7️⃣

Dimensionality Reduction

Status	Done
Files	
Link	
Type	All Around
Course	Data Analysis

Why DR?

- Easier to understand, removing redundancies
- It's easier to visualise
- Less storage
- Less computational time
- Avoid overfitting in supervised learning tasks
- How?
  - Feature selection: just pick a subset of the features
  - Feature extraction: data is transformed from a high-dimensional space to a lower dimensional space. The most common method is called **principal component analysis (PCA)**

Principal Component Analysis (PCA)

> https://www.youtube.com/watch?v=HMOI_lkzW08

> https://www.youtube.com/watch?v=FgakZw6K1QQ&t=65s

> ❗ I really recommend watching these videos to get the intuition, otherwise the topic is very dry

- Linear tranformation technique
- Simple: if variables have high correlation then they are providing the same information so it makes sense to reduce the dimensions

Steps

0. Put all the features in a matrix

1. Standardise the data
    a. Mean = 0 Variance = 1
    b. Not always necessary, useful particularly if data was measured on different scales, usually useful for optimal performance of machine learning algorithms
2. Compute covariance matrix

Covariance Matrix

The classic approach to PCA is to perform the eigendecomposition on the covariance matrix Σ, which is a $d \times d$ matrix where each element represents the covariance between two features. The covariance between two features is calculated as follows:

$$\sigma_{jk} = \frac{1}{n-1} \sum_{i=1}^{N} \left(x_{ij} - \bar{x}_j\right)\left(x_{ik} - \bar{x}_k\right).$$

We can summarize the calculation of the covariance matrix via the following matrix equation:

$$\Sigma = \frac{1}{n-1}\left((\mathbf{X} - \bar{\mathbf{x}})^T (\mathbf{X} - \bar{\mathbf{x}})\right) = \mathbf{X}\mathbf{X}^\mathbf{T}$$

where $\bar{\mathbf{x}}$ is the mean vector $\bar{\mathbf{x}} = \sum\limits_{i=1}^{n} x_i$.
The mean vector is a d-dimensional vector where each value in this vector represents the sample mean of a feature column in the dataset.

3. Compute eigenvectors and eigenvalues of covariance matrix
    a. Eigenvectors determine the direction of the new feature space

b. Eigenvalues determine the magnitude

- How many dimensions to use in our new space? the eigenvalues tell us how much of the variance can be explained from each principle component, and there are usually diminishing returns so its not difficult to tell when to stop

Singular Value Decomposition (SVD)

> https://www.youtube.com/watch?v=UyAfmAZU_WI not the best video but statquest doesn't have one (yet, 2022)

- Based on a theorem that any matrix can be decomposed to 3 matrices that approximate it

$$A = U\Sigma V^T$$

where:

- A is users's ratings matrix (say dimension $n \times m$ if we have n users and m movies,
- U is the user "features" matrix (dimensions $n \times r$),
- Σ is the diagonal matrix of singular values (essentially weights, dimension is $r \times r$),
- V^T is the movie "features" matrix (dimension is $r \times m$).

U and V^T are orthogonal, and represent different things. U represents how much users "like" each feature and V^T represents how relevant each feature is to each movie. r is the rank of matrix A.

To get the lower rank approximation, we take these matrices and keep only the top k features, which we think of as the underlying "tastes and preferences" vectors.