ECE 5984: Introduction to Machine Learning

Topics:

- Supervised Learning
  - Measuring performance
- Nearest Neighbour

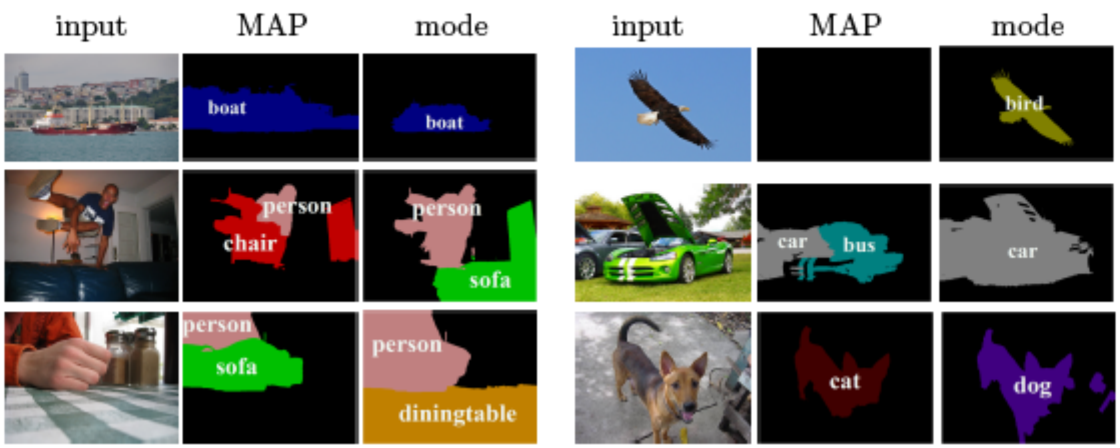
Readings: Barber 14 (kNN)

Dhruv Batra

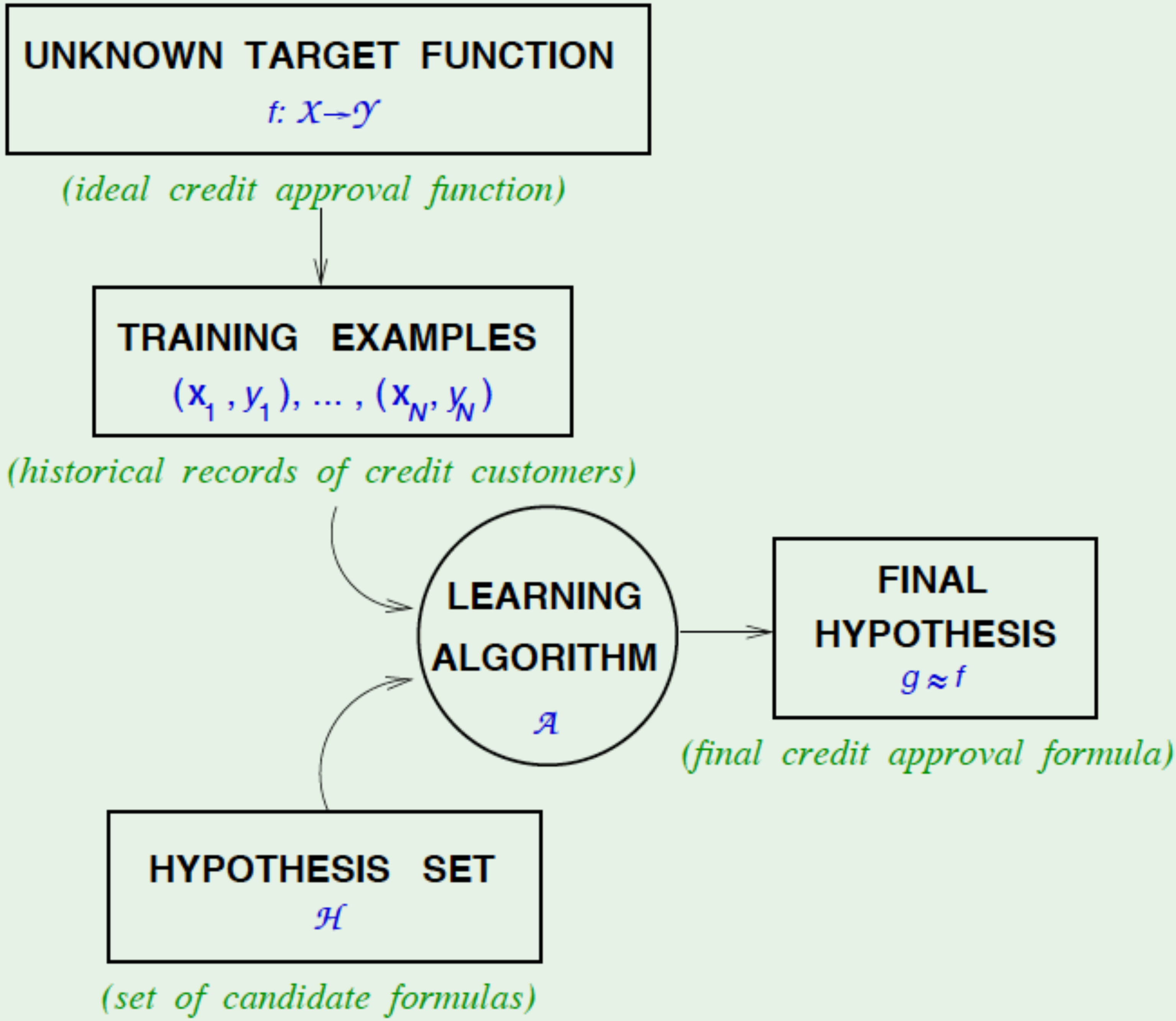
Virginia Tech

TA: Qing Sun

- PhD candidate at ECE department
- Research work/interest:
  - Diverse outputs based on structured probabilistic models
  - Structured-output prediction



Recap from last time


UNKNOWN TARGET FUNCTION
$f: \mathcal{X} \to \mathcal{Y}$
(ideal credit approval function)
TRAINING EXAMPLES
$(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)$
(historical records of credit customers)
LEARNING ALGORITHM
$\mathcal{A}$
FINAL HYPOTHESIS
$g \approx f$
(final credit approval formula)
HYPOTHESIS SET
$\mathcal{H}$
(set of candidate formulas)

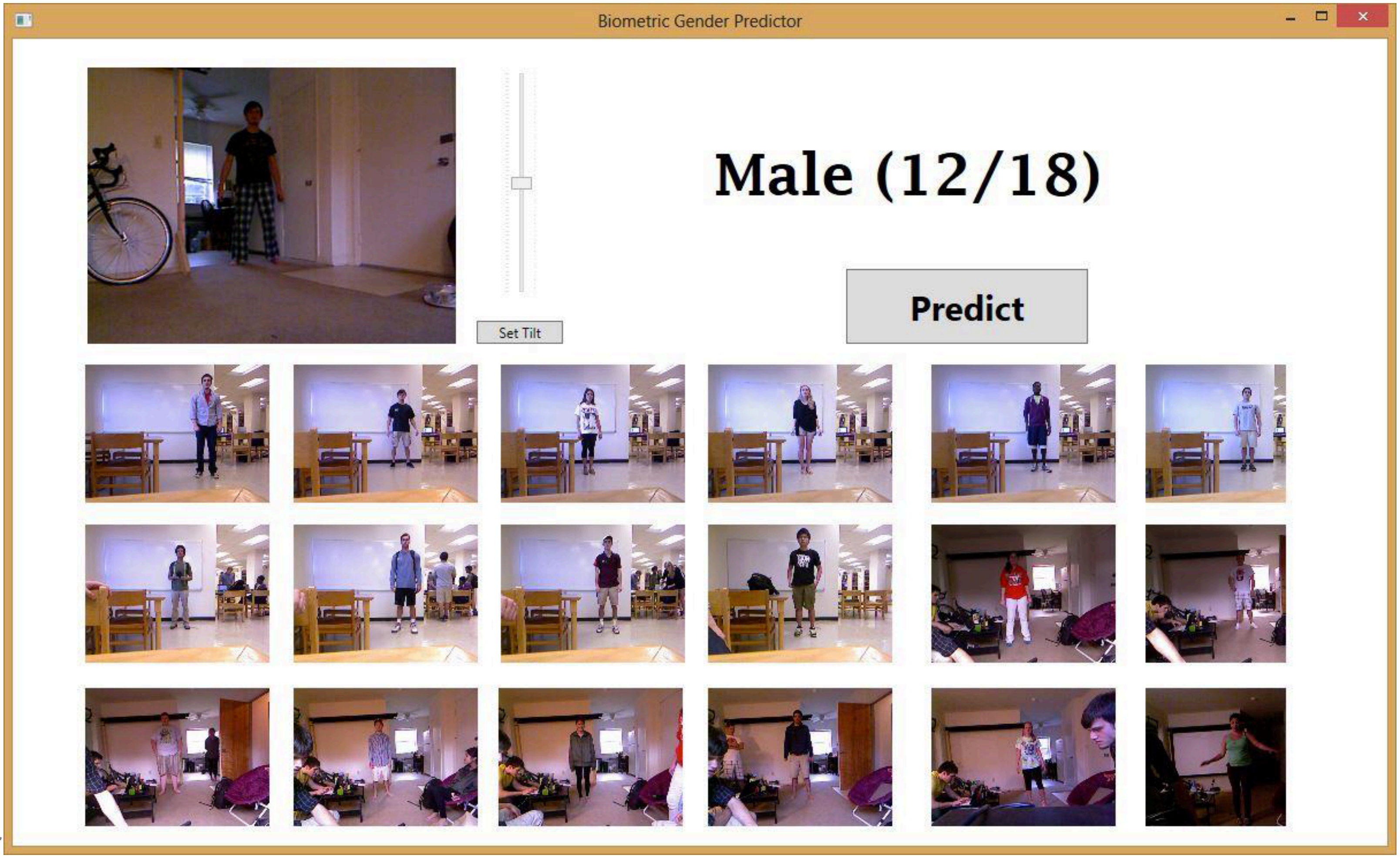
Nearest Neighbour

- Demo 1
  - http://cgm.cs.mcgill.ca/~soss/cs644/projects/perrier/Nearest.html

- Demo 2
  - http://www.cs.technion.ac.il/~rani/LocBoost/

Spring 2013 Projects

- Gender Classification from body proportions
  - Igor Janjic & Daniel Friedman, Juniors



Plan for today

- Supervised/Inductive Learning
  - (A bit more on) Loss functions

- Nearest Neighbour
  - Common Distance Metrics
  - Kernel Classification/Regression
  - Curse of Dimensionality

Loss/Error Functions

- How do we measure performance?

- Regression:
  - L_2 error

- Classification:
  - #misclassifications
  - Weighted misclassification via a cost matrix

  - For 2-class classification:
    - True Positive, False Positive, True Negative, False Negative

  - For k-class classification:
    - Confusion Matrix

- ROC curves
  - http://psych.hanover.edu/JavaTest/SDT/ROC.html

Nearest Neighbours

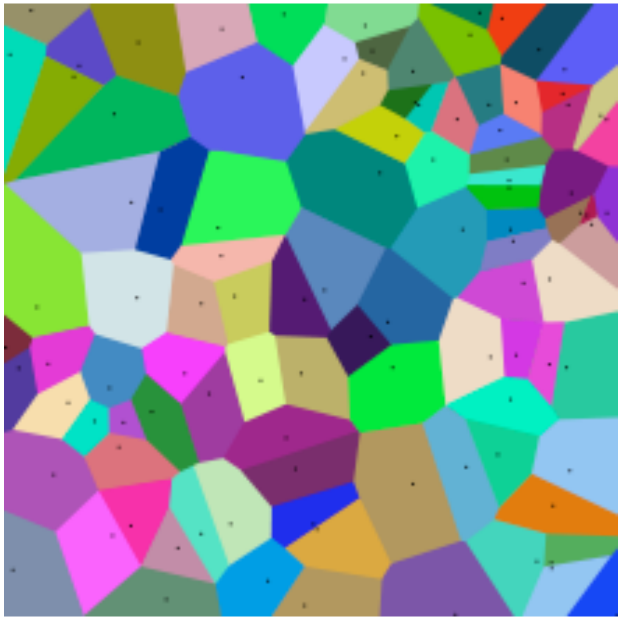


Image Credit: Wikipedia

Instance/Memory-based Learning

Four things make a memory based learner:

- *A distance metric*
- *How many nearby neighbors to look at?*
- *A weighting function (optional)*
- *How to fit with the local points?*

1-Nearest Neighbour

Four things make a memory based learner:

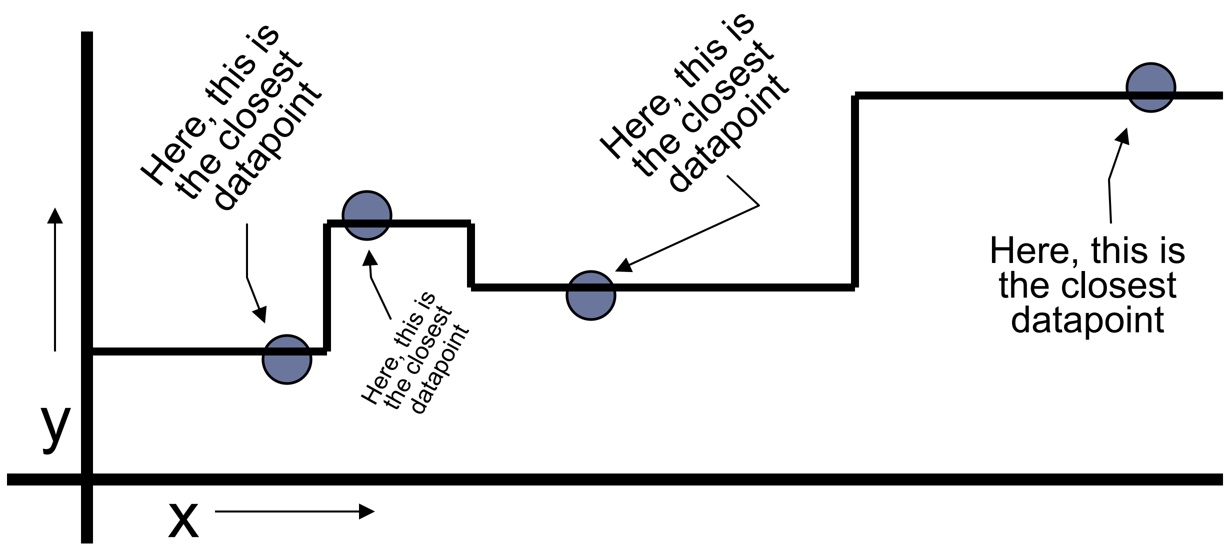
- *A distance metric*
  - **Euclidean (and others)**
- *How many nearby neighbors to look at?*
  - **1**
- *A weighting function (optional)*
  - **unused**
- *How to fit with the local points?*
  - **Just predict the same output as the nearest neighbour.**

k-Nearest Neighbour

Four things make a memory based learner:

- *A distance metric*
  - **Euclidean (and others)**
- *How many nearby neighbors to look at?*
  - **k**
- *A weighting function (optional)*
  - **unused**
- *How to fit with the local points?*
  - **Just predict the average output among the nearest neighbours.**

1-NN for Regression

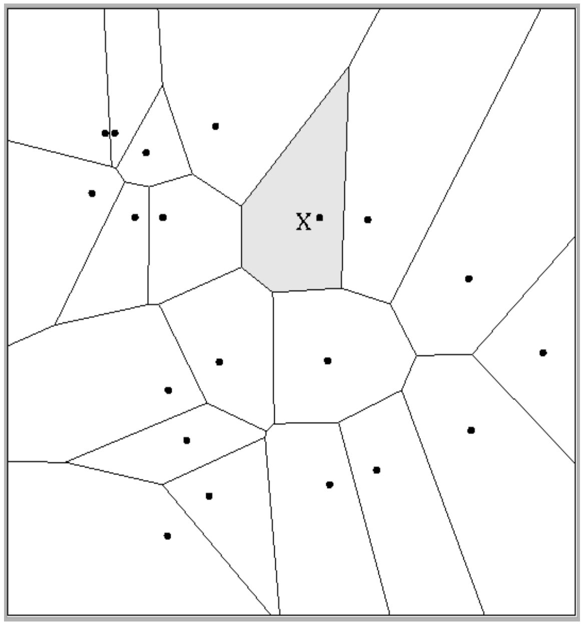


Multivariate distance metrics

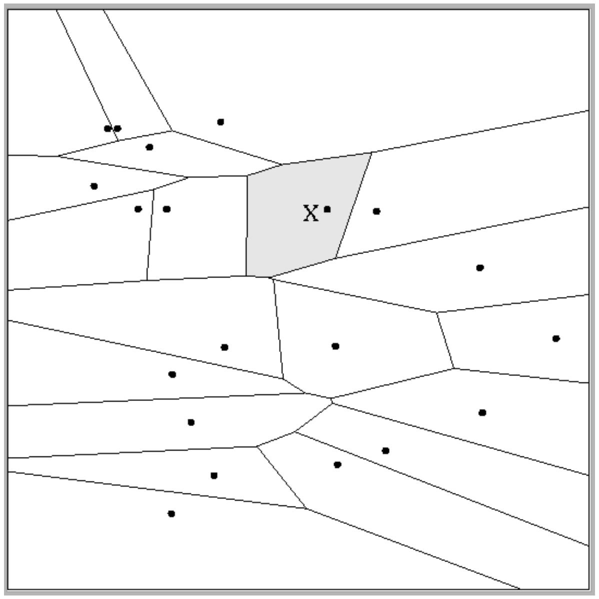
Suppose the input vectors $\mathbf{x}_1$, $\mathbf{x}_2$, …$\mathbf{x}_N$ are two dimensional:

$\mathbf{x}_1 = (x_{11} , x_{12})$, $\mathbf{x}_2 = (x_{21} , x_{22})$, …$\mathbf{x}_N = (x_{N1} , x_{N2})$.

One can draw the nearest-neighbor regions in input space.



$Dist(\mathbf{x}_i,\mathbf{x}_j) = (x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2$



$Dist(\mathbf{x}_i,\mathbf{x}_j) =(x_{i1} - x_{j1})^2+(3x_{i2} - 3x_{j2})^2$

The relative scalings in the distance metric affect region shapes

Slide Credit: Carlos Guestrin

Euclidean distance metric

Or equivalently,

$$D(x, x') = \sqrt{\sum_i \sigma_i^2 (x_i - x'_i)^2}$$

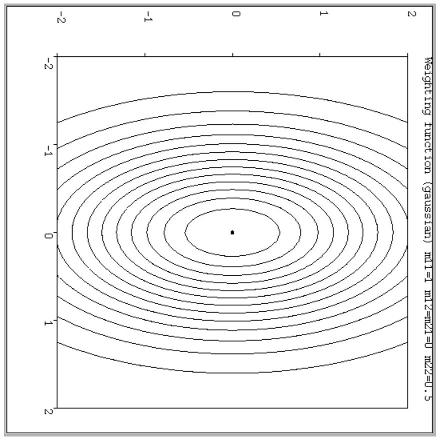
$$D(x, x') = \sqrt{(x_i - x'_i)^T A (x_i - x'_i)}$$

where

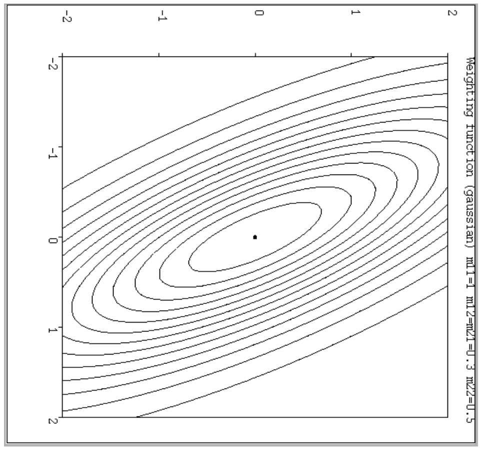
$$A = \begin{bmatrix} \sigma_1^2 & 0 & \cdots & 0 \\ 0 & \sigma_2^2 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & \sigma_N^2 \end{bmatrix}$$

Slide Credit: Carlos Guestrin

Notable distance metrics
(and their level sets)



Scaled Euclidian (L_2)



Mahalanobis
(non-diagonal A)

Minkowski distance

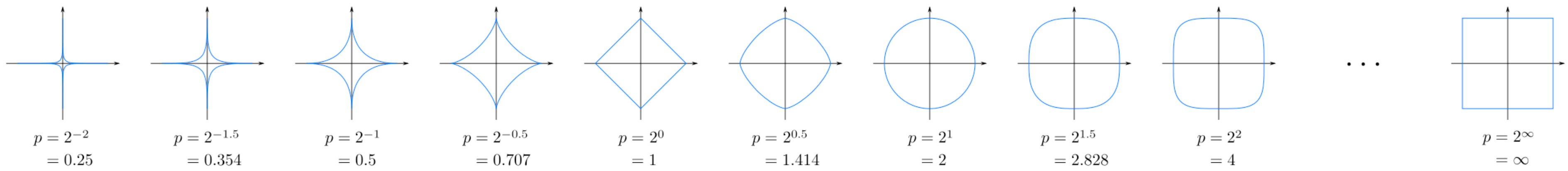
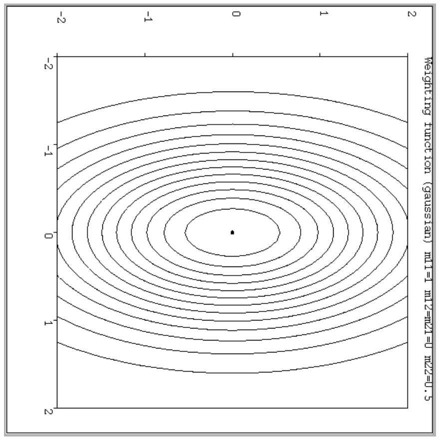
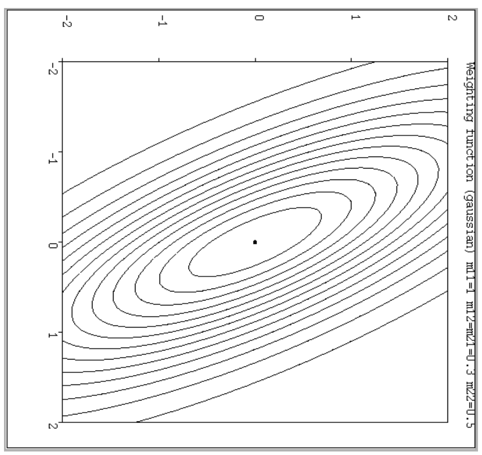


Image Credit: By Waldir (Based on File:MinkowskiCircles.svg)

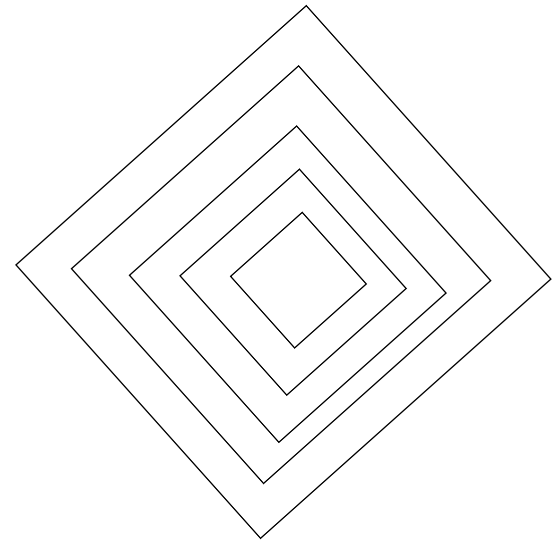
Notable distance metrics (and their level sets)



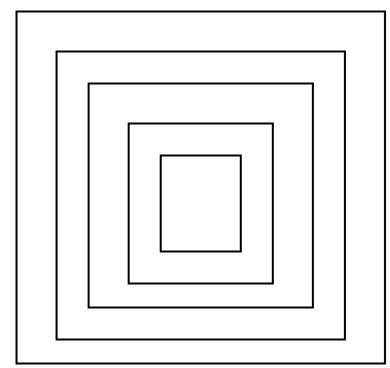
Scaled Euclidian (L_2)



Mahalanobis (non-diagonal A)



L_1 norm (absolute)



L_{inf} *(max) norm*

Slide Credit: Carlos Guestrin

Parametric vs Non-Parametric Models

- Does the capacity (size of hypothesis class) grow with size of training data?
  - Yes = Non-Parametric Models
  - No = Parametric Models

- Example
  - http://www.theparticle.com/applets/ml/nearest_neighbor/

Weighted k-NNs

- Neighbors are not all the same

1 vs k Nearest Neighbour

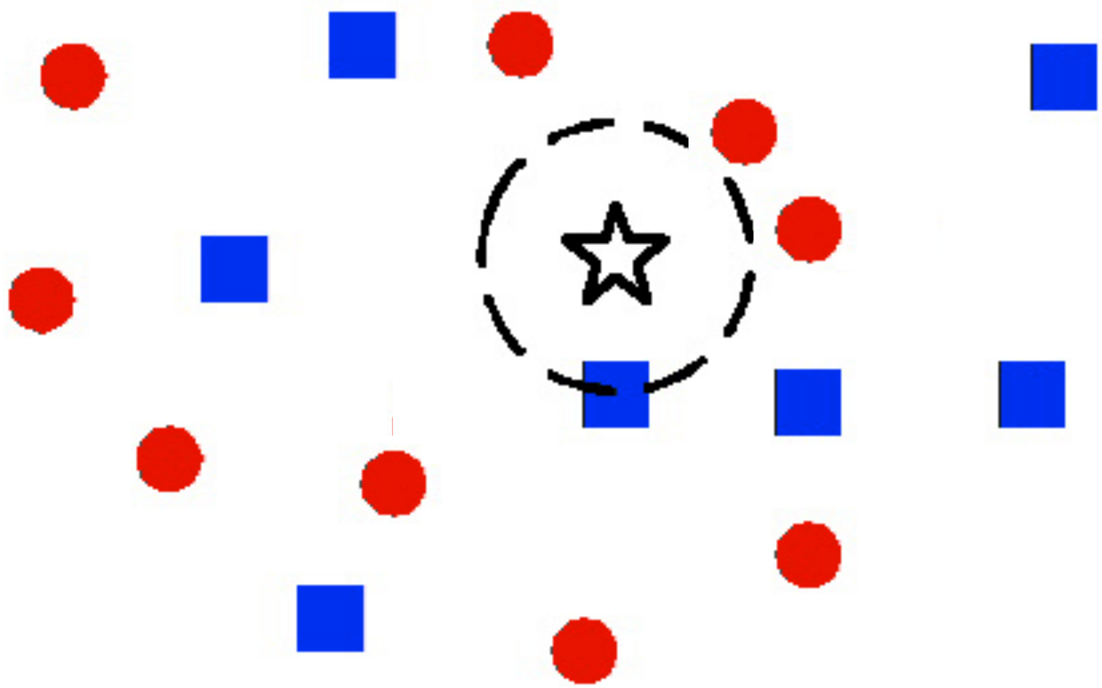
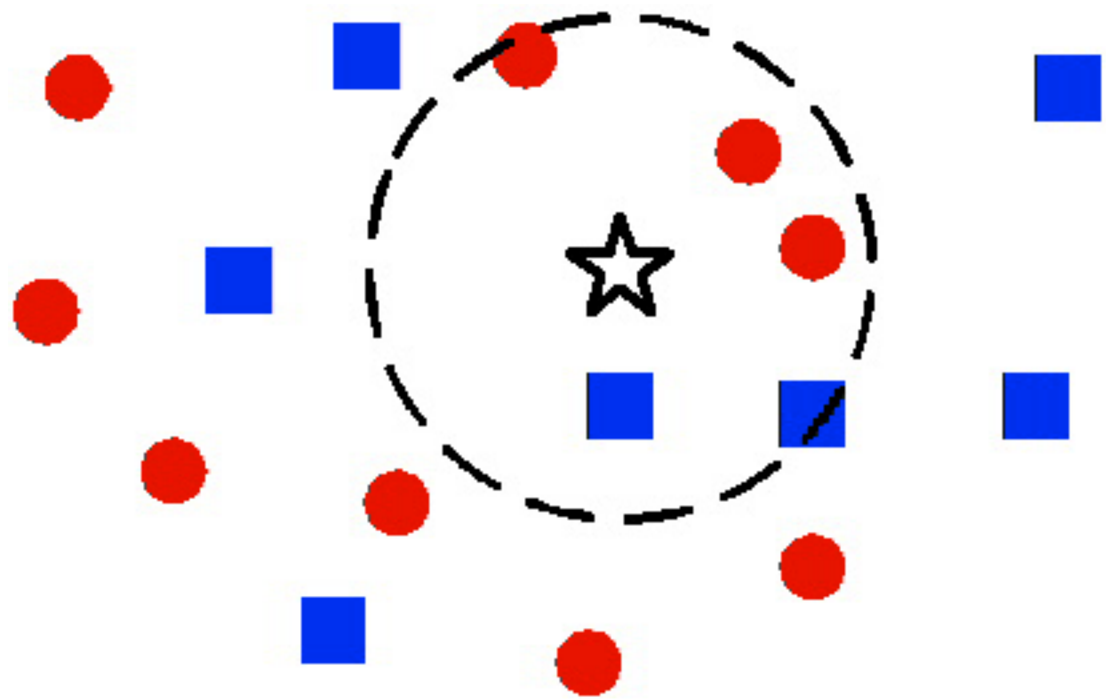
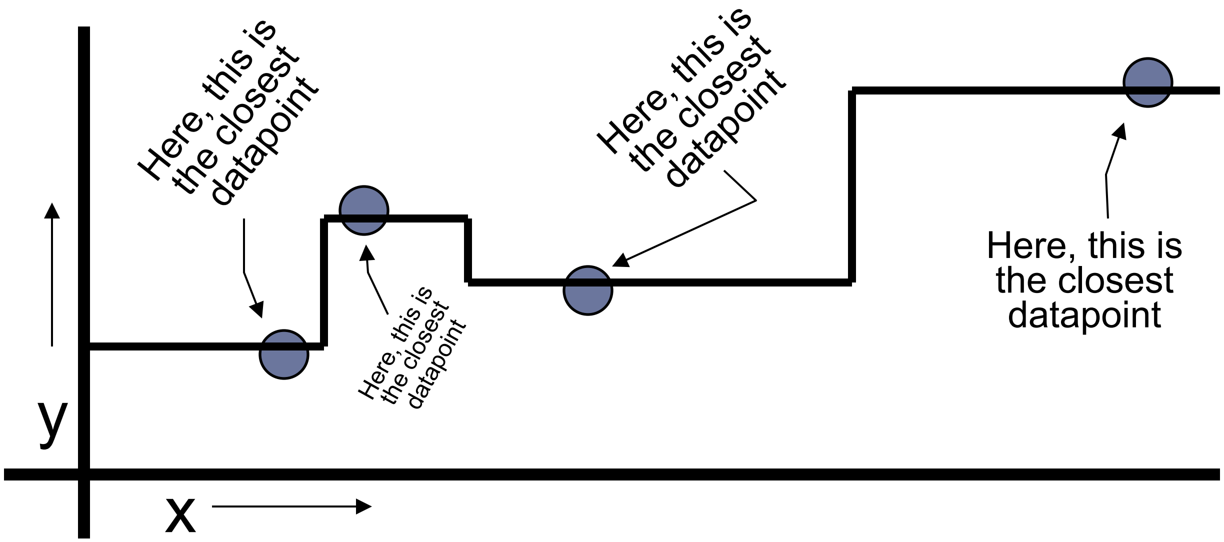


Image Credit: Ying Wu

1 vs k Nearest Neighbour

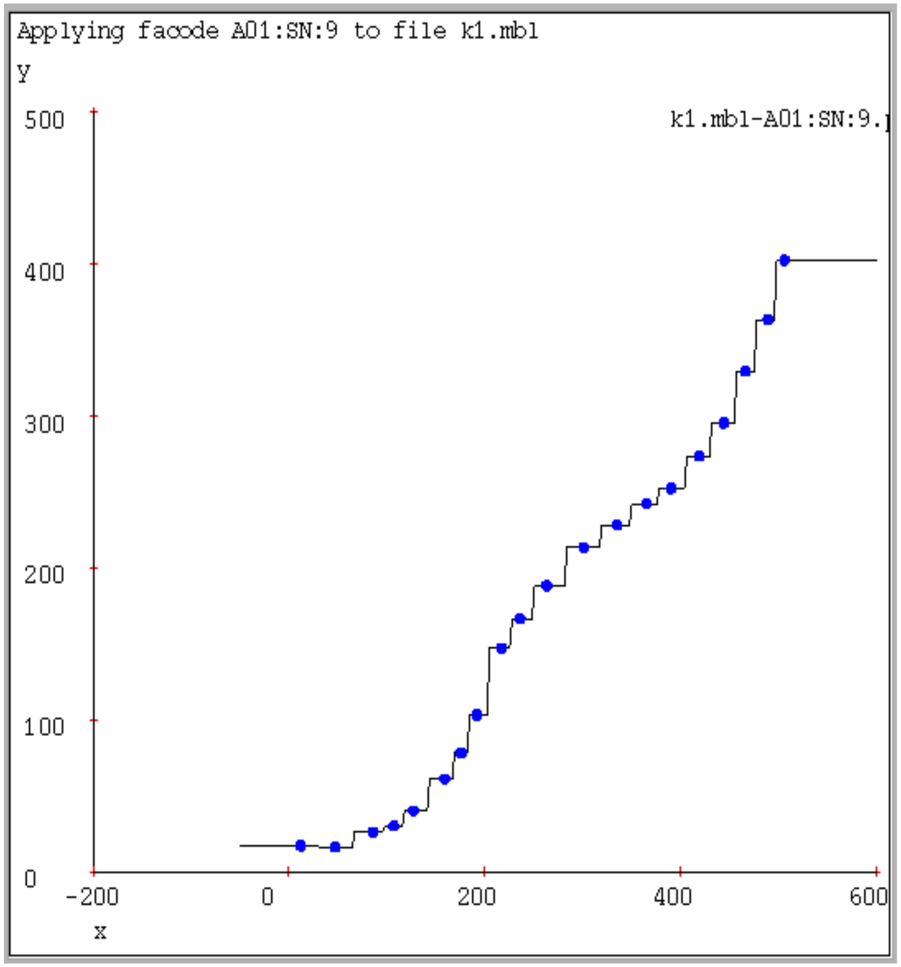


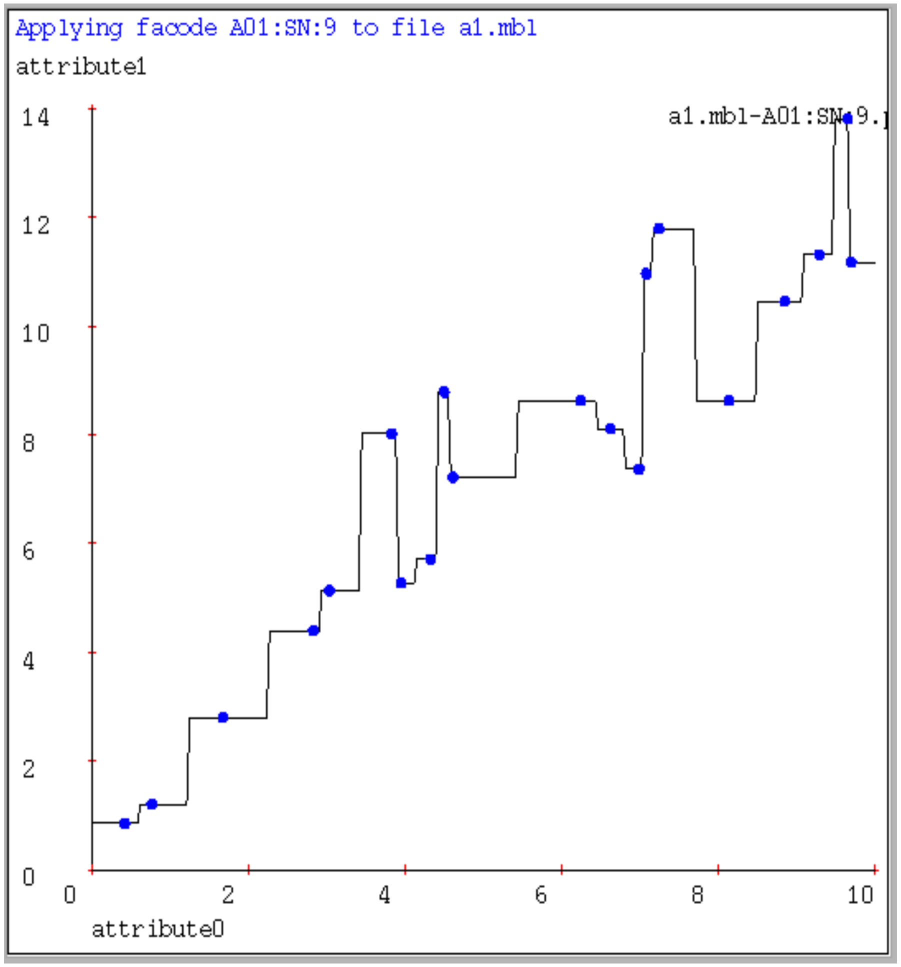
1-NN for Regression



1-NN for Regression

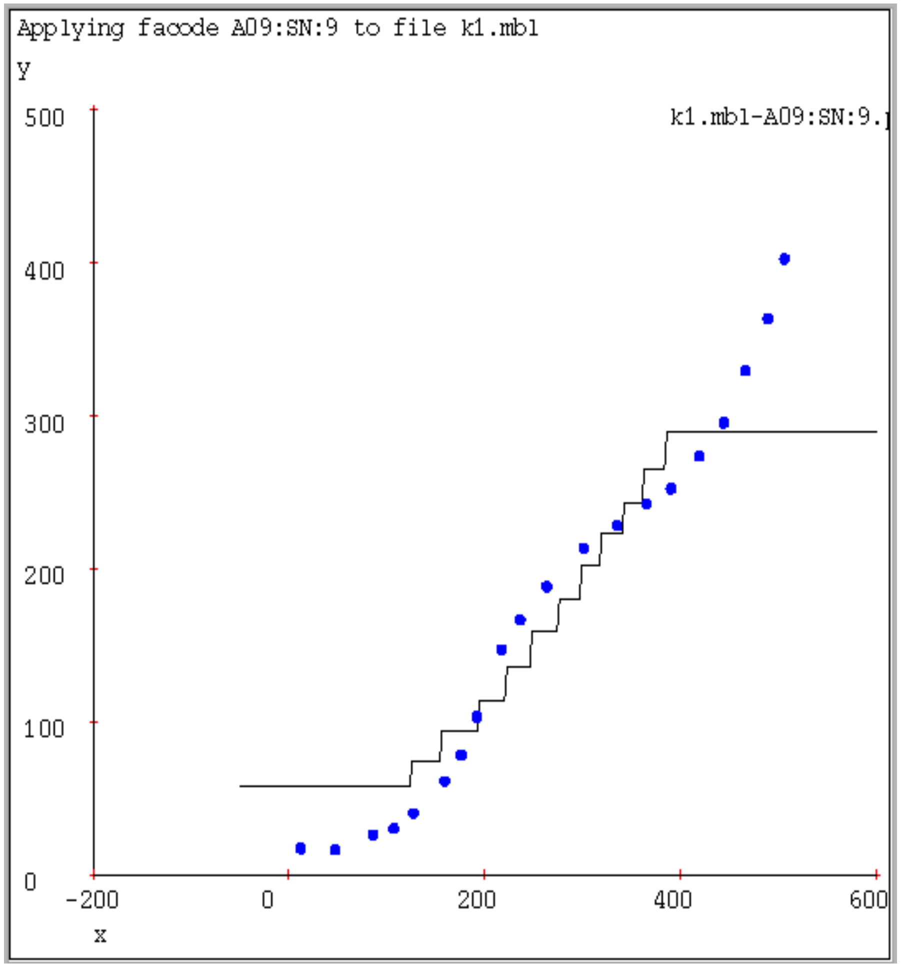
- Often bumpy (overfits)





9-NN for Regression

- Often bumpy (overfits)



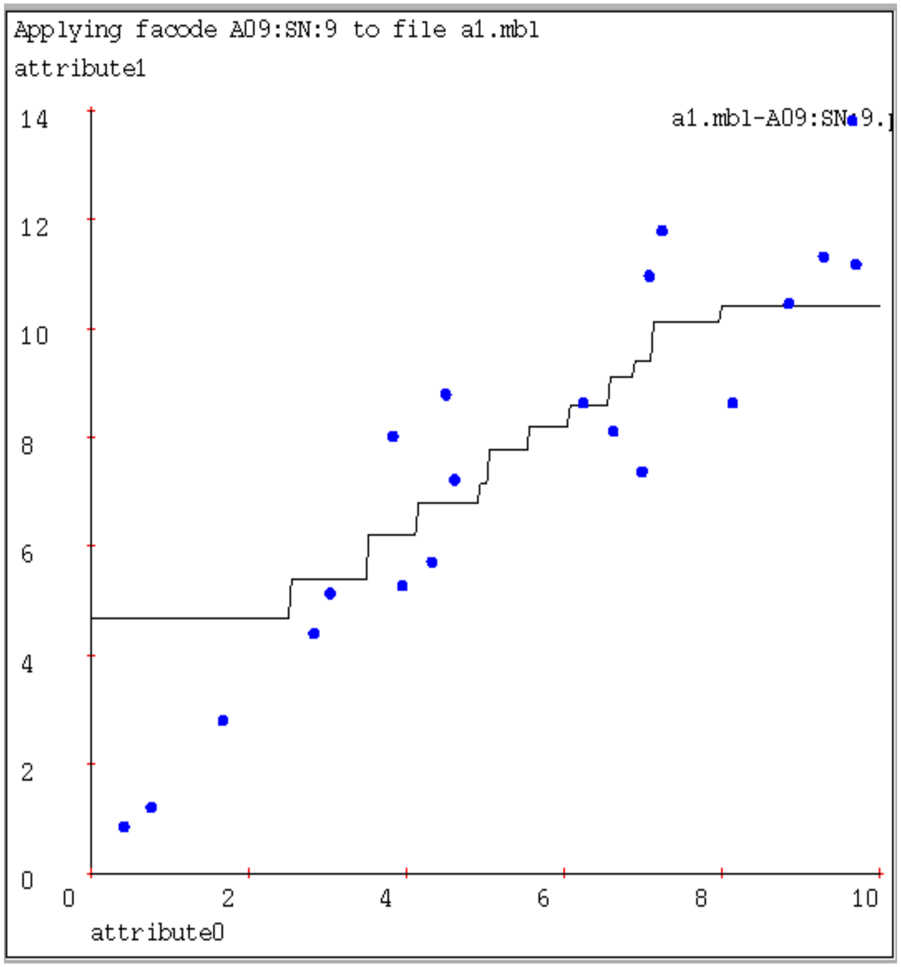


Figure Credit: Andrew Moore

Kernel Regression/Classification

Four things make a memory based learner:

- *A distance metric*
  - **Euclidean (and others)**
- *How many nearby neighbors to look at?*
  - **All of them**
- *A weighting function (optional)*
  - $w_i = exp(-d(x_i, query)^2 / \sigma^2)$
  - Nearby points to the query are weighted strongly, far points weakly. The σ parameter is the **Kernel Width**. Very important.
- *How to fit with the local points?*
  - **Predict the weighted average of the outputs**
    predict = $\Sigma w_i y_i / \Sigma w_i$