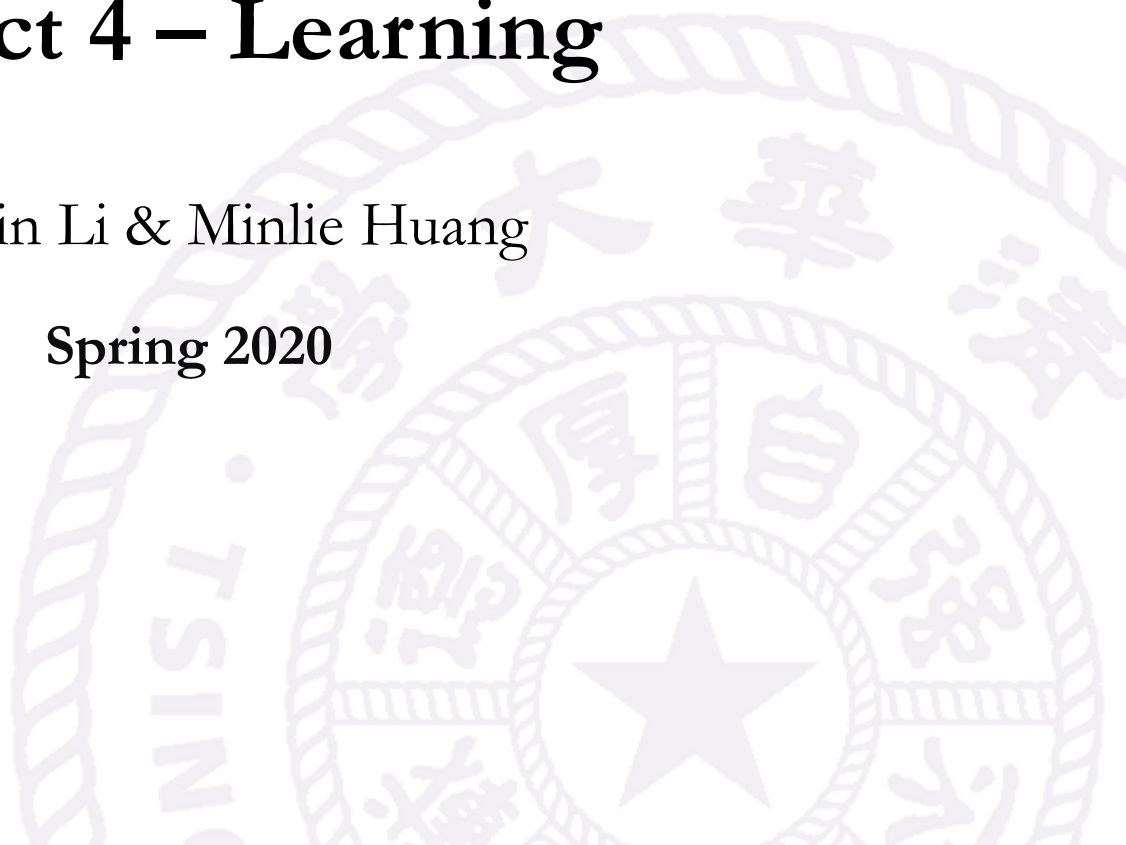


Introduction to Artificial Intelligence

Project 4 – Learning

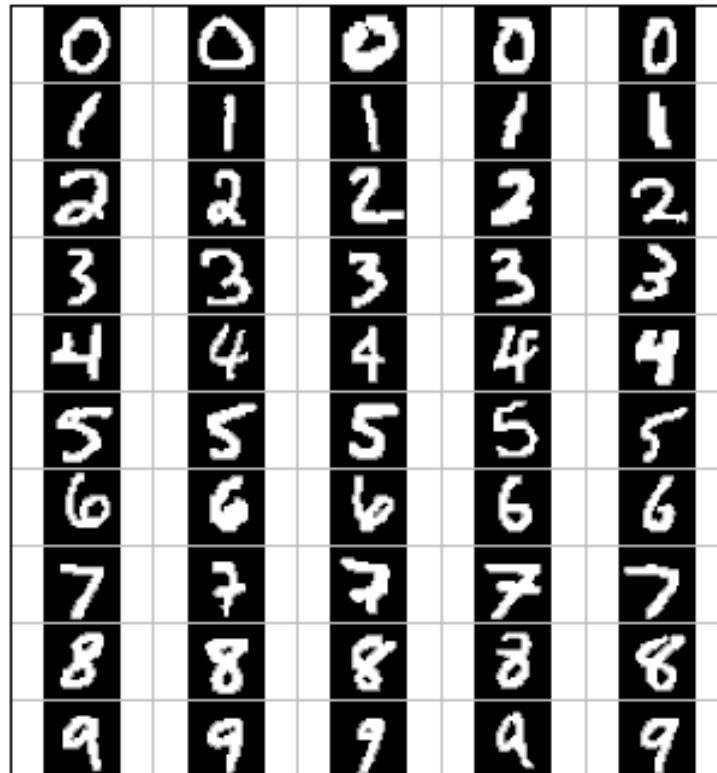
Jianmin Li & Minlie Huang

Spring 2020



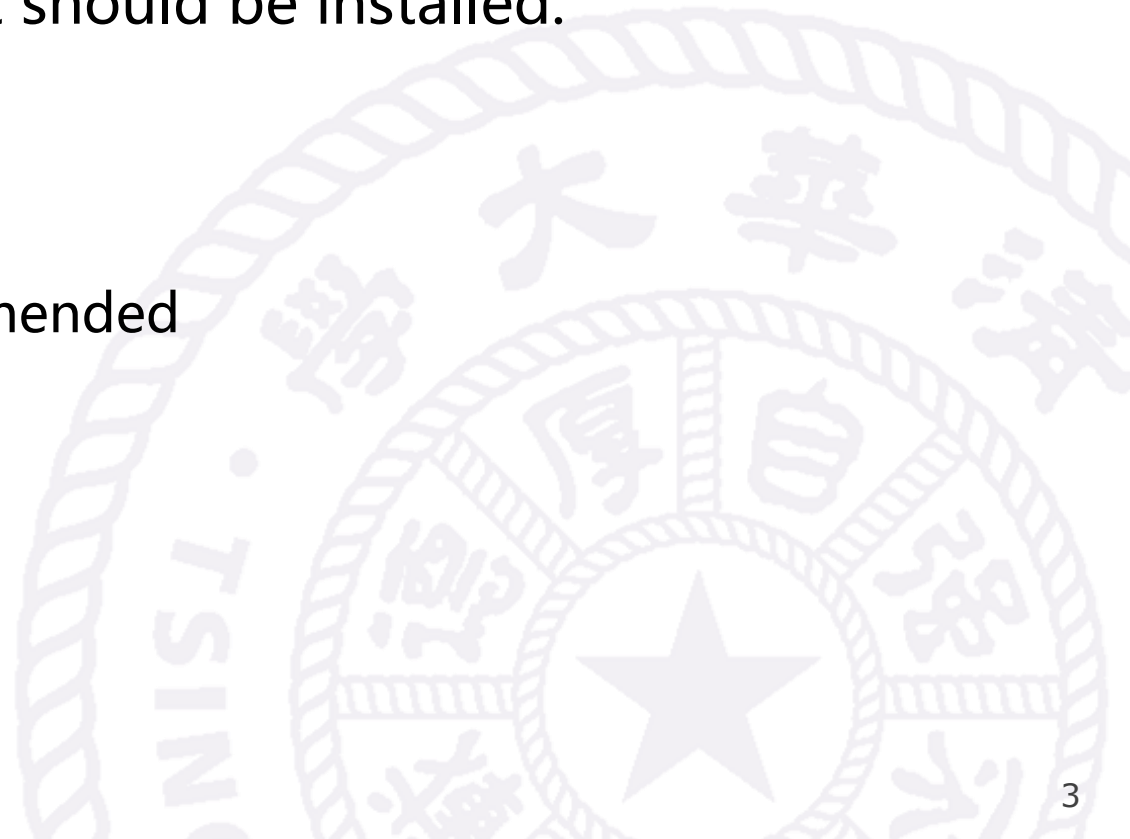
Supervised and Unsupervised Learning

- Training with MNIST



Pre-requirements

- At least 400MB disk space and 800MB memory
- Python package that you should be familiar with
 - *numpy*
 - *sklearn*
- Python packages that should be installed:
 - *numpy*
 - *skimage*
 - *sklearn*
 - Anaconda is recommended



Basic Tasks

- K-Means (4 points)
 - Implement `KMeansCluster.fit` in `featureExtractor.py`
 - `python featureExtractor -f kmeans -s 10`
- KNN (3 points)
 - Implement `KNNClassifier.classify` method in `classifiers.py`
 - `python dataClassifier.py -c knn -n 5`
- Softmax Regression (4 points)
 - Implement `PerceptronClassifier.train` in `classifiers.py`
 - `python dataClassifier.py -c perceptron`
- sklearn **MUST NOT BE USED** in the above two tasks, **OR** you will not pass the autograder.

Basic Tasks

- Training SVM with sklearn (2 points)
 - Implement `SVMClassifier.train`, `SVMClassifier.classify` using package `sklearn`, in `classifiers.py`
 - You should be familiar with some sklearn API
- Obtaining better classification results (2 points)
 - Implement *`BetterClassifier.train`*, *`BetterClassifier.classify`* in *`classifiers.py`*
 - You may make use of `sklearn` package
 - Try to obtain good accuracy as much as you can.

Submission

- A 1-3 pages report (either Chinese or English)
 - You MUST answer **Question 1** in **YOUR REPORT**
 - You will not get full report credits if cannot answer the above questions correctly
 - Some analysis on different algorithms/feature extractor techniques is useful for better grading
- Zip the files as the following structure
 - student_id.zip (e.g. 20090112xx.zip)
 - student_id.pdf
 - classifiers.py
 - featureExtractor.py

Grading

- Due
 - 2019/5/17 23:59:59
- Correctness of algorithms (80%)
- Report (20%)
 - You MUST answer Question 1 in the report

