Introduction to Machine Learning

CS576 MACHINE LEARNING

Dr. Jin Soung Yoo, Professor
Department of Computer Science
Purdue University Fort Wayne

Reference

- Kelleher et al., Machine Learning for Predictive Data Analytics, 2nd ed., Ch1.
- Mitchell, Machine Learning, Ch1 & Ch2
- Alpaydin, Introduction to Machine Learning, 4th ed., Ch1
- Géron, Hands-On Machine Learning, 2nd ed., Ch1
- Watt et al., Machine Learning Refined, 2nd ed., Ch1

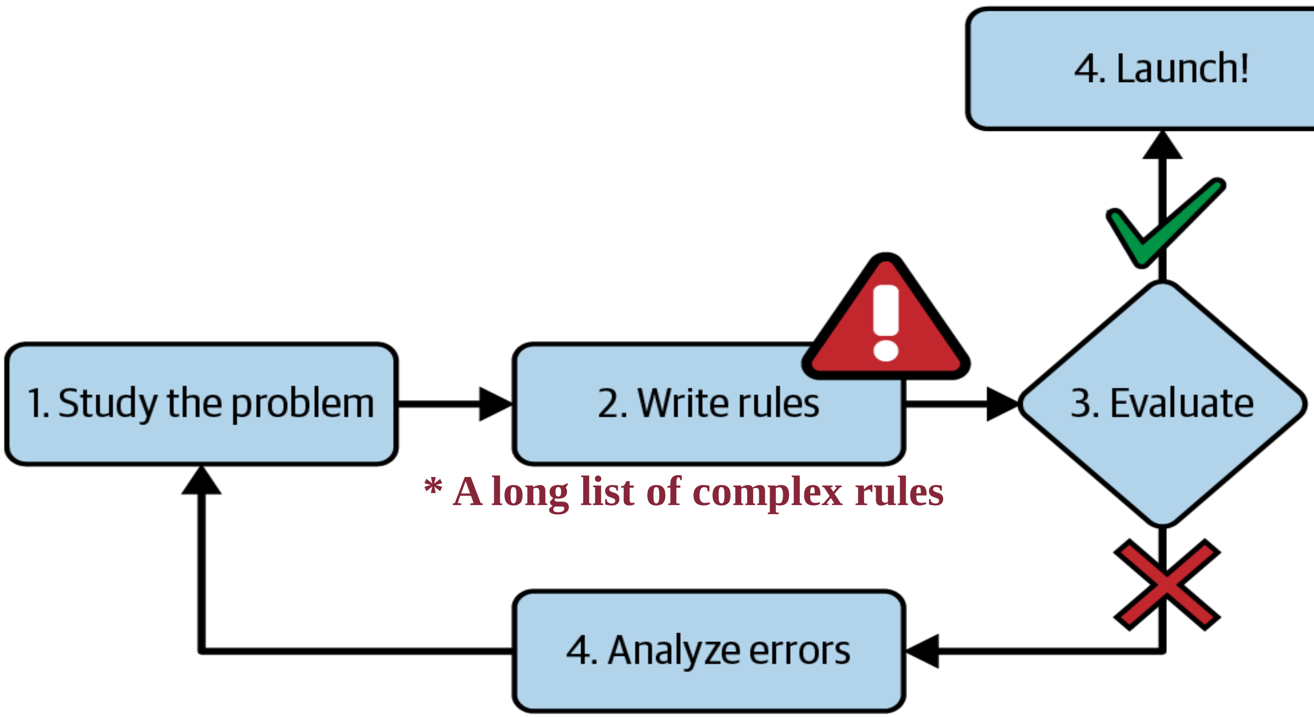
Outline

- What Is Machine Learning?
- Machine Learning Applications
- Machine Learning Experiment
- Goal of Machine Learning
- Machine Learning as a Search Problem
- Types of Machine Learning Systems
- Challenges of Machine Learning
- Disciplines of Machine Learning
- Summary

Introduction

- Machine learning (ML) is no longer science fiction: billions of people use it everyday.
- Hundreds of **ML applications** with some ***learning components***
  - spam filter
  - face recognition, speech recognition
  - automatic translation
  - image search
  - product recommendations,
  - self-driving cars and many more
- Machine learning underlines the coolest technologies today.
- Machine learning is one of the hottest research areas in computer science.

Example: Spam Filter - Traditional Approach

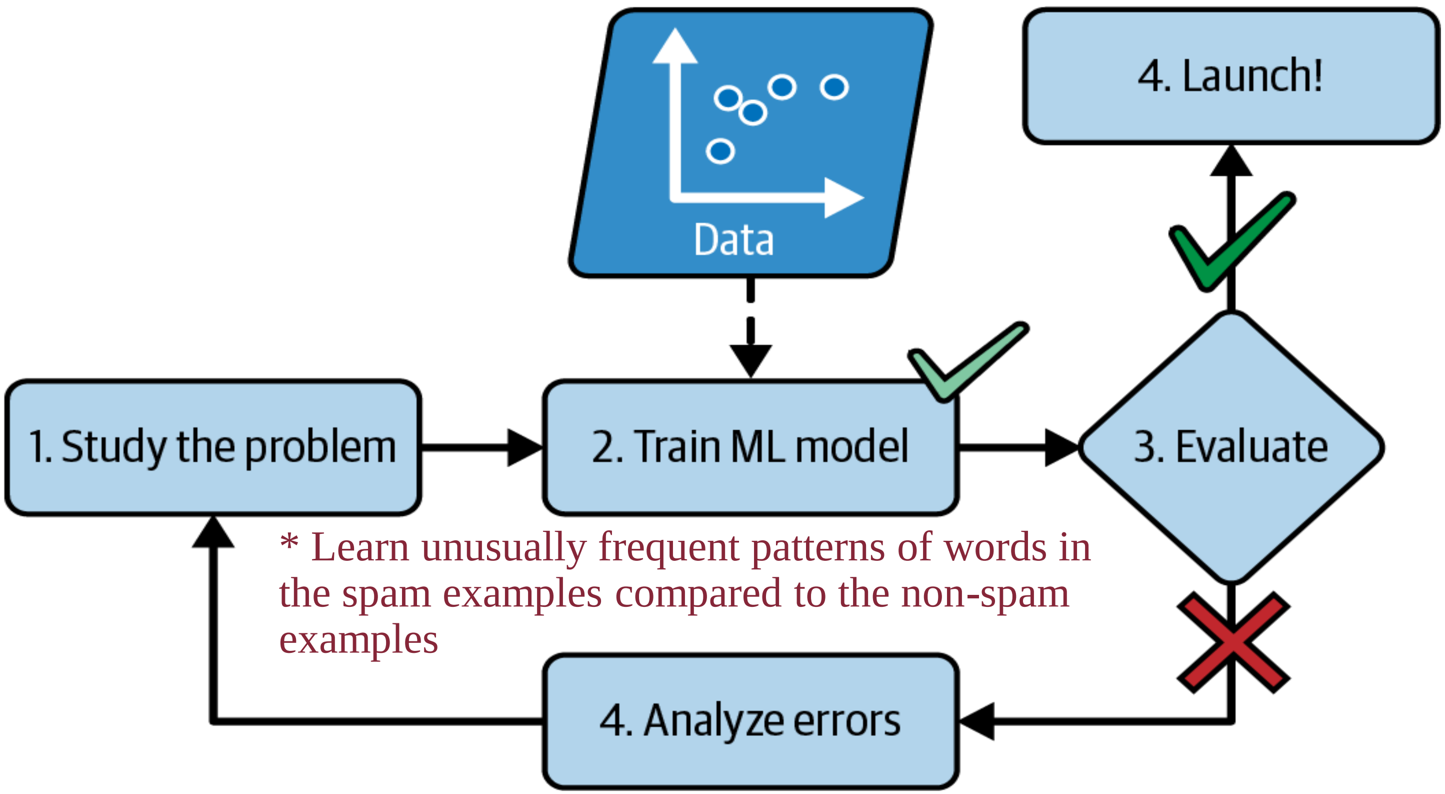


* A long list of complex rules

- **Traditional approach for a spam filter**
  1. Examine what spam typically looks like.
  2. Write a detection program for each of the patterns, e.g., “credit card” in title
  3. Test the program and repeat steps 1 and 2 until it was good enough to launch.

Example: Spam Filter - Machine Learning Approach

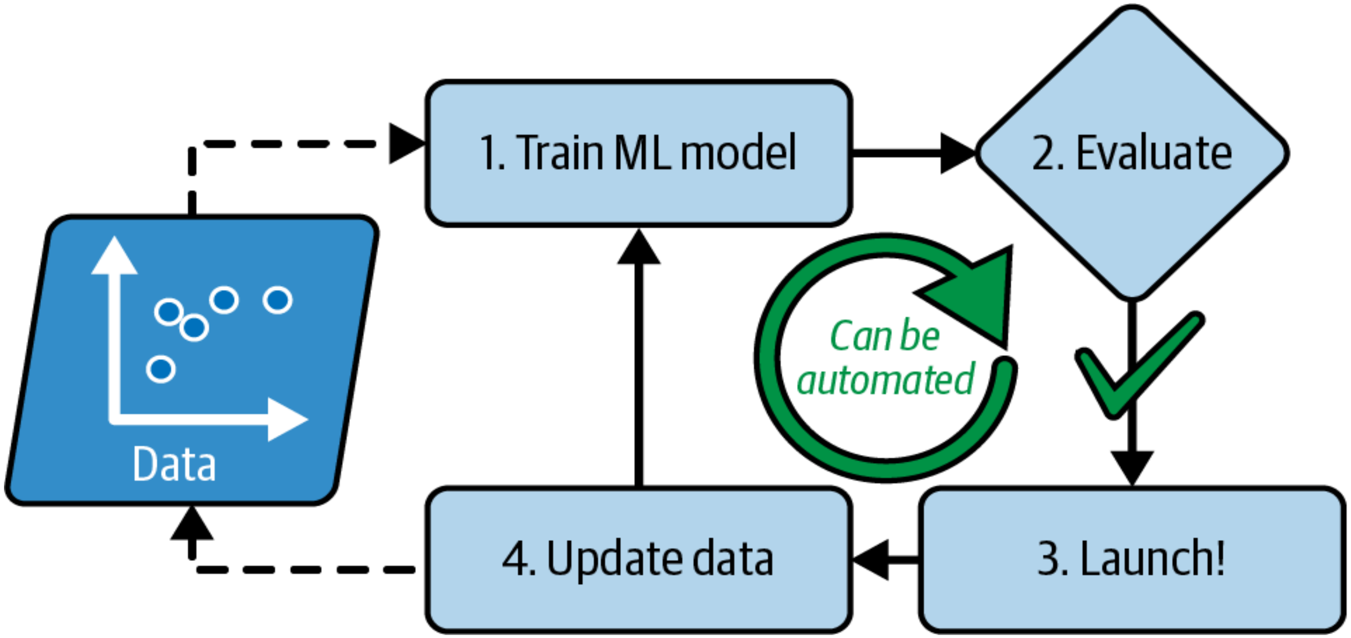
- **A spam filter based on machine learning techniques** automatically ***learns*** which words and phrases are good predictors of spam in the spam examples.



* Learn unusually frequent patterns of words in the spam examples compared to the non-spam examples

Example: Spam Filter - Machine Learning Approach (cont.)

- A spam filter based on learning techniques automatically **can adapt to change**.
  - E.g., All emails containing "4U" flagged by users and they are blocked → Later, emails with "For U" flagged by users. They become blocked → …



History

- **Ever since computers were invented, we have wondered whether computers might be made to *learn*,** e.g.,
  - Learn from medical records which treatments are most effective for new diseases, or
  - Learn the evolving interests of users in order to highlight especially relevant stories from the online morning newspaper.
  - …

Example

- **Arthur Samuel** (1959) showed some study in machine learning using the game of checkers.

Figure: Playing checkers on the 701
- *from IBM*

- Arthur said that "**Machine Learning** is the field of study that gives the computer the ability to ***learn*** without being explicitly programmed."

What We Talk About When We Talk About "Learning"

- What exactly does it mean for a machine to ***learn*** something?
- There are many applications for which we do not have an algorithm but have lots of data
- We may not be able to identify the learning process completely
  - How to transform the input (e.g., an email) to the output (e.g., spam or not)

But we believe we can

- learn **general models** (detect certain patterns of regularities) from a dataset of particular examples,
- construct a model that is *a good and useful approximation* to the data for understanding the learning process or making predictions

Why “Learn” ?

- There is no need to “learn” to calculate payroll
- ***Learning*** is used when:
  - human expertise does not exist (e.g., navigating on Mars),
  - humans are unable to explain their expertise (e.g., speech recognition)
  - there are software applications we can’t program by hand
    - Autonomous driving
    - Vision, speech recognition, robotics
  - solution changes in time (self customizing programs)
    - Newsreader that learns user interests
    - Routing on a computer network
  - solution needs to be adapted to particular cases (e.g., user biometrics)

Machine Learning as Well-posed Learning Problem

- **Tom Mitchell** (1998). **Well-posed Learning Problem:** "A computer program is said to ***learn*** from **experience** *E* with respect to some class of **tasks** *T* and **performance measure** *P*, **if its performance** at tasks in *T*, as measured by *P*, **improves with experience** *E*."

- **Example**: A checkers learning problem:
  - Task ***T***: playing checkers
  - Performance measure ***P***: percent of games won against opponents
  - Training experience ***E***: playing practice games against itself

Figure: *from* Tom Mitchell's homepage

What is Machine Learning?

- The field of **Machine Mearning (ML)** is concerned with the question of **how to construct computer programs that "automatically" improve with experience**
  - i.e., optimize a performance criterion (i.e., *learn*) using example data (training data) or past experience, ideally without much domain-specific expertise
- ML is **a unified *algorithmic framework* designed to identify *computational models*** that accurately describe empirical data and the phenomena underlying it, with little or no human involvement.

Why Machine Learning

- **Data is abundant** and cheap – age of “**big data**”
- **Knowledge is scarce** and expensive
- **Computational power** is available
- **Recent progress in algorithms and theory**
- We need “big theory” to **extract that structure from data for understanding the process, and making predictions for the future**
- Budding **industry** (for applied data science)

Outline

- What Is Machine Learning?
- ☞ **Machine Learning Applications**
- Example of Machine Learning Experiments
- Goal of Machine Learning
- Machine Learning as a Search Problem
- Types of Machine Learning Systems
- Challenges of Machine Learning
- Disciplines of Machine Learning
- Summary

Machine Learning Applications

- **Forecasting task (Regression)**
  - Forecasting your company's revenue next year, based on many performance matrices
- **Segmenting (Clustering)**
  - Segmenting clients based on their purchases so that you can design a different marketing strategy for each segment
- **Recommendation systems**
  - Recommending a product that a client may be interested in, based on past purchases
- **Fraud detection (Anomaly detection)**
  - Detecting credit card fraud
- **Risk assessment**
  - Commercial lending – "lending" or "do not lending"

Machine Learning Applications (cont.)

- **Correlation analysis**
  - Regression in climate science
  - Regression in genetics - How do genes correlate with health factors?
  - Regression in neuroscience - How does neural activity correlate with specific diseases?
- **Causal inference**
  - Causation indicates that one event is the result of the occurrence of the other event
  - Identifying causal effects is an integral part of scientific inquiry, spanning a wide range of questions such as understanding behavior in online systems, effects of social policies, or risk factors for diseases.

Machine Learning Applications (cont.)

- **Document/Text classification**
  - Automatically classifying news articles
  - Sentimental analysis - Automatically flagging offensive comments on discussion forums
- **Text summarization**
  - Summarizing long documents automatically
- **Image classification & Object detection**
  - Analyzing images of products on a production line to automatically classify them
  - Detecting tumors in brain scans
- **Speech recognition**
  - Making your app react to voice commends

Machine Learning Applications (cont.)

- **Robotics & games**
  - Creating a chatbot or a personal assistant
  - Building an intelligent bot for a game, e.g., AlphaGo program at the game of Go
- **Visualization**
  - Representing a complex, high-dimensional dataset in a clear and insightful diagram

Outline

- What Is Machine Learning?
- Machine Learning Applications
- ☞ **Machine Learning Experiment**
- Goal of Machine Learning
- Machine Learning as a Search Problem
- Types of Machine Learning Systems
- Challenges of Machine Learning
- Disciplines of Machine Learning
- Summary

Toy Example: Distinguish Cats from Dogs

- **Task**: Teaching a computer how to distinguish between pictures of *cats* from those with *dogs*.

- How we first learned about the difference between cats and dogs?   - Simple cognitive tasks
  - We are naturally presented with many images of what they are told by a *supervisor* (e.g., a parent) are either cats or dogs, until we fully grasp the two concepts
  - When we encounter new (images of) cats and dogs, and can correctly identify each new example, we can generalize what we have learned to new, previously unseen, examples.

1. Gather Data

- Like human beings, a computer must be trained to recognize the difference between cat and dog by learning from a batch of examples, typically referred to as a ***training set*** of data.

- **Data collection**

  - The training dataset is used to train a machine learning model that can distinguish between future images of cats and dogs.

2. Design Feature Space

- **Feature Design**
  - We do not just look at an image, instead, examine features to distinguish cat and dogs.
  - Color, size, the shape of the ears or nose, and/or some combination of these features to distinguish between the two.
- Designing quality features for **feature space** is typically not a trivial task as it can be very application dependent.

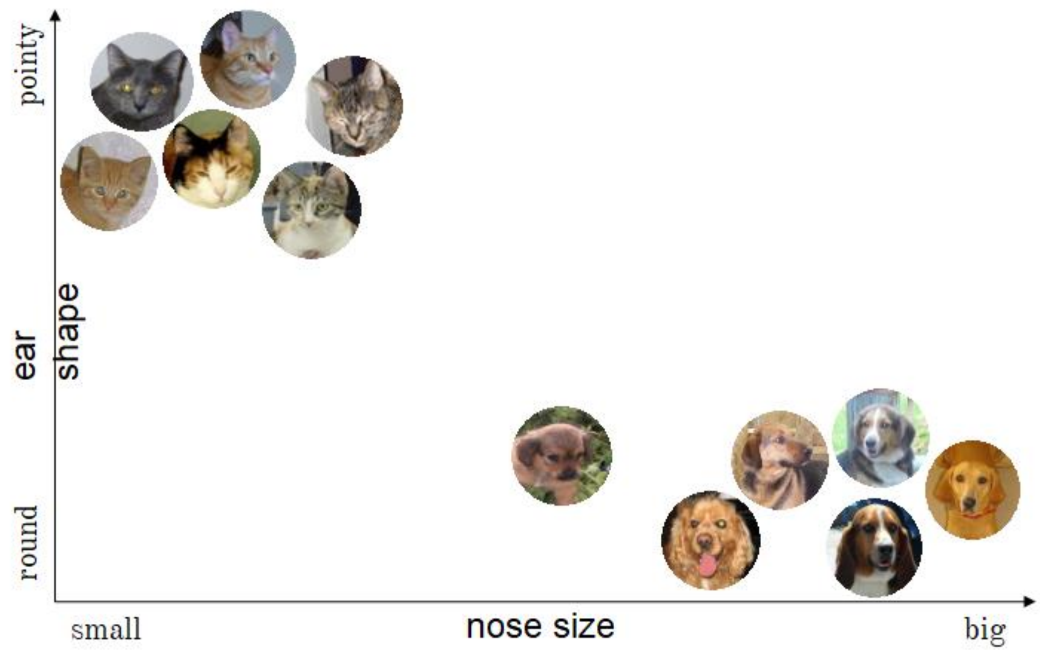
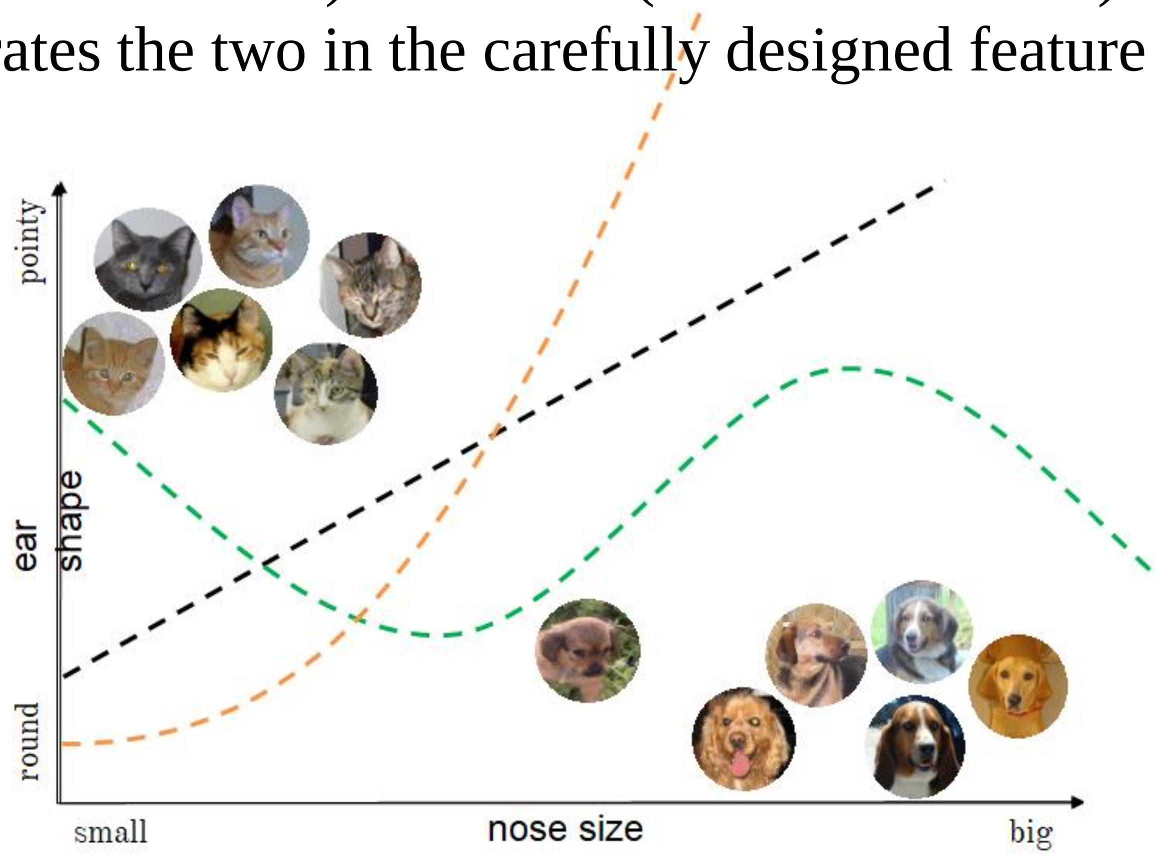


Figure: The feature space is now represented by two numbers of noise size and ear shape.

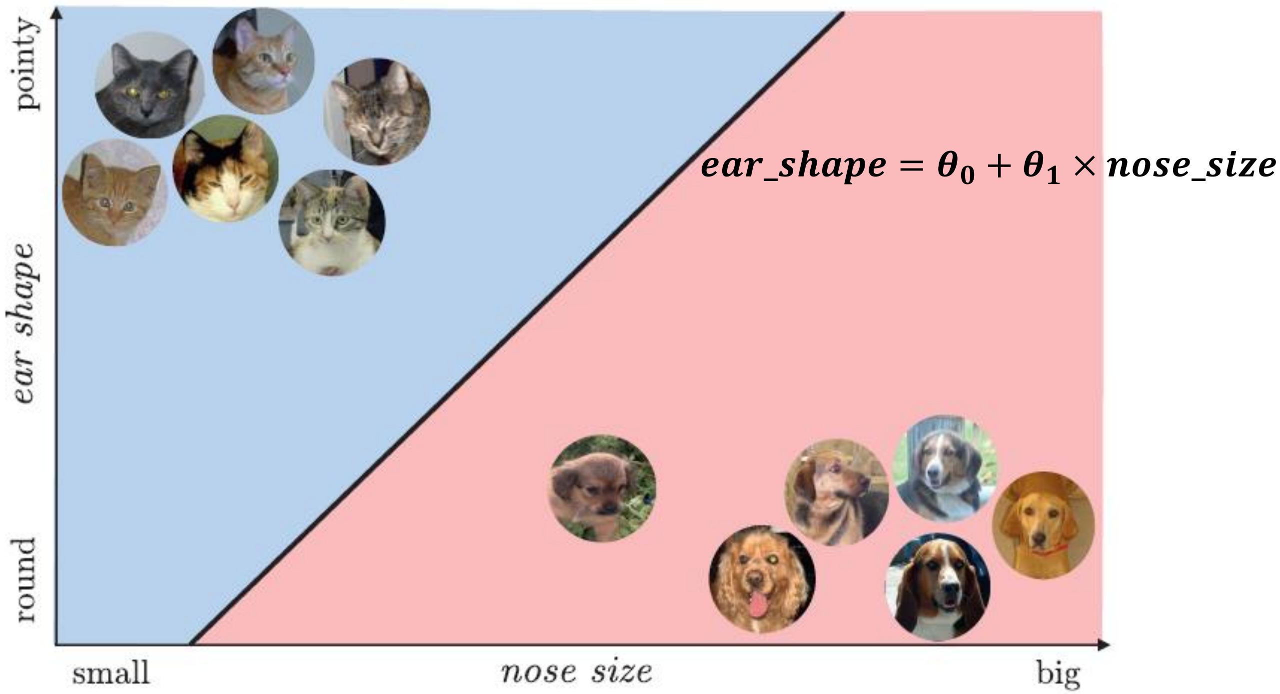
3. Choose a Model

- **Model selection** - How the machine will make distinction
  - Suppose a simple geometric problem – ask the machine find a line (*linear model*) or a curve (*non-linear model*) that separates the two in the carefully designed feature space.



4. Train the Model

- **Model training**
  - Suppose a line model. The machine needs to find the right values for its two parameters – a slope and vertical intercept
  - The process of determining proper parameters for a model relies on a set of tools known as ***mathematical optimization***

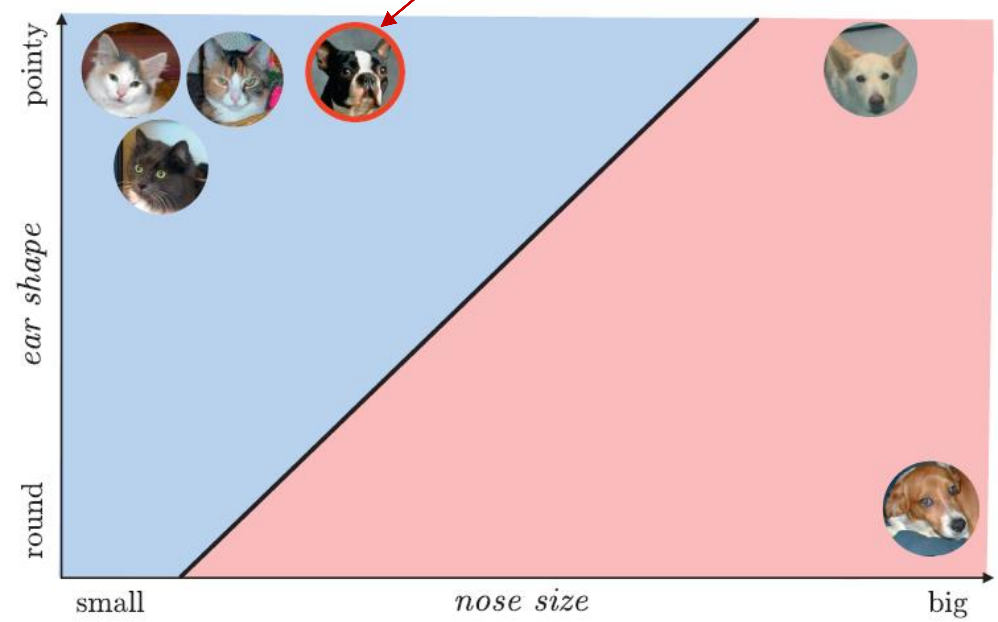


5. Test the model (on new data)

- **Model validation**
  - To validate the efficacy of our trained learner (i.e., model), we now show the computer a batch of previously unseen images of cats and dogs, and see how well it can identify the animal in each image.

A **test set** of data

A wrong prediction



Outline

- What Is Machine Learning?
- Machine Learning Applications
- Machine Learning Experiment
- ☞ **Goal of Machine Learning Goal**
- Machine Learning as a Search Problem
- Types of Machine Learning Systems
- Challenges of Machine Learning
- Disciplines of Machine Learning
- Summary

Predictive Data Analytics

- **Machine learning models** may be ***predictive*** to make predictions in the future, or ***descriptive*** to gain knowledge from data, or both.
- **Predictive data analysis**, a subfield of data analytics, is the art of building and using *models* that make predictions based on patterns extracted from historical data.
- The word "***prediction***" is used with a broad definition as well as has a temporal aspect. A prediction is the assignment of a value to any unknown variable. e.g.,
  - Predict the price that something will be sold for in the future
  - Predict the type of document

ML for Predictive Data Analytics

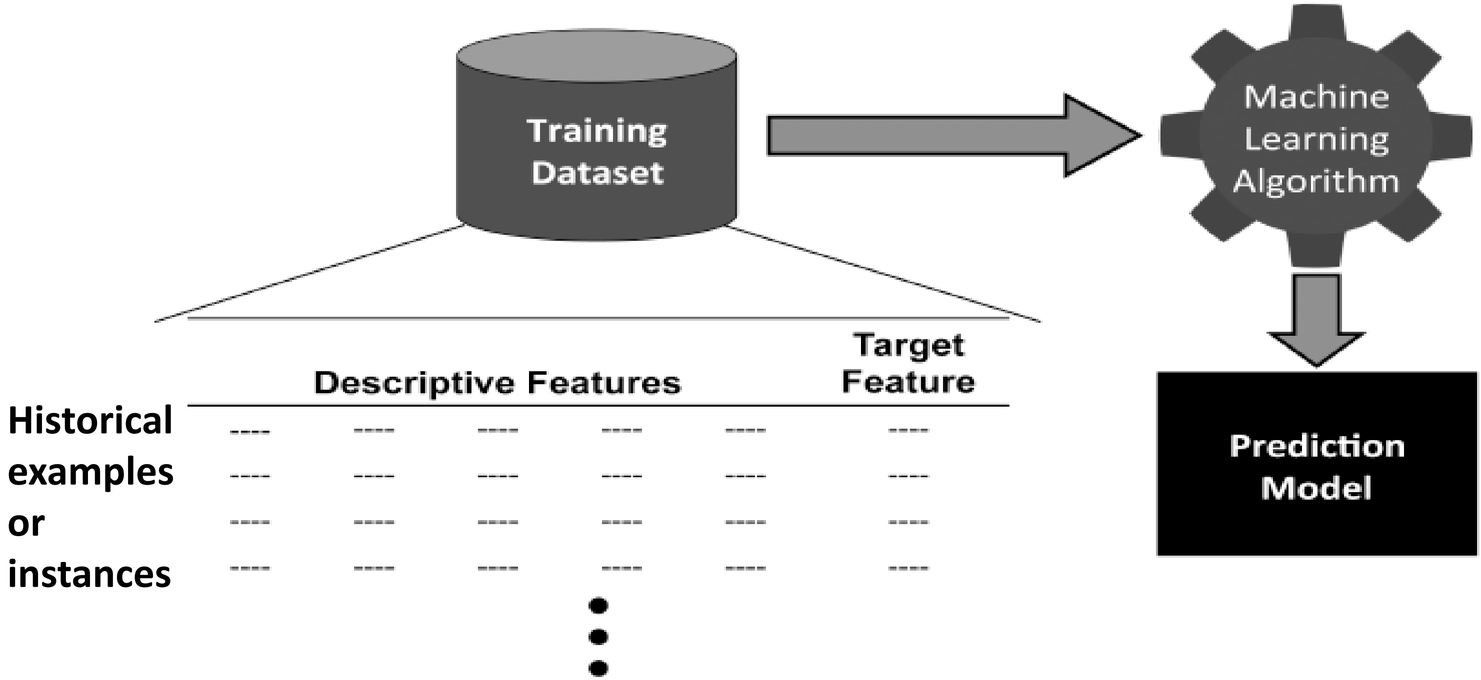


Figure. Using machine learning to induce a prediction model from a training dataset.

Figure. Using the model to make predictions for new instances.

Example I: Training Dataset

descriptive features — target feature

training dataset

ID	OCCUPATION	AGE	LOAN-SALARY RATIO	OUTCOME
1	industrial	34	2.96	repaid
2	professional	41	4.64	default
3	professional	36	3.22	default
4	professional	41	3.11	default
5	industrial	48	3.80	default
6	industrial	61	2.52	repaid
7	professional	37	1.50	repaid
8	professional	40	1.93	repaid
9	industrial	33	5.25	default
10	industrial	32	4.15	default

training examples

Table. A set of historic instances of mortgages that a bank has granted in the past. The *target feature* (OUTCOME) indicates whether the mortgage applicant ultimately defaulted on the load or paid it back in full.

- **Task**: Predict the value of OUTCOME for an arbitrary mortgage applicant, based on the values of its other features.

Example I: Prediction Model

- What is the relationship between the descriptive features (OCCUPATION, AGE, LOAN-SALARY RATIO) and the target feature (OUTCOME)?

- An example of **prediction model**

if LOAN-SALARY RATIO > 3 **then**
  OUTCOME=*'default'*
else
  OUTCOME=*'repay'*
end if

* Notice that this model does not use all the features

Example II: A More Complex Training Data Set

ID	Amount	Salary	Loan-Salary Ratio	Age	Occupation	House	Type	Outcome
1	245,100	66,400	3.69	44	industrial	farm	stb	repaid
2	90,600	75,300	1.2	41	industrial	farm	stb	repaid
3	195,600	52,100	3.75	37	industrial	farm	ftb	default
4	157,800	67,600	2.33	44	industrial	apartment	ftb	repaid
5	150,800	35,800	4.21	39	professional	apartment	stb	default
6	133,000	45,300	2.94	29	industrial	farm	ftb	default
7	193,100	73,200	2.64	38	professional	house	ftb	repaid
8	215,000	77,600	2.77	17	professional	farm	ftb	repaid
9	83,000	62,500	1.33	30	professional	house	ftb	repaid
10	186,100	49,200	3.78	30	industrial	house	ftb	default
11	161,500	53,300	3.03	28	professional	apartment	stb	repaid
12	157,400	63,900	2.46	30	professional	farm	stb	repaid
13	210,000	54,200	3.87	43	professional	apartment	ftb	repaid
14	209,700	53,000	3.96	39	industrial	farm	ftb	default
15	143,200	65,300	2.19	32	industrial	apartment	ftb	default
16	203,000	64,400	3.15	44	industrial	farm	ftb	repaid
17	247,800	63,800	3.88	46	industrial	house	stb	repaid
18	162,700	77,400	2.1	37	professional	house	ftb	repaid
19	213,300	61,100	3.49	21	industrial	apartment	ftb	default
20	284,100	32,300	8.8	51	industrial	farm	ftb	default
21	154,000	48,900	3.15	49	professional	house	stb	repaid
22	112,800	79,700	1.42	41	professional	house	ftb	repaid
23	252,000	59,700	4.22	27	professional	house	stb	default
24	175,200	39,900	4.39	37	professional	apartment	stb	default
25	149,700	58,600	2.55	35	industrial	farm	stb	default

Table. A more complex credit scoring dataset

Example II: Prediction Model

- An example of **prediction model**

if LOAN-SALARY RATIO < 1.5 **then**
  OUTCOME=*'repay'*
else if LOAN-SALARY RATIO > 4 **then**
  OUTCOME=*'default'*
else if AGE < 40 **and** OCCUPATION $=$*'industrial'* **then**
  OUTCOME=*'default'*
else
  OUTCOME=*'repay'*
end if

- When the dataset is large with multiple features, to manually learning a model by examining the data is almost impossible

Consistent Model

- If a model makes a correct prediction for all instances in a training set, the model is called a ***consistent model***
- **Consistency ≈ memorizing the training dataset**
- **Example.** A model built from the simple mortgage data is a consistent model.

    If LOAN-SALARY RATIO > 3 **then** OUTCOME=‘default’

    else OUTCOME =‘replay’ **end if**

- Although models agree on which predictions should be made for examples in the training dataset, they might disagree with regard to which predictions should be returned for new instances that are not in the training dataset.
- Moreover, consistency with noise in the data isn’t desirable.

Goal of Machine Learning

- The **Goal** of machine learning is **to find a model that generalizes beyond the training dataset** and that isn't influenced by the noise in the dataset.

Outline

- What Is Machine Learning?
- Machine Learning Applications
- Machine Learning Experiments
- Goal of Machine Learning
- ☞ **Machine Learning as a Search Problem**
- Types of Machine Learning Systems
- Challenges of Machine Learning
- Disciplines of Machine Learning
- Summary

ML as Search Problem

- The problem of **inducing *general functions*** (i.e., ***models***) **from specific training examples** is central to ***learning***.
- ***Learning*** can be formulated as **a problem of *search*** through a predefined space of potential hypotheses for the hypothesis that best fits the training examples

ML Problem Formulation as Search

- **Given**
  - **Training examples X** : positive and negative examples over which the target function is defined.

    $\langle x_1, c(x_1)\rangle, \ldots, \langle x_m, c(x_m)\rangle,$

    - Instances $X = \{x_1, \ldots, x_m\}$, each x_i described by its attributes, e.g., *Occupation*, *Age*, and *Loan-Salary Ratio*
    - Target concept, c: $X \rightarrow \{0, 1\}$, e.g., where 0 means *repaid* and 1 *default* when *c* is *Outcome*
  - **Hypotheses *H*** *:* a set of all possible hypotheses,
    Each hypothesis *h* in *H* represents a boolean-valued function defined over *X*, i.e., ***h*****:** ***X*** $\boldsymbol{\rightarrow \{0, 1\}}$
- **Find a** (consistent) hypothesis *h* in *H* such that *h(x)=c(x)* for all *x* in X

Inductive Learning Hypothesis

- The only information available about the target concept *c* is its value over the training examples.
- **Inductive learning** algorithms can at best guarantee that the output hypothesis (output model) fits the target concept over the training data.
- Further **we assume** that any hypothesis (called ***inductive learning hypothesis***) found to approximate the target function (concept) well over a sufficiently large set of training examples <u>will also approximate the target function well over other unobserved examples</u>

Example: Hypothesis Space

- **Task**: Classify customer households into the demographic groups (GRP) - single, couple or family, solely on the basis of their shopping habits
- A training data with labels

Table: A simple retail dataset

ID	BBY	ALC	ORG	GRP
1	no	no	no	couple
2	yes	no	yes	family
3	yes	yes	no	family
4	no	no	yes	couple
5	no	yes	yes	single

  - Descriptive features: buy baby food (BBY), alcohol (ALC), organic vegetable products (ORG)
  - Target figure: GRP

Example (cont.)

- **How many different possible models exists?**
  - 3 binary descriptive features, $2^3 = 8$ possible combinations
  - For each of these 8 possible combinations of descriptive feature values, there are 3 possible target feature values
  - $3^8 = 6561$ possible prediction models, $\mathbb{M}_1$ to $\mathbb{M}_{6561}$

BBY	ALC	ORG	GRP	$\mathbb{M}_1$	$\mathbb{M}_2$	$\mathbb{M}_3$	$\mathbb{M}_4$	$\mathbb{M}_5$	...	$\mathbb{M}_{6\,561}$
no	no	no	?	couple	couple	single	couple	couple		couple
no	no	yes	?	single	couple	single	couple	couple		single
no	yes	no	?	family	family	single	single	single		family
no	yes	yes	?	single	single	single	single	single		couple
yes	no	no	?	couple	couple	family	family	family	...	family
yes	no	yes	?	couple	family	family	family	family		couple
yes	yes	no	?	single	family	family	family	family		single
yes	yes	yes	?	single	single	family	family	couple		family

Table: A full set of potential prediction models before any training data becomes available.

Example (cont.)

- Using the training dataset, a machine learning algorithm might reduce the full set of possible prediction models to only those that are *consistent* with the training instances

BBY	ALC	ORG	GRP	$\mathbb{M}_1$	$\mathbb{M}_2$	$\mathbb{M}_3$	$\mathbb{M}_4$	$\mathbb{M}_5$	...	$\mathbb{M}_{6\,561}$
no	no	no	couple	couple	couple	single	couple	couple		couple
no	no	yes	couple	single	couple	single	couple	couple		single
no	yes	no	?	family	family	single	single	single		family
no	yes	yes	single	single	single	single	single	single		couple
yes	no	no	?	couple	couple	family	family	family	...	family
yes	no	yes	family	couple	family	family	family	family		couple
yes	yes	no	family	single	family	family	family	family		single
yes	yes	yes	?	single	single	family	family	couple		family

Table: A sample of the models that are consistent with the training data

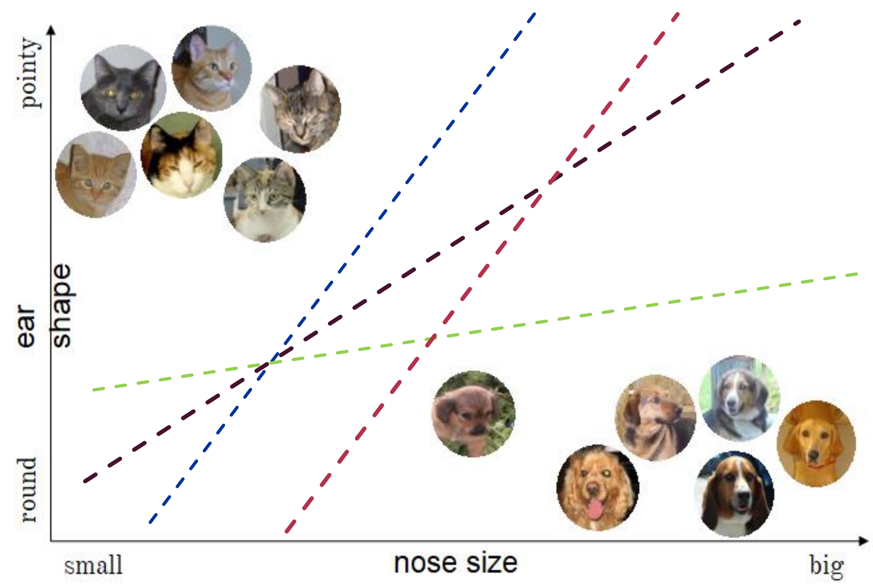
- Notice that there is more than one candidate model left!
  - It is because a single consistent model cannot be found based on a given sample training dataset
- Thus ML is fundamentally an **ill-posed problem**.

Machine Learning as Search

- Machine Learning is an **ill-posed problem** – a problem for which a unique solution cannot be determined using only the information that is available.
- Again, hypothesis space is very large or infinite in most practical learning problems.
- We need **machine algorithms** capable of efficiently searching very large or infinite hypothesis space, to find the hypotheses that <u>best fit</u> the training data.

Machine Learning as Search

- **How do we train a machine properly? <u>How to search a model best fit?</u>**
- A *model* is defined up to some parameters
  - E.g., the slope and intercept of the right separate line between the two classes, cat and dog.



- *Learning* is the execution of a computer program to optimize the parameters of the model (e.g., for line fitting, slope and intercept parameters) using example data
- We can **formalize the search for parameters of a learning model** via well-defined ***mathematical functions***.

Search and Optimization

- The mathematical functions, commonly referred to as ***cost*** or ***loss functions***, take in a specific set of model parameters and return a score indicating how well we would accomplish a given learning task using that choice of parameters.
- The *lower* the score of the cost function the better the machine performs, so we aim to ***minimize* these cost functions.**
  - Machine learning algorithms are guided by the objective of minimizing error and maximizing accuracy.
- This is achieved **through processes such as optimization**, where model parameters are adjusted to fit the data - Training the model requires we *optimize* an associated *cost function*

Example: ML as Optimization Problem

The linear classifiers corresponding to each set of parameters

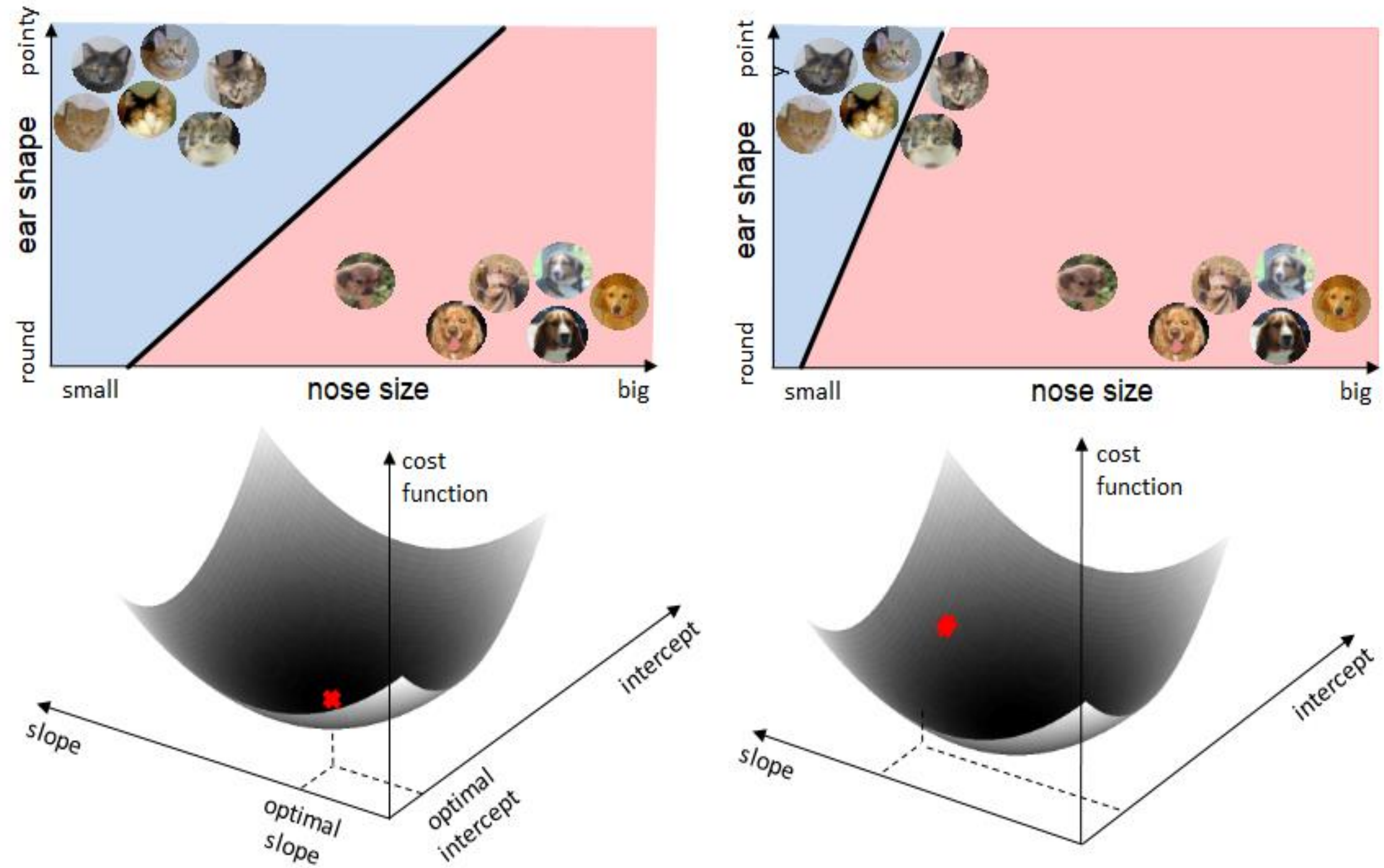


Figure. The figure draws **the two-dimensional cost function** associated with learning the **slope** and **intercept parameters** of **a line model** separating two classes of the toy dataset.
One (left) corresponding to the minimum of the cost function and the other (right) corresponding to a point with larger cost function value. The optimal set of parameters, i.e., **those giving the minimum value of the associated cost function,** allows for the best separation between the two classes.

Inductive Bias

- All the different model section criteria consist of a set of assumption about the characteristics of the model that we would like to induce.
- The set of assumptions that defines the model selection criteria of a machine learning algorithm is known as the **inductive bias** (also known **learning bias**) of the machine learning algorithm

Inductive Bias

- There are two types of inductive bias
  - A **restriction bias** constraints the set of models that the algorithm will consider during the learning process
    - E.g., Multivariable linear regression with gradient decent considers only prediction models that produce prediction on the basis of a linear combination of the descriptive feature values
  - A **preference bias** guides the learning algorithm to prefer certain models over others.
    - E.g., ID3 decision tree algorithm prefers shallower (less complex) trees over larger tree.

Sample Bias

- Inductive bias is not the only type of bias that affects machine learning
- **Sample bias** arises when the sample of data used within a data-driven process is collected in such a way that the sample is not representative of the population the sample is used to represent.
- If a sample of data is not representative of a population, then <u>inferences based on that sample will not generalize to the larger population</u>, no matter how large the sample is.
- Sampling bias is a difficult problem to tackle because it can arise in indirect and non-obvious ways.

Outline

- What Is Machine Learning?
- Machine Learning Applications
- Machine Learning Experiment
- Goal of Machine Learning
- Machine Learning as a Search Problem
- ☞ **Types of Machine Learning Systems**
- Challenges of Machine Learning
- Disciplines of Machine Learning
- Summary

Most Common Taxonomy of Machine Learning

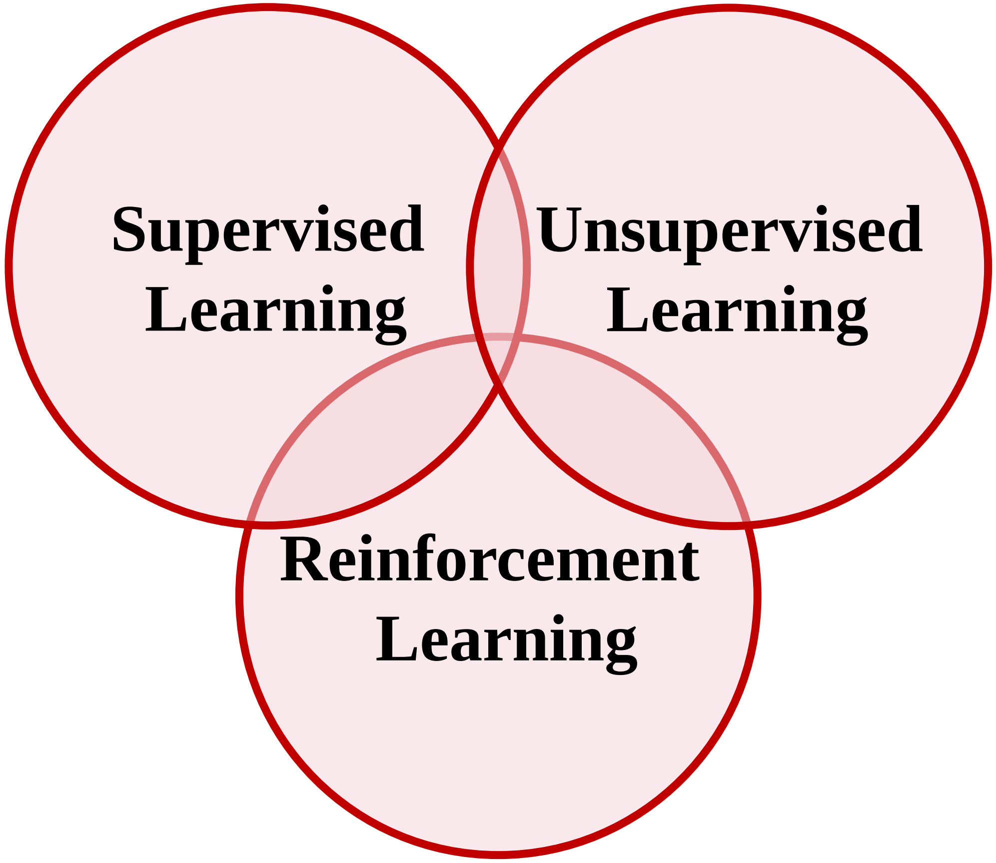


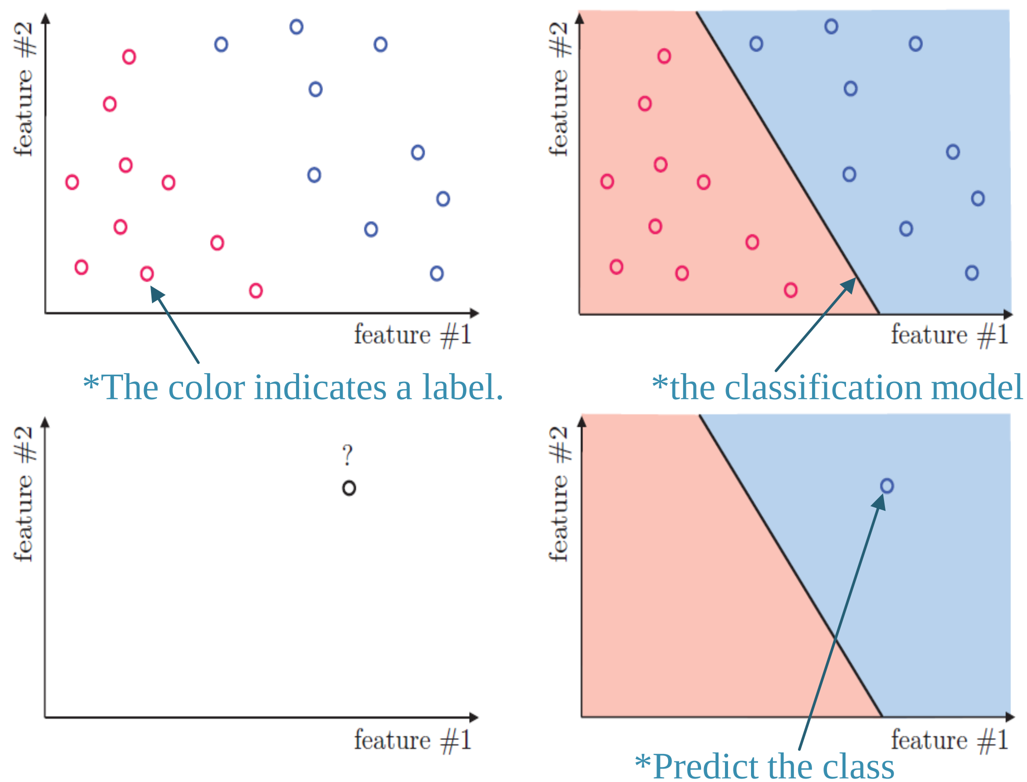
Figure. A common taxonomy based on learning paradigms

Types of Machine Learning Systems

- According to the amount and type of supervision ML systems get during training, they can be categorized to **Supervised**, **Unsupervised**, **Semi-supervised**, and **Self-supervised learning**, and others such as **Reinforcement learning**, and **Transfer learning**
- Whether or not they can learn incrementally on the fly – **Online leaning** vs **Batch learning**
- Whether they work by simply comparing new data points to known data points, or instead by detecting patterns in the training data and building a predictive model – **Instance-based learning** vs **Model-based learning**

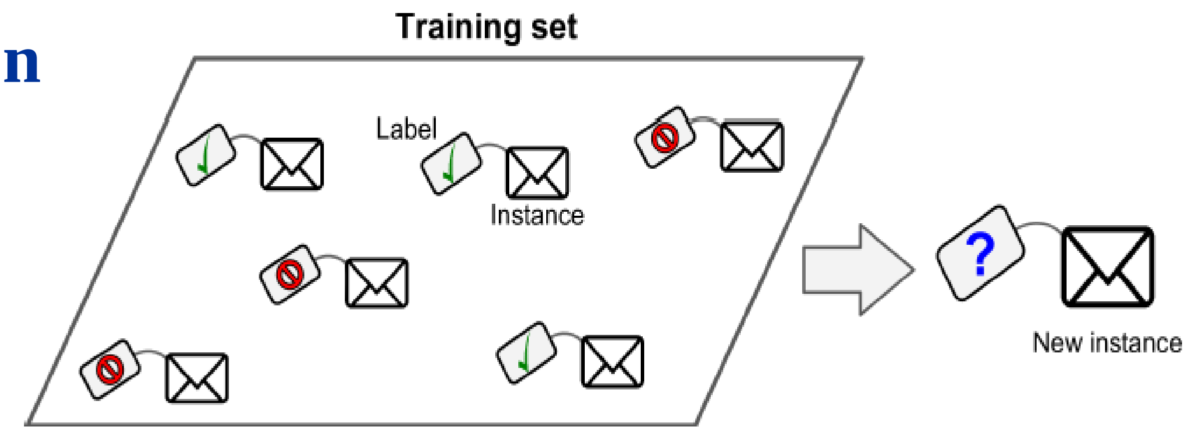
Supervised Learning - Classification

- **Supervised learning** is a machine learning approach that's defined by its **use of labeled datasets**.
  - The training dataset (example data) you feed to the ML algorithm includes the desired solutions, called ***labels***.
  - The dataset is prepared to train ML algorithms into predicting outcomes accurately
- A typical supervised learning task is ***classification*** to predict **a target *class (discrete value)***.

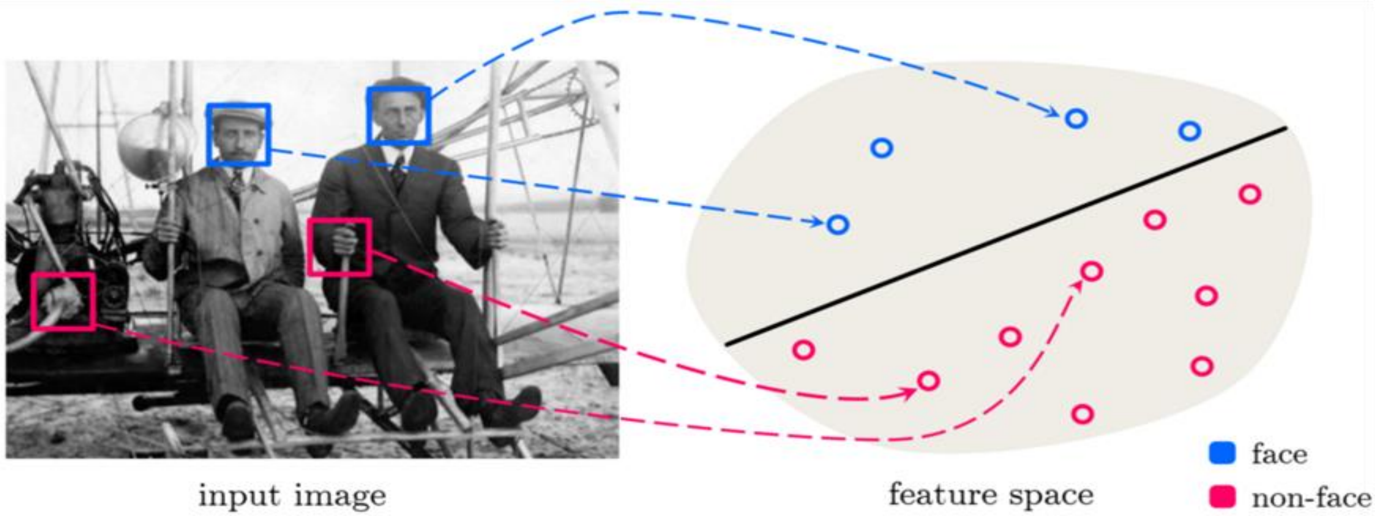


Two-Class Classification Problems

- **Spam detection**

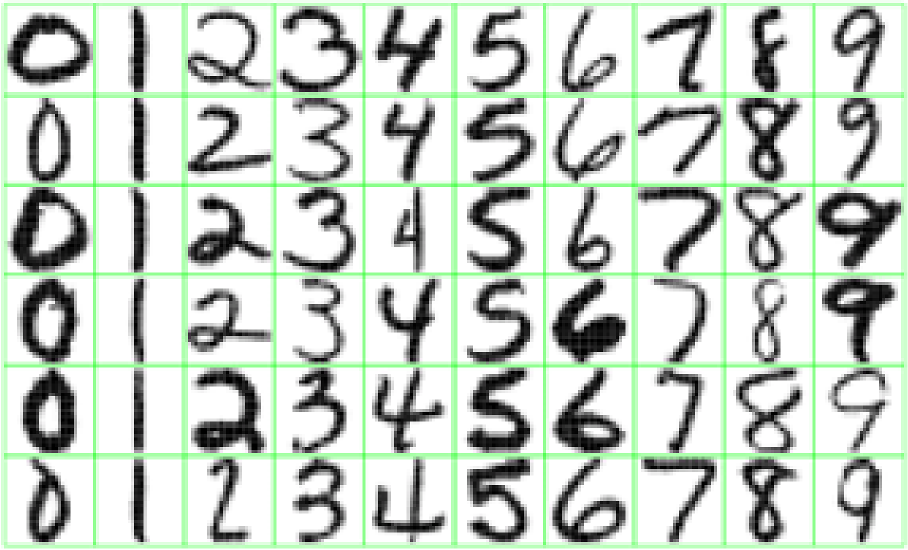


- **Object detection** – Face recognition



Multi-Class Classification Problems

- **Recognition problems with multi classes** where the aim is to train a classifier to automatically distinguish between a collection of things.
  - Human gestures
  - Various visual objects
  - Spoken words
- E.g., Handwritten digit recognition
  - commonly built into software of mobile banking applications



Supervised Learning - Regression

- Another typical task of supervised learning is **regression** to predict **a target *numeric* value**.

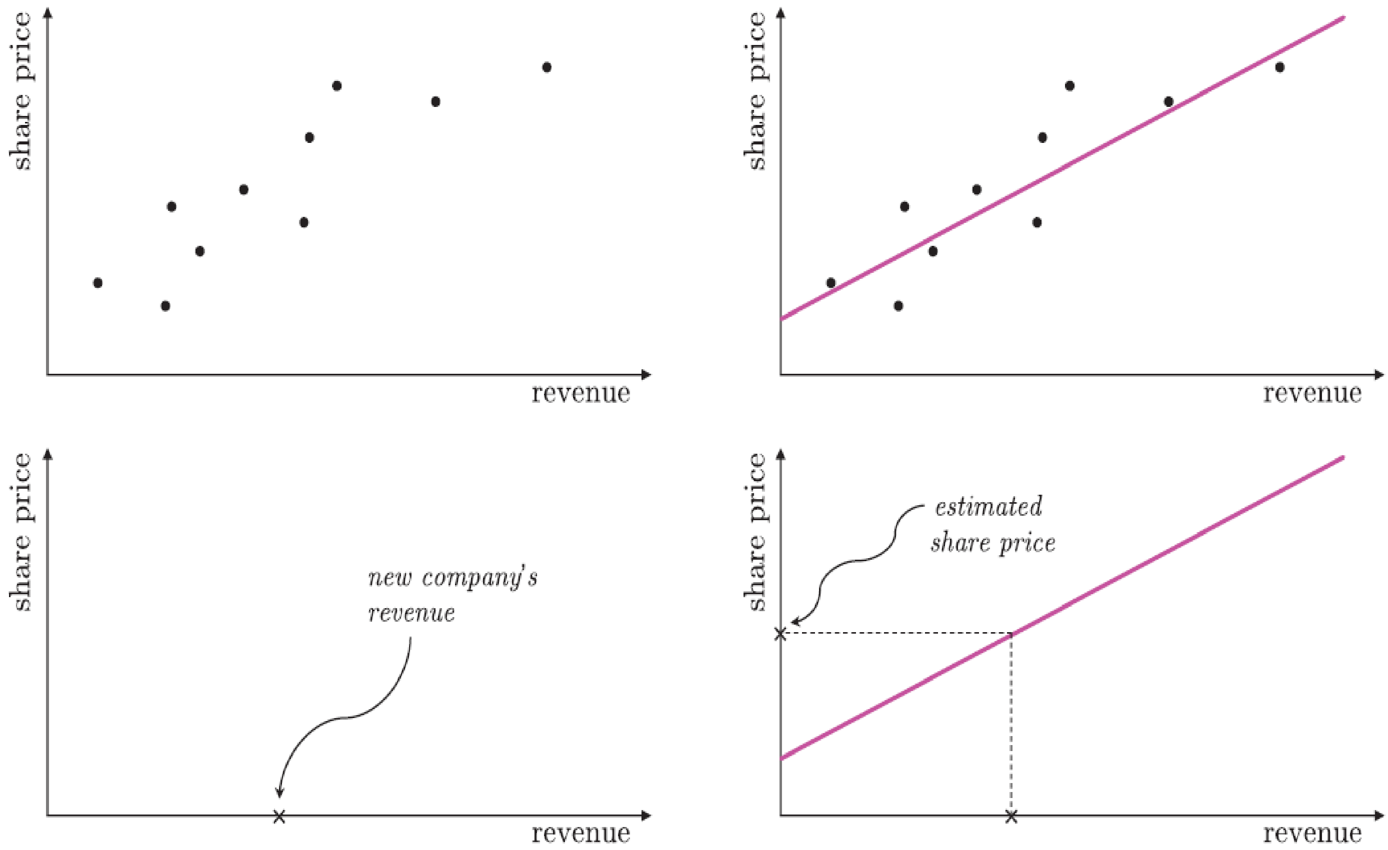


Fig. A *regression* problem: predict a value, given an input feature (usually multiple input features)

Unsupervised Learning

- In **unsupervised learning**, the training data is ***unlabeled***.
  - The system tries to learn without a teacher
- A typical unsupervised learning task is ***clustering*** to detect groups of similar objects

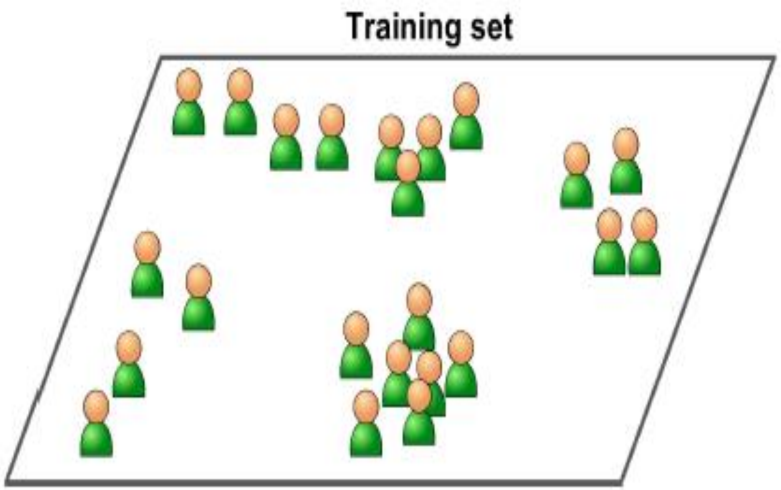
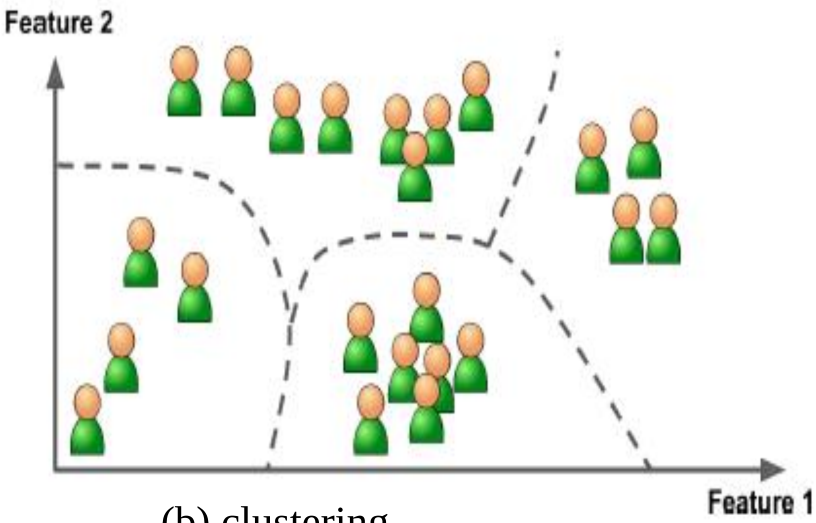


Fig. (a) An unlabeled training set



(b) clustering

Unsupervised Learning Tasks

- **Clustering**
  - Group objects based on their similarity
- **Anomaly detection** and **novelty detection**
  - Find anomaly (outliers) different from normal instances, e.g., detect unusual credit card transactions
- **Association rule mining**
  - Discover interesting relationship patterns from a large transaction dataset
- **Dimension reduction**
  - Simplify the data without losing too much information.
- **Visualization**
  - Given a lot of complex and unlabeled data, visualization algorithms output a 2D or 3D representation of the input data that can easily be plotted.

Examples

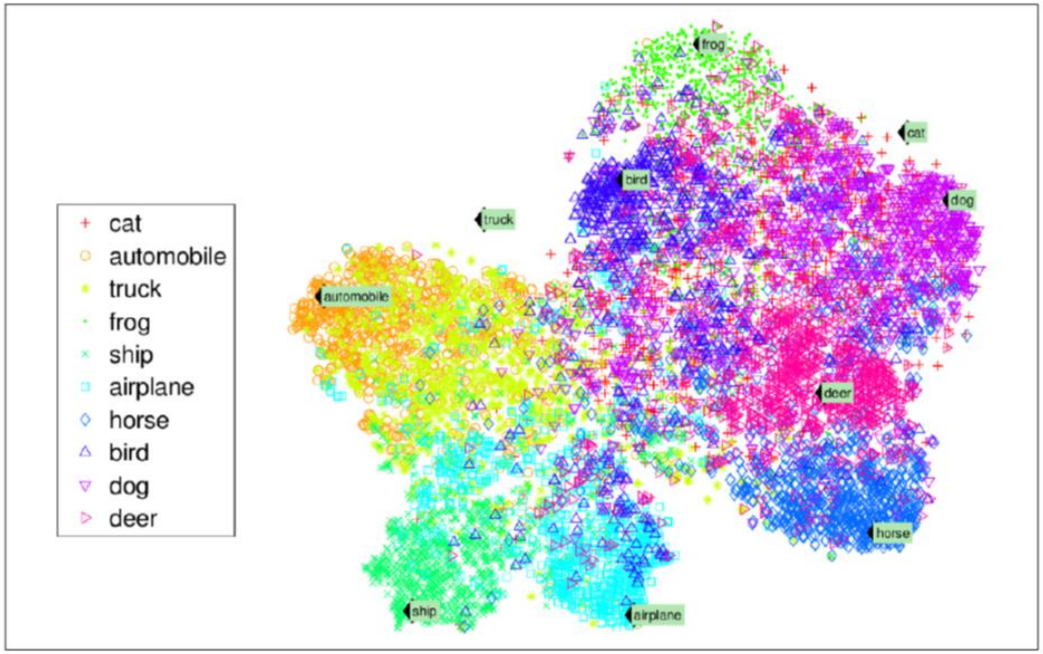


Fig. Visualization highlighting semantic clusters

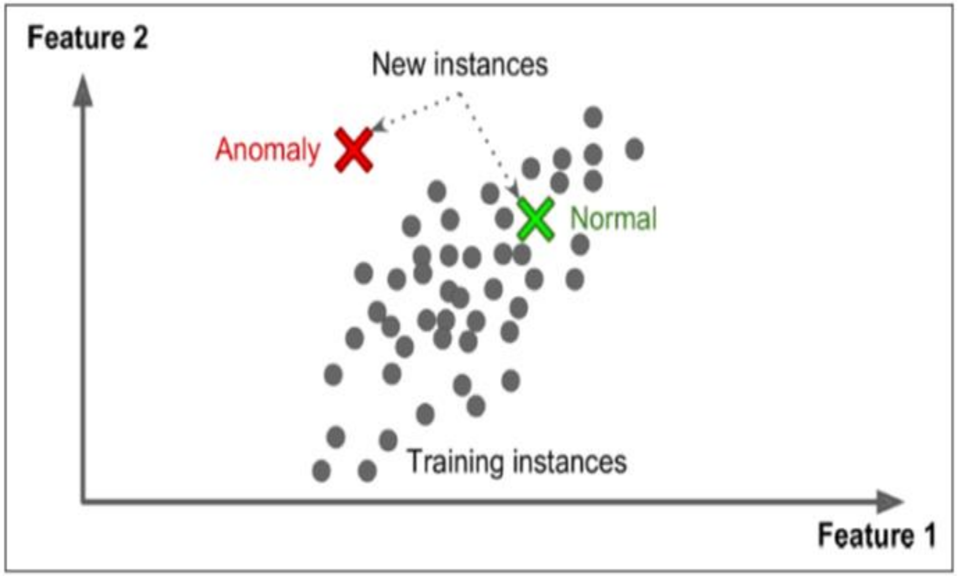


Fig. Anomaly detection

Semi-supervised Learning

- Labeling data is usually time-consuming and costly.
- Semisupervised learning algorithms **deal with data that's partially labeled.**
- **Semi-supervised learning** is a learning problem that involves a small number of labeled examples and a large number of unlabeled examples

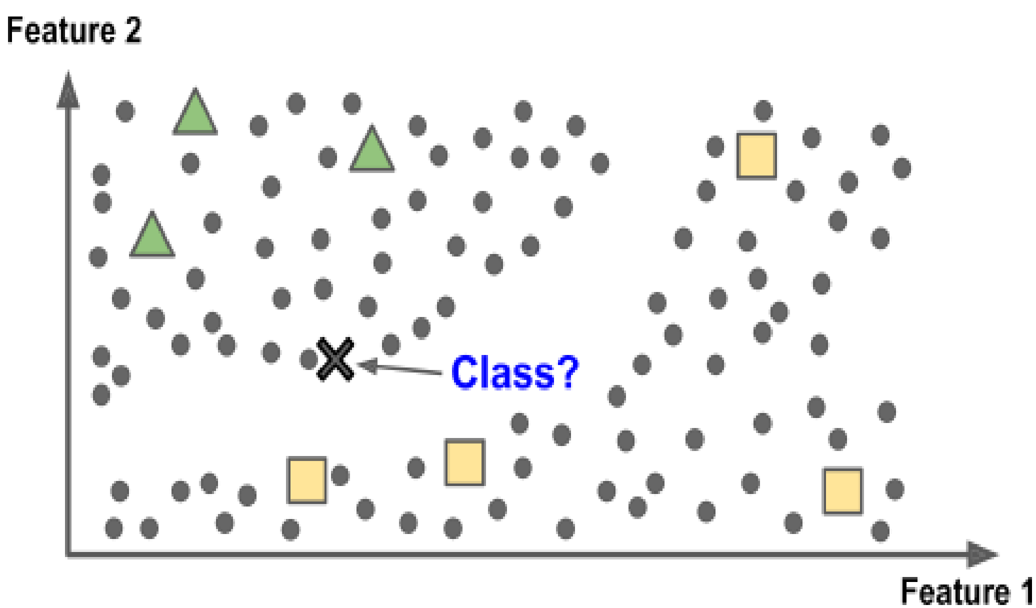


Fig. Semisupervised learning with two classes (triangles and squares); the unlabeled examples (circles) help classify a new instance(the cross) into the triangle class rather than the square class, even though it is closer to the babbled squares.

Semi-supervised Learning

- Most semi-supervised learning algorithms are combinations of unsupervised and supervised algorithms.
  - For example, a clustering algorithm may be used to group similar instances together, and then every unlabeled instance can be labeled with the most common label in its cluster.
  - Once the whole dataset is labeled, it is possible to use any supervised learning algorithm.
- **Application example**: Photo-hosting services (Google Photos)
  - When you just add one label per person in pictures, semi-supervised learning approach is able to name everyone in every photo.

Self-supervised Learning

- **Self-supervised learning** is a machine learning approach which involves actually generating a fully labeled dataset from a fully unlabeled one.
  - Once the whole dataset is labeled, any supervised learning algorithm can be used.
- Unlike traditional supervised learning, self-supervised learning relies on generating supervisory signals from the data itself.
- A model is learned to make predictions about a transformed or modified version of its input data.

Self-supervised Learning

- **Example: Image Inpainting**
  - If you have a large dataset of unlabeled images, you can randomly mask a small part of each image and then train a model to recover the original image
  - The resulting model may be used to repair damaged images or to erase unwanted objects from pictures.

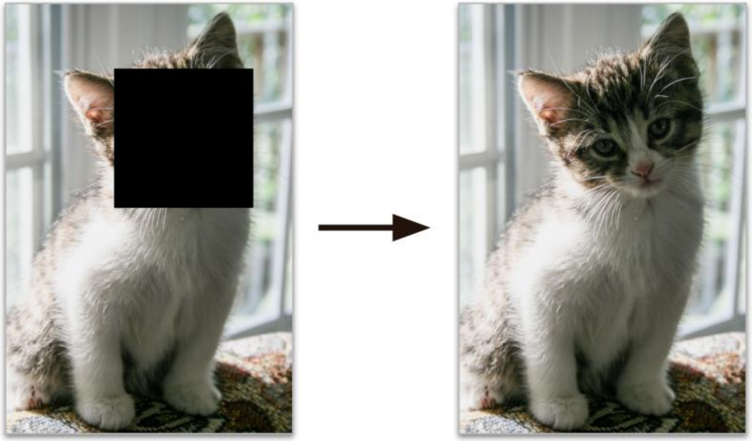


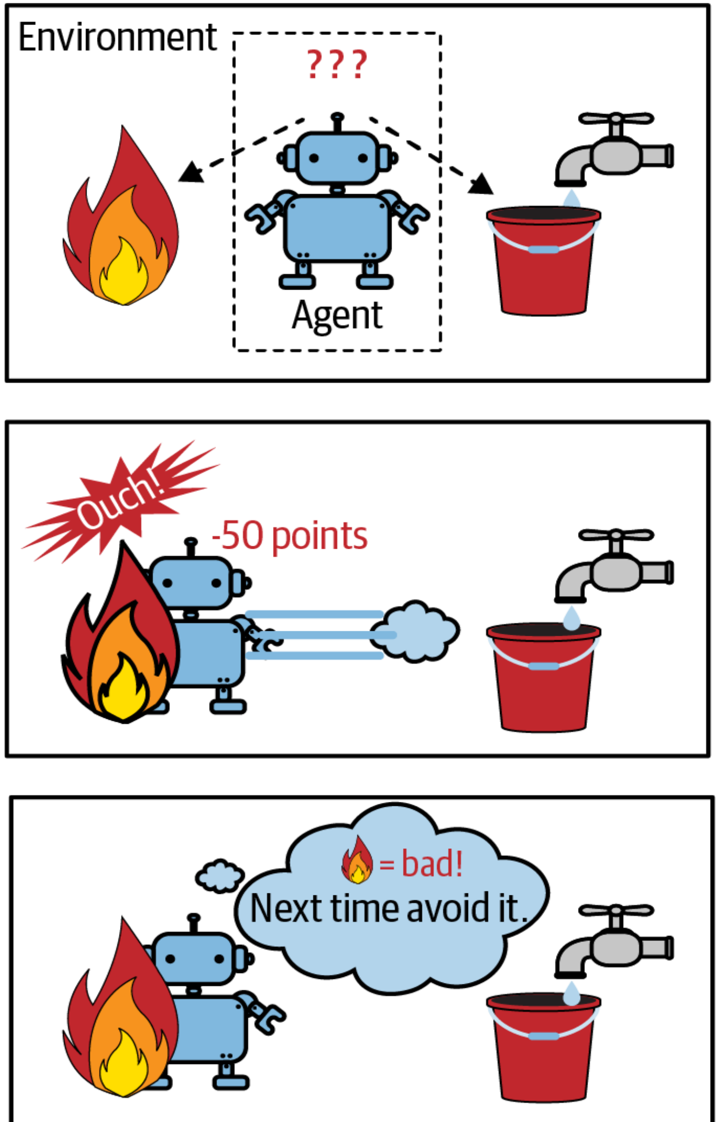
Figure. Self-supervised learning example: input (left) and target (right)

- **Example: Context Prediction**
  - In the context of natural language processing, the model is trained to predict missing words in a sentence.

Reinforcement Learning

- In **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.
- **Examples**:
  - Many robots implement reinforcement learning algorithms to learn how to walk.
  - A game program (e.g., DeepMind's AlphaGo program) learns its winning policy by analyzing millions of games, and then playing many games against itself.

Example: Reinforcement Learning



1. Observe
2. Select action using policy
3. Action!
4. Get reward or penalty
5. Update policy (learning step)
6. Iterate until an optimal policy is found

Batch vs Online Learning

- In **batch learning**, the system must be trained using all the available data. It is incapable of learning incrementally
- In **online learning**, the system can be trained by feeding it data instances sequentially, either individually or mini-batches.

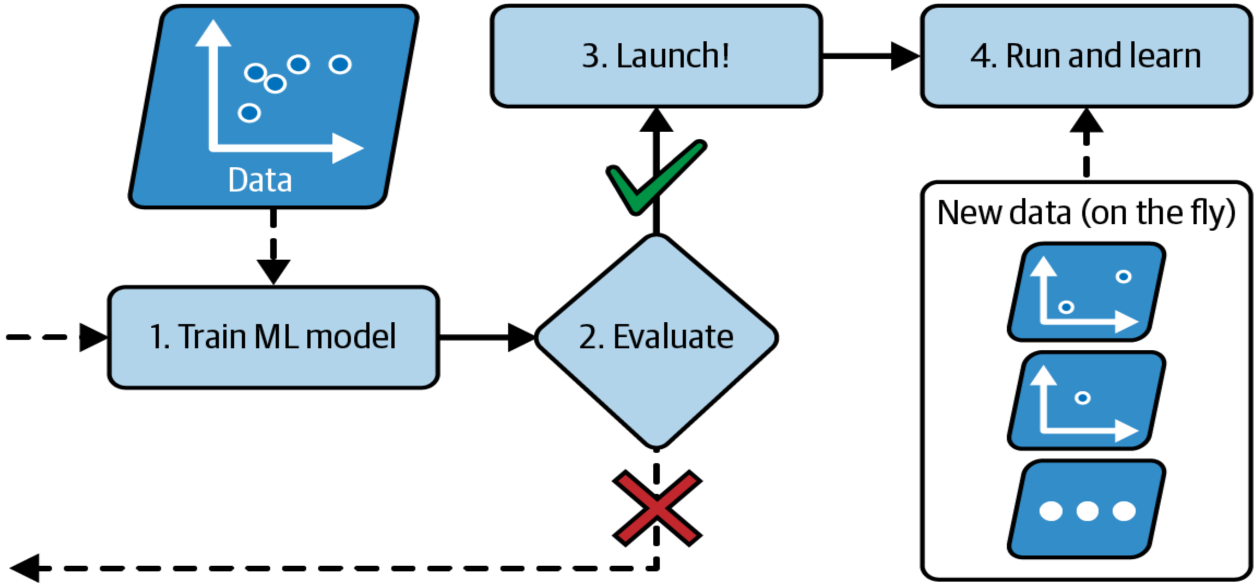


Figure. In online learning, a model is trained and launched into production, and then it keeps learning as new data comes in.

Model-based vs Instance-based Learning

- In **model-based learning**, first build a model of example data and then use that model to make prediction of new cases.
- In **instance-based learning**, new cases are generalized by using a similarity measure to compare them to the learned examples.

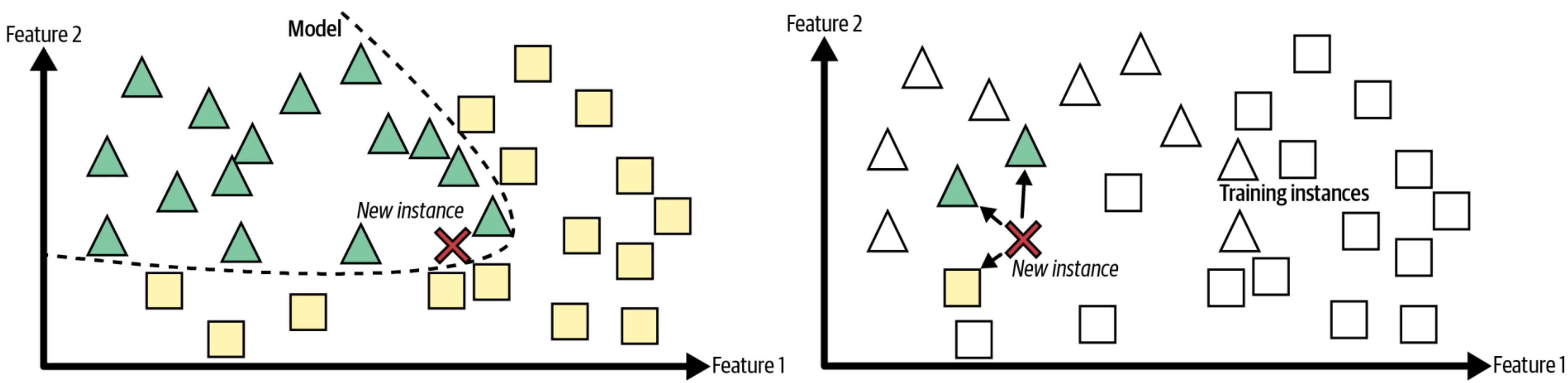


Figure (a) model-based learning (b) instance-based learning

Outline

- What Is Machine Learning?
- Machine Learning Applications
- Machine Learning Experiment
- Goal of Machine Learning
- Machine Learning as a Search Problem
- Types of Machine Learning Systems
- ☞ **Challenges of Machine Learning**
- Disciplines of Machine Learning
- Summary

Challenges of Machine Learning

- **Insufficient quantity of training data**
  - Even for very simple problems thousands of examples are typically needed.
  - For complex problems such as image or speech recognition you may need millions of examples.

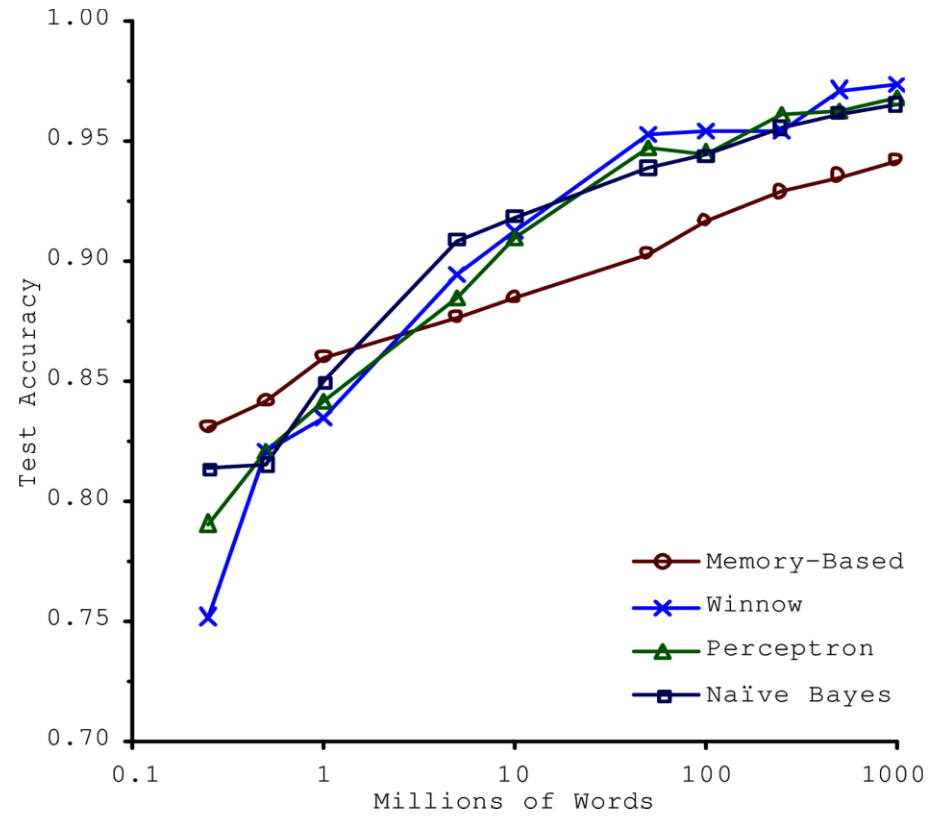


Figure. The importance of data versus algorithms

Challenges of Machine Learning

- **Nonrepresentative training data**
  - In order to generalize well, it is crucial that your training data be representative of the cases you want to generalize to.
  - Even a very large samples can be nonrepresentative if the sampling method is flawed. (called ***sampling bias***)

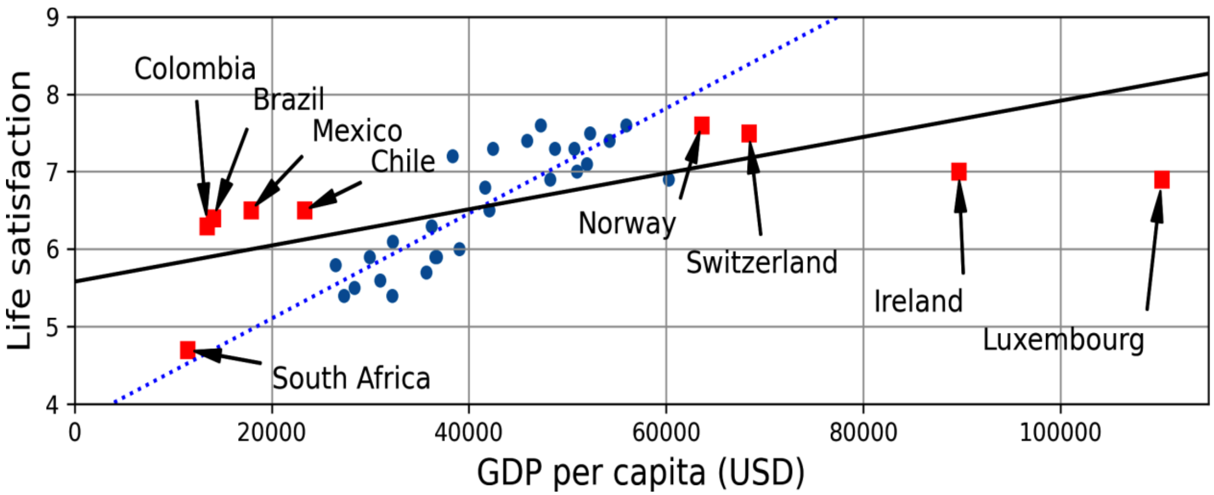


Figure. The dotted line is a model trained with a training sample which do not include countries whose GDP <$23,500 or GDP > $62,5000.
This model is unlikely to make accurate predictions, especially for very poor and very rich countries.
The solid line is a model trained with more representative training sample.

Challenges of Machine Learning

- **Poor-quality data**
  - If your training data is full of errors, outliers, and noise (e.g., due to poor quality measurements), it will make it harder for the system to detect the underlying patterns, so the built model is less likely to perform well.
  - The truth is, most data scientists spend a significant part of their time cleaning up their training data, e.g., outlier handing and the imputation of missing values.

Challenges of Machine Learning

- **Irrelevant features**
  - A critical part of the success of a machine learning project is coming up with a good set of relevant features to train on.
  - This process, called **feature engineering**, involves the following steps:
    - ***Feature selection*** - selecting the most useful features to train on among existing features
    - ***Feature extraction*** - combining existing features to produce a more useful one (dimensionality reduction)
    - Creating new features by gathering new data

Challenges of Machine Learning

- **Overfitting the training data**
  - ***Overfitting*** occurs when the model selected by the ML algorithm is <u>so complex that the model fits the dataset too closely</u> and <u>become sensitive to noise </u>in the data
  - The model will not generate to new instances.

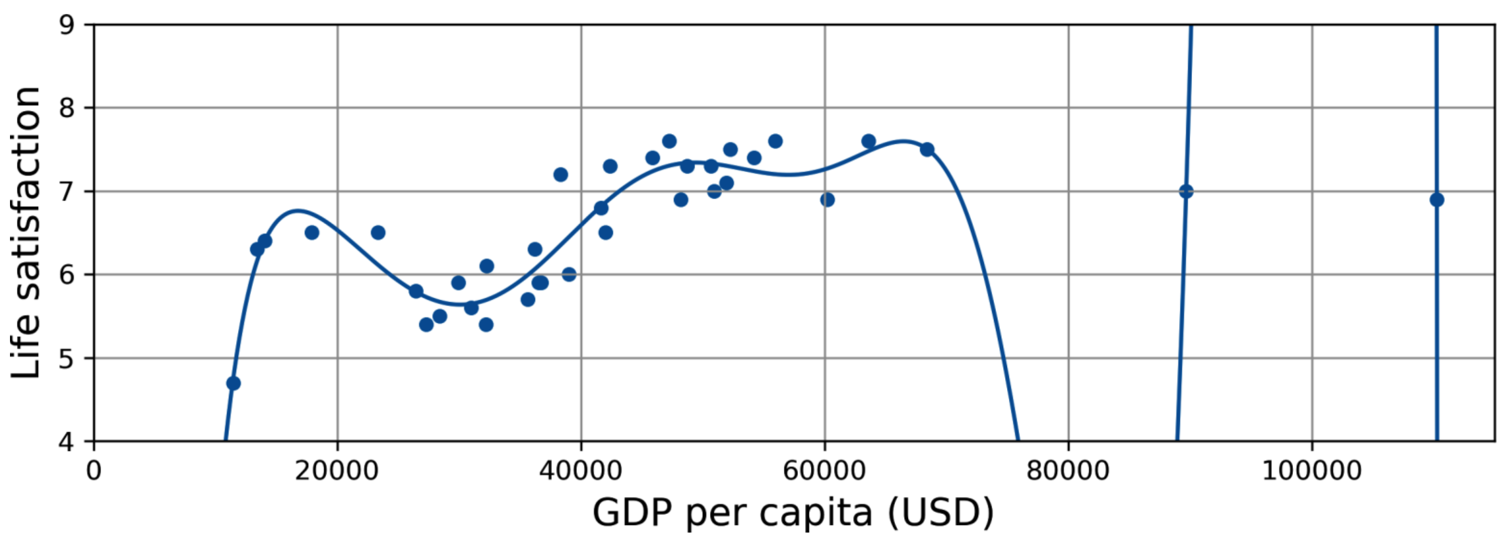


Figure. Overfitting the training data

Challenges of Machine Learning

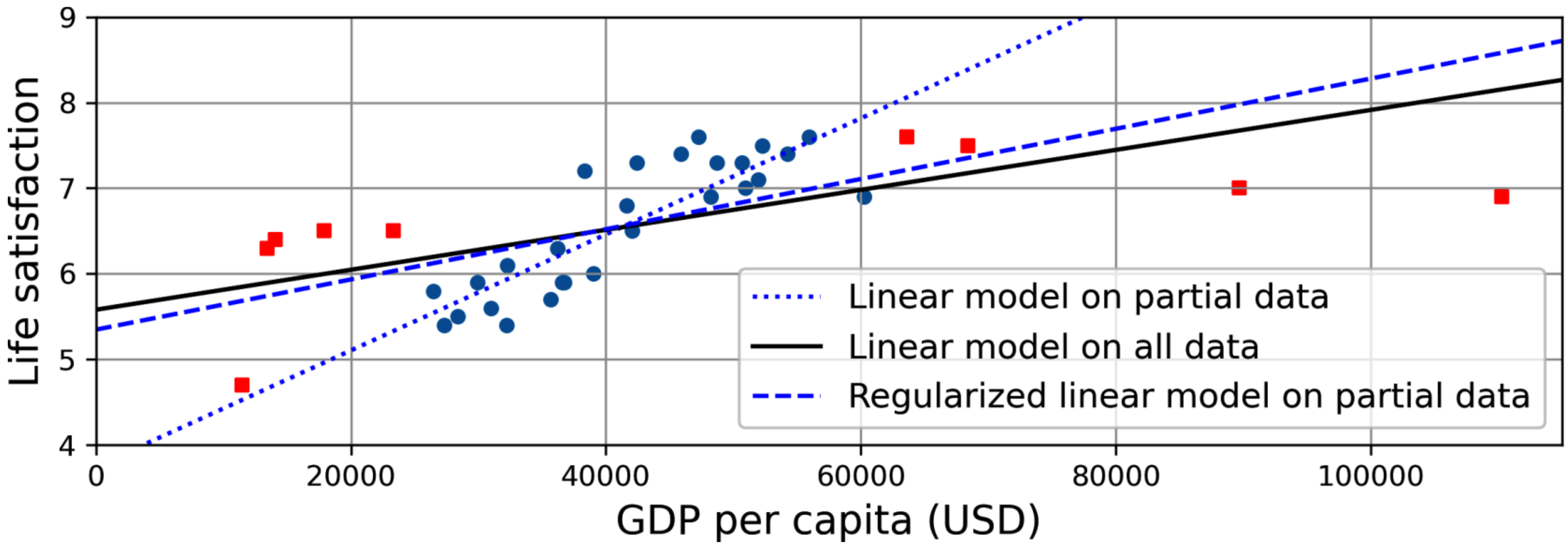
- **Overfitting the training data (cont.)**
  - Constraining a model to make it simpler and reduce the risk of overfitting is called ***regularization***.
  - The amount of regularization to apply during learning can be controlled by a ***hyperparameter***.
    - A hyperparameter is a parameter of a learning algorithm (not of the model).
    - As such, it is not affected by the learning algorithm itself; it must be set prior to training and remains constant during training.
  - Tuning hyperparameters is an important part of building a machine learning system

Example of Regularization

- A simple linear model of life satisfaction with just one attribute, GDP per capita

$$life\_satisfaction = \theta_0 + \theta_1 \times GDP$$

- Here, we can regularize with controlling θ_1,



Challenges of Machine Learning

- **Underfitting the training data**
  - ***Underfitting*** occurs when the prediction model selected by the ML algorithm is too simple to represent the underlying structure of the data.
  - The main options for fixing this problem are
    - Select a more powerful model, with more parameters.
    - Feed better features to the learning algorithm
    - Reduce the constraints on the model (for example by reducing the regularization hyperparameter).
- We want to find *Goldilocks Models*: that are ***just right***, striking a good balance between model complexity and simplicity

Outline

- What Is Machine Learning?
- Machine Learning Applications
- Machine Learning Experiment
- Goal of Machine Learning
- Machine Learning as a Search Problem
- Types of Machine Learning Systems
- Challenges of Machine Learning
- ☞ **Disciplines of Machine Learning**
- Summary

Disciplines of Machine Learning

- ML is inherently a **multidisciplinary field**.
- ML draws on concepts and results from many fields, including statistics, artificial intelligence, philosophy, information theory, biology, cognitive science, computational complexity, control theory, and other fields.

- We might need to view ML from all of these perspectives and to understand the problem settings, algorithms and assumptions that underline each.

Disciplines of Machine Learning

- **Role of Statistics**
  - Theory of statistics for making inference from a sample
- **Role of Computer Science:**
  - Storing and processing the massive amount of data
  - Designing efficient algorithms to solve the optimization problem
  - The algorithmic solution for inference and its representation needs to be efficient in its space and time complexity

Outline

- What Is Machine Learning?
- Machine Learning Applications
- Machine Learning Experiment
- Goal of Machine Learning
- Machine Learning as a Search Problem
- Types of Machine Learning Systems
- Challenges of Machine Learning
- Disciplines of Machine Learning
- ☞ **Summary**

Summary

- Machine learning is about making machines get better at some task by learning from data, instead of having to explicitly code rules.
- There are many different types of ML systems: supervised or not, batch or online, instance-based or model-based.
- In an ML project,
  - you gather data in a training set, and
  - you feed the training set to a learning algorithm.
  - If the algorithm is model-based, it tunes some parameters to fit the model to the training set (i.e., to make good predictions on the training set itself),
  - and then hopefully it will be able to make good predictions on new cases as well.

Summary (cont.)

- The system will not perform well if your training set is too small, or if the data is not representative, is noisy, or is polluted with irrelevant features (garbage in, garbage out).
- Lastly, your model needs to be neither too simple (in which case it will underfit) nor too complex (in which case it will overfit).