VISPAD INSTITUTE OF TECHNOLOGY

Nea onnim no sua a ohu,

Matrix Decomposition & PCA

Fall 2025 💻

(October – December Virtual Internship)

At the end of the session the students or candidates should be able to understand and work with:

- Singular Value Decomposition (SVD)
- Principal Component Analysis

Clap for Yourself

Applications of SVD

- ❑ Creating of latent variables
  - ❖ These are hidden relationships that exist within data. *How can predict whether is it raining or not without information on the weather report?*
  - ❖ With SVD, these variables are present in the matrices U and V.

- These types of variables also allow us to reduce the dimensionality of a data set.

- Approximation of a matrix
  - We can remove data or information that is not relevant to what we are trying to do.
  - We can condense this approximation to a few vectors and scalars, which can extremely useful for storing data more efficiently.

Simple Recommendation System

- Let’s assume we have six different songs that users have rated according to their preferences.
- Our sample has eight users.

$$A = \begin{array}{c c} & \begin{array}{cccccc} song_0 & song_1 & song_2 & song_3 & song_4 & song_5 \end{array} \\ \begin{array}{c} user_0 \\ user_1 \\ user_2 \\ user_3 \\ user_4 \\ user_5 \\ user_6 \\ user_7 \end{array} & \begin{pmatrix} 1 & 2 & 3 & 4 & 1 & 5 \\ 4 & 1 & 0 & 3 & 5 & 1 \\ 3 & 0 & 1 & 2 & 2 & 4 \\ 0 & 2 & 1 & 0 & 1 & 5 \\ 0 & 3 & 1 & 3 & 4 & 1 \\ 2 & 1 & 1 & 4 & 2 & 3 \\ 0 & 4 & 1 & 4 & 0 & 3 \\ 5 & 0 & 3 & 0 & 1 & 5 \end{pmatrix} \end{array}$$

- We need to decompose matrix A into $U\Sigma V^T$.
- Using the full version of SVD, A will be decomposed into: $A_{8\times 6} = U_{8\times 8} \cdot \Sigma_{8\times 6} \cdot V^T_{6\times 6}$
- Matrix U has the following entries:

$$U = \begin{array}{c} \\ user_0 \\ user_1 \\ user_2 \\ user_3 \\ user_4 \\ user_5 \\ user_6 \\ user_7 \end{array} \begin{array}{c} \begin{array}{cccccccc} context_1 & context_2 & context_3 & context_4 & context_5 & context_6 & context_7 & context_8 \end{array} \\ \begin{pmatrix} -0.45 & 0.03 & -0.34 & -0.26 & -0.13 & -0.55 & 0.02 & -0.52 \\ -0.35 & 0.17 & 0.70 & 0.08 & 0.06 & 0.26 & -0.13 & -0.49 \\ -0.35 & -0.23 & 0.14 & -0.06 & -0.46 & 0.17 & 0.71 & 0.19 \\ -0.27 & -0.15 & -0.37 & 0.72 & -0.28 & 0.25 & -0.25 & -0.10 \\ -0.28 & 0.51 & 0.13 & 0.44 & 0.21 & -0.45 & 0.18 & 0.39 \\ -0.36 & 0.12 & 0.07 & -0.36 & -0.37 & 0.04 & -0.57 & 0.48 \\ -0.32 & 0.37 & -0.42 & -0.23 & 0.42 & 0.55 & 0.16 & 0.02 \\ -0.38 & -0.68 & 0.09 & 0.00 & 0.55 & -0.10 & -0.09 & 0.19 \end{pmatrix} \end{array}$$

- Matrix V has the following entries:

$$V = \begin{array}{c} \\ song_0 \\ song_1 \\ song_2 \\ song_3 \\ song_4 \\ song_5 \end{array} \begin{array}{c} \begin{array}{cccccc} context_1 & context_2 & context_3 & context_4 & context_5 & context_6 \end{array} \\ \begin{pmatrix} -0.36 & -0.28 & -0.26 & -0.46 & -0.35 & -0.62 \\ -0.46 & 0.45 & -0.20 & 0.54 & 0.28 & -0.41 \\ 0.55 & -0.31 & -0.19 & 0.00 & 0.61 & -0.44 \\ -0.28 & 0.33 & -0.10 & -0.66 & 0.56 & 0.23 \\ 0.40 & 0.64 & 0.42 & -0.24 & -0.20 & -0.40 \\ 0.34 & 0.34 & -0.82 & -0.02 & -0.27 & 0.17 \end{pmatrix} \end{array}$$

- Matrix Σ has the following entries:

$$\Sigma = \begin{pmatrix} 15.72 & 0 & 0 & 0 & 0 & 0 \\ 0 & 6.82 & 0 & 0 & 0 & 0 \\ 0 & 0 & 6.38 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3.13 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.8 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1.71 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

- If Σ is constructed in a way such that $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n$, then the highest sigma value will be the furthest point of projection.
- This will therefore represent the biggest spread and hence the highest variance.

- The measure of energy is defined as: $\sum_{i=1}^{n} \sigma_i^2$
- The total energy for our system is:

  $\sum_{i=1}^{6} \sigma_i^2 = 350.30$
- If we need around 95% of this energy, we can choose to σ_1, σ_2 and σ_3 and replace the rest with zeros.
- This will result in new U, Σ and V^T matrices and

$$\Sigma = \begin{pmatrix} 15.72 & 0 & 0 & 0 & 0 & 0 \\ 0 & 6.82 & 0 & 0 & 0 & 0 \\ 0 & 0 & 6.38 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

- We are keeping the information represented by the first three singular values.

- Everything will be reduced to 3 dimensions, which in machine is called dimensionality reduction.
- Clustering algorithm work better in lower dimensionality spaces.
- ***Working with a model with all of the available information does not guarantee good results.***

- The new version of three matrices U, Σ and V^T is:

$$\tilde{A}_{m\times n} = U_{m\times t} \cdot \Sigma_{t\times t} \cdot V^T_{t\times n}$$

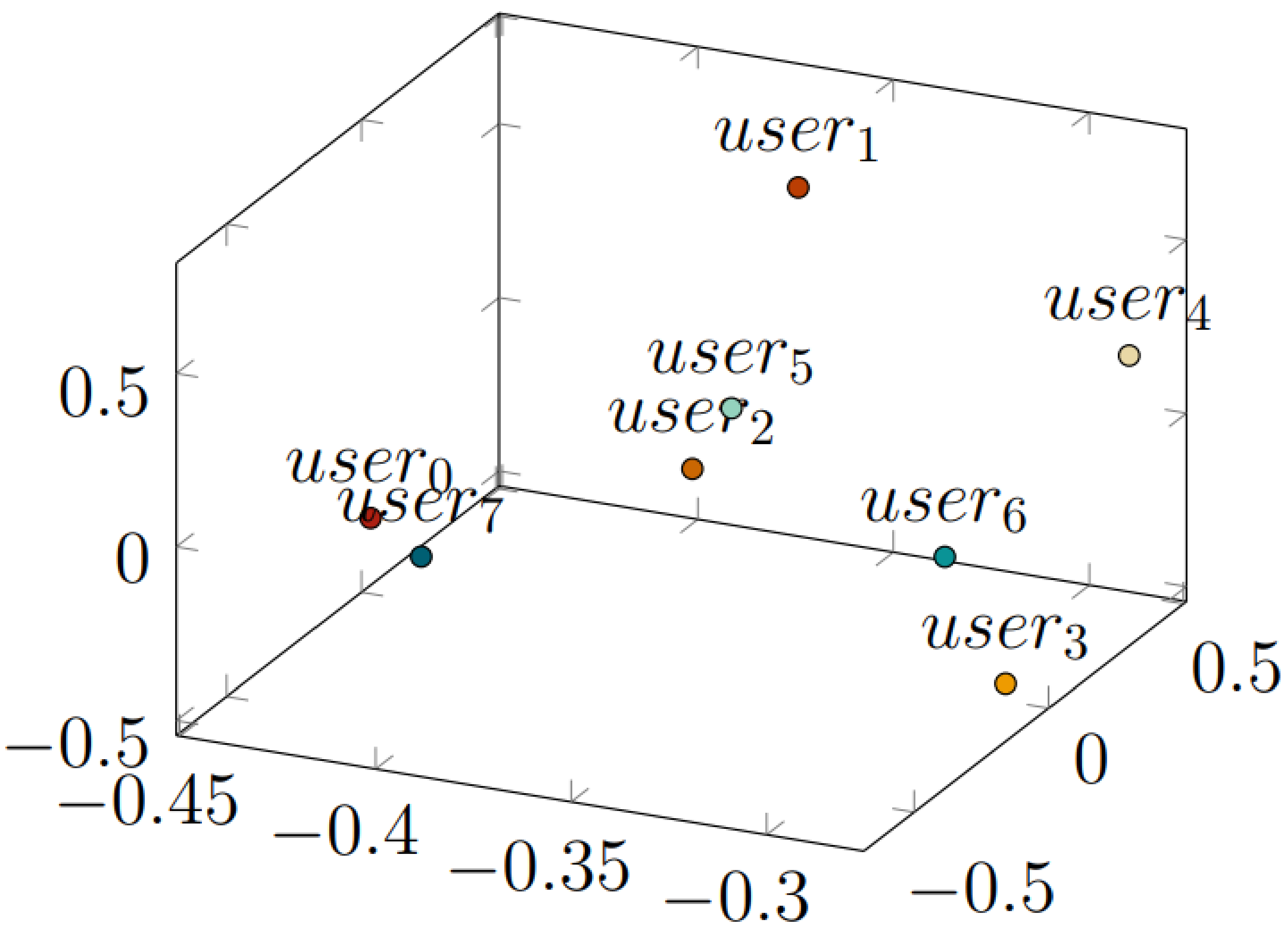
- t is the number of selected singular values.
- $A_{8\times 6} = U_{8\times 3} \cdot \Sigma_{3\times 3} \cdot V^T_{3\times 6}$

- For the truncated version of U, U_t, we can remove the vectors $\vec{u}_4$ to $\vec{u}_8$ because we replaced σ_4 to σ_6 with zero.

$$U_t = \begin{pmatrix} | & | & | & | & | & | & | & | \\ \overrightarrow{u_1} & \overrightarrow{u_2} & \overrightarrow{u_3} & \overrightarrow{u_4} & \overrightarrow{u_5} & \overrightarrow{u_6} & \overrightarrow{u_7} & \overrightarrow{u_8} \\ | & | & | & | & | & | & | & | \end{pmatrix}$$

$$\Sigma_t = \begin{pmatrix} \sigma_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_6 \end{pmatrix} \qquad V_t^T = \begin{pmatrix} — & \overrightarrow{v_1} & — \\ — & \overrightarrow{v_2} & — \\ — & \overrightarrow{v_3} & — \\ — & \overrightarrow{v_4} & — \\ — & \overrightarrow{v_5} & — \\ — & \overrightarrow{v_6} & — \end{pmatrix}$$

$$U_t = \begin{array}{c} \\ user_0 \\ user_1 \\ user_2 \\ user_3 \\ user_4 \\ user_5 \\ user_6 \\ user_7 \end{array} \begin{array}{c} \begin{array}{ccc} context_1 & context_2 & context_3 \end{array} \\ \begin{pmatrix} -0.45 & 0.03 & -0.34 \\ -0.35 & 0.17 & 0.70 \\ -0.35 & -0.23 & 0.14 \\ -0.27 & -0.15 & -0.37 \\ -0.28 & 0.51 & 0.13 \\ -0.36 & 0.12 & 0.07 \\ -0.32 & 0.37 & -0.42 \\ -0.38 & -0.68 & 0.09 \end{pmatrix} \end{array}$$



$user_1$
$user_4$
$user_5$
$user_2$
$user_0$
$user_7$
$user_6$
$user_3$
0.5
0
−0.5
−0.45
−0.4
−0.35
−0.3
0.5
0
−0.5

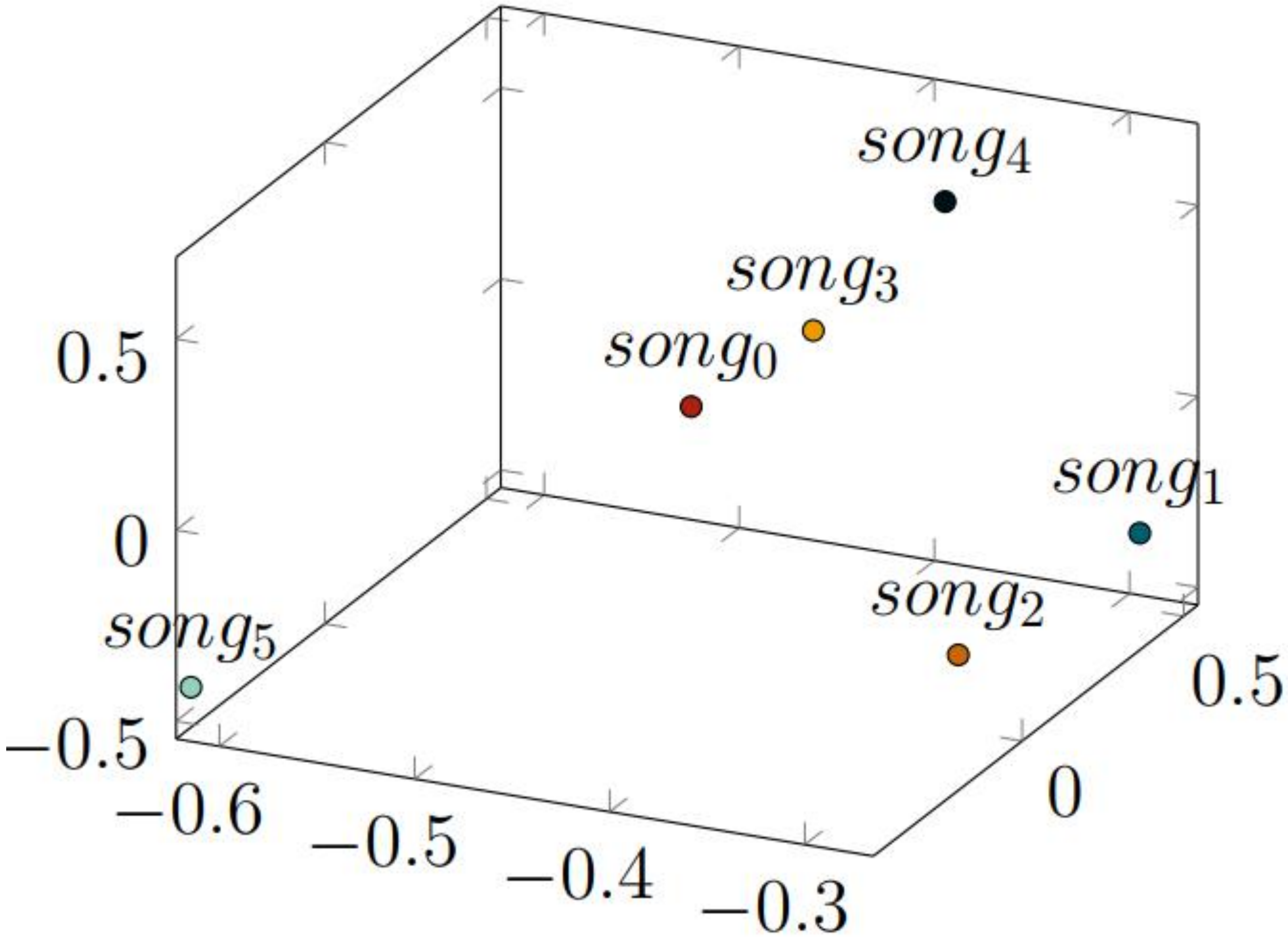
- We could segment users based on their music tastes by applying a clustering algorithm to matrix U.
- In case of higher dimensionalities with singular values that differ more, it would be better to use $U\Sigma$.
- A **clustering algorithm** has the goal of grouping data that shares identical features.

Cluster	Population
1	$user_0$ and $user_7$
2	$user_2$ and $user_5$
3	$user_6$ and $user_3$
4	$user_1$
5	$user_4$

- In machine learning, ***training*** means to find a configuration for a set of number of parameters that characterize a model or an equation.
- In order to train a model, we need to get a configuration of parameters for an equation (or several) that allows us to achieve our goal as well as we can.

- Reducing the number of columns going from A to U is called "***dimensionality reduction***" in machine learning.
- This is because we have moved from a higher dimensional space to a lower one, thus reducing the complexity of our problem.

$$V_t = \begin{array}{c c} & \begin{array}{ccc} context_1 & context_2 & context_3 \end{array} \\ \begin{array}{c} song_0 \\ song_1 \\ song_2 \\ song_3 \\ song_4 \\ song_5 \end{array} & \begin{pmatrix} -0.35 & -0.46 & 0.54 \\ -0.27 & 0.45 & -0.30 \\ -0.26 & -0.19 & -0.19 \\ -0.46 & 0.54 & 0.00 \\ -0.35 & 0.27 & 0.60 \\ -0.62 & -0.41 & -0.44 \end{pmatrix} \end{array}$$



$song_4$
$song_3$
$song_0$
$song_1$
$song_2$
$song_5$
0.5
0
−0.5
−0.6
−0.5
−0.4
−0.3
0.5
0

- By computing U and V^T, we can find hidden relationships in data, represented by latent variables.
- These relationships can be leveraged on to train algorithms.

- A matrix is linear transformation.
- When you multiply a vector by matrix, you will be moving these vectors.
- The movement can take place within the same space or between different space.

Principal Component Analysis

- Principal component analysis (PCA) has assumed an ever increasing role in a wide range of applications, including machine learning, image processing, speech recognition, face recognition, data mining, semantics, and health informatics.

- PCA is used to simplify data by looking for linear, or rather affine, relationships between the measurements of different data points.
- The key idea behind PCA is that the singular vector associated with larger singular values represent important correlations in the data, while those with smaller values indicate relatively unimportant

features or noise.

- Projecting the data onto the principal components yields an effective dimensionality reduction algorithm, which is widely employed in data analysis tasks and other applications, such as visualization of high dimensional data sets and image compression.

- *Principal component analysis* is a data projection into a new set of axes, or a change of basis that occurs via a linear transformation.
- The principal component analysis will create a new set of axes called the principal axes, where we will project the data and get the so-called principal

components.

- These are a linear combination of the original features that will be equipped with outstanding characteristics.
- Characteristics which are not only uncorrelated, but the first components also capture most of the variance in the data.

- The word axes means we need orthogonal vectors.
- If we have a symmetrical positive matrix, the eigenvectors are not only orthogonal but they also have positive and real eigenvalue.
- Multiplying a matrix by its transpose or the other way around will result in a symmetrical matrix, and with this, we can accommodate a matrix of any size.

- With PCA, we need to get a new set of axes onto which we can project data.
- We also need a component that reflects **variance.**
- Multiplying A^T by A will result in symmetrical matrix where the non-diagonal entries represent how much the rows relate to the columns.

- The numbers in the ***covariance matrix*** reflect how much variables vary with each other.
- Covariance matrix is symmetric positive, its eigenvalues are positive and the eigenvectors are perpendicular.

- Eigenvalues represent a scaling factor that comes from a covariance matrix, the largest value will correspond to the highest direction of variance; therefore, the correspondent eigenvector will be the first principal component.

- Given this data which represent features of user for online casino games.

user	totalBetsValue	totalWon	totalDaysPlayed	averageBetSize	totalSessions
1	3000	0	4	30	4
2	10453	0	1	100	1
3	21500	4230	6	50	7
4	10000	2000	12	10	14
5	340	10	10	1	10
6	5430	2000	4	5	70
7	43200	4320	10	4	32
8	2450	100	8	5	12

- Our goal is to transform this data with principal component analysis.
- The first step is to calculate the covariance matrix.
- For that, we need to centre the data.

- We will the equation: $x^* = \frac{x - \bar{x_j}}{\sigma_j}$

- From the equation, for each element in column j, we will subtract the mean of the same column and then divide the results by the standard deviation of this same column.
- If we do this to each column, we have standardized the data.

user	totalBetsValue	totalWon	totalDaysPlayed	averageBetSize	totalSessions
1	3000	0	4	30	4
2	10453	0	1	100	1
3	21500	4230	6	50	7
4	10000	2000	12	10	14
5	340	10	10	1	10
6	5430	2000	4	5	70
7	43200	4320	10	4	32
8	2450	100	8	5	12

Metric	totalBetsValue	totalWon	totalDaysPlayed	averageBetSize	totalSessions
x_j^-	12046.62	1582.50	6.87	25.62	18.75
σ_j	13329.42	1751.10	3.51	32.23	21.264

- The table below shows the standardized data.

user	totalBetsValue$_*$	totalWon $_*$	totalDaysPlayed$_*$	averageBetSize $_*$	totalSessions $_*$
1	-0.68	-0.90	-0.82	0.14	-0.69
2	-0.12	-0.90	-1.67	2.31	-0.83
3	0.71	1.51	-0.25	0.76	-0.55
4	-0.15	0.24	1.46	-0.48	-0.22
5	-0.88	-0.90	0.89	-0.76	-0.41
6	-0.50	0.24	-0.82	-0.64	2.41
7	2.34	1.56	0.89	-0.67	0.62
8	-0.72	-0.85	0.32	-0.64	-0.32

$$A = \begin{pmatrix} -0.68 & -0.90 & -0.82 & 0.14 & -0.69 \\ -0.12 & -0.90 & -1.67 & 2.31 & -0.83 \\ 0.71 & 1.51 & -0.25 & 0.76 & -0.55 \\ -0.15 & 0.24 & 1.46 & -0.48 & -0.22 \\ -0.88 & -0.90 & 0.89 & -0.76 & -0.41 \\ -0.50 & 0.24 & -0.82 & -0.64 & 2.41 \\ 2.34 & 1.56 & 0.89 & -0.67 & 0.62 \\ -0.72 & -0.85 & 0.32 & -0.64 & -0.32 \end{pmatrix}$$

$$A^T = \begin{pmatrix} -0.68 & -0.12 & 0.71 & -0.15 & -0.88 & -0.50 & 2.34 & -0.72 \\ -0.90 & -0.90 & 1.51 & 0.24 & -0.90 & 0.24 & 1.56 & -0.85 \\ -0.82 & -1.67 & -0.25 & 1.46 & 0.89 & -0.82 & 0.89 & 0.32 \\ 0.14 & 2.31 & 0.76 & -0.48 & -0.76 & -0.64 & -0.67 & -0.64 \\ -0.69 & -0.83 & -0.55 & -0.22 & -0.41 & 2.41 & 0.62 & -0.32 \end{pmatrix}$$

- To get the covariance matrix, we need to multiply A^T by A and then divide by the number of entries.

$$\frac{A^T A}{8} = M = \begin{pmatrix} 1.00 & 0.84 & 0.23 & 0.02 & 0.13 \\ 0.84 & 1.00 & 0.29 & -0.14 & 0.33 \\ 0.23 & 0.29 & 1.00 & -0.73 & -0.01 \\ 0.02 & -0.14 & -0.73 & 1.00 & -0.47 \\ 0.13 & 0.33 & -0.01 & -0.47 & 1.00 \end{pmatrix}$$

- M is symmetrical matrix, how do we get the eigen-vectors?
- Let's use the eigendecomposition approach.
- Eigendecomposition returns ***three matrices*** after the decomposition.

$$M = P\Sigma P^{-1}$$

$$P = \begin{pmatrix} 0.45 & 0.53 & 0.44 & -0.43 & 0.34 \\ 0.53 & 0.42 & -0.38 & 0.60 & -0.13 \\ -0.14 & 0.01 & -0.54 & -0.04 & 0.82 \\ 0.69 & -0.68 & -0.11 & -0.19 & 0.05 \\ 0.00 & -0.24 & 0.58 & 0.64 & 0.42 \end{pmatrix}$$

- P is a matrix with the eigenvectors, and this will be where we find our principal axes.

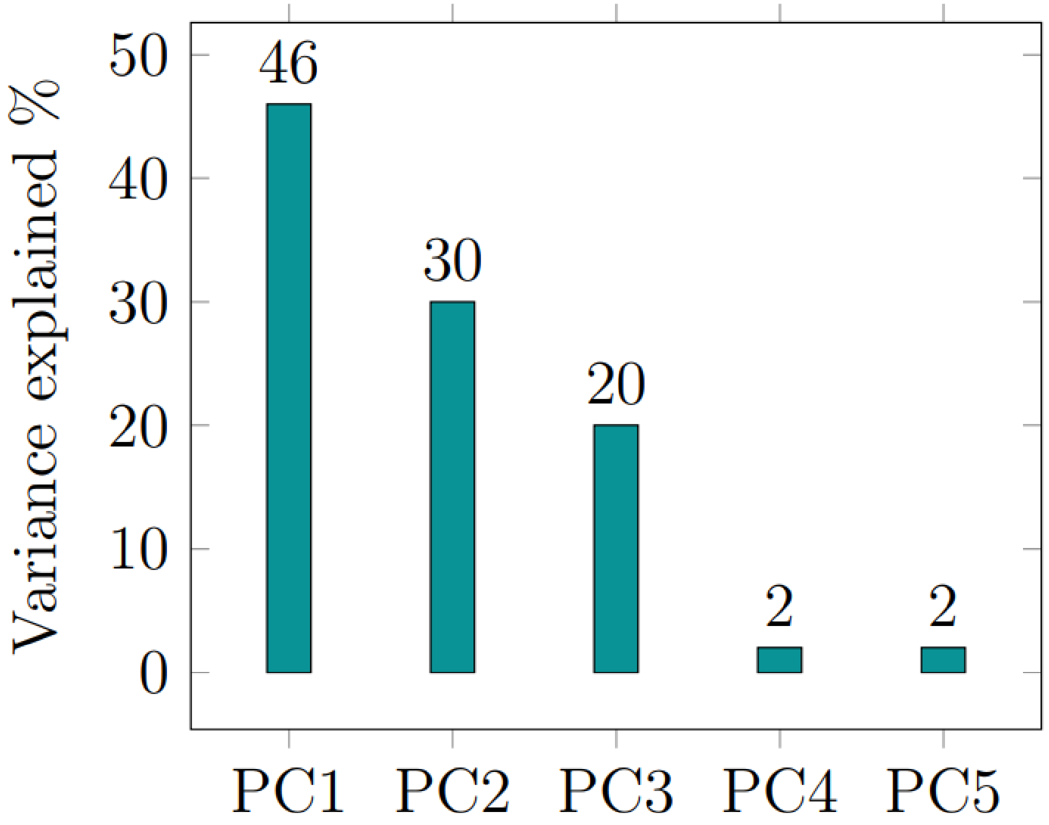
$$\Sigma = \begin{pmatrix} 2.29 & 0 & 0 & 0 & 0 \\ 0 & 1.48 & 0 & 0 & 0 \\ 0 & 0 & 1.02 & 0 & 0 \\ 0 & 0 & 0 & 0.14 & 0 \\ 0 & 0 & 0 & 0 & 0.08 \end{pmatrix}$$

- ❑Σ is the matrix with the eigenvalues.
- ❑These values can be called the explainability of the variance.

- How much of the variance present in the data is "captured" by each component.
- We can find out how much each eigenvalue explain the variance.

$$\sum_{i=1}^{5} \lambda_i = 5$$

- We find the individual percentage of variance that each distinct eigenvalue explains.



- Two components, PC1 and PC2 explains 76% of the variability.
- We pick the two largest eigenvalue and transform our original data set to create the first two components for each users in the data set.

❑The new version of P, P_t

$$P_t = \begin{pmatrix} 0.45 & 0.53 \\ 0.53 & 0.42 \\ -0.14 & 0.01 \\ 0.69 & -0.68 \\ 0.00 & -0.24 \end{pmatrix}$$

- The linear combinations the represent the first two components are described by the two equations:

$$\begin{aligned}\text{PC1} = 0.45 \cdot \text{totalBetsValue} + 0.53 \cdot \text{totalWon} - 0.14 \cdot \text{totalDaysPlayed} \\ +0.69 \cdot \text{averageBetSize} + 0.00 \cdot \text{totalSessions}\end{aligned}$$

$$\begin{aligned}\text{PC2} = 0.53 \cdot \text{totalBetsValue} + 0.42 \cdot \text{totalWon} + 0.01 \cdot \text{totalDaysPlayed} \\ -0.68 \cdot \text{averageBetSize} - 0.24 \cdot \text{totalSessions}\end{aligned}$$

- We have to transform our versions of the scaled data with the matrix P_t

$$M_{reduced} = M_{8\times5} \cdot P_{5\times2}^{T}$$

- We can create the first and second components for all user in the features set:

user	PC1	PC2
1	-1.46	-0.26
2	-2.56	1.69
3	0.51	1.65
4	0.84	-0.81
5	-0.30	-1.60
6	0.63	-0.55
7	2.81	1.10
8	-0.48	-1.22

- These components are called latent or hidden variables; relationships that are hidden in the data and are the result of linear combination.


Thank You