Machine Learning: Introduction

Jason Qiang Guo, New York University

Machine Learning: What is it?

- Unsupervised learning: uncover the hidden or latent structure of unlabeled data
  - score rating for political regimes; topic model
- Supervised learning: learning relationships between inputs and a labeled set of outputs
  - Regression is a typical supervised learning; sentiment analysis / opinion mining

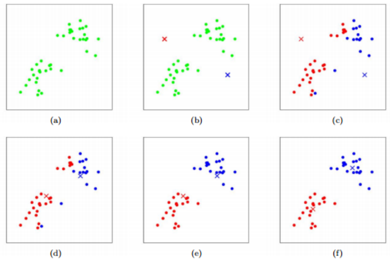
Unsupervised Learning

K-mean Clustering:

Unsupervised Learning

K-mean Clustering:

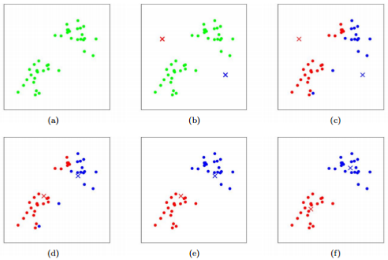
K-Means finds the best centroids by alternating between (1) assigning data points to clusters based on the current centroids (2) chosing centroids (points which are the center of a cluster) based on the current assignment of data points to clusters.



Unsupervised Learning

K-mean Clustering:

K-Means finds the best centroids by alternating between (1) assigning data points to clusters based on the current centroids (2) chosing centroids (points which are the center of a cluster) based on the current assignment of data points to clusters.



1. Initialize **cluster centroids** $\mu_1, \mu_2, \ldots, \mu_k \in \mathbb{R}^n$ randomly.
2. Repeat until convergence: {

   For every i, set
$$c^{(i)} := \arg\min_j ||x^{(i)} - \mu_j||^2.$$
   For each j, set
$$\mu_j := \frac{\sum_{i=1}^m 1\{c^{(i)} = j\} x^{(i)}}{\sum_{i=1}^m 1\{c^{(i)} = j\}}.$$
   }

Unsupervised Learning

Principal Component Analysis:

Unsupervised Learning

Principal Component Analysis:

- Split the total variance of high-dimension data by components.

Unsupervised Learning

Principal Component Analysis:

- Split the total variance of high-dimension data by components.
- Each component is orthogonal to the other and the order of components is determined by the amount of variance a component can explain.

Unsupervised Learning

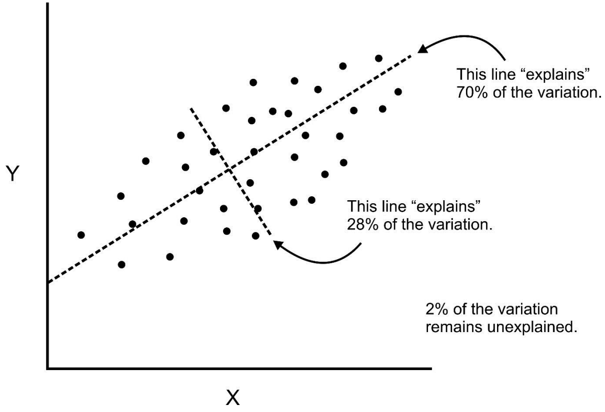
Principal Component Analysis:

- Split the total variance of high-dimension data by components.
- Each component is orthogonal to the other and the order of components is determined by the amount of variance a component can explain.
- For each variable, its importance to a component is determined by how much it contributes to the loadings of a component.

Unsupervised Learning

Principal Component Analysis:

- Split the total variance of high-dimension data by components.
- Each component is orthogonal to the other and the order of components is determined by the amount of variance a component can explain.
- For each variable, its importance to a component is determined by how much it contributes to the loadings of a component.

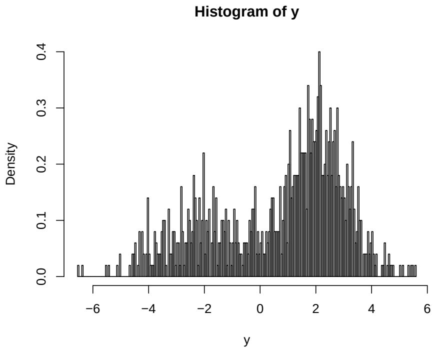


Unsupervised Learning

Mixture Model and EM Algorithm:

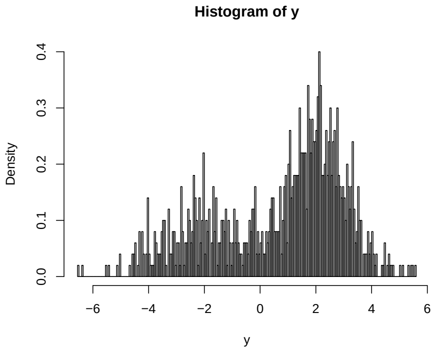
Unsupervised Learning

Mixture Model and EM Algorithm:



Unsupervised Learning

Mixture Model and EM Algorithm:



Two-cluster case: $p(y) = (1-\pi)g_1(y) + \pi g_2(y)$, where $g_1 \sim N(\mu_1, \sigma_1^2)$ and $g_2 \sim N(\mu_2, \sigma_2^2)$

Unsupervised Learning

Mixture Model and EM Algorithm:

Unsupervised Learning

Mixture Model and EM Algorithm:

- MLE? $L(\theta, \pi) = \sum_{i=1}^{N} log[(1-\pi)\phi_{\theta_1}(y) + \pi\phi_{\theta_2}(y)]$
- But the solution to this MLE is not analytically viable.

EM Algorithm:

Unsupervised Learning

Mixture Model and EM Algorithm:

- MLE? $L(\theta, \pi) = \sum_{i=1}^{N} log[(1-\pi)\phi_{\theta_1}(y) + \pi\phi_{\theta_2}(y)]$
- But the solution to this MLE is not analytically viable.

EM Algorithm:

- Give initial values of $\hat{\pi}, \hat{\mu_1}, \hat{\mu_2}, \hat{\sigma_1^2}, \hat{\sigma_2^2}$

Unsupervised Learning

Mixture Model and EM Algorithm:

- MLE? $L(\theta, \pi) = \sum_{i=1}^{N} log[(1-\pi)\phi_{\theta_1}(y) + \pi\phi_{\theta_2}(y)]$
- But the solution to this MLE is not analytically viable.

EM Algorithm:

- Give initial values of $\hat{\pi}, \hat{\mu_1}, \hat{\mu_2}, \hat{\sigma_1^2}, \hat{\sigma_2^2}$
- Expectation: let $\hat{\gamma}_i$ be the probability that ith observation belongs to class 2. Then we have $\hat{\gamma}_i = \frac{\hat{\pi}\phi_{\theta_2}(y_i)}{(1-\hat{\pi})\phi_{\theta_1}(y_i) + \hat{\pi}\phi_{\theta_2}(y_i)}$

Unsupervised Learning

Mixture Model and EM Algorithm:

- MLE? $L(\theta, \pi) = \sum_{i=1}^{N} log[(1-\pi)\phi_{\theta_1}(y) + \pi\phi_{\theta_2}(y)]$
- But the solution to this MLE is not analytically viable.

EM Algorithm:

- Give initial values of $\hat{\pi}, \hat{\mu_1}, \hat{\mu_2}, \hat{\sigma_1^2}, \hat{\sigma_2^2}$
- Expectation: let $\hat{\gamma}_i$ be the probability that ith observation belongs to class 2. Then we have $\hat{\gamma}_i = \frac{\hat{\pi}\phi_{\theta_2}(y_i)}{(1-\hat{\pi})\phi_{\theta_1}(y_i) + \hat{\pi}\phi_{\theta_2}(y_i)}$
- Maximization: Now go back to $L(\theta, \pi) = \sum_{i=1}^{N} log[(1-\pi)\phi_{\theta_1}(y) + \pi\phi_{\theta_2}(y)]$, differentiate it with respect to μ_2 we obtain $\sum_{i=1}^{N} \frac{\hat{\pi}\phi_{\theta_2}(y_i)}{(1-\hat{\pi})\phi_{\theta_1}(y_i) + \hat{\pi}\phi_{\theta_2}(y_i)} \frac{y_i - \mu_2}{\sigma_2^2} = 0$, so $\hat{\mu}_2 = \frac{\sum \hat{\gamma}_i y_i}{\sum \hat{\gamma}_i}$. Derive $\hat{\mu}_1, \hat{\sigma_1^2}$ and $\hat{\sigma_2^2}$ using maximization as well. And we can derive $\hat{\pi} = \sum \hat{\gamma}_i / N$

Unsupervised Learning

Mixture Model and EM Algorithm:

- MLE? $L(\theta, \pi) = \sum_{i=1}^{N} log[(1-\pi)\phi_{\theta_1}(y) + \pi\phi_{\theta_2}(y)]$
- But the solution to this MLE is not analytically viable.

EM Algorithm:

- Give initial values of $\hat{\pi}, \hat{\mu_1}, \hat{\mu_2}, \hat{\sigma_1^2}, \hat{\sigma_2^2}$
- Expectation: let $\hat{\gamma}_i$ be the probability that ith observation belongs to class 2. Then we have $\hat{\gamma}_i = \frac{\hat{\pi}\phi_{\theta_2}(y_i)}{(1-\hat{\pi})\phi_{\theta_1}(y_i) + \hat{\pi}\phi_{\theta_2}(y_i)}$
- Maximization: Now go back to $L(\theta, \pi) = \sum_{i=1}^{N} log[(1-\pi)\phi_{\theta_1}(y) + \pi\phi_{\theta_2}(y)]$, differentiate it with respect to μ_2 we obtain $\sum_{i=1}^{N} \frac{\hat{\pi}\phi_{\theta_2}(y_i)}{(1-\hat{\pi})\phi_{\theta_1}(y_i) + \hat{\pi}\phi_{\theta_2}(y_i)} \frac{y_i - \mu_2}{\sigma_2^2} = 0$, so $\hat{\mu}_2 = \frac{\sum \hat{\gamma}_i y_i}{\sum \hat{\gamma}_i}$. Derive $\hat{\mu}_1, \hat{\sigma_1^2}$ and $\hat{\sigma_2^2}$ using maximization as well. And we can derive $\hat{\pi} = \sum \hat{\gamma}_i / N$
- Iterate E-step and M-step until convergence.

Supervised Learning

Bias-Variance Tradeoff: In supervised learning, for the purpose of prediction, if our goal is to minimize the loss (mean squared error is the most common loss), then to use an unbiased estimator is not always optimal due to the bias-variance tradeoff.

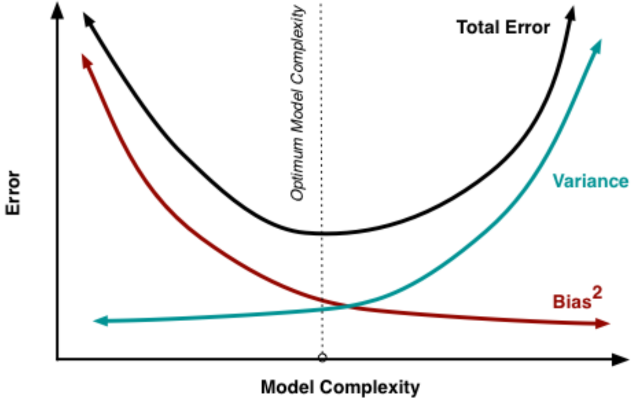
$$\begin{aligned}
\mathbb{E}\left[(\hat{\theta}-\theta^*)^2\right] &= \mathbb{E}\left[\left[(\hat{\theta}-\overline{\theta})+(\overline{\theta}-\theta^*)\right]^2\right] \\
&= \mathbb{E}\left[\left(\hat{\theta}-\overline{\theta}\right)^2\right]+2(\overline{\theta}-\theta^*)\mathbb{E}\left[\hat{\theta}-\overline{\theta}\right]+(\overline{\theta}-\theta^*)^2 \\
&= \mathbb{E}\left[\left(\hat{\theta}-\overline{\theta}\right)^2\right]+(\overline{\theta}-\theta^*)^2 \\
&= \mathrm{var}\left[\hat{\theta}\right]+\mathrm{bias}^2(\hat{\theta})
\end{aligned}$$

In words,

$$\boxed{\text{MSE} = \text{variance} + \text{bias}^2}$$

Supervised Learning

Bias-Variance Tradeoff



Supervised Learning

Lasso and Ridge Regression: Very useful for model selection

- The essence of Lasso and Ridge is shrinkage. Usually we use λ to denote the amount of shrinkage (penalty). In machine learning language we call λ regularization parameter.

Lasso Regression (Tikhonov Form)

The lasso regression solution for regularization parameter $\lambda \geqslant 0$ is

$$\hat{w} = \arg\min_{w \in \mathbf{R}^d} \frac{1}{n} \sum_{i=1}^{n} \left\{ w^T x_i - y_i \right\}^2 + \lambda \|w\|_1,$$

where $\|w\|_1 = |w_1| + \cdots + |w_d|$ is the ℓ_1-norm.

Supervised Learning

Lasso and Ridge Regression: Very useful for model selection

- The essence of Lasso and Ridge is shrinkage. Usually we use λ to denote the amount of shrinkage (penalty). In machine learning language we call λ regularization parameter.

Lasso Regression (Tikhonov Form)

The lasso regression solution for regularization parameter $\lambda \geqslant 0$ is

$$\hat{w} = \arg\min_{w \in \mathbf{R}^d} \frac{1}{n} \sum_{i=1}^{n} \left\{ w^T x_i - y_i \right\}^2 + \lambda \|w\|_1,$$

where $\|w\|_1 = |w_1| + \cdots + |w_d|$ is the ℓ_1-norm.

Ridge Regression (Tikhonov Form)

The ridge regression solution for regularization parameter $\lambda \geqslant 0$ is

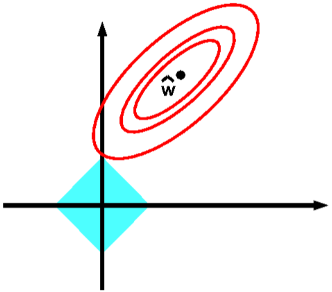
$$\hat{w} = \arg\min_{w \in \mathbf{R}^d} \frac{1}{n} \sum_{i=1}^{n} \left\{ w^T x_i - y_i \right\}^2 + \lambda \|w\|_2^2,$$

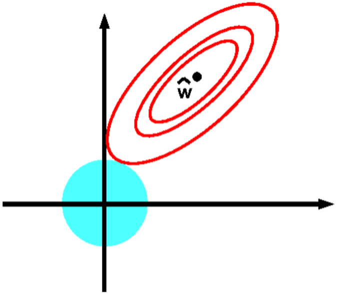
where $\|w\|_2^2 = w_1^2 + \cdots + w_d^2$ is the square of the ℓ_2-norm.

Supervised Learning

Lasso and Ridge Regression: Lasso leads to greater sparsity

- For visualization, restrict to 2-dimensional input space



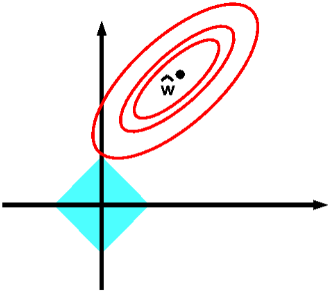


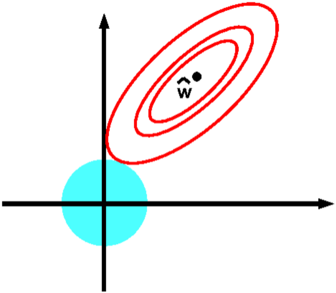
Question: which graph gives a sparse solution?

Supervised Learning

Lasso and Ridge Regression: Lasso leads to greater sparsity

- For visualization, restrict to 2-dimensional input space



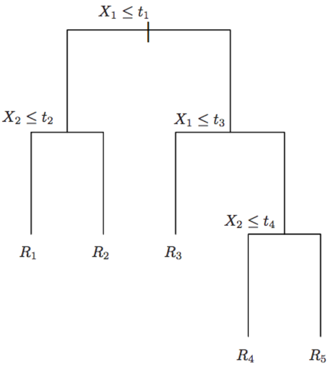


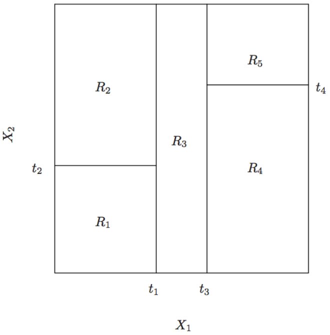
Question: which graph gives a sparse solution?
Answer: The left graph of lasso regression gives a sparse solution.

Supervised Learning

Classification and Regression Trees (CART)

- Consider a binary decision tree on $\{(X_1, X_2)|\}$





Supervised Learning

CART

- Classification Trees

Let node m represent region R_m, with N_m observations

Denote proportion of observations in R_m with class k by

$$\hat{p}_{mk} = \frac{1}{m} \sum_{\{i: x_i \in R_m\}} 1(y_i = k).$$

Predicted classification for node m is

$$k(m) = \arg\max_k \hat{p}_{mk}.$$

Predicted class probability distribution is $(\hat{p}_{m1}, \ldots, \hat{p}_{mK})$.

Supervised Learning

CART

- Regression Trees

Given the partition $\{R_1, \ldots, R_M\}$, final prediction is

$$f(x) = \sum_{m=1}^{M} c_m 1(x \in R_m)$$

How to choose $c_1, \ldots, c_M$?
For loss function $\ell(\hat{y}, y) = (\hat{y} - y)^2$, best is

$$\hat{c}_m = \text{ave}(y_i \mid x_i \in R_m).$$

Supervised Learning

CART

- The number of terminals in a tree measures the complexity of this tree.

Supervised Learning

CART

- The number of terminals in a tree measures the complexity of this tree.
- Trees are easy to interpret, and we can report relative variable importance statistics.

Supervised Learning

CART

- The number of terminals in a tree measures the complexity of this tree.
- Trees are easy to interpret, and we can report relative variable importance statistics.
- However, trees are very unstable, small changes in training set can result in large consequences for classification decisions especially when complexity level is high. This is related to the problems of overfitting in the training set.

Supervised Learning

CART

- The number of terminals in a tree measures the complexity of this tree.
- Trees are easy to interpret, and we can report relative variable importance statistics.
- However, trees are very unstable, small changes in training set can result in large consequences for classification decisions especially when complexity level is high. This is related to the problems of overfitting in the training set.
- What do we do? We can construct many trees using bootstrapped samples and average over them (bootstrap aggregating) or we can combine many trees (forest) and at each split we use a random sample of features (random forest).

Supervised Learning

CART

- The number of terminals in a tree measures the complexity of this tree.
- Trees are easy to interpret, and we can report relative variable importance statistics.
- However, trees are very unstable, small changes in training set can result in large consequences for classification decisions especially when complexity level is high. This is related to the problems of overfitting in the training set.
- What do we do? We can construct many trees using bootstrapped samples and average over them (bootstrap aggregating) or we can combine many trees (forest) and at each split we use a random sample of features (random forest).
- How is decision made in these two kinds of trees? Assign each observation to a final category by a majority vote over the set of trees. Thus, if 51% of the time over a large number of trees a given observation is classified as a "k", that becomes its classification.