Feature scaling helps gradient descent work much faster.

example

$\hat{price} = w_1 x_1 + w_2 x_2 + b$ (size → x_1; x_2 → no. of bedrooms)

x_1 : size (feet^2) range : 300 - 2,000

x_2 : #beds range : 0 - 5

one training example House : $x_1 = 2000$, $x_2 = 5$, price = \$500 K

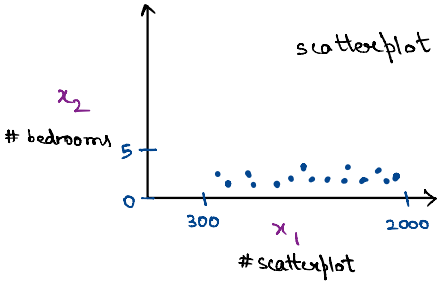
w_1, w_2 ?

$w_1 = 50$, $w_2 = 0.1$, $b = 50$	$w_1 = 0.1$, $w_2 = 50$, $b = 50$
price = $50 \times 2000 + 0.1 \times 50 + 50$	price = $0.1 \times 2000 + 50 \times 50 + 50$
$\Rightarrow$ price = \$100,050.5 K (> \$500K)	$\Rightarrow$ price = \$500K ✓✓ reasonable
↑ overestimate	

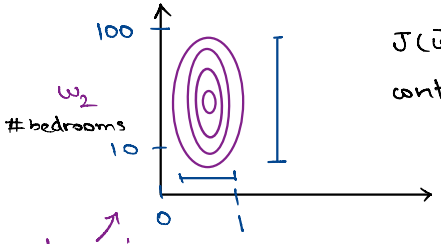
∴ If the features have a very large value then the parameters must be relatively small and vice-versa.

How will feature scaling solve our problem?

Features



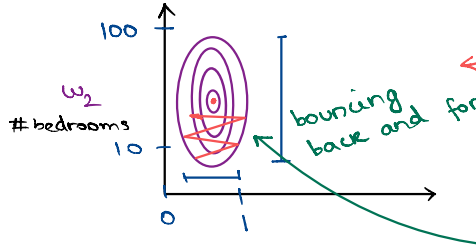
Parameters



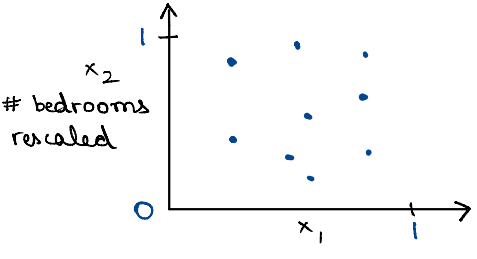
less change in w_1 causes change in prediction whereas much larger change in w_2 causes change in prediction

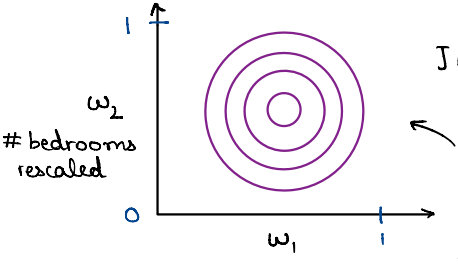
w_1 # size in feet^2

	size of feature w_j	size of parameter w_j
size in feet^2	⟷	⟷
# bedrooms	⟷	⟷



The problem with this is that since the gradient descent is so skinny it will bounce back and forth before it can reach global minimum.





by rescaling it is now easy to find the minimum

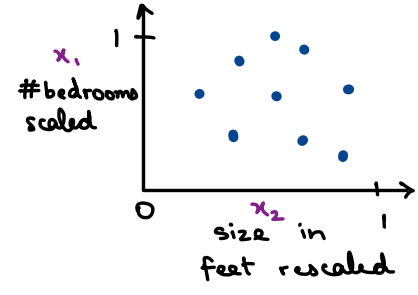
How do you actually scale features?

1. Feature Scaling

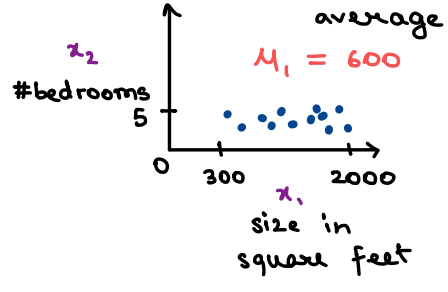
$300 \leq x_1 \leq 2000 \qquad 0 \leq x_2 \leq 5$

$$x_{1\,\text{scaled}} = \frac{x_1}{2000\ (\text{max})}, \quad x_{2\,\text{scaled}} = \frac{x_2}{5\ (\text{max})}$$

$0.15 \leq x_{1\,\text{scaled}} \leq 1 \qquad 0 \leq x_{2\,\text{scaled}} \leq 1$



2. Mean Normalization

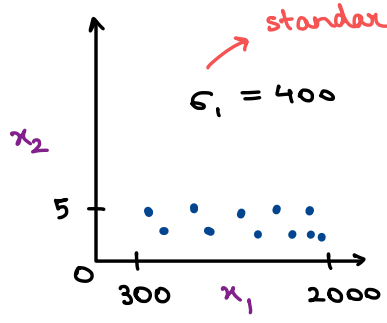


$300 \leq x_1 \leq 2000 \qquad 0 \leq x \leq 5$

$$x_1 = \frac{x_1 - \mu_1}{2000 - 300\ (\text{max} - \text{min})}, \quad x_2 = \frac{x_2 - \mu_2}{5 - 0}$$

$\Rightarrow -0.18 \leq x_1 \leq 0.82, \; -0.46 \leq x_2 \leq 0.54$

3. Z-score normalization



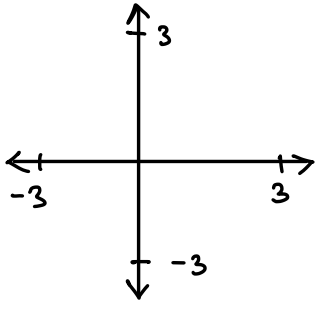
standard deviation $\Rightarrow$ $\sqrt{\frac{\left[\sum_{i=1}^{m}(x_i - \mu)\right]^2}{m-1}}$

$\sigma_1 = 400$

$300 \leq x_1 \leq 2000$ $\qquad$ $0 \leq x_2 \leq 5$

$x_1 = \frac{x_1 - \mu_1}{\sigma_1}$, $x_2 = \frac{x_2 - \mu_2}{\sigma_2}$

$-0.67 \leq x_1 \leq 3.1$ $\qquad$ $-1.6 \leq x_2 \leq 1.9$



Aim for feature scaling

→ aim for $-1 \leq x_j \leq 1$ for each feature x_j

→ " " $-3 \leq x_j \leq 3$ " " "

→ " " $-0.3 \leq x_j \leq 0.3$ " " "

- $0 \leq x_1 \leq 3$
- $-2 \leq x_2 \leq 0.5$

} If a particular value is in this range then still no rescaling is reqd.

- $-100 \leq x_3 \leq 100$ Rescaling required ! (too large)
- $-0.001 \leq x_4 \leq 0.001$ Rescaling required ! (too small)
- $98.6 \leq x_5 \leq 105$ Rescaling required ! (too large)

↑ even though they are real world fahrenheit values.