

# Package ‘cdcfluview’

April 2, 2020

**Type** Package

**Encoding** UTF-8

**Title** Retrieve Flu Season Data from the United States Centers for Disease Control and Prevention ('CDC') 'FluView' Portal

**Version** 0.9.1

**Date** 2020-04-01

**Maintainer** Bob Rudis <bob@rud.is>

**Description** The 'U.S.' Centers for Disease Control and Prevention (CDC) maintain a portal <<https://gis.cdc.gov/grasp/fluview/fluportaldashboard.html>> for accessing state, regional and national influenza statistics as well as mortality surveillance data. The web interface makes it difficult and time-consuming to select and retrieve influenza data. Tools are provided to access the data provided by the portal's underlying 'API'.

**URL** <https://github.com/hrbrmstr/cdcfluview>

**BugReports** <https://github.com/hrbrmstr/cdcfluview/issues>

**License** MIT + file LICENSE

**LazyData** true

**Suggests** testthat, covr

**Depends** R (>= 3.2.0)

**Imports** httr, tools, dplyr, jsonlite, tibble, stats, utils, sf, xml2, purrr, readr, MMWRweek, units (>= 0.4-6)

**RoxygenNote** 7.1.0

**NeedsCompilation** no

**Author** Bob Rudis [aut, cre] (<<https://orcid.org/0000-0001-5670-2640>>),  
Craig McGowan [ctb] (<<https://orcid.org/0000-0002-6298-0185>>),  
JJ Chen [ctb] (<<https://orcid.org/0000-0001-8482-8398>>),  
Sebastian Meyer [ctb] (<<https://orcid.org/0000-0002-1791-9449>>),  
James Turtle [ctb] (<<https://orcid.org/0000-0003-0735-7769>>),  
Andrew Bates [ctb]

**Repository** CRAN

**Date/Publication** 2020-04-02 12:00:17 UTC

## R topics documented:

age_group_distribution	2
cdcfluvview	3
cdc_basemap	3
census_regions	4
geographic_spread	5
get_flu_data	5
get_hosp_data	6
get_state_data	7
get_weekly_flu_report	8
hhs_regions	9
hospitalizations	9
ilinet	10
ili_weekly_activity_indicators	11
mmwr_id_map	12
mmwr_week	13
mmwr_weekday	13
mmwr_week_to_date	14
pi_mortality	15
state_data_providers	16
surveillance_areas	16
who_nrevss	17

<b>Index</b>	<b>19</b>
--------------	-----------

---

age\_group\_distribution

*Age Group Distribution of Influenza Positive Tests Reported by Public Health Laboratories*

---

### Description

Retrieves the age group distribution of influenza positive tests that are reported by public health laboratories by influenza virus type and subtype/lineage. Laboratory data from multiple seasons and different age groups is provided.

### Usage

```
age_group_distribution(years = NULL)
```

### Arguments

years	a vector of years to retrieve data for (i.e. 2014 for CDC flu season 2014-2015). CDC has data for this API going back to 1997. Default value (NULL) means retrieve <b>all</b> years. NOTE: if you happen to specify a 2-digit season value (i.e. 57 == 2017-2018) the function is smart enough to retrieve by season ID vs convert that to a year.
-------	--

## References

- [CDC FluView Portal](#)
- [AGD IPT Portal](#)

## Examples

```
agd <- age_group_distribution(years=2015)
```

---

cdcfluvview	<i>Retrieve Flu Season Data from the United States Centers for Disease Control and Prevention ('CDC') 'FluView' Portal</i>
-------------	--

---

## Description

The U.S. Centers for Disease Control (CDC) maintains a portal <http://gis.cdc.gov/grasp/fluview/fluportaldashboard.html> for accessing state, regional and national influenza statistics as well as Mortality Surveillance Data. The Flash interface makes it difficult and time-consuming to select and retrieve influenza data. This package provides functions to access the data provided by the portal's underlying API.

## Author(s)

Bob Rudis (bob@rud.is)

---

cdc_basemap	<i>Retrieve CDC U.S. Basemaps</i>
-------------	-----------------------------------

---

## Description

The CDC FluView application uses a composite basemaps of coverage areas within the United States that elides and scales Alaska, Hawaii, Puerto Rico & the Virgin Islands and some further provide elided and scaled breakouts for New York City and the District of Columbia.

This function retrieves the given shapefile, projects to EPSG:5069 and returns it as an sf (simple features) object.

## Usage

```
cdc_basemap(  
  basemap = c("national", "hhs", "census", "states", "spread", "surv")  
)
```

**Arguments**

basemap            select the CDC basemap. One of:

- "national": outline of the U.S. + AK, HI, PR + VI
- "hhs": outline of the U.S. + HHS Region Outlines + AK, HI, PR + VI
- "census": outline of the U.S. + Census Region Outlines + AK, HI, PR + VI
- "states": outline of the U.S. + State Outlines + AK, HI, PR + VI
- "spread": outline of the U.S. + State Outlines + AK, HI, PR + VI & Guam
- "surv": outline of the U.S. + State Outlines + AK, HI, PR + VI

**Note**

These are just the basemaps. You need to pair it with the data you wish to visualize.

**Examples**

```
## Not run:
plot(cdc_basemap("national"))

## End(Not run)
```

---

census_regions	<i>Census Region Table</i>
----------------	----------------------------

---

**Description**

This dataset contains the states belonging to the (presently) 4 U.S. Census regions in "long" format. It consists of a `data.frame` with the following columns:

- `region`: the official Census region name (e.g. "East")
- `state`: state belonging to the region

**Usage**

```
data(census_regions)
```

**Format**

A data frame with 51 rows and 2 variables

**Note**

Last updated 2015-08-09.

**References**

<https://www.cdc.gov/std/stats12/images/CensusMap.png>

---

geographic_spread	<i>State and Territorial Epidemiologists Reports of Geographic Spread of Influenza</i>
-------------------	--

---

### Description

State and Territorial Epidemiologists Reports of Geographic Spread of Influenza

### Usage

```
geographic_spread(years = NULL)
```

### Arguments

**years** a vector of years to retrieve data for (i.e. 2014 for CDC flu season 2014-2015). CDC has data for this API going back to 2003 and up until the current, active flu season. Default value (NULL) means retrieve **all** years. NOTE: if you happen to specify a 2-digit season value (i.e. 57 == 2017-2018) the function is smart enough to retrieve by season ID vs convert that to a year.

### Examples

```
## Not run:  
gs <- geographic_spread()  
  
## End(Not run)
```

---

get_flu_data	<i>Retrieves state, regional or national influenza statistics from the CDC (deprecated)</i>
--------------	---

---

### Description

Uses the data source from the **CDC FluView** and provides flu reporting data as either a single data frame or a list of data frames (depending on whether either WHO NREVSS or ILINet (or both) is chosen).

### Usage

```
get_flu_data(  
  region = "hhs",  
  sub_region = 1:10,  
  data_source = "ilinet",  
  years = as.numeric(format(Sys.Date(), "%Y"))  
)
```

**Arguments**

region	one of "hhs", "census", "national", "state"
sub_region	depends on the region_type. For "national", the sub_region should be NA. For "hhs", should be a vector between 1:10. For "census", should be a vector between 1:9. For "state", should be a vector of state/territory names or "all".
data_source	either of "who" (for WHO NREVSS) or "ilinet" or "all" (for both)
years	a vector of years to retrieve data for (i.e. 2014 for CDC flu season 2014-2015). Default value is the current year and all years values should be > 1997

**Details**

A lookup table between HHS regions and their member states/territories is provided in [hhs\\_regions](#).

**Value**

If only a single data\_source is specified, then a single data.frame is returned, otherwise a named list with each data.frame is returned.

**Note**

There is often a noticeable delay when making the API request to the CDC. This is not due to a large download size, but the time it takes for their servers to crunch the data. Wrap the function call in `httr::with_verbose` if you would like to see what's going on.

---

get_hosp_data	<i>Retrieves influenza hospitalization statistics from the CDC (depre- cated)</i>
---------------	---

---

**Description**

Uses the data source from the [CDC FluView](#) and provides influenza hospitalization reporting data as a data frame.

**Usage**

```
get_hosp_data(
  area = "flusurvnet",
  age_group = "overall",
  years = as.numeric(format(Sys.Date(), "%Y")) - 1
)
```

**Arguments**

area	one of "flusurvnet", "eip", "ihsp", or two digit state abbreviation for an individual site. Exceptions are New York - Albany ("nya") and New York - Rochester ("nyr")
age_group	a vector of age groups to pull data for. Possible values are: "overall", "0-4y", "5-17y", "18-49y", "50-64y", "65+y".
years	a vector of years to retrieve data for (i.e. 2014 for CDC flu season 2014-2015). Default value is the current year and all years values should be $\geq 2009$

**Value**

A single data.frame.

**Note**

There is often a noticeable delay when making the API request to the CDC. This is not due to a large download size, but the time it takes for their servers to crunch the data. Wrap the function call in `httr::with_verbose` if you would like to see what's going on.

**Examples**

```
## Not run:
# All of FluSurv-NET, 50-64 years old, 2010/11-2014/15 flu seasons
hosp <- get_hosp_data("flusurvnet", "50-64y", years=2010:2014)

## End(Not run)
```

---

get_state_data	<i>Retrieves state/territory-level influenza statistics from the CDC (deprecated)</i>
----------------	---

---

**Description**

Uses the data source from the CDC' State-levelFluView <https://gis.cdc.gov/grasp/fluview/main.html> and provides state flu reporting data as a single data frame.

This function provides similar data to [get\\_weekly\\_flu\\_report](#) but provides more metadata about the reporting sources and has access to more historical information.

**Usage**

```
get_state_data(years = as.numeric(format(Sys.Date(), "%Y")))
```

**Arguments**

years	a vector of years to retrieve data for (i.e. 2014 for CDC flu season 2014-2015). Default value is the current year and all years values should be $\geq 2008$
-------	---

**Value**

A data.frame of state-level data for the specified seasons (also classed as cdcstatedata)

**Note**

There is often a noticeable delay when making the API request to the CDC. This is not due to a large download size, but the time it takes for their servers to crunch the data. Wrap the function call in `httr::with_verbose` if you would like to see what's going on.

**Examples**

```
state_data <- get_state_data(2014)
## Not run:
gsd <- get_state_data(c(2013, 2014))
gsd <- get_state_data(2010:2014)

## End(Not run)
```

---

`get_weekly_flu_report` *Retrieves (high-level) weekly (XML) influenza surveillance report from the CDC*

---

**Description**

The CDC publishes a **weekly influenza report** detailing high-level flu activity per-state. They also publish a data file (see References) of historical report readings. This function reads that XML file and produces a long `data_frame` with the historical surveillance readings.

This function provides similar data to `get_state_data` but without the reporting source metadata and a limit on the historical flu information.

**Usage**

```
get_weekly_flu_report()
```

**Value**

`tbl_df` (also classed with `cdcweeklyreport`) with six columns: `year`, `week_number`, `state`, `color`, `label`, `subtitle`

**References**

<https://www.cdc.gov/flu/weekly/flureport.xml>

**Examples**

```
## Not run:
wfr <- get_weekly_flu_report()

## End(Not run)
```



---

hhs_regions	<i>HHS Region Table</i>
-------------	-------------------------

---

**Description**

This dataset contains the names, numbers, regional offices for-, and states/territories belonging to the (presently) 10 HHS U.S. regions in "long" format. It consists of a `data.frame` with the following columns:

- `region`: the official HHS region name (e.g. "Region 1")
- `region_number`: the associated region number
- `regional_office`: the HHS regional office for the entire region
- `state_or_territory`: state or territory belonging to the region

**Usage**

```
data(hhs_regions)
```

**Format**

A data frame with 59 rows and 4 variables

**Note**

Last updated 2015-08-09.

**References**

<https://www.hhs.gov/about/agencies/iea/regional-offices/index.html>

---

hospitalizations	<i>Laboratory-Confirmed Influenza Hospitalizations</i>
------------------	--

---

**Description**

Laboratory-Confirmed Influenza Hospitalizations

**Usage**

```
hospitalizations(  
  surveillance_area = c("flusurv", "eip", "ihsp"),  
  region = "all",  
  years = NULL  
)
```

**Arguments**

surveillance_area	one of "flusurv", "eip", or "ihsp"
region	Using "all" mimics selecting "Entire Network" from the CDC FluView application drop down. Individual regions for each surveillance area can also be selected. Use <code>surveillance_areas()</code> to see a list of valid sub-regions for each surveillance area.
years	a vector of years to retrieve data for (i.e. 2014 for CDC flu season 2014-2015). CDC has data for this API going back to 2009 and up until the <i>previous</i> flu season. Default value (NULL) means retrieve <b>all</b> years. NOTE: if you happen to specify a 2-digit season value (i.e. 56 == 2016-2017) the function is smart enough to retrieve by season ID vs convert that to a year.

**References**

- [Hospital Portal](#)

**Examples**

```

hosp_fs <- hospitalizations("flusurv", years=2015)
## Not run:
hosp_eip <- hospitalizations("eip")
hosp_ihsp <- hospitalizations("ihsp")

## End(Not run)

```

---

ilinet

*Retrieve ILINet Surveillance Data*


---

**Description**

The CDC FluView Portal provides in-season and past seasons' national, regional, and state-level outpatient illness and viral surveillance data from both ILINet (Influenza-like Illness Surveillance Network) and WHO/NREVSS (National Respiratory and Enteric Virus Surveillance System).

**Usage**

```
ilinet(region = c("national", "hhs", "census", "state"), years = NULL)
```

**Arguments**

region	one of "national", "hhs", "census", or "state"
years	a vector of years to retrieve data for (i.e. 2014 for CDC flu season 2014-2015). CDC has data for this API going back to 1997. Default value (NULL) means retrieve <b>all</b> years. NOTE: if you happen to specify a 2-digit season value (i.e. 57 == 2017-2018) the function is smart enough to retrieve by season ID vs convert that to a year.

## Details

This function retrieves current and historical ILINet surveillance data for the identified region.

## References

- [CDC FluView Portal](#)
- [ILINet Portal](#) (Login required)
- [WHO/NREVSS](#)

## Examples

```
national_ili <- ilinet("national", years = 2017)
## Not run:
hhs_ili <- ilinet("hhs")
census_ili <- ilinet("census")
state_ili <- ilinet("state")

all_ili <- suppressWarnings(
  suppressMessages(purrr::map_df(c("national", "hhs", "census", "state"), ilinet))

## End(Not run)
```

---

ili\_weekly\_activity\_indicators

*Retrieve weekly state-level ILI indicators per-state for a given season*

---

## Description

Retrieve weekly state-level ILI indicators per-state for a given season

## Usage

```
ili_weekly_activity_indicators(years = NULL)
```

## Arguments

**years** a vector of years to retrieve data for (i.e. 2014 for CDC flu season 2014-2015). CDC has data for this API going back to 2008 and up until the current, active flu season. Default value (NULL) means retrieve **all** years. NOTE: if you happen to specify a 2-digit season value (i.e. 57 == 2017-2018) the function is smart enough to retrieve by season ID vs convert that to a year.

**Note**

These statistics use the proportion of outpatient visits to healthcare providers for influenza-like illness to measure the ILI activity level within a state. They do not, however, measure the extent of geographic spread of flu within a state. Therefore, outbreaks occurring in a single city could cause the state to display high activity levels.

Data collected in ILINet may disproportionately represent certain populations within a state, and therefore may not accurately depict the full picture of influenza activity for the whole state.

All summary statistics are based on either data collected in ILINet, or reports from state and territorial epidemiologists. Differences in the summary data presented by CDC and state health departments likely represent differing levels of data completeness with data presented by the state likely being the more complete.

**References**

- [ILI Activity Indicator Map Portal](#)

**Examples**

```
## Not run:
iwai <- ili_weekly_activity_indicators(2016)

## End(Not run)
```

---

mmwrid\_map

*MMWR ID to Calendar Mappings*

---

**Description**

The CDC uses a unique "Morbidity and Mortality Weekly Report" identifier for each week that starts at 1 (Ref: < <https://www.cdc.gov/mmwr/preview/mmwrhtml/su6004a9.htm>>). This data frame consists of 4 columns:

- `wk_start`: Start date (Sunday) for the week (Date)
- `wk_end`: End date (Saturday) for the week (Date)
- `year_wk_num`: The week of the calendar year
- `mmwrid`: The unique MMWR identifier These can be "left-joined" to data provided from the CDC to perform MMWR identifier to date mappings.

**Format**

A data frame with 4,592 rows and 4 columns

---

mmwr_week	<i>Convert a Date to an MMWR day+week+year</i>
-----------	--

---

**Description**

This is a reformat and re-export of a function in the MMWRweek package. It provides a snake case version of its counterpart, produces a tibble

**Usage**

```
mmwr_week(x)
```

**Arguments**

x a vector of Date objects or a character vector in YYYY-mm-dd format.

**Value**

data frame (tibble)

**Examples**

```
mwk <- mmwr_week(Sys.Date())
```

---

mmwr_weekday	<i>Convert a Date to an MMWR weekday</i>
--------------	--

---

**Description**

This is a reformat and re-export of a function in the MMWRweek package. It provides a snake case version of its counterpart, produces a factor of weekday names (Sunday-Saturday).

**Usage**

```
mmwr_weekday(x, abbr = FALSE)
```

**Arguments**

x a vector of Date objects or a character vector in YYYY-mm-dd format.  
abbr (logical) if TRUE, return abbreviated weekday names, otherwise full weekday names (see Note).

**Value**

ordered factor

**Note**

Weekday names are explicitly mapped to "Sunday-Saturday" or "Sun-Sat" and do not change with your locale.

**Examples**

```
mwday <- mmwr_weekday(Sys.Date())
```

---

mmwr_week_to_date	<i>Convert an MMWR year+week or year+week+day to a Date object</i>
-------------------	--

---

**Description**

This is a reformat and re-export of a function in the MMWRweek package. It provides a snake case version of its counterpart and produces a vector of Date objects that corresponds to the input MMWR year+week or year+week+day vectors. This also adds some parameter checking and cleanup to avoid exceptions.

**Usage**

```
mmwr_week_to_date(year, week, day = NULL)
```

**Arguments**

year, week, day Year, week and month vectors. All must be the same length unless day is NULL.

**Value**

vector of Date objects

**Examples**

```
mwd <- mmwr_week_to_date(2016,10,3)
```

**Description**

The National Center for Health Statistics (NCHS) collects and disseminates the Nation's official vital statistics. NCHS collects death certificate data from state vital statistics offices for virtually all deaths occurring in the United States. Pneumonia and influenza (P&I) deaths are identified based on ICD-10 multiple cause of death codes.

NCHS Mortality Surveillance System data are presented by the week the death occurred at the national, state, and HHS Region levels. Data on the percentage of deaths due to P&I on a national level are released two weeks after the week of death to allow for collection of enough data to produce a stable percentage. States and HHS regions with less than 20% of the expected total deaths (average number of total deaths reported by week during 2008-2012) will be marked as insufficient data. Collection of complete data is not expected at the time of initial report, and a reliable percentage of deaths due to P&I is not anticipated at the U.S. Department of Health and Human Services region or state level within this two week period. The data for earlier weeks are continually revised and the proportion of deaths due to P&I may increase or decrease as new and updated death certificate data are received by NCHS.

The seasonal baseline of P&I deaths is calculated using a periodic regression model that incorporates a robust regression procedure applied to data from the previous five years. An increase of 1.645 standard deviations above the seasonal baseline of P&I deaths is considered the "epidemic threshold," i.e., the point at which the observed proportion of deaths attributed to pneumonia or influenza was significantly higher than would be expected at that time of the year in the absence of substantial influenza-related mortality. Baselines and thresholds are calculated at the national and regional level and by age group.

**Usage**

```
pi_mortality(coverage_area = c("national", "state", "region"), years = NULL)
```

**Arguments**

coverage_area	coverage area for data (national, state or region)
years	a vector of years to retrieve data for (i.e. 2014 for CDC flu season 2014-2015). CDC has data for this API going back to 2009 and up until the current, active flu season. Default value (NULL) means retrieve <b>all</b> years. NOTE: if you happen to specify a 2-digit season value (i.e. 57 == 2017-2018) the function is smart enough to retrieve by season ID vs convert that to a year.

**Note**

Queries for "state" and "region" are not necessarily as "instantaneous" as other API endpoints and can near or over 30s retrieval delays.

For consistency with `ilinet` columns indicating the week now contain 'week' instead of the previously abbreviation 'wk'.

## References

- [Pneumonia and Influenza Mortality Surveillance Portal](#)

## Examples

```
## Not run:
ndf <- pi_mortality()
sdf <- pi_mortality("state")
rdf <- pi_mortality("region")

## End(Not run)
```

---

`state_data_providers` *Retrieve metadata about U.S. State CDC Provider Data*

---

## Description

Retrieve metadata about U.S. State CDC Provider Data

## Usage

```
state_data_providers()
```

## Examples

```
sdp <- state_data_providers()
```

---

`surveillance_areas` *Retrieve a list of valid sub-regions for each surveillance area.*

---

## Description

Retrieve a list of valid sub-regions for each surveillance area.

## Usage

```
surveillance_areas()
```

## Examples

```
sa <- surveillance_areas()
```



---

 who\_nrevss

 Retrieve WHO/NREVSS Surveillance Data
 

---

### Description

The CDC FluView Portal provides in-season and past seasons' national, regional, and state-level outpatient illness and viral surveillance data from both ILINet (Influenza-like Illness Surveillance Network) and WHO/NREVSS (National Respiratory and Enteric Virus Surveillance System).

### Usage

```
who_nrevss(region = c("national", "hhs", "census", "state"), years = NULL)
```

### Arguments

region	one of "national", "hhs", "census", or "state"
years	a vector of years to retrieve data for (i.e. 2014 for CDC flu season 2014-2015). CDC has data for this API going back to 1997. Default value (NULL) means retrieve <b>all</b> years. NOTE: if you happen to specify a 2-digit season value (i.e. 57 == 2017-2018) the function is smart enough to retrieve by season ID vs convert that to a year.

### Details

This function retrieves current and historical WHO/NREVSS surveillance data for the identified region.

### Value

list of data frames identified by

- combined\_prior\_to\_2015\_16
- public\_health\_labs
- clinical\_labs

### Note

HHS, Census and State data retrieval is not as "instantaneous" as their ILINet counterparts.

Also, beginning for the 2015-16 season, reports from public health and clinical laboratories are presented separately in the weekly influenza update. This is the reason why a list of data frames is returned.

**ALSO** The new CDC API seems to be missing some public health lab data fields.

**References**

- [CDC FluView Portal](#)
- [ILINet Portal](#) (Login required)
- [WHO/NREVSS](#)

**Examples**

```
## Not run:
national_who <- who_nrevss("national")
hhs_who <- who_nrevss("hhs")
census_who <- who_nrevss("census")
state_who <- who_nrevss("state")

## End(Not run)
```

# Index

## \*Topic **datasets**

- [census\\_regions](#), 4
  - [hhs\\_regions](#), 9
- [age\\_group\\_distribution](#), 2
- [cdc\\_basemap](#), 3
- [cdcfluview](#), 3
- [census\\_regions](#), 4
- [geographic\\_spread](#), 5
- [get\\_flu\\_data](#), 5
- [get\\_hosp\\_data](#), 6
- [get\\_state\\_data](#), 7, 8
- [get\\_weekly\\_flu\\_report](#), 7, 8
- [hhs\\_regions](#), 6, 9
- [hospitalizations](#), 9
- [ili\\_weekly\\_activity\\_indicators](#), 11
- [ilinet](#), 10, 16
- [mmwr\\_week](#), 13
- [mmwr\\_week\\_to\\_date](#), 14
- [mmwr\\_weekday](#), 13
- [mmwrid\\_map](#), 12
- [pi\\_mortality](#), 15
- [state\\_data\\_providers](#), 16
- [surveillance\\_areas](#), 16
- [surveillance\\_areas\(\)](#), 10
- [who\\_nrevss](#), 17