Pattern Recognition

Chapter 01 – Introduction

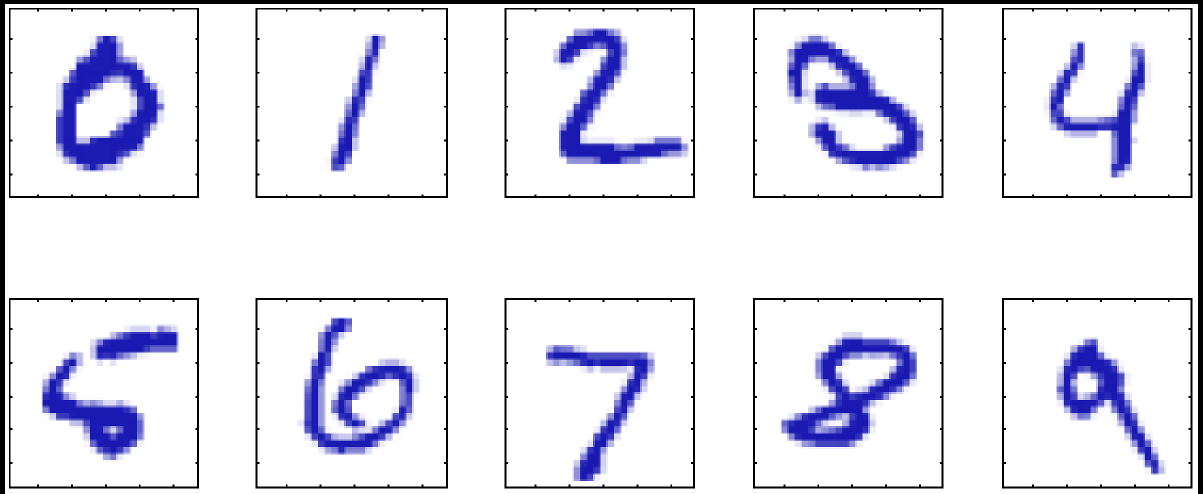
Content

Content
Introduction
Polynomial Curve Fitting
Bayesian Probabilities
Bayesian Curve Fitting

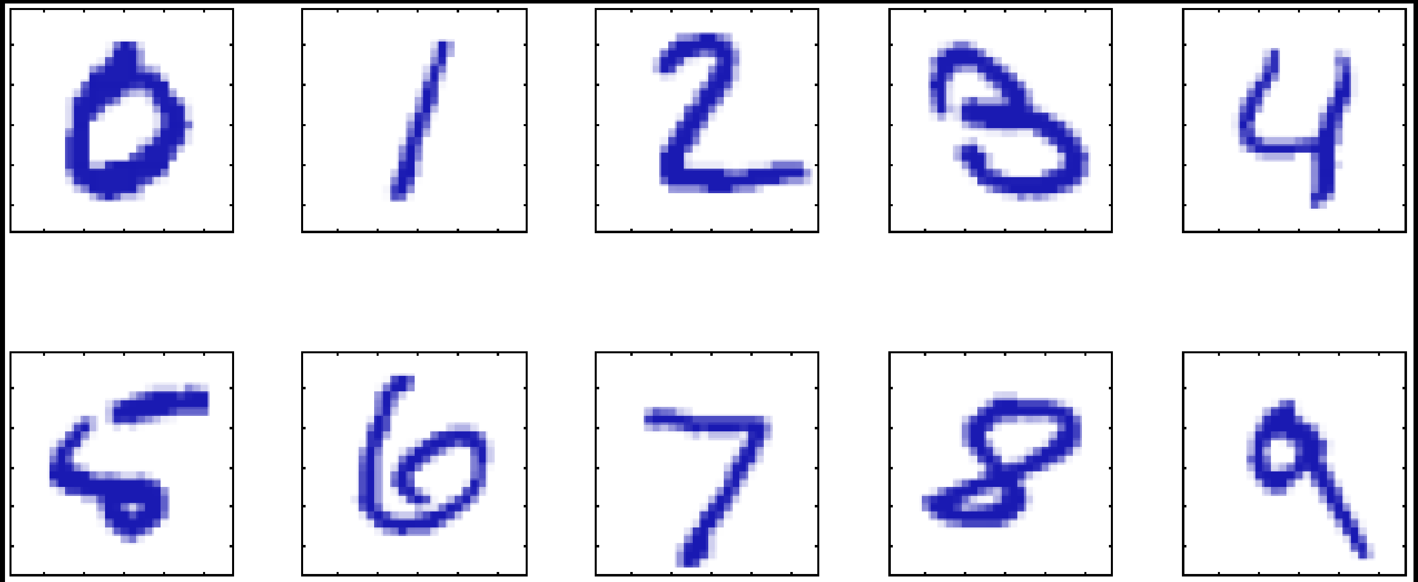
Go through mathematical notation of the PRML book.

Introduction

- **Pattern Recognition**
  is concerned with the automatic discovery of regularities in data using algorithms and to take actions such as classifying the data into different categories.
- **Example**: recognizing handwritten digits



Introduction



- Each digit corresponds to a 28×28-pixel image.
- The image is represented by a vector x comprising 784 real numbers.
- The goal is to build a machine that will take such a vector x as input and produce the identity of the digit 0, . . . , 9 as the output.

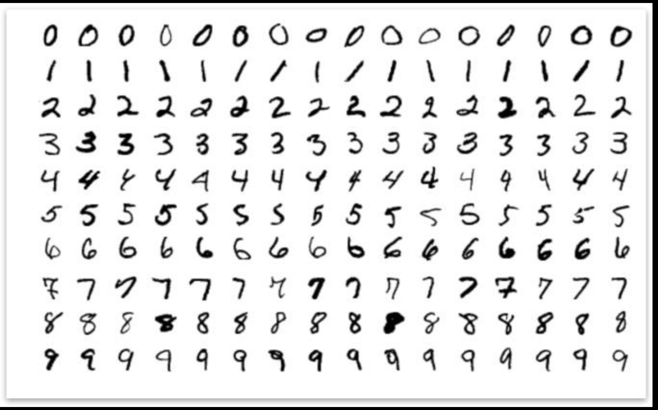
Introduction

To build a model to recognize handwritten digits:

- We need a dataset (training set) to make the make machine **learn** from it.
- The dataset consists of N images of digits. Each image has a label to indicate the target value.
- The target value of the image is what we want to predict.
- The set of target values of the N digits are called **target vector** $\boldsymbol{t}$.
- Each image of the data set is called an **instance (example)** x.

Introduction

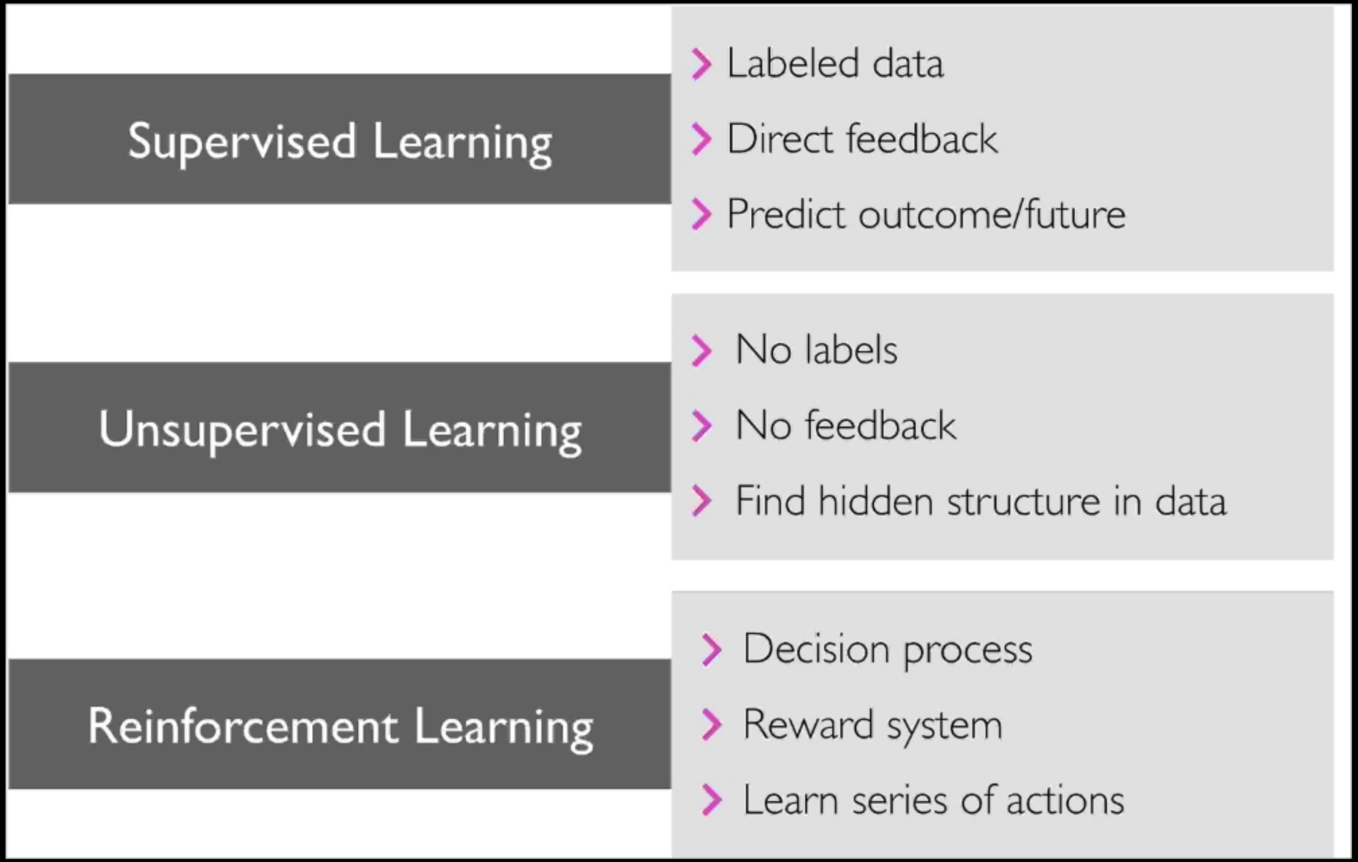
Training dataset → Learn → Predictions



→ Machine learning algorithm → {0, 1, 2, ..., 9}

Introduction

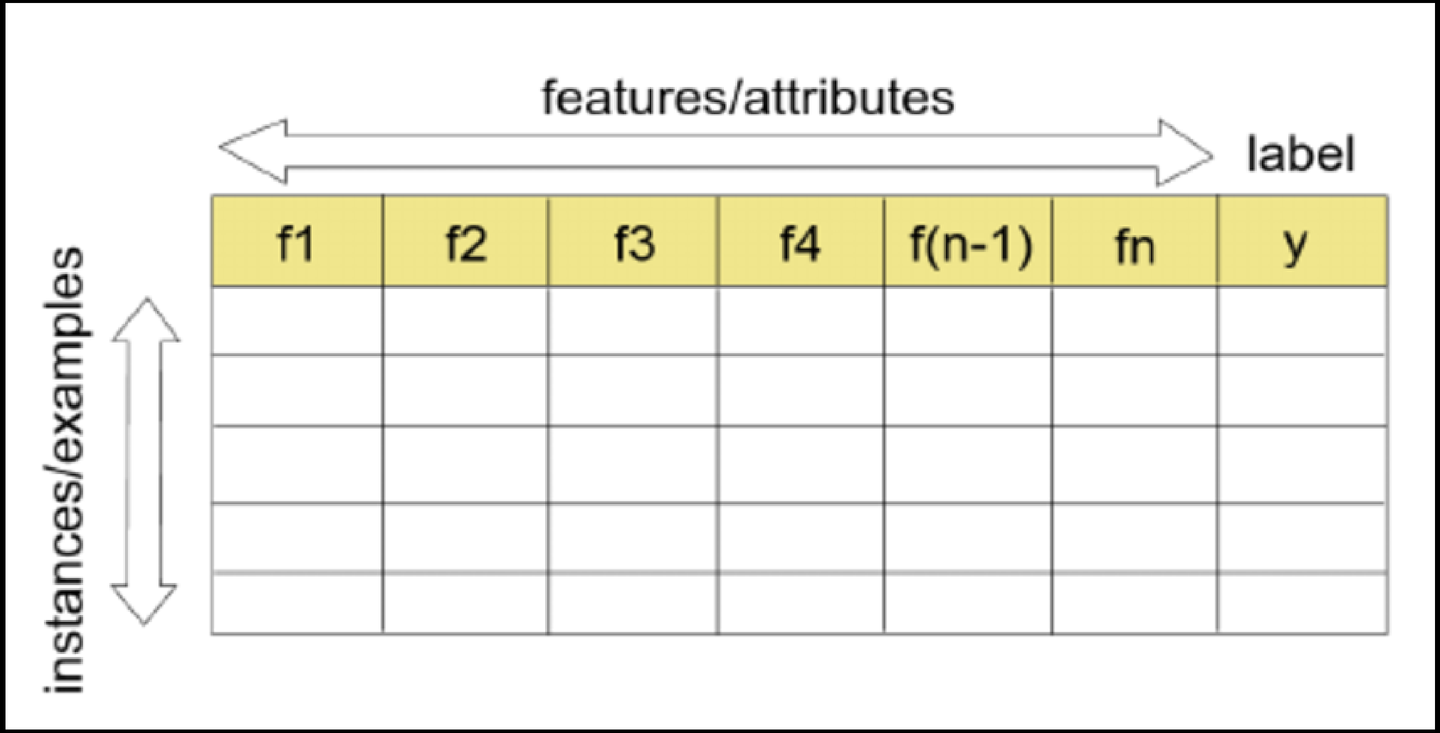
- Types of machine learning



https://subscription.packtpub.com/book/big_data_and_business_intelligence/9781787125933/1/ch01lvl1sec09/the-three-different-types-of-machine-learning

Introduction

- A view of a tabular dataset



https://www.researchgate.net/figure/A-classic-input-dataset-for-a-machine-learning-execution_fig2_283422326

Introduction

- A view of a tabular dataset

	X(Input)			Y(Output)
Student	Test1 marks	Test2 Marks	Study hours	Final result
1	30	35	4	Pass
2	42	45	6	Pass
3	20	17	1	Fail
4	45	48	6	Pass
5	25	22	2	Pass
6	34	40	2	Pass
7	49	47	6	Pass
8	17	10	0	Fail
9	25	20	1	Fail
10	35	38	3	Pass

https://medium.com/analytics-vidhya/identify-the-correct-dataset-for-your-ml-algorithms-supervised-7cb3955c5542

Introduction

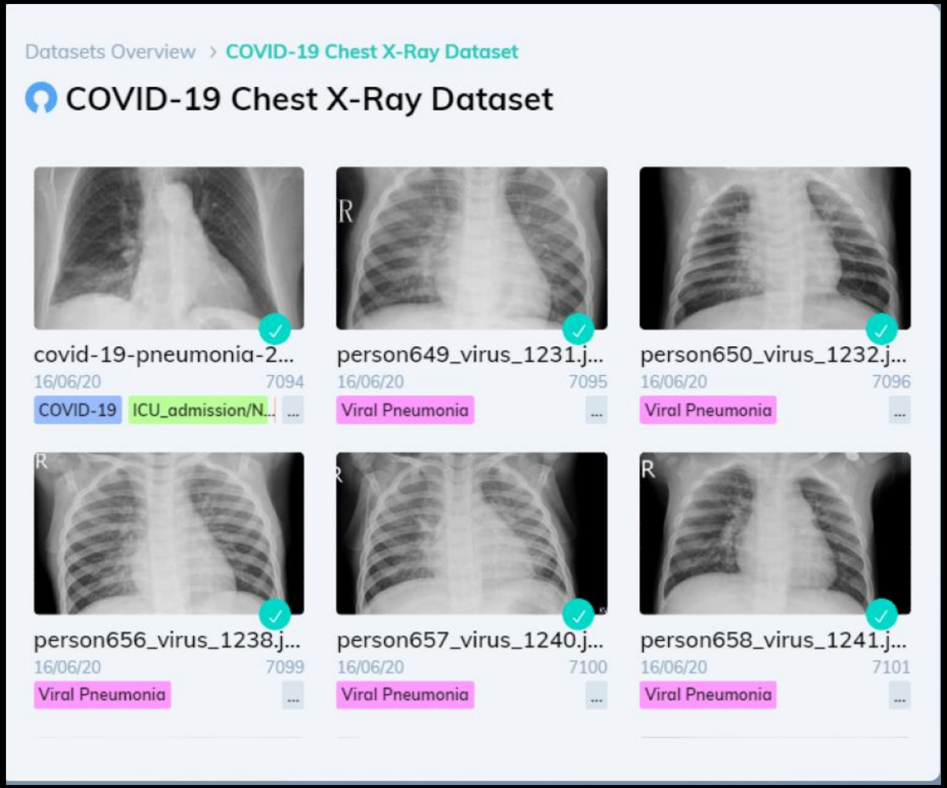
- A view of a tabular dataset

Features					Label
Position	**Experience**	**Skill**	**Country**	**City**	**Salary ($)**
Developer	0	1	USA	New York	103100
Developer	1	1	USA	New York	104900
Developer	2	1	USA	New York	106800
Developer	3	1	USA	New York	108700
Developer	4	1	USA	New York	110400
Developer	5	1	USA	New York	112300
Developer	6	1	USA	New York	114200
Developer	7	1	USA	New York	116100
Developer	8	1	USA	New York	117800
Developer	9	1	USA	New York	119700
Developer	10	1	USA	New York	121600

https://magsimba.com/jcss.asp?iid=113664575&cid=28

Introduction

- A dataset can be images



https://imerit.net/blog/22-free-image-datasets-for-computer-vision-all-pbm/

https://www.v7labs.com/blog/computer-vision-datasets

Introduction

- **Exercise**:
  If a dataset is a set of images, what are considered as features and what is considered as a target?

Content

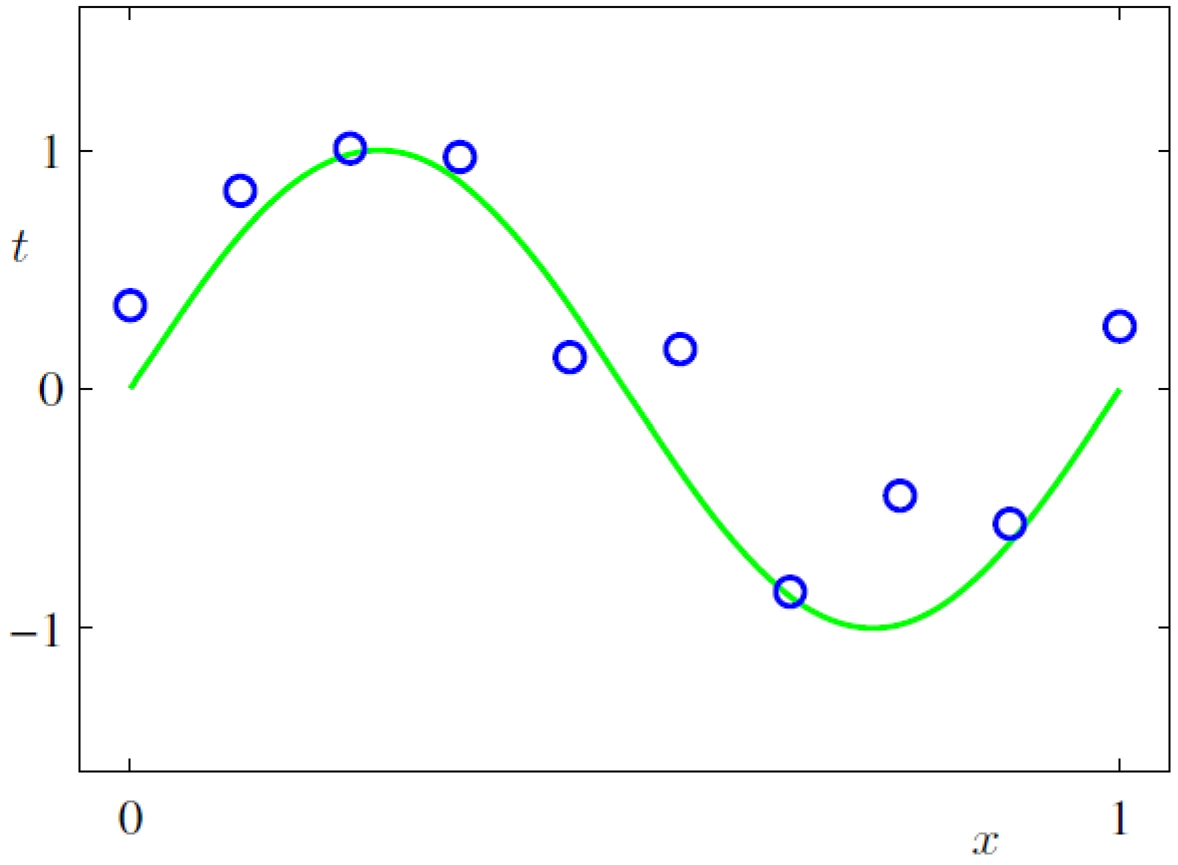
Content
Introduction
Polynomial Curve Fitting
Bayesian Probabilities
Bayesian Curve Fitting

Polynomial Curve Fitting

- We start with a regression example, given a real-valued input x, we want to predict it's target value t.
- We will create an artificial dataset:
  - The number of samples is 10 data points.
  - The feature vector is $\mathbf{x} = (x_1, x_2, \dots x_n)^T$ is a value in range [0, 1]
  - The target vector is $\boldsymbol{t} = (t_1, t_2, \dots, t_n)^T$ derived from the function $\sin(2\pi x)$ with added noise.
- Our goal is to **learn** from the given training data to discover the underlying function of $\sin(2\pi x)$.
  - If we discover the underlying function, we can predict the target value $\hat{t}$ given a new value $\hat{x}$.

Polynomial Curve Fitting

Figure 1.2 Plot of a training data set of $N = 10$ points, shown as blue circles, each comprising an observation of the input variable x along with the corresponding target variable t. The green curve shows the function $\sin(2\pi x)$ used to generate the data. Our goal is to predict the value of t for some new value of x, without knowledge of the green curve.



Polynomial Curve Fitting

- We will implement a **linear regression** model.
- The linear regression model is defined as:

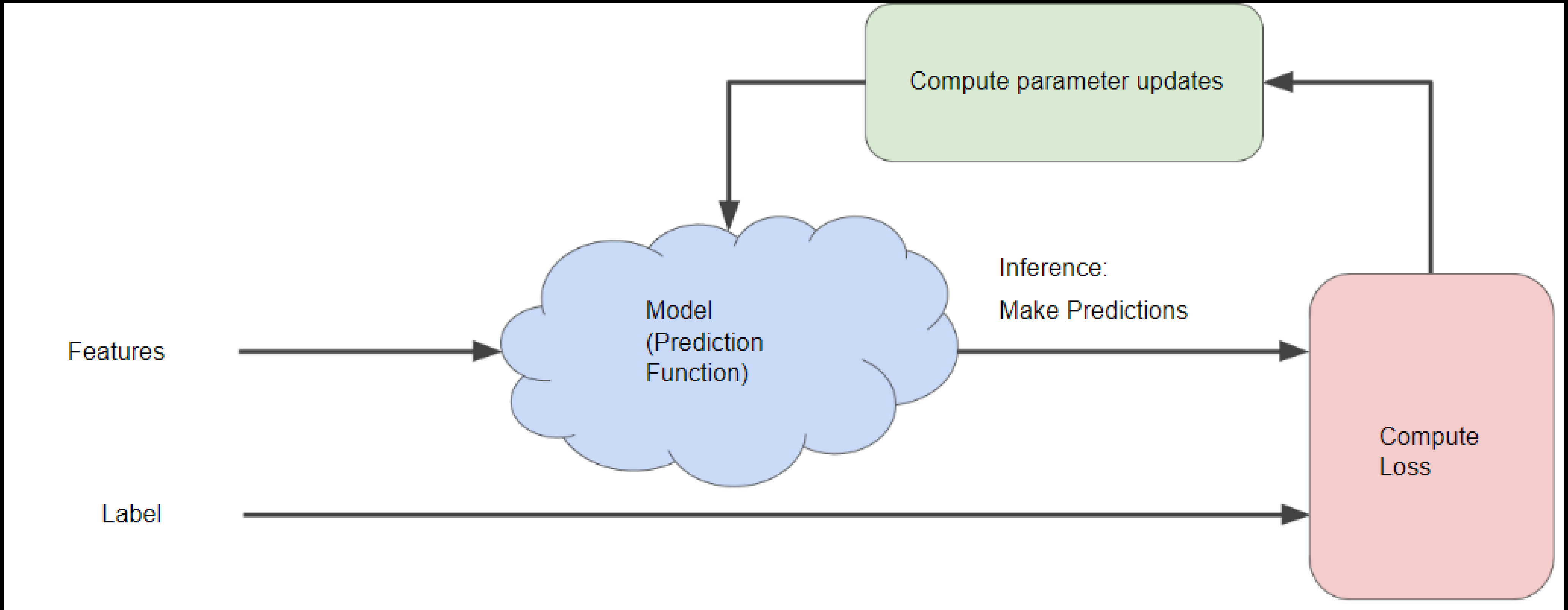
$$y(x, \boldsymbol{w}) = w_0 + w_1 x + w_2 x^2 \ldots + w_M x^M = \sum_{j=0}^{M} w_j x^j$$

  - x is the input vector (features)
  - $\boldsymbol{w}$ is the weights vector.
- The weights are the coefficient that are multiplied with the input vector ($\boldsymbol{x}$) to produce an output (t).
  - So, the basic idea of **learning** is to **find the best set of weights that gives the most accurate results.**

Polynomial Curve Fitting

- To find the best set of weights, we have to minimize an error function.
- The process is as follows
  1. Get the dataset.
  2. Let the machine learn from the data (finding the weights).
  3. Compute the model's error – compare the output of the model ($\hat{t}$) with the true target value of the input (t).
  4. If the error is high, repeat the learning process.
  5. If the error is low, stop the learning process.

Polynomial Curve Fitting



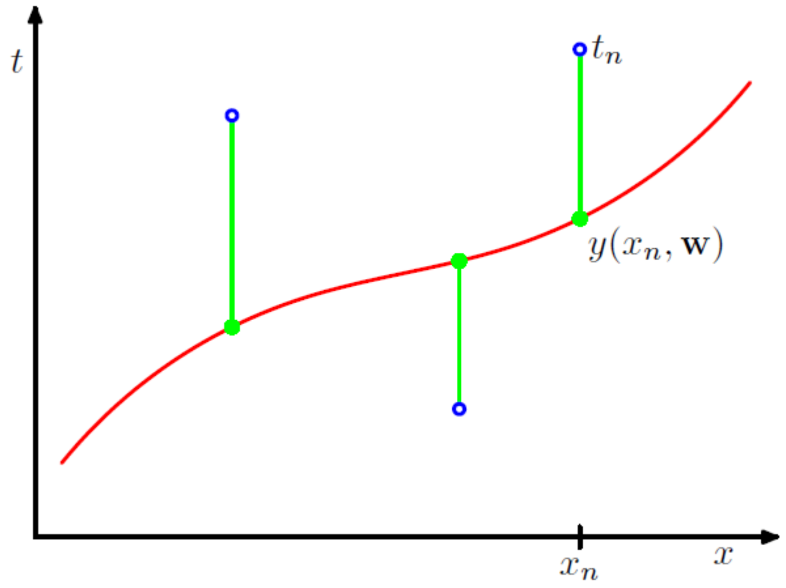
https://developers.google.com/machine-learning/crash-course/reducing-loss/an-iterative-approach

Polynomial Curve Fitting

- To measure the error of the model, we use an error function called *sum of the squares*

$$E(\boldsymbol{w}) = \frac{1}{2}\sum_{n=1}^{N}\{y(x_n, \boldsymbol{w}) - t_n\}^2$$

Figure 1.3 The error function (1.2) corresponds to (one half of) the sum of the squares of the displacements (shown by the vertical green bars) of each data point from the function $y(x, \mathbf{w})$.



Polynomial Curve Fitting

- Interactive demo https://developers.google.com/machine-learning/crash-course/reducing-loss/playground-exercise

Polynomial Curve Fitting

- The linear regression model is a polynomial function:

$$y(x, \boldsymbol{w}) = w_0 + w_1 x + w_2 x^2 \ldots + w_M x^M = \sum_{j=0}^{M} w_j x^j$$

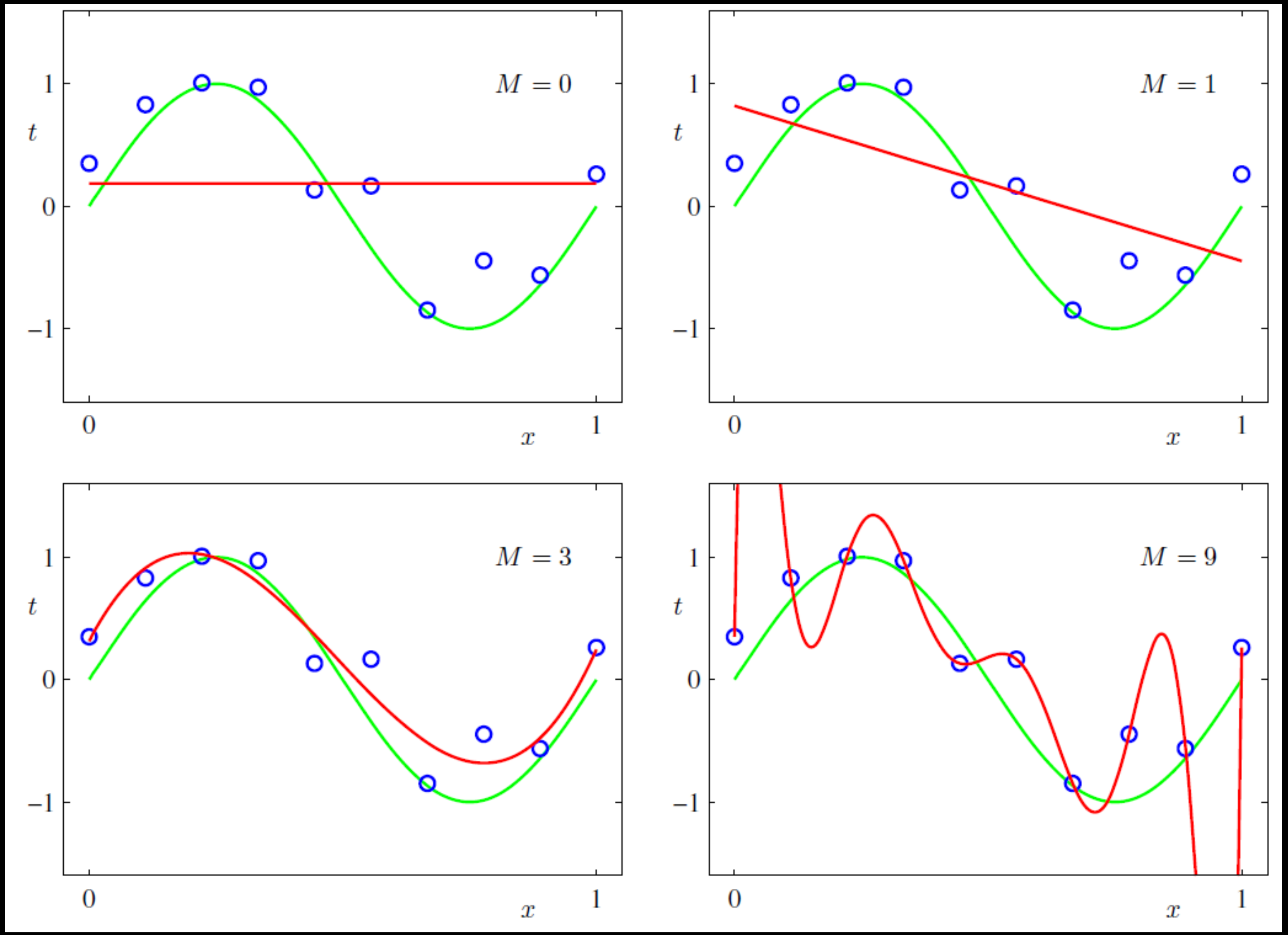
  - When building the model, we have to choose the order M of the polynomial
- Types of Polynomial functions: (https://www.desmos.com/calculator)

Type	Form
Zero Polynomial Function	$P(x) = a = ax^0$
Linear Polynomial Function	$P(x) = ax + b$
Quadratic Polynomial Function	$P(x) = ax^2 + bx + c$
Cubic Polynomial Function	$ax^3 + bx^2 + cx + d$
Quartic Polynomial Function	$ax^4 + bx^3 + cx^2 + dx + e$

Polynomial Curve Fitting

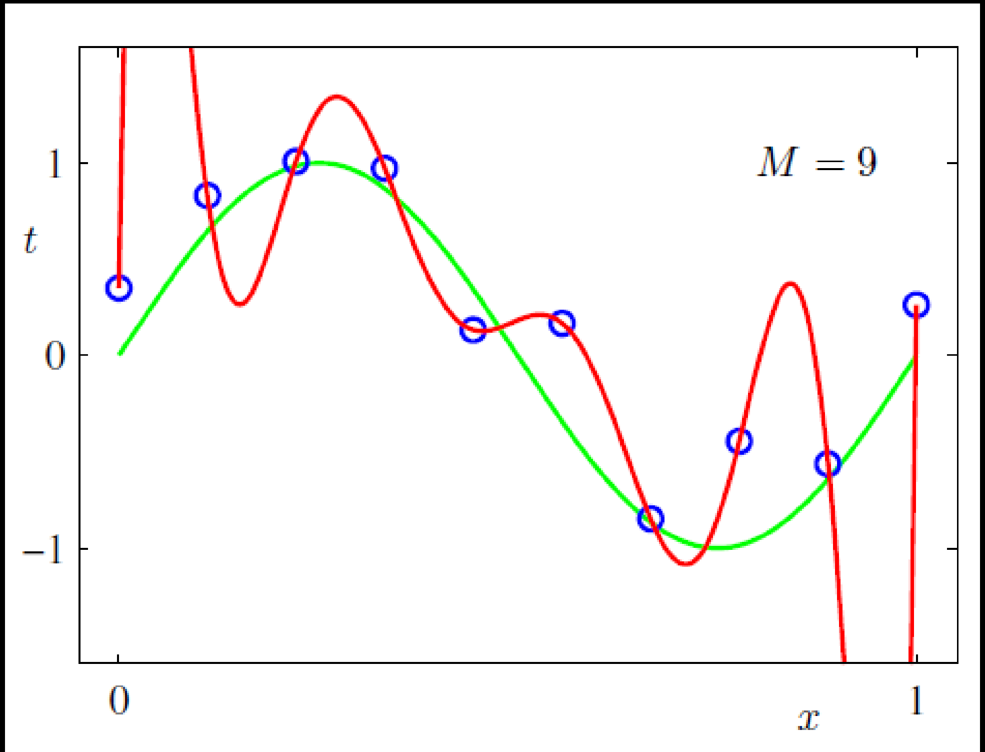
- Four examples of the results of fitting polynomials having orders M = 0, 1, 3, and 9.

- Exercise: which is the best model that represent the data?



Polynomial Curve Fitting

- This a very complex model – *overfitting*.
  - The model only fits the training dataset; it will not recognize new samples.
  - If the model is given a new value (not part of the training set), it will give wrong output.
- We want to achieve a good **generalization**.
  - Generalization means that the model be accurate for unknown, new samples.

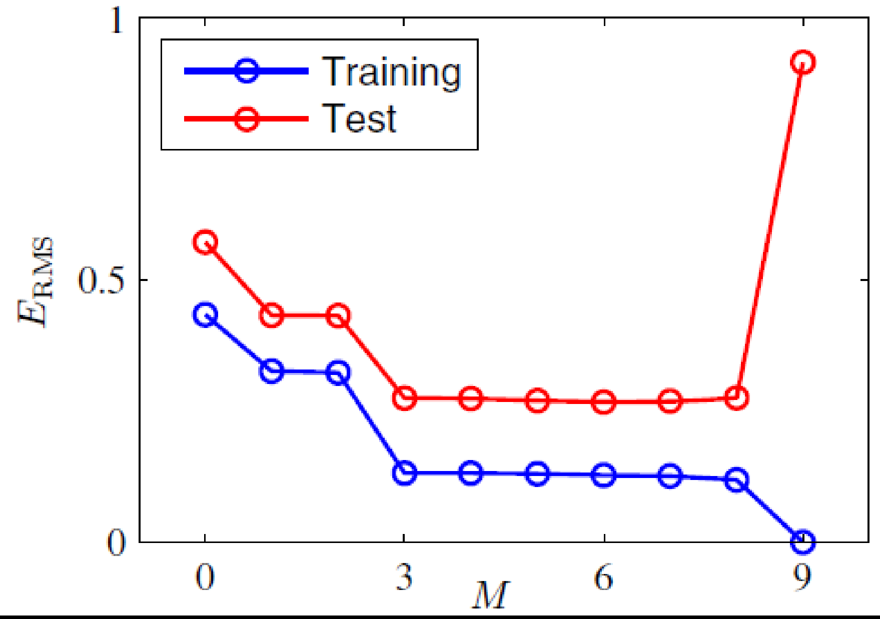


Polynomial Curve Fitting

- Using the RMSE error function to compute error of a model of degree M.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(prediction - true)^2}{N}}$$

Figure 1.5 Graphs of the root-mean-square error, defined by (1.3), evaluated on the training set and on an independent test set for various values of M.



Go to Code

Polynomial Curve Fitting

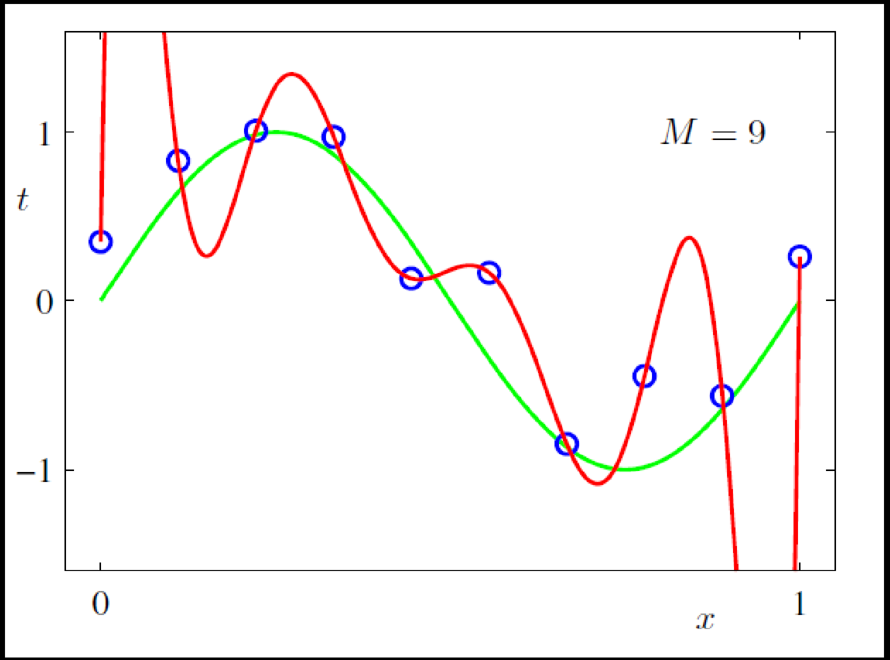
- To avoid overfitting, apply **regularization**.
- Regularization is a penalty term added to the error function to prevent coefficient from reaching large values.
  - Error function with regularization

$$\tilde{E}(\boldsymbol{w}) = \frac{1}{2}\sum_{n=1}^{N}\{y(x_n, \boldsymbol{w}) - t_n\}^2 + \frac{\lambda}{2}||\boldsymbol{w}||^2$$

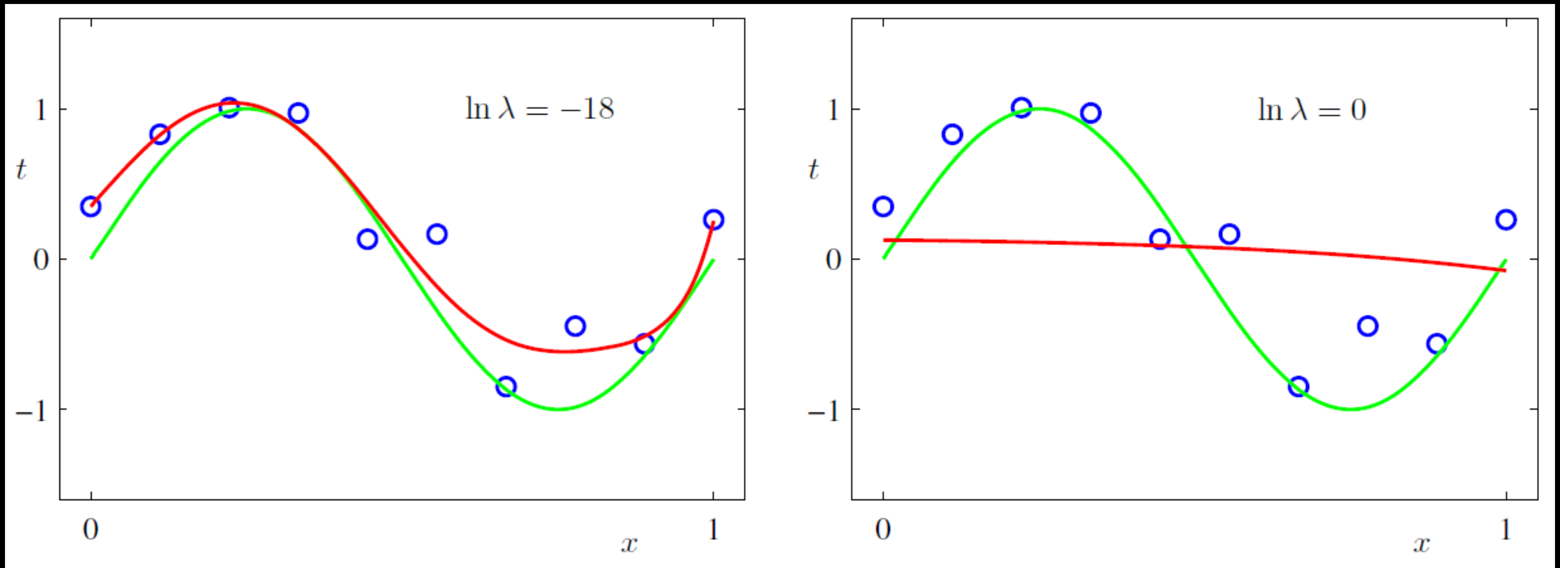
    - λ is a parameter that controls the strength of the regularization
    - $||\boldsymbol{w}||^2 = w_0^2 + w_1^2 + w_2^2 + \cdots w_m^2$

Polynomial Curve Fitting

Before regularization



After regularization



Go to Code

Content

Content
Introduction
Polynomial Curve Fitting
Bayesian Probabilities
Bayesian Curve Fitting

Bayesian Probabilities

- Bayes theorem is used to calculate the conditional probabilities.
  - Calculates the probability of the occurrence of an event given the occurrence of another event.
  - Example: what is the probability that the sky will rain *given that* there are clouds.
- Bayes' theorem is used to describe the uncertainty in model parameters, $\boldsymbol{w}$, after observing the dataset.

$$p(\boldsymbol{w}|D) = \frac{p(D|\boldsymbol{w})p(\boldsymbol{w})}{p(D)}$$

$$p(model|dataset) = \frac{p(dataset|model)p(model)}{p(dataset)}$$

Bayesian Probabilities

- Bayes theorem

$$p(\boldsymbol{w}|D) = \frac{p(D|\boldsymbol{w})p(\boldsymbol{w})}{p(D)} \quad \text{or} \quad p(\boldsymbol{w}|D) \propto p(D|\boldsymbol{w}) * p(\boldsymbol{w})$$

  - $p(\boldsymbol{w})$, is the **prior** probability of the weights before observing data.
  - $p(D, \boldsymbol{w})$, is the **likelihood**, expresses how probable the observed dataset is for different settings of the weights $\boldsymbol{w}$.
  - $p(D)$, is the **normalization constant**, ensures that distribution integrates to one.
  - $p(\boldsymbol{w}|D)$, is the **posterior**, represents what parameters are likely after observing the dataset.

Content

Content
Introduction
Polynomial Curve Fitting
Bayesian Probabilities
Bayesian Curve Fitting

Bayesian Curve Fitting

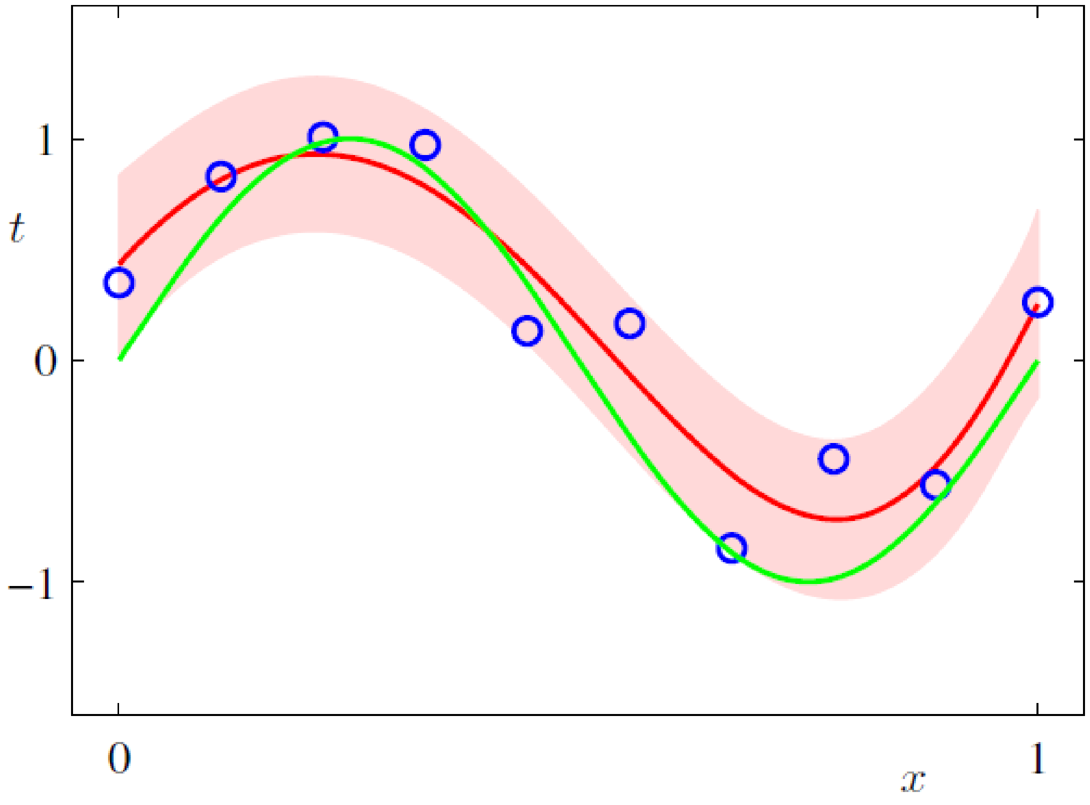
- The dataset D is composed of feature values $\boldsymbol{x}$ and target values $\boldsymbol{t}$. Thus, we can re-write the Bayesian formula as follows:

$$p(\boldsymbol{w}|\boldsymbol{x},\boldsymbol{t},\alpha,\beta) \propto p(\boldsymbol{t}|\boldsymbol{x},\boldsymbol{w},\beta)p(\boldsymbol{w}|\alpha)$$

  Where:
  - D is decomposed into $\boldsymbol{x}$ (feature values) and $\boldsymbol{t}$ (target values)
  - β is a hyperparameter corresponds to the inverse of the variance.
  - α is a hyperparameter controls the distribution of the model's parameters.
  - The hyperparameters α and β can be used to determine a value for the regularization term λ by $\lambda = \frac{\alpha}{\beta}$.

Bayesian Curve Fitting

Figure 1.17 The predictive distribution resulting from a Bayesian treatment of polynomial curve fitting using an $M = 9$ polynomial, with the fixed parameters $\alpha = 5 \times 10^{-3}$ and $\beta = 11.1$ (corresponding to the known noise variance), in which the red curve denotes the mean of the predictive distribution and the red region corresponds to ± 1 standard deviation around the mean.



Go to Code

Task

- What is Feature Extraction?
- What is underfitting?
- What is a neural network? What is *weight decay*?
- What is a hyperparameter?
- Compare the performance of Linear Regression model, Ridge Regression model, and Bayesian Regression model in terms of RMSE value for a dataset of degree 1 to 10. Plot the data and predictions for each model.