

Genome analysis

# faers: An R Package for Seamlessly Bridging the FAERS Database to R and Delivering Unified Pharmacovigilance Workflows

Shixiang Wang<sup>1</sup>, Yun Peng<sup>2</sup>, Zhangyu Wang<sup>1</sup>

<sup>1</sup>Central South University

<sup>2</sup>aaa

To whom correspondence should be addressed. E-mail: qq

Associate Editor: XXX

Received on XXX; revised on XXX; accepted on XXX

## Abstract

**Motivation:** The FDA Adverse Event Reporting System (FAERS) serves as a critical database for monitoring post-marketing drug safety. However, the primary focus on safety signal detection within FAERS has left a significant gap in integrating pharmacovigilance analysis with genetic tools. Thus, we aim to effectively utilize FAERS data to bridge pharmacovigilance with genetic analysis, thereby enhancing the precision of safety assessments and facilitating the development of personalized medicine approaches.

**Results:** We developed the R package faers, which seamlessly connects the FAERS database to the R programming language and provides a unified approach for the seamless execution of pharmacovigilance analysis and the integration of genetic tools in R.

**Availability:** faers is available on CRAN and on GitHub at (<https://github.com/Yunuuuu/faers>).

**Contact:** 212@qq.com

**Supplementary information:** Supplementary data are available at Bioinformatics Online.

## 1 Introduction

The FDA Adverse Event Reporting System (FAERS) is the core database for post-marketing safety monitoring of all approved drugs and therapeutic biologics by the FDA. As required by regulations, it consolidates mandatory adverse event reports submitted by pharmaceutical companies, as well as voluntary first-hand information reported by individuals and healthcare professionals. The data, which has been collected since January 2004 and is continuously updated on a quarterly basis, includes eight types of documents: demographic and administrative information, detailed drug information, indications for use, report sources, drug start and end dates, patient outcomes, reports of therapeutic failure, and adverse events. Its core value is reflected in key areas such as early safety signal detection, risk assessment, and drug labeling revisions, providing direct evidence for the FDA to formulate drug safety policies and risk management measures.

Currently, FAERS research has formed a tripartite landscape of “official—open-source—R ecosystem”: The official path still requires researchers to download quarterly files, deduplicate and clean them, recode with MedDRA, and then manually invoke algorithms. Although the

FAERS Public Dashboard provides zero-code visualization and download, it lacks a batch analysis interface. In the open-source platform, OpenVigil 2.1, by pre-hosting cleaned data from 2004 to 2022, supports one-click output of PRR/ROR after online setting of drug-event-time window, but its cross-platform reproducibility is limited. AERSMine, leveraging Spark to integrate 2.1 million reports, offers interactive differential efficacy forest plots, yet still cannot directly connect to the Bioconductor genetics pipeline. DiAna\_package encapsulates downloading, cleaning, standardization, signaling, and genetics integration into a single R command, being the first to incorporate population genotypes into the signal model, but it leaves gaps in environment locking and full-field integrity, which provides an entry point for the proposed unified faers framework in this paper.

To systematically address the shortcomings of existing tools in terms of reproducibility, integration into the R ecosystem, and in-depth mining of genetic information, we developed the faers package. This package connects the FAERS database to R through a unified pipeline, enabling one-click data download, cleaning, standardization, and signal detection. It also seamlessly integrates with the Bioconductor genetics analysis framework, supporting high-throughput association analysis of population-differentiated polymorphisms and drug pathways,

thereby providing reproducible and scalable decision support for precision medicine.

## 2 Methods

The workflow for pharmacovigilance analysis using the faers package mainly consists of the following four steps: (i) check metadata of FAERS; (ii) downloading and parsing the quarterly FAERS data files; (iii) standardization and deduplication; (iv) conducting pharmacovigilance analysis.

### 2.1 Metadata Inspection

We obtained the FAERS quarterly index with `faers_meta()`, which returns a `data.table` listing year, quarter, period, file URLs and sizes (ASCII/XML, 50–140 MB/quarter) for all releases since 2004Q1. By default (`force = FALSE`, `internal = !curl::has_internet()`), the function first checks a local cache; if absent, it parses the latest index from the FDA site (<https://fis.fda.gov/extensions/FPD-QDE-FAERS/FPD-QDE-FAERS.html>), ensuring reproducibility both online and offline.

### 2.2 Download and Parse quarterly data files from FAERS

Quarterly files are raw data extracted from the FAERS database, typically stored in ASCII or SGML format, and record a series of information related to drug adverse reactions for a specific period, including: `demo`: demographic and administrative details; `drug`: suspect and concomitant drugs; `reac`: adverse reactions coded to MedDRA; `outc`: patient outcomes; `rpsr`: reporter source; `ther`: drug-therapy dates; `indi`: indications (MedDRA-coded diagnoses).

Data acquisition was performed with `faers(years, quarters, dir, compress_dir)`. `years` and `quarters` specify the desired quarters, while `dir` and `compress_dir` indicate the raw and extracted file locations. Internally, `faers()` calls `faers_download()` and `faers_parse()` to preprocess each ASCII file (default format) from the FAERS repository, returning a `FAERSascii` object; multiple quarters are combined via `faers_combine()`. After download, `faers_get()` retrieves any field as a `data.table`.

### 2.3 Standardize and De-duplication

Each quarterly FAERS file stores adverse reactions (`reac`) and drug indications (`indi`) as MedDRA Preferred Terms (PT). To ensure terminological consistency, we first standardise these fields with `faers_standardize(data, meddra_path)`, where `meddra_path` points to the locally unzipped MedDRA ASCII directory. This step embeds the complete MedDRA dictionary (hierarchy + SMQ data) in the `meddra` slot of the returned object. The optional use argument in `faers_meddra()` allows selective retrieval of either component. As a by-product, three columns are automatically appended to `indi` and `reac`: `meddra_hierarchy_idx`, `meddra_hierarchy_from`, `meddra_code` & `meddra_pt`. These columns are generated by `faers_meddra()` and are immediately available via `faers_get("indi")` or `faers_get("reac")`, eliminating manual mapping. Finally, to address FAERS' inherent duplication and incompleteness, we treat two reports as duplicates if they match on drugs and reactions but differ in at most one of: sex, age, reporting country, event date, start date or drug indications. Deduplication is performed with `faers_dedup()`, yielding a clean, case-level dataset ready for downstream analysis.

### 2.4 Pharmacovigilance analysis

Details for Method 2. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum.

Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum.

Details for Method 2. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum.

## 3 Discussion

Discussion of results. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum.

Discussion of results. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum.

Discussion of results. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum.

Discussion of results. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum.

## 4 Conclusion

Anything else? Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum.

Anything else? Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum.

Anything else? Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum.

Anything else? Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum. Introduce your topic. Lorem ipsum ad nauseum.

## Acknowledgements

These should be included at the end of the text and not in footnotes. Please ensure you acknowledge all sources of funding, see funding section below.

Details of all funding sources for the work in question should be given in a separate section entitled ‘Funding’. This should appear before the ‘Acknowledgements’ section.

## Funding

The following rules should be followed:

- The sentence should begin: ‘This work was supported by ...’ -

- The full official funding agency name should be given, i.e. ‘National Institutes of Health’, not ‘NIH’ (full RIN-approved list of UK funding agencies)
- Grant numbers should be given in brackets as follows: ‘[grant number xxxx]’
- Multiple grant numbers should be separated by a comma as follows: ‘[grant numbers xxxx, yyyy]’
- Agencies should be separated by a semi-colon (plus ‘and’ before the last funding agency)
- Where individuals need to be specified for certain sources of funding the following text should be added after the relevant agency or grant number ‘to [author initials]’.

An example is given here: ‘This work was supported by the National Institutes of Health [AA123456 to C.S., BB765432 to M.H.]; and the Alcohol & Education Research Council [hfygr667789].’

Oxford Journals will deposit all NIH-funded articles in PubMed Central. See Depositing articles in repositories – information for authors for details. Authors must ensure that manuscripts are clearly indicated as NIH-funded using the guidelines above.