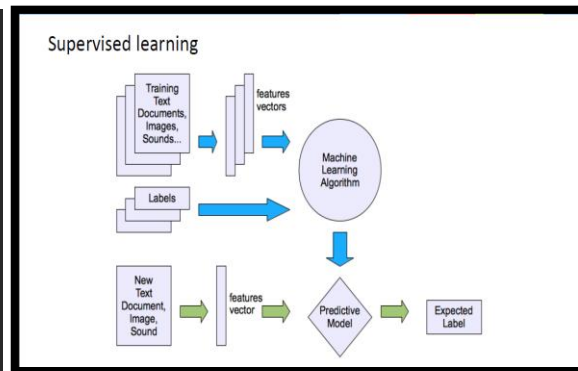
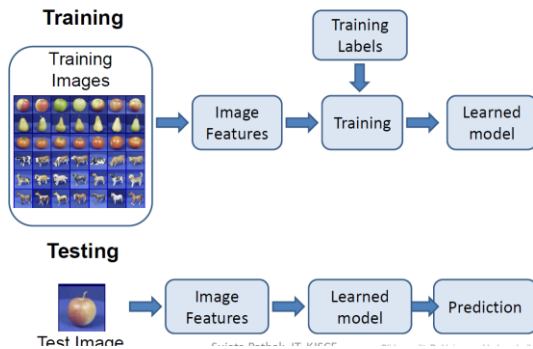


Mod-1

Steps



Generalization

Components of generalization error

Bias: how much the average model over all training sets differ from the true model?

Error due to inaccurate assumptions/simplifications made by the model

Variance: how much models estimated from different training sets differ from each other

Underfitting: model is too “simple” to represent all the relevant class characteristics

High bias and low variance

High training error and high-test error

Overfitting: model is too “complex” and fits irrelevant characteristics (noise) in the data

Low bias and high variance

Low training error and high-test error

Sure, I can explain underfitting and overfitting in machine learning using simple language.

Imagine you're trying to teach a robot to recognize different types of fruits, like apples and oranges. You show the robot pictures of fruits and tell it what each one is. Now, you want the robot to be able to recognize new fruits it has never seen before.

1. **Underfitting:** This is when the robot doesn't learn well enough. It's like if the robot only learned that anything round is an apple. So, even if you show it a picture of an orange, it will still think it's an apple because it didn't learn the differences. In machine learning, underfitting happens when the model is too simple and can't capture the important patterns in the data.
2. **Overfitting:** On the other hand, this is when the robot learns too much from the examples you showed it. It's like if the robot memorized every single fruit you showed it, including all the tiny details and imperfections. When you give it a new fruit, it gets confused because it's trying to match it exactly to what it saw before. In machine learning, overfitting happens when the model is too complex and starts learning the noise or random variations in the data, which is not useful for making predictions on new data.

So, the goal in machine learning is to find the right balance. You want your model to learn enough from the data to make accurate predictions on new, unseen data, without being too simple (underfitting) or too complex (overfitting). It's like teaching the robot to recognize fruits in a way that it can tell apples from oranges, but also correctly identify new fruits it has never encountered before.

Bias-Variance Trade-off

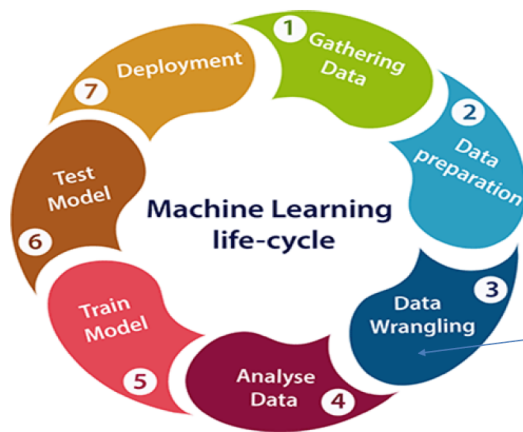
$$E(\text{MSE}) = \text{noise}^2 + \text{bias}^2 + \text{variance}$$

Unavoidable
error

Error due to
incorrect
assumptions

Error due to
variance of training
samples

Process of Machine learning



Data Wrangling:

- Ignoring the missing values
- Removing instances having missing values from the dataset.
- Estimating the missing values of instances using mean, median or mode.
- Removing duplicate instances from the dataset.
- Normalizing the data in the dataset.

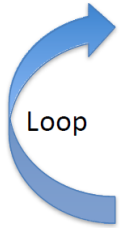
Various Function Representations

- Numerical functions
 - Linear regression
 - Neural networks
 - Support vector machines
- Symbolic functions
 - Decision trees
 - Rules in propositional logic
 - Rules in first-order predicate logic
- Instance-based functions
 - Nearest-neighbor
 - Case-based
- Probabilistic Graphical Models
 - Naïve Bayes
 - Bayesian networks
 - Hidden-Markov Models (HMMs)
 - Probabilistic Context Free Grammars (PCFGs)
 - Markov networks

Various Search/Optimization Algorithms

- Gradient descent
 - Perceptron
 - Backpropagation
- Dynamic Programming
 - HMM Learning
 - PCFG Learning
- Divide and Conquer
 - Decision tree induction
 - Rule learning
- Evolutionary Computation
 - Genetic Algorithms (GAs)
 - Genetic Programming (GP)
 - Neuro-evolution

ML in Practice



- Understand domain, prior knowledge, and goals
- Data integration, selection, cleaning, pre-processing, etc.
- Learn models
- Interpret results
- Consolidate and deploy discovered knowledge

Classification is one of the fundamental tasks in machine learning. It involves assigning predefined categories or labels to input data based on patterns and characteristics in the data. Here's a simple explanation of classification in machine learning:

Imagine you have a bunch of pictures of animals, and you want a computer program to automatically decide whether each picture shows a cat, a dog, or a bird. This task of sorting pictures into categories (cats, dogs, birds) is a classification problem in machine learning.

Classification vs. Prediction

□ Classification:

- predicts categorical class labels
- classifies data (constructs a model) based on the training set and the values (class labels) in a classifying attribute and uses it in classifying new data

□ Prediction:

- models continuous-valued functions, i.e., predicts unknown or missing values

□ Typical Applications

- credit approval
- target marketing
- medical diagnosis
- treatment effectiveness analysis

Aspect	Classification	Prediction (Regression)
Task Objective	Assigning data points to predefined categories or labels.	Estimating a numerical or continuous value.
Output	Discrete categories or labels (e.g., yes/no, classes).	Continuous numerical values (e.g., price, temperature).
Example	Determining if an email is spam or not spam.	Predicting the price of a house based on features.
Algorithms	Logistic Regression, Decision Trees, SVM, K-Nearest Neighbors, etc.	Linear Regression, Random Forest, Neural Networks, etc.
Evaluation Metrics	Accuracy, Precision, Recall, F1 Score, Confusion Matrix, etc.	Mean Absolute Error (MAE), Mean Squared Error (MSE), R-squared, etc.
Loss Function	Cross-Entropy Loss, Hinge Loss, Gini Index, etc.	Mean Squared Error (MSE), Mean Absolute Error (MAE), etc.
Use Cases	Image classification, spam detection, disease diagnosis, etc.	Sales forecasting, stock price prediction, weather forecasting, etc.
Data Type	Categorical or ordinal data (labels or classes).	Numerical data (continuous or discrete).

Classification—A Two-Step Process

- **Model construction:** describing a set of predetermined classes
 - Each tuple/sample is assumed to belong to a predefined class, as determined by the class label attribute
 - The set of tuples used for model construction: training set
 - The model is represented as classification rules, decision trees, or mathematical formulae
- **Model usage:** for classifying future or unknown objects
 - Estimate accuracy of the model
 - The known label of test sample is compared with the classified result from the model
 - Accuracy rate is the percentage of test set samples that are correctly classified by the model
 - Test set is independent of training set, otherwise over-fitting will occur

Classification Techniques

A number of classification techniques are known, which can be broadly classified into the following categories:

1. Statistical-Based Methods
 - Regression
 - Bayesian Classifier
2. Distance-Based Classification
 - K-Nearest Neighbours
3. Decision Tree-Based Classification
 - ID3, C 4.5, CART
5. Classification using Machine Learning (SVM)
6. Classification using Neural Network (ANN)

Naïve Bayesian Classifier

Suppose, for a given instance of X (say $x = (X_1 = x_1)$ and $(X_n = x_n)$).

There are any two class conditional probabilities namely $P(Y = y_i | X=x)$ and $P(Y = y_j | X=x)$.

If $P(Y = y_i | X=x) > P(Y = y_j | X=x)$, then we say that y_i is more stronger than y_j for the instance $X = x$.

The strongest y_i is the classification for the instance $X = x$.

Naïve Bayesian Classifier

Algorithm: Naïve Bayesian Classification

Input: Given a set of k mutually exclusive and exhaustive classes $C = \{c_1, c_2, \dots, c_k\}$, which have prior probabilities $P(C_1), P(C_2), \dots, P(C_k)$.

There are n -attribute set $A = \{A_1, A_2, \dots, A_n\}$, which for a given instance have values $A_1 = a_1, A_2 = a_2, \dots, A_n = a_n$

Step: For each $c_i \in C$, calculate the class condition probabilities, $i = 1, 2, \dots, k$

$$p_i = P(C_i) \times \prod_{j=1}^n P(A_j = a_j | C_i)$$

$$p_x = \max\{p_1, p_2, \dots, p_k\}$$

Output: C_x is the classification

Bayesian Classification: Why?

- Probabilistic learning: Calculate explicit probabilities for hypothesis, among the most practical approaches to certain types of learning problems
- Incremental: Each training example can incrementally increase/decrease the probability that a hypothesis is correct. Prior knowledge can be combined with observed data.
- Probabilistic prediction: Predict multiple hypotheses, weighted by their probabilities
- Standard: Even when Bayesian methods are computationally intractable, they can provide a standard of optimal decision making against which other methods can be measured

Example of Naïve Bayesian:

Unknown sample---- { Red, SUV, Domestic, ? }

Example No.	Color	Type	Origin	Stolen?
1	Red	Sports	Domestic	Yes
2	Red	Sports	Domestic	No
3	Red	Sports	Domestic	Yes
4	Yellow	Sports	Domestic	No
5	Yellow	Sports	Imported	Yes
6	Yellow	SUV	Imported	No
7	Yellow	SUV	Imported	Yes
8	Yellow	SUV	Domestic	No
9	Red	SUV	Imported	No
10	Red	Sports	Imported	Yes

Color	
$P(\text{Red} \text{Yes})=3/5$	$P(\text{Red} \text{No})=2/5$
$P(\text{Yellow} \text{Yes})=2/5$	$P(\text{Yellow} \text{No})=3/5$
Type	
$P(\text{SUV} \text{Yes})=1/5$	$P(\text{SUV} \text{No})=3/5$
$P(\text{Sports} \text{Yes})=4/5$	$P(\text{Sports} \text{No})=2/5$
Origin	
$P(\text{Domestic} \text{Yes})=2/5$	$P(\text{Domestic} \text{No})=3/5$
$P(\text{Imported} \text{Yes})=3/5$	$P(\text{Imported} \text{No})=2/5$

$v = \text{Yes}$ -

$$P(\text{Yes}) * P(\text{Red} | \text{Yes}) * P(\text{SUV} | \text{Yes}) * P(\text{Domestic}|\text{Yes})$$

$$= 5/10 * 3/5 * 2/5 * 1/5 = 0.024$$

and for

$v = \text{No}$ -

$$P(\text{No}) * P(\text{Red} | \text{No}) * P(\text{SUV} | \text{No}) * P(\text{Domestic} | \text{No})$$

$$= 5/10 * 2/5 * 3/5 * 3/5 = 0.072$$

Since $0.072 > 0.024$, our example gets classified as 'NO'

Issues (1): Data Preparation

□ Data cleaning

- Preprocess data in order to reduce noise and handle missing values

□ Relevance analysis (feature selection)

- Remove the irrelevant or redundant attributes

□ Data transformation

- Generalize and/or normalize data

Issues (2): Evaluating Classification Methods

- Predictive accuracy
- Speed and scalability
 - time to construct the model
 - time to use the model
- Robustness
 - handling noise and missing values
- Scalability
 - efficiency in disk-resident databases
- Interpretability:
 - understanding and insight provided by the model
- Goodness of rules
 - decision tree size
 - compactness of classification rules

Types of errors

		Prediction	
		Edge	Not edge
Ground Truth	Edge	True Positive	False Negative
	Not Edge	False Positive	True Negative

Sensitivity versus specificity-

two different measures of a binary classification model. -

Sensitivity-

The true positive rate measures how often we classify an input record as the positive class and its correct classification.

This also is called sensitivity , or recall;

Sensitivity quantifies how well the model avoids false negatives.

$$\text{Sensitivity} = \text{TP} / (\text{TP} + \text{FN})$$

Sensitivity versus specificity-

Specificity-Specificity quantifies how well the model avoids false positives.

$$\text{Specificity} = \text{TN} / (\text{TN} + \text{FP})$$

Accuracy-

Accuracy is the degree of closeness of measurements of a quantity to that quantity's true value.

$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{FP} + \text{FN} + \text{TN})$$

Accuracy can be misleading in the quality of the model when the class imbalance is high.

Precision

The degree to which repeated measurements under the same conditions give us the same results in the context of science and statistics.

Positive prediction value.

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP})$$

A measurement can be accurate yet not precise, not accurate but still precise, neither accurate nor precise, or both accurate and precise.

We consider a measurement to be valid if it is both accurate and precise.

Recall-

Same as sensitivity and is also known as the true positive rate or the hit rate.

F1 score-

In binary classification we consider the F1 score (or F-score, F-measure) to be a measure of a model's accuracy.

Harmonic mean of both the precision and recall measures (described previously) into a single score:

$$F1 = 2TP / (2TP + FP + FN)$$

Sensitivity and Specificity

Count up the total number of each label (TP, FP, TN, FN) over a large dataset. In ROC analysis, we use two statistics:

$$\text{Sensitivity} = \frac{TP}{TP+FN}$$

Can be thought of as the likelihood of spotting a positive case when presented with one.

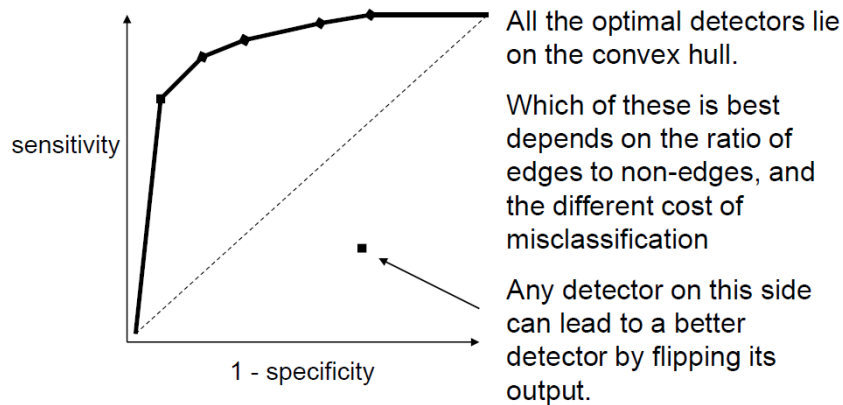
Or... the proportion of edges we find.

$$\text{Specificity} = \frac{TN}{TN+FP}$$

Can be thought of as the likelihood of spotting a negative case when presented with one.

Or... the proportion of non-edges that we find

ROC Analysis



Take-home point : You should always quote sensitivity and specificity for your algorithm, if possible plotting an ROC graph. Remember also though, any statistic you quote should be an average over a suitable range of tests for your algorithm.

Sujata Pany, 2018

Mod-2

2. Basis Functions:

- Linear basis function models extend linear regression by introducing basis functions. Basis functions are mathematical functions that transform the original input features into a new set of features. These transformed features are then used in the linear model.
- The choice of basis functions is flexible and depends on the specific problem. Common basis functions include polynomial functions (e.g., quadratic, cubic), Gaussian functions, trigonometric functions, and more. The choice of basis functions allows the model to capture nonlinear relationships in the data.

LINEAR BASIS FUNCTION MODELS

- **Constructing the Linear Basis Function-**

- The basic linear model for regression is a model that involves a linear combination of the input variables:

$$y(x, w) = w_0 + w_1x_1 + w_2x_2 + \dots + w_Dx_D$$

where $x = (x_1, x_2, \dots, x_D)^T$

- This is what is generally known as linear regression.
- The key attribute of this function-
 - It is a linear function of the parameters w_0, w_1, \dots, w_D and the input variable x_1 .
 - Being a linear function of the input variable x , limits the usefulness of the function.
 - Most of the observations that may be encountered does not necessarily follow a linear relationship.
 - To solve this problem consider modifying the model to be a combination of fixed non-linear functions of the input variable.

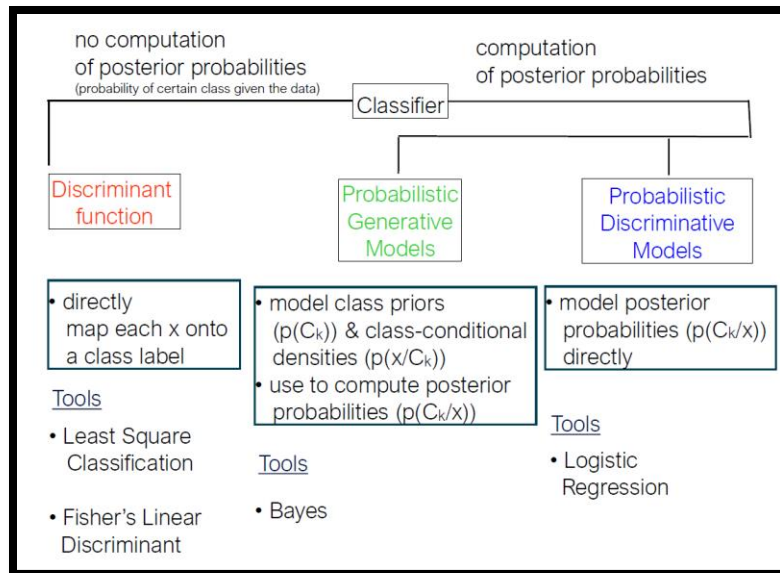
THE LINEAR BASIS FUNCTION

- convenient to define an additional dummy 'basis function'
 $\phi_0(\mathbf{x}) = 1$ so that

$$y(\mathbf{x}, \mathbf{w}) = \sum_{j=0}^{M-1} w_j \phi_j(\mathbf{x}) = \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x})$$

$$\mathbf{w} = (w_0, \dots, w_{M-1})^T \text{ and } \boldsymbol{\phi} = (\phi_0, \dots, \phi_{M-1})^T$$

- In many practical applications of pattern recognition-some form of fixed pre-processing, or feature extraction, to the original data variables
- If the original variables comprise the vector \mathbf{x} , then the features can be expressed in terms of the basis functions $\{\phi_j(\mathbf{x})\}$.

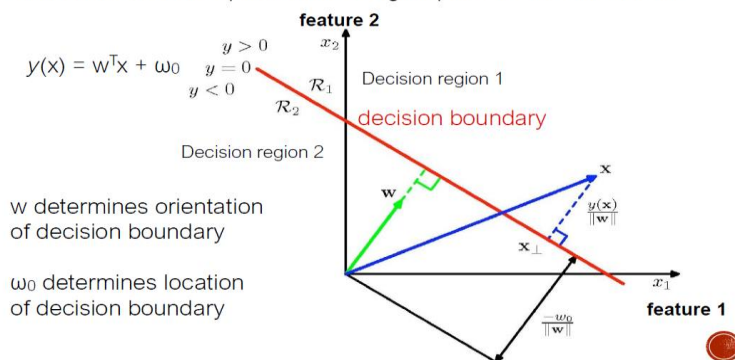


PROS AND CONS OF THE THREE APPROACHES

- Generative models**
 - provide a probabilistic model of *all* variables that allows to synthesize new data – but -
 - generating all this information is computationally expensive and complex and is not needed for a simple classification decision
- Discriminative models**
 - provide a probabilistic model for the target variable (classes) conditional on the observed variables
 - this is usually sufficient for making a well-informed classification decision without the disadvantages of the simple Discriminant Functions

DISCRIMINANT FUNCTIONS

- Functions that are optimized to assign input x to one of k classes



MULTIPLE LINEAR REGRESSION

$$\underline{y} = \underline{\alpha + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_i x_i}$$

Predicted

Response variable

Outcome variable

Dependent

Predictor variables

Explanatory variables

Covariables

Independent variables

TRANSFORMATION

$$P(y|x) = \frac{e^{\alpha + \beta x}}{1 + e^{\alpha + \beta x}}$$

$$\frac{P(y|x)}{1 - P(y|x)}$$

$$\ln \left[\frac{P(y|x)}{1 - P(y|x)} \right] = \alpha + \beta x$$


logit of $P(y|x)$

✓ α = log odds of disease
in unexposed

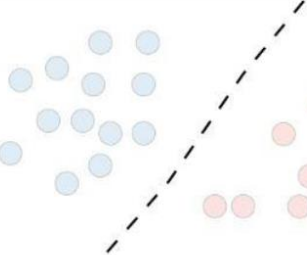
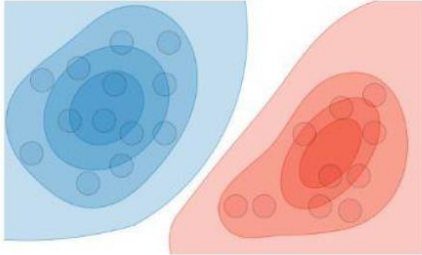
✓ β = log odds ratio associated
with being exposed

✓ e^{β} = odds ratio



Aspect	Discriminative Models	Generative Models
Model Objective	Model the conditional probability of the target class given the input data.	Model the joint probability distribution of both input data and target class.
Use Cases	Typically used for classification tasks, such as text classification, object recognition, sentiment analysis.	Used for tasks involving data generation, denoising, imputation, anomaly detection, and more.
Model Complexity	Often simpler, with fewer parameters, as they focus on the decision boundary.	Typically more complex, as they model both the data distribution and class distribution.
Data Efficiency	Can be more data-efficient since they focus on decision boundaries.	May require larger datasets to model joint distributions effectively.
Interpretability	Coefficients or feature importance scores provide direct interpretability.	Interpretation can be more challenging due to modeling joint distributions.
Examples	Logistic Regression, Support Vector Machines (SVMs), Neural Networks for Classification.	Gaussian Mixture Models (GMMs), Hidden Markov Models (HMMs), Variational Autoencoders (VAEs).
Goal	Find the best boundary to separate classes.	Model how data is generated, allowing for sampling and data generation.

Latent Variables	Less common to incorporate latent variables.	Often involve explicit or implicit modeling of latent variables.
Example Applications	Email spam detection, image classification, sentiment analysis.	Image and text generation, anomaly detection, data imputation.

	Discriminative model	Generative model
Goal	Directly estimate $P(y x)$	Estimate $P(x y)$ to then deduce $P(y x)$
What's learned	Decision boundary	Probability distributions of the data
Illustration		
Examples	Regressions, SVMs	GDA, Naive Bayes

Linear Regression	Logistic Regression
Linear regression is used to predict the continuous dependent variable using a given set of independent variables.	Logistic Regression is used to predict the categorical dependent variable using a given set of independent variables.
Linear Regression is used for solving Regression problem.	Logistic regression is used for solving Classification problems.
In Linear regression, we predict the value of continuous variables.	In logistic Regression, we predict the values of categorical variables.
In linear regression, we find the best fit line, by which we can easily predict the output.	In Logistic Regression, we find the S-curve by which we can classify the samples.
Least square estimation method is used for estimation of accuracy.	Maximum likelihood estimation method is used for estimation of accuracy.
The output for Linear Regression must be a continuous value, such as price, age, etc.	The output of Logistic Regression must be a Categorical value such as 0 or 1, Yes or No, etc.
In Linear regression, it is required that relationship between dependent variable and independent variable must be linear.	In Logistic regression, it is not required to have the linear relationship between the dependent and independent variable.

LOGISTIC REGRESSION

Data: Inputs are continuous vectors of length M. Outputs are discrete.

$$\mathcal{D} = \{\mathbf{x}^{(i)}, y^{(i)}\}_{i=1}^N \text{ where } \mathbf{x} \in \mathbb{R}^M \text{ and } y \in \{0, 1\}$$

Model: Logistic function applied to dot product of parameters with input vector.

$$p_{\theta}(y = 1|\mathbf{x}) = \frac{1}{1 + \exp(-\theta^T \mathbf{x})}$$

Learning: finds the parameters that minimize some objective function. $\theta^* = \underset{\theta}{\operatorname{argmin}} J(\theta)$

Prediction: Output is the most probable class.

$$\hat{y} = \underset{y \in \{0,1\}}{\operatorname{argmax}} p_{\theta}(y|\mathbf{x})$$

Probabilistic Discriminative Models

Two-class case: $p(C_1 | \mathbf{x}) = \sigma(\mathbf{w}^T \mathbf{x} + w_0)$

Multiclass case:
$$p(C_k | \mathbf{x}) = \frac{e^{\mathbf{w}_k^T \mathbf{x} + w_{k0}}}{\sum_j e^{\mathbf{w}_j^T \mathbf{x} + w_{j0}}}$$

Discriminative approach: use the functional form of the generalized linear model for the posterior probabilities and determine its parameters directly using maximum likelihood.

TYPES OF LINEAR REGRESSION

- **Simple linear regression:** This involves modeling the relationship between a single input variable (explanatory variable) and a single output variable (response variable). The model is represented by a straight line, and the goal is to find the line that best fits the data.
- **Multiple linear regression:** This involves modeling the relationship between multiple input variables and a single output variable. The model is represented by a straight line, and the goal is to find the line that best fits the data.
- **Polynomial regression:** This involves modeling the relationship between an input variable and an output variable using a polynomial function. The model is represented by a curve, and the goal is to find the curve that best fits the data.

TYPES OF LINEAR REGRESSION

- **Logistic regression:** This is a type of regression used when the output variable is binary (e.g., 0 or 1, Yes or No). The model is used to predict the probability that a given input belongs to one of the two categories.
- **Ridge regression:** This is a variation of multiple linear regression that adds a penalty term to the objective function to discourage the model from overfitting the data.
- **Lasso regression:** This is another variation of multiple linear regression that adds a penalty term to the objective function to discourage the model from overfitting the data. Unlike ridge regression, lasso regression can zero out some of the coefficients, effectively removing some of the input variables from the model.

LINEAR REGRESSION-EXAMPLE

$$b_0 = \frac{(\sum y)(\sum x^2) - (\sum x)(\sum xy)}{n(\sum x^2) - (\sum x)^2}$$

$$b_1 = \frac{n(\sum xy) - (\sum x)(\sum y)}{n(\sum x^2) - (\sum x)^2}$$

Estimated
(or predicted)
Y value for
observation i

Estimate of
the regression
intercept

Estimate of the
regression slope

Value of X for
observation i

$$\hat{Y}_i = b_0 + b_1 X_i$$