Hands-on Machine Learning

1. Introduction

1.

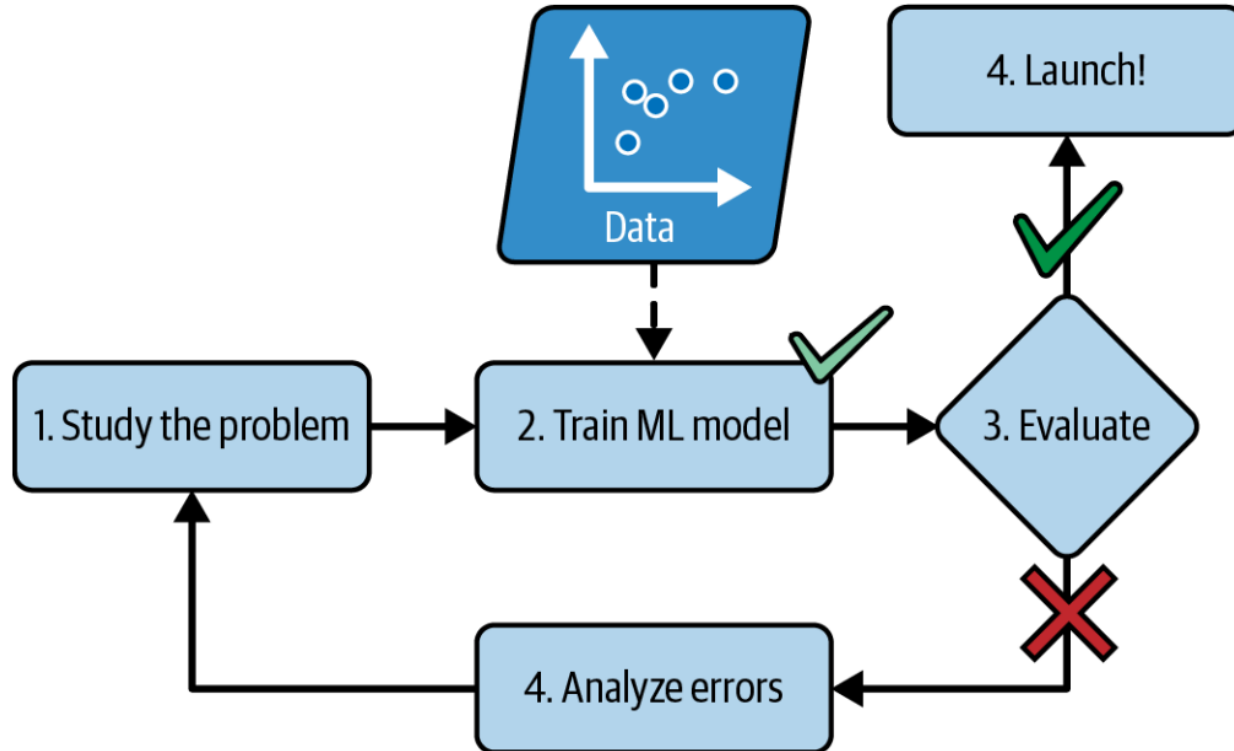# What is Machine Learning?

What is Machine Learning?

➢ Machine Learning is the science (and art) of programming computers so they can *learn from data*.

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

➢ Study of algorithms that improve their performance (P) at some task (T) with experience (E).
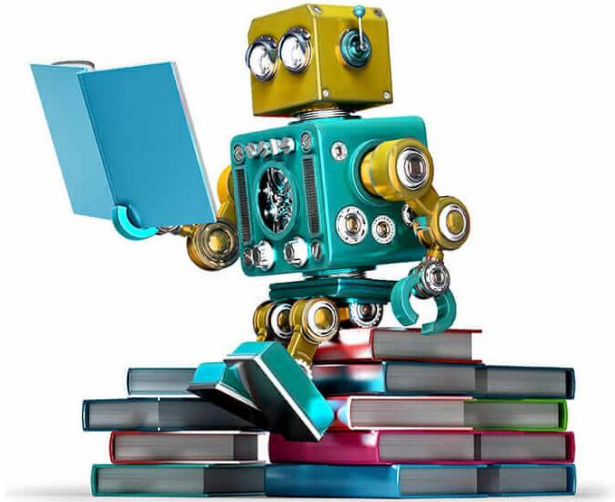
The Traditional Approach
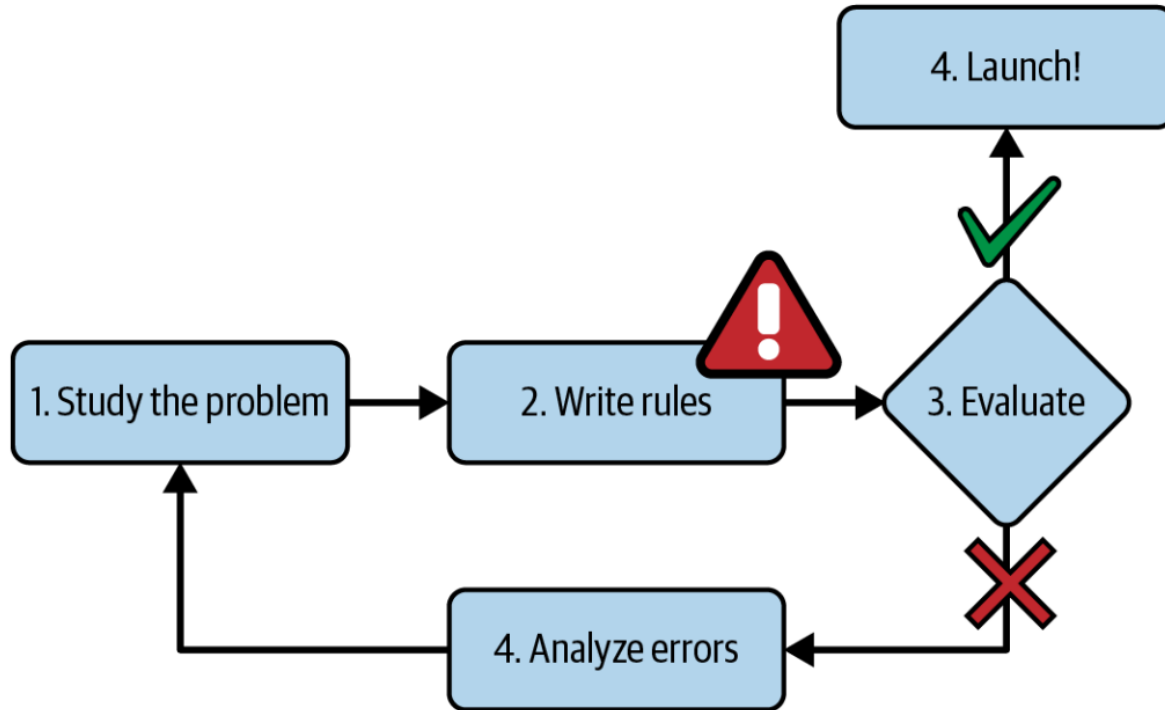
The Machine Learning Approach

5

When to use machine learning?

➢ Problems for which existing solutions require a lot of fine-tuning or long lists of rules.

  ➢ ML can simplify code and perform better than the traditional approach.

➢ Complex problems for which using a traditional approach yields no good solution.

  ➢ The best ML techniques can perhaps find a solution.

➢ Fluctuating environments.

  ➢ ML systems can adapt to new data.

➢ Getting insights about complex problems and large amounts of data.

Automatically adapting to change
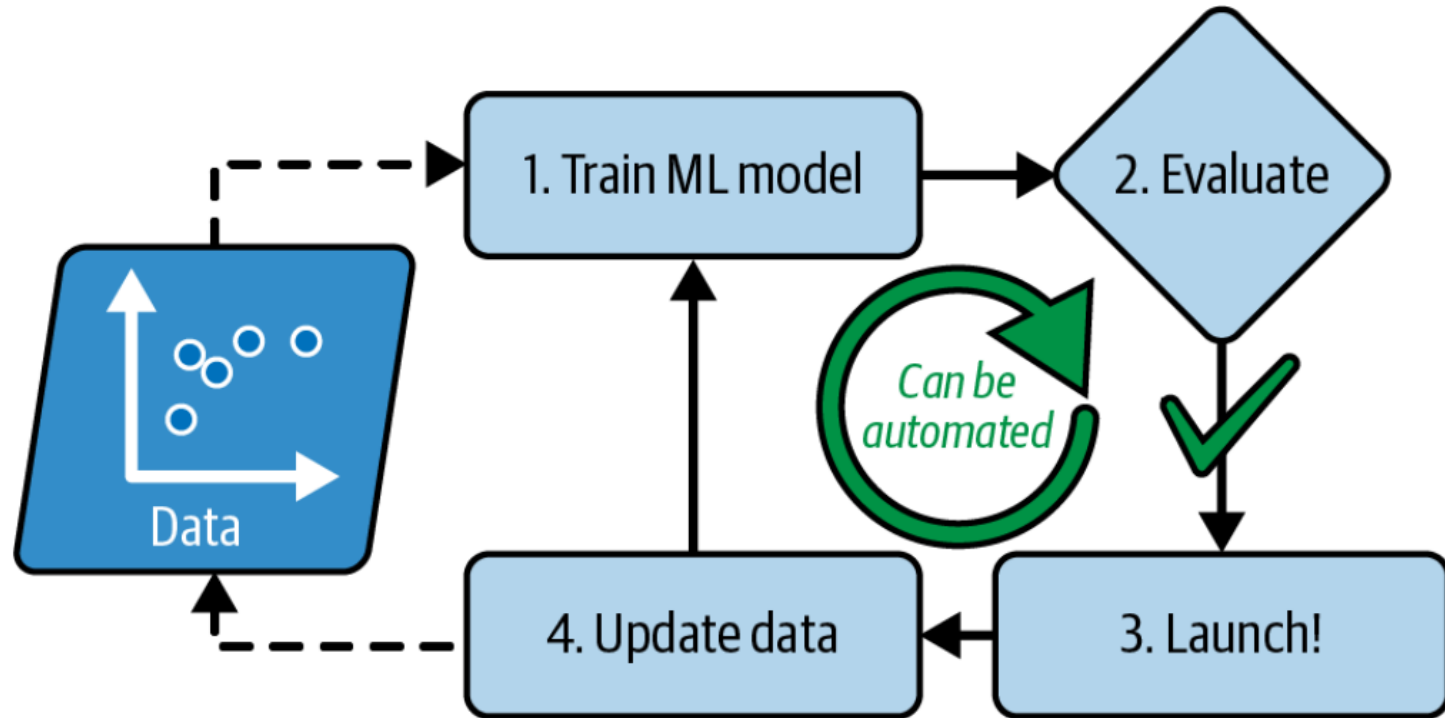
Machine learning can help humans learn
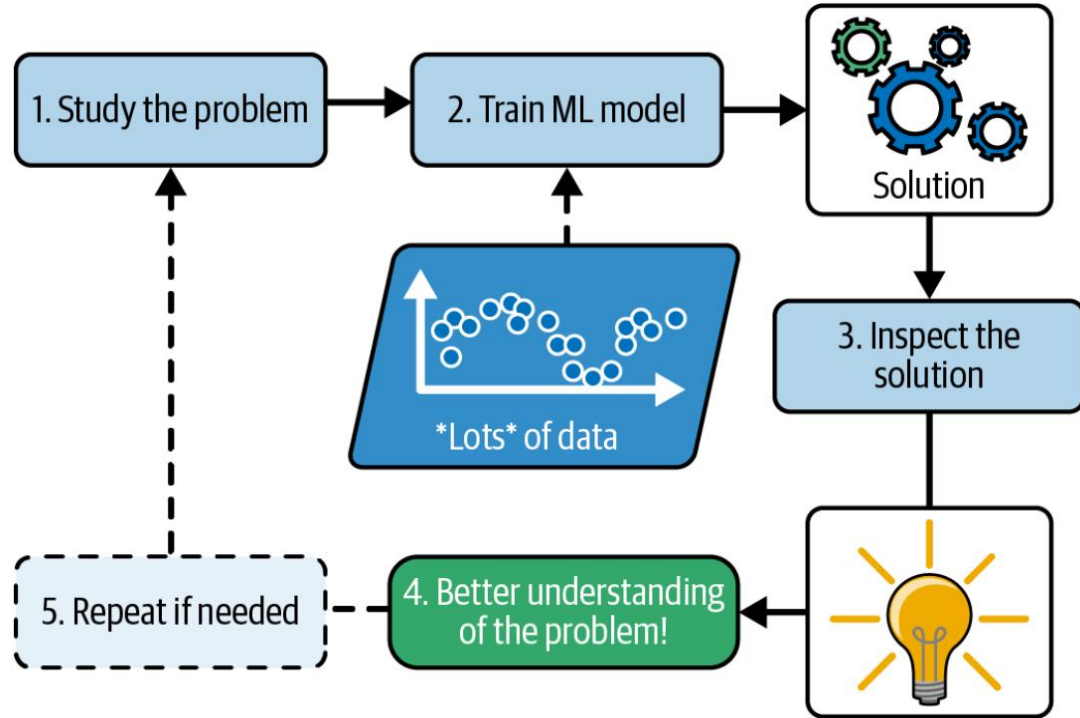
2.
Types of Machine Learning

Types of Machine Learning Systems

➢ Whether or not they are trained with human supervision:

   ➢ **Supervised** learning: data has **known** labels or output

   ➢ **Unsupervised** learning: labels or output **unknown**

   ➢ **Semi-supervised** learning: labels or output known for a **subset** of data

   ➢ **Self-supervised** learning: labels are **generated** from an unlabeled dataset

   ➢ **Reinforcement** learning: **making decisions** based on previous experience

Types of Machine Learning Systems

➢ Whether or not they can learn incrementally on the fly

   ➢ **Online** learning: the system is trained incrementally

   ➢ **Batch** learning: incapable of learning incrementally

Types of Machine Learning Systems

➢ Whether they work by simply comparing new data points to known data points, or by detecting patterns in the training data and building a predictive model:

> ➢ **Instance-based** learning

> ➢ **Model-based** learning

Supervised Learning

➢ In *supervised learning*, the training set you feed to the algorithm includes the desired solutions, called *labels*.

Training set
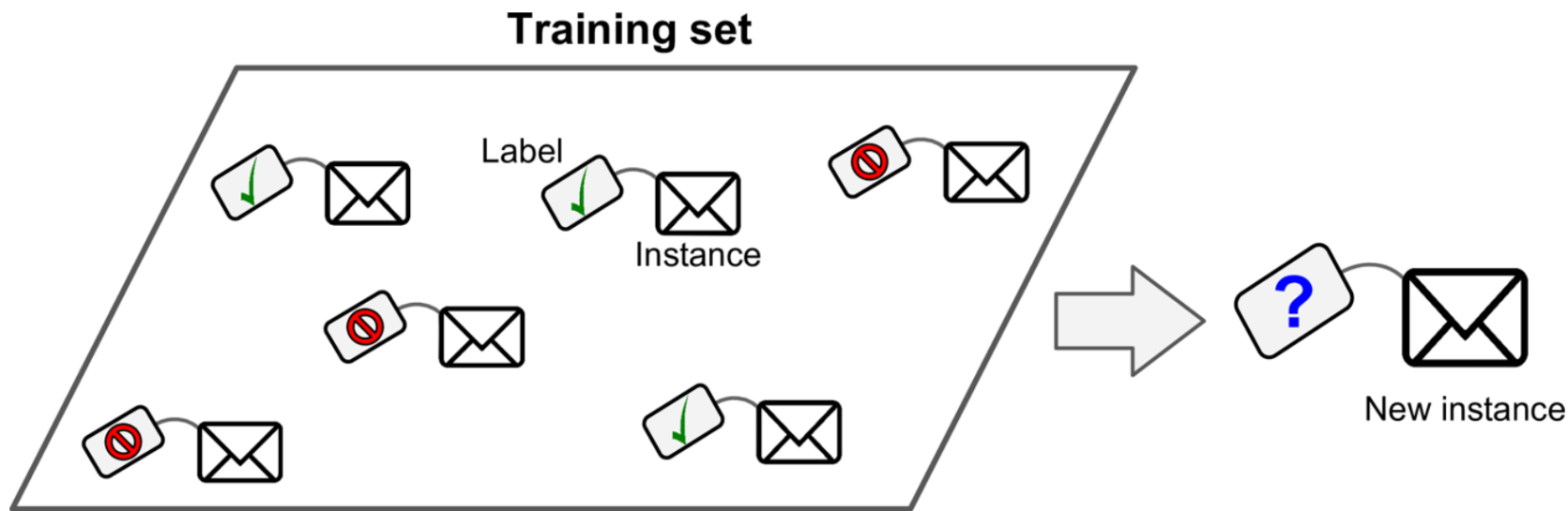
Typical Supervised Learning Tasks

➢ *Classification*: The system should be trained with many instances (e.g. emails) along with their *class* (e.g. spam or ham), and it must learn how to classify new instances.

➢ *Regression*: The system should predict a *target* numeric value, such as the price of a car, given a set of *features* (mileage, age, brand, etc.) called *predictors*.
  ➢ To train the system, you need to give it many examples of cars, including both their predictors and their labels (i.e., their prices).

Classification vs. Regression
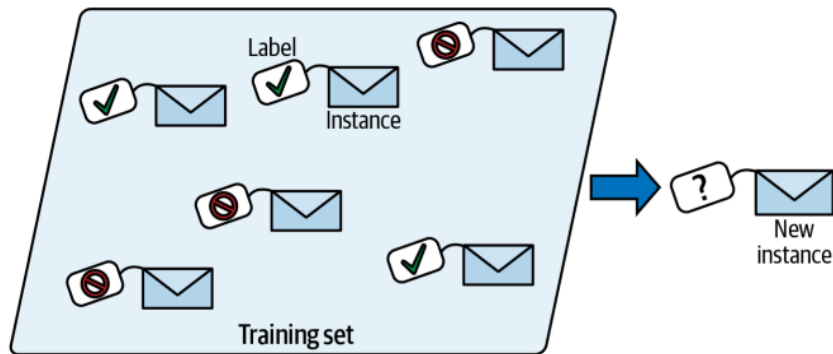
Classification
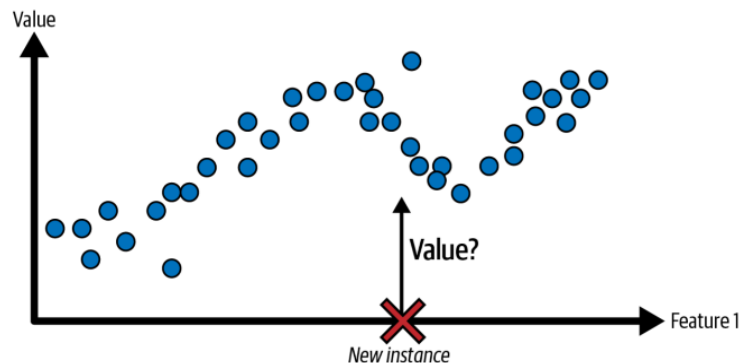
Regression

Label Bias

➢ The performance of a supervised ML model depends heavily on the quality and quantity of the labeled data it's trained on.

➢ Data biases are not special to data collection or sampling, we may have labeling bias.

User reviews 5.1K ›	+ Review

FEATURED REVIEW ★10/10

A Masterpiece

This must rank as the best film (along with part 2)of all time.An ensemble performance that has no weak spot.

Particularly, John Cazale (Fredo) and Richard Castellano (Clemenza) give wonderfully understated performances. You just have to believe that Castellano WAS Clemenza, he brings a real touch to his role.

John Cazale brings the troubled Fredo to life, and you can see the weak Fredo desperately trying to live up to the family reputation but knowing that he can never be what his father wants.

The story of one man's reluctance to be drawn into the murky family business,and his gradual change through circumstance, paints a vivid picture of this violent period of US history.

Do not miss this film!

👍 helpful · 322 👎 132

The-Other-Monkey · Aug 6, 2006

Christopher L. Webber

★☆☆☆☆ **A Bad Book**
Reviewed in the United States on October 12, 2015
Verified Purchase

SAPIENS by Y. N. Harari

I'm afraid I got off to a bad start with this book when I found the author writing carelessly and falsely early on. Perhaps he was so focused on his big theme that he didn't want to sweat the details, but when an author is wrong on matters you know about, it's hard to trust him on things you don't know about.
· on page 38 he writes, "Albert Einstein was far less dextrous with his hands than was an ancient hunter-gatherer." But Einstein was a violinist and I doubt the average hunter-gatherer could play the violin.
· on page 55 he writes about "witch-burning Puritans," but in fact no witches were burned by the Puritans. People were hanged or pressed to death - which is certainly not nice - but they were not burned.
· on page 80 he tells us that in the Great Plains where there was no wheat 10,000 years ago, you can walk today "for hundreds upon hundreds of miles without encountering any other plant." Well, that's nonsense. Ask any wheat farmer about weeds. But apart from weeds, there are also corn and soy and a variety of other crops. I've been there but I doubt Harari has.

Unsupervised Learning

➢ In *unsupervised learning*, the training data is unlabeled.

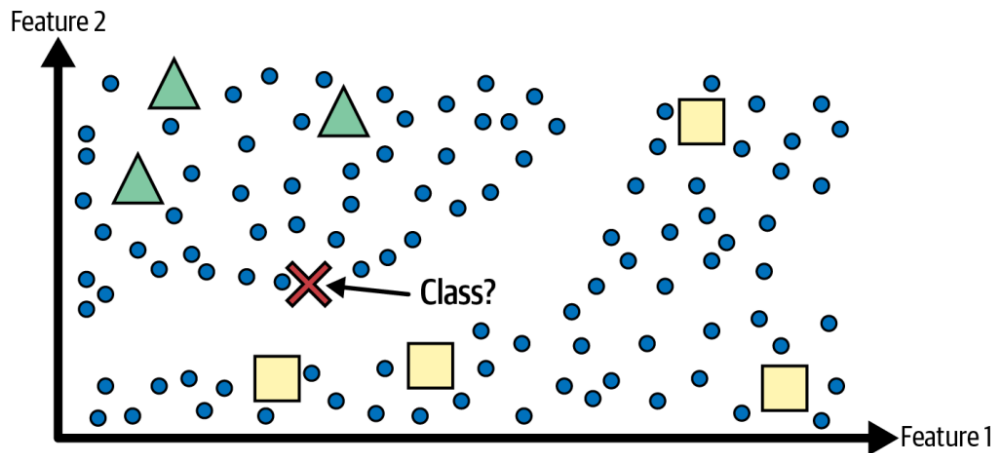   ➢ The system tries to learn without a teacher.
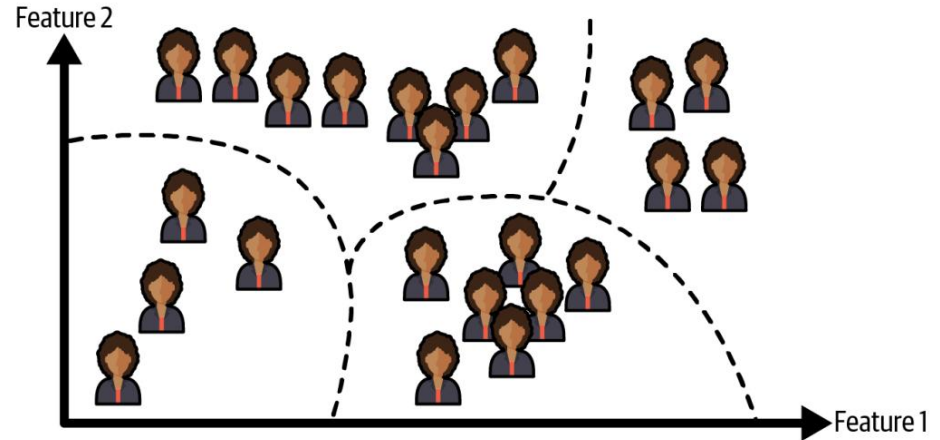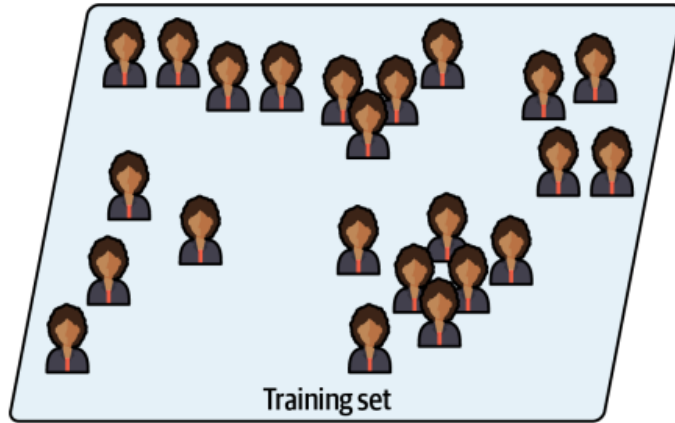
Example of Unsupervised Learning Task

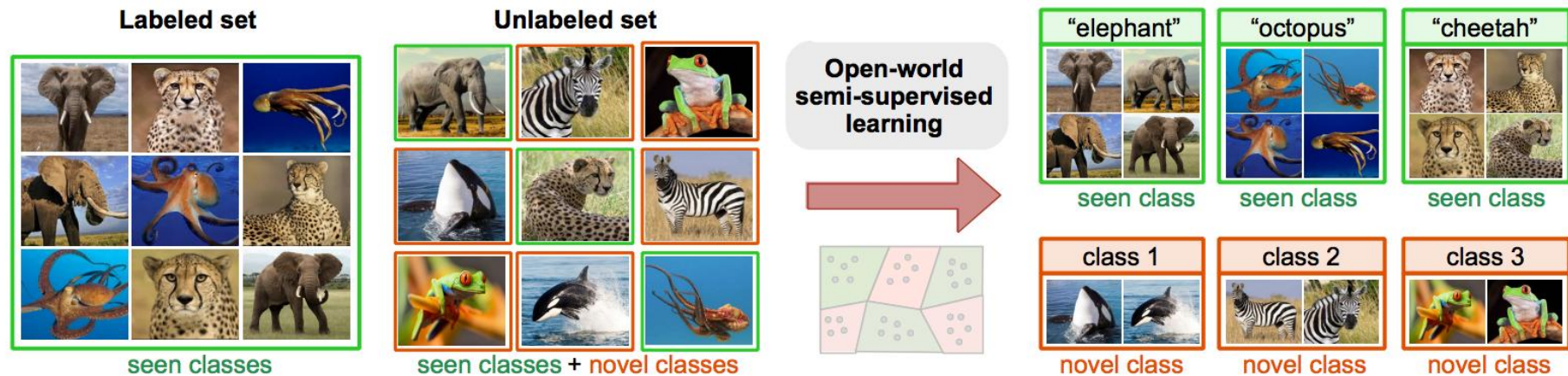

Anomaly Detection
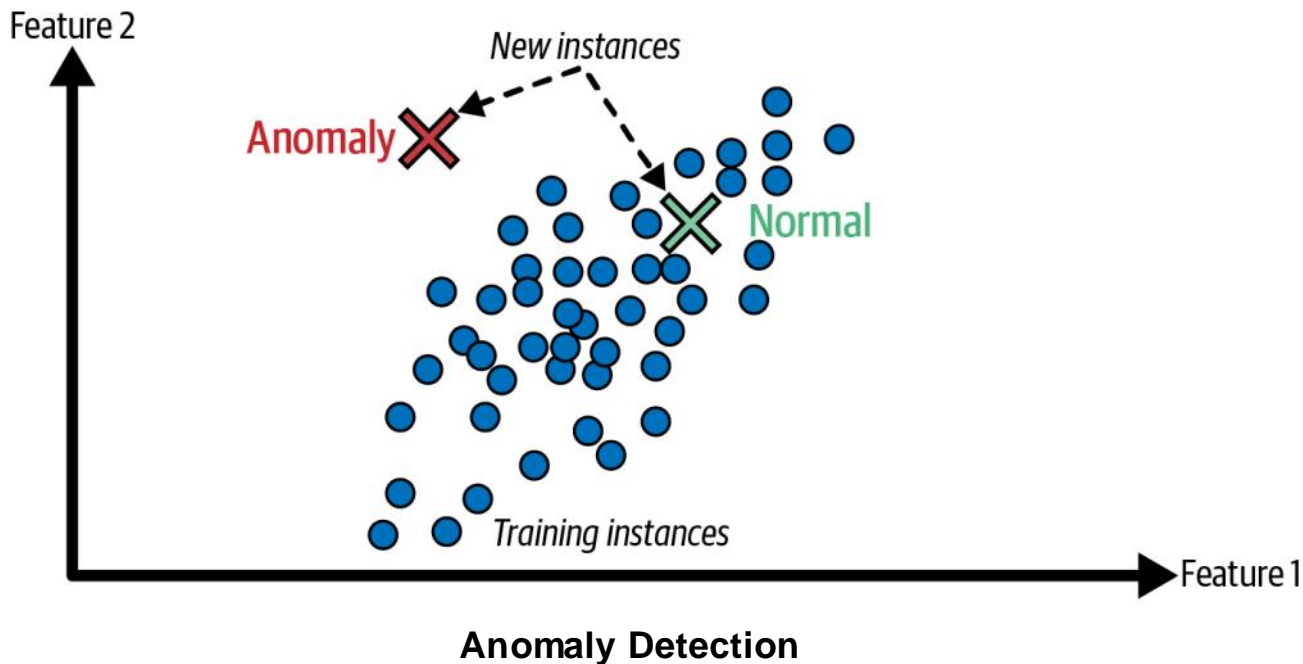
Semi-supervised Learning

➢ Labeling data is time-consuming and costly:

 ➢ We usually have plenty of unlabeled instances, and few labeled instances.

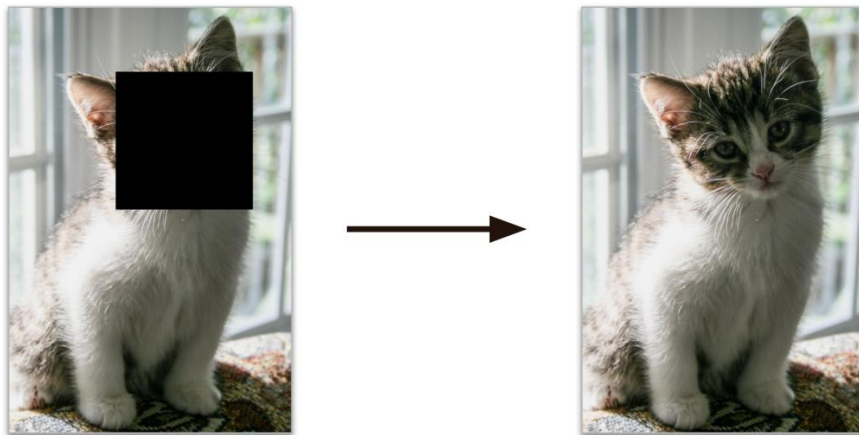➢ *Semi-supervised learning* can deal with data that's partially labeled.

Example of Semi-supervised Learning

Self-supervised Learning

➢ *Self-supervised learning:*
  ➢ Generate a fully labeled dataset from a fully unlabeled one
  ➢ Once the whole dataset is labeled, use a supervised learning algorithm
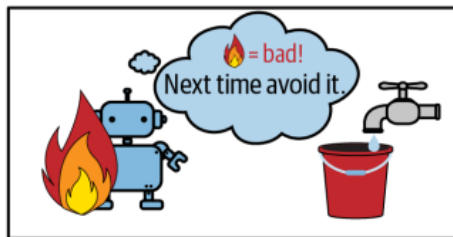
Reinforcement Learning

➢ In *reinforcement learning,* the system (called an *agent*), can observe the environment, select and perform actions, and get *rewards* in return.

➢ 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.
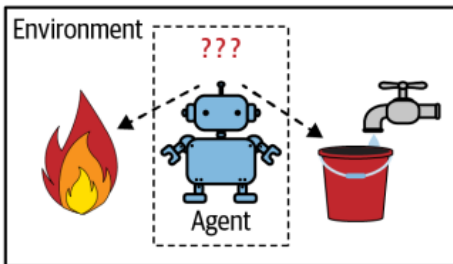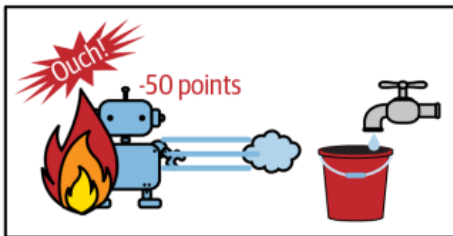
Examples of Reinforcement Learning

Environment Bias: Self-Driving Cars

Environment Bias: Example

Phoenix, Arizona

Bay Area, California

# Environment Bias: Example

Batch Learning

➢ In *batch learning*, the system is incapable of learning incrementally

  ➢ It must be trained using all the available data.

  ➢ First the system is trained, and then it is launched into production and runs without learning anymore.

  ➢ This is also called *offline learning*.

➢ If you want a batch learning system to know about new data, you need to train a new version of the system from scratch.

➢ Not suitable for

  ➢ a system that needs to adapt to rapidly changing data (e.g., stock prices)

  ➢ when the amount of training data is huge

  ➢ a system that needs to be able to learn autonomously and it has limited resources (e.g., a smartphone application or a rover on Mars)
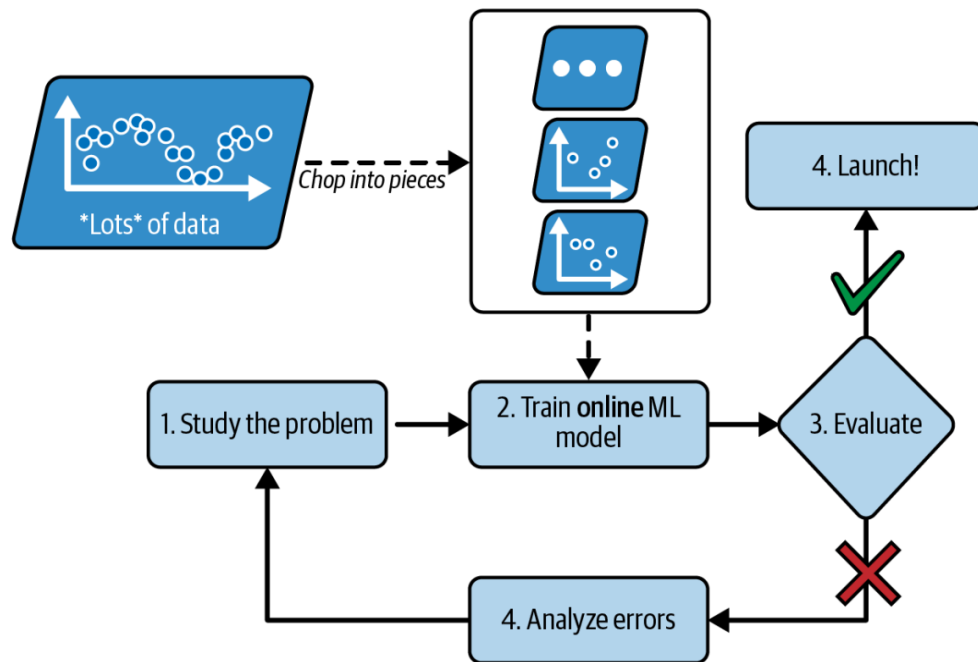
Online Learning

➢ In *online learning*, you train the system incrementally by feeding it data instances sequentially, either individually or in small groups called *mini-batches*.
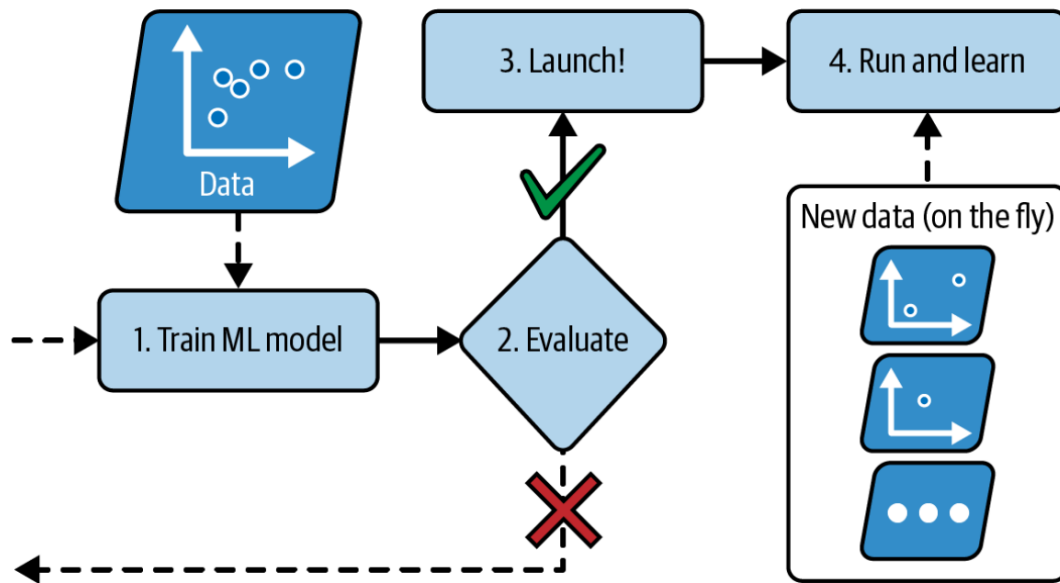
Out-of-Core Learning

➢ *Out-of-core* learning: using online learning to train systems on huge datasets that cannot fit in one machine's main memory.

Learning Rate

➢ *Learning rate*: a parameter that shows how fast an online learning system should adapt to changing data.

➢ High learning rate: the system will rapidly adapt to new data, but it will also tend to quickly forget the old data.

➢ Low learning rate: the system will learn more slowly, but it will also be less sensitive to noise in the new data or to sequences of non-representative data points (outliers).

Bad Data in Online Learning

➢ Challenge with online learning: if bad data is fed to the system, the system's performance will gradually decline.

    ➢ malfunctioning sensor

    ➢ someone spamming the system

    ➢ lousy human operator

➢ To reduce this risk, you need to monitor your system's performance closely and promptly switch learning off and possibly revert to a previously working state.

➢ You may also want to monitor the input data and react to abnormal data.

Dynamic Pricing of Amazon Sellers

Instance-based Learning

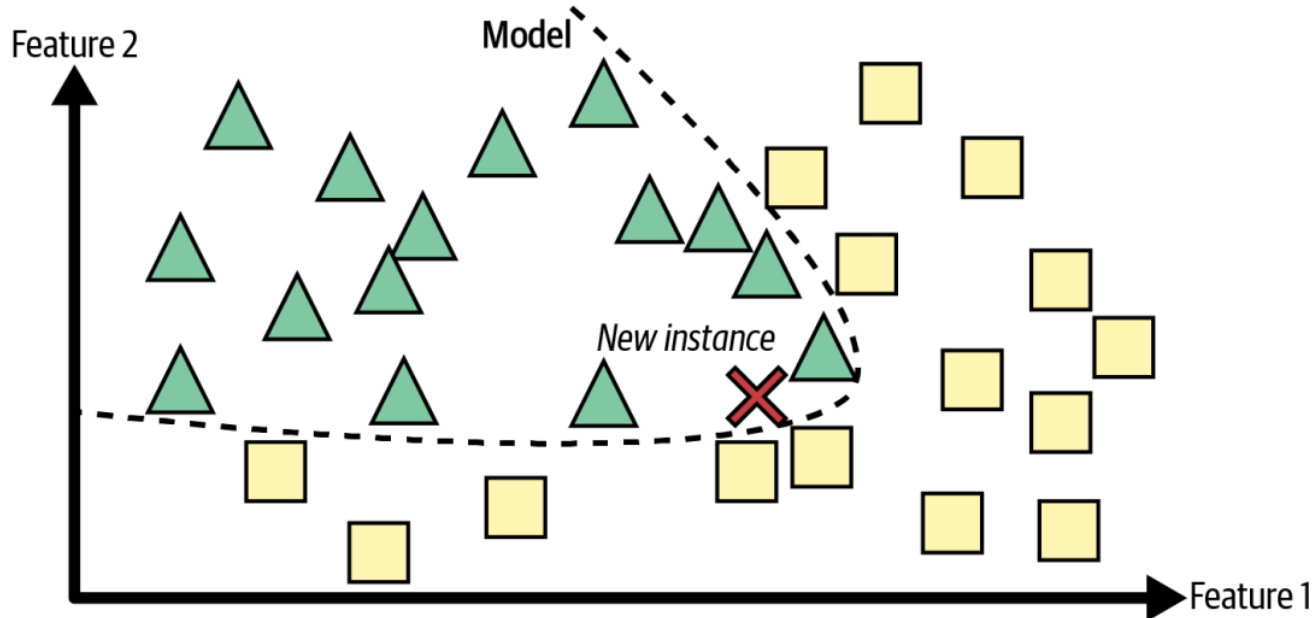➢ *Instance-based learning*: the system learns the examples by heart, then generalizes to new cases by using a similarity measure to compare them to the learned examples (or a subset of them).
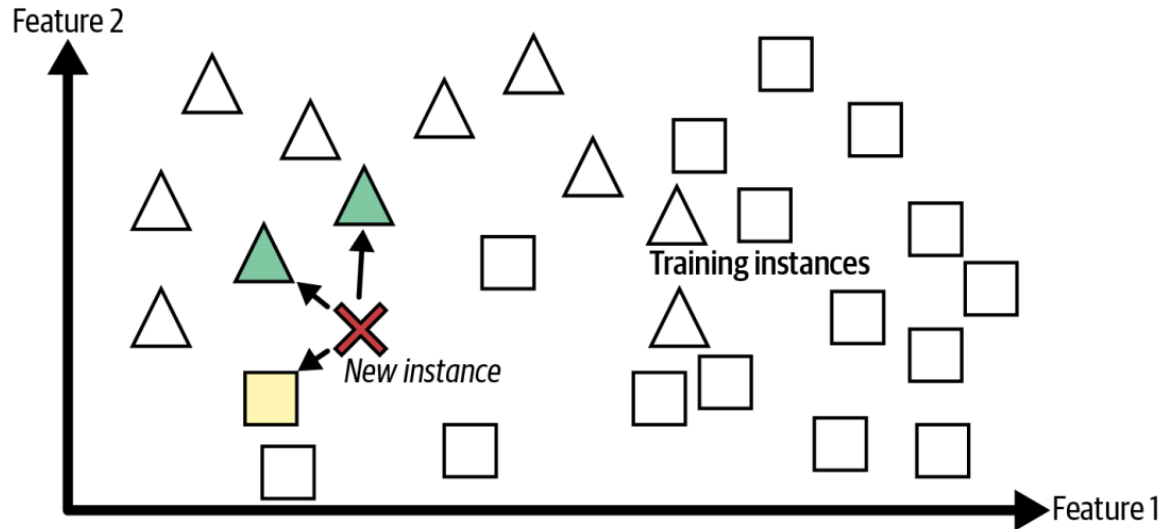
Model-based Learning

➢ Model-based learning: to build a model of the examples and then use that model to make predictions.

Example of Model-based Learning

➢ We want to know if money makes people happy.
➢ We have dataset of countries GDP and Better Life Index.

Country	GDP per capita (USD)	Life satisfaction
Hungary	12,240	4.9
Korea	27,195	5.8
France	37,675	6.5
Australia	50,962	7.3
United States	55,805	7.2
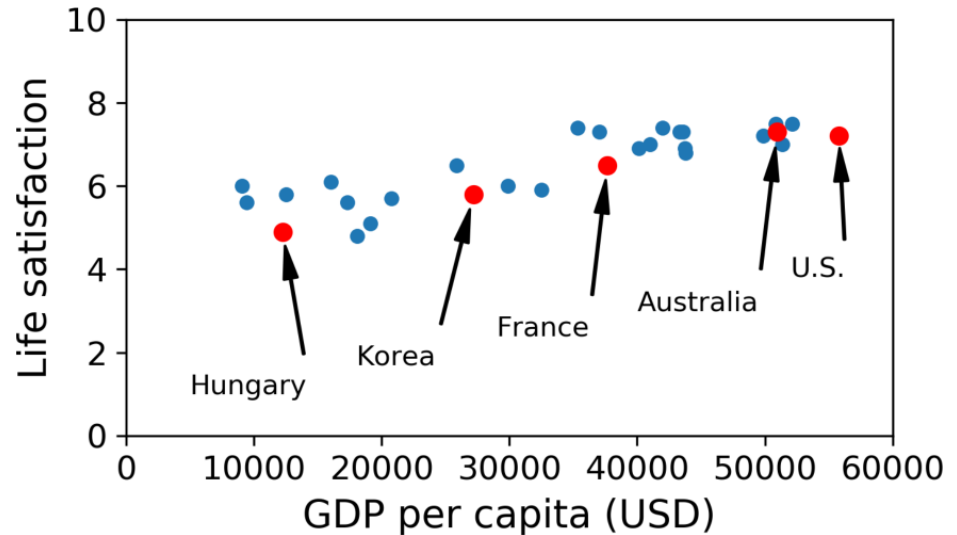
Example of Model-based Learning

➢ We decide to model life satisfaction as a linear function of GDP per capita.

    ➢ This step is called *model selection*:

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

➢ This model has two *model parameters*, θ_0 and θ_1.

➢ How can you know which values for θ_0 and θ_1 will make your model perform best?
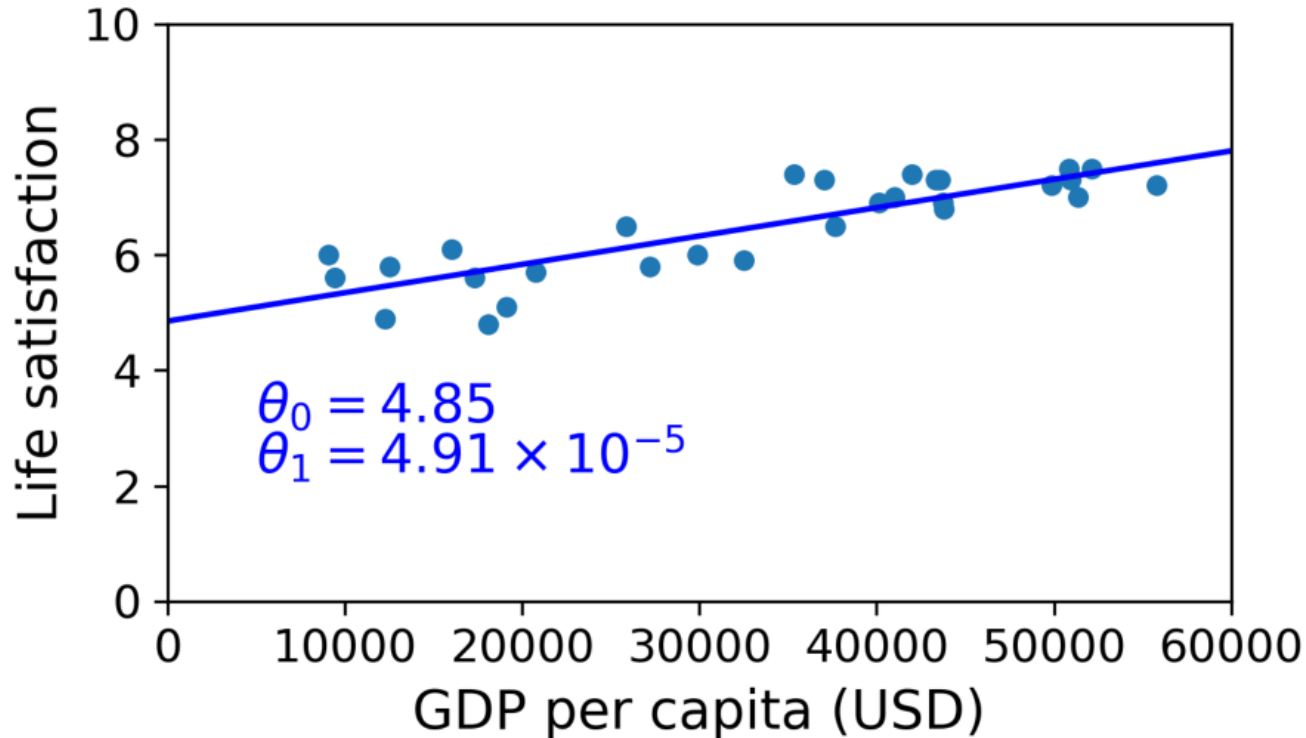
Example of Model-based Learning
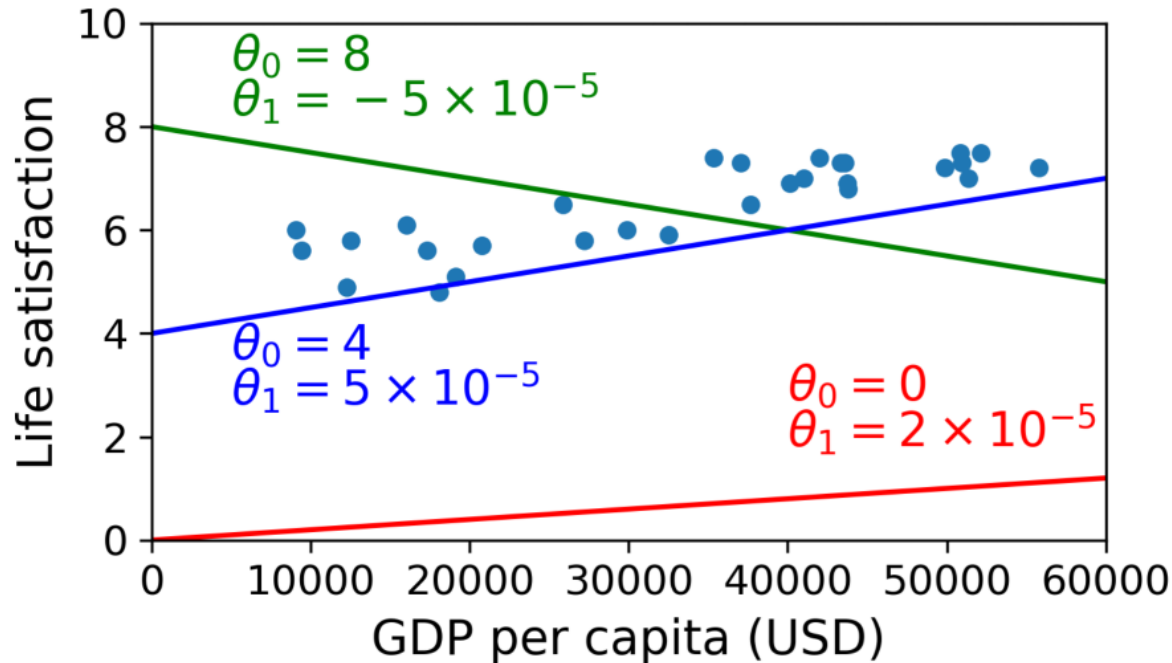
Model-based Learning

➢ We need to specify a performance measure.

➢ A *utility function* (or *fitness function*) that measures how *good* your model is, or you can define a *cost function* that measures how *bad* it is.

  ➢ In linear regression, we use a cost function that measures the distance between the linear model's predictions and the training examples.

  ➢ The objective is to minimize this distance.

➢ Linear Regression algorithm is fed with training examples, and it finds the parameters that make the linear model fit best to the data.

  ➢ This is called *training* the model.

Example of Model-based Learning

Example of Model-based Learning

```python
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
from sklearn.linear_model import LinearRegression

# Download and prepare the data
data_root = "https://github.com/ageron/data/raw/main/"
lifesat = pd.read_csv(data_root + "lifesat/lifesat.csv")
X = lifesat[["GDP per capita (USD)"]].values
y = lifesat[["Life satisfaction"]].values

# Visualize the data
lifesat.plot(kind='scatter', grid=True,
             x="GDP per capita (USD)", y="Life satisfaction")
plt.axis([23_500, 62_500, 4, 9])
plt.show()
```

```python
# Select a linear model
model = LinearRegression()

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

# Make a prediction for Cyprus
X_new = [[37_655.2]]  # Cyprus' GDP per capita in 2020
print(model.predict(X_new)) # output: [[6.30165767]]
```

Typical Machine Learning Project

➢ Study the data.

➢ Select a model.

➢ Train the model using the training data.

    ➢ The learning algorithm searches for the model parameter values that minimize a cost function.

➢ Apply the model to make predictions on new cases.

    ➢ This is called *inference*.