PART 1

Learning rate=0.005

epoch	Θ0	Θ1	cost
100	12.23	15.32	560.49
500	35.14	8.81	273.51
1000	54.00	3.39	116.81

Learning rate=0.0005

epoch	Θ0	Θ1	cost
100	2.85	7.084	1275.38
500	8.40	15.31	622.44
1000	12.24	15.41	560.63

Learning rate=0.05

epoch	Θ0	Θ1	cost
1000	85.72	-5.86	15.15
1500	85.73	-5.80	15.15

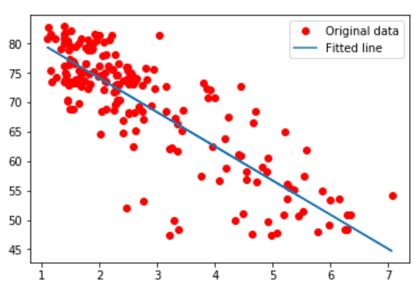
Learning rate=0.5

epoch	Θ0	Θ1	cost
1000	84.67	-6.11	17.56

Learning rate=0.1

epoch	Θ0	Θ1	cost
1000	85.71	-5.88	15.26

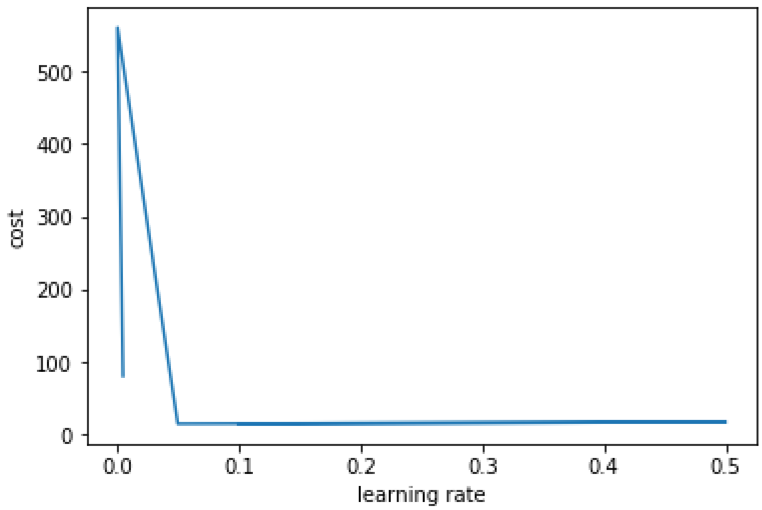
Minimum error/ Cost=15.15 for Θ0=85.72 and Θ1=-5.86.



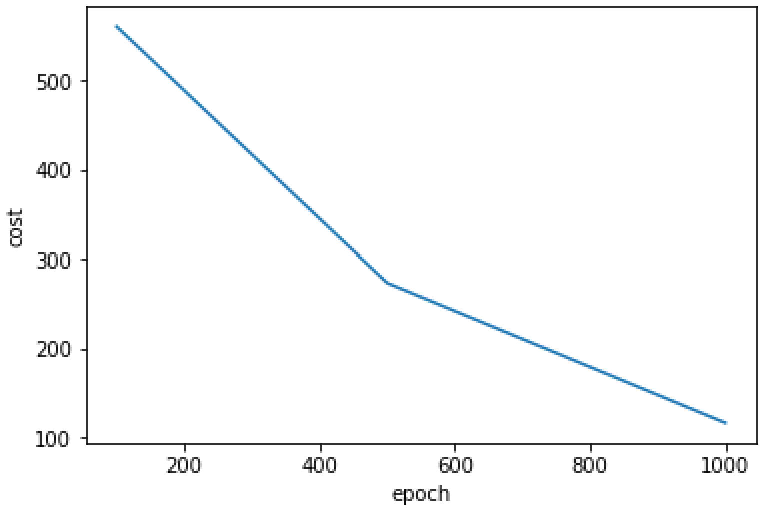
PART 2
a.

From the above collected data

learning rate v/s cost



Epoch v/s cost



b, c:
learning_rate=0.04
epoch=1000

	Batch	Stochastic	Gradient
θ 0	89.20802811	78.53744148623737	85.71903
θ 1	-6.72152272	-5.324618414138	-5.8876305
cost	-	-	15.26245
Time taken	0.33463634500003536	0.8141794109997136	79.91002489599987

We observe that the time complexity increases in the order of Batch < Stochastic <Gradient

We also observe that best approach to initialize Θ0 and Θ1 is Batch as it takes less time.

PART 3

a.

We observe form PART1 cost minimizes as the learning rate and epoch increases. But increasing learning rate beyond a threshold will again lead to increase of error.

From PART 2 we see that optimization also depends on the approach chosen, as gradient descent takes more time and batch approach takes less time.

b.

Another cost function that can be implemented is

$$C^{NC=} \frac{\Sigma(XY)}{(\Sigma X^2)^{1/2} \ (\Sigma Y^2)^{1/2}}$$