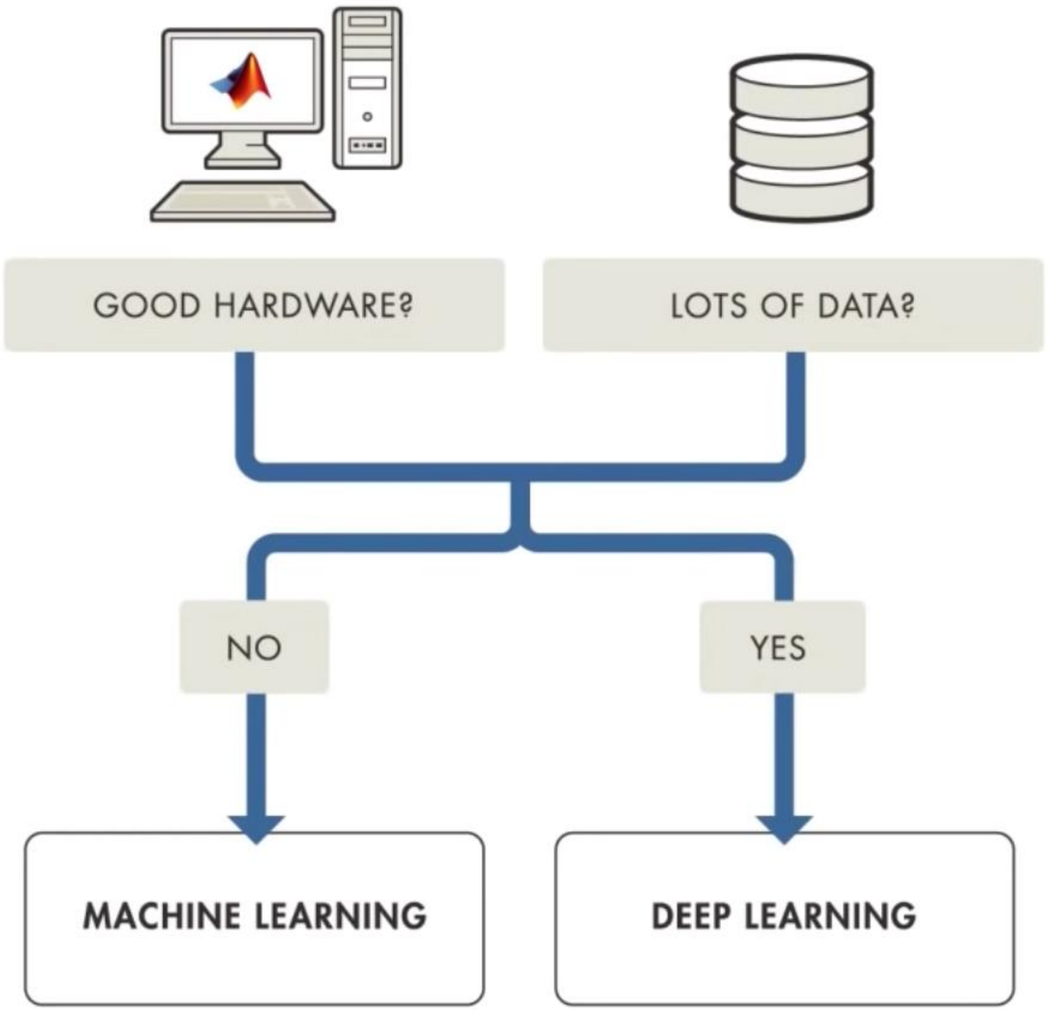
# Machine Learning

## Machine Learning vs. Deep Learning RECAP

Cigdem Beyan

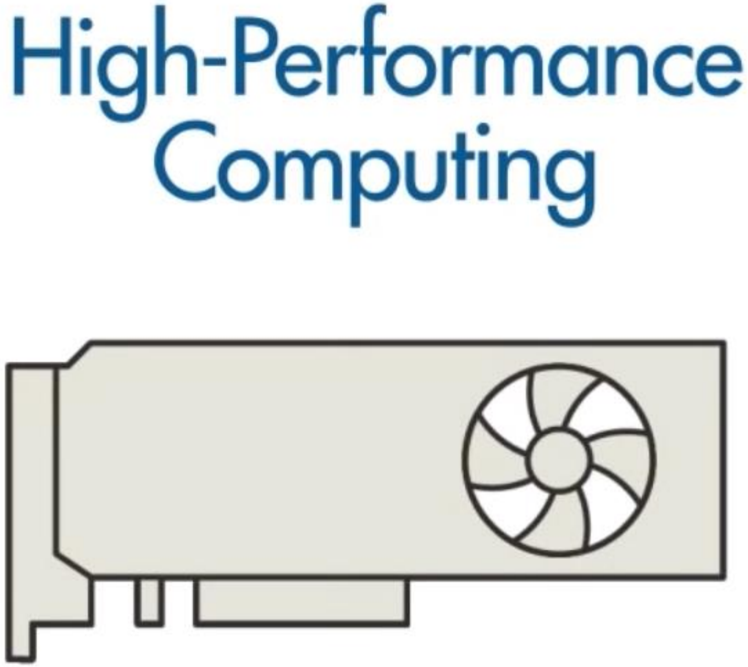
A. Y. 2024/2025

# ML vs DL




Credits to [Matlab tutorial on ML and Deep Learning]

# ML vs DL





Credits to [Matlab tutorial on ML and Deep Learning]

# ML vs DL (In a nutshell)

| | Machine Learning | Deep Learning |
|---|---|---|
| Training dataset | Small | Large |
| Choose your own features | Yes | No |
| # of classifiers available | Many | Few |
| Training time | Short | Long |

Credits to [Matlab tutorial on ML and Deep Learning]

# ML vs DL (In Detail)

- **Data Dependency**
  - **ML:** Performs well with smaller datasets (100s to 1000s).
  - **DL:** Requires larger datasets (tens of thousands to millions of samples) due to high complexity and vast number of parameters
- **Feature engineering**
  - **ML:** Relies on manual feature extraction and selection. When there is the ability to craft meaningful features using domain knowledge.
  - **DL:** Automatically performs feature extraction vis hidden layers in neural networks, with no need for manual feature engineering. Better in complex data such as images, videos or audio.

# ML vs DL (In Detail)

- **Computational resources**
  - **ML:** Less computational intensive. Can run efficiently on CPUs
  - **DL:** Highly resource-intensive, requires GPUs
- **Model Interpretability**
  - **ML:** Often interpretable such as decision trees, linear regression, or logistic regression.
  - **DL:** Often Black box. Requires other techniques like LIME, and SHAP to perform interpretability.
- **Training time**
  - **ML:** Faster to train
  - **DL:** Slower to train.

# ML vs DL (In Detail)

- **Problem Complexity**
  - **ML:** To be used for problems that do not require complex representations.
  - **DL:** Good for complex data structures where hierarchical or abstract feature representation is necessary
- **Flexibilitiy**
  - **ML:** more flexible across a wider range of tasks. With a minor adjustment, the same algorithm can be applied to regression, and classification.
  - **DL:** Specialized in specific architectures tailored to the task at hand. E.g., CNNs for image data, RNNs or Transformers for sequential data.

# ML vs DL (In Detail)

- **Scalability**
  - **ML:** Struggle with scalability. When the data size grows significantly, training time may increase disproportionately
  - **DL:** The performance of deep models improves significantly with more data

# RECAP

# What have we seen?

- A structured presentation (not in the order of our course schedule)

- **Basics:** Definition of machine learning, types of machine learning (supervised, unsupervised), applications, error function, regularization, overfitting/underfitting

- **Data cleaning:** Categorical data to numeric data, missing values, data cleaning, outlier removal, feature normalization, dimensionality reduction (e.g., PCA), feature selection (regression-based, forward feature selection)

- **Model evaluation:** Train/validation/test split, cross-validation, metrics for regression, classification, clustering, grid search for parameter selection

# What have we seen?

- **Supervised Learning (Linear Models):** linear regression, L1/L2 regression, logistic regression, linear SVM, perceptron, etc.

- **Supervised Learning (Instance-based):** K-NN, distance metrics, and their impact on performance

- **Supervised Learning (Decision trees) and Ensemble Learning:** CART algorithm, pruning, entropy, information gain, random forest, bagging, boosting, AdaBoost

- **Support Vector Machines:** Large Margin classifiers, SVM for binary classification, kernel trick for non-linear classification, regularization, and the role of C parameter

# What have we seen?

- **Unsupervised Learning – Clustering:** K-means, hierarchical clustering, DBSCAN, etc. evaluation metrics (silhouette score, Davies-Bouldin index…)

- **Unsupervised Learning – Dimensionality Reduction:** PCA, Fisher Discriminant, t-SNE

- **Probabilities Models:** Naïve Bayes, Gaussian Mixture Models, Hidden Markov Models


Cigdem Beyan
cigdem.beyan@univr.it
https://cbeyan.github.io/