⟧, y5 → yi⟦5⟧}
```
Out[18]=
```
107 501. == 142.818 beta1 + 21 277.2 beta2
```

In[19]:=
```
betaSolution = Solve[{eqn1, eqn2}, {beta1, beta2}]⟦1⟧
```
Out[19]=
```
{beta1 → 10.0836, beta2 → 4.98472}
```

Let us now see what the Fit command gives us.

In[20]:=
```
sampleData = Table[{xi⟦i⟧, yi⟦i⟧}, {i, 1, Length[xi]}]
```
Out[20]=
```
{{119.402, 606.321}, {186.968, 941.534},
 {115.272, 584.466}, {169.904, 857.767}, {122.543, 619.865}}
```

In[21]:=
```
Fit[sampleData, {1, x}, {x}]
```
Out[21]=
```
10.0836 + 4.98472 x
```

The coefficients of the function obtained by the Fit function are the same as the values of β_1 and β_2 obtained from the equations above.

2. Use instrumental variables to recover the correct demand function from the following (generated) data. In each 4-tuple of variables there are two exogenous variables *x* and *y*, the price *p* and the quantity *q*. We wish to estimate $q = \beta_1 + \beta_2\, p + \epsilon$, but *p* and ϵ may be correlated whereas *x* and *y* are assumed not to be correlated with ϵ. If *x* or *y* is correlated with *p* then we can use it as an instrumental variable for *p*. Which variable is a better instrument and what is your fit to the demand?

In[22]:=
```
data = {{23.15691, 15.45811, 44.79242, 308.09241}, {0.5803, 17.38291, 45.42258, 301.61799},
   {-3.25427, 4.27329, 44.13642, 323.21183}, {11.36881, 11.79195, 40.92019, 301.45044},
   {11.84428, -16.22184, 39.15565, 352.2795}, {-1.04015, -3.01254, 47.28803, 347.68113},
   {18.53038, 9.02666, 44.76236, 319.93984}, {2.90757, -7.31793, 43.02779, 344.30074},
   {-19.33563, -8.16812, 41.52154, 337.03374}, {3.1288, 15.29222, 52.31075, 326.97358},
   {20.29793, 2.27929, 39.71285, 318.63956}, {11.50157, -3.30247, 47.07619, 350.13382},
   {6.34872, -3.79671, 34.25256, 311.62085}, {-8.44546, -2.13878, 29.71433, 291.73147},
   {-0.07284, -21.0742, 43.89337, 373.81394}, {-0.68535, -2.53145, 36.04218, 313.05236},
   {2.3228, 27.8266, 52.64595, 302.74921}, {-1.17697, -2.1121, 43.28597, 333.84673},
```

```
{-2.51654, -2.86954, 40.52926, 326.82354}, {-9.05483, 15.53062, 65.13563, 362.5347},
{8.88703, 23.89681, 54.86989, 318.59346}, {7.79212, -9.37178, 41.61776, 345.15526},
{1.37923, 3.0865, 49.06994, 341.31267}, {9.85277, 10.74809, 52.71193, 338.61018},
{5.88387, 18.82951, 57.70474, 336.63196}, {4.15615, -10.00268, 43.33101, 350.82963},
{3.70906, -3.80757, 39.73952, 327.57552}, {17.9821, -12.22712, 41.87243, 353.66793},
{-9.7282, 8.30736, 37.56729, 294.14151}, {0.96172, 6.33469, 43.352, 317.57898},
{4.82072, -2.60321, 39.53513, 324.77595}, {-1.7411, 11.33733, 39.29831, 294.87205},
{-4.15087, -9.10139, 45.65245, 354.32996}, {-13.28273, 8.00082, 49.62256, 330.2095},
{2.07544, -11.61139, 37.451, 335.99086}, {14.46891, -4.62958, 39.17853, 329.68854},
{5.71901, -2.71556, 49.69812, 355.66929}, {4.30963, 6.54786, 53.13563, 347.17307},
{-6.6106, -15.34681, 45.51964, 365.93041}, {-6.31624, -0.85354, 47.04105, 341.56697},
{1.19459, -10.09484, 38.9607, 337.31069}, {-8.38443, -4.43509, 40.30613, 328.11169},
{0.40837, 8.409, 46.62953, 323.15228}, {-5.04852, -16.68449, 32.64133, 330.28327},
{-8.76941, 5.1251, 47.80941, 331.42415}, {-7.83504, -1.22584, 42.63536, 328.79074},
{7.9134, -7.10525, 43.84177, 347.31849}, {5.22274, -9.93568, 26.70029, 301.0168},
{6.99001, -14.482, 44.2646, 363.1558}, {10.33571, -10.47242, 39.41239, 341.24915},
{-6.31857, -12.89055, 26.91837, 305.27248}, {-3.15743, 9.03983, 47.0728, 322.50724},
{-6.71904, -8.47513, 36.60083, 325.40894}, {-26.34009, 8.66462, 50.60614, 329.22117},
{-6.42644, -23.29275, 35.31603, 351.2483}, {8.87515, -2.95819, 36.38052, 316.83296},
{-1.94933, 8.7013, 55.2338, 347.90893}, {-9.39371, 5.15138, 38.06344, 302.00882},
{-1.87301, 5.11028, 56.41534, 358.65087}, {-15.24217, -14.82347, 38.62762, 342.48139},
{10.91847, -8.72139, 41.83635, 345.13555}, {3.59026, 6.41814, 36.93534, 298.6878},
{-1.07962, -13.49562, 36.39688, 335.96595}, {0.81246, 7.91883, 44.76299, 318.6138},
{14.30169, -4.43078, 41.05623, 334.89058}, {-13.21432, -10.4632, 41.45949, 342.66201},
{7.88568, -6.57022, 38.64513, 330.65296}, {-6.59943, -2.35644, 42.75097, 331.64591},
{-3.27187, -3.79816, 47.22235, 348.60901}, {0.2067, -17.30317, 33.33132, 334.64165},
{-1.77237, 5.45733, 42.6951, 316.81617}, {-24.23577, 0.15564, 45.68577, 331.89888},
{-0.53509, -10.47425, 38.84444, 337.37479}, {-1.05229, 7.09557, 51.97876, 341.53469},
{-18.64549, 11.76555, 58.36912, 347.84717}, {-12.63684, 18.80361, 44.2281, 292.54969},
{2.72269, -0.33805, 42.32995, 328.21048}, {-8.9964, -9.42691, 31.45672, 311.4247},
{15.22018, -6.20571, 51.2226, 369.12325}, {13.88659, 22.88012, 48.09756, 301.30976},
{9.69485, 8.75663, 38.76866, 300.73169}, {-21.24667, 3.16382, 48.20028, 334.02389},
{8.99225, -1.48824, 48.79682, 351.16539}, {-8.72465, 13.2605, 44.37723, 304.86574},
{13.12824, -2.1658, 42.23085, 333.6498}, {5.79424, 0.041, 45.30701, 336.9979},
{-4.59088, -3.78289, 39.32913, 324.635}, {5.24257, -8.8453, 36.75014, 328.98953},
{-17.77709, 5.46392, 44.58934, 319.28477}, {-7.03455, 10.61411, 41.12585, 300.74242},
{-2.86172, 5.01476, 40.6782, 311.43275}, {-3.20394, 14.56681, 48.49714, 315.71701},
{-19.69469, -7.68417, 40.50261, 332.93723}, {0.55379, -2.04384, 43.61959, 335.05722},
{-6.72612, -4.08867, 46.95752, 347.70467}, {-19.99442, 3.88219, 35.94153, 296.06133},
{-6.58365, 9.84833, 44.70661, 313.10643}, {-1.70836, 6.08152, 59.47787, 365.92889},
{-30.79313, -1.47208, 41.55051, 321.43707}, {10.81975, 6.98119, 39.78538, 307.55772},
{-11.58116, -6.37349, 42.16796, 336.93464}, {-18.94671, 4.91459, 36.8311, 296.87478},
{-6.22016, -7.92227, 40.33514, 335.60593}, {-16.51646, 7.18554, 49.26256, 330.11332},
{-15.34158, 5.93346, 44.76114, 319.34817}, {0.95649, 15.78826, 41.66904, 293.62189},
{-12.82422, 1.94855, 44.89526, 328.22382}, {-5.10977, 2.83221, 37.78308, 306.66288},
{-5.72964, -2.83507, 38.86676, 321.12448}, {-9.61632, -2.2351, 43.68391, 333.59867},
{5.6565, -3.24011, 36.56272, 317.29967}, {17.71905, -8.0309, 34.91231, 324.34254},
```

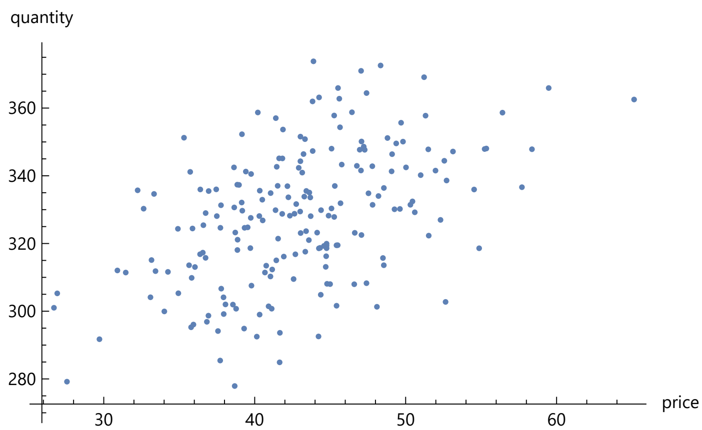
```
{3.85556, 7.24168, 38.55658, 301.95749}, {-13.54511, 7.03883, 45.4198, 319.47272},
{-3.19195, 6.82335, 44.33468, 318.71896}, {8.61982, -2.69195, 38.71298, 323.2468},
{-19.6311, -10.32064, 43.23534, 346.42109}, {5.92018, 20.618, 41.65164, 284.90296},
{-1.43685, 0.13037, 47.79235, 342.82895}, {16.82873, -0.73547, 33.09448, 304.12014},
{0.61231, 9.81315, 50.31555, 331.44282}, {-1.95251, 10.79442, 40.32673, 299.00084},
{-5.42791, 5.50659, 35.78616, 295.2597}, {8.48757, -6.96146, 33.16364, 315.11137},
{-11.288, -11.75214, 39.76195, 340.53252}, {6.61672, 8.00178, 37.94822, 299.16445},
{0.66891, -8.33219, 35.87969, 324.43724}, {-2.36819, 0.22117, 49.10281, 346.39244},
{-6.77262, -7.4812, 42.91689, 342.35857}, {-8.98394, 12.92974, 54.53639, 335.9529},
{0.87418, -10.9193, 45.26484, 357.80795}, {11.24136, 17.16389, 48.55432, 313.58345},
{-5.39501, -3.30329, 36.74563, 315.76449}, {-2.71218, -17.2594, 35.72729, 341.15824},
{-5.01547, 13.35244, 37.71359, 285.4328}, {5.59823, -18.47988, 40.20776, 358.70269},
{-0.62364, 1.46309, 43.7093, 328.07698}, {18.23147, -3.95333, 33.42837, 311.83804},
{-8.25238, -2.72806, 51.31934, 357.76367}, {-10.73687, -2.71643, 41.82318, 328.75504},
{-3.51305, -0.59253, 34.94148, 305.30689}, {-6.14877, 5.78269, 50.99628, 340.19371},
{-15.86046, -4.89875, 35.65213, 313.58178}, {-6.29124, 5.99711, 52.56245, 344.43488},
{-4.41086, -13.42017, 45.60488, 362.77282}, {8.85288, 3.33704, 45.0873, 330.3584},
{-7.3938, -11.84555, 47.40971, 364.44147}, {5.1252, 4.3071, 50.02424, 342.48355},
{-5.83038, -12.90422, 36.95031, 335.4933}, {-1.08939, 1.56282, 44.39642, 329.84577},
{-14.96397, 3.1379, 41.43104, 315.02452}, {-1.46364, 4.73513, 43.58883, 321.00352},
{-12.34532, -3.86841, 43.41972, 335.52692}, {1.26936, 3.47045, 51.50053, 347.81455},
{-5.88928, 7.90892, 45.49964, 319.50324}, {-15.09846, 14.39934, 46.59562, 307.96848},
{-4.79805, -1.79643, 46.76788, 342.93688}, {5.84098, 5.43398, 37.9342, 304.10284},
{5.24915, 0.21556, 49.8248, 350.09311}, {11.48181, 14.63862, 44.99625, 308.00788},
{-3.15234, 18.74464, 38.67486, 277.90483}, {1.16004, -8.88648, 37.76662, 331.30483},
{-8.52187, 0.19755, 34.01324, 299.94024}, {-11.29737, -12.3696, 43.03077, 351.57204},
{-4.83419, 8.97662, 50.45279, 332.43829}, {1.12415, 3.39612, 43.40441, 323.64582},
{-1.85876, -16.60517, 41.3984, 357.03379}, {14.9134, -13.43471, 47.04861, 370.99792},
{5.75011, 9.55555, 44.73494, 316.24374}, {0.66415, 9.05705, 55.34591, 348.05645},
{-2.79575, -3.28889, 45.76582, 343.31609}, {-4.93487, -1.25, 38.85416, 318.07548},
{-14.45952, -1.64716, 43.01327, 329.44222}, {-31.62974, 3.27144, 41.04388, 310.26283},
{4.47884, 4.86706, 40.76327, 313.45147}, {-28.51871, 13.28047, 51.5302, 322.32593},
{-24.05312, 4.68824, 44.25274, 318.5711}, {-0.1003, 16.95893, 47.40552, 308.27863},
{1.82097, 1.93097, 27.56235, 279.18931}, {-1.09971, -9.82962, 46.44257, 358.76699},
{10.7852, -1.77434, 41.38033, 329.8467}, {12.83392, -6.06496, 39.13783, 332.1102},
{-14.79053, -2.19739, 49.37534, 349.56269}, {0.67156, -5.6817, 43.1489, 340.9444},
{-14.24991, 12.53723, 40.13908, 292.49279}, {1.43253, 9.26019, 42.57723, 309.49782},
{-12.14451, 4.36392, 41.15959, 312.32202}, {-2.6449, -5.9928, 37.72798, 324.64057},
{10.67132, -0.12462, 35.82318, 309.85305}, {19.33415, -7.73218, 30.90333, 312.04116},
{9.46737, -12.83408, 48.34068, 372.58367}, {5.62894, 3.58461, 43.04381, 323.08801},
{1.29711, -19.36142, 32.24422, 335.7149}, {-1.5725, 4.47654, 48.55823, 336.40712},
{5.20353, -14.71733, 43.83292, 361.97414}, {-6.45251, 4.17637, 41.92772, 316.13994},
{-12.43022, 2.64785, 47.53768, 334.83129}, {1.69015, 4.14798, 45.26817, 327.8466},
{12.0804, -5.16339, 45.09146, 348.01723}, {3.64827, -7.446, 37.48965, 328.09062}};
```

In[23]:=
```
xvar = data[[All, 1]];
yvar = data[[All, 2]];
pvar = data[[All, 3]];
qvar = data[[All, 4]];
```

We can see the price and quantity plot below.

In[27]:=
```
ListPlot[data[[All, {3, 4}]], AxesLabel → {"price", "quantity"}]
```

Out[27]=



To determine the coefficients β_1 and β_2 of the demand function, we write down the expectation equations, using the instrumental variables. Since they are independent of the disturbance ϵ, the expectation terms involving the product of instrumental variables with the disturbance can be set to 0.

The first equation is

In[28]:=
```
eqn1 = Mean[qvar] == beta1 + beta2 * Mean[pvar]
```

Out[28]=
```
328.052 == beta1 + 42.9484 beta2
```

The second equation, using the instrumental variable x is

In[29]:=
```
eqn2x = Mean[xvar * qvar] == beta1 * Mean[xvar] + beta2 * Mean[xvar * pvar]
```

Out[29]=
```
-497.649 == -1.55324 beta1 - 73.3502 beta2
```

The second equation, using the instrumental variable y is

In[30]:=
```
eqn2y = Mean[yvar * qvar] == beta1 * Mean[yvar] + beta2 * Mean[yvar * pvar]
```

Out[30]=
```
-27.0019 == 0.241112 beta1 + 36.712 beta2
```

Solving equation 1 with equation 2 corresponding to the instrumental variable x, we have

In[31]:=
```
betax = Solve[{eqn1, eqn2x}, {beta1, beta2}][[1]]
```

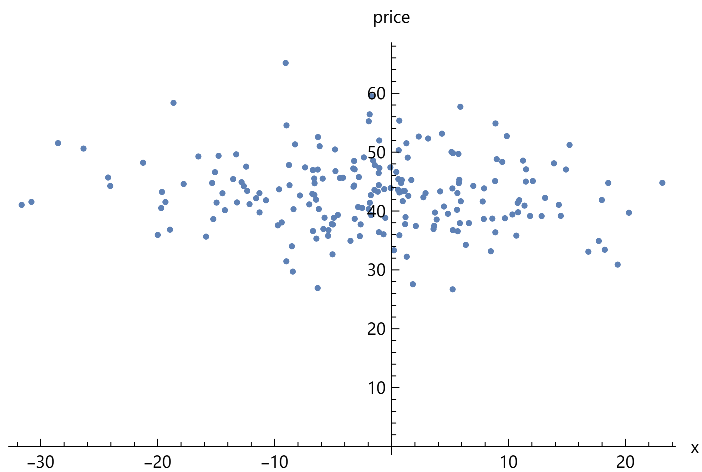
Out[31]=
```
{beta1 → 404.972, beta2 → -1.79098}
```

Solving equation 1 with equation 2 corresponding to the instrumental variable y, we have

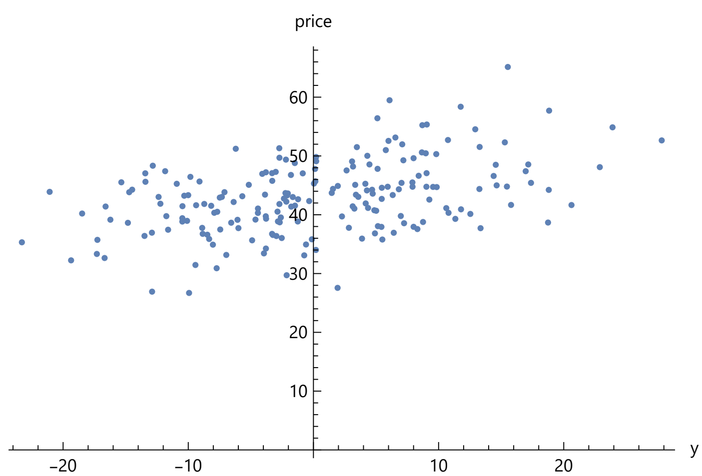
In[32]:=
```
betay = Solve[{eqn1, eqn2y}, {beta1, beta2}][[1]]
```
Out[32]=
```
{beta1 → 500.942, beta2 → -4.02553}
```

To determine the better instrumental variable, we can plot a graph of the instrumental variables versus the price and observe, which of the variables is better correlated to the price.

In[33]:=
```
ListPlot[data[[All, {1, 3}]], AxesLabel → {"x", "price"}]
```
Out[33]=



In[34]:=
```
ListPlot[data[[All, {2, 3}]], AxesLabel → {"y", "price"}]
```
Out[34]=



From the plots, we can see that the instrumental variable y seems to be better correlated with the price, p and the disturbances are contained within a smaller band, as compared to the $x-p$ plot. **Hence, y would be the better choice for an instrumental variable.**

Thus, the estimate for the demand function is

In[35]:=
```
demandfunctionestimate = beta1 + beta2 * p + epsilon /. betay
```
Out[35]=
```
500.942 + epsilon - 4.02553 p
```