Introduction

COEN140

Santa Clara University

COEN140 Course Information

- **Instructor:** Dr. Ying Liu, email: yliu15@scu.edu

  Homepage: http://www.cse.scu.edu/~yliu1/

  Office Hours: Thursday, 10:30AM - 11:30AM, Online, or by appointment

- **TA:** Xuyang Wu, email: xwu5@scu.edu
- **Lectures:** Monday, Wednesday, Friday, 10:30AM - 11:35AM, Online
- **Labs:** COEN140L-01 Wednesday, 2:15PM - 5:00PM, Online
  COEN140L-02 Thursday, 2:15PM - 5:00PM, Online

References

- "Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems", 1st Edition, by Aurélien Géron, ISBN-13: 978-1491962299, ISBN-10: 1491962291.
- "Pattern Recognition and Machine Learning", 1st Edition, by Christopher M. Bishop, Springer, 2006. ISBN-13: 978-0387310732.
- "Learning Python", 5th Edition, by Mark Lutz, O'Reilly Media, Inc., 2013. ISBN: 978-1-449-35573-9.

References

- Pattern Classification, 2nd Edition, by R. O. Duda, P. E. Hart, and D. G. Stork, Wiley 2001. ISBN-13: 978-0471056690
- Machine Learning: A Probabilistic Perspective, by Kevin P. Murphy, The MIT Press, 2012. ISBN-13: 978-0262018029.
- Website: http://cs231n.stanford.edu/

Grading

Lecture and Lab sessions will be given the same letter grade	Percentage (100%)
Homework Assignments	20%
Lab Assignments (python programming)	20%
Midterm Exam	25%
Final Exam	35%

Homework and Lab Assignments

- Strict deadline!
  - 20% deducted per day late, for each assignment submission

    (i.e. if 5 days late, 0 score)

- Excuses such as technical issue (e.g. Internet access or file corruption) won’t be accepted. Because this is unfair to other students who submit the assignments on time. Double check the files when you submit them. Do not wait till the last minute!

Homework and Lab Assignments

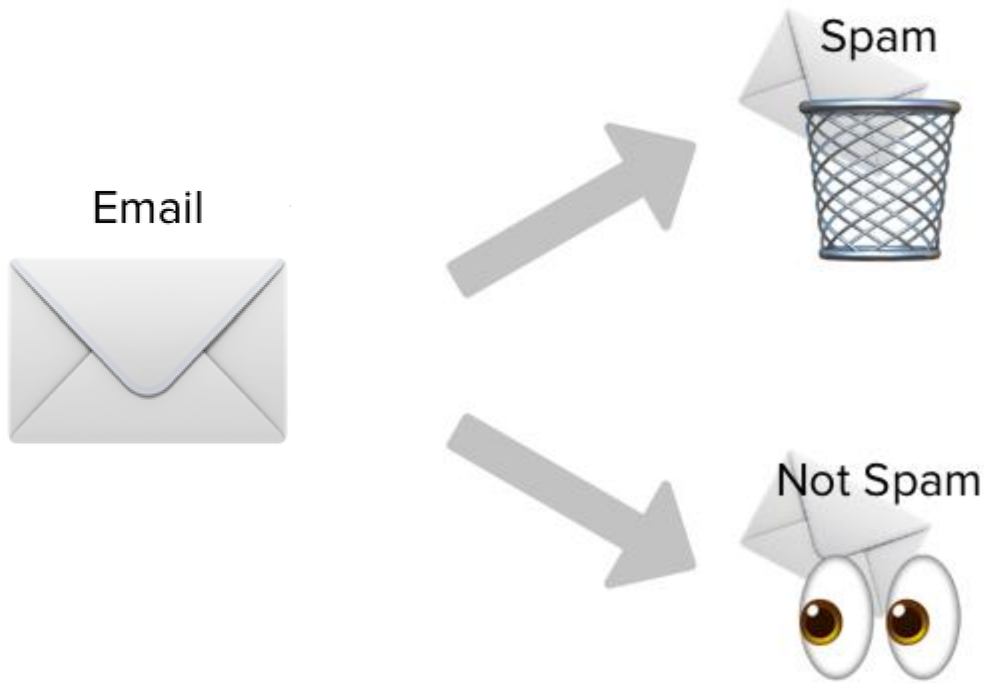
- You are encouraged to discuss the assignments with classmates. However, you must independently write up your own solutions and implement your own code.
- Very similar assignment submissions will be considered as academic dishonesty.
- Make-up exams will not be given except for legitimate reasons with supporting documentation (for example, a medical reason with doctor’s official note), and with prior approval from the instructor

Schedule (subject to change)

- Week 1: Introduction, Linear Algebra Review.
- Week 2: Regression.
- Week 3: Gradient Descent, Clustering.
- Week 4: Logistic Regression.
- Week 5: Neural Network.
- Week 6: Deep Learning.
- Week 7: Maximum Likelihood Estimator, Bayesian Classifier.
- Week 8: Decision Trees.
- Week 9: Principal-Component Analysis, Linear Discriminant Analysis.
- Week 10: Review.

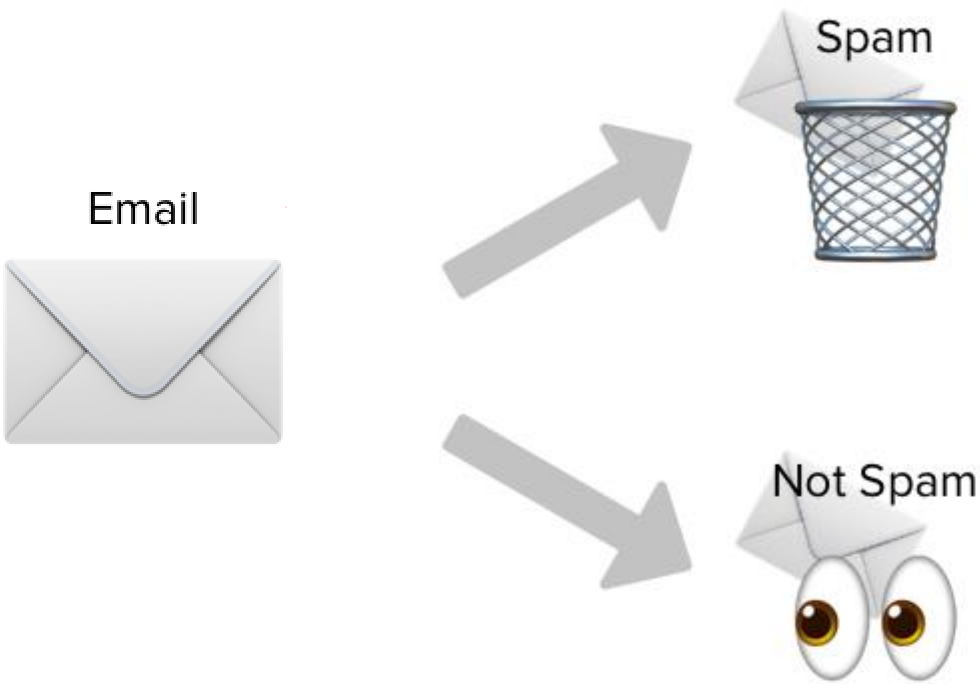
What is Machine Learning?

- Spam Filter Example
- What is a spam filter?



What is Machine Learning?

- How would you develop a spam filter?
  - Develop a computer program



Spam Filter

- How would you develop a spam filter?

- Look at what spam typically looks like
  - Words that appear in the subject

  - Patterns in the sender's name, the email's body, etc.

Spam Filter

- An excerpt

Spam: Wholesale Fashion Watches -57% today. Designer watches for cheap ...
Spam: You can buy ViagraFr$1.85 All Medications at unbeatable prices! ...
Spam: WE CAN TREAT ANYTHING YOU SUFFER FROM JUST TRUST US ...
Spam: Sta.rt earn*ing the salary yo,u d-eserve by o'btaining the prope,r crede'ntials!

Ham: The practical significance of hypertree width in identifying more ...
Ham: Abstract: We will motivate the problem of social identity clustering: ...
Ham: Good to see you my friend. Hey Peter, It was good to hear from you. ...
Ham: PDS implies convexity of the resulting optimization problem (Kernel Ridge ...

- Good features that indicate spam?
  - Words: “for cheap”, “You can buy”: indicators of spam

Spam Filter

- An excerpt

Spam: Wholesale Fashion Watches -57% today. Designer watches for cheap ...
Spam: You can buy ViagraFr$1.85 All Medications at unbeatable prices! ...
Spam: WE CAN TREAT ANYTHING YOU SUFFER FROM JUST TRUST US ...
Spam: Sta.rt earn*ing the salary yo,u d-eserve by o’btaining the prope,r crede’ntials!

Ham: The practical significance of hypertree width in identifying more ...
Ham: Abstract: We will motivate the problem of social identity clustering: ...
Ham: Good to see you my friend. Hey Peter, It was good to hear from you. ...
Ham: PDS implies convexity of the resulting optimization problem (Kernel Ridge ...

- Good features that indicate spam?
  - be all uppercase, or
  - have punctuation embedded in words

    “you deserve” -> “yo,u d-eserve”

Spam Filter

- Write a detection algorithm for each of these abnormal patterns. Your program will flag emails as spam if a number of these patterns are detected.

- You would test your program
  - Let your program take a new email
  - Your program will check whether the input email has those abnormal patterns
  - Your program will output the final decision

Spam Filter

- How to evaluate whether your program is good or not?

- Assume: you have 1000 new emails, which you know the “ground-truth”
  - You test your program with these 1000 emails
  - Collect detection errors

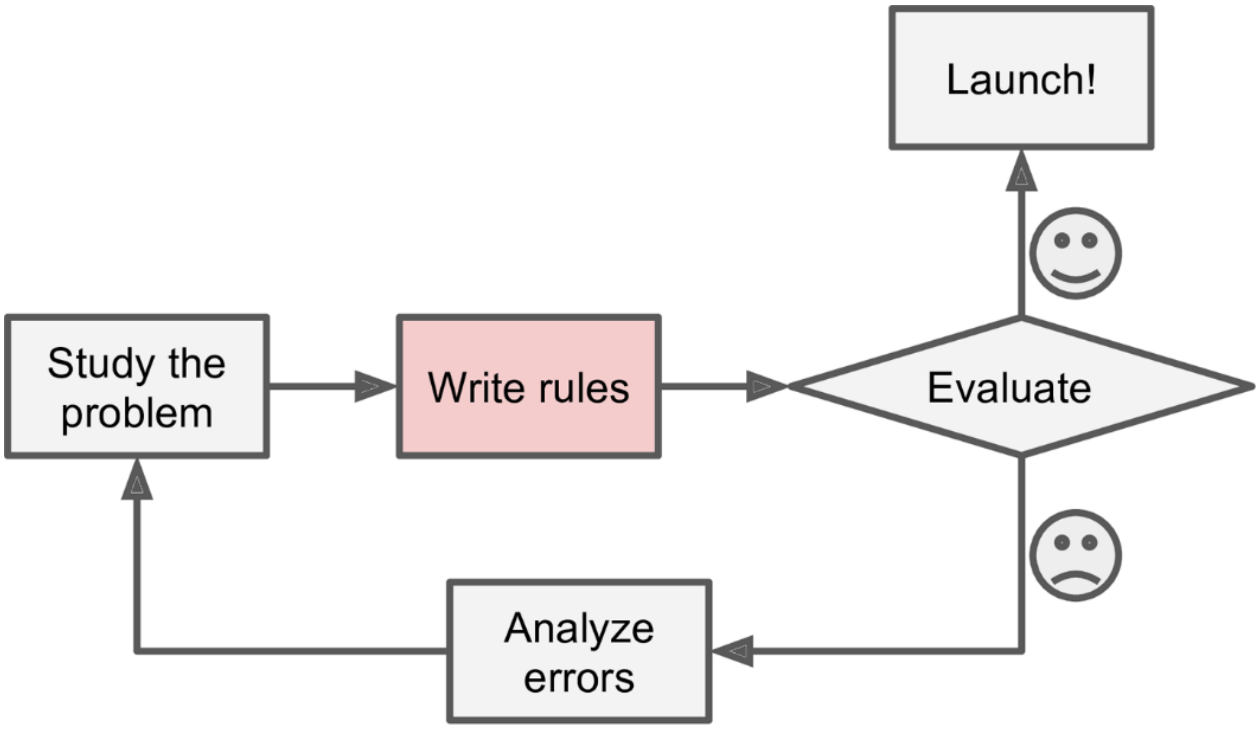
- If the error rate is high, then your program is bad

- Go back to modify those abnormal patterns

  …

Traditional Programming

- Traditional Programming of a Spam Filter
- Rules: hand-tuned

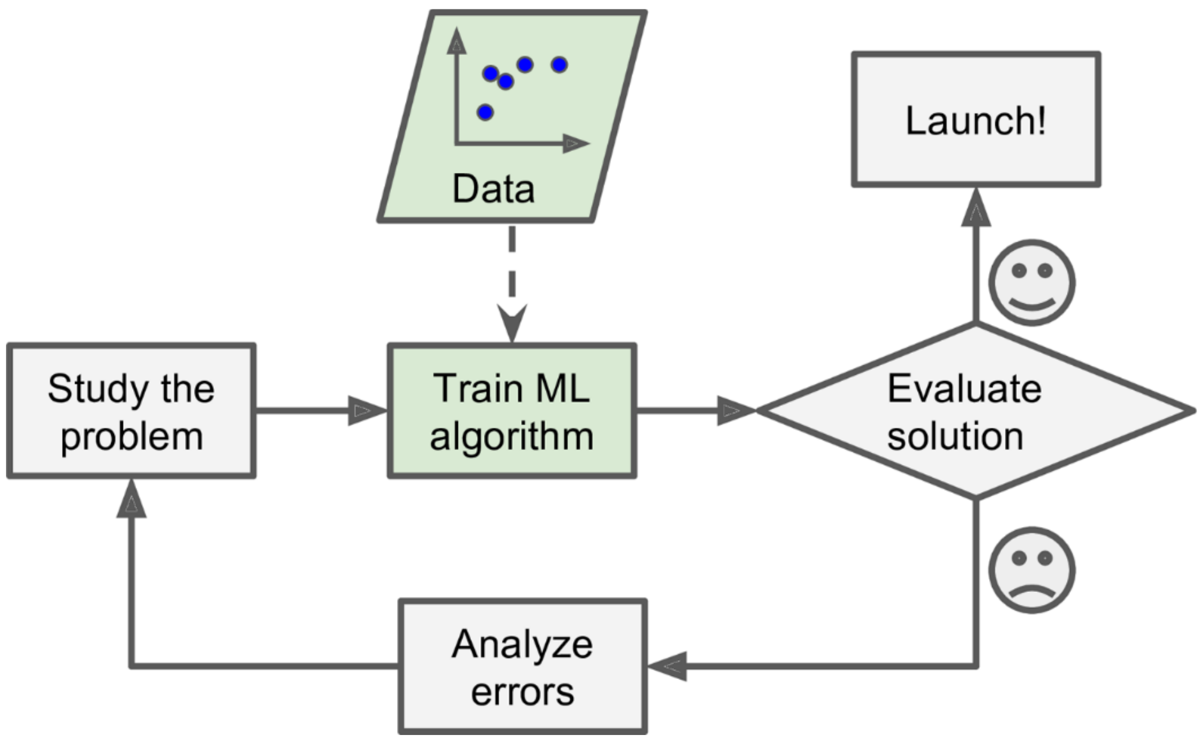


Traditional Programming

- Drawback
  - The problem is nontrivial, your program will likely become a long list of complex rules – hard to maintain.

A spam filter based on Machine Learning

- Rules: automatically generated



A spam filter based on Machine Learning

- Automatically learns which words are good predictors of spam

- The program is much shorter

- Easier to maintain

- Most likely more accurate

Motivation of Machine Learning

- Existing solutions for many problems
  - Require a lot of hand-tuning or long lists of rules

- Machine Learning can often simplify code and perform better

Motivation of Machine Learning

- Machine Learning can
  - Automatically discover patterns/regularities in data
  - Use these regularities to take actions for new data
  - Adapt to environment change and new data
  - Get insights about complex problems and large amounts of data

Elements of Machine Learning

- Data (Training Data)
  - Experience
  - e.g. 10000 emails, some are spam, some are ham
- Task
  - Classification
  - Prediction
- Algorithm (computer program)
  - Automatically generate “rules” to do the task
- Performance Evaluation Metric
  - e.g. detection error rate

Elements of Machine Learning

- An algorithm (computer program) is considered as a Machine Learning algorithm (program) if

  - Its performance on the task, as measured by the evaluation metric, improves with experience (data).

Applications of Machine Learning

- Recognition
  - Face, speech, handwritten character
- Fraud detection (e.g., abnormal credit card transactions)
- Recommender systems (e.g., which movies/products you would like)
- Market prediction (e.g., stock/house prices)
- Defense applications: military attack detection
- Decision Making: Robotics

Supervised vs Unsupervised Learning

- Supervised Learning: explicit feedback in the form of target values
  - Goal: to make predictions (predict prices), or to generate class labels
  - Target values: the true price values, or the true class labels (call them the ground-truths)

- Unsupervised learning: only samples, no target values
  - Goal: to reveal structure in the observed data

Supervised Learning: Classification

- The goal is to assign each input to one of a finite number of discrete categories
  - Face recognition
  - Hand-written digit recognition
  - Speech recognition
  - Medical diagnosis
    - from symptoms to illnesses
  - Web Advertising
    - predict if a user will click on an ad on the Internet

Classification

- You want to classify

versus

- Training Data: 100 monkey images and 200 human images with labels what is what.

$$\{\mathbf{x}_i, t_i = 0\}, i = 1, \cdots, 100$$
$$\{\mathbf{x}_j, t_j = 1\}, j = 1, \cdots, 200$$

  - where $\mathbf{x}$ represents the intensity of the image pixels and $t = 0$ means "monkey" while $t = 1$ means "human".

- (Training) Samples: $\mathbf{x}$
- Target values: labels t
  - The **ground-truth** values

Classification

- You want to classify

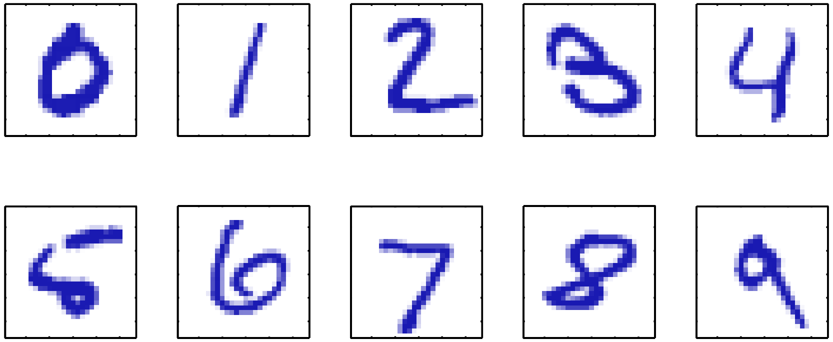
 versus 

- Task: Here is a new image:  monkey or human?

- Challenges
  - lighting, pose , expressions, occlusions (glasses, beard), make-up, hair style

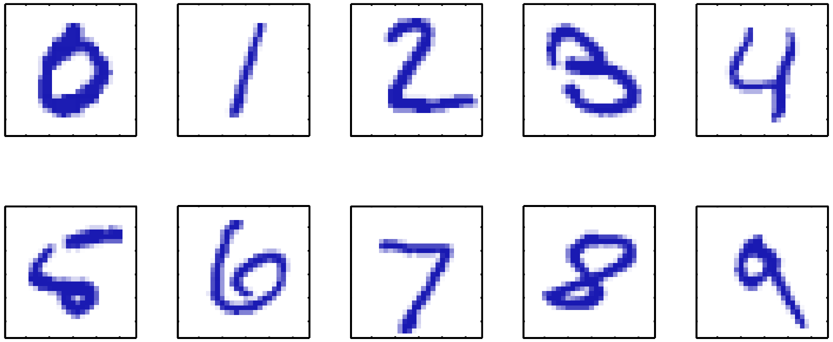
Classification

- Hand-written Digits Recognition
- Each digit: a 28x28 pixel image
- Represented by a vector $\mathbf{x}$: 784 real numbers
- Goal: build a machine that will take $\mathbf{x}$ as input and produce the identity of the digit 0, …, 9 as the output
  - Samples: vectorized images $\mathbf{x}$
  - Target values (labels): 0, 1, 2, …, 9



Classification

- Hand-written Digits Recognition
- Challenges
  - Different hand-writing styles



Classification

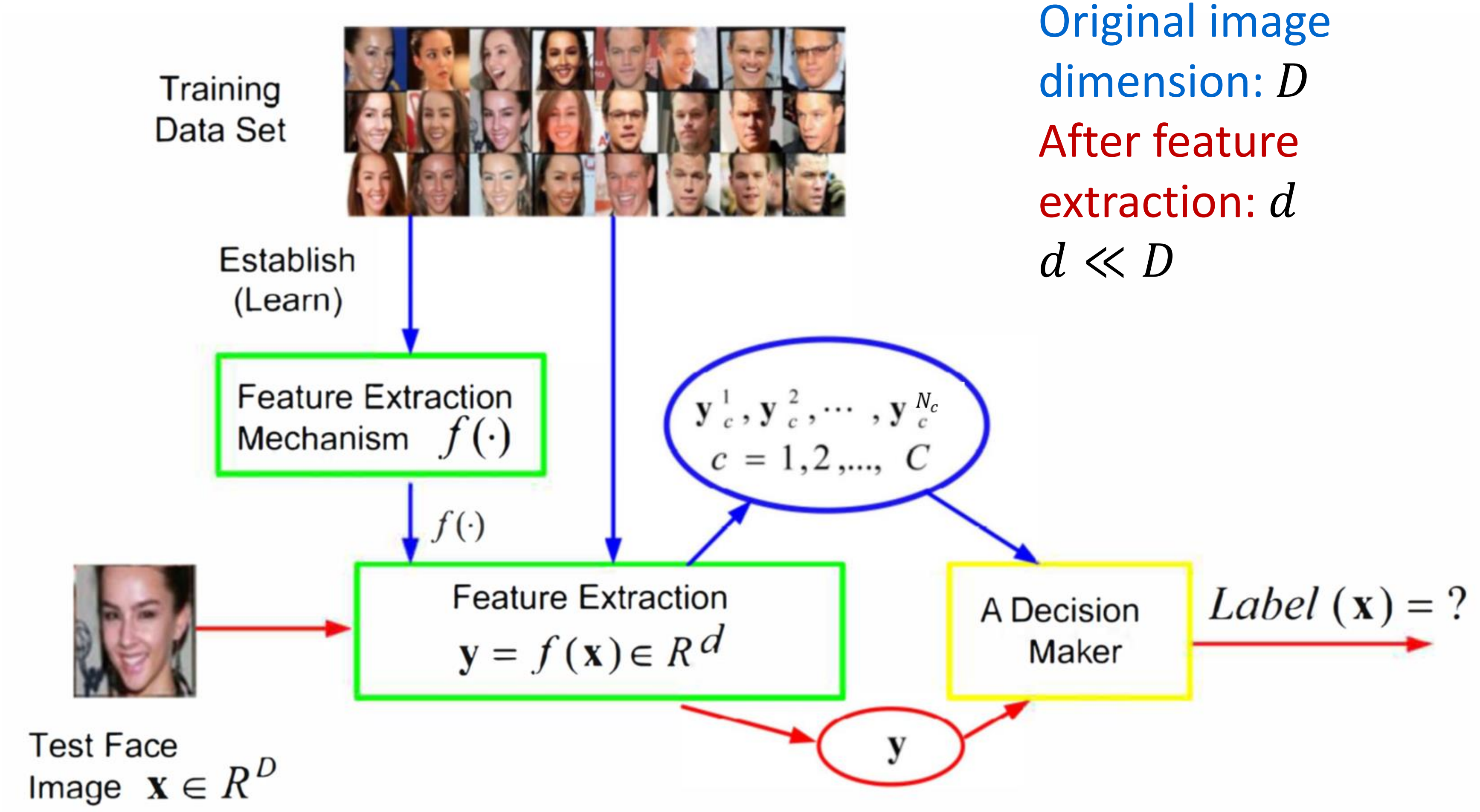
- Skin Detection Problem
  - Sample?
  - Target value?

Test Image

Ground Truth Mask

Classification Result

Classification: Face Recognition System



Assume:

Original image dimension: D

After feature extraction: d

$d \ll D$

Supervised Learning: Regression

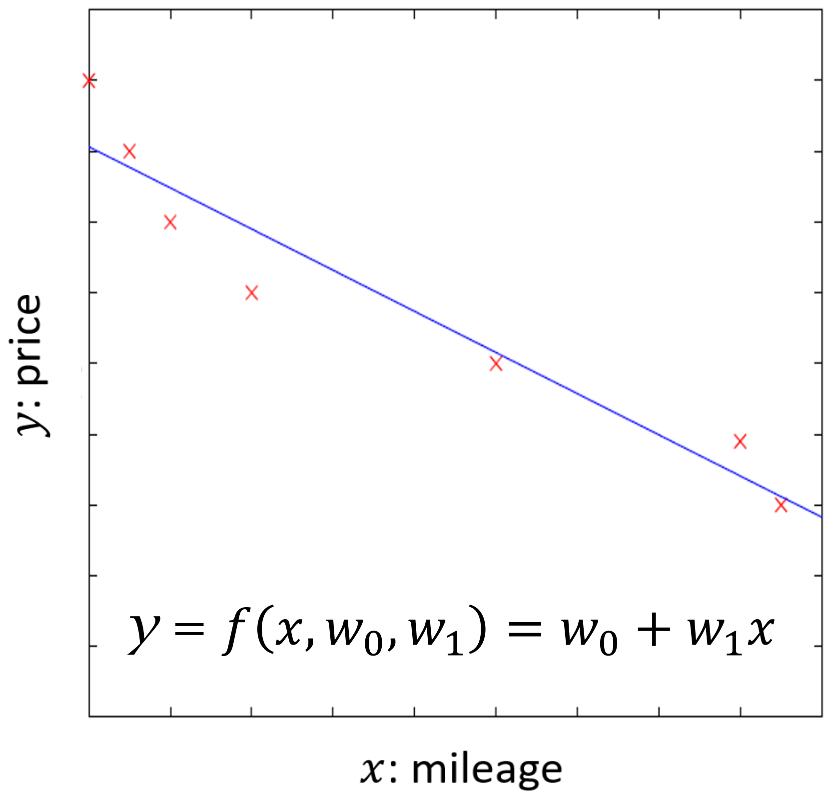
- When the desired output (target value) consists of one or more continuous values
  - Temperature prediction
  - Price prediction
  - ...

Regression

- Predict the price of a used car
- x: car mileage
- t: target price

  (red dots values)
- y: predicted price

  (blue line values)



Regression

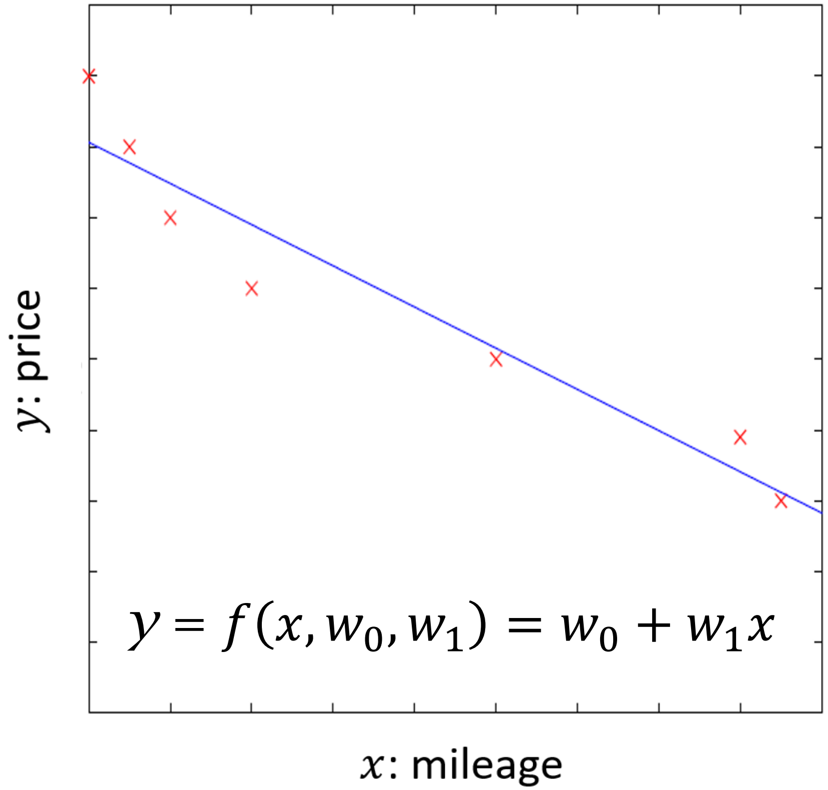
- Predict the price of a used car

- Model: $f(x, w_0, w_1)$

- (model) Parameters:

  w_0, w_1

- To learn a model that generates y close to t



Regression

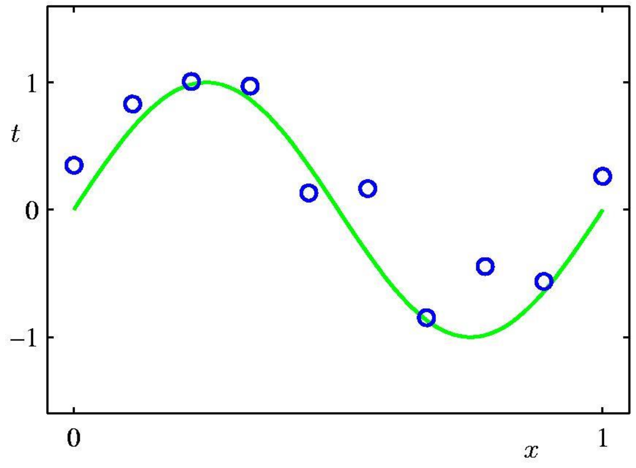
- We are given $N = 10$ samples (data points)

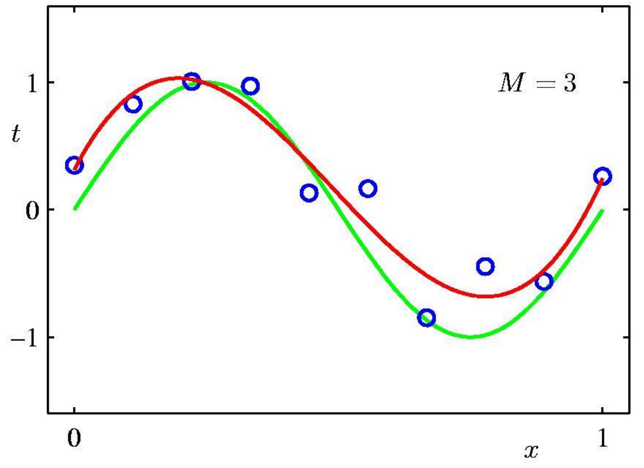
$$x_1, x_{2,\dots,}\, x_N$$

- Observations of the target values

$$t_1, t_{2,\dots,}\, t_N$$

- Green sine curve: the underlying data generation mechanism





Regression

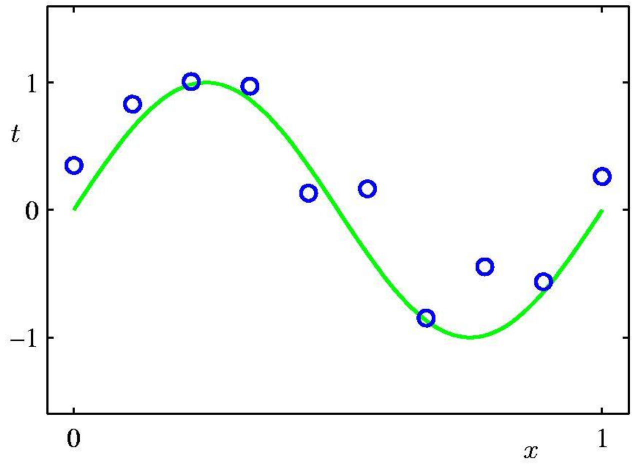
- We are given $N = 10$ samples (data points)

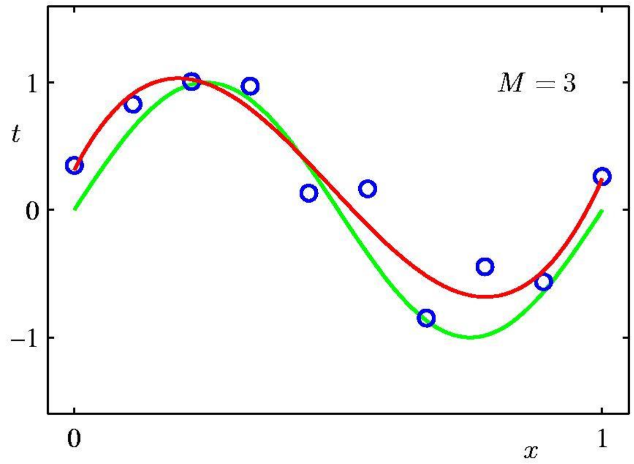
$$x_1, x_{2,\dots,}\, x_N$$

- Observations of the target values

$$t_1, t_{2,\dots,}\, t_N$$

- Blue circles: data points (noisy)





Regression

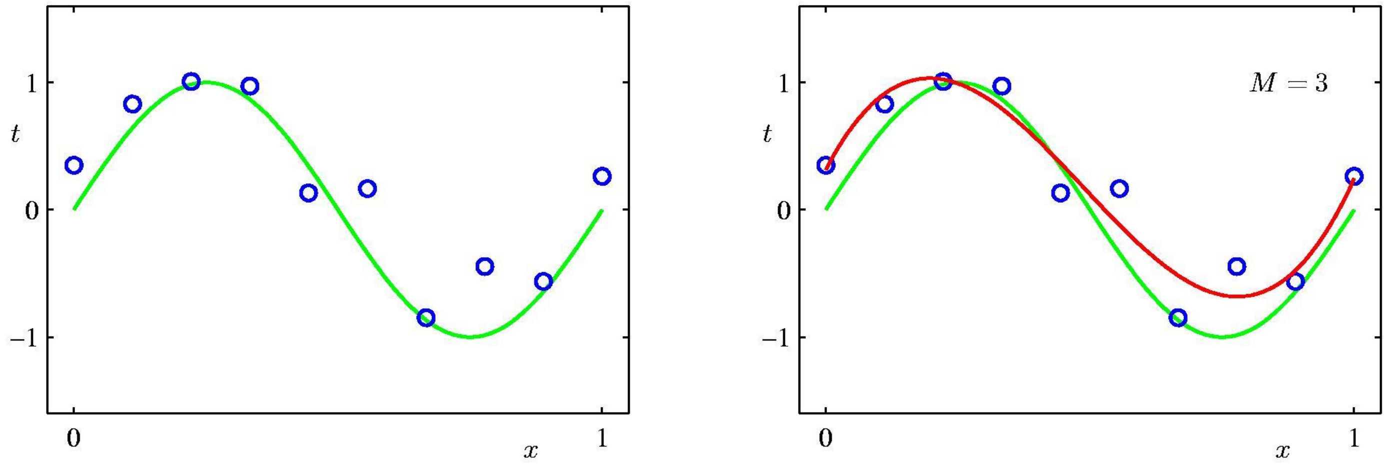
- We are given $N = 10$ samples (data points)

$$x_1, x_{2,\dots,} x_N$$

- Observations of the target values

$$t_1, t_{2,\dots,} t_N$$

- Red curve: the curve learned from these samples



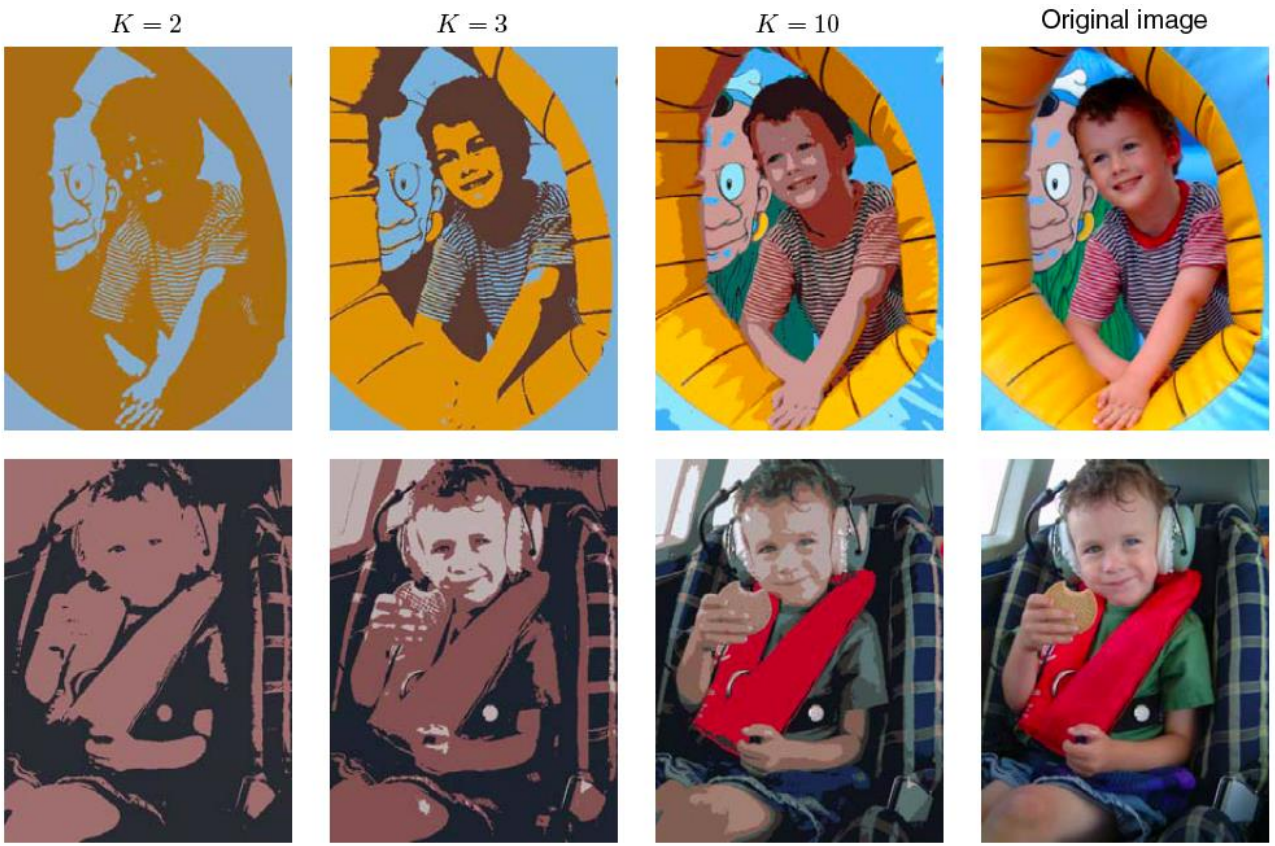
Unsupervised Learning

- The training data consists of a set of samples without any corresponding target values

- Goal: to reveal structure in the observed data

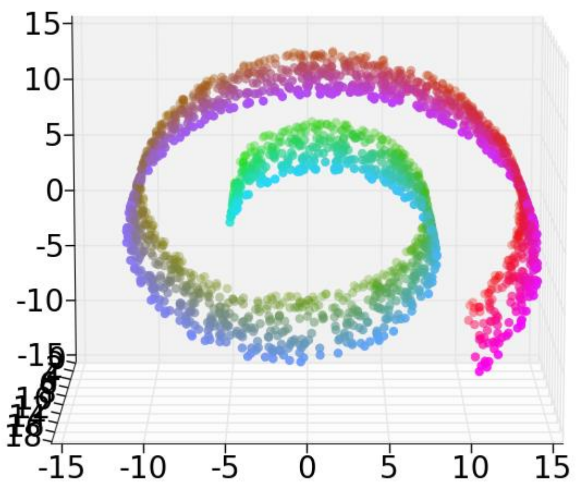
Unsupervised Learning

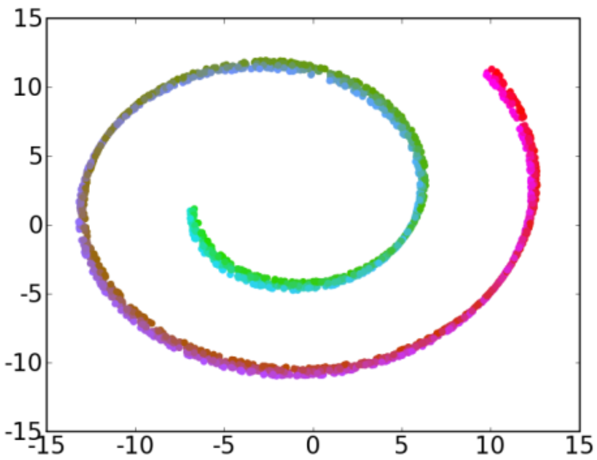
- Clustering: to discover groups of similar samples within the data



Unsupervised Learning

- Dimensionality Reduction: project the data from a high-dimensional space down to a low-dimensional space
  - Discard unimportant features
  - For visualization
  - e.g. The Swiss Roll





Unsupervised Learning

- Anomaly detection: detect unusual instances
  - Anomalous Internet traffic flow

Batch Learning vs Online Learning

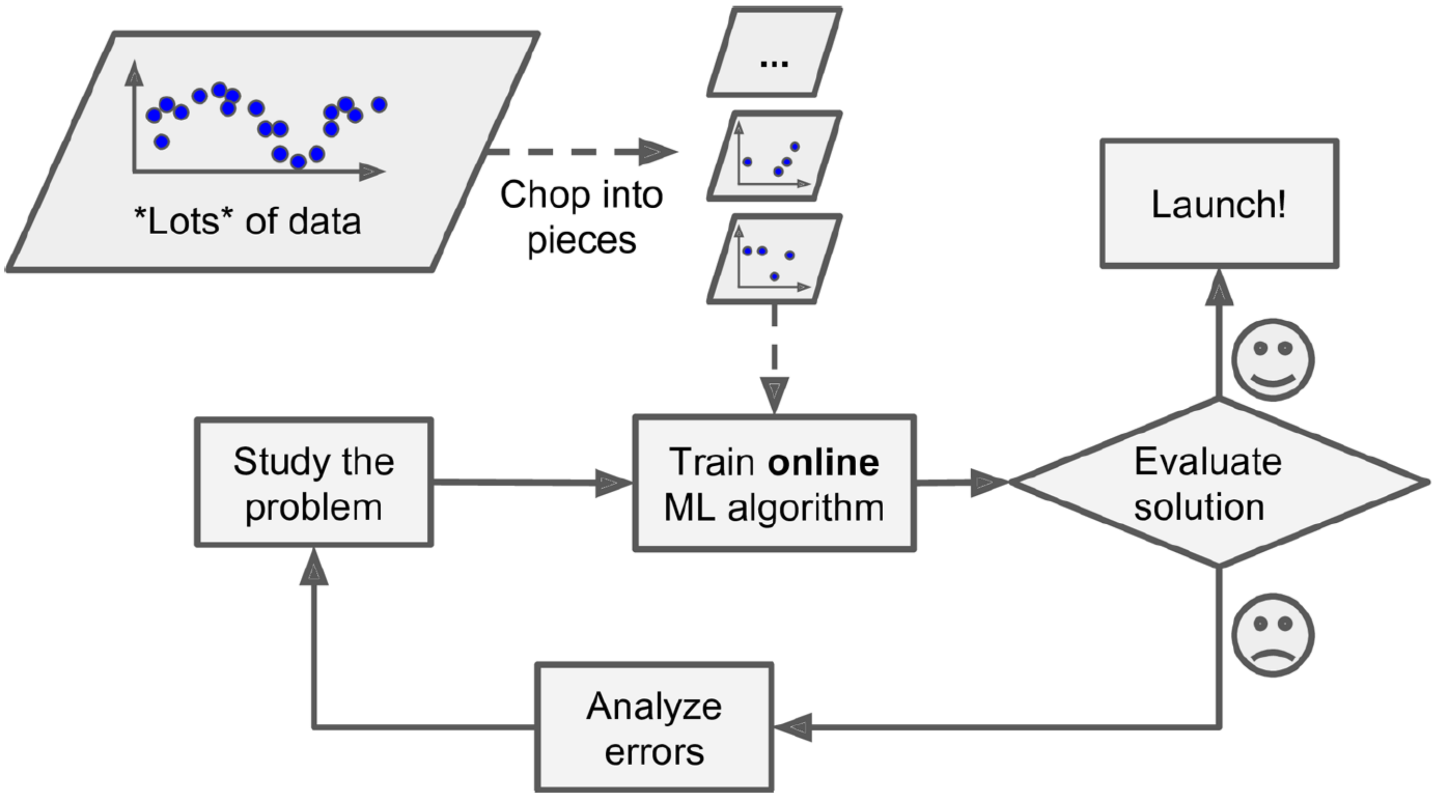
- Batch Learning: the system is trained using the available training data all at once
  - a.k.a. Offline learning

- Take a lot of time and computing resources

- First the system is learned with all available data, and then it is launched into production and runs without learning anymore

Batch Learning vs Online Learning

- Online Learning: you train the system incrementally by feeding it data samples sequentially, either one by one or by small groups of samples called *mini-batches*



Batch Learning vs Online Learning

- Online Learning: Each learning step is fast and cheap, so the system can learn about new data on the fly, as it arrives
  - Good for systems that receive data as a continuous flow (e.g., stock prices) and need to adapt to change rapidly or autonomously
  - Good if your system have limited storage space: once an online learning system has learned from new data samples, it does not need them anymore, so it can discard them

Model-Based Learning

- Learn a model of the training samples, then use that model to make predictions
- Steps:
  - You studied the data.
  - You selected a model (a function that has some unknown parameters)
  - You trained the model on the training data (i.e., the learning algorithm searched for the model parameter values that minimize a cost function).

Model-Based Learning

- Learn a model of the training samples, then use that model to make predictions
- Steps:
  - Finally, you applied the model to make predictions on new samples (this is called *inference*), hoping that this model will generalize well.

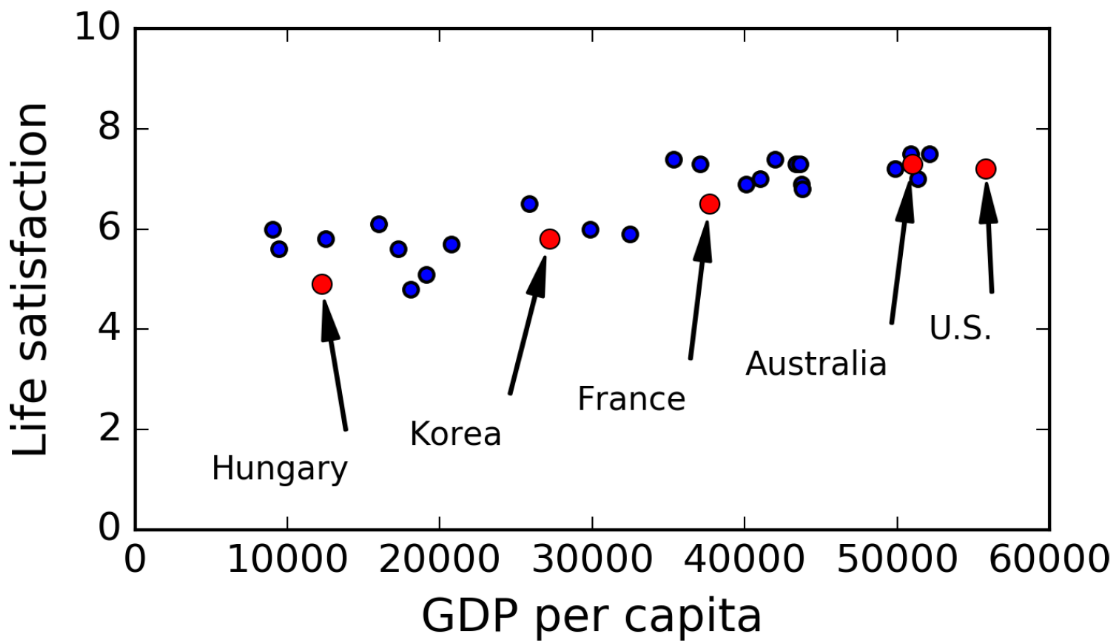
Model-Based Learning

- Example: you want to know if money makes people happy, so you download the *Life Satisfaction* data as well as stats about GDP per capita. Then you join the tables and sort by GDP per capita.

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

Model-Based Learning

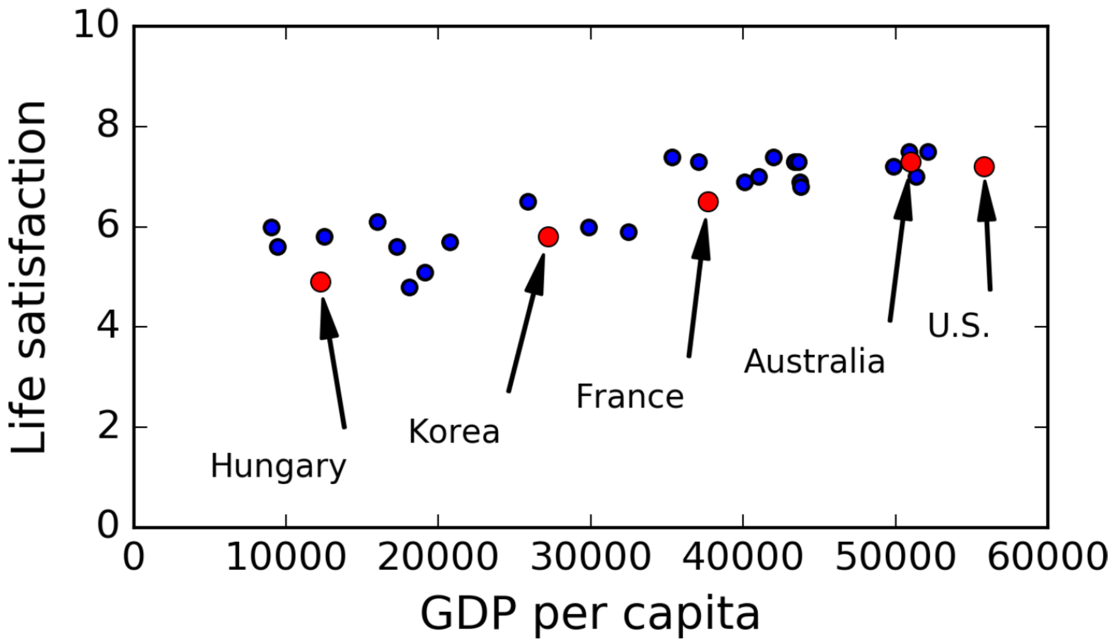
- Plot the data for a few random countries



Model-Based Learning

- Model selection: you selected a *linear model* of life satisfaction with just one attribute, GDP per capita

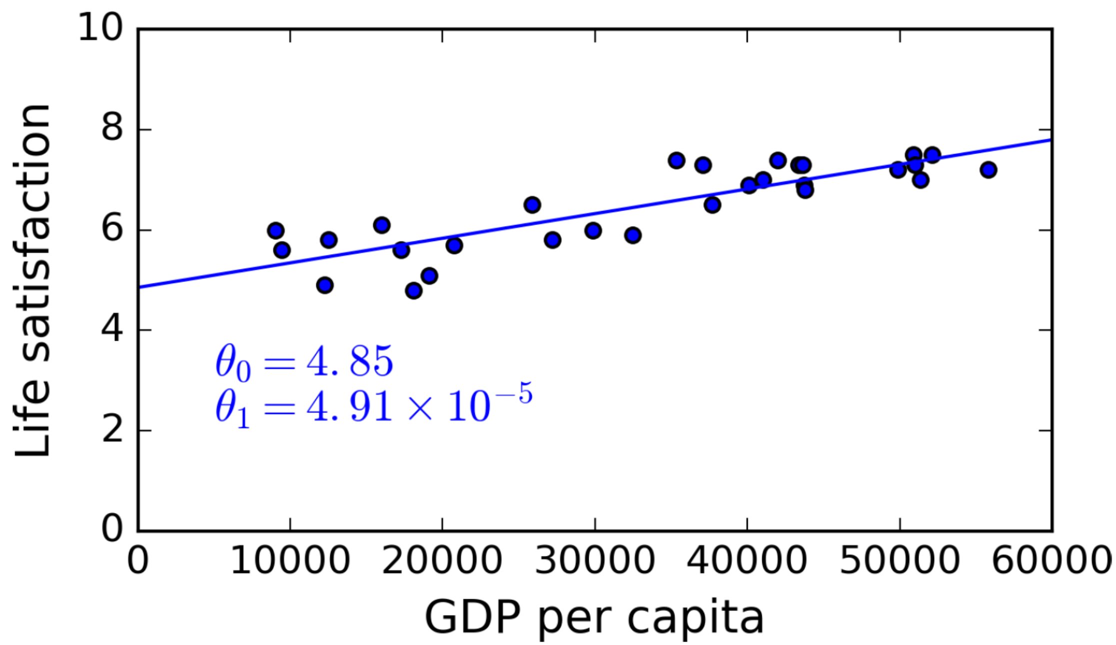
$$life\_satisfaction = \theta_0 + \theta_1 \times GDP\_per\_capita$$



Model-Based Learning

- Train the model: you feed the model your training samples and it finds the parameters that make the straight-line fit best to your data.

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



Training Set

- For machine learning, the training set is a set of data samples that you will use to generate a machine learning model
- Supervised Learning
  - Training set: $(x_n, t_n), \quad n = 1, \dots, N_{train}$
    - N_{train} training samples
    - x_n is the input, t_n is the target value

    - In general, the input and the target value can both be multi-dimensional
    - $(\mathbf{x}_n, \mathbf{t}_n), \quad n = 1, \dots, N$

Training Set

- For machine learning, the training set is a set of data samples that you will use to generate a machine learning model
- Supervised Learning
  - Training set: $(x_n, t_n), \quad n = 1, \dots, N_{train}$
    - A function will be learned to map the input to the corresponding output
    - e.g. $y_n = f(x_n)$
    - We want $f(\cdot)$ to be such that the model output y_n is close to the target output t_n

Test Set

- The test set is a set of data samples that are independent of the training set.
  - Test set: $(x_n, t_n), \;\; n = 1, \dots, N_{test}$
    or multi-dimensional: $(\mathbf{x}_n, \mathbf{t}_n), \;\; n = 1, \dots, N_{test}$

- You will use the test set to evaluate the performance of the learned function/model $f(\cdot)$

- Training Set and Test Set cannot overlap
  - They cannot have common data samples

Challenges in ML: Due to Data

- Insufficient Quantity of Training Data
- Nonrepresentative Training Data
  - e.g. missing important training samples
- Poor-Quality Data
  - When the training data is contaminated by outliers, and noise (e.g., due to poor-quality measurements)
- Irrelevant Features
  - Need feature selection/extraction

Challenges in ML: Due to Algorithms

- Overfitting: the model performs well on the training data, but it does not generalize well.
  - It happens when the model is too complex relative to the amount of the training data.
  - Generalization Capability: whether your trained model works well on new, unseen data samples
- Underfitting: when your model is too simple to learn the underlying structure of the data