# ECE 5984: Introduction to Machine Learning

Topics:

- Supervised Learning
  - General Setup, learning from data
- Nearest Neighbour

Readings: Barber 14 (kNN)

Dhruv Batra

Virginia Tech

# Administrativia

- New class room
  - GBJ 102
- More space
  - Force-adds approved
- Scholar
  - Anybody not have access?
  - Still have problems reading/submitting? Resolve ASAP.
  - Please post questions on Scholar Forum.
  - Please check scholar forums. You might not know you have a doubt.

# Administrativia

- Reading/Material/Pointers
  - Slides on Scholar
  - Scanned handwritten notes on Scholar
  - Readings/Video pointers on Public Website

# Administrativia

- Computer Vision & Machine Learning Reading Group
  - Meet: Fridays 5-6pm
  - Reading CV/ML conference papers
  - Whittemore 654

# Plan for today

- Supervised/Inductive Learning
  - Setup
  - Goal: Classification, Regression
  - Procedural View
  - Statistical Estimation View
  - Loss functions

- Your first classifier: k-Nearest Neighbour

# Types of Learning

- Supervised learning
  - Training data includes desired outputs
- Unsupervised learning
  - Training data does not include desired outputs
- Weakly or Semi-supervised learning
  - Training data includes a few desired outputs
- Reinforcement learning
  - Rewards from sequence of actions

# Supervised / Inductive Learning

- Given
  - examples of a function (x, f(x))

- Predict function f(x) for new examples x
  - Discrete f(x): Classification
  - Continuous f(x): Regression
  - f(x) = Probability(x): Probability estimation

## Appropriate Applications for Supervised Learning

- **Situations where there is no human expert**

  $\mathbf{x}$: Bond graph for a new molecule.

  $f(\mathbf{x})$: Predicted binding strength to AIDS protease molecule.

- **Situations where humans can perform the task but can't describe how they do it.**

  $\mathbf{x}$: Bitmap picture of hand-written character

  $f(\mathbf{x})$: Ascii code of the character

- **Situations where the desired function is changing frequently**

  $\mathbf{x}$: Description of stock prices and trades for last 10 days.

  $f(\mathbf{x})$: Recommended stock transactions

- **Situations where each user needs a customized function $f$**

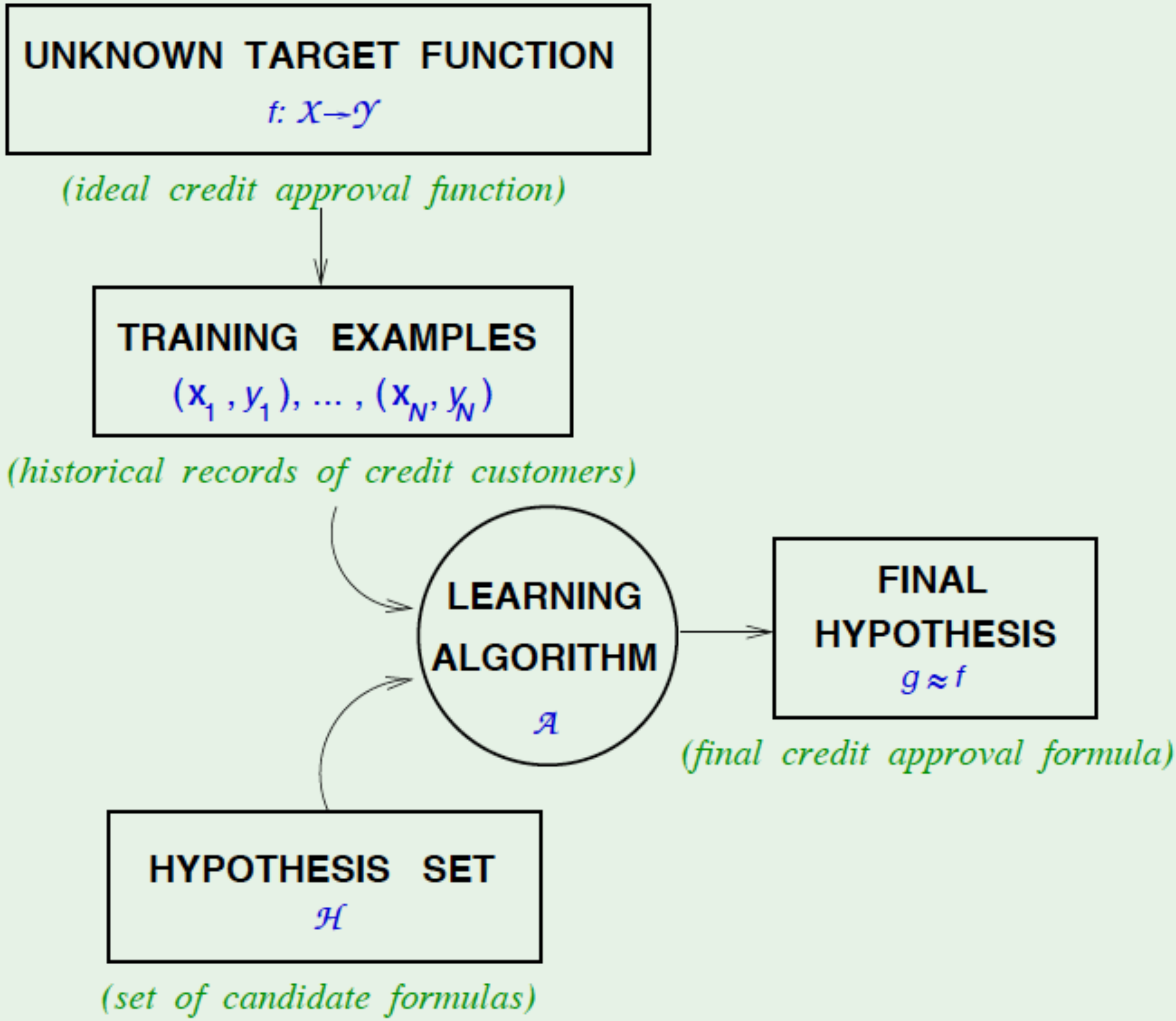
  $\mathbf{x}$: Incoming email message.

  $f(\mathbf{x})$: Importance score for presenting to user (or deleting without presenting).

Slide Credit: Pedro Domingos, Tom Mitchel, Tom Dietterich

# Supervised Learning

- Input: x (images, text, emails…)
- Output: y (spam or non-spam…)
- (Unknown) Target Function
  - f: X → Y (the "true" mapping / reality)
- Data
  - $(x_1,y_1), (x_2,y_2), \ldots, (x_N,y_N)$
- Model / Hypothesis Class
  - g: X → Y
  - $y = g(x) = \text{sign}(w^T x)$
- Learning = Search in hypothesis space
  - Find best g in model class.


UNKNOWN TARGET FUNCTION
$f: \mathcal{X} \to \mathcal{Y}$
(ideal credit approval function)
TRAINING EXAMPLES
$(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)$
(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)

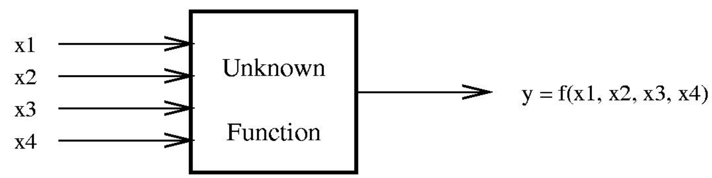
# Basic Steps of Supervised Learning

- **Set up** a supervised learning problem
- **Data collection**
  - Start with training data for which we know the correct outcome provided by a teacher or oracle.
- **Representation**
  - Choose how to represent the data.
- **Modeling**
  - Choose a hypothesis class: H = {g: X → Y}
- **Learning/Estimation**
  - Find best hypothesis you can in the chosen class.
- **Model Selection**
  - Try different models. Picks the best one. (More on this later)
- If happy stop
  - Else refine one or more of the above

# Learning is hard!

- No assumptions = No learning

**A Learning Problem**




| Example | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $y$ |
|---|---|---|---|---|---|
| 1 | 0 | 0 | 1 | 0 | 0 |
| 2 | 0 | 1 | 0 | 0 | 0 |
| 3 | 0 | 0 | 1 | 1 | 1 |
| 4 | 1 | 0 | 0 | 1 | 1 |
| 5 | 0 | 1 | 1 | 0 | 0 |
| 6 | 1 | 1 | 0 | 0 | 0 |
| 7 | 0 | 1 | 0 | 1 | 0 |

# Klingon vs Mlingon Classification

- Training Data
  - Klingon: klix, kour, koop
  - Mlingon: moo, maa, mou

- Testing Data: kap

- Which language?

- Why?

# 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

# Training vs Testing

- What do we want?
  - Good performance (low loss) on training data?
  - No, Good performance on *unseen test data!*

- Training Data:
  - $\{ (x_1,y_1), (x_2,y_2), \ldots, (x_N,y_N) \}$
  - Given to us for learning f

- Testing Data
  - $\{ x_1, x_2, \ldots, x_M \}$
  - Used to see if we have learnt anything

# Procedural View

- Training Stage:
  - Raw Data → x (Feature Extraction)
  - Training Data { (x,y) } → f (Learning)

- Testing Stage
  - Raw Data → x (Feature Extraction)
  - Test Data x → f(x) (Apply function, Evaluate error)

# Statistical Estimation View

- Probabilities to rescue:
  - x and y are *random variables*
  - $D = (x_1,y_1), (x_2,y_2), \ldots, (x_N,y_N) \qquad \sim P(X,Y)$
- IID: Independent Identically Distributed
  - Both training & testing data sampled IID from P(X,Y)
  - Learn on training set
  - Have some hope of *generalizing* to test set

# Concepts

- Capacity
  - Measure how large hypothesis class H is.
  - Are all functions allowed?

- Overfitting
  - f works well on training data
  - Works poorly on test data

- Generalization
  - The ability to achieve low error on new test data

# Guarantees

- 20 years of research in Learning Theory oversimplified:

- If you have:
  - Enough training data D
  - and H is not too complex
  - then *probably* we can generalize to unseen test data

# New Topic: Nearest Neighbours

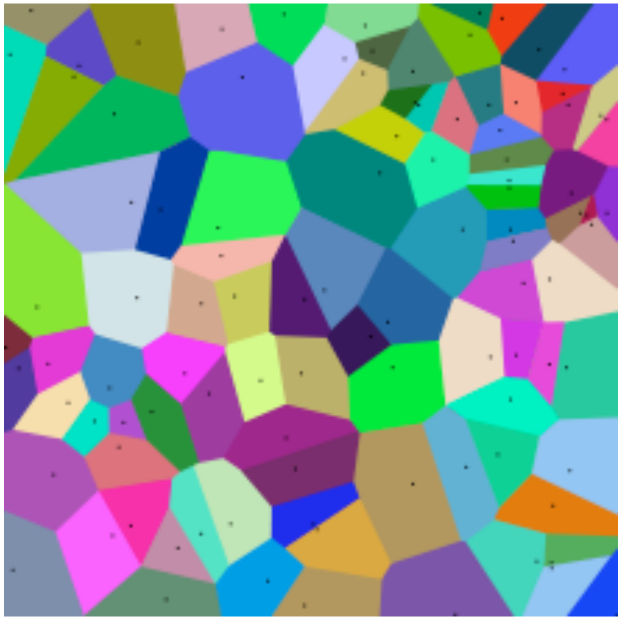




Image Credit: Wikipedia

# Synonyms

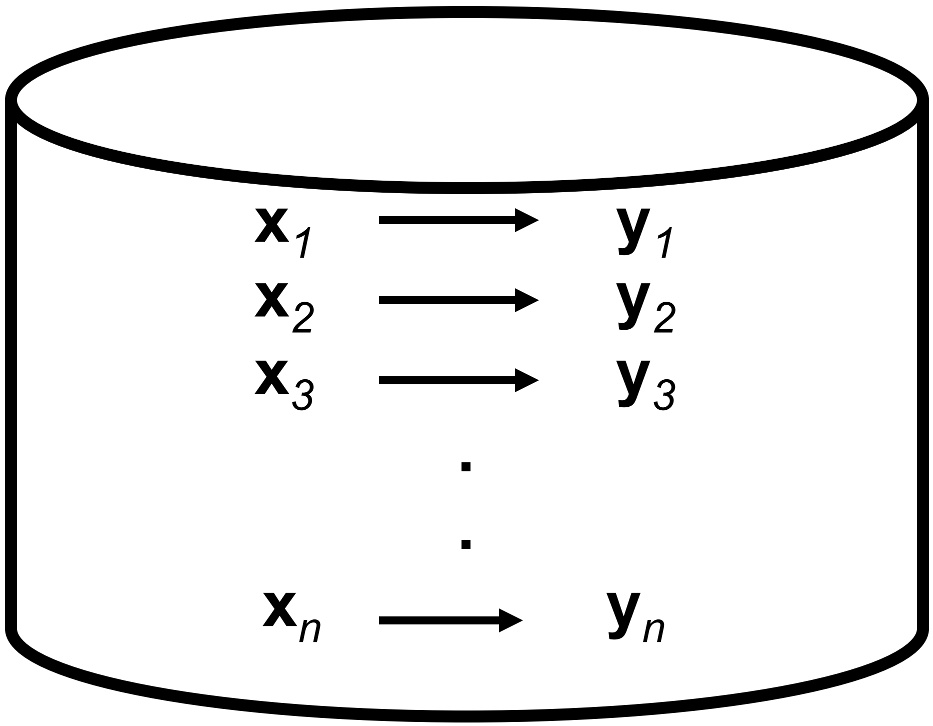
- Nearest Neighbours
- k-Nearest Neighbours
- Member of following families:
  - Instance-based Learning
  - Memory-based Learning
  - Exemplar methods
  - Non-parametric methods

# Nearest Neighbor is an example of….
# **Instance-based learning**

Has been around since about 1910.

To make a prediction, search database for similar datapoints, and fit with the local points.

Assumption: Nearby points behavior similarly wrt y



# 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 vs k Nearest Neighbour

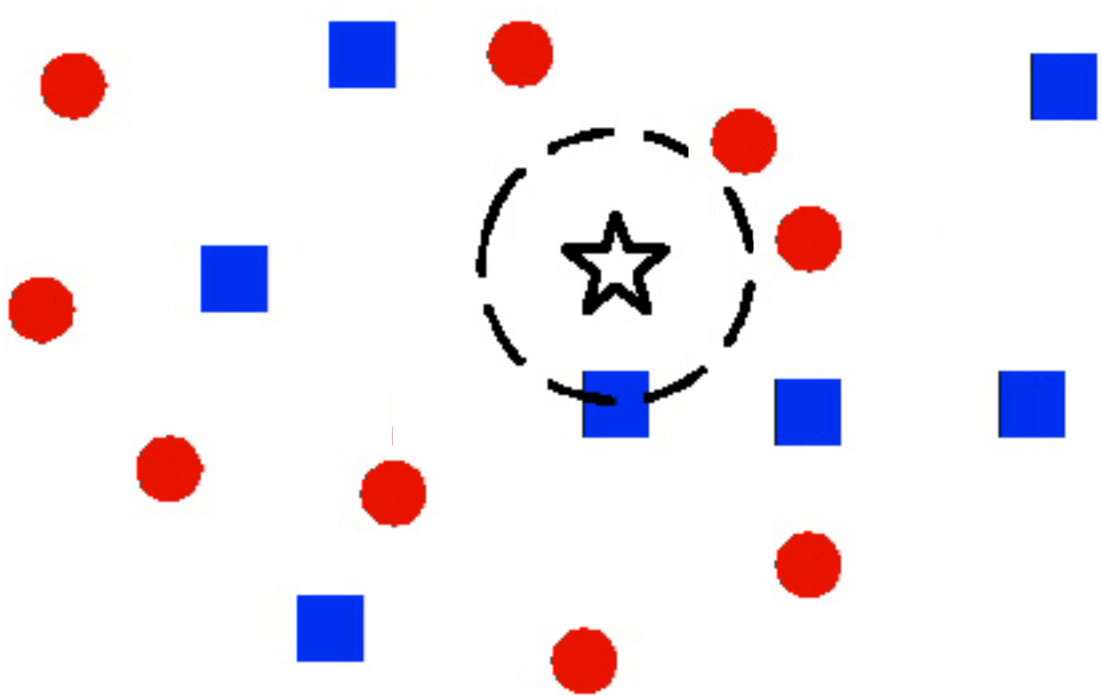




Image Credit: Ying Wu

# 1 vs k Nearest Neighbour

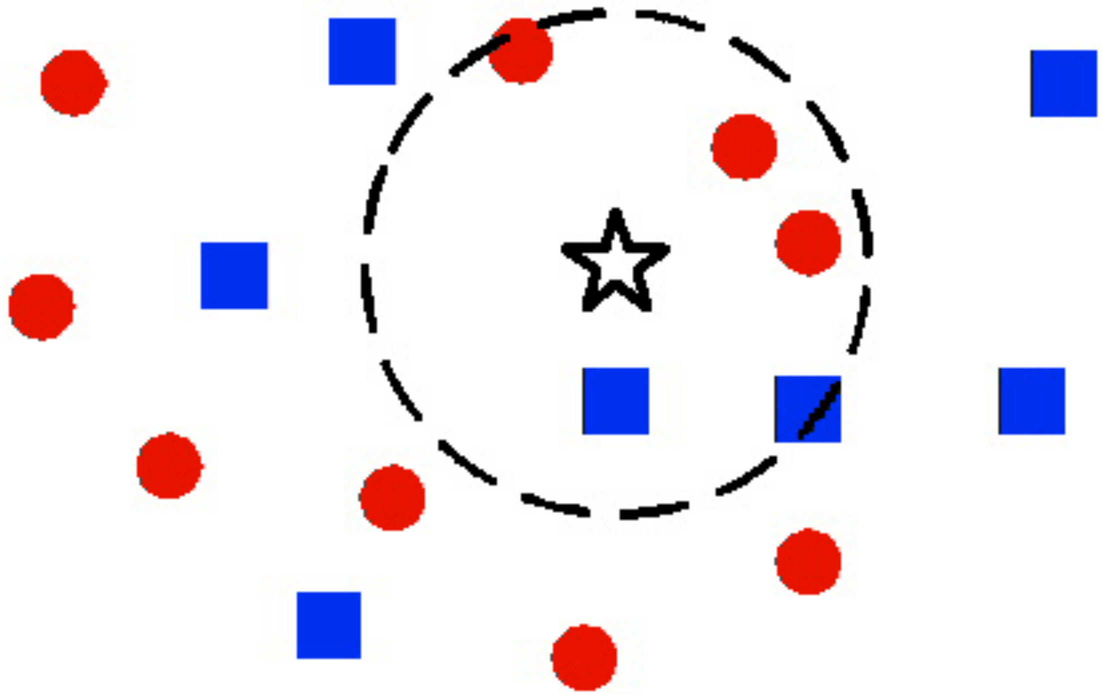




Image Credit: Ying Wu

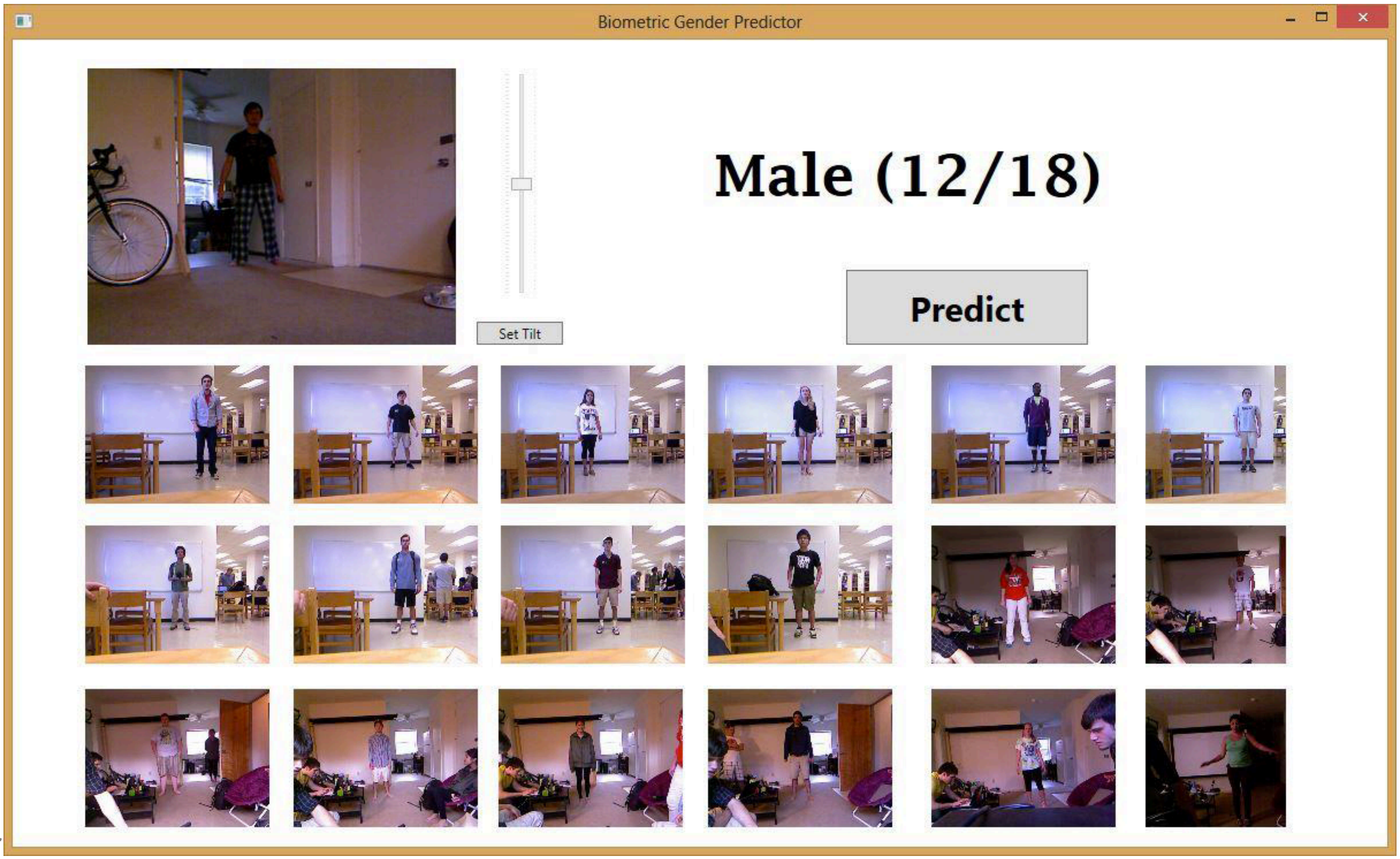
# Nearest Neighbour

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

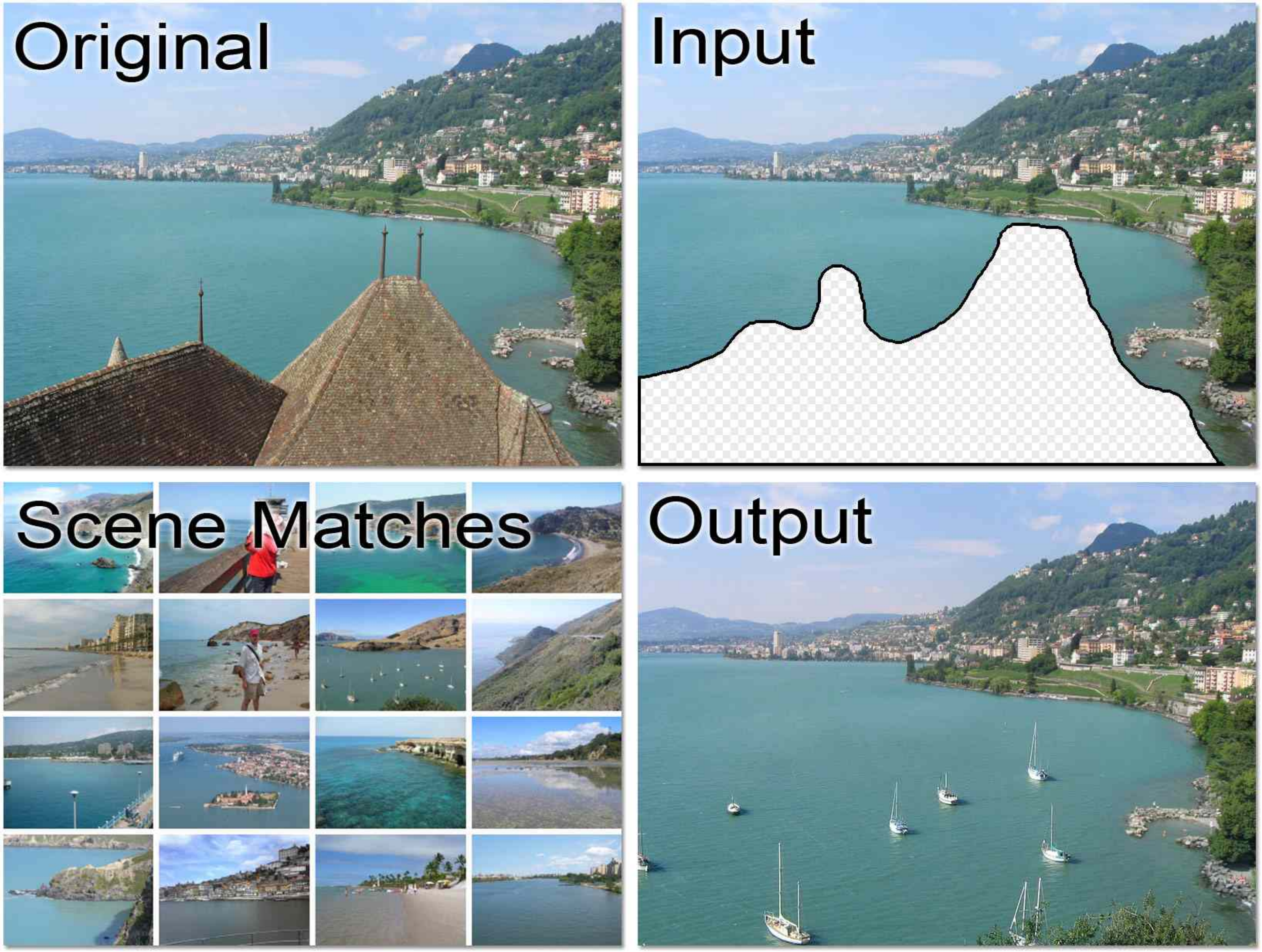
- Demo 2
  - http://www.cs.technion.ac.il/~rani/LocBoost/

# Spring 2013 Projects

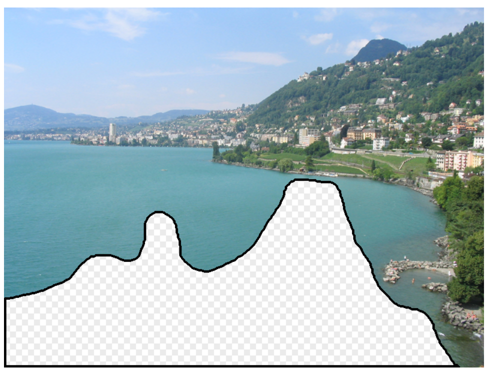
- Gender Classification from body proportions
  - Igor Janjic & Daniel Friedman, Juniors

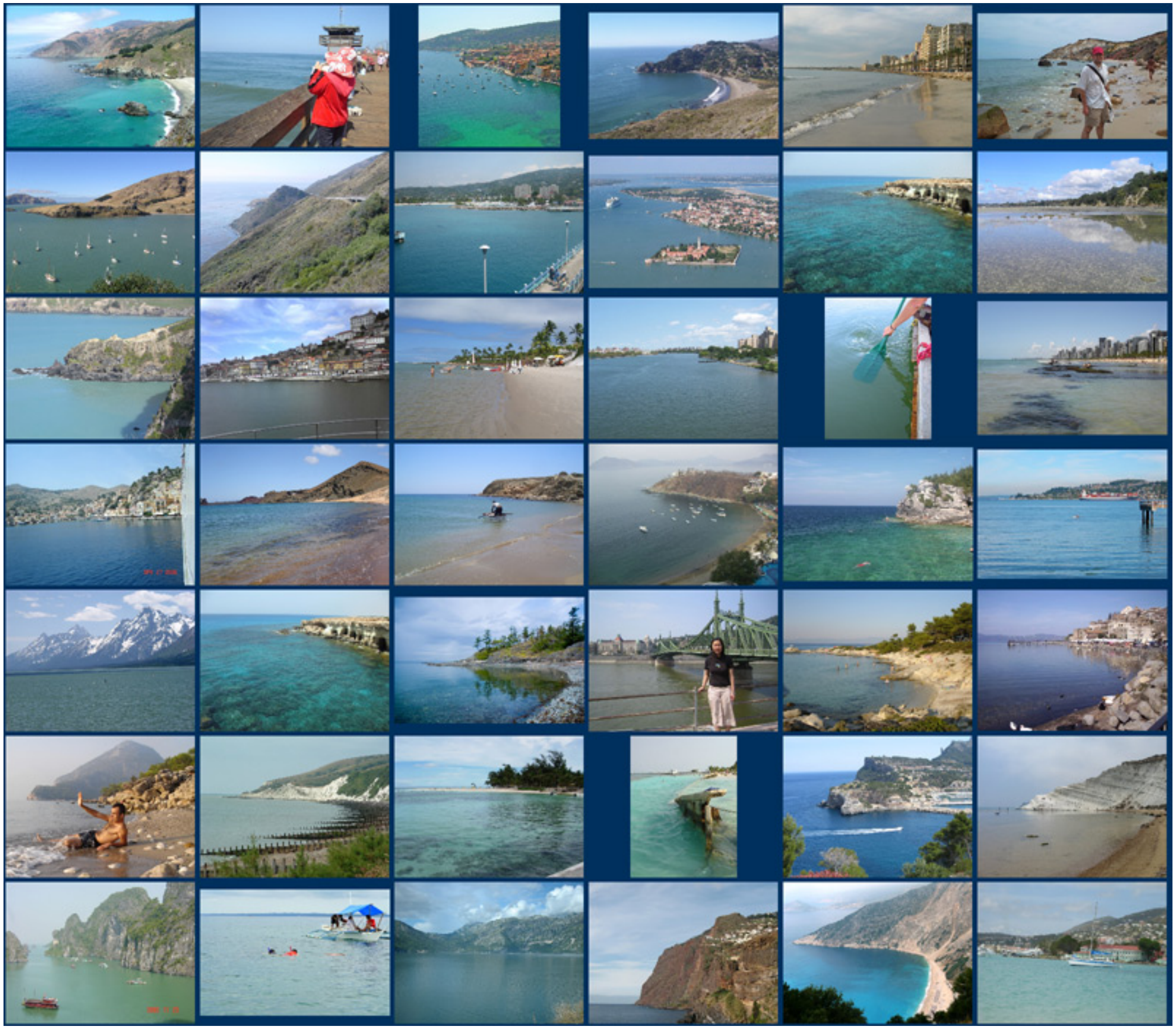


# Scene Completion [Hayes & Efros, SIGGRAPH07]




Hays and Efros, SIGGRAPH 2007

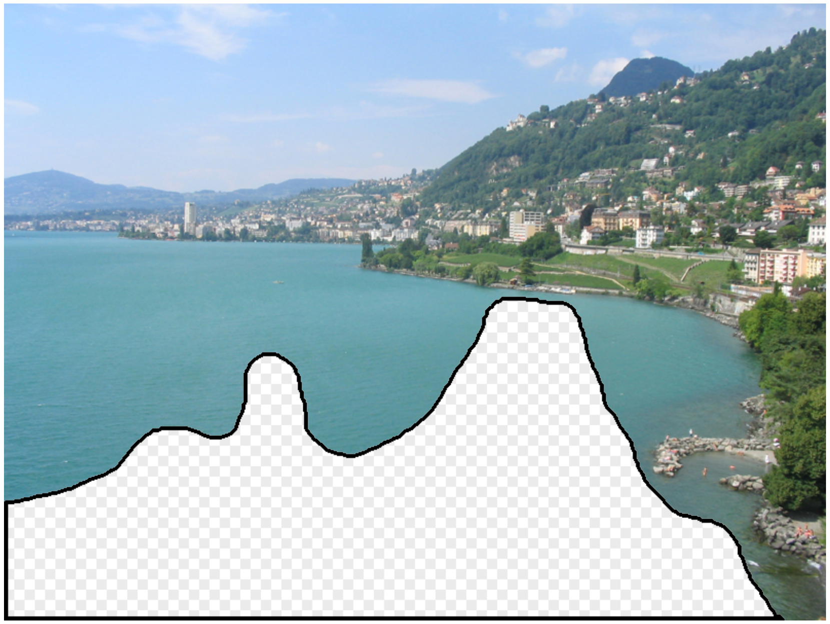




… 200 total

Hays and Efros, SIGGRAPH 2007

# Context Matching



Hays and Efros, SIGGRAPH 2007

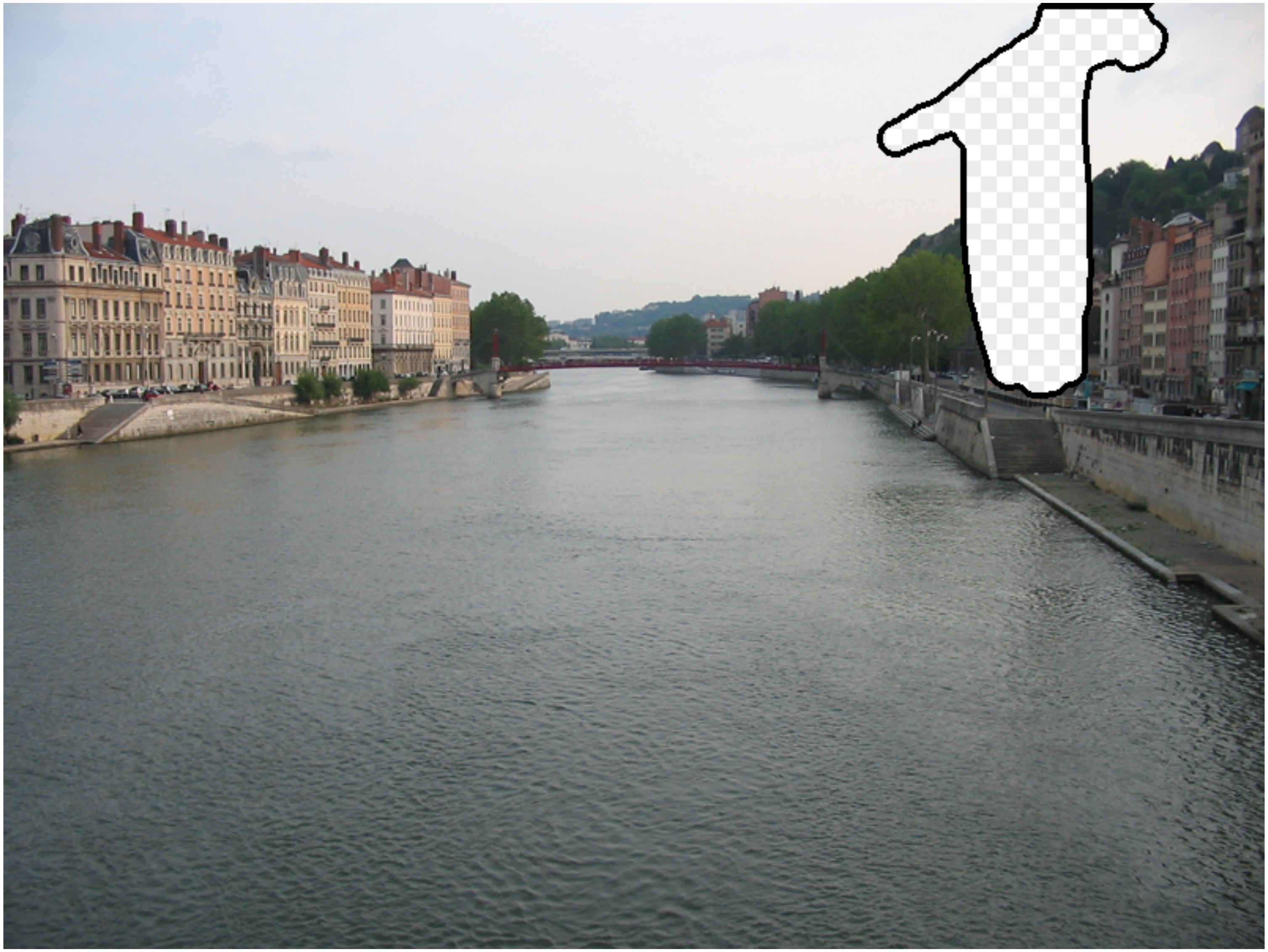

Graph cut + Poisson blending
Hays and Efros, SIGGRAPH 2007

Hays and Efros, SIGGRAPH 2007


Hays and Efros, SIGGRAPH 2007

Hays and Efros, SIGGRAPH 2007

Hays and Efros, SIGGRAPH 2007



Hays and Efros, SIGGRAPH 2007

Hays and Efros, SIGGRAPH 2007