

## 1. The meaning of each JAVA file

### Preprocessing.java

Input 格式為<user, movie, grade, date>

Mapper：讀入 input，並將其 map 成<user, [movie, grade, date]> 的 key-value pairs。

Reducer：共產生兩種 output

(1) 紀錄每個使用者對看過的每部電影最後讀入的評分，並計算對每一部電影最初讀入評分的平均  $\mu$ ，產生<"M", [user, movie, grade- $\mu$ ]>的 output，即 utility matrix M。

(2) 利用對每一部電影最初讀入評分的和 sum，及計算每位使用者總共對幾部電影評分 count，產生<user, [sum, count]>

getMeanFromDFS：利用每個使用者所有評分的和 sum，及看過幾部電影 cardinality，計算其所有評分的平均。

### UVGen.java

目標為產生初始的 U、V Matrix，共產生兩個檔，格式分別為<"U", user, dim, value>及<"V", dim, movie, value>。其中兩者的 value 皆以 random Gaussian 獲得。

### IterationU.java

計算 U Matrix。

Mapper：會有 M、U、V 三種 input。M、U 會以 user 為 key，然後將內容傳到個別的 Reducer 中；而 V 的內容則是會傳到每一個 Reducer 中。

Reducer：根據 M、U、V 來建立 Matrix，並計算產生新的 U Matrix。

### IterationUSol2.java

另一種計算 U Matrix 的方式。與 IterationU.java 相比，在 Mapper 中少了 V 的 input，而是在 Reducer 中將 V 讀入，其餘部分則與 IterationU.java 相似。

### IterationV.java

計算 V Matrix。

Mapper：會有 M、U、V 三種 input。M、V 會以 movie 為 key，然後將內容傳到個別的 Reducer 中；而 U 的內容則是會傳到每一個 Reducer 中。

Reducer：根據 M、U、V 來建立 Matrix，並計算產生新的 V Matrix。

### IterationVSol2.java

另一種計算 V Matrix 的方式。與 IterationU.java 相比，在 Mapper 中少了 U 的 input，而是在 Reducer 中將 U 讀入，其餘部分則與 IterationV.java 相似。

### **Main.java**

Recommender 的流程，先以 preprocessing 求得平均及初始的 U、V Matrix，接著一直重複計算出新的 U、V Matrix 以得到更好的解，這裡所採用的是 solution 2 的方法。

## 2. Flow chart

