Arissa Buchina
4468171
HW2

1 - Cost Function

The cost with (i) theta [0.5 2 1] is 0. The cost with (ii) theta [10 -1 -2] is 18.59. The values from the computeCost function and manually computing match.

```
Cost with (i) theta is J =  0.0
Cost with (ii) theta is J =  18.59375
```

(i) $\theta = [.5 \;\; 2 \;\; 1]$

$$X = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1.5 \\ 1 & 2 & 4 \\ 1 & 3 & 2 \end{bmatrix} \quad y = \begin{bmatrix} 1.5 \\ 4 \\ 8.5 \\ 8.5 \end{bmatrix} \quad \theta = \begin{bmatrix} .5 \\ 2 \\ 1 \end{bmatrix} \quad \cdot\, m = 4$$

$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 \rightarrow$ dot product $X \cdot \theta$

$$h_\theta(x) = \begin{bmatrix} (1)(.5) + (0)(2) + (1)(1) \\ (1)(.5) + (1)(2) + (1.5)(1) \\ (1)(.5) + (2)(2) + (4)(1) \\ (1)(.5) + (3)(2) + (2)(1) \end{bmatrix} = \begin{bmatrix} .5 + 0 + 1 \\ .5 + 2 + 1.5 \\ .5 + 4 + 4 \\ .5 + 6 + 2 \end{bmatrix} = \begin{bmatrix} 1.5 \\ 4 \\ 8.5 \\ 8.5 \end{bmatrix}$$

$$h_\theta(x) - y = \begin{bmatrix} 1.5 \\ 4 \\ 8.5 \\ 8.5 \end{bmatrix} - \begin{bmatrix} 1.5 \\ 4 \\ 8.5 \\ 8.5 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$(h_\theta(x) - y)^2 = \begin{bmatrix} 0^2 \\ 0^2 \\ 0^2 \\ 0^2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right)^2 \quad \frac{1}{2(4)} \cdot (0+0+0+0) = \frac{1}{8} \cdot (0) = 0$$

$$\boxed{J(\theta) = 0}$$

(ii) $\theta = [10 \;\; -1 \;\; -2]$

$$X = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1.5 \\ 1 & 2 & 4 \\ 1 & 3 & 2 \end{bmatrix} \quad y = \begin{bmatrix} 1.5 \\ 4 \\ 8.5 \\ 8.5 \end{bmatrix} \quad \theta = \begin{bmatrix} 10 \\ -1 \\ -2 \end{bmatrix} \quad \cdot\, m = 4$$

$$h_\theta(x) = X \cdot \theta = \begin{bmatrix} (1)(10) + (0)(-1) + (1)(-2) \\ (1)(10) + (1)(-1) + (1.5)(-2) \\ (1)(10) + (2)(-1) + (4)(-2) \\ (1)(10) + (3)(-1) + (2)(-2) \end{bmatrix} = \begin{bmatrix} 10 + 0 + (-2) \\ 10 + (-1) + (-3) \\ 10 + (-2) + (-8) \\ 10 + (-3) + (-4) \end{bmatrix} = \begin{bmatrix} 8 \\ 6 \\ 0 \\ 3 \end{bmatrix}$$

$$h_\theta(x) - y = \begin{bmatrix} 8 \\ 6 \\ 0 \\ 3 \end{bmatrix} - \begin{bmatrix} 1.5 \\ 4 \\ 8.5 \\ 8.5 \end{bmatrix} = \begin{bmatrix} 6.5 \\ 2 \\ -8.5 \\ -5.5 \end{bmatrix}$$

$$(h_\theta(x) - y)^2 = \begin{bmatrix} 6.5^2 \\ 2^2 \\ -8.5^2 \\ -5.5^2 \end{bmatrix} = \begin{bmatrix} 42.25 \\ 4 \\ 72.25 \\ 30.25 \end{bmatrix}$$

$$J(\theta) = \frac{1}{2m} \sum_{i=4}^{4} \left(h_\theta(x) - y \right)^2 = \frac{1}{2(4)} \cdot (42.25 + 4 + 72.25 + 30.25) = 18.59$$

$$\boxed{J(\theta) = 18.59}$$

2 - Gradient Descent

After 15 iterations the estimated theta and associated cost are

```
theta:  [[2.23525343]
 [0.38881471]
 [0.07501385]]
cost:  [8.5387097  8.39098188 8.24603568 8.10381853 7.96427887 7.8273661 7.69303056 7.56122357
 7.43189733 7.30500497 7.18050049 7.05833877
 6.93847554 6.82086736 6.70547162]
```

I ran the function a few times though just to see how much it varied and ensure that the cost was always decreasing, the second run I got this output

```
theta:  [[-0.95283223]
 [ 0.53136389]
 [-0.63878655]]
cost:  [39.45823185 38.71352086 37.98288569 37.26606029 36.56278364 35.87279963
 35.19585703 34.53170931 33.88011465 33.24083575 32.61363984 31.99829853
 31.39458774 30.80228764 30.22118255]
```

These both show as the iterations go on, the cost is going down slowly which is what is expected.

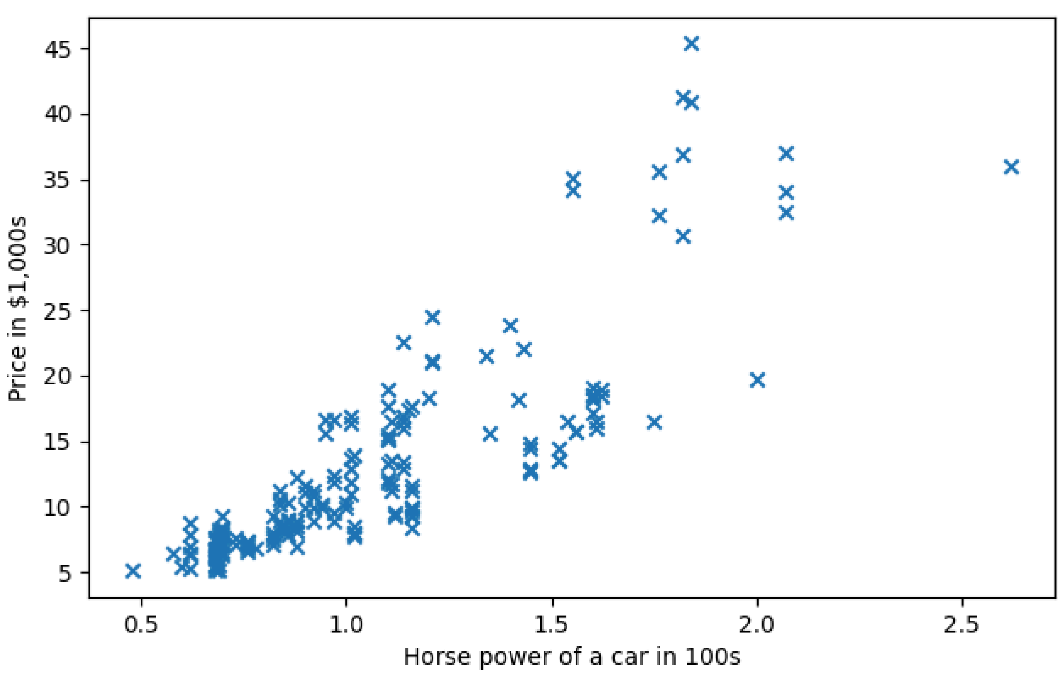
3 - Normal Equation

Looking at the theta output for the normal equation, pictured below, there is a difference between this output for 2 (gradient descent) and 3 (normal equation). This output is a more accurate representation of theta, the theta that gives a cost of 0. Adjusting the alpha and iterations of the gradient descent function would allow the approach to try and get more accurate theta values and a lower cost.

```
theta : [[0.5]
 [2. ]
 [1. ]]
```

4 - Linear Regression with one variable

b) Output plot of power vs price

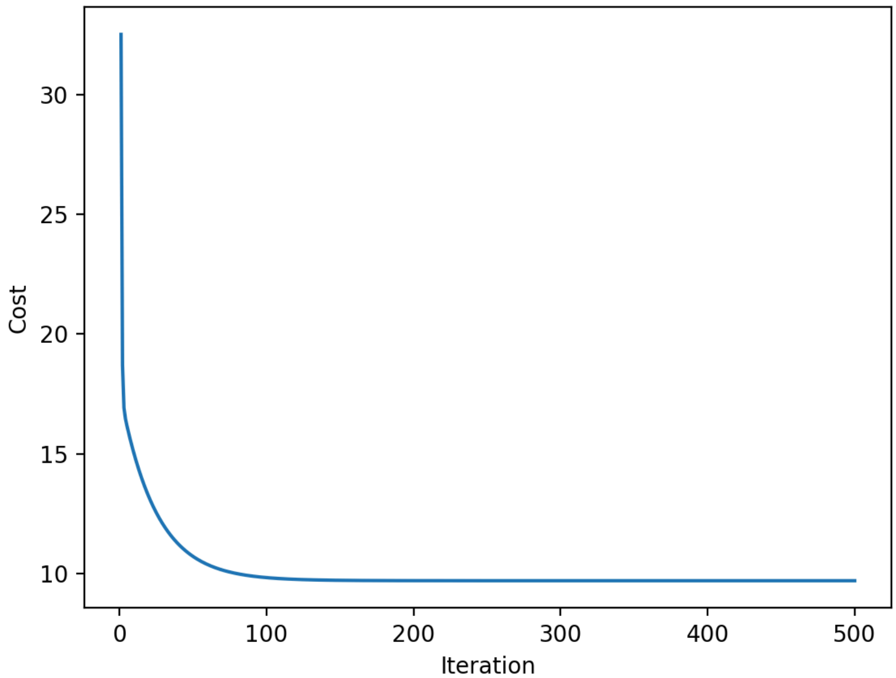


ps2-4-b.png

c) The size of feature matrix X and label vector y:

```
size of matrix X =  (179, 2)  and size of y =  (179, 1)
```

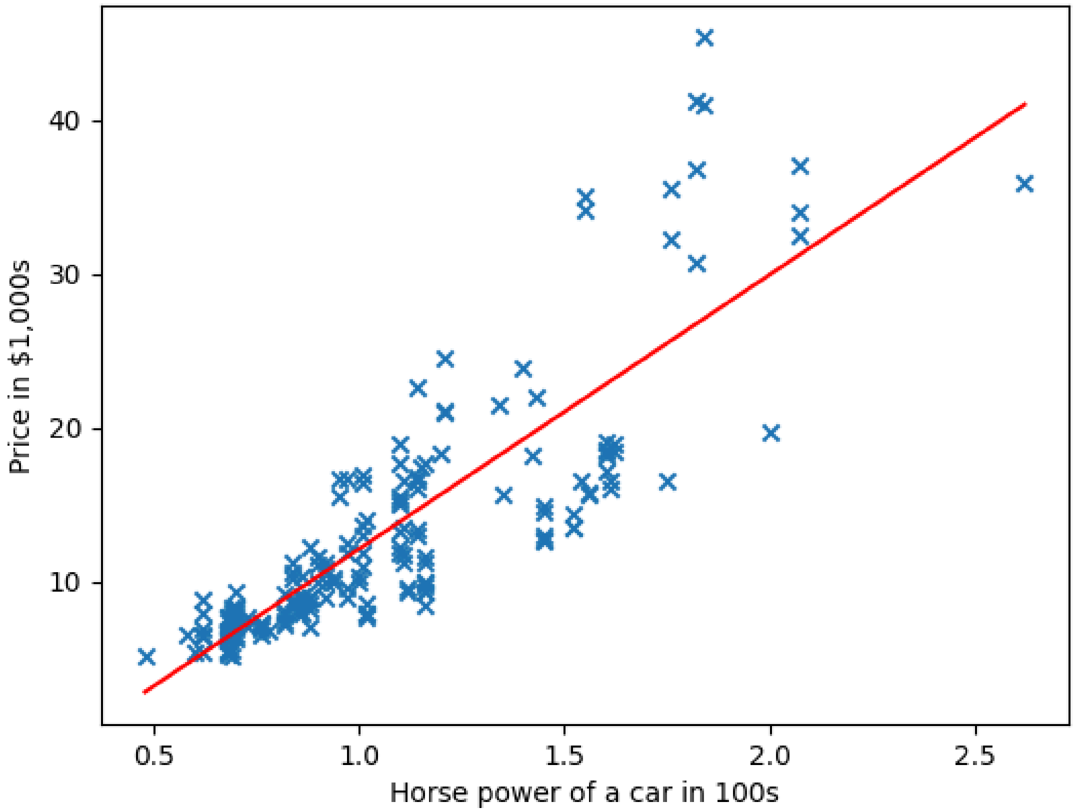
e)



ps2-4-e

```
theta =  [[-5.88156155]
 [17.89635565]]
```

f)



ps2-4-e

g)

Running multiple times to see how much the cost would vary, I got prediction errors ranging from around 3.5 to 12.

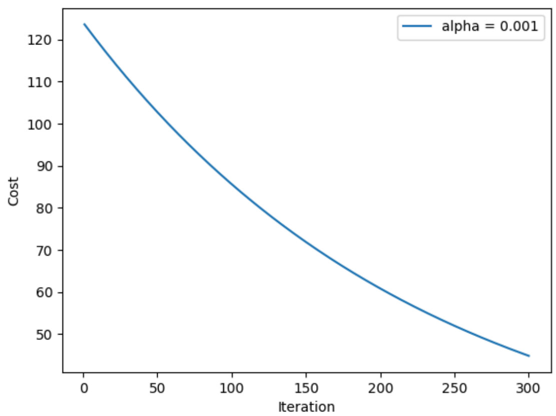
```
prediction error:  3.461546157219365

prediction error:  12.311066360428853
```

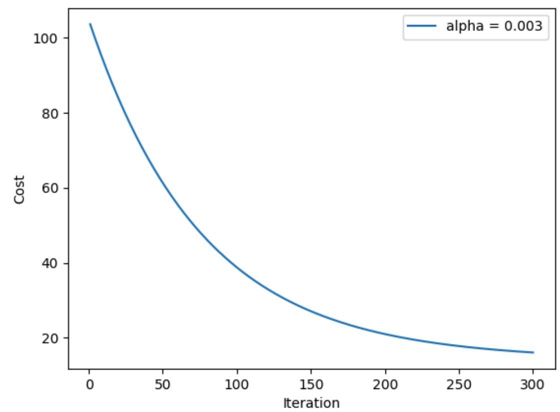
h) Comparing Gradient Descent and the Normal Equation approaches in this problem I received very similar theta values using both, and in turn getting very similar prediction errors for both, the difference between them being very small. I think this shows what was asked in section 3 about how you would change to get the theta's for both approaches to be closer to each other. Running the gradient descent function with more iterations gave it the chance to get closer and closer to the theta values.

```
rissa/ps2.py"
Gradient Descent theta =  [[-5.68428798]
 [17.69589197]]
Gradient Descent prediction error:  5.468387278632265
Normal Eq theta:  [[-5.68481679]
 [17.69636114]]
Normal Eq prediction error: 5.4684098972973745
```

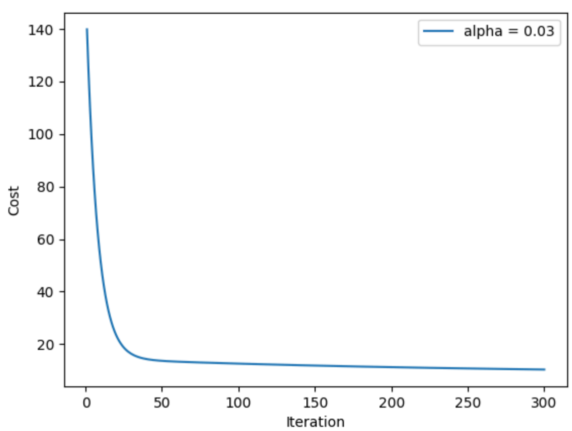
i)



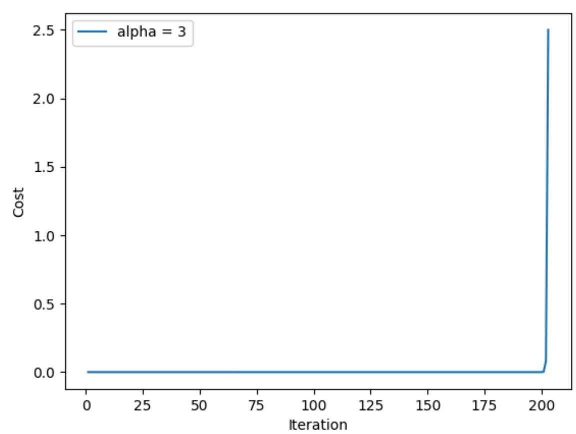
ps2-4-i-1



ps2-4-i-2



ps2-4-i-1



ps2-4-i-2

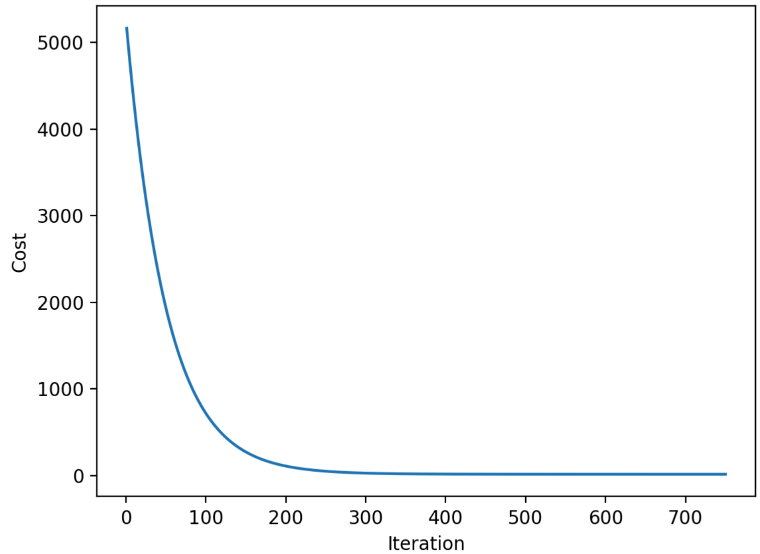
5 - Linear Regression with multiple variables

a)

```
mean: [1611.11111111 1292.27777778]
standard deviation:  [383.53456876 238.73737443]
```

```
size of X is:  (36, 3)
size of y is:  (36, 1)
```

b)



ps2-5-b

```
theta = [[101.97325961]
 [  2.79247863]
 [  2.00379149]]
```

c)

```
Co2 emission prediction:  [104.80368856]
```