Machine Learning

Abdelhak Mahmoudi
abdelhak.mahmoudi@um5.ac.ma

INPT - 2020

Content

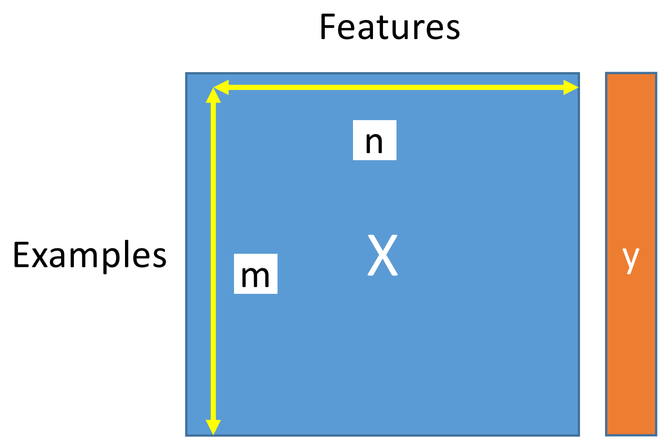
1. The Big Picture
2. Supervised Learning
   - Linear Regression, Logistic Regression, Support Vector Machines, Trees, Random Forests, Boosting, Artificial Neural Networks
3. Unsupervised Learning
   - Principal Component Analysis, K-means, Mean Shift

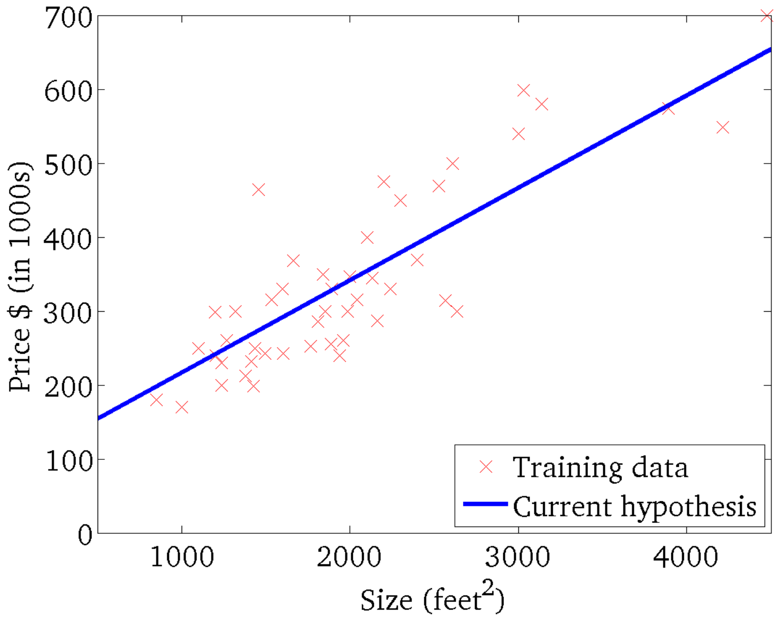
Supervised Learning

- Linear Regression
- Logistic Regression
- Support Vector Machines
- Trees (Decision and Regression)
- Random Forests
- Boosting
- Artificial Neural Networks

Linear Regression

- The output y is continuous
- Fit X with a line $y = w_0 + w_1 x$
- The best line is the line with minimum loss $L(w)$
- Solved using Normal Equations
  - $W = (X^T X)^{-1} X^T y$
  - But not for big X !
- Find W iteratively using gradient descent



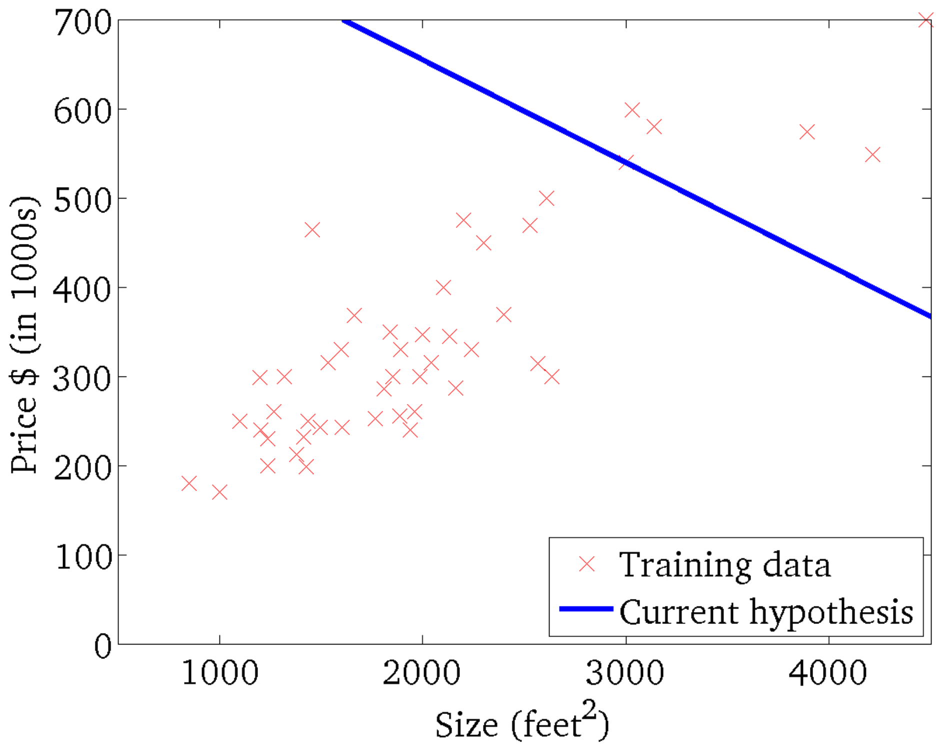


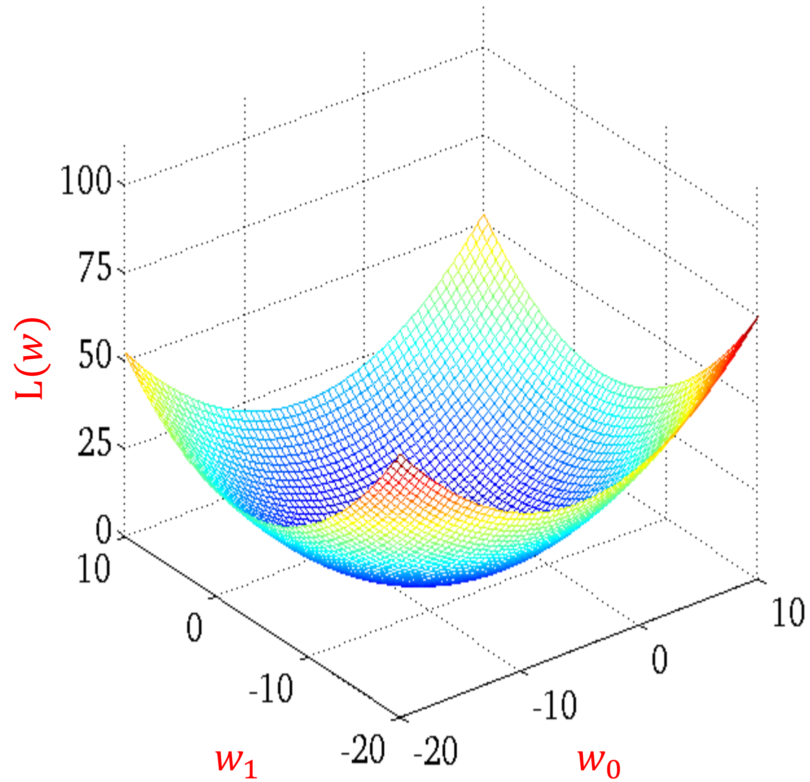
Gradient Descent

(Batch) GD

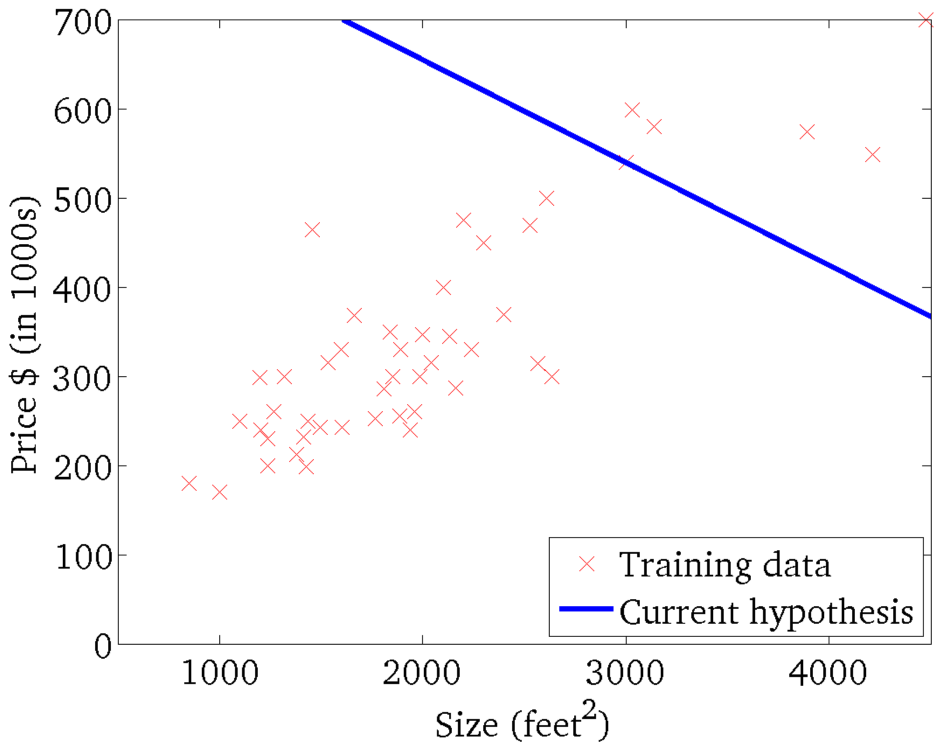
```
X = data_input
Y = data_output
W = initialize_parameters()
for it in range(num_iterations):
    Yhat = h(X, W)
    L = loss(Yhat, Y)
    dW = gradient(L(W))
    W = W − α dW
```

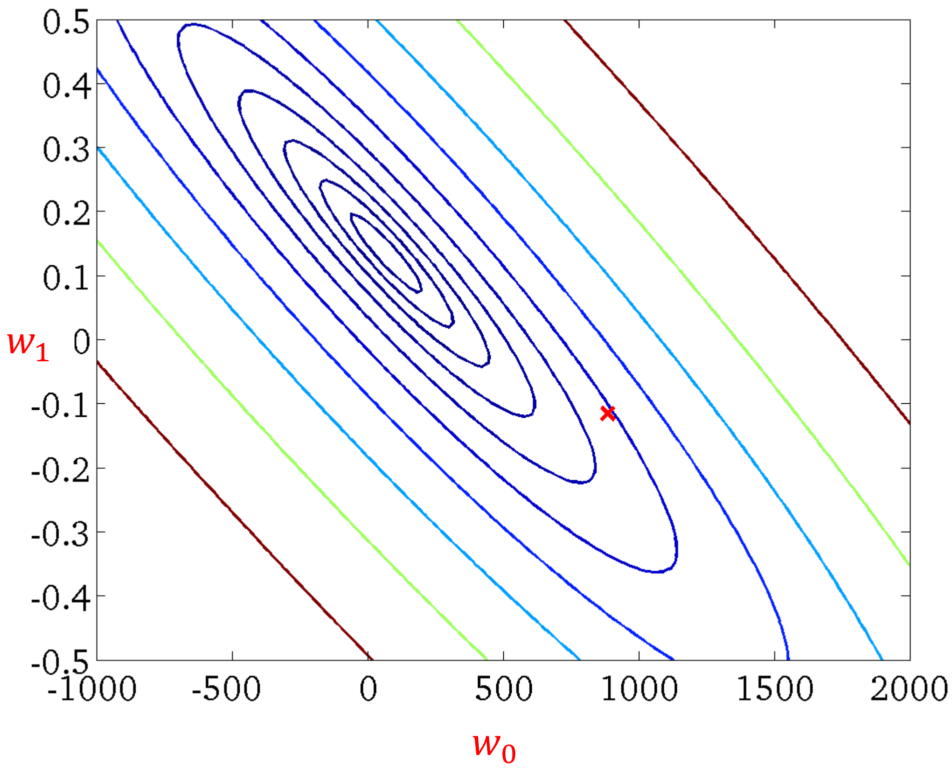
Linear Regression



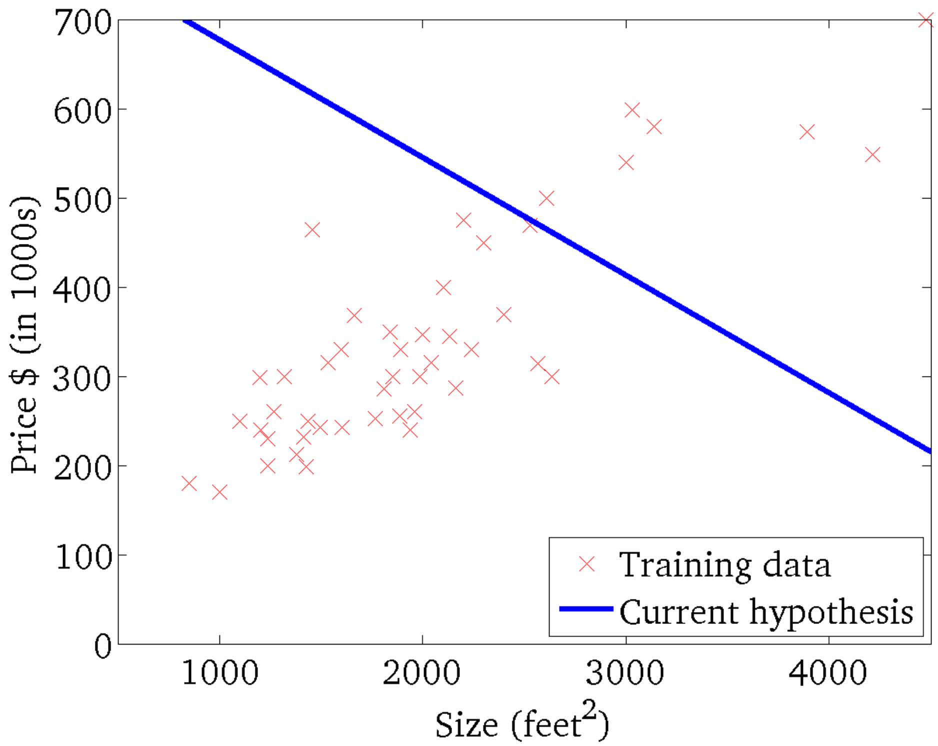


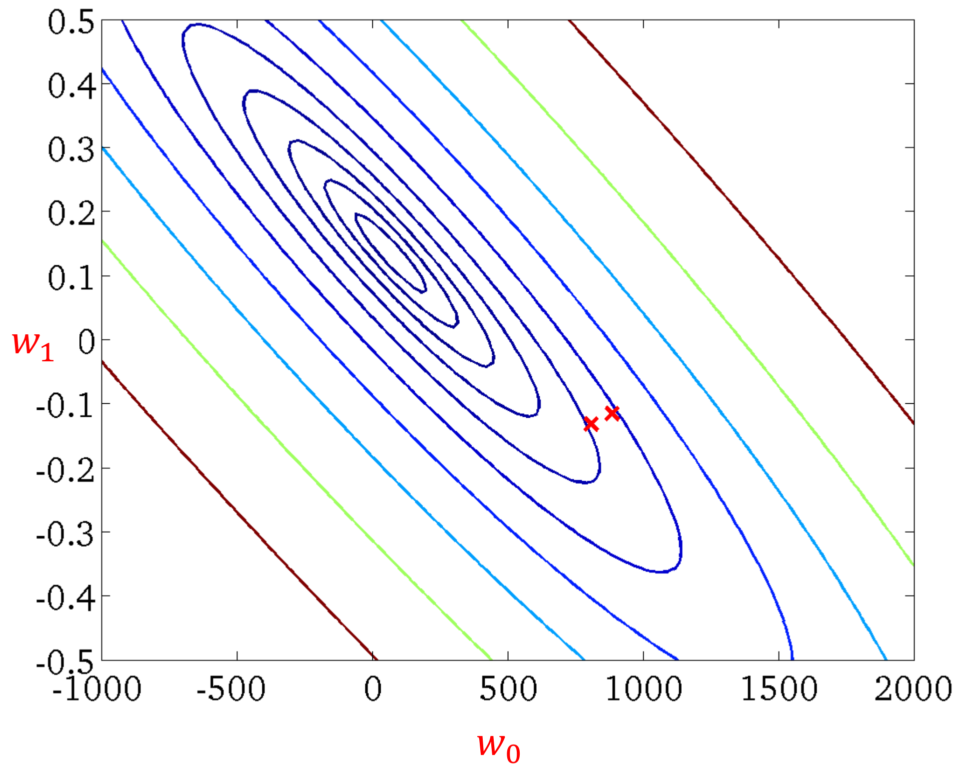
Linear Regression



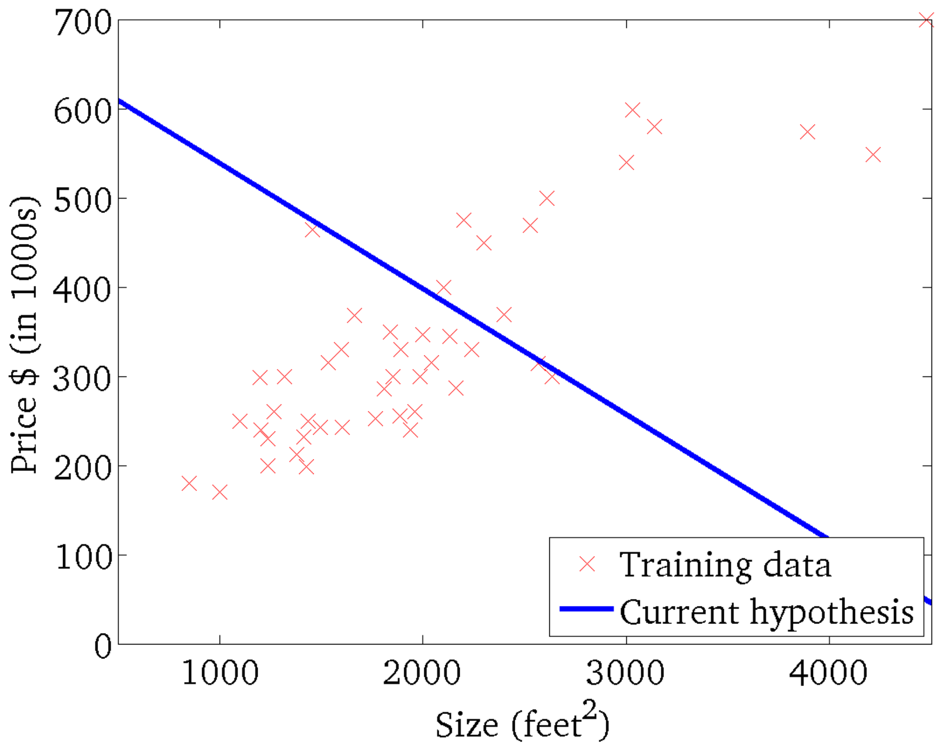


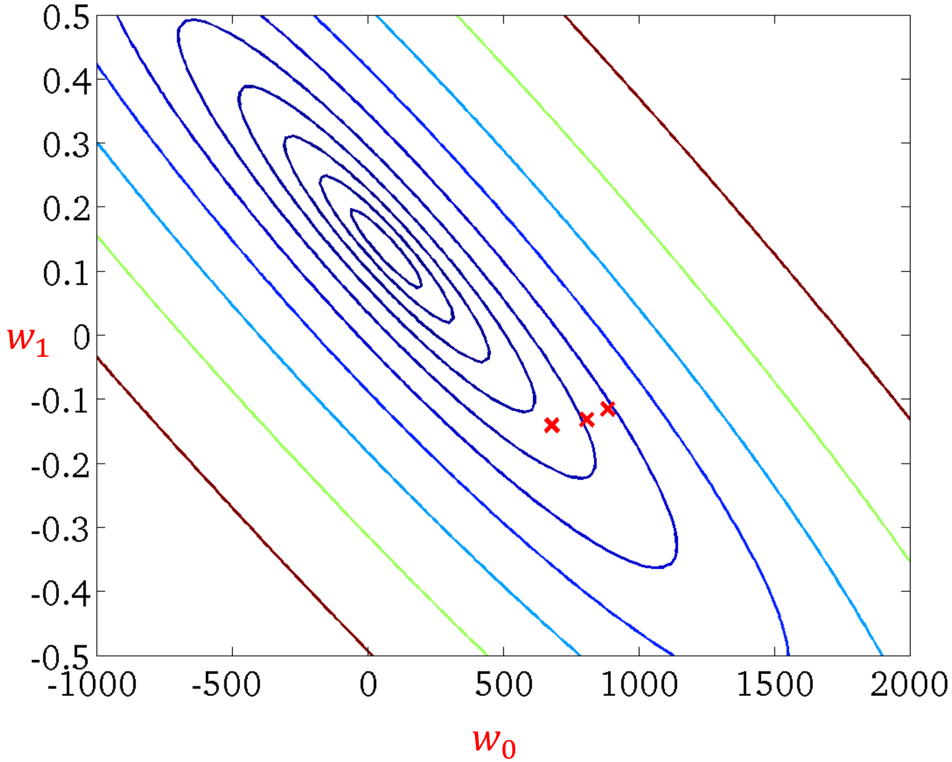
Linear Regression



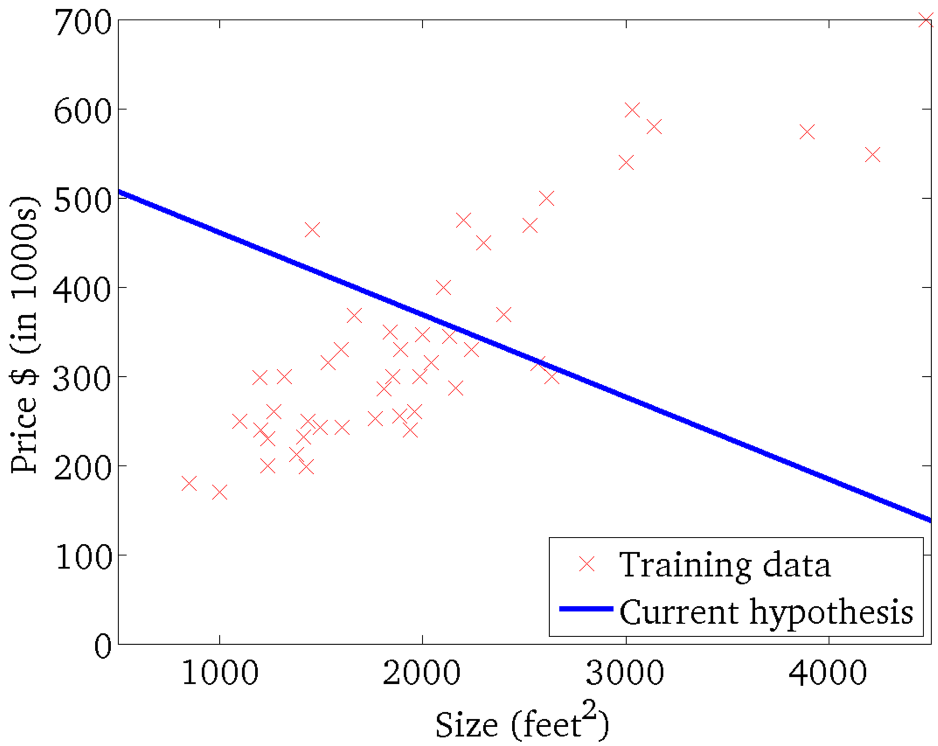


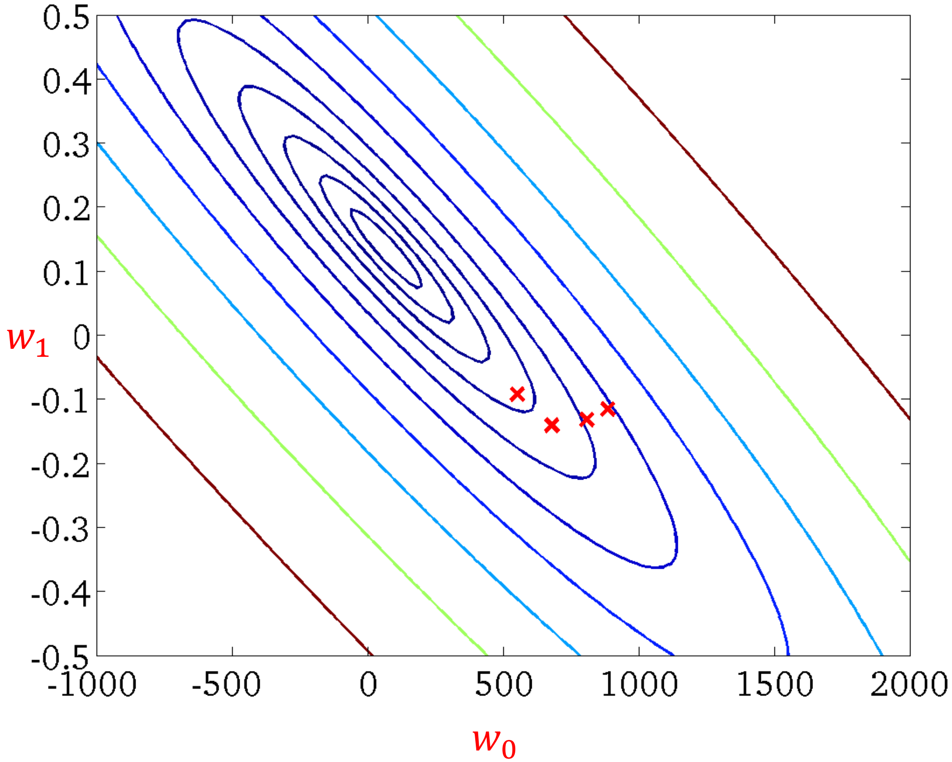
Linear Regression



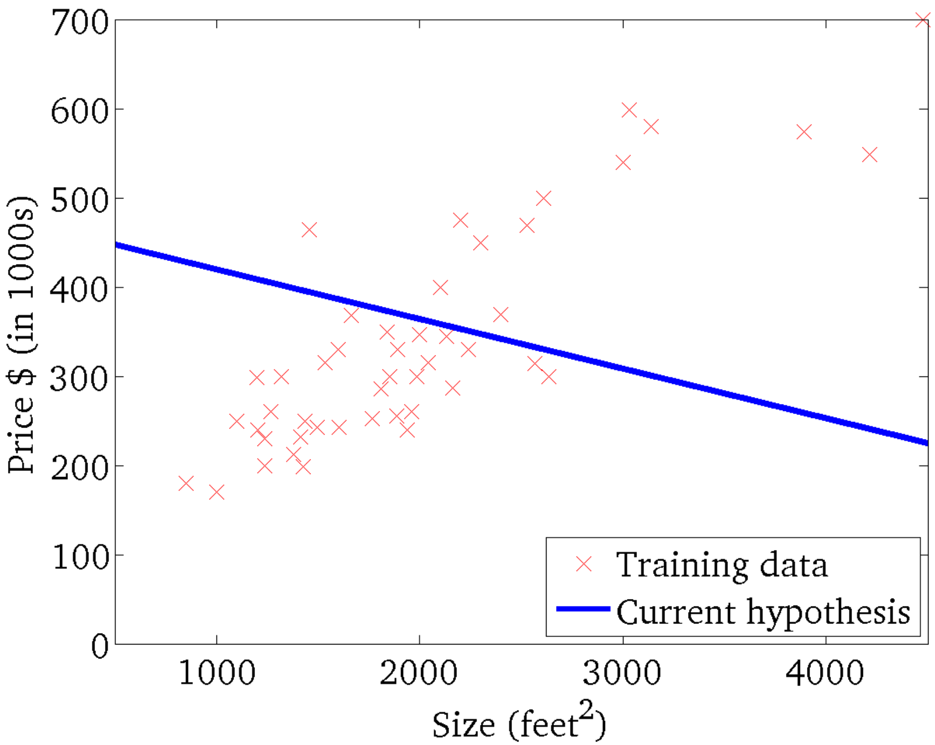


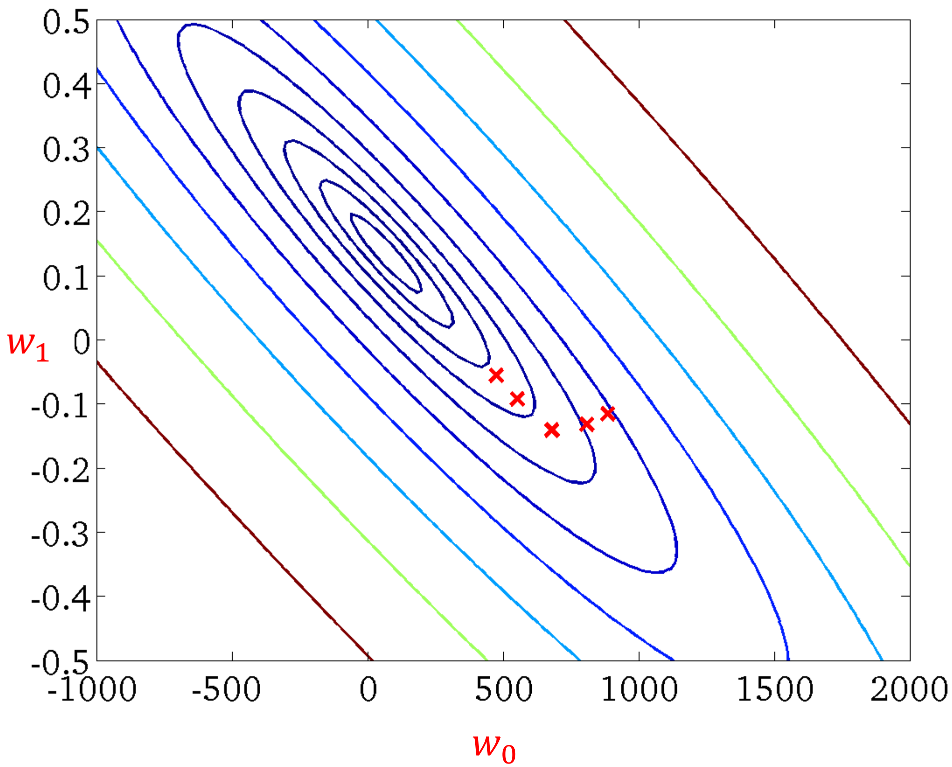
Linear Regression



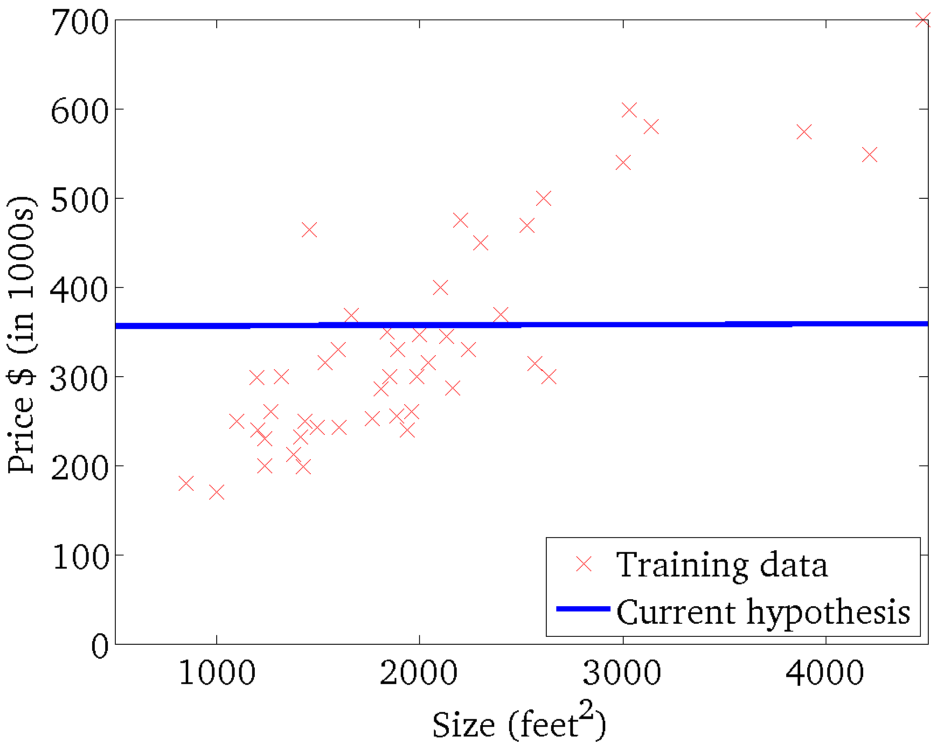


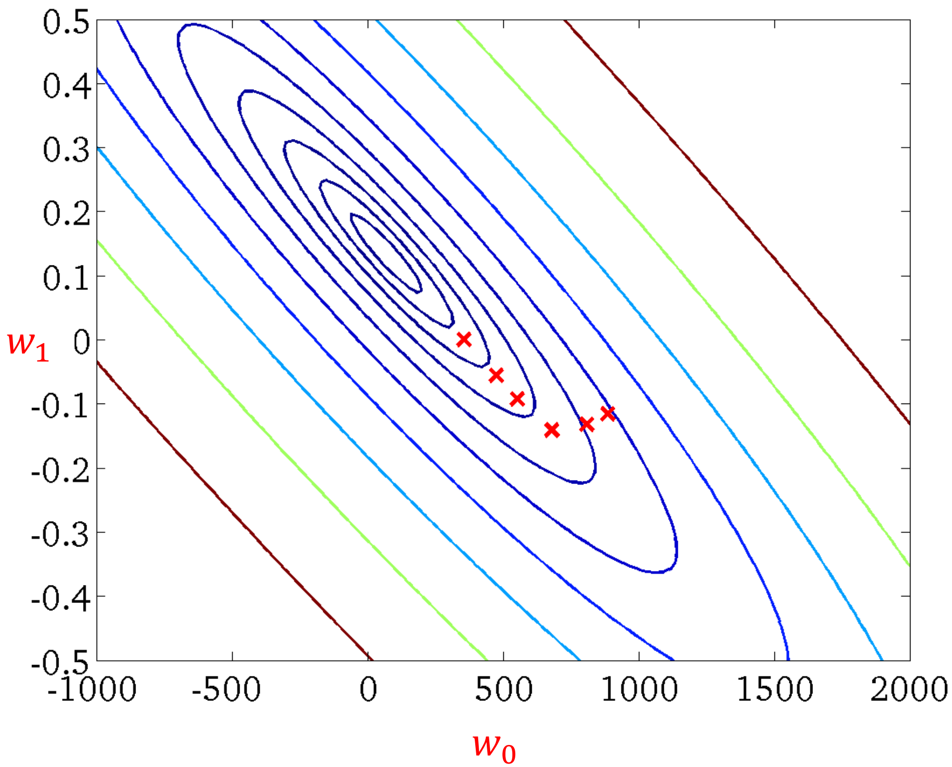
Linear Regression



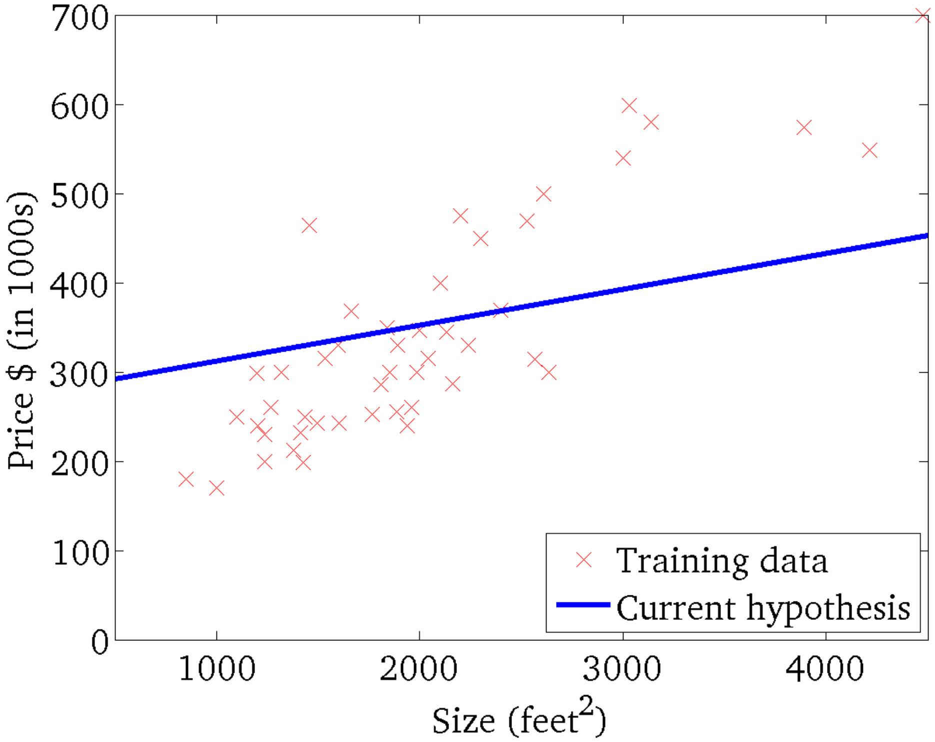


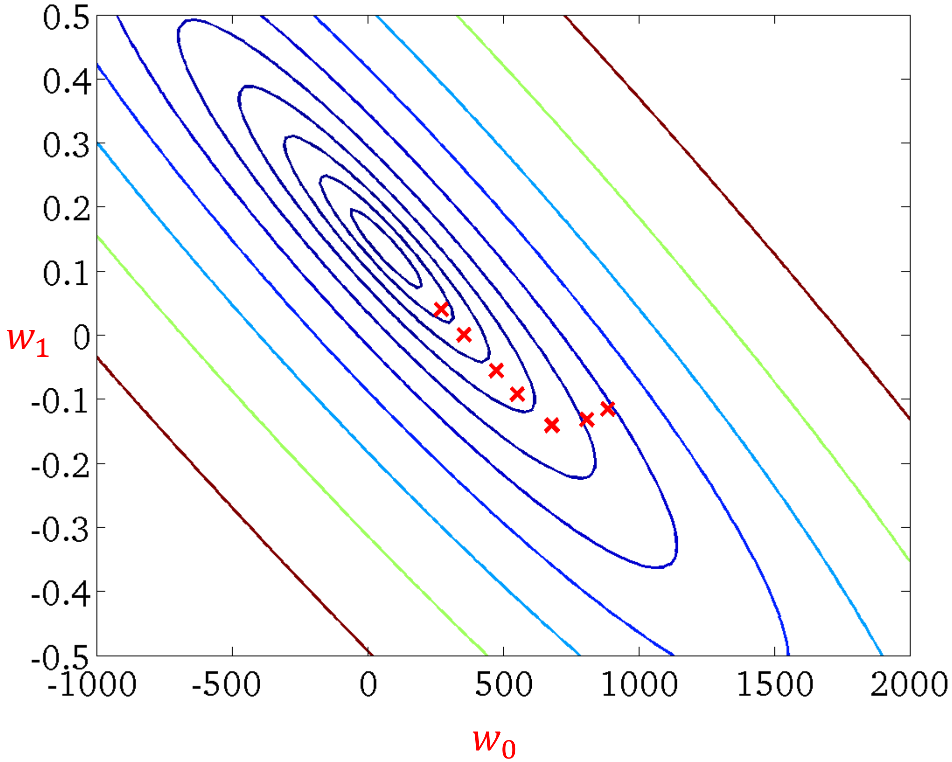
Linear Regression



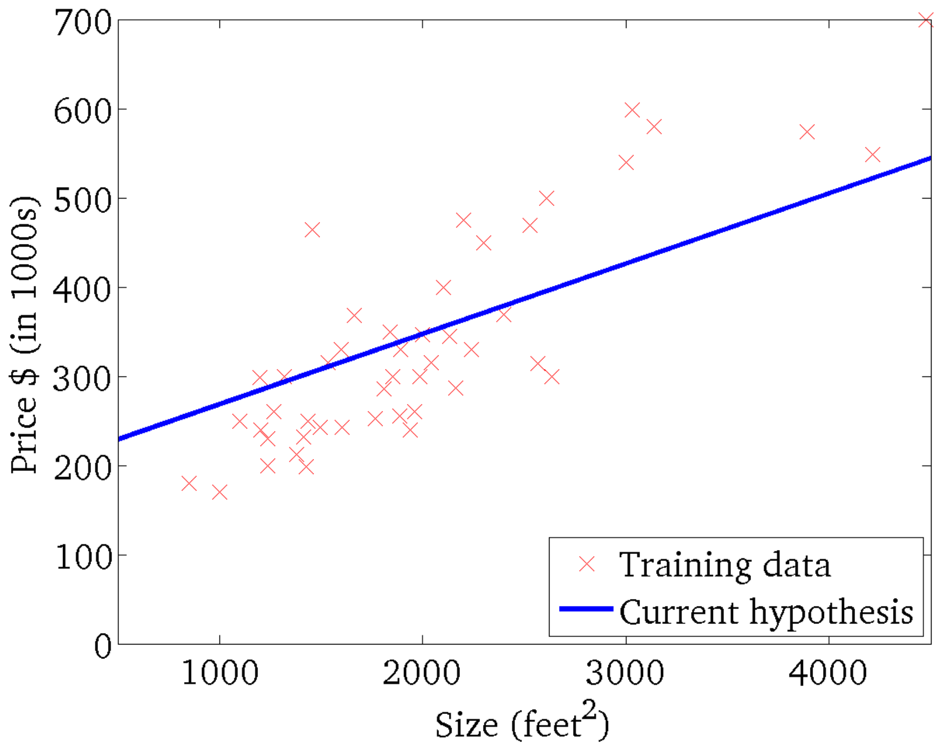


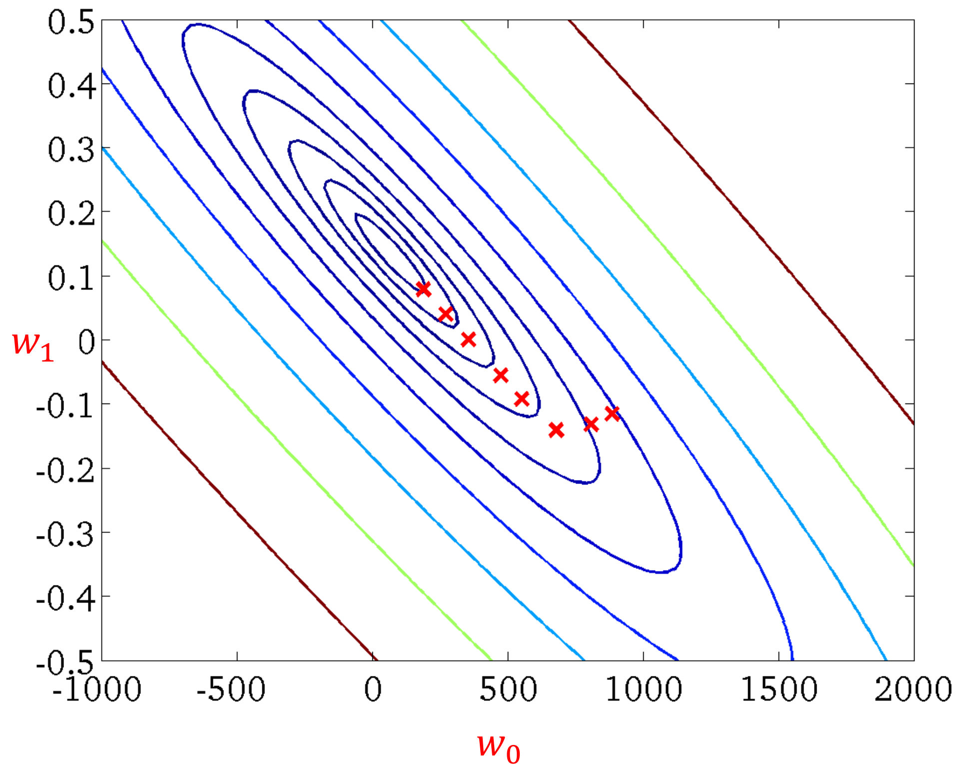
Linear Regression



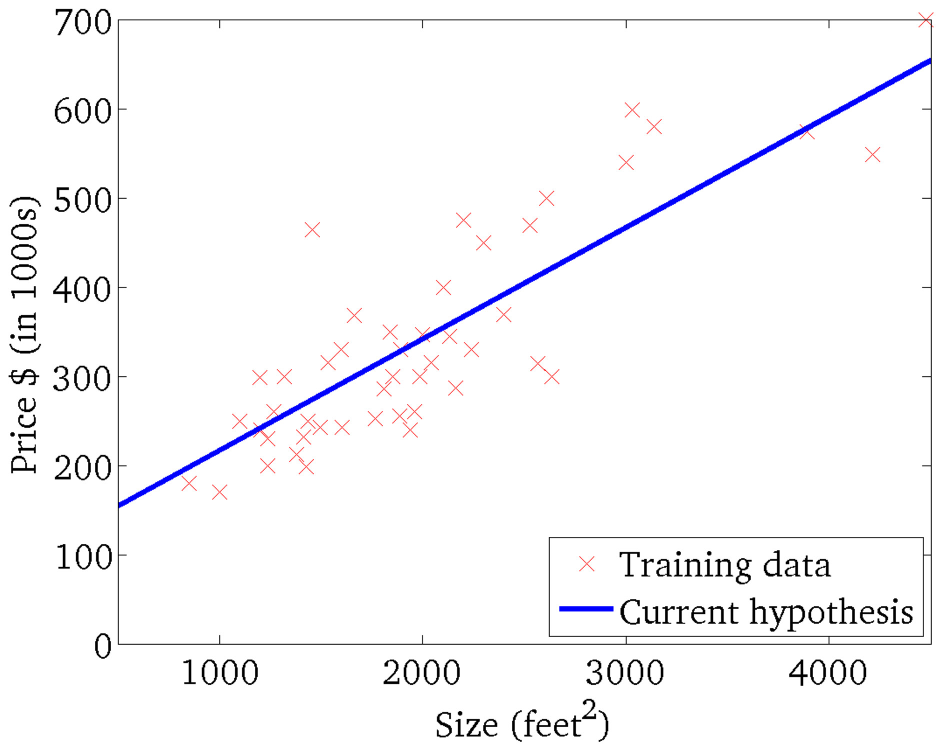


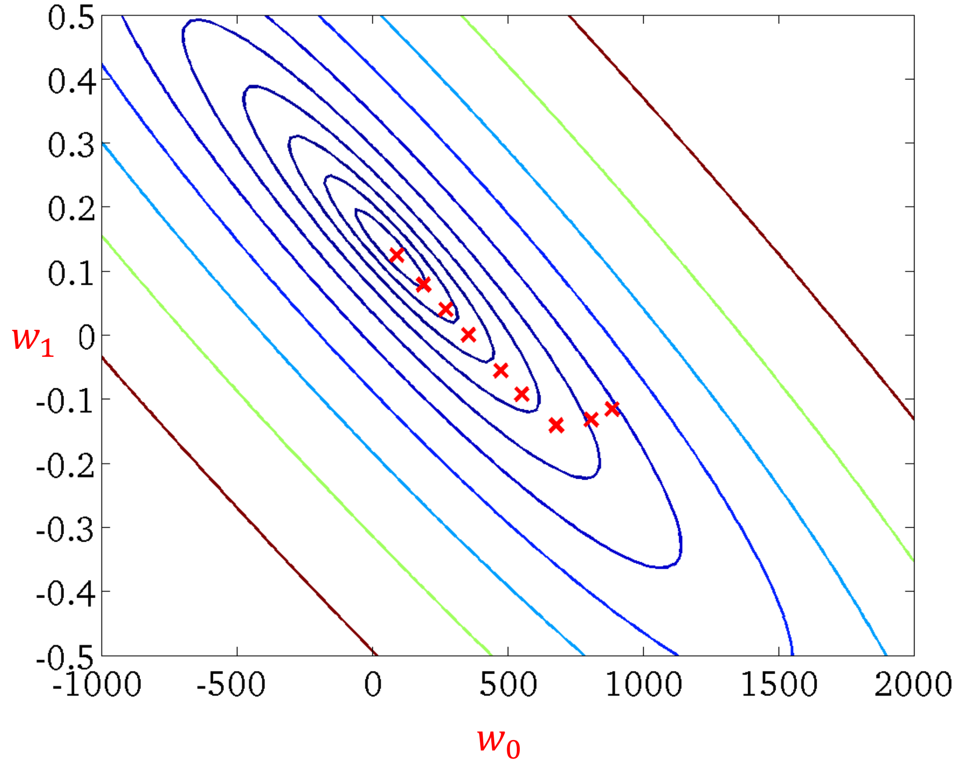
Linear Regression





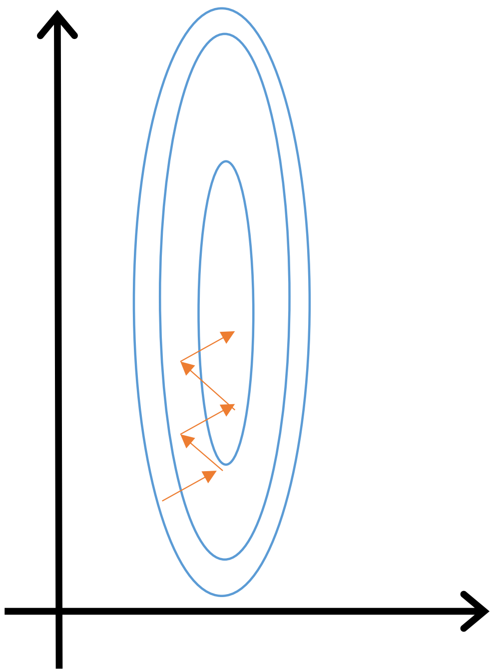
Linear Regression





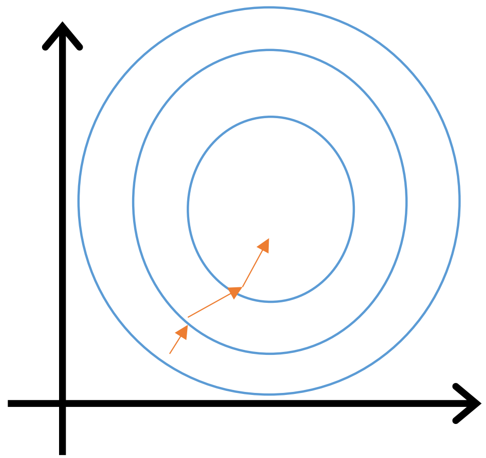
Feature Scaling

Problem: features are not on a similar scale

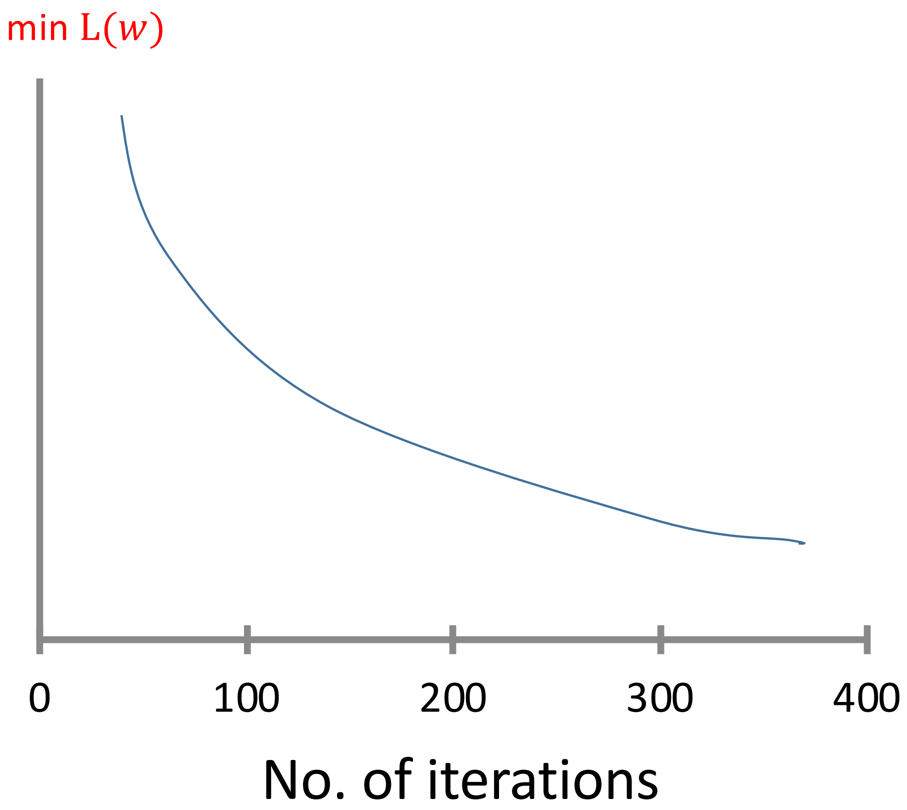


Solution: Mean Normalization

$$\frac{x_j - \mu_j}{\sigma_j} \qquad -1 \leq x_j \leq 1$$

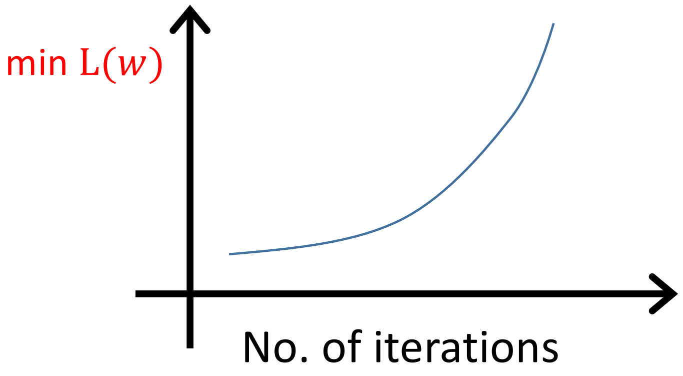


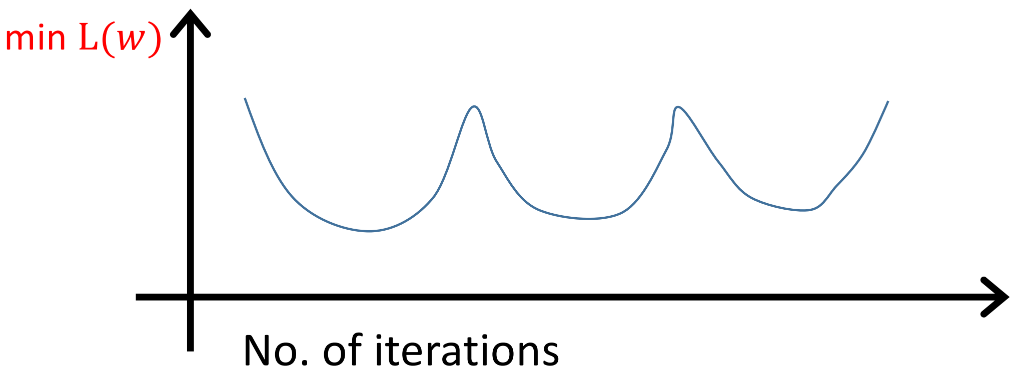
Gradient Descent: Debugging



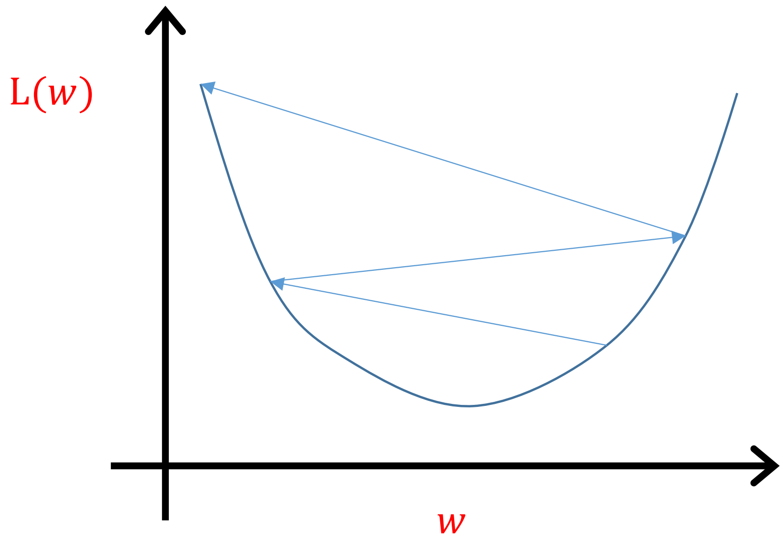
- How to make sure gradient descent is working correctly?
- How to choose learning rate
- Solution: Declare convergence if $\mathrm{L}(w)$ decreases by less than 10^{-3} in one iteration.

Gradient Descent: Debugging





Gradient descent not working. Use smaller α.



- For sufficiently small α, $\mathrm{L}(w)$ should decrease on every iteration.
- But if α is too small, gradient descent can be slow to converge.

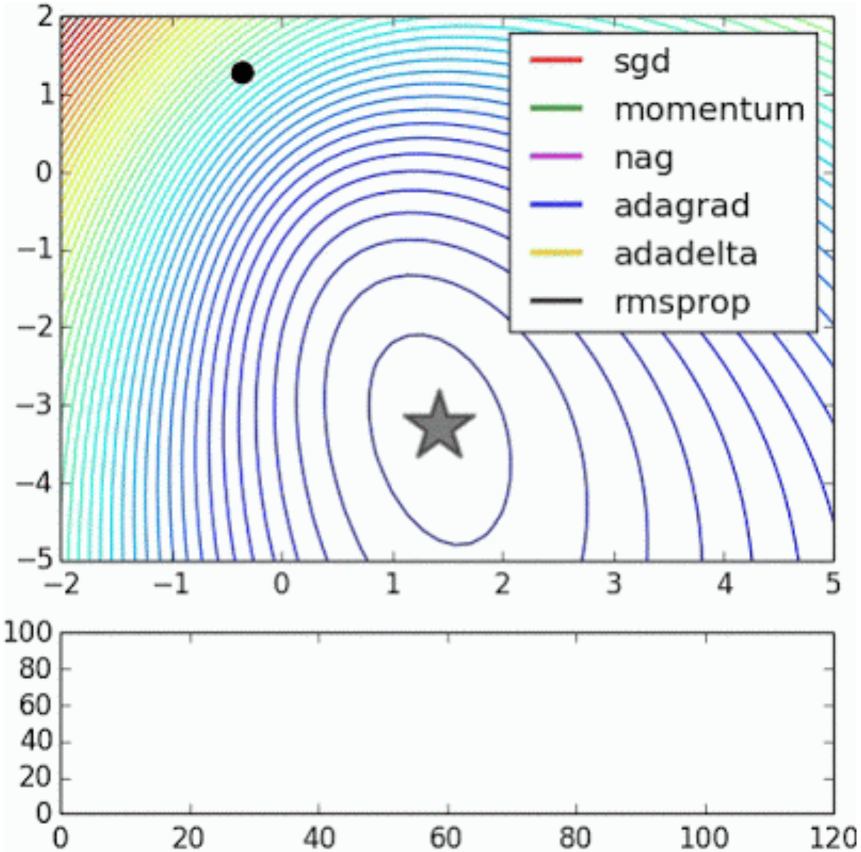
Gradient Descent: Debugging

- If α is too small: slow convergence.
- If α is too large: $\mathrm{L}(w)$ may not decrease on every iteration; may not converge.

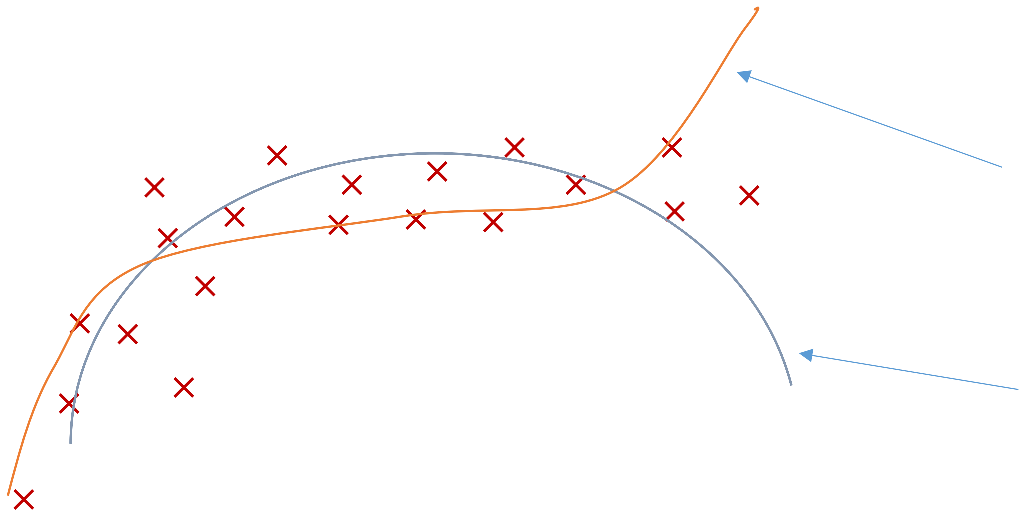
To choose α, try

$$\ldots, 0.001, \qquad , 0.01, \qquad , 0.1, \qquad , 1, \ldots$$

Other Optimization Methods



Polynomial Regression



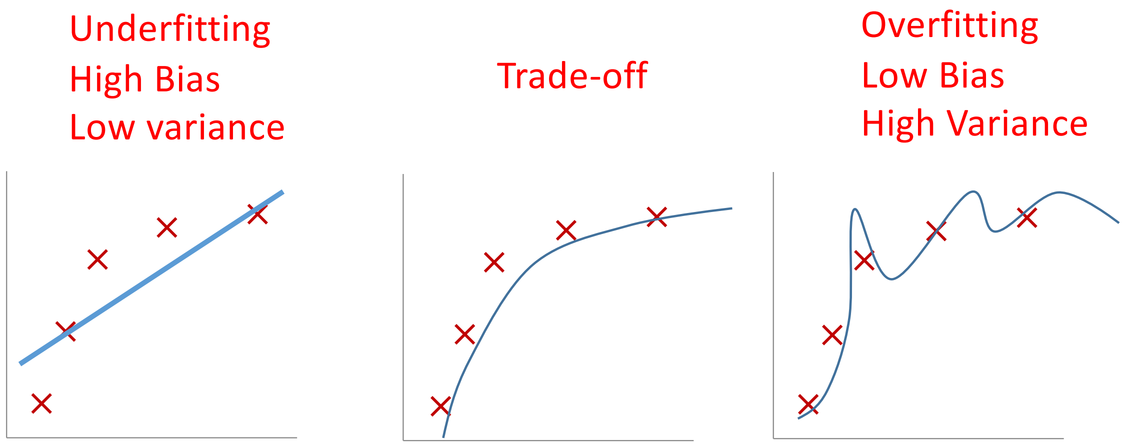
$y = w_0 + w_1 x + w_2 x^2 + w_3 x^3$

Good but go up !

$y = w_0 + w_1 x + w_2 x^2$

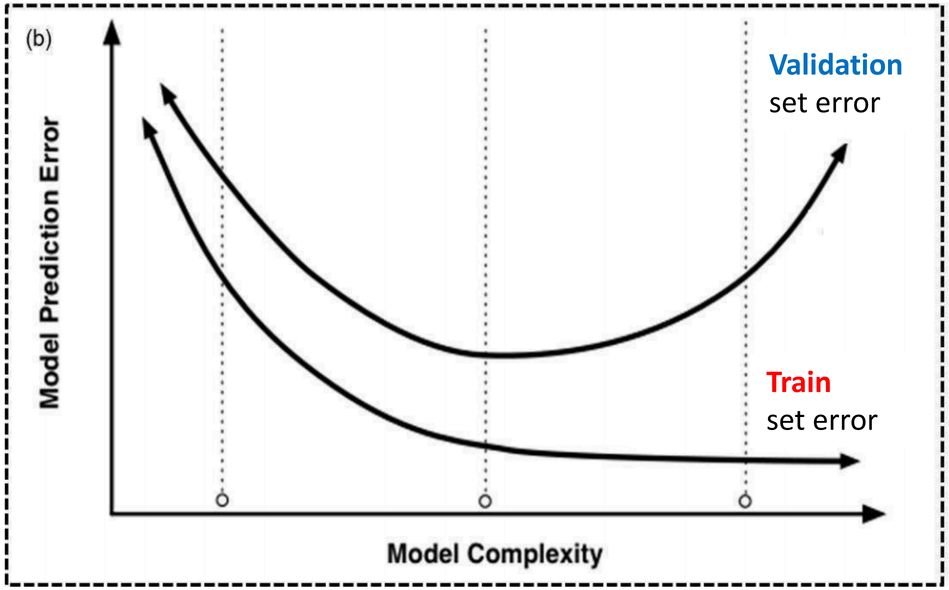
Good but go down !

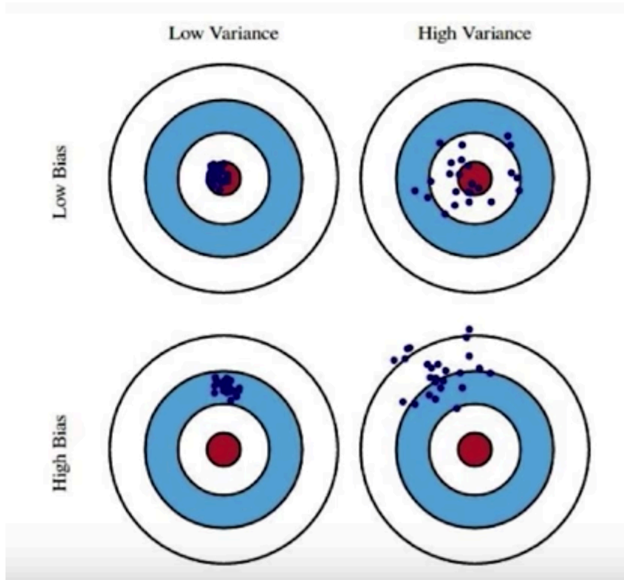
Overfitting vs. Underfitting



Bias-Variance Tradeoff

Expected error (Human or Bayes optimal): **0%**	**Train** set error	1%	15%	15%	0.5%
	Validation set error	11%	16%	30%	1%
		High variance	High bias	High bias High variance	Low bias Low variance

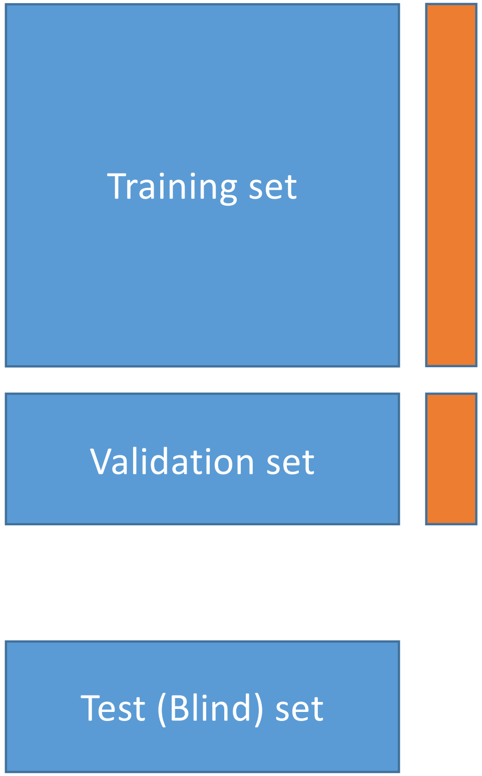


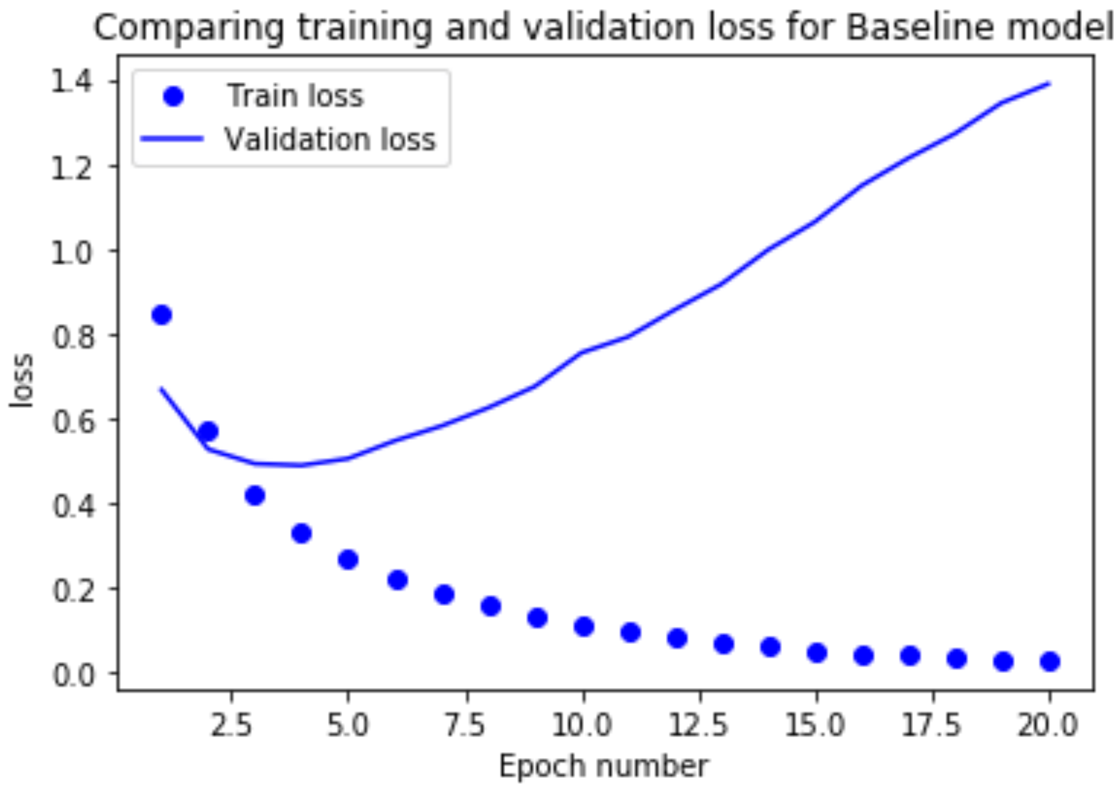


Address Overfitting

- Detect Overfitting
  - Performance analysis (Cross-Validation)
- Avoid Overfitting
  - Fewer features (Feature Selection, Dimensionality Reduction)
  - Constraint the model (Regularization : minimum loss $\mathrm{L}(w) + \lambda\, ww^T$)
  - Model Selection (Tune hyper-parameters using Grid Search)

Performance Analysis





Performance Measures

- Measure of distance between predictions $\hat{y} = \mathrm{h}(\mathrm{x})$ and targets y

- L2 norm: Root Mean Square Error (RMSE)
  - Sensitive to outliers !

$$\mathrm{RMSE}(\mathbf{X}, h) = \sqrt{\frac{1}{m} \sum_{i=1}^{m} \left(h\left(\mathbf{x}^{(i)}\right) - y^{(i)}\right)^2}$$

- L1 norm: Mean Absolute Error (MSE)
  - Derivability !

$$\mathrm{MAE}(\mathbf{X}, h) = \frac{1}{m} \sum_{i=1}^{m} \left| h\left(\mathbf{x}^{(i)}\right) - y^{(i)} \right|$$

Feature Selection

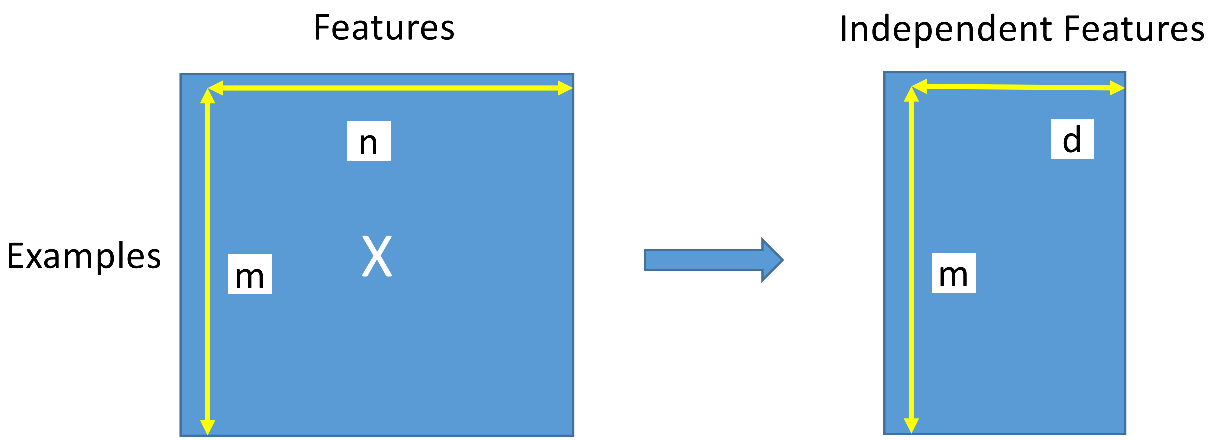
- Best Subset Selection
  Fit a separate least squares regression for each possible combination of the n features: 2^n possibilities!

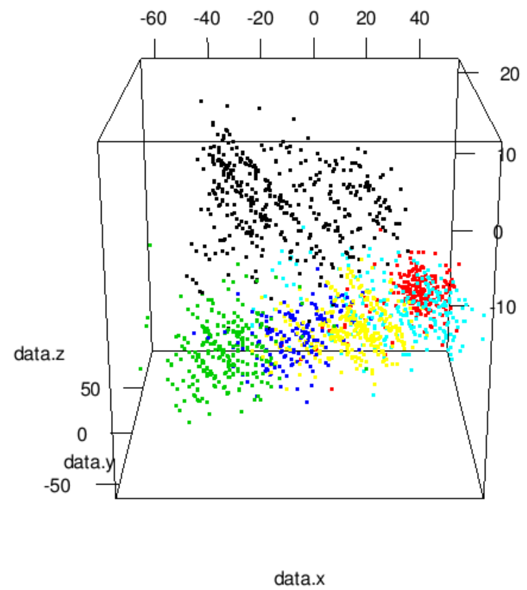
- Forward Stepwise Selection
  Begins with a model containing no feature, and then adds the feature that gives the greatest improvement (smallest cost) to the model, one-at-a-time.

- Backward Stepwise Selection
  Begins with a model containing all feature, and then removes the feature that gives the smallest improvement (highest cost) to the model, one-at-a-time.

Dimensionality Reduction

- Reducing or extracting features





Regularization

- See regularization as a penalty against complexity. Increasing the regularization strength penalizes "large" W

- The goal is to prevent the model from picking up "peculiarities," "noise," or "imagines a pattern where there is none.”

Regularization: Ridge Regression (L$_2$ norm)

Linear Regression

$$\hat{y} = h_w(x) = w_0 + w_1 x_1 + w_2 x_2$$

if λ is set to be extremely large, then w_j have to be very small.

→Algorithm results in underfitting

→Gradient Descent will fail to converge

$$\underset{w}{\text{minimize}} \quad L(y, \hat{y})$$

L$_2$ norm

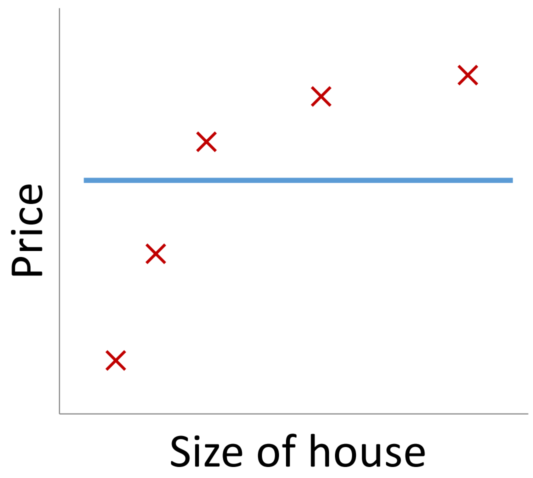
$$\underset{w}{\text{minimize}} \; L(y, \hat{y}) + \lambda \boxed{\sum_{j=1}^{n} w_j^2}$$

Do not regularize for j=0

Training $w_0 = 1, w_1 = 2, w_2 = 0.01$

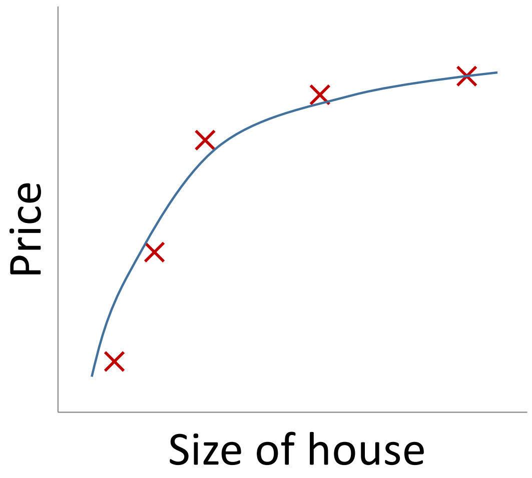
Test $w_0 = 1, w_1 = 2, w_2 = 0$

Regularization: Ridge Regression (L_2 norm)



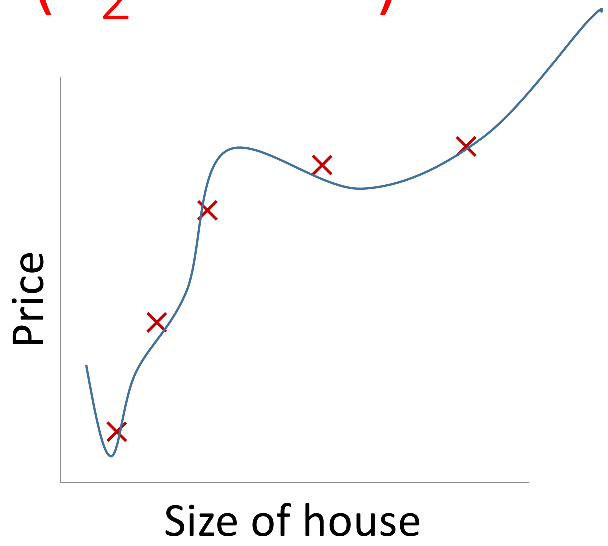
$$y = w_0 + \cancel{w_1} x + \cancel{w_2} x^2 + \cancel{w_3} x^3$$

Underfitting



$$y = w_0 + w_1 x + w_2 x^2 + \cancel{w_3} x^3$$

Tradeoff



$$y = w_0 + w_1 x + w_2 x^2 + w_3 x^3$$

Overfitting

Regularization: LASSO Regression (L_1 norm)

Linear Regression

$$\hat{y} = h_w(x) = w_0 + w_1 x_1 + w_2 x_2$$

- LASSO: Least Absolute Shrinkage and Selection Operator
- LASSO is not differentiable for every value of w, but performs best feature selection

$$\underset{w}{\text{minimize}} \quad L(y, \hat{y})$$

L_1 norm

$$\underset{w}{\text{minimize}} \quad L(y, \hat{y}) + \lambda \boxed{\sum_{j=1}^{n} |w_j|}$$

Do not regularize for j=0

Training $\quad w_0 = 1, w_1 = 2, w_2 = 0$

Test $\quad w_0 = 1, w_1 = 2, w_2 = 0$

Model Selection

- Hyper-Parameters Tuning
  - λ : regularization hyper-parameter
  - d: degree of polynomial
  - Etc.
- Grid Search
- Randomized search