我們藉由大家熟知的數學例子來理解這個事實. 考慮 $0 \leq x \leq 1$, 這表示 $x \leq 1$ and $x \geq 0$. 它的相反, 大家都知是 $x > 1$ or $x < 0$. 我們可以任取一個數 x 令 P 為 $x \leq 1$ 這一個 statement, 而 Q 為 $x \geq 0$, 則 $\neg P$, $\neg Q$ 分別為 $x > 1$, $x < 0$. 也就是說 $0 \leq x \leq 1$ 可以用 $P \wedge Q$ 表示而 $x > 1$ or $x < 0$ 就是 $(\neg P) \vee (\neg Q)$. 由此可以看出 $\neg(P \wedge Q)$ 和 $(\neg P) \vee (\neg Q)$ 為 logically equivalent, 而不是 $(\neg P) \wedge (\neg Q)$ (否則會得到 $x > 1$ and $x < 0$ 這個矛盾).

我們可以用上一節有關於 statement form 的 logically equivalent 的規則來處理 not. 例如將式子 (1.8) 中的 P, Q 分別用 $\neg P$ 和 $\neg Q$ 取代, 可得

$$\neg((\neg P) \wedge (\neg Q)) \sim (\neg(\neg P)) \vee (\neg(\neg Q)).$$

再利用 $\neg(\neg P) \sim P$, 得

$$\neg((\neg P) \wedge (\neg Q)) \sim (P \vee Q).$$

最後兩邊取 not, 得

$$\neg(P \vee Q) \sim (\neg P) \wedge (\neg Q). \tag{1.9}$$

例如考慮 $x \geq 0$ 的情形, 我們知它的相反為 $x < 0$. 若令 P, Q 分別為 $x > 0$, $x = 0$, 則 $x \geq 0$ 即為 $P \vee Q$. 此時 $\neg P$ 為 $x \leq 0$, $\neg Q$ 為 $x \neq 0$. 而 $(\neg P) \wedge (\neg Q)$ 為 $x \leq 0$ and $x \neq 0$, 即為 $x < 0$ 也就是 $x \geq 0$ 的相反.

式子 (1.7), (1.8), (1.9) 對於推導和 not 有關的 statement forms 之間的 logical equivalence 相當重要. 其中式子 (1.8), (1.9) 稱為 *DeMorgan's laws*.

接下來我們自然會問, 怎樣的 statement form 會和 $\neg(P \Rightarrow Q)$ logically equivalent 呢? 或許大家會認為是 $P \Rightarrow \neg Q$, 不過利用 truth table 檢查一下, 大家會發現在 P 是對的時 $P \Rightarrow Q$ 和 $P \Rightarrow \neg Q$ 確實對錯相反, 但是當 P 為錯時 $P \Rightarrow Q$ 和 $P \Rightarrow \neg Q$ 皆為對. 所以 $\neg(P \Rightarrow Q)$ 和 $P \Rightarrow \neg Q$ 並不是 logically equivalent, 千萬要記住.

Question 1.12. *試寫下會使得 $x \geq 0 \Rightarrow x \geq 1$ 為對的所有實數 x, 也寫下會使得 $x \geq 0 \Rightarrow x < 1$ 為對的所有實數 x. 它們是否相反呢?*

大家常忽略的就是 $P \Rightarrow Q$ 中 P 錯的情況, 而造成邏輯的錯誤, 千萬要注意. 不過另一方面, 若 A, B 為 statement form 且 A 為 tautology, 那麼 $\neg(A \Rightarrow B)$ 就和 $A \Rightarrow \neg B$ 為 logically equivalent. 主要的原因是, A 既然全為對, 那麼 $A \Rightarrow B$ 的對錯完全會和 B 的對錯完全一致了.

Question 1.13. *試寫下會使得 $x^2 \geq 0 \Rightarrow x > 0$ 為對的所有實數 x, 也寫下會使得 $x^2 \geq 0 \Rightarrow x \leq 0$ 為對的所有實數 x. 它們是否相反呢?*

要處理 $\neg(P \Rightarrow Q)$ 會和什麼為 logically equivalent, 我們可以換一個角度來看 $P \Rightarrow Q$. 首先回顧一下 $P \Rightarrow Q$ 較通俗的說法是 P 對則 Q 一定對. 所以我們知道 Q 會對, 除非 P 是錯的. 也就是說要不然是 Q 對, 要不然就是 P 錯. 這讓我們想到 $Q \vee \neg P$ 這一個 statement form. 事實上用 truth table 檢驗

P	Q	$\neg P$	$Q \vee \neg P$
T	T	F	T
T	F	F	F
F	T	T	T
F	F	T	T

我們得到
$$(P \Rightarrow Q) \sim (Q \vee \neg P). \tag{1.10}$$
利用 $(Q \vee \neg P) \sim ((\neg P) \vee Q)$ 以及 $\neg(\neg Q) \sim Q$, 我們得 $(P \Rightarrow Q) \sim ((\neg P) \vee \neg(\neg Q))$, 再利用式子 (1.10) 得 $((\neg P) \vee \neg(\neg Q)) \sim ((\neg Q) \Rightarrow (\neg P))$, 故知
$$(P \Rightarrow Q) \sim ((\neg Q) \Rightarrow (\neg P)). \tag{1.11}$$
這和我們提過 $P \Rightarrow Q$ 為對, 表示若 Q 為錯則 P 一定錯, 相吻合.

利用式子 (1.10), 我們可得 $\neg(P \Rightarrow Q) \sim \neg(Q \vee \neg P)$. 而由 DeMorgan's laws 知
$$\neg(Q \vee \neg P) \sim ((\neg Q) \wedge \neg(\neg P))$$
故得
$$\neg(P \Rightarrow Q) \sim (P \wedge (\neg Q)). \tag{1.12}$$
式子 (1.10), (1.11), (1.12) 是我們將來處理 "若 P 則 Q" 這種類型的論述時常用的 logical equivalences.

由式子 (1.10) 我們知道, 所有的 statement form 都可以利用 logical equivalence 寫成 $\neg, \wedge, \vee$ 的組合. 例如由 $P \Leftrightarrow Q$ 的定義, 我們可得
$$(P \Leftrightarrow Q) \sim (Q \vee (\neg P)) \wedge (P \vee (\neg Q)). \tag{1.13}$$
再利用 $\wedge, \vee$ 的分配性 (即式子 (1.3)) 推得
$$(P \Leftrightarrow Q) \sim (P \wedge Q) \vee ((\neg P) \wedge (\neg Q)). \tag{1.14}$$
因此我們可以用 DeMorgan's laws, 式子 (1.7), 以及 $\wedge, \vee$ 之間的關係式 (式子 (1.1),(1.2), (1.3)), 推導出一個 statement form 取 not 之後的 logical equivalence. 例如式子 (1.13) 取 not 可得
$$\neg(P \Leftrightarrow Q) \sim ((\neg Q) \wedge P) \vee ((\neg P) \wedge Q).$$
有趣的是, 若比較式子 (1.14) 中的 Q 用 $\neg Q$ 取代後的結果, 我們得到
$$\neg(P \Leftrightarrow Q) \sim (P \Leftrightarrow \neg Q).$$

當 A 為 statement form 時, $\neg A$ 的對錯完全和 A 的對錯相反, 所以 $A \Leftrightarrow \neg A$ 的 truth table 在任何情況之下皆為錯, 可知 $A \Leftrightarrow \neg A$ 為 contradiction. 反之, 若 B 為 statement form 且 $A \Leftrightarrow B$ 為 contradiction, 表示在任何情況下 A 和 B 的對錯情況相反, 可知 $B \sim \neg A$. 因此我們有以下和 Proposition 1.2.2 相對應的性質.

Proposition 1.3.1. 假設 A,B 為兩個 *statement forms.* 則 $\neg A$ 和 B 為 *logically equivalent* 等同於 $A \Leftrightarrow B$ 為 *contradiction.*

1.4. Quantifiers

我們已經了解在已知各 statement 的對錯情況之下它們用 connective 以及 not 連接之後其對錯的狀況, 我們也知道一個 statement form 的否定為何. 不過一個單一的 statement, 很可能就很複雜, 不容易判斷對錯. 例如在數學上一個 statement 常常會有一些 quantifier (量詞) 出現, 而增加了判斷對錯的困難度. 在本節中我們將介紹常見的 quantifiers, 並探討它們取否定的情形.

數學上常見的 quantifiers 有以下幾種:

- "for all", "for every" (即對所有的), 常用 $\forall$ 表示.
- "there exists", "there is" (即存在, 可以找到), 常用 $\exists$ 表示.
- "there is a unique" (即存在唯一的), 常用 $\exists!$ 表示.

$\exists!$ 牽涉到唯一性的問題, 以後我們在談論證明方法時會提到它, 這裡我們先探討 $\forall$ 和 $\exists$. 首先要說明的是, 在談論這些 quantifiers 時必須說明清楚是在怎樣的集合內. 比方說對所有的整數和對所有的有理數就是完全不同的兩回事, 而存在一個自然數和存在一個偶數也不同. 不過由於我們僅介紹這些 quantifiers 的概念, 而不觸及證明. 所以這裡為了簡單起見我們說明的例子考慮的都是整個實數. 例如我們說 $\forall x$ 或 $\exists x$, 它們分別表示的就是 for all x in $\mathbb{R}$ 或 there exists an x in $\mathbb{R}$, 以後就不再聲明指的是實數了.

我們先看簡單的例子: $\forall x, x^2 \geq 0$. 指的就是所有的實數 x 皆會滿足 $x^2 \geq 0$. 我們知道這個 statement 是對的, 因為每一個實數 x 都對, 沒有例外. 這類的 statement 我們可以用以下的形式表示 "$\forall x, P(x)$". 這裡 $P(x)$ 指的是和 x 有關的條件 (例如上例中 $P(x)$ 就是 $x^2 \geq 0$). 它指的就是所有的 x 皆會滿足 $P(x)$ 這個條件. 這個 statement 要對就必須所有的 x 都對, 一個都不能錯. 例如 $\forall x, x^2 > 0$ 便是錯的 ($x = 0$ 就不成立).

類似的, 我們可以用 "$\exists x, P(x)$" 來表示, 存在 x 使得 $P(x)$ 成立. 這個 statement 要對, 只要能找到一個 x 使得 $P(x)$ 成立即可. 注意它並沒有說有多少個會對, 有可能很多, 有可能只有一個, 所以只要找到一個對即可 (這就是英文用 there exists 的原因). 上面提過 $\forall x, x^2 > 0$ 是錯的, 但若改為 $\exists x, x^2 > 0$ 便是對的 (取 $x = 1$, 即可).

$\forall$ 和 $\exists$ 有著有趣的關係, 例如 "$\forall x, P(x)$" 是對的話, 那麼 "$\exists x, P(x)$" 就一定對 (只要挑隨便一個 x 即可). 不過反過來就不對. 你不能隨便挑幾個 x 符合 $P(x)$, 就聲稱對所有的 x 都會符合 $P(x)$. 另外 $\forall$ 和 $\exists$ 在取否定時關係就更密切了. 當你發現 "$\forall x, P(x)$" 有可能錯時, 如何說明它是錯的呢? 前面說過 "$\forall x, P(x)$" 只要有一個 x 不符合 $P(x)$ 就是錯的, 所以要否定它, 我們只要找到一個 x 讓 $P(x)$ 不成立即可. 用符號表示就是 $\exists x, \neg P(x)$. 例如前面提過 $\forall x$, $x^2 > 0$ 是錯的, 因為我們發現 $\exists x, x^2 \leq 0$.

注意很多同學會誤以為 "$\forall x, P(x)$" 的否定是 "$\forall x, \neg P(x)$". 雖然若 "$\forall x, \neg P(x)$" 是對的可以知道 "$\forall x, P(x)$" 是錯的. 但是 "$\forall x, P(x)$" 是錯的, 並不表示 "$\forall x, \neg P(x)$" 是對的. 所以不能說 "$\forall x, P(x)$" 的否定是 "$\forall x, \neg P(x)$". 例如 $\forall x, x^2 > 0$ 是錯的, 但 $\forall x, x^2 \leq 0$ 也是錯的, 唯

有 $\exists x,\ x^2 \leq 0$ 才會對. 大家千萬注意, 不要弄錯. 總而言之我們有以下的 logical equivalence

$$\neg(\forall x, P(x)) \sim (\exists x, \neg P(x)). \tag{1.15}$$

同理要否定 "$\exists x,\ P(x)$", 表示找不到 x 使得 $P(x)$ 成立. 所以我們便需說明所有的 x 皆不滿足 $P(x)$, 也就是說 $\forall x,\ \neg P(x)$. 同樣的, 很多同學會誤以為 "$\exists x,\ P(x)$" 的否定是 "$\exists x,\ \neg P(x)$". 這是錯的, 因為找到 x 不滿足 $P(x)$ 還是有可能找到另一個 x 會滿足 $P(x)$. 因此光由 "$\exists x,\ \neg P(x)$" 並不能否定 "$\exists x,\ P(x)$". 總而言之我們有以下的 logical equivalence

$$\neg(\exists x, P(x)) \sim (\forall x, \neg P(x)). \tag{1.16}$$

Question 1.14. *試利用式子 (1.15) 以及 logical equivalence 的規則推導出式子 (1.16).*

Quantifier 有時會發生在兩個或更多變數的情形, 這裡我們僅探討兩個變數的情形, 更多變數的情況可以依兩個變數的情況類推下去. 所謂兩個變數的情況, 是形如 "$\forall x, \exists y, P(x,y)$" 的 statement, 這裡 $P(x,y)$ 指的是和 x,y 有關的條件. 例如微積分中, 函數 $f(x)$ 在 $x=a$ 的極限為 l (即 $\lim_{x\to a} f(x) = l$) 的定義 "$\forall \varepsilon > 0, \exists \delta > 0, 0 < |x-a| < \delta \Rightarrow |f(x) - l| < \varepsilon$" 就是兩個變數的情況. 大致上我們會有下面四種類型的 statement.

$$(1)\forall x, \exists y, P(x,y) \quad (2)\exists x, \forall y, P(x,y) \quad (3)\forall x, \forall y, P(x,y) \quad (4)\exists x, \exists y, P(x,y).$$

(1) 指的是: 對於所有的 x 皆可找到 y 使得 $P(x,y)$ 成立. 注意這裡 x 的部分先講, 再提存在 y, 所以這個存在的 y 並不是固定的, 它可能會隨著 x 的選取而變動. 例如 $\forall x, \exists y, x+y=0$ 這個 statement 是對的. 它說任意選取 x, 皆可找到 y 滿足 $x+y=0$. 這裡 y 會隨著 x 而變動, 即 $y=-x$. 例如 $x=1$ 時 $y=-1$, 而 $x=2$ 時 $y=-2$. 這裡 x,y 的先後順序很重要, 千萬要注意.

(2) 指的是: 存在 x 使得對所有的 y 都會滿足 $P(x,y)$. 注意這裡存在的 x 先講, 再提所有的 y, 所以這個存在的 x 並是固定的, 它不可以隨著 y 而變動. 例如 $\exists x, \forall y, x+y=y$ 這個 statement 是對的. 它是說可以找到 x 讓任意的 y 皆滿足 $x+y=y$. 這裡 x 找到後便固定下來了, 即 $x=0$. 不過例如在 (1) 的情形我們知道 $\forall x, \exists y, x+y=0$ 這個 statement 是對的, 但若將 $\forall x$ 和 $\exists y$ 的順序交換得 $\exists y, \forall x, x+y=0$ 這個 statement 便是錯的. 因為我們無法找到一個固定的 y 使的所有的 x 都會滿足 $x+y=0$. 再次強調, 這裡先後順序很重要, "$\forall x, \exists y, P(x,y)$" 和 "$\exists y, \forall x, P(x,y)$" 雖然只是 $\forall x$ 和 $\exists y$ 先後順序調動, 但意義完全不同千萬要注意.

Question 1.15. *$\exists x, \forall y, x+y=y$ 這個 statement 是對的, 但若換成 $\forall y, \exists x, x+y=y$, 是否為對呢? 又換成 $\forall x, \exists y, x+y=y$ 及 $\exists y, \forall x, x+y=y$, 哪一個對呢?*

Question 1.16. *假設 $f(x,y), g(x,y)$ 皆為兩個變數的多項式. 已知 "$\forall x, \exists y, f(x,y)=0$" 和 "$\exists y, \forall x, g(x,y)=0$" 皆為對. 試問 $f(x,y)=0$ 和 $g(x,y)=0$ 在坐標平面上的圖形哪一個一定會包含一條水平直線, 哪一個一定會和鉛直線 $x=101$ 相交?*

(3) 和 (4) 的情況較為單純. (3) 指的是任取一個 x, 對於任意的 y 都會使得 $P(x,y)$ 成立. 利用坐標平面的看法, 我們可以說平面上任一點 (x,y) 都會使得 $P(x,y)$ 成立, 所以此時

$\forall x$ 和 $\forall y$ 變換順序並不會改變整個 statement. 而 (4) 指的是可以找到 x 使得有一個 y 滿足 $P(x,y)$. 利用坐標平面的看法, 我們可以說平面上存在一點 (x,y) 使得 $P(x,y)$ 成立. 因此此時 $\exists x$ 和 $\exists y$ 變換順序並不會改變整個 statement. 例如若我們在 $x=3$ 時, 可找到 $y=7$ 使得 $P(3,7)$ 是正確的, 此時我們也可以說 $y=7$ 時, 可找到 $x=3$ 使得 $P(x,y)$ 為對. 總而言之 (3), (4) 因兩個變數的 quantifier 皆相同, 所以 x,y 的先後不重要. (3) 一般會簡化成 $\forall x,y$, $P(x,y)$, 而 (4) 簡化成 $\exists x,y$, $P(x,y)$.

接下來我們來看有兩個變數的 statement 取否定時 quantifier 的變化情形. 在 (1) 的情形, 即 "$\forall x,\exists y,P(x,y)$". 此時, 我們可以把 "$\exists y,P(x,y)$" 看成是 $H(x)$ 這樣的條件. 所以原 statement 可看成 $\forall x,H(x)$. 利用式子 (1.15), 我們知道它的否定為 $\exists x,\neg H(x)$. 然而式子 (1.16) 告訴我們 $\neg H(x)\sim(\forall y,\neg P(x,y))$, 所以我們得

$$\neg(\forall x,\exists y,P(x,y)) \quad\sim\quad (\exists x,\forall y,\neg P(x,y)).$$

同理我們可得

$$\begin{aligned}\neg(\exists x,\forall y,P(x,y)) &\quad\sim\quad (\forall x,\exists y,\neg P(x,y))\\ \neg(\forall x,\forall y,P(x,y)) &\quad\sim\quad (\exists x,\exists y,\neg P(x,y))\\ \neg(\exists x,\exists y,P(x,y)) &\quad\sim\quad (\forall x,\forall y,\neg P(x,y)).\end{aligned}$$

例如前面所提, 函數 $f(x)$ 滿足 $\lim_{x\to a}f(x)=l$ 的否定應為

$$\exists\varepsilon>0,\forall\delta>0,\neg(0<|x-a|<\delta\Rightarrow|f(x)-l|<\varepsilon).$$

利用式子 (1.12) 我們知

$$\neg(0<|x-a|<\delta\Rightarrow|f(x)-l|<\varepsilon) \quad\sim\quad ((0<|x-a|<\delta)\wedge(|f(x)-l|\geq\varepsilon)).$$

所以 $\lim_{x\to a}f(x)=l$ 的否定應為

$$\exists\varepsilon>0,\forall\delta>0,(0<|x-a|<\delta)\wedge(|f(x)-l|\geq\varepsilon).$$

最後, 我們說明一下 $\forall$ 和 $\exists$ 在習慣上用法的差異. 在習慣上的用語, 我們常會省略 $\forall x$. 例如 $x\geq 3\Rightarrow x^2\geq 9$, 這一個 statement 嚴格來說應寫成 $\forall x,x\geq 3\Rightarrow x^2\geq 9$. 也就是說, 在邏輯上我們說這個 statement 是對的應該是對所有的實數 x 都是對的. 給定一實數 x, 當 $x\geq 3$, 當然可得 $x^2\geq 9$. 而當 $x<3$, 因為它已不符合 $x\geq 3$ 的前提, 我們知道此時 $x\geq 3\Rightarrow x^2\geq 9$ 也是對的. 所以我們可以認定 $\forall x,x\geq 3\Rightarrow x^2\geq 9$ 是對的 (這也是邏輯上定義 P 錯時 $P\Rightarrow Q$ 為對的用意, 希望同學能體會). 要注意的是 $\exists x$ 就絕不能省略, 否則就弄不清楚是 $\forall x$ 或 $\exists x$ 了. 總而言之, 當我們碰到「所有 x 只要符合 $P(x)$ 也會符合 $Q(x)$」這種 statement 時, 原本應寫成「$\forall x,P(x)\Rightarrow Q(x)$」, 我們常省略 $\forall x$ 而用「$P(x)\Rightarrow Q(x)$」來表達. 而如果是「存在 x 符合 $P(x)$ 也符合 $Q(x)$」, 我們就不能省略 $\exists x$. 不過要注意, 即使 $\exists x$ 沒省略, 也不可以寫成「$\exists x,P(x)\Rightarrow Q(x)$」, 而是用「$\exists x,P(x)\wedge Q(x)$」來表達. 這是因為,「$\exists x,P(x)\Rightarrow Q(x)$」表示只要找到 x 使得 $P(x)\Rightarrow Q(x)$ 為對就可, 但所有不滿足 $P(x)$ 的 x 都會使得 $P(x)\Rightarrow Q(x)$ 為對. 這會造成即使沒有 x 滿足 $P(x)$, 也會使得「$\exists x,P(x)\Rightarrow Q(x)$」為對, 這就和原來的「存在 x 符合 $P(x)$ 也符合 $Q(x)$」意義相左了. 例如「存在一個大於 3 的實數 x, 滿足 $x^2=10$」就應

寫成「$\exists x, (x>3)\wedge(x^2=10)$」(此時 $x=\sqrt{10}$), 而不是「$\exists x, (x>3)\Rightarrow(x^2=10)$」(此時 $x=2$ 也會對).

Question 1.17. *假設 $f(x,y)$ 是一個兩個變數的多項式.「存在一實數 $a>0$ 使得 $f(a,y)=0$ 無解」這一個 statement, 數學的表示法為何? 並寫出這 statement 的否定.*