Dimensionality Reduction.

Principal Component Analysis (PCA) problem formulation.

Reduce from n-dimension to k-dimension: find k vectors $u^{(1)}, u^{(2)}, \ldots, u^{(k)}$ onto which to project the data, so as to minimize the projection error.

Data preprocessing:

1. Training set: $x^{(1)}, x^{(2)}, \cdots, x^{(m)}$.
2. preprocessing (feature scaling / mean normalization).

   $$u_j = \frac{1}{m} \cdot \sum_{i=1}^{m} x_j^{(i)}$$

   Replace each $x_j^{(i)}$ with $x_j - u_j$.

   If different features on different scales, scale features to have comparable range of values.

3. PCA algorithm: Reduce data from n-dimension to k-dimension.

   i). Compute "covariance matrix":

   $$\boxed{\Sigma = \frac{1}{m} \cdot \underset{n \times m}{X^T} \cdot \underset{m \times n}{X}}$$

   $$\Sigma = \frac{1}{m} \cdot \sum_{i=1}^{n} \underbrace{(x^{(i)})}_{n \times 1} \underbrace{(x^{(i)})^T}_{1 \times n} = \text{sigma}.$$

   ii) Compute "eigenvectors" of matrix Σ:

   $u: n \times n$.

   $\underline{[U, S, V] = \text{svd}(\text{sigma})}$. // singular value decomposition.

   $$U \cdot S \cdot V^T = \text{sigma}.$$

   iii). To reduce $x \in R^n$ to $z \in R^k$, we do:

   $$U = \Big[\underbrace{u^{(1)} \cdots}_{k} u^{(n)}\Big]_{n \times n}. \Rightarrow z^{(i)} = \underbrace{\Big[u^{(1)}\ u^{(2)} \cdots u^{(k)}\Big]}_{U_{reduce} = n \times k = U(:, 1:k)}^T \cdot x^{(i)} = \underset{k \times n}{\begin{bmatrix} \cdots u^{(1)T} \cdots \\ \cdots u^{(2)T} \cdots \\ \vdots \\ \cdots u^{(k)T} \cdots \end{bmatrix}} \cdot \underset{n \times 1}{x^{(i)}}$$

   Now, z is $k \times 1$ dimensional.

Reconstruction from compressed representation.

$$z = U_{reduce}^T \cdot x \Rightarrow x_{approximate} = U_{reduce} \cdot z$$

z: $k \times 1$, U_{reduce}^T: $k \times n$, x: $n \times 1$; $x_{approximate}$: $n \times 1$, U_{reduce}: $n \times k$, z: $k \times 1$.

choosing k (number of principal components).

i). Average squared projection error: $\frac{1}{m}\sum_{i=1}^{m} \|x^{(i)} - x_{approx}^{(i)}\|^2$.

ii). Total variation in the data: $\frac{1}{m} \cdot \sum_{i=1}^{m} \|x^{(i)}\|^2$.

Typically, choose k to be smallest value so that:

$$\frac{\frac{1}{m} \cdot \sum_{i=1}^{m} \|x^{(i)} - x_{approx}^{(i)}\|^2}{\frac{1}{m}\sum_{i=1}^{m}\|x^{(i)}\|^2} \le 0.01$$ (99% variance is retained)

choosing k algorithm:

1. try PCA with $k=1$.
2. compute U_{reduce}, $z^{(1)}, z^{(2)}, \cdots, z^{(m)}, x_{approx}^{(1)}, \cdots, x_{approx}^{(m)}$)
3. check if $\frac{\frac{1}{m}\sum_{i=1}^{m} \|x^{(i)} - x_{approx}^{(i)}\|^2}{\frac{1}{m}\sum_{i=1}^{m}\|x^{(i)}\|^2} \le 0.01$.

otherwise,

k++.

Shortcut: $[U, S, V] = svd(sigma)$.

$$S = \begin{bmatrix} s_{11} & & & \\ & s_{22} & & \\ & & \ddots & \\ & & & s_{nn} \end{bmatrix}$$

for given k, $$\frac{\frac{1}{m}\sum_{i=1}^{m} \|x^{(i)} - x_{approx}^{(i)}\|^2}{\frac{1}{m} \cdot \sum_{i=1}^{m}\|x^{(i)}\|^2} = 1 - \frac{\sum_{i=1}^{k} S_{ii}}{\sum_{i=1}^{n} S_{ii}}$$

Supervised learning Speedup.

Mapping $x^{(i)} \rightarrow z^{(i)}$ should be defined by running PCA only on the training set. this mapping can be applied as well to the examples $x_{cv}^{(i)}$ and $x_{test}^{(i)}$ in the cross validation and test sets.

Application of PCA:

i). Compression.

- Reduce memory/disk needed to store data.
- Speed up learning algorithm.

ii). Visualization.

$k=2$ or $k=3$.

$\boxed{\text{Bad}}$ use of PCA: to prevent overfitting ×

×

use regularization instead ✓