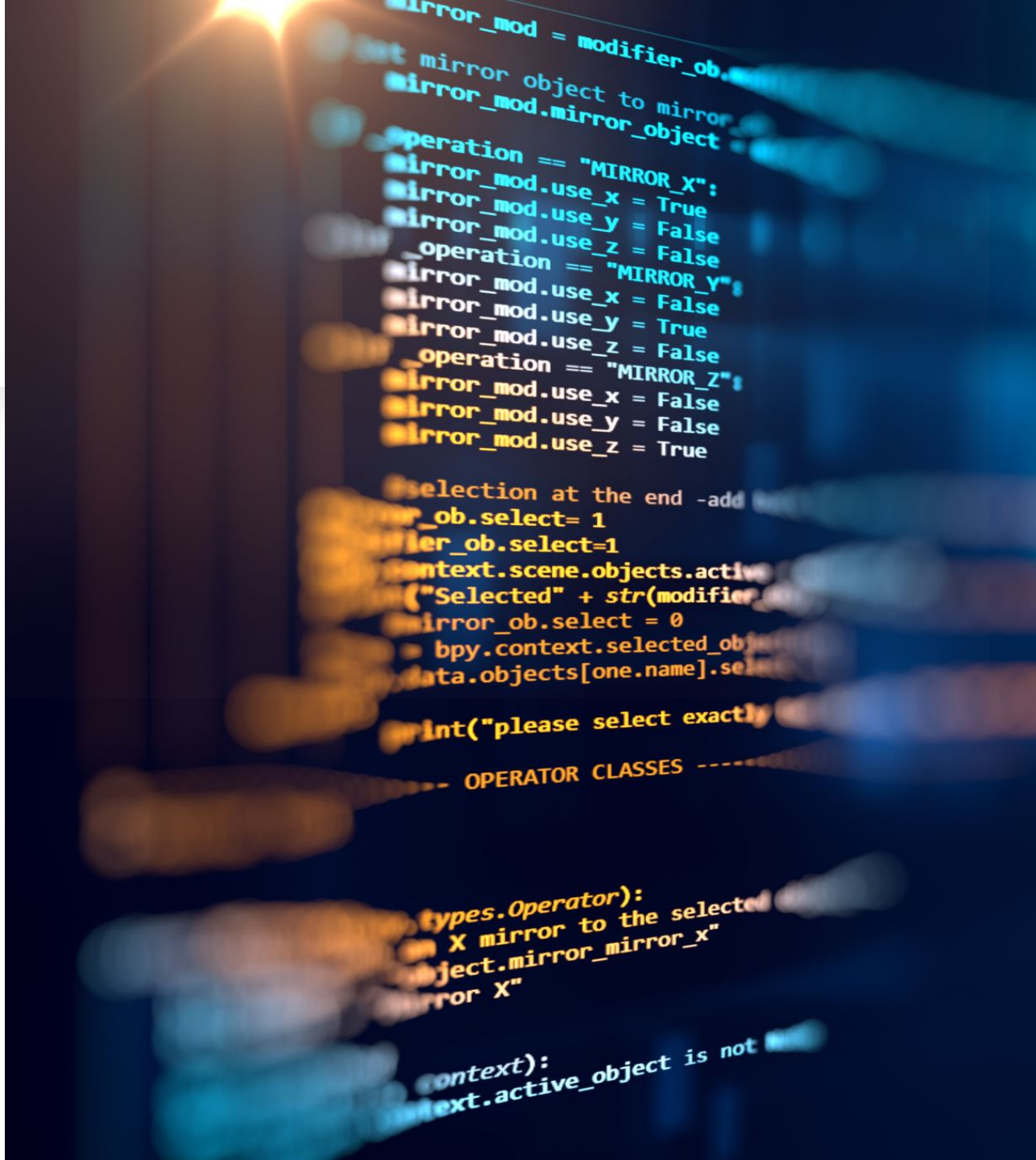
The Machine Learning Landscape

Concepts, Tools, and Techniques to Build Intelligent Systems

Introduction to Machine Learning

- **Definition:**
  - *"Machine learning is the science (and art) of programming computers so they can learn from data."*
  - *"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
- **Key Components:**
  - Training data
  - Model (e.g., neural networks, random forests)
  - Performance measure (e.g., accuracy, RMSE)



Why Use Machine Learning?

- **Advantages:**
  - **Automation:** Reduces manual rule-writing (e.g., spam filters).
  - **Adaptability:** Learns from new data (e.g., evolving spam tactics).
  - **Complex Problems:** Solves tasks with no known algorithm (e.g., speech recognition).
  - **Data Mining:** Uncovers hidden patterns (e.g., customer segmentation).

Example: Spam filter vs. manual rules

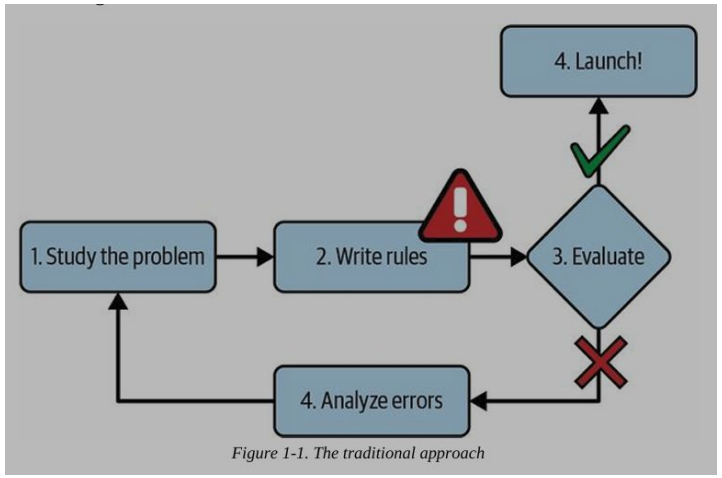


Figure 1-1. The traditional approach

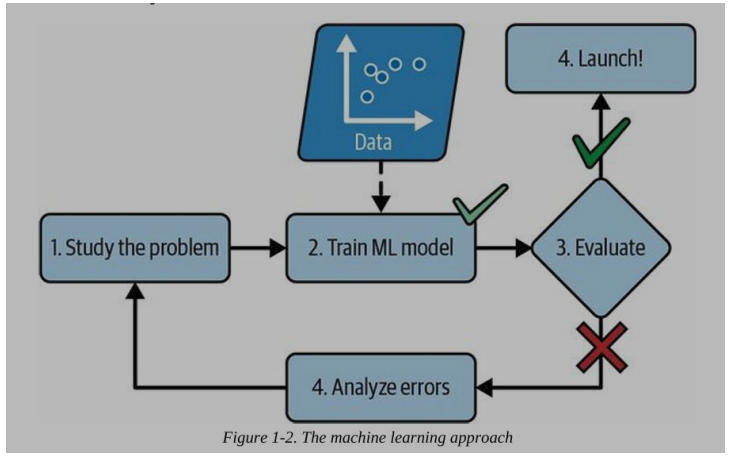


Figure 1-2. The machine learning approach

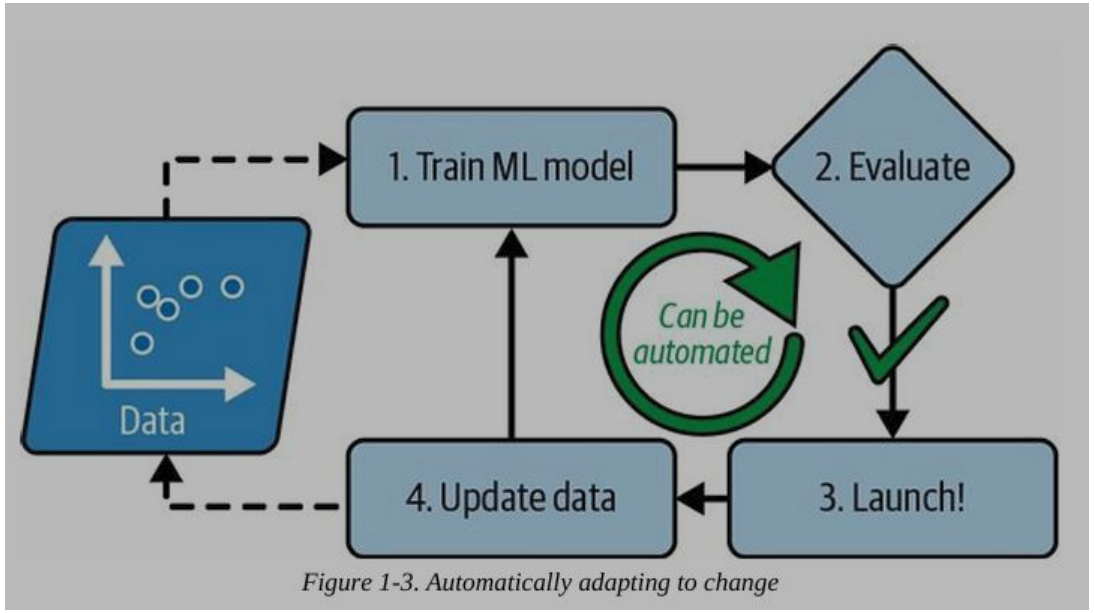


Figure 1-3. Automatically adapting to change

Examples of ML Applications

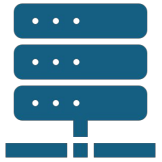
- **Image Classification:** CNNs for product sorting (e.g., tumors in scans).
- **Natural Language Processing:** Text classification (e.g., offensive comments, classifying news articles), Text summarization, Creating a chatbot or a personal assistant.
- **Regression:** Forecasting revenue (e.g., linear regression).
- **Anomaly Detection:** Fraud detection (e.g., credit card transactions).
- **Reinforcement Learning:** Game bots (e.g., AlphaGo).
- **Speech Processing**: RNNs, CNNs, or transformers for speech recognition.
- **Clustering:** Segmenting clients based on their purchases
- **Dimensionality Reduction:** Representing a complex dataset in a clear and insightful diagram.
- **Recommender System:** Recommending a product that a client may be interested in, based on past purchases.

Types of ML Systems

Supervised Learning

Labeled data (e.g., spam/ham emails).

Tasks: Classification (spam filter), Regression (house prices).



Unsupervised Learning

Unlabeled data (e.g., customer segmentation).

Tasks: Clustering, dimensionality reduction, anomaly detection, association rule learning.

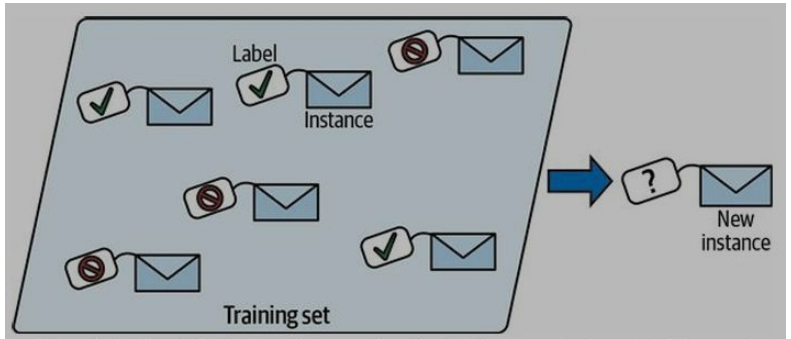


Figure 1-5. A labeled training set for spam classification (an example of supervised learning)

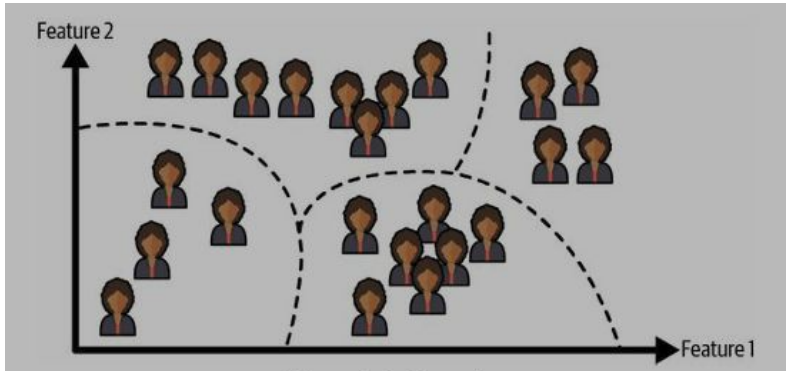


Figure 1-8. Clustering

Types of ML Systems

- **Semi-Supervised Learning**
  - Mix of labeled/unlabeled data (e.g., Google Photos face recognition).
- **Self-Supervised Learning**
  - Generate fully labeled dataset from a fully unlabeled one (e.g., image inpainting).
- **Reinforcement Learning**
  - Agent learns via rewards/penalties (e.g., robot walking).

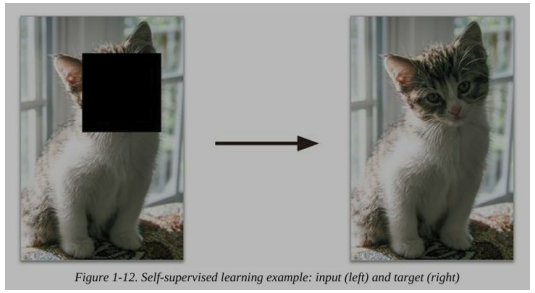
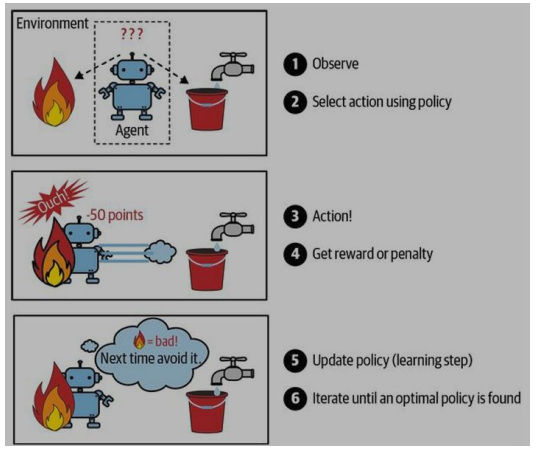
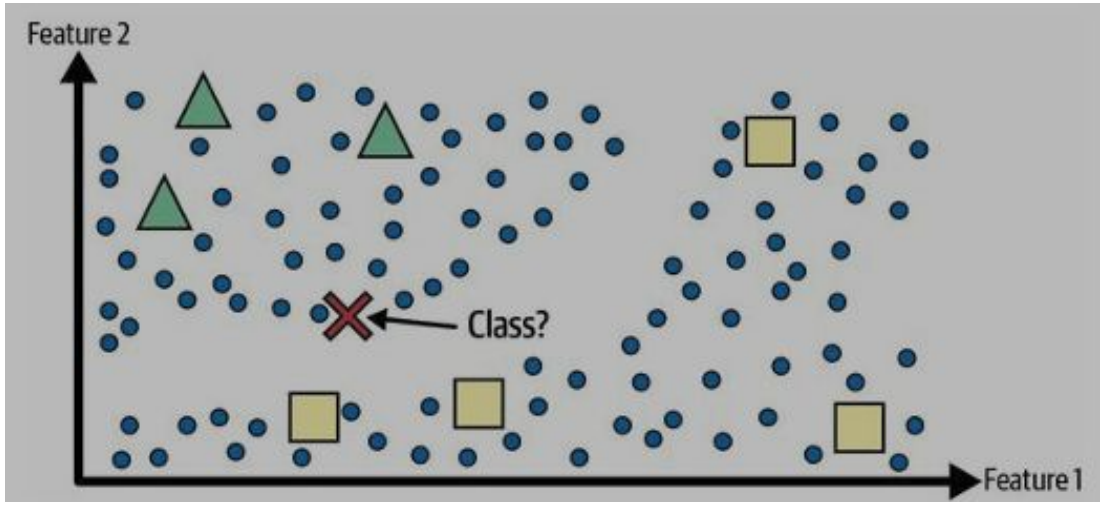


Figure 1-12. Self-supervised learning example: input (left) and target (right)





Batch vs. Online Learning

Batch Learning

- Trained on full dataset (offline).
- **Drawback:** Model rot or data drift (Model's performance tends to decay slowly over time), Require more computing resources.
- **Example:** Classifies pictures of cats and dogs, Making predictions on the financial market

Online Learning

- Incremental updates (system learn about new data on the fly)
- **Drawback:** If bad data is fed to the system, the system's performance will decline.
- **Example:** Stock price predictions.

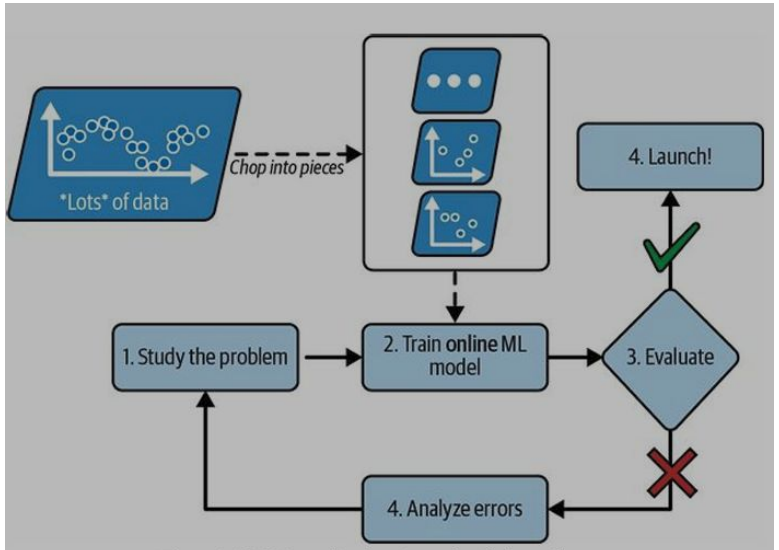


Figure 1-15. Using online learning to handle huge datasets

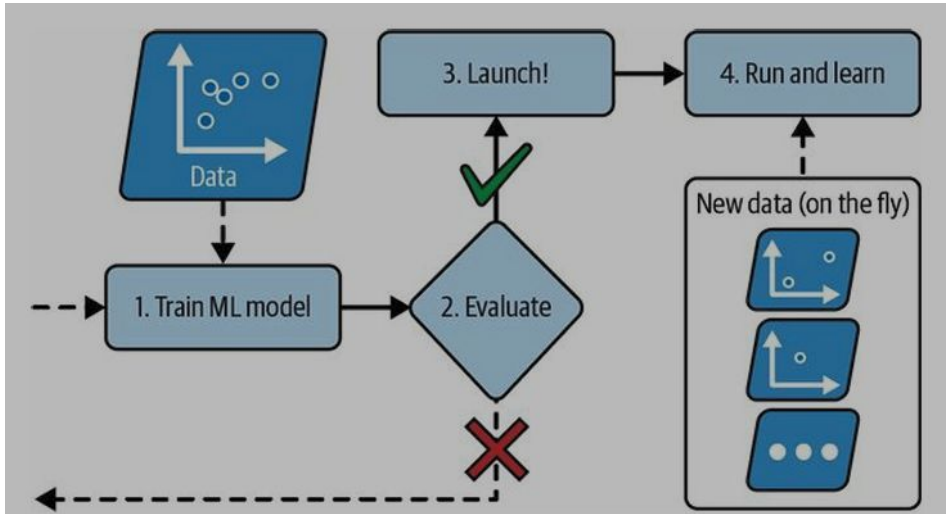


Figure 1-14. In online learning, a model is trained and launched into production, and then it keeps learning as new data comes in

Instance-Based vs. Model-Based Learning

- **Instance-Based:** Memorizes data (e.g., k-nearest neighbors).
  - The system learns the examples by heart, then generalizes to new cases by using a similarity measure to compare them to the learned examples.

- **Model-Based:** Generalizes patterns (e.g., linear regression).
  - Build a model of examples and then use that model to make predictions.
  - **Example:** Predicting life satisfaction: https://colab.research.google.com/drive/1vNRge8bHUc3qKg1FA7qjOO2WOFjEqWYg#scrollTo=R37nSnn2pnub

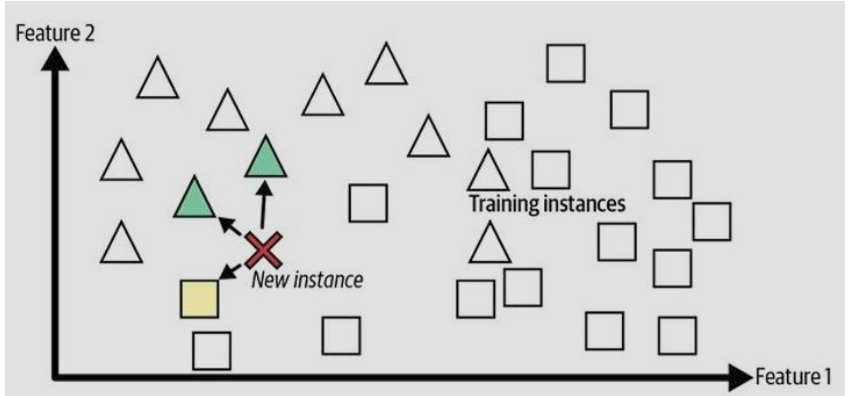


Figure 1-16. Instance-based learning

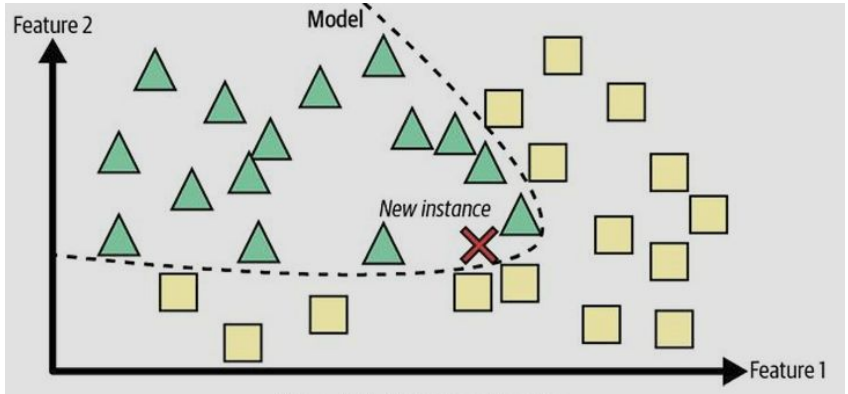


Figure 1-17. Model-based learning

Main Challenges in ML

- **Insufficient Data** – Small datasets limit performance.
- **Nonrepresentative Data** – Sampling noise (the sample is too small), Sampling bias (very large samples can be nonrepresentative if the sampling method is flawed, e.g., 1936 election poll).
- **Poor-Quality Data** – Errors, outliers, missing values.
- **Irrelevant Features** – Garbage in, garbage out.
- **Overfitting** – Model performs well on the training data, but it does not generalize well.
- **Underfitting** – Model too simple.

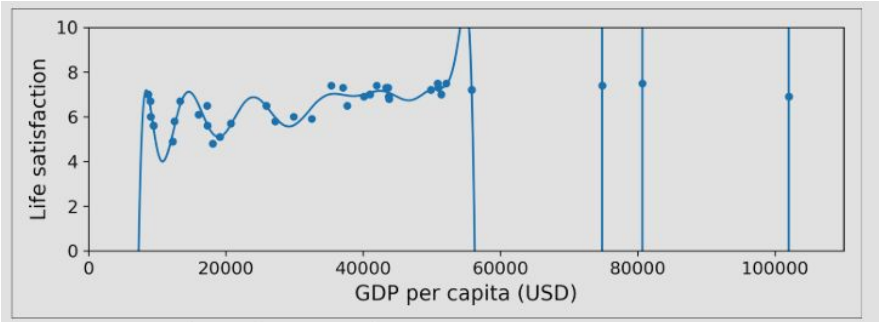
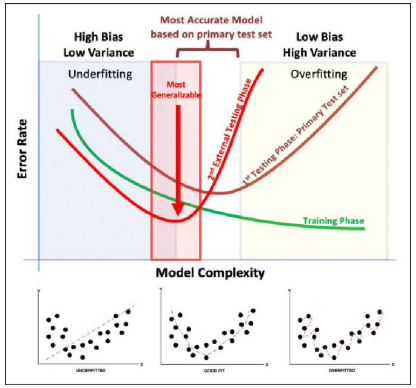
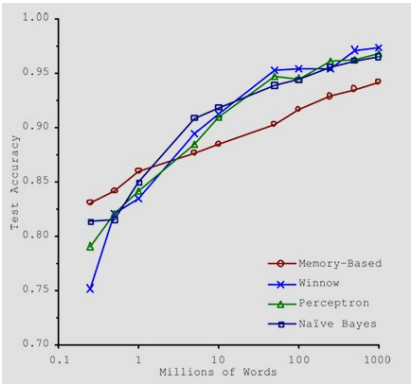


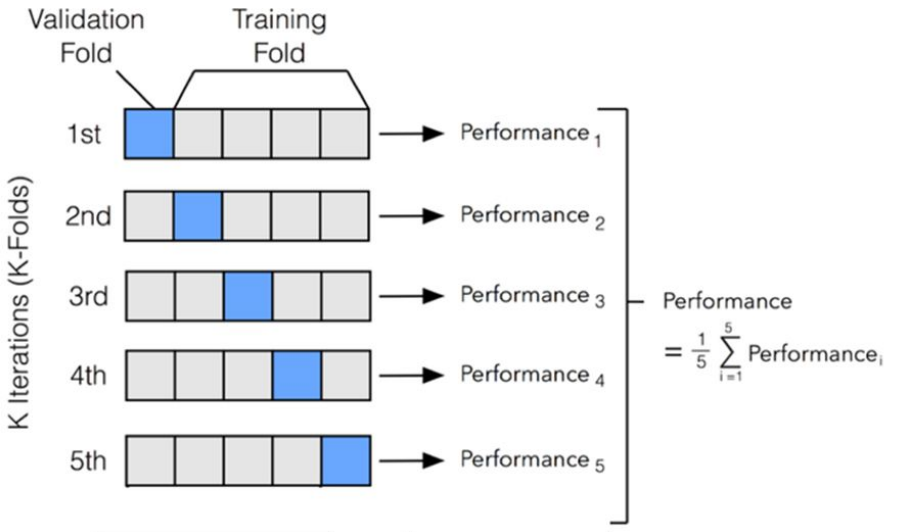
Figure 1-22. Overfitting the training data





Evaluating & Fine-Tuning Models

- **Train/Test Split:** 80% of the data for training and hold out 20% for testing. Compute generalization errors.
- **Cross-Validation:** Reliable performance estimate (e.g., 10-fold CV).
- **Hyperparameter Tuning:**
  - Train multiple models with various hyperparameters on the reduced training set (i.e., the full training set minus the validation set).
  - Select the model that performs best on the validation set.
  - Train the best model on the full training set (including the validation set).
  - Evaluate this final model on the test set to get an estimate of the generalization error.



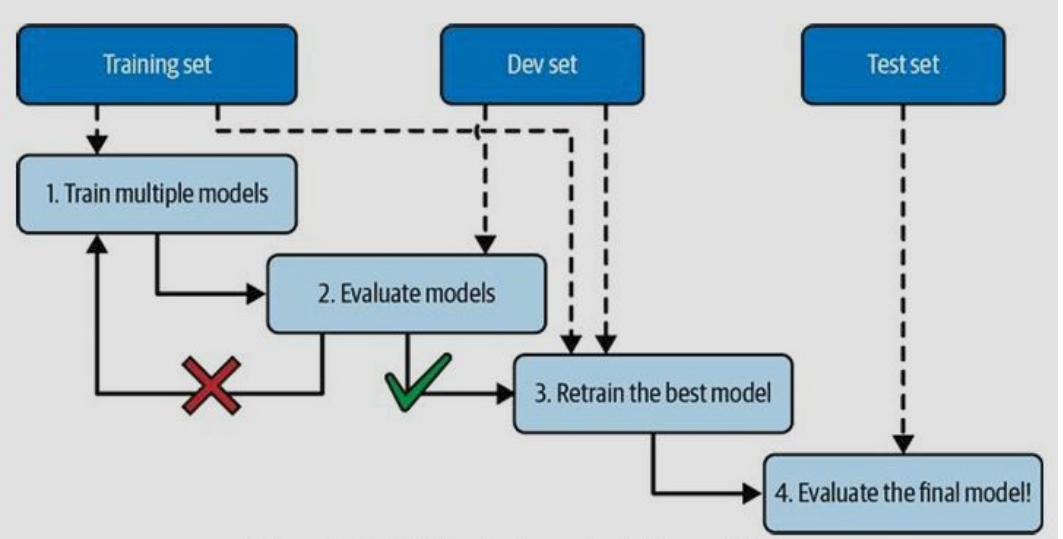


Figure 1-25. Model selection using holdout validation