Hanford Part #1: Finding the Least Square Regression

The Problem

Given problem text: In an article taken from the Journal of Environmental Health, May-June 1965, Volume 27, Number 6, pages 883-897, author Robert Fadely explains that the Atomic Energy Plant in Hanford, Washington has been a plutonium production facility since the Second World War. Some of the waste have been stored underground in the same area. Radioactive waste has been seeping into the Columbia River, and eight Oregon counties and the city of Portland have been exposed to radioactive contamination. The table below lists the number of cancer deaths per 100,000 residents for Portland and these counties. The table also includes an index of exposure that measures the proximity of the residents to the contamination. The index is based on the assumption that city or county exposure is directly proportional to river frontage and inversely proportional both to the distance from Hanford, WA site and to the square of the county's or city's average distance from the river.

Create a Mathematica notebook file that displays a Least Squares regression line and residuals.
- A table displayed
- A graph of the data set
- A graph of the data set with least squares line (not the Fit command)
- A graph of your least squares line residuals
- A value for the sum of residuals for the least-squares line

The following graph presents the data given (the code for it is in the next section):

In[1]:=
```
Grid[{{"Location", "Umatilla", "Morrow", "Gilliam",
    "Sherman", "Wasco", "Hood River", "Portland", "Columbia", "Clatsop"},
  {"Index", 2.5`, 2.6`, 3.4`, 1.3`, 1.6`, 3.8`, 11.6`, 6.4`, 8.3`},
  {"Deaths", 147, 130, 130, 114, 138, 162, 208, 178, 210}},
 Frame → All, BaseStyle → {FontSize → 12}]
```

Out[1]=

Location	Umatilla	Morrow	Gilliam	Sherman	Wasco	Hood River	Portland	Columbia	Clatsop
Index	2.5	2.6	3.4	1.3	1.6	3.8	11.6	6.4	8.3
Deaths	147	130	130	114	138	162	208	178	210

Creating Lists of Data

The lists in order are on of the city names, the 'index' or distance that is given for each city, and finally the deaths for each index.

In[2]:=
```
listLabels = List["Umatilla", "Morrow", "Gilliam",
  "Sherman", "Wasco", "Hood River", "Portland", "Columbia", "Clatsop"]
```

Out[2]= {Umatilla, Morrow, Gilliam, Sherman, Wasco, Hood River, Portland, Columbia, Clatsop}

In[3]:=
```
listIndex = List[2.5, 2.6, 3.4, 1.3, 1.6, 3.8, 11.6, 6.4, 8.3]
```

Out[3]= {2.5, 2.6, 3.4, 1.3, 1.6, 3.8, 11.6, 6.4, 8.3}

In[4]:=
```
listDeaths = List[147, 130, 130, 114, 138, 162, 208, 178, 210]
```

Out[4]= {147, 130, 130, 114, 138, 162, 208, 178, 210}

Then I combined all of the lists in order to create the table in the previous section, then I created a transposed data table of the values. I then created a separate list of just the numerical combined data in order to use it for computations.

In[5]:=
```
listDataWithLabels = Transpose[{listLabels, listIndex, listDeaths}];
```

In[6]:=
```
listLabeled = Insert[listDataWithLabels, {"Location", "Index", "Deaths"}, 1];
```

In[7]:=
```
listCombined = Transpose[{listIndex, listDeaths}]
```

Out[7]= {{2.5, 147}, {2.6, 130}, {3.4, 130}, {1.3, 114},
{1.6, 138}, {3.8, 162}, {11.6, 208}, {6.4, 178}, {8.3, 210}}

In[8]:=
```
Grid[{{"Location", "Umatilla", "Morrow", "Gilliam",
    "Sherman", "Wasco", "Hood River", "Portland", "Columbia", "Clatsop"},
  {"Index", 2.5, 2.6, 3.4, 1.3, 1.6, 3.8, 11.6, 6.4, 8.3},
  {"Deaths", 147, 130, 130, 114, 138, 162, 208, 178, 210}},
 Frame → All, BaseStyle → {FontSize → 12}]
```

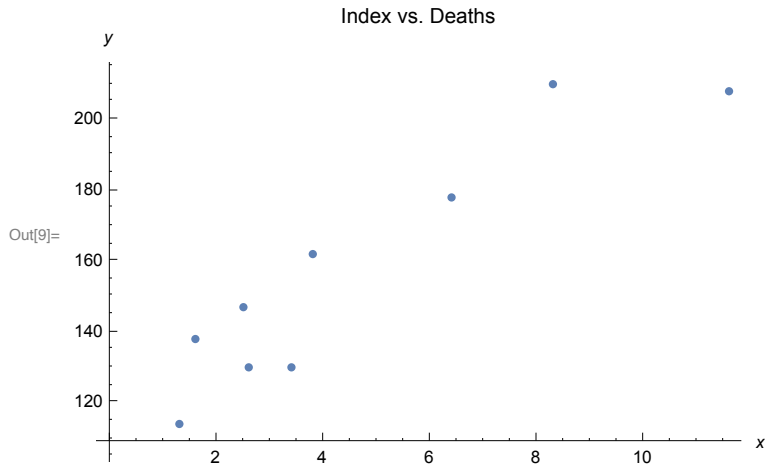
Out[8]=

Location	Umatilla	Morrow	Gilliam	Sherman	Wasco	Hood River	Portland	Columbia	Clatsop
Index	2.5	2.6	3.4	1.3	1.6	3.8	11.6	6.4	8.3
Deaths	147	130	130	114	138	162	208	178	210

Graphing the Data Set

In[9]:=
```
dataPlot = ListPlot[listCombined,
  AxesLabel → {HoldForm[x], HoldForm[y]}, PlotLabel → "Index vs. Deaths"]
```

Out[9]=



Calculating the Least Squares Line

In this section I first defined each of the sums that were used in the derived equations for m and b, and using this I was able to calculated the values of the slope and y-intercept of m & b of the least squares line.

In[10]:=
$$\text{SymA} = \sum_{i=1}^{9} \text{listDeaths}[[i]]^2;$$

In[11]:=
$$\text{SymB} = \sum_{i=1}^{9} \text{listIndex}[[i]]^2;$$

In[12]:=
$$\text{SymC} = \sum_{i=1}^{9} \text{listIndex}[[i]];$$

In[13]:=
$$\text{SymD} = \sum_{i=1}^{9} \text{listIndex}[[i]] * \text{listDeaths}[[i]];$$

In[14]:=
$$\text{SymE} = \sum_{i=1}^{9} \text{listDeaths}[[i]];$$

In[15]:=
$$m = \frac{(\text{SymD} * 9) - (\text{SymE} * \text{SymC})}{(\text{SymB} * 9) - \left(\text{SymC}^2\right)}$$

Out[15]=
$$9.27386$$

In[16]:=
$$b = \frac{(SymD * SymC) - (SymE * SymB)}{SymC^2 - (9 * SymB)}$$

Out[16]= 114.682

Or....

In[17]:=
$$b = \frac{(2 * SymE) - (2 * m * SymC)}{(2 * 9)}$$

Out[17]= 114.682

(Note, here I found the value for b using two methods, one being with the m used in it, and one without being m dependent).

This allowed me to generate and define the following least squares equation for a linear regression fit, and then check it using the fit function:

In[18]:=
```
fit[x_] = 9.27386 * x + 114.682;
```

In[19]:=
```
Fit[listCombined, {1, x}, x]
```

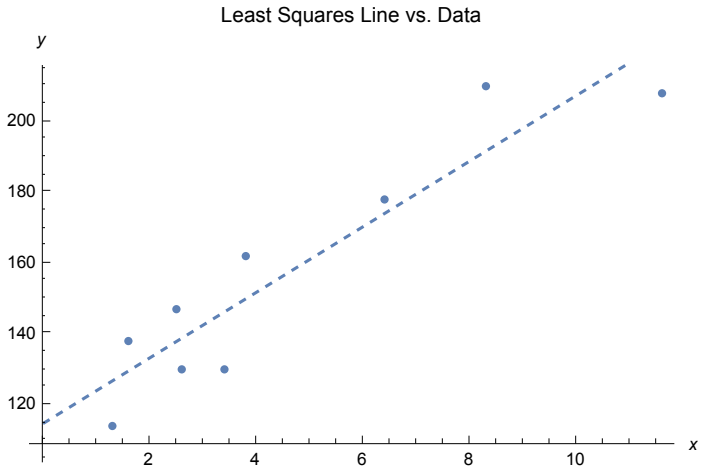
Out[19]= 114.682 + 9.27386 x

Plotting the Least Squares Line With the Data

In[20]:=
```
linePlot = Plot[9.27386 x + 114.682, {x, 0, 12},
    AxesLabel → {HoldForm[x], HoldForm[y]}, PlotStyle → Dashed];
```

In[21]:=
```
Show[ListPlot[listCombined], linePlot,
 AxesLabel → {HoldForm[x], HoldForm[y]}, PlotLabel → "Least Squares Line vs. Data"]
```

Out[21]=



Calculating Residuals

In order to calculate the residuals I first made a list titled 'listPredicted' that consisted of all of the predicted death amounts and took the value at the first index of this list in order to make it a series of

ordered pairs, and created a list titled ‘listRedisuals” that took the val pue at each index of the actual given death data and subtracted from it the predicted deaths at that index. I could then plot this graph.

In[22]:=
```
listPredicted = Table[fit[x], {x, {listIndex}}]
```

Out[22]=
```
{{137.867, 138.794, 146.213, 126.738, 129.52, 149.923, 222.259, 174.035, 191.655}}
```

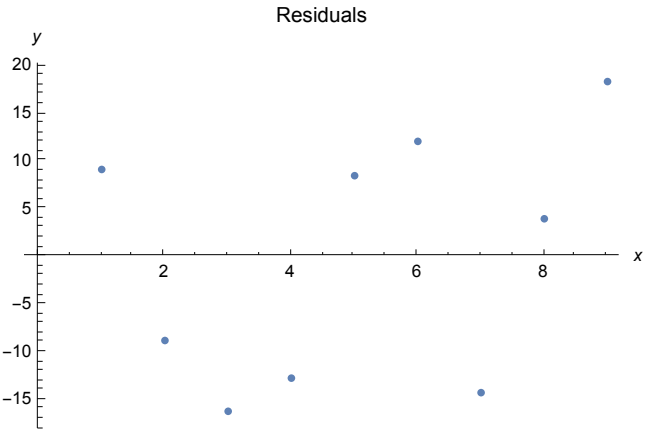
In[23]:=
```
listPredicted = listPredicted[[1]];
```

In[24]:=
```
listResiduals = listDeaths - listPredicted
```

Out[24]=
```
{9.13335, -8.79404, -16.2131, -12.738, 8.47982, 12.0773, -14.2588, 3.9653, 18.345}
```

In[25]:=
```
ListPlot[listResiduals, AxesLabel → {x, y}, PlotLabel → "Residuals"]
```

Out[25]=



Finding the Sum of the Residuals

In this section the only goal was to find the sum of the residuals, and for this I simply added all of the values of my residuals list using the Total function.

In[26]:=
```
residualSum = Total[listResiduals]
```

Out[26]=
```
-0.00319
```

The residuals value came out to -0.00319, which is extremely close to zero even though the residuals were all fairly high. Thus, the value is not entirely representative of the actual accuracy of the line of best fit. Thus, I decided to redo the residual sum through finding the absolute value of each term in the residuals list, and then sum this.

In[27]:=
```
listResidualsAbs = Abs[listResiduals]
```

Out[27]=
```
{9.13335, 8.79404, 16.2131, 12.738, 8.47982, 12.0773, 14.2588, 3.9653, 18.345}
```

In[28]:=
```
residualAbsSum = Total[listResidualsAbs]
```

Out[28]=
```
104.005
```

The new residual sum is 104.005, which is more indicative that form the plot of the line of best fit compared to the actual values, it is fairly far off on most of them.