feature scalling will mak gradient descent much faster.

Feature Scaling:

Get every feature into approximately $-1 \leq x_i \leq 1$ range.

Mean normalization:

Replace x_i with $x_i - \mu_i$ to make features have approximately zero mean. (Don't apply to $x_0 = 1$)

$$\frac{x_i - \mu_i}{s_i}$$

μ_i = average value of x_i in training set

s_i = range of x_i (max - min)

Debugging:

1. ~~grad~~ $J(\theta)$ should decrease after every iteration

if the result is a polinomial $\theta_0 + \theta_1 x + \theta_2 x^2$

1. add a cubic variable: $\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3$

2. or add a $\sqrt{}$ function: $\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 \sqrt{x}$