

# Introduction

(Neural Networks Implementation and Application Tutorial)

Vilém Zouhar, Noon Pokaratsiri Goldstein

28th November 2021

# Overview

- Introduction
- Requirements, Materials, Assignments
- PCA, SVD
- Current assignment
- QA

# Hello

Who am I?

# Hello

Who am I?

Who are you?



# Introduction

Choose and answer at least two questions:

- On scale from 1-10 how proficient are you in programming and mathematics?
- What topics of Neural Networks excite you the most?
- What topics of Neural Networks excite you the least?
- What programming languages do you know?
- How best can the tutotial sessions be helpful to your needs?

Also the following:

- Who is your groupmate?
- Will you be attending Vilém's or Noon's tutorials?

# Requirements

## Tutorial Requirements (exam admission)

- 60% of mandatory points (~10 assignments, 10 points each)
- Tutorial points only for exam admission (no final grade influence)

## Tutorial Bonus Points

- ~2pts for extra exercises in the assignments
- 1pt for answering a question in a tutorial
- ??pt for fixing errors in tutorial presentations
  - ▶ [github.com/zouharvi/uds-nnia-tutorial](https://github.com/zouharvi/uds-nnia-tutorial)

## Final Project

- None

## Transfer from last year

- Maybe possible (tbd)
- Assignments recommended (because of the exam)

# What's available

- Lectures by Prof. Klakow (recorded)
- Tutorials (not recorded, but allowed for private sharing)
- Corrected homework
- Consultations
  - ▶ Only in specific cases
  - ▶ By default **no** email and **no** personal chat
  - ▶ Ask questions during the lecture / tutorials
- Public forum (please use Piazza)
  - ▶ Ask questions
  - ▶ Other students will also benefit from the answers
  - ▶ You can answer someone else's issue

# Assignments

- Mandatory groups of 2
- Usually 2 exercises per assignment + a possible bonus question
- Jupyter notebook templates
  - ▶ Assignment + solution in the same notebook
  - ▶ Can use Google Colab or local runtime
  - ▶ Write solutions in Python files and import them
  - ▶ Submitted notebook must only contain your analysis and outputs
- Only one submission per group
  - ▶ Submit through Teams



# Dates / Times

- Lecture:
  - ▶ Tuesday 14:15-15:45
- Tutorials:
  - ▶ Vilém: Wednesday 16:00-18:00
  - ▶ Noon: Thursday 08:30-10:00
- Assignments
  - ▶ Released (usually) by Wednesday 08:00 (available in Teams)
  - ▶ Deadline (next) by Wednesday 08:00 (submit in Teams)
- Exam: TBD

# Tutorial Content

- Review of the topics covered in class
- Presentation of the past assignment
- Discussing current assignment

# Organization

Questions?

# Assignment 0

- Questions?
- Did it work?
- How long did it take?

## Feedback:

- Change **TODO** to **Solution**.
- Don't forget to write amount of work.
  - ▶ Useful for our estimates of difficulty.

# Linear Algebra Basics

Few definitions (+how are they implemented in Python/Numpy/PyTorch)

- Scalars
- Vectors
- Matrices
- Tensors

# Linear Algebra Basics

Few definitions (+how are they implemented in Python/Numpy/PyTorch)

- Scalars
- Vectors
- Matrices
- Tensors

Identify the following objects (Python lists):

- `[5.0, 3.0]`
- `5.0`
- `[True]`
- `[[5, 1], [0, 4]]`
- `[[True, False], [False, True]]`
- `[ [[0,1], [0,1], [0,1]], [[0,1], [0,1], [0,1]] ]`

# Linear Algebra Basics

## A few operations and properties involving matrices:

- Transpose
- Inverse
- Dot product (i.e. matrix multiplication)
  - ▶  $C = AB, C_{i,j} = \sum_k A_{i,k} B_{k,j}$

## Common Properties:

- $A(B + C) = AB + AC$
- $A(BC) = (AB)C$
- $AB \neq BA$
- $(AB)^T = B^T A^T$

# Linear Algebra Basics

## Definitions:

columns :: column > - Eigenvector, Eigenvalue > -  $Av = \lambda v, v \neq \vec{0}$  > -  
Eigendecomposition > -  $A = Q \cdot L \cdot Q^{-1}$  > - Singular value decomposition (SVD) > -  
 $A = U \Sigma V^T$  > - Principal Component Analysis: > - Eigendecomposition or SVD of covariance  
matrix  $W = \frac{A^T A}{n-1}$  > - Assume ordering of {eigen,singular} values from highest to lowest > -  
Project to  $k$  dimensions:  $A_k = A Q_k$  :: :: column > - Eigenvalue decomposition > - Singular  
value decomposition > - PCA :: ::

## True or False? 🤔

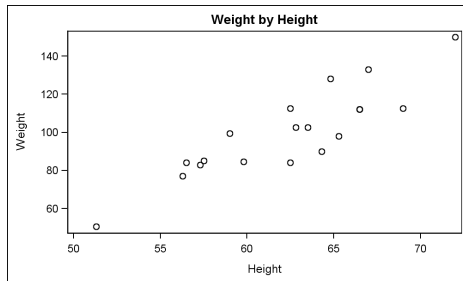
- Every real matrix has an eigenvalue decomposition (in  $\mathbb{R}$ ).
- Every real matrix has a singular value decomposition (in  $\mathbb{R}$ ).
- Every real symmetric matrix has an eigenvalue decomposition (in  $\mathbb{R}$ ).



## Linear Algebra Basics - True or False? 🤔

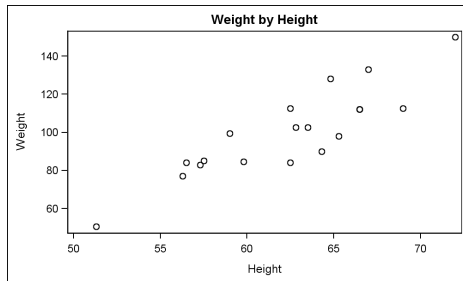
$$A = \begin{pmatrix} 4 & 2 \\ 2 & 4 \end{pmatrix}$$

- 1 Is  $v_1 = (1, -1)$  an eigenvector of  $A$ ?
- 2 Is  $v_2 = (2, 1)$  an eigenvector of  $A$ ?
- 3 Is  $v_3 = (2, 2)$  an eigenvector of  $A$ ?



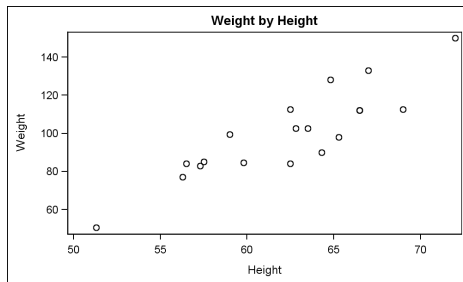
## Questions 🤔

- What will be the first principal component?



## Questions 🤔

- What will be the first principal component?
- Does anyone know how PCA works?



## Questions 🤔

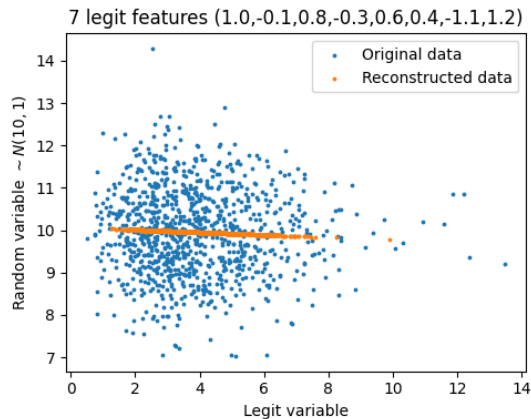
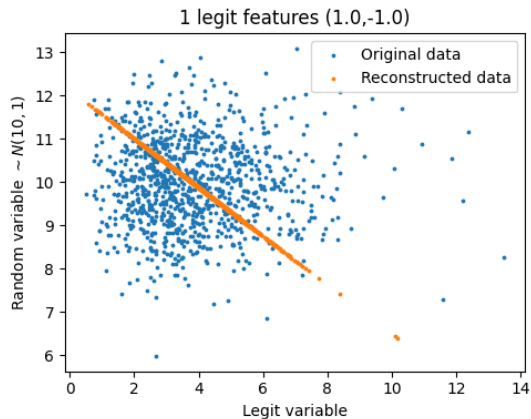
- What will be the first principal component?
- Does anyone know how PCA works?
- What does it mean that we take only  $k$  largest principal components?

# PCA

- Is it safe to say that the first component will always contain the most important information? 🤔

# PCA

- Is it safe to say that the first component will always contain the most important information? 🤔



# Standardization

- Is not normalization! ( $x' = \frac{x}{|x|}$ )
- $X = \frac{X - \text{mean}(X)}{\text{std}(X)}$
- Compute either:
  - ▶ With Numpy: `X = (X-X.mean())/np.std(X)`
  - ▶ With Scikit: `StandardScaler().fit_transform(X)`
- Why do we need standardization for PCA? 🤔

# Assignment 1

- Any questions?



# Typesetting Tips

- Do **not** write  $A * B$ , use `\cdot` or `\times`:  $A \cdot B, A \times B$ .
- Use LaTeX functions when available, e.g. `\log`, `\sin`:  $\log(x), \sin(x)$ , **not**  $\log(x), \sin(x)$ .
- Do **not** write plain text in math mode, use `\text{ComputeEigenvalues}(X)`

# Resources

- ① Course Website:  
[lsv.uni-saarland.de/neural-networks-implementation-and-application-winter-2021-2022-2](https://lsv.uni-saarland.de/neural-networks-implementation-and-application-winter-2021-2022-2)
- ② Piazza: <https://piazza.com/class/kvc3vzhsvh55rt>
- ③ Tutorial repository [github.com/zouharvi/uds-nnia-tutorial](https://github.com/zouharvi/uds-nnia-tutorial)
- ④ Lecture & tutorial teams channels