BLG456E
Robotics
Intro to Reactive Robot Learning

Lecture Contents:

- Why learning.
- Supervised learning.
- Basic reactive controller learning.
- ROS Example

Lecturer:	Damien Jade Duff
Email:	djduff@itu.edu.tr
Office:	EEBF 2316
Schedule:	http://djduff.net/my-schedule
Coordination:	http://ninova.itu.edu.tr/Ders/4709

Some slides taken/adapted from Dr. Sanem Sarıel-Talay

What is learning?

- 1. **Learning**: getting better at doing things from experience.

- 2. **Learning**: When a program C improves its performance in T according to P after incorporating E.
  - Computer program C.
  - Class of tasks T.
  - Performance measure P.
  - Experience E.

(Tom Mİtchell)

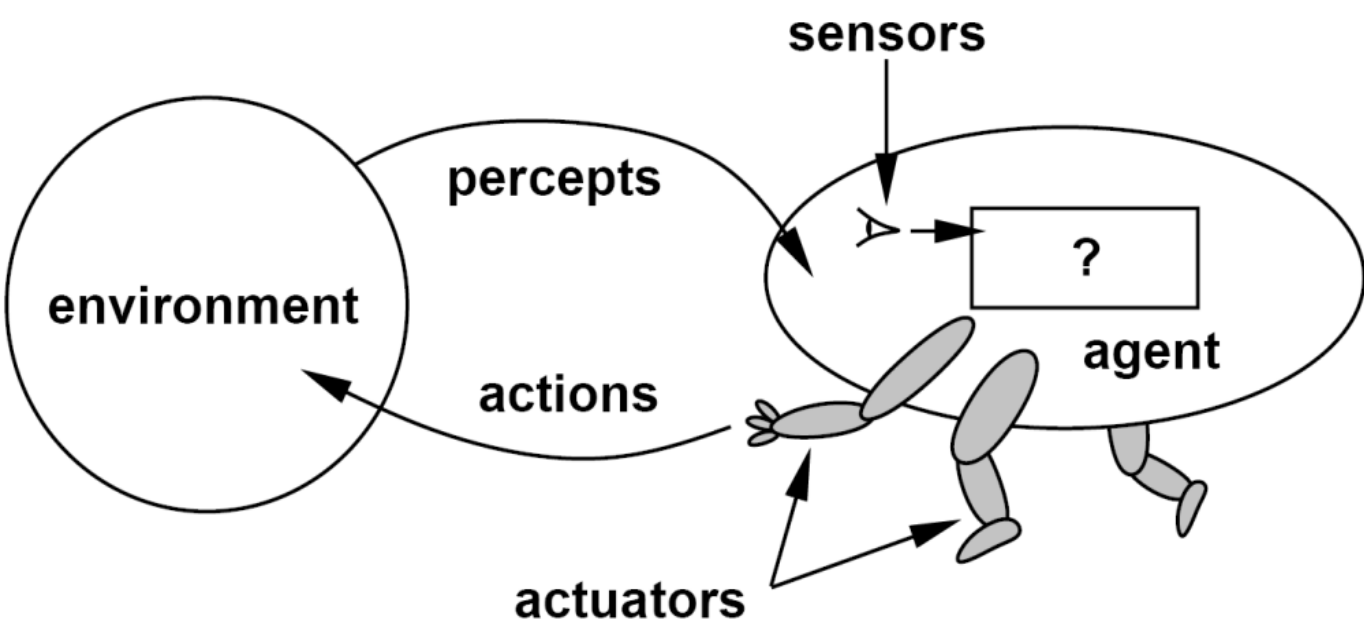
Why learning?

- Not fully specified problems.
  - e.g. unknown environments.
  - e.g. incomplete models.
- Vast amounts of data.
- *Why calibration*?
- Leverage "knowledge" of world itself.
- Adapt.

Successful Examples

- Learning to drive an autonomous vehicle.
- Learning to walk.
- Learning to classify objects.
- Learning to play world-class backgammon.
- Learning to do X better.

Recall the sense-action loop



Let's learn $f: P \rightarrow A$

A percept-action mapping

(simple reactive learning from demonstration)

BLG456E
Robotics
Intro to Reactive Robot Learning

Lecture Contents:
- Why learning.
- Supervised learning.
- Basic reactive controller learning.
- ROS Example

Lecturer: Damien Jade Duff
Email: djduff@itu.edu.tr
Office: EEBF 2316
Schedule: http://djduff.net/my-schedule
Coordination: http://ninova.itu.edu.tr/Ders/4709

Some slides taken/adapted from Dr. Sanem Sarıel-Talay

Kinds of learning

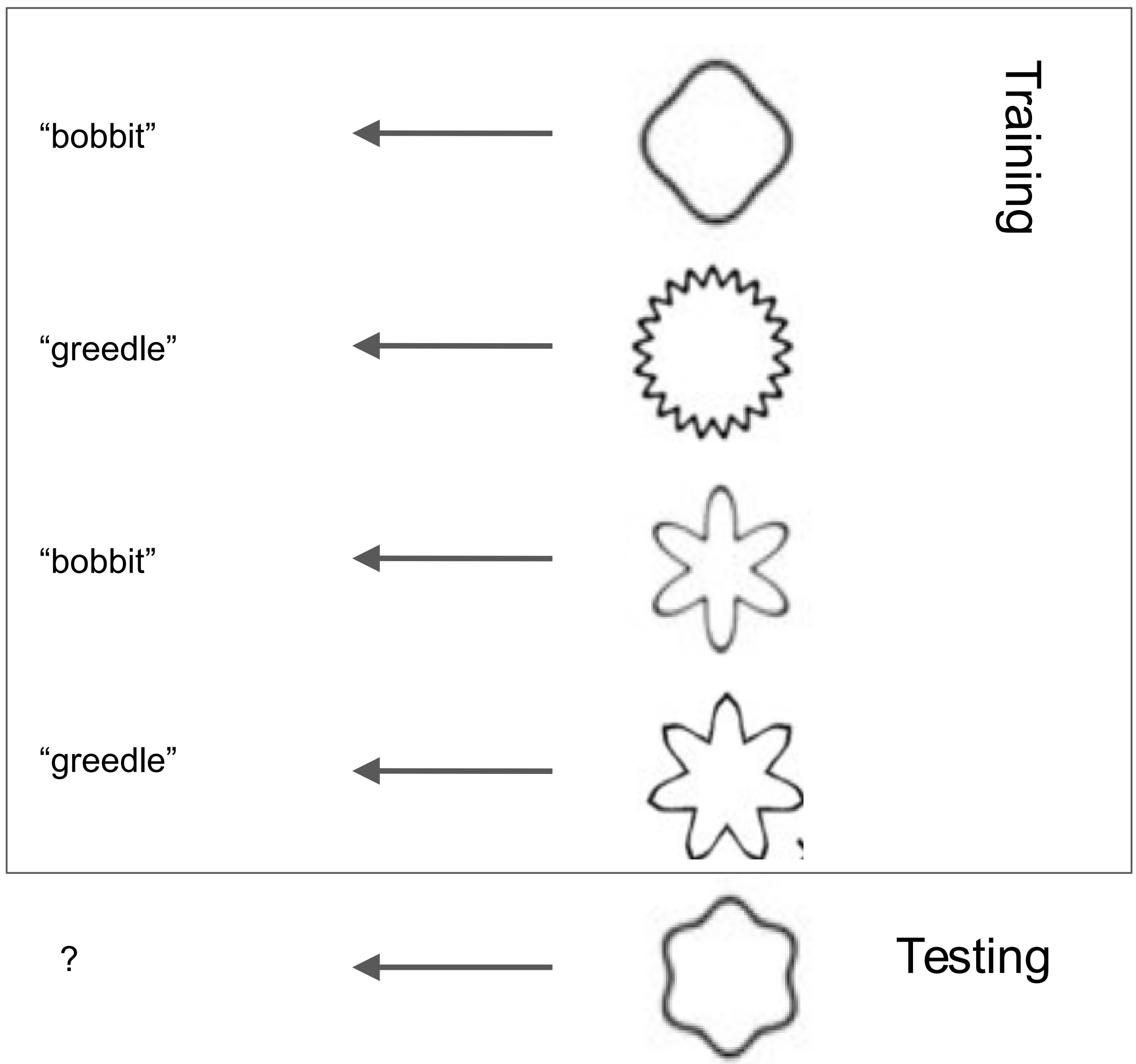
- "*Supervised*"
  - Output (Y) provided for input (X). Learn relationship.
- "*Unsupervised*"
  - Full input/output examples not provided (just X). Learn structure.
- "*Reinforcement learning*"
  - Only occasional feedback.

Supervised learning

- Given input/output pairs (X,Y).
- Learn a mapping f:$X \rightarrow Y$.
- e.g. Sense-action learning from demonstration.

iCub learning to catch:http://www.youtube.com/watch?v=EZDObbQYFFM

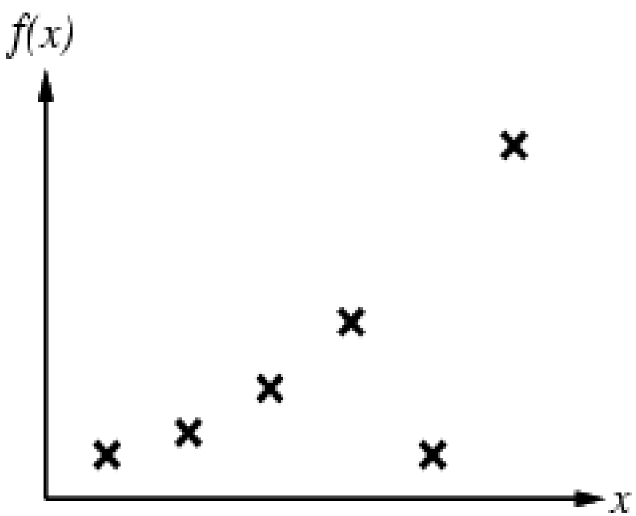
Supervised learning / classification: Learn a mapping from example pairs



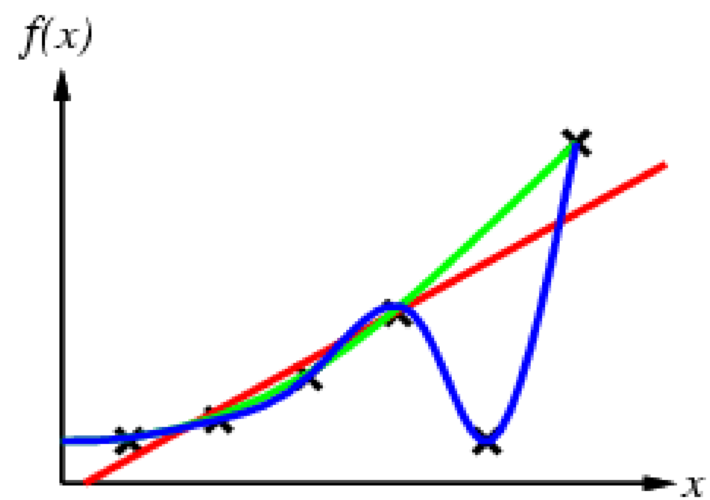
Supervised learning

- Construct hypothesis to agree with training set.
  - Consistent.
- Prefer simplest consistent hypothesis.
  - Ockham's razor.

X , Y examples



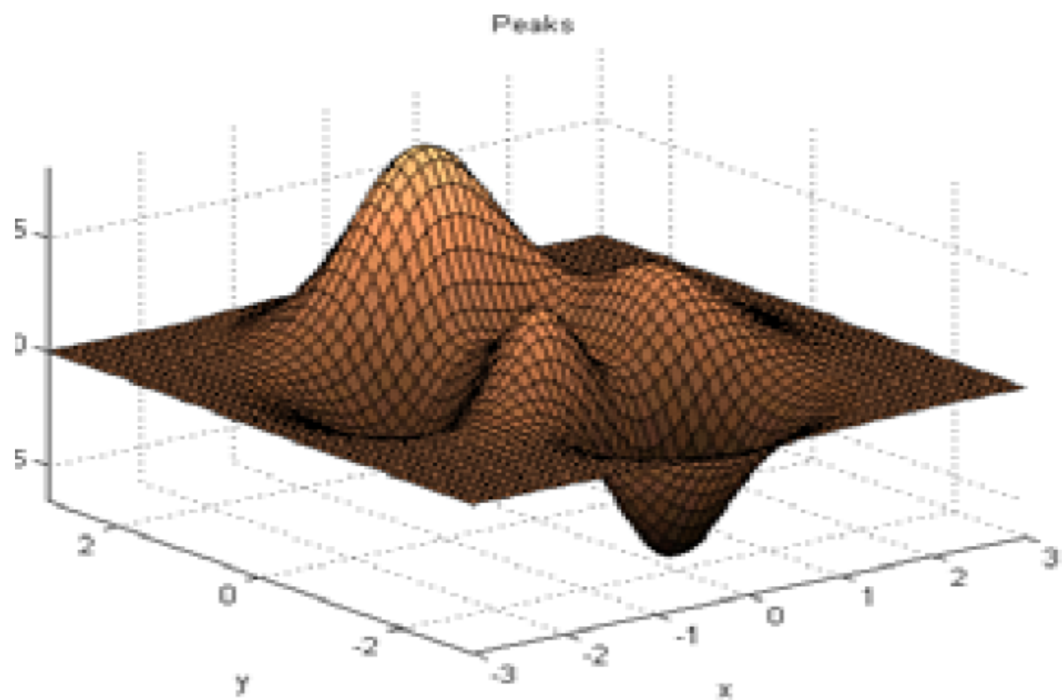
Some possible learnt functions
(hypotheses)



(curve fitting example)

Learning is searching

- All possible solutions exist in the solution space.
- Finding a good or optimal solution.



Each hypothesis curve can be a point in the search space

Representing sensory-action mapping

$$f : \boldsymbol{p} \rightarrow \boldsymbol{a}$$

f is a function from percepts to actions.

Percepts and actions
can be sets of numbers - vectors

e.g. $\boldsymbol{p} = \begin{bmatrix} 5.0 \\ -1.3 \end{bmatrix}$ $\boldsymbol{a} = \begin{bmatrix} 45 \\ 0.3 \end{bmatrix}$

2 laser ranges, 2 motors

Representing sensory-action mapping

e.g. $\boldsymbol{p} = \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} \quad \boldsymbol{a} = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}$

$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \boldsymbol{f}(p_1, p_2)$$

f must be parameterisable.

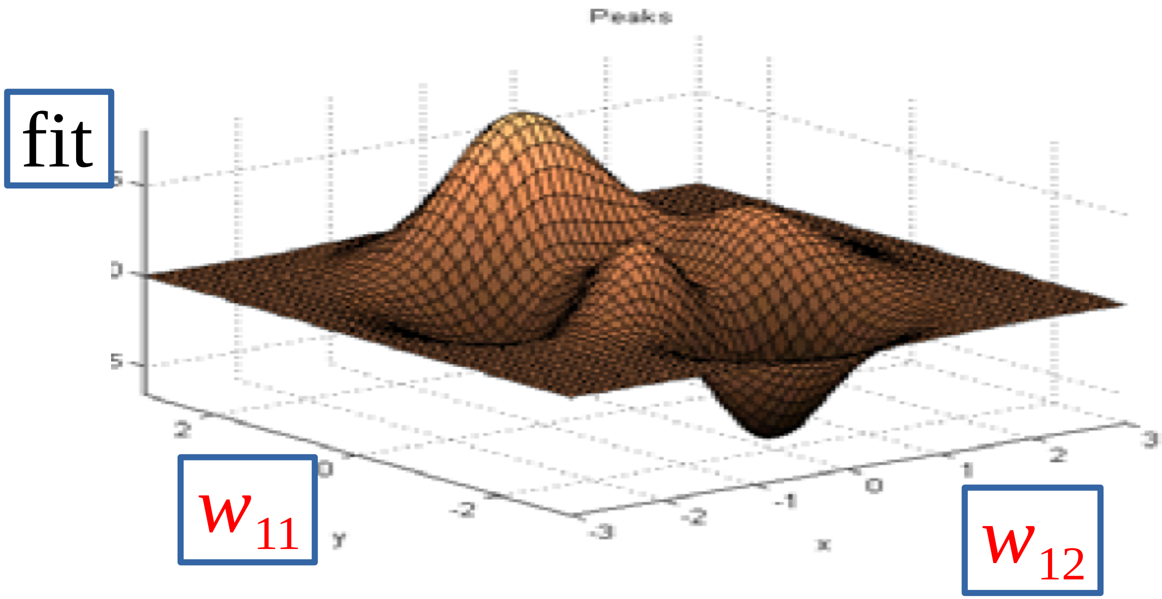
e.g.

$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \boldsymbol{f}(p_1, p_2) = \begin{bmatrix} w_{11} p_1 + w_{12} p_2 \\ w_{21} p_1 + w_{22} p_2 \end{bmatrix}$$

(linear function – but not very general)

Learning is searching for parameters of function

$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \boldsymbol{f}(p_1, p_2) = \begin{bmatrix} w_{11} p_1 + w_{12} p_2 \\ w_{21} p_1 + w_{22} p_2 \end{bmatrix}$$



Learning by reducing loss/error (improving fit)

For a certain percept:

$$\boldsymbol{p} = \begin{bmatrix} p_1 \\ p_2 \end{bmatrix}$$

Learner's current prediction:

$$\hat{\boldsymbol{a}} = \begin{bmatrix} \widehat{a}_1 \\ \widehat{a}_2 \end{bmatrix} = \boldsymbol{f}(p_1, p_2)$$

Observed action:

$$\boldsymbol{a} = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}$$

Error:

$$\epsilon = \|\boldsymbol{a} - \hat{\boldsymbol{a}}\|^2 = \left\| \begin{bmatrix} a_1 - \widehat{a}_1 \\ a_2 - \widehat{a}_2 \end{bmatrix} \right\|^2$$

$$= (a_1 - \widehat{a}_1)^2 + (a_2 - \widehat{a}_2)^2$$

$$= (a_1 - w_{11} p_1 - w_{12} p_2)^2 + (a_2 - w_{21} p_1 - w_{22} p_2)^2$$

Other possible representations of the function f?

- Overlapping bins (CMAC).
- Neural network.
- Exemplar-based (nearest-neighbour search).
- Kernel function.

…

Two NN frameworks

- FANN
  - C
- Keras
  - Python
  - Newer
  - Works with tensorflow/theano (deep learning)
  - Full boot camp lecture: http://files.djduff.net/nn.zip

Keras relevant functions: training

```
model = Sequential()

model.add(Dense(10, input_dim=inputX.shape[1], activation='relu',name='d1'))

model.add(Dense(2, activation='linear',name='f'))

model.compile(loss='mean_squared_error', optimizer='sgd')

model.fit(inputX,outputY, batch_size=512,epochs=60,verbose=1)
```

FANN relevant functions: training

fann* **fann_create_standard**(int nlayers,**int ninputs**,int nhidden,**int noutput**);

void fann_train(fann *ann, **float *input**,**float *output**);

```
void fann_set_activation_function_hidden(fann* ann, FANN_SIGMOID_SYMMETRIC);
void fann_set_activation_function_output(fann* ann, FANN_SIGMOID_SYMMETRIC);
```

Keras relevant functions: running

```
model.save("learnt_network.h5")

model = load_model('learnt_network.h5')

outputY = model.predict(inputX)
```

FANN relevant functions: running

void **fann_save**(fann *ann, "filename")

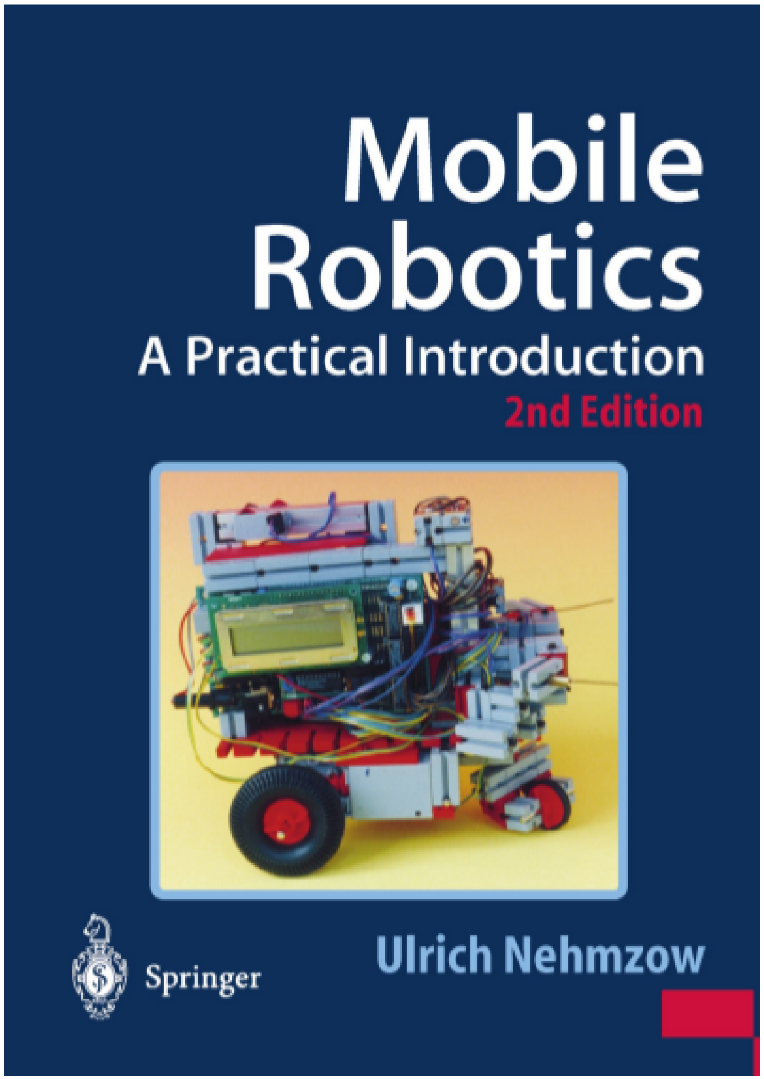
float* **fann_run**(fann *ann, **float *input**);

fann* **fann_create_from_file**("filename");

Problems with supervised learning

- Demonstrator not always available.
  - e.g. learning to run away from lions.
- Demonstration not always applicable.
  - e.g. human demonstrator with human shape.
- “Labelled data” not always available.
  - e.g. recognising all the objects in the world.

Readings II



Ulrich Nehmzow (2003).

Mobile Robotics: A Practical Introduction.

Available from
http://divit.library.itu.edu.tr/record=b1677494*tur

Chapter 4:
Robot Learning: Making Sense of Raw Sensor Data