Least squares in linear regression

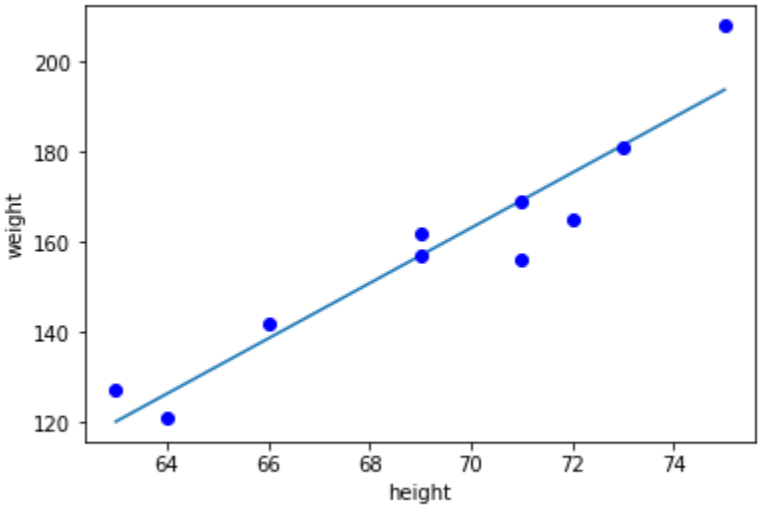
Linear regression is a traditional entry point into the extremely useful, exciting and trending area of computer science called machine learning. Which for our basic purpose means that a service or program can "learn" from data without a programmer explicitly intervening to change things. Some of people favorite computer features such as targeted advertising and predictive text use machine learning algorithms. Or maybe for an example people actually like a service like Spotify. At its core linear regression algorithms are a predictive analysis of data. In its basic form it determines the relationship between a dependent variable and an independent variable (more than one are possible). Some of its best use cases involve time ordered data, or just continuous variable data in general that you want a numerical answer for. In its most basic form, we have the equation $\hat{y}_i = b_0 + b_1 x_i$ and if you think it looks like an equation for a line you would be correct. What you are looking at is the equation for the best fit line where b_0 is some constant, b_1 is the regression coefficient, x_i is the independent variable, and $\hat{y}_i$ is the predicted value. (Programs, 2018)

To actually use linear regression correctly there are a few key assumptions that need to be met or your regression model is going to have flaws some of which are: Linear relationship between dependent and independent vars, Multivariate normality (normal distribution). No multicollinearity which happens when independent variables are correlated with each other. No auto-correlation which is where y(x) is not independent from other values such as y(x-1), y(x+1), etc. Homoscedasticity which is where residuals are equals across the regression line, residuals being the difference between the predicted value and observed value. (Solutions, 2019) Looking at the heights and weights of 10 people col 2 of A are the heights in inches and B is the accompanying weight

$$A=\begin{bmatrix}1 & 63\\ 1 & 64\\ 1 & 66\\ 1 & 69\\ 1 & 69\\ 1 & 71\\ 1 & 71\\ 1 & 72\\ 1 & 73\\ 1 & 75\end{bmatrix} \quad x=\begin{matrix}b_0\\ b_1\end{matrix} \quad \text{and} \quad B=\begin{bmatrix}127\\ 121\\ 142\\ 157\\ 162\\ 156\\ 169\\ 165\\ 181\\ 208\end{bmatrix}$$

$$(A^T A)^{-1}.\ B\ =$$

$$\begin{bmatrix}\frac{48163}{1381} & -\frac{693}{1381}\\ -\frac{693}{1381} & \frac{10}{1381}\end{bmatrix} \cdot \begin{bmatrix}1588\\ 110896\end{bmatrix} = \begin{bmatrix}-\frac{368084}{1381}\\ \frac{8476}{1381}\end{bmatrix}$$



Plot of the height weight data with the least squares solution line

With a best fit line of $\hat{y}_{i(weight)} = -266.534 + 6.1375 x_{i(height)}$.

Regression is possible with 1 …. n variables and as you may suspect it looks something like equation

$$\hat{y}_i = b_0 + b_1 x_i + b_2 x_2 + \cdots + b_n x_n$$

The techniques used previously apply here.

One such example going back to the iris data from (Raschka, 2015) we can use a regression to estimate what a irises sepal length is given the petal length and width. Why do this? Because taking measurements is hard. So, like before we form up the matrix A (only a small portion of the matrix)=

```
[1. , 1.4, 0.2],
[1. , 1.7, 0.4],
[1. , 1.4, 0.3],
[1. , 1.5, 0.2],
[1. , 1.4, 0.2],
[1. , 1.5, 0.1],
[1. , 1.5, 0.2],
[1. , 1.6, 0.2],
[1. , 1.4, 0.1],
[1. , 1.1, 0.1],
[1. , 1.2, 0.2],
[1. , 1.5, 0.4],
[1. , 1.3, 0.4],
[1. , 1.4, 0.3],
[1. , 1.7, 0.3],
```

Applying the normal equation, we are left with $A^T A$ =

```
([[ 150.  ,  563.8 ,  179.8 ],
  [ 563.8 , 2583.  ,  868.97],
  [ 179.8 ,  868.97,  302.3 ]])
```

```
        ([[ 876.5 ],
          [3484.25],
and A^T b =  [1127.65]])
```

Solving with $(A^T A)^{-1}$. $A^T b$ gives the least squares solution

```
([[ 4.18950102],
  [ 0.54099383],
  [-0.31667117]])
```

And the equation for the sepal length from petal length and width is

$\hat{y}_{(sL)}$ =4.18950102+ 0.54099383$x_{1(P\ len)}$-0.31667117$x_{2(P\ wid)}$

Now to see how well we did let's look at some of the predicted values compared to the actual

```
([4.92039172, 4.980278  , 5.00110762, 4.980278  , 5.02974947,
  5.04016428, 4.79020434, 4.88133506, 4.87092025, 4.93080653])
```

Compare to the actual Y of data reserved for testing the first

```
1     4.9
3     4.6
5     5.4
7     5.0
9     4.9
11    4.8
13    4.3
15    5.7
17    5.1
19    5.1
Name: sepal_len
```

With an R^2 value of 0.5748598957765443 which on the surface means it Is an alright fit.

Conclusion

Though I originally wanted to build a classification algorithm for the irises using linear regression and least squares I fell short. Linear regression can serve respectably well as a classification algorithm if the man assumptions are met. For one in the regression I did do petal length and width are quite correlated with each other which you do not want as seen by the correlation matrix

	sepal_len	sepal_wid	petal_len	petal_wid
sepal_len	1.000000	-0.109369	0.871754	0.817954
sepal_wid	-0.109369	1.000000	-0.420516	-0.356544
petal_len	0.871754	-0.420516	1.000000	0.962757
petal_wid	0.817954	-0.356544	0.962757	1.000000

When I was trying to construct the classification algorithm using linear regression to try and determine what variety of iris it was given a petal length and width, I ran into the problem of linear regression not being good for non-continuous variables. So as a whole my multivariate linear regression classification algorithm attempt was a series of lessons in how not to do things.