Machine Learning for Cities
CUSP-GX 5006.001, Spring 2018

Course Overview

Instructor: Daniel B. Neill (daniel.neill@nyu.edu)

Teaching Assistant: Vamsi Ramineedi (vmr286@nyu.edu)

Why machine learning?

Critical importance of addressing urban challenges: disease, crime, terrorism, poverty, environment…

Increasing size and complexity of available data, thanks to the rapid growth of new and transformative technologies.

Much more computing power, and scalable data analysis methods, enable us to extract actionable information from all of this data.

Machine learning techniques have become increasingly essential for urban policy analysis, and for the development of new, practical information technologies that can be directly applied to address critical urban challenges.

Some motivating examples

Early detection of
emerging disease
outbreaks

Discovering new
“best practices”
for patient care

Substance abuse
and overdose
surveillance

Preventing rat infestations
(using “311” service calls)

Predicting civil unrest
(using Twitter data)

Preventing violent crime
(in Chicago & Pittsburgh)

What is machine learning?

Machine Learning (ML) is the study of systems that improve their performance with experience (typically by **learning** from data).

“A computer program is said to learn from experience E wrt. some class of tasks T and performance measure P, if its performance at tasks in T as measured by P improves with experience.” (T. Mitchell)

“Learning denotes changes in the system that are adaptive in the sense that they enable it to do a task, or tasks drawn from the same population, more efficiently and effectively next time.” (H. Simon)

Learning as **generalization**: the ability to perform a task in a situation which has never been encountered before!

ML vs. computer programming

Machine Learning (ML) is the study of systems that improve their performance with experience (typically by **learning** from data).

“A computer program is said to learn from experience E wrt. some class of tasks T and performance measure P, if its performance at tasks in T as measured by P improves with experience.” (T. Mitchell)

“Learning denotes changes in the system that are adaptive in the sense that they enable it to do a task, or tasks drawn from the same population, more efficiently and effectively next time.” (H. Simon)

Learning as **generalization**: the ability to perform a task in a situation which has never been encountered before!

“The Analytical Engine has no pretensions whatever to originate anything. It can do whatever we know how to order it to perform. It can follow analysis; but it has no power of anticipating any analytical relations or truths.” (A. Lovelace, 1842)

ML vs. computer programming

Machine Learning (ML) is the study of systems that improve their performance with experience (typically by **learning** from data).

“A computer program is said to learn from experience E wrt. some class of tasks T and performance measure P, if its performance at tasks in T as measured by P improves with experience.” (T. Mitchell)

“Learning denotes changes in the system that are adaptive in the sense that they enable it to do a task, or tasks drawn from the same population, more efficiently and effectively next time.” (H. Simon)

Learning as **generalization**: the ability to perform a task in a situation which has never been encountered before!

“Programming computers to learn from experience should eventually eliminate the need for much of this detailed programming effort.” (A. Samuel, 1959)

Samuel coined the term “machine learning” and was best known for his self-learning checkers program.

ML vs. measly humans

Machine Learning (ML) is the study of systems that improve their performance with experience (typically by **learning** from data).

“A computer program is said to learn from experience E wrt. some class of tasks T and performance measure P, if its performance at tasks in T as measured by P improves with experience.” (T. Mitchell)

“Learning denotes changes in the system that are adaptive in the sense that they enable it to do a task, or tasks drawn from the same population, more efficiently and effectively next time.” (H. Simon)

Learning as **generalization**: the ability to perform a task in a situation which has never been encountered before!

Computers now consistently beat the top players in the world at checkers (1995), chess (1997), and Go (2016), as well as the game show Jeopardy (2011).

ML vs. measly humans

Machine Learning (ML) is the study of systems that improve their performance with experience (typically by **learning** from data).

“A computer program is said to learn from experience E wrt. some class of tasks T and performance measure P, if its performance at tasks in T as measured by P improves with experience.” (T. Mitchell)

“Learning denotes changes in the system that are adaptive in the sense that they enable it to do a task, or tasks drawn from the same population, more efficiently and effectively next time.” (H. Simon)

Learning as **generalization**: the ability to perform a task in a situation which has never been encountered before!

ML methods have had tremendous success on tasks including control (self-driving cars), image recognition, speech recognition, recommender systems, machine translation, etc.

CMU’s “Boss”, winner of the DARPA Urban Challenge

Examples of performance metrics

Machine Learning (ML) is the study of systems that improve their performance with experience (typically by **learning** from data).

"A computer program is said to learn from experience E wrt. some class of tasks T and performance measure P, if its performance at tasks in T as measured by P improves with experience." (T. Mitchell)

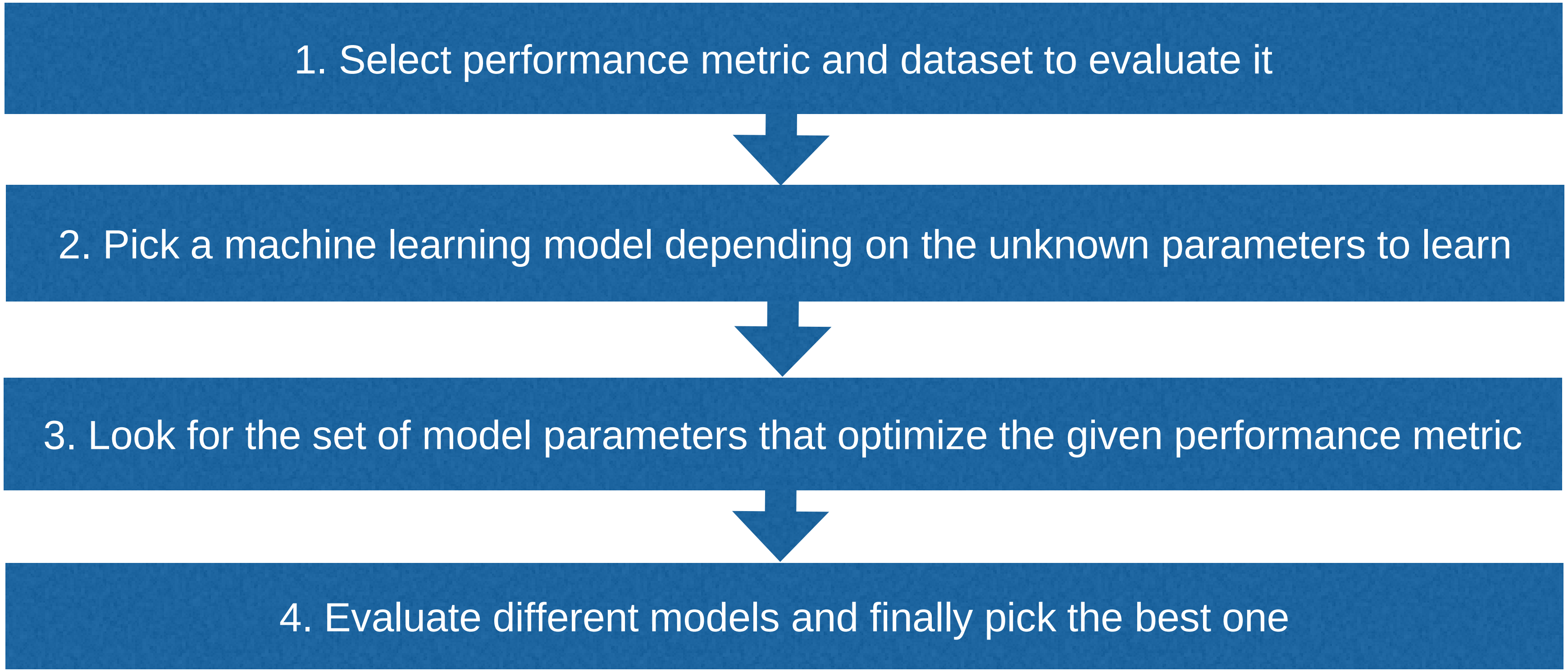
Task	Performance metric	Experience
Play checkers	Percentage of wins vs. given opponent	Games previously played w/ outcomes
Recognize handwritten digits	Percentage of correct recognitions	Set of digit writing w/ labels
Control a self-driving car	Average speed in given conditions provided that safety standards are met	Previous driving record w/ evaluation
Predict stock prices	Average prediction accuracy	History of stock prices

Table credit: Stanislav Sobolevsky

ML as optimization

Machine Learning (ML) is the study of systems that improve their performance with experience (typically by **learning** from data).

“A computer program is said to learn from experience E wrt. some class of tasks T and performance measure P, if its performance at tasks in T as measured by P improves with experience.” (T. Mitchell)



1. Select performance metric and dataset to evaluate it

2. Pick a machine learning model depending on the unknown parameters to learn

3. Look for the set of model parameters that optimize the given performance metric

4. Evaluate different models and finally pick the best one

Figure credit: Stanislav Sobolevsky

ML draws from many disciplines

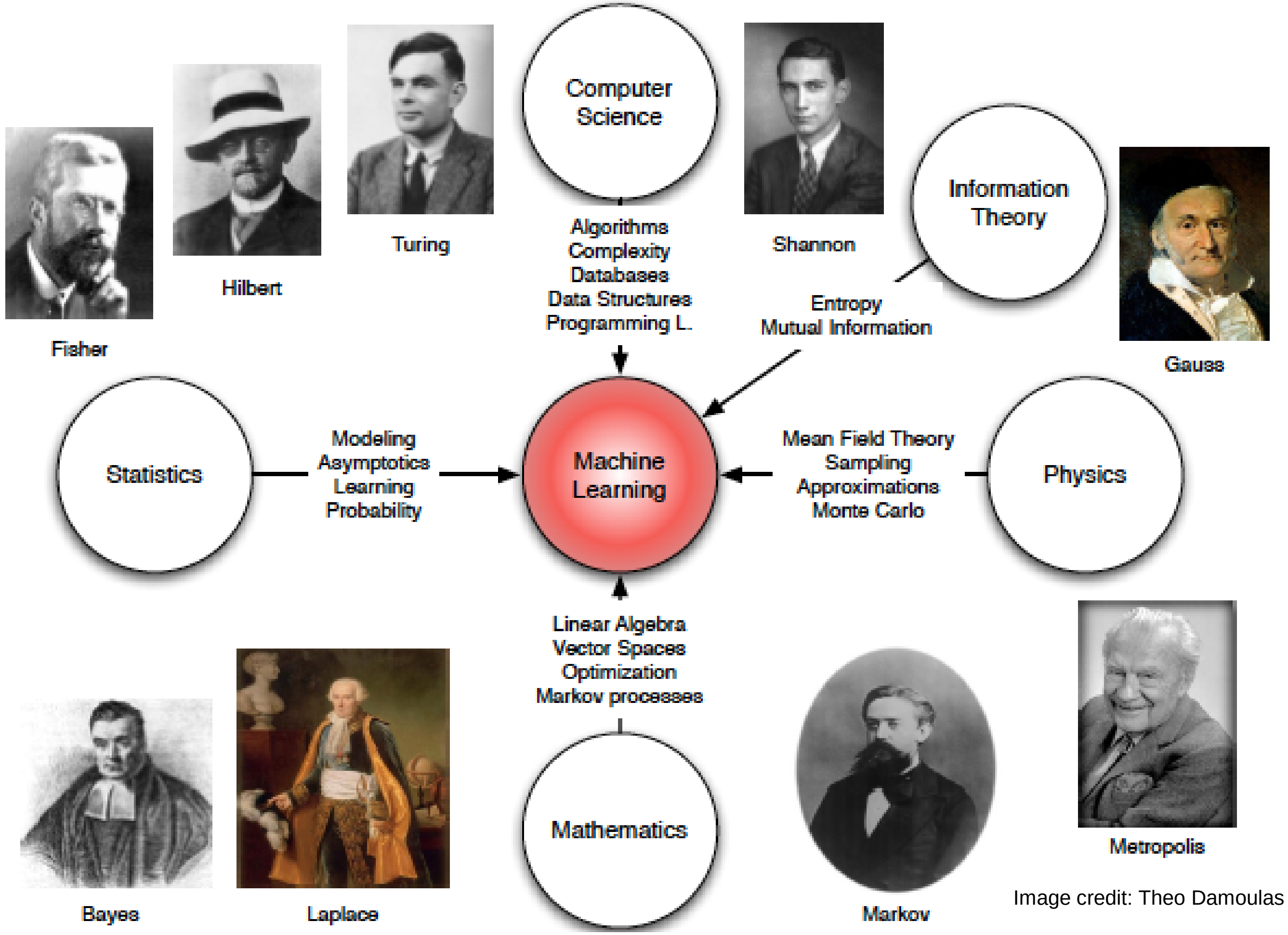


Image credit: Theo Damoulas

Also cognitive psychology, evolution, economics, neuroscience, and many more!

ML and related fields

Machine Learning (ML) is the study of systems that improve their performance with experience (typically by **learning** from data).

Artificial Intelligence (AI) is the science of automating complex behaviors that normally require human intelligence: vision, language understanding, learning, problem solving, decision making, etc.

Data Mining (DM) is the process of extracting useful information from massive quantities of complex data.

I would argue that these are not three distinct fields of study! While each has a slightly different emphasis, there is a tremendous amount of overlap in the problems they are trying to solve and the techniques used to solve them.

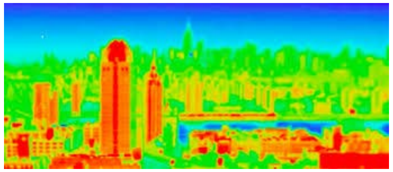
Many of the techniques we will learn are **statistical** in nature, but are very different from classical statistics.

ML/AI/DM systems and methods:

Scale up to large, complex data
Learn and **improve** from experience
Perceive and **change** the environment
Interact with humans or other agents
Explain inferences and decisions
Discover new and useful patterns

Urban applications of ML

- Inferring urban dynamics from heterogeneous data
- Computer vision: pedestrian/traffic counts, security/law enforcement (face recognition), traffic accident detection, remote sensing (air content, IR, etc.)
- Street noise (decomposition, localization, classification)
- Economic patterns detection and prediction
- Health pattern detection and prediction
- Energy usage prediction
- Traffic modeling and prediction
- Land use classification
- 3-D landscape recognition
- Event detection from urban activity
- Detecting trends from social media



Infrared data from CUSP’s Urban Observatory

Course objectives

Goals: familiarize students with modern machine learning methods and show how they can be effectively applied to urban data.

More precisely, this course is intended to enable students to:

1) Understand the **motivation** behind different machine learning methods and their **applicability** in a given practical context.
2) Be able to **implement** methods adapted to the problem at hand using existing software libraries.*
3) Know how to **interpret** the results appropriately.

* This course includes a significant programming component, with **Python** as the primary programming language.

Each class session will typically consist of 90 minutes of lecture, followed by a 1-hour hands-on lab session. Please bring your laptop and follow along.

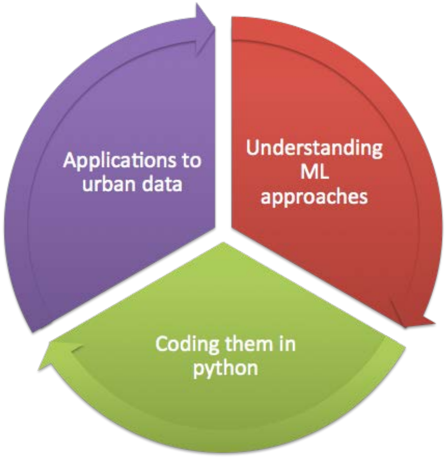


Image credit: Stanislav Sobolevsky

Structure of the course

- 12 lectures (with labs)
  - Combination of core ML methods and ML topics most relevant to urban data analysis
  - Motivating examples and applications.
- Midterm exam and final project presentation.
- First half of course: mainly classification and clustering. Core ML but focus on accuracy vs. interpretability tradeoff.
- Second half of course: ML methods to address some of the unique challenges of urban data.

Interconnected

Spatial

Complex structure

Interconnected

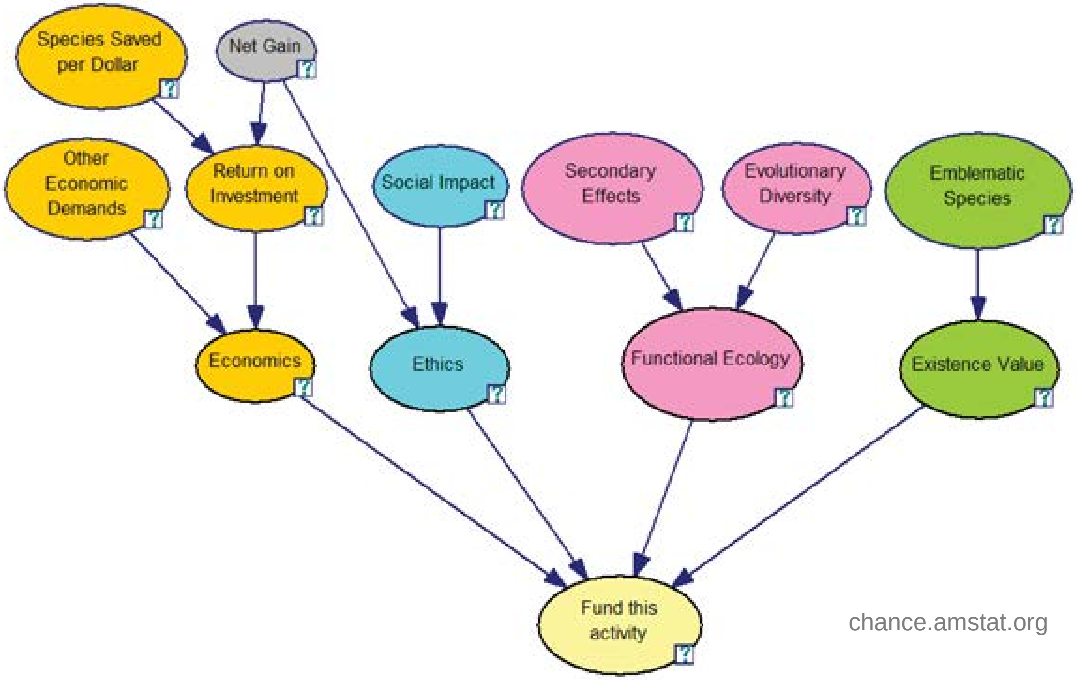
Urban systems consist of many complex, interconnected sub-systems (energy, transportation, water, …)

Early detection of events in one system could **predict** events in another, both at shorter time scales and longer time scales.

We will use **Bayesian networks** to model the dependencies between multiple variables and to infer causal relationships.

Interconnected

Urban systems consist of many complex, interconnected sub-systems (energy, transportation, water, …)

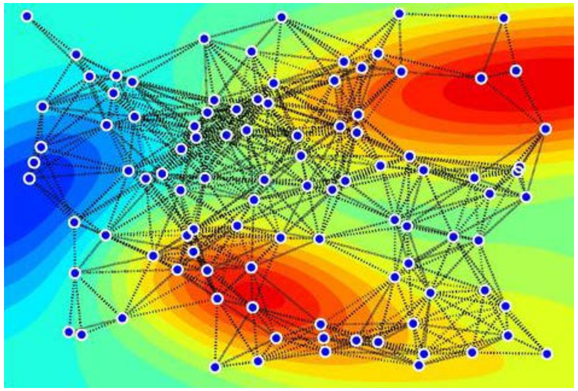


chance.amstat.org

We will use **Bayesian networks** to model the dependencies between multiple variables and to infer causal relationships.

Spatial

Nearby observations tend to be **correlated** so typical assumption of i.i.d. data fails.



We will use **Gaussian Processes** to model and make predictions for spatial and other dependent data.

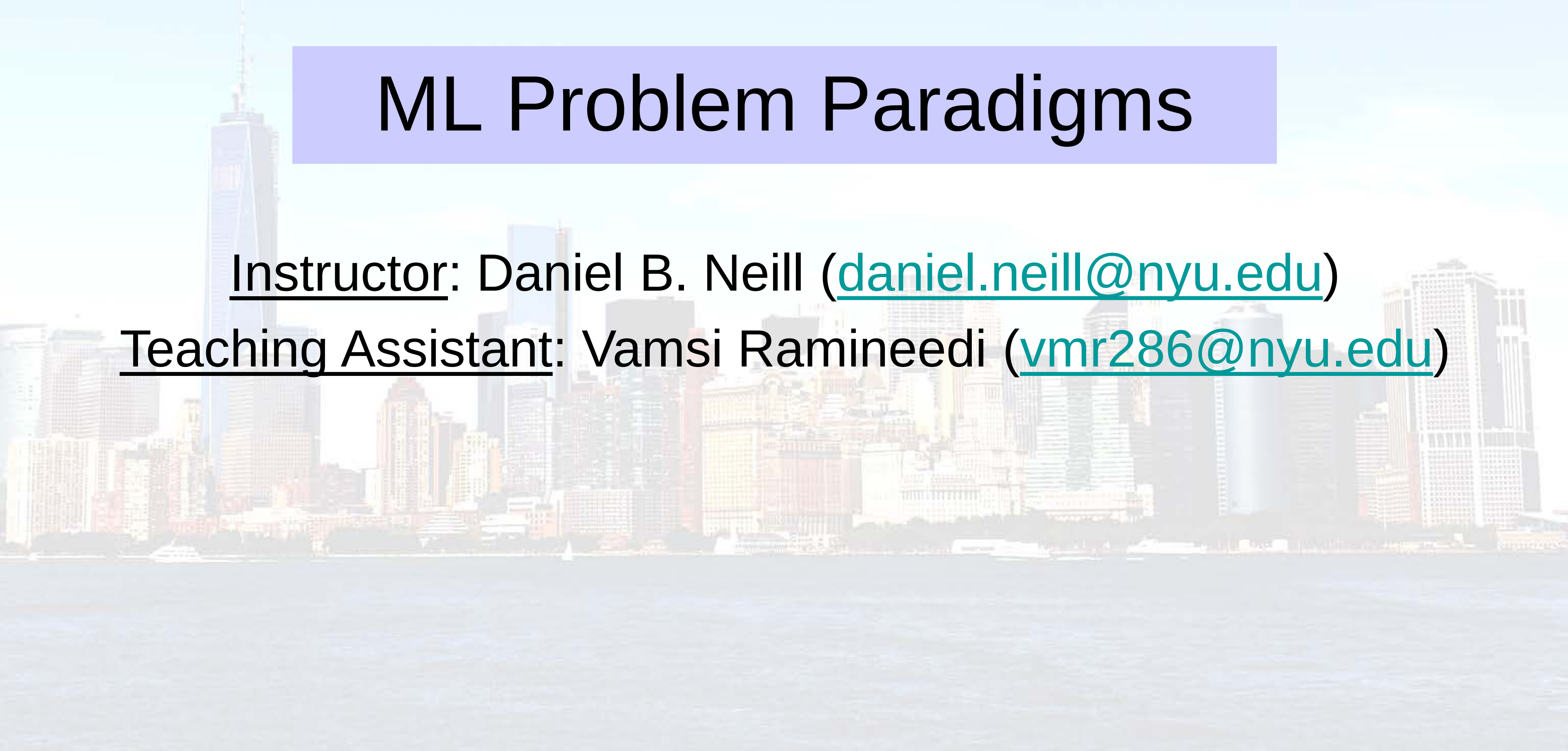
Events tend to affect subsets of the data that are **localized** in space and time.

We will learn how to **detect** anomalies, events, and other patterns in data.

Key idea: a subset of the data, such as several nearby locations, may be **collectively** interesting or anomalous, even if no individual data element is sufficiently interesting on its own.

Structure of the course

- Grades will be based on:
  - Midterm exam (20%)
  - Four homework assignments (10% x 4 = 40%)
  - Final project report (30%) & presentation (10%)
- See syllabus on NYU Classes for:
  - Instructor and TA info (e-mails, office hours, etc.)
  - Detailed course schedule (subject to change!)
  - Description of projects and problem sets
  - Course policies (cheating, late work)



Machine Learning for Cities
CUSP-GX 5006.001, Spring 2018

ML Problem Paradigms

Instructor: Daniel B. Neill (daniel.neill@nyu.edu)

Teaching Assistant: Vamsi Ramineedi (vmr286@nyu.edu)

Common ML paradigms: prediction

In **prediction**, we are interested in explaining a specific attribute of the data in terms of the other attributes.

Classification: predict a discrete value

“What disease does this patient have, given his symptoms?”

Regression: estimate a numeric value

“How is a city’s literacy rate affected by various educational programs?”

Two main goals of prediction

Guessing unknown values for specific instances (e.g. diagnosing a given patient)

Explaining predictions of both known and unknown instances (providing relevant examples, a set of decision rules, or class-specific models).

Example 1: What socio-economic factors lead to increased prevalence of diarrheal illness in a developing-world city?

Example 2: Developing a system to predict whether, where, and when traffic congestion will emerge and spread, and impacts on travel time.

Data set representation

Our dataset consists of a set of **data records** $\{x_i\}$.

Each record has values for a set of **attributes** $\{A_j\}$.

Each data record x_i has a **value** v_{ij} for each attribute A_j.

	A_1 Name	A_2 Gender	A_3 BMI	A_4 Systolic BP	A_5 Diastolic BP	A_6 Diabetes?	A_7 Heart attack risk?
x_1	Bob	Male	37	205	150	Yes	High
x_2	Kathy	Female	23	125	80	No	Low
x_3	John	Male	24	150	80	No	???

Attributes can be real-valued (a number) or discrete-valued (a class).

Some attribute values may be missing (represented here by ???).

The prediction problem

Our dataset consists of a set of **data records** $\{x_i\}$.

Each record has values for a set of **attributes** $\{A_j\}$.

Each data record x_i has a **value** v_{ij} for each attribute A_j.

	A_1 Name	A_2 Gender	A_3 BMI	A_4 Systolic BP	A_5 Diastolic BP	A_6 Diabetes?	A_7 Heart attack risk?
x_1	Bob	Male	37	205	150	Yes	High
x_2	Kathy	Female	23	125	80	No	Low
x_3	John	Male	24	150	80	No	???

The goal of prediction is to guess the missing value of some attribute for a given data point, given the other attributes for that point, as well as the rest of the dataset.

The prediction problem

Our dataset consists of a set of **data records** $\{x_i\}$.

Each record has values for a set of **attributes** $\{A_j\}$.

Each data record x_i has a **value** v_{ij} for each attribute A_j.

	A_1 Name	A_2 Gender	A_3 BMI	A_4 Systolic BP	A_5 Diastolic BP	A_6 Diabetes?	A_7 Heart attack risk?
x_1	Bob	Male	37	205	150	Yes	High
x_2	Kathy	Female	23	125	80	No	Low
x_3	John	Male	24	150	80	No	???

If we are predicting a discrete value (e.g. heart attack risk), this is a classification problem.

If we are predicting a real value (e.g. blood pressure), this is a regression problem.

The prediction problem

Our dataset consists of a set of **data records** $\{x_i\}$.

Each record has values for a set of **attributes** $\{A_j\}$.

Each data record x_i has a **value** v_{ij} for each attribute A_j.

	A_1 Name	A_2 Gender	A_3 BMI	A_4 Systolic BP	A_5 Diastolic BP	A_6 Diabetes?	A_7 Heart attack risk?
x_1	Bob	Male	37	205	150	Yes	High
x_2	Kathy	Female	23	125	80	No	Low
x_3	John	Male	24	150	80	No	???

Let A_p denote the attribute we are trying to predict. Assume that all records either a) have no missing values, or b) have only A_p missing. We call the first set **training records**, and the second set **test records**.

The prediction problem

Our dataset consists of a set of **data records** $\{x_i\}$.

Each record has values for a set of **attributes** $\{A_j\}$.

Each data record x_i has a **value** v_{ij} for each attribute A_j.

	A_1 Name	A_2 Gender	A_3 BMI	A_4 Systolic BP	A_5 Diastolic BP	A_6 Diabetes?	A_7 Heart attack risk?
x_1	Bob	Male	37	205	150	Yes	High
x_2	Kathy	Female	23	125	80	No	Low
x_3	John	Male	24	150	80	No	???

Our goal is to accurately predict the missing values of A_p for each test record, using the training data.

Classification

Maximize proportion of correct predictions

Regression

Minimize mean squared error

Common ML paradigms: modeling

In **modeling**, we are interested in describing the underlying relationships between many attributes and many entities.

Our goal is to produce models of the "entire data" (not just specific attributes or examples) that accurately reflect underlying complexity, yet are simple, understandable by humans, and usable for decision-making.

Relations between entities

Identifying link, group, and network structures

Partitioning or "clustering" data into subgroups

Relations between variables

Identifying significant positive and negative correlations

Modeling dependence structure between multiple variables

Example: Can we model the dependencies between multiple diet-related risk factors and health outcomes?

Common ML paradigms: detection

In **detection**, we are interested in identifying relevant patterns in massive, complex datasets.

Main goal: focus the user's attention on a potentially relevant subset of the data.

a) Automatically detect relevant individual records, or groups of records.

b) Characterize and explain the pattern (type of pattern, H_0 and H_1 models, etc.)

c) Present the pattern to the user.

Some common detection tasks

Detecting **anomalous** records or groups

Discovering **novelties** (e.g. new drugs)

Detecting **clusters** in space or time

Removing **noise** or **errors** in data

Detecting **specific patterns** (e.g. fraud)

Detecting emerging **events** which may require rapid responses.

Example 1: Detect emerging outbreaks of disease using electronic health data from hospitals and pharmacies.

Example 2: Detect patterns of similar crimes that may have been committed by the same perpetrators.

Overview of ML approaches

ML problem paradigms represent a **functional** grouping of methods by what we’re trying to accomplish. A related grouping is based on what the data looks like, and in particular, whether we have **labeled** or **unlabeled** data.

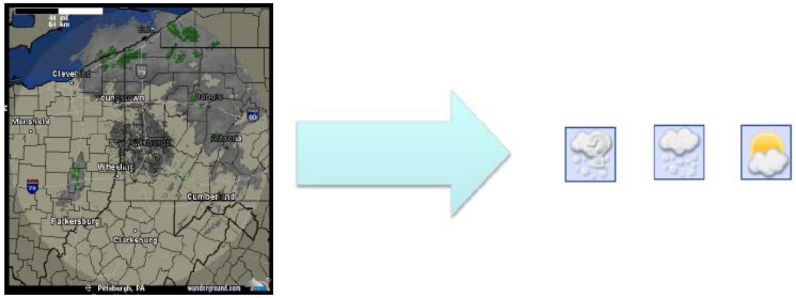
Supervised Learning

Data/input	Labels/output
x_1	y_1
x_2	y_2
$\dots$	$\dots$
x_N	y_N

Learn dependence:

$$y = f(x)$$

Discrete y = classification
Continuous y = regression



Overview of ML approaches

ML problem paradigms represent a **functional** grouping of methods by what we're trying to accomplish. A related grouping is based on what the data looks like, and in particular, whether we have **labeled** or **unlabeled** data.

Supervised Learning

Data/input	Labels/output
x_1	y_1
x_2	y_2
$\dots$	$\dots$
x_N	y_N

Learn dependence:

$$y = f(x)$$

Discrete y = classification
Continuous y = regression

Semi-supervised learning:

Only some data points are labeled; the goal is still typically prediction.

Active learning:

Choose which data points to label; the goal is still typically prediction.

Reinforcement learning:

Sequential actions with delayed rewards; goal is to learn optimal action in each state.

Unsupervised learning:

No labels, just input data x_i.
Various goals including clustering, modeling, anomaly detection, etc.

Supervised learning in basic stats

ML problem paradigms represent a **functional** grouping of methods by what we're trying to accomplish. A related grouping is based on what the data looks like, and in particular, whether we have **labeled** or **unlabeled** data.

Supervised Learning

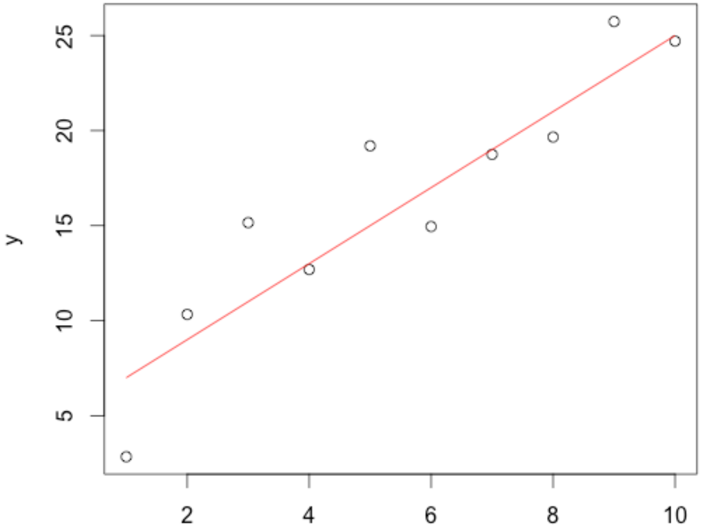
Data/input	Labels/output
x_1	y_1
x_2	y_2
$\dots$	$\dots$
x_N	y_N

Learn dependence:

$$y = f(x)$$

Discrete y = classification
Continuous y = regression

Linear regression



$$y = w_1 x + w_0 + \varepsilon$$

$$y = w^T x + \varepsilon$$

$$p(y|x, w, \sigma) = \mathcal{N}(y|w^T x, \sigma^2)$$

Supervised learning in basic stats

ML problem paradigms represent a **functional** grouping of methods by what we're trying to accomplish. A related grouping is based on what the data looks like, and in particular, whether we have **labeled** or **unlabeled** data.

Supervised Learning

Data/input

x_1
x_2
$\dots$
x_N

Labels/output

y_1
y_2
$\dots$
y_N

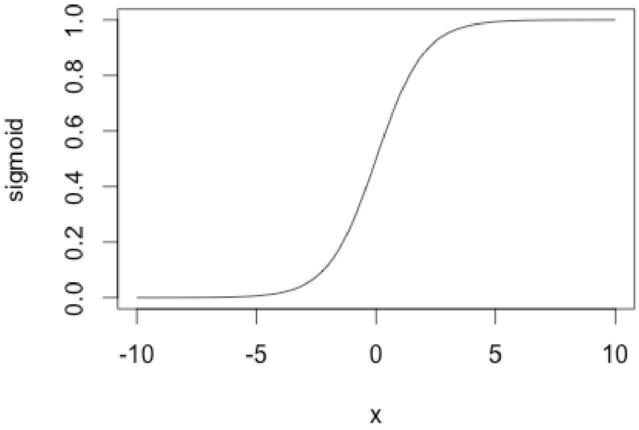
Learn dependence:

$$y = f(x)$$

Discrete y = classification
Continuous y = regression

Logistic regression

(= generalized LR for classification)



$$y \sim Bernoulli\,(f(w^T x))$$

$$f(x) = \sigma(x) = \frac{e^x}{1+e^x} = \frac{1}{1+e^{-x}}$$

Supervised learning in this course

ML problem paradigms represent a **functional** grouping of methods by what we're trying to accomplish. A related grouping is based on what the data looks like, and in particular, whether we have **labeled** or **unlabeled** data.

Supervised Learning

Data/input	Labels/output
x_1	y_1
x_2	y_2
$\cdots$	$\cdots$
x_N	y_N

Learn dependence:

$$y = f(x)$$

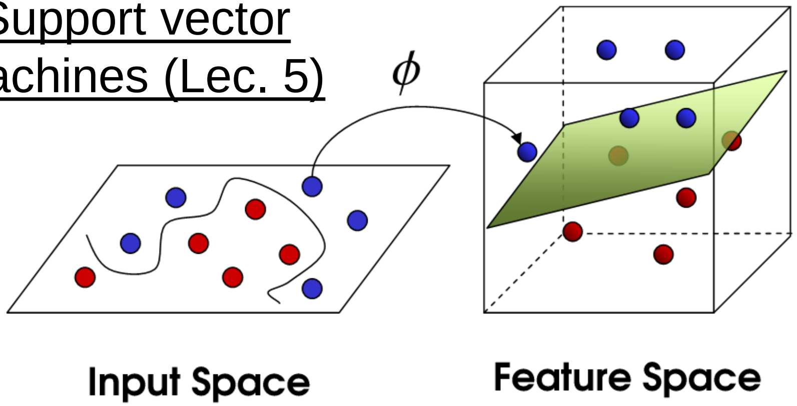
Discrete y = classification
Continuous y = regression

We'll learn about a variety of other prediction approaches, ranging from the simple and interpretable (decision trees, naïve Bayes) to the highly accurate but less interpretable (random forests, support vector machines).

Supervised learning in this course

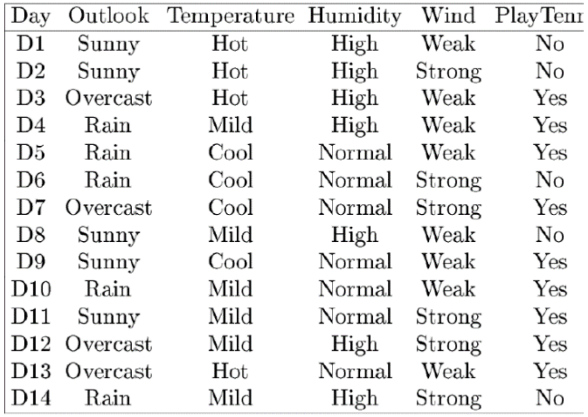
ML problem paradigms represent a **functional** grouping of methods by what we're trying to accomplish. A related grouping is based on what the data looks like, and in particular, whether we have **labeled** or **unlabeled** data.

Support vector machines (Lec. 5)

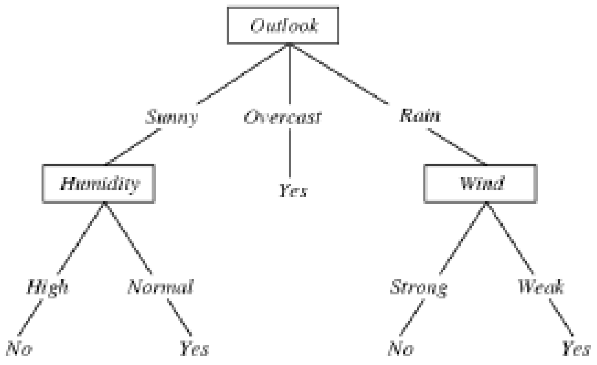


We'll learn about a variety of other prediction approaches, ranging from the simple and interpretable (decision trees, naïve Bayes) to the highly accurate but less interpretable (random forests, support vector machines).

Learning a decision tree (Lec. 2)

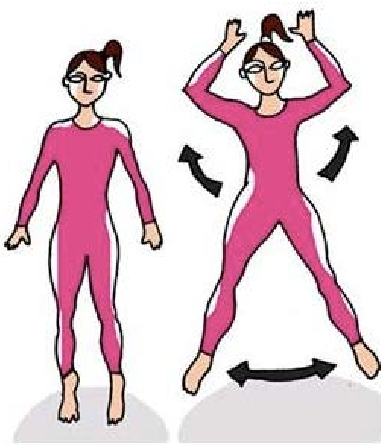


Day	Outlook	Temperature	Humidity	Wind	PlayTen
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No



The rest of today’s class period…

- Questions?
- The all-important 5-10 minute break.
    - Jumping jacks are highly recommended (but not required).



- Review of Python programming
    - Today: basic Python
    - Next week: data science packages (numpy, pandas, sklearn)
    - After that: ML methods and urban applications

```
In [9]: display(i)
```

```
In [3]: from IPython.display import SVG
        SVG(filename='python-logo.svg')

Out[3]:
```