University of Pittsburgh | School of Public Health

Dimension reduction
- Principal Component Analysis
- Independent Component Analysis*

BIOST 2155

Class 9
October 30th, 2026

CLASS 1

Mapping what we have learnt so far…

	Features (input)	Targets/ labels (output)	Task type	ML type	Metric
Prediction of brain volume from gestational age	✅	✅ (Numeric)	Regression	**Supervised learning**	*RMSE, R-squared*
Classification of heart failure using ultrasound or magnetic resonance imaging	✅	✅ (Categorical)	Classification		*Accuracy, Precision, Recall, AUC*
Patient health profiles using wearable device data	✅	🚫	Clustering	**Unsupervised learning**	*Silhouette score, Gap statistic*
Compressing information from 30 features to study similarities/differences in benign and malignant breast tumors	✅	🚫	Dimensionality reduction		*Explained Variance*

CLASS 1

4. Detecting breast cancer from biopsy of cell nuclei

Use ML to visualize high-dimensional tumor data to identify separable groups.

Context: Breast cancer diagnosis relies on analyzing over 30 quantitative features from tumor images, such as cell size, shape, and texture.
Challenge: Manually analyzing 30+ interacting features for every patient is impossible. We cannot "see" the data in 30 dimensions to spot hidden patterns.

By applying PCA to a dataset of tumor measurements, we can distill complex tumor-related information into a simple 2D map.

Original Features: Measurements like mean radius, mean texture, and mean smoothness describe the tumor.

Principal Components (PCs): PCA combines the original 30+ features into two new, powerful summary variables that capture the most important information.

Why do we care?

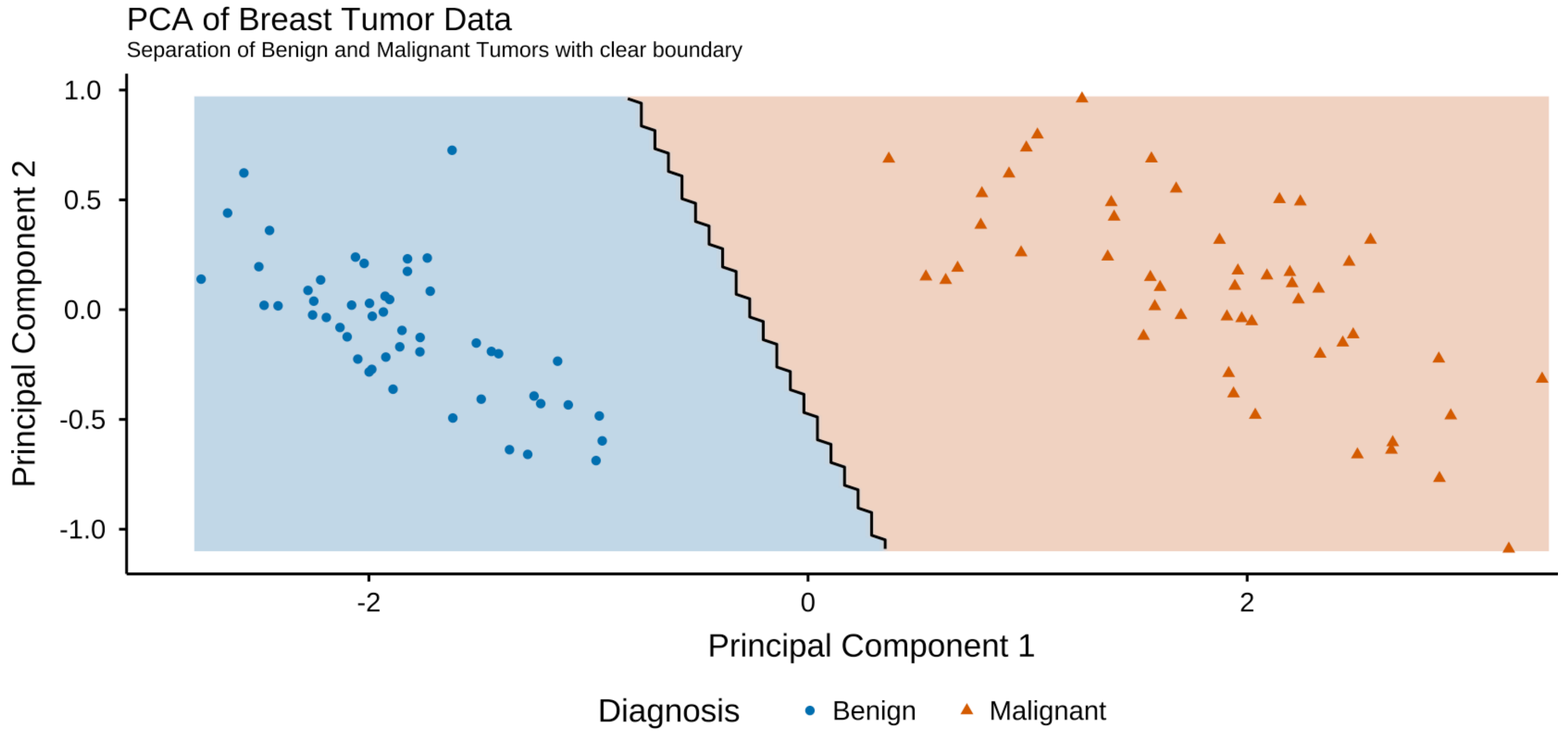
Rapid Pattern Discovery.
Confirms Model Feasibility.
Data-Driven Insights.

CLASS 1

4. Detecting breast cancer from biopsy of cell nuclei

	Benign (N=50)	Malignant (N=50)	Overall (N=100)
radius			
Mean (SD)	9.99 (0.935)	20.1 (1.69)	15.0 (5.26)
Median [Min, Max]	9.99 [7.97, 11.9]	20.1 [16.5, 23.6]	14.2 [7.97, 23.6]
texture			
Mean (SD)	15.2 (1.94)	22.2 (2.84)	18.7 (4.27)
Median [Min, Max]	14.8 [11.1, 19.3]	22.3 [16.8, 29.2]	18.2 [11.1, 29.2]
perimeter			
Mean (SD)	59.8 (4.74)	120 (8.61)	89.9 (31.1)
Median [Min, Max]	59.8 [50.4, 68.9]	121 [98.9, 138]	83.9 [50.4, 138]
area			
Mean (SD)	303 (92.2)	999 (125)	651 (366)
Median [Min, Max]	308 [117, 513]	1020 [740, 1250]	626 [117, 1250]

→



PCA can take a complex dataset with many measurements and distill it into a simple, powerful visualization.

We see a clear underlying pattern that separates **benign** and **malignant** tumors, confirming that the data is highly suitable for building an accurate predictive machine learning model.

On data visualization for $p > 3$

PCA as a tool for EDA and pre-processing (feature engineering)

Suppose we wish to visualize n observations on a set of p features $X_1, X_2, \ldots, X_p$. We could do this via all pairwise scatterplots, but we'd be looking at $\approx p^2$ plots.

We still want to have an informative way to visualize variation in data.
Can we discover subgroup patterns among the features or observations?

Principal components are a good way of doing both of those things!

1. Finds a low-dimensional representation of data that contains maximum possible variation.
2. Although observations lives in $\mathbb{R}^p$, but not all of these dimensions are **equally interesting**.
3. PCA seeks those dimensions that are as **interesting** as possible, where the concept of interesting is measured by the amount that the observations vary along each dimension.
4. Each of the dimensions found by PCA is a **linear combination** of the p features.

Principal Component Analysis

Overview

1. We will first assume that the set of features $X_1, X_2, \ldots, X_p$ have mean zero.
2. The first principal component Z_1 of a is the **normalized linear combination** of the features:

$$Z_1 = \phi_{11}X_1 + \phi_{21}X_2 + \ldots + \phi_{p1}X_p$$

3. $\phi_{11}, \phi_{21}, \ldots, \phi_{p1}$ are the **loadings** of Z_1
4. Normalized linear combinations = **loadings** satisfy $\sum_{j=1}^{p} \phi_{j1}^2 = 1$
5. How to specify $\phi_{11}, \phi_{21}, \ldots, \phi_{p1}$? Maximize sample variance of Z_1. Since $X_1, X_2, \ldots, X_p$ have mean zero, so does Z_1.

$$\max_{\phi_{11}, \ldots \phi_{p1}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{p} \phi_{j1} x_{ji} \right)^2 \right\} \text{ subject to } \sum_{j=1}^{p} \phi_{j1}^2 = 1$$

Can be solved via eigen decomposition!!

Moving beyond the first PC Z_1

Adding a constraint for second PC Z_2

1. Second principal component Z_2 is the linear combination of $X_1, X_2, \ldots, X_p$ that has **maximal variance** among all linear combinations that are uncorrelated with Z_1.
2. Z_2 of a is the **normalized linear combination** of the features:

$$Z_2 = \phi_{12}X_1 + \phi_{22}X_2 + \ldots + \phi_{p2}X_p$$

3. $\phi_{21}, \phi_{22}, \ldots, \phi_{p2}$ are the **loadings** of Z_2
4. Normalized linear combinations = **loadings** satisfy $\sum_{j=1}^{p} \phi_{j2}^2 = 1$
5. How to specify $\phi_{21}, \phi_{22}, \ldots, \phi_{p2}$?

$$\max_{\phi_{12}, \ldots \phi_{p2}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{p} \phi_{j2} x_{ji} \right)^2 \right\} \text{ subject to } \sum_{j=1}^{p} \phi_{j2}^2 = 1 \text{ and } \sum_{j=1}^{p} \phi_{j1}\phi_{j2} = 0$$

Can be solved via eigen decomposition!!

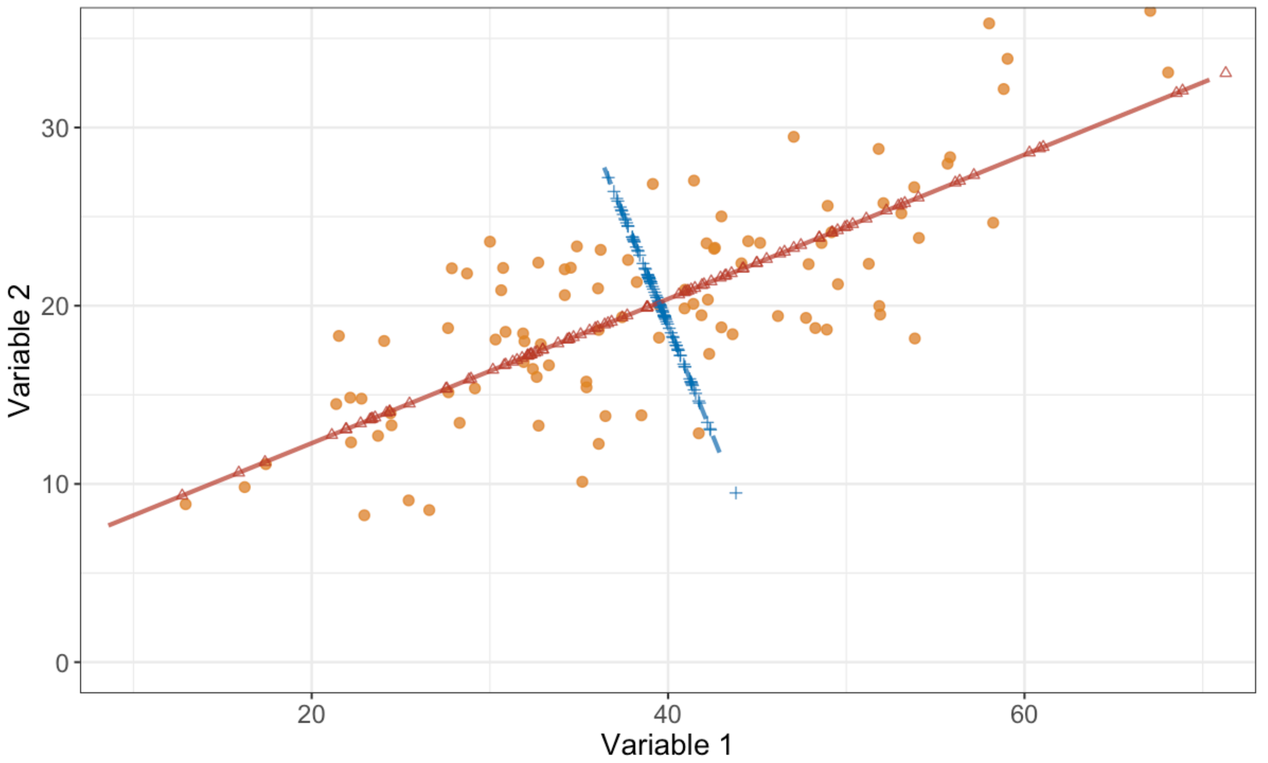
Each PC must be uncorrelated with the others.

The geometry of PCA

Looking for maximal variance

- Loadings $\phi_{11}, \phi_{21},\ldots,\phi_{p1}$ define a direction (obtained using linear combination of $X_1, X_2, \ldots, X_p$) in feature space along which the data **vary the most.**
- If we projected (*looked at the linear combination* $\phi_{12}X_1 + \phi_{22}X_2 + \ldots + \phi_{p2}X_p$) the n data points onto the direction that is specified by $\phi_{11}, \phi_{21},\ldots,\phi_{p1}$, the projections are the principal component scores themselves.

Assigned reading: ISLR2 pg 499 - 501



Cloud of simulated data points (orange dots) and its two PC directions, which represent the axes of greatest variance (PC1: red) and second-greatest variance (PC2, blue).

The red triangles are projections of each original orange dot onto the PC1 line. The blue crosses are projections of each original orange dot onto the PC2 line

We illustrate the use of PCA on the `USArrests` data set. For each of the 50 states in the United States, the data set contains the number of arrests per 100,000 residents for each of three crimes: `Assault`, `Murder`, and `Rape`. We also record `UrbanPop` (the percent of the population in each state living in urban areas). The principal component score vectors have length $n = 50$, and the principal component loading vectors have length $p = 4$. PCA was performed after standardizing each variable to have mean zero and standard deviation one. Figure 12.1 plots the first two principal components of these data. The figure represents both the principal component scores and the loading vectors in a single *biplot* display. The loadings are also given in Table 12.2.1.

	PC1	PC2
`Murder`	0.5358995	−0.4181809
`Assault`	0.5831836	−0.1879856
`UrbanPop`	0.2781909	0.8728062
`Rape`	0.5434321	0.1673186

TABLE 12.1. *The principal component loading vectors, ϕ_1 and ϕ_2, for the* `USArrests` *data. These are also displayed in Figure 12.1.*

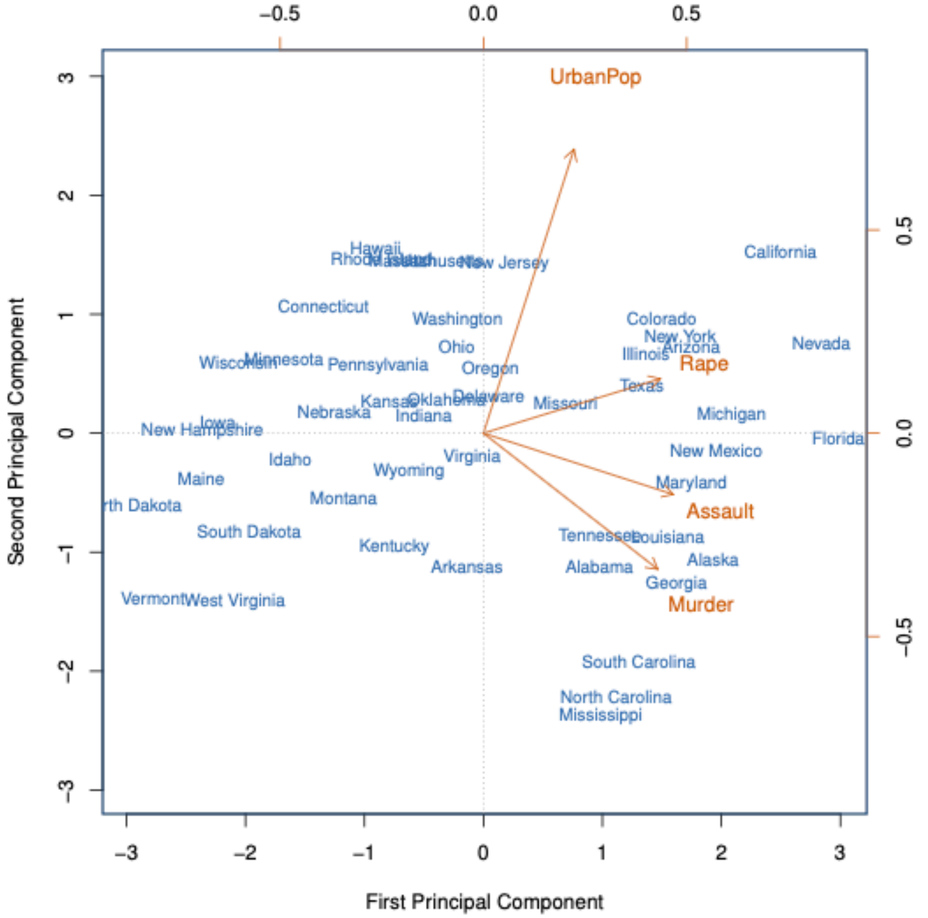


FIGURE 12.1. *The first two principal components for the* `USArrests` *data. The blue state names represent the scores for the first two principal components. The orange arrows indicate the first two principal component loading vectors (with axes on the top and right). For example, the loading for* `Rape` *on the first component is* 0.54*, and its loading on the second principal component* 0.17 *(the word* `Rape` *is centered at the point* $(0.54, 0.17)$*). This figure is known as a biplot, because it displays both the principal component scores and the principal component loadings.*

Assigned reading: ISLR2 pg 499 - 501

Proportion of variance explained

Scree plots

- Interested in knowing the proportion of variance explained (PVE) by each PC.
- Total variance present in the data:

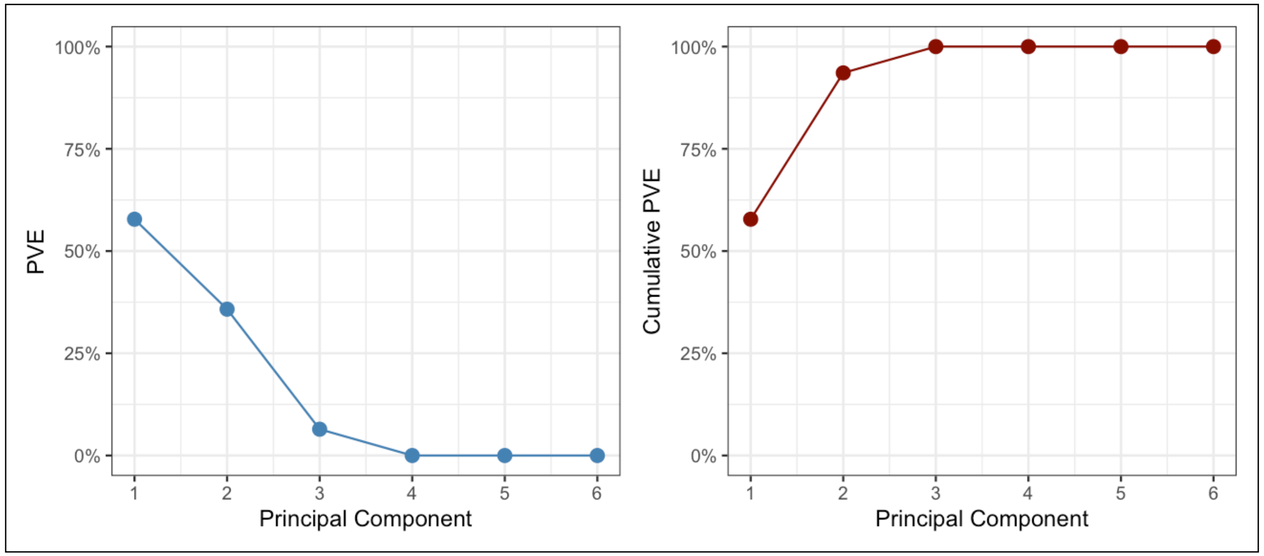
$$\sum_{j=1}^{p} \mathbb{V}(X_j) = \sum_{j=1}^{p} n^{-1} \sum_{i=1}^{n} x_{ij}^2$$

- Variance explained by the m-th principal component Z_m:

$$\mathbb{V}(Z_m) = n^{-1} \sum_{i=1}^{n} z_{im}^2$$

$$\sum_{j=1}^{p} \mathbb{V}(X_j) = \sum_{m=1}^{M} \mathbb{V}(Z_m) \text{ where } M = \min(n-1, p)$$

$$PVE(Z_m) = \frac{\mathbb{V}(Z_m)}{\sum_{j=1}^{p} \mathbb{V}(X_j)} = \frac{\sum_{i=1}^{n} z_{im}^2}{\sum_{j=1}^{p} \sum_{i=1}^{n} x_{ij}^2}$$

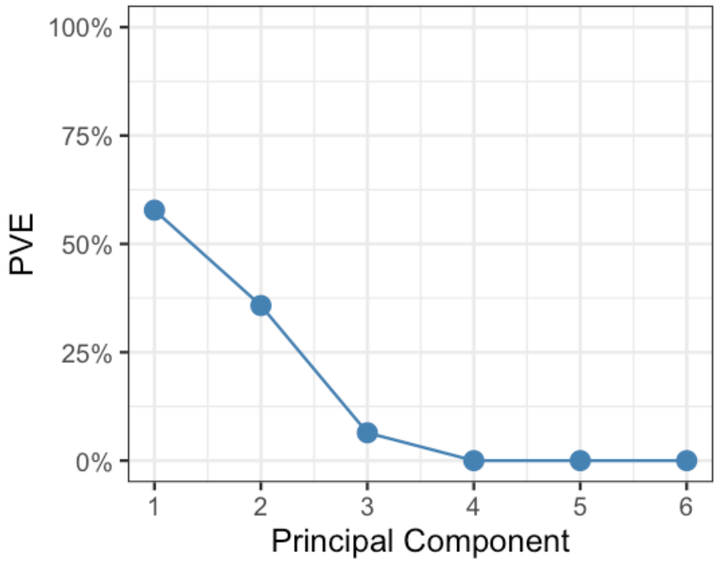


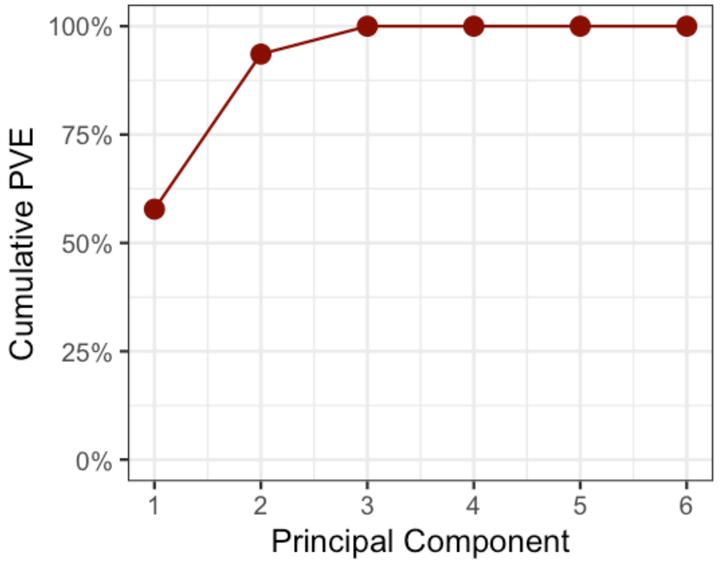
Not all PC are created equal…

How many do we really need?

- **If we use principal components as a summary of our data, how many components are sufficient?**
- No simple answer to this question, as cross-validation is not available for this purpose.
- Instead, the “scree plot” on can be used as a guide: we look for an “elbow”.

Some sort of rule-of-thumb: “pick smallest M that explains 80% of Cumulative PVE”





Slide from week 3

Derived input directions

Introduction (will skip principal components for now)

- Linear regression models using the original predictors $X_1, X_2, \ldots, X_{p-1}, X_p$ with different estimation techniques:
  - Ordinary least squares
  - Subset selection
  - Shrinkage
- Now, we'll explore approaches that transform the predictors and THEN fit a least squares model.

In many situations we have a large number of inputs, often very correlated.

In this section, we produce a small number of linear combinations $Z_m, m = 1,2,\ldots,M$ of the original inputs $X_j, j = 1,2,\ldots,p$, and Z_ms are then used in place of X_js as inputs in the regression.

$m < p$

The methods differ in how the linear combinations are constructed.

Dimension reduction (kind of…)

Slide from week 3

Math notation (🤢)

General schematic of reducing dimensionality of predictors

- **Step 1:** Let $Z_1, Z_2, \ldots, Z_M$ represent $M < p$ linear combinations (weighted average) of our original predictors:

$$Z_m = \sum_{j=1}^{p} \phi_{mj} X_j$$

Use PCs here!!

- **Step 2:** Fit a linear regression model

$$y_i = \theta_0 + \sum_{m=1}^{M} \theta_m z_{im} + \epsilon_i \quad \Longleftrightarrow \quad \boldsymbol{Y} = \boldsymbol{Z\theta} + \boldsymbol{\epsilon}$$

How well/poorly we choose $\phi_{m1}, \phi_{m2}, \ldots, \phi_{mp}$ will make/break this method.

$$\sum_{m=1}^{M} \theta_m z_{im} = \sum_{m=1}^{M} \theta_m \sum_{j=1}^{p} \phi_{mj} x_{ij} = \sum_{j=1}^{p} \sum_{m=1}^{M} \theta_m \phi_{mj} x_{ij} = \sum_{j=1}^{p} \beta_j x_{ij}$$

$$\beta_j = \sum_{m=1}^{M} \theta_m \phi_{mj}$$

<u>Potential gains</u>
Lose a little: higher bias
Overall: less MSE

Dimension reduction serves to constrain the estimated β_j coefficients

Principal components regression (PCR)

More use for derived PCs

- Uses the PCs $Z_1, Z_2, \ldots, Z_M$ of the predictors $X_1, X_2, \ldots, X_p$ as the **new predictors** in a model, instead of using the *original predictors*.

PCR handles multicollinearity

- If $X_1, X_2, \ldots, X_p$ are highly correlated, OLS estimates become unstable.
- Since $Z_1, Z_2, \ldots, Z_M$ are orthogonal (uncorrelated) by design, using them as predictors removes this problem.

PCR helps with dimensionality reduction

- For large number of predictors, OLS cannot be fit or will severely overfit.
- PCR fits the regression on only the first Mcomponents that capture most of the variance.

$$Y = Z\theta + \epsilon$$

Loss of Interpretability: The coefficients θ are for the principal components, not the original predictors. Since each Z is a linear combination of all original X, interpreting its effect on Y is difficult/less meaningful.

Independent Component Analysis*

Introduction

Imagine you are at a party with three microphones recording in different spots. Each microphone picks up a mixture of three people talking simultaneously.

- Observed signals: $X_1(t), X_2(t), X_3(t)$
- Source signals: $S_1(t), S_2(t), S_3(t)$

The goal of ICA is to take the mixed recordings (X) and "unmix" them to recover the original, independent voices (S)

ICA is a method for blind source separation. It assumes observed data is a linear combination of unknown, statistically independent latent sources.

Notation:

1. $\boldsymbol{X} = \left(X_1, X_2, \ldots, X_p\right)^T$: p observed, mixed signals.
2. $\boldsymbol{S} = \left(S_1, S_2, \ldots, S_k\right)^T$: k unknown, independent signals, $k < p$

Aim:

1. Recover the 'mixing matrix' A: $\boldsymbol{X} = A\boldsymbol{S}$
2. Recover the 'unmixed independent signals': $\hat{\boldsymbol{S}} = \hat{A}^{-1}\boldsymbol{S}$

Goal is to find $\hat{A}$ so that components of $\hat{\boldsymbol{S}}$ are as statistically independent as possible.

But if A is invertible, it must be square??

PCA to the rescue!!

Start with p features, obtain first k PCs $Z_1, Z_2, \ldots, Z_k$

Use $Z_1, Z_2, \ldots, Z_k$ as the mixed signals and $S_1, S_2, \ldots, S_k$ as signals.

Independent Component Analysis*

Applications

Artifact Removal in EEG & MEG

Problem: Electroencephalography (EEG) signals are notoriously noisy. The scalp electrodes record a mixture of true neural signals and powerful "artifacts" (noise) like eye blinks, muscle twitches (EMG), and heartbeats (ECG).

ICA Solution: We run ICA on the multi-channel EEG data to identify the independent components that correspond to artifacts (e.g., eye blinks have a very characteristic shape and spatial map). Set the artifact components to zero. Reconstruct the "clean" signal using the remaining components.

Result: A much cleaner EEG signal for clinical or research analysis.

Decomposing Omics Data (Gene Expression)

Problem: A tissue's gene expression profile (e.g., from RNA-seq) is a complex "mixture" resulting from many independent biological processes (e.g., cell cycle, immune response, stress) and technical factors (e.g., batch effects).

ICA Solution: ICA can be applied to a gene expression matrix (n samples × p genes) to find a set of independent components

Result: Each component represents a distinct "transcriptional signature" or regulatory program. This can help researchers discover novel biological pathways or identify and remove strong, unwanted batch effects in an unsupervised way.

Post-class debrief

Today we discussed…

When working with high-dimensional data (i.e., $p > 3$ features), it becomes difficult to visualize and analyze. A simple approach like creating pairwise scatterplots is insufficient, as it would require viewing $\approx p^2$ plots. The primary goal of PCA is to find an informative, low-dimensional representation of the data that captures the maximum possible variation. It seeks to identify the most "interesting" dimensions by measuring how much the observations vary along each new dimension. This also serves as a tool for Exploratory Data Analysis (EDA), pre-processing, and feature engineering

- Key components of implementing PCA:
  - PCA creates new variables, called Principal Components (PCs), which are linear combinations of the original features
  - Geometrically, the loadings define a direction in the feature space along which the data vary the most. The PC scores (Z_1) are the projections of the original data points onto this direction.
  - A scree plot is used to visualize the Proportion of Variance Explained (PVE) by each successive PC

- Principal Components Regression (PCR): PCR uses the first M principal components as the new predictors in a linear regression model instead of the original predictors. Since all PCs are orthogonal (uncorrelated) by design, PCR solves the problem of unstable estimates that OLS suffers from when predictors are highly correlated. It reduces model variance and prevents overfitting by fitting a model on fewer covariates. The main cost is a loss of interpretability. The model coefficients are for the PCs, which are complex linear combinations of all original predictors, making their individual effects difficult to interpret.


That's all Folks!