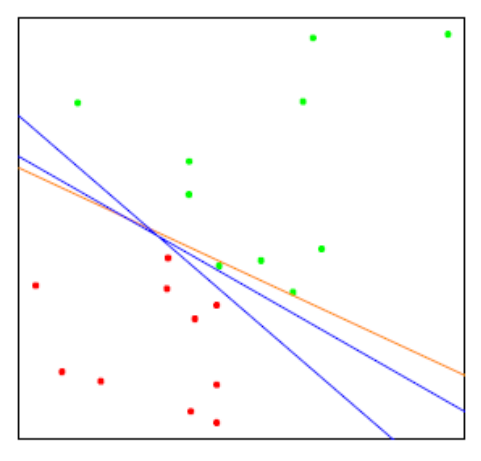


Parallel & Scalable Machine Learning

Introduction to Machine Learning Algorithms

Dr. Gabriele Cavallaro

Postdoctoral Researcher

High Productivity Data Processing Group

Juelich Supercomputing Centre

LECTURE 9

Data Preparation and Performance Evaluation

February 26th, 2019

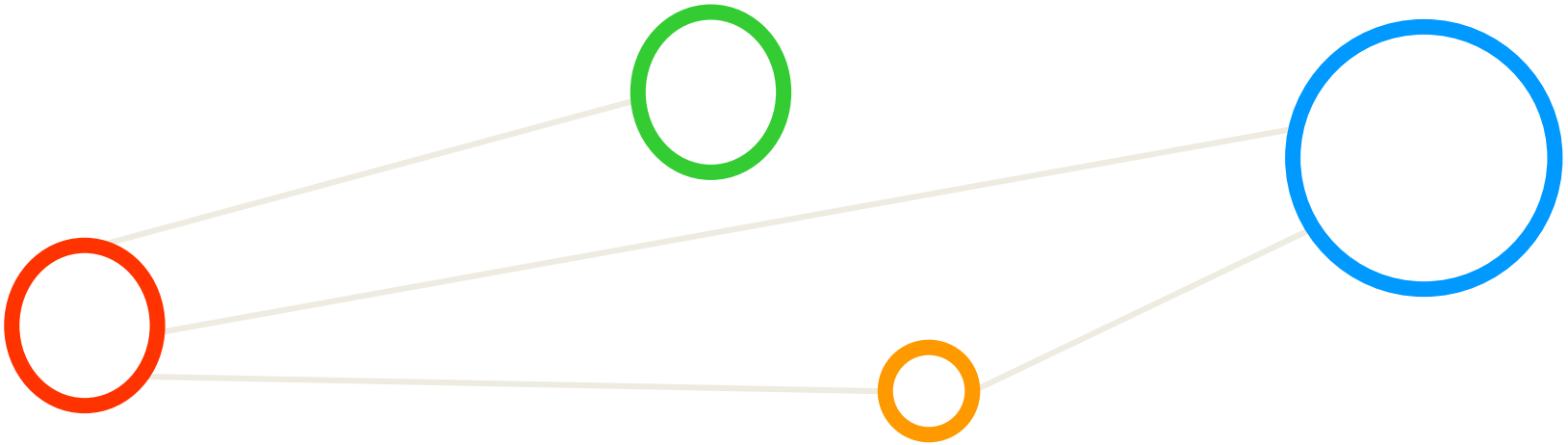
JSC, Germany

UNIVERSITY OF ICELAND

SCHOOL OF ENGINEERING AND NATURAL SCIENCES

FACULTY OF INDUSTRIAL ENGINEERING, MECHANICAL ENGINEERING AND COMPUTER SCIENCE

Outline



Outline of the Course

1. Parallel & Scalable Machine Learning driven by HPC
2. Introduction to Machine Learning Fundamentals
3. Introduction to Machine Learning Fundamentals
4. Feed Forward Neural Networks
5. Feed Forward Neural Networks
6. Validation and Regularization
7. Validation and Regularization
8. Data Preparation and Performance Evaluation
9. Data Preparation and Performance Evaluation
10. Theory of Generalization
11. Unsupervised Clustering and Applications
12. Unsupervised Clustering and Applications
13. Deep Learning Introduction

Theoretical Lectures

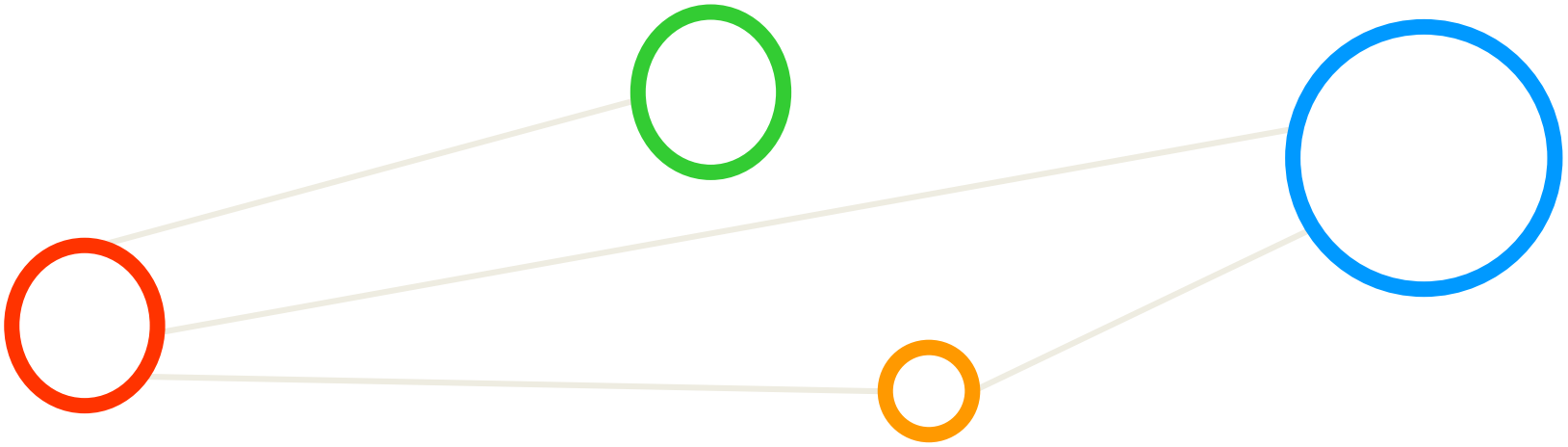
Practical Lectures

Outline

- Principal Component Analysis
  - Main Steps
  - Iris Dataset
  - Eigenvectors and Eigenvalues
  - Covariance Matrix
  - Eigendecomposition and Projection

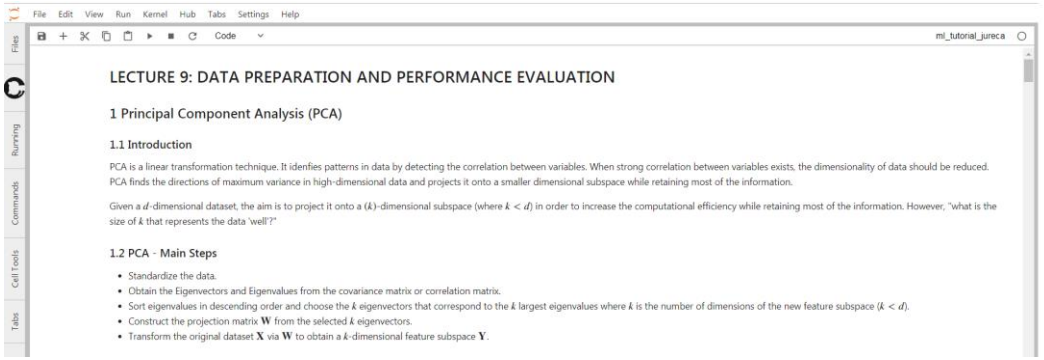
- Classification of Remote Sensing Images
  - Indian pine Hyperspectral dataset
  - Hackathon
    - Standardize the Data
    - Reduce the Data
    - ANN Hyperparameter optimization
    - Accuracy Metrics

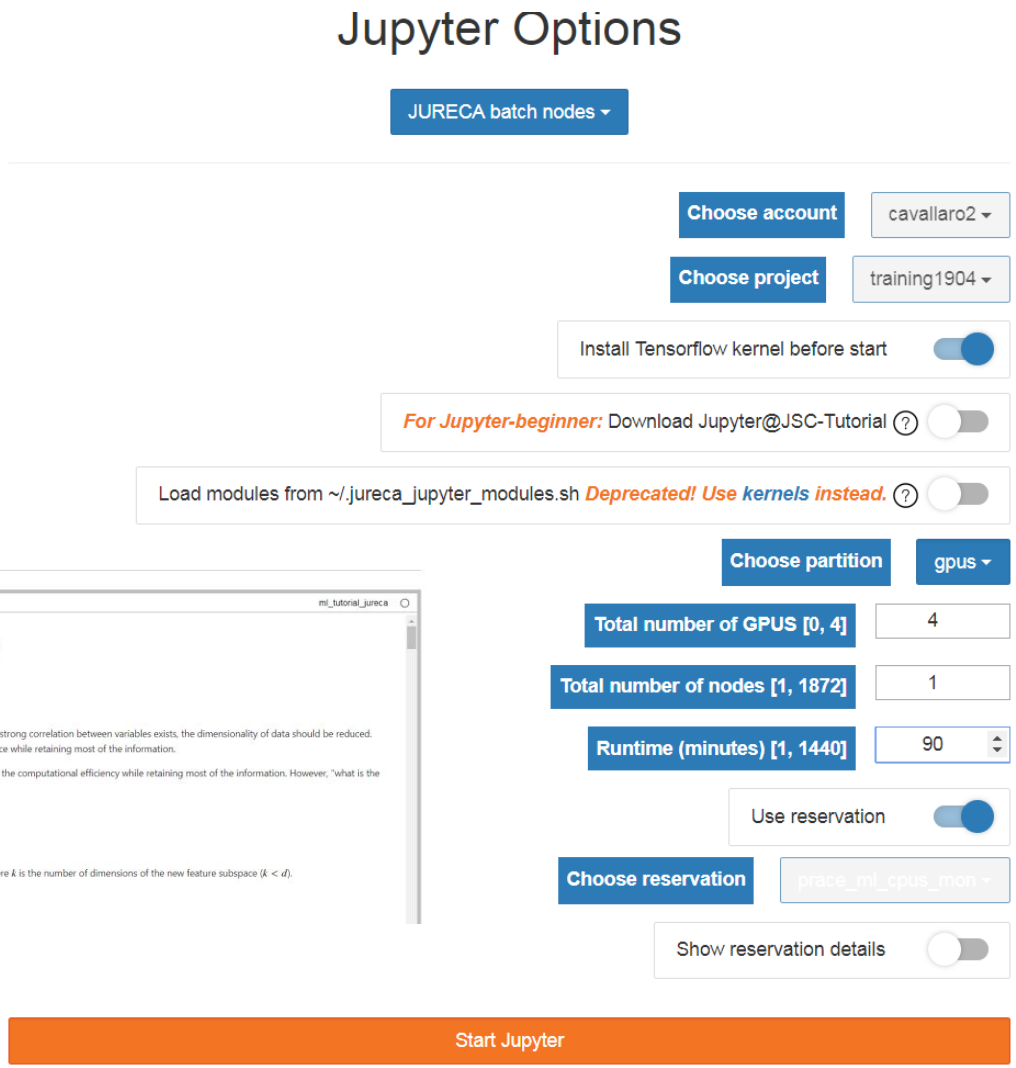
Principal Component Analysis



Jupyter Notebook

1. Access Jupyter
2. Upload lecture_9.ipynb





PCA Approach

- Lets consider a data matrix **X** of size [n x d]
  - n: number of samples
  - d: dimensionality (i.e., number of features)

1. Standardize the data

2. Obtain the **Eigenvectors** and **Eigenvalues** from the **covariance matrix**

3. Sort eigenvalues in descending order and choose the **k** eigenvectors that correspond to the **k** largest eigenvalues
   - **k**: number of dimensions of the new feature subspace (**k≤d**)

4. Construct the projection matrix **W** from the selected k eigenvectors.

5. Transform the original dataset **X** via **W** to obtain a **k**-dimensional feature subspace **Y**

[1] Plotly Open Source Graphing Libraries
[2] Sebastian Raschka, PCA in ipython-notebooks

Iris Flower Data Set Revisited

- **Task:** apply PCA for improving the discriminability of the three classes of flowers

Three classes: Iris setosa | Iris versicolor | Iris virginica

[1] UCI Machine Learning Repository Iris Dataset

(four data attributes for each sample in the dataset)

- sepal length in cm
- sepal width in cm
- petal length in cm
- petal width in cm

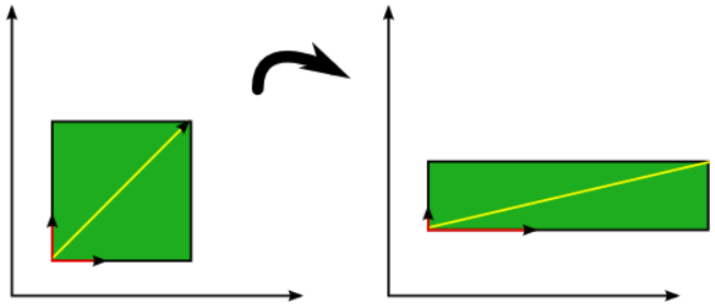
- Can we find a data sub-space with less number of features (i.e., at least <4)?

Eigenvectors and Eigenvalues

- **Eigenvectors (principal components)**
  - Determine the directions of the new feature space

- **Eigenvalues**
  - Provide the magnitude of eigenvectors
  - Explain the variance of the data along the new feature axes

- **Interpretation: e.g., Linear transformation with scaling operation**
  - Scale a vector: $\boldsymbol{v} * \mathbf{A} = \begin{bmatrix} \mathbf{2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0.5} \end{bmatrix}$
  - The direction of some vectors (shown in red) is not affected by this linear transformation
  - They are called eigenvectors of the transformation
  - They uniquely define $\mathbf{A}$ (eigen, "its own" in German)



[2] Eigenvectors and eigenvalues

- Generally, the eigenvector $\boldsymbol{v}$ of a matrix $\mathbf{A}$ is the vector for which the following holds

$$\boldsymbol{Av} = \lambda \boldsymbol{v}$$

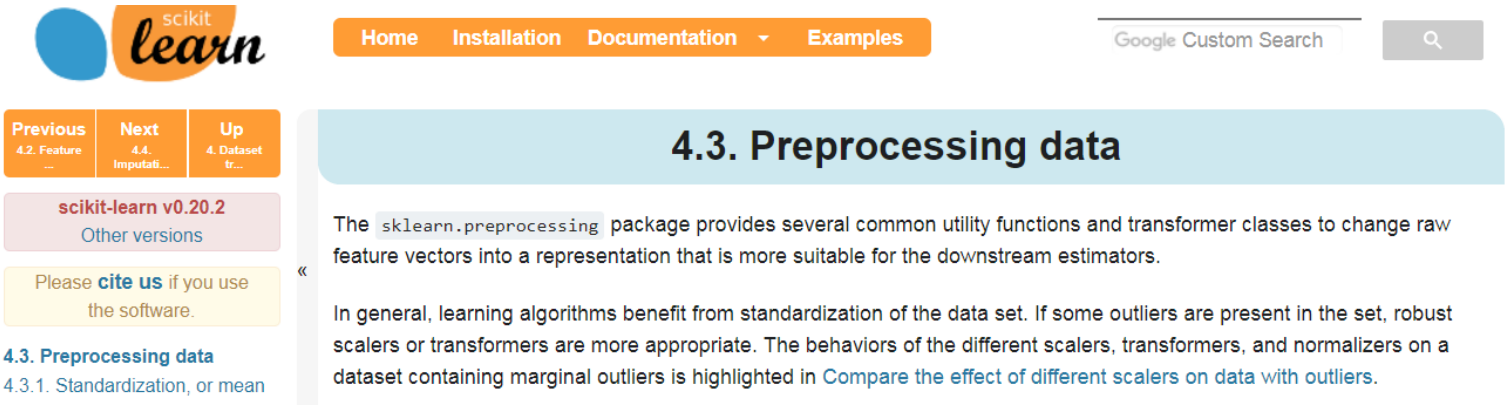
with λ is a scalar value called the 'eigenvalue'.

Recommended video:

[3] Eigenvectors and eigenvalues, Essence of linear algebra

Standardize the Data

- **sklearn.preprocessing package**
  - Provides several common utility functions and transformer classes to change raw feature vectors into a representation that is more suitable for the downstream estimators



[4] Preprocessing Data

Covariance Matrix

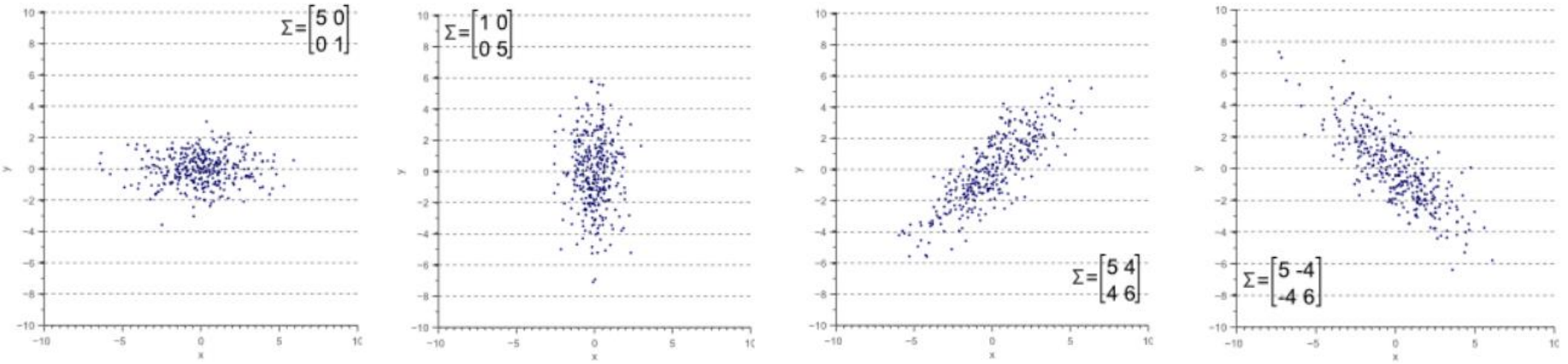
- The covariance matrix Σ defines the shape of N-dimensional data
- Σ [d x d] covariance matrix, each element represents the covariance between two features

$$\theta_{j,k} = \frac{1}{n-1}\sum_{k=1}^{n}(x_{i,j} - \bar{x}_j)(x_{ik} - \bar{x})$$

- Example for 2D data :

$$\Sigma = \begin{bmatrix} \theta(x,x) & \theta(x,y) \\ \theta(y,x) & \theta(y,y) \end{bmatrix}$$

  - The variance ($\theta(x,x)$, $\theta(y,y)$): captures the axis-aligned spread
  - The covariance ($\theta(x,y)$, $\theta(y,x)$): captures the diagonal spread



[5] A Geometric Interpretation of the Covariance Matrix

Eigendecomposition on the Covariance Matrix (1)

- The covariance matrix can be interpreted as a linear operator

- **How eigenvectors and eigenvalues uniquely define the covariance matrix?**
  - i.e., the shape of the data

- The largest eigenvector of the covariance matrix always points into the direction of the largest variance of the data, and its magnitude equals the corresponding eigenvalue

- The second largest eigenvector is always orthogonal to the largest eigenvector, and points into the direction of the second largest spread of the data

[5] A Geometric Interpretation of the Covariance Matrix

Eigendecomposition on the Covariance Matrix (2)

- The covariance matrix can be represented as a function of its eigenvectors and eigenvalues

$$\Sigma = \underbrace{V\ L\ V^{-1}}_{\text{Eigendecomposition}}$$

V : matrix whose columns are the eigenvectors of Σ

L: diagonal matrix whose non-zero elements are the corresponding eigenvalues

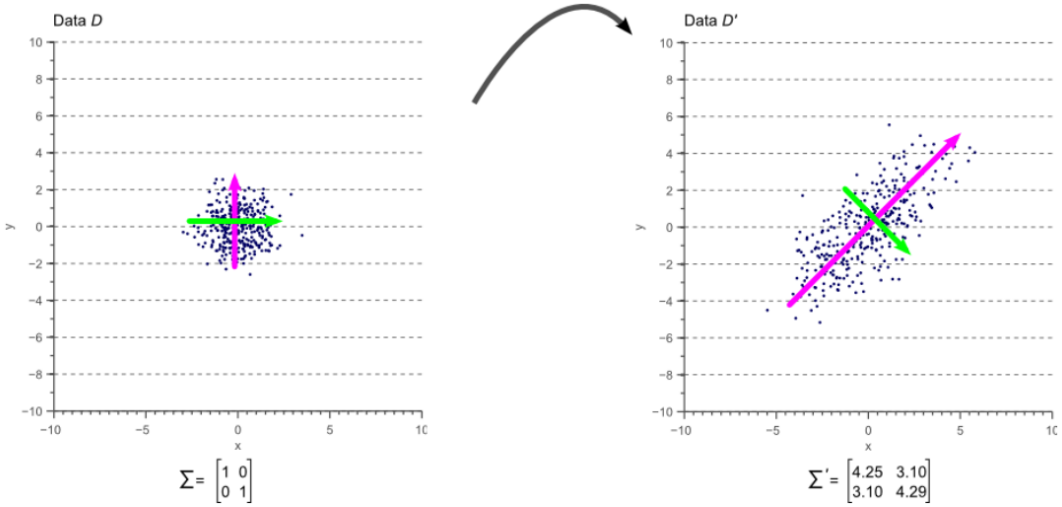
- Can be obtained using a **Singular Value Decomposition** algorithm

[6] SVD

- Apply linear transformation with $\mathrm{D}' = V\ \sqrt{L}\ D$

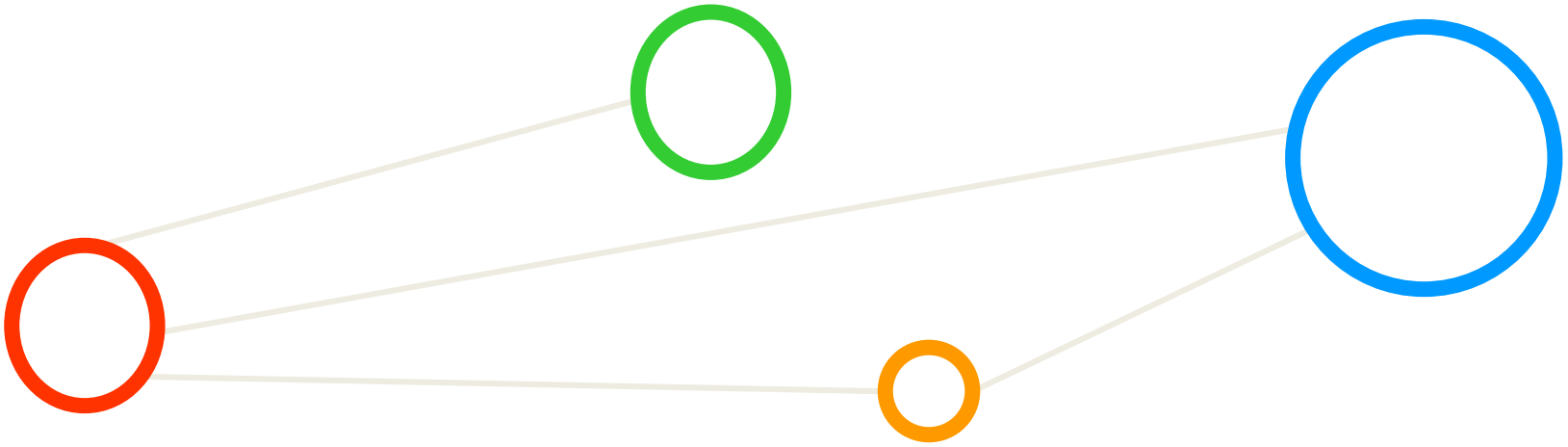
V : rotation matrix

$\sqrt{L}$: scaling matrix



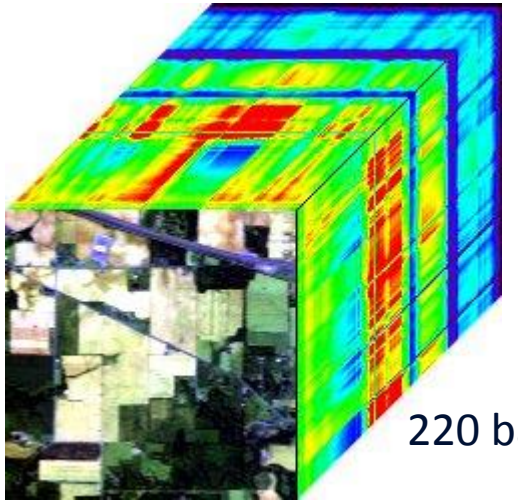
[5] A Geometric Interpretation of the Covariance Matrix

Classification of Remote Sensing Images



Indian Pine Dataset (small site)

- It contains 2/3 agriculture, and 1/3 forest or other natural perennial vegetation
- There are two major dual lane highways, a rail line, as well as some low density housing, other built structures, and smaller roads
- Since the scene is taken in June some of the crops present, corn, soybeans, are in early stages of growth with less than 5% coverage



220 bands

Groundtruth classes for the Indian Pines scene and their respective samples number

#	Class	Samples
1	Alfalfa	46
2	Corn-notill	1428
3	Corn-mintill	830
4	Corn	237
5	Grass-pasture	483
6	Grass-trees	730
7	Grass-pasture-mowed	28
8	Hay-windrowed	478
9	Oats	20
10	Soybean-notill	972
11	Soybean-mintill	2455
12	Soybean-clean	593
13	Wheat	205
14	Woods	1265
15	Buildings-Grass-Trees-Drives	386
16	Stone-Steel-Towers	93

[8] Hyperspectral Images

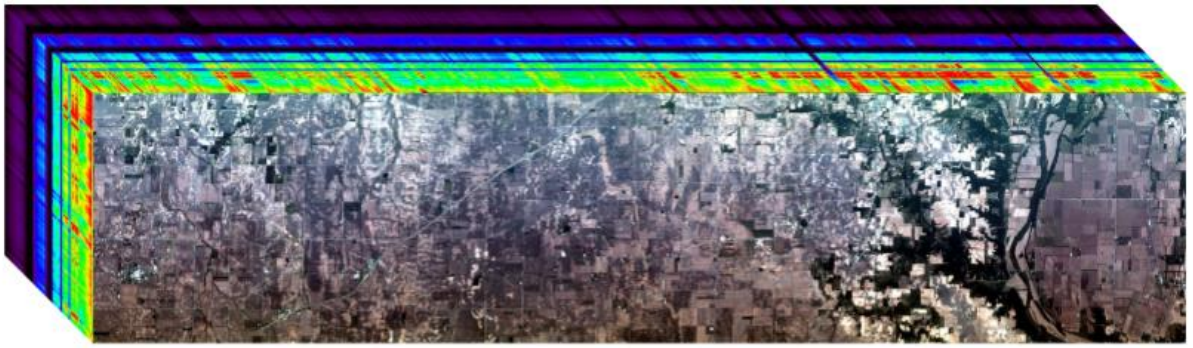
[7] Rasti Behnood, et al.

Competition

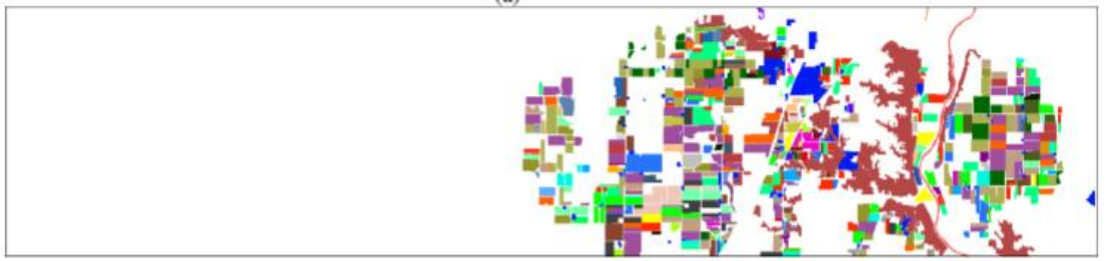
Group Name	Test Accuracy

Group Name	Test Accuracy

Indian Pine Dataset (large site)



(a)

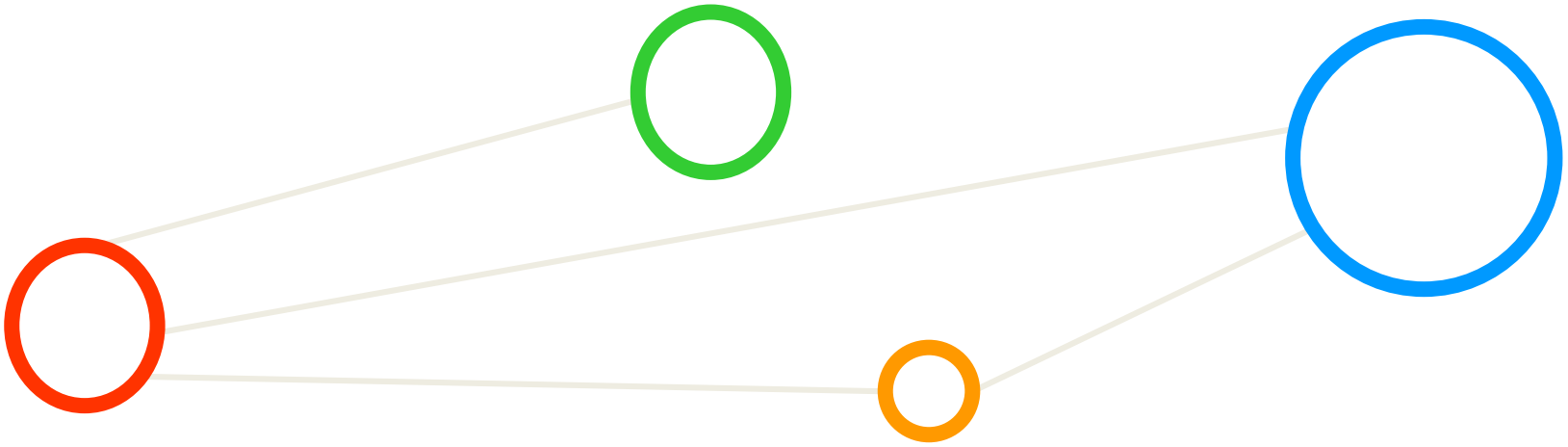


(b)

Thematic classes:

BareSoil (57)	Corn-MinTill (1049)	Hay (1128)	Soybeans-NS (1110)	Soybeans-NaTill (2157)
Buildings (17195)	Corn-MinTill-EW (5629)	Hay (pre.) (2185)	Soybeans-CleanTill (5074)	Soybeans-NaTill-EW (2533)
Concrete/Asphalt (69)	Corn-MinTill-NS (8862)	Hay-Alfalfa (2258)	Soybeans-CleanTill (pre.) (2726)	Soybeans-NaTill-NS (929)
Corn (17783)	Corn-Natill (4381)	Lake (224)	Soybeans-CleanTill-EW (11802)	Soybeans-NaTill-Drilled (8731)
Corn (presumed) (158)	Corn-Natill-EW (1206)	NotCropped (1940)	Soybeans-CleanTill-NS (10387)	Swampy Area (583)
Corn-EW (514)	Corn-Natill-NS (5685)	Oats (1742)	Soybeans-CleanTill-Drilled (2242)	River (3110)
Corn-NS (2356)	Fescue (114)	Oats (pre.) (335)	Soybeans-CleanTill-Weedy (543)	Trees (pre.) (580)
Corn-CleanTill (12404)	Grass (1147)	Orchard (39)	Soybeans-MinTill (15118)	Wheat (4979)
Corn-CleanTill-EW (26486)	Grass/Trees (2331)	Pasture (10386)	Soybeans-Drilled (2667)	Woods (63562)
Corn-CleanTill-NS (39678)	Grass/Pasture-mowed (19)	Pond (102)	Soybeans-MinTill (1832)	Unknown (144)
Corn-CleanTill-NS-Irrigated (800)	Grass/Pasture (73)	Soybeans (9391)	Soybeans-MinTill-Drilled (8098)	
Corn-CleanTill-NS (pre.) (1728)	Grass-runway (37)	Soybeans (pre.) (894)	Soybeans-MinTill-NS (4953)	

Lecture Bibliography



Lecture Bibliography (1)

- [1] Plotly Open Source Graphing Libraries
  Online: https://plot.ly/graphing-libraries/
- [2] Sebastian Raschka, Principal Component Analysis in ipython-notebooks
  Online: https://github.com/plotly/documentation/blob/source-design-merge/_posts/ipython-notebooks/principal_component_analysis.ipynb
- [1] UCI Machine Learning Repository Iris Dataset,
  Online: https://archive.ics.uci.edu/ml/datasets/Iris/
- [2] What are Eigenvectors and Eigenvalues?
  Online: http://www.visiondummy.com/2014/03/eigenvalues-eigenvectors/
- [3] Eigenvectors and Eigenvalues, Essence of Linear Algebra, Chapter 13
  Online: https://www.youtube.com/watch?v=PFDu9oVAE-g&t=684s
- [4] Preprocessing Data
  Online: https://scikit-learn.org/stable/modules/preprocessing.html
- [5] A Geometric Interpretation of the Covariance Matrix
  Online: http://www.visiondummy.com/2014/04/geometric-interpretation-covariance-matrix/
- [6] Singular Value Decomposition (SVD)
  Online: https://en.wikipedia.org/wiki/Singular_value_decomposition
- [7] Rasti, Behnood, "Sparse Hyperspectral Image Modeling and Restoration", 10.13140/RG.2.1.4097.4247, 2014.
  Online: https://www.researchgate.net/publication/277004219_Sparse_Hyperspectral_Image_Modeling_and_Restoration
- [8] Hyperspectral Images
  Online: https://engineering.purdue.edu/~biehl/MultiSpec/hyperspectral.html


services
measurement
within
fusion
new
funding
Policy-based
concepts
device
Analysis
forms
cross-disciplinary
resources
Centre
climate
Computer
expertise
year
like
Computational
Test
E.g
Computing
disciplines
results
Approaches
construction
Fusion
Enable
Cross-Disciplinary
methods
High
big
nodes
PB
community
Environment
Modelling
different
EU
Science
centers
Key
scientific
important
files
model
etc
DKRZ
Use
storage
basis
Infrastructure
international
simulations
ENES
technologies
formats
physical
engineering
References
brain
increasing
via
EUDAT
IPCC
SMAQ
Big
Project
Data
processing
Common
TB
Methods
computing
Using
European
MapReduce
create
databases
GB
Supercomputing
Work
images
codes
analysis
Large
Summary
data-intensive
Service
Scientific
levels
often
use
DLCL
stored
Resources
analyze
external
JSC
computational
Level
Research
Matters
Concrete
performance
research
e.g
science
HPC
large-scale
Juelich
system
datasets
acquisition
compute
Landscape
management
System
Climate
modelling
AR5
Hardware
cores
access
hundreds
Nutshell
Services
distributed
current
network
DLCLs
Understanding
structures
Technologies
Earth
Structure
Germany
Simulation
directory
project
General
Provide
NASA
Energy
systems
day
Processing
Health