UNIVERSITY OF TORONTO
SCHOOL OF CONTINUING STUDIES

3253 - Analytic Techniques and Machine Learning

Module 1: Introduction to Machine Learning

Course Plan

Module Titles
Module 1 – Current Focus: Introduction to Machine Learning
Module 2 – End to End Machine Learning Project
Module 3 – Classification
Module 4 – Clustering and Unsupervised Learning
Module 5 – Training Models and Feature Selection
Module 6 – Support Vector Machines
Module 7 – Decision Trees and Ensemble Learning
Module 8 – Dimensionality Reduction
Module 9 – Introduction to TensorFlow
Module 10 – Introduction to Deep Learning and Deep Neural Networks
Module 11 – Distributing TensorFlow, CNNs and RNNs
Module 12 – Final Assignment and Presentations (no content)

Learning Outcomes for this Module

- Define Machine Learning
- Consider when Machine Learning is applicable
- Enumerate the types of Machine Learning
- Discuss challenges of Machine Learning
- Begin to apply Machine Learning tools and techniques

Topics for this Module

- **1.1** What is machine learning?
- **1.2** Why use machine learning?
- **1.3** Types of machine learning
- **1.4** Modeling
- **1.5** Challenges of machine learning
- **1.6** Tools & Techniques
- **1.7** Resources and Wrap-up

Certificate in Data Science

- Understand the techniques and methods of predictive and Big Data analytics
- Learn how to use tools such as Python and Hadoop to tackle data analysis challenges
- Develop and use models tools to solve business problems and mine data for fresh insights

Certificate in Data Science (Cont'd)

What You'll Learn

- Explore the evolution of data science and predictive analytics
- Know statistical concepts and techniques including regression, correlation and clustering
- Apply data management systems and technologies that reflect concern for security and privacy
- Adopt techniques and technologies including data mining, neural network mapping and machine learning
- Represent big data findings visually to aid decision-makers

Certificate in Data Science (Cont'd)

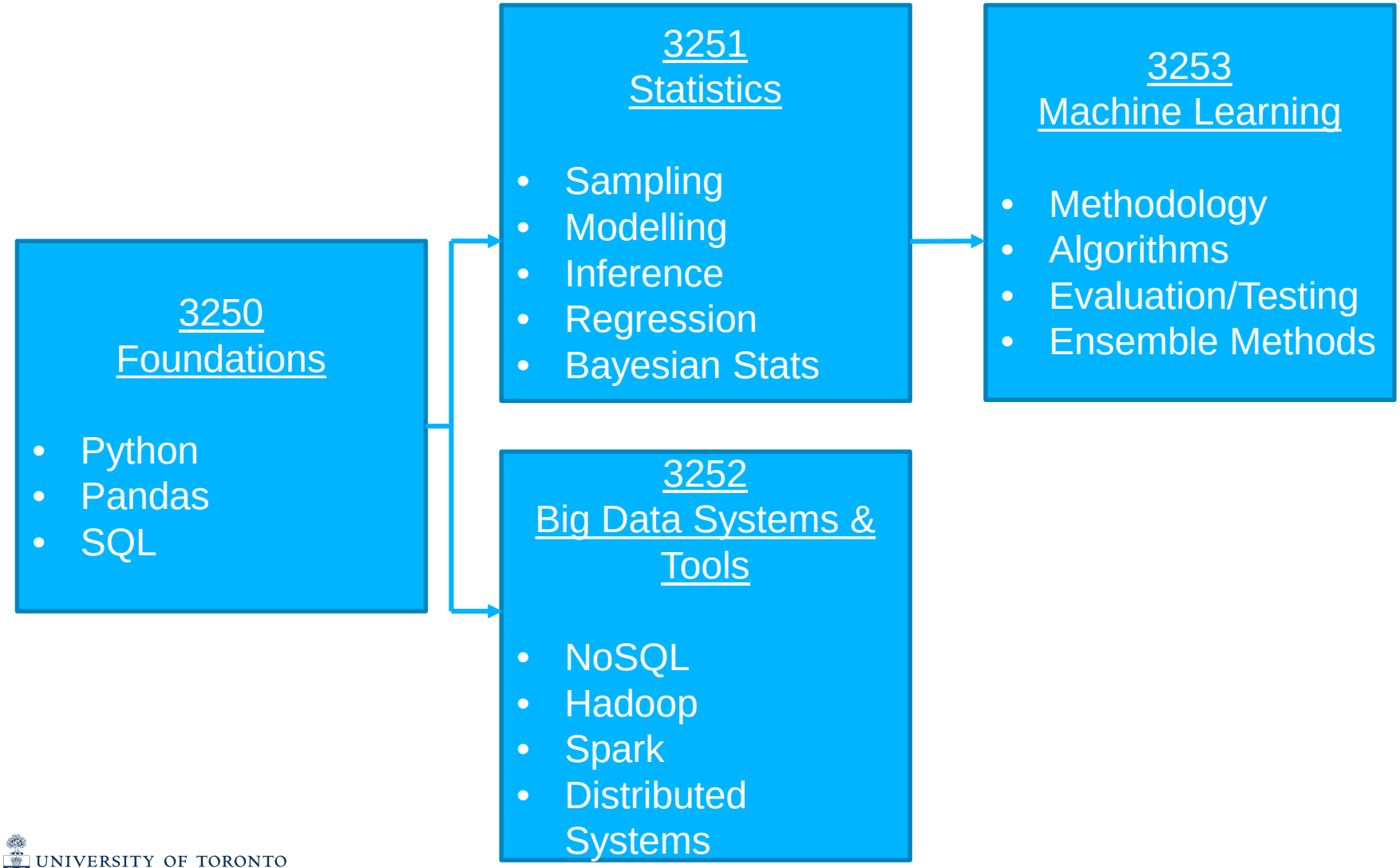
Courses

- SCS 3250 – Foundations of Data Science
- SCS 3251 – Statistics for Data Science
- SCS 3252 – Big Data Management Systems & Tools
- **SCS 3253 – Machine Learning**

Prerequisites

- This course assumes that you are comfortable programming in Python and that you are familiar with Python's main scientific libraries, in particular NumPy, Pandas, and Matplotlib
- Experience with notebook environments
- Understanding of college-level math as well (calculus, linear algebra, probabilities, and statistics)

Certificate in Data Science Fundamentals (Cont'd)



Certified Analytics Professional

- Industry Certification
- Operated by INFORMS, the world's largest professional society for those in the field of analytics, operations research (O.R.), and the management sciences
- Requires experience doing analytics and a related degree (or equivalent additional experience)
- Code of ethics

The CAP Domains

Coverage in this certificate program

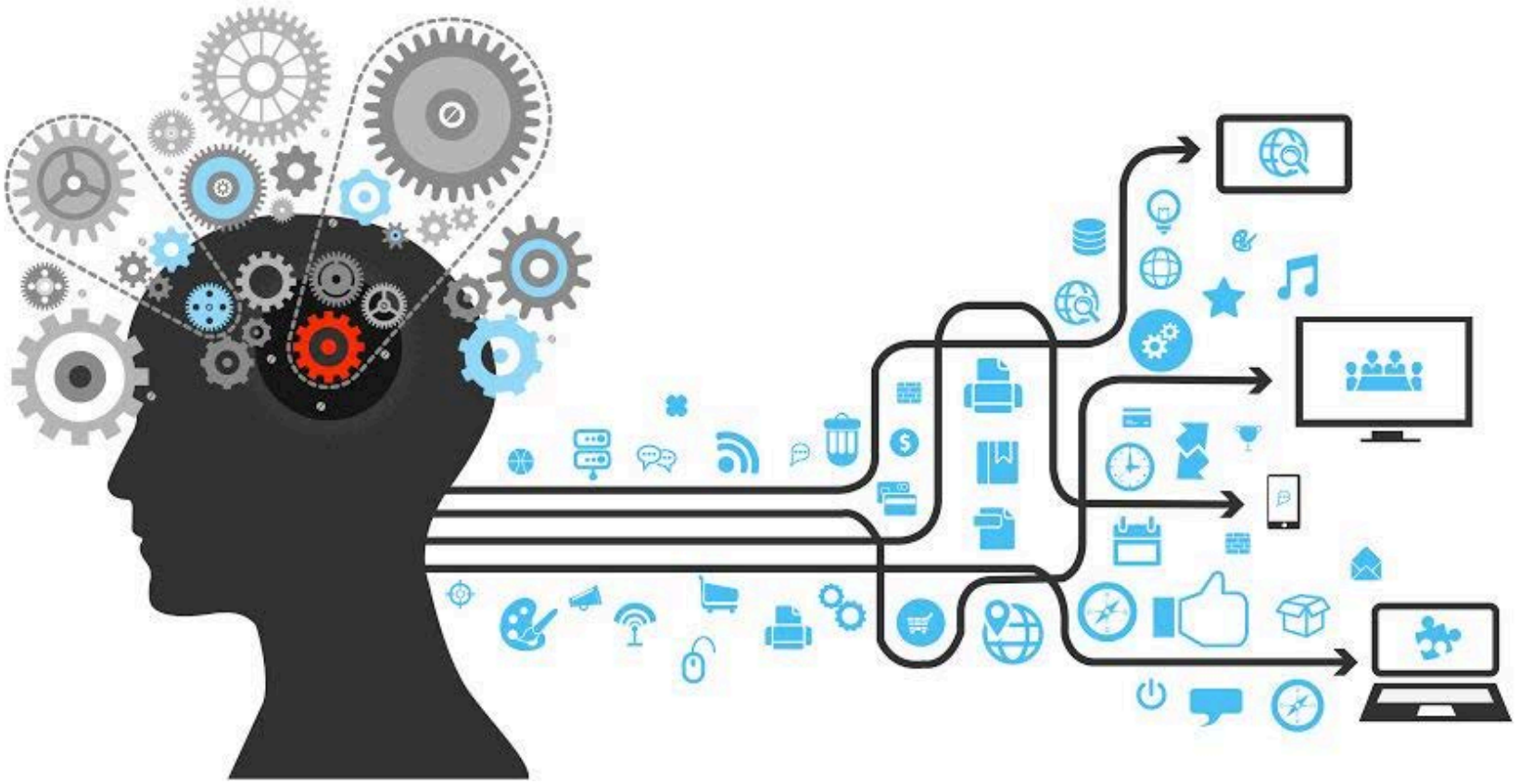
	3250	3251	3252	3253
I. Business Problem (Question) Framing	✓	✓✓	✓	✓✓✓
II. Analytics Problem Framing	✓	✓✓✓	✓	✓✓✓
III. Data	✓✓	✓✓✓	✓✓	✓✓✓
IV. Methodology (Approach) Selection	✓	✓✓		✓✓✓
V. Model Building		✓✓✓	✓	✓✓✓
VI. Deployment			✓✓✓	✓
VII. Model Life Cycle Management			✓✓	✓✓✓

✓ = Introductory content
✓ ✓ = Substantial coverage
✓ ✓ ✓ = Major focus

Module 1 – Section 1

What is Machine Learning?

Machine Learning



Machine Learning

- [Machine Learning is the] field of study that gives computers the ability to learn without being explicitly programmed.

  Arthur Samuel, 1959

- A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience

  E.Tom Mitchell, 1997

Module 1 – Section 2

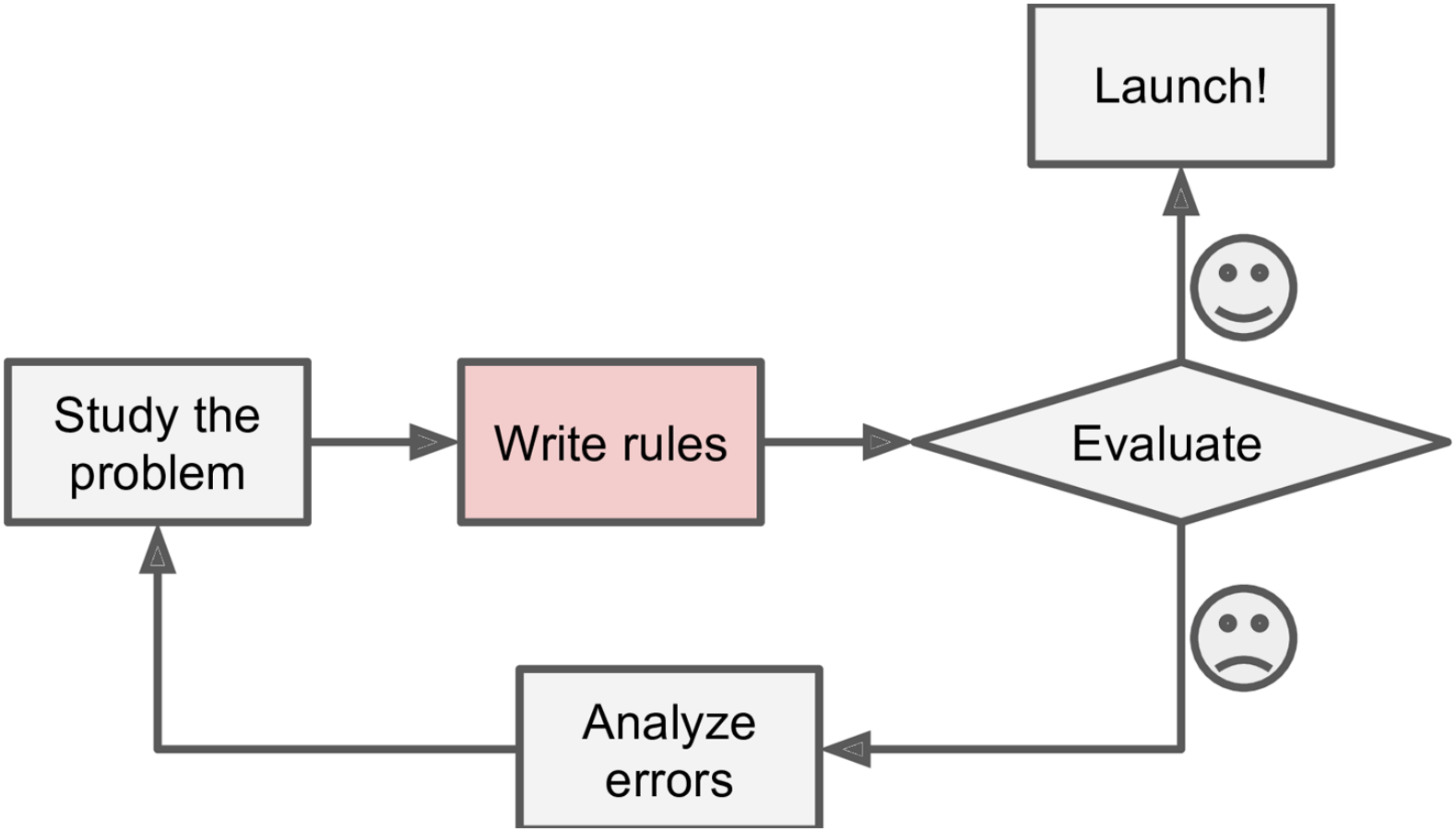
Why Machine Learning?

Why Machine Learning

*IDC Digital Universe report, 2014 http://www.emc.com/infographics/digital-universe-2014.htm

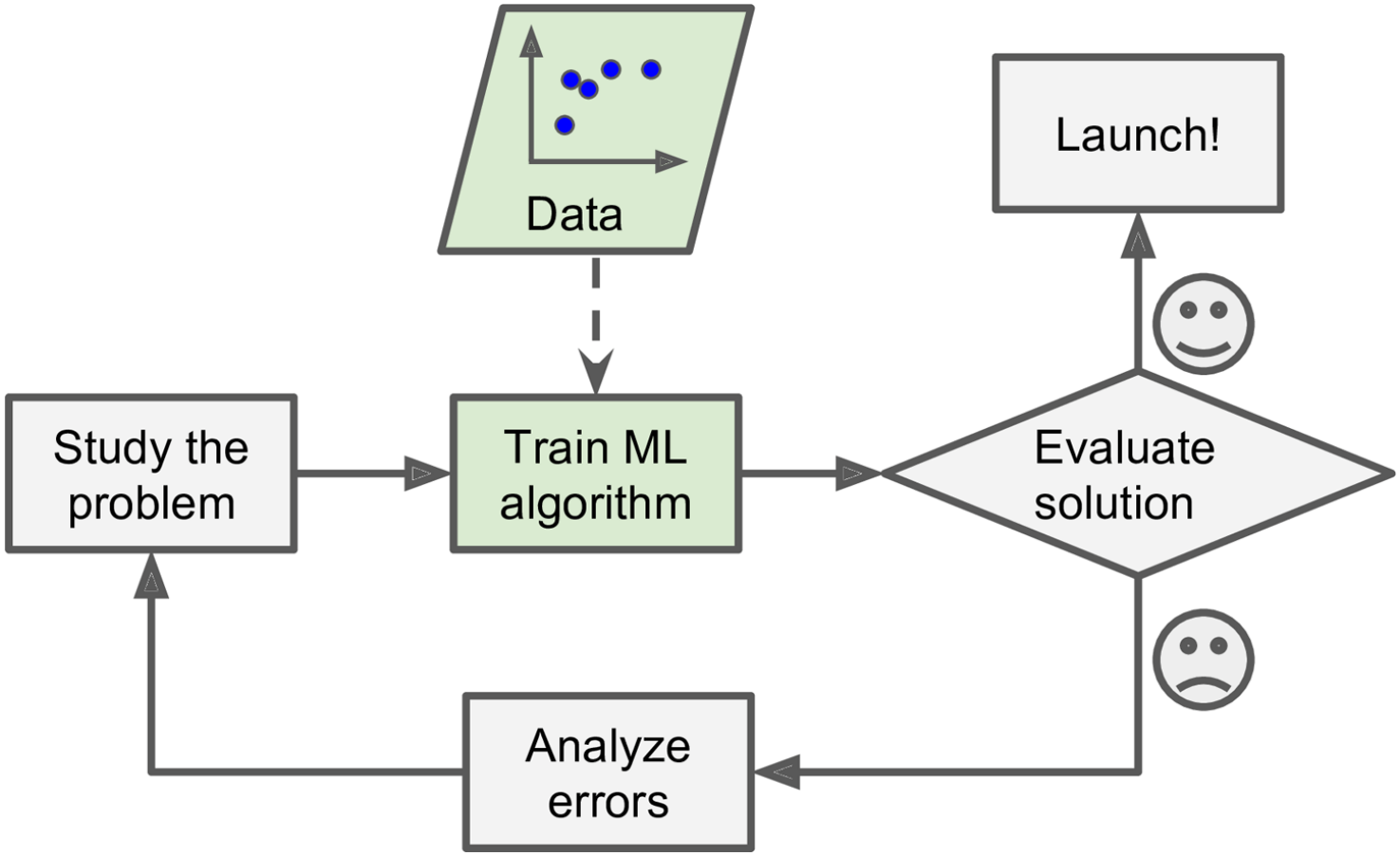
**Data Scientist: The Sexiest Job of the 21st Century, Oct 2012 https://hbr.org/2012/10/data-scientist-the-sexiest-job-of-the-21st-century

Traditional Approach



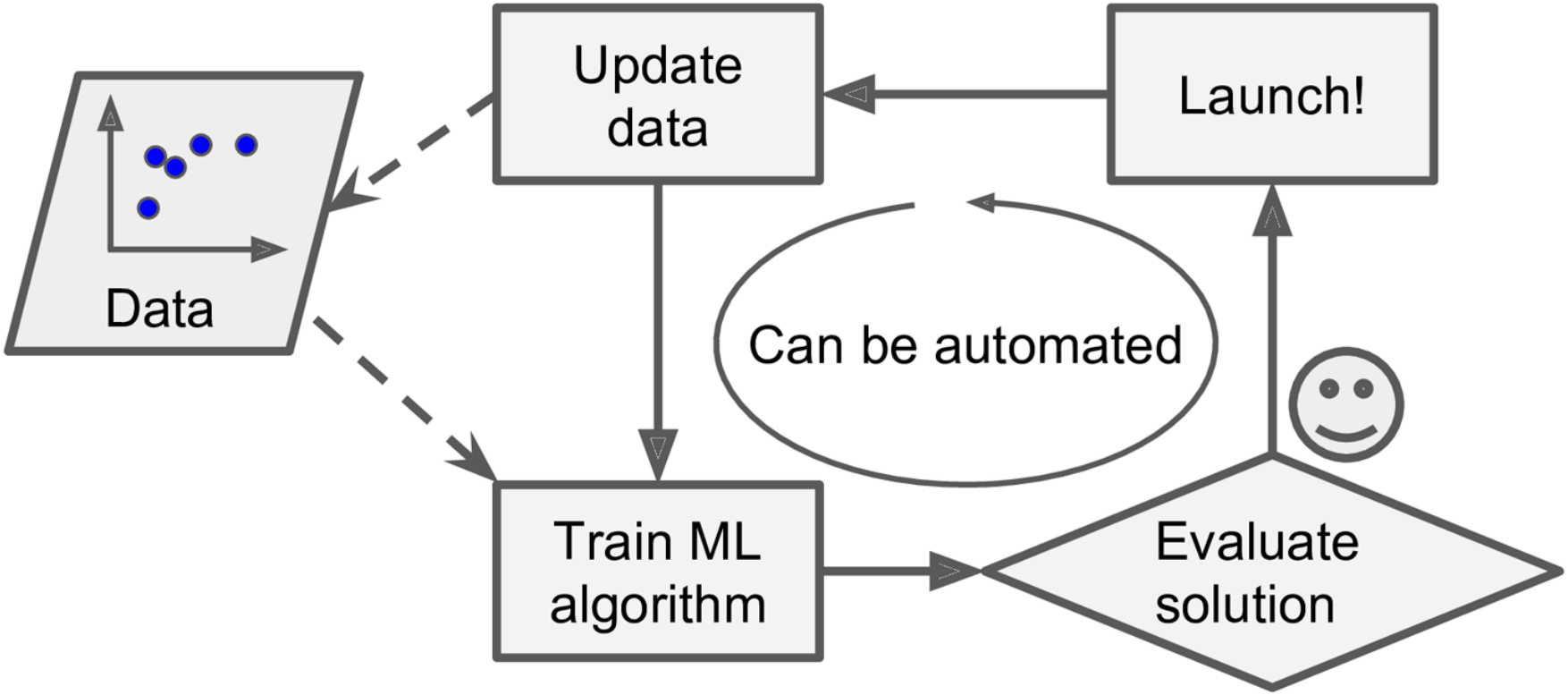
Long list of rules that can break!

Machine Learning Approach



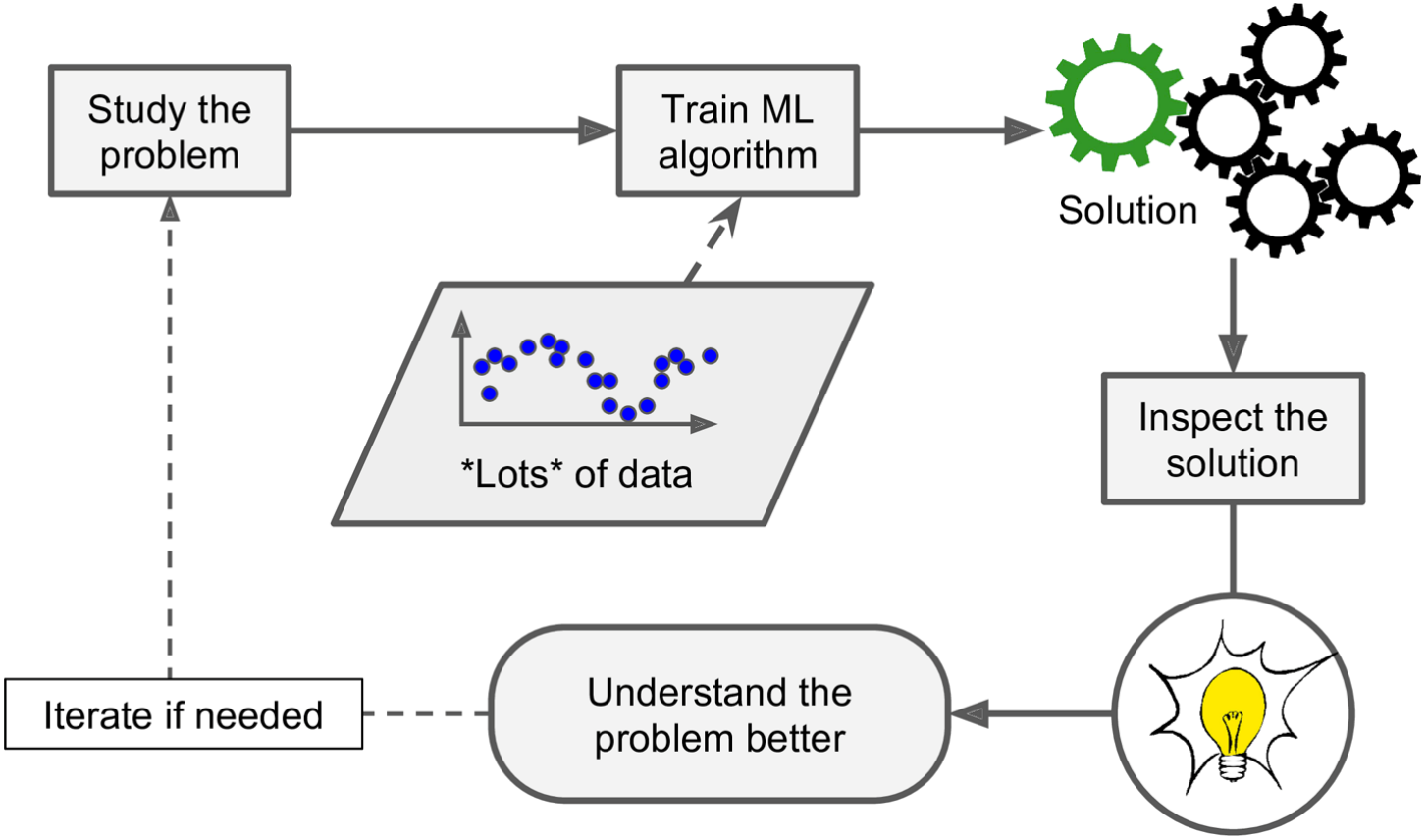
Learn from data. Generalize!

Machine Learning Automation



Update model as new data arrives

Machine Learning Automation (cont'd)



Find non-obvious patterns.


ELON MUSK: HUMANS MUST MERGE WITH MACHINES
OR BECOME IRRELEVANT IN AI AGE

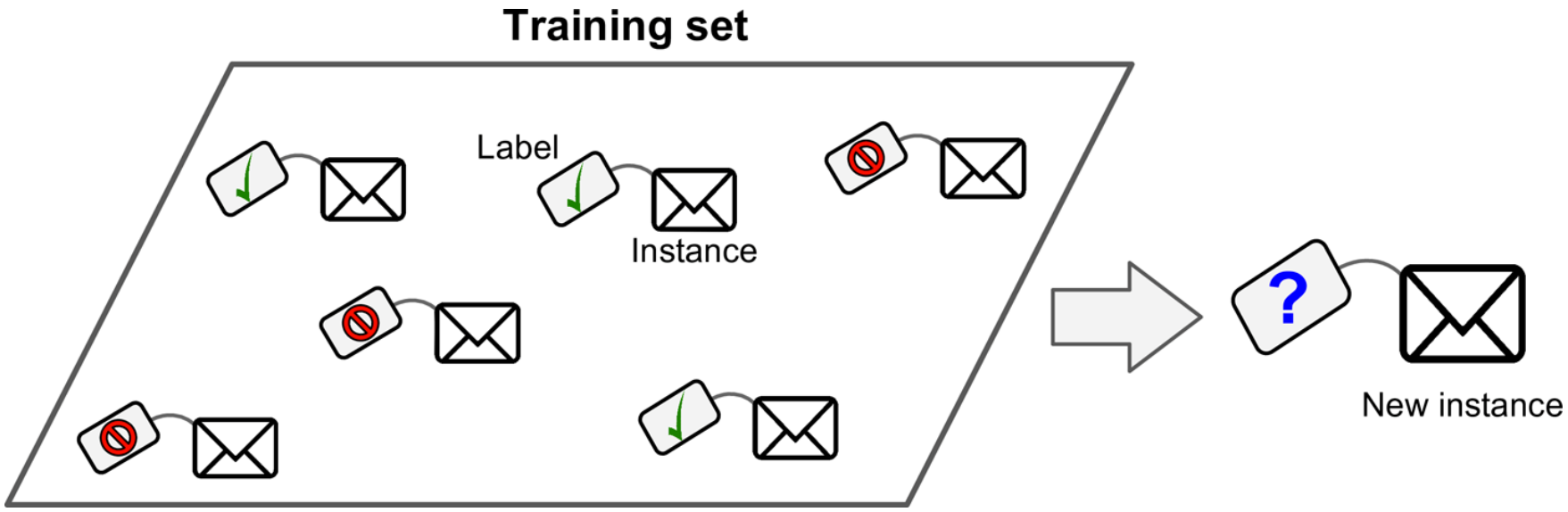
Module 1 – Section 3

Types of Machine Learning

Types of Machine Learning

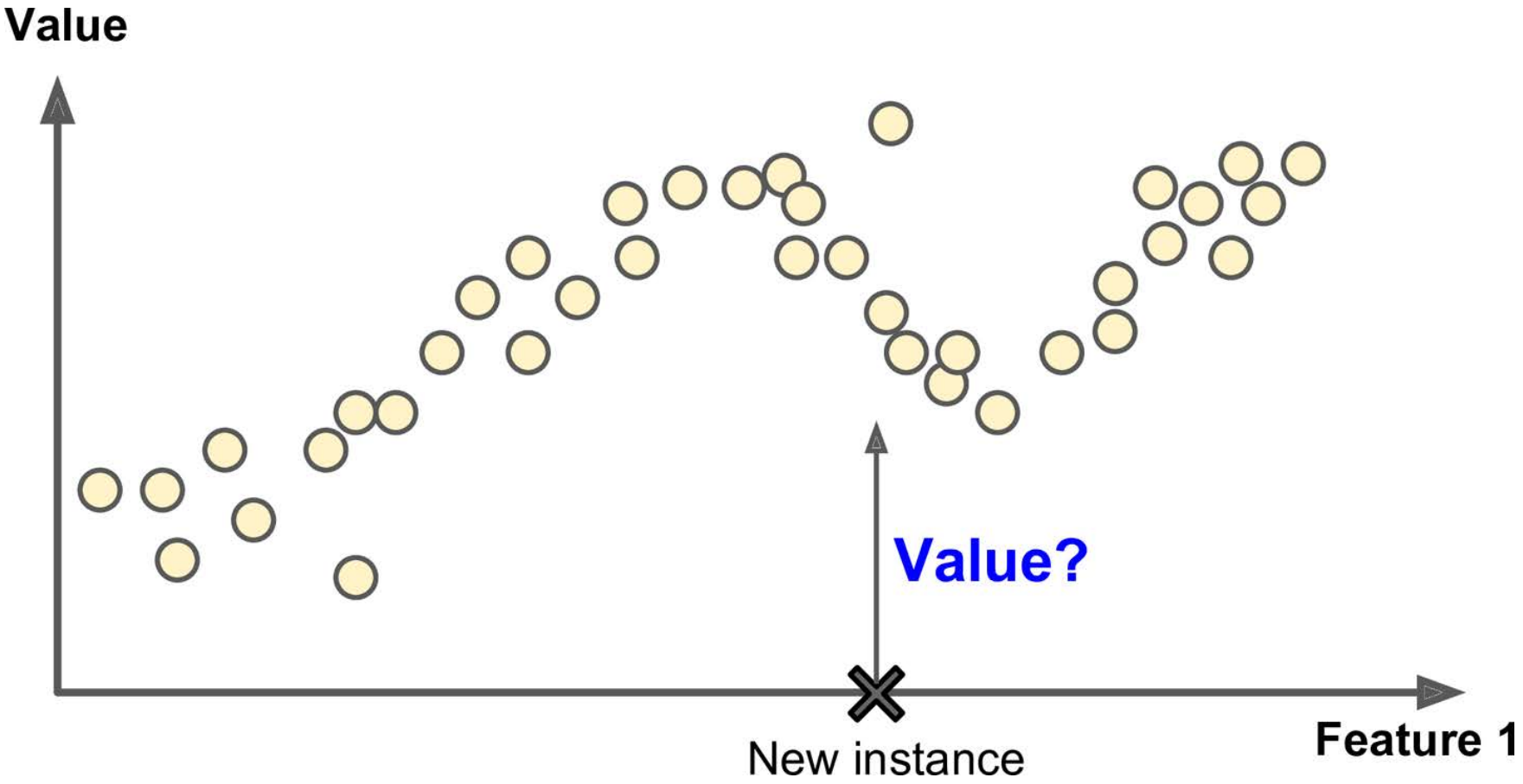
- Whether or not they are trained with human supervision (Supervised, Unsupervised, Semi-supervised, and Reinforcement Learning)
- Whether or not they can learn incrementally on the fly (online versus batch learning)
- Whether they work by simply comparing new data points to known data points, or instead detect patterns in the training data and build a predictive model (instance-based versus model-based learning)

Supervised Learning Classification

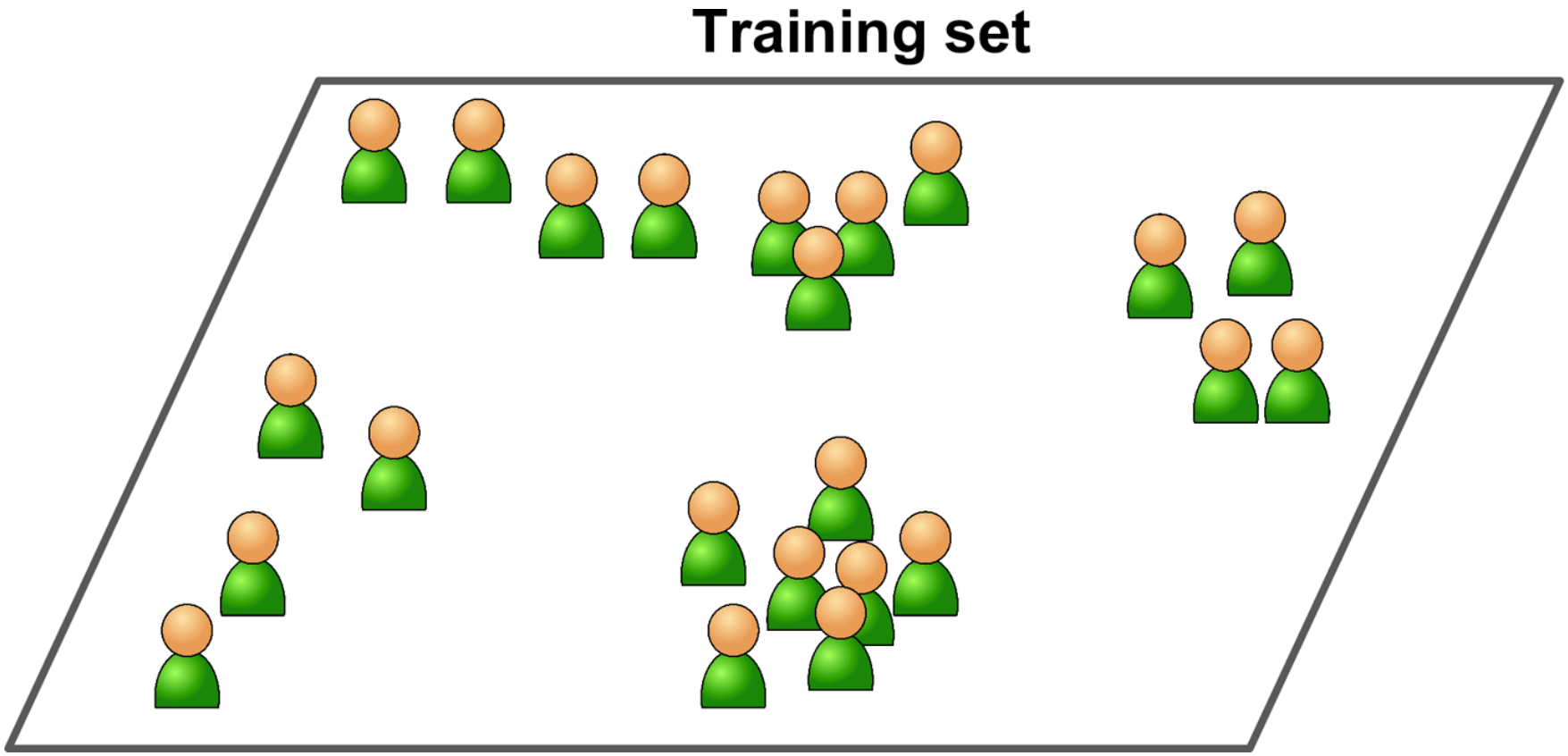


The training data you feed to the algorithm includes the desired solutions, called labels

Supervised Learning Regression

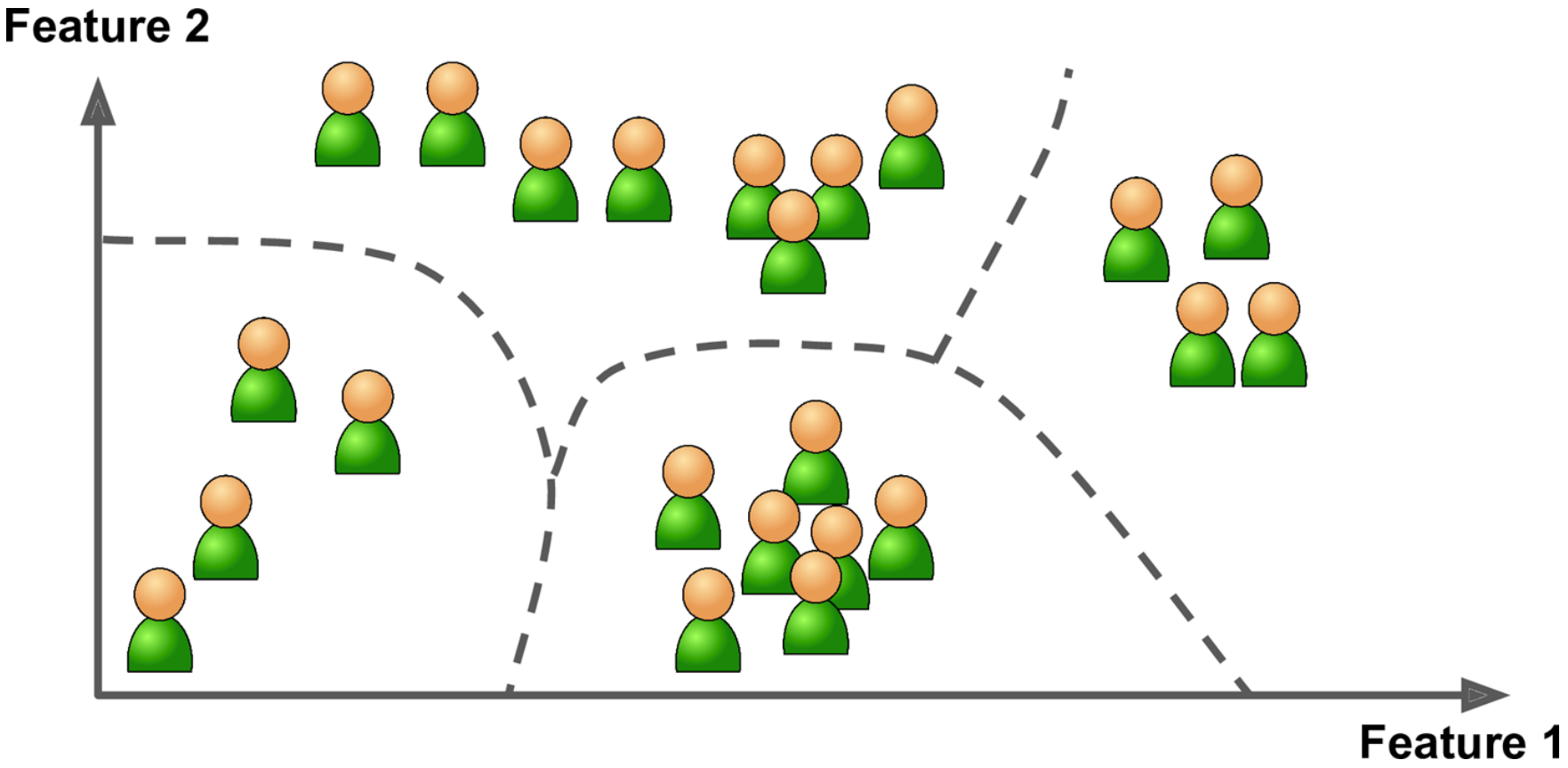


Unsupervised Learning



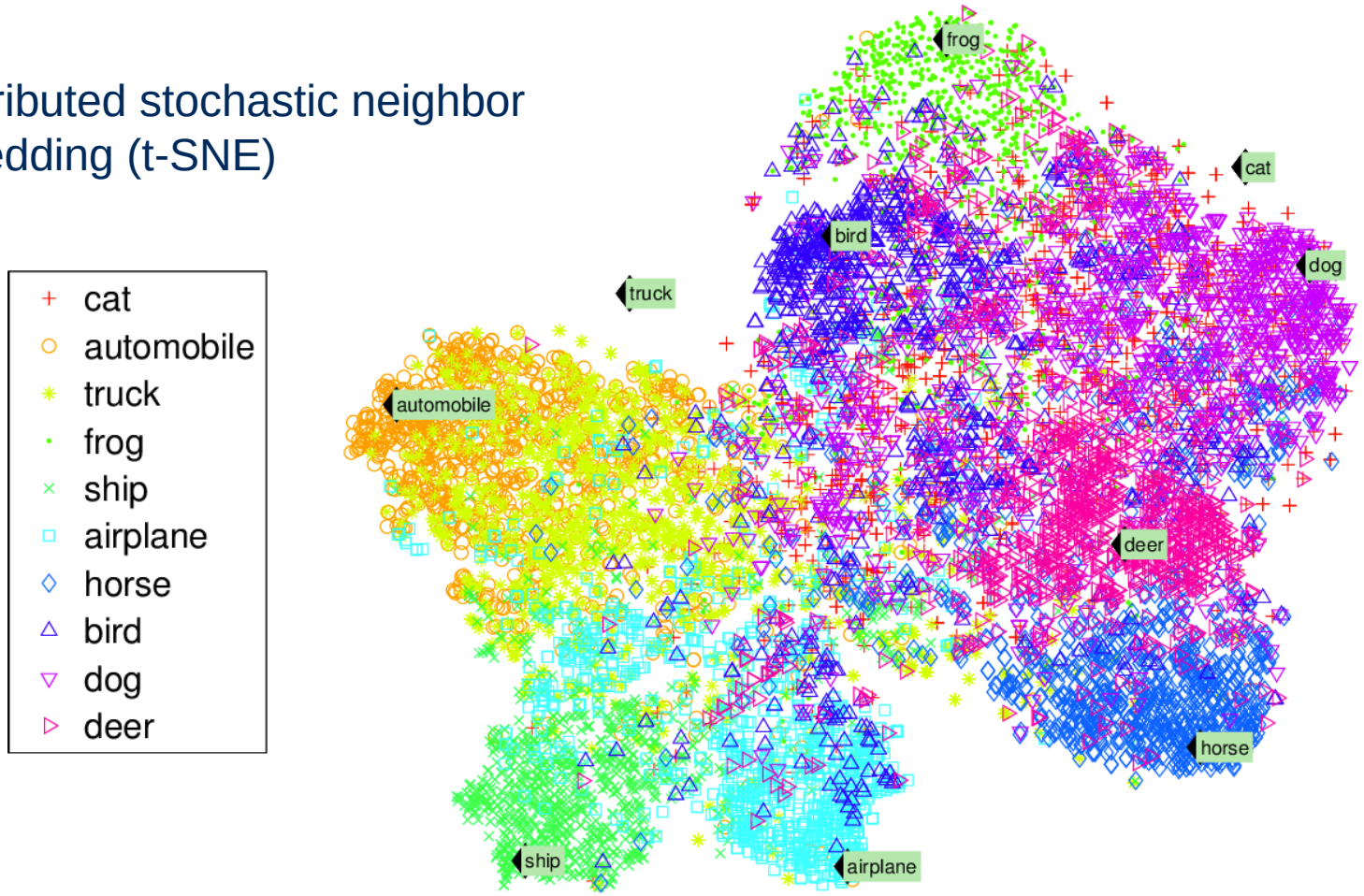
Data is unlabeled

Unsupervised Learning (cont'd)



Unsupervised Learning (cont'd)

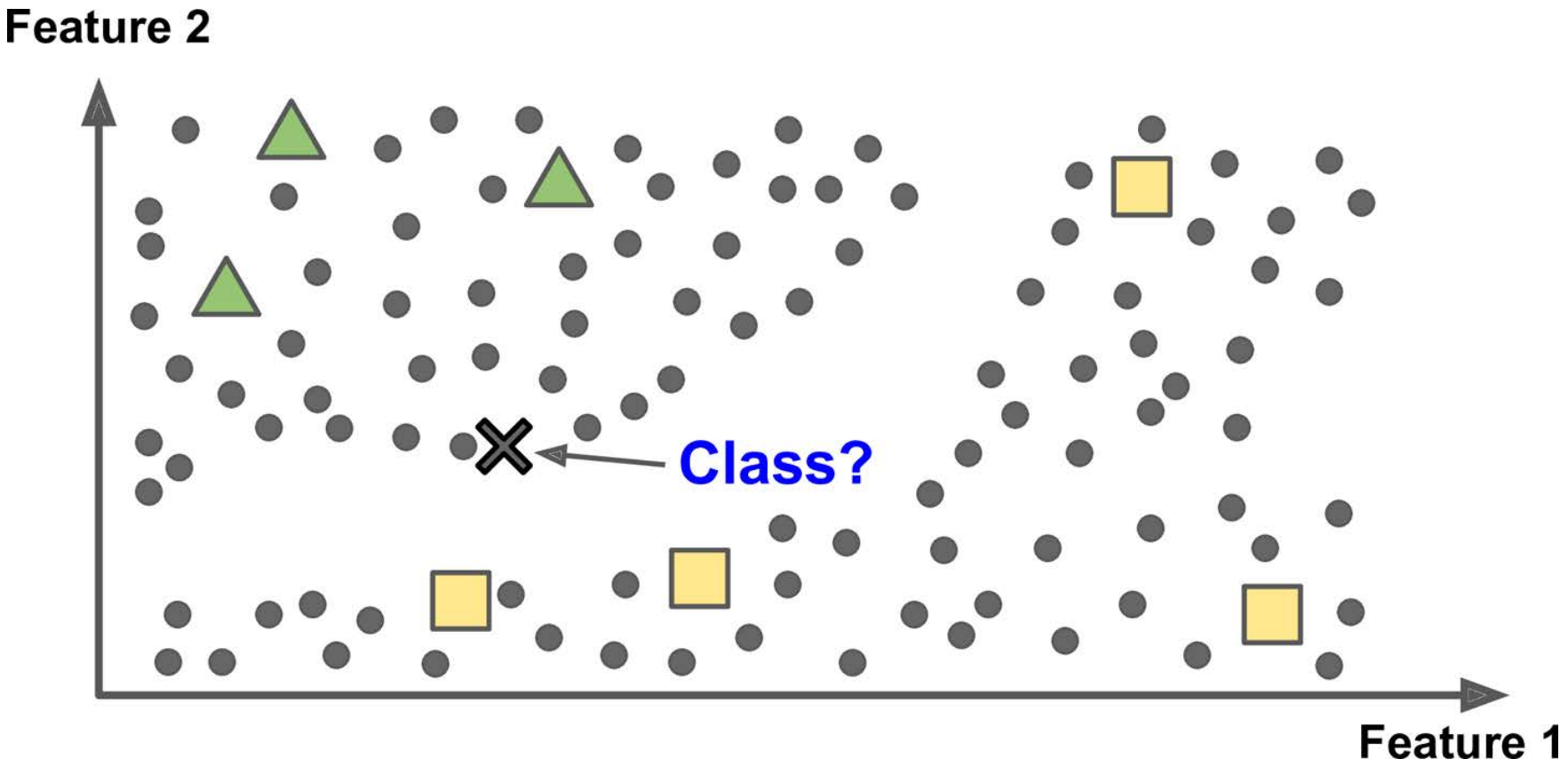
t-distributed stochastic neighbor embedding (t-SNE)



Simplify the data without losing information

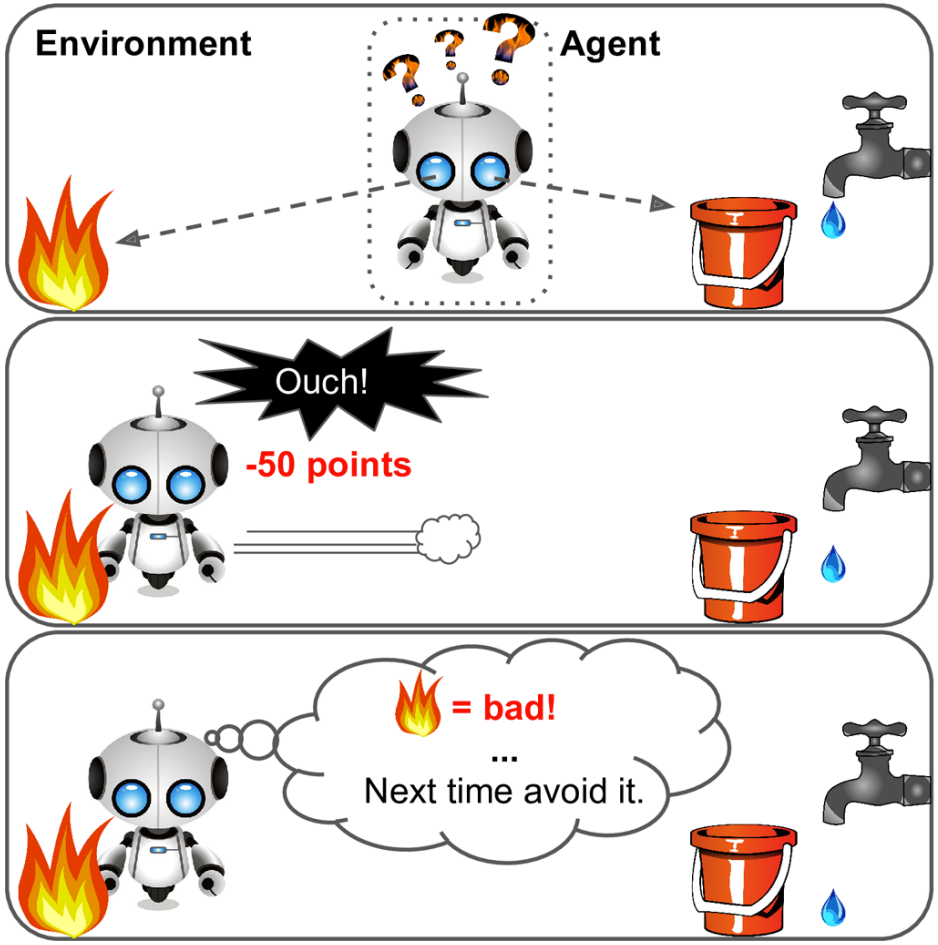
Unsupervised Learning Anomaly Detection

Semi-supervised Learning



Partially labeled training data: usually a lot of unlabeled data and a little bit of labeled data.

Reinforcement Learning

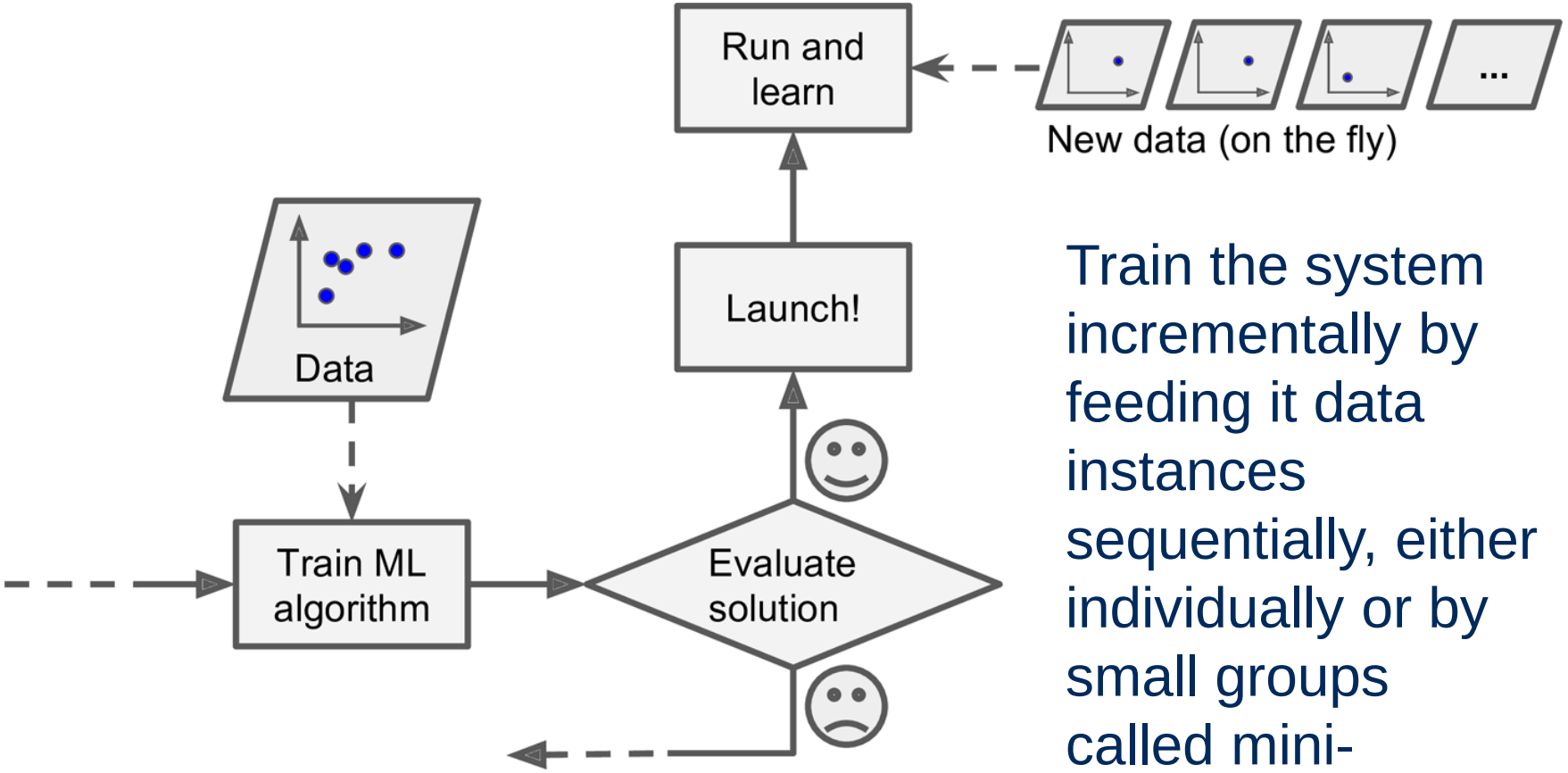


1. Observe
2. Select action using policy
3. Action!
4. Get reward or penalty
5. Update policy (learning step)
6. Iterate until an optimal policy is found

Batch Learning

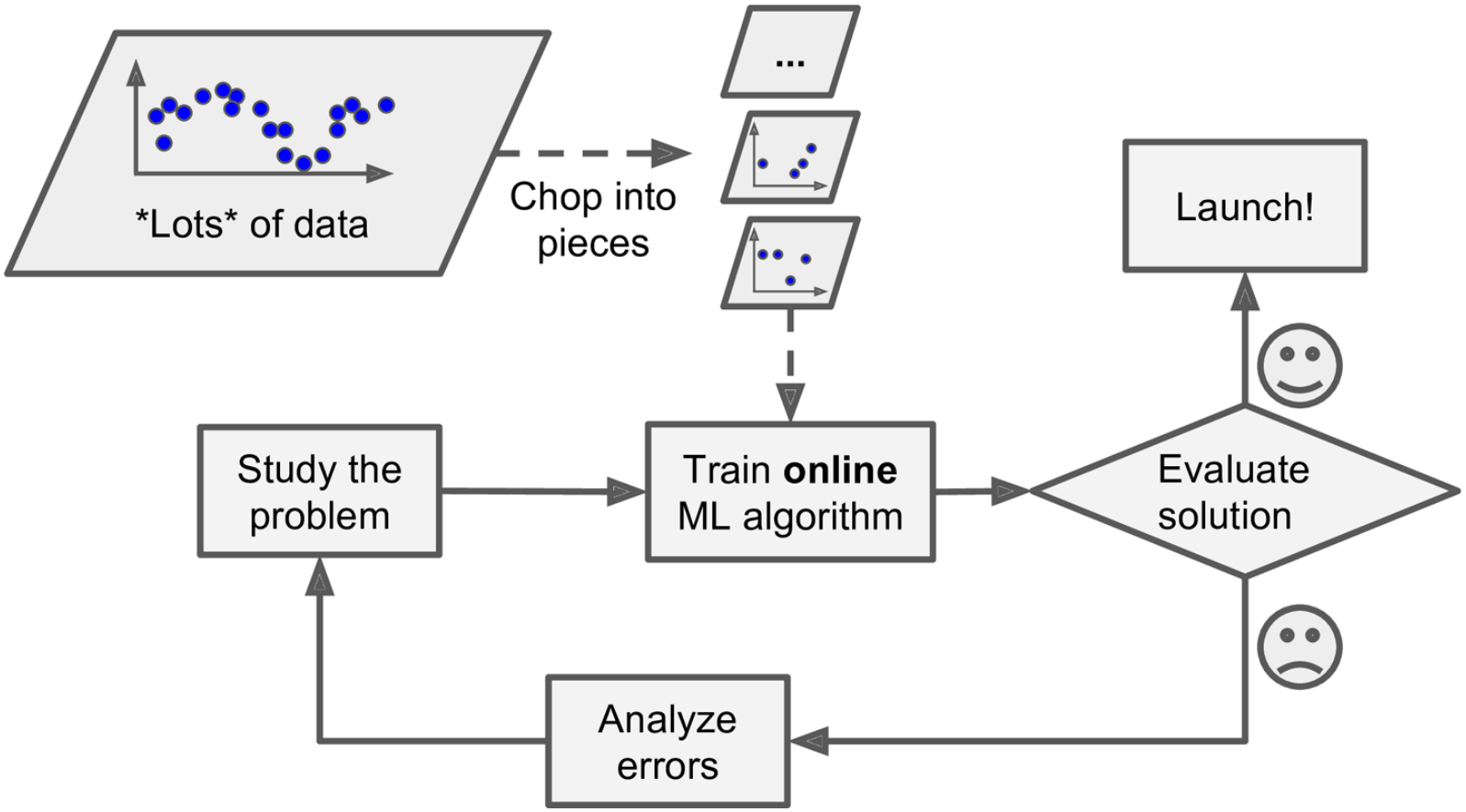
- Train model using all the available data. This will generally take a lot of time and computing resources, so it is typically done offline
- Train algorithm from scratch using new + old data
- Can be automated

Online Learning

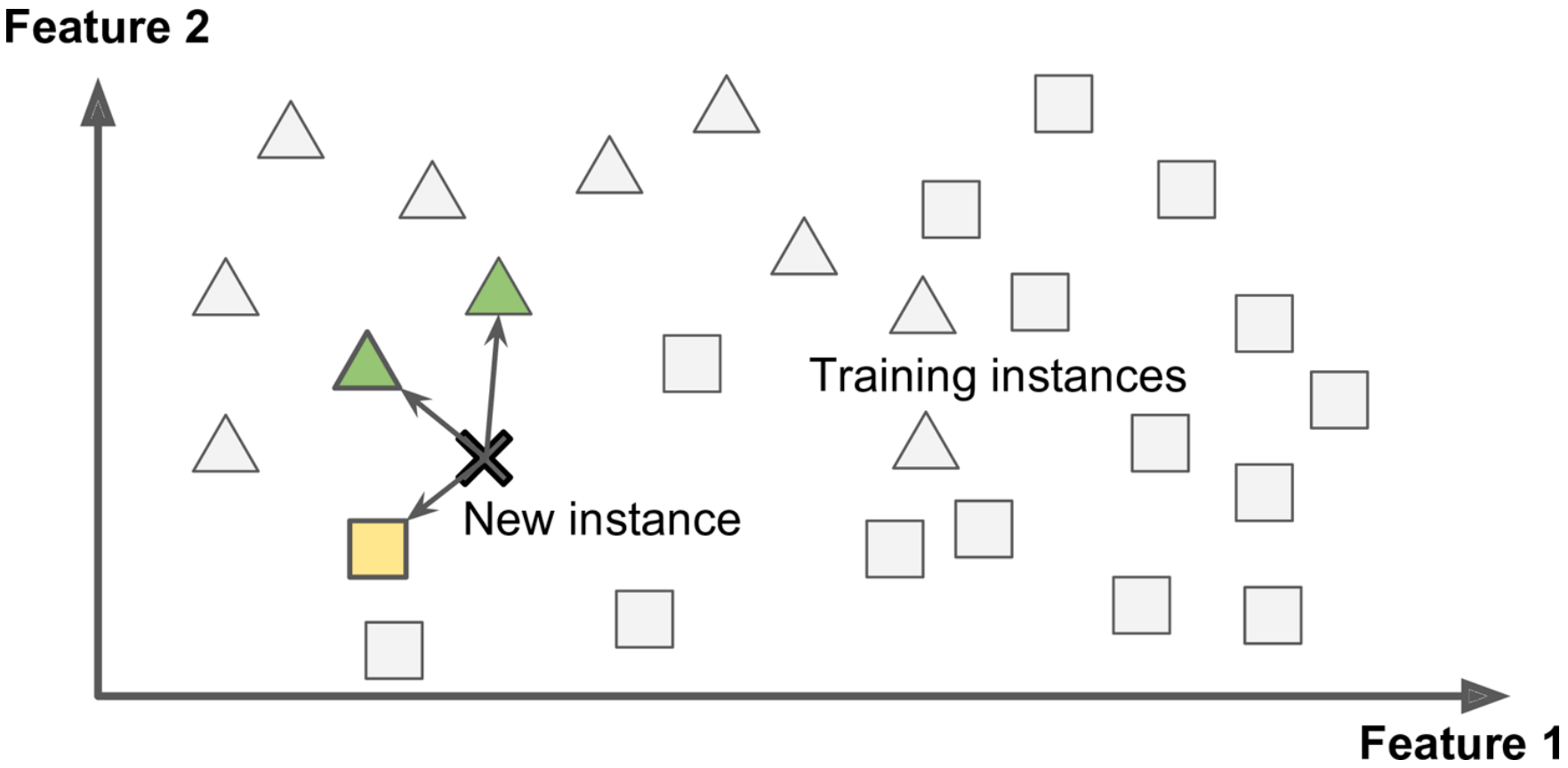


Train the system incrementally by feeding it data instances sequentially, either individually or by small groups called mini-batches.

Online Learning for Large Datasets

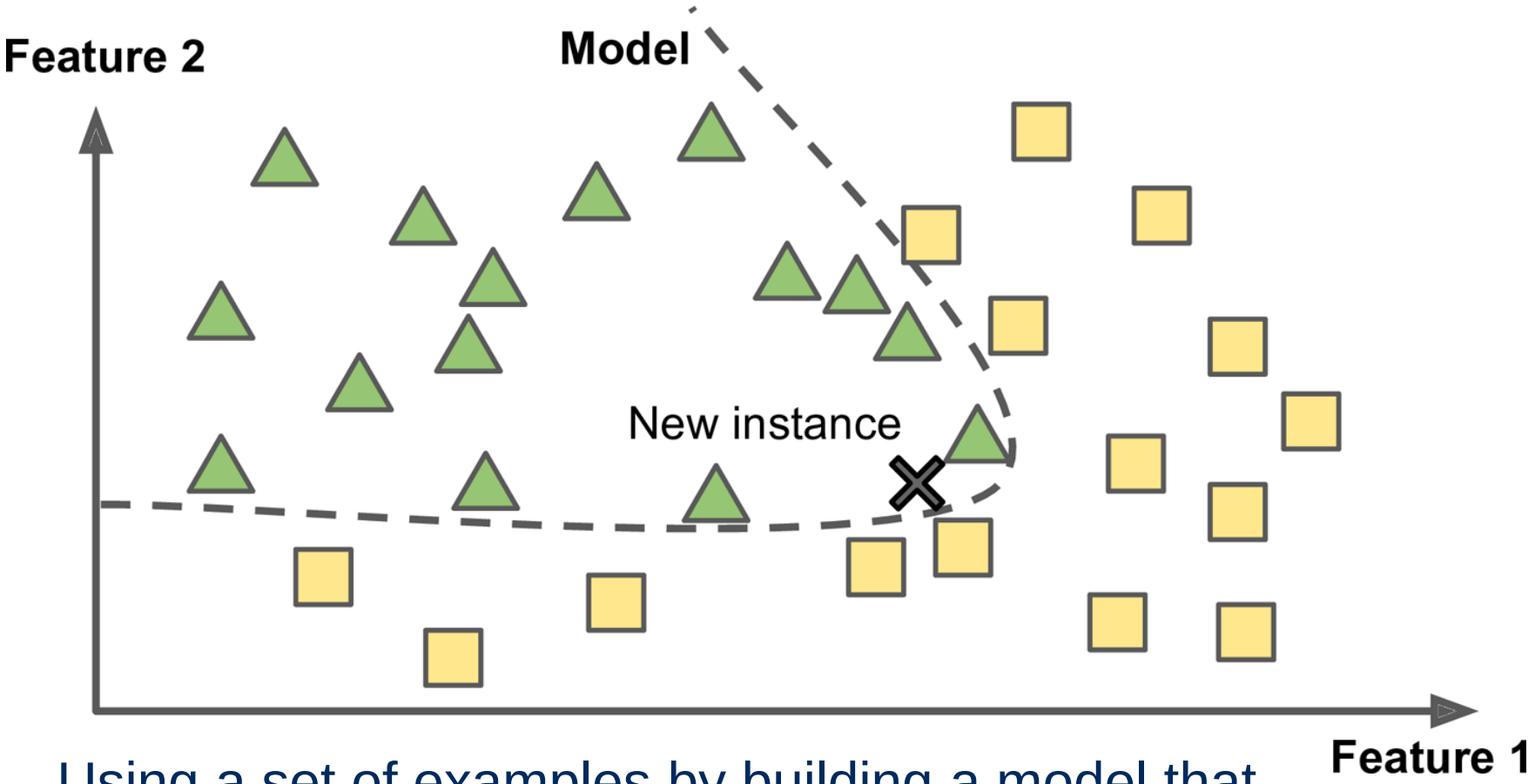


Instance Based Learning



This is called instance-based learning: the system learns the examples by heart, then generalizes to new cases using a similarity measure

Model Based Learning

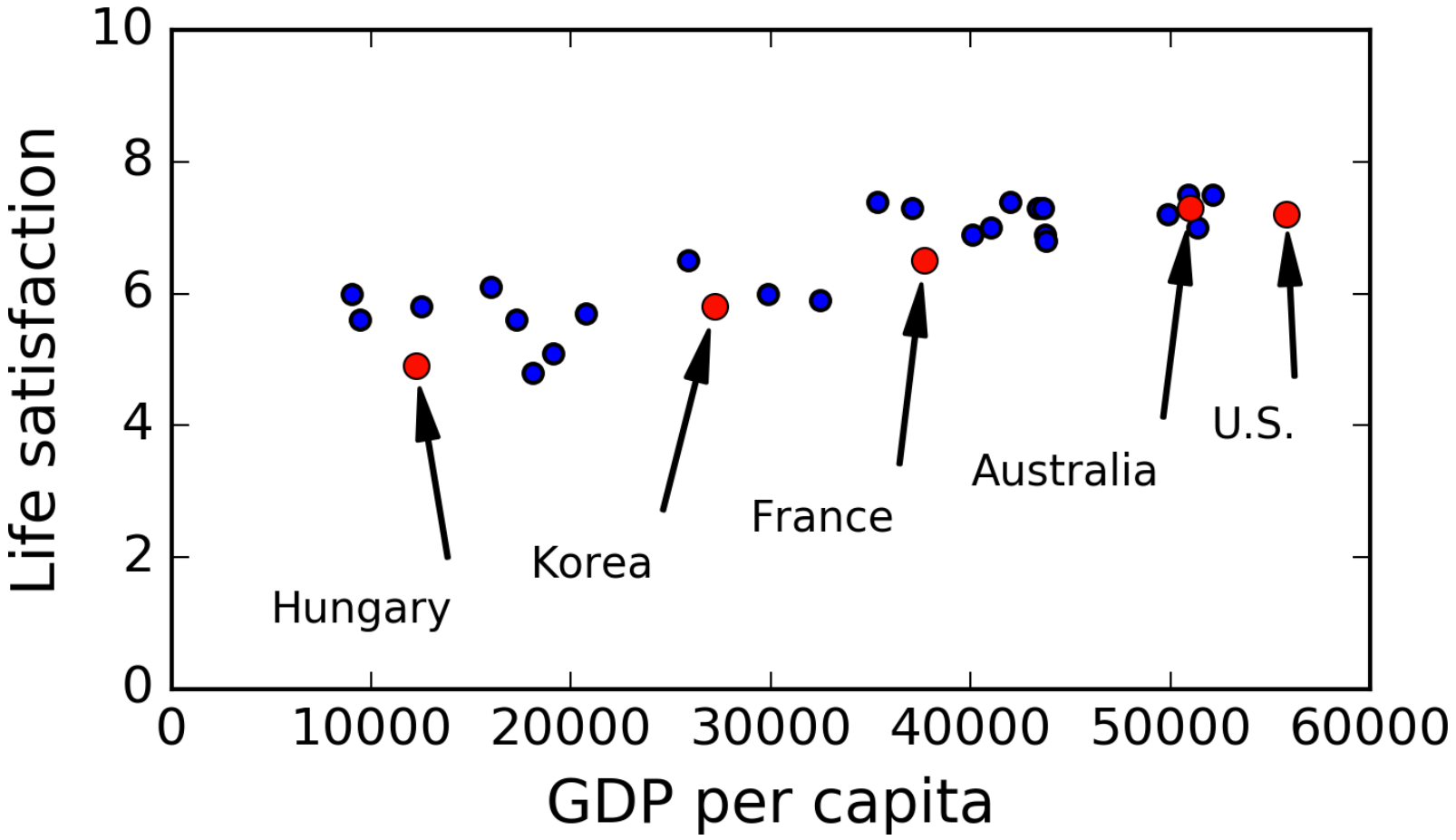


Using a set of examples by building a model that generalizes these examples, then use that model to make predictions.

Module 1 – Section 4

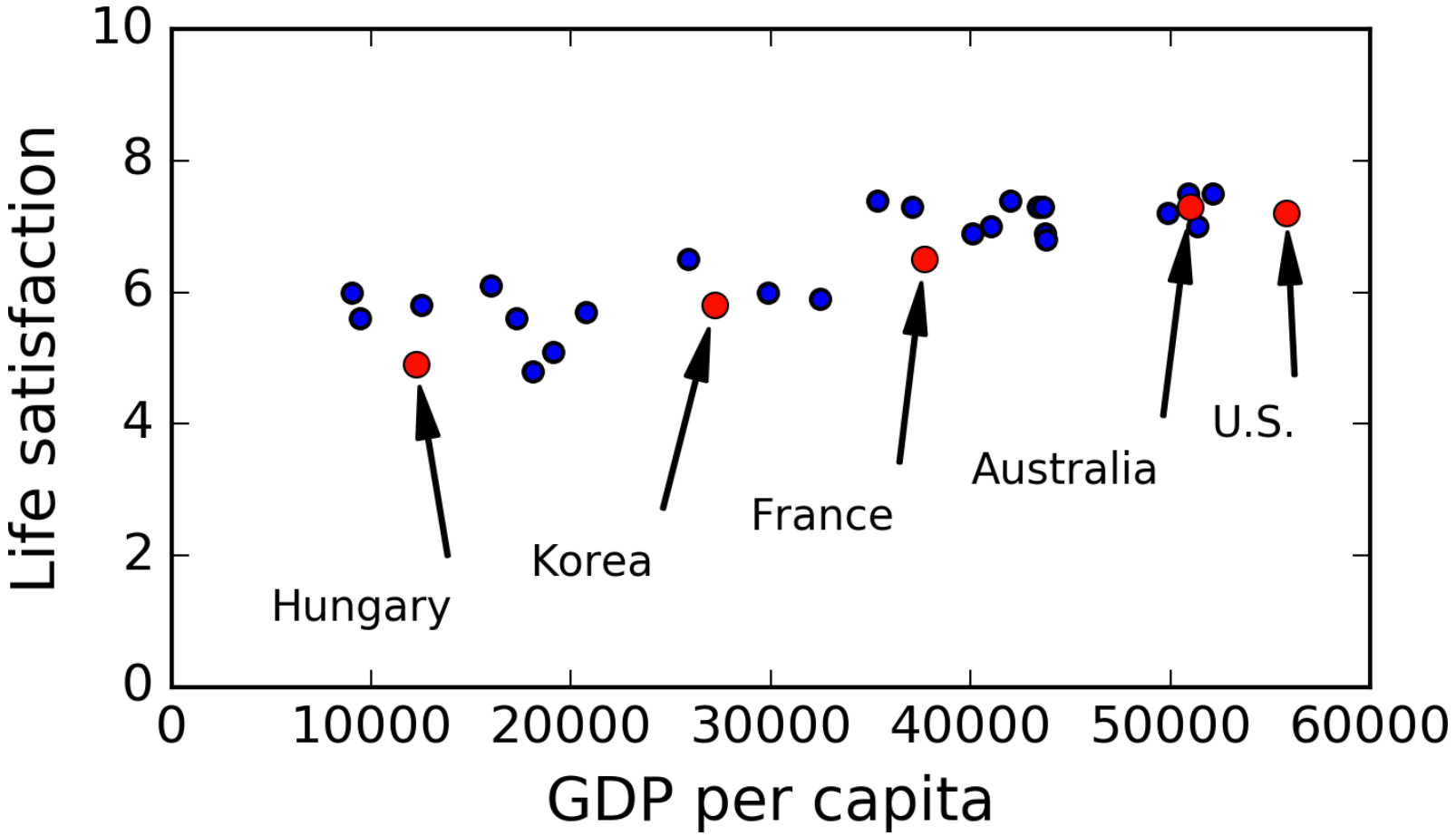
Modeling

Is There a Trend?

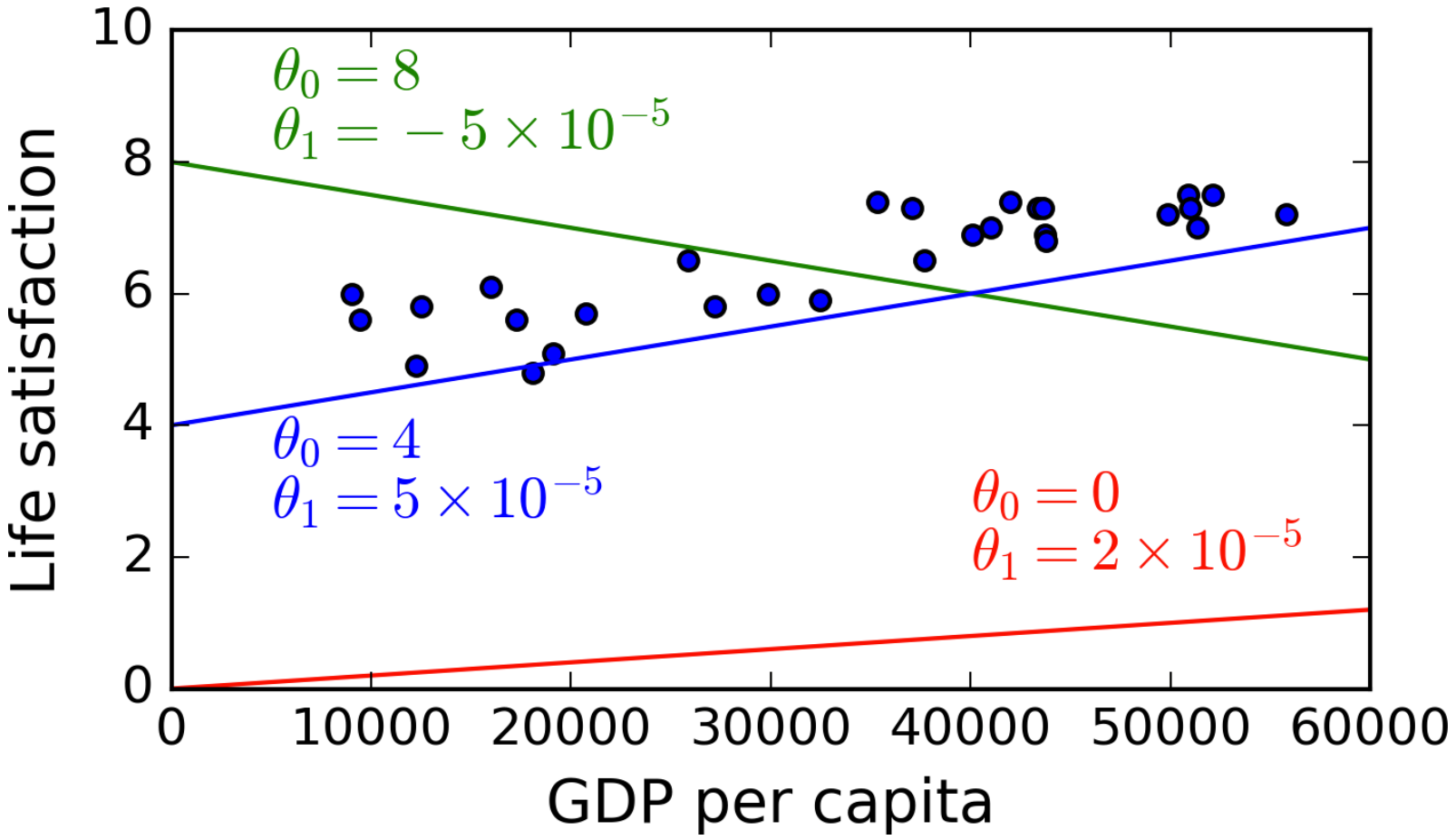


Build Model

$$life\_satisfaction = \theta_0 + \theta_1 \times GDP\_per\_capita$$



Possible Models

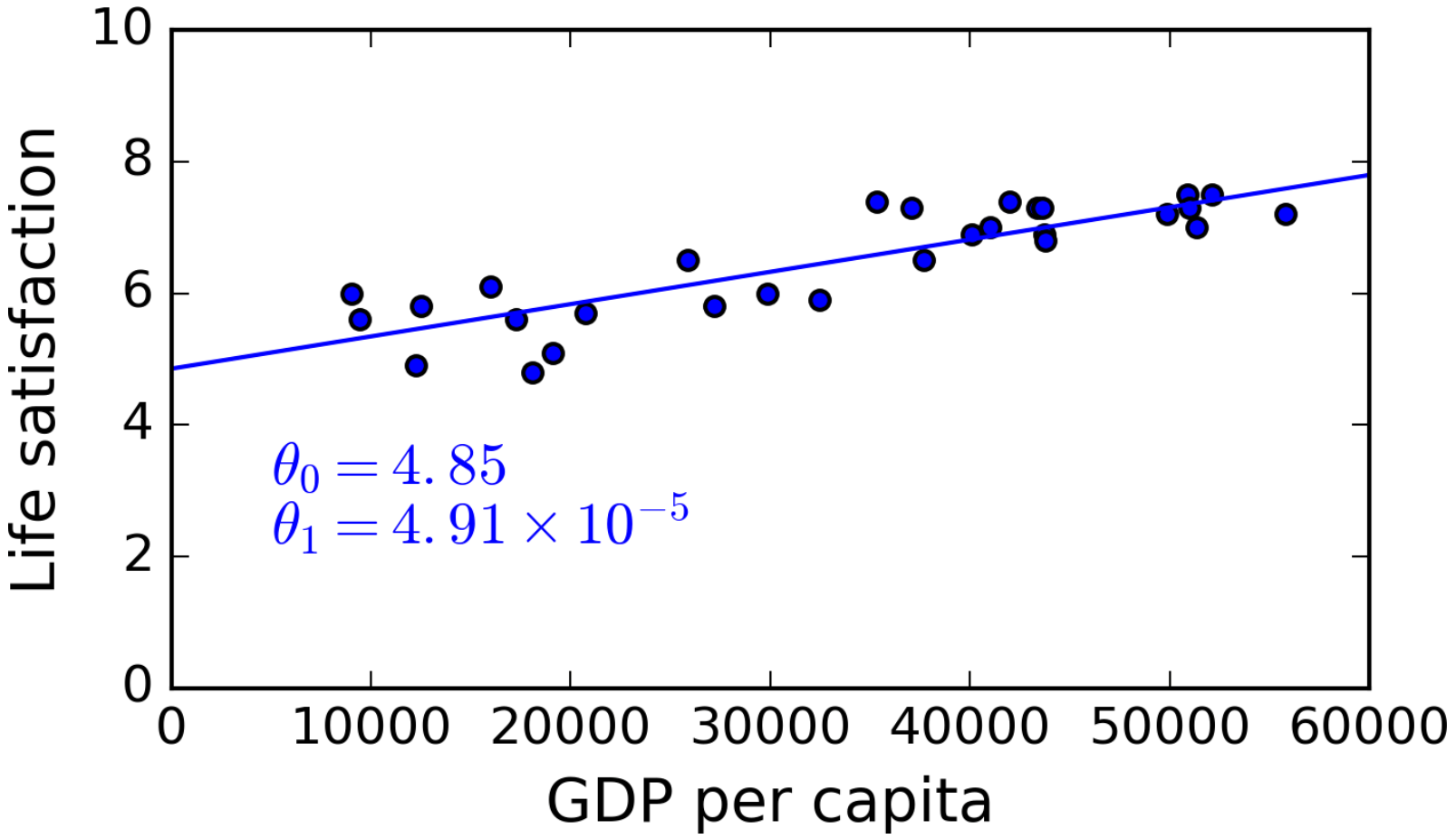


Build Model (cont'd)

$$life\_satisfaction = \theta_0 + \theta_1 \times GDP\_per\_capita$$

- How to define parameters?
- Specify a performance measure or a cost function.
  - i.e. Measure distance between examples and model's prediction
- Find optimal parameters that minimize the cost function

Best Model


Life satisfaction
GDP per capita
$\theta_0 = 4.85$
$\theta_1 = 4.91 \times 10^{-5}$
0
2
4
6
8
10
0
10000
20000
30000
40000
50000
60000

Coding

```
import matplotlib
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import sklearn

# Load the data
oecd_bli = pd.read_csv("oecd_bli_2015.csv", thousands=',')
gdp_per_capita = pd.read_csv("gdp_per_capita.csv",thousands=',',del
                             encoding='latin1', na_values="n/a")

# Prepare the data
country_stats = prepare_country_stats(oecd_bli, gdp_per_capita)
X = np.c_[country_stats["GDP per capita"]]
y = np.c_[country_stats["Life satisfaction"]]

# Visualize the data
country_stats.plot(kind='scatter', x="GDP per capita", y='Life sati
plt.show()

# Select a linear model
lin_reg_model = sklearn.linear_model.LinearRegression()

# Train the model
lin_reg_model.fit(X, y)

# Make a prediction for Cyprus
X_new = [[22587]]  # Cyprus' GDP per capita
print(lin_reg_model.predict(X_new)) # outputs [[ 5.96242338]]
```

ML Process Summary

- You studied the data.
- You selected a model.
- You trained it on the training data (i.e., the learning algorithm searched for the model parameter values that minimize a cost function).
- Finally, you applied the model to make predictions on new cases (this is called inference), hoping that this model will generalize well.

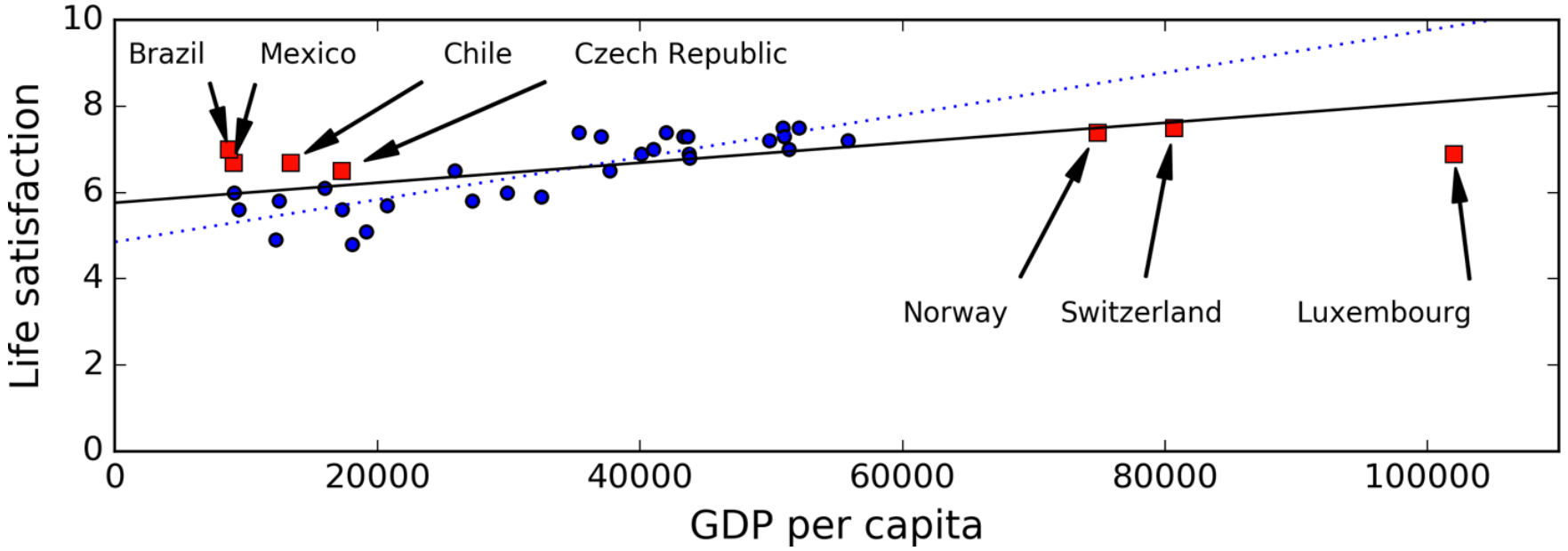
Module 1 – Section 5

Challenges of Machine Learning

Common ML Challenges

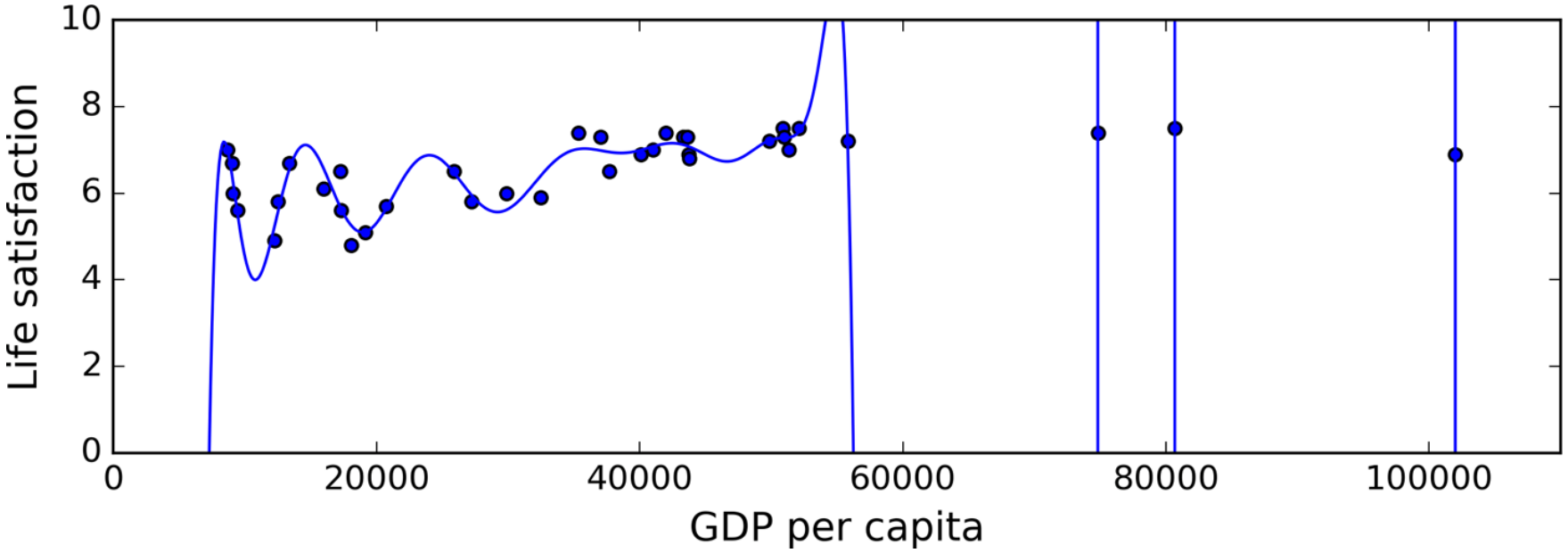
- Insufficient Quantity of Data
- Nonrepresentative Training Data
  - New data has different structure
- Poor-Quality Data
  - Missing data, outliers
- Irrelevant Features
  - Features selection, extraction
- Overfitting or Underfitting

Representative Data



In order to generalize well, it is crucial that your training data be representative of the new cases you want to generalize to. This is true whether you use instance-based learning or model-based learning.

Data Overfitting

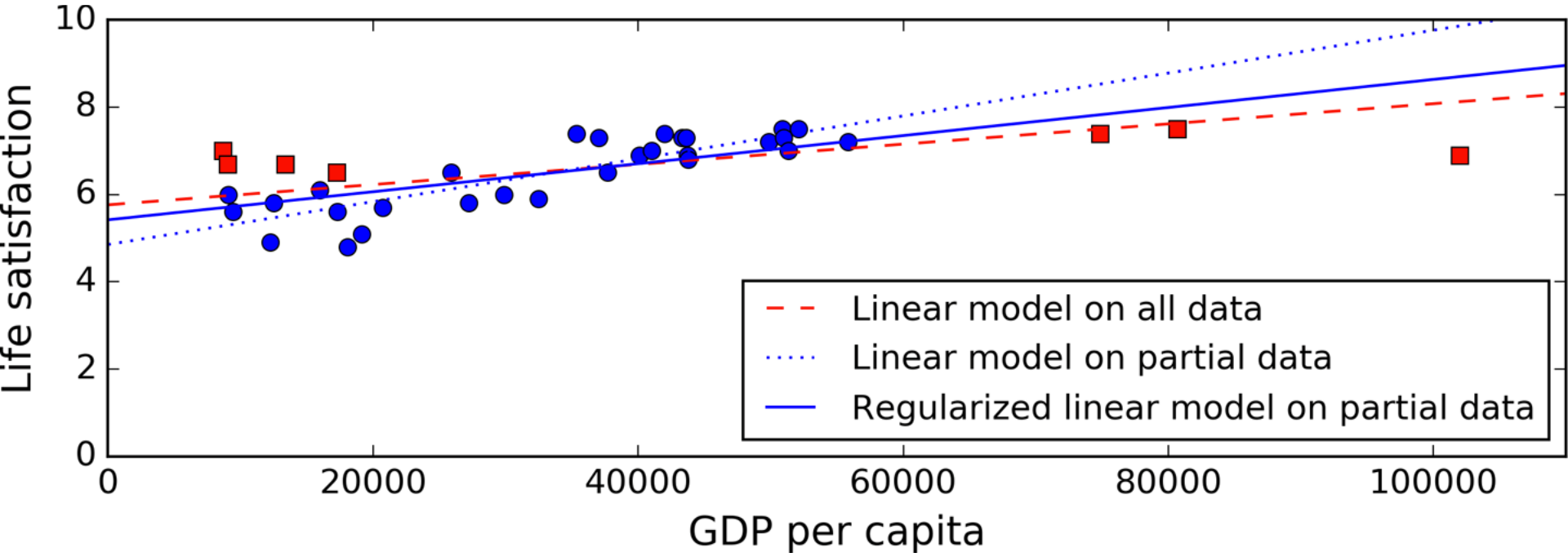


Overfitting means that the model performs well on the training data, but it does not generalize well.

Solutions:

- Simplify model
- Gather more data
- Reduce noise (fix data)

Regularization Data



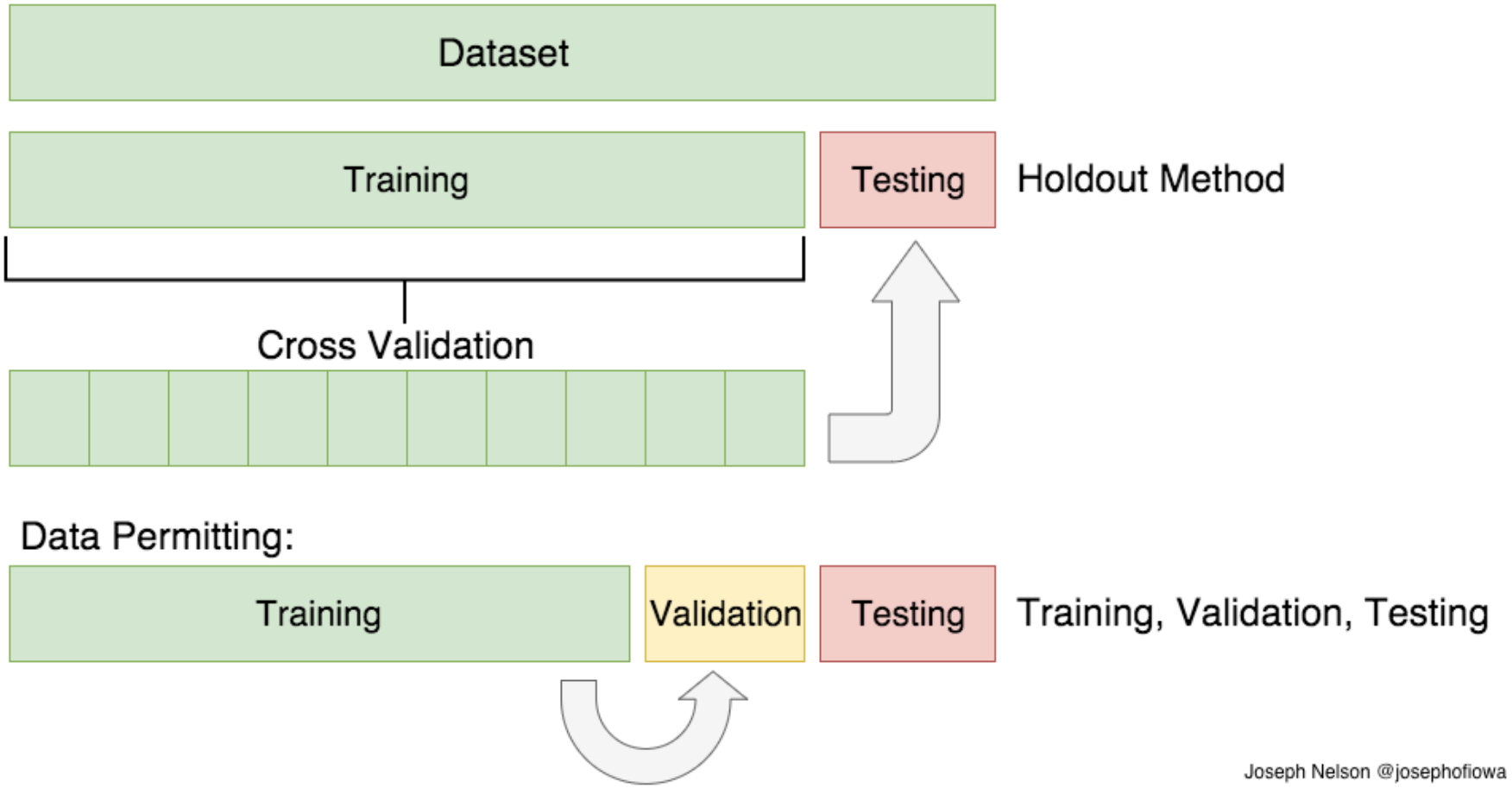
So Far …

- Machine Learning is about making machines get better at some task by learning from data, instead of having to explicitly code rules.
- There are many different types of ML systems: supervised or not, batch or online, instance-based or model-based, and so on.
- In a ML project you gather data in a training set, and you feed the training set to a learning algorithm.

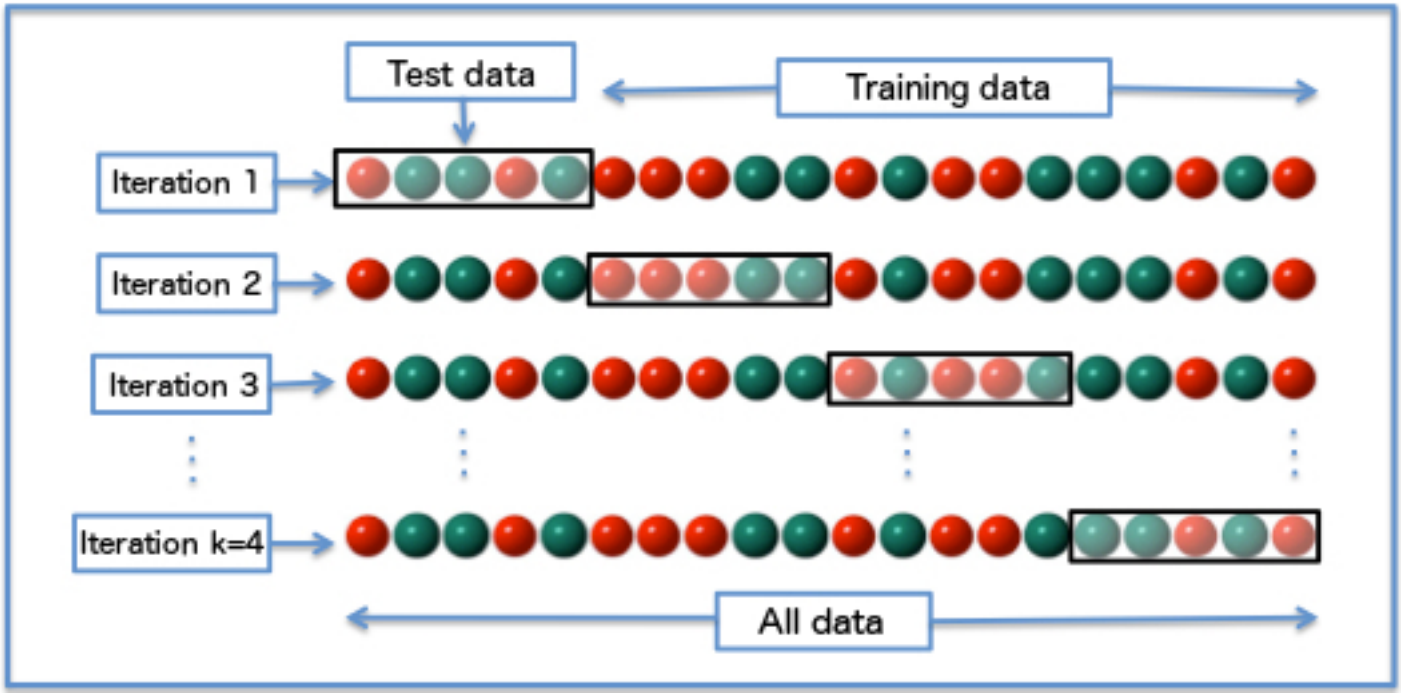
So Far … (cont'd)

- If the algorithm is model-based it tunes some parameters to fit the model to the training set (i.e., to make good predictions on the training set itself), and then hopefully it will be able to make good predictions on new cases as well.
- If the algorithm is instance-based, it just learns the examples by heart and uses a similarity measure to generalize to new instances.
- The system will not perform well if your training set is too small, or if the data is not representative, noisy, or polluted with irrelevant features (garbage in, garbage out).
- Lastly, your model needs to be neither too simple (in which case it will underfit) nor too complex (in which case it will overfit).

Testing and Validation



Cross-Validation

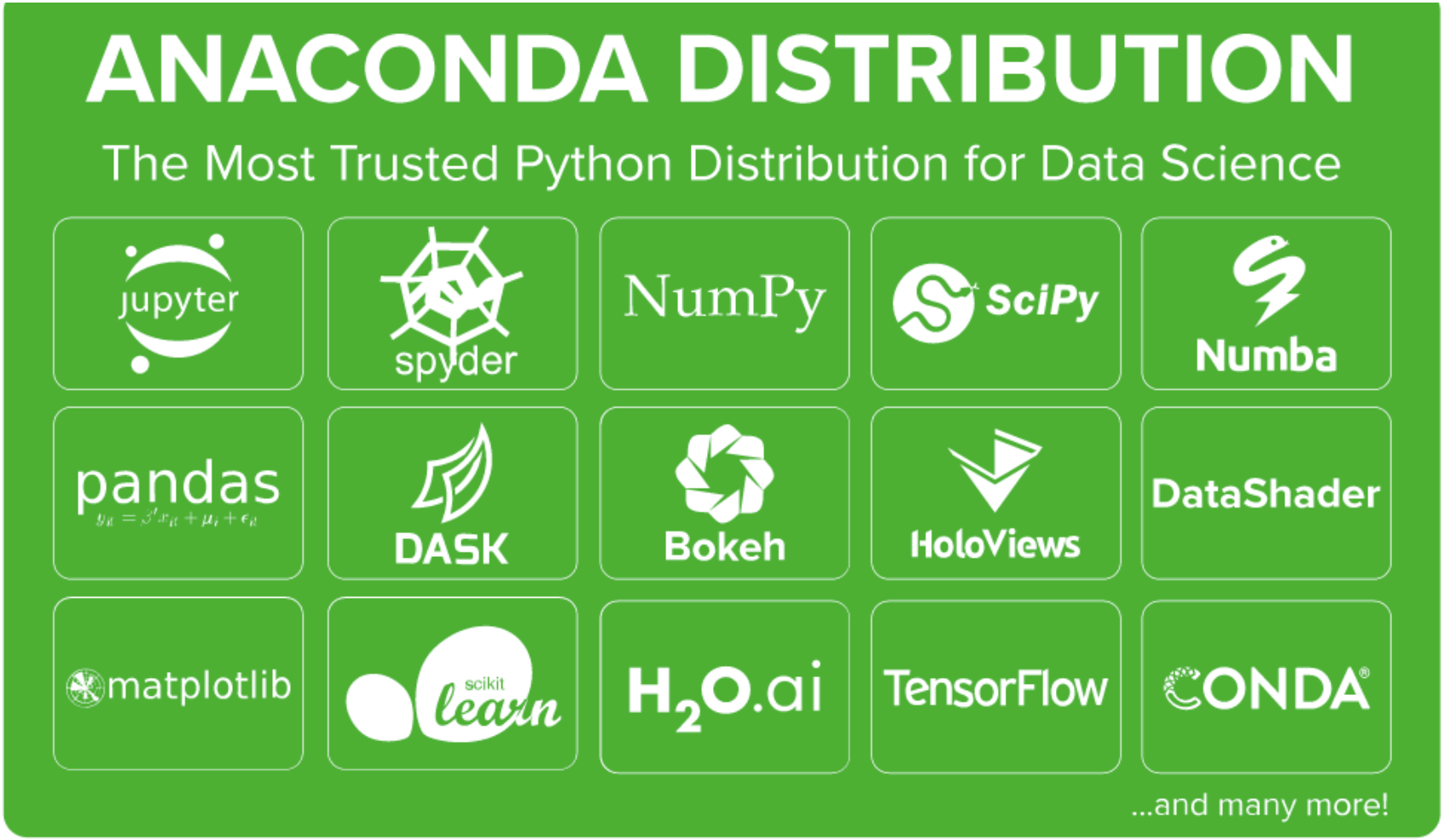


UNIVERSITY OF TORONTO
SCHOOL OF CONTINUING STUDIES

Module 1 – Section 6

Tools & Techniques

Anaconda


ANACONDA DISTRIBUTION
The Most Trusted Python Distribution for Data Science
jupyter
spyder
NumPy
SciPy
Numba
pandas
DASK
Bokeh
HoloViews
DataShader
matplotlib
scikit learn
H2O.ai
TensorFlow
CONDA
...and many more!

sklearn

- Simple and efficient tools for data mining and data analysis
- Accessible to everybody, and reusable in various contexts
- Built on NumPy, SciPy, and matplotlib
- Open source, commercially usable - BSD licence
- http://scikit-learn.org/stable/

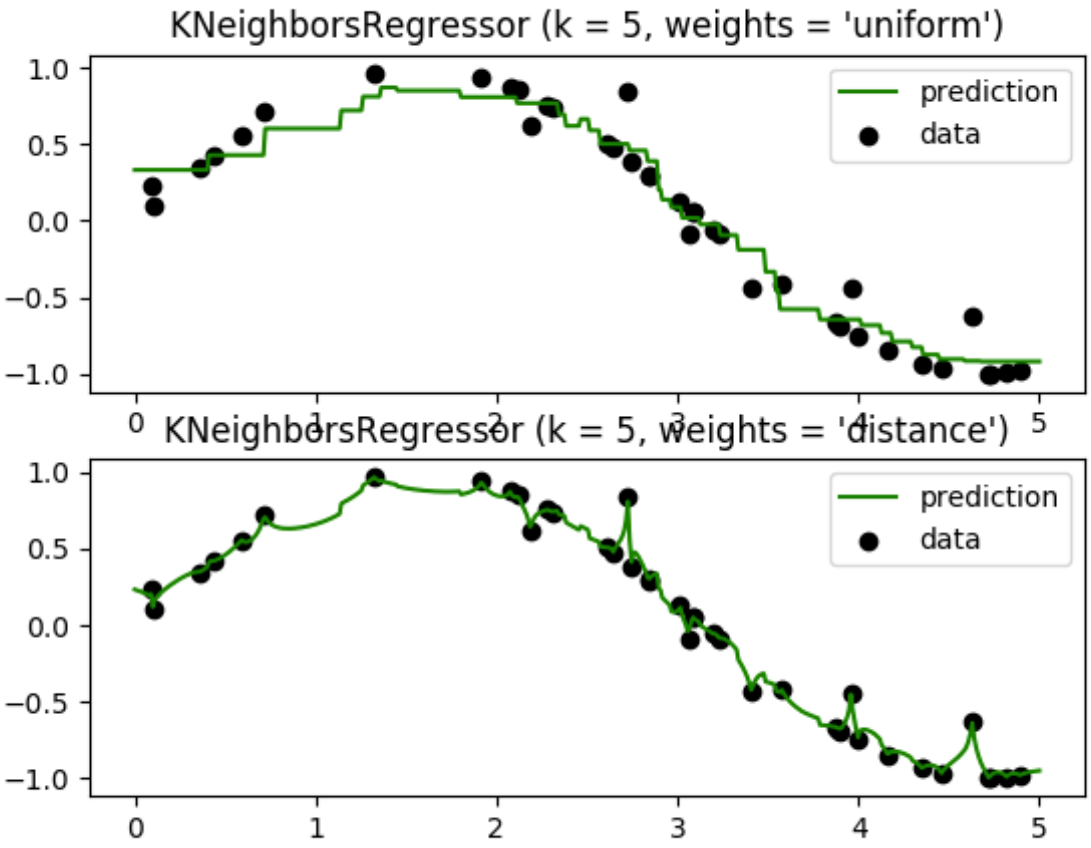
NumPy

- NumPy is the fundamental package for scientific computing with Python. It contains among other things:
  - a powerful N-dimensional array object
  - sophisticated (broadcasting) functions
  - tools for integrating C/C++ and Fortran code
  - useful linear algebra, Fourier transform, and random number capabilities
- NumPy Quickstart tutorial

Non-Parametric Methods

- Methods for approximating discrete-valued or real-valued target functions (classification or regression)
- Learning becomes tied to data storage
- A new instance gets a classification equal to the classification of the nearest instance
- Assumptions:
  - Output varies smoothly with input
  - Non-prior model assumption

k-Nearest Neighbors



Source

Nearest Neighbors

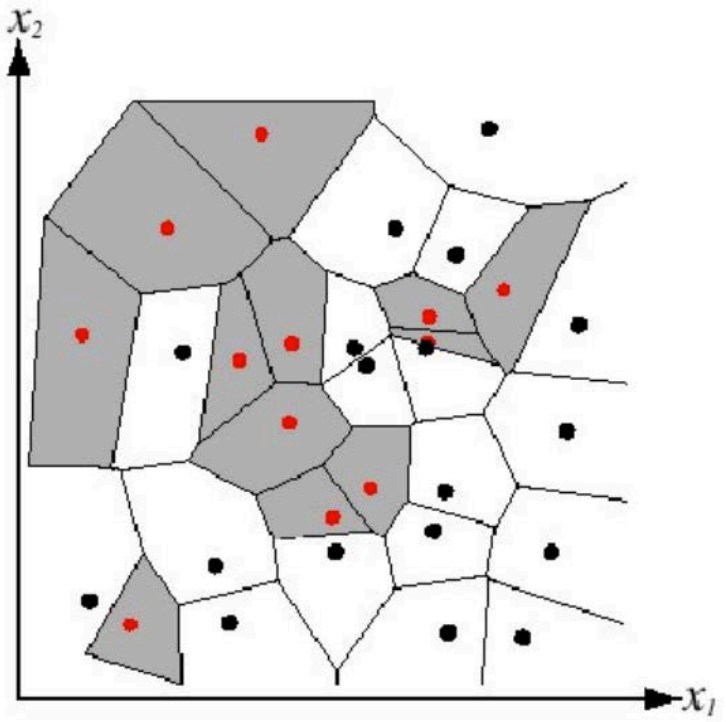
- Training examples correspond to points in d-dim space
- The value of the target function for a new query is estimated from the known value(s) of the nearest training example(s)
- Euclidean distance:

$$||\mathbf{x}^{(a)} - \mathbf{x}^{(b)}||_2 = \sqrt{\sum_{j=1}^{d} (x_j^{(a)} - x_j^{(b)})^2}$$

Nearest Neighbors Boundaries

- Nearest neighbor algorithm does not explicitly compute decision boundaries, but these can be inferred

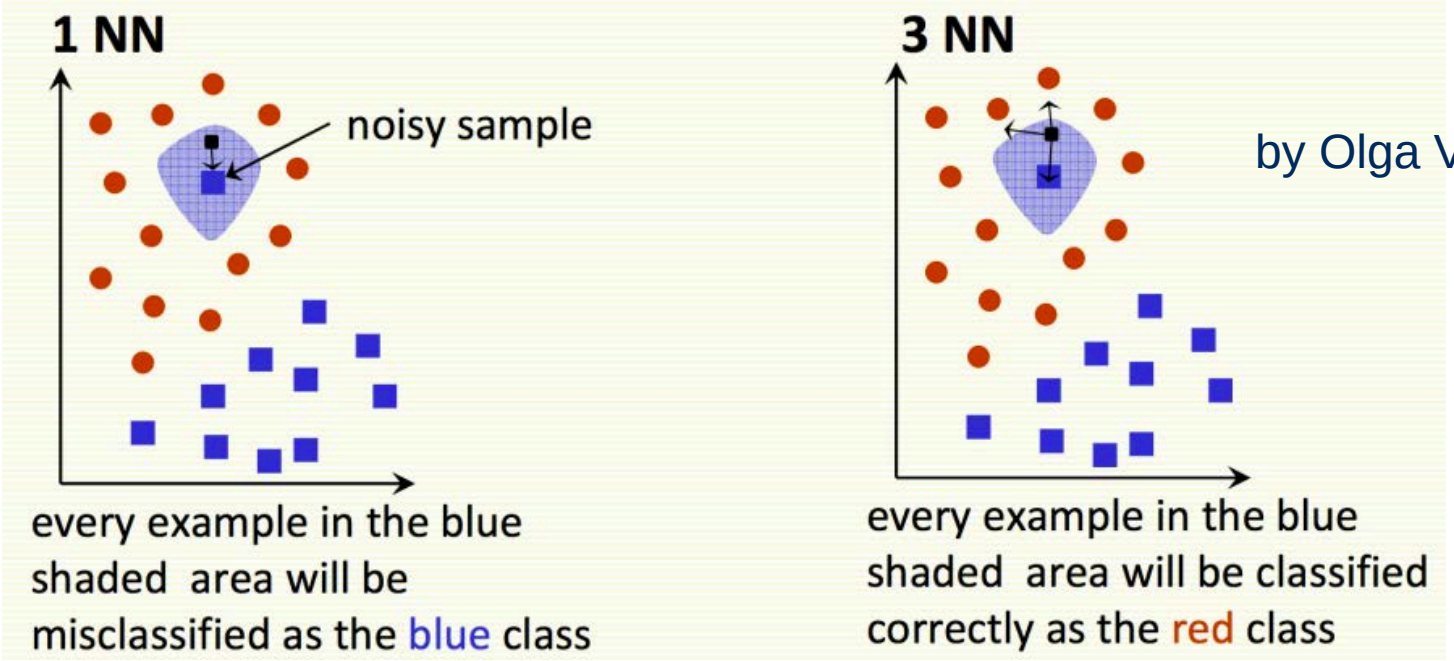
- Decision boundaries: Voronoi diagram visualization
  - show how input space divided into classes
  - each line segment is equidistant between two points of opposite classes



Choosing k

- Larger k may lead to better performance
- But if we set k too large we may end up looking at samples that are not neighbors
- We can use cross-validation to find k
- Rule of thumb is k < sqrt(n), where n is the number of training examples

Mislabeled Data



by Olga Veksler

- Nearest neighbors sensitive to mislabeled data ("class noise"). Solution?
  - Smooth by having k nearest neighbors vote

Other Issues and Solutions

- Features have different dimensions
  - Rescale data
- Irrelevant correlated features
  - Eliminate features, add weights to distance
- Non-metric attributes
  - Use alternative metric: Edit distance (Hamming)
- Expensive at test time
  - Subset of dimension, kd-trees (sort), approximate distance, remove redundant data
- Storage requirements
  - Condense data

UNIVERSITY OF TORONTO
SCHOOL OF CONTINUING STUDIES

Module 1 – Section 7

Resources and Wrap-up

Homework

- Complete the notebook in the assignments section for this week
- Submit your solution here
- Make sure you rename your notebook to
  - W1_UTORid.ipynb
  - Example: W1_adfasd01.ipynb

Next Class

- End to End Machine Learning Project
- Reading: Chapter 2 textbook

Follow us on social

Join the conversation with us online:

facebook.com/uoftscs

@uoftscs

linkedin.com/company/university-of-toronto-school-of-continuing-studies

@uoftscs

Any questions?

UNIVERSITY OF TORONTO
SCHOOL OF CONTINUING STUDIES

Thank You

Thank you for choosing the University of Toronto School of Continuing Studies