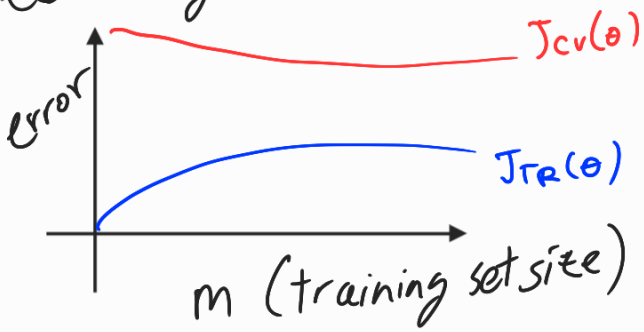
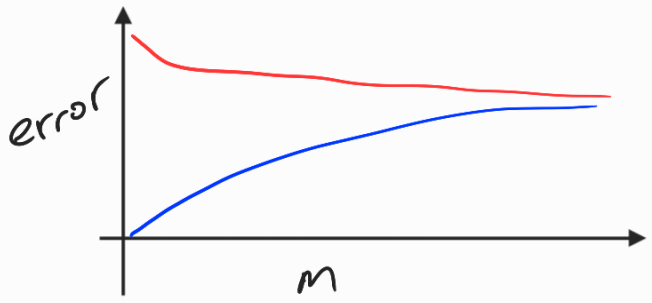
Before going to train with the large dataset, we might pick a small sample and plot the learning curves.





Stochastic gradient descent: 1 example x iteration

1st: Randomly shuffle dataset

2nd: Repeat {

for (i = 1 : m) { for (j = 0 : n) {

$\theta_j = \theta_j - \alpha \left(h_\theta(x^{(i)}) - y^{(i)} \right) \cdot x_j^{(i)}$

} } }

Convergence:

Compute $cost(\theta, (x^{(i)}, y^{(i)}))$ before updating θ using $(x^{(i)}, y^{(i)})$

Then, plot cost averaged

Increases → smaller α

Noisy → increase nº examples to average over

We can slowly decrease α on time $\left(\frac{const\,1}{iter\ number + const\,2} \right)$

Mini-batch gradient descent: b examples/iteration

```
for (i in range(1, n, b)){
    (...)
```

Convergence: plot $J(\theta)$ of the number of iterations.