

# Package: nirs4alldatasets (via r-universe)

June 20, 2026

**Title** Acquire Curated NIRS Reference Datasets ('nirs4all-datasets')

**Version** 0.2.0

**Description** R binding over the 'nirs4all-datasets' C 'ABI' ('n4ds\_\*'):  
resolve a dataset id from the distributable catalog index into  
a version-pinned download contract, fetch the canonical  
'Parquet' ('Dataverse' / 'Zenodo' / 'figshare') with SHA-256  
verification into a local cache, and re-verify a cached  
directory offline. JSON crosses the stable C 'ABI'; analysis of  
the data is left to the host.

**License** MIT + file LICENSE

**Encoding** UTF-8

**NeedsCompilation** yes

**SystemRequirements** Cargo (Rust toolchain), rustc

**RoxygenNote** 7.3.2

**Author** Gregory Beurier [aut, cre]

**Maintainer** Gregory Beurier <gregory.beurier@cirad.fr>

**Config/pak/sysreqs** libclang-dev

**Repository** <https://pamdeveloper.r-universe.dev>

**Date/Publication** 2026-06-19 12:00:08 UTC

**RemoteUrl** <https://github.com/cran/nirs4alldatasets>

**RemoteRef** HEAD

**RemoteSha** 6124f2b3c1ce5de6d5dd9ca3d3f733c5111d26a5

## Contents

n4ds_abi_version . . . . .	2
n4ds_fetch . . . . .	2
n4ds_resolve . . . . .	3
n4ds_verify_cached . . . . .	3

<b>Index</b>	<b>4</b>
--------------	----------

---

n4ds_abi_version	<i>The C ABI version string.</i>
------------------	----------------------------------

---

### Description

The C ABI version string.

### Usage

```
n4ds_abi_version()
```

### Value

The C ABI version as a string.

---

n4ds_fetch	<i>Download + SHA-256-verify a resolved dataset into the cache (JSON status string).</i>
------------	--

---

### Description

Download + SHA-256-verify a resolved dataset into the cache (JSON status string).

### Usage

```
n4ds_fetch(resolved_json, opts_json = "")
```

### Arguments

resolved_json	A resolved contract (from n4ds_resolve) as JSON.
opts_json	Options as JSON: {cache_dir?, token?, instance?, timeout_secs?}; "" for the defaults.

### Value

The fetch status as a JSON string (dir, files:[{name, relpath, path, status}]).

---

n4ds_resolve	<i>Resolve a dataset id against an index JSON into its download contract (JSON string).</i>
--------------	---

---

**Description**

Resolve a dataset id against an index JSON into its download contract (JSON string).

**Usage**

```
n4ds_resolve(index_json, dataset_id)
```

**Arguments**

index_json	The distributable catalog index as a JSON string.
dataset_id	The dataset id to resolve.

**Value**

The resolved download contract as a JSON string (id, tier, instance, doi, dataset\_version, files, origins).

---

n4ds_verify_cached	<i>Re-verify a cached dataset directory against the contract's SHA-256s (JSON report).</i>
--------------------	--

---

**Description**

Re-verify a cached dataset directory against the contract's SHA-256s (JSON report).

**Usage**

```
n4ds_verify_cached(resolved_json, dir)
```

**Arguments**

resolved_json	A resolved contract as JSON.
dir	The cached dataset directory.

**Value**

The verification report as a JSON string (dir, ok, files:[{name, relpath, status}]).

# Index

n4ds\_abi\_version, 2  
n4ds\_fetch, 2  
n4ds\_resolve, 3  
n4ds\_verify\_cached, 3