

Principle Component Analysis (PCA) Algorithm

Reference

<https://www.cs.cmu.edu/~elaw/papers/pca.pdf>

```
@article{shlens2014tutorial,  
  title={A tutorial on principal component analysis},  
  author={Shlens, Jonathon},  
  journal={arXiv preprint arXiv:1404.1100},  
  year={2014}  
}
```

Algorithm Formulation (based on covariance method)

Define the dataset $X \in \mathbb{R}^{n \times m}$, where n is the number of samples, and m is the number of dimensions. Define $Z \in \mathbb{R}^{n \times k}$ as the PCA transformed dataset, where $k < m$ is the new dimension. In practice, $k \ll m$.

We first calculate the mean of each dimension on the dataset: $\mu_j = \frac{\sum_{i=0}^{N-1} X_{i,j}}{N}$. Then, subtract off the mean for each dimension: $\hat{X}_{i,j} = X_{i,j} - \mu_j$.

Next, we calculate the covariance matrix of \hat{X} , denoted by $C \in \mathbb{R}^{m \times m}$.

$$C = \begin{bmatrix} \text{cov}(X_{:,0}, X_{:,0}) & \text{cov}(X_{:,0}, X_{:,1}) & \dots & \text{cov}(X_{:,0}, X_{:,m}) \\ \text{cov}(X_{:,1}, X_{:,0}) & \text{cov}(X_{:,1}, X_{:,1}) & \dots & \text{cov}(X_{:,1}, X_{:,m}) \\ \dots & \dots & \dots & \dots \\ \text{cov}(X_{:,m}, X_{:,0}) & \text{cov}(X_{:,m}, X_{:,1}) & \dots & \text{cov}(X_{:,m}, X_{:,m}) \end{bmatrix}$$

In fact (see the diagonal of C), $\text{cov}(X_{:,j}, X_{:,j}) \equiv \text{variance}(X_{:,j})$.

Rule of Thumb:

Large variance indicates the corresponding feature (dimension) has more interesting dynamics; small variance usually indicates noise.

Large covariance between two features indicates high redundancy (naively, we might want to keep only one of these features).

The computation of covariance matrix can be easily implemented by the following formula:

$$C = \frac{1}{(n-1)\hat{X}^T \hat{X}}$$

Then, find the eigenvectors ($V \in \mathbb{R}^{m \times m}$) and eigenvalues ($\lambda \in \mathbb{R}^m$) of C . We sort λ in descending order, and rearrange V correspondingly. We only keep k columns of V , and define that matrix as $P \in \mathbb{R}^{m \times k}$.

Finally, we can project the original dataset:

$$Z = XP.$$

