**Weekly report of lessons**

**Name:** Mayank Kumar
**Roll No:** 19CS30029
**The week:** 23-08-2021 to 29-08-2021

## The topics covered:

Convergence of CE Algorithm: Convergence, Choosing next TE, Classifying new cases in VS; Effect of Incomplete Hypothesis Space; Inductive Bias: Unbiased Learners, IB, IB as Equivalent Deductive System; Computational Complexity of VS; PAC Learning Model; Sample Complexity of a Learner; Theorem of $\in$-exhausting VS; Sample Complexity of infinite H; Handling Noise in Data, Effect of Inductive Bias, Matching Complexities, Occam's Razor, Triple Trade-Off, Model Selection; Decision Tree: Structure, Principle of Construction, Entropy, Information Gain

## Summary topic wise:

Convergence of Candidate Elimination Algorithm:
Guaranteed convergence if there are no errors in training set and there exists hypothesis h in H describing target concept c. When G=S, we get hypothesis h in H describing target concept c.

Choosing next TE: The next TE should be chosen such that it reduces maximally the number of hypotheses in VS.

Classifying new cases in VS: To classify a new case, a voting procedure is used, where confidence lies in the majority vote.

Effect of incomplete hypothesis space: All target functions might not be represented by some h in H, and preceding algorithms work if target function is in H

Unbiased Learners: No limits on representation of hypotheses

Inductive Bias: Any minimal set of assertions B used to logically infer the value c(x) of any instance x from B, D, and x for any target concept c and training examples D
Inductive bias is made explicit in an equivalent deductive system that logically produces the same output.
For rote learner, B = {}

Computational Complexity of VS: S is linear in no. of features and no. of training examples; G is exponential in no. of training examples

Probably Approximately Correct (PAC) learning model:
True error: $\text{error}_D(h) = P(c(x) \neq h(x))$, D: population distribution
A target concept class C, defined over a set of instances X of length n, is PAC-learnable if $\forall$ c $\in$ C, distribution D over X, $0 < \varepsilon < ½$ and $0 < \delta < ½$, learner L with probability at least $(1-\delta)$ outputs hypothesis h $\in$ H such that TrueError $< \varepsilon$, in time polynomial in $1/\varepsilon$, $1/\delta$, n and size(c).

Sample Complexity of a Learner: No. of TE required to get a successful hypothesis with a high probability

Theorem of $\in$-exhausting VS
$P(VS_{H,D}$ is $\varepsilon$-exhausted$) > |H|e^{-\varepsilon m}$
Hence for PAC learner, $|H|e^{-\varepsilon m} \leq \delta$ and $m \geq (1/\varepsilon)\ (\ln|H| + \ln(1/\delta))$

Sample Complexity of infinite H:
Represented by Vapnik-Chervonenkis Dimension: size of largest finite subset in X shattered by H; $VC(H) \leq \log_2|H|$

Handling noise in data: Major sources are imprecision in measurement of features, error in labeling, and missing additional attributes in representation

Effect of Inductive Bias:
Low training error may provide high errors in unseen inputs
Higher the proportion of training samples, better is the model fitting

Matching Complexities:
Low Complexity - higher training and generalization error
High Complexity - low training error, may have high generalization error (overfitting)

Occam's razor: It states that in comparable empirical error, the simplest model is usually the best, and any unnecessary complexity should be shaved off

Triple trade-off: The trade-off between complexity of hypothesis, amount of training data, and generalization error on unseen inputs

Model Selection: Input is divided into training, validation and test set; choose model by keeping low error for training and validation

Decision Trees:
Disjunction of conjunction of attribute literals; can be represented by logical formula as a tree
Each internal node tests an attribute
Each branch corresponds to attribute value
Each leaf node assigns a classification

Decision Tree Structure: Axis parallel lines are drawn to separate the instances of each class

Principle of Decision Tree Construction:
Start with an empty tree by treating the entire dataset as a single box.
Construct by recursively evaluating different features and using at each node the attribute that best splits the data and reduces its impurity by maximum amount
Grow tree just enough for perfect classification

Entropy: E(S) = Sum over $p_i$ * ($-\log_2 p_i$), i = 1 to n
For binary, E(S) = $-(p_+ \log_2 p_+) - (p_- \log_2 p_-)$
Here, $p_i$ = proportion of case belonging to class i

Information Gain:

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

## Concepts challenging to comprehend:

Inductive Bias and PAC Learning Model were a bit difficult to comprehend but got clear when I read the book

## Interesting and exciting concepts:

Decision Tree is an exciting concept as it has a much better algorithm to approximate a discrete-valued target function. Entropy and Information Gain is also quite interesting.

## Concepts not understood:

None, all the concepts were clear to me.

## Any novel idea of yours out of the lessons:

It should be noted that the Decision Tree searches for a complete hypothesis space that is capable of expressing any finite discrete-valued function, while the CE Algorithm searches for an incomplete hypothesis that can express only a subset of potentially teachable concepts.
Also, in Decision Tree, shorter trees should be preferred over longer ones as it would place the attributes with high information gain closer to the root.

----------------------------------------------------------------------------------------------------------------