Quick Start Tutorial of JPA

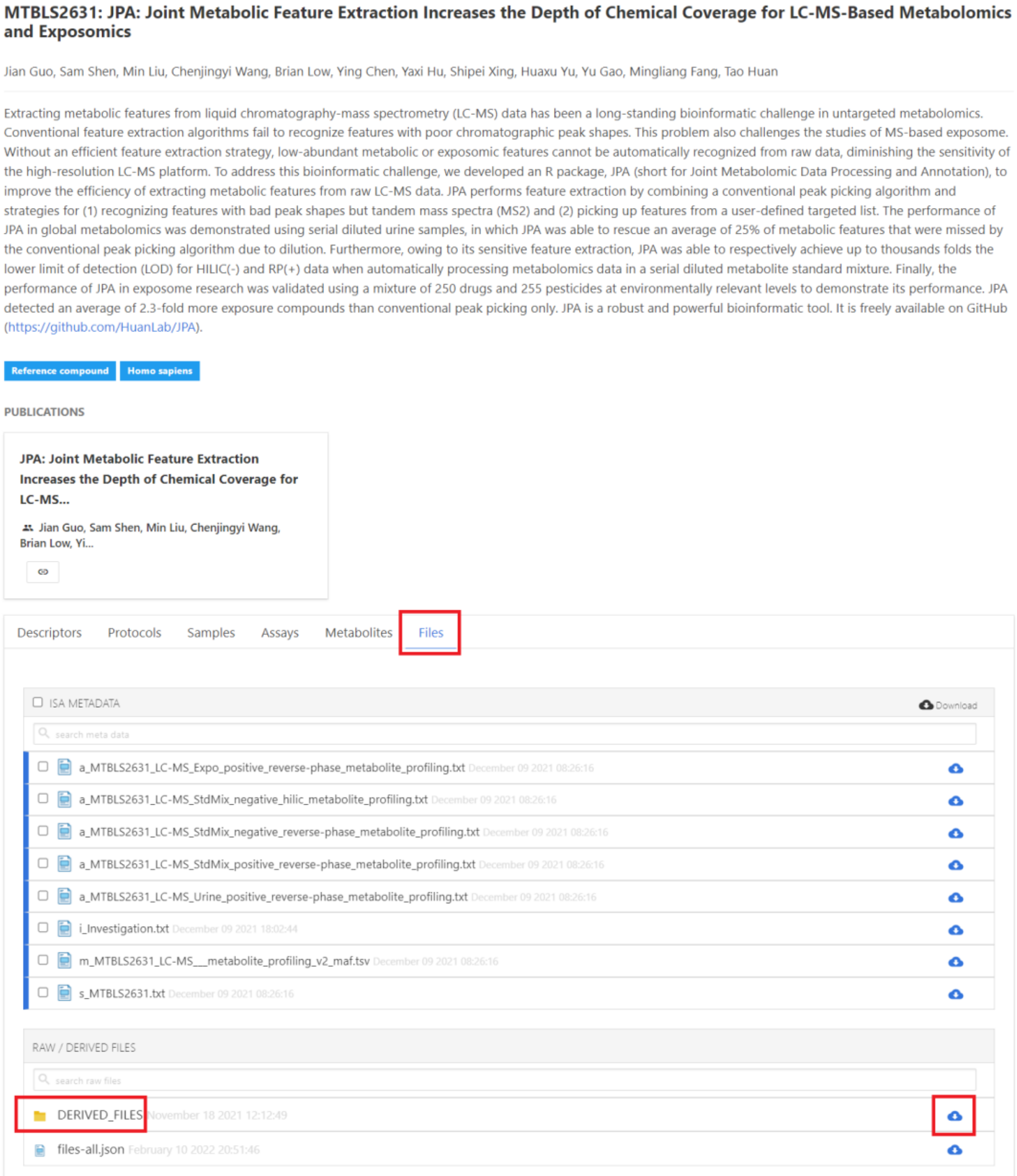
Jian Guo, Tao Huan

2022-02-10

This is a quick start guide for using JPA.

Step 1. Download raw LC-MS data.

Go to the MetaboLights repository at https://www.ebi.ac.uk/metabolights/reviewerc02318d9-6154-4e8d-80e8-11732423d32e. Click the “Files” tab, and then click the folder “DERIVED_FILES”. Download the folder by clicking the cloud icon to the right (Figure 1).



MTBLS2631: JPA: Joint Metabolic Feature Extraction Increases the Depth of Chemical Coverage for LC-MS-Based Metabolomics and Exposomics

Jian Guo, Sam Shen, Min Liu, Chenjingyi Wang, Brian Low, Ying Chen, Yaxi Hu, Shipei Xing, Huaxu Yu, Yu Gao, Mingliang Fang, Tao Huan

Extracting metabolic features from liquid chromatography-mass spectrometry (LC-MS) data has been a long-standing bioinformatic challenge in untargeted metabolomics. Conventional feature extraction algorithms fail to recognize features with poor chromatographic peak shapes. This problem also challenges the studies of MS-based exposome. Without an efficient feature extraction strategy, low-abundant metabolic or exposomic features cannot be automatically recognized from raw data, diminishing the sensitivity of the high-resolution LC-MS platform. To address this bioinformatic challenge, we developed an R package, JPA (short for Joint Metabolomic Data Processing and Annotation), to improve the efficiency of extracting metabolic features from raw LC-MS data. JPA performs feature extraction by combining a conventional peak picking algorithm and strategies for (1) recognizing features with bad peak shapes but tandem mass spectra (MS2) and (2) picking up features from a user-defined targeted list. The performance of JPA in global metabolomics was demonstrated using serial diluted urine samples, in which JPA was able to rescue an average of 25% of metabolic features that were missed by the conventional peak picking algorithm due to dilution. Furthermore, owing to its sensitive feature extraction, JPA was able to respectively achieve up to thousands folds the lower limit of detection (LOD) for HILIC(-) and RP(+) data when automatically processing metabolomics data in a serial diluted metabolite standard mixture. Finally, the performance of JPA in exposome research was validated using a mixture of 250 drugs and 255 pesticides at environmentally relevant levels to demonstrate its performance. JPA detected an average of 2.3-fold more exposure compounds than conventional peak picking only. JPA is a robust and powerful bioinformatic tool. It is freely available on GitHub (https://github.com/HuanLab/JPA).

PUBLICATIONS

JPA: Joint Metabolic Feature Extraction Increases the Depth of Chemical Coverage for LC-MS...

Jian Guo, Sam Shen, Min Liu, Chenjingyi Wang, Brian Low, Yi...

□ ISA METADATA

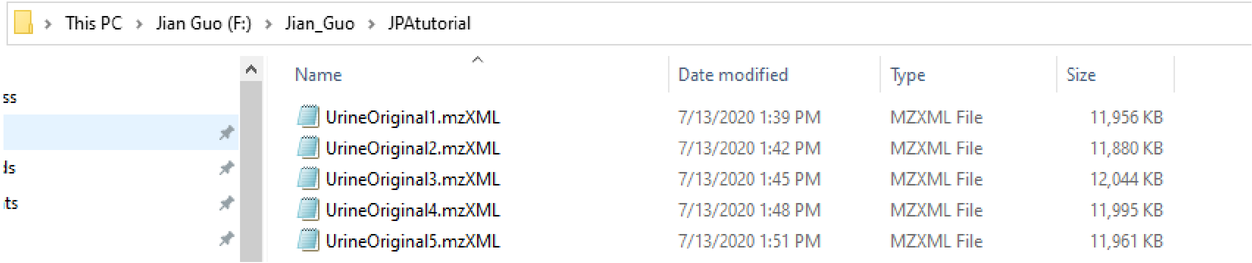
- □ a_MTBLS2631_LC-MS_Expo_positive_reverse-phase_metabolite_profiling.txt December 09 2021 08:26:16
- □ a_MTBLS2631_LC-MS_StdMix_negative_hilic_metabolite_profiling.txt December 09 2021 08:26:16
- □ a_MTBLS2631_LC-MS_StdMix_negative_reverse-phase_metabolite_profiling.txt December 09 2021 08:26:16
- □ a_MTBLS2631_LC-MS_StdMix_positive_reverse-phase_metabolite_profiling.txt December 09 2021 08:26:16
- □ a_MTBLS2631_LC-MS_Urine_positive_reverse-phase_metabolite_profiling.txt December 09 2021 08:26:16
- □ i_Investigation.txt December 09 2021 18:02:44
- □ m_MTBLS2631_LC-MS__metabolite_profiling_v2_maf.tsv December 09 2021 08:26:16
- □ s_MTBLS2631.txt December 09 2021 08:26:16

RAW / DERIVED FILES

- DERIVED_FILES November 18 2021 12:12:49
- files-all.json February 10 2022 20:51:46

Figure 1. The repository for the raw data of the JPA study.

Step 2. Extract the downloaded raw data (.mzXML files) into a folder (Figure 2).

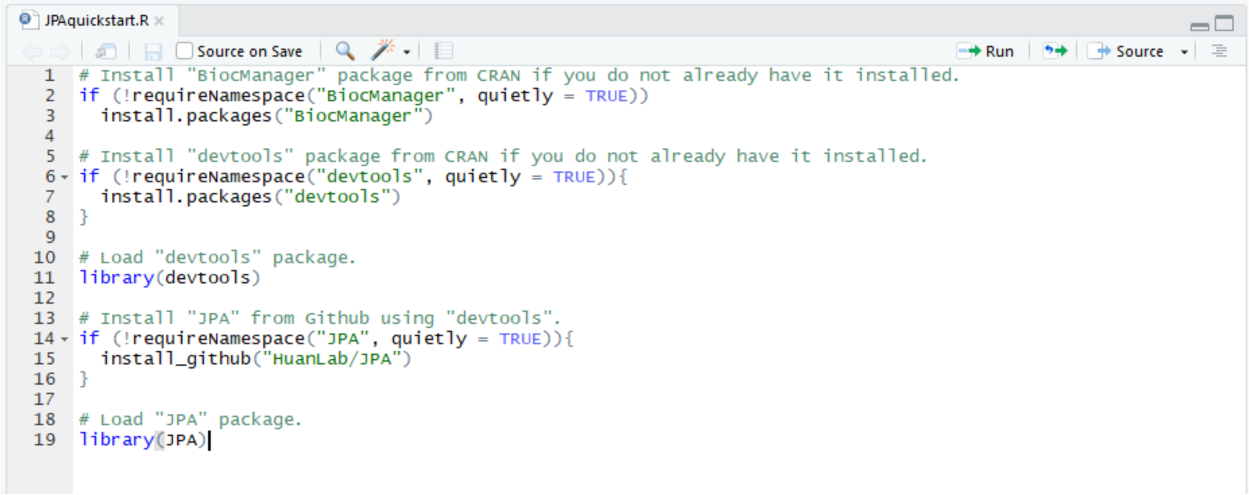


Name	Date modified	Type	Size
UrineOriginal1.mzXML	7/13/2020 1:39 PM	MZXML File	11,956 KB
UrineOriginal2.mzXML	7/13/2020 1:42 PM	MZXML File	11,880 KB
UrineOriginal3.mzXML	7/13/2020 1:45 PM	MZXML File	12,044 KB
UrineOriginal4.mzXML	7/13/2020 1:48 PM	MZXML File	11,995 KB
UrineOriginal5.mzXML	7/13/2020 1:51 PM	MZXML File	11,961 KB

Figure 2. Example of raw data in the folder.

Step 3. Install JPA package.

Open RStudio. Copy the following code block from Part 1 of the user manual (https://github.com/HuanLab/JPA/blob/main/README.md#part-1-introduction-and-installation) into RStudio, and click “Source” to start downloading JPA package (Figure 3).

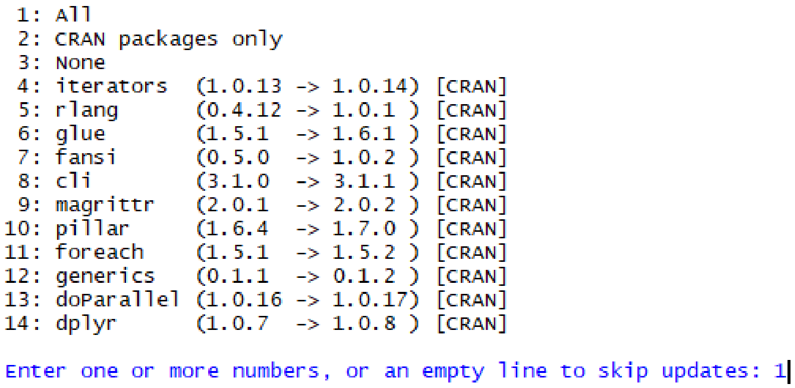


```r
# Install "BiocManager" package from CRAN if you do not already have it installed.
if (!requireNamespace("BiocManager", quietly = TRUE))
  install.packages("BiocManager")

# Install "devtools" package from CRAN if you do not already have it installed.
if (!requireNamespace("devtools", quietly = TRUE)){
  install.packages("devtools")
}

# Load "devtools" package.
library(devtools)

# Install "JPA" from Github using "devtools".
if (!requireNamespace("JPA", quietly = TRUE)){
  install_github("HuanLab/JPA")
}

# Load "JPA" package.
library(JPA)
```

Figure 3. Example of how to install JPA package in RStudio.

Step 4. Update packages (optional).

When prompted to update packages, input “1” and press the “Enter” key (Figure 4) to update all packages, or press “Enter” twice to skip. The package installation is complete when the message “Done (JPA)” shows up in the console.

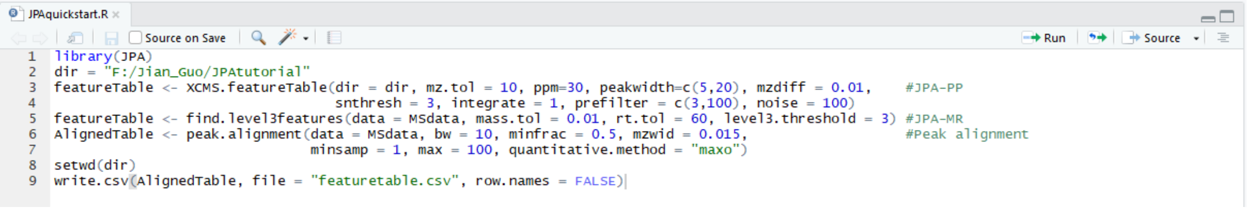


```
 1: All
 2: CRAN packages only
 3: None
 4: iterators  (1.0.13 -> 1.0.14) [CRAN]
 5: rlang      (0.4.12 -> 1.0.1 ) [CRAN]
 6: glue       (1.5.1  -> 1.6.1 ) [CRAN]
 7: fansi      (0.5.0  -> 1.0.2 ) [CRAN]
 8: cli        (3.1.0  -> 3.1.1 ) [CRAN]
 9: magrittr   (2.0.1  -> 2.0.2 ) [CRAN]
10: pillar     (1.6.4  -> 1.7.0 ) [CRAN]
11: foreach    (1.5.1  -> 1.5.2 ) [CRAN]
12: generics   (0.1.1  -> 0.1.2 ) [CRAN]
13: doParallel (1.0.16 -> 1.0.17) [CRAN]
14: dplyr      (1.0.7  -> 1.0.8 ) [CRAN]

Enter one or more numbers, or an empty line to skip updates: 1
```

Figure 4. Package update prompt.

Step 5. Run JPA data processing.

Copy the necessary functions from the user manual into RStudio, and click “Source” to start processing. The functions for JPA-PP and JPA-MR will extract the features from all data files (within a specified directory) into a feature alignment table (Figure 5). Export the feature alignment table as a .csv file. There is an example folder where other workflows are available (https://github.com/HuanLab/JPA/blob/main/Example/Example.R).



```
library(JPA)
dir = "F:/Jian_Guo/JPAtutorial"
featureTable <- XCMS.featureTable(dir = dir, mz.tol = 10, ppm=30, peakwidth=c(5,20), mzdiff = 0.01,     #JPA-PP
                                  snthresh = 3, integrate = 1, prefilter = c(3,100), noise = 100)
featureTable <- find.level3features(data = MSdata, mass.tol = 0.01, rt.tol = 60, level3.threshold = 3) #JPA-MR
AlignedTable <- peak.alignment(data = MSdata, bw = 10, minfrac = 0.5, mzwid = 0.015,                   #Peak alignment
                               minsamp = 1, max = 100, quantitative.method = "maxo")
setwd(dir)
write.csv(AlignedTable, file = "featuretable.csv", row.names = FALSE)
```

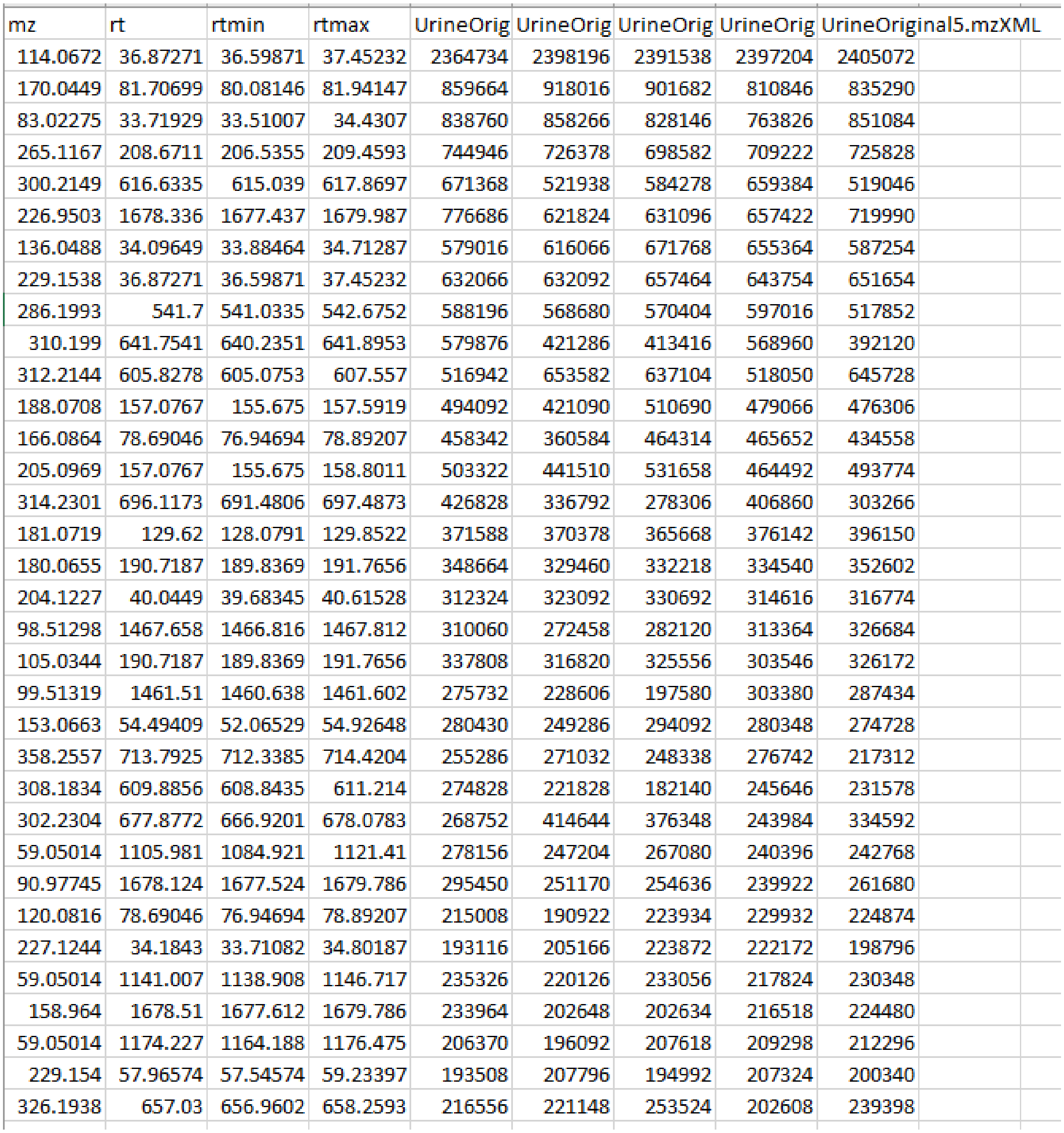
Figure 5. Example workflow of JPA for feature extraction and alignment.

Step 6. Results interpretation.

After the code finishes running, open the feature table file (found in specified directory). Each row is a feature. Column header definitions:

“mz”: the mass to charge ratio of the feature.
“rt”: the retention time of the feature’s peak apex, in seconds.
“rtmin”: the retention time of the feature’s peak left edge, in seconds.
“rtmax”: the retention time of the feature’s peak right edge, in seconds.

The remaining column headers are individual sample names with peak intensities for each feature below. An example feature table is provided in Figure 6.



mz	rt	rtmin	rtmax	UrineOrig	UrineOrig	UrineOrig	UrineOrig	UrineOriginal5.mzXML
114.0672	36.87271	36.59871	37.45232	2364734	2398196	2391538	2397204	2405072
170.0449	81.70699	80.08146	81.94147	859664	918016	901682	810846	835290
83.02275	33.71929	33.51007	34.4307	838760	858266	828146	763826	851084
265.1167	208.6711	206.5355	209.4593	744946	726378	698582	709222	725828
300.2149	616.6335	615.039	617.8697	671368	521938	584278	659384	519046
226.9503	1678.336	1677.437	1679.987	776686	621824	631096	657422	719990
136.0488	34.09649	33.88464	34.71287	579016	616066	671768	655364	587254
229.1538	36.87271	36.59871	37.45232	632066	632092	657464	643754	651654
286.1993	541.7	541.0335	542.6752	588196	568680	570404	597016	517852
310.199	641.7541	640.2351	641.8953	579876	421286	413416	568960	392120
312.2144	605.8278	605.0753	607.557	516942	653582	637104	518050	645728
188.0708	157.0767	155.675	157.5919	494092	421090	510690	479066	476306
166.0864	78.69046	76.94694	78.89207	458342	360584	464314	465652	434558
205.0969	157.0767	155.675	158.8011	503322	441510	531658	464492	493774
314.2301	696.1173	691.4806	697.4873	426828	336792	278306	406860	303266
181.0719	129.62	128.0791	129.8522	371588	370378	365668	376142	396150
180.0655	190.7187	189.8369	191.7656	348664	329460	332218	334540	352602
204.1227	40.0449	39.68345	40.61528	312324	323092	330692	314616	316774
98.51298	1467.658	1466.816	1467.812	310060	272458	282120	313364	326684
105.0344	190.7187	189.8369	191.7656	337808	316820	325556	303546	326172
99.51319	1461.51	1460.638	1461.602	275732	228606	197580	303380	287434
153.0663	54.49409	52.06529	54.92648	280430	249286	294092	280348	274728
358.2557	713.7925	712.3385	714.4204	255286	271032	248338	276742	217312
308.1834	609.8856	608.8435	611.214	274828	221828	182140	245646	231578
302.2304	677.8772	666.9201	678.0783	268752	414644	376348	243984	334592
59.05014	1105.981	1084.921	1121.41	278156	247204	267080	240396	242768
90.97745	1678.124	1677.524	1679.786	295450	251170	254636	239922	261680
120.0816	78.69046	76.94694	78.89207	215008	190922	223934	229932	224874
227.1244	34.1843	33.71082	34.80187	193116	205166	223872	222172	198796
59.05014	1141.007	1138.908	1146.717	235326	220126	233056	217824	230348
158.964	1678.51	1677.612	1679.786	233964	202648	202634	216518	224480
59.05014	1174.227	1164.188	1176.475	206370	196092	207618	209298	212296
229.154	57.96574	57.54574	59.23397	193508	207796	194992	207324	200340
326.1938	657.03	656.9602	658.2593	216556	221148	253524	202608	239398

Figure 6. Example of an output feature table.