CITS5508 Machine Learning
Introduction to Machine Learning

Débora Corrêa (Unit Coordinator and Lecturer)

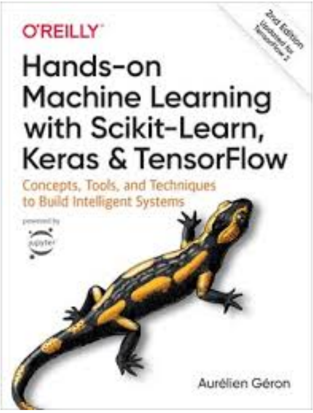
2023

Let's get started!

Now we will roughly follow the text book for the whole unit.

Most of the slide material is from the text book: so can not be circulated and can not be used in your work (without acknowledgement).

Hands-on Machine Learning with Scikit-Learn & TensorFlow


O'REILLY
2nd Edition
Updated for TensorFlow 2
Hands-on Machine Learning with Scikit-Learn, Keras & TensorFlow
Concepts, Tools, and Techniques to Build Intelligent Systems
powered by jupyter
Aurélien Géron

Chapter One

The Machine Learning Landscape

- What is Machine Learning?
- Why use Machine Learning?
- Types of Machine Learning Systems
- Main Challenges of Machine Learning
- Testing and Validating

What is machine learning?

Machine Learning is the science (and art) of programming computers so they can learn from data. Here is a slightly more general definition:

> *[Machine Learning is the] field of study that gives computers the ability to learn without being explicitly programmed. Arthur Samuel, 1959*

And a more engineering-oriented one:

> *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*

Example

For example, your spam filter is a Machine Learning program that can learn to flag spam given examples of spam emails (e.g., flagged by users) and examples of regular (nonspam, also called "ham") emails. The examples that the system uses to learn are called the *training set*. Each training example is called a *training instance* (or *sample*). In this case, the task T is to flag spam for new emails, the experience E is the *training data*, and the performance measure P needs to be defined; for example, you can use the ratio of correctly classified emails. This particular performance measure is called *accuracy* and it is often used in classification tasks.
If you just download a copy of Wikipedia, your computer has a lot more data, but it is not suddenly better at any task. Thus, it is not Machine Learning.

Why use machine learning?

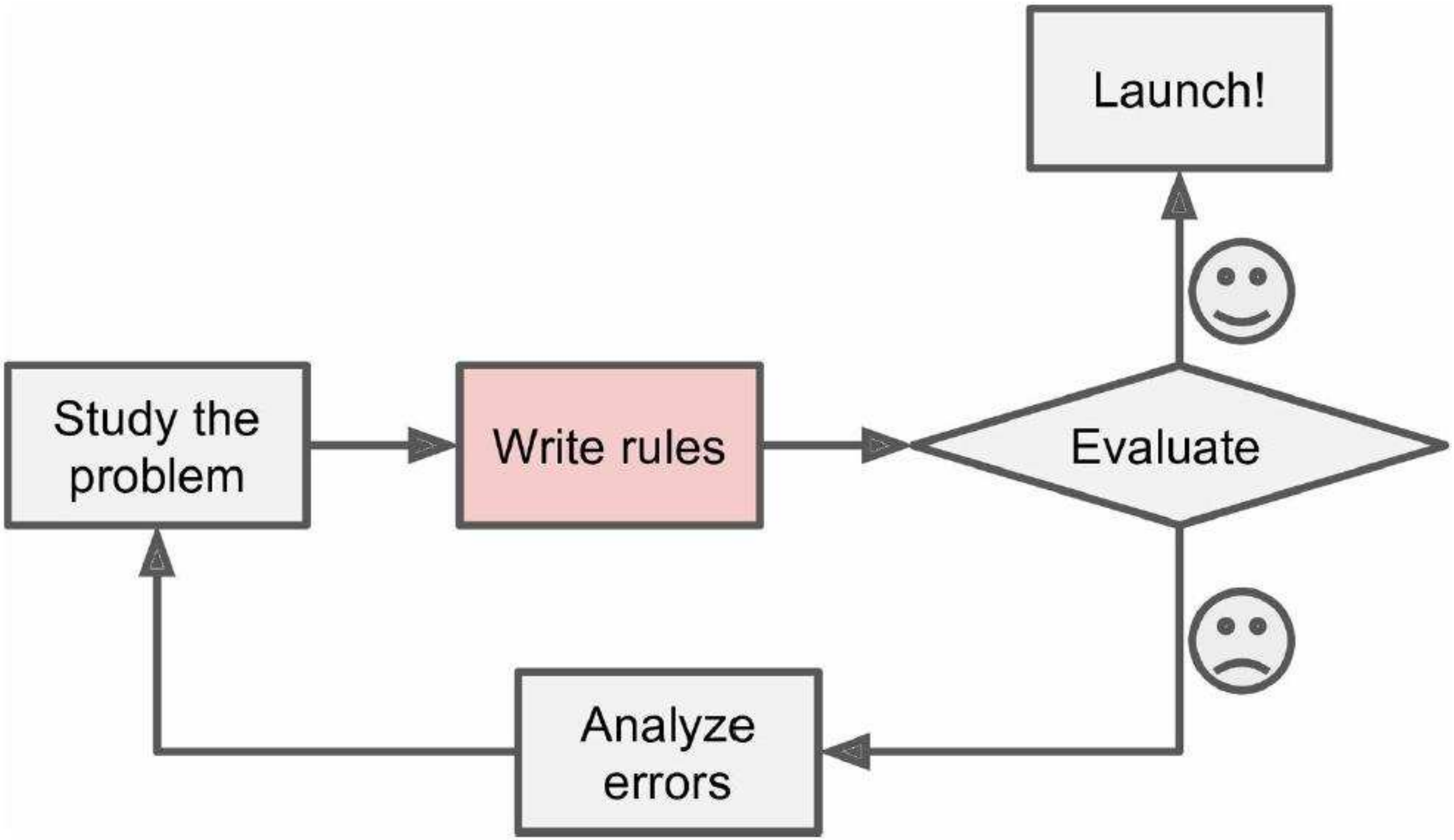


Figure 1-1. The traditional approach

Why use machine learning?

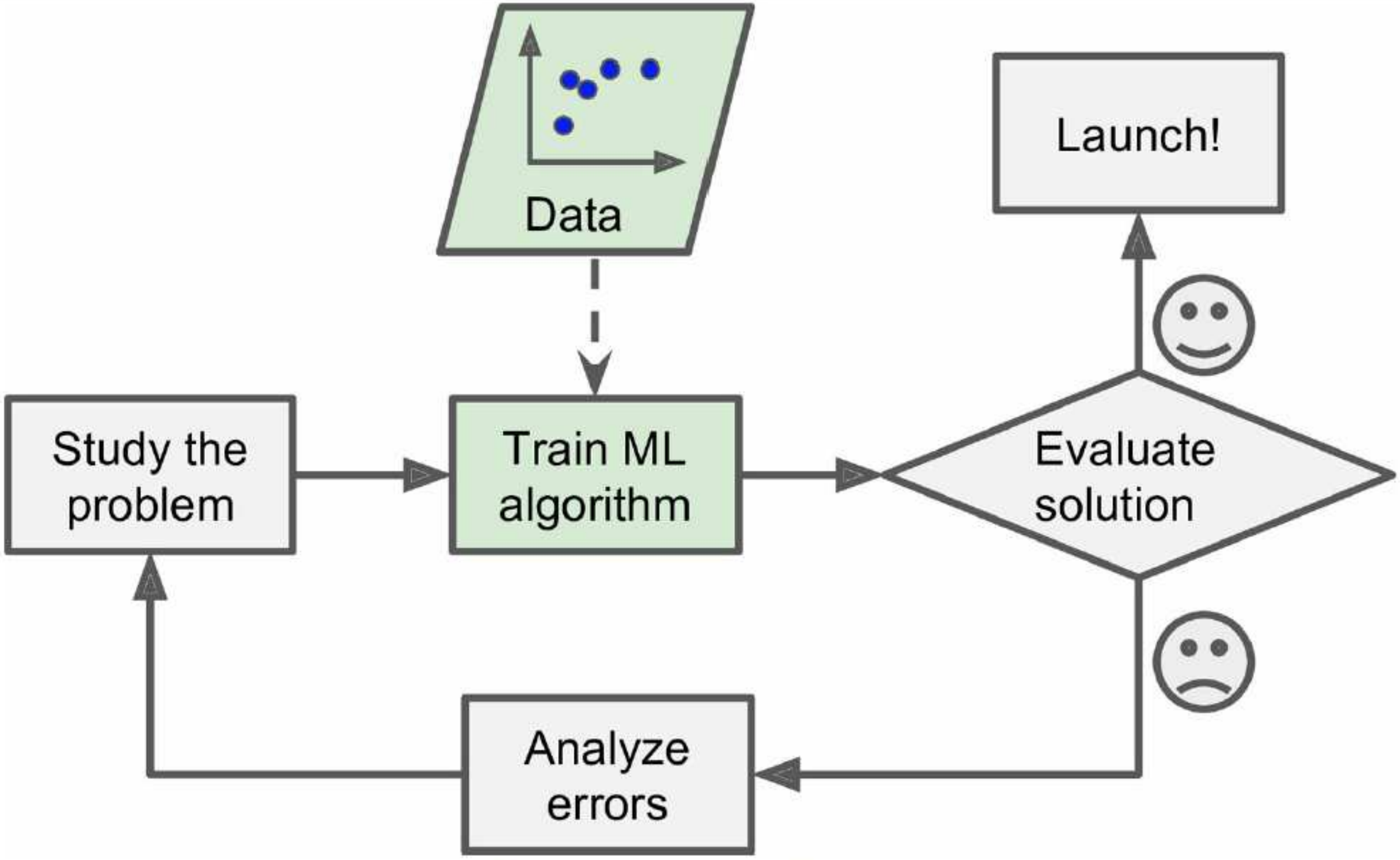


Figure 1-2. Machine Learning approach

Why use machine learning?

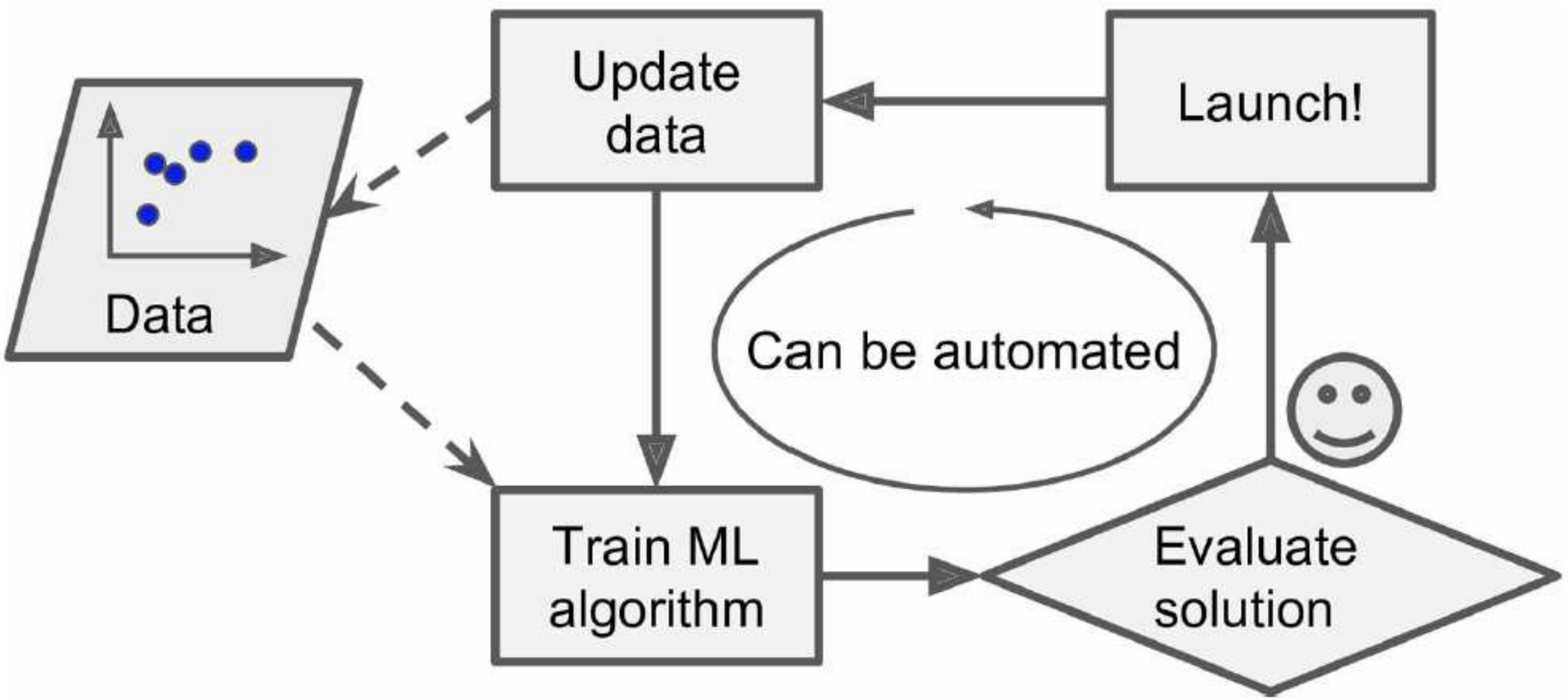


Figure 1-3. Automatically adapting to change

Why use machine learning?

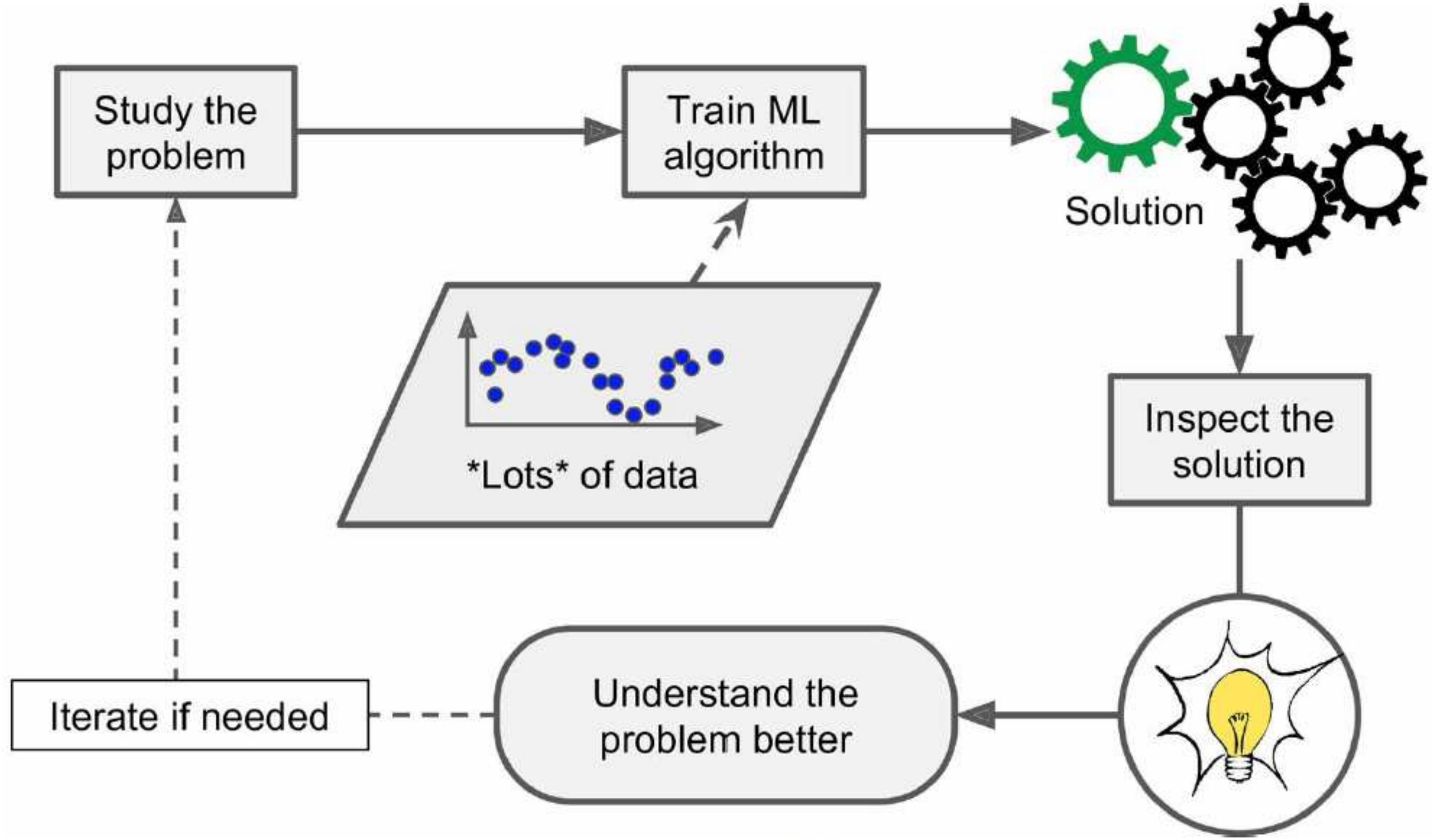


Figure 1-4. Machine Learning can help humans learn

Machine Learning is great for ...

To summarize, Machine Learning is great for:

- Problems for which existing solutions require a lot of hand-tuning or long lists of rules: one Machine Learning algorithm can often simplify code and perform better.
- Complex problems for which there is no good solution at all using a traditional approach: Machine Learning techniques can find a good solution.
- Fluctuating environments: a Machine Learning system can adapt to new data.
- Getting insights about complex problems and large amounts of data.

Types of Machine Learning Systems

Many. Useful to classify them in broad categories based on:

- Whether or not they are trained with human supervision (supervised, unsupervised, semisupervised, and Reinforcement Learning)
- Whether or not they can learn incrementally on the fly (online learning 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, much like scientists do (instance-based learning versus model-based learning)

These criteria are not exclusive; you can combine them in any way you like. For example, a state-of-the-art spam filter may learn on the fly using a deep neural network, trained using examples of spam and ham; this makes it an online, model based, supervised learning system.

Supervised/Unsupervised Learning

Machine Learning systems can be classified according to the amount and type of supervision they get during training.

There are four major categories:

- supervised learning,
- unsupervised learning,
- semisupervised learning, and
- Reinforcement Learning.

Supervised Learning

The training data is *labelled*.

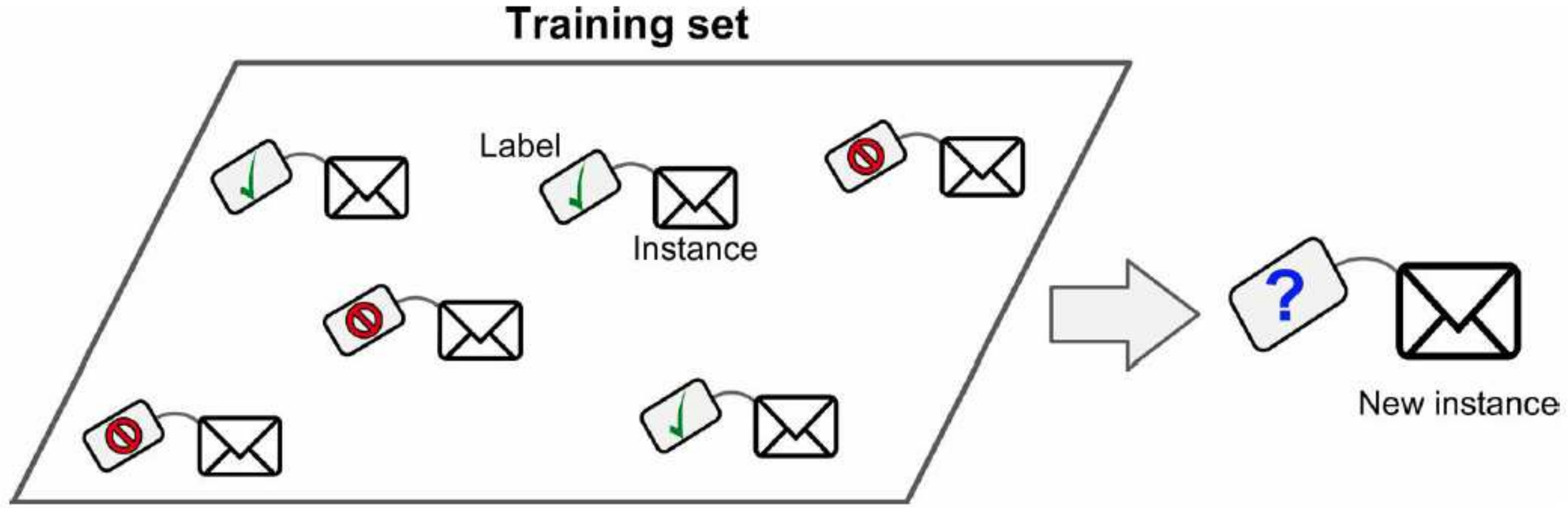


Figure 1-5. A labeled training set for supervised learning (e.g., spam classification)

Supervised Learning: Regression

Instead of a *label*, each training data instance has an associated *target* numeric value.

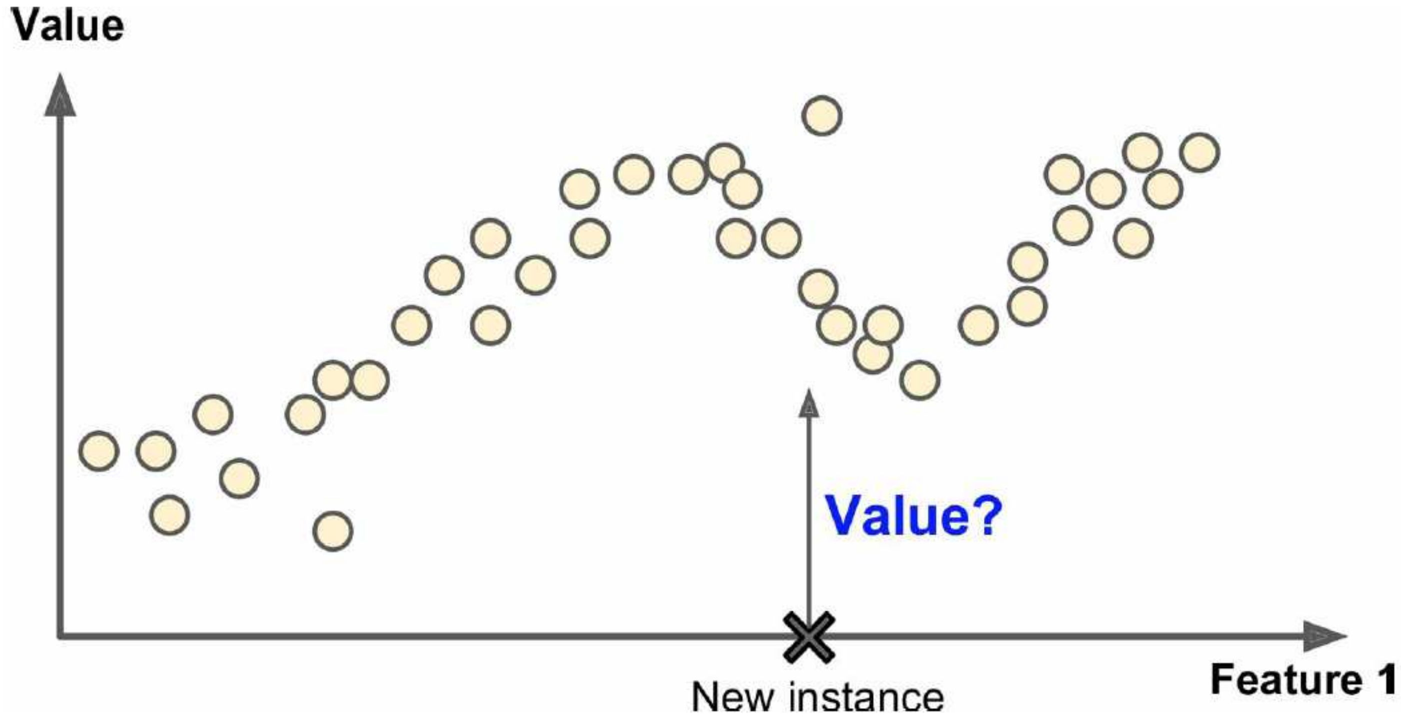


Figure 1-6. Regression

Supervised Learning

Here are some of the most important supervised learning algorithms (covered in the book):

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

Unsupervised Learning

The training data is *unlabelled*.

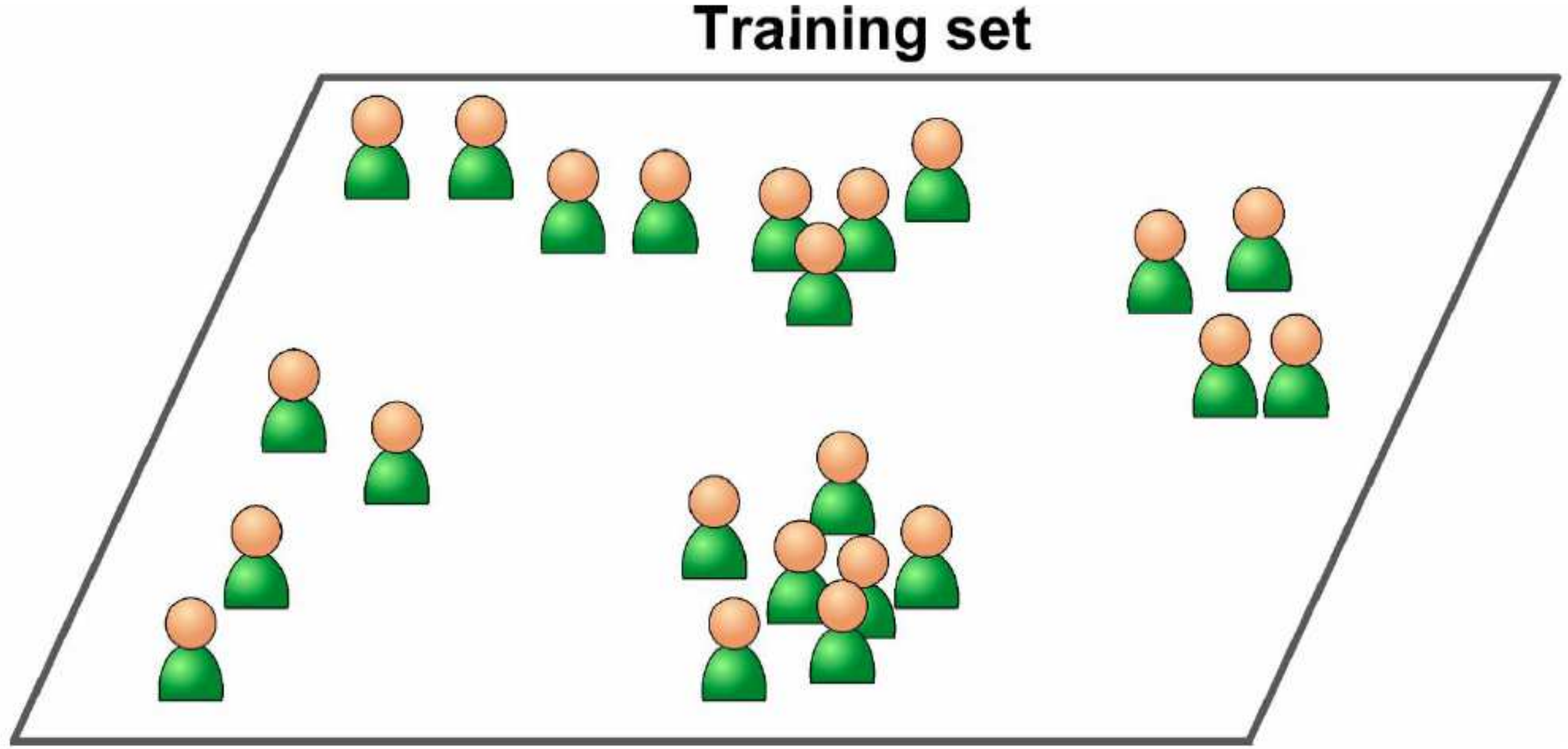


Figure 1-7. An unlabeled training set for unsupervised learning

Unsupervised Learning

Here are some of the most important unsupervised learning algorithms:

- Clustering
  - k-Means
  - DBSCAN
  - Hierarchical Cluster Analysis (HCA)
- Visualization and dimensionality reduction
  - Principal Component Analysis (PCA)
  - Kernel PCA
  - Locally-Linear Embedding (LLE)
  - t-distributed Stochastic Neighbor Embedding (t-SNE)
- Association rule learning
  - Apriori
  - Eclat

Unsupervised Learning: Clustering

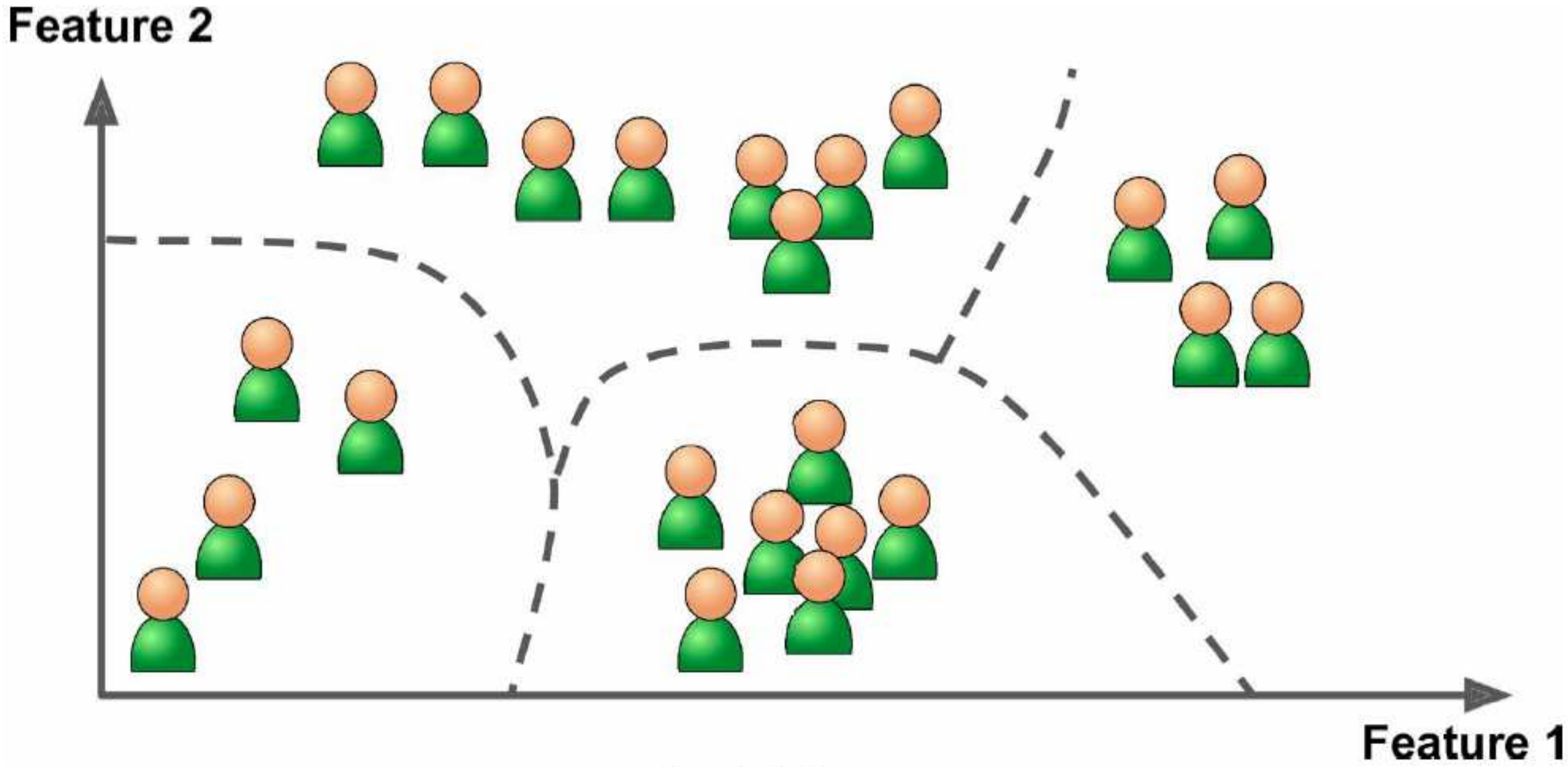


Figure 1-8. Clustering

Unsupervised Learning: Visualisation

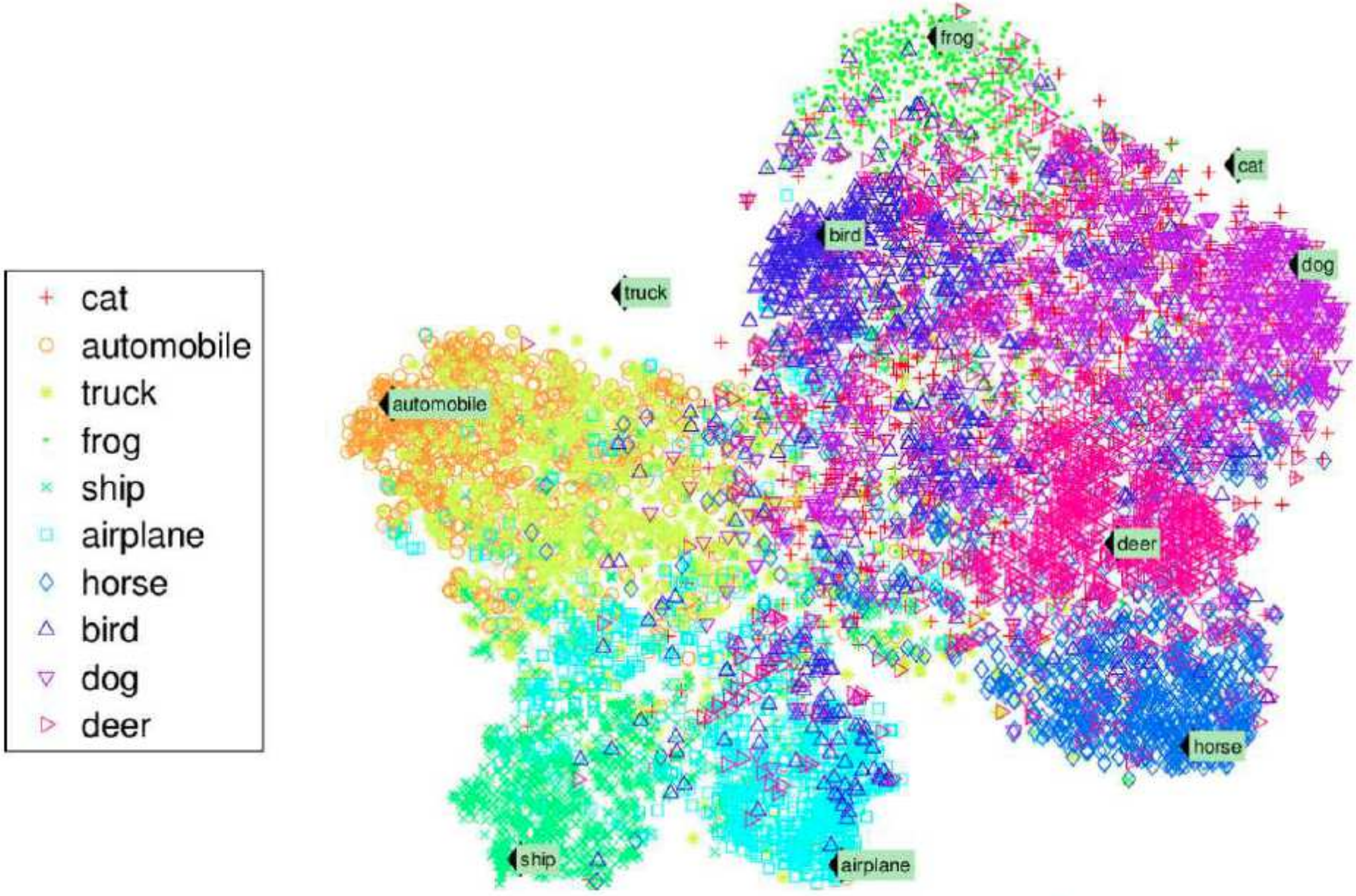


Figure 1-9. Example of a t-SNE visualization highlighting semantic clusters[3]

Unsupervised Learning: Anomaly detection

Objective is to learn what “normal” data looks like, and then use that to detect abnormal instances.

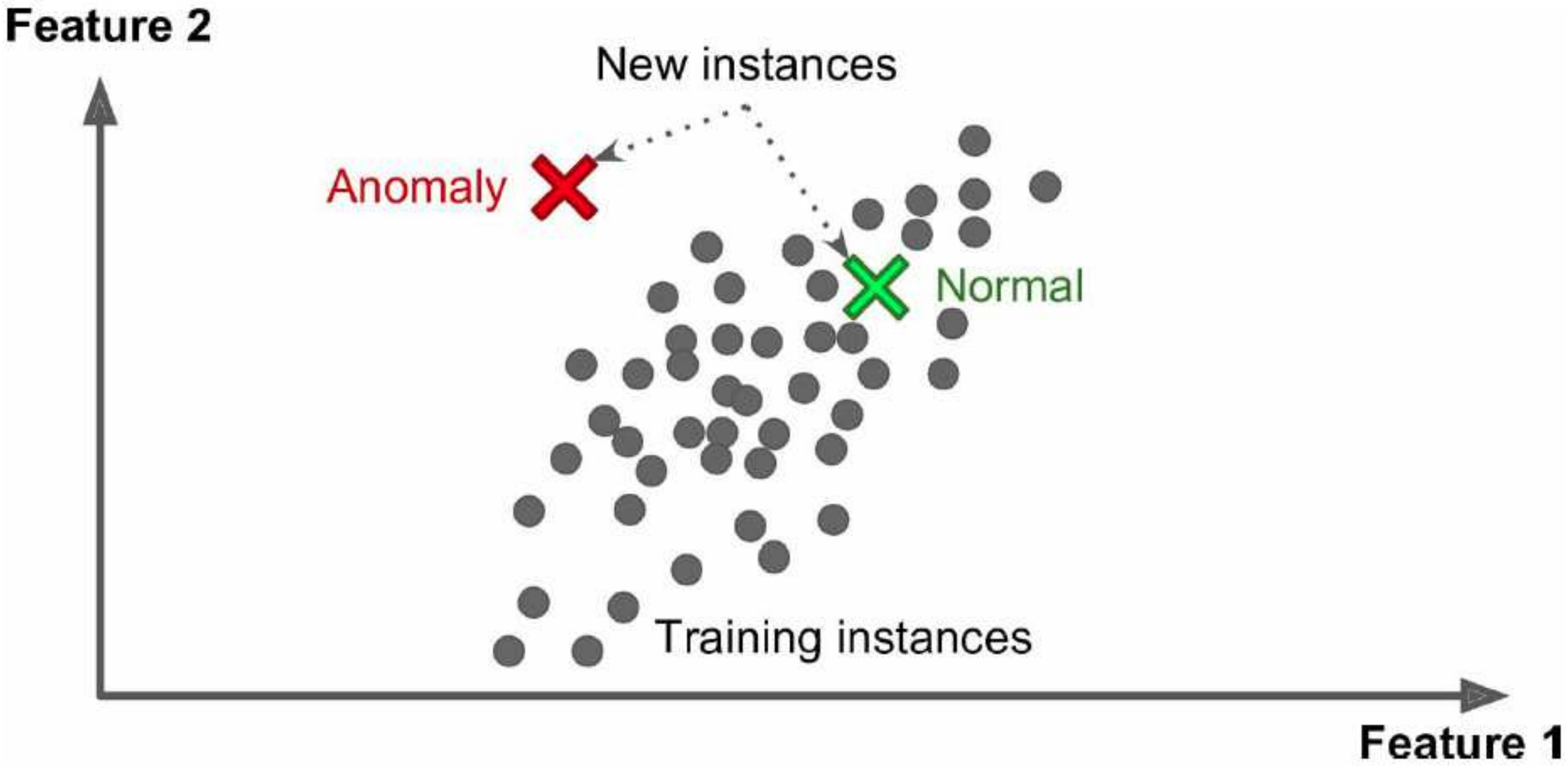


Figure 1-10. Anomaly detection

Semisupervised Learning

Only a few instances in the training data have labels.

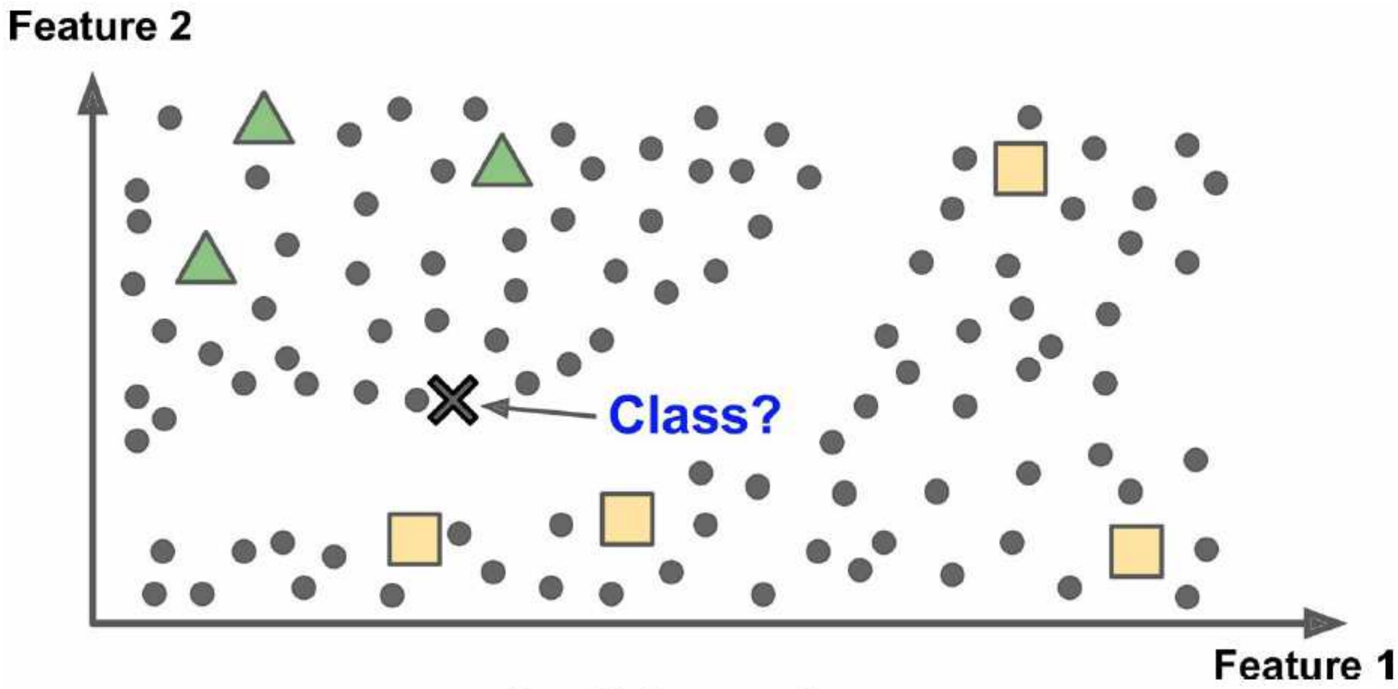


Figure 1-11. Semisupervised learning

Reinforcement Learning

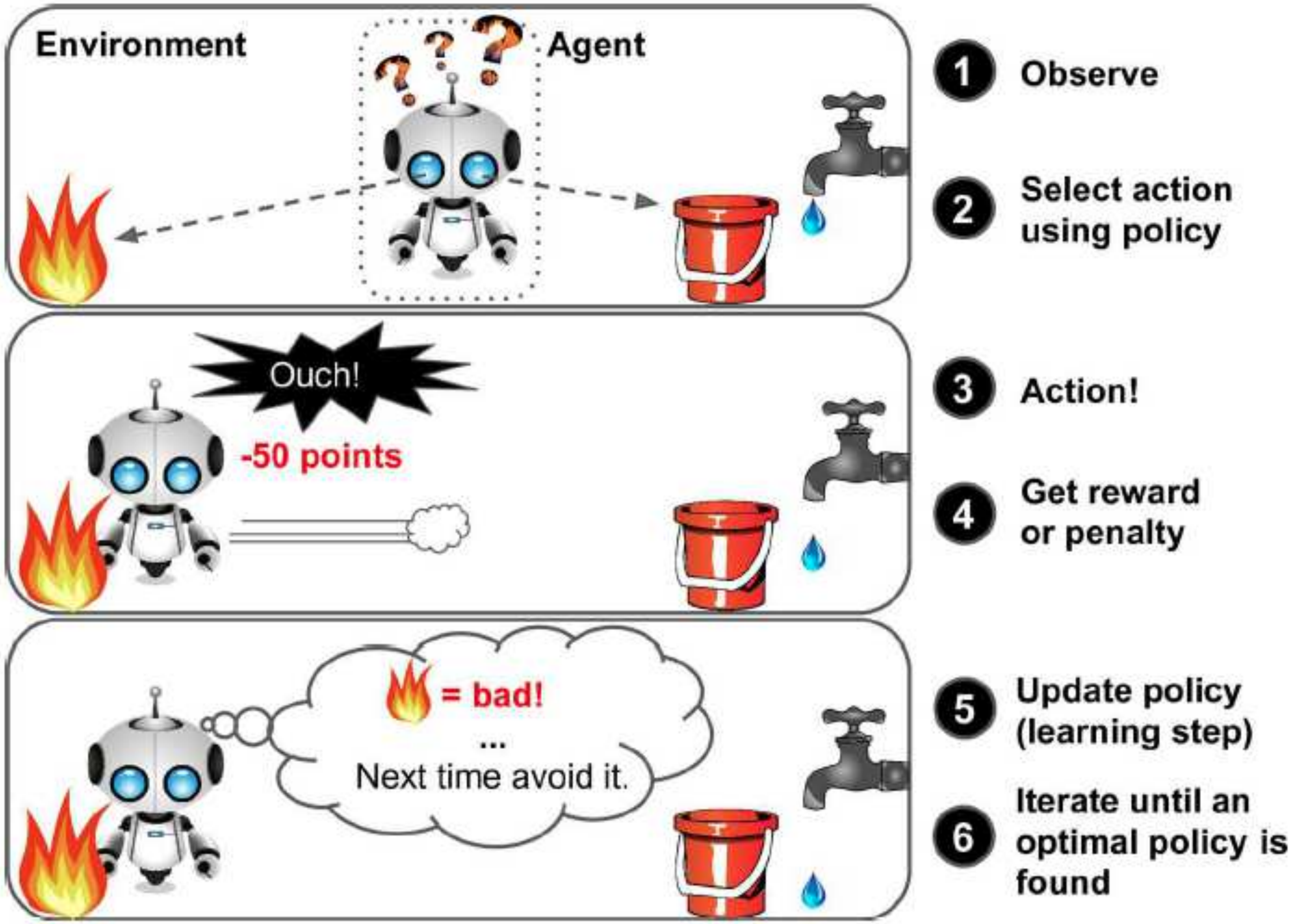

Environment
Agent
1 Observe
2 Select action using policy
Ouch!
-50 points
3 Action!
4 Get reward or penalty
= bad!
...
Next time avoid it.
5 Update policy (learning step)
6 Iterate until an optimal policy is found

Figure 1-12. Reinforcement Learning

Batch Learning vs Online Learning

Classifies Machine Learning systems is whether or not the system can learn incrementally from a stream of incoming data.

- **Batch/Offline Learning learning:** the system is trained using all the available data. First the system is trained, and then it is launched into production and runs without learning anymore; it just applies what it has learned.

- **Online/Incremental Learning:** the system is trained incrementally by sequentially feeding it with data instances, either individually or by small groups called *mini-batches*. Each learning step is fast and cheap, so the system can learn about new data on the fly, as it arrives.

Online Learning

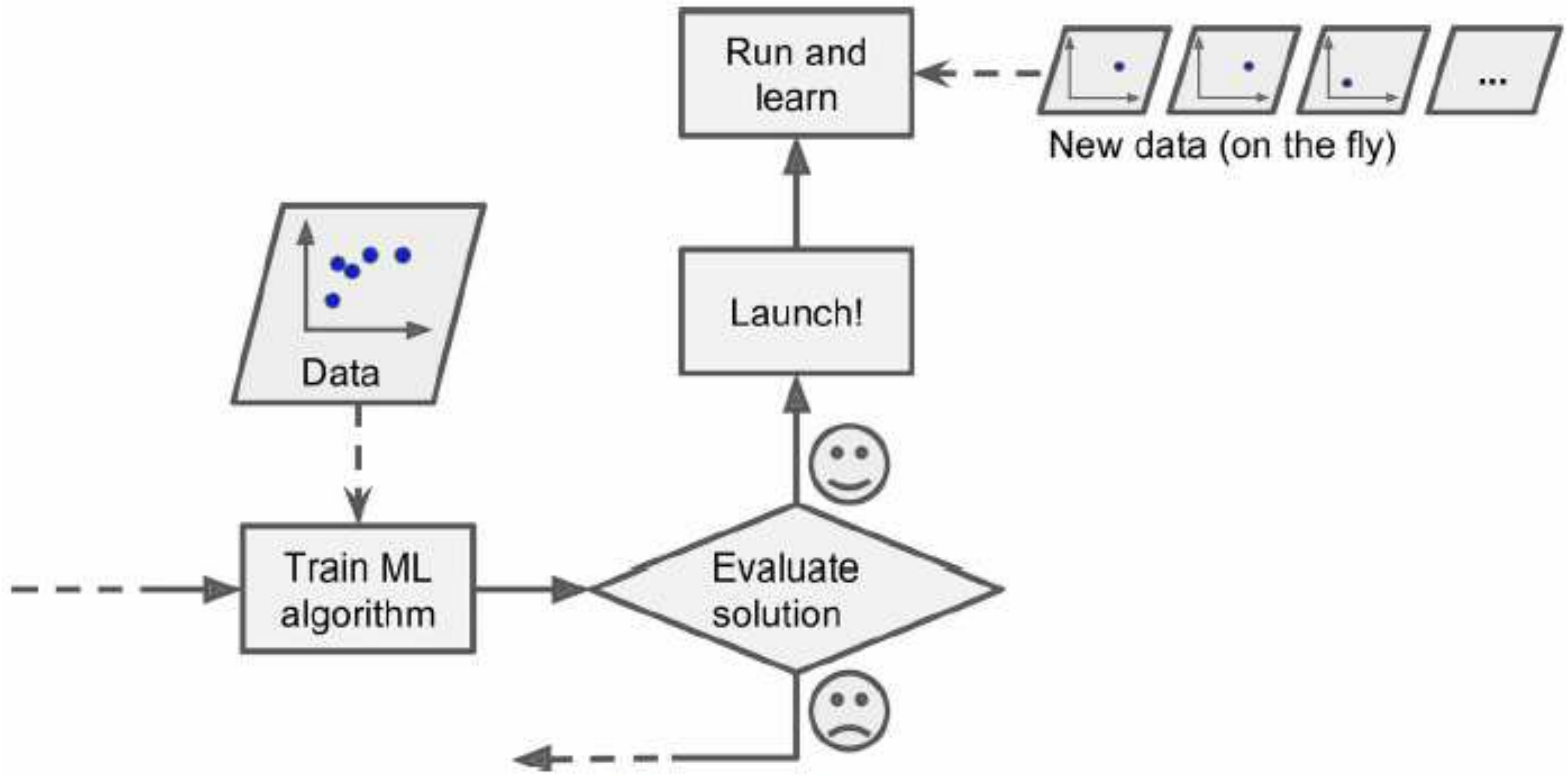


Figure 1-13. Online learning

Online Learning for Huge Datasets

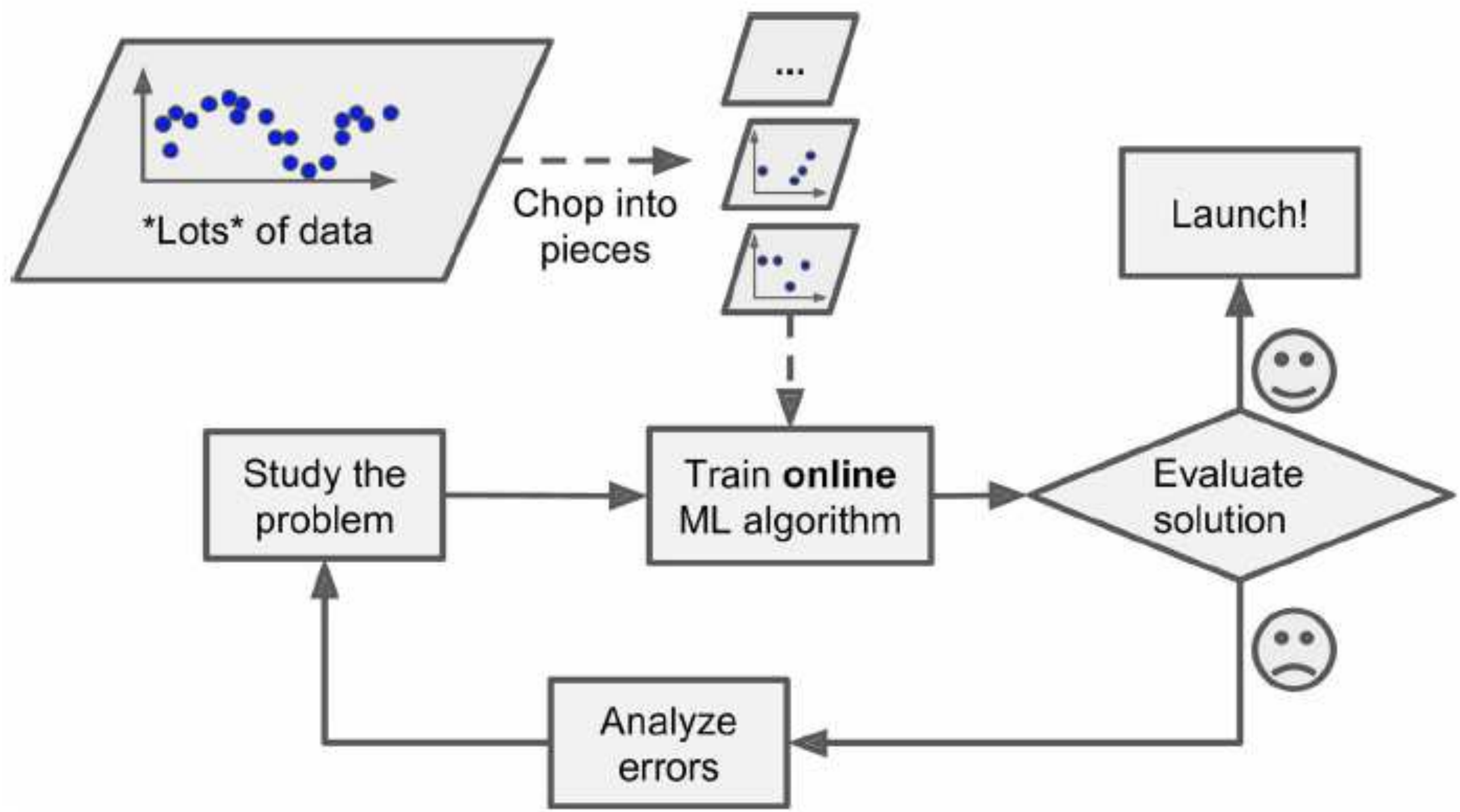


Figure 1-14. Using online learning to handle huge datasets

Instance-based Learning vs Model-based Learning

One more way to categorize Machine Learning systems is by how they generalize. Most Machine Learning tasks are about making predictions. This means that given a number of training examples, the system needs to be able to generalize to examples it has never seen before. Having a good performance measure on the training data is good, but insufficient; the true goal is to perform well on new instances.

There are two main approaches ...

Instance-based Learning

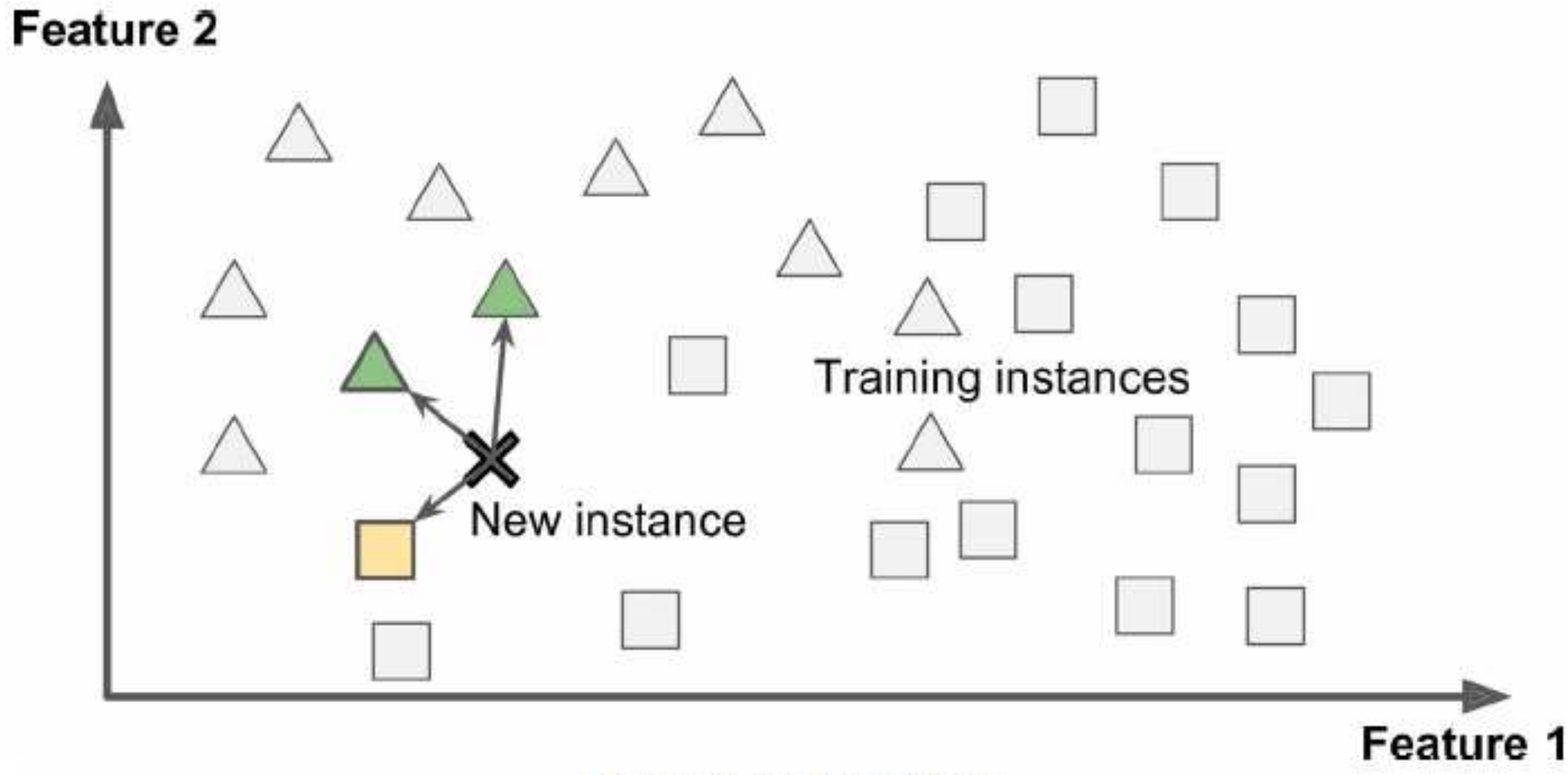


Figure 1-15. Instance-based learning

Model-based Learning

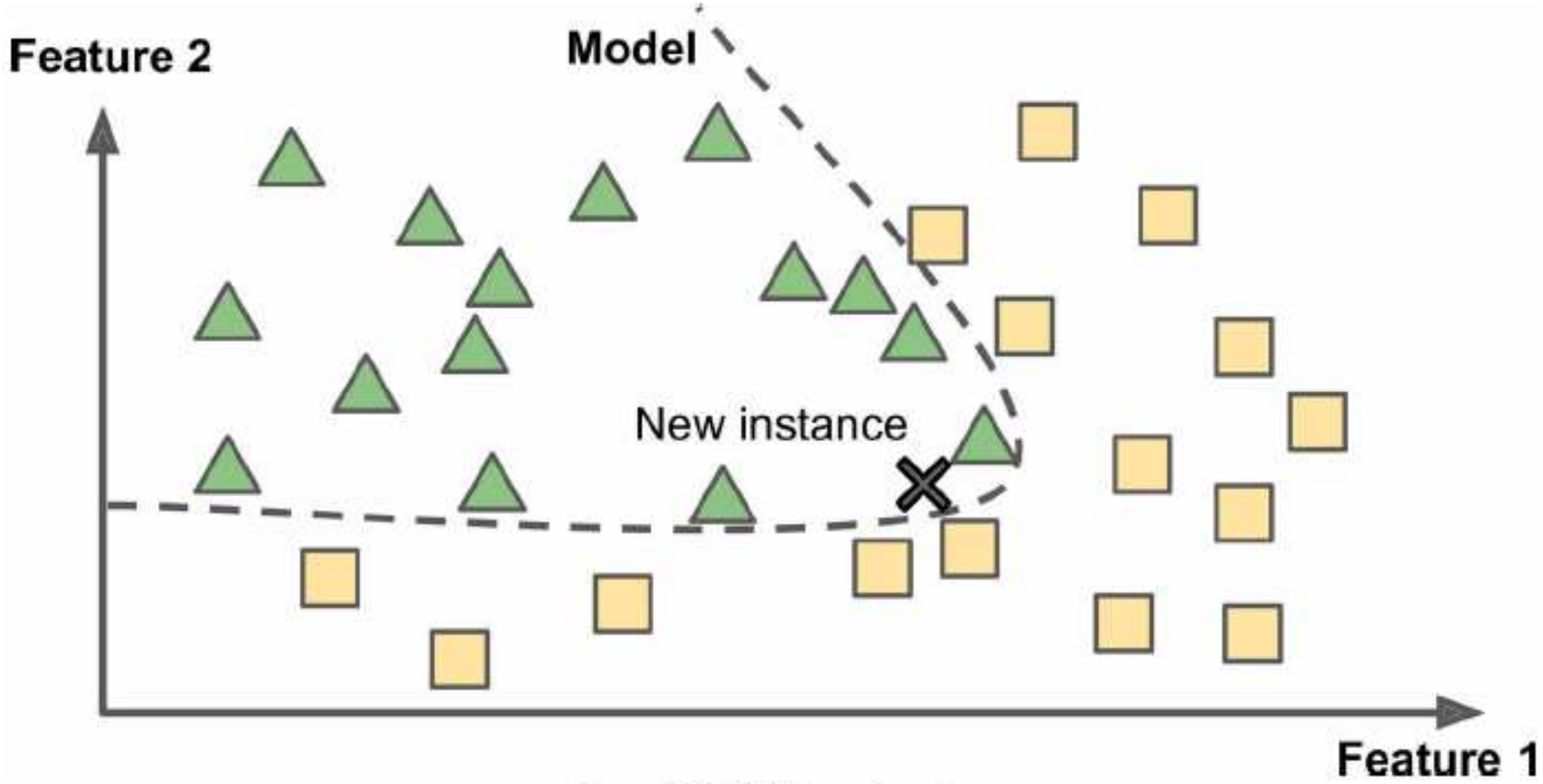


Figure 1-16. Model-based learning

Model-based Learning

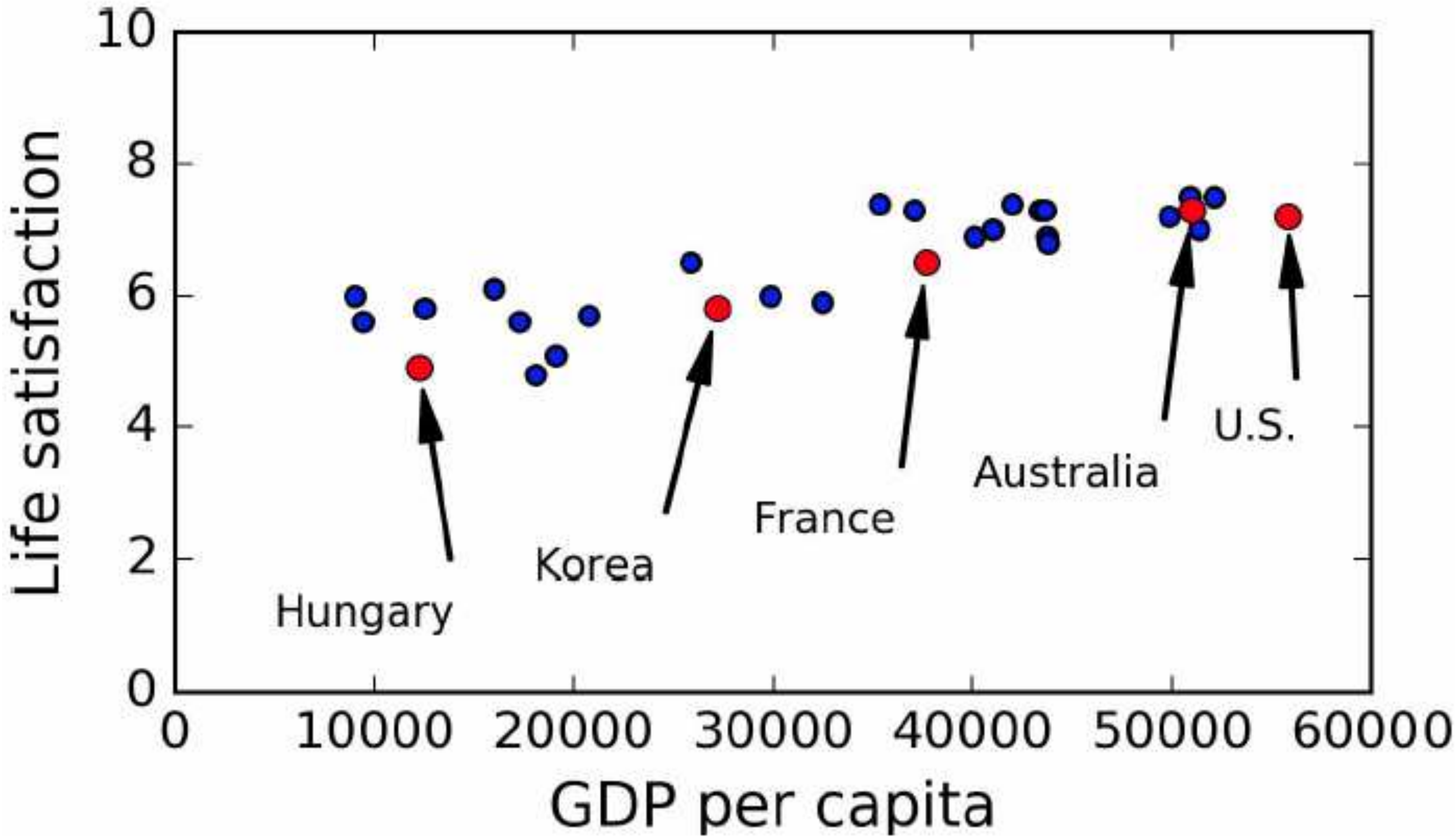


Figure 1-17. Do you see a trend here?

Model-based Learning

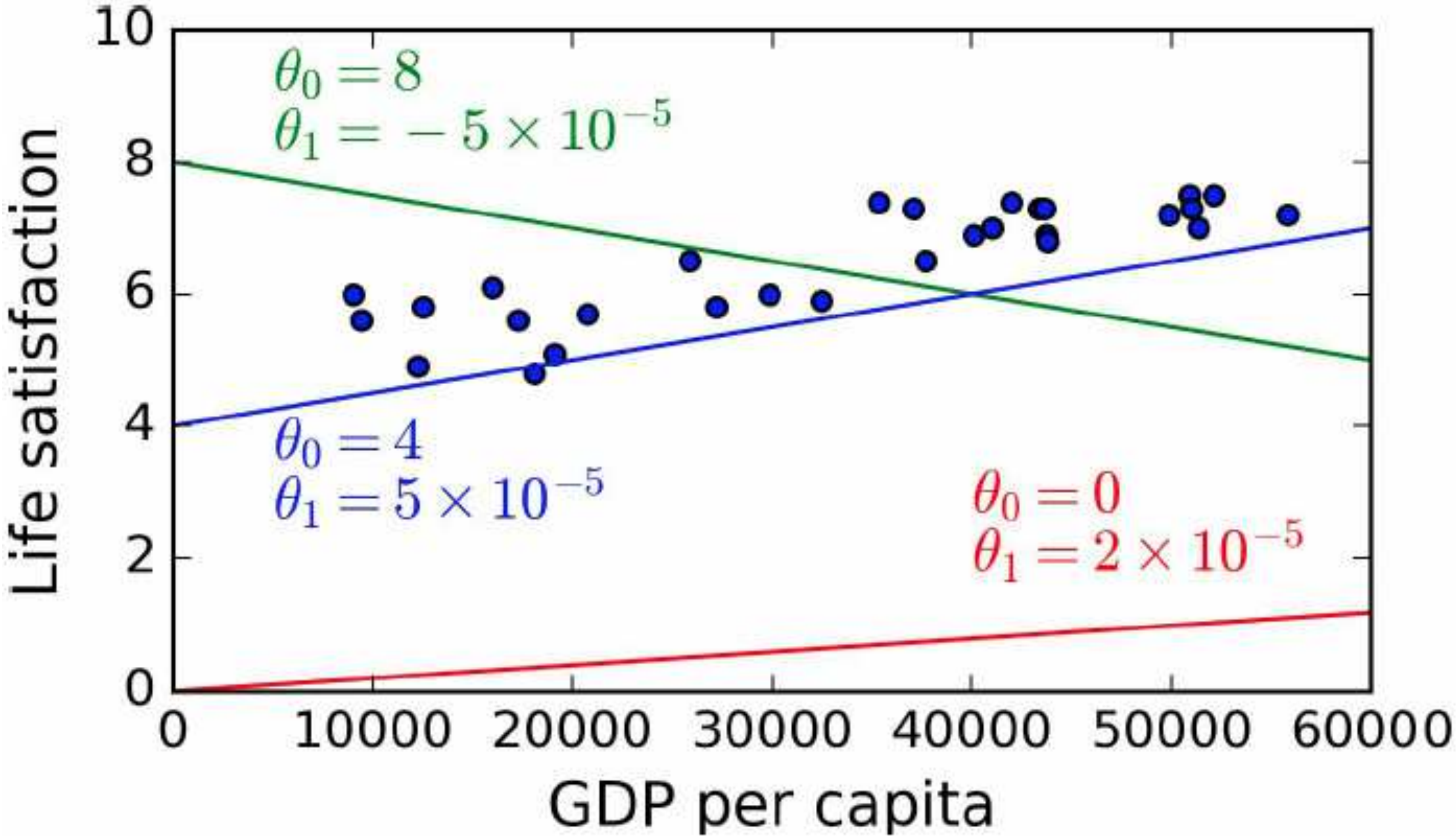


Figure 1-18. A few possible linear models

Model-based Learning

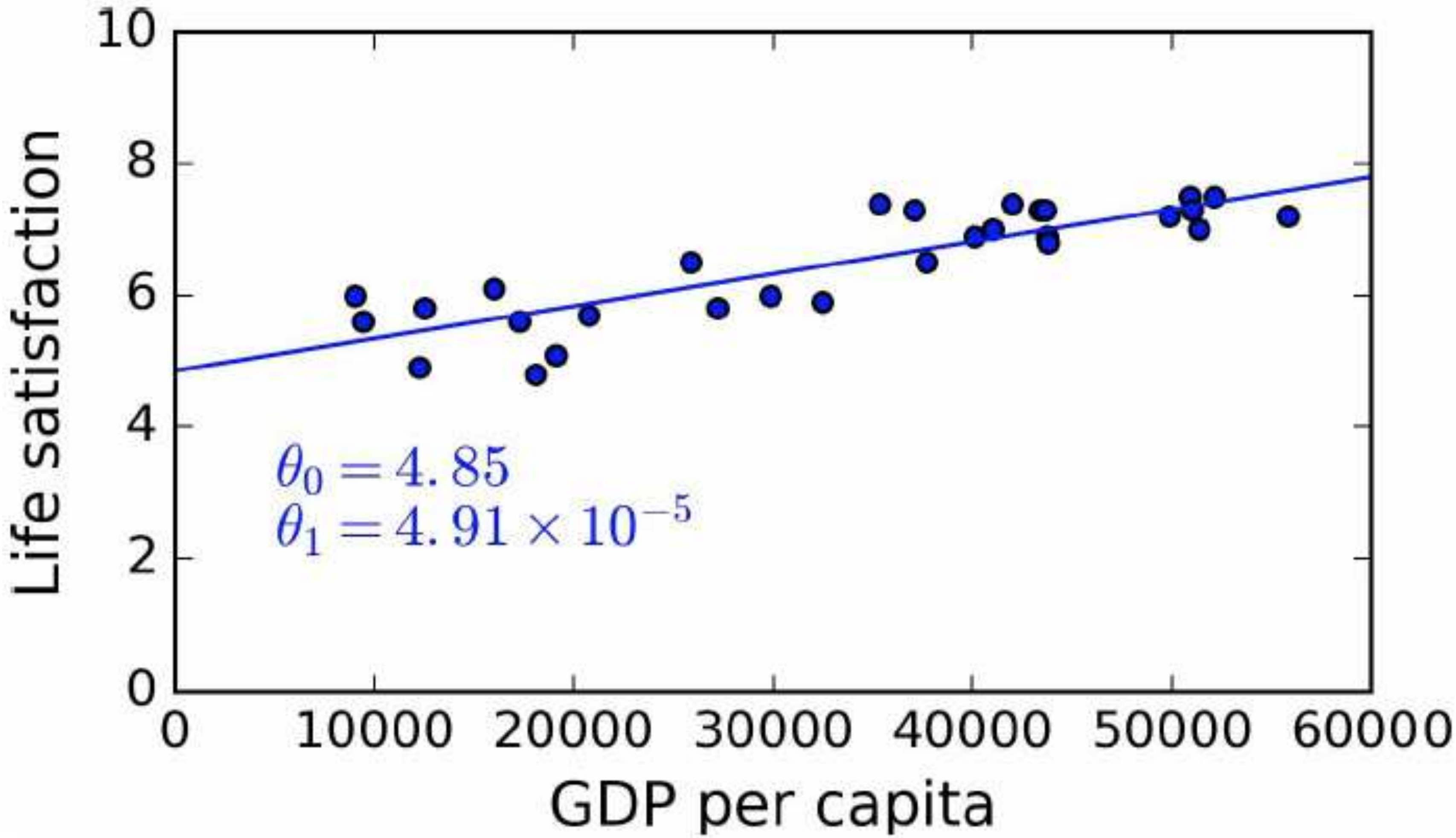


Figure 1-19. The linear model that fits the training data best

Main Challenges of machine learning

- Insufficient Quantity of Training Data
- Non-representative Training Data
- Poor-Quality Data
- Feature Engineering
- Overfitting or Underfitting the Training Data

Non-representative Training Data

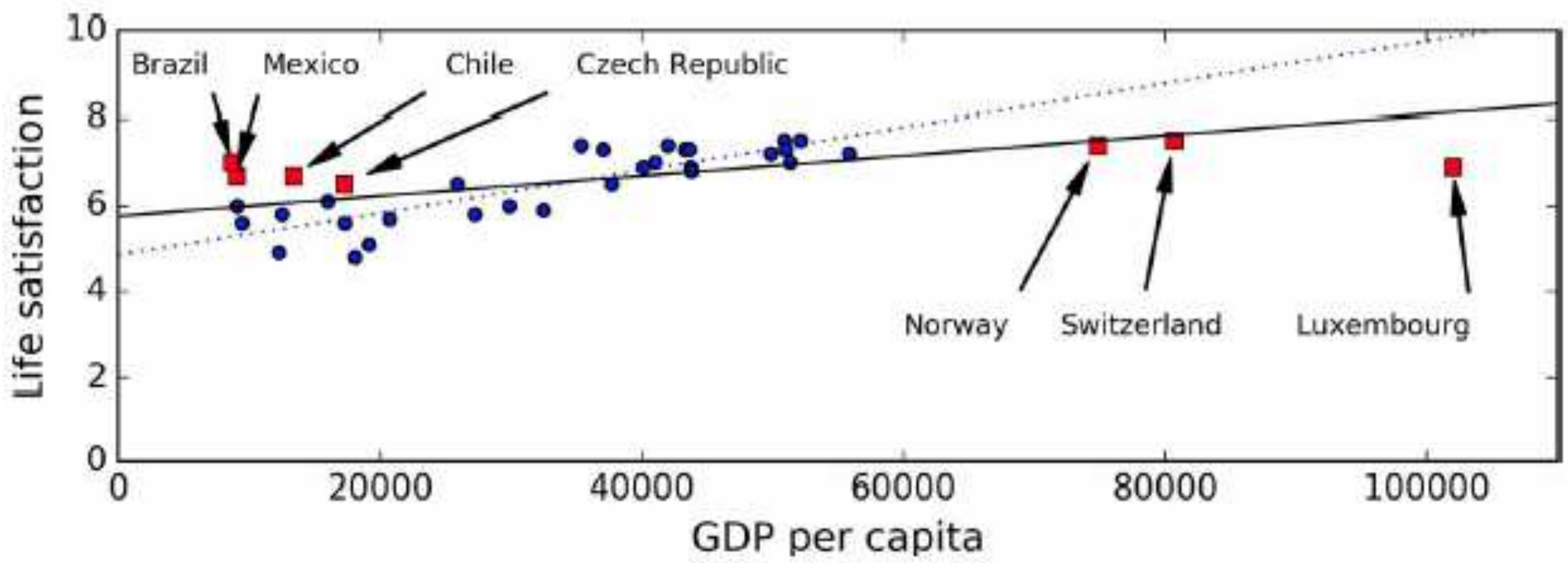


Figure 1-21. A more representative training sample

Overfitting the Training Data

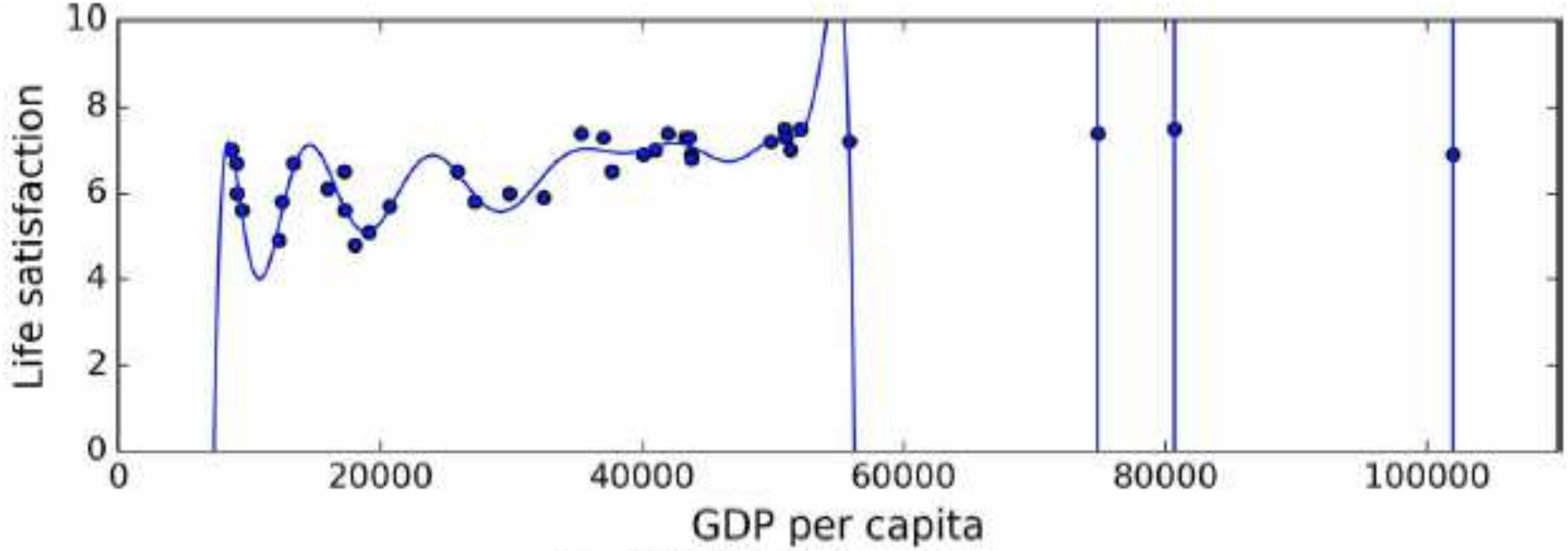


Figure 1-22. Overfitting the training data

Underfitting the Training Data

Underfitting is the opposite of **overfitting**: it occurs when your model is too simple to learn the underlying structure of the data. For example, a linear model of life satisfaction is prone to underfit; reality is just more complex than the model, so its predictions are bound to be inaccurate, even on the training examples.

The main options to fix this problem are:

- Selecting a more powerful model, with more parameters
- Feeding better features to the learning algorithm (feature engineering)
- Reducing the constraints on the model (e.g., reducing the regularization hyperparameter)

For next week

Obtain a copy of the text book and read up to chapter 2.

Set up Python (3.X) on your computer. Write a few simple programs to be familiar with the basic syntax.

And that's all for the first lecture.

Enjoy!