23IT503:MachineLearning

Mr.V.Anil Kumar
AssistantProfessor
DepartmentofIT&DS

References

TextBooks:

1. TomM.Mitchell,MachineLearning,IndiaEdition2013,McGraw HillEducation.

ReferenceBooks:

1. TrevorHastie,RobertTibshirani,JeromeFriedman,hTheElementsof StatisticalLearning, 2nd edition,springer seriesin statistics.
2. EthemAlpaydın,Introductiontomachinelearning,secondedition,MITpress.

Prerequisites

ForMachineLearningCoursewerecommendthatstudentsmeetthefollowing prerequisites:

- Basicprogrammingskills(inPython)
- Algorithmdesign
- Basicsofprobability&statistics

Content

Unit–1	Introduction, Concept Learning, Decision Tree
Unit–2	LearningArtificialNeuralNetworks-1,ArtificialNeural Networks-2, Evaluating Hypothesis,
Unit–3	BayesianLearning,Computationallearningtheory, Instance Based Learning,
Unit–4	GeneticAlgorithms,LearningSetsofRules, Reinforcement Learning
Unit–5	AnalyticalLearning-1,AnalyticalLearning-2,CombiningInductiveand Analytical Learning

UNIT-1

MachineLearning Introduction

Ever sincecomputers wereinvented,wehavewonderedwhethertheymightbe Madetolearn.Ifwecouldunderstandhowtoprogramthemtolearn-toimprove automatically with experience-the impact would be dramatic.

- Imaginecomputerslearningfrommedicalrecordswhichtreatmentsaremost effectivefornewdiseases
- Houseslearningfromexperiencetooptimizeenergycostsbasedontheoftheir occupants.
- Personalsoftwareassistantslearningtheevolvinginterestsoftheirtohighlight especiallyrelevantstoriesfromtheonlinemorningnewspaper

ExamplesofSuccessful Applicationsof MachineLearning

- Learningtorecognizespokenwords
- Learningtodriveanautonomousvehicle
- Learningtoclassifynewastronomicalstructures
- Learningtoplayworld-classbackgammon

WhyisMachineLearningImportant?

- Sometaskscannotbedefinedwell,exceptbyexamples(e.g., recognizing people).
- Relationshipsandcorrelationscanbehiddenwithinlargeamountsof data.MachineLearning/DataMining maybe abletofind these relationships.
- Humandesignersoftenproducemachinesthatdonotworkaswellas desiredintheenvironmentsinwhichtheyareused.
- Theamountofknowledgeavailableaboutcertaintasksmightbetoolargefor explicitencoding by humans (e.g., medical diagnostic).
- Environmentschangeovertime.
- Newknowledgeabouttasksisconstantlybeingdiscoveredbyhumans.Itmay bedifficulttocontinuouslyre-designsystems“byhand”.

AreasofInfluenceforMachineLearning

- ***Statistics:***Howbesttousesamplesdrawnfromunknownprobabilitydistributionstohelp decidefromwhichdistributionsomenewsampleisdrawn?
- ***Brain Models***: Non-linearelementswithweightedinputs(Artificial NeuralNetworks)havebeensuggestedassimplemodelsofbiologicalneurons.
- ***AdaptiveControlTheory***:Howtodealwithcontrollingaprocesshavingunknown parametersthatmustbeestimatedduringoperation?
- ***Psychology***:Howtomodelhumanperformanceonvariouslearningtasks?
- ***ArtificialIntelligence***:Howtowritealgorithmstoacquiretheknowledgehumansareable toacquire,atleast,aswellashumans?
- ***EvolutionaryModels***:Howtomodelcertainaspectsofbiologicalevolutionto improvetheperformanceofcomputerprograms?

MachineLearning:ADefinition

A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured byP,improveswithexperienceE.

Why“Learn”?

Learningissusedwhen:

- Humanexpertisedoesnotexist(navigatingonMars)
- Humansareunabletoexplaintheirexpertise(speechrecognition)
- Solutionchangesintime(routingonacomputernetwork)
- Solutionneedstobeadaptedtoparticularcases(userbiometrics)

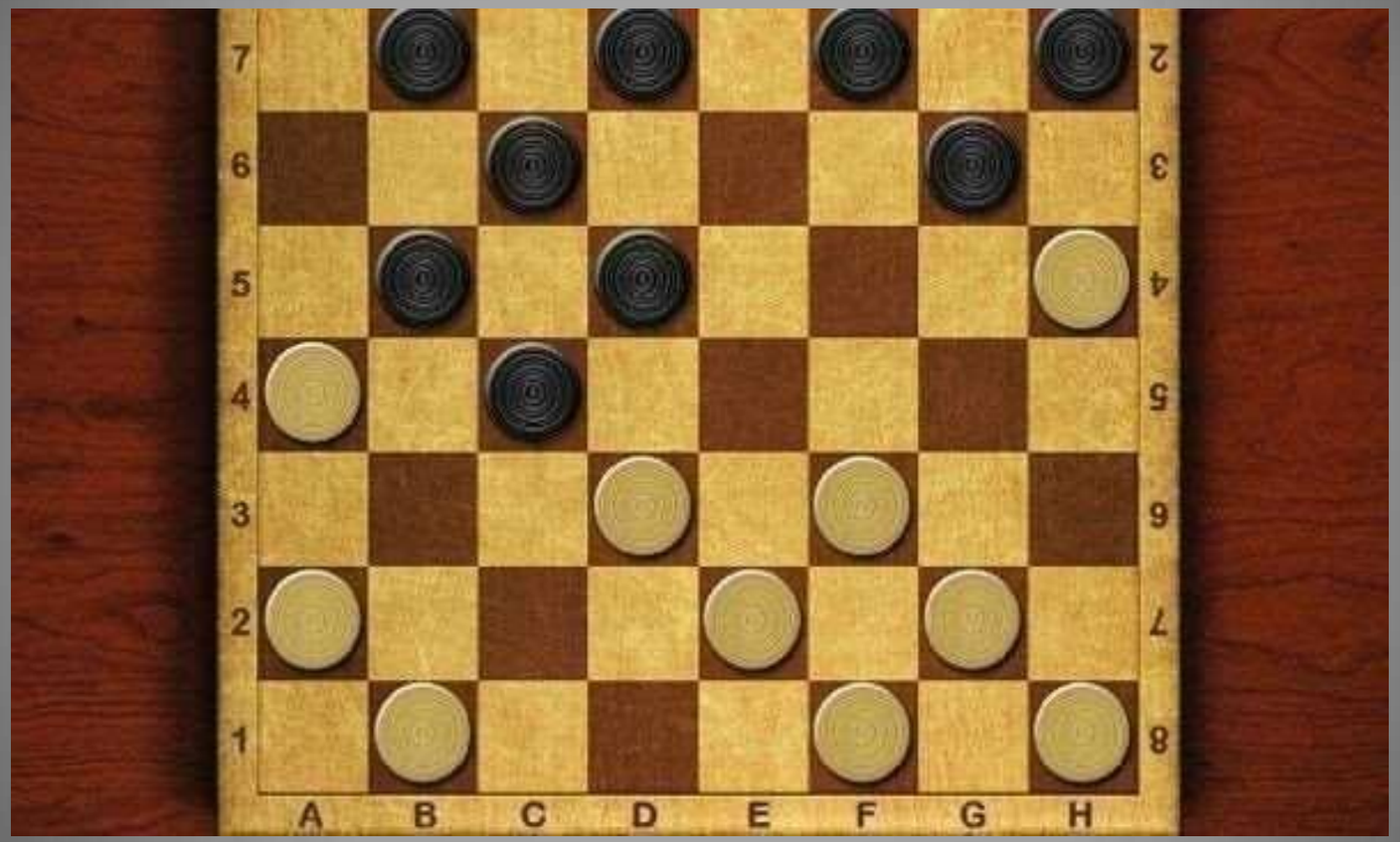
Well-PosedLearningProblem

Definition: A computer program is said to learn from experience E with respect to someclassoftasksT andperformancemeasureP,ifitsperformance attasksinT,as measured by P, improves with experience E.

Tohaveawell-definedlearningproblem,threefeaturesneedstobeidentified:

1. Theclassoftasks
2. Themeasureofperformancetobeimproved
3. Thesourceofexperience

CheckersGame



Computer ScienceandEngineering,NRCM

GameBasics

- Checkers is played by two players. Each player begins the game with 12 coloreddiscs.(Onesetofpiecesisblackandtheotherred.)Eachplayerplaceshis or her pieces on the 12 dark squares closest to him or her. Black moves first. Players then alternate moves.
- Theboardconsistsof64squares,alternatingbetween32darkand32lightsquares.
- Itispositionedsothateachplayerhasalightsquareontherightsidecornerclosest to him or her.
- A player wins the game when the opponent cannot make a move. In most cases, thisisbecausealloftheopponent'spieceshavebeencaptured,butitcouldalsobe because all of his pieces are blocked in.

RulesoftheGame

- Moves are allowedonlyon the dark squares, so pieces always move diagonally. Singlepieces arealways limitedtoforward moves (towardtheopponent).
- Apiecemakinganon-capturingmove(notinvolvingajump)maymoveonlyone square.
- Apiecemakingacapturingmove(ajump)leapsoveroneoftheopponent'spieces, landinginastraightdiagonallineontheotherside.Onlyonepiecemaybecaptured in a singlejump;however,multiplejumpsareallowedduringasingleturn.
- Whenapieceiscaptured,itisremovedfromtheboard.
- Ifaplayerisabletomakeacapture,thereisnooption;thejumpmustbemade.
- Ifmorethanonecaptureisavailable,theplayerisfreetochoosewhicheverheorshe prefers.

RulesoftheGameCont.

- Whenapiecereachesthefurthestrowfromtheplayerwhocontrolsthatpiece,itis crownedandbecomesaking.Oneofthepieceswhichhadbeencapturedisplacedontop of theking so that itis twiceas high as asingle piece.
- Kingsarelimitedtomovingdiagonallybutmaymovebothforwardandbackward. (Rememberthatsinglepieces,i.e.non-kings,arealwayslimitedtoforwardmoves.)
- Kings may combine jumps in several directions, forward and backward, on the same turn. Single pieces may shift direction diagonally during a multiple captureturn,butmustalways jumpforward (towardtheopponent).

Well-DefinedLearningProblem

Acheckerslearningproblem:

- TaskT:playingcheckers
- PerformancemeasureP:percentofgameswonagainstopponents
- TrainingexperienceE:playingpracticegamesagainstitself

Ahandwritingrecognitionlearningproblem:

- TaskT:recognizingandclassifyinghandwrittenwordswithinimages
- PerformancemeasureP:percentofwordscorrectlyclassified
- TrainingexperienceE:adatabaseofhandwrittenwordswith given classifications

Arobotdrivinglearningproblem:

- TaskT:drivingonpublicfour-lanehighwaysusingvisionsensors
- PerformancemeasureP:averagedistancetravelledbeforeanerror(asjudgedby humanoverseer)
- TrainingexperienceE:asequenceofimagesandsteeringcommandsrecorded Whileobservingahumandriver

DesigningaLearningSystem

1. ChoosingtheTrainingExperience
2. ChoosingtheTargetFunction
3. ChoosingaRepresentationfortheTargetFunction
4. ChoosingaFunctionApproximationAlgorithm
    1. Estimatingtrainingvalues
    2. Adjustingtheweights
5. TheFinalDesign

1.ChoosingtheTrainingExperience

- Thefirstdesignchoiceistochoosethetypeoftrainingexperiencefromwhichthe systemwilllearn.
- Thetypeoftrainingexperienceavailablecanhaveasignificantimpacton successorfailureofthelearner.

<u>Therearethreeattributeswhichimpactonsuccessorfailureofthelearner</u>

1. Whetherthetrainingexperienceprovidesdirectorindirectfeedbackregardingthe choices made by the performance system.
2. Thedegreetowhichthelearnercontrolsthesequenceoftrainingexamples
3. Howwellitrepresentsthedistributionofexamplesoverwhichthefinal systemperformancemustbemeasured.

1. Whetherthetrainingexperienceprovidesdirectorindirectfeedbackregarding the choices madeby the performance system ■

Forexample,incheckersgame:

- Inlearningtoplaycheckers,thesystemmightlearnfromdirecttrainingexamplesconsistingofindividual ***Checkersboardstatesandthecorrectmoveforeach***.

- Indirecttrainingexamplesconsistingofthe***movesequencesandfinaloutcomesofvariousgamesplayed***.

- Theinformationaboutthecorrectnessofspecificmovesearlyinthegamemustbeinferredindirectlyfromthe fact thatthegamewas eventuallywonorlost.

- Herethelearnerfacesanadditionalproblemof***creditassignment***,ordeterminingthedegreetowhicheachmovein thesequencedeservescreditorblameforthefinaloutcome.

- Creditassignmentcanbeaparticularlydifficultproblembecausethegamecanbelostevenwhenearlymoves areoptimal,ifthesearefollowedlaterbypoormoves.

- Hence,learningfromdirecttrainingfeedbackistypicallyeasierthanlearningfromindirectfeedback.

2. Asecond important attribute of the training experience *is the degree to which the learner controls the sequence of training examples*

Forexample,incheckersgame:

- Thelearnermightdependsontheteachertoselectinformativeboardstatesandtoprovidethecorrectmove foreach.
- Alternatively,thelearnermightitselfproposeboardstatesthatitfindsparticularlyconfusingandasktheteacherfor thecorrectmove.
- Thelearnermayhavecompletecontroloverboththeboardstatesand(indirect)trainingclassifications,asit doeswhenitlearnsbyplayingagainstitselfwithnoteacherpresent.
- Noticeinthislastcasethelearnermaychoosebetweenexperimentingwithnovelboardstatesthatithasnotyet considered,orhoningitsskillbyplayingminorvariationsoflinesofplayitcurrentlyfindsmostpromising.

3. A third attribute of the training experience is how well it represents *the distributionofexamples*overwhichthefinalsystemperformancemustbe measured ■

Learningismostreliablewhenthetrainingexamplesfollowadistributionsimilartothatoffuturetest examples.

Forexample,incheckersgame:

- Incheckerslearningscenario,theperformancemetricPisthepercentofgamesthesystemwinsinthe worldtournament.

- If its training experience E consists only of games played against itself, there is an danger that this training experiencemight not be fully representative of the distribution ofsituations overwhich it will later be tested. Forexample,thelearnermightneverencountercertaincrucialboardstatesthatareverylikelytobeplayedby the human checkers champion.

- It is necessary to learn from a distribution of examples that is somewhat different from those on which the finalsystemwillbeevaluated.Suchsituationsareproblematicbecausemasteryofonedistributionofexamples will notnecessarylead to strongperformanceover some otherdistribution.

3.ChoosingtheTargetFunction

The next design choice is to determine exactly what type of knowledge will be learned and how thiswillbe used bytheperformance program.

- Let’sbeginwithacheckers-playingprogramthatcangeneratethelegalmoves fromanyboardstate.
- The program needs only to learn how to choose the best move from among these legalmoves.Thislearningtaskisrepresentativeofalargeclassoftasksforwhich the legal moves that define some large search space are known a priori, but for which the best search strategy is not known.

Given thissettingwhere wemust learn to choose among thelegal moves, themost obviouschoiceforthetypeofinformationtobelearnedisaprogram,orfunction,that chooses the best move for any given board state.

1. Let***ChooseMove***bethetargetfunctionandthenotationis

ChooseMove:B $\longrightarrow$ ***M***

Whichindicatethatthisfunctionacceptsasinputanyboardfromthesetoflegal boardstatesBandproducesasoutputsomemovefromthesetoflegalmovesM.

ChooseMove is an choice for the target function in checkers example, but this function will turn out to be very difficult to learn given the kind of indirect training experience available to our system

2. Analternativetargetfunctionis an***evaluationfunction***thatassignsa ***numericalscore*** to any given boardstate

LetthetargetfunctionVandthenotation

$$V:B \longrightarrow R$$

WhichdenotethatVmapsanylegalboardstatefromthesetBtosomerealvalue

We intend for this target function V to assign higher scores to better board states. If thesystemcansuccessfullylearnsuchatargetfunctionV,itcaneasilyuseittoselect the best move from any current board position.

LetusdefinethetargetvalueV(b)foranarbitraryboardstatebinB,asfollows:

1. Ifbisafinalboardstatethatiswon,thenV(b)=100
2. Ifbisafinalboardstatethatislost,thenV(b)=-100
3. Ifbisafinalboardstatethatisdrawn,thenV(b)=0
4. Ifbisanotafinalstateinthegame,thenV(b)=V(b'),

Whereb'isthebestfinalboardstatethatcanbeachievedstartingfromband playing optimallyuntil the end of the game

3. ChoosingaRepresentationforthe TargetFunction

Letuschooseasimplerepresentation-foranygivenboardstate,thefunctionwillbe calculatedas a linearcombinationof the followingboard features:

XL: the number of black pieces on the
boardx2:thenumberofredpiecesonthe
boardx3:thenumberofblackkingsonthe
boardx4:the number of red kings on the
board
X5:thenumberofblackpiecesthreatenedbyred(i.e.,whichcanbe
captured on red's next turn)
x6:thenumberofredpiecesthreatenedbyblack

$$\hat{V}(b) = w_0 + w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4 + w_5x_5 + w_6x_6$$

Where,

- w_0 through w_6 are numerical coefficients, or weights, to be chosen by the learning algorithm.

- Learned values for the weights w_1 through w_6 will determine the relative Importance of the various board features in determining the value of the board
  - The weight w_0 will provide an additive constant to the board value

Partialdesignofacheckerslearningprogram:

- TaskT:playingcheckers
- PerformancemeasureP:percentofgameswonintheworldtournament
- TrainingexperienceE:gamesplayedagainstitself
- Targetfunction:V:Board $\longrightarrow$ R
- Targetfunctionrepresentation

$$\hat{V}(b) = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_3 + w_4 x_4 + w_5 x_5 + w_6 x_6$$

Thefirstthreeitemsabovecorrespondtothespecificationofthelearningtask, whereasthefinaltwoitemsconstitutedesignchoicesfortheimplementationof the learning program.

4. ChoosingaFunctionApproximation Algorithm

- Inordertolearnthetargetfunctionwerequireasetoftrainingexamples,each describingaspecificboardstatebandthetrainingvalueV_{train}(b)forb.

- Eachtrainingexampleisanorderedpairoftheform(b,V_{train}(b)).

- For instance,thefollowingtrainingexampledescribes aboard statebin whichblackhaswonthegame(notex_2=0indicatesthatredhasnoremaining pieces)andforwhichthetargetfunctionvalueV_{train}(b)istherefore+100.

$$((x_1=3,x_2=0,x_3=1,x_4=0,x_5=0,x_6=0),+100)$$

FunctionApproximationProcedure

1. Derivetrainingexamplesfromtheindirecttrainingexperienceavailabletothe learner
2. Adjuststheweights$w_i$tobestfitthesetrainingexamples

Where,
$\hat{v}$isthe learner'scurrentapproximation to V
Successor(b)denotesthenextboardstatefollowingforwhichitisagaintheprogram's turn to move

Ruleforestimatingtrainingvalues

$$V_{train}(b) \leftarrow \hat{v}(Successor(b))$$

Specifythelearningalgorithmforchoosingthe weights w_i tobestfitthesetof training examples $\{(b, V_{train}(b))\}$

Afirststepistodefinewhatwemeanbythebestfittothetrainingdata.

- Onecommonapproach istodefinethebesthypothesis,orsetofweights,as thatwhich minimizes the squared error E between the training values and the values predicted by the hypothesis.

$$E \equiv \sum_{\langle b, V_{train}(b)\rangle \in \text{ training examples}} (V_{train}(b) - \hat{V}(b))^2$$

- Severalalgorithmsareknownforfindingweightsofalinearfunctionthat minimizeE.

Onesuchalgorithmiscalledthe***leastmeansquares***,or***LMStrainingrule***.Foreach observed trainingexampleitadjusts theweights asmallamountinthedirectionthat reduces the error on this training example

LMSweightupdaterule:-Foreachtrainingexample

(b,V_{train}(b))Usethecurrentweightstocalculate$\hat{v}$(b)

Foreachweightw_i,updateitas

$$w_i \leftarrow w_i + \eta(V_{train}(b) - v(b))x_i$$

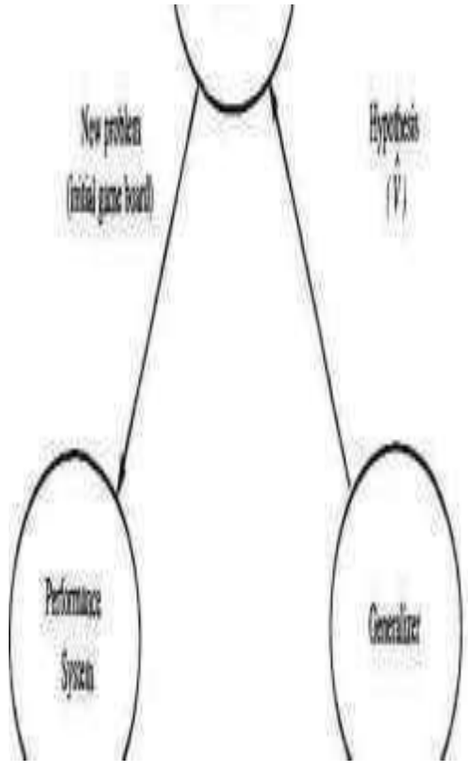
Hereisasmallconstant(e.g.,0.1)thatmoderatesthesizeoftheweight

update.Workingofweightupdaterule

- Whentheerror(Vtrain(b)-V(b))iszero,noweightsarechanged.
- When (Vtrain(b)-V(b))ispositive(i.e.,whenV(b)istoolow),theneachweight is increasedinproportiontothevalueofitscorrespondingfeature.Thiswill raise the valueofV(b),reducingtheerror.
- IfthevalueofsomefeaturexiszerO,thenitsweightisnotalteredregardlessofthe error,sothattheonlyweightsupdatedarethosewhosefeaturesactuallyoccur on the training example board.

Thefinaldesignofcheckerslearningsystemcanbedescribedbyfourdistinctprogram modules thatrepresent the central components in manylearning systems

5. TheFinalDesign

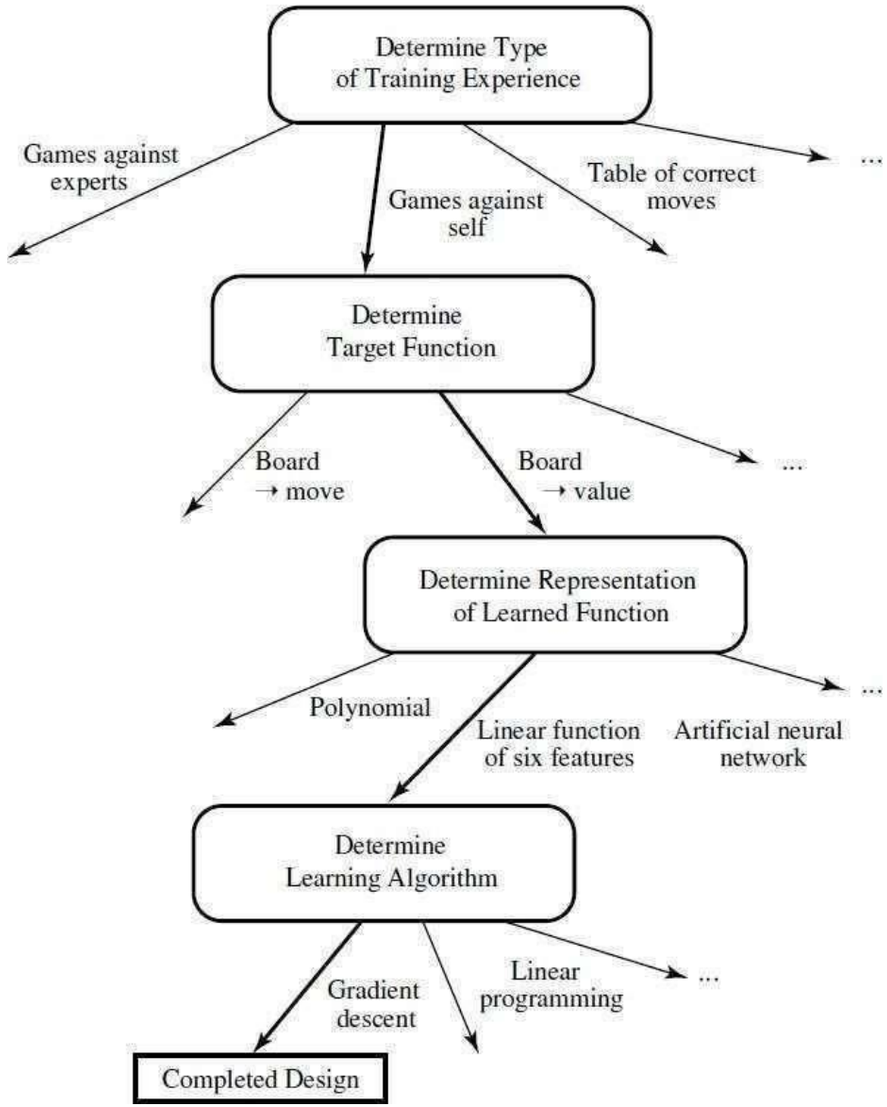

New problem
(initial game board)
Hypothesis
(V)
Performance
System
Generalizer

1. ***The Performance System*** is the module that must solve the given performance taskby using the learned target function(s).
Ittakesaninstanceofanewproblem(newgame)asinputandproducesatraceofits solution (game history) as output.
In checkers game, the strategy used by the Performance System to select its next move at each step is determined by the learned $\hat{v}$evaluation function. Therefore, weexpect its performance to improve as this evaluation function becomes increasingly accurate.

2. ***TheCritic***takesasinputthehistoryortraceof thegame andproduces asoutputa setoftrainingexamplesofthetargetfunction.Asshowninthediagram,eachtraining exampleinthiscasecorrespondstosomegamestateinthetrace,alongwithan estimateV_{train} of the target function value for this example.

3. TheGeneralizertakesasinputthetrainingexamplesandproducesan outputhypothes is that isitsestimateof thetarget function. Itgeneralizesfromthespecifictrainingexamples,hypothesizingageneral functionthatcoverstheseexamplesandothercasesbeyondthetrainingexamples.Inour example,theGeneralizercorresponds totheLMS algorithm,andtheoutput hypothesis isthefunction$\hat{v}$describedbythe learnedweightsw_0,...,W6.

4. TheExperimentGeneratortakesasinputthecurrenthypothesisandoutputsanew problem(i.e.,initialboardstate)forthePerformanceSystemtoexplore.Itsroleisto picknewpracticeproblemsthatwillmaximizethelearningrateoftheoverall system. Inourexample,theExperimentGeneratoralwaysproposesthesameinitialgame board tobeginanewgame.


Determine Type of Training Experience
Games against experts
Games against self
Table of correct moves
...
Determine Target Function
Board → move
Board → value
...
Determine Representation of Learned Function
Polynomial
Linear function of six features
Artificial neural network
...
Determine Learning Algorithm
Gradient descent
Linear programming
...
Completed Design

PerspectivesofMachineLearning

Perspectiveofmachinelearninginvolvessearchingverylarge space of possiblehypothesis todetermineone that

Bestfitstheobserveddataandanypriorknowledgeheldby learner.

Issuesin MachineLearning

- What algorithms exist for learning general target functions from specific training examples? In what settings will particular algorithms converge to the desired function, given sufficient training data?Which algorithms perform best for which types of problems and representations?

- Howmuchtrainingdataissufficient?Whatgeneralboundscanbefound torelate theconfidenceinlearnedhypothesestotheamountoftrainingexperienceandthe character of the learner's hypothesis space?

- Whenandhowcanpriorknowledgeheldbythelearnerguidetheprocessofgeneralizingf romexamples?Canpriorknowledgebehelpfulevenwhenitisonly approximatelycorrect?

- What is thebest strategyforchoosing a useful next training experience, and how does the choiceof this strategy alter the complexityof the learningproblem?

- What is the best way to reduce the learning task to one or more function approximation problems? Put another way, what specific functions should the system attempttolearn? Canthis process itselfbe automated?

- Howcanthelearnerautomaticallyalteritsrepresentationtoimproveitsabilityto representandlearnthetargetfunction?

ConceptLearning

- Learning involves acquiring general concepts from specific training examples. Example: People continually learn general concepts or categories such as "bird,""car,""situationsinwhichIshouldstudymoreinordertopasstheexam,"etc.
- Each such concept can be viewed as describing some subset of objects or events defined over a larger set
- Alternatively, each concept can be though to fasaBoolean-valued function defined over this larger set. (Example: A function defined over all animals, whose value is true for birds and false for other animals).

Conceptlearning-InferringaBoolean-valuedfunctionfromtrainingexamplesofits input andoutput

AConceptLearningTask

Consider theexample task of learning the target concept "DaysonwhichmyfriendAldoenjoyshisfavoritewatersport."

Example	Sky	AirTemp	Humidity	Wind	Water	Forecast	EnjoySport
1	Sunny	Warm	Normal	Strong	Warm	Same	Yes
2	Sunny	Warm	High	Strong	Warm	Same	Yes
3	Rainy	Cold	High	Strong	Warm	Change	No
4	Sunny	Warm	High	Strong	Cool	Change	Yes

Table-Describesasetof example days, each represented byasetofattributes

Whathypothesisrepresentationisprovidedtothelearner?

Let’sconsiderasimplerepresentationinwhicheachhypothesisconsistsofaconjunctionof constraints on the instance attributes.

Leteachhypothesisbeavectorofsixconstraints,specifyingthevaluesofthesixattributes Sky, AirTemp, Humidity,Wind,Water,and Forecast.

Foreachattribute,thehypothesiswilleither

- Indicatebya"?'thatanyvalueisacceptableforthisattribute,
- Specifyasinglerequiredvalue(e.g.,Warm)fortheattribute,or
- Indicatebya"Φ"thatnovalueisacceptable

The hypothesis that PERSON enjoys his favo rite sport only on cold days with highhumidity (independent of the values of the other attributes) is represented by theexpression

(?,Cold,High,?,?,?)

Themostgeneralhypothesis-thateverydayisapositiveexample-isrepresentedby

(?,?,?,?,?,?)

The most specific possible hypothesis-that day is a positive example-is norepresentedby

Notation

Thesetofitemsoverwhichtheconceptisdefinediscalledthesetof ***instances***,whichwedenotebyX.

Example:Xisthesetofallpossibledays,eachrepresentedbytheattributes:Sky, AirTemp, Humidity,Wind,Water,and Forecast

Theconceptorfunctiontobelearnediscalledthe***targetconcept***,whichwedenotebyc. ccan be anyBooleanvalued function defined overtheinstancesXc:X{O,1}

Example:Thetargetconceptcorrespondstothevalueoftheattribute***EnjoySport*** (i.e.,c(x)=1if***EnjoySport***=Yes,andc(x)=0if***EnjoySport***=No).

- Instancesforwhichc(x)=1arecalledpositiveexamples,ormembersofthetarget concept.
- Instancesforwhichc(x)=0arecallednegativeexamples,ornon-membersofthe target concept.
- Theorderedpair(x,c(x))todescribethetrainingexampleconsistingofthe instancex and its target conceptvalue c(x).
- ***D***todenotethesetofavailabletrainingexamples
- Thesymbol ***H*** todenotetheset ofall possiblehypothesesthat thelearner mayconsiderregardingtheidentityofthetargetconcept.Eachhypothesis ***h*** in ***H***represents a Boolean-valued functiondefined over X

$$h:X \longrightarrow \{O,1\}$$

- Thegoalofthelearneristofindahypothesissuchthath(x)=c(x)forallxin ***X***.

Example	Sky	AirTemp	Humidity	Wind	Water	Forecast	EnjoySport
1	Sunny	Warm	Normal	Strong	Warm	Same	Yes
2	Sunny	Warm	High	Strong	Warm	Same	Yes
3	Rainy	Cold	High	Strong	Warm	Change	No
4	Sunny	Warm	High	Strong	Cool	Change	Yes

- **Given:**
  - Instances X: Possible days, each described by the attributes
    - *Sky* (with possible values *Sunny*, *Cloudy*, and *Rainy*),
    - *AirTemp* (with values *Warm* and *Cold*),
    - *Humidity* (with values *Normal* and *High*),
    - *Wind* (with values *Strong* and *Weak*),
    - *Water* (with values *Warm* and *Cool*), and
    - *Forecast* (with values *Same* and *Change*).
  - Hypotheses H: Each hypothesis is described by a conjunction of constraints on the attributes *Sky*, *AirTemp*, *Humidity*, *Wind*, *Water*, and *Forecast*. The constraints may be "?" (any value is acceptable), "$\emptyset$" (no value is acceptable), or a specific value.
  - Target concept c: $EnjoySport : X \rightarrow \{0, 1\}$
  - Training examples D: Positive and negative examples of the target function (see Table 2.1).
- **Determine:**
  - A hypothesis h in H such that $h(x) = c(x)$ for all x in X.

TABLE The *EnjoySport* concept learning task.

Any hypothesis foundto approximate the target function well over a sufficiently large set of training examples will also approximate the target function well overother unobserved examples.

TheInductiveLearningHypothesis

ConceptlearningasSearch

- Conceptlearningcanbeviewedasthetaskofsearchingthroughalargespaceof hypotheses implicitlydefined bythe hypothesis representation.
- Thegoalofthissearchistofindthehypothesisthatbestfitsthetraining examples.

Example,theinstances***X***andhypotheses***H***inthe***EnjoySport***learningtask.The attribute***Sky***hasthreepossiblevalues,and***AirTemp,Humidity ,Wind,WaterForecast***eachhavetwopossiblevalues,theinstancespaceXcontains exactly

- 3.2.2.2.2.2=96Distinctinstances
- 5.4.4.4.4.4=5120syntacticallydistincthypotheseswithinH.

Everyhypothesiscontainingoneormore"Φ"symbolsrepresentstheemptysetof instances;thatis,itclassifieseveryinstanceas***negative.***

- 1+(4.3.3.3.3.3)=973. Semantically distinct hypotheses

General-to-SpecificOrderingofHypotheses

- Considerthetwohypotheses

  h_1=(Sunny,?,?,Strong,?,?)

  h_2=(Sunny,?,?,?,?,?)

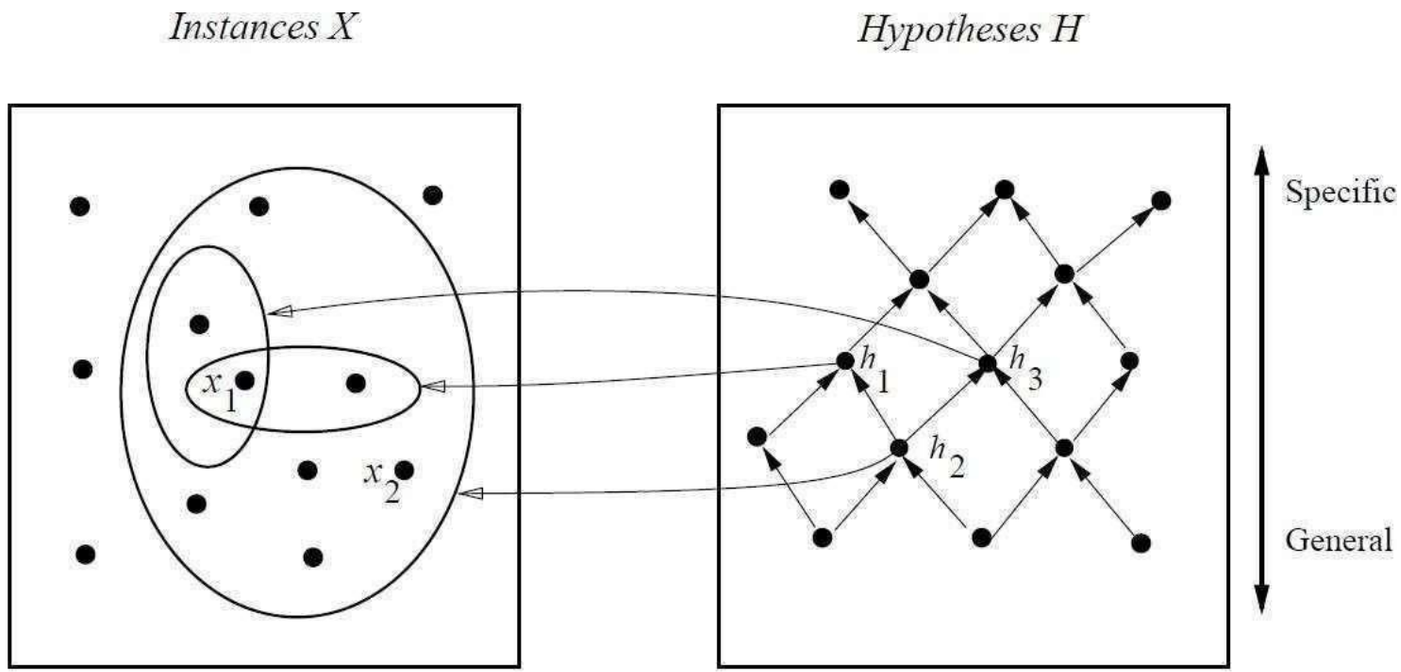
- Considerthesetsofinstancesthatareclassifiedpositivebyh$_1$andbyh_2.
- $h_2$imposesfewerconstraintsontheinstance, itclassifies more instancesaspositive. So,anyinstanceclassifiedpositivebyh$_1$ willalsobeclassifiedpositiveby h_2.Therefore,$h_2$ismoregeneralthanh$_1$.

General-to-SpecificOrderingofHypotheses

- Given hypotheses h_j and h_k, h_j is more-general-than or-equal do h_k if and only if any instance that satisfies h_k also satisfies h_i

Definition: Let h_j and h_k be Boolean-valued functions defined over X. Then h_j is more general-than-or-equal-to h_k (written $h_j \geq h_k$) if and only if

$$(\forall x \in X)[(h_k(x) = 1) \rightarrow (h_j(x) = 1)]$$



x_1 = <Sunny, Warm, High, Strong, Cool, Same>
x_2 = <Sunny, Warm, High, Light, Warm, Same>

h_1 = <Sunny, ?, ?, Strong, ?, ?>
h_2 = <Sunny, ?, ?, ?, ?, ?>
h_3 = <Sunny, ?, ?, ?, Cool, ?>

- Inthe figure, thebox on the leftrepresentsthesetXofallinstances, the box onthe rightthesetH ofall hypotheses.
- Eachhypothesiscorrespondstosomesubsetof X-thesubsetofinstancesthatitclassifies positive.
- The arrows connecting hypothesesrepresent the ***more - general - than***relation,withthearrowpointingtowardthelessgeneralhypothesis.
- Notethesubsetofinstancescharacterizedby$h_2$subsumesthesubset characterizedbyh_1,hence$h_2$ismore-general–thanh_1

FIND-S:FindingaMaximallySpecific Hypothesis

FIND-SAlgorithm

1. Initialize*h*tothemostspecifichypothesisin*H*
2. Foreachpositivetraininginstance*x*

   Foreachattributeconstraint$a_i$in*h*

   Iftheconstraint$a_i$issatisfiedby*x*

   Thendonothing

   Elsereplace$a_i$in*h*bythenextmoregeneralconstraintthatissatisfiedby*x*
3. Outputhypothesis*h*

Example	Sky	AirTemp	Humidity	Wind	Water	Forecast	EnjoySport
1	Sunny	Warm	Normal	Strong	Warm	Same	Yes
2	Sunny	Warm	High	Strong	Warm	Same	Yes
3	Rainy	Cold	High	Strong	Warm	Change	No
4	Sunny	Warm	High	Strong	Cool	Change	Yes

ThefirststepofFIND-Sistoinitialize***h***tothemostspecifichypothesisin***H***

h-(Ø,Ø,Ø,Ø,Ø,Ø)

x_1=<*SunnyWarmNormalStrongWarmSame*>,+

Observing the first training example, it is clear that our hypothesis is too specific. Inparticular, none of the "Ø" constraints in h are satisfied by this example, so each isreplacedbythe nextmoregeneral constraintthatfitsthe example

h_1=<SunnyWarmNormalStrongWarmSame>

Thishisstillveryspecific;itassertsthatallinstancesarenegativeexceptforthesinglepositive training example

x_2=<*Sunny,Warm,High,Strong,Warm,Same*>,+

The second training example forces the algorithm to further generalize h, this timesubstituting a "?' in place of any attribute value in h that is not satisfied by the newexample

h_2=<SunnyWarm?StrongWarmSame>

x3=<Rainy,Cold,High,Strong,Warm,Change>,-

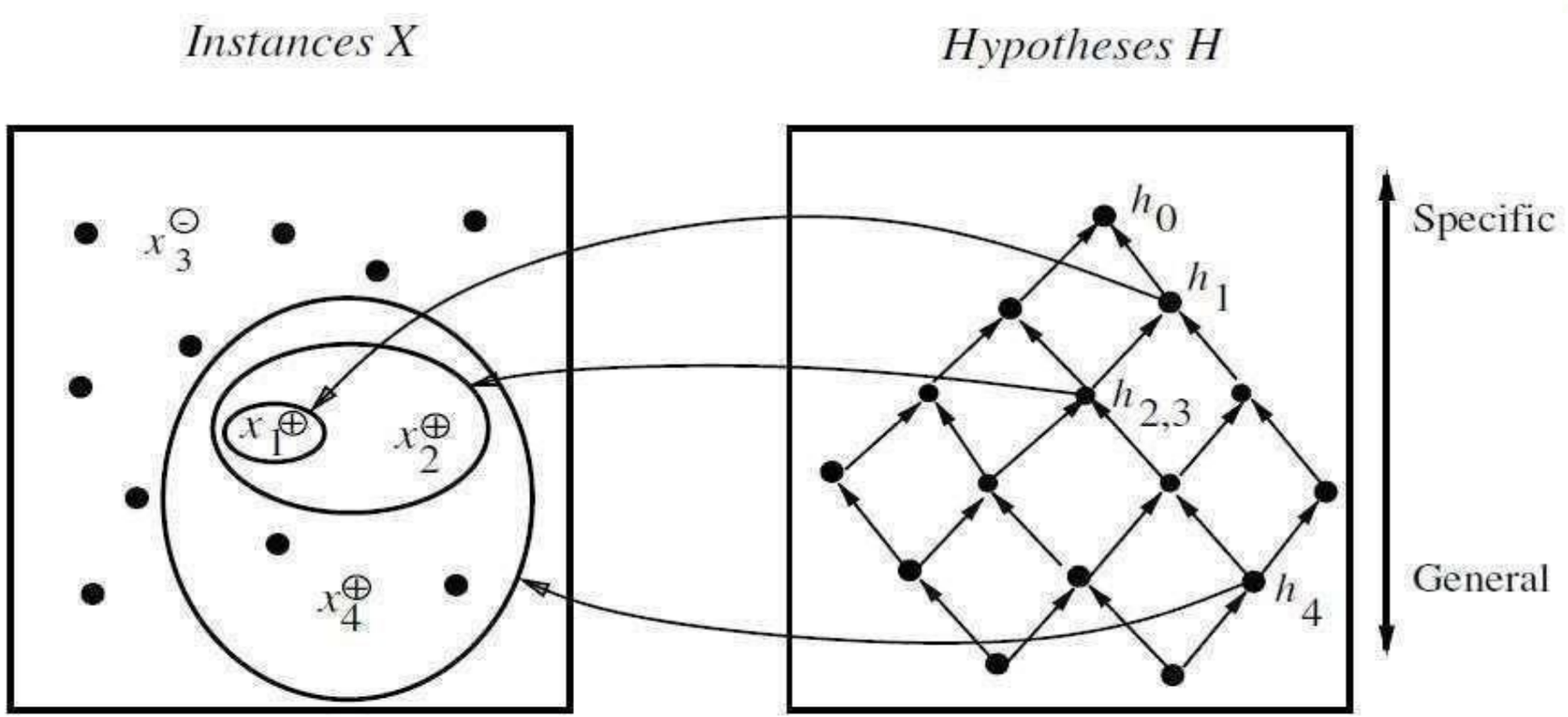
Uponencounteringthethirdtrainingthealgorithmmakesnochangetoh.TheFIND-Salgorithm simplyignores everynegative example.

h3=<SunnyWarm?StrongWarmSame>

x4=<SunnyWarmHighStrongCoolChange>,+

Thefourthexampleleadstoafurthergeneralizationofh

h4=<SunnyWarm?Strong??>



x_1 = <Sunny Warm Normal Strong Warm Same>, +

x_2 = <Sunny Warm High Strong Warm Same>, +

x_3 = <Rainy Cold High Strong Warm Change>, -

x_4 = <Sunny Warm High Strong Cool Change>, +

h_0 = <∅, ∅, ∅, ∅, ∅, ∅>

h_1 = <Sunny Warm Normal Strong Warm San

h_2 = <Sunny Warm ? Strong Warm Same>

h_3 = <Sunny Warm ? Strong Warm Same>

h_4 = <Sunny Warm ? Strong ? ? >

ThekeypropertyoftheFIND-Salgorithmis

- FIND-Sisguaranteedtooutputthemostspecifichypothesiswithin Hthat isconsistentwiththepositivetrainingexamples
- FIND-S algorithm's final hypothesis will also be consistent with the negativeexamples provided the correct target concept is contained in H, and provided thetrainingexamples are correct.

UnansweredbyFIND-S

1. Hasthelearnerconvergedtothecorrecttargetconcept?
2. Whypreferthemostspecifichypothesis?
3. Arethetrainingexamplesconsistent?
4. Whatiftherearesevеralmaximallyspecificconsistenthypotheses?

VersionSpaceandCANDIDATEELIMINATIONAlgorithm

ThekeyideaintheCANDIDATE-ELIMINATIONalgorithmistooutputadescriptionofthesetofall***hypothesesconsistentwiththetrainingexamples***

Representation

- ***Definition:***Ahypothesishis**consistent**with**a**setoftrainingexamples***D***if**and**onlyif***h(x)*** =***c(x)***foreachexample(x,***c(x))***in***D.***

$$Consistent(h,D) \equiv (\forall \langle x, c(x) \rangle \in D) h(x) = c(x))$$

Notedifferencebetweendefinitionsof*consistent*and*satisfies*

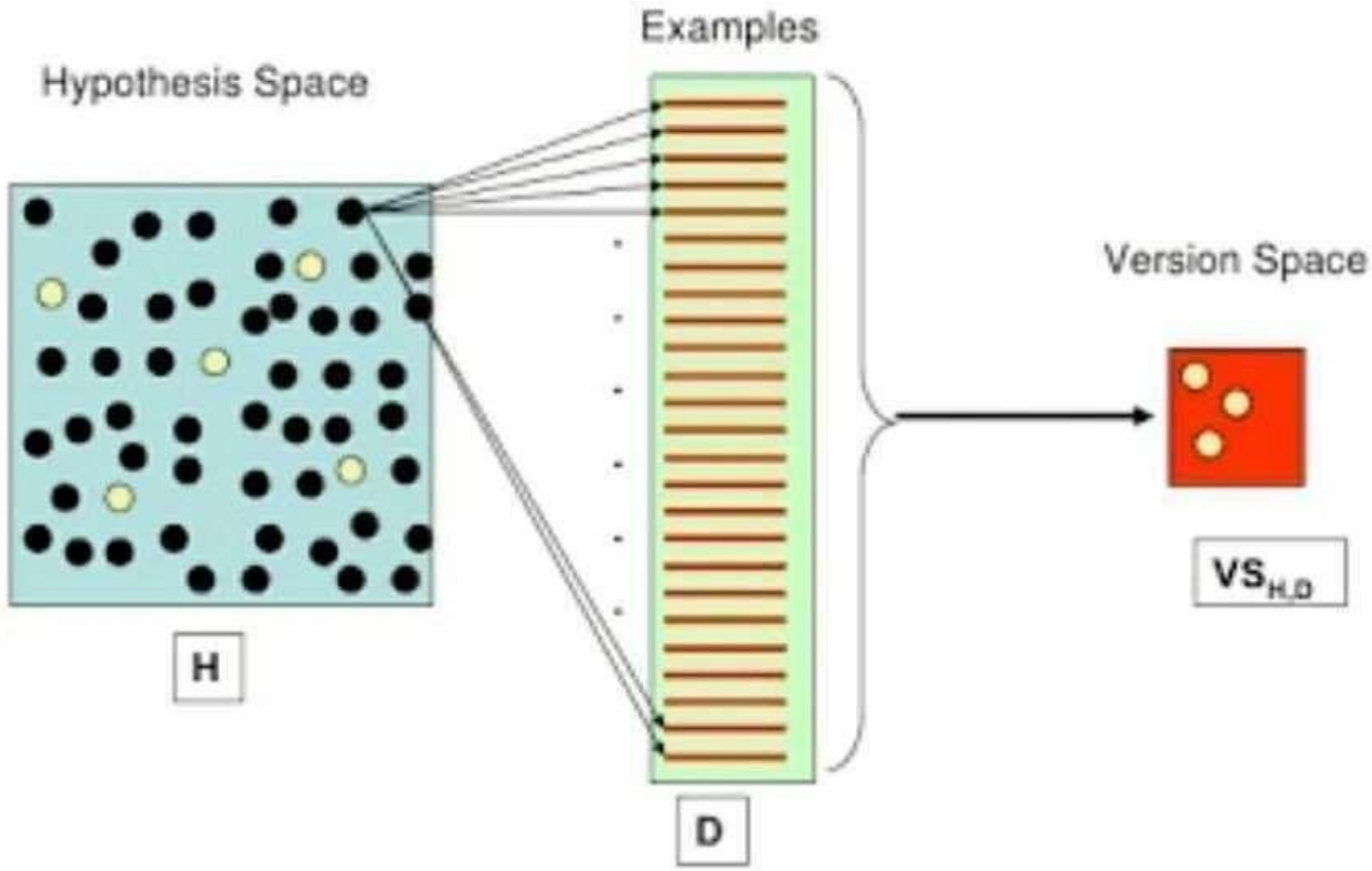
- Anexample*x*issaidto***satisfy***hypothesis***h***when***h(x)***=1,regardless ofwhether*x*isapositiveornegativeexampleofthetargetconcept.
- anexample*x*issaidto***consistent***withhypothesis***h***iff***h(x)=c(x)***

VersionSpace

ArepresentationofthesetofallhypotheseswhichareconsistentwithD

Definition: The **version space,** denoted $VS_{H,D}$with respect to hypothesisspace ***H***and training examples D, is the subset of hypotheses from ***H*** consistent with thetraining examplesin D

$$VS_{H,D} \equiv \{h \in H | Consistent(h,D)\}$$


Examples
Hypothesis Space
Version Space
H
D
VS H,D

TheLIST-THEN-ELIMINATE Algorithm

TheLIST-THEN-ELIMINATEalgorithmfirstinitializestheversionspacetocontainall hypotheses inH andthen eliminatesanyhypothesis foundinconsistentwith anytraining example.

TheLIST-THEN-ELIMINATE Algorithm

1. VersionSpace*a*listcontainingeveryhypothesisinH

2. Foreachtrainingexample,***(x,c(x))***

removefrom***VersionSpace***anyhypothesis***h***forwhich***h(x)≠c(x)***

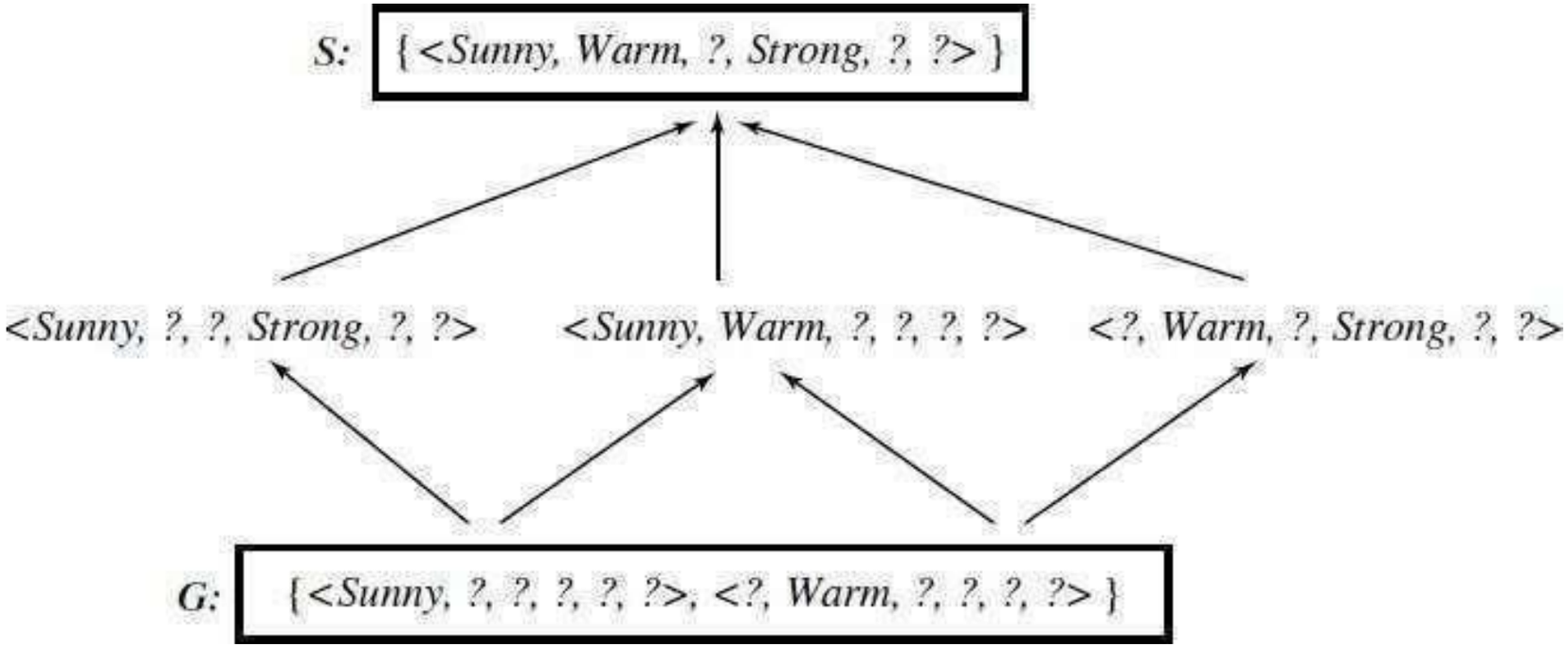
3. Outputthelistofhypothesesin***VersionSpace***

TheLIST-THEN-ELIMINATEAlgorithm

- **List-Then-Eliminate**worksinprinciple,solongasversionspaceisfinite.
- However,sinceitrequiresexhaustiveenumerationofallhypothesesinpracticeitisnot feasible.

AMoreCompactRepresentationforVersionSpaces

- Theversionspaceisrepresentedbyitsmostgeneralandleastgeneralmembers.
- Thesemembersformgeneralandspecificboundarysetsthatdelimittheversionspace withinthepartiallyordered hypothesis space.



Example	Sky	AirTemp	Humidity	Wind	Water	Forecast	EnjoySport
1	Sunny	Warm	Normal	Strong	Warm	Same	Yes
2	Sunny	Warm	High	Strong	Warm	Same	Yes
3	Rainy	Cold	High	Strong	Warm	Change	No
4	Sunny	Warm	High	Strong	Cool	Change	Yes

- Aversionspacewithitsgeneral and specific boundarysets.
- The version space includes allsix hypotheses shown here, butcanberepresentedmoresimply by ***S***and***G.***
- Arrowsindicateinstanceof the*more-general-than* relation.Thisistheversion spaceforthe *Enjoysport*conceptlearning
- problemandtrainingexamples describedinbelowtable

$$G \equiv \{g \in H | Consistent(g, D) \wedge (\neg \exists g' \in H)[(g' >_g g) \wedge Consistent(g', D)]\}$$

Definition: The **specific boundary** S, with respect to hypothesis space *H* and training data *D*, is the set of minimally general (i.e., maximally specific) members of *H* consistent with *D*.

$$S \equiv \{s \in H | Consistent(s, D) \wedge (\neg \exists s' \in H)[(s >_g s') \wedge Consistent(s', D)]\}$$

VersionSpacerepresentationtheorem

Theorem: Let X be an arbitrary set of instances and Let H be a set ofBoolean-valued hypotheses defined over X. Let c : X →{O, 1} be an arbitrary target conceptdefined over X, and let D be an arbitrary set of training examples {(x, c(x))). For allX, H,c, and D such that Sand G are well defined,

$$VS_{H,D}=\{h\in H|(\exists s\in S)(\exists g\in G)(g\geq_g h\geq_g s)\}$$

$$VS_{H,D}=\{h\in H\mid(\exists s\in S)(\exists g\in G)(g\geq_g h \geq_g s)\}$$

ToProve:

1. Everyhsatisfyingtherighthandsideoftheaboveexpressionisin***VS*H,D**
2. Everymemberof***VS*H,D**satisfiestheright-handsideoftheexpression

Sketchofproof:

1. letg,h,sbearbitrarymembersofG,H,Srespectivelywith$g\geq_g h\geq_g s$

Bythe definitionof***S,s***mustbesatisfiedbyallpositiveexamplesinD.Because$h\geq_g s$,hmustalsobe satisfiedbyallpositiveexamplesinD.

Bythedefinitionof***G,***gcannotbesatisfiedbyanynegativeexampleinD,andbecause$g\geq_g h$hcannot be satisfied by any negative example in D. Because h is satisfied by all positive examples in Dandbynonegativeexamples inD,hisconsistentwithD,andthereforehisamemberof$VS_{H,D}$

2. Itcanbeprovenbyassumingsomehin$VS_{H,D}$,thatdoesnotsatisfytheright-handsideoftheexpression,thenshowingthatthisleadstoaninconsistency

TheCANDIDATE-ELIMINATIONLearningAlgorithm

The CANDIDATE-ELIMINTION algorithm computes the ***version space*** containingall hypotheses from H that are consistent with an observed sequence of trainingexamples.

Initialize G to the set of maximally general hypotheses in HInitialize S to the set of maximally specific hypotheses in HForeachtrainingexample d, do

- Ifdisapositiveexample
  - RemovefromGanyhypothesisinconsistentwithd
  - ForeachhypothesissinSthatisnotconsistentwithd
    - RemovesfromS
    - AddtoSallminimalgeneralizationshofssuchthat
      - hisconsistentwithd,andsomememberofGismoregeneralthanh
    - RemovefromSanyhypothesisthatismoregeneralthananotherhypothesisinS

- Ifdisanegativeexample
  - RemovefromSanyhypothesisinconsistentwithd
  - ForeachhypothesisginGthatisnotconsistentwithd
    - RemovegfromG
    - AddtoGallminimalspecializationshofgsuchthat
      - hisconsistentwithd,andsomememberofSismorespecificthanh
    - RemovefromGanyhypothesisthatislessgeneralthananotherhypothesisinG

AnIllustrativeExample

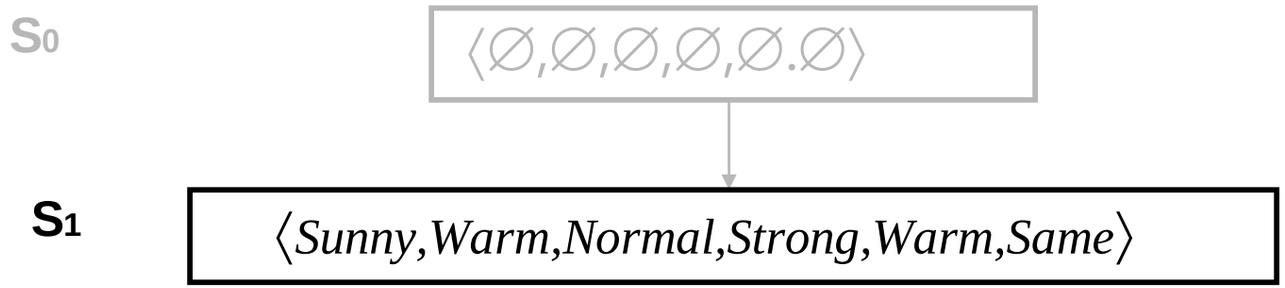
TheboundarysetsarefirstinitializedtoG_0andS_0,themostgeneralandmostspecifichypothesesinH.

S_0 ⟨∅,∅,∅,∅,∅,∅⟩

G_0 ⟨?,?,?,?,?,?⟩

Fortrainingexampled,

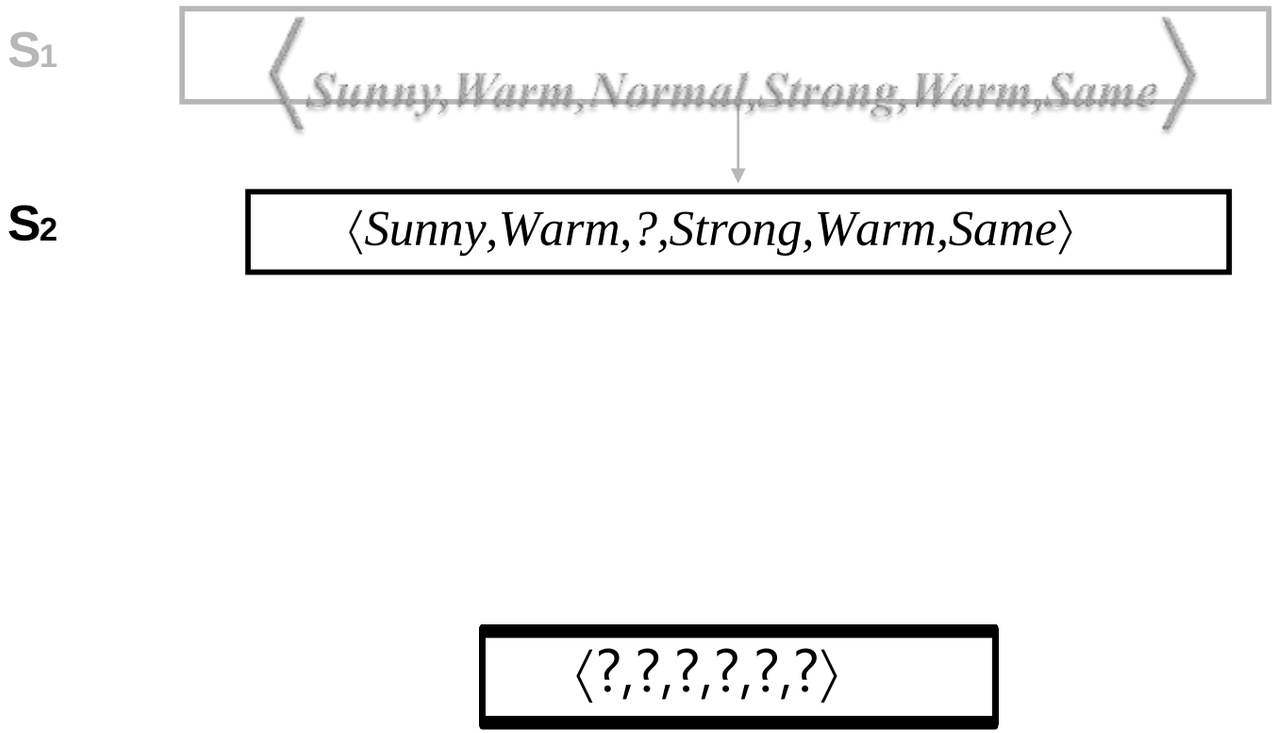
⟨Sunny,Warm,Normal,Strong,Warm,Same⟩+



⟨?,?,?,?,?,?⟩

Fortrainingexampled,

⟨Sunny,Warm,High,Strong,Warm,Same⟩+



Fortrainingexampled,

⟨Rainy,Cold,High,Strong,Warm,Change⟩–

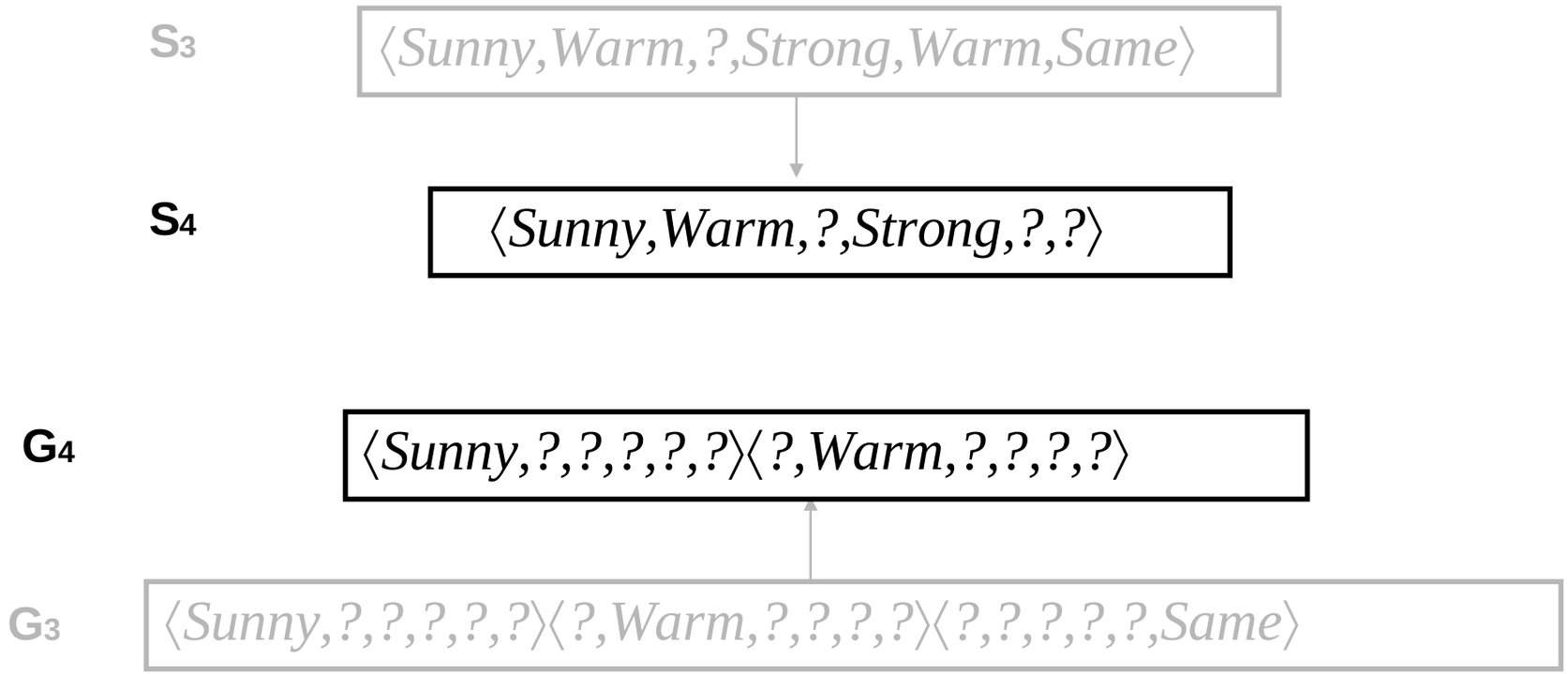
S_2, S_3 ⟨Sunny,Warm,?,Strong,Warm,Same⟩

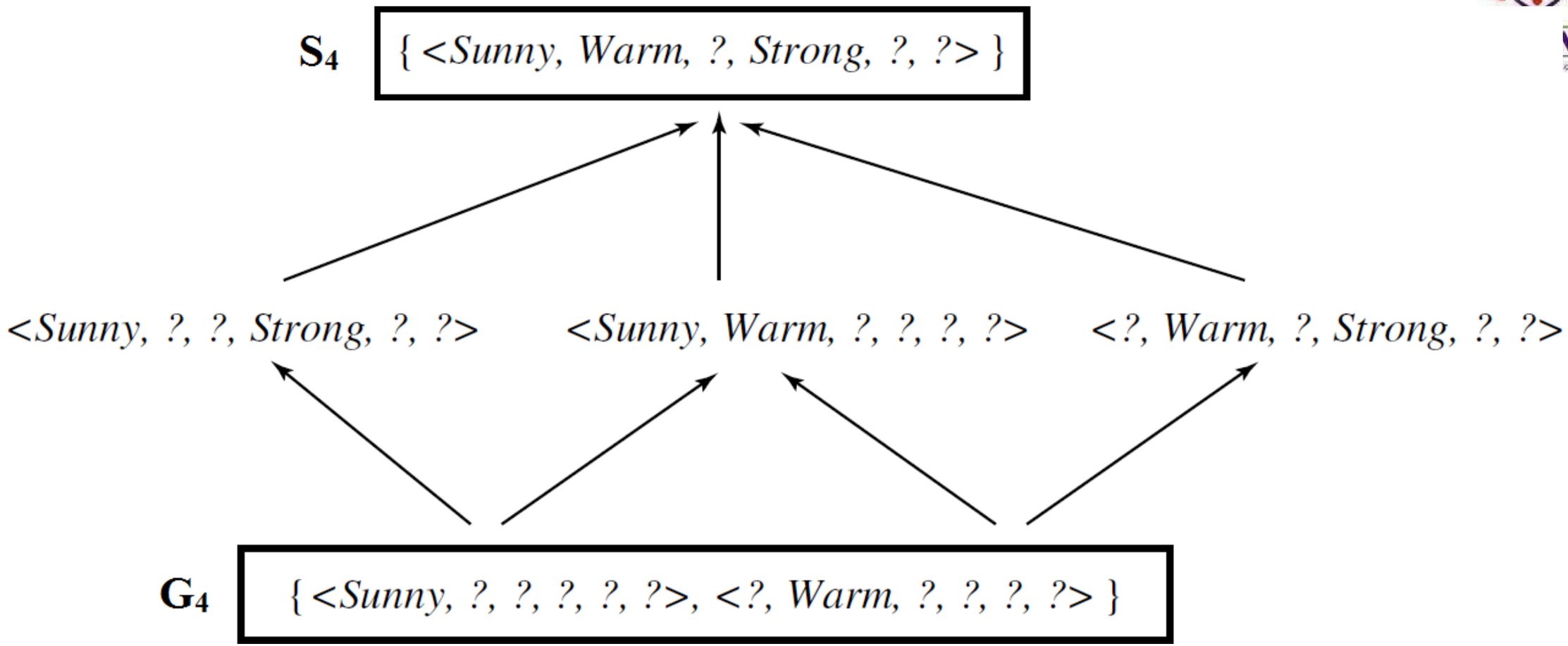
G_3 ⟨Sunny,?,?,?,?,?⟩⟨?,Warm,?,?,?,?⟩⟨?,?,?,?,?,Same⟩

G_2 ⟨?,?,?,?,?,?⟩

Fortrainingexampled,

⟨Sunny,Warm,High,Strong,CoolChange⟩+





Thefinalversionspaceforthe***EnjoySport***conceptlearningproblemand

InductiveBias

Thefundamentalquestionsforinductiveinference

- Whatifthetargetconceptisnotcontainedinthehypothesisspace?
- Canweavoidthisdifficultybyusing ahypothesisspacethatincludeseverypossiblehypothesis?
- Howdoesthesizeofthishypothesisspaceinfluencetheabilityofthealgorithmtogeneralizet ounobservedinstances?
- Howdoesthesizeofthehypothesisspaceinfluencethenumberoftrainingexamples thatmustbeobserved?

Effectofincompletehypothesisspace

Precedingalgorithmsworkiftargetfunctionis inH
Willgenerallynotworkiftargetfunctionnotin H

Considerfollowingexampleswhichrepresenttargetfunction
"sky=sunnyorsky=cloudy":

⟨SunnyWarmNormalStrongCoolChange⟩	Y
⟨CloudyWarmNormalStrongCoolChange⟩	Y
⟨RainyWarmNormalStrongCoolChange⟩	N

IfapplyCandidateEliminationalgorithmasbefore,endupwithemptyVersionSpaceAfterf

irsttwotrainingexample
S=⟨?WarmNormalStrongCoolChange⟩

Newhypothesisisoverlygeneralanditcoversthethirdnegativetrainingexample!Our

AnUnbiasedLearner

Incompletehypothesisspace

- IfcnotinH,thenconsidergeneralizingrepresentationofHtocontainc
- ThesizeoftheinstancespaceXofdaysdescribedbythesixavailableattributesis 96.Thenumberofdistinctsubsetsthatcanbedefinedoverasetxcontaining|X|elements(i.e., thesizeofthepowersetofX)is$2^{|X|}$
- Recallthatthereare96instancesin***EnjoySport***;hencethereare2^{96}possiblehypothesesinfullspaceH
- CandothisbyusingfullpropositionalcalculuswithAND,OR,NOT
- HenceHdefinedonlybyconjunctionsofattributesisbiased(containingonly973h’s)

- Let usreformulatethe ***Enjoysport***learningtaskin anunbiasedway by defininga newhypothesisspace***H'***thatcanrepresenteverysubsetofinstances;thatis,letH'correspondtoth epowersetofX.
- Onewayto definesuchan H'istoallowarbitrarydisjunctions,conjunctions,andnegationsofourearlierhypothes es.

Forinstance,thetargetconcept***"Sky*=*Sunny***or***Sky*=*Cloudy"***couldthenbedescribedas

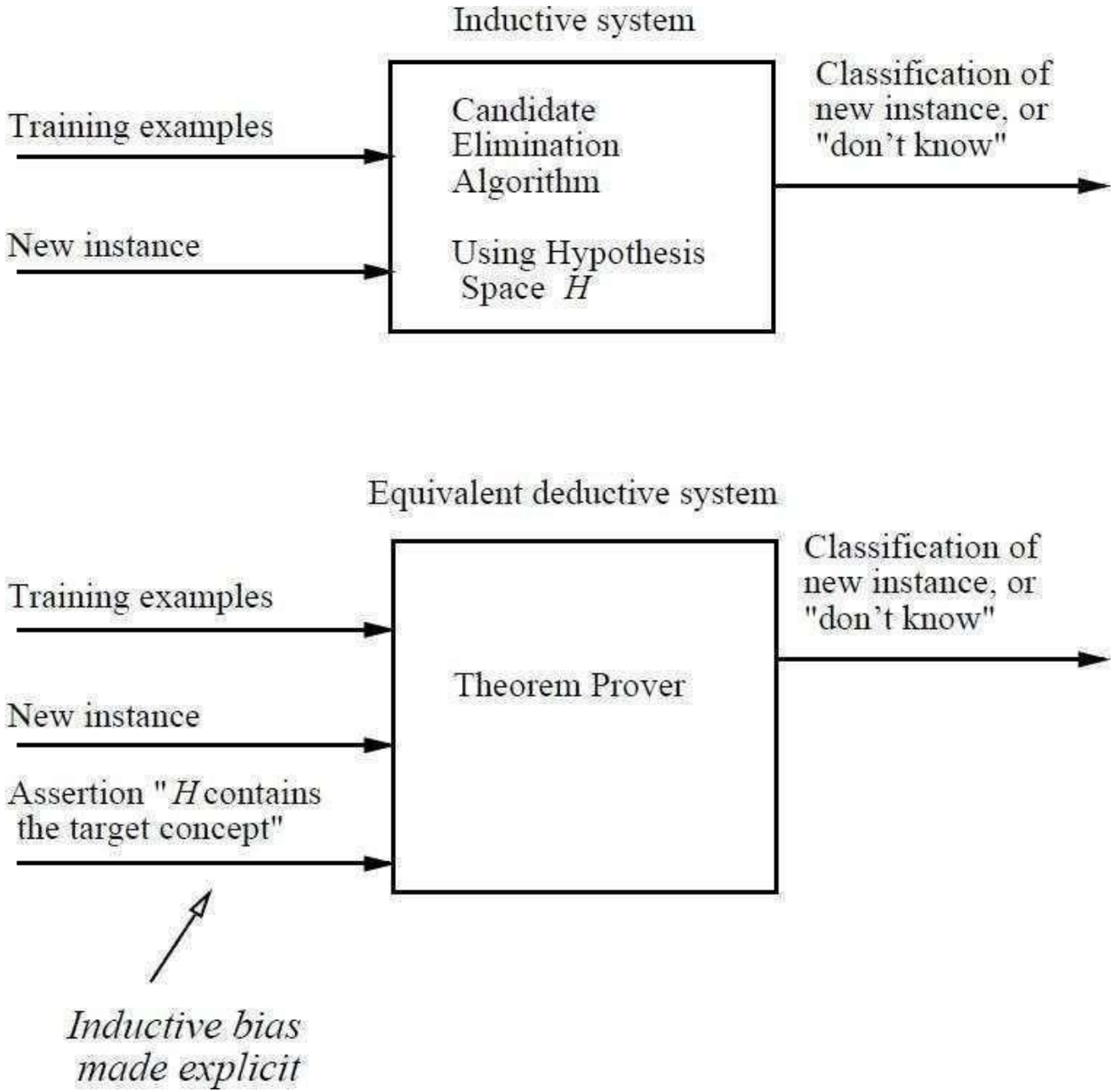
(Sunny,?,?,?,?,?)V***(Cloudy,***?,?,?,?,?)

Definition:

Consideraconceptlearningalgorithm L forthesetofinstances X.

- Letcbeanarbitraryconceptdefinedover X
- Let D_c={(x,c(x))}beanarbitrarysetoftrainingexamplesofc.
- Let $L(x_i, D_c)$ denotetheclassificationassignedtotheinstance x_i by L aftertrainingonthedata D_c.
- TheinductivebiasofLisany minimalsetofassertionsBsuch thatforanytargetconceptcandcorrespondingtrainingexamples D_c

$$(\forall \langle x_i \in X)[(B \wedge D_c \wedge x_i) \vdash L(x_i, D_c)]$$



Modelling inductive systems byequivalentdeductivesystems.

Theinput-outputbehavioroftheCANDIDATE-ELIMINATION algorithmusingahypothesisspace ***H*** is identical to that of a deductivetheoremproverutilizingthe assertion ***"H*** contains the target concept."Thisassertion is therefore called the***inductivebias***ofthe CANDIDATE-ELIMINATION algorithm.

.

characterizinginductivesystems by theirinductivebiasallowsmodellingthembytheirequivalentdeductivesystems.Thisprovidesawaytocompareinductive systemsaccording to theirpolicies forgeneralizingbeyond the observedtrainingdata

DECISIONTREELEARNING

DECISIONTREEREPRESENTATION

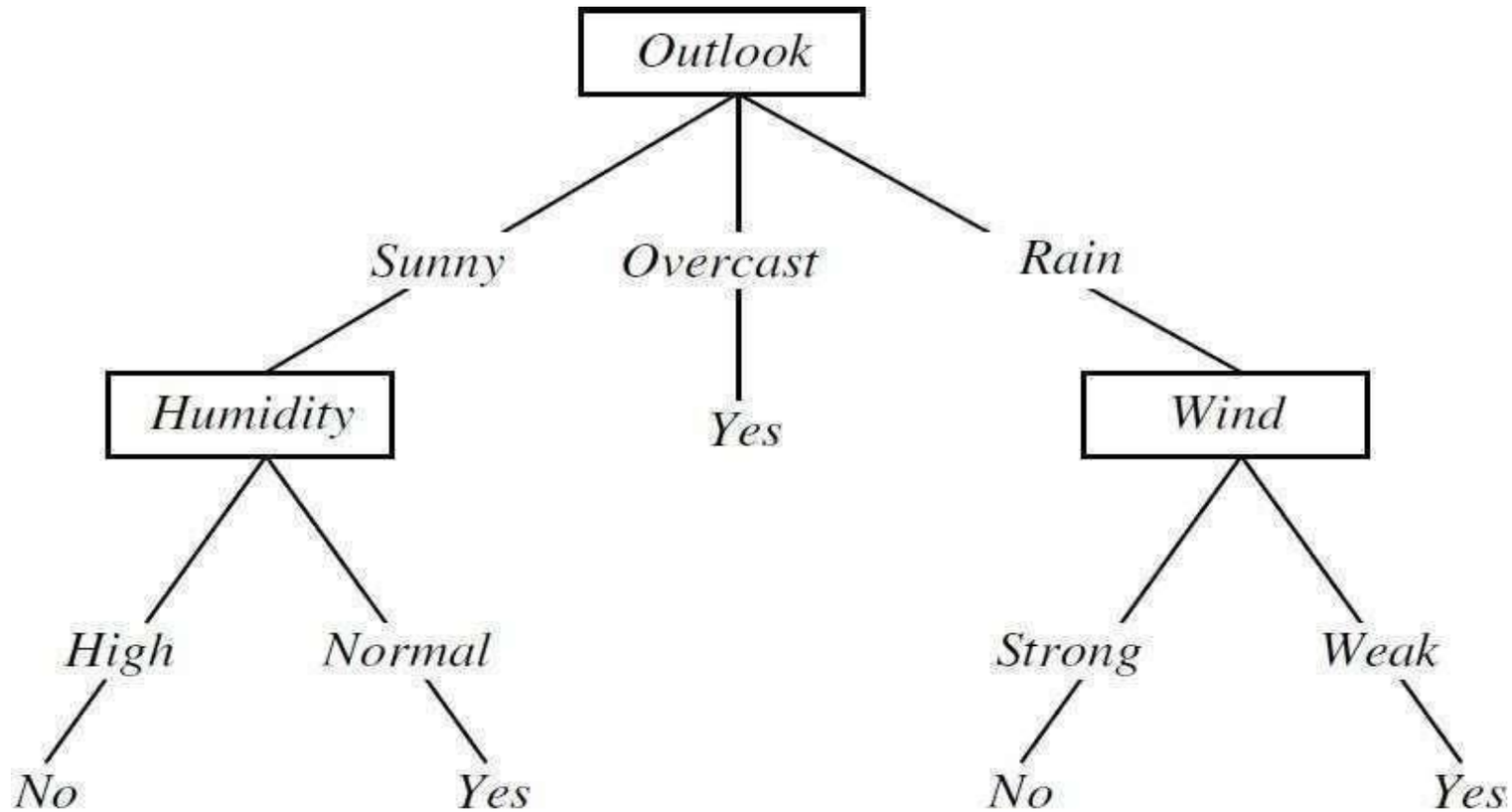


FIGURE: A decision tree for theconcept ***PlayTennis***.Anexam pleisclassified by sortingit through the tree totheappropriateleaf node, then returningtheclassific ationassociated with thisleaf

- Decision trees classify instances by sorting them down the tree from the root tosomeleaf node, which provides the classificationofthe instance.

- Each node in the tree specifies a test of some attribute of the instance, and eachbranch descending from that node corresponds to one of the possible values forthisattribute.

- Aninstanceisclassifiedbystartingattherootnodeofthetree,testingtheattribute specified by this node, then moving down the tree branch correspondingtothevalueoftheattributeinthegivenexample.Thisprocessisthenrepea tedforthe subtree rootedat the new node.

- Decisiontreesrepresentadisjunctionofconjunctionsofconstraintsontheattributevalues of instances.
- Eachpathfromthetreeroottoaleafcorrespondstoaconjunctionofattributetests,and the treeitself toa disjunctionof these conjunctions

Forexample,
Thedecisiontreeshowninabovefigurecorrespondstotheexpression

(Outlook =Sunny AHumidity = Normal)
(Outlook=Overcast)
(Outlook=RainAWind=Weak)

APPROPRIATEPROBLEMSFOR DECISIONTREELEARNING

Decisiontreelearningisgenerallybestsuitedtoproblemswiththefollowingcharacteristics:

1. ***Instancesarerepresentedbyattribute-valuepairs***–Instancesaredescribedbyafixed set of attributes and their values
2. ***The target function has discrete output values*** – The decision tree assigns aBoolean classification (e.g., yes or no) to each example. Decision tree methodseasilyextendtolearningfunctionswithmorethantwopossibleoutputvalues.
3. ***Disjunctivedescriptionsmayberequired***

4. ***Thetrainingdatamaycontainerrors***–Decisiontreelearningmethodsarerobust to errors, both errors in classifications of the training examples and errorsinthe attributevalues that describe these examples.

5. ***Thetrainingdatamaycontainmissingattributevalues–***Decisiontreemethodscanbe used evenwhen some training exampleshaveunknownvalues

- Decision tree learning has been applied to problems such as learning to classify*medical patients by their disease*, *equipment malfunctions by their cause*, and*loan applicants by their likelihood ofdefaulting on payments*.

- Suchproblems,inwhichthetaskistoclassifyexamplesintooneofadiscrete setofpossiblecategories,areoftenreferredtoas***classificationproblems.***

THEBASICDECISIONTREE ALGORITHM

- Mostalgorithmsthathavebeendevelopedforlearningdecisiontreesarevariations on a core algorithm that employs a top-down, greedy search through thespaceofpossibledecisiontrees.Thisapproachisexemplifiedbytheid3algorithmand itssuccessor C4.5

WhatistheID3algorithm?

- ID3standsforIterativeDichotomiser3
- ID3isaprecursortotheC4.5Algorithm.
- TheID3algorithmwasinventedbyRossQuinlanin1975
- Usedtogenerateadecisiontreefromagivendatasetbyemployingatop-down,greedysearch,totesteach attributeat everynodeofthetree.
- Theresultingtreeisusedtoclassifyfuturesamples.

ID3algorithm

ID3(Examples,Target_attribute,Attributes)

Examples are the training examples. Target_attribute is the attribute whose value is tobe predictedby the tree. Attributes is a list of other attributes that may be tested by the learned decision tree.Returnsadecisiontreethatcorrectlyclassifiesthegiven Examples.

- CreateaRootnodeforthetree
- IfallExamplesarepositive,Returnthesingle-nodetreeRoot,withlabel=+
- IfallExamplesarenegative,Returnthesingle-nodetreeRoot,withlabel=-
- IfAttributesisempty,Returnthesingle-nodetreeRoot,withlabel=mostcommonvalueofTarget_attributeinExamples

- OtherwiseBegin
  - A←theattributefromAttributesthatbest*classifiesExamples
  - ThedecisionattributeforRoot←A
  - Foreachpossiblevalue,v_i,ofA,
    - Addanewtreebranchbelow*Root*,correspondingtothetestA=v_i
    - Let$Examples_{vi}$,bethesubsetofExamplesthathavevalue$v_i$for*A*
    - If$Examples_{vi}$,isempty
      - Thenbelowthisnewbranchaddaleafnodewithlabel=mostcommonvalueofTarget_attribuleinExamples
      - Elsebelowthisnewbranchaddthesubtree ID3(*Examples*$_{vi}$,Targe_tattribute,Attributes–{A}))
- End
- ReturnRoot

*Thebestattributeistheonewithhighestinformationgain

WhichAttributeIstheBestClassifier?

- ThecentralchoiceintheID3algorithmisselectingwhichattributetotestat eachnode in thetree.
- Astatisticalpropertycalled***informationgain***thatmeasureshowwellagivenattributeseparatesthetraining examplesaccordingto theirtargetclassification.
- ID3uses***informationgain***measuretoselectamongthecandidateattributesateach step while growingthe tree.

ENTROPYMEASURESHOMOGENEITYOFEXAMPLES

- Todefineinformationgain,webeginbydefiningameasurecalledentropy. *Entropymeasurestheimpurityofacollectionofexamples.*
- GivenacollectionS,containingpositiveandnegativeexamplesofsometargetconcept,theentropyof Srelative to this Booleanclassificationis

$$Entropy\,(S) \equiv -p_{\oplus} \log_2 p_{\oplus} - p_{\ominus} \log_2 p_{\ominus}$$

Where,

p_+isthepropportionofpositiveexamplesinS

p_-isthepropportionofnegativeexamplesinS.

Example:Entropy

- Suppose S is a collection of 14 examples of some boolean concept, including 9positive and 5 negative examples. Thenthe entropy of Srelative to this booleanclassificationis

$$Entropy([9+, 5-]) = -(9/14)\log_2(9/14) - (5/14)\log_2(5/14)$$
$$= 0.940$$

- Theentropyis0ifallmembersofSbelongtothesameclass
- Theentropyis1whenthecollectioncontainsanequalnumberofpositiveandnegative examples
- Ifthecollectioncontainsunequalnumbersofpositiveandnegativeexamples,theentropyisbetween0and 1

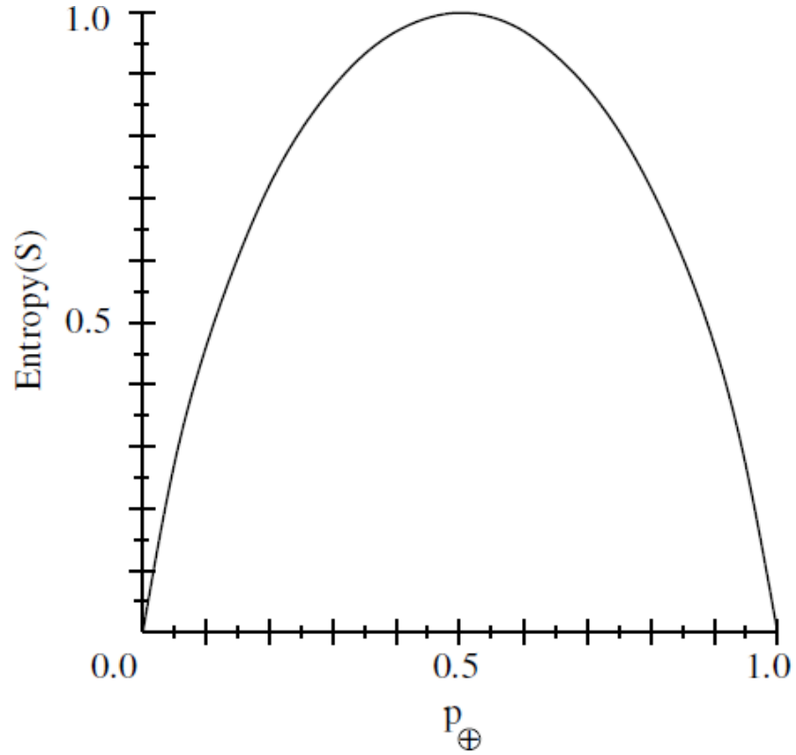


FIGURE The entropy function relative to a boolean classification, as the proportion, $p_{\oplus}$, of positive examples varies between 0 and 1.

INFORMATIONGAINMEASURESTHEEXPECTEDREDUCTIONINENTROPY

- ***Informationgain,***istheexpectedreductioninentropycausedbypartitioning theexamplesaccording tothis attribute.
- Theinformationgain,Gain(S,A)ofanattributeA,relativetoacollectionofexamplesS, is defined as

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

Example:Informationgain

Let, $Values(Wind)=\{Weak, Strong\}$

$S = [9+, 5-]$

$S_{Weak} = [6+, 2-]$

$S_{Strong} = [3+, 3-]$

Informationgainofattribute*Wind*:

$$Gain(S, Wind) = Entropy(S) - 8/14\, Entropy(S_{Weak}) - 6/14\, Entropy(S_{Strong})$$

$$= 0.94 - (8/14)*0.811 - (6/14)*1.00$$

$$= 0.048$$

AnIllustrativeExample

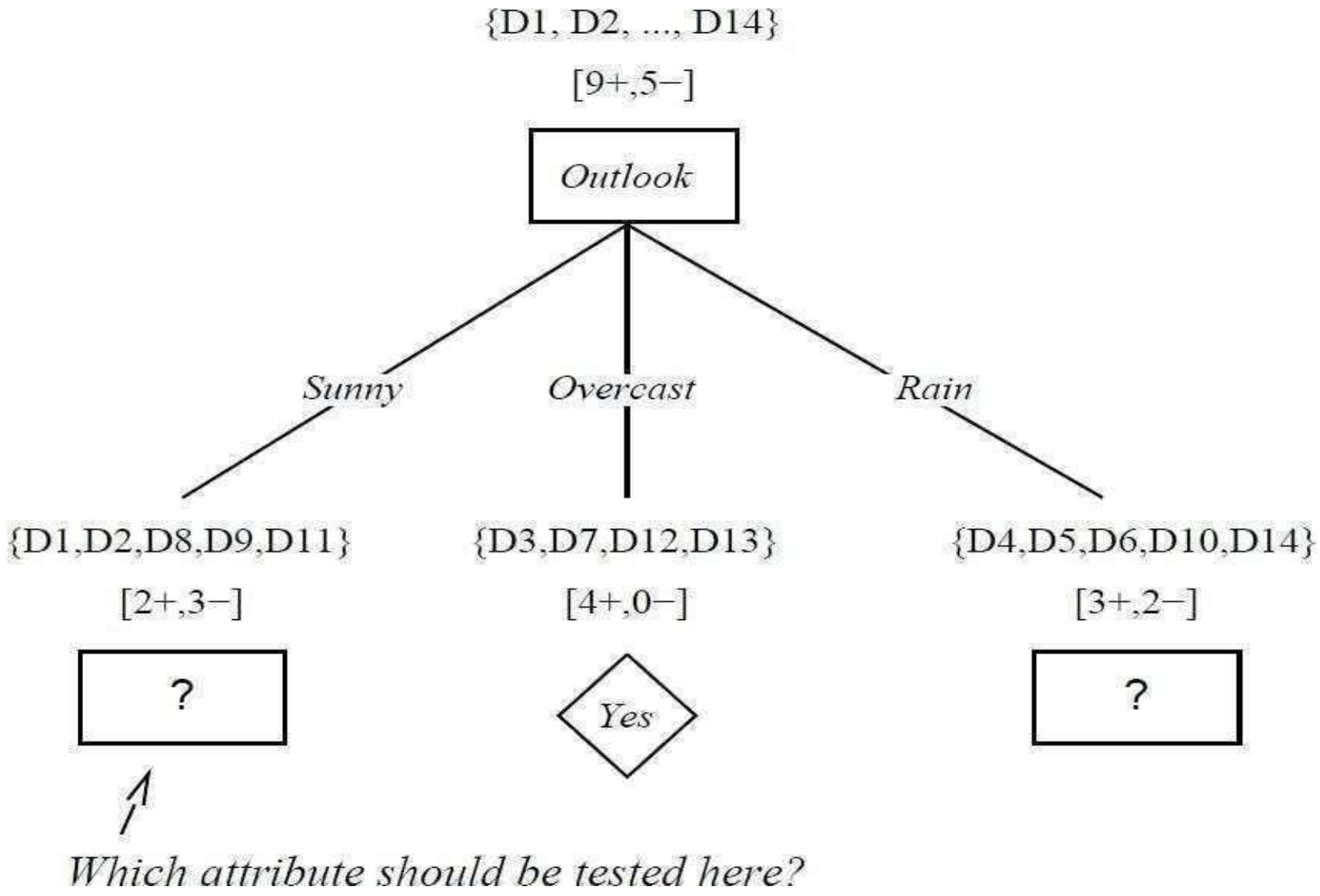
- ToillustratetheoperationofID3,considerthelearningtask representedbythetrainingexamplesofbelowtable.
- Herethetargetattribute***PlayTennis***,whichcanhavevalues***yes*** or***no***fordifferentdays.
- Considerthefirststepthroughthealgorithm,inwhichthetopmostnodeofthedecision treeiscreated.

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Theinformationgainvaluesforallfourattributesare

- Gain(S,Outlook) =0.246
- Gain(S,Humidity) =0.151
- Gain(S,Wind) =0.048
- Gain(S,Temperature) =0.029

- Accordingtotheinformationgainmeasure,the***Outlook***attributeprovidesthebest prediction of the target attribute, ***PlayTennis***, over the training examples.Therefore,***Outlook***isselectedasthedecisionattributefortherootnode, andbranchesarecreatedbelowtherootforeachofitspossiblevaluesi.e.,


{D1, D2, ..., D14}
[9+,5−]
Outlook
Sunny
Overcast
Rain
{D1,D2,D8,D9,D11}
[2+,3−]
{D3,D7,D12,D13}
[4+,0−]
{D4,D5,D6,D10,D14}
[3+,2−]
?
Yes
?
Which attribute should be tested here?

$S_{sunny} = \{D1, D2, D8, D9, D11\}$

$Gain(S_{sunny}, Humidity) = .970 - (3/5)\,0.0 - (2/5)\,0.0 = .970$

$Gain(S_{sunny}, Temperature) = .970 - (2/5)\,0.0 - (2/5)\,1.0 - (1/5)\,0.0 = .570$

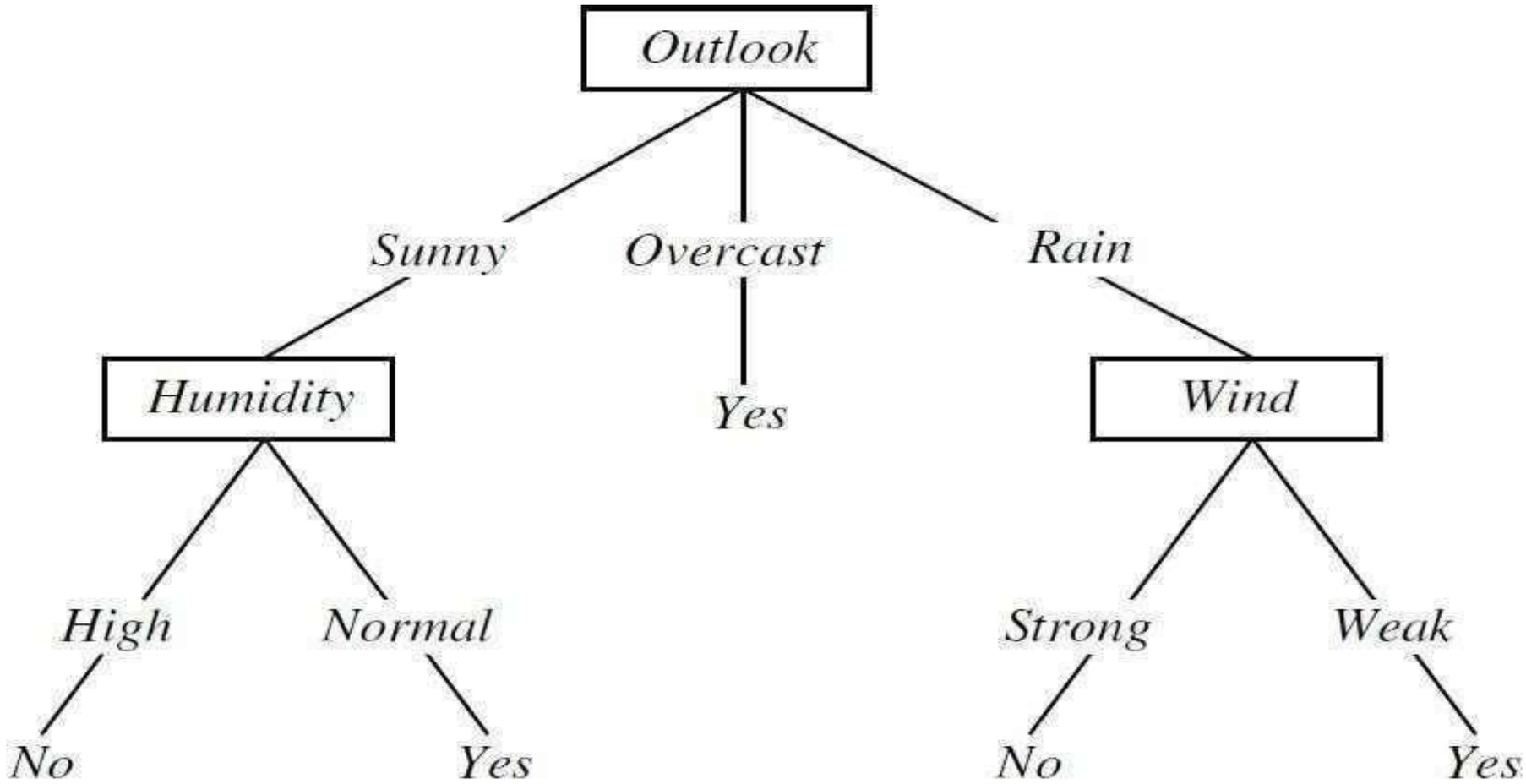
$Gain(S_{sunny}, Wind) = .970 - (2/5)\,1.0 - (3/5)\,.918 = .019$

$S_{Rain} = \{D4, D5, D6, D10, D14\}$

$$Gain(S_{Rain}, Humidity) = 0.970 - (2/5)1.0 - (3/5)0.917 = 0.019$$

$$Gain(S_{Rain}, Temperature) = 0.970 - (0/5)0.0 - (3/5)0.918 - (2/5)1.0 = 0.019$$

$$Gain(S_{Rain}, Wind) = 0.970 - (3/5)0.0 - (2/5)0.0 = 0.970$$


Outlook
Sunny
Overcast
Rain
Humidity
Yes
Wind
High
Normal
Strong
Weak
No
Yes
No
Yes

HYPOTHESISSPACESEARCHINDECISIONTREEL EARNING

- ID3canbecharacterizedassearchingaspaceofhypothesesforonethatfits thetraining examples.
- ThehypothesisspacesearchedbyID3isthesetofpossibledecisiontrees.
- ID3 performs a ***simple-to complex, hill-climbing search*** through this hypothesisspace,beginningwiththeemptytree,thenconsideringprogressivelymoreela boratehypothesesinsearchofadecisiontreethatcorrectlyclassifiesthetrainingdata

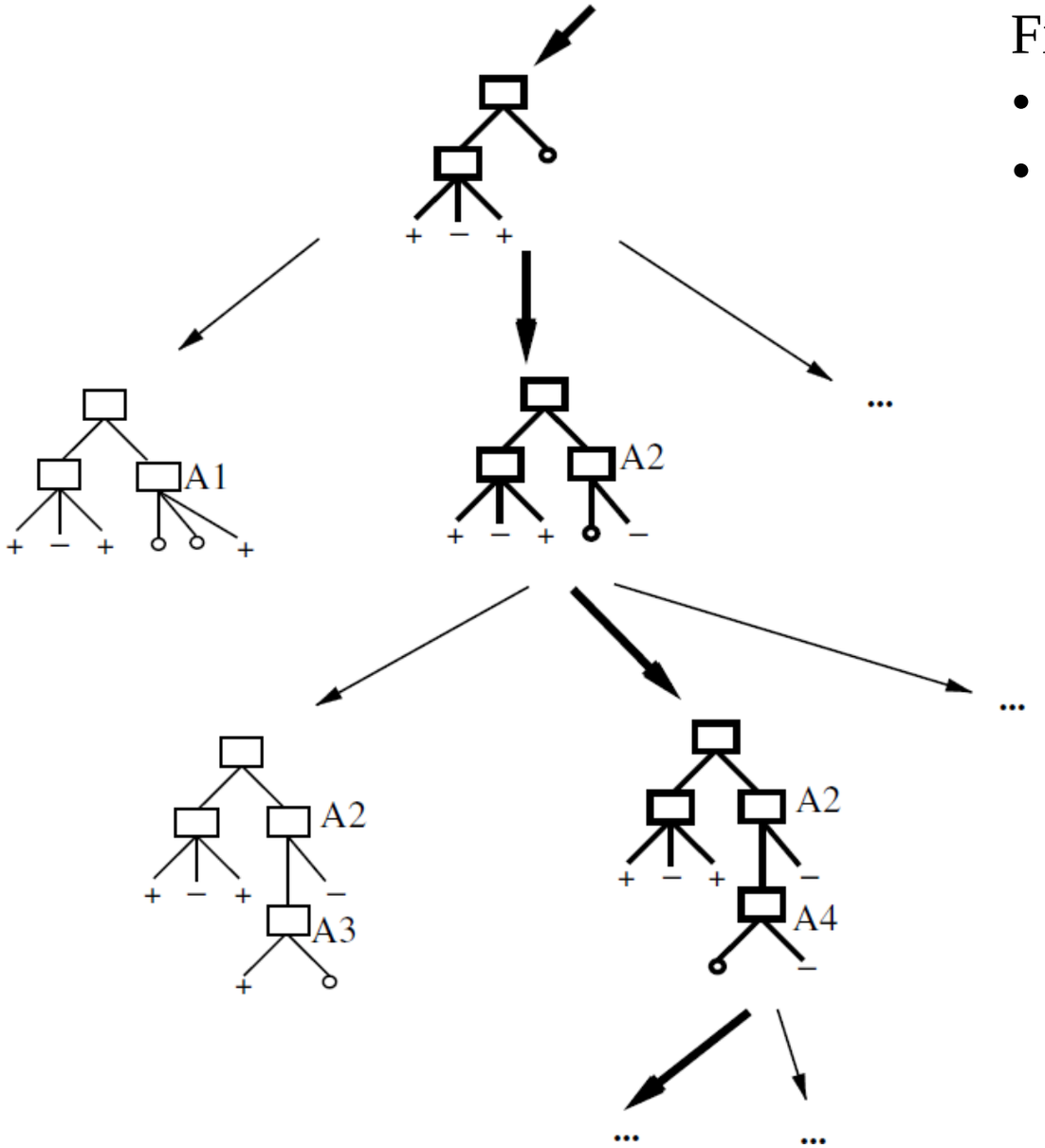


Figure:

- HypothesisspacesearchbyID3.
- ID3searchesthroughthespaceofpossible decision trees from simplest toincreasinglycomplex,guidedbytheinfo rmationgain heuristic

1. ID3's hypothesis space of all deision trees is a ***complete*** space of finite discrete-valued functions, relative to the available attributes. Because every finite discrete-valuedfunction can be represented bysome decision tree

- ID3avoidsoneofthemajorrisksofmethodsthat***searchincompletehypothesisspaces*** :thatthehypothesisspacemightnotcontainthetargetfunction.

2. ID3maintains***onlyasinglecurrenthypothesis***asitsearchesthroughthespaceof decision trees.

Forexample,withtheearlierversionspacecandidateeliminationmethod,whichmaint ains theset of ***all*** hypotheses consistentwiththeavailable trainingexamples.

Bydeterminingonlyasinglehypothesis,ID3losesthecapabilitiesthatfollowfromexplici tlyrepresenting allconsistenthypotheses.

For example, it does not have the ability to determine how many alternativedecisiontreesareconsistentwiththeavailabletrainingdata,orto pose newinstancequeriesthat optimallyresolveamongthesecompetinghypotheses

3. ID3 in its pure form performs ***no backtracking in its search***. Once it selects anattribute to test at a particular level in the tree, it never backtracks to reconsider thischoice.

- In the case of **ID3,** a locally optimal solution corresponds to the decision tree itselects along the single search path it explores. However, this locally optimalsolution may be less desirable than trees that would have been encountered along adifferentbranch of the search.

4. ID3*usesalltrainingexamplesateachstep*inthesearchtomakestatisticallybaseddecisions regarding how to refineits current hypothesis.

- Oneadvantageofusingstatisticalpropertiesofalltheexamplesisthattheresultingsearch ismuch ***less sensitivetoerrors***inindividualtrainingexamples.
- **ID3**canbeeasilyextendedtohandlenoisytrainingdataby modifyingitsterminationcriteriontoacceptthypotheses thatimperfectlyfitthetrainingdata.

INDUCTIVEBIASINDECISIONTREELEARNING

Inductive biasistheset of assumptionsthat,together withthe training data,deductivelyjustifytheclassificationsassignedbythelearnertofutureinstances

Givenacollectionoftrainingexamples,therearetypicallymanydecisiontreesconsistent with theseexamples.Which ofthese decisiontreesdoes ID3choose?

ID3searchstrategy

(a) selectsinfavourofshortertreesoverlongerones

(b) selectstreesthatplacetheattributeswithhighestinformationgainclosesttotheroot.

ApproximateinductivebiasofID3: Shortertreesarepreferredoverlargertrees

- Consideranalgorithmthatbeginswiththeemptytreeandsearches***breadthfirst*** throughprogressivelymorecomplextrees.
- Firstconsideringalltreesofdepth1,thenalltreesofdepth2,etc.
- Onceitfindsadecisiontreeconsistentwiththetrainingdata,itreturns thesmallestconsistenttreeatthatsearchdepth(e.g.,thetreewiththefewestnodes).
- LetuscallthisbreadthfirstsearchalgorithmBFS-ID3.
- BFS-ID3findsashortestdecisiontreeandthusexhibitsthebias"shortertrees arepreferredover longer trees.

- ID3 can be viewed as an efficient approximation to BFS-ID3, using a greedyheuristic search to attempt to find the shortest tree without conducting the entirebreadth-firstsearchthrough thehypothesis space.
- BecauseID3usestheinformationgainheuristicandahillclimbingstrategy, itexhibitsamorecomplexbiasthanBFS-ID3.
- Inparticular,itdoesnotalwaysfindtheshortestconsistenttree,anditisbiased tofavourtreesthatplaceattributeswithhighinformationgainclosesttotheroot.

RestrictionBiasesandPreferenceBiases

DifferencebetweenthetypesofinductivebiasexhibitedbyID3andbytheCANDIDATE-ELIMINATIONAlgorithm.

ID3

- ID3searchesacompletehypothesisspace
- Itsearchesincompletelythroughthisspace,fromsimpletocomplexhypotheses,untilitstermination conditionismet
- Itsinductivebiasissolelyaconsequenceoftheorderingofhypothesesbyitssearchstrategy. Itshypothesisspaceintroducesnoadditionalbias

CANDIDATE-ELIMINATIONAlgorithm

- TheversionspaceCANDIDATE-ELIMINATIONAlgorithmsearchesanincompletehypothesisspace
- Itsearchesthisspacecompletely,findingeveryhypothesisconsistentwiththetrainingdata.
- Its inductive bias is solely a consequence of the expressive power of its hypothesisrepresentation.Itssearchstrategyintroducesnoadditionalbias

RestrictionBiasesandPreferenceBiases

- The inductive bias of ID3 is a*preference* for certain hypotheses over others (e.g.,preference for shorter hypotheses over larger hypotheses), with no hard restrictiononthehypothesesthatcanbeeventuallyenumerated.Thisformofbiasis called a***preferencebias***or ***a search bias***.

- The bias of the CANDIDATE ELIMINATION algorithm is in the form of a*categorical* restriction on the set of hypotheses considered. This form of bias istypicallycalleda***restriction bias***or***alanguagebias***.

Whichtypeofinductivebiasispreferredinordertogeneralizebeyondthetrainingdata,apreferenc e bias or a restriction bias?

- Apreference bias is more desirable than arestriction bias, because it allows thelearner to work within a complete hypothesis space that is assured to contain theunknown target function.
- In contrast, a restriction bias that strictly limits the set of potential hypotheses isgenerally less desirable, because it introduces thepossibilityofexcluding theunknown target function altogether.

Occam'srazor

Occam's razor: is the problem-solving principle that the simplest solution tends to bethe right one. When presented with competing hypotheses to solve a problem, oneshouldselect the solutionwith the fewest assumptions.

Occam'srazor:"***Preferthesimplesthypothesisthatfitsthedata***".

WhyPreferShortHypotheses?

Argumentinfavour:

Fewershorthypothesesthanlongones:

- Shorthypothesesfitsthetrainingdatawhichare*lesslikely*tobe*coincident*
- Longerhypothesesfitsthetrainingdatamightbe*coincident*.

Manycomplexhypothesesthatfitthecurrenttrainingdatabutfailtogeneralizecorrectly tosubsequent data.

Argumentopposed:

- There are few small trees, and our priori chance of finding one consistent with anarbitrary set of data is therefore small. The difficulty here is that there are verymanysmallsetsofhypothesesthatonecandefine*butunderstoodbyfewerlearner*.
- The size of a hypothesisis determined by the representation used ***internally*** by thelearner.Occam'srazorwillproduce*twodifferenthypothesesfromthesametrainingexampleswhenitisappliedbytwolearners*,bothjustifyingtheircontradictory conclusions byOccam'srazor.OnthisbasiswemightbetemptedtorejectOccam'srazor altogether.

ISSUESINDECISIONTREELEARNING

1. AvoidingOverfittingtheData

   Reduced error

   pruningRulepost-pruning

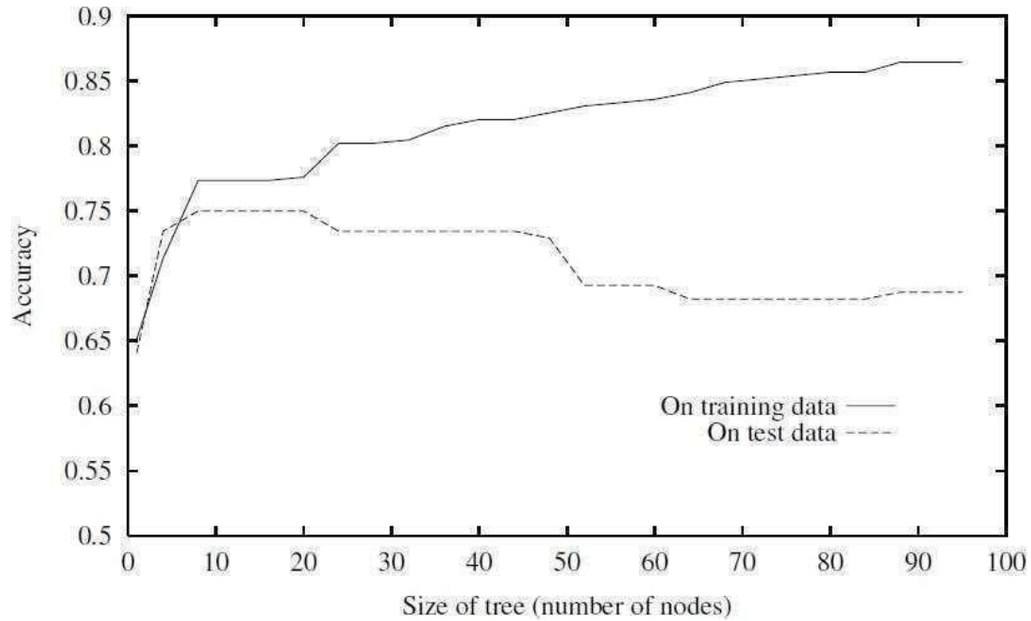
2. IncorporatingContinuous-ValuedAttributes
3. AlternativeMeasuresforSelectingAttributes
4. HandlingTrainingExampleswithMissingAttributeValues
5. HandlingAttributeswithDifferingCosts

1. AvoidingOverfittingtheData

- The ID3 algorithm grows each branch of the tree just deeply enough to perfectlyclassify the training examples but it can lead to difficulties when there is noise inthe data, or when the number of training examples is too small to produce arepresentative sample of the true target function. This algorithm can produce treesthat***overfit***the training examples.

- ***Definition-Overfit:***GivenahypothesisspaceH,ahypothesis$h \in$Hissaidtooverfit the training data if there exists some alternative hypothesis $h' \in$H, suchthat*h*hassmallererrorthan*h'*overthetrainingexamples,but*h'*hasasmallererrorthan *h*over the entire distribution of instances.

- Thebelowfigureillustratestheimpactofoverfittinginatypicalapplicationofdecisiontreelearning.



- The**horizontalaxis**ofthis plotindicatesthetotalnumberofnodesinthedecisiontree,asthetreeisbeingconstructed.The**verticalaxis**indicates theaccuracyofpredictionsmadebythetree.
- The**solidline**showstheaccuracyofthedecisiontreeoverthetrainingexamples.The**brokenline**showsaccuracymeasuredoveranindependentsetoftestexample
- The**accuracy**ofthetreeoverthe**trainingexamplesincreases**monotonicallyasthetreeisgrown.The **accuracy**measuredovertheindependenttestexamples**firstincreases**,**thendecreases**.

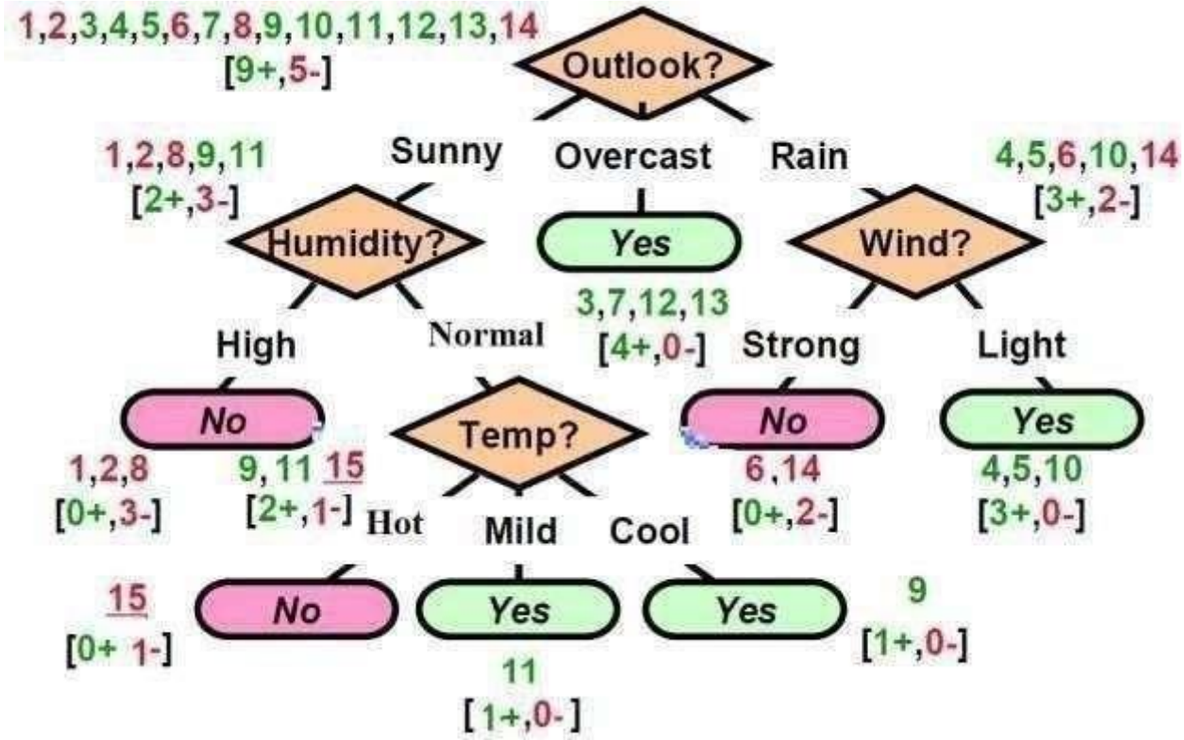
Howcanitbepossiblefortreehtofitthetrainingexamplesbetterthanh',butforittoperformmorepoorlyoversubsequentexamples?

1. Overfittingcanoccurwhenthetrainingexamplescontainrandomerrorsornoise
2. Whensmallnumbersofexamplesareassociatedwithleafnodes.

NoisyTrainingExample

Example15:<*Sunny,Hot,Normal,Strong,*->

- Exampleisnoisybecausethecorrectlabelis+
- Previouslyconstructedtreemisclassifiesit



Criterionusedtodeterminethecorrectfinaltreesize

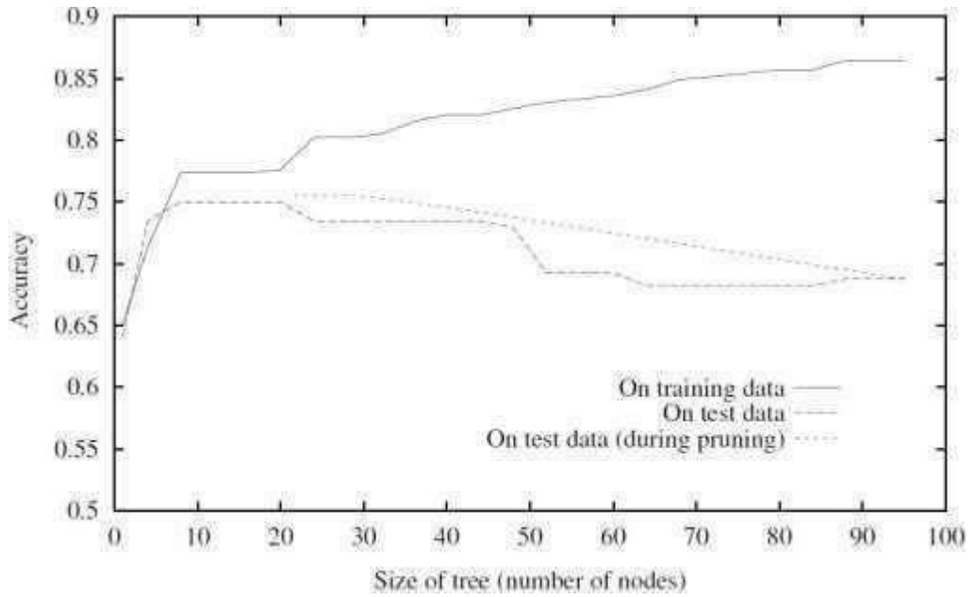
- Useaseparatesetofexamples,distinctfromthetrainingexamples,toevaluatetheutilityof post- pruningnodesfrom thetree
- Usealltheavailabledatafortraining,butapply*astatisticaltest*toestimatewhetherexpandi ngpruning)a particularnodeis likelyto produceanimprovementbeyond thetrainingset (or
- Usemeasureofthecomplexityforencodingthetrainingexamplesandthedecisiontree,h altinggrowth of the tree when this encoding size is minimized. This approach istheMinimumDescriptionLength called
  - *MDL–*

Minimize:size(tree)+size(misclassifications(tree))

Reduced-ErrorPruning

- ***Reduced-error pruning***, is to consider each of the decision nodes in the tree to becandidates for pruning
- ***Pruning*** a decision node consists of removing the subtree rooted at that node,making it a leaf node, and assigning it the most common classification of thetrainingexamples affiliated with that node
- Nodesareremovedonlyiftheresultingprunedtreeperformsnoworsethan-theoriginaloverthevalidationset.
- Reduced error pruning has the effect that any leaf node added due to coincidentalregularitiesinthetrainingsetislikelytobeprunedbecausethesesamecoincidencesare unlikelyto occur inthe validationset

Theimpactofreduced-errorpruningontheaccuracyofthedecisiontreeisillustratedinbelowfigure



- The additional line in figure shows ***accuracy over the test examples*** as the tree is pruned. Whenpruning begins, the tree is at its maximum size and lowest accuracy over the test set. As pruningproceeds,thenumberofnodesisreducedandaccuracyoverthetestsetincreases.
- The available data has been split into three subsets: the training examples, the validation examplesused for pruning the tree, and a set of test examples used to provide an unbiased estimate ofaccuracyoverfutureunseen examples.Theplotshowsaccuracyoverthetrainingandtestsets.

Summary

- Statisticaltheoryprovidesabasisforestimatingthetrueerror ($error_D h$)ofhypothesish,basedonitsobservederror($error_S h$) overasampleSofdata.
- In general, theproblem of estimatingconfidence intervals isapproachedbyidentifyingtheparametertobeestimated($error_D h$)and anestimator($error_S h$)forthisquantity.
- Becausetheestimatorisarandomvariableitcanbecharacterisedbyt heprobabilitydistributionthatgovernsitsvalue.

unbiasedestima$_2$tor,theobservedvalueoftheestimatorislikelyto varyfromoneexperimenttoanother.

Thevarianceofthedistributiongoverningtheestimateisl
ikelyto orcharacteriseshowwidelythis

Confidenceintervalscanthenbecalculatedbydeterminingthein
terval that containsthedesired probabilitymass under
thisdistribution.

Acause ofestimationerroristhevarianceintheestimate.Evenwithan

Prodigy and Explanation Based Learning

Prodigy defines a set of target concepts to learn, e.g., which operator given the current state takes you to the goal state?

An example of a rule learned by Prodigy in the block-stacking problem is:

IF One subgoal to be solved is On(x,y) AND
One subgoal to be solved is On(y,z)
THEN Solve the subgoal On(y,z) before On(x,y)

Prodigy and Explanation Based Learning

The rationale behind the rule is that it would avoid a conflict when stacking blocks.

Prodigy learns by first encountering a conflict, then explaining the reason for the conflict and creating a rule like the one above.

Experiments show an improvement in efficiency by a factor of two to four.

Problems with EBL

- ✓ The number of control rules that must be learned is very large.
- ✓ If the control rules are many, much time will be spent looking for the best rule.

Utility analysis is used to determine what rules to keep and what rules to forget.

Prodigy:

328 possible rules ⟶ 69 pass test ⟶ 19 were retained

Problems with EBL

- ✓ Another problem with EBL is that it is sometimes intractable to create an explanation for the target concept.

  For example, in chess, learning a concept like:
  "states for which operator A leads to a solution"
  The search here grows exponentially.

Summary

- Different from inductive learning, analytical learning looks for a hypothesis that fit the background knowledge and covers the training examples.

- Explanation based learning is one kind of analytical learning that divides into three steps:
  a. Explain the target value for the current example
  b. Analyze the explanation (generalize)
  c. Refine the hypothesis

Summary

- Prolog-EBG constructs intermediate features after analyzing examples.
- Explanation based learning can be used to find search control rules.
- In all cases we depend on a perfect domain theory.

Machine Learning

Chapter 12. Combining Inductive and Analytical Learning

Tom M. Mitchell

Inductive and Analytical Learning

Inductive learning

- Hypothesis fits data
- Statistical inference
- Requires little prior knowledge
- Syntactic inductive bias

Analytical learning

- Hypothesis fits domain the
- Deductive inference
- Learns from scarce data
- Bias is domain theory

What We Would Like

Inductive learning ⟵ Analytical learning

Plentiful data
No prior knowledge

Perfect prior knowledge
Scarce data

General purpose learning method:

- No domain theory → learn as well as inductive methods
- Perfect domain theory → learn as well as Prolog-EBG
- Accomodate arbitrary and unknown errors in domain theory
- Accomodate arbitrary and unknown errors in training data

Domain theory:

Cup ← Stable, Liftable, Open Vessel

Stable ← BottomIsFlat

Liftable ← Graspable, Light

Graspable ← HasHandle

Open Vessel ← HasConcavity, ConcavityPointsUp

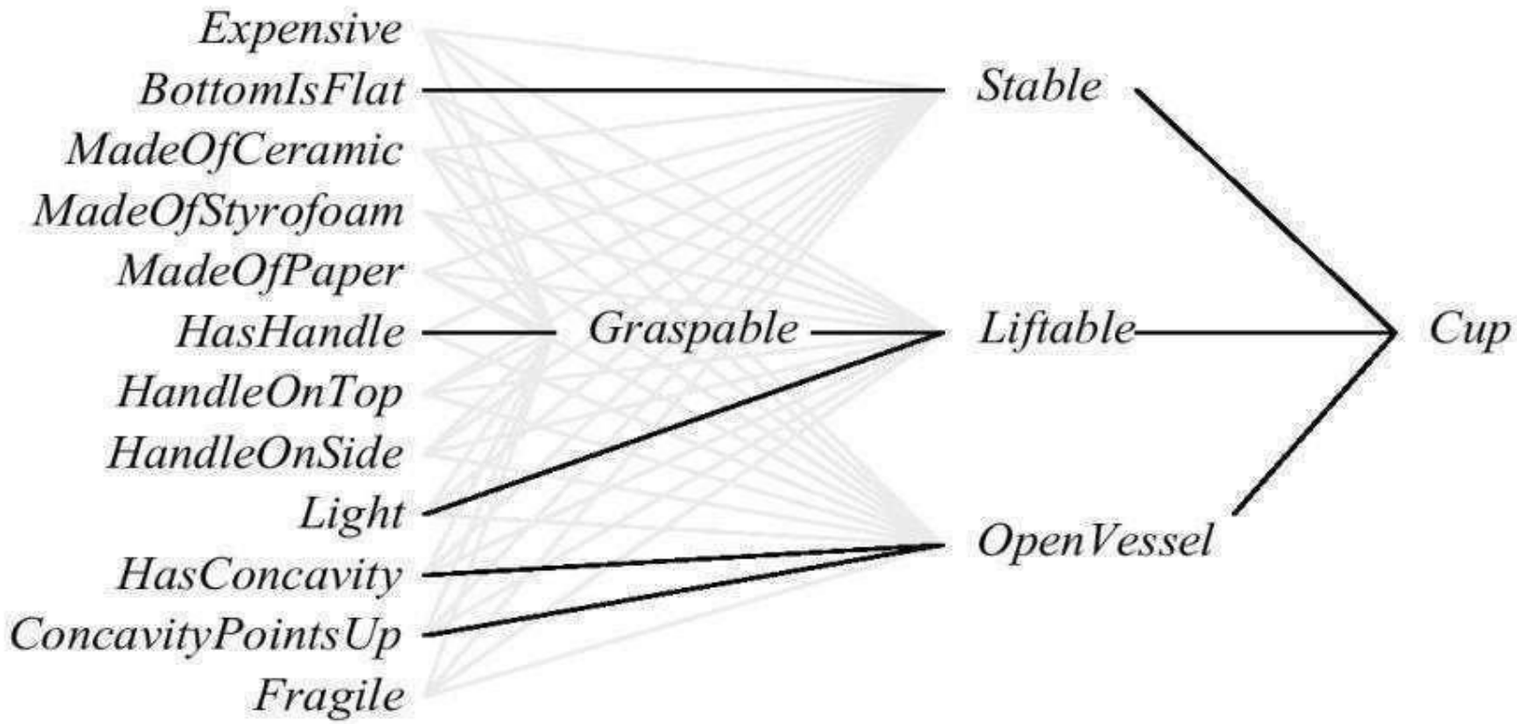
Training examples:

	Cups				Non-Cups					
BottomIsFlat	✓	✓	✓	✓	✓	✓	✓			✓
ConcavityPoints Up	✓	✓	✓	✓	✓		✓	✓		
Expensive	✓		✓				✓		✓	
Fragile	✓	✓			✓	✓		✓		✓
HandleOnTop					✓		✓			
HandleOnSide	✓			✓					✓	
HasConcavity	✓	✓	✓	✓	✓		✓	✓	✓	✓
HasHandle	✓			✓	✓		✓		✓	
Light	✓	✓	✓	✓	✓	✓	✓		✓	
MadeOfCeramic	✓				✓		✓	✓		
MadeOfPaper				✓					✓	
MadeOfStyrofoam		✓	✓			✓				✓

KBANN

- KBANN (data D, domain theory B)
  1. Create a feedforward network h equivalent to B
  2. Use BACKPROP to tune h to t D

Neural Net Equivalent to Domain Theory



Creating Network Equivalent to Domain Theory

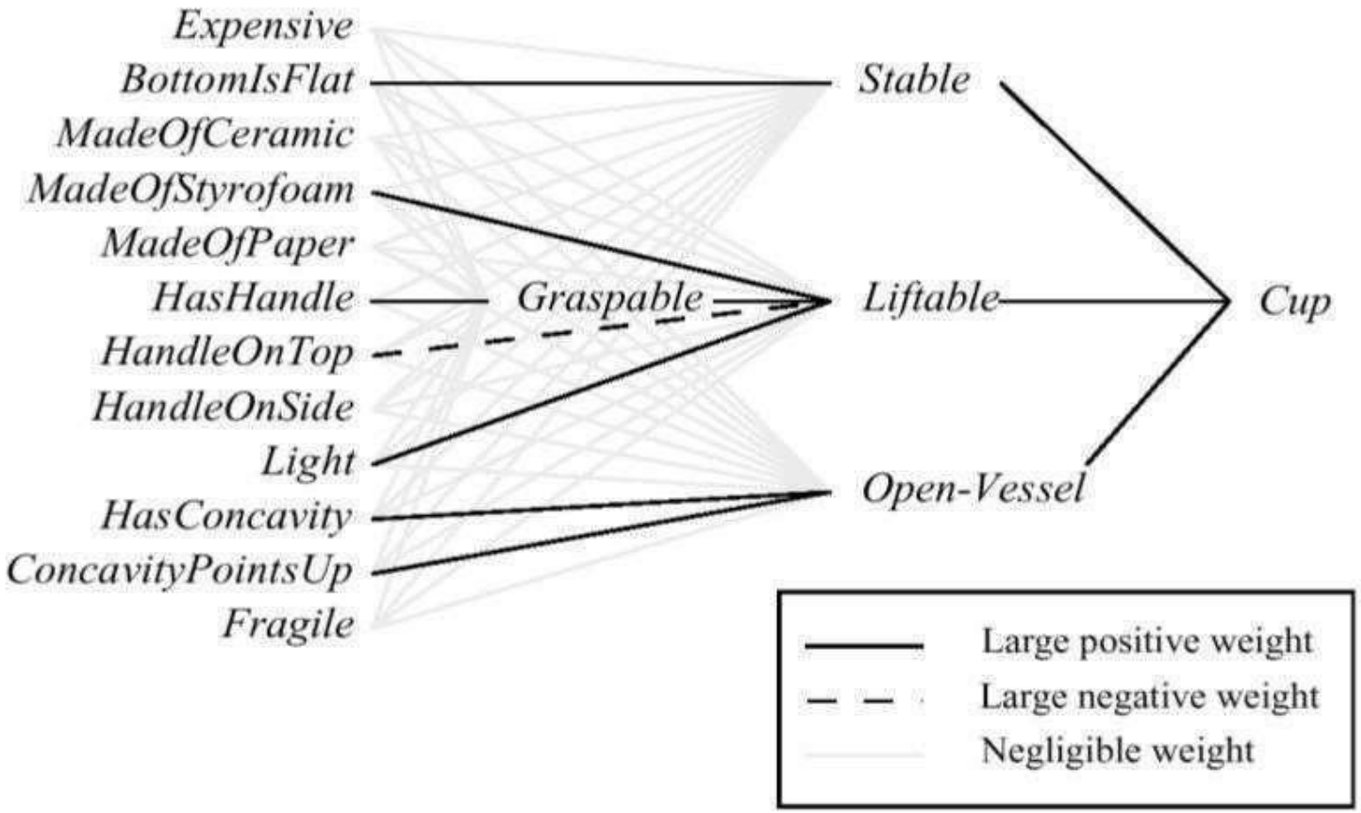
Create one unit per horn clause rule (i.e., an AND unit)

- Connect unit inputs to corresponding clause antecedents
- For each non-negated antecedent, corresponding input weight $w \leftarrow W$, where W is some constant
- For each negated antecedent, input weight $w \leftarrow -W$
- Threshold weight $w_0 \leftarrow -(n-.5)W$, where n is number of non-negated antecedents

Finally, add many additional connections with near-zero weights

$$Liftable \leftarrow Graspable, \neg Heavy$$

Result of refining the network



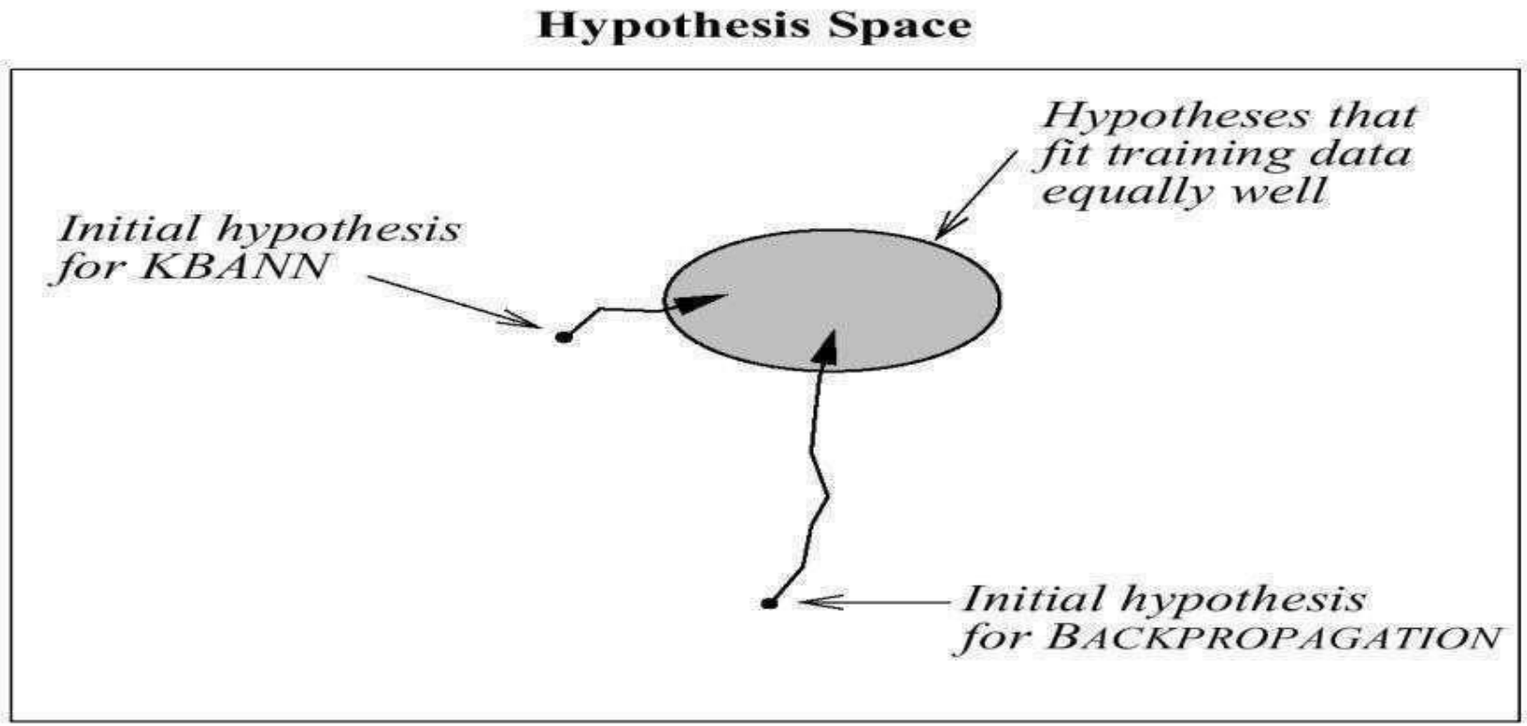
KBANN Results

Classifying promoter regions in DNA leave one out testing:

- Backpropagation : error rate 8/106
- KBANN: 4/106

Similar improvements on other classification, control tasks.

Hypothesis space search in KBANN

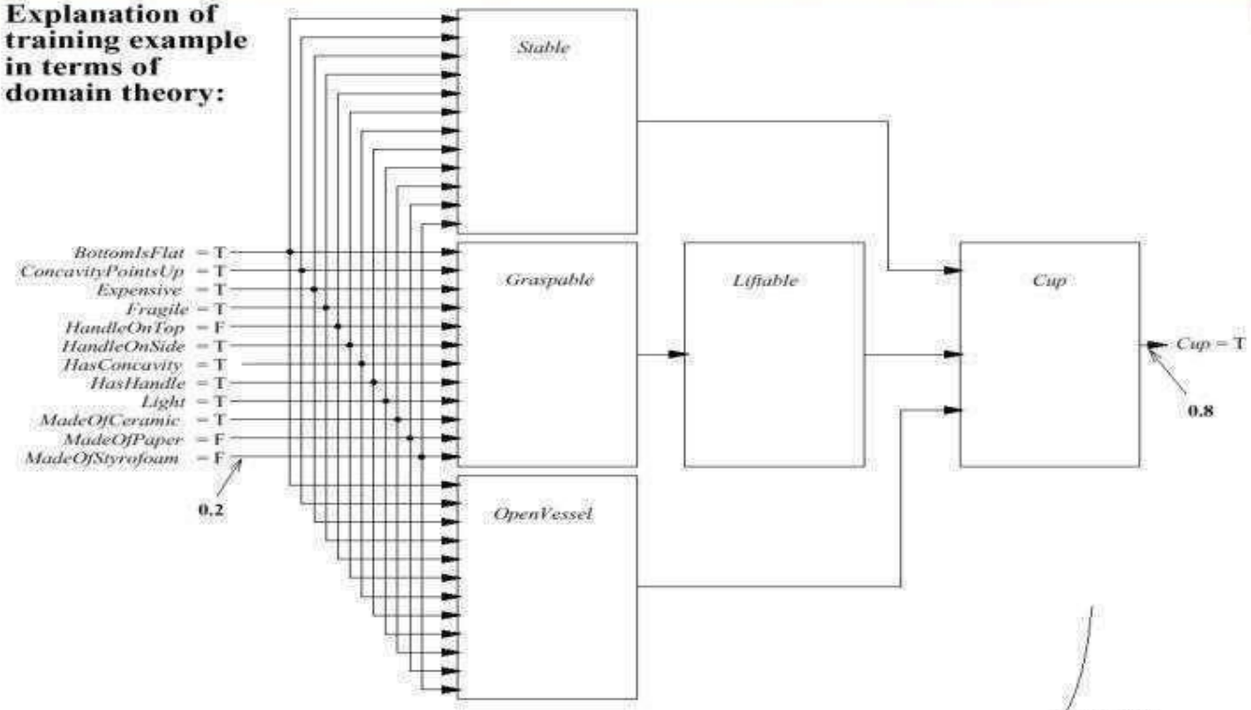


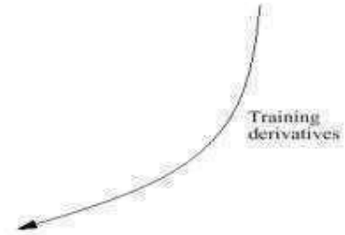
EBNN

Key idea:

- Previously learned approximate domain theory
- Domain theory represented by collection of neural networks
- Learn target function as another neural network

Explanation of training example in terms of domain theory:





Target network:

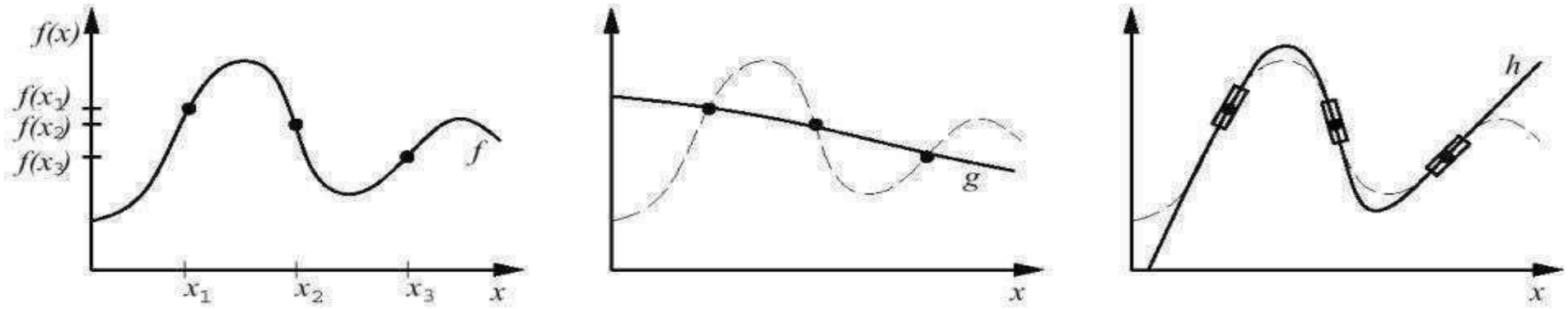
Modified Objective for Gradient Descent

$$E = \sum_i \left[(f(x_i) - \hat{f}(x_i))^2 + \mu_i \sum_j \left(\frac{\partial A(x)}{\partial x^j} - \frac{\partial \hat{f}(x)}{\partial x^j} \right)^2_{(x=x_i)} \right]$$

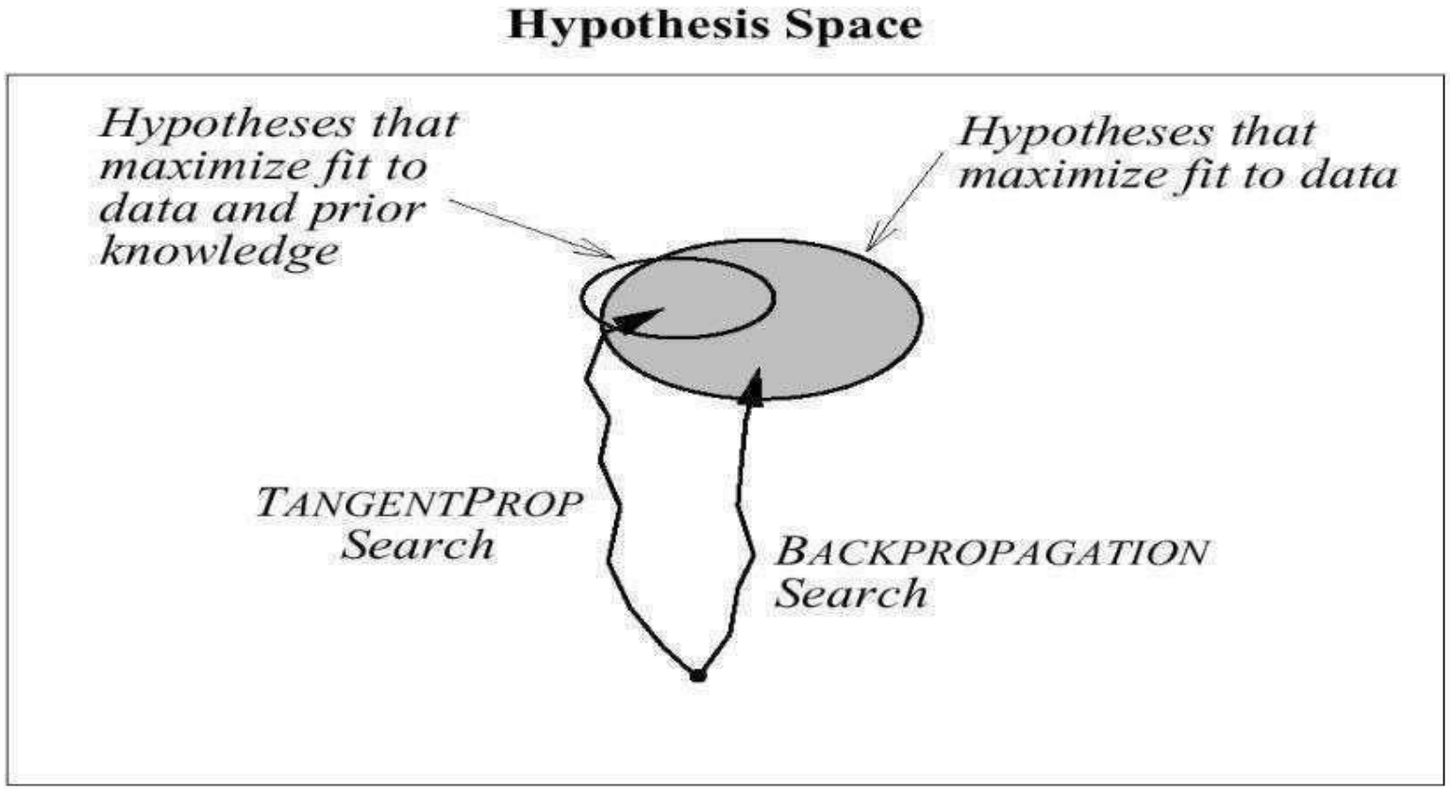
where

$$\mu_i \equiv 1 - \frac{|A(x_i) - f(x_i)|}{c}$$

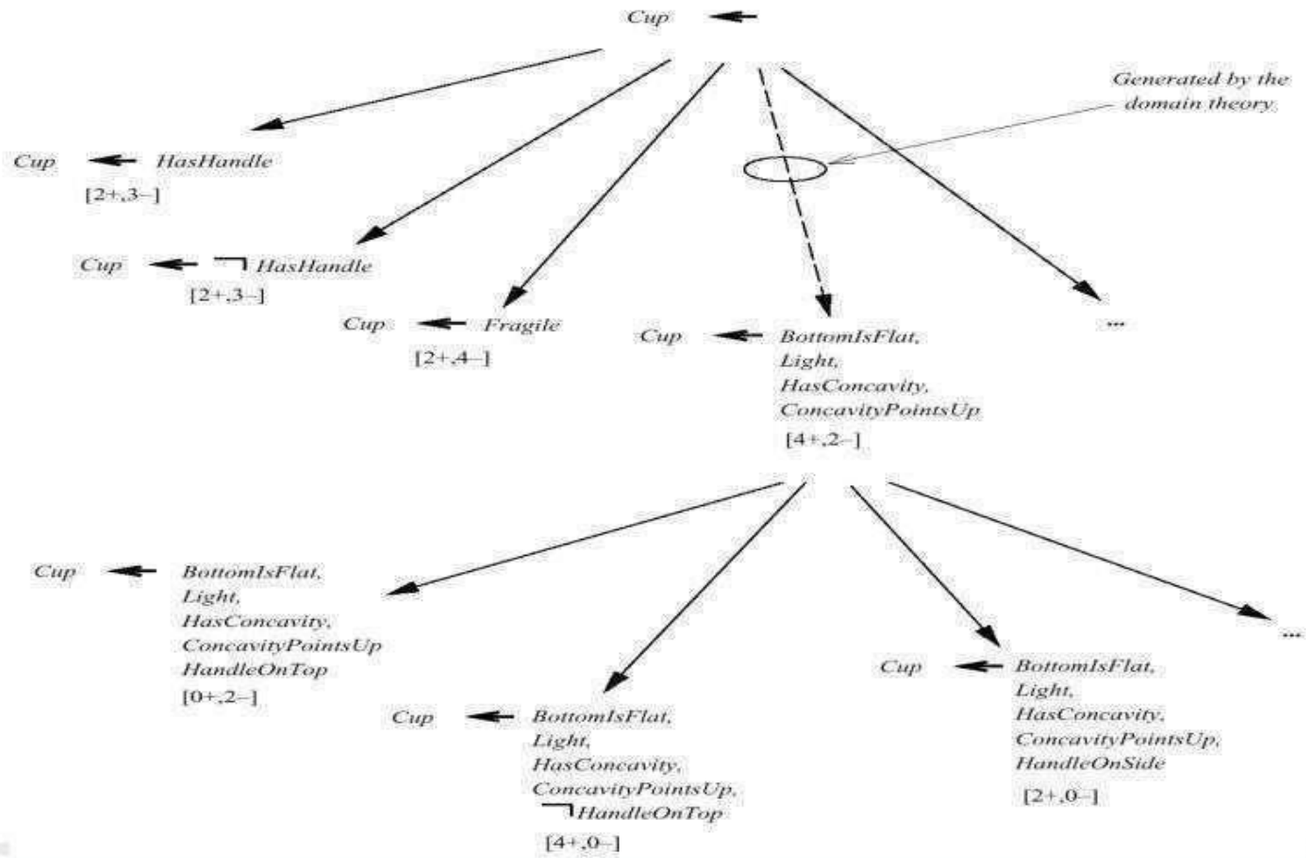
- $f(x)$ is target function
- $\hat{f}(x)$ is neural net approximation to $f(x)$
- $A(x)$ is domain theory approximation to $f(x)$


f(x)
f(x1)
f(x2)
f(x3)
f
x1
x2
x3
x
g
x
h
x

Hypothesis Space Search in EBNN



Search in FOCL



THANKYOU