LECTURE 5:THINKING ABOUT QUANTIFIERS

When the domain of discourse U is finite, think of Quantification as looping through the elements of the domain

To evaluate $\forall x P(x)$- loop through all x in the domain:

If at every step P(x) is TRUE then $\forall x$ P(x) is TRUE

If at a step P(x) is FALSE then $\forall x$ P(x) and the loop terminates.`

EVALUATING EXISTENTIALLY QUANTIFIER

To evaluate $\exists x P(x)$-loop through all x in the domain. If at some step , P(x) is TRUE then $\exists x P(x)$ is TRUE and the loop terminates.

If the loop ends without finding an x for which is TRUE then $\exists x P(x)$ is FALSE.

Even if the domains are infinite still think about quantifiers as above , but the loops will not terminate in some cases.

Precedence of Quantifiers

Operator	Precedence
∀∃	1
¬	2
∩	3
∨	4
→	5
≡(↔)	6

Example

Consider the following statements:

“All human beings are mortal”

“Sachin is a human being”

Does it follow that Sachin is mortal?

Solution

Let H(x): “x is a human being”

Let M(x):” x is mortal” : The domain of discourse U is all human beings

Example (cont)

All human beings are mortal translates:

$\forall x(H(x)\rightarrow M(x))$

Sachin is a human being translates to:

H(Sachin)

Therefore H(Sachin)→M(x)

Sachin is a human being translates to: H(Sachin)

Therefore H(Sachin)→M(Sachin)

Equivalences in Predicate logic

Statements involving predicates and quantifiers are logically equivalent if and only if they have the same truth value

–for every predicate substituted into these statements and

–for every domain of discourse used for the variables in the expressions.

Equivalences in Predicate logic

The notation $S \equiv T$ indicates that S and T are logically equivalent.

Example: $\forall x\ \neg\neg S(x) \equiv \forall x\ S(x)$

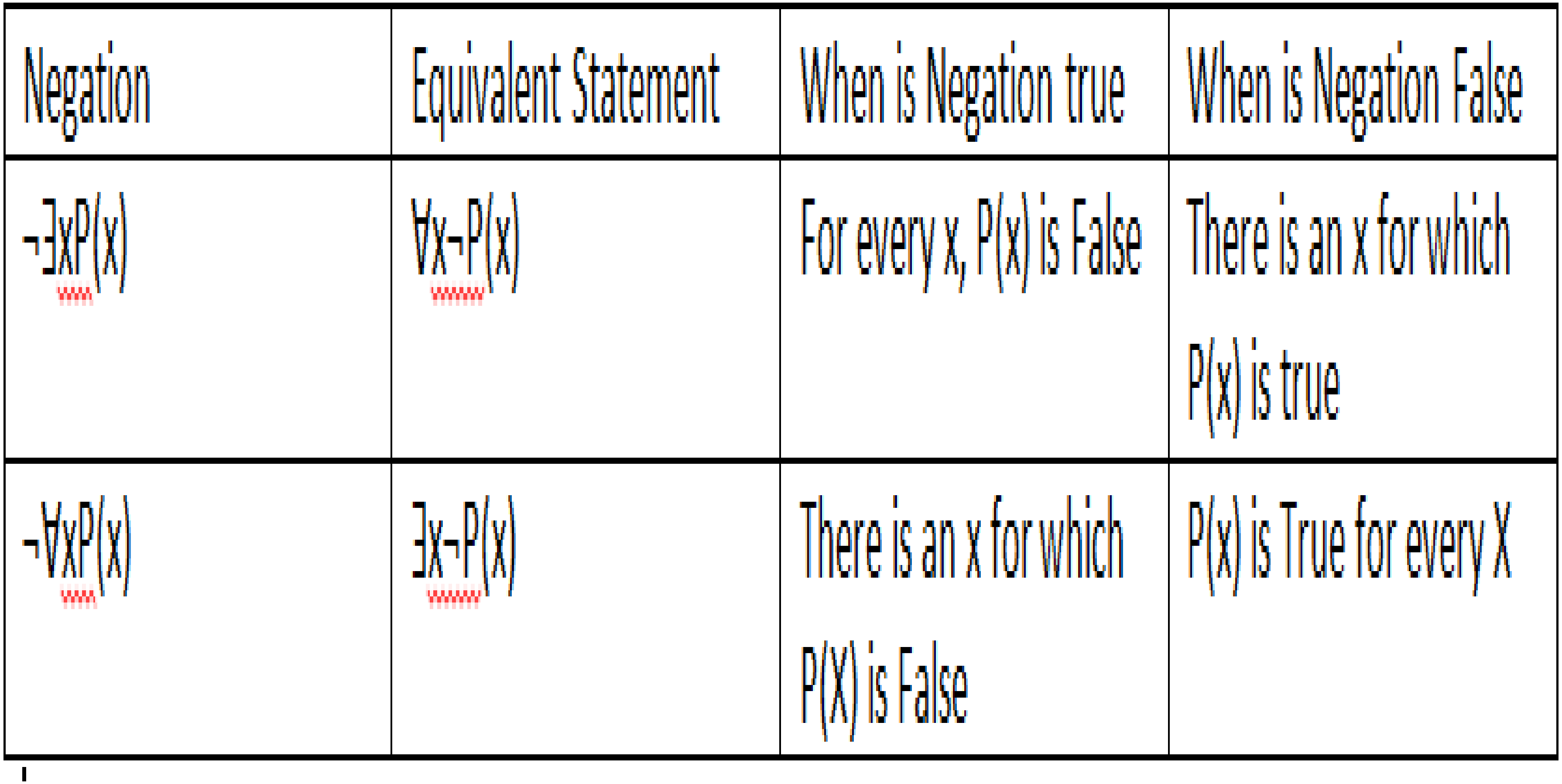
$\neg \exists x P(X) \equiv \forall x \neg P(X)$

$\neg \forall x P(x) \equiv \exists x \neg P(x)$

$\exists x P(x) \equiv \exists Y P(Y)$

$\forall x q(x) \equiv \forall Y q(Y)$

De Morgan’s Laws for Quantifiers



Negation	Equivalent Statement	When is Negation true	When is Negation False
¬∃xP(x)	∀x¬P(x)	For every x, P(x) is False	There is an x for which P(x) is true
¬∀xP(x)	∃x¬P(x)	There is an x for which P(X) is False	P(x) is True for every X

Example

“There is an honest politician”

Let H(x):”x is honest”, U consists of all politicians

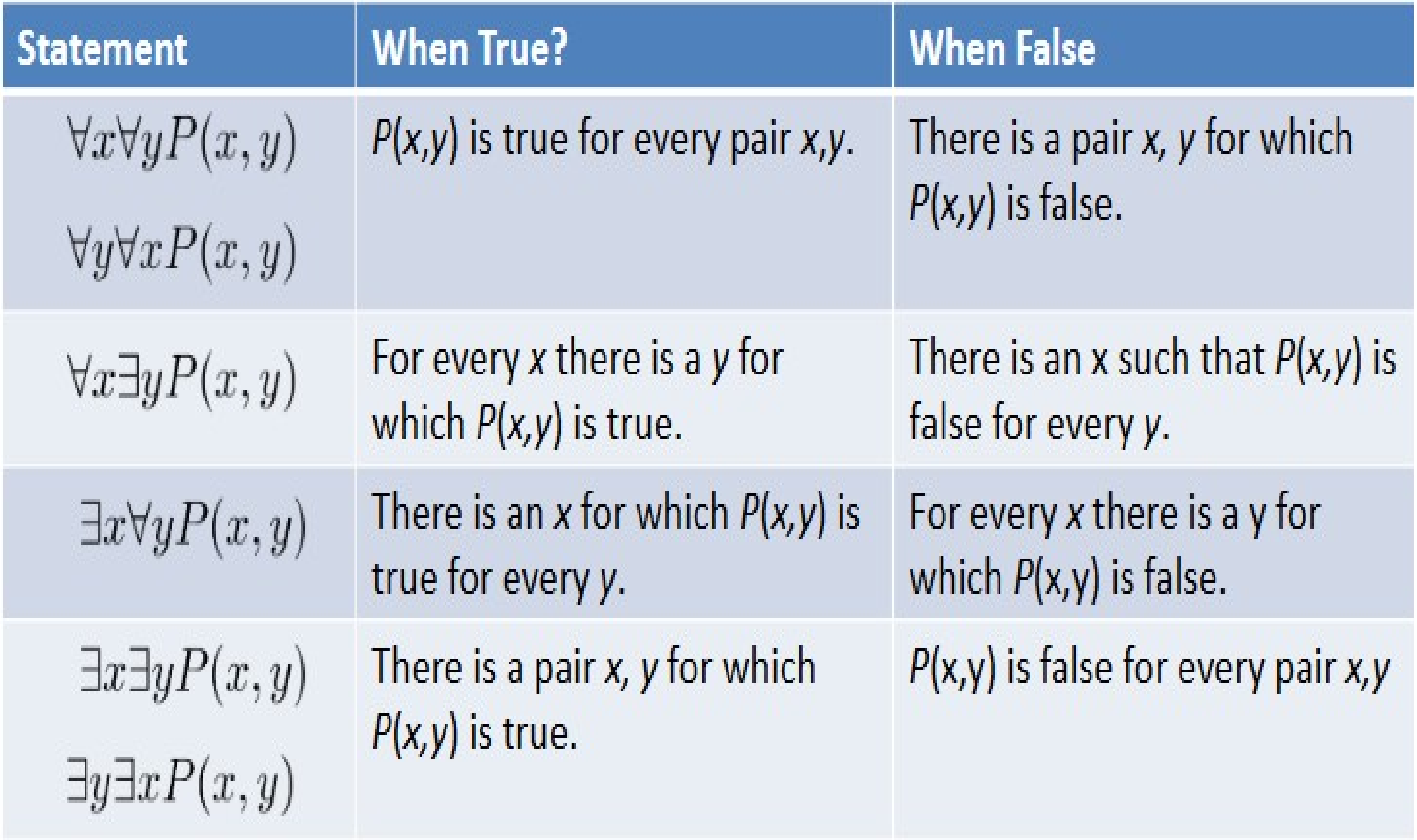
Then $\exists x H(x)$

“There does not exist an honest politician”

$\neg\exists x H(x)$

$\neg\exists x H(x) \equiv \forall x \neg H(x)$

Quantification of two variables



Statement	When True?	When False
$\forall x \forall y P(x,y)$ $\forall y \forall x P(x,y)$	*P*(*x*,*y*) is true for every pair *x*,*y*.	There is a pair *x*, *y* for which *P*(*x*,*y*) is false.
$\forall x \exists y P(x,y)$	For every *x* there is a *y* for which *P*(*x*,*y*) is true.	There is an x such that *P*(*x*,*y*) is false for every *y*.
$\exists x \forall y P(x,y)$	There is an *x* for which *P*(*x*,*y*) is true for every *y*.	For every *x* there is a y for which *P*(x,y) is false.
$\exists x \exists y P(x,y)$ $\exists y \exists x P(x,y)$	There is a pair *x*, *y* for which *P*(*x*,*y*) is true.	*P*(x,y) is false for every pair *x*,*y*

Representing English statements in predicate calculus

Many grammatically correct English sentences can be represented in first order predicate calculus using symbols, connectives and variable symbols.

Important to note:

- There is no unique mapping of sentences into predicate calculus expressions, e.g. an English expression many have any number of different predicate calculus representations

Representing English statements in predicate calculus

- Challenge is for AI programmers to find a scheme for using these predicates that optimizes the expressiveness and efficiency of the resulting representation

Examples of English Statements/sentences represented in predicate calculus:

Examples of English sentences represented in predicate calculus:

1)If it doesn’t rain on Monday, Tom will go to the mountains

$\neg\text{weather}(\text{rain},\text{monday})\rightarrow\text{go}(\text{tom},\text{mountains})$

2)All basketball players are tall

$\forall X(\text{basketball\_player}(x)\rightarrow\text{tall}(x))$

3)Some people like lemons

$\exists X(\text{person}(x)\cap\text{likes}(x,\text{lemons}))$

Examples of English sentences represented in predicate calculus:

4)If wishes were horses beggars would ride

$\text{equal}(\text{wishes},\text{horses}) \rightarrow \text{ride}(\text{beggars})$

5)Nobody likes taxes

$\neg \exists X\ \text{likes}(X,\text{taxes})$

6)Emma is a Doberman and a good dog

$\text{good dog}(\text{emma}) \cap \text{isa}(\text{emma},\text{doberman})$

Examples of English sentences represented in predicate calculus

7)Brothers are siblings

$\forall X \forall Y(\mathrm{brothers}(X,Y) \rightarrow S(X,Y)$

8)Singlehood is symmetric

$\forall X \forall Y(\mathrm{singlehood}(Y,X))$

9) Everybody loves somebody

$\forall X \exists Y \mathrm{loves}(X,Y)$

10)Everyone loves himself

$\forall X \mathrm{loves}(X,X)$

Definition and Types of Inference Rules

Hypothesis means to create valid argument

Argument- is sequence of propositions/premises such that($p1 \cap p2 \cap p3 .. \cap p_n) \rightarrow q$.(q is the conclusion)

Valid argument - is the premises that imply q the conclusion

Inference – driving conclusion from given premises.

Inference Rules

Consider the statements:

If it is raining,(P) I will need an Umbrella(Q)

It is raining. (P)

∴ I will need an umbrella(Q)

This can be written in logic as:

$P \rightarrow Q$

P

∴ Q

1)Modus Ponens

P → Q   If it rains I will need an umbrella R→U

P   It is raining   R

∴ Q   Therefore I need an umbrella   ∴ U

T**autology**:(P∩(P→Q))→Q

Tautology is a formula or assertion that is correct in every possible interpretation

2)Modus Tollens

P→Q

¬Q

∴ ¬P

R→U

¬U

∴ ¬P