

ECE 532 - lecture 27 - Unsupervised learning and K-means

①

Supervised learning: we have training data $\{x_i, y_i\}_{i=1}^N$ that is labeled. Goal is to predict label \tilde{y} for new data \tilde{x} .

★ labels can be continuous; then we're just trying to learn a function $x \rightarrow y$. Example: regression

★ labels can be discrete; then we're performing classification. Examples: LS classification, SVM, kernels, neural networks.

Key: we can evaluate how well we're doing by using cross-validation on testing sets. This helps to prevent over-fitting.

Unsupervised learning: we have data. no labels. the goal is to find hidden structure in the data. i.e. is there a simple representation that approximates the structure of the data?

★ Common problems:

- PCA: find directions (or basis functions) that explain most of the variation in a dataset. We saw how to do this with SVD.
- Clustering: grouping data points so that points in the same group (cluster) are more similar to each other than to those in other groups (clusters).
- Dictionary learning: find a set of vectors (dictionary) that yields a sparse representation for every point in a dataset.
- Topic modeling: find abstract "topics" that occur in a collection of documents.

(2)

- Matrix completion: observe X only on a subset of entries, but suppose full X is rank $r < \min(m, n)$. Find factorization $D \cdot W$ such that $X \approx DW$ at observed entries.

All of these example problems are essentially about learning a "good representation" for a data set.

Common perspective

Let $X \in \mathbb{R}^{n \times m}$ be a data set of m examples (unlabeled), each represented as $x_i \in \mathbb{R}^n$, $i=1, \dots, m$.

All of the above problems (PCA, clustering, dict. learning, topic modeling, and matrix completion), can be posed as finding a matrix factorization so that

$$X \approx DW$$

↑ ↑
dictionary weights.

we'll now take a look at each problem individually to understand why this is the case.

PCA : if our data is x_1, \dots, x_m then

$X = U \Sigma V^T$. choose first k singular values.
($n \times m$) ($n \times r$) ($r \times r$) ($r \times m$)

$\Rightarrow X \approx \hat{U} \hat{\Sigma} \hat{V}^T$. This matrix factorization can be written (using $D = \hat{U}$, $W = \hat{\Sigma} \hat{V}^T$) as:
($n \times k$) ($k \times k$) ($k \times m$)

$\Rightarrow X \approx DW$ \Rightarrow this is the solution to
($n \times k$) ($k \times m$) . $\min_{D,W} \|X - DW\|$ s.t. $\text{rank}(DW) = k$.

Columns of D are k most significant directions ($\approx R(X)$) and columns of W give the representation of each x_i using the directions D . i.e. $x_i = D w_i$.

Clustering if we want to cluster x_1, \dots, x_m into k clusters.

$X \approx DW$. $\Leftrightarrow x_i \approx D w_i$
 \uparrow
 $[d_1 \ d_2 \ \dots \ d_k]$

Here, each w_i should be of the form: $w_i = \begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix}$ "1" in exactly one location, corresponding to best d_i .

Trivial solution: each point is a cluster! $D = X$, $W = I$.

obviously, we want $k \ll m$ so we'll be seeking an approximate factorization.

Dictionary learning

$$X \approx D \cdot W$$

($n \times m$) ($n \times k$) ($k \times m$)

again, $X = [x_1, \dots, x_m]$ is data.
 e.g. each x_i could be an image.

↪ $x_i \approx D w_i \quad i=1, \dots, m.$

Goal is to find D such that W is sparse.

Typically, $n < k \ll m$

★ find D such that W is sparse enough.

"overcomplete". i.e. any x is representable

↪ lots of data!

Topic modeling

$$X \approx D W$$

↑
 x_i is a document
 e.g. word counts.

↙ w_i is sparse, indicating which topics are in doc i .
 ↖ d_i is a "topic", i.e. certain combinations of word occurrences

Typically, we want D to be sparse, W to be very sparse.

Matrix completion (missing data).

minimize $\|X - DW\|_F^2$

$D \in \mathbb{R}^{n \times r}$
 $W \in \mathbb{R}^{r \times m}$

↪ only at locations where X is observed.

where r is chosen in advance.

A common strategy

(5)

In all of these problems, we seek a factorization of our data matrix. $X \approx DW$.

We can pose this as an optimization problem:

$$\min_{D, W} \|X - DW\| \quad (\text{for some choice of norm}).$$

This optimization problem is not jointly convex in D, W , (since we have a product of d_i 's, w_{ij} 's).

So this is a difficult problem in general. There may be many local minima. However, the problem is "biconvex", i.e. if we fix D , it's convex in W and if we fix W it's convex in D . This suggests the strategy:

- 0) initial guess D_0, W_0
 - 1) hold D_i fixed, optimize over $W \rightarrow W_{i+1}$
 - 2) hold W_{i+1} fixed, optimize over $D \rightarrow D_{i+1}$
- repeat 1 and 2 to convergence.

Generally this does not converge to a global optimum, but it works surprisingly well in practice. (choice of D_0, W_0 is important!)

K-means clustering

(6)

Recall $X \approx DW$

$[x_1 \dots x_m]$ data points

$[d_1 \dots d_k]$ K centers.

$[w_1 \dots w_m]$ indicator vectors that select a center for each data point.

(1) if D is fixed, what is the solution to

$$\min_{W \text{ has indicator columns}} \|X - DW\|_F^2 \quad ?$$

$$\begin{aligned} \|X - DW\|_F^2 &= \sum_{i=1}^m \|x_i - Dw_i\|^2 \\ &= \sum_{i=1}^m \min_{1 \leq j \leq k} \|x_i - d_j\|^2 \end{aligned}$$

therefore, $w_i^* = \begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix}$
where 1 is in position $j = \operatorname{argmin}_j \|x_i - d_j\|^2$.

(2) suppose cluster memberships W are known, what is the solution to $\min_D \|X - DW\|_F^2$?

solution is simply $D^* = XW^T$ (least squares).

$$\text{or: } \|X - DW\|_F^2 = \sum_{i=1}^m \|x_i - Dw_i\|^2 = \sum_{i=1}^m \sum_{j: w_{ij}=1} \|x_i - d_j\|^2 = \sum_{j=1}^k \underbrace{\sum_{i: w_{ij}=1} \|x_i - d_j\|^2}_{\text{dist squared of all pts. in a cluster to the cluster center.}}$$

(ref: practice exam!) minimized when $d_j = \frac{1}{n_j} \sum_{i: w_{ij}=1} x_i$

(mean of the points.)

K-means algorithm (Lloyd's algorithm)

(7)

↖ Stuart Lloyd, 1957.

0) start with initial means / centers.

$$d_1^{(0)}, d_2^{(0)}, \dots, d_k^{(0)}.$$

1) assignment:

for each x_i , find closest mean d_j .

$$j_i = \arg \min_j \|x_i - d_j^{(l)}\|^2.$$

$$\text{cluster: } C_j^{(l)} = \left\{ i : \|x_i - d_j^{(l)}\|^2 \leq \|x_i - d_{j'}^{(l)}\|^2 \text{ for all } j' \neq j \right\}.$$

2) update means

$$d_j^{(l+1)} = \frac{1}{|C_j^{(l)}|} \sum_{i \in C_j^{(l)}} x_i$$

(mean of points in each cluster).

then repeat to convergence.

K-means ++

8

★ K-means algorithm is not guaranteed to find a global minimum of $\|X - DW\|_F^2$ with W indicators.

★ solving this problem exactly is hard. would require searching over many clusterings. (impractical for any interesting problem).

★ some methods use approximations to guarantee something.

one such method is "k-means ++". It's a smarter way to initialize k-means.

1) choose one $p_1 \in \{x_1, \dots, x_n\}$ at random. This is d_1 .

2) compute distance to each point + center so far:

$$\delta(x_i) = \min_j \|x_i - d_j\|^2.$$

3) choose next center d_2 at random according to probability

$$P_i \propto \delta(x_i)^2. \quad (\text{likelier to pick pt. farther away.})$$

4) repeat 2, 3 until all centers have been chosen.

5) run k-means using this initialization.

★ fast in practice. If \hat{D}, \hat{W} are optimal soln, \nearrow

★ guarantee: $\mathbb{E}[\|X - \hat{D}\hat{W}\|_F^2] \leq 8(\log k + 2) \|X - D^{\text{opt}} W^{\text{opt}}\|_F^2.$