Empirical Risk Minimization

INDRAPRASTHA INSTITUTE *of*
INFORMATION TECHNOLOGY
DELHI

Learning Process

Three components

1. Generator of random vectors x, i.i.d. from a fixed but unknown distribution $P(x)$

2. A supervisor (oracle/astrologer) which returns an output vector y, for every input vector x, as per the conditional distribution $P(y/x)$, also fixed but unknown.

3. A learning machine capable of implementing a set of functions

$$f(x, w), w \epsilon W$$

Learning Process

- The learning problem is to choose from the given set of functions the one which best approximates the supervisor's response.

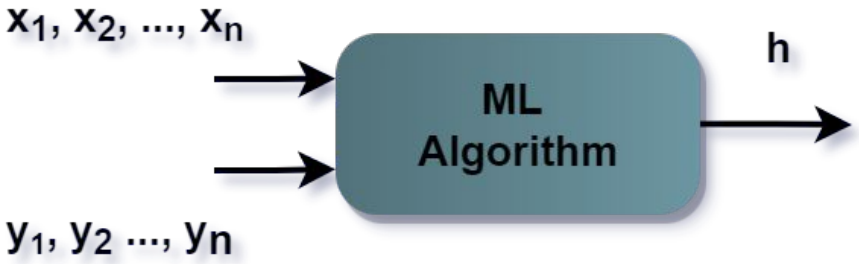
- The selection is based on training samples

$$(x_i, y_i); i = 1, 2, 3...l$$

Problem Setting

- Dataset is a set of possible instances $\mathbf{D} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$
- $\mathbf{x_i} \in \mathbf{X}$**:** Each sample is a vector with $\mathbf{R^d}$ drawn from distribution ***P(x)***
- $\mathbf{y_i} \in \mathbf{Y}$:
  - Classification: Each label is a single integer value out of two [binary] or more classes [multiclass]
  - Regression: Y = R [real number]
- Unknown target function **f : X -> Y** as distribution ***P(y/x)***
- Set of function hypotheses H={ h | h : X->Y }
- **Input:** Training examples $\{<x_i, y_i>\}$
- **Output:** Hypothesis $h \in H$ that best approximates target function f

Training vs Testing



Training Phase

Sample Data

	Person	height(in feet)	weight(in lbs)	foot size(in inches)
0	male	6.00	180	12
1	male	5.92	190	11
2	male	5.58	170	12
3	male	5.92	165	10
4	female	5.00	100	6
5	female	5.50	150	8

- X = <height, weight, foot size>
- Y = <male, female> | <0, 1> | <-1, +1>
- A sample instance (x_1, y_1) = (<6.00, 180, 12>, male)
- Dimensionality d in $X \in R^d = 3$
- Unknown target function f: height, weight, foot size -> male/female

How to choose h?

- Randomly:
  - Advantage: Really fast
  - Disadvantage: Terrible performance

- Scan entire **H** and pick the best:
  - Advantage: Great performance
  - Disadvantage: Terribly slow

- Intelligent Way: Learn it using the performance metric
  - Loss functions help to evaluate the performance of the $\mathbf{h_i}$**'s** $\in$ **H** to identify the **optimal h.**

Loss Functions

- To choose the best function, it makes sense to minimize a loss (or cost or discrepancy) between the response of the supervisor and the learning machine, given an input (x, y)

$$\mathcal{L}(y, f(x))$$

- We want to minimize the loss over all samples

$$L(h) = \frac{1}{n} \sum_{i=1}^{n} l(f(x_i), y_i)$$

- Loss functions are always non-negative.
  - Hence, minimum possible loss is which means we are not making any mistakes.

Loss Functions - Classification

- **0-1 Loss:** Binary Classification with equal weights on misclassification

$$L(h) = \frac{1}{n}\sum_{i=1}^{n} l_{01}(f(x_i), y_i)$$

$$l_{01}(f(x_i), y_i) = \begin{cases} 0, if & f(x_i) = y_i \\ 1, if & f(x_i) \neq y_i \end{cases}$$

ID	Model Prediction $f(x_i)$	Ground Truth y_i	Loss $l_{01}(f(x_i, y_i))$
Mango 1	Good	Good	0
Mango 2	Good	Bad	1
Mango 3	Bad	Good	1
Mango 4	Bad	Bad	0
Total			**0.5**

Loss Functions - h_1 or h_2

- The aim of the function is to select a hypothesis with the lowest loss.

ID	Model Prediction [h_1] $f(x_i)$	Model Prediction [h_2] $f(x_i)$	Ground Truth y_i	Loss $l_{01}(f(x_i, y_i))$	Loss $l_{01}(f(x_i, y_i))$
M1	Good	Good	Good	0	0
M2	Good	Bad	Bad	1	0
M 3	Bad	Good	Good	1	0
M4	Bad	Bad	Bad	0	0
Total Loss				**0.5**	**0**

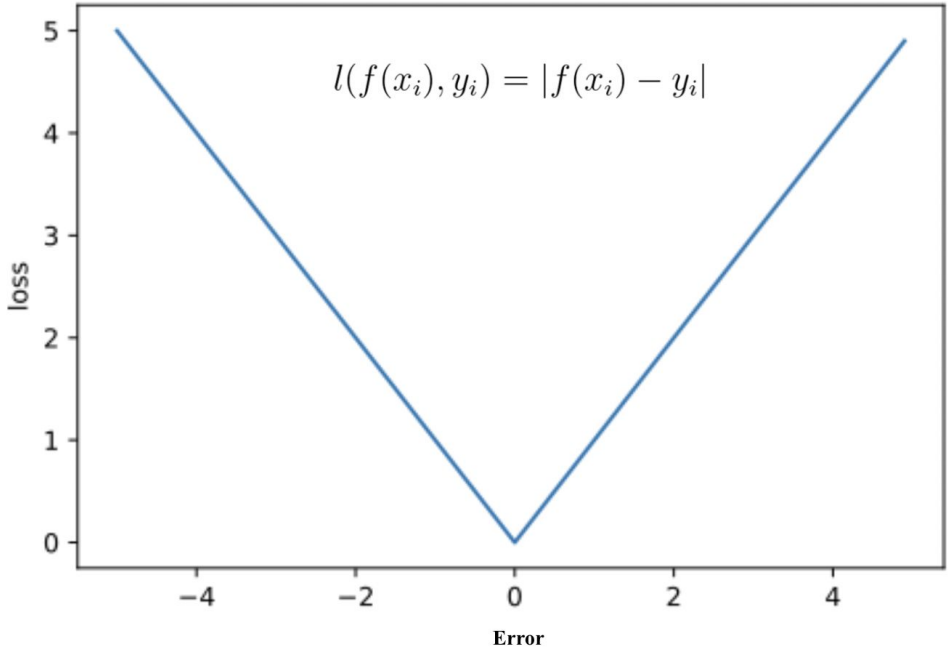
Loss Functions - Unequal Weights

- Classification with unequal weights on misclassification
  - Minimize a 0-10^7-500 loss

$$l(f(x_i), y_i) = \begin{cases} 0, if & f(x_i) = y_i \\ 500, if & f(x_i) = 1, y_i = 0 \\ 10^7, if & f(x_i) = 0, y_i = 1 \end{cases}$$

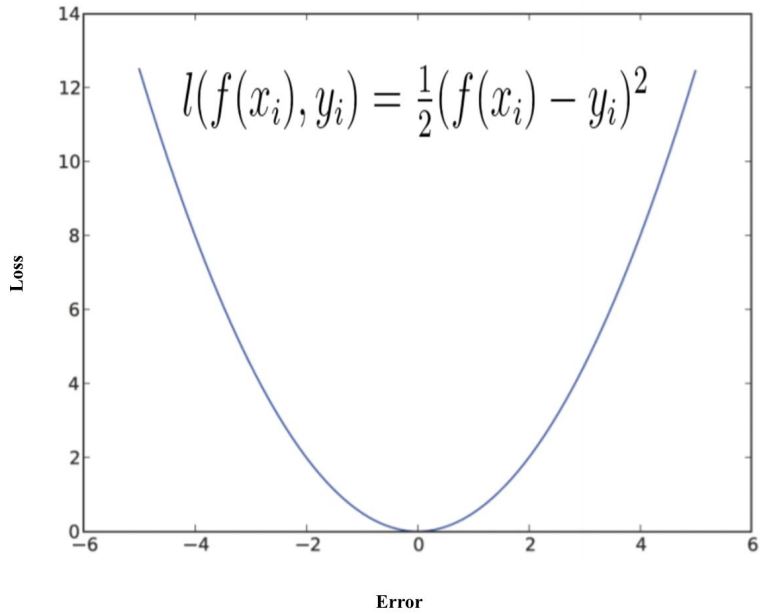
Loss functions: Regression - L_1 Loss

$$l(f(x_i), y_i) = |f(x_i) - y_i|$$



Loss functions: Regression - L_2 Loss

$$l(f(x_i), y_i) = \frac{1}{2}(f(x_i) - y_i)^2$$



Mean Absolute Error vs. Mean Square Error

- Mean Absolute Error (MAE)

$$L(h) = MAE = \frac{1}{n}\sum_{i=1}^{n}|f(x_i) - y_i|$$

- Mean Square Error (MSE)

$$L(h) = MSE = \frac{1}{n}\sum_{i=1}^{n}(f(x_i) - y_i)^2$$

MAE = 1, RMSE = 1.50

ID	Error	|Error|	Error²
1	0	0	0
2	1	1	1
3	2	2	4
4	-0.5	0.5	0.25
5	1.5	1.5	2.25
Total		5	7.50

MSE vs. MAE (L2 loss vs L1 loss)

- Robust to Outliers? L_1 or L_2
  - An individual's height being too tall or small.
- Differentiability?
  - **What:** Continuous and smooth function over a region.
  - **Why Do We Care:** To get the rate of increase/decrease defined at all points.
  - **Buy Why!:** It allows to find the minima/maxima and hence the optimal model.

Generalization

- What about performance on unknown or new data samples?
- Our true goal is to know and minimize the loss of the *unknown test samples* drawn from same distribution *P*

$$h^* = \underset{f(x))}{\operatorname{argmin}} \frac{1}{m-n} \sum_{i=n+1}^{m} l(f(x_i), y_i)$$

- In other words, we want to know and minimize the *Expected Loss l(f(x, y))* and hence the *Risk* associated with function *f*.

$$R(f) = \int \int p(x_i, y_i) l(f(x_i), y_i) dx dy$$

Expected Loss

- Usually we don't know the test points and their labels in advance!!!
  - We do not know the $p(x_i, y_i)$
  - Hence, we do not know the *expected loss* or the $R(f)$

- ***Our goal:*** *Expected loss should be closer to the actual loss*

- The law of large numbers (LLN) states that if the amount of exposure to losses increases, then the predicted loss will be closer to the actual loss.
  - Example: Insurance in Real Life

Empirical Risk Minimization Principle

- So. by LLN the *statistical risk* associated with function *f* becomes equal to the *empirical risk*

$$\frac{1}{m-n}\sum_{i=n+1}^{m} l(f(x_i), y_i) \overset{m\to\infty}{\to} R_{L,P}(f)$$

- Picking the function *f (via hypothesis h)* that minimizes the *empirical risk* is known as *empirical risk minimization*.

$$h^* = f^* = \underset{f(x)}{\operatorname{argmin}} R_{L,P}(f)$$

- **Our hope:**

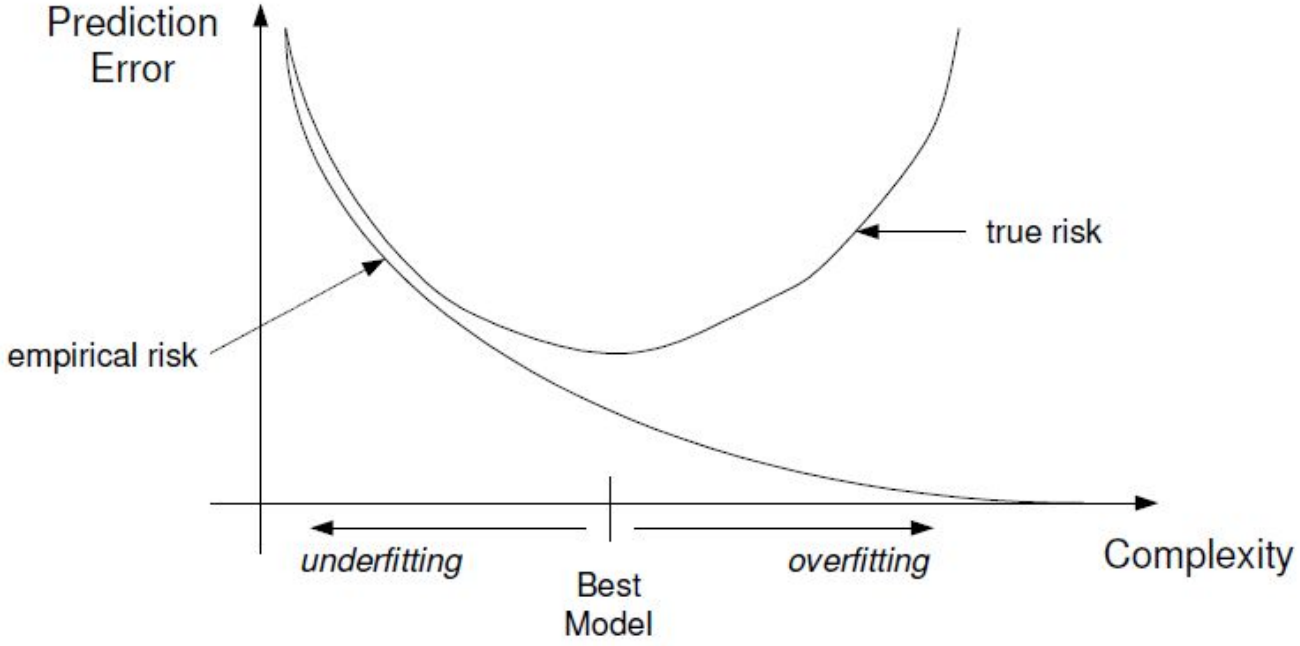
$$\underset{f(x)\epsilon F}{\operatorname{argmin}} R_{L,P}(f) \approx \underset{f(x)\epsilon F}{\operatorname{argmin}} R^{true}_{L,P}(f)$$

Empirical Risk Minimization Principle

Empirical risk minimization depends on following:

1. **How much data we have:** For any given function f, as we get more and more data, we can expect that $R(f) \rightarrow R_{true}(f)$
2. **The true distribution p**: Depending on how "complex" the true distribution is, more or less data may be necessary to get a good approximation of it.
3. **The loss function L**: If the loss function is very "weird"– giving extremely high loss in certain unlikely situations, this can lead to trouble.
4. **The class of functions F**: Roughly speaking, if the size of F is "large", and the functions in F are "complex", this worsens the approximation, all else being equal.

Effect of Function Complexity



- At fixed number of samples, overly complicated models -> overfitting.
- Empirical risk is no longer a good indicator of true risk.

References

- Principles of Risk Minimization for Learning Theory