Comparison of Object Detection by Linear Classifier and Locality-Sensitive Hashing (LSH)

Lun Jiang, Applied Mathematics, University of Washington

COCO Dataset

COCO is a large-scale object detection, segmentation, and captioning dataset.

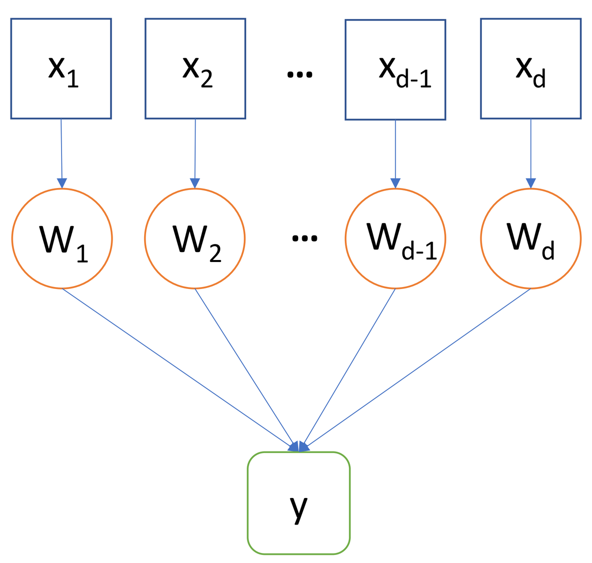
- 18 categories of COCO Dataset
- *Tiny* and *Small* Dataset
- *Tiny* (dim = 1472) and *Small* (dim = 11776) feature vectors.

Linear Regression Model

- Loss Function of Linear Model

$$L(w)=\frac{\lambda}{2}\|w\|^2+\frac{1}{n}\sum_{i=1}^{n}\frac{1}{2}\cdot(y_i-\langle w,x_i\rangle)^2$$

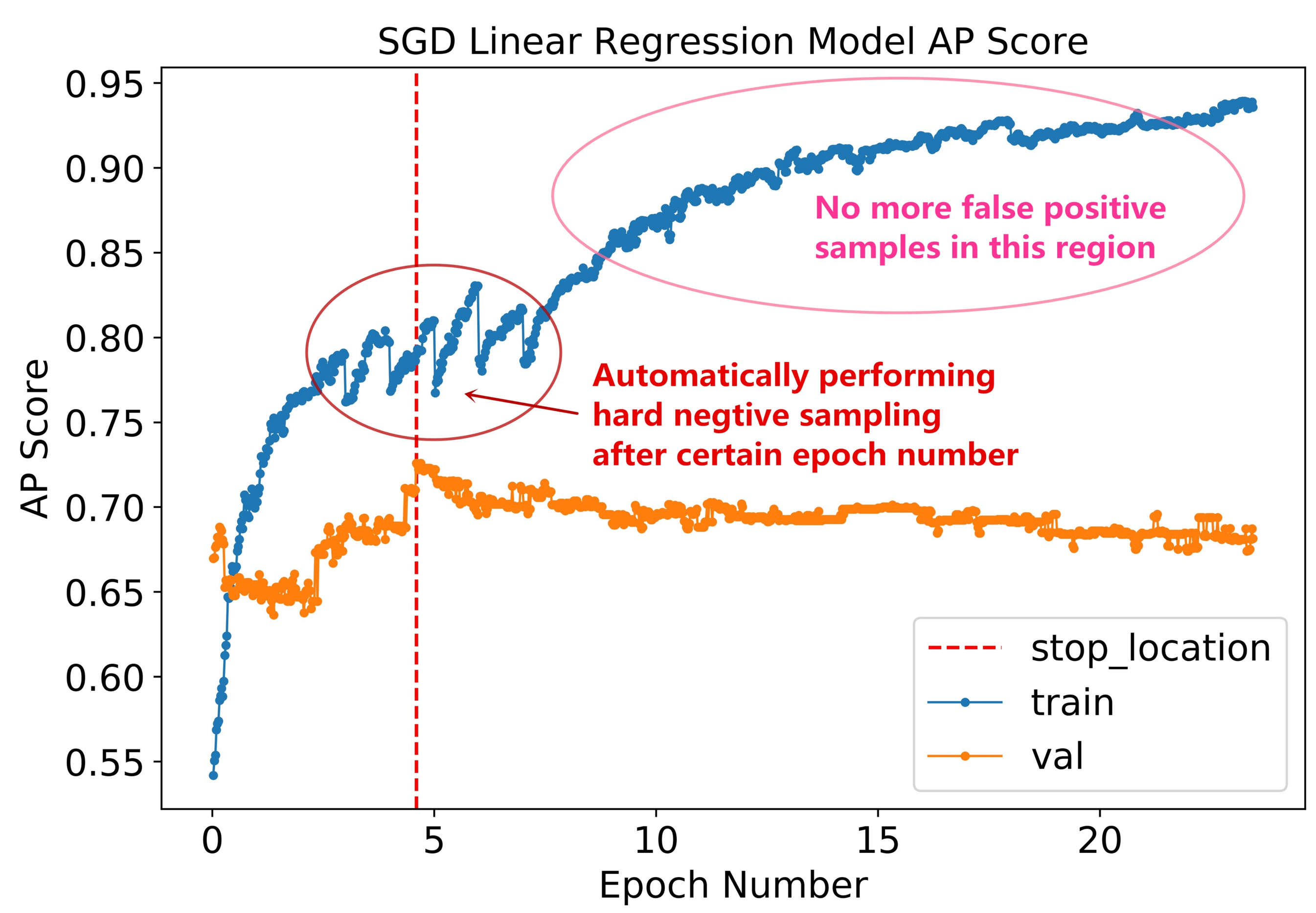
- Structure of One Layer Perceptron



- Perform **linear regression** for each category. The resulting AP score per category and the overall mAP are shown in table.

Category	AP Score
airplane	0.348313737
bear	0.094608968
bicycle	0.109999502
bird	0.103223281
boat	0.128537091
bus	0.249275943
car	0.141795107
cat	0.397089824
cow	0.186116586
dog	0.196843699
elephant	0.423391365
giraffe	0.520327058
horse	0.149371811
motorcycle	0.182861128
sheep	0.246667087
train	0.209196683
truck	0.121703411
zebra	0.444227604
mAP	**0.236308327**

Auto Hard Negative Mining



- **Hard Negative Mining** is used to reduce the false positive predictions by the model, which will be done automatically after certain epochs after convergence.

LSH Algorithm

- **Locality-Sensitive Hashing (LSH)** is an algorithm to reduce the dimensionality of high-dimensional data using hash functions mapping "similar" data into same buckets.
- **Random Projection** is one good way to construct LSH functions due to its distance preservation property*:* if two data points are close in the original space, then they are highly possible to be still close in the projected subspace, with probability given by the *Norm Preservation Theorem:*

$$P\left((1-\epsilon)\|x\|^2\le\|\phi(x)\|^2\le(1+\epsilon)\|x\|^2\right)\ge 1-2e^{-(\epsilon^2-\epsilon^3)m/4}$$

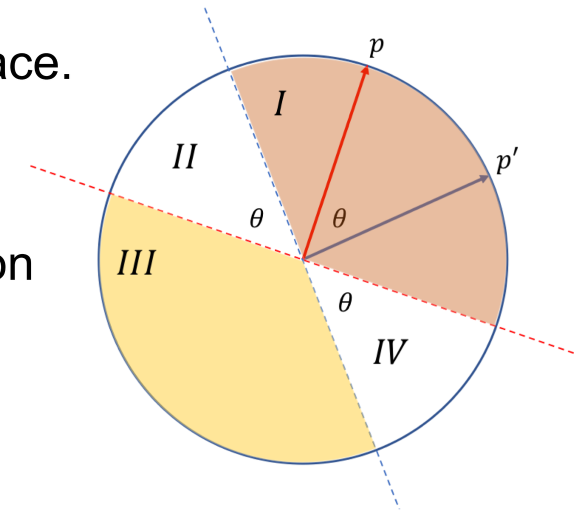
LSH Functions

- **Binary Random Projection:** projecting the high-dimensional data into Hamming space.

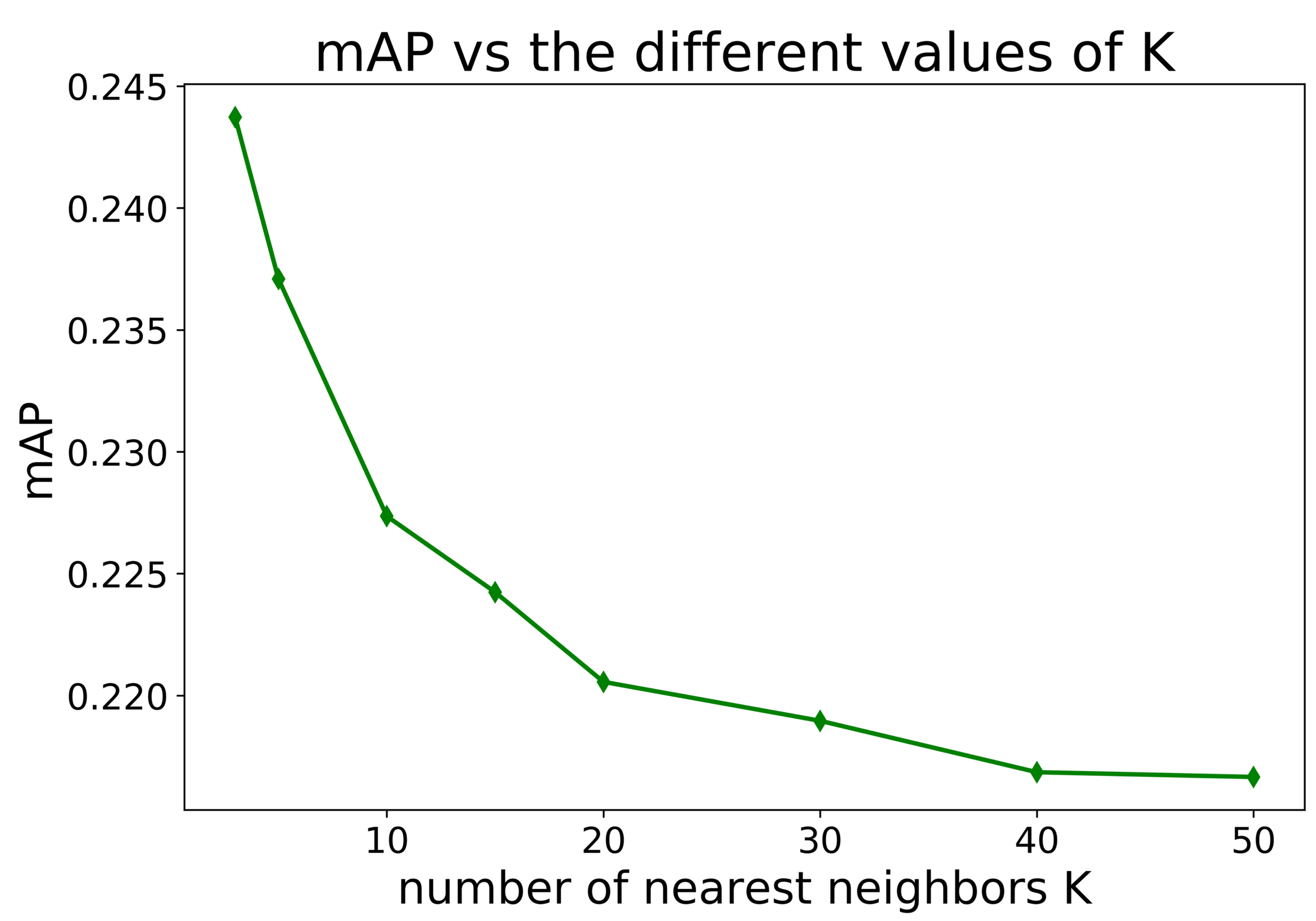
$$h(\mathbf{p})=sign(\mathbf{u}\cdot\mathbf{p})$$

- **e2LSH:** a family of LSH function based on Euclidean space, with p-stable property.

$$h(\mathbf{p})=\left\lfloor\frac{\mathbf{p}\cdot\mathbf{x}+b}{w}\right\rfloor$$

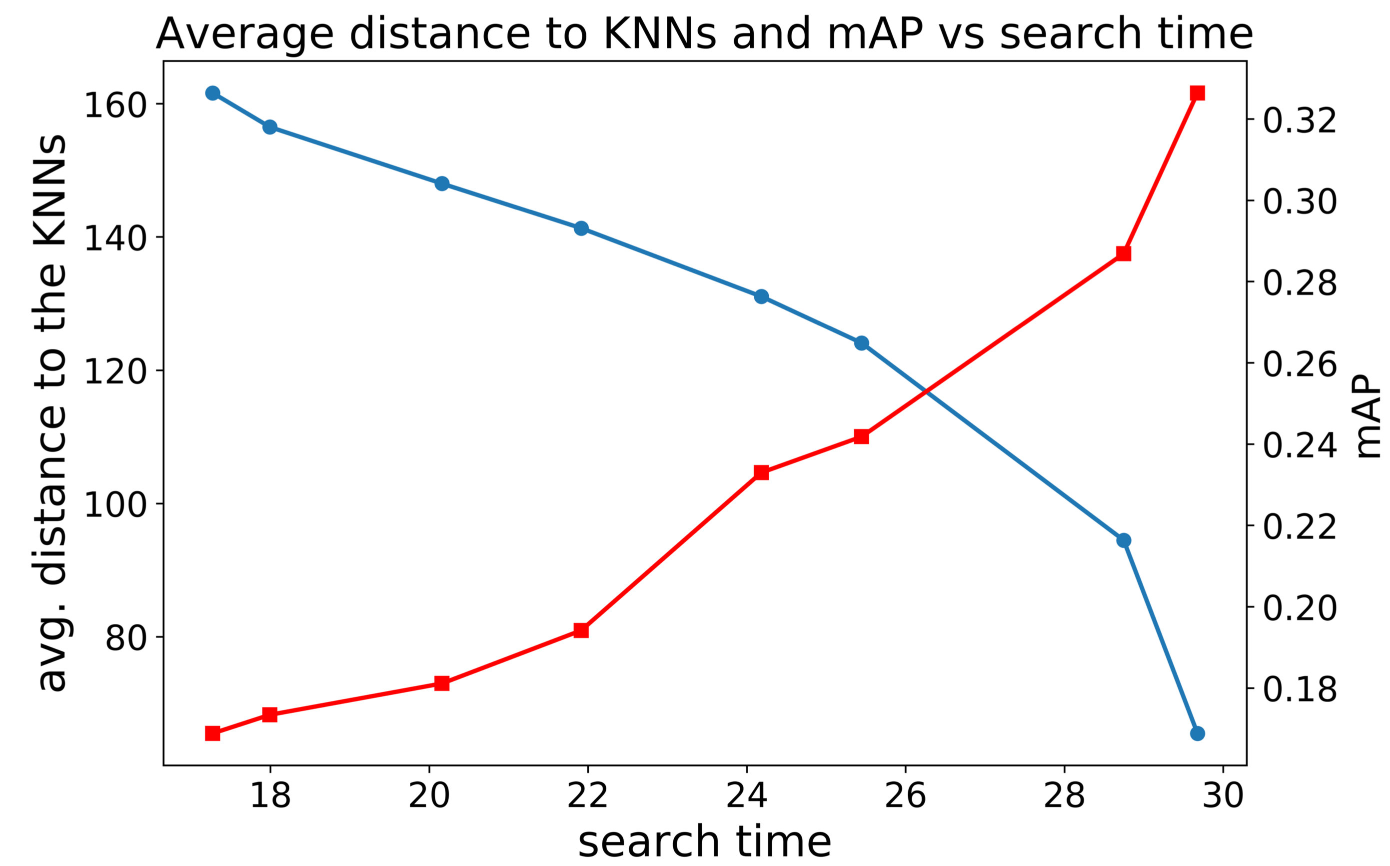


K Nearest Neighbors



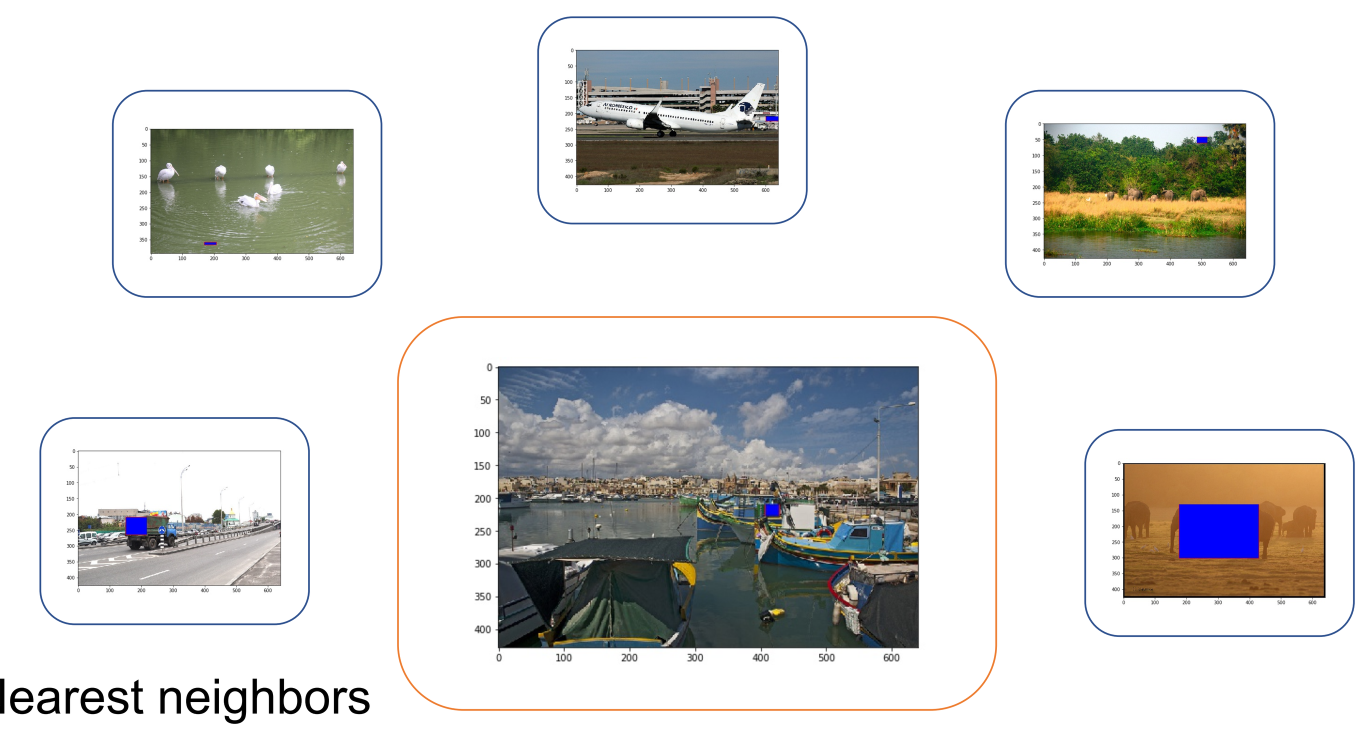
- **K-Nearest Neighbor** search is performed after adjacent data being projected into buckets of LSH tables. Increasing the number of nearest neighbors can cause decreasing of mAP, which may be due to the increasing distance to KNNs.

Speed-Quality Tradeoff



- There is a **tradeoff** between search speed and the quality of the search, as is shown on the figure above: Increasing the search time for a query yields higher mAP score and smaller distance to the K nearest neighbors, indicating better search quality but is also more time consuming.
- With high number K of nearest neighbors, the average distance to k nearest neighbors will be very large, which may lead to low AP performance. In this case, a **cutoff** radius cR is required for the algorithm to find the correct neighbors.

Real-Time Visualization



Nearest neighbors

Query Data Box

- **Real time visualization** is implemented in this code, enabling visual detection of the nearest neighbors of the querying data point, which takes the index of data in the query dataset and returns the boxed images of the corresponding k nearest neighbors of this data point.