

Package ‘zipcodeR’

September 18, 2026

Title Data & Functions for Working with US ZIP Codes

Version 0.4.1

Description Make working with ZIP codes in R painless with an integrated dataset of U.S. ZIP codes and functions for working with them. Search ZIP codes by multiple geographies, including state, county, city & across time zones. Also included are functions for relating ZIP codes to Census data, geocoding & distance calculations. New analyses can select an immutable modern data bundle through the next-generation API, while the historical interface remains compatible with version 0.3.5 for reproducible research.

License GPL-3

URL <https://github.com/gavinrozzi/zipcodeR/>,
<https://www.gavinrozzi.com/project/zipcodeR/>

BugReports <https://github.com/gavinrozzi/zipcodeR/issues/>

Encoding UTF-8

LazyData true

Imports rlang, stringr, raster, tidycensus, tidyr, dplyr, jsonlite,
httr, curl, RSQLite, DBI

Depends R (>= 3.5.0)

Suggests openssl, knitr, rmarkdown, markdown, readr, testthat (>= 3.0.0), covr, tibble

VignetteBuilder knitr, rmarkdown

Config/testthat/edition 3

Config/roxygen2/version 8.1.0

NeedsCompilation no

Author Gavin Rozzi [aut, cre] (ORCID: <<https://orcid.org/0000-0002-9969-8175>>)

Maintainer Gavin Rozzi <gr@gavinrozzi.com>

Repository CRAN

Date/Publication 2026-09-18 15:50:02 UTC

Contents

download_comprehensive_data	3
download_zip_data	4
download_zip_data_bundle	4
geocode_zip	5
geocode_zip_ng	5
get_cd	6
get_cd_ng	6
get_tracts	7
get_tracts_ng	7
is_zcta	8
is_zcta_ng	9
normalize_zip	9
normalize_zip_ng	10
read_zip_data_bundle	10
reverse_zipcode	11
reverse_zipcode_ng	11
search_cd	12
search_cd_ng	12
search_city	13
search_city_ng	13
search_county	14
search_county_ng	14
search_fips	15
search_fips_ng	15
search_radius	16
search_radius_ng	16
search_state	17
search_state_ng	17
search_tz	18
search_tz_ng	18
zcta_crosswalk	19
zip_code_db	20
zip_data_provenance	21
zip_data_version	21
zip_distance	22
zip_distance_ng	23
zip_to_cd	23

Index

25

`download_comprehensive_data`*Download the comprehensive ZIP code database*

Description

The bundled `zip_code_db` is the lightweight ("simple") dataset. A much larger companion database with detailed ACS demographic profiles per ZIP code (the "comprehensive" database, ~450 MB SQLite) is published as a checksum-pinned asset of a `zipcodeR` data release rather than shipped in the package. Its public URL, checksum, and SQLite integrity have passed a clean-machine smoke test.

Usage

```
download_comprehensive_data(force = FALSE)
```

Arguments

<code>force</code>	If TRUE, re-download even if a verified copy is cached.
--------------------	---

Details

This function downloads that database, verifies its SHA256 checksum, and, on R 4.0 or newer, caches it under `tools::R_user_dir("zipcodeR", "data")`; later calls `return` the cached path immediately. On older supported R versions the verified file is kept only in the session temporary directory. It never downloads without being called explicitly. For offline use on R 4.0 or newer, copy the file to that user data directory yourself (the expected file name is the asset name from the data release).

Value

Invisibly, the path to the downloaded SQLite database. Query it with DBI/RSQLite, e.g. `DBI::dbConnect(RSQLite::SQLite(), download_comprehensive_data())`.

Examples

```
## Not run:  
path <- download_comprehensive_data()  
  
## End(Not run)
```

download_zip_data	<i>Legacy in-place data downloader</i>
-------------------	--

Description

This function is retained without behavioral changes for reproducibility with zipcodeR 0.3.5. It attempts to modify files inside an installed package and resolves an upstream "latest" release, so it is unsuitable for a pinned research workflow. New work should use [download_zip_data_bundle\(\)](#) or [read_zip_data_bundle\(\)](#) and pass the resulting object explicitly to the `_ng` API. No runtime warning is emitted because warnings are part of the frozen legacy contract.

Usage

```
download_zip_data(force = FALSE)
```

Arguments

force	Boolean, if set to TRUE will force overwrite existing data files with new version
-------	---

Value

Data files needed for package functionality, stored in data directory of package install

Examples

```
## Not run:
download_zip_data()

## End(Not run)
```

download_zip_data_bundle	<i>Download a versioned zipcodeR data bundle</i>
--------------------------	--

Description

Downloads one explicitly requested data release, verifies its checksum, and caches it in the user's data directory on R 4.0 or newer. On older supported R versions, CRAN policy does not permit a package-managed persistent cache, so the verified file is retained only for the current R session. The function never resolves aliases such as "latest", and no lookup function downloads data implicitly.

Usage

```
download_zip_data_bundle(version, force = FALSE)
```

Arguments

version	Exact registered data version, for example "2026.09".
force	Re-download and replace a verified cached copy.

Value

A validated zipcodeR_data_bundle.

geocode_zip	<i>Returns that lat / lon pair of the centroid of a given ZIP code</i>
-------------	--

Description

Returns that lat / lon pair of the centroid of a given ZIP code

Usage

```
geocode_zip(zip_code)
```

Arguments

zip_code	A 5-digit U.S. ZIP code
----------	-------------------------

Value

tibble of lat lon coordinates

Examples

```
geocode_zip("07762")
geocode_zip("90210")
geocode_zip("90210")$lat
geocode_zip("90210")$lng
```

geocode_zip_ng	<i>Geocode ZIP codes using an explicit data bundle</i>
----------------	--

Description

Geocode ZIP codes using an explicit data bundle

Usage

```
geocode_zip_ng(data, zip_code)
```

Arguments

data A zipcodeR_data_bundle.
zip_code ZIP code(s).

Value

One coordinate row per input, preserving order and duplicates.

get_cd	<i>Get all congressional districts for a given ZIP code</i>
--------	---

Description

Get all congressional districts for a given ZIP code

Usage

get_cd(zip_code)

Arguments

zip_code A U.S. ZIP code

Value

a named list of two-digit state code and two digit district code

Examples

```
get_cd("08731")  
get_cd("90210")
```

get_cd_ng	<i>Get congressional districts using an explicit data bundle</i>
-----------	--

Description

Get congressional districts using an explicit data bundle

Usage

get_cd_ng(data, zip_code)

Arguments

data A zipcodeR_data_bundle.
zip_code One five-digit ZIP code.

Value

A named list with parallel state_fips and district vectors.

get_tracts	<i>Get all Census tracts within a given ZIP code</i>
------------	--

Description

Get all Census tracts within a given ZIP code

Usage

```
get_tracts(zip_code)
```

Arguments

zip_code A U.S. ZIP code

Value

tibble of Census tracts and data from Census crosswalk file found for given ZIP code

Examples

```
get_tracts("08731")  
get_tracts("90210")
```

get_tracts_ng	<i>Get Census tracts using an explicit data bundle</i>
---------------	--

Description

Get Census tracts using an explicit data bundle

Usage

```
get_tracts_ng(data, zip_code)
```

Arguments

- data A zipcodeR_data_bundle.
- zip_code One five-digit ZIP or ZCTA code.

Value

A tract crosswalk tibble.

is_zcta	Returns true if the given ZIP code is also a ZIP code tabulation area (ZCTA)
---------	--

Description

Returns true if the given ZIP code is also a ZIP code tabulation area (ZCTA)

Usage

```
is_zcta(zip_code)
```

Arguments

- zip_code A 5-digit U.S. ZIP code

Value

Boolean TRUE or FALSE based upon whether provided ZIP code is a ZCTA by testing whether it exists in the U.S. Census crosswalk data

Examples

```
is_zcta("90210")
is_zcta("99999")
is_zcta("07762")
```

is_zcta_ng	<i>Test ZCTA membership using an explicit data bundle</i>
------------	---

Description

Test ZCTA membership using an explicit data bundle

Usage

```
is_zcta_ng(data, zip_code)
```

Arguments

data	A zipcodeR_data_bundle.
zip_code	ZIP code(s).

Value

A logical vector.

normalize_zip	<i>Normalize ZIP codes</i>
---------------	----------------------------

Description

Normalize ZIP codes

Usage

```
normalize_zip(zipcode)
```

Arguments

zipcode	messy ZIP code to be normalized
---------	---------------------------------

Value

Normalized zipcode

Examples

```
normalize_zip(0008731)
```

normalize_zip_ng	<i>Normalize ZIP codes with corrected boundary behavior</i>
------------------	---

Description

Normalize ZIP codes with corrected boundary behavior

Usage

```
normalize_zip_ng(zipcode)
```

Arguments

zipcode	Character or numeric ZIP values.
---------	----------------------------------

Value

Normalized five-character ZIP values.

read_zip_data_bundle	<i>Read a versioned zipcodeR data bundle</i>
----------------------	--

Description

Reads and validates a previously downloaded zipcodeR data bundle. Bundles are explicit, immutable inputs for the next-generation (_ng) API; reading one never changes the datasets used by the legacy API.

Usage

```
read_zip_data_bundle(path, sha256 = NULL)
```

Arguments

path	Path to a bundle .rds file.
sha256	Optional expected SHA256 checksum. Supplying the checksum is strongly recommended when the file did not come from download_zip_data_bundle() .

Value

An object of class `zipcodeR_data_bundle`.

reverse_zipcode	<i>Given a ZIP code, returns columns of metadata about that ZIP code</i>
-----------------	--

Description

Given a ZIP code, returns columns of metadata about that ZIP code

Usage

```
reverse_zipcode(zip_code)
```

Arguments

zip_code	A 5-digit U.S. ZIP code or chracter vector with multiple ZIP codes
----------	--

Value

A tibble containing data for the ZIP code(s)

Examples

```
reverse_zipcode("90210")
reverse_zipcode("08731")
reverse_zipcode(c("08734", "08731"))
reverse_zipcode("07762")$county
reverse_zipcode("07762")$state
```

reverse_zipcode_ng	<i>Reverse-geocode ZIP codes using an explicit data bundle</i>
--------------------	--

Description

Reverse-geocode ZIP codes using an explicit data bundle

Usage

```
reverse_zipcode_ng(data, zip_code)
```

Arguments

data	A zipcodeR_data_bundle.
zip_code	Five-digit ZIP code(s).

Value

One row per input, in input order, with duplicates preserved. The result adds state_fips (two digits) and county_fips (the full five-digit county identifier) for the database's predominant county.

search_cd

Get all ZIP codes that fall within a given congressional district

Description

Get all ZIP codes that fall within a given congressional district

Usage

```
search_cd(state_fips_code, congressional_district)
```

Arguments

state_fips_code

A two-digit U.S. FIPS code for a state

congressional_district

A two digit number specifying a congressional district in a given

Value

tibble of all congressional districts found for given ZIP code, including state code

Examples

```
search_cd("34", "03")
search_cd("36", "05")
```

search_cd_ng

Search a congressional district using an explicit data bundle

Description

Search a congressional district using an explicit data bundle

Usage

```
search_cd_ng(data, state_fips_code, congressional_district)
```

Arguments

data A zipcodeR_data_bundle.

state_fips_code

One state FIPS code.

congressional_district

One two-digit district code.

Value

A ZIP-to-district tibble.

search_city	<i>Search ZIP codes for a given city within a state</i>
-------------	---

Description

Search ZIP codes for a given city within a state

Usage

```
search_city(city_name, state_abb)
```

Arguments

city_name	Name of major city to search
state_abb	Two-digit code for a U.S. state

Value

tibble of all ZIP code data found for given city

Examples

```
search_city("Spring Lake", "NJ")
search_city("Chappaqua", "NY")
```

search_city_ng	<i>Search a city using an explicit data bundle</i>
----------------	--

Description

Search a city using an explicit data bundle

Usage

```
search_city_ng(data, city_name, state_abb)
```

Arguments

data	A zipcodeR_data_bundle.
city_name	City name.
state_abb	Two-letter state abbreviation.

Value

A tibble from the selected data vintage.

search_county	<i>Search ZIP codes for a county</i>
---------------	--------------------------------------

Description

Search ZIP codes for a county

Usage

```
search_county(county_name, state_abb, ...)
```

Arguments

county_name	Name of a county within a U.S. state
state_abb	Two-digit code for a U.S. state
...	if the parameter similar = TRUE, then send the parameter max.distance to the base function agrep. Default is 0.1.

Value

tibble of all ZIP codes for given county name

Examples

```
middlesex <- search_county("Middlesex", "NJ")
alameda <- search_county("alameda", "CA")
search_county("ST BERNARD", "LA", similar = TRUE)$zipcode
```

search_county_ng	<i>Search a county using an explicit data bundle</i>
------------------	--

Description

Search a county using an explicit data bundle

Usage

```
search_county_ng(data, county_name, state_abb, ...)
```

Arguments

data	A zipcodeR_data_bundle.
county_name	County name.
state_abb	Two-letter state abbreviation.
...	Set similar = TRUE and optionally max.distance for approximate matching.

Value

A tibble from the selected data vintage.

search_fips	<i>Returns all ZIP codes found within a given FIPS code</i>
-------------	---

Description

Returns all ZIP codes found within a given FIPS code

Usage

```
search_fips(state_fips, county_fips)
```

Arguments

state_fips A U.S. FIPS code
 county_fips A 1-3 digit county FIPS code (optional)

Value

tibble of Census tracts and data from Census crosswalk file found for given ZIP code

Examples

```
search_fips("34")
search_fips("34", "03")
search_fips("34", "3")
search_fips("36", "003")
```

search_fips_ng	<i>Search FIPS codes using an explicit data bundle</i>
----------------	--

Description

Search FIPS codes using an explicit data bundle

Usage

```
search_fips_ng(data, state_fips, county_fips)
```

Arguments

data A zipcodeR_data_bundle.
 state_fips One state FIPS code.
 county_fips Optional one- to three-digit county FIPS code.

Value

A tibble from the selected data vintage.

search_radius	<i>Search for ZIP codes that are within a given radius from a point</i>
---------------	---

Description

Search for ZIP codes that are within a given radius from a point

Usage

```
search_radius(lat, lng, radius = 1)
```

Arguments

lat	latitude
lng	longitude
radius	distance to search in miles, set by default to 1

Value

a tibble containing the ZIP code(s) within the provided radius and distance from the provided coordinates in miles

Examples

```
## Not run:
search_radius(39.9, -74.3, 10)

## End(Not run)
```

search_radius_ng	<i>Search within a radius using an explicit data bundle</i>
------------------	---

Description

Search within a radius using an explicit data bundle

Usage

```
search_radius_ng(data, lat, lng, radius = 1)
```

Arguments

data	A zipcodeR_data_bundle.
lat, lng	Query coordinates.
radius	Radius in miles.

Value

A tibble of ZIP codes and haversine distances.

search_state	<i>Search for ZIP codes located within a given state</i>
--------------	--

Description

Search for ZIP codes located within a given state

Usage

```
search_state(state_abb)
```

Arguments

state_abb	Two-digit code representing a U.S. state
-----------	--

Value

tibble of all ZIP codes for each state code defined in state_abb

Examples

```
search_state("NJ")
search_state(c("NJ", "NY", "CT"))
```

search_state_ng	<i>Search a state using an explicit data bundle</i>
-----------------	---

Description

Search a state using an explicit data bundle

Usage

```
search_state_ng(data, state_abb)
```

Arguments

data A zipcodeR_data_bundle.
state_abb Two-letter state abbreviation(s).

Value

A tibble grouped in query order; repeated states repeat their rows.

search_tz	<i>Search all ZIP codes located within a given timezone</i>
-----------	---

Description

Search all ZIP codes located within a given timezone

Usage

search_tz(tz)

Arguments

tz Timezone

Value

tibble of all ZIP codes found for given timezone

Examples

```
eastern <- search_tz("Eastern")  
pacific <- search_tz("Mountain")
```

search_tz_ng	<i>Search a timezone using an explicit data bundle</i>
--------------	--

Description

Search a timezone using an explicit data bundle

Usage

search_tz_ng(data, tz)

Arguments

`data` A zipcodeR_data_bundle.
`tz` Timezone value(s).

Value

A tibble grouped in query order; repeated timezones repeat their rows.

zcta_crosswalk	<i>ZCTA to Census Tract (2010) Crosswalk</i>
----------------	--

Description

A dataset containing the relationships between ZIP code tabulation areas (ZCTA) and Census Tracts. This contains selected variables from the official crosswalk file.

Usage

```
zcta_crosswalk
```

Format

A data frame with 148897 rows and 3 variables:

ZCTA5 2010 ZIP Code Tabulation Area

TRACT 2010 Census Tract Code

GEOID Concatenation of 2010 State, County, and Tract

Details

This is the exact dataset distributed by zipcodeR 0.3.5. It is intentionally frozen so legacy calls remain reproducible; newer versioned relationships are available only through an explicit data bundle and the `_ng` API.

Source

<https://www.census.gov/geographies/reference-files/time-series/geo/relationship-files.html>

zip_code_db

*ZIP Code Database***Description**

A dataset containing detailed information for U.S. ZIP codes

Usage

zip_code_db

Format

A data frame with 41877 rows and 24 variables:

zipcode 5 digit U.S. ZIP code

zipcode_type ZIP category such as Standard, PO Box, or Unique

major_city Major city serving the ZIP code

post_office_city City of post office serving the ZIP code

common_city_list List of common cities represented by the ZIP code

county Name of county containing the ZIP code

state Two-digit state code for ZIP code location

lat Latitude of the centroid for the ZIP code

lng Longitude of the centroid for the ZIP code

timezone Timezone of the ZIP code

radius_in_miles Radius of the ZIP code in miles

area_code_list List of area codes for telephone numbers within this ZIP code

population Total population of the ZIP code

population_density Population density of the ZIP code (persons per square mile)

land_area_in_sqmi Area of the land contained within the ZIP code in square miles

water_area_in_sqmi Area of the waters contained within the ZIP code in square miles

housing_units Number of housing units within the ZIP code

occupied_housing_units Number of housing units within the ZIP code

median_home_value Median home price within the ZIP code

median_household_income Median household income within the ZIP code

bounds_west Bounding box coordinates

bounds_east Bounding box coordinates

bounds_north Bounding box coordinates

bounds_south Bounding box coordinates

Details

This is the exact 41,877-row database distributed by zipcodeR 0.3.5, including its historical values and limitations. It is intentionally frozen for research reproducibility. Use an explicit data bundle with `_ng` functions for corrected behavior and a pinned source manifest.

Source

https://github.com/MacHu-GWU/uszipcode-project/files/5183256/simple_db.log

zip_data_provenance	<i>Inspect the provenance supplied with a zipcodeR data bundle</i>
---------------------	--

Description

Inspect the provenance supplied with a zipcodeR data bundle

Usage

```
zip_data_provenance(bundle, dataset = NULL, key = NULL)
```

Arguments

bundle	A zipcodeR_data_bundle.
dataset	Optional dataset name ("zip_code_db", "zcta_crosswalk", or "zip_to_cd").
key	Optional record key (ZIP or ZCTA) to select.

Value

A provenance data frame combining source/method records with record-level quality status and reasons.

zip_data_version	<i>Report zipcodeR data-version metadata</i>
------------------	--

Description

With no argument, reports the immutable data bundled with the legacy API. Given a downloaded data bundle, reports that bundle's metadata and verified checksum.

Usage

```
zip_data_version(x = NULL)
```

Arguments

x	NULL for the legacy bundled data, or a zipcodeR_data_bundle object.
---	---

Value

A named metadata list.

zip_distance	<i>Calculate the distance between two ZIP codes in miles</i>
--------------	--

Description

Calculate the distance between two ZIP codes in miles

Usage

```
zip_distance(zipcode_a, zipcode_b, lonlat = TRUE, units = "miles")
```

Arguments

zipcode_a	First vector of ZIP codes
zipcode_b	Second vector of ZIP codes
lonlat	lonlat argument to pass to raster::pointDistance() to select method of distance calculation. Default is TRUE to calculate distance over a spherical projection. FALSE will calculate the distance in Euclidean (planar) space.
units	Specify which units to return distance calculations in. Choices include meters or miles.

Value

a data.frame containing a column for each ZIP code and a new column containing the distance between the two columns of ZIP code

Examples

```
zip_distance("08731", "08901")
```

zip_distance_ng	<i>Calculate ZIP-to-ZIP distance using an explicit data bundle</i>
-----------------	--

Description

Calculate ZIP-to-ZIP distance using an explicit data bundle

Usage

```
zip_distance_ng(
  data,
  zipcode_a,
  zipcode_b,
  lonlat = TRUE,
  units = c("miles", "meters")
)
```

Arguments

data	A zipcodeR_data_bundle.
zipcode_a, zipcode_b	ZIP vectors.
lonlat	Use great-circle haversine distance; FALSE uses a planar equirectangular approximation.
units	"miles" or "meters".

Value

A data frame of paired ZIP codes and distances.

zip_to_cd	<i>ZIP Code to Congressional District Relationship File</i>
-----------	---

Description

A dataset containing mappings between ZIP codes and congressional districts

Usage

```
zip_to_cd
```

Format

A data frame with 45914 rows and 2 variables:

ZIP 5 digit U.S. ZIP code

CD Four digit congressional district code (State FIPS code + district number)

Details

This is the exact pre-2020 HUD-USPS-derived mapping shipped in zipcodeR 0.3.5. It is frozen for legacy reproducibility and should not be interpreted as a current redistricting vintage. Current authoritative ZCTA relationships are available in explicit data bundles.

Source

Historical HUD-USPS-derived snapshot archived at <https://github.com/gavinrozzi/zipcodeR-data>

Index

* datasets

- zcta_crosswalk, [19](#)
- zip_code_db, [20](#)
- zip_to_cd, [23](#)

download_comprehensive_data, [3](#)
download_zip_data, [4](#)
download_zip_data_bundle, [4](#)
download_zip_data_bundle(), [4](#), [10](#)

geocode_zip, [5](#)
geocode_zip_ng, [5](#)
get_cd, [6](#)
get_cd_ng, [6](#)
get_tracts, [7](#)
get_tracts_ng, [7](#)

is_zcta, [8](#)
is_zcta_ng, [9](#)

normalize_zip, [9](#)
normalize_zip_ng, [10](#)

read_zip_data_bundle, [10](#)
read_zip_data_bundle(), [4](#)
reverse_zipcode, [11](#)
reverse_zipcode_ng, [11](#)

search_cd, [12](#)
search_cd_ng, [12](#)
search_city, [13](#)
search_city_ng, [13](#)
search_county, [14](#)
search_county_ng, [14](#)
search_fips, [15](#)
search_fips_ng, [15](#)
search_radius, [16](#)
search_radius_ng, [16](#)
search_state, [17](#)
search_state_ng, [17](#)
search_tz, [18](#)

search_tz_ng, [18](#)

zcta_crosswalk, [19](#)
zip_code_db, [20](#)
zip_data_provenance, [21](#)
zip_data_version, [21](#)
zip_distance, [22](#)
zip_distance_ng, [23](#)
zip_to_cd, [23](#)