What is Machine Learning

He He[1]

CDS, NYU

Feb 2, 2021

[1]Slides based on Lecture 1 from David Rosenberg's course material.

Contents

Machine Learning Problems

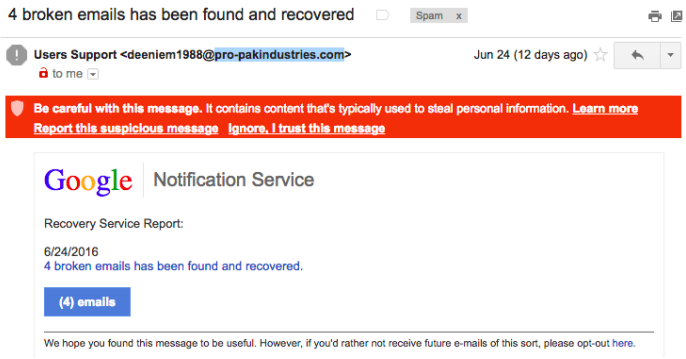
Common theme is to solve a prediction problem:

- given an **input** x,
- **predict** an **output** y.

We'll start with a few canonical examples...

Example: Spam Detection

- **Input:** Incoming email



- **Output:** "SPAM" or "NOT SPAM"
- A **binary classification** problem, because only 2 possible outputs.

Example: Medical Diagnosis

- **Input**: Symptoms (fever, cough, fast breathing, shaking, nausea, ...)
- **Output:** Diagnosis (pneumonia, flu, common cold, bronchitis, ...)
- A **multiclass classification** problem: choosing one of several *discrete* outputs.

How to express uncertainty?

- **Probabilistic classification** or **soft classification**:

$$\begin{aligned} \mathbb{P}(\text{pneumonia}) &= 0.7 \\ \mathbb{P}(\text{flu}) &= 0.2 \\ \vdots &\quad \vdots \end{aligned}$$

Example: Predicting a Stock Price

- **Input:** History of stock's prices
- **Output:** Predict stock's price at close of next day
- A **regression** problem, because the output is *continuous*.

The Prediction Function

- A **prediction function** takes input x and produces an output y.
- We're looking for prediction functions that solve particular problems.
- **Machine learning** helps find the "best" prediction function **automatically with data**
  - What does "best" mean?

What is **not** ML: Rule-Based Approaches

- Consider medical diagnosis.
  1. Consult textbooks and medical doctors (i.e. "experts").
  2. Understand their diagnosis process.
  3. Implement this as an algorithm (a "**rule-based system**")
- Doesn't sound too bad...
- Very popular in the 1980s.

(To be fair, **expert systems** could be much more sophisticated than they sound here. For example, through **inference** they could make new logical deductions from knowledge bases.)

Rule-Based Approach

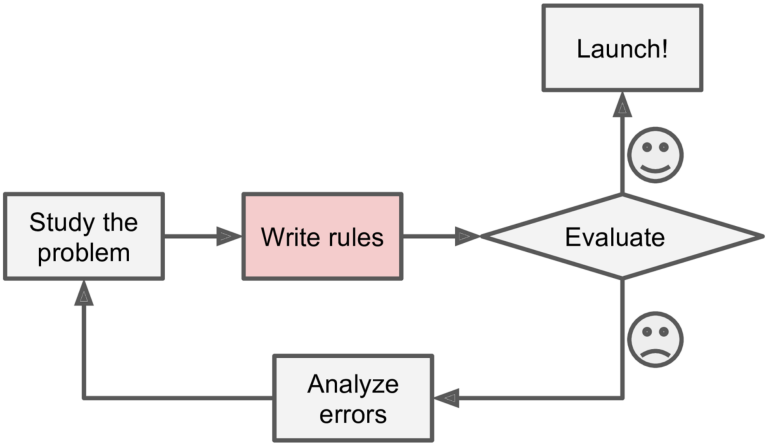


Fig 1-1 from *Hands-On Machine Learning with Scikit-Learn and TensorFlow* by Aurelien Geron (2017).

Rule-Based Systems

Issues with rule-based systems:

- Very labor intensive to build.
- Rules work very well for areas they cover, but **cannot generalize** to unanticipated input combinations.
- Don't naturally handle uncertainty.
- Expert systems seen as brittle

Modern AI: Machine Learning

- Don't reverse engineer an expert's decision process.
- Machine **learns** on its own.
- We provide **training data**: many examples of (input x , output y) pairs, e.g.
  - A set of videos, and whether or not each has a cat.
  - A set of emails, and whether or not each is SPAM.
- Learning from training data of this form is called **supervised learning**.

Machine Learning Algorithm

- A **machine learning algorithm** learns from the training data:
  - **Input**: Training Data
  - **Output**: A prediction function that produces output y given input x.
- The success of ML depends on
  - Availability of large amounts of data
  - **Generalization** to unseen samples (the test set)

Machine Learning Approach

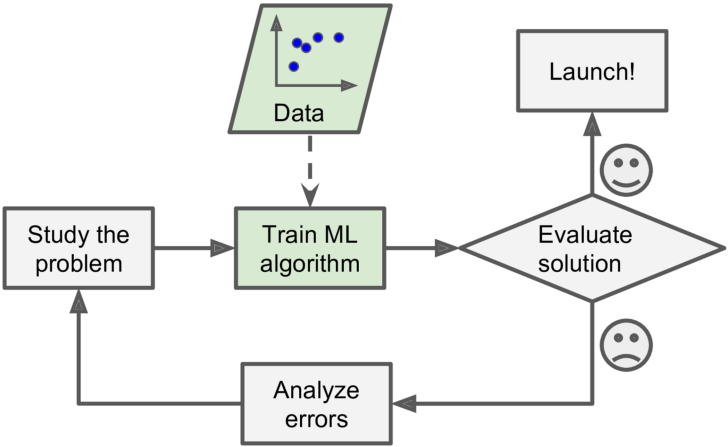


Fig 1-2 from *Hands-On Machine Learning with Scikit-Learn and TensorFlow* by Aurelien Geron (2017).

Key concepts

- Most common **ML problem types**
  - classification (binary and multiclass)
  - regression
- **prediction function**: predicts output y given input x
- **training data**: a set of (input x, output y) pairs
- **supervised learning algorithm**: takes training data and produces a prediction function
- Beyond prediction
  - **Unsupervised learning**: finding structures in data, e.g. clustering
  - **Reinforcement learning**: optimizing long-term objective, e.g. Go
  - **Representation learning**: learning good featurs of real-world objects, e.g. text

Core Questions in Machine Learning

Given any task, the following questions need to be answered:

- **Modeling**: What is the prediction function?
- **Learning**: How to learn the prediction function from data?
- **Inference**: Given a learned model, how to make predictions?