Rationale of rendering quantifiers as modal operators © 2016 by Colin James III All rights reserved.

Definition	Axiom	Symbol	Name	Meaning	2-tuple	Ordinal
1	p=p	T	Tautology	proof	11	3
2	p@p	F	Contradiction	absurdum	00	0
3	%p>#p	N	Non-contingency	truth	01	1
4	%p<#p	C	Contingency	falsity	10	2

Numbered definitions of axioms with symbol, name, meaning, 2-tuple, and ordinal values. The designated proof value is T tautology. Note the meaning of (%p>#p): a possibility of p implies the necessity of p; and some p implies all p. In other words, if a possibility of p then the necessity of p; and if some p then all p.

Note: To avoid confusion with logical jargon, "nvt" means "not validated as tautologous"

The rationale for rendering quantifiers as modal operators in Meth8 has arguments from satisfiability (contra Kuhn) and reproducability of invalidating and validating syllogisms.

1. Satisfiability

From Steven T Kuhn (1979), "Quantifiers as modal operators", Studia Logica 39, 2-3/80, page 147:

> "Either [with Montague's approach as first order models or with Prior's approach as "sequences of individuals"], there is a problem. The atomic formulas of predicate logic cannot all be treated as atoms in the modal language. If we regard *Pxy* and *Pyx*, for example, as distinct sentence letters of the modal language then ∃*x* ∃*y Pxy* & −∃***x*** ∃***y*** *Pyx* will be satisfiable. If we regard them as identical sentence letters then ∃*x* ∃*y* (*Pxy* & −*Pyx)* will be unsatisfiable."

If Pxy and Pyx are distinct sentence letters of the modal language, then this is "satisfiable" as:

((%x&%y)&(p&(x&y))) & ~((%x&%y)&(p&(y&x))) ; nvt; and False; (1.1)

If Pxy and Pyx are identical sentence letters of the modal language, then this is "unsatisfiable" as:

(%x&%y)&(((p&(y&x))&~ ((p&(x&y))); nvt; and False; (1.2)

We ask if Eq 1.1 and Eq1.2 are equivalent as:

(((%x&%y)&(p&(x&y))) & ~((%x&%y)&(p&(y&x)))) =
((%x&%y)&((p&(y&x))&~(p&(x&y)))) ; vt; (1.3)

This means rendition of the quantifiers to modal operators in Meth8 is satisfiable, and hence correct.

What follows is that there is no reason to rely on "the variable-free formulations of logic by Tarski, Bernays, Halmos, Nolin and Quine ... [for] the effect of arbitrary permutations and identifications of the variables occurring in a formula."

2. Reproducability of 24 syllogisms deemed valid in predicate logic

The Square of Opposition (original) produced four combinations for each corner A, I, E, O for 4 ^ 4 = 256 syllogisms. Medieval scholars determined 24 of the 256 syllogisms were valid deductions. Of those, 9 were made valid but only after additional *known* assumptions were applied as fix ups. Meth8 Tautologous none of the 24 syllogisms *before* fix ups. Meth8 also *discovered* correct additional assumptions to render the other 15 syllogisms Tautologous. The fix ups in bold were verified independently by Prover9 (2007). The syllogisms fall into six groups of truth table values *before* fix ups and sorted nearest to the state of Tautologous in Table 2.1.

LET: p x, q F, r G, s H, ~ Not, # Necessity (all), % Possibility (exists), & And, > Imply,
vt Tautologous, nvt Not Tautologous * known fixes (9 of 24 syllogisms)

The expressions for the syllogisms below are derived using functions FGH as qrs for instances of the variable x as p and with the > Imply connective between functions.

<u>Syllogism number</u> <u>Fix up code in bold</u>

```
 II AEE (((#p&((q&p)>( s&p)))&(#p&((r&p)>(~s&p))))&(%p&(r&p)))>(#p&((r&p)&(~q&p)))
 II AEO*(((#p&((q&p)>( s&p)))&(#p&((r&p)>(~s&p))))&(%p&(r&p)))>(%p&((r&p)&(~q&p)))
 II AOO (((#p&((q&p)>( s&p)))&(%p&((r&p)>(~s&p))))&(%p&(r&p)))>(%p&((r&p)&(~q&p)))
 IV AEE (((#p&((q&p)>( s&p)))&(#p&((s&p)>(~r&p))))&(%p&(r&p)))>(#p&((r&p)&(~q&p)))
 IV AEO*(((#p&((q&p)>( s&p)))&(#p&((s&p)>(~r&p))))&(%p&(r&p)))>(%p&((r&p)&(~q&p)))
  TCTTTTTTTCTCTTTT.EUEEEEEEEUEUEEEE.EEEEEEEEEEEEEEEE.EPEEEEEEEPEPEEEE.EIEEEEEEEIEIEEEE
  Model 1.         Model 2.1        Model 2.2        Model 2.3.1      Model 2.3.2

 IV AAI*(((#p&((q&p)>( s&p)))&(#p&((s&p)>( r&p))))&(%p&(q&p)))>(%p&((r&p)&( q&p)))
 IV IAI (((%p&((q&p)>( s&p)))&(#p&((s&p)>( r&p))))&(%p&(q&p)))>(%p&((r&p)&( q&p)))
  TCTTTCTTTTTTTCTT.EUEEEUEEEEEEEUEE.EEEEEEEEEEEEEEEE.EPEEEPEEEEEEEPEE.EIEEEIEEEEEEEIEE

  I AAA (((#p&((s&p)>( q&p)))&(#p&((r&p)>( s&p))))&(%p&(r&p)))>(#p&((r&p)&( q&p)))
  I AAI*(((#p&((s&p)>( q&p)))&(#p&((r&p)>( s&p))))&(%p&(r&p)))>(%p&((r&p)&( q&p)))
  I AII (((#p&((s&p)>( q&p)))&(%p&((r&p)>( s&p))))&(%p&(r&p)))>(%p&((r&p)&( q&p)))
  TCTCTTTTTTTCTTTT.EUEUEEEEEEEUEEEE.EEEEEEEEEEEEEEEE.EPEPEEEEEEEPEEEE.EIEIEEEEEEEIEEEE

  I EAE (((#p&((s&p)>(~q&p)))&(#p&((r&p)>( s&p))))&(%p&(r&p)))>(#p&((r&p)&(~q&p)))
  I EAO*(((#p&((s&p)>(~q&p)))&(#p&((r&p)>( s&p))))&(%p&(r&p)))>(%p&((r&p)&(~q&p)))
  I EIO (((#p&((s&p)>(~q&p)))&(%p&((r&p)>( s&p))))&(%p&(r&p)))>(%p&((r&p)&(~q&p)))
 II EAE (((#p&((q&p)>(~s&p)))&(#p&((r&p)>( s&p))))&(%p&(r&p)))>(#p&((r&p)&(~q&p)))
 II EAO*(((#p&((q&p)>(~s&p)))&(#p&((r&p)>( s&p))))&(%p&(r&p)))>(%p&((r&p)&(~q&p)))
 II EIO (((#p&((q&p)>(~s&p)))&(%p&((r&p)>( s&p))))&(%p&(r&p)))>(%p&((r&p)&(~q&p)))
  TCTCTTTTTCTTTTTT.EUEUEEEEEUEEEEEE.EEEEEEEEEEEEEEEE.EPEPEEEEEPEEEEEE.EIEIEEEEEIEEEEEE

III EAO*(((#p&((s&p)>(~q&p)))&(#p&((s&p)>( r&p))))&(%p&(s&p)))>(%p&((r&p)&(~q&p)))
III EIO (((#p&((s&p)>(~q&p)))&(%p&((s&p)>( r&p))))&(%p&(s&p)))>(%p&((r&p)&(~q&p)))
III OAO (((%p&((s&p)>(~q&p)))&(#p&((s&p)>( r&p))))&(%p&(s&p)))>(%p&((r&p)&(~q&p)))
 IV EAO*(((#p&((q&p)>(~s&p)))&(#p&((s&p)>( r&p))))&(%p&(s&p)))>(%p&((r&p)&(~q&p)))
 IV EIO (((#p&((q&p)>(~s&p)))&(%p&((s&p)>( r&p))))&(%p&(s&p)))>(%p&((r&p)&(~q&p)))
  TCTCTTTCTTTTTTTT.EUEUEEEUEEEEEEEE.EEEEEEEEEEEEEEEE.EPEPEEEPEEEEEEEE.EIEIEEEIEEEEEEEE

III AAI*(((#p&((s&p)>( q&p)))&(#p&((s&p)>( r&p))))&(%p&(s&p)))>(%p&((r&p)&( q&p)))
III AII (((#p&((s&p)>( q&p)))&(%p&((s&p)>( r&p))))&(%p&(s&p)))>(%p&((r&p)&( q&p)))
III IAI (((%p&((s&p)>( q&p)))&(#p&((s&p)>( r&p))))&(%p&(s&p)))>(%p&((r&p)&( q&p)))
  TCTCTCTTTTTTTTTT.EUEUEUEEEEEEEEEE.EEEEEEEEEEEEEEEE.EPEPEPEEEEEEEEEE.EIEIEIEEEEEEEEEE
  Model 1.         Model 2.1        Model 2.2        Model 2.3.1      Model 2.3.2
```

Table 2.1. 24 syllogisms as based on the Square of Opposition, in Meth8 script

Because the 24 syllogisms contain one variable p, they may be reduced in size by removing redundant occurrences of p from Table 2.1. For example the stepped process to do this is presented for II. AAO, of the 15 valid syllogisms and with an additional assumption.

```
 II AOO (((#p&((q&p)>( s&p)))&(%p&((r&p)>(~s&p))))&(%p&(r&p)))>(%p&((r&p)&(~q&p)))
 Steps   ((#p&((q&p)>( s&p)))&(%p&((r&p)>(~s&p))))            >(%p&((r&p)&(~q&p)))
    1:   ((#p&( q   >  s   ))&(%p&( r   > ~s   )))            >(%p&( r   & ~q   ))
    2:   ((p&#( q   >  s   ))&(p&%( r   > ~s   )))            >(p&%( r   & ~q   ))
    3:  (((p&#( q   >  s   ))&(p&%( r   > ~s   )))&(%p&(r&p)))>(p&%( r   & ~q   ))
    4:  (((p&#( q   >  s   ))&(p&%( r   > ~s   )))&(p&% r   ))>(p&%( r   & ~q   ))
    5:  (((p&#( q   >  s   ))&(p&%( r   > ~s   )))&(p&% r   ))>(p&%( r   & ~q   ))
    6:  ((p&(#( q   >  s   ) &   %( r   > ~s   )))&   % r    )>(p&%( r   & ~q   ))
```

The reduced expression in Step 6 as ((p&(#(q>s)&%(r>~s)))&%r)>(p&%(r&~q)) represents a 50% reduction in the number of characters from the original expression in the Meth8 script. Table 2.1 is entirely rewritten in this way as Table 2.2.

```
Syllogism                                                   Fix up
number                                                      code bold

 II AEE ((p&(#( q   >  s   ) &   #( r   > ~s   )))&   % r    )>(p&#( r   & ~q   ))
 II AEO*((p&(#( q   >  s   ) &   #( r   > ~s   )))&   % r    )>(p&%( r   & ~q   ))
 II AOO ((p&(#( q   >  s   ) &   %( r   > ~s   )))&   % r    )>(p&%( r   & ~q   ))
 IV AEE ((p&(#( q   >  s   ) &   #( s   > ~r   )))&   % r    )>(p&#( r   & ~q   ))
 IV AEO*((p&(#( q   >  s   ) &   #( s   > ~r   )))&   % r    )>(p&%( r   & ~q   ))

 IV AAI*((p&(#( q   >  s   ) &   #( s   >  r   )))&   % r    )>(p&%( r   &  q   ))
 IV IAI ((p&(%( q   >  s   ) &   #( s   >  r   )))&   % r    )>(p&%( r   &  q   ))

  I AAA ((p&(#( s   >  q   ) &   #( r   >  s   )))&   % r    )>(p&#( r   &  q   ))
  I AAI*((p&(#( s   >  q   ) &   #( r   >  s   )))&   % r    )>(p&%( r   &  q   ))
  I AII ((p&(#( s   >  q   ) &   %( r   >  s   )))&   % r    )>(p&%( r   &  q   ))

  I EAE ((p&(#( s   > ~q   ) &   #( r   >  s   )))&   % r    )>(p&#( r   & ~q   ))
  I EAO*((p&(#( s   > ~q   ) &   #( r   >  s   )))&   % r    )>(p&%( r   & ~q   ))
  I EIO ((p&(#( s   > ~q   ) &   %( r   >  s   )))&   % r    )>(p&%( r   & ~q   ))
 II EAE ((p&(#( q   > ~s   ) &   #( r   >  s   )))&   % r    )>(p&#( r   & ~q   ))
 II EAO*((p&(#( q   > ~s   ) &   #( r   >  s   )))&   % r    )>(p&%( r   & ~q   ))
 II EIO ((p&(#( q   > ~s   ) &   %( r   >  s   )))&   % r    )>(p&%( r   & ~q   ))

III EAO*((p&(#( s   > ~q   ) &   #( s   >  r   )))&   % r    )>(p&%( r   & ~q   ))
III EIO ((p&(#( s   > ~q   ) &   %( s   >  r   )))&   % r    )>(p&%( r   & ~q   ))
III OAO ((p&(%( s   > ~q   ) &   #( s   >  r   )))&   % r    )>(p&%( r   & ~q   ))
 IV EAO*((p&(#( q   > ~s   ) &   #( s   >  r   )))&   % r    )>(p&%( r   & ~q   ))
 IV EIO ((p&(#( q   > ~s   ) &   %( s   >  r   )))&   % r    )>(p&%( r   & ~q   ))

III AAI*((p&(#( s   >  q   ) &   #( s   >  r   )))&   % r    )>(p&%( r   &  q   ))
III AII ((p&(#( s   >  q   ) &   %( s   >  r   )))&   % r    )>(p&%( r   &  q   ))
III IAI ((p&(%( s   >  q   ) &   #( s   >  r   )))&   % r    )>(p&%( r   &  q   ))
```

Table 2.2 24 syllogisms as based on the Square of Opposition, in minimal Meth8 format

Meth8 demonstrates correct replication of results from the syllogisms in this limited fragment on predicate logic. Meth8 is fully capable of fixing syllogisms deemed valid by predicate logic, and in a minimal format.

3. Pattern steps

Patterns were discovered in fix ups for syllogisms from the original Square of Opposition in the Figure for AEIO for the 24 syllogisms accepted as valid. The Meth8 truth tables of the 24 syllogisms are sorted in Table 2.2 and collated here by Groups.

LET: a *n*, c *n* for antecedent, consequent
in assumption *n* = 1, 2, additional assumption *n* = 3, conclusion *n* = 4

Group	Figure	AEIO combo	Assumption 1	Assumption 2	Additional 3	Conclusion 4
1	II	AEE = AEO* = AOO	q > s	r > ~s	r	r & ~q
1	IV	AEE = AEO*	q > s	s > ~r	r	r & ~q
2	IV	AAI* = IAI	q > s	s > r	q	r & q
3	I	AAAA = AAI* = AII	s > q	r > s	r	r & q
4	I	EAE = EAO* = EIO	s > ~q	r > s	r	r & ~q
4	II	EAE = EAO* = EIO	q > ~s	r > s	r	r & ~q
5	III	EAO* = EIO = OAO	s > ~q	s > r	s	r & ~q
5	IV	EAO* = EIO	q > ~s	s > r	s	r & ~q
6	III	AAI* = AII = IAI	s > q	s > r	s	r & q

Table 3. Patterns of assumptions and conclusions

The format of the syllogisms is with placeholders:

(a1 > c1) & (a2 > c2) [& a3] = (a4 & c4). (3.1)

Because of the main & connective in Eq 3.1 the main literal groups may be reversed. In that case the placeholders remain in the same named order as above, that is, with the antecedent group (a1 > c1).

These rules in pseudo code produce the a3 results for the column Additional 3 above:

```
Step 1:     LET a2 = [assigned]
Step 2:     IF a2 = (a4 OR c4)  THEN
                LET a3 = a2 ! (Group 1, Figure II; Group 3, Figure 1; Group 4, Figures I, II)
Step 3:     ELSE IF a2 = a1 THEN
                LET a3 = a1 ! (Group 2, Figure IV; Group 3, Figure IV)
Step 4:     ELSE IF a2 = c1 THEN
Step 5:         IF c2 = Negated_function THEN
                    LET a3 = Non_negated_function ( c2)  ! (Group 1, Figure IV)
Step 6:         ELSE
                    LET a3 = a1 ! (Group 5, Figure III; Group 6, Figure III)
                END IF
            END IF
```

4. Test of two syllogisms in Meth8

We next test two expressions formatted as syllogisms, manufactured from I and O as IOE and OIA. We use the same technique for the 9 syllogisms above to supply an additional assumption as a fix up.

Example 4.1: IOE

LET	Assumption 1:	I	p&%(s>q)
	Assumption 2:	O	p&%(r>~s)
	Assumption 3:		[*To be determined below.*]
	Conclusion 4:	E	p&#(r&q)

((1) & (2))>(3): (p&(%(s>q)&%(r>~s)))>(p&#(r&~q)) ; nvt (4.1.1)

We build the additional assumption by the rules.

Step 1: a2 = r
Step 2: a3 = r
Assumption 3: p&%r

For: ((p&(%(s>q)&%(r>~s)))&%r)>(p&#(r&~q)) ; nvt (4.1.2)

We test Eq 4.1.2 independently in Prover9 (2007).

Assumption 1:	exists x (H(x) -> F(x)).	
Assumption 2:	exists x (G(x) -> -H(x)).	
Assumption 3:	exists x (G(x)).	
Conclusion 4:	all x (G(x) & -F(x)).	(4.1.3)
Result:	False	

In Example 4.1 Meth8 and Prover9 produce the result of not Tautologous.

Example 4.2: OIA

LET	Assumption 1:	O	p&%(s>~q)
	Assumption 2:	I	p&%(r>s)
	Assumption 3:		[*To be determined below.*]
	Conclusion 4:	A	p&#(q&r)

((1) & (2))>(3): (p&(%(s>~q)&%(r>s)))>(p&#(q&r)) ; nvt (4.2.1)

We build the additional assumption by the rules.

Step 1: a2 = r
Step 2: a3 = r
Assumption 3: p&%r

For: ((p&(%(s>~q)&%(r>s)))&%r)>(p&#(q&r)) ; nvt (4.2.2)

We test Eq 4.2.1 independently in Prover9 (2007).

Assumption 1: exists x (H(x) -> -F(x)).

Assumption 2: exists x (G(x) -> H(x)).
Assumption 3: exists x (G(x)).
Conclusion 4: all x (F(x) & G(x)). (4.2.3)
Result: False

In Example 4.2 Meth8 and Prover9 produce the result of not Tautologous.

5. Tests of syllogistic fallacies

See links from: en.wikipedia.org/wiki/Syllogism

5.1 Undistributed middle

Neither of the premises accounts for all members of the middle term, which consequently fails to link the major and minor term: All C is B. A is B. Therefore, C is A.

LET: q r s, A B C; "is" > Imply, or "is" & And

((#s>r)&(q>r))>(s>q) ; nvt (5.1.1)
((#s&r)&(q&r))>(s&q) ; vt (5.1.2)

Eq 5.1.2 means that the & And connective as the verb "is" does not represent the true state of affairs.

Eq 5.1.1 correctly renders the > Imply connective as the verb "is" because Eq 5.1.1 returns the correct result of a fallacy as not Tautologous.

5.2 Illicit treatment of the major term

From: en.wikipedia.org/wiki/Illicit_major

"Illicit major" is a categorical syllogism that is invalid because its major term is undistributed in the major premise but distributed in the conclusion.

This fallacy has the following argument form: All A are B. No C are A. Therefore, no C are B. In words: All horses have hooves. No humans are horses. Therefore, no humans have hooves.

LET: q r s, A B C, horses hooves humans; "are" & And, or "are" > Imply

((#q>r)&(~s>q))>(~s>r) ; nvt (5.2.1)
((#q&r)&(~s&q))>(~s&r) ; vt (5.2.2)

This means the verb "are" is correctly rendered by the connective > Imply for the correct result of Eq 5.2.1, namely, that the expression is a fallacy as not Tautologous.

Modus Camestros is stated to be a valid syllogism, and not a fallacy, as: All A are B. No C are B. Therefore, no C are A. In words: All horses have hooves. No humans have hooves. Therefore, no humans are horses.

((q>r)&(~s>r))>(~s>q) ; nvt (5.2.3)
((q&r)&(~s&r))>(~s&q) ; vt (5.2.4)

However, by the same measure for the assignment of the verb "are" to the connective > Imply, modus

Camestros returns a mistaken result in Eq 5.2.3, namely, that the expression is not a valid syllogism as not Tautologous. This means that modus Camestros is arguably a fallacy itself.

This leads us to the conclusion that in Meth8 script the correct mapping of the verb "to be" in syllogisms is the connective > Imply, and not the connective & And as mistakenly used.

5.3 Illicit treatment of the minor term

From: en.wikipedia.org/wiki/Illicit_minor

"Illicit minor" is committed in a categorical syllogism that is invalid because its minor term is undistributed in the minor premise but distributed in the conclusion.

For example: Donuts are good. Donuts are unhealthy. Thus, all good is unhealthy.

All A are B. All A are C. Therefore, all C are B.

LET: q r s, A B C

((#q>r)&(#q>s))>(#s>r) ; nvt (5.3.1)

5.4 Exclusive premises

From: en.wikipedia.org/wiki/Fallacy_of_exclusive_premises

Both premises are negative, meaning no link is established between the major and minor terms:

E: No cats are dogs.
O: Some dogs are not pets.
O: Therefore, some pets are not cats.

E: No planets are dogs.
O: Some dogs are not pets.
O: Therefore, some pets are not planets.

LET: q cats / planets, r dogs, s pets

((~q>r)&(%r>~s))>(%s>~q) ; nvt (5.4.1)

5.5 Negative conclusion from affirmative premises

If both premises are affirmative, the conclusion must also be affirmative. A negative conclusion from affirmative premises is a fallacy when a categorical syllogism has a negative conclusion yet both premises are affirmative. The inability of affirmative premises to reach a negative conclusion a basic rule of constructing a valid categorical syllogism.

Exactly one of the premises must be negative to construct a valid syllogism with a negative conclusion. (A syllogism with two negative premises commits the related fallacy of exclusive premises.)

Example of invalid AAE form: All A is B. All B is C. Therefore, no A is C.

LET: q A, r B, s C

((#q>r)&(#r>s))>(~q>s) ; nvt (5.5.1)

Example of invalid IV. AAO form: All A is B. All B is C. Therefore, some C is not A.

((#q>r)&(#r>s))>(%s>~q) ; nvt (5.5.2)

"This is valid only if A is a proper subset of B and/or B is a proper subset of C."

We write this additional assumption as:

((((q<r)+(r<s))+((q<r)&(r<s)))&((#q>r)&(#r>s)))>(%s>~q) ; nvt (5.5.3)

TTNNTTNNTTNNTTTT.EEEEEEEEEEEEEEEE.EEUUEEUUEEUUEEEE.EEIIEEIIEEIIEEEE.EEPPEEPPEEPPEEEE

The quoted assertion is mistaken according to Meth8.

However, this argument reaches a faulty conclusion if A, B, and C are equivalent. In the case that A=B=C, the conclusion of the following simple I. AAA syllogism would contradict the IV. AAO argument above: All B is A. All C is B. Therefore, all C is A.

((#r>q)&(#s>r))>(#s>q) ; vt (5.5.4)

5.6 Affirmative conclusion from a negative premise

From: en.wikipedia.org/wiki/Affirmative_conclusion_from_a_negative_premise

The "illicit negative" is a formal fallacy that is committed when a categorical syllogism has a positive conclusion, but one or two negative premises.

For example: No fish are dogs, and no dogs can fly, therefore all fish can fly.

LET: q dogs, r fish, s fly, p things

((~r>q)&(~q>s))>(#r>s) ; nvt (5.6.1)

"The only thing that can be properly inferred from these premises is that some things that are not fish cannot fly, provided that dogs exist."

The quoted assertion above using "some things" is mistaken and not Tautologous by Meth8:

((~r>q)&(~q>s))>(%q>((%p>~r)>~s)) ; nvt (5.6.2)

TTTTTTTTTTFFTTCT.EEEEEEEEEEUUEEUE.EEEEEEEEEEUUEEEE.EEEEEEEEEEUUEEPE.EEEEEEEEEEUUEEIE

"This could be illustrated mathematically as: If $A \cap B = \varnothing$ and $B \cap C = \varnothing$ then $A \subset C$." (5.6.3)
(Because we dispense with the axiom of the empty set elsewhere, the set expression of Eq 5.6.3 is not evaluated.)

It is a fallacy because any valid forms of categorical syllogism that assert a negative premise must have a negative conclusion.

5.7 Existential fallacy

From: en.wikipedia.org/wiki/Existential_fallacy

In the existential fallacy, *we presuppose that a class has members* when we are not supposed to do so; that is, when we should not assume existential import.

Every *C is* B. Every *C is* A. So, some *A* is *B*.

((#s>r)&(#s>q))>(%q> r) ; nvt (5.7.1)

No *C* is *B*. Every *A* is *C*. So, some *A* is not *B*.

((~s>r)&(#q>s))>(%q>~r) ; nvt (5.7.2)

6. The 24 syllogisms derived by the & And connective

From: en.wikipedia.org/wiki/Syllogism

LET: q r s, M P S; # All, % Some; vt Tautologous, nvt Not Tautologous

In Table 6.1 we map the syllogisms by the & And connective for variables MPS, instead of by the > Imply connective for functions in section 2 above. The expressions below have about 20% fewer characters than those in Table 2.2.

Code	**Name**	**Assumptions 1, 2**	**Assumption 3**	**Conclusion**	**Test**	**Comments**
AAA-1	Modus Barbara	((#q&r)&(#s&q))		>(#s&r)	vt	
AAI-1	Modus Barbari	(((#q&r)&(#s&q))	&%s)	>(%s&r)	vt	* not needed
		((#q&r)&(#s&q))		>(%s&r)	vt	
AAI-4	Modus Bamalip	(((#r&q)&(#q&s))	&%r	>(%s&r)	vt	* not needed
		((#r&q)&(#q&s))		>(%s&r)	vt	
EAE-1	Modus Celarent	((~q&r)&(#s&q))		>(~s&r)	vt	
EAE-2	Modus Cesare	((~r&q)&(#s&q))		>(~s&r)	nvt	* Mistake
		(((~r&q)&(#s&q))	&%r)	>(~s&r)	vt	* Meth8 fix
AEE-2	Modus Camestres	((#r&q)&(~s&q))		>(~s&r)	vt	
AEE-4	Modus Calemes	((#r&q)&(~q&s))		>(~s&r)	vt	
EAO-1	Modus Celaront	(((~q&r)&(#s&q))	&%s)	>(~s&r)	vt	* not needed
		((~q&r)&(#s&q))		>(~s&r)	vt	
EAO-2	Modus Cesaro	(((~r&q)&(#s&q))	&%s)	>(%s&~r)	vt	* not needed
		((~r&q)&(#s&q))		>(%s&~r)	vt	
AEO-2	Modus Camestros	(((#r&q)&(~s&q))	&%s)	>(%s&~r)	vt	* needed
		((#r&q)&(~s&q))		>(%s&~r)	nvt	*
AEO-4	Modus Calemos	(((#r&q)&(~q&s))	&%s)	>(%s&~r)	vt	* not needed
		((#r&q)&(~q&s))		>(%s&~r)	vt	

Code	Name	Assumptions 1, 2	Assumption 3	Conclusion	Test	Comments
AII-1	Modus Darii	((#q&r)&(%s&q))		>(%s&r)	vt	
AII-3	Modus Datisi	((#q&r)&(%q&s))		>(%s&r)	vt	
IAI-3	Modus Disamis	((%q&r)&(#q&s))		>(%s&r)	vt	
IAI-4	Modus Diamatis	((%r&q)&(#q&s))		>(%s&r)	vt	
EIO-1	Modus Ferio	((~q&r)&(%s&q))		>(%s&~r)	vt	
EIO-2	Modus Festino	((~r&q)&(%s&q))		>(%s&~r)	vt	
EIO-3	Modus Ferison	((~q&r)&(%q&s))		>(%s&r)	vt	
EIO-4	Modus Fresison	((~r&q)&(%q&s))		>(%q&~r)	vt	
AOO-2	Modus Baroco	((#r&q)&(%s&~q))		>(%s&~r)	vt	
OAO-3	Modus Bocardo	((%q&~r)&(#q&s))		>(%s&~r)	vt	
AAI-3	Modus Darapti	(((#q&r)&(#q&s))	&%q)	>(%s&r)	vt	* not needed
		((#q&r)&(#q&s))		>(%s&r)	vt	
EAO-3	Modus Felapton	(((~q&r)&(#q&s))	&%q)	>(%s&~r)	vt	* not needed
		((~q&r)&(#q&s))		>(%s&~r)	vt	
EAO-4	Modus Fesapo	(((~r&q)&(#q&s))	&%q)	>(%s&~r)	vt	* not needed
		((~r&q)&(#q&s))		>(%s&~r)	vt	

Table 6.1 Original syllogisms in Meth8 script

For those syllogisms with an additional Assumption 3, we test the same expression without the additional assumption. For those syllogisms not needing the given additional assumption in Meth8 to be Tautologous, we comment "not needed" by Meth8.

Meth8 found two anomalies:

> 6.1 EAE-2 Modus Cesare as written is not Tautologous, but with an additional assumption is corrected and Tautologous.
>
> 6.2 AEO-2 Modus Camestros as written is Tautologous, but the original expression without the additional assumption is not Tautologous. (This case is in variance to the other syllogisms with additional assumptions removed that are also Tautologous.)

We rewrite Table 6.1 with the non-redundant and corrected syllogisms according to Meth8 in Table 6.2.

Code	Name	Assumptions 1, 2	Assumption 3	Conclusion	Test	Comments
AAA-1	Modus Barbara	((#q&r)&(#s&q))		>(#s&r)	vt	
AAI-1	Modus Barbari	((#q&r)&(#s&q))		>(%s&r)	vt	
AAI-4	Modus Bamalip	((#r&q)&(#q&s))		>(%s&r)	vt	
EAE-1	Modus Celarent	((~q&r)&(#s&q))		>(~s&r)	vt	
EAE-2	Modus Cesare	(((~r&q)&(#s&q))	&%r)	>(~s&r)	vt	* Meth8 fix
AEE-2	Modus Camestres	((#r&q)&(~s&q))		>(~s&r)	vt	
AEE-4	Modus Calemes	((#r&q)&(~q&s))		>(~s&r)	vt	
EAO-1	Modus Celaront	((~q&r)&(#s&q))		>(~s&r)	vt	
EAO-2	Modus Cesaro	((~r&q)&(#s&q))		>(%s&~r)	vt	
AEO-2	Modus Camestros	(((#r&q)&(~s&q))	&%s)	>(%s&~r)	vt	* needed
AEO-4	Modus Calemos	((#r&q)&(~q&s))		>(%s&~r)	vt	
AII-1	Modus Darii	((#q&r)&(%s&q))		>(%s&r)	vt	
AII-3	Modus Datisi	((#q&r)&(%q&s))		>(%s&r)	vt	
IAI-3	Modus Disamis	((%q&r)&(#q&s))		>(%s&r)	vt	
IAI-4	Modus Diamatis	((%r&q)&(#q&s))		>(%s&r)	vt	
EIO-1	Modus Ferio	((~q&r)&(%s&q))		>(%s&~r)	vt	
EIO-2	Modus Festino	((~r&q)&(%s&q))		>(%s&~r)	vt	
EIO-3	Modus Ferison	((~q&r)&(%q&s))		>(%s&r)	vt	
EIO-4	Modus Fresison	((~r&q)&(%q&s))		>(%q&~r)	vt	
AOO-2	Modus Baroco	((#r&q)&(%s&~q))		>(%s&~r)	vt	
OAO-3	Modus Bocardo	((%q&~r)&(#q&s))		>(%s&~r)	vt	
AAI-3	Modus Darapti	((#q&r)&(#q&s))		>(%s&r)	vt	
EAO-3	Modus Felapton	((~q&r)&(#q&s))		>(%s&~r)	vt	
EAO-4	Modus Fesapo	((~r&q)&(#q&s))		>(%s&~r)	vt	

Table 6.2 Corrected syllogisms by Meth8

Table 6.2 represents the minimal and most compact mapping of the 24 syllogisms in Meth8. We reiterate that Meth8 found two anomalies which was easily corrected to render validated as true.

Meth 8 on Modus Cesare and Modus Camestros © 2016 by Colin James III All rights reserved.

1. Introduction

The logic model checker Meth8 is based on variant system VŁ4, which corrects and resuscitates the quaternary logic of Łukasiewicz. Of the 24 valid syllogisms (from 256 combinations of the Square of Opposition), 15 are deemed valid, and 9 required additional known assumptions to become valid.

We use Meth8 to replicate the 24 valid syllogisms derived from the original Square of Opposition. In the process we make three recent advances.

2. A third assumption is needed to fix up Modus Cesare EAE-2

3. The third assumption cannot be removed from Modus Camestros AEO-2 (as in other syllogisms with known third assumptions); and

4. No third assumptions are required for the other 22 syllogisms.

In our discussion we also present:

5. Analysis of Modus Cesare EAE-2 with Modus Camestros AEO-2

We use public domain definitions from en.wikipedia.org/wiki/Syllogism as mapped to Meth8 script.

LET: # All, % Exists; vt Tautologous, nvt Not Tautologous;
T E, True Evaluated as designated values

2. Additional assumptions required for Modus Cesare EAE-2

LET: q r s, MPS; reptiles fur snakes

The original definition for Modus Cesare EAE-2 is:

No fur is on reptiles.	(PeM)	(~r&q) &
All snakes are reptiles.	(SaM)	(#s&q) >
∴No snakes have fur.	(SeP)	(~s&r) ; nvt

Here are truth tables in the five models:

```
TTTTTTTTTTCCTTTT.EEEEEEEEEEUUEEEE.EEEEEEEEEEEEEEEE.EEEEEEEEEEPPEEEE.EEEEEEEEEEIIEEEE
Model 1         .Model 2.1       .Model 2.2       .Model 2.3.1     .Model 2.3.2
((~r&q)&(#s&q))>(~s&r)   Step: 11
```

The original definition is not Tautologous by Meth8.

We test an additional existential assumption for "some fur exists".

No fur is on reptiles.	(PeM)	(~r&q) &
All snakes are reptiles.	(SaM)	(#s&q) &
Some fur exists.		(%r) >
∴No snakes have fur.	(SeP)	(~s&r) ; vt

Here are truth tables in the five models:

```
TTTTTTTTTTTTTTTT.EEEEEEEEEEEEEEEE.EEEEEEEEEEEEEEEE.EEEEEEEEEEEEEEEE.EEEEEEEEEEEEEEEE
Model 1         .Model 2.1       .Model 2.2       .Model 2.3.1     .Model 2.3.2
(((~r&q)&(#s&q))&%r)>(~s&r)   Step: 13
```

The modified definition is Tautologous by Meth8.

3. Additional known assumption required for Modus Camestros AEO-2

LET: q r s, MPS; reptiles snakes fur

The original definition is:

All snakes are reptiles.	(PaM)	(#r&q) &
No fur is on reptiles.	(SeM)	(~s&q) &
Some fur exists.		(%s) >
∴Some fur is not on snakes.	(SoP)	(%s&~r) ; vt

The original definition is validated by by Meth8.

In all syllogisms with a known additional assumption, it can be removed with the syllogism still being Tautologous by Meth8. We test Modus Camestros AEO-2 for this condition.

All snakes are reptiles.	(PaM)	(#r&q) &
No fur is on reptiles.	(SeM)	(~s&q) >
∴Some fur is not on snakes.	(SoP)	(%s&~r) ; nvt

The original definition without the known additional assumption is not Tautologous by Meth8.

Here are truth tables in the five models:

```
TTTTTTCCTTTTTTTT.EEEEEEUUEEEEEEEE.EEEEEEEEEEEEEEEE.EEEEEEPPEEEEEEEE.EEEEEEIIEEEEEEEE
Model 1         .Model 2.1       .Model 2.2       .Model 2.3.1     .Model 2.3.2
((#r&q)&(~s&q))>(%s&~r)   Step: 11
```

This means that Modus Camestros AEO-2 does not behave as all other valid syllogisms with known additional assumptions when those assumptions are removed.

4. No additional assumptions required for other 22 syllogisms

We show that the other valid syllogisms with known additional assumptions removed are Tautologous by Meth8 in Table 6.2 above.

5. Analysis of Modus Cesare EAE-2 with Modus Camestros AEO-2

We set Modus Cesare EAE-2 and Modus Camestros AEO-2 in opposite columns for comments about shaded variables.

	q r s MPS reptiles fur snakes			q r s MPS reptiles snakes fur	
EAE-2	**Modus Cesare**			**Modus Camestros**	**AEO-2**
(PeM)	No fur is on reptiles.	(~r&q) &	(#r&q) &	All snakes are reptiles.	(PaM)
(SaM)	All snakes are reptiles	(#s&q) &	(~s&q) &	No fur is on reptiles.	(SeM)
	Some fur exists.	(%r) >	(%s) >	Some fur exists.	
(SeP)	∴No snakes have fur.	(~s&r) ; vt	(%s&~r) ; vt	∴Some fur is not on snakes.	(SoP)

5.1 While syllogism models differentiate between first and second premises as antecedent and consequent around the & And connective, Meth8 does not. Therefore if the variable r is replaced by s or vice versa, in one column, then expressions are the same in both columns.

5.2 If the order of the premises is interchanged, then: Modus Cesare EAE-2 becomes Modus Camestres AEE-2 (without or with an additional assumption of %r); and Modus Camestros AEO-2 becomes EAO-3 or EAO-4 (without or with an additional assumption of %r) if the assignments change to q r s MPS fur reptiles snakes.

5.3 The respective conclusions are identical by variable replacement.

5.4 If the modal operators are removed then both syllogisms with the additional assumptions are still Tautologous. This speaks to what we name the *core voracity* of the syllogisms.

6. Conclusion

Variant system VŁ4 as implemented in the Meth8 modal logic checker in five models:

6.1 Corrects Modus Cesare EAE-2 by an additional assumption;

6.2 Shows Modus Camestros AEO-2 must retain its known additional assumption (unlike the other syllogisms that are Tautologous also without it); and

6.3 Presents the table of correct syllogisms in compact Meth8 scripts.

We further demonstrate that:

6.4 The modal operators of necessity and possibility are useful to map exactly the quantifiers of all and exists; and

6.5 The Meth8 tool is qualified to map, evaluate, analyze, and correct this limited fragment of predicate logic.