CypRules有五種不同的model : 1A2, 2C9, 2C19, 2D6, and 3A4；彼此獨立，有不同的rules，因此必須分開建構，以下大致介紹其程式流程。

1. CypRules執行過程
2. 計算Descriptors (Pubchem 2D fingerprints, PaDEL 2D descriptors, PaDEL 3D descriptors, and Mold2 descriptors)
3. 將來自PaDEL program和Mold2 program計算的descriptors進行merge到同一個file的動作，並同時修改成接下來要匯給預測程式的檔案格式。
4. 由預測程式，判斷descriptors與rules間之對應關係，並output出預測結果。
5. 將預測結果檔案進行解析，並轉成可易於作網路程式(php)讀取的資料格式。
6. 計算Descriptors (不同的model, include不同的 descriptor sets之組合)
7. 1A2: PaDEL 2D
8. 2C9: PaDEL 2D+ Mold2+Pubchem2D+PaDEL 3D
9. 2C19: PaDEL 2D+ Mold2+Pubchem2D+PaDEL 3D
10. 2D6: Pubchem2D
11. 3A4: PaDEL 2D+ Mold2+Pubchem2D+PaDEL 3D
12. 請注意我使用的PaDEL 版本，計算出來的PaDEL 2D descriptors共1444個；PaDEL 3D descriptors共431個; Mold2共777個；Pubchem2d共881個。
13. Merge Descriptors
14. 先去除來自PaDEL檔案和Mold2檔案的第一個row(欄位名稱)
15. 依照PaDEL 2D+ Mold2+Pubchem2D+PaDEL 3D 的順序將descriptors merge在同一個檔案裡(但每一個rows的descriptors目前是依照這個順序作排列)
16. 其中PaDEL 2D descriptors中，必須刪除index i, 第i=1-3, i=459-464, i=581-636, i=1248, i=1411, i=1442之descriptors (i從0起算，例如i=2，是指第3個descriptors.)
17. 針對PaDEL 2D, 或3D，必須增加一個判斷式，如果某個欄位的值是null(指沒有任何資料，因為PaDEL的output結果是用逗號隔開，所以有時候會有幾個欄位裡面兩個逗號間沒有任何資料存在的情況) 或是 值為”Infinity”(資料欄位它直接顯示為”Infinity”)，這兩種情況將descriptors值設為0.
18. 預測程式
19. 將merge好的程式，連同已經建構好的rules file、descriptors name files當作predict程式的input，程式會幫你直接output出預測結果。
20. Input file name必須follow以下格式:

“anyName.cases”: descriptor files

“anyName.rules”: rule files

“anyName.names”: descriptor name files

anyName是自己取的，三個檔案必須一致，其它後面的部分不能更動。

1. 執行的command “./predict/predict -R -f anyName > AnyYourOutputFileName”

“anyName”要跟你input的(b) name一致。

1. Rules files，它是由數個不同的大Rule model所組成，例如共有十組Rule Model；而每個Rule Model例如可以有100種rule set；每一個Rule set則是由數個rule所組成，每個rule可能就像是長成像”Weight>10”這樣。一個compound符合同一個rule set裡的所有rules才算是符合這個rule set。而這個compound在同一個rule model中，符合夠多的active rule sets，該compound”對這個rule model”就是active，反之亦然。但是因為有十種(“十”是個舉例)不同的rule models，要判斷一個compound是active還是inactive，還須採用多數決，因此，一個compounds若在十個rule model中，有六個以上的rule model說它是active，那它就是active，反之亦然。
2. 解析程式
3. predict程式，只會告訴你每一個compound是active or inactive；以及告訴你它符合哪個rule model裡的哪個rule set。但因為這樣會給使用者過多的rule，因此，還須要再進行進一步的解析，只將這些符合的rule set中，出現過最多次的前五名”rule”顯示出來給使用者即可。
4. 經由parse程式分析後，檔案內容舉例如下(只有一個compound時，多個compounds就會有多個rows):

"AUTOGEN\_267",NonInhibitor,5,P20649,0,3.00857,3.27669,D142,1,0.318306,0.251835,P20154,0,-1420.79,-33.5145,F193,1,0,0,P3290,1,1.11692,0

這是表示name叫作AUTOGEN\_267的compound，它被預測為”nonInhibitor”，而且有5個top rules, 在5的逗點之後就會有5x4個欄位資料被逗點隔開，每四個欄位代表一組rule，其中隨後第一個欄位資料代表descriptor name(P20649)；第二個欄位，0表示”>”(大於)；1表示”<=”(小於等於)；第三個欄位為threshold value；第四個欄位代表這個compound，針對這個descriptor的值是多少。以此例子而言，

Top rule為 P20649>3.00857，且AUTOGEN\_267的P20649值為3.27669

Second rule為D142<= 0.318306,且AUTOGEN\_267的D142值為0.251835

…依此類推

六、附件檔案

(a) predict資料夾，提供predict程式 (這是當初我們applied C5.0 program所提供出來的程式，不是我們寫的，主要是怕我們自己重新coding會有誤)，它是c code的程式，有直接給你執行檔，但我不確定是不是就會有跨平台的問題。

(b) 如前面講的merge和parse的程式也都有放在裡面原始的c code(自己寫的)，檔名分別為trim\_xxx.cpp, parse\_xxx.cpp，xxx是model name.

(c) 附上每個model在計算過程中的範例檔，以及rules files:

xxx.case2為PaDEL和mold2跑出來的descriptors files

xxx.case是merge descriptors後的結果

xxx.names是descriptor column name (input給predict程式用)

xxx.rules是model rules

xxx.predict是predict後的結果

xxx.statistics是parse後的結果