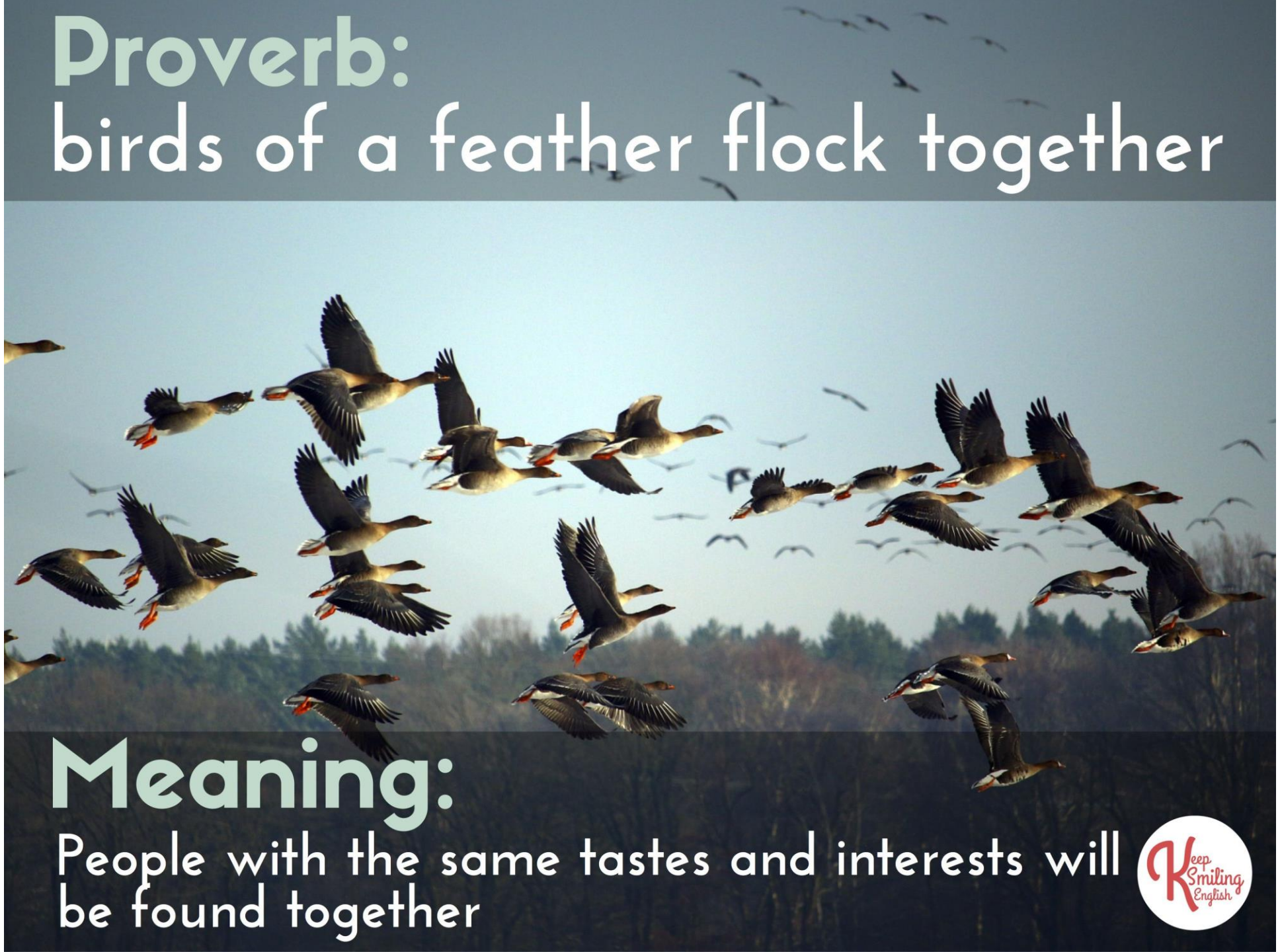
k-Nearest Neighbour

Proverb:
birds of a feather flock together

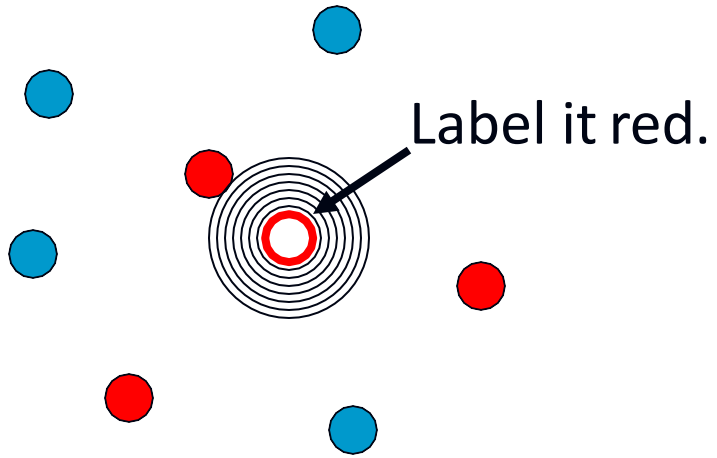


Meaning:
People with the same tastes and interests will be found together

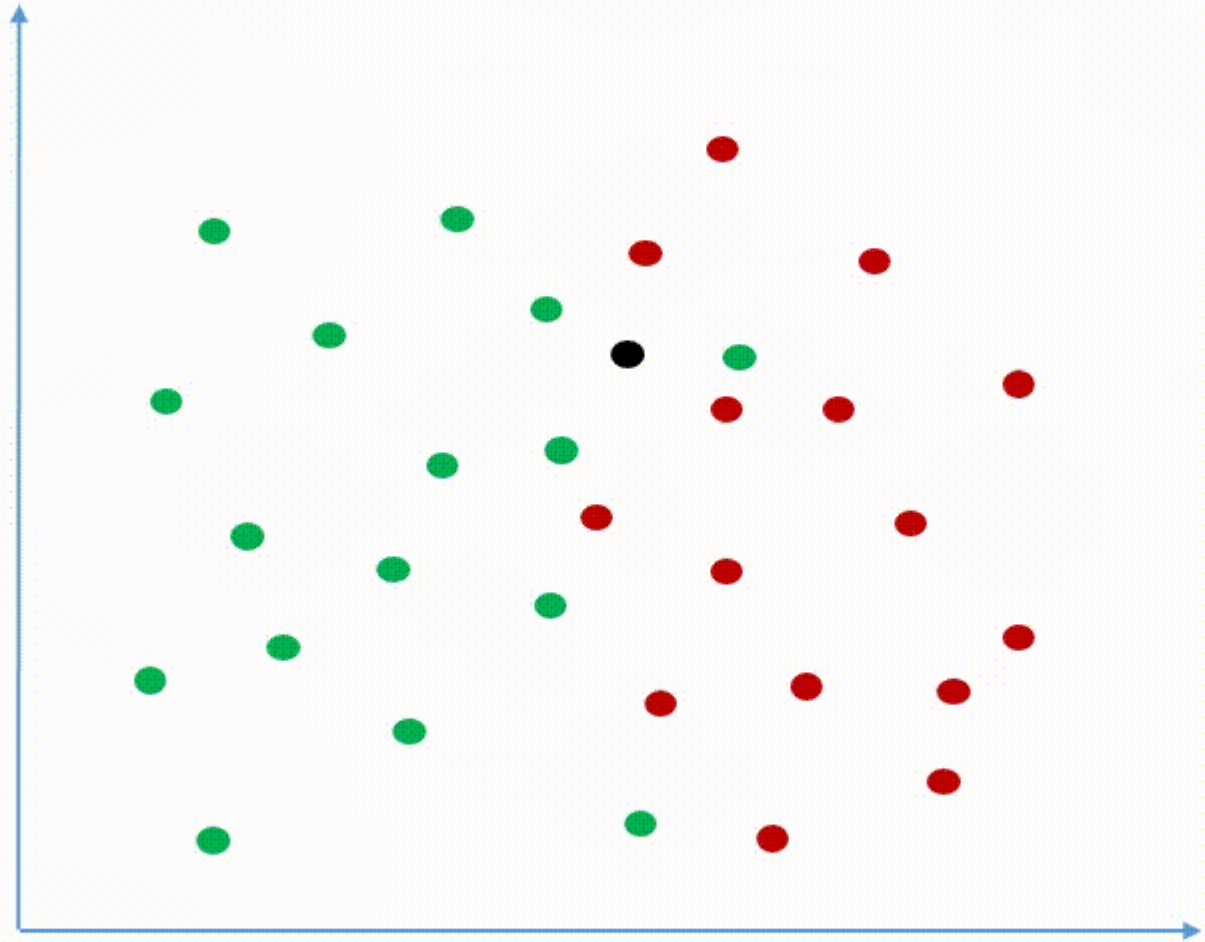
Keep Smiling English

1-Nearest Neighbor

- One of the simplest of all machine learning classifiers
- Simple idea: label a new point the same as the closest known point

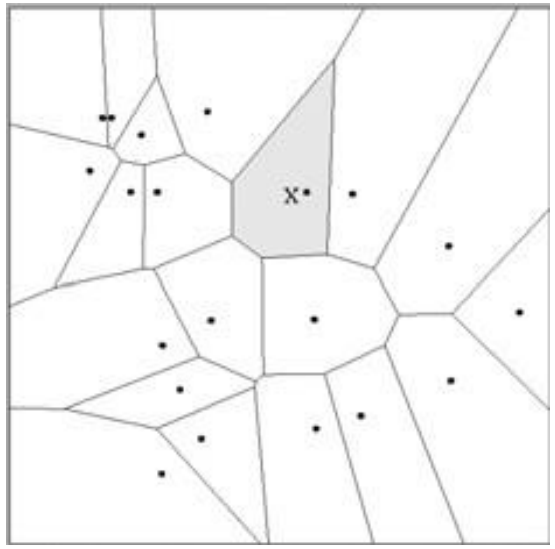


K-Nearest Neighbors Classification



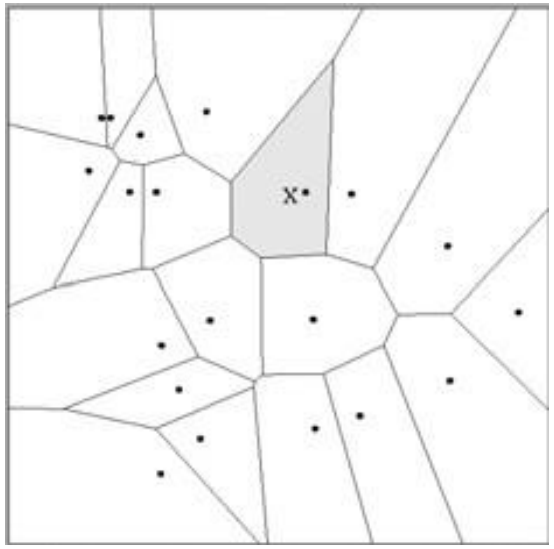
1-Nearest Neighbor

- A type of instance-based learning
  - Also known as “memory-based” learning
- Forms a Voronoi tessellation of the instance space

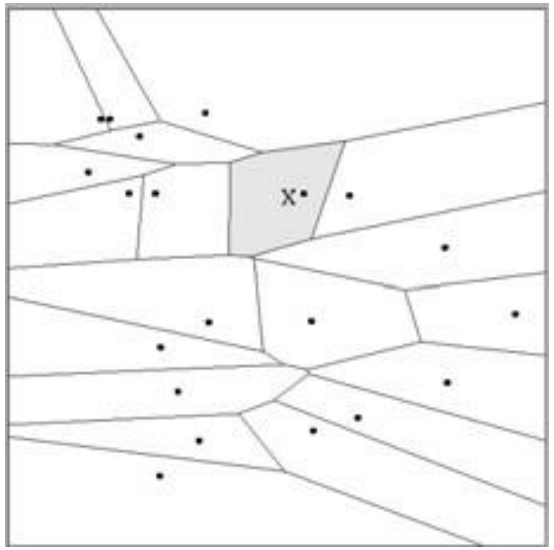


Distance Metrics

- Different metrics can change the decision surface



Dist(**a,b**) =$(a_1 - b_1)^2 + (a_2 - b_2)^2$



Dist(**a,b**) =$(a_1 - b_1)^2 + (3a_2 - 3b_2)^2$

- Standard Euclidean distance metric:
  - Two-dimensional: Dist(a,b) = sqrt($(a_1 - b_1)^2 + (a_2 - b_2)^2$)
  - Multivariate: Dist(a,b) = sqrt($\sum (a_i - b_i)^2$)

Adapted from "Instance-Based Lea4rning" lecture slides by Andrew Moore, CMU.

Four Aspects of an Instance-Based Learner:

1. A distance metric
2. How many nearby neighbors to look at?
3. A weighting function (optional)
4. How to fit with the local points?

Adapted from "Instance-Based Learning" lecture slides by Andrew Moore, CMU.

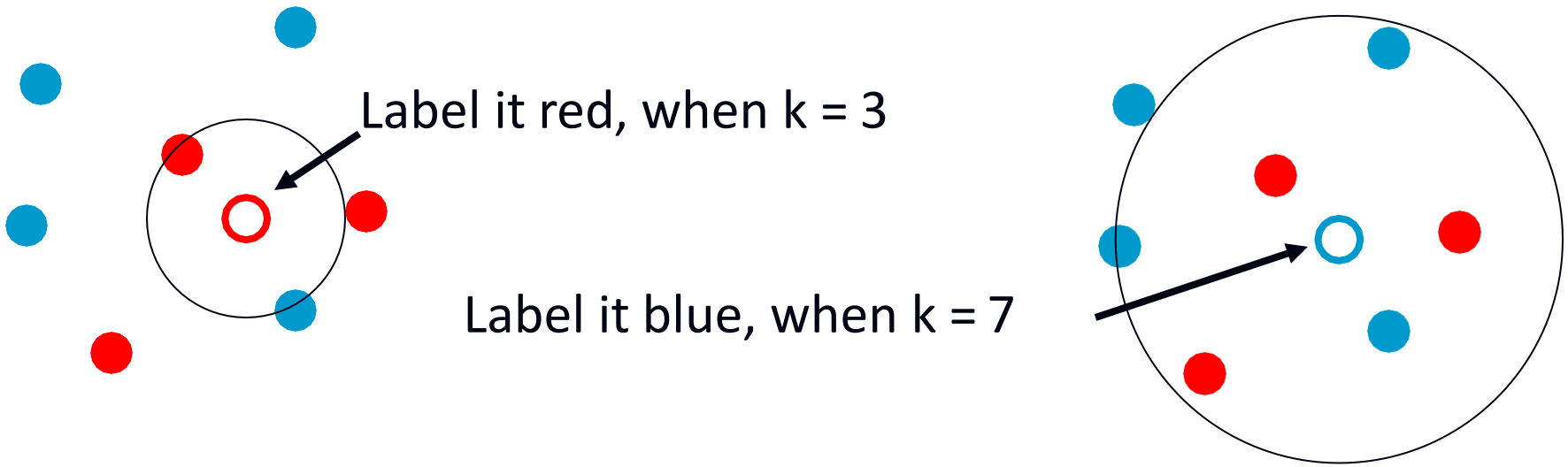
1-NN's Four Aspects as an Instance-Based Learner:

1. A distance metric
   - *Euclidian*
2. How many nearby neighbors to look at?
   - *One*
3. A weighting function (optional)
   - *Unused*
4. How to fit with the local points?
   - *Just predict the same output as the nearest neighbor.*

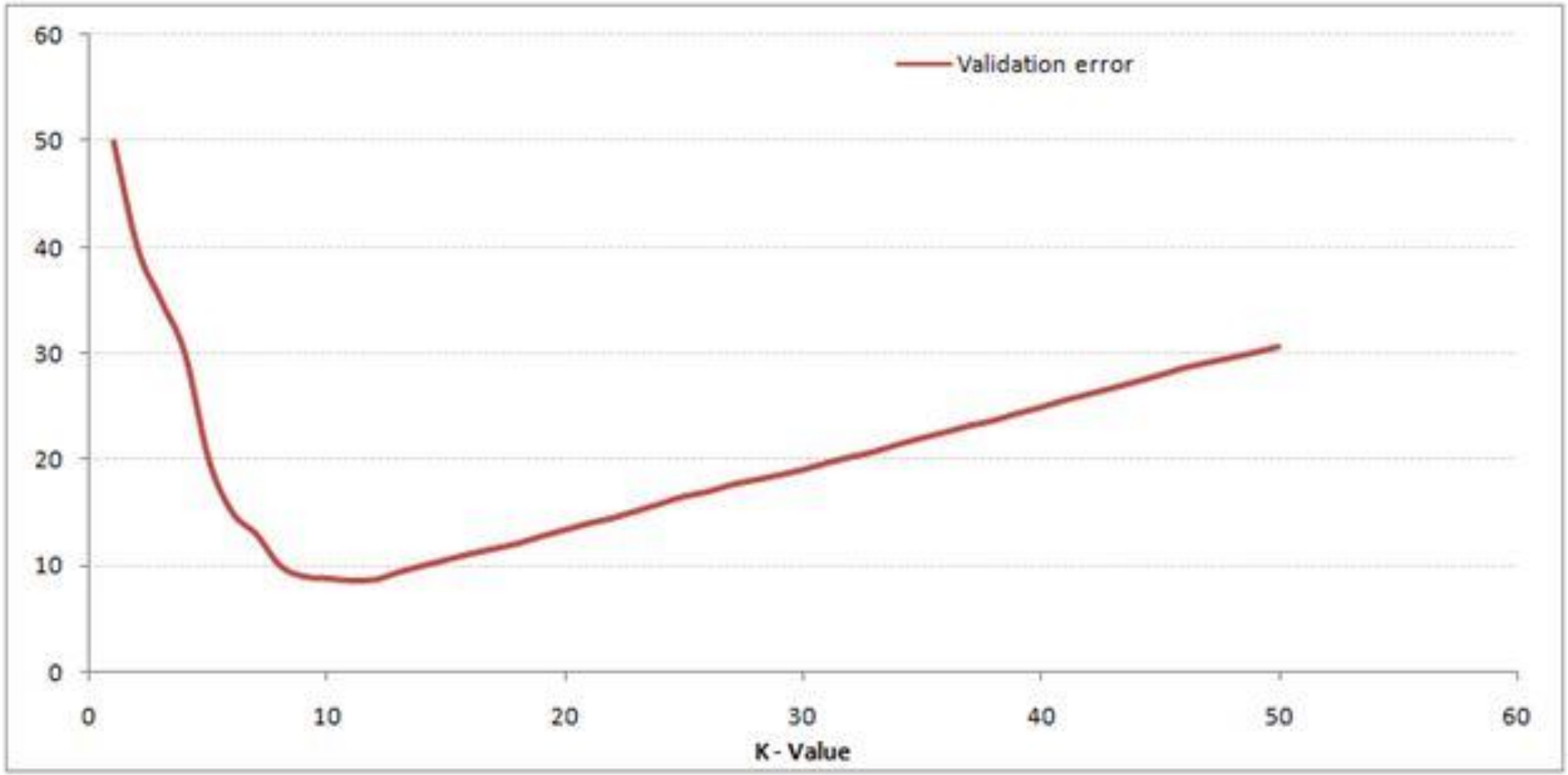
Adapted from "Instance-Based Learning" lecture slides by Andrew Moore, CMU.

k – Nearest Neighbor

- Generalizes 1-NN to smooth away noise in the labels
- A new point is now assigned the most frequent label of its *k* nearest neighbors

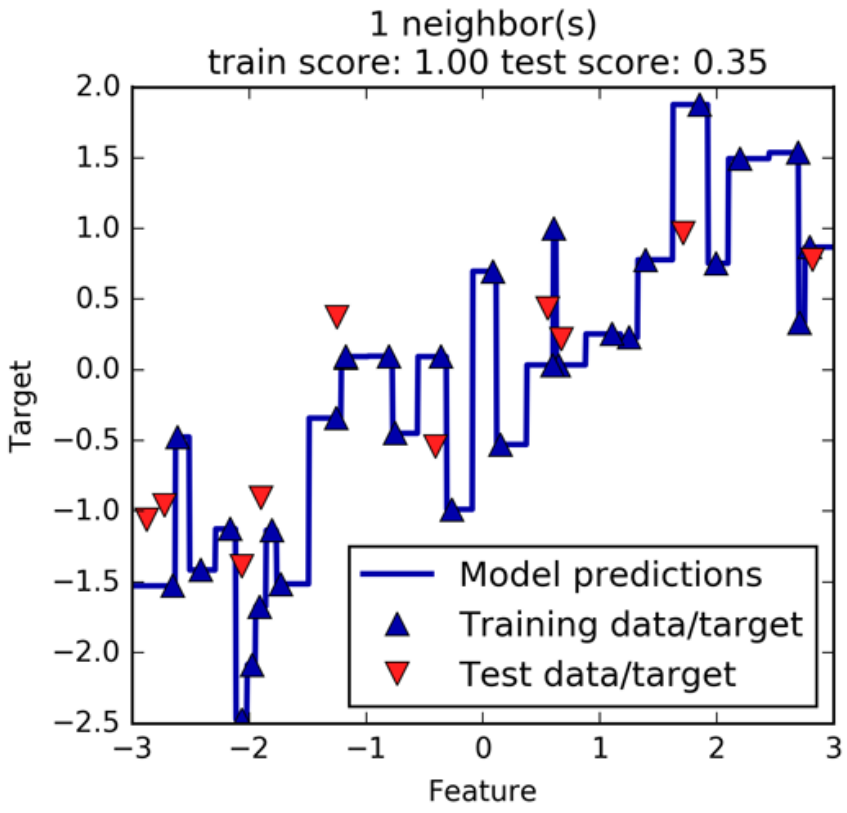


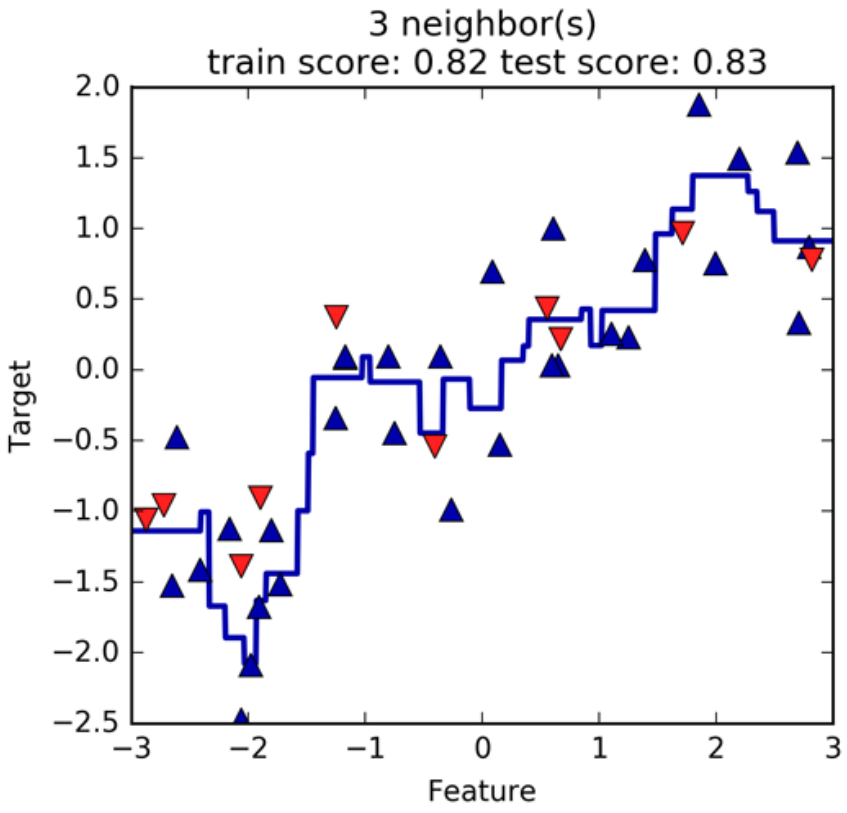
Choosing K

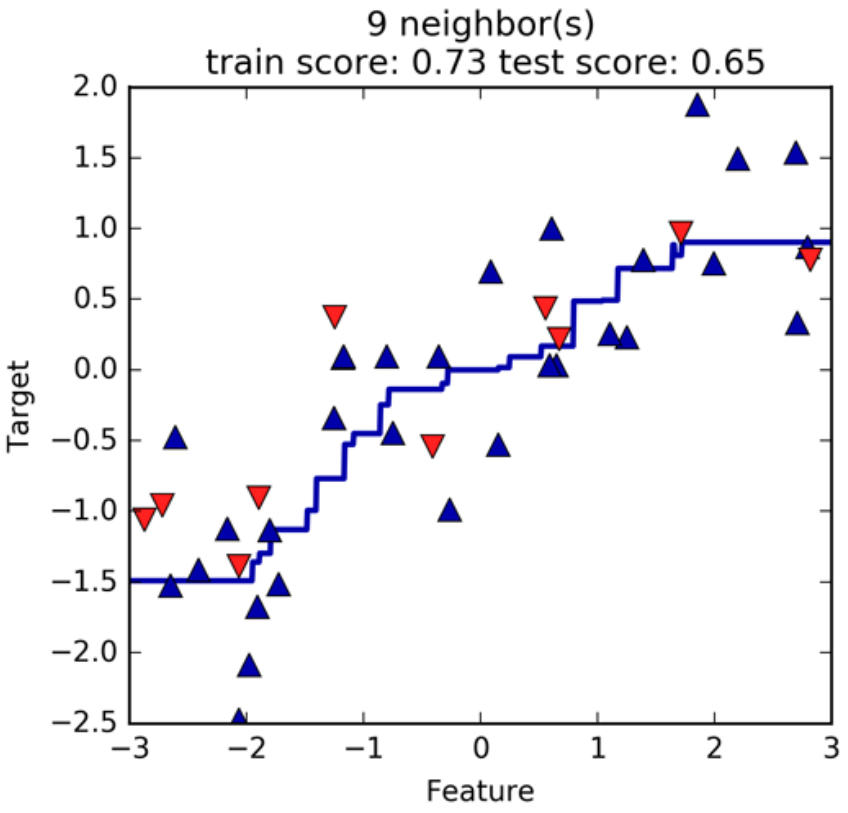


Elbow Method

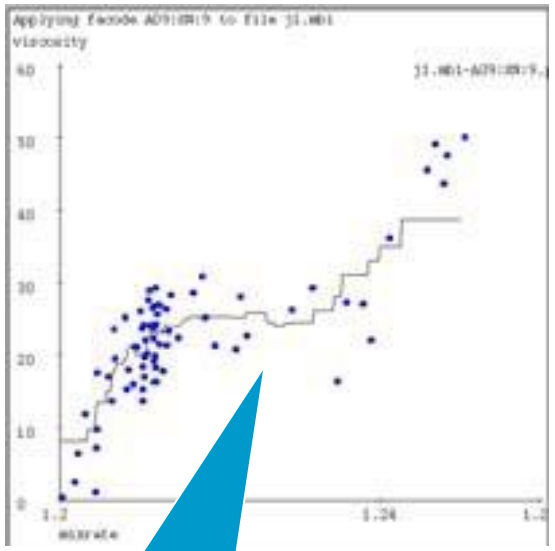
KNN Regression





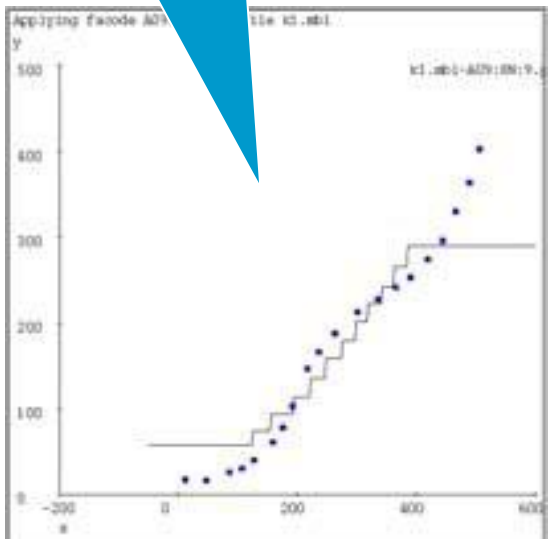


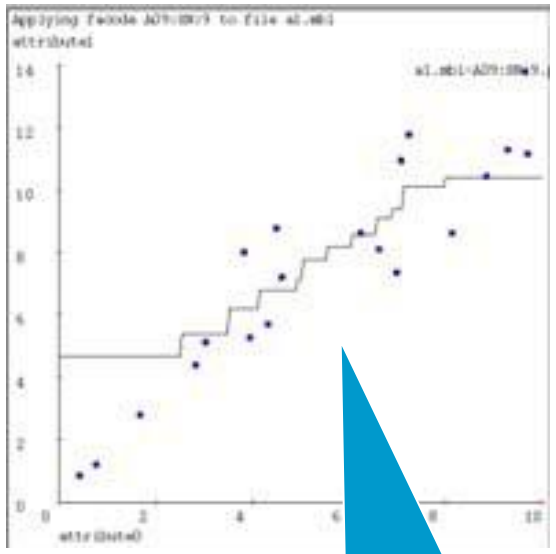
k-Nearest Neighbor (k = 9)



A magnificent job of noise smoothing. Three cheers for 9-nearest-neighbor.
...But the jerkiness isn't good.

Appalling behavior! Loses all the detail that 1-nearest neighbor would give. The tails are horrible!





Fits much less of the noise, captures trends. But still, frankly, pathetic compared with linear regression.

Adapted from "Instance-Based Learning" lecture slides by Andrew Moore, CMU.

Weights in kNN

class sklearn.neighbors.KNeighborsClassifier(n_neighbors=5, *, weights='uniform', algorithm='auto', leaf_size=30, p=2, metric='minkowski', metric_params=None, n_jobs=None, **kwargs)[source]

weights{‘uniform’, ‘distance’} or callable, default=’uniform’
weight function used in prediction. Possible values:

‘uniform’ : uniform weights. All points in each neighborhood are weighted equally.

‘distance’ : weight points by the inverse of their distance. in this case, closer neighbors of a query point will have a greater influence than neighbors which are further away.

Learning Algorithm

class sklearn.neighbors.KNeighborsClassifier(n_neighbors=5, *, weights='uniform', algorithm='auto', leaf_size=30, p=2, metric='minkowski', metric_params=None, n_jobs=None, **kwargs)[source]

algorithm{‘auto’, ‘ball_tree’, ‘kd_tree’, ‘brute’}, default=’auto’
Algorithm used to compute the nearest neighbors:

‘ball_tree’ will use BallTree
‘kd_tree’ will use KDTree
‘brute’ will use a brute-force search.
‘auto’ will attempt to decide the most appropriate algorithm based on the values passed to fit method.

Neighbors in Sklearn

neighbors.BallTree(X[, leaf_size, metric])	BallTree for fast generalized N-point problems
neighbors.DistanceMetric	DistanceMetric class
neighbors.KDTree(X[, leaf_size, metric])	KDTree for fast generalized N-point problems
neighbors.KernelDensity(*[, bandwidth, ...])	Kernel Density Estimation.
neighbors.KNeighborsClassifier([...])	Classifier implementing the k-nearest neighbors vote.
neighbors.KNeighborsRegressor([n_neighbors, ...])	Regression based on k-nearest neighbors.
neighbors.KNeighborsTransformer(*[, mode, ...])	Transform X into a (weighted) graph of k nearest neighbors
neighbors.LocalOutlierFactor([n_neighbors, ...])	Unsupervised Outlier Detection using Local Outlier Factor (LOF)
neighbors.RadiusNeighborsClassifier([...])	Classifier implementing a vote among neighbors within a given radius
neighbors.RadiusNeighborsRegressor([radius, ...])	Regression based on neighbors within a fixed radius.
neighbors.RadiusNeighborsTransformer(*[, ...])	Transform X into a (weighted) graph of neighbors nearer than a radius
neighbors.NearestCentroid([metric, ...])	Nearest centroid classifier.
neighbors.NearestNeighbors(*[, n_neighbors, ...])	Unsupervised learner for implementing neighbor searches.
neighbors.NeighborhoodComponentsAnalysis([...])	Neighborhood Components Analysis

neighbors.kneighbors_graph(X, n_neighbors, *)	Computes the (weighted) graph of k-Neighbors for points in X
neighbors.radius_neighbors_graph(X, radius, *)	Computes the (weighted) graph of Neighbors for points in X

நன்றி வணக்கம்