Lecture 2: Image Classification & Linear Models

Areej Alasiry

CIS 6217 – Computer Vision for Data Representation

College of Computer Science, King Khalid University

Learning Outcomes

- Explain the data-driven approach to computer vision.
- Implement simple image classification methods (KNN, Linear Classifiers).
- Understand the limitations of basic models compared to deep learning.
- Build intuition about loss functions and decision boundaries.

What is Image Classification?

- Assigning a label y to an image x.
- Examples: digit recognition (MNIST), object recognition (CIFAR-10, ImageNet).

The Data-Driven Approach

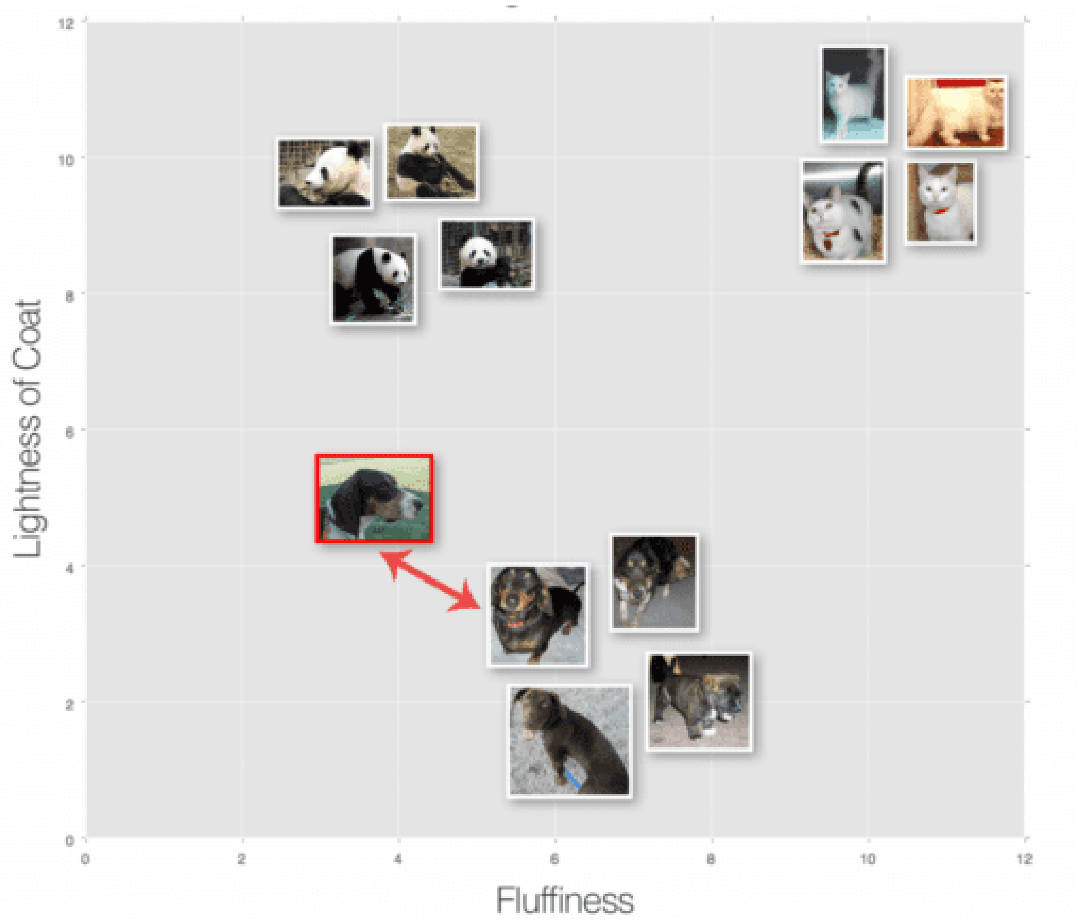
- Traditional CV: hand-crafted features (SIFT, HoG).
- Data-driven: learn features and classifiers directly from data.
- Key enablers: large datasets and computational power.

Nearest Neighbour Classifier

K-Nearest Neighbour is a supervised machine learning algorithm

Nearest Neighbor Classifier

- K-Nearest Neighbor (KNN): classification by comparing distances in feature space.
- Decision depends on the majority class of nearest neighbors.
- Non-parametric, simple, intuitive.



Source: https://url-shortener.me/4GE0

Distance Metrics

- L1 distance (Manhattan): sum of absolute differences.

$$d_1(I_1, I_2) = \sum p|Ip_1 \; - \; Ip_2|$$

- L2 distance (Euclidean): square root of squared differences.

$$d_2(I_1, I_2) = \sqrt{\sum_p (Ip_1 \; - \; Ip_2)^2}$$

- Choice of metric impacts classifier performance.

Illustration

$$
\begin{array}{c} \text{test image} \\ \left|\begin{array}{|c|c|c|c|} \hline 56 & 32 & 10 & 18 \\ \hline 90 & 23 & 128 & 133 \\ \hline 24 & 26 & 178 & 200 \\ \hline 2 & 0 & 255 & 220 \\ \hline \end{array}\right. \end{array}
-
\begin{array}{c} \text{training image} \\ \left.\begin{array}{|c|c|c|c|} \hline 10 & 20 & 24 & 17 \\ \hline 8 & 10 & 89 & 100 \\ \hline 12 & 16 & 178 & 170 \\ \hline 4 & 32 & 233 & 112 \\ \hline \end{array}\right| \end{array}
=
\begin{array}{c} \text{pixel-wise absolute value differences} \\ \begin{array}{|c|c|c|c|} \hline 46 & 12 & 14 & 1 \\ \hline 82 & 13 & 39 & 33 \\ \hline 12 & 10 & 0 & 30 \\ \hline 2 & 32 & 22 & 108 \\ \hline \end{array} \end{array}
\rightarrow 456
$$

Bir, P. (2019) *Image classification with K nearest neighbours*, *Medium*. Available at: https://medium.com/swlh/image-classification-with-k-nearest-neighbours-51b3a289280 (Accessed: 08 September 2025).

Strengths & Weaknesses of KNN

- ✓ Easy to implement and understand.
- ✓ No training phase, only prediction.
- ✗ High memory usage (store all data).
- ✗ Slow predictions for large datasets.
- ✗ Poor generalization to unseen data.

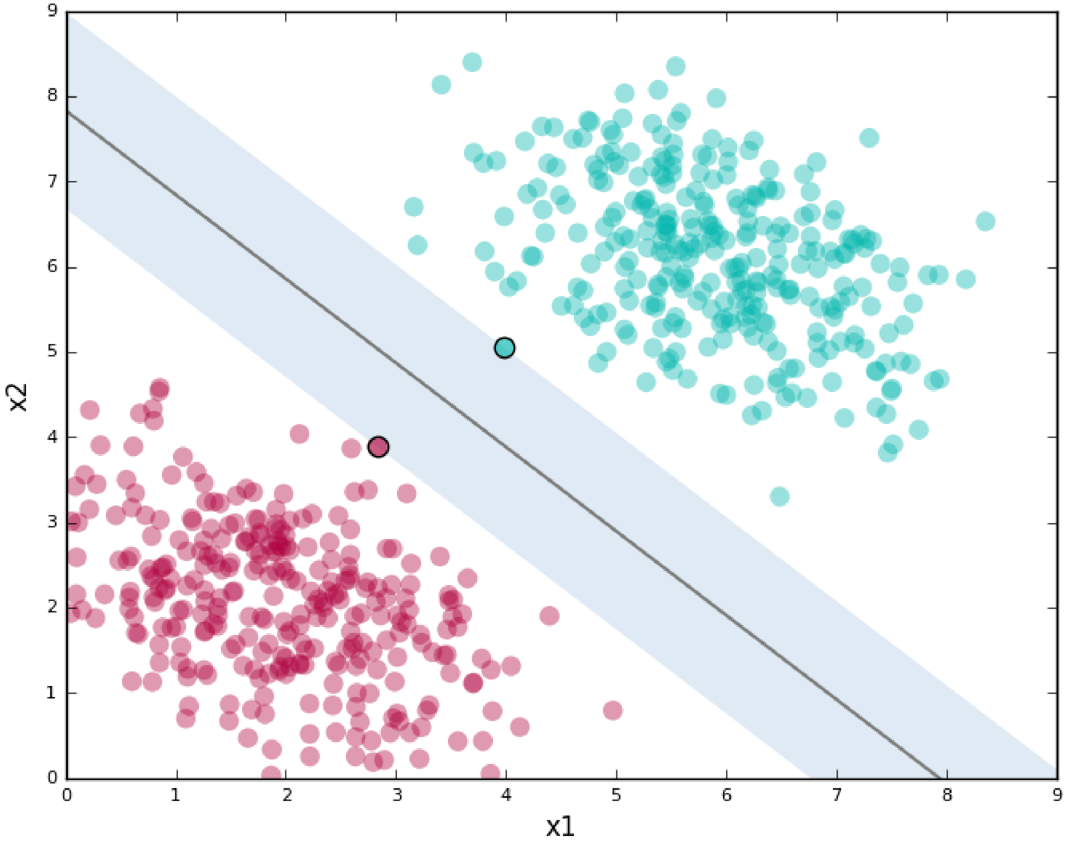
Linear Classification I

Linear Classification: Concept

- Classifier defines a linear decision boundary (hyperplane).
- Equation: $f(x;\, W, b) \;= wx \;+\; b$.
- Separates classes in feature space.

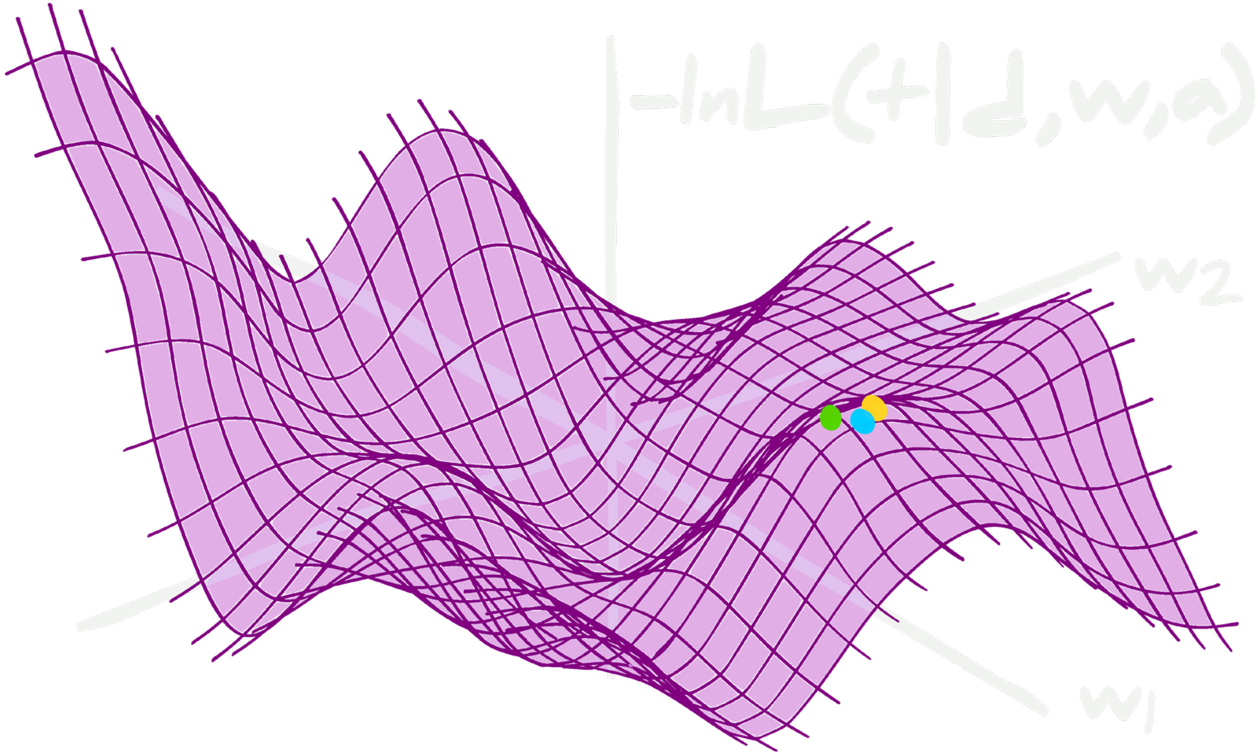
Decision Boundaries

- 2D example: line separating two classes.
- Higher dimensions: hyperplanes.
- Linear classifiers may fail on non-linear datasets.



Gradient Descent

- Try new weights and biases until it reaches its goal of finding the optimal values for the model to make accurate predictions.
- Loss measures how well predictions match labels.
  - Difference between prediction and actual ground-truth



KNN vs Linear Models

- KNN: simple, memory-intensive, slow on large data.
- Linear Models: efficient, generalize better, scalable.
- Linear models form the foundation for neural networks.

Limitations of Basic Models

- KNN struggles with high-dimensional data.
- Linear models cannot capture non-linear boundaries.
- Need for more complex models → Neural Networks.

Hands-on Activity

- Implement KNN on CIFAR-10 using NumPy.
- Train a logistic regression classifier with SGD.
- Visualize decision boundaries in 2D data.
- Compare performance of KNN vs Linear classifier.

Python Example

- Use scikit-learn for KNN.
- Evaluate accuracy on digit recognition dataset.
- Visualize decision boundaries.

Summary

- Image classification assigns labels to images.
- KNN: simple, interpretable, but not scalable.
- Linear models: efficient, scalable, limited by linearity.
- Loss functions guide optimization (hinge, cross-entropy).
- Motivates transition to neural networks in next lectures.

References

- Computer Vision: A Modern Approach – Forsyth & Ponce (2010)