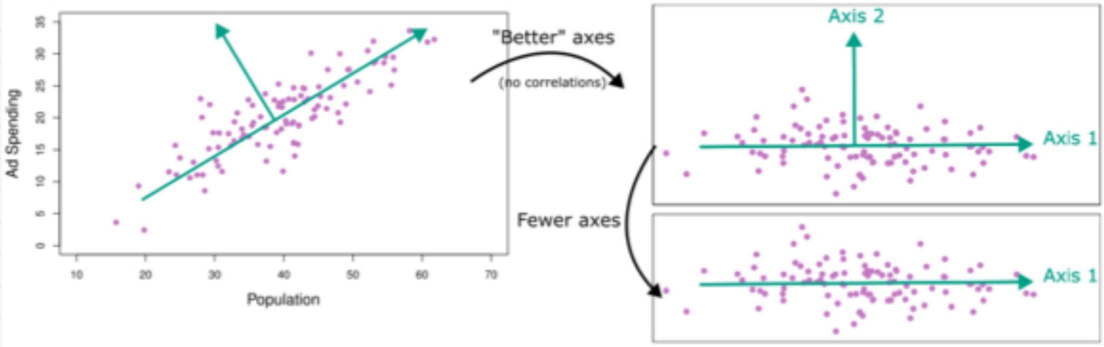
Lecture 10 - 01/10/24

Dimensionality Reduction & Visualization

- Normally our datasets are high dimensional



Principal Component Analysis

- Wish to represent data in a useful way for patterns
- Wanna see if data can be reduced to lower dimensions

PCA is an unsupervised method for such thing

- This means → no information about any class labels is used.
- PCA → finds a pattern
- PCA → looks for direction with greatest variance

*** Finding new Basis!**

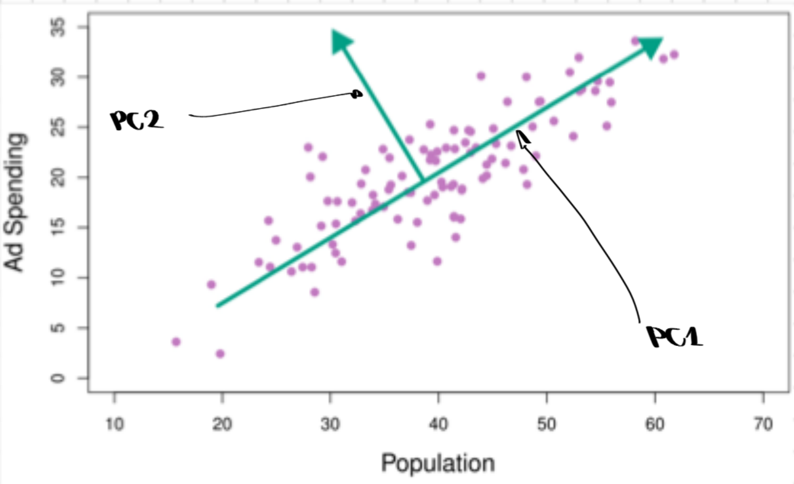
We want to build an orthogonal basis where new basis vectors are chosen to explain direction of the greatest variance.

*** Project to lower dimension**

First K principal components span a K-dimensional subspace that may be seen as the "best" K-dimensional view of data.

- Consider a set of p features $X_1, \ldots, X_p$ each a real-valued random var.

PCA → gives a new set of p features, principle components, each a linear comb of original p



Here the PC1 score of data point i is:

$$z_{i1} = 0.839(\text{pop}_i - \overline{\text{pop}}) + 0.544(\text{ad}_i - \overline{\text{ad}}).$$

- It points in (0.839, 0.544) in original
- New basis vector is centered at the column means for the data

The PC2 is instead orthogonal:

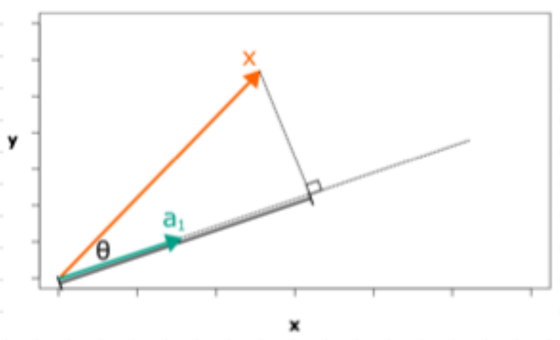
$$z_{i2} = -0.544(\text{pop}_i - \overline{\text{pop}}) + 0.839(\text{ad}_i - \overline{\text{ad}})$$

First Principal Component

- Imagine that we have n data points, each with p features
  - Data point i is : $x_i = \begin{pmatrix} x_{i1} \\ \vdots \\ x_{ip} \end{pmatrix} \xrightarrow{\text{sample mean}} \bar{x} = \sum_{i=1}^{n} x_i / n$

- Wanna find direction of maximal variance
  - We can use Linear Algebra to solve this

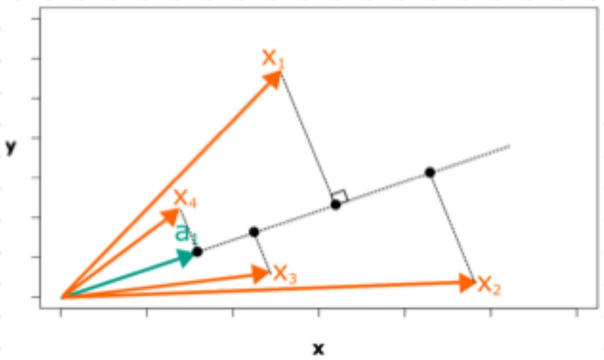
$$x_i \cdot a_1 = \|x_i\| \|a_1\| \cos\theta$$



  * Say that x_i represents one data point
  * Say that a_1 is some other vector
  * Do the dot product
    - ↳ projects one vector onto another

  * We can now call a_1 our first PC

$$a_1 = \begin{pmatrix} a_{11} \\ \vdots \\ a_{1p} \end{pmatrix}$$



  * If a_1 points to max variance, $a_1^T x$ vary a lot
    - ↳ we wanna maximize $\mathrm{Var}(a_1^T X)$

Maximization

$\mathrm{Var}(a_1^T X) = a_1^T S a_1$ where $S = \frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x})(x_i - \bar{x})^T$ = Covariance matrix

maximize this to constraint $a_1^T a_1 = 1$

we can use Lagrange multipliers

$$F(a_1, \lambda_1) = a_1^T S a_1 - \lambda_1 (a_1^T a_1 - 1)$$

↳ function we want to maximize

take partial derivative wrt a_1 :

$$\frac{\partial F}{\partial a_1} = 2 S a_1 - 2 \lambda_1 a_1 \xrightarrow{=0} S a_1 - \lambda_1 a_1 = 0$$

$$S a_1 = \lambda_1 a_1$$

Eigenvector corresponding to eigenvalue λ_1 in decomp.

We also need to max F in λ_1 :

- Rewrite F using a_1 satisfying $S a_1 = \lambda_1 a_1$

$$\begin{aligned} F(a_1, \lambda_1) &= a_1^T S a_1 - \lambda_1 (a_1^T a_1 - 1) \\ &= a_1^T \lambda_1 a_1 - \lambda_1 a_1^T a_1 + \lambda_1 \\ &= \lambda_1. \end{aligned}$$

we choose it to be the largest eigenvalue

Finding the Second & subsequent component!

Maximize:

$$Var(a_2^T X) = a_2^T S a_2 \xrightarrow{\text{constraints}} a_2^T a_2 = 1 \ \& \ a_1^T a_2 = 0$$

As before use **Lagrange multipliers:**

$$F(a_2, \lambda_2, \mu) = a_2^T S a_2 - \lambda_2(a_2^T a_2 - 1) - \mu(a_1^T a_2)$$

From here **partial derivatives, max, etc...**

Eigendecomposition

- With **PCA** we consider **eigendecomposition of S:**

$$S = A \Lambda A^T$$

- We choose Λ to be the **diagonal matrix** with eigenvalues $\lambda_1 \geq \ldots \geq \lambda_p \geq 0$
- The **orthogonal** $p \times p$ matrix $A = [a_1, \ldots, a_p]$ has **eigenvectors** as its **columns**
- This means that:
  - **PCA** chooses eigendecomposition that **orders eigenvectors according to dec eigval**
  - The **k^{th}** principal component is:

$$a_k^T X = a_{k1} X_1 + a_{k2} X_2 + \ldots + a_{kp} X_p$$

- **Coefficients** of principle components = **loadings**
  - Vector of loadings = eigenvector a_k
- **Principal component scores** = obs val of $a_k^T X$

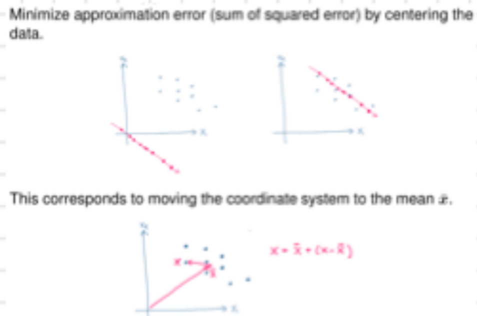
!!!!

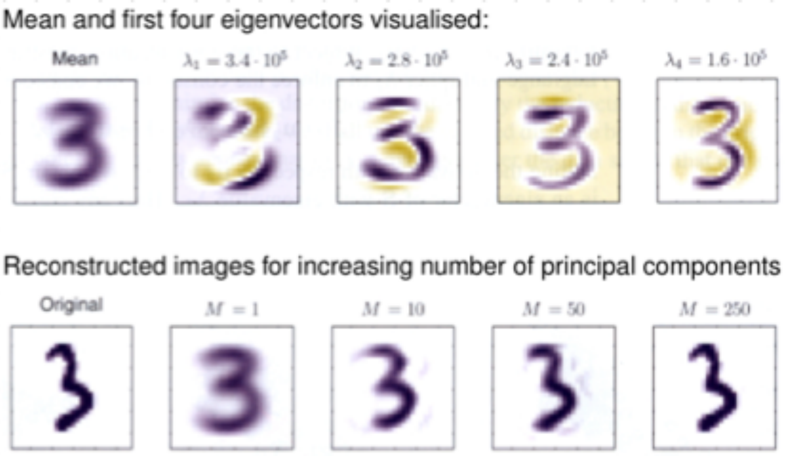
- The PCA **rotates your data** and projects it to $k \leq p$ dimensions
- PCA **creates uncorrelated features** that are **linear comb of existing p.**
- **Variance** of k'th principal component is the k'th **largest eigenvalue** $\rightarrow Var(a_k^T X) = \lambda_k$
- Each **PC** is a **vector of len p** as original data.
- If use **all p PC's** can **reconstruct** data point **x:**

$$x = \sum_{i=1}^{p} \underbrace{(a_i^T x)}_{} a_i$$

i'th PC score for x, meaning coordinate in PC space

- Do the same as above just with **first m PC's & will get an approximation**





How to choose # of PC's?

- ∃ many heuristic to do a good choice
- In supervised analysis, we can treat # as a tuning parameter

Let's talk about Variance!

- Total Sample Variance:

$$\mathrm{Var}(X_1)+\dots+\mathrm{Var}(X_p) = \mathrm{Var}(z_1^T X)+\dots+\mathrm{Var}(z_p^T X) = \lambda_1+\dots+\lambda_p$$

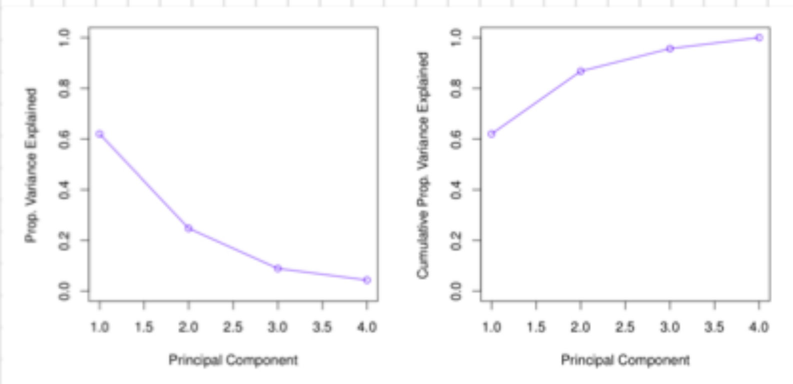
- Proportion of Variance explained by k^{th} PC:

$$\frac{\lambda_k}{\lambda_1+\dots+\lambda_p}$$

Note

Total Var is the same for the original p's & p^*'s given that trace of $S = P\Lambda P^{-1}$ is the same as trace of Λ

So how do we then decide # of components??



Left: How much Var by each PC

Right: Cumulative of left-hand

Centering & Standardization

- A variable may be centered by:

$$\tilde{X} = X - EX$$

- Can then be scaled to unit variance:

$$\tilde{X} = \frac{X}{\sqrt{\mathrm{Var}\,X}} = (\mathrm{Var}\,X)^{-1/2} X$$

- We can have the choice of Standardise features columnwise:
  - All features have unit var & correlation remains
- Another choice might be to Sphere the data:
  - All features have unit Var & no correlation