Regularized linear regression

Topics we'll cover

1. Generalization
2. Regularization
3. Ridge regression
4. Lasso

Least-squares regression

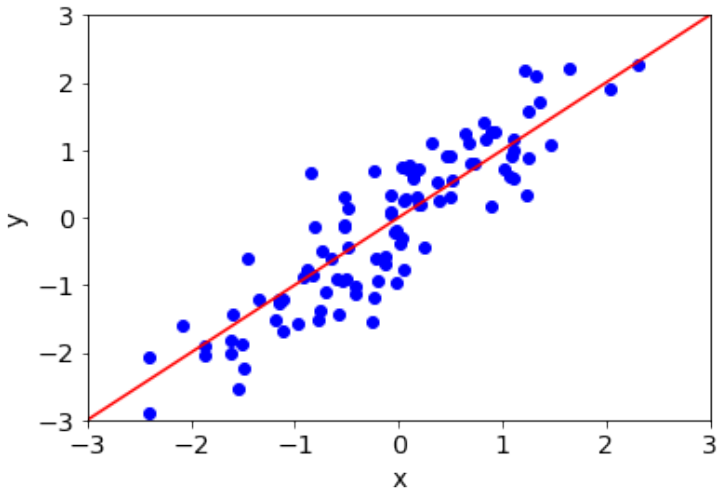
Given a **training set** $(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)}) \in \mathbb{R}^d \times \mathbb{R}$, find a linear function, given by $w \in \mathbb{R}^d$ and $b \in \mathbb{R}$, that minimizes the squared loss

$$L(w, b) = \sum_{i=1}^{n} (y^{(i)} - (w \cdot x^{(i)} + b))^2.$$

Is training loss a good estimate of **future** performance?

- If n is large enough: maybe.
- Otherwise: probably an underestimate.

Example



Better error estimates

Recall: k-**fold cross-validation**

- Divide the data set into k equal-sized groups $S_1, \ldots, S_k$
- For $i = 1$ to k:
  - Train a regressor on all data except S_i
  - Let E_i be its error on S_i
- Error estimate: average of $E_1, \ldots, E_k$

A nagging question:
When n is small, should we be minimizing the squared loss?

$$L(w, b) = \sum_{i=1}^{n} (y^{(i)} - (w \cdot x^{(i)} + b))^2$$

Ridge regression

Minimize squared loss **plus** a term that penalizes "complex" w:

$$L(w, b) = \sum_{i=1}^{n} (y^{(i)} - (w \cdot x^{(i)} + b))^2 + \lambda \|w\|^2$$

Adding a penalty term like this is called **regularization**.

Put predictor vectors in matrix X and responses in vector y:

$$w = (X^T X + \lambda I)^{-1} (X^T y)$$

X : $n \times d$

y : $n \times 1$

w : $d \times 1$

Toy example

Training, test sets of 100 points

- $x \in \mathbb{R}^{100}$, each feature x_i is Gaussian $N(0, 1)$
- $y = x_1 + \cdots + x_{10} + N(0, 1)$

λ	training MSE	test MSE
0.00001	0.00	585.81
0.0001	0.00	564.28
0.001	0.00	404.08
0.01	0.01	83.48
0.1	0.03	19.26
1.0	0.07	7.02
10.0	0.35	2.84
100.0	2.40	5.79
1000.0	8.19	10.97
10000.0	10.83	12.63

The lasso

Popular "shrinkage" estimators:

- **Ridge regression**

$$L(w, b) = \sum_{i=1}^{n} (y^{(i)} - (w \cdot x^{(i)} + b))^2 + \lambda \|w\|_2^2$$

- **Lasso**: tends to produce sparse w

$$L(w, b) = \sum_{i=1}^{n} (y^{(i)} - (w \cdot x^{(i)} + b))^2 + \lambda \|w\|_1$$

Toy example:
Lasso recovers 10 relevant features plus a few more.