mllib已封装好als的算法，

具体使用方法可以参照

<http://spark.apache.org/docs/latest/mllib-collaborative-filtering.html>

和

<https://github.com/apache/spark/blob/master/examples/src/main/scala/org/apache/spark/examples/mllib/MovieLensALS.scala>

我这里主要参考了后者，对她加以改进，MovieLensALS.scala里只有计算rmse的代码，我假如了推荐的代码。

用法1 计算rmse

spark-submit --driver-memory 12G --class org.apache.spark.mllib.als.AlsRec Als.jar --rank 7 --numIterations 20 --lambda 1.0 --kryo recommendation/data/ua.base

具体参数说明可以参看MovieLensALS.scala

要注意的是，recommendation/data/ua.base是评分记录，代码中会将它随机分成两部分，训练集和测试集。

用法2 根据已有评分记录对所有用户做推荐

spark-submit --driver-memory 12G --class org.apache.spark.mllib.als.AlsRec Als.jar --rank 7 --numIterations 20 --lambda 1.0 --kryo --isRec --output recommendation/output/als recommendation/data/ua.base

参数说明：

a –isRec 有这个标志表示做推荐，没有表示计算rmse

b –output推荐结果存放目录

注：

mllib中实现的als的参考论文

Large-scale Parallel Collaborative Filtering for the Netflix Prize

Collaborative Filtering for Implicit Feedback Datasets