Advanced Data Analysis

DATA 71200

Class 1

Course Details

Course Description

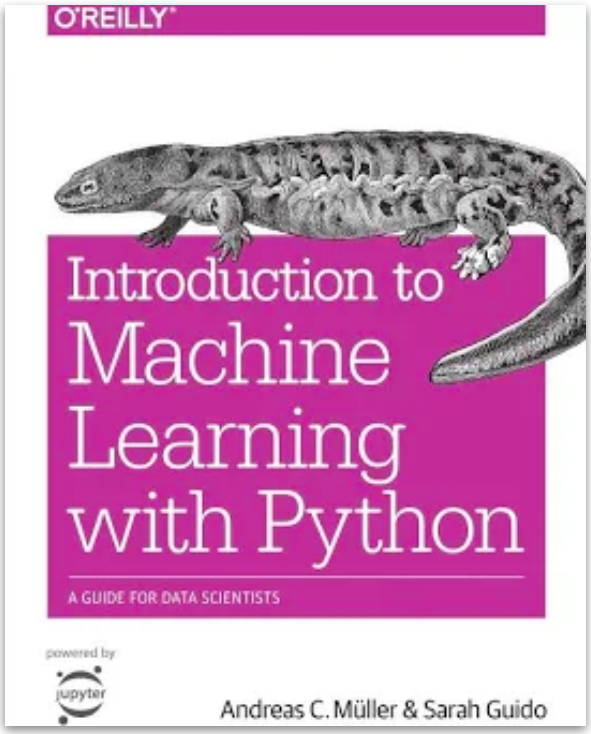
- This course will provide you with skills necessary to a**pply machine learning techniques to data**, and i**nterpret and communicate their results**.
- You will also begin to develop **intuitions** about when machine learning is an appropriate tool versus other statistical methods.
- This course will cover both **supervised methods** (e.g., k-nearest neighbors, naïve Bayes classifiers, decision trees, and support vector machines) and **unsupervised methods** (e.g., principal component analysis and k-means clustering).
    - The supervised methods will focus primarily on **"classic" machine learning techniques** where features are designed rather than learned, although we will briefly look at recent deep learning models with neural networks.
- This is an **applied machine learning class** that emphasizes the intuitions and know-how needed to get learning algorithms to work in practice, rather than mathematical derivations.
- The course will be taught in **Python**, primarily using the **scikit-learn** library.

Course Objectives

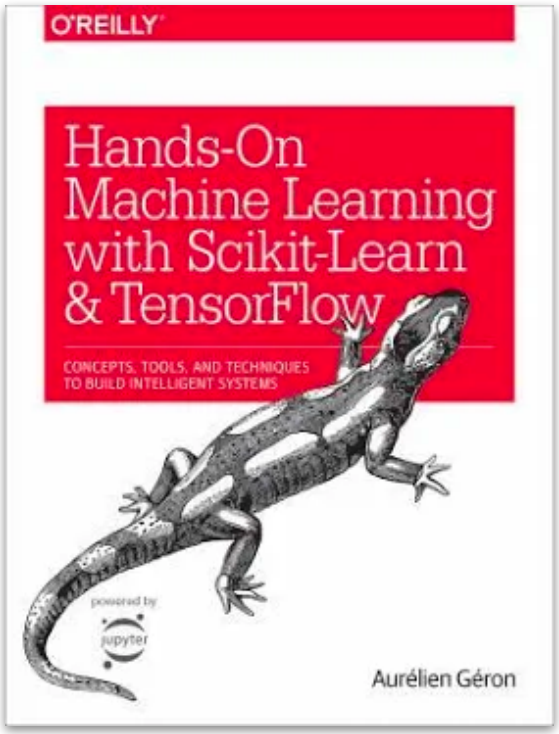
- By the end of the course, you will be able to
  - articulate the main assumptions underlying machine learning approaches
  - demonstrate the basic principles of dataset creation
  - articulate the importance of data representations
  - evaluate machine learning algorithms
  - articulate the difference between supervised and unsupervised learning
  - apply a range of supervised and unsupervised learning techniques

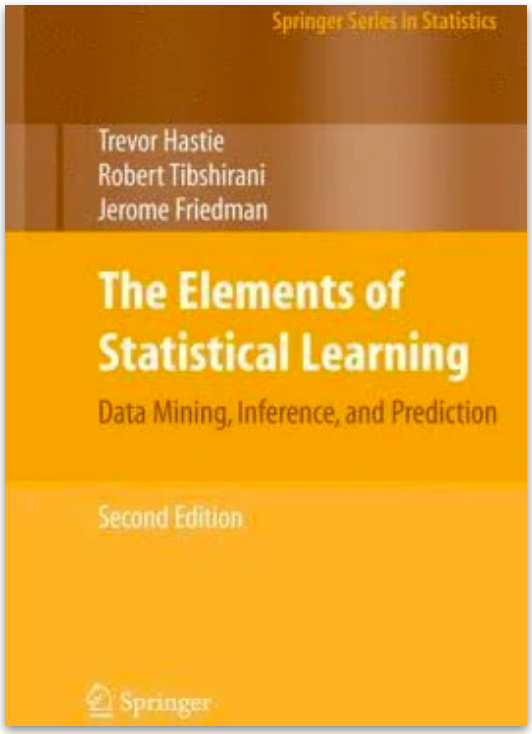
Textbooks

Required



Recommended





Jordan, Michael I. and Tom M. Mitchell. (2015). "Machine Learning: Trends, perspectives, and prospects" *Science.*

Grade Breakdown

Class Participation	10%
Datacamp Assignments	25%
Project 1: Dataset creation	15%
Project 2: Supervised learning	15%
Project 3: Unsupervised learning	15%
Final Paper	20%

Grade Breakdown Details

- **Class Participation: 10%**
  - The participation grade is a combination of attendance (including arriving on time); attentiveness, engagement, and participation during class; and general preparedness for class discussions.
- **Datacamp Assignments: 25%**
  - These projects are hands-on activities designed to both provide coding background and reinforce the concepts covered in class.

Grade Breakdown Details

- **Project 1 (Dataset creation): 15%**
  - Curation and cleaning of a labeled data set that you will use for the supervised and unsupervised learning tasks in project 2 and 3. The dataset can built from existing data and should be stored in your GitHub repostiory.
- **Project 2 (Supervised learning): 15%**
  - Application of two supervised learning techniques on the dataset you created in Project 1. This assignment should be completed as a Jupyter notebook your GitHub repository.

Grade Breakdown Details

- **Project 3 (Unsupervised learning): 15%**
  - Application of two unsupervised learning techniques on the dataset you created in Project 1. This assignment should be completed as a Jupyter notebook your GitHub repository.
- **Final Paper: 20%**
  - A 5–8 page paper describing the work you did in projects 1–3 (your dataset and your supervised and unsupervised experiments). The paper should describe both what you did technically and what you learned from the relative performance of the machine learning approaches you applied to your dataset. This assignment should be posted as a PDF in your GitHub repository.

Course Schedule

28-May	Introduction/ Getting Started with Machine Learning
30-May	Machine Learning Pipeline/ Inspecting Data
3-Jun	Representing Data
4-Jun	*Async: DataCamp Modules*
5-Jun	Evaluation Methods
6-Jun	*Async: DataCamp Modules*

Course Schedule

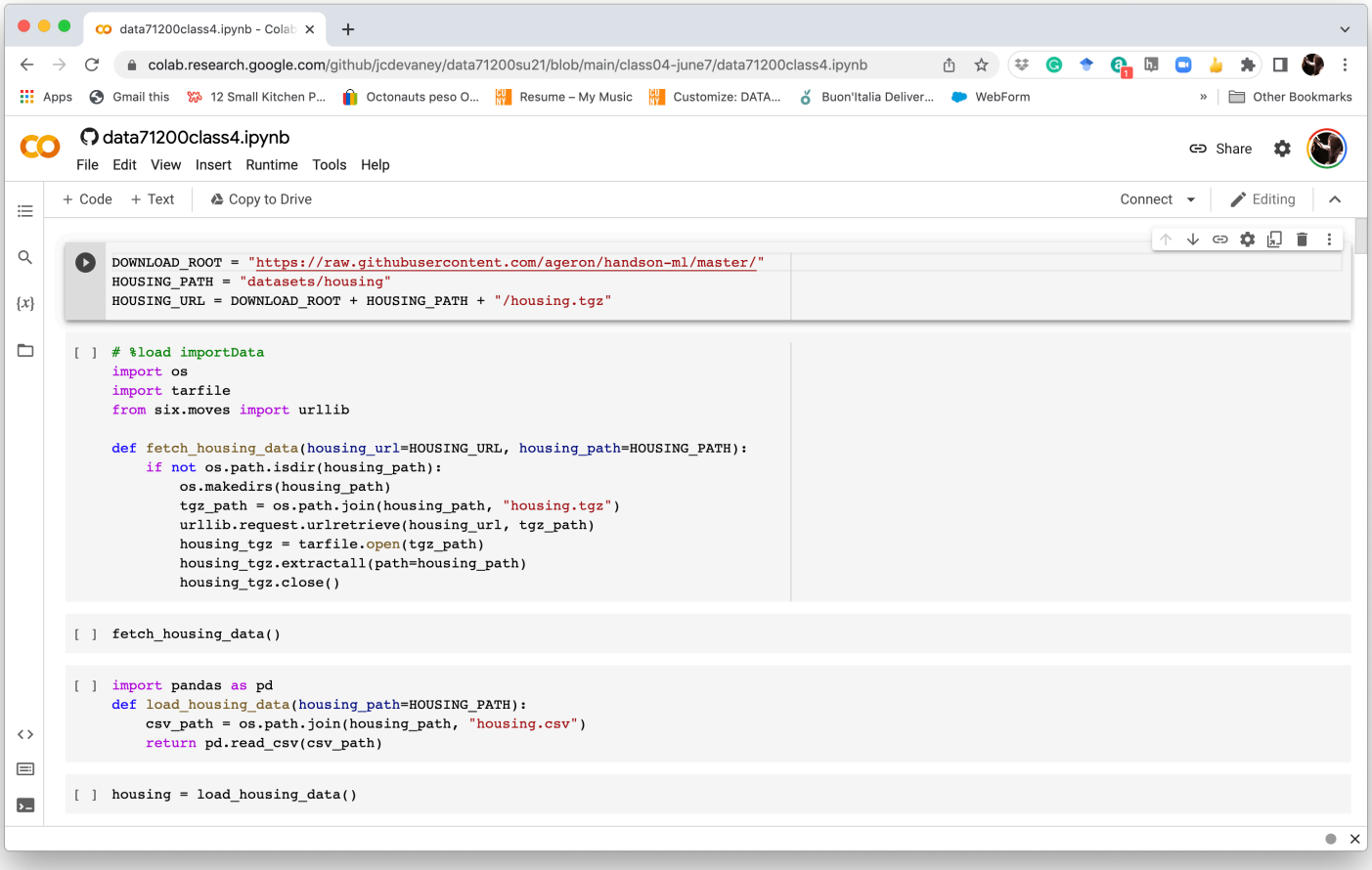
10-Jun	Supervised Learning (k-Nearest Neighbors, Linear Models, and Naive Bayes Classifiers)
11-Jun	*Async: DataCamp Modules*
12-Jun	Supervised Learning (k-Nearest Neighbors, Linear Models, and Naive Bayes Classifiers)
13-Jun	*Async: DataCamp Modules* *Project 1 Due*
17-Jun	Supervised Learning (Decision Trees, Support Vector Machines and Uncertainty estimates from Classifiers)
18-Jun	*Async: DataCamp Modules*

Course Schedule

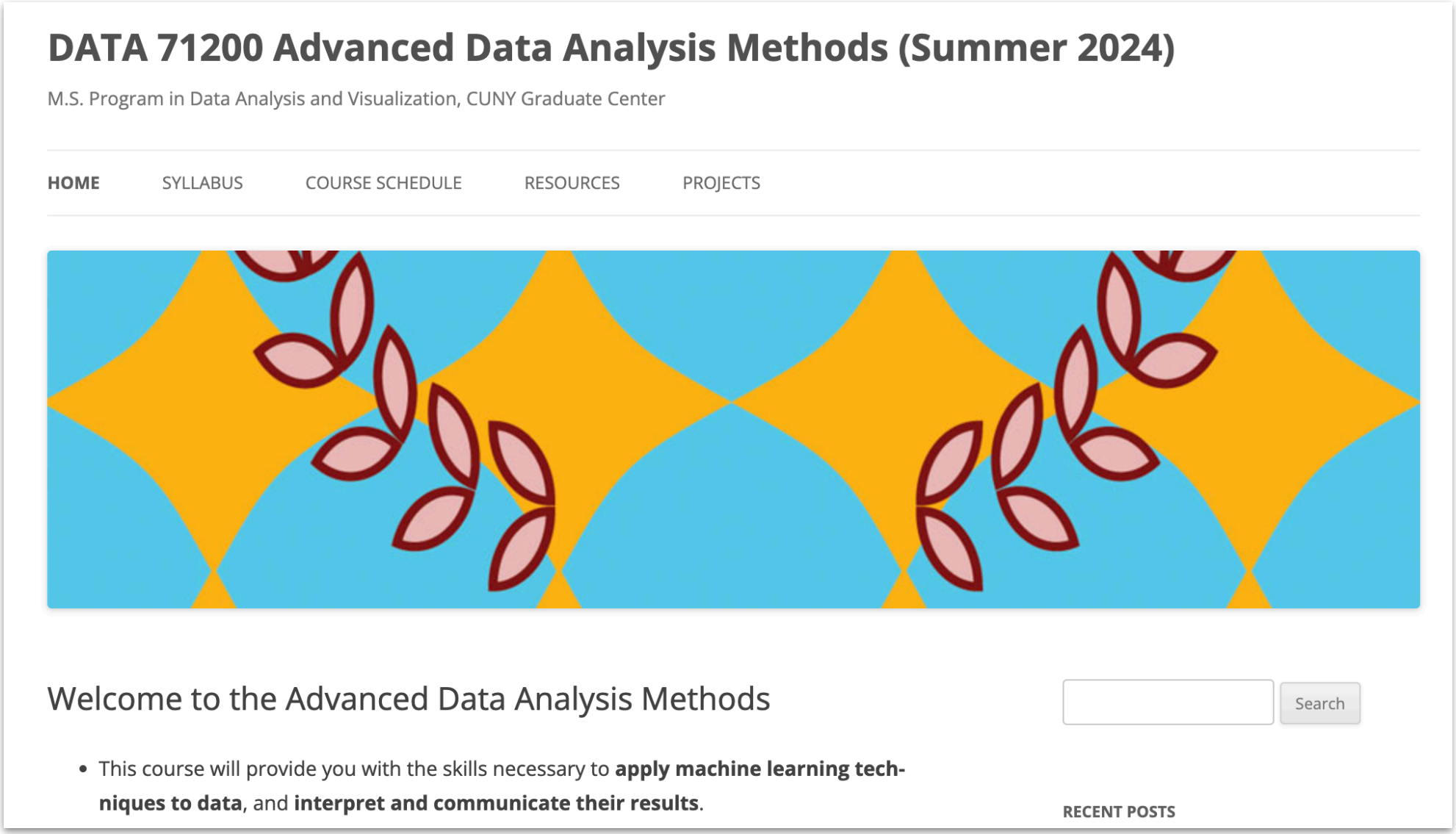
19-Jun	*No Class: Juneteenth*
20-Jun	*Async: DataCamp Modules*
24-Jun	*Unsupervised Learning (Dimensionality Reduction & Feature Extraction, and Manifold Learning)* *Project 2 Due*
25-Jun	*Async: DataCamp Modules*
26-Jun	*Unsupervised Learning (Clustering)*
2-Jul	*Project 3 Due* *Last Day to Finish DataCamp Assignments*
12-Jul	*Final Project Due*

Coding Environment

- **Python 3**
  - matplotlib, NumPy, Pandas, SciPy, scikit learn (+ mglearn)
- **Google Colab**

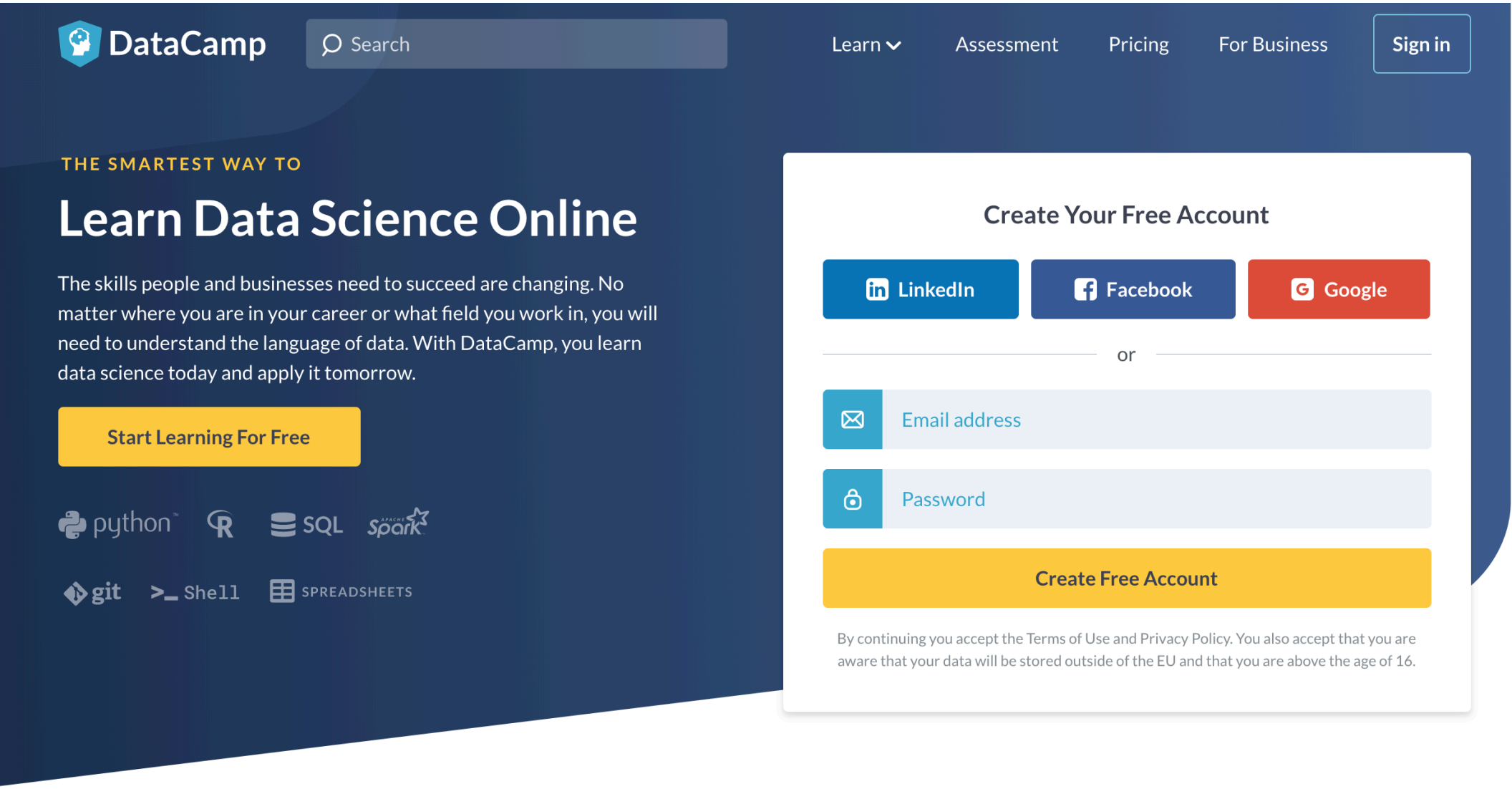


Class Website

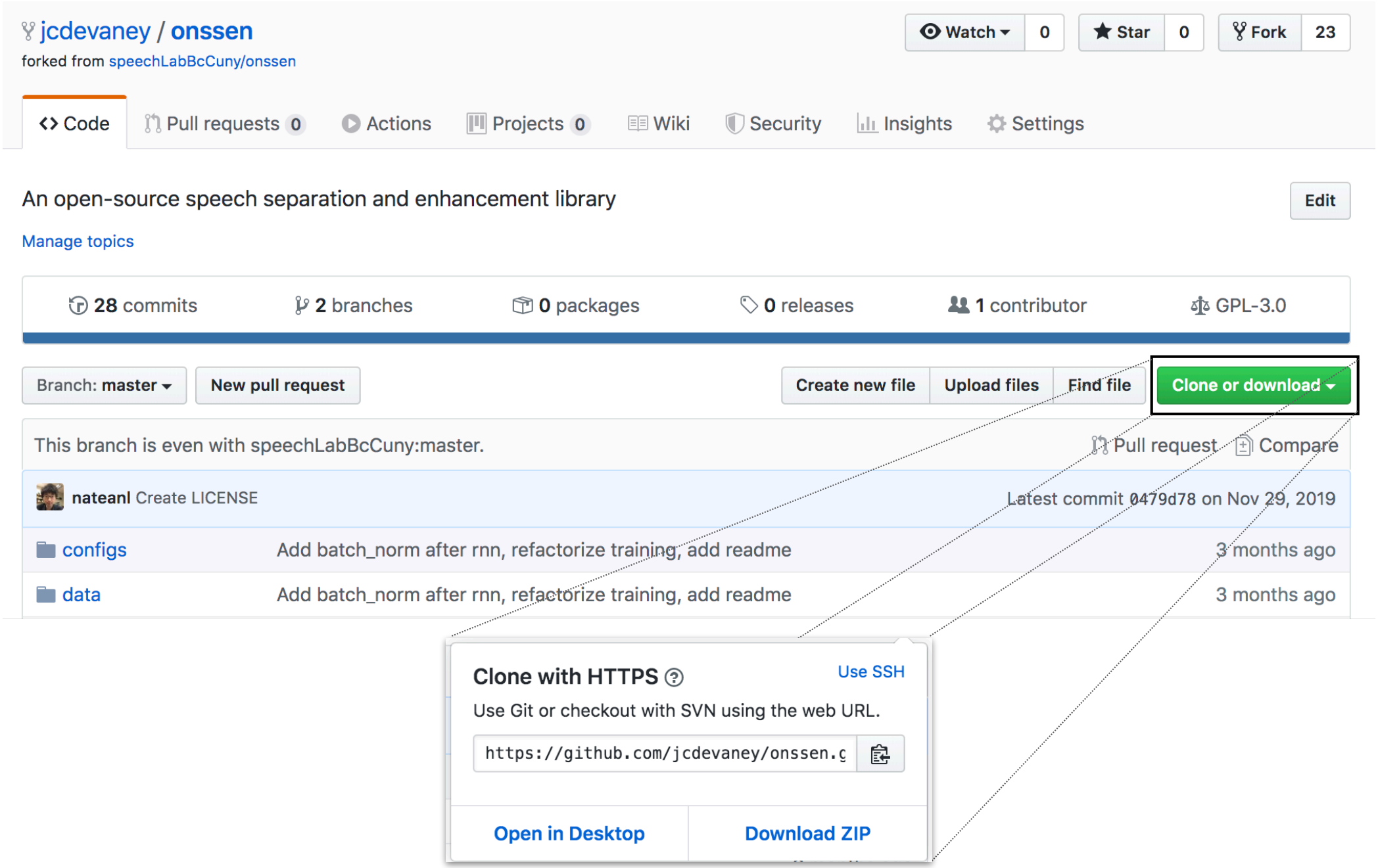


https://data71200su24.commons.gc.cuny.edu/

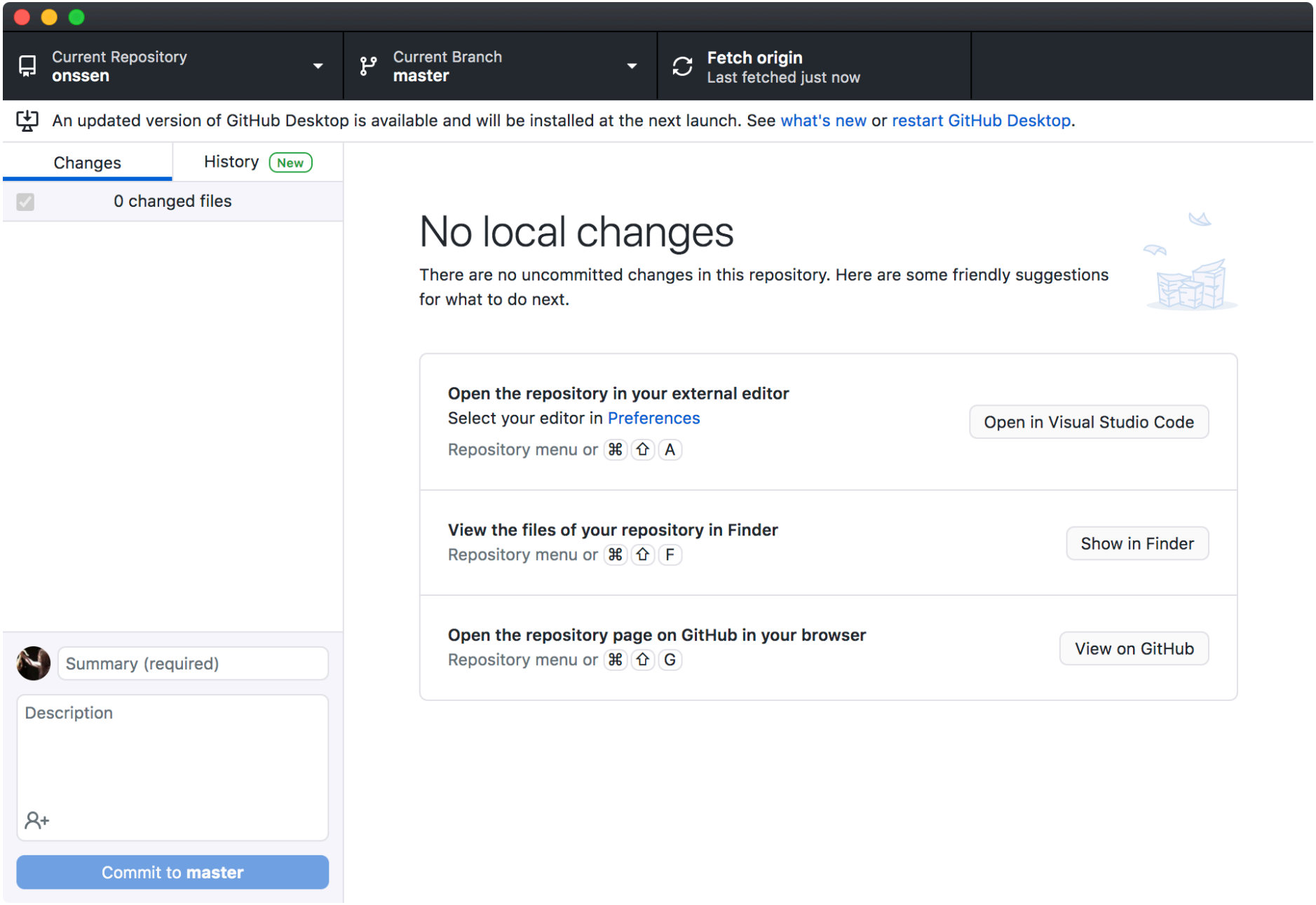
Data Camp


DataCamp
Search
Learn
Assessment
Pricing
For Business
Sign in
THE SMARTEST WAY TO
Learn Data Science Online
The skills people and businesses need to succeed are changing. No matter where you are in your career or what field you work in, you will need to understand the language of data. With DataCamp, you learn data science today and apply it tomorrow.
Start Learning For Free
python
SQL
Spark
git
Shell
SPREADSHEETS
Create Your Free Account
LinkedIn
Facebook
Google
or
Email address
Password
Create Free Account
By continuing you accept the Terms of Use and Privacy Policy. You also accept that you are aware that your data will be stored outside of the EU and that you are above the age of 16.

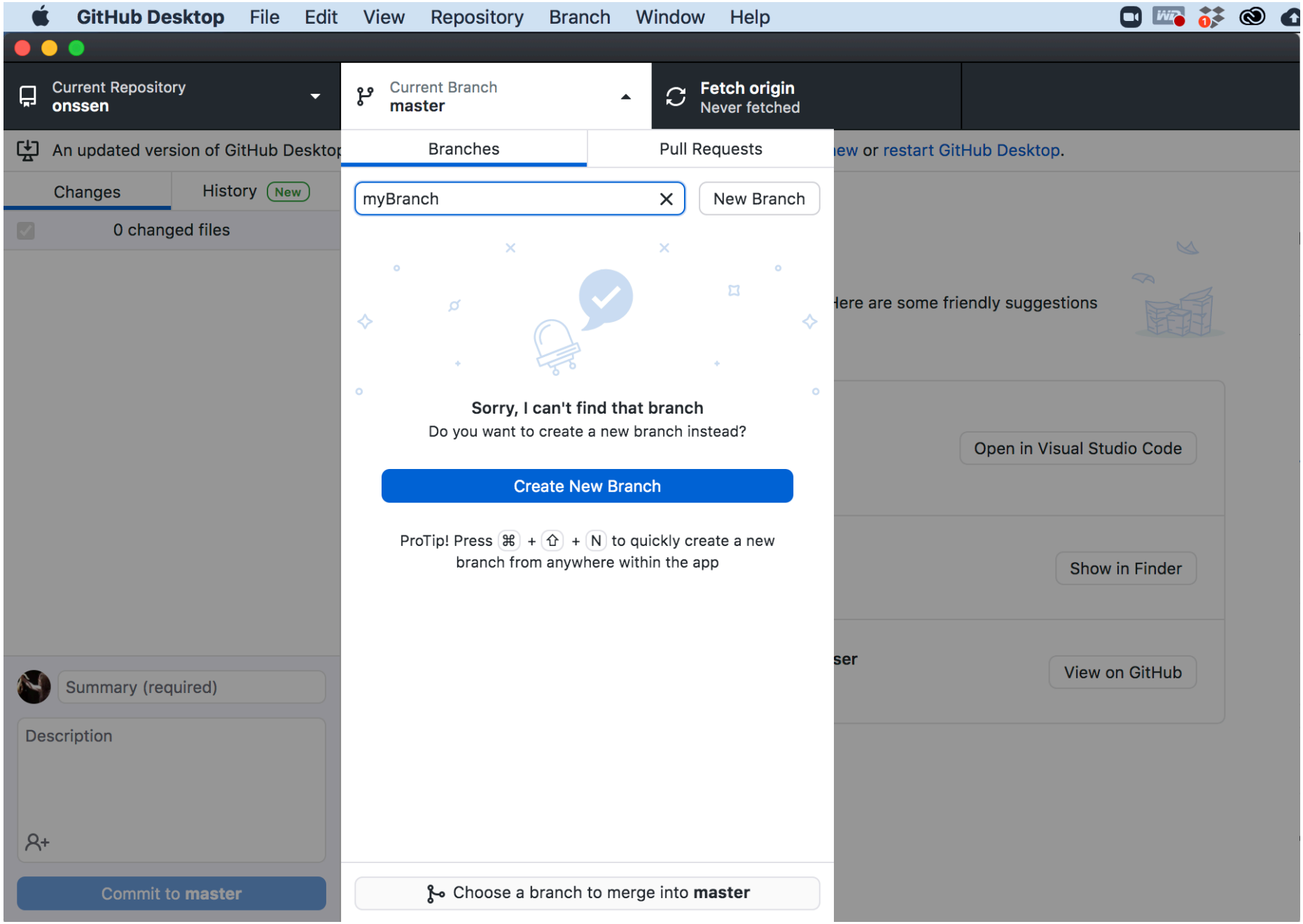
GitHub - Cloning a Repository



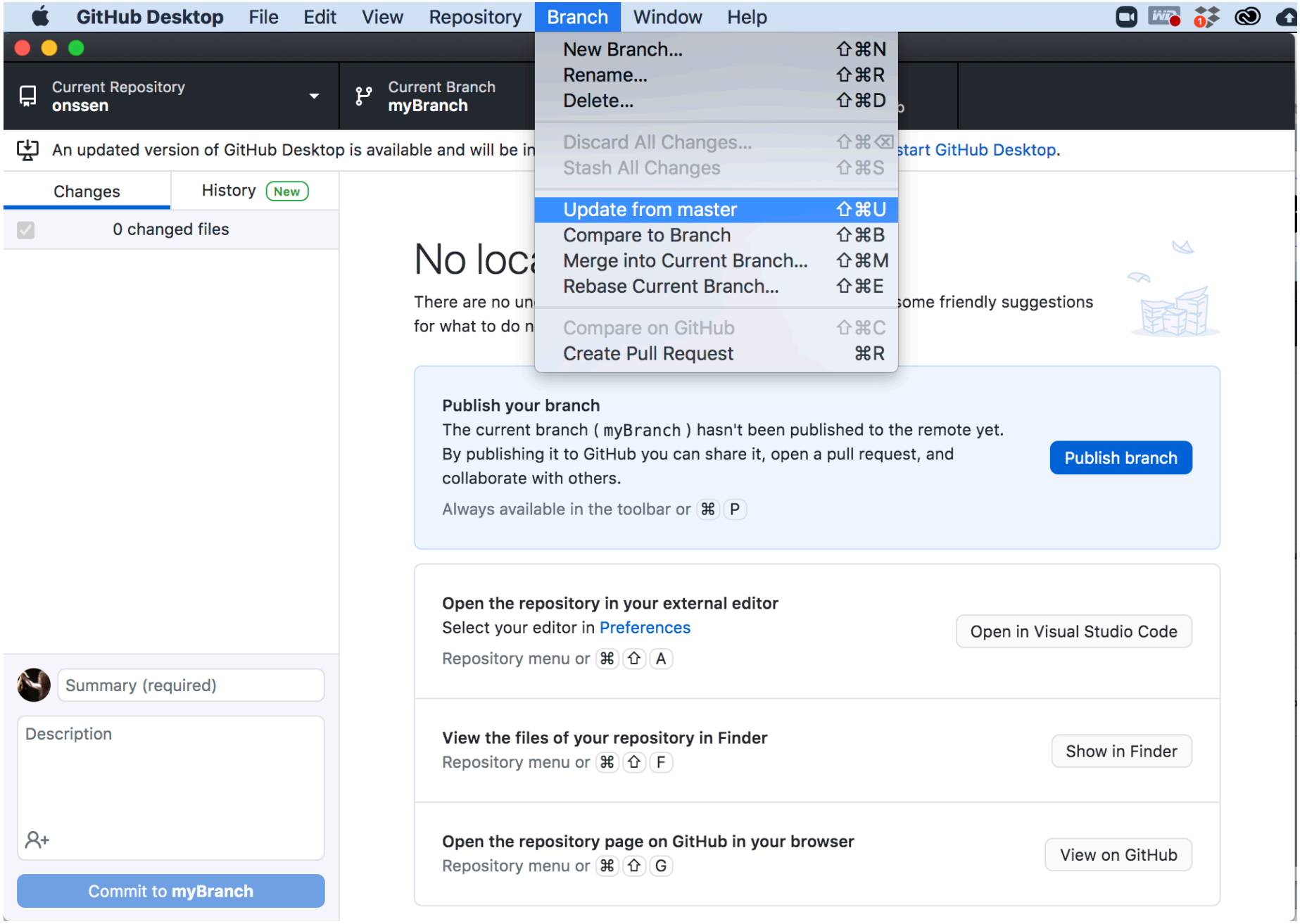
GitHub Desktop



GitHub Desktop - Create a Branch



GitHub Desktop - Update a Branch



Coding

- **Notebooks**
  - It would be useful to the following repositories
    - https://github.com/amueller/introduction_to_ml_with_python
    - https://github.com/ageron/handson-ml
- **Python 3 libraries**
  - import numpy as np
  - import scipy as sp
  - import matplotlib.pyplot as plt
  - import pandas as pd

Notebook
01-introduction.ipynb
[2-8]

https://github.com/amueller/introduction_to_ml_with_python

Coding

Open 01-introduction.ipynb in Google Colab

Copy contents of preamble.py from repo into the first cell, replacing:

```
from preamble import *
```

01-introduction.ipynb

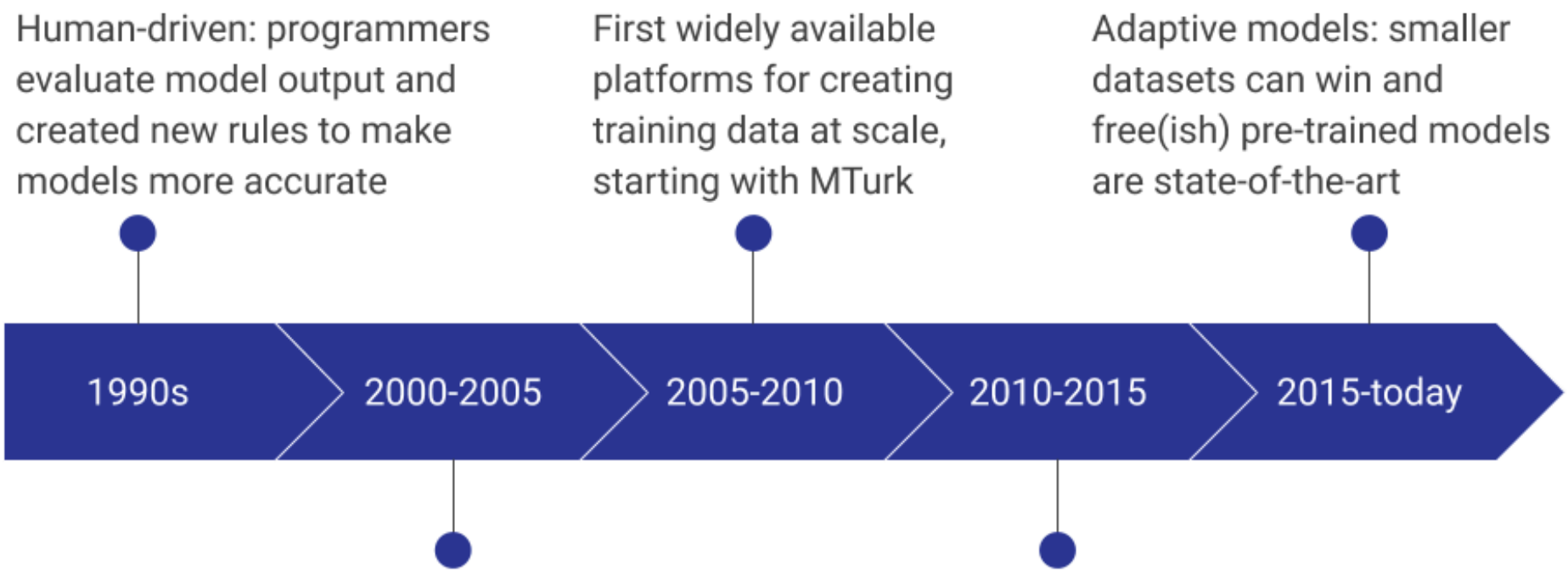
File Edit View Insert Runtime Tools Help Cannot save changes

+ Code + Text | Copy to Drive

```
from IPython.display import set_matplotlib_formats, display
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
!pip install mglearn
import mglearn
from cycler import cycler

set_matplotlib_formats('pdf', 'png')
plt.rcParams['savefig.dpi'] = 300
plt.rcParams['image.cmap'] = "viridis"
plt.rcParams['image.interpolation'] = "none"
plt.rcParams['savefig.bbox'] = "tight"
plt.rcParams['lines.linewidth'] = 2
plt.rcParams['legend.numpoints'] = 1
plt.rc('axes', prop_cycle=(
    cycler('color', mglearn.plot_helpers.cm_cycle.colors) +
    cycler('linestyle', ['-', '-', "--", (0, (3, 3)), (0, (1.5, 1.5))])))

np.set_printoptions(precision=3, suppress=True)

pd.set_option("display.max_columns", 8)
pd.set_option('display.precision', 2)

__all__ = ['np', 'mglearn', 'display', 'plt', 'pd']

%matplotlib inline
```

Introduction to Machine Learning

Classic Machine Learning



https://towardsdatascience.com/a-brief-history-of-training-data-9c513fc95b3e

Key Questions

- **“How can one construct computer systems that automatically improve through experience?”**
- **“What are the fundamental statistical-computational-information-theoretic laws that govern all learning systems, including computers, humans, and organizations?”**
- **“How accurately can the algorithm learn from a particular type and volume of training data?”**
- **“How robust is the algorithm to errors in its modeling assumptions or to errors in the training data”**

Jordan, Michael I. and Tom M. Mitchell. (2015). “Machine Learning: Trends, perspectives, and prospects” *Science*.

Machine Learning vs Traditional Programming

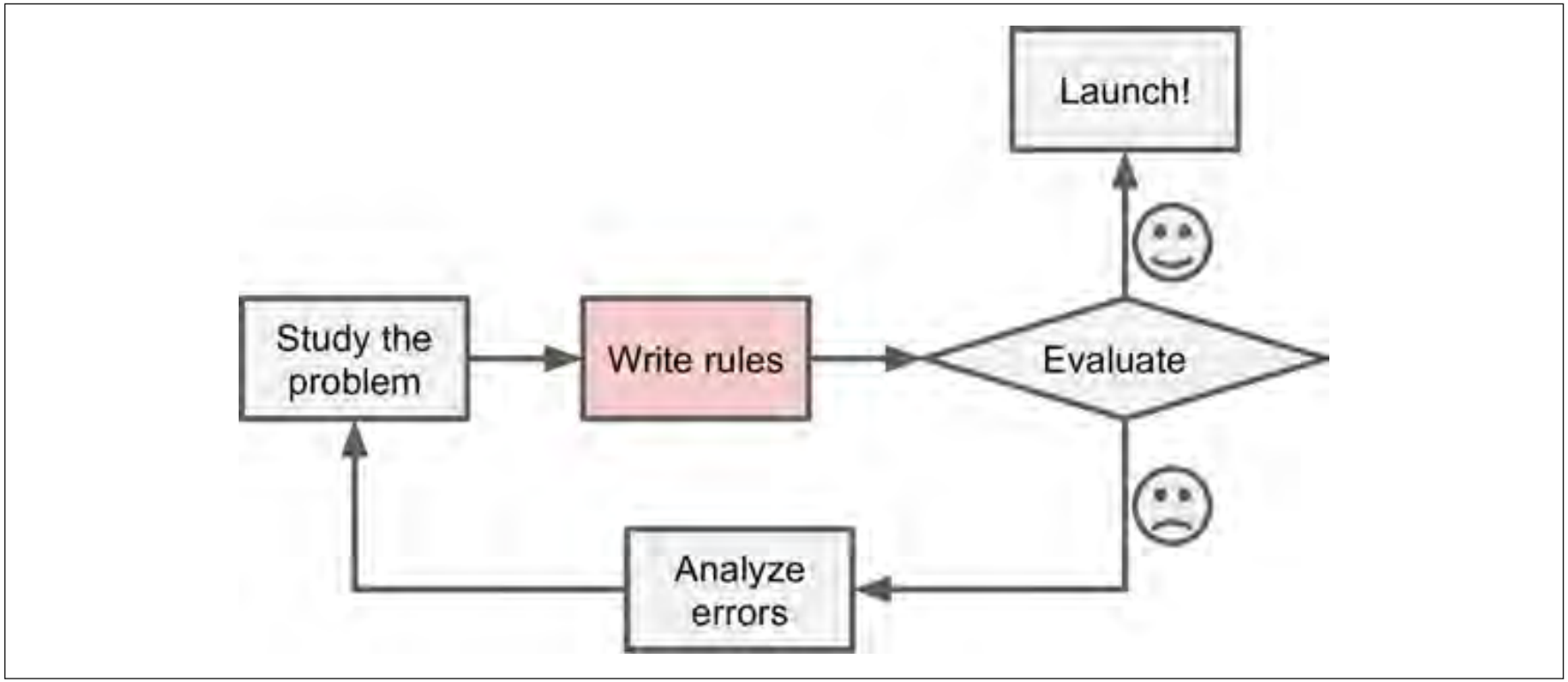


Figure 1-1. The traditional approach

Image from: Géron, Aurélien. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow* O'Reilly Media, Inc.

Machine Learning vs Traditional Programming

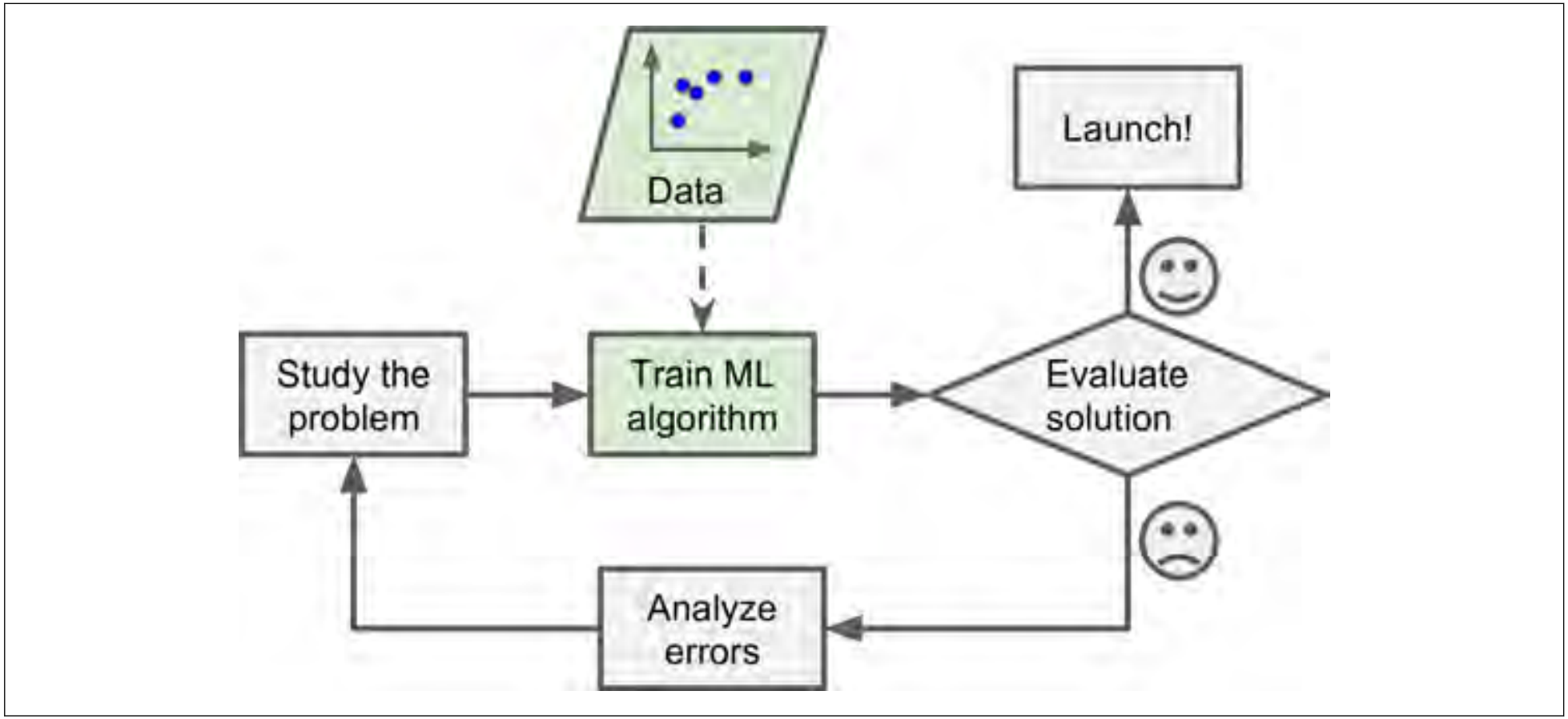


Figure 1-2. Machine Learning approach

Image from: Géron, Aurélien. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow* O'Reilly Media, Inc.

Challenges

- **“huge data sets require computationally tractable algorithms”**
- **“highly personal data raise the need for algorithms that minimize privacy effects”**
- **“the availability of huge quantities of unlabeled data raises the challenge of designing learning algorithms to take advantage of it”**

Jordan, Michael I. and Tom M. Mitchell. (2015). “Machine Learning: Trends, perspectives, and prospects” *Science*.

Supervised Learning

- **Function approximation problem**
  - "the training data take the form of a collection of (x, y) pairs and the goal is to produce a prediction y* in response to a query x*"
  - Task is to learn a mapping, f(x), which outputs a y value for each inputted x value

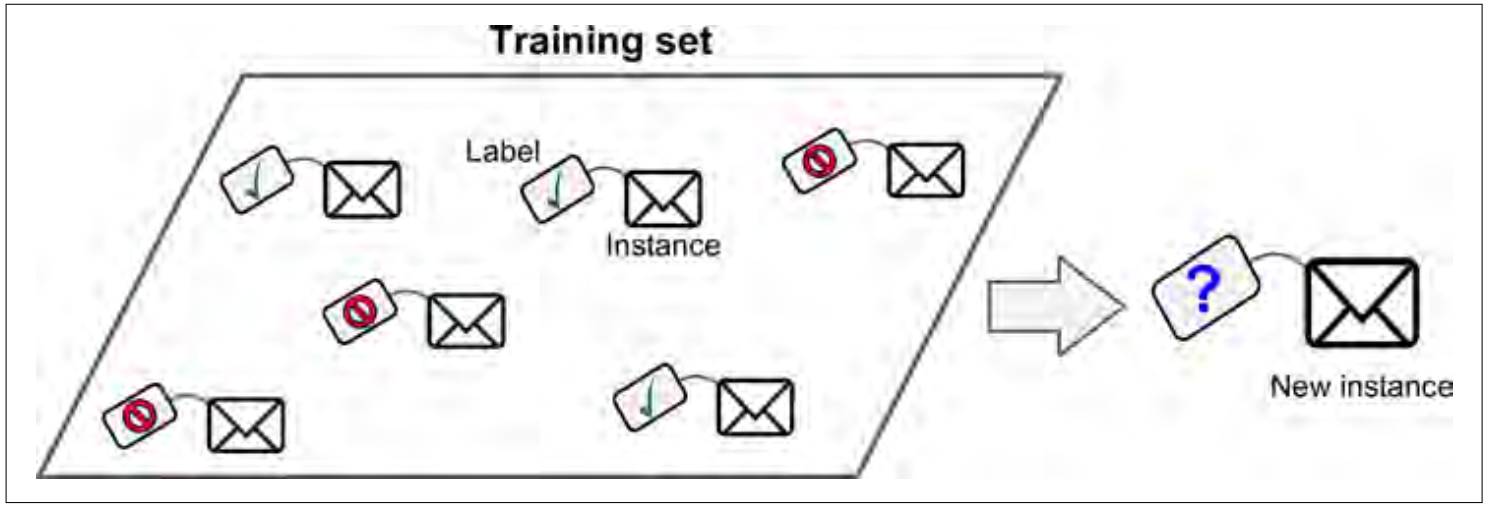


Figure 1-5. A labeled training set for supervised learning (e.g., spam classification)

Jordan, Michael I. and Tom M. Mitchell. (2015). "Machine Learning: Trends, perspectives, and prospects" *Science*.
Image from: Géron, Aurélien. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow* O'Reilly Media, Inc.

Supervised Learning

- ***k*-Nearest Neighbors**
- **Linear Regression**
- **Logistic Regression**
- **Support Vector Machines (SVMs)**
- **Decision Trees and Random Forests**
- **Naive Bayes Classifiers**
- ***Neural networks***

Géron, Aurélien. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow* O'Reilly Media, Inc.

Supervised Learning

- **“diversity of learning architectures and algorithms reflects the diverse needs of applications”**
  - “with different architectures capturing different kinds of mathematical structures, offering different levels of amenability to post-hoc visualization and explanation, and providing varying trade-offs between computational complexity, the amount of data, and performance.”

Jordan, Michael I. and Tom M. Mitchell. (2015). “Machine Learning: Trends, perspectives, and prospects” *Science*.

Unsupervised Learning

- **“the analysis of unlabeled data under assumptions about structural properties of the data (e.g., algebraic, combinatorial, or probabilistic)”**

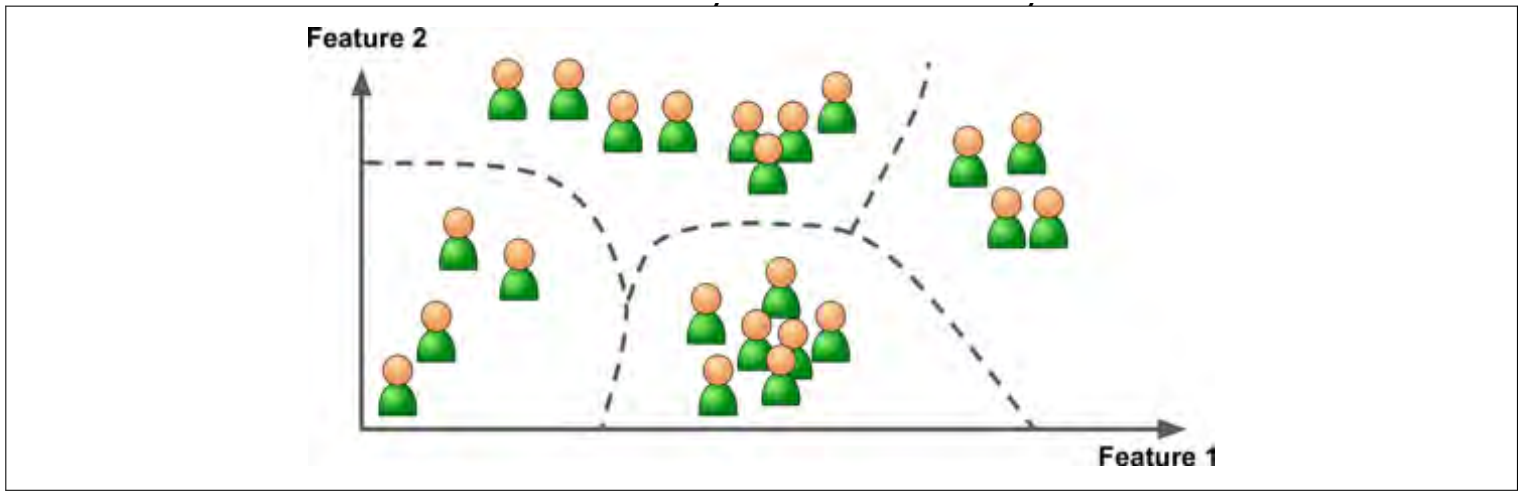


Figure 1-8. Clustering

Jordan, Michael I. and Tom M. Mitchell. (2015). “Machine Learning: Trends, perspectives, and prospects” *Science*.
Image from: Géron, Aurélien. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow* O’Reilly Media, Inc.

Unsupervised Learning

- **The models make the assumption "that data lie on a low-dimensional manifold and aim to identify that manifold explicitly from the data"**
  - Dimensionality reduction (e.g., PCA)
  - Clustering (e.g., *k*-means)

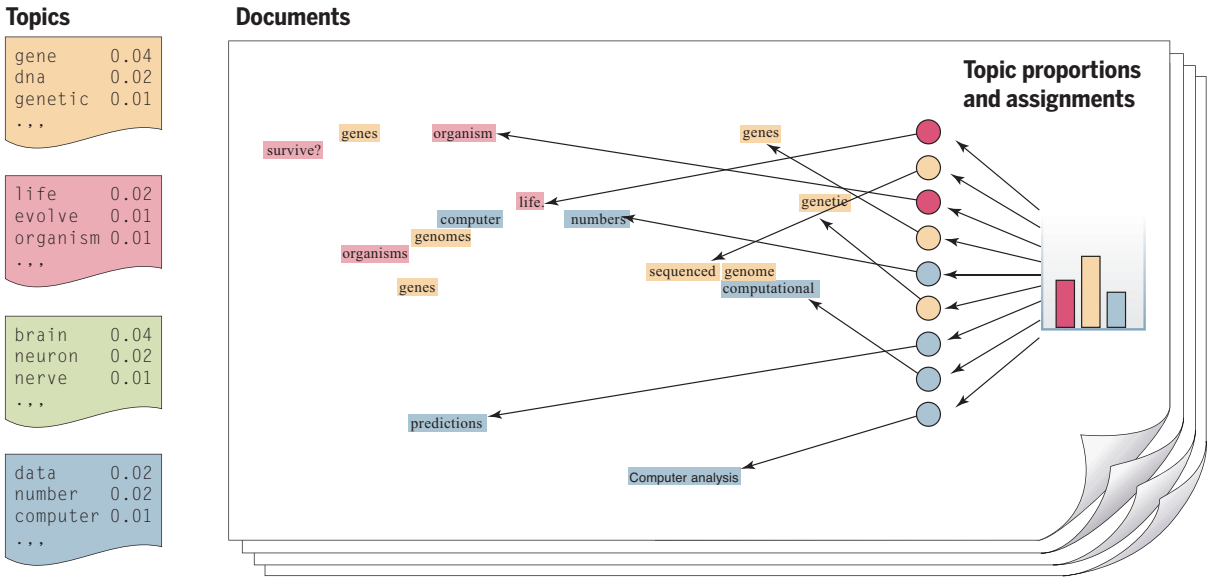


Fig. 3. Topic models. Topic modeling is a methodology for analyzing documents, where a document is viewed as a collection of words, and the words in the document are viewed as being generated by an underlying set of topics (denoted by the colors in the figure). Topics are probability distributions across words (leftmost column), and each document is characterized by a probability distribution across topics (histogram). These distributions are inferred based on the analysis of a collection of documents and can be viewed to classify, index, and summarize the content of documents. [From (*31*). Copyright 2012, Association for Computing Machinery, Inc. Reprinted with permission]

Text and image from Jordan, Michael I. and Tom M. Mitchell. (2015). "Machine Learning: Trends, perspectives, and prospects" *Science*.

Feature Engineering

- **Feature selection**
  - "selecting the most useful features to train on among existing features"
- **Feature extraction**
  - "combining existing features to produce a more useful one (as we saw earlier, dimensionality reduction algorithms can help)"
- **"Creating new features by gathering new data"**

Géron, Aurélien. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow* O'Reilly Media, Inc.

Typical Machine Learning Project Steps

- **“You studied the data.”**
- **“You selected a model.”**
- **Feature Engineering**
- **“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.”**

Géron, Aurélien. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow* O’Reilly Media, Inc.

Main Challenges

- **Insufficient training data**
  - Quantity and/or quality and/or non-representative
- **Irrelevant features**
- **Overfitting training data**
- **Under-fitting training data**

Géron, Aurélien. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow* O'Reilly Media, Inc.

Example: GDP and Happiness

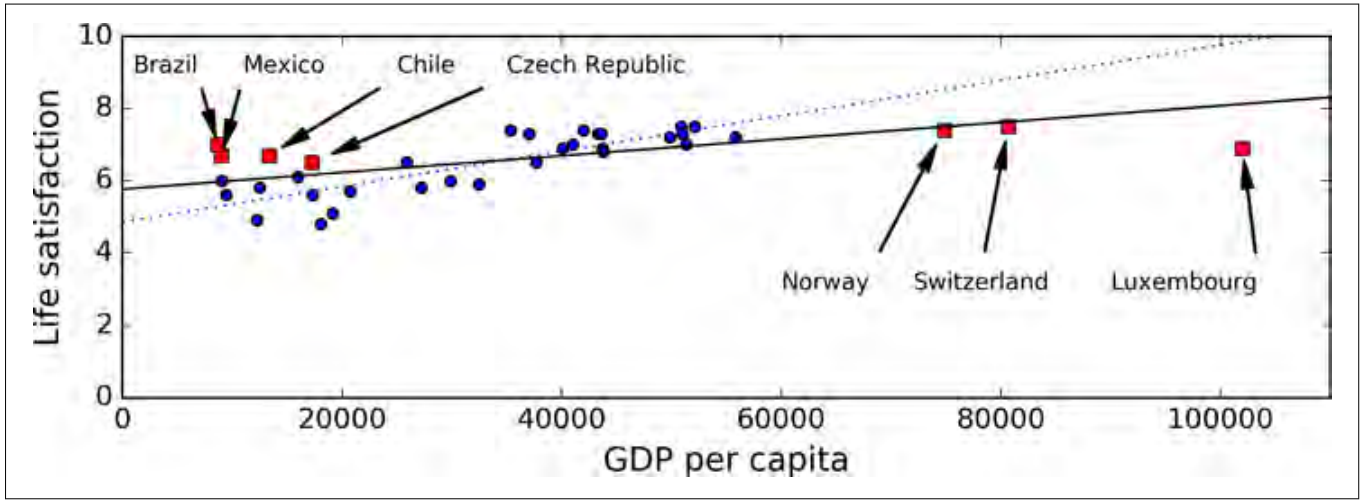


Figure 1-21. A more representative training sample

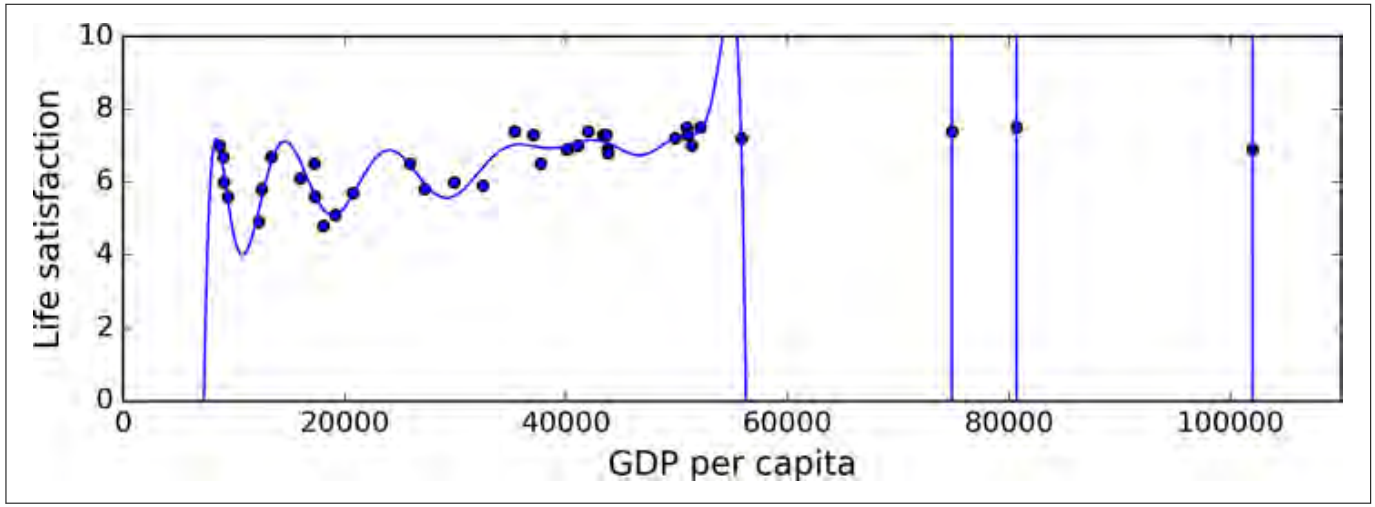


Figure 1-22. Overfitting the training data

Images from: Géron, Aurélien. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow* O'Reilly Media, Inc.

Example: GDP and Happiness

- **regularization**
  - "constraining a model to make it simpler and reduce the risk of overfitting"
- **hyperparameter**
  - "amount of regularization to apply during learning"
  - "need too be set before training"

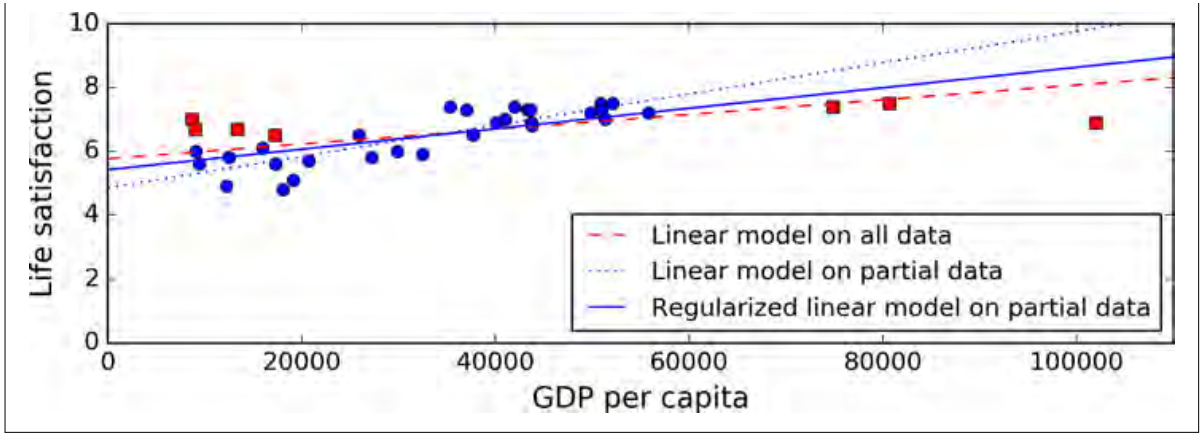


Figure 1-23. Regularization reduces the risk of overfitting

Images from: Géron, Aurélien. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow* O'Reilly Media, Inc.

Data Set Terminology

- **Training set**
  - data used to train the model
- **Testing set**
  - hold out data used to estimate the generalization error on new data
- **Validation set**
  - used to compare models
- **Cross-validation**
  - iteratively holding out a subset of the training data and testing on the rest (typically 80/20: 5-fold cross-validation)

More Terminology

- **Class**
  - "One of a set of enumerated target values for a label."
- **Classification**
  - "A type of machine learning model for distinguishing among two or more discrete classes."

https://developers.google.com/machine-learning/glossary

More Terminology

- **Samples**
  - Individual items
  - **Label**
    - “In supervised learning, the "answer" or "result" portion of an example”
  - **Feature**
    - “An input variable used in making predictions.”

https://developers.google.com/machine-learning/glossary

Data Analysis in Practice

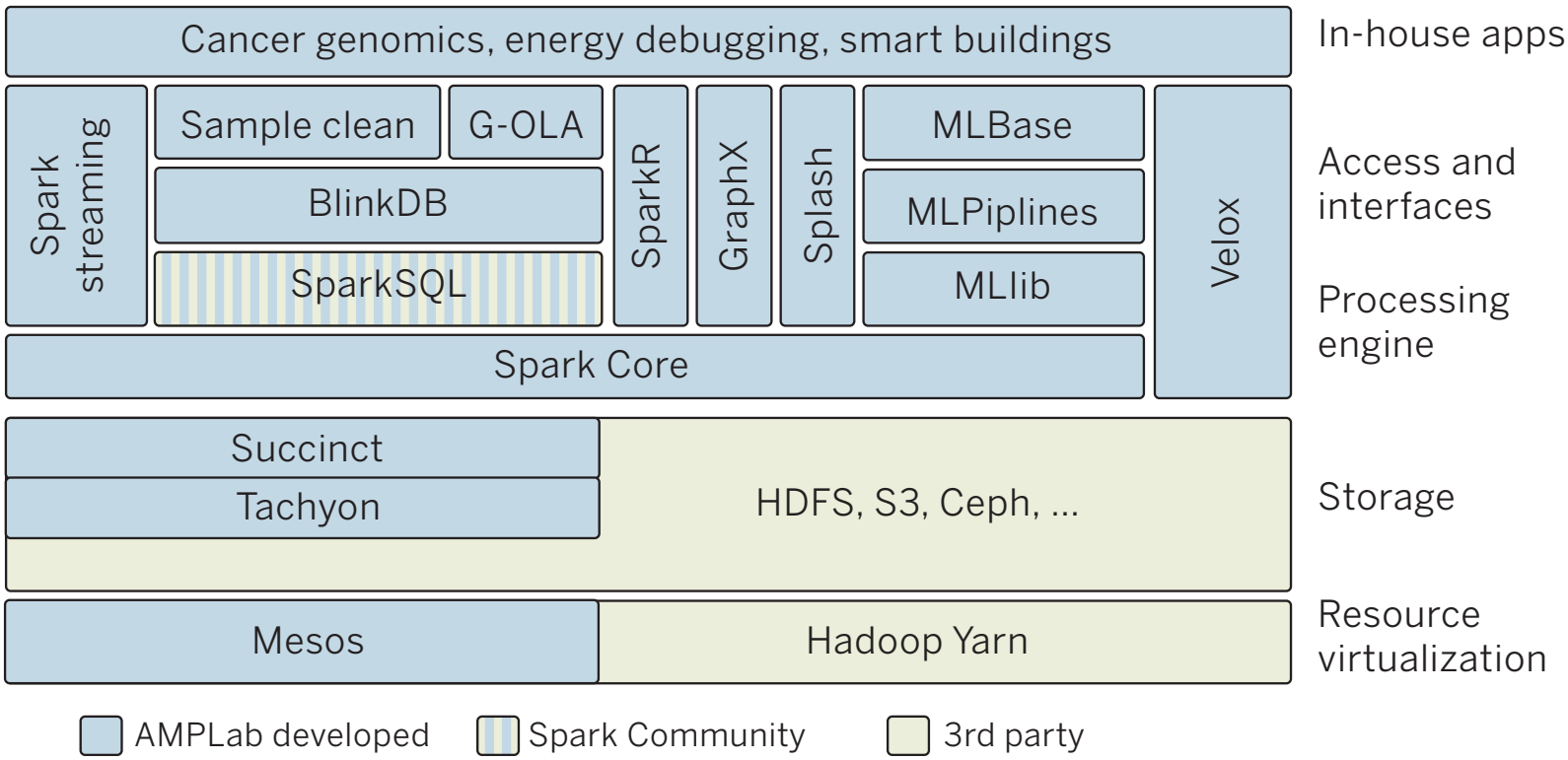


Fig. 5. Data analytics stack. Scalable machine-learning systems are layered architectures that are built on parallel and distributed computing platforms. The architecture depicted here—an open-source data analysis stack developed in the Algorithms, Machines and People (AMP) Laboratory at the University of California, Berkeley—includes layers that interface to underlying operating systems; layers that provide distributed storage, data management, and processing; and layers that provide core machine-learning competencies such as streaming, subsampling, pipelines, graph processing, and model serving.

Image from: Jordan, Michael I. and Tom M. Mitchell. (2015). "Machine Learning: Trends, perspectives, and prospects" *Science*.

Deep Learning

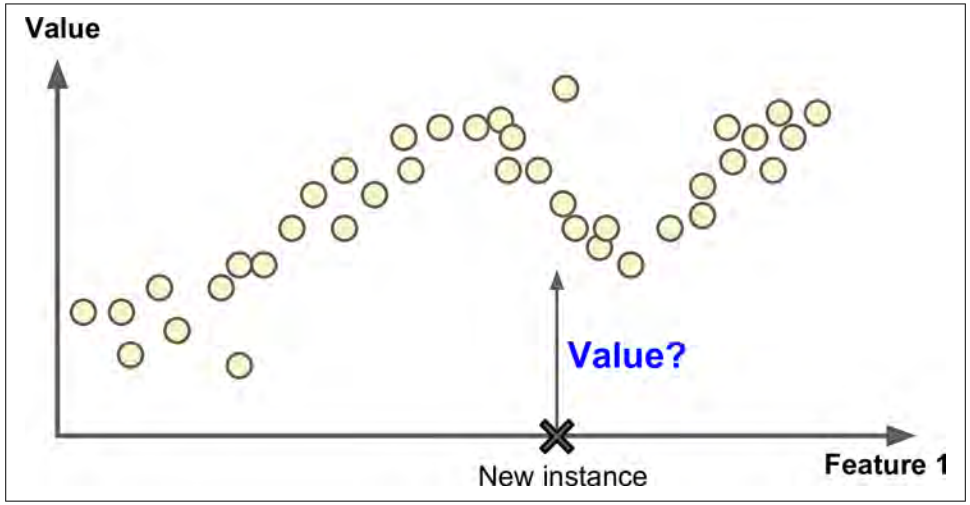


Figure 1-6. Regression

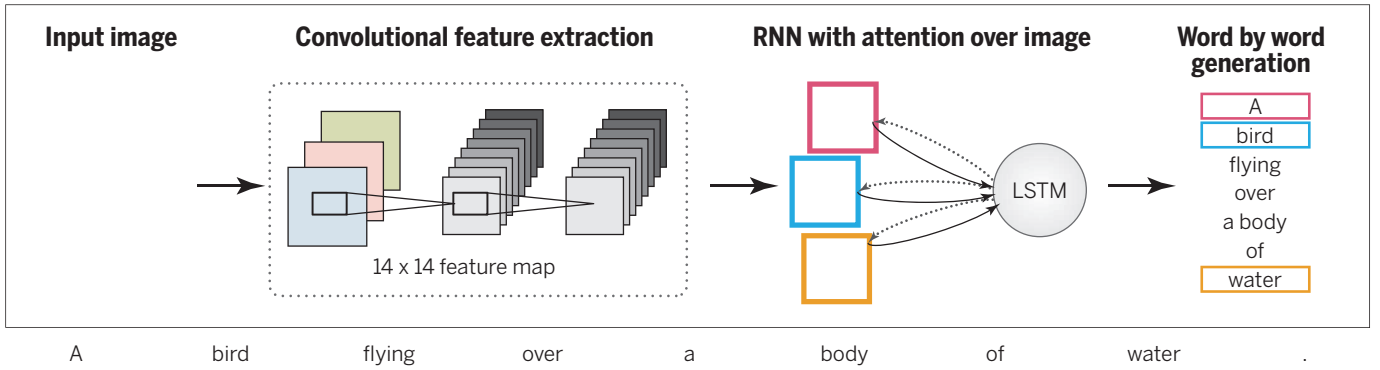


Fig. 2. Automatic generation of text captions for images with deep networks. A convolutional neural network is trained to interpret images, and its output is then used by a recurrent neural network trained to generate a text caption (top). The sequence at the bottom shows the word-by-word focus of the network on different parts of input image while it generates the caption word-by-word. [Adapted with permission from (*30*)]

Top image from: Géron, Aurélien. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow* O'Reilly Media, Inc.
Bottom image from: Jordan, Michael I. and Tom M. Mitchell. (2015). "Machine Learning: Trends, perspectives, and prospects" *Science*.

Instance versus Model-Based Learning

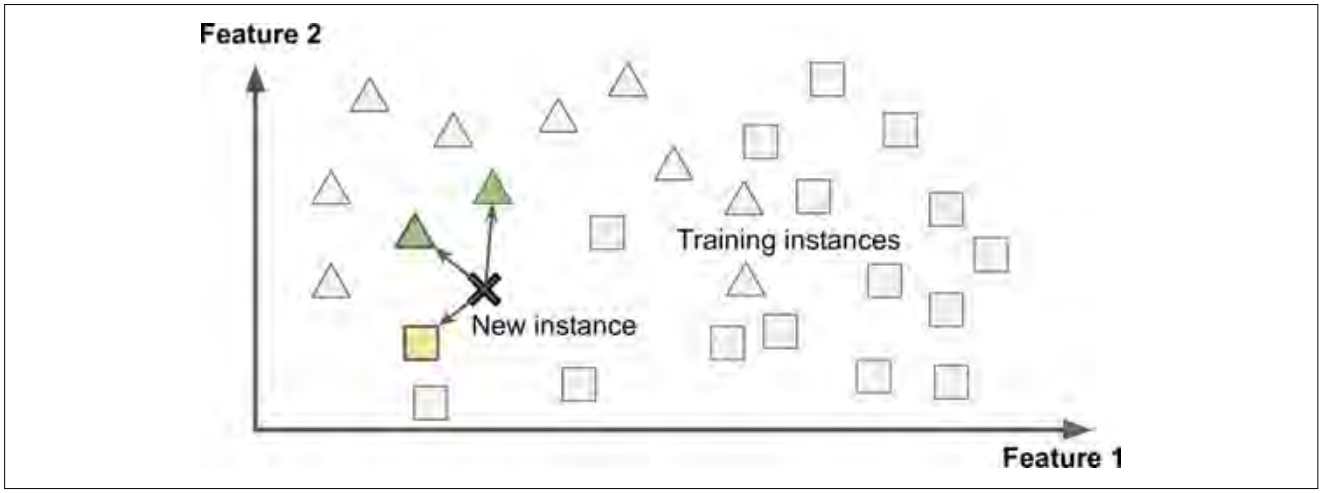


Figure 1-15. Instance-based learning

“the system learns the examples by heart, then generalizes to new cases using a similarity measure”

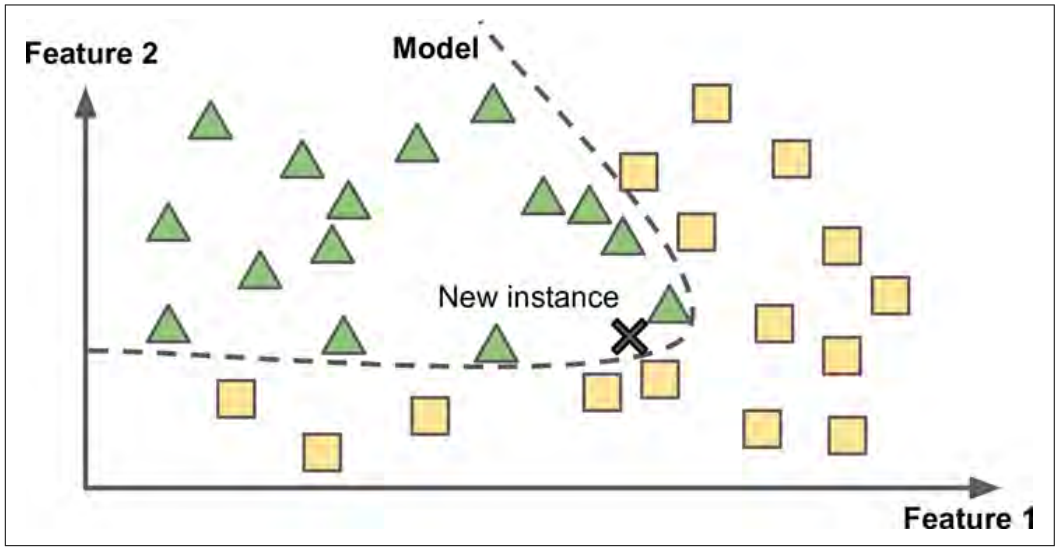


Figure 1-16. Model-based learning

"another way to generalize from a set of examples is to build a model of these examples, then use that model to make predictions”

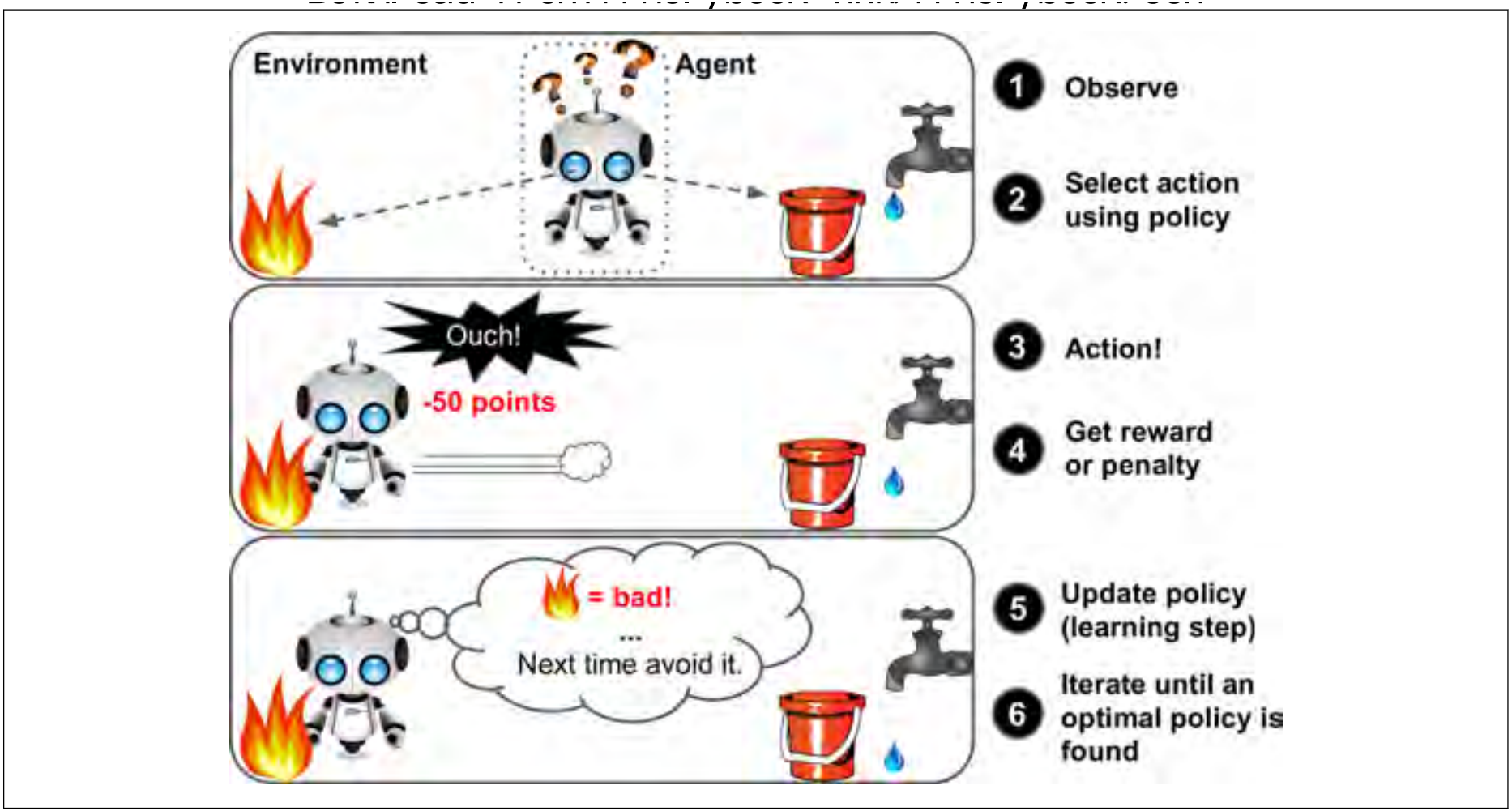
Text and images from: Géron, Aurélien. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow* O’Reilly Media, Inc.

Reinforcement Learning

- **“The learning system, called an agent in this context, can observe the environment, select and perform actions, and get rewards in return (or penalties in the form of negative rewards).”**
- **“It must then learn by itself what is the best strategy, called a policy, to get the most reward over time. A policy defines what action the agent should choose when it is in a given situation.”**

Géron, Aurélien. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow* O’Reilly Media, Inc.

Reinforcement Learning



Géron, Aurélien. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow* O'Reilly Media, Inc.

Reinforcement Learning

- **“Instead of training examples that indicate the correct output for a given input, the training data in reinforcement learning are assumed to provide only an indication as to whether an action is correct or not; if an action is incorrect, there remains the problem of finding the correct action.”**

Jordan, Michael I. and Tom M. Mitchell. (2015). “Machine Learning: Trends, perspectives, and prospects” *Science*.

Semi-supervised Learning

- **“makes use of unlabeled data to augment labeled data in a supervised learning context, and discriminative training blends architectures developed for unsupervised learning with optimization formulations that make use of labels”**

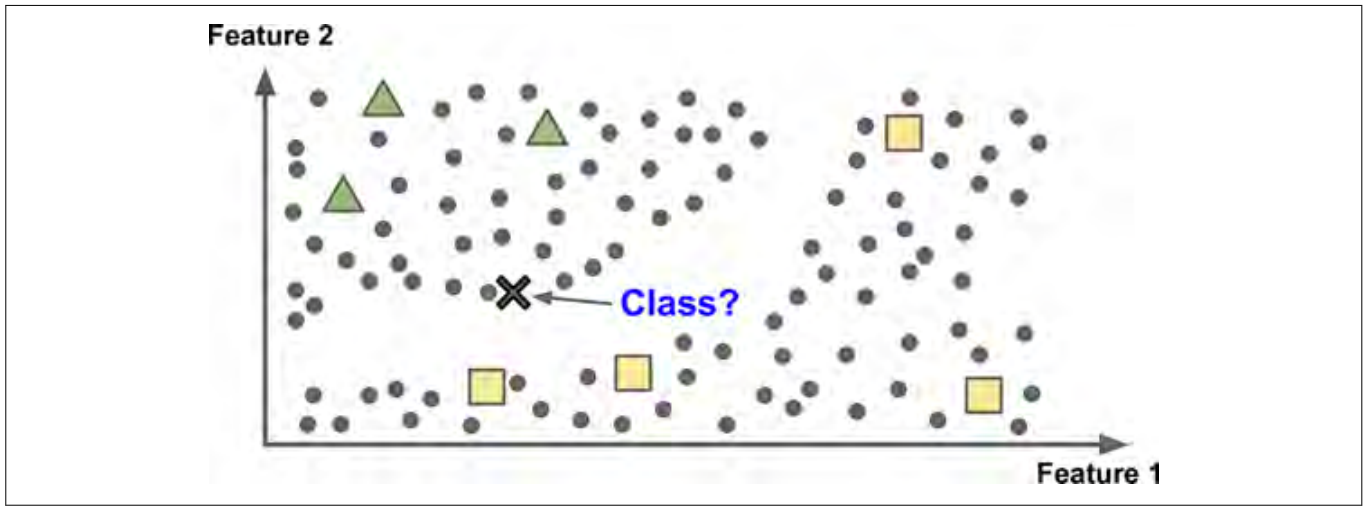


Figure 1-11. Semisupervised learning

Jordan, Michael I. and Tom M. Mitchell. (2015). “Machine Learning: Trends, perspectives, and prospects” *Science*.
Image from: Géron, Aurélien. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow* O’Reilly Media, Inc.

Review Questions

Question Set 1

Géron (p. 4–9)

- **How would you define Machine Learning?**
- **Can you name four types of problems where it shines?**
- **What is a labeled training set?**

Question Set 2

Géron (p. 22–30)

- **Can you name four of the main challenges in Machine Learning?**
- **If your model performs great on the training data but generalizes poorly to new instances, what is happening? Can you name three possible solutions?**
- **What is a test set and why would you want to use it?**
- **What is the purpose of a validation set?**
- **What can go wrong if you tune hyperparameters using the test set?**
- **What is cross-validation and why would you prefer it to a validation set?**

Reading for next class

- Ch 1: “The Machine Learning Landscape” in Géron, Aurélien. (2019). Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow’ O’Reilly Media, Inc. 3–31.
- Ch 1: “Introduction” in Guido, Sarah and Andreas C. Muller. (2016). Introduction to Machine Learning with Python, O’Reilly Media, Inc. 1–25.

DataCamp for next class

- *Introduction to Python (If Needed)*
- *Intermediate Python (If Needed)*
- Data Manipulation with pandas (Required)