Support vector machines unit:

- What PCA is (last week Monday)
- Applications of PCA (last week Wednesday, in lab)
- The math of PCA (last week Friday)
- PCA algorithms (**Today**)
- (Begin neural nets on Wednesday)

Today:

- Finish the math of PCA, maximum-variance view
- Point of comparison: minimum-information-loss view
- Practical parts for implementation

The most important source for all of this was Deisenroth et al, *Mathematics for Machine Learning*, 2020, 286–293.

(Available on Canvas)

Let $\bar{\boldsymbol{x}} = \mathbb{E}[\boldsymbol{x}] = \frac{1}{N}\sum_{n=1}^{N} \boldsymbol{x_n}$ be the mean of the data. We *center* the data so that it has mean $\mathbf{0}$: $\boldsymbol{x_n}^C = \boldsymbol{x_n} - \bar{\boldsymbol{x}}$. From this point on, we assume X has been centered, and so $\mathbb{E}[\boldsymbol{x}] = 0$.

Let $\mathbf{W} = \begin{pmatrix} \boldsymbol{v_0} \\ \vdots \\ \boldsymbol{v_{M-1}} \end{pmatrix} \in \mathbb{R}^{M \times D}$ be a set of orthonormal basis vectors.

Let $\boldsymbol{z_n} = \mathbf{W}\boldsymbol{x_n}$ be the projection of $\boldsymbol{x_n}$ in the vector space defined by these basis vectors.

We can project $\boldsymbol{z_n}$ back ito the original space with $\tilde{\boldsymbol{x}}_{\boldsymbol{n}} = \mathbf{W}^T\boldsymbol{z_n}$.

Let $\boldsymbol{v}_0 \in \mathbb{R}^D$ be the zeroth ("first") principal component (which we want to find).

Let $z_{0,n} = \boldsymbol{v_0}^T \boldsymbol{x_n}$ be the zeroth coordinate in the projection of $\boldsymbol{x_n}$.

Let z_0 be a random variable modeling the value of the zeroth coordinate. Assume data is centered, that is, $\mathbb{E}[z_0] = 0$.

Let $\mathbf{C} = \frac{1}{N} \sum_{n=0}^{N-1} \boldsymbol{x_n} \boldsymbol{x_n}^T$ be the data covariance matrix. Then

$$
\begin{aligned}
\mathrm{Var}[z_0] &= \frac{1}{N} \sum_{n=0}^{N-1} (z_{0,n} - 0)^2 \\
&= \frac{1}{N} \sum_{n=0}^{N-1} (\boldsymbol{v_0}^T \boldsymbol{x_n})^2 \\
&= \frac{1}{N} \sum_{n=0}^{N-1} (\boldsymbol{v_0}^T \boldsymbol{x_n} \boldsymbol{x_n}^T \boldsymbol{v_0})^2 \\
&= \boldsymbol{v_0}^T \left(\frac{1}{N} \sum_{n=0}^{N-1} \boldsymbol{x_n} \boldsymbol{x_n}^T \right) \boldsymbol{v_0} \\
&= \boldsymbol{v_0}^T \mathbf{C} \boldsymbol{v_0} \qquad \text{which we want to maximize subject to } ||\boldsymbol{v_0}||^2 = 1
\end{aligned}
$$

Optimization problem as a Lagrangian:

$$\mathcal{L}(\boldsymbol{v_0}, \lambda_0) = \boldsymbol{v_0}\mathbf{C}\boldsymbol{v_0} + \lambda_0(1 - \boldsymbol{v_0}^T\boldsymbol{v_0})$$

$$\frac{\partial \mathcal{L}}{\partial \lambda_0} = 1 - \boldsymbol{v_0}^T\boldsymbol{v_0}$$

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \boldsymbol{v_0}} = 2\boldsymbol{v_0}^T C - 2\lambda_0\boldsymbol{v_0}^T &= 0 \\ \boldsymbol{v_0}^T C &= \lambda_0\boldsymbol{v_0}^T \end{aligned}$$

Plug that into our formula for the variance:

$$\begin{aligned} \mathrm{Var}[z_0] &= \boldsymbol{v_0}^T\mathbf{C}\boldsymbol{v_0} \\ &= \lambda_0\boldsymbol{v_0}^T\boldsymbol{v_0} = \lambda_0 \end{aligned}$$

To find the next principal component, transform the data by removing the effect of the principal component $\boldsymbol{v_0}$:

$$\mathbf{X_1} = \mathbf{X} - \boldsymbol{v_0}\boldsymbol{v_0}^T\mathbf{X}$$

…compute the corresponding data covariance matrix $\mathbf{C_1}$, find the eigenvector with greatest eigenvalue. As a loop to find M principal components $\boldsymbol{v_0}, \ldots \boldsymbol{v_{M-1}}$:

$\mathbf{C}$ = the data covariance matrix of $\mathbf{X}$
$\boldsymbol{v_0}$ = the eigenvector of $\mathbf{C}$ with greatest eigenvalue
for $m \in [1, M)$:

$\mathbf{W_m} = \sum_{i=0}^{m-1} \boldsymbol{v_i}\boldsymbol{v_i}^T$ (projection matrix into subspace)
$\mathbf{X_m} = \mathbf{X} - \mathbf{W_m}\mathbf{X}$
$\mathbf{C_m}$ = the data covariance matrix of $\mathbf{X_m}$
$\boldsymbol{v_m}$ = the eigenvector of $\mathbf{C_m}$ with greatest eigenvalue

Theorem 1 (Invariant)

For all m, every eigenvector of C is an eigenvector of C_m.

For all m, every eigenvector of C is an eigenvector of C_m.

Proof. Suppose $\boldsymbol{v_j}$ is an eigenvector of C with eigenvalue λ_j, that is, $C\boldsymbol{v_j} = \lambda_j \boldsymbol{v_j}$. Then

$$
\begin{aligned}
\mathbf{C_m}\boldsymbol{v_j} &= \frac{1}{N}\mathbf{X_m}\mathbf{X_m}^T \boldsymbol{v}_j \\
&= \frac{1}{N}(\mathbf{X} - \mathbf{W_m X})(\mathbf{X} - \mathbf{W_m X})^T \boldsymbol{v}_j && \text{do FOIL with } \frac{1}{N}\mathbf{X}\mathbf{X}^T = \mathbf{C} \\
&= (\mathbf{C} - \mathbf{C W_m} - \mathbf{W_m C} + \mathbf{W_m C W_m})\boldsymbol{v}_j \\
&= \begin{cases} \mathbf{C}\boldsymbol{v_j} = \lambda_j \boldsymbol{v_j} & \text{if } j \geq m \\ \mathbf{C}\boldsymbol{v_j} - \mathbf{C}\boldsymbol{v_j} - \mathbf{C}\boldsymbol{v_j} + \mathbf{C}\boldsymbol{v_j} = 0 & \text{if } j < m \end{cases}
\end{aligned}
$$

In the $j \geq m$ case, $\boldsymbol{v_j}$ is orthogonal to all the vectors in $\mathbf{W_m}$, so $\mathbf{W_m}\boldsymbol{v_j} = 0$.
In the $j < m$ case, $\boldsymbol{v_j}$ is a basis vector of the subspace into which $\mathbf{W_m}$ projects, so $\mathbf{W_m}\boldsymbol{v_j} = \boldsymbol{v_j}$. □

To compute the principal components:

Given **data** and M (number of desired components),

Center the data (and store the mean $\bar{\boldsymbol{x}}$)

Compute the covariance matrix

Compute the eigenvectors and corresponding eigenvalues

Sort the eigenvectors by eigenvalues

Return the M eigenvectors with greatest eigenvalues

To transform a data point using principal components:

Given data point $\boldsymbol{x}$ and principal components $\boldsymbol{v_0}, \ldots \boldsymbol{v_{M-1}}$,

Shift the data point based on the centering, $\hat{\boldsymbol{x}} = \boldsymbol{x} - \bar{\boldsymbol{x}}$

Compute the dot products of $\hat{\boldsymbol{x}}$ and each principal component

Assemble the results as a new vector, and return it

Coming up:

Due Fri, Mar 28:
Textbook reading from Chapter 10 (see Canvas)

Due Mon, Mar 31:
Take PCA quiz

Due Fri, Apr 4:
Implement PCA

Due Wed, Apr 9:
Read and respond to two articles about bias in algorithms (See Canvas)

Sometime between Mar 31 and Apr 17:
Make an office-hours appointment for project check-in (Originally the deadline was Apr 11)