Gradient Descent

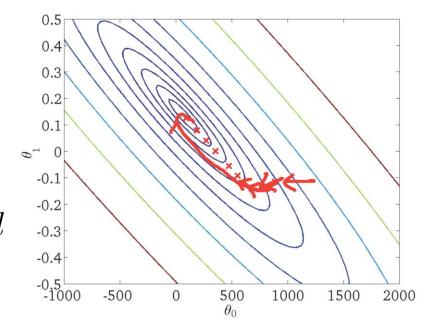
Batch Gradient Descent

```
Initialize θ
```

Repeat {
$$\theta_j \leftarrow \theta_j - \alpha \frac{1}{n} \sum_{i=1}^n \left(h_{\pmb{\theta}}\left(\mathbf{x}_i\right) - y_i\right) x_{ij} \qquad \text{for } j = 0...d$$
 }
$$\frac{\partial}{\partial \theta_j} J(\pmb{\theta})$$

for
$$j=0...d$$

$$rac{\partial}{\partial heta_j} J(oldsymbol{ heta})$$



Stochastic Gradient Descent

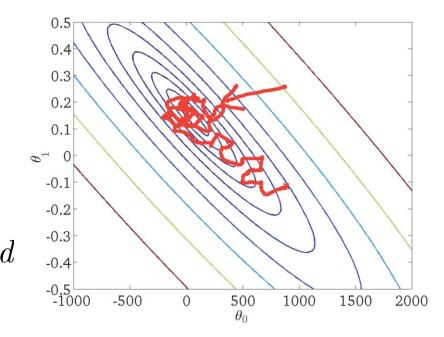
Initialize θ

Randomly shuffle dataset

Repeat {
$$(Typically 1 - 10x)$$

For
$$i=1...n$$
, do $heta_j \leftarrow heta_j - lpha\left(h_{m{ heta}}\left(\mathbf{x}_i
ight) - y_i
ight)x_{ij}$ for $j=0...d$ $heta_j \cot_{m{ heta}}\cot_{m{ heta}}\cot_{m$

for
$$j=0...$$
 $\cot_{m{ heta}}(\mathbf{x}_i,y_i)$



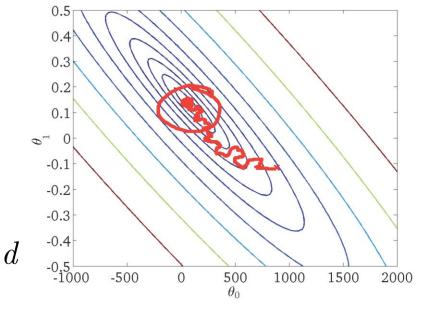
Adaptive alpha is not required for hw3 (Just follow the previous slide for hw3)

Stochastic Gradient Descent

```
Initialize θ
```

Randomly shuffle dataset

```
Repeat { (Typically 1 – 10x)}  \text{For } i = 1...n \text{, do} \\ \theta_j \leftarrow \theta_j - \alpha \left(h_{\boldsymbol{\theta}}\left(\mathbf{x}_i\right) - y_i\right) x_{ij} \qquad \text{for } j = 0...d \\ \frac{\partial}{\partial \theta_j} \mathrm{cost}_{\boldsymbol{\theta}}(\mathbf{x}_i, y_i)
```



Learning rate α is typically held constant. Can slowly decrease α over time if we want θ to converge. (E.g. $\alpha = \frac{\text{const1}}{\text{iterationNumber + const2}}$)